

**“ICC 2020 Men's Cricket T20 World Cup Prediction”**

**Using**

**“Machine Learning”**





Anekant Education Society’s

# Tuljaram Chaturchand, College

# Of Arts, Commerce and Science, Baramati

# 413102

**A Project Report on**

**“ICC 2020 Men's Cricket T20 World Cup Prediction” using “Machine Learning”**

**The Project Report Submitted to**

**DEPARTMENT OF STATISTICS,**

**From MSc-II (2019-2020)**

**Under the guidance of**

**Prof. Kale K. C.**

**By**

Mr. Gaikwad Rupesh Nitin Mr. Bande Vishal Rajendra

Roll. No:-13866 Roll. No:-13868

**“ANEKANT EDUCATION SOCIETY”**

**Tuljaram Chaturchand College**

**Of Arts, Science and Commerce, Baramati**



CERTIFICATE

This is to certify that Mr. Gaikwad Rupesh Nitin, Mr. Bande Vishal Rajendra of M.Sc. Part- II(Statistics) has satisfactorily completed the project on **“ICC 2020 Men's Cricket T20 World Cup Prediction using Machine Learning”**,as per the syllabus laid by the Savitribai Phule Pune University, Pune during the academic year 2019-2020 under my supervision and successfully submitted the report.

Project Guide Examiner HOD of Statistics

Prof. K. C. Kale Dr. A. S. Jagtap

**Place: Baramati**

**Date**:

**ACKNOWLEDGMENT**

The project is an attempt by every student to put his efforts into achieving some objectives under a study and concluding with some valid results. This work is an outcome of many minds. On the successful completion of our project work, we take this opportunity to thank every person associated with it.

We are grateful to our head of department, **Dr. Avinash S. Jagtap** for his timely encouragement and support. Special thanks to our guide, **Prof. Kale K. C.** for her guidance and giving us this opportunity to work on this topic. We also thank all the teaching staff especially Prof. N. A. Jagtap and Prof. P. S. Zanjurne, for their valuable suggestions. We thank the nonteaching staff of our department for their co-operation.

Our thanks and appreciations also go to our family and friends without whom the project would not have been completed successfully.

## ABSTRACT

In Oct 2020, the ICC 2020 Men's T20 World Cup is the 7th edition of the Men’s World T20World Cup and is scheduled to be hosted by Australia. This project work is trying to predict the winner of the 7th version of the ICC T20 world cup using machine learning algorithms. Our chosen machine learning classifiers are namely the Random Forest Algorithm, the Support Vector Machine (SVM) algorithm & the K-Nearest Neighbour (KNN) algorithm and the data reduction algorithm will be presented. Additionally, the steps are taken to achieve the KNN, SVM & Random Forest classifications as applied to all datasets in detail. A classification algorithm will be defined in depth. It will also be mentioned how and why they apply to this project. The selected datasets required cleaning and cleansing and it is done to ensure that they are ready to have the classification algorithm applied to them. Finally, this project, the application of the KNN, SVM & Random Forest algorithms will be discussed in detail as they are applied to the datasets. This project work, the concepts of big data will be used to predict the winner of the ICC 2020 Men’s T20 Cricket World Cup.

**Index Terms:** ICC (International Cricket Council), CWC (Cricket World Cup), Cricket, T2

**Softwares:**

Minitab

MS-Word

MS-excel

Python

R

**MOTIVATION**

The main purpose of this study was to find out which teams will perform consistently in the entire competition. Also for which individual teams qualify for semi-finals, final and the winning of world T20 world cup 2020. Predicting the future sounds like magic whether it be detecting in advance the intent of a potential customer to purchase your product or figuring out where the price of a stock is headed. If we can reliably predict the future of something, then we own a massive advantage.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **TITLE** | **PAGE NO.** |
| 1. | INTRODUCTION | 8 |
| 2. | OBJECTIVE | 9 |
| 3. | DATA COLLECTION | 10 |
| 4. | METHODOLOGY | 11-12 |
| 5. | TERMINOLOGIES | 13-19 |
| 6. | ENVIRONMENT AND TOOLS | 20 |
| 7. | STATISTICALANALYSIS   1. Data cleaning and formatting. 2. Exploratory data analysis. 3. Feature engineering and selection. 4. Compare several machine learning models on a performance metric. 5. Perform hyper parameter tuning on the best model. 6. Evaluate the best model on the testing set. 7. Interpret the model results. 8. Draw conclusions | 21-29 |
| 8. | CONCLUSION | 30-31 |
| 9. | AREA OF FURTHER IMPROVEMENTS AND LIMITATIONS | 32 |
| 10. | REFERENCES | 33 |
| 11. | APPENDIX | 34 |

**INTRODUCTION**

Cricket is a well-known sport and with its increasing popularity and viewership, change of formats and innovations in the tournament played became necessary.A typical Twenty20 game is completed in about three hours, with each innings lasting around 75–90 minutes and a 10–20-minute interval. Each innings is played over 20 overs and each team has 11 players. This is much shorter than previously existing forms of the game, and is closer to the timespan of other popular team sports. It was introduced to create a fast-paced form of the game, which would be attractive to spectators at the ground and viewers on television.

