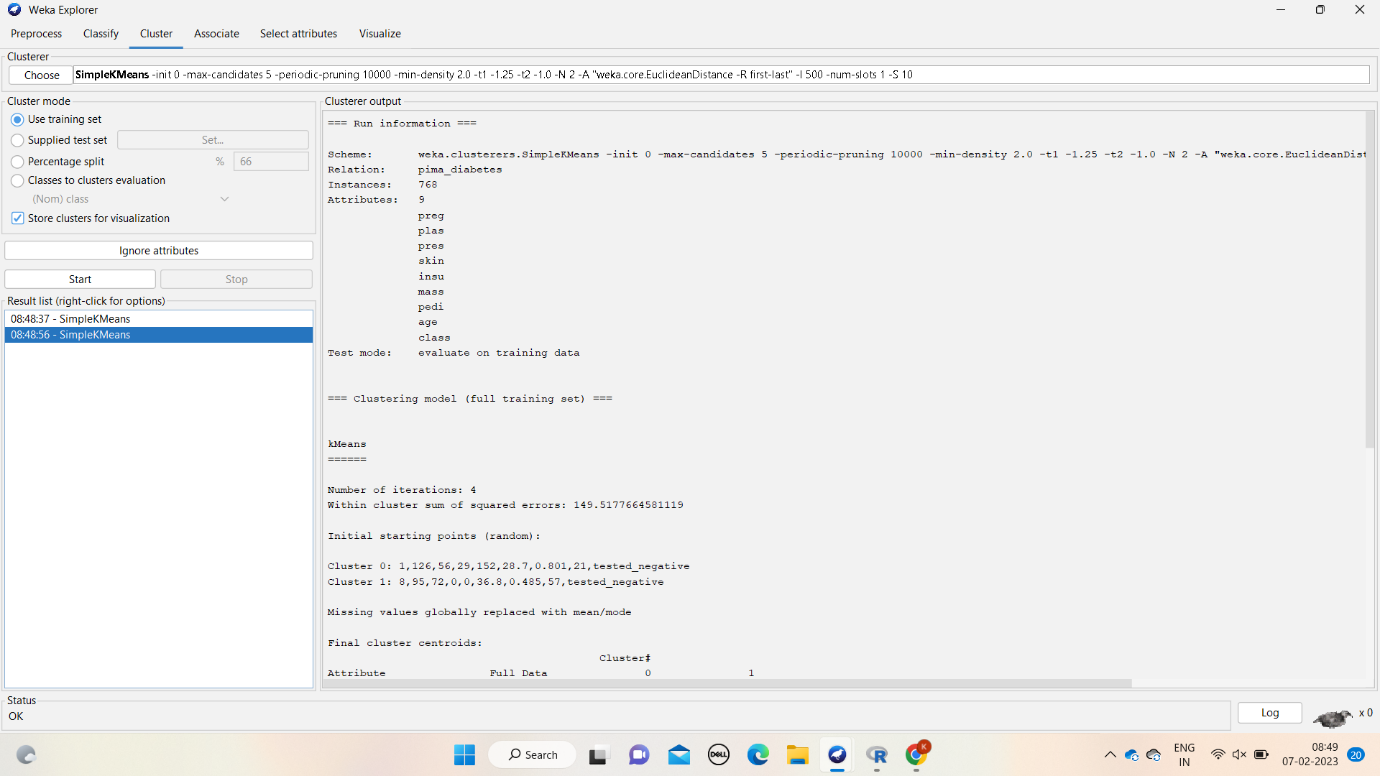
1.write the k-means algorithm and problem using weka tool

