Amazon Products Review

Group 4

Maria De La Oliva, Amelia Mokal, Zixuan Wu
and Rebecca Amare

Agenda

1
Objective
By: Rebecca Amare

2
Our Data
By: Zixuan Wu

3
Business Questions
By: Amelia Mokal

4
Analysis
By: All

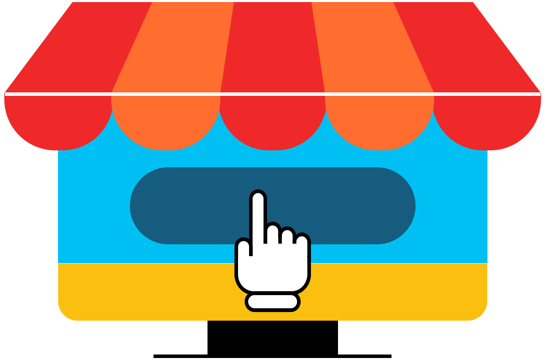
5
Conclusion
By:Maria De La Oliva

Objective

Using a variety of components such as, Impala, Anaconda and Amazon SageMaker to analyze Amazon product reviews to portray any patterns among product purchases. After generating results, come to a conclusion if particular patterns assist Amazon on deciding which products they would need to invest in. In addition, define other programs that could help in the decision making process on product purchasing.

Data Variables

Electronics (1.73GB), Grocery (956MB), Furniture (367MB)



Marketplace: 2 letter country code of the marketplace where the review was written
Customer_id and review_id: Unique customer's and review id
*** Product_id**: The unique Product ID the review pertains to. In the multilingual dataset the reviews for the same product in different countries can be grouped by the same product_id.
Product_parent: Identifier that can be used to aggregate reviews for the same product.
***Product_title:** Name of product
Product_category: Broad product category that can be used to group reviews
***Star_rating**: Ratings from 1-5 (lowest to highest)
***Helpful_votes**: If review was positivity helpful to consumer
***Total_votes:** Total of both positive and negative votes
***Vine:** Reviews written by reviewers in the Amazon Vine Program
***Verified_purchase:** Amazon verified that the person writing the review purchased the product at Amazon and didn't receive the product at a deep discount.
Review_headline: Review subject line
Review_body: Consumers full review of the product
Review_date: Date published

Business Questions

How does the polarity of the reviews vary across the three product categories?”

Question 1

Should Amazon keep the Amazon Vine Program ?

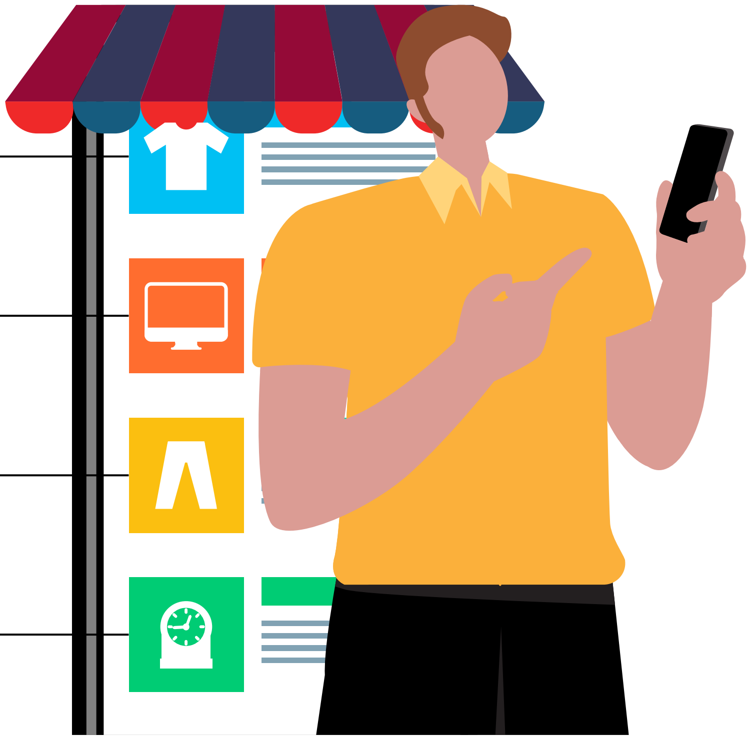
Question 2

Which products have the most positive reviews?

