University of Applied Sciences

HOCHSCHULE
EMDEN·LEER

Business Analytics

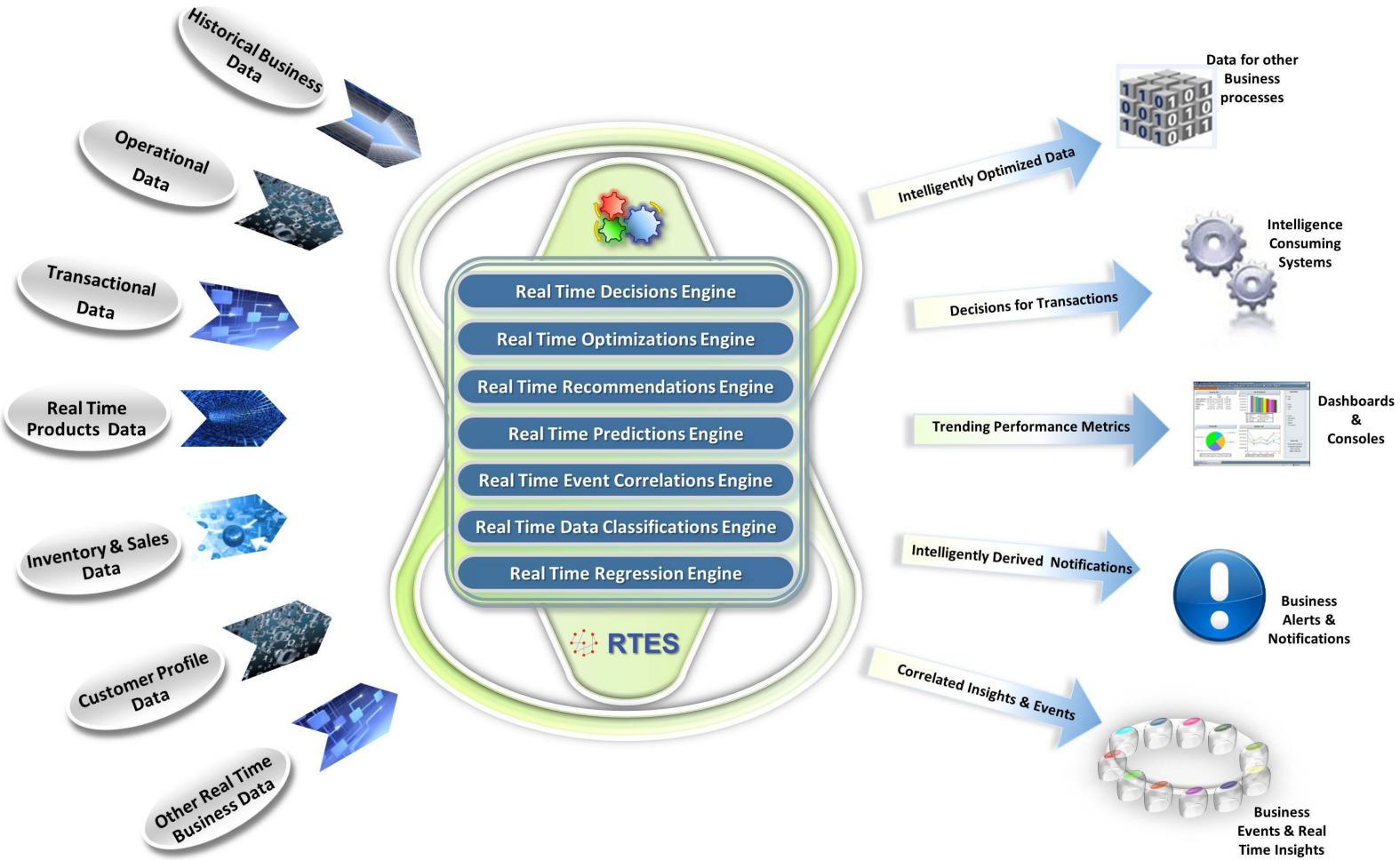
Knowledge Discovery in Databases Process

Prof. Dr. E. Wings

Agenda

Data Mining - Procedure models

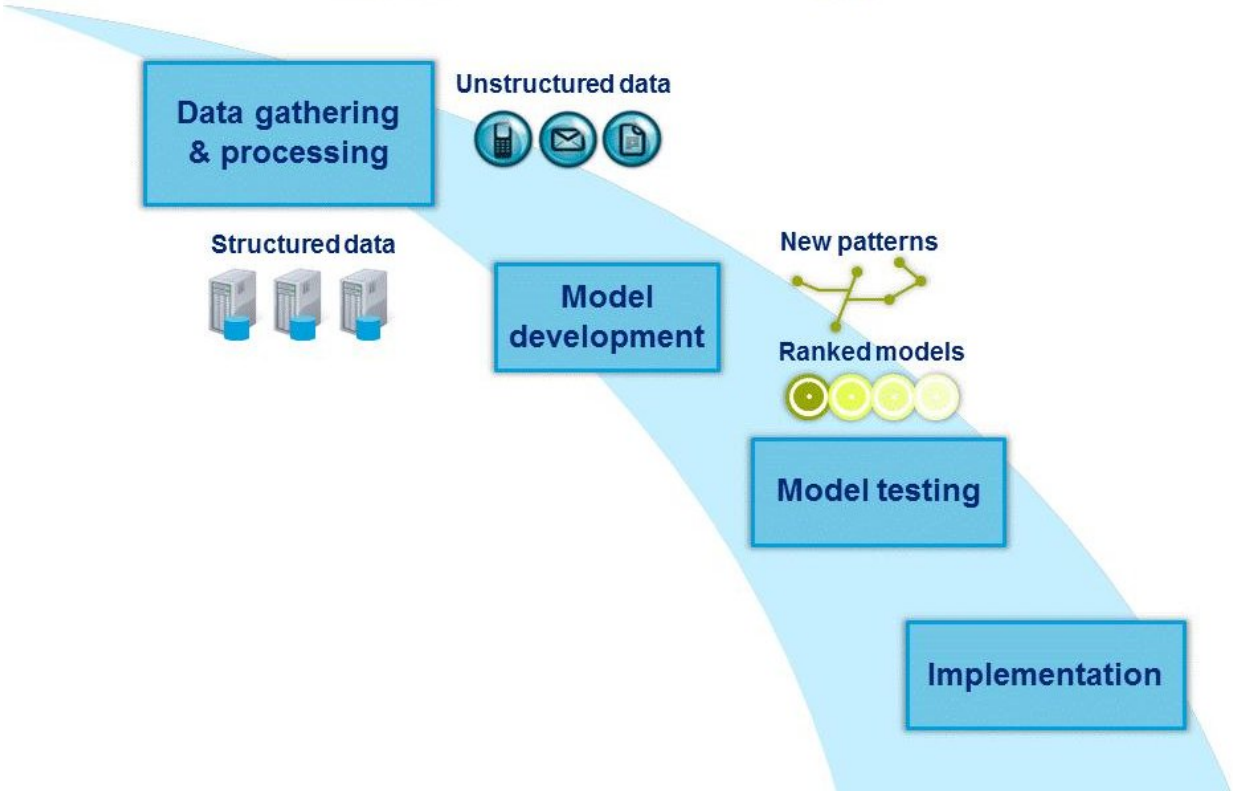
Topics


Historical Business Data
Operational Data
Transactional Data
Real Time Products Data
Inventory & Sales Data
Customer Profile Data
Other Real Time Business Data
Real Time Decisions Engine
Real Time Optimizations Engine
Real Time Recommendations Engine
Real Time Predictions Engine
Real Time Event Correlations Engine
Real Time Data Classifications Engine
Real Time Regression Engine
RTES
Intelligently Optimized Data
Data for other Business processes
Decisions for Transactions
Intelligence Consuming Systems
Trending Performance Metrics
Dashboards & Consoles
Intelligently Derived Notifications
Business Alerts & Notifications
Correlated Insights & Events
Business Events & Real Time Insights
Analyze Data
Identify Potential Patterns
Define Data Relations Hypothesis
Prepare Data for Applicable Engine
Consume Data & Adapt Engine to New Trends
Publish Real Time Business Insights

Typical Data Sources

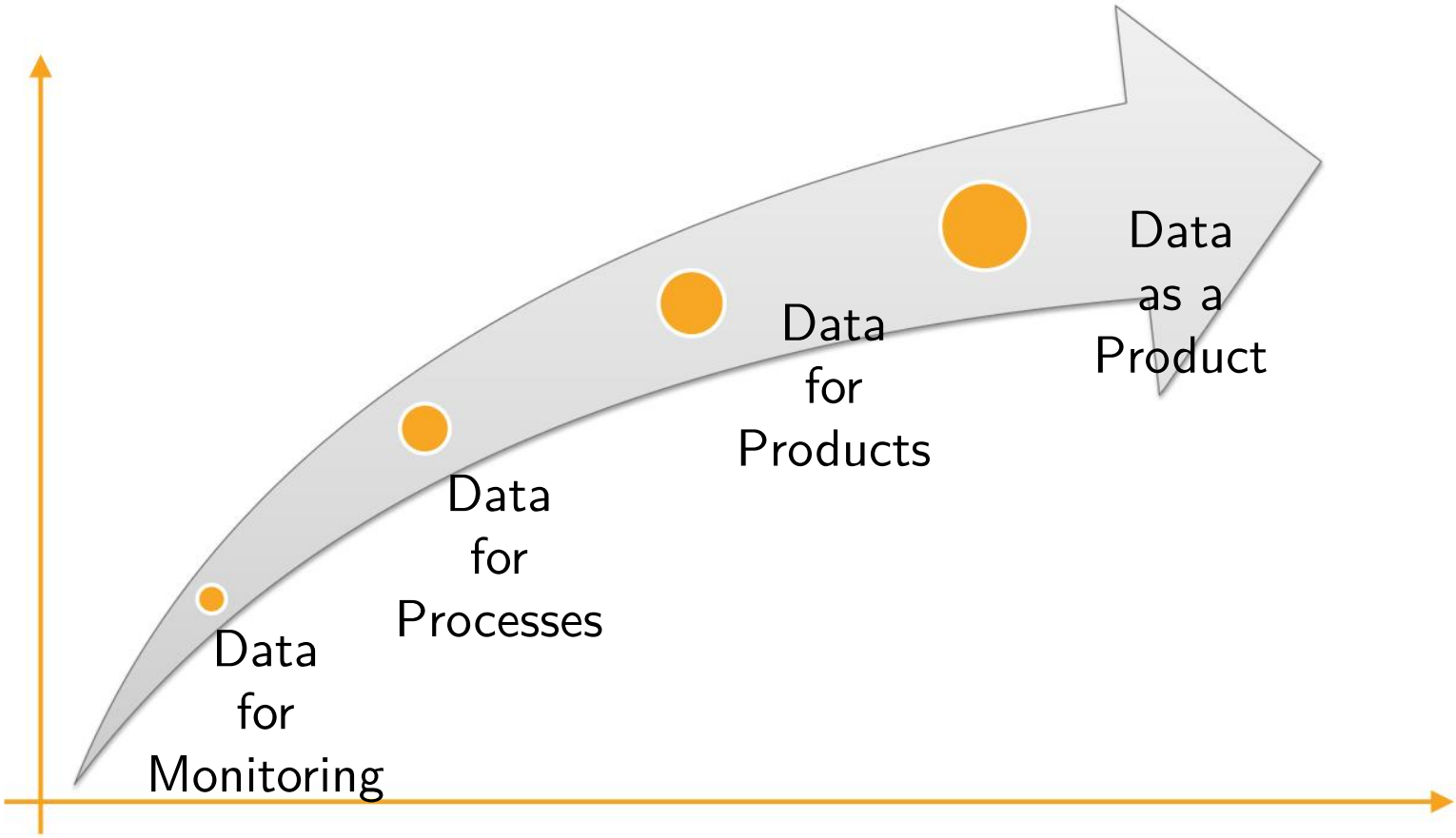
Classical Systems	New Systems		
Structure Data Transactions system (ERP, SCM, CRM)	Semi-structured text (e.g. XML) or unstructured text (*.txt) Sensor data Log data M2M	Mobile IT	Social Web
- customer data - product data - orders - ...	- production data - weather data - geo data - ...	- RFID data - telecommunications - traffic data - ...	- Twitter - Facebook - LinkedIn - youtube - Blogs - Idots

