1 INTRODUCTION

1.1 Overview

A food delivery service has to deal with a lot of perishable raw materials which makes it all, the most important factor for such a company is to accurately forecast daily and weekly demand. Too much inventory in the warehouse means more risk of wastage, and not enough could lead to out-of-stocks and push customers to seek solutions from your competitors. The replenishment of the majority of raw materials is done on weekly basis and since the raw material is perishable, the procurement planning is of utmost importance, the task is to predict the demand for the next 10 weeks.

1.2 Purpose

The main aim of this project is to create an appropriate machine learning model to forecast the raw materials for next few weeks.

2 LITERATURE SURVEY

2.1 Existing problem Existing approaches or method to solve this problem :

Used weka models to solve this problem.

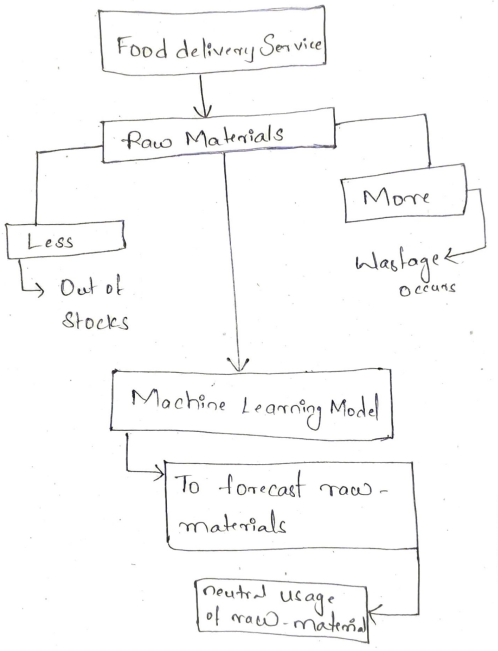
2.2 Proposed solution:

I used weka machine learning modelling approach to get the solution.

3 THEORITICAL ANALYSIS

3.1 Block diagram

Diagrammatic overview of the project.



3.2 Hardware / Software designing

Hardware and software requirements of the project:

Hardware Requirements: 1. PC or laptop with efficient processors and memory.

Software requirements: 1.Eclipse

2.Weka 3.8.5 GUI

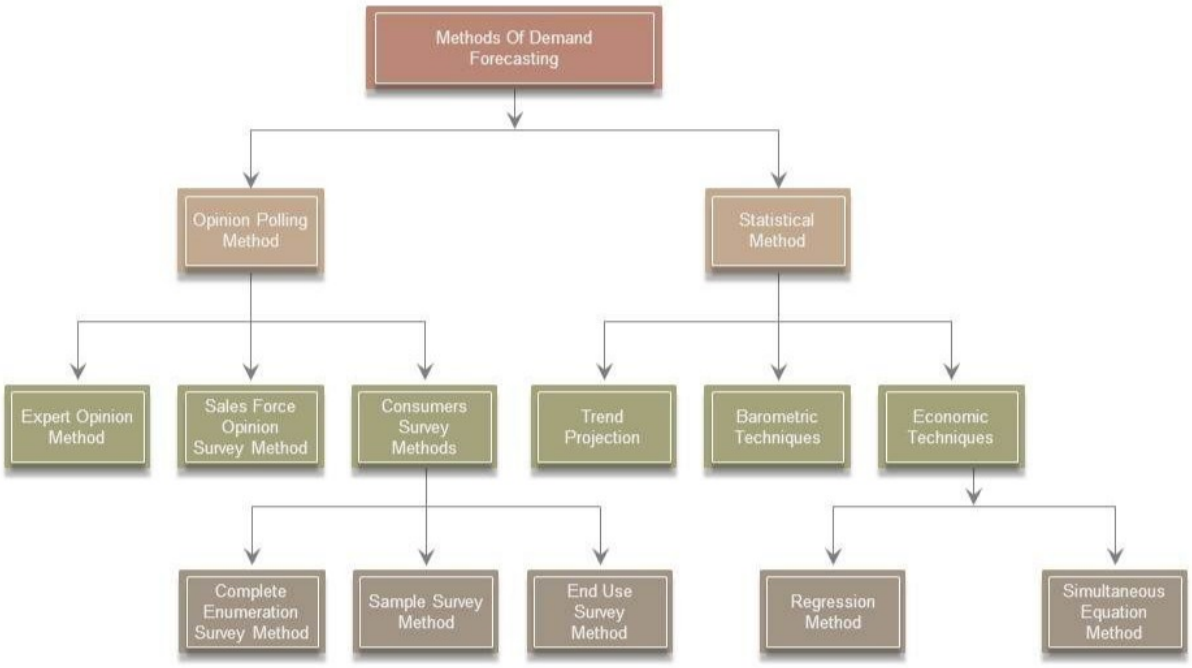
3.Java jdk.