To cater to potential future growth, global market research was commissioned by the International Cricket Council (ICC) which revealed that cricket has more than one billion fans worldwide, with the potential for significant growth. Since its inception the game has been very successful resulting in its spread around the cricket world.Among all formats of cricket, the popularity of Twenty20 Internationals (T20) was the highest with 92%, with 87% of the fans stating that they would like T20 to be included in the Olympic Games.On most international tours there is at least one Twenty20 match and all Test-playing nations have a domestic cup competition.

One of the International Cricket Council's (ICC) main objectives in recent times is to deliver real-time, interesting, storytelling stats to fans through the Cricket World Cup app or website. Players are what fans obsess most about so churning out information on each player's performance is a big priority for ICC and also for the channels broadcasting the matches.

Hence to solving an exciting problem such as determining the features of players and teams that determine the outcome of a T20 match would have considerable impact in the way cricket analytics is done today

The ICC 2020 Men's T20 World Cup is scheduled to be the 7th [ICC Men's T20 World Cup](https://en.wikipedia.org/wiki/ICC_Men's_T20_World_Cup) tournament, with matches to be held in Australia from 18 October to 15 November 2020. The tournament will be contested by 12 teams who will be playing in a two round-robin group, with the top two teams from group A & group B at the end of the group phase progressing to the semi-finals.

Predicting the future sounds like magic whether it be detecting in advance the intent of a potential customer to purchase your product or figuring out where the price of a stock is headed. If we can reliably predict the future of something, then we own a massive advantage. Machine learning has only served to amplify this magic and mystery.

**OBJECTIVES**

* To predict the winner of the ICC 2020 Men’s Cricket T-20 World Cup using “Machine Learning” based on the Team’s past performances.
* Predict the outcome of individual matches for the entire competition.
* Simulate the next matches i.e. semi-finals and finals.

**DATA COLLECTION**

In this study, our approach is to predict ICC 2020 Men’s T20 CWC matches based on past T20 matches results. Now, stronger teams like Pakistan, Australia, England, India, New Zealand, etc. would perform better and weaker teams like South Africa, West Indies, Sri Lanka, Afghanistan, Bangladesh would perish – we are not saying this – but our past T20 matches data study revealed the strong and weak team contender for T20 Men’s World Cup 2020.

Hence, we decided to study past T20 matches from 2012 to 2020. To collect datasets, we followed [How Stats](http://www.howstat.com/cricket/home.asp), ESPN Crickinfo and Cricbuzz websites.

For data collection, we extract, T20 matches year on year [since 2000] and stored the dataset in excel sheets. However, for our study we considered only T20 matched played from 2012 to 2020. Because we believe very old matches results [like the early 2000s] should not have a significant impact on team-wise performance for 2020 T20 CWC. Hence, we decided to study the latest team-wise performances.

**METHODOLOGY**

In this study, as of 31 December 2019, the top eight ranked ICC [Full Members](https://en.wikipedia.org/wiki/List_of_International_Cricket_Council_members#Full_Members) they are Pakistan, India, England, South Africa, New Zealand and West Indies, Afghanistan along with hosts Australia, qualified directly for the 2020 world cup tournament. Of those ten teams, the top eight ranked sides qualified for the Super 12s stage Tournameant [Sri Lanka](https://en.wikipedia.org/wiki/Sri_Lanka_national_cricket_team) and [Bangladesh](https://en.wikipedia.org/wiki/Bangladesh_national_cricket_team) did not qualify for the Super 12s, instead of being placed in the group stage of the competition. In the group stage competition, there are two groups (A & B) each group consist a 4-teams along with Sri Lanka and Bangladesh they are given in below table.

|  |  |
| --- | --- |
| **Group A** | **Group B** |
| Ireland | Bangladesh |
| Oman | Namibia |
| Papua New Guinea | Netherlands |
| Sri Lanka | Scotland |

