“Bestselling Product Prediction using KNN & Classification Trees” ”

- Sarath Kumar Vatyam
- Abhi Ram Sai
- Neeharika Vijjada

Problem Context

In the competitive landscape of e-commerce, accurately predicting whether a product will become a bestseller is paramount for businesses seeking to optimize their operations. Current methods often fall short, relying on limited factors.

Our research aims to fill this gap by developing machine learning models that leverage various features to predict product success more accurately.

Through this study, we aim to identify the key drivers of bestseller status and provide businesses with actionable insights for improving their marketing and sales strategies.

”Identifying whether a product will become a bestseller is crucial for businesses to optimize their inventory, marketing, and sales strategies”.

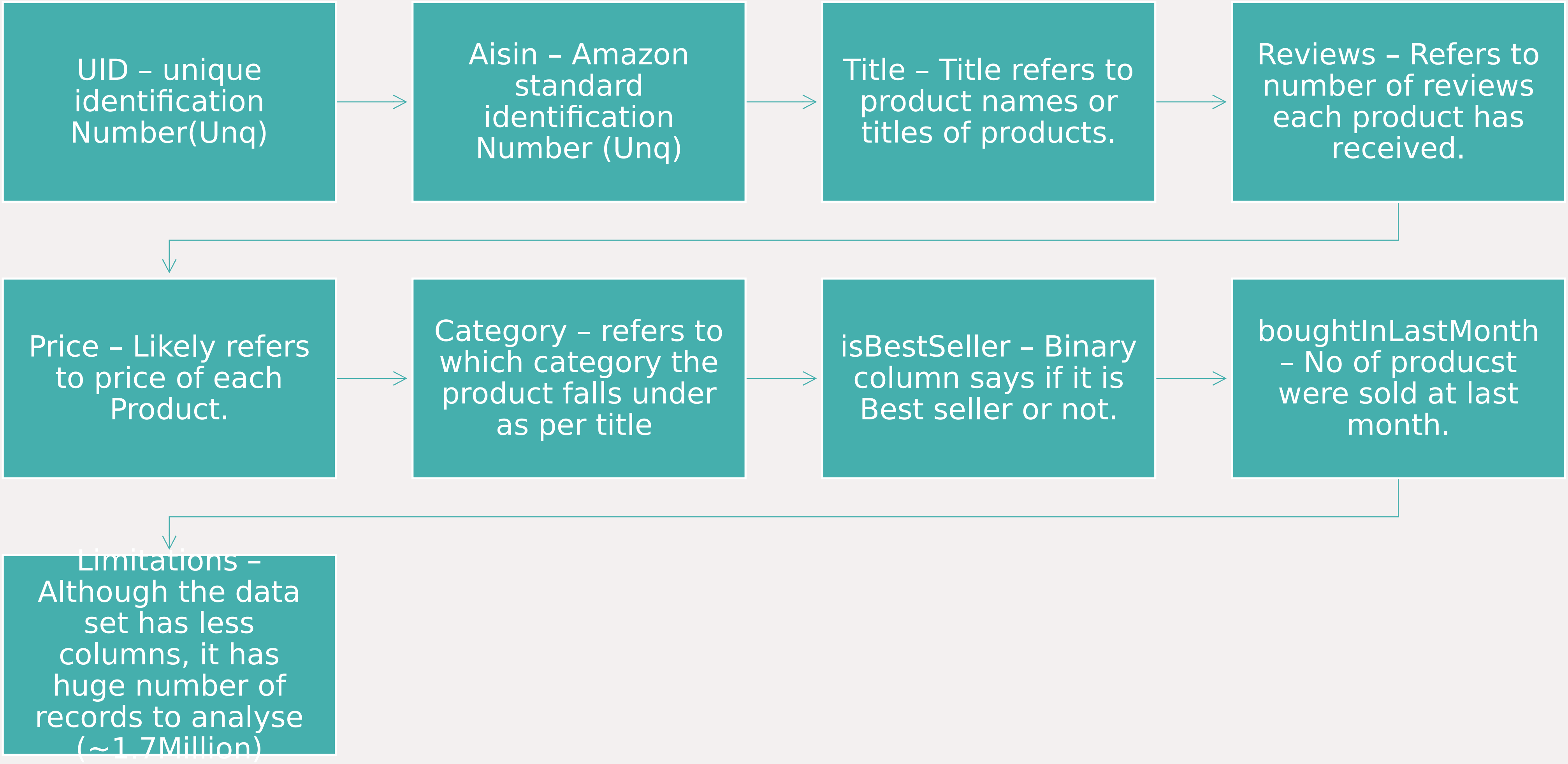
Research Questions-

"What features contribute most to a product becoming a bestseller?”

