NOVA IMS Information Management School

BUSINESS CASES FOR DATA SCIENCE

MASTER DEGREE PROGRAM IN DATA SCIENCE AND ADVANCED ANALYTICS

BUSINESS CASE 3

Gift-A-Lot

Recommender System

GROUP I

Emanuele Aldera, m20210617

Robin Schmidt, m20210602

Muhammad Abdullah, m20210655

Rui Ramos, m20210640

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

INDEX

Summary

1. Introduction and Business Understanding

With the growing volume of online information, recommender systems have been an effective strategy to overcome information overload.

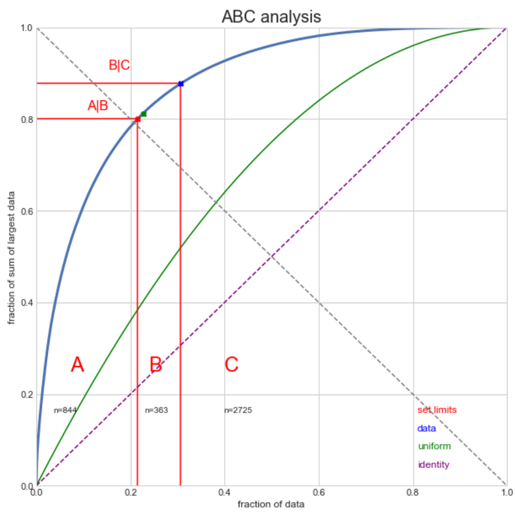
The utility of recommender systems cannot be overstated, given their widespread adoption in many web applications, along with their potential impact to ameliorate many problems related to over-choice. These choices can really make a difference and lead to an increase in sales. As an example, 35% of Amazon sales come from recommendations.

With the data, the company expects us to produce a recommender system that can facilitate user choices by recommending items that the user likes and enhances user experience when making purchases on its website.

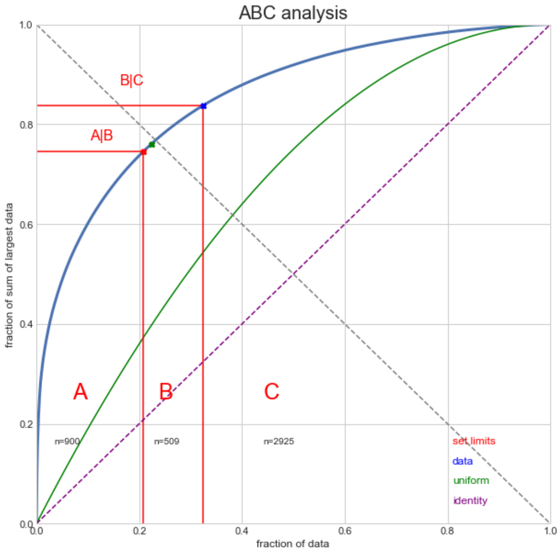
Gift-a-Lot is a UK-based and registered non-store online retailer with about 80 members of staff. The company was established in 1981, selling unique all-occasion gifts.

For this project we received a dataset with 542K observations and 8 features characterizing the invoice by customer and country with timeline date, quantity, unit price and product description, from the company “Gift a lot” that contract our company to help them to build a recommender system.

The following ABC analysis was constructed to help understand which products were the most important ones and proved to be profitable for the company, we can see that around the 80% of the profits made by the company via web purchases regard only 844 products.



By doing the same analysis for customers, we can conclude that around 75% of the same profits regard only 900 customers; this second analysis, however, is less accurate since, as it will be shown in the later chapters, we have some missing values regarding customers.

