TEAM 1:
ECOMMERCE CONSUMER BEHAVIOR

ALEXA BEZZONE

JAMES BENASULI

LIVIA MIYABARA

Ecommerce Consumer Behavior: Topics covered

1. Customer segmentation based on product categories
2. Items frequently bought together
3. Map main topics from customer reviews

Amazon Reviews database
(Project Vine)

1 Can we identify customer segments based on the purchased product categories to better target marketing campaigns?

- APPAREL
- FURNITURE
- MUSIC
- WATCHES
- PERSONAL CARE
- OFFICE PRODUCTS
- VIDEO
- VIDEO GAMES

Amazon Reviews pds → Pyspark → Raw data 17M rows → Pyspark → Postgres → Cleanup data 700K rows → Amazon RDS → Jupyter notebook → Model: 500k+ rows (top customers)

# of products bought by product category and customer

customer_id	apparel	furniture	music	office_products	personal_care_appliances	video_games	videos	watches
10018	NaN	NaN	NaN	NaN	NaN	4.0	NaN	NaN
10206	NaN	NaN	NaN	NaN	NaN	8.0	NaN	NaN
10468	NaN	NaN	NaN	8.0	NaN	NaN	NaN	3.0

Create marketing campaigns to target specific customers segments based on product category to increase sales and conversion rates

1 Method: customer segmentation based on product categories

K-MEANS CLUSTER ANALYSIS

Unsupervised Machine Learning Model

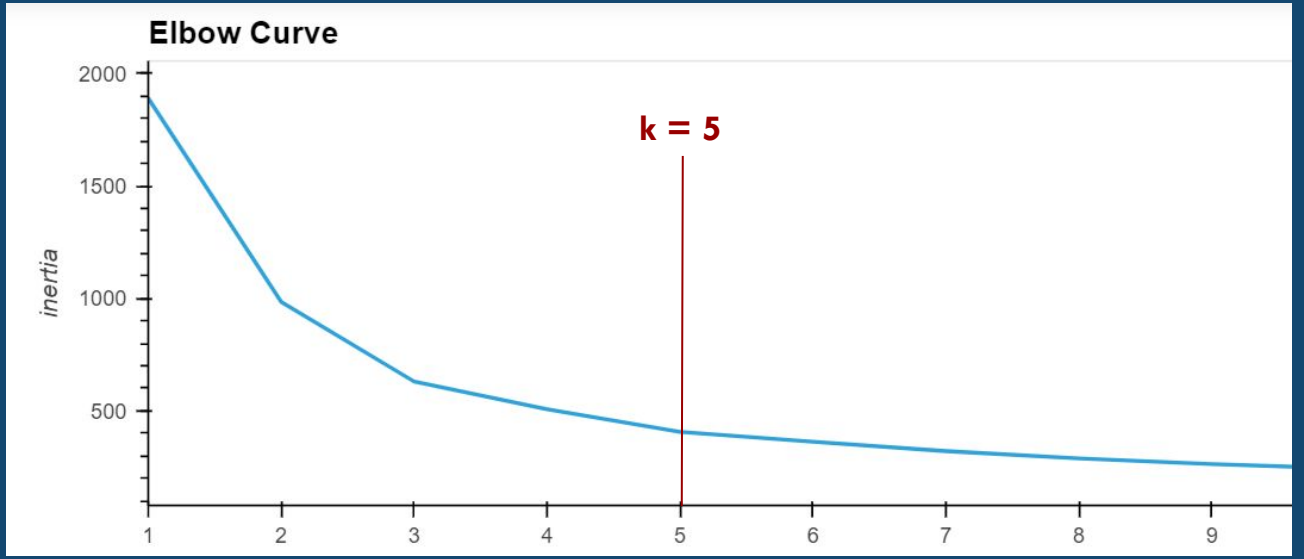
K = # of clusters (groups)

Groups data into groups based on similarity (centroid distance)

PREPARE DATA FOR MODEL

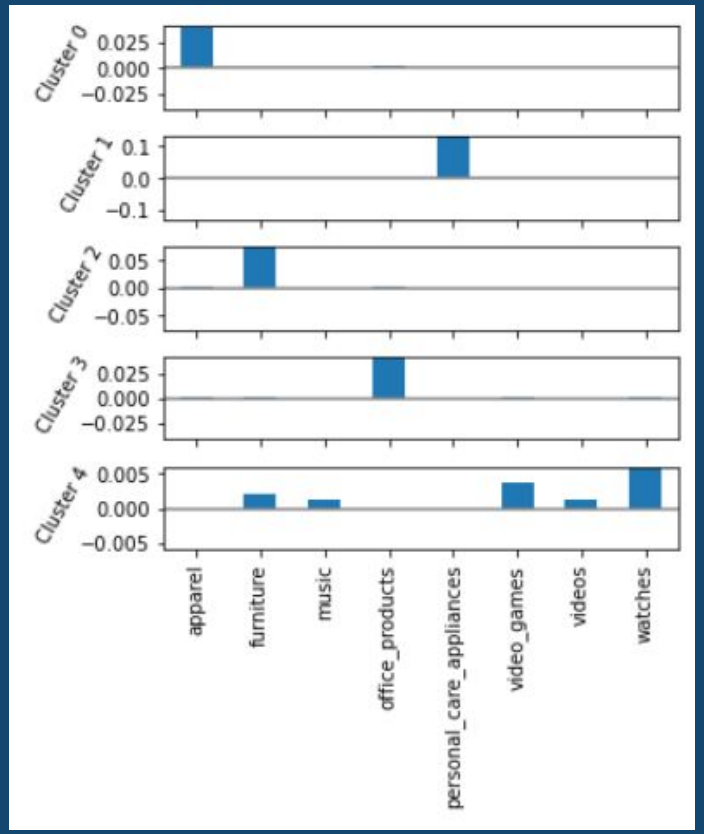
- Scale data (MinMaxScaler)
- Remove outliers from each product category, Kmeans sensitive to outliers

ELBOW CURVE: define k



1 Results: customer segmentation based on product categories

INTERPRET CLUSTERS



Cluster 0 = apparel
Cluster 1 = personal care appliances
Cluster 2 = furniture
Cluster 3 = office products
Cluster 4 = multi-category

NUMBER OF CUSTOMERS BY PRODUCT SEGMENT

- 44% of customers buying products from multi-categories, largest segment
- Cluster 2 (furniture): priority for marketing campaign since there are current customers from Cluster 4 that buys furniture and other products
- Create additional campaigns to cluster 0, 1 and 3 … give discount in other product categories to incentivize product mix and sales

Ecommerce Consumer Behavior: Topics covered

1

Customer segmentation based on product categories

2

Items frequently bought together

3

Map main topics from customer reviews

2 Can we identify which products a customer will **most likely purchase together?**

APPAREL
MUSIC
PERSONAL CARE
VIDEO
FURNITURE
WATCHES
OFFICE PRODUCTS
VIDEO GAMES

Amazon Reviews pds → Pyspark → Raw data 17M rows → Pyspark → Postgres → Cleanup data 12M rows → Amazon RDS → Jupyter notebook → Model: 700k+ rows

customer_id	product_id	quantity
25551507	0788812807	1
25551507	6302320402	1
31354506	6301442733	1

product_id	B0000532OT	B0000532OV	B0000537JP	B0000537JQ	B000068PBJ	B00008J1ZZ	B00009RB1I	B0000YS1BG
customer_id								
10470	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11344	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Help Amazon increase cross selling for revenue growth

2 Method: items frequently bought together

APRIORI ALGORITHM

Maps items purchased together in a single transaction

Find frequent items

Association rules to find relationships between large databases

Confidence shows % of customers who bought those items together

antecedents	consequents	confidence
(B007ZRZWOA, B001BKVWYG)	(B001TCHDPS)	0.815385
(B003GAMPWM, B009G7ZYPY, B00F0O8SZK)	(B005G618JU)	0.800000

