Predicting Starcraft Player Rank from Performance in Ranked Games

Summer Long's Evil Geniuses Data Science Assessment

Overall Best Model: Random Forest with Manually Selected Features

Exploratory Data Analysis

- The data was examined for missing values
  - There were 57 rows with missing values. 55 of these rows were missing Age, Total Hours, and Hours per Week. Of the 2 remaining, 1 was missing both Total Hours and Hours Per Week and 1 was only missing Total Hours.
  - These values were a string, ‘?’, and replaced with NA.

- Each variable was then averaged and graphed by rank, to examine if features were correlated with a particular rank.

Model Selection Metrics

- I chose to use a combination of a *visual evaluation of classification, AUC-ROC score, and accuracy*

- Accuracy alone is insufficient as the data is heavily imbalanced
    - If the model predicted a player to be Platinum every time, it would achieve an accuracy of ~24.0%. This would appear better than a random guess in theory (as a random guess would be 100%/8 = 12.5%), but in practice would be unreliable
    - AUC-ROC score and visual evaluation of the classification mitigates this issue

Features Chosen in Initial Model

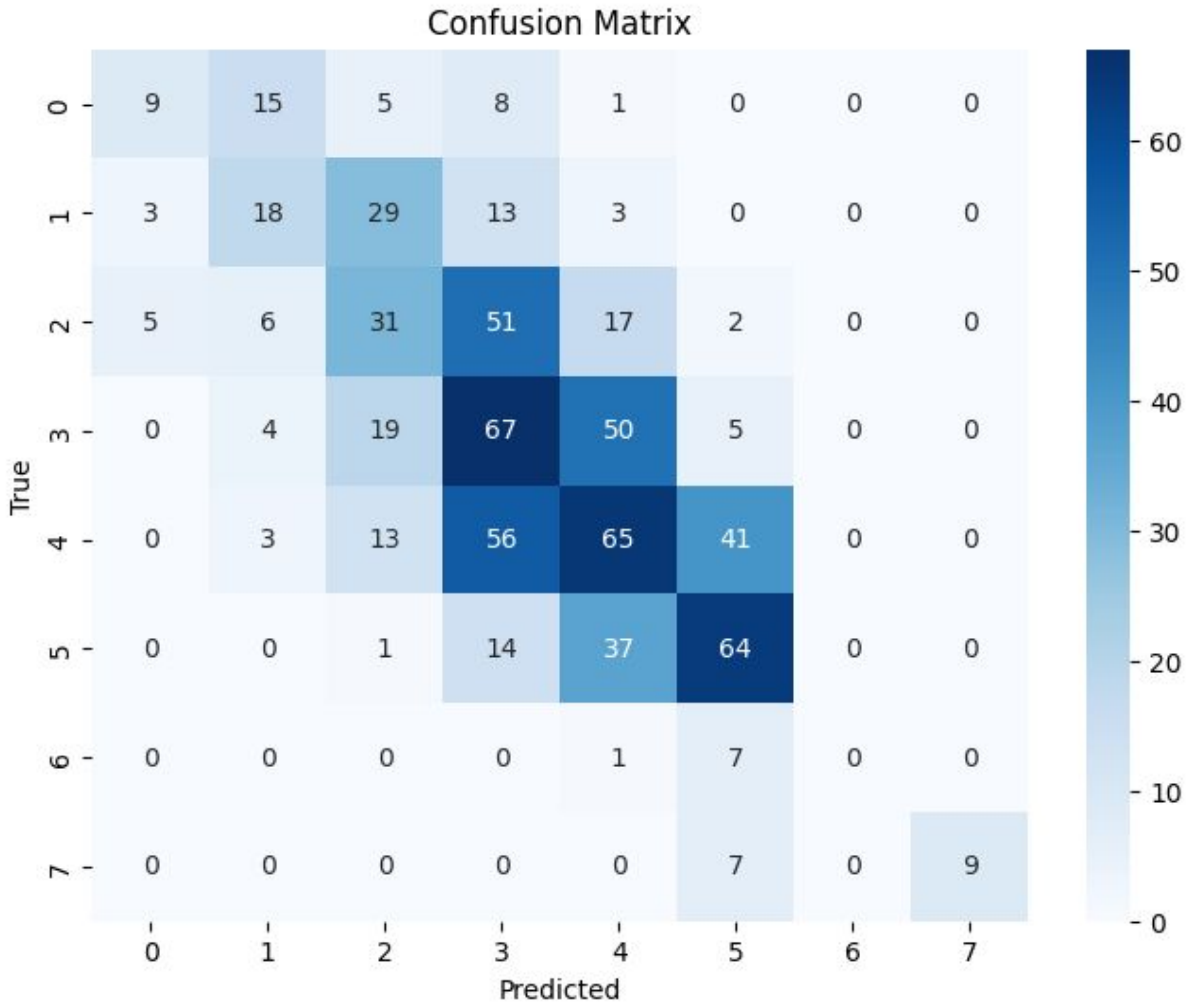
- Age, Total Hours, and Total Hours per Week, and GameID were excluded from the model due to lack of generalizability
  - Since all entries missing Age are Professional Leagues, and the remaining two rows are Diamond, a model may decide to classify based on missing data which would not be generalizable
  - Furthermore, these variables did not appear to be correlated strongly with rank

- Unique Units Made and Actions in PAC did not appear to have strong correlations with the target variable and were also excluded from the initial model

Initial Models Built & Evaluation Metrics

Model	**AUC-ROC**	**Accuracy**
Logistic Regression	.8183	35.35%
Random Forest	*.8259*	*38.73%*
XGBoost	.8072	37.85%

