Data-Oriented Programing

Agenda

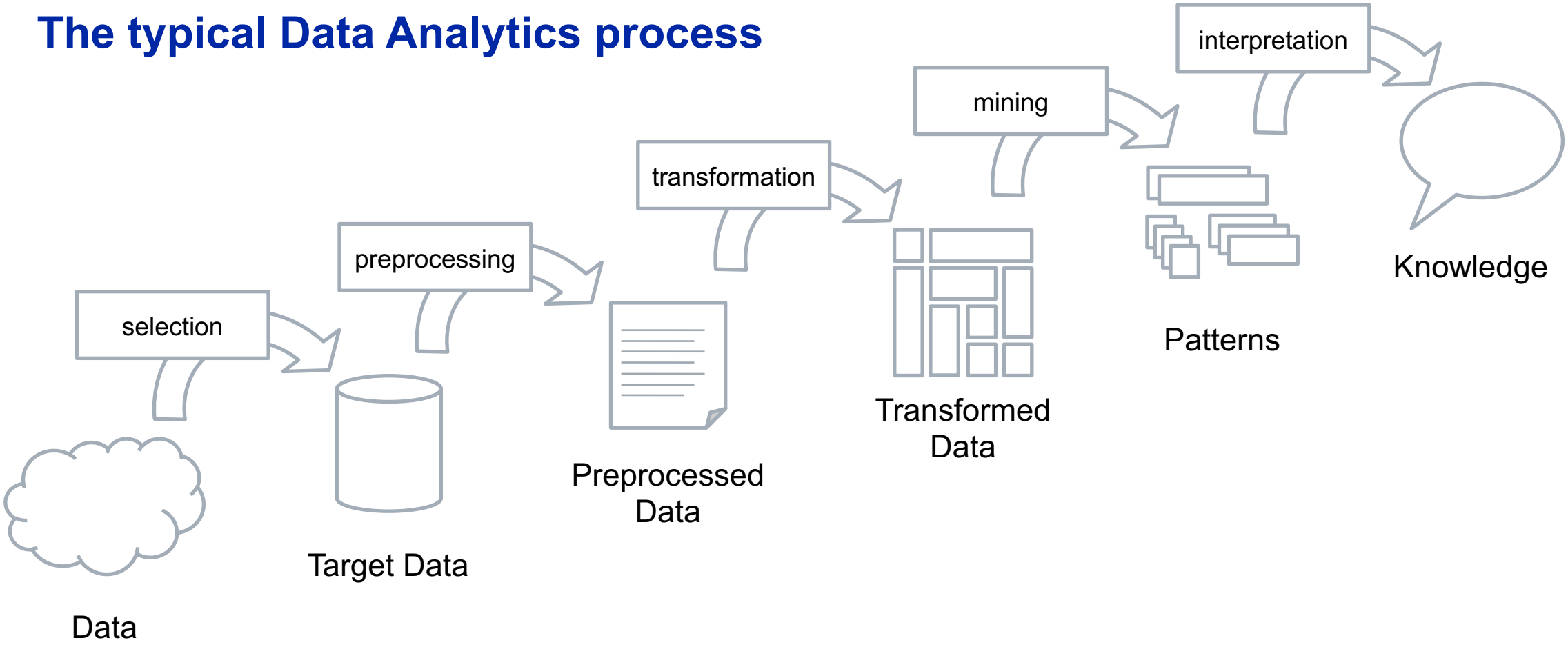
- Data Analysis Primer
- Data Mining and Machine Learning in a Nutshell
- Why Programming for Data Analysis is Different
- A brief introduction to Jupyter
- An even briefer introduction to Pandas, NumPy and scikitl-earn

University of Zurich[UZH]