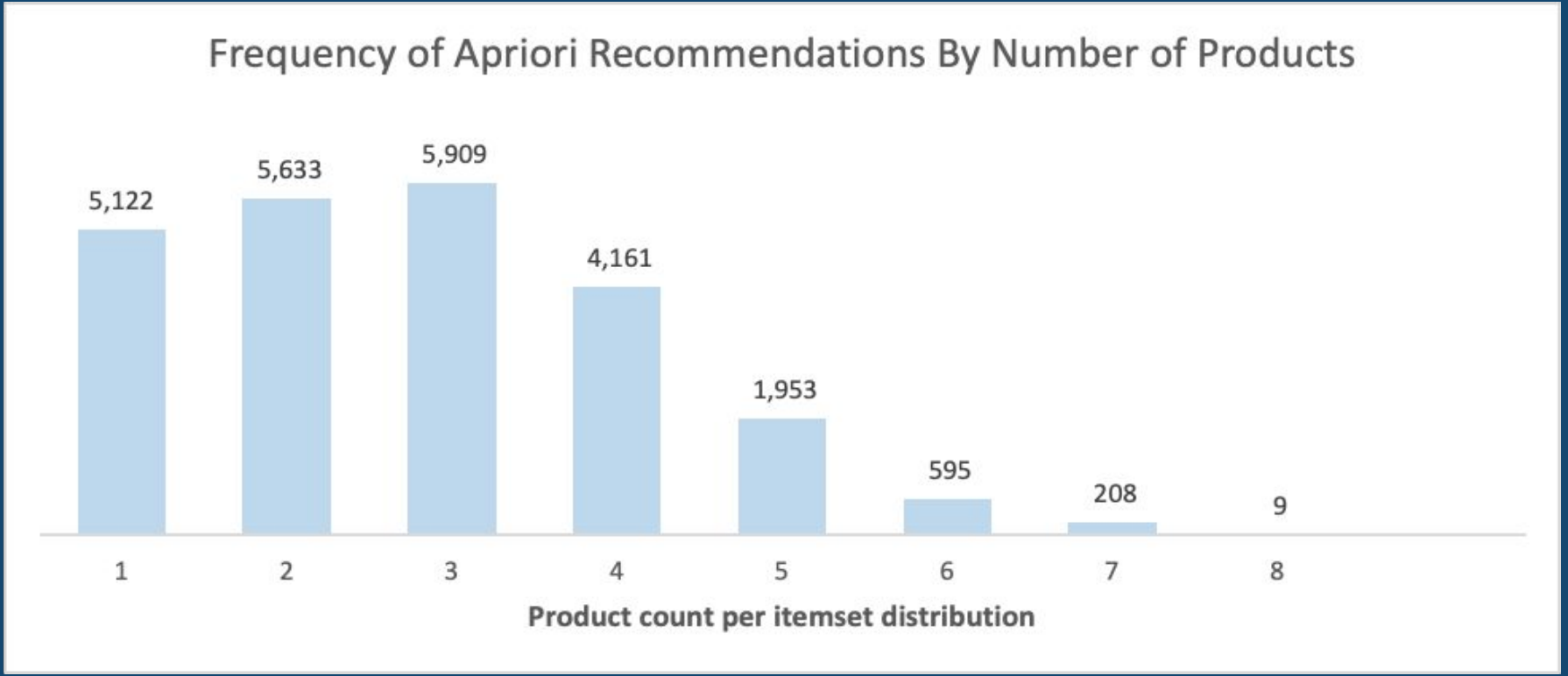
B001BKVWYG: Cold Fact, Sixto Rodriguez (CD)

B001TCHDPS: Coming from Reality, Sixto Rodriguez (CD)

B007ZRZWOA: Searching for Sugar Man, Rodriguez (CD)

≈

2 Visualization: Frequency of Apriori Recommendations By Number of Products

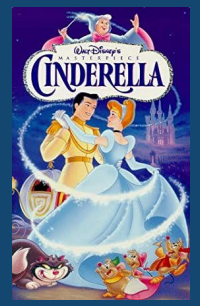


- This histogram depicts the frequency of Apriori associations by itemsets
- Highest number of instances are for 3 product itemsets with about 5,909 associations
- Lowest number of associations are 8 product itemsets with only 9 associations
- Total number of recommendations the Apriori analysis gathered was 23,590

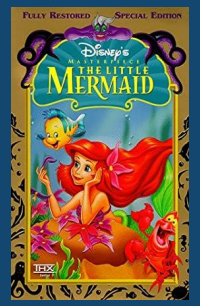
2 Results: items frequently bought together

VIDEOS APRIORI RESULT EXAMPLE

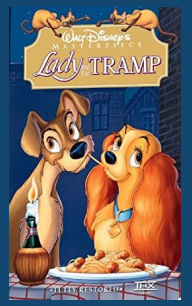
antecedents	consequents	antecedent support	consequent support	support	confidence
(0788802194, 0788812408, 0788812807)	(6304401132, 0788805533, 0788806270)	0.000165	0.000165	0.000165	1.000000



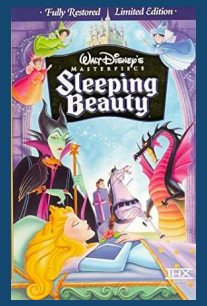
0788802194



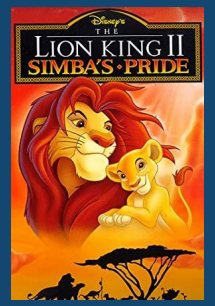
0788812408



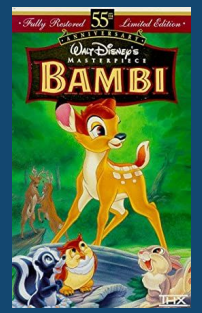
0788812807



6304401132

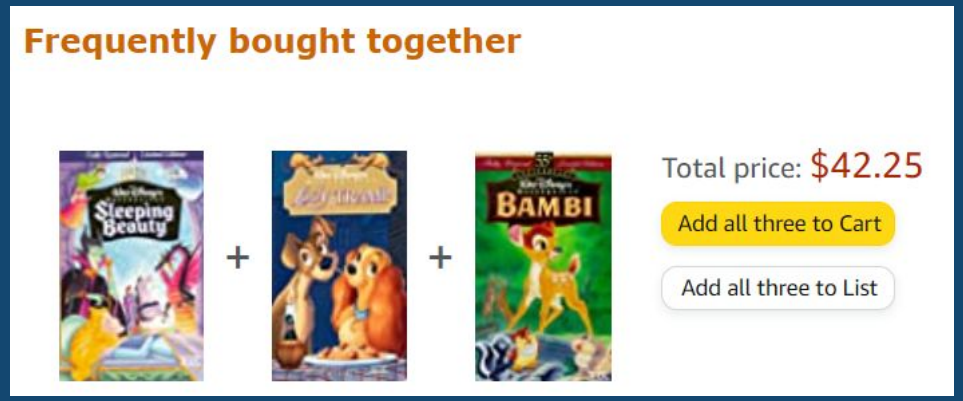


0788805533



0788806270

AMAZON PRODUCT RECOMMENDATION



HIGHLIGHTS

- Higher confidence in same product category instead of various categories (as per data exploration)
- Videos and music categories are the only categories with higher confidence outputs (> 60%)
- Low risk of showing product recommendations with low confidence level … only impact is UX in website

Ecommerce Consumer Behavior: Topics covered

1

Customer segmentation based on product categories

2

Items frequently bought together

3

Map main topics from customer reviews

3

Can we extract key topics within product reviews to help companies analyze customer feedback?

Product Reviews for a specific product:

- **Product ID: B000M0MJU2**
- **Product name: Intex Raised Downy Airbed with Built-in Electric Pump, Queen, Bed Height 22"**

~5k rows

customer_id	review_id	star_rating	review_headline	review_body
51982153	R1DZ76NBD2TX55	5	my wife and i had to pick one of these up over...	my wife and i had to pick one of these up over...
44662747	R3G4HN08IK8Q5W	5	this is big and comfortable it inflatesdeflat...	this is big and comfortable it inflatesdeflat...
17097525	R1S3TBZK71L487	1	horrible it was so comfortable for the first f...	horrible it was so comfortable for the first f...