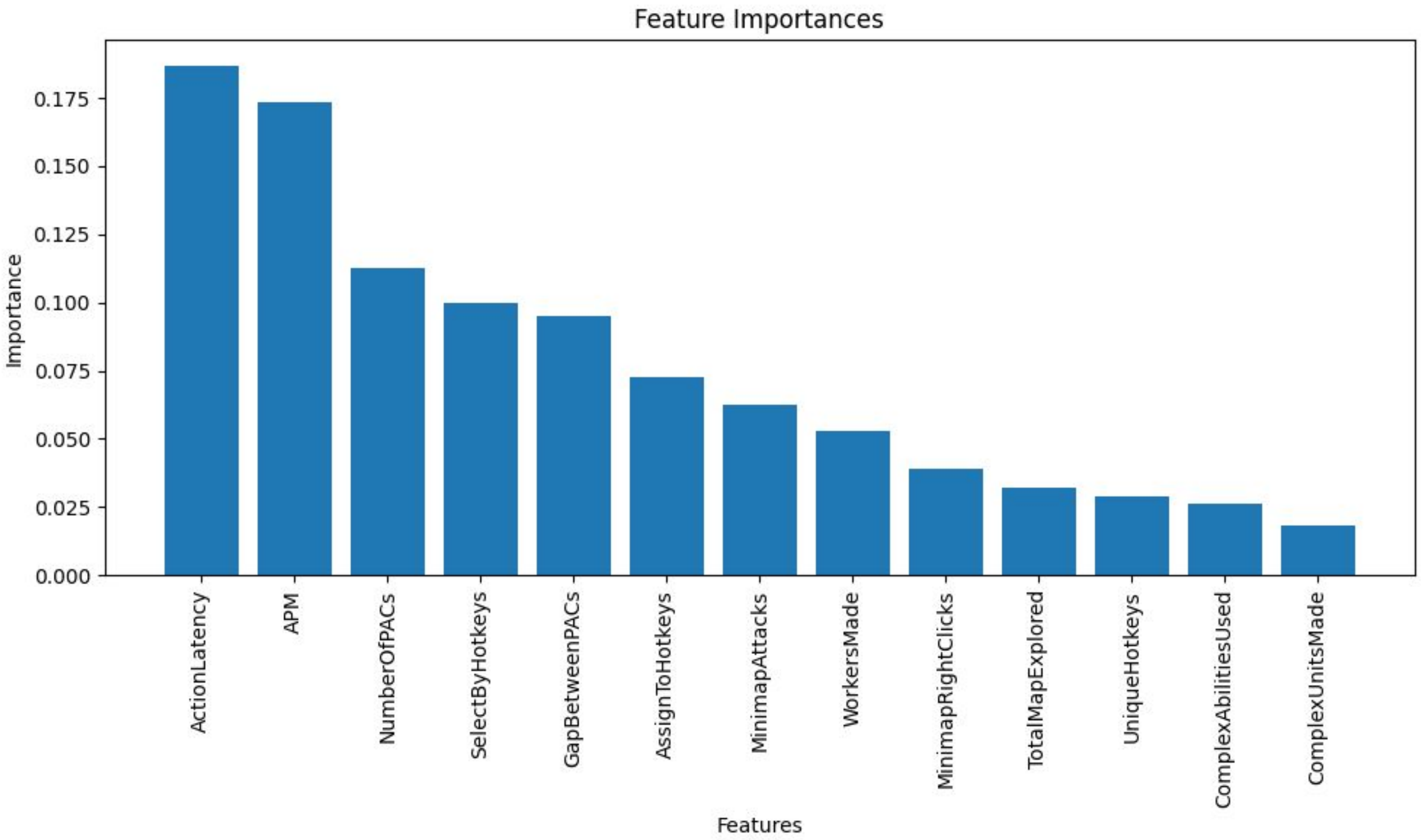
Random Forest is selected as the best initial model as it has the best AUC-ROC and Accuracy.



The Random Forest model performs well at classifying a model within +/- one rank, but not the exact rank (accuracy ~*38.73%*).

For example, the model classifies Platinum (‘4’) as either Gold, Platinum, or Diamond consistently.

An ensemble method was attempted to classify within a category (Platinum, Diamond, or Master, for example) and then a specific class, but this model did not perform as well as the one-step model.



Some features are not particularly importance and could harm the performance of the model, so a second random forest model is trained (deemed the ‘Second-pass with pruned features’) and evaluated compared to the original model

Features < .030 importance are excluded.

Comparison of Metrics between Models

Model	AUC-ROC	Accuracy
Random Forest (Manually Selected Features)	*.8259*	*38.73%*
Random Forest (Second-pass Pruned Features)	.8228	37.11%

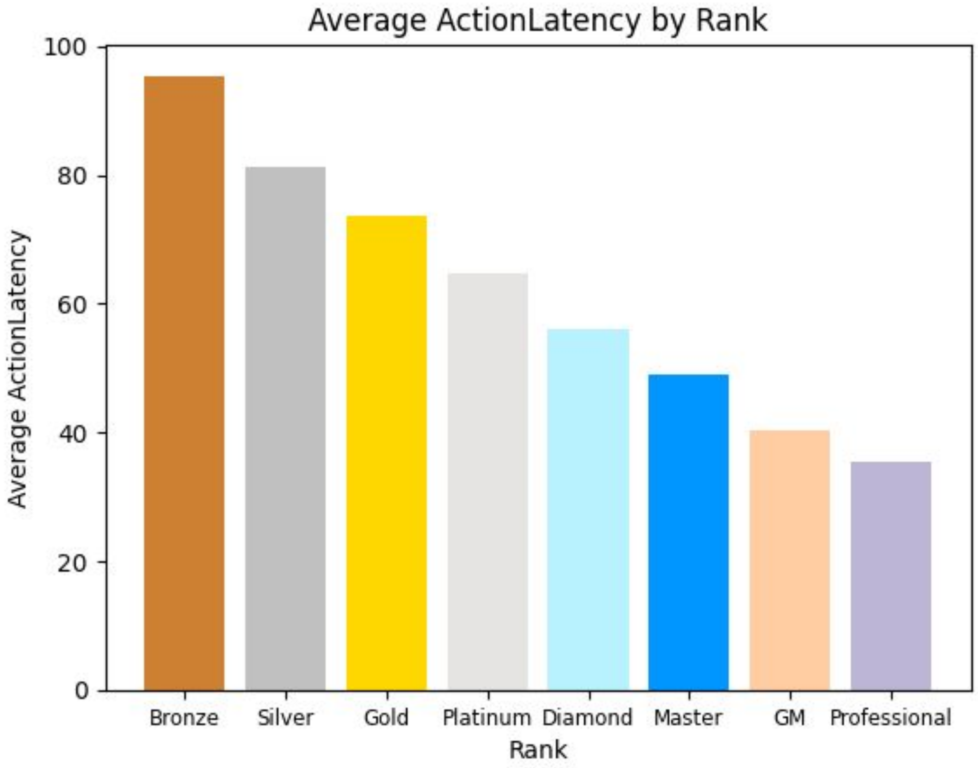
The initial model outperforms the Second-pass Pruned Features model.

