Data Science Academy

Welcome to our new **DATA SCIENCE ACADEMY**

31.08.2021

Pilot Presentation:
for participants of and use in the pilot only

Boehringer Ingelheim

Data Science is the key to our future success

In all industries, companies recognise that data analytics and AI are central to what they do

In the next few years, it will transform our business

It will keep us ahead of our competitors and help solve complex problems along the whole value chain

It will radically improve how we discover, develop, test and market new treatments

It will give us more insight than ever before into what works, and why

It will help us making the right decisions and speed up our operations

We‘ve founded the Data Science Academy to harness the potential of data

And that’s why we’ve created the Data Science Academy

Data Science Academy

For the
BI Leadership Team

For the
BI Data Experts

For all
BI Data Users

Qualifying in the use of data and establishing a data driven culture and mindset

Our different learning formats are tailored to meet your needs

For the
BI Leadership Team

For the
BI Data Experts

For all
BI Data Users

Bootcamps with BI trainers	Flexible world class e-learning	Renowned Academic Institutions
• Basic information and upskilling		
• BI Standards and Data Science-Community Building • Upskilling including the chance to gain recognized qualifications		
• Basic introduction in Data Science and its importance for the BI strategy • Data science methods and tools you can apply to your role • Option to reskill to a data expert role		

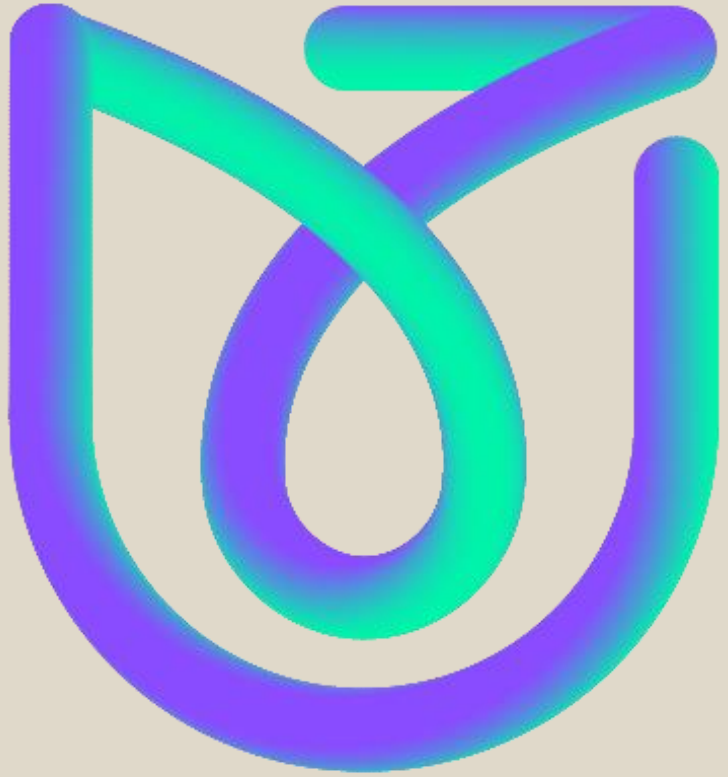
Choose the option that‘s right for you to unlock your potential with Data Science

The BI Data Science Academy will start in October 2021

The Academy will improve the **data literacy of all employees** in all levels, helping us to identify, evaluate and prioritise opportunities

It will foster agile, **cross-functional ways of working** in Data Science projects and create a culture that appreciates and understands the impact data can have

It will provide **state-of-the-art knowledge** in Data Science and data engineering to our data experts and provide them with **attractive career opportunities**



The learning program will be customized to the specific needs of BI

Therefore, your participation and considerations in this pilot are very important for us to develop an outstanding learning experience.

Thank you very much!

Data Science Academy

Data Science for Business –

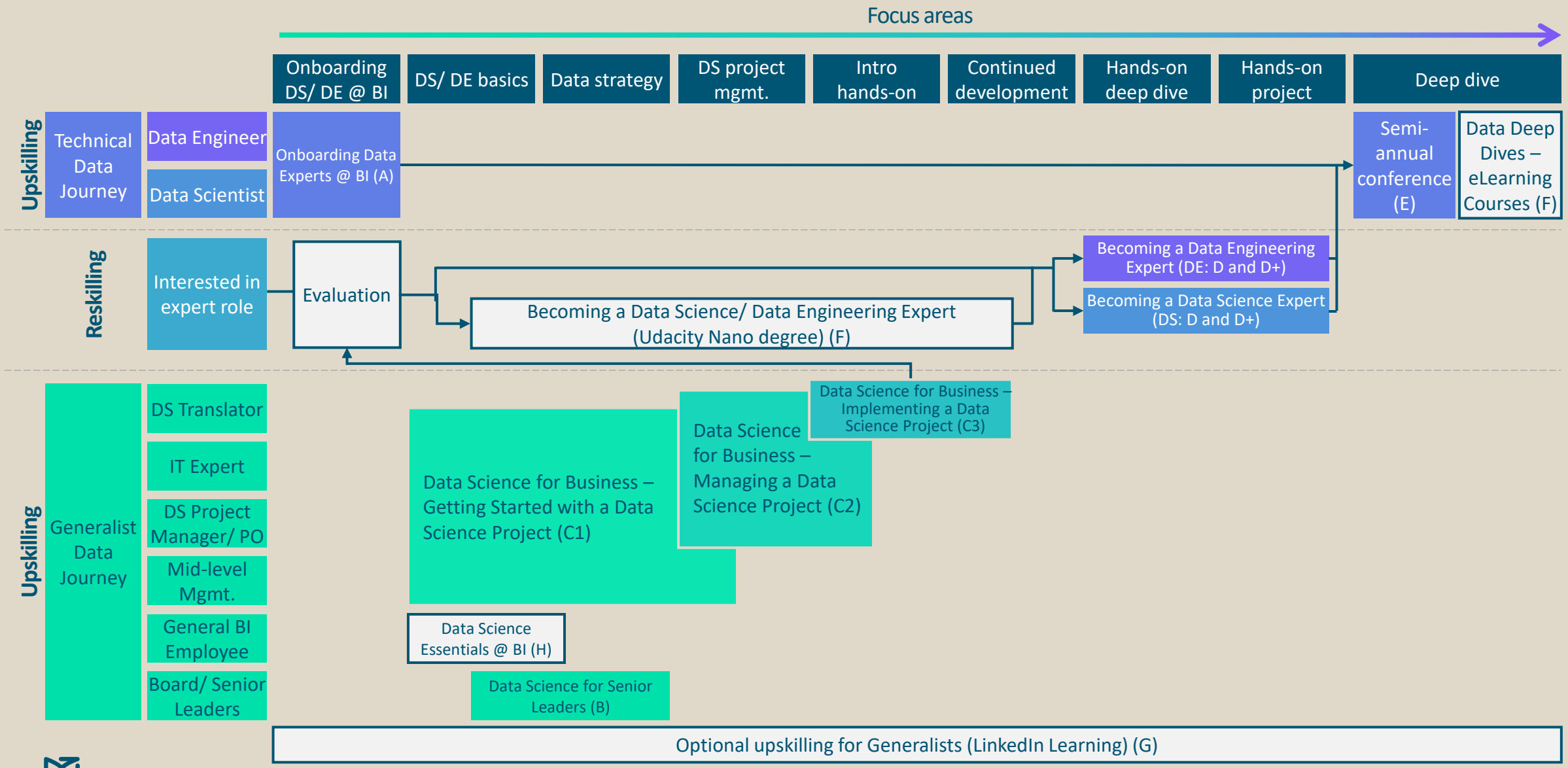
Becoming a Data Science Expert (D)

31.08.2021

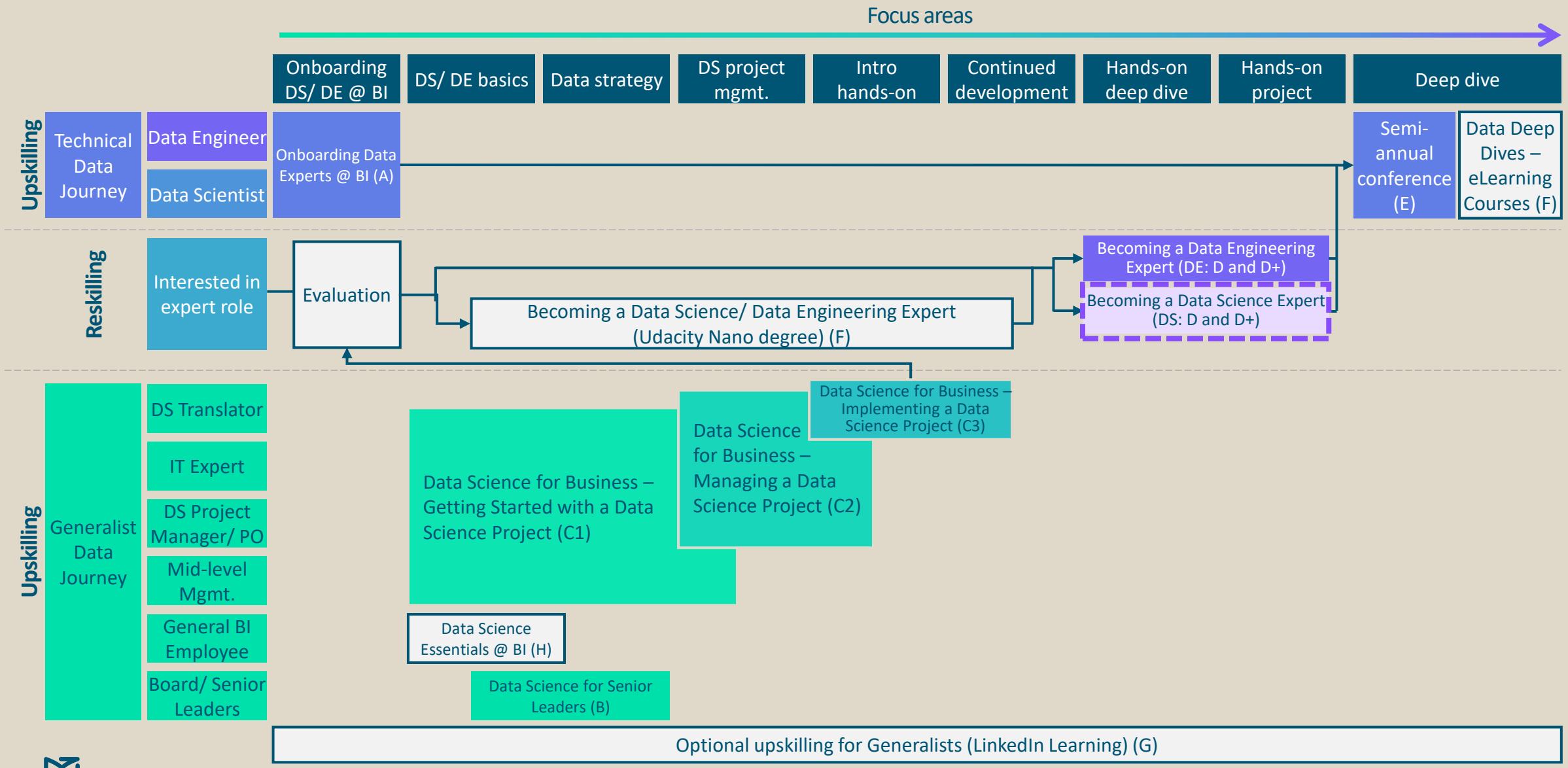
Boehringer Ingelheim

Introduction

DSA learning journeys



DSA learning journeys



Agenda

	Introduction
1	Recap Basic Machine Learning and Python
2	Complex Models
3	Model Evaluation
4	Hyperparameters
5	Unsupervised Learning
6	Gradient Descent
7	Deep Learning and Image Recognition
8	Deep Learning and Natural Language Processing
9	Repetition
10	Bias and Ethics in Machine Learning
11	Introduction to Data Science with AWS

Agenda week one

	Introduction
1	**Recap Basic Machine Learning and Python**
2	**Complex Models**
3	**Model Evaluation**
4	**Hyperparameters**
5	Unsupervised Learning
6	Gradient Descent
7	Deep Learning and Image Recognition
8	Deep Learning and Natural Language Processing
9	Repetition
10	Bias and Ethics in Machine Learning
11	Introduction to Data Science with AWS

Schedule week one

Week 1			
	Day 1 Tuesday, 31.08.2021		Day 2 Wednesday, 01.09.2021
Start: 12:00	Introduction	Start: 12:00	Recap
	1 – Recap Basic Machine Learning and Python		3 – Model Evaluation
14:00 – 15:00	Break	14:00 – 15:00	Break
	2 – Complex Models		4 – Hyperparameters
End: 18:00	Q&A and Feedback	End: 18:00	Q&A and Feedback

We will also have several short coffee breaks in between.

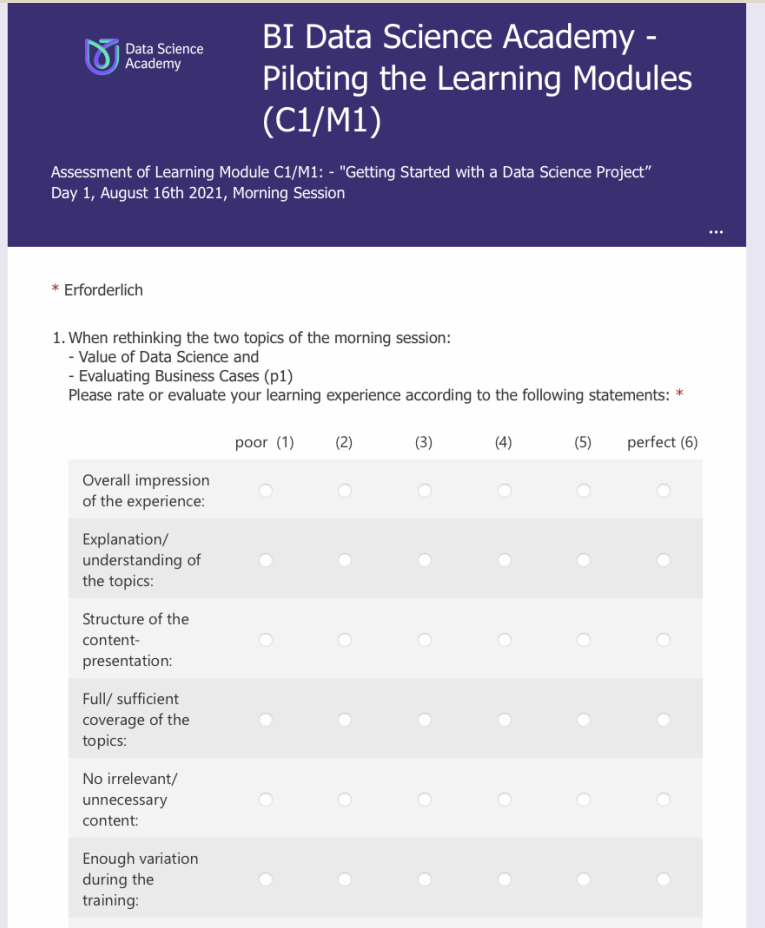
Feedback for pilot training

We aim to provide a great training experience for you and are looking forward to receiving your feedback!

You will have three different ways to give us your feedback on each training day:

1. We will have an **anonymized** feedback collection **after the last session** of each day per **Myforms.**
2. We will have an **open feedback round and discussion** at the **end of each training day.**
3. Please also **take notes** regarding your ideas during the sessions: **locally or via the Mural Board** which you can reach via LINK.

