UNIVERSITY OF MARYLAND 18 56

COLLEGE OF
INFORMATION
STUDIES

Project Report

SENTIMENT ANALYSIS OF AMAZON FINE FOOD REVIEWS

By

Anushka Kher
Karishma Ghiya

INST 737

Prof. Patrice Seyed

TABLE OF CONTENTS

INTRODUCTION:

DATASET:

The Amazon Fine Food Reviews dataset consists of 568,454 food reviews Amazon users left up to October 2012. Source: https://www.kaggle.com/snap/amazon-fine-food-reviews

This dataset consists of a single CSV file, Reviews.csv, and a corresponding SQLite table named Reviews in database SQLite. The columns in the table are:

- **Id**
- **ProductId** - unique identifier for the product
- **UserId** - unique identifier for the user
- **ProfileName**
- **HelpfulnessNumerator** - number of users who found the review helpful
- **HelpfulnessDenominator** - number of users who indicated whether they found the review helpful
- **Score** - rating between 1 and 5
- **Time** - timestamp for the review
- **Summary** - brief summary of the review
- **Text** - text of the review

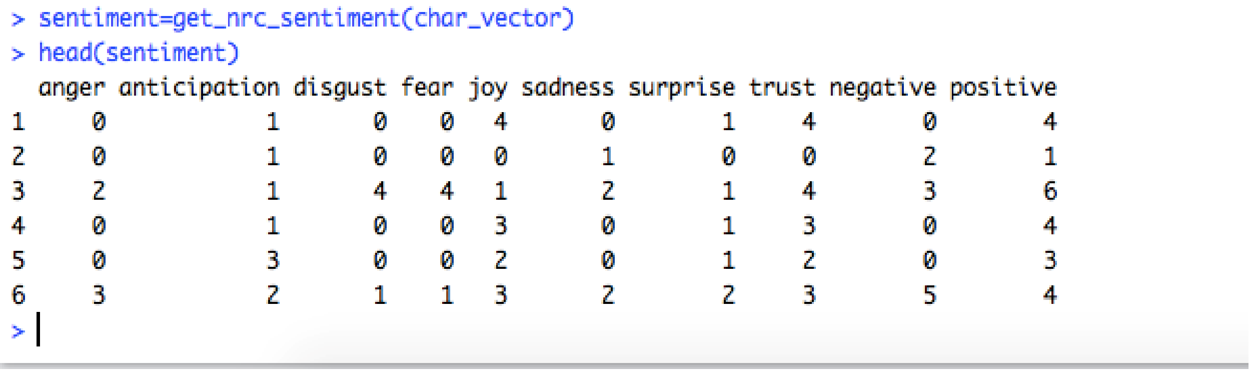
For the purpose of this project, we have conducted our analysis on 9000 reviews.

OBJECTIVE:

- Perform sentiment analysis on Amazon Fine Food Reviews Dataset
- Generate polarity for each review and corresponding summary
- Extracting the product title, description and product group using Amazon API
- Creating visualizations to find interesting patterns among dataset variables
- Building a prediction model to predict if a review is positive or negative
- Creating visualization to find which product group is more helpful

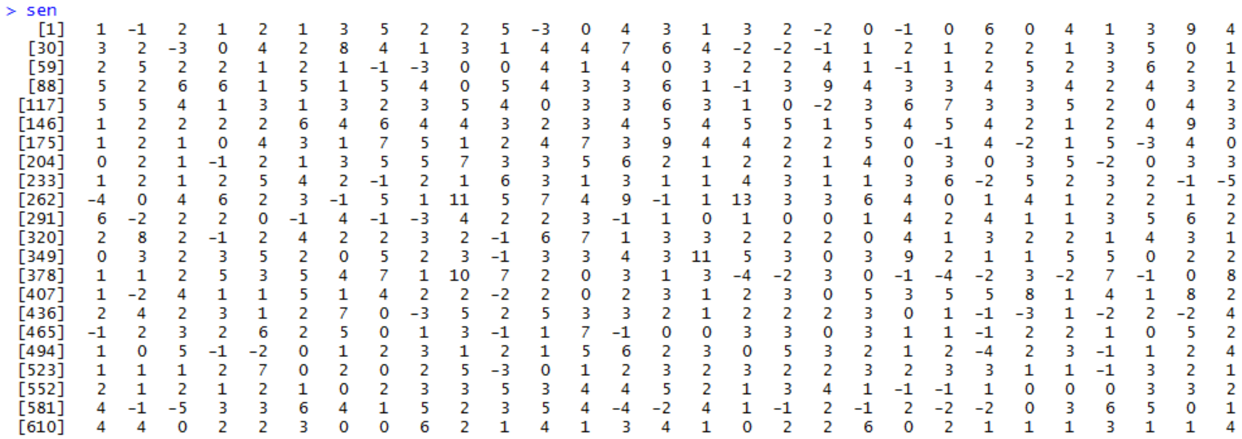
SYUZHET PACKAGE:

- Used CRAN syuzhet package to calculate sentiments associated with each review
- Function : get_nrc_sentiment()
  - Arguments:
    - character vector
  - Return value:
    - a data frame where each row represents a review and columns indicate value of each emotion type for the review



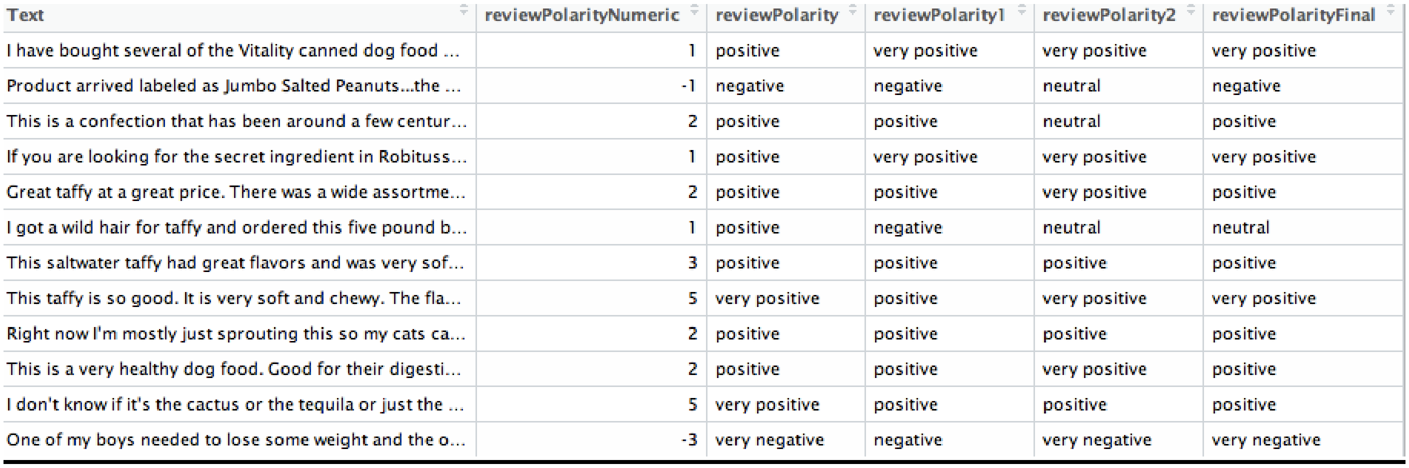
```
> sentiment=get_nrc_sentiment(char_vector)
> head(sentiment)
  anger anticipation disgust fear joy sadness surprise trust negative positive
1     0            1       0    0   4       0        1     4        0        4
2     0            1       0    0   0       1        0     0        2        1
3     2            1       4    4   1       2        1     4        3        6
4     0            1       0    0   3       0        1     3        0        4
5     0            3       0    0   2       0        1     2        0        3
6     3            2       1    1   3       2        2     3        5        4
> |
```

