Machine Learning I
Final Term Project

Michael Spano
Virginia Tech

December 7, 2024

Contents

List of Figures

List of Tables

Abstract

This project applies machine learning techniques to a real-world dataset to explore regression, classification, and clustering methods. The analysis includes data pre-processing, feature selection, and hyperparameter optimization to find optimal models. Regression models were evaluated based on Mean Square Error, while classification models underwent a grid search process to determine the best performer, achieving a test accuracy of 0.95 for a post-pruned decision tree. Clustering analysis revealed no clear or meaningful clusters, but association rule mining provided some insight into potential patterns. These findings demonstrate the effectiveness of applying machine learning techniques gained from the course on a real world dataset.

Chapter 1

Introduction

This report discusses the final term project for the Machine Learning I course at Virginia Tech instructed by Dr. Reza Jafari. The project sets out to enhance understanding of machine learning techniques on a real dataset, and then formally share progress and results.

The objectives of the study are to conduct exploratory data analysis and feature selection, regression, classification, and clustering analysis on the dataset. Regression aims to minimize the Mean Square Error on the test set. Classification analysis employs many different techniques with hyperparameter grid searching, ultimately suggesting the optimal classification model for the dataset. Clustering and association seek to reveal patterns among the data that are not inherently obvious. Most of the analysis performed in this project were taken from the course, supplemented with additional research and experimentation.

The structure of this report follows this introduction with a description of the dataset. It then discusses the pre-processing portion, including data cleaning, feature selection, and encoding. Subsequent sections share results and findings for regression, classification, and clustering. The paper concludes with a summary, lessons learned, and recommendations for future works.

Chapter 2

Description of the Dataset

The selected data set is ‘120 Years of Olympic History’, provided from Kaggle and linked in the references. It contains data on the Olympic Games from Athens 1896 to Rio 2016, with the following features:
Numerical:

- age
- height
- weight

Categorical:

- id
- name
- sex
- country (team)
- country 3-letter code
- game’s year and season
- host city
- sport
- event
- medal

Each observation is the performance of an athlete in an event that scores with a gold, silver, bronze, or no medal. There are a total of 271,116 observations.

Regression analysis:
The dependent variable is weight, with the independent variables being height, age, year, sex, team, sport, and event.

Classification:

The target variable is sex, with the independent variables being weight, height, age, year, team, and sport.

The decision for which features are independent and dropped is explained in the following chapter.

Importance in Industry:
This dataset analyzes athletes and how well they perform in the Olympics, a highly desired area of entertainment and sports. There is a plethora of machine learning being applied to sports currently, including the Olympics itself, NFL, NBA, MLB, and several soccer leagues.

Chapter 3

Phase I - Data Preprocessing

3.1 Filling Missing Values

The data had NA values in 4 columns: age, height, weight, and Medals. Since age, height, and weight are numerical attributes, the NA values were filled with the column mean. The Medal attribute was substituted with the label ‘No Medal’ for NA values.

3.2 Data Encoding

This section describes the encoding technique for each feature. This was performed before any feature selection analysis.

- **ID** - Dropped. Has no influence on the data.
- **Name** - Dropped. An athlete’s name does not influence the data, and it would be difficult to encode such a large variety of labels.
- **Sex** - Boolean encoded an renamed as ‘**is_male**’. 1 for males and 0 for females.
- **Age** - Left as is because it is a floating value.
- **Height** - Left as is because it is a floating value.
- **Weight** - Left as is because it is a floating value.
- **Team** - Target Encoding for regression & classification. Frequency encoded for clustering. One-hot encoding would balloon the dimensionality of the data.

- **NOC** - Dropped, it is identical to Team.
- **Games** - Dropped, expressed by Year and Season.
- **Year** - Left as is because it is an integer value.
- **Season** - Boolean encoded an renamed as '**is_summer**'. 1 for Summer and 0 for Winter.
- **City** - Dropped, it is highly correlated to Year and Season.
- **Sport** - Target Encoding for regression & classification. Frequency encoded for clustering. One-hot encoding would balloon the dimensionality of the data.
- **Event** - Target Encoding for regression & classification. Frequency encoded for clustering. One-hot encoding would balloon the dimensionality of the data.
- **Medal** - Label encoded with the following values: {Gold: 3, Silver: 2, Bronze: 1, No Medal: 0}.

3.3 Scaling

All data was standardized.

3.4 Feature Selection

This section describes the feature selection process for the dataset.

3.4.1 Regression Analysis

For regression analysis, the target feature is **Weight**.

Principal Component Analysis (PCA), backward stepwise regression, and random forest feature importance were used to identify which features to select and drop.

Figure 3.1 shows that only seven features are needed to explain 95% of the variance in the data.

Figure 3.2 illustrates the important features of the data set in regards

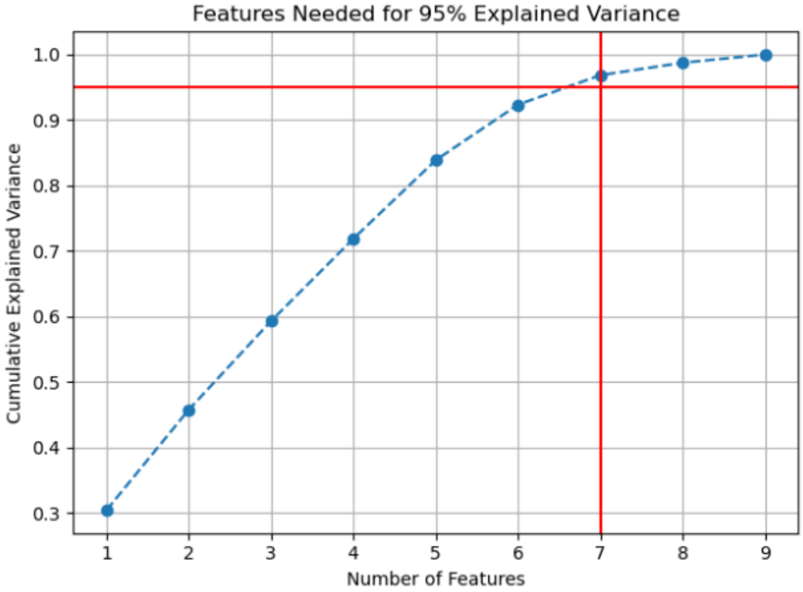


Figure 3.1: PCA for Regression

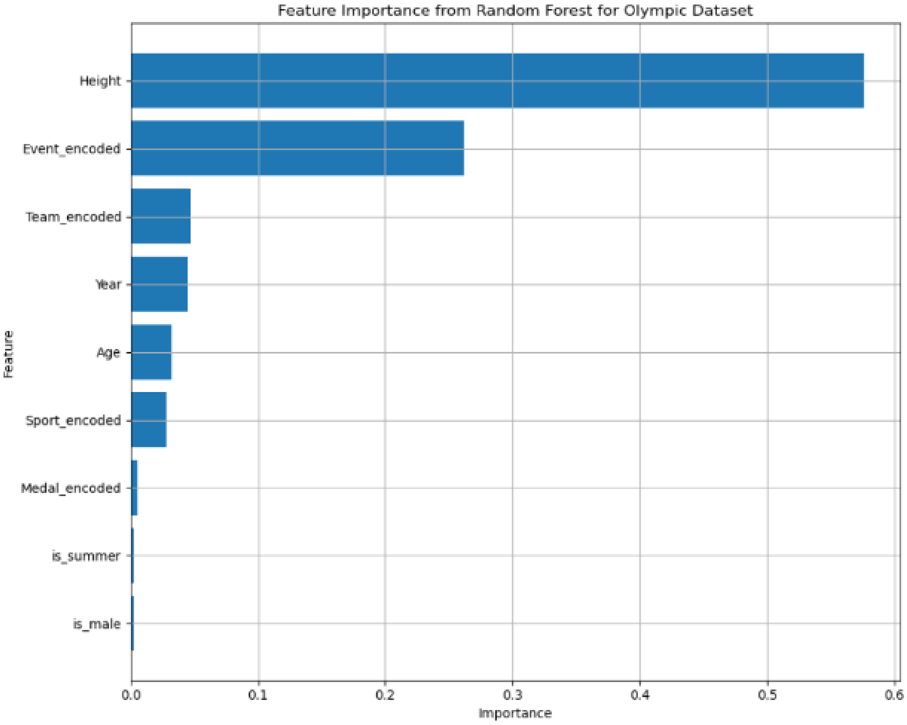
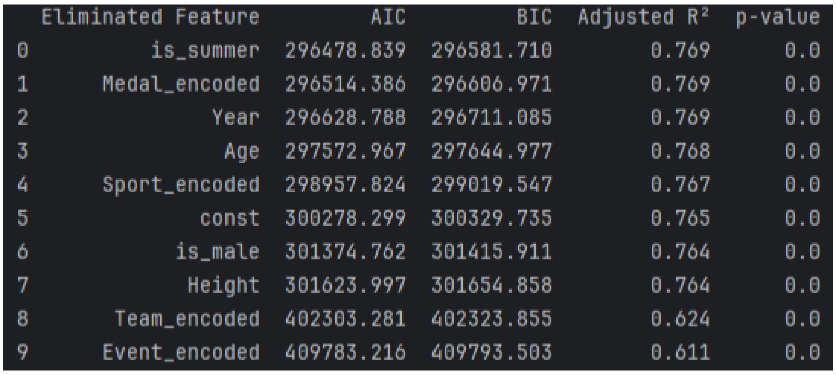


Figure 3.2: RF for Regression Feature Importance



```
  Eliminated Feature         AIC         BIC  Adjusted R²  p-value
0          is_summer  296478.839  296581.710        0.769      0.0
1      Medal_encoded  296514.386  296606.971        0.769      0.0
2               Year  296628.788  296711.085        0.769      0.0
3                Age  297572.967  297644.977        0.768      0.0
4      Sport_encoded  298957.824  299019.547        0.767      0.0
5              const  300278.299  300329.735        0.765      0.0
6            is_male  301374.762  301415.911        0.764      0.0
7             Height  301623.997  301654.858        0.764      0.0
8       Team_encoded  402303.281  402323.855        0.624      0.0
9      Event_encoded  409783.216  409793.503        0.611      0.0
```

Figure 3.3: Backwards Stepwise for Regression

to the target, **weight**.