4 EXPERIMENTAL INVESTIGATIONS

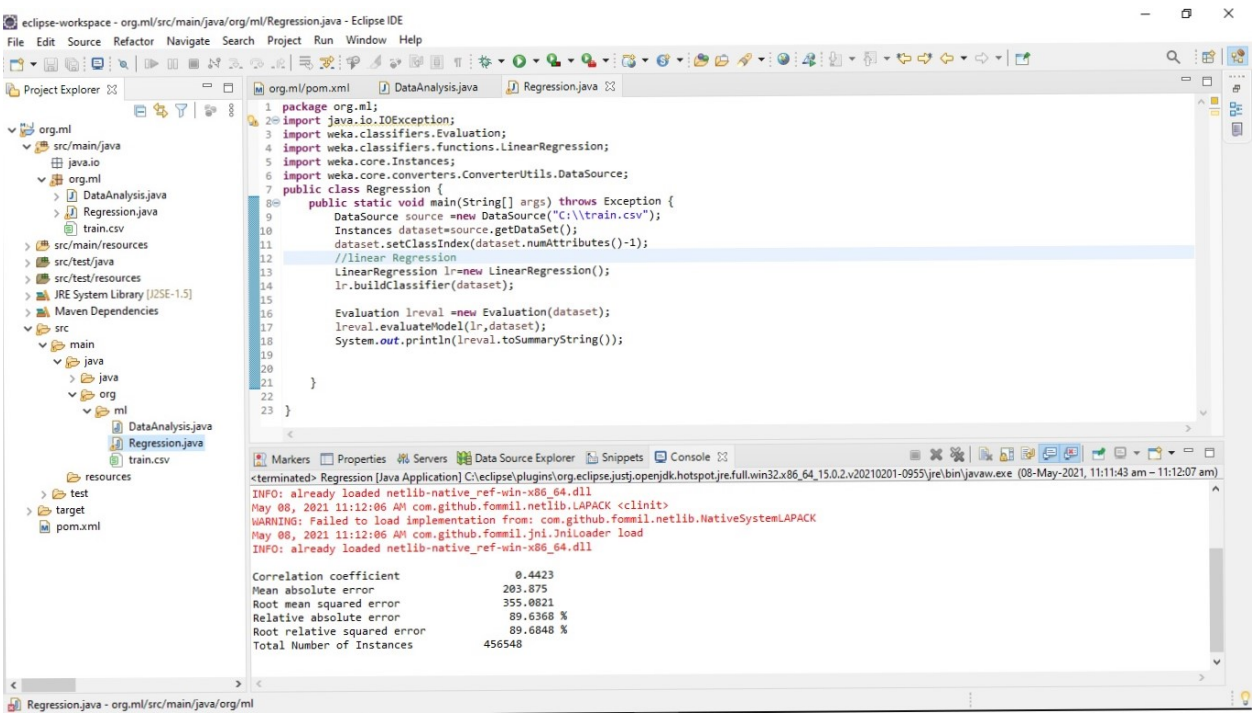
Analysis or investigations made while doing model is the data set given was completely error free.

