Reliable and fair machine learning for risk assessment

Pereira Barata, A.P.

Chapter 1

Introduction

Modern society is increasingly reliant on information and communication technologies. This includes machine learning methods and their employment in artificial intelligence (AI) systems which are rapidly becoming indispensable components of the status quo. As these technologies evolve, their societal integration shapes the manner in which aspects of education, health, economy, and government are conducted [Ahirwar, 2020].

Given their broad range in application and inherent automated nature, the implementation of these technologies comes with associated risks; e.g., on democracy, the rule of law, and distributive justice, or on the human mind itself in the form on opinion manipulation. To prevent and minimise such risks, there is currently a focus on the foundations, realisation, and assessment of trustworthy AI in the European Union (EU), under which the definition of AI systems is as follows [European Commission, 2019a].

Definition 1.1 – ***AI systems***

AI systems are software (and possibly also hardware) systems designed by humans that, given a complex goal, act in the physical or digital dimension by perceiving their environment through data acquisition, interpreting the collected structured or unstructured data, reasoning on the knowledge, or processing the information, derived from data and deciding the best action(s) to take to achieve the given goal.

Some AI systems can adapt their behaviour after analysing how the environment is affected by their previous actions. The task of achieving *trustworthy AI* and applying it to our society has been set in motion, as official framework guidelines are nowadays arising. In accordance with the High-Level Expert Group on AI [European Commission, 2019b], the definition of trustworthy AI follows.

Definition 1.2 – ***Trustworthy AI***

Trustworthy AI is, on a foundational level, an AI system which abides to the four ethical *principles* of trustworthiness: (1) respect for human autonomy; (2) prevention of harm; (3) fairness; and (4) explainability.

Yet, these principles are meant as broad ideological statements, rather than objective instructions. As such, towards their realisation, technical and non-technical *methods* must be employed. On the one hand, *technical* methods relate to concepts such as model development and model testing. On the other hand, *non-technical* methods entail codes of conduct at an organisational level of entities [CLAIRE, 2021].

Depending on the related risks, AI systems have more or less stringent obligations which must be followed. Specifically, AI systems of which the deployment may put the life and health of citizens at risk are termed *high-risk* [European Commission, 2021]. Towards their utilisation, high-risk AI systems will be subject to stringent obligations, such as the minimisation of discriminatory outcomes, adequate assessment of the performance of the system, whilst having appropriate human oversight.

The present thesis focuses on the *technical methods* towards trustworthy AI in Europe, specifically for high-risk AI systems in light of the risk assessment activities enacted by the Human Environment and Transport Inspectorate of the Netherlands: *Inspectie Leefomgeving en Transport* (hereinafter Inspectorate or ILT). Below, in Section 1.1, we provide a brief introduction to the Inspectorate and the risk assessment activities therein acted, with focus on the issues associated with the shift towards a data-driven paradigm. Concretely, we will narrow down these issues from a machine learning perspective and address them with respect to: *reliability*, in the form of the quality of data; and *fairness* in the form of bias in data. We will do so prior to defining the problem statement and research questions, as to provide the necessary context to the reader.

Section 1.2 provides the preliminaries of machine learning. In Section 1.3, we describe the problems related to the quality of real-world data, viz. *missingness* and *noise*. Section 1.4 discusses the concerns of learning from biased data (i.e., fairness) in machine learning. The 3 aforementioned Sections present both formal definitions and examples of practical applications in the Inspectorate of those definitions, as a way to make explicit the points which are relevant towards formulating, in Section 1.5, the problem statement (PS) and the research questions (RQs). It is remarked that each RQ has its own research methodology which is explained when the RQ is addressed. In Section 1.6, we list our research goals. Lastly, the outline of the remainder of this thesis is given in Section 1.7.

1.1 The Inspectorate

In the Netherlands, the ILT is the legal supervising entity responsible for improving safety, confidence, and sustainability in regard to transport, infrastructure, environment and housing. Practical limitations make it impossible to check the compliance of every single aspect of these broad domains.

Consider, for example, the inspection of ships in the port of Rotterdam, arguably the largest and busiest port in Europe. Every year, over $120,000$ vessels transit the port from around the globe —sea-going vessels— and within the Netherlands —inland vessels— amounting to circa $450,000,000$ metric tons of goods [Port of Rotterdam Authority, 2021]. The ILT must decide how to mobilise their resources, promoting efficacy, efficiency, and feasibility of compliance ensurance.

To ensure compliance, the entities of interest to the ILT (such as companies) are requested to report about their activities to the Inspectorate. This process generates data, often in tabular form. The data are gathered so that domain experts (i.e., inspectors working at the ILT) may analyse them and prioritise their risk assessment activities accordingly.

Given the volume of data gathered by the ILT, it is not possible for the inspectors to consider these data adequately with their current tool set, which is largely comprised of labour-intensive manual analysis of tabular data. If data are not adequately considered —as is the case now— then the proficiency of the risk assessment and inspection activities have high potential for improvement. The opportunity for improving upon these activities in a *data-driven* manner by utilising *machine learning* methods provides the motivation for this thesis.

Risk Assessment

According to [Rausand, 2013], risk assessment is defined as follows.

Definition 1.3 – ***Risk assessment***

> Risk assessment is the joint effort of: (1) recognising and analysing possible future occurrences that could harm people, property, or the environment (i.e., hazard analysis); and (2) judging the acceptability of risk based on analysis and taking influencing factors into account (i.e., risk evaluation).

