Wells Fargo
Job Simulation

How to be more relevant?

A classification analysis on search engine optimization

Frank Xu
Duke University

Question

Search engine queries are requests for information from users.

A list of results will be returned for a search query, expecting to be relevant.

Our task is to optimize the relevance: How to be more relevant?

Outline

- Part 1: Exploratory Data Analysis & Preprocessing
  - Data Overview
  - Data Abnormality
  - Data Preprocessing
- Part 2: Models & Comparison
  - Model Metrics
  - Model selection
  - Model Tuning
  - Model Comparison
- Part 3: Feature Interpretation & Recommendation
  - Feature Importance
  - Feature Interpretation
  - Recommendation

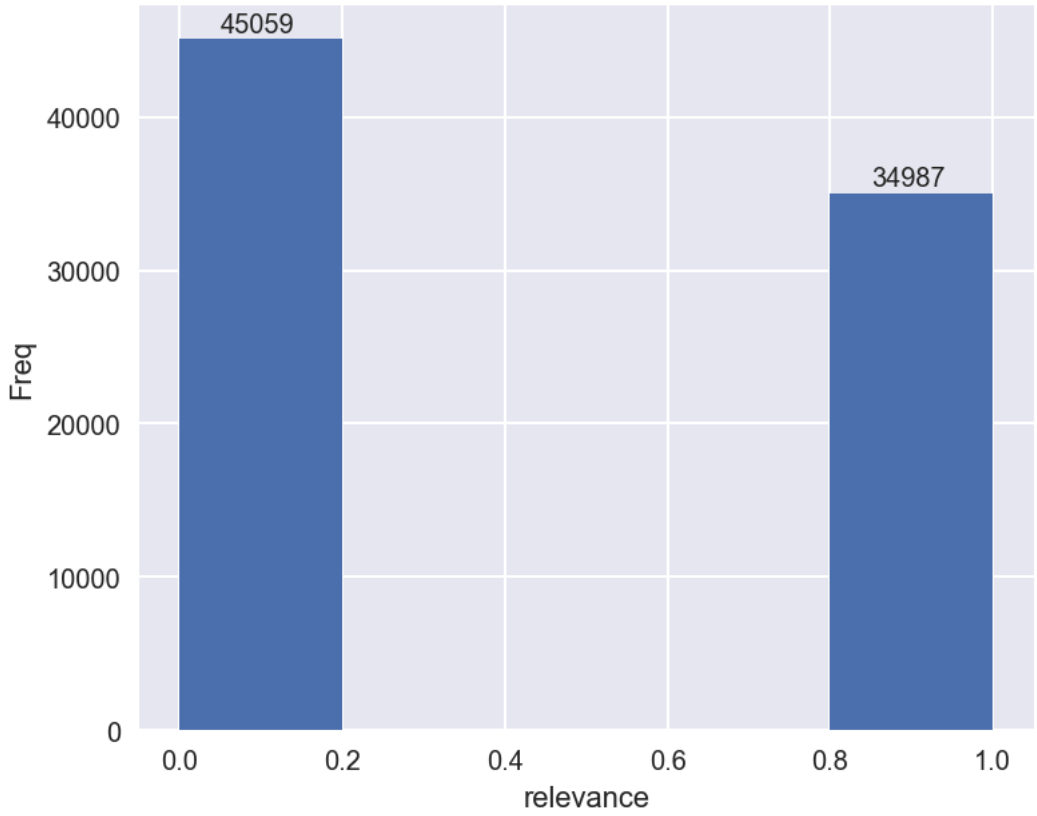
Part 1: Exploratory Data Analysis & Preprocessing

Data Overview

- Data has 80046 Observations, 12 features and 1 response variable.
- Among the 12 features, there are:
  - 2 ID-related features: Query ID & URL ID
  - Query Length & Homepage Indicator
  - 8 unlabeled features: sig1 ~ sig8
- Response variable is binary, with 1 indicates relevant and 0 the opposite.
- All variables are numeric.

