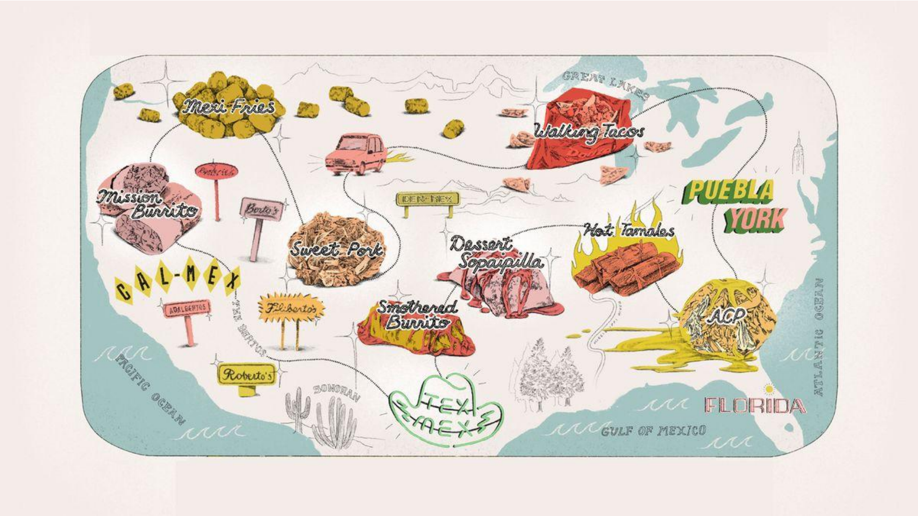
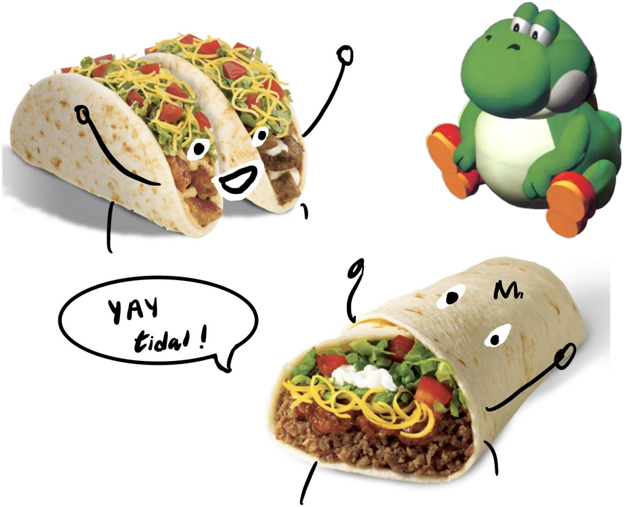
tidaltamu

TETHERED INFORMATICS AND DATA ANALYTICS LAB

Goldman Sachs: Problem

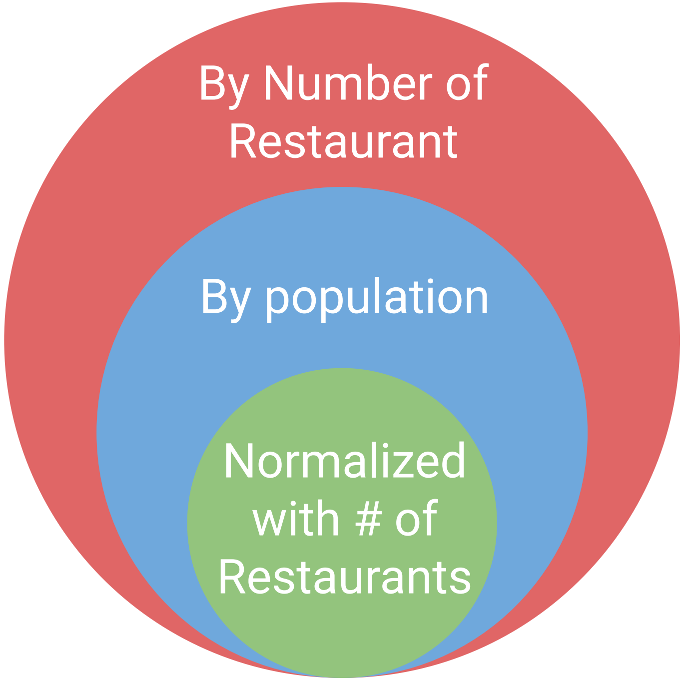
Design a product to visualize various data on restaurants that sell tacos and burritos in order to understand trends across the united states





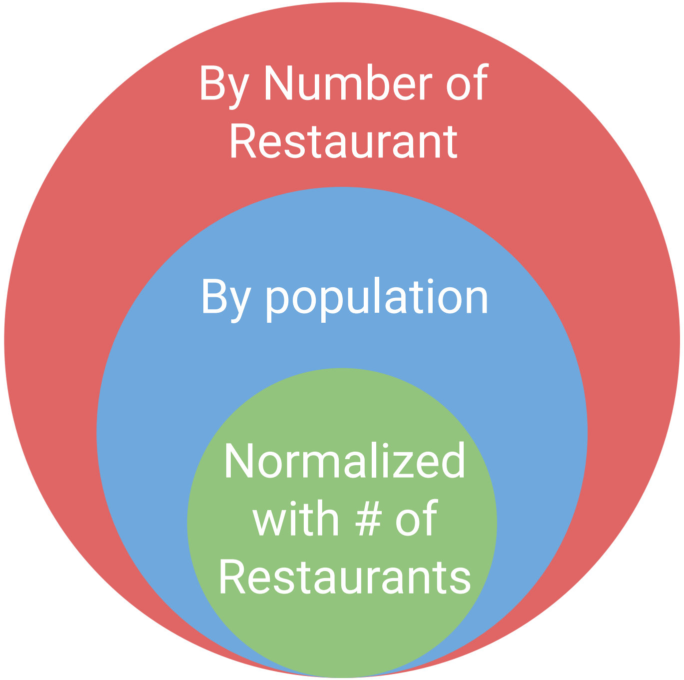
Goldman Sachs: Approach

1. Clean up the data
2. Acquire additional data via census.gov and factual.com
    a. Population of People per state
    b. Ratio of people to restaurants
    c. Ratio of taco and burrito-serving restaurants to all restaurants in a state
3. Create a visual to help compare properties and correlations



Goldman Sachs: Results

Created a python interactive interface with various filtration.



