Proposal: Sports Injury Prediction

Team 49: Muadh George, Tilman Gromme, Alex Kim, Zaid Mohammed, Alexander Thummalapalli

Introduction/Background

Literature Review

The Journal of Experimental Orthopaedics analyzed 249 studies that used machine learning as a tool for sports injury prediction and prevention [1]. They found that random forests, support vector machines, and artificial neural networks helped in detecting injury risk factors but lacked methodological quality, including instances of players appearing in both training and test datasets [1]. Furthermore, the Department of Biomedical Sciences at the Noorda College of Osteopathic Medicine analyzed KNN, K-means, decision tree, random forest, gradient boosting, and neural networks as viable ML algorithms for sports risk prediction and found that all had major weaknesses ranging from data set limitations to struggles with higher dimensionality of complex data [2]. Furthermore, a study involving soccer delves into some of the metrics used in this subject [3]. Metrics including “precision, recall, specificity and F1-score” are class-based and work by being “calculated for each class separately and averaged to provide a single overall score” [3]. However, these metrics are misleading when used on imbalanced datasets [3].

Dataset Description

Our includes six features: player age, weight, height, previous injury status, training intensity, and recovery time. These features predict a labeling of 1 or 0 depending on likelihood of an injury based on about 1000 data points ranging from ages 18-39, with half of them likely to experience injury.

Dataset Link

Problem Definition

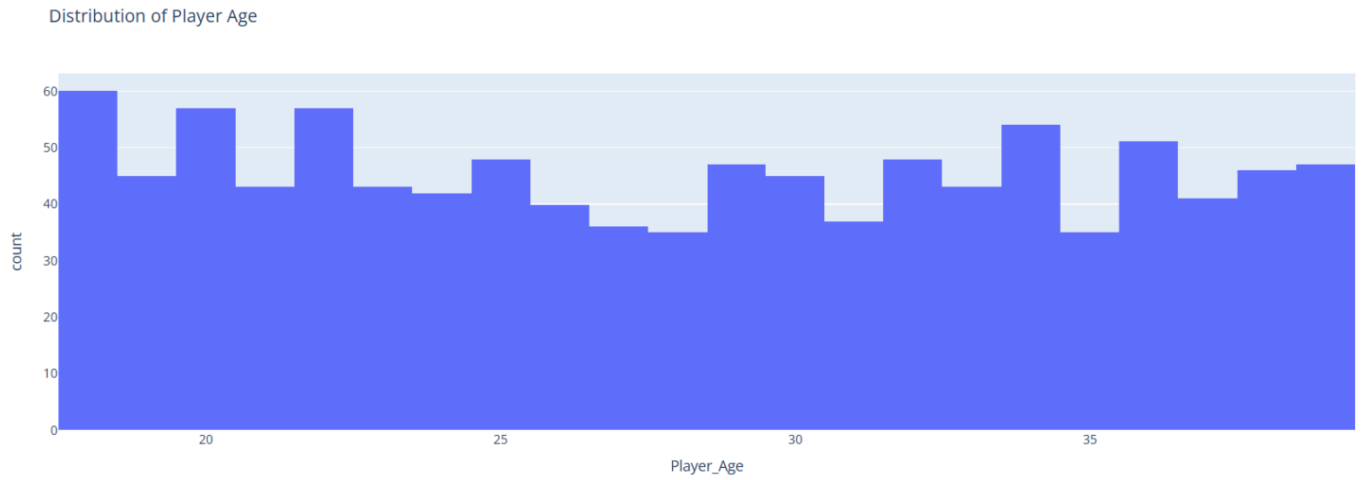
Sports injuries are a common issue among athletes due to repetitive physical strains induced by intense physical demands during competition. Although seemingly random, sports injuries can be predicted using ML by analyzing a variety of factors. This will lead to a better understanding of risk factors leading to higher likelihoods of injury, which can be used to inform athletes about precautionary measures. Our

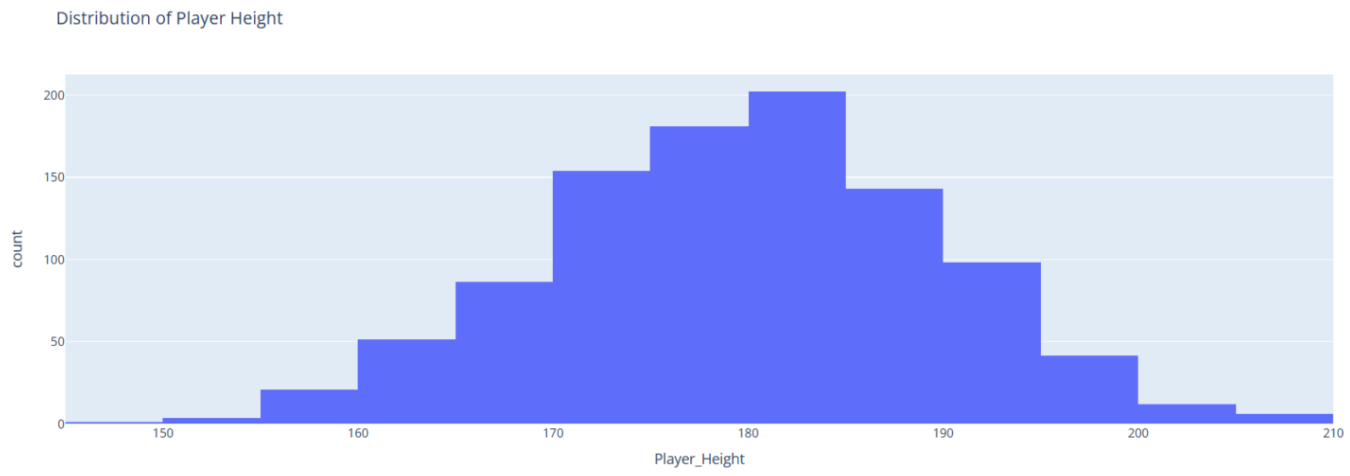
proposal differs from prior literature as we plan to see how employing logistic regression affects our results, and we are using different quantitative metrics.