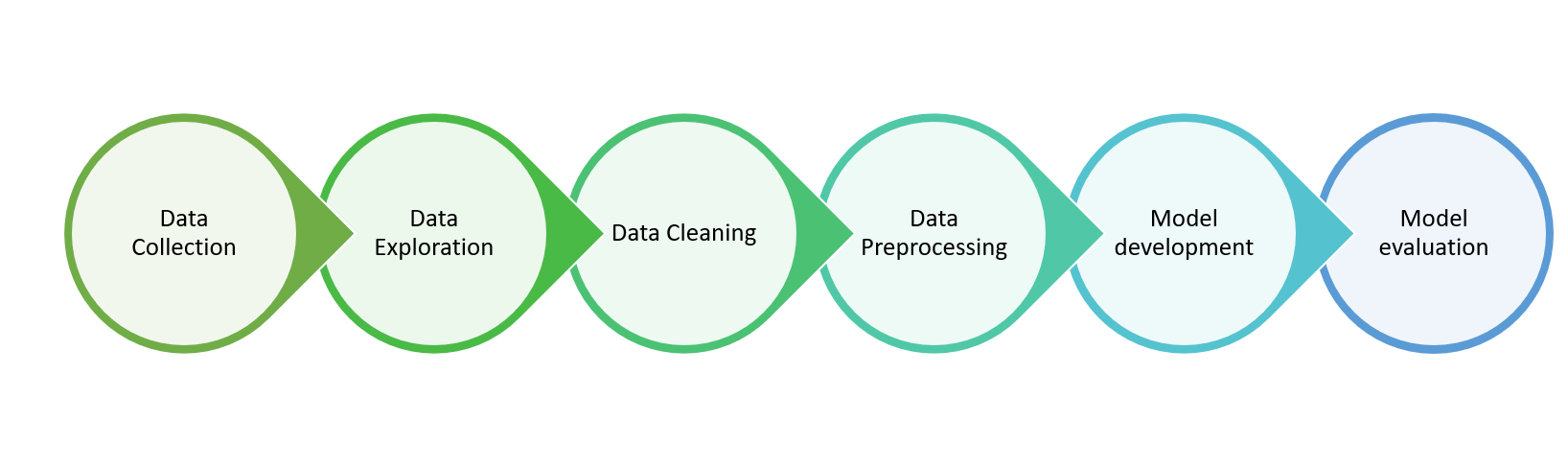
Our study starts when all teams for the super 12s stage competition are there, but now there are only 8 teams are qualified for the super 12s stage competition. So, the other 4 teams we select based on the there records with each other in that particular group. Since, Sri Lanka from group A & Bangladesh from group B both have a high percentage of winning with the other’s teams.

So, now we have 10 teams we ignored the other 2 teams. Since, there winning percentage is very poor with all the 10 teams which are selected for the super 12s stage competition. Hence, if we include them in the study there will be biasedness or in favor of 10 teams. So in our study instead of the super 12s, we only consider 10 teams for the entire competition & they are given in below table.

|  |  |
| --- | --- |
| **Group A Teams** | **Group B Teams** |
| West Indies | Afghanistan |
| Pakistan | England |
| Australia | India |
| New Zealand | South Africa |
| Sri Lanka | Bangladesh |

**We followed the general machine learning workflow step-by-step**

1. Data cleaning and formatting.
2. Exploratory data analysis.
3. Feature engineering and selection.
4. Compare several machine learning models on a performance metric.
5. Perform hyperparameter tuning on the best model.
6. Evaluate the best model on the testing set.
7. Interpret the model results.
8. Draw conclusions.



**TERMINOLOGIES**

# Random Forest:-

# “A model is like a pair of goggles. It puts certain things into focus.”

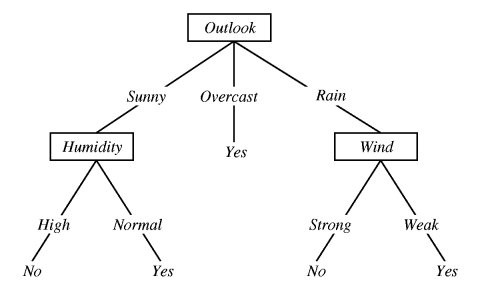
****

# What is the Random Forest?

Random forests are **bagged decision tree** models that split on a **subset of features** on each split. This is a huge mouthful, so let’s break this down by first looking at a single decision tree, then discussing bagged decision trees and finally introduce splitting on arandom subset of features.

**Decision Tree**

Essentially, a decision tree splits the data into smaller data groups based on the features of the data until we have a small enough set of data that only has data points under one label. Let’s look at an example. Below is a decision tree of whether one should play cricket.



In the example above, the decision tree is split on multiple features until we conclude “Yes”, we should play cricket, or “No” we should not play cricket. Follow the lines along the tree to determine the decision. For example, if the outlook is overcast, then “Yes” we should play cricket. If the outlook is sunny and humidity is high, then “No” we should not play cricket.