For the assessment of this pilot we prepared a survey form



Data Science Academy

BI Data Science Academy - Piloting the Learning Modules (C1/M1)

Assessment of Learning Module C1/M1: - "Getting Started with a Data Science Project"
Day 1, August 16th 2021, Morning Session

* Erforderlich

1. When rethinking the two topics of the morning session:
   - Value of Data Science and
   - Evaluating Business Cases (p1)
   Please rate or evaluate your learning experience according to the following statements: *

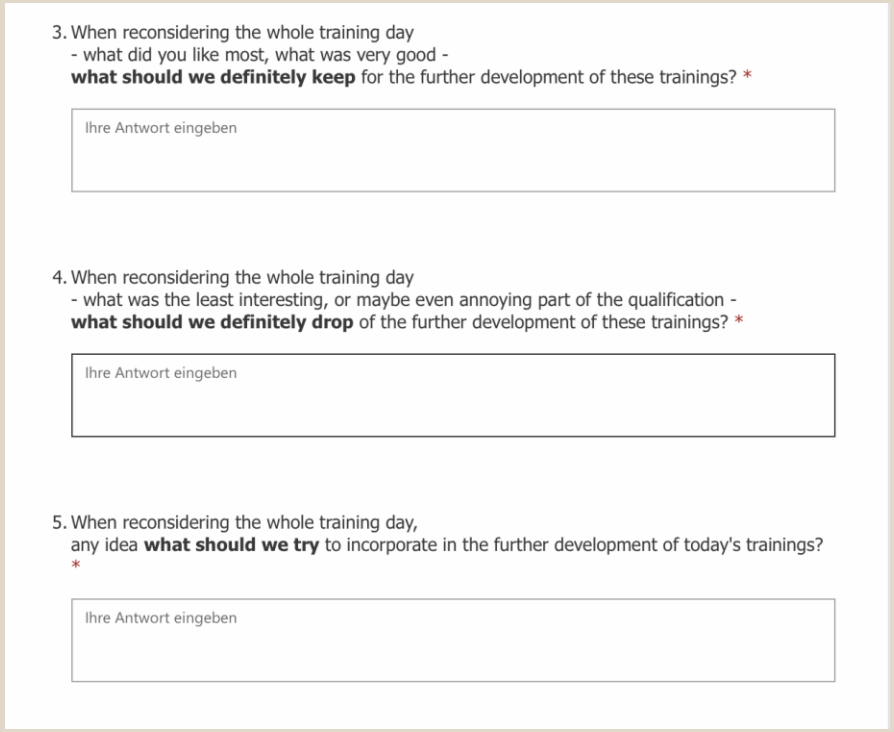
	poor (1)	(2)	(3)	(4)	(5)	perfect (6)
Overall impression of the experience:						
Explanation/ understanding of the topics:						
Structure of the content-presentation:						
Full/ sufficient coverage of the topics:						
No irrelevant/ unnecessary content:						
Enough variation during the training:						

Procedure:

- Prior to the end of the training you will receive a Link from “Wendelin Mueller” leading you to an online Myforms survey.
- Please check your spam box to assure that you got the link.
- The survey consists of a couple of rating scales where you can spontaneously tick whether the pilot session gave you a “perfect” or a “poor“ experience.
- You have to rate several aspects of the session (under-standing, structure, appropriateness, duration, …) that will allow us to further improve the trainings.
- Only instruction you need: “The better your experience, the more on the right side of the scales you may tick.”
- You will also have the chance to leave a written feedback if you want to.

For a convenient procedure, you will receive an e-mail with a link to an online survey.

Please share you experience also in a qualitative manner

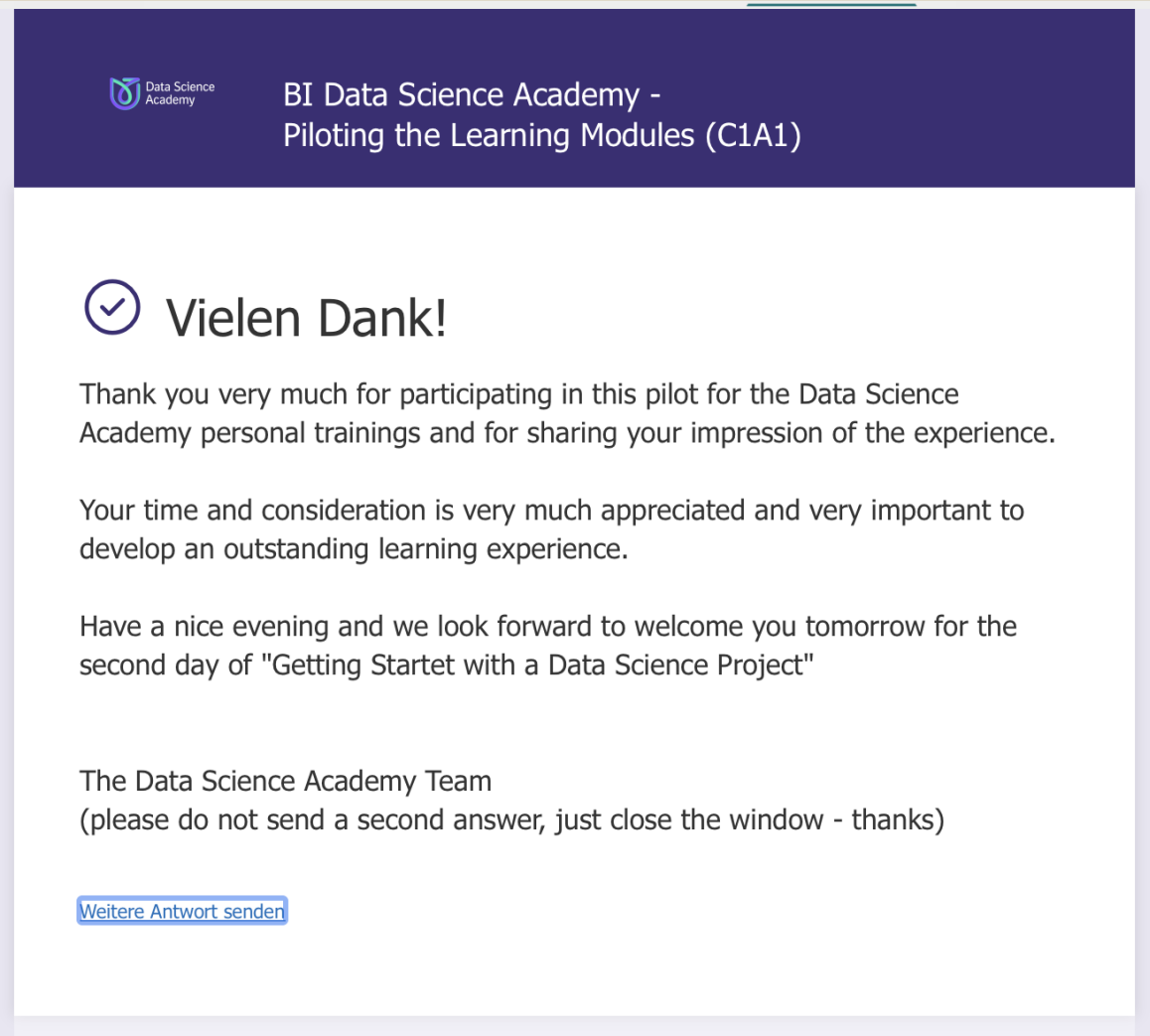


Procedure:

- In addition, the survey link for the afternoon session will include three “keep, drop, try” questions where we ask for your explicit recommendations for the improvement of the learning experience.
- You may take some notes during the day to have enough “food” for these recommendations, as they are very important hints for us to improve the program, the contents, the material and the way we present and include you into these trainings.

To optimize the training for you and your colleagues, please try to provide specific feedback!

Once followed the link, everything is self explaining



Let‘s introduce us to each other

Please introduce yourself.

- Name
- Job role and department
- What expectations do you have for this training
- Rate your experience in the field of Data Science:

I’m just **beginning to learn** about Data Science.

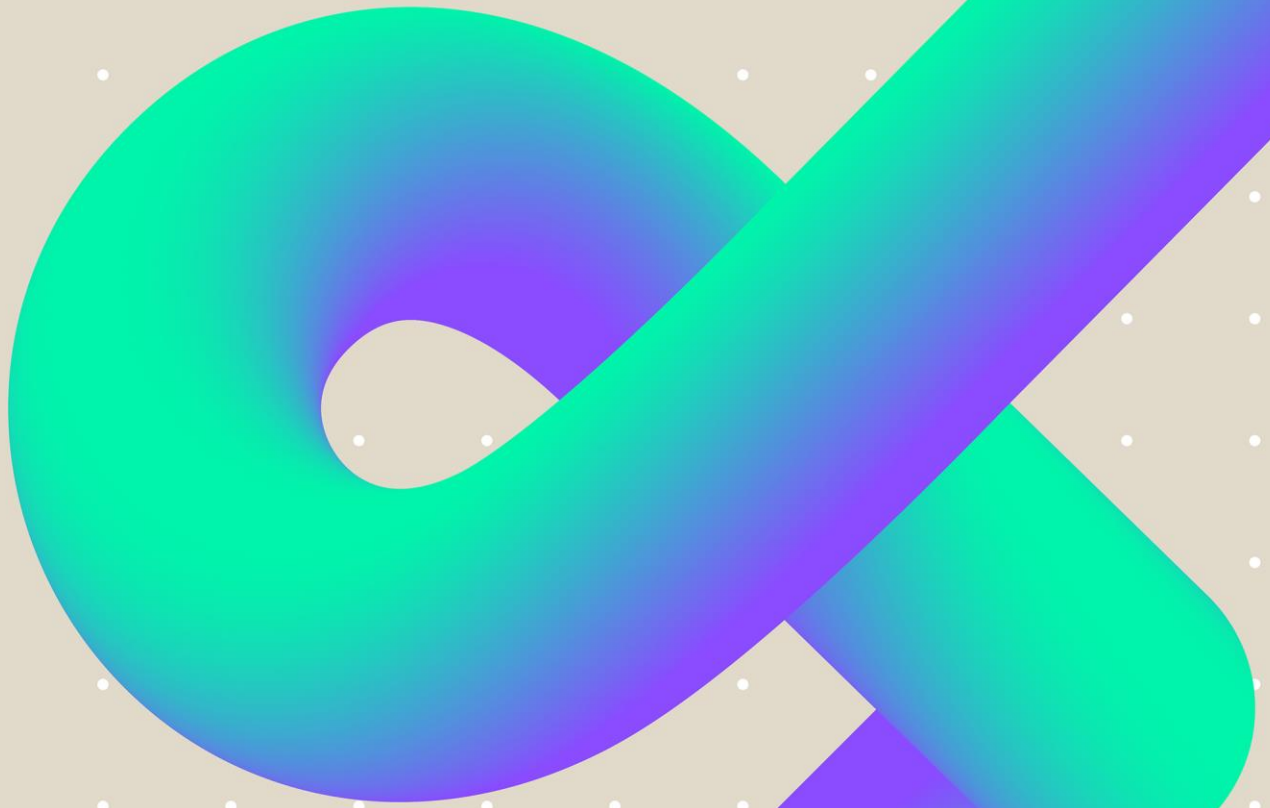
I have **extensive knowledge** about Data Science and know where and how to apply it.

You will have 2 minute per person.

Then please name the next person in the meeting to introduce him- or herself.

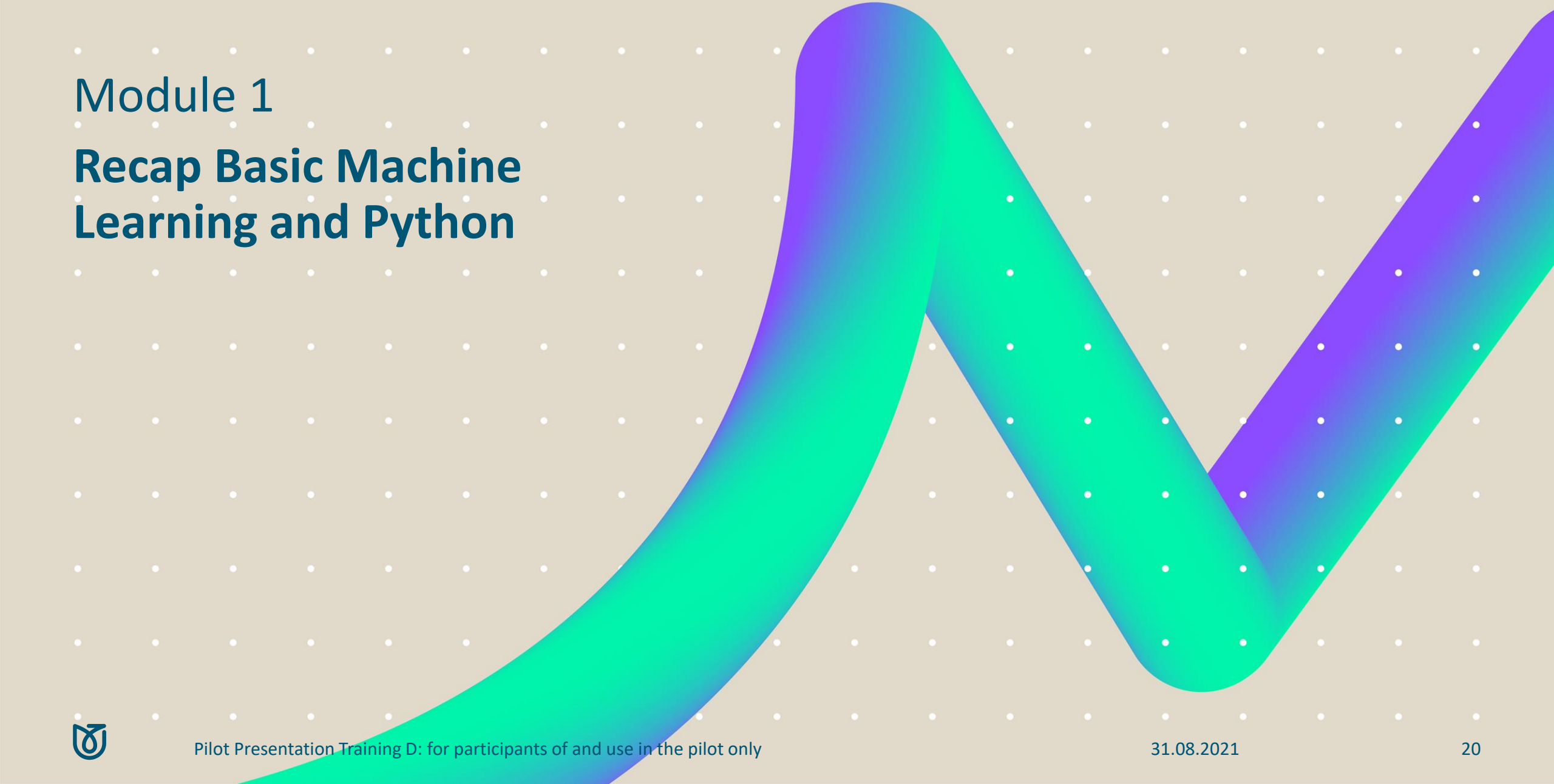
Welcome to the Data Science Academy!