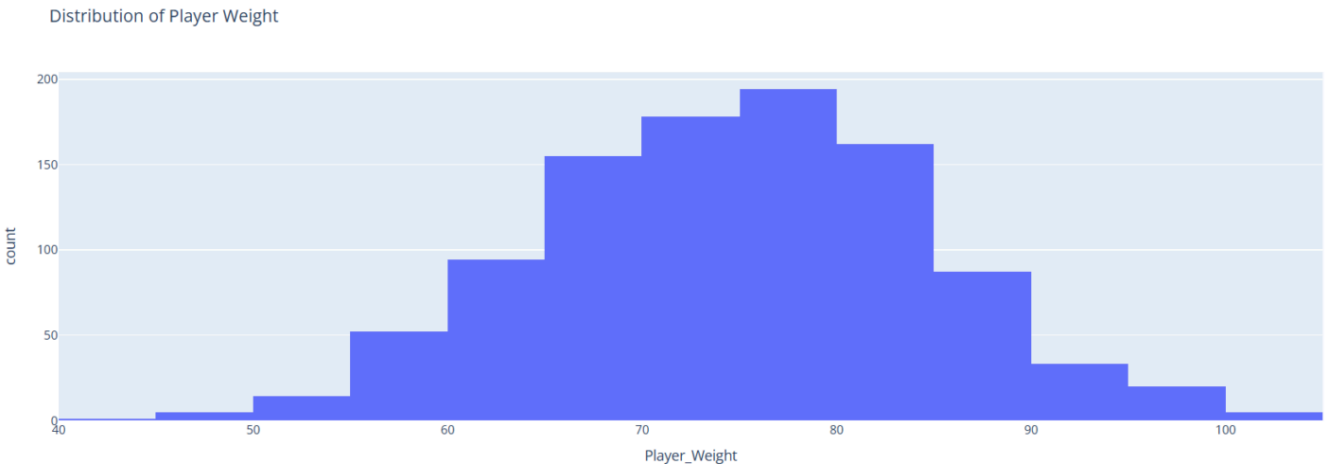
Methods

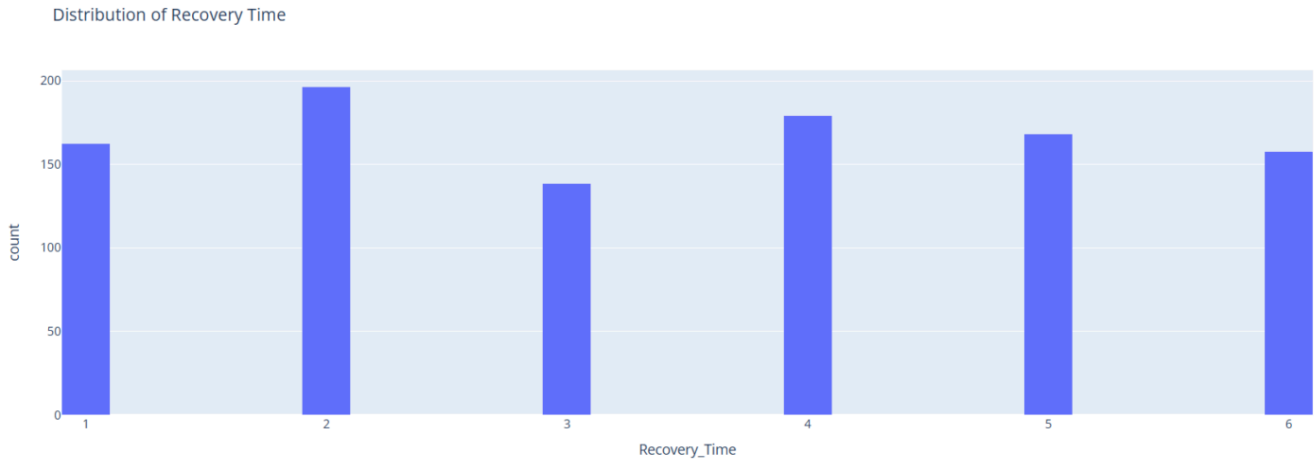
Data Preprocessing

- Normalization: is a technique used to scale data in a way that all of it is between 0 and 1. We used this technique because when analyzing this data set, there were many disparities between the range of values of the different features. Firstly for this, we verified whether player age, player height, player weight, and recovery time (the features with values outside the 0-1 range) are normally distributed to determine whether z-score can be used. We used the Shapiro-Wilk test for this. Then, we used z-score for normalization for the features that had a normal distribution to make all the data in a range that is easier to visualize. We ended up doing this for player weight and player height. For the features that didn't follow a normal distribution, we used min-max normalization to help with visualization. We ended up doing min-max normalization for player age and recovery time for this.

