**DIGITAL ASSIGNMENT-3**

**DATE:24-04-2020**

**NAME: AVULA.ROHITHA**

**REG NO:17BLC1007**

**COURSE TITLE: DATA MINING**

**COURSE CODE: CSE3019**

**SLOT: D2**

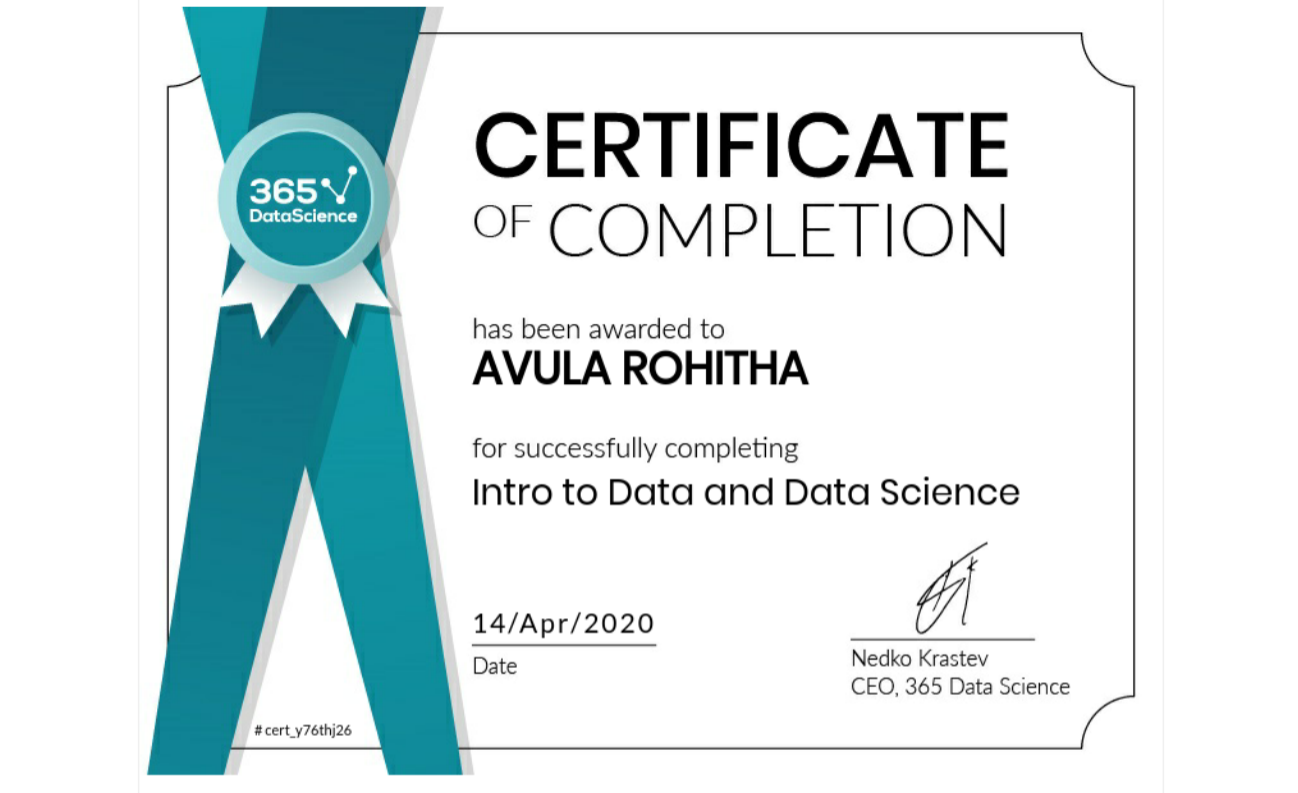
**FACULTY: Dr. ANUSHA K**

**----------------------------------------------------------------------------------------------------------------**

**QUESTION 1:**

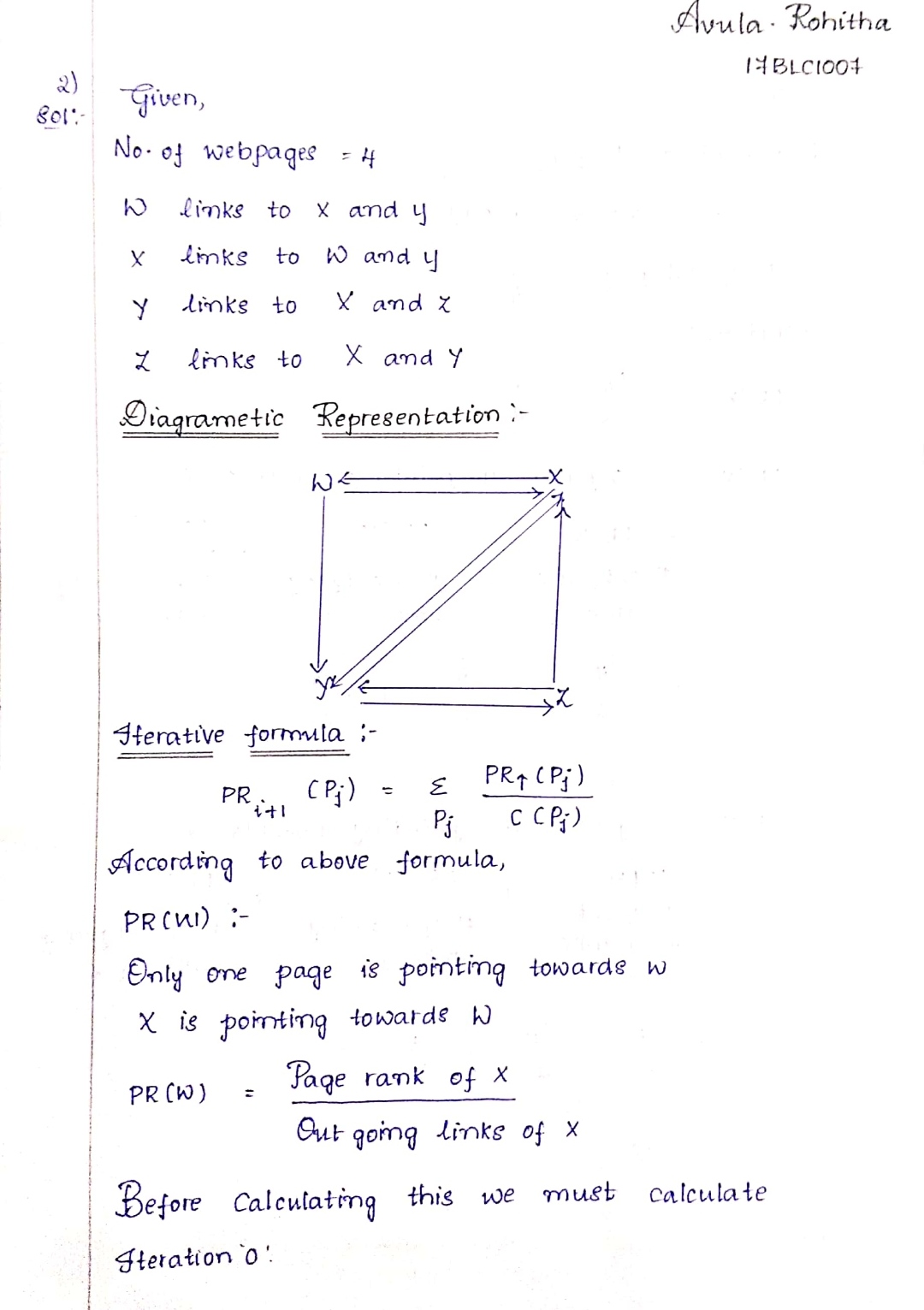
**Register and attend any free online course related to data mining and upload the certificate for the same in moodle**