Takeaways

Model Performance Interpretation

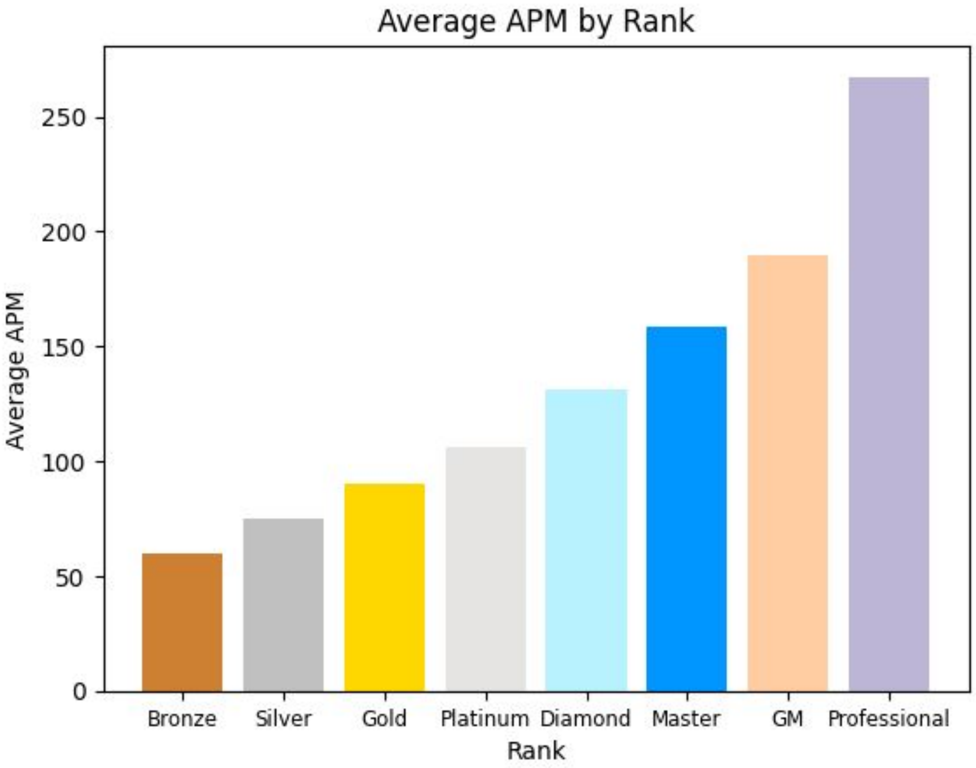
- Although the models did not perform particularly well at predicting a specific rank, it performed relatively well at predicting a player within +/- 1 rank. The models did not, for example, predict a Bronze player as a Grandmaster.

Feature Interpretation - ActionLatency



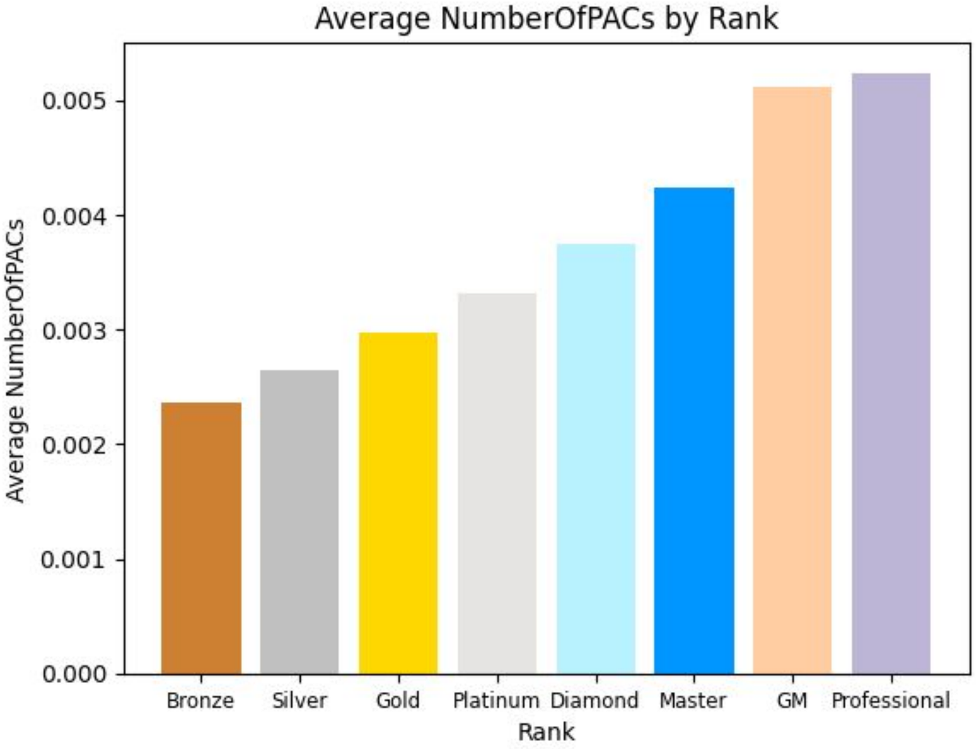
Mean latency from the onset of a PACs to a player's first action in milliseconds appeared to be the most important predictor in the model. It appears that a player with a lower mean latency is ranked higher.

