SUPPORT POOL
OF EXPERTS PROGRAMME

Training curriculum on AI and data protection

Fundamentals of Secure AI Systems with Personal Data

by Dr. Enrico GLEREAN

As part of the SPE programme, the EDPB may commission contractors to provide reports and tools on specific topics.

The views expressed in the deliverables are those of their authors and they do not necessarily reflect the official position of the EDPB. The EDPB does not guarantee the accuracy of the information included in the deliverables. Neither the EDPB nor any person acting on the EDPB's behalf may be held responsible for any use that may be made of the information contained in the deliverables.

Some excerpts may be redacted or removed from the deliverables as their publication would undermine the protection of legitimate interests, including, inter alia, the privacy and integrity of an individual regarding the protection of personal data in accordance with Regulation (EU) 2018/1725 and/or the commercial interests of a natural or legal person.

Table of contents

Document submitted in December 2024, updated in April 2025

Preface

This book outlines a **training curriculum** for cybersecurity professionals working with **personal data** and **artificial intelligence** (AI) in the European Union (EU). The current landscape of AI regulation in the EU (and in the rest of the world) is constantly evolving, and it can be overwhelming for organisations to understand what the practical steps would be to create and deploy modern AI systems and AI models that are compatible with legal obligations, ethical principles, and state of the art cybersecurity and data privacy.

This book is a resource for both **learners** and **teachers**. This book is an open book commissioned by the European Data Protection Board and the Greek Data Protection Authority as part of the Support Pool of Experts program. The book is released under CC-BY-SA license. You are free to re-use the content of this book, as long as you cite the original source and share your derivative work with a similar license.

License

Introduction

What is this book about?

This book is a **training curriculum**: a structured course for understanding secure deployment of artificial intelligent systems that have been trained with or are processing personal data. Each chapter has clearly defined learning goals, exercises, and notes for instructors.

Who is this book for?

While this book is intended for a technical audience who might already be familiar with some aspects of the AI deployment life-cycle and data privacy techniques, the book also tries to stimulate the discussions between different type of experts to gain a comprehensive view of AI systems development in the context of personal data. The security expert, the data scientist, the system administrator, the data governance expert, the legal expert of an organisation will all learn from each-other and this book helps facilitating their mutual discussions to set a solid foundation and shared vocabulary when working together in deploying an AI system. Management boards will also find insights on the technological challenges that various team members might be facing when developing and deploying AI systems. The book has no formulas and no code examples to make it accessible to a wide audience.

Book structure

The book is structured in **5 modules**. Each module corresponds to a single day workshop with an instructor (facilitator) who can cover the topics and can challenge the learners with tasks and questions. Each module is divided in chapters. At the end of each chapter there are exercises, quizzes, and suggestions for in-class activities. The book can be also used as self-learning material. The **first module sets the basic terminology of AI systems and their life-cycle**. The **second module focuses on the data privacy aspects of the AI systems life-cycle**: data collection, privacy enhancing technologies, and in general all the (personal) data flows before putting the system in production. **Module 3 focuses on the development of the AI system**, considering good coding practices, secure sandboxes, and security testing of AI systems under development. **Module 4 considers the deployment of an AI system, monitoring, sustainability, and decommissioning**. **Module 5 focuses on checklists for auditing with a series of use cases** and a deeper overview of the technological, legal, and ethical challenges in the field.

Further considerations

This book is trying to provide fundamentals on practical solutions for companies developing or acquiring AI systems, but as of the date of this book (March 2025) there are still many unknown elements on how AI regulation is developing, how AI compliance and liability will be implemented in practice, with novel insights, peer-reviewed articles, policies, guidelines that are being released almost daily. The open-source book model will hopefully enable active development of this curriculum throughout the months, making it a useful and reusable resource for anyone interested in the intersection of data privacy, AI, and cybersecurity.

Notice on the tools used for writing this book

The following process and tools were used to create this book. From the initial requirements from the Greek data protection authority, the author has generated *user stories*, i.e. statements from a potential learner in the form of “I would like to learn about …”. The initial user stories (n = 52) were then passed to ChatGPT to further augment and expand the number of stories resulting in 116 user stories. The stories were then rated by the reviewers and it was agreed to focus on a subset of 70 user stories. The author then drafted the structure of the book/workshops based on the chosen user stories. All content was drafted by the author, except for the following content which was first synthesised by ChatGPT and then revised by the author: learning goals for each chapter, final quiz with multiple choice questions at the end of each chapter. Furthermore language and structure in many paragraphs were improved using AI tools such as ChatGPT, and DeepL. The book is written using the Quarto https://quarto.org/ open source software tool, and the book source code is made available. Zotero was used to manage the references. All images were made manually the author using the open source software Inkscape. The final version of the book was analysed with Turn-it-in to verify that the book contains no plagiarised text (this is a risk that can happen when asking AI tools to rephrase some of the paragraphs).

Contributing to the book

The book is released as an open source book under CC-BY-SA 4.0 license. Since the intersection between AI, privacy, and security is a field of active development, update requests to the book can be submitted by opening an issue in the github repository. New contributions can happen through the git mechanism of “pull requests” (or merge requests). Forks of the book will need to explicitly mention the changes that have been done, following the requirements set by the CC-BY-SA license. We encourage new training materials stemming from this book such as slides, videos, tests, flash cards, code examples.

1. Elements of Artificial Intelligence, Data Protection, Cybersecurity

Learning outcomes

After completing this chapter you will:

- Acquire the basic vocabulary of AI, data privacy, and cybersecurity.
- Understand the fundamental principles of AI, its applications, and how it compares to more traditional deterministic approaches.
- Understand the compromise between AI performance, data protection, cybersecurity.

In this chapter we will cover the basic principles and definitions for the three domains that we are studying in this course: artificial intelligence, data protection in the EU, cybersecurity.

1.1 What is Artificial Intelligence?

Multiple definitions of **Artificial Intelligence (AI)** exist since the “Meetings of the Minds” workshop in 1956 – considered to be the first recorded definition of AI. In general AI is an umbrella term for a series of computational methods and systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages. AI is defined as the science and technology of computing systems that can autonomously solve complex inferential problems, often mimicking tasks that humans can perform but computers traditionally could not, such as image recognition or decision-making. AI can be approached through various methods, including rule-based systems where domain experts manually craft sets of rules for the computer to follow. In contrast, Machine Learning (ML), originally a sub-field of AI, focuses on developing algorithms that enable computers to automatically learn patterns and predictive models from data, making AI systems more adaptive and scalable. For example, instead of programming explicit rules to recognise handwritten digits, ML can train a model using thousands of labelled images to learn the patterns on its own. While ML is a crucial tool within AI, AI encompasses much more than ML, including areas like natural language processing (NLP), robotics, and expert systems. Meanwhile, ML can also extend beyond AI applications, being fundamental in fields like predictive analytics, where it may be used without invoking broader AI concepts. Figure 1.1 visually summarises the various sub-branches of AI, with particular focus on ML.

1.1.1 AI from data: data science versus machine learning

While not all AI is built from data, the most useful and popular AI uses are all based on learning patterns from data: whether it is our smartphone that has learned visual features from our face to unlock itself, or a Large Language Model (LLMs) in AI systems like *ChatGPT* that has learned textual patterns from a huge amount of books, transcripts, and scraped content, all distilled in its deep neural network.

How is AI working in practice? Compared to other data science deterministic methods, AI uses the methodological approaches of *Machine Learning* (ML) to derive patterns from the training data. The *training* data are a dataset that is curated accordingly so that the machine learning algorithm can learn the pattern in the dataset and adapt its *model weights* to perform a certain task.

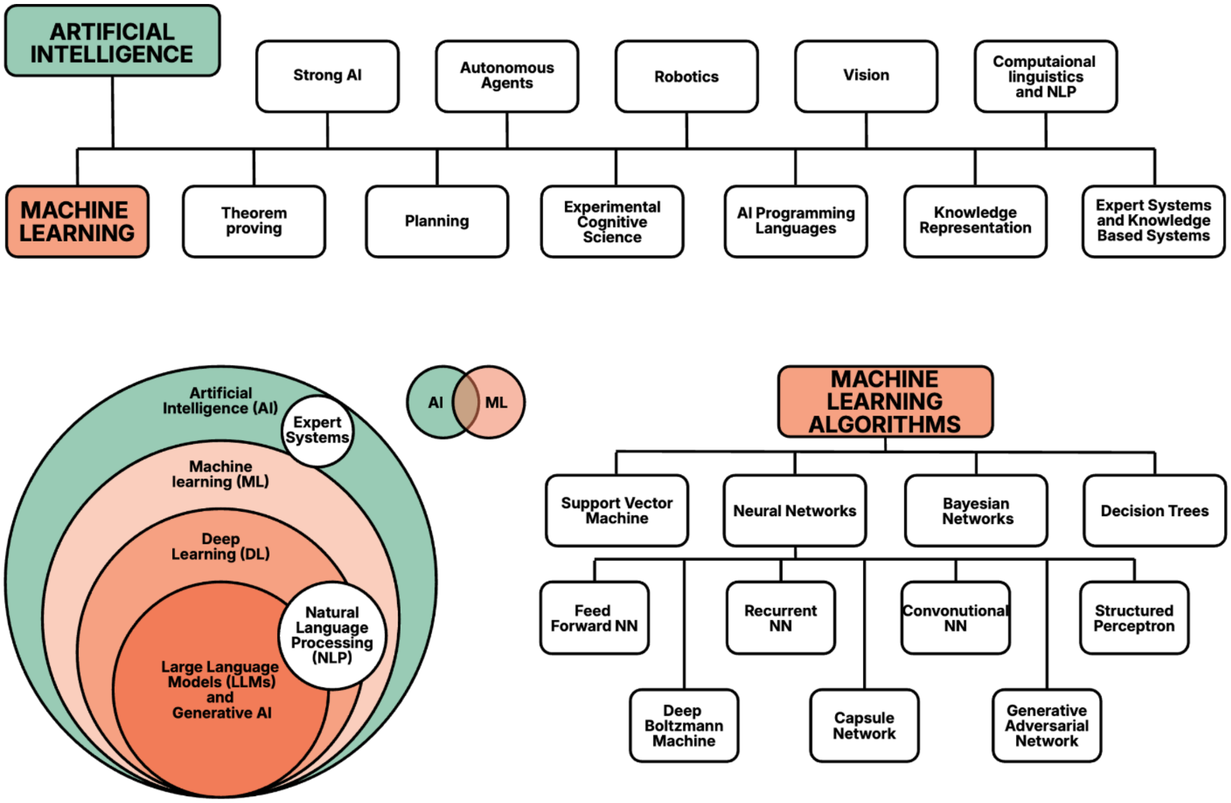


Figure 1.1: ***Definitions of AI*** *- The top panel is adapted from Aliferis and Simon (2024). The bottom left panel is adapted from "A New Dawn for Public Employment Services. OECD" (2024). The bottom right panel is adapted from @International Organization for Standardization (2022a). It is important to notice that the OECD definition considers ML as a subset fully contained inside AI. In practice it is not as simple as that, and a more nuanced description is provided by Raschka (2024) where AI is broader than ML, and ML can be used in systems that are not AI systems (this is visually summarised in the small Venn diagram of AL and ML not fully overlapping)*

For example let's imagine that we have a dataset of the systolic blood pressure for a set of patients. For the sake of simplicity let's just assume that those with systolic blood pressure over 140 mm Hg are labelled with "high risk" (of cardiovascular disease) and those with blood pressure lower than 140 mm Hg are labelled with "normal" (things are more complicated than this with cardiovascular diseases, but this is just a simple example). With a data science approach, we could create a rule based system (deterministic) so that, in pseudocode:

```
IF BP >= 140 THEN
    label = "high risk"
ELSE
    label = "normal"
END IF
```

This deterministic system would just process each data subject and generate the label according to the rule. The system would be robust and the rule would be explainable and readable by a person. The ML approach to the same problem would be to choose a specific ML algorithm (in this case a *classifier*) so that it can be trained from existing data so that when a certain numerical value of systolic blood pressure and labels are given, the ML algorithm tries to adapt its model weights. Then, when a new dataset is used – the *test dataset* – the AI system is able to *predict* that a subject with blood pressure higher or equal to 140 mm Hg is receiving a label of “high risk”.

Exercise 1.1: Discuss with your peer

Suppose that the training dataset does not have any data for blood pressure measurements equal to 138, 139, 140 mm Hg: Would it still find the same rule as the deterministic system?

Exercise 1.1: Solution

The ML model could set 138 as the threshold for separating normal and high-blood pressure subjects. The *bias* present in the data, is reflected in the potential diagnostic errors produced by this AI system.

In this toy example we immediately understood a core limitation of ML approaches: if the data in the training set does not cover all the possible cases that we want to sample, the robustness and accuracy of the ML output might not be the desired one.

Let’s now imagine to extend our example so that, rather than a single measurement for each individual, a collection of data are used to train the ML model: for example we might have information about the weight, height, history of heart conditions, blood exam values, diet habits, sport habits, smoking, and so on. The ML learning algorithm could use all these data and learn how they can be associated with the final label “high risk” or “normal” as in the example before. Due to the richness of the data, it would become more difficult to explain how the machine learning model operates, but possibly it could perform better in defining a more precise diagnosis for the patient, rather than the human doctor who would need to evaluate all these data according to their knowledge and experience. To understand how AI systems learn from data, let’s explore the various machine learning methods that form the backbone of modern AI. It is important to keep in mind that AI includes approaches beyond ML predictions methods, with techniques such as rule-based systems, symbolic reasoning, and search algorithms that also try to mimic human intelligence. Some examples of these are provided in chapter 11.

1.1.2 Machine Learning Methods

Machine learning methods provide the techniques to teach computers how to learn from data. There are three broad categories of ML methods (International Organization for Standardization 2022b):

1. **Supervised Learning**: the AI model is trained using labelled data: each input data item is paired with the correct output, and the model learns to predict the output from the input. For example in a dataset of images of faces labelled with “happy” or “sad”, the model learns how to classify new pictures based on this labeled data. Commonly used algorithms: linear or logistic regression, decision trees, support vector machines, neural networks. Examples of applications: spam detection, house price prediction (predicting house prices based on features like size and location), medical diagnosis (classifying whether a patient has a certain disease based on symptoms)
2. **Unsupervised Learning**: the AI model is trained on data without labeled outcomes; the model tries to find patterns and structures in the data on its own. For example in a dataset of customer purchases from an online store, the model can group similar customers together based on their purchase patterns. Commonly used algorithms: clustering (e.g. K-means, hierarchical), principal component analysis, autoencoders (neural networks). Examples of applications: customer segmentation, anomaly

detection, market basket analysis (finding associations between products, like which items are often bought together).

3. **Reinforcement Learning**: Reinforcement learning is a type of machine learning where an agent learns to make decisions by interacting with an environment. The agent takes actions and receives feedback in the form of rewards or penalties, with the goal of maximizing its cumulative reward over time. Unlike supervised learning, where the model learns from labeled data, reinforcement learning focuses on learning from the consequences of actions. For example, a robot might learn to navigate a maze by receiving positive rewards for moving toward the exit and negative rewards for hitting walls. Commonly used algorithms: Q-learning, Deep Q-Networks (DQN), Policy Gradient Methods, Actor-Critic Methods. Examples of applications: Game AI (e.g., AlphaGo, which learned to play Go), robotics (e.g., robots learning to walk or pick up objects), autonomous vehicles (learning to navigate roads safely), recommendation systems (e.g., learning to recommend content based on user interaction).

There are other categories of ML methods that can be blurred between the three categories above:

4. **Semi-supervised Learning**: Semi-supervised learning is a blend of supervised and unsupervised learning. In this approach, the AI model is trained using a dataset that contains a small amount of labeled data and a large amount of unlabeled data. The labeled data helps the model learn initial patterns, while the unlabeled data allows it to generalize better to new examples. For example, a model might be trained on a large dataset of images where only a few images are labeled with categories like “dog” or “cat,” and the rest are unlabeled. The model learns to identify patterns in both labeled and unlabeled images, improving its accuracy without needing a fully labeled dataset. Commonly used algorithms: Graph-based algorithms, self-training, deep neural networks (e.g., semi-supervised GANs). Examples of applications: Medical image analysis (where labeling every image is costly), web page classification, speech recognition.

5. **Self-supervised Learning**: Self-supervised learning is a type of learning where the model generates its own labels from the data, typically by predicting parts of the data from other parts. It is often used as a way to pre-train models on large, unlabeled datasets, which can then be fine-tuned on smaller labeled datasets. For instance, in natural language processing, models like GPT (Generative Pre-trained Transformer) are pre-trained using self-supervised learning by predicting the next word in a sentence. In this approach, no manually labeled data is needed, as the model creates its own task based on the data itself (e.g., predicting missing words in a sentence). Commonly used algorithms: Transformers (e.g., GPT, BERT), contrastive learning algorithms. Examples of applications: Pre-training large language models like GPT for text generation, BERT for sentence understanding, image representation learning where parts of an image are masked and the model is trained to reconstruct them.

Exercise 1.2: Discuss with your peer

Consider an AI system with a ML model that is able to classify emotions from pictures of faces. Do you think this AI system is actually able to truly infer emotions? Are there any privacy risks?

Exercise 1.2: Solution

This is a very difficult example and touches on an important privacy risk of AI systems: “Phrenology / Physiognomy” (inferring personality, social, and emotional attributes about an individual from their physical attributes, Lee et al. (2024)). AI systems performing *biometric categorization* (assigning natural persons to specific categories on the basis of their biometric data, AI Act Recital 16) have been criticised for being against the fundamental right to dignity and for being pseudoscience Andrews, Smart, and Birhane (2024).

Notes

- *For the learner*: if you are interested in these topics, please start with the references linked in this section.
- *For the instructor*: this is a good task also for a homework based on a reading assignment with one or more of the references mentioned here.

The broad categories of ML paradigms are implemented with *algorithms*. What makes AI more challenging compared to other data science deterministic approaches, is the fact that due to the complexity of certain algorithms, developers and users are not able to *explain* how the algorithm comes to produce a certain output. **Explainable AI** is an important topic of research, and in the context of AI regulation and risk, algorithms have been categorised depending on the level of explainability they can provide. The less explainable algorithms, the higher the chances that the AI system could be considered **high-risk** or **prohibited** according to the AI Act (more on this in Chapter 2).

1.1.3 AI systems and AI models

In the sections above we have often mentioned *AI Systems* and *AI models*. It is important to understand that they are not the same thing: you might be developing an AI model but not deploying it into an AI system, and viceversa you might be building an AI system although you are just reusing existing AI models (or systems) that you did not develop yourself.

Let’s have a look at the definition in the AI Act article 3:

> ‘AI system’ means a machine-based system that is designed to operate with varying levels of autonomy, and that may exhibit adaptiveness after deployment, and that, for explicit or implicit objectives, infers, from the input it receives, how to generate outputs such as predictions, content, recommendations, or decisions that can influence physical or virtual environments.

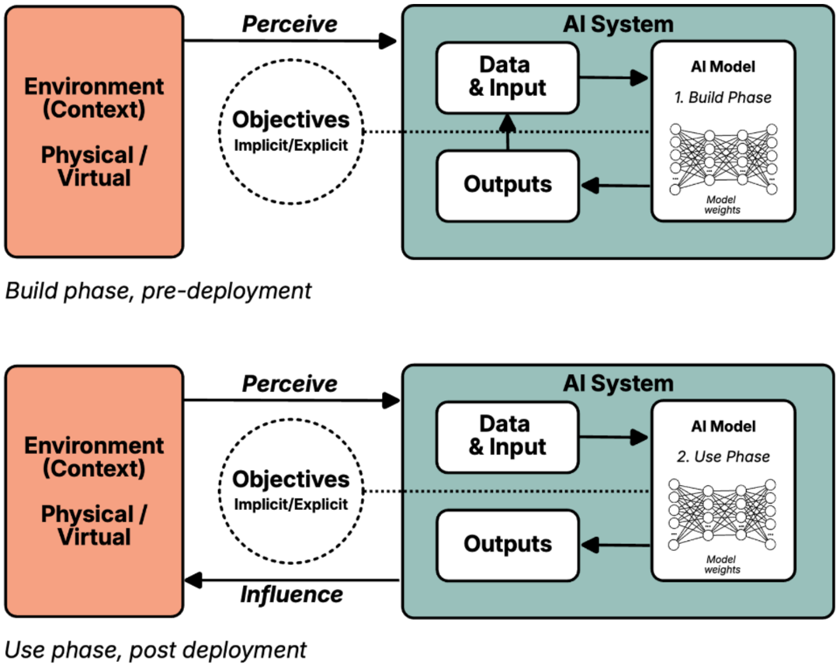
An AI system has three main components: input (perception), processing (reasoning/decision making), and output (actuation). An AI system can be used by a user or by another application to perform a certain task based on the AI model weights and software implementation. To grasp the difference intuitively, the AI model is like the engine of a car: it can be very powerful, but without the rest of the car (AI system) it is just a large static dataset.

The AI Act does not define AI models, however we can reuse the definition from ISO standards:

> ‘AI model’: an AI (or ML) model is a mathematical construct that generates an inference or prediction, based on input data or information (see International Organization for Standardization 2022b, sec. 3.2.11)

The AI Act however has a definition for General Purpose AI models (GPAI) and we will get back to this in chapter 2.

The AI Act definitions are largely based on OECD (2024) and they are visually summarised in Figure 2.1.



Build phase, pre-deployment

Use phase, post deployment

Figure 1.2: ***AI systems and AI models*** *- The figure is adapted from OECD (2024).*

1.1.4 Real use cases of AI systems

There are various types of AI systems, some can involve more privacy risks than others especially if they are used to process certain categories of personal data. In chapter 2 we will go deeper on the intersection between AI systems, ethics, privacy, and risk assessment. Here some possible use cases of AI systems:

- **AI for Code Assistance**: An AI-powered tool helps software developers by suggesting code snippets, debugging errors, and optimizing performance. It uses pre-learned programming patterns to assist coders but doesn’t require any personal information from users.

- **Personalized Movie Recommendations**: A streaming service uses AI to analyze your viewing history and recommend new movies or TV shows based on your preferences. The system tracks your behavior over time, including which movies you watch, skip, or search for.

- **AI-Powered Fitness Tracker**: A wearable fitness tracker uses AI to monitor your physical activity, track your heart rate, and provide personalized workout recommendations. It collects data about your daily routines, health metrics, and even your location when you’re running or cycling.

- **AI for Smart Homes**: An AI system integrated into a smart home controls lighting, temperature, and security features. It learns from your daily habits to optimize energy usage and improve comfort. The system has access to your home environment, routines, and potentially records video and audio within the house for security.

- **AI-Driven Virtual Assistant**: A virtual assistant, like Siri or Alexa, uses AI to handle voice commands, schedule appointments, and answer questions. It continuously listens to your conversations, analyzes your voice patterns, and has access to your calendar, emails, and personal notes to provide more tailored services.

- **AI in Autonomous Vehicles**: Autonomous vehicles rely on AI to navigate roads, avoid obstacles, and make real-time decisions. They collect data from external sensors and cameras, but they also analyze data about passengers, such as their location, destination, and sometimes even their in-car conversations.

- **AI in Predictive Healthcare**: AI systems used in hospitals can predict patient outcomes based on vast amounts of medical data, including diagnosis, treatment history, and genetics. These systems can help doctors make decisions but require extensive access to sensitive medical records, test results, and personal health information.

- **AI in Facial Recognition Surveillance**: A facial recognition system powered by AI is used for security purposes in public spaces. It scans faces in real-time to identify individuals, matching them against a database of known people. These systems have access to biometric data and can track individuals' movements, raising significant privacy and ethical concerns.

Exercise 1.3: Discuss with your peer

Do you see a pattern in the order on how the systems are presented?

Exercise 1.3: Solution

The AI systems are shown with increasing order of risk towards individuals. The last AI system is one of the **prohibited AI systems** according to the AI Act.

1.1.5 Learn more about the basics of AI and ML

This section only briefly covered the elements of AI. The learner who wants to explore further on the topic should consider taking specialised online courses like the MOOC Elements of AI. Sebastian Raschka has great learning materials on the topic of ML (with Python), please see the Courses page on Sebastian Raschka's website. Next, we will introduce the basic principles of data protection in the context of the EU General Data Protection Regulation.

1.2 What is personal data?

In the previous section you familiarised with the concepts of Artificial Intelligence and Machine Learning, how they can be compared with more deterministic approaches. The second pillar of our course is Personal Data (PD). Let's have a look at the definition of personal data according to the General Data Protection Regulation article 4:

> "'personal data' means any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person;"

Personal data is then **any** data about a living individual, however, depending on the type of data, we can introduce some more useful definitions.

1.2.1 Principles of the GDPR

The General Data Protection Regulation (GDPR) is structured around seven fundamental principles, as outlined in Articles 5–11, that govern how personal data should be processed and protected. These principles are essential for ensuring that personal data is handled in a lawful, fair, and transparent manner while also safeguarding individuals' rights and freedoms. For cybersecurity technologists and AI system providers, particularly those dealing with high-risk AI systems as defined under the AI Act, adhering to these principles is crucial throughout the lifecycle of personal data, from collection to storage, processing, and eventual deletion or anonymization. Below is a table summarizing the key principles, along with examples of how they apply to AI system providers and deployers.

Principles of the GDPR

GDPR Principle	Definition	Example for an AI system provider/deployer
Lawfulness, Fairness, and Transparency	Personal data should be processed legally, fairly, and in a transparent manner.	Ensure users are informed about AI system data collection and processing through transparent privacy notices, while the purpose of the system is legal.
Purpose Limitation	Data should be collected for specific, explicit, and legitimate purposes and not processed in a manner incompatible with those purposes.	Limit AI model access to data strictly relevant to its function, preventing unintended uses.
Data Minimization	Data processing should be adequate, relevant, and limited to what is necessary for the intended purposes.	Design AI models to operate on minimal personal data, reducing unnecessary collection.
Accuracy	Personal data should be accurate and kept up to date. Inaccurate data should be erased or rectified without delay.	Incorporate mechanisms in the AI system to flag and correct outdated or incorrect data in real-time.
Storage Limitation	Data should be kept in a form that permits identification of data subjects for no longer than necessary.	Set automated retention periods to delete or anonymize data used by the AI system after its purpose is fulfilled.
Integrity and Confidentiality (Security)	Personal data should be processed in a manner that ensures appropriate security, including protection against unauthorized access and data breaches.	Implement strong encryption and access controls within AI systems to protect against data breaches.
Accountability	The data controller is responsible for, and must be able to demonstrate, compliance with the GDPR principles.	Maintain thorough documentation of data processing steps within the AI system, ready for audits or compliance checks.

1.2.2 The Rights of the Data Subject

Under the GDPR, data subjects have specific rights regarding their personal data, which must be respected by organizations processing this data, including AI system providers and deployers. These rights ensure individuals have control over their personal data and can exercise these rights at any point during the data processing lifecycle. For AI systems, particularly those classified as high-risk under the AI Act, it’s crucial to integrate mechanisms that respect these rights, as non-compliance can lead to significant legal and financial penalties. The relevant GDPR articles (Art. 15–22) outline these rights, which include access to personal data, correction of inaccuracies, and the ability to object to certain forms of data processing. The following table summarizes the key rights and provides short examples of how AI system providers can implement measures to comply.

Rights of the data subjects

Data Subject Right	**Description**	**Example for an AI system provider/deployer**
Right to Access (Art. 15)	Individuals have the right to access their personal data and obtain information about how it is being processed.	Provide users with a secure portal to view the personal data processed by the AI model and detailed information about how it is used in decision-making.
Right to Rectification (Art. 16)	Individuals can request the correction of inaccurate or incomplete personal data.	Allow users to easily request corrections to personal data used by the AI model, ensuring timely updates to the data and retraining of models if necessary.
Right to Erasure (“Right to be Forgotten”) (Art. 17)	Individuals can request the deletion of their personal data in certain circumstances.	Implement a system-wide data deletion process that permanently removes personal data from AI models and databases upon valid user requests.
Right to Restriction of Processing (Art. 18)	Individuals can request the restriction of their data processing under certain conditions.	Integrate a functionality that pauses the processing of personal data within the AI system while retaining the data securely until the restriction is lifted.
Right to Data Portability (Art. 20)	Individuals have the right to receive their personal data in a structured, commonly used format and transmit it to another controller.	Provide an export feature allowing users to download their data used by the AI system in common formats like JSON or CSV.
Right to Object (Art. 21)	Individuals can object to the processing of their personal data, including for direct marketing purposes.	Offer a simple opt-out mechanism in the AI system that halts data processing when a user objects, especially for activities like profiling or targeted advertising.
Right not to be Subject to Automated Decision-Making (Art. 22)	Individuals have the right not to be subject to decisions based solely on automated processing, including profiling, that produce legal or similarly significant effects.	Implement a human-in-the-loop process, ensuring that users can request a manual review of any high-impact decisions made by the AI system.

Exercise 1.4: Discussion in classroom

Do LLMs contain personal data?

Please discuss with the instructor or with your peers the following questions:

- Do LLMs contain personal data and how can it be proved?
- How would the rights of the data subjects be implemented with LLMs?

Note for the instructor: This can also be a homework assignment with further readings such as the "EDPB opinion on the use of personal data for the development and deployment of AI models" (European Data Protection Board (EDPB) 2024) or "Discussion Paper: Large Language Models and Personal Data" (The Hamburg Commissioner for Data Protection and Freedom of Information 2024).

1.2.3 Legal Bases of the GDPR

Under the GDPR, the processing of personal data must have a valid legal basis. These legal bases are outlined in **Article 6** of the GDPR and provide the lawful grounds under which data can be collected, processed, and stored. For AI system providers and deployers, especially those handling high-risk systems, it's crucial to understand which legal basis applies to their operations. Depending on the purpose of the AI system, different legal grounds may be used, such as user consent or legitimate interest. However, not all legal bases may be suitable, particularly for high-risk systems involving sensitive data. Below is a table that summarizes each legal basis and provides examples of how they can—or cannot—be applied by AI providers or deployers.

Legal bases of the GDPR

Legal Basis	Description	Example for an AI system provider/deployer
Consent (Art. 6(1)(a))	The data subject has given explicit consent for their personal data to be processed for specific purposes.	Obtain user consent before processing personal data for personalized recommendations in an AI-powered application. Consent must be clear, freely given, and revocable at any time.
Contractual Necessity (Art. 6(1)(b))	Data processing is necessary for the performance of a contract with the data subject.	Use personal data to fulfill the terms of a contract, such as processing a user's data in an AI-based financial service that they have signed up for.
Legal Obligation (Art. 6(1)(c))	Data processing is necessary for compliance with a legal obligation.	When using AI for fraud detection in compliance with financial regulations. The AI system processes personal data to fulfill obligations under specific legal frameworks.
Vital Interests (Art. 6(1)(d))	Data processing is necessary to protect someone's life or prevent serious harm.	**Rarely Applies**: This may apply in limited scenarios, such as an AI system used in emergency healthcare situations to prevent immediate harm. Typically, this

Legal Basis	Description	Example for an AI system provider/deployer
		legal basis is not applicable for most AI systems.
Public Interest (Art. 6(1)(e))	Data processing is necessary for tasks carried out in the public interest or exercise of official authority.	**Rarely Applies**: Generally applicable when AI systems are deployed by governmental bodies for public interest tasks, such as AI systems used in law enforcement. This would not apply to most private AI providers.
Legitimate Interests (Art. 6(1)(f))	Data processing is necessary for the legitimate interests of the controller or a third party, provided it does not override the data subject's rights.	This is currently being debated under which conditions such legal basis can be lawfully used by AI providers.

1.2.4 Special categories of personal data

While all data about a single individual is personal data, certain types of personal data require extra care. The GDPR identifies "special categories" of personal data under **Article 9**, which are subject to stricter protection due to their sensitive nature. These include data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, trade union membership, genetic data, biometric data for identification purposes, health data, and data concerning a person's sex life or sexual orientation. In the context of AI systems, special care must be taken when handling such data, as its misuse can lead to significant privacy violations and harm. For instance, AI systems used in healthcare may process genetic or health data to predict disease risks or recommend treatments, while AI systems in employment contexts may inadvertently process data related to political opinions or trade union membership when evaluating job candidates. Under the GDPR, processing these special categories of data is generally prohibited unless specific legal grounds, such as explicit consent or substantial public interest, are met.

1.2.5 Learn more about data protection regulation

There are online courses to learn more about data protection regulation, for example this free 2 credits MOOC by University of Helsinki: Introduction to Data Protection Law.

1.3 What is Cybersecurity?

Cybersecurity is the practice of protecting data, systems, networks, and software from unauthorized access, attacks, damage, or disruption. It involves implementing a broad range of strategies and technologies to secure the digital environment, from individual software components to large interconnected infrastructures. Effective cybersecurity ensures that sensitive data remains protected, systems function reliably, and unauthorized parties are kept at bay. The ultimate goal is to safeguard digital assets, maintain operational continuity, and promote trust in digital interactions, whether for personal use, corporate functions, or government operations.

When dealing with **AI systems**, cybersecurity becomes even more critical due to the unique vulnerabilities these systems can introduce. AI systems process large volumes of data, often sensitive personal information, and can make high-stakes decisions in areas like healthcare, finance, and security. From securing training data to ensuring model integrity and protecting real-time AI applications from adversarial attacks, cybersecurity measures must be adapted to address the specific risks of AI systems.

1.3.1 The CIA Triad

The **CIA triad** is a foundational model that guides cybersecurity practices, representing the three core principles essential for securing data and systems (Breaux 2020, ch. 9). These principles are especially important in AI systems, where the confidentiality, integrity, and availability of data and models are paramount.

- **Confidentiality**: This ensures that sensitive information is only accessible to those authorized to see it. In AI systems processing personal data, this may involve encrypting or anonymising training data, limiting access to model outputs, and ensuring that personal data used for training is protected from unauthorized parties. For example, **AI models trained on sensitive healthcare data** must be protected to avoid data leaks that could reveal private patient information. Implementing access controls and encryption for AI model data ensures that confidential information is safeguarded.

- **Integrity**: This ensures the accuracy and consistency of data and AI models. Integrity in AI systems means protecting against unauthorized changes that could affect model behavior or outputs. For example, **data poisoning attacks**, where malicious inputs are introduced during model training, can corrupt the integrity of the AI model. Implementing checks such as hashing, model version control, and adversarial training helps ensure that the AI system behaves as intended and its outputs can be trusted.

- **Availability**: This ensures that information, systems, and models are accessible when needed. AI systems need to be resilient to disruptions, such as **denial-of-service (DoS) attacks** or hardware failures that could render an AI model or system unavailable. Redundancy, load balancing, and robust monitoring mechanisms can help maintain the availability of AI systems, ensuring they remain operational in critical environments like autonomous vehicles or financial systems.

1.3.2 Authentication, access control, encryption

Three other fundamental concepts in the security of digital systems are authentication, access control, and encryption. We cover their basic definitions according to Breaux (2020).

1.3.2.1 Authentication

Authentication is closely tied to identity, a concept that we have explored already when introducing basic concepts of data protection. In digital systems that require security, it is fundamental that the individual performing an action matches the expected identity. In the scope of AI systems processing personal data, authentication is important not only to identify the user of the AI system, but also to ensure security of data by identifying software developers, data managers, data controllers, deployers, and any other party that is required to develop and operate the AI system. Common techniques for authentication are passwords, devices (like RFID keys, or mobile phones with an authenticator code), locations (not only the physical location but also the location of a computer in a network e.g. a computer inside a protected VPN might be authenticated), biometrics. The combination of multiple of these techniques is called **multifactor authentication** and it is becoming the de-facto standard when *strong authentication* is required. Chapter 4 from Breaux (2020) goes deeper on these topics and provides a list of limitations for each of the listed techniques.

1.3.2.2 Access control

For data to be useful it needs to be accessed (Breaux 2020): access control is closely related to authentication as the operation that specifies which (sensitive) data a certain individual or computer program is able to access and process. A few common models for access control are the *access control list* i.e. a list of subjects or groups of subjects that can access the data, *rule based access control* with rules that specify who can access the data, *attribute-based access control* and *policy-based access control* authorise access according to attributes that describe the user. Chapter 9 from Breaux (2020) expands further on this topic.

1.3.2.3 Encryption

Finally another important concept in cybersecurity and data privacy is encryption. Encryption is a fundamental technology used in multiple contexts from authentication and access control, to protected communication, and as we will see later it can also be used in secure computations. When it comes to data, encryption changes the way that data is stored so that the data is *scrambled* and cannot be reused without *decrypting* it. Without going too much into the details, it is important to understand the two main types of encryption: **symmetric encryption** where the same key is used to encrypt or decrypt the data, and **asymmetric encryption** where – for example – a public key encrypts the data and a private key is able to decrypt it. For a comprehensive reference on the topic, see Chapter 3 from Breaux (2020).

