STAT-627 Final Project-NBA

Te-Jou(Carol) Hou, Sean Hsu, Chih-Chen Wang (Gilbert), Fang-Yi Wu(Eva)

2024-11-25

I. Introduction

In this project, we use the NBA player salary data for the 2022-23 season from Kaggle to build models that analyze the relationship between player performance and their salary ranges. We will apply Logistic Regression, LDA and QDA, KNN, Tree-based methods, and SVM to develop predictive models. By utilizing cross-validation, we aim to tune the optimal parameters for each model. Finally, we will compare the error rates of these five models to identify the best-performing model.

We chose to predict salary ranges (e.g., lower, middle, and upper) instead of exact salary amounts for several reasons:

- **Mitigating Extreme Values:** NBA salary data often contains extreme values (e.g., superstar salaries and minimum contracts), which can negatively impact regression models. Dividing salaries into ranges reduces the influence of these extremes, improving model stability.
- **Decision-Making:** Salary ranges help teams and agents assess a player's relative market value, aiding strategic decisions like investments, trades, or contract negotiations. Scouts can use such models to identify promising players for further evaluation.
- **Eliminating Unit Variations:** Salary differences may be influenced by market conditions or timing. Using ranges instead of exact figures makes the model more universal and reduces the impact of these external factors.

II. Data Description

- **Data Source:**

  Kaggle NBA Player Salaries 2022-23 Season

  The dataset contains 467 entries and 31 variables. For the purpose of modeling, we cleaned the data to retain only the relevant columns: Salary, AST (Assists), STL (Steals), BLK (Blocks), TOV (Turnovers), PF (Personal Fouls), and PTS (Points). We also replaced the abbreviated column names with their full names. Additionally, we replaced the original Salary variable with Salary_Group.
- **Salary Grouping:**
  - Budget Tier: Bottom 25%, budget players
  - Mid-Tier: 25%-50%, mid-value players
  - Upper Mid-Tier: 50%-75%, high-value players
  - Premium Tier: Top 25%, elite players

- **Method for Splitting Training and Testing Datasets:**

  To maintain a consistent distribution of salary tiers across both the training and testing datasets, the createDataPartition function was used instead of simple random sampling. This approach guarantees stratified sampling, preserving the proportional representation of each class within the four salary tiers across both datasets.

III. Model Building and Evaluation

1. Logistic Regression

```
 Accuracy
0.4130435
   Class: Budget Tier       Class: Mid-Tier   Class: Premium Tier
            0.6463245            0.5132919            0.7514563
Class: Upper Mid-Tier
            0.5214932
```

We employed a penalized multinomial logistic regression model to predict the four salary tiers using six features (Assists, Steals, Blocks, Turnovers, Personal Fouls, and Points), while applying regularization (decay = 0.1) to prevent overfitting.

The model achieves an overall accuracy of 41.3%. It performs best for the Premium Tier (Balanced Accuracy: 75.15%) and moderately well for the Budget Tier (64.63%). However, it struggles with the Mid-Tier (51.33%) and Upper Mid-Tier (52.15%), indicating room for improvement in distinguishing these classes.

2. LDA and QDA

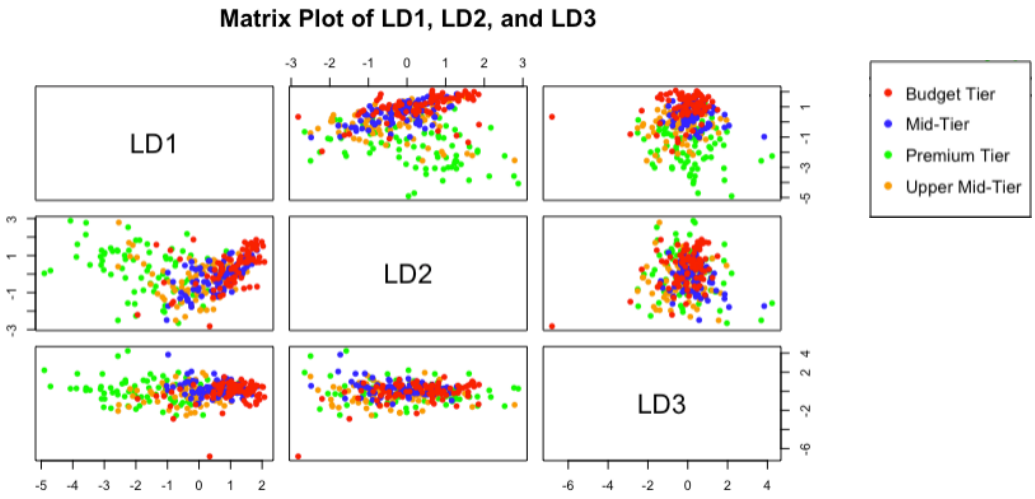
- **LDA**

```
 Accuracy
0.4565217
   Class: Budget Tier       Class: Mid-Tier   Class: Premium Tier
            0.7077670            0.4977376            0.7563107
Class: Upper Mid-Tier
            0.5851244
```

