Dublin Business School
excellence through learning

Module: B9DA103 Data Mining

CA_TWO

TITLE: SENTIMENT ANALYSIS AND PREDICTION ON YOUTUBE VIDEOS

Module Guide:
Terri Hoare

Student Name:
Abhishek Beard [10504842]
Rohan Nagalkar[10403259]
Shantanu Jadhav[10393294]

Content

Introduction

Abstract- This project mainly based on a prediction system that takes video likes dislikes comments and category as input and predict the views of the video. We have also done sentiment analysis based on the comments available on every video. We have used several data classifying techniques to build a trained classifier and to sort our test data. Using this class to Variety magazine, "To determine the year's top-trending videos, YouTube uses a combination of factors including measuring users interactions (number of views, shares, comments and likes). Note that they're not the most-viewed videos overall for the calendar year". Top performers on the YouTube trending list are music videos (such as the famously virile "Gangnam Style"), celebrity and reality TV performances, and the random dude-with-a-camera viral videos that YouTube is well-known for.

Possible uses for this dataset could be:

- Sentiment analysis 9+ in a variety of forms
- Categorising YouTube videos based on their comments and statistics.
- Training ML algorithms like GLM to predict their views.
- Analysing what factors affect how popular a YouTube video will be.

The main goal of the project is to predict the views of the video posted recently so that it will be helpful for the video poster whether it is a company or a person just uploading a video.

We have done sentiment analysis using Python and R as our csv size was too much we were facing issues to load csv file using Rstudio so we had to use Python for reading csv and getting the unique videos from a csv.

There will be enormous benefits of this application like which video can be used to post an advertisement.

Also if we take an example of Gangnam style video there is a history of a video that because of Nine quintillions of views the youtube application got crashed

On 1 December, Google posted a statement saying: "We never thought a video would be watched in numbers greater than a 32-bit integer... but that was before we met Psy." Google, which owns YouTube, later told website **The Verge** that engineers "saw this coming a couple of months ago and updated our systems to prepare for it".

YouTube now uses a 64-bit integer for its video counter, which means videos have a maximum viewer count of 9.22 quintillion.

So this eventually will be used not only for generating revenue but also will be helpful in preventing the application from generating errors.

We have used **CRISP-DM** methodology for our application

This methodology integrates the Security, Trust, Efficiency and Freedom infringement (S-T-E-Fi) dimensions in its evaluation stage. This is a highly innovative approach, as certification has, to date, primarily focused on the assessment of technical requirements for security systems.

Business Understanding

Objectives:

Following are the objectives from the datasets given

- Analyse the people’s comments and get the sentiments analysis from the user's comments.
- Getting the statistical view of the youtube video to analyze the factors
- Using some machine learning algorithms like GLMs to predict the views of the video.
- Analysing what the other factors are that affect the youtube trending video

Evaluating the circumstance:

Nowadays Youtube is a very trending platform, which has a very huge user and it increases day by days. Even now televisions and the other social media platform is replaced by the youtube hence for the industries and the new youtube creators need some statistical analysis of the trending videos to either promote the product or to generate the great content.
The big problem in front of them is how to get good content?
What is the people opinion about the different category?
Which video can be viral and why?

The aim of data mining:

Base on the Trending video data, i.e. its views, likes, dislikes and comment it is beneficial to analyse some statistical parameter which can tell the secrets behind the achievements of the popularity of the videos. The analysis will help the commercial industries about their product reviews earlier, from their promo video also for youtube creators the trending topics and category who are seeking to get more subscribers. This analysis will help the commercial industries to get the weakness of their product even to improve some of the business strategies and the weakness of the product. For the youtube creators, what are parameters factors are affecting the parameter to trend the video

Data

Data collection part is done through Kaggle

The dataset is about the trending video from the USA and Great Britain

This dataset has the 4 CSV file which is one the data of the two countries data which is US and Canada, and the remain 2 are comments of the two countries. The data have the following parameters:

video_id
title
channel_title
category_id
tags
views
likes
dislikes
comment_total
thumbnail_link
date

Data Investigation:
The videos have around Great Britain database have 30581 videos, and also there are 44 categories and 6680 channels In the United, State database have the 40567 videos and 44 categories and 7000 channels data

Data Attribute:
The data contain mixed data which include the unstructured and not English data; hence there are must need of the data cleaning process.

Data Preparation
As there were almost no missing values(can be counted on fingers) present in data we deleted the rows which were having no values or empty values. Removing outlier is also a part of data preparation, there were few outliers in the dataset like a video has around lakhs of views but not a single like or dislike or a single comment for that video. So we removed such data from our data set.

The aim of data mining:

Base on the Trending video data, i.e. its views, likes, dislikes and comment it is beneficial to analyse some statistical parameter which can tell the secrets behind the achievements of the popularity of the videos. The analysis will help the commercial industries about their product reviews earlier, from their promo video also for youtube creators the trending topics and category who are seeking to get more subscribers. This analysis will help the commercial industries to get the weakness of their product even to improve some of the business strategies and the weakness of the product. For the youtube creators, what are parameters factors are affecting the parameter to trend the video