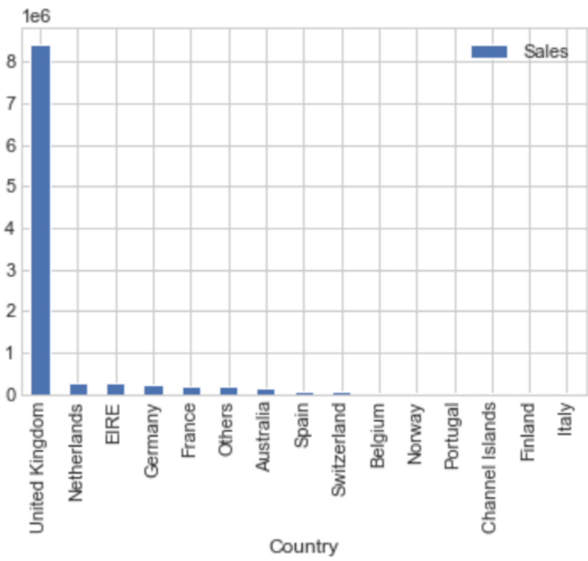


2. Data Exploration

The dataset contained 541909 rows and 8 rows: 2 of them are float variables (“*UnitPrice*” and “*CustomerID*”), 5 are string variables (“*InvoiceNo*”, “*StockCode*”, “*Description*”, “*InvoiceDate*” and “*Country*”) while the last variable, “Quantity”, is an int64 type.

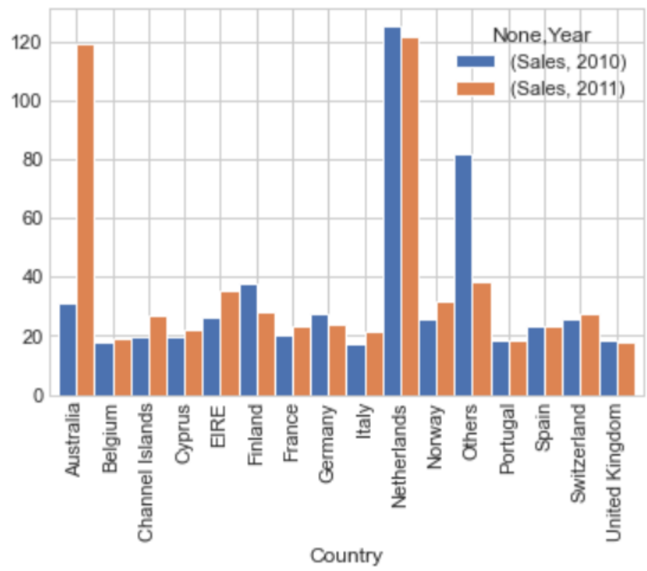
Analysing the data, we could see that the data recorded for two years transactions, 2010 & 2011 respectively, we found the transactions to be in 38 countries 4223 types of products and 4372 customers.

Visualizing the sales by country we understand that we could cluster the countries by sales performance, to simplify the process we could simply compare the purchase behaviors and access the segments. However, for this case we had a high discrepancy between countries and their total sales, so another approach could be comparing sales evolution.



When we compare sales by year we have a distinct perspective, with the histogram we can see that our biggest country in sales is UK which has a flat evolution. This could be investigated and implemented in the recommender analysis. If we could add good recommendations to the country with more sales, certainly it would impact and drive the sales positively.

Also, the relevant countries with low sales but with a good increasing sales ratio like Australia, Channel Islands, Italy, Eire, and Norway will be insightful to further explore. It can be very opportunistic if the company is able to drive the growth of sales by implementing the recommender system. Netherlands too could be an interesting target and contributor towards the growth, if we understand and capitalize on what kind of products to recommend. Contrarily, for a few other countries which had a growth in sales like Cyprus, France, and Switzerland, we noted that the growth in sales ratio was low. For a starting point we will analyse 3 to 4 countries.



Our dataset also presents many canceled orders: they can be found by filtering by negative *Quantity*, and the invoice Number contains a “C” character. In total we found 9200 canceled records, corresponding to 3754 different invoices. To understand the dataset consistency, we checked whether for a cancelation there would be also the record of the order before it was canceled, realizing this was not the case.

After exploring the missing values within the dataset, we found out that only two variables; *CustomerID* and *Description* had missing entries of 133274 and 592, respectively. We tried to recover some values using the invoice number, however invoices containing missing values for the *CustomerID* had a missing value for all records.

After checking the dataset consistency, each invoice must contain only one unique *CustomerID*, we also checked whether for each product (*StockCode*) only one description was allocated; the result shows how some products contain different descriptions. After exploring those differences in descriptions, however, it was decided not to take care of this issue given the nature of the different descriptions; from now on we would consider products with the same Stock Code to be equal.

Furthermore, we found some Stock Codes not related to actual products: they could be filtered since the length of the string was less than 5 digits. In addition to that, 1163 records related to 675 products having *UnitPrice* equal to 0.

