FairPrep: Promoting Data to a First-Class Citizen in Studies on Fairness-Enhancing Interventions

By Schelter Sebastian, Yuxuan He, Jatin Khilnani, and Julia Stoyanovich

Papadogiannakis Emmanouil

CS-562: Advanced Databases Topics
Computer Science Department
University of Crete
Comparison of Papers

- Data related systems and their **fairness**
- **Bias** in early stages of a pipeline
- **Origin** of bias: Models are trained on biased data
- Different approach and methodology
- First paper describes the problem, second paper helps with the solutions
Outline

• Introduction
• Previous Work
• Design
• System Evaluation
• Conclusion
Background
Introduction

• Research on **fairness** focus on the end of the pipeline

• Improving data quality and controlling bias are limited if **early stages** are ignored
FairPrep

- Open-source framework

- Evaluating fairness-enhancing interventions

- Features based on good-practices
  1) Modular data lifecycle
  2) Re-use of fairness metrics implementations
  3) Integration of useful operations
Use case

Alice is a data scientist that works on a **classification** problem following best practices

She uses historic **demographic** data to train models

Selects the classifier with best **accuracy**
Use case

- Alice examines the accuracy of the classifier
- She observes that the accuracy is lower for middle-aged women
- Alice hypothesizes that age is an important feature and tries to remove bias
Use case

- There were no **fairness** issues in the workflow but the classifier was **biased**

- Other challenges:
  - How to integrate more fairness evaluation metrics?
  - Can the effect of fairness-enhancing interventions be quantified?
  - How to follow best-practices when integrating fairness metrics?
Previous Work
Shortcomings

- Machine Learning models are evaluated using a *test set*

- The test set should be *isolated* from the process of model selection

- Popular benchmarking frameworks violate this restriction
Shortcomings

• There are some hyperparameter candidates that need to be evaluated

• Should not be done on the test set. It would violate the isolation restriction

• A validation set should be used to select the hyperparameters
Shortcomings

- Hyperparameters should be tuned even for baseline algorithms
- No guarantee that the default configuration will find a good solution
- Affects fairness and accuracy
Shortcomings

• Records with **missing values** were ignored

<table>
<thead>
<tr>
<th></th>
<th>Name</th>
<th>Age</th>
<th>Gender</th>
<th>Height</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>lynda</td>
<td>10.0</td>
<td>F</td>
<td>125</td>
<td>5/21/2018</td>
</tr>
<tr>
<td>1</td>
<td>tom</td>
<td>NaN</td>
<td>M</td>
<td>135</td>
<td>7/21/2018</td>
</tr>
</tbody>
</table>

• Major issue if **protected groups** are more likely to have missing values

• Numeric features were not **normalized**
Design
Design Principles

• **User code** can only interact with the training set

• Explicit data lifecycle

• **Users can configure and implement components**
Data Lifecycle

1) Train models on the training set
   - Multiple components in this step

2) User selects the best model
   - Based on metrics computed on the validation set

3) Application of selected model on test set
   - Framework reports accuracy and fairness
Data Lifecycle

Model selection on training set and validation set

raw dataset

raw trainset → trainset

complete trainset → featurised trainset

preprocessed trainset

resampler [optional]

missing value handler

feature transform

pre-processor [optional]

classifier

post-processor [optional]

predictions for valset

predictions for valset
Data Lifecycle

1. Model selection on training set and validation set

2. User-defined choice of best model

- raw dataset
  - raw trainset
  - trainset
  - complete trainset
  - featurised trainset
  - preprocessed trainset
  - classifier
  - post-processor [optional]
  - predictions for valset

- raw valset
  - complete valset
  - featurised valset
  - preprocessed valset
  - predictions for valset

- resampler [optional]
- missing value handler
- feature transform
- pre-processor [optional]
- fit

- fit
- fit
- fit

- metrics for model 1
- metrics for model 2
- ...
Data Lifecycle

1. Model selection on training set and validation set
   - raw trainset → trainset → complete trainset → featurised trainset → preprocessed trainset
   - resampler [optional] → missing value handler → feature transform → pre-processor [optional] → classifier
   - fit

