IS 733 Homework 2

Q1. From the feedback you received, what are the takeaways/lessons learned you could apply to future analysis?

From my Youth Player Development project, I discovered some key observations and lessons that will be beneficial for future analysis:

1. Understanding of the Dataset– Working with FIFA player statistics, I had a dataset of 18,207 players with 57 features. This enabled me to distinguish between numerical and categorical variables, identify distributions like overall rating and potential, and set up key factors in player development. Finding trends in the dataset helped me realize how different features impact young player performance.

2. Unraveling Performance Trends Using Temporal Plots – Temporal analysis allowed me to observe trends in player rating by age groups and years joined. I was able to observe player rating developments over time, recognize best age development, and investigate trends for potential vs. total rating amongst youth players. This helped discover when a player's development becomes most influential and where scouting resources need to be deployed.

3. Applying Data Mining Techniques to Classify Players– Perhaps the greatest player development challenge is identifying top talent. By analyzing player potential and total rating data, I was able to classify and sort information to assist in the detection of talent. Understanding how to segment players by position, nationality, and performance trends provided a clearer view of emerging stars in youth football.

4. Applying Distribution Plots for Greater Insight– Histograms and bar charts were useful in visualizing the distribution of player attributes such as rating and potential. Scatter plots and heatmaps were also useful in pointing out correlations between various metrics so that there could be a more data-driven way of analyzing player strengths and weaknesses. These plots were vital in determining how various attributes lead to the success of a player.

5. Building an Interactive Decision-Making Dashboard – Having an interactive dashboard with position, nationality, and rating range filters enhanced dynamic data exploration. Positional filtering and rating slider controls helped to enable scouts and analysts to target talent more accurately. This emphasized the importance of clean, interactive graphics in understanding data.

6. Maximizing Allocation of Resources – By reviewing player value and potential, we were instructed on how to use resources wisely to allocate between position and age groups. This placed even greater significance on the intelligent planning of expenditures in scouting and training programs for the purpose of investing in higher-potential players.

7. Model Evaluation and Selection Role – In talent identification, I exercised different techniques for choosing best performers and used data-driven methods for maximizing the selection process.

Exercising how to evaluate models and validate findings based on measures such as accuracy and trend analysis guaranteed reliability in decision-making.

Key Takeaways:

- Profiling data is crucial in understanding features and structuring analysis.

- Temporal and distribution plots provide critical performance trend insights.

- Data mining approaches improve model classification and talent discovery.

- Interactive dashboards facilitate improved data exploration and decision-making.

- Proper evaluation of trends and model selection provides accurate and reliable insights.

This project enhanced my proficiency in data analysis, interpretation, and presentation in the best possible manner. In the future, I can apply these techniques to other problems involving data, upgrading my analytical process in subsequent studies and practical applications.

Q2. Create a model card

Property	**Decision Tree**	**Naïve Bayes**	**K-Nearest Neighbor**	**Logistic Regression**	**Support Vector Machine (SVM)**
Parametric/Non-parametric	Non-parametric	Parametric	Non-parametric	Parametric	Non-parametric
Input Type	Both (continuous & categorical)	Both	Both	Continuous	Continuous
Output Type	Discrete or Continuous	Discrete	Discrete	Discrete	Discrete
Handles Missing Values?	No (requires imputation)	No (some implementations support handling missing values)	No (requires imputation)	No (requires imputation)	No (requires imputation)
Model Representation	Tree structure	Probabilistic (Bayes' theorem)	Instance-based (stores training examples)	Linear model	Hyperplane in high-dimensional space
Model Parameters	Max depth, min samples split, etc.	Prior probabilities, likelihoods	Number of neighbors (k), distance metric	Coefficients, intercept	Kernel type, regularization parameter
How to Make More Complex?	Increase depth, reduce pruning	Use kernel density estimation	Increase k, change	Add polynomial features,	Use nonlinear kernels

			distance metric	decrease regularization	(RBF, polynomial)
How to Make Less Complex?	Prune tree, limit depth	Use fewer features, apply smoothing	Decrease k	Increase regularization	Use linear kernel, decrease C
Interpretability/Transparency	High (easy to visualize)	Medium (depends on conditional probabilities)	Low (not easily interpretable)	High (coefficients indicate feature importance)	Low (difficult to interpret in high dimensions)

