St. Francis Institute of Technology, Mumbai-400 103

**Department Of Information Technology**

A.Y. 2024-2025

Class: TE-ITA/B, Semester: VI

Subject: **Business Intelligence Lab**

**Experiment – 7: To implement K-means clustering algorithm using open source tools, WEKA & ORANGE**

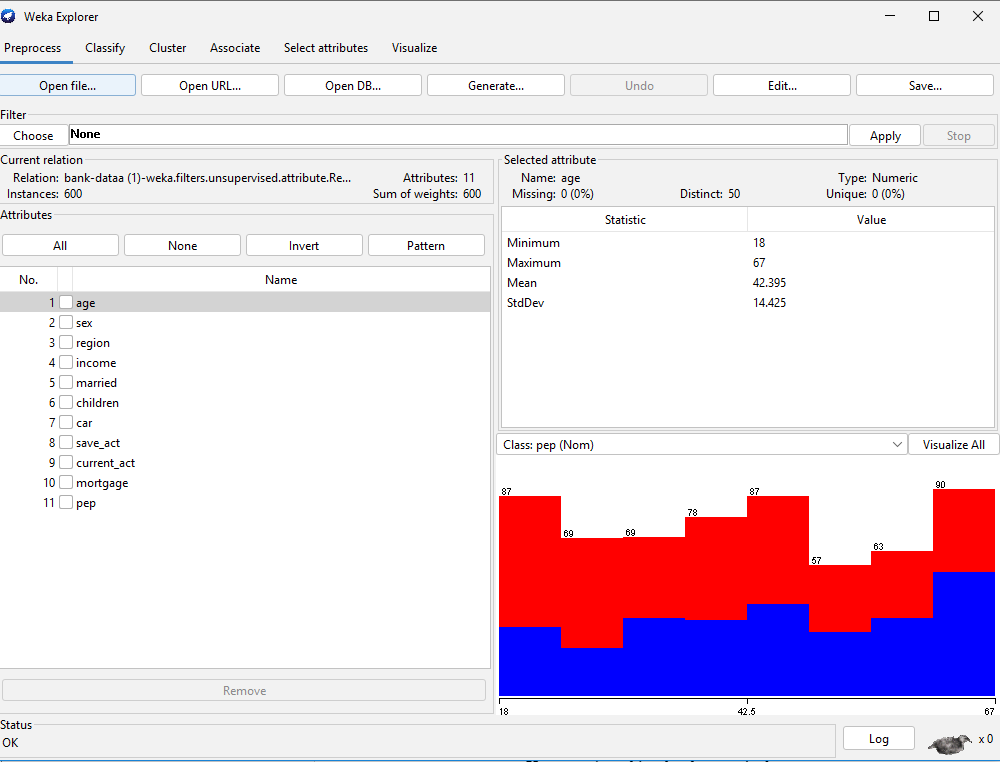
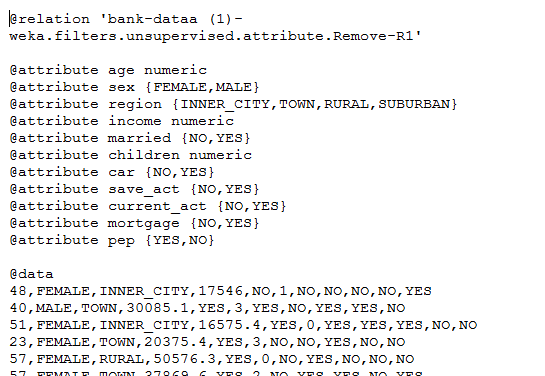
1. **Aim: :** To implement K-means clustering algorithm using open source tools, WEKA & ORANGE
2. **Objectives:** After study of this experiment, the students will be able to

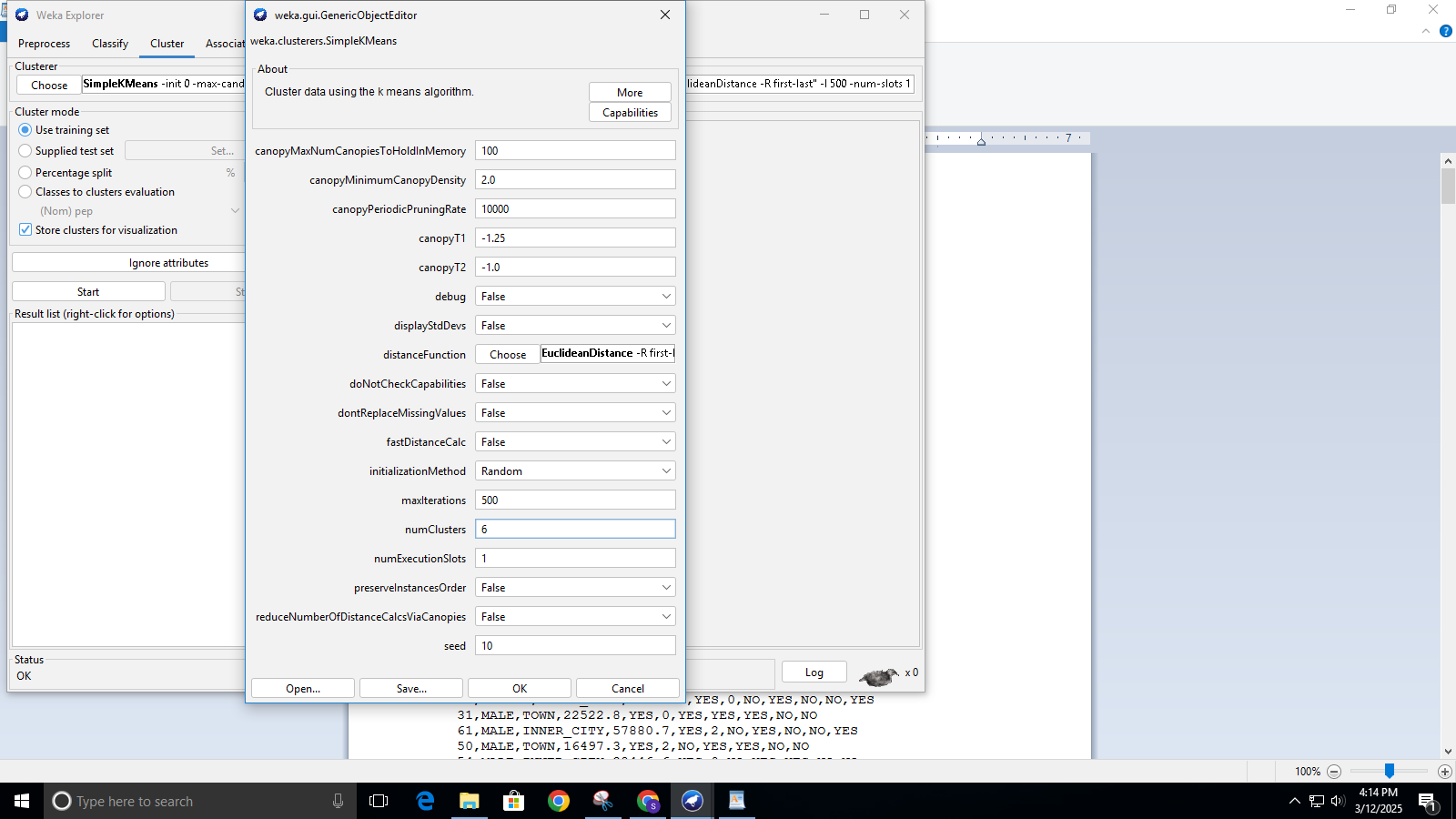
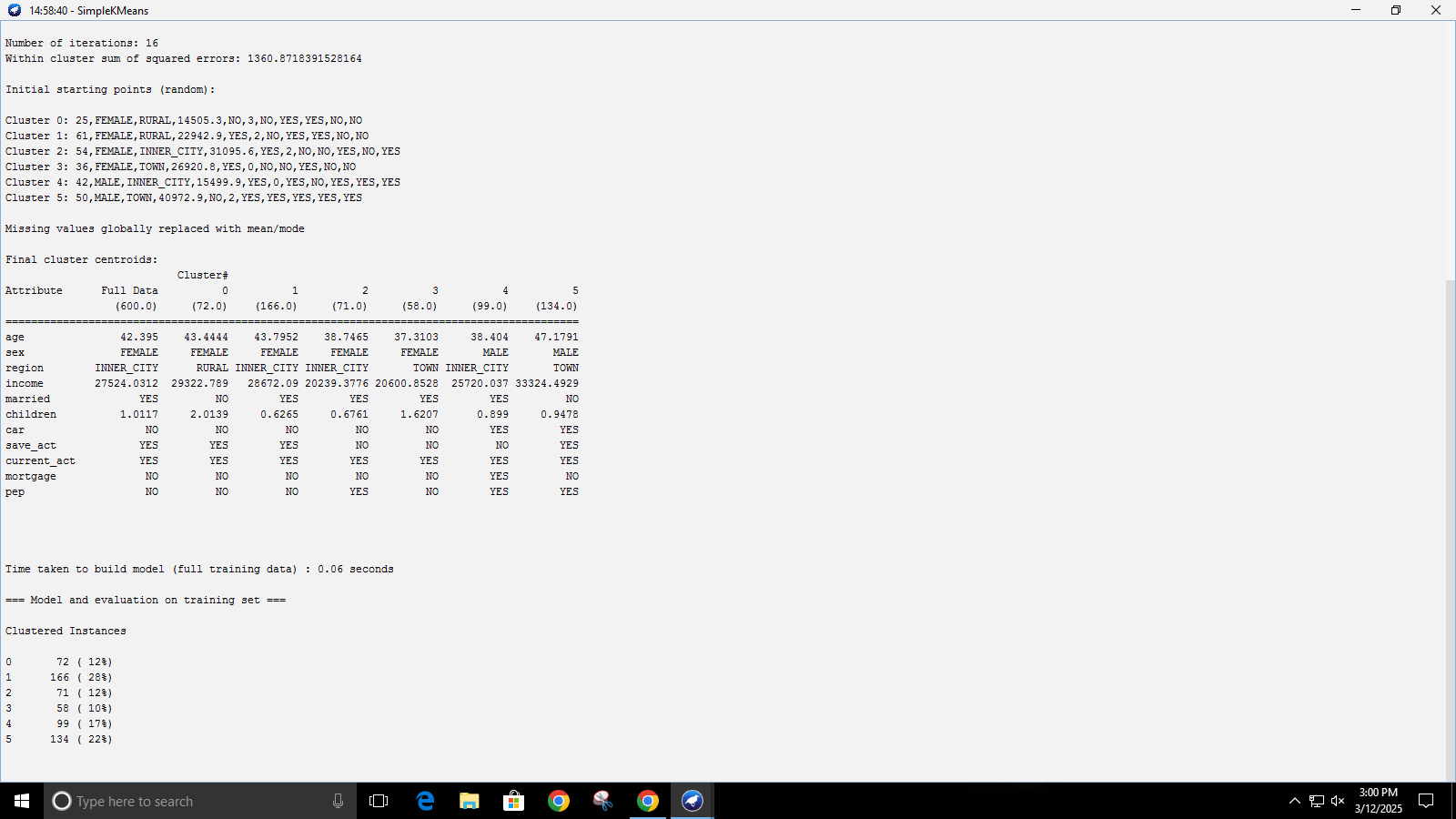
                              Implement K Means

