WARWICK
THE UNIVERSITY OF WARWICK

IM903 - COMPLEXITY IN THE SOCIAL SCIENCES

Algorithmic Fairness

Annika Stechemesser
1664383

31.05.2018

1 Introduction

Since the early 2000s, machine learning methods made their way from theory into our daily lives. As the collection, storage and analysis of large datasets became technologically possible, our society changed significantly. Data driven service optimisation, evidence based policy making - how thorough the data drive in our society is can maybe be understood by having a look at the news. Headlines like "Refugees could be resettled using computer algorithm that doubles chances of finding them employment"[10], "Spotify introduces Time Capsule feature that works out your music taste as a teenager" [15] and "When algorithms control the world"[4] give an impression of the diversity of problems algorithms are applied to. A relatively new technology that has such a profound impact on our daily lives naturally also comes with downsides and ethical problems. Numerous machine learning algorithms operate on human domains. Studies [2][3] showed that the objectivity often associated with machine decisions is questionable. The potential for discrimination against certain groups or individuals, the "algorithmic bias", entails a number of problems. How to measure fairness in algorithmic decisions? Who should be held accountable for unfair decisions? How can policy makers from different backgrounds be equipped with an analytical framework that enables them to approach these issues?

In this essay, we first define algorithmic decision making and explore how algorithms can be unfair. To fill the theory with life, we consider the example of the COMPAS software[14] which is used in recidivism risk prediction in the US. We introduce different mathematical definitions of fairness, apply them to the COMPAS dataset[16] and compare the results. Finally, we discuss the usage of algorithms on human domains and point out ways in the future.

2 Algorithmic decision making - How can algorithms be unfair?

To understand algorithmic fairness we first have to understand (algorithmic) decision making. One definition of human decision making is that it is the outcome of a careful evaluation of alternative options in terms of the likelihood and the value of outcomes associated with these options[18]. In algorithmic decision making, the evaluation and finding of the output value are done by an algorithm. The simplest approach to fairness, and maybe the most intuitive one, is probably the definition given in the Cambridge Dictionary: "Fairness - the fact of treating everyone the same way"[7]. It is obvious that fairness in human decision making is almost impossible. We are aware of our own objectiveness which is one of the reasons why we want to rely on algorithms in the first place. Machines apply the same decision procedure to each input entity, so how can they be unfair?

There are multiple reasons that can lead to disparities in the output of algorithms. Firstly, machine learning algorithms are trained on large datasets and their "behaviour" is dependent on the information extracted from this data. If the training data is biased, the output of the algorithm will be biased too. One example for biased data causing biased outputs are so called word embeddings, a procedure used in natural language processing tasks like sentiment analysis. Bolukbasi et al. showed that word embeddings trained on Google News articles exhibit female/male gender stereotypes due to gender stereotypes in the data[3]. Even if data is completely unbiased and critical information like race or gender are not part of the input, the output may be biased nevertheless. This is due to the great ability of machine learning algorithms to pick up encoded features in data. If the amount of attributes we feed the algorithm is rich and varying enough, the algorithm is likely to conclude race and gender from the features presented[2]. This is precisely what we should expect because picking up non-obvious patterns in data is one of the great strengths of these algorithms. To what extend this indirect use of sensitive information is harmful is often unclear. Furthermore, algorithms may be considered fair or unfair depending on the measure of fairness (see section 4). Lastly, it is possible that the algorithm was purposefully designed to discriminate against certain groups of people or individuals.

3 Example: The COMPAS algorithm

The COMPAS software (Correctional Offender Management Profiling for Alternative Sanctions) is a commercial web-based tool developed by Northpointe Incorporated for assessing recidivism risk in criminal offenders[14] being frequently used by judges in the US. The algorithm assigns criminal offenders a risk score (decile score) between 1 and 10, where 1 corresponds to a very low risk of re-offending and 10 to a very high risk. Defendants are assessed according to a long questionnaire [1], including questions about sex, criminal history, family criminality, substance abuse, education and various others. The algorithm uses a subset of these answers as input. Race is not explicitly taken as an input but nevertheless accusations arose that the COMPAS algorithm is discriminating against African-American defendants. If relying on a software to make decisions about delicate matters like personal freedom, it is of highest importance to answer the question: Is the COMPAS algorithm fair?

3.1 In favor of the fairness of COMPAS

Northpointe provides several references that vouch for the fairness of the algorithm. In 2010, the Center for Criminology and Public Policy Research at Florida State University published a study[9] evaluating over 5000 cases in which COMPAS was used. The results (figure 1) show that the percentage of recidivated cases is comparable in all scoring categories (low/medium/high) for each ethnic background. After the COMPAS system was accused by ProPublica (see section 3.2) Flores et al. [19] published a rectification again confirming the its fairness.

