Laboratory #1

Unsupervised Learning

Contents

[**Redovisning**](#_f3v4of5kc0kh) **1**

[**Inlämning**](#_jv4whhuwotjz) **2**

[**DEADLINE**](#_1a00svy3xarh) **2**

[**Unsupervised Learning**](#_925ps9jjw2xl) **2**

[t1. Agglomerative Clustering [ScreenRec]](#_xv6otdlbl4sg) 2

[t2. k-Means Clustering [ScreenRec]](#_umaekq95oxtg) 3

[**Appendix 1: Data - Agglomerative clustering task**](#_fpsj0s4p7d20) **5**

# 

# Redovisning

Muntlig redovisning under max 5 minuter per uppgift som är markerad med **[ScreenRec]**. Du redovisar genom att använda en sk screen recorder

(streaming skärminspelare). Ange länk till Din produktion i lärplattformen.

* Välj en streaming screen recorder t ex Open Broadcaster Software (OBS) <https://obsproject.com/> eller annan. Du väljer själv om du vill skapa en fil som du skickar in, länkar till eller om du vill publicera på Youtube. Efter att presentationen har kontrollerats kan Du ta bort den permanent. Ange länken i lärplattformen vid den zippade koden.
* Visa och förklara genom demonstration. Beskriv hur Du löste problemen medan du exekverar och demonstrerar Din applikation (Du redovisar alltså hemma genom att spela in/filma Din skärm (inte statiska bilder dvs skärmdumpar), där Du kan visa med muspekare, skriva kod om givetvis tala in ljud.)
* Förklara **sist i videoklippet Din egen reflektion på denna labb**. Vad du tyckte om den, vad var intressant, vad du lärt dig, vad du skulle vilja lära dig mer av, mm mm

# Inlämning

I din inlämning i Learn måste detta finnas med:

1. Källfiler med data
2. En *välkommenterad* Jupyter Notebook **per uppgift** ***t1*** till ***t2*** (\*.ipynb - filer)
   1. Koden måste vara körbar direkt! Tänk på därför på sökvägar
3. Ett filmklipp **per uppgift** (länkar eller videofiler).

# DEADLINE

Enligt Learnplattformen. Sen inlämning räknas till det andra inlämningstillfället av totalt tre.

# Unsupervised Learning

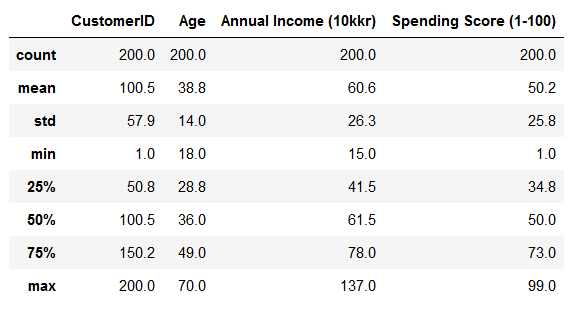
Unsupervised learning problems are about finding patterns. By using it you usually want to discover things you did not know about the data. In the context of reading, describing, analysing and presenting data, the Python package Pandas[[1]](#footnote-1) is very good and de-facto industry standard.

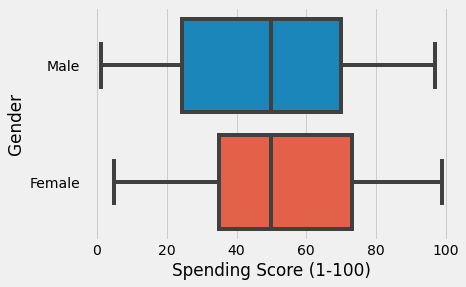
Note: You are to present your findings in max 5 minutes in the screen recording.

## T1. k-Means Clustering [[ScreenRec]](#_f3v4of5kc0kh)

You are working as a data scientist consultant in the retail business. The manager of the store knows that you have been dealing with AI and data science during your studies. She hands over some data to you, see *customerdata4.csv*. The manager group have been wondering about some questions, and she states them to you:

You have to present and discuss at least the following:

1. Using the function describe()[[2]](#footnote-2), display the numbers. See Figure 1. 
2. Using boxplots[[3]](#footnote-3), display the distribution of values in Age, Annual Income and Spending Score according to Gender. Describe what they see, i.e., you need to explain the boxplots. See example in Figure 2.



*Figure 2.*

1. What are your conclusions after analysing **Age** and **Spending Score**? Display data and perform k-means clustering. You must quantitatively argue for your chosen k[[4]](#footnote-4).
2. What are your conclusions after analysing **Annual Income** and **Spending Score**? Display data and perform k-means clustering. You must quantitatively argue for your chosen k[[5]](#footnote-5).

## T2. Agglomerative Clustering [[ScreenRec]](#_f3v4of5kc0kh)

Suppose you work for a wholesale distributing company and your manager asks you to segment the company's clients based on their annual spending on different product categories such as grocery, detergents, paper, etc.

You are handed a dataset, see *t1\_Wholesale customers data.csv*, that includes data of the companies' client's annual spending in monetary units(m.u.). Read the description of the dataset attributes (i.e., features/columns) in [Appendix 1: Data - Agglomerative clustering task](#_fpsj0s4p7d20)

**Your task is to make clusters from this data (using agglomerative clustering) that can segment similar clients together.**

1. Do exploratory analysis using box plot or scatter plot on the columns with m.u

2. Based on the above analysis, choose two columns which you think are suitable for clustering.

3. Use dendrogram to identify optimal number of clusters

4. Perform Agglomerative clustering and explain the clusters and your assumptions

***Hints***

*Hint 1. Clustering visualization:* In real life scenarios, you will probably have more variables in your dataset. As the number of variables increases, visualization becomes more difficult. It is hard to visualize the clusters in high dimension.

You can pick the variables you think are important as per your dataset or task and then visualize those variables (i.e. you can visualize the segmentation of wholesale clients based on their annual spending on Milk and Grocery). But if you want more of an in-depth analysis of the clusters then follow step 2.

*Hint 2. Appending cluster labels to the original dataset:* If you append the cluster labels to the original dataset, you can then derive insights by doing some exploratory analysis of the clustering by looking into patterns in the features relating to the different clusters. (\* Check “dataframe.insert” or “np.insert”)

Good luck!

# Appendix 1: Data - Agglomerative clustering task

The dataset attributes (i.e. features/columns) are described as follows:

FRESH: annual spending (m.u.) on fresh products

MILK: annual spending (m.u.) on milk products

GROCERY: annual spending (m.u.) on grocery products

FROZEN: annual spending (m.u.) on frozen products

DETERGENTS\_PAPER: annual spending (m.u.) on detergents and paper products

DELICATESSEN: annual spending (m.u.) on and delicatessen products

CHANNEL: customers Channel - Horeca (an acronym for HOtel/REstaurant/CAfe) or Retail channel

REGION: customers Region Lisnon, Oporto or Other

The values of product categories 1-6 represent the number of units purchased by each client for each product.

# 

1. See e.g. <https://pandas.pydata.org/docs/user_guide/> , <https://programming-review.com/python/pandas-load-data> [↑](#footnote-ref-1)
2. See e.g. <https://www.tutorialspoint.com/python_pandas/python_pandas_descriptive_statistics.htm> [↑](#footnote-ref-2)
3. See e.g. <https://www.geeksforgeeks.org/box-plot-in-python-using-matplotlib/> [↑](#footnote-ref-3)
4. See e.g. <https://realpython.com/k-means-clustering-python/> [↑](#footnote-ref-4)
5. See e.g. <https://realpython.com/k-means-clustering-python/> [↑](#footnote-ref-5)