

Rubric for Unsupervised Learning:

	Mastery 30 to > 27 points	Approaching Mastery 27 to > 24 points	Progressing 24 to > 20 points	Emerging 20 to > 0 points	Incomplete
Deliverable 1: Preprocessing the Data for PCA	<p>The following preprocessing steps have been performed on the crypto_df DataFrame: (16 pt)</p> <ul style="list-style-type: none"> ✓ All cryptocurrencies that are not being traded are removed ✓ All cryptocurrencies with at least one null value are dropped ✓ All the rows that do not have coins being mined are removed ✓ The IsTrading column is dropped ✓ The CoinName column is dropped from the crypto_df DataFrame <p>AND all the following have been completed.</p> <ul style="list-style-type: none"> ✓ 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) ✓ 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) ✓ The features from the X DataFrame have been standardized using the StandardScaler fit_transform() function (5 pt) 	<p>The following preprocessing steps have been performed on the crypto_df DataFrame: (16 pt)</p> <ul style="list-style-type: none"> ✓ All cryptocurrencies that are not being traded are removed ✓ All cryptocurrencies with at least one null value are dropped ✓ All the rows that do not have coins being mined are removed ✓ The IsTrading column is dropped ✓ The CoinName column is dropped from the crypto_df DataFrame <p>AND all the following have been completed.</p> <ul style="list-style-type: none"> ✓ 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) ✓ 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) ✓ The features from the X DataFrame have been standardized using the StandardScaler fit_transform() function, but there is an error (2 pt) 	<p>The following preprocessing steps have been performed on the crypto_df DataFrame: (14 pt)</p> <ul style="list-style-type: none"> ✓ All cryptocurrencies that are not being traded are removed ✓ All cryptocurrencies with at least one null value are dropped ✓ All the rows that do not have coins being mined are removed ✓ The IsTrading column is dropped <p>AND all the following have been completed.</p> <ul style="list-style-type: none"> ✓ 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) ✓ 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) ✓ The features from the X DataFrame have been standardized using the StandardScaler fit_transform() function (3 pt) 	<p>The following preprocessing steps have been performed on the crypto_df DataFrame: (12 pt)</p> <ul style="list-style-type: none"> ✓ All cryptocurrencies that are not being traded are removed ✓ All cryptocurrencies with at least one null value are dropped ✓ All the rows that do not have coins being mined are removed <p>AND all the following have been completed.</p> <ul style="list-style-type: none"> ✓ 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) ✓ 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) ✓ The features from the X DataFrame have been standardized using the StandardScaler fit_transform() function (3 pt) 	<p>No submission was received</p> <p>-OR-</p> <p>Submission was empty or blank</p> <p>-OR-</p> <p>Submission contains evidence of academic dishonesty</p>
	Mastery 20 to > 18 points	Approaching Mastery 18 to > 16 points	Progressing 16 to > 14 points	Emerging 14 to > 0 points	