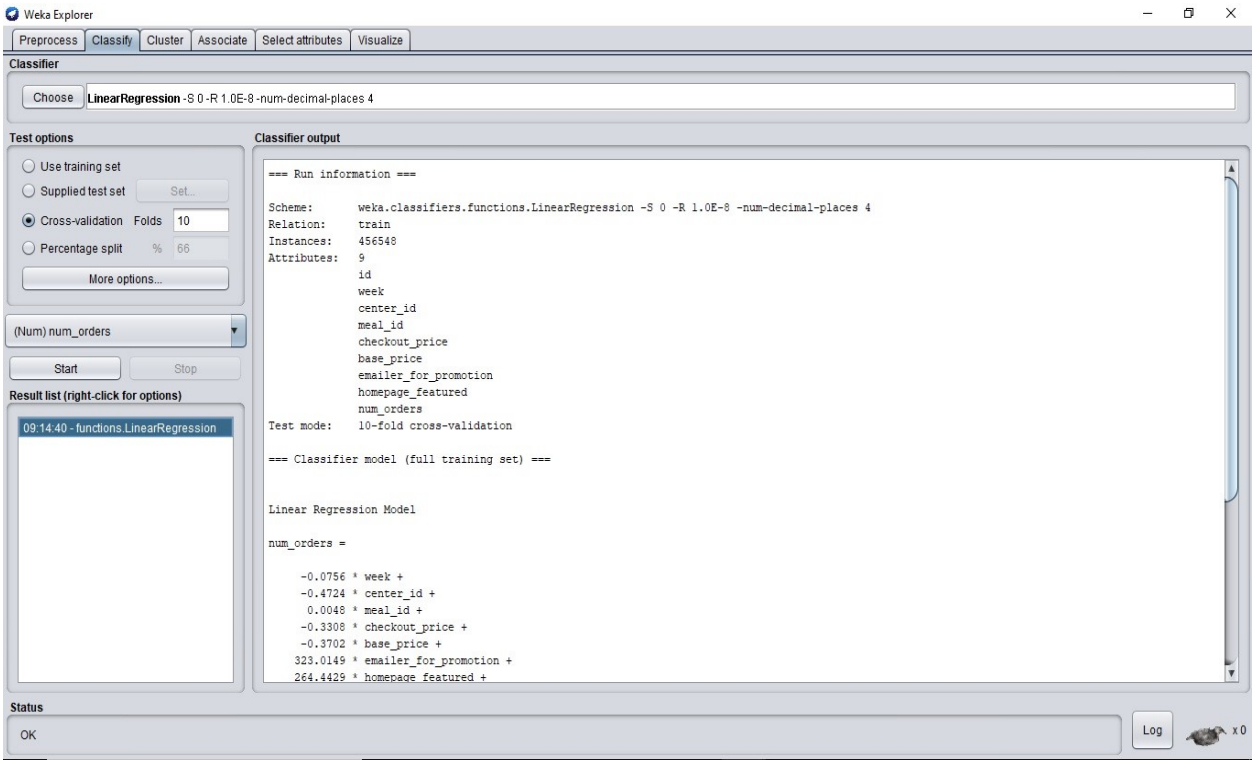
5 FLOWCHART