Question 3

Are there any patterns associated with “helpful_votes?

Question 4



Descriptive Analysis

```sql
/*Number of reviews per category*/
select product_category as "Product Category",
count(*) as "Number of Reviews"
from amazon_reviews group by product_category;
```

product category	number of reviews
Furniture	792113
Grocery	2402458
Electronics	3093872

```sql
/*Number of customers that left a review per category*/
select product_category,
count(distinct customer_id) as "Number of Customers"
from amazon_reviews
group by product_category;
```

product_category	number of customers
Grocery	1363986
Electronics	2154351
Furniture	656007

```sql
/*Number of reviews made by
customers in Amazon Vine Program*/
select count(review_id)
as "No. of Vine Program Reviews"
from amazon_reviews
where vine = "Y";
```

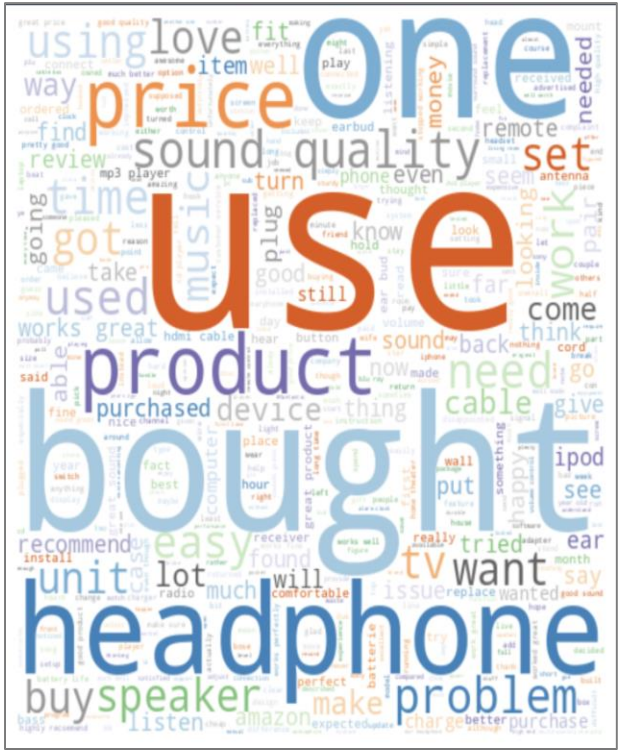
no. of vine program reviews
37899

```sql
/*Number of unique products reviewed per category*/
select product_category,
count (distinct product_parent) as "Unique Products"
from amazon_reviews
group by product_category;
```

product_category	unique products
Grocery	268150
Electronics	166244
Furniture	113252