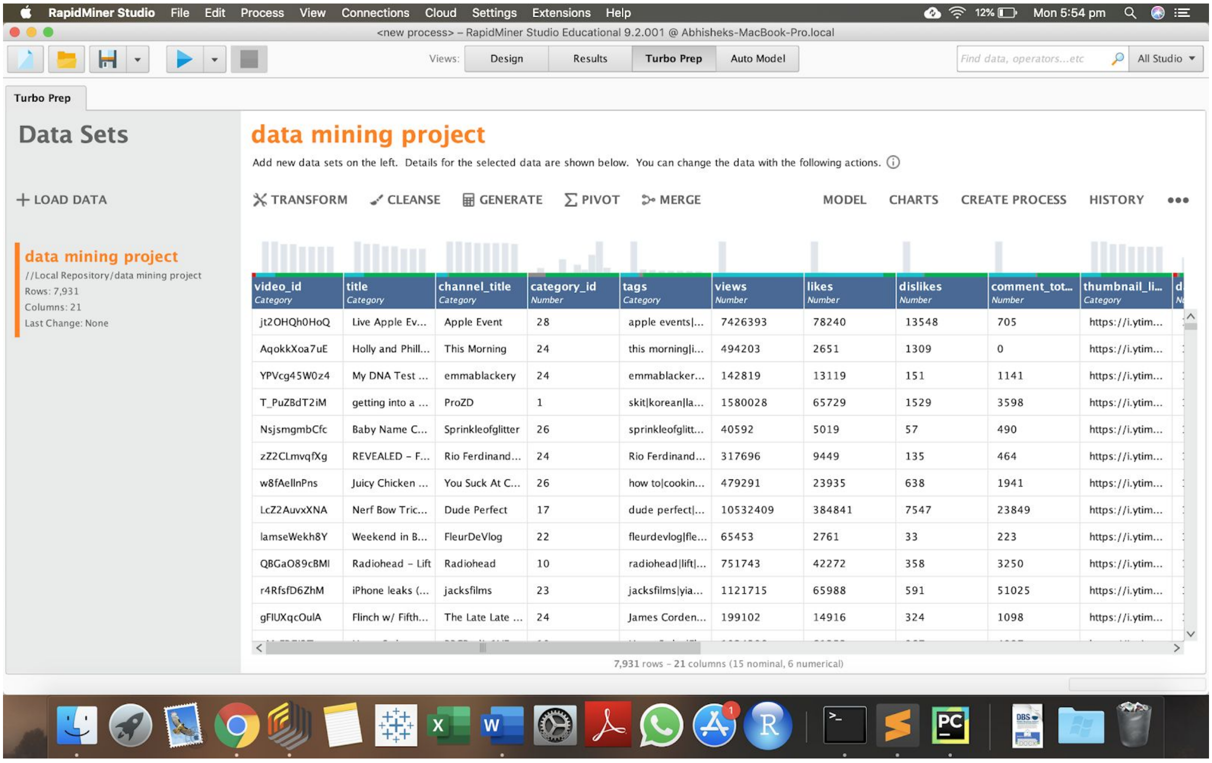
Modeling

How to choose a model in rapid-miner using Auto model.

With the following two factors we can choose an algorithm:

1. Performance: the closer the correlation to 1
2. Rum time: Less the time required for processing better the model.

The actual representation of data is shown below:



data mining project

//Local Repository/data mining project
Rows: 7,931
Columns: 21
Last Change: None

data mining project

Add new data sets on the left. Details for the selected data are shown below. You can change the data with the following actions.

video_id Category	title Category	channel_title Category	category_id Number	tags Category	views Number	likes Number	dislikes Number	comment_tot... Number	thumbnail_li... Category
jt2OHQh0HoQ	Live Apple Ev...	Apple Event	28	apple events|...	7426393	78240	13548	705	https://i.ytim...
AqokkXoa7uE	Holly and Phill...	This Morning	24	this morning|i...	494203	2651	1309	0	https://i.ytim...
YPVcg45W0z4	My DNA Test ...	emmablackery	24	emmablacker...	142819	13119	151	1141	https://i.ytim...
T_PuZBdT2iM	getting into a ...	ProZD	1	skit|korean|la...	1580028	65729	1529	3598	https://i.ytim...
NsjsmgmbCfc	Baby Name C...	Sprinkleofglitter	26	sprinkleofglitt...	40592	5019	57	490	https://i.ytim...
zZ2CLmvqfXg	REVEALED – F...	Rio Ferdinand...	24	Rio Ferdinand...	317696	9449	135	464	https://i.ytim...
w8fAellnPns	Juicy Chicken ...	You Suck At C...	26	how to|cookin...	479291	23935	638	1941	https://i.ytim...
LcZ2AuvxXNA	Nerf Bow Tric...	Dude Perfect	17	dude perfect|...	10532409	384841	7547	23849	https://i.ytim...
lamseWekh8Y	Weekend in B...	FleurDeVlog	22	fleurdevlog|fle...	65453	2761	33	223	https://i.ytim...
QBGaO89cBMI	Radiohead – Lift	Radiohead	10	radiohead|lift|...	751743	42272	358	3250	https://i.ytim...
r4RfsfD6ZhM	iPhone leaks (...	jacksfilms	23	jacksfilms|yia...	1121715	65988	591	51025	https://i.ytim...
gFIUXqcOulA	Flinch w/ Fifth...	The Late Late ...	24	James Corden...	199102	14916	324	1098	https://i.ytim...

7,931 rows – 21 columns (15 nominal, 6 numerical)

For Sentiment analysis, we used python for getting unique video Ids and applied API from R to get the sentiments of each video.

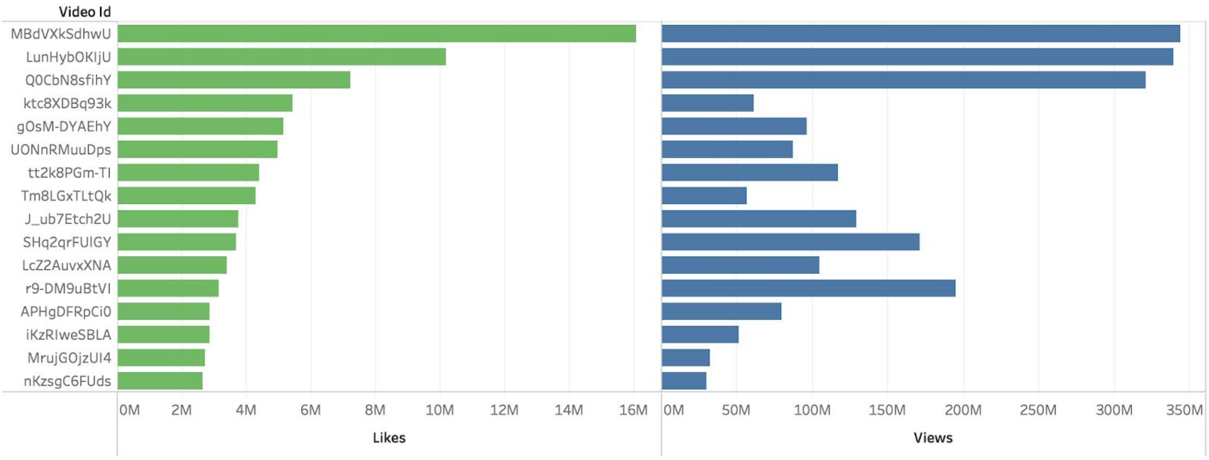
With these, we got 7 types of emotions and two sentiments including

Angry, anticipation, disgust, joy, sadness, surprise, positive and negative

Exploratory Data Analysis Visualizations :

