Boxing Matches

Nicholas Morris

Machine Learning

Wrangling

Imputation

Filling in missing values with random known observations.

Feature Engineering

Transformations

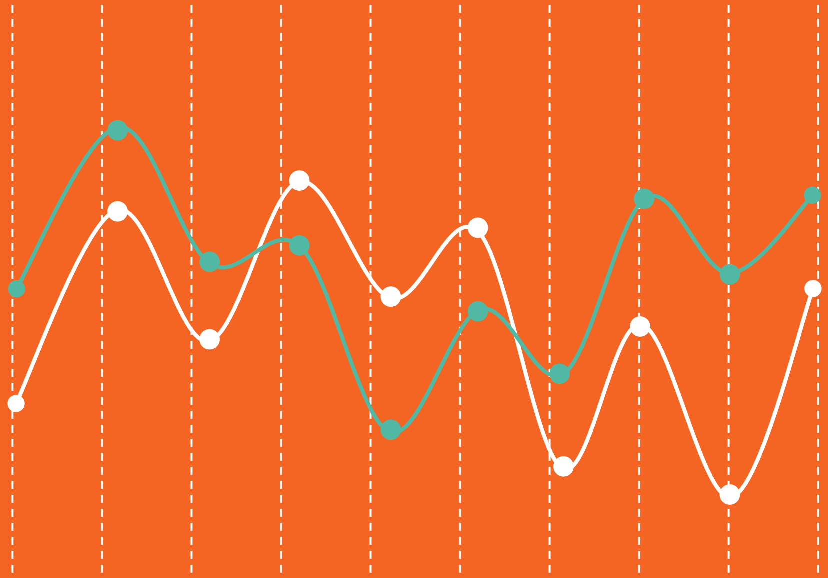
Attempting feature transformations such as Atwood Numbers, Binning, Reciprocals, and Interactions.

Modeling

Predictions

Training logistic regression, XGBoost, and deep learning neural network models. Evaluating performance. Computing feature drift to signal retraining.

Wrangling



Dataset

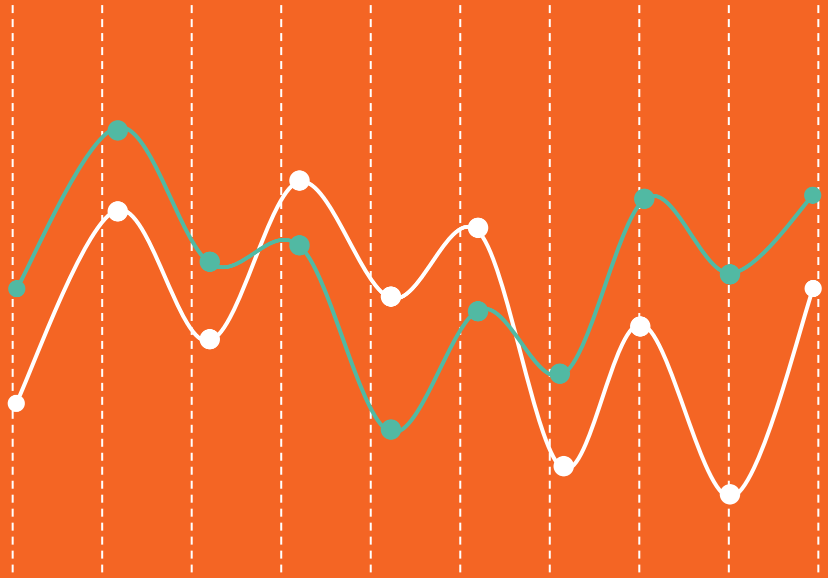
Below is the first two boxing matches in the data. There are 387,427 total matches. The target we are predicting is result. [Link to the dataset]

age_A	age_B	height_A	height_B	reach_A	reach_B	stance_A	stance_B	weight_A	weight_B	...	kos_A	kos_B	result	decision	judge1_A	judge1_B	judge2_A	judge2_B	judge3_A	judge3_B
35.01	27.0	179.0	175.0	178.0	179.0	orthodox	orthodox	160.0	160.0	...	33	34.0	draw	SD	110.0	118.0	115.0	113.0	114.0	114.0
26.01	31.0	175.0	185.0	179.0	185.0	orthodox	orthodox	164.0	164.0	...	34	32.0	win_A	UD	120.0	108.0	120.0	108.0	120.0	108.0

Missing Values

There were over 30 thousand rows of missing values. Imputation by k-Nearest Neighbors ran far too slow. Instead of filling in missing values with the average for numeric data and the mode for categorical data, the data was shuffled and then the last known value was used to do random imputation.

Feature Engineering



Binary Data

stance_A, stance_B, and decision were converted to binary data points.

Percent Differences

A percent difference is a calculation that shows the relative change between two variables. The formula for two variables x and y is: $x / y - 1$

This calculation was done on all pairs of features for fighter A and B; but did not improve model performance, so, it was left out of the final model.

Atwood Numbers

An Atwood Number is a calculation that shows the relative change between two variables. The formula for two variables x and y is:
$(x - y) / (x + y)$

This calculation was done on all pairs of non-binary variables; but did not improve model performance, so, it was left out of the final model.

Binning

Binning is when a non-binary variable is grouped into histogram bins, and represented as binary variables.

Binning did not improve model performance, so, it was left out of the final model.

Reciprocals

A reciprocal is when a non-binary variable x is calculated as 1 / x.

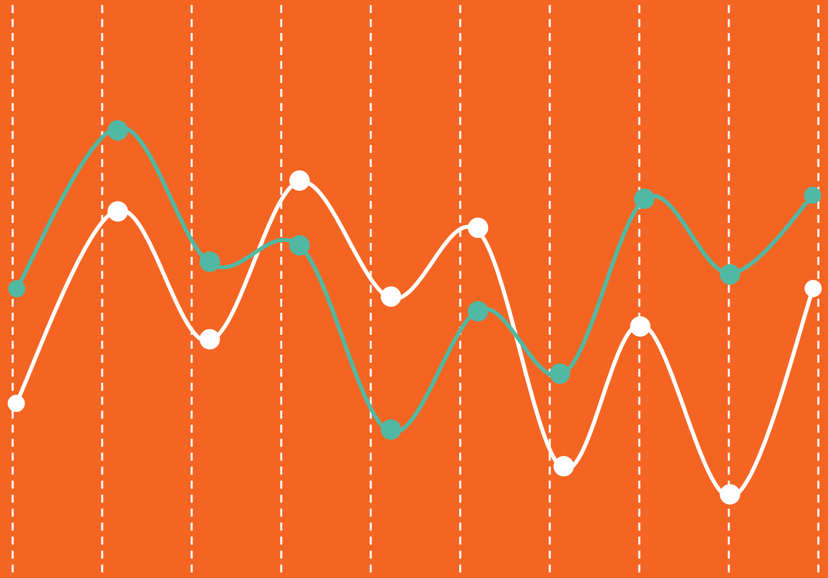
Reciprocals did not improve model performance, so, it was left out of the final model.

