



Unsupervised Learning

Citizen Analytics – An Initiative by Data Science Team

START ►

© 2020 Petroliaam Nasional Berhad (PETRONAS)

All rights reserved. No part of this document may be reproduced in any form possible, stored in a retrieval system, transmitted and/or disseminated in any form or by any means (digital, mechanical, hard copy, recording or otherwise) without the permission of the copyright owner.

Learning Objectives

By the end of this module, you will be able to:



01

Familiarize with the most commonly used techniques in unsupervised learning.

02

Understand the concept of K-Means algorithm and perform it in Azure ML.

03

Understand the concept of Principal Component Analysis (PCA) and perform it in Azure ML.

Content

01. Clustering	04
a. What is Clustering?	
b. K-Means algorithm	
c. Steps to perform K-Means algorithm	
d. Cluster analysis in Azure ML	
02. Principal Component Analysis	26
a. Curse of dimensionality	
b. Principal Component Analysis	
c. Principal Component Analysis in Azure ML	
03. Summary	32

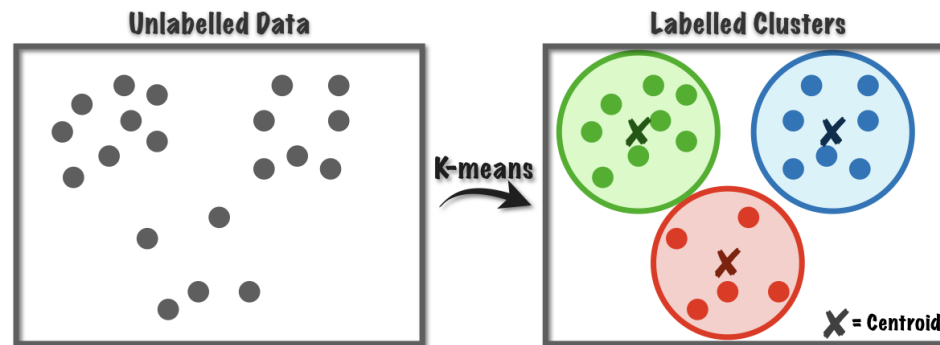
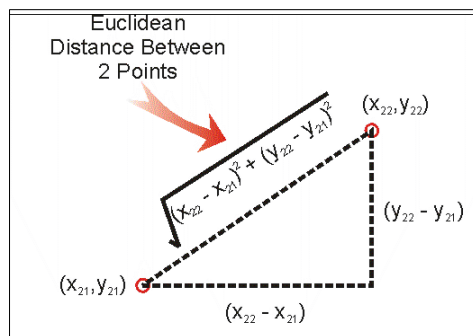
Clustering

What is clustering?

- Clustering or cluster analysis is an area of unsupervised learning. Unsupervised learning - (*`I don't know what I don't know`*). It is an area where we have no labels. The idea is to identify the hidden insights or structures from the data, and provide the same to businesses.
- Pattern mining and clustering are major approaches here in unsupervised learning.
- **The purpose of conducting clustering:**
 - To understand the behavioral differences amongst the different cluster segments
i.e. find out the insights which demarcates cluster segments and assist businesses to target and cater the needs of respective segments in a better way.
- **Examples of clustering:**
 - Recommendation Engine
 - Market Segmentation
 - Social Network Analysis
 - Medical/Health
 - Image Segmentation
 - Anomaly Detection

K-Means Algorithm

- K-Means clustering is one of the most commonly used clustering algorithms for partitioning observations into a set of k groups (i.e. k clusters), where k is pre-specified by the analyst.
- K-Means algorithm is an iterative algorithm that tries to partition the dataset into K pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group.
- It tries to make the inter-cluster data points as similar as possible while also keeping the clusters as different (far) as possible.
- It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum.
- The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.



K Means Clustering Intuition



K-Means Clustering

Algorithm Intuition

- 1 Choose the value of K for number of clusters
- 2 Randomly select K points, the centroid (may not be one of the observation from dataset)
- 3 Assign each data point to the closest centroid to form K clusters
- 4 Compute and place the new centroid of each cluster
- 5 Reassign each data point to the new closest centroid
- 6 If reassigned to new, repeat Step 4, otherwise finish.

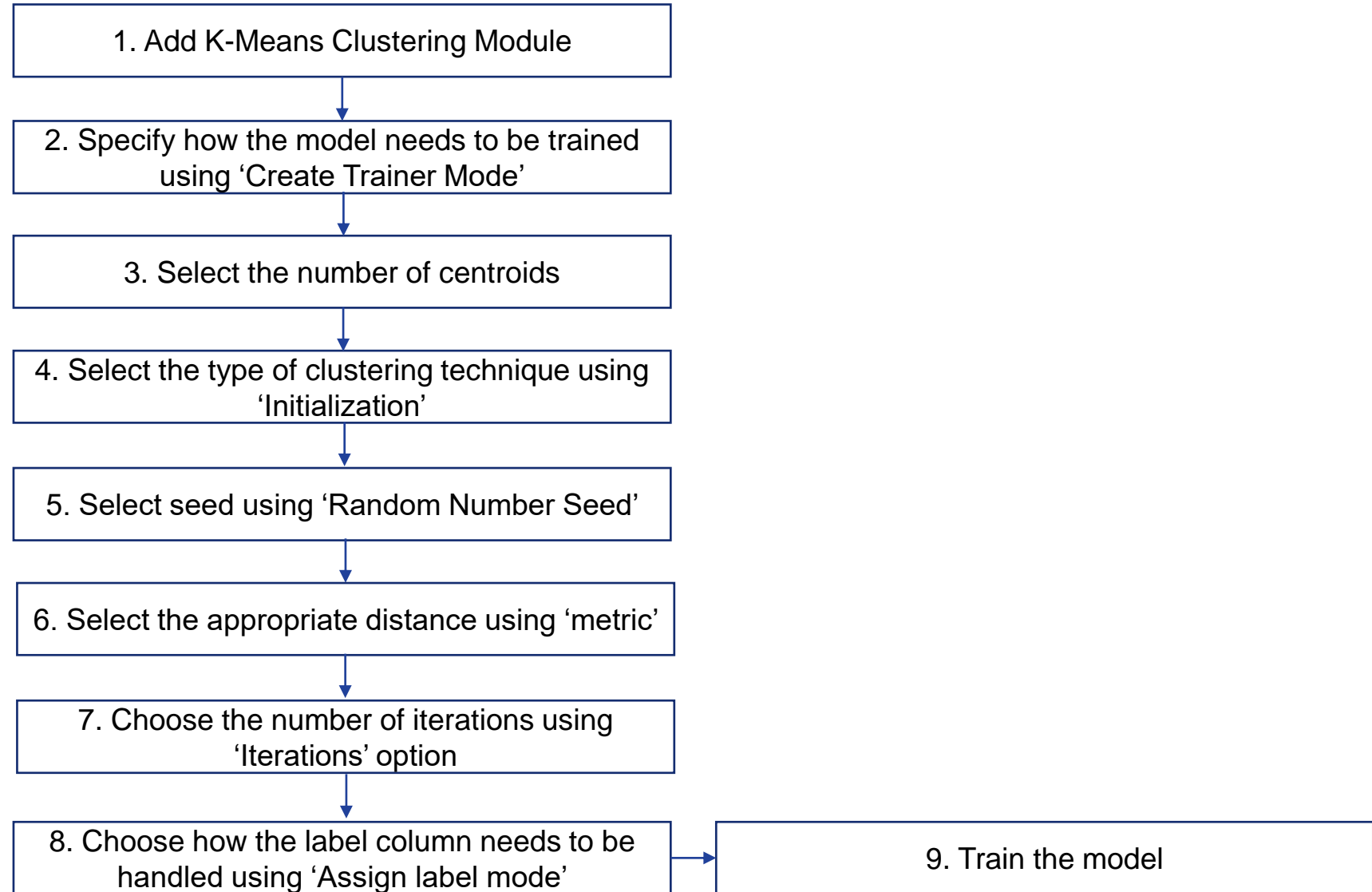
Finding suitable number of clusters



Elbow plot / Scree plot

Elbow method gives us an idea on what a good k number of clusters would be based on the sum of squared distance (SSE) between data points and their assigned clusters' centroids. We pick k at the spot where SSE starts to flatten out and forming an elbow.

Steps to perform K-Means algorithm



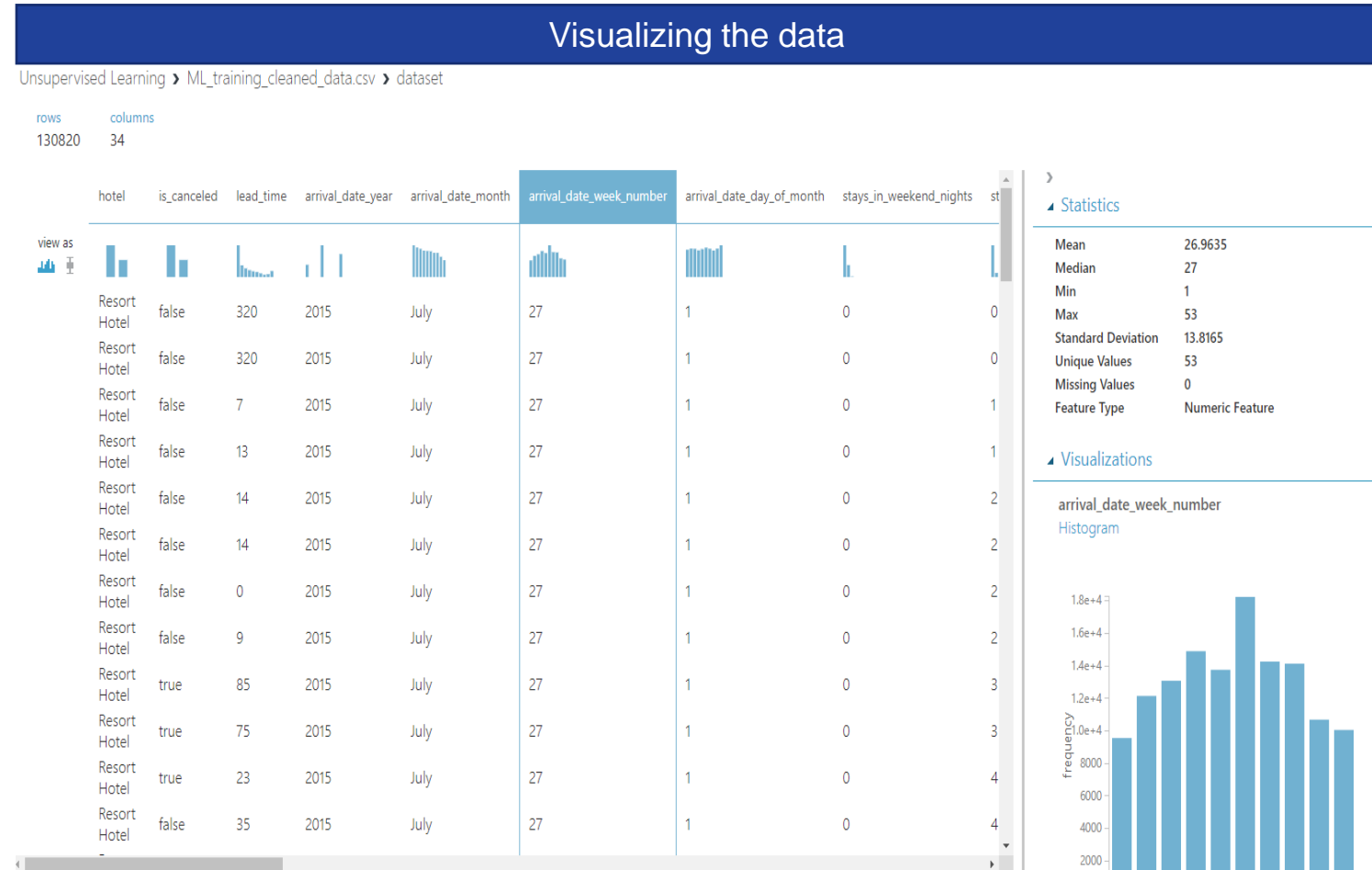
Cluster Analysis in Azure ML

1. Load the data

The screenshot displays the Azure ML studio interface during the 'Loading the data' phase of an 'Unsupervised Learning' experiment. On the left, a sidebar contains a search bar and a list of experiment items: Saved Datasets, Data Format Conversions, Data Input and Output, Data Transformation, Feature Selection, Machine Learning, OpenCV Library Modules, Python Language Modules, R Language Modules, Statistical Functions, Text Analytics, Time Series, Web Service, and Deprecated. The main workspace on the right shows a single data input node labeled 'ML_training_cleaned_data.csv'.