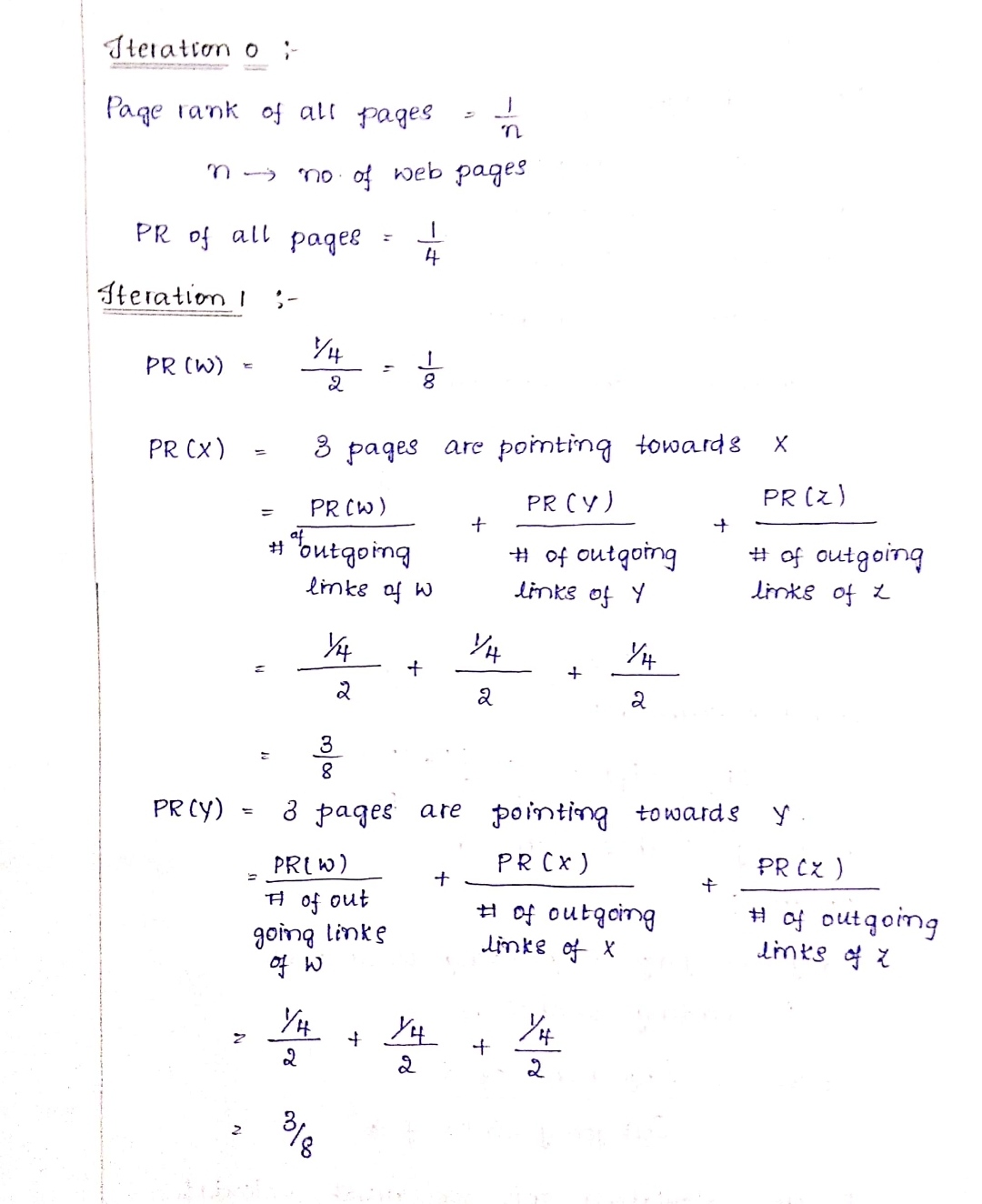
**CERTIFICATE:**

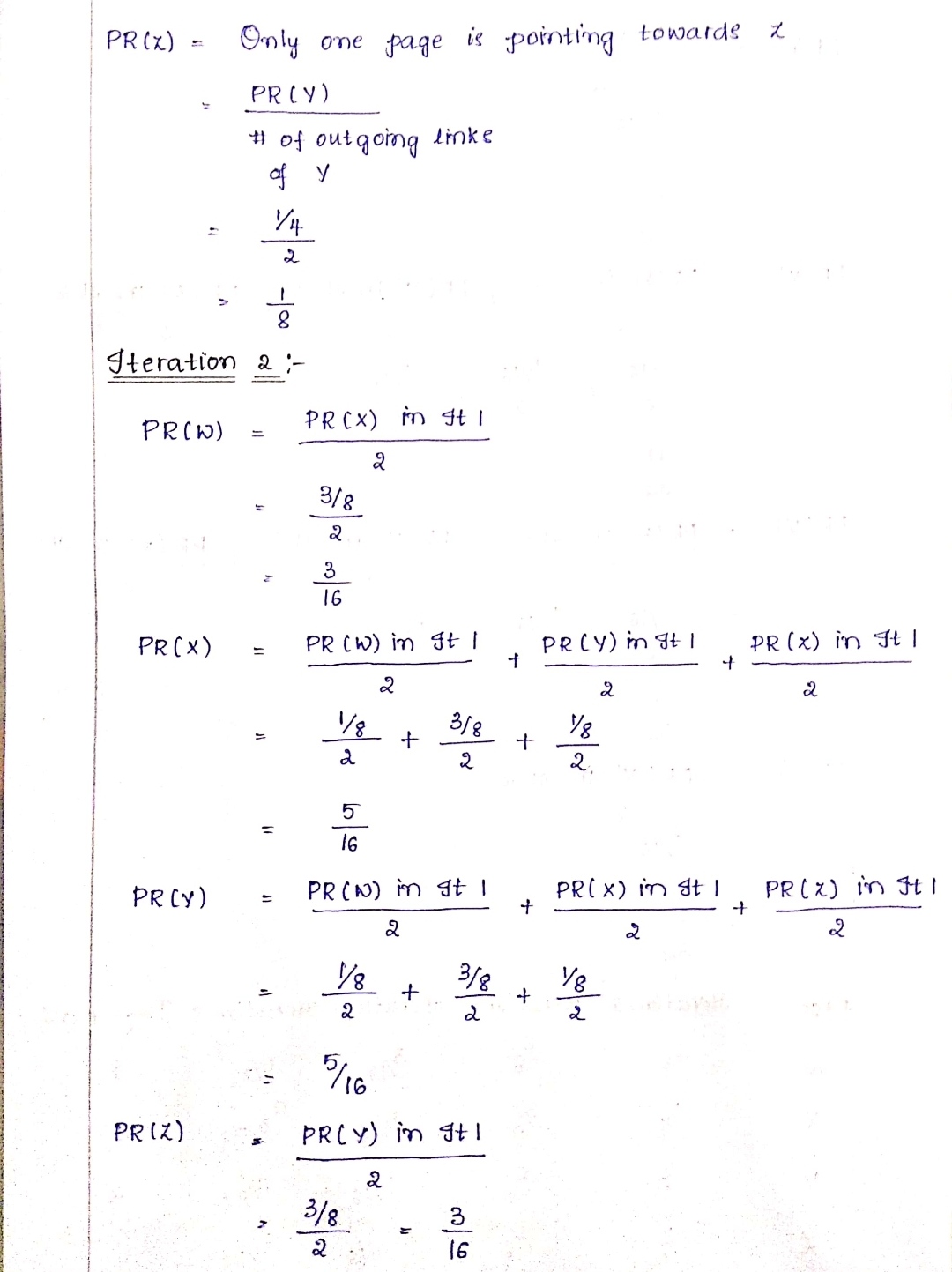