- Function : get_sentiment()
  - Arguments:
    1. character vector,
    2. a string indicating the sentiment method
  - Return value:
    1. a numeric vector of sentiment value for each review



```
> sen
  [1]   1  -1   2   1   2   1   3   5   2   2   5  -3   0   4   3   1   3   2  -2   0  -1   0   6   0   4   1   3   9   4
 [30]   3   2  -3   0   4   2   8   4   1   3   1   4   4   7   6   4  -2  -2  -1   1   2   1   2   2   1   3   5   0   1
 [59]   2   5   2   2   1   2   1  -1  -3   0   0   4   1   4   0   3   2   2   4   1  -1   1   2   5   2   3   6   2   1
 [88]   5   2   6   6   1   5   1   5   4   0   5   4   3   3   6   1  -1   3   9   4   3   3   4   3   4   2   4   3   2
[117]   5   5   4   1   3   1   3   2   3   5   4   0   3   3   6   3   1   0  -2   3   6   7   3   3   5   2   0   4   3
[146]   1   2   2   2   2   6   4   6   4   4   3   2   3   4   5   4   5   5   1   5   4   5   4   2   1   2   4   9   3
[175]   1   2   1   0   4   3   1   7   5   1   2   4   7   3   9   4   4   2   2   5   0  -1   4  -2   1   5  -3   4   0
[204]   0   2   1  -1   2   1   3   5   5   7   3   3   5   6   2   1   2   2   1   4   0   3   0   3   5  -2   0   3   3
[233]   1   2   1   2   5   4   2  -1   2   1   6   3   1   3   1   1   4   3   1   1   3   6  -2   5   2   3   2  -1  -5
[262]  -4   0   4   6   2   3  -1   5   1  11   5   7   4   9  -1   1  13   3   3   6   4   0   1   4   1   2   2   1   2
[291]   6  -2   2   2   0  -1   4  -1  -3   4   2   2   3  -1   1   0   1   0   0   1   4   2   4   1   1   3   5   6   2
[320]   2   8   2  -1   2   4   2   2   3   2  -1   6   7   1   3   3   2   2   2   0   4   1   3   2   2   1   4   3   1
[349]   0   3   2   3   5   2   0   5   2   3  -1   3   3   4   3  11   5   3   0   3   9   2   1   1   5   5   0   2   2
[378]   1   1   2   5   3   5   4   7   1  10   7   2   0   3   1   3  -4  -2   3   0  -1  -4  -2   3  -2   7  -1   0   8
[407]   1  -2   4   1   1   5   1   4   2   2  -2   2   0   2   3   1   2   3   0   5   3   5   5   8   1   4   1   8   2
[436]   2   4   2   3   1   2   7   0  -3   5   2   5   3   3   2   1   2   2   2   3   0   1  -1  -3   1  -2   2  -2   4
[465]  -1   2   3   2   6   2   5   0   1   3  -1   1   7  -1   0   0   3   3   0   3   1   1  -1   2   2   1   0   5   2
[494]   1   0   5  -1  -2   0   1   2   3   1   2   1   5   6   2   3   0   5   3   2   1   2  -4   2   3  -1   1   2   4
[523]   1   1   1   2   7   0   2   0   2   5  -3   0   1   2   3   2   3   2   2   3   2   3   3   1   1  -1   3   2   1
[552]   2   1   2   1   2   1   0   2   3   3   5   3   4   4   5   2   1   3   4   1  -1  -1   1   0   0   0   3   3   2
[581]   4  -1  -5   3   3   6   4   1   5   2   3   5   4  -4  -2   4   1  -1   2  -1   2  -2  -2   0   3   6   5   0   1
[610]   4   4   0   2   2   3   0   0   6   2   1   4   1   3   4   1   0   2   2   6   0   2   1   1   1   3   1   1   4
```

GENERATING POLARITY FOR REVIEWS:



Text	reviewPolarityNumeric	reviewPolarity	reviewPolarity1	reviewPolarity2	reviewPolarityFinal
I have bought several of the Vitality canned dog food ...	1	positive	very positive	very positive	very positive
Product arrived labeled as Jumbo Salted Peanuts...the ...	-1	negative	negative	neutral	negative
This is a confection that has been around a few centur...	2	positive	positive	neutral	positive
If you are looking for the secret ingredient in Robituss...	1	positive	very positive	very positive	very positive
Great taffy at a great price. There was a wide assortme...	2	positive	positive	very positive	positive
I got a wild hair for taffy and ordered this five pound b...	1	positive	negative	neutral	neutral
This saltwater taffy had great flavors and was very sof...	3	positive	positive	positive	positive
This taffy is so good. It is very soft and chewy. The fla...	5	very positive	positive	very positive	very positive
Right now I'm mostly just sprouting this so my cats ca...	2	positive	positive	positive	positive
This is a very healthy dog food. Good for their digesti...	2	positive	positive	very positive	positive
I don't know if it's the cactus or the tequila or just the ...	5	very positive	positive	positive	positive
One of my boys needed to lose some weight and the o...	-3	very negative	negative	very negative	very negative

- First prediction: using get_sentiment() value

  Column_name: reviewPolarityNumeric

  reviewPolarity

- Second Prediction: using positive and negative sentiments from get_nrc_sentiment() vector