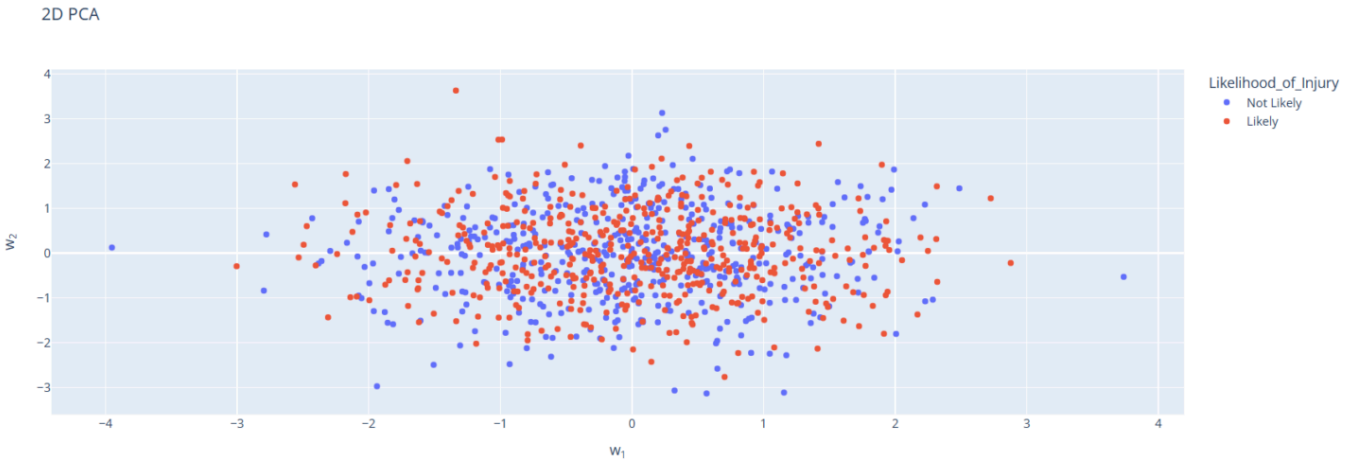


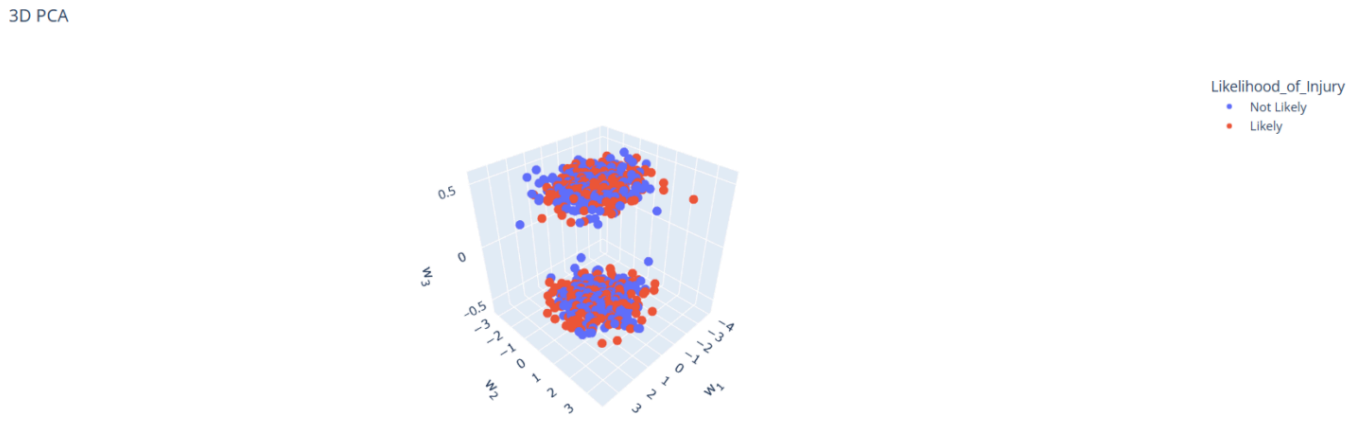




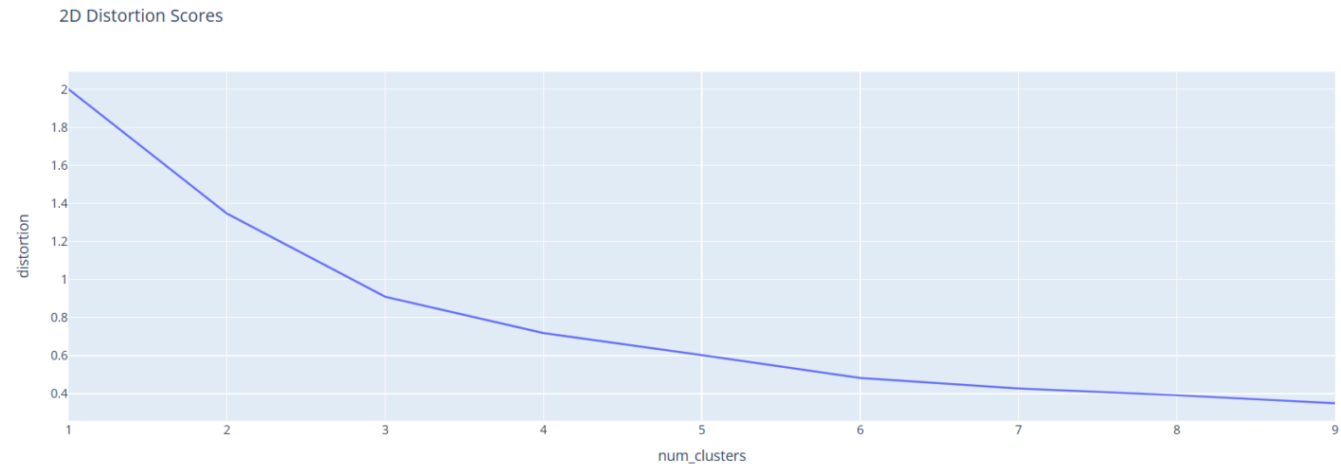


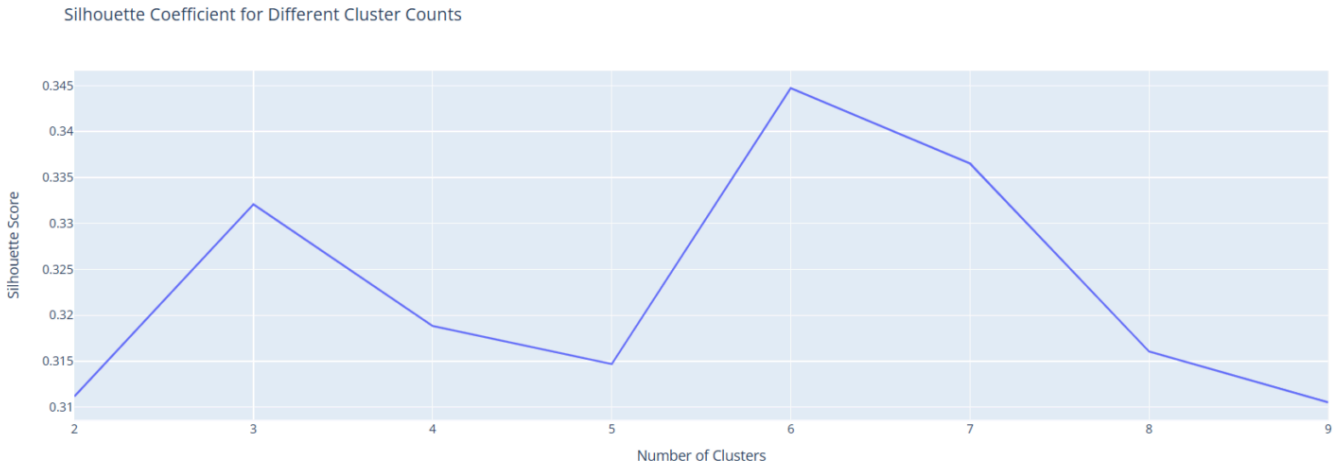
- PCA: is a helpful dimensionality reduction algorithm which looks at which principal components retain the highest amount of variance, and based on some threshold someone has in mind, they will use the principal components which are above that threshold. We used this preprocessing technique because we wanted to reduce the number of features we need to perform our ML models down by analyzing which ones are the most relevant. For our preprocessing, we saw that our first two principal components accounted for 60% of the total variance while the first three principal components accounted for 87% of the total variance. As can be seen from the below visualizations, PCA didn't do a great job of classifying the likelihood of injury.

