BETTING JUST GOT EASIER: THE POWER OF MACHINE LEARNING AND MAKING PREDICTIONS

Honors Thesis

Presented in Partial Fulfillment of the Requirements
For the Degree of Bachelor of Science in Computer Science

In the College of Arts and Sciences
at Salem State University

By

Johnson Ngandjui

Dr. Fatema Nafa
Faculty Advisor
Department of Computer Science

***

Commonwealth Honors Program
Salem State University
2021

Table of Contents

ABSTRACT

There comes a time in your life when you have endeavored to place a wager, whether minuscule or astronomically immense the goal is to victoriously triumph. What if you knew the chances of you winning? In this project, I analyzed The Big Five European soccer leagues data where I predict the probability of what team will win using various machine learning techniques while answering questions to maximize the accuracy of my prediction. The project drives away from the rigorous concepts of numbers, with a visual representation of the analytics. This breaks away from the extensive data into a more conceptualized aspect of betting. Many Bettors bet based on favorites, is that a valid way to place a bet? The first phase of this project is creating a descriptive analysis for understanding the data, the second phase is diving into support vector machines, random forest, and Xgboost to organize data elements and standardize how the data elements relate to one another to answer questions pertaining to wager making. I will make use of PySpark to show distinction between supervised learning models. The complex components will follow a sequential design metric to understand correctly how to maximize your bet. The results will consist of a prototype web application with a descriptive analysis of my findings, this includes betting prediction on my data. Users will get a deep understanding on why the results presented as they did.

1. INTRODUCTION

Like many others, I hate to lose, what is more frustrating than having certainty and it slowly being ripped away into shreds as time elapses. Learning has always been my passion, as I became deeply engaged in a new activity it exposes me to new knowledge, forcing me to read, start projects, set goals, and ask for help when I get stuck. I have always had a passion for numbers, data analytics and machine learning. The ability to almost predict the future by using the past has always fascinated me. Working with real life big data brings me goosebumps. There is so much hidden information one can find.

Every person nurtures an innate desire to feel accepted in the socio-economic circle, placing bets is one of the many ways of doing so. The word betting instantaneously scares people. why? I asked 20 individuals and the one thing they had in common is "the fear of losing". Hopelessness, resentment and loss is something no one should feel. Within this project, I will attempt to make individuals feel safer in where they place their bets by taking them through the process of breaking apart big data, extracting the necessity, and seeing the direct impact of your results with the use of machine learning.

Given the size of the global betting business, anyone with superior prediction techniques may certainly make money, whether through working for betting organizations, selling predictions to professional gamblers, or personal betting. Despite the monetary gain in betting, it goes deeper than that. Many researchers at academic institutions are researching learning model and optimization techniques to find the highest accuracy in predictions. According to *Kampakis*, "It is possible to predict the winner of English county twenty cricket games in almost two thirds of instances." That is nearly 66%.

In this thesis I make use of machine learning models to predict outcomes of the five major soccer leagues (England, Germany, Spain, Italy and France) for the 1995/96 to 2019/20 seasons. Feature selection includes recursive backward selection, forward selection, and chi-squared test. The selected features are used for 4 classification models: *support vector, Xgboost(gradient boosted decision trees), and random forest*. For optimization I used the principal component analysis. The best model will be chosen with comparison to other models. Each will be characterized with reasoning and direction for the project

2. RELATED WORK

Alfred's book on Research Methods provides a comprehensive analysis and methodology on data science research with IBM. This book provides an evaluation of how successful a project should perform. The guide this book offers is very systematic, pertaining mostly to large companies. Despite this large-scale project, Baker manages to keep the book for beginners and relatable no matter the level of experience. In the process of determining a "successful" data science project, Baker claims evaluation research is unlikely to result in program improvement. There are various forms of evaluation and finding what works for you is important. Depending on project type and requirements you can weigh your project success not on completion but on quality of implementation. Why are algorithms chosen? After taking an intuitive and informational approach, Vassant sees the value in the power of predictive models. It provides a new form of thinking in which he states, "computational thinking" [13]. The focus is shifted in the paper to the vacancy of many computational models. Using big companies like Google and IBM Vasant finds the downside within the field. With the amount of data we have, Vassant believes there

should be no need for “average” predictive models. The model that only does half the job. The incompetence of these models is then used to make real-life decisions and generate false statistics for the public. Vasant in his research captures the limitation of predictive modeling. Despite the unprecedented opportunity to develop predictive models, Vasant believes data science is not the answer all and poses many limitations. Many models on the surface may look different but have the same underlying structure. With correct implementation models, despite observational data's limits, Vassant says the size and scale of the data enables one to slice and dice it in a variety of forms without sacrificing sample size.