Feature Interpretation - APM



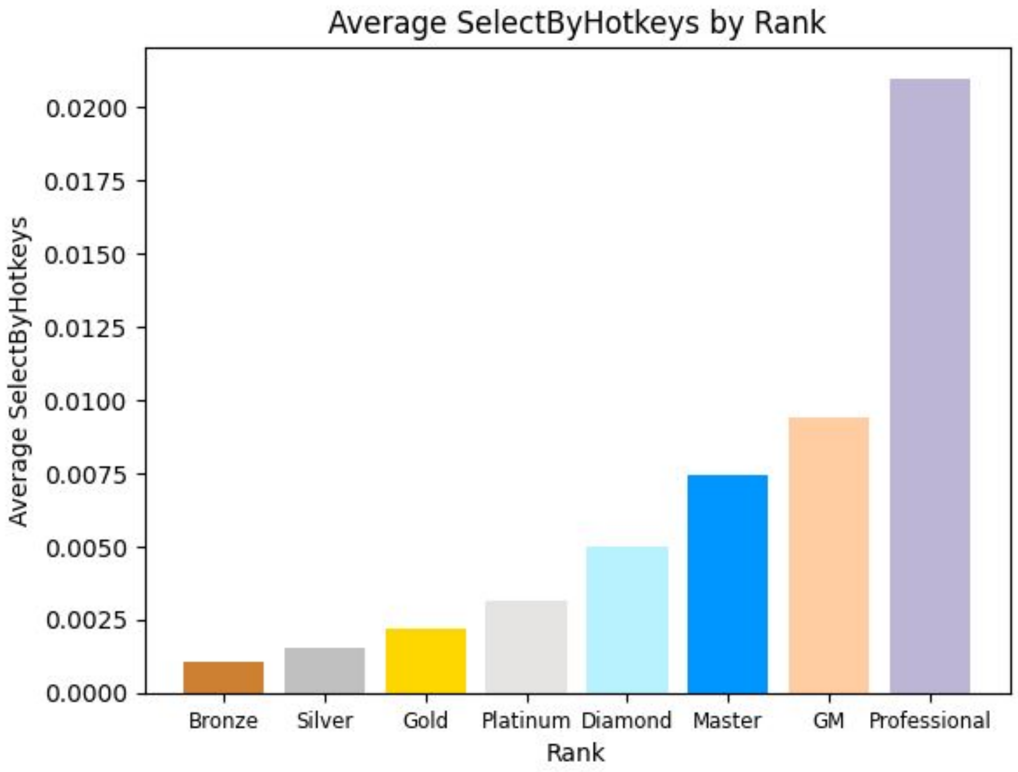
Average action per minute was the second most important predictor in the model. It appears that a player with a higher action per minute is ranked higher.

Feature Interpretation - NumberOfPACs



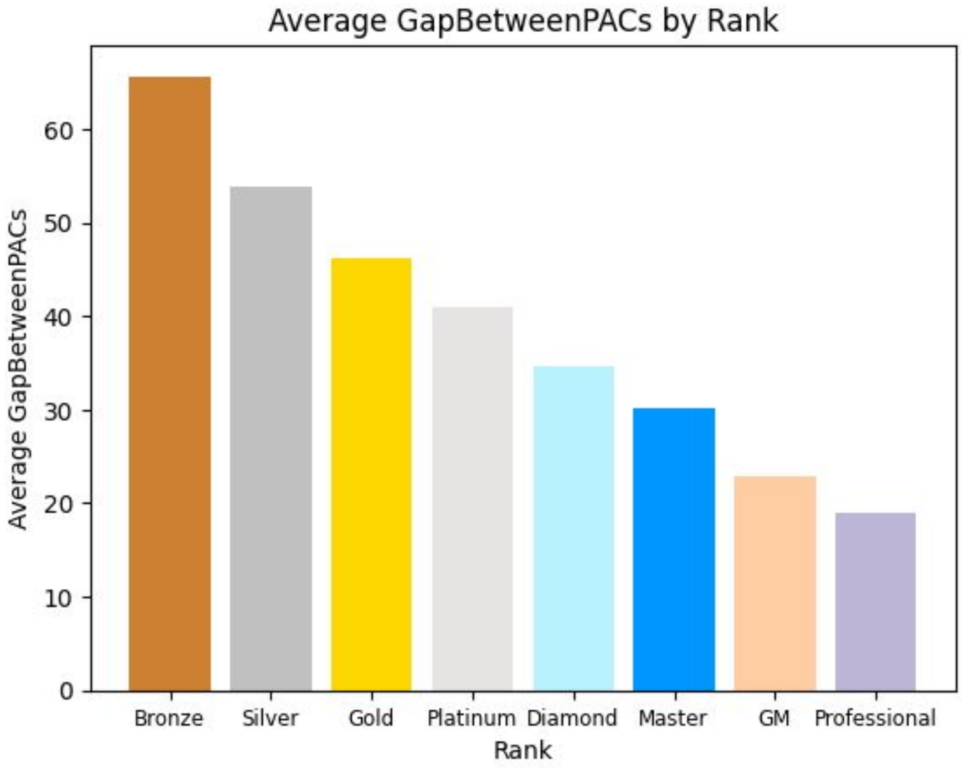
Number of PACs per timestamp was the third most important predictor in the model. It appears that a player with a higher number of PACs per timestamp is ranked higher.