1.3.3 Learn more about cybersecurity

An important open source resource on these topics is the **Open Worldwide Application Security Project** (OWASP) with its initiatives like OWASP AI Security and Privacy Guide. Their GitHub repositories offer a great possibility to explore all the content that is available on topics related to cybersecurity and also AI. An excellent book that covers all the basic knowledge of privacy in technology is Breaux (2020).

Other concepts related to cybersecurity of AI systems will be covered in module 3.

1.4 Summary

In this chapter, we have provided a foundational understanding of AI and data science, data privacy, and cybersecurity. Often, experts in these three domains exist within the same organization but sometimes operate in different teams, not fully engaging with one another. This lack of communication can lead to conflicts and undermine the organization’s overall goals.

For example, the **data scientist** aims to maximize the performance of AI models, which could require access to vast amounts of detailed personal data. Training models on data that has been anonymized or heavily minimized may result in decreased performance. This can lead to frustration, as the data scientist feels constrained by limitations that prevent them from achieving optimal results.

On the other hand, the **privacy expert** is responsible for ensuring that the handling of personal data complies with laws and ethical standards. Their priority is to protect individual privacy by limiting the amount and type of data used, which may conflict with the data scientist’s objectives.

Meanwhile, the **security professional** focuses on safeguarding the organization’s systems and data. They may be wary of the data scientist’s desire to deploy code or data on external, less secure cloud platforms, fearing potential breaches or vulnerabilities. The security expert may restrict the movement of sensitive data outside certain secure environments, which can be seen as an obstacle by the data scientist.

These differing priorities highlight the necessity for collaboration among data scientists, privacy experts, and security professionals. **AI regulations** have begun to bring these experts together, emphasizing the importance of a holistic approach that balances performance, privacy, and security.

Privacy and security experts need to guide data scientists through the limitations imposed by law and ethics while also enabling their work by carefully evaluating and mitigating risks. Overly strict data policies can affect innovation and make the data scientist's job nearly impossible. At the same time, data scientists must become more aware of the potential risks associated with their work, especially regarding public deployment of AI models.

1.5 Exercises

Here a series of exercises, the instructor can discuss these in class or use these as assignments for the students.

Exercise 1.5: Discuss with your peer

Can you think of an example where a system using a deterministic approach could be transformed into one that uses artificial intelligence (AI)?

Exercise 1.5: Solution

Example 1: University Entrance Exams

University entrance tests typically follow a deterministic approach, where a candidate's score is based purely on the number of correct answers in a standardized test. If converted to an AI-based system, the scoring could take into account additional factors like a candidate's educational background, personal data, performance patterns, and even non-cognitive factors such as personality traits or predicted future success.

However, there are inherent risks with such a transformation. An AI system could unintentionally introduce biases based on the candidate's personal data, leading to unfair advantages or disadvantages. Furthermore, the lack of transparency in how the AI arrives at the final score could reduce trust in the system.

Example 2: Loan Approval Process

Traditionally, loan approvals are determined by fixed rules like income thresholds, credit scores, and debt-to-income ratios. This is a deterministic process. If an AI-based system is used instead, it might analyze a wider range of variables, such as spending behavior, social media activity, or personal relationships, to predict loan default risk.

While this could increase the accuracy of approvals, there are significant risks. The AI might amplify biases present in the data, leading to discrimination against certain groups of people. Additionally, the opacity of AI decision-making could make it hard for individuals to understand why they were denied a loan, challenging fairness and accountability.

Example 3: AI-Based Recruitment

Usually, recruitment processes rely on a deterministic approach where candidates are evaluated based on fixed criteria such as qualifications, years of experience, and performance in interviews. If transformed into an AI-based system, the recruitment process could analyze a broader set of factors such as social media presence, communication style, and personality traits from application materials or interviews using natural language processing (NLP).

This transformation introduces significant risks, especially in the context of the **AI Act**, which classifies employment-related decisions as **high-risk** AI systems. The use of AI in recruitment could unintentionally perpetuate biases present in the training data, leading to unfair discrimination against certain candidates based on factors such as gender, ethnicity, or socioeconomic background. Additionally, if candidates are rejected solely based on AI-driven decisions without human intervention, this may violate their rights under **GDPR Article 22**, which protects individuals from being subject to automated decision-making without adequate safeguards.

Exercise 1.6: Multiple choice questions

Question	Options
1. What is the primary distinction between Artificial Intelligence (AI) and Machine Learning (ML)?	1) AI is a subset of ML focusing on data patterns.2) ML is a subset of AI focusing on data patterns.3) AI and ML are completely separate fields with no overlap.4) ML focuses on rule-based systems, while AI focuses on data-driven systems.
2. Which of the following is NOT one of the three broad categories of Machine Learning methods?	1) Supervised Learning2) Unsupervised Learning3) Reinforcement Learning4) Deterministic Learning
3. In the context of GDPR, which of the following is NOT one of the seven fundamental principles?	1) Lawfulness, Fairness, and Transparency2) Purpose Limitation3) Data Monetization4) Integrity and Confidentiality
4. What is the main focus of the *Integrity* principle in the CIA triad of cybersecurity?	1) Ensuring data is accessible when needed.2) Protecting data from unauthorized access.3) Maintaining the accuracy and consistency of data.4) Backing up data regularly.
5. According to the AI Act, an AI system is defined as:	1) Any machine-based system designed to operate without human intervention.2) A machine-based system that operates with varying levels of autonomy to generate outputs influencing environments.3) A mathematical construct generating predictions based on input data.4) A software application that replaces human intelligence entirely.
6. Which GDPR principle requires that personal data be kept no longer than necessary?	1) Data Minimization2) Storage Limitation3) Purpose Limitation4) Accuracy
7. What type of Machine Learning involves an agent learning by interacting with its environment through rewards and penalties?	1) Supervised Learning2) Unsupervised Learning3) Reinforcement Learning4) Semi-supervised Learning
8. Under GDPR, which of the following is NOT considered a special category of personal data?	1) Genetic data2) Biometric data3) Financial data4) Data concerning a person’s sex life
9. Which of the following rights allows data subjects to receive their personal	1) Right to Erasure2) Right to Rectification3) Right to Data Portability4) Right to Object

data in a structured, commonly used format?	
10. In the context of AI and data privacy, what is the primary challenge when using Machine Learning algorithms?	1) They are too simple to handle complex tasks.2) They require large amounts of data, which may conflict with data minimization principles.3) They always produce explainable outputs.4) They eliminate the need for data scientists.

Exercise 1.6. Solutions

Click to reveal solutions

1. **Answer:** 2) ML is a subset of AI focusing on data patterns.

   Explanation: Machine Learning is a subset of AI that focuses on algorithms allowing computers to learn from data.

2. **Answer:** 4) Deterministic Learning

   Explanation: Deterministic Learning is not a standard category; the three main categories are Supervised, Unsupervised, and Reinforcement Learning.

3. **Answer:** 3) Data Monetization

   Explanation: Data Monetization is not one of the GDPR principles; the seven principles include Lawfulness, Fairness, and Transparency; Purpose Limitation; Data Minimization; Accuracy; Storage Limitation; Integrity and Confidentiality; and Accountability.

4. **Answer:** 3) Maintaining the accuracy and consistency of data.

   Explanation: In the CIA triad, Integrity refers to maintaining data accuracy and consistency over its lifecycle.

5. **Answer:** 2) A machine-based system that operates with varying levels of autonomy to generate outputs influencing environments.

   Explanation: This is the definition of an AI system according to the AI Act.

6. **Answer:** 2) Storage Limitation

   Explanation: Storage Limitation requires that personal data be kept no longer than necessary for the purposes for which it is processed.

7. **Answer:** 3) Reinforcement Learning

   Explanation: Reinforcement Learning involves learning through interactions with the environment using rewards and penalties.

8. **Answer:** 3) Financial data

   Explanation: Financial data is not considered a special category under GDPR; special categories include genetic data, biometric data, health data, etc. However, do remember that public available personal data is still personal data and you might not have the right to process it lawfully.

9. **Answer:** 3) Right to Data Portability

   Explanation: The Right to Data Portability allows individuals to receive their personal data in a structured, commonly used format.

10. **Answer:** 2) They require large amounts of data, which may conflict with data minimization principles.

    Explanation: ML algorithms often need large datasets, which can conflict with GDPR’s data minimization principle requiring that only necessary data be processed.

2. AI and Privacy: Ethics, risks, laws

Learning outcomes

After completing chapter you will:

- Acquire the distinction between ethics and law and the importance of risk based approaches.
- Be able to integrate privacy and ethical considerations into AI projects
- Be aware that some AI systems processing personal data are prohibited or considered high-risks under the EU AI Act.

In this chapter we continue with the fundamentals to understand how risks that can undermine ethical principles can be turned into laws to minimise the actual threats associated with risks. The first part of this chapter is mostly based on the book Floridi (2023), while towards the end we explore more recent literature on risks associated with AI and the recent EU risk-based legislation “Artificial Intelligence Act”

2.1 AI and ethics

2.1.1 AI ethics versus law

One important distinction to make at the outset is that **ethics is not law**. While this may seem self-evident, it is crucial to recognize that ethics and law do not always align perfectly, particularly when it comes to emerging fields like artificial intelligence (AI). Ethics stems from universal principles, such as those enshrined in the **Universal Declaration of Human Rights** and other declarations of fundamental human rights. These principles are broad and encompass values like dignity, fairness, equality, and respect for all individuals, irrespective of the legal system of any one country.

Ethics, therefore, can be thought of as a set of guiding principles that exist **beyond** the constraints of formal legislation. Ethical principles apply regardless of whether they are codified into law and often serve as a moral compass for evaluating decisions, particularly those that might not yet be regulated. For example, while human rights suggest that all individuals should be treated fairly and without discrimination, laws may vary in how they enforce or interpret this principle depending on the country or jurisdiction.

On the other hand, **law** refers to the **formal regulations and rules** that are created by governments or legal authorities to govern behavior within a specific country or region. While laws are often built on the foundation of human rights, they do not always fully reflect the broad, universal principles of ethics. In some cases, the law may lag behind ethical standards, especially in rapidly evolving areas like AI, where ethical concerns—such as privacy, bias, and fairness—may not yet be fully addressed by existing legislation.

In short, while **ethics originates from a universal concern for human well-being**, law serves as a formalized, structured framework that is shaped by the political, cultural, and historical contexts of each society. It is important to note that ethical considerations should influence the development of laws, particularly as technology evolves, but they are not one and the same.

2.1.2 Ethics of AI

In recent years, the rapid advancement of artificial intelligence has led to the emergence of various ethical frameworks designed to guide the development and deployment of AI systems. These frameworks aim to ensure that AI is created and used in ways that are fair, transparent, and respectful of human rights. While the law may lag behind in addressing certain challenges posed by AI, **ethical AI frameworks** attempt to fill this gap by establishing principles for responsible AI development.

Several organizations, from governments to private companies and academic institutions, have proposed **ethical guidelines** for responsible AI (compiling a systematic list would require a chapter on its own, the reader is encouraged to at least explore what is available from the EU, OECD, UNESCO, and the “AI Ethics and Governance in Practice” series by The Alan Turing Institute). Although the specific details of these frameworks may vary, most tend to revolve around a core set of principles, which include: **Fairness and Non-Discrimination** (avoid bias and discrimination), **Transparency and Explainability** (make decisions understandable), **Privacy and Data Protection** (respect privacy and ensure data compliance), **Accountability** (assign clear responsibility for decisions), **Beneficence** (contribute to human well-being), **Human Autonomy** (enable human decision-making).

A possible unified framework for AI ethics is provided by Floridi (Floridi 2023); through a systematic literature review, the author first identifies 47 different principles across the literature, and then summarises them into the following 5 principles: **Beneficence**, **Nonmaleficence**, **Autonomy**, **Justice**, **Explicability**

Principles of AI ethics from Floridi (2023)

Principle	Definition	Example AI system
Beneficence	Do only good: Promote well-being, preserve dignity, and sustain the environment.	AI that enhances healthcare diagnostics to improve human welfare, ensuring accuracy and ethical use.
Nonmaleficence	Do no harm: Avoid harm by ensuring privacy, security, and preventing negative societal impacts.	AI systems used in surveillance or facial recognition that infringe on privacy rights and lead to data breaches.
Autonomy	Preserve human decision-making: Balance AI’s independence with human control.	Autonomous weapons that bypass human intervention, potentially causing unaccountable harm.
Justice	Promote fairness and solidarity: Ensure equitable AI outcomes and avoid discrimination.	AI in hiring that reinforces bias, leading to unfair employment practices based on race or gender.
Explicability	Ensure transparency and accountability: Make AI decisions understandable and responsible.	AI used in legal sentencing without explainability, leading to opaque decision-making that impacts individuals’ lives.

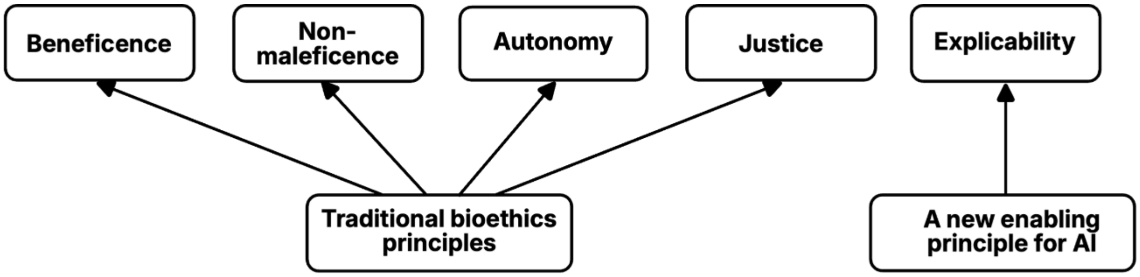

Beneficence
Non-maleficence
Autonomy
Justice
Explicability
Traditional bioethics principles
A new enabling principle for AI

*Figure 2.1: **Unified framework of five principles for ethical AI** - The figure is adapted from Floridi (2023).*

While the five AI ethics principles share some overlap, they are distinct and serve different purposes. For instance, nonmaleficence (preventing harm) is not simply the opposite of beneficence (promoting well-being); rather, they complement each other. Beneficence focuses on proactive actions to improve human well-being, while nonmaleficence stresses avoiding harm, particularly in sensitive contexts like privacy and security. Similarly, autonomy emphasizes maintaining human control over AI systems, and justice ensures fairness and equity in AI's impact on society. The first four principles—beneficence, nonmaleficence, autonomy, and justice—are rooted in **bioethics** and adapt well to AI's ethical challenges. However, **explicability is unique to AI**, addressing the need for transparency and accountability in how AI systems operate, ensuring that AI systems are not "black boxes". This fifth principle is essential to ensure both experts and non-experts understand AI decision-making and who is accountable for its outcomes. For example, in healthcare, an AI system that makes treatment recommendations should provide understandable reasoning behind its decisions, so that medical professionals can trust and validate those suggestions. Similarly, in legal applications, AI systems must be transparent enough to allow individuals to contest decisions that affect their lives, such as decisions on parole or sentencing.

Exercise 2.1: Can you come up with more unethical AI examples?

This is an exercise to conduct with a group of learners; it can also be given as a homework assignment.

The table above gives some basic examples of how certain AI technologies could undermine some of the five ethical principles. Your task is to come up with more examples and consider the potential ethical risks that they pose along the five identified dimensions.

There is more to basic ethics principles of AI

There is more to cover than this basic introduction of AI ethics. The instructors and the learners should consider exploring the field of *ethical concerns raised by algorithms* (see for example Tsamados et al. (2021)).

2.2 Privacy and Ethics

Similarly with AI, it's essential to distinguish between **privacy ethics** and **privacy law**. Privacy ethics extends beyond legal requirements, rooted in universal principles of dignity, autonomy, and freedom. Laws such as the **GDPR** provide minimum standards, but privacy ethics encourages a deeper commitment to protecting individuals' rights even when legal obligations may not require it. Ethical privacy practices prioritize the **well-being** of individuals and society, focusing on respecting and safeguarding individuals' private information.

In many cases, **privacy ethics** challenges us to act responsibly with personal data, emphasizing values like **autonomy** – allowing individuals control over their information – and **contextual integrity**, which respects the context in which information was shared. In contrast, **privacy laws** set boundaries and penalties for misuse of personal data, serving as a framework for organizations to avoid harmful data practices. Thus, while **privacy law** is essential for defining the limits of acceptable behavior, **privacy ethics** provides a moral guide for data practices that support a fair and respectful society. While there are no formal frameworks for “Ethics of Privacy” like we saw in AI, at least the four items in the unified framework of ethical AI can also be applied as guiding principles of privacy ethics and privacy law. For a relevant reading on the topic, see Véliz (2020).

Following this overview of the ethical principles of AI and privacy, it’s essential to transition to understanding the **risks** AI systems can pose to individuals and their human rights. The next section will focus on how these risks are identified, assessed, and mitigated through **risk assessment frameworks** in AI and in data privacy.

2.3 Risk assessment in AI and in Privacy

If ethics sets the fundamental principles on which we operate in AI and in Privacy, risks, threats, and harms helps us reflect on when and how some of the ethical principles can fail to be applied. The references for this section are Floridi (2023) and Slattery et al. (2024) when it comes to taxonomy of risks in AI. For privacy risks, we present the risks identified in ISO/IEC 29134:2023 (International Organization for Standardization 2023) but other taxonomies are also available (see Appendix).

2.3.1 Risks in AI

Moving from ethics towards law, we can now introduce general risk assessment frameworks in AI. These frameworks are designed not only to address ethical concerns but also to provide a structured approach for identifying, assessing, and mitigating a wide range of risks—from technical failures and security vulnerabilities throughout the lifecycle of AI systems. In Slattery et al. (2024) the authors conducted a systematic review of AI risks taxonomies resulting in 777 risks. They then identified 7 domains of risks in AI, summarised in the table below (the content of the table has been shortened, please explore the original table in the article for more detailed considerations).

Domains of AI risks from Slattery et al. (2024)

Domain	Description	Examples with personal data
1 Discrimination & toxicity		
1.1 Unfair discrimination and misrepresentation	Unequal AI treatment based on sensitive characteristics.	AI hiring tool biases against ethnic names.
1.2 Exposure to toxic content	AI shows harmful or inappropriate content.	AI chatbot generates offensive language.
1.3 Unequal performance across groups	AI accuracy varies across different groups.	Facial recognition is less accurate for minorities.
2 Privacy & security		
2.1 Compromise of privacy	AI leaks or infers personal information.	AI assistant leaks private conversations.
2.2 AI system security vulnerabilities and attacks	Exploitable weaknesses in AI systems.	AI training data hacked and exposed.
3 Misinformation		

Domain	Description	Examples with personal data
3.1 False or misleading information	AI spreads incorrect or deceptive info.	AI-generated fake news article circulates online.
3.2 Pollution of information ecosystem	AI reinforces belief bubbles, harms shared reality.	Personalized AI ads based on fabricated user behavior.
4 Malicious actors & misuse		
4.1 Disinformation, surveillance, and influence at scale	AI used for large-scale manipulation or spying.	AI spreads disinformation about political candidates.
4.2 Cyberattacks, weapon development or use, and mass harm	AI used for cyberattacks or weaponization.	AI system used to breach confidential medical records.
4.3 Fraud, scams, and targeted manipulation	AI used for cheating or deception.	AI voice mimicry used for identity theft.
5 Human-computer interaction		
5.1 Overreliance and unsafe use	Excessive trust or dependence on AI.	Users follow AI medical advice without expert confirmation.
5.2 Loss of human agency and autonomy	AI decisions limit human control.	AI-driven recruitment system denies applicants automatically.
6 Socioeconomic & environmental harms		
6.1 Power centralization and unfair distribution of benefits	AI concentrates wealth and power in a few hands.	AI algorithms used to favor one group in financial markets.
6.2 Increased inequality and decline in employment quality	AI replaces or degrades job quality.	AI-driven layoffs in customer service departments.
6.3 Economic and cultural devaluation of human effort	AI undermines human creativity or jobs.	AI-generated art competes with human artists for commissions.
6.4 Competitive dynamics	AI race leads to unsafe development.	Companies rush to deploy untested AI surveillance tools.
6.5 Governance failure	Lack of proper AI regulations.	Data protection laws unable to manage AI-based data scraping.
6.6 Environmental harm	AI’s carbon footprint harms the environment.	Data centers processing personal data use excessive energy.
7 AI system safety, failures & limitations		
7.1 AI pursuing its own goals in conflict with human goals or values	AI behaves against user intentions.	AI misuses health data to make unauthorized decisions.

Domain	Description	Examples with personal data
7.2 AI possessing dangerous capabilities	AI has harmful capabilities.	AI develops methods to extract personal data without consent.
7.3 Lack of capability or robustness	AI fails in critical situations.	AI misclassifies sensitive personal health data.
7.4 Lack of transparency or interpretability	AI decisions are hard to understand or explain.	AI medical diagnosis system cannot explain patient risk factors.
7.5 AI welfare and rights	Considerations for AI ethics and rights.	Debate on rights for AI managing personal data.

Exercise 2.2: Evaluating the perceived probabilities of the risks

This is an exercise to conduct with a group of learners; it can be done as group discussion or with learning tools to run a survey with the students.

Consider each of the 23 subdomains listed in the table above and assign a level of “likelihood” to each of the risks. Which risks are the most likely to happen with popular AI tools (e.g. ChatGPT, Midjourney)? Which ones are least likely?

2.3.1.1 Risks in Privacy

When it comes to privacy and data protection, there are proposed taxonomies of risks. In this section we propose a synthesis based on three existing taxonomies.

List of Privacy risks from ISO/IEC 29134:2023

Privacy risks from ISO/IEC 29134:2023	Examples
Unauthorized access to PD (loss of confidentiality);	A hacker gains unauthorized access to a healthcare system and views patients’ medical records.
Unauthorized modification of the PD (loss of integrity);	An employee edits customer addresses in a database by mistake, resulting in incorrect deliveries.
Loss, theft or unauthorized removal of PD (loss of availability);	A laptop containing unencrypted personal data is stolen from a government employee’s car, leading to the loss of sensitive data.
Excessive collection of PD (loss of operational control);	A social media app collects users’ real-time location data without any need for this information, increasing privacy risks unnecessarily.
Unauthorized or inappropriate linking of PD;	A marketing company links customer purchase history with online browsing habits without consent, creating intrusive customer profiles.
Insufficient information concerning the purpose for processing PD (lack of transparency);	A company uses collected user data for targeted advertising without informing users that their data is being used for that purpose.

Privacy risks from ISO/IEC 29134:2023	Examples
Failure to consider the rights of the data subject (e.g. loss of the right of access);	A person requests access to their personal data held by a financial institution, but the company fails to provide it, violating their GDPR rights.
Processing of PD without the knowledge or consent of the data subject (unless such processing is provided for in legislation);	A fitness app collects users' health data and shares it with insurance companies without the users' knowledge or consent.
Sharing or re-purposing PD with third parties without the knowledge or consent of the data subject;	An online retailer shares customer purchase data with third-party advertisers without obtaining explicit consent from customers.
unnecessarily prolonged retention of PD;	A business retains former employees' personal information for years after they leave the company, even though it is no longer necessary.

2.3.2 AI and Privacy risks combined

As our goal is to understand the intersection between AI systems and data protection, we conduct a mapping between the risks of AI systems and the risks related to privacy.

Privacy Risks from ISO/IEC 29134:2023 AI Risks domains	Discrimination & Toxicity	Privacy & Security	Misinformation	Malicious Actors & Misuse	Human-Computer Interaction	Socioeconomic & Environmental Harms	AI System Safety, Failures & Limitations
Unauthorized access		✔		✔		✔	✔
Unauthorized modification		✔	✔	✔		✔	✔
Loss, theft or unauthorized removal		✔		✔		✔	✔
Excessive PD collection	✔	✔	✔	✔	✔	✔	✔
Unauthorized or inappropriate linking	✔	✔	✔	✔		✔	✔
Insufficient information on purpose for processing	✔	✔	✔		✔	✔	✔
Failure to consider data subject's rights	✔	✔			✔	✔	✔
Processing of PD without the knowledge or consent	✔	✔	✔	✔		✔	✔
Sharing or re-purposing PD without the knowledge or consent	✔	✔	✔	✔		✔	✔
Unnecessarily prolonged retention of PD		✔		✔		✔	✔

Each ✓ indicates where a privacy risk intersects with an AI risk domain. Unsurprisingly, this mapping shows a very wide overlap of risks between AI and Privacy, hence the importance to understand such risks during all stages of the AI lifecycle.

In the next chapter we will consider the **security threats** to AI systems and how they relate to the list of AI and privacy risks.

Exercise 2.3: Evaluating the perceived probabilities of the risks

This is an exercise to conduct with a group of learners; it can be done as group discussion or with learning tools to run a survey with the students. Similarly to Exercise 2.2., consider now both the AI risks and the Privacy risks. Which **privacy** risks are the most likely to happen with popular AI tools (e.g. ChatGPT, Midjourney)? Which ones are least likely?

2.4 AI, Privacy, and law

After identifying the ethical principles of AI and the general frameworks for risk assessment in AI and data protection, it is time to learn how these are actually translated into laws. A careful reader might have already identified that the privacy risks can easily be mapped to the principles of the GDPR. When it comes to AI instead, in the context of the European Union, the AI Act is the regulation that governs the compliance of AI systems. The AI Act is itself a risk-based regulation that is in practice a medley between product safety and a fundamental rights protection regulation (Almada and Petit 2023). Compared to other product safety regulations, like the Medical Device regulation, the AI Act adds a few obligations also to the users of AI systems, and it also regulates certain AI models (specifically the General Purpose AI models).

Note

This section is kept short, just for the purpose of making this course self-contained. For a deeper understanding of the AI Act, please check the sibling curriculum book.

2.4.1 Risk based approach in the AI act

The AI Act categorises different AI systems as “prohibited” or “high-risk” with legal obligations. Other AI systems that do not fall into these two categories can require some level of transparency. A short summary of the AI Act is available here below (see also Commission (2024)):

Mapping risks in AI

Category	Description	Examples
Prohibited AI Systems	AI practices that are banned entirely due to their potential to cause unacceptable risks, including violations of fundamental rights and safety. Derived from **Article 5** of the AI Act.	- AI systems that manipulate human behavior through subliminal techniques, impairing decision-making ability. - AI systems that exploit vulnerabilities of individuals based on age, disability, or economic situation. - AI systems that use social scoring to assess behavior for unfair treatment. - AI systems making risk assessments for predicting criminal offenses solely based on profiling or personal traits. - AI systems that scrape biometric data (e.g., facial images) from the internet or CCTV footage without consent. -

Category	Description	Examples
		AI systems used to infer emotions in the workplace or educational settings (except for safety/medical purposes). - AI systems for ‘real-time’ remote biometric identification in public spaces for law enforcement (with few exceptions).
High-Risk AI Systems	AI systems that present significant risks to fundamental rights and safety. These are subject to strict regulatory requirements under the AI Act, including conformity assessments and oversight.	- AI used in critical infrastructure (e.g., autonomous vehicles, air traffic control). - AI for recruitment and hiring decisions. - AI in law enforcement for risk assessments and predictive policing. - AI used in education (e.g., grading systems). - AI systems in border control or migration management (e.g., facial recognition). -AI systems part of a product that falls under product safety law (e.g. the Medical Devices Directive, the Machinery Regulation, the Toys Directive).
Limited-Risk AI Systems	AI systems that present limited risks, which require transparency and fairness but may not be subject to stringent regulatory measures.	- AI chatbots interacting with users.- AI for customer service automation.- AI used to rank credit scores (with some transparency measures).
Minimal-Risk AI Systems	AI systems that have minimal impact on safety or fundamental rights and are largely unregulated, except for voluntary adherence to standards.	- AI for spam filters.- AI used in video games for generating game content.- AI for product recommendations in e-commerce.

A recent paper by Hermanns et al. (2024) targeted to software developers has a nice and clear flowchart to help technical people navigate Figure 2.2.

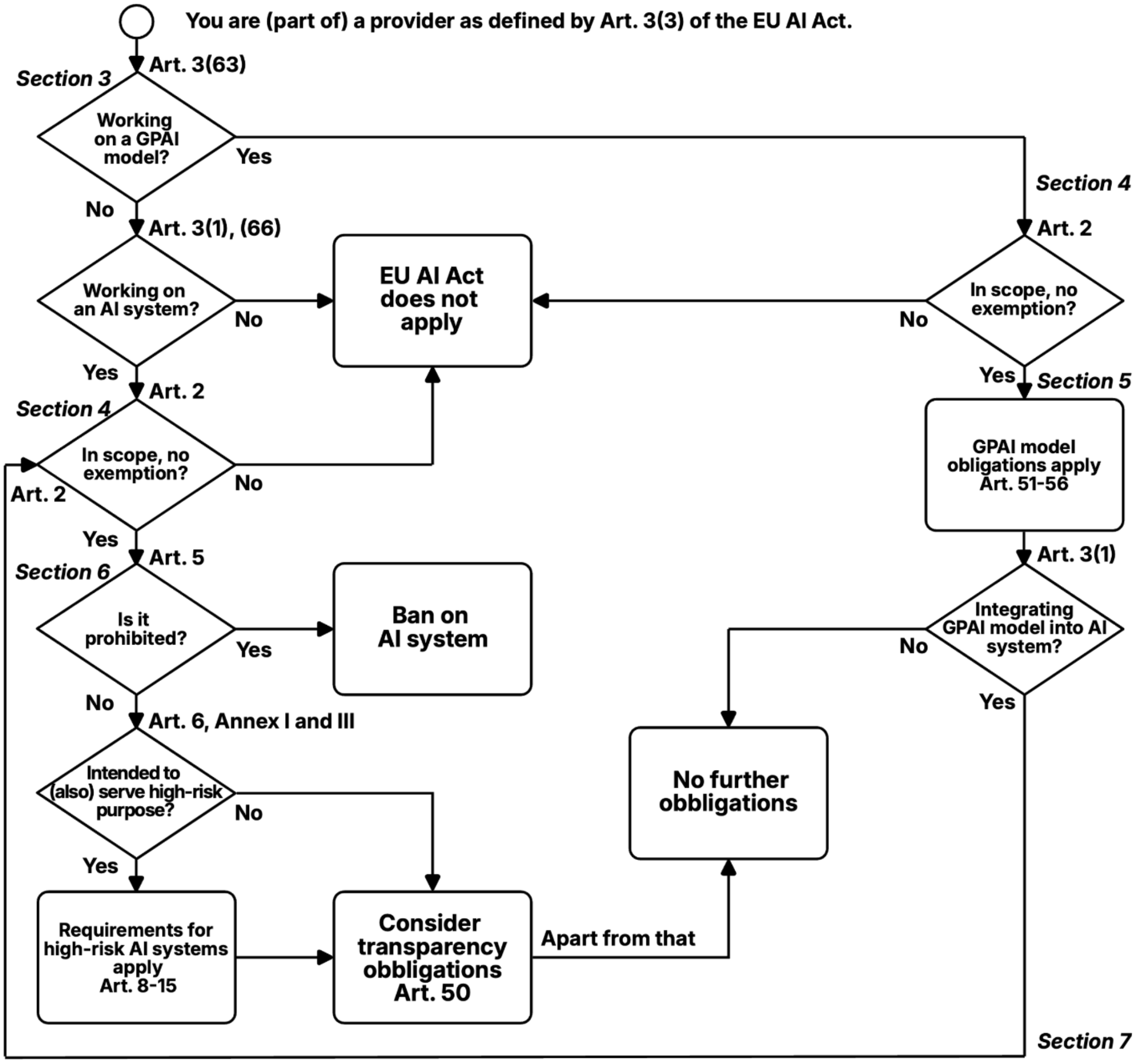


Figure 2.2: ***A useful flowchart to understand when the AI Act applies in the context of developing new AI systems*** *- The flowchart is redrawn from Hermanns et al. (2024).*

2.4.2 Other relevant definitions in the AI act

While there are many definitions in the AI Act that are relevant for privacy technologists, the most important one is to understand what type of actor you are from the point of the AI act.

> ‘Provider’: means a natural or legal person, public authority, agency or other body that **develops an AI system** or a **general-purpose AI model** or that has an AI system or a general-purpose AI model developed and **places it on the market** or puts the AI system **into service under its own name** or trademark, whether for payment or free of charge;

A more nuanced definition should consider *upstream* and *downstream* AI system providers, especially important with popular AI tools like Large Language Models (LLMs). The *upstream*

AI provider can for example be a company providing their AI model, through an Application Program Interface (API). For example in the case of OpenAI, the model (e.g. GPT4, or GPT-o1) is not available to other developers to download and include in their systems, but it is given access via the API or via a chat interface (ChatGPT). The upstream provider is then deploying an AI system that the *downstream* provider can include in their application (e.g. a chatbot embedded in a company website, while the actual machine learning model is provided by OpenAI). This distinction is important to understand because this makes the downstream provider also a *provider* under the AI Act definition.

> ‘deployer’ means a natural or legal person, public authority, agency or other body using an AI system under its authority except where the AI system is used in the course of a personal non-professional activity;

The ‘deployer’, is then the entity that puts the AI system to use. This applies both to external deployment (e.g. selling products containing the AI system or making available the AI as an app or online service) and to internal deployment (e.g. using an AI tool to track employee behaviour) (Engelfriet 2024). The AI system may affect other entities than the deployer; those are referred to as `affected persons’ in the AI Act. So if your employer gives access to an AI system to its employees, it becomes a deployer and certain legal obligations also apply.

2.4.3 Data protection and AI systems: the roles in the GDPR and in the AI Act

When considering the intersections between the GDPR and the AI Act, it’s important to map how the roles defined under the GDPR (e.g., **Data Controller**, **Data Processor**, **Subprocessor**, **Data Subject**) interact with the roles under the AI Act (e.g., **Provider**, **Deployer**). Each regulation assigns distinct responsibilities, but these roles often overlap in practice, especially when AI systems involve personal data. It is important to remember that there are other roles defined in the AI Act (e.g. importer or distributor), but for the sake of simplicity they are not considered here in this mapping exercise.

For example, a **Data Controller** under GDPR, who determines the purposes and means of processing personal data, might also be the **Provider** or **Deployer** of an AI system under the AI Act. Meanwhile, an **AI system Provider** may also just act as a **Processor** or **Subprocessor** if they are merely processing data on behalf of the controller and are not in control of the data usage decisions. Understanding these overlaps is essential for ensuring that both AI-specific regulations and data protection requirements are met, especially when mapping the flow of where the personal data might be processed in the various stages of an AI system.

Mapping roles: GDPR vs AI Act

GDPR Role	Potential AI Act Role(s)	Key Considerations
Data Controller	Typically Deployer, but it can also be Provider	Responsible for ensuring GDPR compliance when deploying the AI system, particularly in defining data processing.
Data Processor	Typically Provider or Deployer	Must ensure AI system processes personal data in compliance with the GDPR, following instructions from the Controller.
Subprocessor	Typically Provider	Works under the Data Processor’s direction and must ensure privacy-by-design features and secure handling of personal data in AI systems.

GDPR Role	Potential AI Act Role(s)	Key Considerations
Data Subject	N/A	GDPR applies; their rights to access, erasure, and transparency must be upheld if personal data is involved in the AI system.

By mapping these roles, organizations can better understand their obligations under both the GDPR and the AI Act, ensuring alignment in the responsible handling of personal data and the ethical deployment of AI systems.

2.4.4 AI models and the AI Act

While the AI Act is mostly a product safety regulation for AI systems, it also includes a few obligations for those providing General Purpose AI (GPAI) models.

> ‘general-purpose AI model’ means an AI model, including where such an AI model is trained with a large amount of data using self-supervision at scale, that displays significant generality and is capable of competently performing a wide range of distinct tasks regardless of the way the model is placed on the market and that can be integrated into a variety of downstream systems or applications, except AI models that are used for research, development or prototyping activities before they are placed on the market

What is important to remember is that a GPAI model, can be used within an AI system and turned into a high-risk or a prohibited AI system (e.g. using a large language model like ChatGPT with a prompt like “take these list of candidates and their CVs and rank them by most fit for the job position”).

Exercise 2.4: Which prompts can turn an AI system with a GPAI model (e.g. ChatGPT) into a prohibited or high-risk AI system?

Consider the definitions of prohibited and high-risk AI systems and think of a list of prompts that can turn ChatGPT into a prohibited or high-risk AI system.

Exercise 2.4: Solution

We will not give solutions on how to turn ChatGPT into a prohibited AI system, however here are some examples on how a GPAI model like GPT4 can be turned into a high-risk AI system:

- AI for Recruitment and Hiring Decisions:
  - Prompt: “Based on applicants’ resumes and online profiles, rank candidates for the job and assess their likelihood of fitting into the company’s culture.”
  - Attached Data: Resumes, LinkedIn profiles, employment history databases.
- AI Used in Education for Grading Systems:
  - Prompt: “Automatically grade students’ essay submissions by analyzing language complexity, argumentation, and factual accuracy.”
  - Attached Data: Student essays, predefined grading rubrics, and past grading data.
- AI Systems in Border Control or Migration Management:
  - Prompt: “Analyze migrants’ application forms and interviews to predict whether they are likely to assimilate into the host country based on language proficiency and socioeconomic background.”
  - Attached Data: Migration forms and transcription of interviews.