Access to AWS



Module 1

Recap Basic Machine Learning and Python



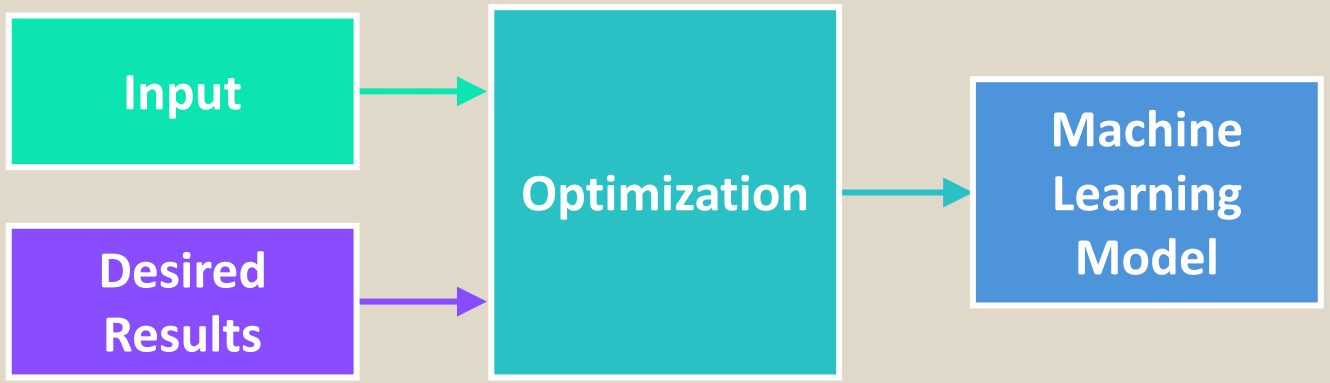
Agenda

	Introduction
1	**Recap Basic Machine Learning and Python**
2	Complex Models
3	Model Evaluation
4	Hyperparameters
5	Unsupervised Learning
6	Gradient Descent
7	Deep Learning and Image Recognition
8	Deep Learning and Natural Language Processing
9	Repetition
10	Bias and Ethics in Machine Learning
11	Introduction to Data Science with AWS

How does Machine Learning work?

Predictors/ Features | **Target Variable**

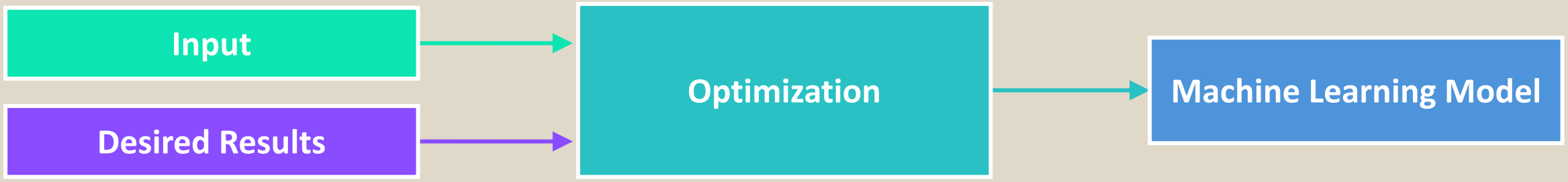
X	Y	Z	A
1	1	3	4
3	7	9	6
5	2	4	5



**Bigger dataset expected in reality*
**Example shown is a supervised learning task*

Machine Learning finds generalizable predictive patterns.

Keys for a successful Machine Learning model

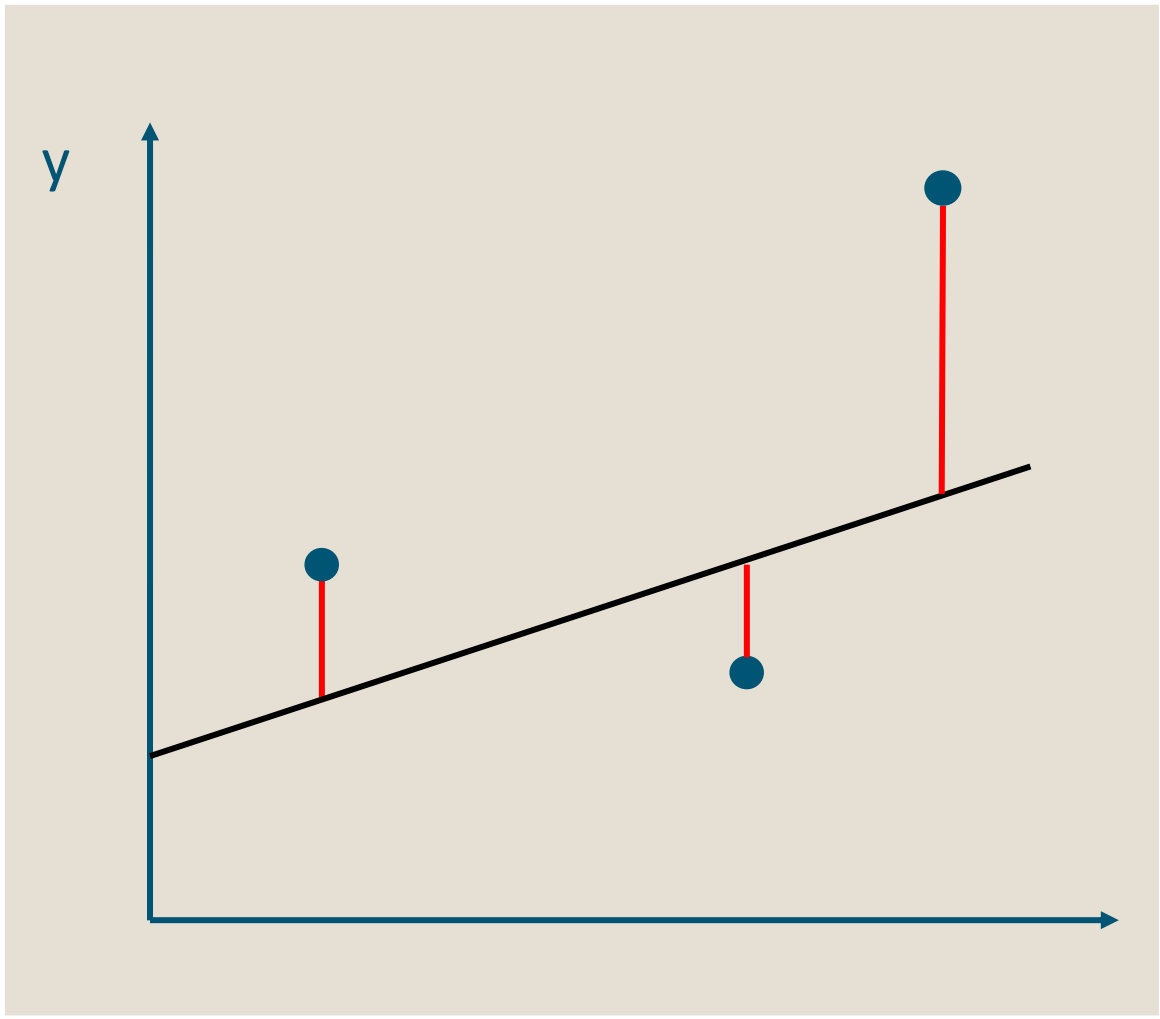


- Data quality
- Enough and relevant predictors
- Making sure data are balanced (for classification)

- Models selection
- Calculation time
- Scaling problem
- Considering ethics and bias

- Interpretability vs. black box
- Performance metrics

Recap: Linear Regression



- Linear regression describes the **linear relationship** between the **dependent variable y** and the **independent variable x**

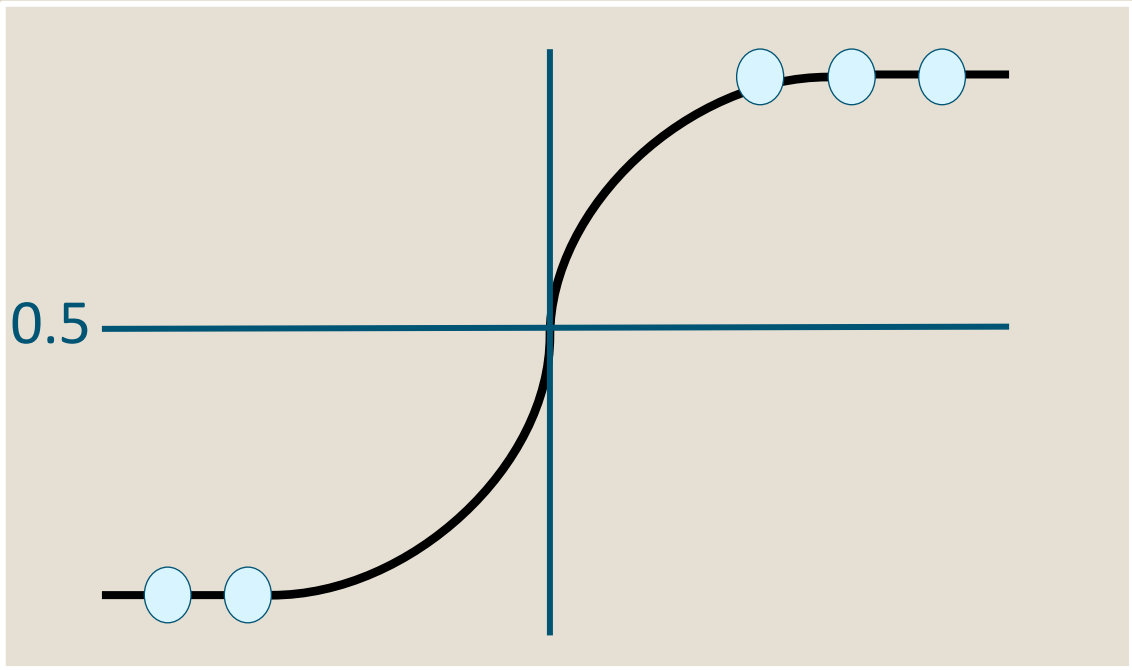
$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ..$$

- Goal: Find a set of β_i that **minimizes the error** between **actual value y** and the **predicted value $\hat{y}$**

$$e = \sum(y_i - \hat{y}_i)^2$$

- The error is squared: **large deviations** produce disproportionally **large error values**, while **small errors are more tolerable**
- A model is never perfect: there will be some **unexplained error ϵ** left
- We can **predict** the unknown value of a **new point** by using the **fitted equation** above

Recap: Logistic Regression



- A **classification** can be modeled as a **regression** where the **class is encoded in the dependent variable as 0 or 1**

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ..$$

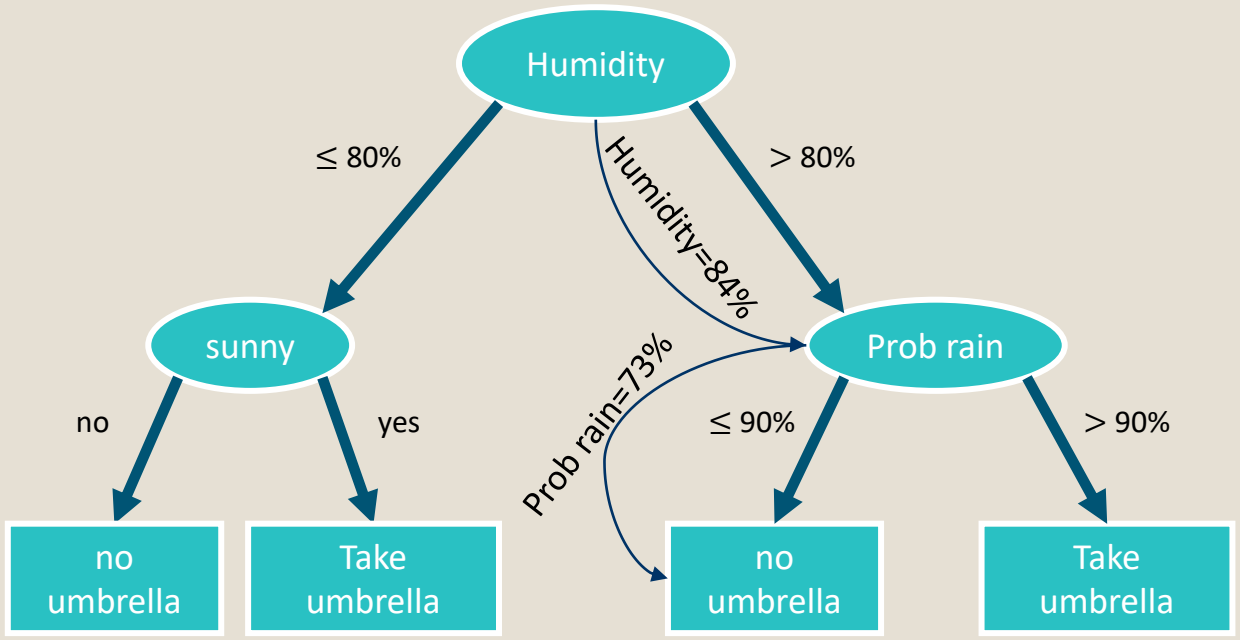
- Since we want to model only values in between 0 and 1 we define our class c as the output of a **logistic function** that scales the values into the desired range

$$\hat{c} = {}^{1}/_{1 + e^{-\hat{y}}}$$

- To find the **correct $\boldsymbol{\beta_i}$** minimize **the sum of squared errors**

Recap: Decision Tree

New sample:
Day = sunny, humidity = 84%, Prob rain = 73%



New sample:
According to our decision rules, which shouldn‘t pack an umbrella

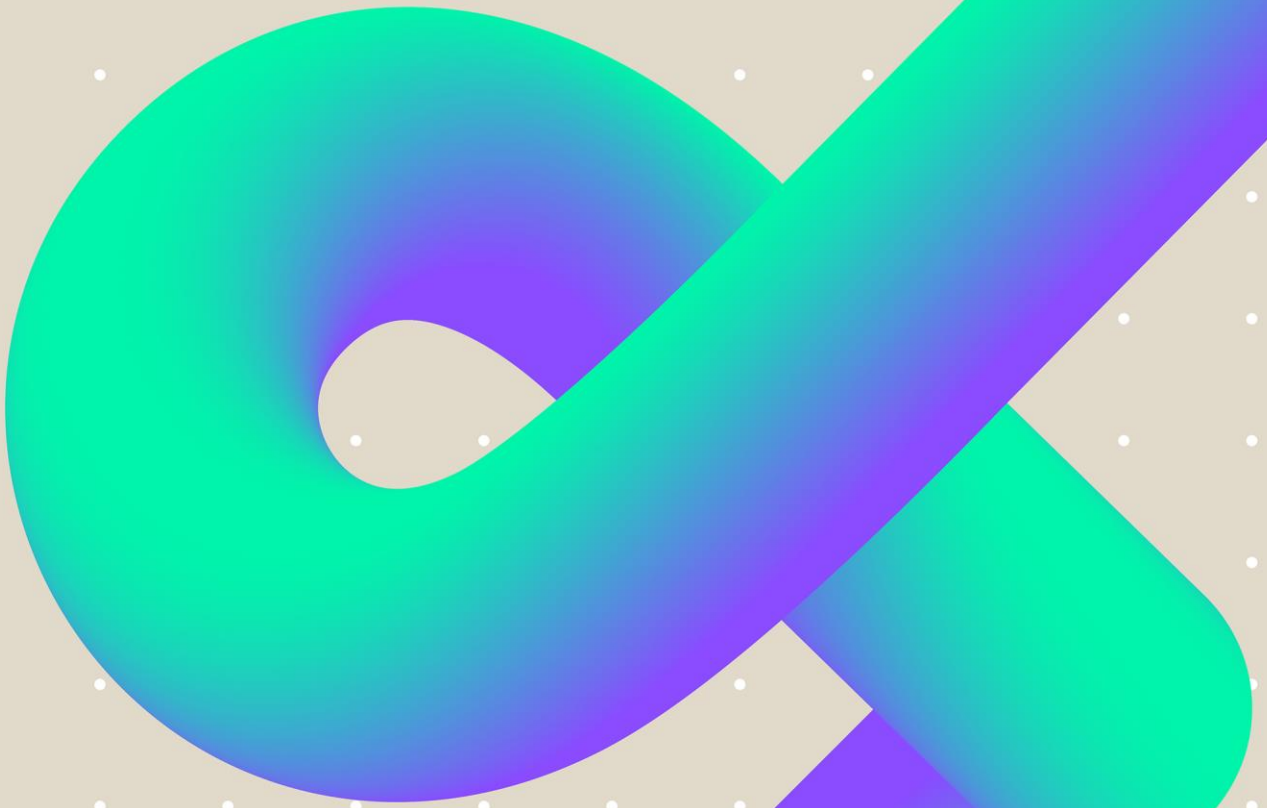
- **Split the data based on the feature that results in the largest information gain (IG)**

$$IG(D_p, f) = I(D_p) - \frac{N_{left}}{N_p} I(D_{left}) - \frac{N_{right}}{N_p} I(D_{right})$$

  - f is the feature
  - D the dataset
  - D_p the dataset at the parent node
  - $D_{left/right}$ the dataset at the child node when split by f
  - I is the impurity measure
  - N_i is the number of samples at the node i
- **The information gain is the difference between the impurity of the parent and the sum of impurity of both child nodes**
- **The final node in a tree contains the label information**

Try it yourself!

