UFC TAU DUO

FIGHT CLUB: GROUP TWO

“First rule of Fight Club...”

MAIN EVENT CARD

DUSTIN SHADDIX
Wellborn, AL

LES MOLINARES
New York City, NY

MIGGY LACSON
Bacolod City, Philippines

THOMAS GRESCO
Manalapan, NJ

ADAM GLANTZ
Annapolis, MD

DARRELL REIVES
New York City, NY

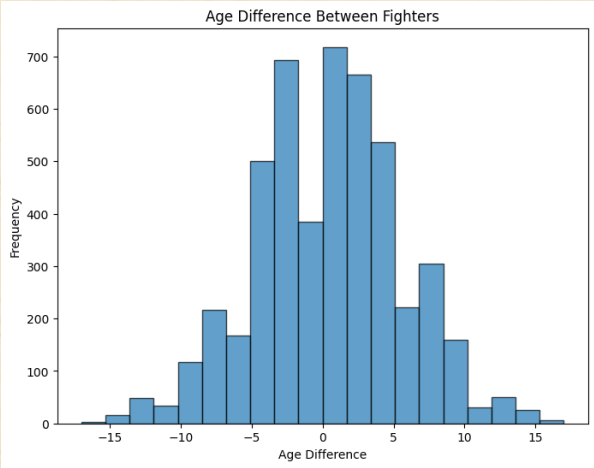
UFC PREDICTION MACHINE LEARNING MODEL

The following presentation details the stages of our predictive machine learning model project, covering elements of its conception, construction, adjustment, and finalization. The purpose of the model itself is to use past UFC match outcome data, along with favorable stats in common with victorious fighters, to see if we can predict the winners for new matchups and more importantly to see if our model gives a better view of the underdogs chance of winning.

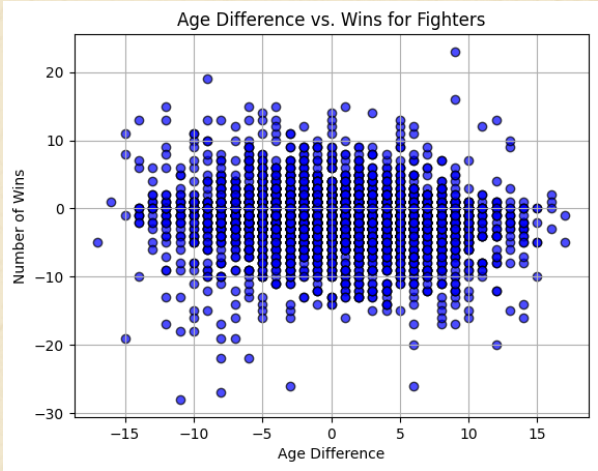
Crafted in a **Jupyter Notebook**, we utilized the following libraries:

- **Pandas** for data manipulation and storage
- **NumPy** for numerical computations
- **Matplotlib** for plotting and visualization
- **Seaborn** for advanced plotting
- **Scikit-Learn** for machine learning models and utilities
- **TensorFlow Keras** for deep learning models
- **MPL Toolkits** for 3D plotting

DATA CONNECTIONS AND RELATIONSHIPS

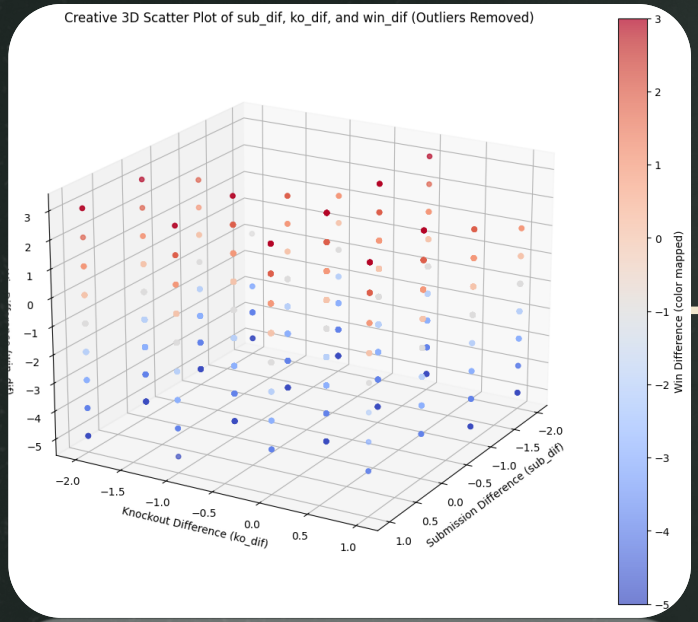


This histogram provides a detailed look at the age differences between fighters in various matches. A striking observation is that the highest frequency occurs at an age difference of zero, indicating that fighters of the same age are often pitted against each other. The distribution tapers off as the age difference increases, suggesting that match-ups between fighters with large age gaps are less common.



