Text Analysis for Social Sciences in Python

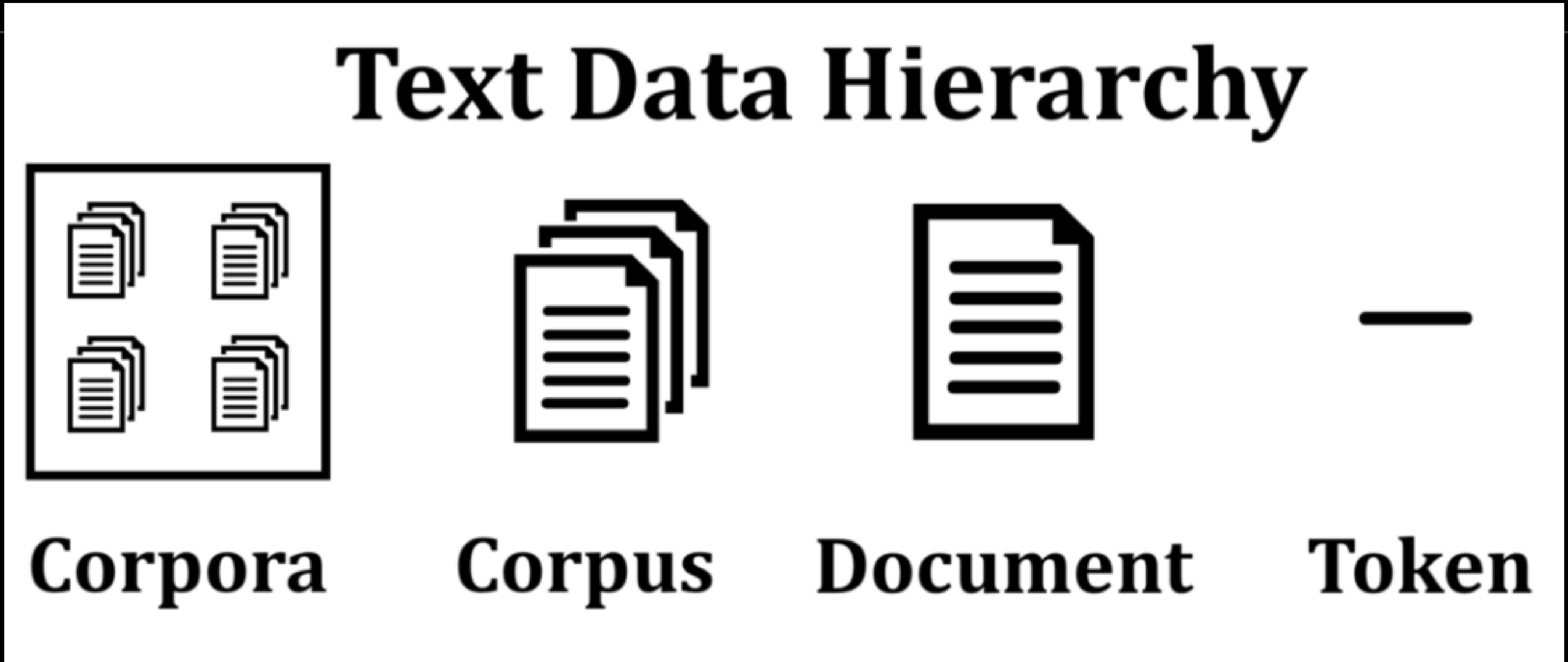
Week 6: Text as Data – Dimensionality Reduction Techniques

WINTER SEMESTER 21/22

HUYEN NGUYEN

HTTP://HUYENTTNGUYEN.COM

Text as Data?



https://medium.com/@raufulazam95218/basics-of-natural-language-processing-nlp-e2b75d2e1dfe

Last week…

Pre-processing data (Amazon Reviews) → Who solved the bug and get the graphs?

Dimension reduction: tf-idf, n-grams

…any questions with the materials?

Consultation Slots for Wednesday 01.12

https://docs.google.com/spreadsheets/d/1E-IFBNDNk7-5GxALoYRfK-P8GO5aYJ8v9j0uEjUsZ5M/edit?usp=sharing

(the same Google Sheet Group Matching link in #group_matching Slack channel, in Sheet “MidtermFeedbackSlot”).

Data-driven dimension reduction techniques

IDEA: Form a small number of linear combinations of predictors from high-dimensional, possibly correlated document-feature matrix

→ use these derived indices as variables in an otherwise standard predictive regression.