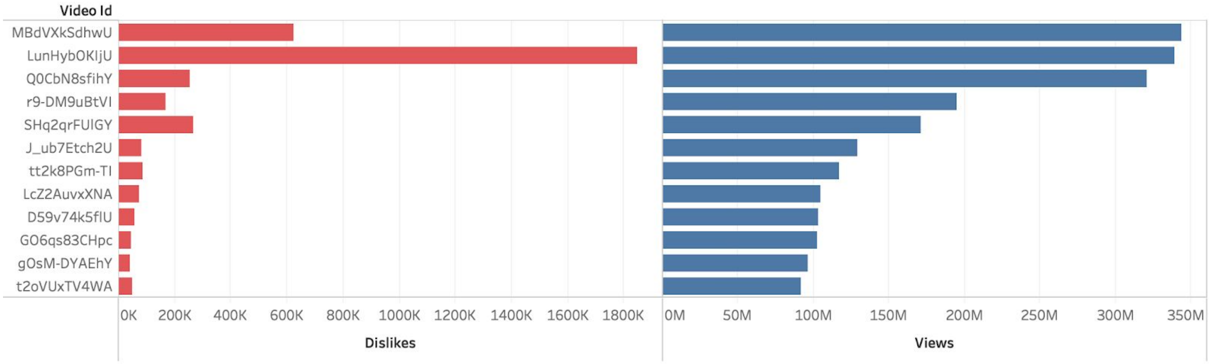
Bar chart of the Likes vs Views:

Total no of likes vs Views



Sum of Likes and sum of Views for each Video Id. The view is filtered on Video Id, which keeps 16 of 1,699 members.

Dislike videos vs Views

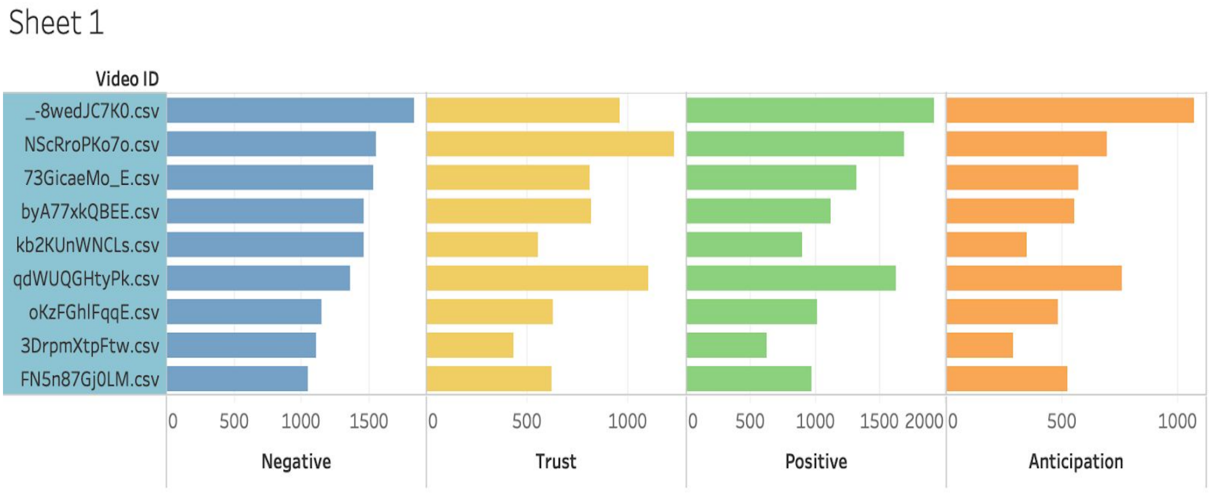


Sum of Dislikes and sum of Views for each Video Id. The view is filtered on Video Id, which keeps 12 of 1,699 members.

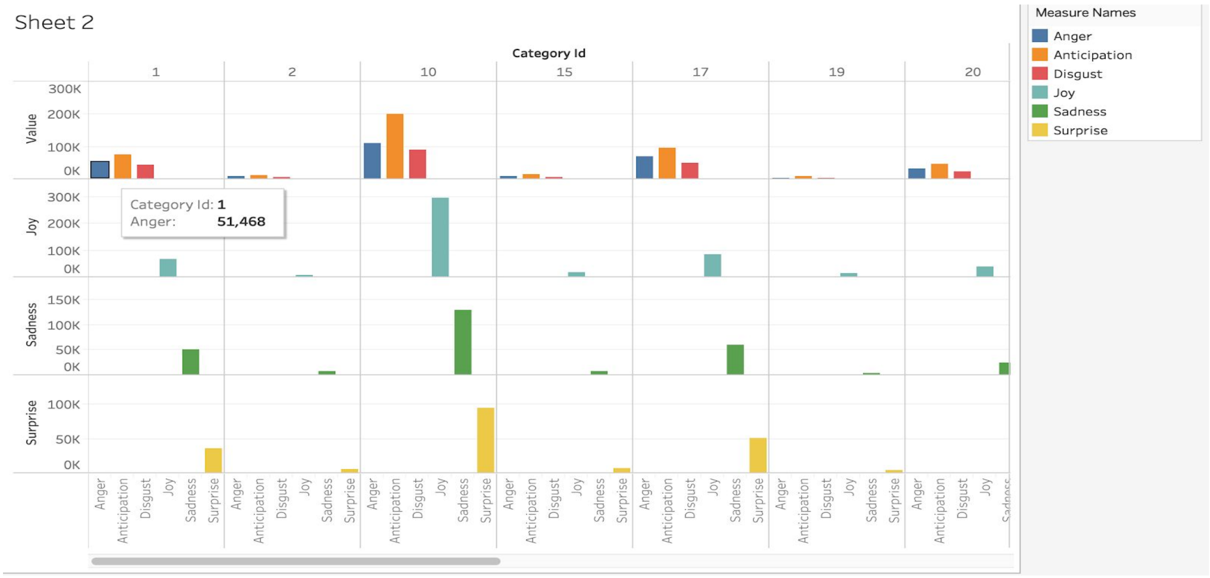
Sentiment Analysis

For Sentiment analysis, we used python for getting unique video Ids and applied API from R named get_nrc_sentiment from syuzhet package to get the sentiments of each video. With this package, we got 7 types of emotions and two sentiments including

Angry, anticipation, disgust, joy, sadness, surprise, positive and negative



Sum of Negative, sum of Trust, sum of Positive and sum of Anticipation for each Video ID. The view is filtered on Video ID, which keeps 9 of 1,654 members.