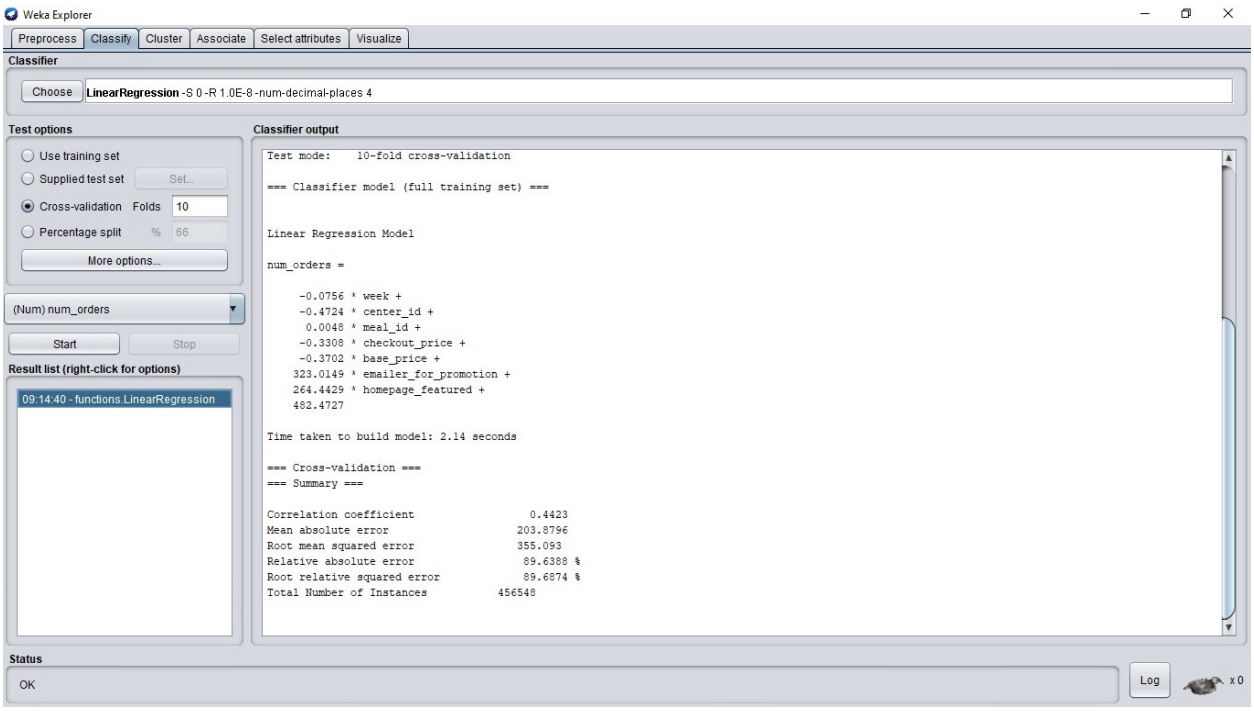
Diagram showing the control flow of the solution

