# \

# Assignment Details

## Pre-processing done on data

The dataset file was read transaction by transaction and each transaction was saved as a list. A mapping was created from the unique items in the dataset to integers so that each item corresponded to a unique integer. The entire data was mapped to integers to reduce the storage and computational requirement. A reverse mapping was created from the integers to the items, so that the item names could be written in the final output file.

## Formulas Used

Confidence (X -> Y) = support (X U Y) / support (X)

Support (X, Y) = support-count (X, Y) / total dataset size

We have used support instead of support count because computations with integers are faster than that of floating point numbers.

*Support (X) = Support count (X) / Total number of transactions*

## Results for different for values of support and confidence

|  |  |  |
| --- | --- | --- |
| **Confidence/Support** | **No. of frequent itemsets** | **No of rules** |
| High confidence(MIN\_CONF=0.5)  High support count(MINSUP=60) | 725 | 60 |
| Low confidence(MIN\_CONF=0.1) High support count(MINSUP=60) | 725 | 1189 |
| High confidence(MIN\_CONF=0.5)  Low support count(MINSUP=10) | 11390 | 4187 |
| Low confidence(MIN\_CONF=0.1)  Low support count(MINSUP=10) | 11390 | 35196 |

frequent\_itemset.txt and association\_rules.txt for different MIN\_CONF and MINSUP values can be found in the RESULTS folder

## Observation

Most of the rules we generated have a common item (*whole milk* and *other vegetables*) on the consequent side. This happens when any item is very frequent in the transactions. This can be avoided by using *lift* instead of confidence.

Lift (X -> Y) = support (X U Y) / support (X) \* support (Y)

The purpose of this assignment is to cluster adults using K-means clustering and Hierarchical Agglomerative clustering models and to visualize clusters for predicted and actual cluster labels.

Your dataset is part of "Adult". You can find more information here:

[https://archive.ics.uci.edu/ml/datasets/adult (https://archive.ics.uci.edu/ml/datasets/adult).](https://archive.ics.uci.edu/ml/datasets/adult) The classification problem is whether they earn more than 50,000$ or not.

You need to submit this ipython file after renaming it.

Preprocessing will be needed for the data as most of the data is in string and needs to be quantified.

In

[ ]:

%%javascript

IPython

.

OutputArea

.

prototype

.

\_should\_scroll

**=**

**function**

(

lines

)

{

**return**

false

;

}

# Required Python Packages

In [ ]: *# Import required Python packages here*

*#Seaborn,numpy,pandas,sklearn,matplotlib only*

# Find the best Hierarchical Agglomerative Clustering Model

In this task, you will be performing Hierarchical Agglomerative clustering with different linkage methods (complete and average) and different similarity measures (cosine, euclidean, and manhattan) in order to find the best pair of linkage method and similarity measure. Use F1 score for evaluation and take n\_clusters = 2.

*###################begin code for Task 2-a: Print out a confusion matrix*

*# Import AgglomerativeClustering*

**from** sklearn.cluster **import** AgglomerativeClustering

*# Import pairwise\_distances for calculating pairwise distance matrix* **from** sklearn.metrics.pairwise **import** pairwise\_distances

*# Import f1\_score*

**from** sklearn.metrics **import** f1\_score

*## Calculate pairwise distance matrix for X\_train* pdm\_train **=** **None**

*## Model and fit the training data to the AgglomerativeClustering model*

*## complete linkage + cosine*

*## Model and fit the training data to the AgglomerativeClustering model*

*## complete linkage + euclidean*

*## Model and fit the training data to the AgglomerativeClustering model*

*## complete linkage + manhattan*

*## Model and fit the training data to the AgglomerativeClustering model*

*## average linkage + cosine*

*## Model and fit the training data to the AgglomerativeClustering model*

*## average linkage + euclidean*

*## Model and fit the training data to the AgglomerativeClustering model*

*## average linkage + manhattan*

print("F1-score for complete linkage + cosine", **None**) print("F1-score for complete linkage + euclidean", **None**) print("F1-score for complete linkage + manhattan", **None**) print("F1-score for average linkage + cosine", **None**) print("F1-score for average linkage + euclidean", **None**) print("F1-score for average linkage + manhattan", **None**)

*###################end code for Task 2-a*

REFERENCES

1. [http://www.recommenderbook.net/teachingmaterial/slides](http://www.recommenderbook.net/teaching-material/slides)
2. Recommender Systems Handbook, Ricci, F.; Rokach, L.; Shapira, B.; Kantor, P.B. (Eds.), 2011, Springer.
3. <http://ijana.in/papers/6.11.pdf>
4. [http://www.win.tue.nl/~laroyo/2L340/resources/rec ommender-systems-e-commerce.pdf](http://www.win.tue.nl/~laroyo/2L340/resources/recommender-systems-e-commerce.pdf)
5. Data Mining Concepts and Techniques 2nd Ed By Kamber pages 234-242.