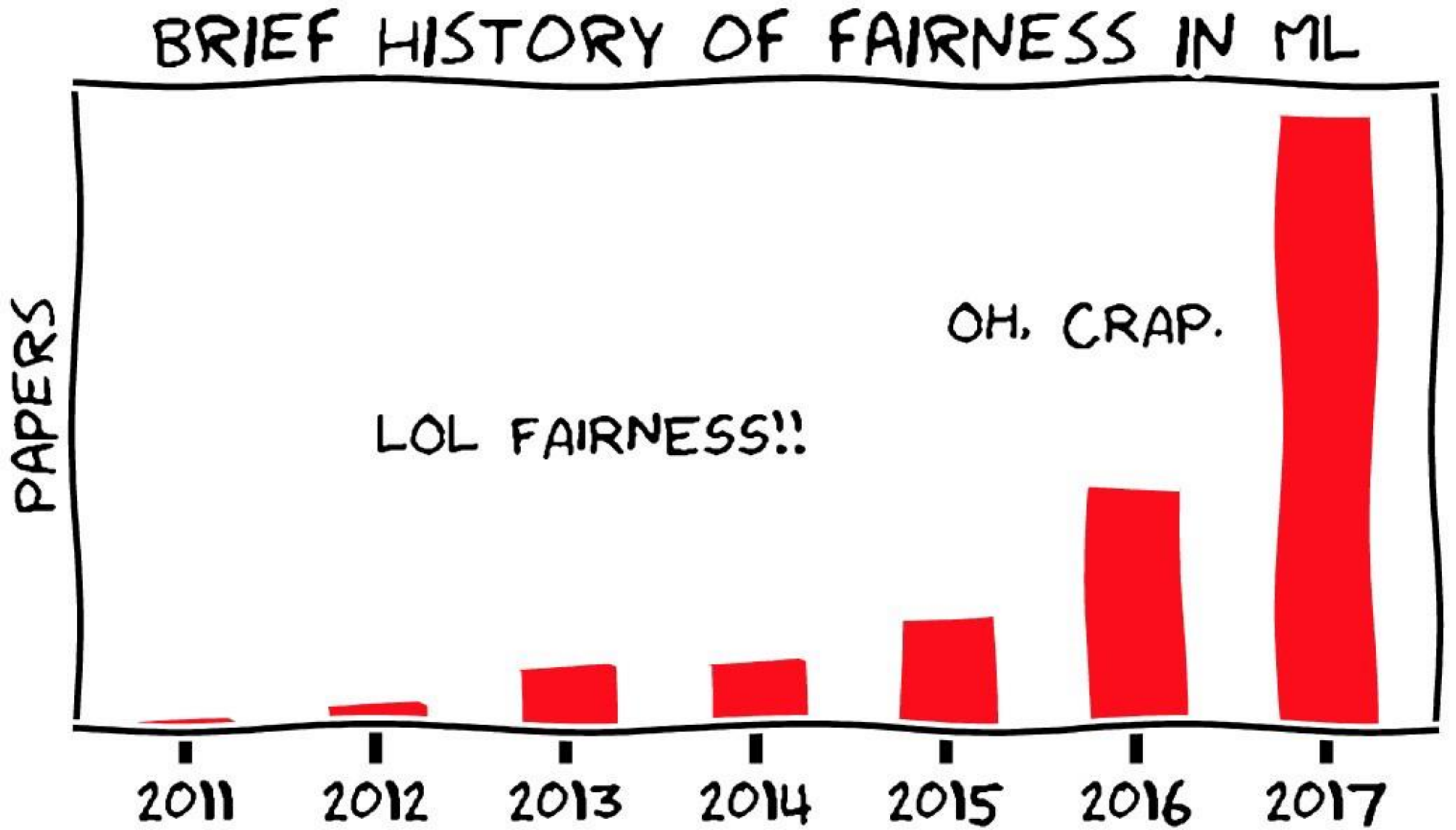
Responsible AI

Lecture 6

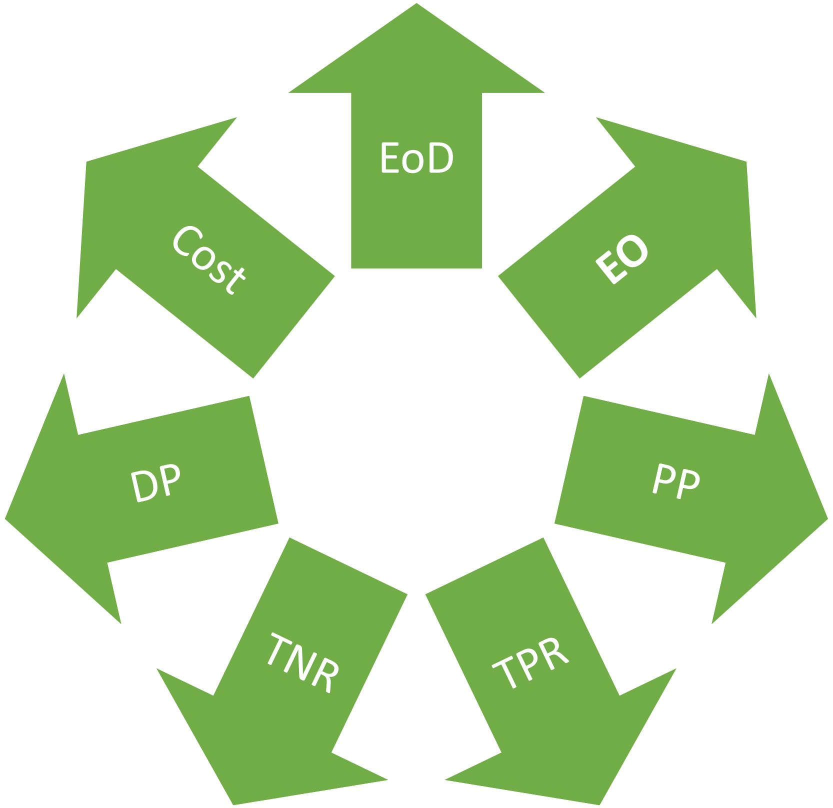
Today

1. Composite feature
2. Additive Counterfactual Fairness (ACF)
3. High level steps for implementing ACF model
4. ACF for classification problems


BRIEF HISTORY OF FAIRNESS IN ML
PAPERS
LOL FAIRNESS!!
OH, CRAP.
2011
2012
2013
2014
2015
2016
2017

Why is fairness needed during Model development?

Fairness Metrics



Composite Features

- Create Boolean combination of features if multiple PII data needs to be handled
- Use statistical and business knowledge – cant be just random

data[‘Combined_protected_group’] = np.where((data[‘WrExLess10’] == 0) & (data[‘MaritalStatus_4.0’] == 0),0, 1)

PS: This allows not only to overcome the issues of reweighing (of handling multipole protected features) but also would allow to combine protected features where the

Problems with other Fairness Algoriths

- CANT HANDLE ALL ALGORITHMS
- TOUGH TO UNDERSTAND
- FAILS WITH MULTIPLE FEATURES
- EITHER REGRESSION OR CLASSIFICATION
- FAILS IF YOU HAVE MIXED PII DATA


BUT WE HAVE A
SOLUTION!!!

What if I change my gender and marital status, will the output change

Additive Counterfactual Fairness

- ACF (additive counterfactually fair) models can be implemented using any machine learning algorithm and are apt for most of regression and classification problems.
- No need for composite features as it can tackle multiple protected features.
- Can handle continuous and categorical protected features.
- ACF may cause a small dip in the accuracy.

