**Annexure 2**

**Business Intelligence and Data Visualization Lab Manual**

**CSL 232**

Project Report

Logo, company name

Description automatically generated

Faculty name: Dr. Poonam Chaudhary Student name: Asha, Aastha

Roll No.: 20CSU374, 20csu321

Semester:5th

Group: DS-B4

**Department of Computer Science and Engineering**

**The NorthCap University, Gurugram- 122001, India**

**Session 2022-23**

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S.No** |  | **Page No.** |
| **1.** | **Project Description** | **1** |
| **2.** | **Problem Statement** | **1** |
| **3.** | **Analysis**  **3.1 Hardware Requirements**  **3.2 Software Requirements** | **1** |
| **4.** | **Design**  **4.1 Data/Input Output Description:**  **4.2 Algorithmic Approach / Algorithm / DFD / ER diagram/Program Steps** | **2** |
| **5.** | **Implementation and Testing (stage/module wise)** | **5** |
| **6.** | **Output (Screenshots)** | **5** |
| **7.** | **Conclusion and Future Scope** | **6** |

**Project Description**

Solves a classification problem on the iris dataset using the K-means and decision tree.

**Problem Statement**

Classifying sales dataset using K nearest Neighbor Algorithm and decision tree.

**Analysis**

**3.1 Hardware Requirements**

A 64-bit operating system with at least 2GB RAM and 2 CPU cores.