In the following exercises

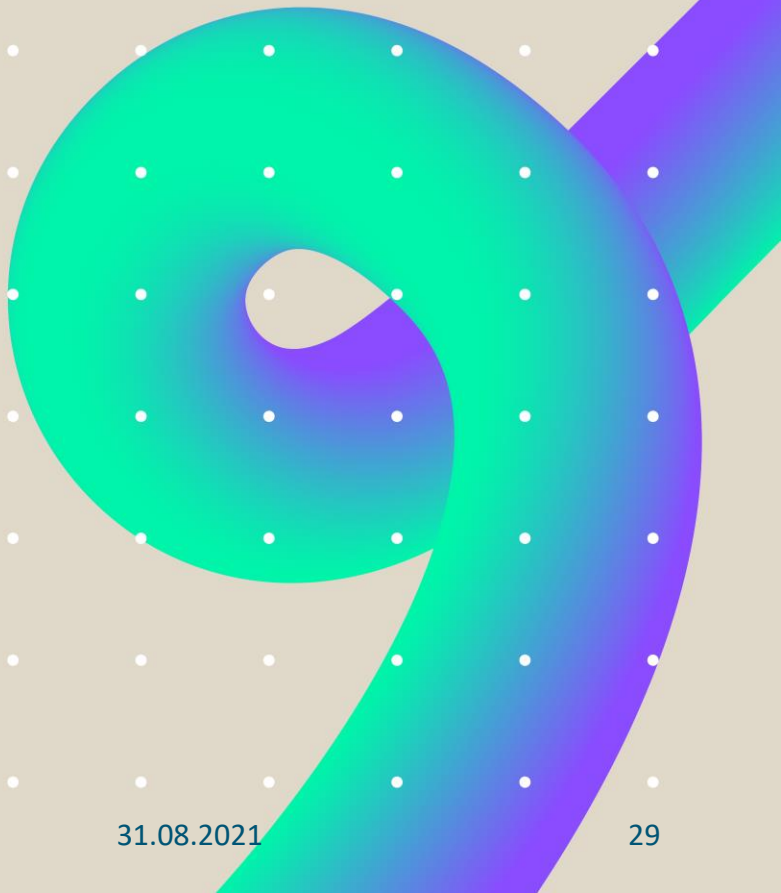


Module 2

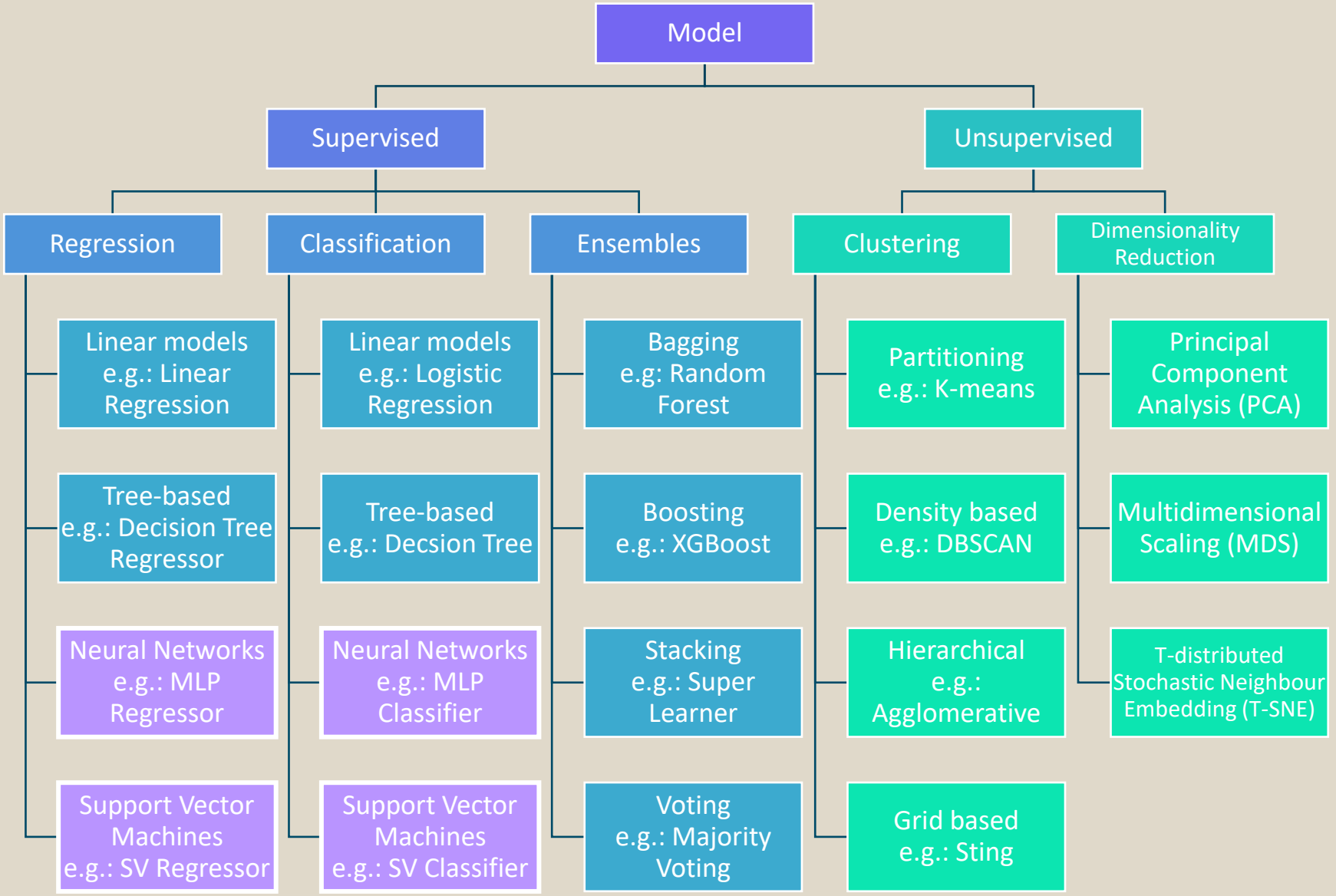
Complex Models

Agenda

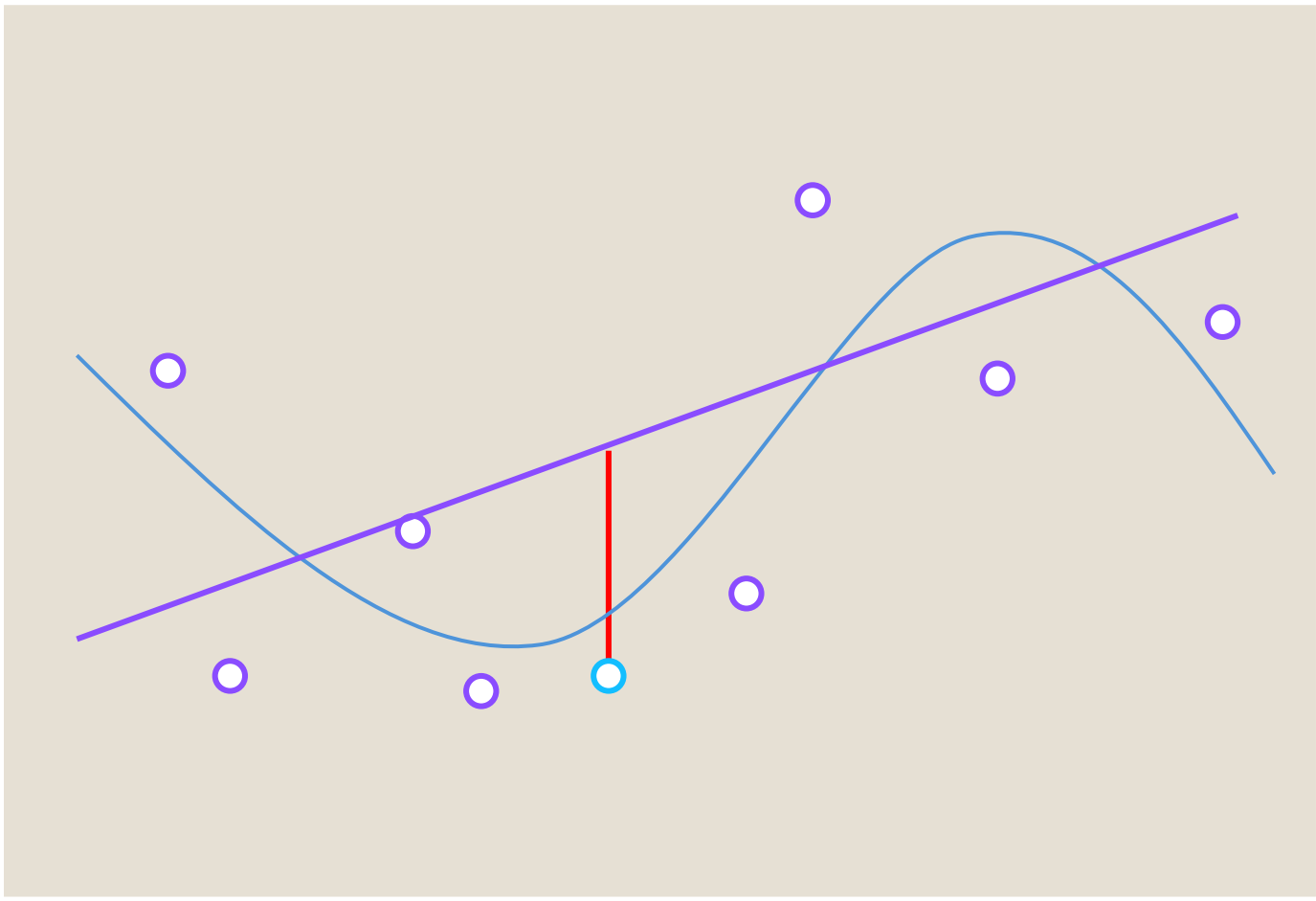
	Introduction
1	Recap Basic Machine Learning and Python
2	**Complex Models**
3	Model Evaluation
4	Hyperparameters
5	Unsupervised Learning
6	Gradient Descent
7	Deep Learning and Image Recognition
8	Deep Learning and Natural Language Processing
9	Repetition
10	Bias and Ethics in Machine Learning
11	Introduction to Data Science with AWS

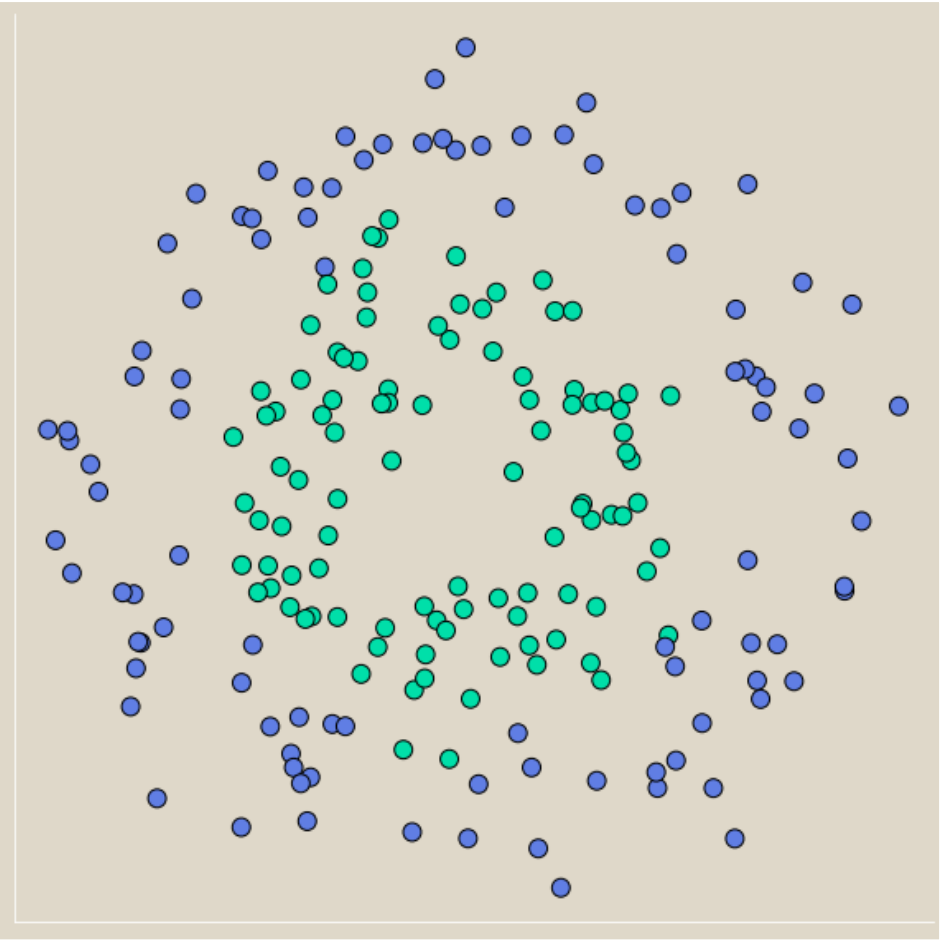


Selected model types



Why do we need Complex Models?





