Predicting Winning Boxers

1) Problem statement:

The Problem:

Can we predict the winner of a boxing match?

Who could benefit from this information?

- Sports analysts/broadcasters
    - Sports prediction and pre-analysis is a large facet of sports broadcasting and entertainment. With more valuable predictions and analysis, certain sports analysts/broadcasters might experience higher ratings.
- Sports betting facilitators?
    - More accurate predictions would forecast more reliable betting odds.
- Boxer managers/trainers
    - Information about the opposing boxer might inform the boxer's game plan as to how to train for the fight
        - For example, if a boxer's opponent tends to win by decision instead of by knockout, then that boxer might fight in a way that forces his opponent into a situation where he has to be more aggressive to accrue points.
        - If a boxer's opponent tends to win by knockout, then the boxer might aim to fight more defensively.

2) Data Wrangling/Cleaning

For my data, I chose a CVS file from the Kaggle database containing various descriptors of the boxers involved in a number of matches, including their physical attributes, and the results of the match.

Cleaning steps:

1. Explore nature of each data column
    a. Using .info, .describe, .plot, etc.
2. Fill in / remove entries where appropriate
    a. Using .where and .fillna

First, I used .info() to explore how many columns contained sufficient data. Upon investigation, I noticed that the columns missing the most data were the physical attribute columns. All other columns had close to 100% of the data. Since some of the physical attribute rows were missing 50% or more of column data, I decided to limit the DataFrame to only those rows where all physical attribute columns contained non-null entries. From there, I eliminated outliers and imputed missing entries, as described below.

- *Physical attribute columns*
    - age, height, weight, reach
        - Using .describe(), .info(), and .plot(), I was able to explore the nature and distribution of these numeric columns. I denoted any values outside of an acceptable range as NaN, and filled the NaN's appropriately. I'll use the "age" columns as an example, although other numeric columns were imputed similarly:
            - Almost all of the data was centered around the ages of 16-41, with a few outliers (such as one entry with an age of 0, and another with an age of 2016). To reconcile these exceptions, I kept any ages between the 1st and 99th percentile of the dataset, and filled all other values with NaN (using .where). Since the distribution of this column was slightly right-skewed, I decided to use the median of the adjusted column to fill in all NaN values.

- *Stance column*
    - Entries in this column were either "Orthodox" or "Southpaw." In this dataset, all of the matches involved boxers with the same stance: either both orthodox, or both southpaw. Since this column will not be useful in predicting the winner of the match, I decided to remove it.

- *Wins, losses, draws, and kos columns*
    - Similar to the other numeric columns, these columns contained some unacceptable values (such as a total win count of 355, which has never been accomplished by any boxer). I revised the column the same way I revised the other numeric columns, filling in NaN values using the median.
    - I also noticed that some boxers had a total "kos" count that was higher than that boxer's "wins" count, which is impossible. Therefore, I restricted the DataFrame to rows where this was not the case.

- *Result and decision columns*
    - Luckily, these categorical columns contained no null entries, so I was able to leave them as they were.
    - Since there were so few matches where the result was "draw" (only about 4%), I decided to eliminate these entries from the data.

- *Judges columns*
    - These numeric columns contained the three judges' scoring for each boxer in the event that the boxing match went the full 12 rounds. However, since the "result" column already denotes the winner of the match, this column is unnecessary. Therefore, I removed these columns from my DataFrame.

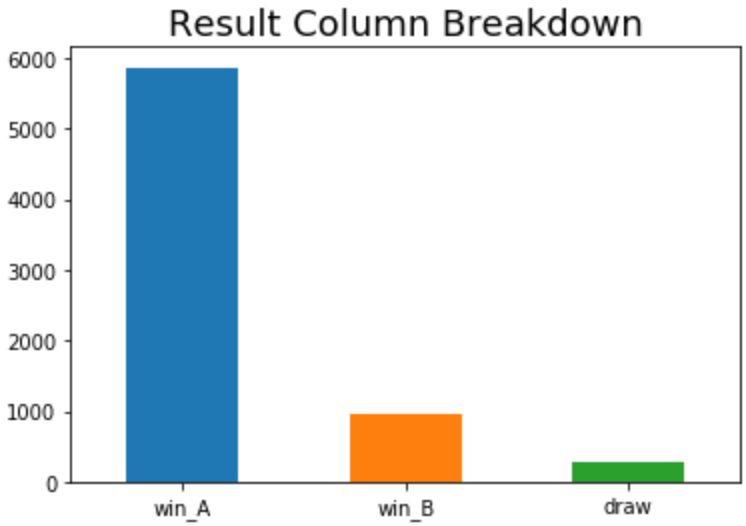
3) Summary of Exploratory Data Analysis Findings

There are a few questions we can ask about our data:

1) What is the relationship between the differentials in the physical attributes of the two boxers and the winner of the match?

