

Graduate Admissions

Semester project for EECS 738

Guided by Dr. Martin Kuehnhausen

Team

Jan Polzer

Kunal Karnik

Nishil Parmar

Rohan Choudhari

Ryan Duckworth

Roadmap

- Problem statement
- Data Set
- Exploratory Data Analysis
- Random Forest Regressor
- XGBoost
- Neural Network
- Multivariate Linear Regression
- Ridge Regression
- Negative Binomial Distribution

Problem Statement

- Applying to Master's program is a very expensive and intensive work!
- Use Machine Learning to help students get to know their chance of admit before they apply
- Rarely few schools with no application fee
- Most application fees fall between \$50 - \$85
- Graduate program application fee at KU \approx \$65 - \$85 too expensive !!

Data Set

<https://www.kaggle.com/mohansacharya/graduate-admissions>

This dataset is created for prediction of Graduate Admissions
Inspired by the UCLA Graduate Dataset

Mohan S Acharya, Asfia Armaan, Aneeta S Antony:

A Comparison of Regression Models for Prediction of Graduate
Admissions

IEEE International Conference on Computational Intelligence in
Data Science 2019

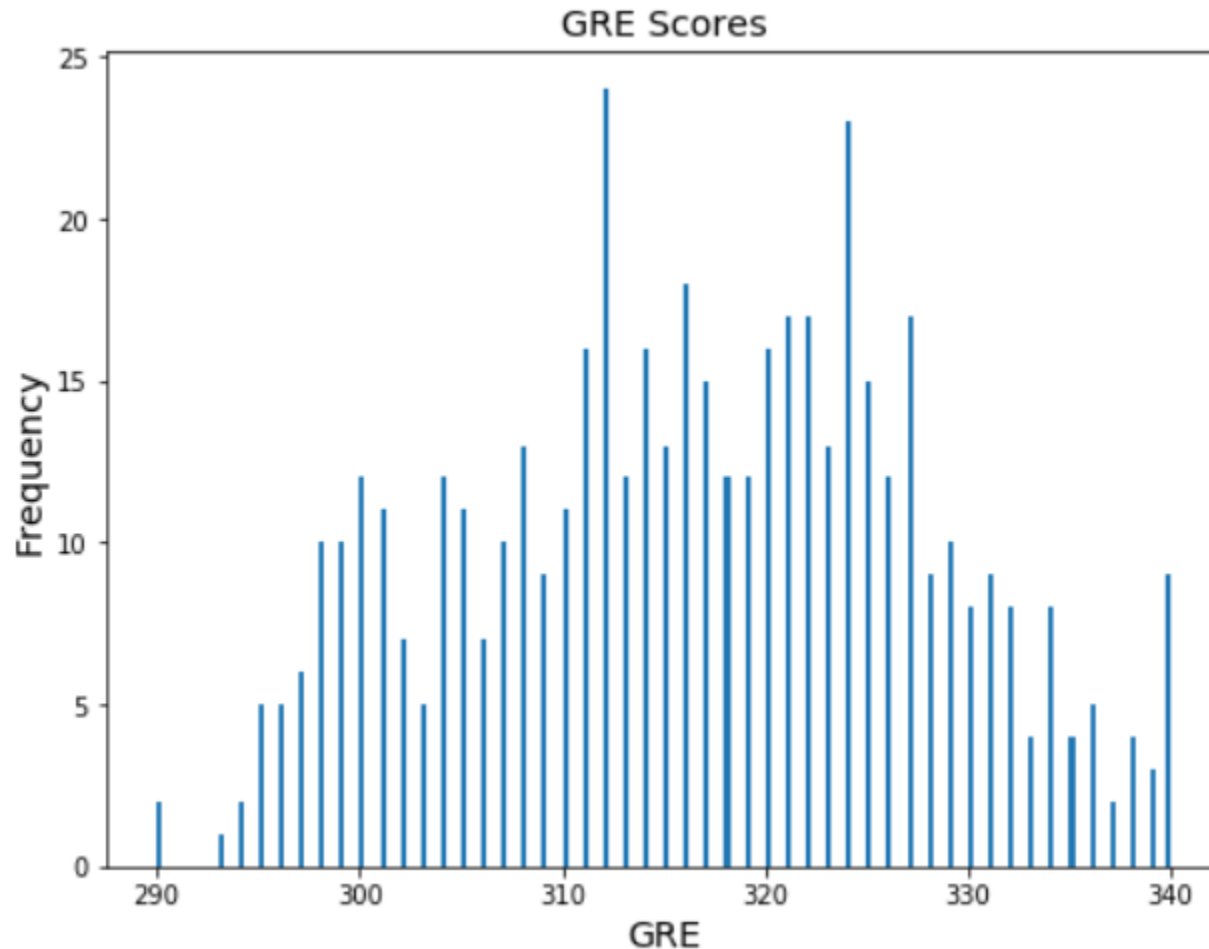
Data Set

The dataset contains several parameters which are considered important during the application for Masters Programs.

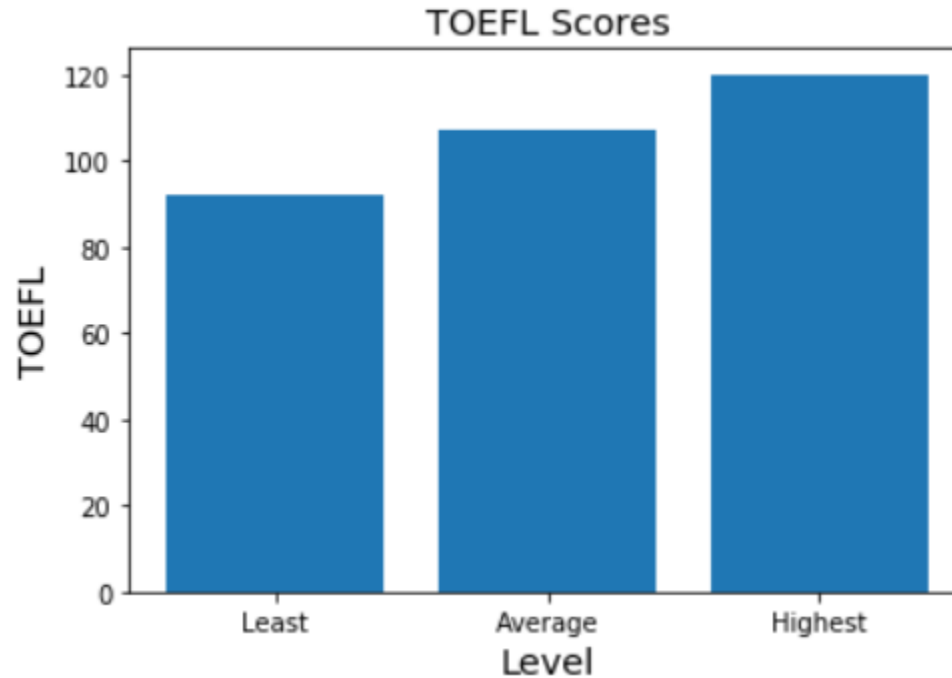
The parameters included are:

1. GRE Scores (out of 340)
2. TOEFL Scores (out of 120) Test of English as a Foreign Language
3. University Rating (out of 5)
4. Statement of Purpose and Letter of Recommendation Strength (out of 5)
5. Undergraduate GPA (out of 10)
6. Research Experience (either 0 or 1)
7. Chance of Admit (ranging from 0 to 1)

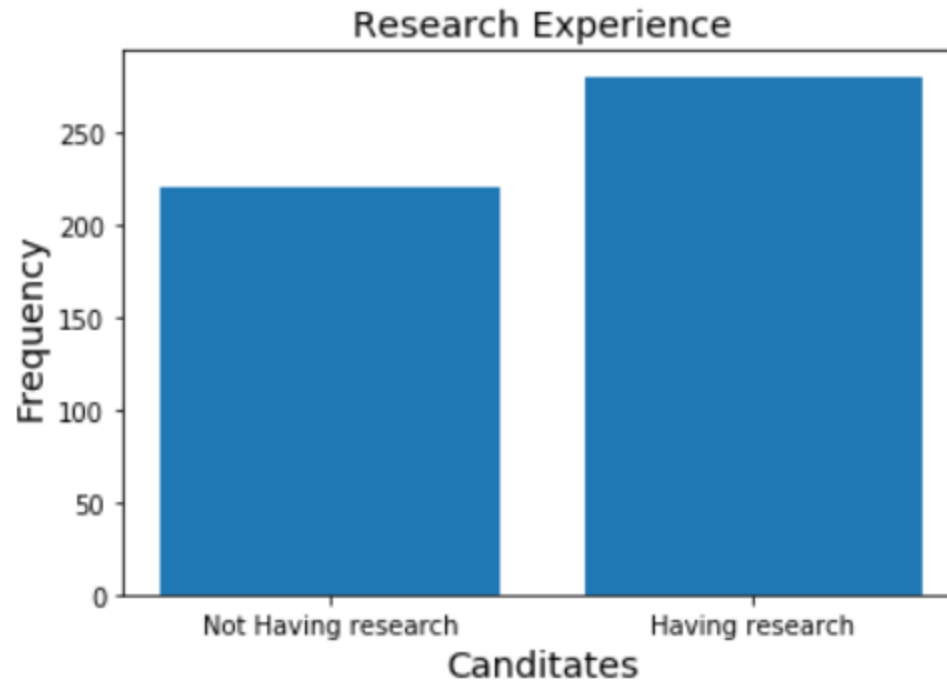
Exploratory Data Analysis

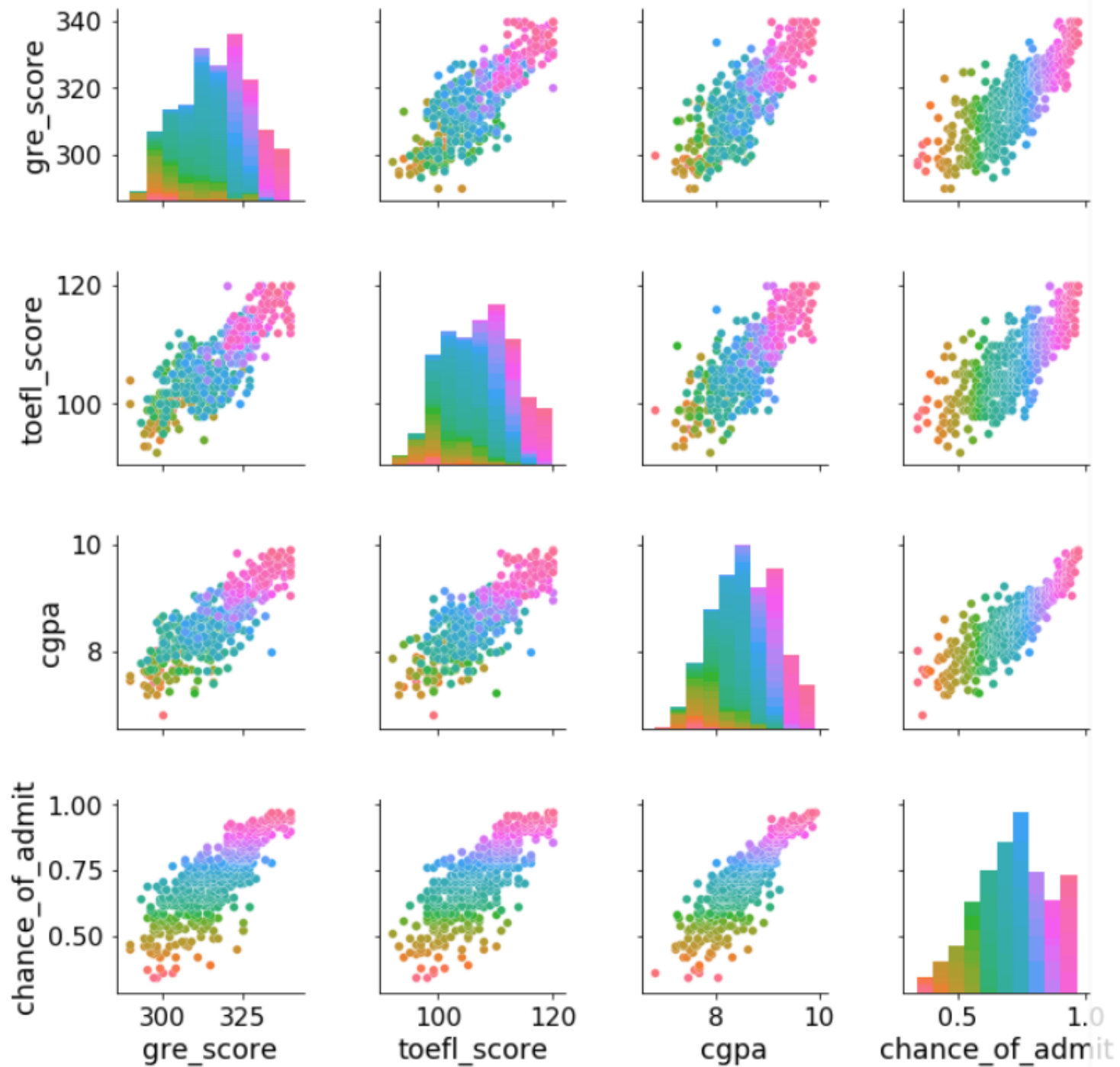


Exploratory Data Analysis

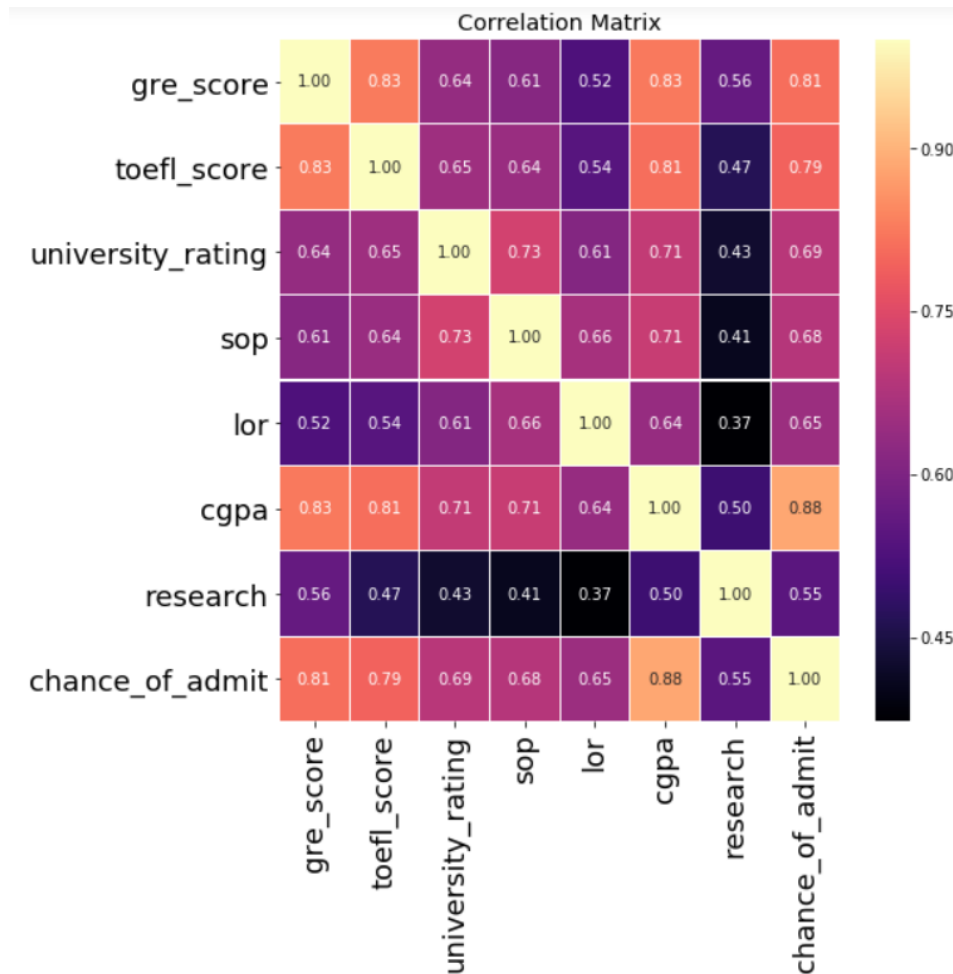


Exploratory Data Analysis





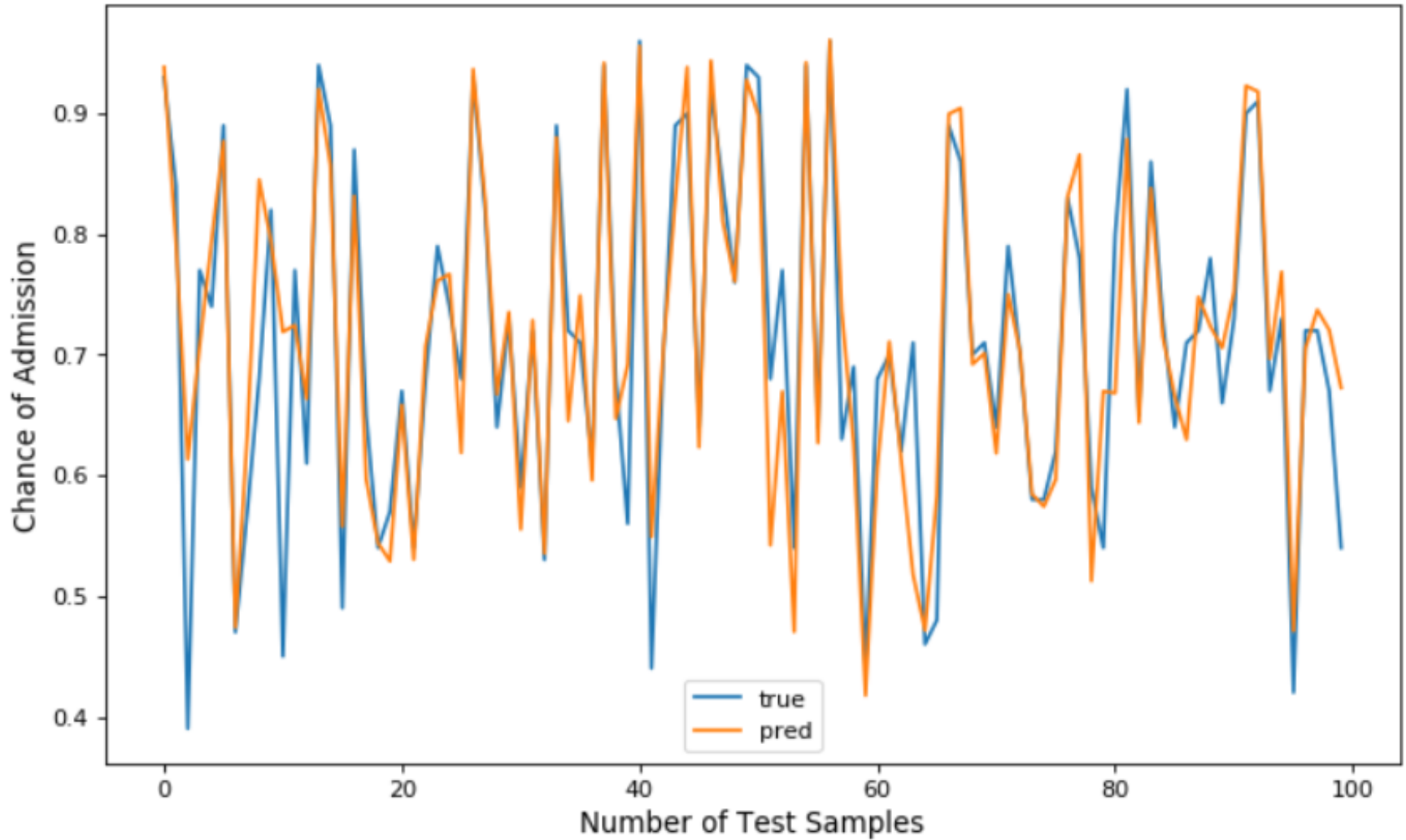
Exploratory Data Analysis



Random Forest Regressor

- Ensemble learning method for regression
- Operates by constructing a multitude of decision trees at training and outputting mean prediction of individual trees
- used with bootstrapping - better model performance, it decreases variance without increasing bias
- Number of trees = 100
- R^2 score = 0.7885063612224901