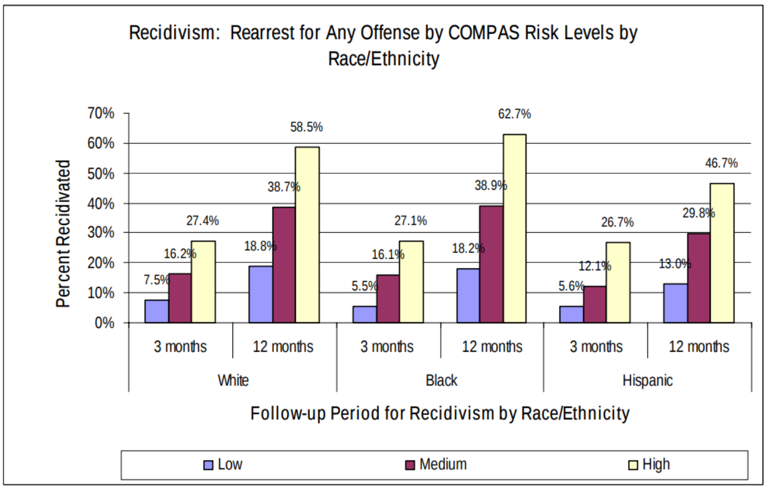


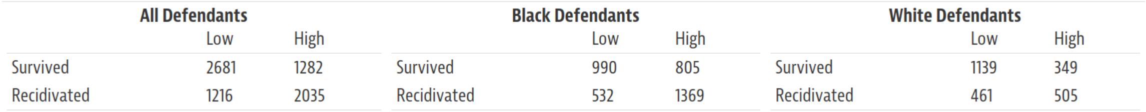
Figure 1: Rearrests for any offense by COMPAS risk levels by Race/Ethnicity; Source: [REFERENZ]

3.2 Against the fairness of COMPAS

ProPublica is an American nonprofit news organisation dedicated to investigative journalism. In 2016 they published a sensational article about COMPAS [13], featuring stories by defendants that were wronged by predictions alongside a profound anaylsis of the COMPAS dataset[16] proving that COMPAS is discriminating against African-American people. The code and methodology of this analysis are public and peer reviewed [12]. Their main findings show that African-American defendants are often assigned a higher risk of re-offense than they actually turned out to be whereas Caucasian defendants are more likely to be assigned a score too low to represent their risk of recidivism (Figure 2). The output of the algorithm is therefore clearly unfair.

3.3 Conclusion

The analysis of the COMPAS data presented in sections 3.1 and 3.2 is contradictory. Which side is right? The astonishing answer found by several scientists, amongst them the Max Planck Institute



All Defendants	Low	High
Survived	2681	1282
Recidivated	1216	2035

Black Defendants	Low	High
Survived	990	805
Recidivated	532	1369

White Defendants	Low	High
Survived	1139	349
Recidivated	461	505

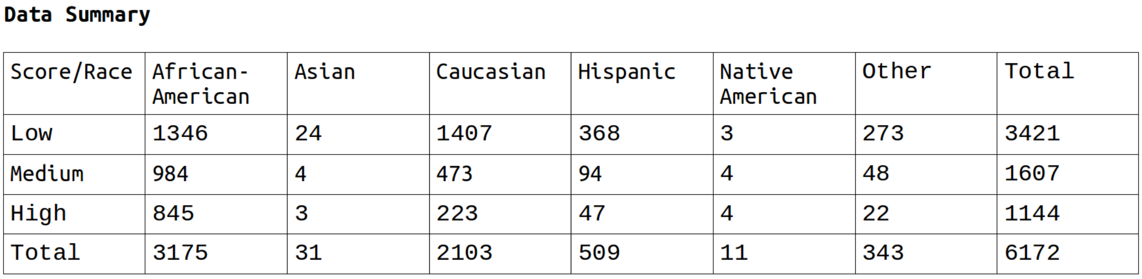
Figure 2: Number of recidivated cases across the scoring categories for all races; Source: [12]

for Software Systems [20], is that both sides are right. How is this possible? The answer lies in the mathematical definition of fairness.

4 Fairness in a mathematical context

Fairness is not one rigorous concept but can be defined and understood in different ways. Here, we briefly introduce four main measures of fairness that are especially interesting with regard to the COMPAS data, following the outline given by Corbett-Davies et al. [6] and apply these definitions to the dataset. All the code for the analysis was written in R and can be found in the appendix.

The relevant features of the COMPAS dataset we use are race, decile score assigned by COMPAS, score category (low/medium/high), number of prior convictions and whether or not the defendant reoffended within two years after release. A summary of the distribution of defendants across the races and scoring categories is shown in Figure 3. Due to the very low number of records we omit "Asian", "Native-American" and "Other" from the analysis. Even though we consider "Hispanic" as well, we focus on comparing "African-American" and "Caucasian" as they are more similar in numbers.



Data Summary

Score/Race	African-American	Asian	Caucasian	Hispanic	Native American	Other	Total
Low	1346	24	1407	368	3	273	3421
Medium	984	4	473	94	4	48	1607
High	845	3	223	47	4	22	1144
Total	3175	31	2103	509	11	343	6172

Figure 3: Summary of the COMPAS data (Score/Race)

4.1 Measures of fairness - Statistical parity

Consider a population that has different groups, for example. a "protected" and a "majority" group. Every individual in the population is assigned a class by the algorithm (i.e. {-1,1}, credit-worthy/not credit-worthy, not dangerous/dangerous). Statistical parity means that all classes get a similar outcome. In context of the COMPAS situation this means that the proportion of defendants in each race group that are scored high/medium/low is equal. Figure 4 shows that the COMPAS outcome clearly doesn't fulfill this condition. Comparing the rates of each group (blue/orange/green) to the general rate (red) we see significant differences. African-American defendants are more than twice as likely to be assigned a high decile score than Caucasian defendants. They are also more than 20% less likely to receive a low score in comparison to the other groups. Considering this measure of fairness, the COMPAS algorithm is clearly discriminating against African-American people.