Accuracy: 45.65% – This indicates that the model correctly classified approximately 46% of the test samples. The model performed better for "Budget Tier" and "Premium Tier" classes compared to "Mid-Tier" and "Upper Mid-Tier" classes. This may indicate overlapping features or insufficient separability in the dataset for these middle tiers.

```
   Budget Tier    Mid-Tier Premium Tier Upper Mid-Tier
3  0.112581578 0.056934957    0.4861354    0.344348069
12 0.000114267 0.001235918    0.9919031    0.006746698
13 0.018719907 0.206933175    0.4257466    0.348600329
```

LDA Posterior Probabilities: These probabilities are distributed across all salary groups, indicating that LDA has a smoother, linear approach to predicting the classes. For instance, for observation 3, LDA predicts the “Premium Tier” with a posterior probability of ~0.486, which is higher than other classes but not overwhelmingly confident.



The points for different salary groups overlap significantly, especially for Mid-Tier and Upper Mid-Tier. This indicates that these classes are not linearly separable, which aligns with the lower sensitivity and specificity observed for these tiers.

- **QDA**

```
 Accuracy
0.4710145
   Class: Budget Tier       Class: Mid-Tier   Class: Premium Tier Class:
Upper Mid-Tier
            0.6277393             0.6218891             0.7800277
0.5596719
```

Accuracy: The model correctly classified approximately 47.1% of test samples, which is slightly better than random guessing.

```
    Budget Tier     Mid-Tier Premium Tier Upper Mid-Tier
3  1.507387e-05 9.384871e-17    0.9931651   0.0068198440
12 5.006216e-11 7.756597e-14    0.9995301   0.0004698639
13 1.460671e-08 4.723957e-01    0.5177591   0.0098452379
```

QDA Posterior Probabilities: QDA shows much sharper probabilities, with predictions leaning heavily towards one class in most cases (e.g., observation 3 predicts "Premium Tier" with 99.3% confidence). This behavior reflects QDA's ability to model non-linear decision boundaries, but it may indicate overfitting, especially for smaller datasets.

- **Comparison Between LDA and QDA**

```
LDA CV Accuracy: 0.4924012
QDA CV Accuracy: 0.4741641
```

LDA CV Accuracy: 49.2%. LDA performs slightly better during cross-validation, suggesting it generalizes marginally better. This makes sense, as LDA assumes simpler (linear) boundaries, which reduces overfitting in smaller datasets.

QDA CV Accuracy: 47.4%. QDA performs slightly worse, likely due to its higher flexibility and sensitivity to data variability, leading to overfitting.

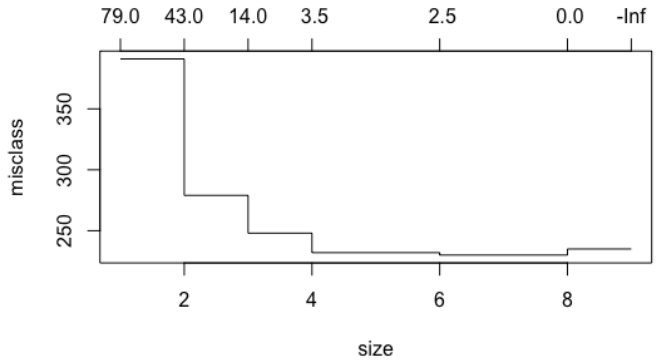
3. KNN

```
[1] 11
 Accuracy
0.4565217
   Class: Budget Tier       Class: Mid-Tier   Class: Premium Tier
            0.6552011            0.5608032            0.7414702
Class: Upper Mid-Tier
            0.5899321
```