High risk is often associated with the activities performed by the ILT, such as evaluating infrastructure integrity which may jeopardise the life and health of citizens, as failure to comply may result in dire negative health, safety, and environmental impacts. AI systems which are used in such activities clearly fall under the category of *high-risk* AI [European Commission, 2021].

In risk assessment, adequately selecting a non-compliant entity for inspection is termed *targeting*. Failure to perform targeting is termed *mistargeting* and comes in two forms: (1) a non-compliant entity is not selected; and (2) an entity which is compliant is wrongfully targeted. Although the nature of the noncompliance may be diverse (e.g., ship emissions, waste transportation, and infrastructure integrity), mistargeting has dire environmental, health, and safety consequences (type 1 mistargetting) and negatively impacts resources while needlessly disturbing the inspected party (type 2 mistargetting). To mitigate these concerns, minimising mistargeting is paramount (see Section 1.3). To improve on risk assessment in a *data-driven* manner, data are required.

The *quality* of data presents difficult challenges towards implementing data-driven solutions, concretely in the form of *missingness* and *noise* in data. Missingness and noise are known to deteriorate the performance of learned models [Sidi et al., 2012]. On the one hand, high quality data are seldom assured in real world applications, since no data generation method is impervious to flaws (e.g., human entry errors or faulty automated sensors). On the other hand, issues related to low quality data are, in themselves, of particular interest in risk assessment: what might be perceived as low quality data may in actuality be noncompliant behaviour. For example, since different costs are associated with the transport of specific waste materials, companies have financial incentive to purposefully manipulate their transport reports.

In their daily activities, inspectors consider a plethora of factors, together with their domain knowledge, leading to risk assessment decisions. However, not all factors contribute equally to the decision-making process. The prioritisation of ships for inspection via country flag is a case in point.

Traditionally, the country flag of a ship is considered as a proxy for inspection priority: ships sailing under specific country flags are more prone to inspection than other ships with other country flags according to a colour coding —white, grey, or black— based on the detention ratio of ships for that country [Paris MoU, 2020]. The flag is a problem for at least two reasons. First, ships may easily change flags, which allows companies to circumvent risk assessment protocols and elude inspection [Cariou and Wolff, 2011]. Second, the colour of the flag might disproportionally influence the inspection process, which may lead to confirmation bias.

Data represents the *administrative* reality of its encapsulating domain. In other words, they do not necessarily represent the *actual* reality: following from the ship inspection example, most inspectors prioritise high-risk country flags, which will generate data *biased* with respect to that selection. Should a data-driven *tool* (or *model*) be generated from a biased representation of the world, the tool itself may also be biased. Techniques must therefore be employed which reduce bias in models learned from biased data (i.e., learning *fair* models).

1.2 Machine Learning Preliminaries

The term *machine learning* was first introduced in the work by [Samuel, 1959], in which a computer was programmed to learn to play the game of checkers. A general definition would later be proposed by [Mitchell, 1997] as follows.

Definition 1.4 – ***Machine learning***

> Machine learning is the study of computer algorithms that improve automatically through experience. A program is said to learn from experience E with respect to task T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

Although immense progress has been made since the introduction of the term *machine learning* —from a game of checkers to speech recognition, computer vision, and fraud detection, to name a few— the general definition still holds.

The responsibility of learning a task is delegated to, what is commonly referred to as, a learning algorithm or *learner*: a set of instructions, under which a *loss* —conversely, *gain*— function is either minimised or maximised, respectively. The learning process should culminate in finding the *target function* (i.e., *model*) which translates to the task being solved, herein defined.

Definition 1.5 – ***Target function (model)***

> The target function (or model) is the learned function which, provided an input, returns an output which solves the task for which it was learned.

For distinct tasks, specific learners and loss functions may be used. Consider the case in which inspectors must select which ships to inspect from a myriad of vessels. The problem may then be posed as:

Given the characteristics of a vessel, should it be inspected?

In this case, the task is to predict whether or not there is a motive to inspect the ship. The goal is to learn the target function f (or model), of which the *input* x is some vector representation of a vessel and the output y is a *class label* indicating the decision to either $inspect$ $(+)$ or $not\text{-}inspect$ $(-)$. Formally, the target function may be given as $f : x \in X \rightarrow y \in Y \subseteq \{+, -\}$, and finding such a function is generally referred to as the *classification* problem [James et al., 2013]. Under a *supervised* learning scenario, learning occurs from a set of observations of which the class labels are known, such that new observations may be classified.

Yet, f may not immediately output a class label $\{+, -\}$. Instead, the output may be given as a *classification score* proportional to the *posterior class probability* $f(x) \propto P(y|x)$. By applying a threshold t to $f(x)$, a class prediction $\hat{y}$ is induced.

The class is predicted as either positive "+" if $f(x) \geq t$, or negative "−" if $f(x) < t$. For simplicity, $\hat{y}_+$ and $\hat{y} = +$ are equivalent; the same holds for (a) $\hat{y}_-$ and $\hat{y} = -$, (b) y_+ and $y = +$, and (c) y_- and $y = -$.