Department of Informatics

Data Analysis Primer

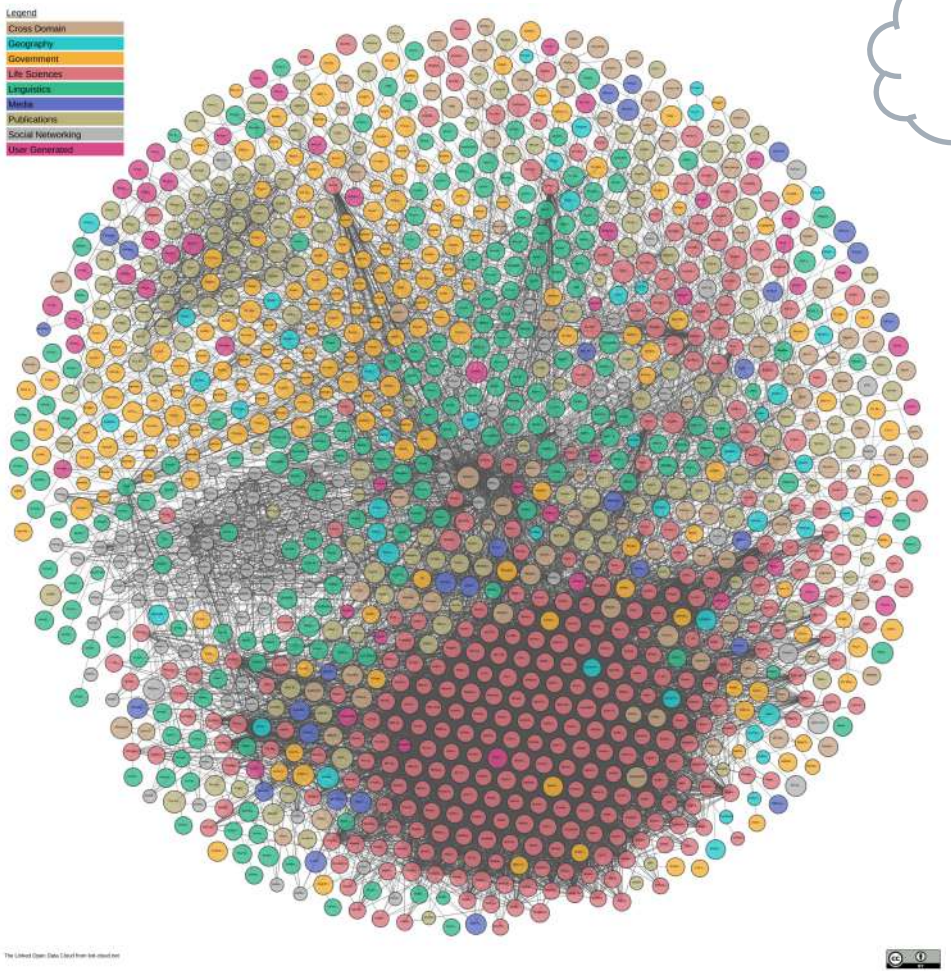
The typical Data Analytics process

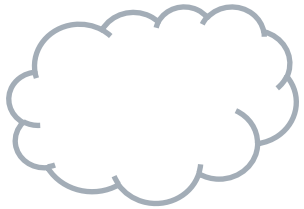


See ***From Data Mining to Knowledge Discovery in Databases,*** *Fayyad et al., 1996*

Step 1: Get some Data

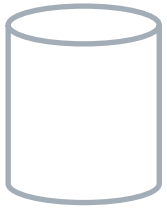
- Public Data sources
  - Open Data
  - Linked Open Data
- Internal Datasources
  - Knowledge Bases, Dokumentation
  - Managerial, Financial, etc.
  - Sensor Measuerements
  - ...
- People
  - Go and talk to them





Data

Step 2: Select the relevant parts



Target Data

Top-Down approach:

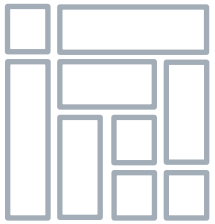
- Formulate the qustion(s) you want to answer
- Break them down into more precise and narrow sub-questions
- Continue until these sub-quesions refer to specific parts of your data
- Use these parts

Bottom-Up approach:

- Explore the datasets you currently have
- Think what questions you could answer using this data

