IMMUTA | FUTURE OF PRIVACY FORUM

WHITE PAPER

Data Protection by Process

How to Operationalize Data Protection by Design for Machine Learning

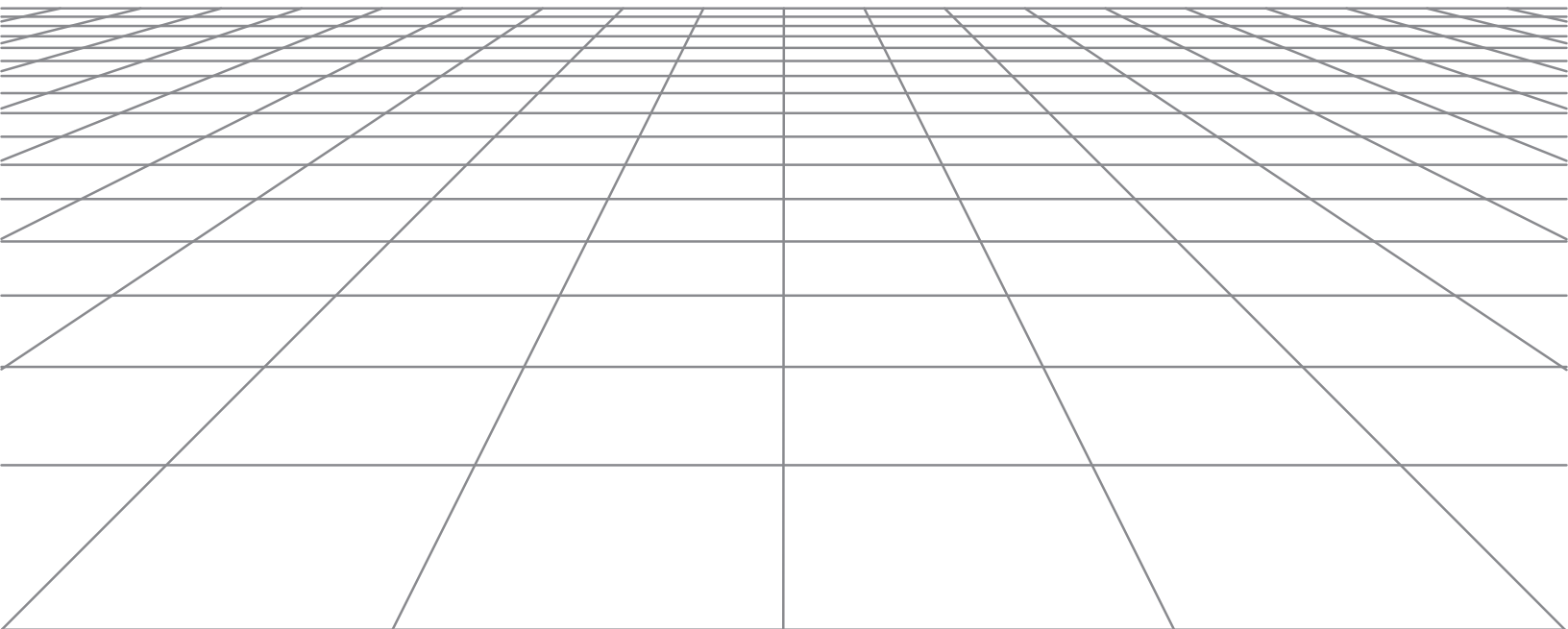

Table of Contents

Introduction 3

What is Data Protection by Design? 4

The DPbD Workflow 7

- Table 1. The Seven (+1) Data Protection Principles 7
- Table 2. Generating a Workflow from the Data Protection Principles 9

What are GDPR controls? 10

What are failure modes? 13

How to Select the Right Controls at the Right Time 15

- Table 3. Failure Modes for Data Minimization 15
- Table 4. Failure Modes for the Principle of Confidentiality 17
- Table 5. Failure Modes for the Principle of Integrity 18
- Table 6. Failure Modes for the Principle of Fairness 19
- Table 7. Examples of Fairness Definitions 20

Appendix 21

Introduction

As the adoption of machine learning (ML) increases, it is becoming clear that ML is shifting the paradigm of model creation. *(By model, we mean systems that aim to support or predict the outcomes of decisions.)* This is because ML relies on the input of vast amounts of data to identify patterns and trends based on correlations between data points.

This fact is often brought up in public discourse to suggest that ML is incompatible with data protection law, which is underpinned by principles like data minimization and purpose limitation (criticized, as a consequence, for being "outdated").

Data Protection by Design (DPbD), a core data protection requirement introduced in Article 25 of The General Data Protection Regulation (GDPR),[1] however, insists that these principles and other data protection imperatives be integrated into any processing of personal data from the design stage (or at the very least when the processing means are being selected) and throughout the whole processing itself.

While several design patterns and strategies have been proposed over time,[2] **there is yet no widely accepted set of best practices for implementing DPbD when building ML models.** This whitepaper shows that in fact ML and data protection requirements, including principles like data minimization, are compatible. It thus clears the path towards effective implementation of DPbD by offering data scientists a set of best practices.

The framework suggested to operationalize DPbD in the context of ML comprises three key stages: setting forth a DPbD workflow, identifying failure modes for the whole ML model lifecycle and selecting controls for each failure mode. It is therefore risk-based and fully consistent with the GDPR approach, including Article 25, and the requirement that all controls be effective.