An example of supervised classifier learning as a solution to the classification problem is illustrated in Fig. 1.1. Here we see the combination of a class label and a feature vector. The class label of each sample is represented by colour: red indicating a positive class label y_+, and blue indicating a negative class label y_-. The feature vector of a sample is, in this case, of length 2 and is represented as a point of which the coordinates are the values of each feature: *Feature 1* is the horizontal axis and *Feature 2* is the vertical axis.

A solution to this classification problem example is given in the form of the learned target function $f(x)$, of which the output is represented as a colour gradient in the feature space (i.e., graph) and colour bar: a solid red colour indicates a high classification score, a white colour indicates a classification score of 0, and a solid blue colour indicates a low classification score. Class predictions can then be induced for unseen observations by considering the threshold $t = 0$, marked as a dotted line: if $f(x) \geq 0$, then the class label is predicted as positive ($\hat{y}_+$); otherwise negative ($\hat{y}_-$).

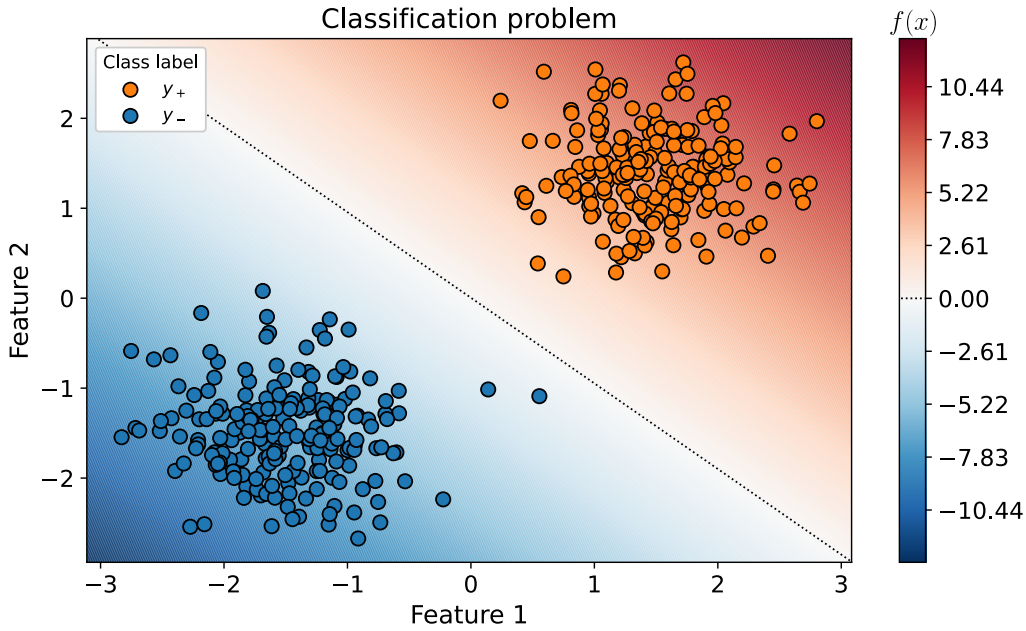

Figure 1.1: **Supervised classifier learning.** The colour gradient represents the classification score $f(x)$ of a classifier learned on the observations shown. The dotted line indicates the threshold $t = 0$ which induces a class label prediction for unseen observations: positive if $f(x) \geq 0$, and negative otherwise.

The performance of a model is estimated on data which was not used for learning, simulating the new (unobserved) real-world observations. These disjoint sets are typically denoted as the *train* (or *learning)* set and the *test* set. Multiple train-test splits (also known as folds) may be used such that the average performance across test sets is computed. This is termed *cross-validation* (CV) and is commonly applied to compute the expected performance of the final deployed model [Stone, 1974].

In the literature, the Area Under the receiver operating characteristic Curve (AUC) [Hanley and McNeil, 1982] is the standard used to measure the performance of a classification model [Flach, 2016]. It is defined as follows.

Definition 1.6 – ***AUC***

AUC is a measure of classification performance which considers the ranking of the output of a model. It quantifies the class separability of a learned model and is given as the probability that, provided a test set, a test sample y_+ selected at random will have a greater classification score than that of a test sample y_- also selected at random.

The curve it plotted in a graph by considering at each threshold t, the true positive rate (TPR) and the false positive rate (FPR): the vertical axis represents the TPR, and the horizontal axis represents the FPR. The greater the AUC — between 0.5 (random ordering) and 1 (perfect ordering)— the greater the performance. To note, other methods exist to compute the AUC which we address in Chapter 5.

It is known that there is no single learning algorithm best suited for all potential classification problems, referred to as the *no free lunch theorem* [Wolpert and Macready, 1997]. Yet, we note that for tabular data —the type of data handled by inspectors— decision tree learning algorithms [Breiman et al., 1984] are known to produce well-performing models —even when learning from low quality data— when applied under bagging (i.e., random forests) or (gradient) boosting strategies [Dogru and Subasi, 2018]. Taking these notions into consideration, the thesis focuses on decision tree learning algorithms.

1.3 Data Quality

The quality of the data used to learn a model often impacts the performance of the *downstream* (or *ulterior*) task of the model (e.g., targeting noncompliance in risk assessment). As per [Fürber, 2016], data quality is defined as follows.

Definition 1.7 – ***Data quality***

Data quality is the degree to which data fulfil requirements. The requirements can thereby be defined by (1) quality requirements of several different individuals of groups of individuals, (2) standards, (3) laws and other regulatory requirements, (4) business policies, or (5) expectations of data processing applications.

