Lecture 1

Introduction to data mining

Part B

Ioanna Miliou, PhD

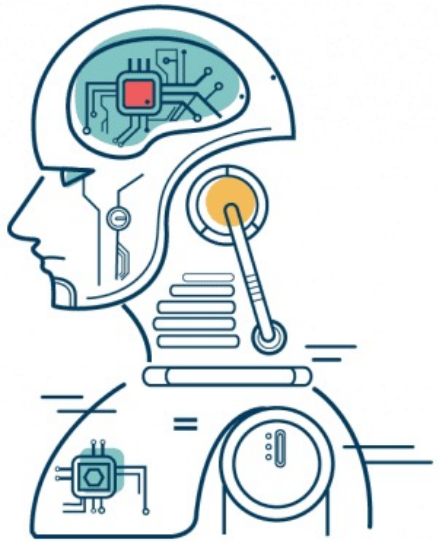
Senior Lecturer, Stockholm University

Stockholms universitet

Machine Learning

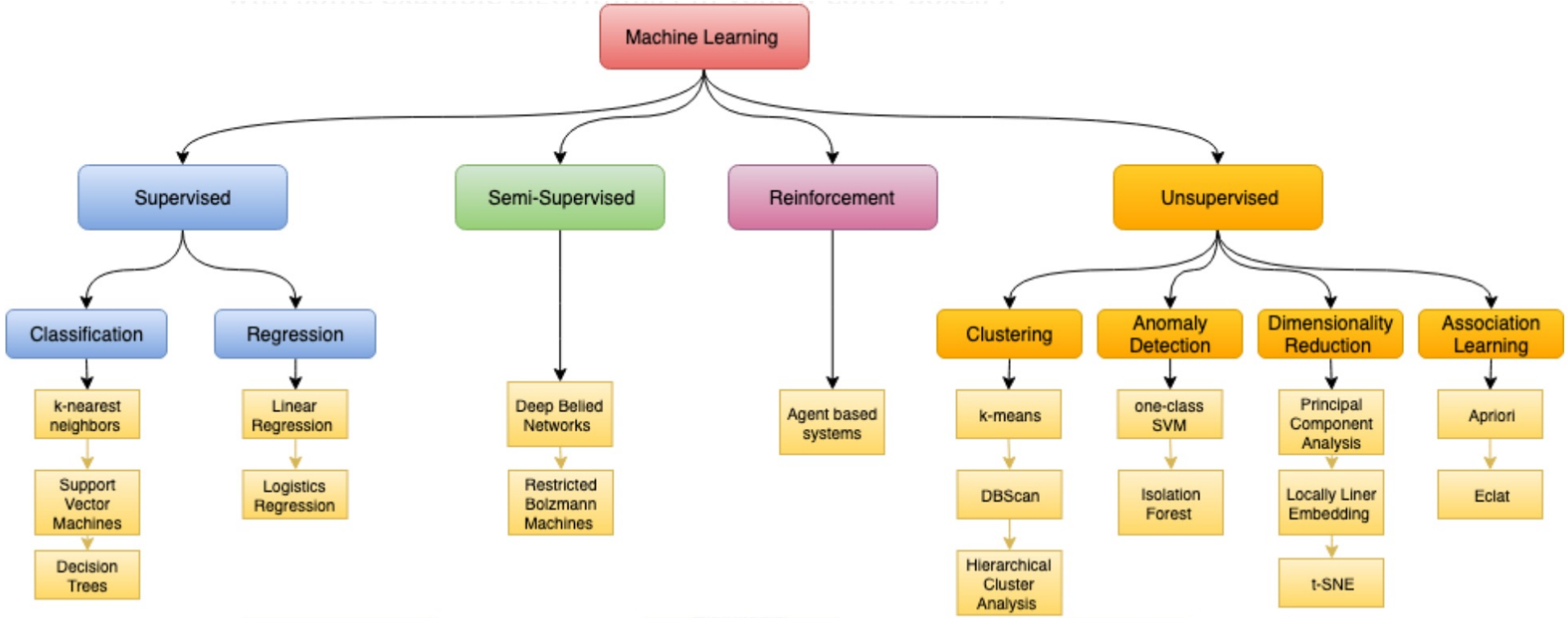
"Learning is any process by which a system improves performance from experience."

- Herbert Simon



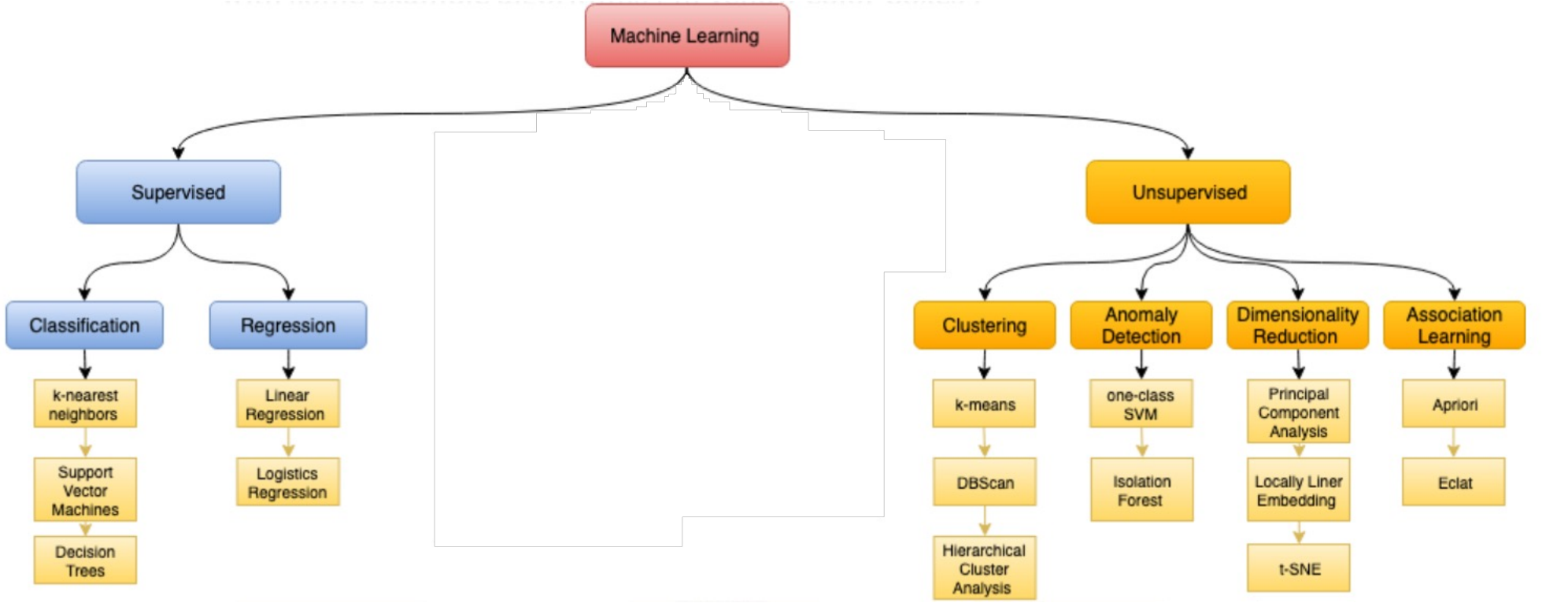
Stockholms universitet

Types of Machine learning



Stockholms universitet

Our focus

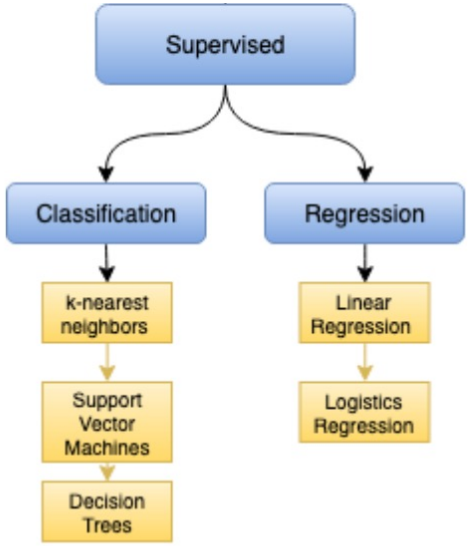


Task Driven
(Predict next value)

Data Driven
(Identify Clusters)

Stockholms universitet

Supervised learning



Experience: objects that have been assigned **class labels**
Performance: typically concerns the ability to **classify** new (**previously unseen**) objects

Predictive data mining

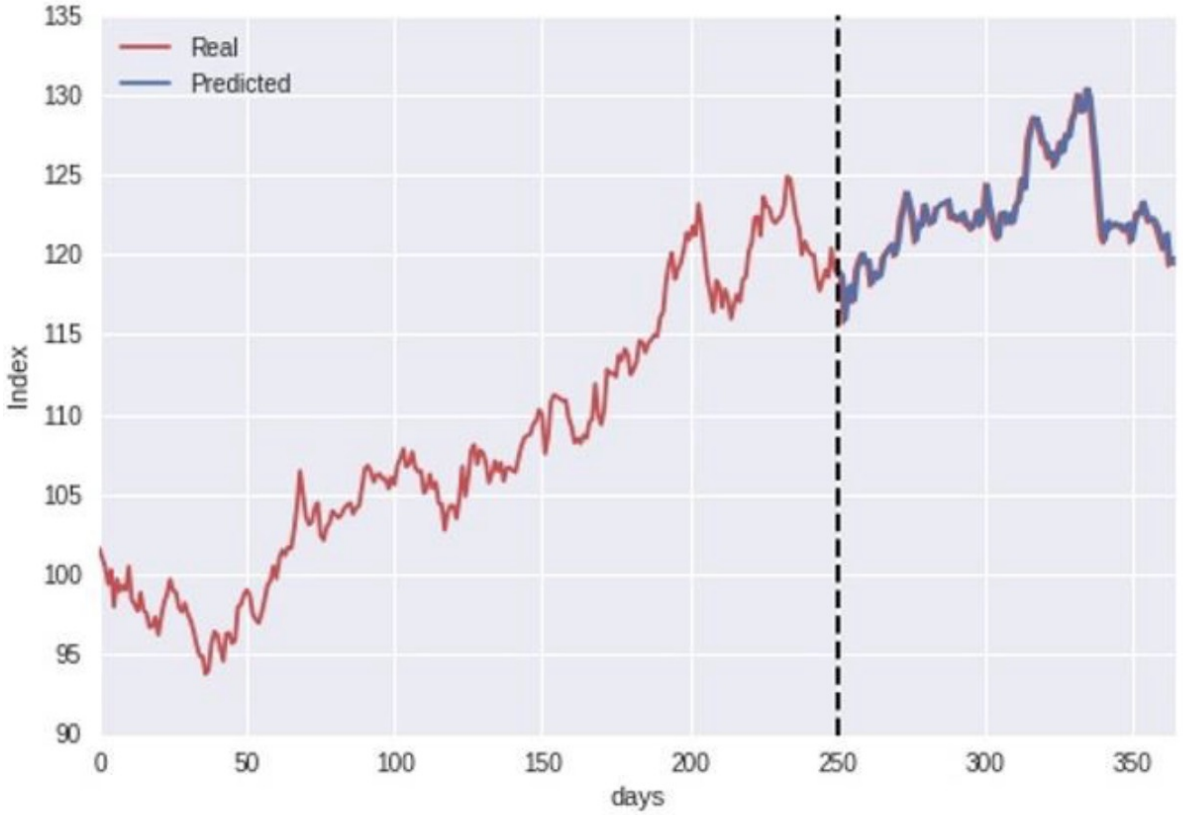
Stockholms universitet

Predictive analytics

- Extract rules
  - If occupation = banker, then salary > 60K SEK per month
- Identify customers who will churn
  - If John stays on *level 40 of Candy crush* for *more than 2 days*, there is an 85% chance that *he will stop playing*
- Predict the effectiveness of the treatment of a patient:
  - If a patient is given *"beta blockers"* and *"inhibitors"* after heart failure, then the chance of survival in 1 year is 90%

Stockholms universitet

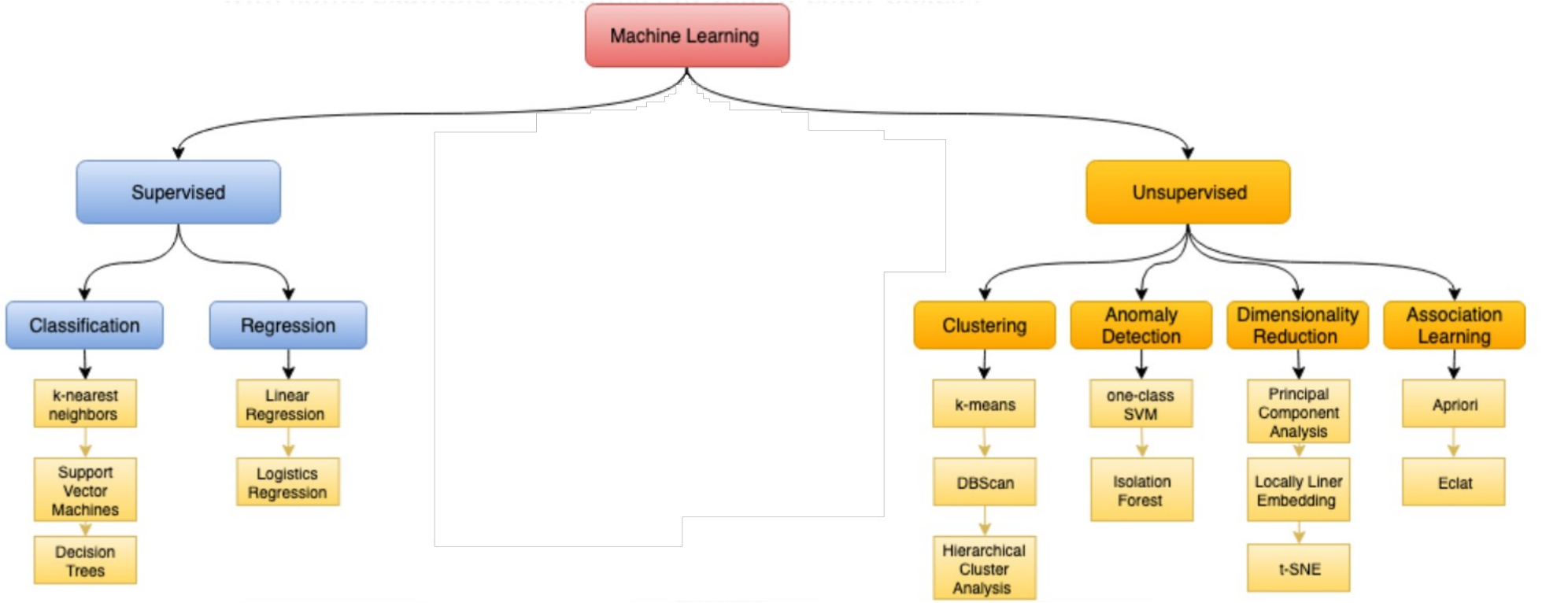
Time series prediction



- Energy comsumption
- Fault detection
- Time to next failure

Stockholms universitet

Our focus



Task Driven
(Predict next value)

Data Driven
(Identify Clusters)

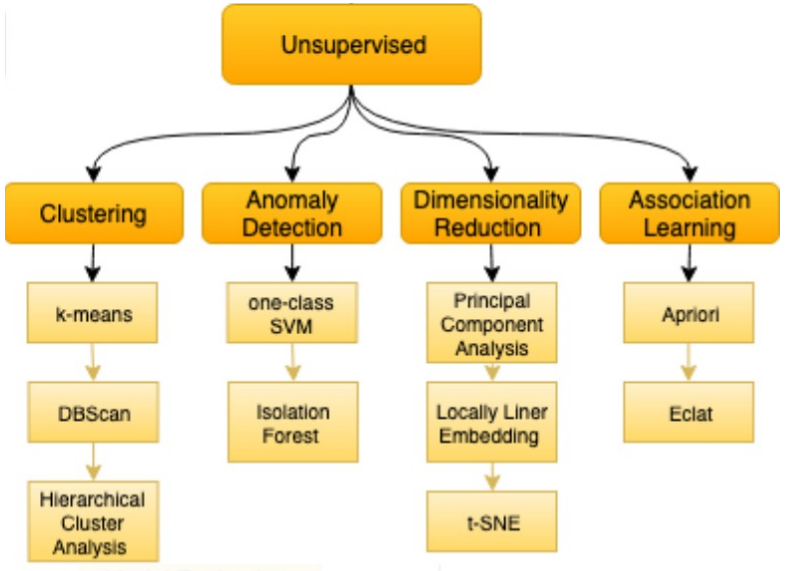
Stockholms universitet

Unsupervised learning

Experience: objects for which **no class labels** have been given

Performance: typically concerns the ability to output useful **characterizations** (or groupings) of objects

Descriptive data mining



Stockholms universitet

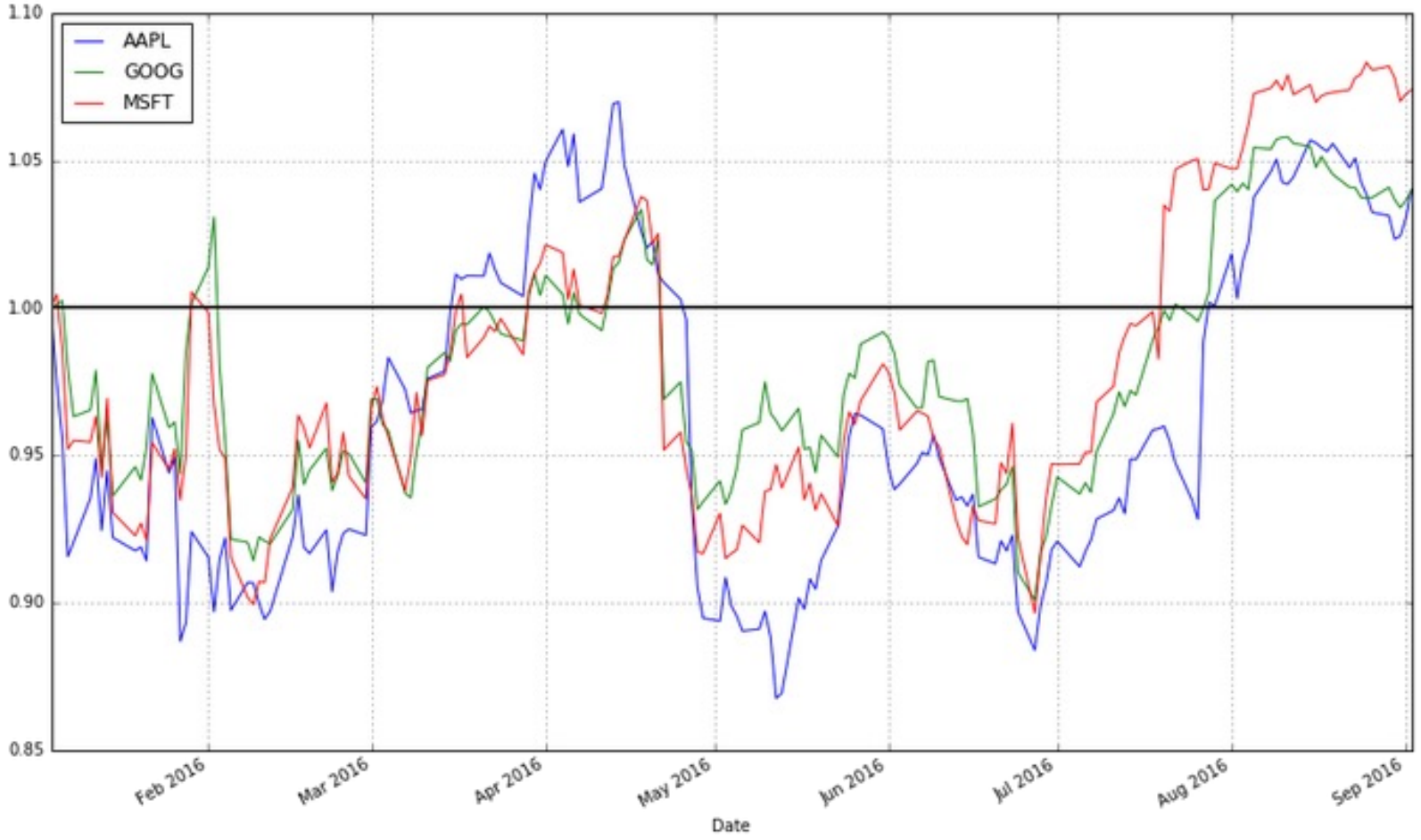
Descriptive analytics

- Extract frequent patterns
  - There are lots of web documents where the following three words co-occur frequently:

    "Stockholm", *"Housing"*, and *"^#@$&^#$@"*
- Extract association rules
  - If a patient is diagnosed with Heart Failure, there is a 65% chance that the patient is prescribed with (RAS) inhibitors + beta blockers
- Find groups of entities that are similar (clustering)
  - groups of Facebook users that *have similar friends/interests*
  - groups drugs that *have similar side-effects*
  - groups of patients with *similar treatment pathways*