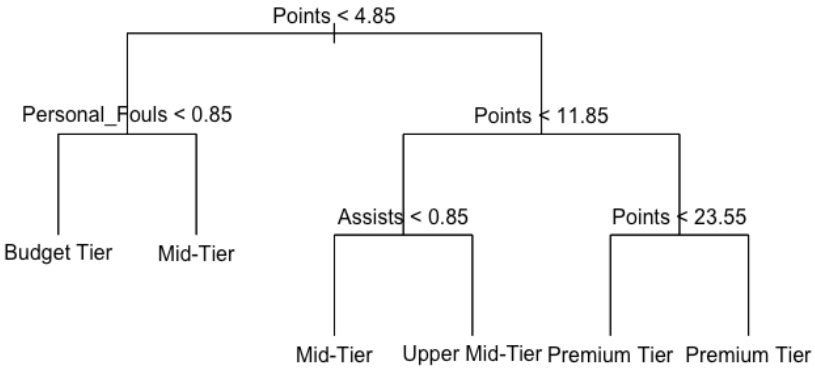
- Accuracy: 45.7%
- The best k value for your KNN model is 11.
- Using the optimal k = 11, the model's accuracy on the test set is 45.7%.

Based on the Balanced Accuracy and overall accuracy (45.65%), the KNN model with the optimal k value of 11 performs best in predicting the Premium Tier (Balanced Accuracy: 74.15%) and moderately well for the Budget Tier (65.52%). However, the model struggles with the Mid-Tier (56.08%) and Upper Mid-Tier (58.99%), indicating that improvements could be made in distinguishing these two classes more effectively. Overall, while the model performs adequately for some classes, there is room for improvement, particularly in handling the Mid-Tier and Upper Mid-Tier categories.

4. Decision Trees



After performing tuning using cross-validation, we found that the misclassification error rate is minimized when the tree size is set to 6. Therefore, we set our tree size to 6. Below is the optimal tree model we obtained:



This decision tree model categorizes NBA players into different salary tiers based on various performance metrics:

- **Points < 4.85**: Players with lower scores are further divided based on *Personal Fouls*:
  ⇒ If personal fouls are **less than 0.85**, these players are classified as "**Budget Tier**".
  ⇒ If personal fouls are **greater than 0.85**, they are classified as "**Mid-Tier**".
- **Points > 4.85**: Players with higher scores are further divided based on *Points and Assists*:
  ⇒ **Points < 11.85**: These players generally belong to the "**Upper Mid-Tier**", but those with assists **below 0.85** are classified as "**Mid-Tier**".
  ⇒ **Points > 11.85**: Most of these players are classified as "**Premium Tier**". Players with even higher scores (**greater than 23.55**) are further confirmed as being in the highest level of the "**Premium Tier**".

```
node), split, n, deviance, yval, (yprob)
      * denotes terminal node

 1) root 467 1295.00 Budget Tier ( 0.252677 0.248394 0.250535 0.248394 )
   2) Points < 4.85 146  283.00 Budget Tier ( 0.554795 0.328767 0.006849 0.109589 )
     4) Personal_Fouls < 0.85 57   72.17 Budget Tier ( 0.789474 0.157895 0.000000 0.052632 ) *
     5) Personal_Fouls > 0.85 89  188.50 Mid-Tier ( 0.404494 0.438202 0.011236 0.146067 ) *
   3) Points > 4.85 321  840.30 Premium Tier ( 0.115265 0.211838 0.361371 0.311526 )
     6) Points < 11.85 205  538.10 Upper Mid-Tier ( 0.146341 0.302439 0.170732 0.380488 )
      12) Assists < 0.85 33   75.21 Mid-Tier ( 0.303030 0.515152 0.090909 0.090909 ) *
      13) Assists > 0.85 172  438.90 Upper Mid-Tier ( 0.116279 0.261628 0.186047 0.436047 ) *
     7) Points > 11.85 116  206.20 Premium Tier ( 0.060345 0.051724 0.698276 0.189655 )
      14) Points < 23.55 89  183.40 Premium Tier ( 0.078652 0.067416 0.606742 0.247191 ) *
      15) Points > 23.55 27    0.00 Premium Tier ( 0.000000 0.000000 1.000000 0.000000 ) *

Classification tree:
snip.tree(tree = TREE, nodes = c(5L, 13L))
Variables actually used in tree construction:
[1] "Points"         "Personal_Fouls" "Assists"
Number of terminal nodes:  6
Residual mean deviance:  2.078 = 958.2 / 461
Misclassification error rate: 0.4497 = 210 / 467
```

In summary, the model reveals that points, personal fouls, and assists are critical factors in determining players' salary tiers, with higher scores typically corresponding to higher salary tiers. The misclassification error rate for this model is 0.4497.