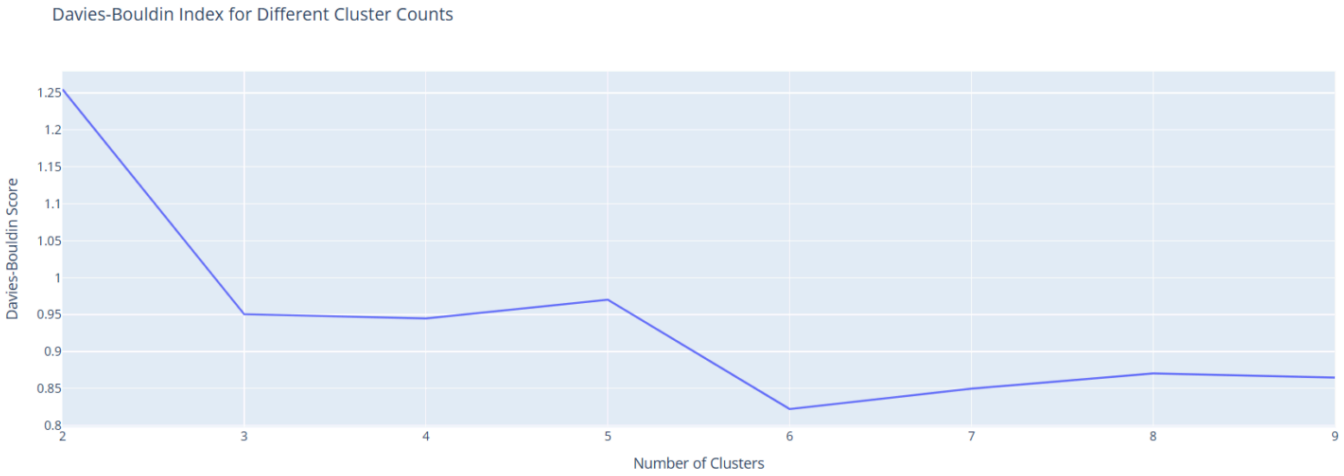


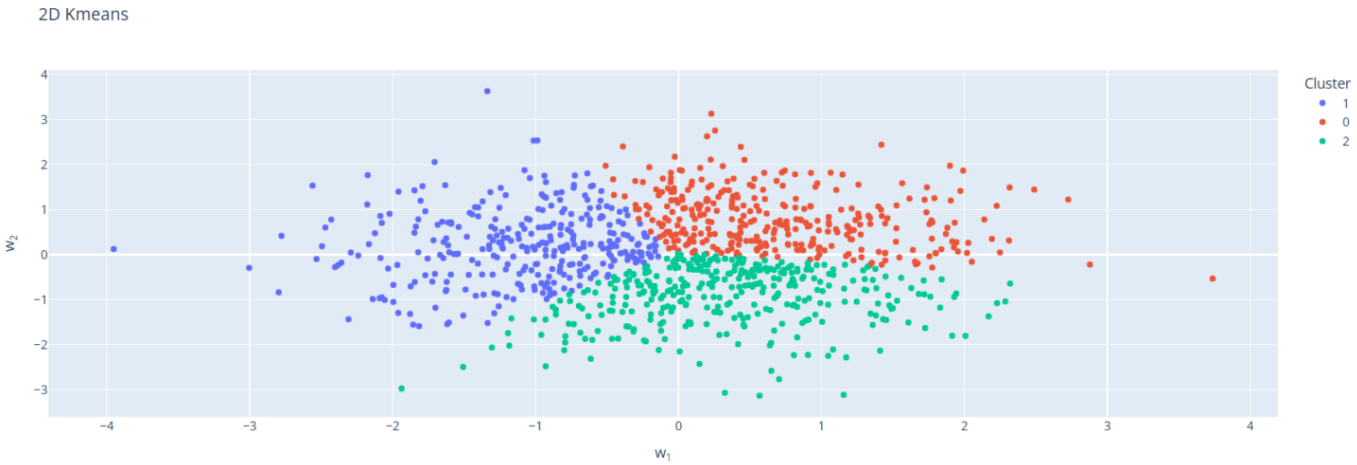


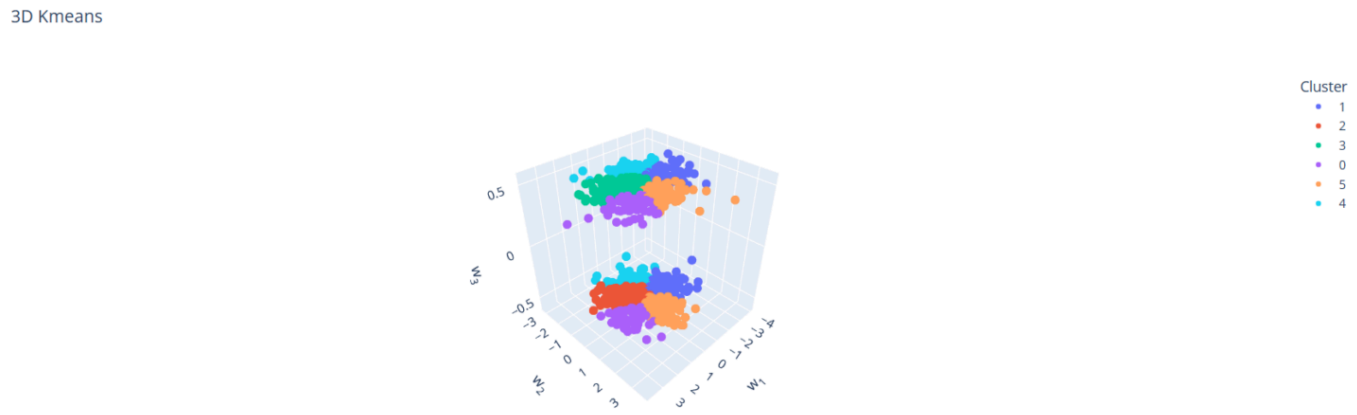
- K Means: We used this technique as a form of exploratory data analysis. Our goal from this was to see what number of clusters work the best by trying k-means on anywhere from 1-10 clusters and finding the elbow point in the graph based on this. Then we used the Silhouette Coefficient and Davies-Bouldin techniques to verify whether the cluster number acquired from k-means was accurate. From the visualizations below, you can tell that k = 3 serves as a pretty good elbow point. Then, we performed the Silhouette Coefficient and Davies Bouldin index and saw that k = 6 was the best for both of those. However, we saw that there was still good reason to try k = 3 based on both visuals, so we still tried that out. We performed k-means on both 3 and 6 clusters and the results are below. Overall, it looks like k-means did a good job of clustering on both.


2D Distortion Scores
distortion
num_clusters
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
1
2
3
4
5
6
7
8
9


Silhouette Coefficient for Different Cluster Counts
Silhouette Score
Number of Clusters
0.345
0.34
0.335
0.33
0.325
0.32
0.315
0.31
2
3
4
5
6
7
8
9


Davies-Bouldin Index for Different Cluster Counts
Davies-Bouldin Score
Number of Clusters
1.25
1.2
1.15
1.1
1.05
1
0.95
0.9
0.85
0.8
2
3
4
5
6
7
8
9





We switched from standardization to normalization as one of our preprocessing techniques because in our scenario, having negative values, for example negative weights or negative ages, doesn't make sense, and normalization keeps everything positive.