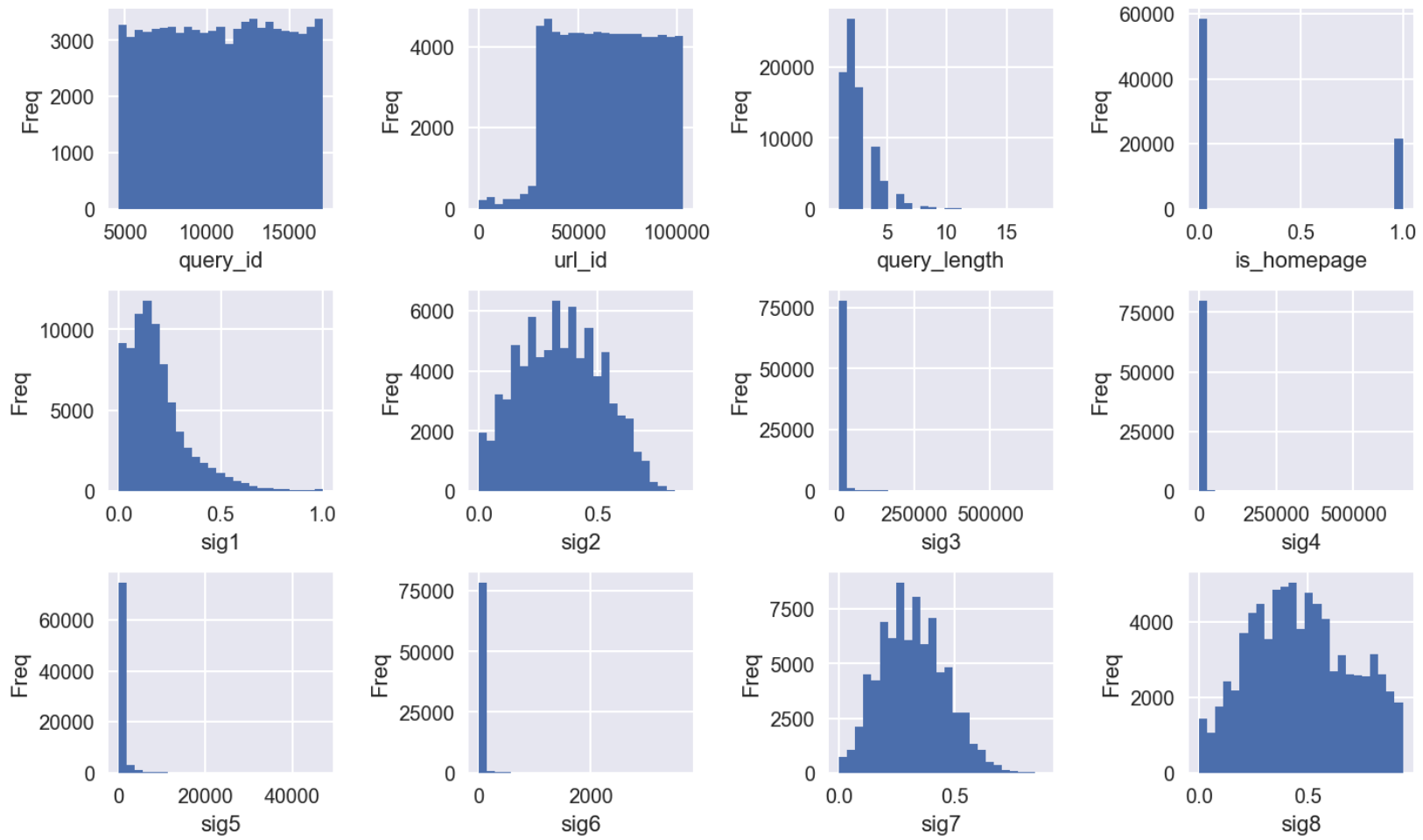
Data Abnormality

- No Null values.
- Response variable (relevance) is nearly balanced.
- 56.3% of data is labeled as 0, and 43.7% labeled as 1.



Data Abnormality

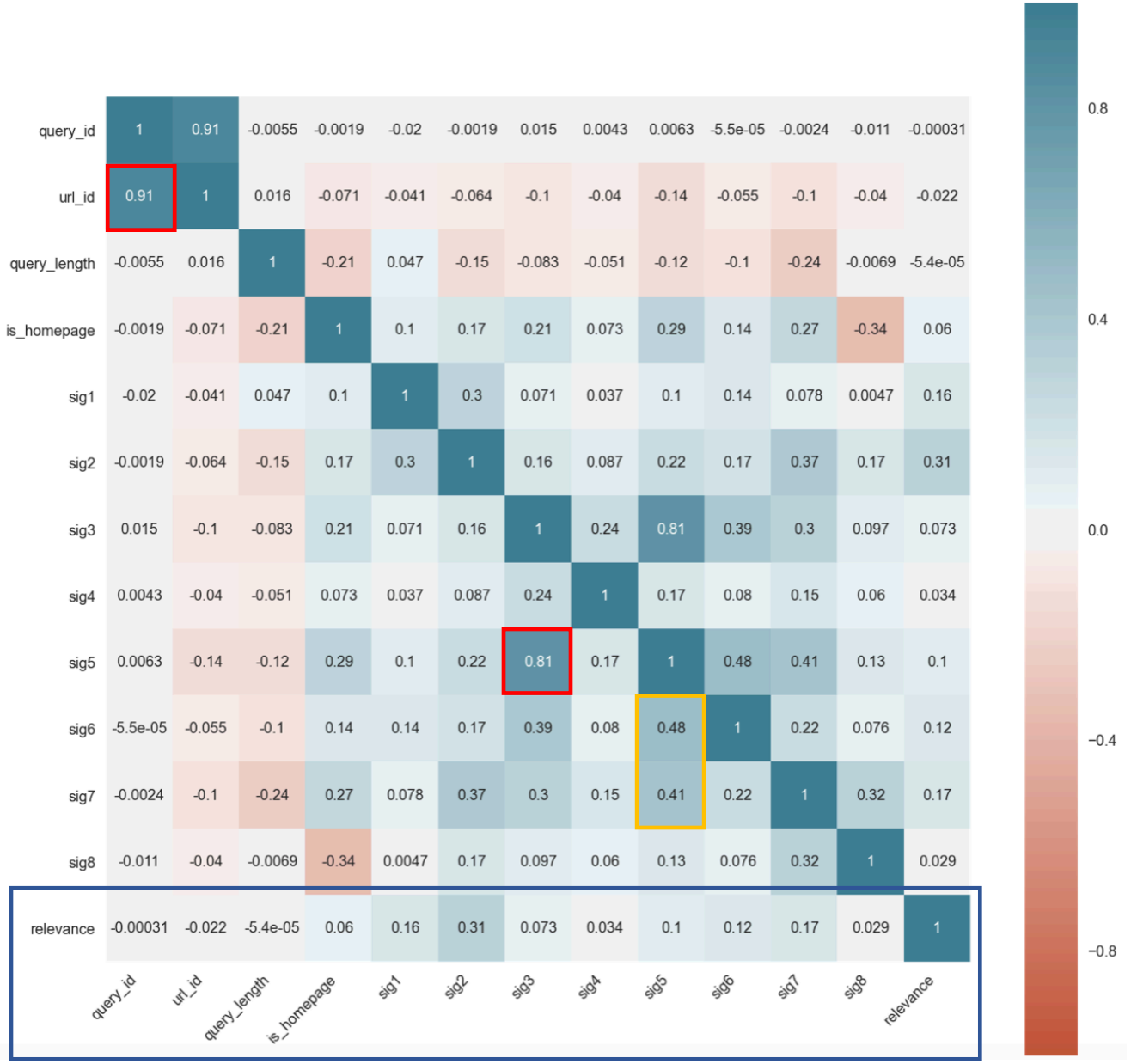
- 2 ID features show almost Uniform distribution.
- Some of features show trends of Poisson distribution.
- Extremely skewed features might need a Log-transformation



WELLS FARGO

Data Abnormality

- All the features show almost no correlation to the response variable.
- Query ID & URL ID, sig3 & sig5 show strong correlations (higher than 0.8).
- Sig5 & sig6 and sig5 & sig7 show moderate correlations (higher than 0.4).
- There is strong multicollinearity exists in the data.



