FairML: ToolBox for Diagnosing Bias in Predictive Modeling

by

Julius A. Adebayo

B.S. Mechanical Engineering, Brigham Young University (2012)

Submitted to the Institute for Data, Systems, and Society & Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degrees of
Master of Science in Technology and Policy
and
Master of Science in Electrical Engineering and Computer Science
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2016

Signature redacted

Author ....
February 29, 2016

Signature redacted

Certified by ....
Dr. Lalana Kagal
Principal Research Scientist, CSAIL
Thesis Supervisor

Certified by ..... Signature redacted ....
Professor Harold Abelson
Class of 1922 Professor of Computer Science and Engineering
Thesis Supervisor

Certified by ..... Signature redacted .....
Professor Alex "Sandy" Pentland
Toshiba Professor of Media Arts and Sciences
Thesis Supervisor

Signature redacted

Accepted by ....
Professor Leslie A. Kolodziejski
Chair, Department Committee on Graduate Students

Signature redacted

Accepted by .....
Professor Munther A. Dahleh
Acting Director, Technology and Policy Program

FairML: ToolBox for Diagnosing Bias in Predictive Modeling

by

Julius A. Adebayo

Submitted to the Institute for Data, Systems, and Society & Department of
Electrical Engineering and Computer Science
on February 29, 2016, in partial fulfillment of the
requirements for the degrees of
Master of Science in Technology and Policy
and
Master of Science in Electrical Engineering and Computer Science

Abstract

Predictive models are increasingly deployed for the purpose of determining access to services such as credit, insurance, and employment. Despite societal gains in efficiency and productivity through deployment of these models, potential systemic flaws have not been fully addressed, particularly the potential for unintentional discrimination. This discrimination could be on the basis of race, gender, religion, sexual orientation, or other characteristics. This thesis addresses the question: *how can an analyst determine the relative significance of the inputs to a black-box predictive model in order to assess the model's fairness (or discriminatory extent)?* We present FairML, an end-to-end toolbox for auditing predictive models by quantifying the relative significance of the model's inputs. FairML leverages model compression and four input ranking algorithms to quantify a model's relative predictive dependence on its inputs. The relative significance of the inputs to a predictive model can then be used to assess the fairness (or discriminatory extent) of such a model. With FairML, analysts can more easily audit cumbersome predictive models that are difficult to interpret.

Thesis Supervisor: Dr. Lalana Kagal
Title: Principal Research Scientist, CSAIL

Thesis Supervisor: Professor Harold Abelson
Title: Class of 1922 Professor of Computer Science and Engineering

Thesis Supervisor: Professor Alex "Sandy" Pentland
Title: Toshiba Professor of Media Arts and Sciences

Acknowledgments

The road to this thesis was somewhat convoluted, and included several detours. I would like to thank several individuals that helped contribute to the final output. I would like to thank Dr. Lalana Kagal, Professor Hal Abelson, & Professor Sandy Pentland for useful feedback and guidance throughout the thesis process, and for supervising this thesis. It is been an honor and a priviledge to learn from them over these past years.

The work presented here is at the intersection of law and computer science. I learned a lot about law through the class taught by Danny Weitzner and Hal Abelson during the spring of 2015. The final paper from the class helped me crystallize my thoughts and eventually arrive at the question that inspired this work. I am particularly indebted to Mikella Hurley, Taesung Lee, Professor David Vladeck, & Professor Alvaro of Georgetown law for guidance, discussions, and helping me understand the legal implications of machine learning. A good chunk of the law section of this thesis resulted from the work done as part of the project for the class. Ultimately, I credit that class for helping me realize that the intersection of computer science and law is quite fascinating, and an area I enjoy working in. I am indeed grateful to Danny & Hal for teaching and putting together such a wonderful class.

I would like to thank the members of the Decentralized Information Group, CSAIL & the Human Dynamics Group, MIT Media Lab for helping me throughout my time as a master's student. I found the atmosphere at these two groups intellectually engaging. I would particularly like to thank Yoshi Suhara, Xiaowen Dong, Yves-Alexandre De Montjoye, Peter Krafft, & Ilaria Liccardi for helpful discussions and feedback on this work in particular.

Throughout my time at MIT, I have made several friends in the Technology and Policy Program (TPP), CSAIL, and across different areas. These friends have made MIT a very special experience for me. I would like to shout out my Jollof Rice Whatsapp [1] group for always making me laugh. I would also like to thank my Nigerian

[1] You know yourselves :)

crew at MIT for making it a lively and enjoyable place. It has been an honor being at MIT with these friends over the past few years, and I look forward to a lifetime of friendship.

I would also like to thank the TPP administrative staff, particularly Barb and Ed for helping me during this process. Oh, thanks Barb, for always laughing at all of my jokes!

My path to research started in the IDeA labs of Prof. Sean Warnick as an undergraduate. Sean has been a tremendous mentor and inspiration. Though he wasn't involved with this work, I would not have gotten this far without his help.

Ultimately, I only made it this far because I have a terrific support structure. Dr. Kunle, Mummy Dara, and the entire Johnson family have been a strong support for me throughout my time here in the US.

I left the best for last. Dad, Mum, Ola, & Bukky are the real heroes. As much as it might seem like this thesis is my accomplishment, it really is *YOUR* accomplishment. I am grateful for all the sacrifices that you have had to make to ensure that I get the opportunities that I have, and hope that I have been able to make you proud, in my own little way.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

> Big data is changing the world. But it is not changing Americans' belief in the value of protecting personal privacy, of ensuring fairness, or of preventing discrimination [58].
>
> Executive Office of the President, United States of America.

Access to large-scale data has led to an increase in the use of predictive modeling to drive decision making, particularly in industries like banking, insurance, and employment services [59]. The increased use of predictive models has led to greater efficiency and productivity. However, improper deployment of these models can lead to several unwanted consequences. One key concern is unintentional discrimination. It is important that decisions made in determining who has access to services are, in some sense, 'fair.' A predictive model can be susceptible to discrimination if it was trained on inputs that exhibit discriminatory patterns. In such a case, the predictive model can learn patterns of discrimination from data leading to high dependence on protected attributes like race, gender, religion, and sexual orientation. A predictive

model that significantly weights these protected attributes would tend to exhibit disparate outcomes for these groups of individuals. Hence, the focus of this thesis is on auditing predictive models to determine the relative significance of a model's inputs in determining outcomes. Given the relative significance of a model's inputs, an analyst can more easily assess a model's fairness (or degree of discrimination).

Before going into a more detailed motivation and overview, we present two scenarios that highlight the goal of the work presented here. These scenarios represent the practical applications of auditing models for bias.

- Imagine you are an analyst at the Federal Trade Commission or the Consumer Financial Protection Bureau. Over the past few days you have received complaints from individuals whose loan applications to Random National Bank were denied. These individuals indicate that they believe the bank is discriminating against them on the basis of either gender, race, or other immutable characteristics, which is prohibited by law. Now, Random National Bank uses a 10-layer convolutional neural network with one million parameters to determine loan recipients. As someone who presumably doesn't have expertise in neural networks, how would you audit Random National Bank's neural network to determine whether it is indeed discriminating against loan applicants on the basis of race or gender?

- Imagine it is the year 2050. MIT has completely transitioned to using ensembles of predictive models (classifiers) for determining who gains admission. At the end of its admissions cycle for year 2051, several applicants with perfect SAT scores and GPAs were rejected. Now, some of these applicants claim that MIT discriminated against them on the basis of their race. How would you audit MIT's admissions predictive model to detect the presence or absence of bias?

Of course, the gold standard approach to definitively answering the motivating questions posed would be to perform a randomized experiment. In the case of the bank, this would involve sending the bank identical applications that differ only based on, for example, gender or race, and then comparing the average number of loans given

to the different groups. In this experiment, randomization isolates the effect of race or gender, and one can now quantify the impact of race or gender on outcomes desired. However, such experimentation can be difficult and costly, and sometimes subject to design errors or execution. In some cases, experimentation is even impossible. Thus, an alternative means of evaluation is often necessary. To remedy this issue, we present a toolbox, FairML, that can be used to audit any predictive model to quantify the relative significance of a model's inputs in determining outcomes.

FairML consists of four ranking algorithms developed to enable interpretability of black-box predictive models. Further, as part of FairML, we present a novel input ranking algorithm, the Iterative Orthogonal Feature Projection Algorithm (IOFP), for iteratively quantifying the relative dependence of a black-box model on its input. Critically, IOFP can also be adapted to query black-box algorithms if direct access is available, which is often not the case for other ranking algorithms. We show through numerical simulations that the performance of our IOFP algorithm improves by 15 percent when direct access to the predictive model is available.

The goal of this work is not to vilify algorithms and predictive modeling, or even worse, condemn 'AI' as evil. On the contrary, increased use of these predictive models offers to usher society into a more productive and efficient era. However, unbridled use of these algorithms, across several domains, could have severely undesirable consequences. With FairML, we seek to tackle one potential issue, unintentional discrimination, that could arise without careful design and oversight of the predictive modeling process, particularly in industries such as insurance, banking, employment, and housing. It is our hope that auditing tools like FairML would actually enable increased use of predictive models, given the capability to now more clearly ascertain the 'reasoning' behind outcomes from these models.

In the rest of this introductory chapter, we present the questions that drive this thesis, an overview of key contributions, and a general outline of the upcoming chapters.

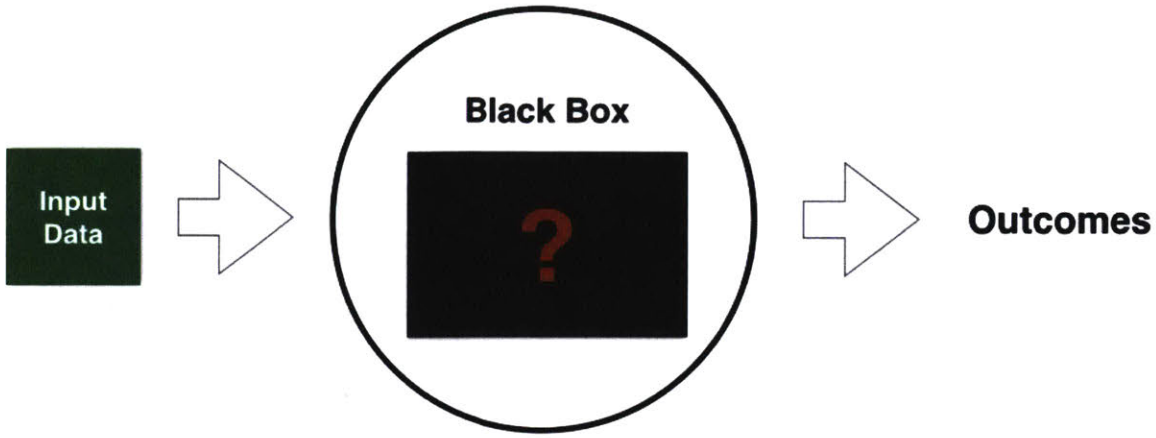


Figure 1-1: Diagram depicting a typical data mining process, and the overall motivation of this thesis: given a predictive model that is perhaps uninterpretable to an observer or data analyst, how can one determine the relative significance (ranking) of the model's inputs for the model's observed output?

1.1 Thesis Questions

We highlight below the key questions that we investigate.

1. **Technical:** Given a predictive model, one that is perhaps cumbersome and uninterpretable to a data analyst, how can one determine the relative significance (ranking) of the model's inputs on the model's observed output?

2. **Qualitative :** Given the relative significance of a predictive model on its inputs, how can we then assess the fairness of such a model?

3. **Qualitative :** What does it mean for a predictive model to be fair?

This thesis addresses question 1 above in a quantitative and technical manner, through a) the implementation of algorithms that can be used to determine the input ranking for any predictive model, and b) simulations demonstrating improved performance given direct access to the predictive model.

For questions 2 and 3, as with any sufficiently difficult and subjective inquiry, our answer is *it depends*. To answer these questions, we take a qualitative approach, and provide a series of discussions on different approaches that can be taken to ascertain

model fairness, given the output of our audits. *Ultimately, we believe model fairness is context dependent and in the eye of the beholder.*

1.2 Contribution

Having highlighted the driving questions for this thesis, we present here an overview of the key contributions.

1. We present FairML, an end-to-end system that can be used to audit any predictive model in order to determine the model's relative predictive dependence on its inputs. FairML outputs the results of its audits to analysts in an interpretable format.

    - As part of FairML, we present a novel input ranking methodology: Iterative Orthogonal Feature Projection Algorithm (IOFP).
    - Critically, IOFP can be applied whether the black-box algorithm is available for iterative querying or not. In the case where the black-box can be queried iteratively, we show through numerical simulations that IOFP provides improved performance compared to other algorithms that can't query the black-box model. Without direct access, IOFP's performance is similar to other popular traditional feature ranking algorithms.
    - Further, we also propose and implement non-linear extensions as part of IOFP in FairML to handle non-linear black-box models.
    - In addition to IOFP, we also implement three other ranking methodologies as part of FairML to increase the robustness of the combined ranking derived from using the platform.

2. To allow for easy access use of the FairML toolbox, we implemented a web application through which analysts can upload data to the toolbox for analysis. With our web application, analysts and users would only have to obtain input-output data from the black-box model of interest and upload this data to FairML for analysis.

3. Ultimately, the goal of FairML is to make it easier for analysts and policy makers to audit predictive models, so we demonstrate FairML on real-world data sets to show how the platform can be used to diagnose bias.

4. In light of the above framework, we also present a review of regulations such as the Equal Credit Opportunity Act, and the Fair Credit Reporting Act that address discrimination in access to services such as credit. We then highlight how these laws are currently inadequately addressing discriminatory data mining, and how they can be improved.

1.3 Overview

Now we provide the reader with a general outline of this thesis.

- In chapter 2, we provide a detailed overview of the general motivation behind the work presented here, taking the reader through the recent data revolution, the emergence of large-scale deployment of predictive models, issues relating to interpretability of predictive models, discriminatory data mining, inadequacies of the current laws, and the need for tools to audit cumbersome and non-interpretable predictive models.

- The work presented in this thesis is of an interdisciplinary flavor. Hence, it draws on previous work from different fields such as machine learning, law, statistics, and economics. In chapter 3, we present an overview of previous work on model compression, discriminatory data mining, and machine learning.

- In chapter 4, we present our end-to-end system, FairML, providing a detailed description of its individual components, in particular, the four key ranking methodologies implemented. We end the chapter with a discussion of nuances, limitations, and possible refinements of the platform.

- In chapter 5, we present results of different simulations designed to evaluate the ranking algorithms implemented as part of FairML. We show that, given direct

access to a predictive model, the performance of our IOFP algorithm improves by about 15 percent.

- In chapter 6, we present different audits performed on real-world data sets. We audit a large European bank's credit limit and probability-of-default models for bias on the basis of gender.

- In chapter 7, we discuss the several implications of this work, touching on issues relating to the notion of algorithmic fairness, causal inference, and future improvements to FairML.

- In chapter 8, we focus on the use of predictive models for credit assessment in the banking industry. We further present an overview of the Fair Credit Reporting Act (FCRA), and the Equal Credit Opportunity Act (ECOA), two major pieces of legislation that regulate practice in the credit industry. In particular, we present several ways in which both pieces of legislation are currently ill-equipped to handle the emergence of predictive models for credit assessment, and provide recommendations for how they can both be improved.

- In chapter 9, we summarize the thesis, highlighting contributions and future work.

CHAPTER 2

MOTIVATION

2.1 Explosion of Data

2.5 quintillion[1] bytes of new data is created everyday: we live in the era of big data [34]. According to various accounts, the total amount of data created, stored, and analyzed since 2011 till date constitutes 90% of all data created over the entire human history. It has now become a cliche to talk about the scale of data and the ubiquity of large scale computing power available for analyzing this data. The famous Mckinsey Global Institute report on big data highlights that going forward, competitive advantage lies with firms and individuals that can make judicious use of data in driving their actions.

A combination of factors have led to the exponential increase in data currently being experienced: a reduction in cost of sensors leading to availability of low-cost data acquisition devices, increased access to the Internet ensuring that data can be transferred more easily, and lower data storage costs. For example, as of 2011, it was estimated that there were 5 billion mobile phones being operated [46]. The ubiquity of mobile phones and other sensors has increased the *volume, variety, and velocity*

[1] 1 quintillion bytes corresponds to 10^{18} bytes

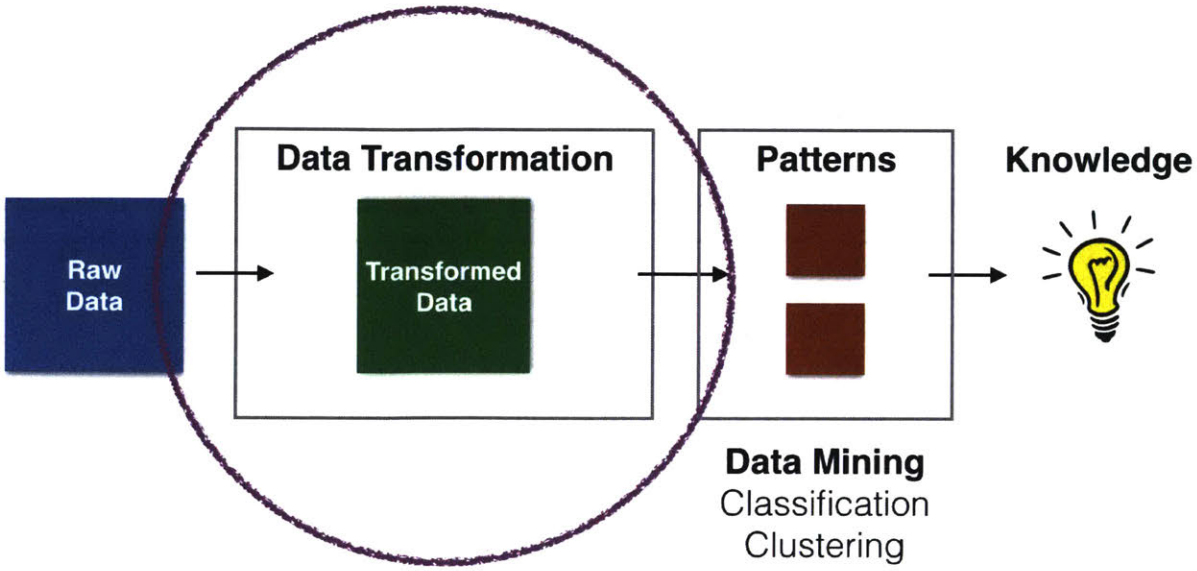


Figure 2-1: A schematic showing a general overview of the predictive modeling process. The goal of predictive modeling is to learn insights that can drive decision making from data.