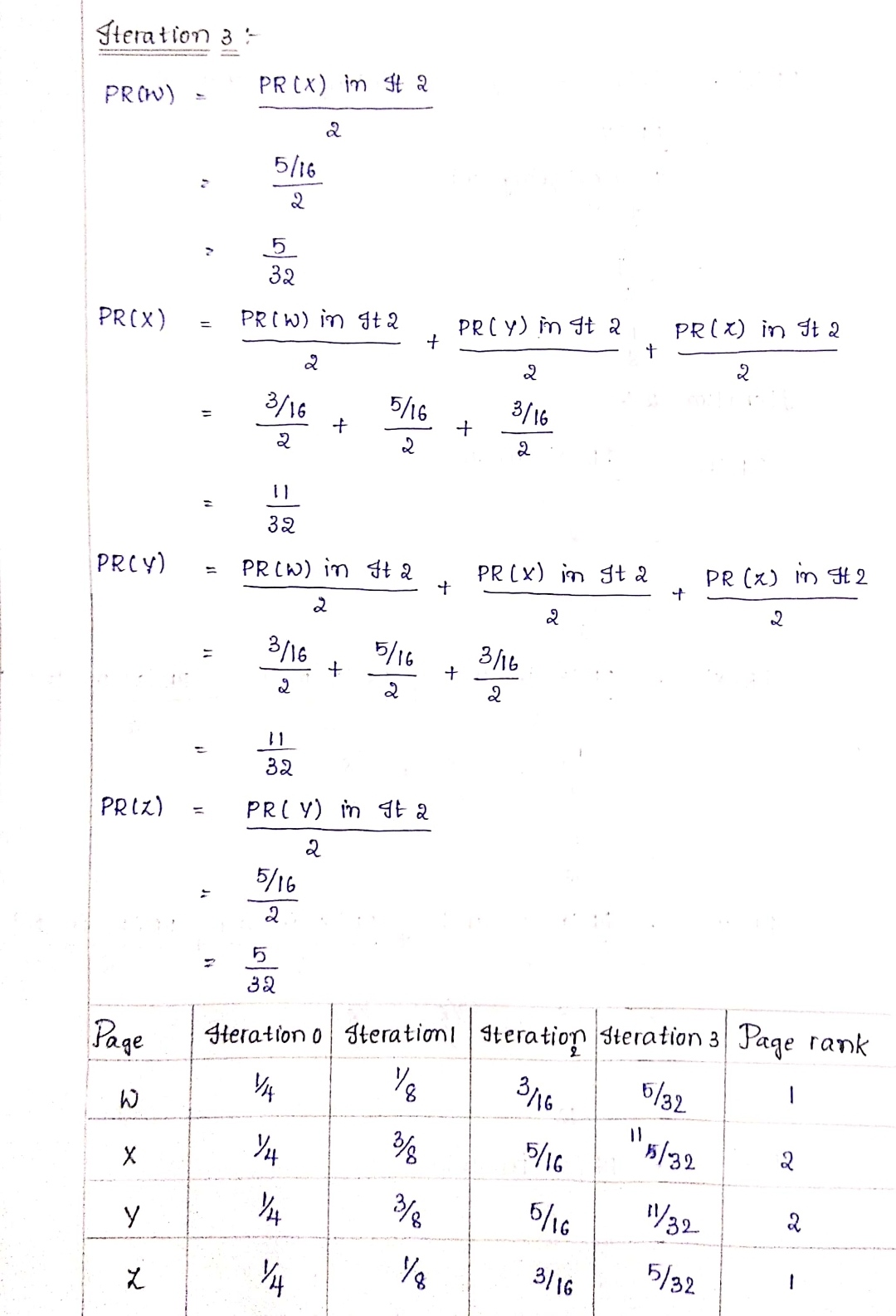
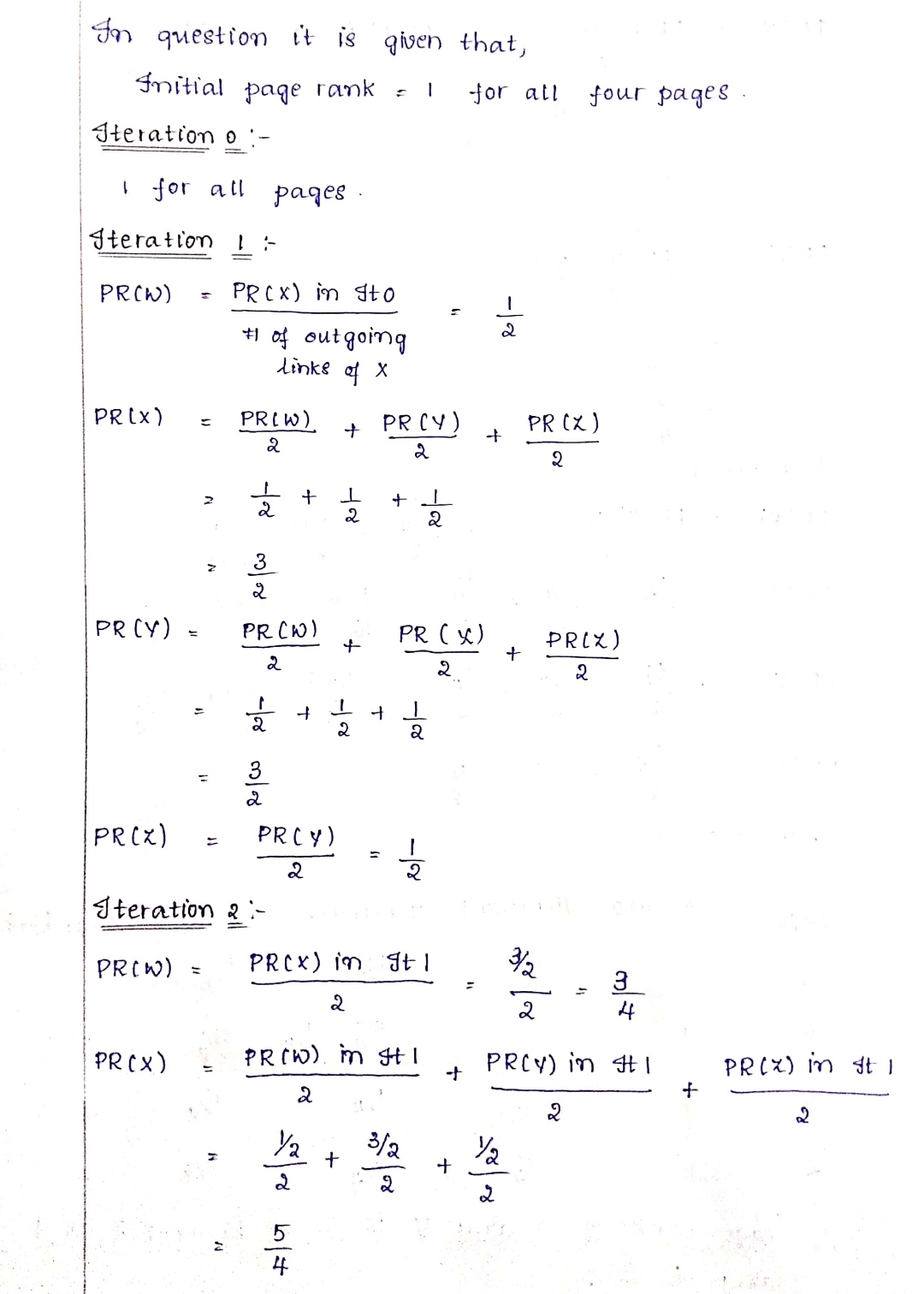
**QUESTION 2:**