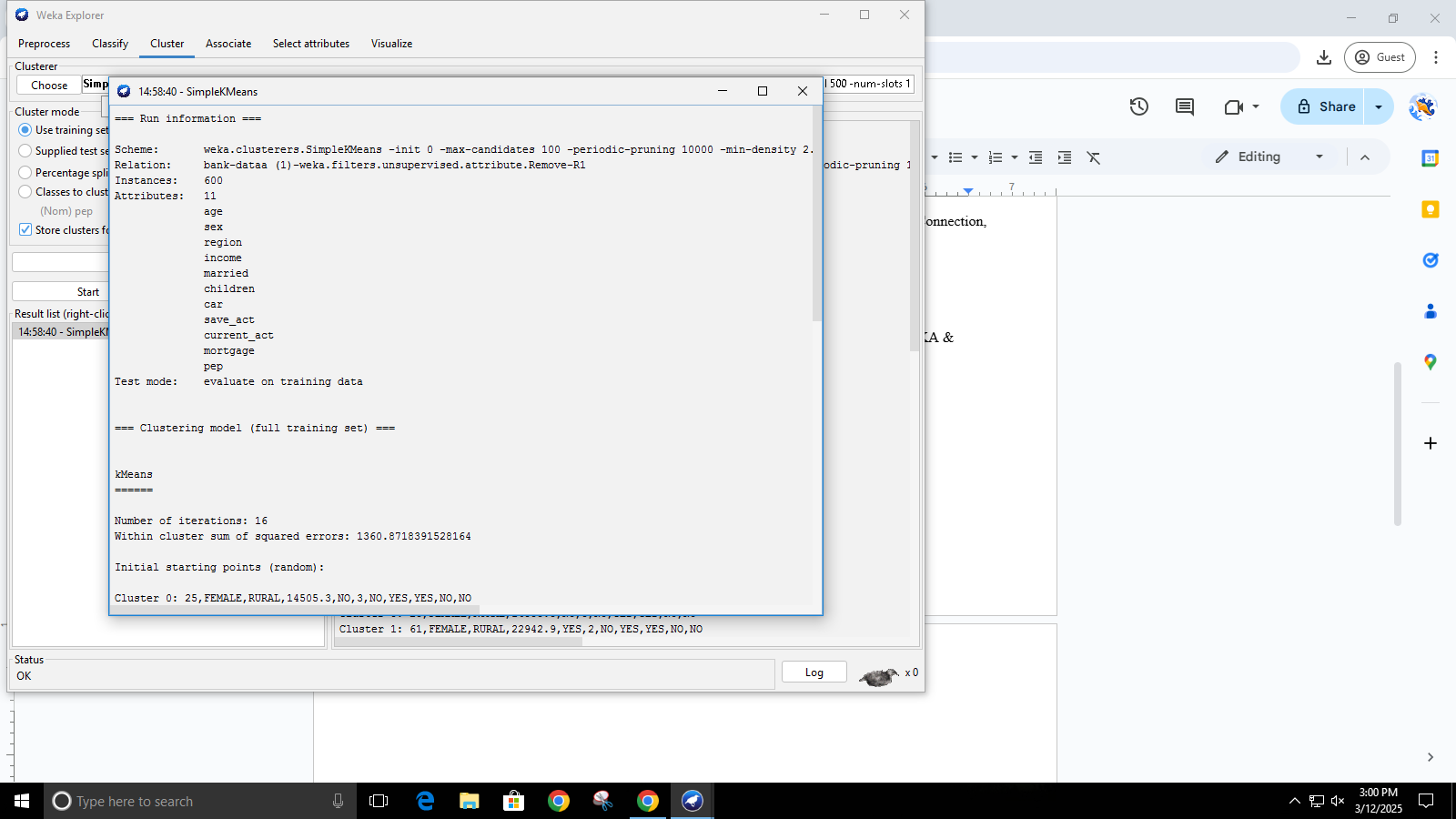
1. **Outcomes:** After study of this experiment, the students will be able to

**CO 4:** Design andImplement various clustering data mining techniques such as Partitioning methods, Hierarchical Methods, Density - Based methods along with identification and analysisof outlier.

