**Experiment no. 3**

**Name: Sonali Dattatray Kaingade**

**PRN: 21620002**

**Title:** To perform Binning of data

**Code:**

#include <iostream>

#include <fstream>

#include <vector>

#include<algorithm>

#include <climits>

#include<cmath>

using namespace std;

//equal frequency

vector<vector<int>> equi\_frequency(vector<int> data,double m)

{

    double n=data.size();

    double ele=ceil(n/m);

    vector<vector<int>> totalbins;

    for(int i=0;i<m;i++)

    {

        vector<int> bin;

        for(int j=i\*ele;j<(i+1)\*ele;j++)

        {

            if(j>=n)

            {

                break;

            }

            bin.push\_back(data[j]);

        }

        totalbins.push\_back(bin);

    }

    return totalbins;

}

//equal width

vector<vector<int>>equi\_width(vector<int> data,int m)

{

    int n=data.size();

    int min\_ele=INT\_MAX;

    int max\_ele=INT\_MIN;

    for(int i=0;i<data.size();i++)

    {

        min\_ele= min(min\_ele,data[i]);

        max\_ele= max(max\_ele,data[i]);

    }

    int w = (max\_ele-min\_ele)/m;

    int min1=min\_ele;

    vector<int> arr;

    for(int i=0;i<m+1;i++)

    {

        arr.push\_back(min1+w\*i);

    }

    vector<vector<int>> arri;

    for(int i=0;i<m;i++)

    {

        vector<int> temp;

        for(int k:data)

        {

            if(k>=arr[i] && k<=arr[i+1])

            {

                temp.push\_back(k);

            }

        }

        arri.push\_back(temp);

    }

return arri;

}

// Write binning outputs to CSV

void writeCSV(string filename, vector<vector<int>> bins)

{

  ofstream outputFile(filename);

  for (int i = 0; i < bins.size(); i++)

  {

    outputFile << "Bin " << i + 1 << ":"<<" ";

    for (int num : bins[i])

    {

      outputFile << num << ",";

    }

    outputFile << "\n";

  }

  outputFile.close();

}

int main()

{

    ifstream inputf("input.csv");

    vector<int> data;

    int val;

    while(inputf>>val)

    {

        data.push\_back(val);

    }

    sort(data.begin(),data.end());

     int method,m;

  cout << "Choose binning method: " << endl;

  cout << "1. Equal Frequency Binning" << endl;

  cout << "2. Equal Width Binning" << endl;

  cout << "\nEnter method number: ";

  cin >> method;

  cout << "\nEnter number of bins: ";

  cin >> m;

  if (method == 1)

  {

    vector<vector<int>> freqBins = equi\_frequency(data, m);

    writeCSV("output\_equi\_frequency.csv", freqBins);

  }

  else if (method == 2)

  {

    vector<vector<int>> widthBins = equi\_width(data, m);

    writeCSV("output\_equi\_width.csv", widthBins);

  }

  else

  {

    cout << "Invalid method choice." << endl;

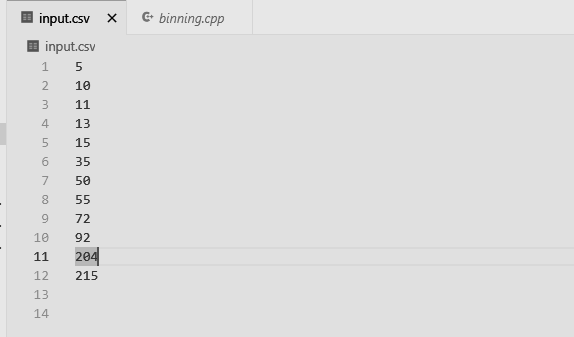
  }

    return 0;