Stockholms universitet

Finding groupings of stocks



Stockholms universitet

INPUT

OUTPUT

Stockholms universitet

Input: example

Features (attributes)

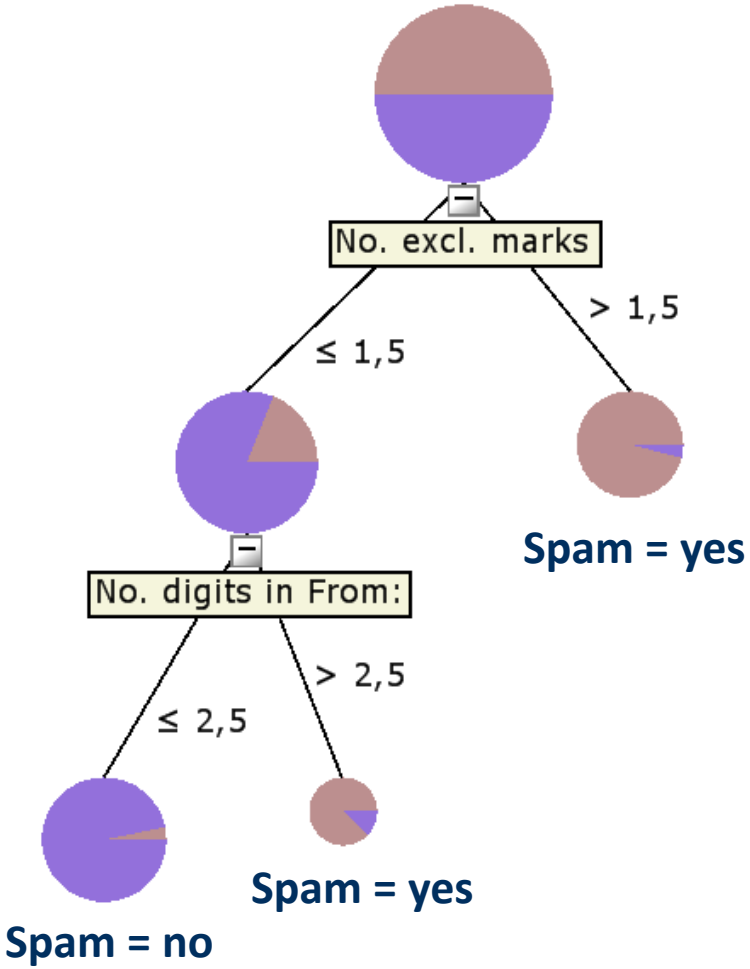
Class label

Examples (observations)

Email	All caps	No. excl. marks	Missing date	No. digits in From:	Image fraction	Spam
e1	yes	0	no	3	0	yes
e2	yes	3	no	0	0.2	yes
e3	no	0	no	0	1	no
e4	no	4	yes	4	0.5	yes
e5	yes	0	yes	2	0	no
e6	no	0	no	0	0	no

Stockholms universitet

Output: example



Task:

- Using the input examples (emails in training set)
- Build a model for predicting the class label (spam)

Stockholms universitet

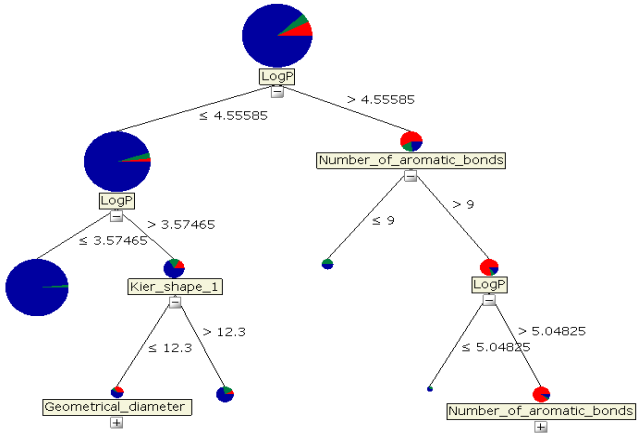
Output: more examples

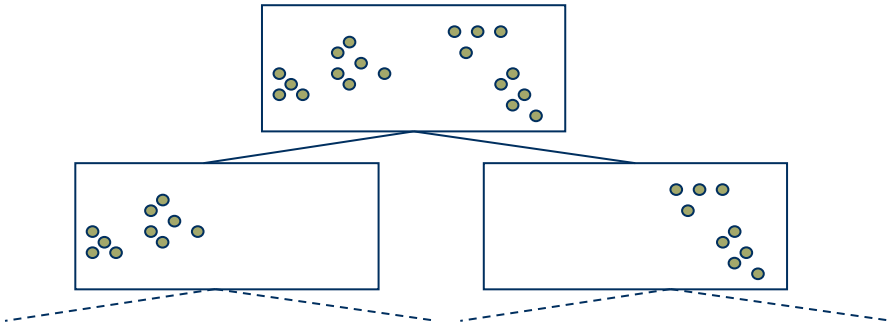
- Interpretable representation of findings
  - equations, rules, decision trees, clusters

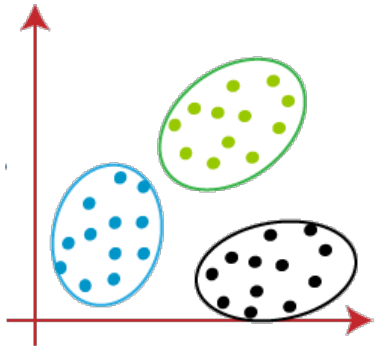
$$y = 0.25 + 4.5x_1 - 2.2x_2 + 3.1x_3$$

$$\textbf{if } x_1 > 3.0 \;\&\; x_2 \leq 1.8 \textbf{ then } y = 1.0$$

BuysMilk & BuysCereals → BuysJuice
[Support : 0.05, Confidence : 0.85]

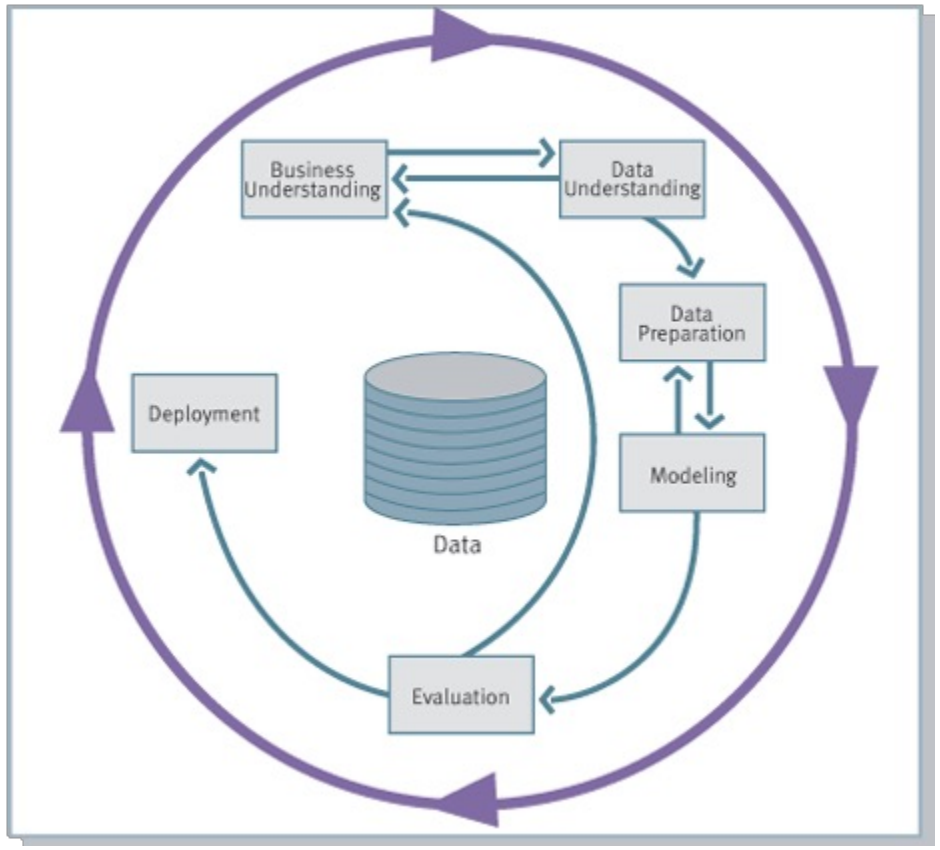






Stockholms universitet

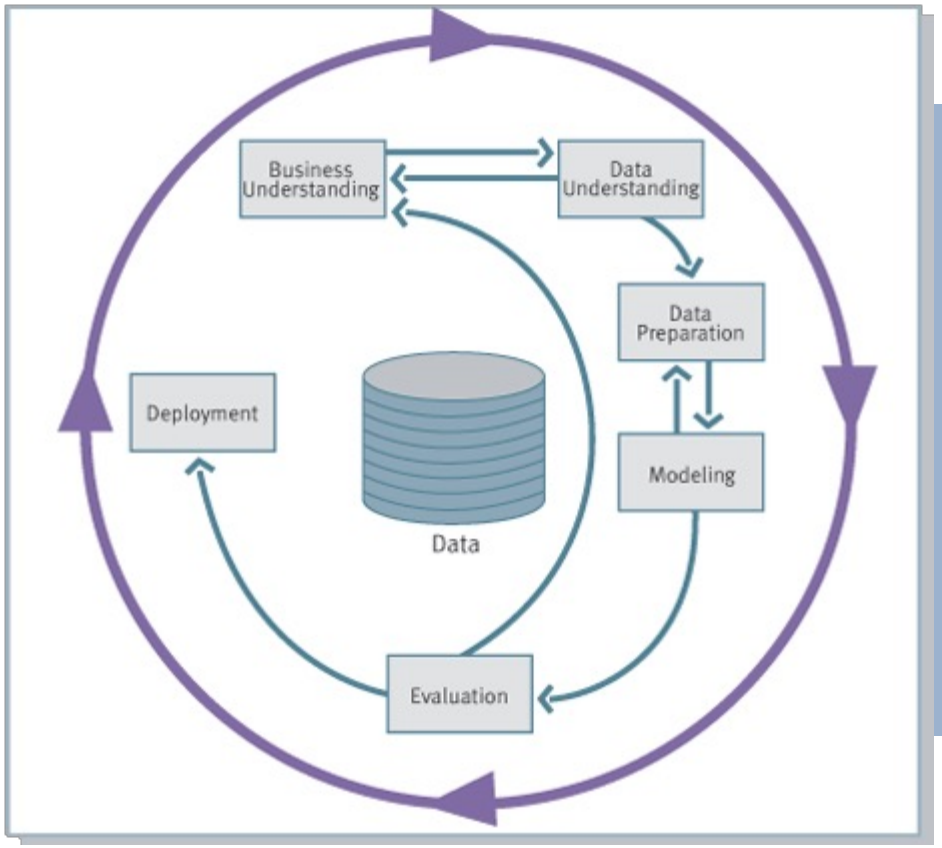
CRISP-DM:
CRoss Industry Standard Process for Data Mining



Shearer C., “The CRISP-DM model: the new blueprint for data mining”, Journal of Data Warehousing 5 (2000)

Stockholms universitet

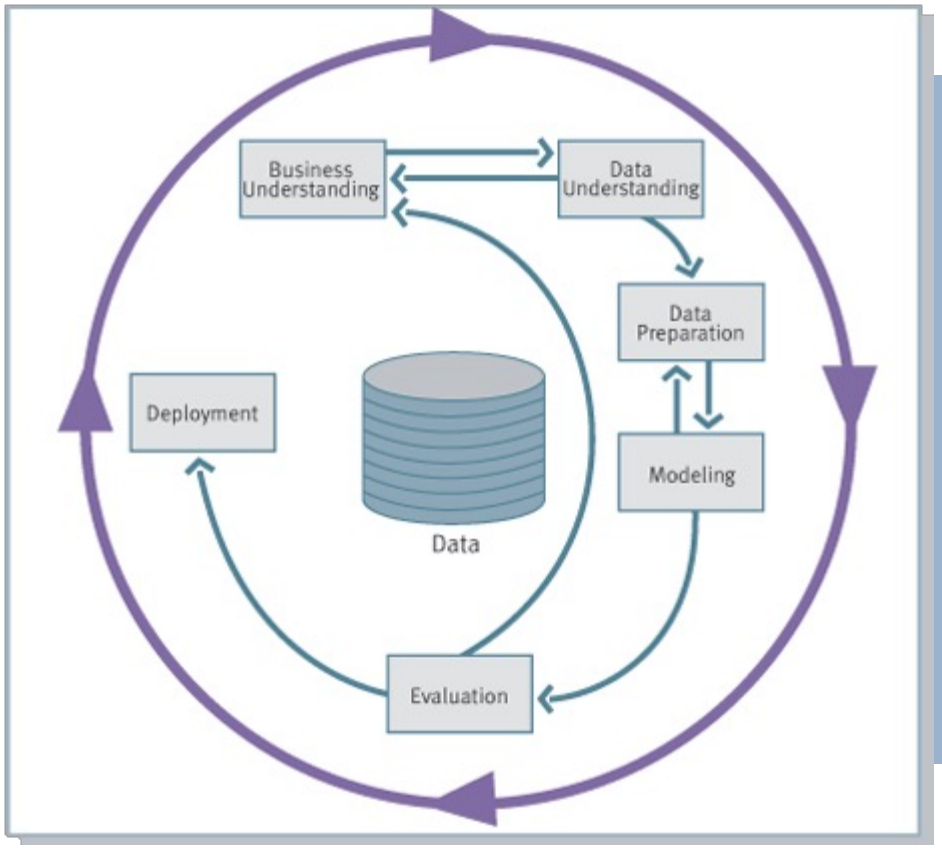
CRISP-DM



- **Business Understanding**
  - understand the project objectives and requirements from a business perspective
  - convert this knowledge into a data mining problem definition
  - create a preliminary plan to achieve the objectives

Stockholms universitet

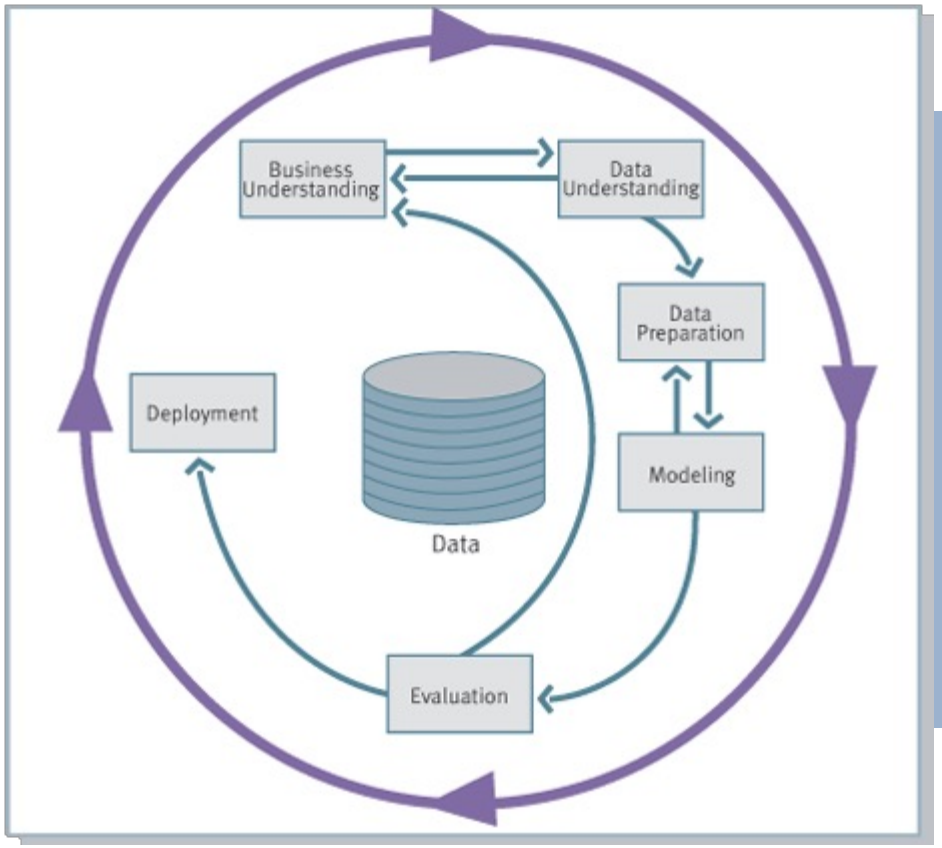
CRISP-DM



- **Data Understanding**
  - initial data collection
  - get familiar with the data
  - identify data quality problems
  - discover first insights
  - detect interesting subsets
  - form hypotheses for hidden information

Stockholms universitet

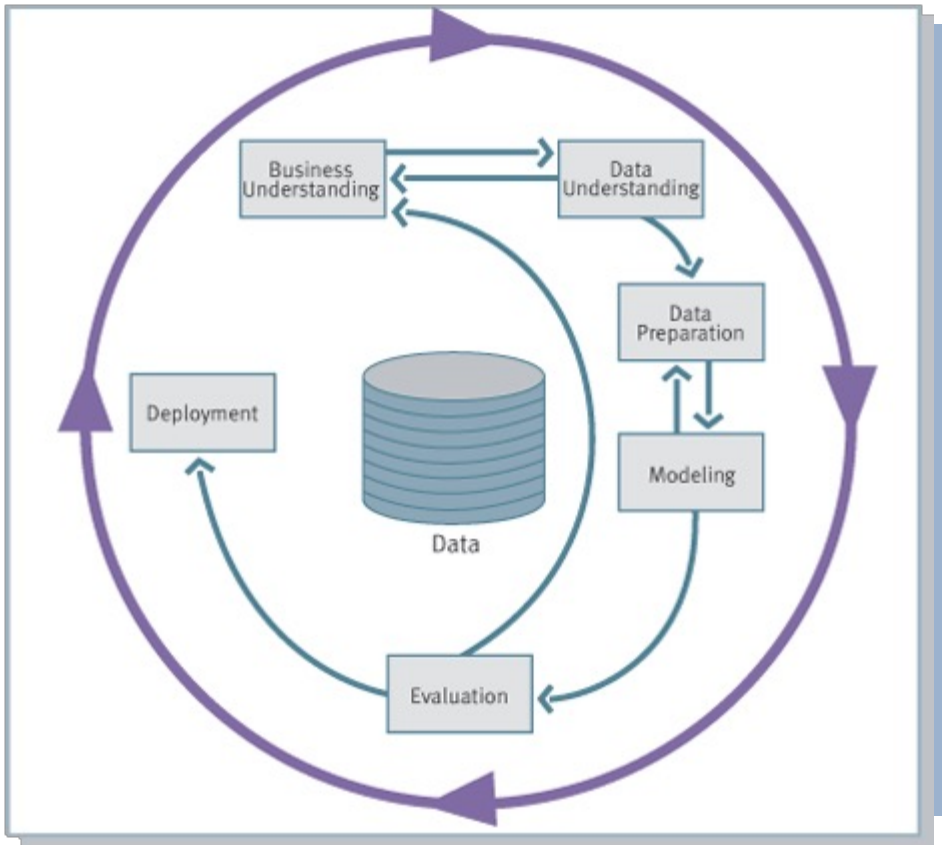
CRISP-DM



- **Data Preparation**
  - construct the final dataset to be fed into the machine learning algorithm
  - tasks here include: table, record, and attribute selection, data transformation and cleaning

Stockholms universitet

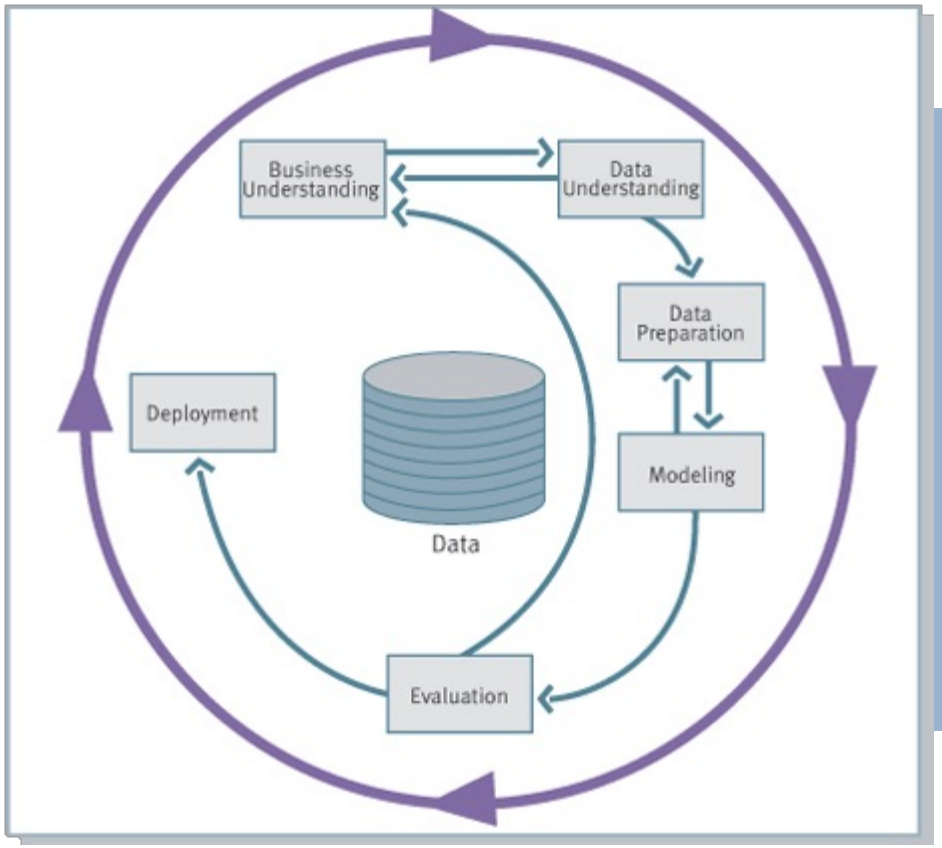
CRISP-DM



- **Modeling**
  - various data mining techniques are selected and applied
  - parameters are learned
  - some methods may have specific requirements on the form of input data
  - going back to the data preparation phase may be needed