  Column_name : reviewPolarity1

- Third prediction: using different sentiments from get_nrc_sentiment() vector

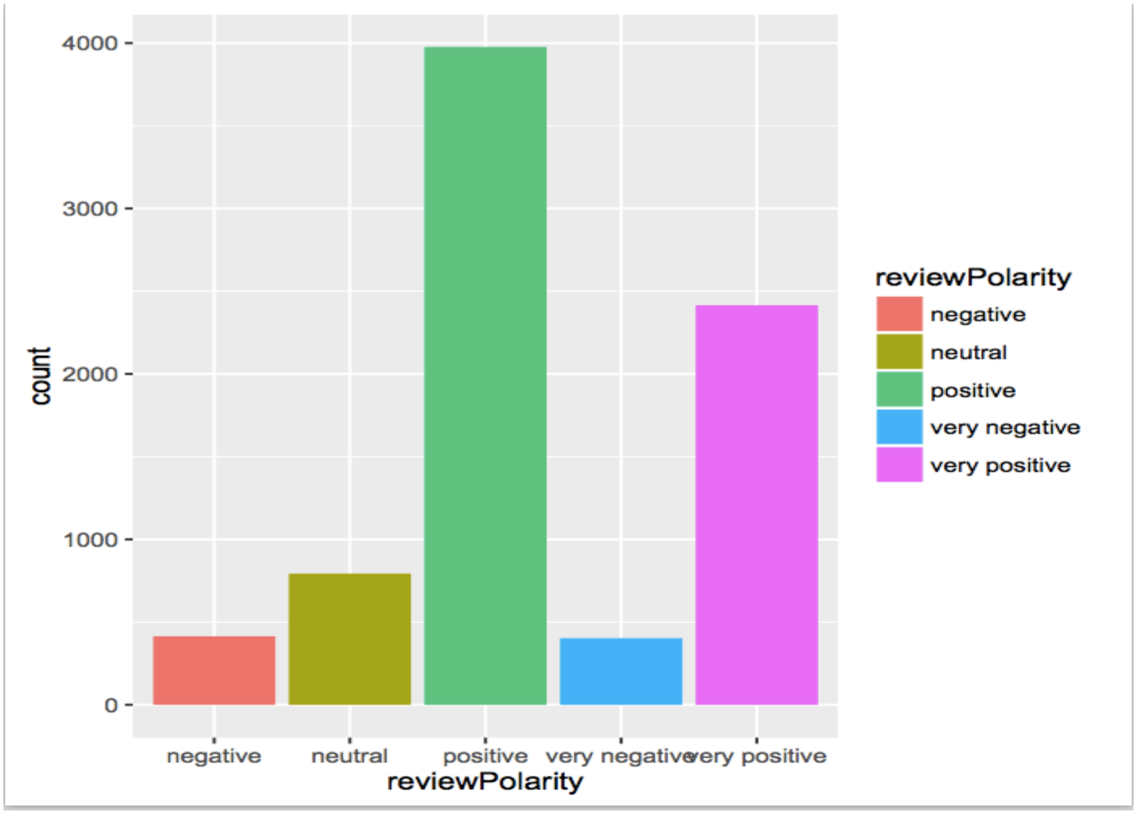
  Column_name: reviewPolarity2

- Final prediction: calculated based on previous three predictions

  Column_name: reviewPolarityFinal

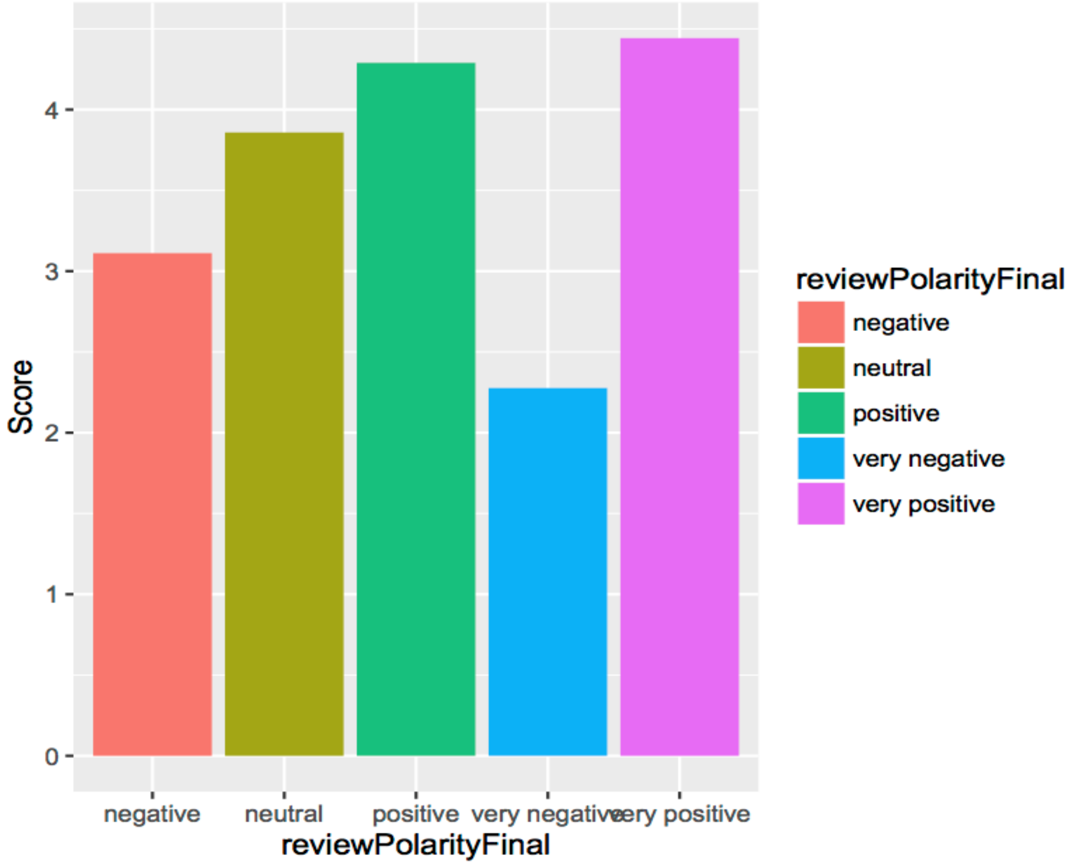
DATA EXPLORATION:

1. Count of Reviews based on their Polarity.



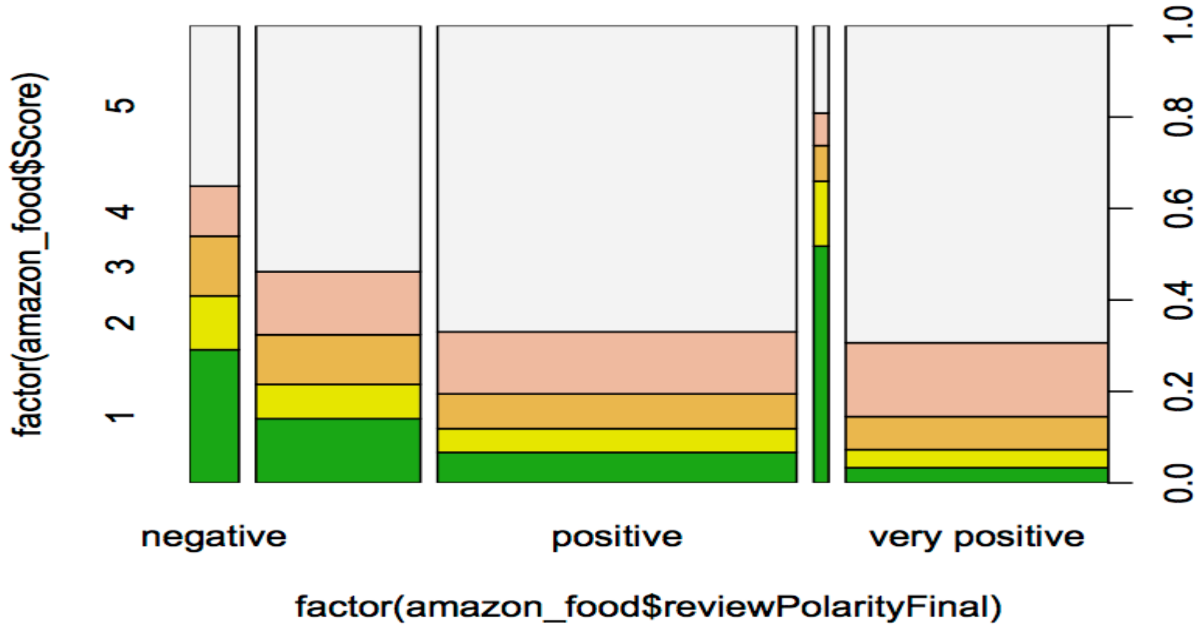
From this chart, we know that we have highest number of positive reviews in our dataset, followed by very positive reviews, followed by neutral, negative and very negative ones, with the latter being the least.

2. Review Polarity v/s Score Ratings:



From this plot, we know that very positive reviews have a higher score than those of positive reviews, followed by the neutral ones, negative and very negative, just as one would anticipate. This shows that the classification of reviews we conducted for their polarity into a scale of very negative to very positive is accurate.

3. Polarity v/s Score: Spineplot
   To get a detailed understanding of the distribution of polarity of reviews against their score, one should look at the spineplot below. This plot is well suited for understanding of relationships between two categorical variables. We can see the fractional breakdown of the categories of the score variable is shown for each category of the polarity variable. Stacked bars are drawn with vertical extent showing fraction of score given polarity and horizontal extent showing fraction of polarity. Thus the areas of tiles formed represent the frequencies in each cross-combination of score and polarity.

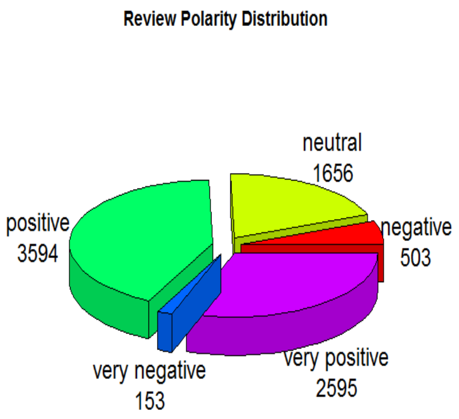


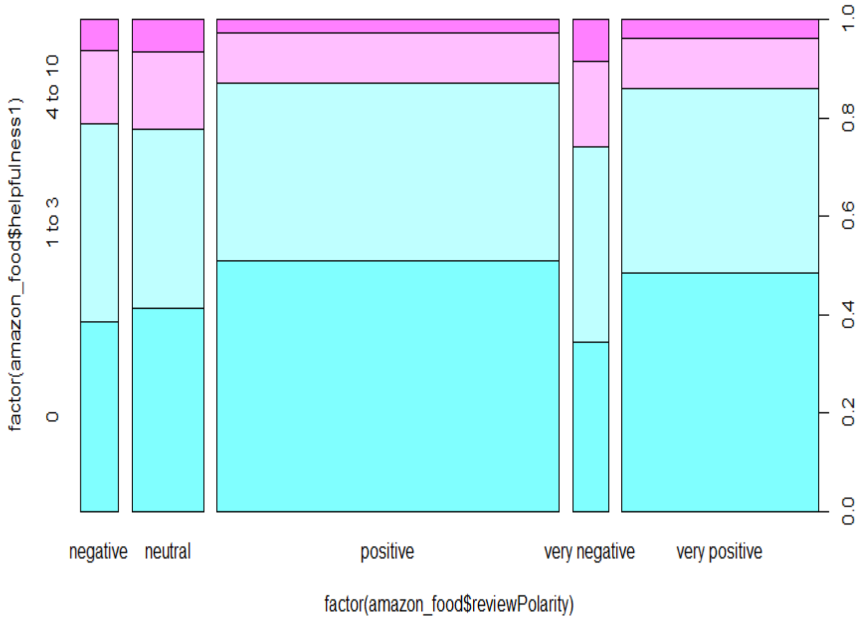
4. Polarity v/s Helpfulness:

   The continuous variable HelpfulnessDenominator is categorized into 4 categories in a variable named helpfulness1 :

   a. Reviews with HelpfulnessDenominator = 0
   b. Reviews with 1 <= HelpfulnessDenominator <=3
   c. Reviews with 4 <= HelpfulnessDenominator <= 10
   d. Reviews with HelpfulnessDenominator > 10