Steps 3 & 4: pre-process and transform

Preprocessed Data



Transformed Data

- Data from different sources might come in various formats
- Some data might be in need of cleaning
  - Missing values
  - Wrong values
  - Inconsistent representation
  - etc.
- All data needs to be transformed into one consistent representation for further analysis

Step 5: ‘mine’

Patterns

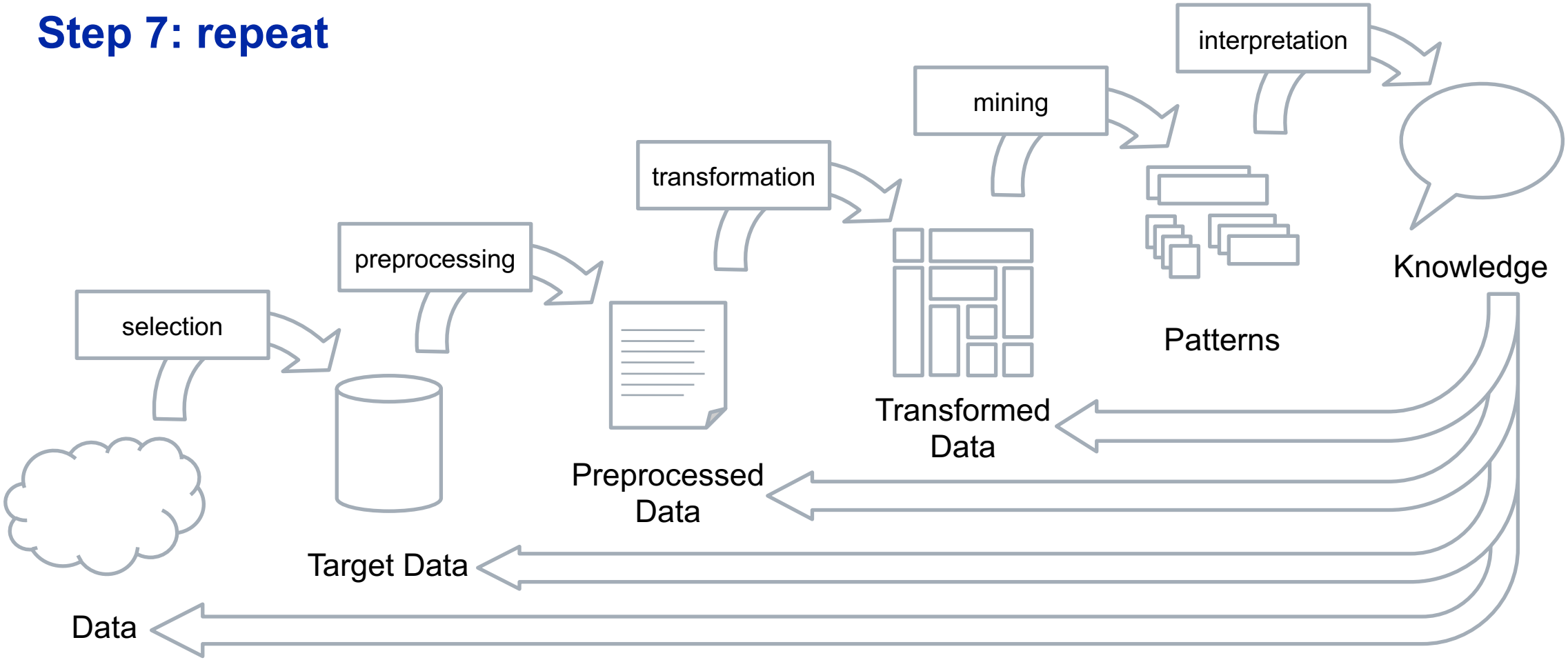
- Develop models which describe the relevant aspects of the data
  - Beware that models are approximations: “All models are wrong, but some are useful”
- Visualize relevant aspects of the data and their relations

Step 6: interpret

Knowledge

- What do the models tell you about your data?
  - How well do the models fit your data?
  - What do the outliers tell you about your data?
- Can you make predictions based on these models?
  - How might the data look in the future?
  - How would certain changes in the data affect the model?
- What are the implications of these interpretations for the source of the data?
  - Do these models actually make sense in the larger context?