2. User-defined choice of best model
   - metrics for model 1
   - metrics for model 2
   - ... metrics for model n

3. Application of best model (and corresponding transformations) on the test set
   - raw testset → complete testset → featurised testset → preprocessed testset
   - missing value handler → feature transform → pre-processor [optional] → classifier → post-processor [optional]
   - fit
   - predictions for testset → metrics for testset
Evaluation
Experimental Evaluation

- How can FairPrep be used
- How it overcomes shortcomings of previous work
- Two experiments:
  - Hyperparameter tuning
  - Incomplete data
Experiment 1 - Hyperparameter Tuning

• **Task:** Predict credit risk of an individual

• **Sensitive attribute** is gender

• **Two versions:**
  - Default hyperparameters of baseline model
  - With hyperparameter tuning
Experiment 1 - Hyperparameter Tuning

• Two baseline models:
  - Logistic regression
  - Decision tree

• With and without Hyperparameter tuning for each model

• Metrics computed on test set
Experiment 1 - Hyperparameter Tuning

- Results showed that tuned hyperparameters result in higher accuracy and lower variance.

Accuracy & FN Difference

Accuracy and FP difference
Experiment 2 - Incomplete Data

- **Predict** if an individual earns more than 50,000$ per year

- **Sensitive attributes** are race and gender

- **Fairness between**
  - Privileged group of white individuals (85%)
  - Unprivileged group of non-white individuals (15%)
Experiment 2 - Incomplete Data

• Observed that missing values do not occur at random

• They are affected by:
  - Marital status
  - Native country
  - Age group
Experiment 2 - Incomplete Data

• Strategies to treat incomplete data:
  1) Remove incomplete records
  2) Replace missing value with most frequent value of the specific feature
  3) Impute missing values using “datawig”

• Datawig is a deep learning model for imputation of missing values in tables

• Techniques that don’t remove incomplete records have similar accuracy
Experiment 2 - Incomplete Data

- Slightly higher **accuracy** when incomplete data are included

- Incomplete records can be classified without any decrease in:
  - Accuracy
  - Fairness
Conclusion
Summary

- FairPrep designed to overcome shortcomings of previous work

- Specific data lifecycle that helps data scientist
  - follow best practices
  - work on fairness-enhancing interventions
Open Problems

• Use more complex discrimination metrics in experiments (e.g. Risk difference, Odds ratio)

• Can be extended to include more fairness-enhancing interventions

• Currently limited to binary classification problems
Personal Evaluation

• **Useful.** Not only for academic purposes but for practical too (industry, research)

• No apparent novelty. A more *systematic* approach to a known problem

• No non-trivial contributions

• Well-justified, Well-written, Easy to follow
Questions?
For more...

- Github Repository: https://github.com/DataResponsibly/FairPrep

- Other framework: https://github.com/algofairness/fairness-comparison

Thank You!
Backup Slides
Definitions

- **Accuracy:**
  "How well a classifier correctly identifies a condition"

\[
\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} = \frac{TP + TN}{TP + TN + FP + FN}
\]

- **False Positive Rate:**
  "Probability of giving a positive result when the true value is negative"

\[
\text{FPR} = \frac{\text{False Positives}}{\text{Actual Negatives}} = \frac{FP}{FP + TN} = \frac{FP}{\text{Ground Truth Negatives}}
\]
Data Lifecycle – Step 1

1) Resampling of training data
2) Treatment of records with missing values
3) Feature transformation (e.g. scaling)
4) Pre-processing interventions
5) Model training
6) Computation of predictions on training and validation set
7) Post-processing interventions
Missing Values

• The “positive” label is associated with 24% of the complete records but only 14% of records with missing values

• Married individuals are the majority of complete records

• Incomplete records have most frequently the “never married” value for marital-status

• The attribute “native-country” is missing four time more frequently for non-white individuals than for white individuals