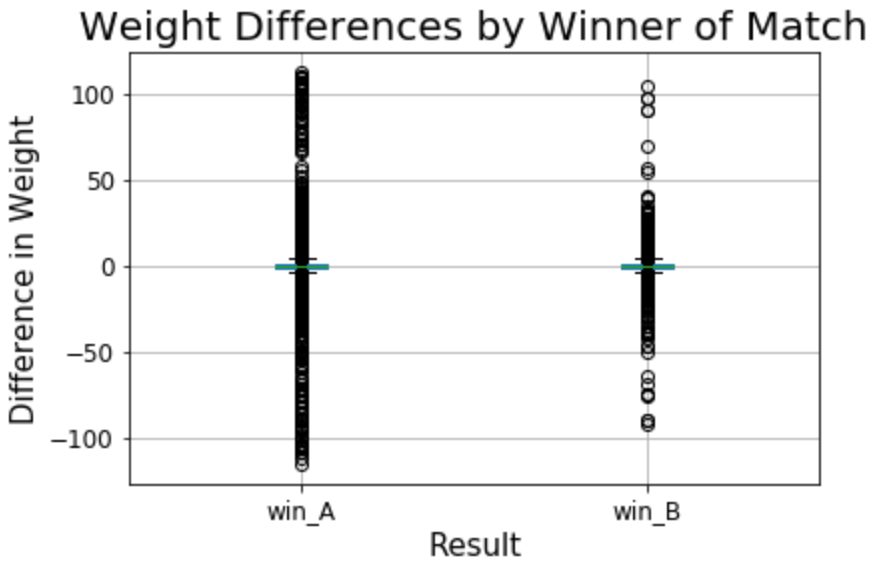
We are not so much interested in considering each boxer's physical attributes in isolation. Rather, since we are attempting to predict who won each match, we are more interested in *how each boxer compares to his opponent*. To this end, we will be exploring the *differentials* in the physical attribute columns.

Upon examining the "result" column, we see that in this dataset, Boxer A is almost always the winner:

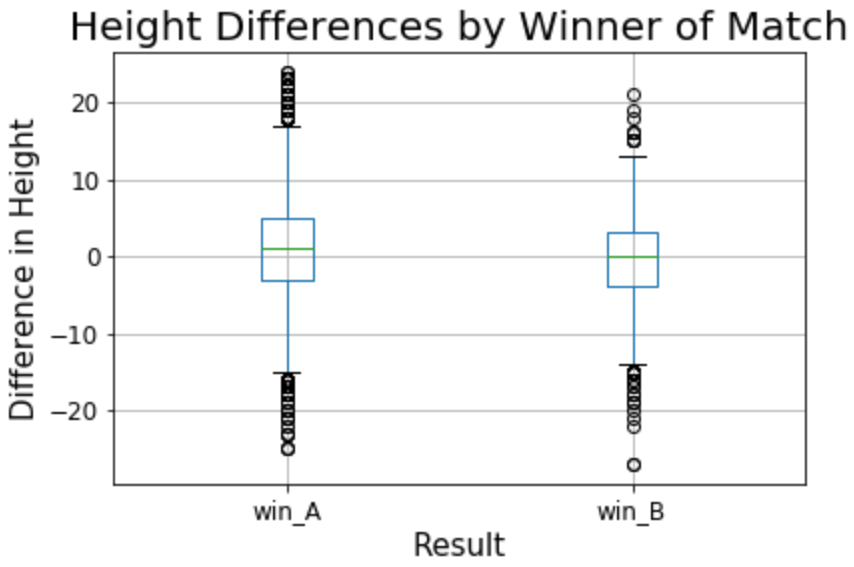


In fact, using .value_counts(), we see that about 82.5% of the matches in the dataset have "win_A" as a result. Going forward in our analysis, we will focus mainly on Boxer A.

