Data Science Academy

Data Science for Business –
Becoming a Data Science Expert (D)

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

21.09.2021

Boehringer Ingelheim

Agenda week four

	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 four

Week 4				
	Day 1 Tuesday, 21.09.2021		Day 2 Wednesday, 22.09.2021
Start: 12:00	Recap	Start: 12:00	Recap
	9 – Repetition (Part 1)		10 – Bias & Ethics in ML
14:00 – 15:00	Break	14:00 – 15:00	Break
	9 – Repetition (Part 2)		11 – Introduction to Data Science with AWS
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.

Module 9
Repetition

Agenda

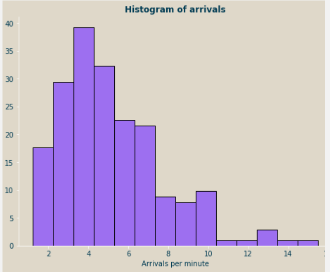
Data Science methods

Data Analysis	Machine Learning		
Descriptive statistics	**Supervised learning**	**Unsupervised learning**	**Reinforcement learning**

Descriptive statistics

Description:
- Basic data analysis, for instance aggregation and summary of data

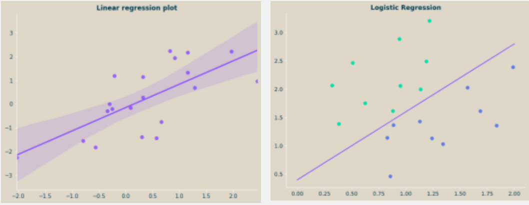
Application:
- Demonstrate distribution of age and sex of clinical trial participants



Supervised learning

Description:
- Use of labeled datasets to train algorithms with the goal of classifying data or predicting outcomes

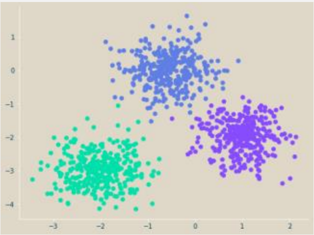
Application:
- Predict if someone would have a stroke or not



Unsupervised learning

Description:
- Analyze unlabeled datasets in order to discover hidden patterns or data groupings

Application:
- Groups of shoppers based on their browsing and purchasing histories



Reinforcement learning

Description:
- A computer program interacts as an agent with a dynamic environment in which it must perform a certain goal.

Application:
- Go-playing agents

Popularity/ business usage in the market

Data Science methods

Data Analysis	Machine Learning		
Descriptive statistics	**Supervised learning**	**Unsupervised learning**	**Reinforcement learning**

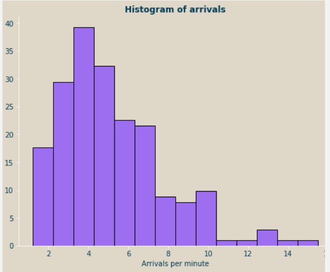
Descriptive statistics

Description:

- Basic data analysis, for instance aggregation and summary of data

Application:

- Demonstrate distribution of age and sex of clinical trial participants



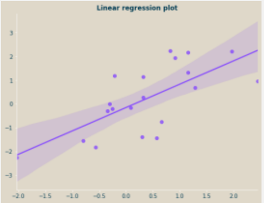
Supervised learning

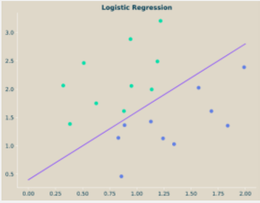
Description:

- Use of labeled datasets to train algorithms with the goal of classifying data or predicting outcomes

Application:

- Predict if someone would have a stroke or not





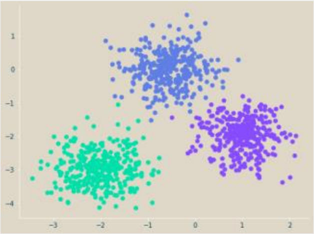
Unsupervised learning

Description:

- Analyze unlabeled datasets in order to discover hidden patterns or data groupings

Application:

- Groups of shoppers based on their browsing and purchasing histories



Reinforcement learning

Description:

- A computer program interacts as an agent with a dynamic environment in which it must perform a certain goal.