Goldman Sachs: Conclusion

Based on our findings, we now know that:

- California and Texas have highest count of places that sell tacos or burritos
- Alaska has the most places that sell tacos or burritos per person
- The southwestern region has the highest percentage of restaurants that serve tacos and burritos!

If we had more time, we would have:

- Gathered more census data to see where the restaurants are in relation to income, household number, etc.
- Create a GUI to identify a taco or burrito restaurant to go to based on their own personal preferences by answering questions

ConocoPhillips: Problem

Be able to predict when equipment will fail with the data from 107 sensors (very important for run time)

1000 Failures

59,000 Non-failures

59,000/60,000=.98333

If you guess all zeros, you will still achieve high accuracy score!!!

ConocoPhillips: Approach

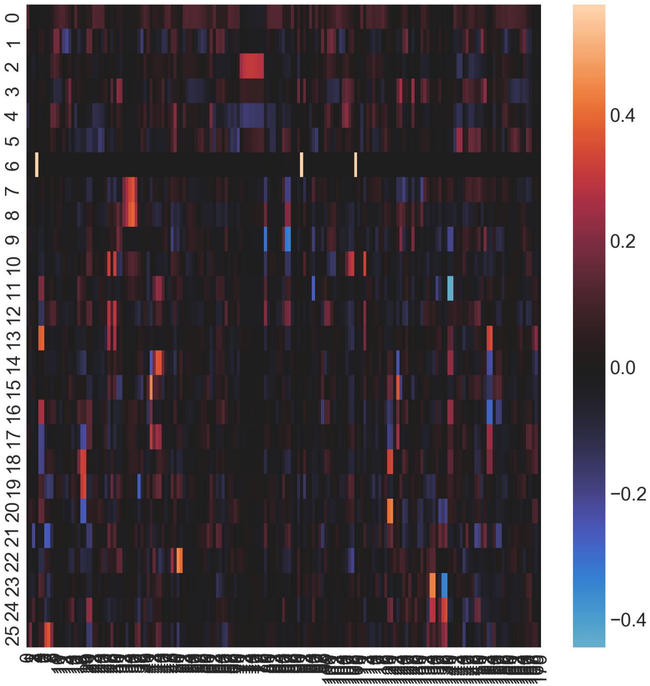
1. Clean up the data
2. Identify any correlation in the feature set and visualize (**PCA**, t-SNE, Pearson Correlation)
3. Build classifiers (**tree based**, gaussian, linear)
4. Identify important features (**feature importance**, feature extraction and/or engineering)
5. Predict on the test set (**weighted model averages**, soft or hard voting, BMA)

PCA Before Feature Importance

Why PCA?

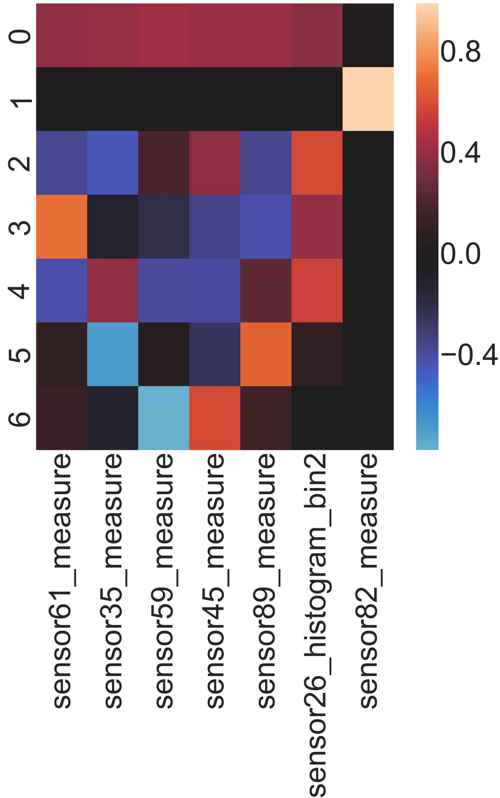
Good to use over t-SNE and Correlation methods in data that is incomplete or has missing data.