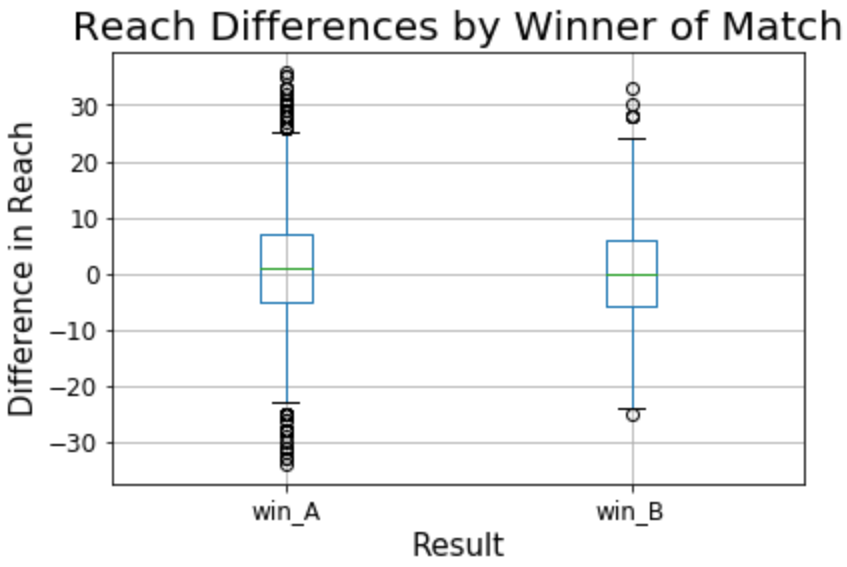
For this question, I began by creating four columns in my DataFrame, corresponding to the differences in Boxer A and Boxer B in age, height, weight, and reach. I then plotted the four columns with boxplots, separating each plot by the result of the match:



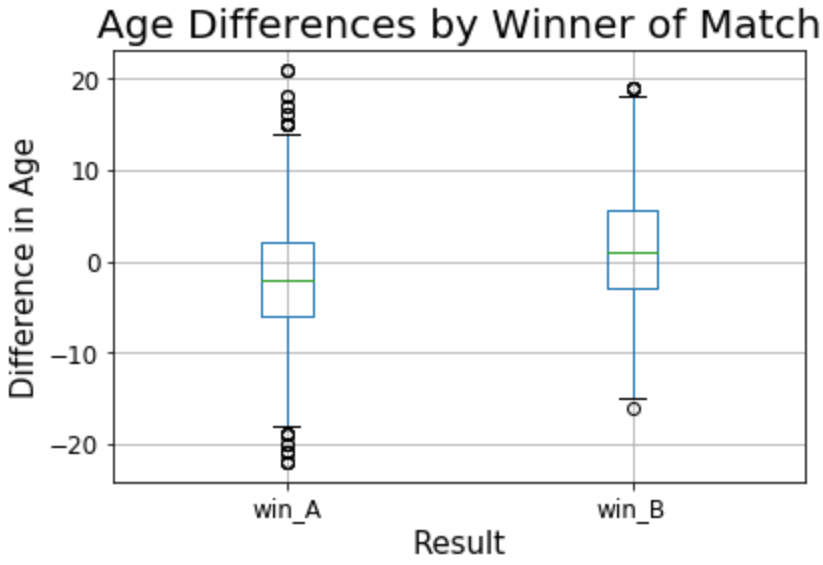
No significant trends can be noted from the weight differentials. Typically, boxers will weigh in very close to the upper bound of their respective weight classes. Therefore, it makes sense that on average, the difference in boxer weight for a given match is very small.



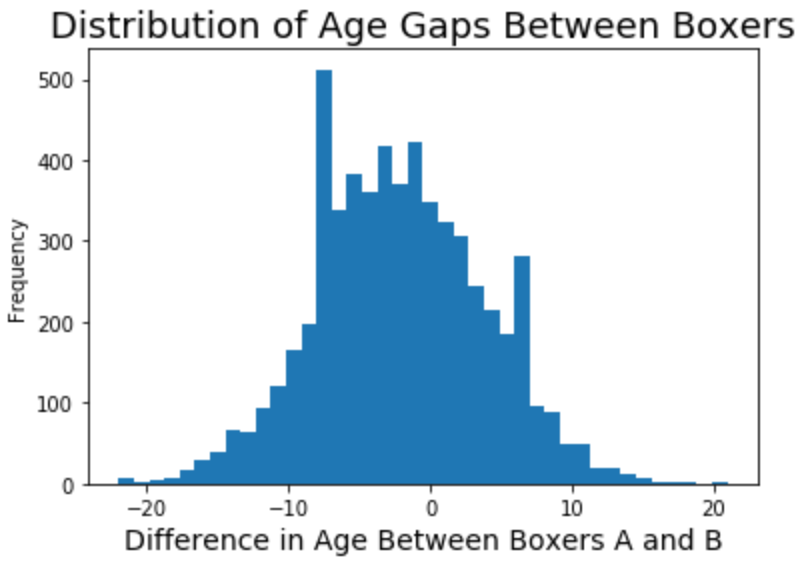
From this graph, we can see that when Boxer A wins, the median of the data is above 0. This means that more than 50% of the time, Boxer A was taller than Boxer B.