Cluster Analysis in Azure ML

2. Right click on the data module and select Visualize



Cluster Analysis in Azure ML

3. Add the K-Means Clustering module and select the parameters as below in the Properties section of it.

- Create Trainer Mode – **'Single Parameter'**, If you know the exact parameters you want to use in the clustering model, you can provide a specific set of values as arguments
- Number of Centroids – **4**, type the number of clusters you want the algorithm to begin with
- Initialization – **K-Means++**- K-means ++ improves upon standard K-means by using a different method for choosing the initial cluster centers.
- Random Number Seed – **123** - optionally type a value to use as the seed for the cluster initialization
- Metric – **Euclidean** - The Euclidean distance is commonly used as a measure of cluster scatter for K-means clustering. This metric is preferred because it minimizes the mean distance between points and the centroids
- Iterations – **100** - number of times the algorithm should iterate over the training data before finalizing the selection of centroids
- Assign label mode – **Ignore label column** -The values in the label column are ignored and are not used in building the model.

Adding the K-Means clustering module

Unsupervised Learning

ML_training_cleaned_data.csv

K-Means Clustering

Machine Learning

Initialize Model

Clustering

K-Means Clustering

Score

Assign Data to Clusters

Train

Sweep Clustering

Train Clustering Model

Deprecated

Assign to Clusters

Properties

Project

K-Means Clustering

Create trainer mode

Single Parameter

Number of Centroids

4

Initialization

K-Means++

Random number seed

123

Metric

Euclidean

Iterations

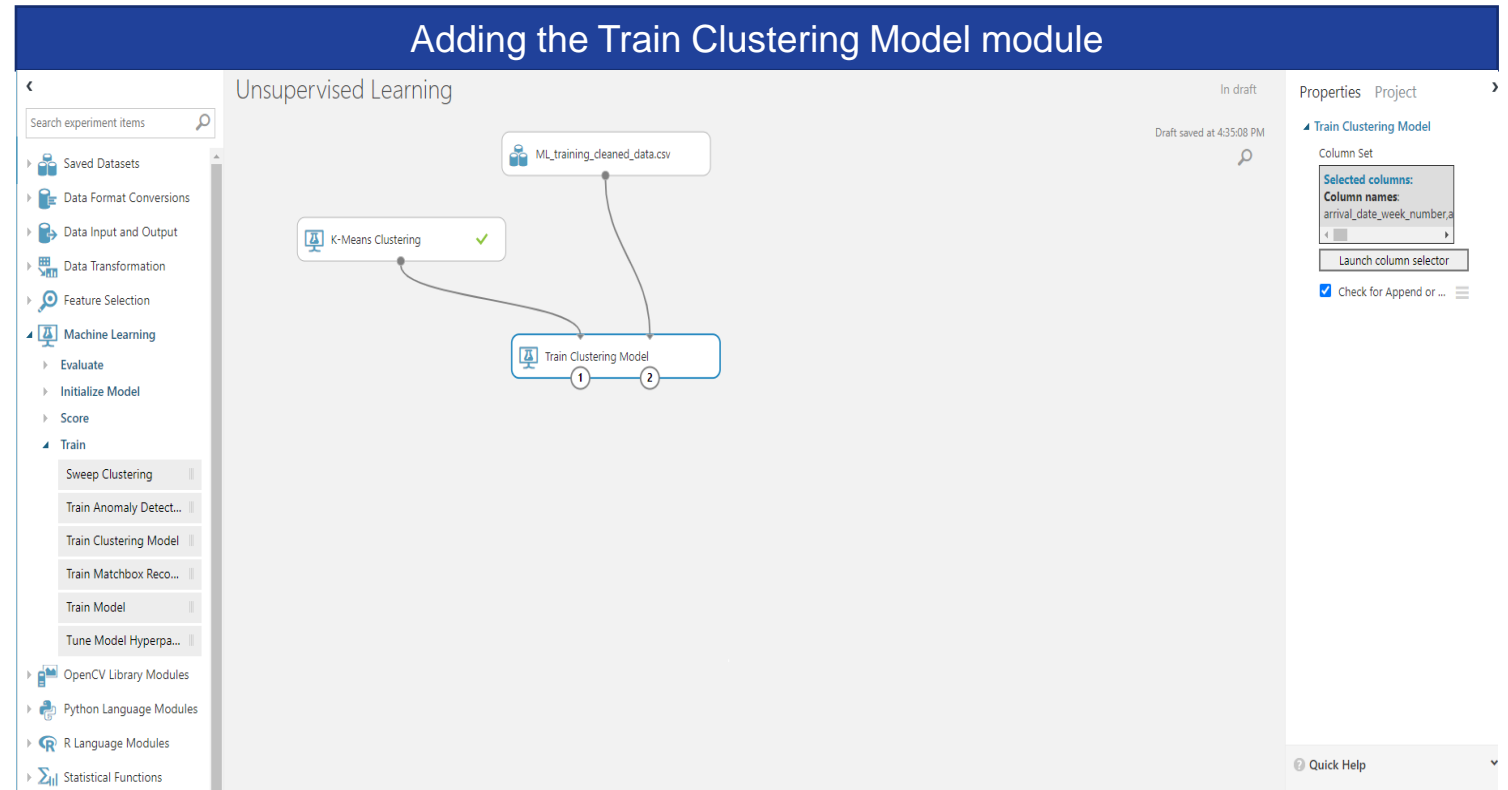
100

Assign Label Mode

Ignore label column

Cluster Analysis in Azure ML

4. Add 'Train Clustering Model' module



Cluster Analysis in Azure ML

5. Select the relevant features from the 'Launch Column Selector' in the Properties tab of 'Train Clustering Model' module

SELECTING THE RELEVANT COLUMNS

SELECT columns

BY NAME

WITH RULES

AVAILABLE COLUMNS

All Types search columns

hotel
is_canceled
lead_time
arrival_date_year
arrival_date_month
adults
country
market_segment
distribution_channel
is_repeated_guest
previous_cancellations
previous_bookings_not_canceled
reserved_room_type
assigned_room_type
company

18 columns available

SELECTED COLUMNS

All Types search columns

arrival_date_week_number
arrival_date_day_of_month
stays_in_weekend_nights
stays_in_week_nights
children
babies
meal
booking_changes
deposit_type
agent
days_in_waiting_list
customer_type
required_car_parking_spaces
total_of_special_requests
Add(children_adults)