In a decision tree model, these splits are chosen according to a purity measure. That is, at each node, we want information gain to be maximized. For a regression problem, we consider the residual sum of square (RSS) and for a classification problem, we consider the Gini index or entropy.

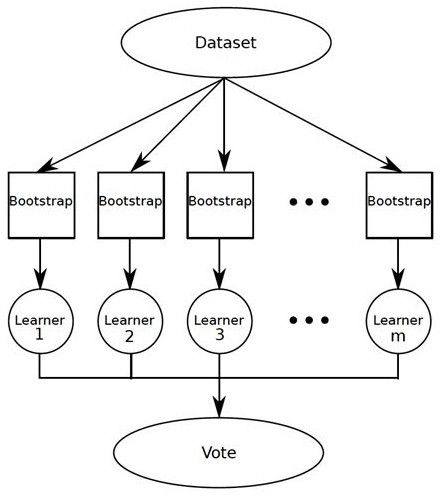
**Bagged Trees**

Now take the decision tree concept and let’s apply the principles of bootstrapping to create bagged trees.

**Bootstrapping** is a sampling technique in which we randomly sample with replacement from the data set.

Note: When bootstrapping, we use only about 2/3 of the data. Approximately 1/3 of the data (“out-of-bag” data) is not used in the model and can conveniently be used as a test set.

**Bagging**, or bootstrap aggregating, is where we create bagged trees by creating X number of decision trees that are trained on X bootstrapped training sets. The final predicted value is the average value of all our X decision trees. One single decision tree has high variance (tends to overfit), so bybagging or combining many weak learners into strong learners, we are averaging away the variance. It’s a majority vote!



**Random forest**

Random forest improves on bagging because it **de-correlates**the trees with the introduction of splitting on a**random subset of features**. This means that at each split of the tree, the model considers only a small subset of features rather than all of the features of the model. That is, from the set of available features n, a subset of m features (m=square root of n) is selected at random. This is important so that variance can be averaged away. Consider what would happen if the data set contains a few strong predictors. These predictors will consistently be chosen at the top level of the trees, so we will have very similar structured trees. In other words, the trees would be highly correlated.

So in summary of what was stated initially, random forests are bagged decision tree models that split on a subset of features on each split.

# ****Why is Random Forest So Cool?****

**Impressive in Versatility:**

Whether you have a regression or classification task, a random forest is an applicable model for your needs. It can handle binary features, categorical features, and numerical features. There is very little pre-processing that needs to be done. The data does not need to be rescaled or transformed.

**Parallelizable:**

They are parallelizable, meaning that we can split the process into multiple machines to run. This results in faster computation time. Boosted models are sequential in contrast, and would take longer to compute.

**Great with High dimensionality:**

Random forests are great with high dimensional data. Since, we are working with subsets of data.

**Quick Prediction/Training Speed :**

It is faster to train than decision trees because we are working only on a subset of features in this model, so we can easily work with hundreds of features. Prediction speed is significantly faster than training speed because we can save generated forests for future uses.

**Robust to Outliers and Non-linear Data :**

Random forest handles outliers by essentially binning them. It is also indifferent to non-linear features.

**Handles Unbalanced Data :**

It has methods for balancing error in class population unbalanced data sets. Random forest tries to minimize the overall error rate, so when we have an unbalanced data set, the larger class will get a low error rate while the smaller class will have a larger error rate.

**Low Bias, Moderate Variance :**

Each decision tree has a high variance, but low bias. But because we average all the trees in a random forest, we are averaging the variance as well so that we have low bias and moderate variance model.

# K-Nearest Neighbors (KNN)

The KNN algorithm assumes that similar things exist nearby. In other words, similar things are near to each other.



Notice in the image above that most of the time, similar data points are close to each other. The KNN algorithm hinges on this assumption being true enough for the algorithm to be useful. KNN captures the idea of similarity (sometimes called distance, closeness) with some mathematics we might have learned in our childhood— calculating the distance between points on a graph.

There are other ways of calculating distance, and one way might be preferable depending on the problem we are solving. However, the straight-line distance (also called the Euclidean distance) is a popular and familiar choice.

**Steps for the choice of K-value :**

1. As we decrease the value of K to 1, our predictions become less stable. Just think for a minute, imagine K=1 and we have a query point surrounded by several reds and one green (I’m thinking about the top left corner of the colored plot above), but the green is the single nearest neighbor. Reasonably, we would think the query point is most likely red, but because K=1, KNN incorrectly predicts that the query point is green.

1. Inversely, as we increase the value of K, our predictions become more stable due to majority voting / averaging, and thus, more likely to make more accurate predictions (up to a certain point). Eventually, we begin to witness an increasing number of errors. It is at this point we know we have pushed the value of K too far.
2. In cases where we are taking a majority vote (e.g. picking the mode in a classification problem) among labels, we usually make K an odd number to have a tiebreaker.