4.2 Conditional statistical parity

Considering the same setting as in 4.1, conditional statistical parity is achieved if across all groups ("protected"/"majority", "African-American"/"Caucasian") individuals with a similar relevant known **attribute** are classified in the categories at equal rate. This attribute depends on the context. In the COMPAS setting we choose a similar prior number of convictions as our legitimate

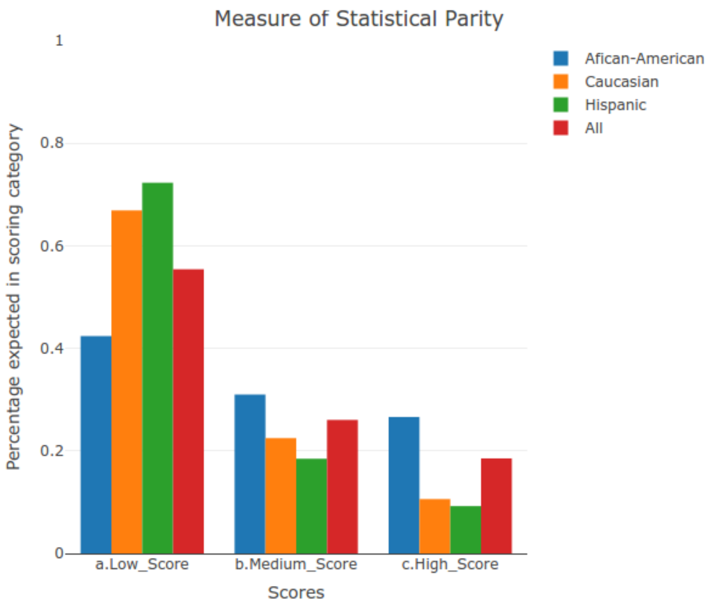


Figure 4: Fairness measure: Statistical Parity; Result: Unfair

attribute. We plot a histogram of the prior convictions (Figure 5) and group the prior convictions in three groups according to it (few(0-2)/medium(3-10)/high(11+)). Conditional statistical parity with respect to this attribute is achieved if all defendants in the same category according to their prior convictions are equally scored high, medium and low across all race groups.
Figure 5 shows the percentage of defendants in each scoring category for few, medium and high prior arrests. The rates are very different in each of these categories. Most importantly, African-American defendants with few prior arrests are a more than twice as likely to receive a high decile score than Caucasian defendants. Co-occurring, defendants with a high number of prior arrests are a more than twice as likely to receive a low score if they are Caucasian instead of African-American. According to this measure, the algorithm is clearly unfair.

4.3 Predictive equality

Predictive equality is achieved if the rate of misclassification is equal across all groups. A false classification happened if the algorithm assigned a person to a group that they retrospectively didn't belong to. Some proportion of misclassification is normal but if it is disproportional across the groups one can argue that the algorithm is discriminative. In the COMPAS context, a defendant would be misclassified if she/he either received a low score but reoffended (false negative) or if he/she received a high score but didn't reoffend (false positive).

Figure 6 considers the distribution across the scoring categories of all defendants that **did reoffended** within two years and all defendants that **didn't reoffend**. We see that the rate of false positive among African-Americans is 20% higher than among Caucasians. Co-occurring, the rate for false negatives is significantly higher for Caucasian people than for African-Americans. These results match the results of ProPublica (figure 2). According to this measure, the algorithm is clearly unfair.

4.4 Calibration

All the definitions above define the fairness of the decision made by the algorithm. In some contexts it can make sense to consider the fairness of a **risk score** (for example the COMPAS output means the risk of recidivism). To measure the fairness of risk scores the most common criterion is calibration. Calibration is achieved if among all individuals with a given label (i.e. a high risk of recidivism) the proportion of individuals who match the label (i.e. reoffend) is equal across all groups.