ACF: Fundamental

The primary concept of ACF is based on causality. A causal graph is counterfactually fair if the predicted outcome $\hat{Y}$ in the graph does not depend on a descendant of the protected attribute S. For example, a predictive outcome $\hat{Y}$ of defaulter vs non-defaulter for a loan application is typically dependent on credit score, credit amount, disposable income and years of work experience

ACF: under the hood

The ACF attempts to remove these causal dependencies through explicitly mod-elling all input variables as a linear combination of the protected class variables. By taking the residuals as a difference between actual (X) and predicted input variables ($\hat{}$X), we can effectively remove the correlation between the protected classes to the input variables.

ACF: How to

ACF, within the scaffold of counterfactual fairness, is the concept of modelling the relationship between S and features in X by training additive models to predict each feature X*j* (as the outcome feature) with S as the predictors

Then, we can compute the residuals εij between predicted values ($X - \hat{X}$) and true feature values (X) for each observation *i* and non-protected feature Xj. The final model is then trained on the residuals (εij) as features to predict the outcome feature Y.

ACF: Methodology

1. Develop a separate model to predict each of the independent features (non- protected) using protected features as the predictor features.
2. Compute the residuals for each independent feature.
3. Develop a model with Y as response and residuals as predictors.

$$\widehat{X_1} = f_1\left(S_1, S_2, \ldots, S_n\right)$$

$$\vdots$$

$$\widehat{X_n} = f_n\left(S_1, S_2, \ldots, S_n\right)$$

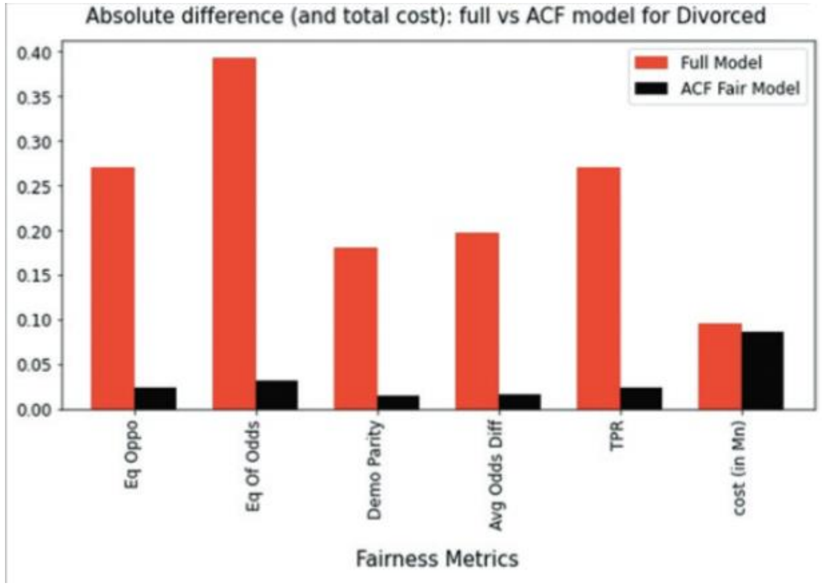
$$\epsilon_{X1} = X_1 - \widehat{X_1}$$

$$\vdots$$

$$\epsilon_{Xn} = X_n - \widehat{X_n}$$

$$\hat{Y} = f_y\left(\epsilon_{X1}, \epsilon_{X2}, \ldots, \epsilon_{Xn}\right)$$

ACF: for Classification



Use one algorithm and test the impact on various PII data

Note the change in accuracy (ACF/F1/..) pre and post ACF

What innovation you can bring in

- Use different algorithm to model each independent features with sensitive features
- Use different methods of finding residuals (actual independent features vs predicted independent features)
- Note that there will be different methods for residual calculation if your independent features is continuous vs when its is binary
- Hyper tune the above two methods
- Optimizing decision boundary

In group of three replicate the ACF method for regression problem

ACF Regression

- Accuracy metrics (MAPE, RMSE, MSE) Regression vs ACF
- Delta in accuracy metrics for each protected features - Regression vs ACF