5. SVM

The parameter tuning results indicate that 10-fold cross-validation was used to identify the optimal pa- rameters for the SVM model. The best parameters are **cost = 1** and **gamma = 0.5**, achieving a model performance of 0.4799242. The **cost** parameter in SVM acts as a regularization parameter, balancing the trade-off between achieving a smooth decision boundary and minimizing misclassification. The **gamma** parameter in the radial basis function (RBF) kernel determines the influence range of each data point in the high-

dimensional feature space. The tuning process, conducted using the **tune()** function, evaluated multiple combinations of **cost** and **gamma** values to find the combination that yielded the best performance through cross-validation. The results are stored in a data frame, which highlights the optimal parameter values (**cost = 1, gamma = 0.5**).

```
[1] "Accuracy: 0.46"
```

The SVM model with a radial kernel achieved an accuracy of 46% in predicting NBA player salary groups, indicating limited predictive performance. The confusion matrix reveals frequent misclassifications, particularly between adjacent salary tiers, which may result from class overlap or imbalance.

```
Call:
best.tune(METHOD = svm, train.x = Salary_Group ~ ., data = train_data,
    ranges = list(cost = c(0.1, 1, 10, 100), gamma = c(0.01,
        0.1, 1)), kernel = kernel_type)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1

Number of Support Vectors:  299

[1] "Best Accuracy: 0.5067"
```

The SVM model, using a radial kernel with a cost parameter of 1, achieved its best performance during 10-fold cross-validation with an error rate of 49.33%. This translates to an accuracy of approximately 50.67%, indicating that the model's predictions are only slightly better than random guessing for four salary groups. The model used 299 support vectors, which highlights that many data points are close to the decision boundaries or are misclassified. Although the radial kernel is suitable for non-linear relationships, the relatively low accuracy suggests potential issues such as overlapping salary groups, insufficient feature discrimination, or class imbalance.

IV. Conclusion

The project aimed to predict NBA player salary groups (Budget, Mid-Tier, Upper Mid-Tier, and Premium) based on their performance metrics. We implemented five modeling approaches—Logistic Regression, LDA and QDA, KNN, Decision Trees, and SVM—and evaluated their predictive performance using accuracy and cross-validation metrics. The analysis considers each method’s strengths and limitations.

1. Cross-Model Comparison and Summary

Model	Accuracy	Summary
Logistic Regression	41.3%	• Assists negatively correlated with higher salary groups. • Points and blocks are key predictors of premium-tier salaries.
LDA	45.65%	• LDA performs better for extreme salary groups (Budget and Premium). • Poor performance for mid-range groups due to overlapping features.
QDA	47.1%	• QDA provides sharper class probabilities but risks overfitting. • QDA's flexibility caused sensitivity to dataset size and variability. • Poor performance for mid-range groups due to overlapping features.
KNN (optimal k=11)	45.7%	• Captures non-linear relationships better than logistic regression. • High misclassification for Mid-Tier and Upper Mid-Tier due to class overlap.
Decision Trees (tree size = 6)	55.03%	• Points, personal fouls, and assists were key decision splits. • Effective at segmenting distinct salary groups like Budget and Premium. • Limited accuracy for mid-range salary groups. • Risk of overfitting without careful pruning.
SVM	50.67%	• Poor sensitivity for Mid-Tier and Upper Mid-Tier. • Radial kernel, Cost = 1, gamma = 0.5 provided optimal performance.

2. Best Model: Decision Trees

- With an accuracy of 55.03%, Decision Trees outperformed other models
- Its interpretability provides actionable insights for NBA teams regarding salary classification.

3. Challenges in Mid-Tier Classification

Significant overlap between Mid-Tier and Upper Mid-Tier reduced model performance across all methods.

4. Future Improvements

- Data Augmentation: Increase the dataset size to enhance model generalization.
- Advanced Models: Consider ensemble methods (e.g., Random Forest) for improved classification.

This study highlights the importance of selecting appropriate modeling techniques and tuning parameters to address data challenges such as class imbalance and feature overlap. With additional data and refined features, predictive performance can further improve, providing valuable insights for NBA salary analysis.

V. Team Contribution

Model	Contributor
Logistic Regression	Fang-Yi Wu(Eva)
LDA and QDA	Chih-Chen Wang (Gilbert)
KNN	Fang-Yi Wu(Eva)
Trees	Te-Jou(Carol) Hou
SVM	Sean Hsu