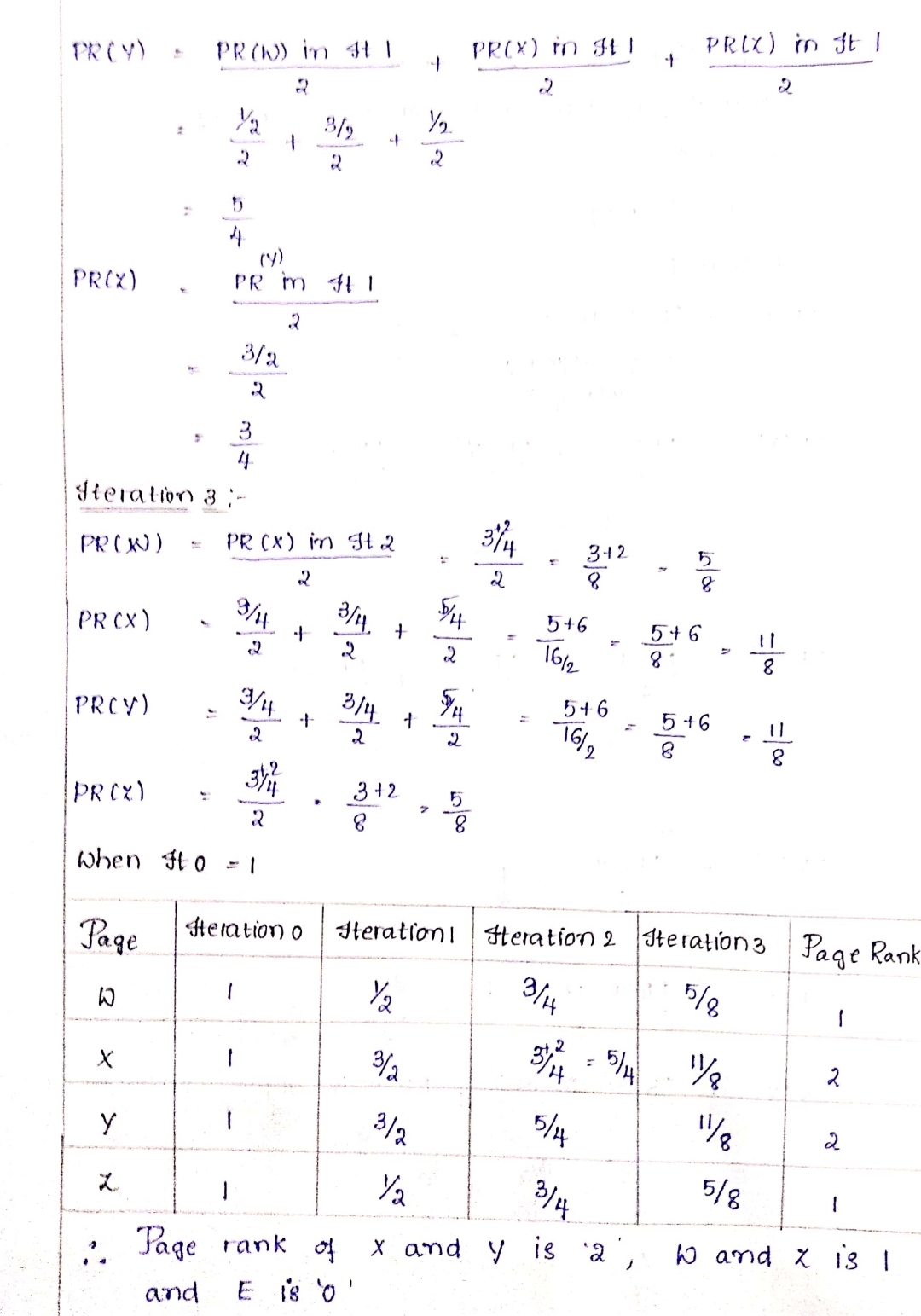
**Suppose the World Wide Web consisted of four web pages labelled W, X, Y, and Z. Suppose that W links to X and Y, X links to W and Y, Y links to X and Z, and Z links to X and Y. Determine the PageRank for each of these five pages, assuming that the PageRank "E" value is set to 0. Show three iterations of the PageRank algorithm, assuming that you start with an initial PageRank of 1 for all four pages.**







s



**QUESTION 3:**

**Implement collaborative filtering technique on certain basket/item data (from Ebay or Amazon, for instance) and prepare a document for the same.**

Recommendation algorithms are best known for their use on e-commerce Web sites,1 where they use input about a customer’s interests to generate a list of recommended items. Many applications use only the items that customers purchase and explicitly rate to represent their interests, but they can also use other attributes, including items viewed, demographic data, subject interests, and favourite artists. At Amazon.com, we use recommendation algorithms to personalize the online store for each customer. The store radically changes based on customer interests, showing programming titles to a software engineer and baby toys to a new mother. The click-through and conversion rates — two important measures of Web-based and email advertising effectiveness — vastly exceed those of untargeted content such as banner advertisements and top-seller lists. E-commerce recommendation algorithms often operate in a challenging environment.

For example:

• A large retailer might have huge amounts of data, tens of millions of customers and millions of distinct catalog items.

• Many applications require the results set to be returned in Realtime, in no more than half a second, while still producing high-quality recommendations.

• New customers typically have extremely limited information, based on only a few purchases or product ratings.

• Older customers can have a glut of information, based on thousands of purchases and ratings.

• Customer data is volatile: Each interaction provides valuable customer data, and the algorithm must respond immediately to new information

A well-developed recommendation system will help businesses improve their shopper's experience on website and result in better customer acquisition and retention.

The recommendation system is designed in 3 parts based on the business context:

Recommendation system part I: Product popularity-based system targeted at new customers

Recommendation system part II: Model-based collaborative filtering system based on customer's purchase history and ratings provided by other users who bought items similar items

Recommendation system part III: When a business is setting up its e-commerce website for the first time without any product rating

When a new customer without any previous purchase history visits the e-commerce website for the first time, he/she is recommended the most popular products sold on the company's website. Once, he/she makes a purchase, the recommendation system updates and recommends other products based on the purchase history and ratings provided by other users on the website. The latter part is done using collaborative filtering techniques.

**RECOMMENDATION SYSTEM PART-1:**

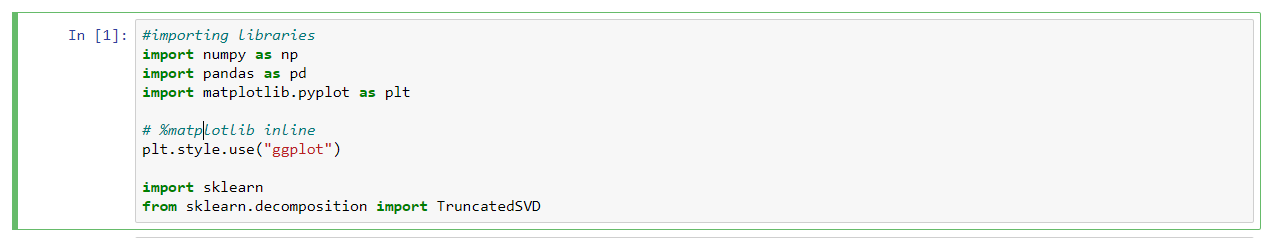
**PRODUCT POPULARITY BASED RECOMMENDATION SYSTEM TARGETED AT NEW CUSTOMERS:**

Popularity based are a great strategy to target the new customers with the most popular products sold on a business's website and is very useful to cold start a recommendation engine.