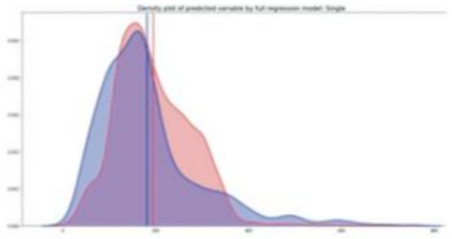
Single	
Mean	1.0744
Skewness	0.4424
Kurtosis	0.4371

Table 5.16 Model accuracy parameters for linear regression vs ACF

	Mean squared error	Root mean squared error	Mean absolute Percentage Error
Linear regression	1991.3528	44.6245	22.8620
ACF	2442.6078	49.4227	28.2511

Linear Regression Model

Table 5.17 Model accuracy differences for various protected groups when using linear regression model

Linear regression model	Mean squared error difference	Root mean squared error difference	Mean absolute percentage error difference
Single	517.3058	6.1346	1.0619
Married	873.3461	10.8661	2.0646
Divorced	917.6351	10.7927	0.6545
No of Dependants less than 3	296.3293	3.4520	4.3778

ACF Model

Table 5.18 Model accuracy differences for various protected groups when using ACF model

ACF model	Mean squared error difference	Root mean squared error difference	Mean absolute percentage error difference
Single	840.4945	9.1932	3.6977
Married	1279.3378	11.8464	12.0020
Divorced	130.5891	1.3272	3.1517
No. of Dependants less than three	773.6410	7.2958	7.7870

Recap

- We begin by separating our dataset into protected features (Sj) and independent features (Xj)
- We create a separate model to predict each independent feature using the protected features as the predictor, or inputs. We represent these predictions as(Xj)
- We calculate the residuals as the difference between the actual and predicted values for the independent features. This is represented by εj.

Residual: Next level

- Absolute value
- Residual squared
- What if residual value is very small after squaring (0.5 square is ??)
    - Over come this by adding M
- When the Y is binary
    - Pearson Residual
    - Deviance Residual

Test ACF via Unfairness

- Create ACF model
- Generate the predictions using the ACF model
- Calculate the error in prediction
- Invert the protected features – change privileged to unprivileged and vice versa
- Predict independent features using inverted protected features (S') as the input (predictors) to the model from above
- Calculate the residuals
- The residuals (ε') act as the input to the ACF model to generate the predictions
- Calculate the error in prediction
- Calculate the counterfactual unfairness (CUF)
- Compare CUF of ACF model vs Baseline model