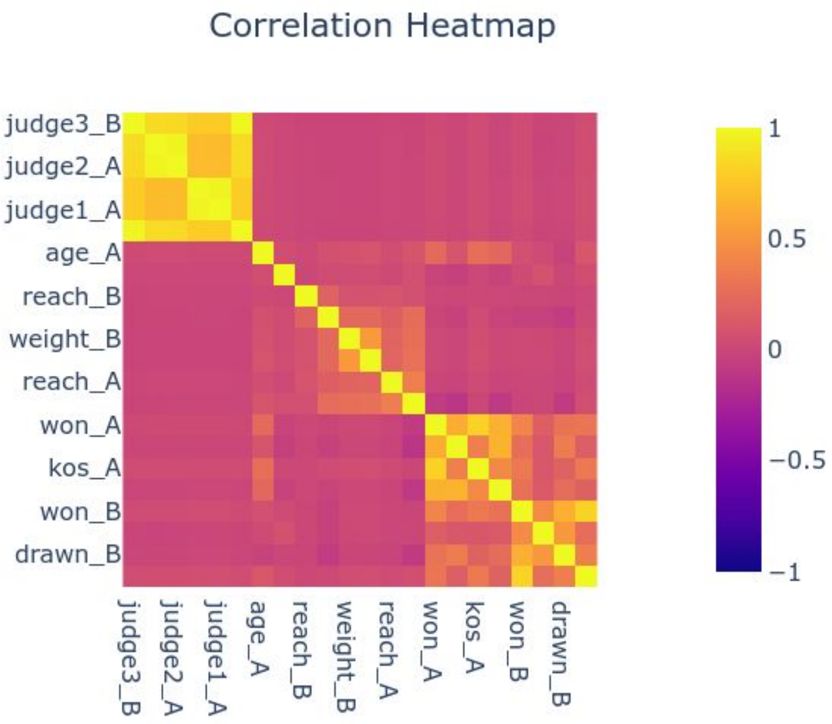
Interactions

An interaction is when two variables x and y are calculated as x * y. Reciprocals were fed into this calculation to generate x / y as well.

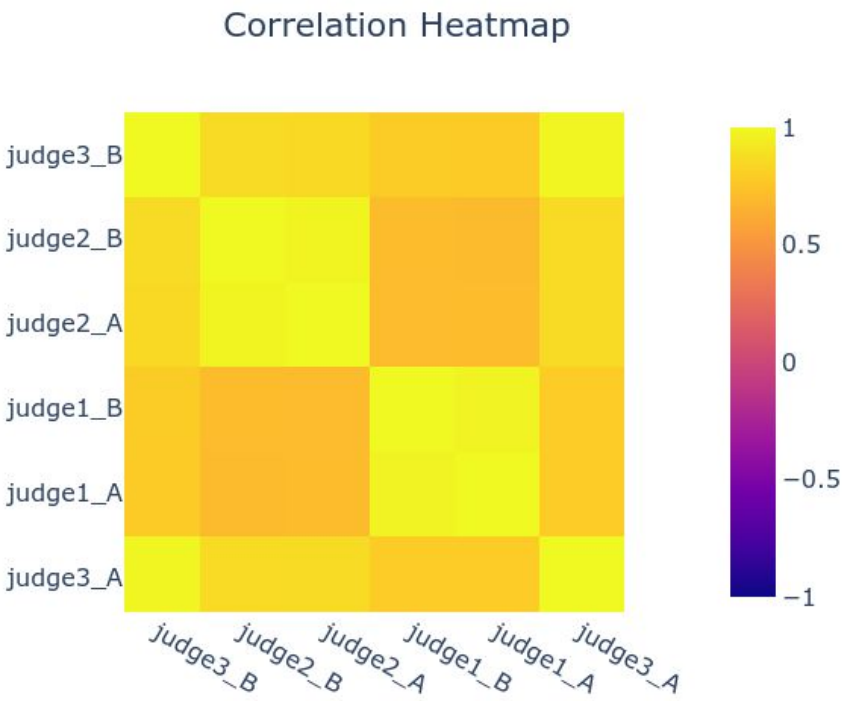
Interactions did not improve model performance, so, it was left out of the final model.

Data Exploration

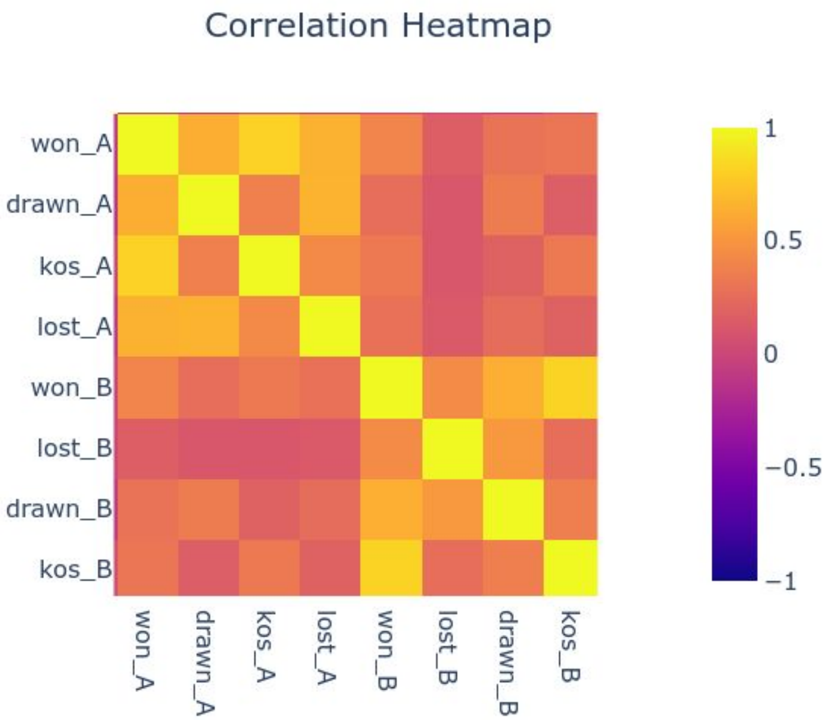




There are three zones of correlation with the top-left one showing strong correlation, the bottom-right showing some correlation, and the center showing weaker correlation.