Stockholms universitet

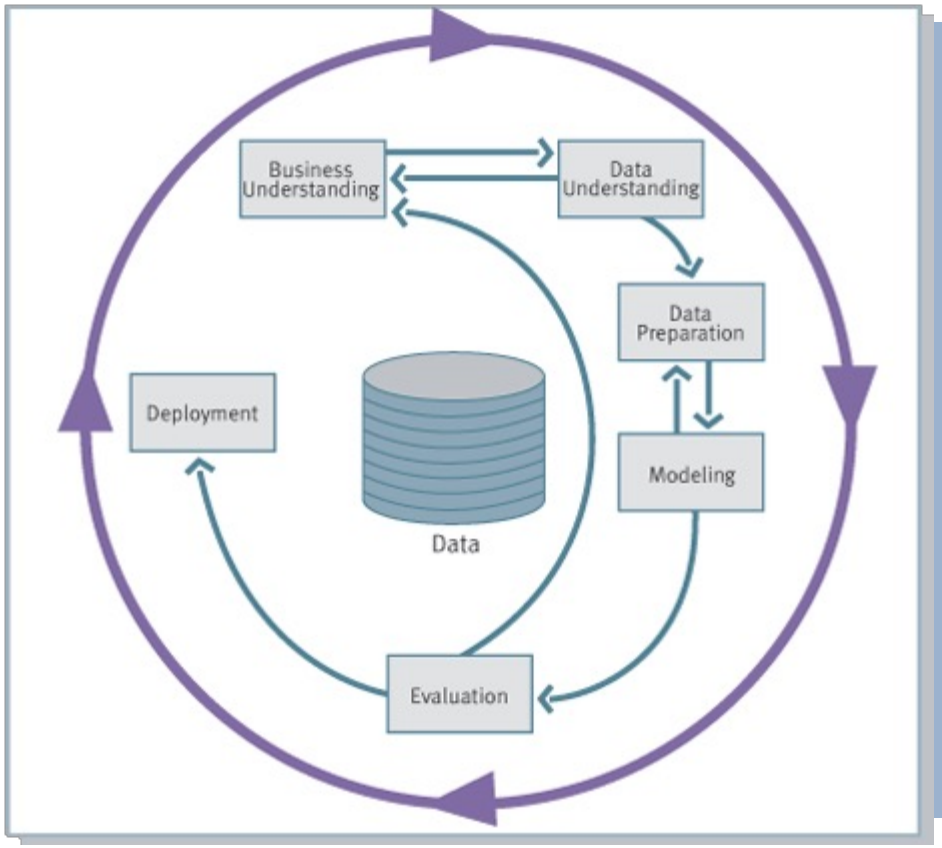
CRISP-DM



- **Evaluation**
  - current model should have high quality from a data mining perspective
  - before final deployment, it is important to test whether the model achieves all the business objectives

Stockholms universitet

CRISP-DM



- **Deployment**
  - just creating the model is not enough
  - the new knowledge should be organized and presented in a usable way
  - generate a report
  - implement a repeatable data mining process for the user or the analyst

Stockholms universitet


Basic and Fundamental Problems

Finding the majority element

- Given a set of labeled elements, e.g.,

  {C, B, C, C, A, C, C, A, B, C, …}

- Identify the **majority element**: element that occurs **more than 50%** of the time (assuming there exists one)

- How can you find it?

- … using no more than a ***few memory locations***?

Stockholms universitet

Finding the majority element
(solution: Boyer-Moore's Algorithm)

```
A = first item you see; count = 1
for each subsequent item B
        if (A==B) count = count + 1
        else {
                count = count - 1
                 if (count == 0) {
                     A=B;
                     count = 1
                     }
                }
endfor
```

Every time you see the same element increase the counter

Every time you see a different element decrease the counter

If the counter becomes 0, replace the element and set the counter back to 1

Stockholms universitet

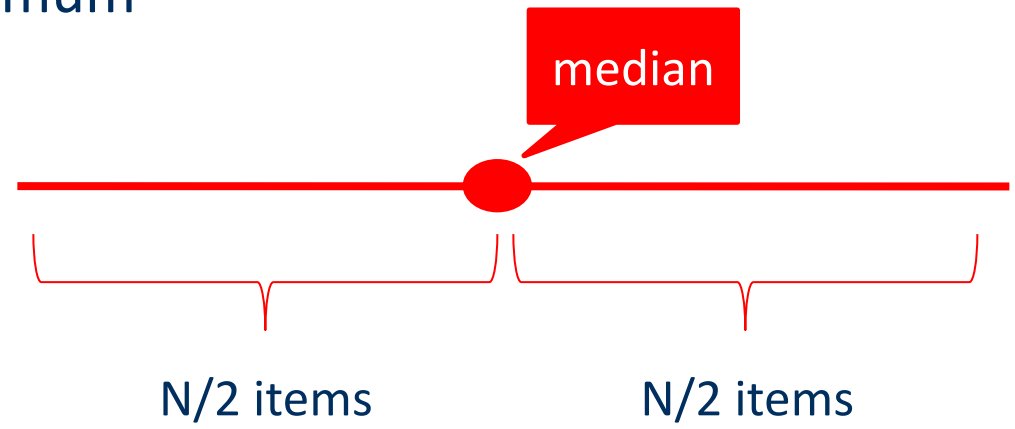
Finding a number in the top half

- Given a set of **N** numbers (**N** is very large)
- Find a number **x** such that **x** is *likely* to be **larger** than the median of the numbers
- Simple solution?
  - **Sort** the numbers and store them in sorted array A
  - Any value **larger** than A[N/2] is a solution
- Other solutions?

Stockholms universitet

Finding a number in the top half *efficiently*

- A solution that uses **small number of operations**
  - ***Randomly sample*** **K** numbers from the file
  - Output their maximum



- Failure probability $p = (1/2)^K$
- If $K = 10$, then $p = 0.0009765625$

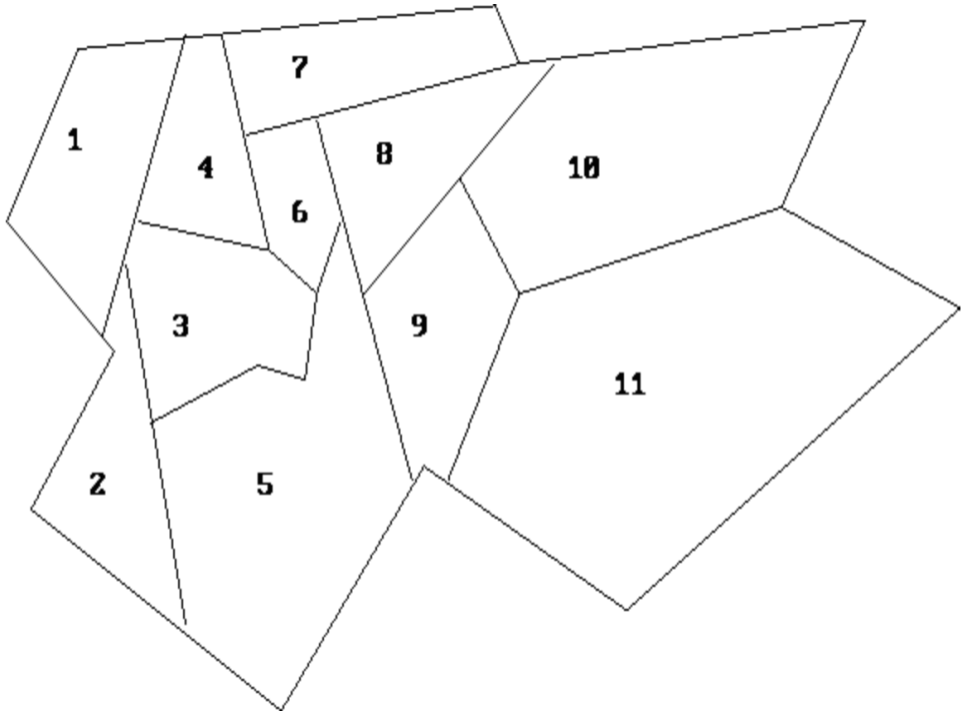
Stockholms universitet

The Set Cover Problem

- A trickier data mining task…
- A common algorithmic problem…
- One of the **MOST USEFUL** problems in Computer Science!

The Set Cover Problem: Example I

- The mayor of a city wants to place fire stations to cover each neighborhood
- Each fire station covers:
  - own neighborhood
  - all adjacent ones



Challenge:

- Where shall we place the fire stations to minimize the city's expenses?
- Each fire station costs X SEK per month

Stockholms universitet

The Set Cover Problem: Example I

- A hospital ER needs to keep doctors on call
- A qualified individual is available to perform every medical procedure that might be required (there is an official list of such procedures)
- For each procedure:
  - Several doctors can be available on-call duty
  - Additional salary needs to be paid

	Doc 1	Doc 2	Doc 3	Doc 4	Doc 5	Doc 6
Procedure 1	✓			✓		
Procedure 2	✓				✓	
Procedure 3		✓	✓			
Procedure 4	✓					✓
Procedure 5		✓	✓			✓
Procedure 6		✓				

- Goal: Choose doctors so that **each procedure is covered** at a minimum cost!