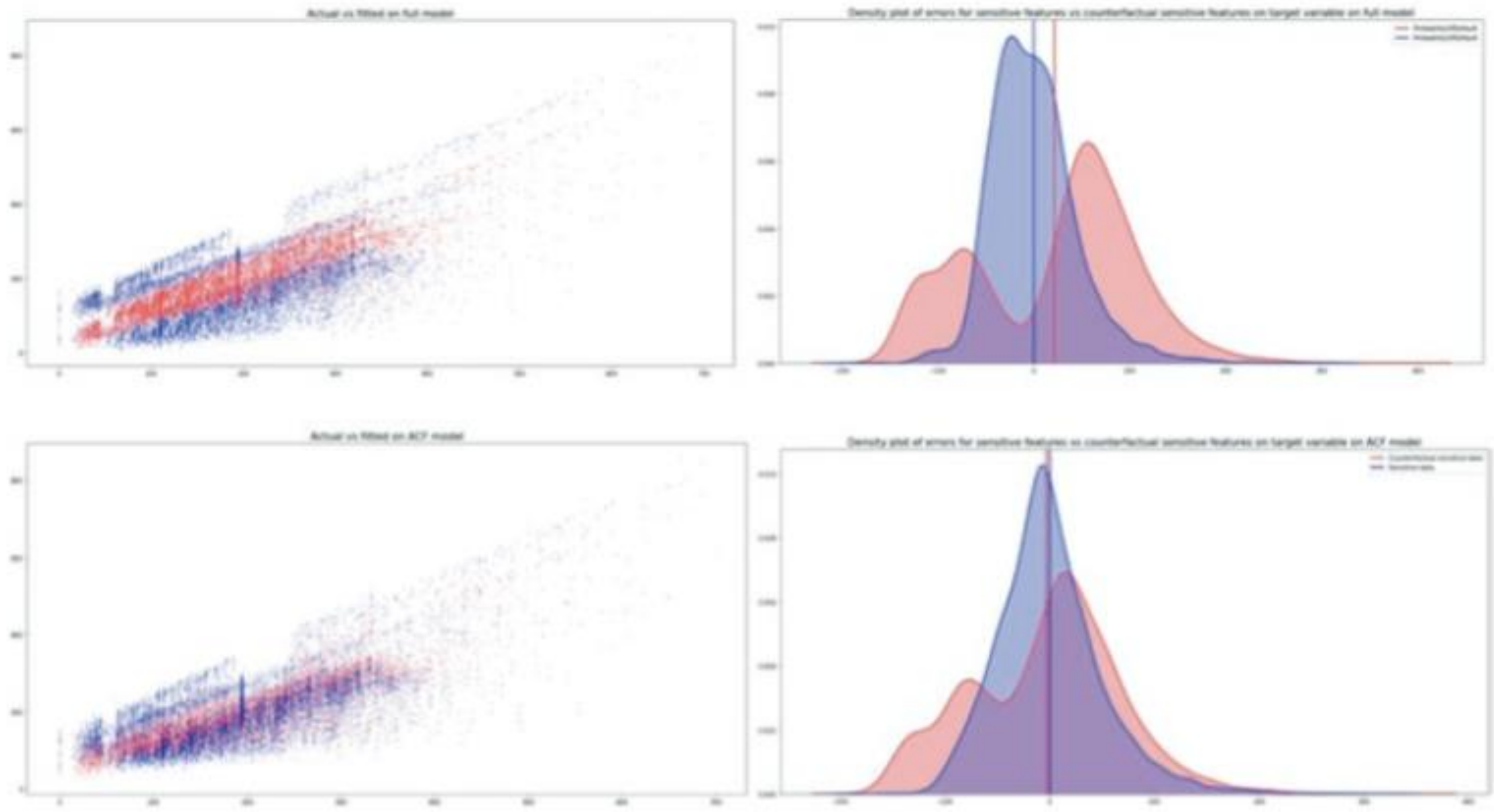