Application:

- Go-playing agents

Popularity/ business usage in the market

Supervised Learning

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

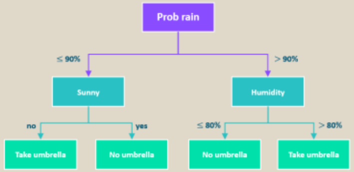
Strengths

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

Weaknesses

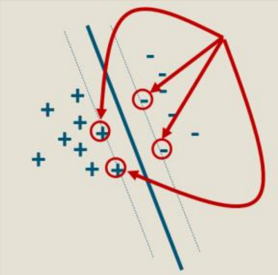
- Prone to overfitting
- Difficult to interpret

Important complex models



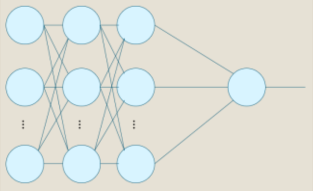
Decision Tree

- Use recursive splitting of nodes to increase the homogeneity in each leaf
- The final prediction is the majority class in each leaf



Support Vector Machines

- SVMs try to maximize the margin between support vectors
- The model can utilize kernels to transform the data into higher dimensional spaces

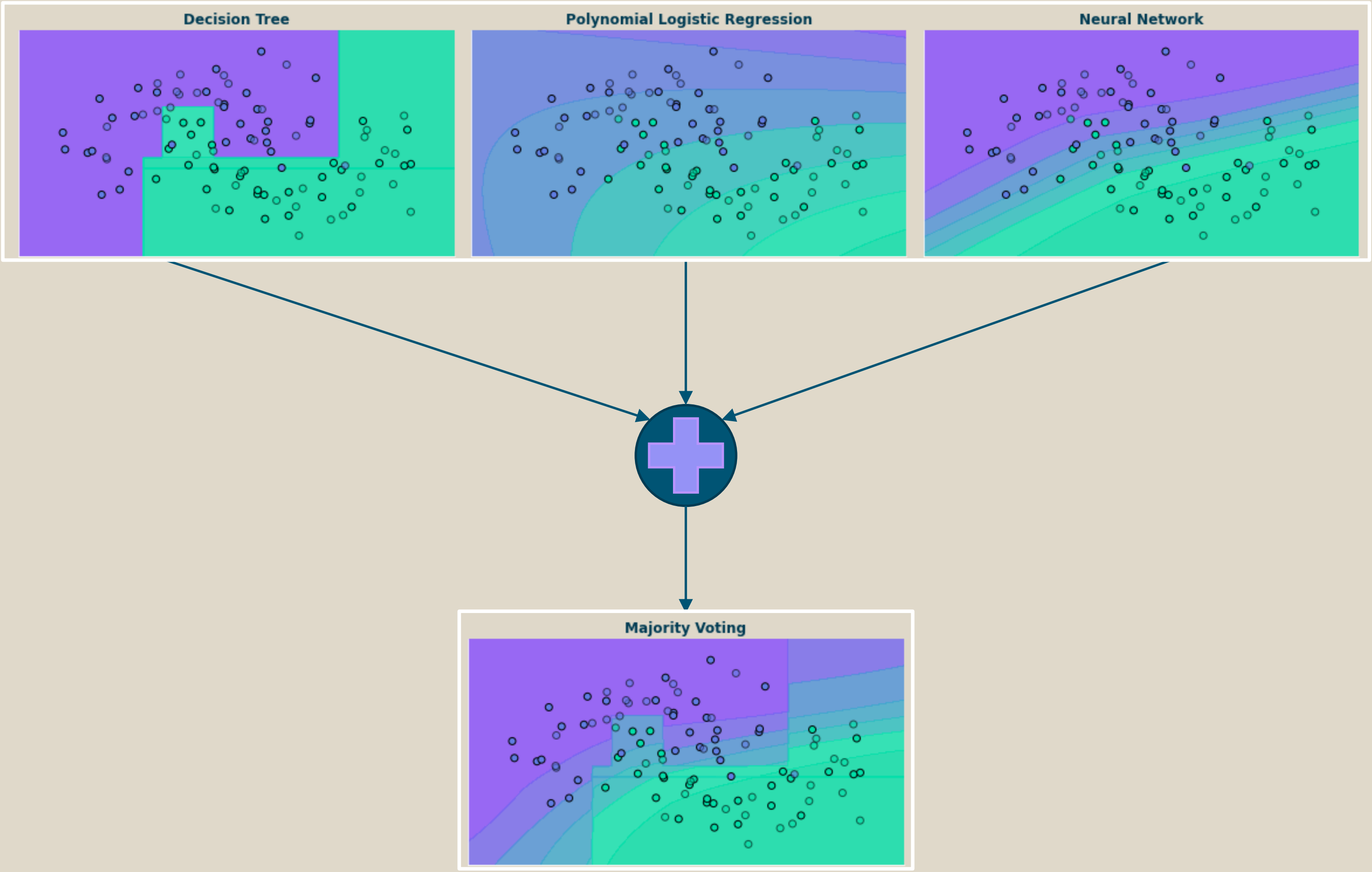


Neural networks

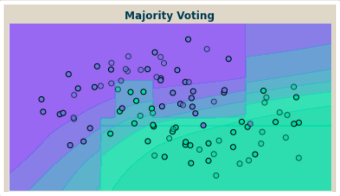
- Neural Networks stack multiple layers and combine them with a non-linear activation function
- They can be seen as a universal function approximator