**3.2 Software Requirements**

KNIME Analytics

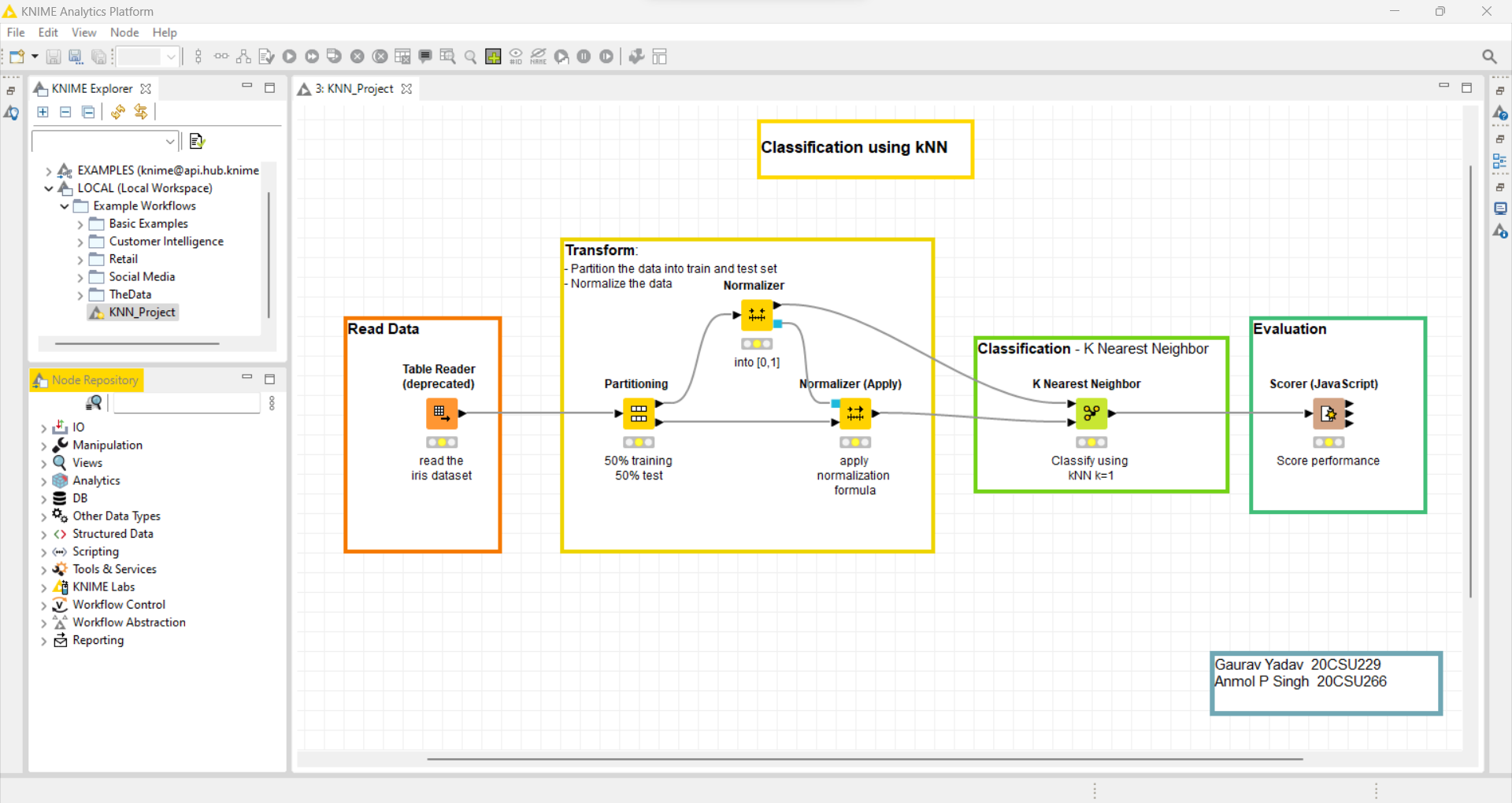
**Design**

**4.1 Data Input/Output Description:**

* The dataset contains the information of sales and revenue, and customer’s feedback details on the basis of sales.
* The dataset was taken from the site Kaggle and was updated till last date.

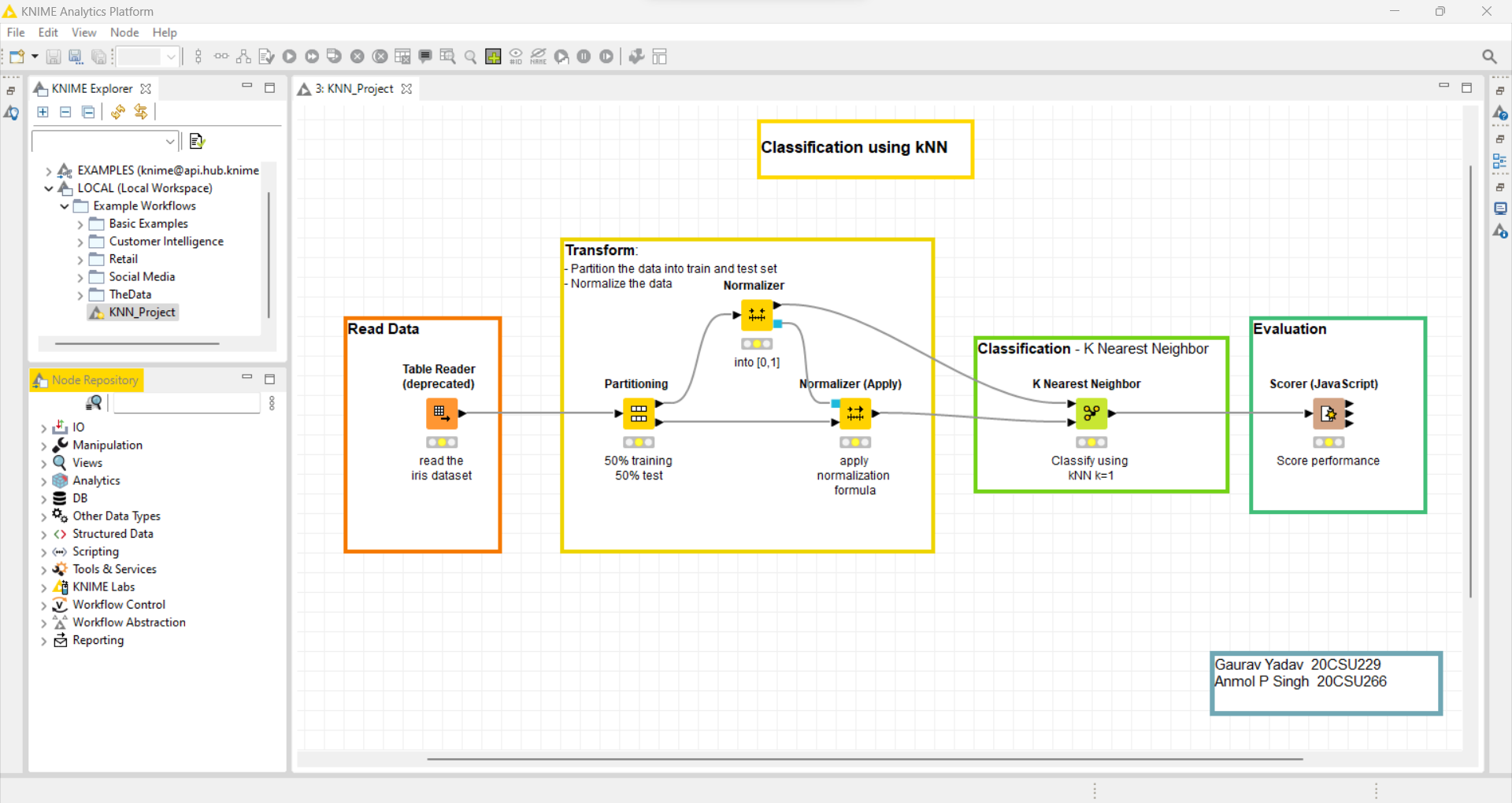
**4.2 Algorithmic Approach / Program Steps**

**Reading Data**

****

This node reads files that have been written using the Table Writer node (which uses an internal format). It retains all meta information such as customer feedbacks, sales and revenue.