Easily and readily extract key topics within customer reviews to identify positive or negative trends, drive product improvements and better customer service

3 Method: customer reviews topic analysis

LATENT DIRICHLET ALLOCATION (LDA)

Unsupervised Machine Learning Model

User defines # of topics

Each topic represents a group of words

LDA maps all documents to the topics and frequent group of words

DATA CLEAN UP FOR THE MODEL

- Remove special characters
- Exclude stop words (I, you, it)
- Take out non value added words (i.e. mattress)
- Considers adjectives only

Spit analysis between good (5-stars) and bad reviews (1-star) to make it easier to identify positive and negative feedback

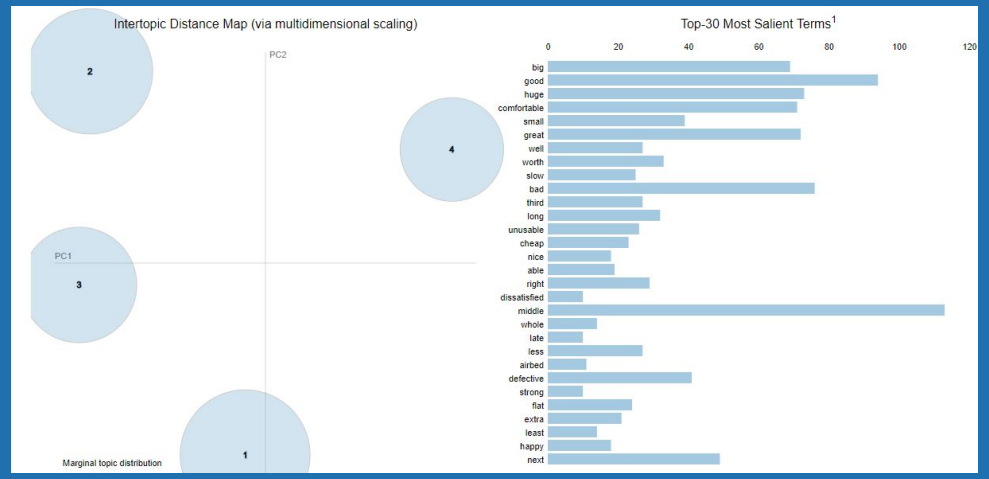
Adjust number of topics based on overlaps in words frequency (pyLDAvis)

3 Results: customer reviews topic analysis

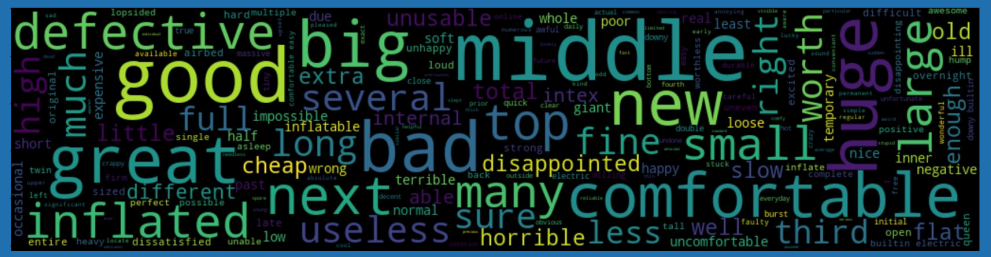
LINK TO DASHBOARD

★☆☆☆☆

TOPIC VISUALIZATION

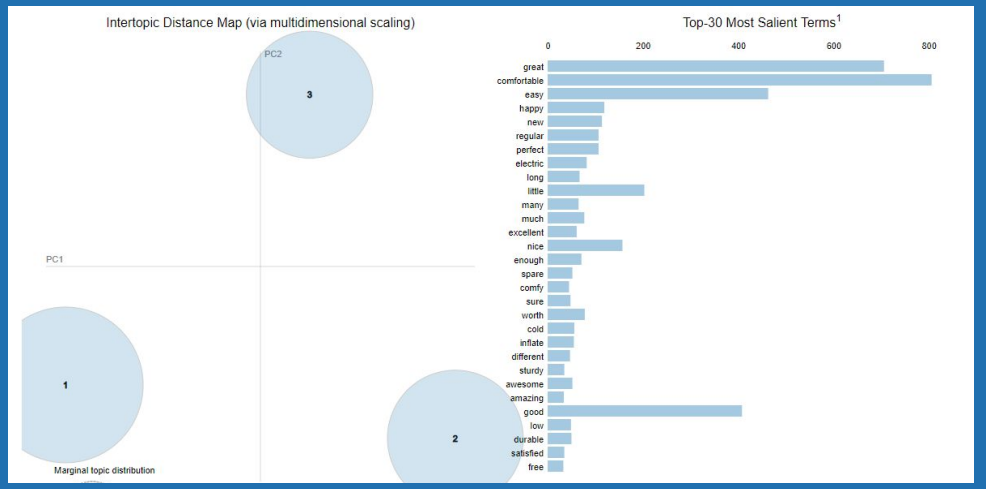


WORD CLOUD

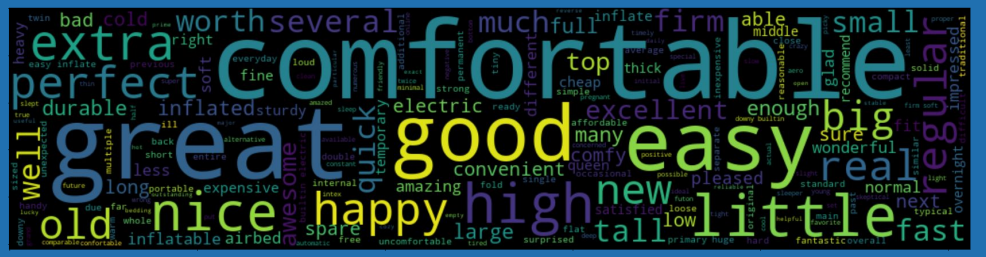


★★★★★

TOPIC VISUALIZATION



WORD CLOUD



3 Results: customer reviews topic analysis

Star	Topic	Frequent words	Topic Interpretation
	Topic 1	Good, big, middle, defective, inflated, flat, comfortable, horrible	Complains about **air mattress deflating**, not comfortable
	Topic 2	Small, great, huge, unusable, comfortable, cheap, disappointed	Complains about **size of air mattress**, not comfortable
	Topic 3	Bad, huge, middle, worth, happy	Overall complaints about **product dissatisfaction**
	Topic 4	Comfortable, middle, long, slow, bad, useless	Complaints about product **not being comfortable**
	Topic 1	Great, good, easy, little, nice, high, real	Product it's **good and easy to use**
	Topic 2	Comfortable, easy, good, perfect, electric, excellent	Product **comfortable, easy to use**
	Topic 3	Comfortable, happy, regular, worth, comfy	Product it's **comfortable**, customer satisfied with purchase

- **Similar words between topics for good and bad reviews with different connotation**
- **Analysis can be biased by person interpreting the outputs, hard to extract meaning of topics**
- **Hard to identify different topics, similar words and feedback, recommended only for a superficial analysis**
- **Need to improve corpus to combine words for more accurate analysis**

Recommendations for future analysis

1

Items frequently bought together

- Need more processing power, since it limited the number of products in analysis
- Used customer reviews as a proxy to customer purchases, for next analysis it will be good to access real sales data

2

Customer segmentation based on product categories

- Used customer reviews as a proxy to customer purchases, for next analysis it will be good to access real sales data
- Create A/B testing on website within customer segments to see if targeted marketing campaigns based on K-means customer segmentation can increase sales

3

Map main topics from customer reviews

- Need to improve corpus to combine words for more accurate analysis
- Expand analysis to a larger products set to test and validate improved corpus
- Create an analysis based on specific time frames to evaluate if different topics emerge

Technologies, languages, tools, and algorithms used throughout the project

Languages:

- Python
- Postgres SQL
- HTML
- Javascript

Tools:

- PgAdmin
- Amazon RDS
- Pyspark API
- Google colab
- Jupyter notebook
- Github

Technologies:

- collections
- hvplot.pandas
- matplotlib.pyplot
- mlxtend.frequent_patterns
- nltk
- nltk.corpus
- nltk.stem
- nltk.tokenize
- numpy
- os
- pandas
- PIL
- pyLDAvis
- pyLDAvis.gensim_models
- Pyspark
- plotly.express
- re
- seaborn
- sklearn.cluster
- sklearn.preprocessing
- spacy
- sqlalchemy
- wordcloud

APPENDIX

Research Questions

Can we predict which products a customer will **most likely purchase together** within various product segments?

Can we **identify customer segments** based on the purchased **product categories** to better target marketing campaigns?

Can we extract key topics within product reviews to help companies **analyze and interpret customer feedback**?

Why these topics?

- Data analysis is key for strategic and well-informed decision making
- Big data allows e-commerce businesses to understand customers better through customer behavior analysis
- Helps target specific customers segments to upsell products, increase conversion rates and grow sales
- Better customer segmentation to improve targeted marketing campaigns and increase sales
- Product reviews is a great source of customer feedback and one of the main drivers for conversion rates, developing an automated way to process them can help drive product enhancements and accelerate decision making

Anything the team would have done differently?