Feature Interpretation - SelectByHotkeys



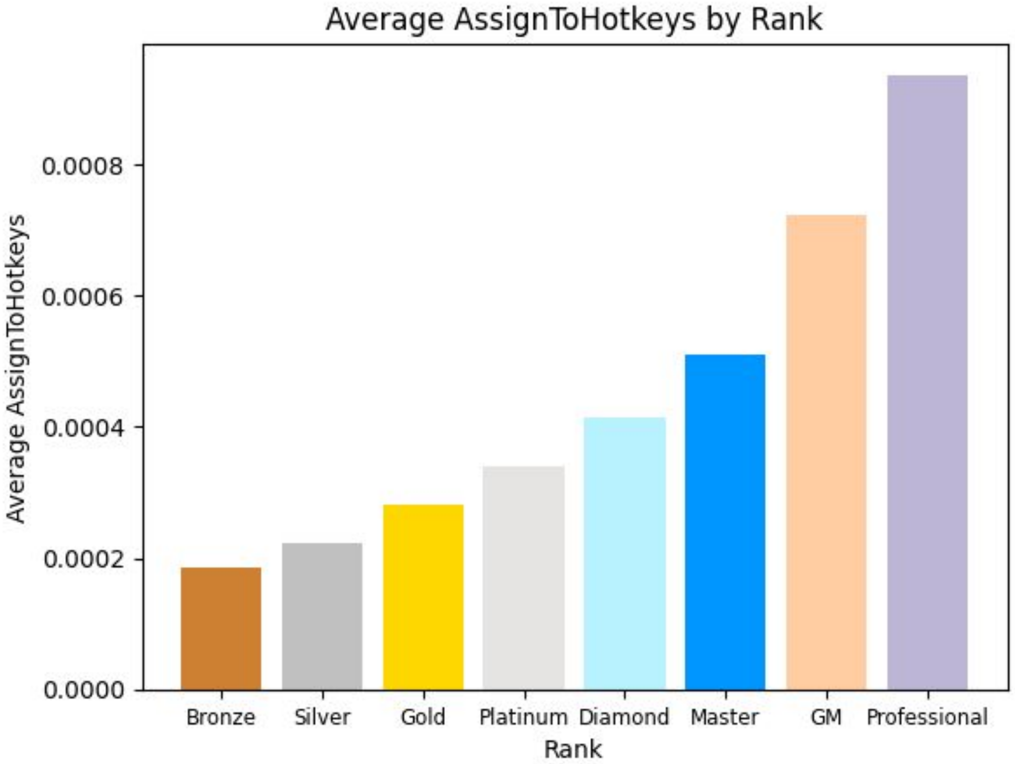
Number of unit or building selections made using hotkeys per timestamp was the fourth most important predictor in the model. It appears that a player with a higher number of unit or building selections made using hotkeys per timestamp is ranked higher.

Feature Interpretation - GapBetweenPACs



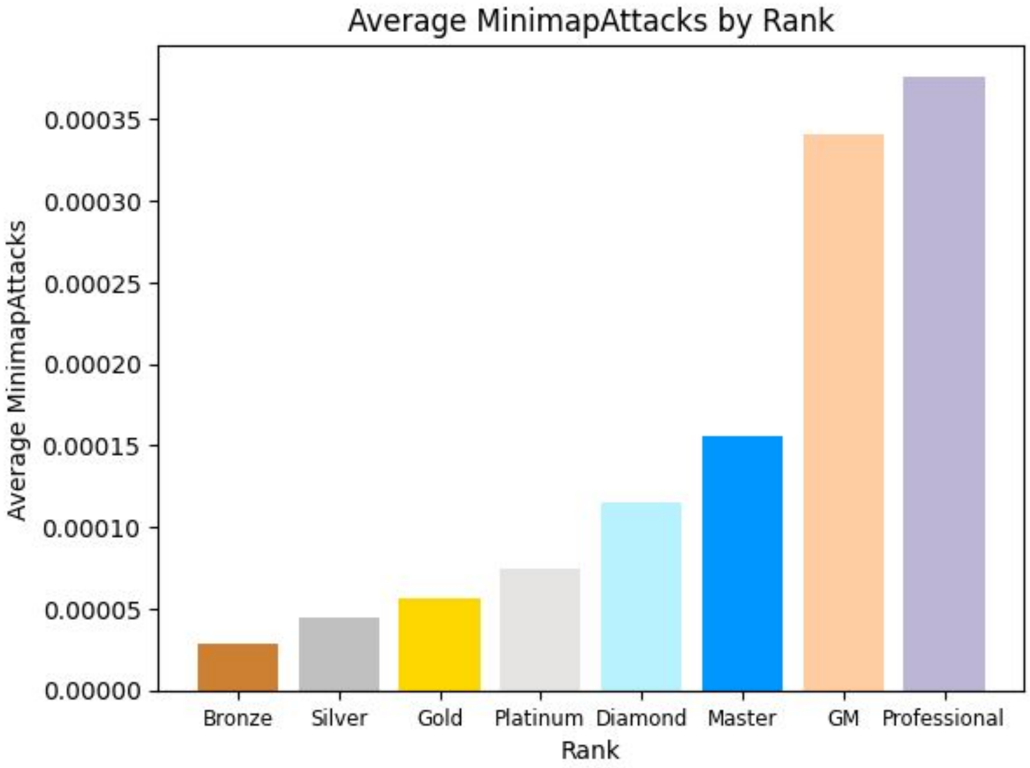
Mean duration in milliseconds between PACs was the fifth most important predictor in the model. It appears that a player with a higher mean duration in milliseconds between PACs is ranked lower.