Best practices specifically tailored to ML are critical to the future automation of decision-making. Without them, organizations will not be able to convince individuals of the benefits of automation, even if humans are kept in the loop.

Building upon the approach initially developed by the German Federal and State Commissioners,[3] this whitepaper is designed to raise awareness of emerging best practices for detecting ML model failure modes and embedding data protection principles within ML model life cycles as early as possible, thereby contributing to the safe and responsible use of ML models. It is a follow-up to our previous whitepaper, "Warning Signs - The Future of Privacy and Security in the Age of Machine Learning."[4]

The whitepaper is organized in five sections. We start the analysis by unpacking the requirement of DPbD and then extract the DPbD workflow. We then explain what controls and failure modes are and how they relate to the DPbD workflow. Finally, we show how to select controls to address key nodes of the DPbD workflow.

1 Article 25 GDPR builds upon Recital 46 of Directive 95/46 of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data, OJ L 281, 23.11.1995, pp. 31–50; the recommendations of the Article 29 Working Party as laid out in 'The Future of Privacy. Joint contribution to the Consultation of the European Commission on the legal framework for the fundamental right to protection of personal data' (2009) 02356/09/EN, WP168; the Privacy-by-Design approach developed in the 1990's by Dr. Ann Cavoukian in her seven foundational principles available at https://www.ryerson.ca/content/dam/pbdce/seven-foundational-principles/The-7-Foundational-Principles.pdf.

2 See Enisa, Privacy and Data Protection by Design, (2012), available at https://www.enisa.europa.eu/publications/privacy-and-data-protection-by-design referring in particular to the work of Seda Gürses, Carmela Troncoso, and Claudia Díaz, Engineering Privacy by Design, Presented at the Computers, Privacy & Data Protection conference, January 2011; Marit Hansen, Top 10 mistakes in system design from a privacy perspective and privacy protection goals, Privacy and Identity for Life, volume 375 of IFIP AICT, IFIP Inter- national Federation for Information Processing, Springer, (2012), pp. 14–31; Jaap-Henk Hoepman, Privacy design strategies – (extended abstract), ICT Systems Security and Privacy Protection - 29th IFIP TC 11 International Conference, SEC 2014, Marrakech, Morocco, Proceedings, (June 2-4, 2014), pp. 446–459.

3 The German Federal and State Commissioners, The Standard Data Protection Model, v.1.0.1, (2016), p. 24, available at https://www.datenschutzzentrum.de/uploads/sdm/SDM-Methodology_V1.0.pdf (SDM).

4 Sophie Stalla-Bourdillon, Brenda Long, Patrick Hall, and Andrew Burt, Warning Signs - The Future of Privacy and Security in the Age of Machine Learning, Future of Privacy Forum and Immuta Whitepaper, (2019), available at https://www.immuta.com/warning-signs-the-future-of-privacy-and-security-in-the-age-of-machine-learning/.

What is Data Protection by Design?

The European Data Protection Board explains that the requirement in Article 25 GDPR "is for controllers[5] to have data protection designed into and as a default setting in the processing of personal data."[6] The UK Supervisory Authority (the Information Commissioner's Office https://ico.org.uk/) expresses the DPbD requirement in these terms: "you have to integrate or 'bake' data protection into your processing activities and business practices, from the design stage right through the lifecycle."[7]

Lawyers have been debating the exact content and effects of the legal obligations provided by GDPR Article 25,[8] which is usually criticized for its lack of specificity and reach.[9] For example, one criticism is that not all system designers are subject to this requirement.[10] However, there seems to be consensus that the starting point of a DPbD strategy are the seven data protection principles listed in Article 5.

We argue that DPbD is the backbone of the GDPR, as complying with Article 25 should lead to complying with the data protection principles, as detailed by Article 5, and to enable all data subject rights, as listed in Articles 12-22.

5 The entities that establish the means and purposes of a processing operation (see Article 4 GDPR).

6 European Data Protection Board, Guidelines 4/2019 on Article 25 Data Protection by Design and by Default, at p. 5, adopted 13 November 2019, available at https://edpb.europa.eu/sites/edpb/files/consultation/edpb_guidelines_201904_dataprotection_by_design_and_by_default.pdf

7 ICO, Guide to General Data Protection Regulation, available at https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/accountability-and-governance/data-protection-by-design-and-default/. See also Reuben Binns and Valeria Gallo, An overview of the Auditing Framework for Artificial Intelligence and its core components, ICO Blog, (26 March 2019), available at https://ico.org.uk/about-the-ico/news-and-events/news-and-blogs/2019/03/an-overview-of-the-auditing-framework-for-artificial-intelligence-and-its-core-component, which lists DPbD as a core component of the ICO Auditing Framework for Artificial Intelligence.

8 For a recent discussion see e.g. Lina Jasmontaite, Irene Kamara, Gabriela Zanfir–Fortuna, and Stefano Leucci, Data Protection by Design and by Default: Framing Guiding Principles into Legal Obligations in the GDPR, European Data Protection Law Review 4(2), (2018), pp. 168 – 189.

9 See e.g. Lee Bygrave, Data Protection by design and by default: Deciphering the EU's legislative requirements, Oslo Law Review 4(2), (2017), pp.105–120.