Word Frequency

GROCERY

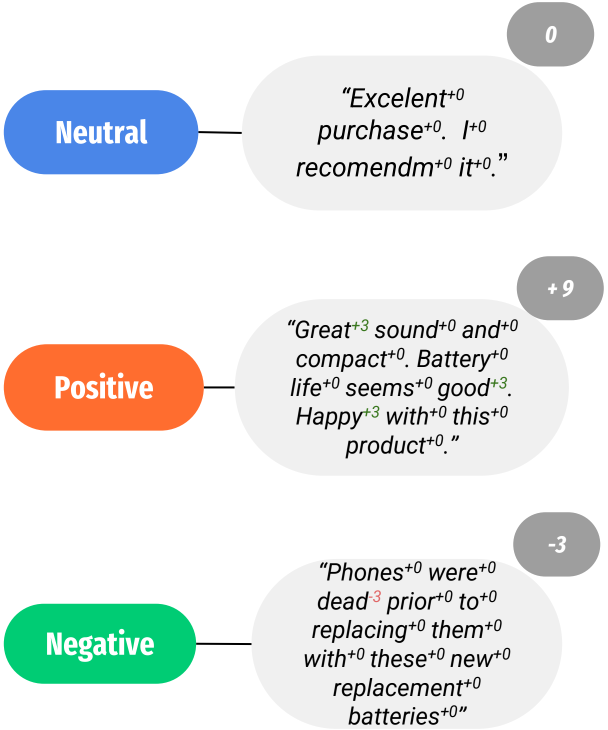


ELECTRONICS

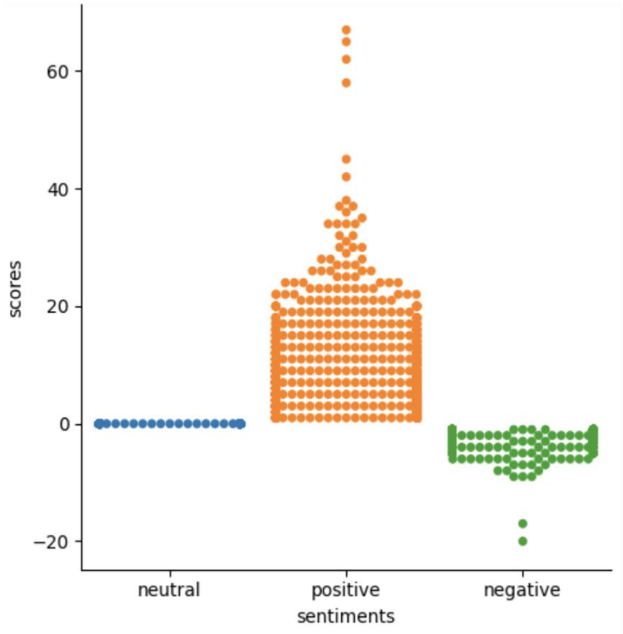
FURNITURE

Sentiment Analysis using AFINN

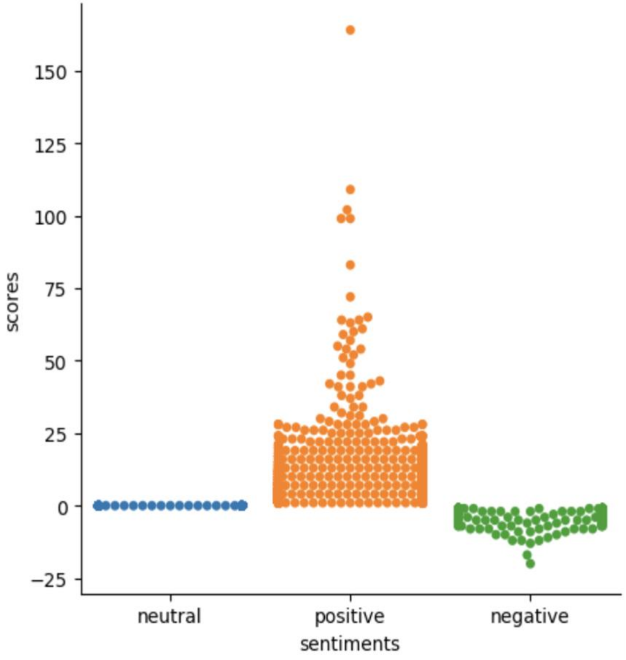
As described.	0.0	neutral
It works as advertising.	0.0	neutral
Works pissa	0.0	neutral
Did not work at all.	0.0	neutral
Works well. Bass is somewhat lacking but is pr...	3.0	positive
The quality on these speakers is insanely good...	6.0	positive
Wish I could give this product more than five ...	5.0	positive
works great	3.0	positive
Great sound and compact. Battery life seems go...	9.0	positive
It works well~~~	0.0	neutral