16 columns selected

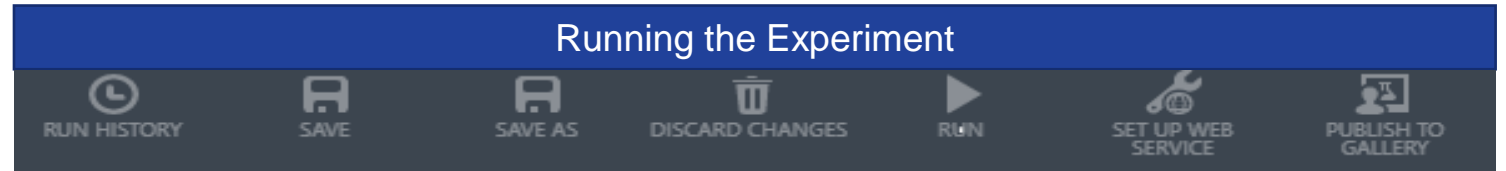
>

<

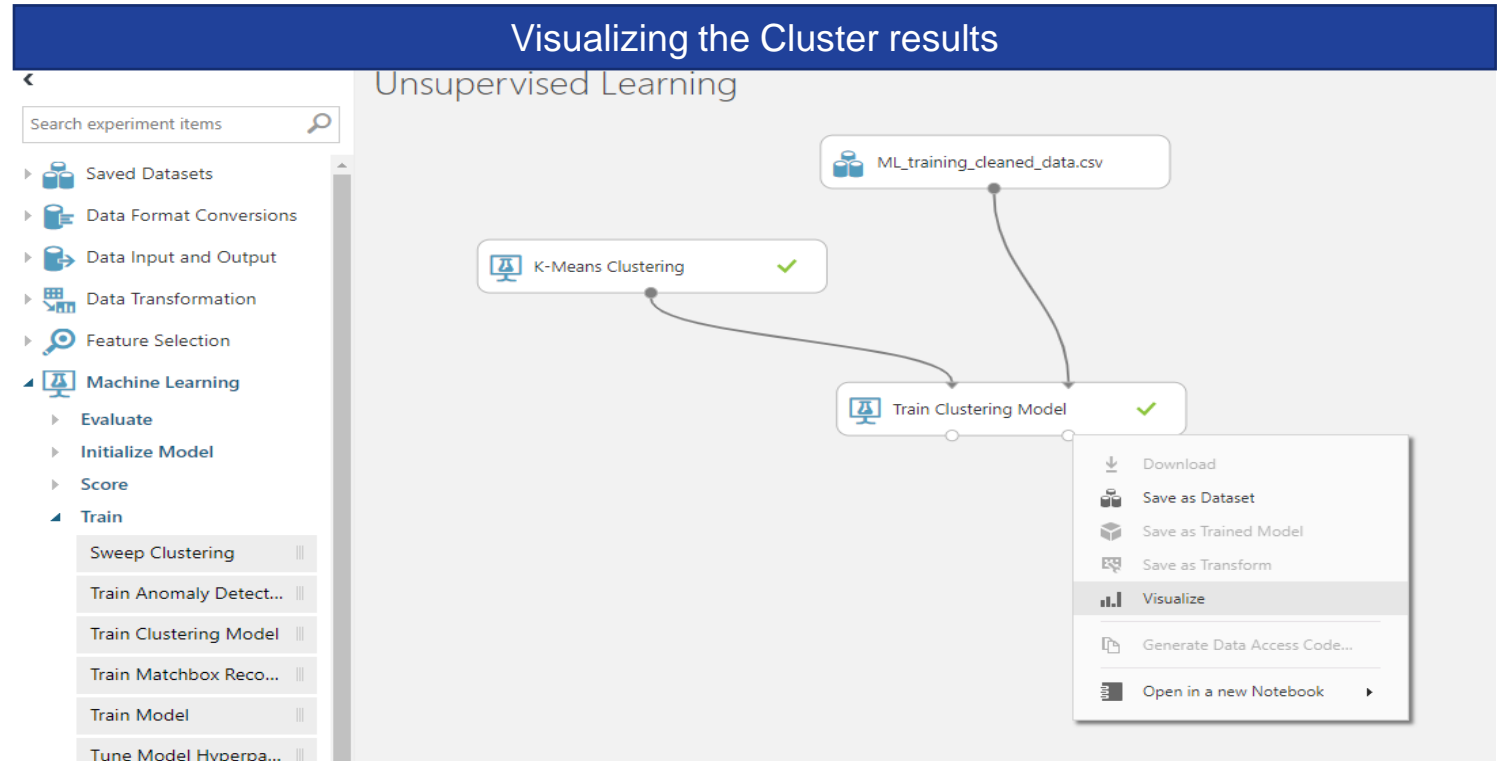
✓

Cluster Analysis in Azure ML

6. Run the experiment



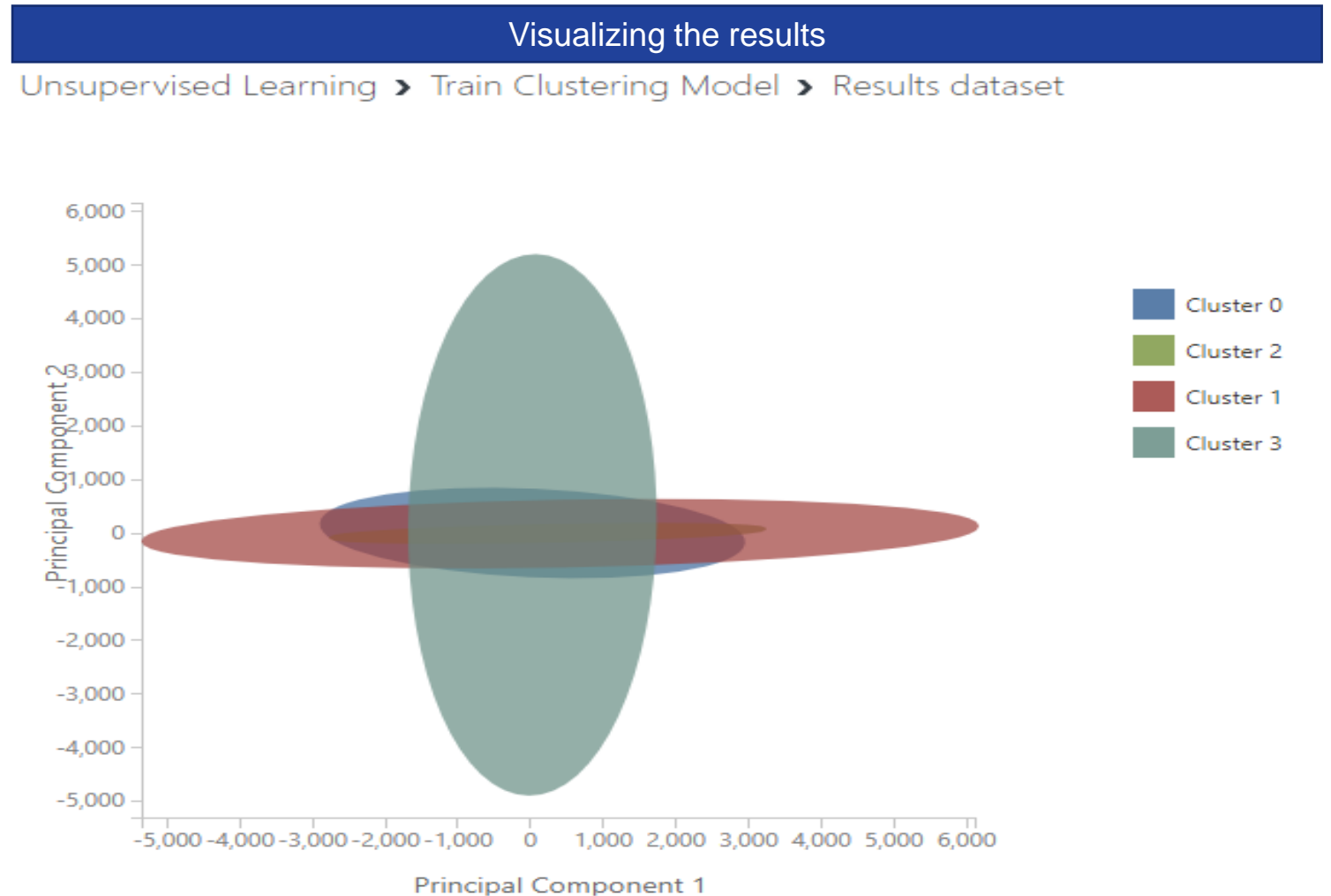
7. Visualize the Clustering results by running 'Visualize' option in 'Train Clustering Module'



Cluster Analysis in Azure ML

8. Visualizing the Results

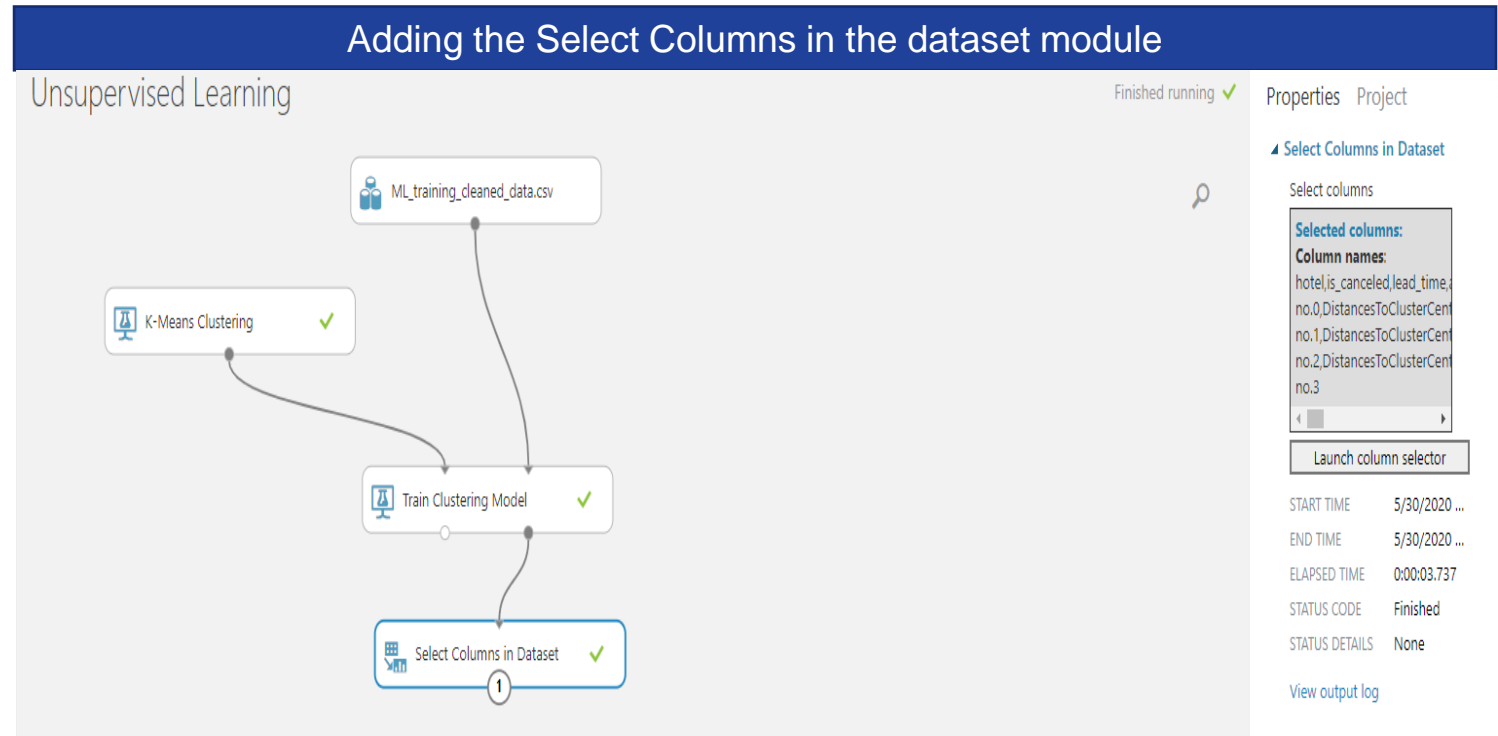
- The chart is generated by using Principal Component Analysis, which is a technique in data science for compressing the feature space of a model.
- The chart shows some set of features, compressed into four dimensions, that best characterize the difference between the clusters.
- By visually reviewing the general size of the feature space for each cluster and how much the clusters overlap, you can get an idea of how well your model might perform



Cluster Analysis in Azure ML

9. Let us try to examine the clustering results in a more intuitive way.

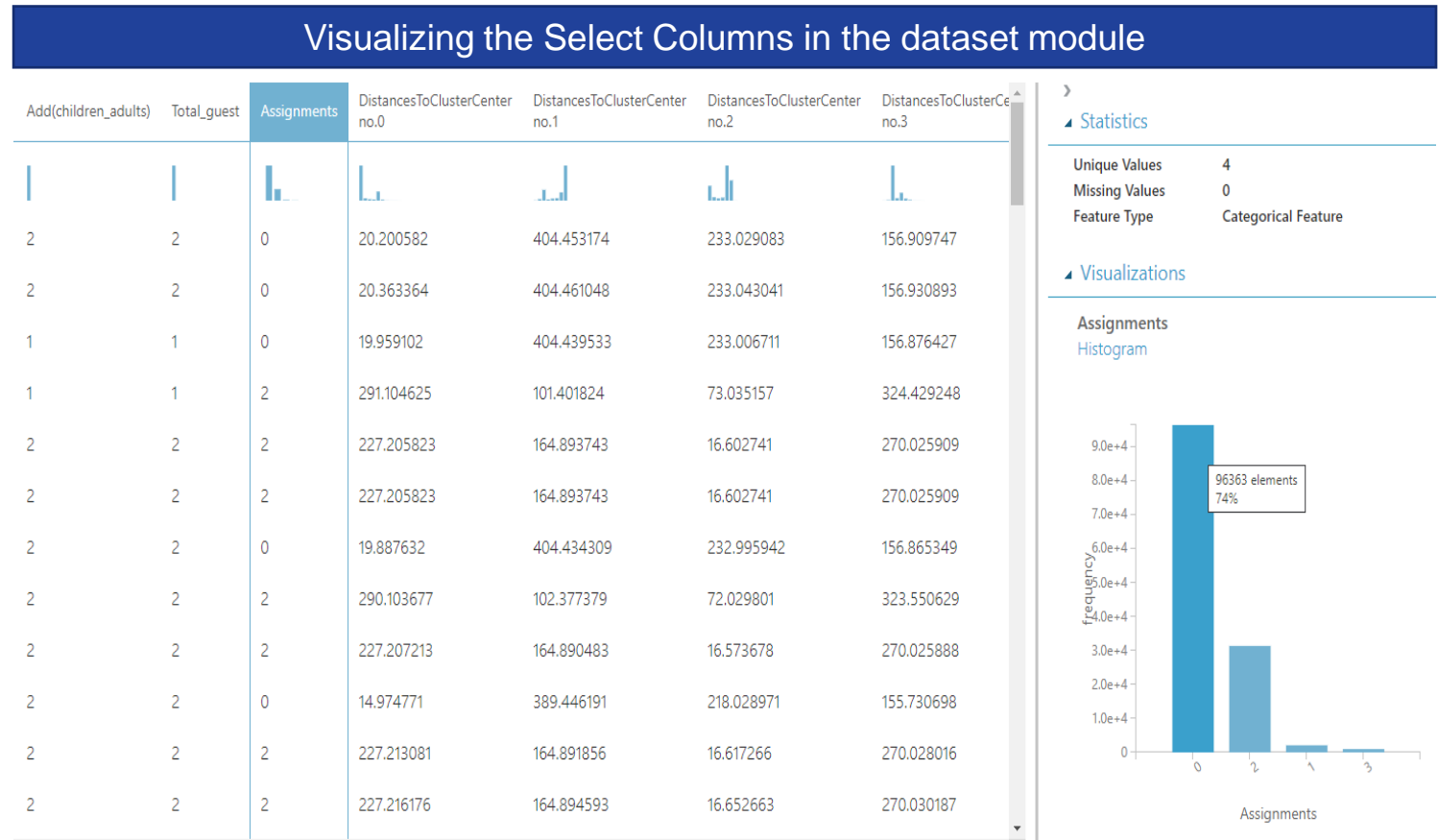
- Drag the **'Select Columns in Dataset'** module and connect it to **'Train Clustering Model'** module
- Select all the columns of the dataset using the **'Launch Column Selector'** of the **'Select Columns in Dataset'** module