Feature Interpretation - AssignToHotkeys



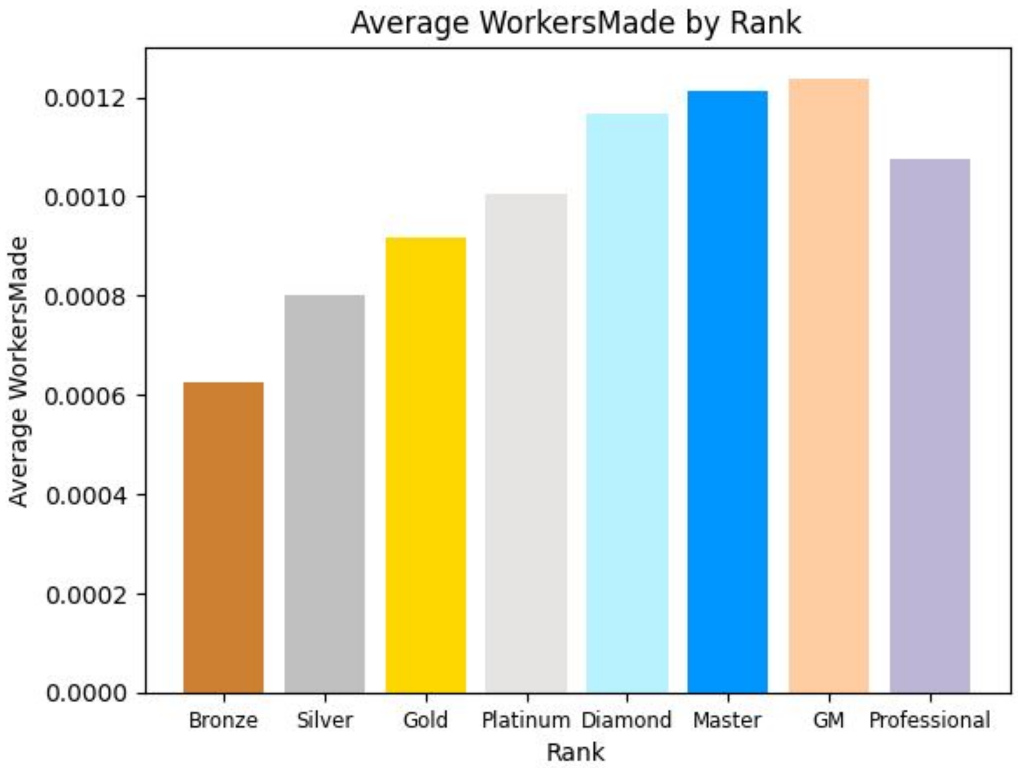
Number of units or buildings assigned to hotkeys per timestamps was the sixth most important predictor in the model. It appears that a player with a higher number of units or buildings assigned to hotkeys per timestamps is ranked higher.

Feature Interpretation - MinimapAttacks



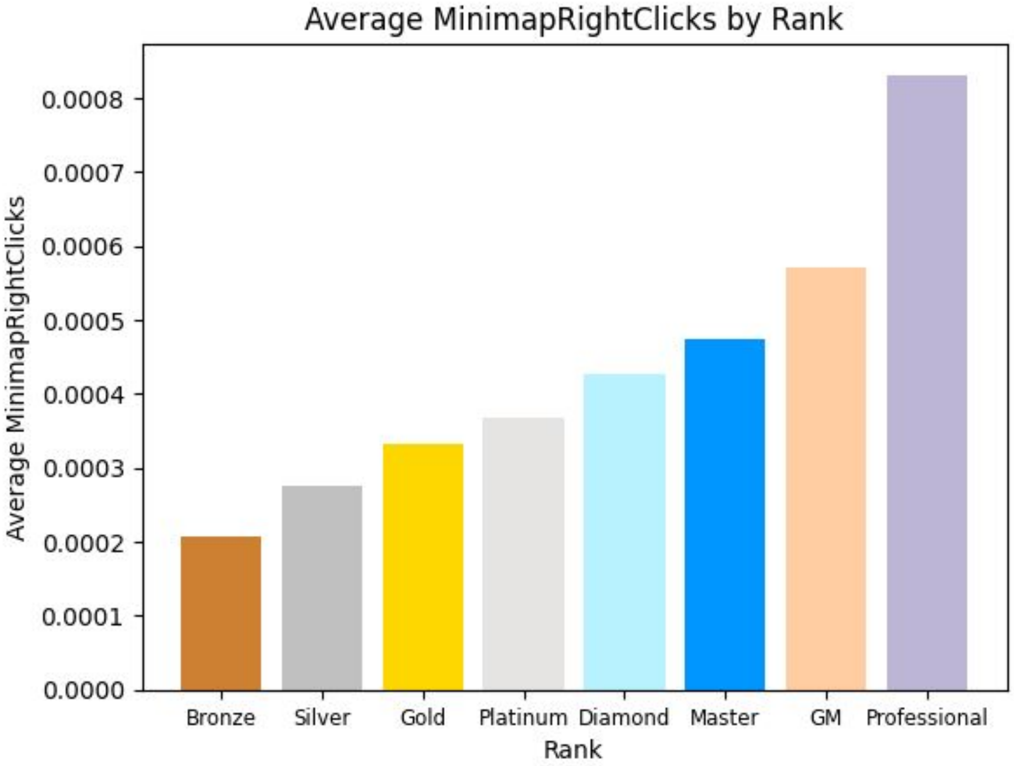
Number of attack actions on minimap per timestamp was the seventh most important predictor in the model. It appears that a player with a higher number of attack actions on minimap per timestamp ranked higher.