"How accurately can we predict whether a product will be a bestseller based on historical data?”

We have tried KNN and Regression tree to find out ‘isBestseller’ from the dataset given.

Description of Dataset



UID – unique identification Number(Unq)

Aisin – Amazon standard identification Number (Unq)

Title – Title refers to product names or titles of products.

Reviews – Refers to number of reviews each product has received.

Price – Likely refers to price of each Product.

Category – refers to which category the product falls under as per title

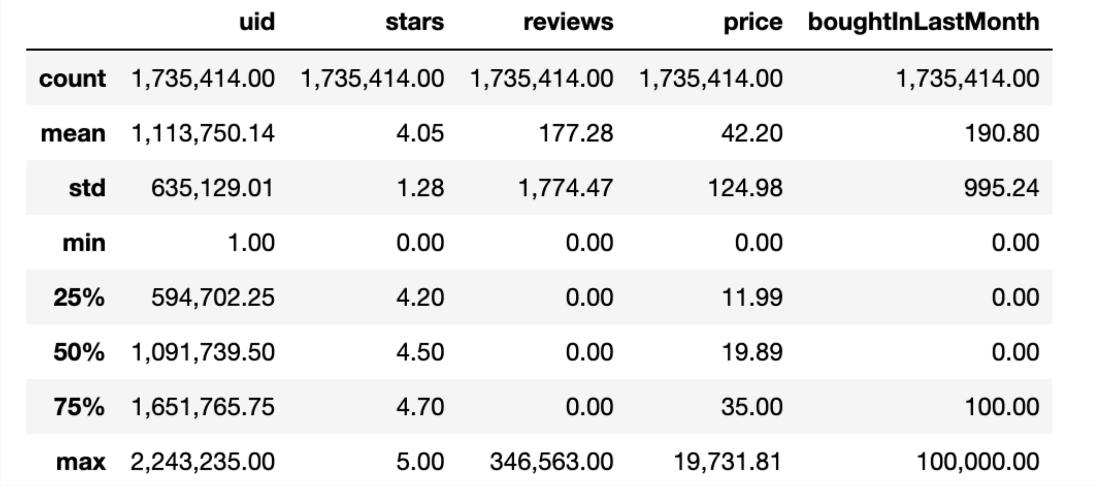
isBestSeller – Binary column says if it is Best seller or not.

boughtInLastMonth – No of producst were sold at last month.

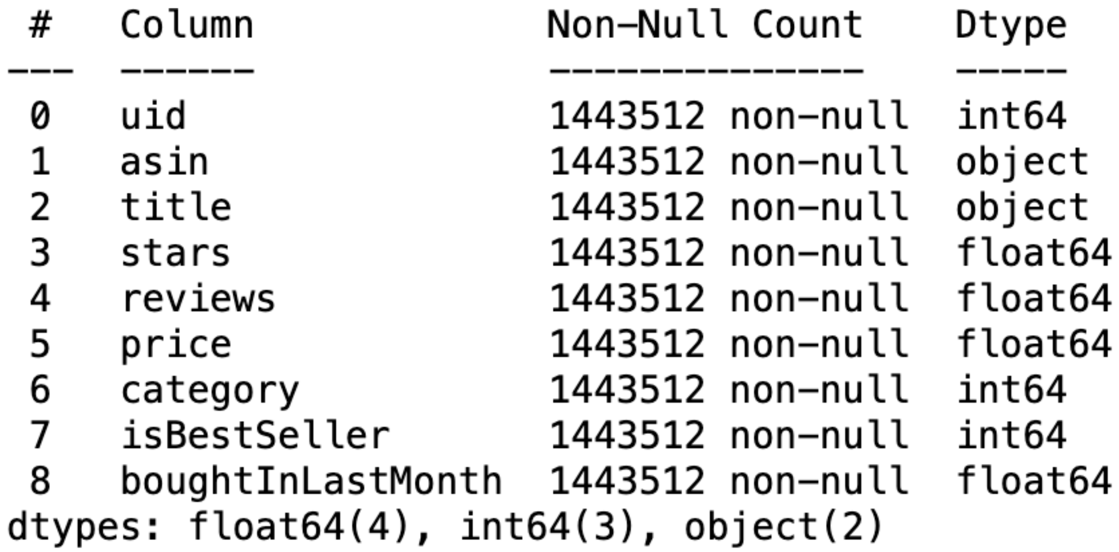
Limitations – Although the data set has less columns, it has huge number of records to analyse (~1.7Million)