## Advantages :

1. The algorithm is simple and easy to implement.
2. There’s no need to build a model, tune several parameters, or make additional assumptions.
3. The algorithm is versatile. It can be used for classification, regression, and search (as we will see in the next section).

## Disadvantages :

1. The algorithm gets significantly slower as the number of examples and/or predictors/independent variables increase.
2. **Support Vector Machine**

A support vector machine is another simple algorithm that every machine learning expert should have in his/her arsenal. The support vector machine is highly preferred by many as it produces significant accuracy with less computation power. Support Vector Machine can be used for both regression and classification tasks. But, it is widely used in classification objectives.

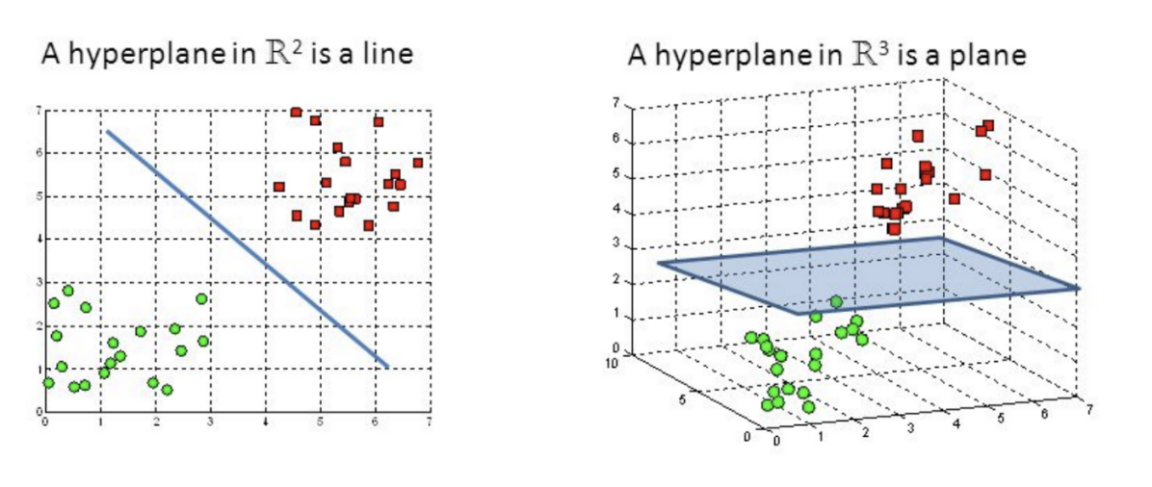
## What is Support Vector Machine?

The objective of the support vector machine algorithm is to find a hyperplane in N-dimensional space(N — the number of features) that distinctly classifies the data points.



To separate the two classes of data points, many possible hyperplanes could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

## Hyperplanes and Support Vectors :



Hyperplanes are decision boundaries that help classify the data points. Data points falling on either side of the hyperplane can be attributed to different classes. Also, the dimension of the hyperplane depends upon the number of features. If the number of input features is 2, then the hyperplane is just a line. If the number of input features is 3, then the hyperplane becomes a two-dimensional plane. It becomes difficult to imagine when the number of features exceeds 3.



Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier. Deleting the support vectors will change the position of the hyperplane. These are the points that help us build our SVM.

**ENVIRONMENT & TOOLS**

1. Jupyter Notebook
2. Numpy
3. Pandas
4. Seaborn
5. Matplotlib
6. Scikit-learn

**STATISTICAL ANALYSIS**

We followed the general machine learning workflow step-by-step:

We started by importing all the libraries and dependencies.

We loaded the Xls file containing the results of matches played between 2012 and 2020.

Out[63]:

| **#** | **date** | **Team\_1** | **Team\_2** | **Winner** | **Margin** | **Ground** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 2012-02-01 | Australia | India | Australia | 31 runs | ANZ Stadium |
| 2 | 2012-02-03 | Australia | India | India | 8 wickets | Melbourne Cricket Ground |
| 3 | 2012-02-11 | New Zealand | Zimbabwe | New Zealand | 7 wickets | Eden Park |
| 4 | 2012-02-14 | New Zealand | Zimbabwe | New Zealand | 5 wickets | Seddon Park |
| 5 | 2012-02-17 | New Zealand | South Africa | New Zealand | 6 wickets | Westpac Stadium |

Then we also loaded the Xls file containing the details of each team’s history in previous T20 WC’s.

The above pie chart shows % of each team’s appearances in previous T20 WC’s Semifinals.