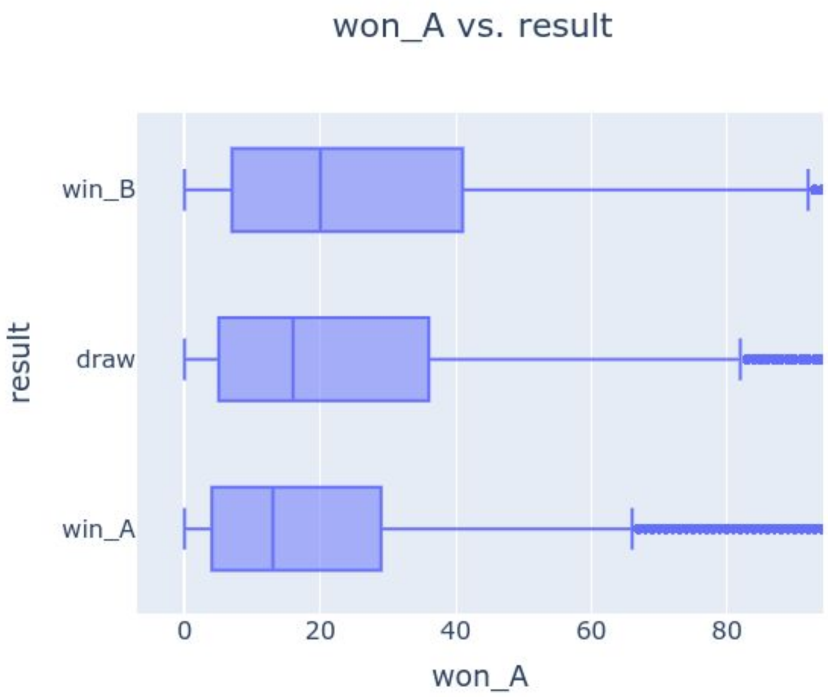


The strongest zone shows that all judges are in agreement with one another when making a decision. This means that when one judge places a higher or lower score for a fighter, the other judges also place a similar score.

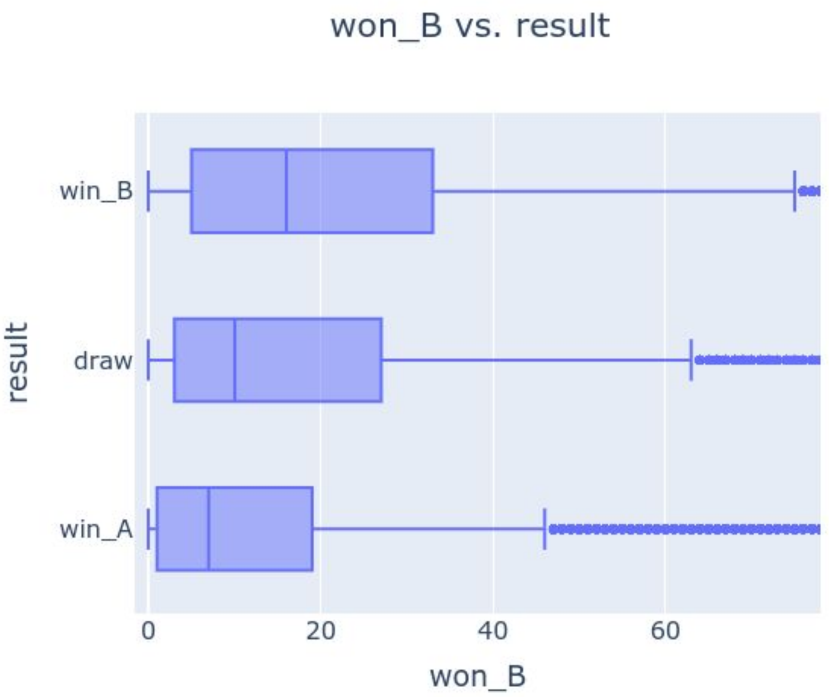


The bottom-right zone has two smaller zones within it. These two sub-zones show agreement between the wins, losses, draws, and knockouts for each fighter.

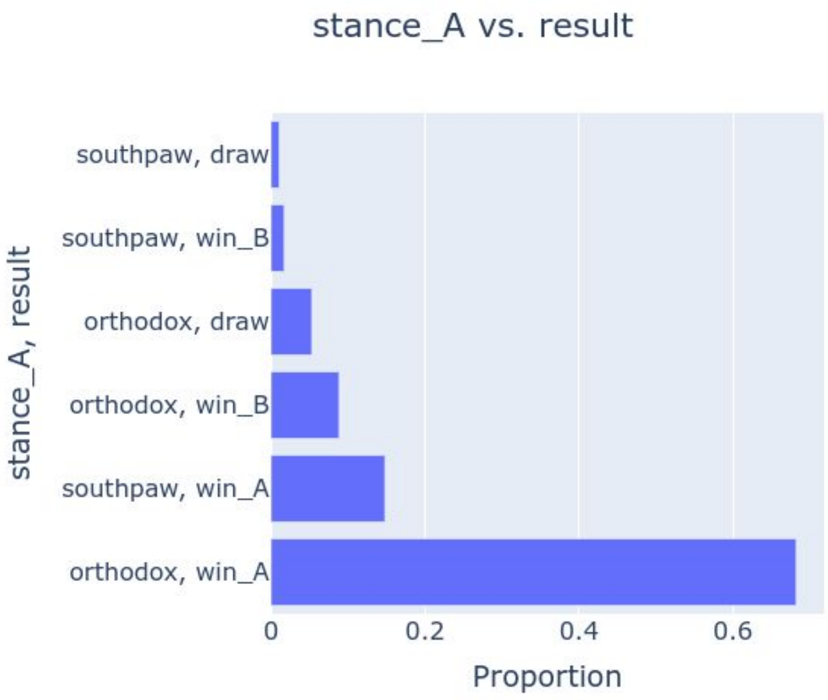
When a fighter has more wins they also have more knockouts. When a fighter has more wins there is a tendency to also have some more losses and draws. But when there's more knockouts there isn't necessarily more draws.