Source, Key Statistics

- We have sourced this dataset from KAGGLE.
- This Dataset contains ~1.7Mn records with 9 columns.
- Key Statistics –



	uid	stars	reviews	price	boughtInLastMonth
count	1,735,414.00	1,735,414.00	1,735,414.00	1,735,414.00	1,735,414.00
mean	1,113,750.14	4.05	177.28	42.20	190.80
std	635,129.01	1.28	1,774.47	124.98	995.24
min	1.00	0.00	0.00	0.00	0.00
25%	594,702.25	4.20	0.00	11.99	0.00
50%	1,091,739.50	4.50	0.00	19.89	0.00
75%	1,651,765.75	4.70	0.00	35.00	100.00
max	2,243,235.00	5.00	346,563.00	19,731.81	100,000.00



```
 #   Column             Non-Null Count    Dtype
---  ------             --------------    -----
 0   uid                1443512 non-null  int64
 1   asin               1443512 non-null  object
 2   title              1443512 non-null  object
 3   stars              1443512 non-null  float64
 4   reviews            1443512 non-null  float64
 5   price              1443512 non-null  float64
 6   category           1443512 non-null  int64
 7   isBestSeller       1443512 non-null  int64
 8   boughtInLastMonth  1443512 non-null  float64
dtypes: float64(4), int64(3), object(2)
```

