Predicting UFC Fight Outcomes by Implementing Machine Learning Algorithms

Edbert Felix Fangasadha  
*Computer Science Department  
Bina Nusantara University*Jakarta, Indonesia 11480  
edbert.fangasadha@binus.ac.id

Steffi Soeroredjo  
*Computer Science Department  
Bina Nusantara University*Jakarta, Indonesia 11480  
steffi003@binus.ac.id

Cindy Amanda Onggirawan  
*Computer Science Department  
Bina Nusantara University*Jakarta, Indonesia 11480  
cindy.onggirawan@binus.ac.id

***Abstract— Mixed martial arts (MMA) has developed very rapidly in the last 25 years and has attracted the attention of fans for its erratic chances and victories. Predicting who will win a match in sport is tricky, because there are many external factors, not only the body physique, strategy, and skill set of a reliable fighter. In this paper, we develop a model to predict the winner of the future, from the past UFC fights dataset, by comparing the accuracy values ​​of using several machine learning algorithms, which are Logistic Regression, K-Nearest Neighbor, Support Vector Machine, and Decision Tree Classifier. After setting up the model, we emphasize that the use of hyperparameters is highly recommended for future experiments because of the large number of parameters could cause the model's accuracy to decrease drastically.***

***Keywords— MMA, UFC Prediction, Machine Learning, Logistic Regression, K-Nearest Neighbor, Support Vector Machine, Decision Tree Classifier***

# Introduction

Mixed martial arts (MMA) is a full-contact fighting sport in which fighters utilize techniques from a range of combat sports and martial arts to attack and wrestle one another while standing or on the ground [1]. This includes techniques to immobilize opponents such as boxing, muay thai, karate, taekwondo, judo, jujitsu, and wrestling. Those presented examples of legal fighting techniques in MMA prove that it is essentially an unarmed fight competition with its own set of rules enforced for safety measurements [5]. As of now, the world’s biggest fight publicity in mixed martial arts  is the UFC or Ultimate Fighting Championship. Between 2013 and 2017 alone, the promotion had presented over 1400 fights and counting with an event being held bi-monthly and having multiple fights per event [2].

People adore MMA because the best fighters can't be avoided. If a fighter triumphs, he or she will be compelled to face the strongest challengers to show that they are capable of competing in that bout. So, it was impossible for a champion to suddenly go up against a new challenger who clearly had little skill or experience, even though the weight class was the same. As a result, the fights are becoming more competitive and spectators are more intrigued in seeing the titles change hands so frequently. Moreover, the UFC only has eight heavyweights (plus four women's divisions) [6]. Fewer divisions means a bigger weight difference between them.

The final outcome of the matches, comparable to many other sporting events, is typically used to appraise the performance and effectiveness of the fighters. Considering the significance of the rounds, it doesn’t come out as a surprise that many enthusiasts try to predict how the fight will end, whether it may be as simple as a key to many stakeholders to place their odds on or to give coaches the probable risks of losing when taking up fight requests.

Predicting the outcomes of matches in sporting events has proven to be a difficult but fascinating undertaking. As a result, there has been a recent boom in the use of various approaches to improve scientific research prediction findings. A substantial amount of relevant data, in either structured or unstructured form, is a prerequisite for developing a good prediction model [7]. Furthermore, a thorough understanding of the observed process is required to comprehend the predicting process, how past events influence future events, and the causes and effects of certain process actions [7].

In this paper, we integrated and evaluated the accuracy of different machine learning algorithms in order to determine which approach is the most suitable for predicting the outcome of the two fighters’ bouts before they fight in UFC matches. Ideally, we aim to predict a fighter's performance against their opponents by utilizing four machine learning methods to map their performance on a dataset., namely Logistic Regression, K-Nearest Neighbor (KNN), Support Vector Machine (SVM) and Decision Tree Classifier. The Logistic Regression, KNN, and SVM models are utilized to categorize existing data in this project, additionally linear regression is specifically used to predict each fighter’s reach by using their height. On the other hand, the Decision Tree Classifier is used to combine prediction results from the three models stated previously.

The remaining part of the paper is structured out as follows: Section 2 presents brief descriptions of the associated work. The data set and proposed model utilized in this paper are described in section 3. Section 4 contains statistical models along the result analysis summary and at the last part in section 5, the paper is concluded

# Literature Review

We were able to find a few articles relevant to this subject because this research seeks to predict the outcomes of UFC bouts based on prior fight statistics and the physical characteristics of each fighter:

* Reference [2] retrieved each fighter's current figures and deducted their per-fight statistics starting from the year 2013, where the fighters had fought less than 10 times each year, to create a time-dependent dataset by using: Random forests, which make a clustered prediction using a group of trained decision trees; Decision trees that logically incorporate a series of simple tests; SVM for classification and regression; KNN to designate marks in their input set to the dominant class between its nearest neighbors; Naive Bayes as a probabilistic classifier that has strong independence assumptions; Perceptron could be considered as building blocks in a single layer of a neural network, and finally Stochastic Gradient Descent (SGD) used to minimize an objective function. Their model can be used to generate matchups in which both boxers have an equal chance of winning as to arrange more exciting fights alongside with balancing out the odds for fighters before each matches begin.
* Reference [3] compared logistic regression, SVM, random forest, and neural network performance to predict the result of the match and determine who to bet or not bet at all in a specific match. Here are advantages and disadvantages to each prediction model: Although logistic regression is accurate, the training model is slow and the variance is minimal. SVM RBF and polynomial models took a long time to train and were prone to overfitting. However, the SVM linear terms model only uses first-order linear weighting of the feature set, whereas random forest allows us to train the model quicker and fine-tune hyper-parameters to improve prediction accuracy. While the authors originally believed that the neural network could be useful, the model's accuracy was shown to be lower than that of other models. Consequently, the author decided to leverage the random forest model output and create a single-shot decision problem for each tennis match in order to maximize projected earnings.
* Reference [5] employed a multilayer neural network-based model to estimate the outcomes of UFC matchups based on the fighter's available information from previous matches. As it is a binary class problem with a winner of either red or blue, the authors of this study took the number of nodes in the input layer and their output layer has one node. When their model's input is collected, the weighted sum including bias is measured at a neuron, and a predetermined activation function is utilized to evaluate if the neuron should just be fired or activated. The neuron will then transmit those simulated results as inputs to the next layer of neurons until the output value enters the output layer.
* Reference [7] used team-related features and binary classification to compare which algorithm, validation method, and data preparation method produces better results in predicting basketball game outcomes. In prediction results by using disjoint dataset and train & test validation method, the best model prediction is obtained by KNN algorithm by using one to three training seasons and one testing season. Just like before, in prediction results by using disjoint dataset and cross-validation method, the best model prediction is obtained by KNN algorithm where the average prediction accuracy decreases as the number of seasons used increases. The comparison of prediction results of train & test validation method and cross-validation method by using disjoint dataset shows that all machine learning algorithms, especially KNN algorithm, and except decision tree, produce better average prediction results using cross-validation method when it is possible to predict the accurate future events data. As well as all comparisons before, in prediction results by using up-to-date data and train & test validation method, KNN algorithm also creates the best model prediction by using the same number of training and testing seasons. However, up-to-date dataset produce better predictions than disjoint dataset because the test phase data is added to the training dataset.

# Method

## Dataset

The original dataset, data.csv, is available on Kaggle and includes a list of all UFC fights from 1993 to 2022. Each row represents information on match details, two fighters (blue and red), and the winner.

* Dimensions: 5144 rows x 145 columns
* 9 categorical features, 136 numerical features
* Target (categorical: Blue/Red) specifies the winner

Each line is a compilation of the stats of the two fighters. The 'red' and 'blue' clusters depict fighters. As a result, each opposing side has the equivalent average stats from all of its previous fights. Statistics include harm inflicted by a fighter to an opponent and the other way around (represented by 'opp' in the column) in all fights this fighter has had, excluding those that have not occurred or are still in progress (not available in the data). The only column that tells you what happened is 'Winner,' which is the target variable [4].