- RMSE = 0.06576507365615947

Random Forest Regressor



XGBoost

- XGBoost stands for e**X**treme **G**radient **B**oosting.
- "XGBoost is an open-source software library which provides a gradient boosting framework" –Wikipedia

<https://en.wikipedia.org/wiki/XGBoost>

Boosting

"Boosting is an ensemble technique where new models are added to correct the errors made by existing models. Models are added sequentially until no further improvements can be made. A popular example is the AdaBoost algorithm that weights data points that are hard to predict."

Gradient Boosting

"Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models and then added together to make the final prediction. It is called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models."

XGBoost

Machine Learning presentation given in June 2016:

"17 out of 29 winning solutions in Kaggle last year used XGBoost "

- Tianqi Chen

"...CERN recognized it as the best approach to classify signals from the Large Hadron Collider. This particular challenge posed by CERN required a solution that would be scalable to process data being generated at the rate of 3 petabytes per year and effectively distinguish an extremely rare signal from background noises in a complex physical process. XGBoost emerged as the most useful, straightforward and robust solution."

Gradient Boosting

"Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function. " –Wikipedia

https://en.wikipedia.org/wiki/Gradient_boosting

Why is XGBoost so popular?

Machine Learning Challenge Winning Solutions

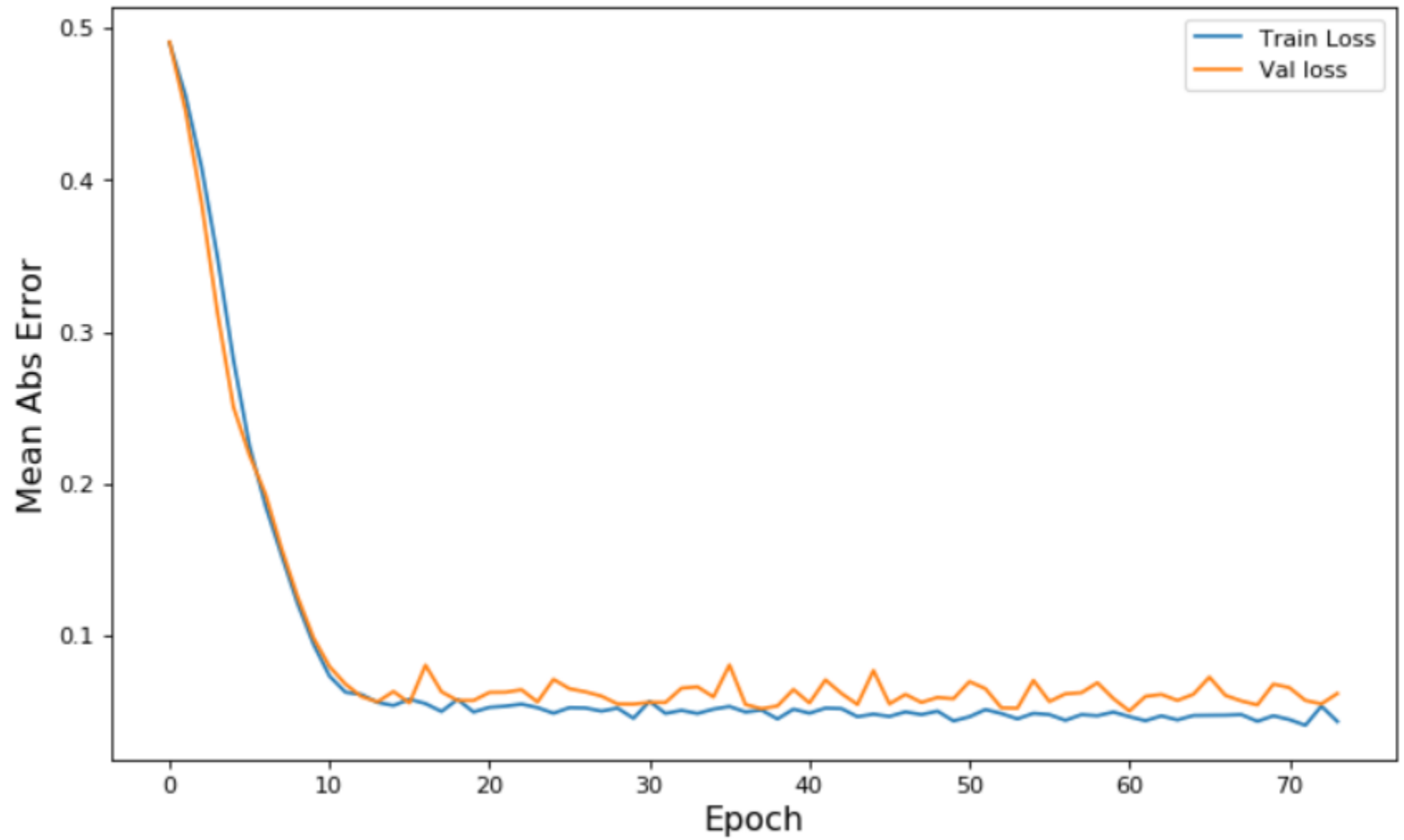
XGBoost is extensively used by machine learning practitioners to create state of art data science solutions, this is a list of machine learning winning solutions with XGBoost. Please send pull requests if you find ones that are missing here.