WELLS FARGO

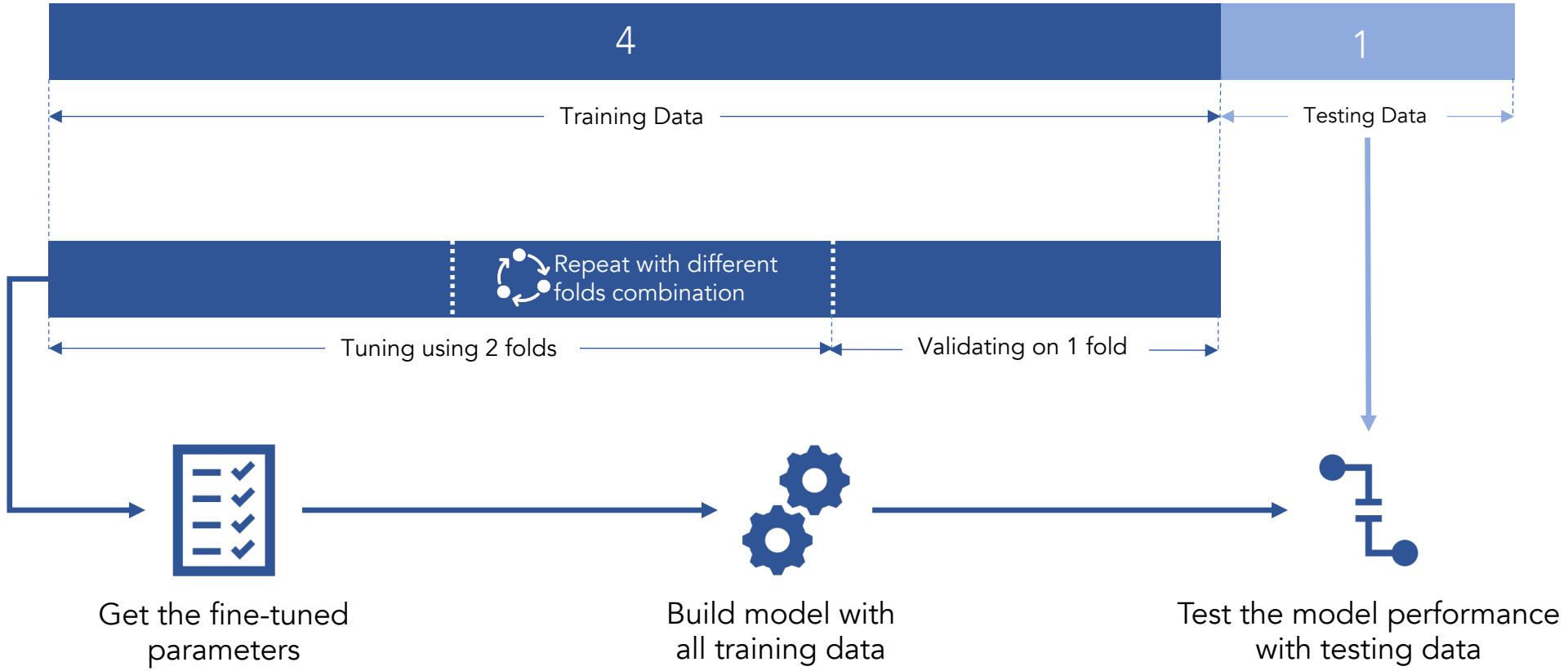
Data Preprocessing

With results coming from EDA, we need to preprocess our data:

- Remove 2 ID-related features, since they are logically just arbitrary indexes.
- Add Log-transformed features on Query Length, sig1, sig3, sig4, sig5, sig6.
- Normalize all features. This includes centering them to 0 and scales to unit variance.
- Split dataset into train and test sets with a ratio of 4:1.

Part 2: Models & Comparison

Model Pipelines

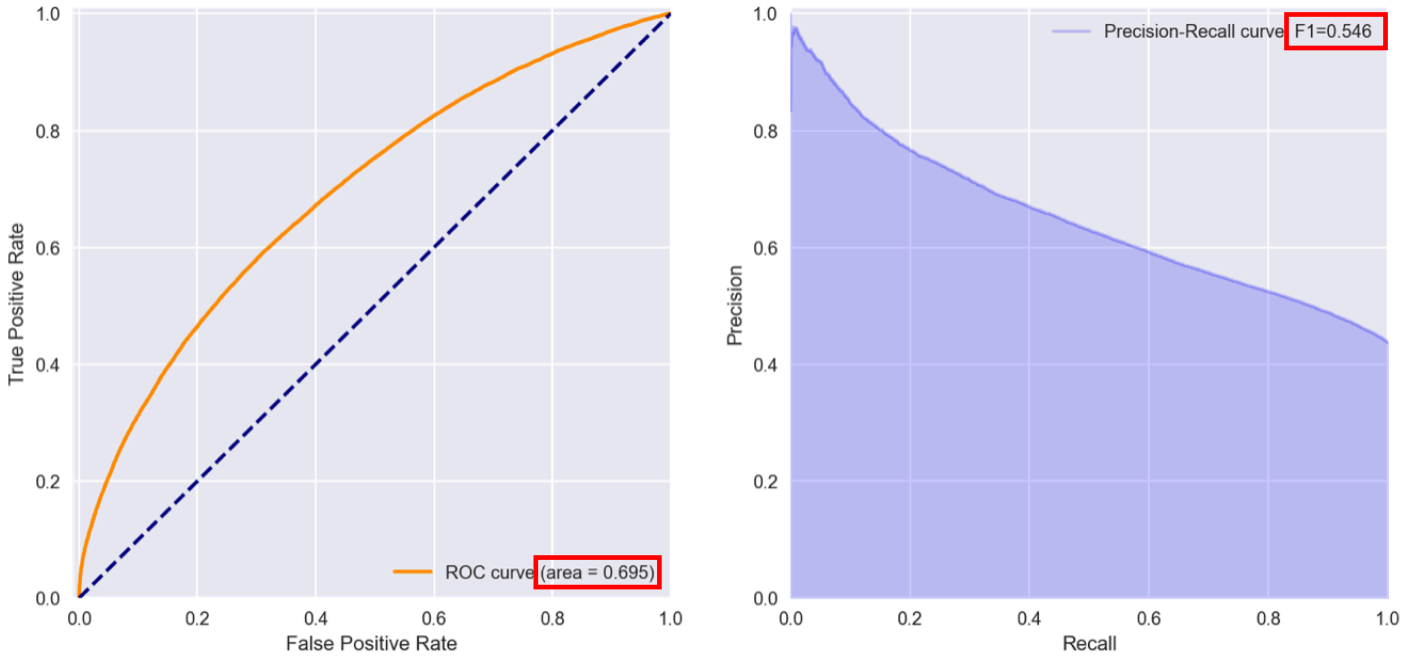


Model Metrics

This is a classification problem, and we will mainly use Receiver Operating Characteristic (ROC) curve and Precision Recall Curve. Value-wise, we will focus on AUC (Area Under Curve) score and F1 score.

- **AUC score**: Represents model's ability of distinguishing categories. Biased when data is strongly imbalanced.
- **Precision**: Basically 1-Type 1 Error (False Positive Rate)
- **Recall**: Basically 1-Type 2 Error (False Negative Rate)
- **F1 score**: Harmonic mean of Precision and Recall:

$$F1 = \frac{2*Precision*Recall}{Precision+Recall}$$



A random show case on ROC and Precision Recall curves

Model Selection: Logistic Regression

- The baseline model we used is Logistic Regression with L2 Regularization, also known as a **Ridge Regression**.
- Log-transformed terms are preferred as they are less skewed and would bring better results according to the assumptions.
- Ridge Regression would prefer data to be of a same scale, because the penalty term added to the loss function.