Despite the difference in dataset and the answers these researchers pose, they provide a solid backbone to the foundation of this project. Predictive Modeling has its limits, but what is the extent of these limits? How far can they be pushed? With the use of PySpark as a foundation to this learning metric, scaling an improving the “incompetence” of this models to something of a more adequate prediction model will prove the power of machine learning.

Big data has not reached its peak yet, in this research, Foster considers this era as Big data 1.0. This is just the ability to process large data. Big data 2.0, brought a shift in creative thinking [12]. With this being said, it should be easier to adapt models for prediction with some certainty

3. MATERIALS AND METHODS

This section presents the statement of problem, feature selection, machine learning algorithms (support vector, Xgboost(gradient boosted decision trees), and random forest), and optimization techniques. The methodologies used in this project include Supervised

learning. Supervised learning is used when the result is known prior for the model you are building on, winning, losing, or a tie. Even if one has a list of input variables, black box predictive models can be such complex functions of the variables that no human can comprehend how the variables are related to one another to arrive at a conclusion. The supervised learning models used in this project include random forest and gradient boost. The pro about these models is they generate multiple simple models, in this thesis their performances will be displayed.

3.1 Problem Statement

Given Big Five European soccer leagues data. Features related to soccer leagues data is found, the question is, can the probability of a team to win be predicted using various machine learning techniques? The concepts of numbers can be rigorous especially with a dataset with an unforeseen number of features. Breaking away from the extensive numbers and focusing on the why and less so the how. The main goal of the project is to take complex data and break it down into a comprehensible format so the impact of a predictive model can be shown, while making predictions and maximizing probability in wager making.

3.2 Feature Selection and Dimensionality Reduction

The process of selecting a subset of the most significant features to perform the model's prediction is known as feature selection[14].Dimensionality reduction is reducing the number of input variables for a predictive model. In this work, for feature selection recursive backward selection, forward selection, and chi-squared test were used. Principal

Component Analysis(PCA) has been used for dimensionality reduction[15]. in this section I will introduce each of these techniques

3.2.1 Vectorization

Vectorized query execution groups many rows in a columnar manner, and each operator iterates through data within a batch using simple loops. This feature significantly minimizes the amount of time the CPU is used for reading. We'll need to utilize a Vector Assembler to convert the characteristics into features because they're all numerical or discrete numeric. A vector assembler is a transformer that converts a group of features into a single vector column, also called an array of features. The use of columns is prominent in this area. Then, in Vector Assembler, use the transform() function to convert the input into a vector column called a feature.

3.2.2 Chi Square Selector

The purpose of this test is to determine whether a discrepancy between observed and expected data is due to chance or there is an actual correlation between the data. Degrees of freedom refer to the maximum number of independent values, or values having the possibility to fluctuate, in a dataset. It is necessary to calculate degrees of freedom when attempting to comprehend the significance and validity of a chi-square statistic. Because the data must first be vectorized.

We cannot use all the features we created; we must find those who have the most impact on the outcome. If the data contains more features compared to the number of instances, the trained model will not generalize to the new samples. Occam's Razor: Model explain ability decreases, when the input data has a lot of features, hence making it difficult to

interpret the model. It is not possible to have statistics before the game's outcome, it's required to generate some additional variables. game is played since the goal is predicting the outcome. Features that were accessible due to post game stats were removed. There are many ways of feature selection, but one offered by spark is the chi square selector. Chi-Square is a straightforward approach for selecting univariate features for classification. It does not consider the relationships between features. This is because it works best with categorical data, although it can also be utilized with data like this in some instances. Knowing these parameters, to select my features I also made use of backward selection. It is used to remove features with no correlation to the dependent prediction of output. The process of backward selection was ideal, it represented the relation of the data in all aspects. Since I chose the significance value of .05, it was a process of elimination, any p value greater than .05 was eliminated. a p-value of . 05 are considered on the borderline of statistical significance.

	coef	std err	t	P>|t|	[0.025	0.975]
x1	0.0247	0.006	4.092	0.000	0.013	0.037
x2	0.0054	0.001	6.773	0.000	0.004	0.007
x3	0.0032	0.001	4.047	0.000	0.002	0.005
x4	-0.1363	0.066	-2.072	0.039	-0.266	-0.007
x5	0.1434	0.085	1.693		-0.023	0.310
x6	0.3612	0.050	7.244		0.263	0.459