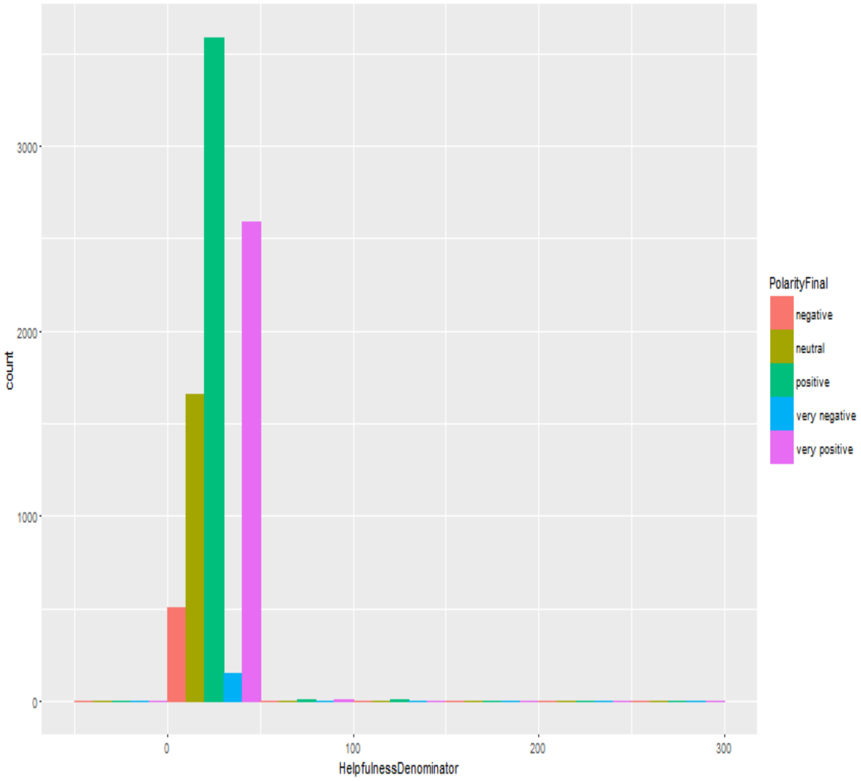
   Pie Chart for Review Polarity Distribution:



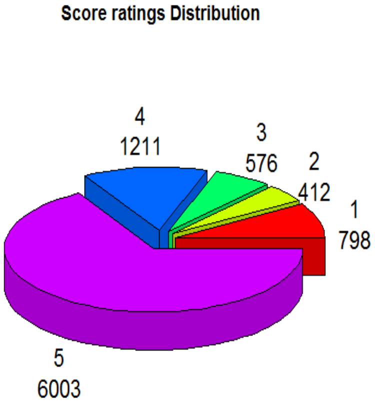


Spineplot showing the fractional breakdown of helpfulness v/s polarity of the reviews. The areas of tiles formed represent the frequencies in each cross-combination of helpfulness and polarity.

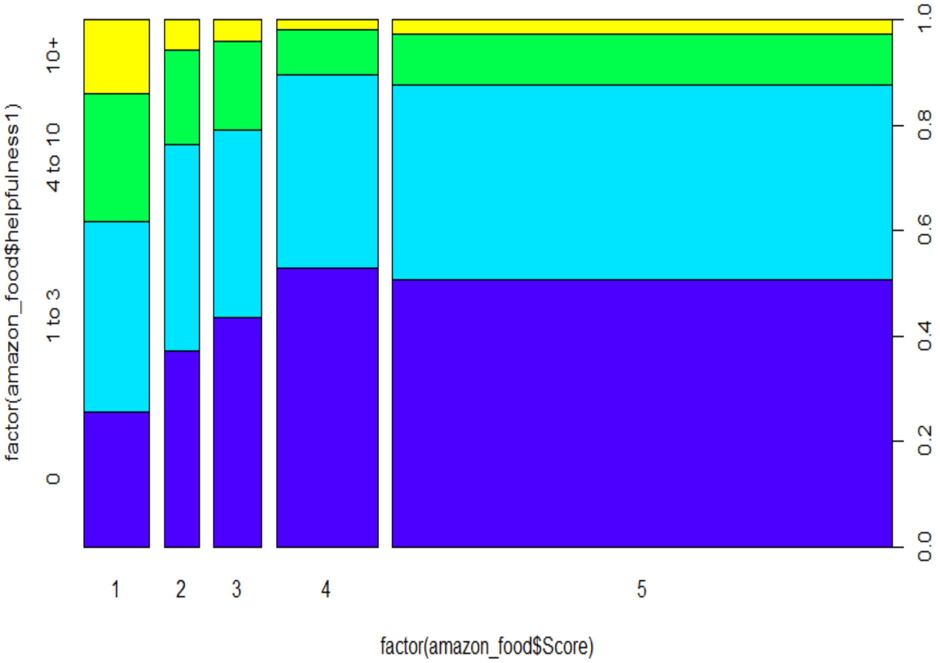
The bar chart below just gives the count of each reviews based on their polarity and helpfulness value.



5. Score v/s Helpfulness:
   Pie chart showing distribution of score ratings in the dataset.

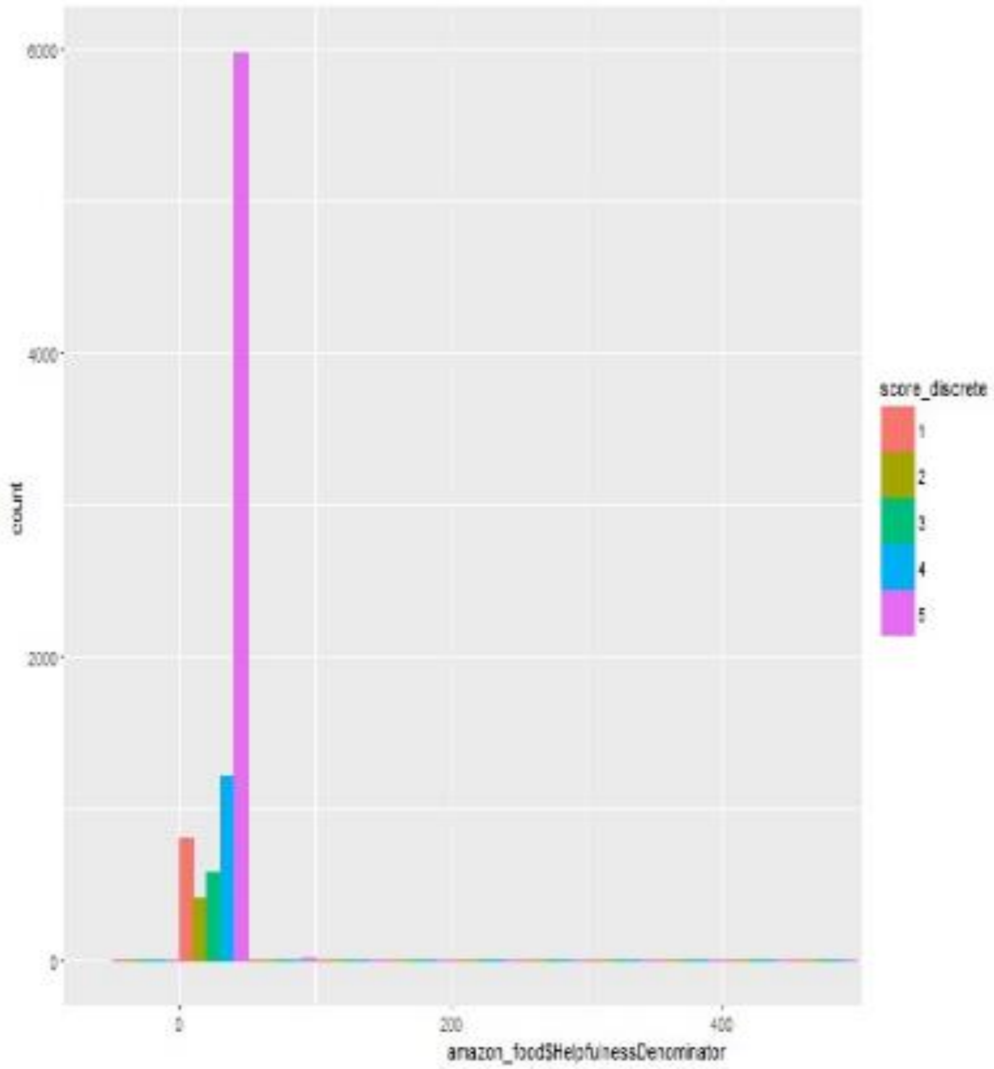


Spine plot of helpfulness v/s score:



Majority of the reviews are with a score of 5 and have a helpfulness values between 1 to 3. Least reviews are with a score of 2 and helpfulness values greater than 10.