[23] of the data being gathered. Data being gathered today ranges from traditional kinds like weather information, census data, and health records to social media data from Twitter, and mobile phone records [39].

2.2 Deriving Insights From Data: Predictive Models

Given access to granular large-scale data, the challenge now shifts to digesting and analyzing these large data sets to derive insights that can drive beneficial action. The renewed focus on data analysis has rejuvenated fields such as statistics that emphasize data analysis, as well as created new ones such as *data science*. Large-scale data combined with sound data analysis to derive actionable insights can typically lead to dramatic increases in efficiency [57]. For example, consider a bank that needs to process thousands of loan applications to identify suitable ones to which it should allocate capital. Traditionally, such tasks have been human-driven, and can often be rife with errors, given ad hoc rules on creditworthiness. However, with predictive models, such tasks can be easily automated with much higher accuracies.

Application of predictive modeling from machine learning and statistics to large-scale data is now quite pervasive. Predictive models ingest raw data to forecast the probability of a given outcome. Banks and financial firms have long adopted predictive models as a means for forecasting the performance of various financial indices. In policy and planning, predictive models have been shown to be quite effective and accurate at identifying areas with high incidence of poverty [66, 9], crime [10, 30], and traffic [31, 14]. Predictive models have also seen city-level deployment in various regions of the world. For example, the city of Chicago created a Data Science Division to "store, analyze, research, visualize, publish, and liberate data for city users and the public" [52]. The division has been able to leverage its data to more easily identify restaurants that might pose a health risk to residents. In another case, the city has been able to leverage its crime databases to identify individuals that are susceptible to committing particularly heinous crimes [52].

2.3 Black-Box Models and Interpretability

The potential increased efficiency and societal gains from leveraging predictive modeling seem limitless, and have rightly led to the widespread adoption of these models. As a case in point, Figure 2-2 shows the growth of the application of a particular type of predictive model, a deep neural network, at Google. The significant growth demonstrated in Figure 2-2 foreshadows large-scale deployment of predictive models beyond traditional domains in the coming years. In particular, use of predictive modeling for decision making in determining access to services is starting to become the de facto standard in industries such as banking, insurance, housing, and employment.

As the need for more accurate forecasts or predictions has heightened, there has been an increase in the use of complicated, often uninterpretable predictive models in making forecasts from data. Increasingly, these predictive models tend to have millions of parameters and are typically considered black-boxes by practitioners. This is because the models often generate highly accurate results, but an in-depth understanding of the underlying reasons behind these accurate results is generally lacking.

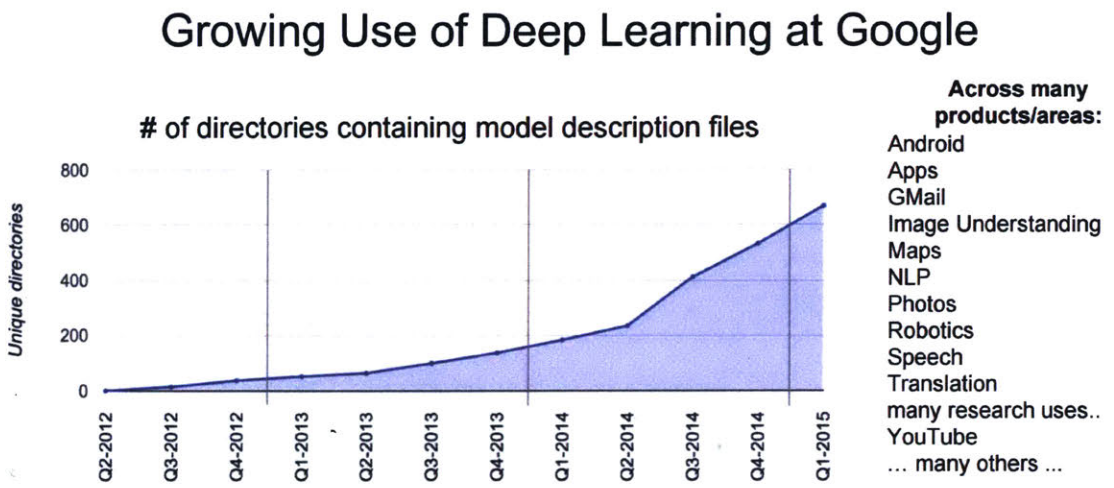


Figure 2-2: Figure showing the growth of the deep learning (neural networks) methodology across Google products, especially several user-facing products [21]. Neural networks are a class of predictive models loosely modeled after the human brain that are particularly suited for approximating classes of functions with high-dimensional inputs. Neural networks have surged in use since 2012 due to their high predictive accuracies in vision and speech recognition tasks.

Hence, practitioners resort to feeding in input to these black-box models, then generating results without truly understanding why their models are performing well. In fields such as computer vision or speech recognition where the task is often to identify or recognize a signal structure, a lack of true understanding of the internal workings of the underlying model generating the predictions can be excused. However, in industries such as banking, insurance, and employment where access to these services is essential for livelihood, it is of **paramount** importance that the practitioner applying a predictive model in this setting truly understands her model's internal workings.

If a predictive model is to be used to make decisions in a policy setting or a real-world context, it is important that the model builder is able to explain the reasons why the model assigns higher probabilities to certain events than others. Such fundamental justification is at the core of policy making at all levels. However, a broad class of predictive models exist for which interpretability of the model's probabilistic outcomes is difficult. These predictive models are typically referred to as **black-box models**.

There are two major reasons why some commonly used predictive models are difficult to interpret. First, highly accurate predictive models tend to consist of an

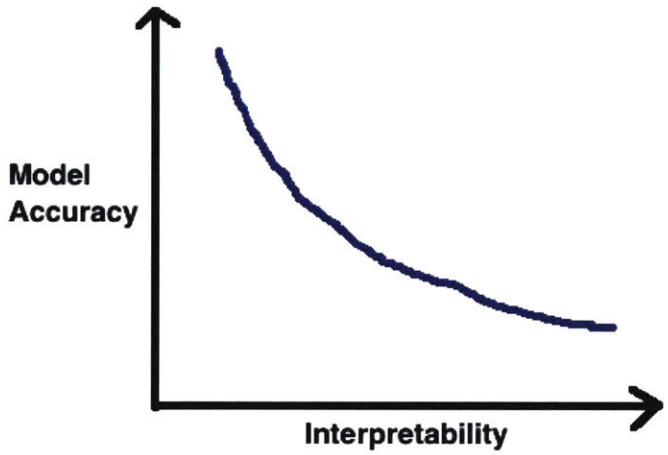


Figure 2-3: Figure showing the accuracy vs complexity tradeoff for predictive models.

aggregation of several other predictive models. However, combining ensembles of predictive models often makes it difficult to gain insights into how these ensembles are assigning different probabilities to outcomes. Second, increased accuracy is often derived from repeated applications of non-linear transformations to input data, but repeated non-linear transformations applied to input data often makes it challenging to figure out the exact relationship between the original input and the derived transformations. For example, neural networks apply transformations such as sigmoid, tanh, and other activation-like functions to input data. If an input data consists of attributes such as age, sex, and education level, it can prove particularly difficult for data analysts to interpret repeated non-linear transformations of these attributes. As shown in Figure 2-3, there seems to exist a fundamental trade-off between accuracy and model complexity. Methods that can begin to overcome these challenges will help us to more easily develop powerful, yet accurate and interpretable predictive models.

2.4 Big Data and Discrimination

The use of predictive models for decision making in industries like employment, housing, banking, and policy brings with it several potential difficulties. Because these industries are highly regulated, governments and law-making bodies expect companies and other entities operating in these industries to comply with rules and regulations

on the books. Often, a human is tasked with ensuring that companies in these industries comply with the required rules and regulation. With a shift from traditional human decision making to the use of predictive models, non-experts in data analysis and statistics will find it difficult to audit predictive models for compliance with the law. Typically, rules and regulations require entities in regulated industries to provide detailed feedback and reasoning for all decisions made in the process of performing any transaction. For example, in the United States, if one applies for a credit card and is denied, the credit granting entity, typically a bank, is required to provide the applicant a written letter describing the reasons for rejection [16]. Consequently, if the bank were using a difficult-to-interpret predictive model to assess customer credit worthiness, it would have difficulty providing justification for its decisions as required by law.

The principal concern that motivates this thesis is the increase in unintentional discrimination that can result from pervasive deployment of predictive models without care. Given the ease with which predictive models can learn patterns from data, proper care has to be taken to ensure that predictive models used in determining access to services do not place a undue emphasis on race, gender, religion, sexual orientation, and other protected attributes. In their paper titled *Big Data's Disparate Impact*, Barocas and Selbst point out that,

> advocates of algorithmic techniques like data mining argue that they eliminate human biases from the decision-making process. But an algorithm is only as good as the data it works with. Data mining can inherit the prejudices of prior decision-makers or reflect the widespread biases that persist in society at large. Often, the "patterns" it discovers are simply preexisting societal patterns of inequality and exclusion. Unthinking reliance on data mining can deny members of vulnerable groups full participation in society. Worse still, because the resulting discrimination is almost always an unintentional emergent property of the algorithm's use rather than a conscious choice by its programmers, it can be unusually hard to identify the source of the problem or to explain it to a court [8].

Data mining practitioners do not set out to discriminate; however, given unrepresentative data sources and slightly biased input, it is possible for algorithms to learn patterns of discrimination from input data. Discriminatory effects that arise from predictive models will often be unintended, as practitioners in the field will seldom want to explicitly use their predictive models to exclude or discriminate against certain groups of people. Hence, it is crucial to develop methods that can help in auditing the predictive modeling process to identify cases of unintentional discrimination.

While it is crucial to identify the presence of discrimination in a predictive modeling process, it is also important to be able to confirm its absence. As noted earlier, predictive modeling can lead to increased efficiency and improved decision making; consequently, it is important that legitimate uses of predictive models even in regulated industries are not hampered by inadequate and potentially damaging rules and regulations. A predictive model auditing process should be based on sound technical principles that can reveal the relative significance of a predictive model's inputs to its output.

2.5 Current Laws are Inadequate

In the United States, and in most developed countries, current rules and regulations are incapable of assessing predictive models for bias [8]. In the United States, willful and premeditated discrimination is termed "disparate treatment" while unintended discrimination resulting from facially neutral policies is termed "disparate impact" [8, 29]. To show disparate treatment, one has to be able to conclusively demonstrate that a practice was specifically designed to discriminate against certain groups of people. In most discrimination cases one can seldom show disparate treatment. More generally, discrimination-related cases are often disparate impact cases. For example, in the Griggs v. Duke Power Co. Supreme Court case of 1970, Duke Power Co. was instructed to do away with the requirement for "intelligence test scores and high school diplomas, qualifications largely correlated with race, to make hiring decisions" [29]. This Supreme Court decision gave birth to disparate impact.

In ascertaining unintended discrimination, i.e., disparate impact on a particular group of individuals, one is usually required to specify a threshold above which the disparate effects of a policy on different groups constitute disparate impact. A specific threshold for disparate impact does not currently exist. However, the U.S. Equal Employment Opportunity Commission (EEOC) currently advocates an 80 percent rule [8, 29]. For example, suppose we have a predictive model that is making loan decisions for different groups of people, if the probability of the model approving a loan for a female is less than 80 percent of the probability of the model approving a loan for a male, then, given an 80 percent threshold, this model exhibits disparate impact.

Despite its wide usage, disparate impact alone falls short as a means of diagnosing discrimination in a predictive model. Disparate impact focuses on the output of a decision making process, while for predictive models, it is often more informative to assess the inputs. In certain cases, particularly for human-driven decisions, disparate impact can be effective at identifying unintentional discrimination; however, for predictive models, disparate impact is inadequate. Even though the outcomes of a predictive model can be skewed across different groups, there is still a non-trivial possibility that the predictive model is not discriminatory. In fact, in a few cases, a predictive model can be engineered to produce favorable outcomes for members of a minority group. In such a situation, one might still observe a skewed outcome across groups despite favoritism towards members in the minority group. Because disparate impact doctrine does not get at the internal workings of predictive models, it can fail as a tool for assessing discrimination in predictive modeling.

2.6 Tools for Diagnosis

Given that disparate impact and outcome-based discrimination diagnosis tools are inadequate, it is important to develop technical tools and frameworks for diagnosing bias and performing comprehensive audits of the predictive modeling process. The goal of an audit framework for predictive models would be to quantify the model's

relative dependence on its inputs, and also represent the internal state of the predictive model in a manner that analysts can easily interpret. Specifically, one crucial goal for an audit framework would be quantify and reveal the weight that a predictive model assigns to protected attributes like race, gender, and sexual orientation. A predictive model that places a high significance on these attributes relative to other inputs would almost surely produce discriminatory effects in outcomes. The weights that a predictive model assigns to its inputs can be used as a way to reveal the functional workings of such a model. Hence, it is important to develop auditing frameworks that can robustly identify the weights a predictive model assigns to its inputs.

Overall, an auditing framework would aid analysts at government agencies, as well as other practitioners, to gain an in-depth understanding of the predictive models that they are building, or asked to investigate. We believe access to such diagnosis tools for predictive models is critical and necessary for progress in this domain. Consequently, we have focused the work of this thesis on developing such tools.

CHAPTER 3

RELATED WORK

3.1 Chapter Overview

Auditing predictive models for bias is becoming an increasingly popular area of focus for researchers in machine learning and statistics. However, the study of discrimination is not new, and dates back more than 50 years across a variety of fields. The difference between past studies of discrimination and the current emerging studies lies in the increasing use of predictive models for decision making. In the past, discrimination studies have mostly focused on an analysis of human-driven decision making; however, given the increasing deployment of predictive models across all areas, we arrive at a situation where studies of discrimination ought to be targeted at the predictive models. In trying to present an overview of emerging literature, we have chosen to focus on key topical areas and an overview of general techniques similar to those applied in this thesis.

3.2 Model Compression and Knowledge Distillation

A general strategy for improving model accuracy or performance in machine learning involves training ensembles of different models and 'averaging' them. However, ensembles of models are typically cumbersome and more difficult to interpret. Model Compression (knowledge distillation) aims to solve these issues by developing methodologies that use the input and output of a cumbersome ensemble model to learn a simpler and more interpretable model that is just as accurate as the ensemble model. This is exactly the problem that this thesis seeks to solve. Increasingly, predictive models used in industries like employment, housing, and banking are cumbersome and uninterpretable. To diagnose these models for bias, one needs to learn simpler and interpretable versions of these models that are also as accurate as the cumbersome ensemble model.

In machine learning, discriminative predictive models typically take the form

$$Y = f(x) \tag{3.1}$$

where Y is the output of interest which can be a measure of creditworthiness in banking or health riskiness in insurance. x is the input data used in building a predictive model. x can consist of attributes such as age, gender, income level, amongst others. In general, the model building process involves an optimization process that tries to find a 'good' function $f(x)$, such that given unseen data, x_{new}, generates accurate predictions $\hat{Y}$. The function $f(x)$ is exactly the predictive model that is the focus of this thesis. Consequently, model compression refers to the task of trying to find a simpler and interpretable function $g(x)$ that closely mimics $f(x)$ as much as possible.

Buciluǎ et. al. in their 2006 paper presented pioneering work on model compression [12]. Since 2006, researchers have steadily contributed to the model compression literature by demonstrating how different ensembles of models can be compressed into simpler forms. More recently, researchers have started to pay more attention to model compression due to their applicability to deep neural networks. Deep neural networks are a class of models that attempt to learn higher-level representations of

input data through repeated applications of non-linear transformations to the input data [40]. Deep neural networks tend to consist of multiple layers, millions of parameters, specialized hardware, as well as other architecture dependent intricacies. In general, deep neural networks can be quite cumbersome and unwieldy [33, 7]. The need to develop simpler versions of these models has led to a resurgence in model compression.

In [7], Ba et. al. investigate model compression for large deep neural networks. In their work, they show that 'shallow' and more manageable feed-forward networks can indeed learn to mimic large scale deep neural networks with similar performance. Ba et. al. also present a simple mimic-learning algorithm for compressing larger models using regression with a regularized L-2 loss function. The intuition behind the work by Ba et. al. follows from several works that show that simple neural networks can learn arbitrariliy complex functions. For example, in their paper on *Learning Polynomials with Neural Networks*, Andoni et. al. show that "for a randomly initialized neural network with sufficiently many hidden units, the generic gradient descent algorithm learns any low degree polynomial" [4]. The expressive ability of simpler and shallow neural networks makes them amenable to compressing large-scale neural networks. Similarly, Papamakarios, in his thesis, presents a generic framework for model compression using derivative information during the compression process [53].