Feature Interpretation - WorkersMade



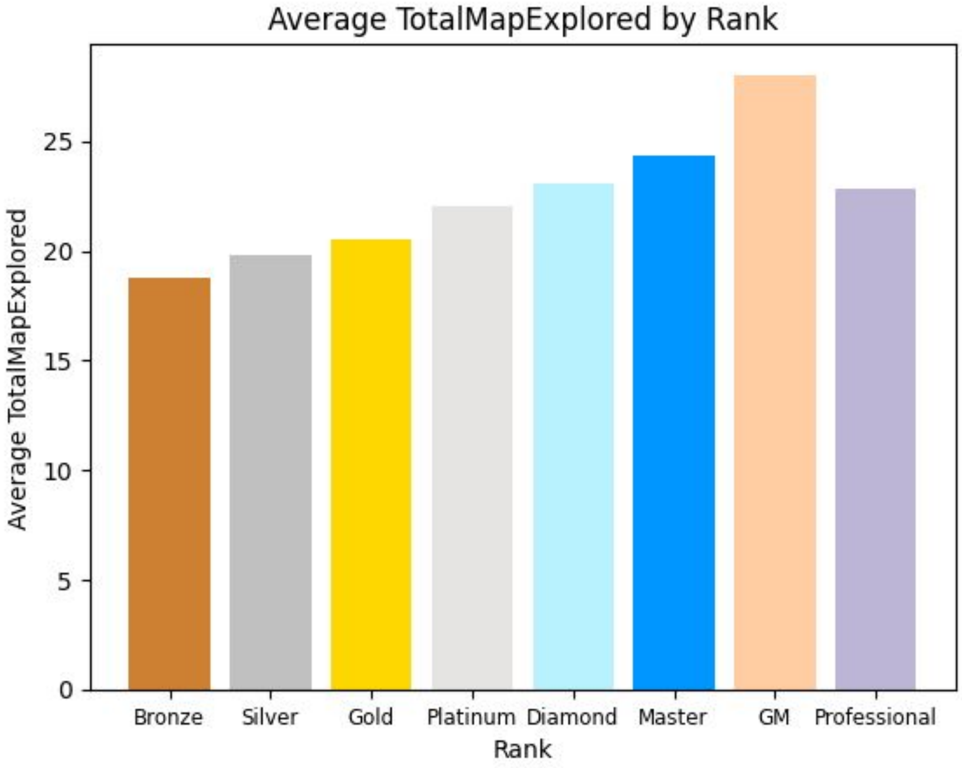
Number of SCVs, drones, and probes trained per timestamp was the eighth most important predictor in the model. It appears that a player with a higher number of SCVs, drones, and probes trained per timestamp was ranked higher, except for a Professional league player.

Feature Interpretation - MinimapRightClicks



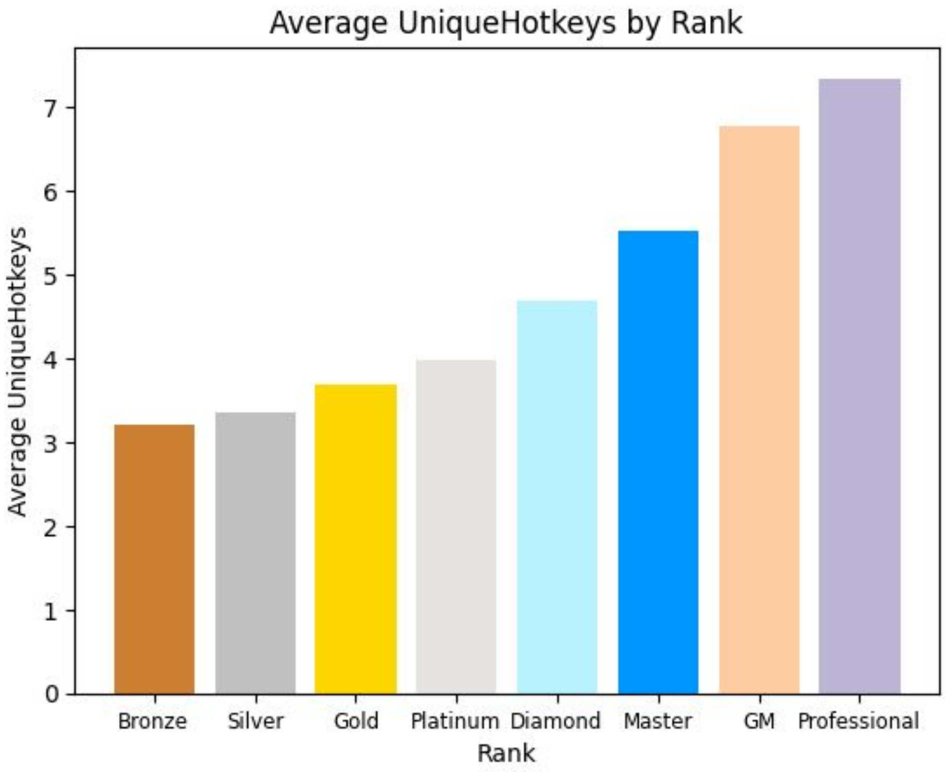
Number of right-clicks on minimap per timestamp was the ninth most important predictor in the model. It appears that a player with a higher number of right-clicks on minimap per timestamp was ranked higher.