Cluster Analysis in Azure ML

10. Let us try to understand the results section of the **'Select Columns in Dataset'** module by clicking on the **'Visualize'** tab

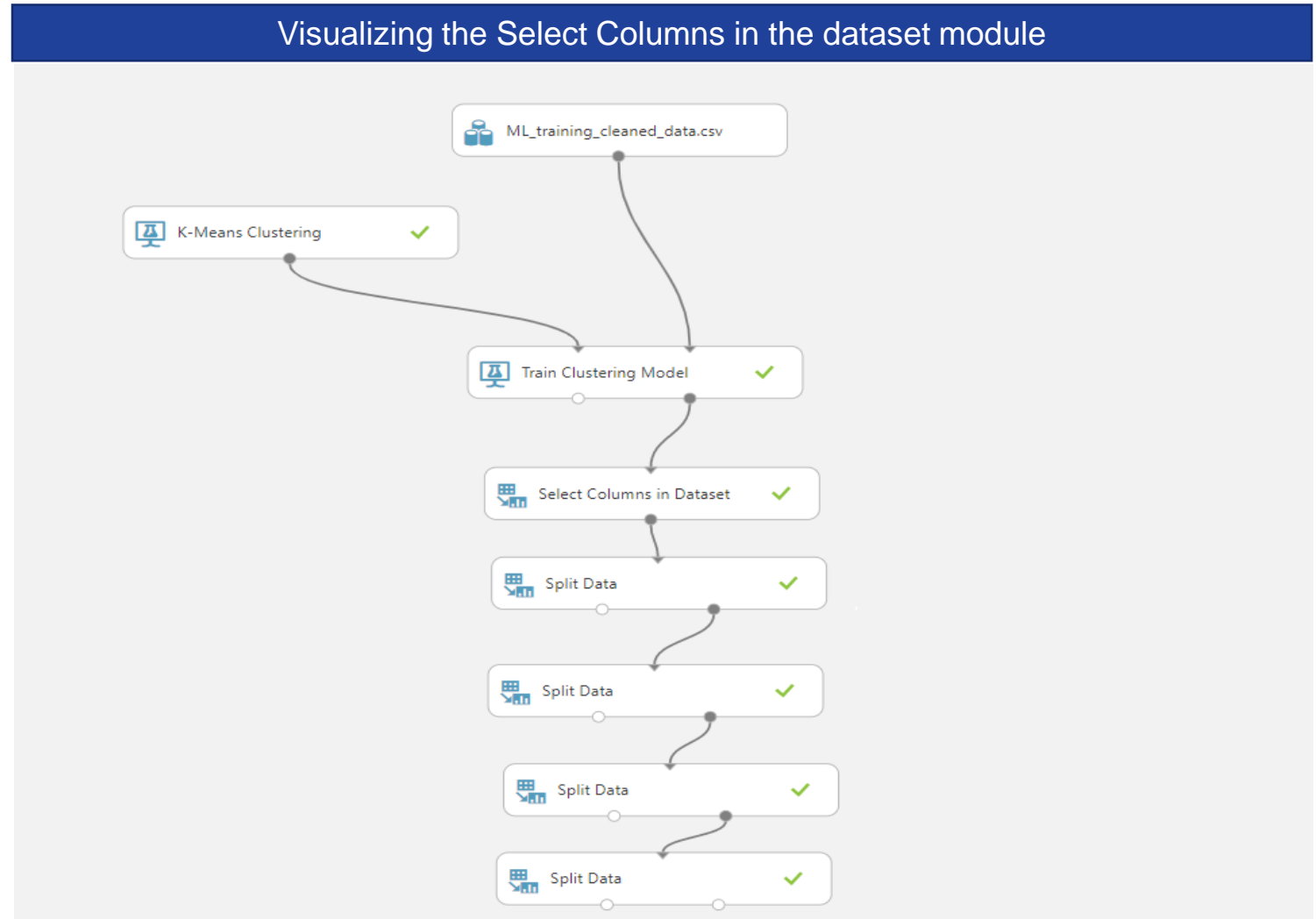
- The first 30 columns are columns in the data
- **'Assignments'** gives the cluster id to which the row is assigned. We can check the number of datapoints in a cluster by hovering on the histogram in the Assignments section
- **'DistancesToClusterCenter no.*'** gives the distance of each row to each cluster centroid



Cluster Analysis in Azure ML

11. Let us split the data into 4 data frames according to the cluster id

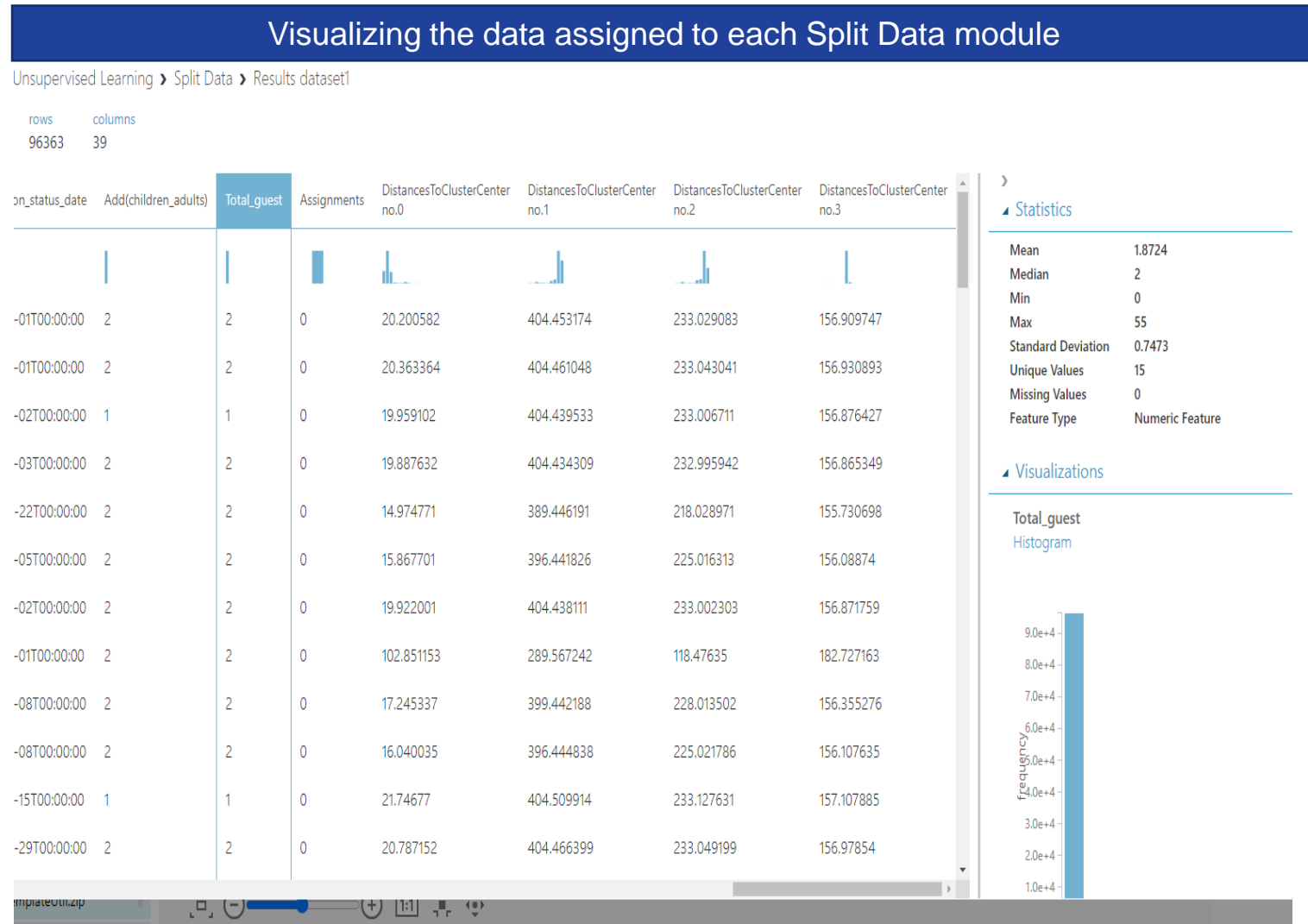
- Drag the '**Split Data**' module and connect it to the '**Select Columns in Dataset**' module
- From the properties tab of Split Data, select the 'Splitting mode' as Relative Expression
- Select the Relational expression as 'Assignments' < 1 to filter the data with cluster id 0
- Repeat above steps to filter the data for cluster ids 1 to 3 by adding the Split Data tabs and setting the Relation expression as 'Assignments' < 2, 3, 4 respectively