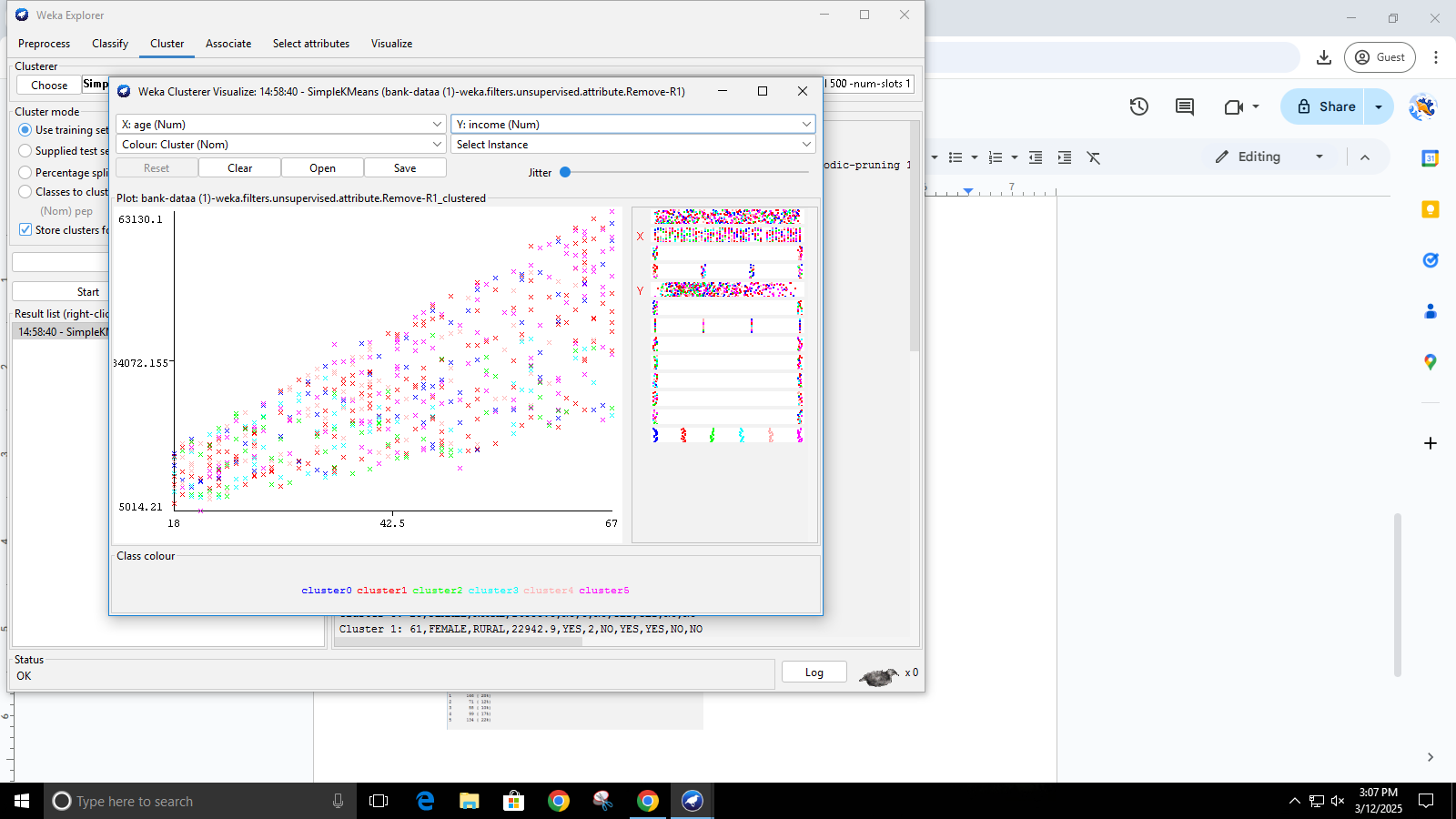
1. **Prerequisite:** Introduction to all the three clustering algorithms & Problem solving approach.
2. **Requirements:** Personal Computer, Windows XP operating system/Windows 7, Internet Connection, Microsoft Word, WEKA tool, ORANGE tool.
3. **Theory:**
4. Explain K means (graph) algorithm
5. Explain K medoids algorithm
6. **Laboratory Exercise:** Implementation of K means clustering algorithm using WEKA & ORANGE along with screenshots.
7. **Post-Experiments Exercise**
8. **Questions:**
   * Explain advantages and disadvantages of K means
   * K means (graph) solved numerical
9. **Conclusion:**
   * Summary of Experiment
   * Importance of Experiment
   * Application of Experiment