Analytics Process

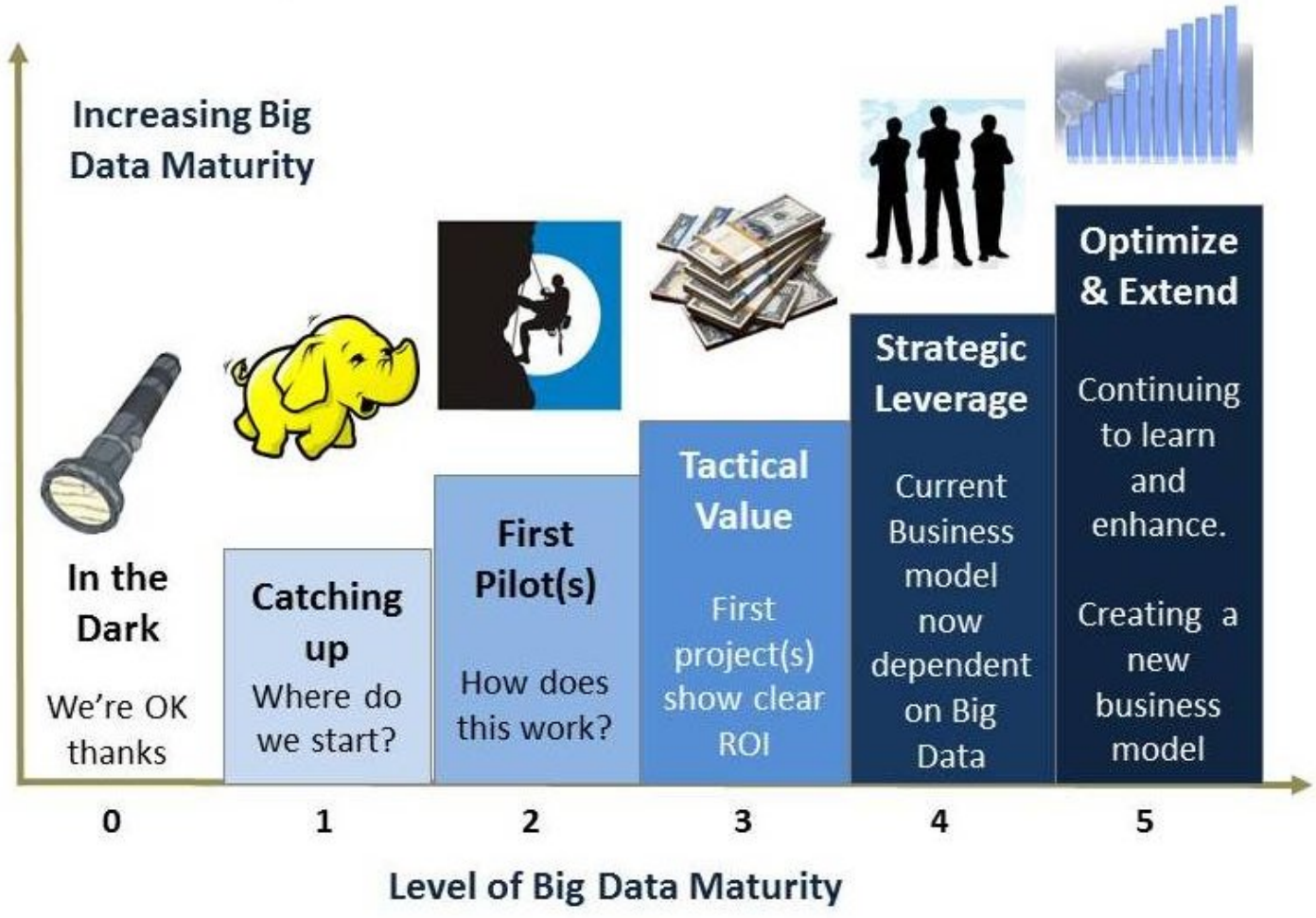


Source: https://sctr7.com

Data in Industry 4.0 Strategy



Big Data Maturity Model



Source: Radcliffe Advisory

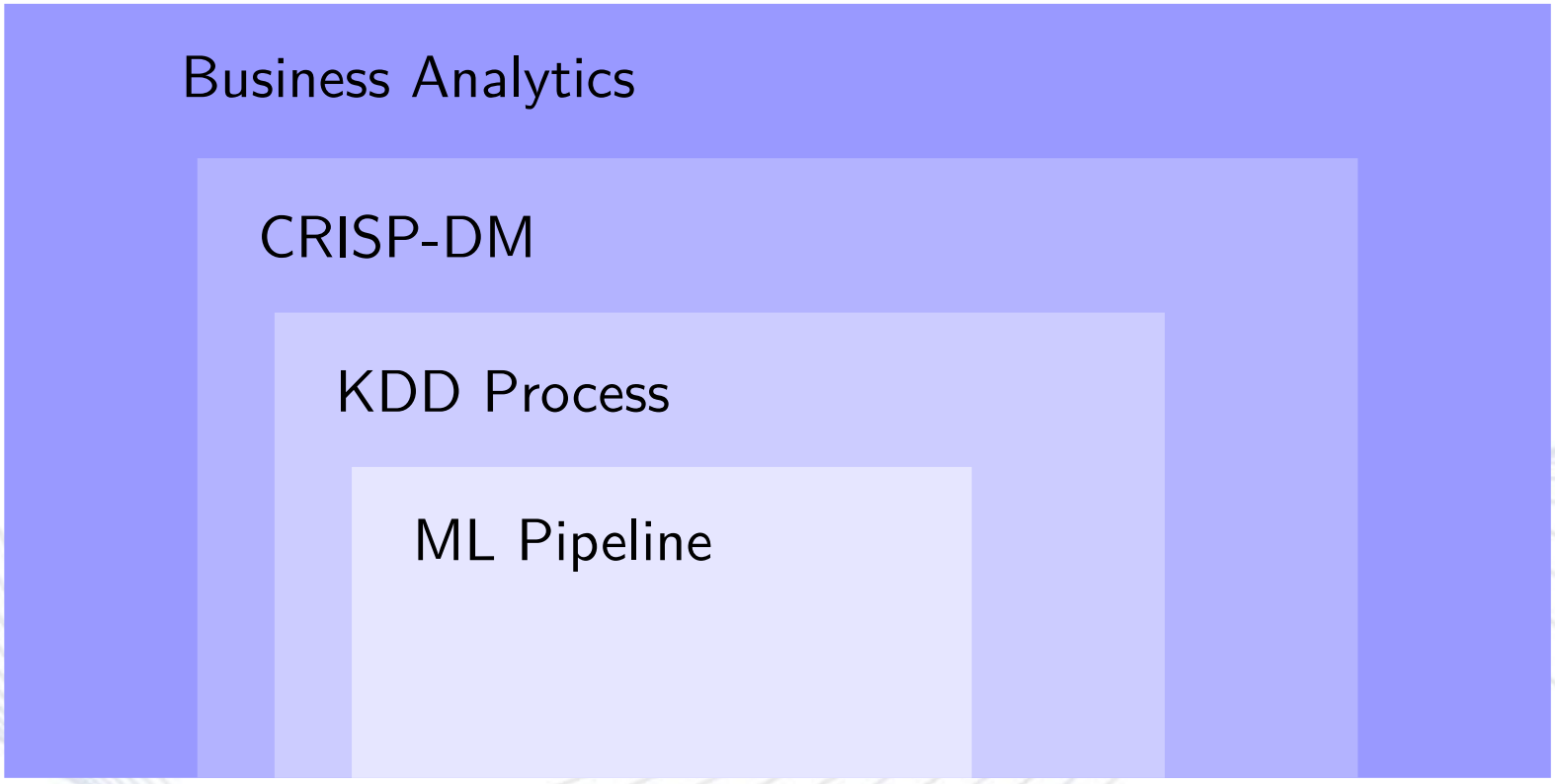
#BigData_MatMod

Data Mining - Procedure models

- Knowledge Discovery in Databases (KDD)
  - Process model for discovery from data
  - Procedure for selecting and preparing data
  - Data mining describes analysis methods and is a sub-process step of KDD model
- CRoss-Industry Standard Process for Data Mining (CRISP-DM)
  - Iterative and agile process model for data mining
  - Standard process with six phases for analyzing data sets according to patterns, trends and relationships

$\Rightarrow$ Process models structure the evaluation of Big Data

Data Mining - Procedure models



Knowledge Discovery in Databases Process

Knowledge Discovery in Databases (KDD) - General

- KDD is an approach to data analysis to discover and make explicit knowledge available in extensive data sets
- Recognition of relationship patterns e.g. regularities or anomalies in large amounts of data
- Derivation of previously unknown correlations that are valid for as large a proportion of the database as possible

Knowledge Discovery in Databases (KDD) - Process

- Focus on the entire knowledge discovery process
- Comprehensive handling of large amounts of data
- Process includes selection, preparation, specification, analysis and interpretation of data Data Mining is a sub-process

$\Rightarrow$ Comprehensive process for discovering knowledge from databases

Modeling Process

- Define Goals
- Gather Data
- Decide Model
- Prepare Data
- Variable Selection and Elimination
- Build Candidate Model
- Finalize Model
- Implementation and Monitoring

Difficulties in Model Building Process

- Lack of clarity in problem definition
- Using data too old or otherwise not relevant
- using too simple/complex model
- Not considering outliers
- Relying only on linear variable selection or compress techniques
- Going deep on single specialized technique
- Not rebuilding final model optimally using all appropriate data
- Errors in implementation process

Difficulties in Model Building Process - Problem

Understanding on the problem means to understand the four different dimensions of data mining contexts:

- The application domain is the specific area in which the data mining project takes place
- The data mining problem type describes the specific class(es) of objective(s) that the data mining project deals with
- The technical aspect covers specific issues in data mining that describe different (technical) challenges that usually occur during data mining
- The tool and technique dimension specifies which data mining tool(s) and/or techniques are applied during the data mining project

Difficulties in Model Building Process - Problem

	Data Mining Context			
Dimension	**Application Domain**	**Data Mining Problem Type**	**Technical Aspect**	**Tools and Technique**
Example	Response Modeling	Description and Summarization	Missing Values	Clemetine
	Churn Prediction	Segmentation	Outliers	MineSet
	Control System	Concept Description	Getting Data	Decision Tree
	Domain Knowledge	Classification Prediction	...	IDE
	...	Dependency Analysis		...

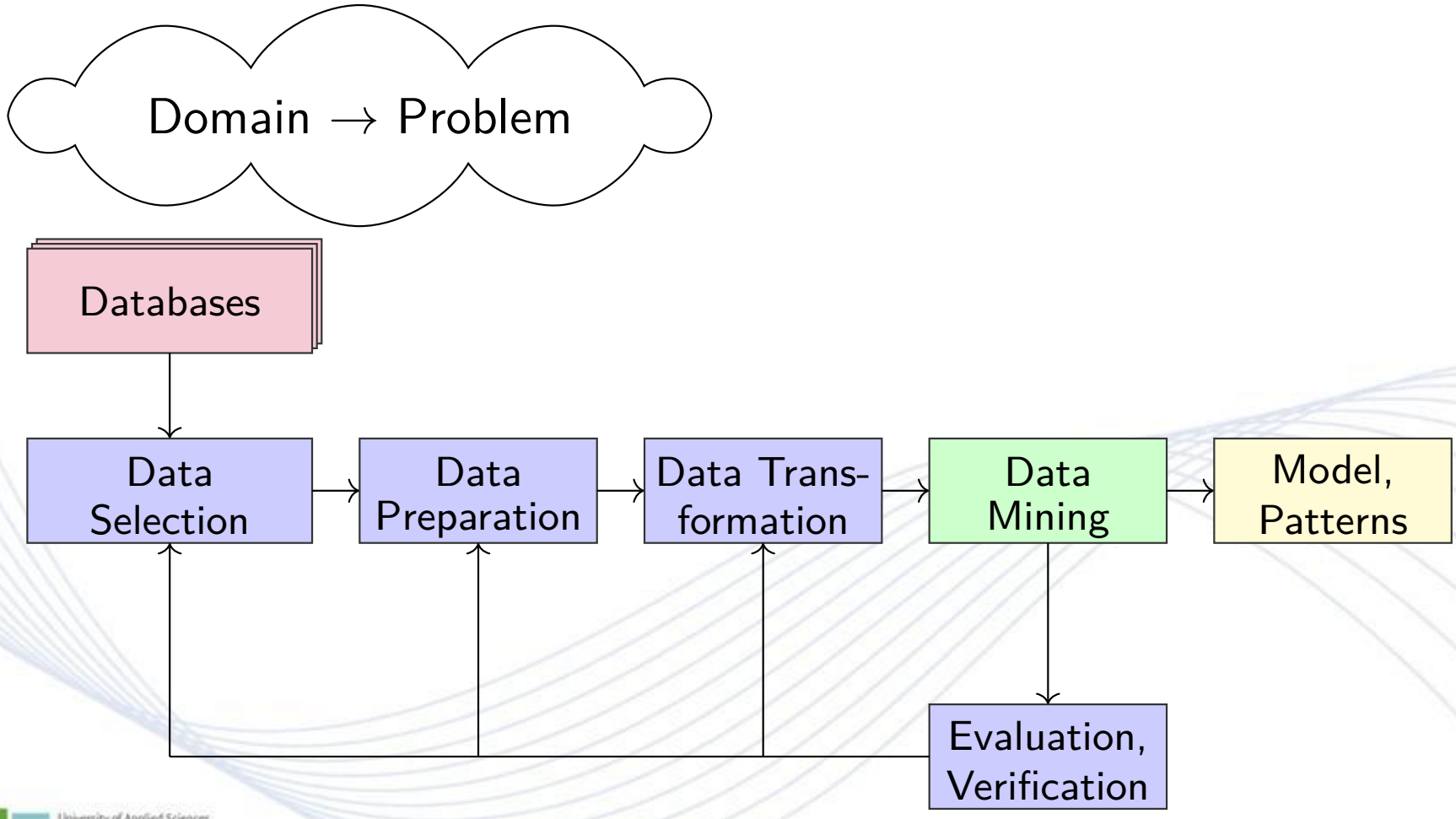
Dimensions of data mining contexts and examples

A specific data mining context is a concrete value for one or more of these dimensions.