Cluster Analysis in Azure ML

12. Let us verify the results of data assigned to each split(data filtered per each cluster).

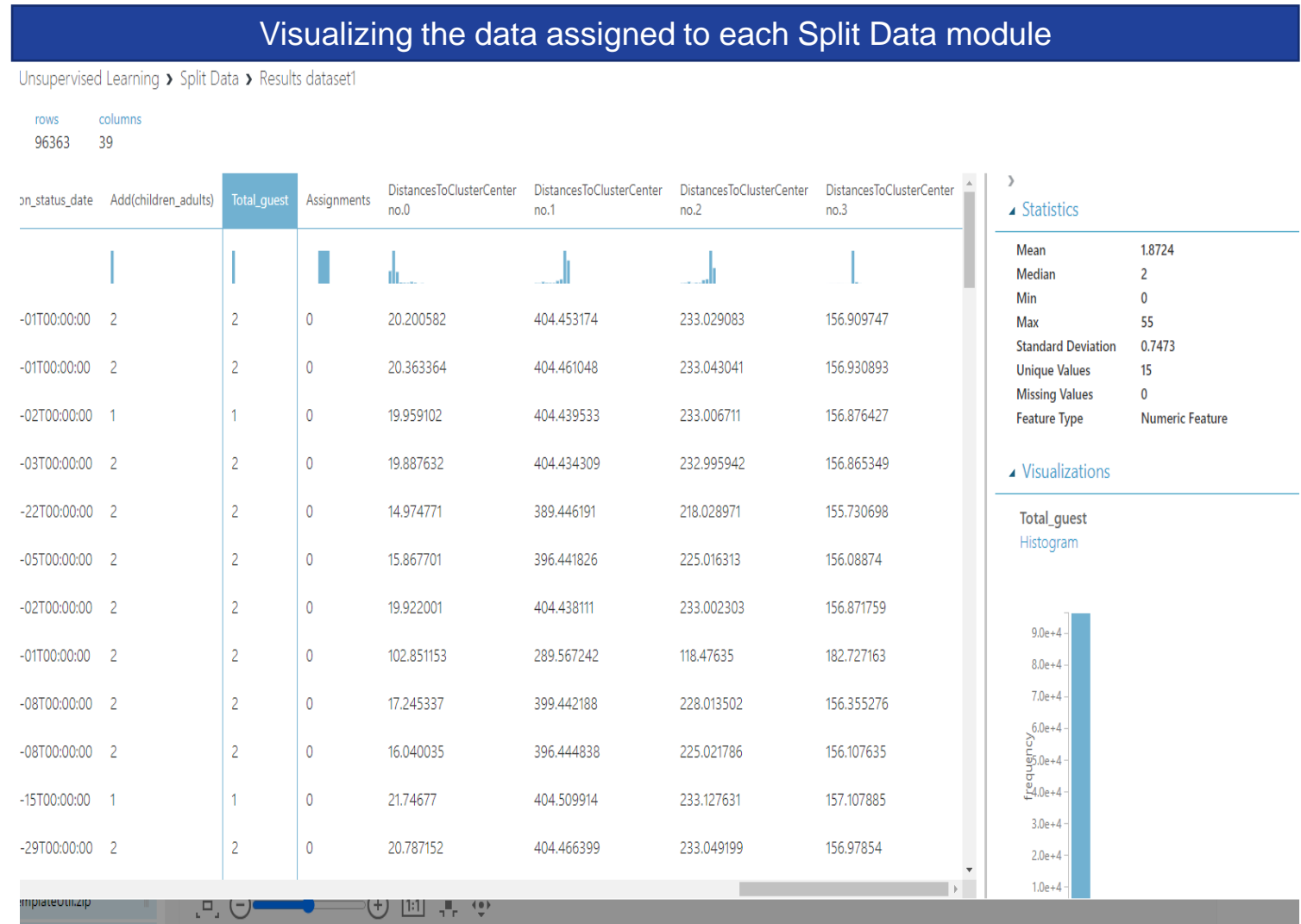
- Run the Visualize tab for each split
- We can check the cluster id assigned to each row
- The summary statistics for each variable can be verified in the 'Statistics' tab
- The distribution of the variable can be visualized in the 'Visualizations' tab



Cluster Analysis in Azure ML

13. Let us verify the results of data assigned to each split(data filtered per each cluster).

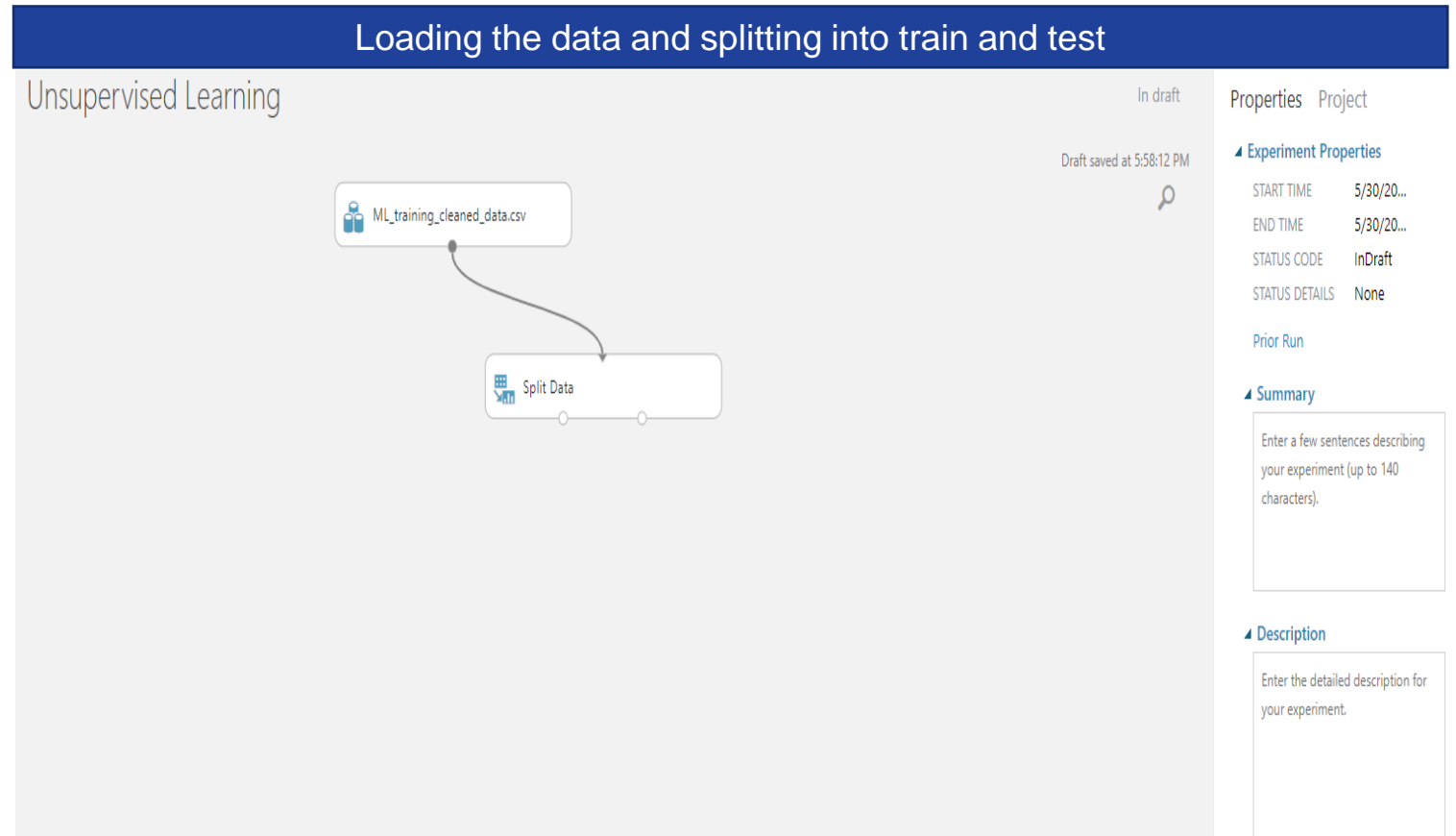
- Run the Visualize tab for each split
- We can check the cluster id assigned to each row
- The summary statistics for each variable can be verified in the 'Statistics' tab
- The distribution of the variable can be visualized in the 'Visualizations' tab
- The same process can be followed to understand the patterns of the variables in each cluster and understand the differences between them



Cluster Analysis in Azure ML

Now, let us have a look at how we can train a clustering model, then predict the unseen data using the existing model and evaluate the model.

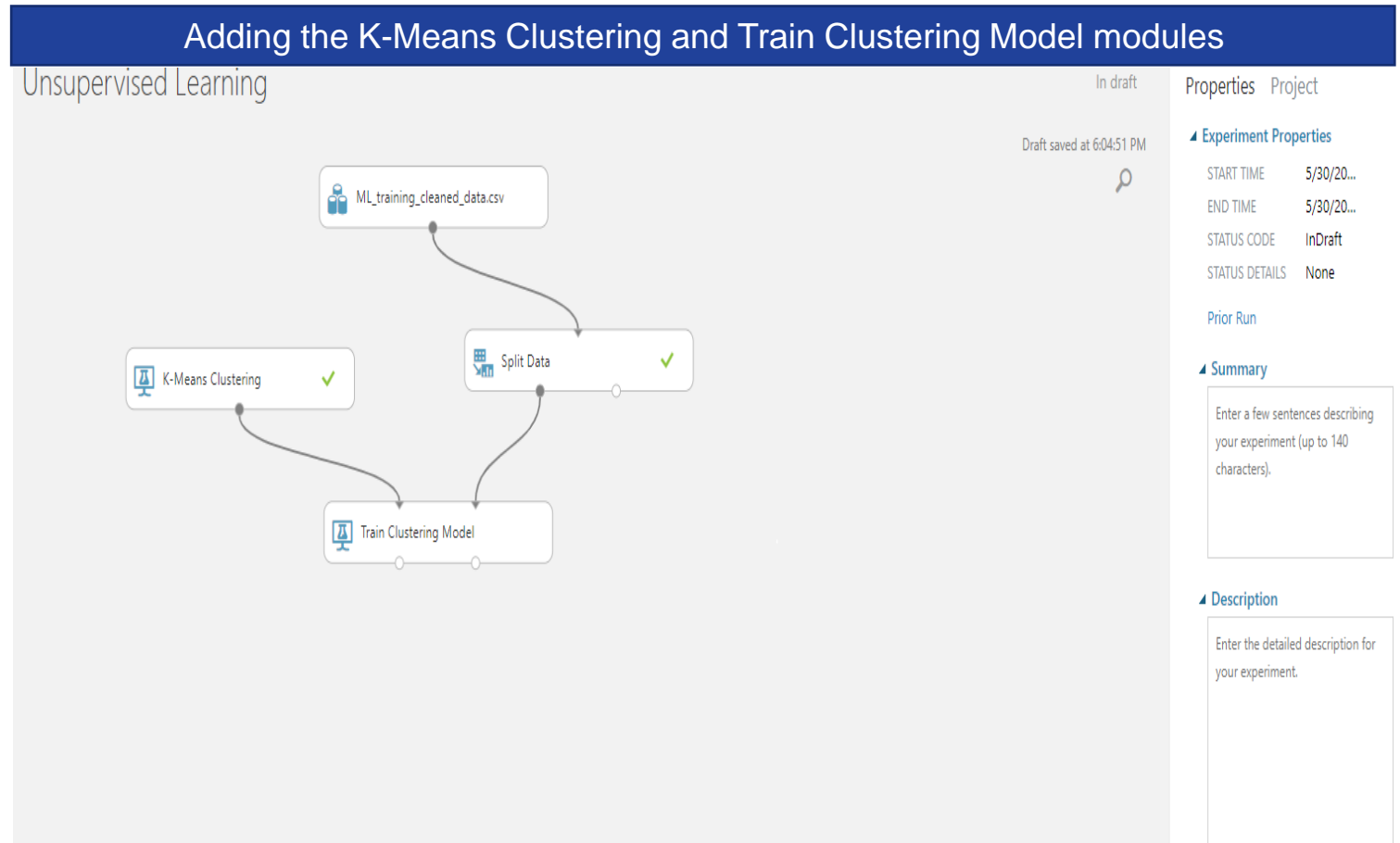
- Load the data and split it into train test in the ratio of 70:30 using the 'Split Data' module
- In the Split Data properties, select the below parameters.
 - Splitting mode – Split Rows
 - Fraction of rows in the first output dataset – 0.7
 - Random seed – 123
 - Stratified Split - False