Following the definition, the fifth requirement is our main data quality provision. Broadly speaking, data are typically considered of either of *high quality* or *low quality* if they are well-suited or ill-suited, respectively, for the intended downstream task. The latter case being often anecdotally referred to as *garbage in, garbage out* [Rose and Fischer, 2011].

Poor data quality may manifest itself differently. For example, learning from data with insufficient sample size leads to a poor-performing model and, hence, a low performance of the downstream task. Specifically in this work, the focus on data quality is in terms of *missingness* and *noise*, given their prevalence in the domain of the ILT. While missingness is the absence of values in data, noise relates to data values which are inconsistent or erroneous [Sidi et al., 2012]. The names of these types of data quality issues are preceded by an *M* (missingness), or an *N* (noise), see below. Towards building reliable models, these issues must be considered.

1.3.1 Missingness

When dealing with real-world data, the absence of feature values in samples is bound to occur. This occurrence is termed missingness in data and is defined as follows [Beale and Little, 1975].

Definition 1.8 – ***Missingness***

Missingness in data is the occurrence of absence (i.e., missing values) in one or more features of one or more samples.

Missingness is characterised according to the relationship between the missing entries, the observed data, and the values which are missing; these relationships are categorised into three mechanisms: (1) missingness completely at random (MCAR); (2) missingness at random (MAR); and (3) missingness not at random (MNAR). We define each of the three mechanisms below, following [Little and Rubin, 2019].

Definition 1.9 – ***MCAR***

MCAR is the mechanism of missingness which assumes that the probability that a data value is missing is the same for all samples and features.

Under this mechanism, there is *neither* a relationship between the missing values and the remainder of the observed (non-missing) entries *nor* a relationship with the missing value itself. This means the distribution of missingness is independent of the data. An example of this mechanism is a sulphur sensor which runs out of power: some of the data will be missing due to a random event. Nevertheless, MCAR is generally atypical when dealing with real-world data.

Definition 1.10 – ***MAR***

MAR is the mechanism of missingness which assumes that the events that lead to missingness are dependent of the feature values in the observed (i.e., non-missing) data.

The MAR mechanism assumes that missing entries have some form of dependency with respect to the observed entries. For instance, if certain ports do not have sulphur sensors, then vessels travelling through those ports will not generate data regarding those sulphur measures.

Definition 1.11 – ***MNAR***

MNAR is the mechanism of missingness which assumes that the missingness is dependent on actual value that is missing as well as the observed values.

The MNAR mechanism occurs when the absence of entries is *dependent* on both (a) the unseen values and (b) the observed data. For example, when companies which fail to report on their emissions are the most likely to have systematically higher emission levels.

Traditionally, learning algorithms are incapable of handling data with missing values [García-Laencina et al., 2010]. The data must first be artificially made *complete*, i.e., without missing values, via *missing data-handling techniques*. In the missingness literature, two prominent categories of missing data-handling techniques are considered [Enders, 2010]: (1) imputation; and (2) missing-indicator. These are defined below. To note, several imputation methods exist in the literature [Enders, 2010]. We elaborate further on this topic in Chapter 2.

Definition 1.12 – ***Imputation***

Imputation is the process of *filling in* missing values based on the available data.

Definition 1.13 – ***Missing-indicator***

Missing-indicator is the method by which missingness is *encoded*, by generating an additional binary feature representing the presence or absence of values, and by assigning the same value to all missing values of the feature of concern.

It is known that (a) the mechanism of missingness, (b) the choice of missing data-handling technique, and (c) the learning algorithm jointly play a crucial role in the final model performance [Garciarena and Santana, 2017]. For instance, under non-MCAR, the missing-indicator method is a viable solution towards classifier learning [Lipton et al., 2016].

Discerning which missing mechanism is present is a challenging task. While it is impossible to distinguish between MAR and MNAR —as the necessary information for the distinction is itself missing— a test for MCAR vs not-MCAR has been proposed [Little, 1988]. Yet, the outcome of the test is not entirely reliable, as false positives and false negatives may still occur.

Real-world data are seldom MCAR [Van Buuren, 2018]. However, testing for MCAR does not provide a guaranteed result. Even though the assumption of non-MCAR data is generally correct in risk assessment, the MCAR mechanism is still possible and not necessarily detectable. Putting it differently, the issue is to find a solution to missingness which mitigate the detriment to the performance of the downstream task in cases where the assumption of non-MCAR is false. We will focus on this issue in Chapter 2.

1.3.2 Noise

In classification, noise in data is defined as follows [Angluin and Laird, 1988].

Definition 1.14 – ***Noise***

Noise in data is the presence of elements in (a) the feature(s) and/or (b) the class label which obscures their relationship and complicates model learning.

The major consequence of noise is the performance degradation of the final learned model when it is ignored [Wilson and Martinez, 2000]. It negatively impacts the performance of the learned model by obscuring the relationship between features and class label. Noise is denominated as either (a) *feature noise* or (b) *(class) label noise*, depending on the elements affected [Sáez et al., 2014]; these denominations are defined below.

Definition 1.15 – ***Feature noise***

Feature noise is the presence of elements in the feature values of samples which obscures the relationship between the features and the class label.

Definition 1.16 – ***Label noise***

Label noise is the presence of erroneous class labels in samples (i.e., mislabels).