We need complex models to solve non-linear problems

Why aren’t we always using complex models

Linear models

Strengths

- Fast
- Need of less data
- Less prone to overfitting
- More interpretable

Weaknesses

- Can only fit simple linear problems
- Prone to underfitting

Complex models

Strengths

- Can fit non-linear problems
- Less prone to underfitting

Weaknesses

- Prone to overfitting
- Difficult to interpret

Polynomial Regression

Transition between simple and complex models

Allow linear regression to solve non-linear problems

1. From the original dataset, create "new Features" $x^{3,}x$^2
2. Do ordinary linear regression with that improved dataset

Strengths

- Can solve non-linear problems

Weaknesses

- The choice of the number of polynomials is important
- Can easily overfit

Example – Polynomial Regression

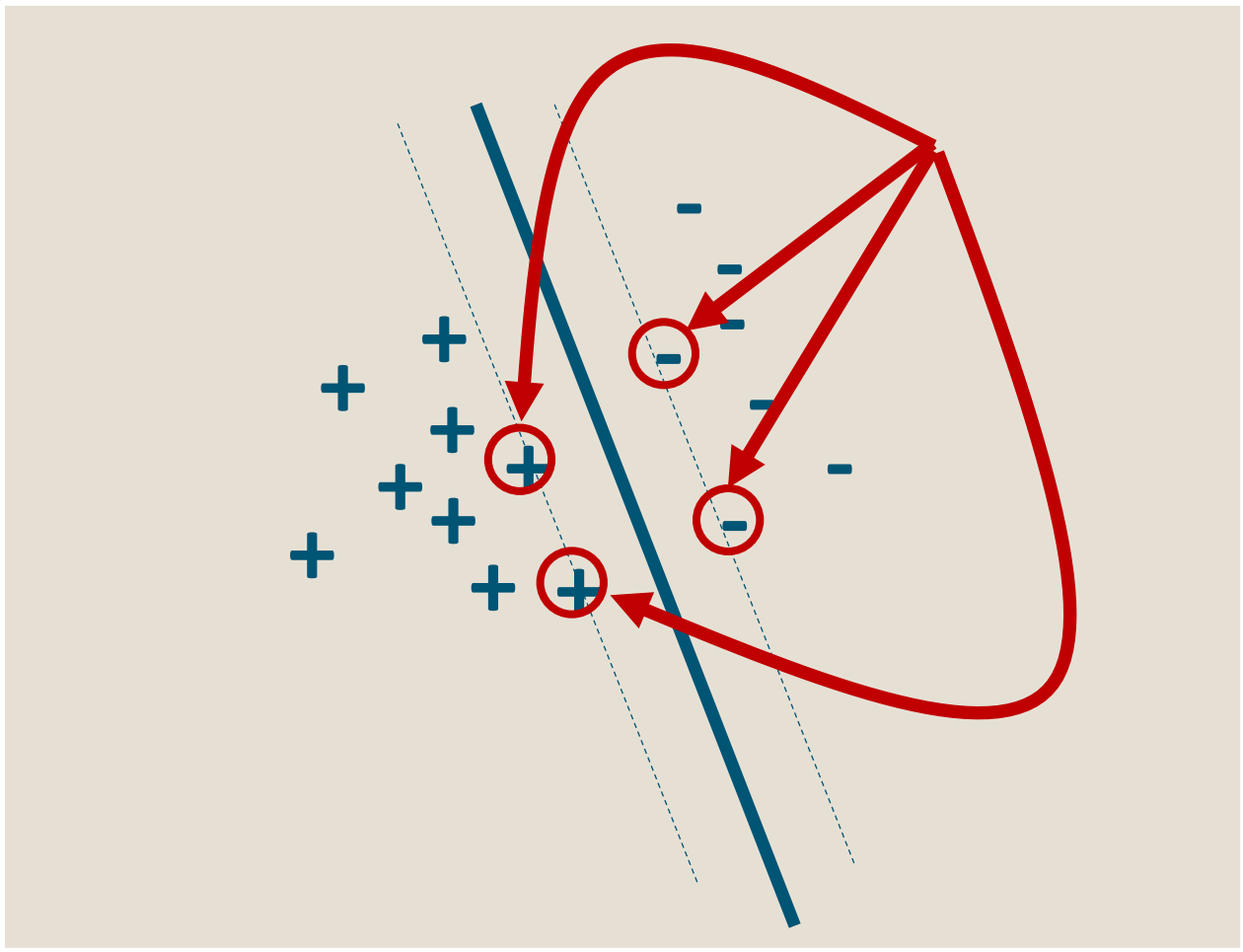
X	Y
1	1.2
2	3.9
3	9.1
…	…

X	X^2	X^3	Y
1	1	1	1.2
2	4	8	3.9
3	9	27	9.1
…			…

$$f(x) = dx^3 + cx^2 + bx + a$$

Support Vector Machines

SVM = Support Vector Machine



SVM combine two geometric ideas

- Widest street gives ‘most stable’ linear separator
- If data is not linearly separable, push everything into a much more complex space and then solve linearly in that complex space

Why is it called SVM?

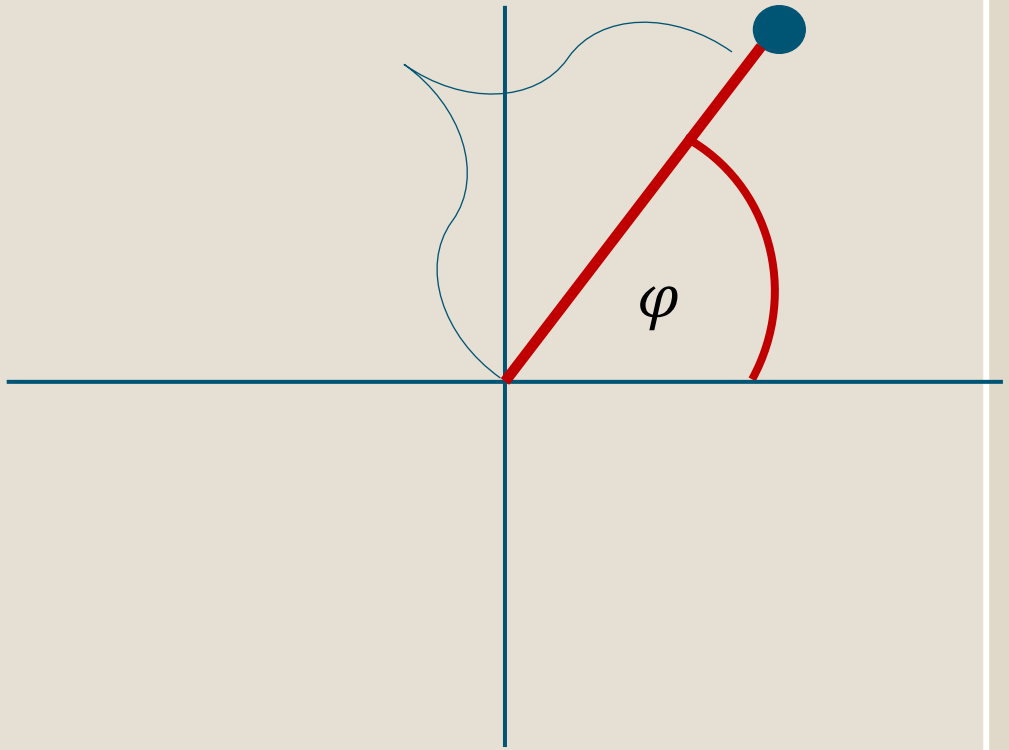
Support Vectors: They support (bound) the street. Only because of them the street cannot get any wider…

Transformation into different space

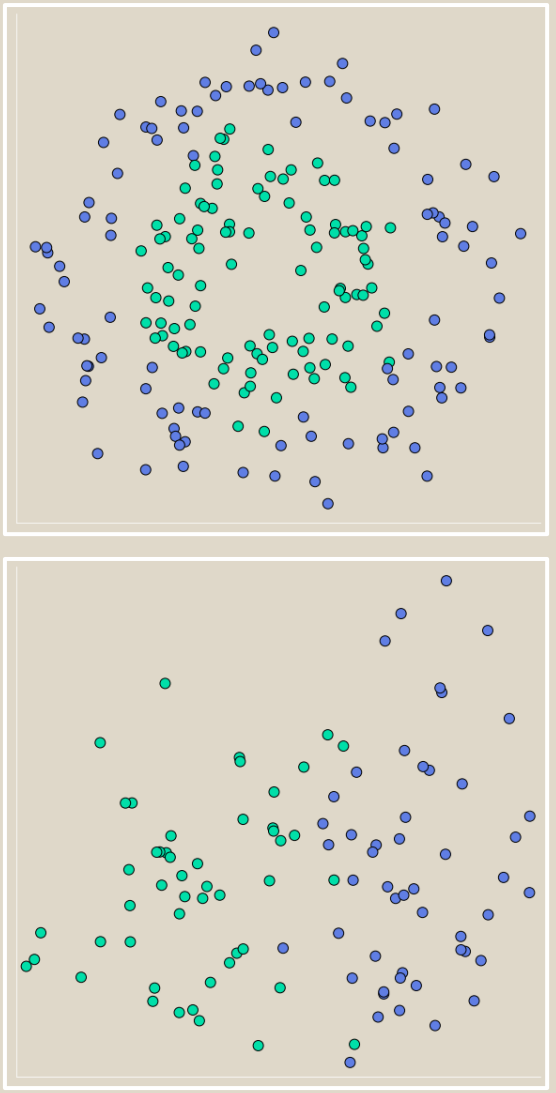
If data not linearly separable then push to other space

Example: Transform to polar coordinates

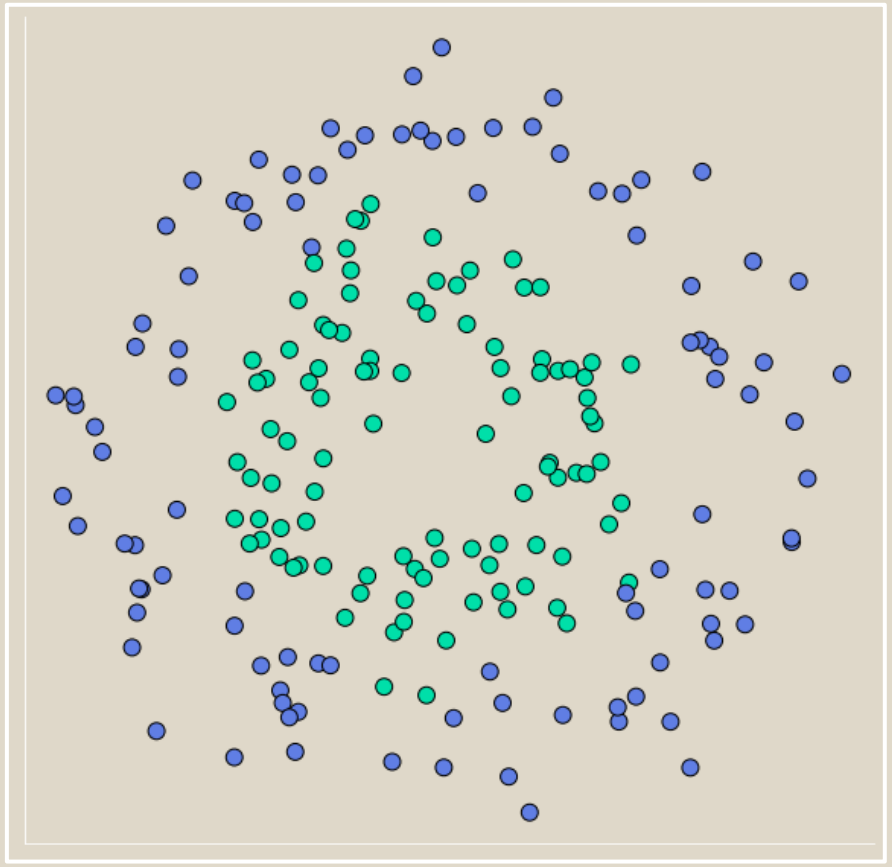
Every point in x-y-space can be considered in radius-angle-space

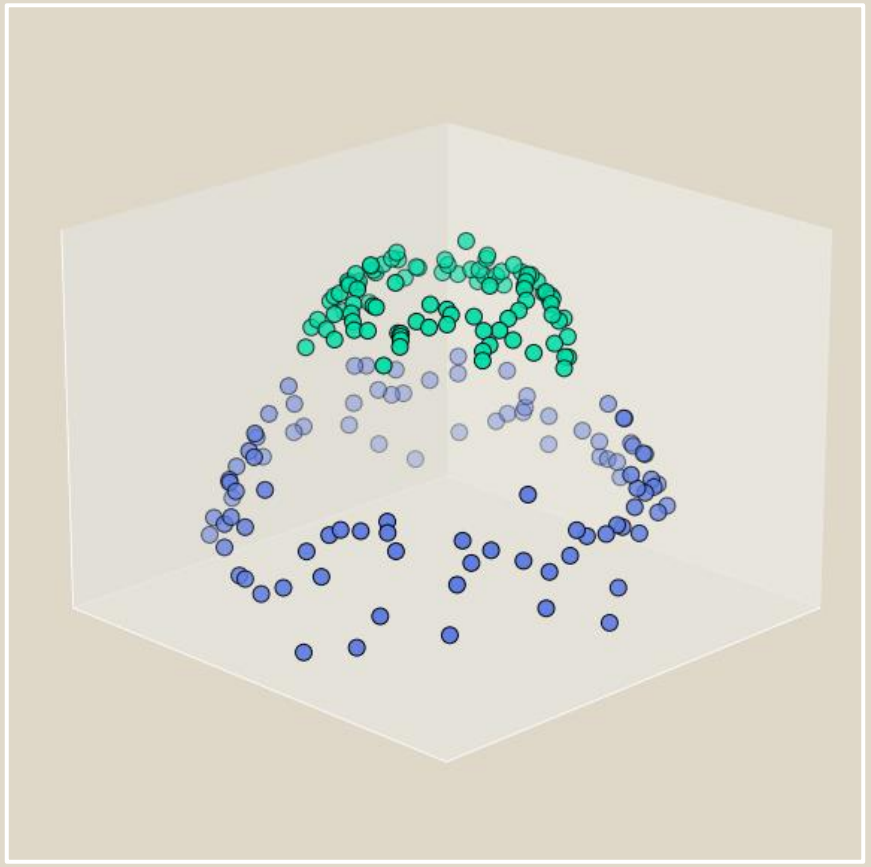


$$(x, y) \rightarrow (r, \varphi)$$



Example: Transform to higher dimension





$$(x, y) \rightarrow (x, y, x^2 + y^2)$$

RBF kernel

- The RBF kernel transforms from the target space into a new space with all monomials of the input space
  - This is called **'kernel trick'**
- In this space (l^2) the SVM solves linearly
- Example: For two dimensions (x,y) the RBF kernel corresponds to a push

$$(x, y) \rightarrow (1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3, \dots)$$

Strengths

- Effective in high dimensional spaces
- Needs only a small amount of data (support vectors)