Similarly, we can see from this chart that when Boxer A wins, more than 50% of the time, he had the longer reach.



Here is the most prominent trend. When Boxer A won, nearly 75% of the time, Boxer A was younger than Boxer B! This might lead us to suspect that younger boxers have a significant edge over older boxers. Here is the distribution of age gaps for those matches where Boxer A won:



As we can see, the largest proportion of the data lies where Boxer A is somewhere between 1 to 10 years younger than Boxer B.

2) What are the characteristics of a winning boxer?

Here are the statistics for the differential columns, filtered for when Boxer A wins:

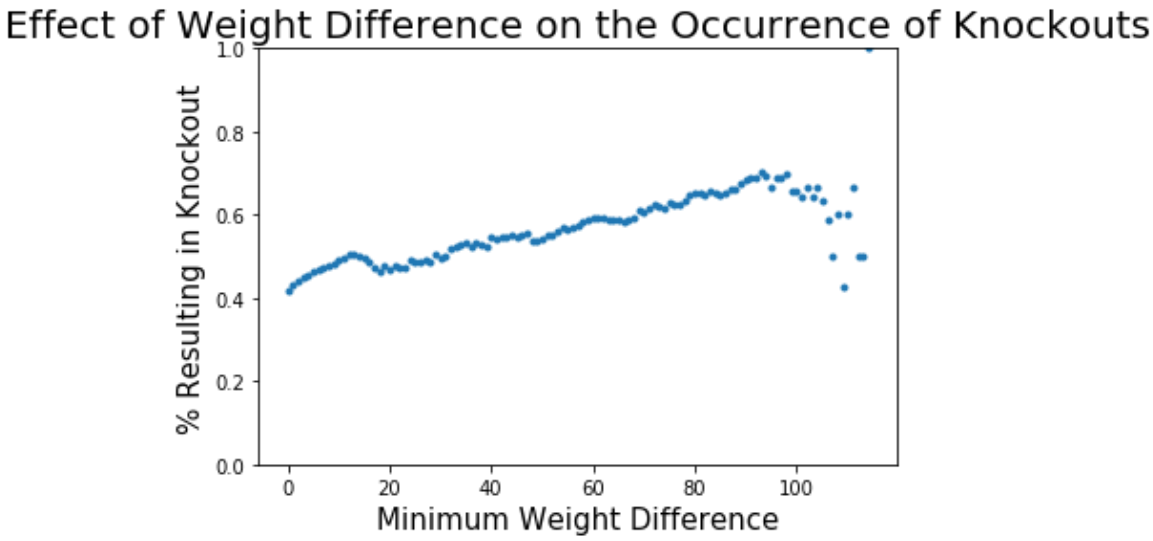
	diff_age	*diff_height*	*diff_weight*	*diff_reach*
mean	-2.081003	0.940655	-0.224932	1.097885
std	5.991132	6.801658	14.718452	9.656878

As we can see from the means, the winning boxers tend to be younger, taller, lighter, and longer.

3) How do knockouts relate to the differential columns described above?