When fighter A has more wins on their record there is a slight tendency for fighter B to win. The median wins for fighter A is 20 when fighter B wins the match. And the median wins for fighter A is 13 when fighter A wins the match.

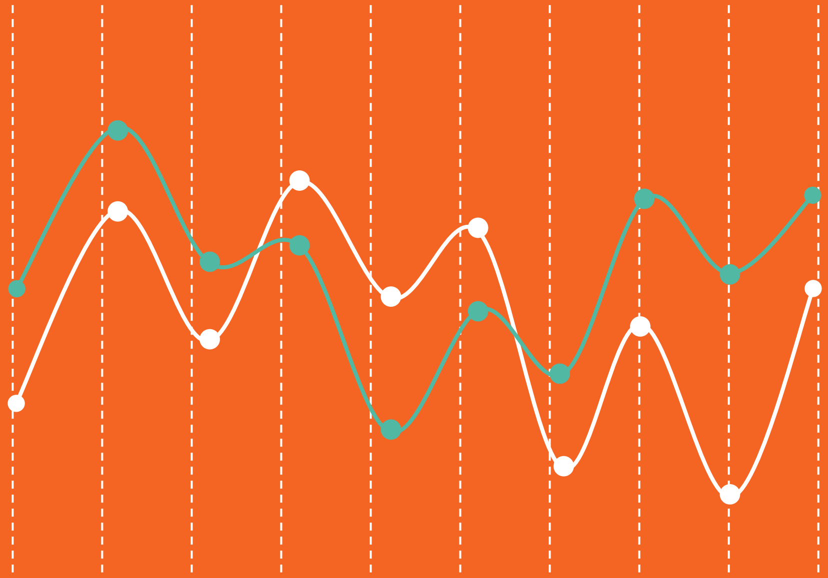


When fighter B has more wins on their record there is a slight tendency for fighter B to win. The median wins for fighter B is 16 when fighter B wins the match. And the median wins for fighter B is 7 when fighter A wins the match.



Most of the wins come from fighter A with an orthodox stance. The second most wins come from fighter A with a southpaw stance. Fighter A wins 83% of the matches in the dataset.

Modeling



Model Parameters

Logistic Regression

Library: scikit-learn
Penalty: L1
Number Of Alphas: 16
Cross Validation Folds: 3
Tolerance: 1e-4
Max Iterations: 100

XGBoost

Library: xgboost
Boosting Rounds: 100
Learning Rate:
  0.001, 0.01, 0.1
Max Depth:
  5, 7, 10, 14, 18
Min Child Weight: 1
Column Sampling: 0.8
Row Sampling: 0.8
Cross Validation Folds: 3

Neural Network

Library: Tensorflow
Epochs: 500
Learning Rate:
  0.0001, 0.001, 0.01
Batch Size: 16
Layers: 10
Nodes Per Layer:
  32, 64, 128, 256, 512
Solver: Adam
Cross Validation Folds: 3

Model Comparison

Logistic Regression

Accuracy: 0.65
F1: 0.42
In Control: 99.43%

Model Indicators:
1. lost_B
2. lost_A
3. won_B
4. judge2_A
5. judge2_B

XGBoost

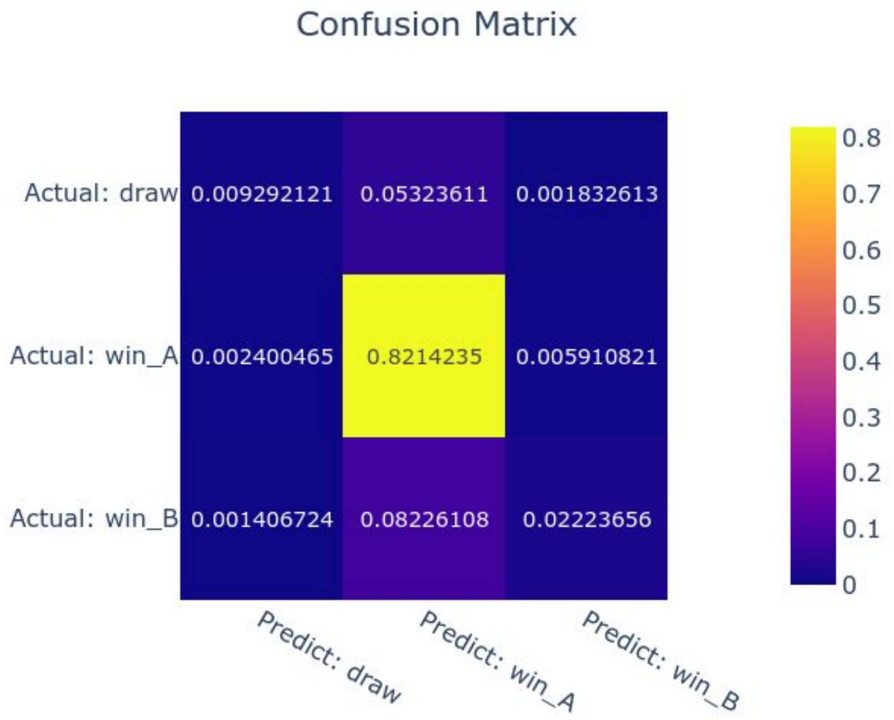
Accuracy: 0.85
F1: 0.50
In Control: 99.1%

Model Indicators:
1. decision_TD
2. decision_PTS
3. decision_NWS
4. decision_MD
5. decision_SD