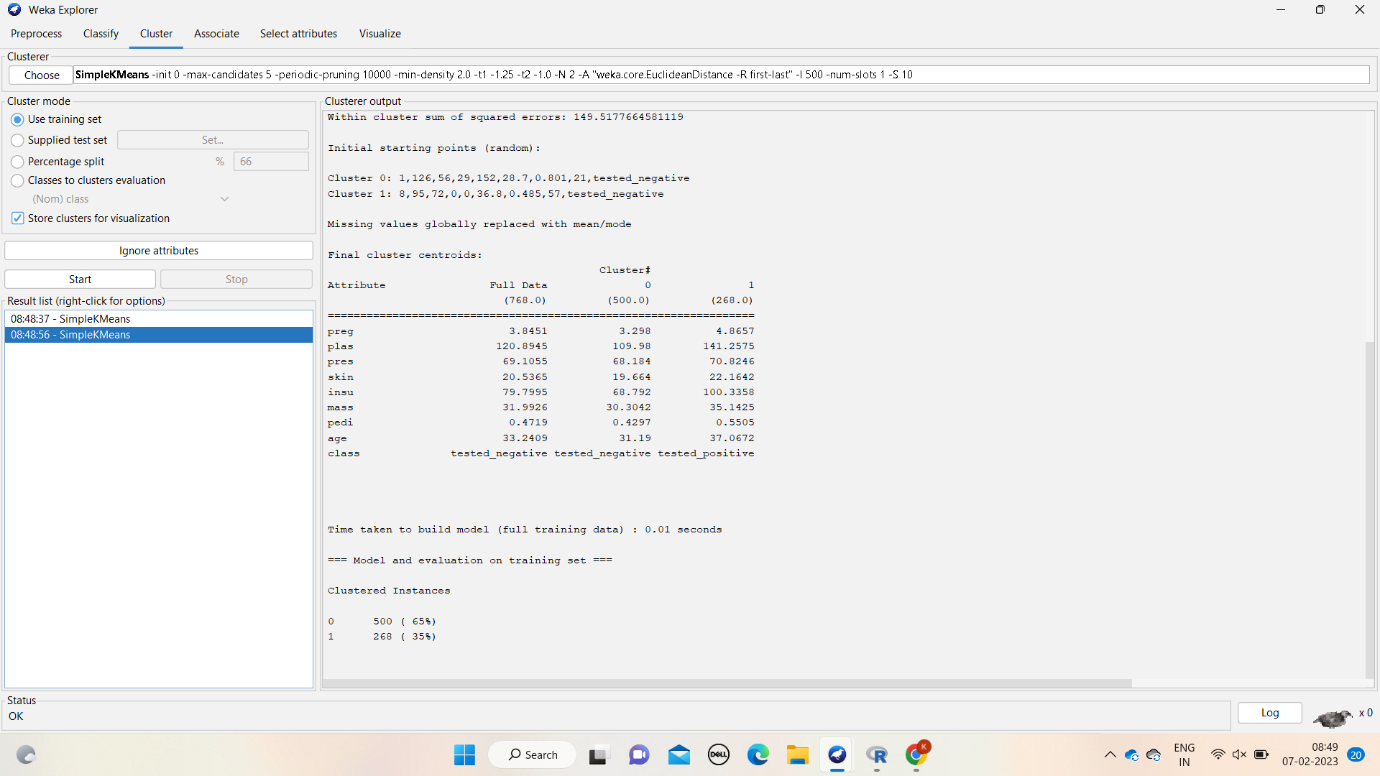
**DATA SET:** DIABETICS

**ALGORITHM:**

1. First, we need to specify the number of cluster(k)need to assign.
2. Next, randomly select the K data points into the cluster
3. Now, it will compute the clusters centroids.
4. Iterate following steps until to find the final steps.
5. Sum of squares of distance between data points and centroids is less when compared to others.
6. At last compute the centroids for cluster by taking average of all data points of cluster.
7. Compute centroids of k-means clustering.

**OUTPUT:**

****



**RESULT:** Hence, k-means clustering has proven using weka tool.

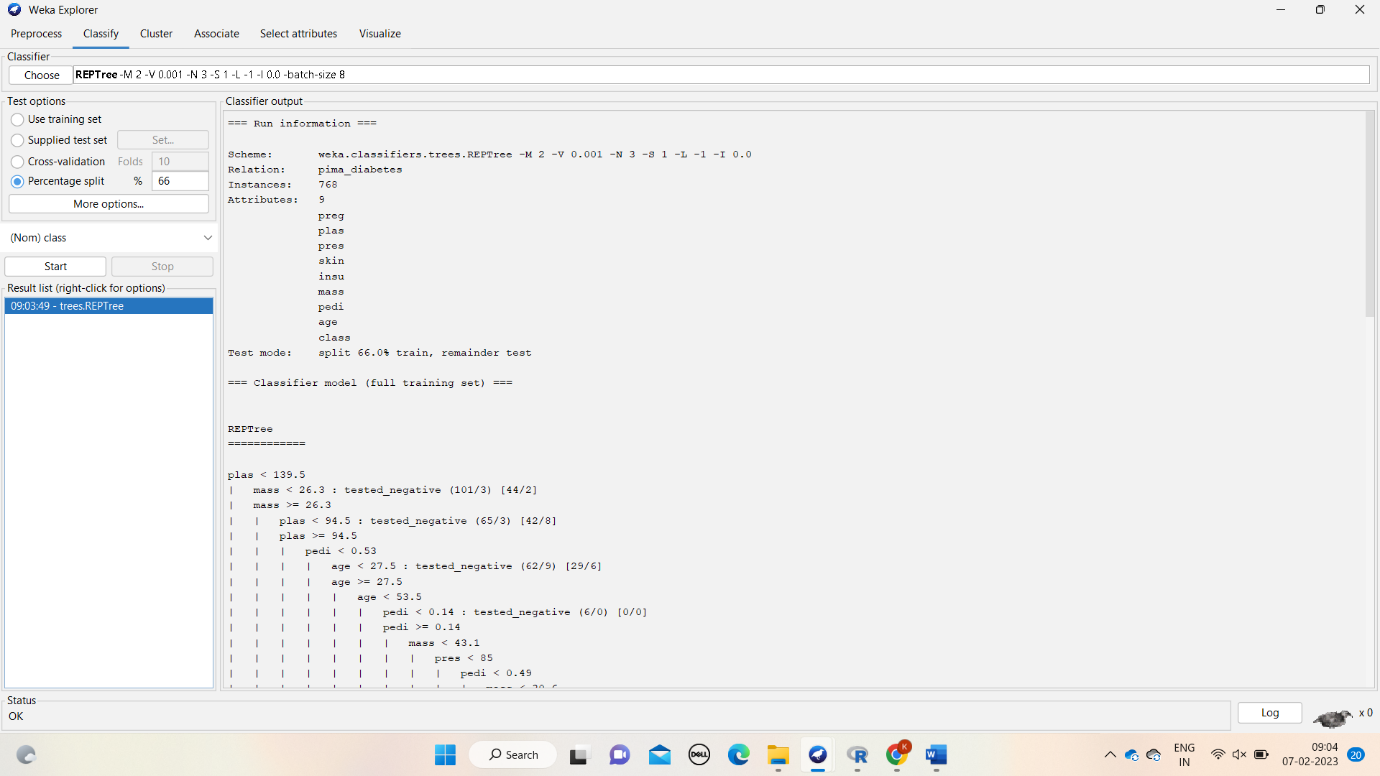
2.construct the decision tree using weka tool.