3.3 Fairness, Accountability, & Transparency in Predictive Modeling

As the use of predictive models for automated decision-making continues to grow, researchers have begun to look at the issue of fairness and discrimination in data mining [28]. Increasingly, the emerging subfield around this issue is now commonly referred to as discrimination aware data mining, or fairness aware data mining [55]. The literature on fairness is broad, including works from social choice theory, game theory, economics, and law [61]. In the computer science literature, work on identify-

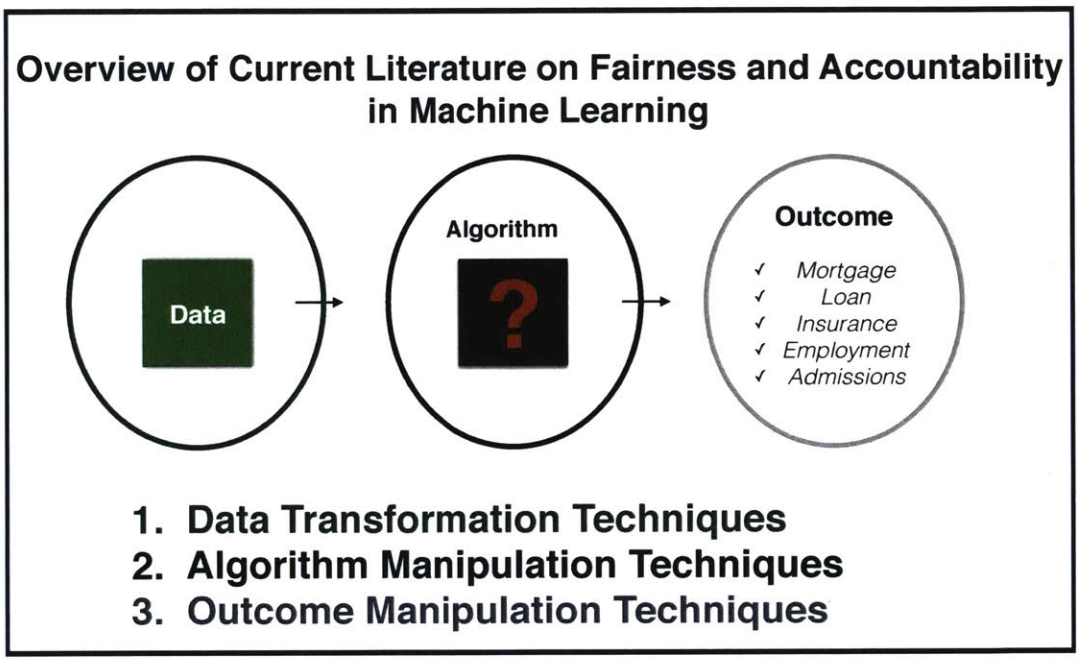


Figure 3-1: Figure showing a break down of the different emerging research on discrimination aware data mining. These works can be broadly classified into 3 broad categories: data transformation, algorithm manipulation, and outcome manipulation methodologies.

ing and studying bias in predictive modeling has only begun to emerge in the past few years. Romei, and Ruggieri in [61] give a comprehensive overview of the emerging field. More recently, a new class of studies has emerged. These studies seek to identify and correct bias throughout the predictive modeling process. In general, these works can be broadly classified into three categories: data transformation, algorithm manipulation, and outcome manipulation methodologies [61, 70].

For data transformation techniques, the input data to the predictive model is transformed as a means of quantifying bias in the data. In [13], Kamiran and Calders present a method for transforming data labels to remove discrimination. With their proposed method, a Naive-Bayes classifier is trained on positive labels, then a set of highly ranked negatively-labeled items from the protected set are changed to achieve statistical parity of outcomes. The modified data generated is then used to learn a 'fairer' model. Here, fairness refers to a more equitable distribution of positive outcomes among different classes. In [29], Friedler et. al. also present a data 'repair' methodology for transforming data into one that predictive models can hopefully

learn fair models on. In general, data transformation methodologies are typically seeking to learn fair representations of a dataset upon which less biased predictive models can be developed [61, 70, 29, 13].

As another class of methodologies, algorithm manipulation methods seek to augment the underlying algorithm in order to reduce discrimination. Algorithm augmentation is usually done via a penalty that adds a cost of discrimination to a model's cost function [70]. These algorithms typically add regularizers that determine the degree of bias acceptable. A seminal work in this area is the study by Kamishima et. al. in [36] where they quantify prejudice by adding a mutual information based regularizer to the cost function of a logistic regression model. Since the Kamishima et. al. work, more approaches that seek to change underlying cost functions with regularizers for statistical parity have emerged for other kinds of algorithms like decision trees and support vector machines [70]. Techniques presented in this area typically only work for one particular method like logistic regression or Naive Bayes, so the overall impact can be limited. Algorithm manipulation methods also assume that underlying predictive models are known, completely specified, and with well-behaved cost functions. It is often not the case that ensemble models have well-defined cost functions, so it can be difficult to apply algorithm manipulation techniques to ensemble models.

In the third approach, other studies have presented work that manipulates the outcomes of predictive models towards achieving statistical parity across groups. In [55], Pedreschi et. al. alter the confidence of classification rules inferred. In [13], Calders and Verwer transform the output probabilities of a Naive-Bayes model, and in [35] Kamiran et. al. re-label outcomes from a decision tree classifier to obtain parity across groups.

In a separate category, researchers have also sought to develop theoretical frameworks for guaranteeing fairness in a data mining process while balancing the model utility and predictive performance. In [24], Cynthia Dwork presents a task-specific metric for determining the degree to which individuals are similar given a model. Further, Dwork presents an algorithm for maximizing utility subject to a fairness constraint that similar individuals are treated similarly. Dwork ends by examining

the relationship between fairness and privacy as well as how techniques developed in the context of differential privacy can be applied to study fairness of algorithms.

3.4 Deconstructing Predictive Models

Given the need to ascertain fairness in algorithmic decision-making, several important questions remain unaddressed. As Zemel et. al. indicated in their work on learning fair representations, an important problem that has remained largely untackled "is understanding how to deconstruct a given classifier to determine to what extent it is fair" [70]. As noted by Zemel et. al, often predictive models are not interpretable so the underlying cost function can not be changed, hence, methodologies that can infer the underlying weights that a predictive model assigns to various input attributes would be particularly useful in quantifying fairness. Work on deconstructing classifiers has only started to emerge. Two notable works in this area include *Deconstructing binary support vector classifiers* by Ali. et. al. (2014), and *Adversarial learning* by Lowd et. al. [3, 43]. In [3], Ali et. al. present a comprehensive framework for deconstructing black box support vector machine (SVM) models. The deconstruction includes automatically learning the SVM's kernel and support vectors given training data. In [43], Lowd et. al. introduce the adversarial classifier reverse engineering algorithm for learning the underlying weights of a classifier. Their algorithm however only works for linear classifiers.

3.5 Bringing it all together

Here, we have provided an overview of the technical grounding and previous work related to the contributions made in this thesis. We have skipped an in-depth treatment of legal and policy related literature for providing a more technical overview. This decision was made to ensure that the reader is able to contextualize the technical aspects of FairML. We provide more in-depth discussions on legal theory, and discrimination in future chapters.

CHAPTER 4

FAIRML

4.1 Chapter Overview

In this chapter, we give a detailed overview of FairML, our end-to-end system for auditing predictive models. We begin with a general overview of the different modules that comprise FairML. We then provide a detailed treatment of our novel Iterative Orthogonal Feature Projection algorithm (IOFP), and the three remaining ranking algorithms implemented as part of FairML. We end the chapter with nuances, limitations, and ideal uses of FairML.

4.2 FairML Overview

The need for FairML arose from a variety of different motivations. However, the key motivation was to provide an easy-to-use tool for auditing predictive models. While a comprehensive audit of the entire data analysis and predictive modeling pipeline would be ideal, we narrow our focus to quantifying the relative significance (weights) of input attributes to a predictive model. Specifically, FairML provides analysts easy-to-understand plots showing the relative significance of input attributes to a

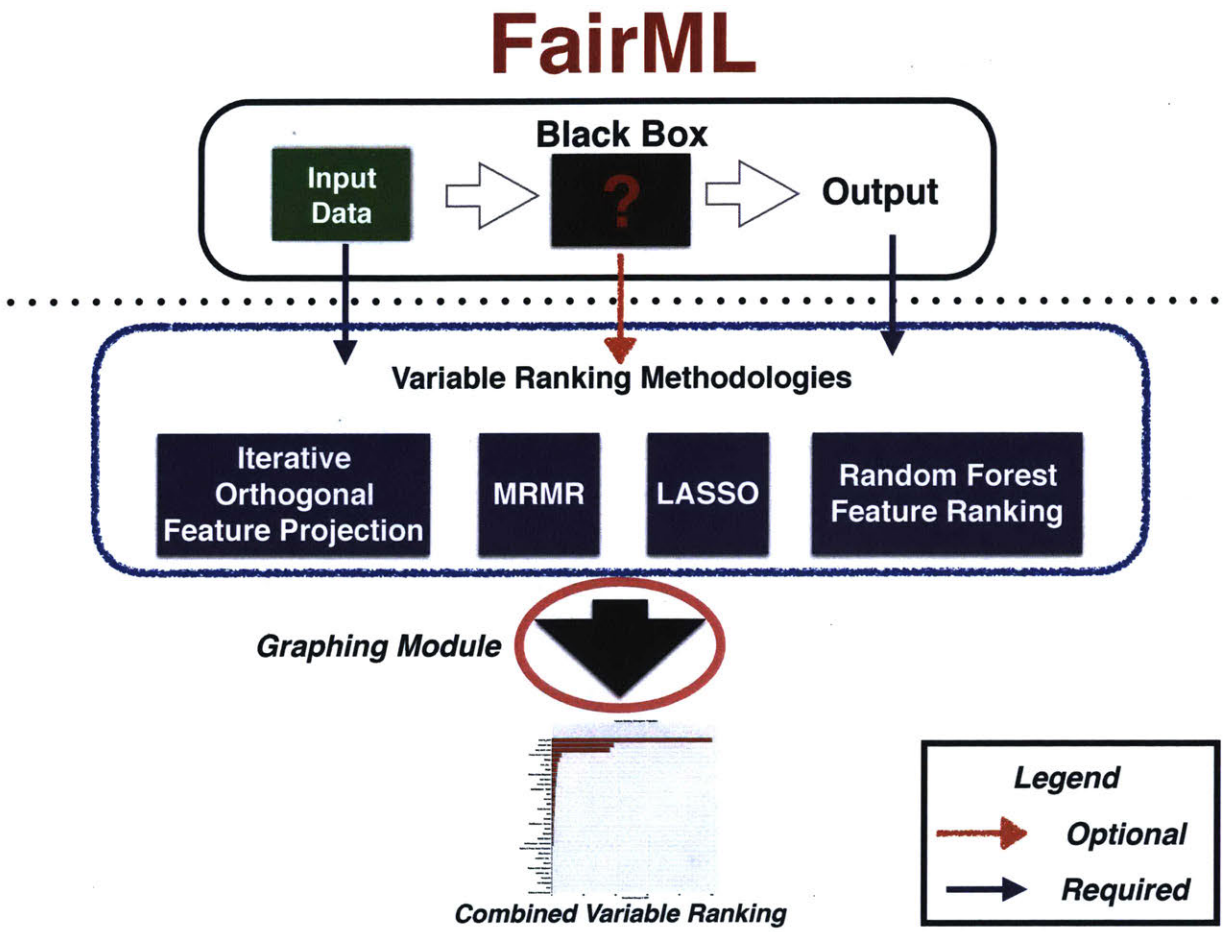


Figure 4-1: Figure shows the overall architecture of FairML. FairML consists of a data processing module (not pictured), a ranking module, and a graphing module. The final output of FairML are a series of bar charts showing the normalized variable importance among the different input variables. If the Black-Box can be iteratively queried, then the black-box can also be an input to the entire platform.

black-box predictive model. We have designed FairML for a large cadre of possible users, however, our primary target audience are analysts, policy makers, and other individuals without solid technical background in statistics and machine learning who would like to audit predictive models. Figure 4-1 shows the FairML architecture.

4.2.1 Data and Input Processing Module

The data and input processing module, not pictured in figure 4-1, is a crucial component and back-bone of the entire platform.[1] As part of the data processing module, we perform different kinds of tests on the input data. We perform tests to detect and handle any missing data, check for duplicate attributes as well as other data processing requirements. Each ranking algorithm implemented as part of FairML manipulates a matrix of input-output values, hence, the goal of the data and input processing module is to validate the input-output data and convert it to a matrix representation that is suitable for ranking.

4.2.2 Ranking Module

The ranking module is the major component of FairML. The ranking module consists of four algorithms: the Iterative Orthogonal Feature Projection Algorithm (IOFP), minimum Redundancy, Maximum Relevance (mRMR) feature selection, the Least Absolute Shrinkage and Selection Operator (LASSO) regression algorithm, and the Random Forest (RF) feature selection algorithm. Each algorithm has its strengths and weakness, however, together these 4 algorithms are uniquely suited for ranking the inputs to any predictive model. IOFP and LASSO can easily handle linear dependencies between input variables, while Random Forest and mRMR both handle non-linear interactions. Each ranking algorithm produces a score for each input attribute. We aggregate the four sets of scores obtained from the ranking algorithms into a combined score for each attribute. The final ranking of an input attribute is determined by the combined score obtained.

4.2.3 Graphing Module

The scores generated from the ranking algorithms serve as input to the graphing module. In the graphing module, each set of scores is normalized so that it takes on

[1] It is often repeated as a joke that most of the work involved in a data analysis task typically has to do with data cleaning and processing even before any analysis has taken place [41].

values between -100 and 100. For each set of scores, we then produce a bar plot showing the relative scores of the different input attributes generated by a particular ranking algorithm. Beyond this, we also produce a combined bar plot showing the aggregated score of each input attribute across the four ranking methodologies.

When an analyst deploys FairML for auditing a predictive model of interest, bar plots are generated corresponding to the combined ranking across all methodologies, and independent rankings from each methodology. The combined ranking provides the analyst with a robust ranking of the input variables. Further, the analyst can also examine the bar plots generated from the four underlying ranking methodologies to get a sense for how the importance of some variables change across the different methodologies.

4.2.4 Interfacing With the Platform

FairML is designed to be an extensible and modular platform that can be easily deployed by individuals of different technical backgrounds. There are two ways that analysts can interact with FairML. The first and more challenging way involves downloading all of the code and modules from our open source repository and running local versions of FairML on their own machines. This option is particularly suited to individuals who are more technically adept, and would like to make modifications to the underlying algorithms and modules. We anticipate that this option would be beneficial for statisticians, data scientists, and other researchers who are familiar with variable ranking methods and don't want to create their own version of FariML from scratch.

As a second option, we have developed a public facing web application derived from FairML where individuals can access FairML functionalities through their browsers. In this case, FairML generates a PDF report explaining the results of the entire audit for the analyst. For this option, analysts can simply upload their dataset to the application through the public facing interface, and quickly obtain a detailed report summarizing the results of the audit.

4.3 Variable Ranking Algorithms

4.3.1 Overview of Iterative Orthogonal Feature Projection Algorithm (IOFP)

One of the key technical contributions of this thesis is the proposed IOFP method for determining the relative significance of the inputs to a predictive model. Orthogonal projection as a feature selection tool is not new. In several previous works such as [19, 37, 65], orthogonal projection of input data was shown to be useful in selecting significant input variables. Here, we build on these previous works [19, 37, 65] to formalize an algorithm, based on orthogonal projection, for ranking input variables used in learning a predictive model.[2] Critically, given direct access to the predictive model of interest, IOFP can be used to iteratively query the predictive model with transformations of the input data to generate a ranking.

To reemphasize the power of being able to iteratively query the black-box predictive model, we recall one of our opening motivating questions. Suppose MIT decides to use a classifier to make admissions decisions. If MIT is then accused of discriminating on the basis of gender (here assumed to be male or female) in its admissions decisions, how would one go about investigating this? Hypothetically, we would find two groups of applicants that are similar in all characteristics except gender, and send in applications from these individuals to MIT's classifier for an admission decision. Given the outcomes, we then compare the average difference in outcomes between the two groups. If the difference in outcomes between the two groups is 'statistically significant'[3], then one can conclude that MIT's classifier is discriminating on the basis of gender. While performing the experiment described above is ideal for investigating any predictive model, it is often infeasible due to the difficulty of generating 'new'[4]

[2]This is one of the critical differences between our method and traditional PCA or even constrained PCA. Here, we manipulate the actual features or input data, in PCA, the original data is transformed to a new subspace. One common criticism (depending on your view) of PCA is that the new transformed features are difficult to interpret in terms of the original input data, i.e., one is unsure what each of the principal components is composed of.

[3]As usual, one has to be rigorous about what it means for a difference to be statistically significant. Given that this is a hypothetical example, we choose to gloss over such definition here.

[4]By 'new' we mean entirely new applications for the MIT classifier to classify. Traditional ex-

input data sources.

One way to reason about the experiment described above is that we seek multiple versions of the input data to a predictive model with which we can compare the outcomes across the multiple versions of input data. These different versions of the input represent transformations of the data where attributes of interest have been removed completely from the data. This reasoning suggests we need a data transformation procedure that 'removes' all the information with respect to an attribute of interest, and then compares the change in the predictive model's performance with a baseline case where the initial data is not transformed. For example, imagine we develop a model to predict the price of a house given the number of windows, and the square footage of the house. To know the importance of the number of windows to our predictive model, we can take the following steps:

- Obtain a baseline accuracy or performance, p_i, for the predictive model of interest using a training set D_t that includes both number of windows and square footage,
- Obtain a transformation of the data, D_T, such that all information characterizing number of windows has been removed,
- Obtain a new performance of the predictive model, p_t, on transformed the data set D_T
- The significance of number of windows for the predictive model developed is thus $k * |(p_i - p_t)|$ where k is a constant across all variables.

To obtain the model's dependence on square footage, we repeat the above steps for the variable square footage. Now, if we have 10 different input variables, we perform 10 different transformations in order to obtain the significance, $k * |(p_i - p_t)|$, for each variable. The iterative strategy described above underlies our IOFP algorithm.

Now, the key question involves finding a suitable transformation for input data D_t that can remove all information with respect to a variable of interest. Before

perimentation in this manner is ideal, but often costly and time consuming. In an ideal world, we would want a low-cost alternative that captures the essence of experimentation.

proceeding, we restate the question more formally. Given an n x k data matrix, X, that can be decomposed into $\vec{x_1}, \vec{x_2}, \ldots, \vec{x_k}$ column vectors, we seek a transformation T_i for a given vector $\vec{x_i} \in X$, such that $T_i(X)$ produces a transformed matrix X_{-i} where X_{-i} *contains no information*[5] about $\vec{x_i}$.

A tempting transformation that seems to achieve the desired effect is one that just removes $\vec{x_i}$ from X, producing a n x $k-1$ matrix. This transformation would be appropriate should all the columns of X be linearly independent. Another way to think about: if the original columns of the input data matrix X are all linearly independent, then a simple transformation that takes out the attribute of interest $\vec{x_i}$ is valid. However, input attribute independence is almost never the case for most real-world data sets [45, 27]. In general, real-world data from industries like insurance, banking, and employment consists of variables like age, gender, socio-economic status, and education level that are all highly correlated with one another [45, 27, 62]. Hence, simple feature removal, a common practice in machine learning, will not work here as a suitable transformation.

In IOFP, we propose using orthogonal projection to transform each vector $\vec{x_i} \in X$ to obtain a transformed data where all linear[6] information with respect to $\vec{x_i}$ has been removed. This procedure produces an n x $k-1$ matrix X_{-i} where $\vec{x_i}$ has been removed, and each of the columns of, X_i, is orthogonal to $\vec{x_i}$. With the iterative orthogonal projection routine, we can generate multiple copies of the input data. As described in the housing prices example, the significance of an attribute can be measured as the change in accuracy of the predictive model when the model is evaluated on a transformed input. If an attribute is relatively significant for a predictive model, then the attribute has a strong effect on the predictive performance of that model. However, attributes with low significance will have little to no effect on the performance of the predictive model on transformed inputs.

[5]the use of the phrase *contains no information* here is quite informal. In some sense, by contains no information, we mean that it should be difficult to obtain $\vec{x_i}$ from X_{-i} by taking *linear* combinations of its columns.

[6]Orthogonal projection is a linear transformation, so it is incapable of eliminating non-linear information

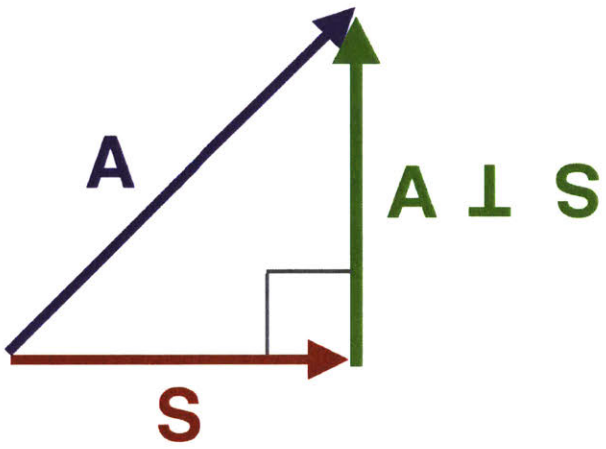


Figure 4-2: Depiction of Orthogonal projection of A onto S.

4.3.2 Underlying Theory IOFP

In this section, we describe the underlying theory behind IOFP, starting with a general overview of orthogonal projection transformation. We then provide a detailed algorithmic formulation of IOFP, and close with implementation details.

Orthogonal Projection

The orthogonal projection is an example of a linear transformation. One can think of the orthogonal projection as a mapping that transforms an input vector into one that is orthogonal or perpendicular to the subspace on which the projection is defined. Figure 4-2 demonstrates the orthogonal transformation in two dimensions. Intuitively, an orthogonal transformation of a vector $\vec{r}$ using a projection, P, produces a new vector, $\vec{s}$ such that no linear combination of vectors in the subspace from which P is derived can produce $\vec{s}$. Two vectors whose inner product is zero are orthogonal to one another.

In Figure 4-2, the vector $A_{\perp}S$ shown is the component of the vector A that is orthogonal to S. Hence, orthogonal projection is the process of obtaining $A_{\perp}S$ given A and S. As expected, the concept of orthogonal projection maps to higher dimensions. Having provided a brief background on orthogonal projection, we formalize our proposed IOFP ranking methodology in the next section.

Outline of IOFP

Algorithm 1 outlines the process of obtaining a transformed input data where the linear effect of a given attribute has been removed. Algorithm 1 outlines the step-by-step process of transforming input data given a specific attribute using orthogonal projections. With algorithm 1, one can transform each input attribute in a data set iteratively to obtain different copies of the original dataset where each attribute has been 'removed.' The change in performance of a predictive model on these multiple copies of the original data set then represents the significance of each of the different input attributes to the predictive model.

Algorithm 1 Linear Feature Transformation Algorithm

INPUT: An n x k data matrix X_{pre} that can be decomposed into $\vec{x_1}, \vec{x_2}, \ldots, \vec{x_k}$ column attribute vectors, where:
n = number of samples,
k = number of features or attributes being considered, and
$\vec{x_1}$ = current attribute of interest.
OUTPUT: An $n \times k-1$ transformed matrix X_{new} that can be decomposed into $\vec{x_2^*}, \vec{x_3^*}, \ldots, \vec{x_k^*}$ where each vector $x_i^* \in X_{new}$ is orthogonal to current feature $\vec{x_1}$.

Delete current vector $\vec{x_1}$ from X_{pre} returning X_{del}
Initialize an $n \times k-1$ vector X_{new}
for each attribute vector $\vec{x_i}$ in X_{del} **do**
  obtain x_i^*, the component of $\vec{x_i}$ that is orthogonal to current feature vector $\vec{x_1}$
  where $x_i^* = \vec{x_i} - (\frac{\vec{x_1} \cdot \vec{x_i}}{\vec{x_1} \cdot \vec{x_1}})\vec{x_1}$
  join x_i^* column wise to X_{new}
end for
Return X_{new}

In algorithm 2, we introduce the IOFP ranking methodology. Algorithm 2 provides a step-by-step procedure for determining the significance of a set of input variables to a predictive model. For the general framework presented we assume that the black-box can be queried iteratively to obtain a change in predictive performance given each transformation of the input data. This requirement is not always fulfilled. In cases where access to the black-box is not available, one can learn equivalent representations of the black-box through model compression. We now proceed to implementation details of the IOFP as part of FairML.

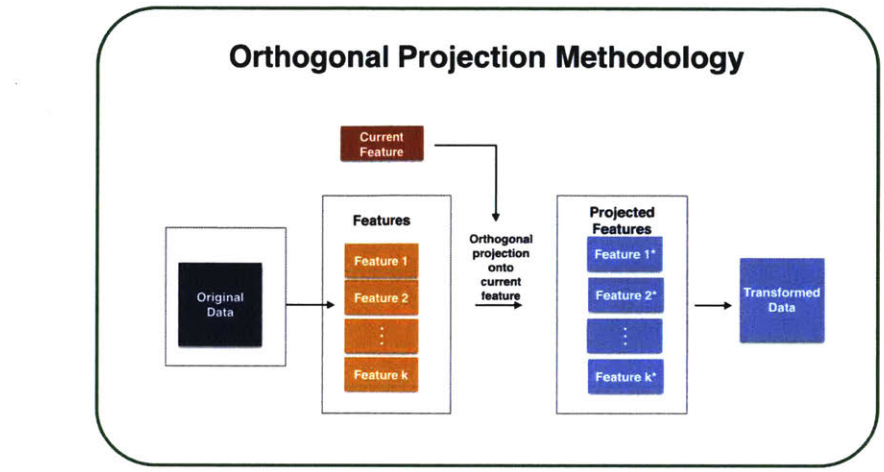


Figure 4-3: An overview of our proposed orthogonalization process.

Algorithm 2 General Ranking Framework

INPUT: An n x k data matrix X that can be decomposed into $\vec{x_1}, \vec{x_2}, \ldots, \vec{x_k}$ attribute vectors

Output of the black-box algorithm is y

Initial baseline predictive performance b of the black-box algorithm.

OUTPUT: Vector $R \in \mathbb{R}^k$ of predictive dependence of the black-box on each input feature

for each feature $\vec{x_i}$ in X **do**

Combine non-linear transformations (log, polynomial, exponential etc) of each feature $\vec{x_i}$ with vector X as X_{poly}

obtain X_{new} from the Feature Transformation Algorithm given X_{poly}

obtain black-box's predictive performance (MSE or classification accuracy) given X_{new} as b_{new}

predictive dependence on $\vec{x_i} =\mid b - b_{new} \mid$

store $\vec{x_i} =\mid b - b_{new} \mid$ in R

end for

Return R

Implementation Details

We implemented IOFP as described in section 4.3.2 as part of the ranking module for FairML using the Python programming language [69]. The Python programming language includes modules that are well[7] optimized for easily performing matrix manipulations. As part of the module, users can specify whether they want to use the simple linear transformation procedure or the more general non-linear basis expansion procedure. Further, users can specify the sets of functions that they wish to include for transforming the input data.

4.3.3 minimum Redundancy, Maximum Relevance Feature Selection (mRMR)

The mutual information between two random variables is a measure of how much information one can learn from one variable given the other. The mutual information measure has been particularly important in feature selection algorithms. Unlike correlation, the mutual information between two variables is more general and can measure potentially non-linear information between two variables. The mRMR feature ranking methodology is built upon the mutual information between an input feature and a target variable of interest.

To properly contextualize the discussion about mRMR, we provide the formal definition of mutual information. Given two random variables x, and y, with marginal and joint densities $p(x)$, $p(y)$, and $p(x,y)$, the mutual information between x and y is defined as follows:

$$I(x;y) = \iint p(x,y) log \frac{p(x,y)}{p(x)p(y)} dxdy.$$

For the discrete analog of the mutual information, we sum over the different values of x, and y instead of integrating. Because mutual information measures dependence between two variables, it is particularly suited for feature selection and ranking. To

[7] As always with programming languages, different people have their favorites. We believe python was well suited for the task described here.

rank a set of input variables that are most useful for learning a particular target variable, one can simply rank the input variables by their mutual information with the target variable.

In terms of feature ranking for a black-box predictive model, we can also leverage mutual information in determining the significance of the inputs to the model by ranking the inputs based on their mutual information with the output of the black-box predictive model. Consequently, input variables that have a high mutual information score are more significant to the black-box than those that do not. One potential problem with this ranking procedure is that it does not take into account dependencies among input variables. If two input variables both have high mutual information with one another, these two variables are also more likely to have high mutual information with the target. However, it could be the case that only one of the two input variables is significant to the predictive model in question. Hence, selection based on mutual information with the output of the black-box is subject to redundancy.

Researchers have formulated several methods for overcoming the aforementioned redundancy. One of the more successful approaches is the minimum Redundancy, Maximum Relevance (mRMR) approach. mRMR is an iterative forward feature selection technique. At each step, mRMR selects an input feature such that the feature exhibits a high mutual information with the target variable but a low total sum mutual information with all the previously selected variables. We now provide a brief overview of selection criterion.

Given an n x k input data matrix, X, with features indexed as, X_i, where i ranges from 1 to k, in the first step of mRMR, the feature X_i which has the highest mutual information with the target T is selected. The set of selected features is X_S. After each step, a selected feature is added to X_S. For the first step, we add the feature with the highest mutual information with target, T[8], to X_S. For subsequent steps, X_S is updated given the input variable that maximizes the following criteria:

$$s_k = I(X_k, T) - \frac{1}{\mid X_S \mid} \sum_{X_i \in X_S} I(X_k, X_i)$$

[8]Note here that target T for our application is the output of a black-box algorithm.

where T is the target variable, and $I(X_k, T)$ is the mutual information between X_k and T [56, 22].

The mRMR algorithm has been shown to be very effective in computational biology, particularly for tasks that involve selecting a subset of significant variables from a larger one. Here we leverage the score generated by the mRMR algorithm in order to rank the different inputs to the black-box model. We have included the mRMR ranking methodology as part of FairML because its use of mutual information as a primary metric makes it likely to detect non-linear interactions among input data. A more detailed and technical presentation of mRMR is presented in [56] and [22].

4.3.4 Implementation Details: mRMR

The mRMR ranking module in FairML was programmed in R leveraging the mRMRe package in the CRAN package manager [20, 67]. The mRMRe package is a general purpose package for learning predictive networks from high-dimensional genomic data. As part of the core modules of the package, the developers implemented the mRMR ranking algorithm described here, which we leverage to score and rank the inputs to a predictive model.

4.3.5 LASSO

Linear regression has been referred to as the 'workhorse' of statistics and data analysis [50]. Linear regression constitutes a large class of models where a target variable of interest is modeled as function of one or more explanatory variables [5]. Given data consisting of input-output pairs $\{y_i, X_i\}^n$, where X_i can be decomposed as $x_{i1}, x_{i2}, \ldots, x_{id}$, for d explanatory variables, then a linear regression model can be specified as:

$$y_i = \beta_1 x_{i1}, + \cdots + \beta_d x_{id} + \epsilon \tag{4.1}$$

where ϵ refers to noise. Linear regression models are used widely to represent the relationship between variables and also to make forecasts. These models are used across fields like economics, statistics, medicine, and others.

Linear regression can also be used to quantify the strength of the relationship between two or more variables. For example in equation 4.1 above, the magnitude of a β value in the equation can typically be used to quantify the strength of the relationship between the explanatory variables and the target variable. Imagine we want to obtain an estimate of the relationship between the price of a house and explanatory variables such as number of windows in the house, and square footage of the house. We can estimate a regression model with these variables where the price of the house is the target variable, and obtain regressions coefficients βs as shown in the table 4.1.

Variable	**Beta Value**
Number of Windows	0.4
Square Footage	5.86

Table 4.1: Example demonstrating linear regression coefficients

From table 4.1, one can *reasonably*[9] The use of regressions coefficients as a measure of variable significance underlies the LASSO regression.

Overview of the LASSO

LASSO stands for Least Absolute Shrinkage Selection Operator. LASSO was introduced by Robert Tibshirani in his landmark paper, "'Regression Shrinkage and Selection via the lasso." Tibshirani motivates his paper by saying:

> There are two reasons why the data analyst is often not satisfied with the OLS [Ordinary Least Squares] estimates. The first is prediction accuracy: the OLS estimates often have low bias but large variance; prediction accuracy can sometimes be improved by shrinking or setting to 0 some coefficients. By doing so we sacrifice a little bias to reduce the variance of the predicted values and hence may improve the overall prediction

[9]There is a whole world on the exact conditions under which one can make conclusions from regression estimates. We choose to gloss over this. In general, given ideal specifications of the regression model, one can often conclude that the absolute value of β corresponds to the strength of the relationship between an attribute and the target of interest.

> accuracy. The second reason is interpretation. With a large number of predictors, we often would like to determine a smaller subset that exhibits the strongest effects. [68]

As noted above, the LASSO is specifically designed to identify significant explanatory variables. We take advantage of this property in FairML to rank the input variables of a model by their LASSO coefficient estimates. We formalize the LASSO estimation problem below:

Given y_i, an outcome variable of interest, and $x_i := (x_1, x_2, x_3, \ldots, x_p)^T$ a vector of explanatory variables for sample i in a data set with N total samples, then the LASSO objective function can be specified as:

$$\min_{\beta_0,\beta} \frac{1}{N} \sum_{i=1}^{N} (y_i - \beta_0 - x_i^T \beta)^2$$
$$subject \quad to \tag{4.2}$$
$$\sum_{j=1}^{p} |\beta_j| \leq t.$$

In equation 4.2, t 'regulates' the shrinkage of the coefficients obtained from solving the objective function. The L-1 penalty on the βs tends the β values to zero, so insignificant explanatory variables are more likely to have zero coefficient values. Significant explanatory variables will tend to have coefficient values whose absolute value is greater than zero. There have since been several extensions to the formulation above that seek to improve the stability of LASSO estimates; however, the underlying theory and motivation presented above still ground several of these variations [48, 71, 49].

In FairML, we use the value of the LASSO coefficients, βs, to rank the variables used in building a predictive model. We consider the input variables to the predictive model as our explanatory variables, and the output of the black-box as the target variable. We then estimate the LASSO regression coefficients for using these explanatory variables. The ranking of the input variables is thus determined by the absolute value of coefficients estimated from the LASSO regression model, which signifies the

strength of the relationship between the output of the black-box predictive model and the explanatory variables.

4.3.6 Implementation Details: LASSO

For the LASSO ranking in FairML, we leverage the implementation provided through the popular Scikit-Learn package in python [54]. We leverage a particular variant of the LASSO algorithm that performs stability selection with several bootstrap samples of the data in order to obtain robust estimates of the regression coefficients.

4.3.7 Random Forest Feature Selection (RF)

The Random forest algorithm has become one of the mostly widely used predictive models in the data science community [25]. Random forests are an ensemble learning method that combine several decision trees typically in a majority voting rule to form an ensemble model. In this section, we present a general and relatively brief overview of random forests. We refer readers to sources cited in this section for a more in-depth reference on random forests.

Random forests combine several decision tree models learned on bootstrap samples of the input data into an ensemble model. The 'random' in random forest comes from the fact that each decision tree in the 'forest of trees' is learned from on a random subset of the data. Beyond learning on random subsets of data, a certain class of random forests can also be constructed such that for each split in a decision tree, the variable chosen to split the input data is selected from a random subset of the original and complete set of input attributes. Each of the trees in random forest is trained using one of the popular tree learning algorithms such as ID3, CART, C4.5, CHAID, MARS, depending on the algorithm's suitability to the problem at hand. The CART tree learning algorithm is typically used in most random forest implementations [51, 11]. The use of random subsets of data and random subsets of features for splitting helps to reduce the variance associated with predictions and also helps to prevent overfitting [11].