Plot of Helpfulness Denominator v/s count, grouped according to the score values:



Scores with a ratings of 5 have the maximum count in the dataset and highest helpfulness value and the ones with ratings of 2 have the least count.

<u>AMAZON’S PRODUCT ADVERTISING API:</u>

Use of API to extract product title of the given product ID for all reviews in the dataset.

<u>Necessary parameters to use API:</u>

1. AWS Access Key ID
2. Amazon Secret Key
3. Associate Tag

Amazon Product Advertising API: ItemLookup() function

1. <u>Extract Product Titles</u>

Parameters passed: Product Id

Return Value: product title

Code/ filename: fetch_id.py

2.<u>Extract Product Groups</u>

Parameters passed: Product Id

Return Value: product group

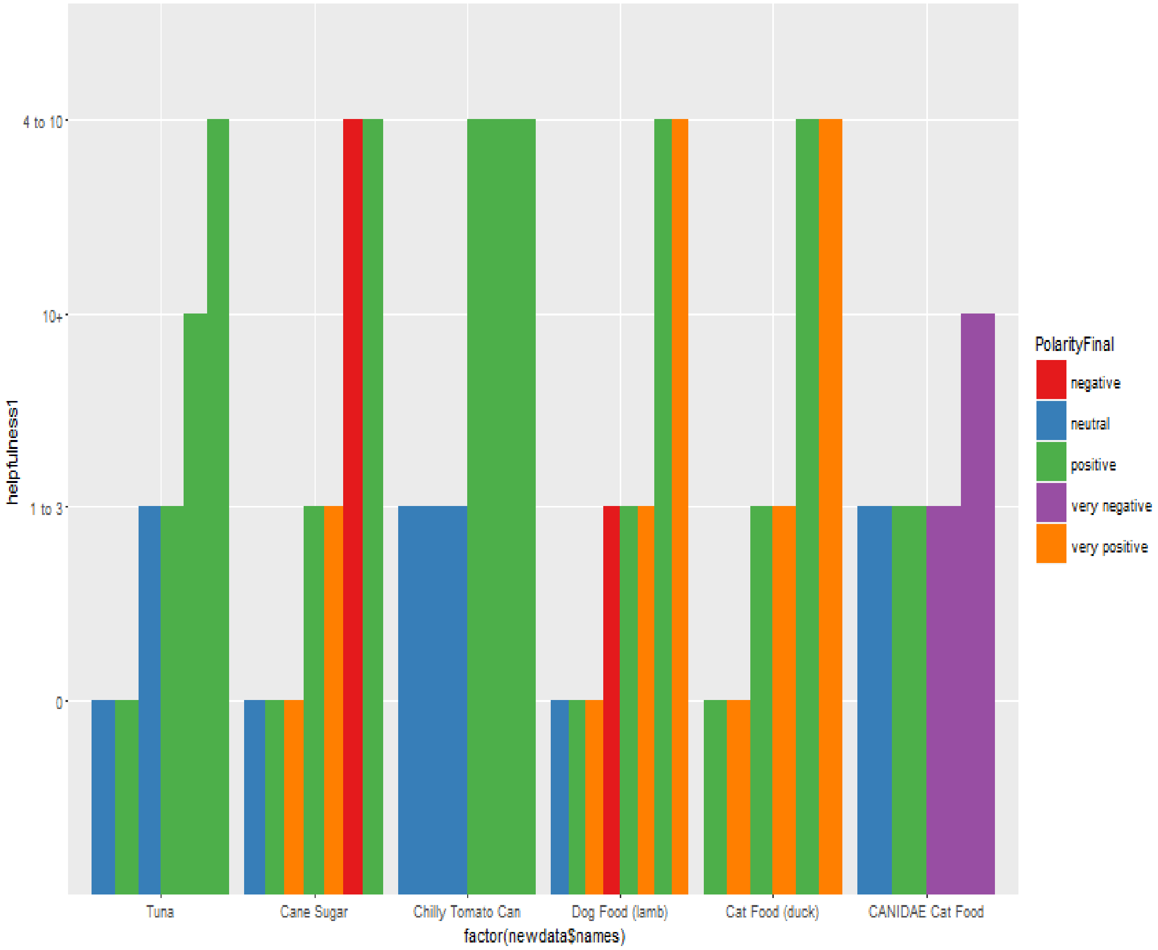
Code/filename: fetch_group.py

<u>**COMPARISONS MADE:**</u>

<u>PET FOOD v/s HUMAN FOOD:</u>

<u>Pet Food: </u> Dog Food(Lamb), Cat Food (Duck), CANIDAE Cat Food

<u>Human Food</u>: Tuna, Cane Sugar, Chilly Tomato Can



From the above plot, we know that human food has more positive reviews that are helpful as compared to the positive reviews of pet food. Cat food has highest number of negative reviews with helpfulness values varying between 1 to 10. Users have found the very positive reviews of pet food to be more helpful than those of human food.

ACROSS PRODUCT GROUPS:

The various product groups found in the dataset are:

Alcoholic Beverage (renamed to Alcohol)

Baby Product (renamed to Baby)

Beauty

BISS (Business, Industrial, and Scientific Supplies)

CE (Consumer Electronics)

Grocery

Health and Beauty (renamed to Health)

Home Improvement (renamed to Home)

Kitchen

Lawn & Patio

Musical Instruments (Renamed to Music)

Pantry

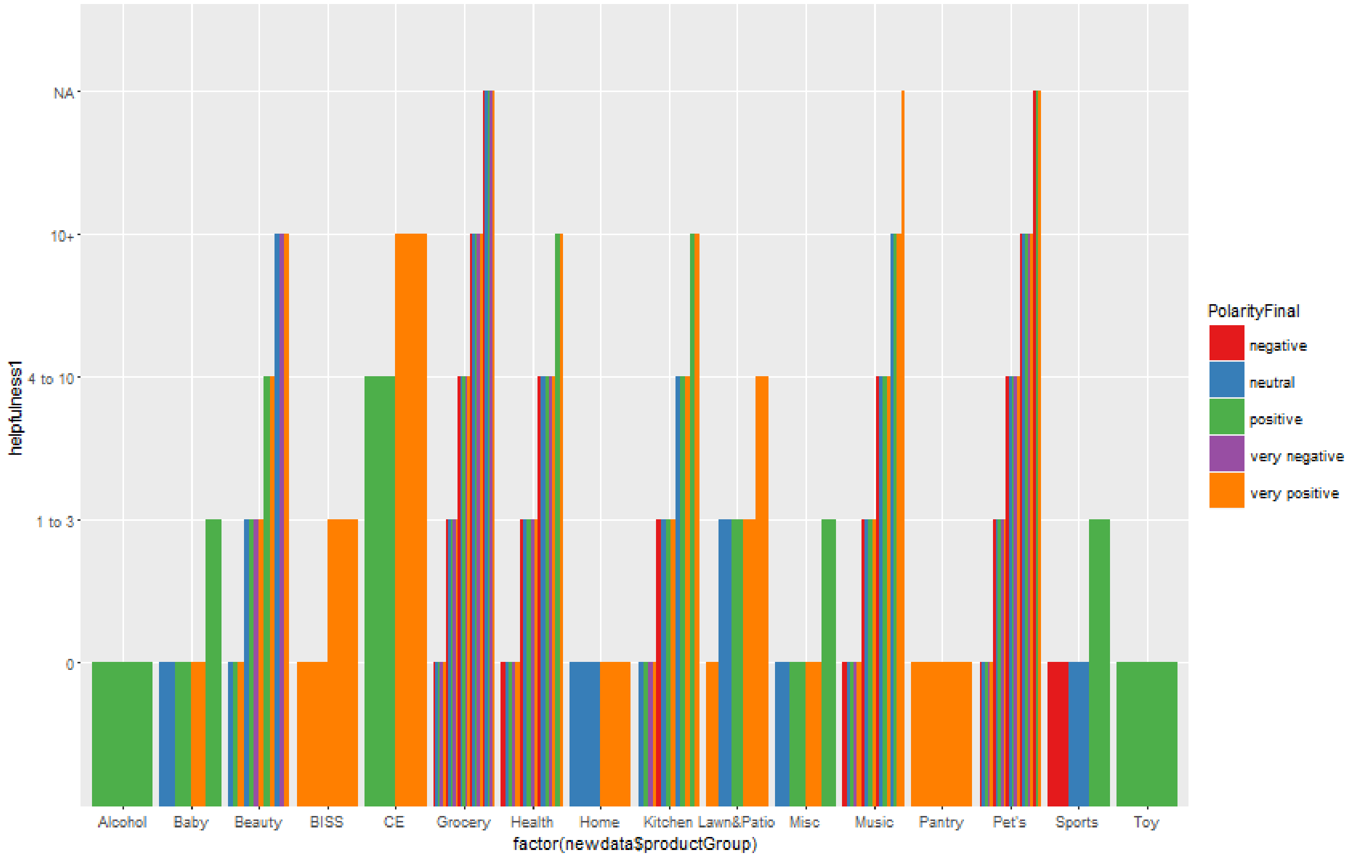
Pet Products (Renamed to Pet’s)

Single Detail Page Misc. (Renamed to Misc.)