Feature Interpretation - TotalMapExplored



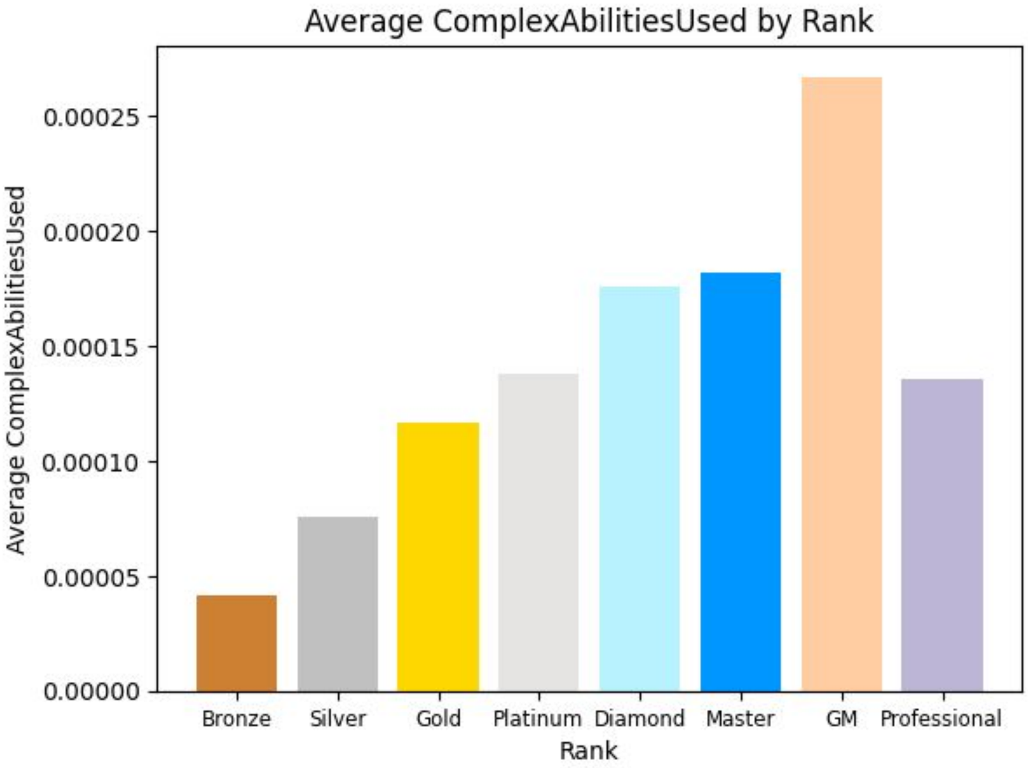
The number of 24x24 game coordinate grids viewed by the player per timestamp was the tenth most important predictor in the model. It appears that a player with a higher number of 24x24 game coordinate grids viewed by the player per timestamp was ranked higher, excluding Professional league players.

Feature Interpretation - UniqueHotkeys



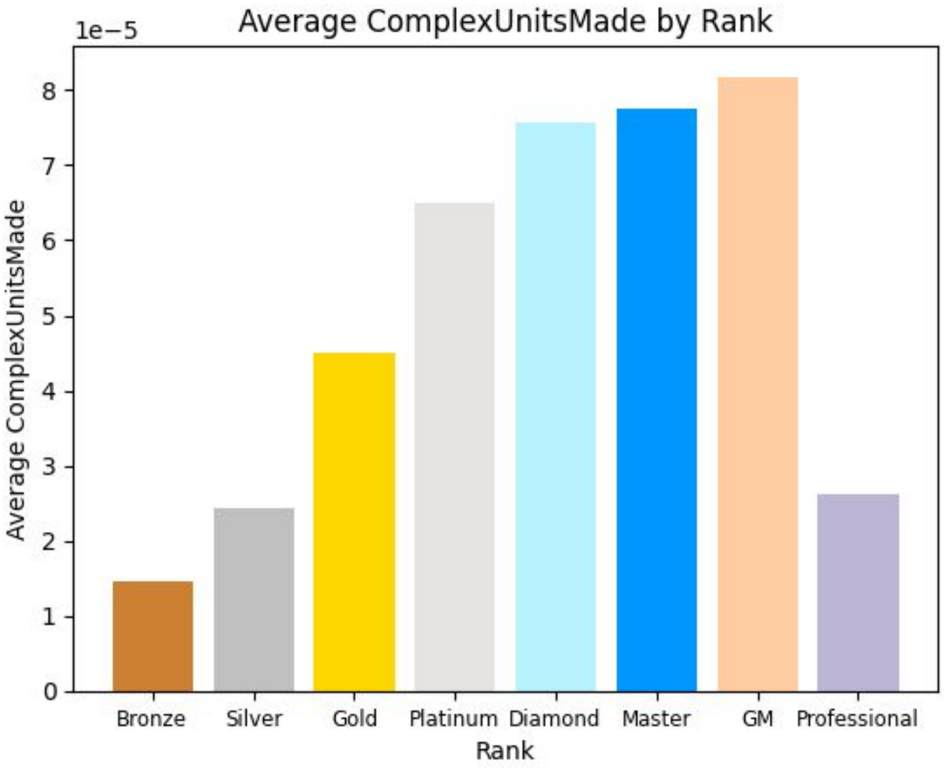
Number of unique hotkeys used per timestamp was the eleventh most important predictor in the model. It appears that a player with a higher number of unique hotkeys used per timestamp was ranked higher.

Feature Interpretation - ComplexAbilitiesUsed



Number of abilities requiring specific targeting instructions used per timestamp was the tweltfh most important predictor in the model. It appears that a player with a higher number of abilities requiring specific targeting instructions used per timestamp was ranked higher, except Professional league players.

Feature Interpretation - ComplexUnitsMade



Number of ghosts, infestors, and high templars trained per timestamp was the thirteenth most important predictor in the model. It appears that a player with a higher Number of ghosts, infestors, and high templars trained per timestamp was ranked higher, except Professional league players.

Further Data to Collect

- In order from most to least important, the following data should be collected
  - ActionLatency
  - APM
  - NumberOfPACs
  - SelectByHotkeys
  - GapBetweenPACs
  - AssignToHotkeys
  - MinimapAttacks
  - WorkersMade
  - MinimapRightClicks
  - TotalMapExplored
  - UniqueHokeys
  - ComplexAbilitiesUsed
  - ComplexUnitsMade

Data to Deprioritize

- The following data should be deprioritized for this particular task:
    - Age
    - HoursPerWeek
    - TotalHours
    - UniqueUnitsMade
    - ActionsInPAC
    - GameID

  These data points are largely uncorrelated with the rank of a player.