Loading the preprocessed file in weka  
  
  
Opened the preprocessed file with wordpad  


Applying k means with no. of clusters 6  
  
  
Output of K-means clustering

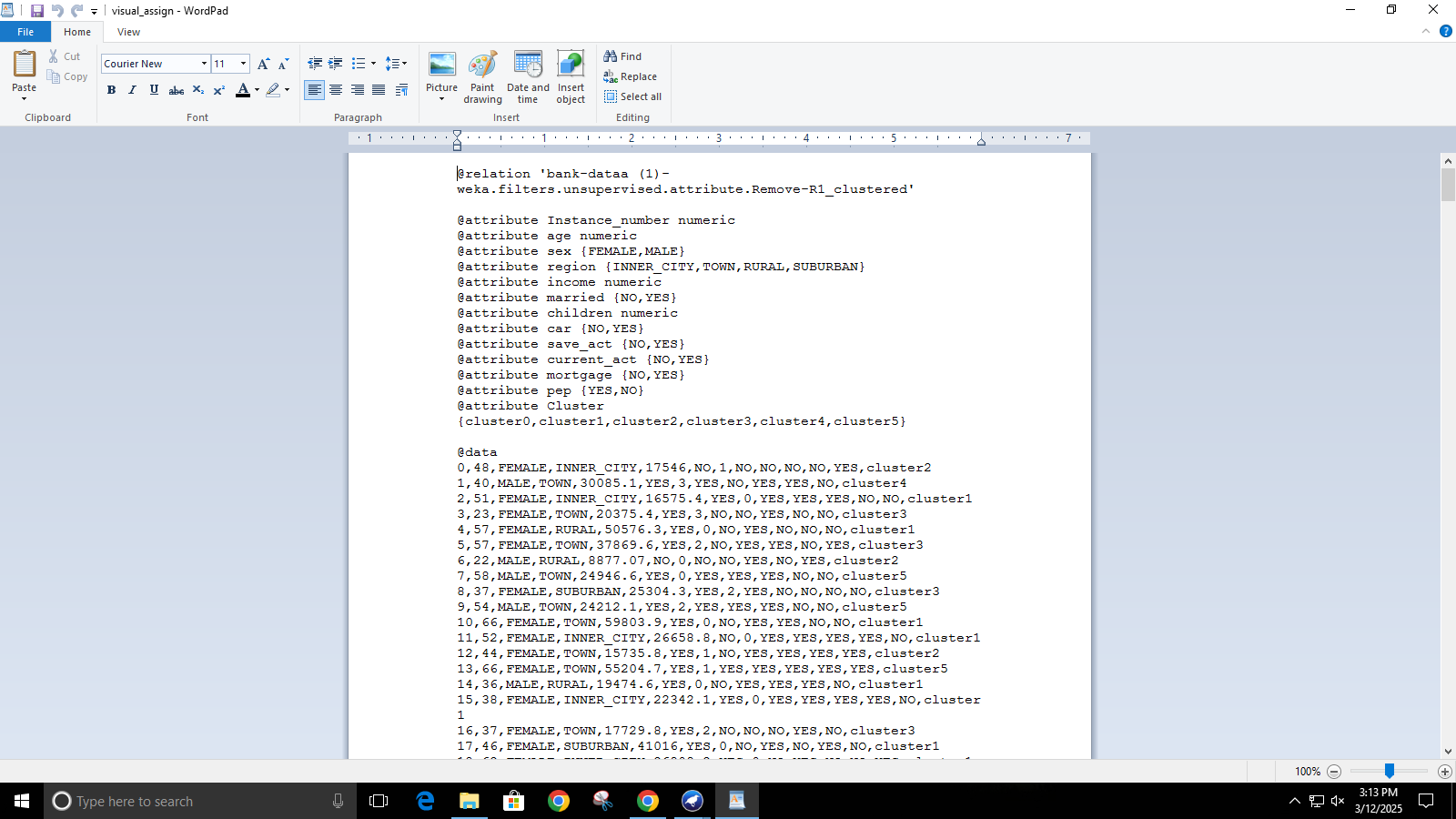


Visualizing cluster assignment

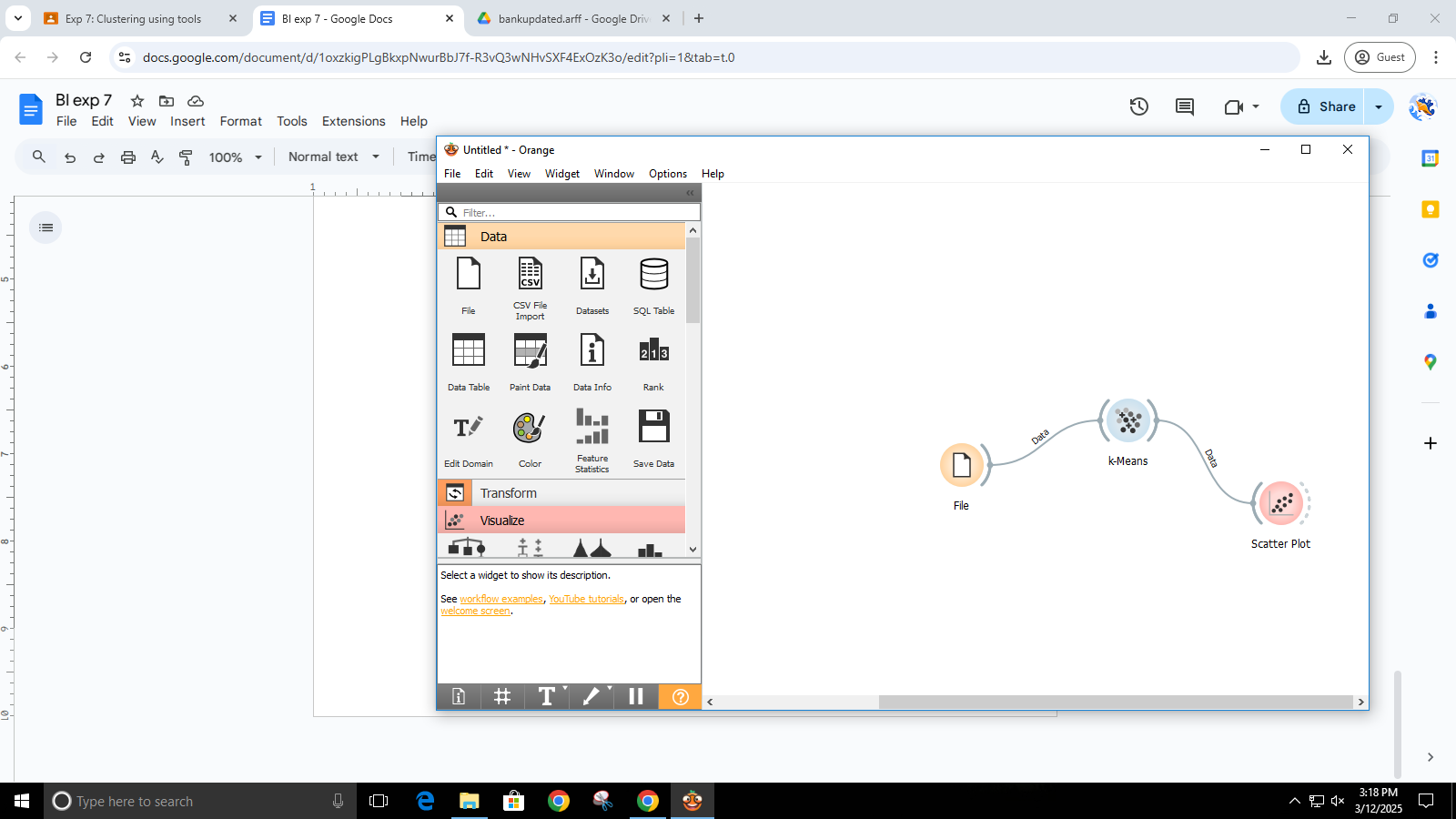


On trying to visualize Age vs Income this pattern of cluster is found fro which we can say that with increasing age there is increase in the income.

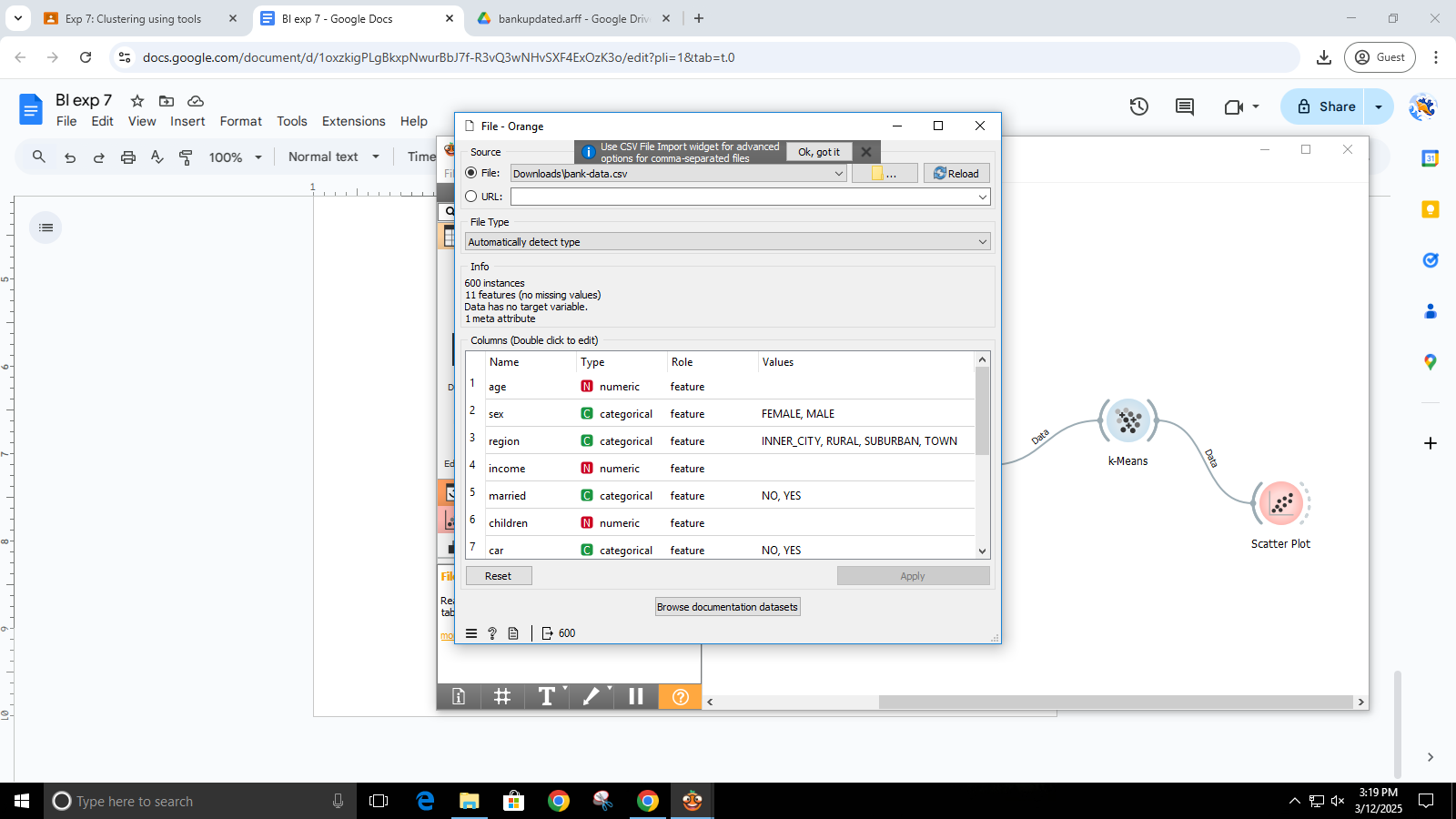
resultant dataset with clustering information.