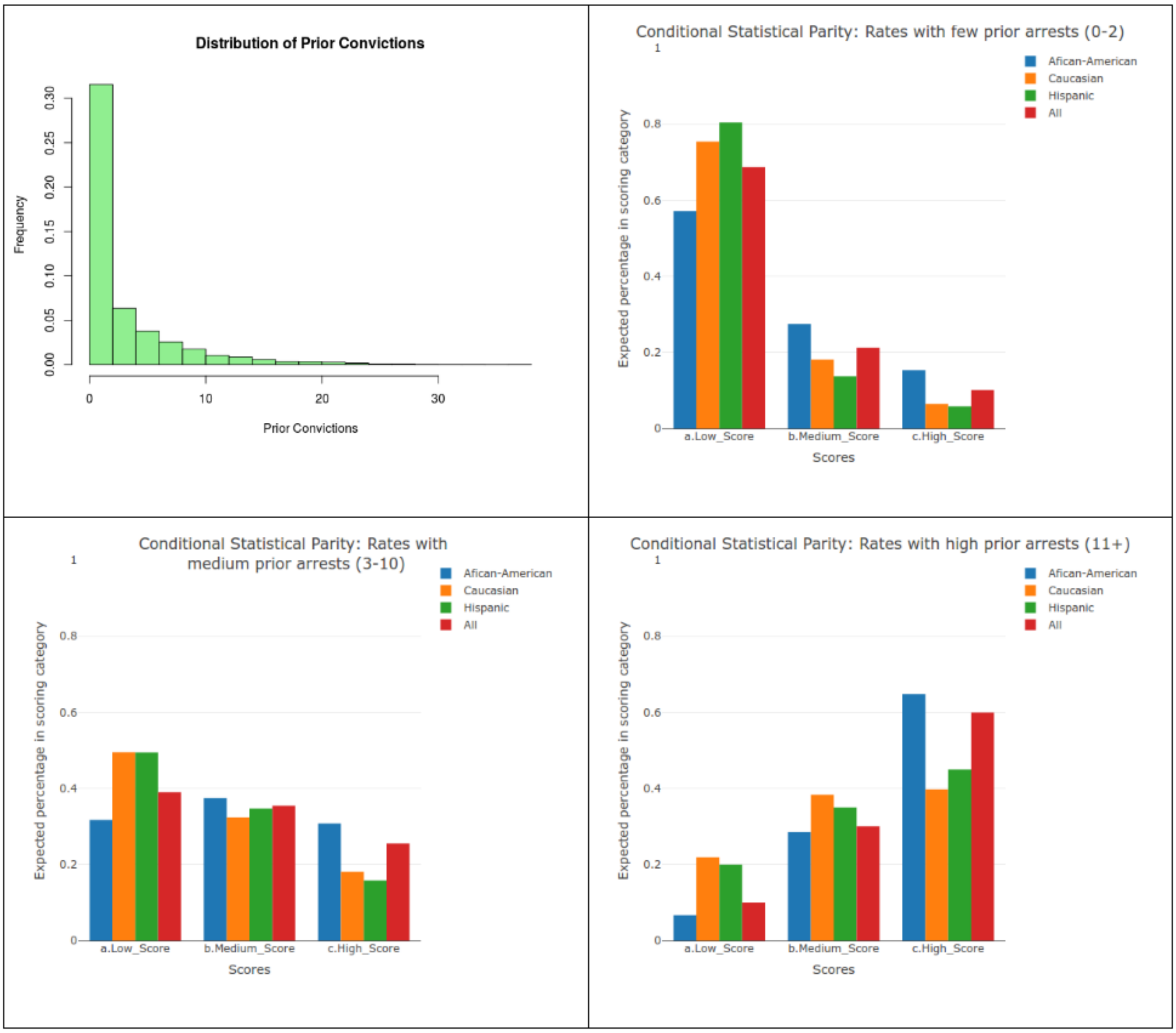


Figure 5: Fairness measure: Conditional Statistical Parity; Result: Unfair

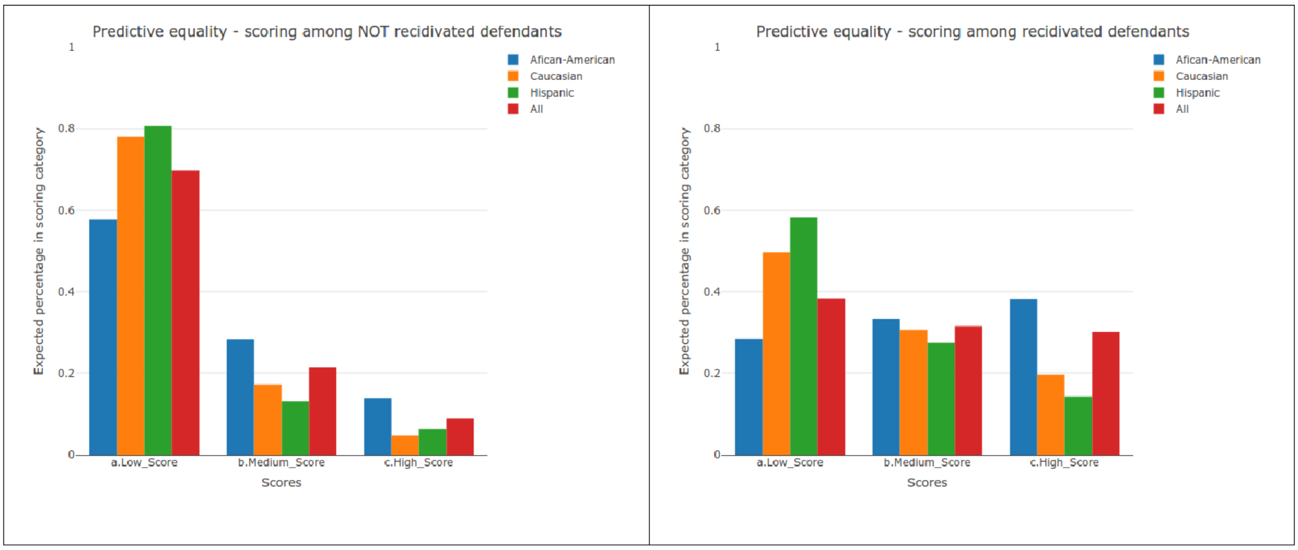


Figure 6: Fairness measure: Predictive Equality ; Result: Unfair

The percentage of reoffenses in each group is shown in figure 7. The rate is of similar magnitude for all classes and races. The percentage of reoffenses of Hispanic people that were scores high is lower, but as there are only 47 records in this category this doesn't necessarily indicate unfairness. These results match the results of Northepoint (Figure 1). According to the calibration measure, the algorithm is fair.