Deliverable 2: Reducing Data Dimensions Using PCA	<ul style="list-style-type: none"> ✓ The PCA algorithm reduces the dimensions of the X DataFrame down to three principal components. (10 pt) ✓ The pcs_df DataFrame is created (5 pt) <p>And, all the following are completed:</p> <ul style="list-style-type: none"> ✓ The pcs_df DataFrame has three columns; PC 1, PC 2, and PC 3 (2 pt) ✓ The pcs_df DataFrame uses the index from the crypto_df DataFrame (3 pt) 	<ul style="list-style-type: none"> ✓ The PCA algorithm reduces the dimensions of the X DataFrame down to three principal components. (10 pt) ✓ The pcs_df DataFrame is created (5 pt) <p>And, ONE of the following are completed:</p> <ul style="list-style-type: none"> ✓ The pcs_df DataFrame has three columns; PC 1, PC 2, and PC 3 (2 pt) ✓ The pcs_df DataFrame uses the index from the crypto_df DataFrame (3 pt) 	<ul style="list-style-type: none"> ✓ The PCA algorithm reduces the dimensions of the X DataFrame down to three principal components. (10 pt) ✓ The pcs_df DataFrame is created (5 pt) ✓ Code is written to add the three columns; PC 1, PC 2, and PC 3 to the pcs_df DataFrame (1 pt) 	<ul style="list-style-type: none"> ✓ The PCA algorithm reduces the dimensions of the X DataFrame down to three principal components. (10 pt) ✓ Code is written to create the pcs_df DataFrame with the three columns; PC 1, PC 2, and PC 3 (4 pt) 	
	Mastery 20 to > 18 points	Approaching Mastery 18 to > 15 points	Progressing 15 to > 12 points	Emerging 12 to > 0 points	
Deliverable 3: Clustering Cryptocurrencies Using K-means	<ul style="list-style-type: none"> ✓ An Elbow Curve is created using hvPlot to find the best value for K (10 pt) ✓ Predictions are made on the K clusters of the cryptocurrencies' data (5 pt) ✓ A new DataFrame is created with the same index as the crypto_df DataFrame and has NINE columns (5 pt) 	<ul style="list-style-type: none"> ✓ An Elbow Curve is created using hvPlot to find the best value for K (10 pt) ✓ Predictions are made on the K clusters of the cryptocurrencies' data (5 pt) ✓ A new DataFrame is created with the same index as the crypto_df DataFrame and has SEVEN of the NINE columns (3 pt) 	<ul style="list-style-type: none"> ✓ An Elbow Curve is created using hvPlot to find the best value for K (10 pt) ✓ Predictions are made on the K clusters of the cryptocurrencies' data (5 pt) 	<ul style="list-style-type: none"> ✓ An Elbow Curve is created using hvPlot to find the best value for K (10 pt) ✓ Code is written to make the predictions on the K clusters of the cryptocurrencies' data (2 pt) 	
	Mastery 30 to > 27 points	Approaching Mastery 27 to > 25 points	Progressing 25 to > 22 points	Emerging 22 to > 0 points	
Deliverable 4: Visualizing Cryptocurrencies Results	<ul style="list-style-type: none"> ✓ The clusters are plotted using a 3D-Scatter and each data point shows the CoinName and Algorithm on hover (10 pt) ✓ A table with tradable cryptocurrencies is created using the hvplot.table() function (3 pt) ✓ The total number of tradable cryptocurrencies is printed (2 pt) 	<ul style="list-style-type: none"> ✓ The clusters are plotted using a 3D-Scatter and each data point shows the CoinName and Algorithm on hover (10 pt) ✓ A table with tradable cryptocurrencies is created using the hvplot.table() function (3 pt) ✓ The total number of tradable cryptocurrencies is printed (2 pt) 	<ul style="list-style-type: none"> ✓ The clusters are plotted using a 3D-Scatter and each data point shows the CoinName and Algorithm on hover (10 pt) ✓ A table with tradable cryptocurrencies is created using the hvplot.table() function (3 pt) ✓ The total number of tradable cryptocurrencies is printed (2 pt) 	<ul style="list-style-type: none"> ✓ The clusters are plotted using a 3D-Scatter and each data point shows the CoinName and Algorithm on hover (10 pt) ✓ A table with tradable cryptocurrencies is created using the hvplot.table() function (3 pt) ✓ The total number of tradable cryptocurrencies is printed (2 pt) 	

	<p>✓ A DataFrame is created that contains the clustered_df DataFrame index, the scaled data, and the "CoinName" and "Class" columns (5 pt)</p> <p>✓ 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 data it shows the "CoinName" (10 pt)</p>	<p>✓ A DataFrame is created that contains the clustered_df DataFrame index, the scaled data, and the "CoinName" and "Class" columns (5 pt)</p> <p>✓ 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)</p>	<p>✓ A DataFrame is created that contains the scaled data, the clustered_df DataFrame index, and the "CoinName" column (3 pt)</p> <p>✓ A hvplot scatter plot is created where the X-axis is "TotalCoinsMined", the Y-axis is "TotalCoinSupply", and when you hover over the data it shows the "CoinName" (7 pt)</p>	<p>✓ A DataFrame is created that contains the scaled data and the clustered_df DataFrame index. (2 pt)</p> <p>✓ A hvplot scatter plot is created where the X-axis is "TotalCoinsMined", the Y-axis is "TotalCoinSupply" (5 pt)</p>	
--	---	--	---	--	--