**Module 18 Challenge Report**

**Unsupervised Learning-Cryptocurrencies**

**Background**

Unsupervised learning is a Machine Learning technique that derives from test data that is unclassified, unlabeled or uncategorized depending on the presence or absence of common factors, as well as those patterns available in the data sample. This exercise is intended to demonstrate and highlight the process involved in the application of this particular technique.

**Objective**

The objective is to analyze the cryptocurrencies (crypto\_data.csv) dataset by applying unsupervised machine learning technique and processes.

**Development Environment** Python Pandas Jupyter Notebook Data file (crypto\_data.csv)

**Code Plan** 1. Read in or load the csv data file (crypto\_data.csv) and convert to a DataFrame. 2. Apply pre-processing, which includes data transformation and data cleaning as necessary. 3. Convert non-numerical data to numerical. In this case create a dummy variable for a DataFrame 4. Standardize the data In the DataFrame using Pandas StandardScaler() 5. Reduce the dimensions of the DataFrame down to three principal components 6. Derive the best K value 7. Create an Elbow Curve 8. Initialize K-means Model, Model fit and Model predict 9. Create original DataFrame with column for calss 10. Cluster and Plot the data using 3D scatter plot

**Summary**

Pre-processing of data is the first step in this exercise. This describes the process in which we inspect the data and manage unnecessary columns, rows with null values, NaN, mixed data types and formatting.

Data is then scaled using the StandardScaler function. The purpose of this is to standardize all of the data in the DataFrame. After that, we apply the PCA technique to speed up the machine learning algorithm where the number of input features are too high. In other words, PCA transforms a larger set of variables into smaller one, which contains most of the information in the original large set.

Since our data contains a set of clusters, we apply an algorithm that determines the best K values. The model is then trained with the k-means algorithm, which iterates and determines the best centroid for each and every K cluster values.

Data visualization is done with the application of 2D and 3D hvplot scatter plot in plotly library for plotting the outcome or result of the analyzed data. The K-means algorithm also use the hvplot.line to determine the best K value from the Elbow Curve.

Note: For the purpose of improved visualization, I scaled the data and re-plot both 2D and 3D plots. However, the data appeared to be significantly reduced as shown on the plots. Further investigation is required to understand exactly why this happened.