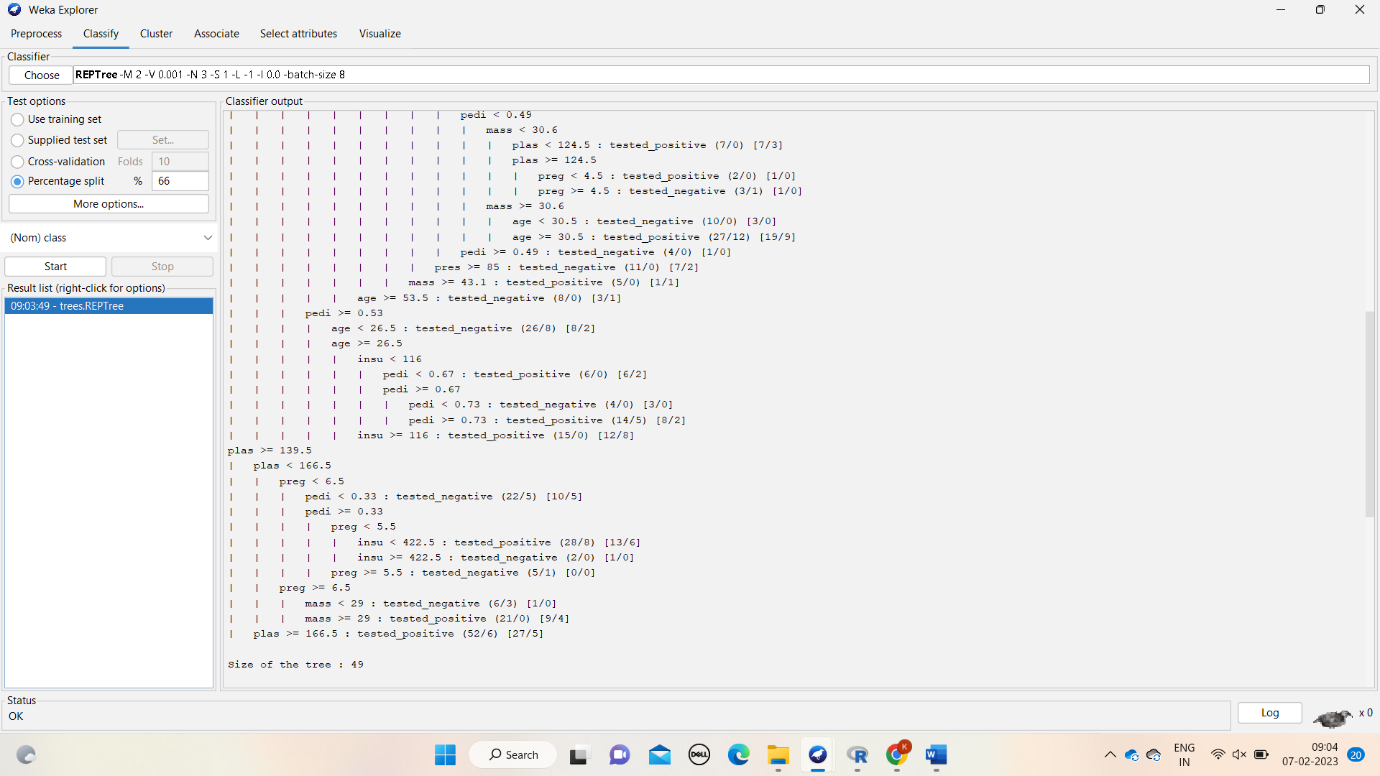
**DATASET:** diabetics

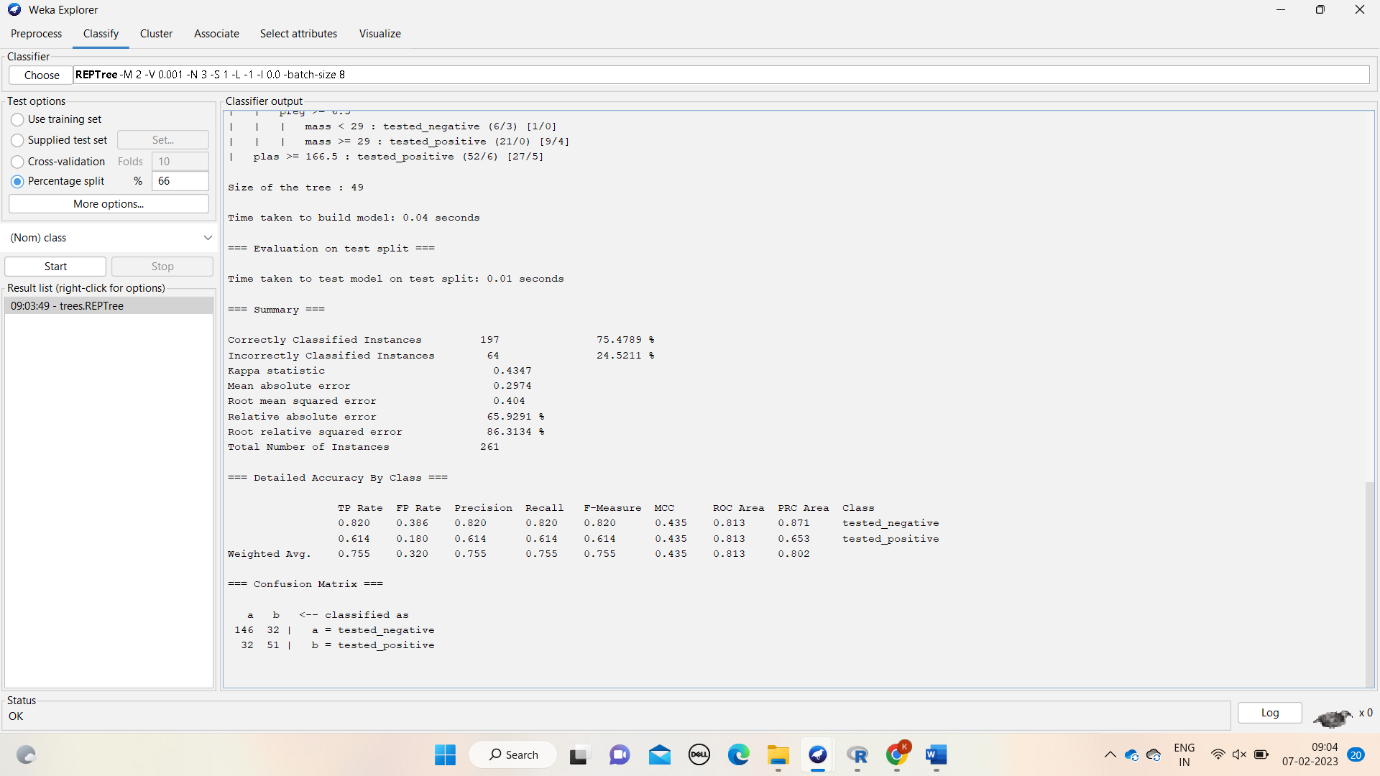
**ALGORITHM:**

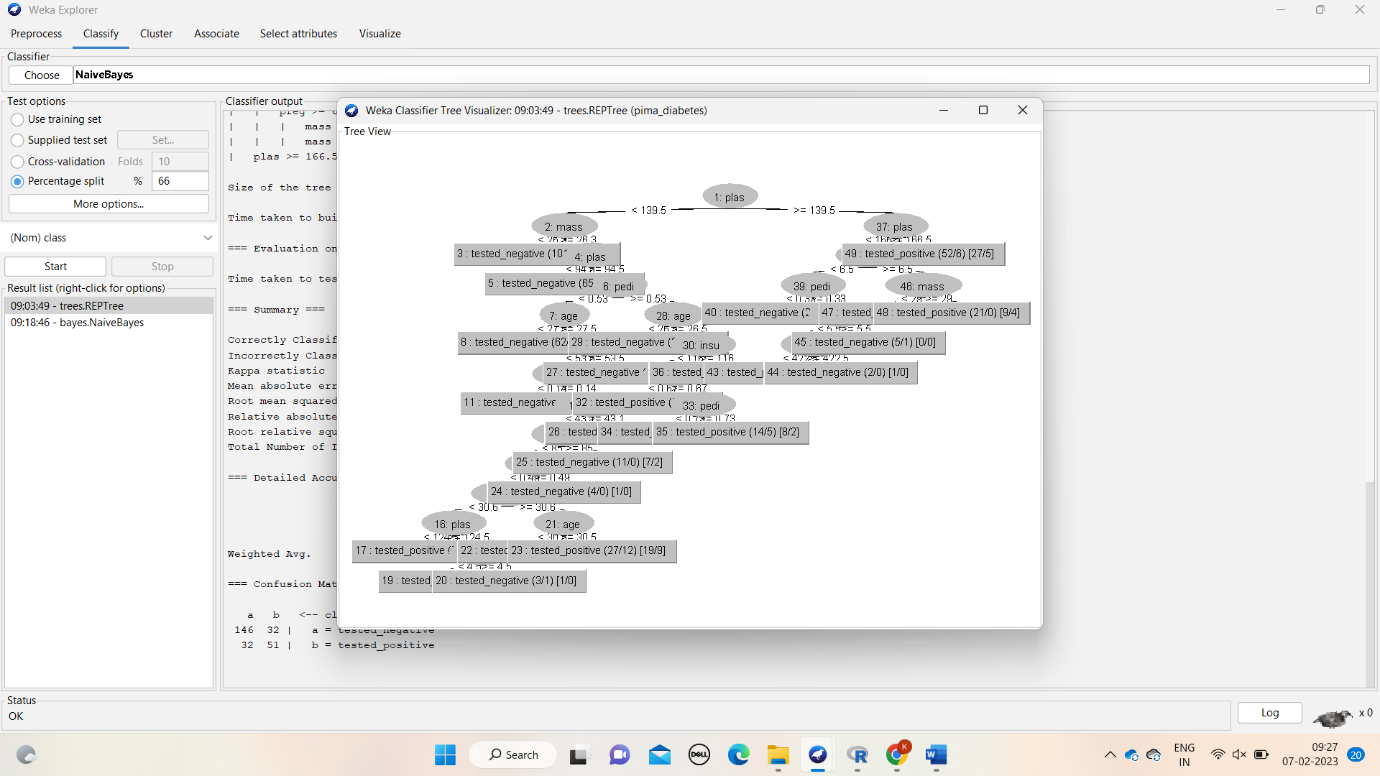
1. Determine the root node.
2. Calculate the entropy for the classes.
3. **ENTROPY(a,b**)=  **Σ p(c)log2e(c) where, p©= probility ,e©=entropy.**
4. Calculate entropy after split for each attribute.
5. Calculate information gain
6. **IG**= (parent node - sum of weights of entropy of child node).
7. Perform split(branch).
8. Perform further spilt.
9. Compute decision tree.