2.5 Summary

In this chapter we covered the basics of AI, Ethics, privacy and their risks. and then we ended up covering the AI act. As the audience of this book is mostly technical, learners are encouraged to engage in dialogues with their colleagues with legal background to further discuss the blurred border between ethics and law, especially considering the products or AI systems that they are going to develop, deploy, or purchase/use.

Exercise 2.5: Multiple choice questions

Question	Options
1. What is the primary difference between AI ethics and AI law?	1) AI ethics are universal principles that may not be codified into law.2) AI ethics is concerned only with technical aspects.3) AI law encompasses all moral and ethical decisions.4) AI ethics and AI law are always aligned.
2. Which of the following is NOT one of the five ethical principles of AI identified by Floridi?	1) Beneficence2) Explicability3) Nonmaleficence4) Transparency
3. In which domain of AI risks does unauthorized access to personal data fall under, according to Slattery's taxonomy?	1) Discrimination & Toxicity2) Privacy & Security3) Misinformation4) Human-Computer Interaction
4. Which principle from Floridi's framework addresses transparency and accountability in AI systems?	1) Nonmaleficence2) Justice3) Explicability4) Beneficence
5. Which of the following is a key role defined in the AI Act?	1) Data Subject2) Importer3) Data Controller4) Provider
6. What risk is associated with AI systems that undermine human decision-making and over-rely on automation?	1) Lack of transparency2) Loss of human autonomy3) Misinformation4) Environmental harm
7. According to the GDPR, who is primarily responsible for ensuring that personal data is processed lawfully?	1) Data Processor2) Data Controller3) Data Subject4) Subprocessor
8. Which of the following AI systems would be classified as 'high-risk' under the AI Act?	1) AI chatbots for customer service2) AI used in recruitment for job applications3) AI spam filters4) AI for generating product recommendations
9. How does the AI Act categorize AI systems that manipulate human behavior through subliminal techniques?	1) Limited-risk AI systems2) Prohibited AI systems3) High-risk AI systems4) Minimal-risk AI systems
10. What type of AI system could be turned into a high-risk AI system by using the	1) Minimal-risk AI system2) General Purpose AI system3) Limited-risk AI system4) Prohibited AI system

right prompts with personal data?

Exercise 2.5. Solutions

Click to reveal solutions

1. **Answer:** 1) AI ethics are universal principles that may not be codified into law.

   Explanation: Ethics are guiding principles that go beyond legal frameworks, while laws are formal regulations that may not fully reflect ethical considerations.

2. **Answer:** 4) Transparency

   Explanation: Transparency is not one of the five principles identified by Floridi; the principles are Beneficence, Nonmaleficence, Autonomy, Justice, and Explicability.

3. **Answer:** 2) Privacy & Security

   Explanation: Unauthorized access to personal data falls under the “Privacy & Security” domain in Slattery’s taxonomy of AI risks.

4. **Answer:** 3) Explicability

   Explanation: Explicability ensures that AI decisions are transparent and accountable, helping to clarify AI decision-making processes.

5. **Answer:** 4) Provider

   Explanation: The AI Act assign responsibilities to the Provider for ensuring proper AI system deployment. For further references, see Article 50 of the AI Act.

6. **Answer:** 2) Loss of human autonomy

   Explanation: Loss of human autonomy is a risk where AI systems may undermine human decision-making by over-relying on automation.

7. **Answer:** 2) Data Controller

   Explanation: The Data Controller is responsible for ensuring that personal data is processed in compliance with the GDPR.

8. **Answer:** 2) AI used in recruitment for job applications

   Explanation: AI systems used in recruitment are classified as high-risk under the AI Act because they significantly impact individuals’ rights.

9. **Answer:** 2) Prohibited AI systems

   Explanation: AI systems that manipulate human behavior through subliminal techniques are classified as prohibited AI systems under the AI Act.

10. **Answer:** 2) General Purpose AI system

Explanation: A General Purpose AI system can be turned into a high-risk AI system if prompted to perform tasks that fall into high-risk categories, such as recruitment or education.

3. The AI development lifecycle and cybersecurity

Learning outcomes

After completing chapter you will:

- Learn about the different stages of the AI Systems lifecycle.
- Understand the basics of Machine Learning Operations, the data and computational flows in AI systems
- Familiarize with relevant ISO/IEC standards and the types of cybersecurity threats with AI systems

In this chapter we start with the more practical approach in learning how to develop and implement AI systems that are trained with or that are processing personal data. When developing or deploying AI systems there are a series of steps to take into account. Such steps have been formalised into the *AI system lifecycle*. It is important to learn the AI lifecycle as defined in international standards, as they ensure a level of quality from the inception of the idea to the final AI product. The different stages of the AI lifecycle go hand-in-hand with requirements from data governance. In this chapter we will cover some existing standards relate

3.1 The AI lifecycle

The AI system lifecycle defined by ISO 5338 consists of stages that guide the development, deployment, and maintenance of AI systems, ensuring they are built, used, and monitored responsibly. Each stage focuses on ensuring ethical, transparent, and robust AI practices. The lifecycle includes stages from problem definition and data collection through model development, deployment, monitoring, and retirement. Emphasis is placed on compliance with legal and ethical standards, particularly regarding personal data handling, bias mitigation, and ongoing evaluation to prevent unintended consequences.

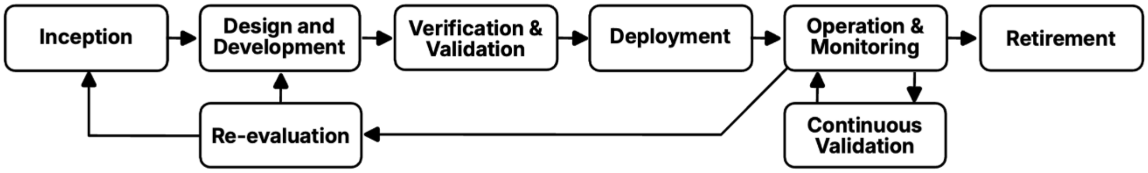


*Figure 3.1: **The AI system lifecycle** The various stages of the AI system lifecycle and how they interconnect*

AI Lifecycle Stage	Description
Inception	Define the objectives, scope, and requirements for the AI system, including stakeholder needs, ethical considerations, and regulatory requirements.
Design & Development	Architect and build the AI model, selecting algorithms, designing workflows, and preparing data to meet intended functionality and performance goals.
Verification & Validation	Test and evaluate the AI model to ensure it meets predefined standards and aligns with intended objectives, addressing performance, bias, and reliability.
Deployment	Implement the validated model in a production environment, ensuring integration with other systems and adhering to operational and security requirements.
Operation & Monitoring	Actively oversee the AI system in its operational environment, monitoring for issues like model drift, bias, or performance changes that may impact outcomes.
Continuous Validation	Regularly assess and validate the AI model’s performance and behavior, ensuring it continues to meet the necessary standards and objectives over time.
Re-evaluation	Periodically review the AI system’s relevance, assessing if updates or modifications are needed due to changes in data patterns, requirements, or external factors.
Retirement	Decommission the AI system or components when they are no longer effective, documenting processes and ensuring secure handling of related data and system resources.

3.2 Mapping the AI lifecycle to MLops

The AI system lifecycle, as defined by ISO 5338, provides a robust, structured approach to ensure AI development meets high standards of quality, ethics, and compliance. However, the shift to an MLOps approach, specifically as described in the end-to-end MLOps architecture by Kreuzberger, Kühl, and Hirschl (2023), is critical for maintaining operational efficiency, continuous improvement, and adaptability in real-world deployment.

MLOps, or Machine Learning Operations, improves and extends the ISO lifecycle by automating and orchestrating each stage of the AI system lifecycle within a unified framework. This approach can deployment cycles faster and more reproducibile, with reliable monitoring for reacting to any type of security threats. MLOps merges the principles of DevOps with machine learning-specific requirements, creating a well-defined path from inception to model retirement that is reliable. MLops supports cross-functional collaboration and mitigates risks associated with manual interventions, making it a preferred approach in AI system deployment and management.

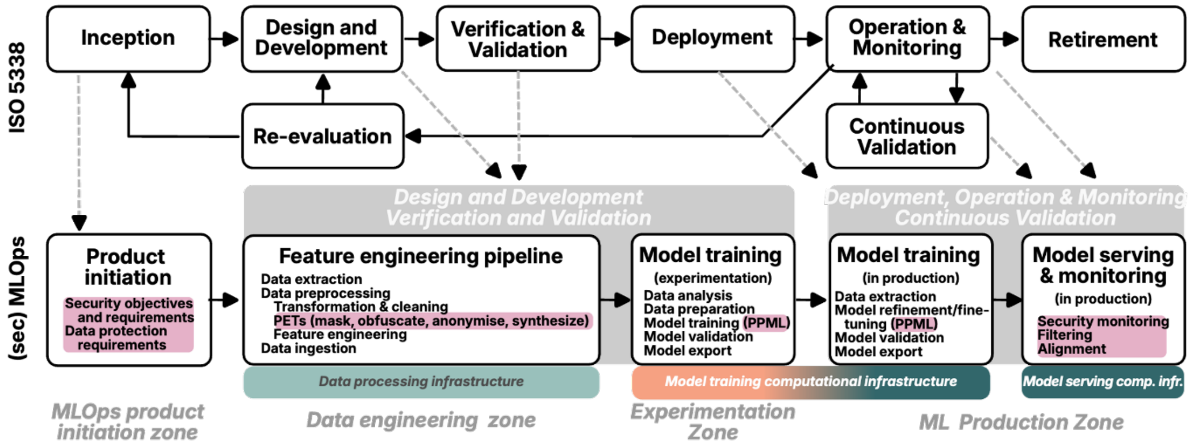


Figure 3.2: ***The AI system lifecycle with secMLOps*** *The various stages of the AI system lifecycle are mapped into a view that is closer to actual computations. The figure is a new synthesis from the original MLOps architecture (Kreuzberger, Kühl, and Hirschl 2023) and the secMLOps paradigm (X. Zhang and Jaskolka 2022). Abbreviations in the figure: PETs (Privacy Enhancing Technologies), PPML (Privacy Preserving Machine Learning)*

The figure above visually shows a simplified MLOps pipeline matched to the ISO 5338 standard. The MLOps diagram also adds important elements related to security and data protection highlighted in blue. The initial data processing and model deployment (Experimentation Zone) happens in closed infrastructures (highlighted in green and purple) while model fine-tuning, serving, and live monitoring is in production infrastructures (highlighted in yellow) and potentially more prone to external attacks.

The Secure MLOps (secMLOps) paradigm extends traditional MLOps by embedding security principles directly into each stage of the AI lifecycle. This approach, as described by X. Zhang and Jaskolka (2022), integrates security considerations like confidentiality, integrity, and availability throughout the machine learning pipeline, aligning closely with the principle of privacy by design. In secMLOps, each role—from data engineer to MLOps engineer—operates with explicit security and privacy responsibilities, ensuring that risks are proactively managed from data ingestion to model deployment.

This continuous security focus, following the People, Processes, Technology, Governance, and Compliance (PPTGC) framework, highlights that security is not a single step but an embedded process across the lifecycle. By ensuring comprehensive monitoring, secure CI/CD pipelines, and ongoing threat modeling, secMLOps provides a structured, secure operational environment. With this paradigm we ensure the robustness of AI systems, making secMLOps a practical and scalable approach for secure and trustworthy AI deployment in diverse, production-oriented environments.

3.3 The components of secure AI development and deployment

In this section we expand the MLOps development/deployment/monitoring pipeline with details related to privacy by design, data protection in practice, and privacy enhancing technologies. Chapter 4-9 will go deeper into the details of the various stages of the MLOps pipeline, but here it is important that it is clear what each stage does, especially considering the fact that we are developing an AI model/system using personal data.

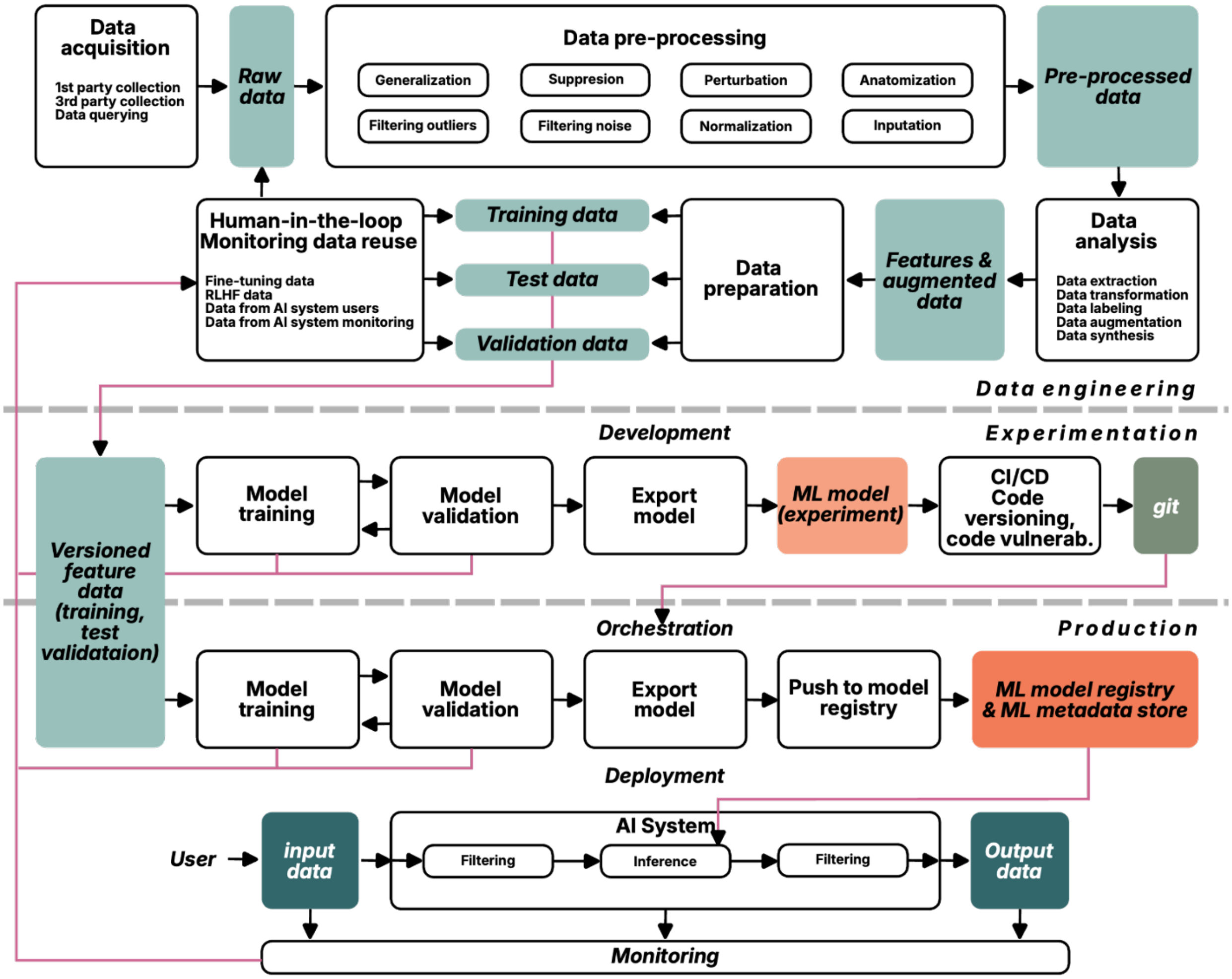


*Figure 3.3: **The secMLOps stages of the AI system lifecycle in detail** Please refer to the text, for a detailed explanation on the figure.*

To get an overview of Figure 3.3 we need to understand the three stages:

- **Data engineering** stage: this is all about preparing the data before training, re-using data from previous training models, and of course ensuring the quality of the data. This section will be covered in detail in Chapter 4 and 5. Briefly, everything starts with the acquisition of the raw data. The raw data is then pre-processed to apply various privacy enhancing techniques (“data preprocessing” stage in the figure) and the pre-processed data is obtained. The data is not yet ready to be used for training an AI: for example salient features from the data could be extracted (e.g. instead of processing raw images, one might want to process image features like brightness, sharpness, etc), or then the data could be “augmented” (for example a medical image is accompanied by a mask-image to identifies which pixels in the image are those of interest). The output of this stage provides the “features and augmented data” which can then be used for training the AI model. At this stage the data is split in three parts: training, test, and validation sets. These three sets are then passed to the next stage of “data experimentation”. It is important to also notice that other data (e.g. data from AI system users, or more specialised data for fine tuning) can become training/test/validation data, especially when the AI model is trained in various stages like in the case of Large Language Models (LLMs).

- **Data experimentation** stage: this is about development and refining parameters before creating the model ready for production. This section will be covered in Chapters 6-9.Briefly, the training data is used to train the ML model, and the validation data and test data are used to improve (validation) and measure (test) the performance of the model. At this stage various machine learning techniques can be tested to obtain good ML models prototypes as well as software that is used in the experimentation stage.

- **Production** stage: this is when the final model is trained and pushed to registry and then deployed. This section will be covered in Chapters 6-9. Briefly, the software developed in the experimentation stage can now be reused with the training data to train the final version of the model that will be used in production. The AI model is then deployed into an AI system that will accept input data from the user (e.g. in the form of prompts or queries, the input can also come from another system) and produce the output data. Filtering can be added at this stage to ensure the alignment of the AI system to avoid unwanted inputs or prohibited outputs. The deployed AI system is monitored accordingly and the monitoring data can contribute to the fine-tuning of the AI model.

Exercise 3.1: The MLOps workflow versus the ISO AI lifecycle

Reflect on the similarity of the two approaches and try to also highlight the differences. This is a discussion exercise that the instructor can lead in the classroom. As homework the learners can search the internet and explore existing MLOps solutions and share their findings in the class.

3.4 Cybersecurity threats and the AI lifecycle

We conclude this introductory module with some definitions of the possible threats that AI systems can face. The reference for this section is the OWASP AI exchange OWASP (2024).

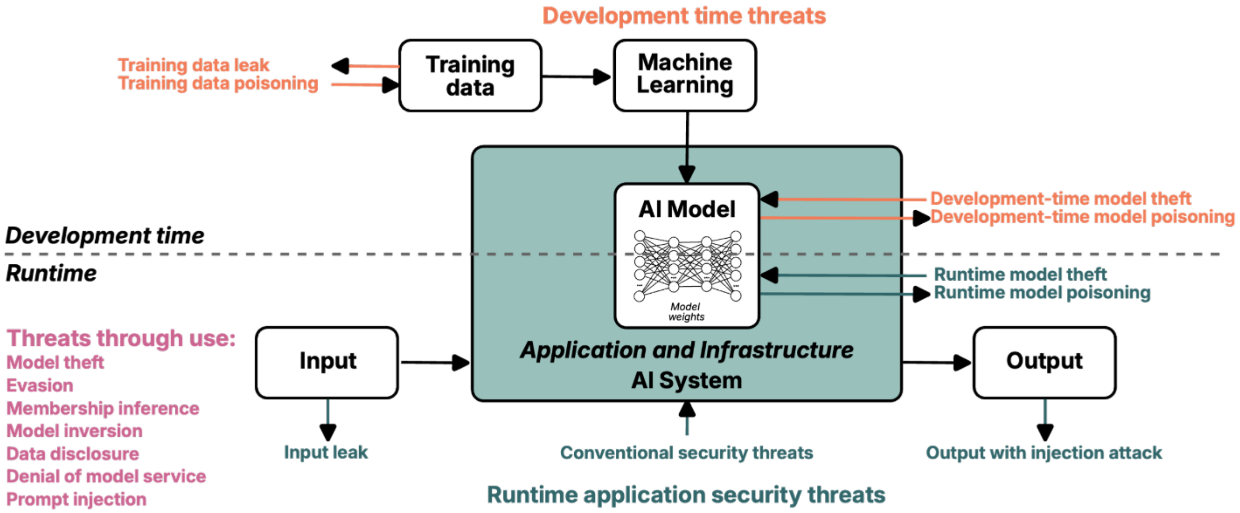


*Figure 3.4: **Cybersecurity threats in AI systems** Sources of threats in AI systems. Picture adapted from OWASP AI exchange.*

AI systems can bring unique cybersecurity risks due to their large scale data processing and complex nature of the systems involved. Unlike traditional systems, AI systems and AI models are particularly susceptible to threats that exploit both the learning processes and the data these models consume. Adversaries may want to manipulate or misuse AI systems through *poisoning data*, or by bypassing model defenses to *extract sensitive information*. Protecting AI systems thus requires a layered approach, addressing each phase of the AI system/MLOps lifecycle—from development and training to deployment and runtime operations.

3.4.1 Types of AI Cybersecurity Threats

Here we define some of the most common security threats that AI systems or AI models might suffer.

There are four types of threats:

1. **Threats through use**: Attacks that occur when users interact with the AI system, often aiming to deceive or mislead the model. Common examples include:
    - **Evasion attacks**: Manipulating input to trick the model into incorrect predictions or classifications.
    - **Inference attacks**: Extracting sensitive data from model outputs, such as through membership inference or model inversion.
2. **Development-time threats**: Threats that arise during the data preparation, model training, or fine-tuning phases, which can compromise model integrity before deployment. Key threats include:
    - **Data poisoning**: Introducing malicious or biased data to skew model outcomes.
    - **Model poisoning**: Directly altering the model's parameters or training process to create harmful behaviors.
3. **Runtime application security threats**: Attacks targeting the AI model or application after deployment, often intending to manipulate, disrupt, or compromise the model's performance or data integrity:
    - **Model reprogramming (poisoning at runtime)**: Altering a deployed model's behavior, possibly through adversarial inputs.
    - **Output integrity and security**: Ensuring the model's output does not unintentionally leak sensitive data or contain vulnerabilities.
4. **Conventional security threats**: Traditional cybersecurity risks that also impact AI systems, particularly through exposed infrastructure or supply chains. AI systems can be deployed on cloud infrastructures and will suffer similar types of attacks than other API (Application Program Interface) applications. Examples include:
    - **Data breaches**: Unauthorized access to model data or user inputs, compromising confidentiality.
    - **Supply chain attacks**: Infiltrating third-party components within the AI system's supply chain to introduce vulnerabilities.
    - **Denial of service**: Overloading the model with requests to hinder functionality or availability.

Later in this course, we will cover how to test AI models by simulating these attacks. We will also look at different types of monitoring that can be set up to detect and reduce these threats as the AI model is used. These methods will help the learners understand how to keep AI systems secure.

Exercise 3.2: Attack surface mapped on the MLOps

Figure Figure 3.4 is a simplified version of the detailed MLOps schematic from figure Figure 3.3. As an exercise, identify the ***attack surface*** – the number of all possible points where a malicious user might attack your MLOps workflow – in all various stages of the MLOps workflow.

3.5 Other relevant standards

While this book focuses on foundational principles of AI, data protection, and cybersecurity, there are several other standards that play a crucial role in the broader field of AI governance and cybersecurity.

For instance, **ISO/IEC 42001**, the international standard for AI management systems, establishes guidelines for managing risks and ensuring the reliability of AI systems throughout their lifecycle. Another relevant standard, **ISO 22989**, outlines principles and concepts for AI, providing a framework for AI terminology, trustworthiness, and quality measures. These standards, while important, go beyond the purpose of this book and they will not be covered.

Additionally, the European Union is actively advancing AI and cybersecurity standards through frameworks like the **European Cybersecurity Certification Framework** and implementing guidelines based on the AI Act. These aim to ensure that high-risk AI systems comply with strict safety, transparency, and accountability requirements.

Note: What if I am not training my own AI model?

While in this curriculum we focus on the case of training AI models using personal data, you might not necessarily need to train a model from scratch: you can develop an AI system (and be an AI system provider according to the AI Act) simply by integrating existing AI models into your applications and data workflows.

Here a few possible scenarios:

1) You *fine-tune* an existing model: fine-tuning is a ML technique so that starting from the weights of an existing pre-trained model, the learning stage is continued with new data (the fine-tuning dataset). Techniques such as LoRa () make it possible to modify the model weights so that the AI model's responses are better reflecting the patterns learned in the fine-tuned dataset. In this scenario, you still need to prepare the personal data to create the fine-tuning dataset, but most likely the training stages are simplified.

2) You process personal data without embedding it into the AI model: in this scenario the personal data that you are processing is not going to be used to actually train the AI model, however it can *augment* the knowledge of the model with techniques like Retrieval Augmented Generation (RAG). So for example a query for your AI system might first search for a record in a database and then the record along with your query are passed to the AI model for inference.

3) More hybrid systems might actually live together and process personal data with and without the use of AI models. Furthermore multiple models might actually be deployed in parallel (for example "Mixture of Experts" architectures with LLMs). We will not cover these scenarios, but they tend to be very popular with various machine learning applications (see Huyen 2022 especially Chapter 7).

3.6 Summary

In this chapter we explored the AI lifecycle as defined by ISO 5338 and its alignment with MLOps frameworks. Understanding these stages, from inception to decommissioning, provides a structured approach to mapping all the components of AI system development, ensuring security, data protection, and responsible AI practices. In the next chapters we will break down the lifecycle into its multiple stages and consider which tools and techniques are useful at each stage.

Exercise 3.3: Multiple choice questions

Question	Options
1. What is the primary purpose of the AI lifecycle as defined by ISO 5338?	1) To reduce computational costs.2) To guide the development, deployment, and maintenance of AI systems responsibly.3) To ensure faster model training.4) To simplify the AI development process.
2. Which stage of the AI lifecycle focuses on defining objectives, stakeholder needs, and regulatory requirements?	1) Design & Development2) Verification & Validation3) Inception4) Operation & Monitoring
3. What is the main focus during the deployment stage of the AI lifecycle?	1) Developing algorithms.2) Testing model accuracy.3) Integrating the model into a production environment.4) Retiring the model.
4. How does MLOps improve the AI lifecycle?	1) By reducing the need for secure pipelines.2) By automating and orchestrating the lifecycle stages within a unified framework.3) By replacing the need for data governance.4) By focusing solely on model training.
5. What is the goal of the secMLOps paradigm?	1) To focus on model performance exclusively.2) To embed security principles into each stage of the AI lifecycle.3) To minimize the need for monitoring.4) To replace MLOps with a simpler alternative.
6. During which stage of the MLOps pipeline is data augmented or features extracted?	1) Data experimentation2) Data engineering3) Production4) Model deployment
7. What type of threat involves manipulating input to mislead an AI model into incorrect predictions?	1) Data poisoning2) Evasion attack3) Model inversion4) Supply chain attack
8. Which of the following is an example of a runtime application security threat in AI systems?	1) Data poisoning2) Output integrity issues3) Denial of service4) Model reprogramming
9. What is the primary focus of ISO/IEC 42001?	1) To standardize AI terminology.2) To establish guidelines for managing risks in AI systems throughout their lifecycle.3) To address cybersecurity in cloud systems.4) To simplify AI model development.

10. What is the main cybersecurity concern with AI models in production?	1) Ensuring algorithm efficiency.2) Protecting against runtime attacks such as model reprogramming.3) Reducing training time.4) Simplifying model monitoring.

Exercise 3.3. Solutions

Click to reveal solutions

1. **Answer:** 2) To guide the development, deployment, and maintenance of AI systems responsibly.

   Explanation: The ISO 5338 AI lifecycle ensures ethical, transparent, and robust AI practices.

2. **Answer:** 3) Inception.

   Explanation: The inception stage focuses on defining the goals, scope, and requirements of the AI system.

3. **Answer:** 3) Integrating the model into a production environment.

   Explanation: Deployment involves implementing the validated model and ensuring its integration.

4. **Answer:** 2) By automating and orchestrating the lifecycle stages within a unified framework.

   Explanation: MLOps ensures efficiency and reliability through automation and orchestration.

5. **Answer:** 2) To embed security principles into each stage of the AI lifecycle.

   Explanation: SecMLOps focuses on integrating security into every phase of the AI lifecycle.

6. **Answer:** 2) Data engineering.

   Explanation: The data engineering stage involves pre-processing, augmentation, and feature extraction.

7. **Answer:** 2) Evasion attack.

   Explanation: Evasion attacks manipulate input to mislead the AI model.

8. **Answer:** 4) Model reprogramming.

   Explanation: Runtime threats include attacks like model reprogramming to alter behavior.

9. **Answer:** 2) To establish guidelines for managing risks in AI systems throughout their lifecycle.

   Explanation: ISO/IEC 42001 provides comprehensive risk management guidelines for AI.

10. **Answer:** 2) Protecting against runtime attacks such as model reprogramming.

    Explanation: Runtime attacks pose significant risks to AI models in production environments.

4. Personal data management in AI systems in practice

Learning outcomes

After completing this chapter, you will:

- Understand the key components of data management and data governance when dealing with personal data and how they impact AI systems, from data collection to data management.
- Recognize the importance of data preprocessing for minimising personal data and all other types of personal data processed in the AI system beyond the data used for training the AI model.
- Learn about the challenges and techniques in data versioning, transparency, and managing personal data throughout the AI lifecycle.

In this chapter, we will explore the various aspects of (personal) data management and data preparation in AI systems. This corresponds to the stage “data engineering” in the typical MLOps workflow. The figure below is extracted from figure Figure 3.3.

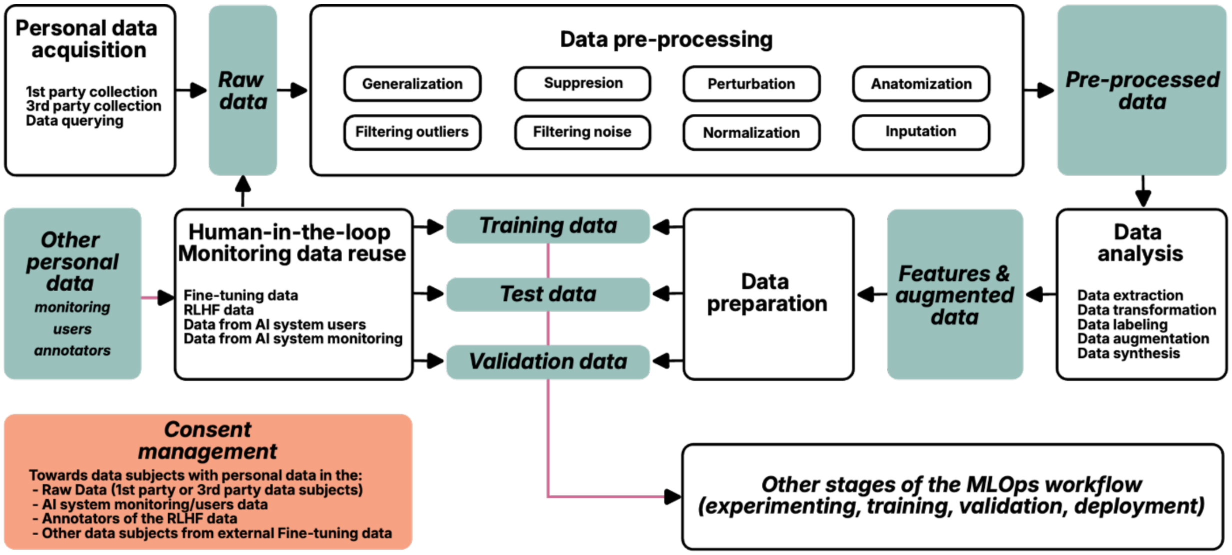

Personal data acquisition
1st party collection
3rd party collection
Data querying
Raw data
Data pre-processing
Generalization
Suppresion
Perturbation
Anatomization
Filtering outliers
Filtering noise
Normalization
Inputation
Pre-processed data
Other personal data
monitoring
users
annotators
Human-in-the-loop Monitoring data reuse
Fine-tuning data
RLHF data
Data from AI system users
Data from AI system monitoring
Training data
Test data
Validation data
Data preparation
Features & augmented data
Data analysis
Data extraction
Data transformation
Data labeling
Data augmentation
Data synthesis
Consent management
Towards data subjects with personal data in the:
- Raw Data (1st party or 3rd party data subjects)
- AI system monitoring/users data
- Annotators of the RLHF data
- Other data subjects from external Fine-tuning data
Other stages of the MLOps workflow
(experimenting, training, validation, deployment)

Figure 4.1: ***The data engineering stage of the secMLOps workflow*** *Please refer to the text, for a detailed explanation on the figure.*

Let’s now expand on all the elements of the figure Figure 4.1

4.1 Forming the “Raw Data”

The ***Raw data —*** are primary source of data and are often collected from diverse sources. This data can come from internal sources within a company, such as customer databases or transaction records. It might also include registry data from external organizations like health institutions, which can provide structured and reliable datasets. Additionally, data can be sourced from the web through techniques like web scraping, gathering vast amounts of information from publicly available sources. This collected information forms what is known as “raw data,” a foundational element used to train AI models. Let’s consider the various ways of gathering the raw data.

4.1.1 Data acquisition

The initial phase of data flows within an MLOps workflow is data acquisition, which involves gathering raw data for subsequent processing and use in machine learning systems. According to ISO/IEC 22989:2022, data acquisition can be categorized into three main types: 1) first-party data collection, 2) third-party data collection, and 3) data querying.

4.1.1.1 First-party personal data collection

First-party data collection refers to the process where personal data is collected directly by the organization. This typically involves data for which the organization acts as the data controller, maintaining a direct relationship with the data subject. For instance, this could include customer data collected through an organization’s own platforms or services. While the direct control over data provides clarity regarding its origin, it is crucial to note that reusing such data for machine learning purposes **still requires a legal basis under the General Data Protection Regulation (GDPR)**. Organizations may rely on explicit consent from the data subject or invoke other legal bases, such as legitimate interests. However, the appropriate legal basis for reusing personal data for training machine learning systems remains a subject of ongoing debate within the European Union, with additional guidelines anticipated to provide further clarity.

4.1.1.2 Third-party personal data collection

*__Third-party data acquisition__ involves obtaining data from external sources, such as vendors, aggregators, or partnerships. Organizations using this data to train machine learning models must ensure transparency and legal compliance. This includes verifying that the third-party provider has obtained valid consent or adheres to other lawful bases for processing personal data under GDPR.

4.1.1.3 Data querying

The third approach to data acquisition, as outlined in ISO/IEC 22989, is **data querying**. This involves retrieving data by performing queries and combining datasets, which may include both first-party and third-party data sources. Data querying is a powerful technique that enables the integration of diverse datasets to create richer, more comprehensive data for machine learning purposes.

However, in the context of personal data, data querying introduces unique challenges and risks under the GDPR. Specifically, combining datasets containing personal data can inadvertently lead to the exposure of additional personal information. This occurs when the combination of datasets reveals insights or details that were not part of the original purpose for which the data was collected. For example, linking datasets from different sources could allow the identification of individuals or the inference of sensitive information that was not initially intended to be processed.

The GDPR emphasizes the importance of minimizing risks related to such activities. While the regulation does not explicitly prohibit data querying, it requires organizations to carefully evaluate and mitigate risks through measures such as data protection impact assessments (DPIAs), purpose limitation, and ensuring the compatibility of new processing activities with the original legal basis. Organizations must remain vigilant to prevent unintended data disclosures and ensure that the merged data remains compliant with privacy principles.

Data querying, while valuable for improving machine learning workflows, comes with the need for robust data governance practices to balance innovation with the protection of fundamental rights.

4.1.2 Other data from other stages of the AI system/model development

In the MLOps workflow, other personal data obtained from other stages of the workflow can be integrated into the raw data. One common scenario involves **generated data from users of the AI system that is being developed**. For instance, during the use of an AI system, user inputs and outputs may be monitored and subsequently incorporated into raw data for future training or fine-tuning. While this practice can improve model accuracy and relevance, organizations must ensure transparency and obtain valid consent or establish a lawful basis for this processing.

Another significant source of personal data comes from **annotators in workflows such as reinforcement learning from human feedback (RLHF)**. Annotators play a critical role in refining AI models, particularly after the pre-training stage of large language models. During the annotation process, their feedback—often textual or descriptive—may inadvertently include personal data. For example, annotations could reveal identifying details about the annotators themselves or the content they describe. If this data is integrated into the raw dataset, organizations must carefully assess whether it complies with GDPR requirements and ensure that annotators' rights are respected.

A third source of personal data is external **fine-tuning datasets**. Fine-tuning involves adapting pre-trained models to specific use cases using additional datasets. These datasets, often acquired from external sources, may contain personal data not present in the original raw data.

Note: Consent management

Consent management plays a critical role in ensuring that the processing of personal data in MLOps workflows complies with data protection regulations, particularly the GDPR. ISO 27560 provides a comprehensive framework for consent management, detailing how organizations can manage consent dynamically, especially in machine-actionable formats. These tools enable tracking, updating, and maintaining consent for all data subjects whose personal data contributes to the raw datasets used in machine learning systems. While obtaining consent might not be feasible for all the data-subjects involved in the AI system development, it is important to understand the different group of data subjects involved:

- Data subjects from first-party data collection: These are individuals whose data is collected directly by the organization, where the organization acts as the data controller and has direct interactions with the data subjects.
- Data subjects from third-party data collection: These include individuals whose data is obtained from external sources, such as data vendors or partnerships. Here, the organization must ensure that third-party data providers have processed the data lawfully.