Weaknesses

- This technique is slow for big data cases

Single Layer Perceptron: Neuron

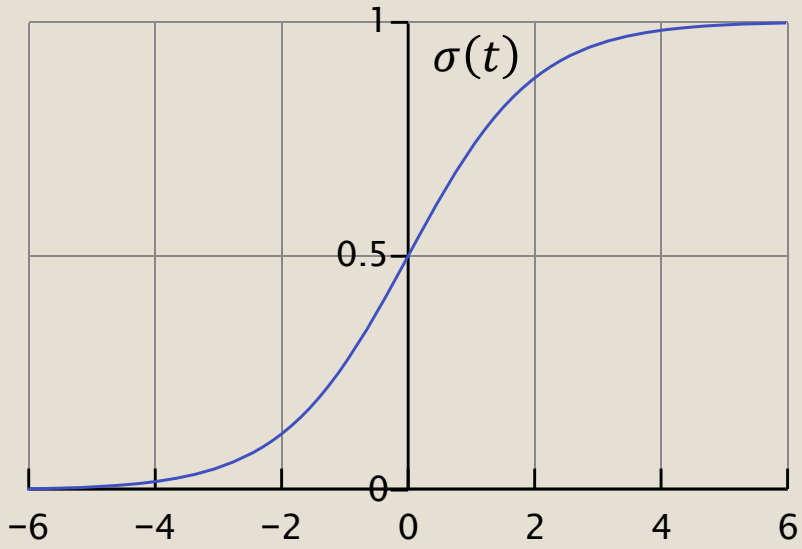
- A single neuron is nothing but a logistic regression
- Logistic regression prediction function was

$$(x, y) \to \sigma(cx + by + a)$$

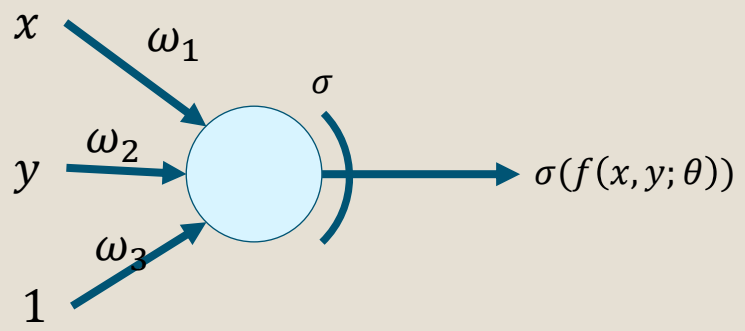
- i.e. $\sigma\big(f(x, y; \theta)\big)$, here,

$$\sigma(t) = \frac{1}{1 + e^{-t}}$$

- The term σ is called activation function

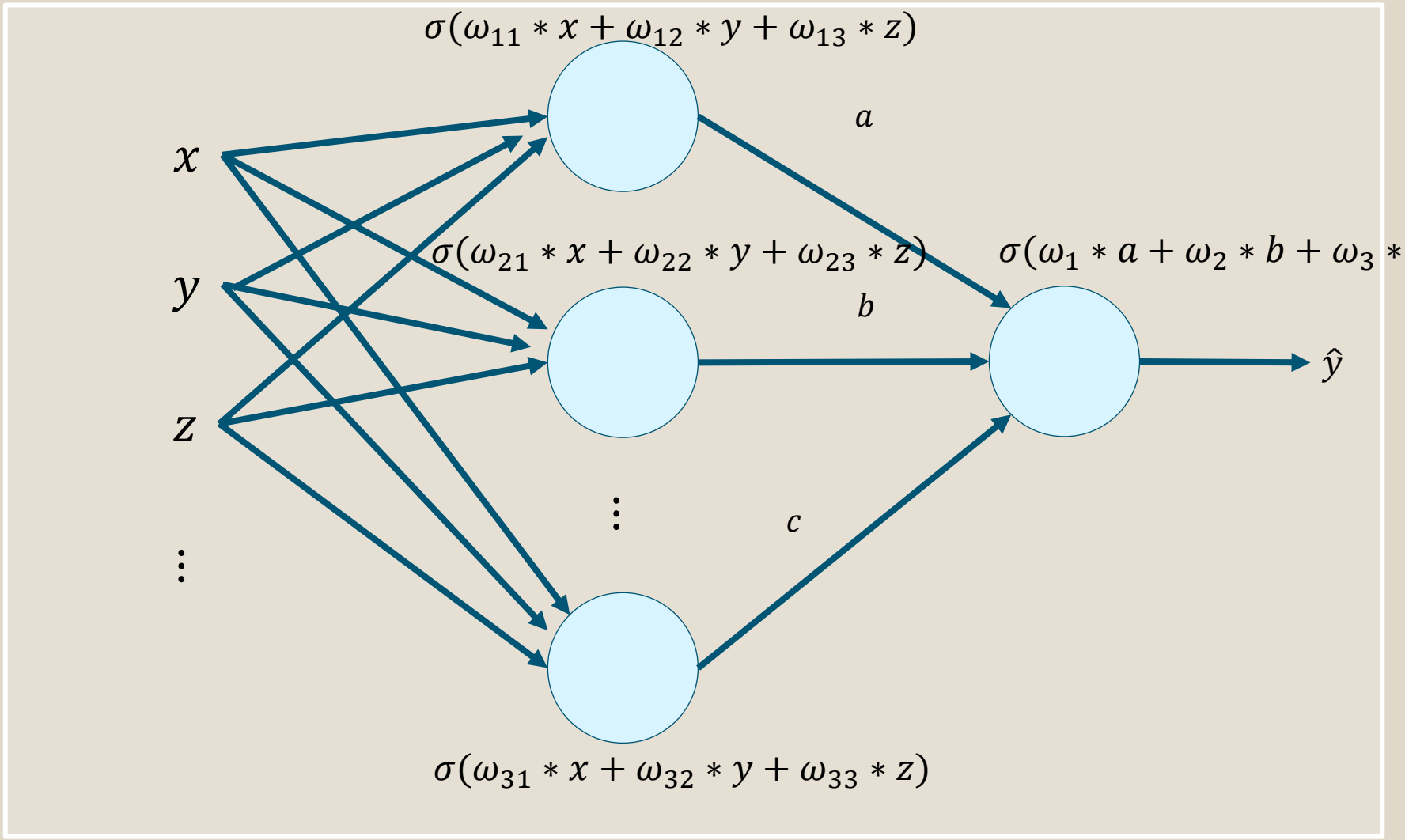


We can represent the logistic regression in this form



Multi-layer Perceptron: Neural Networks

- Combining multiple neurons into layers and fully connecting the layers leads to a Neural Network
- A neural network is nothing more than multiple logistic regressions stacked on top of each other
- Each input into a neuron is multiplied by a weight ω_{ij}
- The goal is to find the weights that minimize our training error



Deep learning explained simply

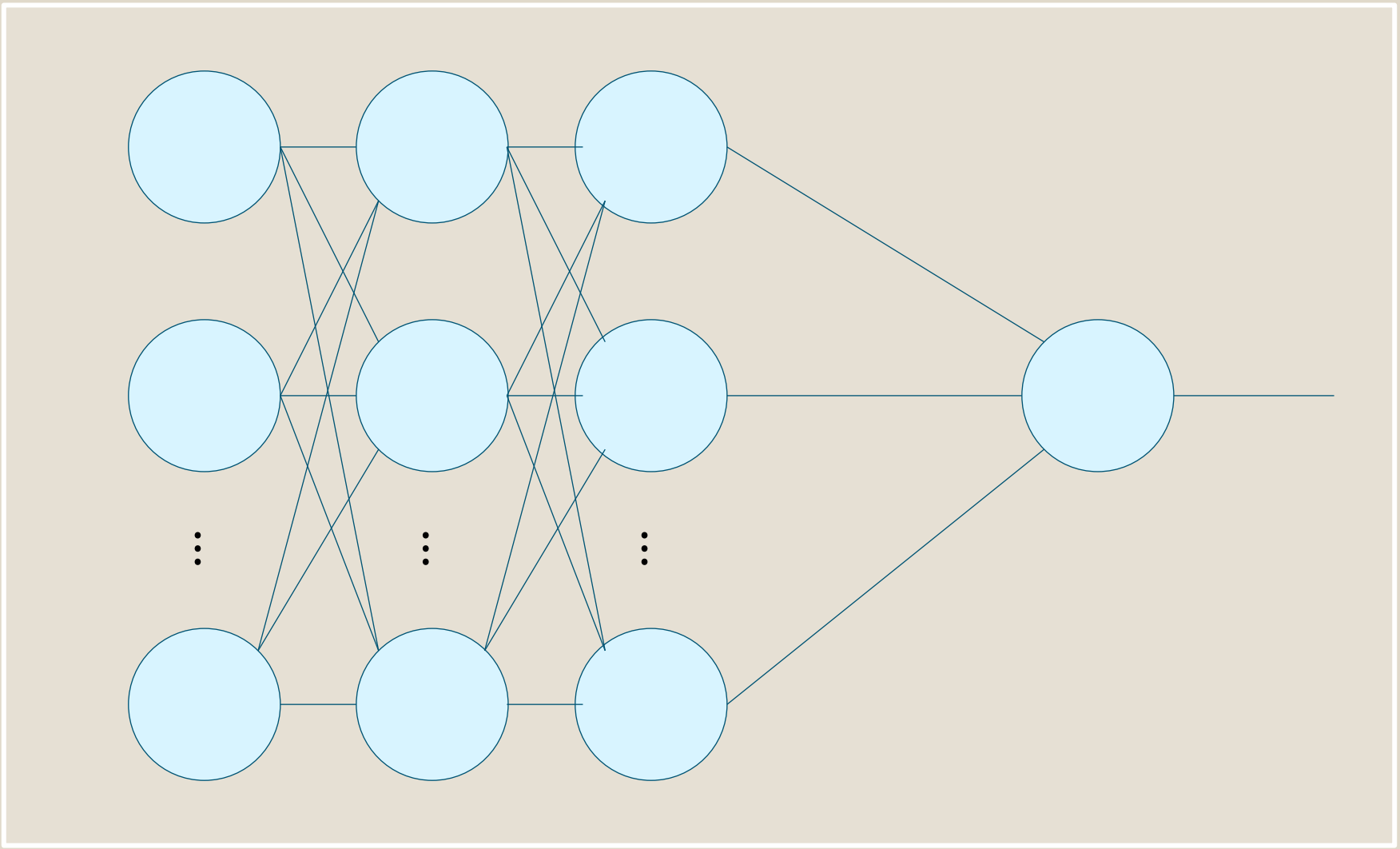
- By repeatedly stacking layers on top of each other we create so called Deep Neural Networks

Strengths

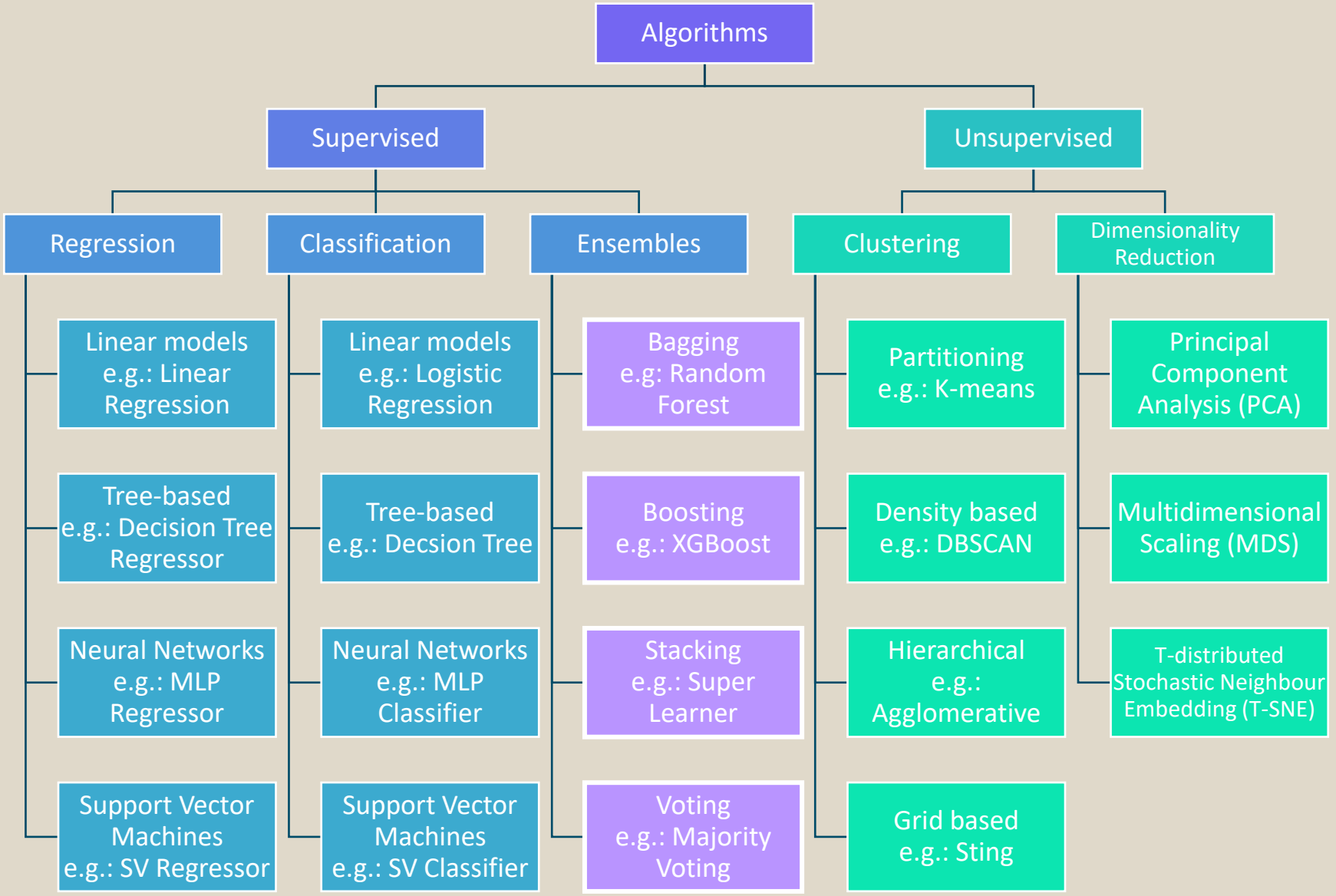
- With increased depth/complexity neural networks can learn increasingly hard problems
- Can learn from large amounts of data

