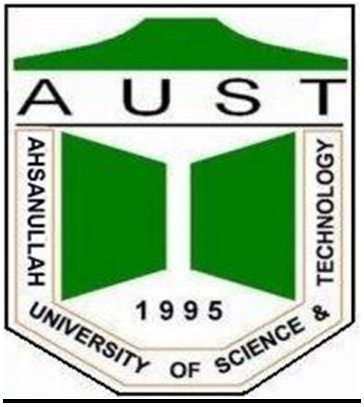
AHSANULLAH UNIVERSITY OF SCIENCE & TECHNOLOGY



Course No: CSE 4108

Course Name: Artificial Intelligence Lab

Section: B Lab Group: B2

Semester: Fall 2020

Report on Project: Valorant Game Result Prediction

Submitted to:

Tonmoy Hossain
Department of CSE, AUST.

Md. Siam Ansary
Department of CSE, AUST.

Submitted By:

Name: Md. Abir Hossain
ID: 170204106

Introduction:

Valorant is a 5V5 online tactical shooter game. A player can choose different types of characters, and maps in a match. The player has to kill his opponents, and do a certain task with his teammate to win the match. So to predict the match's outcome, a machine learning model can be used.

Dataset Description:

To predict the match's result, player's total kill, total death, average combat score, character choice and map choice values can be used. So the dataset contains 5 features of a match (kill, death, avg_score, agent, map), and the match result, whether the match won or lost. Character's names and map's names are replaced with numeric values and win/lose values are replaced with 1/0 respectively. There are total 100 rows in the dataset that represents 100 match history.

Description of Models

1. **K-Nearest Neighbor Classification:** Our first model for the project is K-nearest neighbor. K-nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure. KNN has been used in statistical estimation and pattern recognition. A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbor. We used K=7 for distance function in this model.

2. **Gaussian Naive Bayes Classification:** Our first model for the project is Gaussian Naive Bayes. Naive Bayes can be extended to real-valued attributes, most commonly by assuming a Gaussian distribution. This extension of naive Bayes is called Gaussian Naive Bayes. Other functions can be used to estimate the distribution of the data, but the Gaussian (or Normal distribution) is the easiest to work with because you only need to estimate the mean and the standard deviation from your training data.

Result Table:

K-Nearest Neighbor and Gaussian Naive Bayes Classification, these two models are used for prediction. For both models 66% of the data from dataset has been used for training the model, and the rest is used for testing the prediction. To compare the results four "Performance Metric" score is calculated. They are Accuracy, Recall, Precision, and F1 score.

The result comparison table is given below.

Classifiers	**Accuracy**	**Recall**	**Precision**	**F1 Score**
KNN	69.69%	85.0%	70.83%	77.27%
GNB	75.75%	77.5%	81.58%	79.48%

From the result table it is seen that **Gaussian Naive Bayes (GNB) Classification** has gained the highest accuracy, which is **75.75%.**

Conclusion:

After the model analysis here,

For the Gaussian Naive Bayes (GNB) Classification the Accuracy is75.75%, which is the highest. The Precision of GNB is 81.58%, which means it is correct around 81.58% of the time whether the player won or lost. The Recall of GNB is 77.5%, which means 77.5% of the times the model is been able to identify the relevant data.

For the K-Nearest Neighbor (KNN) Classification the Accuracy is 69.69%. The Precision of GNB is 70.83%, which means it is correct around 70.83% of the time whether the player won or lost. The Recall of GNB is 85.0%, which means 85.0% of the times the model is been able to identify the relevant data.

So to conclude that we can say that, The Gaussian Naive Bayes (GNB) Classification works well for the project.