NOTE: Features, Dimensions, and Variables are used interchangeably in DS/ML language.

Today’s agenda

1. Data-driven dimension reduction techniques
   - Principal Component Analysis/Regression (PCA/R)
   - Partial Least Squares
2. Training vs. evaluation data sets

1. Principal Component Analysis (PCA)

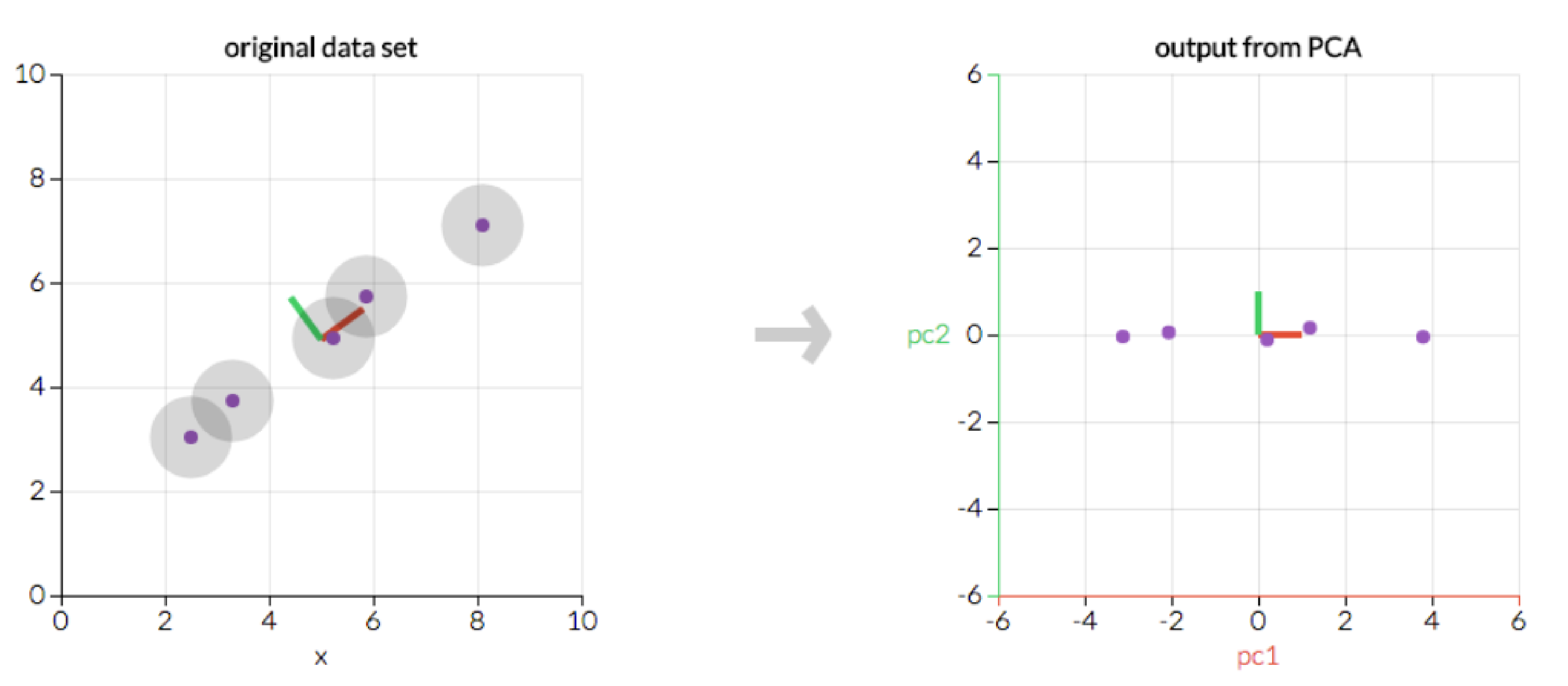
A linear dimensionality reduction technique used to extract information from a high-dimensional space, by projecting it into a lower-dimensional sub-space.