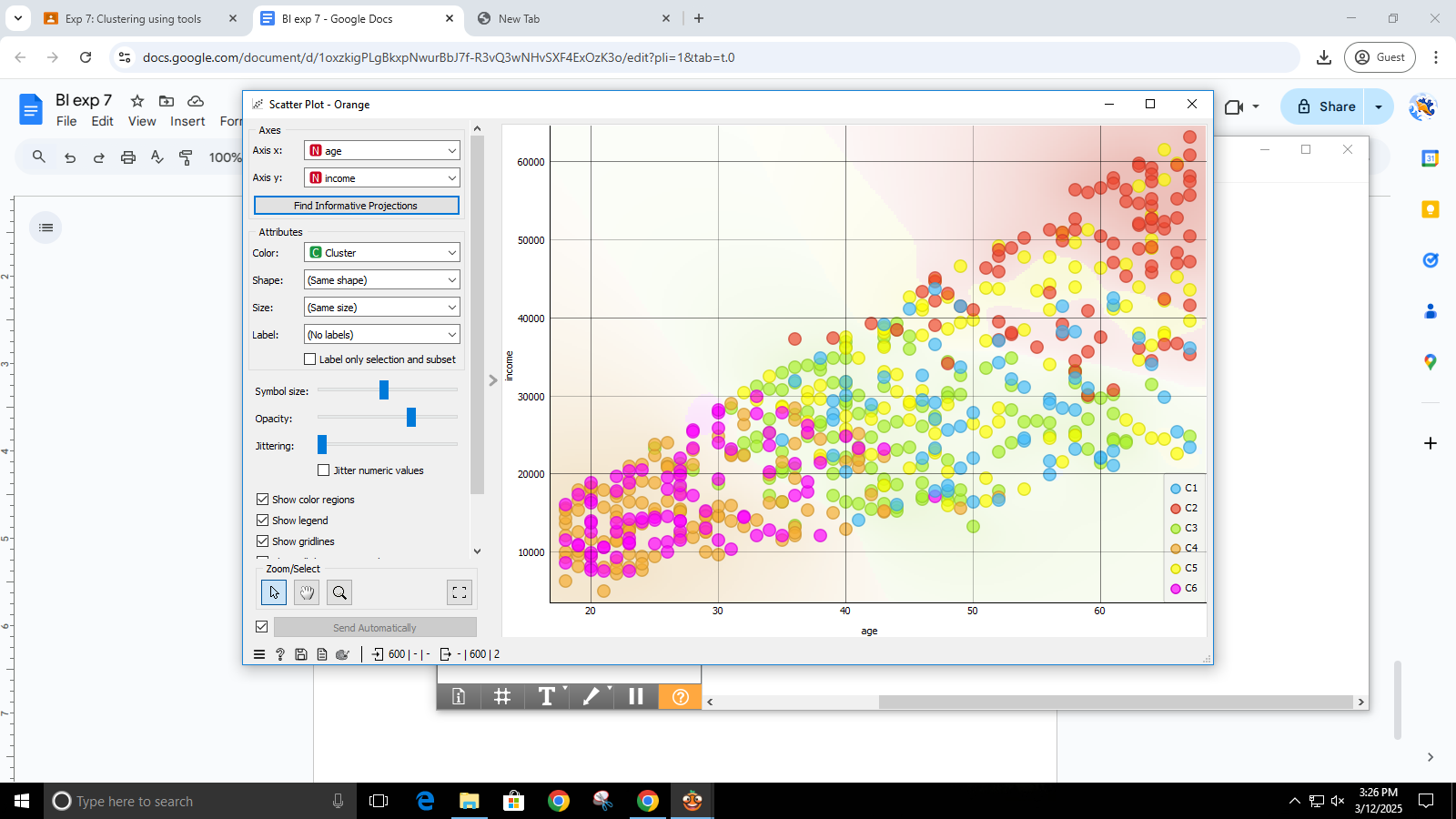


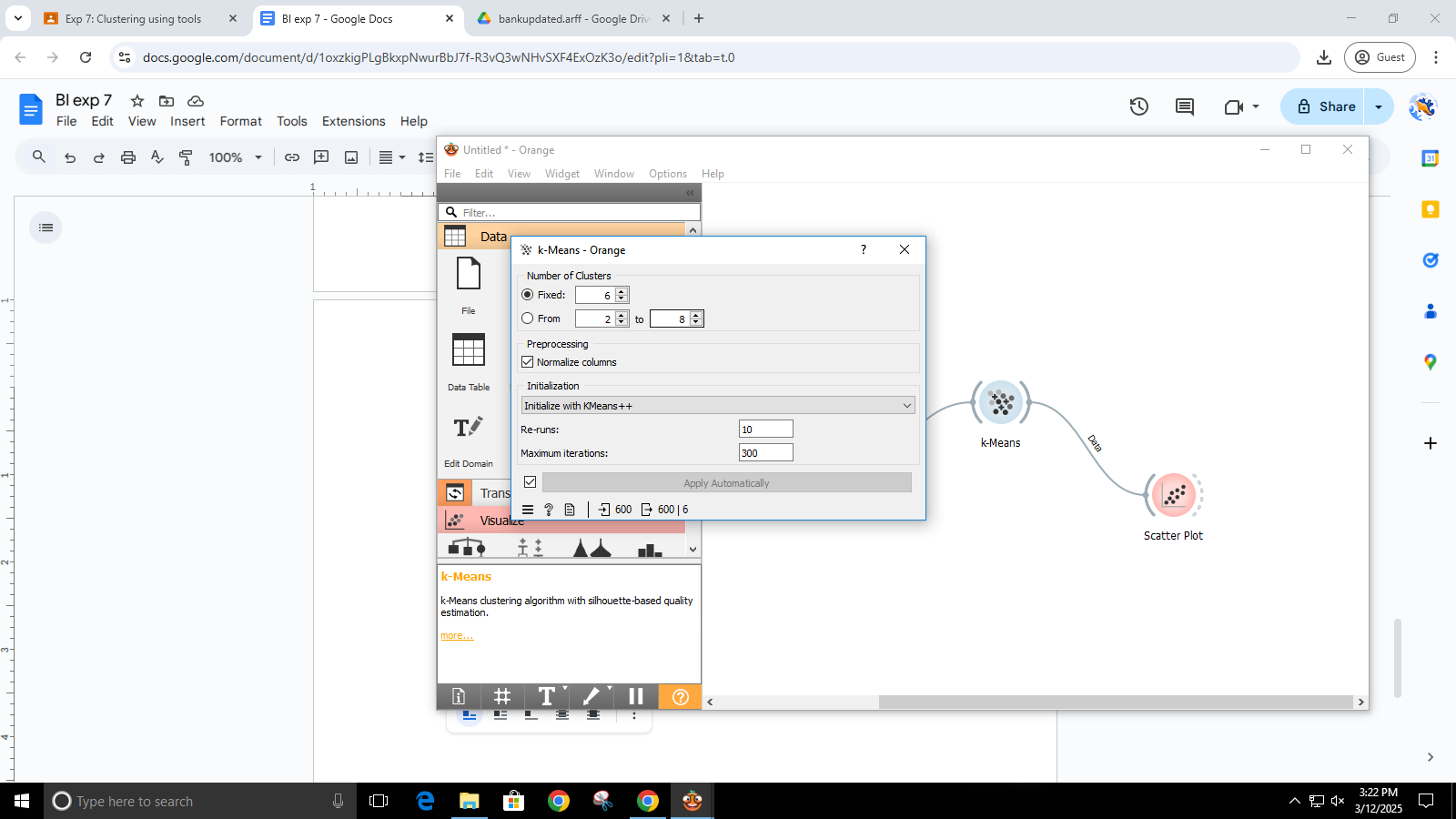
Using K-means in Orange



Loading dataset in Orange

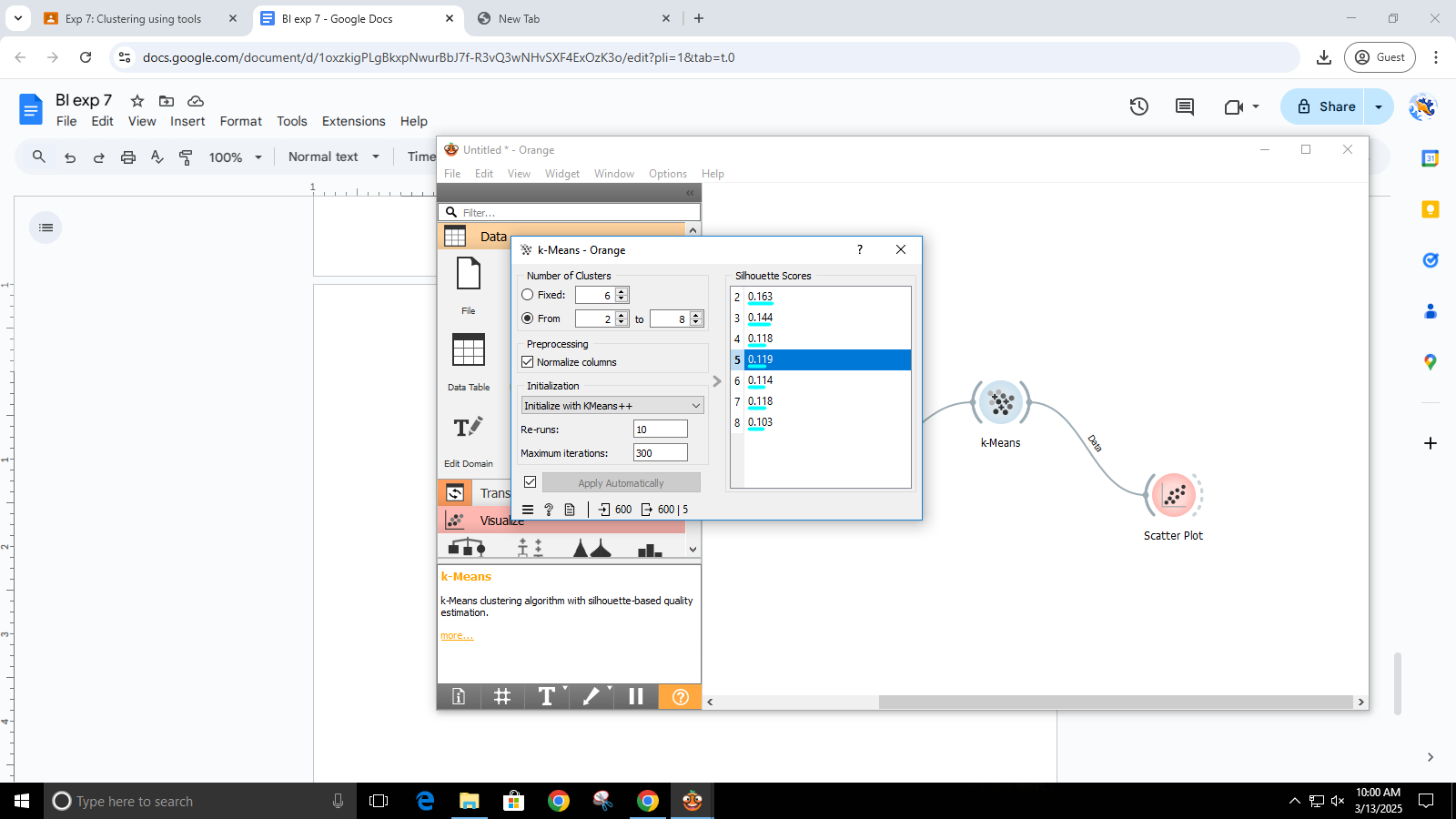