Random forests are particularly useful for determining variable importance [42]. There are a variety of ways to determine variable importance in random forests, however, we present two example ways here. In the first way, one can use the depth of an attribute in a decision tree as a measure of importance. Attributes that are higher in the tree can be thought of as affecting a significant portion of the total samples used to learn the tree model. Hence, such measure of importance, tries to quantify, "the expected fraction of the samples [that a particular attribute] contribute[s] to" in the tree model derived [64].

For the second method of determining variable importance, one makes use of permutation to quantify variable significance. To do this, one obtains a baseline accuracy of a given tree on samples that were not used for learning the tree. Next, for each attribute, one randomly permutes the values of that attribute in the out of sample cases, then estimate the accuracy of tree given on the out of sample cases where the attribute of interest has been randomly permuted. This procedure is performed for all trees in the forest, and the change in performance is averaged across all trees. If an attribute is highly significant, then the change in accuracy would be high. The methodology described is popularly known as mean decrease in accuracy variable importance estimation [42].

The overview of random forests presented here is quite brief; however, random forests are a rich ensemble model whose interesting properties have been subjects of doctoral dissertations by themselves. If the reader is curious about the peculiarities of random forests beyond the general overview given here, we encourage the reader to take a look at several of the citations in this section.

4.3.8 Implementation Details: Random Forest

For the random forest ranking in FairML, we leverage the scikit-Learn package in the python programming language. The random forest implementation as part of scikit-learn is quite robust, and includes several parameters that can be easily tuned. Further, obtaining feature importances from each tree is also straightforward through the scikit-learn interface [54].

4.4 Notes on FairML

4.4.1 Linearity and Non-Linear Feature Projection

One valid limitation of our IOFP method is that it is better suited for auditing linear predictive models. We address this limitation in two ways. First, we transform the input data using an expanded basis that includes several non-linear and interaction parameters, so that the IOFP method can learn the significance of these non-linear expansions. The non-linear transformations implemented as part of FairML include polynomial transformations, logarithms, exponentials, cosines, sines, and first order interaction terms between different features. Secondly, we include the mRMR and Random Forest ranking methodologies as part of FairML because both these methods are uniquely suited for characterizing and quantifying non-linear interactions in input data.

4.4.2 Interpreting Non-Linear Functions

One particularly challenging aspect of this work has been in thinking about how to interpret non-linear transformations of input data. If an algorithm takes as input age, and the algorithm then applies a non-linear transformation to this input, multiple times, as a neural network would, and the resulting transformation turns out to be a significant feature, does that mean the original input is significant? How does one interpret the final non-linear transformation? Further, if one finds that an interaction term between two inputs is significant, how does one present that in an interpretable way to analysts? These sets of questions are more subtle and require more time for an adequate resolution.

4.4.3 The Broader Platform

With FairML, we hope to open up a discussion around complicated and cumbersome predictive models that are being increasingly used in industries like banking, insurance, and employment services. With FairML, analysts can easily and quickly

perform routine audits to ensure that the predictive models that they are building is not overly dependent on protected attributes. With an auditing system in place, practitioners can now confidently deploy predictive models for deriving insights and automating tasks.

CHAPTER 5

EVALUATION

5.1 Chapter Overview

In this section, we present the results of simulation experiments performed to test the four ranking algorithms implemented as part of FairML. Through these simulations, we investigate the performance of the four ranking methodologies under different conditions to understand how the rankings obtained might differ with changing conditions. Primarily, we test the performance of the black-box models under different amounts of noise in a predictive model. Further, for the IOFP method, we also test the impact of direct access to a black-box. Through these simulations, we observe that when a predictive model is available to be queried iteratively, then the performance of the IOFP methodology improves by up to 15 percent.

5.2 Overview of the Simulation Experiments

To test the different ranking methodologies when the black-box algorithm is not accessible, we generate linear functions from synthetic inputs combined in a pre-specified manner and then pass this input data and output from these functions to

FairML for ranking.

We construct functions of the form $y = \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \epsilon$, where $k = 20$, and ϵ represent noise added to the function. Each x_i is drawn independently from a normal distribution with zero mean and variance 1, and each β is drawn uniformly from a range (1,50).

With this, we can also add in nuisance input variables whose β values are zero. The number of nuisance variables added to the collection of inputs is also drawn from a uniform distribution from (0,10). Hence the minimum number of inputs is 20, and the maximum is 30.

Given the above procedure, we generate functions randomly synthetically and then sample data from these functions to feed as input to FairML. For each black-box function constructed, we construct a data set of 10000 samples for FairML.

Given functions generated in the manner described, we obtain a true ranking for our 'black-box' algorithm by ranking the inputs x based on the values of the β. We then produce a new ranking from each of our ranking methodologies, and calculated the Kendall-Tau rank correlation coefficient of the ranking produced from each method with the true ranking for over 10000 different functions.

5.3 Standard Situation: Black-Box not Accessible

In this section, we present the primary results of our evaluation, which involves varying the amount of noise in a black-box model and observing the corresponding rankings produced by FairML. In doing this analysis, we hope to determine the robustness of the rankings implemented as part of FairML. An auditing platform that is robust to varying levels of noise is desired because we expect our black-box platform to be used in different industries across a variety of applications.

5.3.1 FairML Rankings: No Noise

We begin with an initial analysis of the rankings produced by FairML for a black-box that does not contain any noise. In this case, we set ϵ to zero and then compare the

rankings obtained from FairML to the true rankings.

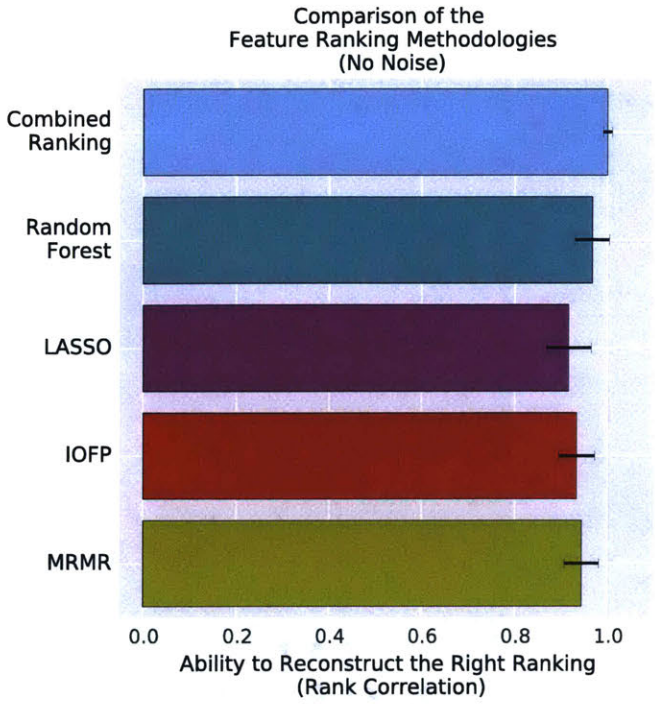


Figure 5-1: Figure shows the performance of the rankings produced by FairML compared to the true ranking. The bar represents the average rank correlation coefficient between the ranking generated by each method and the true ranking of the black-box.

From figure 5-1, we see that each of the different methods perform well when there is no noise added to the black-box. The higher the rank correlation, score, the better the performance of a particular ranking methodology. One relatively interesting point to note is that the combined ranking generated by FairML produces the true ranking in the black-box when there is no noise. We also see that each of the different methods falls within similar ranges for the rank correlation produced. From figure 5-1, we can conclude that under a linear setting, the outputs of FairML correspond exactly to the ranking of the underlying black-box.

5.3.2 FairML Rankings: Low Noise

In this and subsequent sections, we now test FairML rankings produced for a black-box with implicit noise. For a low noise setting, we keep the same specifications as

described earlier; however, now we draw ϵ from a normal distribution with zero mean and unit variance. This setting constitutes very little noise.

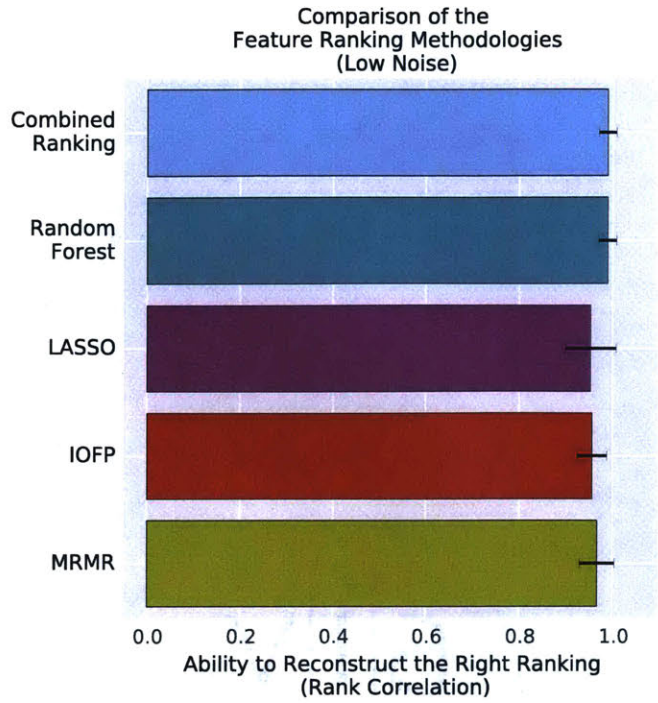


Figure 5-2: Figure shows the performance of the rankings produced by FairML compared to the true ranking under low noise

Compared to the results obtained in figure 5-1, we see that results shown in figure 5-2 indicate a slight decline in the performance of the ranking methodologies. For this case, we also observe that the combined ranking generated is almost exactly the true underlying ranking of the black-box. From the results presented, it is difficult to make claims about the superiority of any one ranking methodology given the relative closeness in the rankings obtained.

5.3.3 FairML Rankings: Medium Noise

Now, we increase the amount of noise implicit in the black-box and replicate the process described above. We draw ϵ from a normal distribution with mean 0 and variance 8. This setting constitutes a medium level of noise.

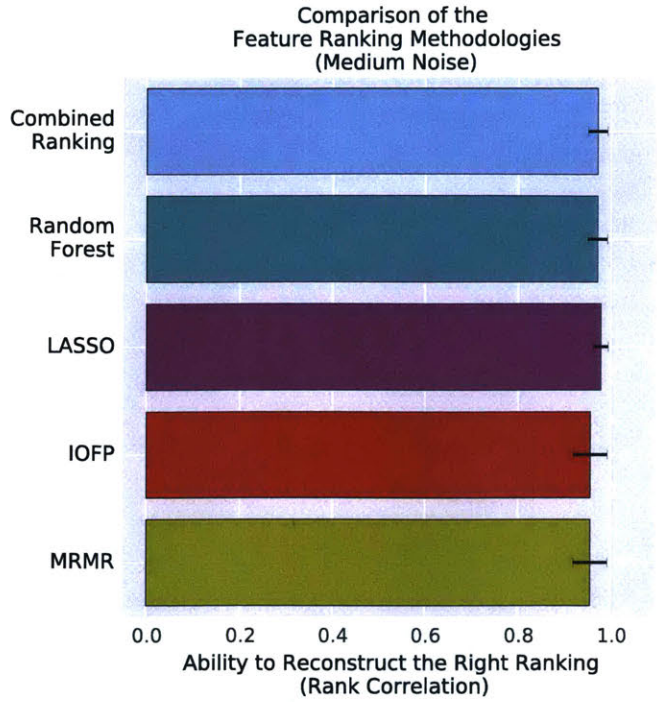


Figure 5-3: Figure shows the performance of the rankings produced by FairML compared to the true ranking under low noise

For this case, figure 5-3, we also observe that the combined ranking generated is almost exactly the true underlying ranking of the black-box. Still, it is difficult to make claims about the superiority of any one ranking methodology given the relative closeness in the rankings obtained. From figure 5-3, one can reasonably conclude that these ranking methodologies are robust to modest amount of noise.

5.3.4 FairML Rankings: High Noise

Now, we increase the level of noise that is added to the the output of the black-box for ranking by FairML. To do this, we draw ϵ from a normal distribution with mean 10 and variance 100. This setting constitutes a high level of noise for the problem specified.

We immediately notice that the overall rankings across all four ranking methods drops. As expected, this can be attributed to the level of noise included. In this

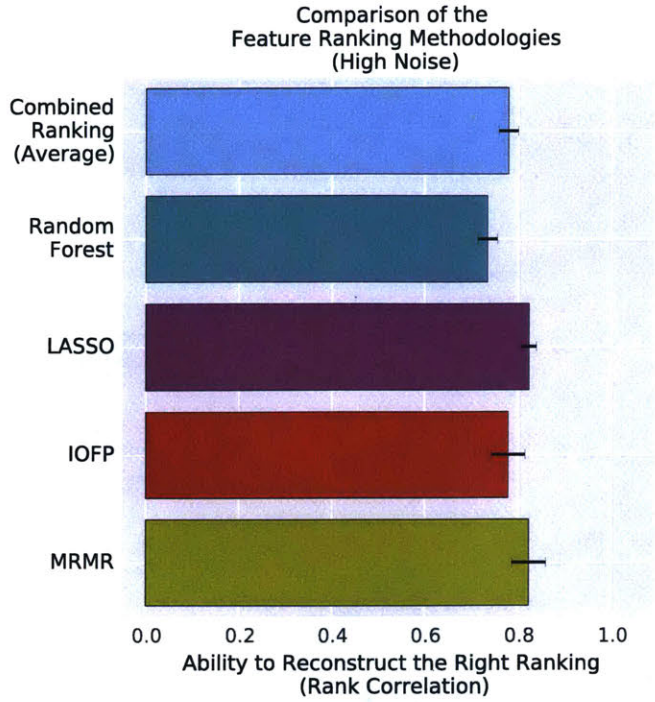


Figure 5-4: Figure shows the performance of the rankings produced by FairML compared to the true ranking under low noise

case, we immediately observe that the LASSO methodology seems to be the best performing of the different methodologies, though by a small margin. This confirms previous research noting that the LASSO methodology can be robust to noise, and particularly suited for determining the significance of input variables in a model.

In each of the previous tests performed, we note that the combined ranking improves upon any one individual ranking. However, this is not the case for the results shown in figure 5-4. Since the combined ranking is a simple average of the previous rankings, the combined ranking is susceptible to outliers. The results of our simulations in this section suggest that in the presence of a high amount of noise, instead of a simple average, more weight should be placed on the LASSO rankings.

Having tested FairML extensively across different levels of noise, we see that the platform is robust against implicit noise in a black-box. This suggests that the rankings provided by FairML can indeed be used as a means of investigating the strength of a predictive model's association with its inputs.

5.4 Black-Box Accessible

For our second sets of tests, we constructed functions in a similar manner to the approach described in the previous section; however, here, we made it possible to iteratively query the functions generated with transformations of the input data. In doing this, we test one of the major hypothesis put forth in this work, i.e., direct access to a predictive model allows one to get a better ranking of the internal state of the predictive model.

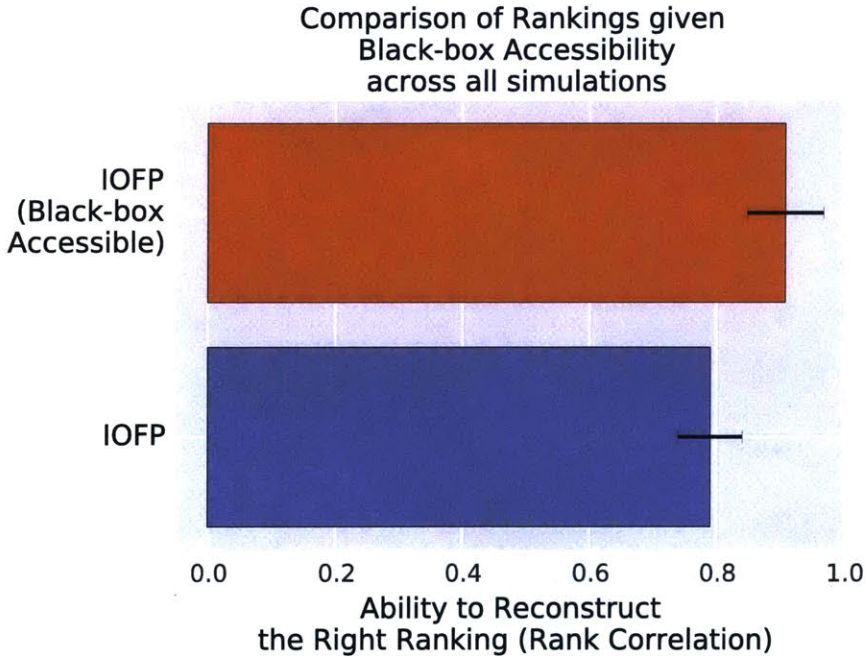


Figure 5-5: Figure shows the performance of the IOFP method for the two cases tested. We note the 15% improvement in performance when direct access to the black-box is allowed.

In this case, we immediately note that the performance of the orthogonal projection algorithm improves significantly. There are several reasons for this. In the cases where we didn't have direct access to the black-box algorithm, we had to learn our own versions of the black-box with which we then performed iterated transformations. Under such circumstances, it is virtually guaranteed that the new versions of the black-box model being learned are not perfect matches with the true black-box

model. With direct access to the black-box one can get a true value of the change in predictive dependence of the algorithm given repeated transformations of each input to the black-box, thereby simulating an experimental scenario that ought to theoretically learn the true ranking of each input variable.

CHAPTER 6

FAIRML AUDITS OF DIFFERENT DATA SETS

6.1 Chapter Overview

In this section, we show how FairML can be used to audit predictive models. In our case study, we have millions of credit card transactions data from a major bank in Europe. The Bank has developed internal models for calculating credit card limit and the probability that a customer will default on a loan. Both these models are critical in determining the bank's revenue, and also determine how the bank treats its customers. Suppose regulators in the Bank's region get complaints that the bank is discriminating on the basis of gender, one can use FairML to quantify how dependent the bank's models are on gender. As we show here, we find that both the bank's credit limit, and probability-of-default models places little weight on gender. An analyst using our platform to audit the bank's models will ultimately absolve the bank of any wrongdoings.