**OUTPUT:**







****

**RESULT:** hence, decision tree has proved using the weka tool.

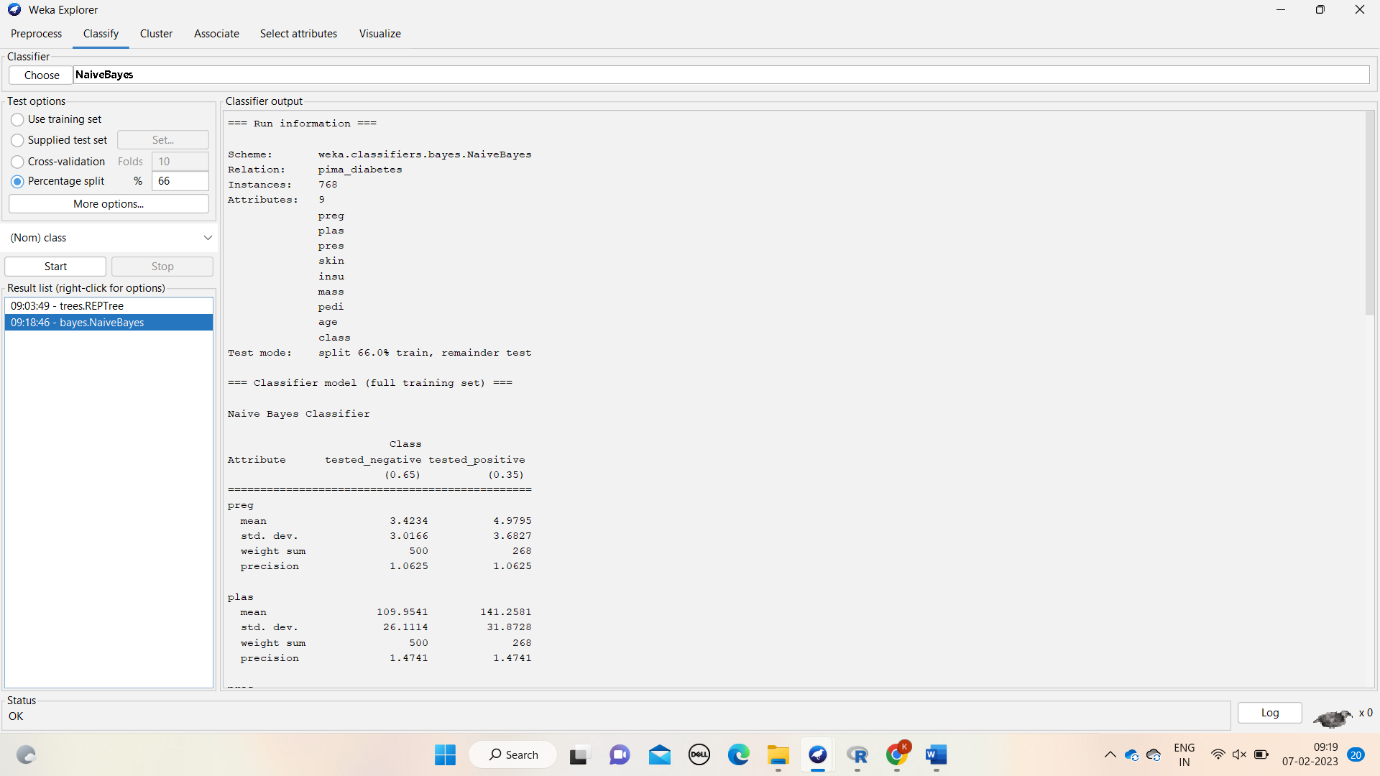
3.construct the Bayesian classification using weka tool