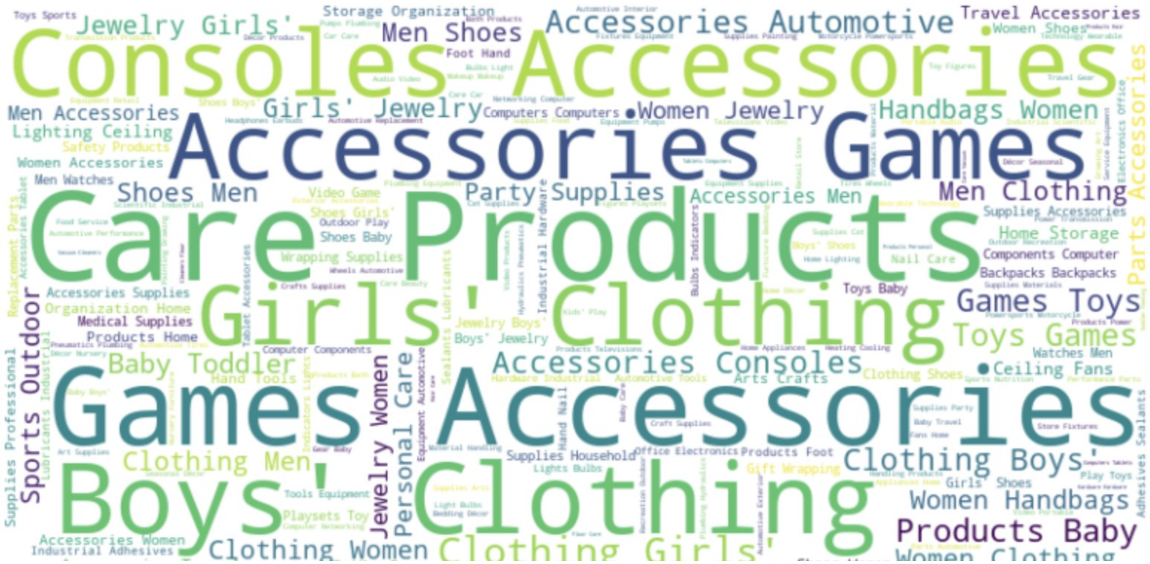
Data preprocessing

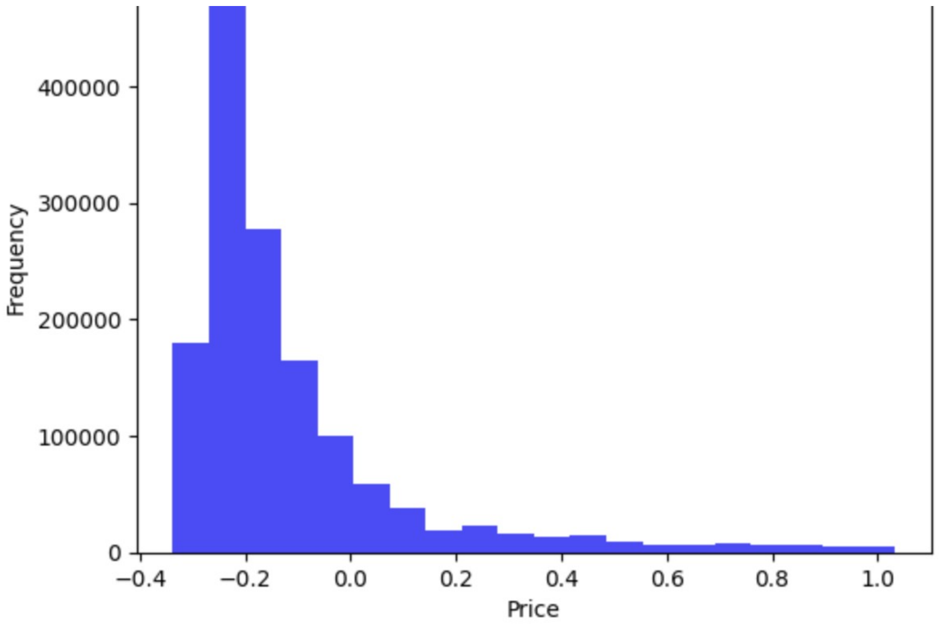
- We ensured that there are no missing values or duplicates in the columns.
- We have removed uid , aisin, title columns as they are product lookup references , however not useful for analysis part.
- We have created dummies for categorical variables in

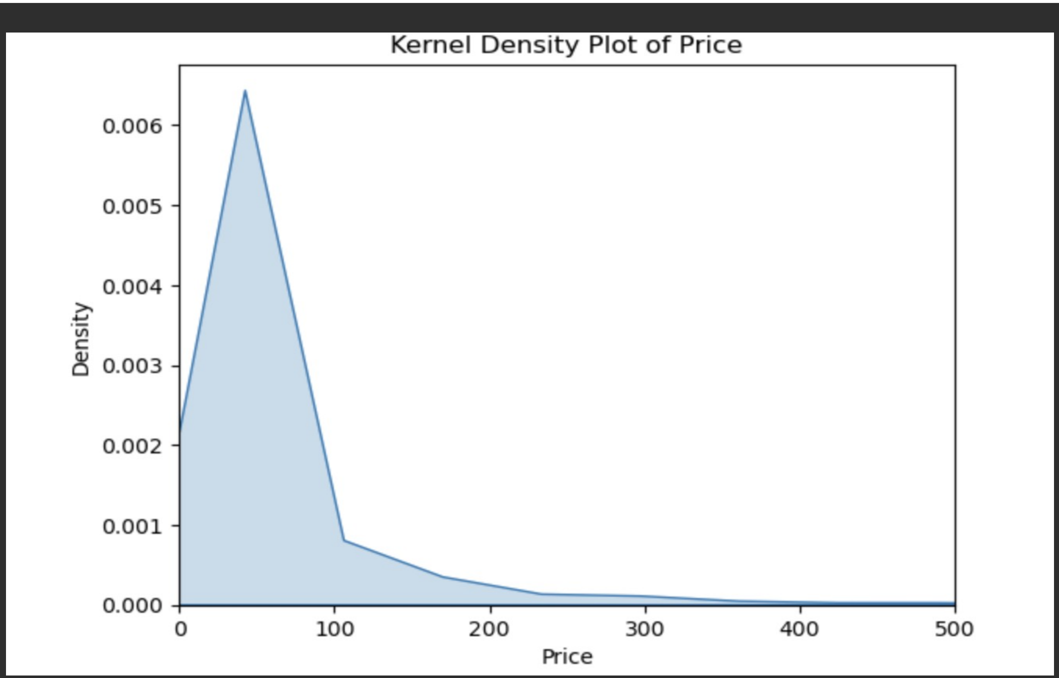
-the dataset for ’Category’ and ‘iBestSeller’ Columns.

- Checked datatypes of all the columns.
- We have removed the columns for which prices are 0 ,

- Approximately ~37k Records.







- Word cloud image shows highest sold products category names bigger as they appear.
- In this dataset, the price range (-0.27, -0.2] has the highest frequency (484,119), suggesting that a significant number of items fall within this price interval.
- As per Kernel density plot , we the more price density in 0-100

K-NN Model

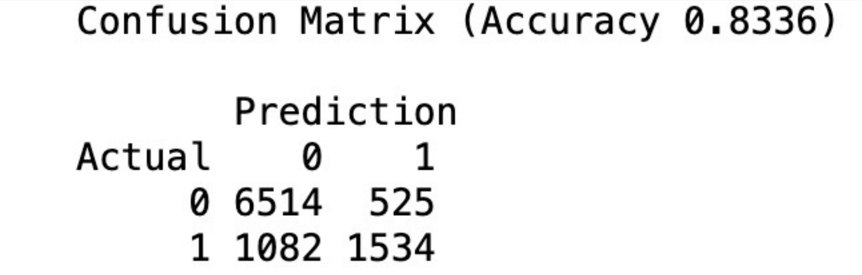
- K – Nearest Neighbors is a non – parametric , supervised learning classifier , which uses proximity to make predictions about the grouping of an individual dataset.
- After pre-processing and data exploration , we have split the data into train data and validation data to run the model
- We have normalized the predictor columns and used the z-score values for analysis and prediction.
- We have trained the model accordingly on the train dataset.

Findings

- We have run the model on train dataset and figured the highest accuracy of K on training data set at K = 13 I.e., 83.59% of accuracy

- Hence, we have decided to use K= 13 and run the model on that validation dataset .

- After running K= 13 , we have got 83.36% of accuracy in prediction with respect to validation dataset.

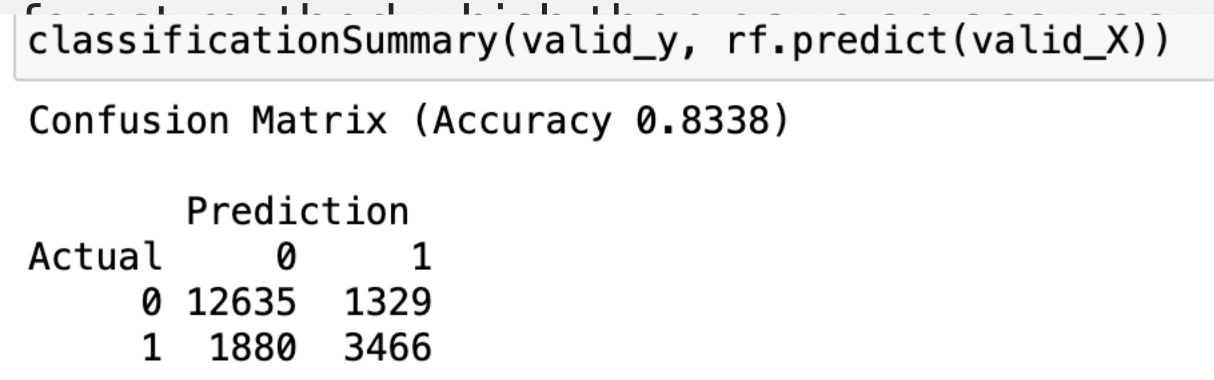


```
Confusion Matrix (Accuracy 0.8336)

       Prediction
Actual    0    1
     0 6514  525
     1 1082 1534
```

Classification tree

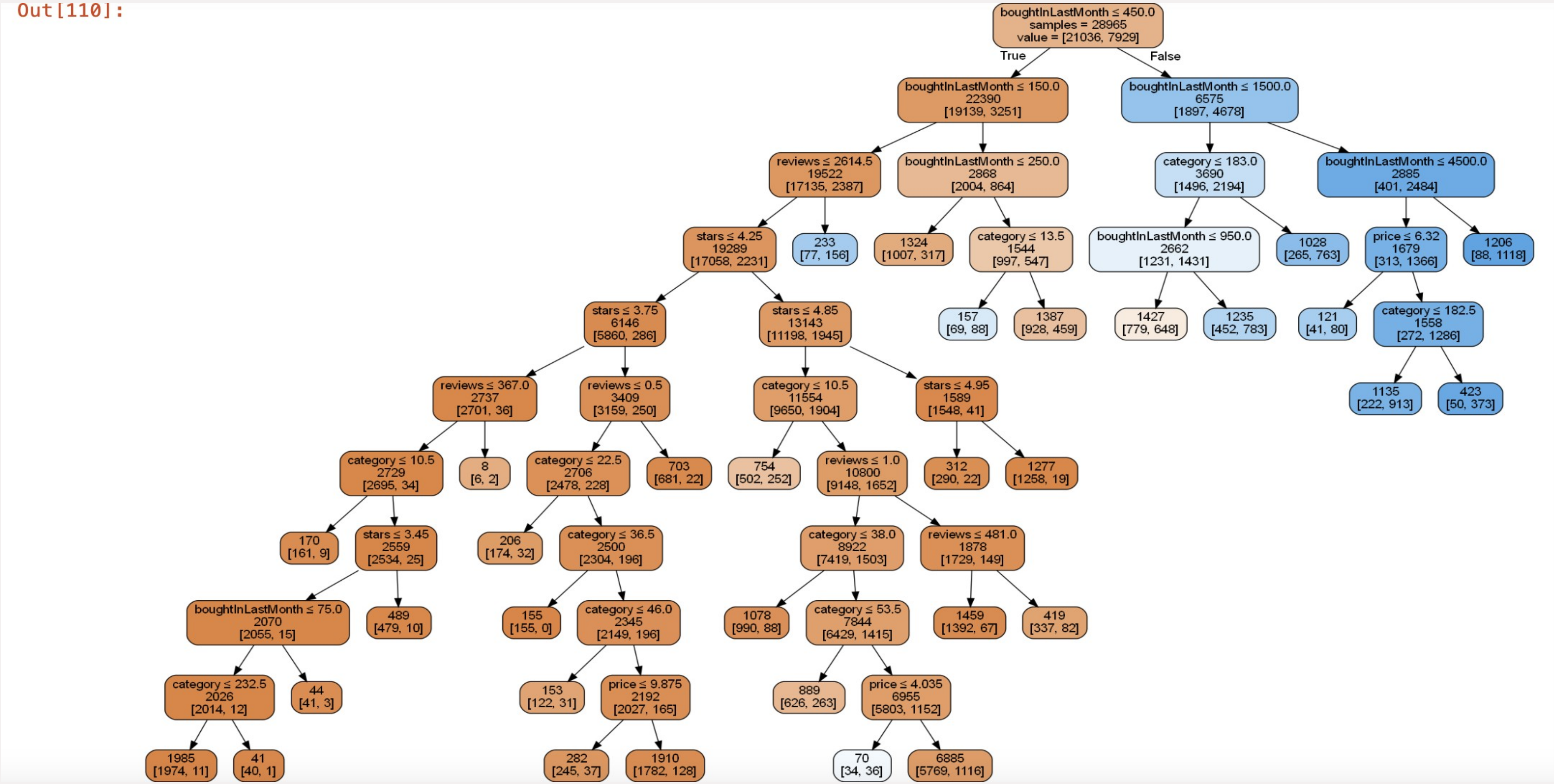
- A classification model was developed using decision tree classifiers and was fed with the same data as KNN
- Trees with various depths has been developed to check the accuracy on validation data, Later using grid search the parameters for the best tree was identified and used in model.
- Accuracy from validation data is 82.7% and this model was improved by random forest method which then gave an accuracy of 83.3%



```
classificationSummary(valid_y, rf.predict(valid_X))
```

```
Confusion Matrix (Accuracy 0.8338)

       Prediction
Actual     0     1
     0 12635  1329
     1  1880  3466
```


Out[110]:
boughtInLastMonth ≤ 450.0 samples = 28965 value = [21036, 7929]
True
False
boughtInLastMonth ≤ 150.0 22390 [19139, 3251]
boughtInLastMonth ≤ 1500.0 6575 [1897, 4678]
reviews ≤ 2614.5 19522 [17135, 2387]
boughtInLastMonth ≤ 250.0 2868 [2004, 864]
category ≤ 183.0 3690 [1496, 2194]
boughtInLastMonth ≤ 4500.0 2885 [401, 2484]
stars ≤ 4.25 19289 [17058, 2231]
233 [77, 156]
1324 [1007, 317]
category ≤ 13.5 1544 [997, 547]
boughtInLastMonth ≤ 950.0 2662 [1231, 1431]
1028 [265, 763]
price ≤ 6.32 1679 [313, 1366]
1206 [88, 1118]
stars ≤ 3.75 6146 [5860, 286]
stars ≤ 4.85 13143 [11198, 1945]
157 [69, 88]
1387 [928, 459]
1427 [779, 648]
1235 [452, 783]
121 [41, 80]
category ≤ 182.5 1558 [272, 1286]
reviews ≤ 367.0 2737 [2701, 36]
reviews ≤ 0.5 3409 [3159, 250]
category ≤ 10.5 11554 [9650, 1904]
stars ≤ 4.95 1589 [1548, 41]
1135 [222, 913]
423 [50, 373]
category ≤ 10.5 2729 [2695, 34]
8 [6, 2]
category ≤ 22.5 2706 [2478, 228]
703 [681, 22]
754 [502, 252]
reviews ≤ 1.0 10800 [9148, 1652]
312 [290, 22]
1277 [1258, 19]
170 [161, 9]
stars ≤ 3.45 2559 [2534, 25]
206 [174, 32]
category ≤ 36.5 2500 [2304, 196]
category ≤ 38.0 8922 [7419, 1503]
reviews ≤ 481.0 1878 [1729, 149]
boughtInLastMonth ≤ 75.0 2070 [2055, 15]
489 [479, 10]
155 [155, 0]
category ≤ 46.0 2345 [2149, 196]
1078 [990, 88]
category ≤ 53.5 7844 [6429, 1415]
1459 [1392, 67]
419 [337, 82]
category ≤ 232.5 2026 [2014, 12]
44 [41, 3]
153 [122, 31]
price ≤ 9.875 2192 [2027, 165]
889 [626, 263]
price ≤ 4.035 6955 [5803, 1152]
1985 [1974, 11]
41 [40, 1]
282 [245, 37]
1910 [1782, 128]
70 [34, 36]
6885 [5769, 1116]

If boughtinlastmonth >4500 , then surely that product isBestSeller Product.

Technically, Classification tree(83.38%) has more accuracy compared with KNN(83.36).

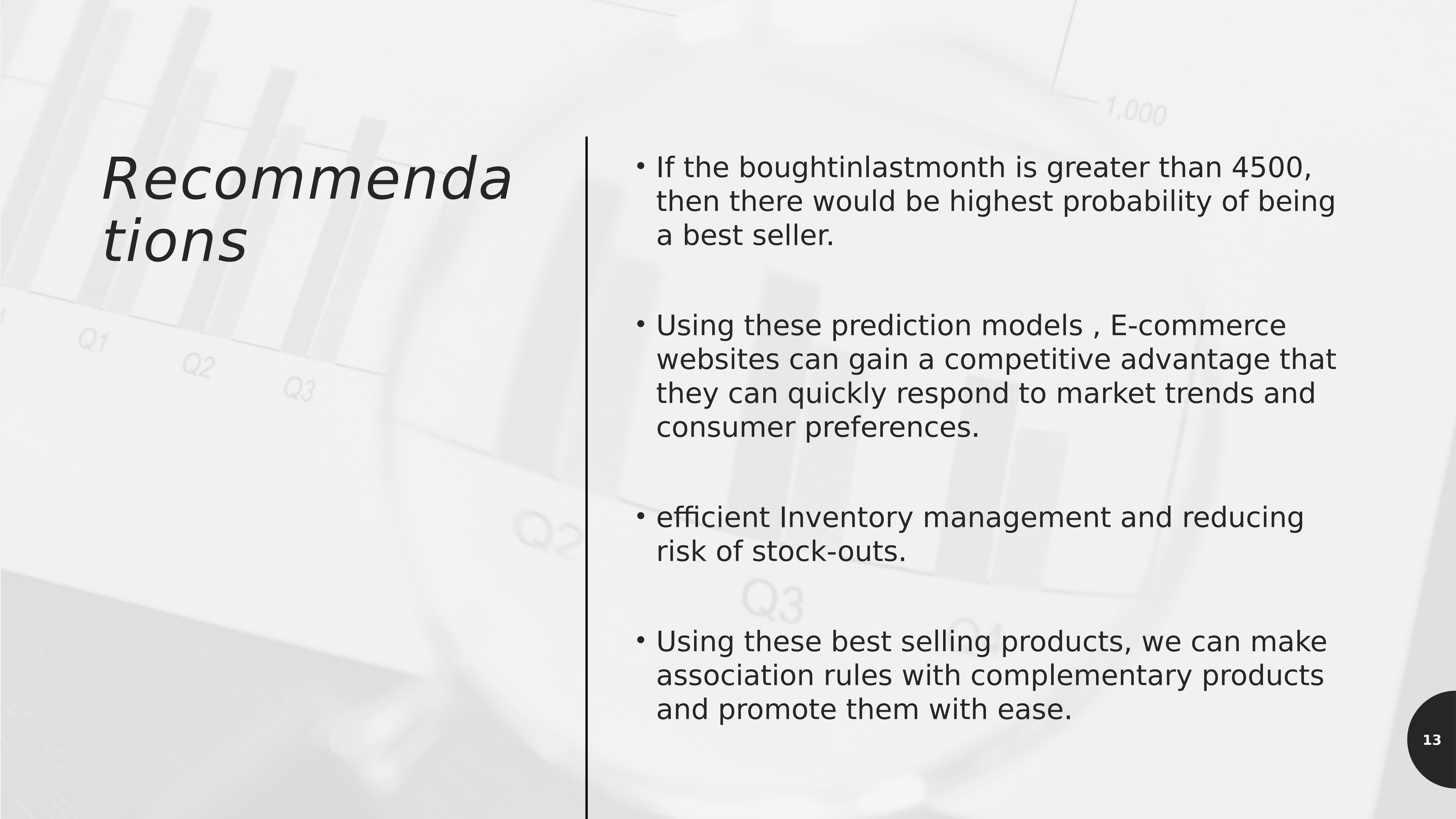
However , Classification tree is more complex though we have achieved utmost accuracy which is hard for interpretation.

As we see , both the models are similarly accurate , Hence we can choose KNN being a simplest model comparatively .

The choice between KNN and decision trees should be based on factors such as the nature of the data, the interpretability requirements, and the computational resources available.

We are trading off 0.02% of accuracy by choosing simpler model KNN for ’isBestseller’ prediction.

Model Comparison



Recommendations

- If the boughtinlastmonth is greater than 4500, then there would be highest probability of being a best seller.
- Using these prediction models , E-commerce websites can gain a competitive advantage that they can quickly respond to market trends and consumer preferences.
- efficient Inventory management and reducing risk of stock-outs.
- Using these best selling products, we can make association rules with complementary products and promote them with ease.