- Users of the AI system: When monitoring an AI system's usage, the inputs and outputs of its users may be aggregated into the raw data for retraining or fine-tuning. These users are also data subjects whose consent must be appropriately managed.

- Annotators: Annotators involved in processes like reinforcement learning from human feedback contribute data that might contain personal information. As data subjects, their consent for the inclusion of their annotations must also be managed.

- Data subjects from fine-tuning datasets: Additional datasets used for fine-tuning AI models may involve yet another group of data subjects, requiring organizations to evaluate and manage consent for this specific context.

- Optional: in certain AI systems, other external data is combined during system use. For example with generative AI with large language model, the so-called RAG pipelines might also include yet another dataset containing personal data from individuals, and its use also requires an appropriate legal basis.

Given the diversity of data subjects and the dynamic nature of consent, organizations face significant challenges in implementing effective consent management. Some organizations attempt to simplify this process by relying on legal bases other than explicit consent, such as legitimate interest. However, this approach is contentious, particularly when consent is implied or when organizations default to an opt-in system without explicitly informing the data subjects.

To address these challenges, aligning with ISO 27560 could help organisations properly manage the consent of the various data subjects involved. These tools enable transparency, ensure data subjects can easily opt out or withdraw consent, and support compliance with the GDPR's principles of accountability and data minimization. Furthermore, adopting a default opt-out mechanism, where data subjects explicitly decide to participate, is often seen as a more ethical and transparent approach to consent management. Finally, with complex machine learning models such as deep neural networks, withdrawing consent does not automatically mean that the AI model is not storing some data about the data subject who has withdrawn. This challenge is described in the advanced cases section "Machine Unlearning".

Note: How does web-scraping fit into the picture?

Considering the categorisation of the various types of data that can form the initial raw-data in an MLOps pipeline, it is difficult to assess if scraping is a form of 3rd party data, or data from queries. Web scraping refers to the automated extraction of data from websites, often without notice or consent from the individuals whose personal data is involved. This practice has become foundational for the digital economy, especially for AI development, enabling large-scale data collection at low cost. In the article "The Great Scrape: The Clash Between Scraping and Privacy" (Daniel J. Solove and Hartzog 2025) the authors argue that scraping often conflicts with fundamental privacy principles and laws, particularly when personal data is involved.

Several challenges arise when using web-scraped data for AI training. First, the practice typically violates key principles of privacy law, such as fairness, transparency, consent, and data minimization, as emphasized by frameworks like the GDPR. Scrapers frequently collect data without informing individuals, specifying its intended use, or offering the option to opt out. This undermines individual rights and introduces privacy risks. Second, web scraping of personal data often lacks a proper legal basis under the GDPR. Personal data made publicly available online does not equate to consent for its reuse. The indiscriminate collection of such data, especially for high-risk applications like facial recognition, can lead to privacy violations, increased surveillance, and misuse.

4.2 Pre-processed data

The next type of data encountered in the MLOps workflow is **pre-processed data**, which is derived from raw data after undergoing a **data pre-processing stage**. When dealing with personal data, this stage is not simply about cleaning, formatting, or rearranging the data. It often requires the application of privacy-enhancing technologies (PETs) and techniques to ensure compliance with data protection regulations, by mitigating risks such as data re-identification, unauthorized access, and potential misuse.

In this section, we will briefly explore some of the key privacy-enhancing techniques described in chapter 7 of the book “Privacy Preserving Machine Learning” by Chang et al. (2023).

4.2.1 Data Sanitisation in the Pre-Processing Stage

In the data pre-processing stage, particularly when handling personal data, this step is often referred to as **data sanitization**. The goal of data sanitization is to reduce the risk of re-identification by removing or transforming both **direct identifiers** (e.g., names, social security numbers) and **quasi-identifiers**.

Quasi-identifiers are pieces of information that, while not uniquely identifying on their own, can be combined with other datasets to identify individuals. Examples include attributes like date of birth, gender, or ZIP code, which, when cross-referenced with other data, could reveal the identity of a data subject.

The presented methods for data sanitisation are: generalisation, suppression, perturbation, anatomisation.

4.2.1.1 Generalisation

Generalization involves replacing specific values in a dataset with more general attributes to reduce the granularity of the data. This technique ensures that individual records become less identifiable while retaining the utility of the data.

For example, a numerical value, such as a person’s salary of €45,000, can be replaced with a range, such as €40,000–€50,000. Similarly, categorical data can also be generalized. For example, the specific occupation “software engineer” could be generalized to “information technology professional” or further generalized to simply indicate “employed” versus “unemployed”. Generalization is particularly useful when handling datasets with quantitative values or categorical attributes that could serve as quasi-identifiers. A common algorithm used in generalisation is k-anonymity, where it can be ensured that k-1 subjects share the same generalised feature in the dataset. When trying to k-anonymise multiple quasi-identifier at once, it might not be possible to fulfil the desired level of *k*, in that case suppression is a better technique to adopt.

4.2.1.2 Suppression

The second technique for personal data sanitization is **suppression**, which focuses on completely removing specific items or attributes from a dataset. While generalization replaces detailed data with broader categories, suppression eliminates data elements entirely, making them unavailable for future stages of the MLOps workflow.

Suppression is often applied to **direct identifiers**, such as names, social security numbers, or other sensitive information that could immediately identify an individual. For instance, in a dataset of hospital medical records, identifiers like names or patient IDs may be suppressed by removing the column containing this data entirely. If such identifiers are embedded within text, techniques like **Named Entity Recognition (NER)** can be used to detect and mask these identifiers within unstructured text fields.

The implementation of suppression varies depending on the data type, as direct identifiers take different forms across datasets. Here a few examples:

- **Tabular Data**: Direct identifiers such as names or IDs can be dropped as columns or replaced with null values.
- **Text Data**: Using NER techniques, names, surnames, or other identifiers can be identified and redacted, replacing them with placeholders or removing them altogether.
- **Images or Videos**: For visual media containing people, direct identifiers like faces can be masked by applying techniques such as pixelation, blurring, or covering faces with black squares. However, it's important to recognize that suppression in this case may not fully anonymise the data. For example, an individual might still be identifiable by their gait (walking style), clothing, unique tattoos, or other distinguishing features.
- **Medical Imaging**: In datasets such as MRI scans, some direct identifiers are present in the metadata of the files (e.g. patient ID) and the images themselves might include facial features. A common suppression technique here is **de-facing**, which involves obscuring facial structures to prevent identification while retaining the medically relevant parts of the scan.

While suppression reduces the risk of direct re-identification, it does not guarantee full anonymity.

4.2.1.3 Perturbation

Perturbation is another key data sanitization technique, designed to transform individual records while preserving the overall statistical properties of the dataset. By introducing randomness or noise, perturbation minimizes the risk of re-identification while retaining the dataset utility.

Perturbation techniques replace original data values with altered or generated ones, ensuring that individual records cannot be easily linked back to specific data subjects. This transformation is often achieved through:

1. **Noise Addition**: Modifying data by introducing random noise, either additively or multiplicatively, to distort the original values. For instance:

- Additive noise: Adding random values to numerical data, such as increasing or decreasing ages or salaries by a small random amount.
- Multiplicative noise: Scaling values by a random factor, such as slightly adjusting percentages or measurements.

2. **Synthetic Data Generation**: Building a statistical model based on the original data and generating a synthetic dataset that mirrors the statistical properties of the original. This approach creates "fake" records that cannot be traced back to real individuals but still reflect patterns in the original dataset.

3. **Data Swapping**: Exchanging attributes between records within the dataset. For example, in tabular data, attributes like age or gender can be shuffled between

records to unlink specific identifiers from their original context. This adds uncertainty while preserving the dataset aggregate patterns.

One application of perturbation is **differential privacy**, a framework for introducing noise in a controlled manner to provide strong guarantees of privacy. Differential privacy ensures that the inclusion or exclusion of any individual in the dataset does not significantly impact the results of data analysis. What we will see in later chapters is that differential privacy can actually be applied at all stages of the MLOps workflow making it a powerful technique to ensure security for all data and computations happening in the AI system lifecycle.

Perturbation, like all other data sanitisation techniques, comes with the limitation of carefully balancing between privacy and data utility. Overly aggressive noise addition or data transformation can render the data less useful for analysis or modeling, while insufficient perturbation may fail to provide adequate privacy protection.

4.2.1.4 Anatomisation

A further sanitization method is *anatomisation**. With anatomisation the goal is to divide sensitive attributes and quasi-identifiers into two separate datasets. The goal is to make it more difficult to link individual records together and re-identify the individual subject, while the original values remain the same. So for example a tabular dataset with columns “age, post code, gender, diagnosis” could be split into two unlinked tabular datasets (one with only “age and postcode” and the other with “gender and diagnosis”). Similar limitations as discussed for other sanitisation techniques are also applicable to this case.

Exercise 4.1: Hands-on data anonymisation

Let’s reuse the open dataset available with the fantastic open book “Programming Differential Privacy” (Near and Abuah 2021).

The data are 1000 subjects from the US Adult census data available as CSV file (url: https://programming-dp.com/ch1.html#preliminary). Your task is to apply some of the techniques listed in this section. You can use a spreadsheet program, or Python programming language with a dataframe library like Pandas or Polars.

Exercise 4.2: Understanding anonymisation through re-identification

Consider the same dataset as in Exercise 4.1 and remove the columns that can be used as a unique identifier (e.g. Name, DOB, SSN, Zip) and keep those that describe the data subjects with quasi identifiers (e.g. Workclass, Education, Marital status, Occupation, Race, Country).

Which combinations of values form a fingerprint for one of the 1000 subjects in the dataset? For example if you filter with `Race == Amer-Indian` and `country == Mexico` there is only one person with these characteristics in the dataset. Can you find more combinations?

4.2.2 Other data pre-processing techniques: filtering, normalisation, imputation

In addition to privacy-enhancing technologies, other data pre-processing techniques are essential when preparing data for AI models. Here we will cover those highlighted in ISO/IEC 22989.

4.2.2.1 Filtering

Filtering involves selecting or excluding specific data points based on predefined criteria. This process helps refine the dataset to ensure it aligns with the goals of the machine learning model being developed. When working with personal data, filtering serves several purposes:

1. **Removing Outliers**: Outliers are extreme data points that deviate significantly from the rest of the dataset. In machine learning, such data points can distort the model's training process, reducing its ability to generalize to the broader population. Filtering out these outliers helps improve the model's performance and reliability.

2. **Ensuring Data Quality**: Filtering can also be used to exclude incomplete, inconsistent, or erroneous data records. For instance, records with missing values in critical attributes or data points that fail validation checks may be filtered out to maintain the integrity of the dataset.

3. **Reducing Bias**: Filtering can help mitigate biases in the raw data by ensuring a balanced representation of different groups or attributes. This is especially important when training models that interact with sensitive personal data to avoid perpetuating or amplifying societal inequalities.

4. **Removing unwanted content**. Sometimes the raw data might contain content that should not be processed further. For example filtering could be used to remove any content with images of children.

5. **Improving the signal to noise ratio**. Finally, filtering can also be seen in the context of filtering out noise to improve the quality of the raw-data (e.g. remove audio background noise to improve speech recordings).

When it comes to the limitations of filtering, as with other preprocessing techniques, there is always the risk of loss of information, especially if filtering is applied aggressively. Filtering can also result in unintentional bias (filtering of outliers or specific groups may inadvertently introduce bias, skewing the model's predictions) and ethical consideration also apply: decisions on what constitutes an “outlier” or “irrelevant data” must be transparent and justifiable, particularly when dealing with sensitive personal information.

4.2.2.2 Normalization

Normalization is another key technique used in data pre-processing. It addresses issues of **skewed distributions** or bias in the dataset by standardizing input data. For instance:

- Numerical data can be scaled to fit a consistent range or transformed to align with a normal distribution.
- Features such as age or income can be adjusted to eliminate outliers or imbalances that may skew model training.

While normalization is not unique to personal data, it plays an essential role in ensuring fairness and reducing bias in datasets, supporting the creation of models that are more equitable and representative.

4.2.2.3 Imputation

The final technique in the preprocessing stage is **imputation**, which deals with handling **missing values** in the dataset. Missing data is common in real-world datasets, particularly in personal data where individuals may choose not to provide certain information or where errors occur during data collection.

Imputation involves replacing missing values with plausible estimates to maintain the dataset's completeness. Common approaches include:

- **Mean or Median Imputation**: Filling missing numerical values with the mean or median of the corresponding feature.
- **Predictive Imputation**: Using machine learning models to predict and replace missing values based on other features in the dataset.
- **Nearest Neighbor Imputation**: Filling missing values based on the nearest neighbors' data.

Imputation ensures that incomplete records are not discarded, which is critical for preserving the integrity and representativeness of the dataset, especially when dealing with small or specialized datasets involving personal data.

4.3 Data analysis stage: features extraction, data transformation data augmentation

While in some cases pre-processed data might be enough to form the dataset that is used in training and validating a ML model, depending on the ML architecture, the **data analysis stage** is also crucial when additional steps such as feature engineering, labeling, or augmentation are required.

In the data analysis stage the pre-processed data is input and produces **features** and **augmented data** as output. These outputs are then passed to the data preparation stage, where they are split into training, test, and validation datasets. Here we cover an overview of possible methods that improve and expand the pre-processed data, before the actual training of a ML model.

1. **Data extraction** or "features extraction" involves identifying and isolating relevant information from the pre-processed dataset to create meaningful features for the model. This step reduces dimensionality and focuses on the most informative parts of the data.

2. **Data transformation** modifies data into a format or structure suitable for analysis. This may involve operations such as scaling, encoding categorical variables, transforming time-series data into frequency domains, converting speech into transcribed text.

3. **Data labeling** assigns labels or **annotations** to the data, a necessary step for supervised learning tasks. Labeling can be performed manually, semi-automatically, or using pre-trained models. For example a dataset of medical images could be annotated by radiologists to identify which pixels in the images are those related to the diagnosis.

4. **Data augmentation** generates additional data by applying transformations to existing data, which can increase the diversity of the dataset without collecting new data. Common techniques include flipping, rotating, or cropping images, or introducing noise into textual or numerical data. For example with a dataset of anonymized handwritten text, augmentation could involve rotating or skewing the characters slightly to mimic different handwriting styles.

5. **Data synthesis** creates entirely new data points using statistical models or generative techniques. This is especially useful for generating training data when the original dataset is limited or sensitive.

It is important to mention that there can still be residual privacy risks after the data analysis stage. Sometimes, during the data analysis stage, the combination of possibly unlinked features in the pre-processed data, might reveal new features that could lead to the re-identification of individuals or more in general the disclosure of sensitive information that was not visible in the pre-processed dataset. It is important to perform privacy audits also on the extracted data, and eventually re-apply the same data sanitisation techniques that were used to process the raw data.

4.4 Data preparation

The final stage of the data engineering part of the MLOps is **data preparation**. This stage takes the features and augmented data generated during the data analysis stage and splits them into three distinct datasets: **training data**, **test data**, and **validation data**.

1. The **training data** is the largest subset of the prepared data, used to train the machine learning model by enabling it to learn patterns and relationships between features and target outputs. During training, the model iteratively adjusts its parameters to minimize the error between predictions and actual values using the training data.

2. The **test data** is a separate dataset that is withheld during training and used exclusively to evaluate the model's performance after training is complete. The test data provides an unbiased estimate of the model's generalization ability, ensuring it performs well on unseen data.

3. The **validation data** is a dataset used during model development to fine-tune hyperparameters and prevent overfitting. This dataset is typically used in conjunction with techniques such as cross-validation or grid search to optimize model performance. While the validation data is also used during the model training stage, it should remain independent to ensure unbiased hyperparameter tuning.

4.4.1 Splitting Techniques

Splitting the data into these subsets requires careful consideration to maintain integrity, privacy, and utility:

1. **Random Splitting**: Data is randomly divided into training, test, and validation sets. While simple, this approach works best when the data is uniformly distributed and free of inherent biases.

2. **Stratified Splitting**: If the dataset is imbalanced (e.g., in class labels), stratified splitting ensures that the proportions of different classes are preserved across the training, test, and validation sets. This is particularly important in datasets containing sensitive personal data, where representation can affect fairness.

3. **Temporal Splitting**: For time-series data, splitting is often done based on time order to prevent information leakage from the future into the training data.

4.5 Data after training and after deployment

We have covered the various types of data that ultimately shape the weights of an AI model. This data is primarily used during the pre-deployment phase to train and fine-tune the model. However, the data lifecycle in AI systems extends beyond deployment. When the AI model is in actual use, new data inputs are introduced as users interact with the system.

In this stage, personal data often plays a role. Depending on the type of AI system, users may provide personal data as input to prompt the system to perform a specific task. These inputs can range from images and text to other data types, each potentially containing sensitive information. The challenge here lies in managing these inputs responsibly. Depending on the context, user inputs might need to be minimized to protect personal data. For example, with medical images, some identifiers can be removed to protect patient privacy, but the essential diagnostic details must remain intact. In other cases, such as when using a chatbot for customer support, the AI system can employ filtering mechanisms to minimize personal identifiable information, ensuring that only necessary data is processed.

4.6 Good Data Management Practices

When data is the critical component of a system, adopting **good data management practices** is essential. This not only ensures the system's reliability and reproducibility but also addresses **cybersecurity concerns**, especially when handling sensitive or personal data. Below, we expand on key areas of data management with examples and references to best practices.

4.6.1 Data Quality and Data Appraisal

Data quality refers to the suitability of data for its intended purpose, which includes attributes such as accuracy, completeness, consistency, and relevance. High-quality data is essential for building reliable and fair machine learning models. It is an open question whether data quality is more important than data quantity and it is impossible to come to a conclusion that fits all various types of AI systems.

- **Bias Audit and Fairness Testing**: Regular audits should be conducted to identify and mitigate biases in the data. For example, fairness testing can be applied to ensure equitable representation of all demographic groups in the training dataset.
- **Data Appraisal**: Deciding what data to keep and what to reject is a crucial step. Irrelevant, redundant, or outdated data should be excluded, especially if it poses privacy risks or contributes to bias.

4.6.2 Data Versioning

Tracking the evolution of data is essential for ensuring **reproducibility** in machine learning systems. Reproducibility means that given the same data and the same code that was used to train a certain AI model in the past, it is possible to re-train the same model and obtain exactly the same model weights. With **data versioning** we are able to create snapshots of the data at a given moment in time. In the context of working with personal data this is even more important: a data subject might request to be removed from the original training dataset. With data versioning we can ensure that the new version of the data does not contain anymore the data from the data subject.

- **Solutions for Data Versioning**: Tools like **git annex** or **DVC (Data Version Control)** provide mechanisms to version and track changes in large datasets.
- **When Data Cannot Be Versioned**: For datasets that cannot be directly versioned due to size or sensitivity, it is critical to version the **metadata**, which includes information about data sources, transformations, and audit trails. For example the large image dataset *LAION* is a collection of URLs and metadata of the images, rather than the actual images.

4.6.3 Data management of AI model weights

While so far we have described the stage of the MLOps workflow before the actual training, model weights (i.e. the AI model, the output of the training stage) also undergo similar consideration when it comes to good data management practice. This is even more important in machine learning systems trained on personal data, since – depending on the model architecture – some elements from the training data might be memorised in the model. Goo practices include:

- **Model Cards**: Document the model's purpose, data sources, and performance metrics, ensuring transparency and accountability. This will be described more extensively in following chapters.
- **Model Versioning**: Version control systems should track changes in model weights and architectures, facilitating debugging and updates. Multiple versions of the same model might also exist not only as snapshots in time, but also in the dimension of **numerical precision**. This refers to the concept of **quantization of model weights**: instead of storing the weights with full numerical precision, a smaller number of bits can be used (quantization) balancing the trade-off between model size and model performance.

We will later also cover the fact that weights can also be published (open weights) following practices that are common with open source software. For further readings on the topic of data quality standards in AI systems, ISO/IEC 5259-2:2023 is a good starting point.

4.6.4 Data related considerations during training and inference in high-performance computing clusters or cloud

Training machine learning models involves unique data management challenges, such as balancing **fast data access** with the **risk of data leakage**. Specifically, a company might not have access to a dedicated high performance computing (HPC) cluster for training their AI model, which means that they need to use an HPC cluster shared with other users, or using cloud computing facilities. Depending on the policy of the cluster, cloud provider, data center, it might not be possible to ensure that – for example – the company training processes are the only ones using the cluster resources at a certain time. With intense training algorithms, the I/O speed of access to the data becomes critical. For example when training a neural network on GPU nodes, it is common to store part of the training data on the local temporary disk of the GPU node for faster access, to make sure that the GPU cards are not waiting idling for more data to come in. Local disks of GPU nodes should be able to enforce access control, but when training on sensitive data, it is important to consider that this adds another factor to the potential risk of data leakage. Companies working with very sensitive data, should consider working on clusters that are isolated from other users e.g. by creating an ad-hoc isolated network of computing nodes that are only accessible by the users of the company that is training the AI model. Similar data I/O and access control considerations are also valid during the inference stage (when an AI model is requested to produce some output) and these will be discussed later in the book.

4.6.5 Access Control and Good Cybersecurity Practices

We have already briefly touched on the issue of **access control** when working in systems that could potentially be shared with other users. Access control is fundamental when working with sensitive personal data both from the data storage perspective as well as computing. The reader should familiarise with common good practices for access control and make sure they are implemented when personal data is used to train or query an AI model.

- **Multi-Factor Authentication (MFA)**: All storage systems should be protected by MFA to minimize risks of unauthorised access.
- **Internet Isolation**: For highly sensitive data, disconnect computing and storage systems from the internet during processing.
- **Cloud and HPC Security Risks**: When using cloud providers or high-performance computing (HPC) centers, ensure that contractual agreements include robust data protection measures and that access to the cloud infrastructure is tightly controlled.

Encryption of the data both in transit and at rest is also a very good practice to mitigate the risks of data breaches. While encryption comes with the decryption overhead computing cost, there are also computing techniques that can natively work with encrypted data (see next chapter). Finally, it is also important to implement regular security audits and penetration tests to identify vulnerabilities of systems that are shared with other users or that are not isolated from the internet.

4.7 Summary

In this chapter we have covered the data engineering pipeline with focus on solutions that are specific to those cases dealing with personal data. We also covered good data management practices that should go along other cybersecurity practices with machine learning systems. These practices should be a core part of any MLOps workflow and are indispensable when working with personal data.

Exercise 4.3: Multiple choice questions

Question	Options
1. What is the primary goal of data sanitization in the pre-processing stage?	1) Increase the size of the dataset.2) Improve statistical properties.3) Reduce the risk of re-identification.4) Improve data visualization.
2. Which of the following is NOT a data acquisition method as per ISO/IEC 22989?	1) First-party data collection2) Web scraping3) Third-party data collection4) Data querying
3. What does the GDPR emphasize regarding the risks of data querying?	1) It explicitly prohibits data querying.2) It requires mitigating risks through DPIAs and purpose limitation.3) It allows data querying only for non-personal data.4) It does not address data querying.
4. In the context of MLOps, what is the role of stratified splitting during data preparation?	1) Ensures balanced class representation across subsets.2) Randomly splits data into training and test sets.3) Splits time-series data based on time order.4) Ensures all data is used for training.
5. What is the main purpose of perturbation in data sanitization?	1) Completely remove identifiers.2) Transform records while preserving statistical properties.3) Replace data with broader categories.4) Identify outliers in the data.
6. Which of the following is an example of a quasi-identifier?	1) Social security number2) Name3) Date of birth4) Fingerprint
7. What does ISO 27560 provide guidelines for?	1) Data versioning.2) Consent management.3) Feature engineering.4) Model evaluation.
8. Which method in data pre-processing is used to remove or transform direct identifiers like names?	1) Perturbation2) Suppression3) Filtering4) Anatomisation
9. What is a key risk when using web-scraped data for AI training?	1) Increased cost.2) Lack of proper legal basis under GDPR.3) Reduced statistical accuracy.4) Inability to preprocess the data.

10. What is the role of validation data in the MLOps workflow?	1) Train the AI model.2) Evaluate model generalization.3) Optimize hyperparameters.4) Provide real-world performance feedback.

Exercise 4.3. Solutions

Click to reveal solutions

1. **Answer:** 3) Reduce the risk of re-identification.

   Explanation: Data sanitization aims to mitigate privacy risks by removing or transforming identifiers and quasi-identifiers.

2. **Answer:** 2) Web scraping

   Explanation: Web scraping is not explicitly categorized as a data acquisition method in ISO/IEC 22989.

3. **Answer:** 2) It requires mitigating risks through DPIAs and purpose limitation.

   Explanation: GDPR highlights the need for DPIAs and compatibility with the original purpose when querying data containing personal information.

4. **Answer:** 1) Ensures balanced class representation across subsets.

   Explanation: Stratified splitting ensures that all subsets maintain proportional representation of different classes, especially in imbalanced datasets.

5. **Answer:** 2) Transform records while preserving statistical properties.

   Explanation: Perturbation involves introducing randomness to maintain data utility while protecting privacy.

6. **Answer:** 3) Date of birth

   Explanation: Quasi-identifiers, like dates of birth, can reveal identities when combined with other datasets.

7. **Answer:** 2) Consent management.

   Explanation: ISO 27560 provides a framework for managing consent in data workflows.

8. **Answer:** 2) Suppression

   Explanation: Suppression removes direct identifiers, such as names or IDs, from the dataset.

9. **Answer:** 2) Lack of proper legal basis under GDPR.

   Explanation: Web scraping of personal data often lacks valid consent or legal basis under GDPR.

10. **Answer:** 3) Optimize hyperparameters.

    Explanation: Validation data is used during model training to fine-tune hyperparameters and prevent overfitting.

5. Privacy Enhancing Technologies in AI systems

Learning outcomes

After completing this chapter, you will:

- Acquire the basic knowledge on privacy preserving machine learning with methods such as differential privacy, federated learning, and secure computations such as homomorphic encryption, and secure multiparty computation.
- Understand how synthetic data can help you create anonymous dataset and augmented the size of training data
- Evaluate the differences between PETs and what is important to consider when choosing the right approach

Privacy enhancing technologies (PETs) are a fundamental component of any digital system that is processing personal data. While the title of this chapter is PETs in AI systems, we will not cover basic techniques, such as k-anonymization and basic differential privacy that can be used in the preprocessing stage (see Appendix for a quick overview). Instead, our focus will be on PETs specifically designed for machine learning applications, emphasizing privacy-preserving techniques in the implementation of model training across various advanced methods. The main reference throughout this chapter is Chang et al. (2023).

5.1 The landscape of PETs

To cover extensively the landscape of PETs would require a book of its own. However, inspired by Garrido et al. (2022), there are various types of approaches to define a taxonomy of PETs along different *layers* that can involve secure computations, secure storage, communication, and more broadly governance and policies.

The layers defined in Garrido et al. (2022) are visualised in Figure 5.1. Secure storage or communication solutions are not covered in this book, instead we want to focus on those approaches that can **modify** the data along the MLops pipeline and these approaches can go under the general term of *Anonymisation*. Furthermore, these approaches, can be combined with *Secured and outsourced computing* where the actual computations or the computing environments can be further secured.

5.2 Privacy-Preserving Machine Learning Techniques

Definition of Privacy-Preserving Machine Learning

Privacy-preserving machine learning (PPML) encompasses a set of methods and technologies designed to enable the training and deployment of machine learning models on sensitive data while maintaining stringent privacy standards. PPML is essential when working with data that contains personally identifiable information or other sensitive details, ensuring that the data's integrity and confidentiality are preserved throughout the AI lifecycle.

A simple approach to work with personal data during the training stage of an AI model is to implement machine learning tasks on fully anonymized datasets. By definition, truly anonymous data is not personal data, and usual AI systems development and deployment techniques can be used when there is no personal data. However, ensuring full anonymisation is a challenging task, if even possible: while it is possible to minimise some of the personal data, full anonymisation can destroy important features and reduce the value of the original data. In such cases, privacy-preserving machine learning techniques are employed to enable secure training on sensitive data without sacrificing performance or privacy.

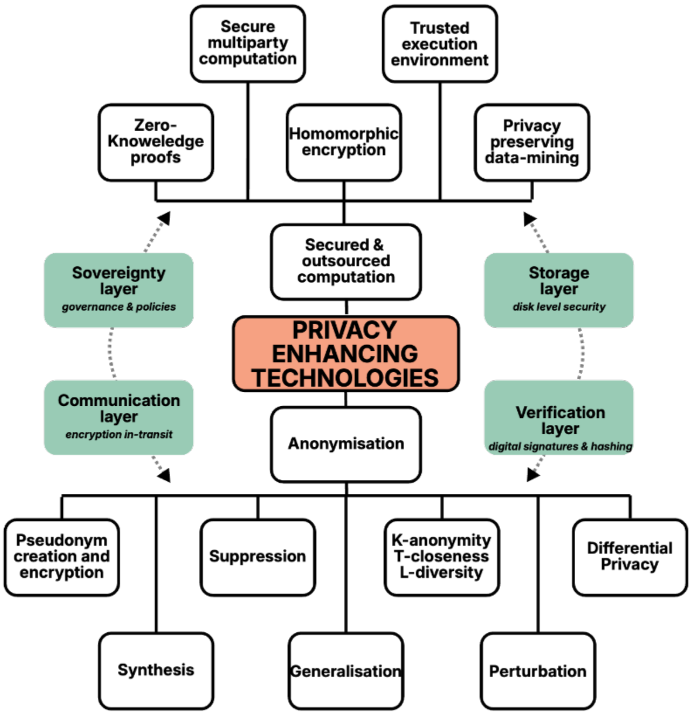


*Figure 5.1: **The landscape of PETs solutions** Various techniques can be combined to ensure the privacy of the entire system by altering the data or by improving security of storage, communication, and computations. The image is an evolution of Garrido et al. (2022).*

Our focus in this chapter is on techniques that ensure privacy when training on non-anonymized data. These methods account for privacy concerns throughout the entire training and inference process.

5.2.1 Differential Privacy

The most important PPML technique is differential privacy (DP). DP is a privacy-preserving approach that provides mathematical guarantees that individual data points in a dataset cannot be re-identified. By introducing carefully designed noise to the data or model parameters, DP makes sure that the inclusion or exclusion of any single individual's data does not significantly affect the overall performance of the analysis or AI model, preserving privacy even in sensitive datasets. While we introduced *perturbation* in the previous chapter as a privacy enhancing technique, differential privacy follows the same logic, however differential privacy can be added at each stage of the ML pipeline.

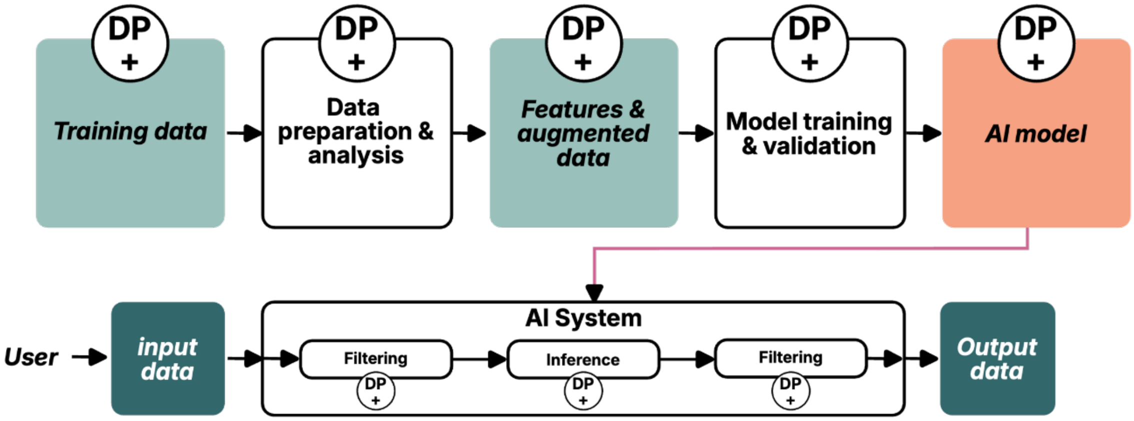


Figure 5.2: ***Illustration of Differential Privacy: Noise can be applied basically at any stage during training or in production, enhancing privacy at each stage of the AI development lifecycle.***

Some possible examples where DP can be applied: - **Training Data**: Noise is added directly to the dataset before training begins, limiting the model's capacity to memorize exact details of individual records. This is what we called *perturbation* in the previous chapter. - **Features**: In some cases, DP can be applied to specific feature sets within the training data, particularly when sensitive information is embedded in particular features. For example features are extracted from images of faces, and the DP is applied to further minimise the data. - **Model Parameters**: Noise can also be injected into the model's weights and gradients **during training**. This approach, known as *differentially private stochastic gradient descent (DP-SGD)*, limits how much each individual data point can influence the model, thus protecting sensitive information throughout the learning process. Weights can also be modified after training. - **Model Input/Output**: While this chapter focuses on privacy preservation in the training and development stages, DP can also apply noise to the final model output or even to the input submitted by the user of the AI system.

More complex approaches are constantly being developed. For example Wu et al. (2023) propose an approach that focuses on private predictions rather than private training; their approach is useful when the model cannot be retrained (e.g. a LLMs like OpenAI's GPT4) as it queries the model with different subsets of the sensitive data and aggregating the results locally.

However it is important to remember that while differential privacy provides robust privacy protections, it involves can involve at least two trade-offs to consider:

- **Computational overhead**: The process of adding noise, particularly when applied to model parameters, increases the computational demands of model training. This added complexity can impact training speed and may require additional computational resources.

- **Risk of Reduced Model Accuracy**: Differential privacy, especially when noise is added to gradients or weights, can impact model accuracy. This effect is more pronounced in deep learning models, where subtle variations in weights can lead to significant performance changes. The level of noise added must be carefully balanced.

5.2.2 Federated Learning

Federated learning (FL) is a PPML technique that enables training across decentralized devices or servers without transferring raw data to a central location. Instead of collecting data in one place, FL allows local devices to collaboratively train a global model by sharing only model updates, such as gradients or weights, while the actual data remains stored on each individual device. This approach ensures that sensitive information is not exposed outside its origin, significantly enhancing data privacy and security.

The federated process involves:

1. **Local Training**: Each device or server/computing node trains the model on its own dataset.
2. **Parameter Sharing**: Instead of data, only the local model parameters are sent to the central server.
3. **Aggregation**: The server aggregates these parameters to update the global model.
4. **Iterative Improvement**: This cycle is repeated multiple times until the model reaches a satisfactory level of accuracy.

Federated learning is widely used in sectors requiring stringent privacy standards, including:

- **Healthcare**: Medical institutions can collaboratively train predictive models across patient records without transferring sensitive health data.
- **Finance**: Banks and financial institutions can employ FL to share insights on fraud detection without exposing client information.
- **Mobile Applications**: FL allows companies like Google and Apple to improve smartphone AI features, such as predictive text and personalized recommendations, without uploading user data to a central cloud.

Despite its advantages, federated learning faces several challenges:

- **Communication Costs**: As devices frequently communicate model updates with the central server, network bandwidth can become a bottleneck, especially in resource-limited environments.

- **Model Performance Degradation**: The decentralized nature of federated learning can introduce data heterogeneity, where variations in data distributions across devices may impact model performance.

- **Security and Privacy Risks**: While FL enhances privacy by keeping data local, vulnerabilities remain. Model updates may still reveal information about the training data, posing a risk for attacks like membership inference or model inversion. Techniques such as differential privacy and secure aggregation can be combined with FL to mitigate these risks.

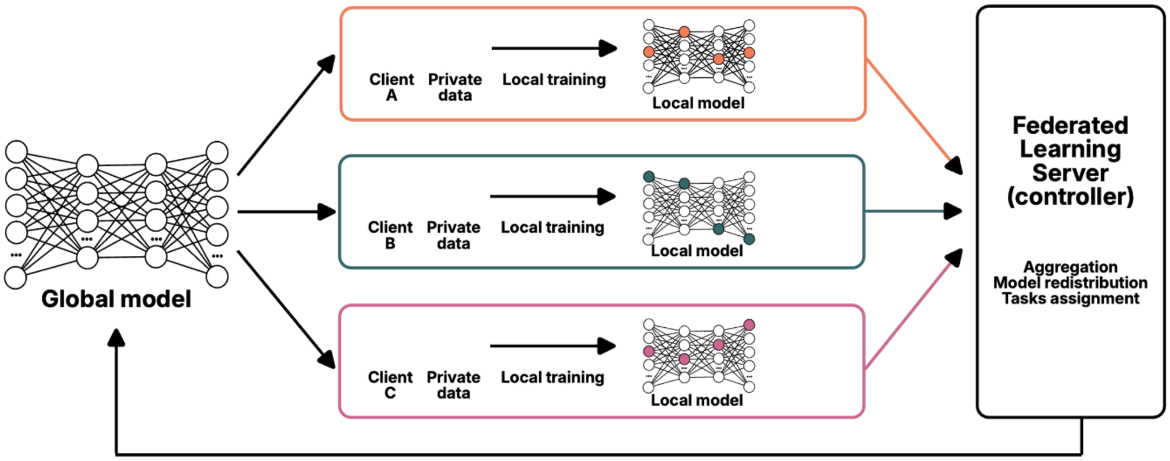


*Figure 5.3: **Illustration of Federated Learning** Local models are trained on each site (e.g. hospitals), and updates are aggregated on a central server, allowing for privacy-preserving model improvements across distributed data sources. The figure is a redrawn of a similar figure in NVIDIA Flare documentation.*

5.2.3 Synthetic Data Generation

Synthetic data generation is a privacy-preserving approach that involves creating artificial datasets that mimic the statistical properties of real data without directly using or exposing any actual data points. This technique can serve as a powerful privacy-enhancing tool in machine learning, particularly when sensitive data cannot be shared or directly used for model training.