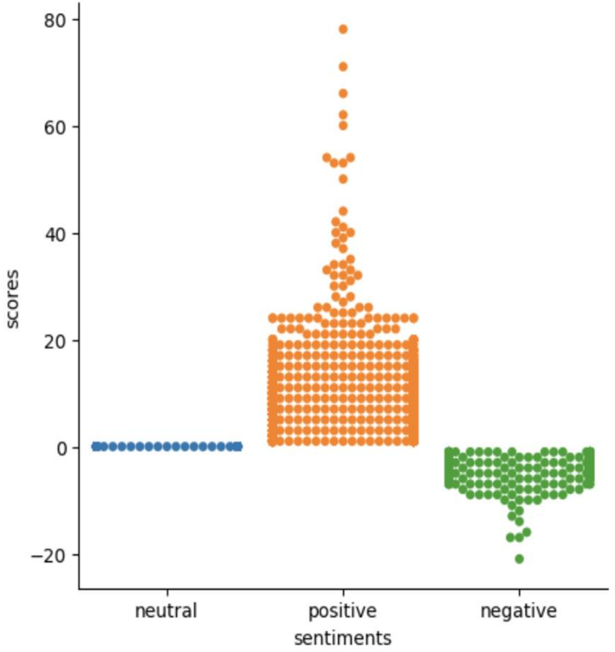
Sentiment Analysis Results



GROCERY



ELECTRONICS



FURNITURE

Sentiment Analysis - Predictive Model

1	What are we predicting?	The Polarity of product reviews
2	Expected results	Will have an high accuracy score for positive reviews
3	What model is chosen?	Random Forest

Random Forest Model

Platform

AWS SageMaker

Baseline

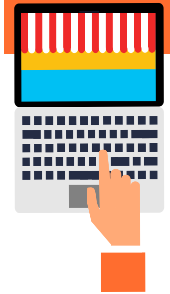
Cross validation : 78%

Data

~ 2 million rows
37 % of the dataset

Frequency

Positive : 78%
Neutral : 7%
Negative: 15%



Distribution

Random Sampling

```
[4]: data_key_Electronics = "project/Electronics.txt"
     data_location_e = "s3://{}/{}".format(bucket,data_key_Electronics)


     Electronics = pd.read_csv(data_location_e, sep="\t")
```

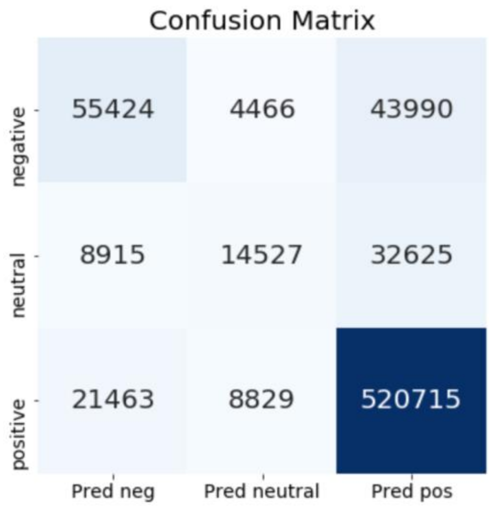
Receiving Furniture dataset

```
[7]: data_key_furniture = "project/Furniture.txt"
     data_location_f = "s3://{}/{}".format(bucket,data_key_furniture)

     Furniture = pd.read_csv(data_location_f, sep="\t")
```

Receiving Grocery dataset

```
[8]: data_key_Grocery = "project/Grocery.txt"
     data_location_g = "s3://{}/{}".format(bucket,data_key_Grocery)

     Grocery = pd.read_csv(data_location_g, sep="\t")
```

Random Forest Model



	precision	recall	f1-score	support
negative	0.65	0.53	0.58	103880
neutral	0.52	0.26	0.35	56067
positive	0.87	0.95	0.91	551007
accuracy			0.83	710954
macro avg	0.68	0.58	0.61	710954
weighted avg	0.81	0.83	0.82	710954

	F1-Score Comparison			
	Combined	Electronics	Furniture	Grocery
Negative	.58	.47	.52	.44
Neutral	.35	.07	.11	.10
Positive	.91	.87	.88	.90

Conditions - One star ratings vs Five star ratings

Where: 1= True (Number of stars),
0= False (Star rating does not fit condition)

```
CombinedGrouped['star1']= np.where(CombinedGrouped['star_rating']== 1, 1,0)
CombinedGrouped['star1']
print('\nRow Condition :\n', CombinedGrouped['star1'])

Row Condition :
 0         0
1         0
2         0
3         0
4         0
          ..
792108    0
792109    0
792110    0
792111    0
792112    0
Name: star1, Length: 6288433, dtype: int64
```

```
CombinedGrouped['star5']= np.where(CombinedGrouped['star_rating']== 5, 1,0)
CombinedGrouped['star5']
print('\nRow Condition :\n', CombinedGrouped['star5'])

Row Condition :
 0         1
1         1
2         1
3         1
4         1
          ..
792108    1
792109    1
792110    1
792111    0
792112    1
Name: star5, Length: 6288433, dtype: int64
```