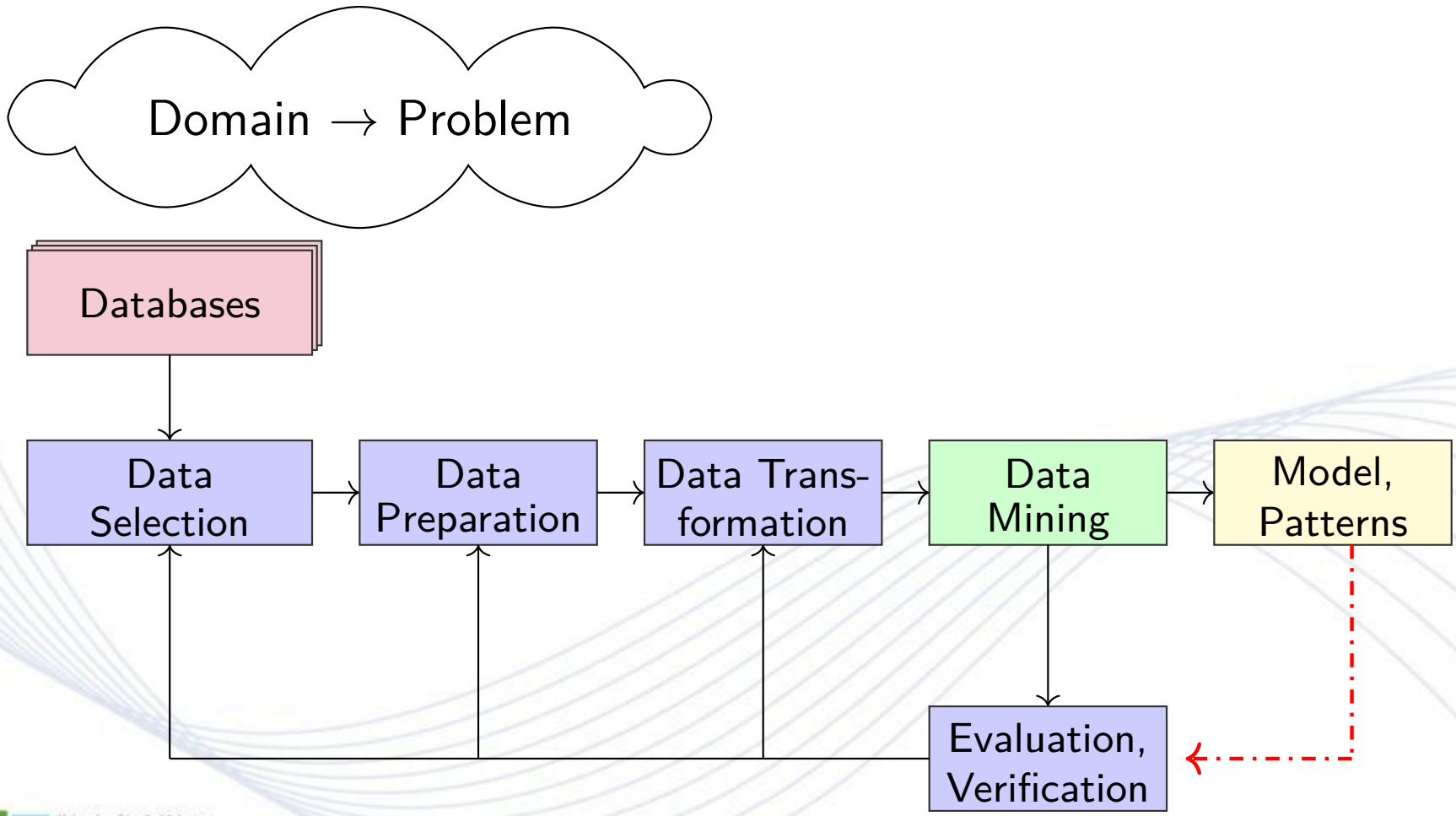
For example, a data mining project dealing with a regression problem in inverse kinematics of robots constitutes one specific context.

The more values for different context dimensions are fixed, the more concrete is the data mining context.

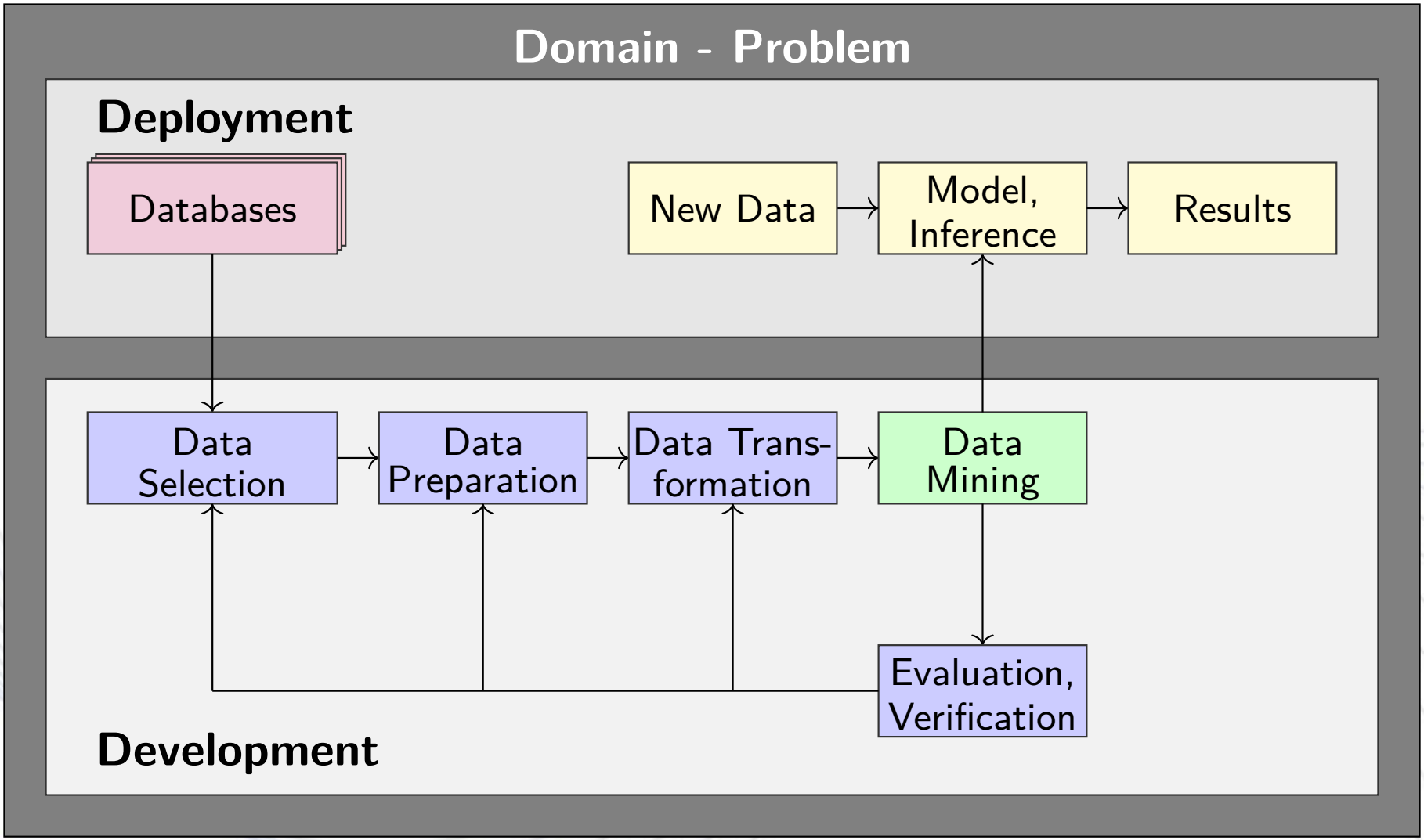
Knowledge Discovery in Databases Process



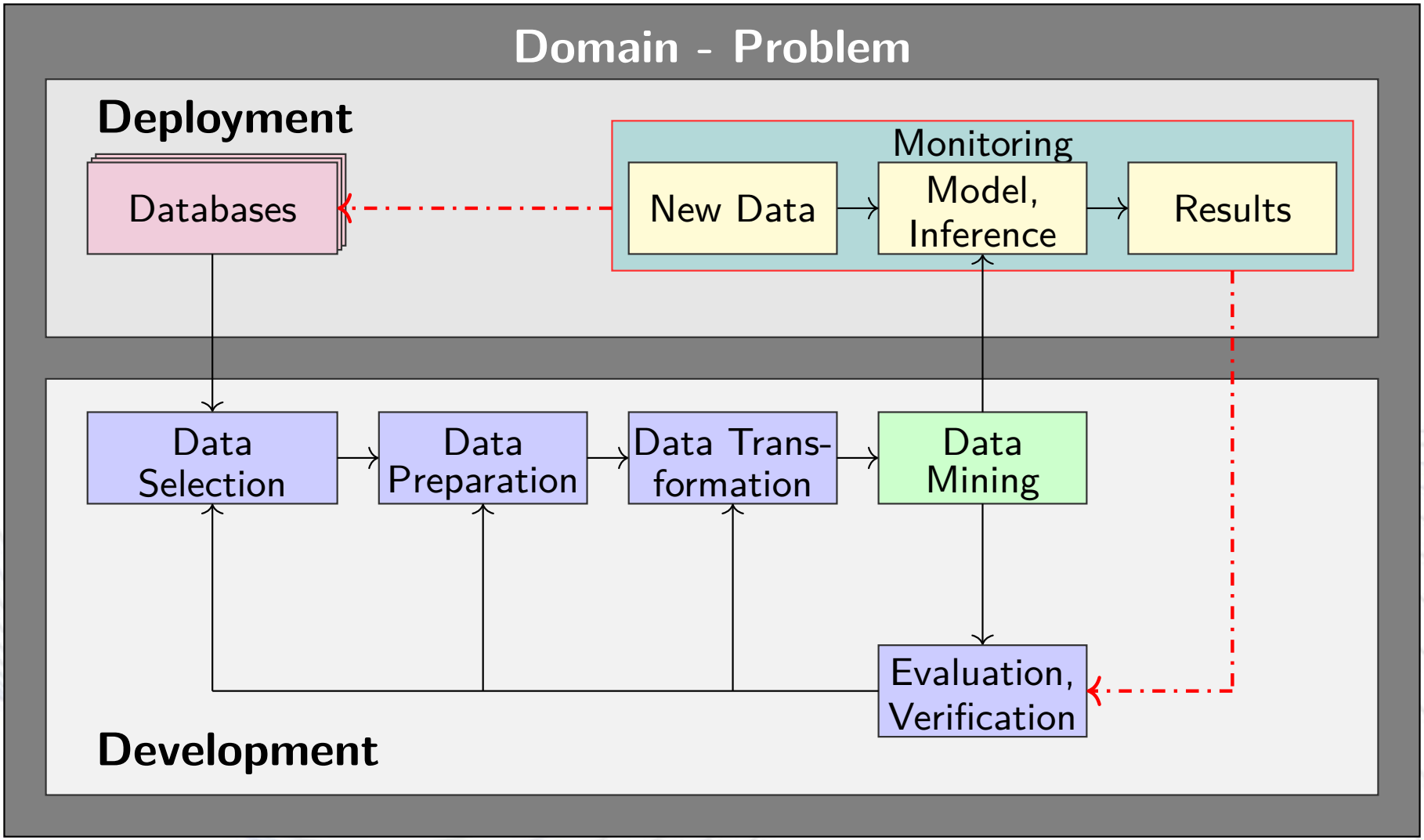
Knowledge Discovery in Databases Process



KDD Process - Machine Learning Pipeline



KDD Process - Machine Learning Pipeline



Knowledge Discovery Databases Process Steps

Learning the application domain:

- relevant prior knowledge and goals of application
  $\Rightarrow$ Creating a target data set: data selection
- Data cleaning and preprocessing: (may take 60% of effort!)
- Data reduction and transformation:
  $\Rightarrow$ Find useful features, dimensionality/variable reduction, representation
- Choosing functions of data mining
  $\Rightarrow$ summarization, classification, regression, association, clustering
- Evaluation/Verification