Figure 3.3 shows a table that lists from top to bottom the values eliminated in order of highest p-values. The upper rows are the least significant features and are shown by the non-decreasing adjusted R^2.

Considering the PCA, RF feature importance, and backwards stepwise regression, the 7 selected features are [**height, age, year, sex (is_male), team, sport, event**]. The 2 dropped features are [**season, medal**].

Figure 3.4 and Figure 3.5 are heatmaps showing the correlation and covariance heatmaps for the selected features.

3.4.2 Classification Analysis

For classification analysis, the target feature is **Sex**.

Similarly to the previous section, PCA, backward stepwise regression, and random forest feature important were used to identify which features to select and drop.

It should be noted that initially, **Event** was included and dominated any feature importance. A very simple logistic regression model was able to achieve 99% accuracy on test data. Upon further inspection, it was due to the event being a ‘Women’s’ or ‘Men’s’, such as ‘Men’s Basketball’ or ‘Women’s 100m’. Because of this, **Event** was dropped from feature selection.

Figure 3.6 shows that only 6 features are needed to explain 95% of the

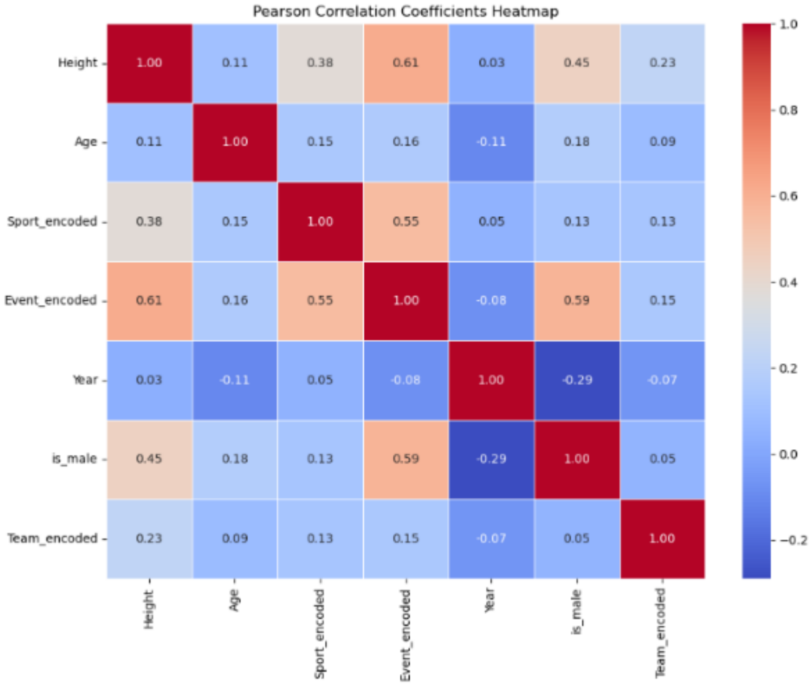


Figure 3.4: Pearson Correlation Coefficient Heatmap for Regression Features

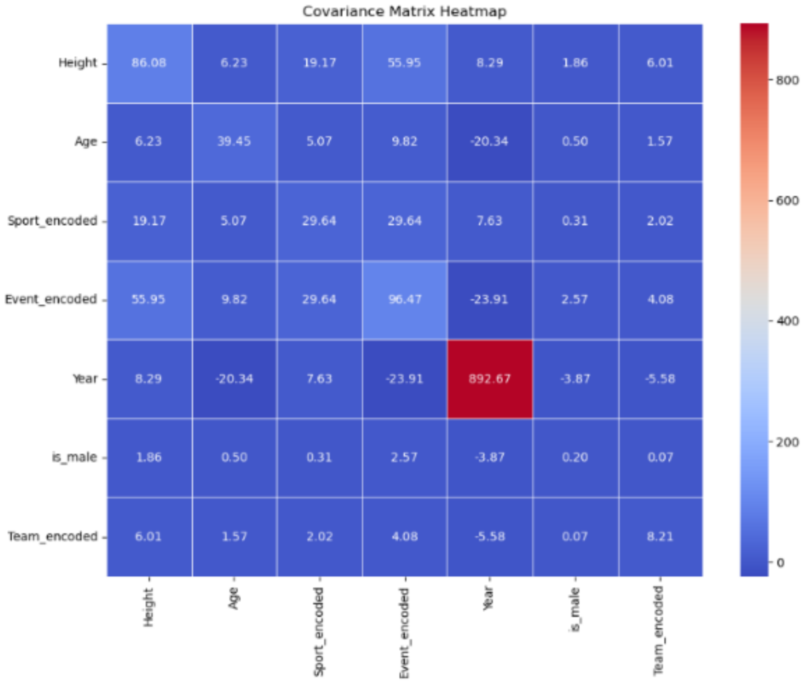


Figure 3.5: Covariance Heatmap for Regression Features

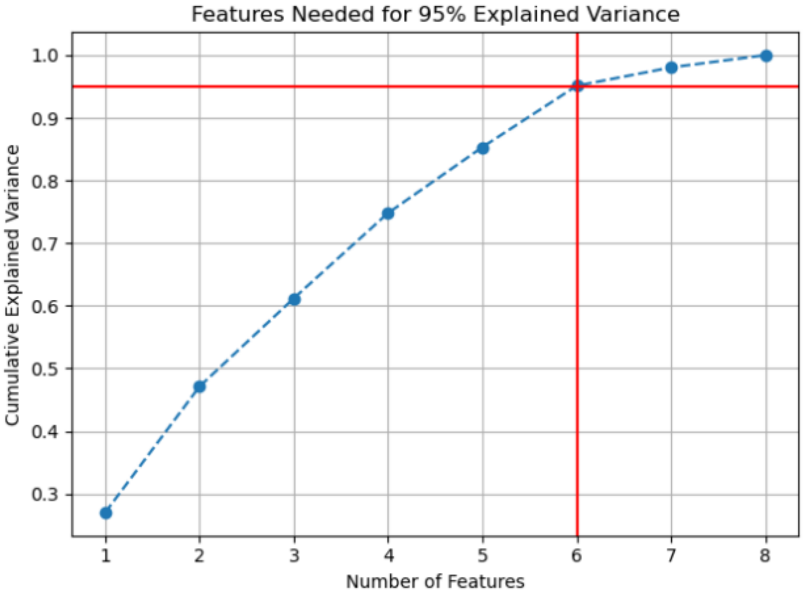


Figure 3.6: PCA for Classification

variance in the data.

Figure 3.7 illustrates the important features of the data set in regards to the target, **Sex**.

Figure 3.8 shows a table that lists from top to bottom the values eliminated in order of highest p-values. The upper rows are the least significant features and are shown by the non-decreasing adjusted R^2.

Considering the PCA, RF feature importance, and backwards stepwise regression, the 6 selected features are [**weight, height, age, year, team, sport**]. The 2 dropped features are [**season, medal**].

Figure 3.9 and Figure 3.10 are heatmaps showing the correlation and covariance heatmaps for the selected features.

3.5 Balancing

For **classification analysis**, the target value of **Sex** is unbalanced as shown in Figure 3.11. The data was balanced using Synthetic Minority Oversampling Technique (SMOTE), and is shown in Figure 3.12.