Q3. Wine-Tasting Machine

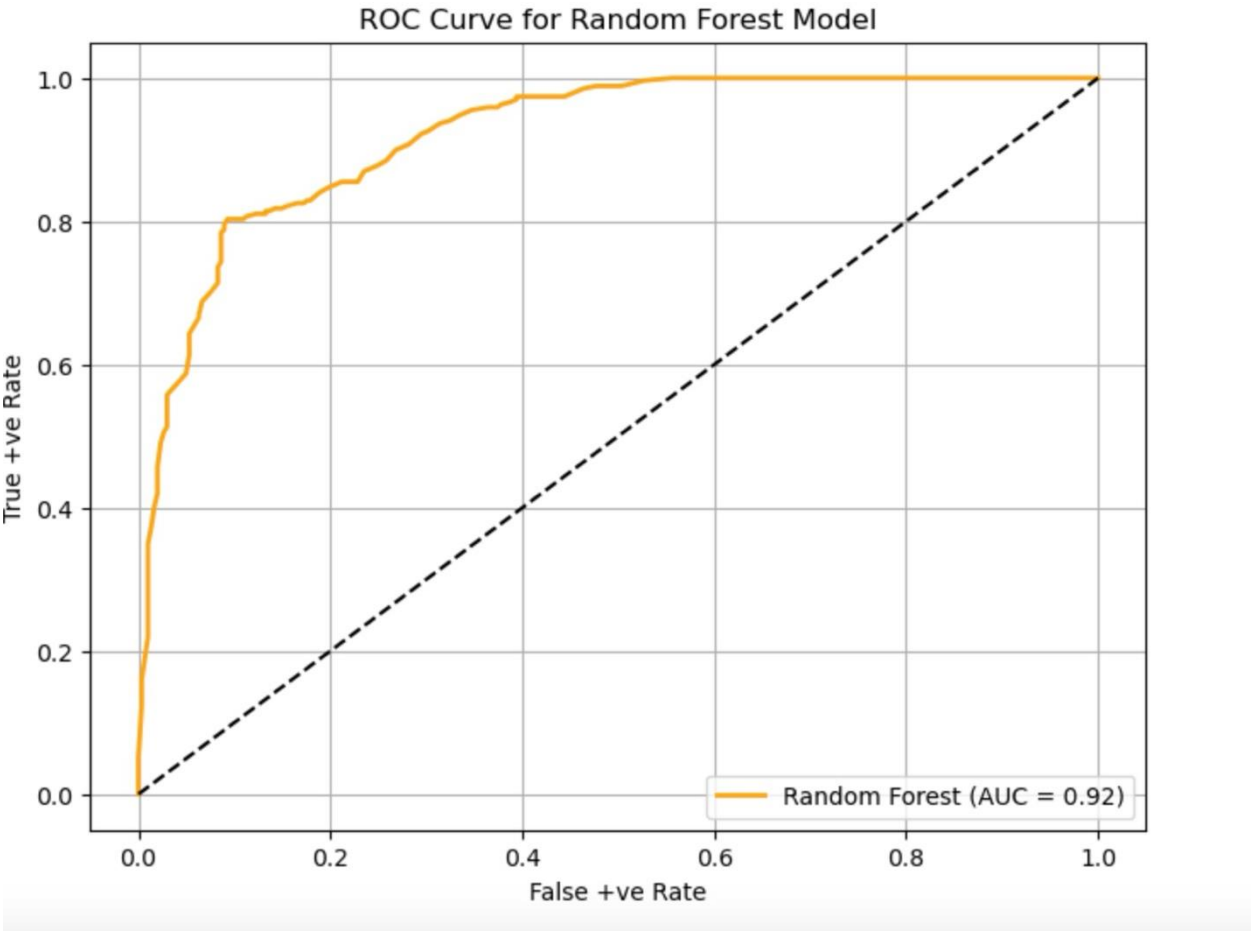
3.1 The Github link below has the python code for Data profiling and HTML file for the red-wine.csv file
https://github.com/Omkark2323/is7332025/blob/a77ea866ac1e9a393f5e07474ee873fa7aa7ce11/data-mining-project-repo/HW2/AF85213_HW2%20(1).ipynb

3.2

Model	Baseline	Logistic Regression	Naive Bayes	Decision Tree	SVM - Linear	SVM - RBF	Random Forest
AUC	0.5	0.87094	0.88248	0.76021	0.87130	0.91059	0.86945
Accuracy	52.89 %	79.51 %	82.14%	75.83 %	78.98 %	82.31 %	80.04 %

3.3 The Github link below has the python code for generating the ROC curve for the Random Forest classifier.

https://github.com/Omkark2323/is7332025/blob/a77ea866ac1e9a393f5e07474ee873fa7aa7ce11/data-mining-project-repo/HW2/AF85213_HW2%20(1).ipynb



3.4 The highest performing model in Q2, based on the AUC score, is the SVM-RBF model which gave an AUC score of 0.9105. When applied on the white-wine.csv dataset, the model yielded the following values:

- Accuracy: 81.25%
- AUC Score: 0.9455

81.25% accuracy indicates that the model is precise in predicting 81 out of 100 cases, indicating high performance. Additionally, the extremely high AUC score of 0.9455 indicates that the model does an excellent job in distinguishing different classes of the type of attribute.

The high AUC value also indicates that the model can make reliable predictions even while dealing with some extent of class imbalance.

For the entire Python code utilized to calculate the AUC score and accuracy for the white-wine dataset based on the SVM-RBF model, see the GitHub link below.

https://github.com/Omkark2323/is7332025/blob/a77ea866ac1e9a393f5e07474ee873fa7aa7ce11/data-mining-project-repo/HW2/AF85213_HW2%20(1).ipynb

3.5 If all models are similar in performance, Decision Tree and Logistic Regression would be excellent choices to produce insights for wine-tasting experts. They are selected since they are easy to explain and present to non-technical individuals.

Decision Tree offers an easy, graphical display of the data in a tree structure, which is easy to understand and intuitive.

Logistic Regression provides feature importance in the form of coefficients; thus, experts can see how each factor impacts the wine quality rating as a "High" or "Low."

While SVM and Random Forest models are equally usable, they are more complex and less interpretable. Random Forest, e.g., is an ensemble of numerous decision trees, thus it is difficult to explain one prediction, especially for those who are not machine learning experts.

Hence, for wine-tasting professionals who require explainable and transparent outcomes, Decision Tree and Logistic Regression are more suitable.