BEFORE

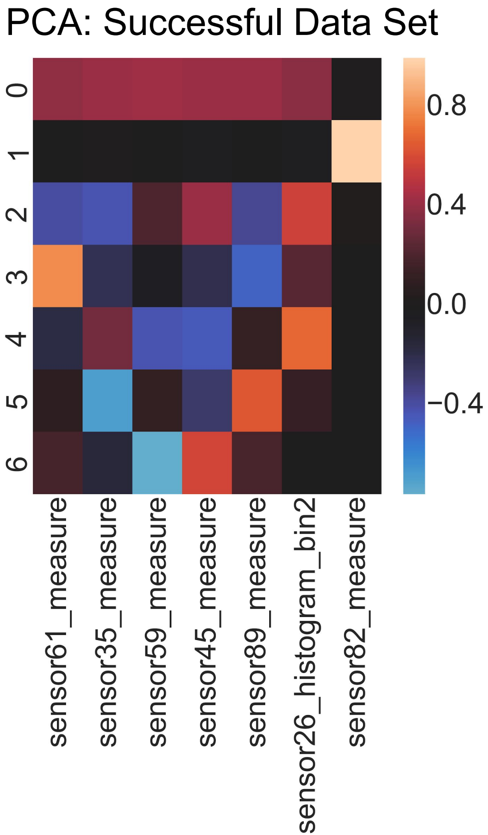


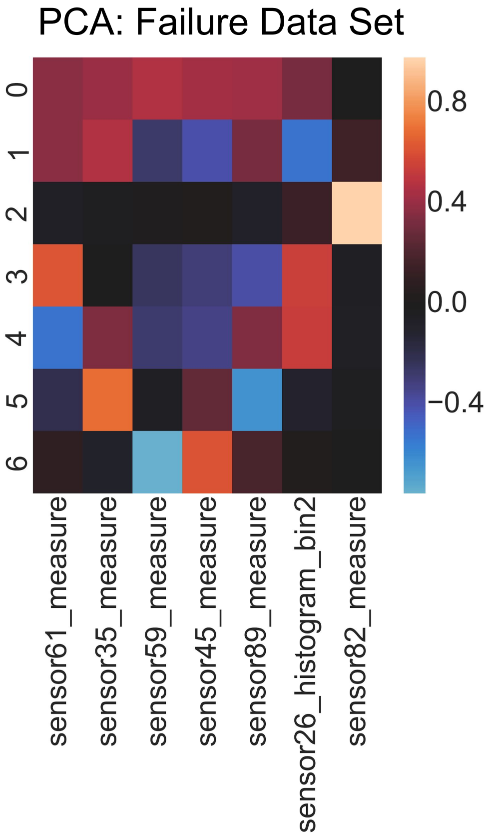
PCA: Total Data Set

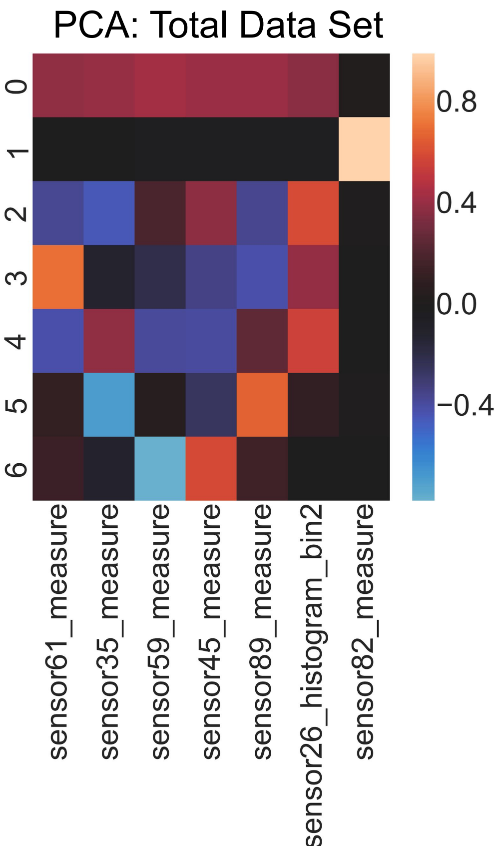
AFTER



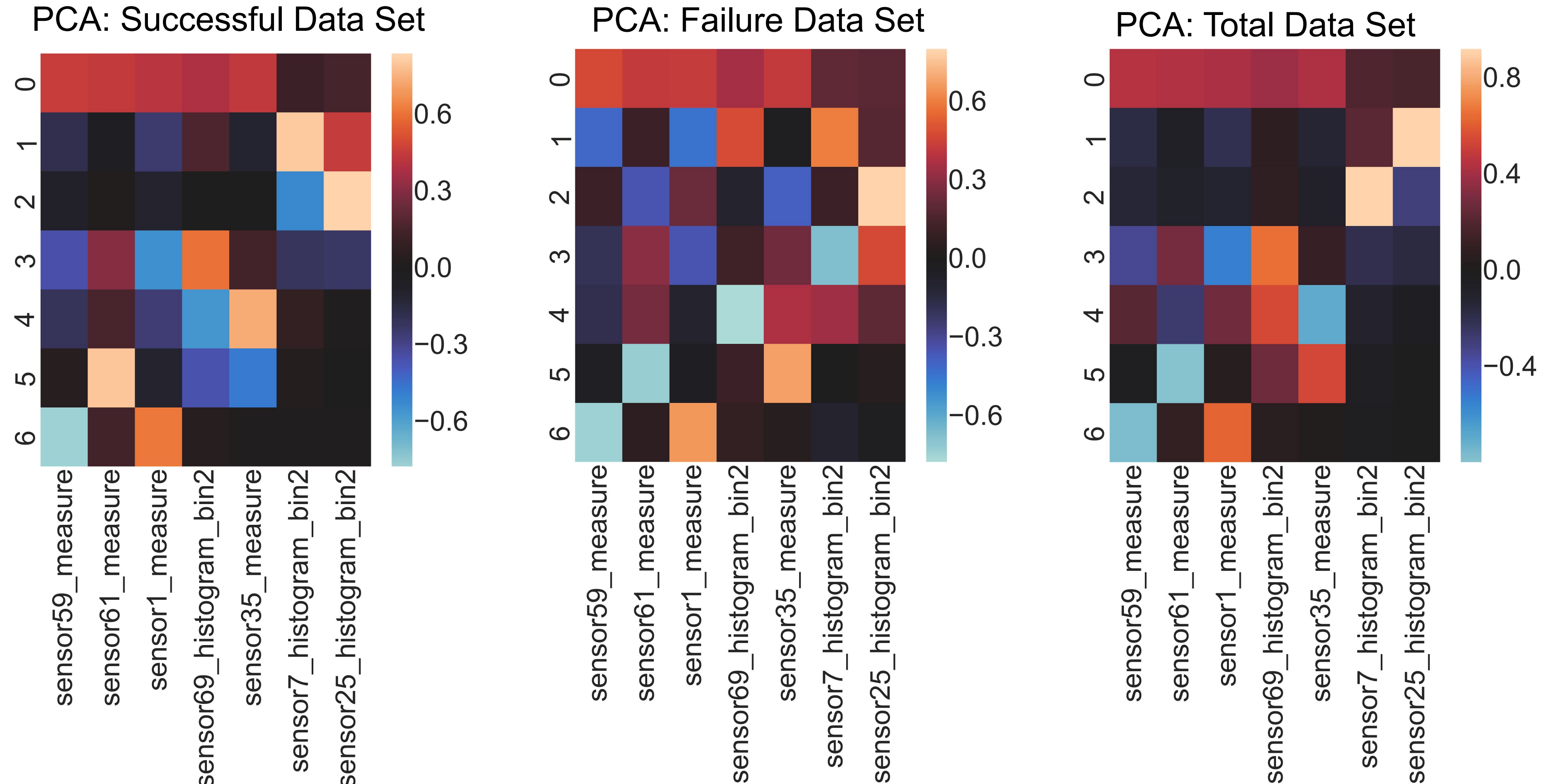
PCA: FI Average



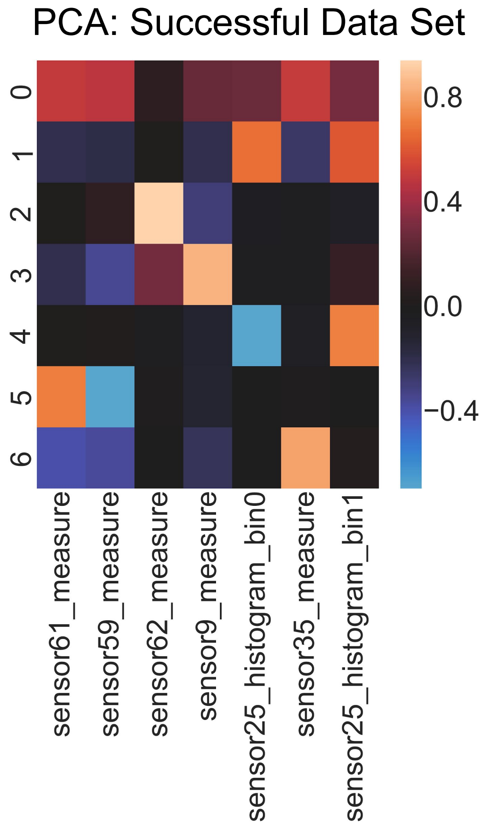