By checking duplicated records on the subset of *InvoiceNo* and *StockCode*, around 2.02% of records would appear as duplicated, meaning that for some invoices the ordered products would appear singularly instead of adding up on the *Quantity* variable.

3. Data Transformation

Starting from our original dataset, we created some features to better understand purchasing behaviours from the customers. We focused especially on getting information regarding the invoice's date: first we cast the *InvoiceDate* to *datetime* type, and after we created variables such as *Year, Month, Day, Weekday, WeekdayName, IsWeekend, Time, Hour* and *Daytime* (this last one has 6 unique values corresponding to what period of the day it is : 'Late Night', 'Early Morning', 'Morning', 'Noon', 'Eve', 'Night').

Cancelations were removed from our dataset before applying the models since the orders they could create a bias. Stock codes not related to products were also removed from the dataset.

Duplicated records on the subset *InvoiceNo* and *StockCode* were grouped together into a single record containing the sum of all individual quantities. After having done that, the variable *TotalPrice* was added for containing the product of *Quantity* and *UnitPrice.* A similar variable, *Sales,* was created before all feature engineering to have a look into information for Business Understanding.

We also decided to reduce the cardinality of the variable *Country,* by grouping the ones appearing in less than 500 records into a "Others" category.

4. Basket Market Analysis

4.1. Association rules

The applications of Association Rule Mining are found in Market Basket Analysis, which is a key technic used by large retailers like Amazon, to analyze customer buying habits by finding associations between the different items that customers place in their "shopping baskets".

Taking in account the big differences in sales from countries and to give some perspective outside UK the main country we also select Netherlands and Italy to apply this technique.

There is a difference between association rules and Recommendation that we need to have in account, but they could be seen as complementary.

Association rules doesn't extract an individual's preference, rather find relationships between sets of elements of every distinct transaction.

4.2. Association Rules Metrics

'support' → How frequently the combination occurs

$$\text{support}(A \to C) = \text{support}(A \cup C), \text{range: } [0,1]$$

'confidence' → The strength of an association

$$\text{confidence}(A \to C) = \text{support}(A \to C)\text{support}(A), \text{range: } [0,1]$$

'lift' → Increase in the sale of A when you sell B

$$\text{lift}(A \to C) = \text{confidence}(A \to C)\text{support}(C), \text{range: } [0,\infty]$$

4.3. Apriori Algorithm

To implement the apriori algorithm we used the mlxtend library, and here are the following steps we took:

1. Group by country
2. Create a matrix to understand which products are bought together for each invoice
3. We tested the algorithm with two levels of minimum support: 0.05 and 0.03
4. For the lift metric the min threshold consider was 1 (100% confidence)
5. Sort values by high confidence and lift to see the products who are more often bought together and the opposite for substitutes.

4.4. United Kingdom

	antecedents	consequents	support	confidence	lift
3	(PINK REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER)	0.031898	0.820768	15.896012
18	(PINK REGENCY TEACUP AND SAUCER)	(ROSES REGENCY TEACUP AND SAUCER)	0.030184	0.776671	14.680469
4	(GREEN REGENCY TEACUP AND SAUCER)	(ROSES REGENCY TEACUP AND SAUCER)	0.038753	0.750535	14.186451
5	(ROSES REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER)	0.038753	0.732497	14.186451
9	(JUMBO BAG PINK POLKADOT)	(JUMBO BAG RED RETROSPOT)	0.043397	0.677308	6.321891

Table 1 – UK products with higher confidence and lift (Complementary)

We can consider logical this associations because of the English culture of drinking tea, they bought together Teacup and saucer of different colors maybe to be able to serve the tea with different sets each day or in specific occasions.

	antecedents	consequents	support	confidence	lift
7	(JUMBO BAG RED RETROSPOT)	(JUMBO BAG BAROQUE BLACK WHITE)	0.031511	0.294118	5.846477
10	(JUMBO BAG RED RETROSPOT)	(JUMBO SHOPPER VINTAGE RED PAISLEY)	0.036320	0.339009	5.412479
12	(JUMBO BAG RED RETROSPOT)	(JUMBO STORAGE BAG SUKI)	0.038587	0.360165	5.765510
16	(LUNCH BAG RED RETROSPOT)	(LUNCH BAG PINK POLKADOT)	0.030626	0.397989	7.156276
8	(JUMBO BAG RED RETROSPOT)	(JUMBO BAG PINK POLKADOT)	0.043397	0.405057	6.321891