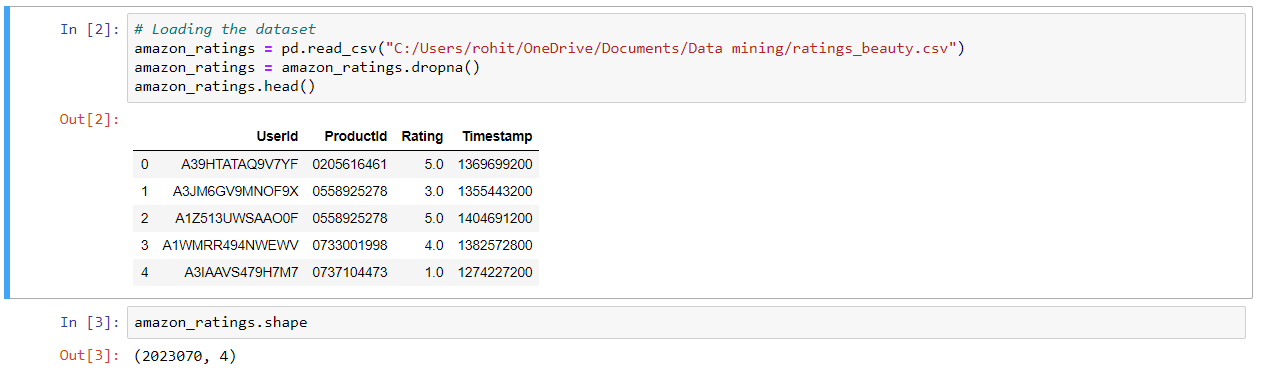
**DATASET:** AMAZON BEAUTY RATINGS

**DATASET LINK:** <https://www.kaggle.com/skillsmuggler/amazon-ratings>

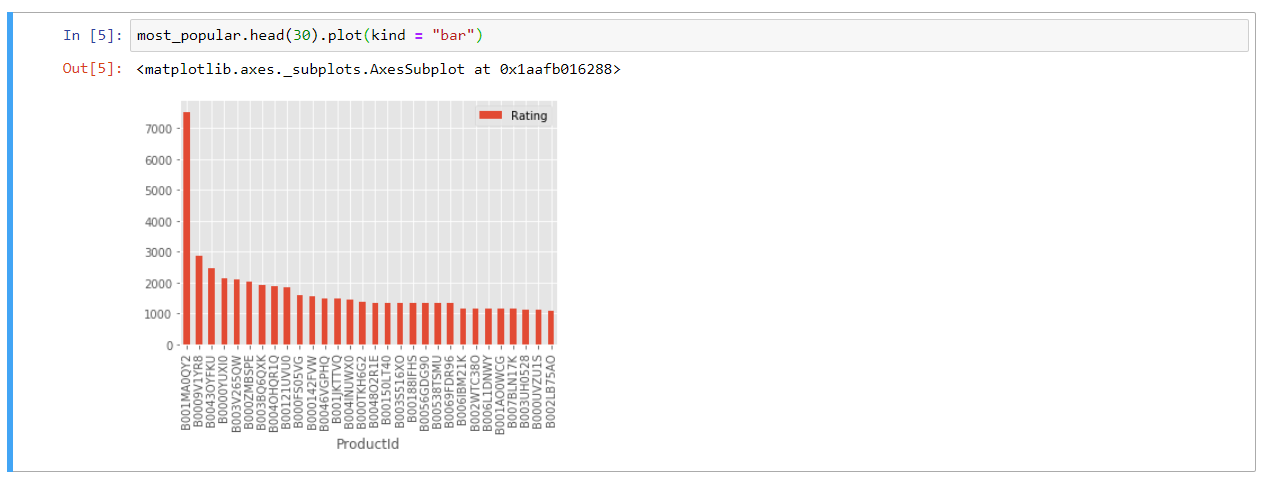
**IMPORTING LIBRARIES:**



**LOADING DATASET:**







**ANALYSIS:**

The above graph gives us the most popular products (arranged in descending order) sold by the business.

For example, product, ID # B001MA0QY2 has sales of over 7000, the next most popular product, ID # B0009V1YR8 has sales of 3000, etc.

**COLLABORATIVE FILTERING SYSTEM:**

Recommend items to users based on purchase history and similarity of ratings provided by other users who bought items to that of a particular customer.

A model based collaborative filtering technique is chosen here as it helps in making predicting products for a particular user by identifying patterns based on preferences from multiple user data.

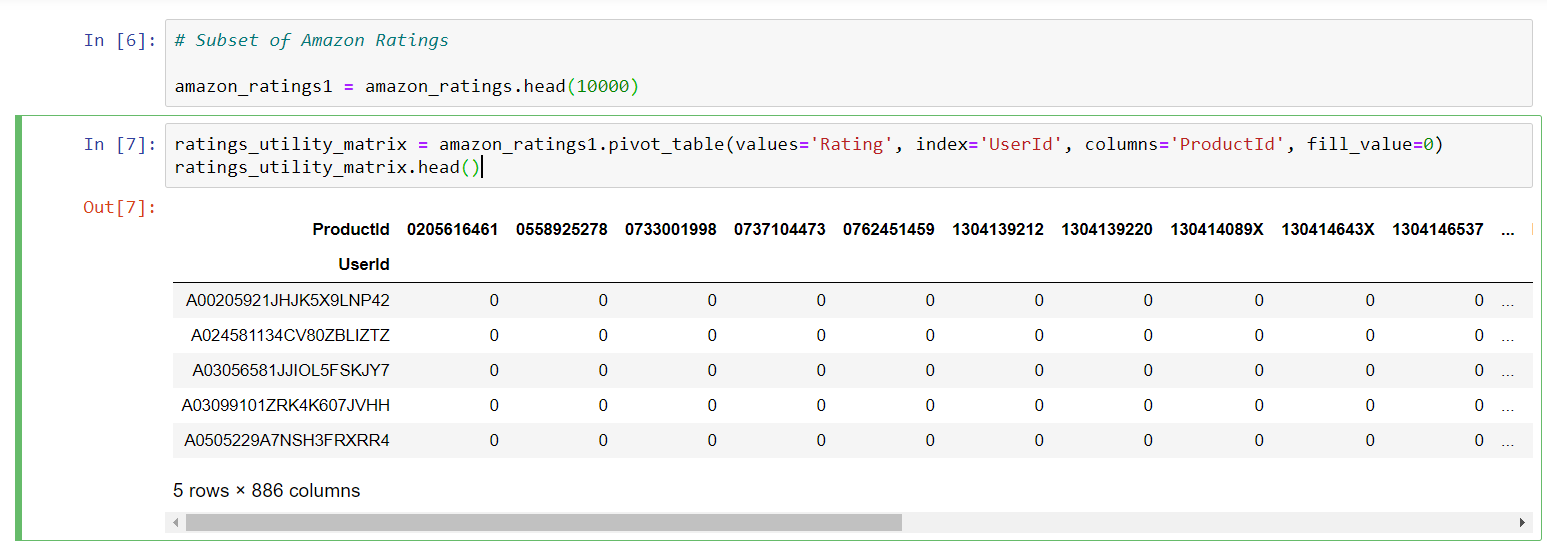
Rather than matching the user to similar customers, item-to-item collaborative filtering matches each of the user’s purchased and rated items to similar items, then combines those similar items into a recommendation list.

To determine the most-similar match for a given item, the algorithm builds a similar-items table by finding items that customers tend to purchase together. We could build a product-to-product matrix by iterating through all item pairs and computing a similarity metric for each pair. However, many product pairs have no common customers, and thus the approach is inefficient in terms of processing time and memory usage.