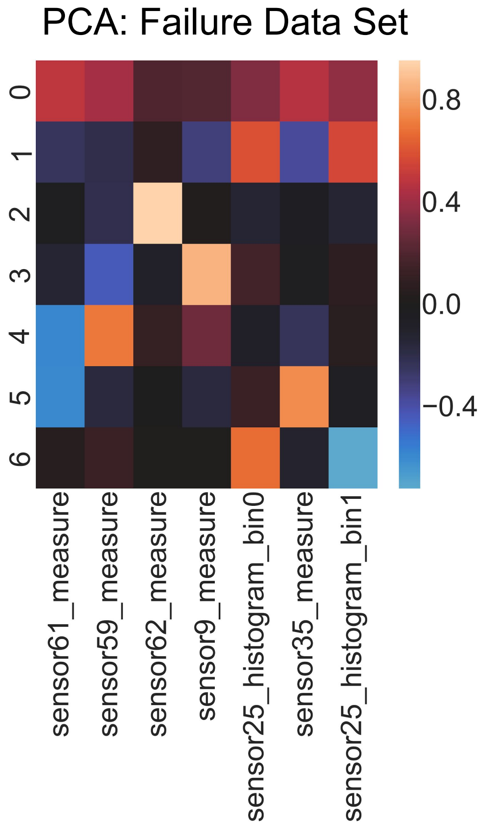


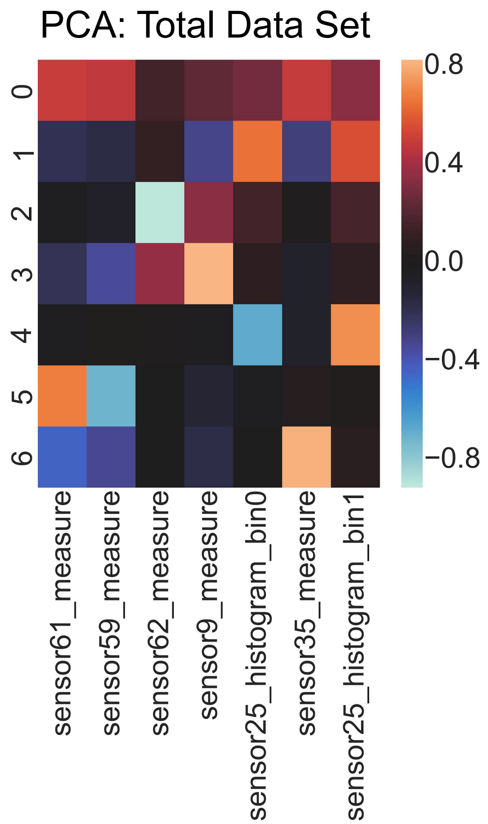
PCA: FI XGboost



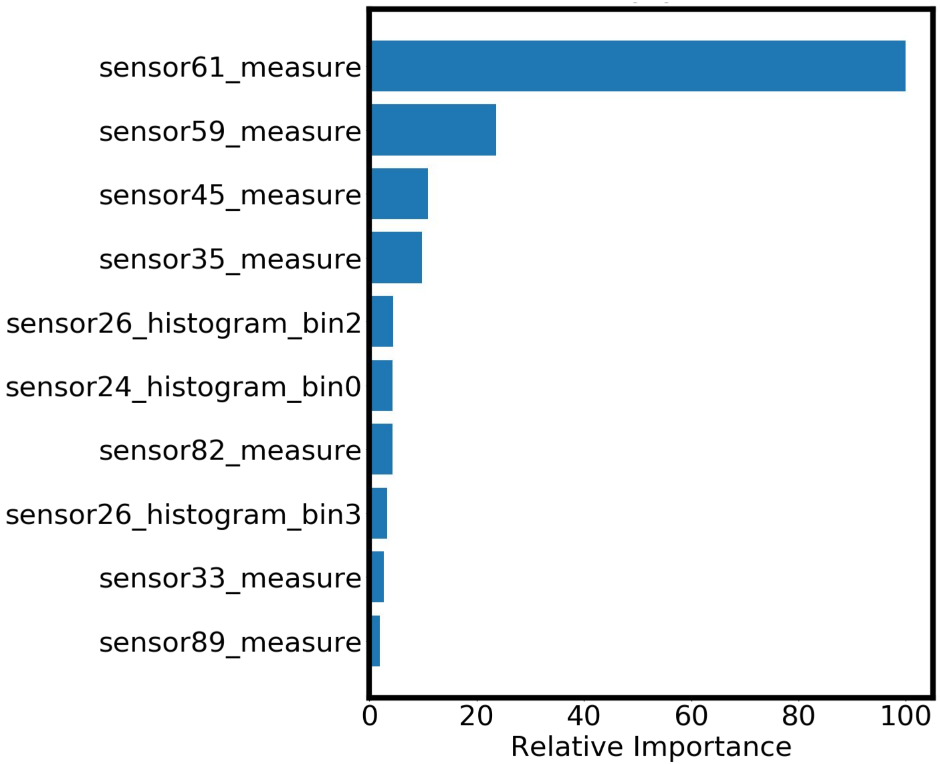
PCA: FI Gradient Boosting







Classification in High Dimension