}

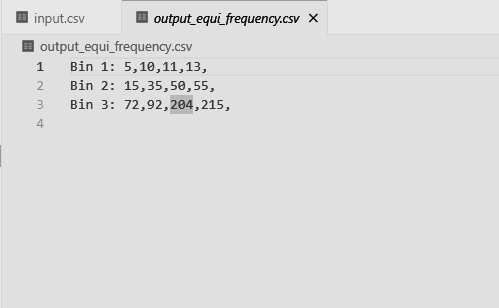
**Output:**

**Input.csv**

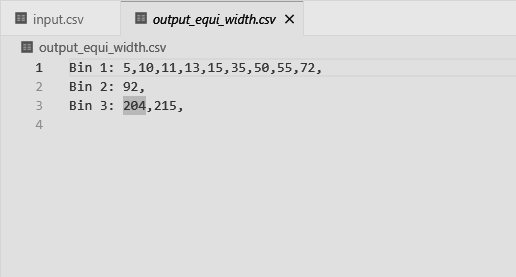


**Output.csv**

**1.Equal frequency**



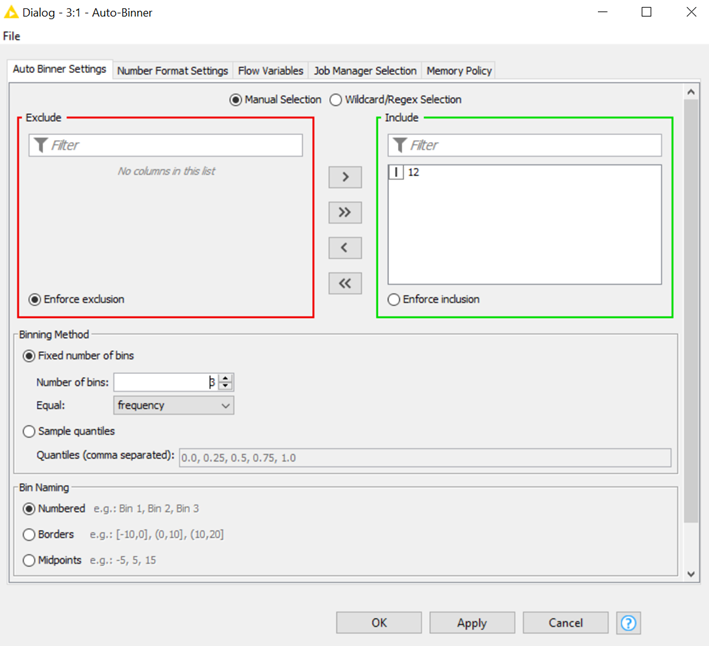
**2. Equal width**

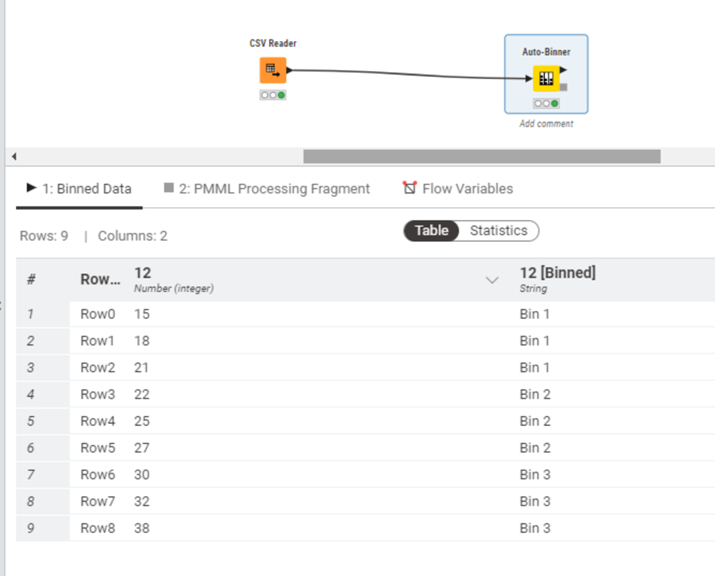


Knime:

**Binning methods:**

1. Equi- Frequency:





1. Equi width:

