**Rubric for Unsupervised Learning:**

| | **Mastery**<br>**30 to > 27 points** | **Approaching Mastery**<br>**27 to > 24 points** | **Progressing**<br>**24 to > 20 points** | **Emerging**<br>**20 to > 0 points** | **Incomplete** |
|---|---|---|---|---|---|
| **Deliverable 1: Preprocessing the Data for PCA** | The following preprocessing steps have been performed on the crypto_df DataFrame: (16 pt)<br>● ✓ All cryptocurrencies that are not being traded are removed<br>● ✓ All cryptocurrencies that do not have a defined algorithm are removed<br>● ✓ All cryptocurrencies with at least one null value are dropped<br>● ✓ All the rows that do not have coins being mined are removed<br>● ✓ The IsTrading column is dropped<br>● ✓ The CoinName column is dropped from the crypto_df DataFrame<br>AND all the following have been completed.<br>✓ A new DataFrame is created that stores the names of all cryptocurrencies from the CoinName column and has the index from the crypto_df DataFrame (4 pt)<br>✓ The get_dummies() method is used to create variables for all of the text features, which are then stored in a new DataFrame, X (5 pt)<br>✓ The features from the X DataFrame have been standardized using the StandardScaler fit_transform() function (5 pt) | The following preprocessing steps have been performed on the crypto_df DataFrame: (16 pt)<br>● ✓ All cryptocurrencies that are not being traded are removed<br>● ✓ All cryptocurrencies that do not have a defined algorithm are removed<br>● ✓ All cryptocurrencies with at least one null value are dropped<br>● ✓ All the rows that do not have coins being mined are removed<br>● ✓ The IsTrading column is dropped<br>● ✓ The CoinName column is dropped from the crypto_df DataFrame<br>AND all the following have been completed.<br>✓ A new DataFrame is created that stores the names of all cryptocurrencies with the CoinName column and has the index from the crypto_df DataFrame (4 pt)<br>✓ The get_dummies() method is used to create variables for the TWO text features, which are then stored in a new DataFrame, X (5 pt)<br>✓ The features from the X DataFrame have been standardized using the StandardScaler fit_transform() function, but there is an error (2 pt) | The following preprocessing steps have been performed on the crypto_df DataFrame: (14 pt)<br>● ✓ All cryptocurrencies that are not being traded are removed<br>● ✓ All cryptocurrencies that do not have a defined algorithm are removed<br>● ✓ All cryptocurrencies with at least one null value are dropped<br>● ✓ All the rows that do not have coins being mined are removed<br>● ✓ The IsTrading column is dropped<br>AND all the following have been completed.<br>✓ A new DataFrame is created that stores the names of all cryptocurrencies with the CoinName column and has the index from the crypto_df DataFrame (4 pt)<br>✓ The get_dummies() method is used to create variables for the ONE of the TWO text features, which are then stored in a new DataFrame, X (3 pt)<br>✓ The features from the X DataFrame have been standardized using the StandardScaler fit_transform() function (3 pt) | The following preprocessing steps have been performed on the crypto_df DataFrame: (12 pt)<br>● ✓ All cryptocurrencies that are not being traded are removed<br>● ✓ All cryptocurrencies that do not have a defined algorithm are removed<br>● ✓ All cryptocurrencies with at least one null value are dropped<br>● ✓ All the rows that do not have coins being mined are removed<br>AND all the following have been completed.<br>✓ A new DataFrame is created that stores the names of all cryptocurrencies with the CoinName column BUT does not have the index from the crypto_df DataFrame (2 pt)<br>✓ The get_dummies() method is used to create variables for the ONE of the TWO text features, which are then stored in a new DataFrame, X (3 pt)<br>✓ The features from the X DataFrame have been standardized using the StandardScaler fit_transform() function (3 pt) | No submission was received<br>-OR-<br>Submission was empty or blank<br>-OR-<br>Submission contains evidence of academic dishonesty |

| | **Mastery**<br>**20 to > 18 points** | **Approaching Mastery**<br>**18 to > 16 points** | **Progressing**<br>**16 to > 14 points** | **Emerging**<br>**14 to > 0 points** | |
|---|---|---|---|---|---|
| **Deliverable 2: Reducing Data Dimensions Using PCA** | ✓ The PCA algorithm reduces the dimensions of the X DataFrame down to three principal components. (10 pt)<br>✓The pcs_df DataFrame is created (5 pt)<br>And, all the following are completed:<br>✓ The pcs_df DataFrame has three columns; PC 1, PC 2, and PC 3 (2 pt)<br>✓ The pcs_df DataFrame uses the index from the crypto_df DataFrame (3 pt) | ✓ The PCA algorithm reduces the dimensions of the X DataFrame down to three principal components. (10 pt)<br>✓The pcs_df DataFrame is created (5 pt)<br>And, ONE of the following are completed:<br>✓ The pcs_df DataFrame has three columns; PC 1, PC 2, and PC 3 (2 pt)<br>✓ The pcs_df DataFrame uses the index from the crypto_df DataFrame (3 pt) | ✓ The PCA algorithm reduces the dimensions of the X DataFrame down to three principal components. (10 pt)<br>✓ The pcs_df DataFrame is created (5 pt)<br>✓ Code is written to add the three columns; PC 1, PC 2, and PC 3 to the pcs_df DataFrame (1 pt) | ✓ The PCA algorithm reduces the dimensions of the X DataFrame down to three principal components. (10 pt)<br>✓ Code is written to create the pcs_df DataFrame with the three columns; PC 1, PC 2, and PC 3 (4 pt) | |
| | **Mastery**<br>**20 to > 18 points** | **Approaching Mastery**<br>**18 to > 15 points** | **Progressing**<br>**15 to > 12 points** | **Emerging**<br>**12 to > 0 points** | |
| **Deliverable 3: Clustering Cryptocurrencies Using K-means** | ✓ An Elbow Curve is created using hvPlot to find the best value for K (10 pt)<br>✓ Predictions are made on the K clusters of the cryptocurrencies' data (5 pt)<br>✓ A new DataFrame is created with the same index as the crypto_df DataFrame and has NINE columns (5 pt) | ✓ An Elbow Curve is created using hvPlot to find the best value for K (10 pt)<br>✓ Predictions are made on the K clusters of the cryptocurrencies' data (5 pt)<br>✓ A new DataFrame is created with the same index as the crypto_df DataFrame and has SEVEN of the NINE columns (3 pt) | ✓ An Elbow Curve is created using hvPlot to find the best value for K (10 pt)<br>✓ Predictions are made on the K clusters of the cryptocurrencies' data (5 pt) | ✓ An Elbow Curve is created using hvPlot to find the best value for K (10 pt)<br>✓ Code is written to make the predictions on the K clusters of the cryptocurrencies' data (2 pt) | |
| | **Mastery**<br>**30 to > 27 points** | **Approaching Mastery**<br>**27 to > 25 points** | **Progressing**<br>**25 to > 22 points** | **Emerging**<br>**22 to > 0 points** | |
| **Deliverable 4: Visualizing Cryptocurrencies Results** | ✓ The clusters are plotted using a 3D-Scatter and each data point shows the CoinName and Algorithm on hover (10 pt)<br>✓ A table with tradable cryptocurrencies is created using the hvplot.table() function (3 pt) | ✓ The clusters are plotted using a 3D-Scatter and each data point shows the CoinName and Algorithm on hover (10 pt)<br>✓ A table with tradable cryptocurrencies is created using the hvplot.table() function (3 pt) | ✓ The clusters are plotted using a 3D-Scatter and each data point shows the CoinName and Algorithm on hover (10 pt)<br>✓ A table with tradable cryptocurrencies is created using the hvplot.table() function (3 pt) | ✓ The clusters are plotted using a 3D-Scatter and each data point shows the CoinName and Algorithm on hover (10 pt)<br>✓ A table with tradable cryptocurrencies is created using the hvplot.table() function (3 pt) | |

| | | | | | |
|---|---|---|---|---|---|
| | ✓ The total number of tradable cryptocurrencies is printed (2 pt)<br>✓ A DataFrame is created that contains the clustered_df DataFrame index, the scaled data, and the “CoinName” and “Class” columns (5 pt)<br>✓ A hvplot scatter plot is created where the X-axis is “TotalCoinsMined”, the Y-axis is “TotalCoinSupply”, and the data is ordered by “Class”, and when you hover over the the data it shows the “CoinName” (10 pt) | ✓ The total number of tradable cryptocurrencies is printed (2 pt)<br>✓ A DataFrame is created that contains the clustered_df DataFrame index, the scaled data, and the “CoinName” and “Class” columns (5 pt)<br>✓ A hvplot scatter plot is created where the X-axis is “TotalCoinsMined”, the Y-axis is “TotalCoinSupply, and the data is ordered by “Class”, but the “CoinName” doesn’t appear on hover (7 pt) | ✓ The total number of tradable cryptocurrencies is printed (2 pt)<br>✓ A DataFrame is created that contains the scaled data, the clustered_df DataFrame index, and the “CoinName” column (3 pt)<br>✓ A hvplot scatter plot is created where the X-axis is “TotalCoinsMined”, the Y-axis is “TotalCoinSupply, and when you hover over the the data it shows the “CoinName” (7 pt) | ✓ The total number of tradable cryptocurrencies is printed (2 pt)<br>✓ A DataFrame is created that contains the scaled data and the clustered_df DataFrame index. (2 pt)<br>✓ A hvplot scatter plot is created where the X-axis is “TotalCoinsMined”, the Y-axis is “TotalCoinSupply” (5 pt) | |