- Maksims Volkovs, Guangwei Yu and Tomi Poutanen, 1st place of the [2017 ACM RecSys challenge](#). Link to [paper](#).
- Vlad Sandulescu, Mihai Chiru, 1st place of the [KDD Cup 2016 competition](#). Link to [the arxiv paper](#).
- Marios Michailidis, Mathias Müller and HJ van Veen, 1st place of the [Dato Truly Native? competition](#). Link to [the Kaggle interview](#).
- Vlad Mironov, Alexander Guschin, 1st place of the [CERN LHCb experiment Flavour of Physics competition](#). Link to [the Kaggle interview](#).
- Josef Slavicek, 3rd place of the [CERN LHCb experiment Flavour of Physics competition](#). Link to [the Kaggle interview](#).
- Mario Filho, Josef Feigl, Lucas, Gilberto, 1st place of the [Caterpillar Tube Pricing competition](#). Link to [the Kaggle interview](#).
- Qingchen Wang, 1st place of the [Liberty Mutual Property Inspection](#). Link to [the Kaggle interview](#).
- Chenglong Chen, 1st place of the [Crowdfunder Search Results Relevance](#). Link to [the winning solution](#).
- Alexandre Barachant ("Cat") and Rafał Cycoń ("Dog"), 1st place of the [Grasp-and-Lift EEG Detection](#). Link to [the Kaggle interview](#).
- Halla Yang, 2nd place of the [Recruit Coupon Purchase Prediction Challenge](#). Link to [the Kaggle interview](#).
- Owen Zhang, 1st place of the [Avito Context Ad Clicks competition](#). Link to [the Kaggle interview](#).
- Keiichi Kuroyanagi, 2nd place of the [Airbnb New User Bookings](#). Link to [the Kaggle interview](#).
- Marios Michailidis, Mathias Müller and Ning Situ, 1st place [Homesite Quote Conversion](#). Link to [the Kaggle interview](#).

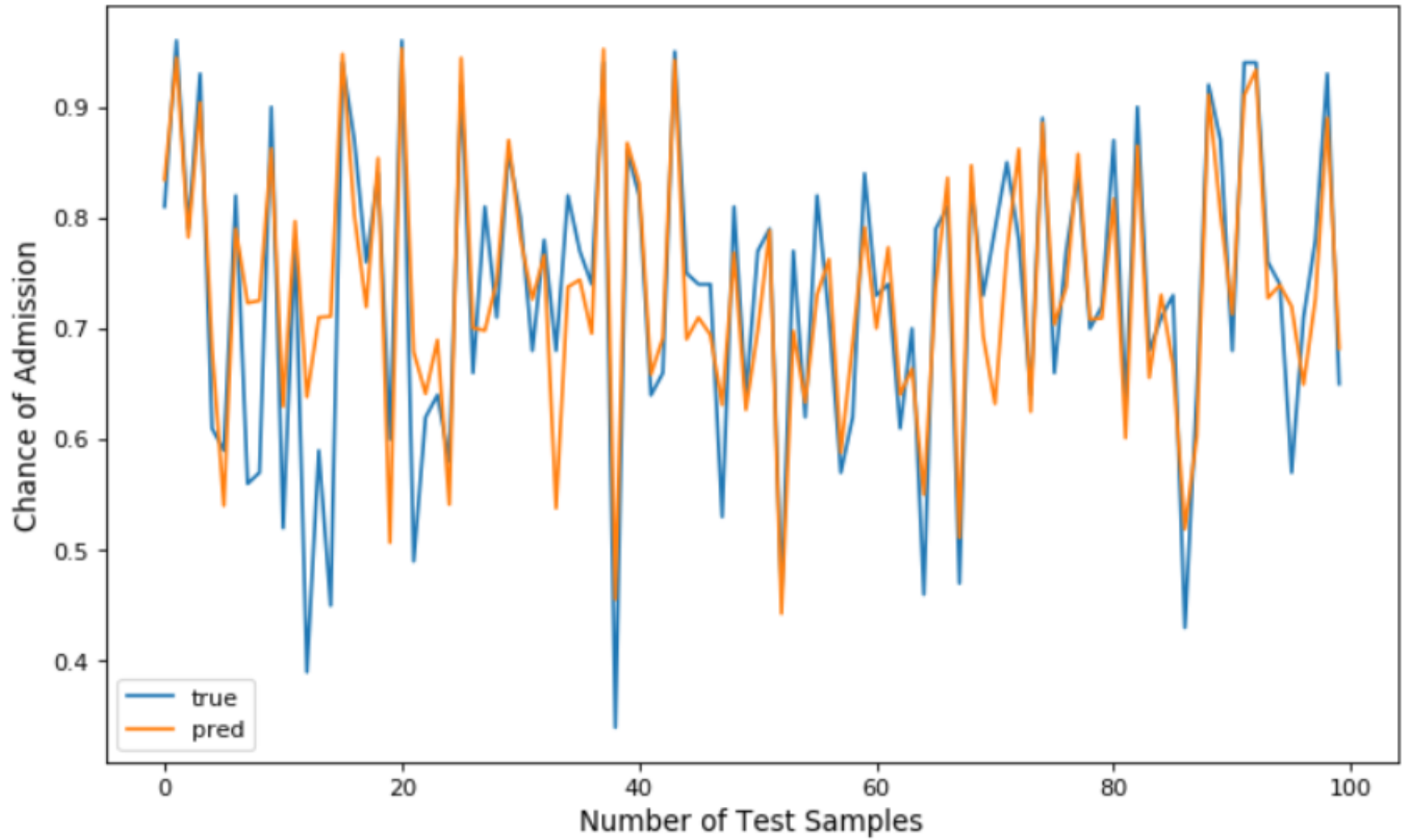
<https://github.com/dmlc/xgboost/tree/master/demo#machine-learning-challenge-winning-solutions>

Neural Network

- Used sequential model
 - linear stack of layers
- 3 layer dense neural network
- Activation function: *ReLu -Rectified linear units*
- Loss function: mean squared error
- Learning rate: 0.001
- Epochs: 80
- RMSE = 0.07099454017742718