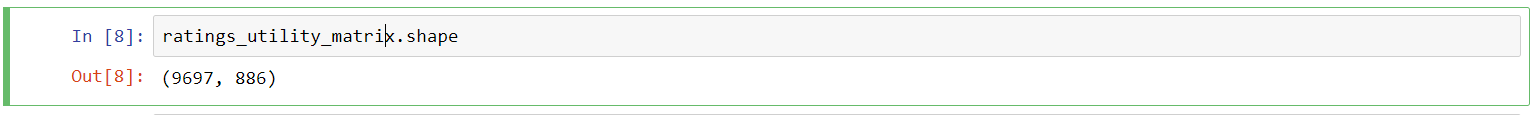
**UTILITY MATRIX BASED ON PRODUCTS SOLD AND USER REVIEWS:**

Utility Matrix

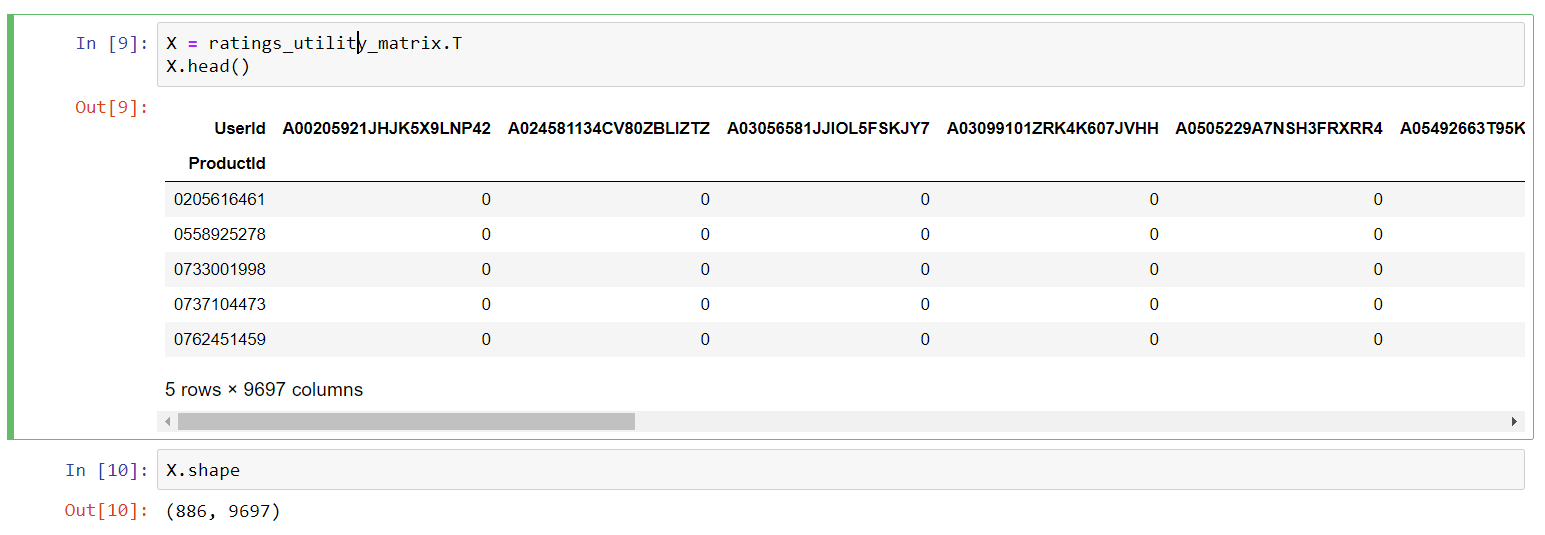
A utility matrix is consisting of all possible user-item preferences (ratings) details represented as a matrix. The utility matrix is sparse as none of the users would buy all the items in the list, hence, most of the values are unknown.



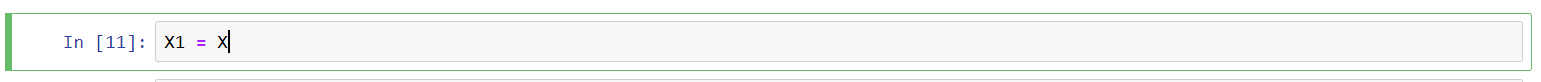
As expected, the utility matrix obtained above is sparse, filled up the unknown values with 0.