Iterative execution of the process steps

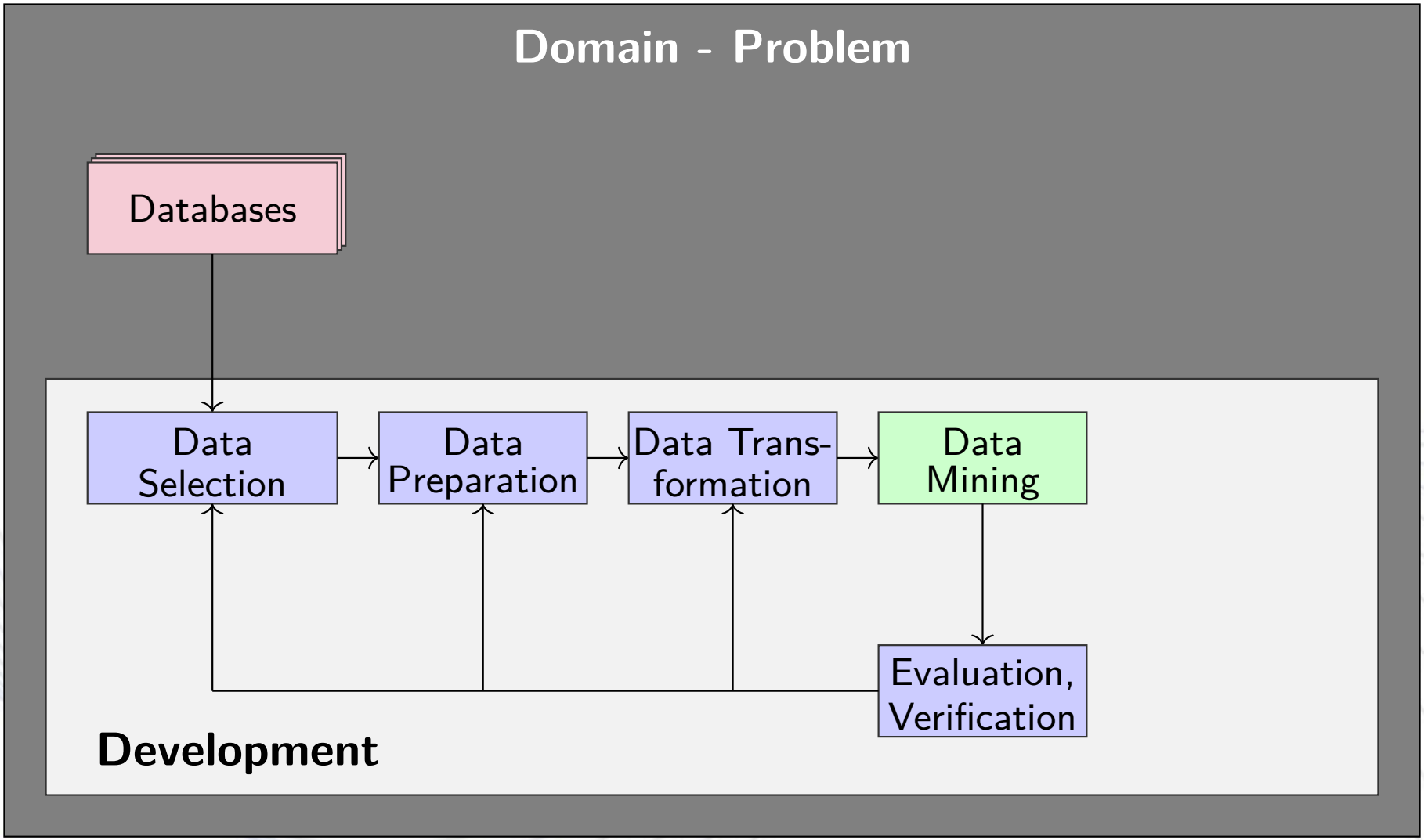
KDD Process Steps

Choosing functions of data mining

- Choosing functions of data mining
  $\Rightarrow$ summarization, classification, regression, association, clustering
- Choosing the mining algorithm(s)
- Data mining: search for patterns of interest
- Pattern evaluation and knowledge presentation
  $\Rightarrow$ visualization, transformation, removing redundant patterns, etc.
- Use and integration of discovered knowledge

Development/Modeling Process

KDD Process - Machine Learning Pipeline



Task “Problem Understanding”

Task "Problem Understanding"

The first objective of the data analyst is to thoroughly understand, what is really needed.

Often the customer has many competing objectives and constraints that must be properly balanced.

The analyst's goal is to uncover important factors, at the beginning, that can influence the outcome of the project.

A possible consequence of neglecting this step is to expend a great deal of effort producing the right answers to the wrong questions.

“Problem Understanding” – Outputs: Background

Record the information that is known about the problem at the beginning ofthe project.

"Problem Understanding" – Outputs: Problem's Objectives

Describe the customer's primary objective. In addition to the primary objective, there are typically other related questions that the customer would like to address.

“Problem Understanding” – Outputs: Success Criteria

Describe the criteria for a successful or useful outcome to the project.

This might be quite specific and able to be measured objectively, for example, accuracy of a regressionreduction of customer churn to a certain level, or it might be general and subjective, such as “give useful insights into the relationships”.

In the latter case, it should be indicated who makes the subjective judgment.

Example: Definition of the problem "Inverse Kinematic"

Now we define our problem:

We want to calculate the inverse kinematic a serial robot.

Example: Definition of the problem "Inverse Kinematic"

Normally, there are serveral algorithms in use:

- Newton-Raphson,
- Newton-Raphson with Jacobian Transpose
- Cyclic Coordinate Descent (CCD),
- Damped Least Squares (Levenberg-Marquardt),
- Broyden-Fletcher-Shanno (BFSD) algorithm
- Lagrange Multiplier
- ...

If you want to use this algorithms, their time consumption is not well-defined.
Therefore, you have to use special hardware in a real-time environment.

Example: Definition of the problem "Inverse Kinematic"

Now, we use a neural network

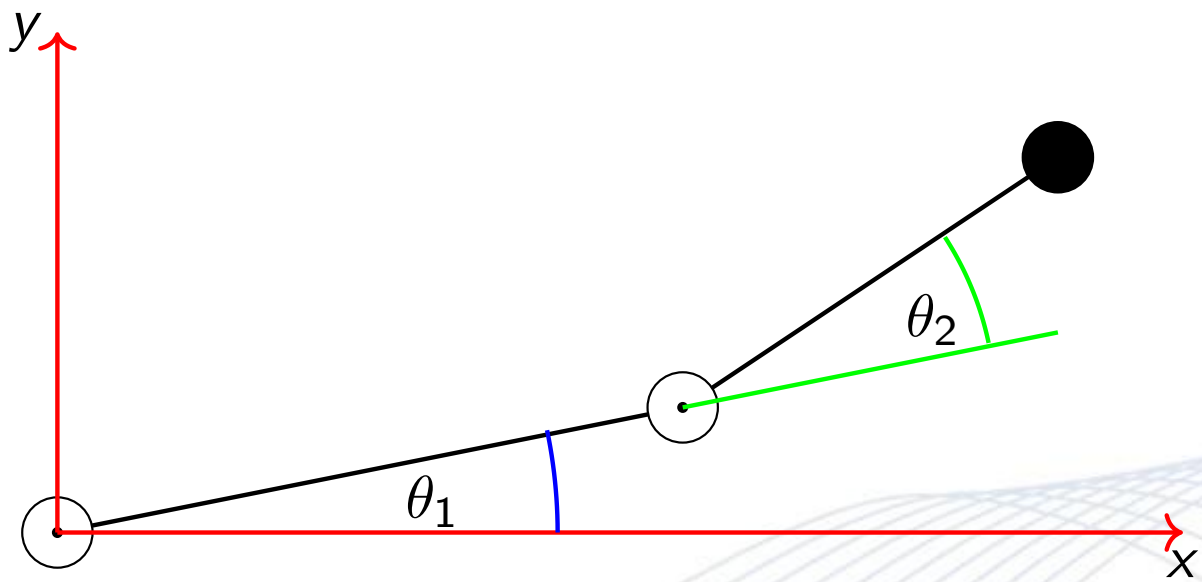
- well-defined time consumption
- fast
- easy to implement

Question:

- accuracy?
- reliable?

Application - Direct and Inverse Kinematics

Database application example $\Rightarrow$ end-effector position

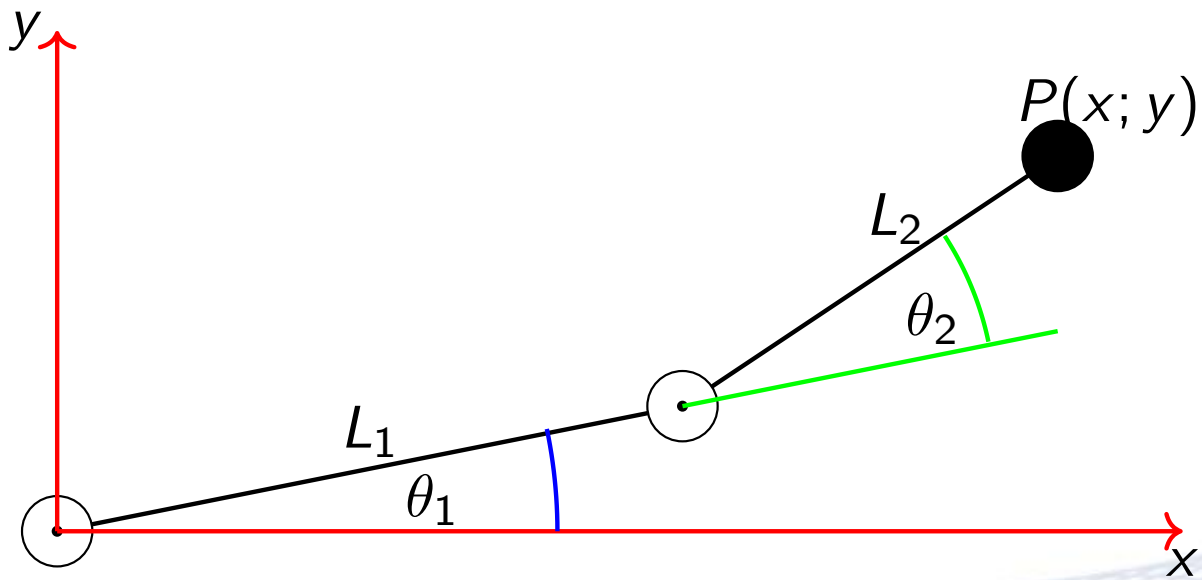


input: $(\theta_1, \theta_2) \rightarrow$ output: $\binom{x}{y}$

θ_1, θ_2 are joint position

x,y are end-effector positions

Example: Direct Kinematic of a SCARA



Is $P(x, y)$ the end point of the second arm, so we get for the coordinates of P:

$$\begin{aligned} x &= L_1 \cdot \cos(\theta_1) + L_2 \cdot \cos(\theta_1 + \theta_2) \\ y &= L_1 \cdot \sin(\theta_1) + L_2 \cdot \sin(\theta_1 + \theta_2) \end{aligned}$$

Example: Direct Kinematic of a SCARA

We get the function F:

$$F(\theta_1, \theta_2) = \begin{pmatrix} L_1 \cdot \cos(\theta_1) + L_2 \cdot \cos(\theta_1 + \theta_2) \\ L_1 \cdot \sin(\theta_1) + L_2 \cdot \sin(\theta_1 + \theta_2) \end{pmatrix}$$

Task “Database”

Task “Database”

I want to know . . .

- What is Data?
- What is information?
- What kind of data do we need?

Task “Database”

I want to know . . .

- What is Data?
- What is information?
- What kind of data do we need?

Task “Database”

This task involves more detailed fact-finding about all of the resources, constraints, assumptions concerning the data.

Outputs: Description of the Database

- data
  - fixed extracts,
  - access to live,
  - warehoused, or
  - operational data,
  - structure
  - database type(s)
- personnel
  - domain experts,
  - data experts,
  - technical support,
- computing resources (hardware platforms), and
- software (database tools, other relevant software).

Database

- Domain Knowledge

  $\Rightarrow$ choose your database:

- Data as much as possible
  - Image: Ok or not Ok
  - Voice: Ok or not Ok
  - Text: Ok or not Ok
  - e.g. (face, name)

Database

Database is a systematic collection of data;
Computer structures that save, organize, protect, and deliver data

Database Management System (DBMS) is a collection of programs which enables its users to access database, manipulate data, and help in representation of data; also helps control access to the database by various users

Database

Text Database:

- The simplest form of database
- When data is organized in a text file in rows and columns, it can be used to store, organize, protect, and retrieve data
- For example: name , family name, age , . . .

Database

Desktop database programs:

- more complex than a text database but intended for a single user
- A Microsoft Excel spreadsheet or Microsoft Access
- 4 major commands:
  - Select,
  - update,
  - insert, and
  - delete
- benefit: speed of changing data, and the ability to store large amounts of data while keeping performance of the system

Database - Examples

Open Datasets:

- MS-COCO
- ImageNet
- VisualQA
- CIFAR-10
- AUVSI: Unmanned Vehicles and Robotics products operating air ground and maritime domain, finding new platforms, new information is added to database daily email

http://www.analyticsvidhya.com/blog/2018/03/comprehensive-collection-deep-learning-datasets/

Database Application

Direct Kinematic

$$F(\theta_1, \theta_2) \quad = \quad \begin{pmatrix} x \\ y \end{pmatrix} \quad \Rightarrow \quad \text{data set} \quad (\theta_1, \theta_2, x, y)$$

How much data?

As much as possible?