Setting number of cluster as 6 in Orange

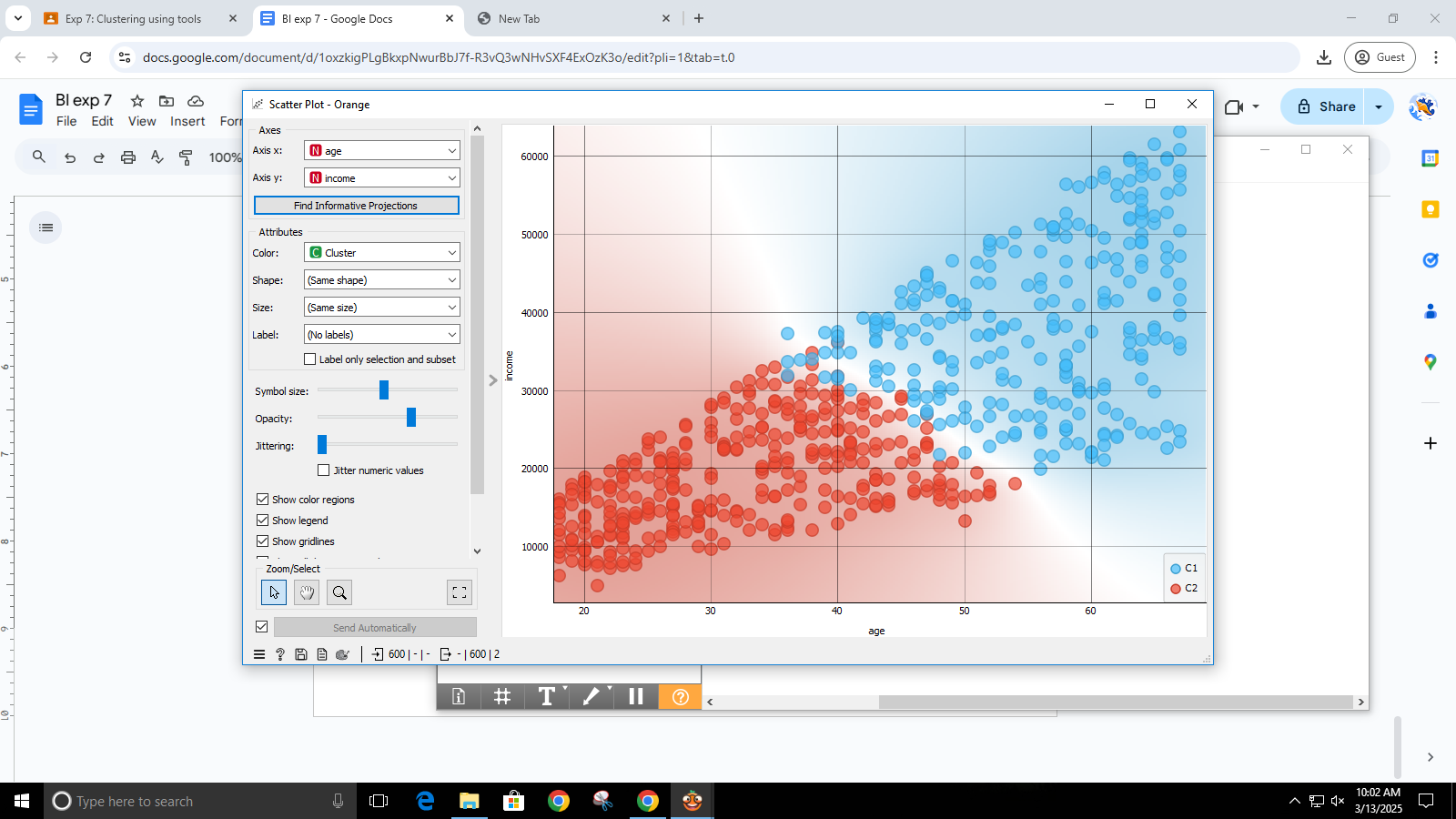


Result of k-means visible in scatter plot

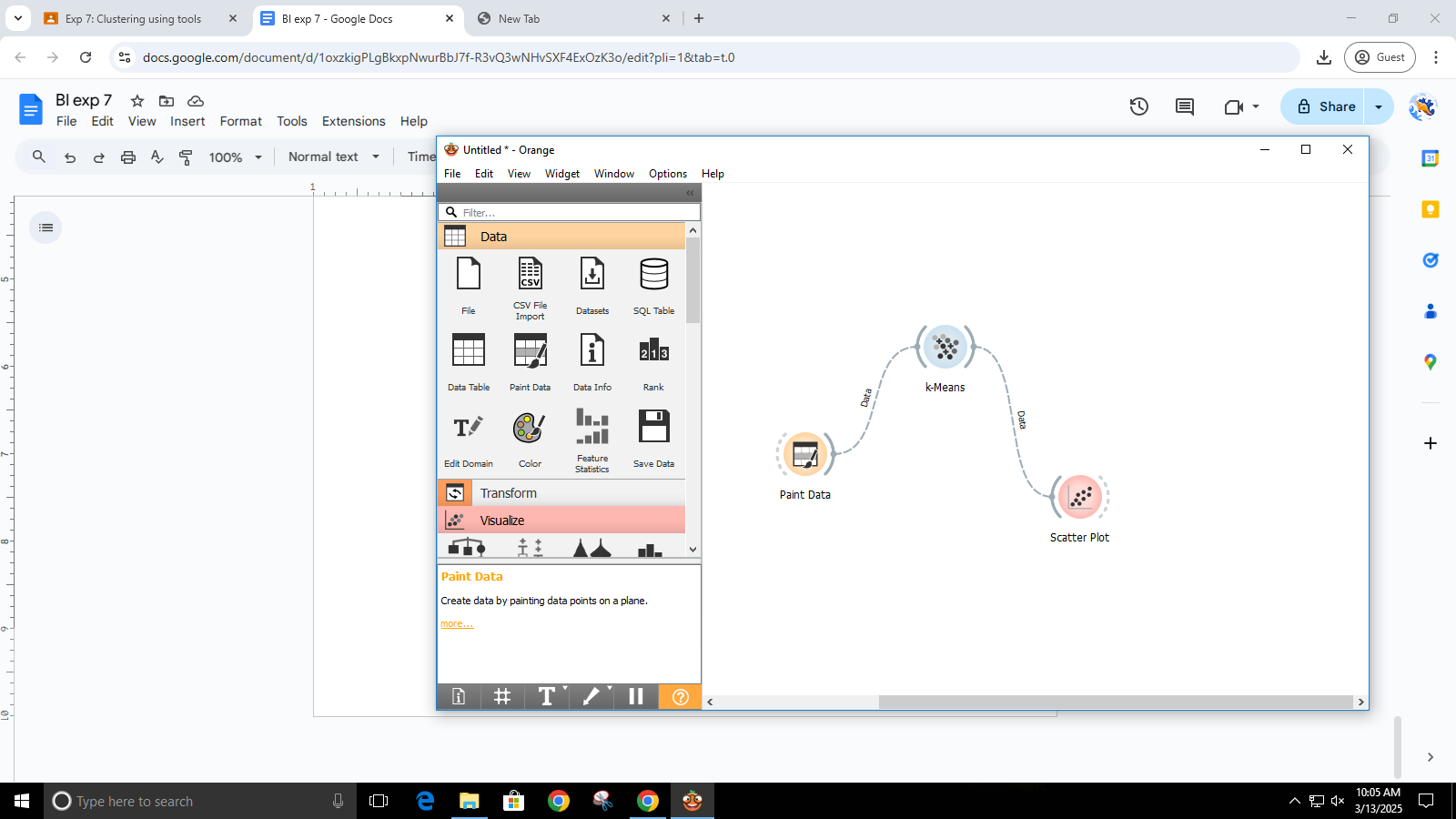