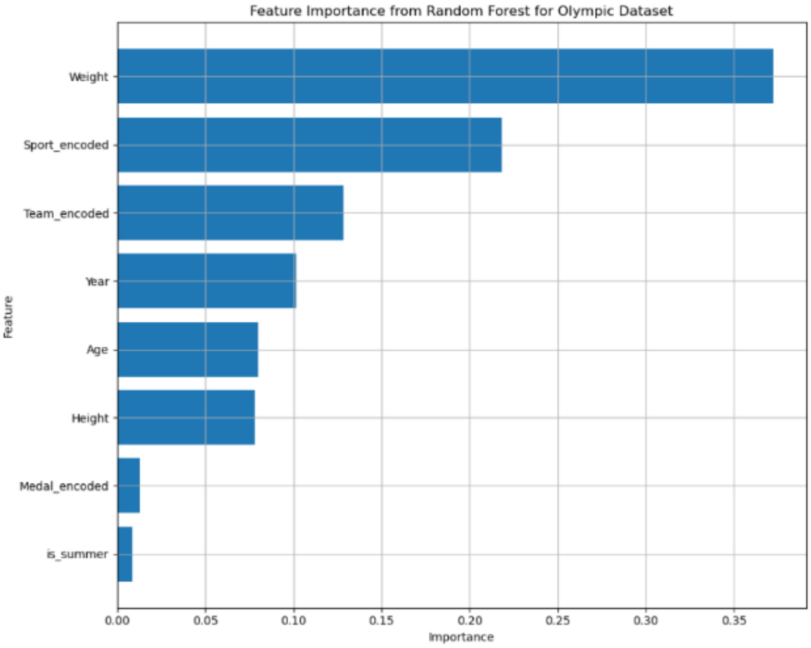
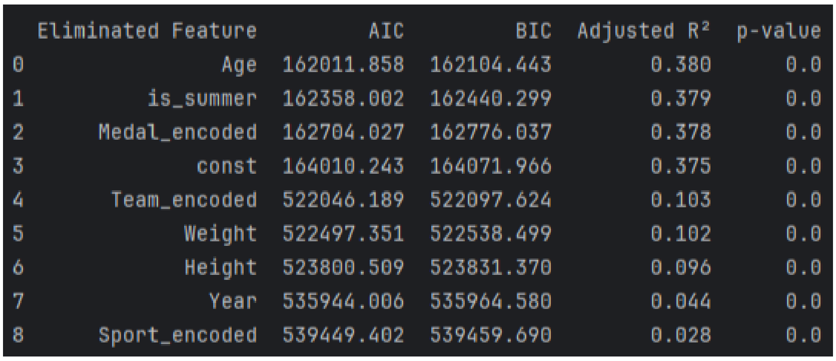


Figure 3.7: RF for Classification Feature Importance



```
  Eliminated Feature         AIC         BIC  Adjusted R²  p-value
0                Age  162011.858  162104.443        0.380      0.0
1          is_summer  162358.002  162440.299        0.379      0.0
2      Medal_encoded  162704.027  162776.037        0.378      0.0
3              const  164010.243  164071.966        0.375      0.0
4       Team_encoded  522046.189  522097.624        0.103      0.0
5             Weight  522497.351  522538.499        0.102      0.0
6             Height  523800.509  523831.370        0.096      0.0
7               Year  535944.006  535964.580        0.044      0.0
8      Sport_encoded  539449.402  539459.690        0.028      0.0
```

Figure 3.8: Backward Stepwise for Classification

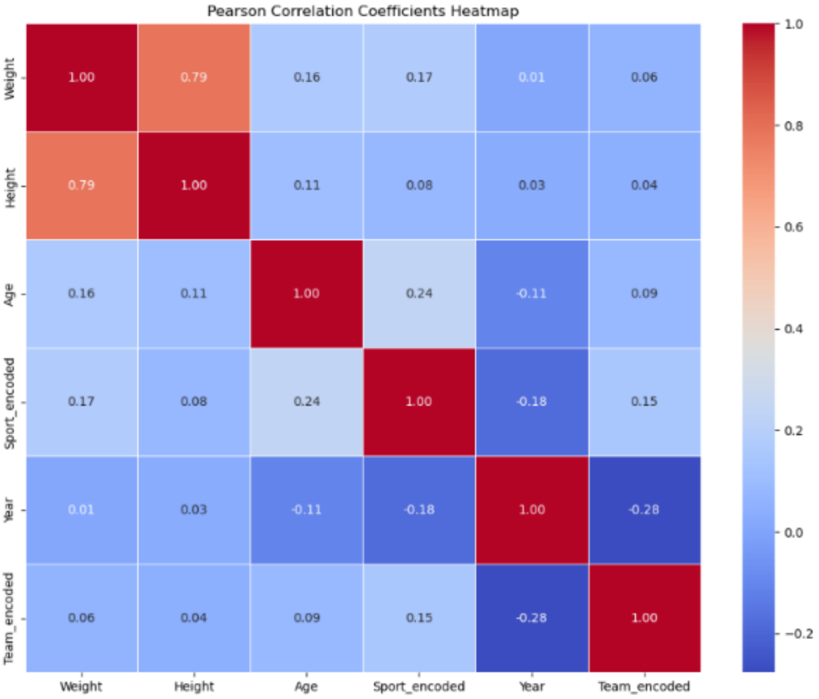


Figure 3.9: Pearson Correlation Coefficient Heatmap for Classification Features

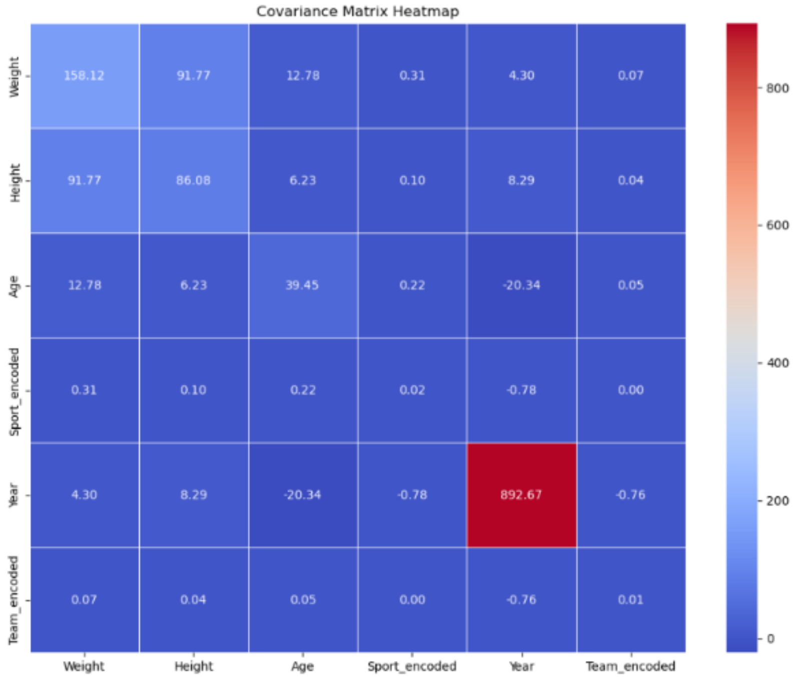


Figure 3.10: Covariance Heatmap for Classification Features

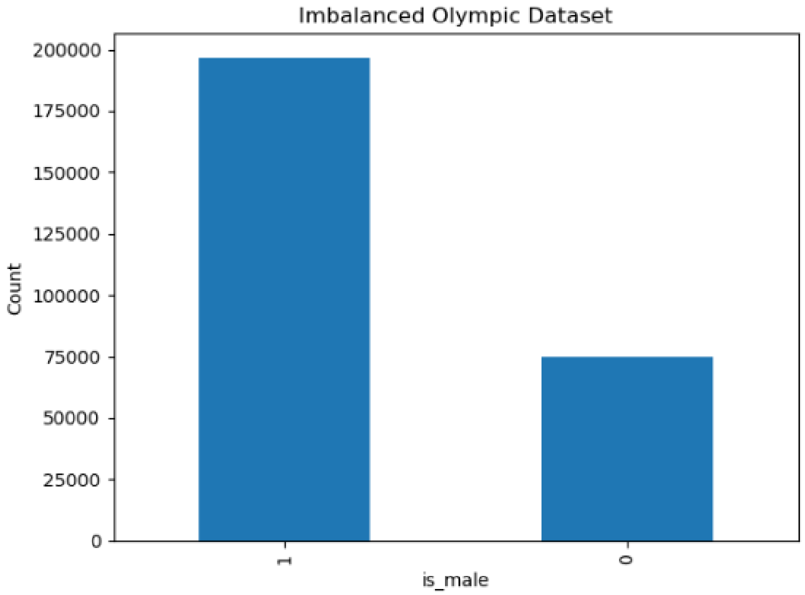


Figure 3.11: Unbalanced Data for Target Sex

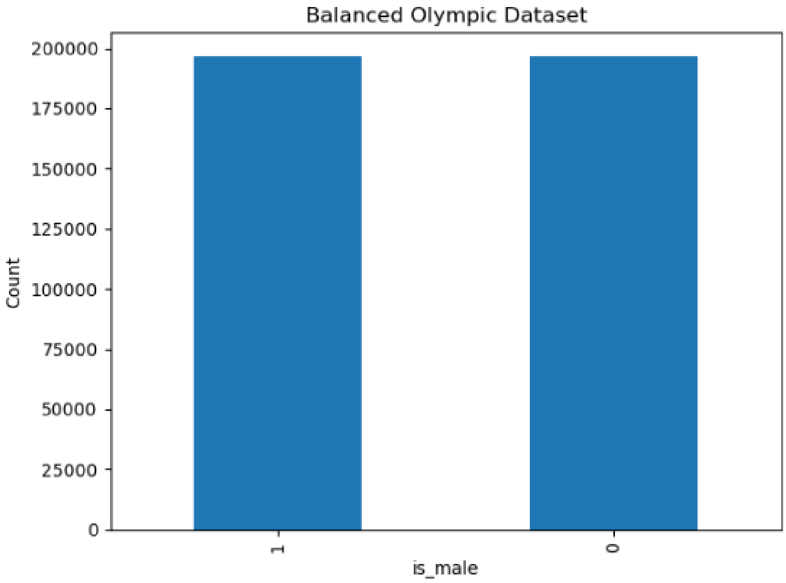


Figure 3.12: Balanced Data for Target Sex

Chapter 4

Phase II - Regression Analysis

A multiple linear regression model was developed to predict the target variable **weight**. A grid search was conducted to find the optimal degree. This is shown in Figure 4.1, with an **optimal degree of 5**, having a Root Mean Square Error of 0.425 on the standardized test data.