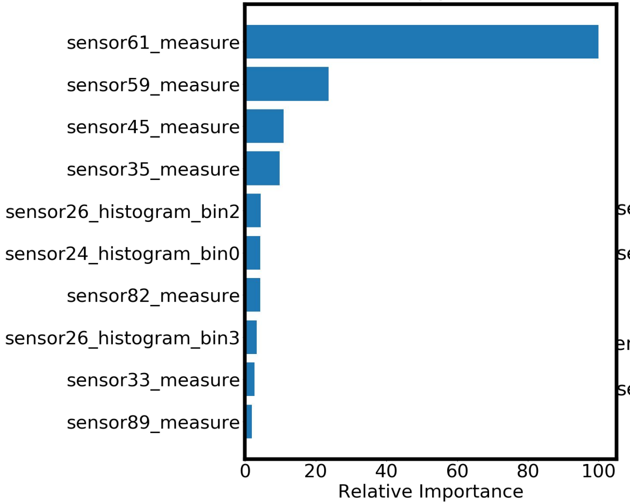


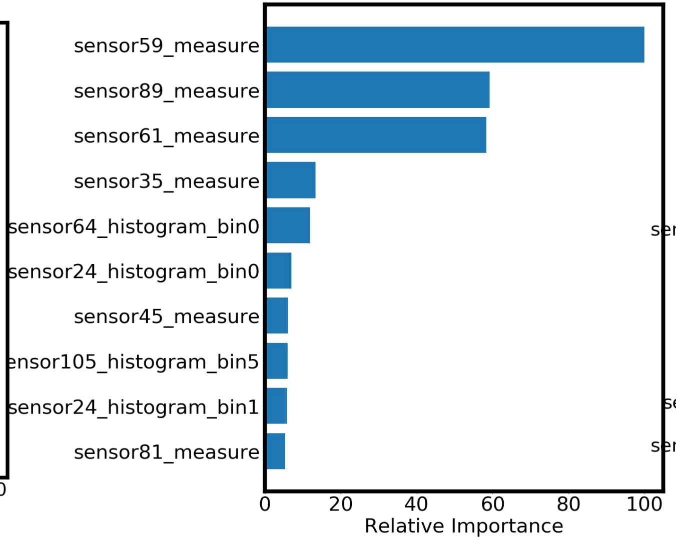
Why Tree Based?

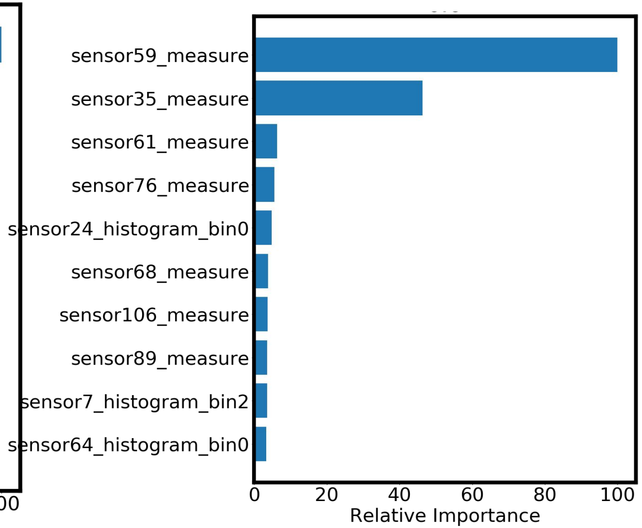
Tree based algorithms handle higher dimensional and sparse data better than most classifiers like Gaussian, naive, linear, etc