Deciding on k–values with the help of silhouette score



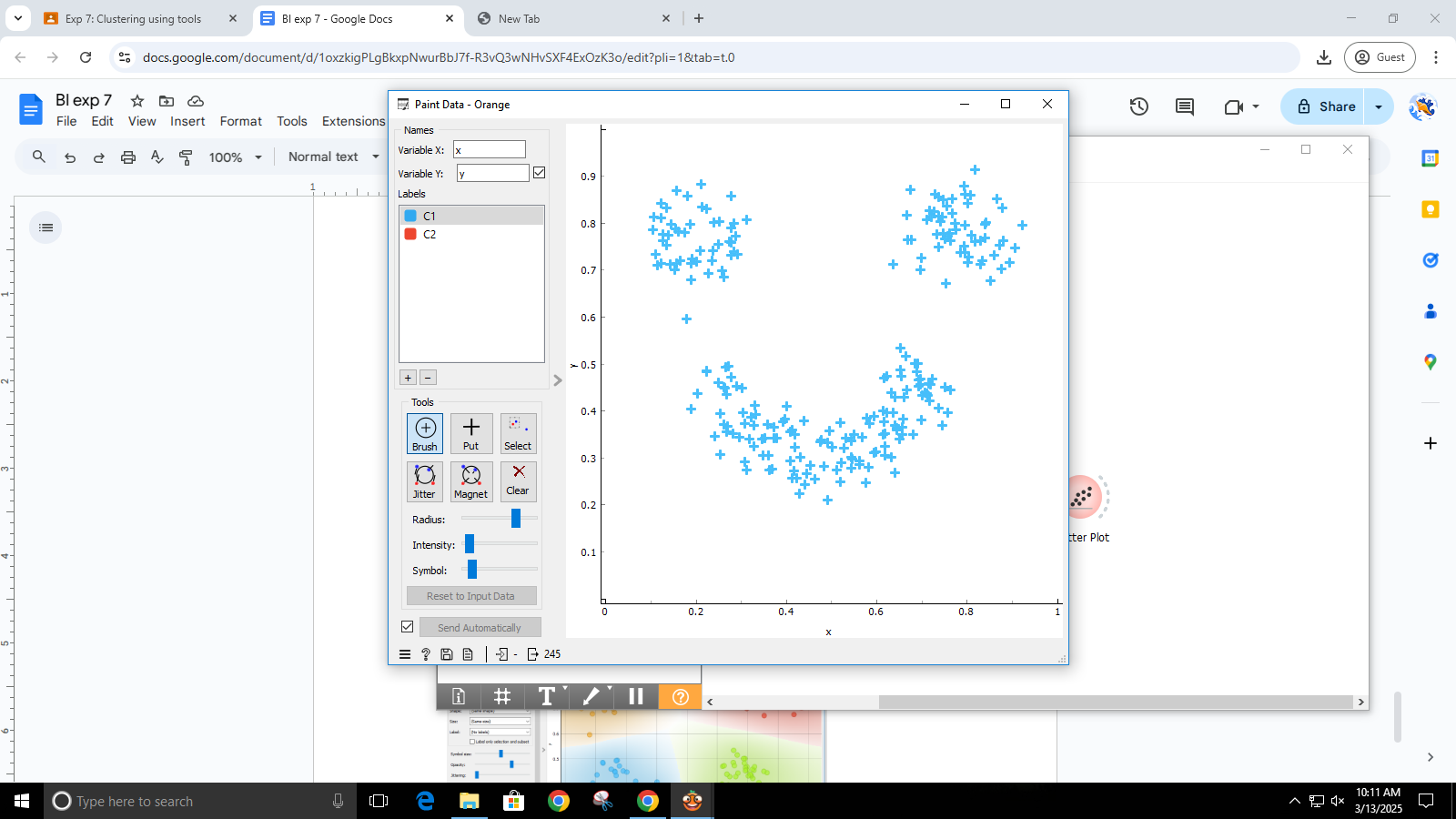
output of k-means with 2 clusters visible in scatter-plot