Neural Network



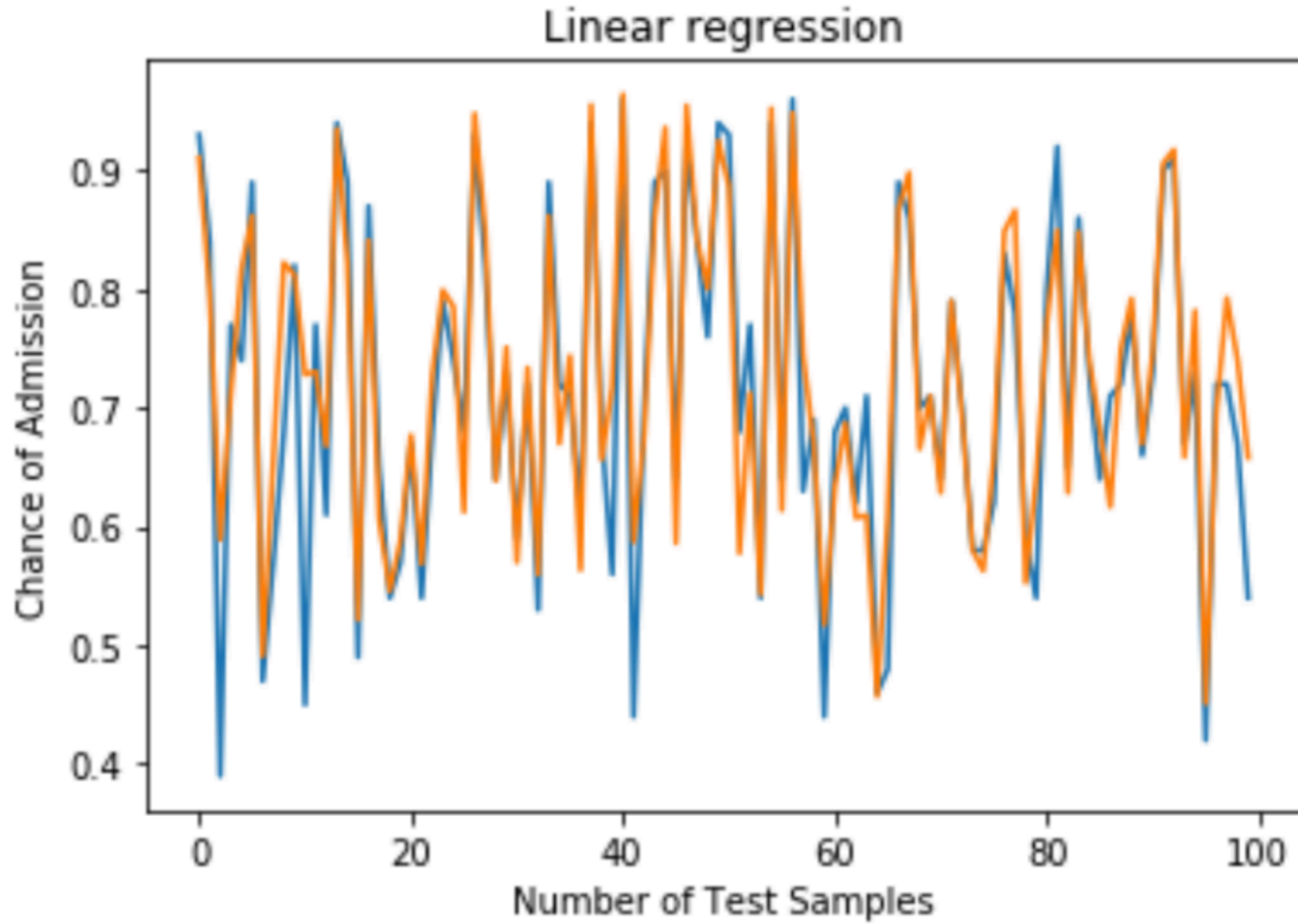
Multivariate Linear Regression

Multiple linear regression attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data

$$Y_i = \alpha + \beta_1 x_i^{(1)} + \beta_2 x_i^{(2)} + \dots + \beta_n x_i^{(n)}$$

Multivariate Linear Regression

- Used 3 modules
 - Hypothesis function
 - Gradient Descent Algorithm function
 - Linear Regression principal function
- Learning rate: 0.001
- Epochs: 1000
- RMSE = 0.0618767204629624



Ridge Regression

- Ridge Regression is a technique for analyzing multiple regression data that suffer from **multicollinearity**. When multicollinearity occurs, least squares estimates are unbiased, but their variances are large so they may be far from the true value. By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors. It is hoped that the net effect will be to give estimates that are more reliable.

Multicollinearity

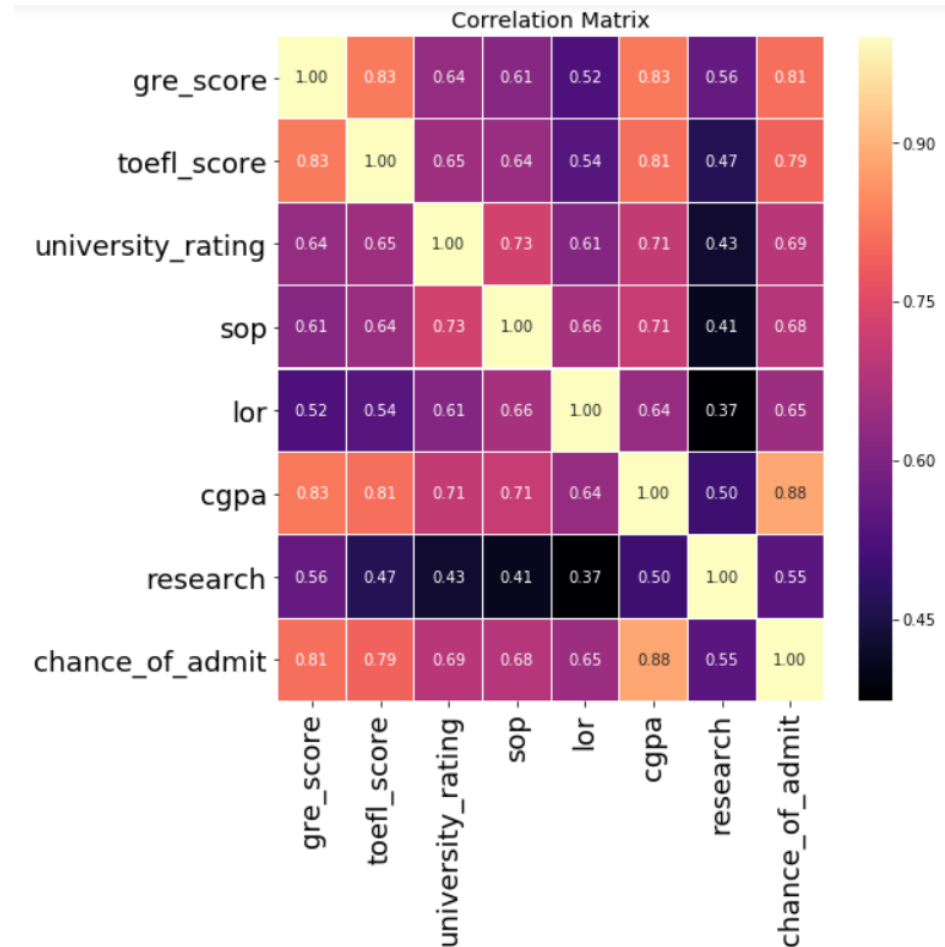
- Multicollinearity is the existence of near-linear relationships among the independent variables.
- Effects:
 - can create inaccurate estimates of the regression coefficients
 - inflate the standard errors of the regression coefficients
 - degrade the predictability of the model

Detection of Multicollinearity

- Correlation Matrix
 - Construction of a correlation matrix among the explanatory variables will yield indications as to the likelihood that any given couplet of right-hand-side variables are creating multicollinearity problems. Correlation values (off-diagonal elements) of at least 0.4 are sometimes interpreted as indicating a multicollinearity problem.