ML Models

- K Nearest Neighbor: We used this machine learning model because of its interpretability and for ease of generating meaningful graphics from its results. We supplied this model with our dataset after performing PCA and retaining the three most important features. Then we generated a train/test split with 80% going towards training and 20% going towards testing.

- Decision Trees: We opted to use this model for its ease of interpretability which allowed us to generate graphics that helped to explain its decision making process. In addition, the training of a decision tree model is less resource intensive making it suitable for our application as computing resources are limited. We supplied this model with our dataset after performing PCA on it and retaining the 3 most important features that retain most of the variance.

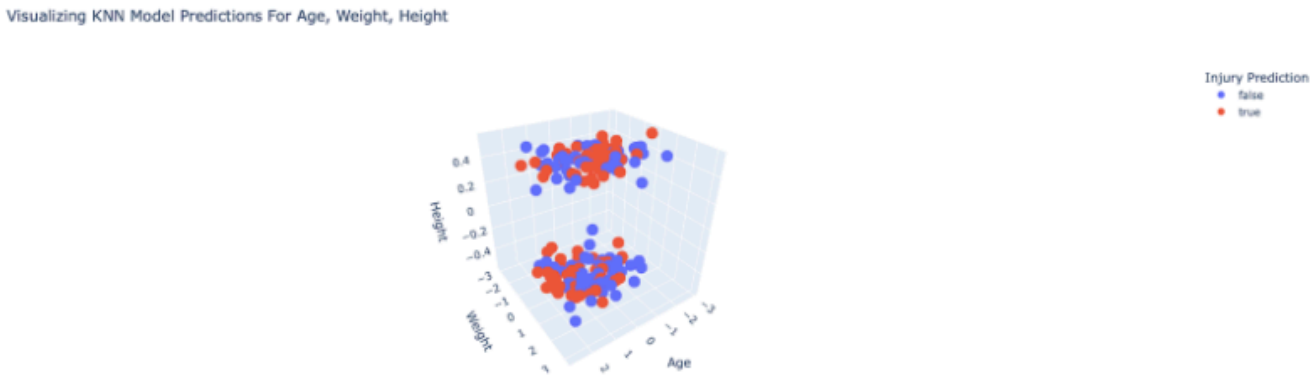
- Logistic Regression: Logistic regression is an algorithm that can be used for binary classification which is our use case here. Additionally, there is the use of probability coupled with the threshold value which makes the binary classification easy. Lastly, the use of the regularization technique benefits in helping reduce underfitting or overfitting to make sure we get a good classification.

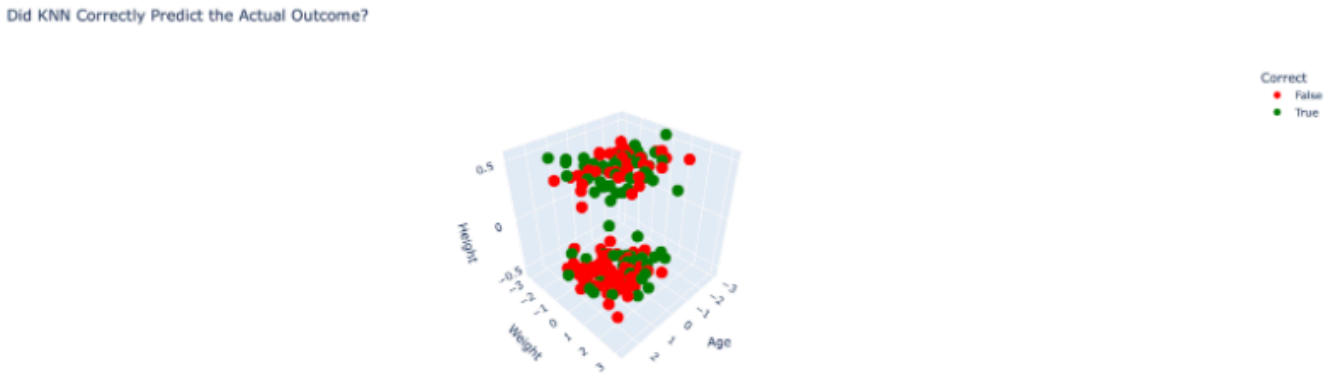
Results and Discussion

KNN

Using KNN to generate a model resulted in the following metrics being produced.

```
Model Accuracy: 0.44
Confusion Matrix:
 [[40 52]
 [60 48]]
Classification Report:
              precision    recall  f1-score   support

           0       0.40      0.43      0.42        92
           1       0.48      0.44      0.46       108

    accuracy                           0.44       200
   macro avg       0.44      0.44      0.44       200
weighted avg       0.44      0.44      0.44       200
```