Neural Network

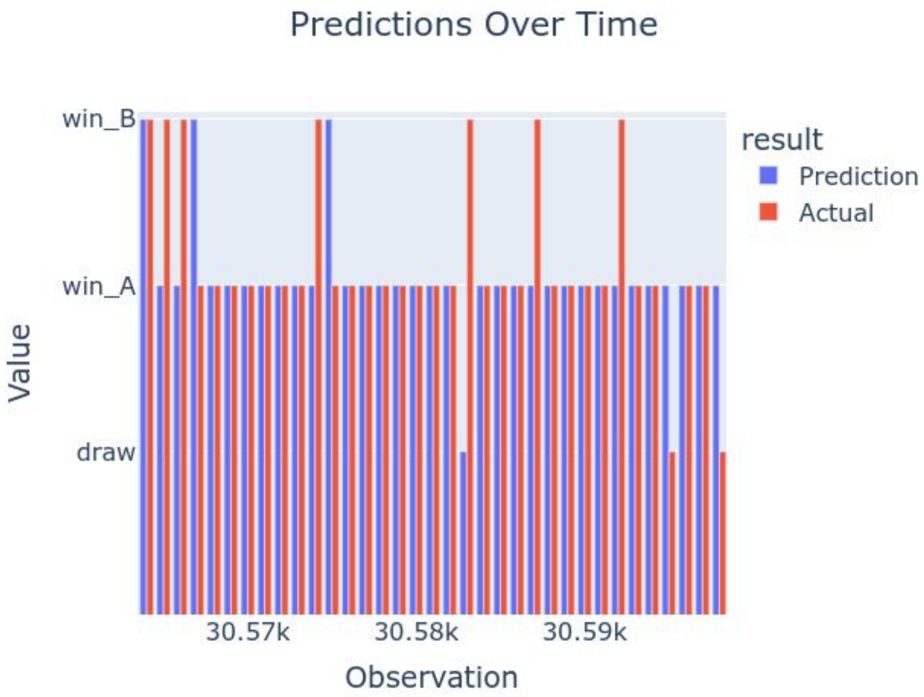
Accuracy: DNF
F1: DNF
In Control: DNF

Model Indicators:
DNF



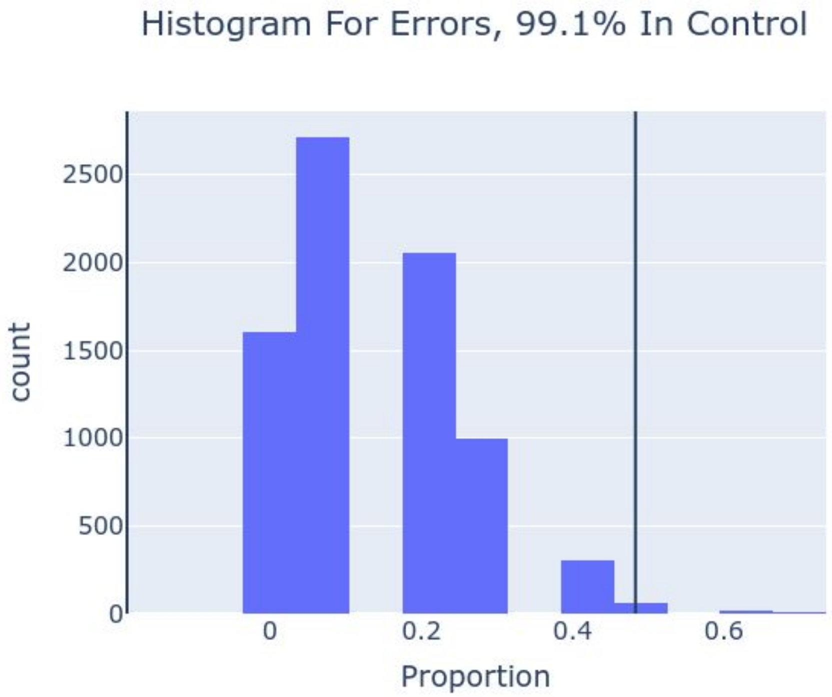
These predictions come from the XGBoost model. These predictions are done on 20% of the data that the model did not see during training. Only 14.8% of the predictions are wrong, and 85.2% of the predictions are correct. There's a slightly stronger tendency to predict draws and fighter B wins as fighter A wins.

Fighter A wins 83% of the time so the model isn't just predicting that fighter A always wins.



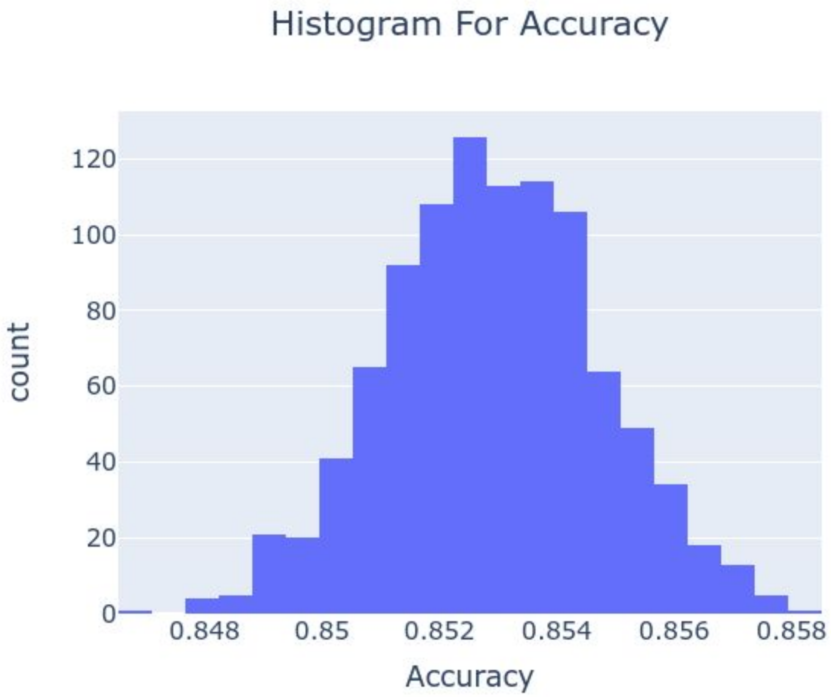
This is a snapshot of predictions.

We can see that the model correctly predicts fighter B winning 3 times. The model correctly predicts fighter A winning most of the time. The model incorrectly predicts fighter A winning when fighter B won 3 times. The model incorrectly predicts a draw when fighter B won 1 time. And the model incorrectly predicts fighter A winning when there was a draw 2 times.