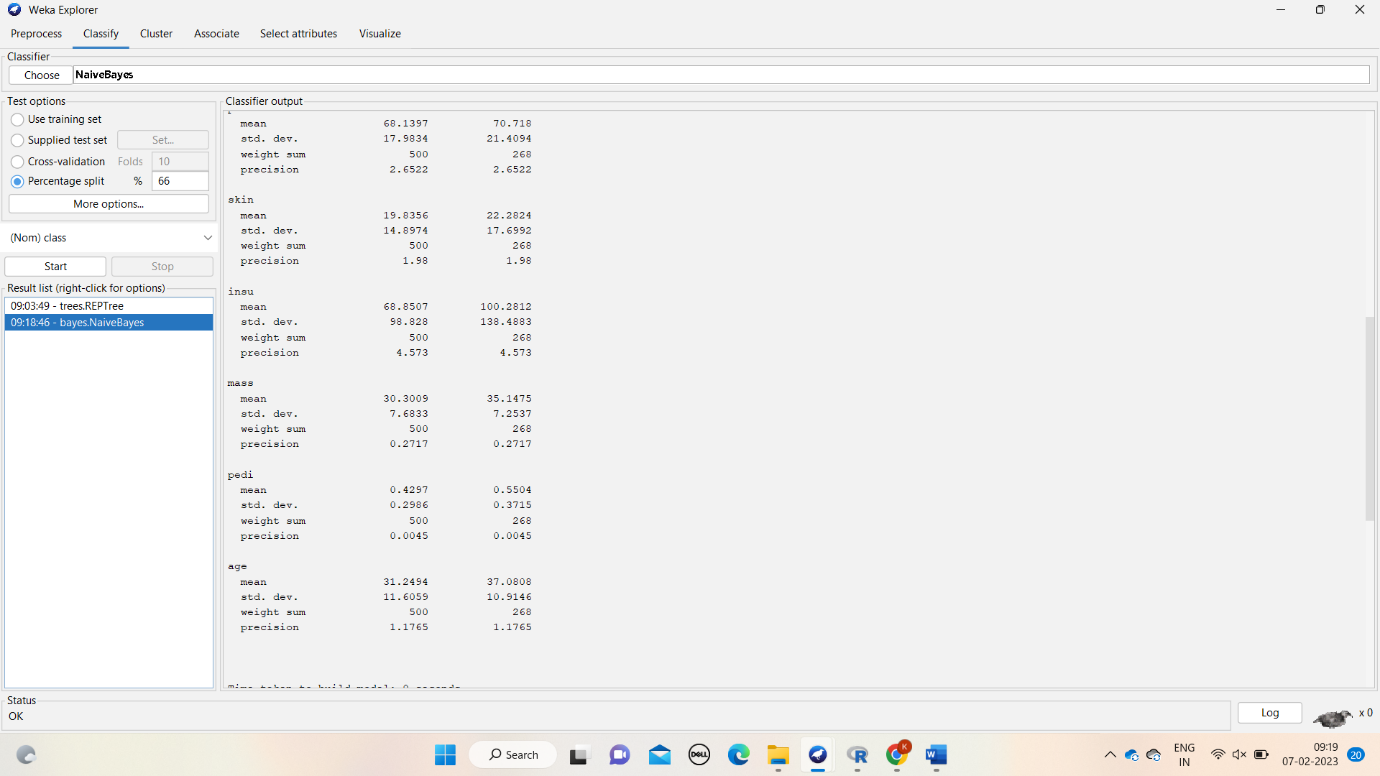
**DATASET:** diabetics

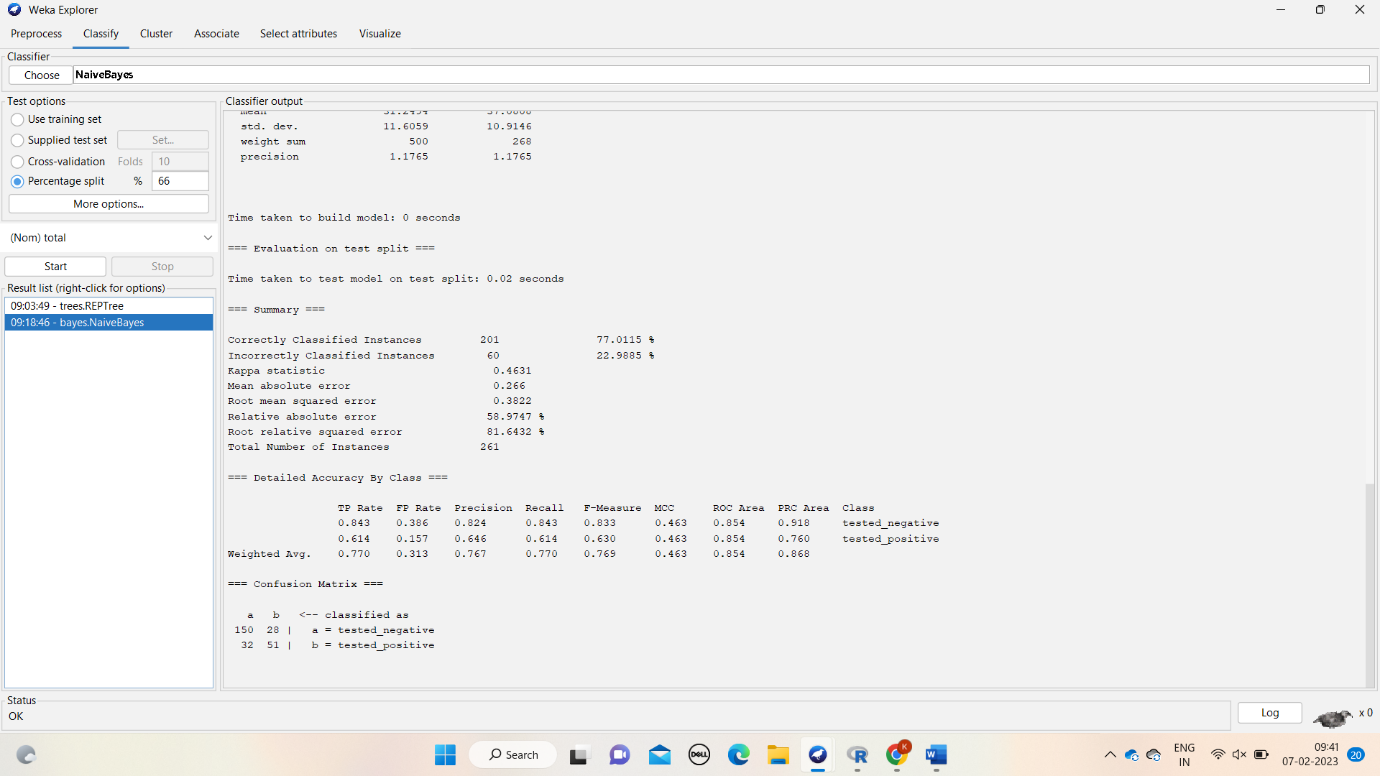
**ALGORITHM:**

1. Convert dataset into the frequency table.
2. Now convergt table into another table to find probability
3. Use bayes thorem to above table.
4. BAYES THOREM = p(a/b)= p(b/a) p(a)/p(b)
5. Use above theorem to all the attributes columns .
6. Now compare the all the output of given probabilities.
7. See the efficient one which has less probability value.
8. Compute probability of Bayesian classification.