6.2 Auditing a Bank's Credit Limit and Probability of Default Algorithms

In this section, we assume that an analyst needs to perform routine audits of a bank's credit limit and probability-of-default models to determine whether the bank's algorithm is dependent on Gender for making these decisions.

6.2.1 Overview of Data Set

Our dataset consists of demographic information for 400 thousand customers of a large bank. For each of these individuals, the bank has internal predictive models to calculate the credit limit and future probability of default on a loan. These two factors are very critical for the bank's survival as a business, and fundamental to how the bank treats its customers. In this analysis, we have access to a set of input variables that the bank uses in its models, and the output of the bank's predictive models.

Input Data Attributes
Income
Customer Age
Asset
Marrital Status
Gender
Employment Status
Education Level

Table 6.1: Table listing the input attributes available for audit of the Credit Limit and the Probability of Default Algorithms

Outputs of the Black-Box
Credit Card Limit
Probability of Default on a Loan

Table 6.2: Table listing outputs of the black-box algorithms

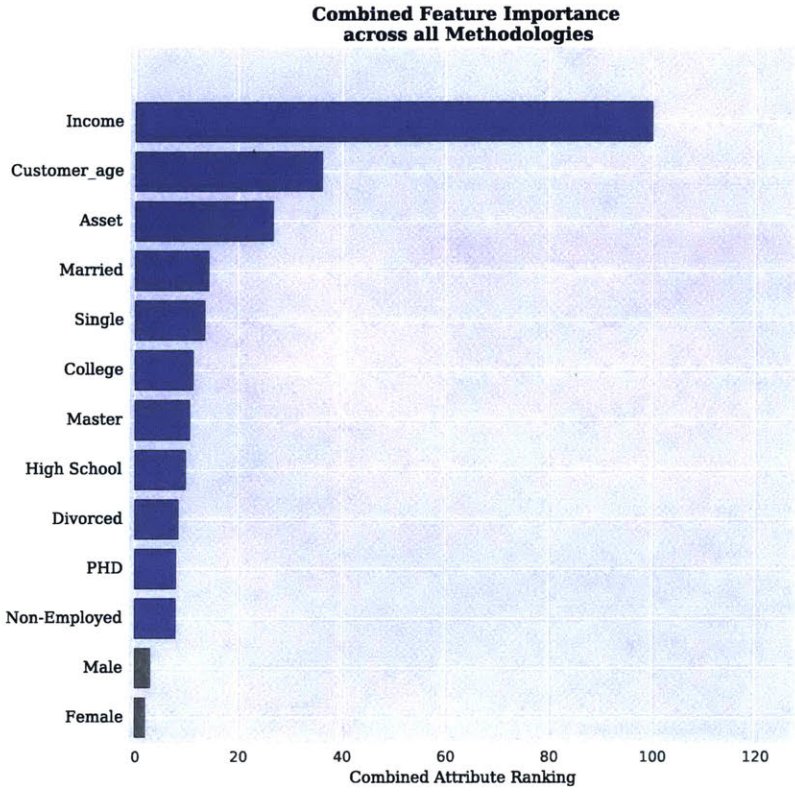


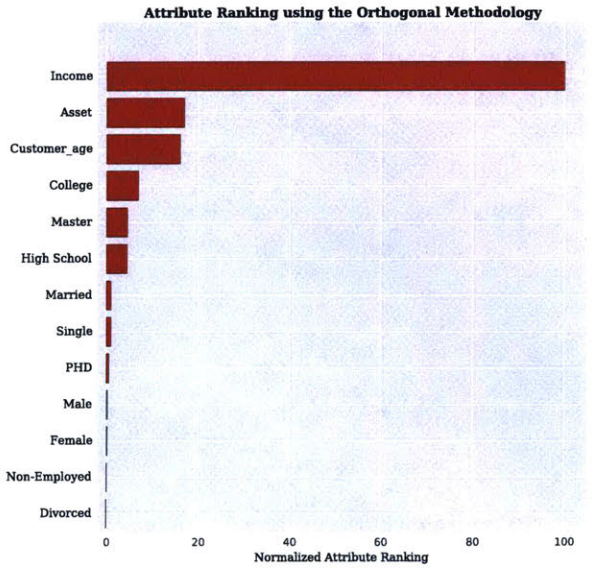
Figure 6-1: FairML output showing the combined ranking for each of the different inputs on the bank's credit limit algorithm. As we see in the combined ranking, the gender variables (male and female), rank dead last compared to all of the other variables.

6.2.2 Credit Limit Audit

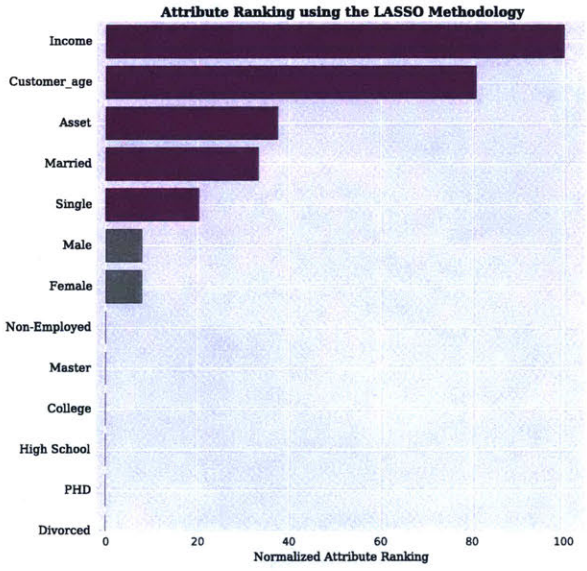
To perform the audit, we pass the input data available and the two target outputs to FairML. At the end of this analysis, FairML produces five different bar plots that characterize the dependence of the bank's predictive model on each of its inputs. We begin here with the results for the credit limit algorithm as shown in figure 6-1.

Figure 6-1 shows the combined ranking from all four ranking methodologies implemented as part of the FairML. As we see in the combined ranking, the gender variables (male and female), rank dead last compared to all of the other variables. We also see that the top ranked variables are income, customer age, and asset.

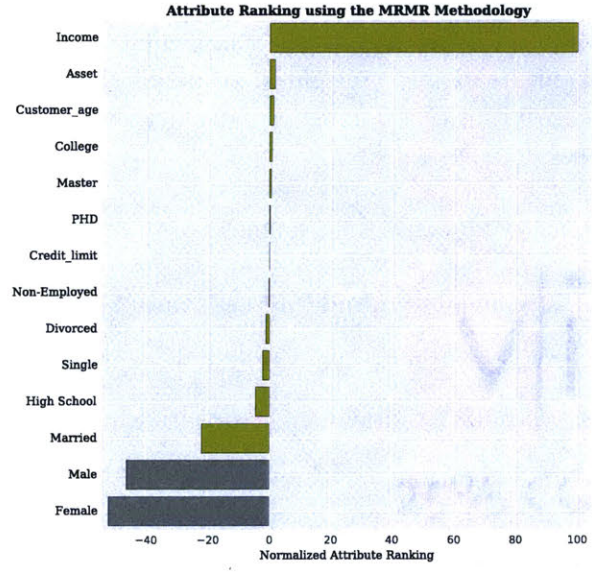
Going beyond the combined ranking, we can also drill down to examine the four rankings produced by each methodology. We provide this option so that analysts can examine the dependence of a variable of interest in multiple ways.



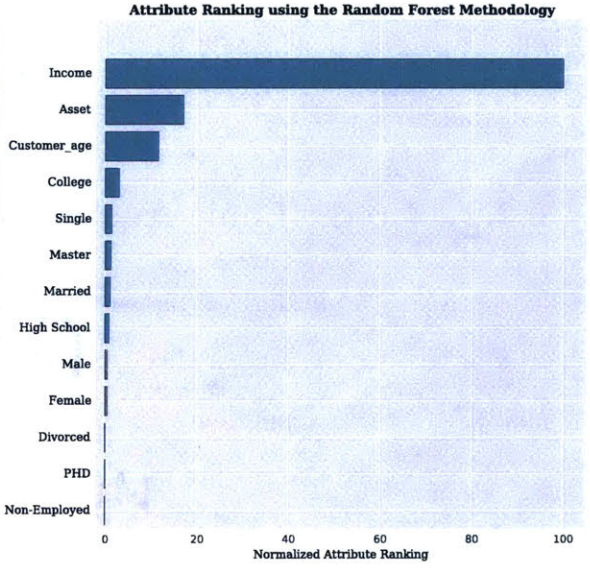
(a) IOFP Ranking Results



(b) LASSO Ranking Results



(c) mRMR Ranking Results



(d) Random Forest Ranking Results

Figure 6-2: Figure shows the input attribute ranking across all for ranking methods in FairML for the Bank's Credit Limit Audit

In examining each of the rankings produced across the four different methods shown in figure 6-2, we see a consistent story on the impact of the gender variables on the credit limit model. In each of the rankings produced, the predictive dependence of the bank's credit limit algorithm on gender is consistently low across the board indicating that the bank's model has little dependence on gender in making credit limit determinations. For the mRMR methodology, we even observe a negative impact on the bank's model of taking into account gender.

The results above strongly suggest that the bank's credit limit algorithm is not dependent on Gender. With simple bar plots like the ones shown above, we hope analysts and other non-experts can use FairML to make determinations about the relative weights a model places on its inputs. Further, analysts can use the graphs and results from FairML in presentations and talks as they seek to share their analyses with others. In cases where our audit show a particularly strong dependence on an input attribute such as gender or race, the analysts can more easily engage in conversations with the bank or entity being audited.

6.2.3 Probability of Default Algorithm Audit

Now we shift to an audit of the bank's probability-of-default model. Like in the previous audit, we are trying to investigate the dependence of the bank's probability of default algorithm on gender. As usual, we pass in the input variables and the output of the bank's algorithm to FairML for an audit. Similarly, we get a series of five plots that demonstrate the results of the audit.

Figure 6-3 shows the combined ranking obtained from the audit. Again, we see that the gender variables rank dead last among all of the variables tested. Further, we see that Income, Education level (college), and Customer age are the most significant variables for the bank's algorithm.

As we did for the credit limit audit, we can also drill down to further examine the individual rankings obtained from each ranking methodology. As we see again in figure 6-4, the predictive dependence of the bank's probability-of-default model on gender is consistently low across the board for all the different ranking methods.

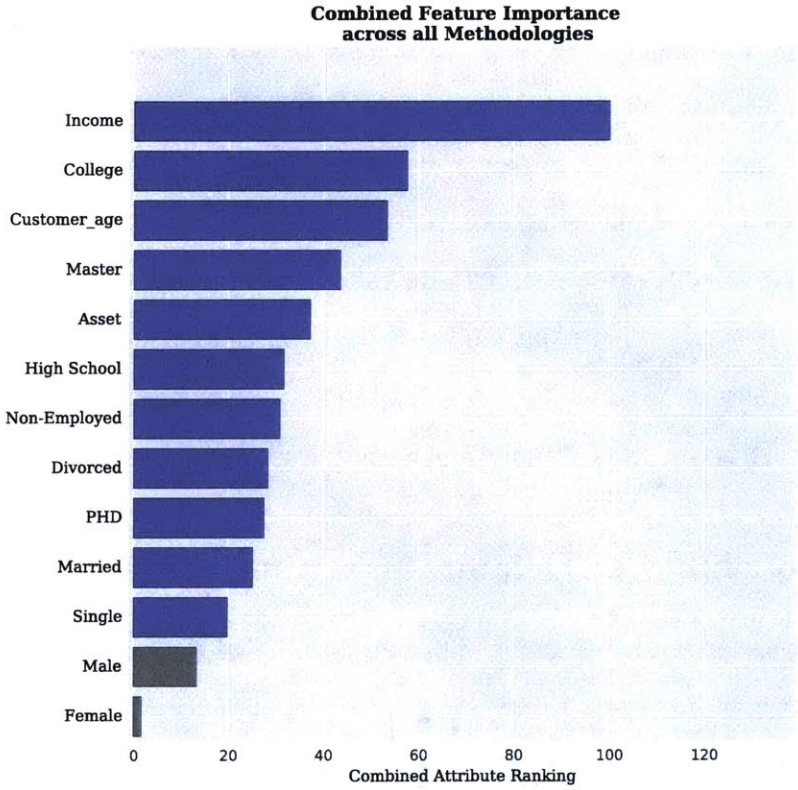
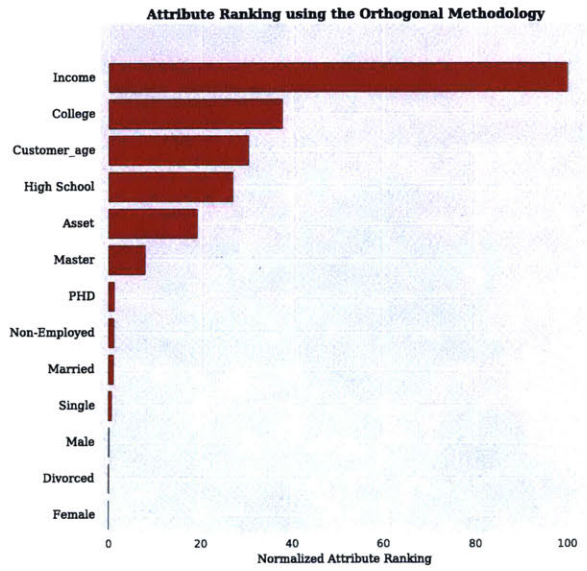


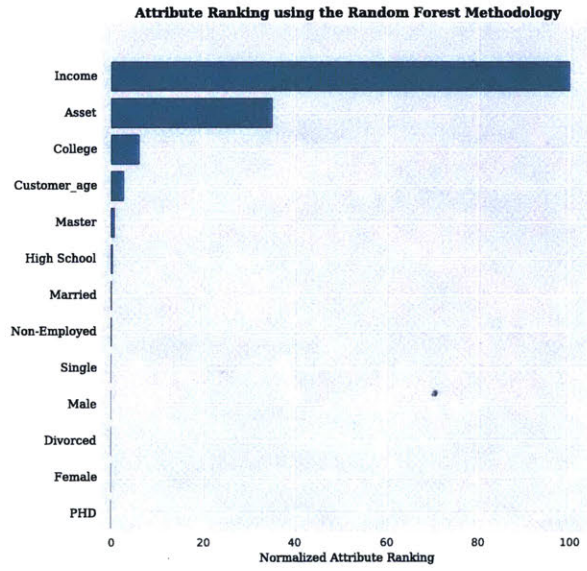
Figure 6-3: FairML output showing the combined ranking for each of the different inputs on the bank's probability of default algorithm. Here as well, we see that the gender variables (male and female), rank dead last compared to all of the other variables.

This indicates that the bank's model is not overly dependent on gender in making probability-of-default determinations. Again, the mRMR ranking procedure indicates that gender variables have a large negative effect on the bank's probability-of-default model.

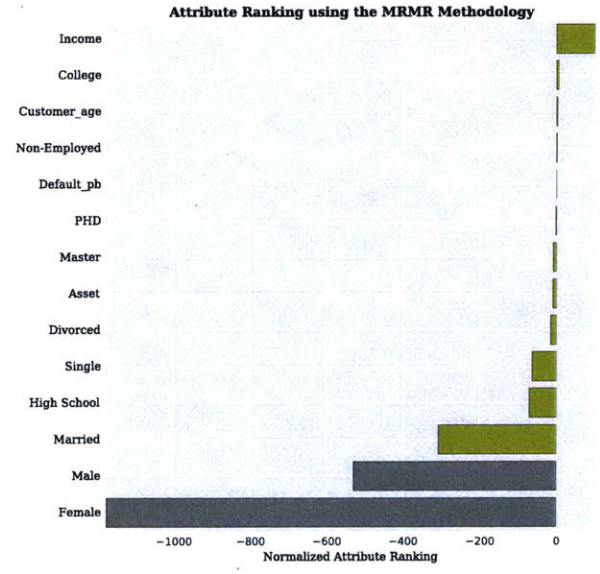
Through the series of audits presented in this chapter, we have shown how FairML can be used by analysts and other practitioners to gain a better understanding of predictive models.



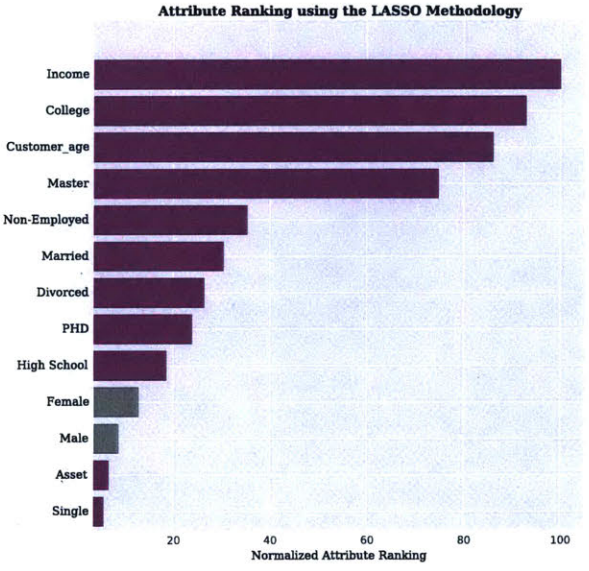
(a) IOFP Ranking Results



(b) Random Forest Ranking Results



(c) mRMR Ranking Results



(d) LASSO Ranking Results

Figure 6-4: Figure shows the input attribute ranking across all for ranking methods in FairML for the Bank's probability-of-default model audit

CHAPTER 7

DISCUSSION

7.1 Chapter Overview

In this chapter, we discuss several implications, touching on issues relating to the notion of algorithmic fairness, and causal inference. We begin with a discussion of the difference between traditional algorithms and predictive models, then describe different ways that bias can be introduced into a predictive model. Further, we describe, in detail, one popular definition of fairness, statistical parity, and highlight the advantages and disadvantages of this definition. We then present an overview of how we expect FairML to help bring about transparency to the predictive modeling process.