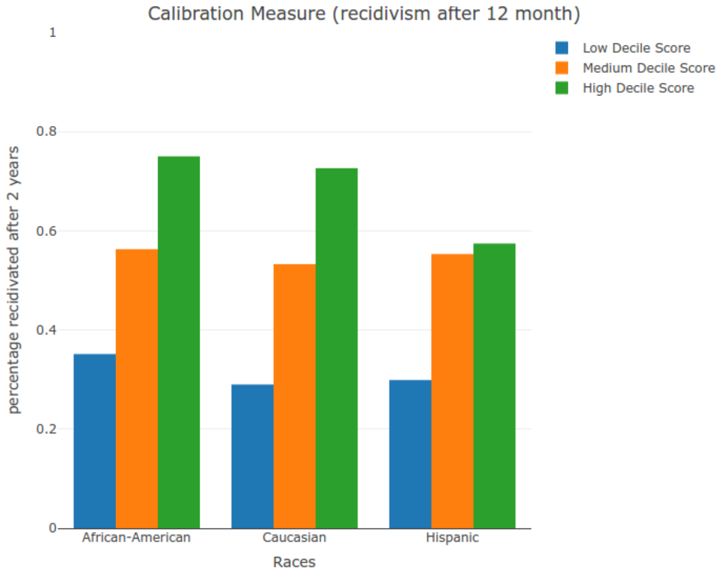


Figure 7: Fairness measure: Calibration ; Result: Fair

4.5 Discussion: Fairness and Trade-offs

While all the above defined notions of fairness are sensible, they conflict each other. Chouldechouva showed [5] that there is an unavoidable trade-off between calibration and the alternative fairness definitions. Which measure should be considered? This decision needs to be made by policymakers. The fairness the algorithm can achieve is dependent on the **decision rule** given to it. A decision rule is the principle according to which the algorithm assigns a category to each input. It is dependent on the **goal** the usage of the algorithm is meant to fulfill. Considering the COMPAS algortihm, the goal of the COMPAS scoring is to prevent crime committed by released defendants while minimising the cost for imprisonment. Fulfilling any of the fairness definitions outlined in section 4 presents a restriction to the optimal solution that will limit the effect of the decision rule. In the case of the COMPAS algorithm this could for example mean a reduced public safety. This trade-off is a decision that needs to be balanced carefully.

5 Ways in the future

The example of the COMPAS algorithm illustrates that algorithmic fairness is a multi-layered problem. We have seen that it is impossible to fulfill the four definitions of fairness outlined in section 4 simultaneously, let alone all possible mathematical definitions of fairness. Should algorithms on human domains then be used at all?

There isn't an easy answer to this question. While neither the technology nor the system it is used in are perfect, we as humans are not able to maintain our society as we now know it without algorithmic help. Nevertheless algorithmic outputs should never be taken as ground truth. Especially when concerning crucial decisions they can provide a guideline at most and should always be backup-ed by trained professionals.

Setting up a framework for algorithms to work hand in hand with humans is one of the greatest challenges in interdisciplinary work. Policymakers, scientists and professionals using algorithms need to agree on shared principles and find a common language to understand each others needs.

The Max Planck Institute for Software Systems summarised the principles algorithmic decision making should satisfy in the "FATE of algorithms" [20], standing for Fairness, Accountability, Transparency and Explainability. If communicating these principles between the disciplines and to the generic public is successful algorithmic decision making can improve our lives in manifold ways. The push towards new ethical standards in data science is clearly visible at the moment. Conferences like the Global Summit of "Doing AI for good" [17], hosted by the UN in May 2018, unify researches from different disciplines with a common goal. The British government discusses the possibility of a "Hippocratic Oath" for data science, a morally binding contract making sure AI is used responsibly [11]. The EU released a new "General Data Protection Regulation" [8] in 2018, designed to empower the EU citizens data privacy.

While it is still a long way, it is possible that with joint efforts of scientists of all disciplines "doing AI for good" could become our reality, using algorithmic support where it is useful in a transparent, understandable way and human empathy and understanding for individualism where it is needed.

References

[1] J. Angwin (ProPublica). Sample-COMPAS-Risk-Assessment-COMPAS-"CORE". retrieved from https://www.propublica.org/documents/item/2702103-Sample-Risk-Assessment-COMPAS-CORE, 2018-05-21 at 13:20.

[2] S. Barocas and A. D. Selbst. Big Data's Disparate Impact. SSRN Scholarly Paper ID 2477899, Social Science Research Network, Rochester, NY, 2016.

[3] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai. Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. page 9.

[4] J. W. Cellan-Jones, Rory. When algorithms control the world. *BBC News*, Aug. 2011. retrieved from http://www.bbc.com/news/technology-14306146, 2018-05-19 at 18:05.

[5] A. Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *arXiv:1610.07524 [cs, stat]*, Oct. 2016. arXiv: 1610.07524.

[6] S. Corbett-Davies, E. Pierson, A. Feller, S. Goel, and A. Huq. Algorithmic decision making and the cost of fairness. *arXiv:1701.08230 [cs, stat]*, Jan. 2017. arXiv: 1701.08230.

[7] C. Dictionary. fairness Meaning in the Cambridge Essential English Dictionary. retrieved from https://dictionary.cambridge.org/dictionary/essential-british-english/fairness, 2018-05-28 at 10:22.

[8] EU. EU GDPR Information Portal. retrieved from http://eugdpr.org/eugdpr.org-1.html, 2018-05-28 at 10:03.

