kaggle

Prudential

PRUDENT CHOICE

Build a model to help Prudential come up with a response to life insurance applicants

Poonam Rath
Data Science Final Project, 2016
General Assembly, San Francisco

PROJECT GOAL

Analyze features collected from life insurance applicants provided by Prudential and build a model to predict the company's likely response.

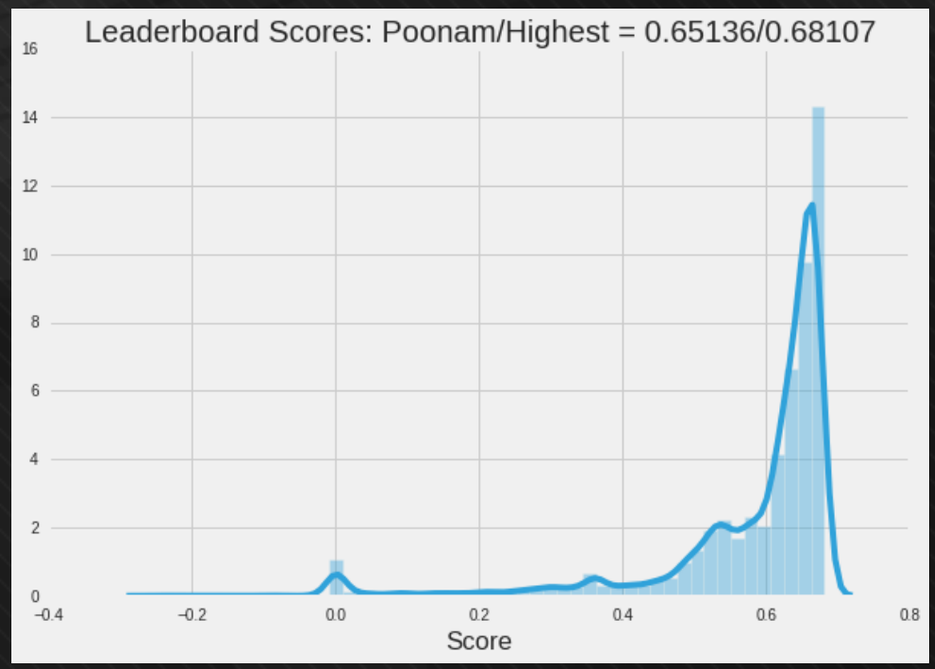
Motivation for the problem:

- Less labor-intensive for customers
- Quicker time-to-decision for Prudential
- Potential to gain efficiencies on both sides of the equation

AGENDA

1. Analysis approach
2. Results
3. Conclusions
4. Next Steps

RESULT



Initial Model

Quadratic weighted Kappa = 0.46

[Used Random Forest classifiers with no feature engineering]

???