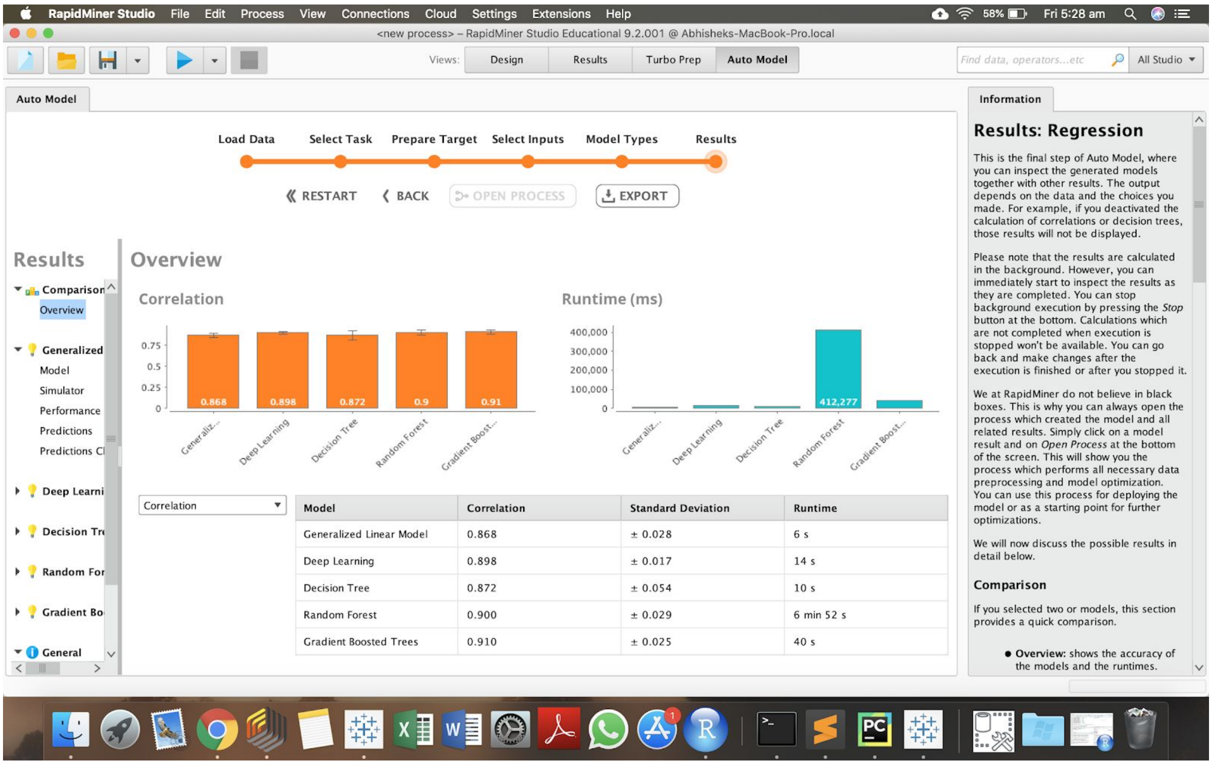
R code for sentiment analysis:

```
library('syuzhet')
library("stringi")
library("ggplot2")



getwd()
setwd('/Users/abhishekberad/Desktop/dm')
file_data = list.files('/Users/abhishekberad/Desktop/dm')

file_matrix = matrix(
   ncol=11,
   byrow = TRUE)
file_count = 1

for (video_file in file_data){
  video_id = read.csv(video_file)
  if (length(video_id$comment) < 10){
    next
  }
  video_id$comment <- tolower(video_id$comment)
  video_id$comment <- gsub("rt", "",video_id$comment)
  video_id$comment <- gsub("rt", "",video_id$comment)
  video_id$comment <- gsub("@\\w+", "",video_id$comment)
  video_id$comment <- gsub("@\\w+", "",video_id$comment)
  video_id$comment <- gsub("[[:punct:]]", "",video_id$comment)
  video_id$comment <- gsub("[[:punct:]]", "",video_id$comment)
  video_id$comment <- gsub("http\\w+", "",video_id$comment)
  video_id$comment <- gsub("http\\w+", "",video_id$comment)
  video_id$comment <- gsub("[ |\t]{2,}", "",video_id$comment)
  video_id$comment <- gsub("[ |\t]{2,}", "",video_id$comment)
  video_id$comment <- gsub("^ ", "",video_id$comment)
  video_id$comment <- gsub("^ ", "",video_id$comment)
  video_id$comment <- gsub(" $", "",video_id$comment)

  sentiment_video<-get_nrc_sentiment((video_id$comment))
  Sentimentscores_video<-data.frame(colSums(sentiment_video[,]))

Sentimentscores_video<-cbind("sentiment"=rownames(Sentimentscores_video)
,Sentimentscores_video)
  video = c(video_file)
  count = 1
  if (nrow(Sentimentscores_video)<1){

    next
  }
  for (count in 1 : nrow(Sentimentscores_video)){
    video = c(video,
Sentimentscores_video$colSums.sentiment_video.....[count])

  }
  file_matrix2 =  matrix(
       data = video, # the data elements
       nrow=1,             # number of rows
       ncol=11,             # number of columns
       byrow = TRUE)
  file_matrix = rbind(file_matrix, video)

}
file_matrix
colnames(data_comment)[1] <- "Video_ID"
colnames(data_comment)[2] <- "Anger"
colnames(data_comment)[3] <- "anticipation"
colnames(data_comment)[4] <- "disgust"
colnames(data_comment)[5] <- "fear"
colnames(data_comment)[6] <- "joy"
colnames(data_comment)[7] <- "sadness"
colnames(data_comment)[8] <- "surprise"
colnames(data_comment)[9] <- "trust"
colnames(data_comment)[10] <- "negative"
colnames(data_comment)[11] <- "positive"

data_comment2  <- gsub("[.csv ]","" , data_comment ,ignore.case = TRUE)

write.csv(data_comment,"sentiment_youtube.csv")

write.csv(data_comment, file = "sentiment_youtube.csv",row.names=FALSE)
data_comment <- as.data.frame(file_matrix)
data_comment = data_comment[-1,]
nrow(file_matrix)
```

Modeling and Evaluation

Auto Model:

1) Regression:



Model	Correlation	Standard Deviation	Runtime
Generalized Linear Model	0.868	± 0.028	6 s
Deep Learning	0.898	± 0.017	14 s
Decision Tree	0.872	± 0.054	10 s
Random Forest	0.900	± 0.029	6 min 52 s
Gradient Boosted Trees	0.910	± 0.025	40 s

Results: Regression

This is the final step of Auto Model, where you can inspect the generated models together with other results. The output depends on the data and the choices you made. For example, if you deactivated the calculation of correlations or decision trees, those results will not be displayed.