Table 2 – UK products with Lower confidence and lift (Substitutes)

4.5. Netherlands

	antecedents	consequents	support	confidence	lift
650	(FOLDING BUTTERFLY MIRROR RED)	(FOLDING BUTTERFLY MIRROR HOT PINK)	0.053191	1.0	18.8
651	(FOLDING BUTTERFLY MIRROR HOT PINK)	(FOLDING BUTTERFLY MIRROR RED)	0.053191	1.0	18.8
1351	(FOOD CONTAINER SET 3 LOVE HEART, CARD DOLLY G...	(10 COLOUR SPACEBOY PEN)	0.053191	1.0	18.8
1354	(10 COLOUR SPACEBOY PEN)	(FOOD CONTAINER SET 3 LOVE HEART, CARD DOLLY G...	0.053191	1.0	18.8
1381	(CARD DOLLY GIRL, STRAWBERRY LUNCH BOX WITH CU...	(10 COLOUR SPACEBOY PEN)	0.053191	1.0	18.8

Table 3 – Netherlands products with higher confidence and lift (Complementary)

In the Netherlands we see association between card Dolly girl food container and space boy pen products of primary school it seems logical that they are bought together.

	antecedents	consequents	support	confidence	lift
204	(SPACEBOY LUNCH BOX)	(CARD DOLLY GIRL)	0.053191	0.178571	1.398810
1314	(SPACEBOY LUNCH BOX)	(SET OF 3 REGENCY CAKE TINS)	0.053191	0.178571	1.398810
220	(SPACEBOY LUNCH BOX)	(CARD GINGHAM ROSE)	0.053191	0.178571	1.865079
2763	(SPACEBOY LUNCH BOX)	(ROUND SNACK BOXES SET OF4 WOODLAND, CARD DOLL...	0.053191	0.178571	1.865079
2787	(SPACEBOY LUNCH BOX)	(CARD DOLLY GIRL, SPACEBOY BIRTHDAY CARD)	0.053191	0.178571	1.865079

Table 4 – Netherlands products with Lower confidence and lift (substitutes)

4.6. Italy

	antecedents	consequents	support	confidence	lift
22	(RED LOVE HEART SHAPE CUP)	(BAKING MOULD CHOCOLATE CUPCAKES)	0.057143	1.0	17.5
23	(BAKING MOULD CHOCOLATE CUPCAKES)	(RED LOVE HEART SHAPE CUP)	0.057143	1.0	17.5
50	(HOME BUILDING BLOCK WORD)	(BATH BUILDING BLOCK WORD)	0.057143	1.0	17.5
51	(BATH BUILDING BLOCK WORD)	(HOME BUILDING BLOCK WORD)	0.057143	1.0	17.5
160	(CHRISTMAS CRAFT WHITE FAIRY)	(CHRISTMAS CRAFT LITTLE FRIENDS)	0.057143	1.0	17.5

Table 4 – Italy products with higher confidence and lift

Here we have combinations about chocolate and valentines' gifts that also makes sense taking in account that Italian likes to offer and to show affection. Also, what looks home decoration and Christmas products are the most who are bought together. It shows more mix in their buying behavior in Italy this could identify some important information from marketing perspective.

	antecedents	consequents	support	confidence	lift
1011	(SET OF 3 CAKE TINS PANTRY DESIGN)	(REGENCY CAKESTAND 3 TIER)	0.057143	0.222222	1.555556
297	(SET OF 3 CAKE TINS PANTRY DESIGN)	(DOORMAT SPOTTY HOME SWEET HOME)	0.057143	0.222222	1.944444
965	(SET OF 3 CAKE TINS PANTRY DESIGN)	(RECIPE BOX PANTRY YELLOW DESIGN)	0.057143	0.222222	1.944444
981	(SET OF 3 CAKE TINS PANTRY DESIGN)	(RECIPE BOX RETROSPOT)	0.057143	0.222222	1.944444
1048	(SET OF 3 CAKE TINS PANTRY DESIGN)	(SET 3 RETROSPOT TEA,COFFEE,SUGAR)	0.057143	0.222222	1.944444

Table 5 –Netherlands products with Lower confidence and lift (substitutes)

5. Recommender Systems

With the growing volume of online information, recommender systems have been an effective strategy to overcome information overload.