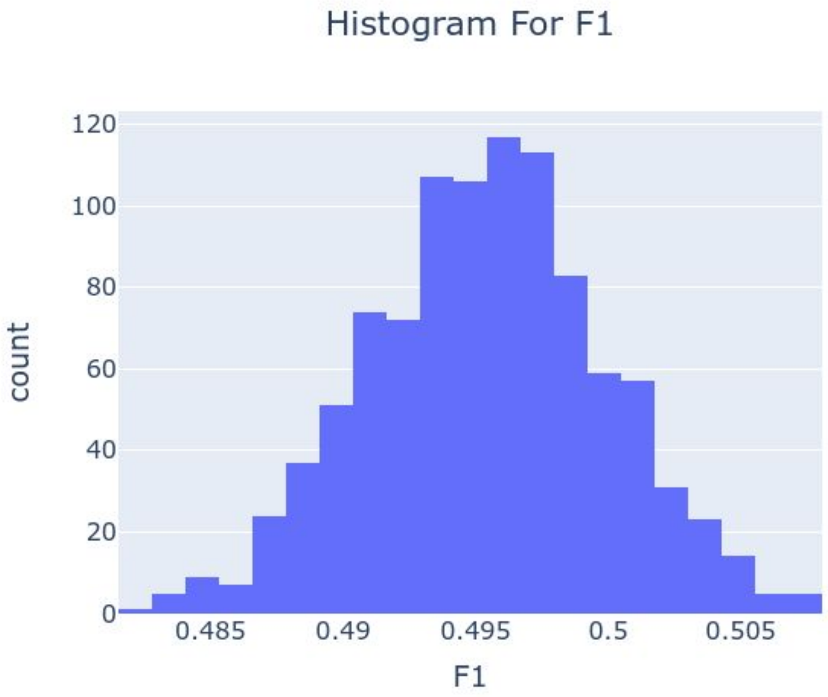
The errors are the fraction of 10 predictions that were wrong.

The errors are most likely to be 10%. Control limits were computed on the errors and we can see that the prediction error is mostly under control. There is a skew to the right, which isn't good. We would like a bell shape for the errors.



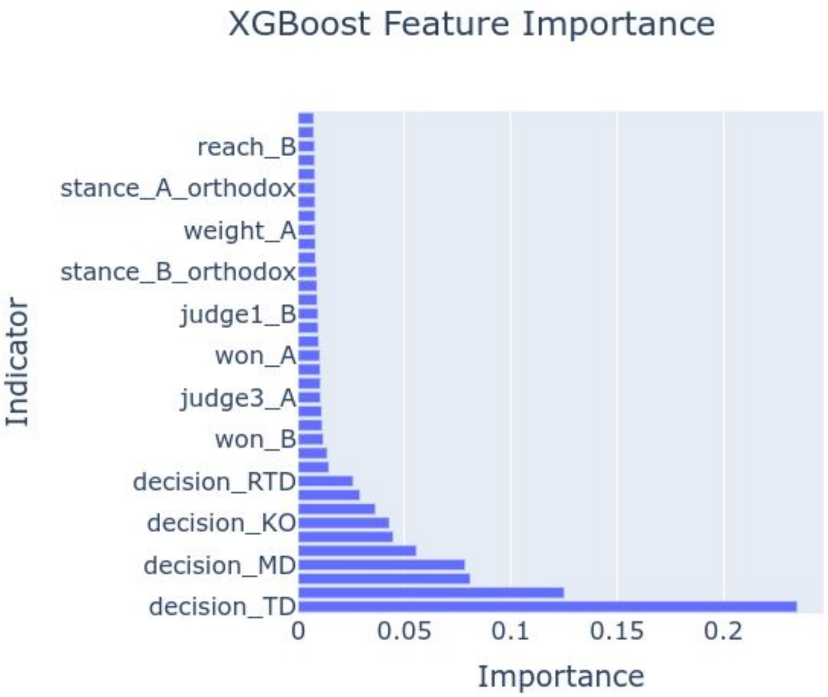
The prediction error was resampled 1000 times at a 50% sampling rate with replacement. Then Accuracy was computed on each sample to get a distribution.

Accuracy has a tight range between 0.848 and 0.858, which is good. Accuracy has a bell shape, which is good.

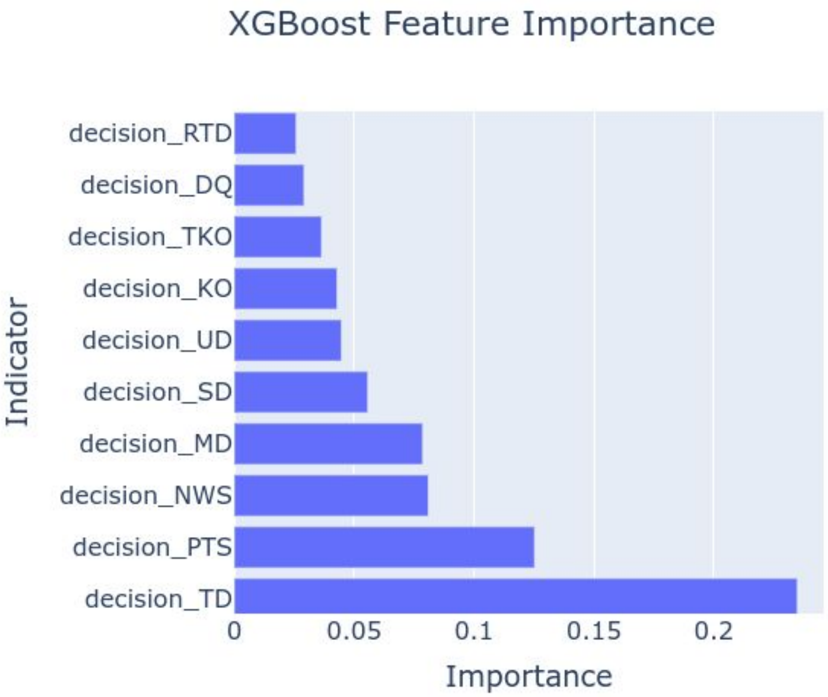


The prediction error was resampled as previously mentioned to get a distribution for F1. F1 is a combination of Precision and Recall. Precision tells us how well the model doesn't label a match result as another one. Recall tells us how well the model labels all match results.