**Transforming Data**



Partitioning: The input table is split into two partitions (i.e., row-wise), e.g., train and test data. The two partitions are available at the two output ports.

Normalizer: This node normalizes the values of all (numeric) columns. In the dialog, you can choose the columns you want to work on.

Normalizer (Apply): This node normalizes the input data according to the normalization parameters as given in the model input (typically coming from the Normalizer node). It will apply an affine transformation to all columns in the input data that are contained in the model input. This node is typically used when test data shall be normalized the same way the training data has been normalized (using the "Normalizer" node).

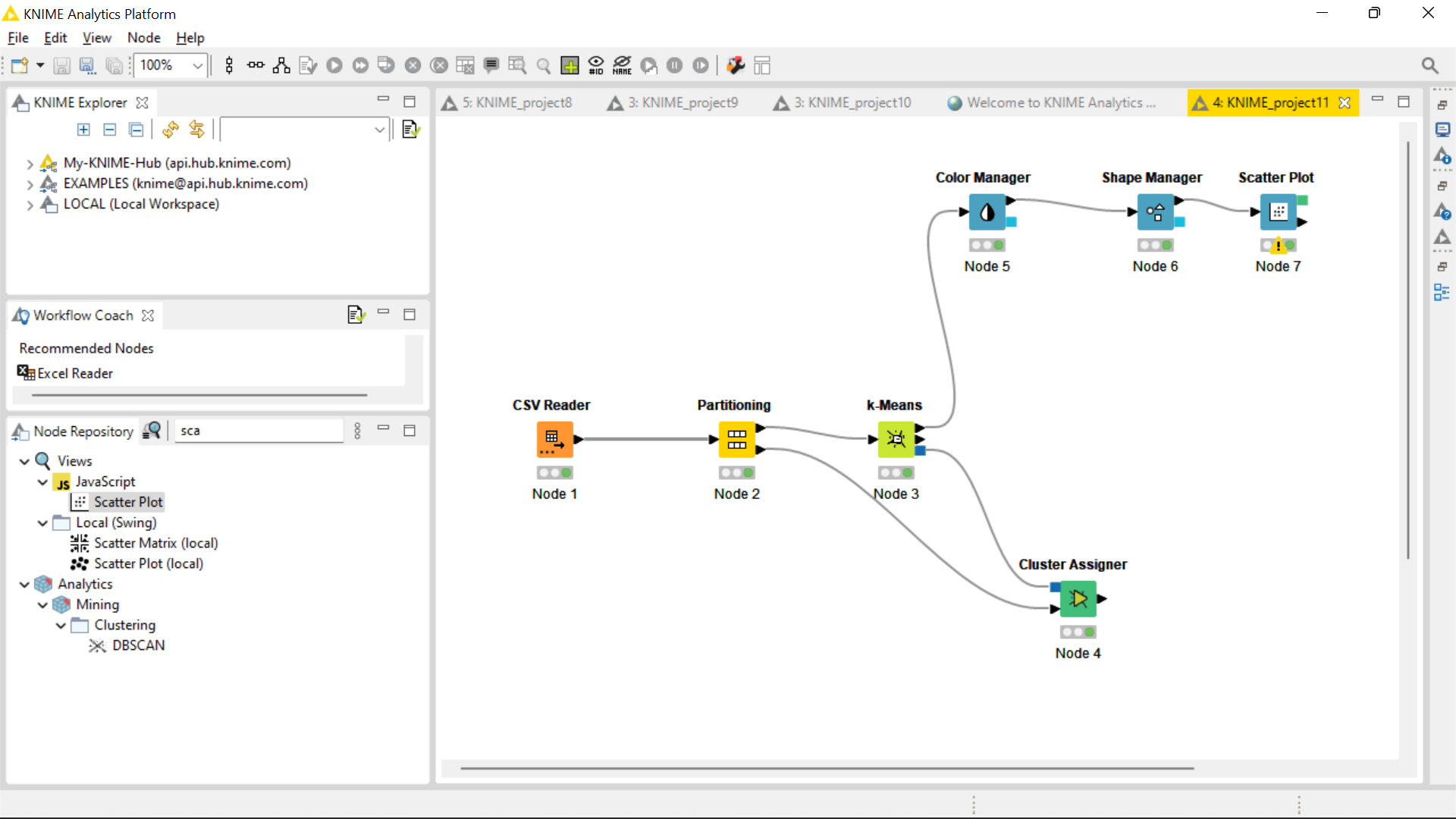
**Classification**

Diagram

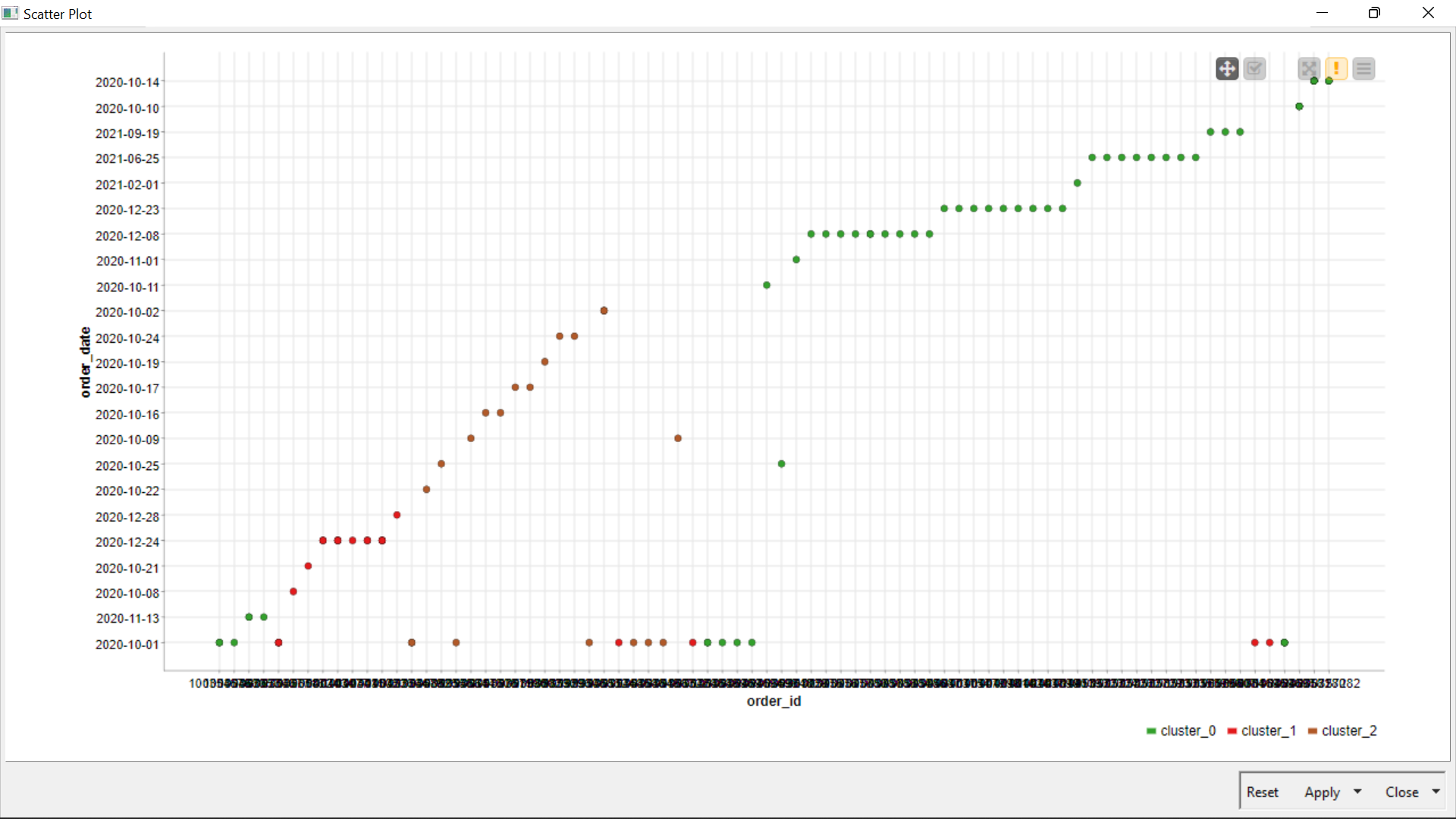
Description automatically generated

K Nearest Neighbor: Classifies a set of test data based on the k Nearest Neighbor algorithm using the training data. The underlying algorithm uses a KD tree and should therefore exhibit reasonable performance. However, this type of classifier is still only suited for a few thousand to ten thousand or so training instances. All (and only) numeric columns and the Euclidean distance are used in this implementation. All other columns (of non-numeric type) in the test data are being forwarded as-is to the output.

**Implementation and Testing**



**Output**



**Implementation:**