Depending on the type of noise (feature or label), the compromise in performance of the downstream classification task varies. In general, we may notice that feature noise tends to be less detrimental than label noise [Zhu and Wu, 2004]. Here we note that in cases where both feature and label noises are present, the noise is denoted as a special case of label noise [Frénay and Verleysen, 2013]. As such, hereinafter, the terms *noise* and *label noise* are synonymous, and are both denoted as N.

Label noise is described according to the relationship between the mislabels and data characteristics. Three *label noise mechanisms* are used to categorise these relationships [Frénay and Verleysen, 2013]: (1) label noise completely at random (NCAR); (2) label noise at random (NAR); and (3) label noise not at random (NNAR). We define each of the mechanisms below.

Definition 1.17 – ***NCAR***

NCAR is the label noise mechanism which assumes that the probability that a sample is mislabelled is the same for all samples.

NCAR occurs when the proportion of mislabels is the same across classes. It is associated with random errors in the data generation process; e.g., automated sensor errors.

Definition 1.18 – ***NAR***

NAR is the label noise mechanism which assumes that the probability that a sample is mislabelled be dependent on the class label of the sample.

NAR entails different proportions of mislabels across the different classes. In other words, samples of one class are more prone to being mislabels than samples from another class. This might result from ill-calibrated tests to determine the outcome of risk assessment; e.g., targeting protocols which are too stringent or too relaxed.

Definition 1.19 – ***NNAR***

NNAR is the label noise mechanism which assumes that the probability that a sample is mislabelled is dependent on the class label and/or the features values.

In NNAR, mislabels may be associate with specific regions of the feature space, and their proportion may or may not be the same across classes. This mechanism is particularly interesting as it relates to the report-manipulation example from Section 1.1.

Since label noise may correlate to noncompliance within the context of risk assessment, especially in conjunction with feature noise, we focus on the NNAR scenario. Addressing label noise towards model *learning* often involves a prerequisite in the form of a sample *detection* step. Generally, label noise detection approaches leverage supervised learning methods into producing mislabelling *detection scores*. Detection scores are used to identify samples such that higher detection scores indicate higher likelihood of mislabel, and can be generated by exploiting classification scores $f(x)$: the lower the classification score of a sample towards its class label, the higher the detection score for being a noisy label. Given that these scores quantify the amount of label noise in samples [Jeatrakul et al., 2010], they may be used to better learn a classification model trained on label-noisy data [Liu and Tao, 2015].

Under the current risk assessment scenario, it would be advantageous to exploit these scores two-fold. First, when label noise mechanisms may translate to non-compliance, detection scores can directly be used as risk assessment scores. Second, detection scores may be incorporated into model learning such that the performance of the final learned classifier is the least compromised by noise, promoting better-performing risk assessment models. The work in this thesis addresses these two topics in Chapter 3 and Chapter 4, respectively.

1.4 Fairness

Machine learning algorithms model all sorts of relations between features and outcomes in historical training data, including potential societal biases [Richardson, 2022]. Within the Inspectorate, confirmation bias in historical inspection data is a case in point (we expand on it in Chapter 5).

The problem is to learn a mostly *unbiased* model from biased data. We mention *mostly* since, as detailed further, a completely unbiased model has a completely random classification output, rendering it useless. Learning with biased data is a problem traditionally termed *fairness* [Barocas et al., 2017] and is defined as follows.

Definition 1.20 – ***Fairness***

Fairness in machine learning is the study and tentative correction of algorithmic bias which results from learning with biased data.

A model is deemed more or less fair if its output has lesser or greater dependency (i.e., bias), respectively, towards some *sensitive* characteristic, such as nationality, age, or gender; i.e., a model is deemed less fair if it *favours* certain groups or individuals over others. To note, although the term *bias* may have different meanings in other fields (e.g., the bias-variance trade-off [Kohavi et al., 1996, Meertens et al., 2021]), in this context it is used antithetically to the term *fairness*.

Without loss of generality, the sensitive attribute of a sample is denoted $s \in S \subseteq \{-, +\}$ and represents sensitive group information, such as gender; the male and female groups are represented as S_+ and S_- or vice-versa, and samples pertaining to each group are respectively s_+ and s_-. Measures of fairness attempt to quantify the disparity of the model output between the groups conditioned on the sensitive attributes. The model output considered may be either (a) the class label prediction $\hat{y}$ induced by a set threshold t or (b) the classification score $f(x)$ [Venkatasubramanian, 2019]. Below we define two prevalent measures of fairness as described in the literature. They consider each of these outputs and are called: (1) *demographic parity* [Feldman et al., 2015] and (2) *strong demographic parity* [Jiang et al., 2020].

The definition of demographic parity follows.

Definition 1.21 – ***Demographic parity***

Demographic parity is the fairness measure which considers the difference in the proportion of positive outcomes (i.e., positive class label predictions) between two sensitive groups S_+ and S_-.

An example of demographic parity is the difference in the proportion of men who are hired vs the proportion of women who are hired. By extending the definition of demographic parity to account for the classification score (instead of the induced class prediction), the measure of strong demographic parity can be defined as follows.

Definition 1.22 – ***Strong demographic parity***

Strong demographic parity is the fairness measure which quantifies the fairness of a learned model by considering the difference in the ranking of classification scores across sensitive groups S_+ and S_-.

