**University of Vienna**
**Faculty of Computer Science**
Prof. Claudia Plant
Ass.-Prof. Sebastian Tschiatschek
Dipl.-Ing. Lukas Miklautz

# Foundations of Data Analysis : Lab Exercise A4

SoSe 2020

## General Remarks:

- The deadline for the submission is at 09:45 a.m. on 18.06.2020. Please upload your solutions on Moodle. No deadline extension is possible.
- After the above deadline, the peer review process on Moodle will start and you have to evaluate two other students' work until 25.06.2020 at 11:59 p.m. You can find all the further details about the peer review process on Moodle.
- For answering the questions, please use Python and utilize the provided Python templates. Your code must be well-documented and readable; see the Style Guide for Python[1] for common style conventions.
- Upload your solutions as a zip archive with the following naming scheme **matrikelnumber_A4.zip**. The archive should contain your code as well as a PDF report with your assumptions, results and a description how to compile and run your code.
- Mark and cite external sources you are using in the code and PDF report.
- If you have problems do not hesitate to contact Claus Hofmann (claus.hofmann@univie.ac.at)/Lukas Miklautz (lukas.miklautz@univie.ac.at) or post a question on the dedicated Moodle forum.

## Task -1 Dimensionality Reduction, Latent Semantic Indexing (30 P)

Latent semantic indexing (LSI) is an indexing and retrieval method that uses the singular value decomposition to identify patterns in the relationships between the terms and concepts contained in unstructured text. In this exercise it is highly recommended to use existing libraries such as `sklearn` and `gensim`[2]. We also provide a python template `lsi_template.py` for the preprocessing of text. This has the advantage, that your results will be comparable to others.

(a) Use the `20newsgroups` dataset and use `gensim` to apply the LSI transformation on `tf-idf` model. Make yourself familiar with the `tf-idf` measure (10 points).

(b) Pick a random document and try to query it with the terms contained in the document. Summarize your findings and interpret the result in the report (20 points).

[1] `https://www.python.org/dev/peps/pep-0008/`
[2] gensim URL: `https://radimrehurek.com/gensim/`

## Task -2 Clustering (30 P)

In this task you try different clustering algorithms and evaluate them using Normalized Mutual Information (NMI) and the Silhouette Score. Use three different datasets: *noCluster2_1K.csv*, *noCluster2_2K.csv*, and *noCluster3_1K.csv*.

(a) Import and plot the data and if necessary preprocess it using *sklearn.preprocessing*.

(b) Try three different clustering techniques such as DBSCAN, KMeans and Average Link, which are already implemented in *scikit-learn*. Please state which algorithms you have used and what parameters you have chosen and why you have chosen them.

(c) Evaluate each clustering technique using Normalized Mutual Information[3] as well as the Silhouette Score. For the evaluation, use the *sklearn.metrics* package.

(d) Briefly explain in your own words what the two evaluation metrics measure and discuss your results, e.g. for which data sets and cluster algorithms do the scores agree or disagree?

## Task -3 Apriori Algorithm for Recommender Systems (40P)

**Task Definition:**
In this programming assignment, you are required to implement the Apriori algorithm and apply it to mine frequent itemsets for movie recommendation. You are required to implement the algorithm from scratch only using native Python libraries and numpy.

**Input:** The provided input file (movies.txt) is based on the MovieLens dataset[1]. It contains the favourite movies of 8892 users. Each line in the file corresponds to a user and represents a list of movies the user likes. An example:

*Avengers: Infinity War - Part II;Jurassic World: Fallen Kingdom*

In the example above, the corresponding user likes the movies "Avengers: Infinity War - Part II" and "Jurassic World: Fallen Kingdom".

**Output:** You need to implement the Apriori algorithm and use it to mine sets of movies that are frequent in the input data. After implementing the Apriori algorithm, please set the relative minimum support to 0.05 and run it on the 8892 lists of movies. In other words, you need to extract all the itemsets that have an absolute support larger than 444.

(a) Output all the length-1 frequent movies with their absolute supports into a text file named "oneItems.txt" and place it in the root of your zip file. Every line corresponds to exactly one frequent movie and should be in the following format:

*Support:movie*

For example, suppose a movie (e.g. Bohemian Rhapsody) has an absolute support 503, then the line corresponding to this frequent item set in "oneItems.txt" should be:

*503:Bohemian Rhapsody*

(10 points)

(b) Please write all the frequent itemsets along with their absolute supports into a text file named "patterns.txt" and place it in the root of your zip file. Every line corresponds to exactly one frequent itemset and should be in the following format:

[3] https://nlp.stanford.edu/IR-book/html/htmledition/evaluation-of-clustering-1.html

*support:movie_1;movie_2;movie_3;...*

For example, suppose an itemset (Bohemian Rhapsody; Aquaman) has an absolute support 446, then the line corresponding to this frequent itemset in "patterns.txt" should be:

*446:Bohemian Rhapsody;Aquaman*

(20 points)

(c) Imagine you should recommend a movie to a user. You know that the user likes the movies "Ant-Man and the Wasp" and "Spider-Man: Far from Home". Based on the result of the apriori algorithm, give a movie recommendation for this user by maximizing the confidence that the user will like the movie. Explain your choice and report the confidence score for your recommendation.
(*Hint*: This task is also possible to do without a finished implementation of the apriori algorithm)

(10 points)

# References

[1] Harper, F.M., Konstan, J.A.: The movielens datasets: History and context. ACM Trans. Interact. Intell. Syst. **5**(4) (Dec 2015). https://doi.org/10.1145/2827872, `https://doi.org/10.1145/2827872`