Using an OLS model with a degree of 5 achieved an MSE of 29.011 on 54,224 test points. The first 300 predicted vs test points are shown in Figure 4.2. The 95% confidence interval is shown for the first 100 points in Figure 4.3.

The OLS Model Summary is shared in Figure 4.4. The adjusted R^2 value of .822 suggests that 82.2% of the variance in target **weight** is explained by the model. The high F-statistic suggests the model is statistically significant, supported by the extremely low p-value. The model's independent variables have a strong relationship with the dependent variable. All of this affirms that the model sufficiently predicts the **weight** given [**height, event, team, sport, age, year, sex**].

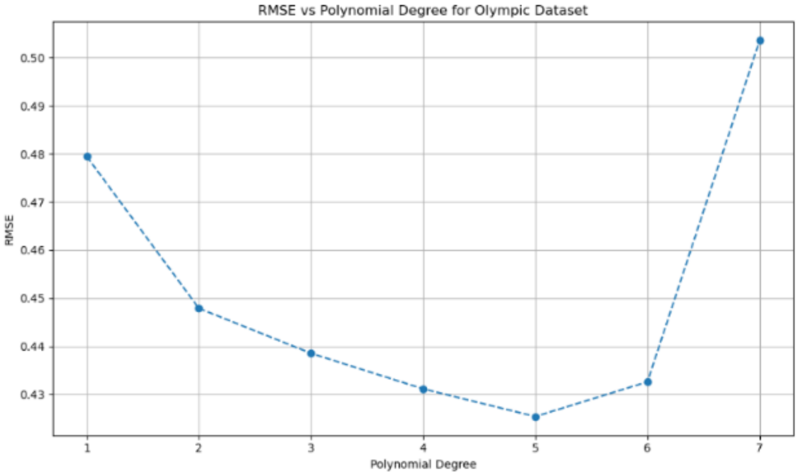


Figure 4.1: Optimal Degree for Multiple Linear Regression

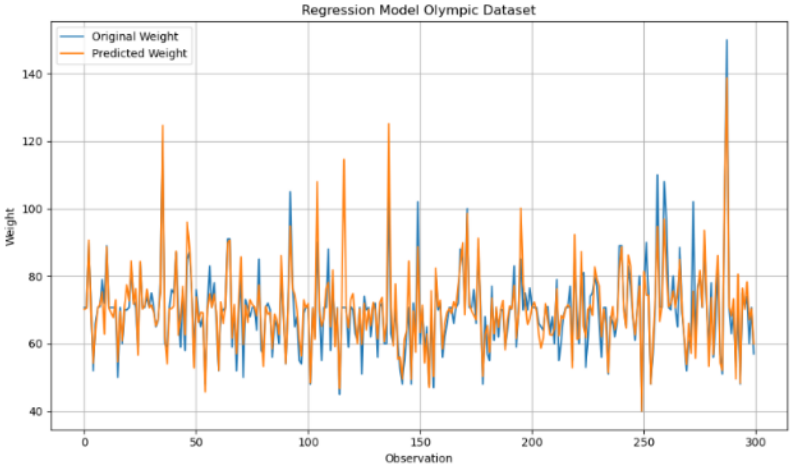


Figure 4.2: Test Prediction for Multiple Linear Regression

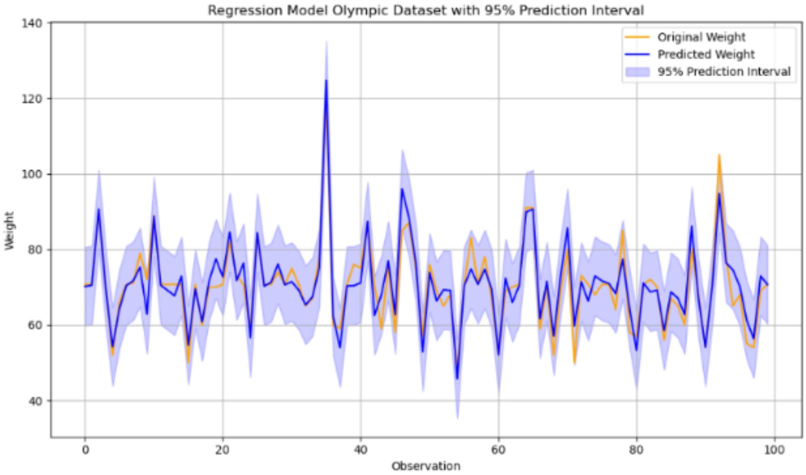
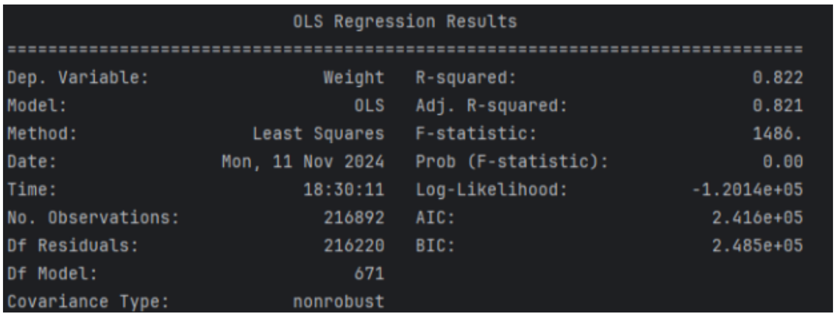


Figure 4.3: 95% CI for Multiple Linear Regression



```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Weight   R-squared:                       0.822
Model:                            OLS   Adj. R-squared:                  0.821
Method:             Least Squares   F-statistic:                     1486.
Date:                Mon, 11 Nov 2024   Prob (F-statistic):               0.00
Time:                        18:30:11   Log-Likelihood:             -1.2014e+05
No. Observations:              216892   AIC:                         2.416e+05
Df Residuals:                  216220   BIC:                         2.485e+05
Df Model:                         671
Covariance Type:            nonrobust
```

Figure 4.4: OLS Regression Results Summary

Chapter 5

Phase III - Classification Analysis

This phase discusses the classification section on the binary target **Sex** using different classification techniques. The independent variables are [**weight, height, age, year, team, sport**]. Every model was trained using Stratified 5-K Fold Cross Validation. After walking through every model, this section finishes with a comparison of all models and suggests the best approach for classifying on the chosen dataset.

5.1 Logistic Regression

A grid search was performed on the parameter grid outlined in Table 5.1. The best model was selected on which set of parameters had the highest accuracy on the testing set.

Parameter	Values
$penalty$	$l1, l2, elasticnet$
C	np.logspace(-3, 1, 10)
$l1\_ratio$	np.linspace(0, 1, 30)

Table 5.1: Param Grid for Logistic Regression

The best hyperparameters for Logistic Regression on the parameter grid were:

- **penalty:** `l2`
- **C:** `0.00774`
- **l1_ratio:** `0`

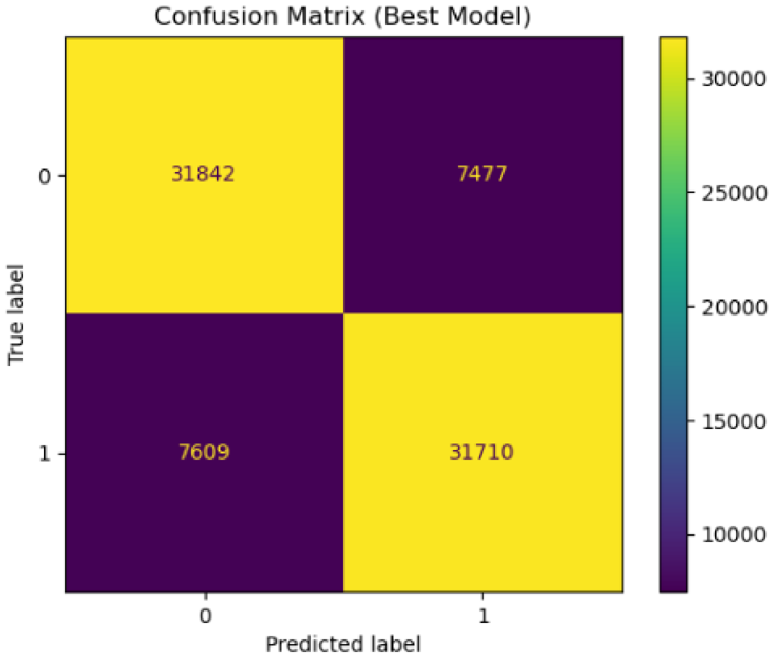


Figure 5.1: Confusion Matrix for Logistic Regression

The results of this model are described in Table 5.2. The confusion matrix is shown in Figure 5.1 and the ROC curve in Figure 5.2.