10 The obligations under Article 25 only apply to controllers (entities that establish the purposes and means of a processing operation), and not to processors (entities processing personal data on behalf of controllers, like service providers).

The DPbD requirement consists of five components:[11]

01 A positive obligation for the controller

The controller must be proactive and, in particular, implement both technical and organizational measures.

02 A broad compliance goal

These measures shall aim to meet all GDPR requirements — in particular the data protection principles and the rights of the data subjects.

03 Effective measures

These measures should be implemented to effectively achieve this goal.

04 A risk-based approach

In order to select appropriate technical and organizational measures, the controller must consider the state of technology; the cost of implementation; the nature, scope and context of processing; the purposes of processing; and the various risks posed to the rights and freedoms of the data subjects.

05 Timeliness and continuity

The measures need to be put in place prior to the processing, when the means of processing are selected, and while the processing is being conducted.

Building a DPbD strategy thus requires closely sticking to the data protection principles (lawfulness, fairness and transparency; purpose limitation; data minimization; accuracy; storage limitation; integrity and confidentiality; accountability) and determining controls for each.

11 For a detailed analysis of each of them, see Lina Jasmontaite et al, fn8.

It also requires protecting the rights of the data subjects, which are prerogatives guaranteed by the law to enable data subjects to intervene in how their personal data are collected and used (through access, correction, deletion, portability, or objection). By reorganizing the data protection principles and including a principle of intervenability,[12] it is possible to make the main nodes of a DPbD workflow emerge. Then, by integrating controls within the DPbD workflow and distinguishing the key steps of ML model lifecycles, it is possible to create a roadmap for DPbD adapted to the development and deployment of ML models.

Importantly, selecting appropriate controls implies conducting a risk assessment and identifying as many failure modes as possible. Put simply, a failure mode is one possible way a system can fail.[13]

While data protection impact assessment methodologies are progressively going beyond the traditional triad of "data alteration, unavailability and unauthorized access," they are not specifically tailored to ML models. What is more, they are difficult to work with when building ML models as they rely upon a list of static data protection requirements, which are not usually meant to be tuned over time. Yet, given how ML models evolve over time, it is crucial that any assessment of their parameters is not static.

> Engineering data protection principles when building ML models requires tuning the intensity or strength of the protection over time until the moment the model is actually used. This is what we call "Data Protection by Process."

And even when the training phase is over, it is likely that some controls will have to be maintained to protect model outputs and the model itself. This approach is fully compatible with EDPB's Guidelines on DPbD, which emphasize that "to ensure effective data protection at the time of processing, the controller must regularly review the effectiveness of the chosen measures and safeguards."[14]

Accurate ML models require sufficient training data to be built. However, acquiring sufficient data is not necessarily contradictory to the requirement of processing a minimum amount of data, as we will explain below. In addition, sufficient data at the beginning of the training should not mean the same thing at the end of the training. Therefore, selecting the right control at the right time to meet each data protection principle as early as possible requires the careful design of a process branching into a variety of controls that should be triggered at different points in time. We explore this process and these controls in further detail below.

12 See SDM, fn 3.

13 Failure Modes and Effects Analysis (FMEA) is a methodology for analyzing causes of failures and understanding their frequency and impact. See e.g. Erik Fadlovich, Performing Failure Mode and Effect Analysis, Embedded Technology, (December 31, 2007), available at https://web.archive.org/web/20111117172649/http://www.embeddedtechmag.com/component/content/article/6134

14 See EDPB Guidelines, fn 6.

The DPbD Workflow

To begin, it is important to recall the substance of each data protection principle, as listed in GDPR Article 5. An eighth principle, Intervenability, has been added to capture the requirement that DPbD entails the integration of safeguards to protect the rights of data subjects.[15] (The data protection principles are defined in Table 1 below.)

TABLE 1

The Seven (+1) Data Protection Principles

DATA PROTECTION PRINCIPLES	DEFINITIONS
Lawfulness, Fairness, and Transparency	*"Personal data shall be processed lawfully, fairly and in a transparent manner."*[16] In order to be processed lawfully, at least one legal basis should be identified within a list of six legal bases (Article 6). If the data is sensitive data within the meaning of Article 9, another justification should be added to be found in a list of 10 additional legal bases.
Purpose Limitation	*"The data shall be collected for specified, explicit and legitimate purposes and not further processed in a manner that is incompatible with those purposes; further processing for archiving purposes in the public interest, scientific or historical research purposes or statistical purposes shall, in accordance with Article 89(1), not be considered to be incompatible with the initial purposes."*[17]
Data Minimization	*"The data shall be adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed."*[18]
Accuracy	*"The data shall be accurate and, where necessary, kept up to date; every reasonable step must be taken to ensure that personal data that are inaccurate, having regard to the purposes for which they are processed, are erased or rectified without delay."*[19]