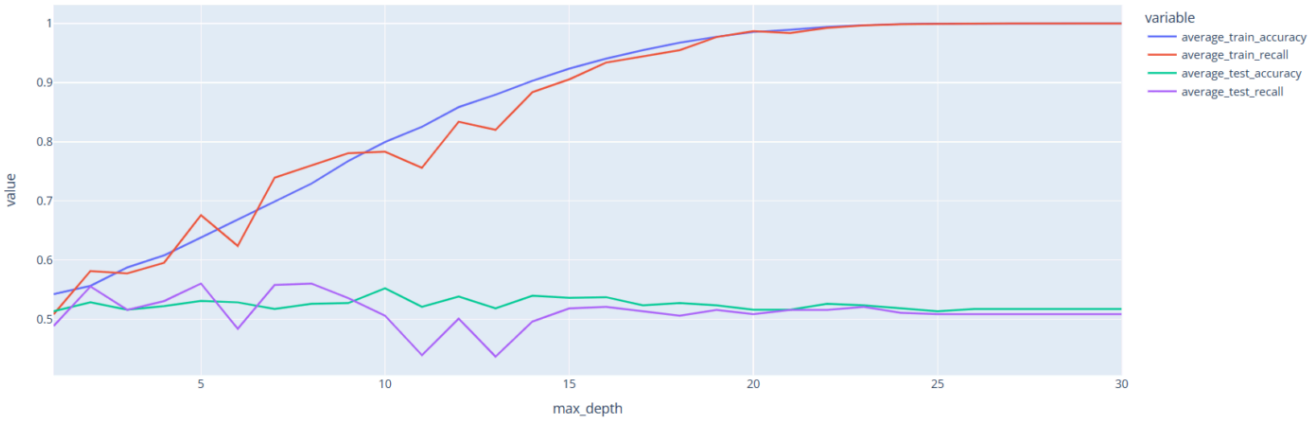




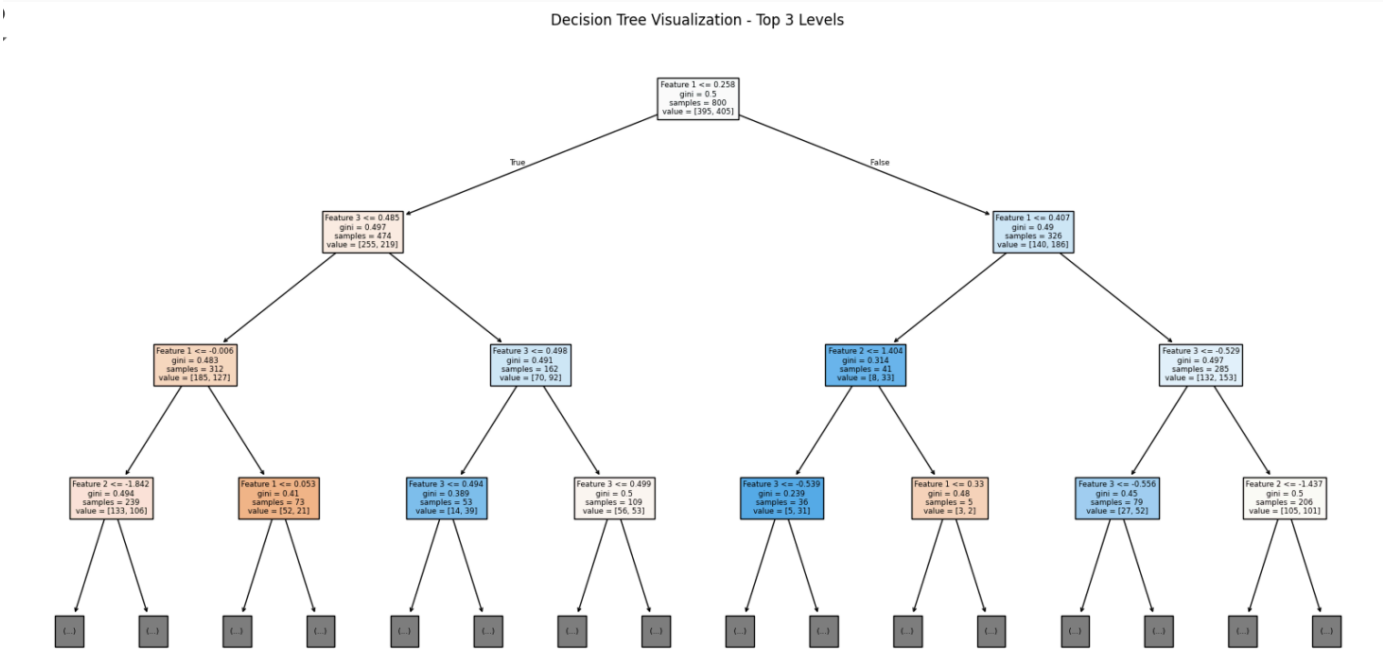
Accuracy involves how many of the predicted outcomes using KNN matched the actual outcomes of the data set. Looking at the visualization of what KNN produced and the visualization of did KNN correctly predict the actual outcome, you can see it didn't do a great job. Furthermore looking at the quantitative metric of 0.44 for accuracy, it further proves the point that KNN performed poorly. Precision measures how many times the true outcome was discovered out of all times the model predicted a true outcome. Recall measures how many times the true outcome was discovered out of all times that the true true values were found and the false false values were found. For both injury likely (true) and injury not likely (false) in this context, precision and recall were very low indicating bad results. F1-score calculates the mean of precision and recall, and since that is also low, it indicates that KNN did a bad job. To sum up, there was generally poor performance across the board when we used the KNN model to classify points by either a low or high likelihood of an injury. We attribute this poor performance due to the high dimensionality of our data. What we discovered here is that KNN overall doesn't do a great job with our dataset and in order to possibly get better results, other avenues must be explored. We will try to perform logistic regression and random forests as they might show better performance for these dimensionality issues.

We changed our last quantitative metric from area under the ROC curve to f1-score because f1-score relates well with the precision and recall we have already calculated.