The utility of recommender systems cannot be overstated, given their widespread adoption in many web applications, along with their potential impact to ameliorate many problems related to over-choice.

5.1. Train Test Split

Splitting the data into training and testing sets is an important part for evaluating our predictive collaborative filtering models. We used an 80:20 ratio for our train-test set size. Our training portion will be used to develop a predictive model, while the other to evaluate the model's performance.

5.2. Popularity Model as Baseline

The popularity model takes the most popular items for recommendation. These items are products with the highest number of sells across customers. Three different approaches were chosen and evaluated: the total quantity (purchase_count), the number of transactions (purchase_dummy) and the normalized quantity across users.

5.3. Collaborative Filtering Model

In collaborative filtering, we would recommend items based on how similar users purchase items. For instance, if customer 1 and customer 2 bought similar items, e.g. 1 bought X, Y, Z and 2 bought X, Y, we would recommend an item Z to customer 2. First, we created a user-item matrix, where index values represent unique customer IDs and column values represent unique product IDs. As a next step, we created an item-to-item similarity matrix. The idea is to calculate how similar a product is to another product. We used 'cosine' and 'pearson' for the similarity measurement. For each customer, we then predict his likelihood to buy a product (or his purchase counts) for products that he had not bought. Again, three different approaches were chosen and evaluated: the total quantity (purchase_count), the number of transactions (purchase_dummy) and the normalized quantity across users (purchase_norm).

6. Evaluation

For evaluating recommendation engines, we used the concept of RMSE and precision-recall. The RMSE (Root Mean Squared Errors) measures the error of predicted values and therefore, the lesser the RMSE value, the better the recommendations. The recall indicates what percentage of products that a user buys are actually recommended. If a customer buys 5 products and the recommendation decided to show 3 of them, then the recall is 0.6. The precision indicates how many, out of all the recommended items, the customer actually liked. If 5 products were recommended to the customer out of which he buys 4 of them, then precision is 0.8. The corresponding score tables can be found in the Appendix.

Mean Precisicion	Count	Dummy	Normalized
Popularity	0.000	0.002	0.000
Cosine	0.016	**0.133**	0.013
Pearson	0.000	0.002	0.000

Mean Recall	Count	Dummy	Normalized
Popularity	0.000	0.001	0.000
Cosine	0.013	**0.091**	0.011
Pearson	0.000	0.001	0.000

Popularity vs. Collaborative Filtering: For purchase counts, we observed that collaborative filtering algorithms outperform the popularity model. Indeed, the popularity approach does not provide any personalization, as each user receives the identical list of recommended goods.

Precision and recall: Looking at the scores we detected that purchase_dummy > purchase_norm > purchase_count.

RMSE: Since RMSE is higher using pearson distance than cosine, we would choose model the smaller mean squared errors, which in this case would be cosine.

Therefore, our final selected model was purchase_dummy with cosine distance.

Cold Start: For the Cold Start we selected the outcome of the Popularity Count Engine, which is based on the purchase count of each product. The 20 most popular products are presented in the table below.

StockID	Purchase Count	Description
85123A	2203	WHITE HANGING HEART T-LIGHT HOLDER
85099B	2092	JUMBO BAG RED RETROSPOT
22423	1989	REGENCY CAKESTAND 3 TIER
47566	1686	PARTY BUNTING
20725	1565	LUNCH BAG RED RETROSPOT
84879	1455	ASSORTED COLOUR BIRD ORNAMENT
22197	1392	SMALL POPCORN HOLDER
22720	1387	SET OF 3 CAKE TINS PANTRY DESIGN
21212	1320	PACK OF 72 RETROSPOT CAKE CASES
22383	1285	LUNCH BAG SUKI DESIGN
20727	1273	LUNCH BAG BLACK SKULL.
22457	1249	NATURAL SLATE HEART CHALKBOARD
22386	1218	JUMBO BAG PINK POLKADOT
22469	1202	HEART OF WICKER SMALL
21931	1184	JUMBO STORAGE BAG SUKI
22411	1175	JUMBO SHOPPER VINTAGE RED PAISLEY
22961	1162	JAM MAKING SET PRINTED
22086	1160	PAPER CHAIN KIT 50'S CHRISTMAS
22382	1157	LUNCH BAG SPACEBOY DESIGN
20728	1150	LUNCH BAG CARS BLUE

7. Deployment

After having analyzed in depth the market basket analysis and the different recommender systems, it is now important to understand how these models can be deployed to be able to create some value to the company Gift-A-Lot.