##### Table I

##### Column Title and Definition

|  |  |
| --- | --- |
| **Column  Title** | **Column  Definition** |
| R\_  B\_ | prefix represents the red and blue corner fighter statistics correspondingly |
| \_opp\_ | mean of the harm sustained by the opponent to the fighter is shown by the columns |
| KD | number of knockouts |
| SIG\_STR | amount of noteworthy 'landed or attempted' strikes |
| SIG\_STR\_pct | substantial percentage of strikes |
| TOTAL\_STR | final tally 'landed or attempted' blows |
| TD | number of takedowns |
| TD\_pct | percentages of takedown |
| SUB\_ATT | number of attempts at submission |
| PASS | number of times the guard was bypassed |
| REV | amount of landed Reversals |
| HEAD | count of critical strikes to the head ’landed or attempted' |
| BODY | count of critical strikes to the body 'landed of attempted' |
| CLINCH | count of critical strikes in the clinch 'landed or attempted' |
| GROUND | count of critical strikes on the ground 'landed or attempted' |
| win\_by | way of winning |
| last\_round | last round of the fight |
| last\_round\_time | time when the fight ended in the last round |
| Format | format of the fight (3 rounds, 5 rounds etc.) |
| Referee | name of the Referee |
| date | date of the fight held |
| location | location where the fight took place |
| Fight\_type | weight class and title (bout or not) |
| Winner | winner of the fight |
| Stance | stance of the fighter (orthodox, southpaw, etc.) |
| Height\_cms | height of the fighter in centimeter |
| Reach\_cms | arm span of the fighter in centimeter |
| Weight\_lbs | weight of the fighter in pounds (lbs) |
| age | age of the fighter |
| title\_bout | boolean value of the title (fight or not) |
| weight\_class | weight class the fight (Bantamweight,  heavyweight, Women's flyweight, etc.) |
| no\_of\_rounds | number of scheduled fight rounds |
| current\_lose\_streak | amount of current concurrent losses of the fighter |
| current\_win\_streak | amount of current concurrent wins of the   fighter |
| draw | number of draws in the fighter's ufc career |
| wins | number of wins in the fighter's ufc career |
| losses | number of losses in the fighter's ufc career |
| total\_rounds\_fought | mean of total rounds done by the fighter |
| total\_time\_fought (seconds) | total time spent fighting in seconds |
| total\_title\_bouts | total number of title bouts the fighter participated in |
| win\_by\_Decision\_Majority | number of wins by majority judges decision in the fighter's ufc career |
| win\_by\_Decision\_Split | number of wins by split judges decision in the fighter's ufc career |
| win\_by\_Decision\_Unanimous | number of wins by unanimous judges decision in the fighter's ufc career |
| win\_by\_KO/TKO | number of wins by knockout in the fighter's ufc career |
| win\_by\_Submission | number of wins by submission in the fighter's ufc career |
| win\_by\_TKO\_Doctor\_Stoppage | number of wins by doctor stoppage in the fighter's ufc career |

## Preprocessing

The data acquired during preprocessing are columns that depict the fighter's skills in dealing with his opponents, such as the R\_avg\_KD column, experience data such as R\_total\_time fought (seconds), and measurement data such as R\_weight\_cms. R\_win\_by\_Decision\_Unanimous data is omitted since it is regarded unimportant to the model to be trained. If the data is not removed, the effectiveness of the model to be trained may degrade.

Following the deletion of identity or less relevant data, categorical data will be sorted and labeled depending on the specified columns. The categorization of these columns focuses on helping computers to read existing data. Columns with more than two rows of data will be classified into different columns. One instance is the division of the R\_Stance column into five categories with values ranging from 0 to 1. This is done to prevent a significant correlation between R\_Stance and other columns when labeling them [1][8].

In addition to column labeling, there's also data scaling. The objective of scaling in this case is to make the data less diversified and closer together. The existing data can be more organized as a result of this procedure, which aids the model in grouping existing data [9].

After that, in this project, we will use Linear regression to fill in the empty values ​​of the fighter's reach. This is done because the height of a fighter has a ratio of 1.618 which is the ideal value for the length of the hand to the total height [13].

## Models

As explained in section I, the models that will be implemented in this AI are Logistic Regression, K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Decision Tree Classifier and Linear Regression. It should be emphasized that Logistic Regression, KNN, and SVM will be used to classify the data, whilst the Decision Tree Classifier will merge the data generated by the previous three models into a single data set. It should also be noted that Linear Regression is required to determine a fighter's reach.

As stated in the preceding paragraph, the model proposed in preprocessing is to convert height to reach in order to attain the golden ratio. The linear regression formula is as follows :



It can be seen from the formula above that x is the number of inputs, β is a parameter and ε which is a measurement error [14].