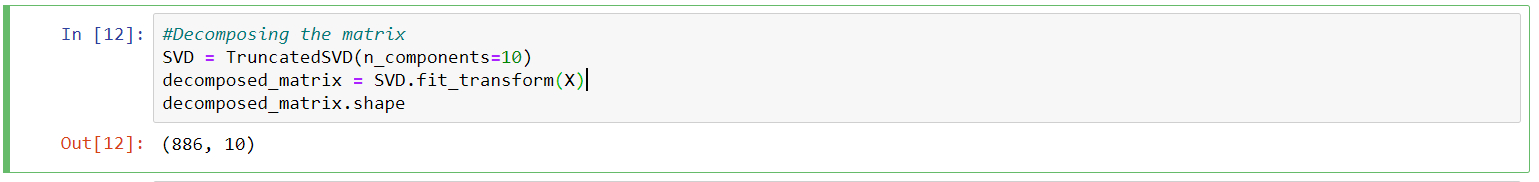
**TRANSPOSING THE MATRIX:**



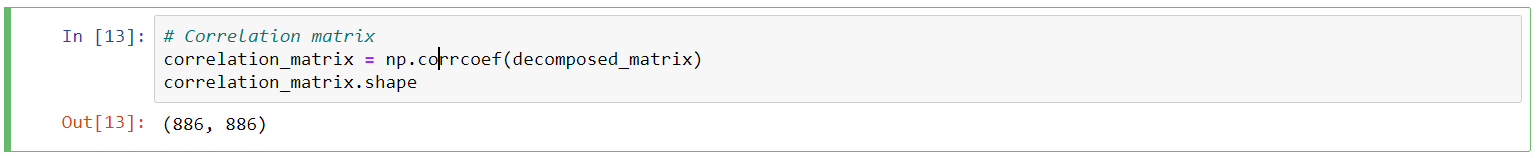
Unique products in subset of data

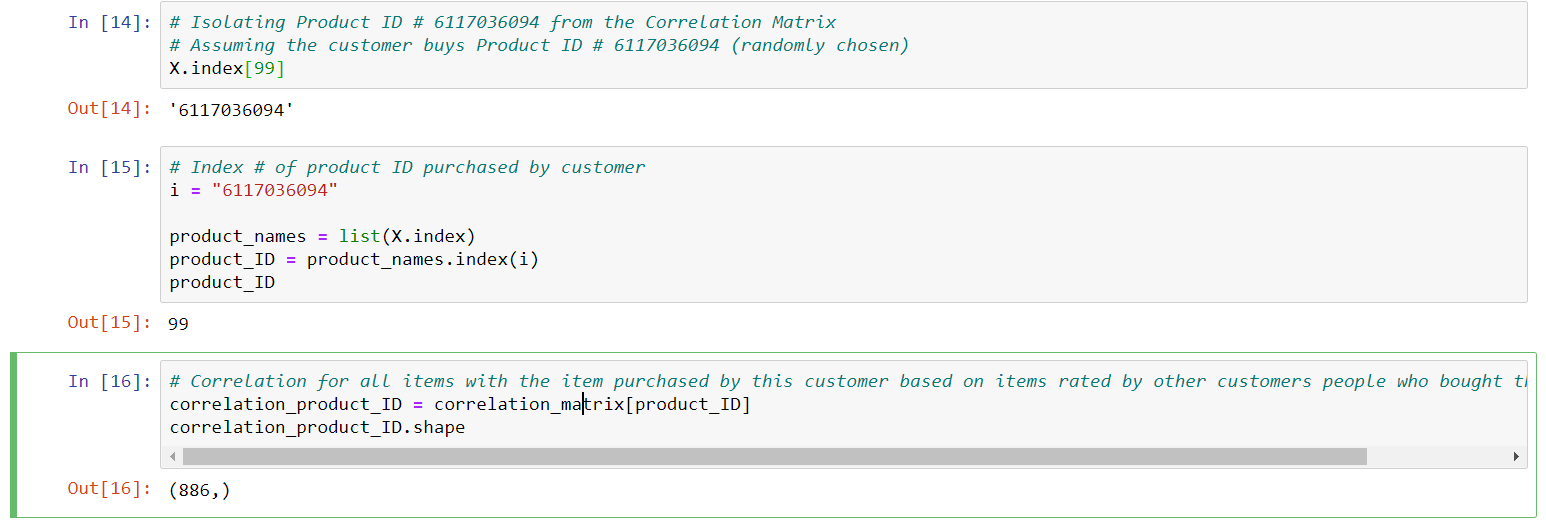


**DECOMPOSING THE MATRIX:**

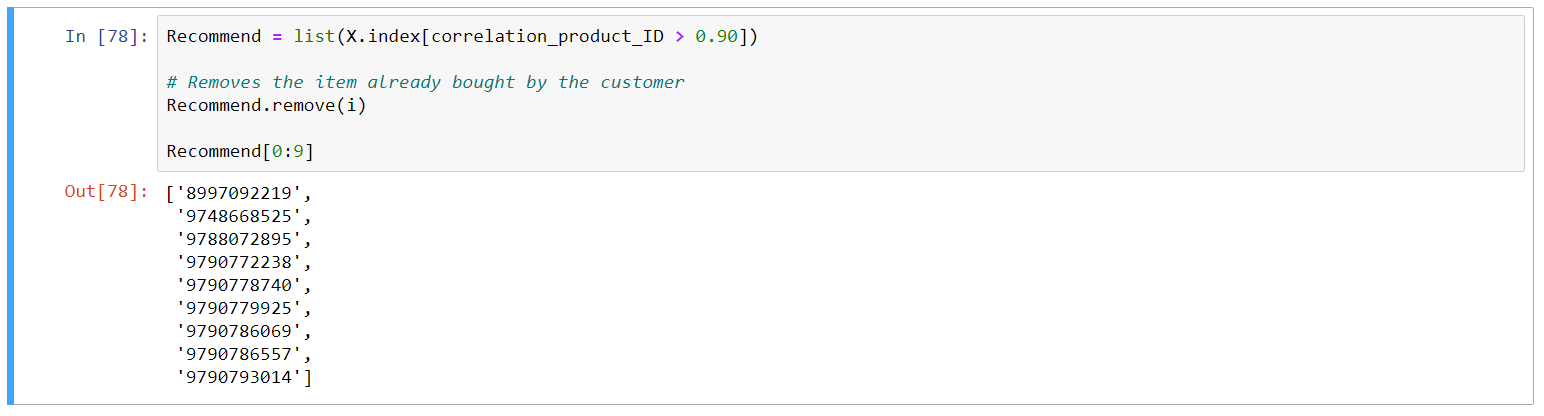


**CORRELATION MATRIX:**





**RECOMMENDING TOP 10 HIGHLY CORRELATED PRODUCTS IN SEQUENCE:**



Product Id #

Here are the top 10 products to be displayed by the recommendation system to the above customer based on the purchase history of other customers in the website.

**RECOMMENDATION SYSTEM PART-III:**

For a business without any user-item purchase history, a search engine based recommendation system can be designed for users. The product recommendations can be based on textual clustering analysis given in product description.

**IMPORTING LIBRARIES:**

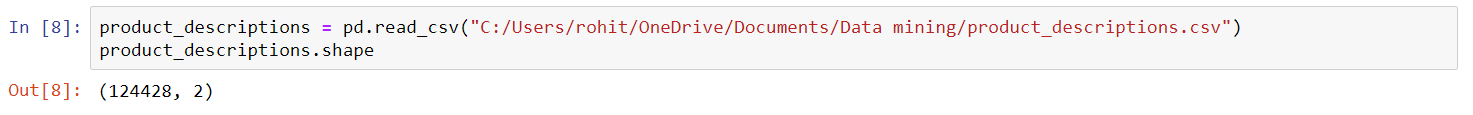


**PRODUCT DATASET LINK:**

<https://www.kaggle.com/c/home-depot-product-search-relevance/data>

**ITEM TO ITEM BASED RECOMMENDATION SYSTEM BASED ON PRODUCT DESCRIPTION:**

Applicable when business is setting up its E-commerce website for the first time

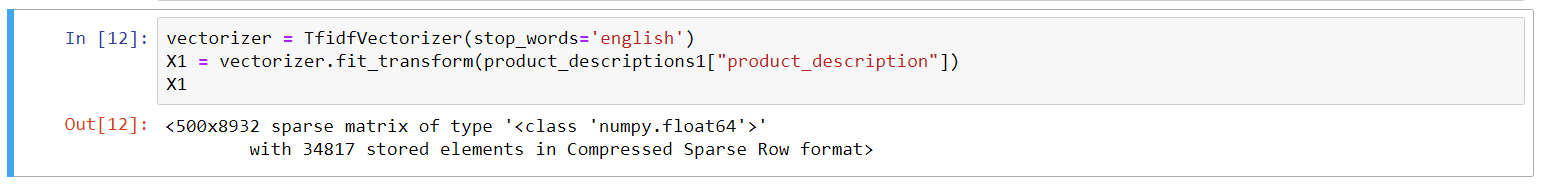


**CHECKING MISSING VALUES:**



**FEATURE EXTRACTION FROM PRODUCT DESCRIPTIONS:**

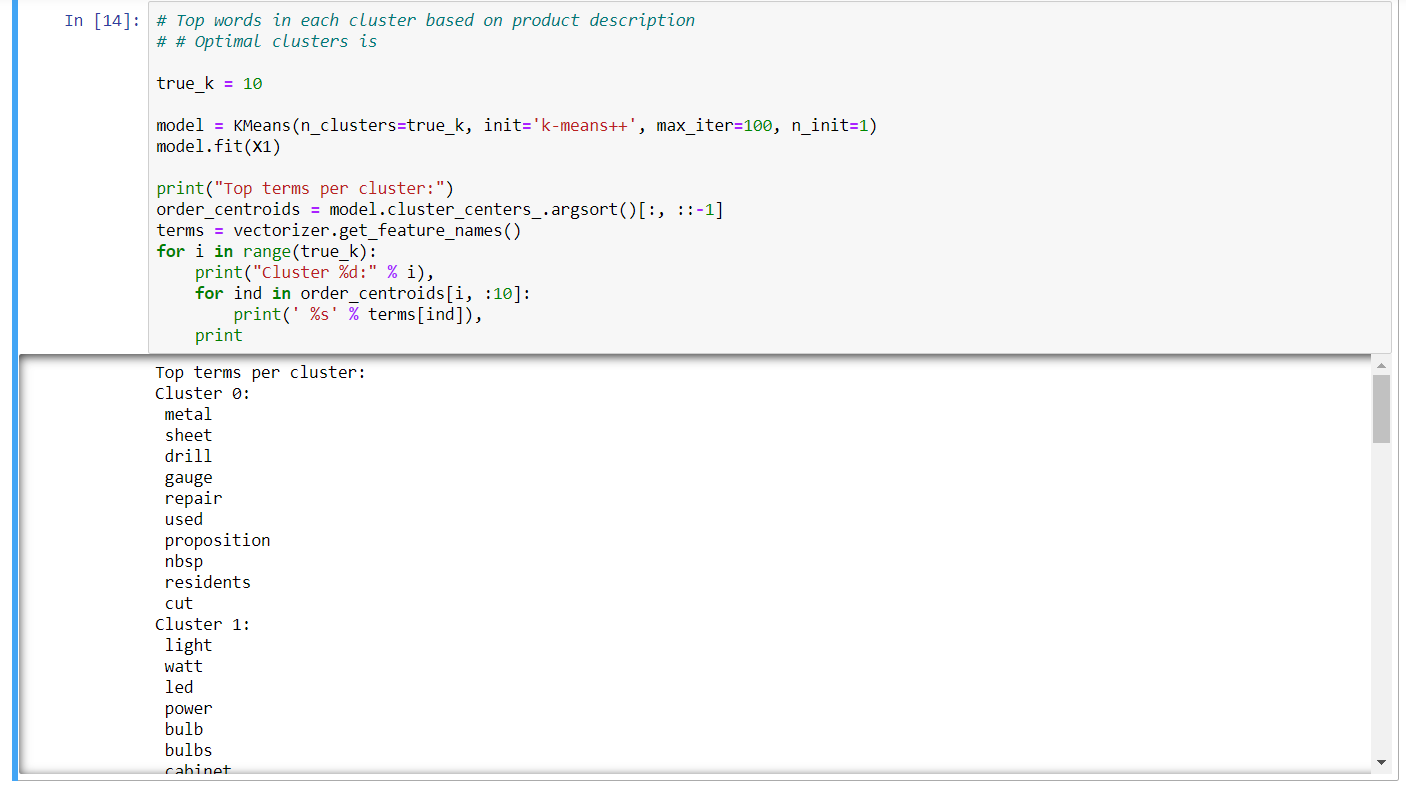
Converting the text in product description into numerical data for analysis