- Defining weekly roles did not hindered optimal work flows, team members can decide how to best allocate tasks
- Refer to last rubric (segment 4) and work backwards
- Allow team members to determine best github structure (e.g. have branches based on work streams and not user names)

Question 1: Can we predict which products a customer will most likely purchase together within various product segments?

Goal of Question 1: Help Amazon identify products frequently bought together by customers to increase conversion rates and revenues (cross sell) by analyzing Amazon Marketplace segment data.

Customer ID

Product ID

Machine Learning Plans

- **Association Data Mining**
  - **Apriori Algorithm:**
    - **Utilize Apriori Algorithm to populate items that are most frequently bought together within various product segments.**

Source: https://www.hackerearth.com/blog/developers/beginners-tutorial-apriori-algorithm-data-mining-r-implementation/

Data Summary

Team will be using Amazon.com product segment data from S3

- **Data Source:** Amazon S3
- **Datasets:** 8 different product segments
  - Apparel
  - Furniture
  - Music
  - Office Products
  - Personal Care Appliances
  - Video Games
  - Videos
  - Watches
- **Number of Columns:** 15 (raw); 3 (after load in postgres)
- **Type:** Structured

Extracted database sample to Postgres

	customer_id integer	review_id [PK] character varying	product_id character varying
1	24509695	R3VR960AHLFKDV	B004HB5E0E
2	34731776	R16LGVMFKIUT0G	B0042TNMMS
3	1272331	R1AIMEEPYHMOE4	B0030MPBZ4
4	45284262	R1892CCSZWZ9SR	B005G02ESA
5	18311821	RLB33HJBXHZHU	B00AVUQQGQ
6	42943632	R1VGTZ94DBAD6A	B00CFY20GQ
7	43157304	R168KF82ICSOHD	B00FKC48QA
8	51918480	R20DIYIJ0OCMOG	B00N9IAL9K
9	14522766	RD46RNVOHNZSC	B001T4XU1C
10	43054112	R2JDOCETTM3AXS	B002HRFLBC

Raw Data Frame Example

marketplace	customer_id	review_id	product_id	product_parent	product_title	product_category	star_rating	helpful_votes	total_votes	vine	verified_purchase	review_headline	review_body	review_date
US	24509695	R3VR960AHLFKDV	B004HB5E0E	488241329	Shoal Creek Compu...	Furniture	4	0	0	N	Y	... desk is very ...	This desk is very...	2015-08-31
US	34731776	R16LGVMFKIUT0G	B0042TNMMS	205864445	Dorel Home Produc...	Furniture	5	0	0	N	Y	Five Stars	Great item	2015-08-31

Data Processing Plan

EXTRACT

Databases varies from 800k to 5M rows, Pyspark faster than Pandas

8 different tables,
all with same schema

TRANSFORM

- Load Amazon product segment into Spark DataFrame
- Perform preliminary cleaning
  - Drop unnecessary columns
  - Filter data to present only verified purchases
  - Drop the verified purchased column after filtering
- Create Apriori Analysis dataframe
  - Drop additional unnecessary columns in preparation for Apriori Analysis
- Repeat this process with various product segments

LOAD

- Download PostgresSQL driver that will allow PySpark to interact with PostgresSQL
- Configure settings for PostgresSQL
- Write the cleaned table into PostgresSQL
- Merge tables in postgres
- Connect jupyter notebook to RDS and load dataframe

Data Processing Plan: Extract

The team selected 8 different **product segments** from Amazon data:

- Music
- Video Games
- Videos
- Watches
- Furniture
- Office Products
- Personal Care Appliances
- Apparel

Each segment has has the same data schema as example below:

Raw data frame Example

marketplace	customer_id	review_id	product_id	product_parent	product_title	product_category	star_rating	helpful_votes	total_votes	vine	verified_purchase	review_headline	review_body	review_date
US	24509695	R3VR960AHLFKDV	B004HB5E0E	488241329	Shoal Creek Compu...	Furniture	4	0	0	N	Y	... desk is very ...	This desk is very...	2015-08-31
US	34731776	R16LGVMFKIUT0G	B0042TNMMS	205864445	Dorel Home Produc...	Furniture	5	0	0	N	Y	Five Stars	Great item	2015-08-31
US	1272331	R1AIMEEPYHMOE4	B0030MPBZ4	124663823	Bathroom Vanity T...	Furniture	5	1	1	N	Y	Five Stars	Perfect fit for m...	2015-08-31
US	45284262	R1892CCSZWZ9SR	B005G02ESA	382367578	Sleep Master Ulti...	Furniture	3	0	0	N	Y	Good enough	We use this on a ...	2015-08-31
US	30003523	R285P679YWVKD1	B005JS8AUA	309497463	1 1/4" GashGuards...	Furniture	3	0	0	N	N	Gash Gards for da...	The product is fi...	2015-08-31

Data types

```
|-- marketplace: string (nullable = true)
|-- customer_id: integer (nullable = true)
|-- review_id: string (nullable = true)
|-- product_id: string (nullable = true)
|-- product_parent: integer (nullable = true)
|-- product_title: string (nullable = true)
|-- product_category: string (nullable = true)
|-- star_rating: integer (nullable = true)
|-- helpful_votes: integer (nullable = true)
|-- total_votes: integer (nullable = true)
|-- vine: string (nullable = true)
|-- verified_purchase: string (nullable = true)
|-- review_headline: string (nullable = true)
|-- review_body: string (nullable = true)
|-- review_date: string (nullable = true)
```

PySpark was selected due to faster processing capabilities

Data Processing Plan: Transform

1. Load Amazon product segment into PySpark DataFrame
2. Perform preliminary cleaning
    - Drop unnecessary columns
        - Columns: ‘marketplace’, ‘product_parent’, ‘vine’, ‘review_headline’, ‘review_headline’, ‘review_body’, ‘review_date’
    - Filter data to present only verified purchases
        - verified_purchase = ‘Y’
    - Drop the verified purchased column after filtering
        - Column: verified_purchase
3. Create Apriori Analysis dataframe
    - Drop additional unnecessary columns in preparation for Apriori Analysis
        - Columns: ‘review_id’, ‘product_id’, ‘product_title’, ‘star_rating’, ‘helpful_votes’, ‘total_votes’
4. Repeat this process with various product segments.

Data Processing Plan: Load

- Download Postgres driver that will allow Spark to interact with PostgresSQL
- Configure settings for PostgresSQL
- Write the cleaned table into PostgresSQL.
  - Write cleaned product segment table that is prepped for Apriori Analysis into PostgresSQL

Example of Cleaned Apriori Table Ready for PostgresSQL

customer_id	review_id	product_id
10140119	R3LI5TRP3YIDQL	B00TXH4OLC
27664622	R3LGC3EKEG84PX	B00B6QXN6U
45946560	R9PYL3OYH55QY	B001GCZXW6
15146326	R3PWBAWUS4NT0Q	B000003EK6
16794688	R15LYP3O51UU9E	B00N1F0BKK
32203364	R1AD7L0CC3DSRI	B00V7KAO7Q
1194276	R32FE8Y45QV434	B000094Q4P
45813052	R3NM4MZ4XWL43Q	B00JMK0P1I
12795687	R3H4FXX6Q7I37D	B008OW1S3O
36673840	R30L5PET7LFFDC	B00VI2L3L4
49453576	REFRE1LEKLAF	B0000041EV
3285047	R3JTJ5EQN74E9H	B00005YW4H
24471201	R1W2F091LCOAW5	B00Q9KEZV0
28049396	RYUMFQRRB1FNM	B00GFXRKHW
41137196	RHCS6VVXWV3Q3	B004L3AQ10

Apriori Algorithm

Definition

Apriori algorithm is a classical algorithm in **data mining.** It is u**sed for mining frequent itemsets and relevant association rules.**

The parameters “support” and “confidence” are utilized.

Support = items’ frequency of occurrence
Confidence = conditional probability

How it works?