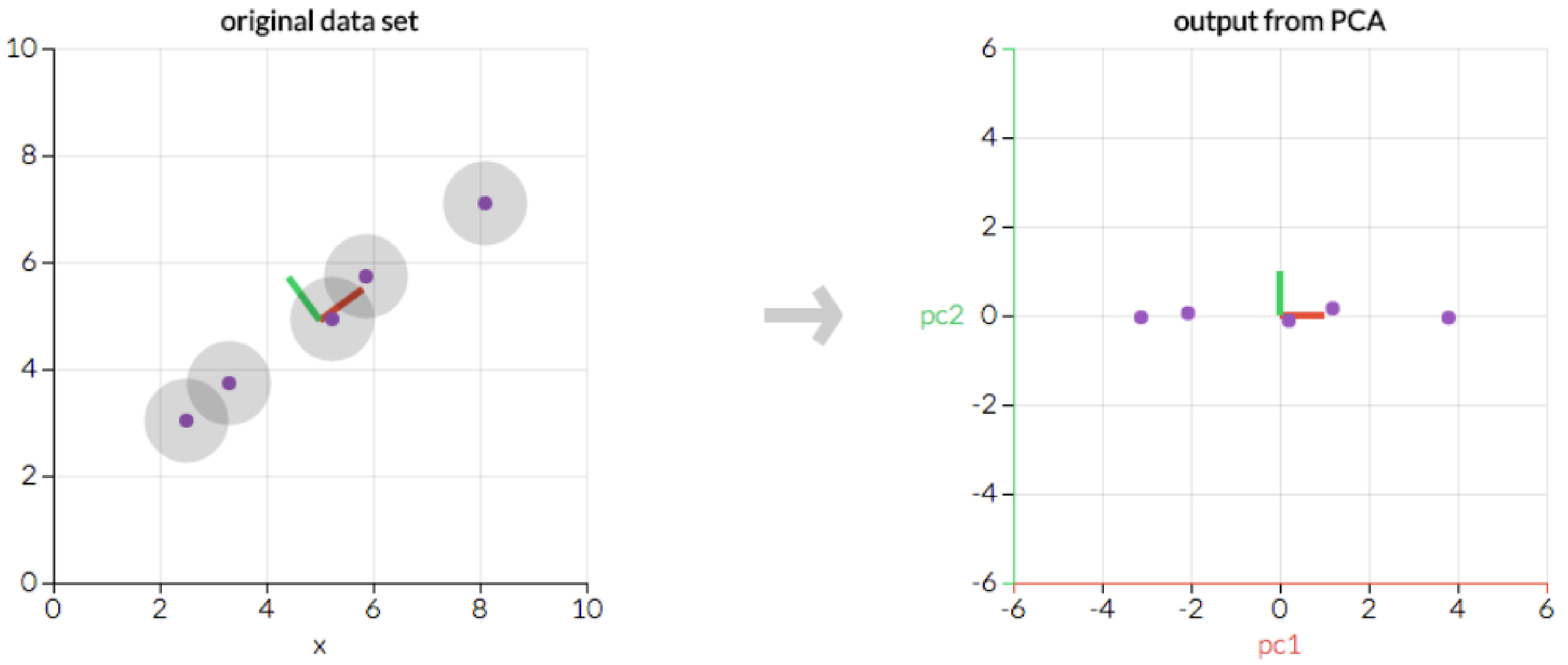
1. Principal Component Analysis (PCA)

A linear dimensionality reduction technique used to extract information from a high-dimensional space, by projecting it into a lower-dimensional sub-space.

HOW? PCA uses an orthogonal transformation to convert a set X of n observations/values of possibly correlated variables (i.e. dimension $X \times n$) into a set of values of linearly uncorrelated variables = principal components (.e. those that explain most of the variability in data).