The computational definitions of demographic parity and strong demographic parity are given in Chapter 5.

To note, other fairness measures exist which consider different relations between model output and sensitive information; for example, the TPR and/or FPR conditioned on the sensitive groups (i.e., equal opportunity and equalised odds) [Pessach and Shmueli, 2022]. Yet, in the thesis we focus on the aforementioned (strong) demographic parities, given their link between class prediction and classification score. For both demographic parity measures, values closer to 0 indicate *model fairness*, whereas values closer to 1 indicate *model bias* Moreover, we remark that the strong demographic parity is conceptually similar to the AUC performance measure, but it is conditioned on the sensitive attribute values. We make use of this observation in Chapter 5.

1.4.1 Performance-Fairness Trade-Off

There exists a phenomenon under which the following holds. As model fairness increases, the more likely it is that the predictive performance decreases. This is known as the *performance-fairness trade-off* [Zafar et al., 2017]. It is a result of the decorrelation between the features and sensitive attribute, under the assumption of bias in data [Kleinberg et al., 2016].

Nevertheless, the performance-fairness trade-off is not necessarily balanced: greatly improving model fairness does not require a large decrease in model performance. Depending on the dataset, the corresponding correlation between sensitive attributes, and the target variable, it is possible to ensure adequate model fairness with limited decrease in predictive performance. In other words, the trade-off can be leveraged to find the optimal performance-fairness pair of a specific scenario.

The *tunability* of the performance-fairness trade-off in a model should, therefore, be considered. Towards its adequate implementation, we decompose the requirements of the tunability process into the two following: (1) granularity, and (2) intuitiveness.

First, the granularity of the tunability must be implemented such that an optimal performance-fairness pair may be found. If the granularity is insufficient, then the optimal trade-off between performance and fairness may not be reached. To put it differently, by incorporating fine-tuning into the trade-off, it is assured that the *sweet spot* of the performance-fairness is attainable.

Second, the tunability should be incorporated in an intuitive manner towards model usability. Alongside their domain knowledge, relevant stakeholders (in conjunction with the aid of the machine learning expert) should be able to decide which performance-fairness trade-off point is the optimal solution given a specific problem. By making the tunability process accessible, this task-dependent optimality can be assured. We explore the two aforementioned requirements in Chapter 5.

1.4.2 Addressing Fairness in Machine Learning

Taxonomically, three distinct mechanisms have been proposed to address fairness in machine learning [Pessach and Shmueli, 2020]. Each mechanism addresses fairness at different stages of the model learning process: (1) pre-processing; (2) post-processing; and (3) in-processing.

First, *pre-processing* relates to changes made within the training set prior to learning a model; e.g., by manipulating the training set specifically towards the homogenisation of the distributions across the different sensitive groups, making it more difficult for the final learned classifier to distinguish between the sensitive groups [Feldman et al., 2015]. This mechanism is sub-optimal because sample manipulation neglects the dependency over the classification task; i.e., the bias in data may still be exploited [Goldfarb-Tarrant et al., 2020].

Second, *post-processing* relates to changes made to the output of the final trained model, correcting decisions over sensitive groups [Hajian et al., 2015]; for instance, by having different decision thresholds t for each group [Corbett-Davies et al., 2017]. However, using sensitive information as input to determine a final outcome —e.g., hire if male and not-hire if female, for the same model score— is often not viable and potentially illegal under the General Data Protection Regulation in EU law [Goddard, 2017].

Third, *in-processing* encompasses the development and/or modification of classification algorithms. In this manner, models account for both predictive performance and fairness during learning by exploiting the relation between the features and sensitive attributes [Bechavod and Ligett, 2017].

Across the three bias-addressing mechanisms, in-processing is the most prevalent in the current literature, with overall superior classification performance and fairness, and the possibility to adequately tune the trade-off [Kamishima et al., 2012, Goh et al., 2016, Woodworth et al., 2017].

The applicability of the mechanisms relates to the degree of freedom of the developer. With pre-processing, there is only access to the data and not the model or its output; i.e., it is most useful for third party model development. In post-processing, only the output of a model is accessible; e.g., closed source algorithms. In-processing implies full developmental privileges (data, model, and output), allowing the relevant requirements to be combined.

Towards accomplishing the work presented in this thesis, we were allowed access by the ILT to their data. Moreover, full model development privileges were provided, including model output. Since, as stated, we are in full control of the algorithmic development, and given the prevalence and overall superiority of in-processing in the current literature, model fairness is addressed in an in-processing fashion in this thesis.

1.5 Problem Statement and Research Questions

In this thesis we are motivated by the real-world operations of the inspectors of the ILT towards risk assessment, which benefit from data-driven (i.e., machine learning) methodologies.

1.5.1 Problem Statement

The shift towards a data-driven paradigm in the operations of the risk assessment experts harbours considerable concerns given the high-risk profile of their endeavours. Based on this observation, we formulate the following PS.

> **PS:** *How can machine learning methods advance data-driven risk assessment by the Inspectorate in a reliable and fair manner?*

To address the PS, we will decompose it into three tractable RQs.

1.5.2 Research Questions

Missingness is a data-quality issue that impacts the performance of a downstream task on a model learned from a dataset. This impact must be considered, as to minimise the performance decrease during operational deployment. The performance of the downstream task of a model learned on data with missingness may vary depending on the joint selection of (a) the missing data-handling technique —imputation, missing-indicator, complete-case analysis, to name a few— (b) the choice of learning algorithm, and (c) the underlying missing mechanism.