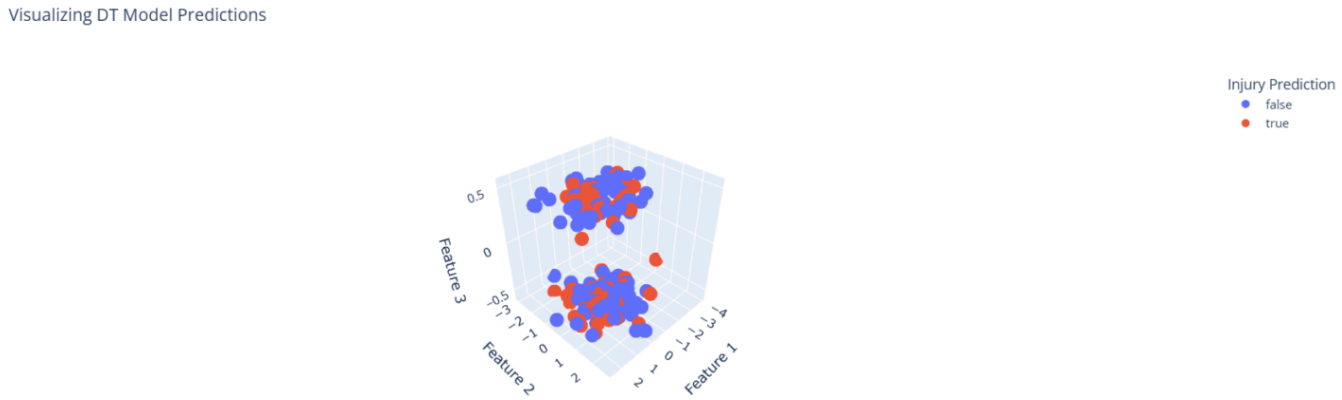
Decision Tree



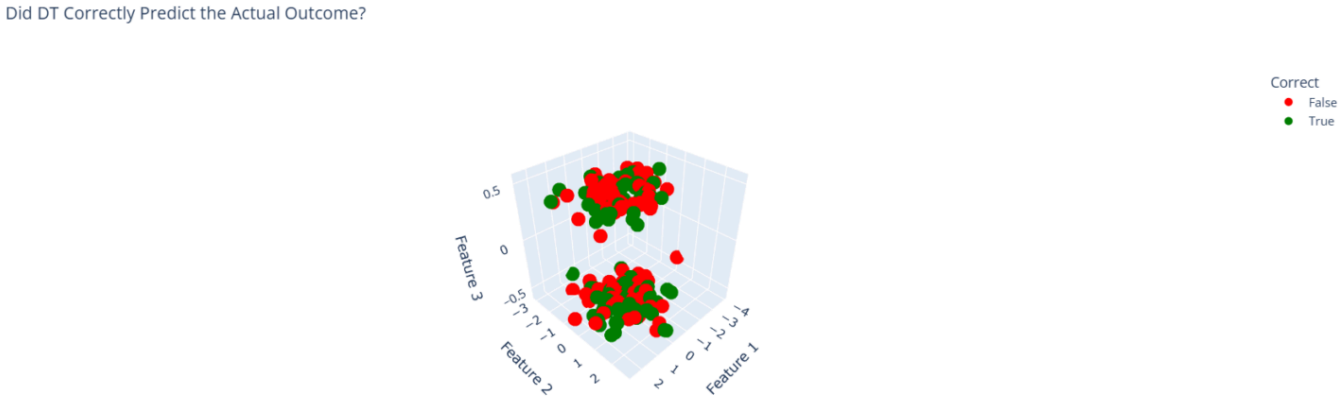
When trying to analyze how much depth we want for the graph, we tried to test out how various levels of depth work while maintaining high recall and accuracy to prevent overfitting. We determined that depth 15 was the best value to use.



This is an image of how the top three levels of the decision tree looked.



This is an image of how the predictions look from the decision trees algorithms. Based on how the visualizations look, it seems like the decision tree model doesn't do a great job in separating the difference between true and false predictions.



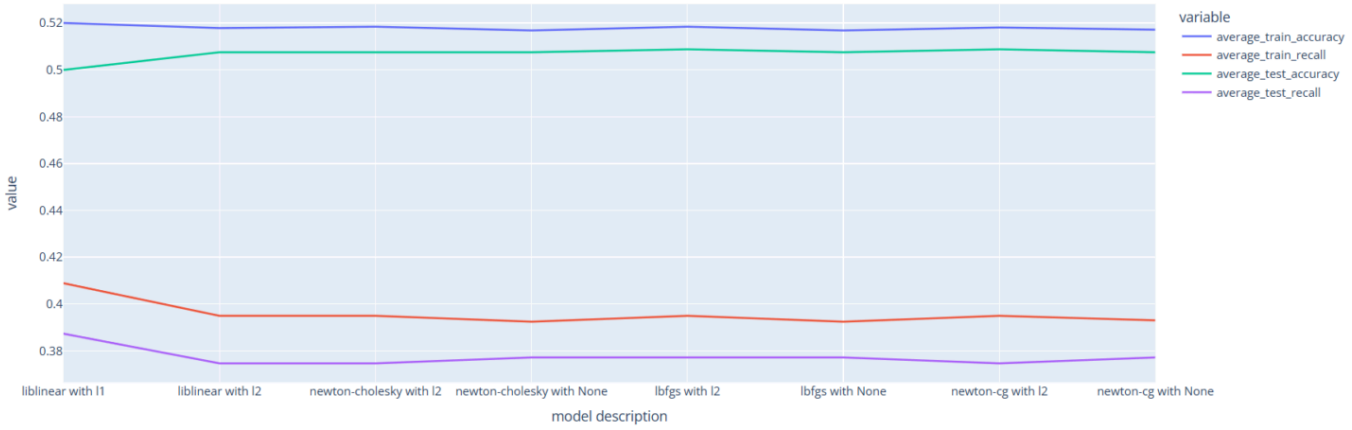
This image is a visual image of the accuracy of the decision tree model predicting the actual outcome, and as can be seen, it didn't do a great job.

```
Model Accuracy: 0.47
Confusion Matrix:
 [[61 44]
 [63 32]]
Classification Report:
              precision    recall  f1-score   support

           0       0.49      0.58      0.53       105
           1       0.42      0.34      0.37        95

    accuracy                           0.47       200
   macro avg       0.46      0.46      0.45       200
weighted avg       0.46      0.47      0.46       200
```

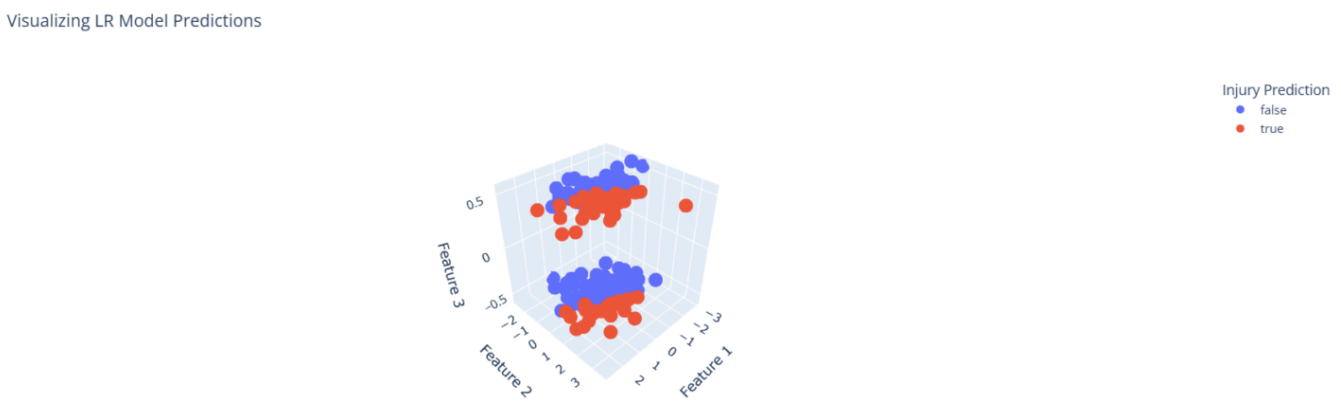
This is the quantitative results for the decision tree model. As can be seen, the accuracy of performing decision trees is 0.47 showing that the results aren't great. This proves the point from the above visuals seen. For both true and false in this context, the precision, recall, and f1-scores were bad all across the board. This overall shows that the decision tree models didn't do a great job in being able to classify whether an injury took place or not.

We attribute this poor performance to many factors, with the main two being the small size of our dataset and poor handling of continuous variables. A small dataset could contribute to poor performance as there are not enough data points to accurately convey patterns to the model so it would have trouble understanding them. In addition, this model is being fed by purely continuous variables which might not have an optimal place to be split at as decision trees will make binary splits to classify information. These nonoptimal splits might result in the loss of information and thus a poor classification performance. Some next steps might be to find a better dataset.