Ht_team1 dropped

x1	0.0247	0.006	4.070	0.000	0.013	0.037
x2	0.0052	0.001	6.584	0.000	0.004	0.007
x3	0.0031	0.001	3.938	0.000	0.002	0.005
x4	-0.0623	0.049	-1.263		-0.159	0.035
x5	0.3721	0.050	7.497		0.274	0.470

FT_Team1 dropped

x1	0.0233	0.006	3.903	0.000	0.012	0.035
x2	0.0048	0.001	6.582	0.000	0.003	0.006
x3	0.0027	0.001	3.750	0.000	0.001	0.004
x4	0.4046	0.042	9.530	0.000	0.321	0.488

Figure 1: Backward Selection

The 4 selected features displayed in **Figure 1** show a p value of 0.00 which is less than our .05 feature, can now be used for our model. This was done outside of PySpark. Since it does not offer set model properties when using the formula interface, to get around these the generic python imports were used without PySpark.

3.3 Machine Learning Algorithms

3.3.1 Gradient Boosting

The ensemble approach uses additive simple tree models that are gradually fitted to reduce residuals, and each cycle uses a random subsample of training data. Gradient boosting, like random forest, assembles an ensemble from a collection of simple individual models[1]. Business additive simple tree models that are sequentially fit to lower the residual of each step using a random subsample of training data. Gradient boosting business additive simple tree models that are sequentially fit to reduce the residual of each step where each step uses a random subsample of training data [1]. While random forest builds independent models in parallel, gradient boosting business additive simple tree models that are sequentially fit to reduce the residual of each step where each step uses a random subsample of training data. Each model iteration focuses on accurately forecasting scenarios where the prior model failed. It can be used for both purposes.

3.3.2 Random Forest

The results of several decision trees are combined in a random forest to provide a more accurate and dependable prediction[3]. Each tree makes a prediction, and the final prediction is based on a voting system, resulting in a collection of independent voting trees[1]. There's no need to be concerned about the lack of linearity in this scenario because, because we're using trees, random forests can manage and work with high-dimensional data better than generalized linear modeling techniques and can be utilized for both classification and regression.

3.3.3 Support Vector Machine(SVM)

The data points closest to the hyperplane are called support vectors, and these are the points in a data set that, if removed, would change the position of the dividing hyperplane[3]. Therefore, these can be considered important parts of the dataset. SVM performs advanced data manipulation according to the function define by the kernel and attempts to optimize the line between the data points based on the define classes.

3.4 Optimization Techniques

Principal Component

For a multivariate dataset, it summarizes the most relevant information and creates new variables called principal components, which are a linear combination of the originals[9]. When many independent features are integrated into primary components, principal component analysis is generally used to reduce the number of variables. The major components and their variable loadings can also be used to determine significant qualities

and the amount of variance explained by the independent dataset. We employ PCA in this paper to try to comprehend qualities on a global scale.

4. EXPERIMENT SETUP AND RESULTS

This section describes the experiment setup and obtains results. The experimental results show that when performing the study's proposed task, the techniques were effective in the prediction.

4.1 Data Set

This section discusses the benchmark dataset used in the experiments. An experiment was conducted on five major soccer leagues (England, Germany, Spain, Italy and France) for the 1995/96 to 2019/20 seasons. The data consist of 44269 rows with 15 features. 2019 England data was used, separating the data by country, and the entire data set[12].

4.2 Technical Tools

This section covers the technical tools needed to apply the categorization model to a new dataset. Python 3.8 was utilized as the programming language. Python offers numerous advantages, including the fact that it is simple to develop and read. Anyone with a little time on their hands can learn Python syntax. Python code is similar to English in that it allows students to concentrate on the end product. Python's expressiveness allows it to carry out even the most complex tasks. Python is free and open source. The general population can help improve the language by assisting and contributing. Python runs on a variety of operating systems, including Windows, Linux, and Unix. Python has a huge

standard library with many modules and functions[11]. Support vector machine, random forest, and Xgboost are the models that were used in this research.