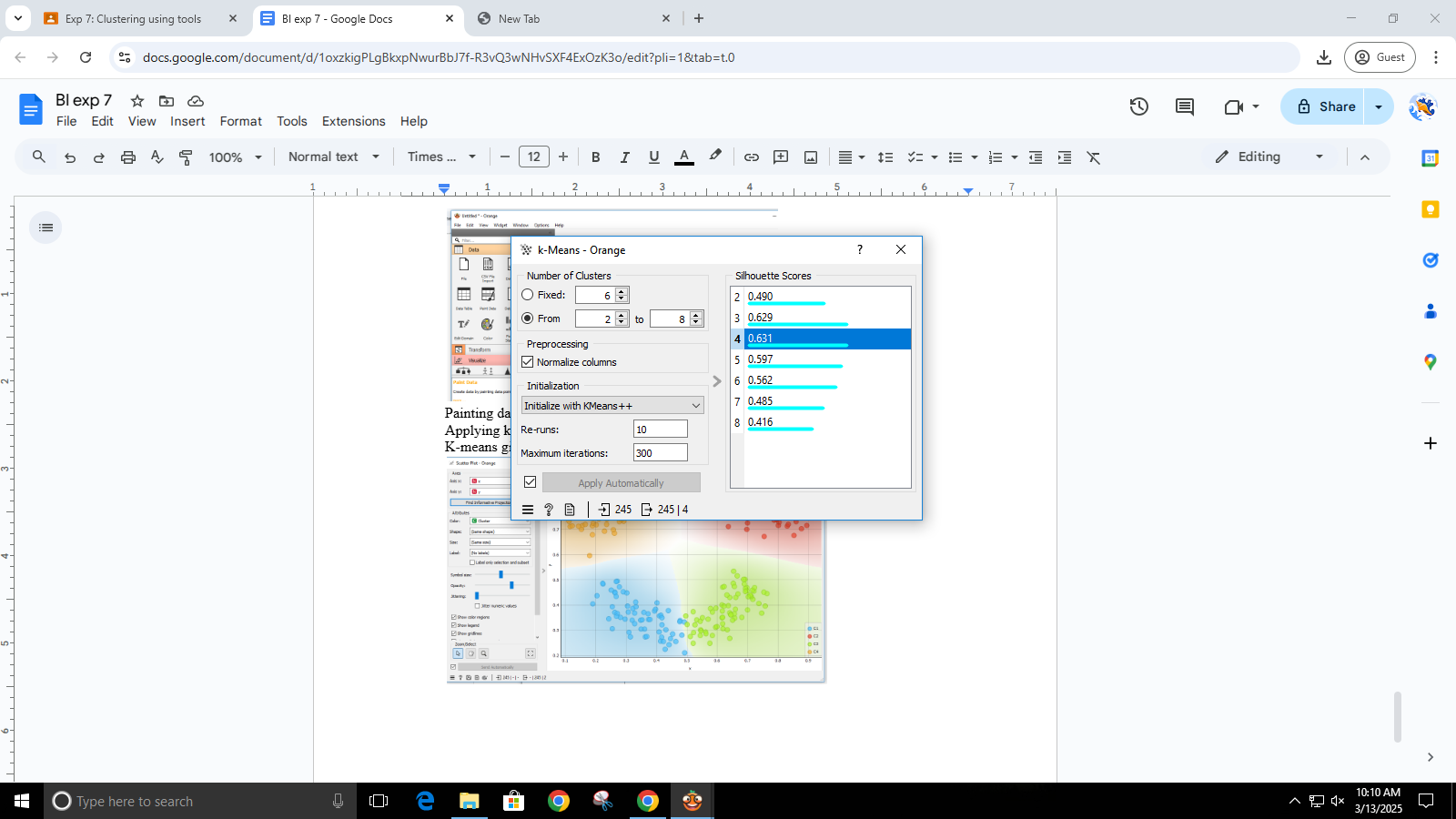
Applying k-means on painted data



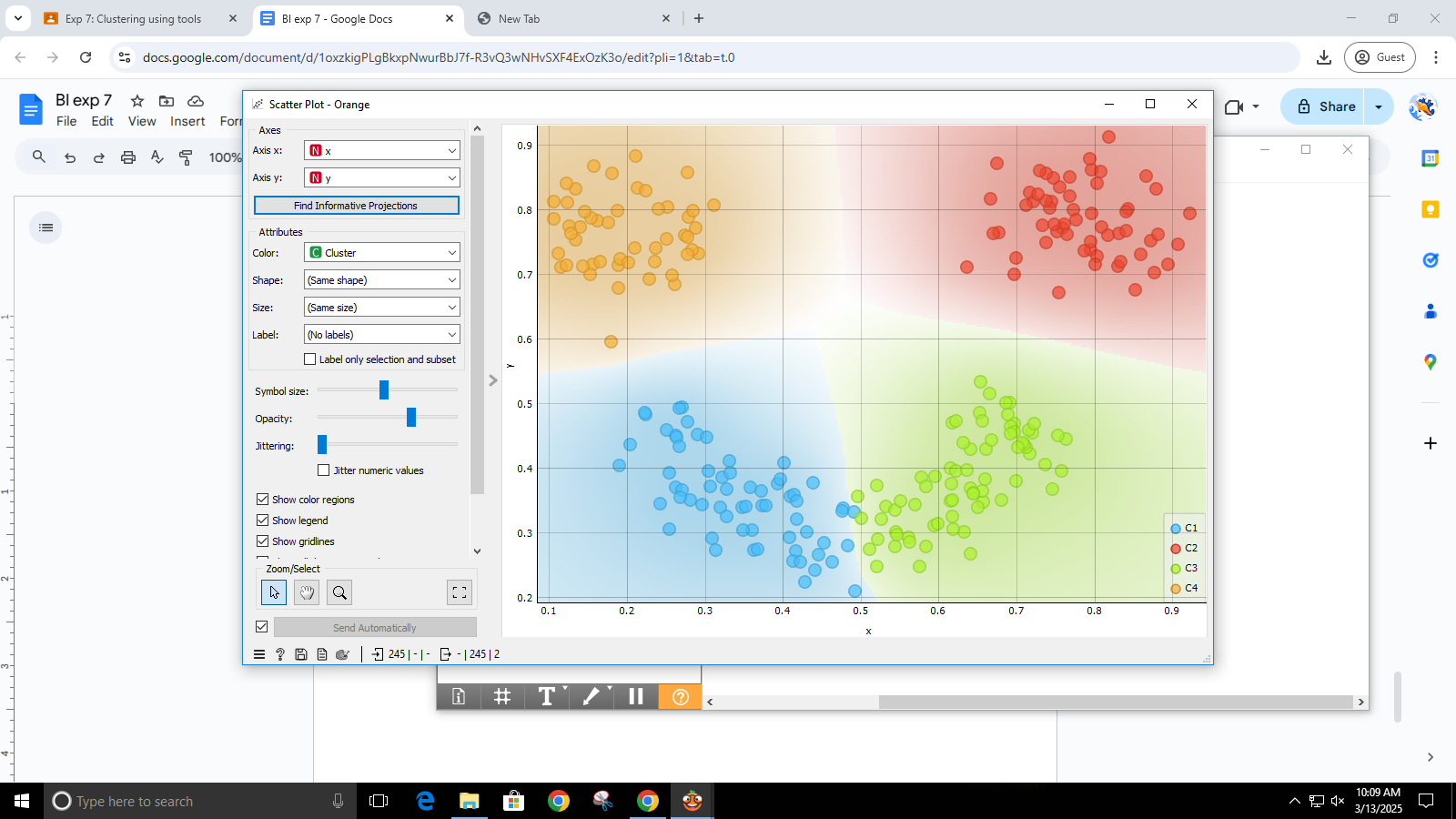
Painting data-points



Applying k-means



K-means giving inaccurate clusters because of elongated shape



Extended theory:

Q.1 Explain K means (graph) algorithm

Ans:

K-Means is a widely used unsupervised machine learning algorithm for clustering data into a predefined number of groups, denoted by K. The goal of the algorithm is to partition the data in such a way that each data point belongs to the cluster with the nearest mean (also known as the centroid), minimizing the within-cluster variance. The process begins by randomly selecting K data points as initial centroids. Each data point is then assigned to the cluster whose centroid is closest in terms of Euclidean distance. After all points have been assigned, the centroids of each cluster are recalculated as the mean of all the points in that cluster. These steps of assignment and updating continue iteratively until the centroids stabilize or a maximum number of iterations is reached. K-Means is efficient and works well with large datasets, but it has some limitations, such as sensitivity to outliers and noise, and the assumption that clusters are spherical and evenly sized. Additionally, it may converge to a local optimum depending on the initial placement of centroids and requires the number of clusters to be specified in advance.

Q.2 Explain K medoids algorithm

Ans:

K-Medoids is a clustering algorithm similar in concept to K-Means, but it is more robust to noise and outliers. Unlike K-Means, which uses the mean of the cluster to define the center, K-Medoids selects actual data points as the center of clusters, known as medoids. The algorithm starts by randomly choosing K data points as the initial medoids. Each remaining data point is then assigned to the nearest medoid based on a chosen distance metric, typically Euclidean distance or Manhattan distance. After assignment, the algorithm attempts to improve the clustering by swapping each medoid with a non-medoid point and calculating the total cost, which is the sum of the distances between each point and its assigned medoid. If the swap results in a lower cost, it is accepted; otherwise, it is rejected. This process continues until no further improvements can be made or a specified number of iterations is reached. Since medoids are actual data points, the results are more interpretable and stable in the presence of outliers. However, K-Medoids is computationally more expensive than K-Means, especially with large datasets, due to the pairwise comparison needed in the medoid update step.