15 SDM, fn3.

16 GDPR, Article 5(1)(a).

17 GDPR, Article 5(1)(b).

18 GDPR, Article 5(1)(c).

19 GDPR, Article 5(1)(d).

Storage Limitation	*"The data shall be kept in a form which permits identification of data subjects for no longer than is necessary for the purposes for which the personal data are processed; personal data may be stored for longer periods insofar as the personal data will be processed solely for archiving purposes in the public interest, scientific or historical research purposes or statistical purposes in accordance with Article 89(1) subject to implementation of the appropriate technical and organisational measures required by this Regulation in order to safeguard the rights and freedoms of the data subject."* [20] In order to be processed lawfully, at least one legal basis should be identified within a list of six legal bases (Article 6). If the data is sensitive data within the meaning of Article 9, another justification should be added to be found in a list of 10 additional legal bases.
Integrity and Confidentiality	*"The data shall be processed in a manner that ensures appropriate security of the personal data, including protection against unauthorised or unlawful processing and against accidental loss, destruction or damage, using appropriate technical or organisational measures."* [21] The principle of integrity and confidentiality should be coupled with Article 32, which contains a richer list of system requirements: confidentiality, integrity, availability and resilience of processing systems and services. In addition, GDPR Article 32 provides for the monitoring and testing of processing activities, while GDPR Articles 33-24 impose upon data controllers personal data breach notification obligations.
Accountability	*"The controller shall be responsible for and be able to demonstrate compliance with all data protection principles."* [22] GDPR Article 30 imposes recording obligations and Article 35 provides for the performance of data protection impact assessment in situations of high risks for the rights and freedoms of data subjects.
Intervenability	*"The data subject's rights to intervene are explicitly derived from the provisions on rectification, blocking, erasure, and the right of objection (Articles 16-17 GDPR). They may also result from a weighting of interests within the framework of statutory criteria for lawful processing. Once again, the controller must, pursuant to Article 5(1)(d) GDPR provide the prerequisite for guaranteeing such rights, both at organisational and, where required, at technical level."* [23]

20 GDPR, Article 5(1)(e).

21 GDPR, Article 5(1)(f).

22 GDPR, Article 5(2).

23 SDM, fn3, p. 24.

Reorganizing the data protection principles described in Table 1, it is possible to make the main nodes of a DPbD workflow emerge, as illustrated in Table 2:

TABLE 2

Generating a Workflow from the Data Protection Principles

DATA PROTECTION PRINCIPLES	DPBD WORKFLOW NODES
1. Purpose Limitation	Express purpose when defining model objectives, assumptions and limitations.
2. Lawfulness	Identify legal basis.
3. Data Minimization	Calibrate amount of data to purpose and training phases.
4. Accuracy	Check data accuracy.
5. Fairness	Assess fairness considering model assumptions, limitations, quality of training data, and impact of decision-making pipeline upon data subjects.
6. Storage Limitation	Express data retention period for the processing of training data and set timeframe for project.
7. Integrity and Confidentiality	Ensure integrity and availability, and prevent unauthorized disclosure of training data/model/model outputs.
8. Transparency	Make processing activities transparent and translate description of processing activities into meaningful language for data subjects.
9. Intervenability	Build capabilities to interact with data subjects (e.g., interface).
10. Accountability	Monitor and audit data usage from acquisition of training data to model usage.

The above workflow reflects mutual good practice for data management and should therefore be top of mind for all data governance specialists.

What are GDPR controls?

Organizational and technical measures aimed at implementing data protection principles as early as possible can be directive, detective, preventive or corrective.[24]

Directive Controls

Ensuring that data scientists are aware of risks and obligations inherent in data processing, that they are well-supervised by competent managers, and that processes are well-documented and understood are directive controls, which are essential to building ML models. While this might be obvious to some, it should be recalled each time a new data science project is initiated. These controls should be repeated for all failure modes.

Detective Controls

The monitoring of how data and model output is being accessed are as important as directive controls. This is because as each data protection principle requires the implementation of a set of controls that should be triggered at different points in time, full oversight of the process is needed. These controls should also be repeated for all types of failure modes. Furthermore, interpretability methods[25] to better understand how ML models are performing should be seen as detective controls.

24 See e.g. Phil Kenkel, Types of internal controls, (2013), available at http://agecon.okstate.edu/coops/files/types_of_internal_controls.docx for anecdotal examples.

25 See e.g. Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi, A Survey of Methods for Explaining Black Box Models, ACM Computing Surveys (CSUR), 51(5), (2018), p. 93, available at https://arxiv.org/pdf/1802.01933.pdf.; W. James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu, Interpretable machine learning: definitions, methods, and applications, arXiv:1901.0459, (2019), available at https://arxiv.org/abs/1901.04592.

Preventive Controls

Preventive controls are in principle the most powerful controls, as they are self-executable. Given the complexity of ML models and the training phase, the initial assumption is often that ML makes preventive controls impossible or useless. The most obvious preventive control, however, is access control, which in the case of ML models, is relevant both for input and output data as well as for the model itself. Additionally, solutions such as data virtualization,[26] which makes read-only access to training data possible, are crucial for ensuring data integrity over time. Randomization methods, such as differential privacy,[27]decentralized architectures or intermediaries, such as federated learning,[28]can also act as key preventive controls.

Although preventive controls are useful they are only one tool within the toolbox, as they do not address all data protection principles. In addition, they do not always offer formal guarantees. For example, a model produced via federated learning or secure multi-party computation is still vulnerable to model inversion attacks.[29] Moreover, preventive controls could require a series of steps to be fully effective. In some instances, differential privacy will require fine-tuning to reach an acceptable level of privacy loss, as explained in detail below.