PySpark is a Python library that allows you to communicate with Apache Spark. The PySpark shell, which sanctions users to interactively analyze data in a distributed environment, as well as the facility to develop Spark applications utilizing Python APIs. Most Spark technologies, such as Spark SQL, Data Frame, Streaming, Machine Learning, and Core, are supported by PySpark. Spark SQL includes programming abstraction, which is used in this project. [2] The data we utilize in this project may be classified as big data, so PySpark is a better option than pandas data frames in this scenario; it requires less effort on small datasets but can readily grow up.

4.3 Exploratory Analysis

This section presents the data preprocessing step used in this thesis. The section then illustrates the details of a statistical analysis along with specific description of the finding.

4.3.1 Data Pre-processing

There are pre-processing steps given to that dataset, raw data is hard to work with, it cannot be inserted in a model with the exception of producing evident results[1]. Preprocessing is the concept of manipulating hard to read data into usable data (dataframe). Before diving in let's look at the distribution of each feature in the used dataset as displayed in **Figure 2.**

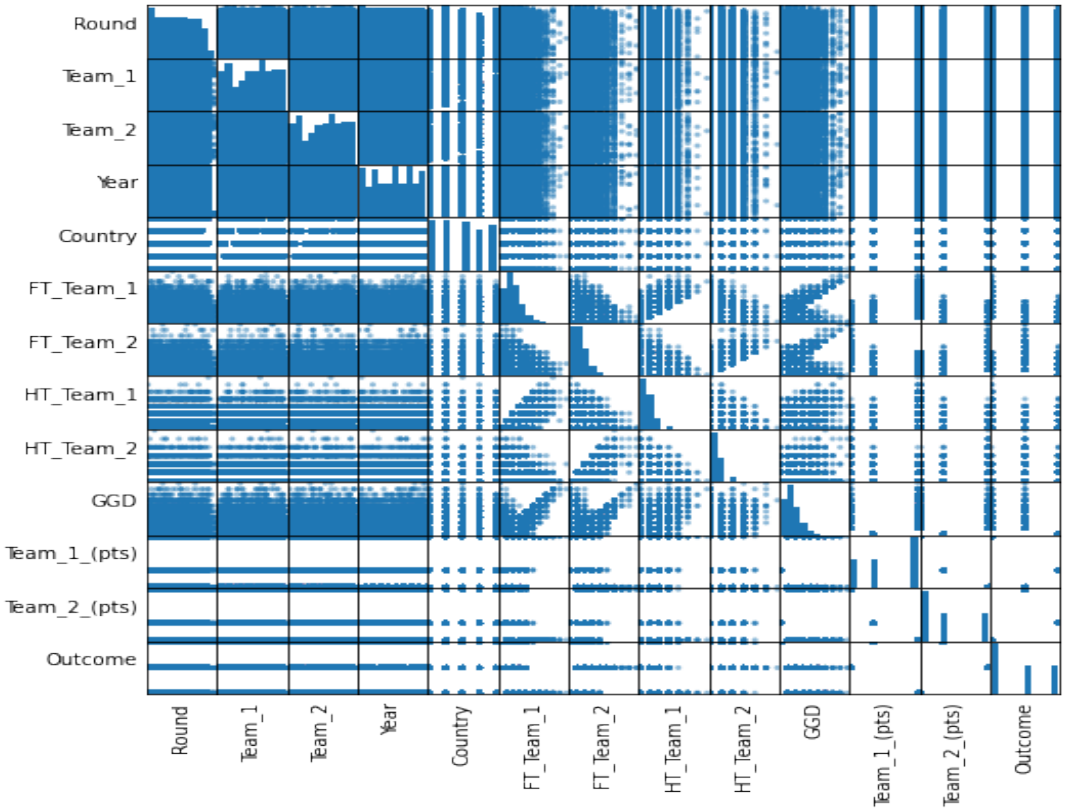


Figure 2: Scatter Matrix of Features Relation

Right away the scatter matrix[1] above shows us how each feature correlates. The majority of the majority of the relationship lies within FT_Team_1, FT_Team_2, HT_Team_1, HT_Team_2, Team_1, Team_2. The way these features correlate you can see a positive, negative or clusters in the scatter plots. The rest have zero correlation.

To start the preprocessing more features are added so we start with 25 features instead of 15 as shown in **Figure 2**. These features are created by analyzing a given feature. For example, to get second half goals we subtract fulltime and first half goals. Models work better with numerical values, so we label encode[2] date. This process involves splitting the date into month, day and day of the week, depending on how many unique values we

1 scatter plots used to visualize relationships between combinations of variables

2 For categorical variables, this is a widely used encoding approach. Each label is given a unique integer based on alphabetical order in this technique.

have in each of these partitions, they are given unique numerical values. For days of the week we have severe Mon – Sun, the labelencoder assigned values from 0-6. With all our features turned to numerical it is time to vectorize the data. By vectoring the data it is easier and faster for PySpark to iterate through. In this case all our data is put into an array, almost like a list.