Please note that the results are calculated in the background. However, you can immediately start to inspect the results as they are completed. You can stop background execution by pressing the *Stop* button at the bottom. Calculations which are not completed when execution is stopped won't be available. You can go back and make changes after the execution is finished or after you stopped it.

We at RapidMiner do not believe in black boxes. This is why you can always open the process which created the model and all related results. Simply click on a model result and on *Open Process* at the bottom of the screen. This will show you the process which performs all necessary data preprocessing and model optimization. You can use this process for deploying the model or as a starting point for further optimizations.

We will now discuss the possible results in detail below.

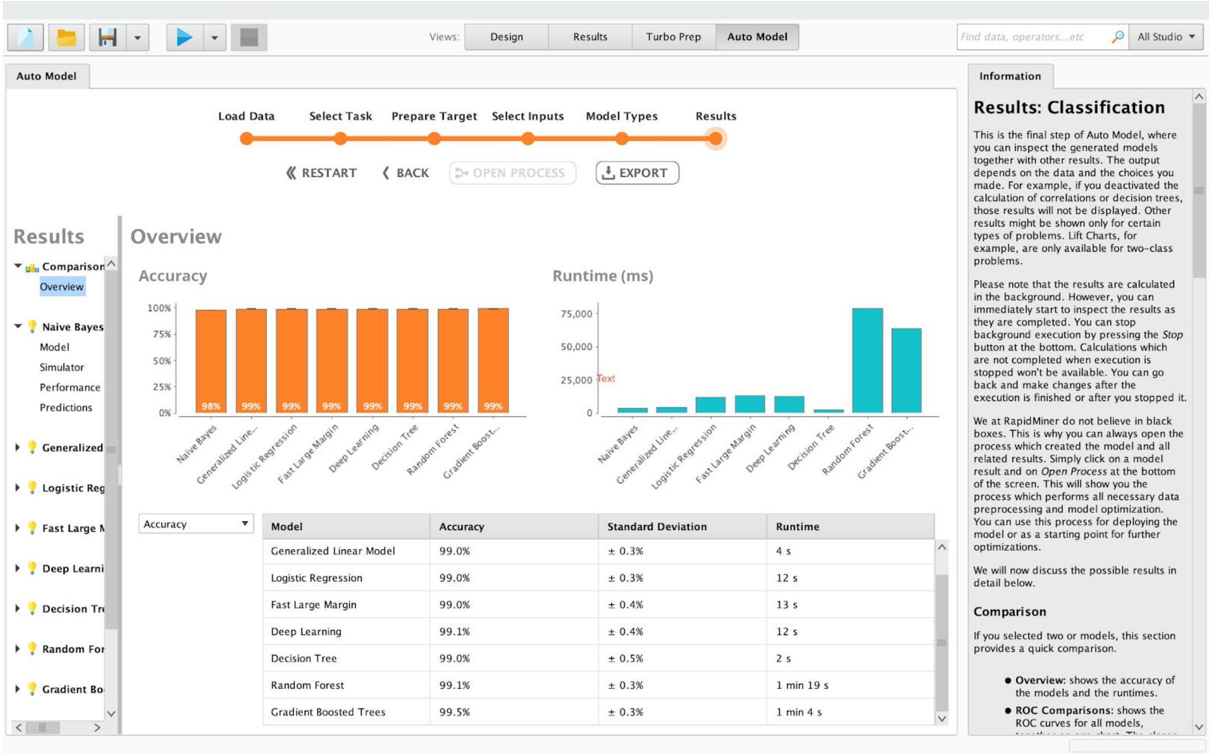
Comparison

If you selected two or models, this section provides a quick comparison.

- **Overview:** shows the accuracy of the models and the runtimes.

After applying the auto model we got better correlation for Gradient boost but the runtime was very low for GLM, so we proceeded with the GLM algorithm for our views prediction.

2) Classification over regression:



Model	Accuracy	Standard Deviation	Runtime
Generalized Linear Model	99.0%	± 0.3%	4 s
Logistic Regression	99.0%	± 0.3%	12 s
Fast Large Margin	99.0%	± 0.4%	13 s
Deep Learning	99.1%	± 0.4%	12 s
Decision Tree	99.0%	± 0.5%	2 s
Random Forest	99.1%	± 0.3%	1 min 19 s
Gradient Boosted Trees	99.5%	± 0.3%	1 min 4 s

Information

Results: Classification

This is the final step of Auto Model, where you can inspect the generated models together with other results. The output depends on the data and the choices you made. For example, if you deactivated the calculation of correlations or decision trees, those results will not be displayed. Other results might be shown only for certain types of problems. Lift Charts, for example, are only available for two-class problems.

Please note that the results are calculated in the background. However, you can immediately start to inspect the results as they are completed. You can stop background execution by pressing the *Stop* button at the bottom. Calculations which are not completed when execution is stopped won't be available. You can go back and make changes after the execution is finished or after you stopped it.

We at RapidMiner do not believe in black boxes. This is why you can always open the process which created the model and all related results. Simply click on a model result and on *Open Process* at the bottom of the screen. This will show you the process which performs all necessary data preprocessing and model optimization. You can use this process for deploying the model or as a starting point for further optimizations.

We will now discuss the possible results in detail below.