This scatter plot offers a comprehensive view of the relationship between the age difference of fighters and their victories in the UFC dataset. Each data point represents a fighter, plotted according to the age difference with their opponent and the number of wins they've secured. Notably, outliers at both ends of the age difference spectrum suggest that significant age gaps can either positively or negatively impact win rates.

DATA CONNECTIONS AND RELATIONSHIPS



SUB DIFF

The disparity in the number of knockouts between two fighters in a match.

KO DIFF

The difference in the number of submission wins each fighter has achieved.

WIN DIFF

The gap in total number of wins between two competing fighters.

ML MODEL GAUNTLET

DECISION TREE

Splits the dataset into subsets based on feature values, making it useful for both classification and regression tasks.

DUMMY CLASSIFIER

Provides a simple base for comparison by using basic rules like predicting the most frequent class.

SVM MODEL

Support Vector Machines find the best hyperplane to separate data into classes, effective in high-dimensional spaces.

RANDOM FOREST

An ensemble method that builds multiple decision trees for classification or regression, reducing overfitting.

KERAS MODEL

A flexible neural network framework that can model complex, non-linear relationships between inputs and outputs.

ADABOOST MODEL

Boosts the performance of weak classifiers by focusing on classification errors and adjusting subsequent models.

MODEL SELECTION

{ ? }

DECISION TREE

Versatile in handling both classification and regression tasks. Its interpretability makes it popular in business decision-making and medical fields.

RANDOM FOREST

Ideal for large datasets and works well for both classification and regression. Commonly used in customer segmentation and stock market forecasting.

KERAS

Highly versatile, suitable for complex tasks like image recognition and natural language processing. Often seen in applications requiring advanced feature learning.

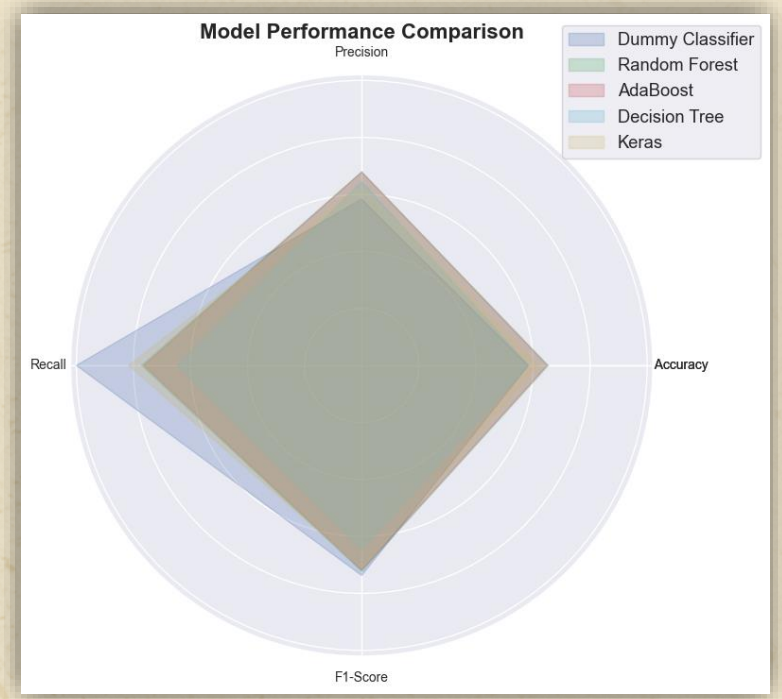
ADABOOST

Excels in classification tasks, especially with imbalanced class distribution. Often applied in customer churn predictions and medical diagnoses.

DUMMY CLASS

Serves as a baseline to compare other models' performance. Not designed for real-world applications, but to set a minimum performance threshold.

METRICS AND OPTIMIZATION



Recall: Measures the ability of the model to identify all relevant instances. High recall is crucial in medical diagnostics and fraud detection where missing a positive case can have severe consequences.