After the data preprocessing, we can see just by using pythons build in describe function when two teams competed, considering all 5 countries together, on average the home team (FT Team 1 vs FT Team 2 wins the match by 1.5 to 1.1 and the average home team scores on average 1.7 points, while the away falls below at 1.1 points. Does this show a definite favorability of home field advantage? Let's dive deeper.[3] In a football season in all five leagues from 1995-2019, 25.65% of the matches have been a tie and the 27.86% of the matches have been a defeat for the home team.

4.3.2 Statistical Analysis

I collected data from 20 individuals who are soccer fanatics asking them what is a confident number of goals they think should guarantee a team victory in a soccer match? The average answer was 3.3 goals which I rounded to 3. Using this I was able to draw these statistics from my data:

- In 39.69% of the home victories 3 or more goals have been scored. In the 4.39% of the draws 3 or more goals have been scored and finally in the 36.21% of the away team victories 3 or more goals have been scored.

  This leads to the question, in a match 3 or more goals are scored, what is the probability that this is a home victory ?

3 Out of all the 5 countries from 1995-2019 the PSG appeared 2 times as they beat ES Troyes AC in 2015 away and EA Guingamp in 2018 at home with a goal difference of 9

- I found that in a match where 3 or more goals are scored, the probability to have a victory for the home team is: 49.43%.

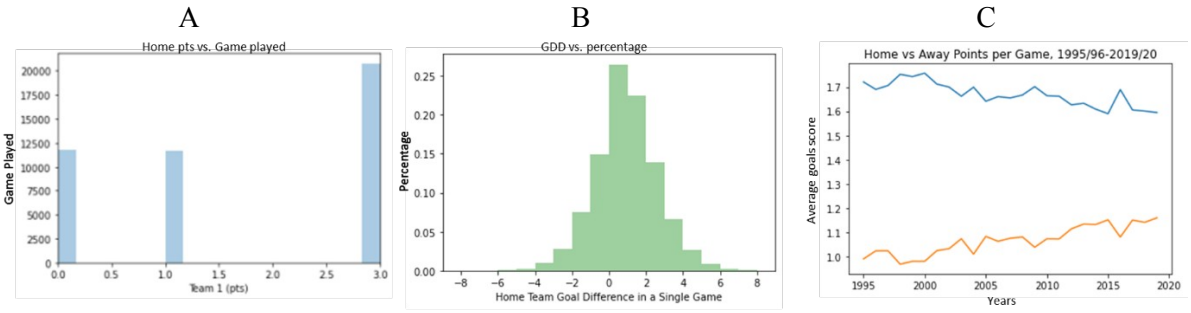


Figure 3: Home team points distribution

Figure 3 displays information about home team points distraction. It is clear from the figure that starting from a home advantage has an impact. Of the 45000 games played about half resulted in a win for the home giving the 3.0 points. The rest of the distribution is split almost evenly for losing (0 points) and a draw (1 point). Followed closely, in the plot displayed in b, the home team goal difference is portrayed. This is an effective way to see how many goals were conceded in the home team's victory and loss. In this plot, +1 represents a one goal win for the home team and -2 represents the away team winning by two goals, and the most common goal difference is zero which dictates a draw. Overtime home field advantages begin to degrade. Taking a close observational look at the trends in plot c, the representation of home is presented by blue and away is orange.The average points per game as the year increases is gradually decreasing. One can assume home field advantage will not be a thing in the years coming. From a betting standpoint , the probability a team wins given they are playing at home, has an impact on this dataset. Placing a wager on one intuition of home field advantage is not ideal, more digging needs to be done. This concept is referred to as Bayes' theorem. 'A' is an event of interest. As a commencement point, P(A) represents our prior credence: probability of

event A occurring. With incipient evidence B, the posterior credence or updated probability is represented P(A|B): probability of event A given evidence B has occurred [16]. Information retrieved as following: avg goals score when at home: 1.5302933673469388 and avg goals score when away: 1.13031462585034.