[Demo Link: K-Means Clustering Demo - Cluster Assignment | Azure AI Gallery](#)

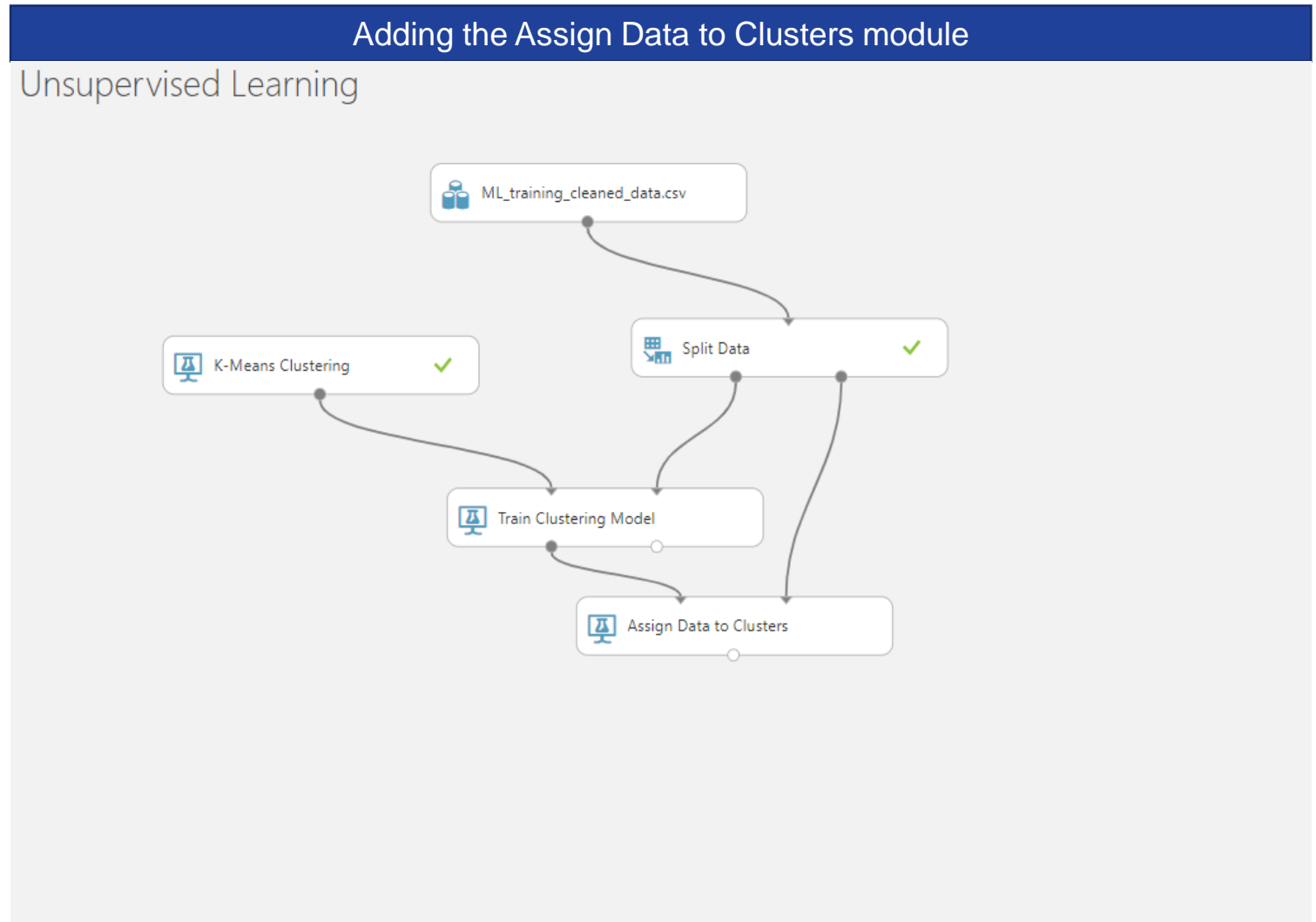
Cluster Analysis in Azure ML

Add the K-Means Clustering and Train Clustering Model modules



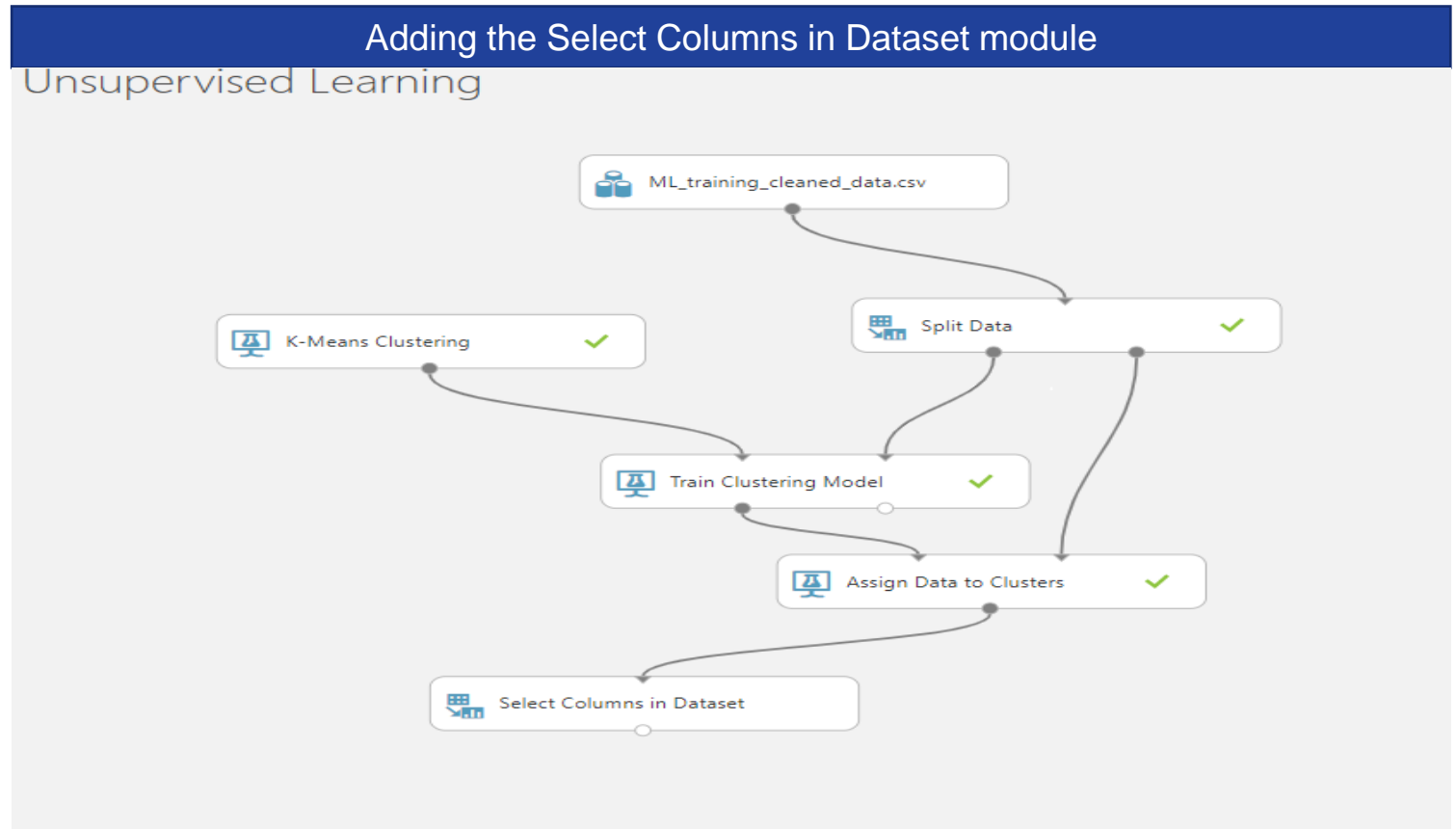
Cluster Analysis in Azure ML

- Add 'Assign Data to Clusters' module
- Connect the Train data from the 'Split Data' to 'Train Clustering Model' module and Test Data to 'Assign Data to Clusters' module
- Click 'RUN'



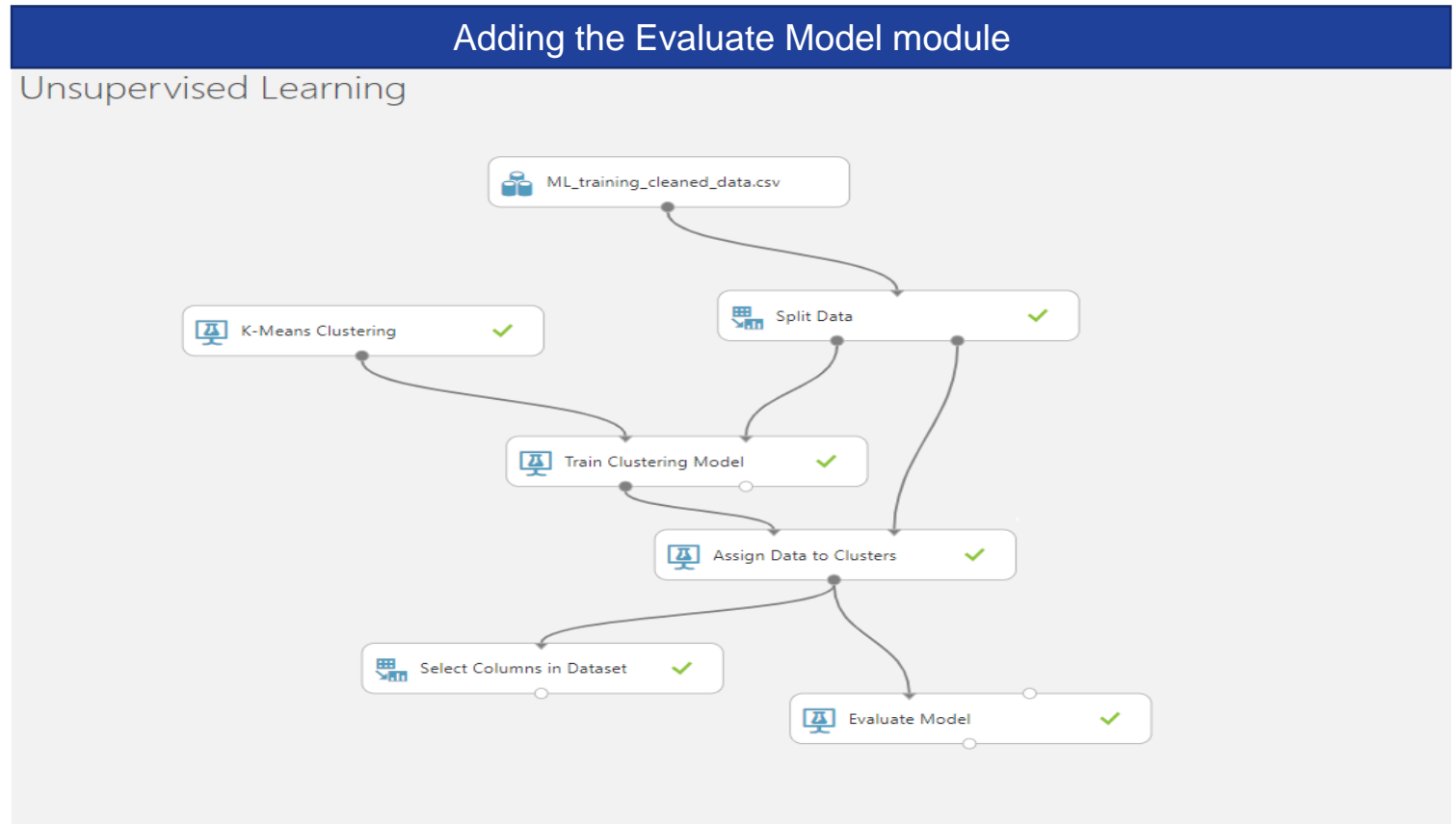
Cluster Analysis in Azure ML

- Add 'Select Columns in Dataset' module and connect it to 'Assign Data to Clusters' module to visualize the results