Precision: Focuses on the quality of the positive predictions. Important in scenarios where false positives are more costly than false negatives, like email spam detection.

Accuracy: Gives a general idea of how well the model is performing but may not be informative if the classes are imbalanced. It's a good general-purpose metric.

F1 Score: Harmonic mean of precision and recall, useful when you want a balance between these two metrics. Ideal for imbalanced datasets.

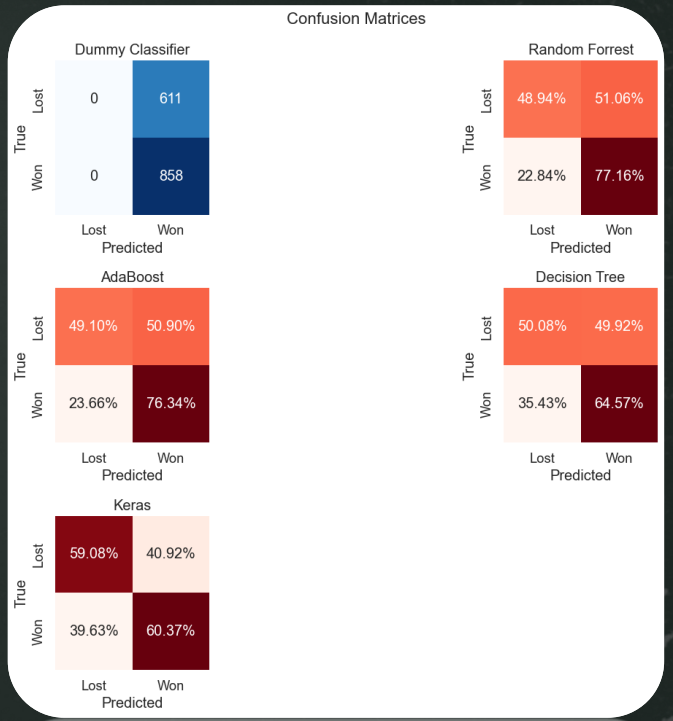
METRICS AND OPTIMIZATION

These five confusion matrices evaluate the recall metric for each machine learning model we explored.

The value in each cell, such as a given percentage in the "lost/lost" cell, represents how well the model predicts losses when they occur.

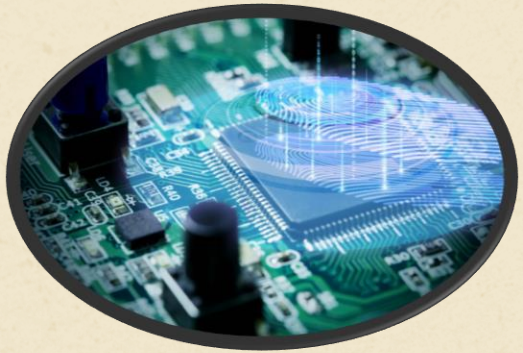
For instance, a 56.63% means the model accurately predicts a loss 56.63% of the time among all instances where a loss actually happened.

The **Dummy Classifier** simply predicts the majority class and has limited predictive power. Its metrics are expressed in absolute values to serve as a baseline for comparing the performance of more sophisticated models.



FURTHER METRICS: THE ROC CURVE

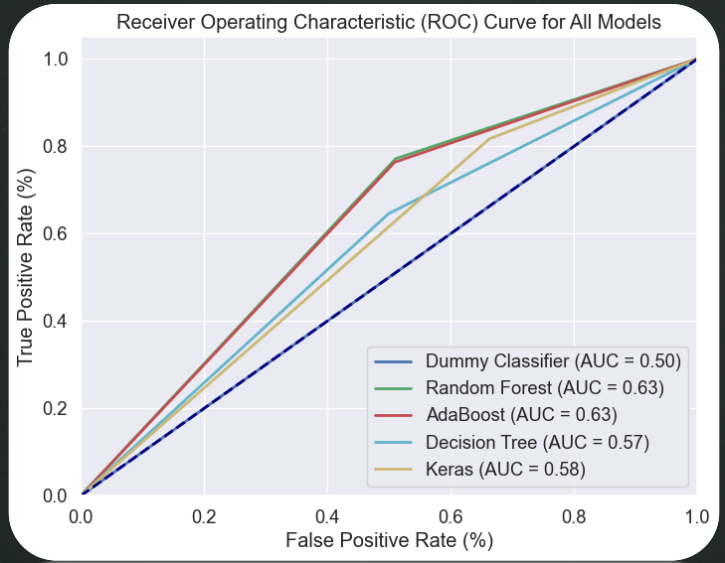
An **ROC Curve (Receiver Operating Characteristic Curve)** is a graph that shows how well a machine learning model distinguishes between two classes or categories—like "Yes" and "No." The curve plots the model's **True Positive Rate** against its **False Positive Rate**. The closer the curve is to the top-left corner, the better the model is at classification. It's a quick way to compare the performance of models like Random Forest, AdaBoost, Dummy Classifier, Keras, and Decision Tree.