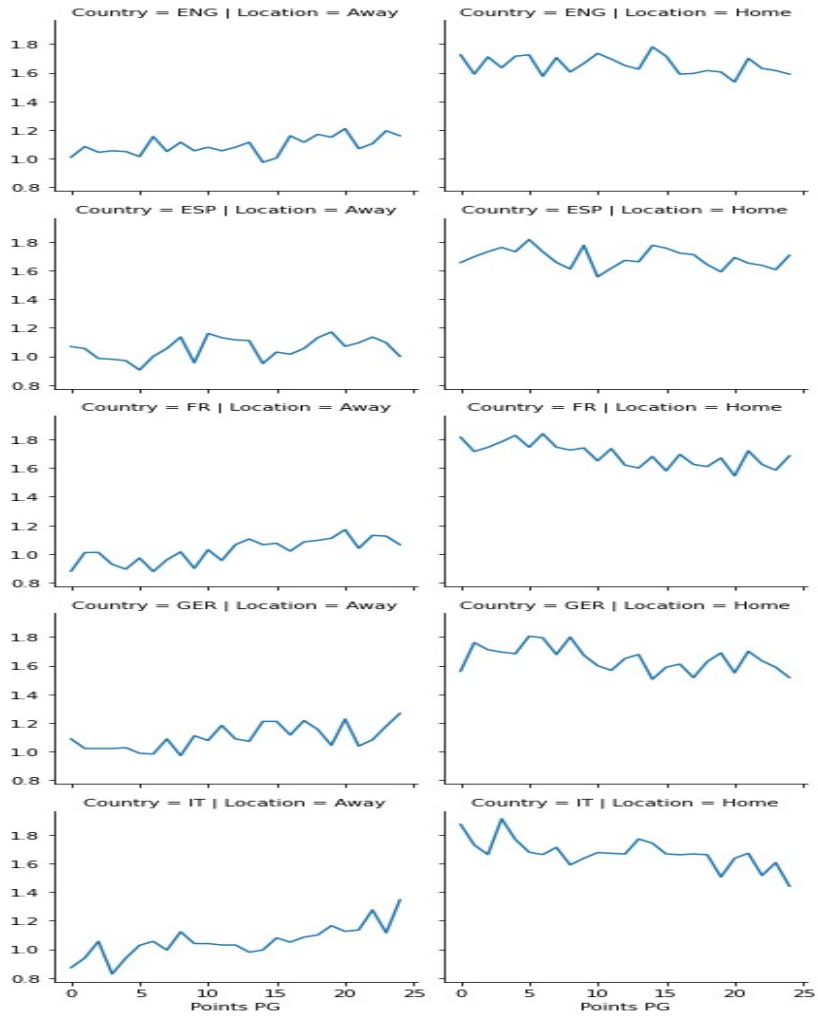


Figure 4: Average Goal Distribution by Country Home and Away

Each league has significantly different levels, but the reduction in home advantage looks to be present across all leagues. Most dramatically in Italy, and less so in England. This piece aimed to introduce one approach to this dataset. We started by creating new variables to show half-by-half and overall, In **Figure 4** shows the relation of average

goals of home and away by country. The evolution of home and away evolved over time and tried to unpack why these changes happened. We found that home advantage overall has dropped over time in each of the top 5 leagues. Additionally, when we split out home and away form by the level of the team, the drop in home performance is only seen among the poorer teams. The better teams in the league are maintaining their home level.

4.3.3 Finding of SVM, Random Forest, and Xgboost techniques

For this project SVM, Random Forest, and Xgboost have been used as the supervised learning models. Random forest classifier and Xgboost are both trees classifying models. The steps started using only the selected features, but this proved inconvenient. Random forest, gradient boost and many other tree defining models improves on bagging[4] because it decorrelates the trees with the introduction of splitting on a random subset of features. Even if we passed in all the features in our model by the end of the tree not all of them are being used. Bagging and random subs pacing prevent overfitting of the model which allows better predictions. The predictions in this thesis will not be biased to this sample. The random forest classifier takes trees as a parameter. It is hard to determine the number of trees needed to get a "perfect" model. Decision trees on their own tend to overfit the data. Limiting the maximum depth of a single decision tree helps reduce overfitting and for random forest; specifying the n number of trees helps as well, this is the maximum number of trees that will be created by the model. For the case of gradient boosting, we have seeds as a parameter which tells us the number of trees we should create before we combine them. For both Gradient boost and Random Forest, it was best if you can see visual and changing results as the prediction is made. I used spark to run and test my