7.2 The Notion of Algorithmic Fairness

A commonly held belief regarding algorithms is that they are facially neutral. Indeed, a particular class of algorithms are neutral. However, predictive models derived from input data are subject to the intricacies and subtleties of such data. In this section, we present clarifying details on how bias can seep into the model-making process. Next,

we explore one possible definition of fairness, statistical parity, and present both its advantages and disadvantages. We end this section with a note on our recommended approach for addressing model fairness using FairML.

7.2.1 Traditional Deterministic Algorithms

An algorithm "is any well-defined computational procedure that takes some value, or set of values, as input and produces some value, or set of values, as output. An algorithm is thus a sequence of computational steps that transform the input into the output." [17] Informally, algorithms can be considered as "mathematical formulas" or procedures that transform input data into specific solutions [17]. For example, the arithmetic mean is an algorithm that takes a series of numbers as input and outputs the average. Similarly, the algorithm to find the circumference of a circle takes as input the radius of a circle, and produces the circumference. Algorithms can range from simple mathematical formulas to series of procedures combining several formulas depending on the complexity of the problem and the nature of the input data. Simple mathematical formulas are particularly suited to tackling well-defined and deterministic problems. The formulas for arithmetic mean, circumference of a circle, and the area of circle are all examples of simple algorithms. Algorithms that solve deterministic problems often have a fixed number of inputs, and a formula whose structure is generally constant regardless of input.

7.2.2 Predictive Models: Learning Algorithms in Non-Deterministic Settings

The traditional definition of an algorithm breaks down for problems where there is uncertainty regarding the nature of the input, and where the outcomes are not deterministic. In cases where the problem is unstructured, a simple mathematical formula is often not sufficient to provide a reasonable solution. For example, the problem of identifying groups of similar customers given past purchase history is an unstructured problem. The number of customers considered, overall number of items purchased,

as well as other factors can often vary from situation to situation, so a pre-defined mathematical formula similar to the arithmetic mean would not be appropriate in segmenting customers. Similarly, an algorithm to determine an individual's credit worthiness would also not fit into the traditional definition of algorithm. First, there are multiple ways of defining credit worthiness for an individual. Secondly, there are potentially unlimited number of input data points that one can consider when specifying such input data. For unstructured, and ill-posed questions like the ones referenced above, it is often difficult to specify a predefined set of instructions from which one can arrive at a solution.

For a certain class of unstructured problems where data is available, it is often possible to identify patterns from historical data to construct "mathematical formulas" for solving such problem. Algorithms learned from data typically leverage historical patterns to identify a solution. For example, a bank can learn to identify credit-worthy[1] customers by observing historical data from its past loan applications, and identifying the crucial characteristics associated with credit-worthiness.

To make the explanation in the previous paragraph more concrete, we present an hypothetical example. Given historical data, a bank can learn a precise mathematical formula for determining credit worthiness. Methodologies exist in machine learning and statistics with which one can learn such algorithms even in the presence of uncertainty. Let's say the bank collected historical data on income, asset, educational level, and marital status; and was able to learn a formula for credit worthiness from historical data specified as:

$$creditworthiness = 3 * income + asset + 0.5 * educationallevel - marritalstatus^2.$$

Given the above formula, the bank now possesses an algorithm with which it can rate new loan applications. The bank can set a threshold above which it will award individuals loans.

The data driven algorithm learning process described above represents a shift from

[1]Credit worthiness here is defined as the likelihood of an individual to repay a loan.
[2]This equation is clearly hypothetical, and presented for purely pedagogical purposes only.

the deterministic mathematical formulas described earlier. This process is usually referred to as *learning from data* [32]. The mathematical formula *learned* from the data is called a predictive model. This is because the formula is derived using historical data, and to be used for predicting a target variable of interest on unseen data. For traditional algorithms like the average of a set of numbers, the mathematical formula derived doesn't change with a change input, and the definition of the dependent variable is typically clear cut. However, this isn't the case for most predictive models. The exact formula for a predictive model depends on the historical data it is learned from, specific definition of the outcome of interest, and sometimes the methodology used in learning that formula.

7.2.3 Sources of Bias in Predictive Modeling

Because the algorithm learning process for predictive models is often tightly coupled to the available historical data, bias in such data will often lead to learning a biased predictive model. Predictive models are typically only as good as the data available to them to learn patterns from. Several researchers have attributed the resurgence in predictive modeling and increased accuracy of these predictive models mainly to increased access to high quality data. In the following points, we describe ways in which predictive models can encode bias as part of the mathematical formulas learned.

It is possible for bias to be incorporated into a predictive model through the definition of the outcome to be learned from data. For example, a bank could define credit worthiness on the basis of neighborhood income. In this case, an individual who would likely repay a loan, but lives in a low income neighborhood would be denied a loan by the bank. Such a definition of credit worthiness would be biased based on its definition. More generally, outcomes that are highly correlated with attributes like race, gender, sexual orientation, and religion can exhibit bias in a predictive model if not properly specified. Here, though an institution might not intend to discriminate against a particular group of individuals, the definition of the outcome variable could result in a predictive model, mathematical formula, that unintentionally discriminates.

Bias can also be incorporated into a predictive model if the data such model is learned from does not conform to the underlying population distribution. Most predictive models can only learn from examples in the data shown to them. Returning to our bank example, if in the data that a model is learned from, all individuals with a particular attribute are of low credit worthiness, then the model learned from the data can associate individuals of that attribute with poor credit worthiness. In her article titled, *The Hidden Biases of Big Data*, researcher Kate Crawford says,

> While massive datasets may feel very abstract, they are intricately linked to physical place and human culture. And places, like people, have their own individual character and grain. For example, Boston has a problem with potholes, patching approximately 20,000 every year. To help allocate its resources efficiently, the City of Boston released the excellent Street-Bump smartphone app, which draws on accelerometer and GPS data to help passively detect potholes, instantly reporting them to the city. While certainly a clever approach, StreetBump has a signal problem. People in lower income groups in the US are less likely to have smartphones, and this is particularly true of older residents, where smartphone penetration can be as low as 16%. For cities like Boston, this means that smartphone data sets are missing inputs from significant parts of the population – often those who have the fewest resources [18].

Crawford highlights a critical issue with learning predictive models from data; data misrepresentation. Often, if the underlying data from which a predictive model is learned is not representative of the population from which the data is drawn, and proper technical precautions are not taken to counteract this fact, then insights drawn from a predictive model learned from this data can often be flawed. There are multiple ways in which data misrepresentation can occur. First, the sample from which a model is learned can under represent a particular group in not providing a coverage of different collection of instances for such group. In other cases, a sample can over represent a particular group, there by creating a distortion that can be learned by

a predictive model. As noted in Crawford's article, data misrepresentation can be particularly challenging to address, and requires that individuals building predictive models pay particular attention to the inputs with which they are learning models that is to be used for making decisions.

The use of input attributes that are highly correlated with membership in a protected class[3] but facially different as part of input data for learning predictive models can also result in bias. In the article *Big Data's Disparate Impact*, Barocas et. al. describe this issue in the employment context.

> Cases of decision-making that do not artificially introduce discriminatory effects into the data mining process may nevertheless result in systematically less favorable determinations for members of protected classes. Situations of this sort are possible when the criteria that are genuinely relevant in making rational and well-informed decisions also happen to serve as reliable proxies for class membership [8].

A predictive model learned from data containing features that are highly correlated with membership in a protected class can also encode bias in the predictive model.

In this section, we have summarized a few different ways in which bias can be introduced into the predictive modeling process. The different ways described in this section are not exhaustive; however, they represent a few key ways that bias can arise.

7.2.4 One Definition of Fairness: Statistical Parity

In this section, we explore one definition of fairness, statistical parity, with regards to predictive models. To start, we provide a mathematical formulation of statistical parity as described in [38].

Given a population set, G, with a subset P consisting of samples from a protected class. For example, consider G to consist of all individuals, and P, a subset characterizing people of religion R. Returning to our previous example, we wish to audit

[3] As noted in earlier chapters and section, protected classes consists of groups of individuals that can be identified on the basis of race, gender, sexual orientation, religion and others as stated by law.

the loan granting process at a bank. Let D represent the population distribution characterizing the likelihood of an individual to seek a loan. Now, let's say we learn a data-driven predicted model $f : G \rightarrow \{-1, 1\}$ that assigns credit worthiness scores to individuals, where individuals with outcome -1 get denied applications for loans, and individuals with outcome 1 are accepted. We now define bias, b_f as follows:

$$b_f(G, P, D) = Pr(f(g) = 1 | g \in P^C) - Pr(f(g) = 1 | g \in P).$$

$Pr(f(g) = 1 | g \in P^C)$ corresponds to the probability of the bank giving an individual a loan given that they do not belong to the protected class, while $Pr(f(g) = 1 | g \in P)$ corresponds to the probability of the bank giving a loan to an individual given that such individual is a member of the protected class. The difference between these two probabilities is the bias of the model f.

A predictive model f is said to achieve statistical parity given (G, P, D), if $|b_f(G, P, D)| < \epsilon$, where ϵ corresponds to an infinitesimal number. Statistical parity is achieved when the difference in outcome probabilities for individuals in a protected class, and individuals not in a protected class is infinitesimally small.

One noticeable advantage of statistical parity is the elimination of any perceived disadvantage as a result of membership in a protected class. This property of statistical parity has made it one of the more popular definitions of fairness with regards to predictive models. However, statistical parity also exhibits several severe disadvantages. Dwork et. al highlight a few disadvantages of statistical parity in their work, two of which they term: Self-fulfilling prophecy and Subset Targeting [24].

In self-fulfilling prophecy, unqualified members of a particular class are chosen specifically to justify future discrimination against members of that class. In selecting members of the protected class that are clearly unqualified for a particular service, such selection can then be used to justify future discrimination by arguing that members of the protected class are generally unqualified.

For subset targeting, Dwork et. al offer the following example "consider an advertisement for a product X which is targeted to members of S that are likely to

be interested in X and to members of S^c that are very unlikely to be interested in X. Clicking on such an ad may be strongly correlated with membership in S (even if exposure to the ad obeys statistical parity)" [24]. As indicated above, statistical parity for a set does not imply statistical parity for subsets of that set.

As shown in the scenarios above, statistical parity has its disadvantages, and a few ways in which it can be exploited. Currently, there is no widely accepted definition of fairness within the research or legal community pertaining to the use of predictive models. The current state of affairs leaves the decision of ascertaining fairness to the courts, and analysts at national agencies typically on a case-by-case basis.

7.2.5 Going Forward: Using FairML to Audit Predictive Models

Our goal in this work is not to advocate for any particular definition of algorithmic fairness. It is our opinion that a definition of fairness that can be strictly applied across different cases will be almost impossible to nail down. The notion of fairness generally goes beyond any technical measure. Any notion of fairness quantifies and intrinsically contains the values of a society, and the ideals that such society strives toward. For example, western society has codified laws outlawing discrimination on the basis of race, gender, and other protected attributes. This means as a society, the western world has arrived at the conclusion that discrimination on the basis of these attributes is not an acceptable norm. As society continues to evolve and change, the general class of protected attributes might increase or decrease changing scenarios that could have been considered fair. Hence, it is our view that ascertaining fairness, especially regarding predictive models, requires human judgment as well as a complete understanding of the situation, industry, and task being considered.

With FairML, we provide analysts with a tool with which they can use to open up black-boxes in order to further understand how predictive models are assigning different probabilities. We expect analysts and other policy makers using our tool to make fairness calls on a case-by-case basis as they audit predictive models. The

variable ranking plots produced by FairML can be used by analysts to engage the individuals, institutions, and entities using the predictive model that is being audited.

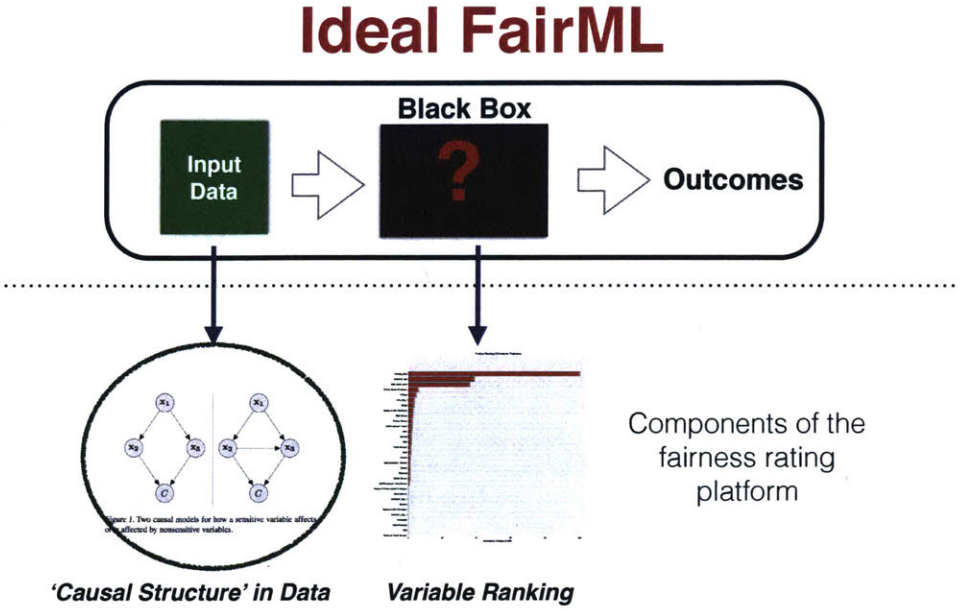
7.3 Causal Inference : The Ideal FairML

In the course of this chapter, we have described different ways in which a predictive model can biased. The primary way to introduce bias to a predictive model is through the data on which the model is learned. Consequently, a key aspect of an ideal auditing process ought to involve an audit of the underlying data on which a predictive model is to be learned as well. Over the past few years, and as noted in the chapter on related work, researchers have focused on quantifying bias in a data set. This line of work has resulted in the development of methodologies for transforming a given input data set into one from which a fair model can be learned.

In figure 7-1, we show what an ideal architecture of an auditing platform like FairML should consist of. As will be noted, the missing capability involves a module for quantifying bias encoded within a data set alone. With such a module, one can then distinguish bias encoded in the model from that encoded in the data. Inclusion of a module characterizing the structural dependence among the input variables would allow analysts to figure out if the source of bias in a model is attributable to the input data set that such model was learned from.

There are multiple ways of characterizing and quantifying structural bias given input data. One way would be to learn a graphical model, bayesian network, from the data in order to understand the conditional dependencies encoded in the data. Figure 7-1 shows a graphical model representing the conditional independence relationships among inputs. With a graphical model, one can now quantify the impact of each attribute given all the other attributes. Such representation will allow analysts to more clearly understand the source of bias in a model. Further, a graphical model can be used to adjust the rankings produced by FairML in order to indicate the bias contribution from the input data to a predictive model alone.

Aside learning a causal model from data, recent work has also demonstrated tech-



centering

Figure 7-1: Figure shows what an ideal architecture for a complete FairML would look like. This causal inference module would include algorithms for determining the dependence among input variables given just the input data.

niques through which one can quantify the amount of bias in a given input data set. In [29], Friedler et. al. propose a test for bias[4] given an input data set as well as a set of protected attributes against which one wishes to quantify discrimination. Given the test proposed by Friedler et. al., one can certainly quantify the level of bias in a data set. A slight modification of the algorithm presented by Friedler et. al. to create attribute level bias scores can be further used to adjust predictive models that are being learned from data. Given a biased data set, Friedler et. al also propose a repair methodology for learning a fair data set. Such methodology can be included in FairML in order to repair the underlying data set used in generating a predictive model.

With the version of FairML presented in this thesis, we hope to add to the conversation on fairness in data mining. Addition of structure learning and data diagnosis capabilities will help ensure that FairML audits provide thorough feedback of the entire model building process.

[4]In the paper, the test was particularly to measure disparate impact; however, the definition of bias can be changed depending on the context and application in question.

CHAPTER 8

LEGAL IMPLICATIONS: THE CREDIT ASSESSMENT INDUSTRY

8.1 Chapter Overview

In this chapter, we take a deeper look at the credit assessment industry in the United States. We use the credit assessment industry as a case-study to highlight how current laws and legislation are ill-equipped to handle a world where decisions are primarily driven by data and predictive modeling. The several recommendations presented in this chapter are specifically tailored to the credit assessment industry; however, we expect that the underlying principles can be broadly applied across industries where predictive modeling is becoming prevalent.

Predictive models are becoming the defacto tools for assessing credit-worthiness. Given this emergent phenomena, we examine two major pieces of legislation that govern the credit scoring industry: The Fair Credit Reporting Act (FCRA), and the Equal Credit Opportunity Act (ECOA). Specifically, we present the deficiencies and inadequacies of FCRA and ECOA, particularly in relation to predictive models used for credit scoring. We end with a set of recommendations for improving these laws.

8.2 Big Data & Alternative Credit Scoring Industry

This section seeks to put big-data credit scoring tools in context by describing the factors that have spurred their development, the problems they seek to address, and the potential risks they pose.

8.2.1 Challenges of Traditional Credit Scoring

A credit score is a "summary of a person's apparent creditworthiness that is used to make underwriting decisions," and is used to "predict the relative likelihood of a negative financial event, such as a default on a credit obligation" [60]. Over the course of the past three decades, automated credit-scoring systems such as the Fair and Issac's Company's FICO score have become a fundamental determinant in the broad majority of U.S. consumers' fiscal lives [15]. Without a sufficiently favorable score from a major credit bureau, a consumer likely cannot "buy a home, build a business, or send [her] children to college" [6]. While mainstream, automated underwriting tools are generally viewed as a better alternative than reliance on loan officer discretion because they are far less time-intensive and can avoid certain forms of bias, there is increasing concern that these tools unjustifiably disadvantage certain borrowers. An astounding number of U.S. consumers – 64 million according to an Experian report – are currently classed as "unscorable," meaning that they cannot access traditional forms of credit [26].