Database Application

Better: **a plan!**

- $\rightarrow$ covering the working space
- $\rightarrow$ considering a tolerance ε_1

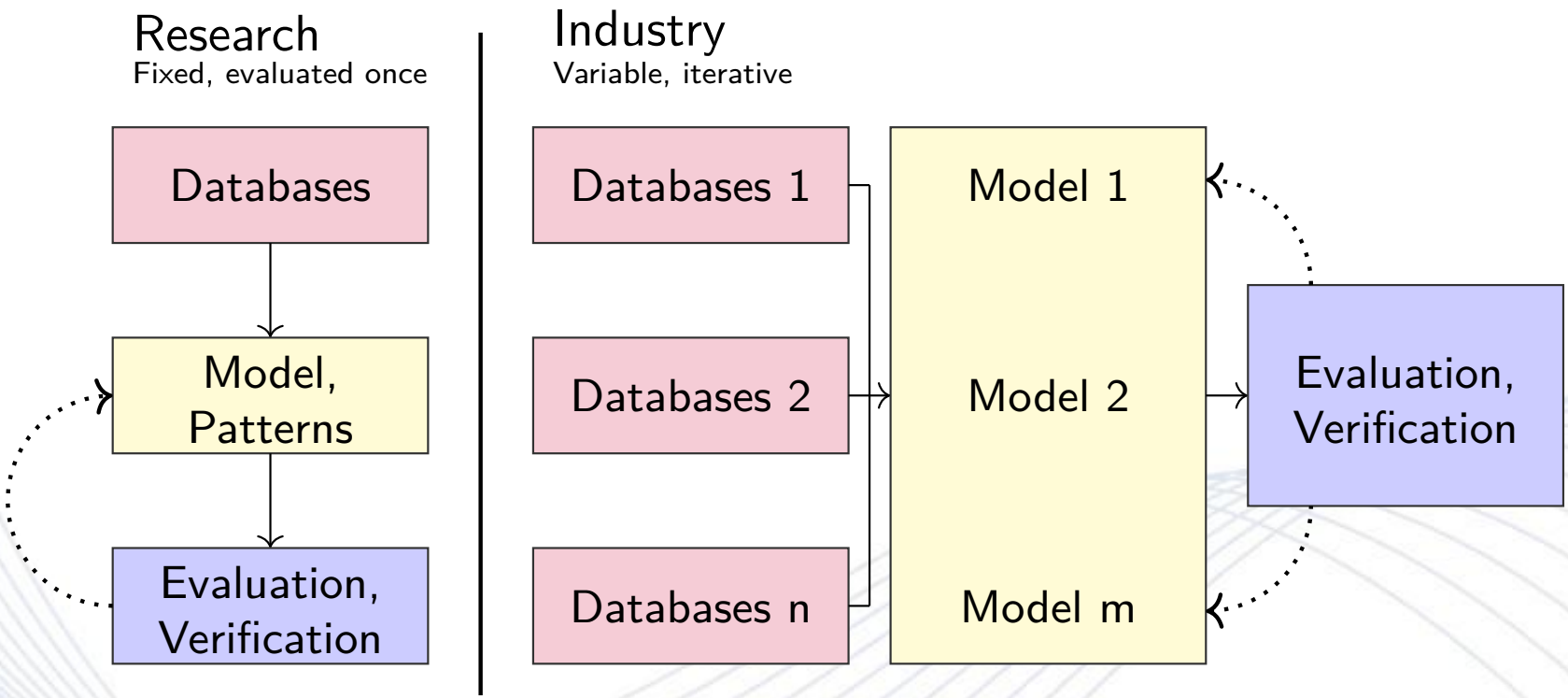
$$\left\| \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} - \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} \right\| < \varepsilon_1$$

- $\rightarrow$ well-defined grid

as little as possible, but enough

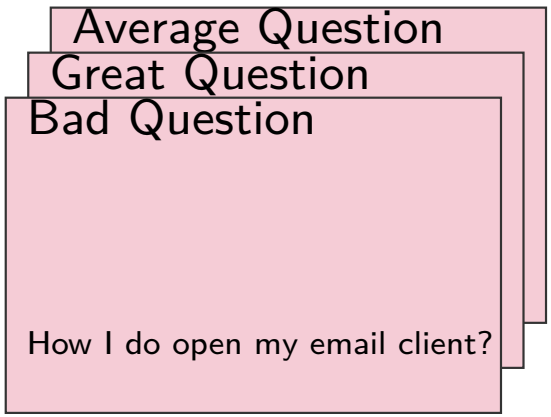
$\Rightarrow$ creating a database based on domain knowledge

Attention: Research/Learning vs. Industry



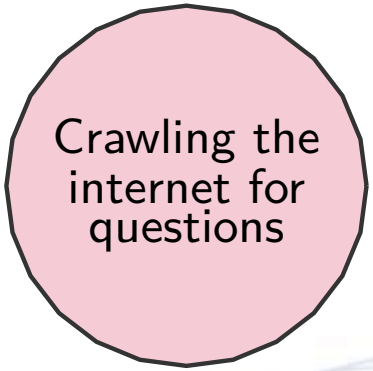
Datasets are fixed in research, but part of the product in industry

Attention: Research/Learning vs. Industry



Labeled data: our labels are exactly what we are trying to mesure

Weak labels: our labels (likes) are only slightly correlated with our product goal



Unlabeled data: we need to gather and label data ourselves

Data availability versus data usefulness

Phase: “Data Selection”

Phase: "Data Selection"

- The formulation of the task should be sufficiently general and sufficiently concrete to measure both hypotheses and the success of the process
- Checking the data stock for availability, completeness and admissibility of the analysis

Phase: "Data Selection"

Decide on the data to be used for analysis. Criteria include relevance to the data mining goals, quality, and technical constraints such as limits on data volume or data types. Note that data selection covers selection of attributes (columns) as well as selection of records (rows) in a table.

Phase: "Data Selection" – Output: Rationale for Inclusion/Exclusion

List the data to be included/excluded and the reasons for these decisions.

Phase: "Data Selection" – Outputs

- Determination of the starting position for the process:
  - What data is available?
  - How is the data available?
- Selection of the relevant data:
  - Which part of the data is relevant?
  - Is further data needed?
  - Are there dependencies between the data?
- Definition of the task:
  - What should be investigated? (e.g. probability of failure of a machine)
  - Which decision should be made? (e.g. premature maintenance of a machine)

“Data Selection”

Data mapping: The process of defining how individual fields are

- mapped,
- modified,
- joined,
- filtered,
- aggregated,
- etc.

to produce the final desired output.

It’s a traditionally work.
Try to define the transformation rules (e.g. visual ETL tools, transformation languages).

“Data Selection”

- Code generation:
  The process of generating executable code (e.g. SQL, Python, R, or other executable instructions) that will transform the data based on the desired and defined data mapping rules.

- Code execution:
  The step whereby the generated code is executed against the data to create the desired output.
  - The executed code may be tightly integrated into the transformation tool, or
  - it may require separate steps by the developer to manually execute the generated code.

“Data Selection” – Code Generation

The process of generating executable code (e.g. SQL, Python, R, or other executable instructions) that will transform the data based on the desired and defined data mapping rules.

“Data Selection” – Code Execution

The step whereby the generated code is executed against the data to create the desired output.

- The executed code may be tightly integrated into the transformation tool, or
- it may require separate steps by the developer to manually execute the generated code.

“Data Selection” – Data Review

- The final step in the process, which focuses on ensuring the output data meets the transformation requirements.
- It is typically the business user or final end-user of the data that performs this step.
- Any anomalies or errors in the data that are found and communicated back to the developer or data analyst as new requirements to be implemented in the transformation process.

Task "Collect Initial Data" (CID)

Acquire the data (or access to the data) listed in the project resources. This initial collection includes data loading, if necessary for data understanding.

For example, if you use a specific tool for data understanding, it makes perfect sense to load your data into this tool. Thiseffortpossiblyleadsto initialdata preparation steps.

Note: if you acquire multiple data sources, integration is an additional issue, either here or in the later data preparation phase.

“Data Selection” – Output: Initial Data Collection Report

List the dataset(s) acquired, together with their locations, the methods used to acquire them, and any problems encountered. Record problems encountered and any resolutions achieved.

This will aid with future replication of this project or with the execution of similar future projects.

“Data Selection” – Describe Data

Examine the “gross” or “surface” properties of the acquired data and report on the results.

“Data Selection” – Output: Data description report

Describe the data that has been acquired, including

- the format of the data,
- the quantity of data (for example, the number of records and fields in each table),
- the identities of the fields, and
- any other surface features

which have been discovered.

Evaluate whether the data acquired satisfies the relevant requirements.

Example Direct and Inverse Kinematic

As an example in robotics some data may have different unit, which need to be unified:

- Angle, (Degree, Radian, . . .)
- Rotation speed (Revolution per minute, Radian per second, . . .)
- Operation modes (mode A, mode B)
- Also, some data may not be relative to the knowledge we are interest to extract.
- For example, if we are investigating about angular speed, the data about the temperature may not be applicable.

“Data Selection”

Inverse Kinematic

- Not easy
- Not unique answer

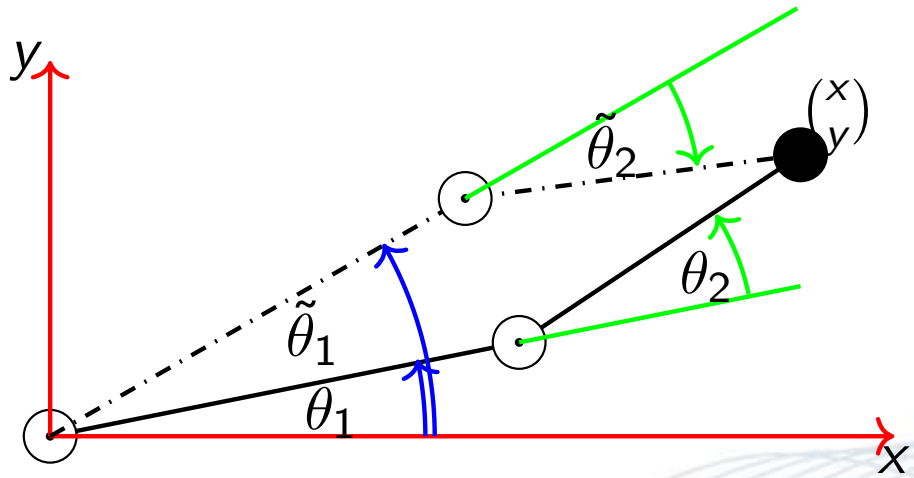
but

Direct Kinematic

easy to calculate $\rightarrow$ $F(\theta_1, \theta_2) = \binom{x}{y}$

“Data Selection” – Calculate the data

calculate all values using F and put them in a database



$$F(\theta_1, \theta_2) = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = F(\tilde{\theta}_1, \tilde{\theta}_2)$$

“Data Selection” - Calculate the Data

calculate all values using F and put them in a database, but

$$\tilde{\theta}_2 = -\theta_2$$

choose, e.g.:

$$\theta_2 > 0$$

We can not use all combinations!
It depends on the application!!

“Data Selection”

Unique solution is needed!

restriction is needed! $\leftarrow$ we need domain knowledge!!

Phase “Preparation”

Data Preparation

- Goal:
  Quality improvement of the selected data regarding completeness and consistency
- Differentiation in data organization and presentation based on several data sources, transfer to a uniform database; e.g. merging of data from the machine and from work preparation
- Adjustment approaches to the data:
  - Enrichment: Subsequent collection of missing and incorrect values is very costly
  - Reduction: Removal of data records with missing and incorrect values can lead to loss of relationship patterns
  - Integration: Replacement of missing and incorrect characteristics or addition of further characteristics leads to higher complexity
- Feasibility of the analysis depends on complexity

Data Preparation

How does the algorithm deal with anomalies?

- Missing values?
- Checking values? plausible?
- outliers?

Data Preparation

In a knowledge extraction process, all of the process is relied on the data source. Good quality input data leads to good quality results. In KDD process:

Garbage in - Garbage out.

The raw recorded data may have different resolution, bias, range, validity, noise, or etc. Always there is a chance that a datum is not recorded properly which may ruin the knowledge extraction process.

Data Preparation – Example

As an example, in a robotic system some gathered data may not be use and should be removed or replaced:

- Disconnected/broken sensor
- Dirty sensor
- Too much noise
- Not calibrated
- Data is recorded in an undesired time span

“Preparation” – Explore Data

This task addresses data mining questions using

- querying,
- visualization, and
- reporting techniques.

“Preparation” – Explore Data

This includes

- distribution of key attributes (for example, the target attribute of a prediction task),
- relationships between pairs or small numbers of attributes,
- results of simple aggregations,
- properties of significant sub-populations, and
- simple statistical analyses.

These analyses may directly address the data mining goals; they may also contribute to or refine the data description and quality reports, and feed into the transformation and other data preparation steps needed for further analysis.

“Preparation” – Output: Data Exploration Report

Describe results of this task, including first findings or initial hypothesis and their impact on the remainder of the project.

If appropriate, include graphs and plots to indicate data characteristics that suggest further examination of interesting data subsets.

“Preparation” – Verify Data Quality

Examine the quality of the data, addressing questions such as:

- Is the data complete (does it cover all the cases required)?
- Is it correct, or does it contain errors and, if there are errors, how common are they?
- Are there missing values in the data? If so,
  - how are they represented,
  - where do they occur, and
  - how common are they?

“Preparation” – Output: Data Quality Report

List the results of the data quality verification; if quality problems exist, list possible solutions. Solutions to data quality problems generally depend heavily on both data and business knowledge.

“Preparation” – Clean Data

Raise the data quality to the level required by the selected analysis techniques. This may involve selection of clean subsets of the data, the insertion of suitable defaults, or more ambitious techniques such as the estimation of missing data by modeling.