[9] C. for Criminology, P. P. R. C. of Criminology, and C. J. F. S. U. T. Florida. Validation of the compas risk assessment classification instrument. Sept. 2010. retrieved from http://www.northpointeinc.com/downloads/research/COMPASFSUValidation11-09-10-FINAL.pdf, 2018-05-30 at 12:06.

[10] J. Gabbatiss. Refugees could be resettled using computer algorithm that doubles chances of finding them employment. *The Independent*, Jan. 2018. retrieved from http://www.independent.co.uk/news/science/refugees-resettled-computer-algorithm-doubles-chances-employment-dartmouth-college-a8168866.html, 2018-05-19 at 17:47.

[11] O. Hansard. Ethics and Artificial Intelligence - Debate in the House of Commons. retrieved from https://hansard.parliament.uk/Commons/2018-01-17/debates/033B61B9-65CE-417A-91EC-4407B466197F/EthicsAndArtificialIntelligence, 2018-05-28 at 10:46.

[12] J. A. Jeff Larson. How We Analyzed the COMPAS Recidivism Algorithm. *ProPublica*, May 2016. retrieved from https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm, 2018-05-28 at 9:08.

[13] L. Julia, Angwin Jeff. Machine Bias. *ProPublica*, May 2016. retrieved from https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing, 2018-05-28 at 9:16.

[14] Northpointe. Practitioners guide to compas. Aug. 2012. retrieved from http://www.northpointeinc.com/files/technicaldocuments/FieldGuide2081412.pdf, 2018-05-30 at 12:04.

[15] R. O'Connor. Spotify introduces Time Capsule feature that works out your music taste as a teenager. *The Independent*, Sept. 2017. retrieved from http://www.independent.co.uk/arts-entertainment/music/news/spotify-time-capsule-feature-old-songs-blink-182-destinys-child-shania-twain-playlist-how-to-get-a7973456.html, 2018-05-19 at 17:54.

[16] ProPublica. Propublica compas data and analysis. May 2016. retrieved from https://github.com/propublica/compas-analysis, 2018-05-30 at 12:09.

[17] I. T. Union. AI for Good Global Summit 2018. retrieved from https://www.itu.int/en/ITU-T/AI/2018/Pages/default.aspx, 2018-05-27, at 12:30.

[18] J. van der Pligt. Decision Making, Psychology of. In N. J. Smelser and P. B. Baltes, editors, *International Encyclopedia of the Social & Behavioral Sciences*, pages 3309–3315. Pergamon, Oxford, 2001.

[19] A. W. Flores, K. Bechtel, and C. Lowenkamp. False Positives, False Negatives, and False Analyses: A Rejoinder to "Machine Bias: There's Software Used Across the Country to Predict Future Criminals. And it's Biased Against Blacks.". *Federal probation*, 80, Sept. 2016.

[20] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi. Discrimination in Decision Making: Humans vs. Machines. page 54.