Step 7: repeat



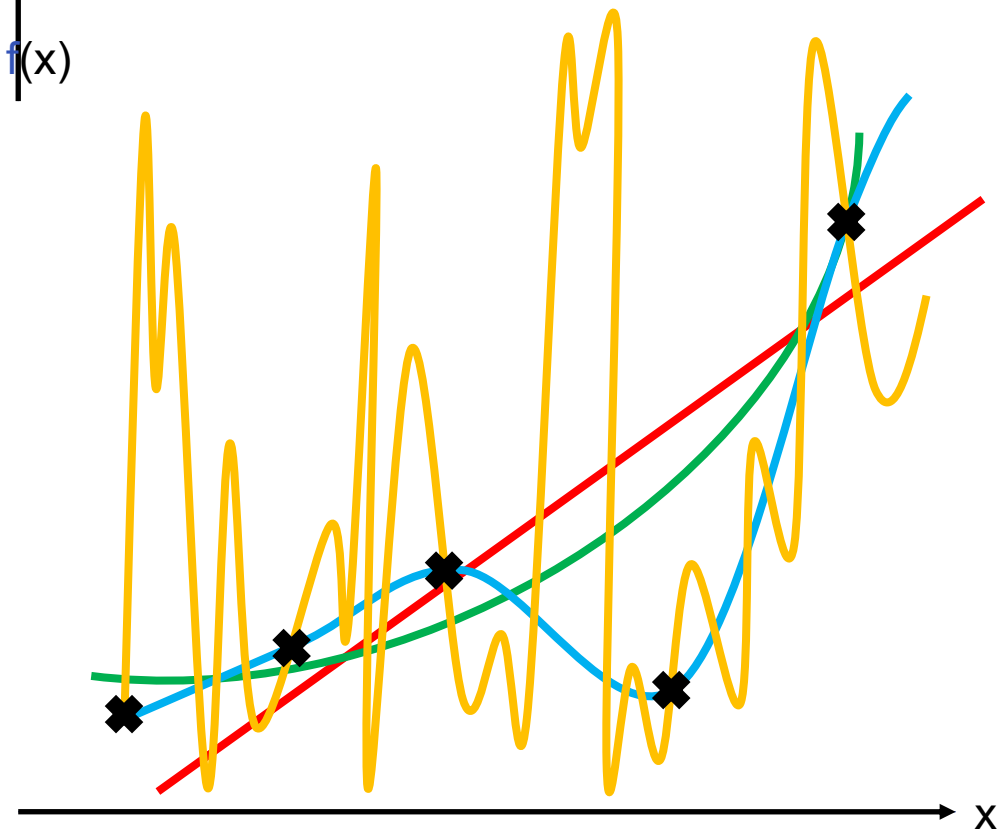
University of Zurich[UZH]