**VISUALIZING PRODUCT CLUSTERS IN SUBSET OF DATA:**



**TOP PRODUCTS IN EACH CLUSTER BASED ON PRODUCT DESCRIPTION:**

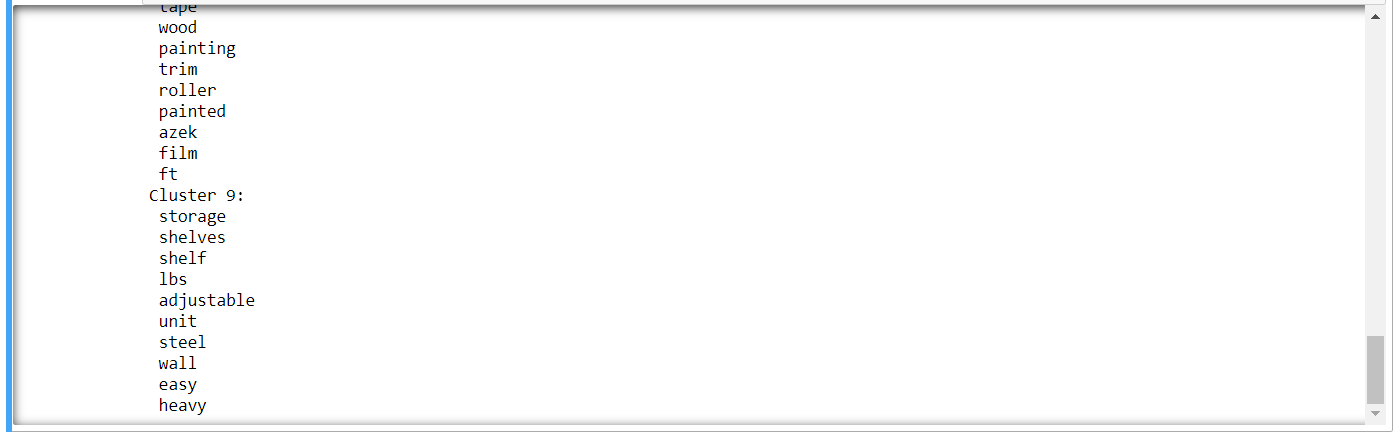












**PREDICTING CLUSTERS BASED ON KEY SEARCH WORDS:**



In case a word appears in multiple clusters, the algorithm chooses the cluster with the highest frequency of occurrence of the word.

Once a cluster is identified based on the user's search words, the recommendation system can display items from the corresponding product clusters based on the product descriptions**.**

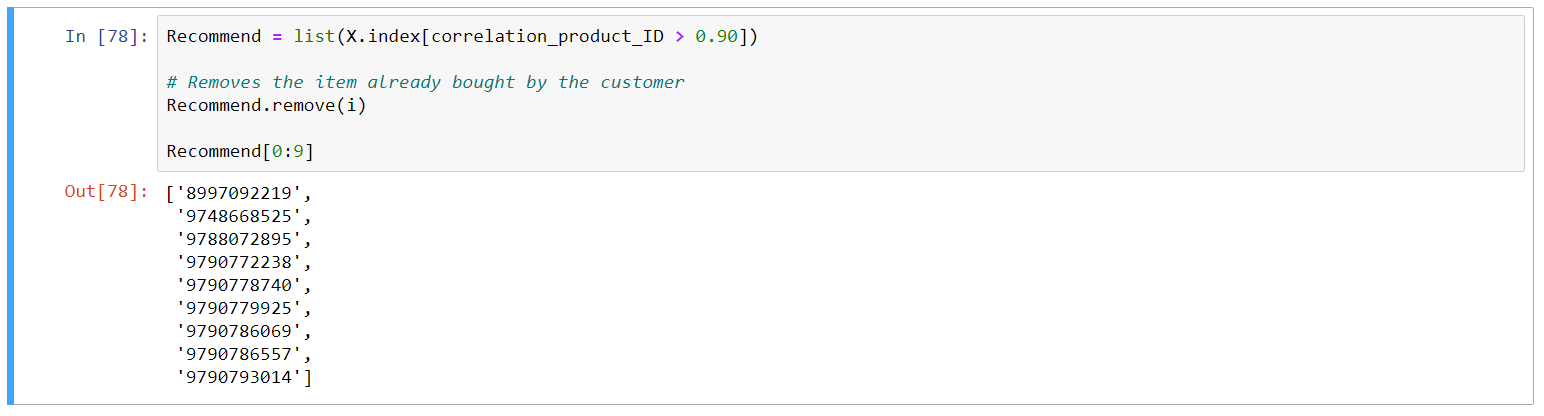
**OVERALL VIEW:**

This works best if a business is setting up its e-commerce website for the first time and does not have user-item purchase/rating history to start with initially. This recommendation system will help the users get a good recommendation to start with and once the buyers have a purchased history, the recommendation engine can use the model based collaborative filtering technique.

The key to item-to-item collaborative filtering’s scalability and performance is that it creates the expensive similar-items table offline. The algorithm’s online component - looking up similar items for the user’s purchases and ratings - scales independently of the catalog size or the total number of customers; it is dependent only on how many titles the user has purchased or rated. Thus, the algorithm is fast even for extremely large data sets. Because the algorithm recommends highly correlated similar items, recommendation quality is excellent. Unlike traditional collaborative filtering, the algorithm also performs well with limited user data, producing high-quality recommendations based on as few as two or three items.

**MODEL BASED COLLABORATIVE FILTERING SYSTEM BASED ON CUSTOMERS PURCHASE HISTORY AND RATINGS PROVIDED BY OTHER USERS WHO BOUGHT ITEMS SIMILAR ITEMS**

Recommending top 10 highly correlated products in sequence based on a customer's purchase history:



**CONCLUSION:**

Recommendation algorithms provide an effective form of targeted marketing by creating a personalized shopping experience for each customer. For large retailers like Amazon.com, a good recommendation algorithm is scalable over very large customer bases and product catalogs, requires only sub second processing time to generate online recommendations, is able to react immediately to changes in a user’s data, and makes compelling recommendations for all users regardless of the number of purchases and ratings. Unlike other algorithms, item-to-item collaborative filtering is able to meet this challenge.

Product Id #: Here are the top 10 products to be displayed by the recommendation system to the above customer based on the purchase history of other customers in the website.