4 a method for fitting several models to distinct subsets of a training dataset using an ensemble algorithm

model accuracy but also designed a user interaction streamlit page for visuals as in **Figure 5.**

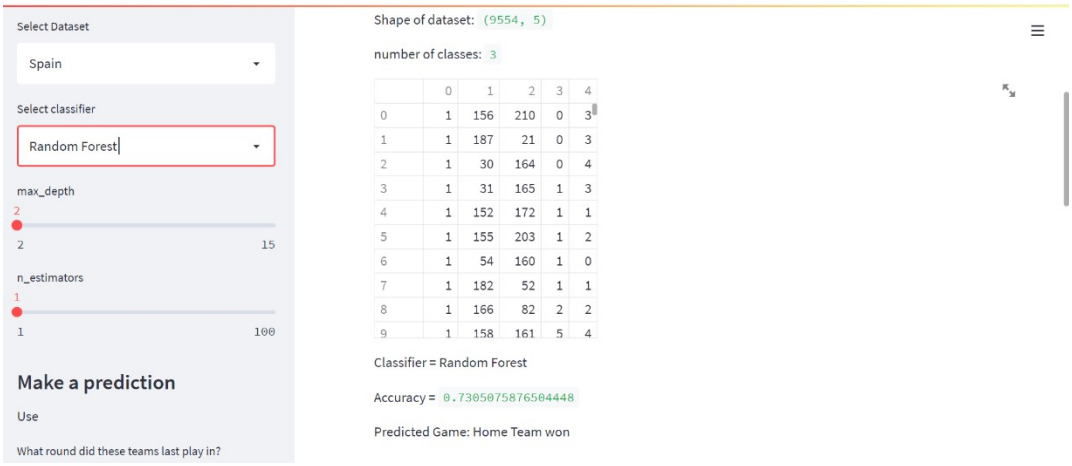


Figure 5: UI Design Model Testing.

SVM was a little hesitant to be used with this dataset because it is nonlinear and can cause a lot of issues. SVMs with non-linear kernels may learn complex decision boundaries in a higher-dimensional space, making them very strong. SVM uses kernels as an interchangeable parameter. The higher the kernel, the more dimensionality there is. SVMs, on the other hand, do not scale well as the number of data grows. The dimensionality of the model expands as the kernel is increased from the default value, and the model takes much longer to process.

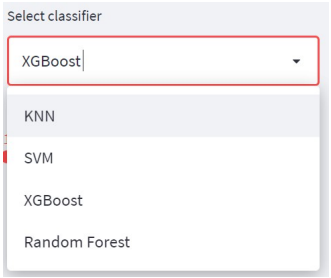


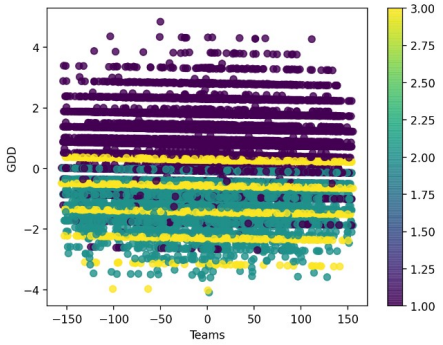
Figure 6: UI Design for Select Classifier.

5. DISCUSSION

In this study, the performing model was the Random Forest classifier. This is due to the hyper parameters that accompany this model. Because hyperparameter tuning is based on experimental findings rather than theory, the best technique for determining the optimal settings is to evaluate the performance of each model using a variety of different combinations. To find the highest accuracy for the black box algorithms, it trails an error base. Constantly running up the parameters because even decimal places influenced our model. Depicted below is a picture of the PCA, purple means the home team won, green the away team won and yellow shows a tie. The goal of this is to dimensionalize our results and scale so all the data points we are using fit on one graph. We can see the relationship where the home team wins more when there is a higher goal difference, most of the ties usually happen away.

Table 1: The accuracy for (Xgboost ,SVM, and Random Forest,)

Classifiers	Accuracy score
Xgboost (seeds=100)	0.891
SVM(kernel=3.0)	0.892
Random Forest (max depth = 10, n_estimator=100)	0.898



6. CONCLUSION

This experiment illustrates that, at least with current variables, football prediction remains a challenging endeavor, and that new variables are required to aid in the prediction of results. A machine learning system can already "think" about which side to bet on and outperforms others who are inexperienced with the sports predictions. When compared to the probability of a random guess, the prediction has a nearly 17 percent advantage. If you were to place a bet on the following soccer teams using the data set provided, you would look at who is playing at home and who is the favorite (the team in the season that has accumulated the most wins). Since not all the models were run on PySpark, it would be unfair to hand random forest the crown. PySpark has a significant advantage as it is faster for big data. In comparison to the python library Support vector machine is the better model. The downside of SVM is the increased dimensionality comes with the time constraint that comes with it. Predictions do not always go the way you want and so do machine learning models, there is a big increase in your beating if you have data(background knowledge) to question the “facts”.