26 "Data virtualization is an umbrella term used to describe an approach to data management that allows an application to retrieve and manipulate data without requiring technical details about the data, such as how the data is formatted or where it is physically located. The goal of data virtualization is to create a single representation of data from multiple, disparate sources without having to copy or move the data." Margaret Rouse, What is Data Virtualization?, TechTarget.com, (last updated 2019), available at https://searchdatamanagement.techtarget.com/definition/data-virtualization.

27 "Differential Privacy" describes a promise, made by a data holder, or curator, to a data subject: "You will not be affected, adversely or otherwise, by allowing your data to be used in any study or analysis, no matter what other studies, data sets, or information sources, are available." Cynthia Dwork & Aaron Roth, The Algorithmic Foundations of Differential Privacy, Foundations and Trends in Theoretical Computer Science 9(3–4), (2014), pp. 211–407, available at https://www.cis.upenn.edu/~aaroth/Papers/privacybook.pdf.

28 See Y. Liu, T. Chen, and Q. Yang, Secure federated transfer learning,"CoRR, vol. abs/1812.03337, (2018), available at http://arxiv.org/abs/1812.0333, where "Federated Transfer Learning (FTL) was introduced to improve statistical models under a data federation that allow knowledge to be shared without compromising user privacy, and enable complementary knowledge to be transferred in the network" as explained by Shreya Sharma, Xing Chaoping, Yang Liu, and Yan Kang, Secure and Efficient Federated Transfer Learning, arXiv:1910.13271, (2019), p. 1, available at https://arxiv.org/abs/1910.13271.

29 "Secure Multi-Party Computation (SMPC) is an important subset of cryptography. It has the potential to enable real data privacy. SMPC seeks to find ways for parties to jointly compute a function using their inputs, while keeping these inputs private." Shaan Ray, What is Secure Multi Party Computation?, Hackernoon Blog, (11 June 2019), available at https://hackernoon.com/what-is-secure-multi-party-computation-232caef900b9. See also I. Damgård, V. Pastro , N. Smart, and S. Zakarias, Multiparty Computation from Somewhat Homomorphic Encryption, in R. Safavi-Naini , R. Canetti (eds) Advances in Cryptology – CRYPTO 2012, Lecture Notes in Computer Science, vol 7417, Springer, Berlin, Heidelberg, (2012).

Corrective controls

Corrective controls should not be neglected since other types of controls are unlikely to appropriately mitigate all failure modes. Breach mitigation strategies are, therefore, a must-have and should include complaint mechanisms. In particular, the potential of corrective controls for ensuring fair decisions should be further explored. It is often assumed that controls set to avoid unfair algorithms[30] are more effective than corrective controls set after the design stage, such as decisions ensuring that algorithms are not used in ways that disadvantage those at high risk. Yet, debiasing algorithms can in fact have perverse effects.[31]

This fourfold distinction of controls is particularly useful to distinguish key steps within a workflow. More fundamentally, all types of controls fall into two high-level categories: process controls (identifying who should do what and how) and system controls (self-executing decisions, such as Privacy Enhancing Technologies[32] or PETs). System controls should always be complemented by process controls.

30 Such as anti-classification, outcome and error parity and equal calibration. For an insightful overview of bias and AI systems see Reuben Binns and Valeria Gallo, Human bias and discrimination in AI systems, ICO Blog, (25 June 2019), available at https://ico.org.uk/about-the-ico/news-and-events/ai-blog-human-bias-and-discrimination-in-ai-systems/.

31 See e.g. Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel, Fairness through awareness, in Proc. ACM ITCS, (2012), pp. 214–226, available at https://arxiv.org/pdf/1104.3913.pdf; Moritz Hardt, Eric Price, and Nati Srebro, Equality of Opportunity in Supervised Learning, NIPS, (2016), available at http://papers.nips.cc/paper/6374-equality-of-opportunity-in-supervised-learning.pdf.

32 For a recent overview of these technologies see Royal Society, Protecting Privacy in Practice, The current use, development and limits of Privacy Enhancing Technologies in data analysis, (March 2019), available at https://royalsociety.org/topics-policy/projects/privacy-enhancing-technologies/.

What are failure modes?

Identifying failure modes is crucial to deriving a comprehensive list of controls. Notably, guidance on this point is still in its infancy.

The European Data Protection Board (EDPB) and national supervisory authorities, for example, only identify inherently risky activities or hazards.33 These activities are not illegal per se but would require the implementation of appropriate safeguards to reduce the level of risk before the beginning of the processing. Within the EDPB's list of criteria to take into account when deciding what processing leads to high risks and warrants the execution of a DPIA (Data Protection Impact Assessment), one can find "[i]nnovative use or applying new technological or organizational solutions" and "matching or combining data sets."[34] As the EDPB explains, if a processing meets any two criteria on the list, a DPIA is likely to be necessary. These two inclusions taken together seem to suggest that building ML models should systematically be considered a hazard. The key question is therefore what safeguards or controls to put in place to reduce the level of residual risk to an acceptable level.