The objective of the company is to build a recommender system that can facilitate user choices by recommending items the user likes and improve user experience when making purchases on its website.

To do that, we strongly suggest to the company to change their website to be able to add the recommendations engine; after the customer has added to the virtual basket the items he's interested in and before the payment website, a page will appear with the items the engine recommends for that specific order. It is important that we combine Market Basket Analysis together with the Recommender Systems; by doing that, we can suggest complementary products and avoid substitute products to the ones in the basket and aggregate this information with the one provided us by the engine.

These models allow us also a way to interact and hopefully re-attract customers by sending them suggestions of items they might be interested in, based on their purchasing history, via e-mail; in some cases, we can offer them a small discount to encourage them to purchase the items we are suggesting them.

The models will keep improving themselves after each purchase, meaning that with time they'll keep being more accurate and providing more insights and value to the company.

We are confident that deploying these models will have positive benefits to the company since it will increase sales. This will come with some challenges and costs however: first of all, the cost of maintaining those models and changing the front end of the website. On top of that, the challenge of "cold start" also must be addressed: what can we suggest to new customers we don't have information about? We suggest a two-way approach: when a costumer signs up to the website, it is presented a page in which there's a list of keywords and he can select the ones he's most interested in; a similar approach is taken for example by some streaming platform, such as Netflix, to have a general idea of the customer's preferences. It is important that this questionnaire to fill is very simple and fast to not lose customer's interest. If the customer decides to fill this information, the suggestions will be suited to the keywords he selected; after the customer has made some sales, we'll be able to suggest him products more suited to him/her. In the case the customer decides to skip the questionnaire, the suggestions to face the cold start problem will be a mix of the most popular products and the ones that are more profitable to the company.

8. Appendix

Precision and recall summary statistics by cutoff

cutoff	mean_precision	mean_recall
1	0.00024545900834560 6	0.00024545900834560 6
2	0.00012272950417280303	0.00024545900834560606
3	0.000490918016691212 6	0.000503616043504669 8
4	0.000613647520864015 3	0.000538076126499076 3
5	0.000638193421698576 5	0.000557413341694290 8
6	0.000572737686139748 1	0.000557769079387544 8
7	0.000490918016691211 9	0.00055776907938754 5
8	0.000460235640648011 3	0.000558124817080799 2
9	0.00040909834724267715	0.000558124817080799 4
10	0.000490918016691212 4	0.000758497182401618 3

[10 rows x 3 columns]

Figure 1 Scores Count Popularity

Precision and recall summary statistics by cutoff

cutoff	mean_precision	mean_recall
1	0.026509572901325478	0.004234794800269071 6
2	0.0220913107511045 6	0.007280770961106658 4
3	0.0192276223204058 3	0.00889975967342432 2
4	0.016507118311242016	0.01000006067694221 7
5	0.014825724104074647	0.01091672823793190 4
6	0.013909343806251013	0.01273135586132219 6
7	0.0131846553054211 7	0.01393074132555444 6
8	0.0122729504172803 1	0.01491947720463640 4
9	0.011618393061692052	0.0156060613003670 8
10	0.011119293078056003	0.01631194772153209 8

[10 rows x 3 columns]

Figure 2 Scores Count Cosine

Precision and recall summary statistics by cutoff

cutoff	mean_precision	mean_recall
1	0.000245459008345606 2	0.000245459008345606 2
2	0.00012272950417280309	0.00024545900834560617
3	0.000490918016691212 1	0.000503616043504670 2
4	0.000675012272950417 4	0.000619895795947611 7
5	0.000687285223367697 4	0.00063923301114282 6
6	0.000613647520864015 3	0.000639588748836080 3
7	0.000525983589312013 7	0.000639588748836080 2
8	0.000490918016691212 7	0.000639944486529334 8
9	0.00046364479354170135	0.000701309238615736 2
10	0.000540009818360334 5	0.000901681603936555 2