Department of Informatics

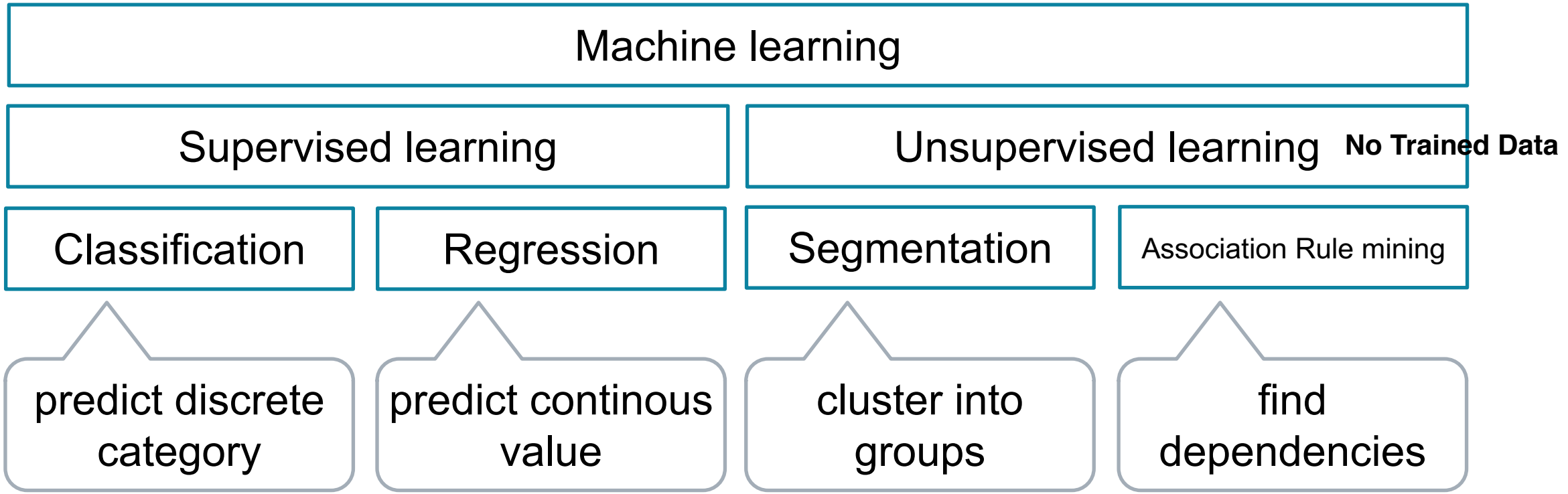
Data Mining and Machine Learning in a Nutshell

Inductive learning

- Construct **hypothesis** h based on **samples** from **unknown function** f
- If h **agrees** with **all known samples** of f, we call it consistent
- There might be arbitrarily many possible h for any given set of samples. Which one is the best?
- Ockham's razor: maximize a combination of consistency and simplicity



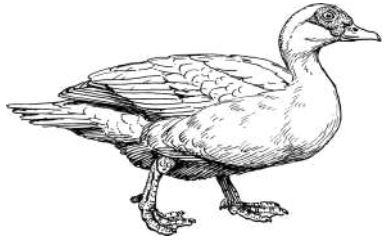
Machine learning methods



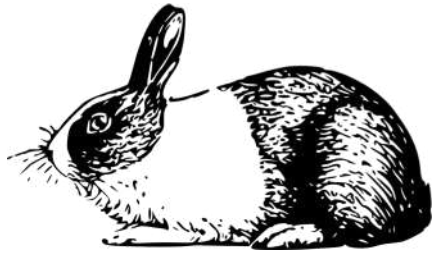
Classification

- Assign one of a set of **known labels** to a piece of data
- Assignment can be hard or probabilistic