Items in a transaction = item set

Algorithm identify frequent, individual items (items with higher frequency than the support)

Expands analysis to larger frequent itemsets

Example

Support* = 3, confidence = 80%

Transaction ID	Items
T1	I1, I2, I3, I4
T2	I2, I3
T3	I3, I4
T4	I2, I3, I4

- I2 with I3, confidence = 3/3 = 100%

Source: https://www.educative.io/edpresso/what-is-the-apriori-algorithm

Apriori Algorithm

Data preprocessing

- Connect to RDS
- Load pivot table (product_ids as headers)
- Set item association function

Feature selection

- Understand items brought by the same customer can increase conversion rates in ecomm and drive revenues growth (cross sell)
- Apriori algorithm is popular for this type of analysis
- Apriori gives confidence level of recommendation that helps data analysts decide the right threshold for website recommended products

Model selection

Apriori Algorithm

- Benefits
  - Most simple algorithm among association rule learning
  - Broadly adopted for basket analysis
  - Easy to understand and interpret
  - Exhaustive: finds all rules with confidence levels
- Limitations
  - Not good for small datasets
  - Takes time to run

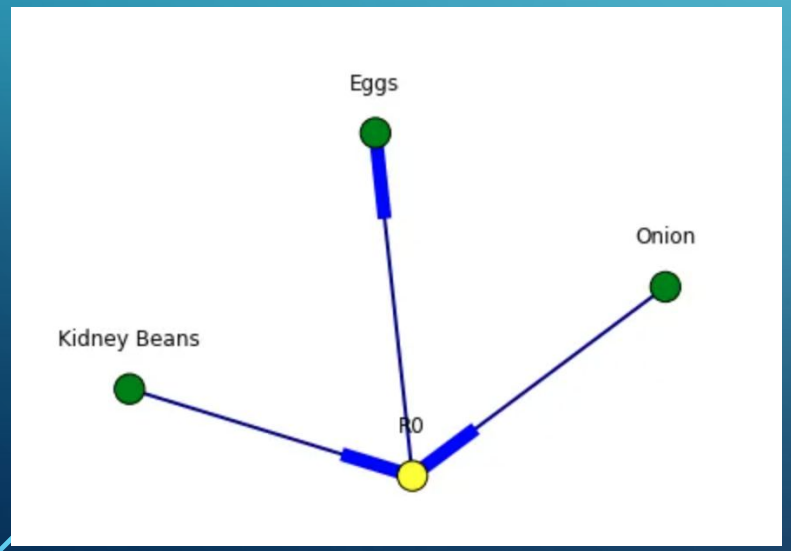
Question #1: Can we predict which products a customer will most likely purchase together within various product segments?

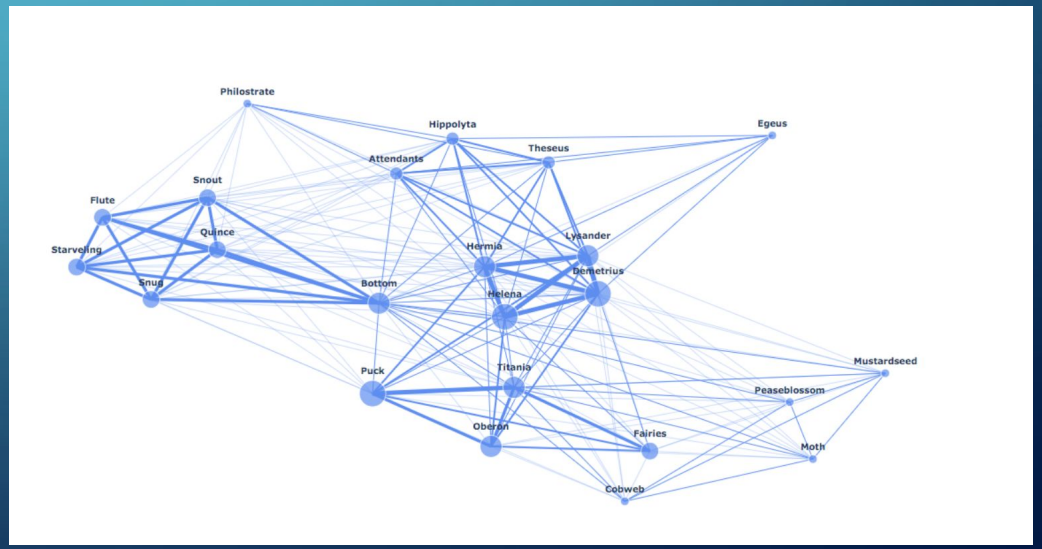
Visualization Plans:

- NetworkX module for charting association rules: https://intelligentonlinetools.com/blog/2018/02/10/how-to-create-data-visualization-for-association-rules-in-data-mining/
  - NetworkX python module represents association rules through a diagram.
  - Each diagram represents one rule association and provides arrows that connect the associative products together, please see image below for an example.
  - The root node (R0) represents one association rule with incoming and outcoming edges attached to the root. The diagram represents the association between products.

Visualization Tools:

- Python's NetworkX Library and Plotly
- https://towardsdatascience.com/tutorial-network-visualization-basics-with-networkx-and-plotly-and-a-little-nlp-57c9bbb55bb9





Question 2: Can we identify customer segments based on the purchased product categories to better target marketing campaigns?

Goal of Question 2: Help Amazon learn and predict which customers are more likely to purchase products within product segments. Identifying this trend can help Amazon target advertisements to specific customers within certain product segments in efforts to increase sales and revenues.

Customer ID

Product Segment

Machine Learning Plans

- Unsupervised Machine Learning
  - K-Means Cluster Analysis
    - A K-Means Cluster Analysis model will be performed to cluster customers into various product types based on purchasing behavior within various product segments.

Data Summary

Team will be using Amazon.com product segment data from S3

- **Data Source:** Amazon S3
- **Datasets:** 8 different product segments
  - Music
  - Video Games
  - Videos
  - Watches
  - Furniture
  - Office Products
  - Personal Care Appliances
  - Apparel
- **Number of Columns:** 15
- **Type:** Structured

Raw Data Frame Example

marketplace	customer_id	review_id	product_id	product_parent	product_title	product_category	star_rating	helpful_votes	total_votes	vine	verified_purchase	review_headline	review_body	review_date
US	24509695	R3VR960AHLFKDV	B004HB5E0E	488241329	Shoal Creek Compu...	Furniture	4	0	0	N	Y	... desk is very ...	This desk is very...	2015-08-31
US	34731776	R16LGVMFKIUT0G	B0042TNMMS	205864445	Dorel Home Produc...	Furniture	5	0	0	N	Y	Five Stars	Great item	2015-08-31

Data Processing Plan

EXTRACT

PySpark

Databases varies from 800k to 15M rows, Pyspark faster than Pandas

8 different tables,
all with same schema

TRANSFORM

- Load Amazon product segment into PySpark dataFrame
- Perform preliminary cleaning
  - Drop unnecessary columns
  - Filter data to present only verified purchases
  - Drop the verified purchased column after filtering.
- Create Segmentation Analysis dataframe
  - Drop additional unnecessary columns in preparation for K-Means Cluster Analysis
  - Group the columns by customer ID and product category
  - Filter the top results
- Repeat this process with various product segments.

LOAD

- Download Postgres driver that will allow PySpark to interact with PostgresSQL
- Configure settings for PostgresSQL
- Write the cleaned table into PostgresSQL.

Data Processing Plan: Extract

The team selected 8 different **product segments** from Amazon data:

- Music
- Video Games
- Videos
- Watches
- Furniture
- Office Products
- Personal Care Appliances
- Apparel

Each segment has the same data schema as example below:

Extracted database sample to Postgres

	customer_id [PK] integer	furniture integer
1	45212655	33
2	35178127	27
3	20845991	25
4	36020793	25
5	12609448	24
6	40418760	22

Raw Data Frame Example

marketplace	customer_id	review_id	product_id	product_parent	product_title	product_category	star_rating	helpful_votes	total_votes	vine	verified_purchase	review_headline	review_body	review_date
US	24509695	R3VR960AHLFKDV	B004HB5E0E	488241329	Shoal Creek Compu...	Furniture	4	0	0	N	Y	... desk is very ...	This desk is very...	2015-08-31
US	34731776	R16LGVMFKIUT0G	B0042TNMMS	205864445	Dorel Home Produc...	Furniture	5	0	0	N	Y	Five Stars	Great item	2015-08-31
US	1272331	R1AIMEEPYHMOE4	B0030MPBZ4	124663823	Bathroom Vanity T...	Furniture	5	1	1	N	Y	Five Stars	Perfect fit for m...	2015-08-31
US	45284262	R1892CCSZWZ9SR	B005G02ESA	382367578	Sleep Master Ulti...	Furniture	3	0	0	N	Y	Good enough	We use this on a ...	2015-08-31
US	30003523	R285P679YWVKD1	B005JS8AUA	309497463	1 1/4" GashGuards...	Furniture	3	0	0	N	N	Gash Gards for da...	The product is fi...	2015-08-31