Accuracy	.81
Recall	.81
Precision	.81
F1 Score	.81
Specificity	.81

Table 5.2: Results for Logistic Regression

5.2 Decision Tree

Both pre-pruning and post-pruning were performed for the decision tree application.

For pre-pruning, grid search was performed on the parameter grid outlined in Table 5.3.

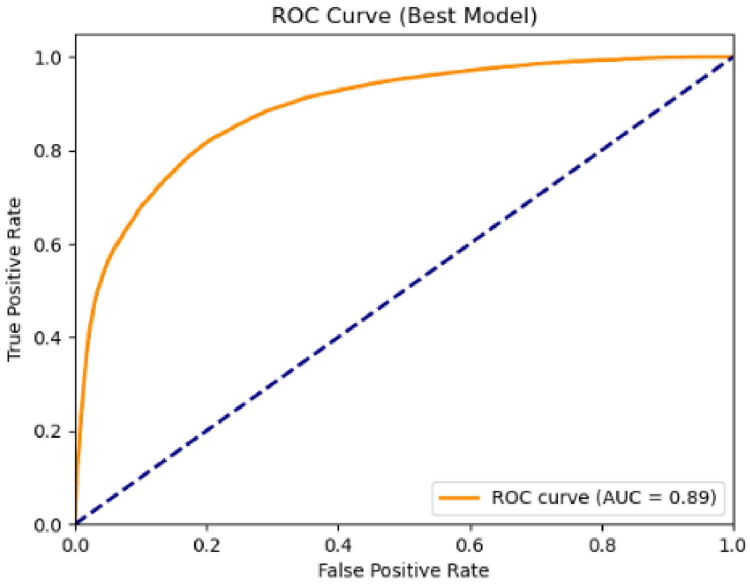


Figure 5.2: ROC Curve for Logistic Regression

Parameter	Values
max_depth	5, 10, 20, 30
$min\_samples\_split$	50, 100, 200
$min\_samples\_leaf$	10, 50, 100
$criterion$	gini, entropy
$splitter$	best, random
$max_{f}eatures$	sqrt, log2

Table 5.3: Param Grid for Decision Tree

The best hyperparameters for pre-pruning on the parameter grid were:

- **max_depth:** 30
- **min_samples_split:** 50
- **min_samples_leaf:** 10
- **criterion:** entropy
- **splitter:** best
- **max_features:** sqrt

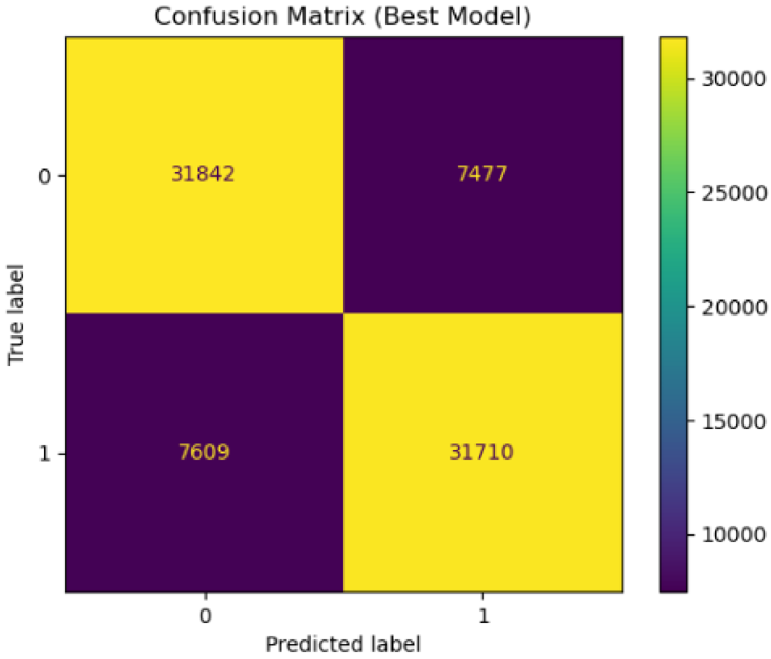


Figure 5.3: Confusion Matrix for Decision Tree

and achieved a test accuracy of **.90**.

For post-pruning, the Cost Complexity Function was optimized, with the best **alpha** being **2.11e-06**. Test accuracy was **.95**.

Therefore, the best and selected model was the post-pruned decision tree.

The results of this model are described in Table 5.4. The confusion matrix is shown in Figure 5.3 and the ROC curve in Figure 5.4.

Accuracy	.95
Recall	.95
Precision	.95
F1 Score	.95
Specificity	.95

Table 5.4: Results for Decision Tree

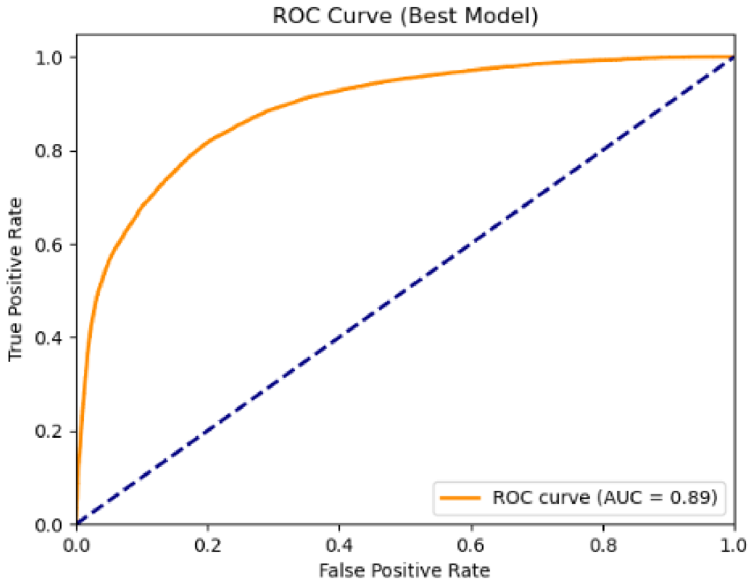


Figure 5.4: ROC Curve for Decision Tree

5.3 KNN

A KNN model was trained and evaluated on values of K from 1 to 30. The MSE of these values are displayed in Figure 5.5.

Interestingly, the best K is 1, selecting simply the closest neighbor.

The results of this model are described in Table 5.5. The confusion matrix is shown in Figure 5.6 and the ROC curve in Figure 5.7.