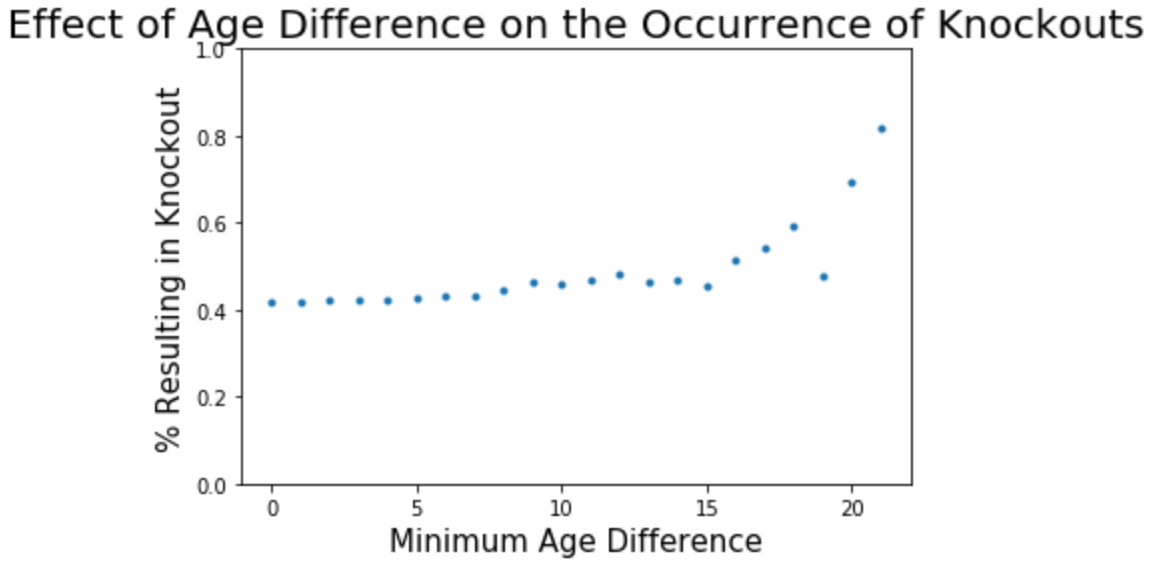
We have examined the physical attributes of winning boxers compared to their opponents. In the same vein of thought, we are interested in determining how the *size* of the differentials in these physical attributes are related to the outcome of the match; in particular, to the occurrence of knockouts. It is reasonable to suspect that, for example, if one boxer is significantly heavier than the other, then there is a good chance that the heavier boxer might knockout his opponent.

I constructed four graphs to explore this idea. The following graph shows the percentage of matches that resulted in knockout, given the weight difference was greater than or equal to a given quantity (x-axis):

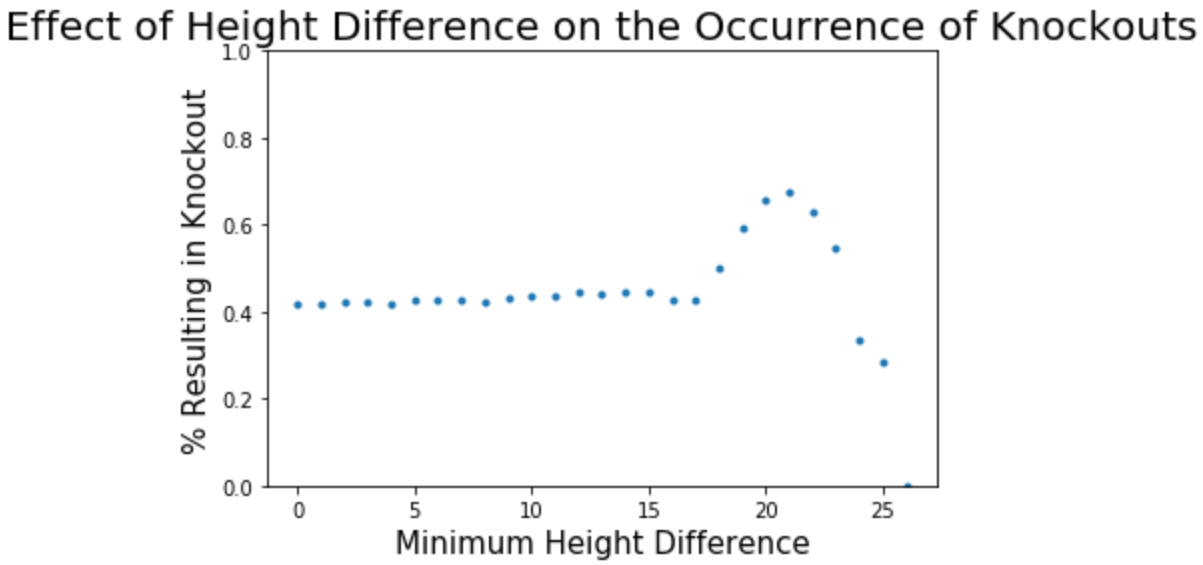


Notice that as we increase the minimum disparity in weight between the two boxers, the percentage of matches resulting in knockout tends to increase.