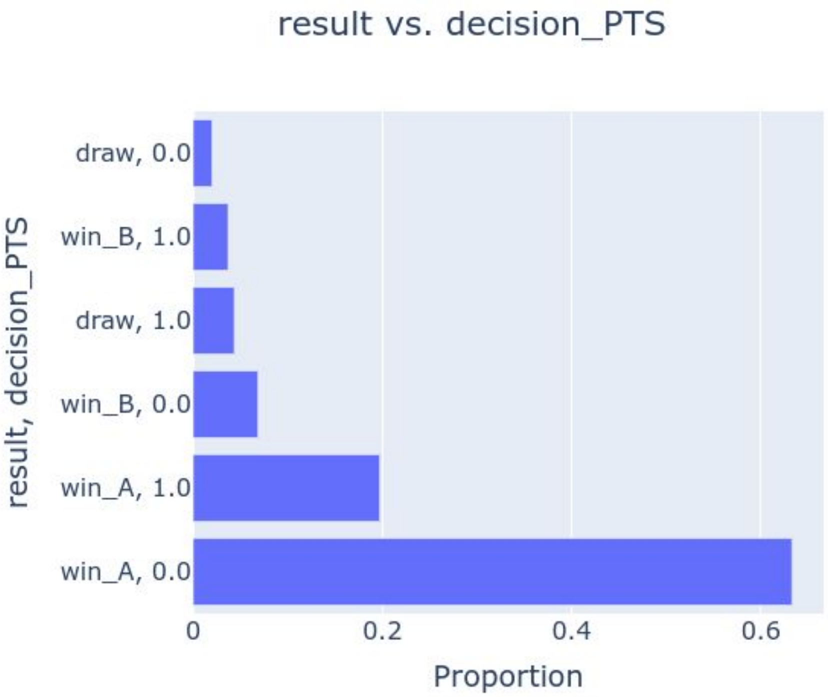
F1 has a tight range between 0.485 and 0.505, which is good. F1 has a bell shape, which is good; but the score is low (1 is the best, 0 is the worst)



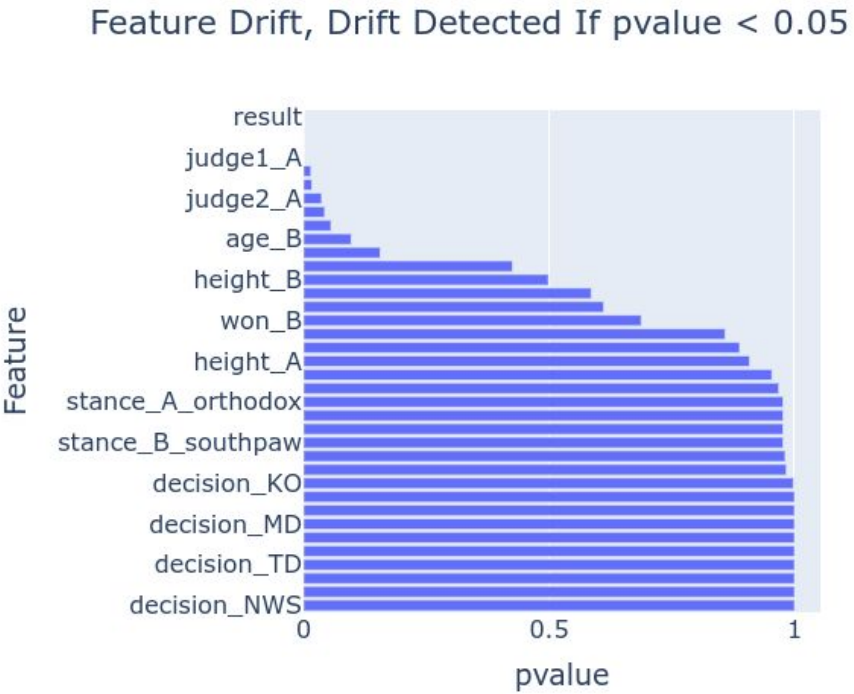
The feature importance is heavily weighted to a handful of features with the remaining features showing minimal support.



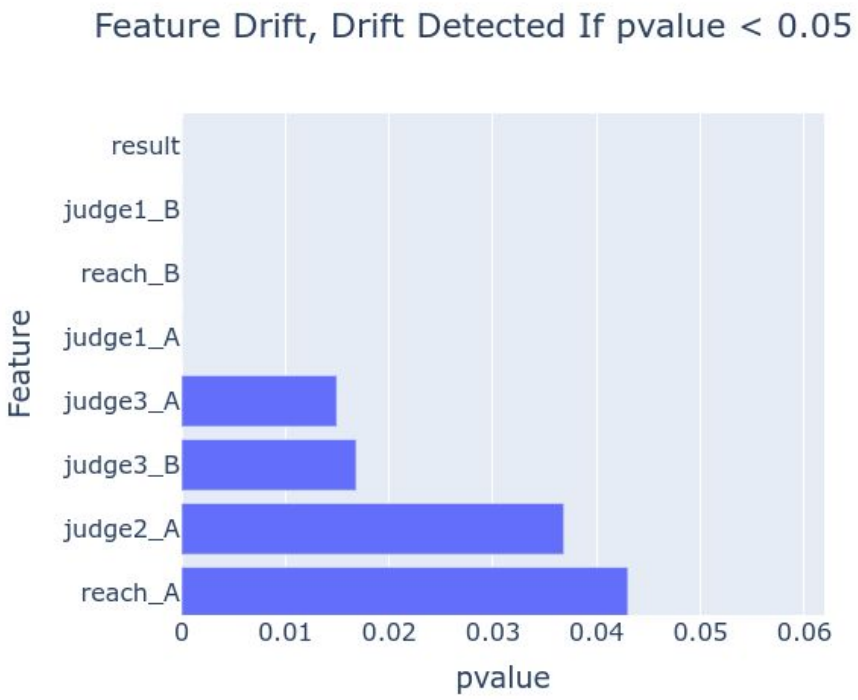
These are the top ten indicators of match result. They are all related to the match decision. The most important decision is a Technical Decision, which can result from fighters headbutting. The second most important decision is one made on Points.