Accuracy	.95
Recall	.92
Precision	.97
F1 Score	.95
Specificity	.97

Table 5.5: Results for KNN

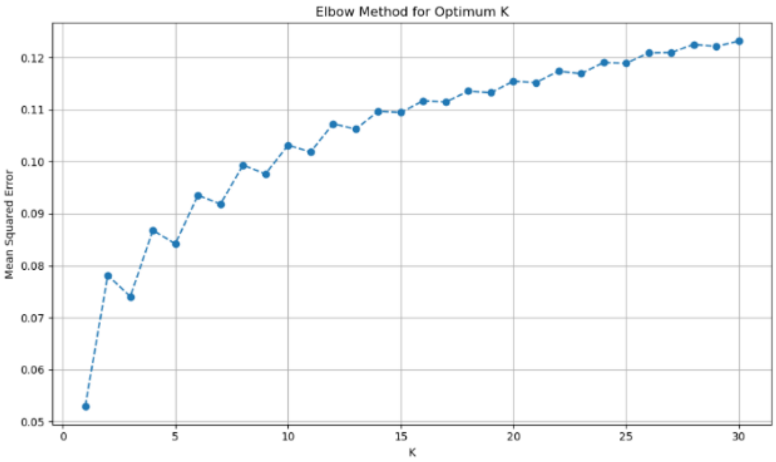


Figure 5.5: MSE for Different Values of K

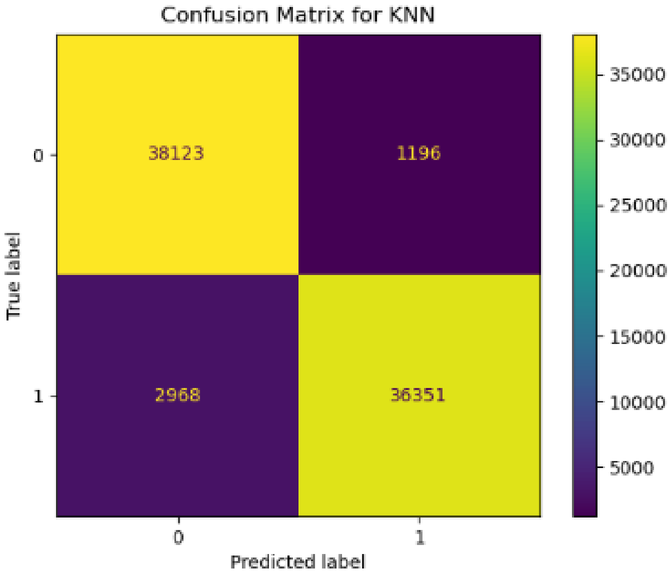


Figure 5.6: Confusion Matrix for KNN

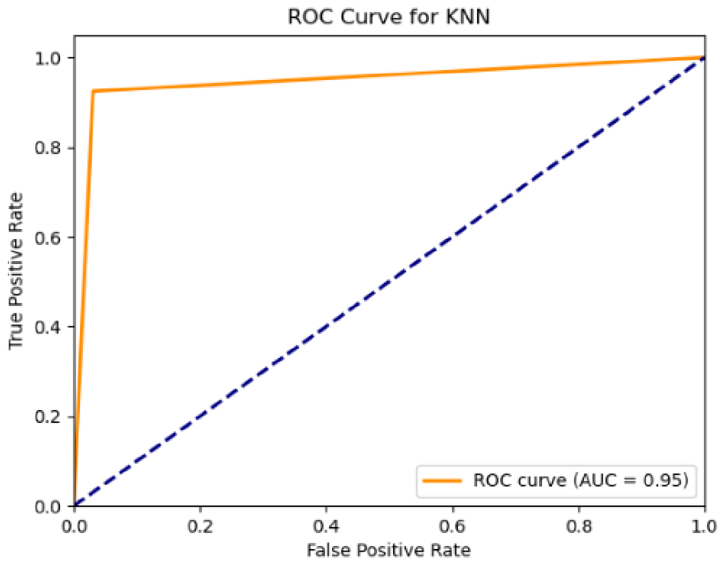


Figure 5.7: ROC Curve for KNN

5.4 Naive Bayes

A simple Naive Bayes was trained, with no hyperparameter optimization.

The results of this model are described in Table 5.6. The confusion matrix is shown in Figure 5.8 and the ROC curve in Figure 5.9.

Accuracy	.80
Recall	.80
Precision	.80
F1 Score	.80
Specificity	.81

Table 5.6: Results for Naive Bayes

5.5 Support Vector Machine

Linear, polynomial, and radial base function(rbf) kernels were employed for the support vector machine (SVM).

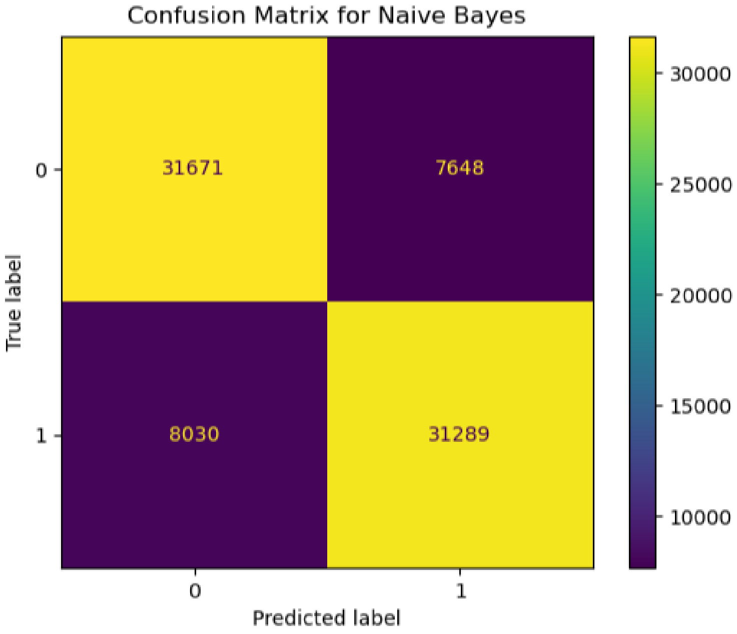


Figure 5.8: Confusion Matrix for Naive Bayes

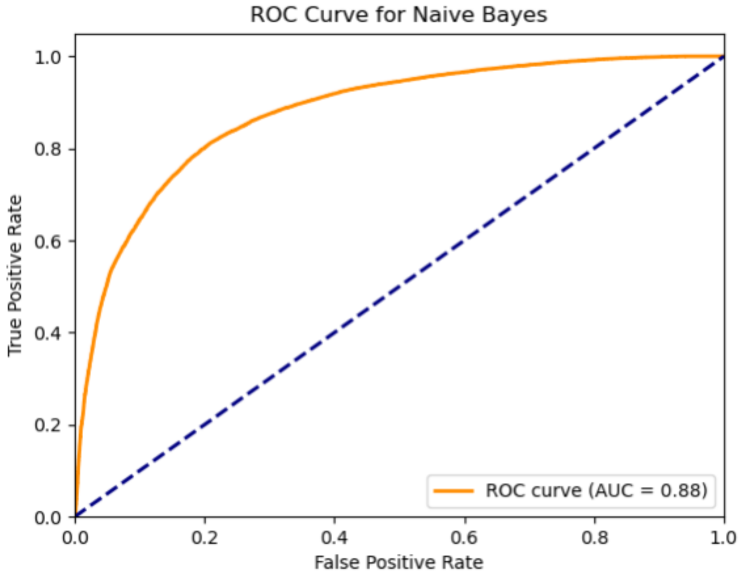


Figure 5.9: ROC Curve for Naive Bayes

5.5.1 Linear Kernel

A grid search was performed on the parameter grid outlined in Table 5.7.

Parameter	Values
C	0.1, 1, 10, 100

Table 5.7: Param Grid for Linear Kernel SVM

The best hyperparameters for Linear Kernel SVM on the parameter grid were:

- **C:** `100`

with a test accuracy of 0.81.

5.5.2 Polynomial Kernel

A grid search was performed on the parameter grid outlined in Table 5.8.

Parameter	Values
C	0.1, 1, 10, 100
$degree$	2, 3, 4, 5

Table 5.8: Param Grid for Polynomial Kernel SVM

The best hyperparameters for Polynomial Kernel SVM on the parameter grid were:

- **C:** `10`
- **l1_ratio:** `3`

with a test accuracy of 0.80.

5.5.3 RBF Kernel

A grid search was performed on the parameter grid outlined in Table 5.9.

The best hyperparameters for RBF Kernel SVM on the parameter grid were:

Parameter	Values
C	0.1, 1, 10, 100
gamma	'scale', 'auto', 0.01, 0.1, 1, 10

Table 5.9: Param Grid for RBF Kernel SVM

- **C:** `1`
- **gamma:** `1`

with a test accuracy of 0.86.

Therefore, the RBF Kernel SVM was selected since it had the highest test accuracy.

The results of this model are described in Table 5.10. The confusion matrix is shown in Figure 5.10 and the ROC curve in Figure 5.11.

Accuracy	.86
Recall	.85
Precision	.86
F1 Score	.85
Specificity	.86

Table 5.10: Results for SVM

5.6 Neural Network (MLP)

A Multi Layer Perceptron model was trained with 3 hidden layers and 1 output layer. The size of the layers were fine tuned to find the highest test accuracy. The final architecture used is described in Table 5.11.

Table 5.12 shows the compilation parameters used for the model. Table 5.13 shows the parameters for training the model.

The results of this model are described in Table 5.14. The confusion matrix is shown in Figure 5.12 and the ROC curve in Figure 5.13.