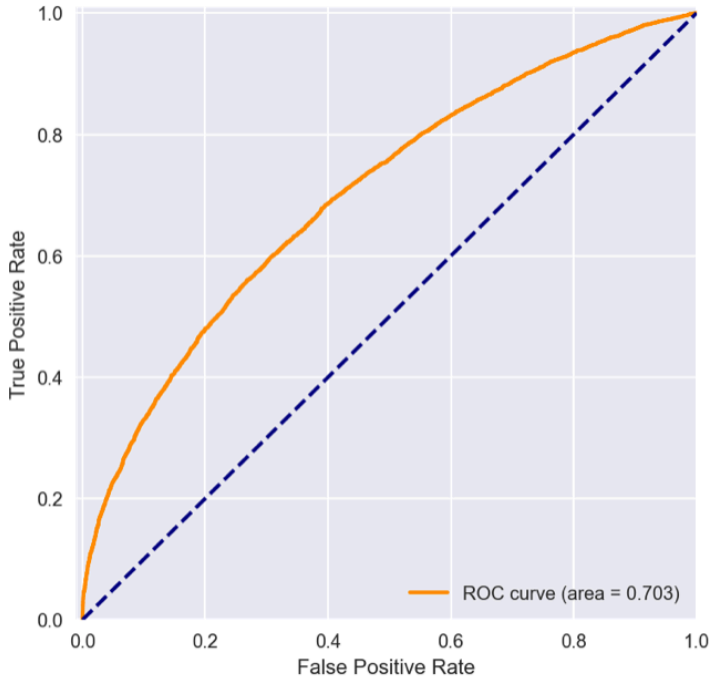
Feature Used in Ridge Regression:

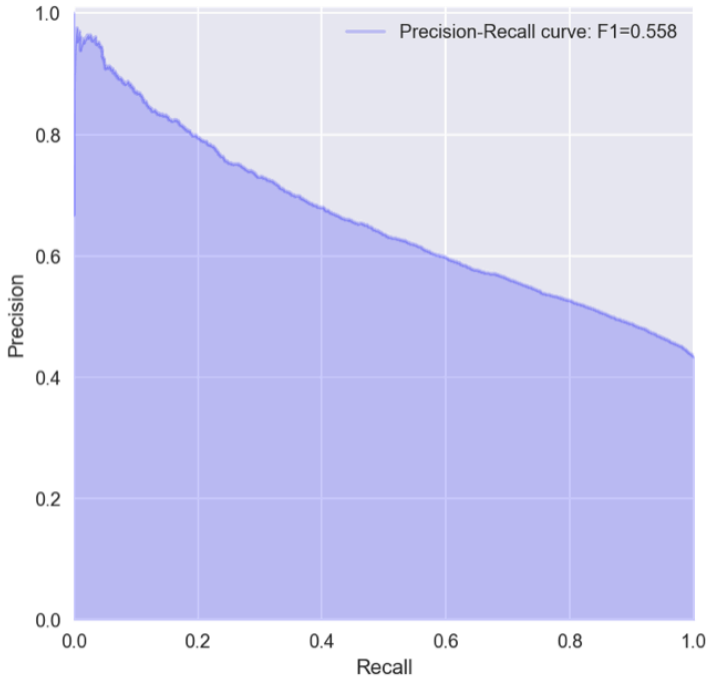
Feature Name	Query Length	Homepage Indicator	sig1	sig2	sig3	sig4	sig5	sig6	sig7	sig8
Normalized	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Log-transformed & Normalized	✓		✓		✓	✓	✓	✓		

Model Tuning: Logistic Regression

Data preprocessing affects the performance of Logistic Regression a lot.

Data	AUC score	F1 score
Original	0.560	0.288
Normalized	0.692	0.543
Log-transformed & Normalized	0.703	0.558



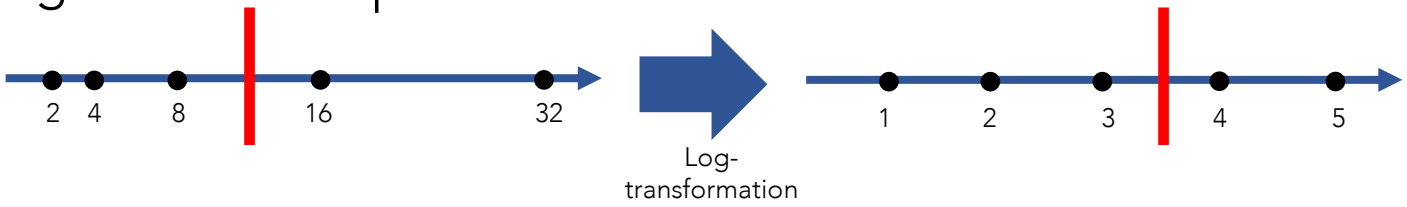


Final Logistic Regression Performance

WELLS FARGO

Model Selection: Random Forest

- Random Forests are by nature robust with multicollinearity.
- As a tree-based model, scaling, normalizing and Log-transformation does not change the model performance:



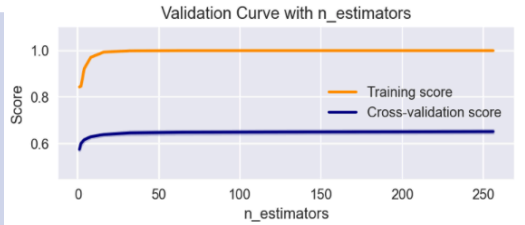
- Still, normalized data was used for consistency.

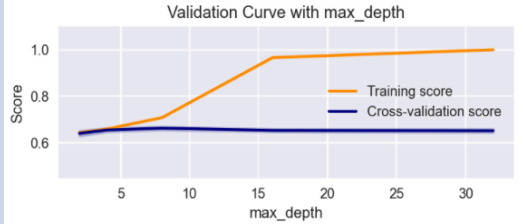
Feature Name	Query Length	Homepage Indicator	sig1	sig2	sig3	sig4	sig5	sig6	sig7	sig8
Normalized	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Log-transformed & Normalized										

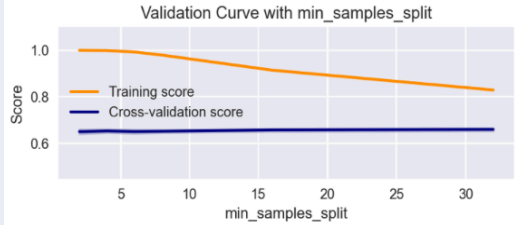
Model Tuning:
Random Forest

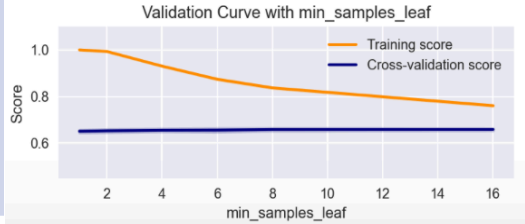
- Before "Brute Force" and searching for the best parameters, let's look at validation curves of single parameters.
- Validation curves are like "Partial Dependent Plots" focused a single parameter change.