Out[64]:

| **#** | **Team** | **Group** | **Appearances** | **Titles** | **Finals** | **Semifinals** | **Ranking** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | West Indies | A | 6 | 2 | 2 | 4 | 7 |
| 2 | Sri Lanka | A | 6 | 1 | 3 | 4 | 9 |
| 3 | India | B | 6 | 1 | 2 | 3 | 2 |
| 4 | Pakistan | A | 6 | 1 | 2 | 4 | 1 |
| 5 | Australia | A | 6 | 0 | 1 | 3 | 4 |

The above pie chart shows % of each team’s appearances in previous T20 WC’s Finals.

**1. Data cleaning and formatting**

Next, let’s display the details of matches played by India.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **date** | **Team\_1** | **Team\_2** | **Winner** | **Margin** | **Ground** |
| 1 | 2012-02-01 | Australia | India | Australia | 31 runs | ANZ Stadium |
| 2 | 2012-02-03 | Australia | India | India | 8 wickets | Melbourne |
| 3 | 2012-03-30 | South Africa | India | South Africa | 11 runs | Wanderers |
| 4 | 2012-08-07 | Sri Lanka | India | India | 39 runs | Pallekele |
| 5 | 2012-09-08 | India | New Zealand | No Result | NaN | Dr. YS Rajasekhara Reddy |

We continued by creating a column to display the details of matches played in 2012 and taking it as a reference for future work.

|  |  |
| --- | --- |
| Out[67] |  |
| date | 38 |
| Team\_1 | 38 |
| Team\_2 | 38 |
| Winner | 38 |
| Margin | 35 |
| Ground | 38 |
| dtype: | int64 |

**2. Exploratory data analysis**

After that, We merged the details of the teams participating this year with their past results.We deleted the columns like the date of the match, the margin of victory, and the ground on which the match was played. These features don’t look important for our prediction.

Out[69]:

| **#** | **Team\_1** | **Team\_2** | **Winner** |
| --- | --- | --- | --- |
| 1 | New Zealand | South Africa | New Zealand |
| 2 | New Zealand | South Africa | South Africa |
| 3 | New Zealand | South Africa | South Africa |
| 4 | England | South Africa | South Africa |
| 5 | England | South Africa | England |

**3. Feature engineering and selection**

This is probably the most important part of the machine learning workflow. Since, the algorithm is dependent on how we feed data into it, feature engineering should be given topmost priority for every machine learning project.

Feature engineering is the process of transforming raw data into features that better represent the underlying problem to the predictive models, resulting in improved model accuracy on unseen data.

**Advantages of feature engineering :**

* **Reduces Overfitting**: Less redundant data means less opportunity to make decisions based on noise.
* **Improves Accuracy**: Less misleading data means modeling accuracy improves.
* **Reduces Training Time**: fewer data points reduce algorithm complexity and algorithms train faster.

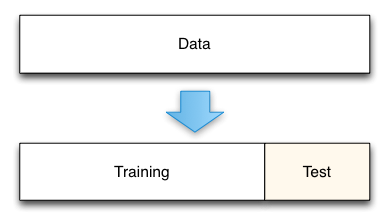
So continuing with the work, we created the model. If team-1 won the match, we assigned it to label 1, else if team-2 won, I assigned it to label 2.

Out[70]:

| **#** | **Team\_1** | **Team\_2** | **Winner** |
| --- | --- | --- | --- |
| 0 | New Zealand | South Africa | New Zealand |
| 1 | New Zealand | South Africa | South Africa |
| 2 | New Zealand | South Africa | South Africa |
| 3 | England | South Africa | South Africa |
| 4 | England | South Africa | England |

Then we converted team-1 and team-2 from categorical variables to continuous inputs using pandas function *pd.get\_dummies.*This variable has only two answer choices: team 1 and team 2. It creates a new data frame that consists of zeros and ones. The data frame will have one depending on the team of a particular game in this case.

Also, we separated training and test sets with 80% and 20% in training and validation sets respectively.



Train/Test Split

**4. Compare several machine learning models on a performance metric**

We used Support Vector Machines, Random Forests and K Nearest Neighbors for training the models we get following results:

|  |  |  |
| --- | --- | --- |
| **Algorithm** | **Training Set Accuracy (%)** | **Test Set Accuracy (%)** |
| **K-Nearest Neighbour (KNN)** | **56.70** | **56.70** |
| **Support Vector Machine(SVM)** | **60** | **60** |
| **Random Forest** | **60** | **62.5** |

The random forest combines hundreds or thousands of decision trees, trains each one on a slightly different set of the observations, splitting nodes in each tree considering a limited number of the features. The final predictions of the random forest are made by averaging the predictions of each tree.