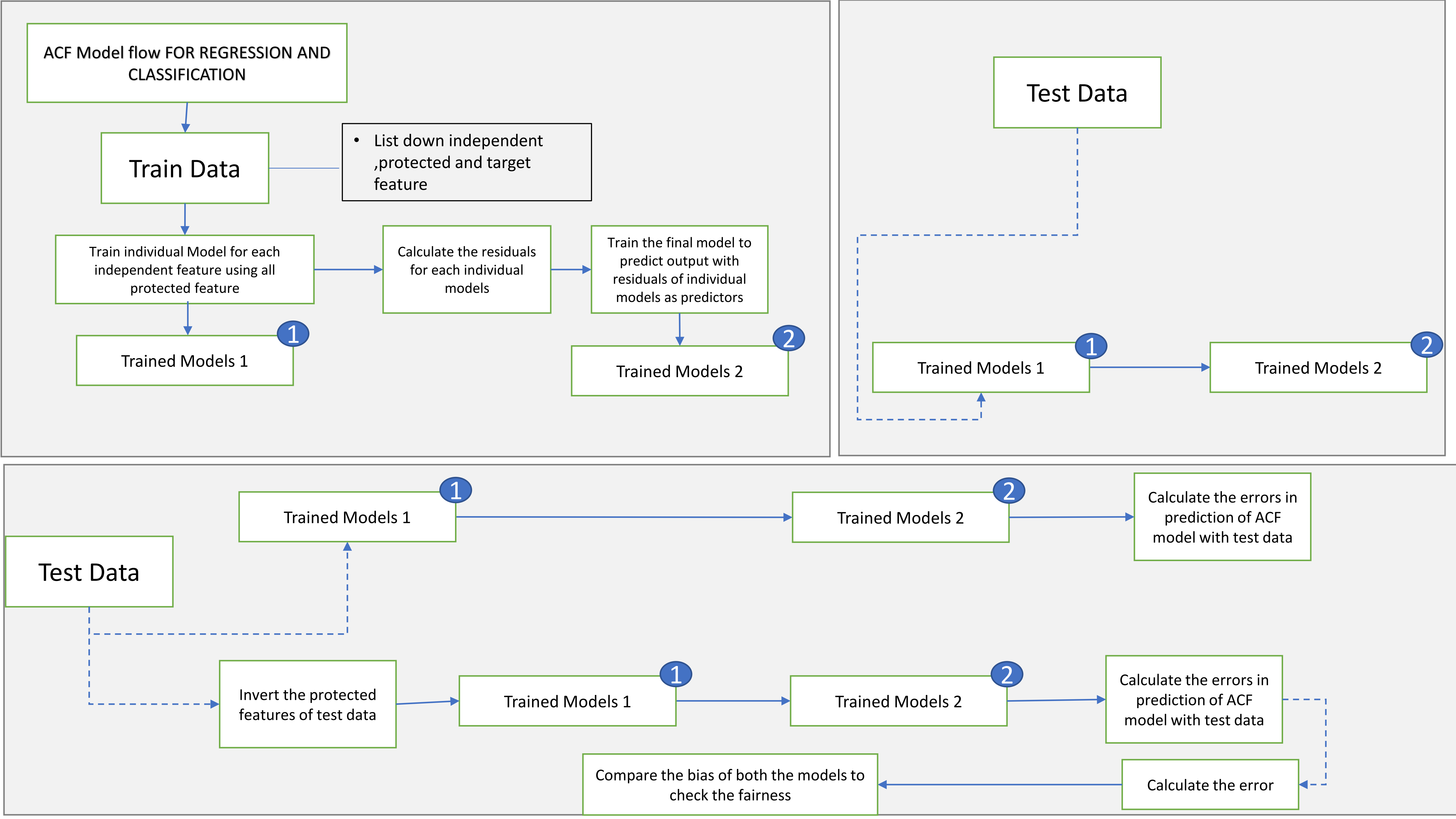