These consumers may be "immigrants or recent college grads [with] little to no credit history," or "people who haven't had an active credit account for at least six months" [26]. Because traditional credit models consider a relatively limited set of data points, they oftentimes struggle to account for many "thin-file" consumers' actual creditworthiness [15, 26]. The traditional FICO score, for instance, principally looks at a consumer's payment history, the amounts she owes, the length of her credit history, new credit, and types of credit used, while omitting factors such as employment history, salary, and other items that might suggest credit worthiness [60].

The perceived inability of traditional, automated credit scores to adequately cap-

ture "thin file" borrowers has prompted the emergence of alternative, big-data tools that promise to offer lenders a way to "squeeze additional performance" out of their underwriting processes [15, 26, 60]. Although a complete picture of the alternative credit scoring market is not possible, several emerging companies appear to use proprietary models to sift and sort through thousands of data points on each consumer.

The use of alternative credit scoring tools has led several individuals and regulatory agencies like the Federal Trade Commission, and the Consumer Financial Protection Bureau to call for these models to be banned. However, the use of these credit scoring models promises to provide credit to those who might not have credit under the previous system, which would increase access to capital for people. At the same point, the public is increasingly worried that these tools could perpetuate bias by scoring consumers on the basis of factors and classifications that are beyond their control. Careful and deliberate action ought to be taken to ensure that the tools being used abide by the law, however, improper regulation, also promises to lead to a possible collapse of the industry.

8.3 The Fair Credit Reporting Act

The Fair Credit Reporting Act (FCRA) was enacted in 1970 to serve the dual goals of ensuring fairness in consumer credit reporting, and safeguarding consumers' privacy through limitations on how consumer credit information can be disclosed or used [60]. The FCRA furthers these goals by "requir[ing] that consumer reporting agencies adopt reasonable procedures for meeting the needs of commerce for consumer credit, personnel, insurance, and other information in a manner which is fair and equitable to the consumer, with regard to the confidentiality, accuracy, relevancy, and proper utilization of such information" [2, 47]. The Act also seeks to ensure that consumers can access information about their scores, correct errors, and understand how their personal and credit data are being used by third parties who use it to make credit, employment, and insurance decisions [2, 47].

Whether a particular entity or reporting activity falls under the FCRA principally

depends on the types of information involved, the actual or expected uses of that information, and whether the information is reported by a "consumer reporting agency" (CRA). The FCRA governs "consumer reports," which are defined as containing "any information . . . bearing on a consumer's credit worthiness, credit standing, credit capacity, character, general reputation, personal characteristics, or mode of living" [2, 47]. The information need only satisfy one of these factors, with the practical implication that almost any information about a consumer might qualify.

8.4 The Equal Credit Opportunity Act and Regulation B

The purpose of the 1974 Equal Credit Opportunity Act(ECOA) is to "promote the availability of credit to all creditworthy applicants" in the fair lending context [63]. In order to achieve this objective, the ECOA makes it "unlawful for any creditor to discriminate against any applicant, with respect to any aspect of a credit transaction" about a prohibited basis such as "race, color, religion, national origin, sex or marital status, or age" [1, 44]. The basic directive against discrimination applies regardless of whether the transaction is a commercial or consumer one. An applicant is defined as any person who requests or who has received an extension of credit from a creditor, and includes any person who is or may become contractually liable regarding an extension of credit [1, 44]. A creditor is broadly defined as "a person who regularly extends, renews, or continues credit; any person who regularly arranges for the extension, renewal, or continuation of credit; or any assignee of an original creditor who participates in the decision to extend, renew, or continue credit" [1, 44]. A creditor may not inquire about "the race, color, religion, national origin, or sex of an applicant or any other person in connection with a credit transaction" [1, 44].

The United States Federal Trade Commission has overall enforcement authority under ECOA while the traditional banking regulatory agencies–such as FDIC–have enforcement power in the banking industry. Currently, the United States Consumer

Financial Protection Bureau has taken the lead role in issuing rulemakings such as Regulation B. These regulations prohibit discrimination on any of the grounds proscribed by the statute and establish rules on extending credit, taking applications, and evaluating applications [1, 44].

8.4.1 ECOA's Regulation B

Regulation B has two basic prohibitions: 1) a creditor shall not discriminate against an applicant on a prohibited basis regarding any aspect of a credit transaction, and 2) a creditor shall not make any oral or written statement, in advertising or otherwise, to applicant or prospective applicants that would discourage, on a prohibited basis, a reasonable person from making or pursuing an application. Generally, Regulation B notice requirements are triggered when adverse action–such as when an applicant is denied a loan–is taken on a credit application or an existing credit account [1, 44].

8.5 The Way Forward: Inadequacies of FCRA & ECOA, and Recommendations

This section summarizes the principal shortcomings of these laws, as well as what legislators can do to close the most problematic gaps and ensure that credit scoring is transparent and non-discriminatory.

The federal FCRA sets a minimum standard of transparency for credit reporting by providing consumers with the right to review the data in their reports. The FCRA does not, however, give consumers the right to understand how a particular credit assessment tool judges risk or weighs particular factors. While the FCRA's transparency requirements may be adequate for credit scoring tools that use relatively few factors, which are logically related to creditworthiness, the Act's transparency rules are ill-designed to govern credit scoring tools that use complex algorithms and thousands of data points.

Existing laws like FCRA establish accuracy requirements for the data used in

credit assessment tools, however consumers bear the burden of identifying and disputing inaccuracies. FCRA's accuracy requirements may only go so far, even in the conventional credit-scoring context. As credit assessment tools integrate more data points, many of which may be difficult for consumers to verify or dispute, the time has come to shift the burden of accuracy to the shoulders of the credit scorers themselves.

The federal ECOA currently prohibits lenders from basing lending decisions on certain factors such as race, ethnicity, and sex, but omits other sensitive characteristics such as gender. The Act also does not clearly govern the activities of developers of credit assessment tools unless these developers also engage in lending.

8.5.1 Recommendations

1. To tackle the lack of transparency, FCRA and ECOA can incorporate requirements mandating developers of big data credit scoring tools to reveal the most significant variables used in their assessment, as well as more general sources of data used.

2. To tackle the issue of accuracy of data used, FCRA and ECOA should require users of credit assessment tools to maintain rigorous standards of accuracy, conduct regular reviews of their data, and self-certify that they comply.

3. Lastly, government agencies can invest in research and development in order to develop platforms like FairML that can be used to perform an end-to-end audit of a predictive modeling process. Results of such audits can then be used to make judgement calls on the overall 'fairness' or discriminatory extent of a credit assessment tool.

CHAPTER 9

CONCLUSION

Predictive models for deriving actionable decisions have become pervasive and are starting to drive decisions in industries like employment services, insurance, housing, and banking. The use of these predictive models hopes to usher society into a better future; however, careful deployment is essential to prevent side effects such as discrimination. Predictive models are captive to the data from which they were derived, and it is possible for data to encode bias. Given the tremendous potential that leveraging these predictive models holds, it is important to develop tools and frameworks to help identify situations where a predictive model is susceptible to bias. Consequently, in this thesis, we have focused on developing a toolbox, FairML, for auditing predictive models for bias.

With FairML, an analyst can assess the strength of the dependence between a predictive model and inputs to that model. The strength of the relationship between a model and a particular input points to the significance of that input in assigning outcomes. The rankings produced by FairML can be used to understand the influence of each input attribute on the outcome of a model. If the influence of a particular attribute such as race, gender, sexual orientation, or religion is excessively high, one can conclude that such a model will perpetuate bias. Overall, FairML enables

analysts to engage with modelers and entities developing predictive models to better understand potential pitfalls in the model building process.

More concretely, FairML consists of four variable ranking methodologies that quantify the relative significance of an input to a black-box predictive model. FairML produces bar plots that are easily understandable. Further, as part of FairML, we present a novel input ranking algorithm, the Iterative Orthogonal Feature Projection Algorithm (IOFP), for quantifying the dependence of a black-box predictive model on its inputs. Through simulation, we are able to show that our IOFP can be adapted to iteratively query black-box algorithms if direct access is available, which leads to a 15% improvement in performance.

We have resisted the urge to make an explicit and precise definition, technically or qualitatively, of what it means for a predictive model to be biased or fair. We believe that such a definition, one that is generally applicable independent of domain, is currently unnecessary, and perhaps impossible to attain. Instead, we believe that the process of determining the fairness of a particular predictive model is iterative, and should involve humans at every stage. Hence, we present FairML to help analysts as they wrestle with ascertaining bias in a modeling process.

As part of this work, we have also examined current laws and legislation, such as FCRA and ECOA, that regulate the credit industry where predictive models are becoming increasingly deployed. We have specified a few ways in which these laws are currently deficient and ill-equipped to address the use of predictive models in the credit industry. Hence, we offer a few recommendations to remedy several of these deficiencies. While these recommendations are mostly specific to FCRA and ECOA, the principles underlying them can be broadly applied to other industries as well.

Ultimately, eliminating bias from the predictive modeling process requires coordinated effort between practitioners and policymakers. In this thesis, we have tackled one missing piece: the need for easy-to-use tools that can diagnose bias in predictive modeling. It is our hope that the tools, underlying methodologies, and ideas presented in this thesis will help practitioners, analysts, policymakers, and others seeking to develop, deploy or audit predictive models.

BIBLIOGRAPHY

[1] Equal Credit Opportunity Act. Equal credit opportunity act. *Jeffrey I. Langer and.*

[2] Fair Credit Reporting Act. Usc § 1681 et seq. In *Senate and House of Representatives of the United States of America in Congress assembled (available at http://www. ftc. gov/os/statutes/031224fcra. pdf (last accessed 31 August 2006))*, 15.

[3] Mohsen Ali and Jeffrey Ho. Deconstructing binary classifiers in computer vision. In *Computer Vision–ACCV 2014*, pages 468–482. Springer, 2015.

[4] Alexandr Andoni, Rina Panigrahy, Gregory Valiant, and Li Zhang. Learning polynomials with neural networks. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1908–1916, 2014.

[5] Joshua D Angrist and Jörn-Steffen Pischke. *Mostly harmless econometrics: An empiricist's companion.* Princeton university press, 2008.

[6] Ashoka. Banking the unbanked: A how-to. 2013.

[7] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Advances in Neural Information Processing Systems*, pages 2654–2662, 2014.

[8] Solon Barocas and Andrew D Selbst. Big data's disparate impact. *Available at SSRN 2477899*, 2014.

[9] Joshua Blumenstock, Gabriel Cadamuro, and Robert On. Predicting poverty and wealth from mobile phone metadata. *Science*, 350(6264):1073–1076, 2015.

[10] Andrey Bogomolov, Bruno Lepri, Jacopo Staiano, Nuria Oliver, Fabio Pianesi, and Alex Pentland. Once upon a crime: Towards crime prediction from demographics and mobile data. In *Proceedings of the 16th International Conference on Multimodal Interaction*, pages 427–434. ACM, 2014.

[11] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[12] Cristian Bucilua, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541. ACM, 2006.

[13] Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21(2):277–292, 2010.

[14] Enrique Castillo, José María Menéndez, and Santos Sánchez-Cambronero. Predicting traffic flow using bayesian networks. *Transportation Research Part B: Methodological*, 42(5):482–509, 2008.

[15] Natural Consumer Law Center. Big data: A big disappointment for scoring consumer credit risk. pages 27–28, 2014.

[16] Danielle Keats Citron and Frank A Pasquale. The scored society: due process for automated predictions. *Washington Law Review*, 89, 2014.

[17] Thomas H.. Cormen, Charles Eric Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms*, volume 6. MIT press Cambridge, 2001.

[18] Kate Crawford. The hidden biases in big data. *HBR Blog Network*, 1, 2013.

[19] Ying Cui and Jennifer G Dy. Orthogonal principal feature selection. 2008.

[20] Nicolas De Jay, Simon Papillon-Cavanagh, Catharina Olsen, Nehme El-Hachem, Gianluca Bontempi, and Benjamin Haibe-Kains. mrmre: an r package for parallelized mrmr ensemble feature selection. *Bioinformatics*, 29(18):2365–2368, 2013.

[21] Jeff Dean. *Large Scale Deep Learning for Intelligent Computer Systems*, 2015 (accessed January 1, 2016).

[22] Chris Ding and Hanchuan Peng. Minimum redundancy feature selection from microarray gene expression data. *Journal of bioinformatics and computational biology*, 3(02):185–205, 2005.

[23] Laney Douglas. 3d data management: Controlling data volume, velocity and variety. *Gartner. Retrieved*, 6, 2001.

[24] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 214–226. ACM, 2012.

[25] Ahmed El Deeb. The unreasonable effectiveness of random forests, 2015.

[26] Blake Ellis. Millions without credit scores not so risky afer all. 2013.

[27] Donald E Farrar and Robert R Glauber. Multicollinearity in regression analysis: the problem revisited. *The Review of Economic and Statistics*, pages 92–107, 1967.

[28] ICML FATML. *Fairness, Accountability, and Transparency in Machine Learning*, 2009 (accessed January 1, 2016).

[29] Sorelle Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. *arXiv preprint arXiv:1412.3756*, 2014.

[30] Matthew S Gerber. Predicting crime using twitter and kernel density estimation. *Decision Support Systems*, 61:115–125, 2014.

[31] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.

[32] Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data. *Intelligent Systems, IEEE*, 24(2):8–12, 2009.

[33] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[34] IBM. *Bringing Big Data to the Enterprise*, 2009 (accessed January 1, 2016).

[35] Faisal Kamiran and Toon Calders. Classifying without discriminating. In *Computer, Control and Communication, 2009. IC4 2009. 2nd International Conference on*, pages 1–6. IEEE, 2009.

[36] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. Fairness-aware learning through regularization approach. In *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*, pages 643–650. IEEE, 2011.

[37] Bruce R Kowalski and CF Bender. An orthogonal feature selection method. *Pattern Recognition*, 8(1):1–4, 1976.

[38] Jeremy Kun. One definition of algorithmic fairness: statistical parity, 2015.

[39] David Lazer, Alex Sandy Pentland, Lada Adamic, Sinan Aral, Albert Laszlo Barabasi, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, et al. Life in the network: the coming age of computational social science. *Science (New York, NY)*, 323(5915):721, 2009.

[40] Yann LeCun and M Ranzato. Deep learning tutorial. In *Tutorials in International Conference on Machine Learning (ICMLâĂŹ13)*. Citeseer, 2013.

[41] Steve Lohr. For big-data scientists, janitor work is key hurdle to insights. *The New York Times*, 17, 2014.

[42] Gilles Louppe, Louis Wehenkel, Antonio Sutera, and Pierre Geurts. Understanding variable importances in forests of randomized trees. In *Advances in Neural Information Processing Systems*, pages 431–439, 2013.

[43] Daniel Lowd and Christopher Meek. Adversarial learning. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 641–647. ACM, 2005.

[44] Earl M Maltz and Fred H Miller. Equal credit opportunity act and regulation b, the. *Okla. L. Rev.*, 31:1, 1978.

[45] Edward R Mansfield and Billy P Helms. Detecting multicollinearity. *The American Statistician*, 36(3a):158–160, 1982.

[46] James Manyika, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, and Angela H Byers. Big data: The next frontier for innovation, competition, and productivity. 2011.

[47] Robert M McNamara Jr. Fair credit reporting act: A legislative overview, the. *J. Pub. L.*, 22:67, 1973.

[48] Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, pages 1436–1462, 2006.

[49] Nicolai Meinshausen and Peter Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473, 2010.

[50] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[51] Sreerama K Murthy. Automatic construction of decision trees from data: A multi-disciplinary survey. *Data mining and knowledge discovery*, 2(4):345–389, 1998.

[52] City of Chicago. *City of Chicago: Data Science*, 2011 (accessed January 1, 2016).

[53] George Papamakarios. Distilling model knowledge. *arXiv preprint arXiv:1510.02437*, 2015.

[54] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, 12:2825–2830, 2011.

[55] Dino Pedreshi, Salvatore Ruggieri, and Franco Turini. Discrimination-aware data mining. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 560–568. ACM, 2008.

[56] HC Peng, Chris Ding, and FH Long. Minimum redundancy-maximum relevance feature selection, 2005.

[57] Alex Pentland. *Social Physics: How Good Ideas Spread-The Lessons from a New Science*. Penguin, 2014.

[58] John Podesta. *Big Data: Seizing Opportunities Preserving Values.* 2014.

[59] Foster Provost and Tom Fawcett. Data science and its relationship to big data and data-driven decision making. *Big Data*, 1(1):51–59, 2013.

[60] Robinson and Yu. Knowing the score: New data, underwriting, and marketing in the consumer credit marketplace. pages 7–8, 2014.

[61] Andrea Romei and Salvatore Ruggieri. A multidisciplinary survey on discrimination analysis. *The Knowledge Engineering Review*, 29(05):582–638, 2014.

[62] Mary Ann Schroeder. Diagnosing and dealing with multicollinearity. *Western journal of nursing research*, 12(2):175–84, 1990.

[63] Milton R Schroeder. *The Law and Regulation of Financial Institutions*, volume 3. Warren, Gorham & Lamont, 1995.

[64] Scikit-Learn. Ensemble methods: Feature importance evaluation, 2015.

[65] Lindsay I Smith. A tutorial on principal components analysis. *Cornell University, USA*, 51:52, 2002.

[66] Forrest R Stevens, Andrea E Gaughan, Catherine Linard, and Andrew J Tatem. Disaggregating census data for population mapping using random forests with remotely-sensed and ancillary data. *PloS one*, 10(2):e0107042, 2015.

[67] R Core Team. R language definition, 2000.

[68] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[69] Guido Van Rossum et al. Python programming language. In *USENIX Annual Technical Conference*, volume 41, 2007.

[70] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 325–333, 2013.

[71] Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429, 2006.