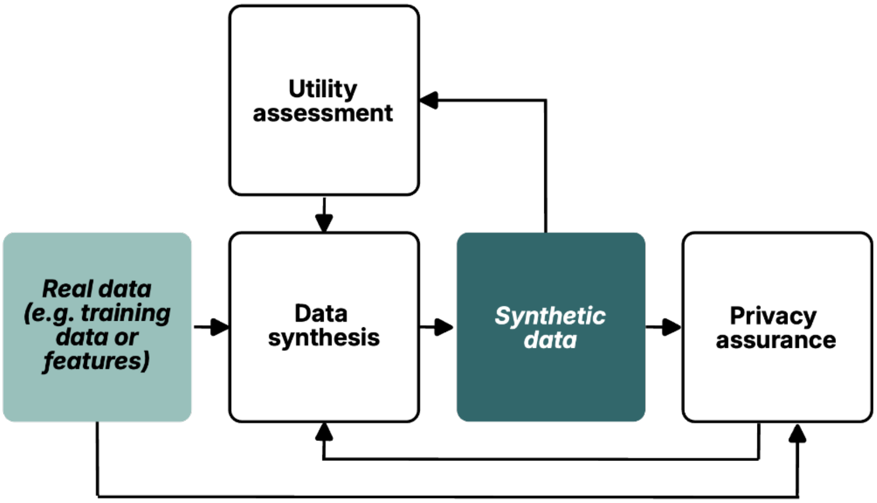


*Figure 5.4: **Data synthesis process** By learning patterns from the original data, synthetic data can be generated. It is important to assess the utility of the synthesised data and to ensure that there are no residual privacy risks. Figure re-adapted from Gal and Lynskey (2023).*

Synthetic data is typically generated using advanced machine learning models that learn and replicate the patterns and structures inherent in original datasets:

- **Generative Adversarial Networks (GANs)**: GANs consist of two neural networks, a generator and a discriminator, that work together to produce synthetic data. The generator creates new data points, while the discriminator evaluates how similar the generated data is to real data, iteratively improving the quality of synthetic data. See Tanaka and Aranha (2019).
- **Variational Autoencoders (VAEs)**: VAEs are another generative model that learns to encode data into a latent representation and then decodes it, generating new synthetic data points. VAEs are especially useful when generating data with complex structures or high-dimensional features.
- **Differentially private synthesis models**: By incorporating differential privacy mechanisms, these models add controlled noise during the data generation process, ensuring that the synthetic data remains statistically accurate while providing strong privacy guarantees.

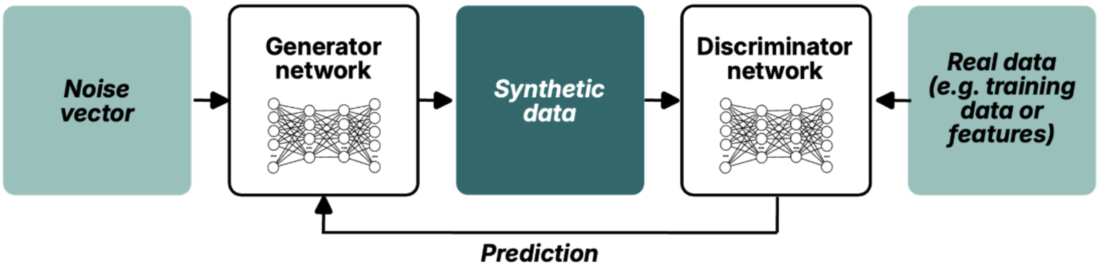


Figure 5.5: ***Data synthesis process using Generative Adversarial Networks (GANs)*** *With GANs we first train two neural networks, the generator that, given noise, it is able to transform it into a similar sample from the real dataset, and then the discriminator that has learned patterns from the data and helps preserving synthetic samples that are meaningful. Figure derived from Tanaka and Aranha (2019).*

Synthetic data generation is widely used across various fields where sensitive data is involved, and privacy concerns are high: for example in medical research synthetic data can be analysed and openly shared without compromising patient confidentiality. In retail and e-commerce, customer behaiouvr could be modelled with synthetic data. Synthetic data can also be used to enhance data diversity and mitigate possible biases that are in the data. Finally, in many scenarios lacking sufficient real-world data, synthetic data can be used to train and test AI systems, by simulating rare or high-risk situations not easily captured through traditional data collection.

Synthetic data generation has considerable advantages, but it also involves trade-offs:

- **Data utility vs. privacy**: While synthetic data retain essential statistical properties of the original dataset, it may not fully capture all nuances of the real data, potentially limiting model accuracy and reliability.
- **Risk of re-identification**: Synthetic samples could inadvertently resemble original data too closely, posing re-identification risks. Differential privacy and a strict validation processes help mitigate these risks, but depending on the case it could still be challenging to classify synthetic data as truly anonymous data.
- **Computational overhead**: High-quality synthetic data generation, particularly with GANs or VAEs, requires substantial computational resources, adding complexity to the data preparation and model training process.

The Habsburg AI Problem

The “Habsburg AI problem” refers to a challenge in synthetic data generation where the synthetic dataset unintentionally replicates distinctive, identifying features of the original dataset. This is named after the Habsburg jaw, a hereditary trait of the Habsburg royal family in Europe, which could be used to identify members within a population. In the context of privacy-preserving machine learning, if synthetic data too closely resembles specific individuals from the original data—akin to reproducing the “Habsburg jaw”—it risks re-identification, undermining privacy guarantees.

To address the Habsburg AI problem, synthetic data generation processes need to ensure that generated data reflects broader patterns without capturing rare or unique traits that could lead to identification of individuals. Techniques such as controlling similarity metrics or applying differential privacy to synthetic data models can mitigate this risk and improve the privacy-resilience of synthetic data.

Recently this issue has gained attention in the ML community in a paper by Shumailov et al. (2024). The paper highlights the phenomenon of “model collapse,” where generative AI models trained recursively on data generated by previous models progressively lose fidelity to the original data distribution. Over successive generations, the models forget low-probability (tail) events and converge to narrower, less diverse outputs, leading to irreversible degradation of performance. The findings stress that maintaining access to original, human-generated data is critical for avoiding these issues. While synthetic data seems very promising, developers should avoid relying solely on model-generated data during training or at least should establish robust mechanisms for verifying and maintaining the quality and diversity of synthetic training datasets.

5.2.4 Methods based on secure computations: Homomorphic Encryption & Secure Multiparty Computation

The methods described above broadly belong to the category of “Anonymisation” in the taxonomy by Garrido et al. (2022). Those techniques can also be combined with methods that related to the security of the computations, rather than the privacy that is introduced by manipulating the data.

Homomorphic encryption (HE) is a cryptographic technique that allows computations to be performed directly on encrypted data without requiring decryption. This capability is especially valuable in privacy-preserving machine learning, as it enables data processing in cloud environments and other external systems without exposing the underlying data. Homomorphic encryption is based on the principle that mathematical operations performed on encrypted data provide an encrypted result that, when decrypted, matches the result of operations as if they were performed on the plaintext data. Despite its advantages, homomorphic encryption has a few limitations, mostly due to its performance overhead (slower compute times) and increase complexity.

Somewhat related to federated learning, secure multiparty computation (SMPC) is another cryptographic approach that allows multiple parties to jointly compute a function over their combined inputs without revealing those inputs to each other. SMPC however differs from **federated learning** in several important ways:

- **Data Distribution and Control**: In federated learning, the model is trained locally on each participant’s device, and only the model parameters are shared with a central server. In contrast, SMPC does not involve local model training; instead, each participant’s data is split and distributed across multiple parties in a way that prevents any single party from reconstructing the original data.

- **Joint Computation Model**: While federated learning focuses on distributed model training across local datasets, SMPC is used for joint computations where all parties can contribute data without central aggregation. This enables SMPC to be highly secure for computations that require collaboration without data centralization.
- **Privacy Guarantees**: Both SMPC and federated learning aim to preserve privacy, but SMPC offers cryptographic guarantees that no individual party can view another's data, even indirectly. In federated learning, privacy is maintained by restricting data access and centralizing only model updates.

Similarly to HE, SMPC can also suffer from increased computational complexity (more resources needed and longer computing times) and the additional communication costs: Securely sharing data fragments and conducting joint computations involve substantial communication overhead, which can affect performance, especially in real-time or large-scale applications.

5.3 Comparison of Privacy-Preserving Techniques

Each privacy-preserving machine learning (PPML) technique discussed offers unique strengths and trade-offs. The following table compares federated learning, differential privacy, homomorphic encryption, secure multiparty computation, and synthetic data generation based on privacy guarantees, computational cost, use cases, and limitations.

Technique	Privacy Guarantees	Computational Cost	Use Cases	Limitations
Differential Privacy	High - Adds noise to data, features, or model outputs to ensure individual data cannot be re-identified	Moderate to High	Government data releases, customer data analysis	Privacy-utility trade-off; may reduce model accuracy due to added noise
Federated Learning	Medium - Only model parameters are shared, data remains local	Moderate	Mobile apps, finance, healthcare	Vulnerable to inference attacks, requires frequent communication between devices
Synthetic Data Generation	High - Protects privacy by generating data that mirrors real data patterns without actual information	Moderate	Medical research, finance, regulatory reporting	Risk of re-identification if synthetic data closely resembles real data, potential reduction in data utility
Homomorphic Encryption	Very High - Allows computation on encrypted data without decryption	Very High	Secure cloud computation, medical imaging	Computationally intensive, high latency, challenging for real-time applications
Secure Mult	Very High - Ensures data confidentiality by distributing data	High	Cross-institutional healthcare	High communication overhead, complex

Technique	Privacy Guarantees	Computational Cost	Use Cases	Limitations
iparty Computation	fragments for joint computation		research, fraud detection	implementation, requires trusted setup

Each technique contributes uniquely to a comprehensive PPML system. **Federated learning** prioritizes data locality, **differential privacy** focuses on noise injection for individual privacy, **homomorphic encryption** allows secure computation in untrusted environments, **secure multiparty computation** enables collaborative computation without central

5.4 Privacy-preserving techniques in the AI development lifecycle

Embedding PETs throughout the AI development lifecycle is essential to ensure data protection at every stage. Different PETs offer unique advantages and might be useful only on specific phases of AI development. Choosing the appropriate PETs depends heavily on the specific context in which an AI system is deployed. Combining multiple PETs can often achieve a balance between privacy and usability. If you are unsure which PET you want to test first, differential privacy is the safest bet.

Exercise 5.1: PETs mapped on the MLOps workflow

Consider the processing blocks of Figure 3.3: which of the presented PETs can be added to each processing block?

Note for the instructor: this can be a lengthy exercise that can also be assigned as homework. There is not a simple solution, since differences also depend on the type of data / AI system that one is considering. The instructor can also assign different systems to different groups of students based on the property of the data (e.g. one group considers an AI system that is processing tabular data, another group could work with text data, another with images, and so on).

5.5 Summary

In this chapter we covered the most important PETs that can be used when training AI models with sensitive data. Some of these approaches can also be adopted during the production phase of an AI system, i.e. - even if you are not training an AI model from scratch - techniques such differential privacy can also ensure that input data (e.g. prompts of an LLM AI system) is also transformed before querying the AI model. This chapter also concludes module 2, at this stage the training data is minimised - and sometimes even fully anonymised. In the next module we will focus on the development and deployment stage of the AI model and AI system.

Exercise 5.2: Multiple choice questions

Question	Options
1. What is the main goal of privacy-preserving	1) To improve model accuracy.2) To train and deploy AI models on sensitive data while maintaining privacy.3) To

machine learning (PPML)?	centralize data for model training.4) To reduce computational costs of training.
2. Which of the following describes federated learning?	1) A method to anonymize training data before model development.2) A decentralized approach where raw data stays local and only model updates are shared.3) A technique for adding noise to model parameters.4) A cryptographic method for secure computations on encrypted data.
3. What is a key challenge associated with federated learning?	1) Ensuring data quality.2) High communication costs due to frequent model updates.3) Centralized data storage.4) Lack of privacy protection for model parameters.
4. Differential privacy protects individual data by:	1) Encrypting all data during processing.2) Adding carefully calibrated noise to data or model parameters.3) Splitting data into shares distributed across multiple parties.4) Generating synthetic data.
5. What is the primary limitation of homomorphic encryption in machine learning?	1) It cannot handle computations on encrypted data.2) It is computationally expensive and introduces latency.3) It requires centralized data storage.4) It is incompatible with differential privacy.
6. Secure multiparty computation (SMPC) differs from federated learning by:	1) Training models locally on participant devices.2) Using cryptographic protocols for joint computation without central aggregation.3) Adding noise to protect data privacy.4) Generating synthetic datasets for privacy.
7. Which PPML technique generates artificial datasets mimicking real data?	1) Differential privacy2) Homomorphic encryption3) Synthetic data generation4) Secure multiparty computation
8. A major risk of synthetic data generation is:	1) High computational cost.2) Data re-identification if synthetic data closely resembles real data.3) Inability to capture statistical patterns of the original dataset.4) Lack of support for high-dimensional data.
9. Which PPML technique is most suitable for distributed data across multiple hospitals?	1) Homomorphic encryption2) Federated learning3) Synthetic data generation4) Secure multiparty computation
10. What does the “Habsburg AI problem” refer to?	1) Challenges in training models with encrypted data.2) Synthetic data replicating unique traits from the original data, risking re-identification.3) Communication bottlenecks in federated learning.4) Loss of statistical utility in synthetic data.

Exercise 5.2. Solutions

Click to reveal solutions

1. **Answer:** 2) To train and deploy AI models on sensitive data while maintaining privacy.

   Explanation: PPML ensures privacy throughout the AI lifecycle while enabling the use of sensitive data.

2. **Answer:** 2) A decentralized approach where raw data stays local and only model updates are shared.

   Explanation: Federated learning trains models locally and shares updates instead of data, enhancing privacy.

3. **Answer:** 2) High communication costs due to frequent model updates.

   Explanation: Frequent parameter sharing between devices and the server can increase network overhead in federated learning.

4. **Answer:** 2) Adding carefully calibrated noise to data or model parameters.

   Explanation: Differential privacy uses noise to protect individual data while maintaining overall utility.

5. **Answer:** 2) It is computationally expensive and introduces latency.

   Explanation: Homomorphic encryption is resource-intensive, limiting its use in real-time applications.

6. **Answer:** 2) Using cryptographic protocols for joint computation without central aggregation.

   Explanation: SMPC ensures privacy by distributing data fragments and performing secure joint computations.

7. **Answer:** 3) Synthetic data generation

   Explanation: Synthetic data generation creates artificial datasets that mimic real data for privacy-preserving purposes.

8. **Answer:** 2) Data re-identification if synthetic data closely resembles real data.

   Explanation: Synthetic data can inadvertently replicate real data patterns, risking privacy breaches.

9. **Answer:** 2) Federated learning

   Explanation: Federated learning allows distributed training across hospitals without sharing sensitive patient data.

10. **Answer:** 2) Synthetic data replicating unique traits from the original data, risking re-identification.

    Explanation: The Habsburg AI problem highlights privacy risks when synthetic data too closely mirrors original datasets.

6. Secure code development for AI Systems

Learning outcomes

After completing this chapter, you will:

- Understand the principles of secure software development, including the Secure Software Development Lifecycle (SDLC) and privacy by design in practice.
- Explore best practices for maintaining secure code, such as version control, code reviews, and continuous integration.
- Identify common security vulnerabilities in popular machine learning tools and understand how to ensure reproducibility through practices like containerization.

After covering the fundamental principles of personal data protection, AI, cybersecurity, data preparation, and privacy-preserving machine learning, this chapter focuses on essential practices of secure AI systems implementation with the Secure Software Development Lifecycle (SDLC) and practical applications of privacy by design during code development.

This chapter also covers concepts like version control, continuous integration, and code reviews. While common to software development, these practices are essential here to ensure the transparency and accountability of an AI system trained on personal data.

The chapter further examines the security of common machine learning frameworks and touches on the topic of “reproducibility” to ensure that AI models remain secure and consistent across different platforms, and to ensure the possibility of auditing past versions of an AI system.

6.1 Secure MLOps: experimentation.

If we go back to our secMLOps pipeline figure, we are now going to cover the stages of “experimentation” and “production”, although the production stage only focuses on the training of the final ML model, without the deployment phase.

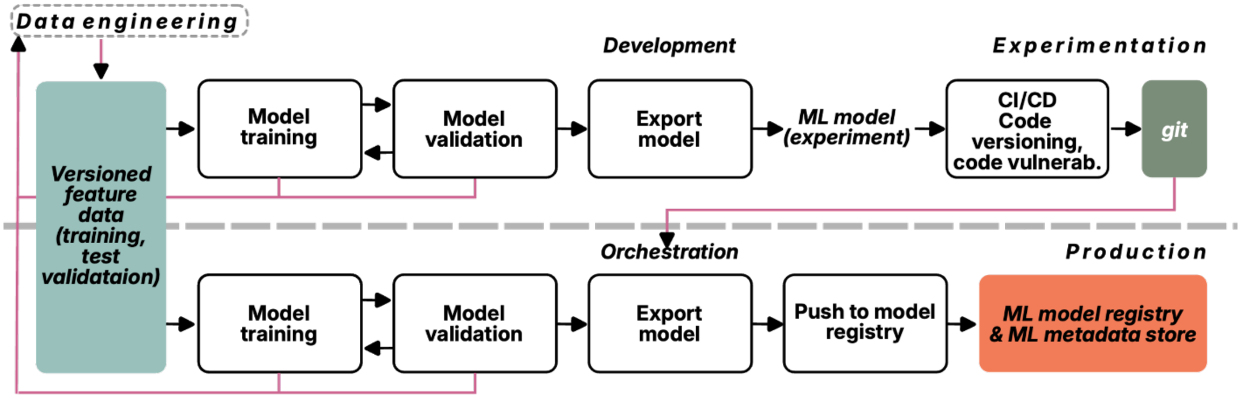


Figure 6.1: ***The data engineering stage of the secMLOps workflow*** *Please refer to the text, for a detailed explanation on the figure.*

This is the typical scenario of a team of developers working on training an AI model from scratch:

1. Prepare the code and the related dependencies to perform the actual training.
2. Test the code with a small dataset by running the “model training” and “model validation” stages while running monitoring tools to assess the performance of the code.
3. Repeat steps 1 and 2 with larger amounts of data and larger amounts of computing nodes. Depending on the computational resources (local cluster, shared HPC cluster, cloud computing) different strategies might need to be adopted to bring data and code to each computational node
4. Trained models are exported and versioned along with versions of the code.

6.2 Secure Software Development Lifecycle in AI Systems

Designing AI systems, especially those trained on personal data, demands a comprehensive integration of security throughout the Software Development Lifecycle (SDLC). The Secure Software Development Lifecycle (SSDLC, although it is almost always written SDLC) enhances the traditional SDLC by embedding proactive and reactive security measures at every phase. This section is based on Fujdiak et al. (2019).

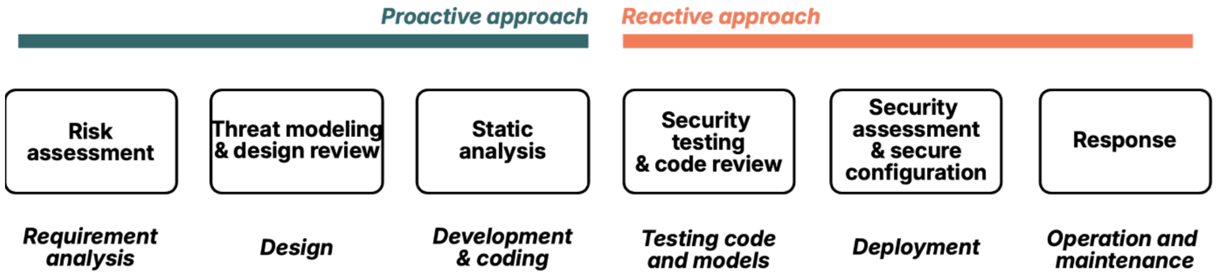


*Figure 6.2: **The secure software development lifecycle** Different stages of the AI lifecycle can be mapped into the software development lifecycle. It should also be noted that with sensitive data, often the “proactive approach” stages are happening in secure computing environments, while the stages related to reactive approach are happening in the live/production environment which has a larger attack surface*

6.2.1 Proactive vs. Reactive Security Approaches

The SSDLC incorporates both proactive and reactive approaches:

1. **Proactive Security**: Focuses on preventing vulnerabilities during early SDLC stages. Measures include security training, threat modeling, and defining security requirements before coding begins. Proactive efforts minimize resource costs by addressing issues before they escalate.

2. **Reactive Security**: Addresses security post-development or during deployment through activities like penetration testing, dynamic analysis, and incident response planning. While critical, reactive approaches often add complexity and resource demands if not preceded by proactive measures.

6.2.2 Blind analysis: separating development teams?

When developing AI systems trained on personal data, the SSDLC must account for the risks associated related the sensitive data being used: unauthorized access, data breaches, and non-compliance with GDPR. From the software developer perspective, these are risks related to data access, rather than to software vulnerabilities per se. If access to the personal data is strictly controlled, a possible alternative is for the developers teams to split in two: one team develops the software using only synthetic data that mimics the properties and structure of the real data. A second team – with access to the sensitive data – obtains the software from the first team and run it in secure processing environments. Separating the two development teams further minimises potential risks associated with the sensitive nature of the data. This approach is sometimes called Role-Based Access Control, and mimics what is sometimes done in science to remove potential biases from those who analyse scientific data (MacCoun and Perlmutter 2015).

6.3 Version control, testing, continuous integration, code coverage

Version control, testing, continuous integration, and code coverage are essential practices in modern software development that help teams work efficiently and ensure the quality of their code.

- **Version control** is a system that keeps track of changes made to files, allowing developers to collaborate, track history, and revert to previous versions if needed; **Git** is a popular tool for this.
- **Testing** involves running checks on the code to ensure it works as expected and avoids bugs; this can include small tests for specific pieces (unit tests) or tests for the whole system.
- **Continuous integration (CI)** is a practice where code changes are automatically tested and integrated into a shared codebase frequently, ensuring new changes don’t break the system.
- **Code coverage** measures how much of the code is tested by your tests, showing areas that may need more attention. Together, these practices make software development smoother, more reliable, and less error-prone. In the next chapter we will look at another type of testing, not the code unit tests to ensure the validity of the code, but the actual testing of the AI model and its resilience to potential attacks.

6.4 Other Good Practices for Software Security in AI Systems Development

In general, as in any software development scenario, it is critical to remember that a larger *attack surface* increases the likelihood of encountering risks due to weak software security. This is especially relevant in the development of AI systems, where **supply chain vulnerabilities** must be carefully addressed. These vulnerabilities not only encompass risks related to the libraries used for training and deploying AI models but also extend to the reuse of pre-trained machine learning models and serialization libraries.

6.4.1 Libraries for ML Training and How to Check for Vulnerabilities

Some of the most commonly used libraries for machine learning training include **TensorFlow**, **PyTorch**, **Scikit-learn**, and **Keras**. While these libraries are widely adopted and regularly updated, they may still contain vulnerabilities that could be exploited.

Mitigation strategies:

1. **Check for Known Vulnerabilities**:
   Use tools such as **Dependency-Check**, **Safety**, or **Snyk** to scan your ML environment for known security issues in dependencies.

2. **Stay Updated**
   Regularly update libraries to the latest stable versions to include security patches.

Monitoring GitHub releases or PyPI feeds for your dependencies helps avoid lagging behind on critical fixes.

3. **Verify Source Integrity**
   Always obtain libraries and models from official repositories or trusted sources. For extra assurance, verify checksums or digital signatures when provided.

4. **Monitor Security Advisories**
   Subscribe to mailing lists, GitHub security advisories, or RSS feeds of the ML libraries you use. Projects like TensorFlow maintain detailed CVEs and release notes that flag patched vulnerabilities.

5. **Monitor Dependencies and Supply Chain** ML pipelines often involve dozens of transitive dependencies. Use tools like pipdeptree or pip-audit to map and assess these. Consider pinning dependency versions in `requirements.txt` or `pyproject.toml` for reproducibility, but weigh the trade-off against missing out on important security patches.

6. **Inspect Network Traffic** Some ML libraries or training pipelines may download additional data (e.g. weights, datasets) automatically or have telemetry enabled. Use tools like `netstat`, `tcpdump`, or a firewall to monitor unexpected outbound requests during training and inference phases.

7. **Run Software in Sandboxed or Isolated Environments** Use **containers** (e.g., Docker or Singularity) or **virtual environments** to isolate the ML environment from your host system. Containers also help manage dependencies securely and ensure reproducibility.

6.4.2 Risks of Reusing Pre-trained Machine Learning Models

Pre-trained models, such as those available in repositories like **Hugging Face**, **TensorFlow Hub**, or **ONNX Model Zoo**, are valuable for saving time and computational resources. However, they introduce significant risks that may compromise the security and integrity of AI systems (for a detailed overview, please see Goldblum et al. (2022) and Wang et al. (2022)):

1. **Backdoors and Clean-Label Poisoning** Attackers can embed backdoors in models during training, which are triggered only under specific input conditions. In *clean-label attacks*, the poisoned data appears legitimate and retains correct labels, making it difficult to detect.
2. **Transfer Learning Exploits** Many real-world attacks exploit transfer learning by crafting poisoned data or model components that persist even after fine-tuning. These attacks rely on the reuse of common model architectures and pre-trained feature extractors.
3. **Unknown Provenance and Poisoned Pre-training Data** Lack of transparency in the datasets or methods used during pre-training may introduce subtle biases, vulnerabilities, or targeted behaviors that are hard to detect post-hoc.
4. **Tampered Models and Model-Reuse Attacks** Downloaded models may be directly tampered with to include malicious logic. Attackers may also poison the training of models expected to be reused by others — anticipating how those models will later be adapted.

Mitigation Strategies:

- Use pre-trained models from **trusted sources** with clear documentation and version history.

- Perform **behavioral testing** (e.g. outlier detection, gradient inspection) to uncover unexpected behaviors.
- Fine-tune models with robust methods (e.g. differential privacy, certified defenses) and validate with multiple surrogate datasets.
- Check **digital signatures** or hashes when distributing or ingesting model files.

6.5 Risks of Serialization Libraries

Serialization libraries such as **Pickle** and **Joblib** in Python are convenient for saving and loading models. However, they carry significant risks, particularly when handling models from external or untrusted sources:

1. **Arbitrary Code Execution**: Pickle can execute arbitrary Python code embedded within serialized objects. If a malicious actor crafts the file, simply unpickling it may compromise the system.
2. **Tampering and Hidden Logic**: Serialized models may be manipulated to subtly change predictions, trigger specific outputs, or install unauthorized functionality.
3. **Lack of Integrity Checks**: Formats like Pickle do not include built-in mechanisms for detecting file corruption or unauthorized modification.

Safer Alternatives and Best Practices:

- **Avoid** using insecure serialization formats like Pickle for untrusted or external files.
- Prefer **ONNX**, **TorchScript**, or **JSON-based formats** with stronger validation mechanisms.
- Always validate the **content** and **structure** of deserialized models before use.
- Use **cryptographic hashes or digital signatures** to verify the integrity and authenticity of serialized files.

6.6 Additional Considerations

Beyond these risks, there are also a few potential risks that should also be considered when developing AI models:

- **Federated Learning Vulnerabilities**: Poisoned data or model updates in decentralized training can lead to widespread model corruption, especially when adversaries control edge devices.
- **Broader Attack Objectives**: Poisoning can be used to subvert fairness, reduce overall accuracy, or specifically degrade performance for certain subgroups or individuals.
- **Advanced Techniques**: Methods such as **bilevel optimization**, **feature collision**, and **influence functions** allow for more effective and targeted poisoning, including attacks that remain invisible after standard model auditing.
- **Scalable and Automated Attacks**: Generative approaches are being explored to automate poisoning across large datasets or at industrial scale.

Vulnerability Story: the case of ShellTorch

TorchServe, an open-source model-serving library for deploying PyTorch models, plays a critical role in AI infrastructure for some of the world's largest organizations. However, researchers discovered severe vulnerabilities collectively named "ShellTorch," which exposed thousands of TorchServe servers to Remote Code Execution (RCE) attacks. These vulnerabilities arose from issues like unauthenticated management API access, misconfigured Server-Side Request Forgery (SSRF), insecure deserialization of YAML files via the SnakeYAML library, and the infamous ZipSlip directory traversal flaw. Attackers could exploit these flaws to gain complete control over the server, upload malicious models, extract sensitive data, and even alter AI model behavior. The severity of these vulnerabilities (CVSS scores as high as 9.9) made them a major security concern, especially since some Fortune 500 companies were affected. The vulnerabilities showed how misconfigurations and insecure practices in AI infrastructure can lead to devastating consequences.

To mitigate such risks, the key lessons from ShellTorch are about the importance of secure configurations, proper access controls, and cautious use of third-party libraries. First, users must ensure management APIs are not exposed to external networks without authentication. Second, inputs like YAML files should always be validated using secure parsers like SafeConstructor in SnakeYAML. Third, enabling features like domain allow-lists for model registration can prevent SSRF attacks. Additionally, regularly updating dependencies to patched versions (e.g., TorchServe version 0.8.2 or higher) and using vulnerability scanners are crucial steps. The ShellTorch incident demonstrates the critical need for combining AI innovation with robust security practices to prevent malicious exploitation of AI infrastructure (Kaplan, Katz, and Lumelsky 2024).

6.7 Reproducibility in AI Systems

Reproducibility in AI systems, particularly those trained on personal data, is critical for maintaining transparency, ensuring compliance, and enabling updates to models when data changes. Reproducing the exact code, dependencies, and configurations allows us to recreate AI models faithfully. This capability is fundamental for scenarios such as responding to data subject requests, where specific personal data may need to be removed. With reproducible pipelines, the same model can be retrained on an updated dataset, ensuring that the AI system continues to function as intended without reliance on outdated or removed data.

In Python, tools like **conda** and **pyenv** are commonly used to manage dependencies and environments for reproducible code execution. However, containers like **Docker** and **Singularity/Apptainer** are increasingly the standard for AI development and deployment, especially in environments like **HPC clusters** (using Slurm) or **cloud-based orchestration systems** (e.g., Kubernetes). Containers encapsulate the entire runtime environment, including code, libraries, and dependencies, making it easy to deploy and share reproducible pipelines across diverse systems. While Docker is widely used, it poses risks due to its reliance on root access, which can be problematic in shared computing environments. In such cases, container systems like **Singularity** or **Apptainer** are preferred, as they are designed with security in mind, avoiding the root access issue while maintaining portability and ease of use.

6.8 Secure Processing Environments for Training AI Systems with Personal Data

We have covered various aspects of secure and responsible code development in the context of ML and personal data, but secure code is tightly coupled with the computing infrastructure where the code is going to run. Secure processing environments, often referred to as **Trusted Research Environments (TREs)**, are controlled workspaces designed to ensure that the personal data used for training AI systems is processed securely. These environments implement strict access controls, encryption, and monitoring to minimize the risk of data breaches, unauthorized access, or misuse. While TREs are essential for compliance with regulations like GDPR, they also bring significant challenges when handling **distributed learning scenarios**.

6.8.1 Distributed Learning in Secure Environments

What is distributed learning? In distributed learning (Huyen 2022), training an AI model often requires splitting the workload across multiple computing nodes. This approach is necessary when: 1) **data is too large to fit into one node's memory**; 2) the **model is too large to fit in a single node**; or 3) when adopting **pipeline parallelism** to optimize computations.

Whether we use **data parallelism** (the dataset is split across multiple nodes, and each node trains a local copy of the model on its subset of data) or **model parallelism** (the model is too large to fit into a single computing node, so the model itself is split across multiple nodes) or a mix of both, the training process can become more complex in TREs due to the increased security in the communication between nodes and the potential cost of reserving a large computing cluster isolated from the internet only for the current training process, to avoid sharing the infrastructure with other users like it is common with HPC clusters. The cost and complexity of setting up and maintaining a distributed TRE can be significant, especially for smaller organizations or research teams. The **scalability** of TREs may also be a challenge for large-scale AI training requiring extensive computational resources.

6.9 Beyond security: how to choose the right ML model?

To choose the best ML model, it is important to start with the simplest model, avoid the state-of-the-art trap, consider human biases, evaluate performance trade-offs, understand model assumptions, and consider if an ensemble model might be appropriate. Chapter 6 from Huyen (2022) is a great starting point to learn more about what to consider when developing an AI model from scratch. In the context of personal data and AI models (see European Data Protection Board (EDPB) (2024)), it is important to consider the guiding principles of the GDPR – especially transparency and accountability – when choosing the best ML architecture. Explainable AI methods might be more favorable than black-box algorithms, but this might result in a compromise when it comes to the performance of the model. Like we saw with anonymisation techniques, it is always a balance between protecting the privacy of the data subjects in the training set, and the performance of the AI model that is developed. For more information on the topic, see Appendix "Machine Learning algorithms and their explainability" which summarises Leslie et al. (2024).

6.10 Beyond the AI model: developing the AI system

While the main goal of MLOps workflow is to train the best performing AI model, it is also fundamental to consider the secure development of the AI system as a whole. If the AI model is like a strong powerful car engine, it won't be able to do anything unless we build the rest of the car around it. This means that also the *inference* part of the AI system (how the model is queried and how predictions are made) need to be developed securely, as well as the actual *interface* part: how the inputs will be gathered and turned into queries for the model, and how outputs will be processed.

When it comes to personal data at the level of the AI system, inputs might bring new personal data that we do not necessarily want to use to query the model. And similarly outputs might contain unwanted personal data. While it is difficult to provide a general solution that works for all AI systems, the reader can already think of a few ways to filter and monitor unwanted inputs/outputs in the context of personal data (including attacks from malicious users, which we will cover next).

While this book is mostly targeted towards developers and deployers of AI models, many SME might rely on third party AI models/systems to form the core of the company managed AI system. With third party AI systems all the careful minimisation techniques and best practices for secure processing are delegated to the third party. Was the training of the AI model performed by the third party lawful according to the GDPR? This scenario is covered in the recent European Data Protection Board (EDPB) (2024), but the guidelines recommend a case-by-case assessment.

6.11 Conclusions and where to learn more about this

This section has presented multiple tools and practices that should be part of the daily skillset of an AI engineer. In the stages of the AI lifecycle (ISO 5338, ISO 22989), development happens at the "Design and Development" stage. As we saw in this chapter, the development stage involves the training data and the validation data to optimise the model hyperparameters. While we described the importance of code testing and code coverage, the actual testing of the AI system happens on the next "Verification & Validation" stage of the AI lifecycle. Many of the tools presented in this chapter would require their own dedicated course. We collect here an overview of existing open-source courses that can help the student to learn deeper some of the tools presented here.

Hint for instructors

Depending how the instructor is developing the course, this module is the one that could be expanded with some of the lessons described below. Once the instructor has decided what type of infrastructure they can teach with (are the learners just using their laptops? Do they have access to a *slurm* or *kubernetes* cluster? what are the sizes of the data used for training and evaluation?) they can then decide if it's important to focus on some ML tools like *scikit-learn*, or MLOps tools like MLflow, or good coding practices beyond ML like git version control and continuous integration. There is a lot to learn/teach to beginners, so we do hope the instructor does not feel discouraged.