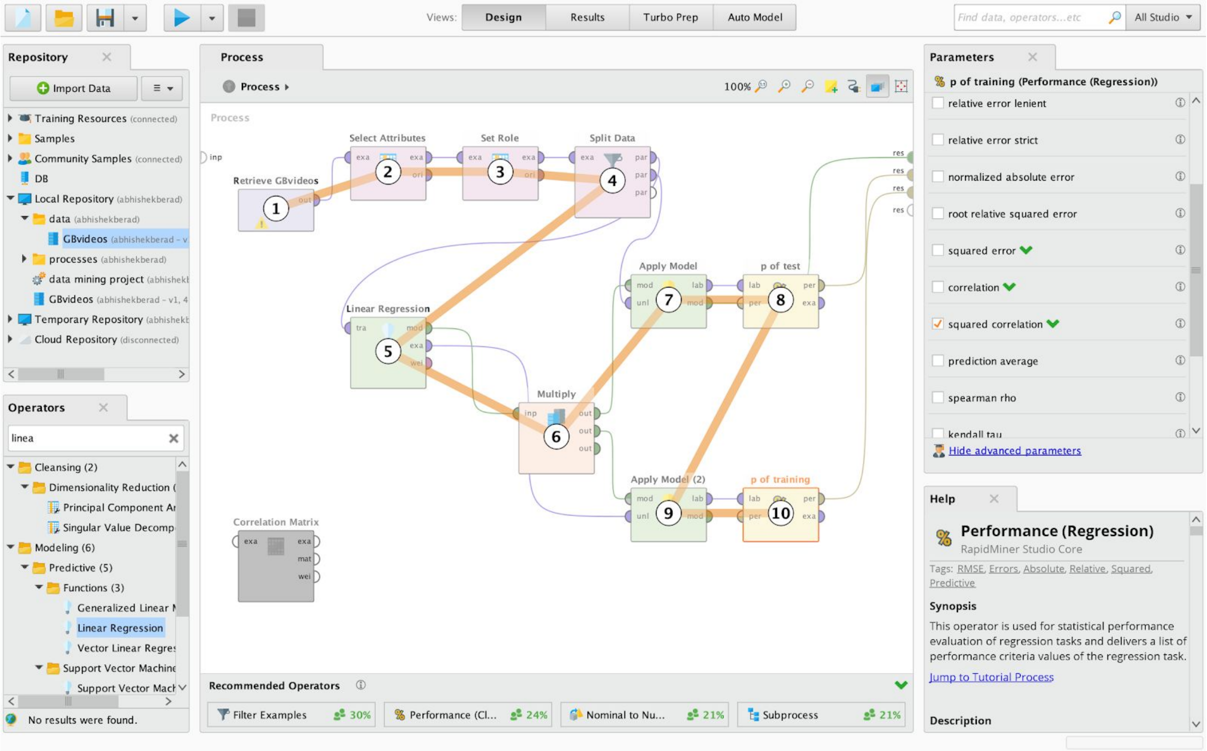
Comparison

If you selected two or models, this section provides a quick comparison.

- **Overview:** shows the accuracy of the models and the runtimes.
- **ROC Comparisons:** shows the ROC curves for all models,

We were not getting an accurate prediction when we applied GLM but when we applied classification on our data we got a better prediction of the views.

Following is the process we got when we applied auto model:



Parameters

p of training (Performance (Regression))

- ☐ relative error lenient
- ☐ relative error strict
- ☐ normalized absolute error
- ☐ root relative squared error
- ☐ squared error ✓
- ☐ correlation ✓
- ☑ squared correlation ✓
- ☐ prediction average
- ☐ spearman rho
- ☐ kendall tau

Hide advanced parameters

Help

Performance (Regression)

