Implementation ETL Report

Elsa Carlson, Jerad Ipsen, Shannon Bayless, Ryan-Arnold Gamilo

September 22nd, 2022

Introduction

Machine learning is vital to operations within companies of all types and is used for things like customer recommendations to fraud detection. For this project we were tasked with implementing the KNN (K-nearest neighbor) algorithm to a data set we have previously worked with and then compare with results to a similar algorithm. Score is the element we will be training the algorithm to predict based on modifiers and attributes such as wisdom and height. We will also be doing some feature engineering, as well as hyperparameter tuning after reviewing initial results.

Data Sources

We used the LASSO-data-set.csv provided in the model optimization exercise. This CSV contained data regarding the Dungeons & Dragons game stat system.

Extraction

This CSV was provided and readily able to be downloaded. We imported the csv into a pandas dataframe through Google Colab's python notebooks.

Initial Transformation

1. When doing a quick review of the data we found that column “module 8” had a large amount of outliers, those being the value of 11, but when reviewing DND documentation we found that this may be standard and decided to leave these in.
2. We found no null or 0 values so there was no need to do any imputations.
3. We did standardize the data using StandardScaler().fit(df) and set this to a scalar. Then we set df_scaled = pd.DataFrame(scaler.transform(df)) where df_scaled stored our standardized dataframe.
4. We then renamed the new columns of df_scaled to the original columns of df using df_scaled.columns = df.columns

Feature Engineering Transformations

1. We found that there were multiple columns that had low correlation with Score which we decided to remove in order to enhance the model.
    a. These included strength, dexterity, wisdom, intelligence, weight, modifier5, modifier6, modifier7, and modifier8.

Conclusion

There was little cleaning required in the process of preparing this data for algorithm training. We did however notice standardizing our data was useful in the training process and noticed an increase in the R^2 value by $.121$. Removing columns with low correlation did result in a lower R^2 value, but only by a small amount. The

models we compared KNN with, those being Linear Regression and LASSO, outperformed KNN, but these were results that we expected.