n_estimators	Number of trees in random forest
max_features	Number of features to consider at every split
max_depth	Maximum number of levels in tree
min_samples_split	Minimum number of samples required to split a node
min_samples_leaf	Minimum number of samples required at each leaf node







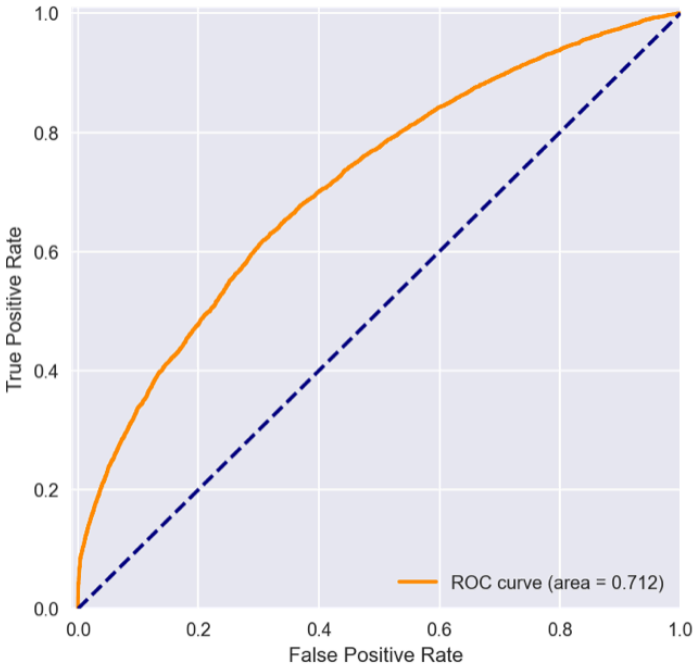


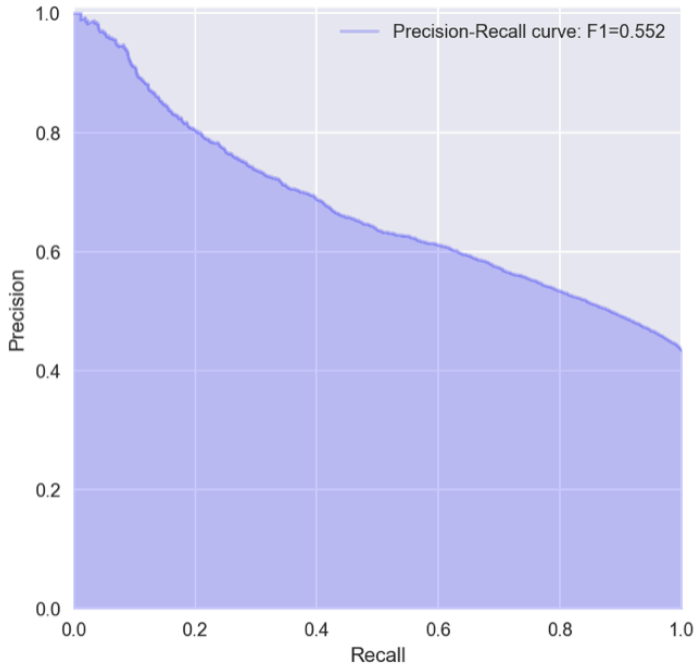
WELLS FARGO

Model Tuning: Random Forest

After having some ideas on parameters' range, we then used Grid Search for the best parameter combination.

n_estimators	[50, 100, 150, 200]	200
max_features	[2, 3, 4, 5]	2
max_depth	[4, 8, 16, 32]	16
min_samples_split	[2, 4, 6, 8]	6
min_samples_leaf	[4, 6, 8, 16]	8





Model Selection: Light GBM

- Light GBM is a gradient boosting framework by Microsoft.
- Same as Random Forests, scaling, normalizing and Log-transformation does not change the GBM performance.
- Considering we might use regularization in Light GBM, we still used normalized data:

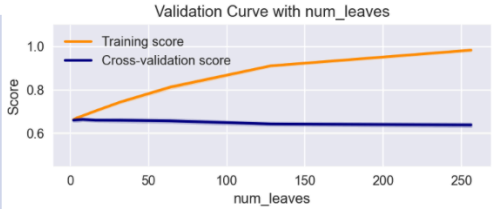
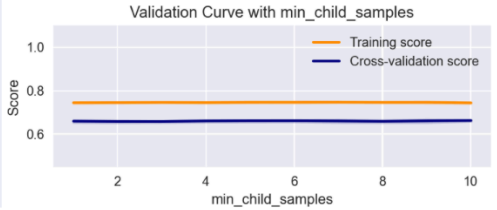
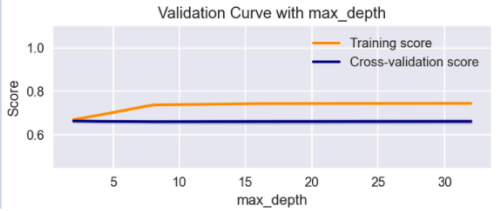
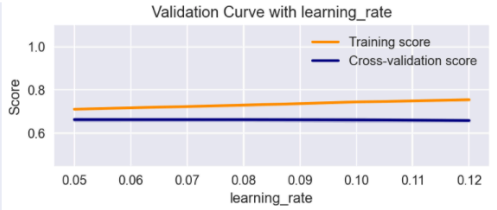
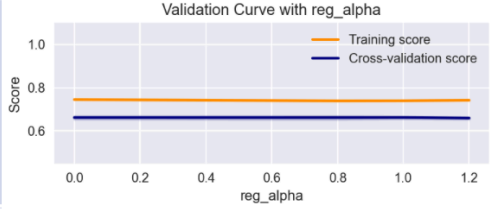
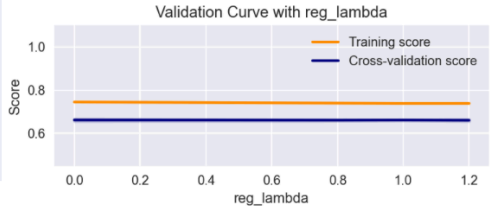
Feature Name	Query Length	Homepage Indicator	sig1	sig2	sig3	sig4	sig5	sig6	sig7	sig8
Normalized	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Log-transformed & Normalized										