This graph shows that 20% of fighter A wins come from a decision on points.



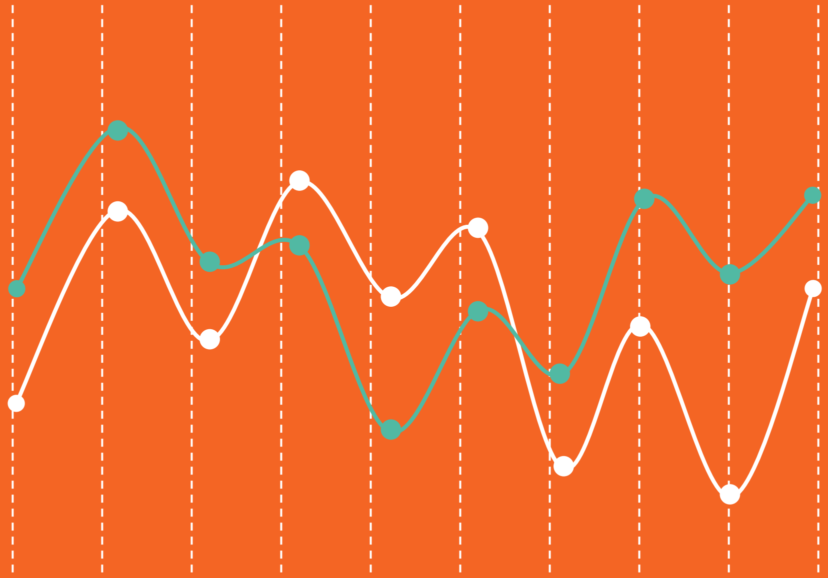
A Kolmogorov-Smirnov test was performed for each column in the data to see if the distribution of the testing data is the same as the training data. If the testing data does not share the same distribution as the training data, then there is a drift, which signals for model retraining. Most of the columns do not experience a drift, which is good.

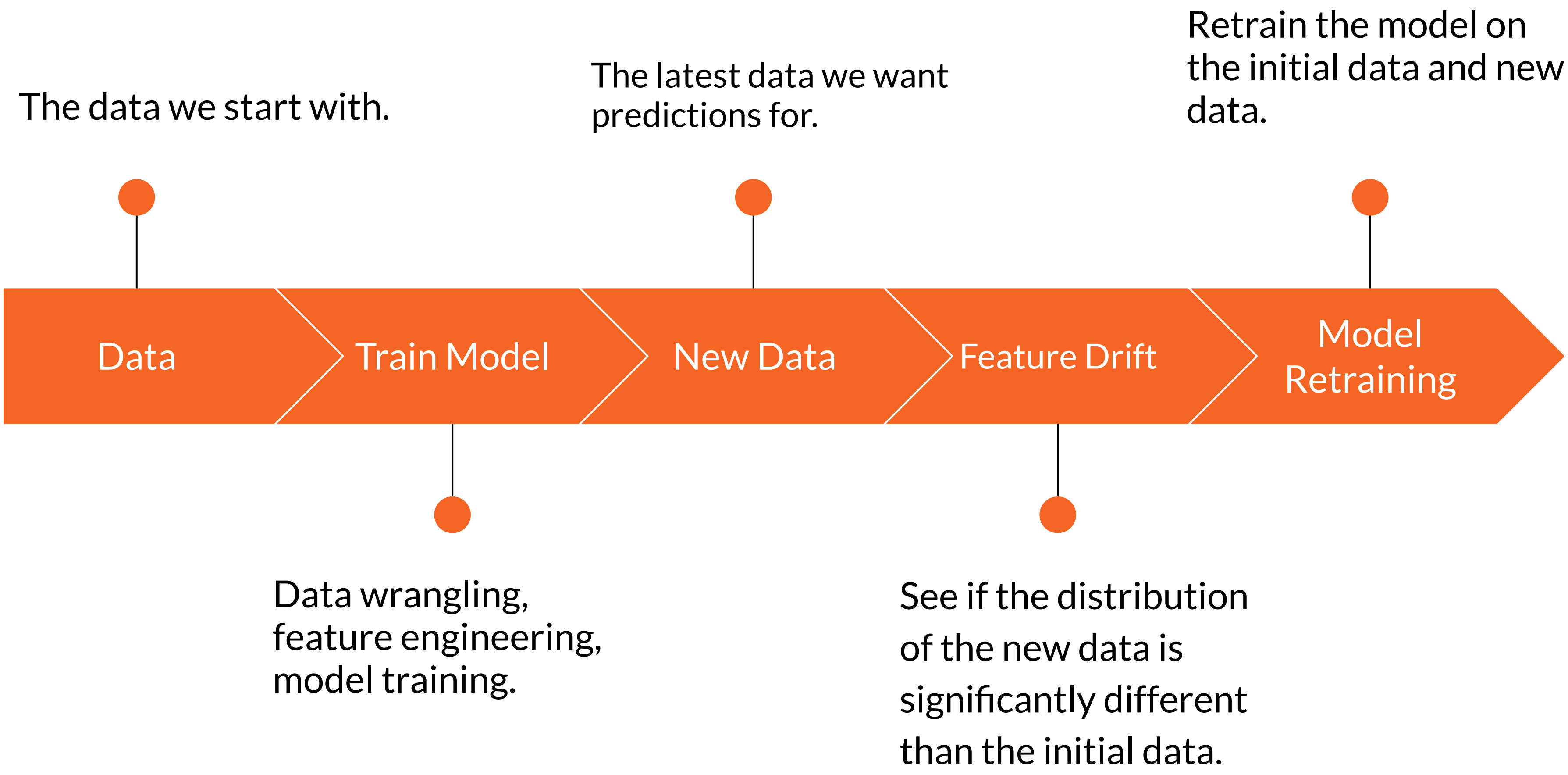


8 features are experiencing a drift. These are match result, the reach of the fighters, and the judges' scores.

The model was retrained to include the test data.

Deployment




The data we start with.
The latest data we want predictions for.
Retrain the model on the initial data and new data.
Data
Train Model
New Data
Feature Drift
Model Retraining
Data wrangling, feature engineering, model training.
See if the distribution of the new data is significantly different than the initial data.

Thank You