Sports

Toy

HELPFULNESS COMPARISON ACROSS PRODUCT GROUPS:

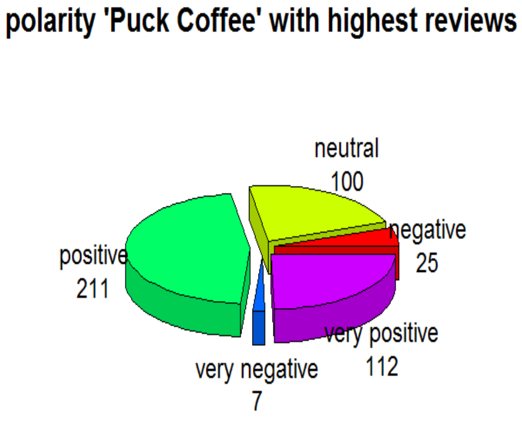


From the above plot we know that the product groups BISS (Business, Industrial, and Scientific Supplies) and CE (Consumer Electronics) have the highest number of very positive reviews, with reviews of Consumer Electronics being the most helpful (value greater than 10). Toys and Alcohol have positive reviews but no one indicated that these were helpful (helpfulness value = 0). Products under Pet Product and Grocery categories have all types of reviews with varied degrees of helpfulness. Most reviews under Sports category are either negative or neutral that are not helpful at all and some which are positive are helpful as indicated by 1 to 3 persons. Products under pantry have all very positive reviews but not helpful at all. Beauty and Health products too have varied levels of helpfulness for each review types, with Health Products having a few negative reviews as well. Similarly, Lawn & patio product types have good reviews, most of them being very positive with some being helpful as indicated by 1 to 10 persons.

PRODUCTS WITH LEAST REVIEWS AND HIGHEST REVIEWS:

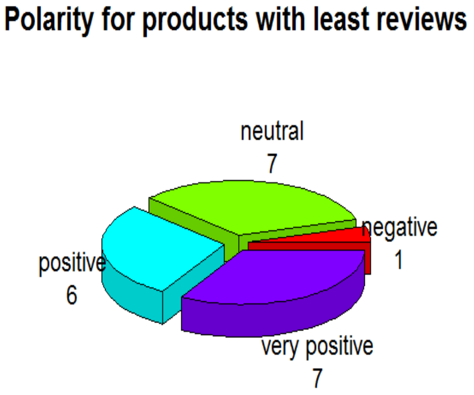
Product with highest reviews: COFFEE:

455 reviews for Puck Coffee, highest number in the dataset

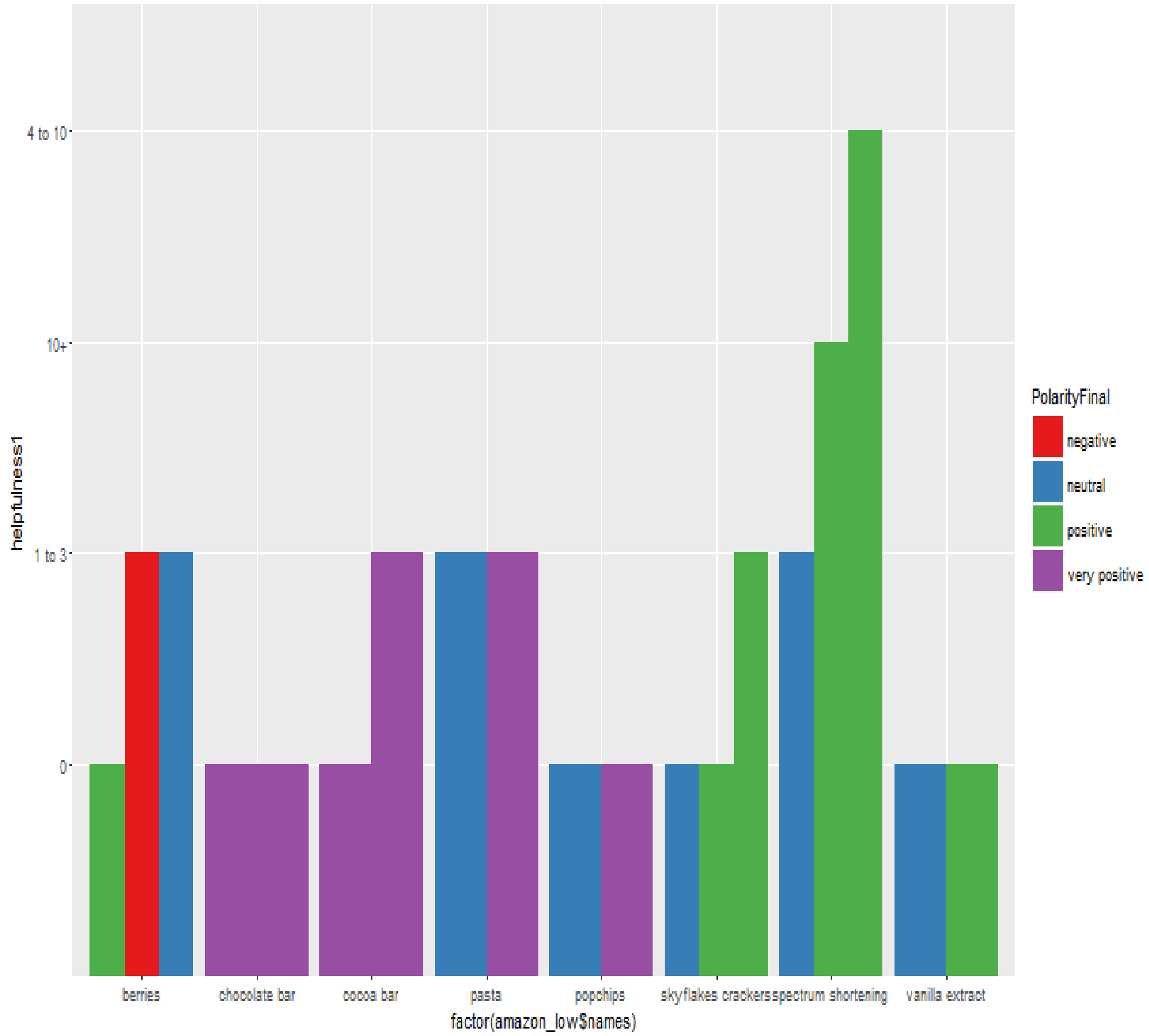


Products with least reviews (1 to 3 reviews):

Vanilla extract, Chocolate Bar, Pasta, Popchips, Skyflakes Crackers, Spectrum Shortening, Berries and Cocoa Bar