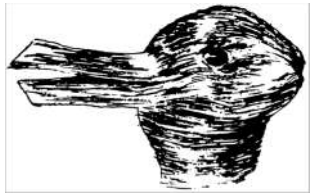
Duck: 43%
Rabbit: 57%



Duck: 92%
Rabbit: 8%

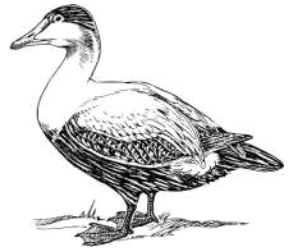


Duck: 4%
Rabbit: 96%



Duck: 58%
Rabbit: 42%

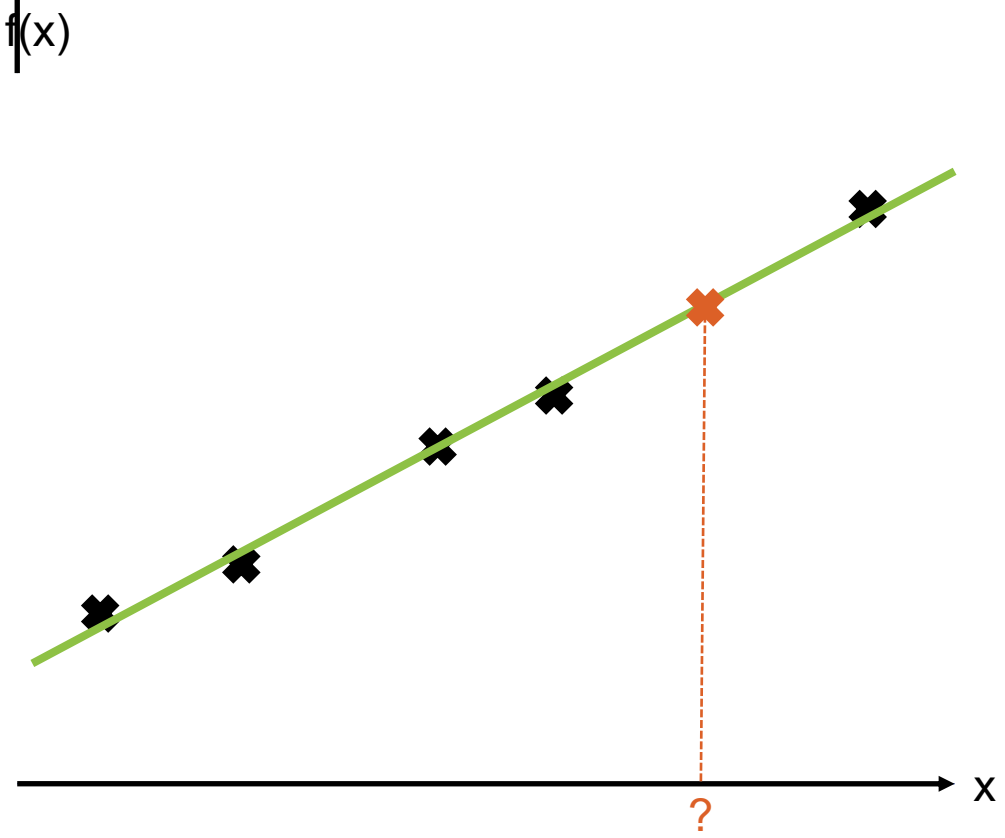
Duck: 3%
Rabbit: 97%



Duck: 99%
Rabbit: 1%

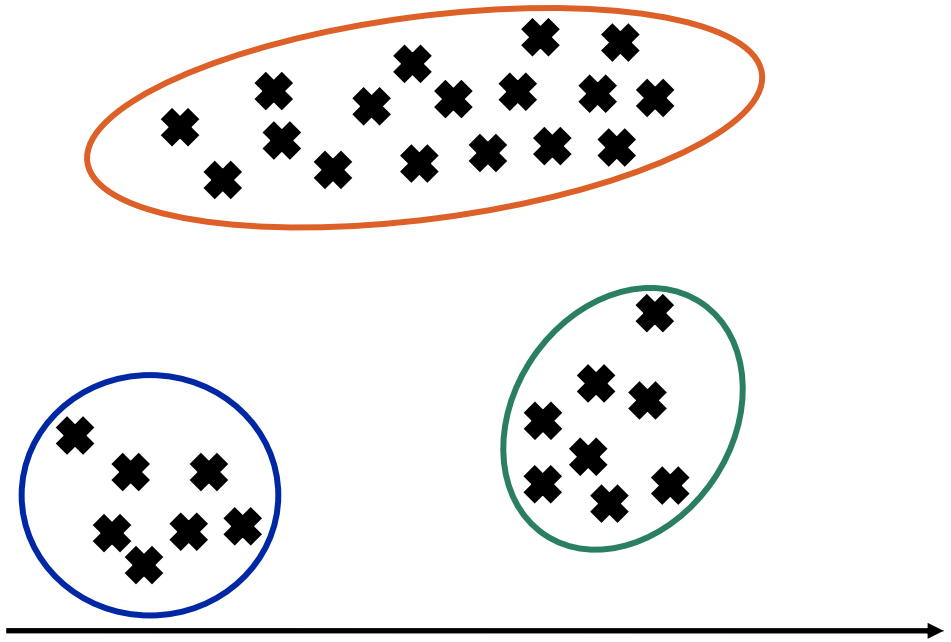
Regression

- Fit a known type of function with some free parameters to known data with minimal error
  - e.g. Linear Regression: f(x) = a*x + b
- Predict unknown aspects of new data