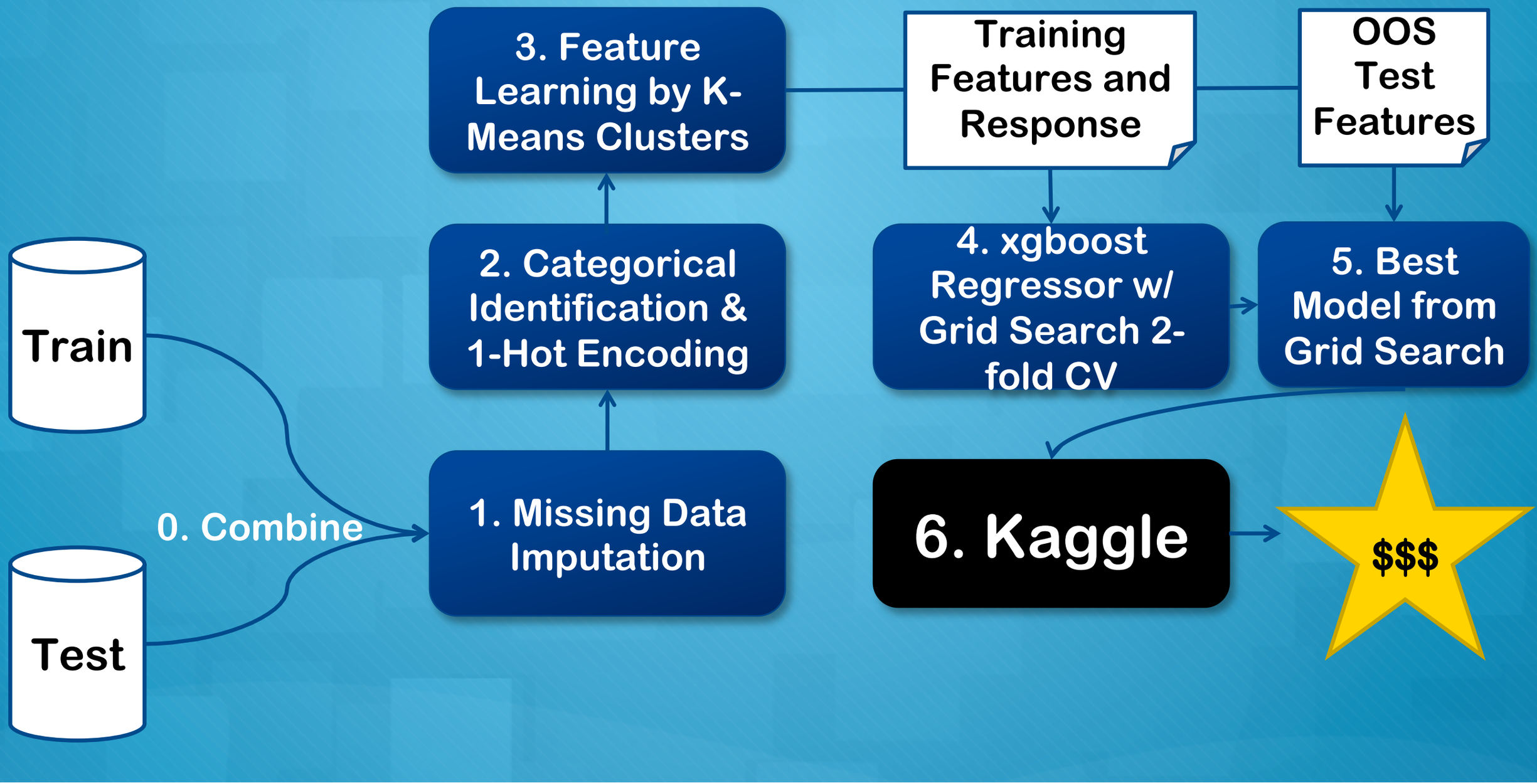
Final Model

Quadratic weighted Kappa = 0.65136

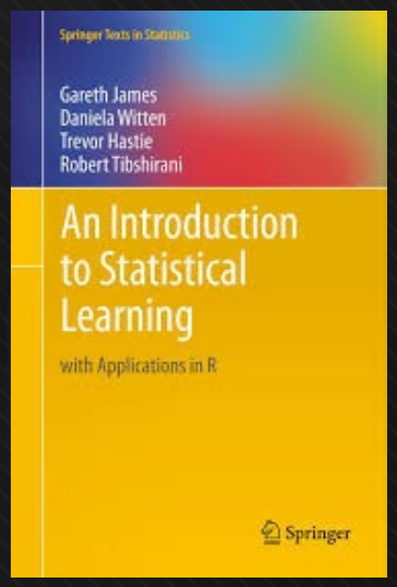
[Used xgboost + grid search + custom features]

... How did I do this ?