The first model that will be used in the main model is the Logistic Regression model. This model is designed to predict which winner will arise between the red fighter or the blue fighter. The logistic regression formula is as follows [1][10] :



According to the formula above, x is the inputted data, μ is the location parameter, and s is the location scale [10]. However, the formula can be simplified by changing it to the following form.



With β0 and β1 as constants and coefficients and x as data input [10].

KNN is the second model utilized in this main model project. The KNN model is a machine learning classification that classifies itself using as many references as k neighbors. KNN provides numerous methods for calculating the distance between two points. However, the focus of this study work will be on Euclidean distance [10][11].



It can be seen from the formula above that T is a new case, S is the value of proximity to neighbors, n is the value of each case's attribute, i is an individual attribute ranging from 1 to n, f is the similarity of the function attribute of i between case T and case S, and w is the weight assigned to attribute i.

The SVM is the third model used for initial prediction. SVM is a classification model that is similar to logistic regression in that it uses hard and soft margins to determine categorization [12]. SVM employs the following formula :



Where w denotes the accepted weight, x is the input, and b signifies the constant. The formula is a linear formula based solely on the formula above. The formula can be derived again by looking for the hard margin i.e. [1].





For soft margin can be seen by the following formula [1] :



The Decision Tree Classifier is the final model applied in this research.This model is utilized to enhance the three models that were previously involved while making the best decisions for a better outcome. This Decision Tree focuses on the division of False Positives and False Negatives as well as the True Values that have been generated by the Decision Tree. The probability value will be calculated, searched for, and compared to each column/value. Once the probability is obtained, the column with the highest probability value is made into a node, and the loop is repeated until all of the columns are inputted [15].

# Results

There are three types of experimental results from this UFC fight prediction. This is due to the fact that the researchers in this study conducted three experiments. The first experiment uses the same dataset as the second, but the columns processed in the models were different. Experiment 1 focuses on the results of the match while experiment 2 focuses on the match's match match match. For more details, examples of the results of the match are columns with win\_by or round\_fought and combat data for example such as SIG\_STR and SIG\_STR\_att. Experiment 3 will train a test set as well as be part of the experiment. The outcomes of such experiments will be shown in the table below.

TABLE II

ACCURACY OF THE MODEL

|  |  |  |  |
| --- | --- | --- | --- |
|  | Experiment I | Experiment II | Experiment III |
| Akurasi Linear Regression | 66.66667% | 67.26804% | 70.27491% |
| Akurasi Linear Regression Training Set | 69.4158% | 69.67359% | 67.13917% |
| Akurasi KNN | 64.5189% | 62.45704% | 69.75945% |
| Akurasi KNN Training Set | 71.84278% | 71.02663% | 64.92697% |
| Akurasi SVM | 66.06529% | 64.77663% | 85.05154% |
| Akurasi SVM Training Set | 70.5756% | 80.2835% | 66.68814% |
| Akurasi Decision Tree Classifier | 64.77663% | 64.77663% | 70.61855% |
| Akurasi Decision Tree Classifier Training Set | 72.7878% | 80.2835% | 67.80498% |

Before evaluating the experiment's overall accuracy, numerous points should be highlighted. Each model will be trained using an existing training set, including the Decision Tree Classifier, and the training set's projected outcomes will be entered into the model.

Table II shows that the classification of data is better when what is anticipated is the training set excluding those from experiment III. These results are possible because the training set data serves as a reference for the categorization. As a result, the training set's accuracy will be higher than the test set's. Another factor that may contribute to such low accuracy is an excessive number of inputs or parameters.This can lead to confusion for the model to determine what needs to be fixed. A hyperparameter is needed to remind the model of this precision in order to assist the model in determining better results.

The comparison between experiments I and II is not statistically significant, but it should be addressed. Overall, Experiment II produces superior findings, even if the differences are little. This is because experiment II focuses on fighter performance in the octagon as opposed to experiment I, which focuses on fight results. Because a fighter can come out on top but have a relatively poor fighting performance, and also the fight performance cannot be lied to. Usually if a fighter has excelled in performance statistics, then the fighter can most likely win by decision if the fighter manages to get through all of the available rounds, unless there is controversy.

For experiment III, the test set's accuracy is always greater than the training set's. This is due to the fact that experiment III contained all of its datasets in order for the model to be trained on them, and the percentage of the test set was only approximately 1000 data compared to the training set, which included about 5000. So even if the number of training sets is actually more than the number of test sets, still the test set will get a larger percentage as a result of the data.