**OUTPUT:**







**RESULT**: hence, bayseian classification has proven using the weka tool.

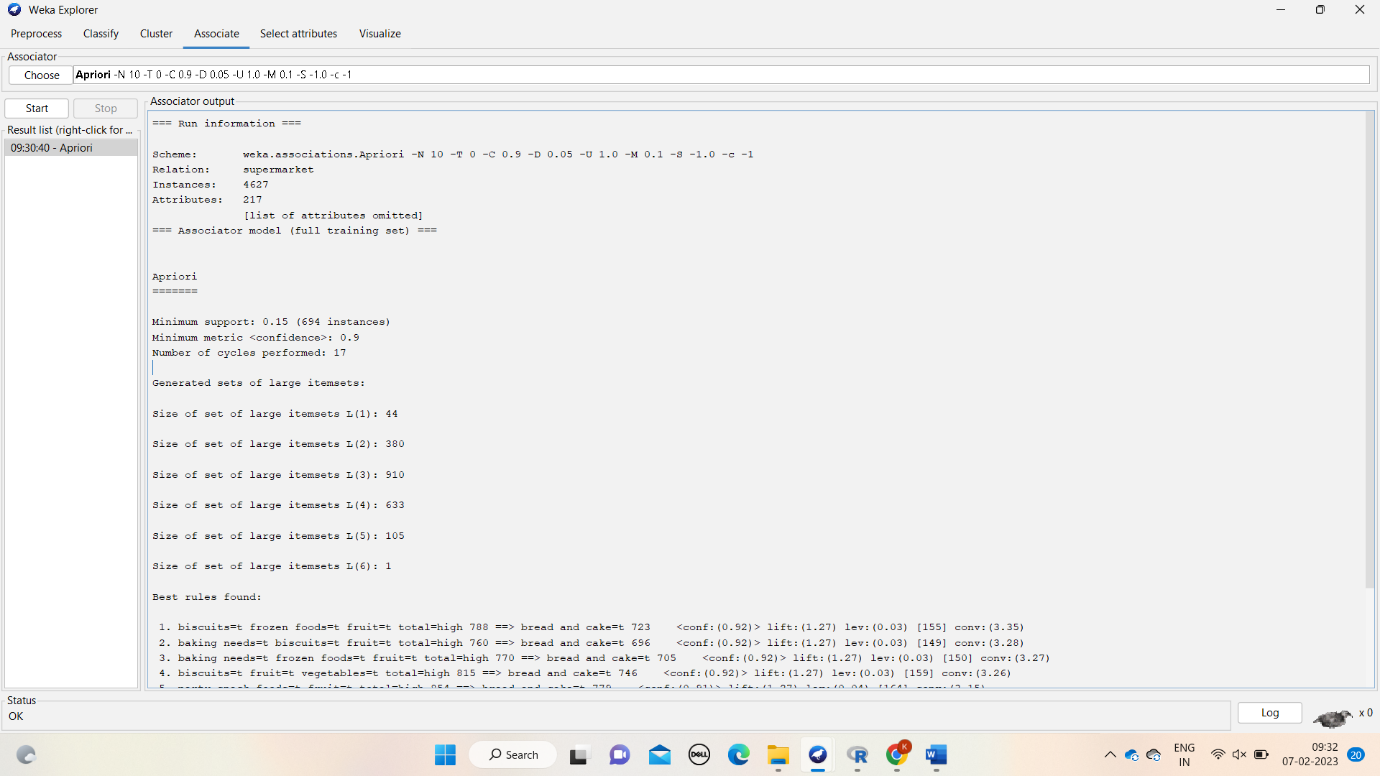
4.construct the apriori using weka tool

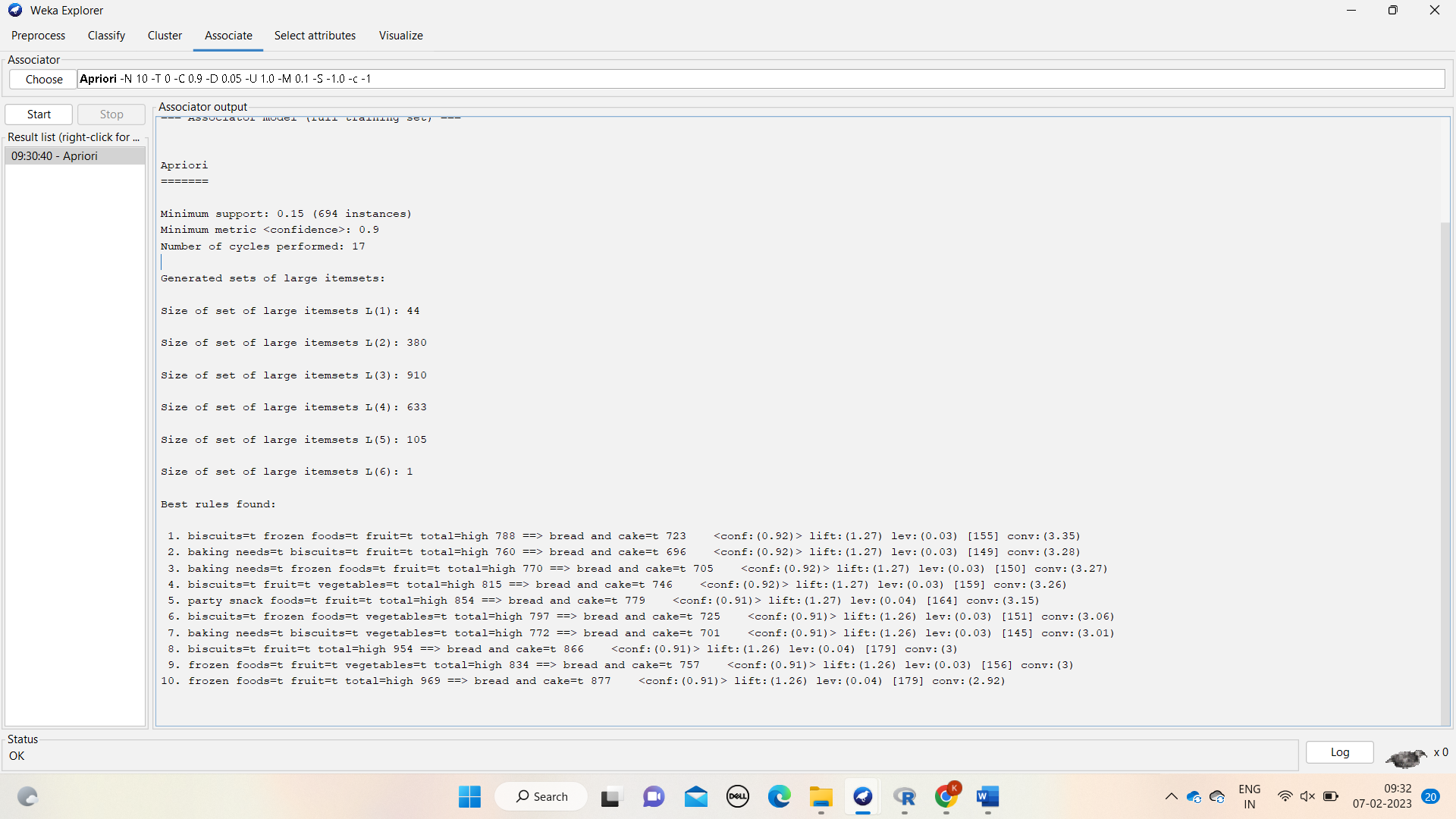
**DATA SET:** supermarket

**ALGORITHM:**

1. We have to draw frequency table for items and support count and called as c1(candidate key)
2. Compare c1 support count with minimum support . if support count is <= msc , then remove items.
3. Sort the elements in descending order after elimination
4. Iterate the above process for all the subsets of item until the support count ==0.
5. **CONFIDENCE**=sup( A ∩ B)/Sup(a).
6. Now do the confidence to all the item e=which is equal or greter than msv.
7. **Percentage**=confidence.100=100%.
