5243 Project 4
Machine Learning Fairness

Group 7: Louis Cheng, Sameer Kolluri, Sangmin Lee, Fu Wang, Judy Wu

Algorithm 3: Maximizing Fairness under Accuracy Constraints

In this section, to achieve fairness we will use an in-processing method, where we modify the learning algorithm. We aim to implement a convex margin-based classifier that avoids Disparate Treatment and Disparate Impact and still achieves "business necessity." In this part, we implemented 2 models, Gamma Logistic Regression Model and Fine-Gamma Logistic Regression Model.

Gamma-LR Model

$$\begin{aligned} &\text{minimize} && \boxed{\left|\frac{1}{N}\sum_{i=1}^{N}(\mathbf{z}_i - \bar{\mathbf{z}})\, d_{\boldsymbol{\theta}}(\mathbf{x}_i)\right|} \\ &\text{subject to} && L(\boldsymbol{\theta}) \le (1+\gamma) L(\boldsymbol{\theta}^*), \end{aligned}$$

Covariance between sensitive variable and the distance between decision bound and feature vector

The optimal loss function from optimizing the loss function of Logistic Regression without constraint

Fine-Gamma-LR Model

$$\begin{aligned} &\text{minimize} && \left|\frac{1}{N}\sum_{i=1}^{N}(\mathbf{z}_i - \bar{\mathbf{z}})\, \boldsymbol{\theta}^T\mathbf{x}_i\right| \\ &\text{subject to} && L_i(\boldsymbol{\theta}) \le (1+\gamma_i) L_i(\boldsymbol{\theta}^*) \quad \forall i \in \{1, \ldots, N\}, \end{aligned} \tag{8}$$

The individual loss function associated with the i-th point in the training set.

The individual optimal loss function from optimizing the loss function of Logistic Regression without constraint

COMPAS Dataset Preprocessing

1. Removed columns with over 25% missing values, fill other missing values with 0
2. Removed columns containing dates
3. Encoded the categorical variables
4. Added intercept to features for further training
5. Selected sensitive feature as *x_sensitive* and remove the "*race*" column from feature set

Model Evaluation Metrics

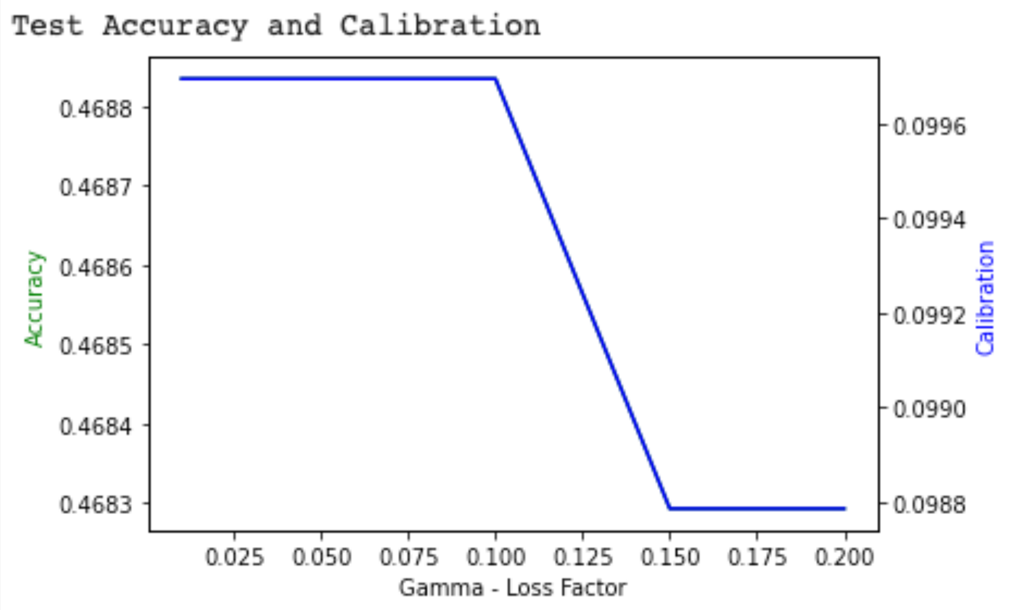
1. Accuracy
2. $P(\hat{Y} = Y | x_{sensitive} = 1) - P(\hat{Y} = Y | x_{sensitive} = 0)$