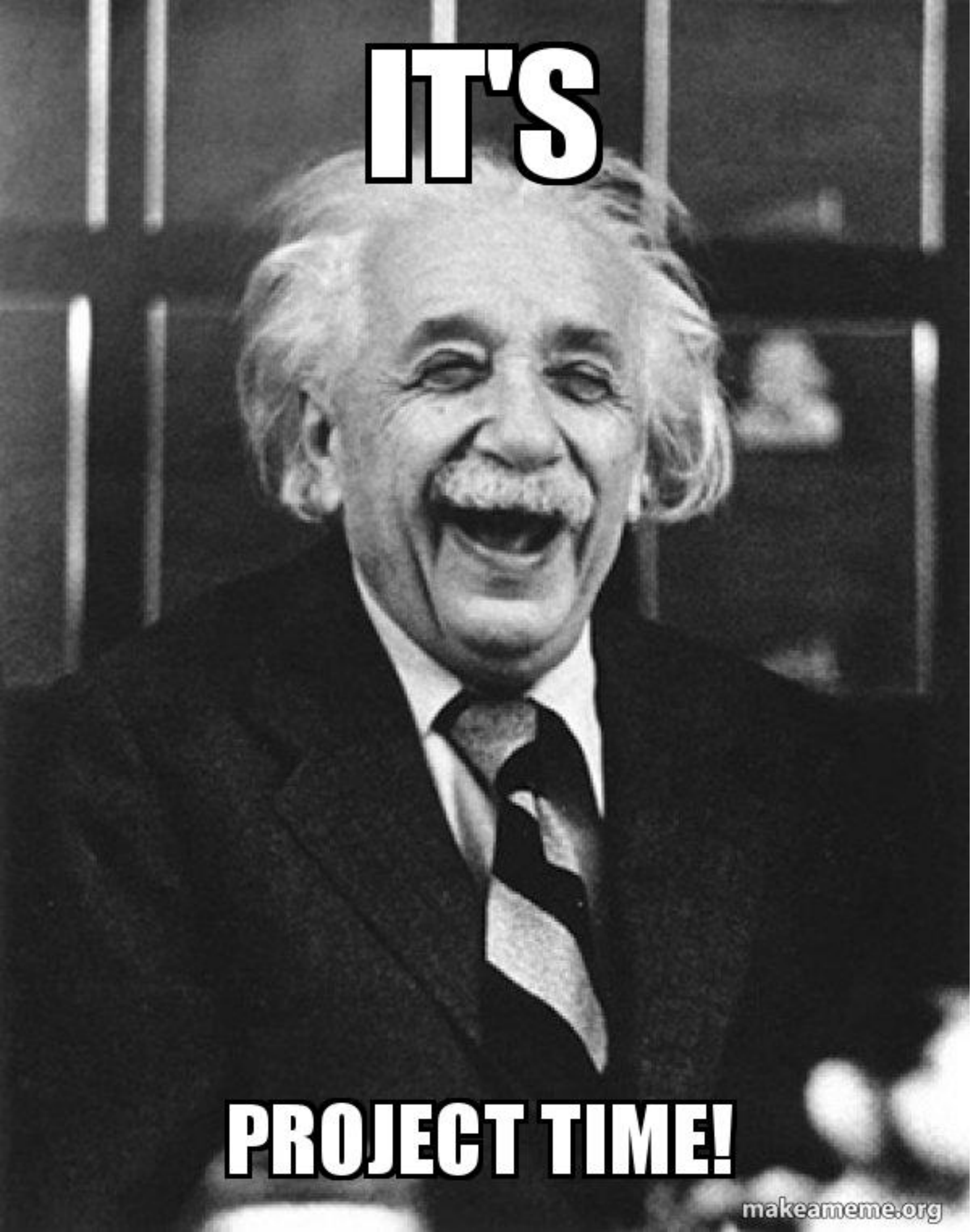