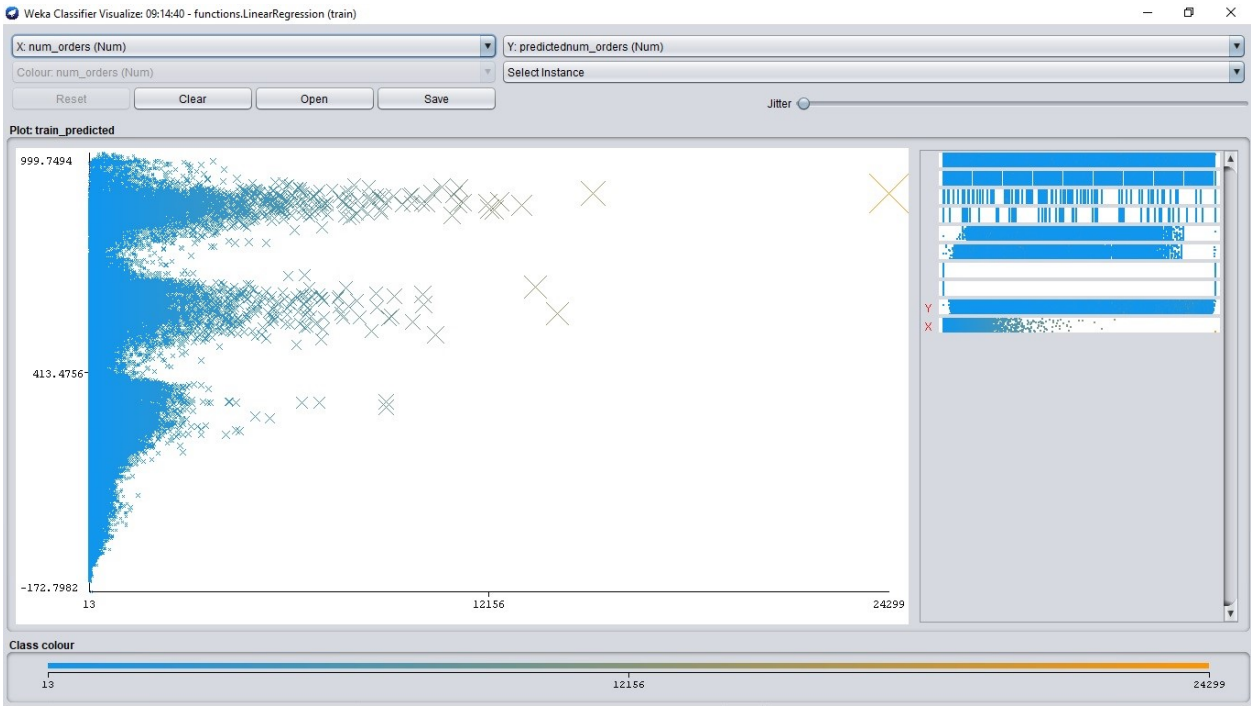


6 RESULT


eclipse-workspace - org.ml/src/main/java/org/ml/Regression.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
org.ml
src/main/java
java.io
org.ml
DataAnalysis.java
Regression.java
train.csv
src/main/resources
src/test/java
src/test/resources
JRE System Library [J2SE-1.5]
Maven Dependencies
src
main
java
java
org
ml
DataAnalysis.java
Regression.java
train.csv
resources
test
target
pom.xml
org.ml/pom.xml
DataAnalysis.java
Regression.java
1 package org.ml;
2 import java.io.IOException;
3 import weka.classifiers.Evaluation;
4 import weka.classifiers.functions.LinearRegression;
5 import weka.core.Instances;
6 import weka.core.converters.ConverterUtils.DataSource;
7 public class Regression {
8 public static void main(String[] args) throws Exception {
9 DataSource source =new DataSource("C:\\train.csv");
10 Instances dataset=source.getDataSet();
11 dataset.setClassIndex(dataset.numAttributes()-1);
12 //linear Regression
13 LinearRegression lr=new LinearRegression();
14 lr.buildClassifier(dataset);
15
16 Evaluation lreval =new Evaluation(dataset);
17 lreval.evaluateModel(lr,dataset);
18 System.out.println(lreval.toSummaryString());
19
20
21 }
22
23 }
Markers Properties Servers Data Source Explorer Snippets Console
<terminated> Regression [Java Application] C:\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_15.0.2.v20210201-0955\jre\bin\javaw.exe (08-May-2021, 11:11:43 am – 11:12:07 am)
INFO: already loaded netlib-native_ref-win-x86_64.dll
May 08, 2021 11:12:06 AM com.github.fommil.netlib.LAPACK <clinit>
WARNING: Failed to load implementation from: com.github.fommil.netlib.NativeSystemLAPACK
May 08, 2021 11:12:06 AM com.github.fommil.jni.JniLoader load
INFO: already loaded netlib-native_ref-win-x86_64.dll
Correlation coefficient 0.4423
Mean absolute error 203.875
Root mean squared error 355.0821
Relative absolute error 89.6368 %
Root relative squared error 89.6848 %
Total Number of Instances 456548
Regression.java - org.ml/src/main/java/org/ml