“Preparation” – Output: Data Cleaning Report

Describe what decisions and actions were taken to address the data quality problems reported during the task “Verify Data Quality” of the Data Understanding phase.
Transformations of the data for cleaning purposes and the possible impact on the analysis results should be considered.

“Preparation” – Construct Data

This task includes constructive data preparation operations such as the production of derived attributes or entire new records, or transformed values for existing attributes.

“Preparation” – Outputs: Derived Attributes

Derived attributes are new attributes that are constructed from one or more existing attributes in the same record.

Example: area = length · width.

“Preparation” – Outputs: Generated Records

Describe the creation of completely new records.

Example: Create records for customers who made no purchase during the past year. There was no reason to have such records in the raw data, but for modeling purposes it might make sense to explicitly represent the fact that certain customers made zero purchases.

“Preparation” – “Integrate Data”

These are methods whereby information is combined from multiple tables or records to create new records or values.

"Preparation" – Output: Merged Data

Merging tables refers to joining together two or more tables that have different information about the same objects.

Example: a retail chain has one table with information about each store's general characteristics (e.g., floor space, type of mall), another table with summarized sales data (e.g., profit, percent change in sales from previous year), and another with information about the demographics of the surrounding area. Each of these tables contains one record for each store. These tables can be merged together into a new table with one record for each store, combining fields from the source tables.

“Preparation” – Output: Merged Data

Merged data also covers aggregations. Aggregation refers to operations in which new values are computed by summarizing information from multiple records and/or tables.

For example, converting a table of customer purchases where there is one record for each purchase into a new table where there is one record for each customer, with fields such as number of purchases, average purchase amount, percent of orders charged to credit card, percent of items under promotion, etc..

Phase: "Data Transformation"

Data Transformation

- The type and sequence of the analysis are determined by determining the analysis procedure

"Data Transformation"

- Is θ's unit radiant or degree?
- Definition set:
  - $\theta \in [0; \pi]$?

    or
  - $\theta \in [-75°; 280°]$?

    or
  - $\theta \in [0; 1]$ (that's a standard!)?

    or
  - $\theta \in [-1; 1]$ (that's a standard, too!)?

$\Rightarrow$ depending on the algorithm θ will be chosen

“Dta Transformation”

How does the algorithm need the input data?

- list
- list of lists
- arrays

Is there a difference between

- just input
- input and training

“Data Transformation” – Format Data

Formatting transformations refer to primarily syntactic modifications made to the data that do not change its meaning, but might be required by the modeling tool.

"Data Transformation" – Output: Reformatted Data

Some tools have requirements on the order of the attributes, such as the first field being a unique identifier for each record or the last field being the outcome field the model is to predict.

It might be important to change the order of the records in the dataset. Perhaps the modeling tool requires that the records be sorted according to the value of the outcome attribute. Commonly, records of the dataset are initially ordered in some way, but the modeling algorithm needs them to be in a fairly random order. For example, when using neural networks, it is generally best for the records to be presented in a random order, although some tools handle this automatically without explicit user intervention.

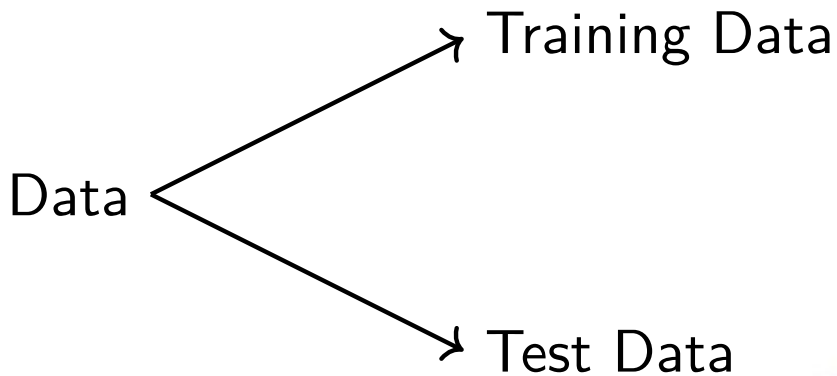
“Data Transformation” – Output: Reformatted Data

Additionally, there are purely syntactic changes made to satisfy the requirements of the specific modeling tool.

Examples: removing commas from within text fields in comma-delimited data files, trimming all values to a maximum of 32 characters.

Data Transformation

Split the data:



How?

Data Transformation

Split the data:

How?

The normal proportion is 80% training data and 20% test data.

- randomly?
- first 80%
- every 5^{th} element is a test data
- ...

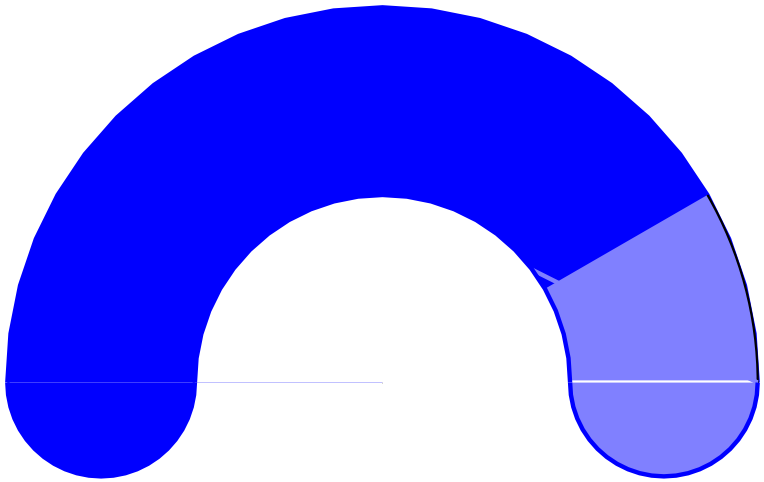
Data Transformation

Normally choosing data isn't optimal!

Use different strategies.

Data Transformation - Example

Covering the working space?



Perhaps, you get 80% left side of data

Is some of data missing?

$\Rightarrow$ Domain knowledge is useful

Data Transformation

A large part of variables in a robotic system are dynamic values, and their values change in during the operation. Therefore the true knowledge can be discovered, only when the values of a parameter is known in time domain.

Normally are all recorded data in a database for a robotic system is a time-stamped observation, described by a set of feature-value pairs.

The time value is given on the time scale in a given resolution, such as millisecond, days, months, or years.

Data Transformation - Example Direct and Inverse Kinematic

The data transformation is critical to detect important events in robotic systems such as:

- Predicting events and behaviors
- Change points in time domain
- Parameters associated with time, like: velocity and acceleration etc.
- Aging process
- Trajectory

Data Mining

Data Mining

- Goal:
  Determine relationship patterns in the database and build a model
- Approach to data analysis in order to determine relationship patterns in a database and to map them through logical or mathematical descriptions
- Data Mining is an analytical approach to test hypotheses by means of questions such as “How many units of the product group backpacks were sold in all stores using the standard shipping method?”
- Possible result of the analysis in the form of a model: “The sale of goods of the departmental group backpacks in the North branch was carried out via the standard shipping method in 95% of the cases.”

Data Mining

Data mining is a process of finding previously unknown patterns in databases to build predictive models

Learning Algorithms:

- Supervised Learning
- Unsupervised Learning
- Reinforced Learning

Supervised Learning

Not only the algorithm has the input, but also the desired output
complete and labelled data are given to the algorithm
A teacher guides the system to learn the pattern
The algorithm should predict and put labels to the unlabelled data
It has feedback mechanism

Supervised Algorithms:

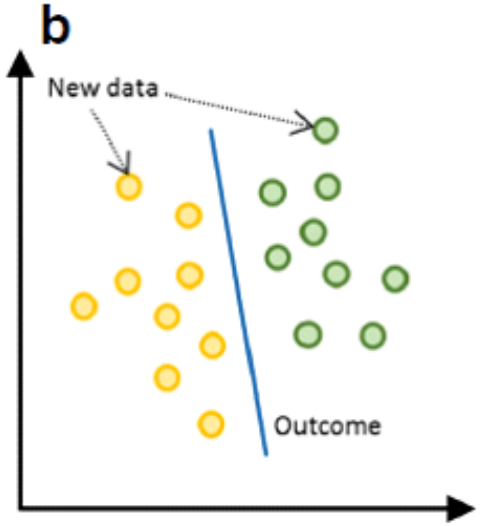
- Decision Tree
- Random Forest
- Support Vector Machine
- Artificial Neural Networks
- K Nearest Neighbors

Supervised Learning Types

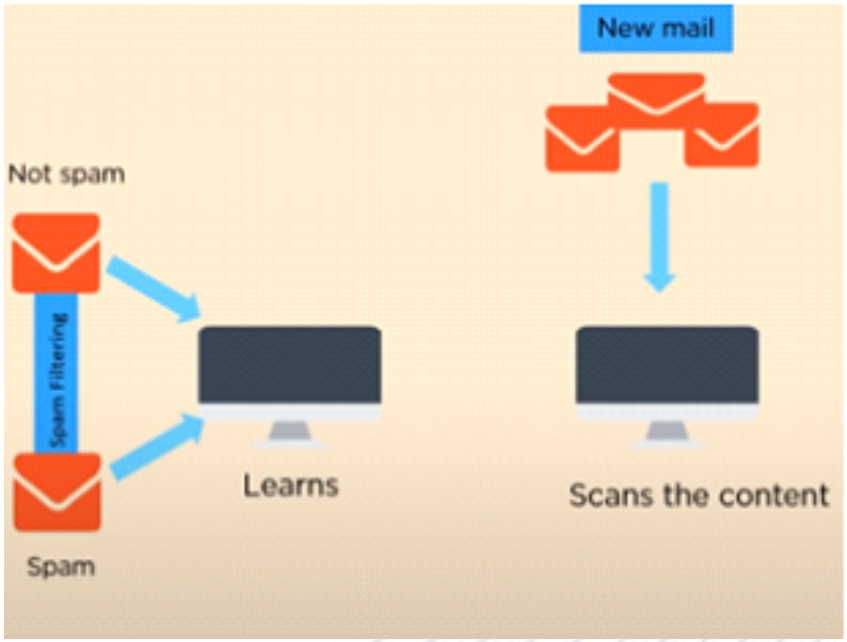
- Classification
- Regression

Classification: When the output variable is categorical e.g. 2 or more classes (yes/no, true/false), we make use of classification

Classification Examples:



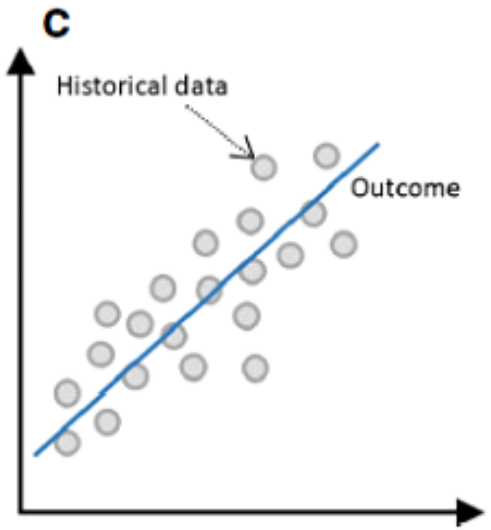
Supervised Learning Types



Source: my own

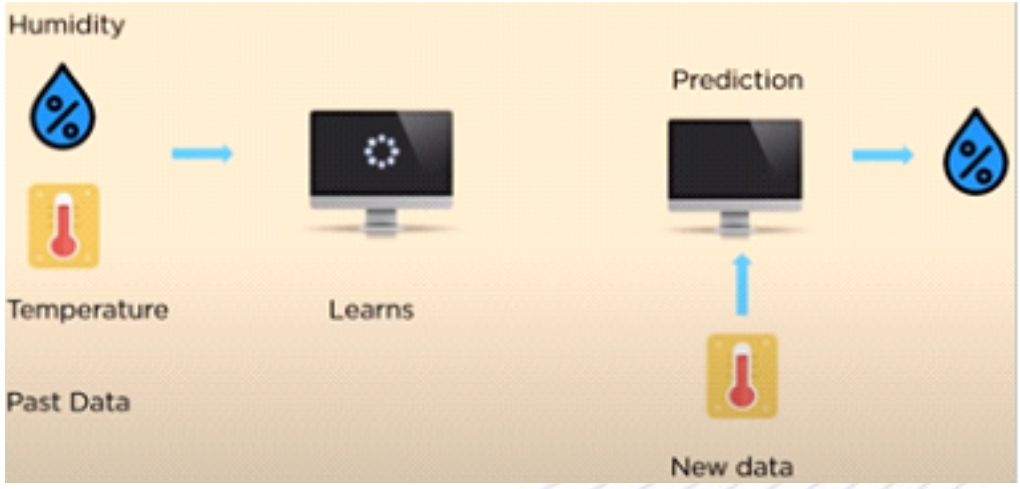
Supervised Learning Types

Regression: Relationship between two or more variables where a change in one variable is associated with a change in other variable there are some examples of Regression:



Supervised Learning Types

Weather Prediction

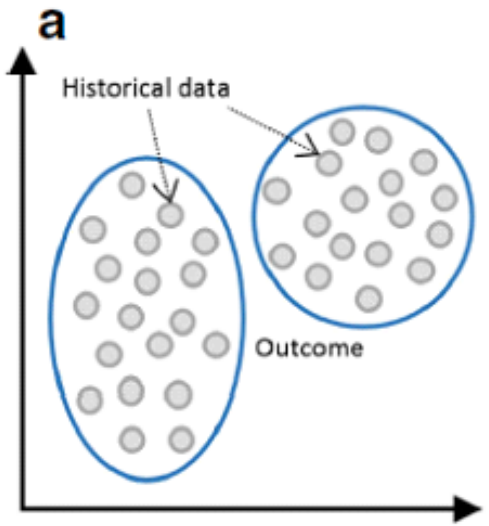


Source: my own

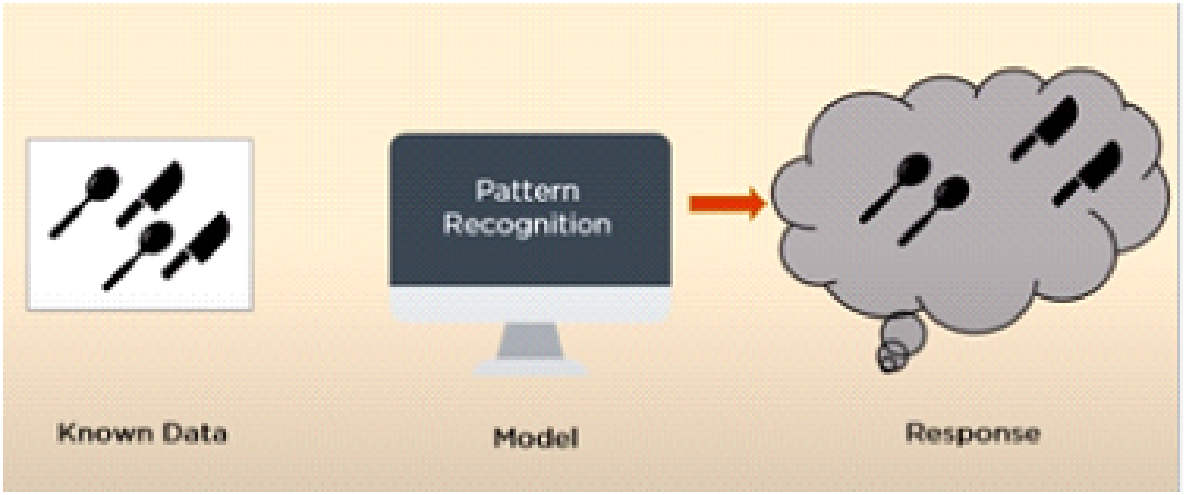
Unsupervised Learning

Unsupervised Learning: The desired output is not available, the algorithm is trained using data that is unlabelled. It does not have feedback mechanism.

Unsupervised Learning Example:



Unsupervised Learning



Source: my own

Unsupervised Algorithms:

- K Means
- Mean Shift Clustering

Unsupervised Learning Types

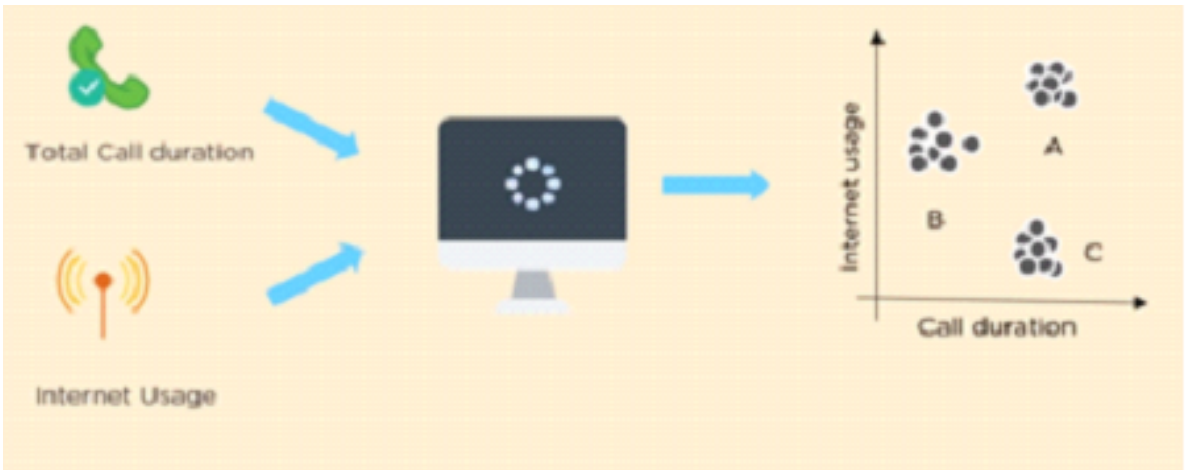
- Clustering
- Association

Clustering: The method of dividing the objects into clusters which are similar between them and are dissimilar to the objects belonging to another cluster

Unsupervised Learning Types

There are examples of Clustering:

An example of Clustering: Suppose a telecome company wants to reduce its customer churn rate by providing personalized call and data plans:

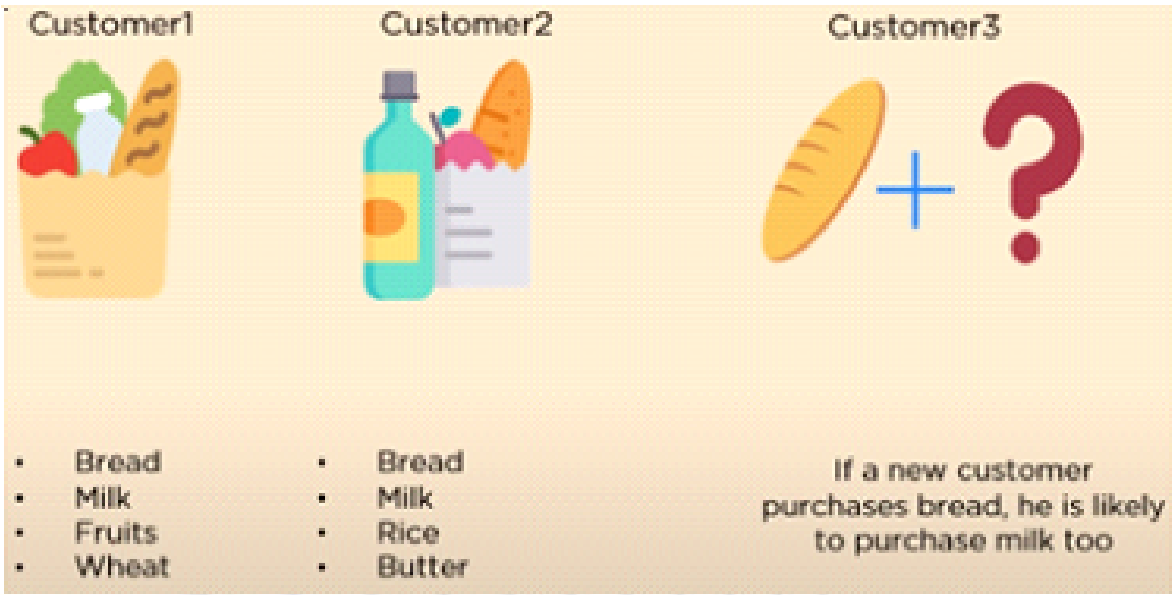


Source: my own

Unsupervised Learning Types

Association: Discovering the probability of the co-occurrence of items in a collection

There are some examples of Association:



Source: my own

Reinforced Learning

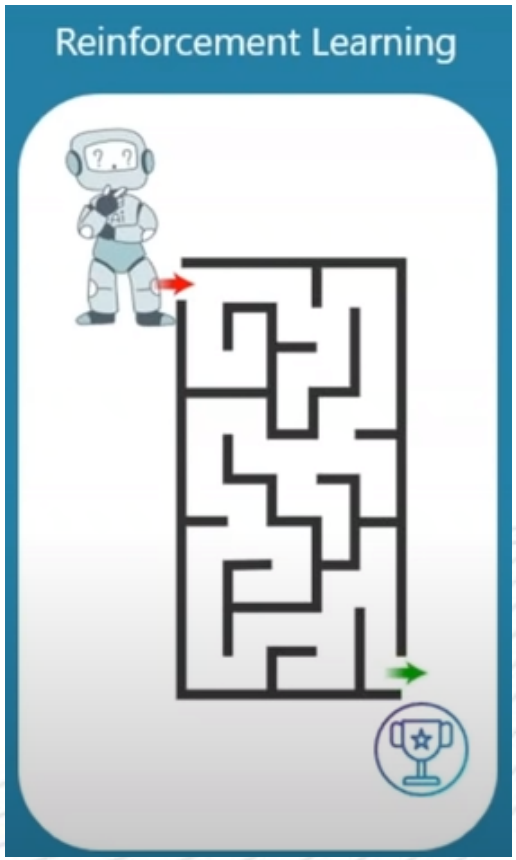
Reward and punishment based learning. For certain ways achieving a target, we reward the algorithm and for other ways of achieving we punish the system.

We encourage the algorithm to find the solution or achieve the target in such a way that is most rewarding and least punishing

Example: Reaching to a toy by a child or robot

Reinforced Learning

Example: Reaching to a toy by a child or robot



Source: my own

Reinforced Learning

Reinforcement Learning Algorithms:

- Policy Gradient
- AlphaZero

Data Mining Applications

- Data mining is a process of finding previously unknown patterns in databases to build predictive models.
- Data mining applications:
- Robot Control
  - Routing, localization in a wireless sensor network (WSN) $\rightarrow$ Genetic Algorithm
  - Control a robotic arm and gripper $\rightarrow$ Support Vector Machine
  - Behavioral model for prevention of collision situation
  - Constructing a map of unknown environment by autonomous robots $\rightarrow$ K-means

Data Mining Applications - Robot Vision

- Being aware of environment by rescue autonomous robots which are equipped with vision system.
- Making decision based on visual input (image and video)
- Autonomous Underwater Robots imagery of biological species, physical formations, and object detection
- Robot Behavior
- classify robot actions for imitation learning by hybrid neural data mining technique
- Robot Behavior
  - Improving robots' behavior through web mining, social media, and experienced patterns as a knowledge source alongside with contextual information to discover users' needs

Data Mining Application – Inverse Kinematics

We use a simple neural network with one hidden layer and 50 neurons.

“Data Mining” – Select Modeling Technique

As the first step in modeling, select the actual modeling technique that is to be used.

Although you may have already selected a tool, this task refers to the specific modeling technique, e.g., decision-tree building with 5.0, or neural network generation with back propagation.

If multiple techniques are applied, perform this task separately for each technique.

“Data Mining” – Outputs: Modeling Technique

Document the actual modeling technique that is to be used.

Especially, document all hyperparameters.

“Data Mining” – Outputs: Modeling Assumptions

Many modeling techniques make specific assumptions about the data.

For example, that all attributes have uniform distributions, no missing values allowed, class attribute must be symbolic, etc..

Record any such assumptions made.

“Data Mining” – Generate Test Design

Before we actually build a model, we need to generate a procedure or mechanism to test the model’s quality and validity.

For example, in supervised data mining tasks such as classification, it is common to use error rates as quality measures for data mining models. Therefore, we typically separate the dataset into train and test sets, build the model on the train set, and estimate its quality on the separate test set.

“Data Mining” – Output: Test Design

Describe the intended plan for training, testing, and evaluating the models. A primary component of the plan is determining how to divide the available dataset into training, test, and validation datasets.

“Data Mining” – Build Model

Run the modeling tool on the prepared dataset to create one or more models.

“Data Mining” – Outputs: Parameter Settings

With any modeling tool, there are often a large number of parameters that can be adjusted.

List the parameters and their chosen values, along with the rationale for the choice of parameter settings.

“Data Mining” – Outputs: Models

These are the actual models produced by the modeling tool, not a report.

“Data Mining” – Outputs: Model Descriptions

Describe the resulting models.

Report on the interpretation of the models and document any difficulties encountered with their meanings.

“Data Mining” – Assess Model

The data mining engineer interprets the models according to

- his domain knowledge,
- the data mining success criteria, and
- the desired test design.

“Data Mining” – Assess Model

The data mining engineer tries to rank the models. He assesses the models according to the evaluation criteria.

In most data mining projects, the data mining engineer applies a single technique more than once, or generates data mining results with several different techniques.

In this task, he also compares all results according to the evaluation criteria.

“Data Mining” – Outputs: Model Assessment

Summarize results of this task, list qualities of generated models (e.g., in terms of accuracy), and rank their quality in relation to each other.

“Data Mining” – Outputs: Revised Parameter Settings

According to the model assessment, revise parameter settings and tune them for the next model building.

Iterate model building and assessment until you strongly believe that you have found the best model(s).

Document all such revisions and assessments.

Evaluation and Verification

Evaluation and Verification

The last step in a KDD process is to verify and evaluate the results.

Definition

Verification:

Verification is the proof that a presumed or asserted fact is true.
→ "Is the result plausible/okay?"

Evaluation:

Evaluation refers to the systematic analysis and assessment of measures and processes.
→ "Is the process okay?"

2 nd Definition

Verification:

"Is the model developed correct?"
or
"Is the model specification complete and is the model implementation correct?"

Evaluation:

"Has the right model been developed?"
or
"Does the model represent the system under investigation and its behaviour sufficiently accurately to answer the research question?"