ANALYSIS APPROACH

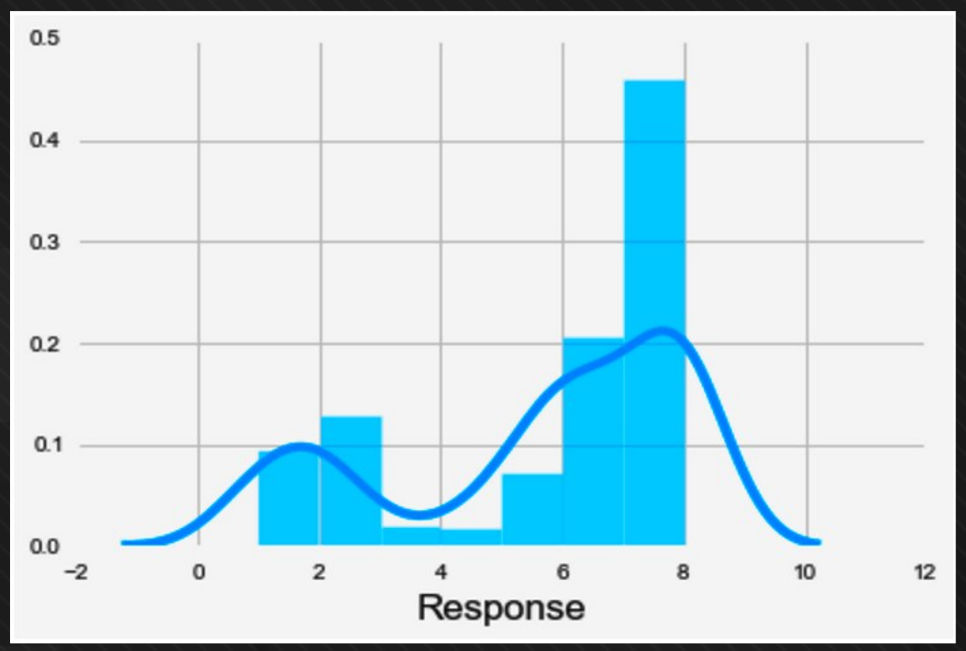


RESOURCES USED FOR ANALYSIS



OVERVIEW OF DATA

- The training data has ~60k rows and 127 columns: 126 features and 1 response column. Outcome column is called “Response” and is nominal with 8 values (1,2,3,4,5,6,7,8).



OVERVIEW OF DATA

- Most features are de-identified; we have a vague idea about what they might represent: “Product_Info”, “Employment_Info”, “Medical_keyword”.
- All but one feature columns are numeric.
- Features are normalized (according to Kaggle). If not, I would have looked at histograms, determined mean and stdev and scaled accordingly.

FEATURE ENGINEERING

- **Missing value imputation:** Used the median value of the columns.
- **1-Hot Encoding**: Obtained dummy variables for columns for which unique values were less than 0.5% of the total values.
- NaN count: Counted all NaN's row-wise
- Medical keyword count: Count all occurrences row-wise