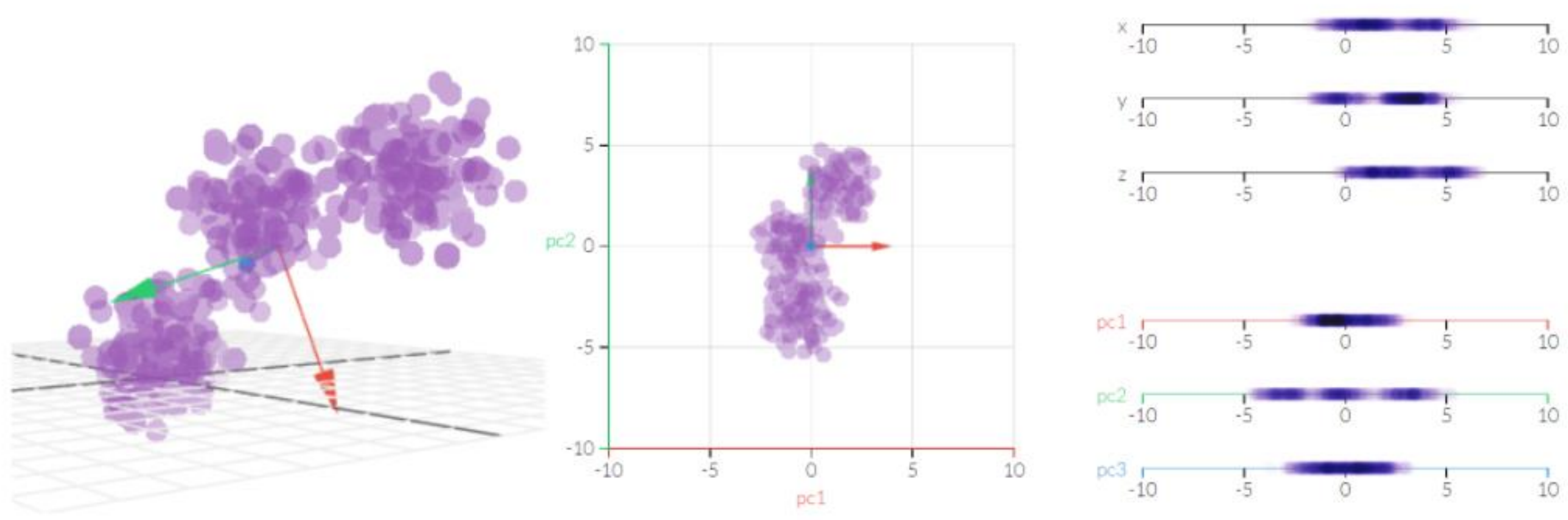
A 2-D example


original data set
10
8
6
4
2
0
0
2
4
6
8
10
x
output from PCA
6
4
2
pc2
0
-2
-4
-6
-6
-4
-2
0
2
4
6
pc1



A 2-D example

A 3-D example



1.1 Principal Component Analysis (PCA)

An unsupervised linear dimensionality reduction technique used to extract information from a high-dimensional space, by projecting it into a lower-dimensional sub-space.

HOW? PCA uses an orthogonal transformation to convert a set X of n observations/values of possibly correlated variables (i.e. dimension $X \times n$) into a set of values of linearly uncorrelated variables = principal components (.e. those that explain most of the variability in data).

→ (1) Solves both multicollinearity and high dimension issues ☺

→ (2) Pre-condition to cluster analysis

1.1 PCR steps

1. Find principal components from the data matrix of original regressors.

1.1 PCR steps

1. Find principal components from the data matrix of original regressors.

2. Regress the outcome variable on the selected principal components, which are covariates of the original regressors. The regression used should be OLS.

1.1 PCR steps

1. Find principal components from the data matrix of original regressors.

2. Regress the outcome variable on the selected principal components, which are covariates of the original regressors. The regression used should be OLS.

3. Transform the findings back to the scale of the covariates using PCA loadings to get a PCR estimator so that regression coefficients can be estimated.

1.1 PCA Assumptions

PCA does NOT require statistical assumptions.

BUT, it ....

- Is sensitive to the scale of the data

  → data must be standardized (standard deviation = 1 and mean =0).

  → column means in the n*p matrix will be 0.

- (data’s PCA) needs to be based on the correlation matrix, NOT on the covariance matrix. (PCA only predicts the observed covariance matrix)

1.1 PCA Assumptions & Pitfalls

PCA is set out to find linear combinations that best describe original regressors WITHOUT a target variable

→ Uncertain that the found principal components are the best to use to predict the target variable.

→ Solution?

1.2 Partial Least Squares (PLS)

A <u>supervised</u> linear dimensionality reduction technique → Computing the new set of features as linear combinations of the original regressors means making use of the target variable

→ Results = explain BOTH the linear combinations of original features AND the response variable.

1.2 PLS pitfalls & assumptions

- ↑ risk of overlooking ‘real’ correlations and sensitivity to the relative scaling of the descriptor variables.
- Difficulty in interpreting the loadings of the independent latent variables
- Unknown distribution properties
- Lack of model test statistics