“Evaluation” – Evaluate Results

This step assesses the degree to which the model meets the objectives and seeks to determine if there is some reason why this model is deficient.

Another option is to test the model(s) on test applications in the real application.

Moreover, evaluation also assesses other data mining results generated.

Data mining results involve models that are necessarily related to the original objectives and all other findings that are not necessarily related to the original objectives, but might also unveil additional challenges, information, or hints for future directions.

"Evaluation" – Outputs: Assessment of Data Mining Results

Attention: Results with respect to Success Criteria

Summarize assessment results in terms of success criteria, including a final statement regarding whether the project already meets the initial objectives.

“Evaluation” – Outputs: Approved Models

After assessing models with respect to business success criteria, the generated models that meet the selected criteria become the approved models.

“Evaluation” – Review Process

At this point, the resulting models appear to be satisfactory and to satisfy the needs.

It is now appropriate to do a more thorough review of the data mining engagement in order to determine if there is any important factor or task that has somehow been overlooked. This review also covers quality assurance issues.

For example: Did we correctly build the model? Did we use only the attributes that we are allowed to use and that are available for future analyses?

“Evaluation” – Output: Review of Process

Summarize the process review and highlight activities that have been missed and those that should be repeated.

Verification

- Static verification tests
  - Cross check
  - Structured code walk-throughs
  - Structural analysis
  - Formal methods
- Dynamic verification tests
  - Double implementation
  - Unit testing
  - Structured debugging walkthroughs

Validation

- Tests of the model structure
  - Tests of theories and assumptions
  - Plausibility check
- Tests of the model behaviour
  - Behavioural reproduction tests
  - Comparison with other methods
  - Behavioural prediction tests
  - Sensitivity analysis
- Tests for model structure and model behaviour
  - Tests of the extreme conditions
  - Tests on the adequacy of the boundaries

Verification – Cross Check

A competent person who has not developed the model or the relevant part of the model reviews the annotated code of the model using the specification.

Verification – Structured Code Walk-Throughs

The implementation of the model is presented in group sessions, ideally consisting of members of the model development team as well as external persons.

The developers present each part of the model in detail and explain the code.

The audience can critique and check for correctness.

Verification – Structural Analysis

This approach uses a control flow diagram/programme flowchart of the model structure. The graph is a representation of all the paths that the model could traverse during its execution.

An examination of the programme flowchart can reveal structural anomalies, such as multiple entry and exit points, excessive levels of nesting within a structure, and the use of unconditional branching.

Verification – Formal Methods

Formal methods are used to try to prove correctness with respect to the implementation of a model.

When this is possible, it is extremely useful, but for models of some complexity, these methods are not practical.

Verification – Double Implementation

Two independent implementations of a model must produce the same result with the same input values and parameter settings.

The teams doing the different implementations should consist of different people.

Verification – Unit Testing

Each component of the model is tested individually.

This is a bottom-up approach, as you start with the smallest building blocks, e.g. functions, and test larger and larger structures.

Verification – Structured Debugging Walk-Throughs

This is appropriate if the code contains bugs that lead to execution errors.

This test tracks the model execution of different test cases line by line. This allows the value of each variable in each state to be examined. This allows the audience to see at which lines of code the model execution leads to an error.

Validation – Tests of Theories and Assumptions

The assumptions underlying the model can be tested on data using statistical methods.

Validation – Plausibility Check

Experts in the field review the conceptual model to determine whether it is correct and appropriate for the intended purpose.

Validation – Behavioural Reproduction Tests

These tests are one of the most important validation techniques.

They consist of comparisons of model and system behaviour under different test scenarios with data not used to parameterise the model. The most common method is the graphical comparison of output variables.

But for stochastic outputs there is also the possibility of hypothesis testing and confidence interval calculations.

Validation – Comparison with other Methods

Another test is to compare the model results with the results of other valid models that address the same question.

The reasons for divergent results should be discussed.

Validation – Behavioural prediction tests

These tests are similar to behavioural reproduction tests with the only exception that the model should reproduce future behaviour.

Therefore, they can be performed at a later time to re-evaluate the model when the future of the system is already known.

Validation – Sensitivity analysis

These tests are important for the development of any model.

The effect on model performance shows whether the model reacts to changes like the real system and whether it behaves plausibly in unusual parameter ranges.

In addition, sensitive parameters need to be determined more precisely, as they significantly influence the results.

Validation – Tests of the extreme conditions

The model should work correctly under extreme conditions.

It is often quite clear what behaviour a real system will show in such a situation.
edskip
E.g. an epidemic disease will die out if the probability of infection is zero.

Validation – Tests on the Adequacy of the Boundaries

The model boundary determines what is included in a model and which parts are not considered important.

Therefore, it must be checked whether there are omitted parts that could have an influence on the model result.

The model passes the test if no theory can be established for any part that would explain such an influence.

Verification and Validation – Example Direct and Inverse Kinematic

In a robotic system, besides the cross-check also usually we can validate the result by comparing it with the known fundamental physics equations.

For example the extracted knowledge normally should never contradict with the mass conservation, energy conservation or Newton's laws of motion.

Other methods?

Review of Results

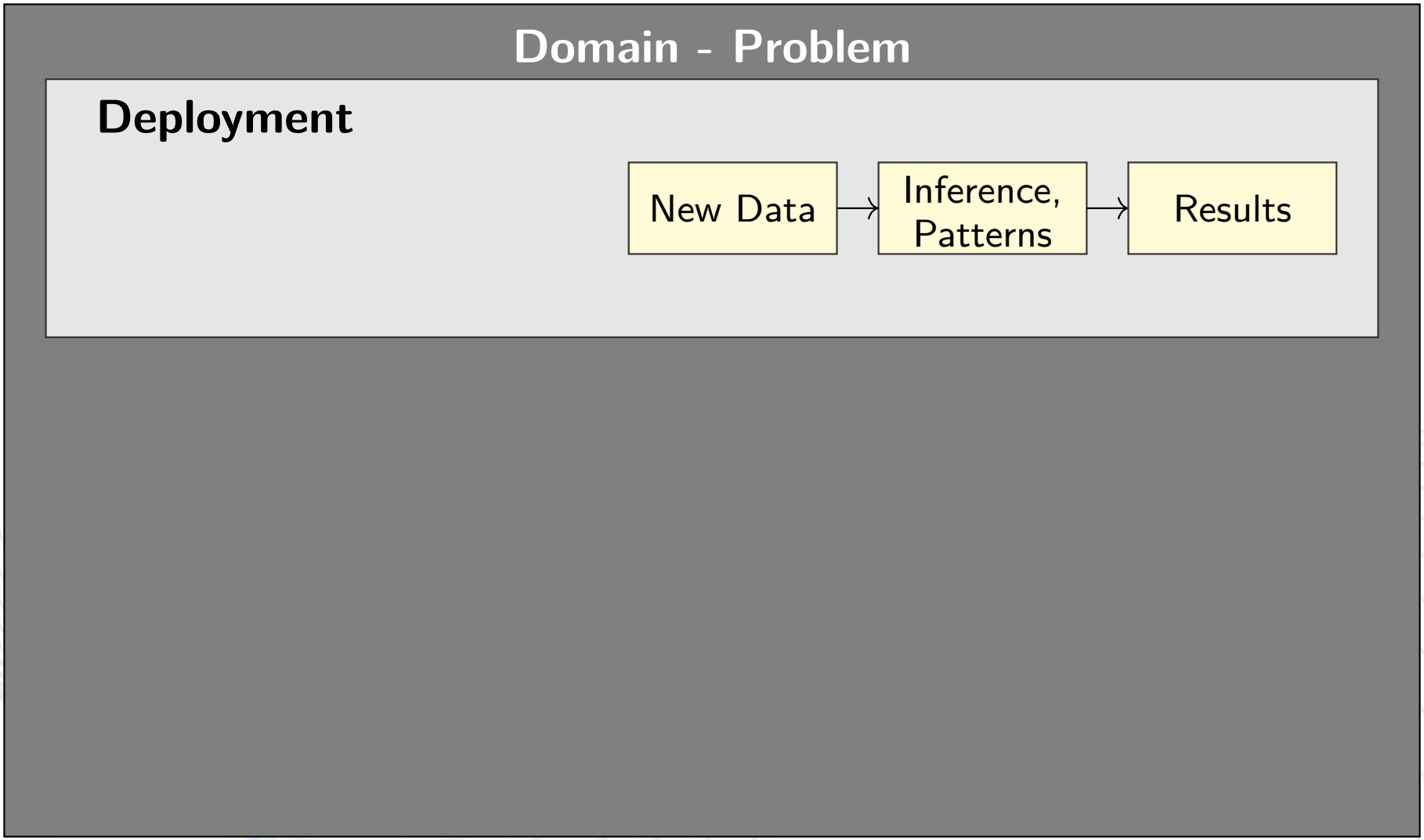
Review of results throughout the KDD process

- The process should be automated if possible, but: Change the situation/disruption, e.g. COVID-19
- Improved algorithms, e.g. Tensorflow 7.0
- ...

$\Rightarrow$ KDD process must be questioned again and again
$\Rightarrow$ KDD process must be set up again if necessary.

Deployment/Inference, Model, Patterns

KDD Process - Machine Learning Pipeline



“Deployment” – Plan Deployment

This task takes the evaluation results and determines a strategy for deployment.

If a general procedure has been identified to create the relevant model(s), this procedure is documented here for later deployment.

“Deployment” – Output: Deployment Plan

Summarize the deployment strategy, including the necessary steps and how to perform them.

“Deployment” – Inference, Model, Patterns

Is the most important phase of a knowledge discovery process
This model can be

- Detecting temporal and sequential patterns
- Expressed as fuzzy relationships between events in consecutive periods

“Deployment” – Inference, Model, Patterns

Transfer the results of the development process in the practice:

- What is the use case?
- How to set the result?
  - ONNX, NNEF, TensorFlow-format, ...
  - C++ code
  - ...
- Realtime?
- Updates?
- Monitoring and Maintenance?

“Deployment” – Inference, Model, Patterns

The model extraction can be done within a supervised or unsupervised KDD process.

- In supervised process: the model of the system is known
- In unsupervised process, the mode is not well known, and will be detected using the data (e.g. the clustering the vibration patterns of the system which may or may not lead to a system fault.)

“Deployment” – Examples

Example of application of discovered models and patterns:

- Providing a detailed model which describes the behavior of a complicated robotic system that can help us to control the system in a more efficient way.
- Fault detection and trouble shooting..
- Detecting the suspicious behavior change of a system, which can be an early alarm for necessity of maintenance.

“Deployment” – Inverse Kinematics

We implement a function with the neural network in the robot’s control system.

The best way:

In our development environment, we save the neural network using a standard data exchange format, e.g. onnx or nnef.
We implement a function in the robot’s control system which read in the neural network.

Phase: “Monitoring and Maintenance”

Phase: "Monitoring and Maintenance"

Monitoring and maintenance are important issues if the data mining result becomes part of the day-to-day business and its environment.

The careful preparation of a maintenance strategy helps to avoid unnecessarily long periods of incorrect usage of data mining results. In order to monitor the deployment of the data mining result(s), the project needs a detailed monitoring process plan. This plan takes into account the specific type of deployment.

“Monitoring and Maintenance” – Output: Monitoring and Maintenance Plan

Summarize the monitoring and maintenance strategy, including the necessary steps and how to perform them.

“Maintenance” – Review of Results

Review of results throughout using the model

- Change in the situation, e.g. COVID-19?
- Change of the process, e.g. defects or new sensors?
- Improved algorithms, e.g. Tensorflow 7.0?
- Improved data management, e.g. doubling the amount of data?
- ...

$\Rightarrow$ KDD process must be questioned again and again
$\Rightarrow$ KDD process must be set up again if necessary.

Final Report and Review of Results

“Final Report” – Produce

At the end of the project, the project team writes up a final report. Depending on the deployment plan, this report may be only a summary of the project and its experiences (if they have not already been documented as an ongoing activity) or it may be a final and comprehensive presentation of the data mining result(s).

“Final Report” - Output: Final Report

This is the final written report of the data mining engagement. It includes all of the previous deliverables, summarizing and organizing the results.

“Final Report” - Output: Final Presentation

There will also often be a meeting at the conclusion of the project at which the results are presented to the customer.

“Review Project”

Assess what went right and what went wrong, what was done well and what needs to be improved.

“Review Project” - Output: Experience Documentation

Summarize important experience gained during the project.

For example,

- pitfalls,
- misleading approaches, or
- hints

for selecting the best suited data mining techniques in similar situations could be part of this documentation. In ideal projects, experience documentation also covers any reports that have been written by individual project members during previous phases of the project.

Guideline

Documents

- Checklist
- Projectplan
- ReadMe.md
- Documentation of the KDD process
- Manual
- Code
- External Documents
- ...

see course “project management” or presentation “Best Practise”

Structure of the Documentation of the KDD process

- Introduction
- Domain knowledge
  - Task's domain knowledge
  - Technical / Machine Learing's domain knowledge
- Methology
  - KDD process, or
  - CRISP-DM, or
  - ...
- Development
- Deployment
- Monitoring/Maintenance
- Conclusion/Open Questions/Optimization
- Literature

Structure of the Documentation of the KDD process

Try to apply the KDD Process to our example:

List the structure.