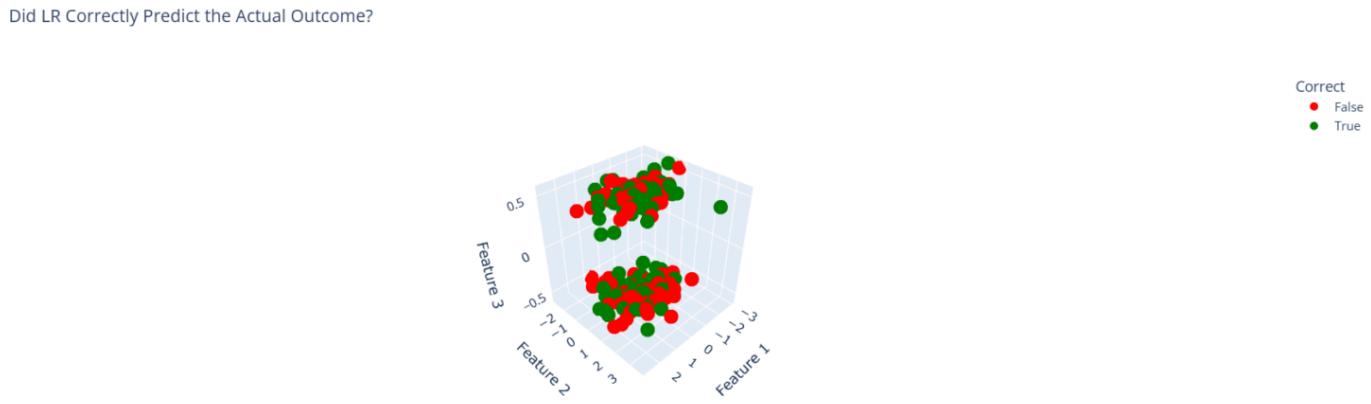
Logistic Regression



We tested out different regularization techniques with different solvers and saw that the liblinear with the l2 regularization combination was the one that yielded the best results as seen from the graph below.



This is an image of how the logistic regression algorithm tries to divide the likelihood of there being an injury or not. Looking at it, it seems like it divided the prediction with a plane as can be seen with the way true and false data points are coupled together.



Looking at whether logistic regression predicted the actual outcome, again it looks like it didn't do a great job.

```
Model Accuracy: 0.48
Confusion Matrix:
 [[63 32]
 [72 33]]
Classification Report:
              precision    recall  f1-score   support

           0       0.47      0.66      0.55        95
           1       0.51      0.31      0.39       105

    accuracy                           0.48       200
   macro avg       0.49      0.49      0.47       200
weighted avg       0.49      0.48      0.46       200
```

This is the quantitative metrics resulting from the logistic regression. Looking at the accuracy, it is 0.48 which again shows really poor performance. Looking at the precision, recall, and f1-score, again there were bad values throughout the various metrics again. This again proves the point as seen from the visual. This proves that the way it tried to divide using a plane didn't do a good job.

There are a few reasons why we think the model didn't work well similar to the problems listed earlier. We believe our dataset doesn't have enough data which could lead to patterns not being correctly established. Also, we believe that there might not be any correlation between features so the logistic

regression model isn't able to properly build correlations to classify things. Some next steps are to possibly find a better dataset.

Comparison of the Three Models

Looking back at all three algorithms, they all did a pretty bad job in predicting the likelihood of injuries. The accuracies were all really low, and precision, recall, and f1-scores were all suboptimal. If we're forced to pick one, picking logistic regression might be the best out of the three as it had the highest accuracy, but still it wasn't good. Picking a better dataset might be a better avenue to take in the future. Some strengths of each of them were that KNN was good for clustering the data points and when datasets are small, logistic regression was good as it was able to build a plane to try to separate data, and decision trees were good because it tried to at each level of the tree try to isolate different differences to try to classify the data. Some limitations were that KNN is bad for high dimensional data which we had here as PCA didn't reduce dimensions effectively. Decision trees are bad in the sense where too much depth leads to too much complexity, and logistic regression didn't work as we had uncorrelated data. Some tradeoffs are that logistic regression is good for data that is prone to overfitting or underfitting as regularization is possible, KNN is good for low dimensional small datasets, and decision trees are good for datasets that are easy to classify with different types of distinctions.

Next Steps

For future steps, as aforementioned, we would like to use a different dataset. We would like to pick a dataset that has more datapoints so that it picks up on more meaningful relationships which would help in better classification of the data.

Gantt Chart

Name:	Task Assignments:
Tilman Gromme	Proposal, EDA, Model Comparison, Midterm, Tuning, Final
Zaid Mohammed	Data Collection, Proposal, Feature Engineering, Midterm, Final
Alex Kim	Proposal, Research, EDA, Midterm, Final Implementation, Tuning, Final
Alex Thummalapalli	Proposal, Feature Engineering, Model Comparison, Final
Muadh George	Proposal, Model Comparison, Final Implementation

Contribution Table

Name:	Task Assignments:
Tilman Gromme	Discussing results, update website, visualize data, update Github, update website, work on video, compare algorithms
Zaid Mohammed	Discussing methods, Discussing results, update website, visualize data, update Github, update website, work on video, compare algorithms
Alex Kim	Implementing logistic regression, visualize data, quantitative results
Alex Thummalapalli	Implementing decision tree, visualize data, quantitative results
Muadh George	Discussing methods, discuss results, update website, work on video

References

[1] H. Van Eetvelde, L. D. Mendonça, C. Ley, R. Seil, and T. Tischer, “Machine learning methods in sport injury prediction and prevention: a systematic review,” *Journal of Experimental Orthopaedics*, vol. 8, no. 1, Apr. 2021, doi: https://doi.org/10.1186/s40634-021-00346-x.

[2] A. Amendolara et al., “An overview of machine learning applications in sports injury prediction,”

Cureus, Sep. 2023. doi: https://doi.org/10.7759%2Fcureus.46170

[3] Majumdar, A., Bakirov, R., Hodges, D. et al. Machine Learning for Understanding and Predicting Injuries in Football. Sports Med - Open 8, 73 (2022). https://doi.org/10.1186/s40798-022-00465-4

[4] tkunzler, “Injury Prediction + EDA 📊 | ENG / PT-BR,” Kaggle.com, Mar. 13, 2024. https://www.kaggle.com/code/tkunzler/injury-prediction-eda-eng-pt-br