Fig. 5.11 Error vs actual for two models and the impact of counterfactual treatment (red denotes counterfactual sensitive features and blue denotes original)



ACF Model flow FOR REGRESSION AND CLASSIFICATION
Train Data
List down independent ,protected and target feature
Train individual Model for each independent feature using all protected feature
Calculate the residuals for each individual models
Train the final model to predict output with residuals of individual models as predictors
Trained Models 1
1
Trained Models 2
2
Test Data
Trained Models 1
1
Trained Models 2
2
Trained Models 1
1
Trained Models 2
2
Calculate the errors in prediction of ACF model with test data
Test Data
Invert the protected features of test data
Trained Models 1
1
Trained Models 2
2
Calculate the errors in prediction of ACF model with test data
Compare the bias of both the models to check the fairness
Calculate the error



2^{nd} of 3 parts of final project

Implement ACF in your selected data

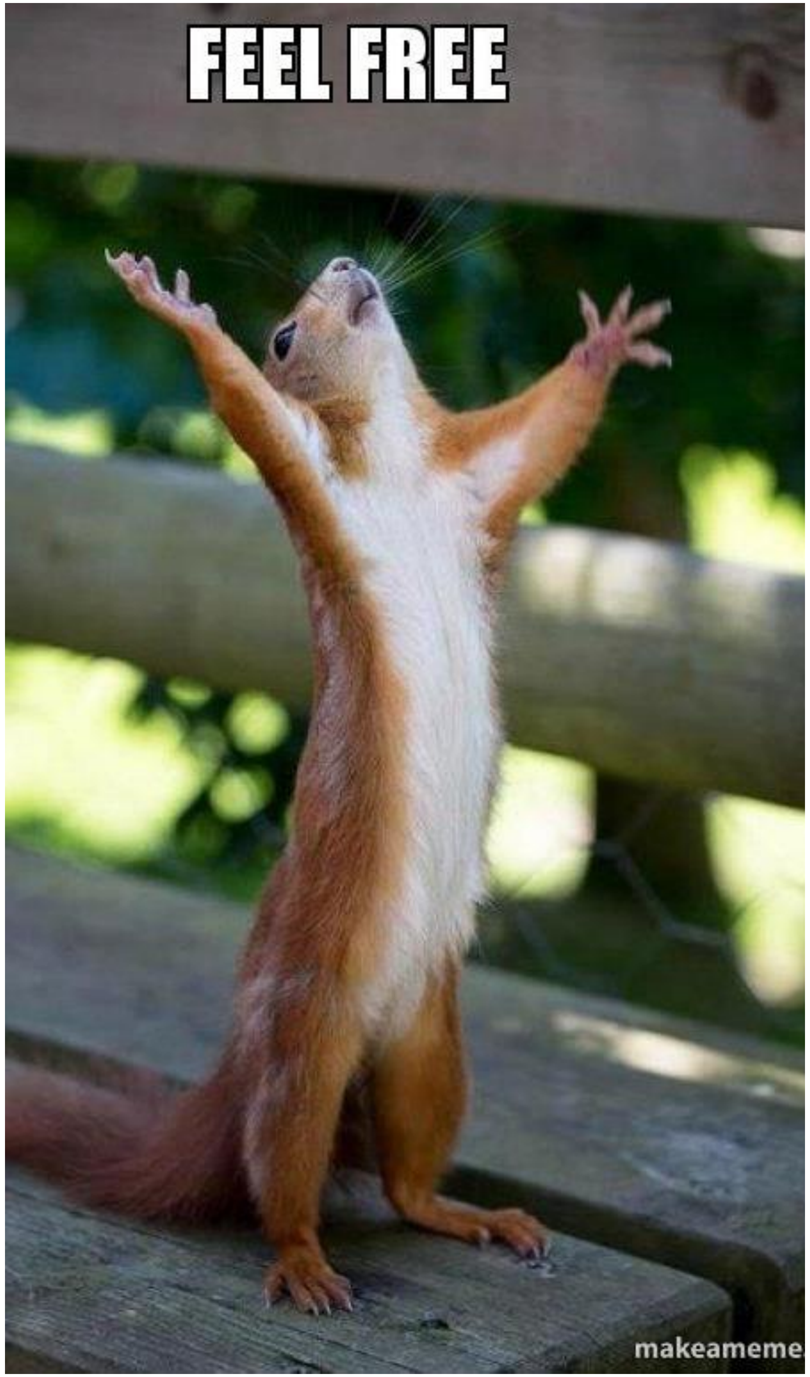
1. All protected features (max 4)
2. All Independent features (max 6)
3. Compare with relevant metrics
4. Calculate CUF
5. Repeat with different residual formula
6. Do step 1 to 5 on DP data
7. Do step 1 to 6 on models with weights (add weights to final model of ACF)

Next

1. Reject option classifier
2. Optimising the ROC
3. Handling multiple features in ROC

Revise:

- **Chapters from book**
- **Fairness Metrics**
- **Classification Algorithms**
- **Decision boundary**


FEEL FREE
makeameme

YOU CAN CALL ME

DID YOU REALLY JUST SEND THAT EMAIL!

KNOCK, KNOCK!