Albeit real-world data —such as the one generated by the ILT— is seldom MCAR, it is still a possibility and testing for it does not provide a guaranteed result. Having made these observations, the first RQ is formulated as follows.

> **RQ1:** *Given data with missing values, which (a) missing data-handling technique and (b) learning algorithm should be jointly selected such that, regardless of the missing mechanism, the detriment to the downstream task performance is minimal when compared to the non-missing (unavailable) case?*

Data permeated with noise is detrimental to model learning. Moreover, the detection of these noisy samples is a prerequisite for model learning with noisy data. While noise in classification is strictly any disruption of the relationship between features distribution and labels, we tackle label noise as mislabels in the data under specific feature distribution conditions (i.e., with additional feature noise). Our reasoning for this is two-fold: (1) this noise may be indicative of noncompliant behaviour in risk assessment; and (2) label noise is more detrimental to model learning than feature noise alone.

It is advantageous to produce noisy-sample detection scores usable for both (a) noncompliance targeting, and (b) model learning. Accordingly, the second RQ is a *compound* one —decomposable into RQ2(a) and RQ2(b)— and follows.

> **RQ2:** *Given data with label noise, how can noisy-samples be (a) adequately detected, and (b) used to learn a well-performing model?*

Fairness in machine learning is paramount to handle biased data. Not only must the learned model exhibit adequate predictive performance, it must also ensure that the predictions are the least impacted by the data bias.

To measure the impact of biased data in the learned model, different fairness measures exist which have an analogous (i.e., corresponding) performance measure. Moreover, the performance-fairness trade-off is a well-known phenomenon and may be exploited to achieve the most fairness increase at the cost of the least performance decrease. To exploit the performance-fairness trade-off, model learning must allow for its tunability. An in-processing approach to fairness enables this. Merging these remarks, we arrive at our third RQ.

> **RQ3:** *How can we, from biased data, learn a model tunable with respect to the performance-fairness trade-off such that the selection of the trade-off point is made intuitive for the relevant stakeholders?*

1.5.3 Research Methodology

To provide an answer to the PS, the work in this thesis follows the well-established CRoss Industry Standard Process for Data Mining (CRISP-DM) [Martínez-Plumed et al., 2019], defined as follows and depicted in Fig 1.2.

Definition 1.23 – ***CRISP-DM***

CRISP-DM is a process model which decomposes a data science process into six phases: (1) business understanding; (2) data understanding; (3) data preparation; (4) modelling; (5) evaluation; and (6) deployment.

Here, we explicitly denote that, despite the subject of this thesis not being data mining, CRISP-DM still offers a valuable approach which helps guide our research. To begin our work, communication between the domain experts, stakeholders, and us researchers was fulcral; jointly, efforts were had to promote the first two phases to be best of our capacity. In this thesis, however, the focus is not in detailing the communication processes, but rather to describe the technical methods developed and their performance; i.e., *data preparation* (3), *modelling* (4), and *evaluation* (5). We further denote that the deployment phase is outside of the scope of this thesis, as it depends not solely on the adequacy of the technical methods, but also on changes at the organisational level.

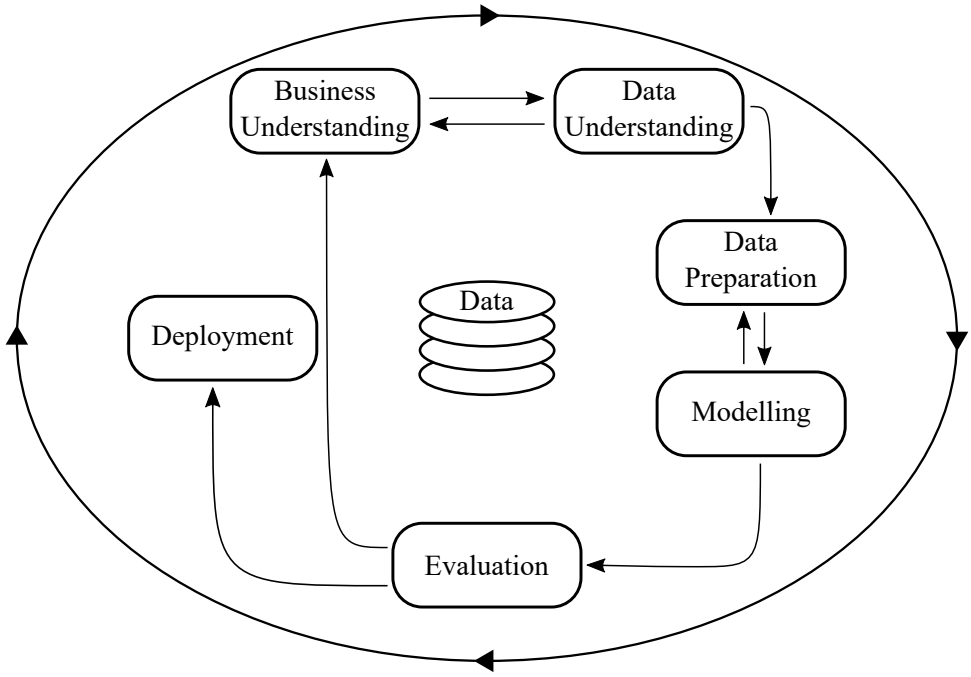

Figure 1.2: **CRISP-DM.** Process model comprised of six sequential phases.