Multinomial Logit Model and R-Squared

```
Optimization terminated successfully.
         Current function value: 0.061601
         Iterations 10
                           Logit Regression Results
==============================================================================
Dep. Variable:            vine_program   No. Observations:               928159
Model:                           Logit   Df Residuals:                   928156
Method:                            MLE   Df Model:                            2
Date:                 Sun, 21 Aug 2022   Pseudo R-squ.:                 -0.7036
Time:                         21:27:04   Log-Likelihood:                -57176.
converged:                        True   LL-Null:                       -33562.
Covariance Type:             nonrobust   LLR p-value:                     1.000
=================================================================================
                  coef    std err          z      P>|z|      [0.025      0.975]
---------------------------------------------------------------------------------
star_rating    -1.5668      0.006   -277.243      0.000      -1.578      -1.556
total_votes    -0.3459      0.008    -42.493      0.000      -0.362      -0.330
helpful_votes   0.3544      0.008     42.593      0.000       0.338       0.371
=================================================================================
```

Variables

Dependent Variable-Vine Program

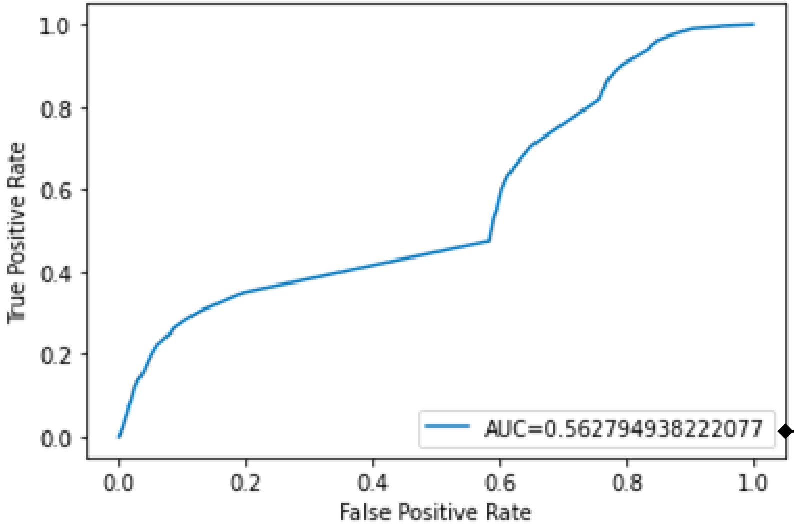
Independent Variables- Star Ratings, Total Votes, Helpful Votes.

```
-0.4428249127024606
```

R-Squared

-44%

ROC Curve



AUC

“Area under the curve”