In anomaly detection, tree based classifiers, when tuned correctly, can detect anomaly with good accuracy. Especially tree based algorithms with a built in cost function like gradient boosting

Random Seed: Feature Importance



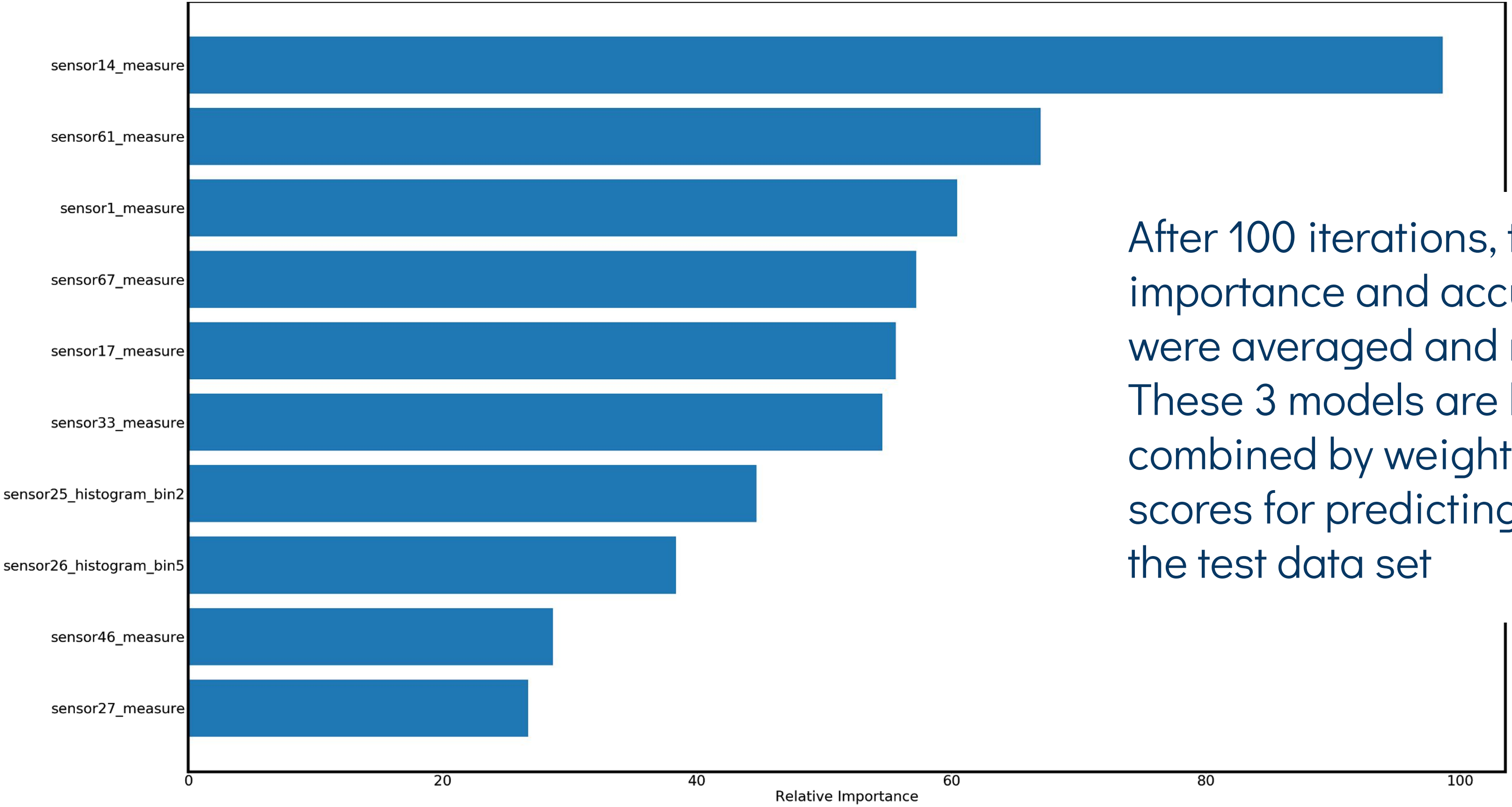




The data is randomly undersampled, so 1000 data points for target=0 and another 1000 data points for target=1 are used in a classifier.