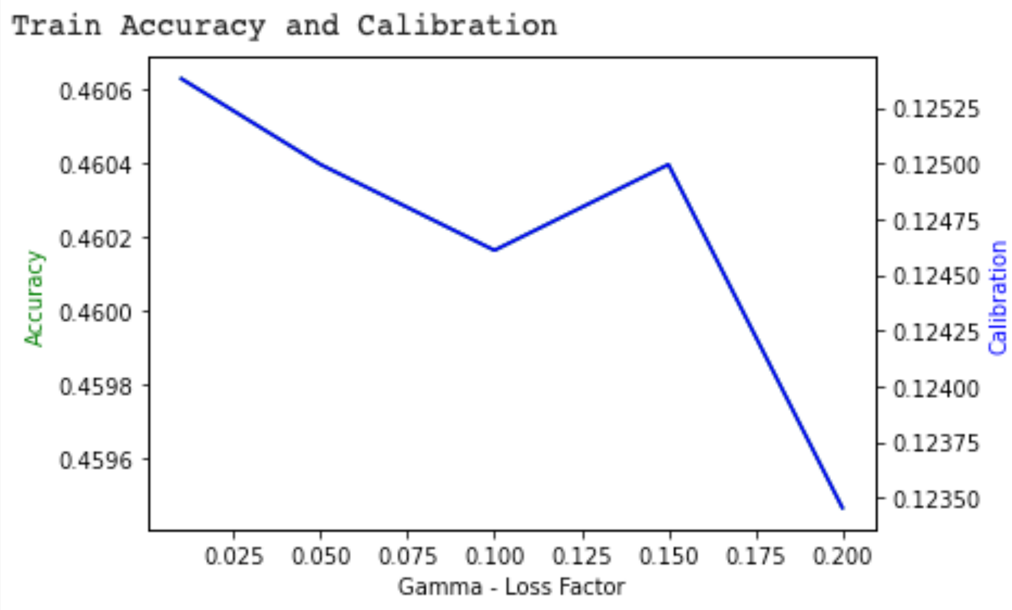
Model Performance Evaluation

Baseline Logistic Regression Model

```
train accuracy: 0.9639225181598063
test accuracy: 0.9564270152505446
train calibration: 0.001321664218600693
test calibration: -0.01036096256684493
```

Gamma- LR Model





Fine-Gamma- LR Model

```
Using gamma =  0.1
train accuracy: 0.47046004842615013
test accuracy: 0.49019607843137253
train calibration difference: 0.13672643056429867
test calibration difference: 0.05113636363636359
```

Algorithm 6: Handling Conditional Discrimination

In A6, instead of using an in-processing method to deal with discrimination like we did above, we are going to use pre-processing methods. The idea is to balance the data set before using it to train the model. To be more specific, we implemented the local massaging and local preferential sampling algorithm as pre-processing methods below.

Local Massaging

In the locally massaging algorithm, we alter the training data to achieve our "debiasing" goal. Specifically, we relabel data points that are close to the decision boundary.

Algorithm 1: Local massaging

input : dataset $(\mathbf{X}, \mathbf{s}, \mathbf{e}, \mathbf{y})$
output: modified labels $\hat{\mathbf{y}}$

`PARTITION` $(\mathbf{X}, \mathbf{e})$ (Algorithm 3);
for *each partition* $X^{(i)}$ **do**
  learn a ranker $\mathcal{H}_i : X^{(i)} \rightarrow y^{(i)}$;
  rank males using $\mathcal{H}_i$;
  `relabel DELTA` (male) males that are the closest to the decision boundary from $+$ to $-$ (Algorithm 4);
  rank females using $\mathcal{H}_i$;
  `relabel DELTA` (female) females that are the closest to the decision boundary from $-$ to $+$
end

Local Preferential Sampling

The intuition behind this is to remove the samples and resample them close to the decision boundary.

Algorithm 2: Local preferential sampling

input : dataset $(\mathbf{X}, \mathbf{s}, \mathbf{e}, \mathbf{y})$
output: resampled dataset (a list of instances)

`PARTITION` $(\mathbf{X}, \mathbf{e})$ (see Algorithm 3);
for *each partition* $X^{(i)}$ **do**
  learn a ranker $\mathcal{H}_i : X^{(i)} \rightarrow y^{(i)}$;
  `rank` males using $\mathcal{H}_i$;
  `delete` $\frac{1}{2}$`DELTA` (male) (see Algorithm 4) males $+$ that are the closest to the decision boundary;
  `duplicate` $\frac{1}{2}$`DELTA` (male) males $-$ that are the closest to the decision boundary;
  `rank` females using $\mathcal{H}_i$;
  `delete` $\frac{1}{2}$`DELTA` (female) females $-$ that are the closest to the decision boundary;
  `duplicate` $\frac{1}{2}$`DELTA` (female) females $+$ that are the closest to the decision boundary;
end

Algorithm 3: subroutine PARTITION$(\mathbf{X}, \mathbf{e})$

find all unique values of e: $\{e_1, e_2, \ldots, e_k\}$;
for $i = 1$ **to** k **do**
  make a group $X^{(i)} = \{X : e = e_i\}$;
end

Algorithm 4: subroutine DELTA(gender)

return $G_i |p(+|e_i, \text{gender}) - p^{\star}(+|e_i)|$,
where $p^{\star}(+|e_i)$ comes from (Eq. (4)),
G_i is the number of gender people in $X^{(i)}$;

**Here we defined Gi as a constant (hyperparameter)

COMPAS Dataset Preprocessing

Similar steps as done in Algorithm 3, except:

1. Kept columns containing dates
2. Set the correlation threshold to be on less than 0.2 and removed the features that don’t satisfy this condition

Model Performance Evaluation

Baseline Random Forest Model

The rate of Recidivism

rate_af,rate_ca
(0.51,0.26)

-Accuracy
-Confusion Matrix
-Calibration

```
0.9766937669376694
[[932  43]
 [  0 870]]
0.02107992678462478
```

Local Massaging Model

The rate of Recidivism

rate_af,rate_ca
(0.46, 0.44)

-Accuracy
-Confusion Matrix
-Calibration

```
0.9463414634146341
[[936  70]
 [ 29 810]]
0.04396028681742969
```

Local Preferential Sampling Model

The rate of Recidivism

rate_af,rate_ca
(0.5, 0.4)

-Accuracy
-Confusion Matrix
-Calibration

```
0.9588075880758807
[[929  73]
 [  3 840]]
0.06168413311270449
```

Conclusion

We implemented two different methods to balance the model‘s fairness, namely: the in-processing method from A3 and the pre-processing method in A6. In A3, we did this by adding constraints to the model to achieve accuracy while optimizing the fairness. In A6, we pre-processed the data using local massaging and preferential sampling to achieve a balanced data set. In both algorithm, we can see the accuracy drop in order to achieve fairness.
To this end, we believe that implementing the models above, does result in a drop in accuracy, but in return for compliance with model fairness, is worth a shot.