Correlation Matrix

We can see there's significantly high correlation (>0.8) between 'GRE' and 'TOEFL', 'CGPA' and 'GRE', and 'TOEFL' and 'CGPA'



Detection of Multicollinearity

- Studying pairwise scatter plots
 - We take a look at scatter plots of pairs of independent variables.
 - We consider 'GRE', 'TOEFL', 'CGPA' because the correlation matrix suggested that the three have high correlation

Scatter plots

A cursory look at the scatter plots hint at the presence of multicollinearity. However, multicollinearity does not always show up when considering two variables at a time. We need to take a few more steps before we confirm multicollinearity



Detection of Multicollinearity

- Variance Inflation Factors (VIF)
 - VIF is the ratio of variance in a model with multiple terms, divided by the variance of a model with one term alone
 - It quantifies the severity of multicollinearity and provides an index that measures how much the variance of an estimated regression coefficient is increased because of collinearity.

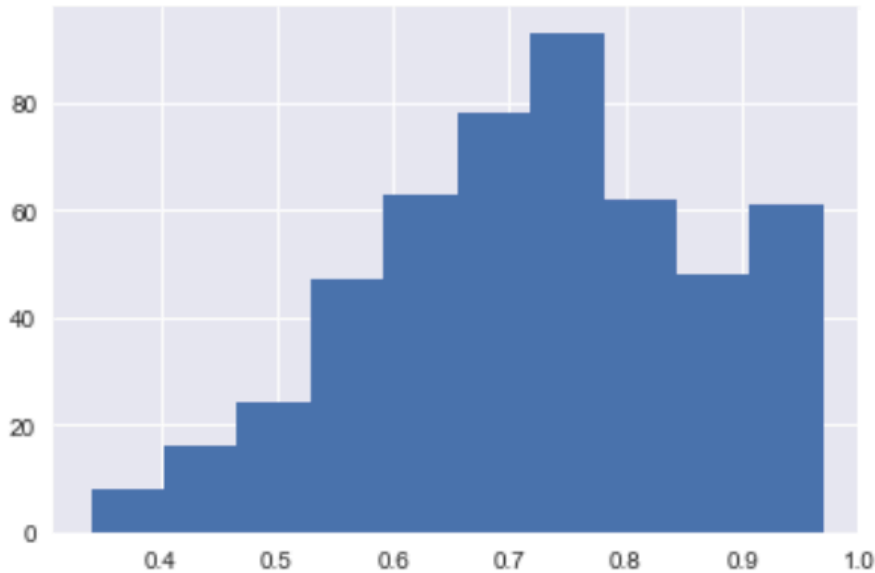
Detection of Multicollinearity

A $VIF > 5$ is indication that the particular explanatory variable is highly collinear with other explanatory variables. When a variable is independent, it is not affected at all by any other variables. When a variable isn't independent for certain, it's an explanatory variable.

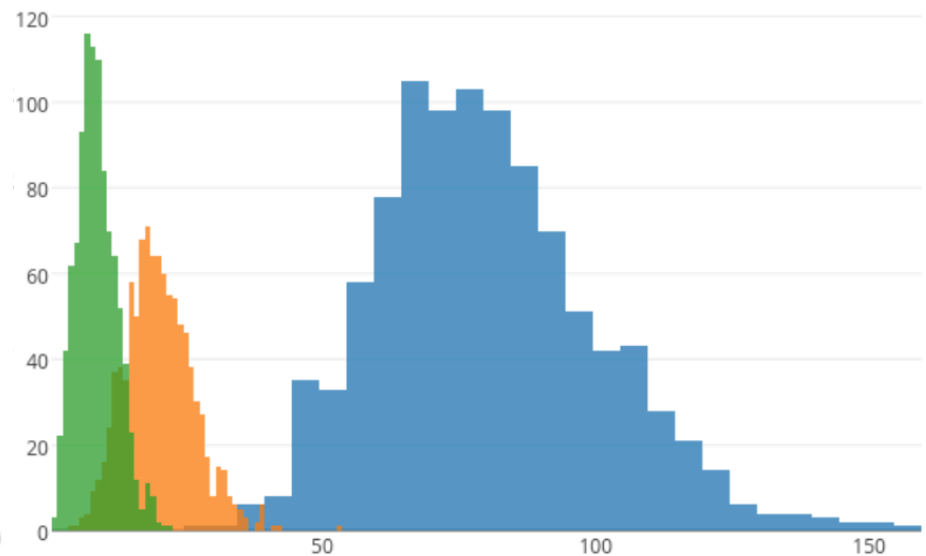
```
VIF for GRE: 1504.894809819234
VIF for TOEFL: 1243.5023702219323
VIF for UR: 21.590945361773844
VIF for SOP: 35.444504213092884
VIF for LOR : 31.73387111365384
VIF for CGPA: 1238.0957552630935
VIF for Research: 3.2512997969491497
VIF for Acceptance: 117.11781672786486
```


Negative Binomial Regression

Chance of admit



The Negative Binomial Distribution



Negative Binomial Regression

- 2 Versions :
 - first version counts the number of the trial at which the r th success occurs
 - second version counts the number of failures before the r th success.