RFs train each tree independently, using a random sample of the data. This randomness helps to make the model more robust than a single decision tree, and less likely to overfit on the training data.

**5. Perform hyperparameter tuning on the best model**

Training set accuracy: 0.600  
The test set accuracy: 0.625

**The popularity of the Random Forest model is explained by its various advantages:**

* Accurate and efficient when running on large databases.
* Multiple trees reduce the variance and bias of a smaller set or single tree.
* Resistant to overfitting.
* Can handle thousands of input variables without variable deletion.
* Can estimate what variables are important in classification.
* Provides effective methods for estimating m1issing data.
* Maintains accuracy when a large proportion of the data is missing.

**6. Evaluate the best model on the testing set**

Let’s continue, We added ICC T20 rankings of teams giving priority to the higher-ranked team to win this year.

The above line plot shows the current ICC rank of an individual’s teams by there past performance.

Next, We added new columns with the ranking position for each team and slicing the dataset for the first 20 games since there are 20 league stage games in total.

Out[77]:

|  | **Round** | **1st Position** | **2nd Position** | **Date** | **Location** | **Team\_1** | **Team\_2** | **Group** | **Result** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 15 | 1 | 4.0 | 9.0 | 2020-11-05 | Adelaide Oval, Adelaide | India | Bangladesh | B | NaN |
| 16 | 1 | 2.0 | 6.0 | 2020-11-06 | Melbourne Cricket Ground, Melbourne | Australia | New Zealand | A | NaN |
| 17 | 1 | 10.0 | 8.0 | 2020-11-07 | Melbourne Cricket Ground, Melbourne | West Indies | Sri Lanka | A | NaN |
| 18 | 1 | 5.0 | 9.0 | 2020-11-08 | Sydney Cricket Ground, Sydney | South Africa | Bangladesh | B | NaN |
| 19 | 1 | 7.0 | 4.0 | 2020-11-08 | Sydney Cricket Ground, Sydney | Afghanistan | India | B | NaN |

Then we added teams to a new prediction dataset based on the ranking position of each team.

Out[78]:

|  | **Team\_1** | **Team\_2** | **winning\_team** |
| --- | --- | --- | --- |
| 0 | Pakistan | Australia | None |
| 1 | India | South Africa | None |
| 2 | New Zealand | West Indies | None |
| 3 | England | Bangladesh | None |
| 4 | Australia | West Indies | None |

After that, we added scripts for getting dummy variables and added missing columns compared to the model training dataset.

Out[79]:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Afghanistan** | **Australia** | **Bangladesh** | **England** | **India** | **Netherlands** | **New Zealand** | **Pakistan** | **Sri Lanka** | **South Africa** |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**7. Interpret the model results**

Finally, we getting the results for every league stage match.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Match No. | Team | Against | Team | Winner |
| 1 | Australia | vs | Pakistan | Pakistan |
| 2 | South Africa | vs | India | India |
| 3 | West Indies | vs | New Zealand | New Zealand |
| 4 | Bangladesh | vs | England | England |
| 5 | West Indies | vs | Australia | Australia |
| 6 | Bangladesh | vs | Afghanistan | Afghanistan |
| 7 | Sri Lanka | vs | Pakistan | Pakistan |
| 8 | South Africa | vs | England | England |
| 9 | New Zealand | vs | Pakistan | Pakistan |
| 10 | Sri Lanka | vs | Australia | Australia |
| 11 | Afghanistan | vs | South Africa | South Africa |
| 12 | India | vs | England | England |
| 13 | Sri Lanka | vs | New Zealand | New Zealand |
| 14 | West Indies | vs | Pakistan | Pakistan |
| 15 | Afghanistan | vs | England | England |
| 16 | Bangladesh | vs | India | India |
| 17 | New Zealand | vs | Australia | Australia |
| 18 | West Indies | vs | Sri Lanka | Sri Lanka |
| 19 | Bangladesh | vs | South Africa | South Africa |
| 20 | Afghanistan | vs | India | India |

So the four teams to march to the semi-finals are

|  |
| --- |
| Australia |
| Pakistan |
| India |
| England |

We ran the function for semi-finals prediction we getting results for semi-final matches as,

|  |  |  |  |
| --- | --- | --- | --- |
| Team | Against | Team | Winner |
| Australia | vs | Pakistan | Pakistan |
| India | vs | England | England |

Hence the two finalists are **Pakistan** and **England** which is quite evident as they are considered the favorites to win this year.

And then we created a function to repeat the above work. This is the final function to predict the winner of the ICC 2020 Men’s Cricket World Cup.