RapidMiner Studio Core

Tags: RMSE, Errors, Absolute, Relative, Squared, Predictive

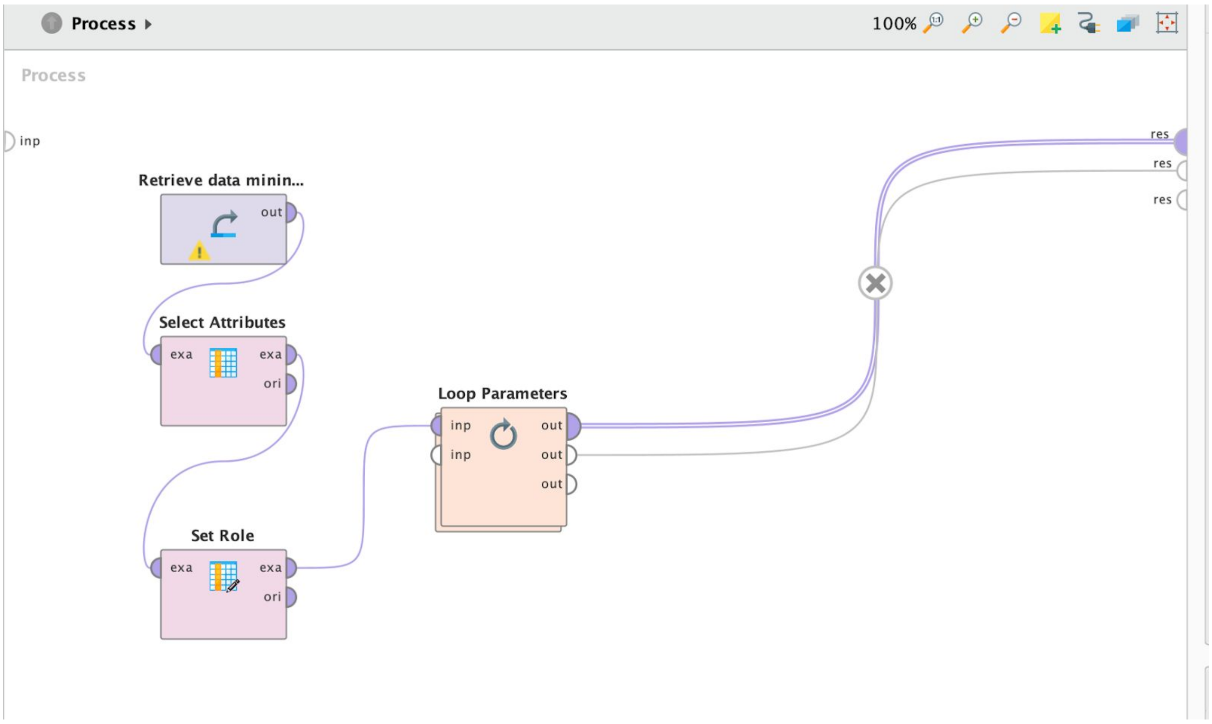
Synopsis

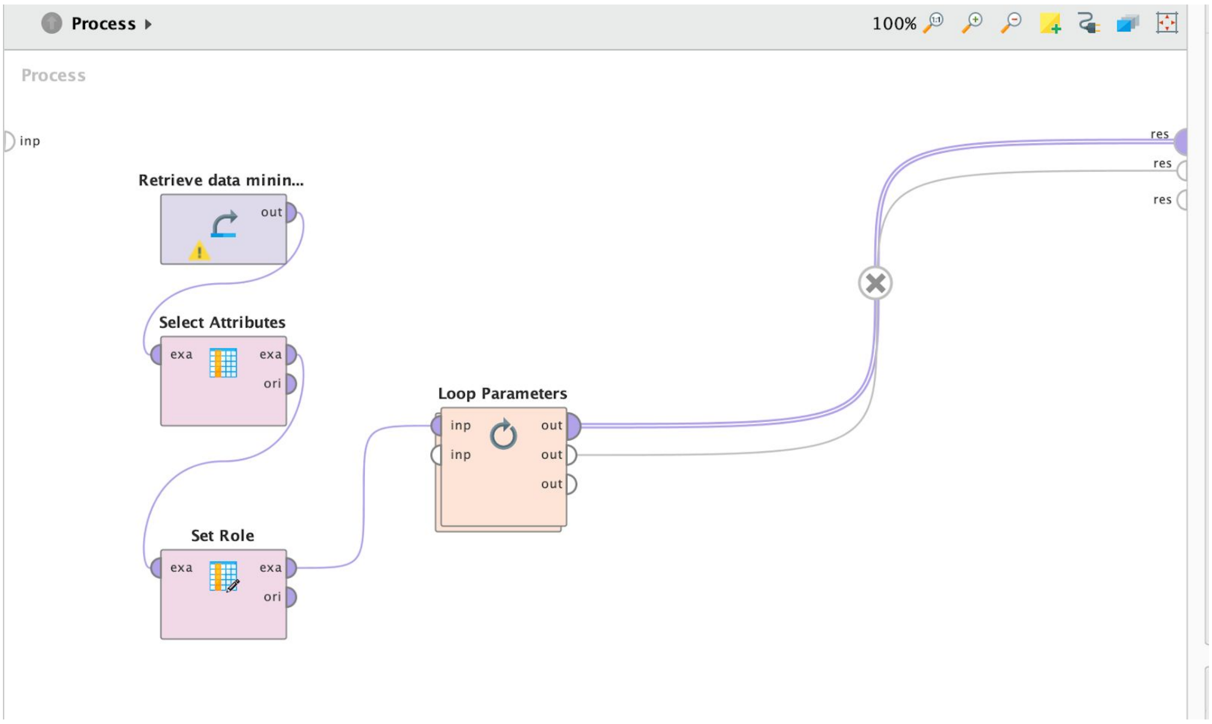
This operator is used for statistical performance evaluation of regression tasks and delivers a list of performance criteria values of the regression task.

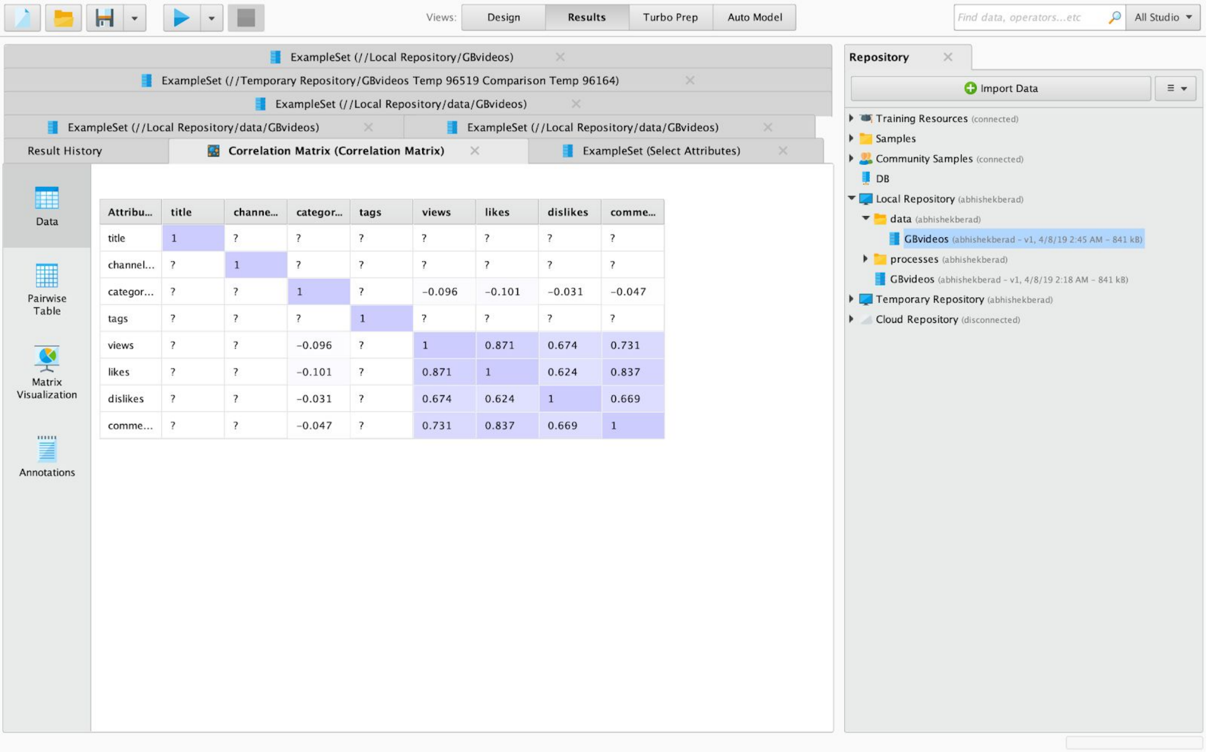
Jump to Tutorial Process

Description

Following is the process we did in Rapidminer:

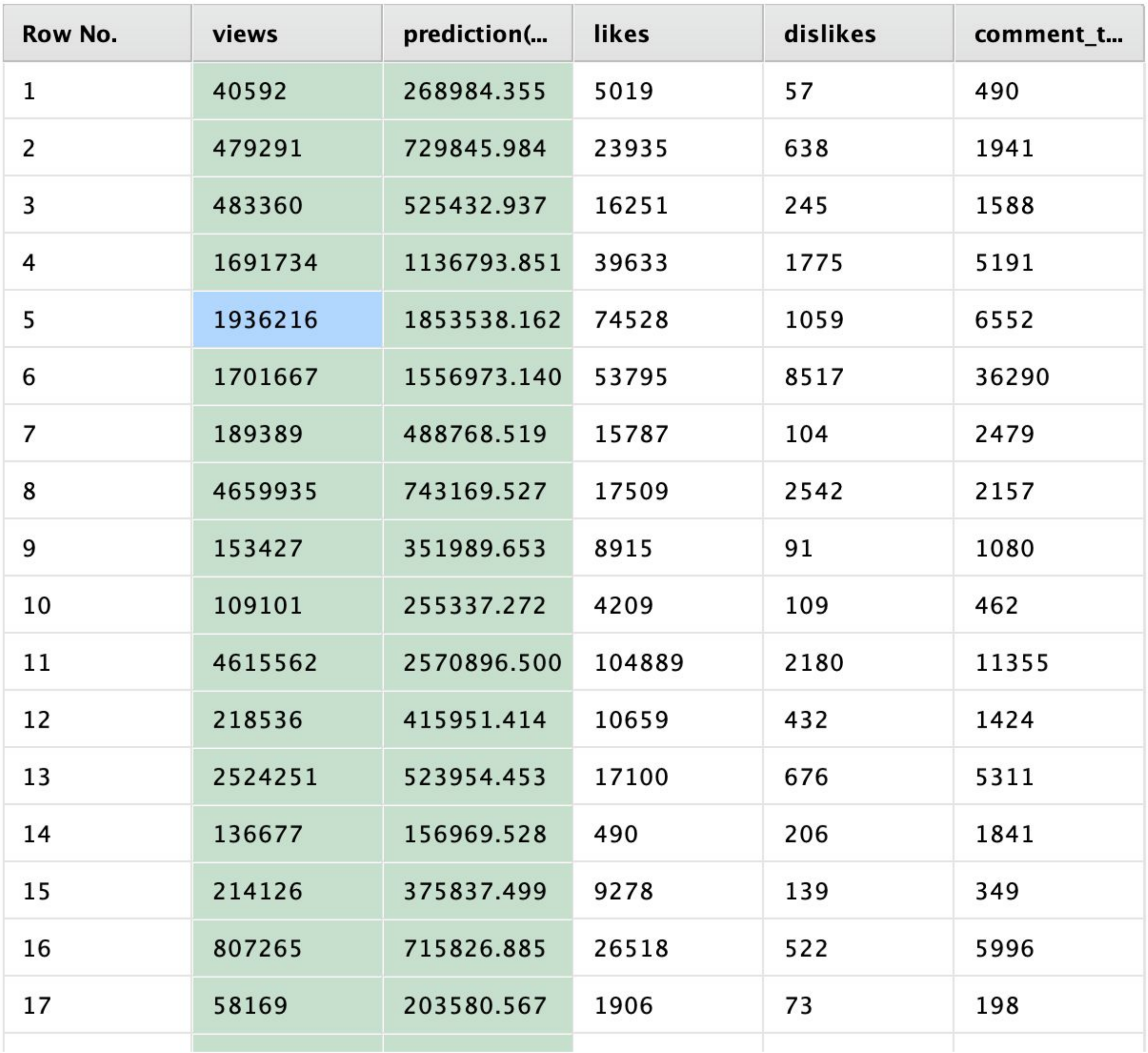



Process
100%
Process
inp
Retrieve data minin...
out
Select Attributes
exa
exa
ori
Set Role
exa
exa
ori
Loop Parameters
inp
inp
out
out
out
res
res
res


Views:
Design
Results
Turbo Prep
Auto Model
Find data, operators...etc
All Studio
ExampleSet (//Local Repository/GBvideos)
ExampleSet (//Temporary Repository/GBvideos Temp 96519 Comparison Temp 96164)
ExampleSet (//Local Repository/data/GBvideos)
ExampleSet (//Local Repository/data/GBvideos)
ExampleSet (//Local Repository/data/GBvideos)
Result History
Correlation Matrix (Correlation Matrix)
ExampleSet (Select Attributes)
Data
Pairwise Table
Matrix Visualization
Annotations
Repository
Import Data
Training Resources (connected)
Samples
Community Samples (connected)
DB
Local Repository (abhishekberad)
data (abhishekberad)
GBvideos (abhishekberad - v1, 4/8/19 2:45 AM – 841 kB)
processes (abhishekberad)
GBvideos (abhishekberad - v1, 4/8/19 2:18 AM – 841 kB)
Temporary Repository (abhishekberad)
Cloud Repository (disconnected)

Attribu...	title	channe...	categor...	tags	views	likes	dislikes	comme...
title	1	?	?	?	?	?	?	?
channel...	?	1	?	?	?	?	?	?
categor...	?	?	1	?	-0.096	-0.101	-0.031	-0.047
tags	?	?	?	1	?	?	?	?
views	?	?	-0.096	?	1	0.871	0.674	0.731
likes	?	?	-0.101	?	0.871	1	0.624	0.837
dislikes	?	?	-0.031	?	0.674	0.624	1	0.669
comme...	?	?	-0.047	?	0.731	0.837	0.669	1

From the above diagram, we can find the correlation that correlation of the likes,Dislikes and comment count is more as compared to the the other features



Row No.	views	prediction(...	likes	dislikes	comment_t...
1	40592	268984.355	5019	57	490
2	479291	729845.984	23935	638	1941
3	483360	525432.937	16251	245	1588
4	1691734	1136793.851	39633	1775	5191
5	1936216	1853538.162	74528	1059	6552
6	1701667	1556973.140	53795	8517	36290
7	189389	488768.519	15787	104	2479
8	4659935	743169.527	17509	2542	2157
9	153427	351989.653	8915	91	1080
10	109101	255337.272	4209	109	462
11	4615562	2570896.500	104889	2180	11355
12	218536	415951.414	10659	432	1424
13	2524251	523954.453	17100	676	5311
14	136677	156969.528	490	206	1841
15	214126	375837.499	9278	139	349
16	807265	715826.885	26518	522	5996
17	58169	203580.567	1906	73	198

This is the final prediction we got after applying GLM without classification

Conclusion:

While working on this project we not only got to learn Rapidminer but also we got a chance to explore Databricks and Azure blob store.

We stored a CSV file on the Azure storage and we wrote a data bricks notebook which will fetch the CSV and apply GLM application and give the prediction based on the algorithm.