FURTHER METRICS: THE ROC CURVE

- In this chart, we're sizing up our chosen models to find superiority in their ROC curves. The curve illustrates each model's classification ability.

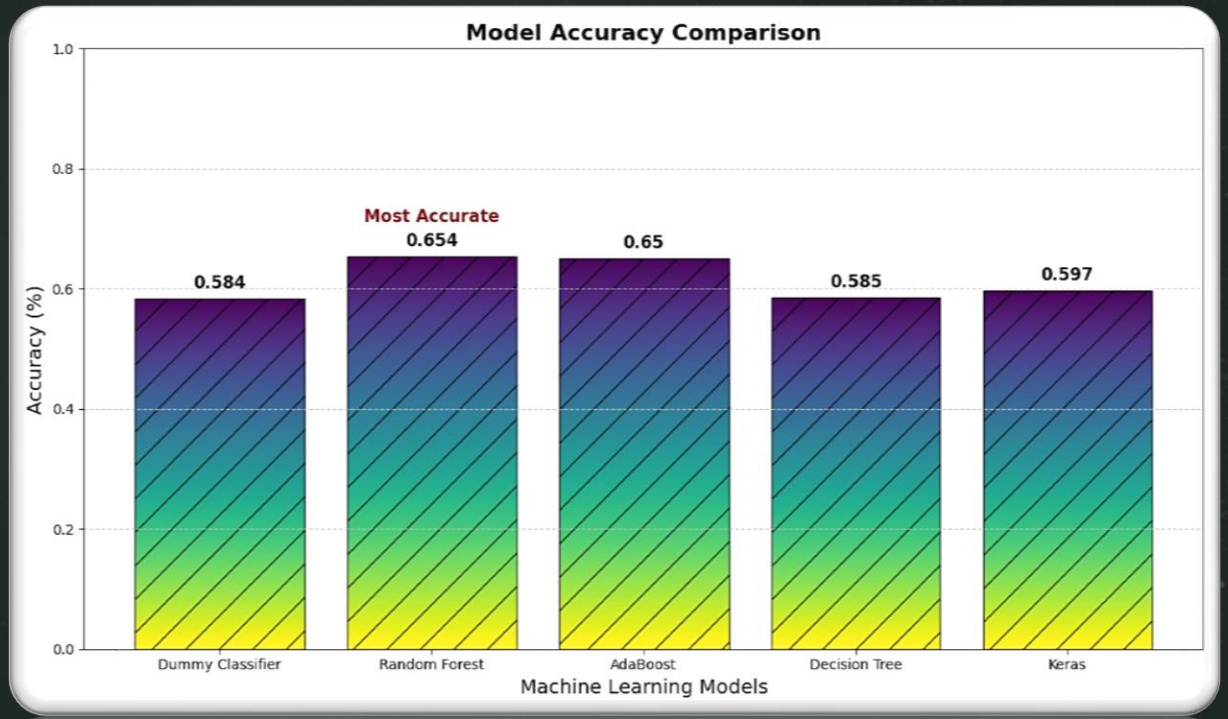
- The **AUC (Area Under the Curve)** scores in the legend range from 0 to 1, with higher scores meaning better performance.



- **Random Forest** and **AdaBoost** top the chart with AUCs of 0.63, indicating strong performance.
- **Decision Tree** trails with an AUC of 0.57, showing moderate effectiveness.
- **Keras**, with an AUC of 0.52, is just above random guessing.
- **Dummy Classifier**, our baseline, has an AUC of 0.50, akin to flipping a coin.