Model Tuning: Light GBM

Validation Curves

Grid Search

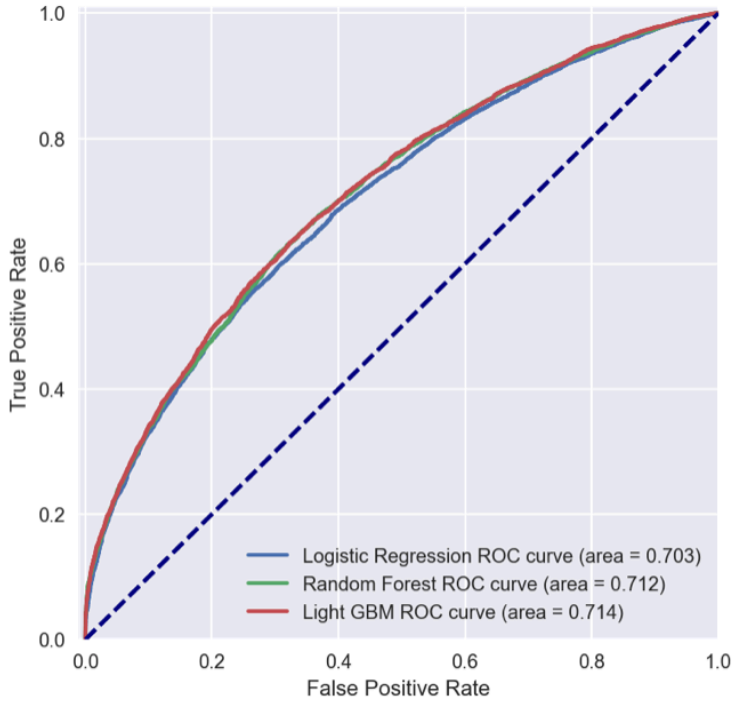
Get Best Parameters

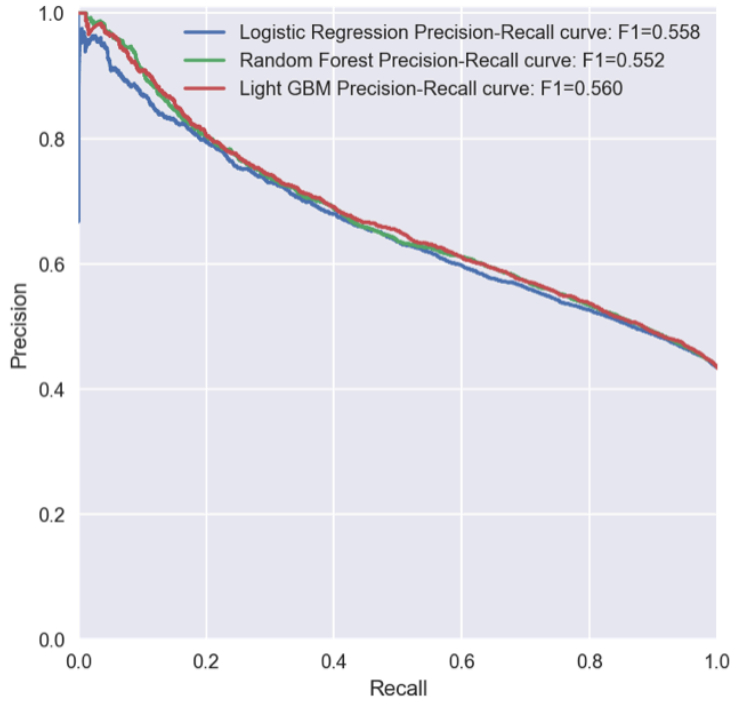
num_leaves	Maximum tree leaves for base learners.		[4, 8, 16, 32, 64]	16
min_child_ samples	Minimum number of data needed in a child (leaf).		[1, 2, 4, 8, 16, 32]	2
max_depth	Maximum tree depth for base learners.		[3, 4, 5, 6, 7, 8]	6
learning_rate	Learning Rate		[0.05, 0.08, 0.1, 0.12]	0.08
reg_alpha	L1 Regularization		[0, 0.8, 1, 1.2]	1
reg_lambda	L2 Regularization		[0, 0.8, 1, 1.2]	0

WELLS FARGO

Model Comparison

Model	AUC score	F1 score	Run Time (s)
Logistic Regression	0.703	0.558	0.190
Random Forest	0.712	0.552	7.537
Light GBM	**0.714**	**0.560**	**0.455**





WELLS FARGO

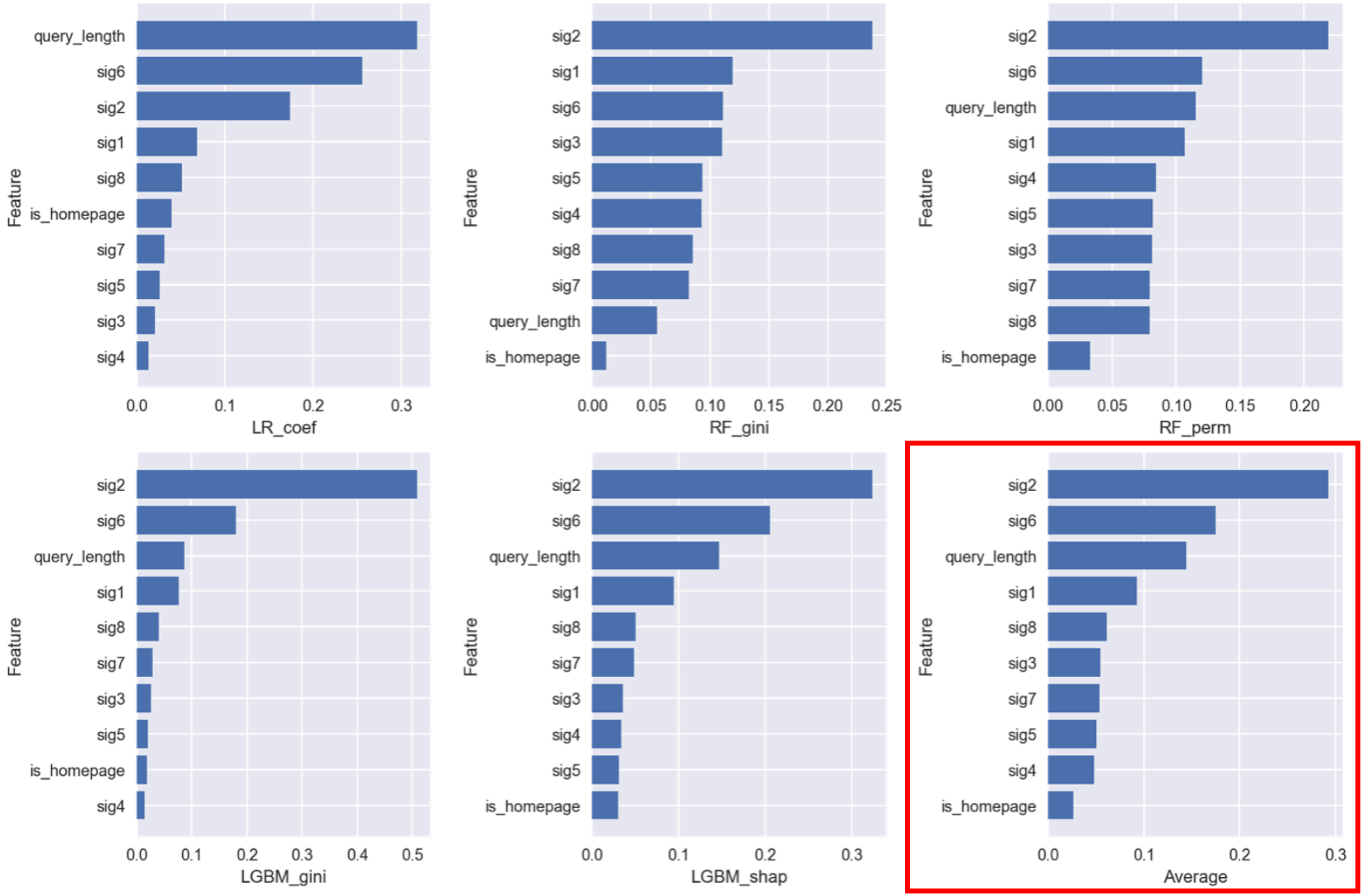
Part 3: Feature Interpretation & Recommendation