Cluster Analysis in Azure ML

- Add the 'Evaluate Model' module and connect it with 'Assign Data to Clusters' module and RUN



Cluster Analysis in Azure ML

- Visualize the results of the 'Evaluate Model' module
- This module gives us the information on how many data points were assigned to each cluster? Amount of separation between clusters and how tightly the data is clustered?
- Combined evaluations contain lists the average scores for the clusters created for this model. This comparison helps us to compare different models
- **Average Distance to Cluster Center** - represents how close the data points within the cluster are
- **Average Distance to Cluster Center** - How close all the data points in a cluster to the centroids of the other cluster
- **Number of points** - num of points assigned to each cluster
- **Maximal Distance To Cluster Center** - the sum of distances between each point and the centroid of that point's cluster, how widely spread the clusters are








Verifying the results of Evaluate Model module

rows

5

columns

5

	Result Description	Average Distance to Cluster Center	Average Distance to Other Center	Number of Points	Maximal Distance To Cluster Center
<div>view as</div> <div>   </div>					
	Combined Evaluation	52.228527	207.191112	39246	5215.512214
	Evaluation For Cluster No.0	47.489819	231.232016	27440	390.516723
	Evaluation For Cluster No.1	101.73654	272.255215	1904	470.079257
	Evaluation For Cluster No.2	57.824912	129.169846	7417	297.570904
	Evaluation For Cluster No.3	49.918076	124.744141	2485	5215.512214

[Demo Link: K-Means Clustering Demo - Cluster Assignment | Azure AI Gallery](#)

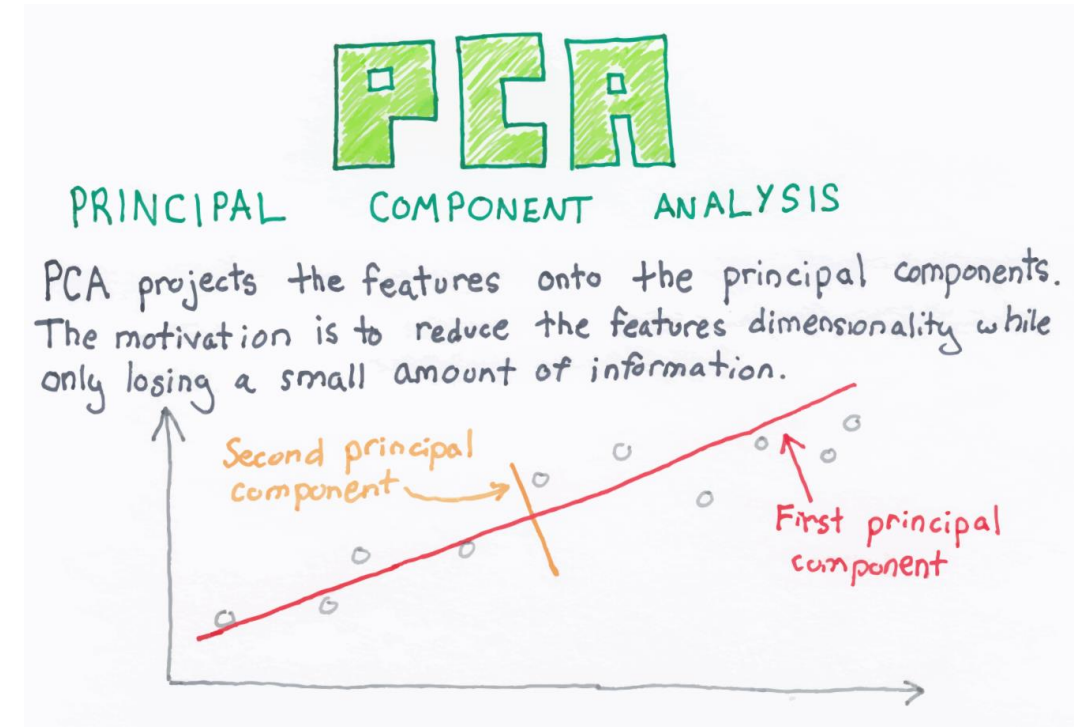
Principal Component Analysis

Curse of Dimensionality

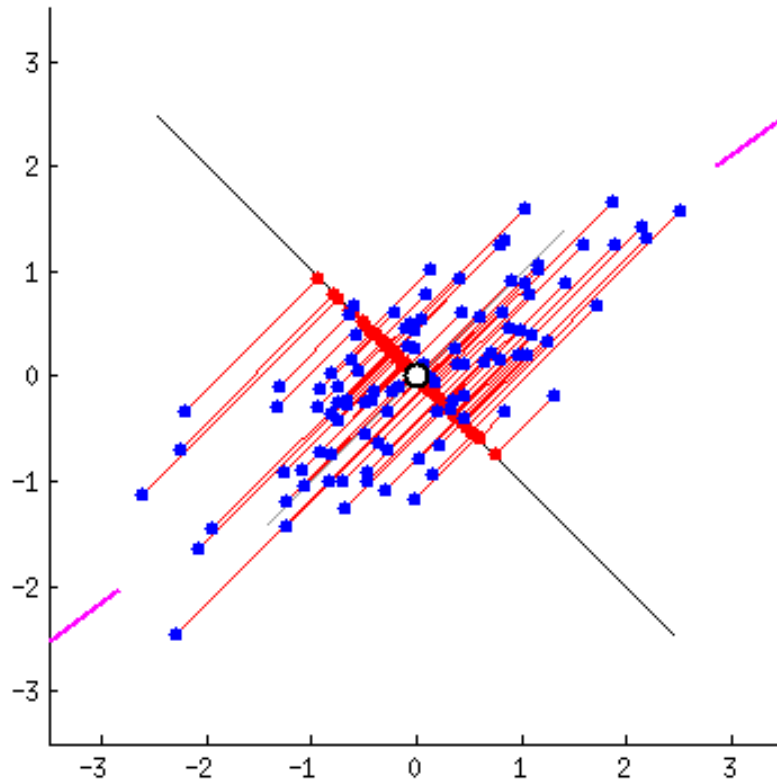
- As the number of features or dimensions grows, the amount of data we need to generalize grows exponentially
- Suppose we have n observations and d variables in our dataset and we wish to study the relationship between different variables as part of EDA.
- For a larger value of d , let's say 60, we get $d(d-1)/2$ two-dimensional scatter plots.
- Such a huge number of plots (1770, in this case) makes it certainly difficult to identify the relationship between features.
- Further, these 2D plots contain only a fraction of the total information present in the dataset
- In the next slides, we will discuss how Principal Component Analysis helps us to achieve dimensionality reduction

Principal Component Analysis

- PCA aims to reduce the dimensionality by projecting the data to a lower dimensional subspaces (Nth principal) which capture the “essence” of the data.
- Started with finding the first linear combination of input variables, which maximizes the variance of the data. This is called as first principal component and variance explained by this component has to be maximum by design.
- Then find the other linear combination, which maximizes the variance that has not been explained by the first combination (i.e residual variance). This component is called as second principal component.
- And so on for third principal component etc.
- PCA gives us the artificial variables through combination of input variables
- By nature, sum of variance of all PCs = total variance = variance of individual variables
- $\text{Var of PC1} > \text{Var of PC 2} > \dots$
- If we choose the first K PC's, they explain the big portion of total variance with much lesser number of PC's – helps in dimensionality reduction



Visualizing PCA



- First step is to find the direction, where the data has maximum variance

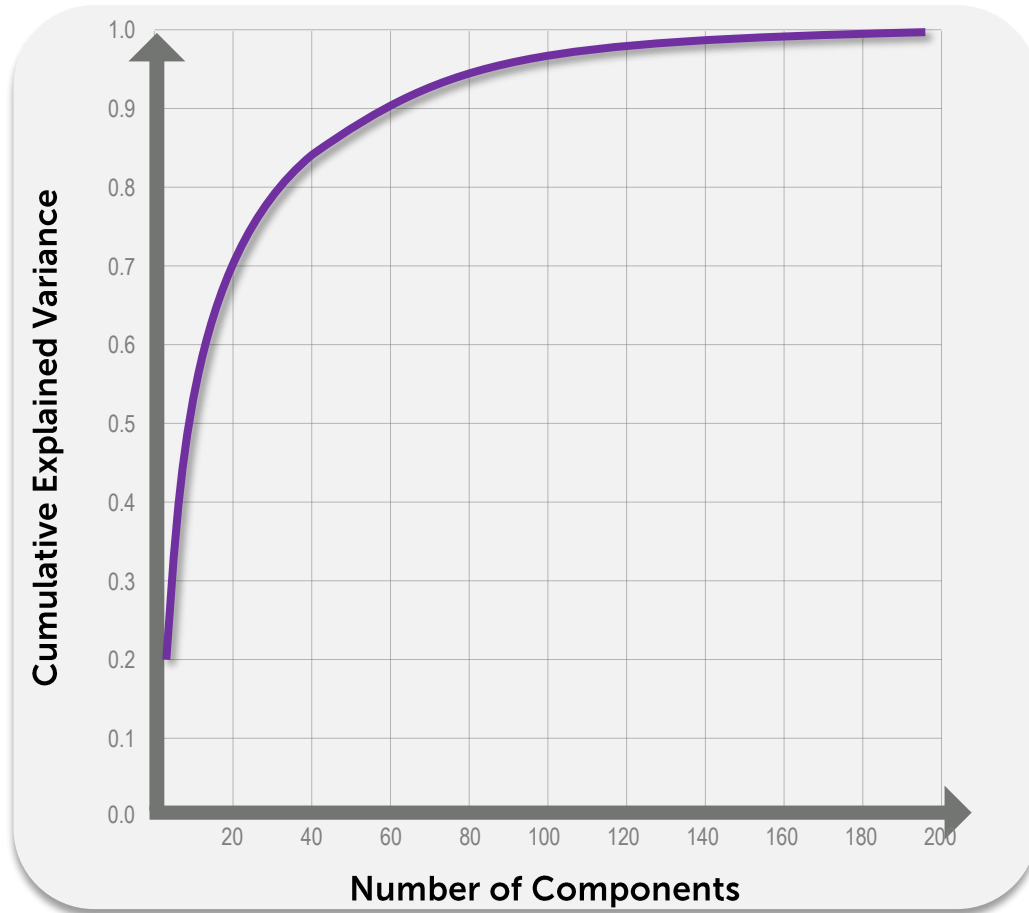
- This direction is called the direction of first principal component (PC 1)

- The variance of data in the direction of first principal component is also called
 - the variance of the first principal component or
 - the first eigen value

- Next, the direction perpendicular to the first principal component will have the residual variance