ML MODEL TRIAL RUNS

In the models we experimented with, our final metric of priority was **accuracy**, followed by efficiency. The **SVM** model not pictured in this chart returned an accuracy of 65.8% but took several minutes to run. Though the score differences are marginal at best, we chose the **Random Forest** model to build upon.



METRICS AND OPTIMIZATION

66%

The highest accuracy metric we could reach with any of the models.

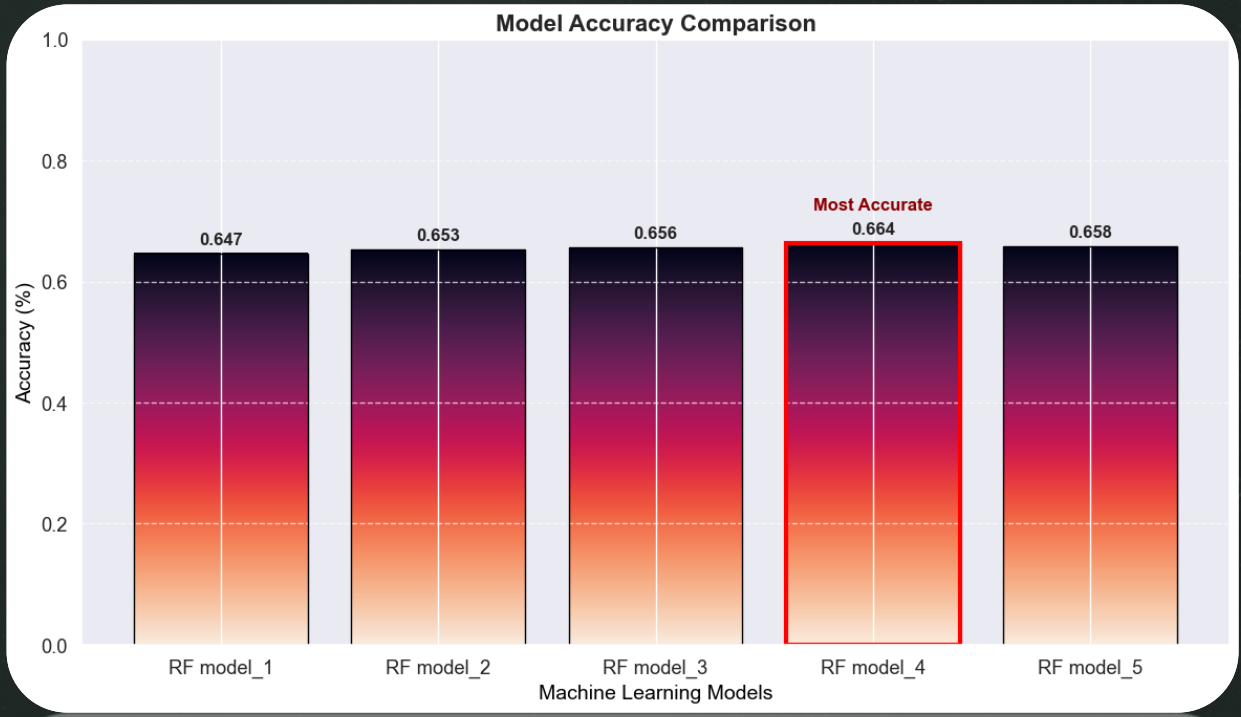
- **Random Forest**: Importance of metrics like recall and precision can vary depending on the application (e.g., medical diagnosis might prioritize recall).
- **AdaBoost**: Focus on precision and recall may help in applications where one wants to give more weight to a specific class.
- **Dummy Classifier**: These metrics help reveal that the model is not learning anything meaningful from the data; a good Dummy Classifier should have poor scores.
- **Keras**: High accuracy is usually the goal, but in healthcare applications, a high F1 score or recall may be more important.
- **Decision Tree**: The interpretability of a decision tree makes precision and recall easy to understand at each decision node.
- Each of these models might prioritize these metrics differently based on the problem they're solving.

OPTIMIZATION TWEAKS

MODEL 1	64%	Uses 240 trees and a square root rule for max features, with leaf nodes requiring a minimum of 10 samples and a max depth of 10. This configuration aims for more robustness by increasing the number of trees.
MODEL 2	65%	Similar to Model 1 but with 280 trees and leaf nodes requiring at least 20 samples. The increased minimum leaf sample size aims for more generalized predictions.
MODEL 3	65%	Rolls back to 180 trees and restores max features to 7. It also has a minimum leaf sample size of 6, aiming for a balance between robustness and generalization.
MODEL 4	66%	Uses 130 trees and is trained on a subset of features with importance above 0.03. This aims to focus the model on the most relevant predictors.
MODEL 5	65%	Similar to Model 4 but with a more restricted feature set with importance above 0.06, and 180 trees. The reduced feature set is aimed at pinpointing the most influential variables.