Stockholms universitet

The Set Cover Problem: Example II

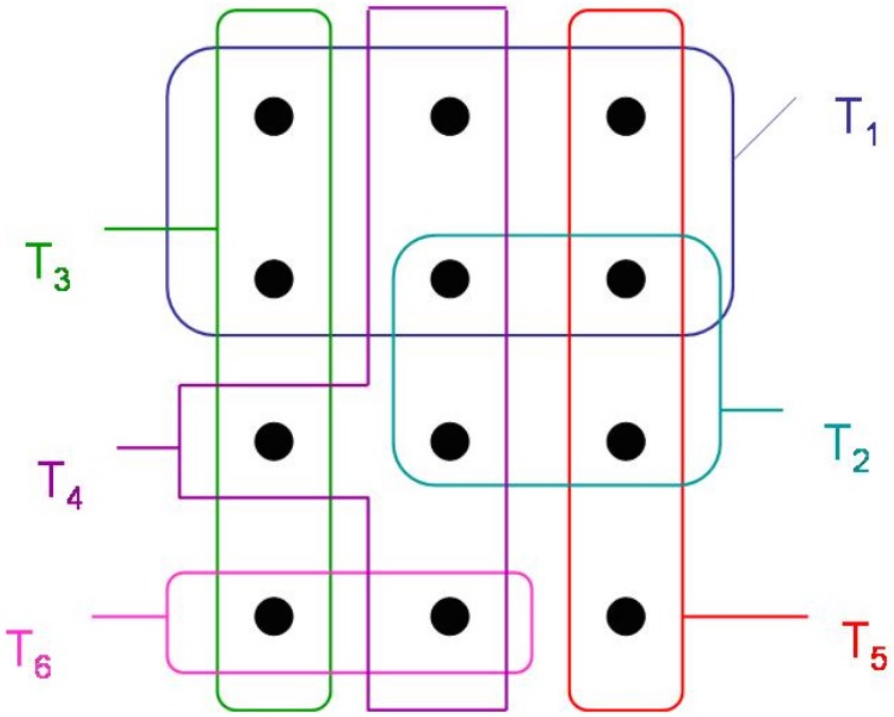
- IBM wants to identify computer viruses
- **Elements: 5000** known viruses (their machine code)
- **Sets: 9000** substrings of **20** or more consecutive bytes from viruses, not found in ‘good’ code
- **A set cover of 180 was found!**

It suffices to search for these 180 substrings to verify the existence of known computer viruses

Stockholms universitet

The Set Cover Problem

- A set of objects
- Some sets **T** that cover the objects



- Find the set of **T**s that cover all objects!
- Find the smallest set!

Stockholms universitet

Formal Definition

- **Setting:**
  - Universe of **m** elements **U = {U_1,…,U_m}**
  - A set of **n** sets **T = {T_1,…,T_n}**
  - Find a collection **C** of sets in **T** (**C** subset of **T**) such that **C** contains all elements from **U**

Formal Definition

- **Set-cover problem:** Find the smallest collection **C** of sets from **T** such that all elements in the *universe* **U** are covered

- **Solution?**
  - Try all sub-collections of **T**
  - Select the smallest one that covers all the elements in **U**
  - The running time of the trivial algorithm is $O(2^n|U|)$
  - This is way **too slow**

Stockholms universitet

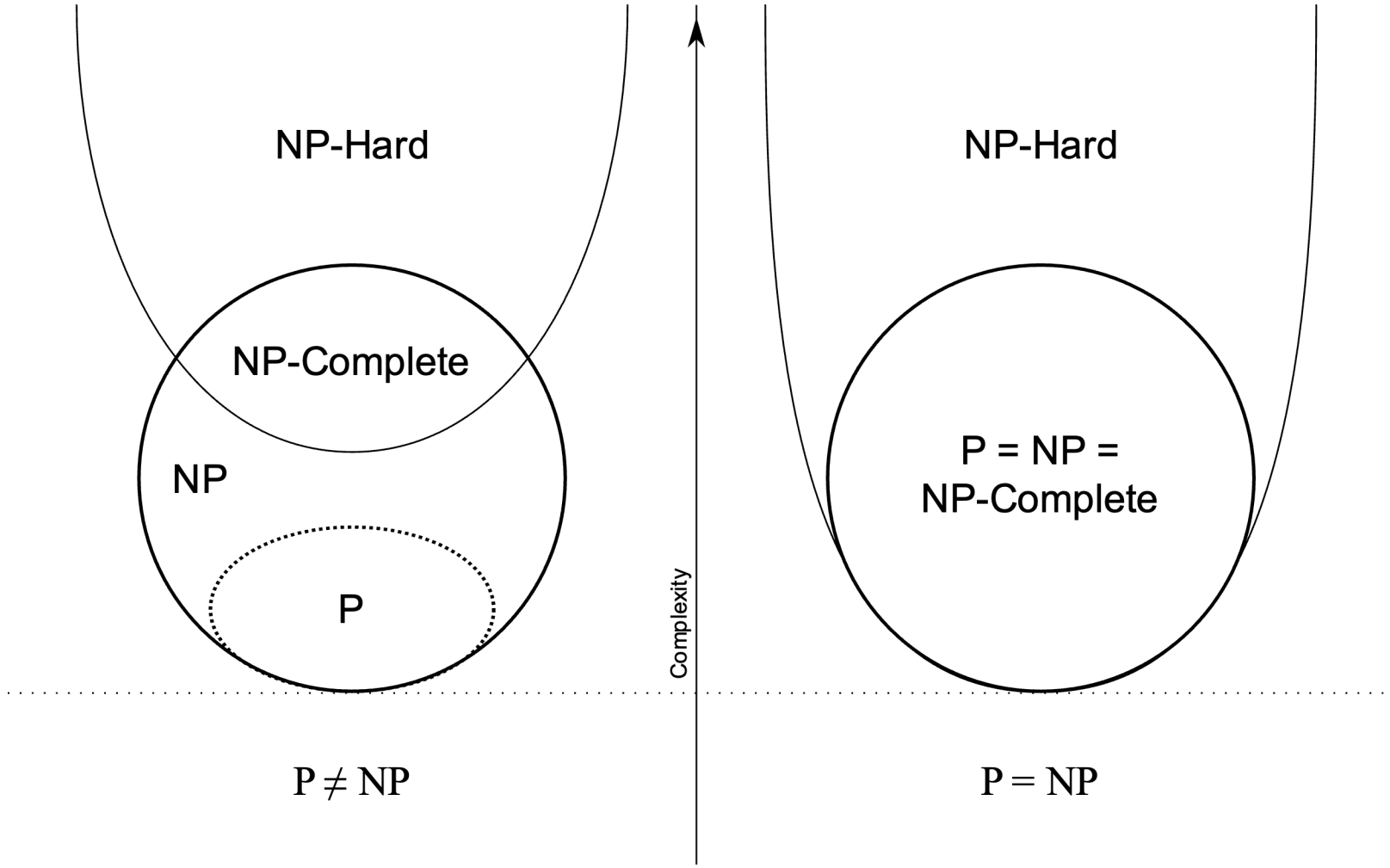
Formal Definition

- **Set-cover problem:** Find the smallest collection **C** of sets from **T** such that all elements in the *universe* **U** are covered

- The set cover problem is *NP-hard*

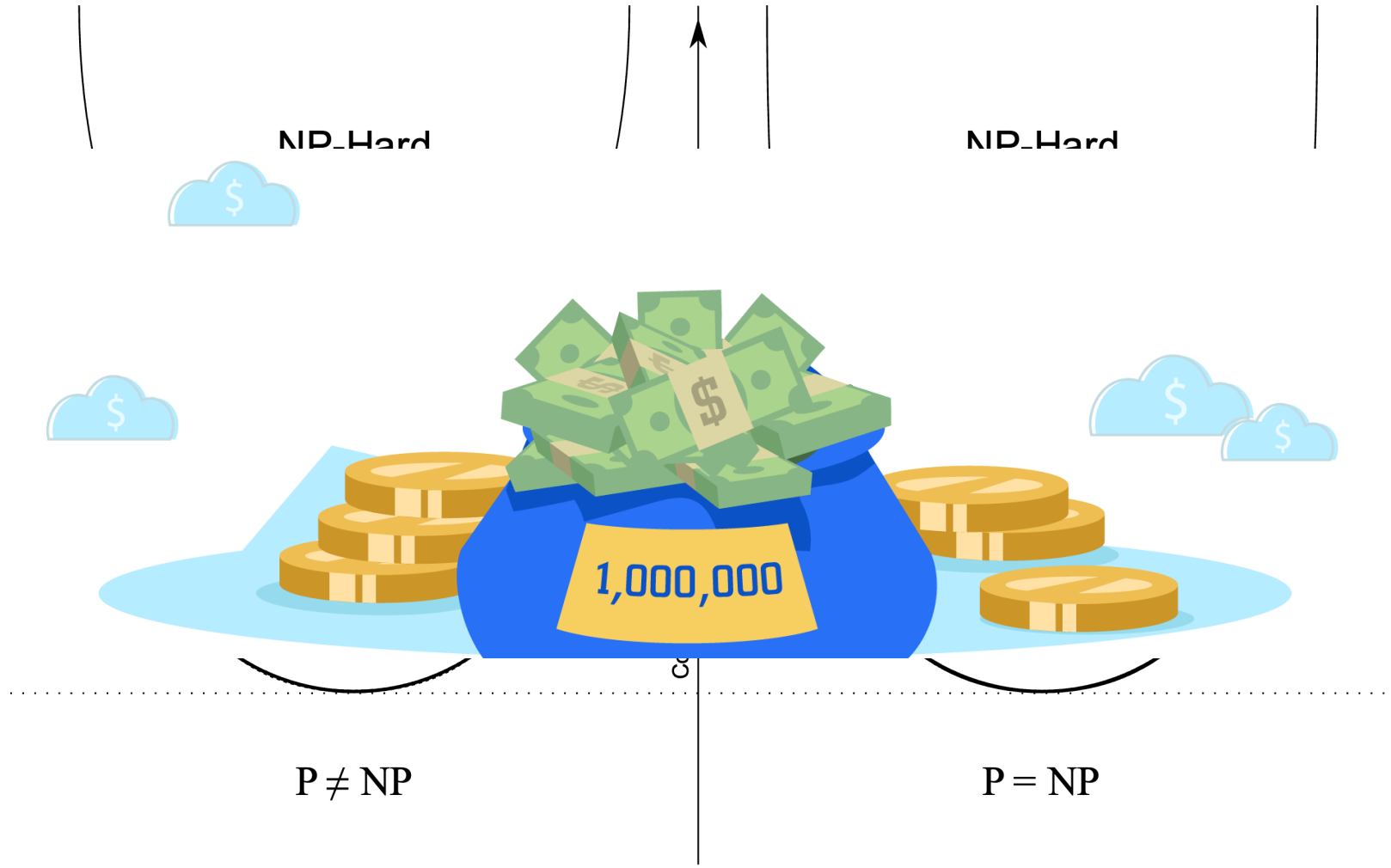
- Simple ***approximation algorithms*** with provable properties are available and very useful in practice