6 Appendix

The appendix contains the code used for the analysis in section 4. It is written in R.

```
library(dplyr)
library(ggplot2)
library(survival)
library(ggfortify)
library(plotly)


## load and preprocess data
## The dataset used here is the COMPAS dataset provided by ProPublica
#(https://github.com/propublica/compas-analysis)
f <- file.choose()
raw_data <- read.csv(f)
nrow(raw_data)

df <- dplyr::select(raw_data, age, c_charge_degree, race, age_cat, score_text, sex, p
                    days_b_screening_arrest, decile_score, is_recid, two_year_recid, c
  filter(days_b_screening_arrest <= 30) %>%
  filter(days_b_screening_arrest >= -30) %>%
  filter(is_recid != -1) %>%
  filter(c_charge_degree != "O") %>%
  filter(score_text != 'N/A')
nrow(df)

# summary of race distribution

summary(df$race)
summary(df$score_text)
xtabs(~ score_text + race, data=df)

# Compute statistical parity: Statistical paritiy means E[d(X)|g(X)]=E[d(X)]
#where d is a decision rule (here the assigned decile "category"
#(low, medium high)) and g is the group membership (here race)

# Compute E[d(X)]

A <- c(mean(df$score_text=="Low"),mean(df$score_text=="Medium"),mean(df$score_text=="

#Compute E[d(X)|g(X)]

AF <- c(mean(df$score_text[df$race=="African-American"]=="Low"),
        mean(df$score_text[df$race=="African-American"]=="Medium"),
        mean(df$score_text[df$race=="African-American"]=="High"))
C <- c(mean(df$score_text[df$race=="Caucasian"]=="Low"),
       mean(df$score_text[df$race=="Caucasian"]=="Medium"),
       mean(df$score_text[df$race=="Caucasian"]=="High"))
H <- c(mean(df$score_text[df$race=="Hispanic"]=="Low"),
       mean(df$score_text[df$race=="Hispanic"]=="Medium"),
       mean(df$score_text[df$race=="Hispanic"]=="High"))


Scores <- c("a.Low_Score","b.Medium_Score","c.High_Score")
data <- data.frame(Scores,AF,C,H,A)

#plotting
plot_ly(data, x = ~Scores, y = ~AF, type = 'bar', name = 'Afican-American') %>%
  add_trace(y = ~C, name = 'Caucasian') %>% add_trace(y = ~H, name = 'Hispanic') %>% a
  layout(title = "Measure␣of␣Statistical␣Parity",yaxis = list(title = 'Percentage␣exp



#Compute Conditional Statistical parity: Mathematically: E[d(X)|(l(X),g(X)] = E[d(X),
#where d is a decision roule (here the assigned decile "category" (low, medium high))
#and g is the group membership (here race), l is a projection of all the attributes to
#the "legitimate" attributes. Legitimate attribute chosen in this case: prior convict


#plot a histogram of the number of prior convictions, choose to split into three grou
hist(df$priors_count,freq=FALSE, xlab='Prior␣Convictions',ylab='Frequency',
     main='Distribution␣of␣Prior␣Convictions', col='lightgreen')

# 3 groups: low number of prior convictions: LNP (priors_count in [0,2])
#           medium number of prior convictions: MNP (priors_count in [3,10])
#           high number of prior convictions: HNP (prior_count in [11,inf])

#split into three subdatasets according to number-of-prior-conviction-groups

Low_conv<- subset(df, priors_count <3)

Med_conv <- subset(df,priors_count > 2 & priors_count <11)

High_conv <- subset(df,priors_count >10)

# look at Low_conv:

# E[d(X)|l(X)]
A <- c(mean(Low_conv$score_text=="Low"),
       mean(Low_conv$score_text=="Medium"),
       mean(Low_conv$score_text=="High"))

#E[d(X)|l(X),g(X)]
AF <- c(mean(Low_conv$score_text[Low_conv$race=="African-American"]=="Low"),
        mean(Low_conv$score_text[Low_conv$race=="African-American"]=="Medium"),
        mean(Low_conv$score_text[Low_conv$race=="African-American"]=="High"))

C <- c(mean(Low_conv$score_text[Low_conv$race=="Caucasian"]=="Low"),
       mean(Low_conv$score_text[Low_conv$race=="Caucasian"]=="Medium"),
       mean(Low_conv$score_text[Low_conv$race=="Caucasian"]=="High"))

H <- c(mean(Low_conv$score_text[Low_conv$race=="Hispanic"]=="Low"),
       mean(Low_conv$score_text[Low_conv$race=="Hispanic"]=="Medium"),
       mean(Low_conv$score_text[Low_conv$race=="Hispanic"]=="High"))

Scores <- c("a.Low_Score","b.Medium_Score","c.High_Score")

data <- data.frame(Scores,AF,C,H,A)

#plotting
plot_ly(data, x = ~Scores, y = ~AF, type = 'bar', name = 'Afican-American') %>%
  add_trace(y = ~C, name = 'Caucasian') %>% add_trace(y = ~H, name = 'Hispanic') %>% a
  layout(title = "Conditional␣Statistical␣Parity:␣Rates␣with␣few␣prior␣arrests␣(0-2)"


# look at Med_conv:
```

```
# E[d(X)|l(X)]

A <- c(mean(Med_conv$score_text=="Low"),
       mean(Med_conv$score_text=="Medium"),
       mean(Med_conv$score_text=="High"))

#E[d(X)|l(X),g(X)]

AF <- c(mean(Med_conv$score_text[Med_conv$race=="African-American"]=="Low"),
        mean(Med_conv$score_text[Med_conv$race=="African-American"]=="Medium"),
        mean(Med_conv$score_text[Med_conv$race=="African-American"]=="High"))

C <- c(mean(Med_conv$score_text[Med_conv$race=="Caucasian"]=="Low"),
       mean(Med_conv$score_text[Med_conv$race=="Caucasian"]=="Medium"),
       mean(Med_conv$score_text[Med_conv$race=="Caucasian"]=="High"))

H <- c(mean(Med_conv$score_text[Med_conv$race=="Hispanic"]=="Low"),
       mean(Med_conv$score_text[Med_conv$race=="Hispanic"]=="Medium"),
       mean(Med_conv$score_text[Med_conv$race=="Hispanic"]=="High"))

data <- data.frame(Scores,AF,C,H,A)

#plotting
plot_ly(data, x = ~Scores, y = ~AF, type = 'bar', name = 'Afican-American') %>%
  add_trace(y = ~C, name = 'Caucasian') %>% add_trace(y = ~H, name = 'Hispanic') %>% a