Data types

```
|-- marketplace: string (nullable = true)
|-- customer_id: integer (nullable = true)
|-- review_id: string (nullable = true)
|-- product_id: string (nullable = true)
|-- product_parent: integer (nullable = true)
|-- product_title: string (nullable = true)
|-- product_category: string (nullable = true)
|-- star_rating: integer (nullable = true)
|-- helpful_votes: integer (nullable = true)
|-- total_votes: integer (nullable = true)
|-- vine: string (nullable = true)
|-- verified_purchase: string (nullable = true)
|-- review_headline: string (nullable = true)
|-- review_body: string (nullable = true)
|-- review_date: string (nullable = true)
```

Pyspark was selected due to faster processing capabilities

Data Processing Plan: Transform

1. Load Amazon product segment into PySpark DataFrame
2. Perform preliminary cleaning
    - Drop unnecessary columns
        - Columns: ‘marketplace’, ‘product_parent’, ‘vine’, ‘review_headline’, ‘review_headline’, ‘review_body’, ‘review_date’
    - Filter data to present only verified purchases
        - verified_purchase = ‘Y’
    - Drop the verified purchased column after filtering
        - Column: verified_purchase
3. Create Segmentation Analysis dataframe
    - Drop additional unnecessary columns in preparation for K-Means Cluster Analysis
        - Columns: ‘review_id’, ‘product_id’, ‘product_title’, ‘star_rating’, ‘helpful_votes’, ‘total votes’
    - Group the data by customer id and product category
        - Group by ‘customer_id’ and count ‘product_category’ (# reviews = # transactions)
    - Filter the top results
        - Filter the data to show 100,000 rows displaying the top customer purchases within the chosen product category
4. Repeat this process with various product segments.

Data Processing Plan: Load

- Download Postgres driver that will allow PySpark to interact with PostgresSQL
- Configure settings for PostgresSQL
- Write the cleaned table into PostgresSQL.
  - Write cleaned product segment table that is prepped for K-Means Cluster Analysis into PostgresSQL

Example of Cleaned Segmentation Analysis Table Ready for PostgresSQL

customer_id	Music
29791894	1089
51184997	984
47423754	976
38192329	881
52562189	850
27364030	821
49939297	775
52469795	774
52467002	742
47883385	716
51228286	679
49877557	595
18116317	549
50910905	480
50135456	469
50345651	462
53075795	440
15536614	414
45772507	413
44861557	409

K-Means Clustering Machine Learning Model

Data preprocessing

- Connect to RDS
- Replace NaN values with zeros
- Drop columns (product categories) not needed
- Scale the data
- Customer_id as index

Feature selection

- Create customer segmentation based on product category
- Goal is to target specific segments based on categories purchased
- Unsupervised model, since there's no dependent variable (Y)
- No need to split and train the data for unsupervised model

Model selection

KMeans clustering

- Benefits
  - Simple to implement
  - Runs relatively quickly
  - Can scale large datasets
- Limitations
  - Sensitive to outliers
  - User defines # of clusters
  - Hard to interpret output since there's no Y variable

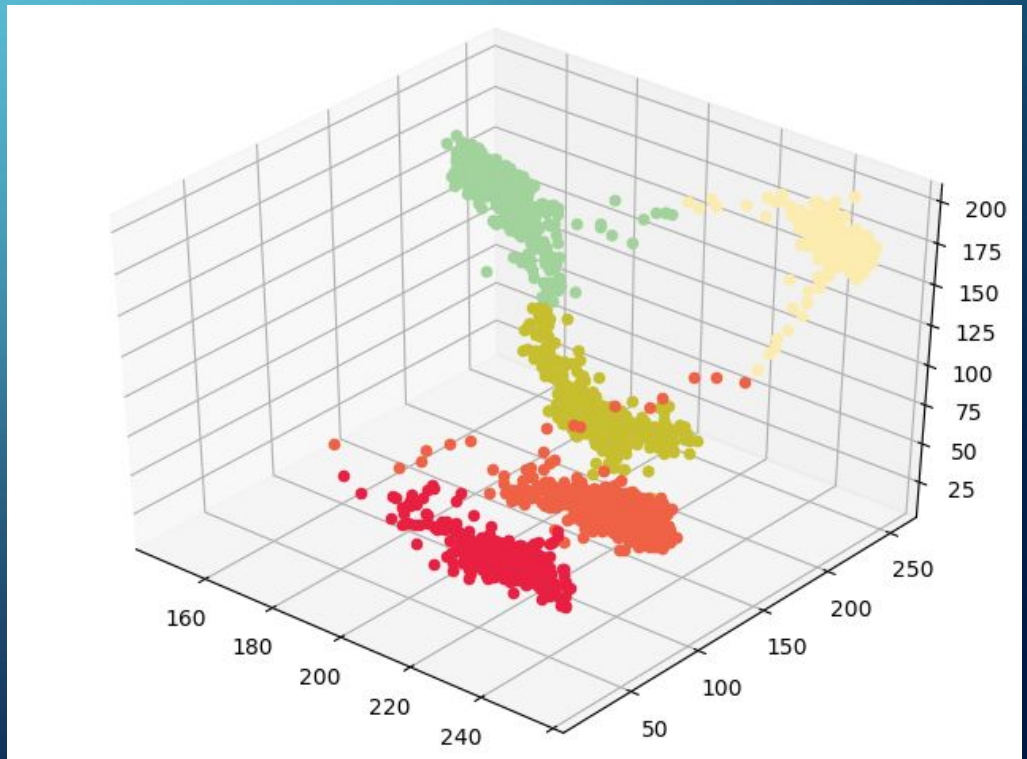
Question #2: Can we identify customer segments based on the purchased product categories to better target marketing campaigns

Visualization Plans:

- Visualize K-Means clusters through a 2D and 3D scatter plot visualizing 3 product segments.
- The goal of this scatter plot is for the viewer to easily identify customer segments based on the purchased product categories. For example, a user can visually see high customer activity within a product category or view low customer activity within a product category.
- https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

Visualization Tools:

- Using Python, the 2D graph will be created using Hvplot.pandas dependency.
- Using Python, the 3D scatter graph will be created using Plotly.express
- https://hvplot.holoviz.org/
- https://plotly.com/python/plotly-express/



Question 3: Can we extract key topics within product reviews to help companies analyze customer feedback?

Goal of Question 3: Help companies easily and readily extract key topics within product reviews to understand the customer feedback of their products. This will help companies identify positive or negative trends with their products and allow them to improve their products and customer service without having to read review by review.

Product Reviews for a specific product (B000M0MJU2; air mattress)

Machine Learning Plans

- Natural Language Processing
  - Topic Analysis
    - Use NLP to remove words that are not aggregating to analysis
    - Utilize Topic Analysis to enable companies to easily and readily view key topics from their product reviews in efforts to improve customer and product services.
    - Use Latent Dirichlet Allocation (LDA) machine learning model for topic discovery

Data Summary

Team will be using Amazon.com product segment data from S3

- **Data Source:** Amazon S3
- **Datasets:** 1 product segments
  - Outdoors
- **Number of Columns: 15**
- **Type: Structured**

Raw Data Frame Example

marketplace	customer_id	review_id	product_id	product_parent	product_title	product_category	star_rating	helpful_votes	total_votes	vine	verified_purchase	review_headline	review_body	review_date
US	24509695	R3VR960AHLFKDV	B004HB5E0E	488241329	Shoal Creek Compu...	Furniture	4	0	0	N	Y	... desk is very ...	This desk is very...	2015-08-31
US	34731776	R16LGVMFKIUT0G	B0042TNMMS	205864445	Dorel Home Produc...	Furniture	5	0	0	N	Y	Five Stars	Great item	2015-08-31