NP-hardness



Stockholms universitet

NP-hardness



Stockholms universitet

Greedy algorithm for set cover

- Select first the largest-cardinality set **t** from **T**
- Remove the elements of **t** from **U**
- Re-compute the sizes of the remaining sets in **T**
- Go back to the first step

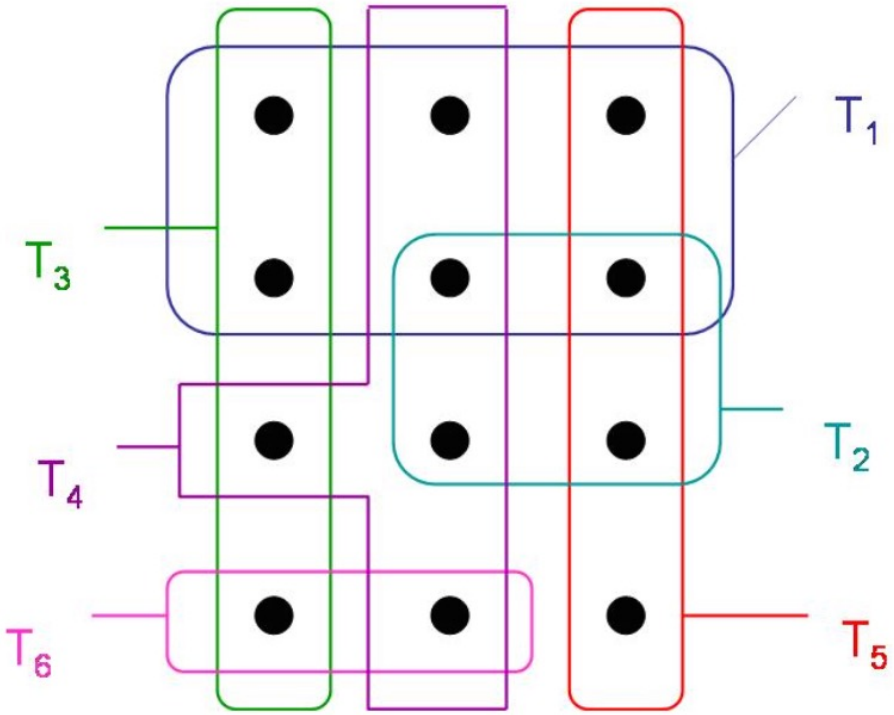
The Greedy algorithm

- **X** = {}
- **While U** is not empty **do**
  - For all **tϵT** let **a_t=|t intersection |**
  - Let **t** be such that **a_t** is *maximal*
  - **X = X U {t}**
  - **U = U\ t**

Stockholms universitet

Recall…

- We want to find a set of Ts such that will cover all the objects

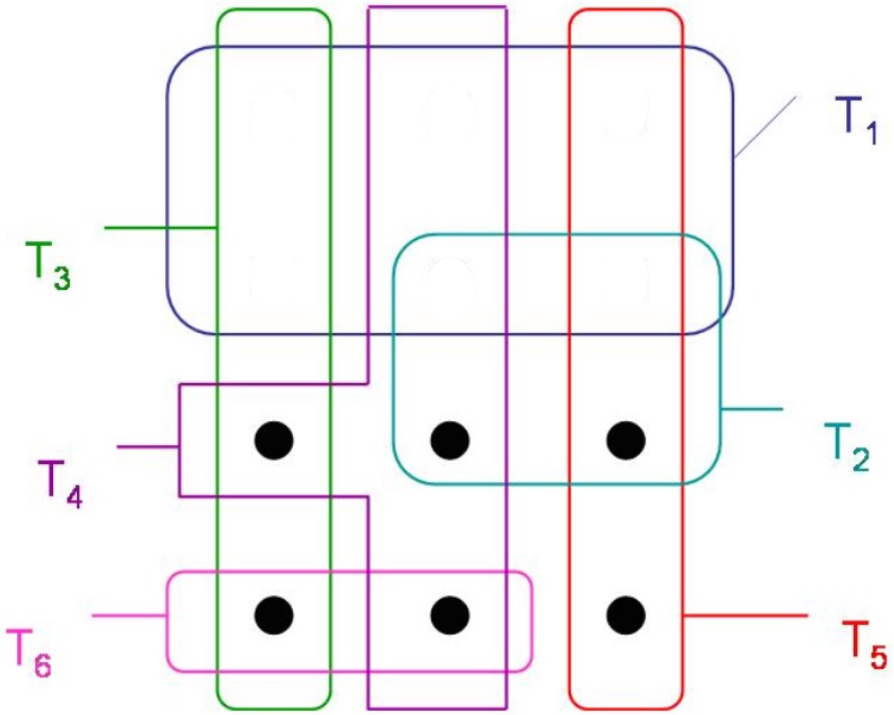


- What would the greedy algorithm find?

Stockholms universitet

Example

- Select biggest set: T_1
- Remove all elements covered by T_1



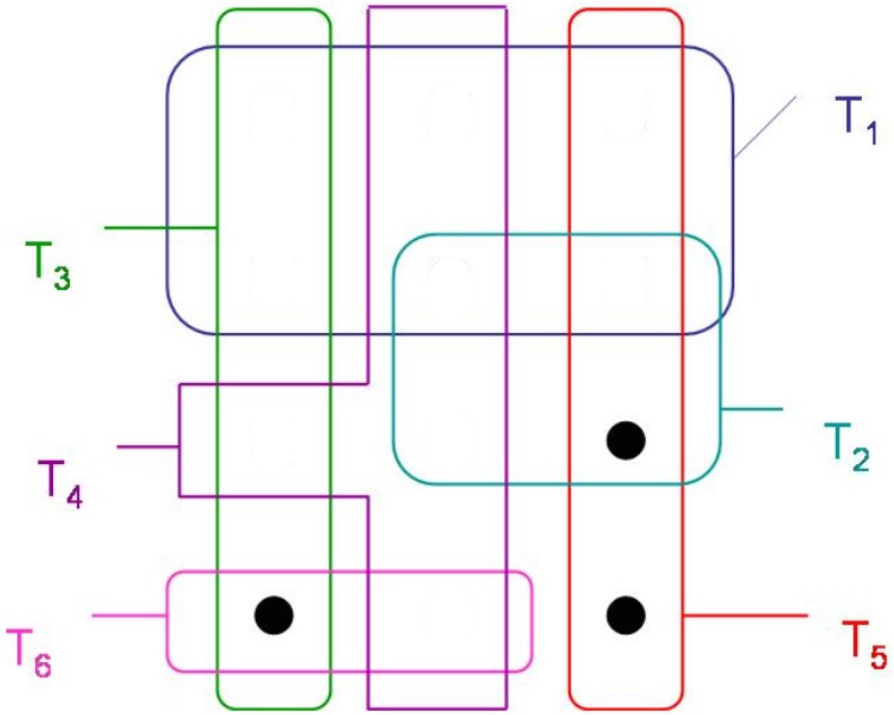
Current solution:

$X = \{T_1\}$

Stockholms universitet

Example

- Select the next biggest set: T_4
- Remove all elements covered by T_4



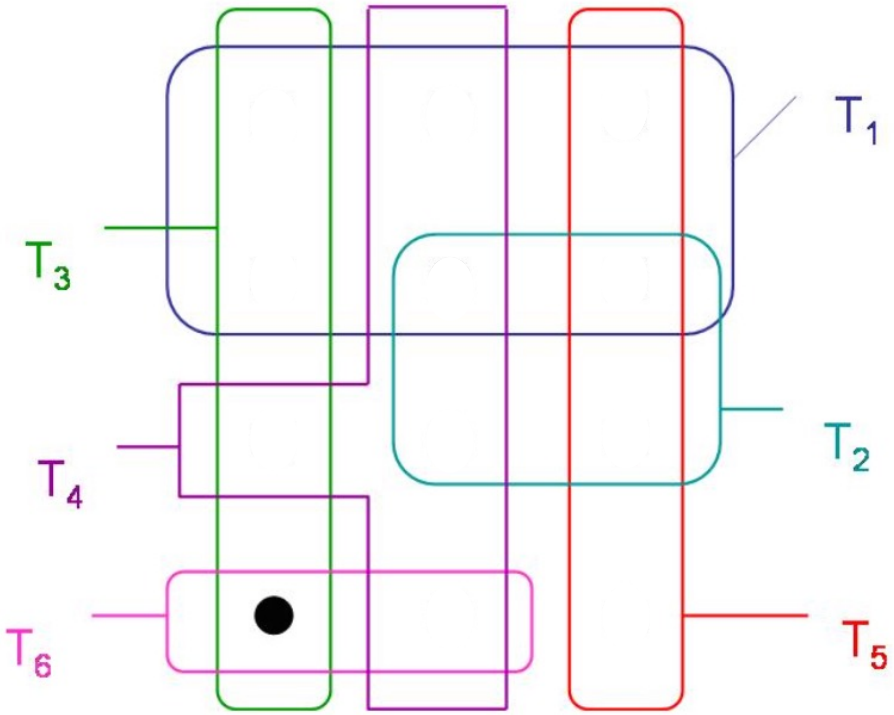
Current solution:

$X = \{T_1\}$

Stockholms universitet

Example

- Select the next biggest set: T_5
- Remove all elements covered by T_5



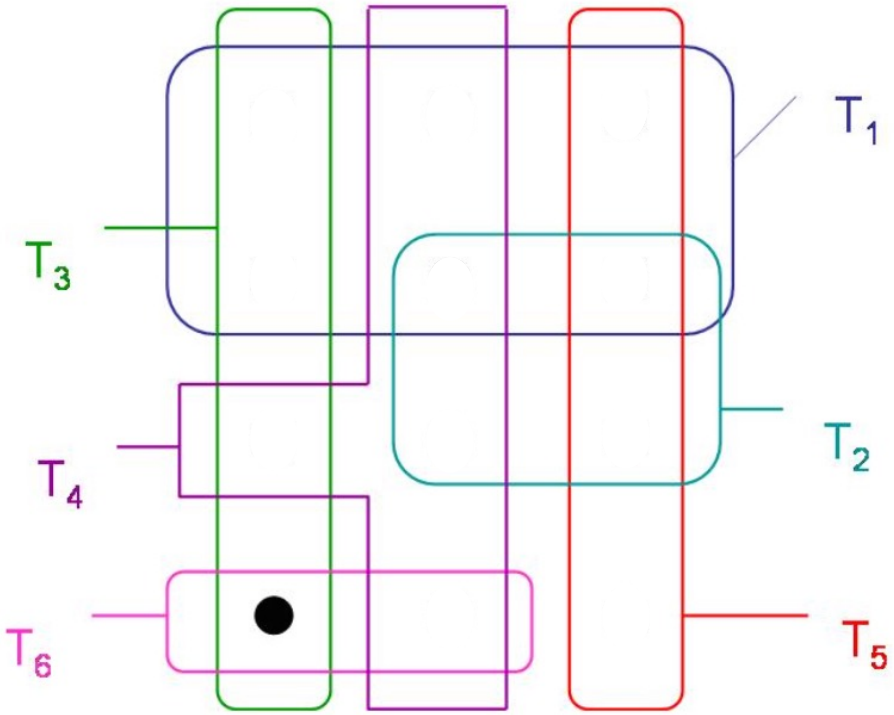
Current solution:

$X = \{T_1, T_4\}$

Stockholms universitet

Example

- Select the next biggest set: T_5
- Remove all elements covered by T_5



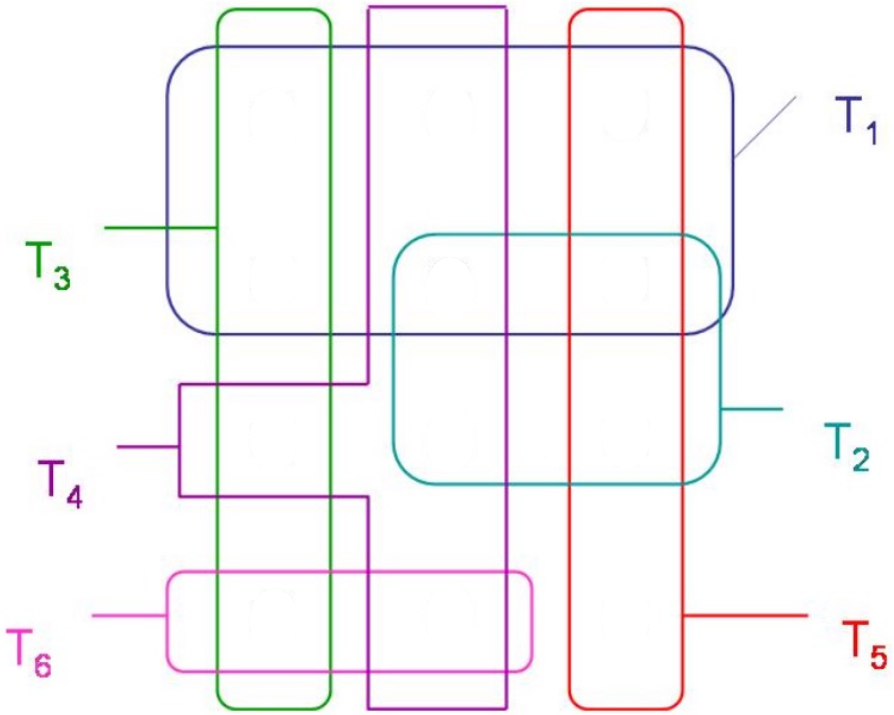
Current solution:

$X = \{T_1, T_4, T_5\}$

Stockholms universitet

Example

- Select the next biggest set: T_6
- Remove all elements covered by T_6



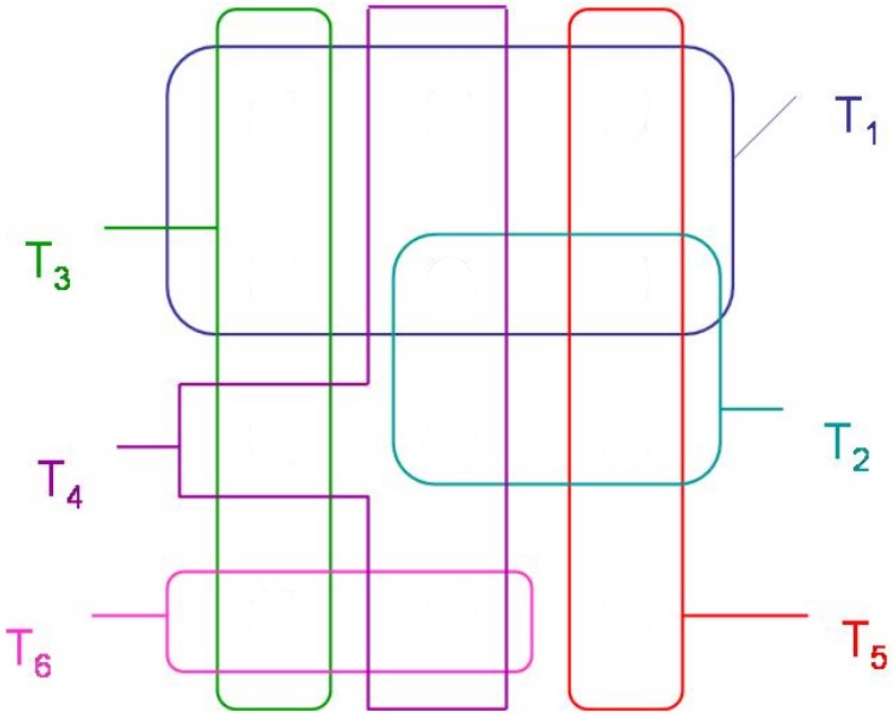
Current solution:

$X = \{T_1, T_4, T_5\}$

Stockholms universitet

Example

- Select the next biggest set: T_6
- Done!



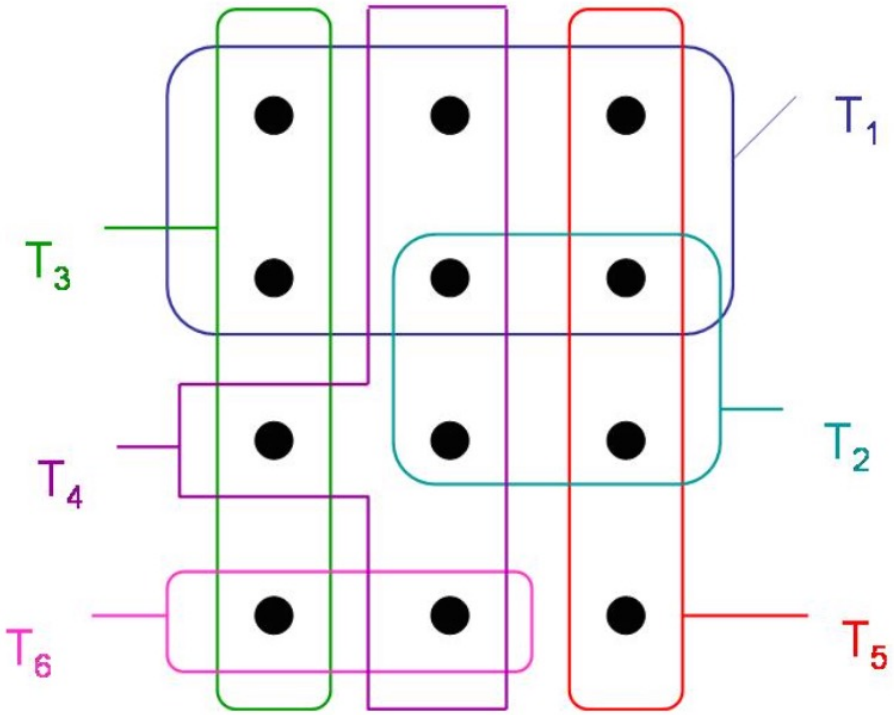
Current solution:

$X = \{T_1, T_4, T_5, T_6\}$

Stockholms universitet

Example

- What is the optimal solution?
- Recall: we want the *smallest* possible set!



An optimal solution:

$X^* = \{T_3, T_4, T_5\}$

Greedy solution:

$X = \{T_1, T_4, T_5, T_6\}$

Stockholms universitet

Today…

Why do we need **Data Analysis**?

What is **Data Mining**?

What are the types of **Machine Learning**?

Examples where Data Mining has been **useful**

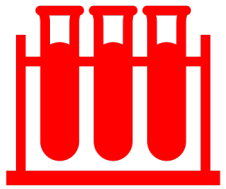
Some (basic) Data Mining prototype **problems**

Stockholms universitet

TODOs

Reading:

Main course book: Chapter 1



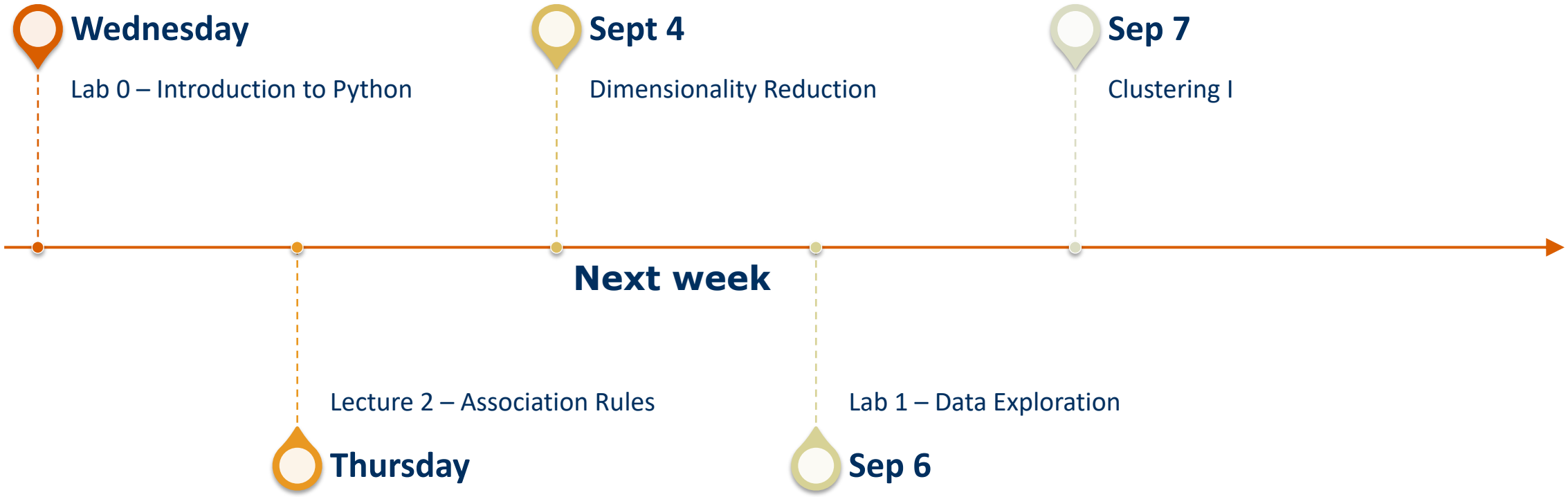
Lab 0

Recommended to complete the lab before the end of the week

Quiz 1

Stockholms universitet

Coming up next



Stockholms universitet