Machine Learning Engineering Nanodegree

Capstone Proposal

National Football League

Krishna Bhatia

27th October,2017

Domain background

The book (and later a movie) **Moneyball** by Michael Lewis tells the story of how the USA baseball team Oakland Athletics in 2002 leveraged the power of data instead of relying on experts. The traditional way to select players was through scouting but Oakland and his general manager Billy Bean selected the players **based on their statistics** without any prejudice. Specifically, his assistant – the Harvard graduate Paul DePodesta looked at the data to find which ones were the undervalued skills.

This project outlines the method used in the book on the National Football League. I was amazed how different stats could be used to identify skilled players. Machine Learning uses the statistics to help the scouts to choose the best players for their teams.

Problem Statement

The problem is to determine which of the NFL players are best based on their stats. One way to solve this problem is to predict whether they will make the Probowl's or not. This would help the scouts make an informed decision on the selections.

Datasets and Inputs

After surfing the web a bit, I found a website http://www.pro-football-reference.com/ . This website contains data which is easily available and is machine readable. The data which I extracted from this site contains combine results of the players. The data is dirty, so would require some cleaning like dropping the duplicates, modifying feet-inches to just inches and most importantly cleaning the names of the players as the list comes with the names containing '+' or '*' to designate special status of the player.

The data contains all the information on the players such as their heights, weights, their position, 40 yard dash, vertical jump, bench press, broad jump, 3 cone drill and shuttle drill. Moreover, I added an additional column to the data, whether the players made it to probowl or not. This data would be extremely helpful in predicting the outcome.

Solution Statement

A solution to this problem would be predicting correctly whether the player made it to the probowl's or not. Keeping data in mind I would add a column to it which would be named as 'Probowl' and it would have values either "True" or "False". So, the task at hand would be to see whether the model is correctly able to predict this. Since it is clearly a problem of Supervised Learning, I would be using algorithm's like XGBoost and Decision Trees to train and test my model.
I would also be using Principal Component Analysis(PCA) to reduce the dimensionality of the data. But before applying the algorithm, I would analyze the data in order to find the feature which describes the data in the best way possible.

Benchmark

The benchmark to consider in this case will be accurately predicting the outcome whether the player made it to the probowl or not. I would measure how well does the algorithm makes the model fit and would determine this on the basis of F2 score.

Evaluation Metrics

The evaluation metric used for this model is F1 score and R^2 score.

The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. The relative contribution of precision and recall to the F1 score are equal. The formula for the F1 score is:

F1 = 2 * (precision * recall) / (precision + recall)

R^2 (coefficient of determination) regression score function. Best possible score is 1.0 and it can be negative (because the model can be arbitrarily worse). A constant model that always predicts the expected value of y, disregarding the input features, would get a R^2 score of 0.0.

Project Design

The first task at hand is to get the data. I would do web scrapping and since the data is directly taken from the website it would be dirty. Thus, I would need to perform some cleaning before I can start analyzing the data. Once the data has been cleaned, I would analyze the data and pre-process it. I would also split my data into training and testing set.

Having pre-processed the data, I would then try to apply different models. I will try multiple algorithms such as XGBoost and Decision Tress. I would also apply PCA in order to reduce the dimensionality of the data. As said earlier, the metrics used would be F1 score and R^2 score. I would first try to apply this algorithm to a small subset of the data and find out its performance. Then I would apply the algorithms to the whole data.