- This direction is second principal component and the variance of the data in this direction is called the variance of the second principal component or the second eigen value

Determining Number of Components

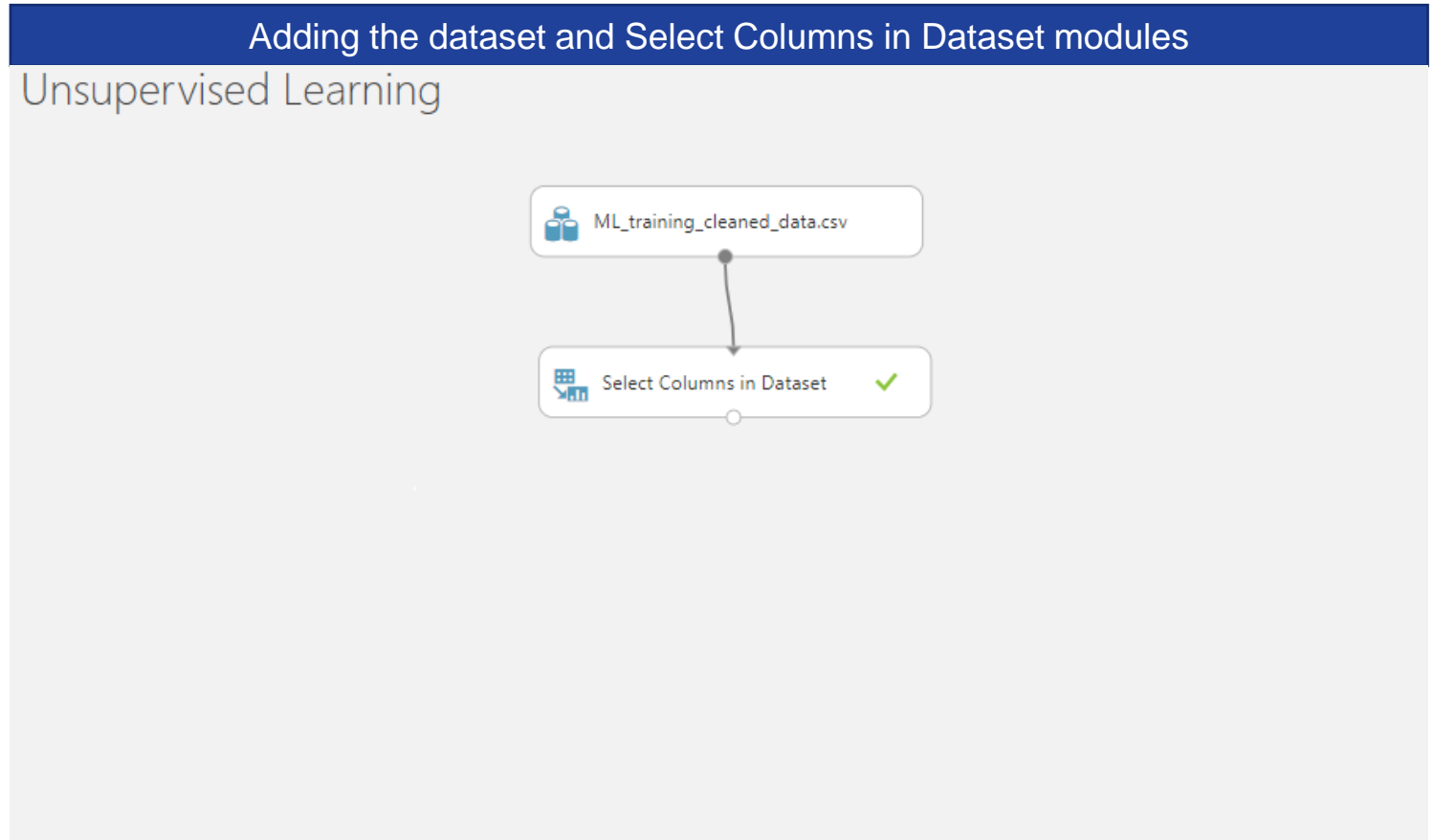


Cumulative Explained Variance

- A scree plot depicts this ratio explained by each of the principal components
- The elbows of the plot signify the optimal number of principal components
- The curve shown in Fig. quantifies how much of the total variance is contained within the first n components

Principal Component Analysis in Azure ML

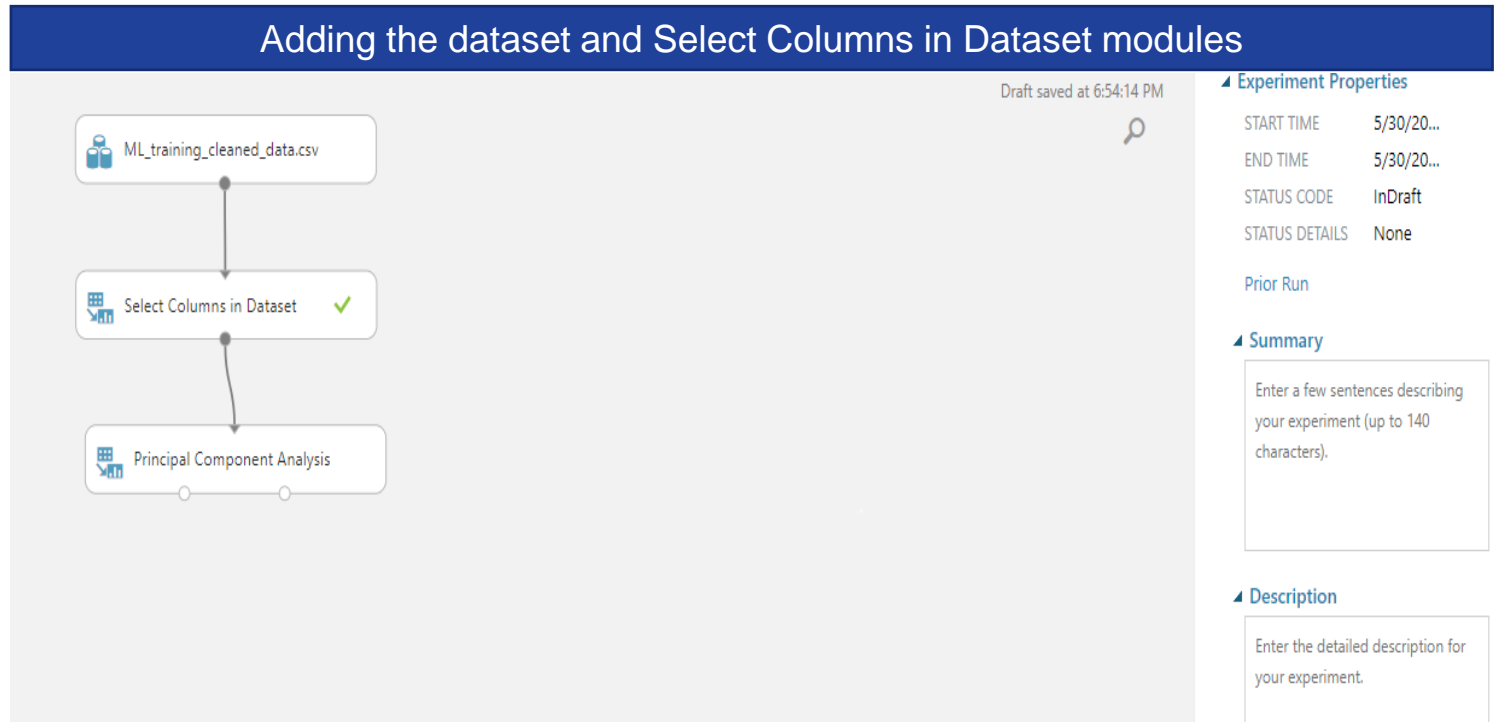
- Add the dataset module
- Add 'Select Columns in Dataset' module



<https://gallery.azure.ai/Experiment/PCA-Demo>

Principal Component Analysis in Azure ML

- Add the 'Principal Component Analysis' module
- Select the relevant columns in the 'Launch Column Selector'
- Enter **4** in the 'Number of dimensions to reduce' - The desired number of columns in the final output. Each column represents a dimension capturing some part of the information in the input columns
- Click RUN



Principal Component Analysis in Azure ML

- Right Click on the 'Principal Component Analysis' module and select 'Visualize'
- 'Col1', 'Col2', 'Col3', 'Col4' are the new principal components created which can be used as input features to train the Supervised Learning models

Adding the dataset and Select Columns in Dataset modules									
is_canceled	lead_time	customer_type	adr	reservation_status	reservation_status_date	Col1	Col2	Col3	Col4
false	320	Transient	0	Check-Out	2015-07-01T00:00:00	-1.605894	26.963735	1.00614	-0.39225
false	320	Transient	0	Check-Out	2015-07-01T00:00:00	-1.605548	26.964193	1.005291	-0.406918
false	7	Transient	75	Check-Out	2015-07-02T00:00:00	-1.610272	26.959868	1.016464	-0.417711
false	13	Transient	75	Check-Out	2015-07-02T00:00:00	-1.612326	26.959985	1.017373	-0.402294
false	14	Transient	98	Check-Out	2015-07-03T00:00:00	-1.608623	26.971038	1.022546	-1.083834
false	14	Transient	98	Check-Out	2015-07-03T00:00:00	-1.608623	26.971038	1.022546	-1.083834
false	0	Transient	107	Check-Out	2015-07-03T00:00:00	-1.608286	26.969312	1.021032	-1.118544
false	9	Transient	103	Check-Out	2015-07-03T00:00:00	-1.604745	26.971296	1.02098	-1.149962
true	85	Transient	82	Canceled	2015-05-06T00:00:00	-1.6093	26.974513	1.028719	-1.468983
true	75	Transient	105.5	Canceled	2015-04-22T00:00:00	-1.607547	26.973109	1.027151	-1.557222
true	23	Transient	123	Canceled	2015-06-23T00:00:00	-1.609104	26.97605	1.032847	-1.896853
false	35	Transient	145	Check-Out	2015-07-05T00:00:00	-1.608224	26.976585	1.033324	-1.942371
false	68	Transient	97	Check-Out	2015-07-05T00:00:00	-1.597724	26.981202	1.031682	-1.942569
false	18	Transient	154.77	Check-Out	2015-07-05T00:00:00	-1.604588	26.983897	1.0311	-2.168866
false	37	Transient	94.71	Check-Out	2015-07-05T00:00:00	-1.609104	26.97605	1.032847	-1.896853
false	68	Transient	97	Check-Out	2015-07-05T00:00:00	-1.597724	26.981202	1.031682	-1.942569
false	37	Contract	97.5	Check-Out	2015-07-05T00:00:00	-1.609104	26.97605	1.032847	-1.896853
false	12	Transient	88.2	Check-Out	2015-07-02T00:00:00	-1.609176	26.965645	1.015562	-0.705538

Summary

Summary

1

- Clustering and Principal Component Analysis are the most commonly used techniques in unsupervised learning.

2

- The purpose of conducting clustering is to understand the behavioral differences amongst the different cluster segments i.e. find out the insights which demarcates cluster segments and assist businesses to target and cater the needs of respective segments in a better way.

3

- Principal Component Analysis helps us to achieve dimensionality reduction.

4

- The goal of PCA is to find the first linear combination of input variables, which maximizes the variance of the data. This is called as first principal component and variance explained by this component has to be maximum by design.

Thank you for your passion!