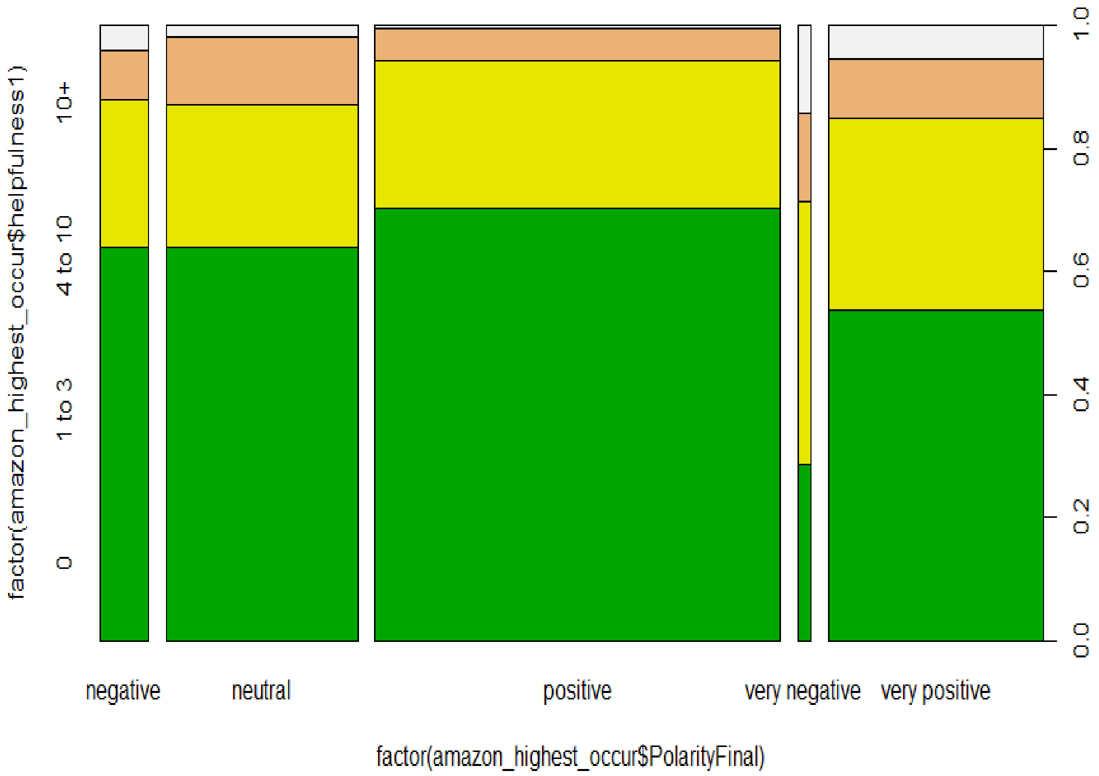


Helpfulness Comparison across products with least reviews:



Chocolate and cocoa bars have all very positive reviews, with only a few reviews of cocoa bar being helpful for 1 to 3 persons. Pasta and pop chips have either neutral or very positive reviews, with only those of pasta being helpful. Spectrum Shortening has mostly positive reviews and helpful for most people. The negative and neutral reviews of berries were helpful for 1 to 3 people. Most reviews of sky flakes crackers are positive with some of them being helpful while the remaining reviews are neutral in their sentiment levels. Vanilla extract either has neutral or positive reviews but none are helpful.

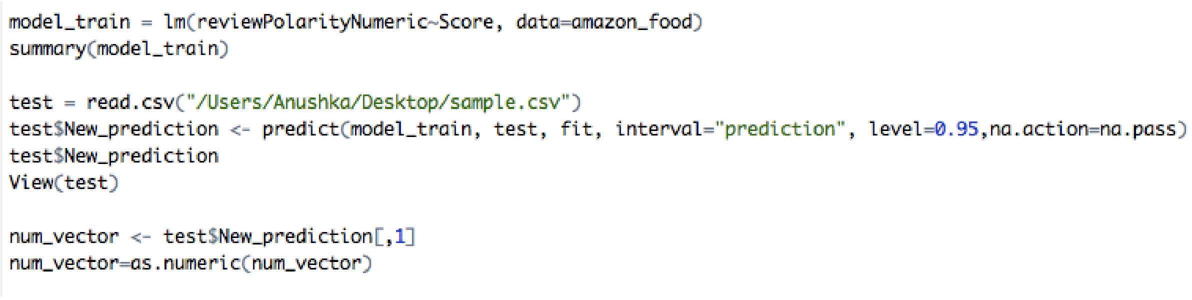
Helpfulness Comparison for Product with Highest Reviews:



Coffee mostly has positive or very positive reviews with quite a few of them being helpful. It has the least negative and very negative reviews, with a few of them being helpful for 1 to 10 people.

PREDICTION MODELS:

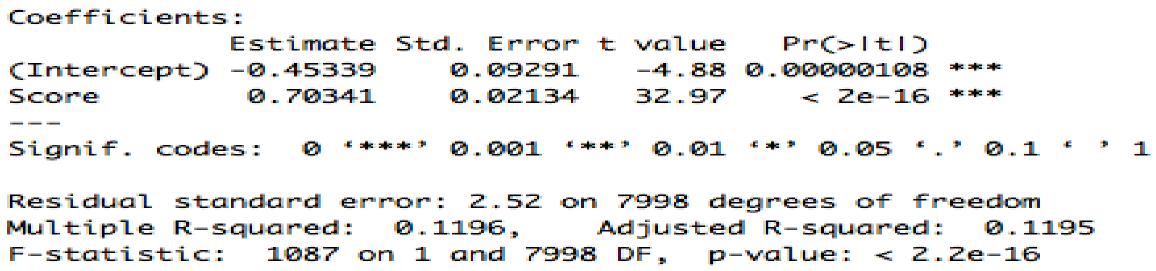
PREDICTION MODEL 1:



```
model_train = lm(reviewPolarityNumeric~Score, data=amazon_food)
summary(model_train)

test = read.csv("/Users/Anushka/Desktop/sample.csv")
test$New_prediction <- predict(model_train, test, fit, interval="prediction", level=0.95,na.action=na.pass)
test$New_prediction
View(test)

num_vector <- test$New_prediction[,1]
num_vector=as.numeric(num_vector)
```



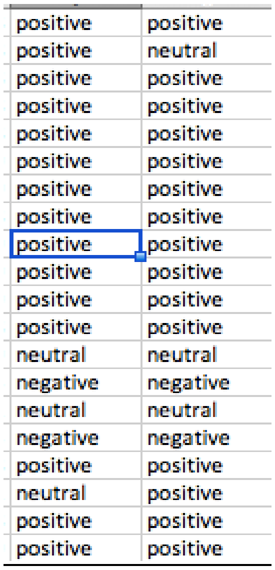
```
Coefficients:
                Estimate Std. Error t value    Pr(>|t|)    
(Intercept) -0.45339    0.09291   -4.88 0.00000108 ***
Score        0.70341    0.02134   32.97    < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.52 on 7998 degrees of freedom
Multiple R-squared:  0.1196,	Adjusted R-squared:  0.1195 
F-statistic:  1087 on 1 and 7998 DF,  p-value: < 2.2e-16
```

Built a linear regression model considering

- IV: Score
- DV: sentiment value

Sample of predicted polarity against original polarity:



positive	positive
positive	neutral
positive	positive
positive	positive
positive	positive
positive	positive
positive	positive
positive	positive
positive	positive
positive	positive
positive	positive
positive	positive
neutral	neutral
negative	negative
neutral	neutral
negative	negative
positive	positive
neutral	positive
positive	positive
positive	positive

While making final predictions, only three main categories were considered:

positive, negative and neutral

Accuracy of prediction model using linear regression with only score as IV. Number of data points considered in this case were 1000.

Accuracy = (truly classified as negative) + (truly classified as positive) / truly classified as negative) + (truly classified as positive) + (falsely classified as positive) + (falsely classified as negative)

truly classified as negative = 40

truly classified as positive = 583

falsely classified as negative = 114

falsely classified as positive = 167

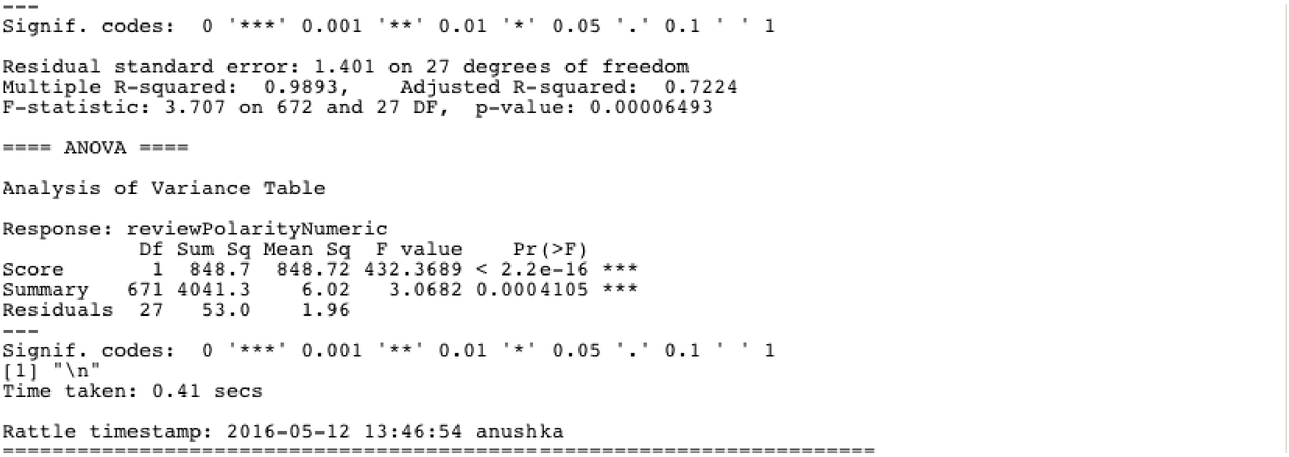
= 40 + 583 / 40+583+114+167

623/904

= 68.9%

Hence the accuracy of regression model with only score as IV is 68.9% which is considered a decent accuracy for sentiment analysis.