Failure modes are the mechanisms or processes through which a hazard materializes into an adverse outcome or harm. Of note, harm under the GDPR can be both material and nonmaterial, according to Recital 75. And as we have explained in our previous whitepaper,[35] the development, deployment and usage of ML models can generate three types of harm: informational (generated by the unintended or unanticipated leakage of training data), behavioral (generated by manipulating the behavior of the model) and collective harm (related to the harm felt by individuals whose training data has not necessarily been used to train the model, but who are nonetheless impacted by the predictions of the model). Failure modes can be grouped by the type of harm they actually generate to make sure a complete picture of harm is being drawn.

One reason why data protection principles are so critical is that they act as hints as to the potential relevant failure modes a particular hazard could lead to. Let's take the example of the hazard "matching or combining data sets." Potential failure modes would include processing inaccurate data, processing too much data or engaging in surveillance, inferring sensitive attributes, retaining the data for too long, unauthorized access, and failure to enable data subjects to exercise their rights.

ML model lifecycles can be divided into five stages: 1) define project objectives, 2) acquire and explore data, 3) develop and test the model, 4) deploy and 5) use the model. For each stage a list of failure modes should be established and each data protection principle should be addressed by at least one failure mode if not more.[36] Controls should then be mapped against each failure mode and should include directive, detective, preventive and corrective measures. Once this is done, it becomes possible to quantify the residual risks generated by these failure modes,

33 See for example EDPB, Guidelines on data protection impact assessment (DPIA) and determining whether processing is "likely to result in a high risk" for the purposes of Regulation 2016/679, wp248rev.01, (2017), available at https://ec.europa.eu/newsroom/article29/item-detail.cfm?item_id=611236.

34 EDPB, Guidelines on DPIA, fn33, p. 11.

35 Warning Signs, fn4..

36 An indicative list of failure modes is included in the *Appendix* to this document.

which will require reestablishing the likelihood of these events, reassessing their detectability and reidentifying the severity of the harm once controls are in place.[37] Controls should then be mapped against each failure mode and should include directive, detective, preventive and corrective measures. Once this is done, it becomes possible to quantify the residual risks generated by these failure modes, which will require reestablishing the likelihood of these events, reassessing their detectability and reidentifying the severity of the harm once controls are in place.

It is important to remember that data protection failure modes are not always attacks against the training data, model outputs or even models themselves. In fact failure modes can result from any type of bad or inappropriate data protection practice. Bias in training data,[38] which can be replicated into the model, could lead to discriminatory model outputs. Overfitting[39] could mean that the algorithm is in fact memorising the training data even if such an outcome was not intended.

37 The first version of CNIL's methodology on privacy impact assessment is particularly interesting here in that harm is said to be a factor of the degree of identifiability and severity of impact. CNIL, Methodology for Privacy Impact Assessment, (2012), available at https://www.cnil.fr/sites/default/files/typo/document/CNIL-ManagingPrivacyRisks-Methodology.pdf.

38 An example might be when a data set of successful candidates mainly contain male candidates.

39 An example might be the algorithm learning that 'the home at 123 Allcroft Rd costs £2,000,000' instead of 'homes in the Camden area are typically £950,000' — the latter can be extrapolated to other homes, while the former is an example of 'over-fit.'

How to Select the Right Controls at the Right Time

With sufficient data recorded from a stable population, the introduction of new training data results in diminishing improvements.

This essentially means that while the value of the data to the data subject remains constant over time, more data will have diminishing value to the model owner.

For example, let's assume that the value of Mike's DNA sequence is fixed to him. A new research initiative, starved for data, may highly value his DNA. In contrast, a long-existing program, rich in data, would not value his DNA as much.

Such an observation has direct implications for **the principle of data minimization**, for which at least three failure modes could be identified, as illustrated in Table 3 below: inclusion of unnecessary attributes, overly large training set size, and use of out-of-date training data. An adverse outcome in the form of informational harm for such a failure mode could be nonmaterial damage in the form of distress due to the fear that an attack could happen and sensitive data could be leaked.

TABLE 3

Failure Modes for Data Minimization

FAILURE MODES	DESCRIPTION	CONTROLS
Inclusion of unnecessary attributes	Training set contains fields which are unrelated to the target variable and may be memorized by the model.	Suppress or remove unrelated attributes or features.[40]
Overly large training set size	Training set size much larger than is required for model convergence.	Reduce the sample size. Note, it may be necessary to use stratified sampling techniques to ensure that minority classes remain well-represented.
Use of out-of-date training data	Training set contains older data that may no longer be as relevant to predicting the target variable.	Suppress or remove old records.

40 Reduction of model features also serves to reduce the risk at query time by reducing or eliminating potentially sensitive attributes from the query.

Addressing the principle of data minimization should entail setting up specific processes. For example, let's assume a data subject has consented for her data to be used for a specific purpose, say, traffic routing. Only certain metadata related to this individual are relevant for traffic routing. It makes little sense to unnecessarily maintain irrelevant metadata, as this data does not increase model performance but increases the organization's liability in the event of breach. Suppressing or removing unnecessary attributes reduces the risk that the record could be linked with other records pertaining to the same individual.

A similar argument can be made for a reduction in the number of records. If enough records are present to accurately capture traffic behavior down to the minute, then additional data is redundant. At this point the marginal benefit of additional data has vanished, yet each makes a non-zero contribution to the organisation's overall liability. In such cases it makes sense to either stop collecting data or, if the data is already present, discard all but a minimal representative sample at the highest necessary resolution.