Weaknesses

- Difficult to train right
- Computationally expensive training

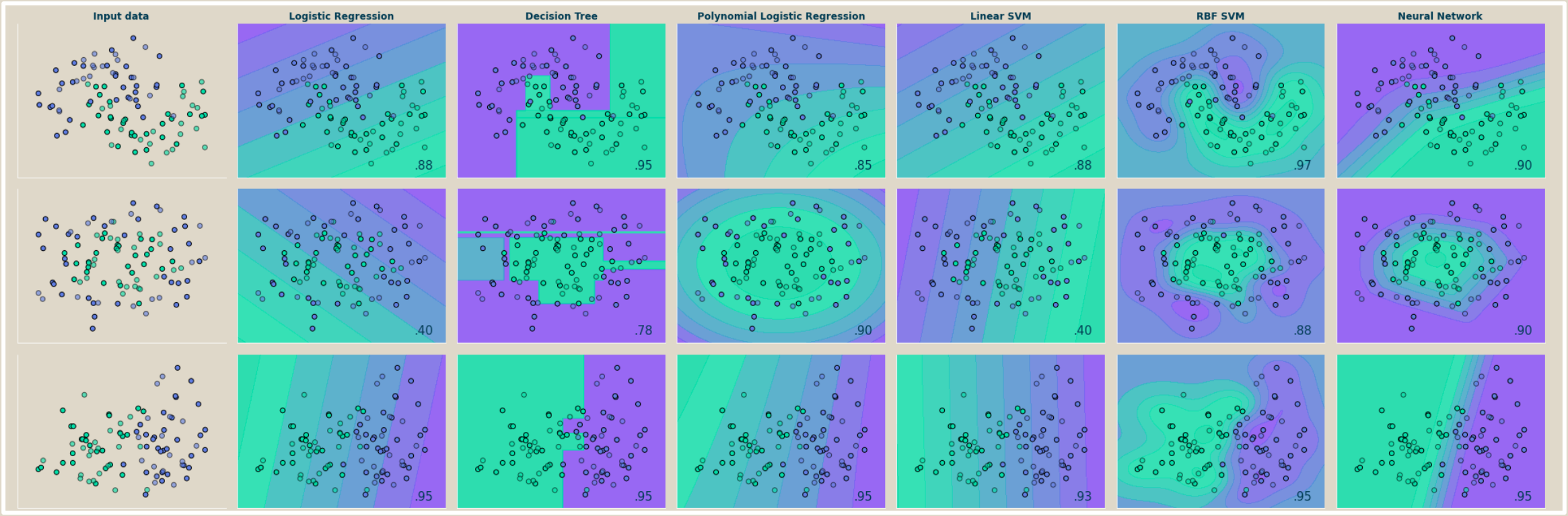


Selected model types



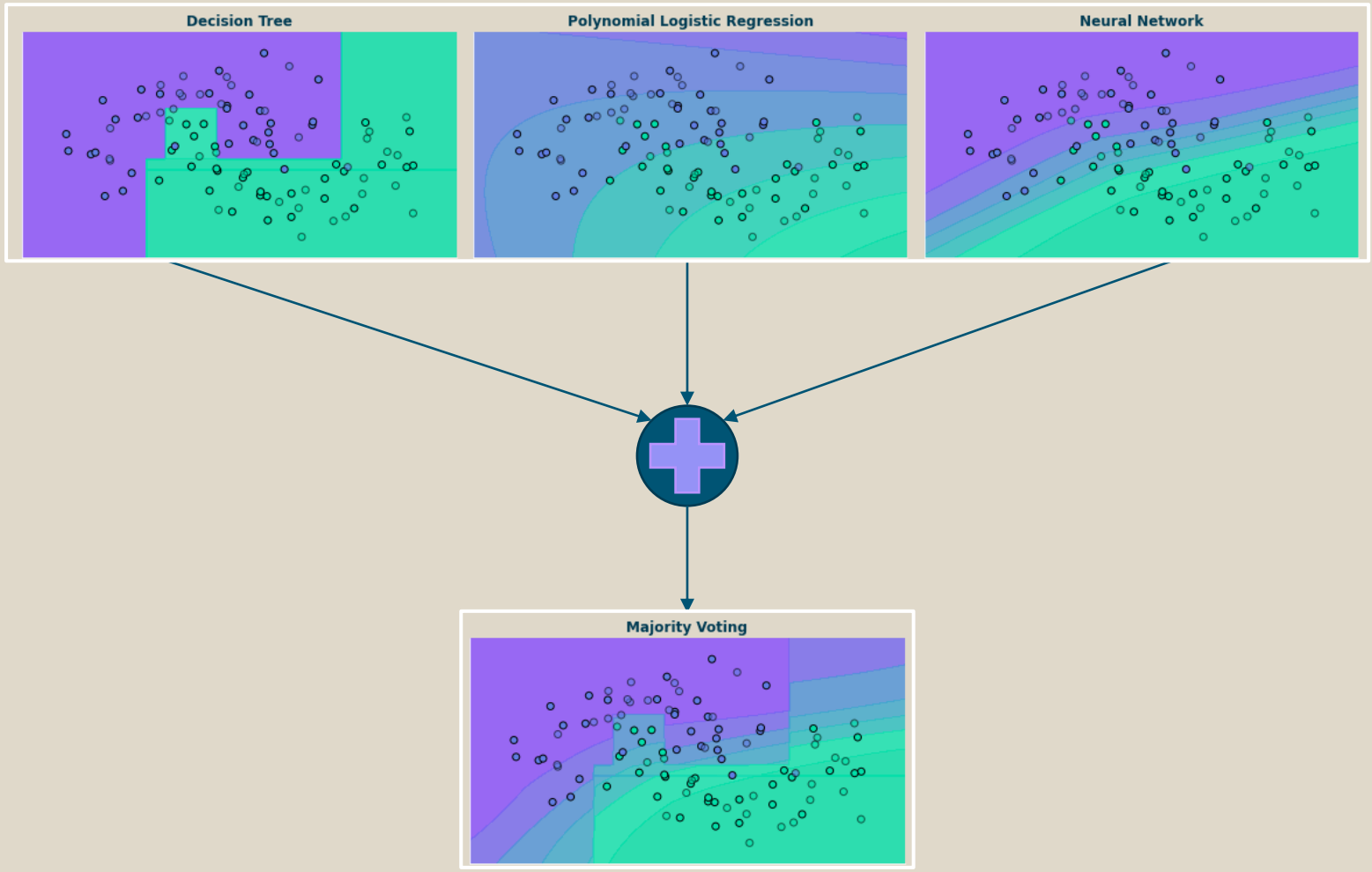
Model overview

- Each model describes the data a bit different
- Each model has certain advantages and disadvantages
- We can combine multiple models to create a better and more complex model



Ensembles: Combining the views of multiple models

- **Combine the learners to get a more stable model**
- **Methods of combination are**
  - Majority voting
  - Weighted voting
  - Stacking
- **The combined models should be as uncorrelated as possible**
  - Highly correlated models predict the same way. Adding more of them does not add information



Systematically building Ensembles: Bagging

Correlated Model

- We want to generate uncorrelated models
- Otherwise, they will all predict the same thing

Bagging

- Use Bagging (Bootstrap Sampling)
- Each model is trained on a subset of the data
  - Subset: A part of the samples and a selection of the features.

Training

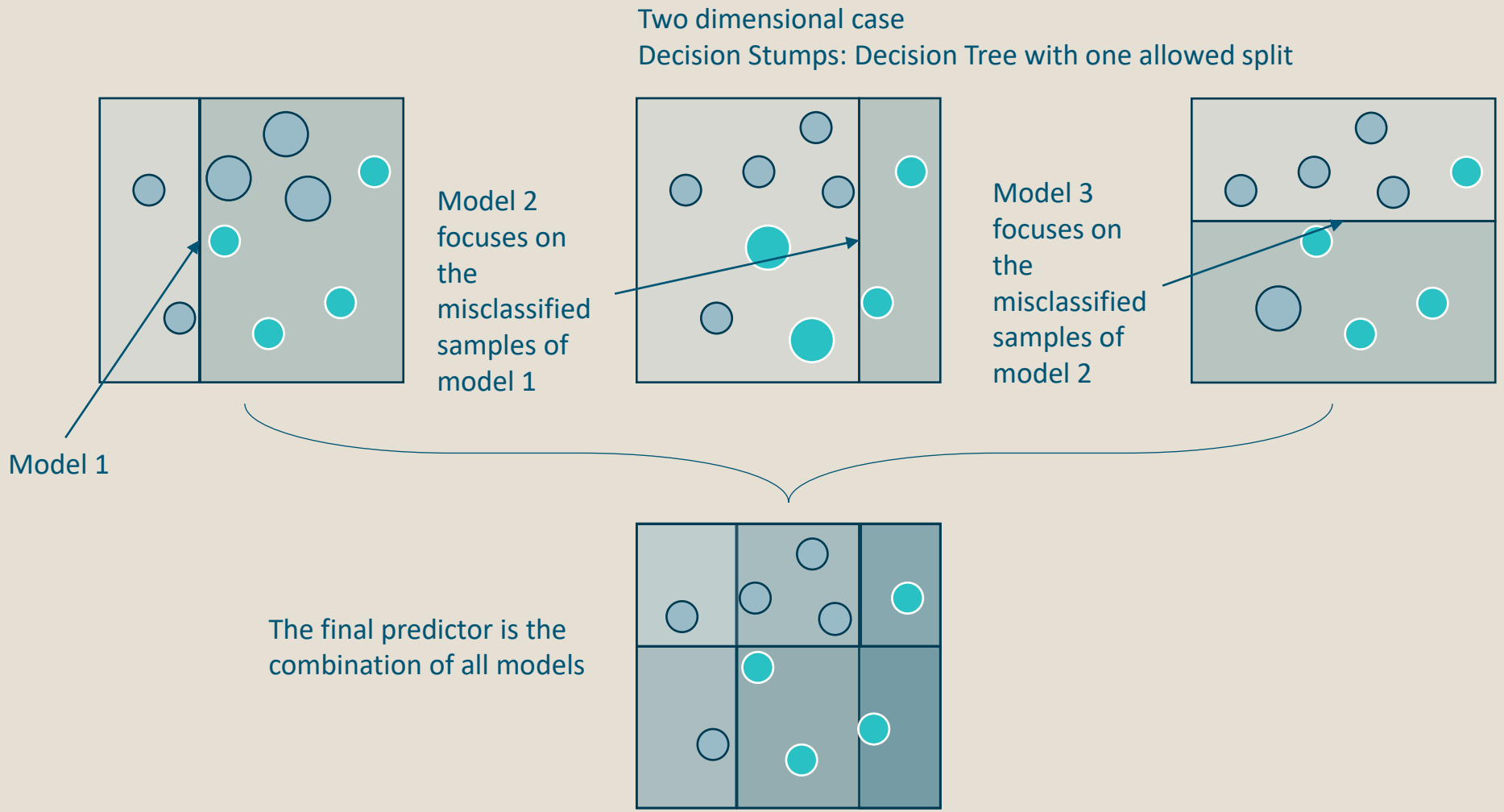
- Each model is trained with another view on the data
- Each model specializes on a specific part of the data

Predicting

- Each model in the ensemble is doing its own prediction
- All predictions are averaged to get the final prediction

- A well-known example of bagging is the **Random Forest**
  - Where each tree is trained on a subset of data
  - For the final prediction, the predictions of all models are averaged

Systematically building Ensembles: Boosting

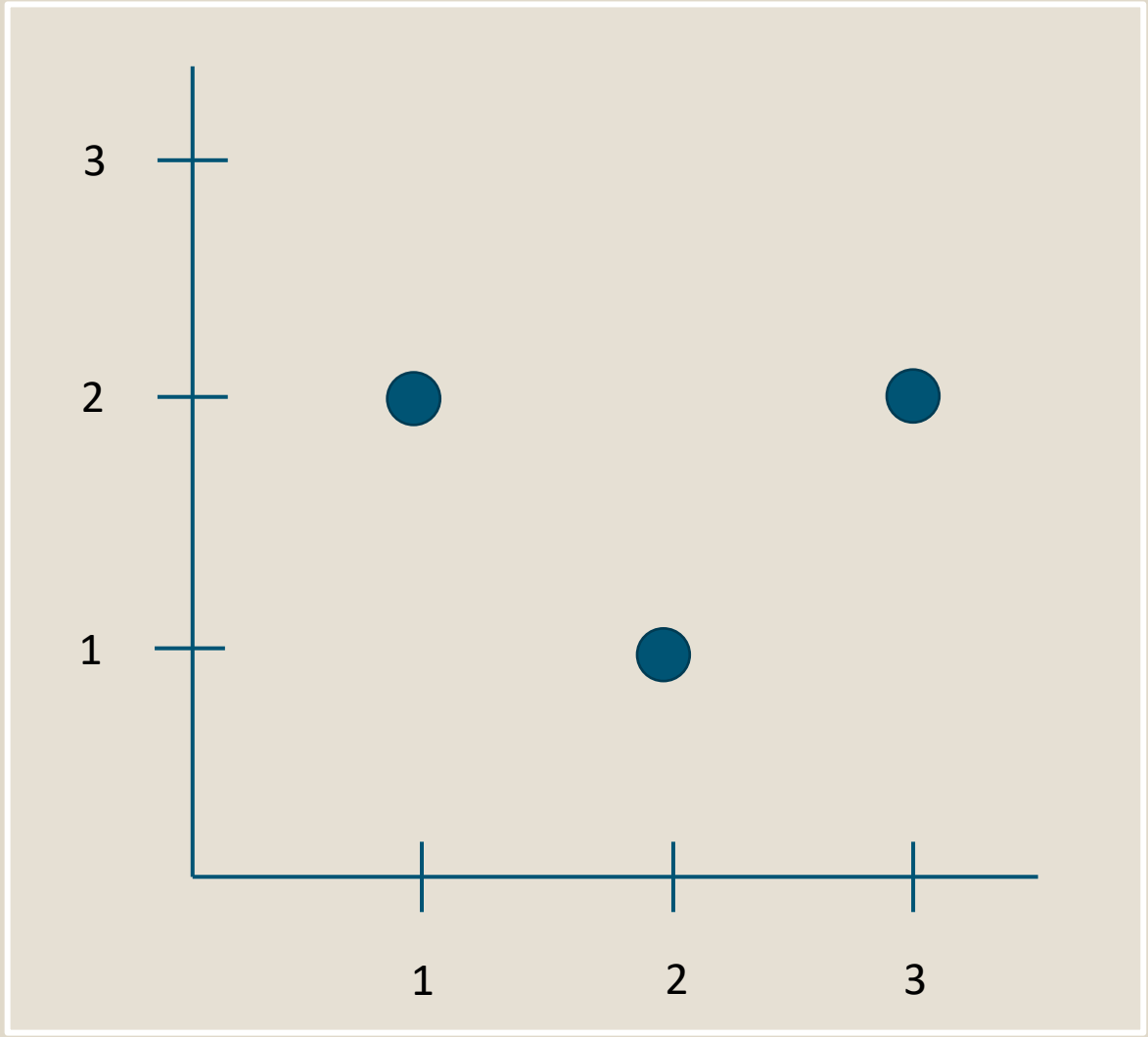


Example of boosting: Gradient Boosting

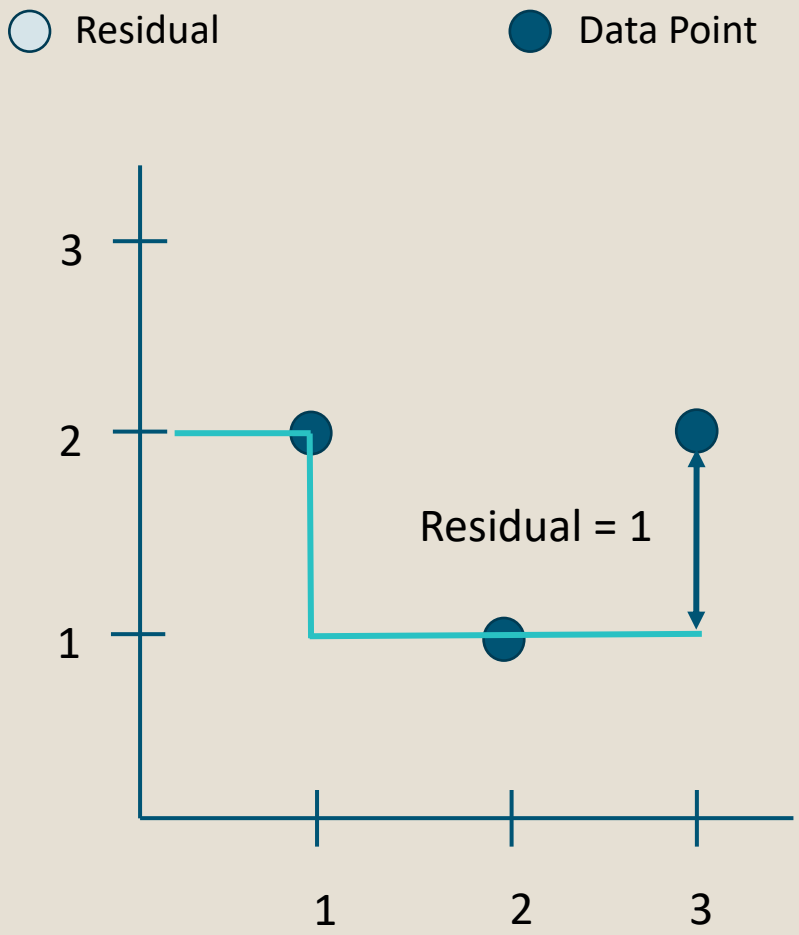
The same idea as general boosting, but instead of changing the weights of the samples, gradient boosting tries to fit the next predictor to the residual error made by the previous tree

