Campus Analytics Challenge 2021 - Rationale

For the Campus Analytics Challenge 2021, the task was to build a classification model to predict elder fraud in the digital payments space. My model had to handle missing variables, maximize the F1 score, and use the least amount of feature variables possible. For my model, I used a novel combination of existing machine learning methods, rather than develop my own algorithm. My approach made good use of the python libraries pandas, numpy, seaborn, and sklearn to help with visualizing the data and identifying possible correlations between various attributes of a transaction and the possibility of fraud. Once I read in the training dataset as a pandas dataframe, I filtered it to only have rows in which the customer age was 60 years and above. This step was taken to ensure that the model can specifically be trained to predict elder fraud more accurately. Filtering the dataset by customer age also enabled me to see certain patterns that may not be visible otherwise.

In addition to filtering out the dataset, I used the seaborn library to visualize and plot graphs between various columns of the data set and the FRAUD_NONFRAUD column. Using the seaborn graphs helped me to get a sense of which variables have a stronger correlation with fraud/nonfraud. I also added new columns for categorical variables by assigning them a numerical code based on the category. This enabled me to use categorical variables in modeling, as well as take care of missing values, as if there was a missing value, it would have a value of -1.

For the actual model, I decided to use the Random Forest Classifier from the sklearn library for several reasons. One reason is that the random forest algorithm is known for higher dimensionality and accuracy, which is important in fraud detection. Another reason is that random forest uses a multitude of decision trees with subsets of features, instead of just one, to reduce overfitting of the data. The final reason is that the random forest algorithm works well with categorical and continuous variables, which are inevitable in a transaction-related dataset. Despite these advantages, my first thought was not to use the random forest algorithm. Instead, I first tried using logistic regression and k-nearest neighbors. However, the F1 score for those two algorithms were only 0.84, and 0.90, respectively. The random forest algorithm gave a F1 score of 0.95, so it was naturally the best choice for this scenario. All in all, this challenge not only enabled me to work like a data scientist, but also increased my passion and drive for studying data science.