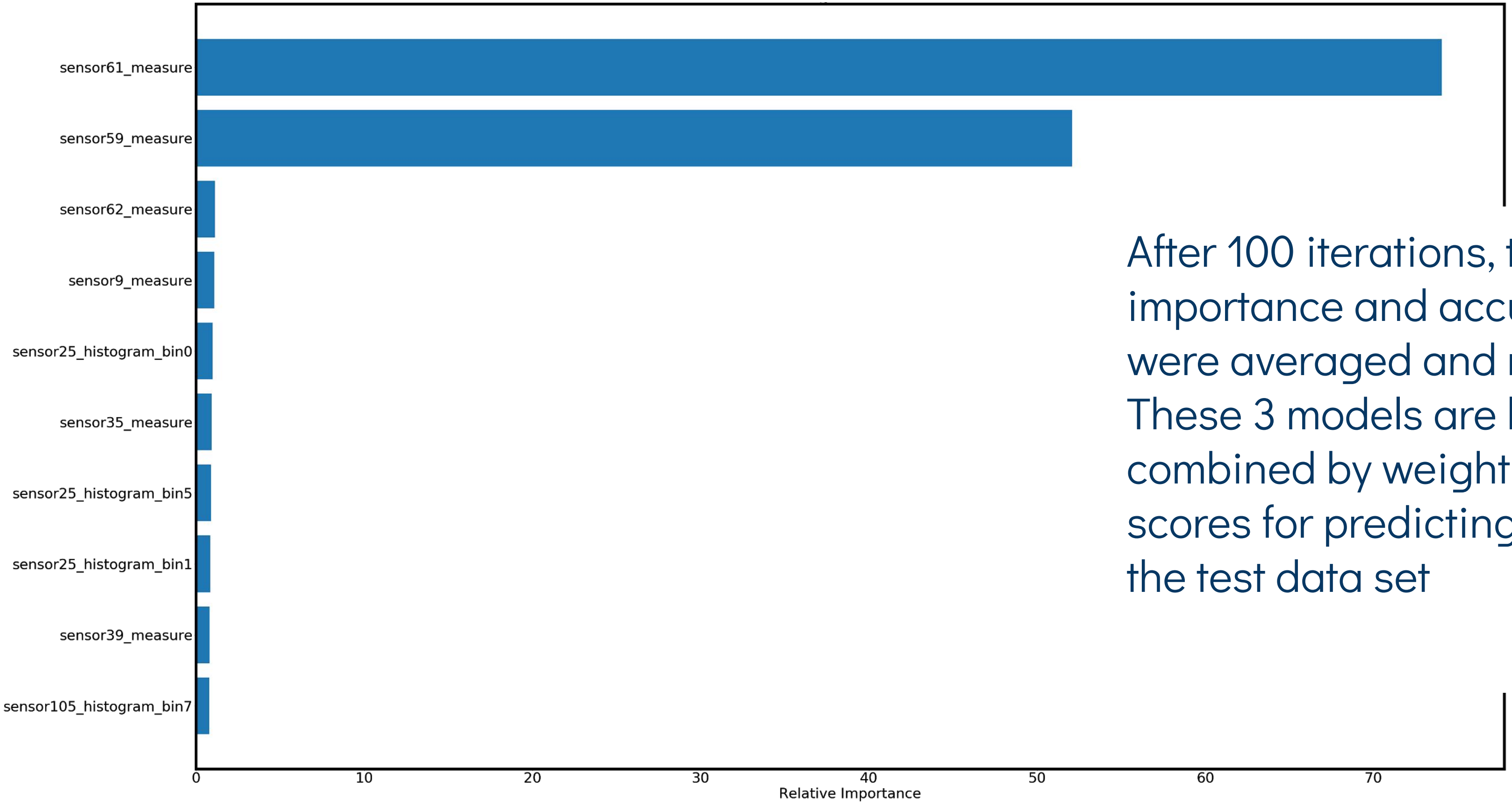
With each random seed, the important features change.

Feature Importance: Random Forest



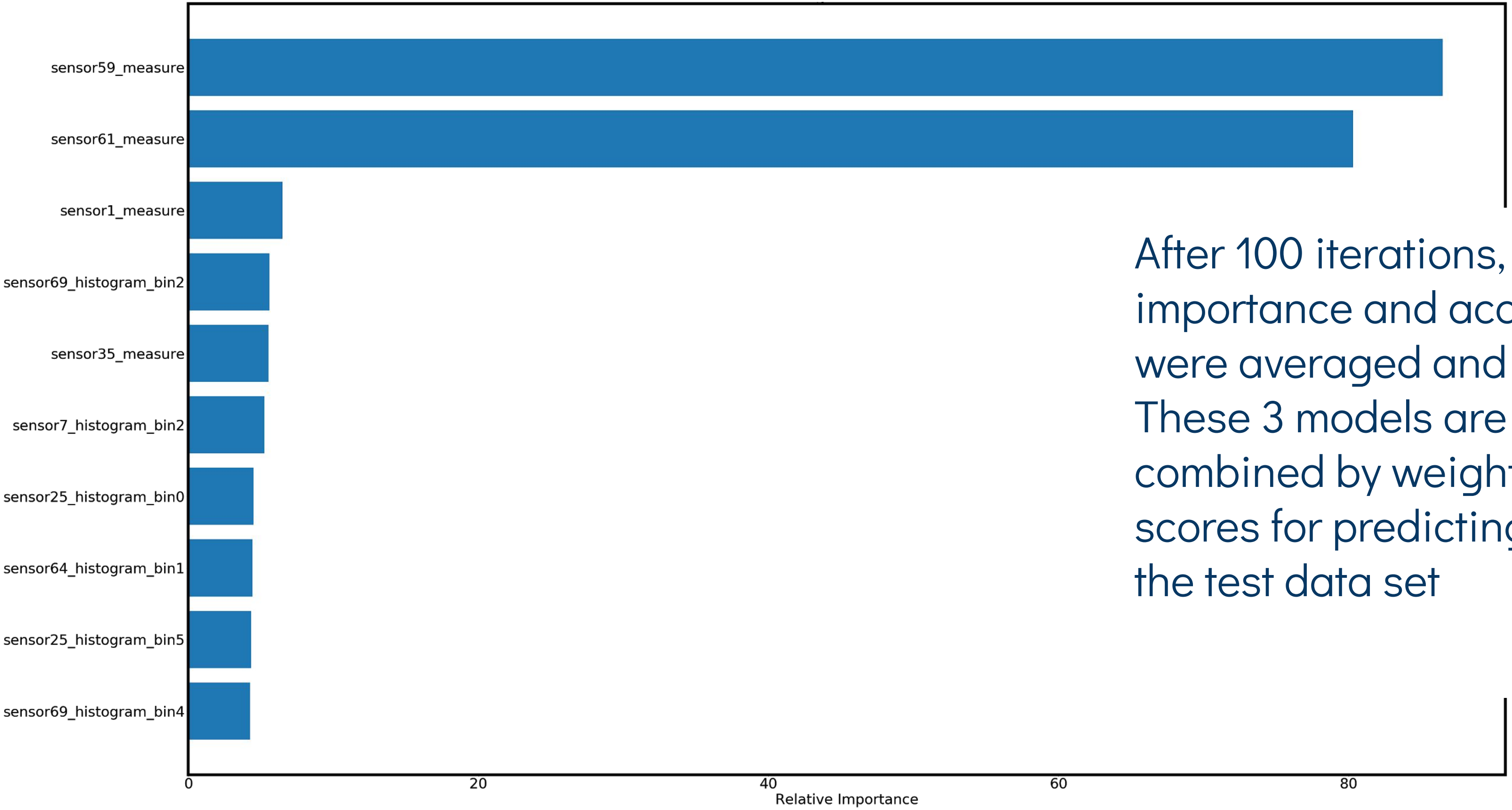
After 100 iterations, the feature importance and accuracy scores were averaged and reported. These 3 models are later combined by weighting by the scores for predicting targets on the test data set