**8. Draw conclusion**

Finally, on running the main function.

|  |
| --- |
| Pakistan Vs England |
| Winner: Pakistan |

**According to this model, Pakistan is likely to win this World T20 World Cup 2020.**

**CONCLUSION**

1. Based on team’s individual past performances team Pakistan will be the winner of ICC 2020 Men’s T20 cricket world cup.
2. We get the outcomes of individual matches for every league stage competition will be given below,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Match No. | Team | Against | Team | Winner |
| 1 | Australia | vs | Pakistan | Pakistan |
| 2 | South Africa | vs | India | India |
| 3 | West Indies | vs | New Zealand | New Zealand |
| 4 | Bangladesh | vs | England | England |
| 5 | West Indies | vs | Australia | Australia |
| 6 | Bangladesh | vs | Afghanistan | Afghanistan |
| 7 | Sri Lanka | vs | Pakistan | Pakistan |
| 8 | South Africa | vs | England | England |
| 9 | New Zealand | vs | Pakistan | Pakistan |
| 10 | Sri Lanka | vs | Australia | Australia |
| 11 | Afghanistan | vs | South Africa | South Africa |
| 12 | India | vs | England | England |
| 13 | Sri Lanka | vs | New Zealand | New Zealand |
| 14 | West Indies | vs | Pakistan | Pakistan |
| 15 | Afghanistan | vs | England | England |
| 16 | Bangladesh | vs | India | India |
| 17 | New Zealand | vs | Australia | Australia |
| 18 | West Indies | vs | Sri Lanka | Sri Lanka |
| 19 | Bangladesh | vs | South Africa | South Africa |
| 20 | Afghanistan | vs | India | India |

1. Based on Team’s outcomes of individual matches for every league stage competition

we will get teams for semi-finals will be,

|  |
| --- |
| Australia |
| Pakistan |
| India |
| England |

And teams for final will be,

|  |
| --- |
| Pakistan & England |

**AREAS OF FURTHER IMPROVEMENT**

1. Trying more complex Machine Learning algorithms like XG-Boost and fine-tuning the hyper parameters.
2. A confusion matrix would be great to analyze which games the model got wrong.
3. We could ensemble that is we could try stacking more models together to improve the accuracy.
4. Going even further and making a model based on player statistics.

**LIMITATIONS**

The main limitation in carrying out this project was the limited dataset, which we had at our disposal. The next logical step in the direction to improve the accuracy of prediction problem at hand would be to test out the approaches and various methodologies proposed in this project using a larger and more representative dataset. Also we would like to extend the candidate classifier set considered to a more exhaustive list and compare the performances among them.

**REFERENCES**

1. <https://arxiv.org/abs/1806.03208>
2. <https://github.com/abhinavsagar/ICC-2019-WC-prediction>
3. <https://medium.com/analytics-vidhya/the-love-of-cricket-45b9a7d09d07>
4. Why Random Forest is My Favorite Machine Learning Model, <https://towardsdatascience.com>
5. [A machine learning framework for sports result prediction, www.sciencedirect.com](http://A machine learning framework for sports result prediction, www.sciencedirect.com)
6. [Why Did Artificial Intelligence Fail in the FIFA World Cup 2018?](https://medium.com/futuristone/artificial-intelligence-failed-in-world-cup-2018-6af10602206a" \t "https://towardsdatascience.com/_blank)

[Here is what we learned from the AI’s failure to predict the results of the FIFA World Cup 2018 Russia.](https://medium.com/futuristone/artificial-intelligence-failed-in-world-cup-2018-6af10602206a" \t "https://towardsdatascience.com/_blank)  <https://medium.com>

1. A. Aburas, Machine Learning Algorithms for Big Data Project, Durban: University of Kwa-Zulu Natal, 2018

**APPENDIX**

1.Coding link <https://drive.google.com/file/d/1tMVX-z1S3L5Oao6Ayiv70j2lWu3Z3ewH/view?usp=sharing>

2 .Ranking link

<https://drive.google.com/file/d/1I9uvFhSkPaMsDzwOqSLGUu49hG-b-Qsv/view?usp=sharing>

3 .World\_cup\_group link

<https://drive.google.com/file/d/1DwNqsYSNsBFAoBTIropS8IO28h69WjAo/view?usp=sharing>

4 .Prev\_results\_clear link

<https://drive.google.com/file/d/1xN5Li1LLCVmVuban7m0j8b22dzTnpWB2/view?usp=sharing>

5 .Fixtures link

<https://drive.google.com/file/d/1d5FoORdc5mDwweWcoDvMwQ2QxnSTcgQi/view?usp=sharing>

6 .Percentage of winning link

<https://drive.google.com/file/d/17yztZYa6adb6nf9aXH8y7mhS7vmwHT79/view?usp=sharing>