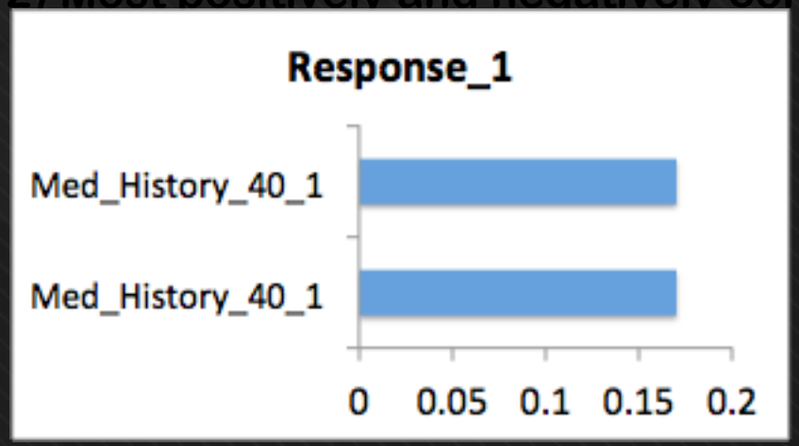
FEATURE LEARNING

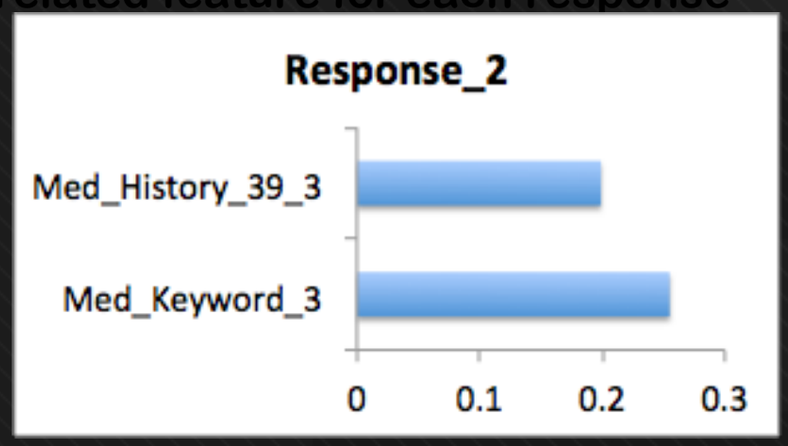
- Picked up idea from Kaggle forums
- Run K-Means clustering (K=200) on entire dataset
- For each row, measure distance from cluster center
- Add K=200 additional feature columns

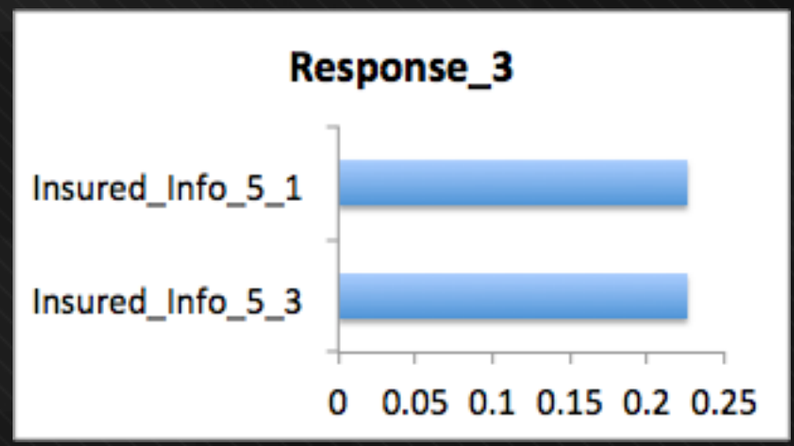
We do this to capture information about how rows are grouping in high-dimensional space.

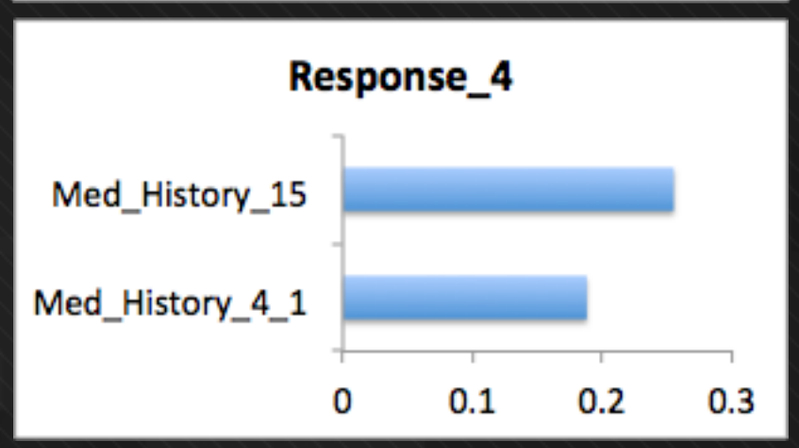
Exploratory data analysis -I

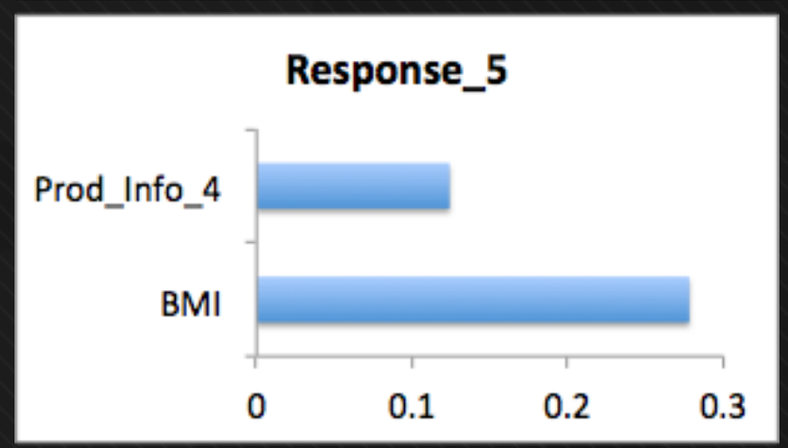
2) Most positively and negatively correlated feature for each response

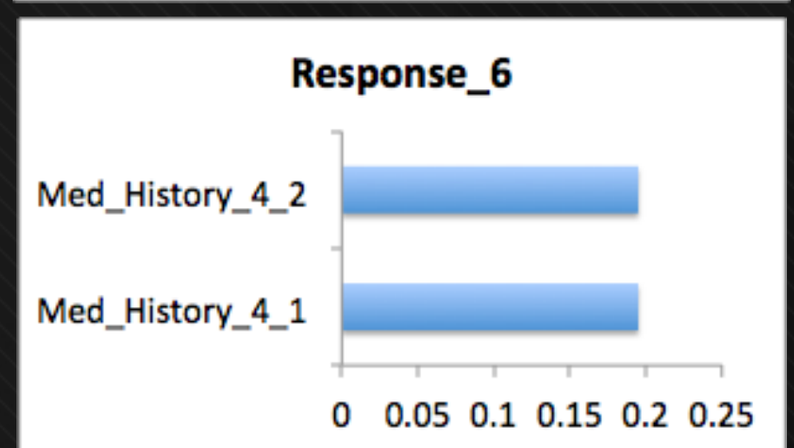


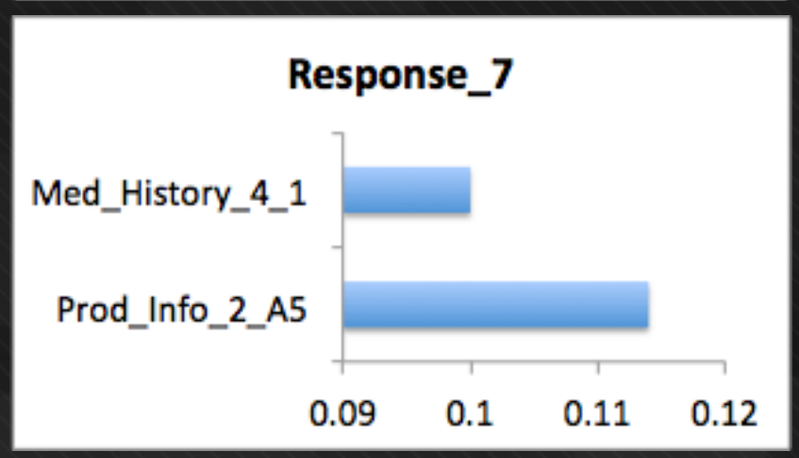


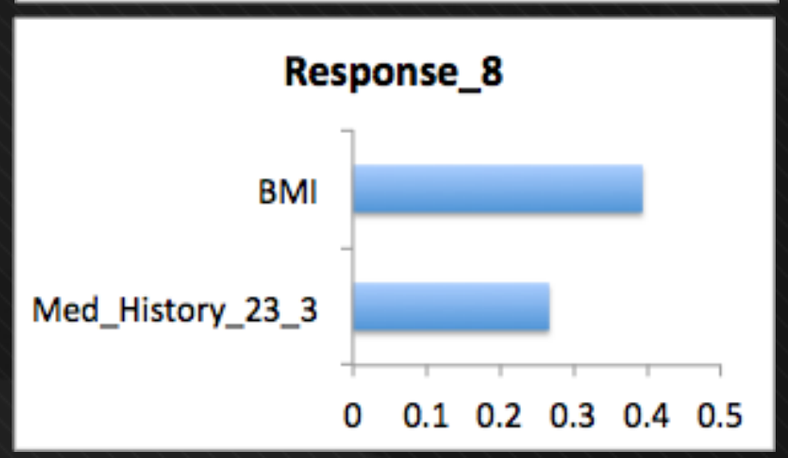






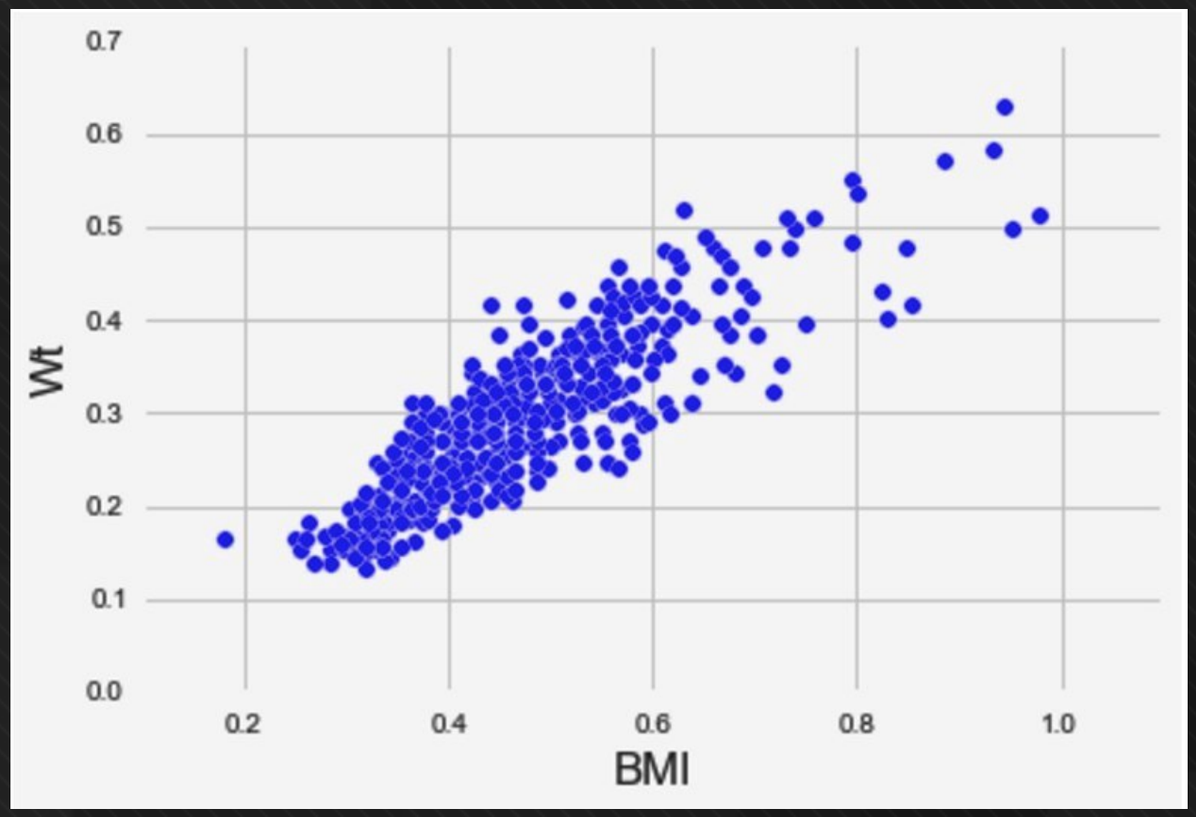






Top = +ve
Bottom = -ve

Exploratory data analysis -II

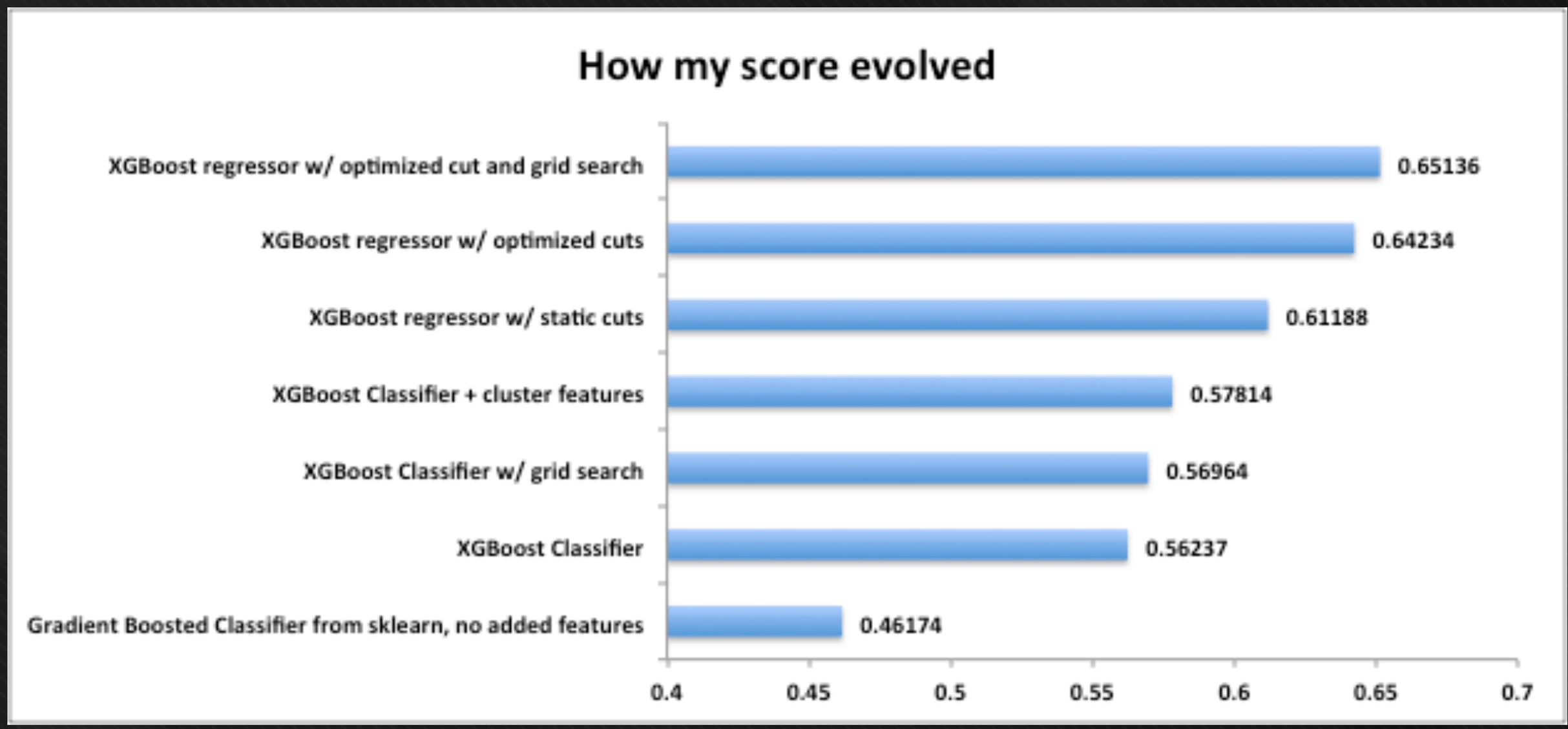


BMI is positively correlated with Weight (subset of data)

Measuring Model Performance

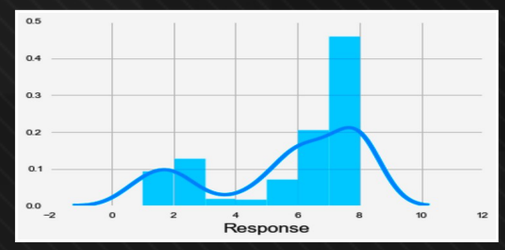
- Logistic Regression: 33% accuracy
- K-Nearest Neighbor Classifier: 21%
- Naive Bayes: 19%
- Gradient Boosting Classifier: 46.6% accuracy

CONCLUSIONS



NEXT STEPS

- Make Response column more even : SMOTE/undersample [though the forums said it would not make a difference.]



- Use data from my estimator for feature selection
- Try to learn and use neural networks!