The technical methods used to address RQ1 are detailed in Chapter 2, RQ2(a) in Chapter 3, RQ2(b) in Chapter 4, and RQ3 in Chapter 5.

1.6 Research Contributions

Below, we list the four main contributions of our research.

- *Contribution 1.* We show that towards supervised classifier learning with real-world missing data, a combination of (a) the missing-indicator method and (b) a decision tree learning algorithm —namely, gradient boosting— should be used to minimise the detriment in classification performance.

  According to the literature, non-MCAR scenarios can benefit from the missing-indicator method, measured as the downstream task performance. In the scenario of a non-MCAR assumption being falsely made, we compare several imputation methods to the missing-indicator method, quantifying their differences measured as the downstream classification performance. We empirically demonstrate that across different learning algorithms, (gradient) boosting architectures which incorporate feature selection processes are the least susceptible —if at all— to the sub-optimal decision of applying the missing-indicator method when dealing with data generated under MCAR.

- *Contribution 2.* We propose an approach to targeting and accounting noisy observations which, subsequently, allows for better learning with noisy data, which outperforms the competing literature methods.

  The scenario in which both label and feature noise permeate data is considered. By leveraging an already existing robust decision tree learning algorithm via gradient boosting, noisy-sample detection scores are computed in a CV manner, generating a model with well-calibrated output. Empirically, we performed extensive experimentation to compare our approach to other methods from both outlier and mislabel detection publications. Our novel method —termed EXPOSE— exhibited an overall improved performance over the methods against which it was compared.

- *Contribution 3.* We develop a strategy towards classifier learning for data with label noise through sample weighing which exhibits competitive performance when compared to the current literature, particularly adequate for datasets with large proportions of mislabels.

  Based on the aforementioned contribution in noisy-sample detection, the detection scores are leveraged to compute individual observation weights. The weights are applied within the learning process as coefficients in the logistic loss function. We empirically show that via well-calibrated posterior probability estimations, the *log-odds of an observation* may be leveraged in learning. Through an exhaustive experimental design, comprised of different proportions of both label and feature noise, we validate our proposed method —DENOISE— by comparing it to the state-of-the-art in learning from noisy data under the NNAR scenario. On average, our method achieves superior performance compared to the state-of-the-art.

- *Contribution 4.* We design a fair tree classifier which is independent of threshold in the performance loss as well as the fairness criterion loss. The classifier can be easily adjusted to assess performance-fairness trade-off points.

  The threshold-independent fairness measure of strong demographic parity is used and, by drawing from its analogy to the classification performance measure AUC, we arrive at the Splitting Criterion AUC For Fairness, or SCAFF. Incorporated in SCAFF, the orthogonality parameter Θ which regulates the performance-fairness trade-off. Our learning algorithm considers multiple sensitive attributes simultaneously of which the values may be multicategorical. When compared to other fair tree learning splitting criteria, our experiments with real-world data show our method is able to achieve classification performance and fairness which are on par at worst and superior at best, against those of the competing approaches in the fairness literature.

1.7 Thesis Overview

In Chapter 1, we introduced the current movement towards *trustworthy AI* in Europe. Then, we narrowed our scope towards the particular domain of risk assessment, with distinct concern for data-driven solutions within the Inspectorate of the Netherlands. We further established the required foundations to achieve these solutions. We formulated our PS, and decomposed it into the three RQs of the thesis. Next, we proposed our four contributions. The remainder of the thesis is given below.

- **Chapter 2** answers RQ1, resulting in Contribution 1. The content of the chapter is identical to that of the work by
  Pereira Barata, A., Takes, F. W., van den Herik, H. J., and Veenman, C. J. (2019). Imputation methods outperform missing-indicator for data missing completely at random. In *2019 International Conference on Data Mining Workshops (ICDMW)*, pages 407–414. IEEE

- **Chapter 3** answers RQ2(a). The result of the chapter is Contribution 2. The content of the chapter is identical to that of the work by
  Pereira Barata, A., Takes, F. W., van den Herik, H. J., and Veenman, C. J. (2021). The eXPose approach to crosslier detection. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2312–2319. IEEE

- **Chapter 4** provides an answer to RQ2(b). The result corresponds to Contribution 3. The content of the chapter is identical to that of the work by
  Pereira Barata, A., Takes, F. W., van den Herik, H. J., and Veenman, C. J. (2022). Noise-resilient classifier learning. *Pattern Recognition (under review)*

- **Chapter 5** answers RQ3, culminating in Contribution 4. The content of the chapter is identical to that of the work by
  Pereira Barata, A., Takes, F. W., van den Herik, H. J., and Veenman, C. J. (2022). Fair tree classifier using strong demographic parity. *Machine Learning (under review)*

- **Chapter 6** entails the conclusions of the present thesis, in three distinct sections. We (1) answer the three research questions, (2) provide an answer to the problem statement, and (3) discuss future work directions.

The papers presented in this thesis were produced as a joint collaboration between the supervisors and the PhD candidate. Discussions on research topics and how to tackle the problems described were addressed as a team. The writing was performed by the candidate, incorporating the commentary provided by the supervisors. The implementation of the experimental designs and gathering of the results were performed by the candidate.