$$P(X_1 = x | p, r) = \binom{x-1}{r-1} p^r (1-p)^{x-r}.$$

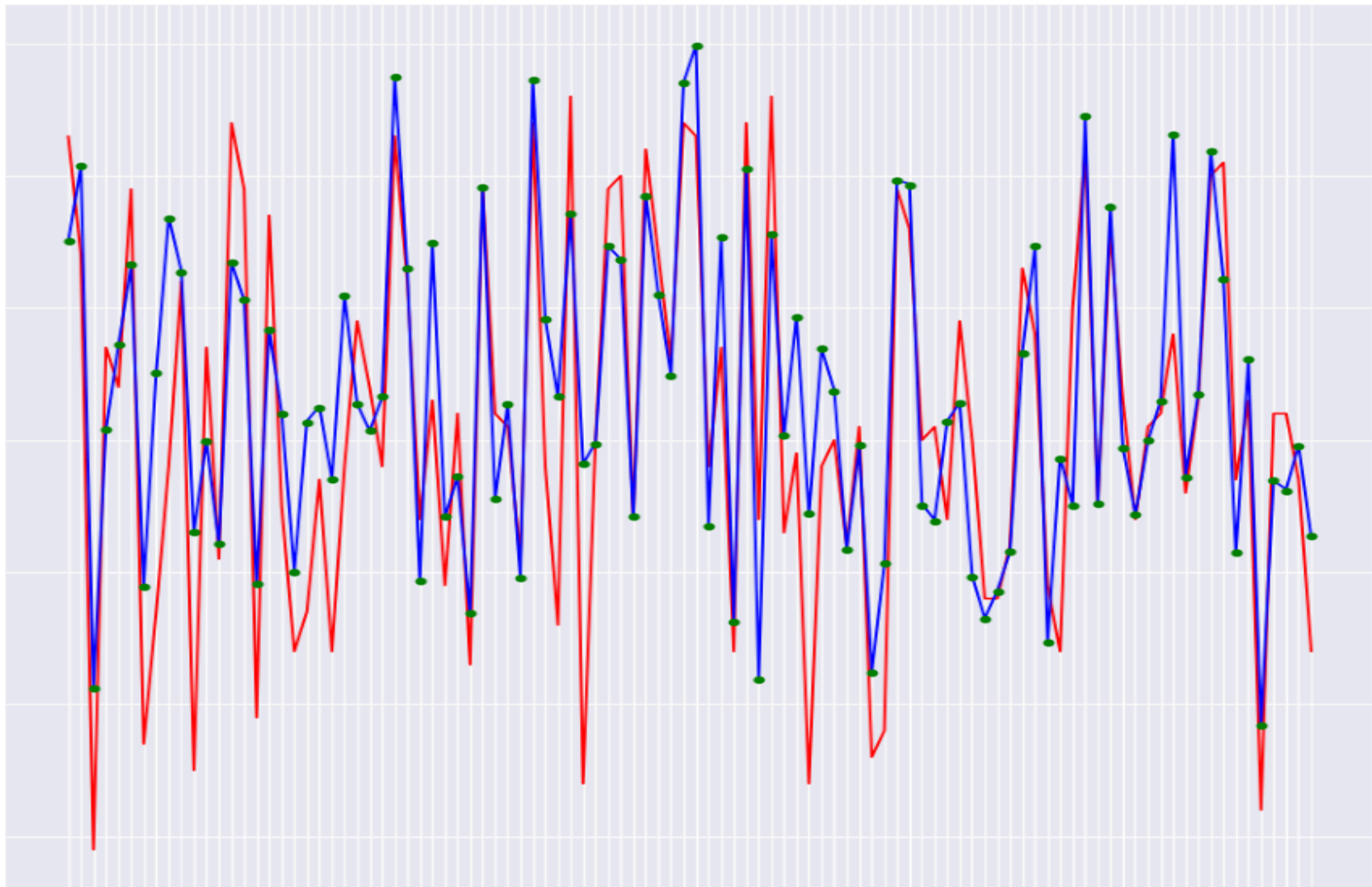
$$P(X_2 = x | p, r) = \binom{r+x-1}{x} p^r (1-p)^x.$$

- Poisson-gamma distribution
 - Unequal mean and Variance

https://www.johndcook.com/negative_binomial.pdf

Suppose $X | \Lambda$ is a Poisson random variable and Λ is a $\text{gamma}(\alpha, \beta)$ random variable. We create a new kind of random variable by starting with a Poisson but making it more variable by allowing the mean parameter to itself be random.

- Acceptance
- Prediction Negative Binomial
- Prediction Poisson



36 73 76 83 84 72 44 50 39 40 63 73 85 53 11 65 90 28 11 46 66 67 10 28 47 32 81 7 32 34 69 90 63 84 63 15 37 0 14 11 22 7 25 12 11 15 51 41 80 46 32 62 65 79 39 0 46 31 62 63 22 7 31 0 72 6 69 0 46 5 42 7 73 2 50 6 0 31 23 68 17 35 32 43 1 67 63 85

```


-----
ValueError                                Traceback (most recent call last)
<ipython-input-60-cf3f7a8ec689> in <module>()
----> 1 accuracy_score(y_test, np rint(y_nb)), mean_absolute_error(y_test, np rint(y_nb))
      2 accuracy_score(y_test, np rint(y_p)), mean_absolute_error(y_test, np rint(y_p))

/anaconda3/lib/python3.6/site-packages/sklearn/metrics/classification.py in accuracy_score(y_true, y_pred, normalize,
sample_weight)
    174
    175     # Compute accuracy for each possible representation
--> 176     y_type, y_true, y_pred = _check_targets(y_true, y_pred)
    177     if y_type.startswith('multilabel'):
    178         differing_labels = count_nonzero(y_true - y_pred, axis=1)

/anaconda3/lib/python3.6/site-packages/sklearn/metrics/classification.py in _check_targets(y_true, y_pred)
    79     if len(y_type) > 1:
    80         raise ValueError("Classification metrics can't handle a mix of {0} "
--> 81                            "and {1} targets".format(type_true, type_pred))
    82
    83     # We can't have more than one value on y_type => The set is no more needed

ValueError: Classification metrics can't handle a mix of continuous and binary targets

```

-
- 5 To whoever lands here: this answer is **plain wrong**; the cause of the error is the attempt to use accuracy as a metric in a *regression* setting (notice that OP's model is `LinearRegression` , *not* `LogisticRegression`), which is meaningless... – [desertrnaut](#) Jan 31 at 0:08 
-

Future Work

- Build a web interface
- Add more algorithms

References

Dataset

<https://www.kaggle.com/mohansacharya/graduate-admissions>

Mohan S Acharya, Asfia Armaan, Aneeta S Antony :

A Comparison of Regression Models for Prediction of Graduate Admissions, IEEE International Conference on Computational Intelligence in Data Science 2019

References

<https://en.wikipedia.org/wiki/XGBoost>

<https://github.com/dmlc/xgboost/tree/master/demo#machine-learning-challenge-winning-solutions>

https://en.wikipedia.org/wiki/Gradient_boosting

<http://datascience.la/xgboost-workshop-and-meetup-talk-with-tianqi-chen/>

<https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/>

<https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/>

https://www.johndcook.com/negative_binomial.pdf

Thank you!

Questions ?