  layout(title = "Conditional Statistical Parity: Rates with \n medium prior arrests (

# look at High_conv:

# E[d(X)|l(X)]
A <- c(mean(High_conv$score_text=="Low"),
       mean(High_conv$score_text=="Medium"),
       mean(High_conv$score_text=="High"))

#E[d(X)|l(X),g(X)]
AF <- c(mean(High_conv$score_text[High_conv$race=="African-American"]=="Low"),
        mean(High_conv$score_text[High_conv$race=="African-American"]=="Medium"),
        mean(High_conv$score_text[High_conv$race=="African-American"]=="High"))

C <- c(mean(High_conv$score_text[High_conv$race=="Caucasian"]=="Low"),
       mean(High_conv$score_text[High_conv$race=="Caucasian"]=="Medium"),
       mean(High_conv$score_text[High_conv$race=="Caucasian"]=="High"))

H <- c(mean(High_conv$score_text[High_conv$race=="Hispanic"]=="Low"),
       mean(High_conv$score_text[High_conv$race=="Hispanic"]=="Medium"),
       mean(High_conv$score_text[High_conv$race=="Hispanic"]=="High"))

data <- data.frame(Scores,AF,C,H,A)

#plotting
plot_ly(data, x = ~Scores, y = ~AF, type = 'bar', name = 'Afican-American') %>%
  add_trace(y = ~C, name = 'Caucasian') %>% add_trace(y = ~H, name = 'Hispanic') %>% a
  layout(title = "Conditional Statistical Parity: Rates with high prior arrests (11+)"


# Compute predictive equality
```

```
#Consider two year recidivism rate. Split the dataset in two groups, recidivated or n
#Predictive equality means mathematically: E[d(X)|Y=0,g(X)] = E[d(X)|Y=0] where d is
# (here the assigned decile "category" (low, medium high)) and g is the group members
# and Y is binary for recidivated/not recidivated. We compute false positivse and fals

# create subdatasets

No_rec<- subset(df, two_year_recid==0)

Rec<- subset(df,  two_year_recid==1)

# look at No_rec:

# E[d(X)|Y=0]

A <- c(mean(No_rec$score_text=="Low"),
       mean(No_rec$score_text=="Medium"),
       mean(No_rec$score_text=="High"))


#E[d(X)|l(X),g(X)]

AF <- c(mean(No_rec$score_text[No_rec$race=="African-American"]=="Low"),
        mean(No_rec$score_text[No_rec$race=="African-American"]=="Medium"),
        mean(No_rec$score_text[No_rec$race=="African-American"]=="High"))

C <- c(mean(No_rec$score_text[No_rec$race=="Caucasian"]=="Low"),
       mean(No_rec$score_text[No_rec$race=="Caucasian"]=="Medium"),
       mean(No_rec$score_text[No_rec$race=="Caucasian"]=="High"))

H <- c(mean(No_rec$score_text[No_rec$race=="Hispanic"]=="Low"),
       mean(No_rec$score_text[No_rec$race=="Hispanic"]=="Medium"),
       mean(No_rec$score_text[No_rec$race=="Hispanic"]=="High"))

Scores <- c("a.Low_Score","b.Medium_Score","c.High_Score")
data <- data.frame(Scores,AF,C,H,A)

#plotting
plot_ly(data, x = ~Scores, y = ~AF, type = 'bar', name = 'Afican-American') %>%
  add_trace(y = ~C, name = 'Caucasian') %>% add_trace(y = ~H, name = 'Hispanic') %>% a
  layout(title = "Predictive equality - scoring among NOT recidivated defendants",yax

# look at Rec:

# E[d(X)|Y=1]
A <- c(mean(Rec$score_text=="Low"),
       mean(Rec$score_text=="Medium"),
       mean(Rec$score_text=="High"))


#E[d(X)|l(X),g(X)]
AF <- c(mean(Rec$score_text[Rec$race=="African-American"]=="Low"),
        mean(Rec$score_text[Rec$race=="African-American"]=="Medium"),
        mean(Rec$score_text[Rec$race=="African-American"]=="High"))

C <- c(mean(Rec$score_text[Rec$race=="Caucasian"]=="Low"),
       mean(Rec$score_text[Rec$race=="Caucasian"]=="Medium"),
       mean(Rec$score_text[Rec$race=="Caucasian"]=="High"))
```

```
H <- c(mean(Rec$score_text[Rec$race=="Hispanic"]=="Low"),
        mean(Rec$score_text[Rec$race=="Hispanic"]=="Medium"),
        mean(Rec$score_text[Rec$race=="Hispanic"]=="High"))

data <- data.frame(Scores,AF,C,H,A)

#plotting
plot_ly(data, x = ~Scores, y = ~AF, type = 'bar', name = 'Afican-American') %>%
  add_trace(y = ~C, name = 'Caucasian') %>% add_trace(y = ~H, name = 'Hispanic') %>% a
  layout(title = "Predictive equality - scoring among recidivated defendants",yaxis =


#calibration

# Mathematically: Proportion(Y=1|s(X),g(X)) = Proportion(Y=1|s(X)) where Y=1 means re
# g expresses the group membership (here race) and s is the given riskscore

# Split in datasets according to decile score

Low<- subset(df, score_text =="Low")

Med<- subset(df, score_text =="Medium")

High<- subset(df, score_text =="High")

#compute rate of recidivism for each race


Low_Score <- c(sum(Low$two_year_recid == 1 & Low$race == "African-American")/sum(Low$
                sum(Low$two_year_recid == 1 & Low$race == "Caucasian")/sum(Low$race ==
                sum(Low$two_year_recid == 1 & Low$race == "Hispanic")/sum(Low$race ==
Medium_Score <- c(sum(Med$two_year_recid == 1 & Med$race == "African-American")/sum(M
                   sum(Med$two_year_recid == 1 & Med$race == "Caucasian")/sum(Med$race
                   sum(Med$two_year_recid == 1 & Med$race == "Hispanic")/sum(Med$race =
High_Score <- c(sum(High$two_year_recid == 1 & High$race == "African-American")/sum(H
                 sum(High$two_year_recid == 1 & High$race == "Caucasian")/sum(High$rac
                 sum(High$two_year_recid == 1 & High$race == "Hispanic")/sum(High$race

Races <- c("African-American","Caucasian","Hispanic")
data <- data.frame(Races,Low_Score,Medium_Score,High_Score)


#plotting
plot_ly(data, x = ~Races, y = ~Low_Score, type = 'bar', name = 'Low Decile Score') %>%
  add_trace(y = ~Medium_Score, name = 'Medium Decile Score') %>% add_trace(y = ~High_
  layout(title = "Calibration Measure (recidivism after 12 month)",yaxis = list(title
```