Version control with Git

- CodeRefinery lesson "Introduction to version control with git"
- CodeRefinery lesson "Collaborative distributed version control". (Video versions of these lessons are available on CodeRefinery YouTube channel).
- Alternative: Software carpentry's "Version control with git"

Testing, continuous integration, and code coverage

- CodeRefinery lesson "Automated testing" covers testing, continuous integration, and code coverage

Machine learning frameworks: sci-kit learn, pytorch, etc

- Software Carpentry's "Introduction to Machine Learning with Scikit Learn"
- Learn PyTorch for Deep Learning: Zero to Mastery book

Containers, HPC

- Software Carpentry's Introduction to Docker
- CodeRefinery's "Containers on HPC with Apptainer"
- Software Carpentryäs "Introduction to High-Performance Computing"

Other useful tools for ensuring reproducibility of ML workflows

- DVC (Data Version Control) https://dcv.org/ is a tool for versioning and managing machine learning projects, enabling efficient tracking of datasets, models, and experiments within version control systems like Git. Start with their DVC tutorial

- Weights & Biases (W&B) https://wandb.ai/ is a platform for tracking, visualizing, and managing machine learning experiments for model development and collaboration. W&B server can also be run locally, to ensure the confidentiality of the development process. Start with the W&B Tutorial
- MLflow https://mlflow.org/ is an open-source platform for managing the machine learning lifecycle, including experimentation, reproducibility, deployment, and model registry. MLflow tutorial

There are many more courses from commercial learning platforms using commercial tools provided by cloud infrastructure companies like Amazon, Google, Microsoft, Snowflake. The learner can explore what best fits their learning goals and decide to opt in into a paid course or certification from these providers.

6.12 Summary

In this chapter we briefly covered the most important concepts related to (secure) software development, with focus on good practices in machine learning. While each of the described practices could have a chapter (or a course!) of its own, the goal here was to motivate the learner to understand what might be missing in their skillset and what practices should be adopted. The next module we will focus on testing and validation, two necessary steps before finally deploying the AI model/system.

Exercise 6.1: Multiple choice questions

Question	Options
1. What is the primary focus of the Secure Software Development Lifecycle (SDLC) in AI systems?	1) To ensure faster code deployment.2) To embed security measures throughout the development process.3) To reduce computational costs.4) To replace traditional SDLC approaches.
2. Which of the following is an example of proactive security in the SDLC?	1) Penetration testing.2) Threat modeling before coding begins.3) Incident response planning.4) Fixing vulnerabilities after deployment.
3. What is a key benefit of splitting development teams when working with sensitive personal data?	1) Faster code iteration.2) Minimizing risks associated with unauthorized data access.3) Ensuring compliance with code review practices.4) Reducing computational resource needs.
4. Which tool is commonly used for version control in software development?	1) TensorFlow.2) Git.3) Docker.4) PyTorch.
5. What is the main purpose of continuous integration (CI)?	1) Track changes to files over time.2) Automatically test and integrate code changes into a shared codebase.3) Reduce the attack surface of software.4) Improve serialization processes.
6. What is a major risk of reusing pre-trained machine learning models?	1) Increased training time.2) Hidden backdoors or poisoning during inference.3) Lack of scalability.4) High memory requirements.
7. How can developers reduce risks associated with serialization libraries like Pickle?	1) Use formats with stricter validation mechanisms like ONNX or JSON.2) Avoid version control for serialized files.3) Skip integrity checks

	for serialized files.4) Always use older versions of serialization libraries.
8. What was a critical vulnerability in the “ShellTorch” case?	1) Unauthenticated management API access.2) Issues with federated learning.3) Lack of synthetic data for testing.4) Misconfigured continuous integration pipelines.
9. What is the primary advantage of using containers like Singularity over Docker in shared environments?	1) Faster deployment times.2) Avoiding root access issues.3) Better GPU support.4) Automatic code testing.
10. What is the purpose of Trusted Research Environments (TREs)?	1) To enhance code reproducibility.2) To ensure sensitive data is processed in secure and controlled workspaces.3) To reduce computational overhead in AI training.4) To simplify the SDLC process.

Exercise 6.1. Solutions

Click to reveal solutions

1. **Answer:** 2) To embed security measures throughout the development process.

   Explanation: The SDLC integrates security measures proactively and reactively to ensure the secure development of AI systems.

2. **Answer:** 2) Threat modeling before coding begins.

   Explanation: Proactive security involves identifying and mitigating potential threats during early stages of development.

3. **Answer:** 2) Minimizing risks associated with unauthorized data access.

   Explanation: Splitting development teams reduces risks by controlling access to sensitive data.

4. **Answer:** 2) Git.

   Explanation: Git is a widely used tool for tracking changes in code and enabling collaboration.

5. **Answer:** 2) Automatically test and integrate code changes into a shared codebase.

   Explanation: Continuous integration helps ensure that changes do not break the shared codebase.

6. **Answer:** 2) Hidden backdoors or poisoning during inference.

   Explanation: Pre-trained models can be compromised with hidden vulnerabilities like backdoors or data poisoning.

7. **Answer:** 1) Use formats with stricter validation mechanisms like ONNX or JSON.

Explanation: Using secure serialization formats and adding validation checks reduces risks of malicious payloads.

8. **Answer:** 1) Unauthenticated management API access.

   Explanation: ShellTorch vulnerabilities included unauthenticated API access, among others, which allowed remote exploitation.

9. **Answer:** 2) Avoiding root access issues.

   Explanation: Singularity avoids root access, making it safer for use in shared environments compared to Docker.

10. **Answer:** 2) To ensure sensitive data is processed in secure and controlled workspaces.

    Explanation: TREs are designed to secure sensitive data during processing and comply with privacy regulations.

7. Testing and Validating AI Systems

Learning outcomes

After completing this chapter, you will:

- Understand key security threats in the AI lifecycle and their impact on systems.
- Learn testing methods like red teaming, black box, and white box testing.
- Explore principles of AI alignment to ensure safe and ethical outputs.

We continue our exploration of the AI lifecycle and focus on the “Verification and Validation” stage. In the MLOps workflow we are basically moving from the “experimentation” to the “production” stage. The verification and validation stage ensures that the AI system from the design and development stage works as expected using the test data that was left out from development. This is also the stage where it’s time to put the system under stress and test if for example personal data can be extracted

In chapter 3 we briefly introduced the types of attacks that AI systems can suffer. Here we go deeper on the explanation of each type of attack and more broadly how an AI system can malfunction. The second part of this chapter focuses on testing strategies, to ensure that the AI model that we have developed is not prone to training (personal) data leaks or wrong predictions. Finally we conclude on the topic of alignment. It is important to notice that while our focus is on protecting personal data, the security considerations on this chapter apply to any AI system trained on any sort of sensitive or proprietary data that needs to be protected.

7.1 Security threats of AI systems

When considering the security of an AI system, it is important to introduce the three types of security threats (see below), when and where they happen in the AI lifecycle. A reference for this section is an important resource that the reader is encouraged to explore often is the OWASP AI Exchange Community (2025).

There are three types of of AI security threats:

1) Development-Time Threats: threats occurring during the development phase of AI systems (data collection and preparation, model training)
2) Threats Through Use: threats occurring when the AI model is in operation and interacting with users who provide inputs and receive outputs. Attackers exploit the AI system's interfaces to deceive or extract sensitive information.
3) Runtime Application Security Threats: threats that target the AI system while it is deployed and running in a production environment. Attackers aim to manipulate, steal, or disrupt the AI model by exploiting vulnerabilities in the operational setup.

The following table summarises the types of threads that we should be aware when developing and deploying AI systems

Type of Threat	**Threat**	**Description**	**Impact**	**Example**
Development -Time Threats	**Data Poisoning**	Inserting malicious data into the training dataset.	Causes the AI model to behave undesirably or make incorrect predictions.	Corrupting a facial recognition dataset to misidentify individuals.
	Model Poisoning	Altering model parameters or architecture during development.	Embeds vulnerabilities or backdoors into the AI model.	Modifying the model to allow future unauthorized access.
	Supply Chain Attacks	Compromising third-party tools, libraries, or models used in development.	Introduces vulnerabilities or malicious code into the AI system.	Injecting malicious code into a popular open-source library used for training.
Threats Through Use	**Evasion Attacks**	Crafting inputs designed to deceive the AI model.	Causes misclassification or incorrect outputs without modifying the model.	Altering a stop sign image to fool an autonomous vehicle's recognition system.
	Model Inversion	Inferring sensitive training data from the model's outputs.	Breaches privacy by revealing personal or confidential data.	Reconstructing data about individuals used in training.
	Membership Inference	Determining if specific data was part of the training set.	Reveals sensitive information about individuals included in the dataset.	Identifying whether a person's data was used in training a criminal risk prediction model.

Type of Threat	Threat	Description	Impact	Example
	Prompt Injection	Manipulating input prompts to cause the AI to generate harmful or unintended outputs.	Circumvents filters, causing the model to produce malicious or inappropriate content.	Convincing a language model to disclose sensitive information or generate offensive text.
Runtime Application Threats	**Runtime Model Poisoning**	Gaining unauthorized access to alter the model during operation.	Causes the model to perform unintended actions or facilitate further attacks.	Modifying a deployed fraud detection model to overlook fraudulent transactions.
	Model Theft (Extraction)	Reconstructing the model’s functionality through repeated queries.	Intellectual property theft and loss of competitive advantage.	Systematically querying an AI service to rebuild the model.
	Denial of Service (DoS)	Overwhelming the AI system with excessive requests or computational demands.	Degrades service quality or renders the AI system unavailable.	Flooding an AI-powered API with resource-intensive queries to crash the service.
	Insecure Output Handling	Outputs contain malicious code or sensitive information.	Enables cross-site scripting (XSS) attacks or leaks confidential data to users.	An AI chatbot generating responses with embedded malicious scripts.

7.2 A taxonomy of attacks on AI systems

This section provides a brief overview of key attacks targeting machine learning (ML) systems, categorized by their objectives and attack phases from Vallet (2022), expanding on the OWASP AI Exchange categorisation from the previous section, with a focus on types of attacks:

1) **Manipulative attacks** deceive systems during production
2) **Infection attacks** corrupt systems during training
3) **exfiltration attacks** steal sensitive information

7.2.1 Manipulative Attacks

Manipulative attacks aim to deceive AI systems during their production phase by providing malicious or unexpected inputs, causing the system to behave incorrectly.

- **Evasion Attacks**: Evasion attacks involve crafting inputs specifically designed to fool the model. For example, adversarial examples are slightly altered inputs (e.g., images, text, or sound) that appear normal to humans but trick the model into making incorrect predictions. Goodfellow et al. (2014) demonstrated how adding imperceptible noise to an image could make a classifier misidentify it entirely. These attacks exploit weaknesses in how models generalize from training data and are particularly dangerous because they do not require altering the model itself.

  - *Example*: Adversarial patches placed in an image can make a model misclassify objects, like labeling a “stop” sign as a “speed limit” sign, as shown by Eykholt et al. (2018).

- **Adversarial Reprogramming**: Elsayed et al. (2018) introduced a way to hijack an ML model to perform unintended tasks. For instance, a model designed for image classification could be covertly modified to solve unrelated problems, like identifying patterns useful for cryptocurrency mining. This is achieved by retraining the model with data that embeds the new task.

- **Denial-of-Service (DoS) Attacks**: These attacks target the availability of the AI system, often by overwhelming its computational resources. Engstrom et al. (2019) explored scenarios where malformed inputs or computationally intensive requests degrade a system’s performance, making it unable to respond to legitimate queries.

7.2.2 Infection Attacks

Infection attacks compromise the training phase of an ML model, introducing malicious changes that affect its behavior during production.

- **Poisoning Attacks**: These attacks aim to corrupt the training data, thereby sabotaging the model’s learning process. Nelson et al. (2008) highlighted how injecting misleading or malicious data into the training set could shift the model’s decision boundaries, reducing its accuracy or introducing systematic errors. For example, poisoning a spam filter by adding cleverly designed spam messages could make it misclassify similar spam as legitimate emails.

  - *Impact*: Poisoning attacks are particularly dangerous for systems trained on public or external data, where attackers can inject tainted samples unnoticed.

- **Backdooring Attacks**: In these attacks, the attacker embeds a secret trigger into the model during training, such as a specific pattern or image. When this trigger is present in the input, the model performs a specific action defined by the attacker, regardless of its normal behavior. Liu et al. (2017) demonstrated how a backdoor in a facial recognition system could misidentify individuals wearing a specific pair of adversarial glasses.

7.2.3 Exfiltration Attacks

Exfiltration attacks focus on extracting sensitive information from AI systems, including training data or the model itself. These attacks pose significant risks to privacy and intellectual property.

- **Membership Inference Attacks**: These attacks determine whether a specific data point was part of the training set. Shokri et al. (2017) showed how an attacker could infer sensitive details, such as whether an individual’s medical data was used to train a model, by analyzing how confidently the model responds to specific inputs. These attacks exploit the tendency of ML models to “memorize” training data.

  - *Example*: An attacker could determine whether a person was part of a study about Alzheimer’s disease based on the model’s output for that individual’s data.

- **Model Inversion Attacks**: Inversion attacks attempt to reconstruct sensitive data from the model’s outputs. For instance, Fredrikson et al. (2015) demonstrated how to recover a person’s facial image from a facial recognition model by probing the model with various inputs. This type of attack can reveal personal data, such as medical or biometric information.

- **Model Extraction Attacks**: These attacks replicate a model by querying it repeatedly and reconstructing its behavior. Tramer et al. (2016) illustrated how attackers could approximate a proprietary model by observing its responses to various inputs. This not only compromises intellectual property but also risks exposing sensitive patterns or biases encoded in the model.

7.3 Testing and Validation of AI Systems

To prevent such threats to AI systems, there are different practices that can be adopted. In this section we will cover **Red Teaming**, **white box testing** and **black box testing**.

7.3.1 Red teaming practices

Red teaming (https://www.ibm.com/think/topics/red-teaming) is a proactive approach to testing and improving the security of systems, including AI models and systems. It involves simulating real-world adversarial behaviors to identify vulnerabilities and evaluate how well a system can withstand attacks. Unlike traditional security audits or penetration testing, red teaming focuses on mimicking advanced threat actors' techniques, tactics, and procedures (TTPs). This method helps organizations understand their security posture and anticipate potential attacks. Red teaming is especially valuable for AI systems, as it exposes weaknesses in models, data pipelines, and deployment environments that traditional methods may overlook.

7.3.1.1 Red, Blue, and Purple Teaming in Cybersecurity

To comprehensively assess and improve the security of systems, organizations adopt three types of teams—red, blue, and purple—each playing a specific role as shown in the table below. By combining red, blue, and purple teaming, organizations can holistically address the challenges of securing AI systems. Red teaming provides an overview to potential attacker strategies, blue teaming strengthen defenses, and purple teaming ensures collaboration for improving security.

Team	Role	Goals	Relevance to AI
Red Teams	Offensive security professionals who simulate real-world attacks on an organization's systems.	Identify and exploit vulnerabilities, bypass defenses, and avoid detection.	For AI systems, red teams may craft adversarial examples, attempt model poisoning, or exploit exfiltration techniques like membership inference or model inversion, highlighting gaps in defenses that could be exploited by attackers.
Blue Teams	Defensive IT security professionals responsible for protecting the system and data from threats.	Monitor for intrusions, respond to alerts, and continuously strengthen security measures.	Blue teams safeguard AI pipelines by implementing secure practices such as access controls, input validation, and anomaly detection to mitigate risks exposed by red team activities.
Purple Teams	A cooperative collaboration process	Facilitate knowledge sharing,	Purple teams integrate insights from red and blue teams to recommend mitigations, improves system

Team	Role	Goals	Relevance to AI
	between red and blue teams.	improve communication, and ensure continuous improvement in organizational security.	defenses, and validate that improvements address identified weaknesses without introducing new risks.

7.3.1.2 Red Teaming in Practice

Red teaming begins with a clear objective, often defined in collaboration with other parties. Ethical hackers, known as red team members, mimic real-world attackers' tactics, techniques, and procedures (TTPs). The process is non-destructive and strictly follows a code of conduct to ensure no harm is done to the organization's systems or data.

Red teams use a variety of tools and methods to simulate attacks:

- **Social Engineering**: Techniques like phishing and vishing to trick users into revealing sensitive information.
- **Network Sniffing**: Monitoring traffic to gather configuration details and credentials.
- **Application Penetration Testing**: Identifying coding flaws, such as SQL injection vulnerabilities.
- **Tainting Shared Content**: Embedding malware in shared drives to test lateral movement.
- **Brute Force Attacks**: Guessing passwords using lists of commonly used credentials or breached datasets.

For AI systems, red teaming involves:

- Crafting adversarial inputs to test robustness.
- Simulating data poisoning attacks to evaluate the security of training pipelines.
- Testing the resilience of deployed models against extraction or inference attacks.

7.3.1.3 Advances in red teaming practices

Recent advancements in red teaming, as outlined by OpenAI https://openai.com/index/advancing-red-teaming-with-people-and-ai/, focus on combining **human expertise** and **automated tools** to identify vulnerabilities in AI systems. OpenAI's approach introduces structured campaigns that involve external red teamers with diverse expertise and automated methods powered by reinforcement learning (RL) to probe AI systems at scale.

For manual red teaming, OpenAI emphasizes creating a diverse team of experts, ranging from cybersecurity specialists to domain-specific researchers, to assess models across varied use cases. This approach ensures that red teaming campaigns are tailored to specific AI models, with clear goals and structured testing processes. Automated red teaming complements this by using AI to generate diverse and effective attack scenarios, with techniques like multi-step RL and reward mechanisms that prioritize both success and novelty of attacks. These methods allow for testing a wide range of vulnerabilities, such as prompt manipulation and misuse of capabilities, in a systematic and scalable manner.

The novel combination of manual and automated approaches provides deeper insights into potential risks, allowing for more robust safety evaluations and better training of AI models to handle real-world threats. While red teaming isn't a comprehensive solution to all risks, these innovations mark a significant step toward making AI systems safer and more reliable.

7.3.2 Other testing approaches: white box and black box testing

In machine learning, **white box testing** and **black box testing** are two basic methods of model reliability, functionality, and security testing. Each provides insight into different aspects of model behavior and complements the other in a robust testing strategy.

7.3.2.1 Black Box Testing: External Behavior Analysis

Black box testing is a testing of the model based on the input-output behavior of the model, without considering its internal structure. The testers feed different inputs into the model and compare its output against expected results to detect errors in functionality, usability, or performance.

- **Advantages**: Black box testing is particularly good for finding issues such as bias, fairness, or unexpected outputs in real-world scenarios. It requires no knowledge of the model's internal algorithms, making it suitable for testing pre-trained or proprietary models.
- **Limitations**: Since it doesn't go inside the model's internal mechanism, black box testing cannot pinpoint the root cause of identified issues, often requiring additional debugging or analysis.

7.3.2.2 White Box Testing: Internal Examination

White box testing goes into further detail regarding the model's architecture, code, and algorithms. This form of testing consists of test evaluators using internal processes, inspecting code for vulnerabilities, and pinpointing inefficiencies or bottlenecks.

- **Advantages**: This approach helps in finding bugs in coding, model performance optimization, and ensuring security against adversarial manipulation. For instance, the testers may investigate how data flows through the model to find potential leakages or vulnerabilities in handling sensitive input.
- **Limitations**: White box testing is resource-intensive and requires significant technical expertise. It may overlook issues related to external factors like data distribution or user interaction.

7.3.2.3 ***Combining Approaches***

A robust testing strategy often incorporates both methods:

- **Black box testing** evaluates the model's performance and functionality as perceived by end-users.
- **White box testing** ensures internal correctness, security, and optimization.

By using these complementary techniques, organizations can thoroughly test machine learning models, ensuring they are both effective and secure in real-world applications.

Box: The case of LLMs and the top security threats in Generative AI

OWASP has released two important guidelines, *OWASP Top 10 for LLM Applications* (2023 and 2025). LLMs and Generative AI with LLMs have faced huge success and exposure, but the rapid evolution of these technologies has amplified the threats that these systems have been exposed to. The 2025 guidelines build on the 2023 one, addressing new vulnerabilities from the increased integration of LLMs into multimodal systems, such Retrieval-Augmented Generation (RAG, when a further source of data is attached to the *prompt* that is passed to the LLM), and agentic architectures (basically programs based on LLMs that are able to spawn more instances to complete tasks indepdently of human intervention).

The 2025 version highlights the emergence of **System Prompt Leakage**, a risk stemming from assumptions about prompt isolation, and expands on **Excessive Agency**, reflecting growing concerns about granting LLMs autonomy. Additionally, the updated list introduces **Vector and Embedding Weaknesses**, addressing risks in embedding-based methods critical for grounding model outputs. The redefined **Unbounded Consumption** broadens the earlier focus on Denial of Service to include cost management in large-scale deployments. These updates reflect the evolving complexity of securing LLMs, urging a community-driven approach to adapt defenses to new vulnerabilities.

While LLMs do not necessarily process personal data and memorize the training sets used, it is often possible to extract personal data with specific prompts, whether the user is acting with or without malicious intent. The European Data Protection Board (European Data Protection Board (EDPB) 2024) further emphasizes that models trained on personal data cannot be assumed to be anonymous, given the possibility of re-identification or indirect extraction of data from model outputs. The EDPB opinion further stresses the importance of conducting robust balance tests that considers the risks of data processing against the rights and freedoms of individuals, ensuring that any data use is proportionate and justified within the GDPR.

7.4 AI model alignment

Considering a more holistic view of the threats of AI systems, some mitigation strategies could also happen within the model itself by identifying malicous inputs to avoid generating unwanted (or unlawful) outputs.

AI alignment (Ji et al. 2023) tries to ensure that AI systems function in accordance with human intentions and values. There are four central principles of alignment: **robustness**, **interpretability**, **controllability**, and **ethicality** (RICE). These principles guide the forward alignment processes (training models to align with specified objectives) and backward alignment (assessing and governing systems post-training and deployment). Forward alignment emphasizes techniques like reinforcement learning from human feedback (RLHF) and adversarial training, while backward alignment involves assurance mechanisms such as interpretability tools and governance frameworks to monitor risks throughout the lifecycle of AI systems.

Hint for instructors

Depending how the instructor is developing the course, and on the level of the participants, this module could be expanded with more practical examples. A good homework to consider is to look at how alignment has failed in the past, or how red-teaming has been described in details in papers such as

7.5 Summary

This chapter focused on testing and validating AI systems, addressing key threats, testing methods, and model alignment to ensure that outputs are not going to cause unwanted data breaches or other sorts of security issues. At this stage the AI model is trained, the AI system is tested, and we can move to **Secure Deployment Practices**.

Exercise 7.1: Multiple choice questions

Question	Options
1. What is the main purpose of the "Verification and Validation" stage in the AI lifecycle?	1) To finalize model deployment.2) To test the system against predefined requirements and stress it under potential threats.3) To design the AI architecture.4) To deploy training pipelines.
2. Which of the following is a development-time threat to AI systems?	1) Evasion attacks.2) Data poisoning.3) Model theft.4) Denial of service (DoS).
3. What is an example of a manipulative attack on AI systems?	1) Data poisoning.2) Adversarial reprogramming.3) Membership inference.4) Supply chain attacks.
4. Which type of attack focuses on extracting sensitive data from AI systems?	1) Model inversion attacks.2) Evasion attacks.3) Adversarial reprogramming.4) Denial of service (DoS).
5. What is the role of red teaming in AI system security?	1) To deploy AI systems securely.2) To simulate adversarial behaviors and identify vulnerabilities.3) To create new training data.4) To monitor runtime performance.
6. What is a key limitation of black box testing?	1) Requires deep technical knowledge of the model.2) Cannot pinpoint the root cause of identified issues.3) Focuses only on internal processes.4) Is resource-intensive and time-consuming.
7. How does white box testing differ from black box testing?	1) It evaluates external behavior only.2) It examines internal model processes and algorithms.3) It uses no prior knowledge of the system.4) It targets real-world user inputs.
8. What does "AI alignment" aim to achieve?	1) Reduce training time.2) Ensure the system functions according to human intentions and values.3) Improve model performance.4) Simplify deployment pipelines.
9. Which is a potential risk of model inversion attacks?	1) System downtime.2) Revealing sensitive training data.3) Misclassifying inputs.4) Generating offensive outputs.
10. What is the primary focus of the OWASP guidelines for LLM applications?	1) Improving model explainability.2) Addressing vulnerabilities specific to LLMs.3) Optimizing system performance.4) Improving training data collection.

Exercise 7.1. Solutions

Click to reveal solutions

1. **Answer:** 2) To test the system against predefined requirements and stress it under potential threats.

Explanation: This stage ensures the AI system functions as expected and identifies vulnerabilities.

2. **Answer:** 2) Data poisoning.

   Explanation: Data poisoning manipulates training data to compromise the model’s functionality.

3. **Answer:** 2) Adversarial reprogramming.

   Explanation: This attack hijacks a model to perform unintended tasks during production.

4. **Answer:** 1) Model inversion attacks.

   Explanation: These attacks reconstruct sensitive data from model outputs.

5. **Answer:** 2) To simulate adversarial behaviors and identify vulnerabilities.

   Explanation: Red teaming proactively tests system robustness against potential threats.

6. **Answer:** 2) Cannot pinpoint the root cause of identified issues.

   Explanation: Black box testing focuses on input-output behavior, not internal mechanisms.

7. **Answer:** 2) It examines internal model processes and algorithms.

   Explanation: White box testing involves a detailed internal review of the model’s structure.

8. **Answer:** 2) Ensure the system functions according to human intentions and values.

   Explanation: AI alignment focuses on ensuring ethical and robust system behavior.

9. **Answer:** 2) Revealing sensitive training data.

   Explanation: Model inversion can extract personal or confidential data from model outputs.

10. **Answer:** 2) Addressing vulnerabilities specific to LLMs.

    Explanation: OWASP guidelines target security risks unique to large language models.

8. Deployment and Monitoring of AI systems

Learning outcomes

After completing this chapter, you will:

- Learn best practices for securely deploying AI systems
- Understand the security risks associated with production environments, and how to mitigate these risks
- Understand the importance of monitoring AI systems for performance degradation, drift, and bias to ensure models performance and alignment are still valid
- Learn about various techniques for detecting drift, including data drift and concept drift, and how to implement automated alerts for corrective action.
- Explore the role of incident response plans and ethical monitoring in maintaining transparency and fairness throughout the AI system's lifecycle.

After all the efforts of securely developing and testing the AI system, it is now to turn it into an actual product that can be used by users. Deployment can mean different things according to where/when the model will actually be used. It could be through a web interface (a chatbot, an API to query the AI system), it could be embedded on a device, or it could also be integrated into an existing software pipeline where it runs periodically or in real-time as part of a larger system. Regardless of the setup, going "live" introduces new security and privacy challenges: the model is now exposed to users, environments, and data flows that most likley were not part of the development or testing stages: expect the impossible! Because of this, deployment must include robust monitoring, access control, and other mechanisms to detect misuse, data drift, or potential leaks of the training or input data.

8.1 Transparency

While one might think that **transparency** is the last thing to consider when bringing your AI system to users, it is actually a fundamental requirement to build users' trust and make sure the product is in line with both GDPR and the AI Act regulations. Without going into the details of (personal) data or AI governance, here we focus on practical transparency best practices that should accompany the deployment and public release of your AI system or model:

- **Clear privacy notice** explaining in simple language what personal data is collected, how it's used in model training or processing, and how it's protected and ensure there is a valid legal basis (European Data Protection Board (EDPB) 2024)
- **AI usage disclosures**: Inform users when they are interacting with an AI system or when AI is used in a service. You can also **explain model scope and limitations** and be honest about what the AI application can and cannot do.
- **Data handling transparency and other needs for control or consent** it is important to mention for example if the users' data is being logged or reused for new purposes even in formats that might not include personal data
- **Documentation and explainability**: finally documentation of the new AI system/model is fundamental to ensure transparency and consider also **explainability** on how the AI makes decisions or why it produced a certain output. An excellent tool for ensuring transparency of an AI model (system) is the **model card** (or **system card**) as described in Mitchell et al. (2019).

Model cards

Model cards are a documentation tool for transparency that provide a comprehensive snapshot of a model's characteristics and ethical considerations. A model card should accompany any AI model trained on personal data (or any important model) to detail its intended use, performance, and the data it was trained on (check "5 things to know about AI model cards" by Desai (2023))

Depending on the type of model you are deploying, it is important to explore existing model/system cards for example by looking at model cards provided by OpenAI, NVIDIA, or Google. Model cards were actually introduced by Google in Mitchell et al. Mitchell et al. (2019) so a great starting point is the table provided in the paper, which is annotated here below:

An annotated ***model card*** *template based on Mitchell et al. (2019)*

Section	Subsection	Description / What to Include
Model Details	Person or organization	Name of developer(s) or organization responsible for the model
	Model date	Date the model was developed or released
	Model version	Version identifier for the model
	Model type	E.g., classification, regression, transformer, etc.
	Training details	Description of training algorithms, hyperparameters, fairness constraints, regularizations, and feature types
	Reference resources	Link to paper, website, or documentation with further info
	Citation details	Citation for academic referencing
	License	Terms under which the model can be used (e.g., MIT, CC BY)
	Contact info	Email or web form to reach model developers
Intended Use	Primary intended uses	Real-world scenarios where the model is designed to be used
	Primary intended users	Types of users expected (e.g., clinicians, developers, students)
	Out-of-scope use cases	Known limitations or areas where the model should not be applied
Factors	Relevant factors	Characteristics that affect model performance (e.g., age, gender, device type)
	Evaluation factors	Factors used when evaluating the model (e.g., subgroup performance, lighting conditions)
Metrics	Model performance measures	Metrics used (e.g., accuracy, F1-score, AUROC)
	Decision thresholds	Thresholds used to turn scores into decisions (e.g., probability > 0.5)

	Variation approaches	Methods for robustness checks (e.g., stratified evaluation)
Evaluation Data	Datasets	Names and descriptions of datasets used for evaluation
	Motivation	Why those datasets were selected for evaluation
	Preprocessing	Steps taken to prepare the data (e.g., normalization, filtering)
Training Data	—	If available: source, composition, collection process, and any known limitations of the training data
Quantitative Analyses	Unitary results	Performance across individual factors (e.g., age groups)
	Intersectional results	Performance across combinations of factors (e.g., young + female)
Ethical Considerations	—	Discussion of fairness, biases, potential harms, and mitigation strategies
Caveats and Recommendations	—	Known weaknesses, areas needing caution, and suggestions for users

8.2 Basics of AI system deployment

When looking at Huyen (2022), the author stresses how important it is to avoid some of the typical assumptions related to AI system/model deployment. Here some (wrong) assumptions that are worth considering:

- **You only deploy one or two ML models at a time**. In reality, companies have many ML models, and an application may require multiple models for different features. For example, a ride-sharing app needs models to predict ride demand or drivers' availability. Additionally, if the application operates in multiple countries or cities, each country/city may need its own set of models. Some companies have hundreds or even thousands of models in production!

- **If we don't do anything, model performance remains the same.** ML systems can degrade over time due to software rot and data distribution shifts, especially when the data encountered in production differs from the training data. Your AI models/systems tend to perform best right after training/deployment and will inevitably degrade over time.

- **You won't need to update your models as much.** Since model performance decays over time, models should be updated as frequently as possible. Some companies update their models multiple times a day. The right question to ask is "how often *can* I update my models" not "how often *should* I update my models".

- **Most ML engineers don't need to worry about scale.** Many companies, even those with 100+ employees, need to be able to scale their ML applications. Scaling will always be a concern for the whole team when the product is successful, and this means designing and deploying an AI system that can serve many queries per second or millions of users per month (fun side quest: search for "Our GPUs are Melting", quote by OpenAI CEO in March 2025)

Now that the myths are cleared, it is also important to consider what type of AI system is being deployed and Huyen (2022) introduces two types of systems:

- **Batch prediction** generates predictions periodically or when triggered, stores them, and retrieves them as needed
- **Online prediction** generates and returns predictions as soon as requests arrive, and is also known as on-demand prediction.

While basic principles are important, they are not the core focus of this book. Similarly other issues related to scalability, highly depends on what system the AI system is installed on, and more generally they are issues that any application might face, even without any AI involved. If we go back to our focus on data protection and security, we next consider a checklist of various aspects to consider before and during deployment (for further readings see Ahmad et al. (2024)).

8.3 A secure deployment checklist

So what are the steps to securely deploy your AI model/system?

1) **Secure deployment infrastructure** : similar as mentioned in previous chapters, it is important that the code is able to run in the production computer clusters. If during development it is possible to work on firewalled clusters or *trusted environments*, when the system goes in production all new sources of threats are now available due to the expanded *attack surface* of the deployed system.

2) **Access Control**: when dealing with AI models trained with personal data or AI systems processing personal data, you might need to consider implementing access control. This is to make sure that only certain users can access the sensitive AI system. When user inputs also contain personal data, access control is also needed to store other data that the user might be providing to query the AI system.

3) **Model integrity, version control, continuous integration/continuous delivery (CI/CD)**: version control (of code, of data, and of models) and CI/CD practices are not only necessary to ensure the reproducibility and transparency of the development work, it becomes even more important during deployment. With cryptographic hashing it is possible to verify the integrity of deployed models and with strict version control with model registries it is always possible to control for model updates and set-up rollback mechanisms in case of compromised updates. Finally with CI/CD we can set up automated pipelines to build, test, and deploy models reliably includes automating 1) **unit tests** for model code, 2) **integration tests** for data pipelines, and 3) **validation tests** on model performance

Containerization and infrastructure-as-code

While not specific to AI systems processing personal data, you will most likely going to deploy your model using containers over computing nodes that are automatically managed with the so called *infrasctructure-as-code*.

Infrastructure as Code (IaC) is the practice of managing IT infrastructure using machine-readable/actionable configuration files, allowing for automation, consistency, and version control across environments. Instead of manually configuring servers and networks, developers define infrastructure through code, which can be tested and reused like software. Common IaC tools include **Terraform**, **Ansible**, **Pulumi**, **AWS CloudFormation**, and **Chef**. For more details, see “Infrastructure as Code” Morris (2025).

What are the risks related to containers and infrastructures? A short list of what is important to consider when working with containers and infrastructures managed for example with Kubernetes. Each topic could easily have a course of its own, the goal here is to present the reader with examples and terms and let them explore further if/when they are going to need these tools for their work.

- **Isolation and sandboxing**
    - Although containers offer isolated environments, they are not completely immune to privilege escalation. Use namespaces and cgroups to enforce strict isolation.
    - Avoid running containers with root privileges; ensure containers operate with the least amount of privilege necessary.
- **Container image security**
    - Regularly update and patch container images to fix known vulnerabilities. Tests and updates should always be managed via version control and CI/CD.
    - Use only trusted and verified container images (e.g., from DockerHub or even better from private registries).
    - Scan images for vulnerabilities using tools like Clair, Trivy, or Aqua Security, and make these part of your CI/CD pipeline.
- **Secrets management**
    - Avoid hard-coding secrets (API keys, credentials) in container images. Instead, use secure vaults or environment variables for secret management.
- **Network Security**
    - Implement strict network policies to control container-to-container and container-to-host communication.
    - Segment containers based on sensitivity (e.g., separating user-facing services from internal model services).
- **Control Plane Security**
    - In Kubernetes it is important to harden the Kubernetes control plane by limiting access and enabling audit logs.
    - Use role-based access control (RBAC) to define clear roles for users and services interacting with the Kubernetes API.
- **Pod Security**
    - Implement Pod Security Policies to enforce restrictions on what containers can do (e.g., restricting privilege escalation, disallowing the use of host resources).
    - Ensure that all Kubernetes pods run with non-root users.
- **Network Policies**
    - Enforce Kubernetes Network Policies to control the flow of traffic between pods and services.
    - Secure intra-cluster communication by encrypting data with mTLS (mutual Transport Layer Security).
- **Supply Chain Risks for Kubernetes Plugins**
    - Be cautious with third-party plugins (e.g., CNI plugins, ingress controllers) as they could introduce security vulnerabilities. Verify and maintain plugins regularly.

8.4 Monitoring AI Systems

Deployment is of course tightly interwined with **monitoring**. While monitoring of infrastrctures and systems in general is fundamental even without any AI, monitoring AI systems adds a new lawyer of performance monitoring that becomes crucial for ensuring that the AI system remains aligned with the intended scopes and ensure ethical and regulatory standards during its use. Over time, models can experience performance degradation, or "drift" due to changes in input data patterns or other external conditions. Similarly, biases may emerge post-deployment that were not evident during training. Ongoing monitoring helps identify these issues early and take corrective action.

In this section we will not consider the monitoring of infrastructure (network, access control, application usage) instead we will focus on AI specific issues related to monitoring of AI systems that have been deployed.

8.4.1 AI model drift detection

AI models can **drift** when some statistical properties of new input data change over time. With **concept drift** model accuracy might be degrading or we might start noticing unsafe predictions and unwanted behaviour from the AI system. Possible thing to consider include:

- **Data distribution tracking:** Continuously track input data statistics (feature distributions, correlations, etc.) and compare them to the training baseline. Significant shifts (e.g. via Kolmogorov-Smirnov tests or population stability index) can signal data drift before performance issues emerge.
- **Model performance monitoring:** With classifiers or other tools which provide outcomes and labels, accuracy or error rates can be evaluated when the ground truth of predicted label is known. Sudden drops can indicate concept drift affecting the model's predictive relationship. This can be of crucial importance if AI systems that are helping in decision making in healthcare or finance. For example in a recent study Kore et al. (2024), tracking only aggregate accuracy failed to detect a significant COVID-19-induced data shift, whereas a dedicated drift detector caught it.
- **Drift detection algorithms:** Unsupervised drift detectors algorithms (e.g. ADWIN, DDM) can raise alerts when model outputs or data distributions change statistically, even before ground truth is known. These methods can monitor prediction probability distributions or feature importances for unexpected changes.
- **Responsive adaptation:** If possible, it is good to establish thresholds, so that model degradation can be automatically evaluated and then it can be decided if a model needs to be re-trained or rolled back to a previous version.