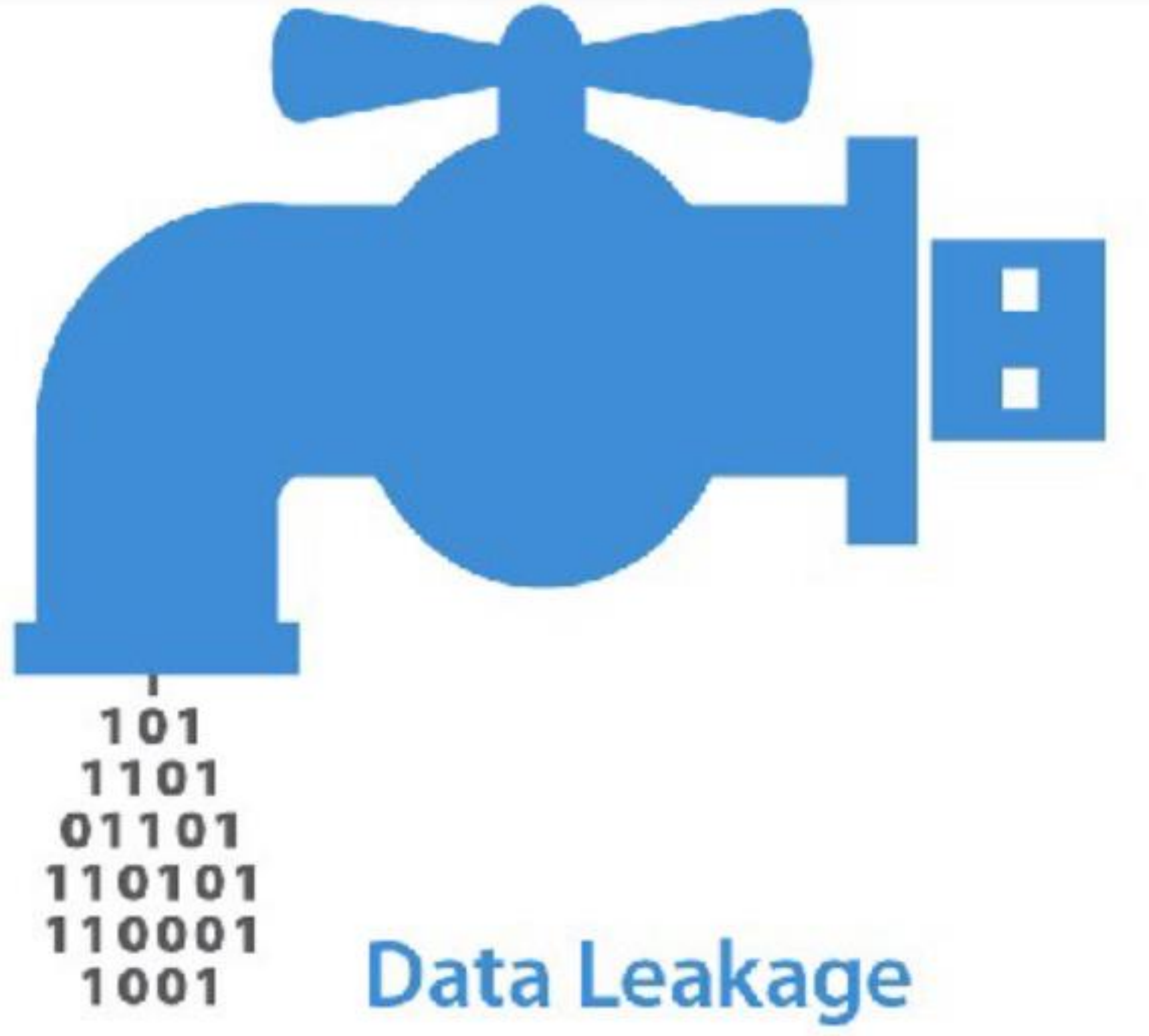
2. Train-Test Split Data Set?

Imagine a scenario where you have tested your model well, and you get absolutely perfect accuracy.

After getting the accuracy, you are happy with your work and say well done to yourself, and then decide to deploy your project.

However, when the actual data is applied to this model in the production, you get poor results.

Why?