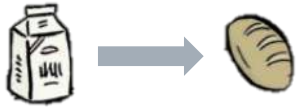
Segmentation

- Similar to Classification, but without knowing the classes beforehand
- Commonly performed by Clustering



Association Rule mining

- Find strong associations / correlations in the data
- Can be used to predict incomplete data or make suggestions



University of Zurich[UZH]

Department of Informatics

Why Programming for Data Analysis is Different

Software Engineering vs Data Analytics Programming

- Long-lived application, designed to be in operation for a long time
- Complex code-base with many files
- Sub-optimal soltions get replaces with better ones and removed from the code
- Code is produced and run once complete
- The Code is the product

- Short-lived application, often used only one
- Small code-base, often only one source-file
- Everything that produces insight stays in the code
- Code is added and evaluated in small increments
- The Knowledge is the product

University of Zurich[UZH]

Department of Informatics

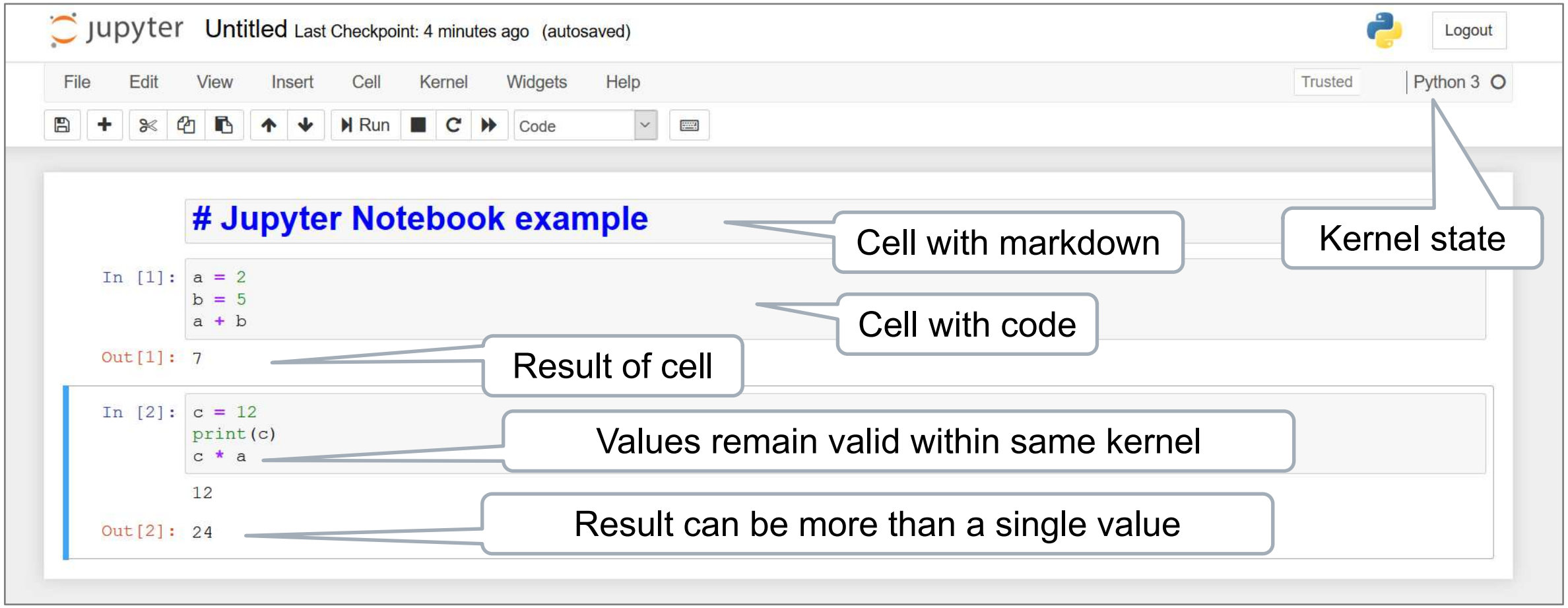
A brief introduction to Jupyter

Jupyter in a nutshell

Jupyter Notebook

- Browser-based development envoronment
- Supports ‘Kernels’ for various programming languages
- Combines REPL-like evaluation with traditional persistent code files
- Supports block-wise evaluation of code
- Displays various types of outputs, including tables, plots, etc.