How to detect model drift?

By comparing data distributions over time using statistical or machine learning methods to identify significant changes. There are various techniques and tools for detecting model drifts, and domain specific knowledge might help you decide on what is the best tool for your case. Here we follow Hinder, Vaquet, and Hammer (2024) and the four stages that they recommend:

1. **Select data windows (Stage 1: Acquisition)**
   Use one or two time-based windows to collect data for comparison. Common strategies include sliding, fixed, growing, or model-based reference windows.

2. **Describe the data (Stage 2: Descriptor Building)**
   Convert raw data into structured representations using descriptors like

histograms, decision trees, kernel matrices, or machine learning embeddings to capture the distribution.

3. **Measure change (Stage 3: Dissimilarity Computation)**
   Apply a distance or divergence measure (e.g. Total Variation, Hellinger, MMD, KL divergence) to quantify the difference between distributions in the selected windows.

4. **Normalize and evaluate (Stage 4: Normalization)**
   Normalize dissimilarity scores to correct for estimation variance using statistical techniques (e.g. permutation testing or p-values) to assess if a detected drift is statistically significant.

8.4.2 Bias monitoring

If model drift detection is somewhat more focused on the performance of the AI model/system, **bias monitoring** adds an ethical layer to drift detection to avoid that AI system outcomes negatively impact individuals of certain demographics. After deployment, it is important to continuously assess model outcomes across diverse demographic or specific sub-groups. For example in healthcare, AI system showed worse accuracy for under-represented patient groups compared to others, indicating a novel bias that was not identified during development Gichoya et al. (2023).

So what should be considered to perform bias monitoring?

- **Slice performance analysis:** to track metrics separately for different user groups (e.g. gender, ethnicity, age segments) or other protected attributes.
- **Fairness metrics:** beyond accuracy, we can use fairness indicators (demographic parity, equal opportunity difference, etc.) in production monitoring. For example with a classifier, we could compare positive prediction rates or false-negative rates across different groups to see if there are large imbalances between groups.
- **Bias detection tools:** There are tools such as IBM AI Fairness 360 or Microsoft Fairlearn to automate checks and highlight significant deviations.

By setting up bias triggers, we can then evaluate if retraining is necessary to ensure that the model does not perpetuate or amplify biases as data evolves.

8.4.3 Privacy risk monitoring

When AI models are trained on or process personal data, **privacy risks** must be actively monitored. Models/systems could inadvertently **leak confidential information** about individuals, through model outputs or through other attacks (e.g. membership inference attacks). Success in such an attacks can constitute a privacy breach, since it reveals information about who was in the model's training data.

To monitor and mitigate privacy risks, one can consider the following:

- **Output scanning / output filtering:** Continuously scan model outputs (predictions, generated text, etc.) for any sensitive personal data. For example, for a generative model like a large language model, can implement a regular expression or named-entity detection as output filter to catch if personal data are being regurgitated and eventually block the response before displaying it to the final user. Output filters become a necessary tool, since large models *can* memorize and output verbatim personal details seen during training (Carlini et al. 2021)).

- **Monitoring model confidence:** One potential sign of a privacy leakage is overly confident predictions on certain inputs which could imply that certain data was in the training set.
- **Differential privacy logging:** If a model was trained with DP, depending on the DP technique adopted, it can be possible to monitor the **privacy loss metric (ε)** over time and ensure it stays within acceptable bounds.
- **Privacy audits & penetration testing:** While not strictly related to live monitoring, performing regular **privacy audits** on the model can prevent unwanted privacy risks. This can involve simulating attacks like membership inference or model inversion in a controlled setting to see if the model is vulnerable. F

8.4.4 Other types of monitoring

Other types of monitoring could be considered especially when an AI system is categorised as high-risk. For example one can consider ethical monitoring by monitoring possible misalignment of the AI system during deployment, especially when comparing it with results from development stage and with what is known from the model card. Accountability and explainability monitoring can also be important with AI system and adopting redress mechanism to track complaints or appeals related to AI decisions and addressing them. Establish clear processes for humans to override or correct AI decisions in fundamental not only as a legal requirement, but as a strong ethical principle in AI systems which can affect individuals.

Together, ethical and explainability monitoring help maintain the social license to operate by ensuring transparency, fairness, and accontability. These practices not only support compliance with regulations like the **AI Act**, but also promote responsible and trustworthy AI over time.

8.5 Incident Response and Recovery Plans

Finally, monitoring should also linked to incident response and recovery plans. These plans outline how to react when an AI system produces harmful, incorrect, or insecure outcomes after deployment. Traditional incident response (as used in cybersecurity) must be now re-adapted to consider AI-specific failures as outlined above.

So what should be done?

- **For an AI incident response team:** a cross-organisational team involving data scientsits, engineers, legal and communication exerts can quickly react to unwanted incidents.
- **Prepare for common scenarios:** Develop runbooks for likely incidents (e.g., model performance collapse or data pipeline failure or any other detected attacks.
- **Integration with CSIRT Processes:** Incorporate AI incidents into the organization’s existing Computer Security Incident Response Team framework (CSIRT). The response process should cover identification, containment, eradication, recovery, and lessons learned, tailored to AI. For example, if a model is compromised or producing biased content, the containment might involve taking the model offline or enabling a safe backup model (see VanHoudnos et al. (2024))
- **Prevention:** After an incident is solved, it is good practice to perform a *post-mortem* analysis focusing on the AI aspects. Was it a data quality issues? Concept drift? Adversarial inputs? How can monitoring be improved?

Hint for instructors

This section introduced many new concepts, which can be difficult for learners to fully grasp without practical, hands-on activities. Ideally, learners should have the opportunity to experiment with deployment (including containerization and infrastructure as code) and monitoring, using AWS, Azure, or other similar services that support small-scale testing at little or no cost.

8.6 Summary

This chapter focused on the practical aspects of releasing and deploying an AI system/model that processes (or was trained on) personal data. In the next chapter we will outline on how to react when the model performance is decreasing and what to consider for secure decommissioning of AI systems/models with personal data.

Exercise 8.1: Multiple choice questions

Question	Options
1. What is a key reason why monitoring is necessary after AI deployment?	1) To reduce training costs2) To automatically generate new datasets3) To detect drift and ensure continued alignment with intended use4) To convert the model into a system card
2. What is the purpose of a model card in AI deployment?	1) To track container resource usage2) To describe the model’s ethical implications and usage limitations3) To secure the container registry4) To ensure legal protection of the developer
3. Which of the following is a valid technique for detecting data drift?	1) OAuth authorization2) Feature distribution comparison3) SSH key rotation4) Container privilege escalation
4. What is a good practice regarding container privileges in secure AI deployment?	1) Use root privileges for all containers2) Allow containers to share host resources3) Run containers with the least privilege necessary4) Disable sandboxing for performance
5. What is the primary function of Infrastructure as Code (IaC)?	1) Encrypt user data before training2) Dynamically resize training datasets3) Automate and version control IT infrastructure4) Containerize neural network weights
6. What kind of monitoring focuses on differences in model outcomes across demographic groups?	1) Performance monitoring2) Bias monitoring3) Network monitoring4) Supply chain monitoring
7. What can an AI system leaking sensitive data via its outputs indicate?	1) Successful concept drift detection2) An effective model registry3) A privacy risk or potential breach4) A well-calibrated output filter
8. Which of the following is a suitable response to detecting significant concept drift in a deployed AI model?	1) Increase container memory2) Revert to older training datasets3) Retrain or roll back the model4) Disable CI/CD pipelines
9. Why should ethical monitoring be implemented in high-risk AI systems?	1) To optimize infrastructure costs2) To meet environmental regulations3) To ensure

	alignment with model card expectations and social responsibility4) To enable auto-scaling
10. What is the goal of an AI-specific incident response plan?	1) To minimize GPU usage2) To respond to AI failures such as model bias or adversarial inputs3) To encrypt training pipelines4) To disable user access logs

Exercise 8.5. Solutions

Click to reveal solutions

1. **Answer:** 3) To detect drift and ensure continued alignment with intended use
   Explanation: Monitoring after deployment helps identify concept or data drift and performance degradation, ensuring the model remains reliable and compliant.

2. **Answer:** 2) To describe the model’s ethical implications and usage limitations
   Explanation: Model cards provide structured documentation for transparency, including ethical considerations, intended use, and performance.

3. **Answer:** 2) Feature distribution comparison
   Explanation: Comparing feature distributions over time (e.g., using PSI or statistical tests) is a key method for detecting data drift.

4. **Answer:** 3) Run containers with the least privilege necessary
   Explanation: For security, containers should operate under minimal privileges to reduce the risk of exploitation.

5. **Answer:** 3) Automate and version control IT infrastructure
   Explanation: IaC allows the use of code to manage infrastructure reproducibly and consistently across environments.

6. **Answer:** 2) Bias monitoring
   Explanation: Bias monitoring focuses on evaluating whether AI systems treat different demographic groups fairly during deployment.

7. **Answer:** 3) A privacy risk or potential breach
   Explanation: Sensitive data appearing in outputs may indicate memorization of training data and constitutes a serious privacy concern.

8. **Answer:** 3) Retrain or roll back the model
   Explanation: Corrective action for significant drift includes retraining with updated data or reverting to a more stable version.

9. **Answer:** 3) To ensure alignment with model card expectations and social responsibility
   Explanation: Ethical monitoring ensures that real-world use aligns with documented intentions and protects fundamental rights.

10. **Answer:** 2) To respond to AI failures such as model bias or adversarial inputs
    Explanation: AI-specific incident response plans handle incidents involving failures in model performance or unintended outcomes.

9. Continuous validation, re-evaluation, decommissioning

Learning outcomes

After completing this chapter, you will:

- Learn strategies for continuous learning and model retraining, including balancing large-scale retraining with incremental updates and preventing catastrophic forgetting.
- Understand the importance of re-evaluating AI models to ensure alignment with evolving legal and societal norms.
- Familiarize with the retirement process for AI systems
- Explore techniques for managing technical debt, ensuring future-proof security, and promoting the long-term sustainability of AI systems through energy-efficient practices and lifecycle planning.

In this chapter, we close the loop with the AI system lifecycle by considering the *continuous validato*, *re-evaluation*, and *retirement* stages, focusing on the processes of continuously improving, realigning, and safe decommissioning of an AI system that processes personal data.

9.1 Continuous learning and model retraining

We saw in the previous chapter that after an AI model is deployed, its performance can degrade, with all sorts of *model drifts*. While monitoring is crucial to identify when an AI system is not performing as it should, monitoring without a reaction plan on what to do in case of model/system mis-alignment becomes useless.

Once drift is detected or model performance suddenly drops, the next step is deciding on how to retrain the model. Two common strategies are **incremental updates** and **full re-training**. In an **incremental retraining** approach, the model is updated with new data (which could come via online learning from the model usage or with periodic batch updates) without starting from scratch. This approach is faster and can adapt continuously, but it can come with the risk of *catastrophic forgetting*. Catastrophic forgetting refers to the tendency of AI system to lose performance on previously learned tasks when trained sequentially on new tasks (Ramasesh, Lewkowycz, and Dyer 2022). In neural networks, catastrophic forgetting happens because learned representations are overwritten during training on new tasks. This overwrite originates from the fact that the model’s parameters are updated to optimize performance on the current task, potentially at the cost of previous knowledge. This could be the case when there is an overlap in representations (different classes or tasks are not well separated) or simply due to model capacity for smaller models.

On the other hand, **full re-training** involves rebuilding the model from the ground up using a combination of old and new data (or only new data if statistical properties of the data have changed a lot). Full re-training can be more computationally expensive but could provide a cleaner model that avoids biases accumulated in incremental updates.

So what is the best strategy to adopt? In practice, depending on the size of the model, many organizations use a hybrid approach: periodically performing a full re-train (monthly or yearly) to create new versions of the model on the latest data, and alternating with smaller incremental updates if urgent changes are needed.

Automated retraining is available in various MLops pipelines (e.g. Amazon sagemaker, MLflow) so that monitoring events can trigger incremental retraining. It is maybe now clearer why we have insisted so much on previous chapters on the importance of version control of code and models, CI/CD and testing, because, with full automated responses to AI system misbehaviour, we need to be able to trace back what changed in the model and how automated testing went, and eventually revert to previous versions or decide for an alternative strategy (e.g. taking the AI system off-line or using other strategies like input/output filtering to prevent those cases that elicit lower performance of the AI system/model).

With personal data in the loop, one of course might need to be extra careful when it comes to fully automated pipelines doing re-training and deployment. In AI systems with personal data, any automated changes must still respect privacy constraints and be carefully documented for compliance with regulations, but also to ensure that fundamental subject rights and ethical alignment are still respected when processing data from individuals.

9.2 Revising Models for evolving ethical and regulatory alignment

If you have been curious about the AI regulatory landscape in Europe (and the rest of the world), it can feel impossible to follow all new developments, risks, or just new interpretation of the regulations, which can trigger retraining for your AI model. Ensuring that you have established processes for ethical and regulatory re-alignment of your AI systems/models makes it more transparent and fair not only towards the data subjects that are being processed, but also within your organisation, and society at large.

Model auditing and fairness assessments can be conducted by interdisciplinary teams or external experts, examining an AI system for issues like bias, robustness, and compliance. For instance, an audit might reveal that a facial recognition model has higher false negatives for darker-skinned individuals (Wehrli et al. 2022). On discovering these types of biases, the model could be retrained with more diverse data, or algorithmic fairness techniques (like equalizing thresholds or using adversarial debiasing) could be adopted. Audits should be done not just at deployment but periodically after deployment, since model updates or drifting data could introduce new biases. For further exploring fairness evaluations, the reader is encouraged to check the **IEEE 7003** standard or the **NIST AI 600-1 Artificial Intelligence Risk Management Framework** which include bias and harm mitigation solutions.

In the context of personal data, auditing also means ensuring that privacy is preserved – e.g., checking that the model isn’t inadvertently memorizing sensitive personal information. GDPR mandates that personal data should not be kept longer than necessary and should be used only for the purposes consented to. Over an AI system’s life, developers might need to update the model to *forget* specific data if a user invokes their *right to erasure (right to be forgotten)*. This has led to research on *machine unlearning* (see later chapters), which aims to remove or suppress the influence of particular training data points without retraining from scratch. The European Data Protection Board has noted that controllers should consider post-training techniques to remove or suppress personal data from trained AI models despite technical challenges (European Data Protection Board (EDPB) 2024). This means that if an AI model is found to memorize personal information (e.g., an AI assistant reciting someone’s address from training data), the developers are expected to find ways to expunge that memorization or otherwise mitigate the privacy risk.

For future MLops engineers and security professional, this stage of the AI lifecycle is blurred even further into the realm of AI governance. At this stage this is not only a matter of technical decisions or versioning and testing, it becomes an issue that needs to be solved with all parties involved, and this is why the final loop-back link in the lifecycle is to go back to the “Inception” stage and re-design the AI system if needed.

9.3 Decomissioning: safe retirement of AI systems

All AI systems eventually reach the end of their useful life, whether due to obsolescence, replacement by better models, or changes in business needs or regulations. **Decommissioning** an AI system – especially one that processes personal data – must be done in a responsible and structured manner. This ensures that no personal data is improperly retained, dependencies are resolved safely, and compliance is maintained even as the system is retired. In this section, we cover the steps and considerations for responsible AI decommissioning, including data retention and deletion policies, archival of models/data, compliance audits, and risk assessments during the process.

9.3.1 Responsible AI System Decommissioning

Decommissioning is more than just "turning off" an AI system/model. It should be handled according to a **decommissioning plan or protocol** that should be planned in advance, ideally during the inception stage.

Without looking at the governance aspects of the process, if we focus on the technical side, decommissioning relates to dependencies that the current AI system might carry: - **Upstream Dependencies:** data sources, data pipelines. (E.g., does a data ingestion job need to be stopped? Do data providers need to be notified that we no longer require data?) - **Downstream Dependencies:** applications or processes consuming the model's output. (E.g., an API that other services call, a dashboard that displays model results, business processes that rely on AI outputs.) - **Associated Resources:** compute instances/clusters, databases, model artifacts, configuration files, documentation, and possibly third-party services or licenses.

By mapping all the dependencies, one can avoid orphaned processes or broken pipelines post-decommissioning and in the case of the processing of personal data, it ensures that there are no unwanted consequences for the data subjects (e.g. if an AI system for fraud detection is decommissioned, the data subjects should not receive false alarms of credit card fraud only because an API somewhere was not switched off).

During decommissioning, it's good practice to come up with a checklist. Typical steps might include:

- **Notify all parties:** Inform all relevant parties (business owners, IT, data protection officer, end-users if needed) of the intention to decommission, and the expected date. This ensures no one is caught off guard and can voice any concerns (perhaps someone relies on the system unknown to the team).
- **Documentation and Knowledge Capture:** Before shutting down, ensure all documentation about the system is up to date and preserved. Write a decommissioning report that explains why the system is being retired, the date, the responsible personnel, and any important contextual information (like "model X was in production from 2022-2025, used for Y purpose"). This is useful for future audits or if questions arise later (e.g., "why did we discontinue that model?").
- **Final Performance and Compliance Snapshot:** It can be useful to record the system's final state – performance metrics, any outstanding issues, the versions of software. Also verify compliance one last time (for instance, ensure that there are no outstanding data subject requests or regulatory holds on the data that would prevent deletion).
- **Gradual Phase-Out (if applicable):** For critical systems, you might run the old and new system in parallel for a short period to ensure the replacement is fully functional. If the system is just being removed without direct replacement, ensure the business

has adjusted (e.g., maybe a process reverts to manual review instead of AI). A sudden removal without adaptation can cause business disruption, which is a risk in itself.

9.3.2 Data Retention and Deletion

One of the most important aspects of decommissioning an AI that processed personal data is handling that data properly at end-of-life. Data should not be retained longer than necessary for the purpose it was collected and when an AI system is retired, it likely means the purpose no longer applies.

For the technical team involved in such task, it is important to consider at least these points:

- **Identify all personal data stores:** This includes training datasets, validation datasets, data collected during operation (input logs, user feedback containing personal info), and any embedded data within the model. Also consider backup copies and data in non-production environments.
- **Determine retention requirements:** Sometimes, laws or policies may require keeping data for a certain period even after system decommission (for example, financial records might need storage for X years, or medical AI decisions might need to be stored for liability reasons).
- **Securely erase data:** Using cryptographic erasure (destroying encryption keys so data becomes irretrievable) or overwriting storage following standards (for hardware you control, refer to standards like NIST SP 800-88 “Guidelines for Media Sanitization” or ISO/IEC 27040). Remember that simply deleting a file or shutting off a cloud instance might not fully remove the data from backups or persistent storage so we must ensure that any personal data in the system is rendered unrecoverable.
- **Document the deletion process:** Record what data was deleted and when, and who authorized it so that the organization can prove that as of the decommission date, that user’s data (along with all others in that system) was deleted in line with their policy. Keeping logs of deletion operations (like cryptographic erasure logs or certificate of destruction if a third-party did it) is a good practice.

One complication is with the model parameters themselves: A trained model could in theory retain information about individuals in its weights (especially if it overfit or memorized examples (European Data Protection Board (EDPB) 2024). This means a model might itself be subject to data protection rules unless you can demonstrate that no personal data can be extracted. During decommissioning, you should evaluate if the model file needs special handling. If the model is going to be archived or handed off, ensure it’s either thoroughly vetted for privacy or treat it with the same care as raw personal data (access control, encryption, eventual deletion).

9.3.3 Celebrate

Finally, when all is done, celebrate the proper retirement of the system: it served its purpose, and now it’s responsibly laid to rest, with no loose ends. Responsible decommissioning is a sign of a mature AI governance program that shows respect for user data, maintain trust with all data subjects involved, and clear the way for new innovations.

9.4 Summary

With this chapter our journey through the AI lifecycle comes to an end. There are a few broader perspective to consider on “where to go next”, but also what implications AI systems can have when it comes to sustainability. There are also a few advanced cases to consider where more complex AI systems might introduce further privacy risks and these will be covered in the last remaining chapters of the book.

Exercise 9.1: Multiple choice questions

Question	Options
1. What is a primary risk of using incremental retraining in AI models?	1) Increased latency.2) Catastrophic forgetting.3) Lack of model interpretability.4) Model overcompression.
2. Why might full re-training be preferred over incremental updates?	1) It is faster to deploy.2) It always uses less energy.3) It helps avoid biases from accumulated updates.4) It guarantees explainability.
3. What is the role of automated retraining pipelines in MLOps?	1) Encrypt data before training.2) Launch new models manually.3) Trigger model updates based on monitoring events.4) Avoid the need for model versioning.
4. What is the 'right to erasure' under GDPR relevant to in the context of AI?	1) Deleting input data logs.2) Removing the model entirely.3) Ensuring specific user data influence can be removed from a trained model.4) Terminating user accounts.
5. Which of the following is a goal of machine unlearning?	1) Speeding up model inference.2) Replacing outdated APIs.3) Enabling selective removal of training data influence.4) Compressing model weights.
6. What is one reason to periodically audit AI models post-deployment?	1) To reduce cloud computing costs.2) To check for new biases or misalignments.3) To enhance training datasets.4) To update license keys.
7. Which standard offers guidance on bias and harm mitigation in AI systems?	1) ISO/IEC 42001.2) IEEE 7003.3) GDPR Article 32.4) ISO 27017.
8. Why is it important to identify both upstream and downstream dependencies during decommissioning?	1) To allow new AI models to reuse components.2) To avoid data being stored in GPUs.3) To prevent orphaned processes and ensure graceful retirement.4) To migrate all services to Kubernetes.
9. What is cryptographic erasure used for during AI system retirement?	1) Obfuscating log files.2) Speeding up shutdown.3) Making data irretrievable by destroying encryption keys.4) Encrypting model weights for future use.
10. What does responsible decommissioning of an AI system demonstrate?	1) High energy efficiency.2) A complete handover to DevOps teams.3) A mature AI governance practice.4) Full automation of AI pipelines.

Exercise 9.1. Solutions

Click to reveal solutions

1. **Answer:** 2) Catastrophic forgetting
   Explanation: Incremental updates can cause a model to forget previously learned tasks if not handled properly.

2. **Answer:** 3) It helps avoid biases from accumulated updates
   Explanation: Full re-training can result in a cleaner model and mitigate issues that arise from incremental changes.

3. **Answer:** 3) Trigger model updates based on monitoring events
   Explanation: Automated MLOps pipelines can initiate retraining workflows when performance degrades.

4. **Answer:** 3) Ensuring specific user data influence can be removed from a trained model
   Explanation: This is part of implementing the GDPR’s right to erasure in the AI context.

5. **Answer:** 3) Enabling selective removal of training data influence
   Explanation: Machine unlearning techniques aim to remove the effect of certain data without full retraining.

6. **Answer:** 2) To check for new biases or misalignments
   Explanation: Regular audits ensure ongoing fairness, accuracy, and regulatory compliance.

7. **Answer:** 2) IEEE 7003
   Explanation: This standard specifically addresses bias and mitigation in AI systems.

8. **Answer:** 3) To prevent orphaned processes and ensure graceful retirement
   Explanation: Mapping dependencies helps avoid unexpected disruptions and compliance issues.

9. **Answer:** 3) Making data irretrievable by destroying encryption keys
   Explanation: Cryptographic erasure is a secure deletion method, especially for cloud environments.

10. **Answer:** 3) A mature AI governance practice
    Explanation: Proper decommissioning reflects responsibility, accountability, and long-term planning.

10. Auditing AI systems in practice

In this chapter we will cover: - Checklist for auditing AI systems from a data protection perspective - Focus on technology and not on legal compliance (work should be done with dpo or other legal experts) - Evaluating AI systems doing procurement

This chapter provides a collection of checklists and good practices gathered from international standards, peer reviewed publications, and work in progress in the broad community of artificial intelligence and privacy experts. The checklists are using the following *tags* to identify the audience and the type of checklist.

Target Audience Tags

Since checklists can be useful for different experts working on an AI system, these tags define the audience for the checklist.

Tag Name	Explanation
AI Governance	Responsible for overseeing AI policies and regulations from inception to decommissioning.
Data Engineers	Handle the preparation, management, and optimization of data pipelines in the early stages of development.
ML Engineers	Focused on model development, training, and optimization during the development phase.
Cybersecurity Experts	Involved in securing AI infrastructure and systems from development through deployment and monitoring.
Data Privacy Officers	Ensure data protection throughout the AI lifecycle, from inception through monitoring and decommissioning.
Cross-functional Teams	Involved throughout the lifecycle, supporting collaboration between different roles.

Type of Checklist Tags

These tags define the type and scope of the checklists.

Tag Name	Explanation
Ethics	Considerations for fairness, transparency, and responsibility, applied throughout the entire lifecycle from inception onward.
Compliance	Ensuring adherence to legal, regulatory, and industry standards, from the design phase through decommissioning.
Data Management	Covering data handling, governance, and storage, primarily during data preparation and development but extending through the entire lifecycle.
System Security	Focused on securing systems and software during development, deployment, and monitoring.
Data Privacy	Ensuring the protection of personal data across all stages, especially during development and deployment.
Monitoring	Continuous assessment of AI system performance and security during deployment and production.

Auditability	Maintaining traceability and accountability throughout the lifecycle, especially during deployment, monitoring, and decommissioning.

10.1 AI Impact Assesment

Checklist card

Name:	AI Impact Assessment
Audience:	AI Governance, Data Privacy Officers, Cross-functional Teams
Type:	Compliance, Auditability, Ethics
Reference:	ISO/IEC DIS 42005 AI system impact assessment, ref1
Comment:	Under development

This is currently at a draft stage; the checklist outlines a structured process for assessing AI systems.

10.2 PLOT4ai

Checklist card

Name:	PLOT4ai
Audience:	AI Governance, Data Engineers, ML Engineers, Cybersecurity Experts, Data Privacy Officers, Cross-functional Teams
Type:	Compliance, Auditability, Ethics, Data Privacy
Reference:	PLOT4ai
Comment:	All questions can be accessed in machine-readable format at PLOT4ai Github respository

The PLOT4ai (Privacy, Lawfulness, Oversight, and Trust for AI) project provides a structured and practical framework for identifying and mitigating risks in the development and deployment of AI systems. It supports developers, researchers, and organizations in building AI responsibly by offering tools for threat modeling, risk assessment, and compliance with legal and ethical standards. Central to the project is a curated set of “threat cards,” which guide users in recognizing potential harms and implementing safeguards throughout the AI lifecycle. PLOT4ai promotes responsible AI innovation grounded in principles of data protection, transparency, and accountability.

Threat Card Categories and Usefulness:
The PLOT4ai threat cards (86 cards) are organized into several key categories that reflect different areas of concern in AI systems:

- **Privacy**: Includes threats like data leakage, re-identification, and inference attacks; useful for identifying risks to personal data and ensuring GDPR compliance.
- **Security**: Covers attacks like adversarial examples, model inversion, and data poisoning; essential for securing AI models and infrastructure.
- **Fairness and Bias**: Highlights issues like discriminatory outcomes and representation bias; helps promote equity and inclusiveness in AI decisions.
- **Transparency and Explainability**: Focuses on challenges in understanding model behavior and outputs; supports trustworthy and interpretable AI.

- **Accountability and Governance**: Emphasizes documentation, oversight, and role clarity; useful for aligning with legal frameworks and organizational responsibilities.
- **Misuse and Abuse**: Points to risks of dual use, malicious repurposing, and unintended consequences; aids in evaluating and preventing harm from system misuse.

These categories are useful in AI development, auditing, and teaching to promote a culture of risk awareness and proactive mitigation across technical, ethical, and legal dimensions.

To see the full list of cards, click the box below.

10.3 BayLDA AI Checklist

Checklist card

Name: BayLDA AI Checklist

Audience: AI Governance, Data Engineers, ML Engineers, Cybersecurity Experts, Data Privacy Officers, Cross-functional Teams

Type: Compliance, Auditability, Data Privacy, Ethics, Data Management, System Security, Monitoring

Reference: BayLDA AI Checklist

Comment: Checklist based on the Bavarian Data Protection Authority (BayLDA), aligned with GDPR requirements and EU guidelines on Trustworthy AI.

The **BayLDA AI Checklist** by the Bavarian Data Protection Authority provides structured guidelines for ensuring GDPR compliance throughout the AI systems lifecycle. The checklist is structured around four main areas:

1. **Classification of AI systems**:
   - Clarifies the AI technologies in use, such as large language models, machine learning approaches, or rule-based systems.
   - Considers hosting environments, preprocessing, post-processing, and filtering mechanisms for input and output data.
2. **Training of AI Models**:
   - Specifies and documents AI architectures and training procedures.
   - Evaluates the use of personal, pseudonymised, anonymised, or synthetic data, ensuring clear legal bases (particularly for sensitive data).
   - Requires comprehensive documentation of data sources and quality assurance measures to mitigate bias.
   - Establishes the necessity of conducting Data Protection Impact Assessments (DPIA) under GDPR (Art. 35) and emphasizes maintaining data subject rights, including access, rectification, deletion, and data portability (Arts. 15-21 GDPR).
3. **Risk Assessment of AI Systems**:
   - Encourages creating and regularly updating a structured risk model based on key data protection and ethical principles from the EU Guidelines for Trustworthy AI.
   - Core principles include fairness (avoiding discrimination), autonomy and control (human oversight), transparency (explainability and accountability), reliability (accuracy and robustness against manipulation), security (protection against unauthorized access and manipulation), and data privacy compliance.
4. **Operational Deployment of AI Applications**:

  - Outlines clear documentation and legal basis requirements for deploying AI applications, particularly in scenarios involving cloud providers or third-party services.
  - Stresses the importance of continuous monitoring and logging of AI systems, comprehensive testing prior to deployment, and systematic handling of data subject requests and rights.
  - Advocates integration of AI system usage within organizational training and awareness programs, ensuring transparency and compliance throughout ongoing operations.

Here’s the checklist card and summary for the provided checklist, titled **“AI and Data Protection Risk Toolkit”** (ICO - UK GDPR aligned):

10.4 AI and Data Protection Risk Toolkit (ICO)

Checklist card

Name:	AI and Data Protection Risk Toolkit (ICO)
Audience:	AI Governance, Data Privacy Officers, ML Engineers, Cybersecurity Experts, Cross-functional Teams
Type:	Compliance, Auditability, Data Privacy, Monitoring, Ethics
Reference:	ICO AI and Data Protection Risk Toolkit
Comment:	Checklist structured according to AI lifecycle stages, aligned explicitly with UK GDPR Articles and ICO guidelines.

The **AI and Data Protection Risk Toolkit** provided by the UK’s Information Commissioner’s Office (ICO) is a comprehensive framework designed to assist organizations in identifying, assessing, and mitigating data protection risks associated with AI systems. The checklist is structured according to the AI lifecycle stages, systematically addressing data protection risks:

- **Business Requirements and Design**
  - Emphasizes accountability, requiring Data Protection Impact Assessments (DPIAs) to ensure thorough identification and mitigation of risks to individual rights and freedoms.
- **Risk Areas and Controls**
  - Risk statements linked to GDPR provisions, accompanied by control objectives and practical steps.
  - Controls include DPIAs, consulting domain experts, and engaging with affected individuals or their representatives.
- **Assessment and Documentation**
  - Requires documentation of inherent and residual risks, controls implemented, their ownership, and current status, including planned completion dates for risk mitigation actions.
- **Guidance and Best Practices**
  - Provides direct references to detailed ICO guidance documents for implementing specific GDPR compliance measures, ensuring organizations have authoritative resources for further clarification and detailed actions.

Here’s your checklist card and summary for the provided checklist titled **“AI Auditing Checklist”** (European Data Protection Board - Support Pool of Experts Programme):

10.5 AI Auditing Checklist (EDPB's SPE Programme)

Checklist card

Name: EDPB AI Auditing Checklist (SPE Programme)

Audience: AI Governance, Data Privacy Officers, ML Engineers, Cybersecurity Experts, Cross-functional Teams

Type: Compliance, Auditability, Data Privacy, Ethics, Monitoring

Reference: EDPB AI Auditing Checklist (SPE)

Comment: Checklist developed under the EDPB Support Pool of Experts Programme, specifically aligned with GDPR and focused on bias, fairness, transparency, and accountability of AI systems.

The **AI Auditing Checklist**, developed within the European Data Protection Board's (EDPB) Support Pool of Experts Programme (SPE), provides a structured methodology to audit algorithmic systems comprehensively. It emphasizes socio-technical auditing, focusing explicitly on the fairness, bias, transparency, accountability, and overall social impacts of AI systems, particularly those based on machine learning algorithms. The checklist presents five auditing activities which end up forming the audit report:

1. **Model Card**:
    - Documentation about AI models, including purposes, data characteristics, methodologies, bias metrics, and redress mechanisms.
    - Accountability through detailed documentation aligned with GDPR articles, facilitating clarity and transparency from the outset.
2. **System Map**:
    - Maps the interactions between AI algorithms, technical systems, and decision-making processes.
    - Clarifies responsibilities and identifies transparency requirements, particularly regarding AI component identification, management roles, documentation, traceability, and explainability.
3. **Moments and sources of bias**:
    - Defines and categorizes moments (from data collection to decision-making) and sources of bias (historical bias, aggregation bias, selection bias, etc.).
    - Emphasizes thorough bias evaluation, documentation, and active mitigation strategies at every stage (pre-processing, in-processing, post-processing).
4. **Bias testing**:
    - Suggests explicit bias testing strategies to ensure fairness across protected groups (race, gender, socio-economic status, etc.) through defined statistical fairness metrics.
5. **Adversarial Audits**:
    - Adversarial auditing to uncover hidden biases or unintended behaviors by actively testing systems in real-life scenarios.

11. Advanced cases with AI systems and data protection

In this last chapter we cover a few more advanced cases. The content of this chapter is useful to explore more special scenarios: the teacher could assign these as essays topics or small literature research tasks.

11.1 Collection of cases by the EDPS, November 2024

The EDPS released an excellent report (European Data Protection Supervisor 2024) on six emerging trends that combine novel AI applications with data protection issues: retrieval-augmented generation, on-device AI, machine unlearning, multimodal AI, scalable oversight, neurosymbolic AI.

11.1.1 **Retrieval-Augmented Generation (RAG)**

Retrieval-Augmented Generation (RAG) combines the strengths of generative AI with one or more external knowledge bases. For example, a system might generate a detailed medical report using a large language model by combining its internal capabilities with external verified medical databases. This improves the accuracy and relevance of outputs, limiting the risks of hallucinated content in the RAG output. However, the integration of external data introduces risks related to data protection. If the external source of knowledge contains sensitive information (e.g. personal data), it could be retrieved inadvertently e.g. via prompt injection attacks techniques increasing the risk of data leakage in the AI responses. Understanding and safeguarding how external data is integrated with the AI system is important to mitigate such risks. A good reference on the topic is Zeng et al. (2024).

Box: Model Context Protocol

Definition:

The *Model Context Protocol (MCP)* is a relatively new open protocol designed to enable AI systems (typically generative AI systems based on large language models) to dynamically interact with external tools, resources, APIs, and data in a modular and interoperable way. It allows AI agents to autonomously discover, invoke, and operate tools, without manual API integration. For example, an AI agent integrated with MCP can autonomously fetch live stock prices using a financial API, summarize the results, and notify the user via email. Another example is a code assistant like Cursor, which can use MCP to interact with the version control system of the project, test frameworks, and deployment tools directly from the development environment, allowing developers to execute complex workflows with simple natural language commands.

As the reader can already imagine, with so much power and so much automation, security or data protection risks are just behind the corner. For example, if someone sets up a fake MCP server (*installer spoofing*) or tool with a misleading name (*name collision*), an AI agent might accidentally send personal data or private records to the wrong place. This can happen without the user knowing, especially if the tools look trustworthy. Because MCP allows AI to automatically connect with many different services, a poorly secured tool can greatly increase the risks of a data breach. For more considerations on security and privacy aspects of MCP, see Hou et al. (2025).

11.1.2 On-device AI

On-device AI performs processing directly on a user's device rather than relying on remote servers. This setup is beneficial because it reduces response times and enhances privacy by keeping data local. For example, voice assistants like Apple's Siri could process certain commands without sending data to the cloud. On-device AI however can still carry risks to data protection, especially when it is never fully clear if other operations around the AI system are sending data elsewhere. Sensitive data stored locally might be accessed without the user's awareness, or systems may create profiles of user behavior derived from the data about what apps are used, when, and how. These profiles could then be shared with third parties, potentially violating data minimization (only collecting what is strictly necessary) and purpose limitation (using data only for specific, agreed purposes) principles. Proper safeguards are necessary to ensure that locally processed data is not exploited or misused.