Feature Importance: GBoosting



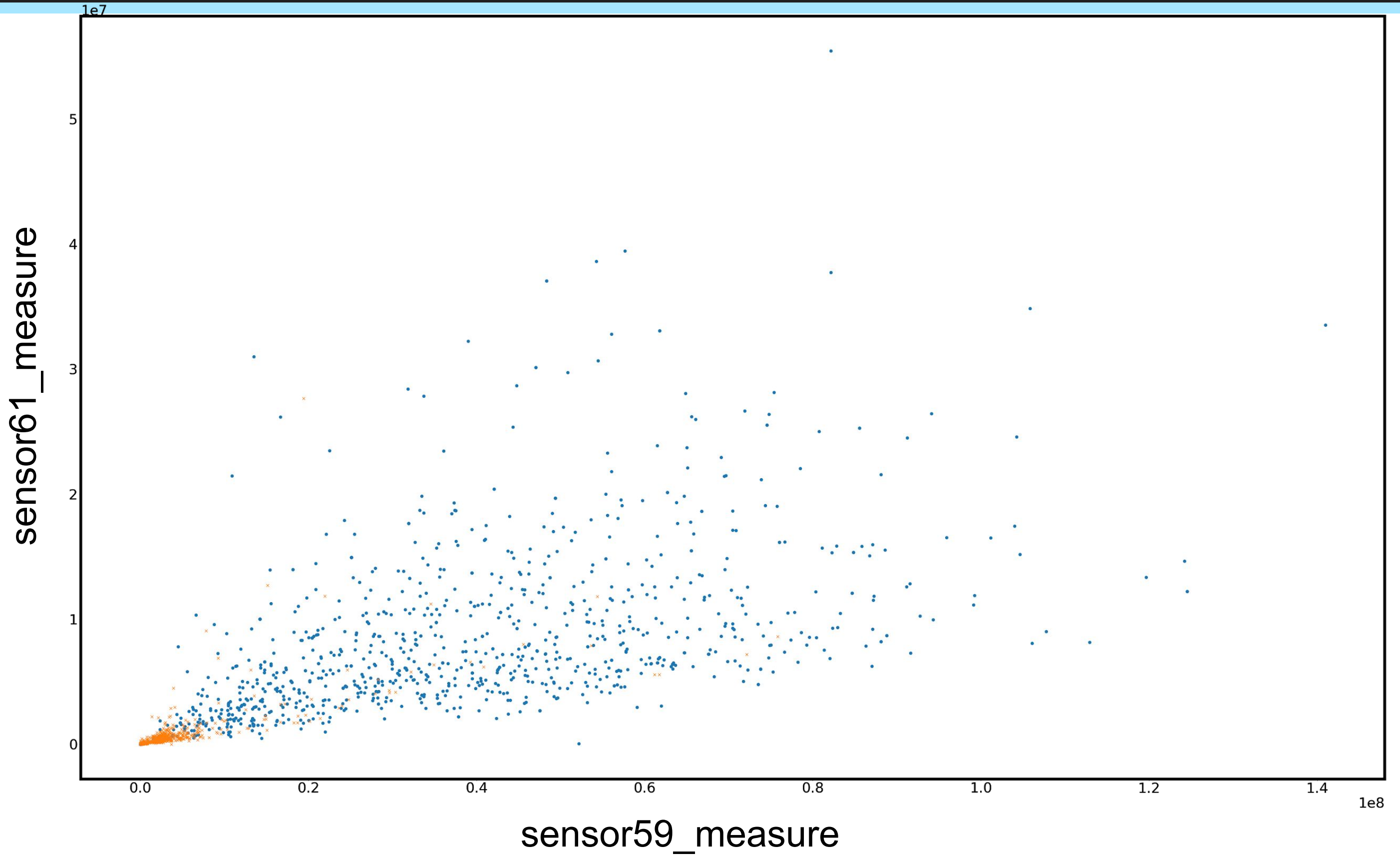
After 100 iterations, the feature importance and accuracy scores were averaged and reported. These 3 models are later combined by weighting by the scores for predicting targets on the test data set

Feature Importance: XGboosting



After 100 iterations, the feature importance and accuracy scores were averaged and reported. These 3 models are later combined by weighting by the scores for predicting targets on the test data set

Visualizing Two Naughty Features



The scatter plot shows the data does seem to be somewhat separable by the two most important features from GBoost and XGBoost classifiers.

ConocoPhillips: Conclusion

Using PCA and feature importance, relative features to detecting the target were identified.

By random undersampling, the classifiers were fit and weighted based on **accuracy scores**. The voting classifier doesn't give us full control over the weighting and BMA is unnecessary for models so similar, especially considering tree based classifiers aren't based on a normal distribution.