Additionally, underlying trends can change and make old data no longer meaningful. Older data is certainly less relevant to predicting the current traffic patterns than more recent data is. At that point, older data could easily be excluded from the analysis, both in fulfillment to the data minimization principle and to the betterment of the model.

In a setting where there are multiple objectives (or purposes), a data controller may maintain more data than whatever is necessary to fulfill any single objective. Nevertheless, she should not maintain significantly more data than is necessary to fulfill all objectives. The principle of data minimization should be implemented dynamically for each objective or purpose.

A similar approach can be taken for **the principle of confidentiality**. Potential failure modes for this principle are listed in Table 4 below and include unauthorized access to the training data, to model outputs or the model itself. An adverse outcome could be in the form of both material and nonmaterial damage as a result of identity theft, a form of informational harm. This outcome could happen indirectly when an attacker is able to make inferences about training data through a model inversion attack. A good technique for preventing model inversion attacks is simply keeping unnecessary data out of the training set. First, the data scientist should build a version of the model without differential privacy. (She should not release the model to the public at this stage.) She would note its baseline performance and then throw away the model. She would then iteratively build models with more noise until she reaches a minimum acceptable threshold for performance, or a maximum acceptable threshold for privacy loss. Assuming, then, that the privacy loss is acceptable, she could release the model into production.

TABLE 4

Failure Modes for the Principle of Confidentiality

FAILURE MODES	DESCRIPTION	CONTROLS
Unauthorized access to training data	Training readable to parties who do not possess a purpose to know.	Limit read access to training data.[41]
"Whitebox" inference of training data from examination of model state (e.g., examination of weights)	Model data structures are available to parties who may use them to make inferences about individuals represented in the training data.	1. Limit read access to internal model state. 2. Employ local differential privacy. 3. Employ differential privacy during training.
"Blackbox" inference of training data from query access to the model	Model query access is available to parties who may use them to make inferences about individuals represented in the training data.	1. Enforce access and control limitations around model query access. 2. Employ local differential privacy. 3. Limit the number of queries. 4. Employ differential privacy during training. 5. Employ differential privacy as a post process (if possible).

Regarding the principle of integrity, failure modes could lead to either informational or behavioral harm, even when there is no direct read access to the training data (this is what is happening with poisoning or model evasion). Behavioral harm can become collective (e.g., when training or poisoning a model to intentionally discriminate against a group of people). Failure modes for this principle are listed in Table 5 on the next page.

41 Note that models trained with federated learning enhance privacy by sharing local model updates instead of raw training data, and thus provides a soft mitigation through the avoidance of sharing training data. While federated learning alone provides no formal guarantee that training data cannot be inferred from model updates, such techniques can be hardened with additional protections.

TABLE 5

Failure Modes for the Principle of Integrity

FAILURE MODES	DESCRIPTION	CONTROLS
Unauthorized access to training data	Training data may be inadvertently altered or purposefully corrupted, resulting in disruption of the model training pipeline and degradation of model performance.	Enforce access and control limitations on stored training data.
Poisoning	*"Model poisoning occurs when an adversary is able to insert malicious data into training data in order to alter the behavior of the model at a later point in time."* [42]	Enforce access and control limitations on insertion and modification of model training data at points of acquisition and storage.
Model evasion	*"Evasion occurs when input data is fed into an ML system that intentionally causes the system to misclassify that data."* [43]	AT TRAINING TIME: 1. Mitigate poisoning. 2. Employ techniques during model development and training to penalise known classes of adversarial samples (e.g., gradient masking, GANs where the adversary is a noise source, noisy voting over models trained on separate shards). AT RUN TIME: 1. Enforce access and control limitations around model query access. 2. Patterns of irregular access may indicate attempts to find an adversarial input, warranting use of anomaly detection.

Addressing **the principle of fairness** is more challenging because the list of failure modes ultimately depends upon the definition of fairness initially adopted. Yet, there is no consensus about the substance of that definition. Assuming it should be interpreted to mean that models should not produce decisions that privilege or harm certain individuals or groups to the detriment or benefit of others without a legitimate justification, a failure mode for this principle would be when model output depends on some legally or, more broadly, socially irrelevant characteristics of the data.

An adverse outcome could be both material and nonmaterial damage arising from a loss of opportunity, such as when an algorithm overlooks more qualified candidates for job placement, or when a loan applicant is denied a loan due to reinforcement of historical bias present in the training data.[44] These are great examples of collective harm, which are not behavioral harm because they are not generated through the intentional manipulation of the behavior of the model. There are a number of ways that ML models[45] learn to be socially biased, as illustrated below in Table 6:

42 Warning Signs, fn 4, p.5.

43 Warning Signs, fn 4, p.5.

44 An example would be when the model tends to deny based on zip code, due to its correlation with historical denials on the basis of race.

45 Solon Barocas et al, Big Data's Disparate Impact, California Law Review 104(3), (2016), p. 671. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2477899##.