[10 rows x 3 columns]

Figure 3 Scores Count Pearson

Precision and recall summary statistics by cutoff

cutoff	mean_precision	mean_recall
1	0.0007387343018960846	2.9137187472572872e-05
2	0.0006156119182467374	3.5219141522434894e-05
3	0.00098497906919478	0.00018673630585843348
4	0.0012312238364934744	0.00023158586130832251
5	0.00128047278995214	0.00030979406324840603
6	0.0012312238364934757	0.0003421453654921604
7	0.001160868188693848	0.0004163741272363126
8	0.0011696626446688015	0.0005511773524113674
9	0.0011491422473939076	0.0006065228280400883
10	0.001181974883033736	0.0006866989168672667

[10 rows x 3 columns]

Figure 4 Score Dummy Population

Precision and recall summary statistics by cutoff

cutoff	mean_precision	mean_recall
1	0.18936222605269634	0.03255256878897112
2	0.16609209554296972	0.0528576914652512
3	0.14848559468111275	0.06703162821786555
4	0.13931297709923665	0.07919007877085432
5	0.13060822457522764	0.09044750429206527
6	0.12230156775835178	0.0996512729370338
7	0.11619235234108417	0.10934818579493152
8	0.11087170647623737	0.11740619725639011
9	0.10624093682453686	0.12443924777217517
10	0.10206845604530895	0.13113853554105206

[10 rows x 3 columns]

Figure 5 Scores Dummy Cosine

Precision and recall summary statistics by cutoff

cutoff	mean_precision	mean_recall
1	0.001231223836493473	0.0001773727963485111
2	0.0013543462201428226	0.00025160155809266307
3	0.0013953870146926036	0.00033303052021519026
4	0.0014159074119674952	0.00035186178390840376
5	0.0013789706968726906	0.0005030072860975491
6	0.0014364278092423867	0.0005732905198109209
7	0.0014071129559925428	0.0008592078411192507
8	0.0013235656242304838	0.0009439899500333474
9	0.0012859448958931843	0.0010509292756248007
10	0.0012312238364934748	0.0010687640209054695

[10 rows x 3 columns]

Figure 6 Scores Dummy Pearson

Precision and recall summary statistics by cutoff

cutoff	mean_precision	mean_recall
1	0.000246062992125984	2.196991001124859e-06
2	0.00036909448818897616	3.619253596589901e-05
3	0.00032808398950131233	3.915715032886265e-05
4	0.00043061023622047276	0.00012784825137073834
5	0.0005413385826771655	0.00016085165271894142
6	0.0005331364829396324	0.0002289921736153678
7	0.0004921259842519681	0.0002293529697915052
8	0.00046136811023622133	0.00022971376596764339
9	0.0004101049868766405	0.00022971376596764339
10	0.0004183070866141734	0.0002490024846150105

[10 rows x 3 columns]

Figure 7 Scores Norm Popularity

Precision and recall summary statistics by cutoff

cutoff	mean_precision	mean_recall
1	0.022391732283464562	0.004527086302445464
2	0.017347440944881897	0.006643743034083751
3	0.013943569553805784	0.007989714648472162
4	0.012303149606299206	0.009136747171397923
5	0.010826771653543326	0.009737538175027396
6	0.0098015091863517	0.010590941015190451
7	0.00917463442069741	0.011361847359045881
8	0.008581446850393708	0.011964217689318173
9	0.00817475940507435	0.01250774410233907
10	0.007652559055118124	0.013084939808993643

[10 rows x 3 columns]

Figure 8 Scores Norm Cosine

Precision and recall summary statistics by cutoff

cutoff	mean_precision	mean_recall
1	0.0004921259842519686	4.949542945062903e-05
2	0.0004921259842519687	7.166102287088466e-05
3	0.0004101049868766402	7.202181904702247e-05
4	0.00043061023622047243	0.00012159521364835707
5	0.0003937007874015742	0.00012195600982449511
6	0.0004101049868766398	0.00014124472847186237
7	0.0003515185601799773	0.0001412447284718623
8	0.00039985236220472435	0.00021143577430267205
9	0.0003827646544181976	0.00022168839897458804
10	0.00034448818897637795	0.000221688398974588

[10 rows x 3 columns]

Figure 9 Scores Norm Pearson