Feature Importance

Feature importance is hard to define:

- Coefficients: Parameter from linear models that can reflect the importance of feature.
- Gini importance: Mean Decrease in Impurity, defined as number of splits with the feature, weighted by the number of samples it splits.
- Permutation importance: Mean Decrease in Accuracy, defined as the decrease in model score when a single feature value is randomly shuffled.
- Shap Values: A way to distribute prediction among the features, defined as the average difference between the model with and without a feature. Originated from Game Theories.

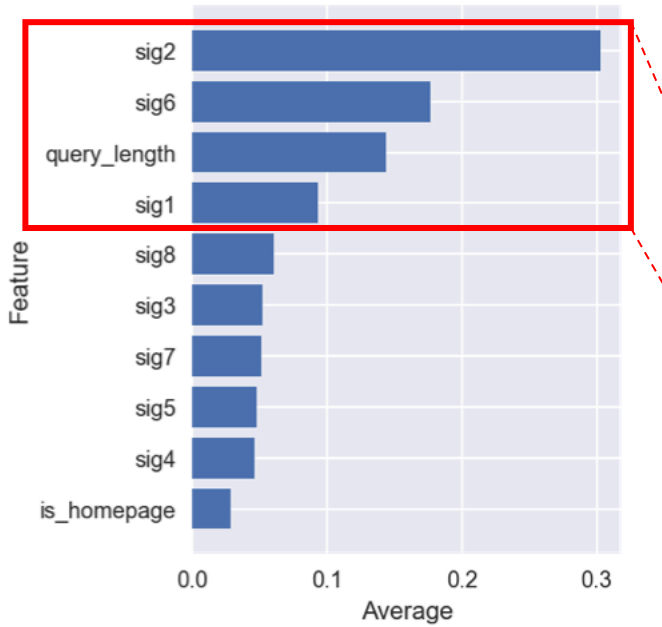
Feature Importance



WELLS FARGO

Feature Importance

In fact, the top 4 features are so important that if we only include those 4 features and fit the Light GBM model, we still have adequate performance:



Model	AUC score	F1 score
Logistic Regression	0.703	0.558
Random Forest	0.712	0.552
Light GBM	0.714	0.560
Light GBM with 4 features	**0.702**	**0.544**

WELLS FARGO

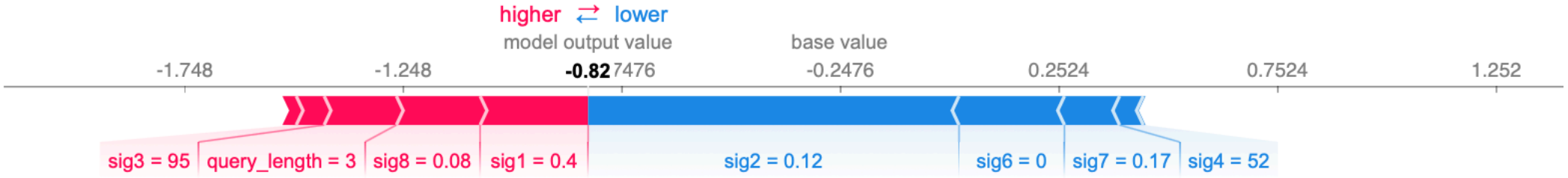
Feature Interpretation

Now if we focus on a random row from data:

Feature Name	Query Length	Homepage Indicator	sig1	sig2	sig3	sig4	sig5	sig6	sig7	sig8	Relevance
Original Value	3	0	0.40	0.12	95	52	22	0	0.17	0.08	0

We can explain our model prediction by using Shap Values:

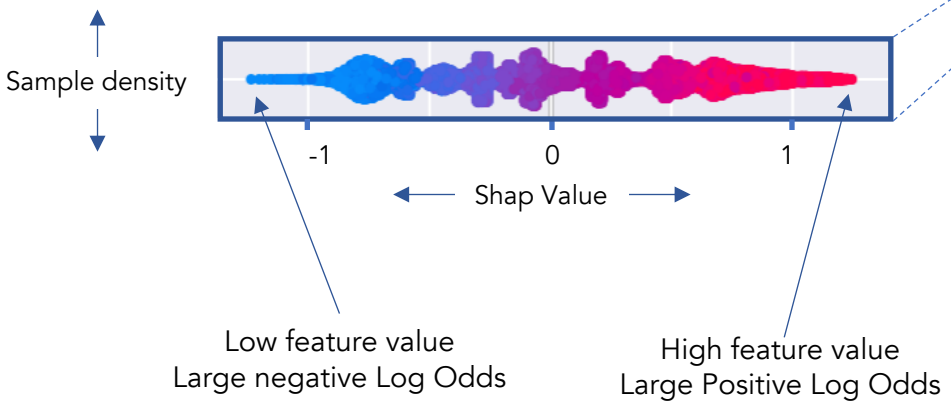
- Output value here is Log Odds.
- Negative value indicates a prediction towards 0, vice versa.