TABLE 6

Failure Modes for the Principle of Fairness

FAILURE MODES	DESCRIPTION	CONTROLS
Skewed training data	A sample is skewed when class memberships are disproportionately represented. For instance, measuring crime rates by crime reports introduces a dependency on report frequency. An area of high police activity may result in more reports of crime, even though the true crime rate is less than that of areas with fewer reports. A model which decides to send more police to these areas may reinforcing existing non-uniformities in policing due to social biases.	Counteract feedback loops by filtering the training set, by probabilistically adding events at a rate inversely related to its representation.[46]
Tainted training data	Training data is tainted when one or more of its examples have been affected by bias.	Iteratively re-weight data during training to achieve the desired notion of fairness.[47]
Limited Features	Disproportionate feature quality across classes leads to differing model accuracy across classes.	Discard features which are poor predictors for minority groups.
Sample size disparity	Disproportionate representation of class sizes leads to differing model accuracy across classes.	Optimise for a model which distributes the error rates evenly among all protected classes, regardless of size.
Proxies	Proxies are features which are correlated with class membership.	Minimally perturb the input so that class membership cannot be predicted.

As indicated above, model characteristics remain sensitive to how fairness is formalised. Because of the diversity of failure modes and the diversity of fairness definitions as illustrated in Table 6, it is clear that there is no one-size-fits-all solution and that a step-by-step, contextual approach relying upon a variety of controls is required, starting from the formulation of fairness assumptions and acknowledgement of their limitations.

46 Danielle Ensign, Sorelle A. Friedler, Scott Neville, Carlos Scheidegger, and Suresh Venkatasubramanian, Runaway Feedback Loops in Predictive Policing, Proceedings of the 1st Conference on Fairness, Accountability and Transparency, PMLR 81, (2018), pp. 160–171, available at http://proceedings.mlr.press/v81/ensign18a.html.

47 Heinrich Jiang and Ofir Nachum, Identifying and Correcting Label Bias in Machine Learning, CoRR abs/1901.04966, (2019), available at https://arxiv.org/abs/1901.04966.

TABLE 7

Examples of Fairness Definitions

TYPE OF FAIRNESS	DEFINITION
Statistical Parity	There should be little-to-no difference in acceptance rates between those in a protected class and those outside of it.
Conditional Independence	The predicted score, given a particular value of the target variable, should be the same across all groups.
Sufficiency	The target becomes independent of class membership given the score.

These four examples show that **continued monitoring of the whole DPbD workflow** as identified in Table 2 is the key to the success of DPbD strategies applied to ML models building, even if PETs should also be part of the solution. As recalled in the recent report of the Royal Society, "[t]he use of PETs [should] not detract from the need to assess whether it is legal or ethical to carry out an analysis or give access to the data in the first place." [48] And a good DPbD workflow should also mean a better ML model.

This whitepaper aimed to outline a framework to operationalize DPbD for developing and deploying ML models, and we welcome suggestions or comments to improve this framework. Please reach out to governance@immuta.com with feedback.

48 Royal Society, fn32, p. 27.

Appendix

Indicative List of Failure Modes for ML Model Lifecycle

MODEL LIFECYCLE STAGE	POTENTIAL FAILURE MODES
01 **Define project objectives**	Misdefined objective/purpose
	Misdefined basis of processing
	Misdefined assumptions and limitations
	Misdefined risk profile of model (impact)
	Misdefined project team composition, capability and resources
	Failure to create adequate documentation
02 **Acquire and explore data**	Too much/irrelevant input data (see table 3)
	Unaddressed / unmanaged protected category data and proxies (see tables 6 & 7)
	Inaccurate data (see tables 3, 6, & 7)
	Corrupted data (see table 5)
	Poisoned data (see table 5)
	Non-stationary data assumed to be stationary
	Unavailable data
	Misclassified data (e.g. sensitive data)
	Access & control failure for training data (see table 4)
	Failure to enable data subject rights for training data
	Failure to create adequate documentation

03 **Develop & test model**	Flawed assumption
	Maths error
	Inappropriate modelling method
	Unstable model (model generally behaves erratically – small input perturbations drastically change output result; see also evasion in table 5)
	Failure to tailor the amount of training data over time (see table 3)
	Overfitting
	Biased/discriminatory/unfair model (see tables 6 & 7)
	Inadequate testing of edge cases (awareness of edge cases)
	Inadequate cross validation
	Access & control failure for both training data, and model definition (see table 4)
	Poisoning (see table 5)
	Failure to enable data subject rights for training data
	Failure to create adequate documentation (including of model dependencies and assumptions; c.f. Google ModelCard docs)
04 **Deployment**	Undetected error (coding etc)
	Access & control failure for both model definition (see table 4), query input, and/or model output.
	Up & downstream dependency management failure (in model infrastructure)
	Failure to employ version control procedures on model change
	Failure to create adequate documentation of model deployment

05 **Model usage Develop & test model**	Unvalidated model in usage
	Inappropriate model purpose
	Corrupted or unavailable runtime parameters
	Concept drift (data and correlated drift, system change)
	Query data differs significantly in character from training data
	Reliance on low confidence outputs
	Reliance on risky (protected category) outputs
	Failure to detect environmental changes
	Biased/discriminatory/unfair model outputs (see tables 6 & 7)
	Access and control failure for input data, model outputs and model (see table 4)
	Model inversion (see table 4)
	Blackbox duplication of model
	Model evasion (see table 5)
	Failure to enable data subject rights upon input and output data
	Failure to maintain auditable usage logs

IMMUTA FUTURE OF PRIVACY FORUM

25 Thomson Place, 4th Floor, Boston, MA 02210 immuta.com (800) 655-0982