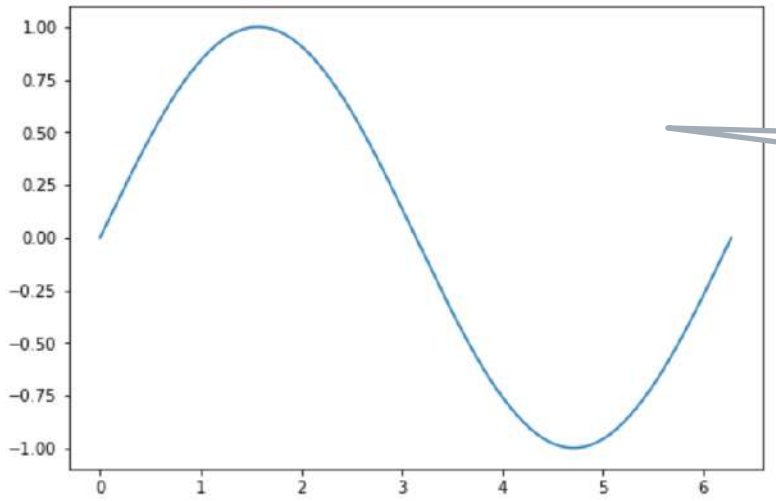
Jupyter Notebook overview



Jupyter Notebook overview

In [4]:
```
from matplotlib import pyplot as plt
import numpy as np
import math
%matplotlib inline
x = np.arange(0, 2 * math.pi, 0.01)
y = np.sin(x)
plot = plt.figure()
axes = plot.add_axes([0, 0, 1, 1])
axes.plot(x, y)
```

Out[4]:
```
[<matplotlib.lines.Line2D at 0x1fe95141550>]
```



Libraries can be used regularly

Output does not need to be text

University of Zurich[UZH]

Department of Informatics

An even briefer introduction to Pandas, NumPy and scikit-learn

Recap: data types in python

- Boolean: True
- Integer: 5
- Float: 5.0
- String: “5”
- List: [4.0, 5, “6”]
  - List of Lists: [[3, “4”], [5.0, 6], [“7”, True]]

- What about tables?
- What about matrices?

Pandas

- Python library for data manipulation and analysis
- Implements table-like structure called **DataFrame**
- Many functions for manipulating DataFrames
  - Selecting and Projecting
  - Grouping and Aggregating
  - Slicing and Merging
  - Loading and Storing
  - ...

Pandas in action

- Creating a DataFrame
  - `df = pandas.DataFrame({'column1': [1,2,3], 'column2': [2,4,6]})`
- Loading a DataFrame from a file
  - `df = pandas.read_csv("titanic.csv")`
- Accessing a row
  - `df[2]`
- Accessing a column
  - `df['column1']`

	column1	column2
0	1	2
1	2	4
2	3	6

Pandas in action

- Access a range of rows
  - `df[2:5]`
- Conditional selection
  - `df[df['column1']>=2]`
- Some statistics
  - `df.mean()`

NumPy

- Python library for high-level mathematical functions
- Implements ndarray datastructure which is similar in some ways to a list
- ndarrays often also used with more than one dimension (matrix math)
- Many functions for manipulating ndarrays

Scikit-learn

- Python library for machine learning
- Implements various algorithms for
  - Classification
  - Regression
  - Clusering
  - etc.
- Comes with all the required functionality for fitting and evaluating models

Scikit-learn in action

- Create a new model
  - `lm = LinearRegression()`
- Fit model to training data
  - `lm.fit(independent_variables, dependent_variables)`
- Predict unknown dependent value
  - `lm.predict(independent_value)`