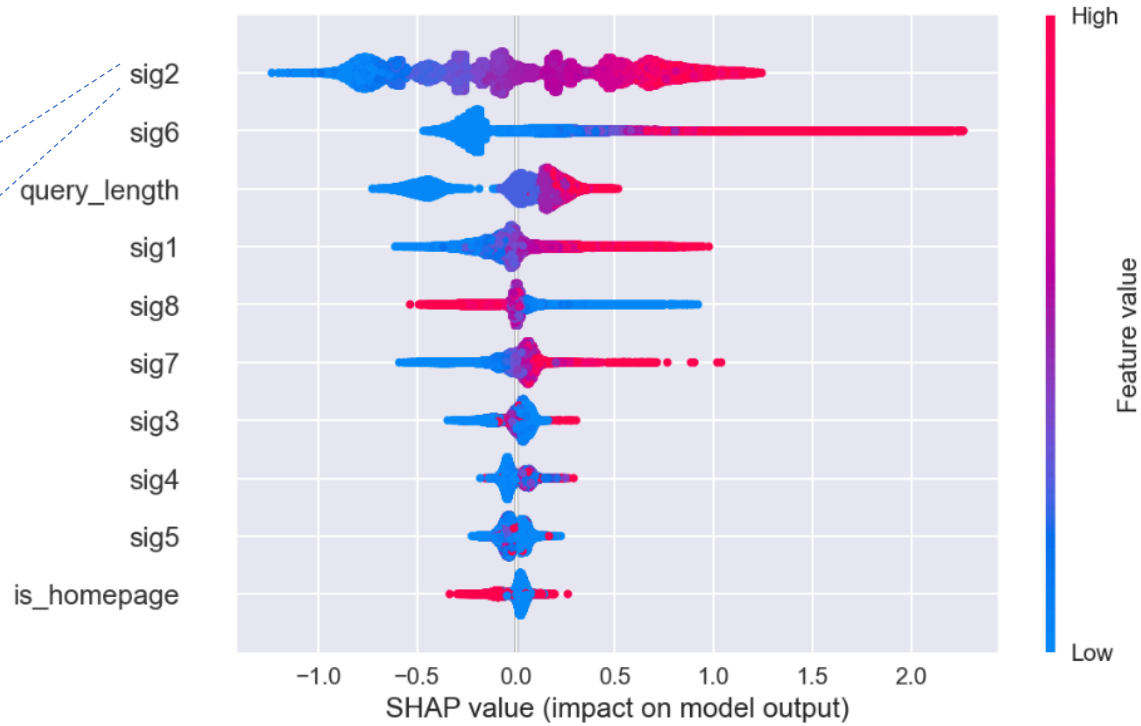


WELLS FARGO

Feature Interpretation

We can also get a general interpretation of all our data:

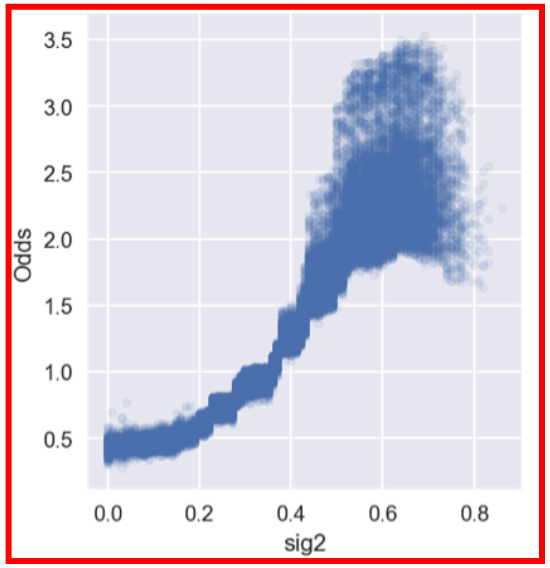


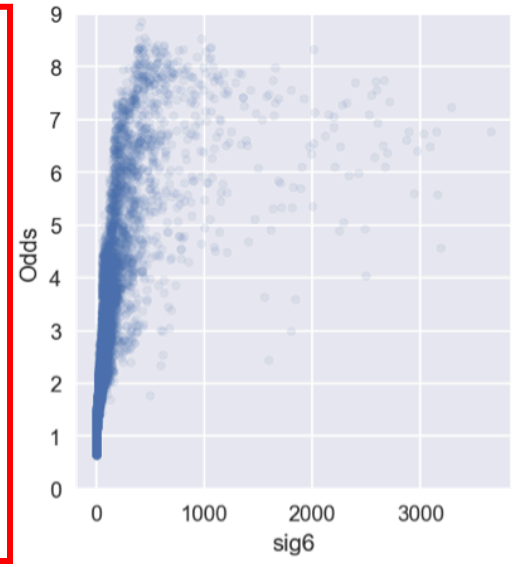


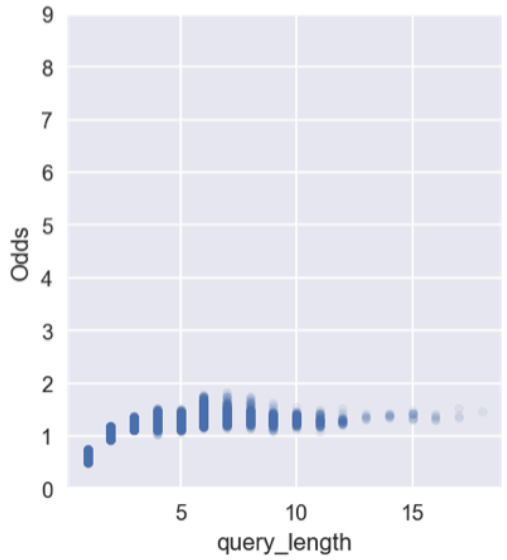
Feature Interpretation

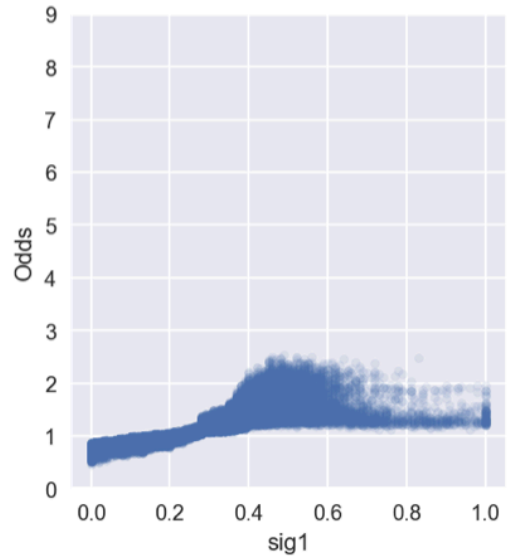
Using Shap Values, we can interpret features' influences on predictions.

Comparing with	New value	Odds ratio	Interpretation
sig2 = 0, odds = 0.5	sig2 = 0.2, odds = 0.6	$\frac{0.6}{0.5}$ =1.2	20% more chance to be relevant
	sig2 = 0.4, odds = 1.3	$\frac{1.3}{0.5}$ =2.6	160% more chance to be relevant
	sig2 = 0.6, odds = 2.2	$\frac{2.2}{0.5}$ = 4.4	340% more chance to be relevant









WELLS FARGO

Recommendation

- We should focus more on the following 4 features:
  sig2, sig6, Query Length, sig1

- These 4 features show positive impact on response: the higher their values are, the more likely the response will be relevant.

- Longer queries results in higher relevance: develop better key word associate system.

- If possible, focus on actual meanings of sig2, sig6 and sig1.