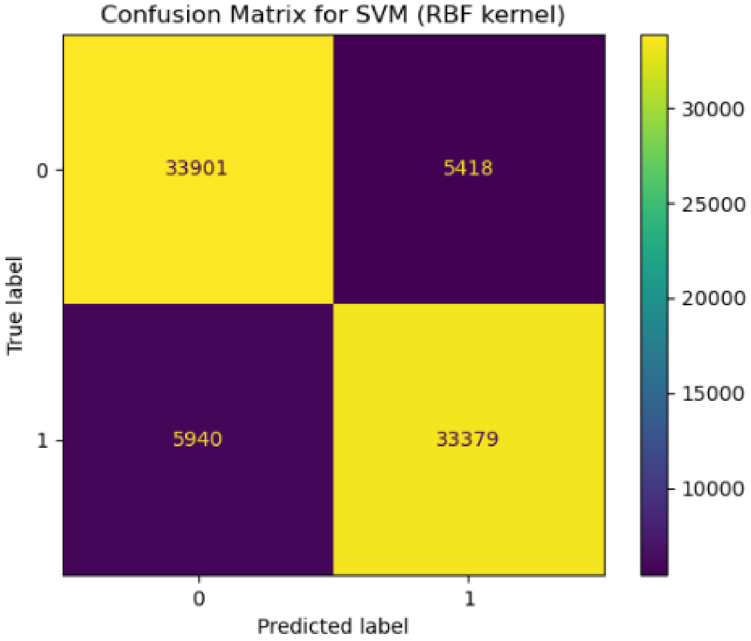


Figure 5.10: Confusion Matrix for SVM

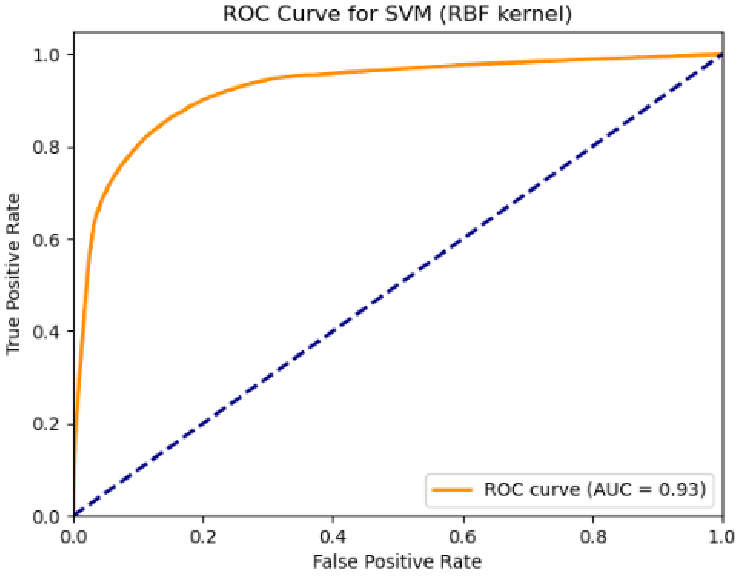


Figure 5.11: ROC Curve for SVM

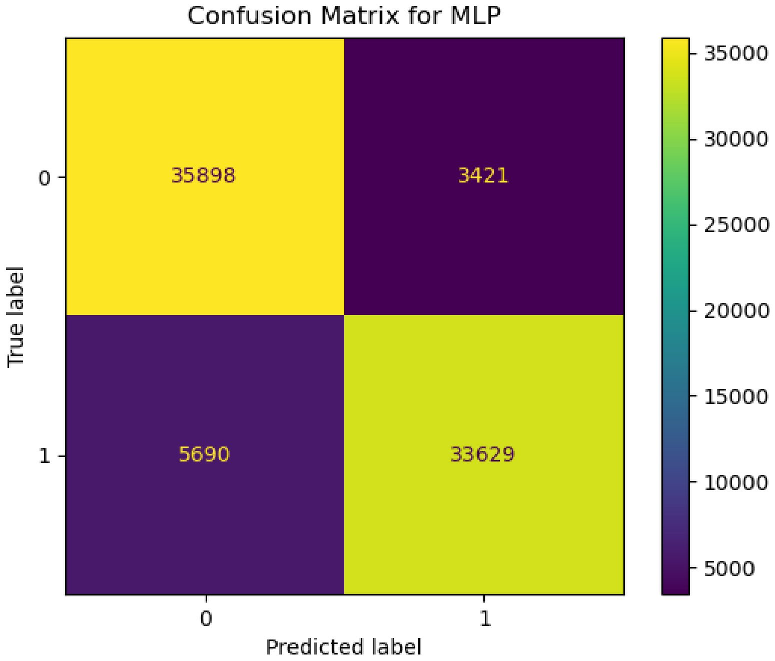


Figure 5.12: Confusion Matrix for MLP

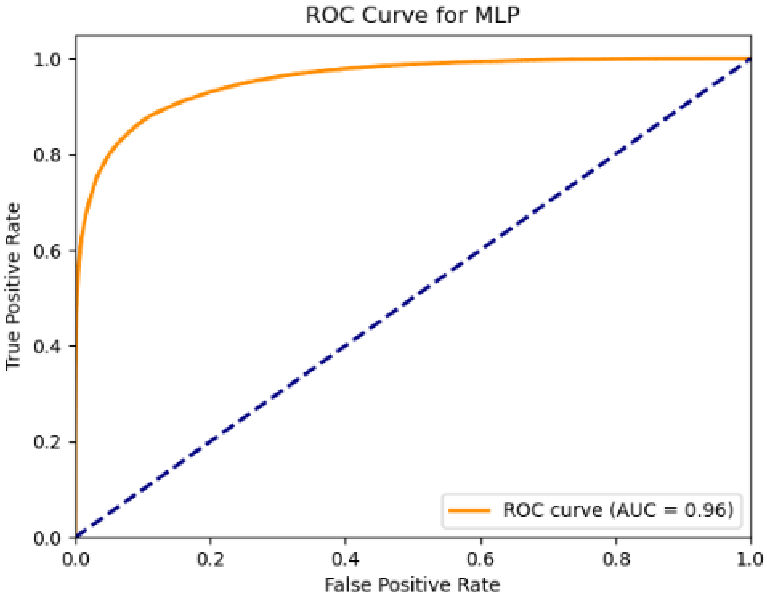


Figure 5.13: ROC Curve for MLP

Dense	512
Activation	ReLU
Dropout	0.3
Dense	256
Activation	ReLU
Dropout	0.3
Dense	128
Activation	ReLU
Dropout	0.4
Dense	1
Activation	Sigmoid

Table 5.11: Architecture for MLP

Optimizer	Adam
Learning Rate	0.001
Loss	Binary Cross Entropy
Metrics	Accuracy

Table 5.12: Compilation Params for MLP

Epochs	50
Batch Size	32
Validation Split	0.2

Table 5.13: Training Params for MLP

Accuracy	.88
Recall	.86
Precision	.91
F1 Score	.88
Specificity	.91

Table 5.14: Results for MLP

5.7 Summary and Comparison

Table 5.15 shows the accuracy of each model and highlights the highest as the **Decision Tree**. Figure 5.14 plots all of the ROC's for each model. When taking in both of these, the optimal classification model for the dataset is a **Post-Pruned Decision Tree with** $alpha = 2.113e - 06$. The KNN model

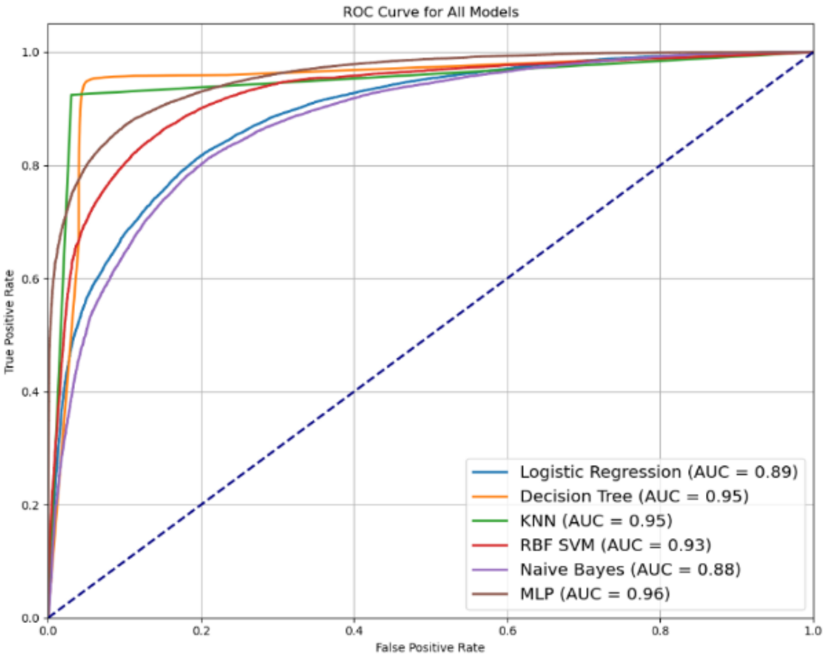


Figure 5.14: ROC Curve for All Models

is a close second, and would most likely perform similarly.

Logistic Regression	.81
Decision Tree	.95
KNN	.95
SVM	.86
Naive Bayes	.80
MLP	.88

Table 5.15: Classification Accuracy Comparison

Chapter 6

Phase IV - Clustering and Association

For Clustering, the K-Mean++ and DBSCAN were applied to find potential clusters in the data. For Association, the Apriori algorithm suggested rules in the dataset.

6.1 Clustering

6.1.1 K-Mean++

For values of $2 \leq K \leq 20$, the Silhouette Score and Within-Cluster Sum of Squares (WCSS) is shown in Figure 6.1 and Figure 6.2, respectively. What this shows is that there are most likely between 6-8 clusters on the dataset. The identified clusters are most likely either sports or countries. There are some countries that have a lot of athletes, such as the USA, Britain, Russia, China. There are also a select amount of Sports that dominate the events, such as Gymnastics, Swimming, Athletics, etc. However, this is not likely the clusters being identified as there would be more than 8 clusters, especially considering the combination between Summer and Winter sports.

6.1.2 DBSCAN

The DBSCAN algorithm was grid searched upon the hyper parameters shown in Table 6.1.