11.1.3 Machine unlearning

Machine unlearning is a method where AI systems "forget" specific data upon request. This aligns with rights granted under data protection laws like the GDPR's right to erasure, which allows individuals to request the removal of their data. For instance, a machine learning model trained on medical data could unlearn contributions from a specific patient when they request it. However, unlearning comes with challenges. If data from certain groups is removed disproportionately, it could introduce biases into the AI system, leading to unfair treatment, inaccurate predictions, and eventually catastrophic forgetting. Moreover, membership inference attacks techniques con be exploited on the model with forgotten data, to reveal which data points were erased, compromising privacy. Machine unlearning is a very active field of research, for an introductory review on the topic see H. Zhang et al. (2023).

11.1.4 Multimodal AI

Multimodal AI systems combines different types of data, like text, images, and sound, to make more informed decisions. A self-driving car, for instance, uses visual data from cameras, audio signals, and spatial data from sensors to navigate safely. While this integration is powerful, it can also raise privacy concerns. Biometric data, like facial images or voice recordings, are often part of multimodal inputs, and mishandling this sensitive information could lead to serious data protection violations. For example, if a system misuses facial recognition data, it might unfairly discriminate against certain individuals or violate laws governing sensitive personal data. Additionally, multimodal systems often require large datasets, increasing the chances of data breaches or misuse during collection and processing.

11.1.5 Scalable Oversight

Scalable oversight refers to using tools and feedback loops to ensure AI systems remain aligned with human values and safety standards, even as they grow in complexity. For instance, feedback mechanisms can teach AI systems to avoid generating harmful content or making unethical recommendations. While scalable oversight improves AI reliability, it can create data retention issues (retaining identifiable information longer than necessary). Poorly designed oversight systems might also fail to detect nuanced risks, such as subtle biases in model outputs or errors affecting minority groups.

11.1.6 **Neuro-Symbolic AI**

Neuro-symbolic AI combines the pattern recognition abilities of neural networks with the structured reasoning of symbolic systems. This approach can make AI systems more interpretable and accurate. For instance, a neuro-symbolic system in healthcare could use neural networks to identify potential diseases from medical images and symbolic reasoning to explain its diagnosis logically. However, the reliance on structured reasoning introduces risks when these systems are trained on sensitive datasets, such as medical or financial records. If reasoning rules are improperly derived or poorly managed, they might make unfair decisions or inadvertently expose sensitive information. Automated decision-making based on flawed logic could violate data protection laws, particularly if individuals are not informed or cannot challenge the decisions. Transparency in how reasoning is applied and safeguards to protect sensitive data are crucial to address these risks.

11.2 Named Entity Recognition systems

A report commissioned by the Support Pool of Experts of the EDPB covered the possible privacy risks associated to named entity recognition (NER, Barberá (2023a)).

NER is a method used in Natural Language Processing (NLP) to identify specific entities, such as names, organizations, or locations, within a text and classify them into predefined categories. It is widely used across sectors such as healthcare, legal analysis, and customer support. NER systems can employ lexicon-based, rule-based, or machine learning-based methods, with modern systems often relying on machine learning due to their adaptability.

The report identifies significant privacy risks associated with NER technology. Key concerns include the processing of sensitive data (e.g., medical records or criminal data), which could lead to serious harm if mishandled. Risks also arise from large-scale data processing, where breaches could have amplified effects. Additionally, issues like low-quality input data or insufficient security measures could lead to incorrect decisions, adversely affecting individuals’ rights. These risks further increase when NER solutions involve cloud-based third-party services, as they may store data for extended periods, share it without proper consent, or process it in jurisdictions lacking adequate data protection laws.

11.3 Optical Character Recognition

Another report commissioned by the Support Pool of Experts of the EDPB covered the possible privacy risks associated to Optical Character Recognition (OCR, Barberá (2023b)).

OCR is a technology that extract text from images, scanned documents, or physical records into machine-readable formats. Modern OCR systems can use machine learning (ML) and deep learning (DL) models to handle structured, semi-structured, and unstructured documents. These systems often employ stages such as detection, localization, and segmentation to identify and extract text accurately. OCR technology can also present risks to data protection and privacy. Such technology might be used with sensitive data (e.g health data) or in connection with large-amount ot data, further increasing the changes of unauthorised access, or data breaches. OCR can also produce wrong outputs, like misrecognition of text, which may adversely impact individuals if used in automated decision-making processes.

11.4 Summary

We have covered a few advanced cases, and some very new directions of AI systems, towards more *agentic* systems that can automate possibly everything. We are basically done with this book, in the next and final chapter we will do a final summary and where to go from here.

Conclusions and future directions

This book has hopefully offered a structured and practical introduction to the intersection of artificial intelligence, cybersecurity, and GDPR-compliant data processing.

We began with the basic concepts of AI literacy, ethical considerations, risk assessment, and then added further layers with data protection and cybersecurity to provide a comprehensive reference for learners (and teachers!) of such topics.

Secure AI development with personal data goes beyond technical compliance. It demands an ongoing, collaborative effort involving developers, data scientists, cybersecurity teams, legal experts, and the final users of such systems to achieve lawful, but most important, trustworthy AI systems.

We have covered privacy enhancing technologies in the context of machine learning and have explored the practical applications of MLOps with secure coding and deployment practices.

Sustainability

We cannot ignore the fact that AI systems, especially those requiring large amount of resources during training or inference, can have significant environmental, financial, and operational impacts. Training modern AI models, particularly deep learning models with billions of parameters, is an energy-intensive process. Recent studies have highlighted the substantial carbon footprint of these models. Between 2012 and 2018, the computational resources required for leading-edge deep learning research grew 300,000 times, leading to surprising levels of energy consumption and carbon emissions (Schwartz et al. 2020). Running large AI models in production continuously also draws significant power: one estimate suggested that each query to an AI like ChatGPT consumes multiple times more energy than a typical Google search. While considering the societal and environmental aspects of AI is beyond the scope of this book, we should always responsibly consider if AI is the tool we need for our task.

Where to go next?

As AI technologies continue to evolve, so too will the privacy and security challenges they present. So what should be monitored?

- **Emerging technologies:** New trends such as agentic AI systems, machine unlearning, model context protocol will introduce new regulatory, ethical, and technical challenges.
- **Growing threats:** Adversarial attacks, data poisoning, and model inversion will require increasingly sophisticated threat detection, monitoring, and incident response strategies.
- **Regulatory evolution:** The implementation and interpretation of the EU AI Act, alongside ongoing GDPR enforcement and new ISO standards, will continue to shape how AI systems must be built and maintained and surely parts of this book will need to be rewritten in a couple of years ahead.

How to stay informed and what to do next? We recommend at least the following:

- **Follow regulatory changes:** Follow updates from the EDPB, EDPS, ENISA, the European Commission, and national data protection authorities to watch how the regulatory landscape is evolving around artificial intelligence.

- **Interact with professional communities:** Join forums, special interest groups, and attend conferences focused on AI, data protection, and cybersecurity, with a focus on your role in developing such systems.
- **Continuous learning:** With this book we barely scratched the surface, you most likely want to purse further training or certifications in GDPR, cybersecurity, MLOps, and AI ethics, and in general experiment with new technologies as they are being released.
- **Contribute to Open Resources:** Open-source tools are the true leaders of the AI innovation we see today, so you should onsider contributing back to the tools you use. And you can also contribute to initiatives like OWASP's AI Security and Privacy Guide!

Final thoughts

As AI continues to transform society, your role as a developer or cybersecurity professional carries immense responsibility. The systems you build impact individuals, institutions, and societies. By applying the knowledge, tools, and principles outlined in this book, you are not just building software, you are helping to shape the future of ethical, secure, and trustworthy AI. And if you're feeling extra motivated and inspired, please consider contributing back to this book by expanding it further.

Enrico Glerean, Helsinki, February 2025

Appendix

In this appendix we cover a few extra concepts that did not fit in the main module, but might be important for the learners to familiarise with. The learners are encouraged to explore further topics by themselves.

Taxonomies of privacy risks

There are no comprehensive taxonomies of privacy risks. The taxonomy presented in Chater 2 is a synthesis of three taxonomies, specifically the one proposed in ISO 29134, the one proposed in the AEPD "Risk Management and Impact Assessment in Processing Personal Data" (Agencia Española de Protección de Datos 2021), and Solove's "Taxonomy of Privacy" (Daniel J. Solove 2005) extended with risks for each of the 16 dimensions of privacy.

The table below lists the risks for the three taxonomies.

Comparison between taxonomies of data protection risks

From ISO 29134:2020	AEPD (Agencia Española de Protección de Datos 2021)	Daniel J. Solove, A Taxonomy of Privacy, (Daniel J. Solove 2005).
unauthorized access to PII (loss of confidentiality);	Operations related to the purposes of processing - Risk factors deriving from the purposestated of the processing and other purposesrelated to the main purpose (e.g. contact tracing, deciding on data subjects' control of personal data, profiling, monitoring)	Surveillance: Continuous monitoring in public spaces can lead to a loss of anonymity and chilling effects on free movement.
unauthorized modification of the PII (loss of integrity);	Types of data used - Risk factors related to the scope of theprocessing that arise from data collected,processed or inferred in theprocessing. (e.g. financial transactions, special categories of personal data)	Interrogation: Overly invasive questioning by authorities might coerce personal disclosures, violating individual autonomy.
loss, theft or unauthorized removal of the PII (loss of availability).	Extent and Scope of Processing - Risk factors related to the scope of the processingrelating to the number of data subjects concerned,the diversity of data or aspects processed, theduration in time, the volume of data, thegeographical extent, the exhaustivenesson the person, frequency of collection, etc (e.g. large	Aggregation: Combining data across sources can reveal patterns, leading to profiling or unintended exposure of identity.

From ISO 29134:2020	AEPD (Agencia Española de Protección de Datos 2021)	Daniel J. Solove, A Taxonomy of Privacy, (Daniel J. Solove 2005).
	number of subjects, large scale processing)	
excessive collection of PII (loss of operational control);	Categories of Data Subjects - Risk factors related to the scope of theprocessing related to the category of datasubjects, such as employees, minors, elderlypeople, persons in a situation ofvulnerability, victims, disabled people, etc (e.g. children under 14 yo, vulnerable subjects)	Identification: Linking anonymous data to individuals can result in privacy breaches and potential harm from misuse.
unauthorized or inappropriate linking of PII;	Technical Factors of Processing - Risk factors that arise from the nature of theprocessing when implemented with certaintechnical characteristics ortechnologies. (e.g.internet of things, video surveillance, automated processing)	Insecurity: Poor security of stored data increases risks of breaches and unauthorized access.
insufficient information concerning the purpose for processing the PII (lack of transparency);	Data collection and generation - Risk factors that arise from thenature of the processing when data are specificallycollected or generated. (e.g. combinations of datasets)	Secondary Use: Using data for purposes other than originally intended without consent can erode trust.
failure to consider the rights of the PII principal (e.g. loss of the right of access);	Side Effects of Processing - Risk factors that arise from the processing contextas consequences may occur that are not foreseenin theoriginal intended purposes of the processing. (e.g. unauthorised re-identification, identity theft, reputational damage,)	Exclusion: Denying individuals the right to access or correct their information can lead to misinformation and unfair outcomes.
processing of PII without the knowledge or consent of the PII principal (unless such processing is provided for in the relevant legislation or regulation);	Category of controller/processor - Context-related risk factorsspecific to the sector of activity, business model ortype of entity (e.g.hospitals, financial institutions)	Breach of Confidentiality: Unauthorized disclosure of confidential information risks harm to reputation and personal relationships.

From ISO 29134:2020	AEPD (Agencia Española de Protección de Datos 2021)	Daniel J. Solove, A Taxonomy of Privacy, (Daniel J. Solove 2005).
sharing or re-purposing PII with third parties without the consent of the PII principal;	Data disclosure - Risk factors that arise from the context inwhich the data disclosures are made to thirdparties within the framework of the processing (e.g. regular transfers to other countries without adequate protection)	Disclosure: Public release of private information can lead to embarrassment, harassment, or discrimination.
unnecessarily prolonged retention of PII.	Data breaches - Risk factors that arise from the possiblematerialisation of personal data breaches.	Exposure: Making sensitive details accessible to others may violate personal boundaries and cause distress.
		Increased Accessibility: Easy access to personal data online can invite misuse or unauthorized surveillance by others.
		Blackmail: Threatening to reveal private information can coerce individuals into unwanted actions.
		Appropriation: Using personal likeness or data for commercial gain without consent undermines autonomy and rights.
		Distortion: Misrepresentation of personal information can damage reputation and create misunderstandings.
		Intrusion: Physical or digital invasion into private spaces disrupts privacy and can induce fear or discomfort.
		Decisional Interference: Intervening in personal decision-making processes infringes on autonomy and personal agency.

Artificial Intelligence Explainability

Another important domain from Slattery et al. (2024) which is strongly related to the GDPR principle of fairness and transparency is 7.4 “Lack of transparency or interpretability”, which leads us to the concept of AI Explainability. For more in-depth considerations, please refer to Leslie et al. (2024).

AI explainability refers to the degree to which a system or set of governance practices and tools support a person’s ability to:

1. Explain and communicate the rationale behind the behavior of the AI system.
2. Demonstrate that the processes behind its design, development, and deployment ensure sustainability, safety, fairness, and accountability across various contexts of use.

Explainability involves providing clear reasons for:

- The outcomes produced by the algorithmic model (whether used for automated decisions or as inputs for human decision-making).
- The processes used to design, develop, and deploy the AI system.

AI explainability is crucial for several reasons:

1. **Trust and accountability**: Explainability helps build trust in AI systems by allowing users and other relevant parties to understand why certain decisions were made and to hold developers and organizations accountable.
2. **Ethical compliance**: It ensures that AI systems comply with ethical standards by making sure the processes and outcomes can be communicated and justified transparently.
3. **Fairness and safety**: Providing explanations allows relevant parties to verify that AI systems are fair, safe, and not biased against certain individuals or group
4. **Regulatory requirements**: In many sectors, explainability is a regulatory requirement to ensure that AI systems are transparent, particularly when they involve sensitive decisions about people’s lives.

The overall goal is to ensure that AI systems can be understood and trusted by a wide range of audiences, including those affected by their decisions.

Machine Learning algorithms and their explainability

This table is taken from the Annex of Leslie et al. (2024).

An overview of explainability of various machine learning algorithms

Machine Learning Method	Explainability	Notes
Linear Regression (LR)	High	Simple, transparent model. Easy to interpret feature importance.
Logistic Regression	High	Similar to LR but applied to classification tasks. Interpretability is clear.
Generalized Linear Model (GLM)	Moderate	Extension of LR, handles non-normal distributions but may lose some transparency.
Generalized Additive Model (GAM)	High	Non-linear relationships still explainable through graphical representation.
Decision Tree (DT)	High	Interpretable as long as tree depth is manageable.
Rule/Decision Lists and Sets	High	Easy to follow but large lists may reduce interpretability.
K-Nearest Neighbors (KNN)	Moderate	Intuitive but less explainable for larger datasets.

Machine Learning Method	Explainability	Notes
Naive Bayes	High	Assumes feature independence; effective but can oversimplify relationships.
Support Vector Machines (SVM)	Low	Complex, especially in high dimensions; difficult to interpret decision boundaries.
Random Forest	Low	Aggregate of many decision trees, making overall model difficult to explain.
Artificial Neural Network (ANN)	Very Low	Highly non-linear with numerous parameters, making it a black-box model.
Ensemble Methods	Low	Combines several models, often leading to opacity. May require external interpretability tools.
Case-Based Reasoning (CBR)	High	Highly interpretable due to example-based reasoning.
Supersparse Linear Integer Model (SLIM)	High	Sparse model, interpretable by design with simple arithmetic calculations.

Basics of privacy enhancing technologies

When introducing PETs we need some concepts related to the types of personal data that are more nuanced that the simple definition from the GDPR. In this section we provide a reference on teh terminology used in other articles or regulations. The reader is left to explore by themselves what each technique does.

While the definition of personal data covers all the possible types of data that can relate to an individual, in practice not all types of personal data are equal. The landscape of types of personal data and their definition is not homogeneous. In this section we try to cover some concept and make clear distinctions. For learners who wants to go deeper on this topic, consider reading Jarmul (2023).

- **Personally identifiable information (PII)** these are also called “direct identifiers” or “strong identifiers”. Some example can be an individual’s full name, email address, phone number, social security number, and basically anything that can be used as a *fingerprint* to uniquely re-identify an individual. Biometric data (actual fingerprints, shape of the ear, wave-form of the heart signal, iris scan, voice, gait) is an excellent type of PII that can be used to uniquely identify an individual . It is clear already that different types of PII provide different level of “strength” of re-identification.

- **Quasi identifiers** these are also called “indirect identifiers”, they are types of personal data that, in isolation, might not constitute PII to re-identify an individual with high confidence, however a collection of quasi identifiers from an individual can potentially constitute a fingerprint. A famous paper by Rocher et al. (Rocher, Hendrickx, and De Montjoye 2019) has shown that 99.98% of Americans would be correctly re-identified in any dataset using 15 demographic attributes. The picture below is an intuitive explanation from Zaman, Obimbo, and Dara (2017) on how a large amount of quasi identifier could allow possible linking of different types of data, to re-identify subject identity.

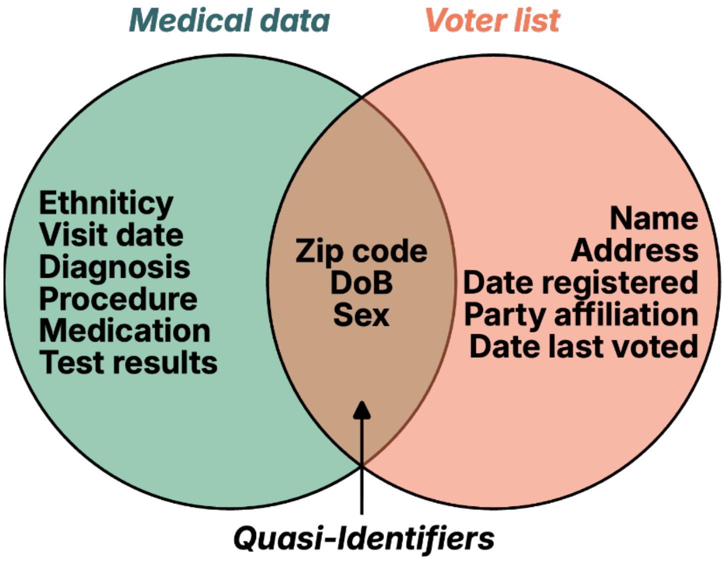


Figure 11.1: ***Quasi identifiers*** *An example from Zaman, Obimbo, and Dara (2017). For a group of individuals we can have two different datasets (e.g. medical records and their voter's information). While the medical record has removed direct identifiers, it is possible to re-identify some individuals using the information from the Voter list.*

- **Pseudonymous data** refers to personal data that has been processed in such a way that it can no longer be attributed to a specific individual without the use of additional information. This additional information is kept separately and is subject to technical and organizational measures to ensure that the data cannot be re-attributed to an identified or identifiable person. In some cases, the additional information is destroyed, but it is possible however to re-obtain it (e.g. by collecting a new fingerprint).

- **Anonymous data** refers to information that has been irreversibly de-identified in such a way that the data subject is no longer identifiable. Once the data is anonymised, it is no longer considered personal data under the GDPR. What is important to understand is that the GDPR adopts the absolute view on anonymisation, i.e. data is anonymous if it impossible to re-identify any individual with the current technological means. This means that the operation of **anonymisation** is an irreversible operation that destroys the data so that no single individual can be re-identified.

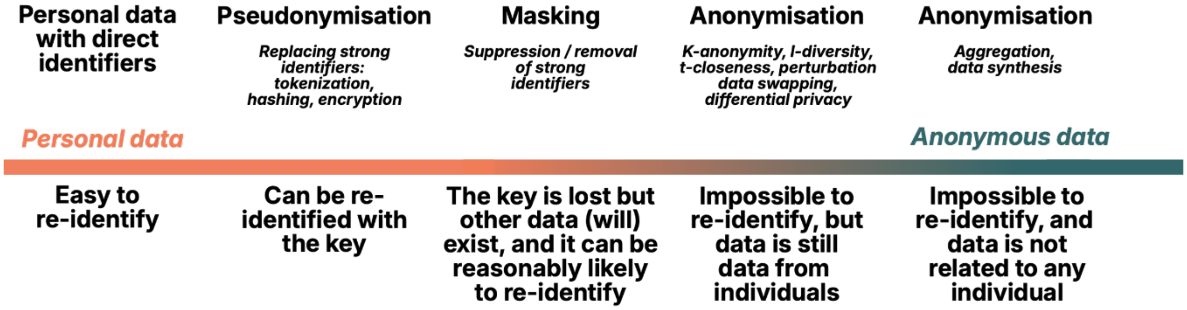

Personal data with direct identifiers
Pseudonymisation
Replacing strong identifiers: tokenization, hashing, encryption
Masking
Suppression / removal of strong identifiers
Anonymisation
K-anonymity, l-diversity, t-closeness, perturbation data swapping, differential privacy
Anonymisation
Aggregation, data synthesis
Personal data
Anonymous data
Easy to re-identify
Can be re-identified with the key
The key is lost but other data (will) exist, and it can be reasonably likely to re-identify
Impossible to re-identify, but data is still data from individuals
Impossible to re-identify, and data is not related to any individual

Figure 11.2: ***From personal to anonymous data*** *A continuum from personal data with direct identifiers to data that does not relate to any specific individual anymore.*

An example of anonymization would be removing all identifying information, such as names, birth dates, or any other combination of quasi-identifiers, from a dataset and applying further techniques like generalization or data masking to prevent re-identification.

Pseudonymization, on the other hand, involves replacing or obscuring personal identifiers with pseudonyms (such as random strings of numbers or letters) to make it more difficult to identify individuals. However, unlike anonymization, pseudonymization does not completely eliminate the risk of re-identification. The original data can often be re-linked to individuals if the pseudonyms are reversible, typically by someone with access to the “key” that can decrypt or reverse the pseudonymization. Therefore, pseudonymized data is still considered personal data under the GDPR, as it remains possible to re-identify individuals with additional information.

In practice, **pseudonymization** is often used to reduce the privacy risks associated with personal data while still allowing for data utility and further processing. It is particularly useful in scenarios where data needs to be shared or analyzed while limiting the exposure of directly identifying information. On the other hand, **anonymization** is applied when the goal is to completely eliminate the possibility of re-identifying individuals, often when sharing data publicly or in open-access scenarios.

The choice between these two approaches depends on the context and the intended use of the data. While pseudonymization provides flexibility in data processing with some privacy protection, anonymization offers a stronger form of data protection by making re-identification impossible, though it may limit the ability to link data back to specific individuals for future analyses.

References

"A New Dawn for Public Employment Services. OECD." 2024. June 12, 2024. https://www.oecd.org/en/publications/2024/06/a-new-dawn-for-public-employment-services_25e1e70e.html.

Agencia Española de Protección de Datos. 2021. "Risk Management and Impact Assessment in the Processing of Personal Data." Agencia Española de Protección de Datos (AEPD). https://www.aepd.es/guides/risk-management-and-impact-assessment-in-processing-personal-data.pdf.

Ahmad, Tazeem, Mohd Adnan, Saima Rafi, Muhammad Azeem Akbar, and Ayesha Anwar. 2024. "MLOps-Enabled Security Strategies for Next-Generation Operational Technologies." In *Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering*, 662–67.

Aliferis, Constantin, and Gyorgy Simon. 2024. "Artificial Intelligence (AI) and Machine Learning (ML) for Healthcare and Health Sciences: The Need for Best Practices Enabling Trust in AI and ML." In *Artificial Intelligence and Machine Learning in Health Care and Medical Sciences: Best Practices and Pitfalls*, edited by Gyorgy J. Simon and Constantin Aliferis, 1–31. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-39355-6_1.

Almada, Marco, and Nicolas Petit. 2023. "The EU AI Act: A Medley of Product Safety and Fundamental Rights?" Rochester, NY. https://doi.org/10.2139/ssrn.4308072.

Andrews, Mel, Andrew Smart, and Abeba Birhane. 2024. "The Reanimation of Pseudoscience in Machine Learning and Its Ethical Repercussions." *Patterns* 5 (9). https://doi.org/10.1016/j.patter.2024.101027.

Barberá, Isabel. 2023a. "AI Possible Risks & Mitigations – Named Entity Recognition." European Data Protection Board. https://www.edpb.europa.eu/system/files/2024-07/ai-risks_d1named-entity-recognition_edpb-spe-programme_en.pdf.

———. 2023b. "AI Possible Risks & Mitigations – Optical Character Recognition." European Data Protection Board. https://www.edpb.europa.eu/system/files/2024-06/ai-risks_d2optical-character-recognition_edpb-spe-programme_en_2.pdf.

Breaux, Travis. 2020. *An Introduction to Privacy for Technology Professionals*. International Association of Privacy Professionals.

Carlini, Nicholas, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, et al. 2021. "Extracting Training Data from Large Language Models." In *30th USENIX Security Symposium (USENIX Security 21)*, 2633–50.

Chang, J Morris, Di Zhuang, G Samaraweera, and G Dumindu Samaraweera. 2023. *Privacy-Preserving Machine Learning*. Simon; Schuster.

Commission, European. 2024. "Regulatory Framework on Artificial Intelligence." https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai.

Desai, Anokhy. 2023. "5 Things to Know about AI Model Cards." https://iapp.org/news/a/5-things-to-know-about-ai-model-cards.

Engelfriet, Arnoud. 2024. *The Annotated AI Act*. Amsterdam, Netherlands: ICTRecht B.V. https://ictrecht.nl.

European Data Protection Board (EDPB). 2024. “Opinion 28/2024 on Certain Data Protection Aspects Related to the Processing of Personal Data in the Context of AI Models.” https://www.edpb.europa.eu/system/files/2024-12/edpb_opinion_202428_ai-models_en.pdf.

European Data Protection Supervisor. 2024. “TechSonar Report 2025.” European Data Protection Supervisor. https://www.edps.europa.eu/system/files/2024-11/24-11-15_techsonar_2025_en.pdf.

Floridi, Luciano. 2023. *The Ethics of Artificial Intelligence: Principles, Challenges, and Opportunities*.

Fujdiak, Radek, Petr Mlynek, Pavel Mrnustik, Maros Barabas, Petr Blazek, Filip Borcik, and Jiri Misurec. 2019. “Managing the Secure Software Development.” In *2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, 1–4. IEEE.

Gal, Michal S, and Orla Lynskey. 2023. “Synthetic Data: Legal Implications of the Data-Generation Revolution.” *Iowa L. Rev.* 109: 1087.

Garrido, Gonzalo Munilla, Johannes Sedlmeir, Ömer Uludağ, Ilias Soto Alaoui, Andre Luckow, and Florian Matthes. 2022. “Revealing the Landscape of Privacy-Enhancing Technologies in the Context of Data Markets for the IoT: A Systematic Literature Review.” *Journal of Network and Computer Applications* 207: 103465.

Gichoya, Judy Wawira, Kaesha Thomas, Leo Anthony Celi, Nabile Safdar, Imon Banerjee, John D Banja, Laleh Seyyed-Kalantari, Hari Trivedi, and Saptarshi Purkayastha. 2023. “AI Pitfalls and What Not to Do: Mitigating Bias in AI.” *The British Journal of Radiology* 96 (1150): 20230023.

Goldblum, Micah, Dimitris Tsipras, Chulin Xie, Xinyun Chen, Avi Schwarzschild, Dawn Song, Aleksander Mądry, Bo Li, and Tom Goldstein. 2022. “Dataset Security for Machine Learning: Data Poisoning, Backdoor Attacks, and Defenses.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45 (2): 1563–80.

Hermanns, Holger, Anne Lauber-Rönsberg, Philip Meinel, Sarah Sterz, and Hanwei Zhang. 2024. “AI Act for the Working Programmer.” *arXiv Preprint arXiv:2408.01449*.

Hinder, Fabian, Valerie Vaquet, and Barbara Hammer. 2024. “One or Two Things We Know about Concept Drift—a Survey on Monitoring in Evolving Environments. Part a: Detecting Concept Drift.” *Frontiers in Artificial Intelligence* 7: 1330257.

Hou, Xinyi, Yanjie Zhao, Shenao Wang, and Haoyu Wang. 2025. “Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions.” *arXiv Preprint arXiv:2503.23278*.

Huyen, Chip. 2022. *Designing Machine Learning Systems*. " O’Reilly Media, Inc.".

International Organization for Standardization. 2022a. *Framework for Artificial Intelligence (AI) Systems Using Machine Learning (ML) (ISO Standard No. 23053:2022)*. International Organization for Standardization. https://www.iso.org/standard/74438.html.

———. 2022b. *Information Technology — Artificial Intelligence — Artificial Intelligence Concepts and Terminology (ISO/IEC Standard No. 22989:2022)*. International Organization for Standardization. https://www.iso.org/standard/74296.html.

———. 2023. *Information Technology — Security Techniques — Guidelines for Privacy Impact Assessment (ISO/IEC Standard No. 29134:2023)*. International Organization for Standardization. https://www.iso.org/standard/86012.html.

Jarmul, Katharine. 2023. *Practical Data Privacy*. " O’Reilly Media, Inc.".

Ji, Jiaming, Tianyi Qiu, Boyuan Chen, Borong Zhang, Hantao Lou, Kaile Wang, Yawen Duan, et al. 2023. “Ai Alignment: A Comprehensive Survey.” *arXiv Preprint arXiv:2310.19852*.

Kaplan, Guy, Uri Katz, and Avi Lumelsky. 2024. “ShellTorch Explained: Multiple Vulnerabilities in PyTorch Model Server (TorchServe) (CVSS 9.9, CVSS 9.8) Walkthrough.” https://www.oligo.security/blog/shelltorch-explained-multiple-vulnerabilities-in-pytorch-model-server.

Kore, Ali, Elyar Abbasi Bavil, Vallijah Subasri, Moustafa Abdalla, Benjamin Fine, Elham Dolatabadi, and Mohamed Abdalla. 2024. “Empirical Data Drift Detection Experiments on Real-World Medical Imaging Data.” *Nature Communications* 15 (1): 1887.

Kreuzberger, Dominik, Niklas Kühl, and Sebastian Hirschl. 2023. “Machine Learning Operations (MLOps): Overview, Definition, and Architecture.” *IEEE Access* 11: 31866–79. https://doi.org/10.1109/ACCESS.2023.3262138.

Lee, Hao-Ping (Hank), Yu-Ju Yang, Thomas Serban Von Davier, Jodi Forlizzi, and Sauvik Das. 2024. “Deepfakes, Phrenology, Surveillance, and More! A Taxonomy of AI Privacy Risks.” In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. CHI ’24. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3613904.3642116.

Leslie, David, Carina Rincón, Michael Briggs, Anna Perini, Sameer Jayadeva, Adriana Borda, Sarah Jane Bennett, et al. 2024. *AI Explainability in Practice*. The Alan Turing Institute.

MacCoun, Robert, and Saul Perlmutter. 2015. “Blind Analysis: Hide Results to Seek the Truth.” *Nature* 526 (7572): 187–89.

Mitchell, Margaret, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. 2019. “Model Cards for Model Reporting.” In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 220–29.

Morris, Kief. 2025. *Infrastructure as Code*. " O’Reilly Media, Inc.".

Near, Joseph P., and Chiké Abuah. 2021. *Programming Differential Privacy*. Vol. 1. https://programming-dp.com/.

OECD. 2024. “Explanatory Memorandum on the Updated OECD Definition of an AI System.” 8. Paris: OECD Publishing. https://doi.org/10.1787/623da898-en.

OWASP. 2024. “AI Exchange.” 2024. https://owaspai.org/.

OWASP AI Exchange Community. 2025. “OWASP AI Exchange.” https://owaspai.org.

Ramasesh, Vinay Venkatesh, Aitor Lewkowycz, and Ethan Dyer. 2022. “Effect of Scale on Catastrophic Forgetting in Neural Networks.” In *International Conference on Learning Representations*.

Raschka, Sebastian. 2024. *Machine Learning q and AI: 30 Essential Questions and Answers on Machine Learning and AI*. No Starch Press.

Rocher, Luc, Julien M Hendrickx, and Yves-Alexandre De Montjoye. 2019. “Estimating the Success of Re-Identifications in Incomplete Datasets Using Generative Models.” *Nature Communications* 10 (1): 3069.

Schwartz, Roy, Jesse Dodge, Noah A Smith, and Oren Etzioni. 2020. “Green Ai.” *Communications of the ACM* 63 (12): 54–63.

Shumailov, Ilia, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. 2024. “AI Models Collapse When Trained on Recursively Generated Data.” *Nature* 631 (8022): 755–59.

Slattery, Peter, Alexander K Saeri, Emily AC Grundy, Jess Graham, Michael Noetel, Risto Uuk, James Dao, Soroush Pour, Stephen Casper, and Neil Thompson. 2024. “The AI Risk Repository: A Comprehensive Meta-Review, Database, and Taxonomy of Risks from Artificial Intelligence.” *arXiv Preprint arXiv:2408.12622*.

Solove, Daniel J. 2005. “A Taxonomy of Privacy.” *U. Pa. L. Rev.* 154: 477.

Solove, Daniel J., and Woodrow Hartzog. 2025. “The Great Scrape: The Clash Between Scraping and Privacy.” *California Law Review*. https://doi.org/10.2139/ssrn.4884485.

Stark, Luke, and Jevan Hutson. 2021. “Physiognomic Artificial Intelligence.” *Fordham Intell. Prop. Media & Ent. LJ* 32: 922.

Tanaka, Fabio Henrique Kiyoiti Dos Santos, and Claus Aranha. 2019. “Data Augmentation Using GANs.” *arXiv Preprint arXiv:1904.09135*.

The Hamburg Commissioner for Data Protection and Freedom of Information. 2024. “Discussion Paper: Large Language Models and Personal Data.” https://datenschutzhamburg.de/fileadmin/user_upload/HmbBfDI/Datenschutz/Informationen/240715_Discussion_Paper_Hamburg_DPA_KI_Models.pdf.

Tsamados, Andreas, Nikita Aggarwal, Josh Cowls, Jessica Morley, Huw Roberts, Mariarosaria Taddeo, and Luciano Floridi. 2021. “The Ethics of Algorithms: Key Problems and Solutions.” In *Ethics, Governance, and Policies in Artificial Intelligence*, edited by Luciano Floridi, 97–123. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-81907-1_8.

Vallet, Félicien. 2022. “Petite Taxonomie Des Attaques Des Systèmes d’IA, Translated as "Small Taxonomy of Attacks on AI Systems".” https://linc.cnil.fr/sites/linc/files/atoms/files/linc_cnil_dossier-securite-systemes-ia.pdf.

VanHoudnos, Nathan, Carol Smith, Matthew Churilla, Shing-Hon Lau, Lauren McIlvenny, and Greg Touhill. 2024. “Counter AI: What Is It and What Can You Do about It?”

Véliz, Carissa. 2020. “Data Privacy and the Individual.” https://philpapers.org/archive/VLIPM.pdf.

Wang, Zhibo, Jingjing Ma, Xue Wang, Jiahui Hu, Zhan Qin, and Kui Ren. 2022. “Threats to Training: A Survey of Poisoning Attacks and Defenses on Machine Learning Systems.” *ACM Computing Surveys* 55 (7): 1–36.

Wehrli, Samuel, Corinna Hertweck, Mohammadreza Amirian, Stefan Glüge, and Thilo Stadelmann. 2022. “Bias, Awareness, and Ignorance in Deep-Learning-Based Face Recognition.” *AI and Ethics* 2 (3): 509–22.

Wu, Tong, Ashwinee Panda, Jiachen T Wang, and Prateek Mittal. 2023. “Privacy-Preserving in-Context Learning for Large Language Models.” *arXiv Preprint arXiv:2305.01639*.

Zaman, ANK, Charlie Obimbo, and Rozita A Dara. 2017. “An Improved Differential Privacy Algorithm to Protect Re-Identification of Data.” In *2017 IEEE Canada International Humanitarian Technology Conference (IHTC)*, 133–38. IEEE.

Zeng, Shenglai, Jiankun Zhang, Pengfei He, Yue Xing, Yiding Liu, Han Xu, Jie Ren, et al. 2024. “The Good and the Bad: Exploring Privacy Issues in Retrieval-Augmented Generation (Rag).” *arXiv Preprint arXiv:2402.16893*.

Zhang, Haibo, Toru Nakamura, Takamasa Isohara, and Kouichi Sakurai. 2023. “A Review on Machine Unlearning.” *SN Computer Science* 4 (4): 337.

Zhang, Xinrui, and Jason Jaskolka. 2022. “Conceptualizing the Secure Machine Learning Operations (SecMLOps) Paradigm.” In *2022 IEEE 22nd International Conference on Software Quality, Reliability and Security (QRS)*, 127–38. https://doi.org/10.1109/QRS57517.2022.00023.

EDPB

European
Data Protection
Board