Weka Explorer
Preprocess Classify Cluster Associate Select attributes Visualize
Classifier
Choose LinearRegression -S 0 -R 1.0E-8 -num-decimal-places 4
Test options
Use training set
Supplied test set Set...
Cross-validation Folds 10
Percentage split % 66
More options...
(Num) num_orders
Start Stop
Result list (right-click for options)
09:14:40 - functions.LinearRegression
Classifier output
=== Run information ===
Scheme: weka.classifiers.functions.LinearRegression -S 0 -R 1.0E-8 -num-decimal-places 4
Relation: train
Instances: 456548
Attributes: 9
id
week
center_id
meal_id
checkout_price
base_price
emailer_for_promotion
homepage_featured
num_orders
Test mode: 10-fold cross-validation
=== Classifier model (full training set) ===
Linear Regression Model
num_orders =
-0.0756 * week +
-0.4724 * center_id +
0.0048 * meal_id +
-0.3308 * checkout_price +
-0.3702 * base_price +
323.0149 * emailer_for_promotion +
264.4429 * homepage_featured +
Status
OK
Log
x 0


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
LinearRegression -S 0 -R 1.0E-8 -num-decimal-places 4
Test options
Use training set
Supplied test set
Set...
Cross-validation Folds 10
Percentage split % 66
More options...
(Num) num_orders
Start
Stop
Result list (right-click for options)
09:14:40 - functions.LinearRegression
Classifier output
Test mode: 10-fold cross-validation
=== Classifier model (full training set) ===
Linear Regression Model
num_orders =
-0.0756 * week +
-0.4724 * center_id +
0.0048 * meal_id +
-0.3308 * checkout_price +
-0.3702 * base_price +
323.0149 * emailer_for_promotion +
264.4429 * homepage_featured +
482.4727
Time taken to build model: 2.14 seconds
=== Cross-validation ===
=== Summary ===
Correlation coefficient 0.4423
Mean absolute error 203.8796
Root mean squared error 355.093
Relative absolute error 89.6388 %
Root relative squared error 89.6874 %
Total Number of Instances 456548
Status
OK
Log
x 0


Weka Classifier Visualize: 09:14:40 - functions.LinearRegression (train)
X: num_orders (Num)
Y: predictednum_orders (Num)
Colour: num_orders (Num)
Select Instance
Reset
Clear
Open
Save
Jitter
Plot: train_predicted
999.7494
413.4756
-172.7982
13
12156
24299
Y
X
Class colour
13
12156
24299

7 ADVANTAGES & DISADVANTAGES

Advantages:

1.forecast the raw materials for next 10 weeks.

2.Easy use of model

Disadvantages:

1.Needs model operating knowledge. So workers with less knowledge cannot use it.

8 APPLICATIONS

This model can widely use in RETAIL sector of food delivering systems.

9 CONCLUSION

To conclude this, the model which is built is perfectly used in retail sector. It gives proper solution.

10 FUTURE SCOPE

Enhancements that can be made in the future are we can add new filters by adding instances and testcases.

11 BIBILOGRAPHY

References:

1.https://github.com/pradeepthiduggaraju.

2.https://edumine.wordpress.com/2014/08/15/data-pre-processing-with-weka-part-1/

3.https://projects.eclipse.org/proposals/eclipse-advanced-visualization-project

APPENDIX A. Source Code

```
package org.ml;
import java.io.IOException;
import weka.classifiers.Evaluation;
import weka.classifiers.functions.LinearRegression;
import weka.core.Instances;
import weka.core.converters.ConverterUtils.DataSource;
public class Regression {
	public static void main(String[] args) throws Exception {
		DataSource source =new DataSource("C:\\train.csv");
		Instances dataset=source.getDataSet();
		dataset.setClassIndex(dataset.numAttributes()-1);
		//linear Regression
		LinearRegression lr=new LinearRegression();
		lr.buildClassifier(dataset);

		Evaluation lreval =new Evaluation(dataset);
	   lreval.evaluateModel(lr,dataset);
		System.out.println(lreval.toSummaryString());


	}

}
```