The best parameters found, based on the highest silhouette score, are:

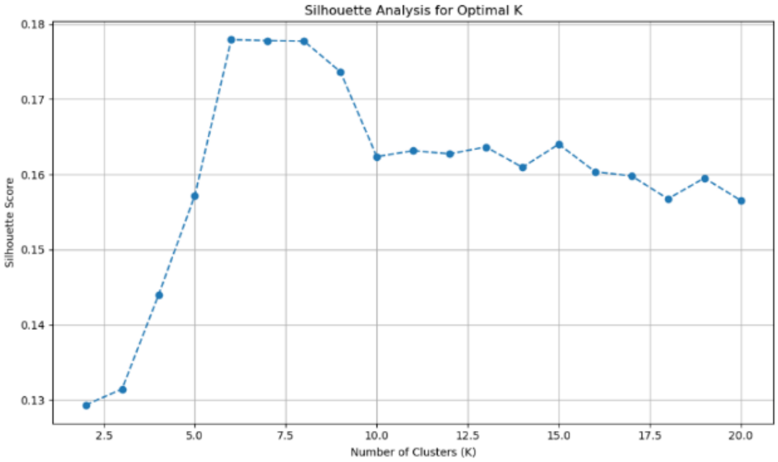


Figure 6.1: Silhouette Scores for Different K Clusters

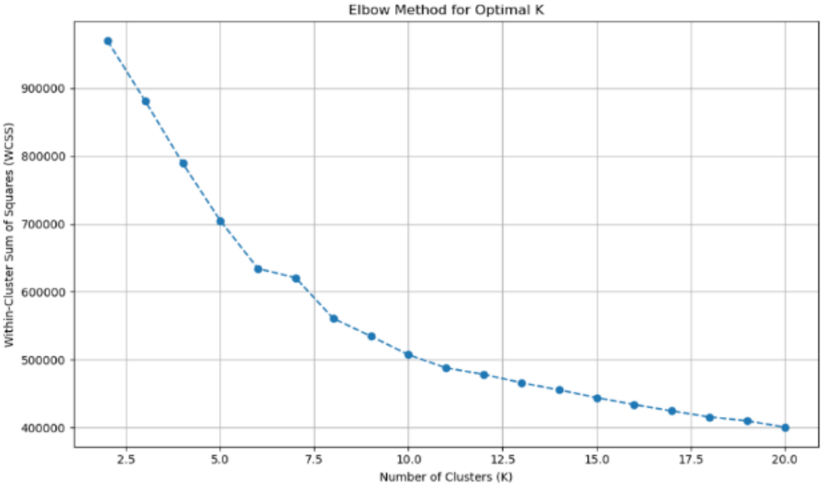


Figure 6.2: WCSS for Different K Clusters

eps	0.3, 0.5, 0.7, 1.0, 1.5
minsamples	3, 5, 10, 15

Table 6.1: Param Grid for DBSCAN

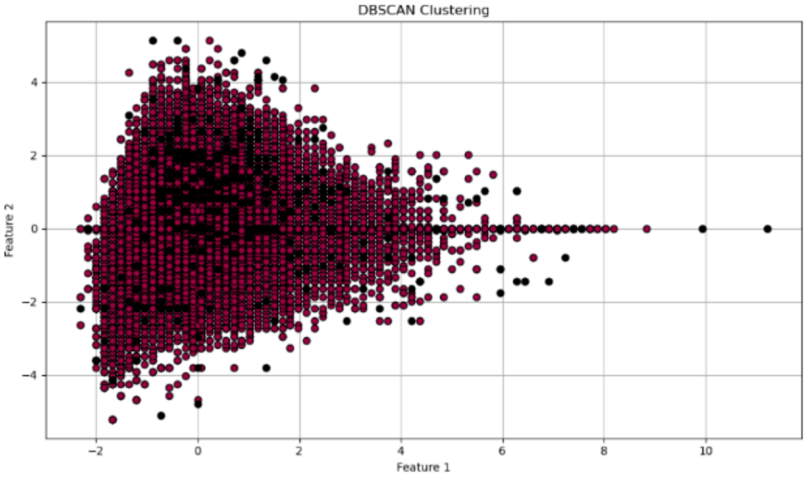


Figure 6.3: DBSCAN Results

- *eps*: `1.5`
- *minsamples*: `10`

with a **silhouette score of 0.39**.

The number of clusters identified was **1**. The results are shown in Figure 6.3.

The 1 cluster identified is most likely all observations that have 'No Medal'. A very large portion of the data has no medal awarded.

6.1.3 Summary

Results from both K-Mean++ and DBSCAN show that there are not strong clusters in the data, and not meaningful interpretations to be taken and extracted. The low silhouette scores reveal that the clusters are not very tight and are overlapping.

6.2 Association

Two separate analyses were performed: team performance and demographic performance. For both, only rules that included **Bronze, Silver, or Gold Medals** and a **lift $\geq$ 1** were considered. The features **Age, Height, Weight, and Year** were binned into categorical features using pandas cut function:

- **Age** = Under 20, 20-30, 30-40, 40+
- **Height** = Very Short, Short, Average, Tall, Very Tall
- **Weight** = Very Light, Light, Medium, Heavy, Very Heavy
- **Year** = Decades, 1890-2010s

6.2.1 Team Performance

Team performance aims to see how well countries have performed in the Olympics, and see any trends across time from countries.

Team performance used the apriori algorithm with the following features:

- Team
- Medal
- Sport
- Event
- Decade

The only rules that were identified with a **lift $\geq$ 1** that are interesting are that the **United States** won **Gold Medals with a lift = 2.81** and **Silver Medals with a lift = 1.75**.

6.2.2 Demographic Performance

Demographic performance seeks to identify winning athletes and their qualities.

Demographic performance used the apriori algorithm with the following features:

- Age Group
- Height Group
- Weight Group
- Sport
- Event
- Medal
- Decade

The following list describes interesting rules found:

- Age 30-40 had a *higher* lift for **Gold Medals** than Age 20-30 (1.041 vs 1.111)
    - Both are still close to 1 and are most likely not significant
- The most likely demographic to win a **Gold Medal** was the **Light, Tall, 20-30 years old** group, with a **lift = 1.529**.
    - In fact, this was the only group that showed significant lifts out of all demographics.
    - Combinations of these groups performed well, e.g. **Tall, 20-30** has a **lift = 1.57**
    - Also the best performing group for **Silver Medals**.

There were no trends among specific events, sports, or decades. Heavy and short groups never had an association with any other features with a lift $\geq$ 1.

Chapter 7

Recommendations

7.1 Summary

7.1.1 EDA

Data cleansing, standardization, and encoding set up the data to be useful in feature selection, regression, classification, and clustering analysis. Different encoding techniques were explored to efficiently encode categorical data while still capturing information.

Principal Component Analysis (PDA), Random Forest feature importance, and backward stepwise regression were all techniques employed to identify relevant features and reduce dimensionality.

7.1.2 Regression

A polynomial regression model was fine tuned to a degree of 5 and trained on the training set to predict the target **weight**, with a Mean Square Error of 29.011 on over 51,000 testing points. Most important features were **height, event, and country**.

7.1.3 Classification

All of the following techniques were trained and fine tuned to classify **Sex**:

- Logistic Regression
- Decision Tree
- KNN

- Support Vector Machine
- Naive Bayes
- Multi Layer Perceptron

The best classifier was the **post-pruned Decision Tree** with a test accuracy of **0.95**. Most relevant features were **weight, sport, and country**.

7.1.4 Clustering

From K-mean++, about 8 clusters were weakly identified, most likely representing countries that dominated athletes at the Olympics.

From DBSCAN, only 1 cluster was weakly identified most likely referring to all the athletes who competed and were awarded **No Medal**.

From the apriori algorithm, the only signification associations found were that the United States have strong associations with winning Gold and Silver medals. In addition, the best performing athletes are light, tall, and between 20-30 years old.

7.2 Lessons Learned

This project allowed me to strengthen my understanding of machine learning techniques through hands-on experience. The following are key lessons gained:

- **Feature Selection** - identifying relevant features and reducing dimensionality of the independent space.
- **Linear Regression** - implementing multiple linear regression to predict a variable and maximizing adjusted R^2.
- **Grid Searching Hyperparameters** - lots of experience with fine tuning models to achieve highest test accuracy.
- **Comparing Classifiers** - one technique will not be king in all domains. It was helpful to expand my toolset and learn the importance of matching the model to the data's characteristics.

- **Introduction to Clustering and Association** - Although the smallest portion of the project, an introduction to these unsupervised learning techniques deepened my understanding of how to approach clustering and association data mining problems.
- **Training Time** - this project emphasized a very real challenge with training some models. Training time can be very expensive, especially on larger datasets, and can factor in to which models are chosen to be trained.

These lessons have enhanced my technical experience and skills, allowing me to better approach and solve machine learning problems moving forward.

7.3 Future Works

To improve classification, perhaps including the name of the athlete might yield better results. It's possible that it could dominate all the features and directly identify males or females, but that was not checked in this project.

In addition, there's a chance that only selecting more recent data could lead to better results. Older Olympic years had much fewer females competing, and was severely unbalanced. Recent years have undone this. Maybe selecting and focusing on the data after a certain year could improve models.

A final approach for improving classification would be employing Random Forest ensemble models. The Decision Tree was the best model, and so it leads to the logical step of attempting RFs, which tend to be an improvement upon Decision Trees.

More research should be put into clustering this dataset to see if any stronger clusters can be identified. In particular, it would be interesting to see if there are clusters of **Gold, Silver, or Bronze Medals** that could identify winning athletes.

Chapter 8

Appendix

Code can be accessed at https://github.com/trippspano/CS5805-Final-Term-Project.

Chapter 9

References

120 Years of Olympic History. https://www.kaggle.com/.

Machine Learning I, Virginia Tech. Prof Reza Jafari