7. FUTURE WORK

In the future, I recommend looking into and identifying new variables that might be useful, such as injuries or more facts about each team's players; perhaps FIFA data could help bring more information into the database. Spark is a growing library and using it to train Xgboost and SVM on a spark data frame can yield more consistent results and improve your forecast. For the future predicting goals scored per team would be ideal. This is more comprehensive; we must build two models for home and away. For

example, if the result was predicted as a home team victory, the next question would be how many goals were scored? Intuition and betting coexist. "Let's bet on it," is a phrase that many people utilize. What is the relationship between intuition and statistics? Perhaps this post will serve as a source of inspiration for future models that are better and more complicated. GitHub: https://github.com/Johnsonngandjui/capstone521-data-science-project

8. REFERENCES

1. R. Anden and E. Linstead, "Predicting eye movement and fixation patterns on scenic images using Machine Learning for Children with Autism Spectrum Disorder," 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2020, pp. 2563-2569, doi: 10.1109/BIBM49941.2020.9313278.

2. Apache, PySpark. “PySpark Documentation¶.” PySpark Documentation - PySpark 3.2.0 Documentation, https://spark.apache.org/docs/latest/api/python/.

3. Kampakis, Stylianos. “Using Machine Learning to Predict the Outcome of English County Twenty over Cricket Matches.” University College London, William Thomas, University College London, pp. 1–17.

4. Kempa, Matheus. “My_Udacity_Capstone/udacity_capstone_post.Pdf at Master ...” Githib.com, https://github.com/Matheuskempa/My_Udacity_Capstone/blob/master/UDACITY_CAPSTONE_POST.pdf.

5. R. Anden and E. Linstead, "Predicting eye movement and fixation patterns on scenic images using Machine Learning for Children with Autism Spectrum Disorder," 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2020, pp. 2563-2569, doi: 10.1109/BIBM49941.2020.9313278.

6. Raymond. Anden, "Predicting eye movement and fixation patterns on scenic images using Machine Learning for Children with Autism Spectrum Disorder," Ph.D. dissertation, Chapman University, Orange, CA, Year. https://doi.org/10.36837/chapman.000311

7. Rudin, C., & Radin, J. (2019). Why Are We Using Black Box Models in AI When We Don't Need To? A Lesson From An Explainable AI Competition. Harvard Data Science Review, 1(2). https://doi.org/10.1162/99608f92.5a8a3a3d

8. Stevefrench, JDA. "Measuring Home Advantage." Kaggle, Kaggle, 8 May 2020, https://www.kaggle.com/stevefrench/measuring-home-advantage.

9. Ringnér, M. (2008). What is principal component analysis?. Nature biotechnology, 26(3), 303-304.

10. Cohn, R., & Holm, E. (2021). Unsupervised machine learning via transfer learning and k-means clustering to classify materials image data. Integrating Materials and Manufacturing Innovation, 1-14.

11. Khoirom, S., Sonia, M., Laikhuram, B., Laishram, J., & Singh, T. D. (2020). Comparative analysis of Python and Java for beginners. *Int. Res. J. Eng. Technol*, *7*(8), 4384-4407.

12. Provost, Foster and Fawcett, Tom. Data Science and its Relationship to Big Data and Data-Driven Decision Making, 13 Feb 2013.

13. Dhar,Vasant. Data Science and Prediction. New York University, Leonard N. Stern, May 2012

14. Zhang, X., Fan, M., Wang, D., Zhou, P., & Tao, D. (2020). Top-k feature selection framework using robust 0-1 integer programming. IEEE Transactions on Neural Networks and Learning Systems.

15. Geladi, P., & Linderholm, J. (2020). Principal Component Analysis.

16. Zhu, S., Kim, Y., Zheng, J., Luo, J. Y., Qin, R., Wang, L., ... & Bi, X. (2020, April). Using Bayes' Theorem for Command Input: Principle, Models, and Applications. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (pp. 1-15).