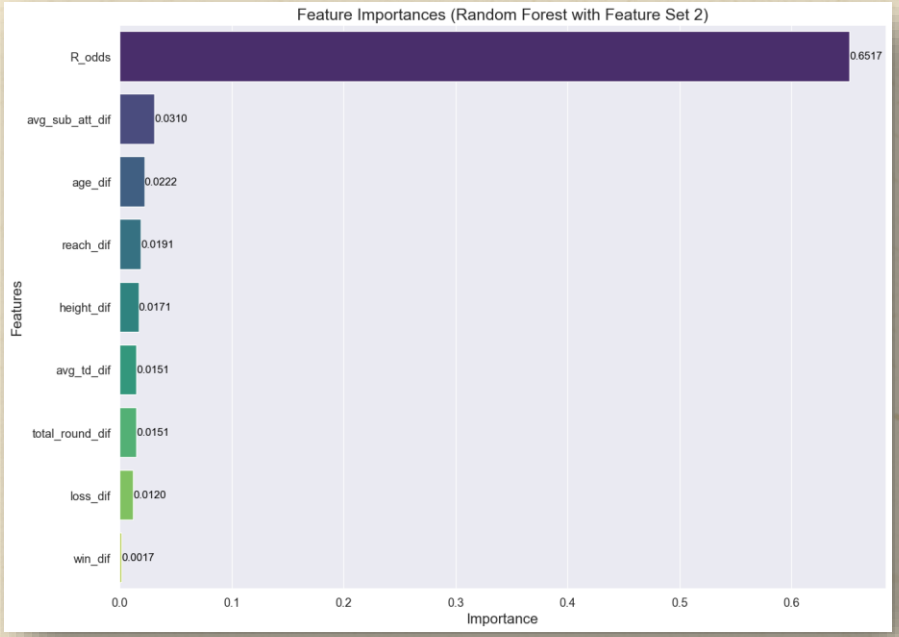
OPTIMIZATION TWEAKS

Each model's configuration tweaks aim to balance robustness, accuracy, and generalization based on feature importance, number of trees, and other hyperparameters. Though marginal at best, the **Fourth Random Forest** was chosen as the best.



OPTIMIZATION TWEAKS

- **R_odds**: With an importance score of 0.651, this is by far the most impactful feature, indicating that odds play a significant role in the model's decision-making process.
- **Avg_sub_att_dif**: At 0.031, this feature is considerably less important than "R_odds" but still holds some weight in the model.
- **age_dif**: Even less important with a score of 0.022, indicating that age difference has some but limited influence on the outcome.
- **reach_dif** & **height_dif**: These physical attributes have similar low importance scores of 0.019 and 0.017, respectively.
- **avg_td_dif** & **total_round_dif**: These have very close importance scores (0.015), suggesting that takedown differences and total round differences are of similar importance but less so than others.
- **loss_dif**: This feature has a low importance of 0.012, making it relatively less significant in this model.
- **win_dif**: With an almost negligible importance of 0.001, this feature appears to have the least impact on the model's decisions.



CONCLUSION

As we wrap up our UFC prediction project, it's worth noting that we chose the **Random Forest** model for its strong performance, but there's a whole array of machine learning models out there suited for various tasks. Models like **Keras** and **Decision Trees**, and even models we did not include in the end like **XGBoost** and **Support Vector Machines**, have their own unique strengths, depending on the application.

Keep in mind that predictions, especially in something as unpredictable as UFC fights, are influenced by a multitude of factors, making the task challenging yet intriguing.

Thanks for joining us on this journey, and let's stay curious about the endless possibilities machine learning offers!

CREDITS

- *Data collected from Kaggle*
- *PowerPoint template collected from SlidesGo*
- *Pokémon Type Icons from Bulbapedia*
- *Fight Club soap logo from PNG Wing*
- *UFC Icons from UFC (wow)*
- *Code written and visualizations generated in Visual Studio Code*

"...you do not talk about Fight Club."