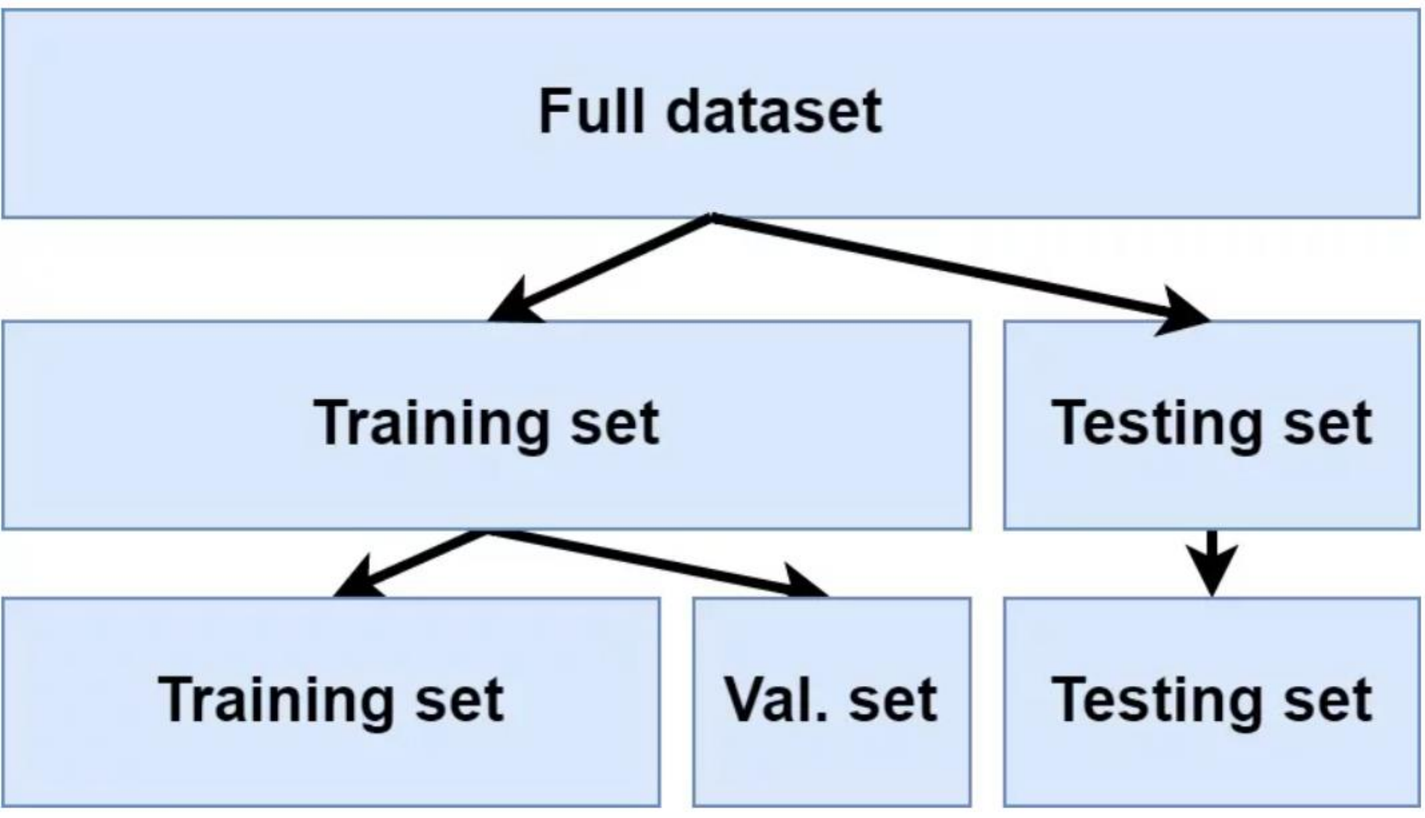
The employed model is already aware of some part of test data after training => overfitting!

Common data leakage:

1) Presence of giveaway features (features from the set of all features that expose the information about the target variable Y and would not be available after the model is deployed.)

2) Leakage during data preprocessing for analysis

2. Train-Test Split Data Set?



Full dataset

Training set

Testing set

Training set

Val. set

Testing set

Why?

The test set is used to simulate the real-world data which is unseen to that model.

For more detailed discussions on solutions and detection of data leakage, please read this post https://www.analyticsvidhya.com/blog/2021/07/data-leakage-and-its-effect-on-the-performance-of-an-ml-model/

Practice time ☺

Open your Google Colab/ Jupyter Notebook

Practice time ☺

https://github.com/httn21uhh/Text-Analysis-for-Social-Sciences-in-Python

→ W6_dimreduction.ipynb + housing_cali.csv

- Download them
- Run the files on your own laptop
- The iPython file is NOT meant for passive scrolling!

This week…

Open your Google Colab/ Jupyter Notebook.

PCR vs. PLS with scikit-learn

- Follow closely the illustrated examples and replicate yourself as the session proceeds.
- Raise your hands to ask questions at any point, including when you think things go too fast/slow for you.