PREDICTION MODEL 2:



```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.401 on 27 degrees of freedom
Multiple R-squared:  0.9893,    Adjusted R-squared:  0.7224
F-statistic: 3.707 on 672 and 27 DF,  p-value: 0.00006493

==== ANOVA ====

Analysis of Variance Table

Response: reviewPolarityNumeric
           Df Sum Sq Mean Sq  F value    Pr(>F)
Score       1  848.7  848.72 432.3689 < 2.2e-16 ***
Summary   671 4041.3    6.02   3.0682 0.0004105 ***
Residuals  27   53.0    1.96
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
[1] "\n"
Time taken: 0.41 secs

Rattle timestamp: 2016-05-12 13:46:54 anushka
======================================================================
```

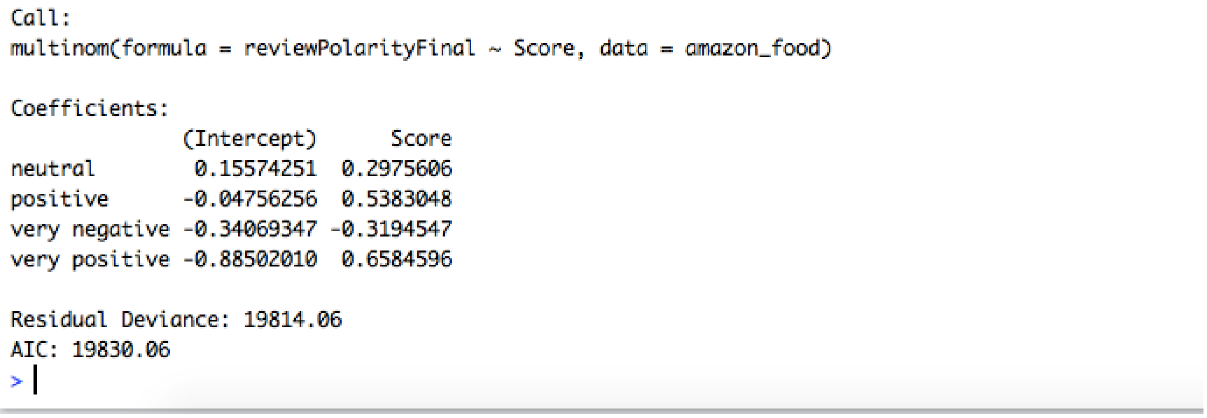
- IV: Score , Summary of text

- DV: Sentiment value

Although this model is a better model than the linear regression model taking only score as IV, it is not possible to make a prediction model based on a summary/text as IV. A document-term matrix needs to be built from the character vector to use it in prediction model.

MULTINOMIAL REGRESSION:

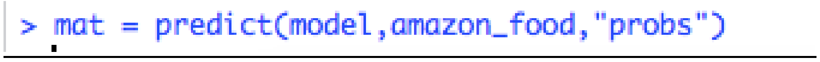
- Is used as an extension of binomial logistic regression, to predict different outcomes of a categorical DV. (categories > 2).



```
Call:
multinom(formula = reviewPolarityFinal ~ Score, data = amazon_food)

Coefficients:
              (Intercept)      Score
neutral        0.15574251  0.2975606
positive      -0.04756256  0.5383048
very negative -0.34069347 -0.3194547
very positive -0.88502010  0.6584596

Residual Deviance: 19814.06
AIC: 19830.06
> 
```

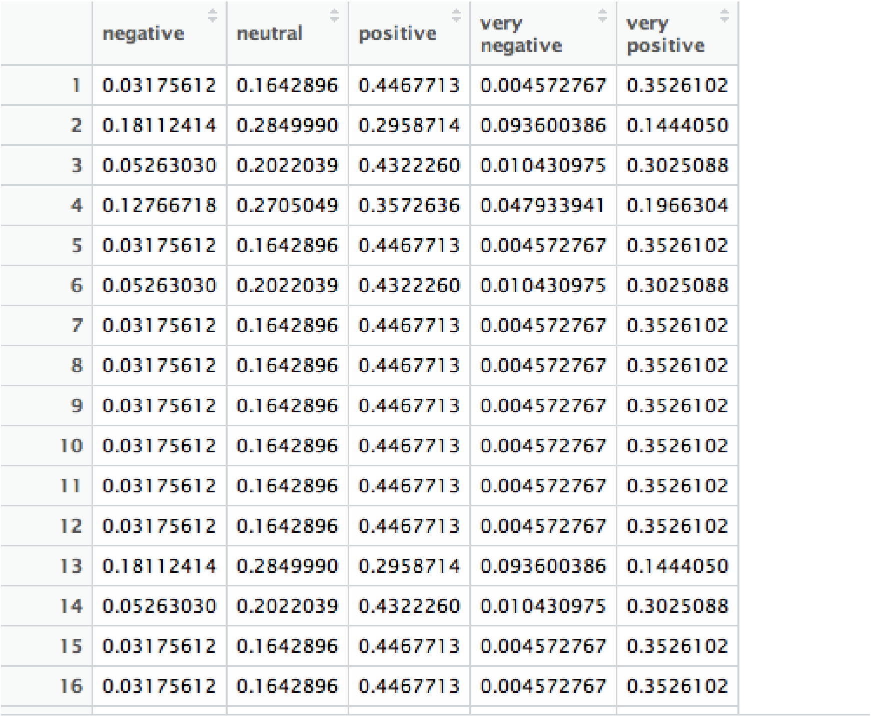


```
> mat = predict(model,amazon_food,"probs")
```

- Builds a prediction model and predicts the probabilities of each category of polarity

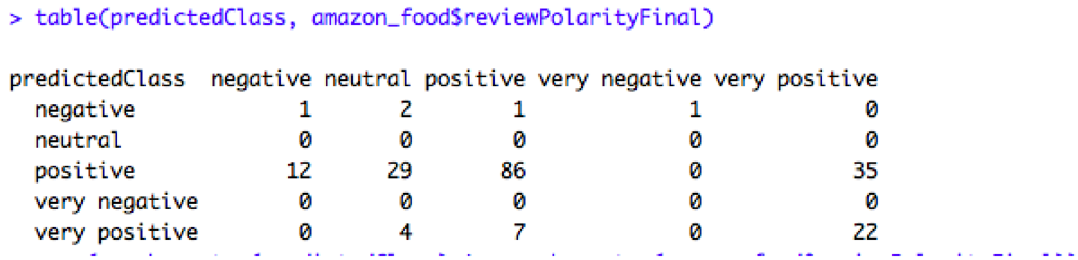
The table below gives the probabilities of each category of polarity. Looking at the following table and comparing with the findings of our earlier exploratory analysis, we know that this is a very accurate output/ prediction.

Table showing probabilities of each category of polarity:



	negative	neutral	positive	very negative	very positive
1	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102
2	0.18112414	0.2849990	0.2958714	0.093600386	0.1444050
3	0.05263030	0.2022039	0.4322260	0.010430975	0.3025088
4	0.12766718	0.2705049	0.3572636	0.047933941	0.1966304
5	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102
6	0.05263030	0.2022039	0.4322260	0.010430975	0.3025088
7	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102
8	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102
9	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102
10	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102
11	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102
12	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102
13	0.18112414	0.2849990	0.2958714	0.093600386	0.1444050
14	0.05263030	0.2022039	0.4322260	0.010430975	0.3025088
15	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102
16	0.03175612	0.1642896	0.4467713	0.004572767	0.3526102

Accuracy of prediction model using multinomial regression. Number of data points considered in this case were 200.



```
> table(predictedClass, amazon_food$reviewPolarityFinal)

predictedClass  negative neutral positive very negative very positive
  negative             1       2        1             1             0
  neutral              0       0        0             0             0
  positive            12      29       86             0            35
  very negative        0       0        0             0             0
  very positive        0       4        7             0            22
```

truly classified as negative = 2

truly classified as positive = 150

falsely classified as negative = 3

falsely classified as positive = 12 + 29 + 4 = 45

2+150/2+150+3+45

= 76%

LIMITATIONS & FUTURE WORK:

Limitations:

- The data consists of reviews until October 2012.
- Helpfulness, product description, review text was not used to build a prediction model
- Entire dataset was not used to perform sentiment analysis

Future Work:

- Text of review can be used directly to make predictions instead of generating a sentiment value
- Some more analysis can be shown between product description and other features of the dataset.

REFERENCES:

- J. McAuley and J. Leskovec. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. WWW, 2013.