Closer to 1- Better the model

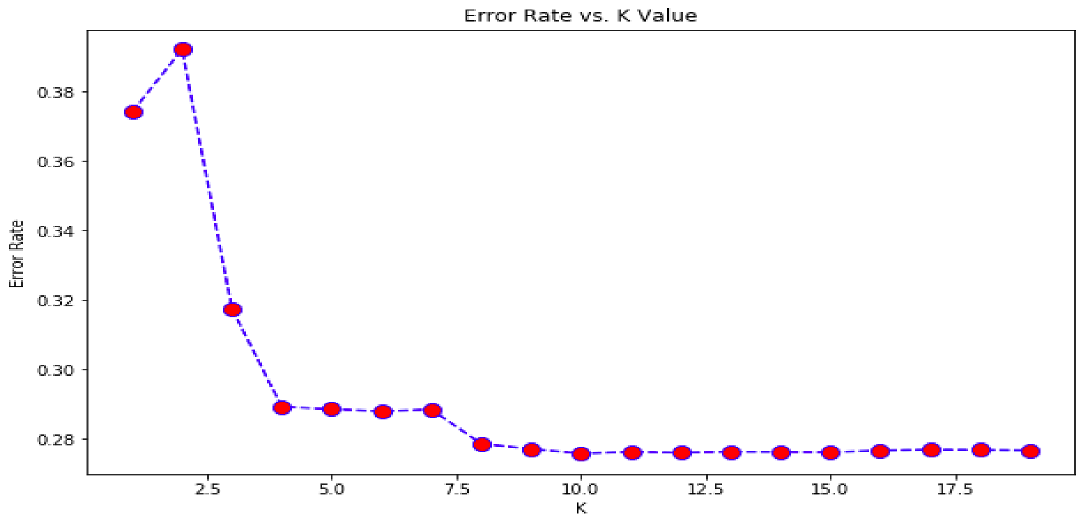
K-Means Clustering

Find Clusters (the Elbow Method)

Dataset (Product Category, first 100,000 rows each)	Best K-value
Grocery	9
Furniture	18
Electronics	18

```
error_rate = []
for i in range(1,20):
    knn = KNeighborsClassifier(n_neighbors=i)
    knn.fit(X_train,y_train)
    pred_i = knn.predict(X_test)
    error_rate.append(np.mean(pred_i != y_test))

plt.figure(figsize=(10,6))
plt.plot(range(1,20),error_rate,color='blue', linestyle='dashed',
         marker='o',markerfacecolor='red', markersize=10)
plt.title('Error Rate vs. K Value')
plt.xlabel('K')
plt.ylabel('Error Rate')
print("Minimum error:-",min(error_rate),"at K =",error_rate.index(min(error_rate)))
```

```
Minimum error:- 0.2757 at K = 9
```



Compare Clusters

[29]:
```
Grocery_S.groupby('cluster').mean()
```

[29]:

cluster	star_rating	helpful_votes	total_votes	verifiedpurchase_Y	vine_Y
0	2.002956	0.376248	0.764319	0.856542	0.008605
1	5.000000	274.222222	291.666667	1.000000	0.000000
2	1.000000	1377.000000	1463.000000	1.000000	0.000000
3	3.294574	51.821705	58.426357	0.790698	0.000000
4	3.590909	137.272727	151.272727	0.636364	0.000000
5	3.000000	583.000000	693.000000	0.500000	0.000000
6	4.867070	0.205473	0.281409	0.894788	0.004128
7	3.679684	4.391716	5.522091	0.791716	0.005325
8	3.486819	17.989455	21.304042	0.783831	0.008787

[81]:
```
Cluster2=Grocery_S[Grocery_S.cluster == 2]
Cluster2
```

[81]:

	product_id	product_title	star_rating	helpful_votes	total_votes	verifiedpurchase_Y	vine_Y	cluster
47130	B00V4KWGEI	Epic All Natural Meat Bar, 100% Natural	1	1377	1463	1	0	2

[14]:
```
Electronics_S.groupby('cluster').mean()
```

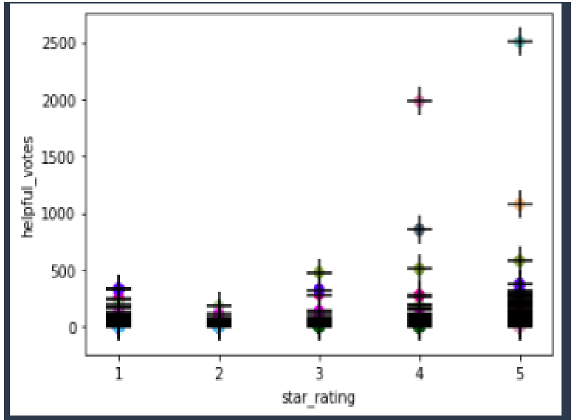
[14]:

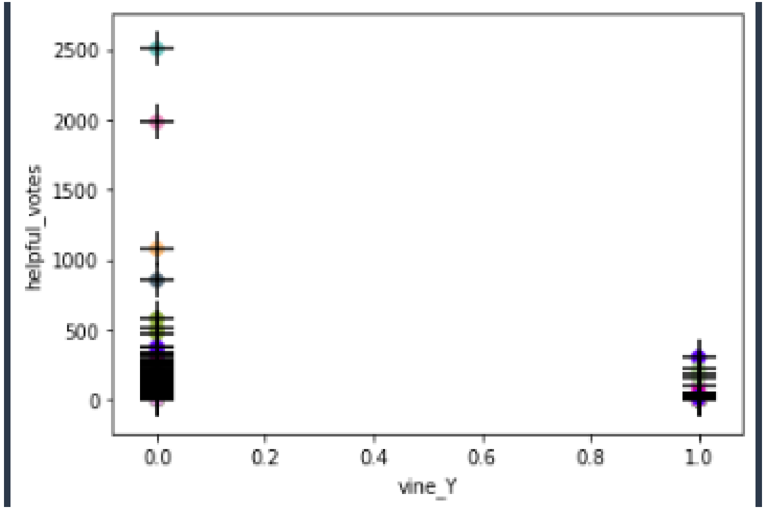
cluster	star_rating	helpful_votes	total_votes	verifiedpurchase_Y	vine_Y
0	4.689652	1.278039	1.616542	0.857380	0.014031
1	4.000000	1982.000000	2045.000000	1.000000	0.000000
2	3.909091	262.000000	276.363636	0.727273	0.000000
3	5.000000	1076.000000	1142.000000	1.000000	0.000000
4	3.678899	41.990826	55.137615	0.706422	0.018349
5	5.000000	2506.000000	2720.000000	1.000000	0.000000
6	3.950000	182.500000	202.600000	0.700000	0.150000
7	3.750000	327.125000	358.000000	0.625000	0.125000
8	3.218892	4.447194	6.262461	0.772046	0.021262
9	4.000000	518.666667	563.333333	0.666667	0.000000
10	1.320552	0.303743	0.600525	0.916875	0.001116
11	3.515152	25.409091	31.363636	0.715909	0.022727
12	3.800000	73.711111	83.000000	0.666667	0.000000
13	4.000000	851.000000	876.000000	1.000000	0.000000
14	4.029412	116.441176	131.029412	0.676471	0.029412
15	3.674071	0.034688	0.116212	0.931991	0.006511
16	3.539642	11.860614	15.317136	0.719949	0.021739
17	5.000000	0.000000	0.047194	0.937839	0.001767

Findings:

Using Electronics product category as an example

- Helpful votes & Star_rating - positive impact;
- Found products which have the most positive reviews:
  - Grocery: San Francisco Bay One Cup
  - Electronics: Panasonic ErgoFit In-Ear Earbud Headphone
  - Furniture: Zinus SC-SBBK-14NT-FR Smartbase Bed Frame Metal, Narrow Twin)
- Vine Program & Helpful_votes - Undetermined!






```
Five_star.product_title.value_counts()

Panasonic ErgoFit In-Ear Earbud Headphone
16864
Mediabridge ULTRA Series HDMI Cable (3 Foot) - High-Speed Supports Ethernet, 3D and Audio Return [Newest Standar
d]      13520
AmazonBasics High-Speed HDMI Cable - 6.5 Feet (2 Meters) Supports Ethernet, 3D, 4K and Audio Return
13365
AmazonBasics High Speed HDMI Cable
8997
CABTE High speed HDMI 1.4 HDMI cable 10ft 1080p with mesh&filters supports 3D&blue ray
8461

...
ABLEGRID @ Trademarked AC DC Adapter For Sony ZS H10CP ZSH10CP Radio CD MP3 Player Boombox power wire cord Brand
New        1
OYAIDE HPC-62HDX Black 1.3m Headphone cable
1
Inova Solutions 4-Digit PoE Network Clock - Off-White Plastic - Red LEDs
1
New LCD Video Cable for 15.4 Inch Acer Aspire 3020 3610 5020 TravelMate 2410 4400 series laptop. (Not fit 15 inc
h)          1
JVC RX-668 Audio/Video Receiver
1
Name: product_title, Length: 123503, dtype: int64
```

Business Impacts & Conclusion

1. Increase marketing and improvements
2. Next Steps Vine program
3. Investigation and Decision-making

Thank you!