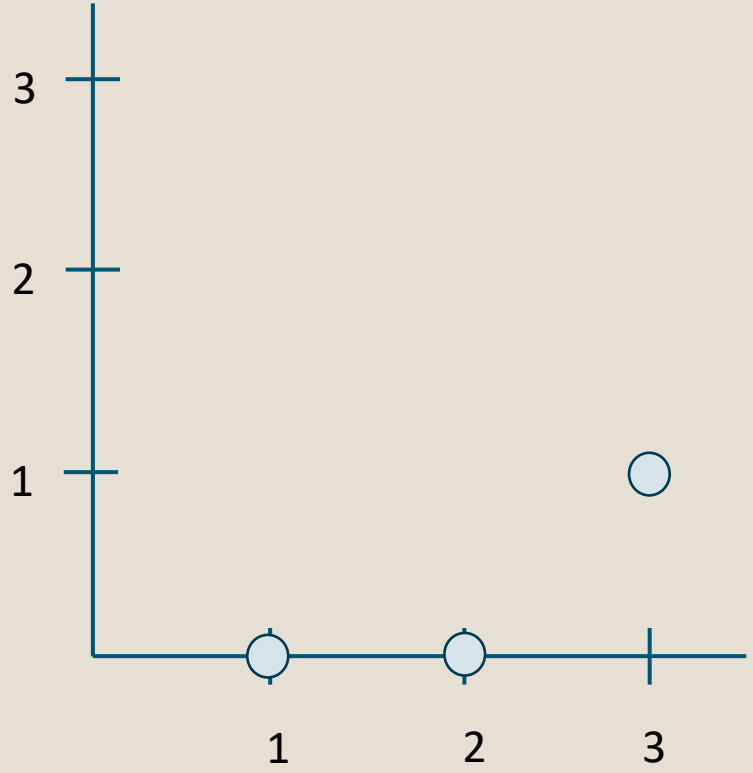
Example calculation

- We are looking at a regression problem
- The decision stump is a Decision Tree regressor with a single split
- For the dataset on the right, a single split tree can not solve the problem
- The method must combine multiple decision stumps to solve the problem

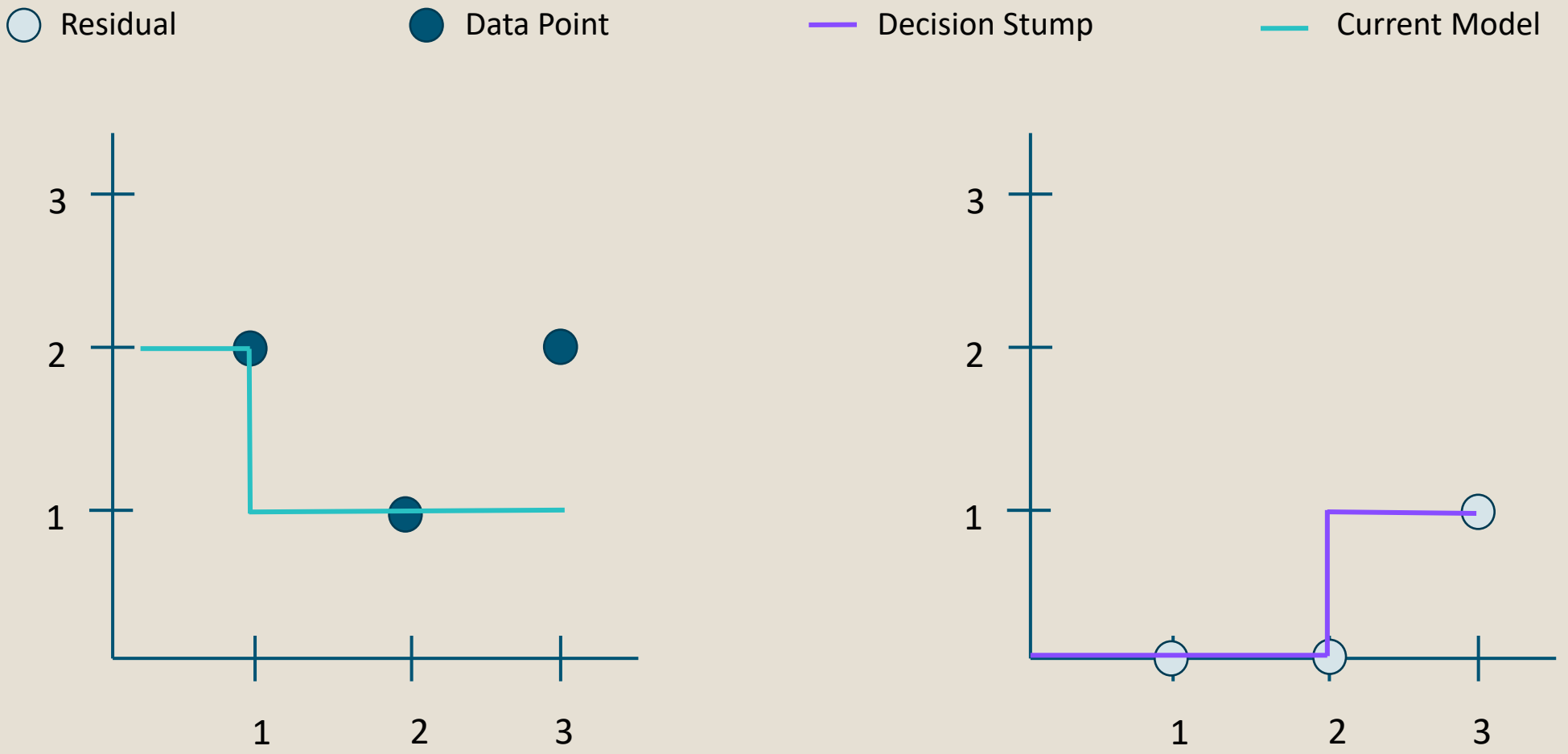


Example of boosting: Gradient Boosting

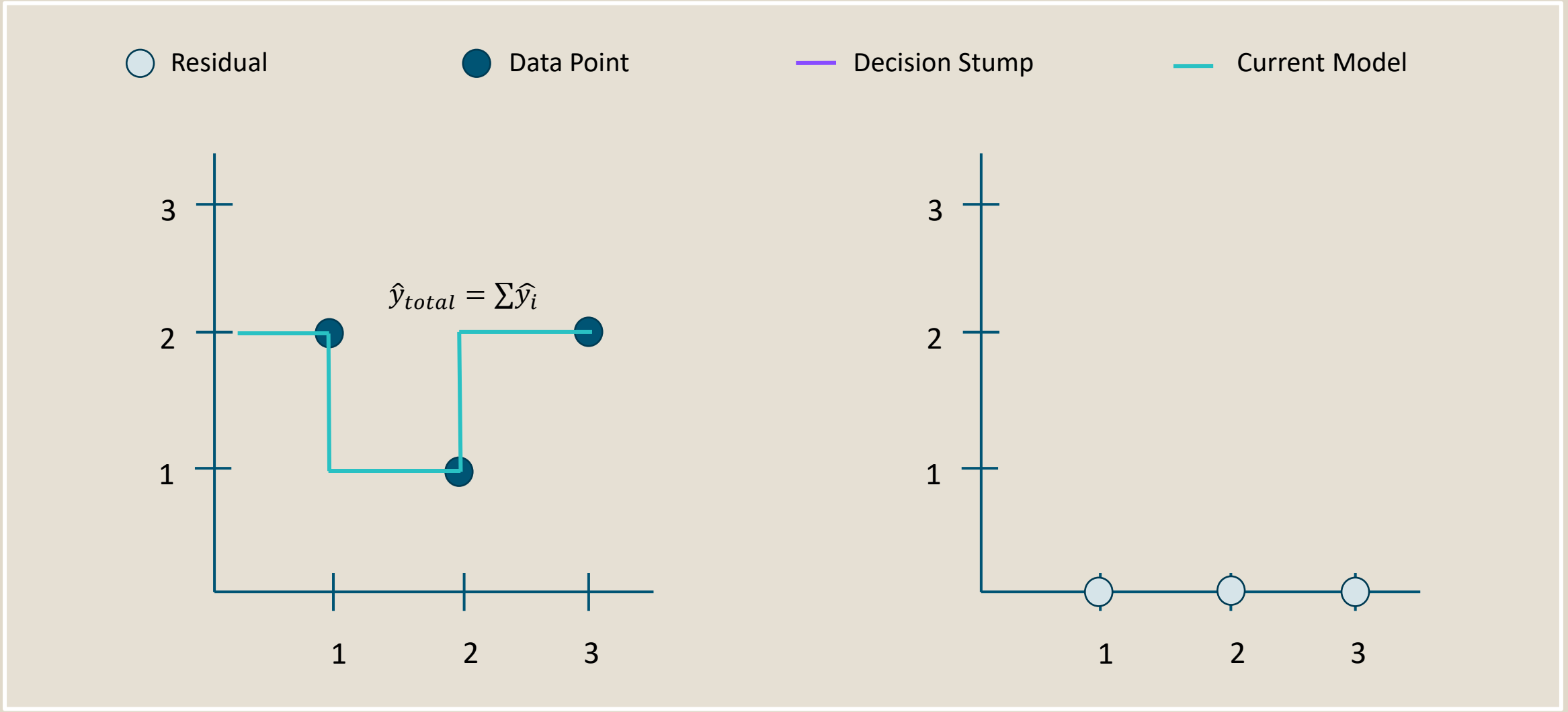




Example of boosting: Gradient Boosting



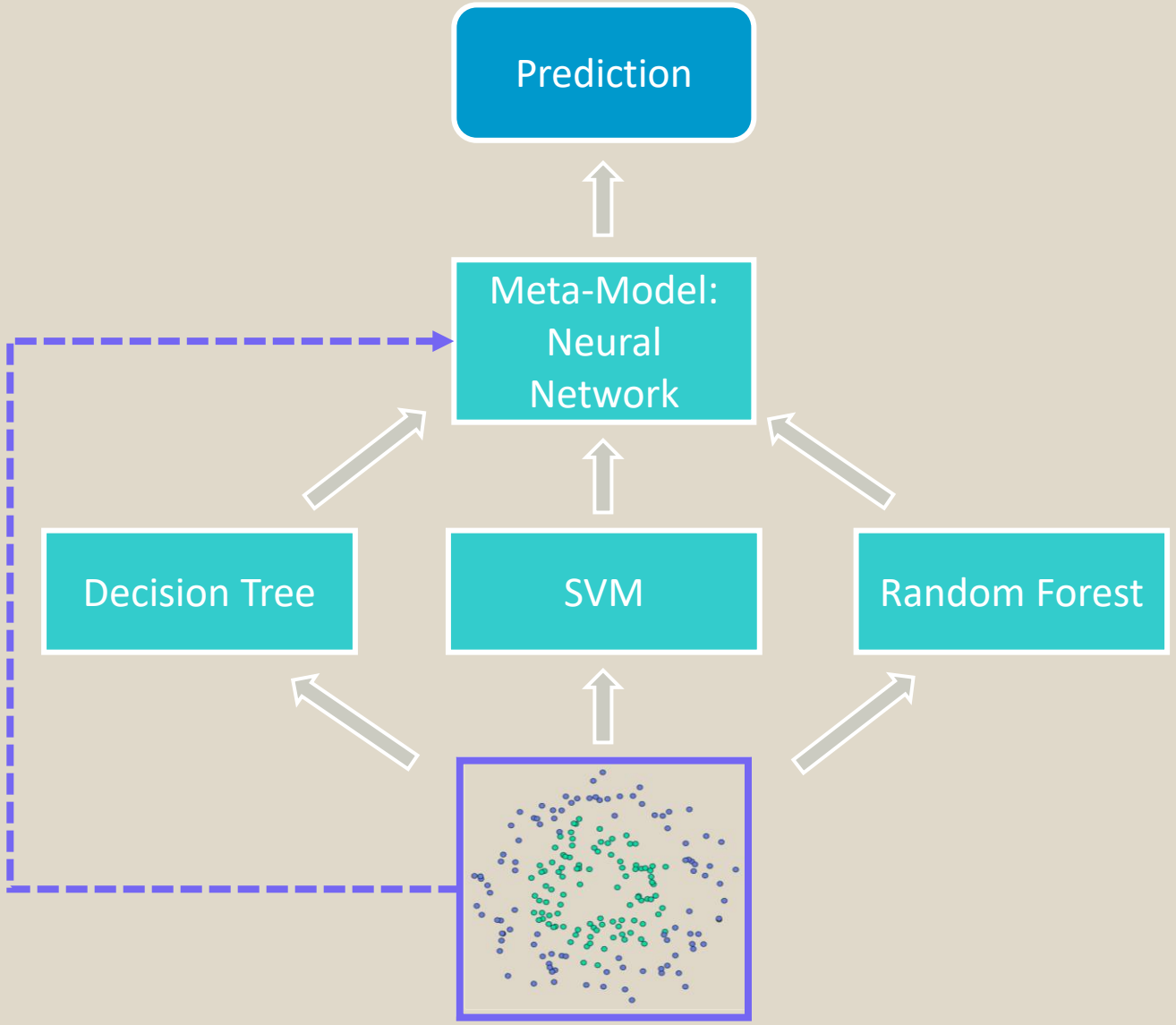
Example of boosting: Gradient Boosting



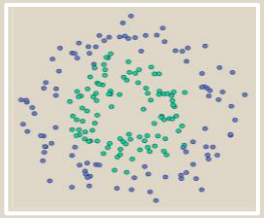
Combining multiple models with Stacking

Instead of predicting the majority vote of all classifiers we train a new classifier that combines the prediction of all models in the ensemble

- Depending on the current data sample, a specific model would be suited to do the prediction
- The meta-model learns to weight the decisions of the models depending on the context
- We can give the meta-model access to the dataset, for it to improve on the prediction of the previous models
- We can also have multiple layers of models that each improve on the prediction of the previous layer

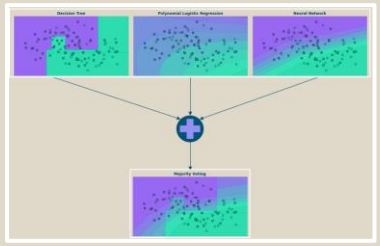


Important takeaways



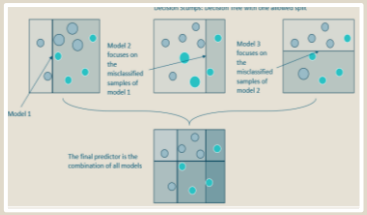
We can use **complex models to solve non-linear problems**

- Even though, complex models are more powerful it might be better to use simple models first



We can use **ensemble techniques** to combine multiple models

- By the process of combination, we gain a much more powerful model
- Possible techniques for combination are majority voting, weighted voting and stacking



We can systematically create **uncorrelated weak learners** to build ensembles

- **Bagging** is the method of building models on subsets of the data
- **Boosting** is the process of building models sequentially to improve on the error of the previous learner

We use complex models to solve non-linear problems

Quiz: Complex models

Please join at slido.com with #031 077.

Let’s go through some questions together.

Let’s see what you think. All answers will be anonymous.

Try it yourself!

In the following exercises

Feedback and Q&A

Thank you

If you would like any further information please contact
Werner,Dr.,Fabian_Georg (BI X) BIX-DE-I
<fabian_georg.werner@boehringer-ingelheim.com>