Extracted database sample to Postgres

	customer_id integer	review_id [PK] character varying	product_id character varying	product_parent integer	product_title character varying	product_category character varying	star_rating smallint	helpful_votes integer	total_votes integer	vine text	verified_purchase text	review_headline character varying	review_body character varying	review_date date
1	46387114	R26RZ3C5VL3H5W	B000M0MJU2	805416447	Intex Raised Downy Air...	Outdoors	5	0	0	N	Y	Five Stars	Very comfortable and ...	2015-08-31
2	44581842	R2A498KG3CWVC3	B000M0MJU2	805416447	Intex Raised Downy Air...	Outdoors	1	1	1	N	N	Cannot Recommend	This airmatress does n...	2015-08-31
3	32473989	R33Z46RRJXS3O7	B000M0MJU2	805416447	Intex Raised Downy Air...	Outdoors	5	0	0	N	Y	Five Stars	Good product Great qu...	2015-08-31
4	7668480	R1W6FG4HPA0K6C	B000M0MJU2	805416447	Intex Raised Downy Air...	Outdoors	5	0	0	N	Y	Five Stars	Super comfortable!!! I ...	2015-08-31

Data Processing Plan

EXTRACT

Only 1 database due to analysis focus (reviews of 1 product). Product category = outdoors

TRANSFORM

- Load Amazon product segment into PySpark outdoors dataFrame
- Identify products with the larger number of reviews
- Select the product with the highest volume of reviews
- Filter data by specific product id (B000M0MJU2) the air mattress
- Drop not needed columns: marketplace
- Transform date_review in datetime
- Convert to pandas dataframe to clean up data
- Use ntlk to remove punctuation, make it lower case and handle strange characters for review_body and review_headline

LOAD

- Connect to AWS RDS instance and write dataframe into the table
- Download Postgres driver that will allow PySpark to interact with PostgresSQL
- Configure settings for PostgresSQL
- Write the cleaned table into PostgresSQL.

Data Processing Plan: Extract

The team selected 1 specific **product segments** from Amazon data and selected 1 product:

- Product segment: outdoors
- Product_id: B000M0MJU2

Each segment has the same data schema as example below:

Raw Data Frame Example

marketplace	customer_id	review_id	product_id	product_parent	product_title	product_category	star_rating	helpful_votes	total_votes	vine	verified_purchase	review_headline	review_body	review_date
US	24509695	R3VR960AHLFKDV	B004HB5E0E	488241329	Shoal Creek Compu...	Furniture	4	0	0	N	Y	... desk is very ...	This desk is very...	2015-08-31
US	34731776	R16LGVMFKIUT0G	B0042TNMMS	205864445	Dorel Home Produc...	Furniture	5	0	0	N	Y	Five Stars	Great item	2015-08-31
US	1272331	R1AIMEEPYHMOE4	B0030MPBZ4	124663823	Bathroom Vanity T...	Furniture	5	1	1	N	Y	Five Stars	Perfect fit for m...	2015-08-31
US	45284262	R1892CCSZWZ9SR	B005G02ESA	382367578	Sleep Master Ulti...	Furniture	3	0	0	N	Y	Good enough	We use this on a ...	2015-08-31
US	30003523	R285P679YWVKD1	B005JS8AUA	309497463	1 1/4" GashGuards...	Furniture	3	0	0	N	N	Gash Gards for da...	The product is fi...	2015-08-31

Data types

```
|-- marketplace: string (nullable = true)
|-- customer_id: integer (nullable = true)
|-- review_id: string (nullable = true)
|-- product_id: string (nullable = true)
|-- product_parent: integer (nullable = true)
|-- product_title: string (nullable = true)
|-- product_category: string (nullable = true)
|-- star_rating: integer (nullable = true)
|-- helpful_votes: integer (nullable = true)
|-- total_votes: integer (nullable = true)
|-- vine: string (nullable = true)
|-- verified_purchase: string (nullable = true)
|-- review_headline: string (nullable = true)
|-- review_body: string (nullable = true)
|-- review_date: string (nullable = true)
```

Pyspark was selected due to faster processing capabilities

Data Processing Plan: Transform

1. Load Amazon product segment into PySpark outdoors dataFrame
2. Identify products with the larger number of reviews
3. Select the product with the highest volume of reviews
4. Filter data by specific product id (B000M0MJU2) the air mattress
5. Drop not needed columns: marketplace
6. Transform date_review in datetime
7. Convert to pandas dataframe to clean up data
8. Use NTLKto remove punctuation, make it lower case and handle strange characters for review_body and review_headline

Data Processing Plan: Load

- Download Postgres driver that will allow PySpark to interact with PostgresSQL
- Configure settings for PostgresSQL
- Write the cleaned table into PostgresSQL.
  - Write the cleaned air mattress table containing the review data into PostgresSQL in preparation for Topic Analysis.
- Use Amazon RDS to connect to database and load dataframe into jupyter notebook for machine learning model

Example of Cleaned Air Mattress Table Ready for PostgresSQL

	customer_id	review_id	star_rating	review_headline	review_body
0	51982153	R1DZ76NBD2TX55	5	my wife and i had to pick one of these up over...	my wife and i had to pick one of these up over...
1	44662747	R3G4HN08IK8Q5W	5	this is big and comfortable it inflatesdeflat...	this is big and comfortable it inflatesdeflat...
2	17097525	R1S3TBZK71L487	1	horrible it was so comfortable for the first f...	horrible it was so comfortable for the first f...
3	29924839	R9P8YG335IDYV	5	we bought this so our friends kids would have ...	we bought this so our friends kids would have ...
4	46198682	R5VTP1LCQIATH	4	this bed exceeded my expectations in sturdines...	this bed exceeded my expectations in sturdines...

Latent Dirichlet Allocation (LDA) Machine Learning Model

Definition

Topic modelling is the task of identifying topics that best describes a set of documents. These topics will only emerge during the topic modelling process (therefore called latent)

LDA is a popular model in topic discovery.

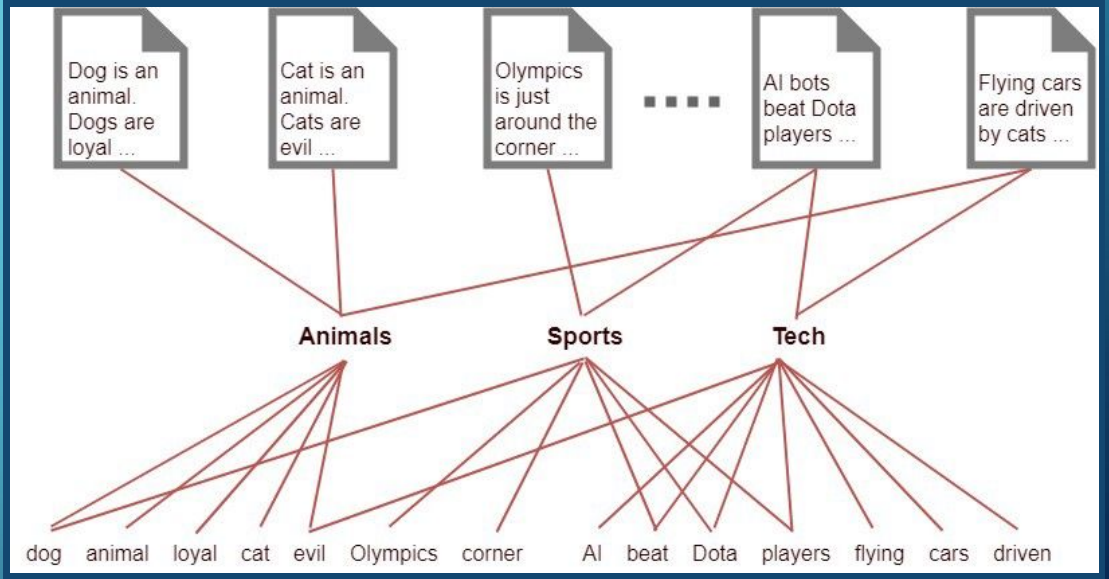
How it works?

fixed # of topics

each topic represents a group of words

LDA maps all documents to the topics

Example



Source: https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158

Latent Dirichlet Allocation (LDA) Machine Learning Model

Data preprocessing

- Connect to RDS
- Remove unwanted characters, numbers, symbols and stop words
- Remove non value table
- Separate data between 1 star and 5 star reviews
- Use nlp to remove worlds are not aggregating (only noun and adjectives)

Feature selection

- Topic discovery for customer reviews to gather feedback and identify themes: product qualities and what has to be improved
- Find 'relevant' topics and identify trends
- Topic Modelling is an unsupervised approach used for finding and observing the bunch of words (called "topics") in large clusters of texts

Model selection

LDA

- Benefits
  - Largely used for topic discovery
  - Simple to implement
  - Runs relatively quickly
  - Probabilistic model
- Limitations
  - User defines # of topics
  - Hard to interpret output since there's no Y variable

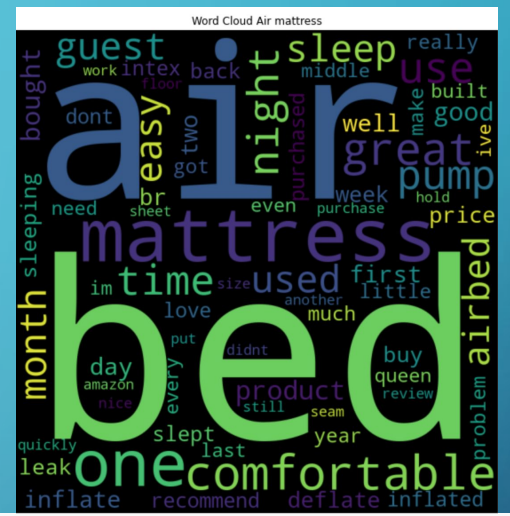
Question #3: Can we extract key topics within product reviews to help companies analyze and interpret customer feedback?