TABLE III

PRECISION OF THE MODEL

|  |  |  |  |
| --- | --- | --- | --- |
|  | Experiment I | Experiment II | Experiment III |
| Precision Linear Regression | 57.89474% | 58.06452% | 65.15156% |
| Precision Linear Regression Training Set | 58.79629% | 57.26073% | 48.10127% |
| Precision KNN | 47.97688% | 41.53005% | 63.91753% |
| Precision KNN Training Set | 62.58322% | 60% | 40.87791% |
| Precision SVM | 68.96552% | 47.77778% | 98.33333% |
| Precision SVM Training Set | 85.71429% | 96.0443% | 45.17958% |
| Precision Decision Tree Classifier | 64.77663% | 64.77663% | 85.05155% |
| Precision Decision Tree Classifier Training Set | 72.7878% | 80.28351% | 66.68814% |

For experiments I and II, the precision of experiment I was higher than that of experiment II. This can happen because the results of experiment II tend to favor the red winner or 0, causing the precision of experiment II to fall fairly significantly. Similarly, in experiment III, the majority of the results had a red winner or a 0 result, causing the precision to decline.

# Conclusion

The most valuable experiment is Experiment II, which will be used in the project. This is because, despite the fact that the test set of experiment II is not adequate, the comparison of the training set and the test set is one to five, causing experiment II to be the best compared to other experiments. The usage of hyperparameters is strongly suggested for future experiments due to the massive number of parameters or inputs that will be received by the model, which causes the model's accuracy to decline dramatically.

##### References

1. S. V. Stan, Strategic management in sports. The rise of MMA around the world—The evolution of the UFC, Ovidius University Annals, Economic Sciences Series, 1st ed., vol. 19, pp. 540-545, 2019.
2. K. Aggarwal., N. Yadav & M. Dwivedy, A Comparative Study of Machine Learning Algorithms for Prior Prediction of UFC Fights, Harmony Search and Nature Inspired Optimization Algorithms, Springer, Singapore,  pp. 67-76, 2019.
3. A. Cornman, G. Spellman & D. Wright, Machine learning for professional tennis match prediction and betting, Stanford University, 2017.
4. rezan21, UFC Prediction using Machine Learning methods | Python: Tensorflow, Keras.  Scikit-Learn. github.com. https://github.com/rezan21/UFC-Prediction (accessed June 15, 2022)
5. A. K. Uttam, and Gaurav Sharma. Application of artificial neural network based supervised machine learning based model for forecast of winner in mixed martial arts. In Recent Trends in Communication and Electronics, pp. 416-421. CRC Press, 2021.
6. Is MMA (UFC) More Popular Than Boxing?. wayofmartialarts.com. https://wayofmartialarts.com/is-mma-ufc-more-popular-than-boxing/ (accesed June 15, 2022)
7. T. Horvat, L. Havaš, and D. Srpak. The impact of selecting a validation method in machine learning on predicting basketball game outcomes. Symmetry 12, no. 3 (2020): 431.
8. F. Kamiran, and T. Calders. Data preprocessing techniques for classification without discrimination. Knowledge and information systems 33, no. 1 (2012): 1-33.
9. D. Li, B. Zhang, and C. Li. A feature-scaling-based $ k $-nearest neighbor algorithm for indoor positioning systems. IEEE Internet of Things Journal 3, no. 4 (2015): 590-597.
10. D. G. Kleinbaum., and M. Klein. Introduction to logistic regression. In Logistic regression, pp. 1-39. Springer, New York, NY, 2010.
11. G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer. KNN model-based approach in classification. In OTM Confederated International Conferences" On the Move to Meaningful Internet Systems, pp. 986-996. Springer, Berlin, Heidelberg, 2003.
12. S. Yue, P. Li, and P. Hao. SVM classification: Its contents and challenges. Applied Mathematics-A Journal of Chinese Universities 18, no. 3 (2003): 332-34
13. Danikas, Dimitrios, and Georgia Panagopoulos. "The golden ratio and proportions of beauty." Plastic and reconstructive surgery 114, no. 4 (2004): 1009.
14. G. A. F. Seber,, and A. J. Lee. Linear regression analysis. John Wiley & Sons, 2012.
15. W. Du, and Z. Zhan. Building decision tree classifier on private data.  (2002).