8. Hence, find the more efficient one among all.

**OUTPUT:**





**RESULT:** hence, apriori has proven using the weka tool

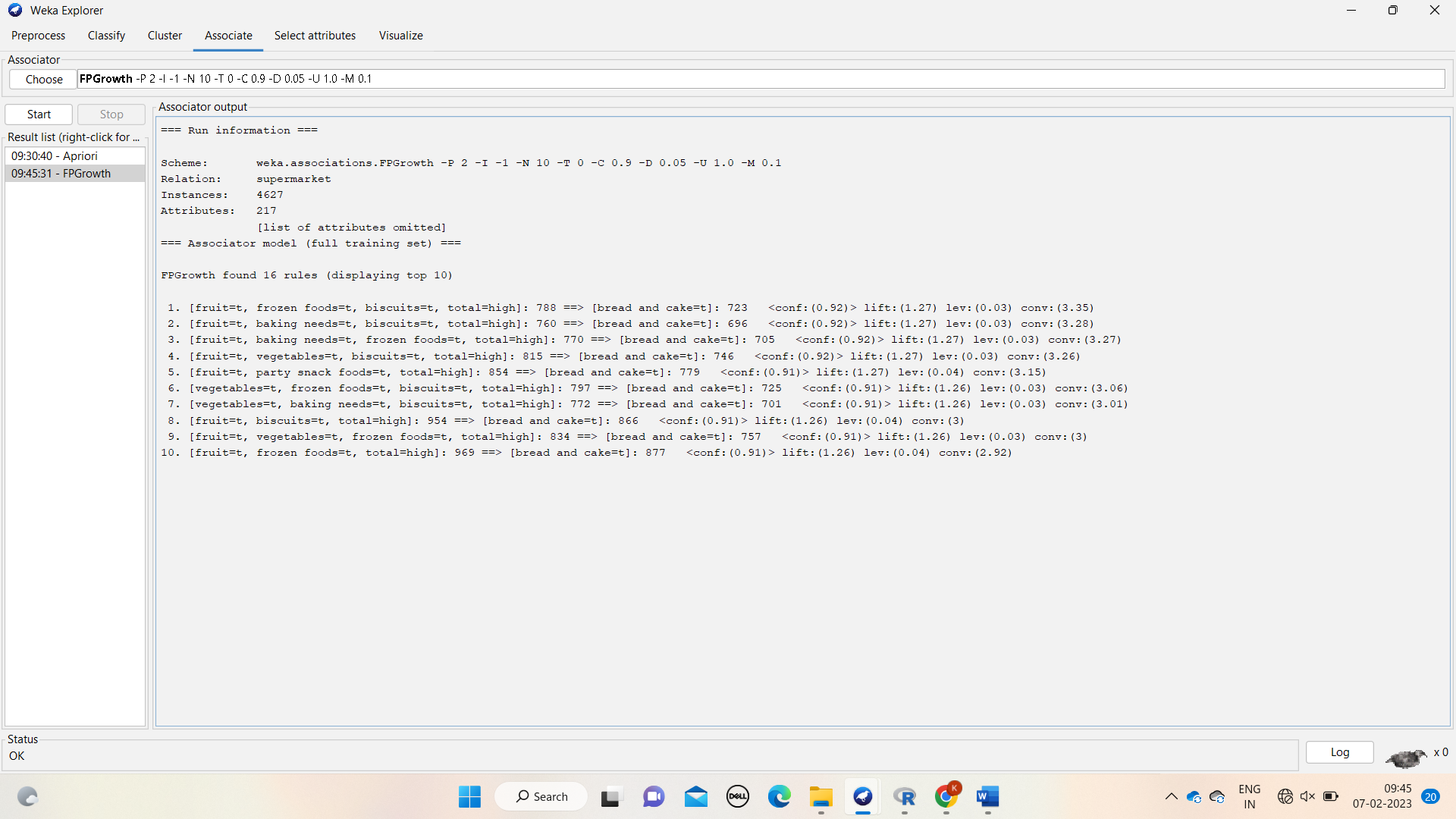
5.construct the FB -GROWTH using the weka tool

DATASET: supermarket

ALGORITHM:

1. We have to draw frequency table for items and support count and called as c1(candidate key)
2. Compare c1 support count with minimum support . if support count is <= msc , then remove items.
3. Sort the elements in descending order after elimination
4. Sore the items in ordered item sets .
5. Construct the fb growth tree
6. Compute the conditional pattern .
7. Compute the conditional frequency pattern
8. Compute the fp growth pattern .

OUTPUT:



RESULT: hence, fp-growth has proven using weka tool.