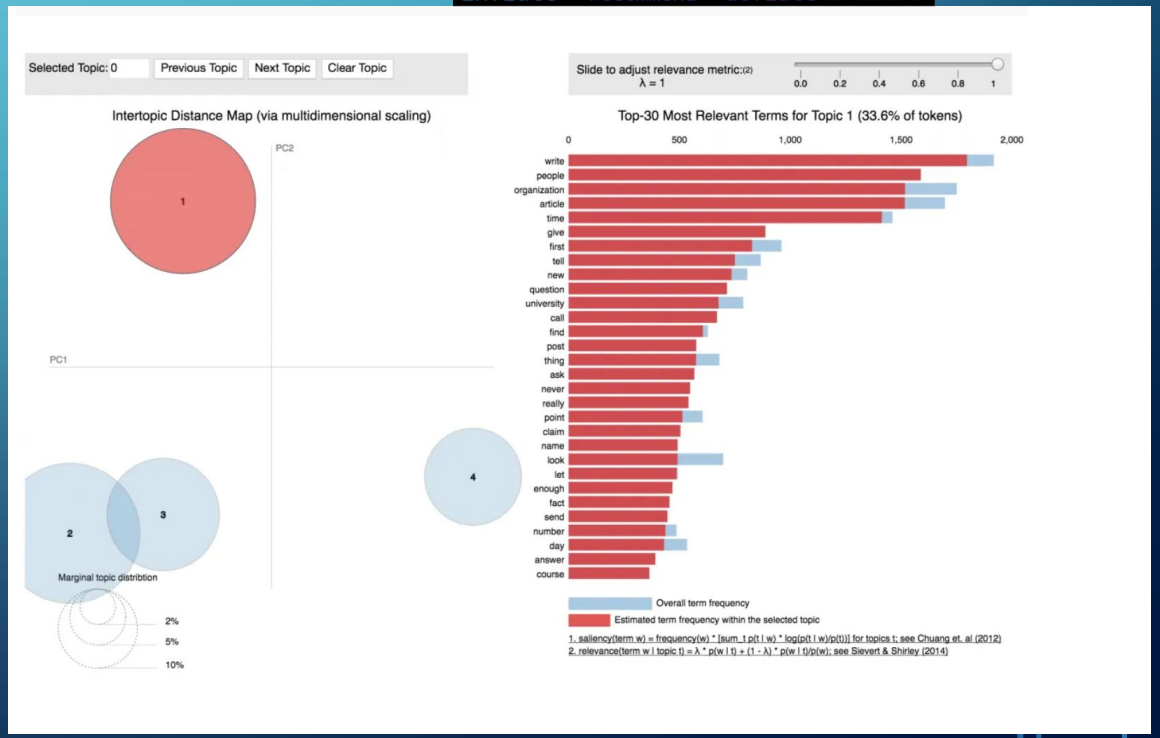
Visualization Plan

- In order to visualize highlighted topics within review descriptions, a bar chart and a bubble chart can be used to display the weight and frequency of a word through a Latent Dirichlet Allocation model (LDA). https://www.machinelearningplus.com/nlp/topic-modeling-visualization-how-to-present-results-lda-models/
- To display this visualization, the user can select a certain word from a dropdown. After selecting a word, a graph and a bubble chart would pop up showing the frequency and the weight of the word. A longer bar and a larger bubble would represent a word with high frequency and heavy weight. The color of the bubble and bar chart would also represent whether the sentiment of the world is negative or positive.
- https://www.machinelearningplus.com/nlp/topic-modeling-visualization-how-to-present-results-lda-models/

Visualization Tools

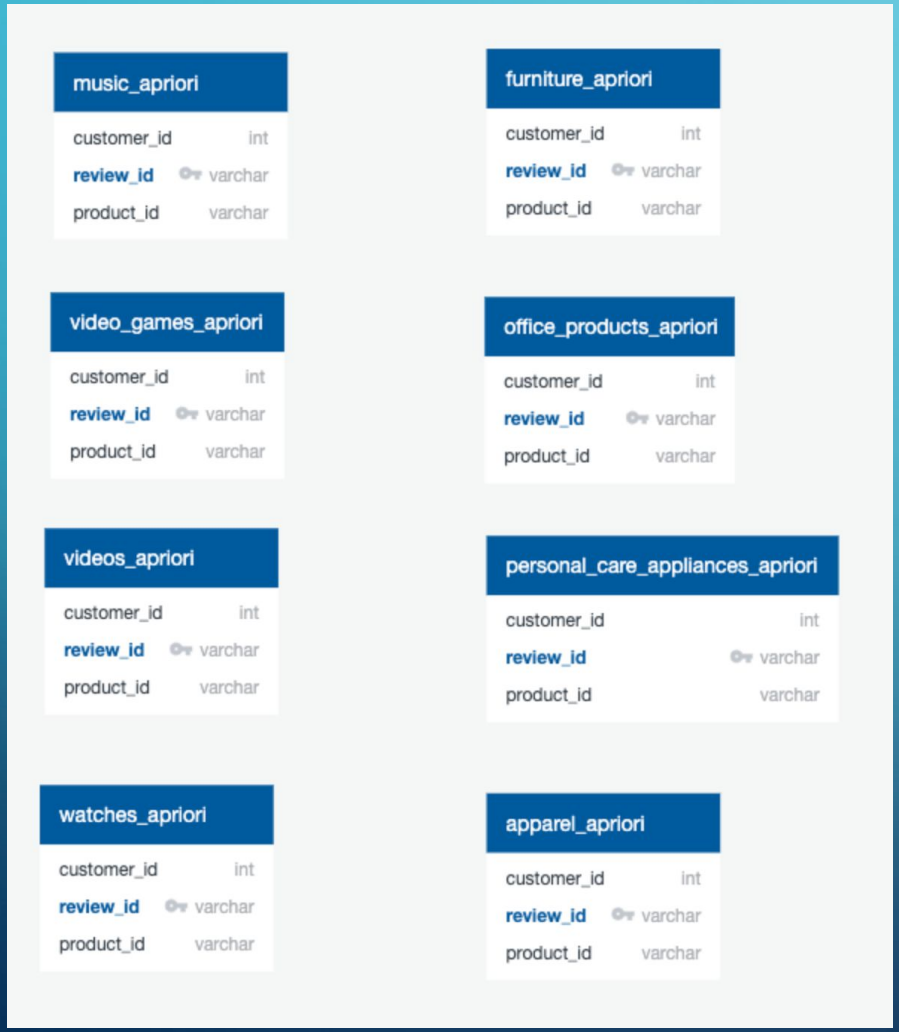
- To visualize the bubble and bar charts, the Python dependency PyLDAvis.genism can be utilized to display the visualization.
  - https://pyldavis.readthedocs.io/en/latest/modules/API.html
- To create the dropdown menu, the Pyphi module and Jsonify dependency can be utilized to display the visualization through HTML.
  - https://pyphi.readthedocs.io/en/latest/api/jsonify.html
- The word cloud can be visualized through the Python Matplotlib dependency https://matplotlib.org/stable/contents.html





ERD's For Apriori Analysis Tables & K-Means Segmentation Analysis Tables Before Joins

Apriori Analysis Tables



K-Means Segmentation Analysis Tables