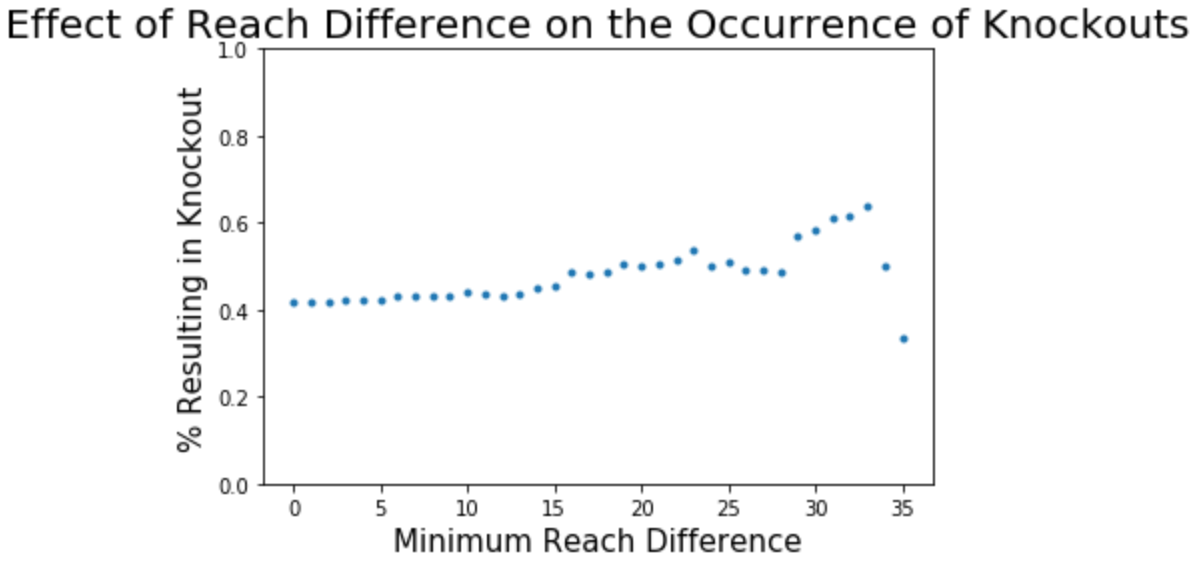
The following three displays show similarly constructed graphs for age, height, and reach differentials:



There is a gradual increase in the percent of matches resulting in knockout as the minimum age difference increases. The greatest uptake is noted for minimum age differences of at least 15.

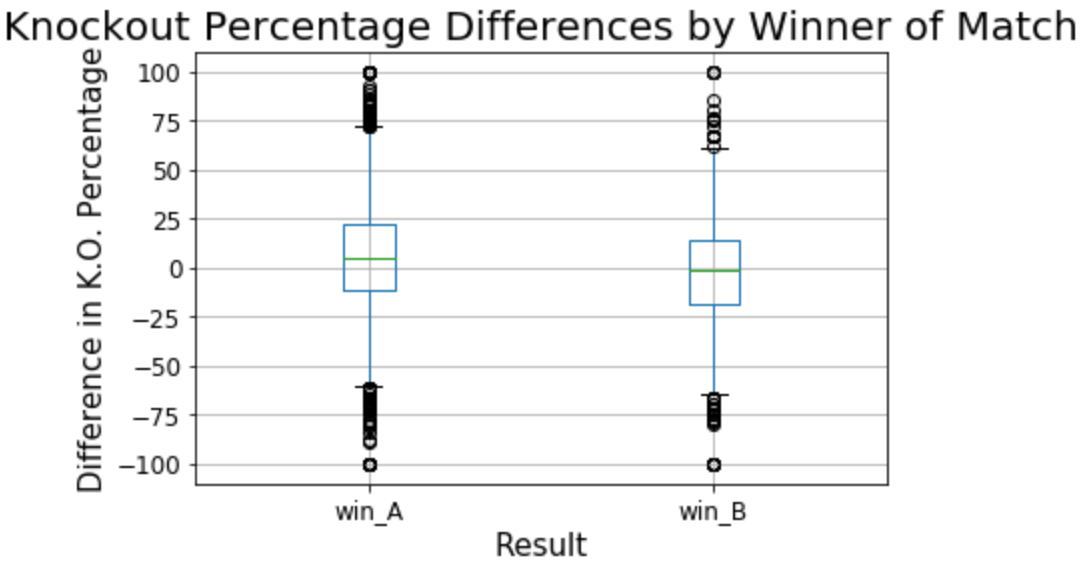


There doesn't appear to be any consistent trend between the minimum height difference and percentage of matches resulting in knockout.



Finally, we notice a steady increase in the percentage of matches resulting in knockout and minimum reach difference (with the exception of the last two data points).

It would appear that as the physical differences between the two boxers becomes more extreme (with the exception of height), the occurrence of knockouts becomes more and more likely. To this end, I decided to add a new differential column to analyze: the difference in boxer knockout percentages. Here is the boxplot of K.O. differentials, separated by the winner of the match:



Judging by the upper quarter of the data in the “win_A” boxplot, we see that in 25% of the matches where Boxer A won, he had a knockout percentage that was approximately 25 or more percentage points higher than his opponent! More than 50% of the time, Boxer A had a larger knockout percentage when he won. This column seems similarly predictive of the winner of the match, and so I will be adding it to my set of candidate predictive features.

4) Statistics to Support Findings

Our data story posed two main questions that we wish to support using inferential statistics:

1) Is there a relationship between the physical attribute differentials of the two boxers and who won the match?
2) Is there a correlation between the size of the physical attribute differentials and the occurrence of knockouts?

Hypothesis testing on physical attribute differentials

From our exploratory data analysis, we expected that the distributions of the physical attribute differential columns were different based on the winner of the match. In other words, if Boxer A won, we expected, on average, different distributions than if Boxer B won. If we find that the distributions are in fact different, then we can confidently use those columns as predictive features when we later build our model.

I'll explain how the hypothesis test was constructed to test this using the age differentials as an example. The hypothesis tests for the differentials in height, weight, reach, and knockout percentage were conducted similarly.

The test was on two datasets: the age differential data for when Boxer A won, and the age differential data for when Boxer B won. I then ran a permutation hypothesis test, testing the hypothesis that the two differential distributions were the same (using the difference in sample means as the test statistic). This test returned an extremely small p-value, and so we conclude that the distributions are not identical.

The same test was conducted on the differential columns for height, weight, reach, and knockout percentage. The results were as follows:

age → *statistically significant*
height → *statistically significant*
reach→ *statistically significant*
knockout percentage → *statistically significant*
weight → *NOT statistically significant*

Based on the boxplots for the weight differentials analyzed above, this result is not entirely surprising. The distributions did not show any apparent difference. The boxplots for age, height, reach, and knockout percentage differentials, on the other hand, led us to expected conclusions: there IS in fact a difference in the differentials for winning Boxer A's vs. winning Boxer B's.

Hypothesis testing on correlation between size of physical attribute differentials and percentage of matches resulting in knockout

We are now interesting in verifying the following: as the differences in physical attributes between the boxers becomes larger and larger, is a knockout more and more likely? We saw in the scatterplots above that there appeared to be trends, but are these trends statistically significant? To answer this, I conducted four hypothesis tests on correlation.

Here are the observed correlation coefficients, as well as the result of each hypothesis test:

age-diffs vs. percent of matches resulting in knockout→ r = 0.731

- After running a hypothesis test on correlation, we were able to reject the null hypothesis that there was no correlation. The large value for the observed correlation coefficient reveals not only statistical significance, but practical significance as well. We can conclude that as the age gap between the two boxers grows, it is more and more likely that a knockout will occur.

height-diffs vs. percent of matches resulting in knockout → r = 0.104

- This hypothesis test returned a p-value of 0.3064. Since we obtained an extremely small empirical correlation coefficient, it is unsurprising that this test proved statistically insignificant. We cannot conclude that a correlation exists between height difference and the occurrence of knockouts.

weight-diffs vs. percent of matches resulting in knockout → r = 0.748

- Once again, our hypothesis test led us to a statistically significant p-value. This is the largest empirical correlation coefficient out of the four tests. It would appear that weight difference has a very large influence on the occurrence of knockouts. Perhaps hitting power, which is directly proportional to weight, leads to heavier boxers knocking out lighter opponents.

reach-diffs vs. percent of matches resulting in knockout → r = 0.619

- Finally, we were able to reject the hypothesis that there is no correlation between the size of the reach differentials and percentage of matches resulting in knockout. Once again we observed a strong empirical correlation, and so this test elicits both statistical and practical significance.

5) In-Depth Analysis: Machine Learning

Now it is time to fit several machine learning algorithms to a portion of our data, the training set, and use the model to predict the winner of the match on a holdout set (test set). I will be testing three separate models on the same data:

- Logistic Regression
- Decision Tree
- Support Vector Classifier

1) I will be using four of the five differential columns we have discussed as my predictive features: age, height, reach, and knockout percentage. We saw that the weight difference distributions were not statistically different, so we will exclude this column as a predictive feature. The remaining four features will make up our “X”
2) Our target is the winner of the match, found in the “result” column. I created our target variable, “y,” so that a 1 represents Boxer A winning the match, and a 0 represents Boxer B winning the match.
3) I performed a train_test_split on the data, using 30% of the data as a holdout test set.
4) As discussed in the EDA section, we noticed that most of the matches resulted in Boxer A winning the match. In fact, only about 15% of the matches saw Boxer B as the winner.

As such, we must solve the problem of class imbalance in our target variable. To this end, I tried three methods of evening out the classes: RandomOverSampling, SMOTE (oversampling), and undersampling. The best model produced overall, based on the accuracy and classification reports, was the SVC using ROS, which I will discuss below.

Logistic Regression

I chose to tune the "C" hyperparameter of the logistic regression model using GridSearchCV on the training set. I then trained the GridSearch model on the training data, and predicted the labels using the testing data.

For all three sampling methods, the accuracy score on the testing data hovered around 61 to 64 percent, with F1 scores for both classes hovering around 0.61 to 0.65. Overall, this model performed reasonably well for both classes, but not optimally. Here is the classification report for the top performer, using SMOTE oversampling:

	Precision	Recall	F1 score	Support
Boxer A	0.63	0.64	0.64	1744
Boxer B	0.64	0.63	0.64	1775

Decision Tree

For this classifier, I chose to leave the default parameters as they were.

The most optimal performance for this model was noted using RandomOverSampling. Here is the classification report:

	Precision	Recall	F1 score	Support
Boxer A	0.99	0.85	0.91	1744
Boxer B	0.87	0.99	0.93	1775

Not only was this model able to find all positive samples very effectively for both classes, but was also able to avoid incorrectly labeling negative samples as positive. With an accuracy score of approximately 92%, this model well outperformed the logistic regression. However, it did not perform as well as our optimal model, the SVC, described below.

Support Vector Classifier

There were three hyperparameters I chose to tune for this model: gamma, C, and kernel.

Once again, RandomOverSampling produced the most optimal performance of the three class-balancing techniques. Here is the classification report:

	Precision	Recall	F1 score	Support
Boxer A	0.99	1.00	0.99	1744
Boxer B	1.00	0.99	0.99	1775

This classifier produced staggering results, with nearly perfect precision, recall, and F1 scores for both classes, and an accuracy on the test data of 99.3%! It would appear our four differential predictive features were very strong in predicting the winner of boxing matches using this SVC model.

6) Conclusion: Final Thoughts and Further Research

We began with a simple premise: what makes a winning boxer? Compared to his opponent, we supposed that a winning boxer might exhibit particular physical characteristics. We then explored four interesting attribute differentials: age, weight, reach, and height. We found that winning boxers tended, on average, to be younger, taller, lighter, and longer.

Upon further investigation, we were curious about another metric: a boxer's knockout percentage. We found that as the differentials between the two boxers became larger for a given attribute, so did the percentage of matches resulting in knockout. This lead us to add the differential in knockout percentages between the two boxers to our set of predictive features.

Finally, we fit three different models to our data: logistic regression, decision tree, and support vector classifier. After tuning hyperparameters and experimenting with over and under

sampling to even out the class imbalances, we noted that the support vector classifier using ROS performed the best, with an amazing 99.3% accuracy.

What else might be interesting to explore/include?

1) Can we alter our model to predict a different target, such as the continuous values contained in the judges' scoring columns?
2) Could we predict a target variable with more than two classes, such as how the match ended (knockout, judges' decision, etc.)?
3) Could we include more or fewer features to improve our model?
4) What other information might we be able to gather about the boxers that would strengthen our predictive capabilities? Nationality? Diet? Number of years experience as a professional boxer?