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

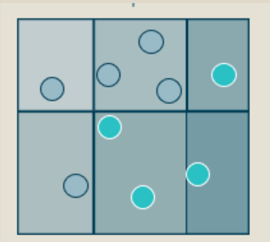


Ensembles



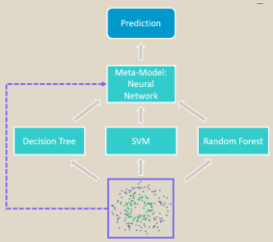
Bagging (Random Forest):

- By building weak learners on subsets of the data, we can systematically create an ensemble of uncorrelated learners



Boosting:

- By sequentially building weak learners to correct the error of the previous one, we can build a good performing ensemble
- A special case of boosting is gradient boosting



Stacking:

- Stacking combines arbitrary models into an ensemble
- A meta learner decides for each instance the weighting of the different decisions from the base leaner

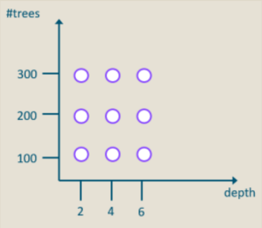
Measure how well the model is performing

Level	Example	Area	Advantages	Disadvantages	Comment
1. Loss function	• Mean squared error loss • Cross entropy loss	**Model training**	• Easy to compute the derivative • Evaluates goodness of fit	• Difficult to interpret	• How well do the current model parameters work on the training set?
2. Evaluation metrics	• Mean absolute error (MAE) • Accuracy	**Data Science**	• Easy to understand • General applicability	• Influence on the business is not apparent	• How well does the model generalize?
3. Business metrics	• Return on investments (ROI) • Click through rate (CRT)	**Management**	• Shows influence on business	• Has to be defined • Strongly depends on definition	• How much money will we gain/loose from applying this model in production?

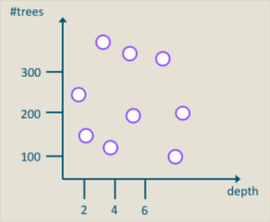
Overview: Supervised metrics

Classification Metrics	Accuracy	$accuracy = \frac{TP+TN}{P+N}$
	Precision and recall	$Precision = \frac{\mathrm{TP}}{\mathrm{TP+FP}}$ and $\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP+FN}}$
	F1 score	$F1 = \frac{2 * \mathrm{precision} * \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}$
	Sensitivity and false positive rate	$Sensitivity = \frac{TP}{P}$ and $\mathrm{FPR} = \frac{FP}{N}$
Regression Metrics	Mean absolute error	$MAE = \frac{1}{N} \sum \lvert y_i - \hat{y}_i \rvert$
	Mean squared error	$MSE = \frac{1}{N} \sum (y_i - \hat{y}_i)^2$
	Root mean squared error	$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{N}}$

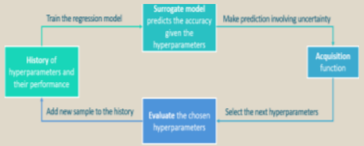
HyperParameter Optimization



Grid search:
- Define a discrete list of parameters that should be checked
- Check all combinations of parameters



Random search:
- Define ranges and distributions of values for each hyperparameter
- Sample parameters from the distribution



Bayesian Optimization:
- Predict the best next hyperparameter combination to evaluate
- Update the model with the evaluated hyperparameter combination
- Loop until convergence

Data Science methods

Data Analysis	Machine Learning		
Descriptive statistics	**Supervised learning**	**Unsupervised learning**	**Reinforcement learning**

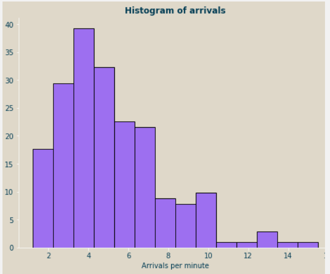
Descriptive statistics

Description:

- Basic data analysis, for instance aggregation and summary of data

Application:

- Demonstrate distribution of age and sex of clinical trial participants



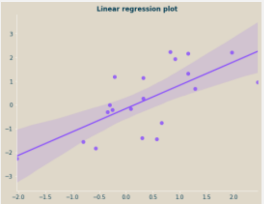
Supervised learning

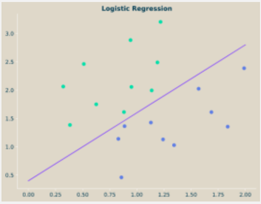
Description:

- Use of labeled datasets to train algorithms with the goal of classifying data or predicting outcomes

Application:

- Predict if someone would have a stroke or not





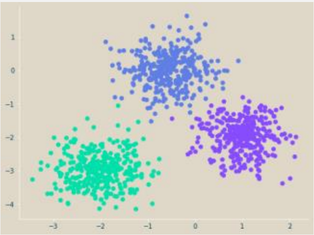
Unsupervised learning

Description:

- Analyze unlabeled datasets in order to discover hidden patterns or data groupings

Application:

- Groups of shoppers based on their browsing and purchasing histories



Reinforcement learning

Description:

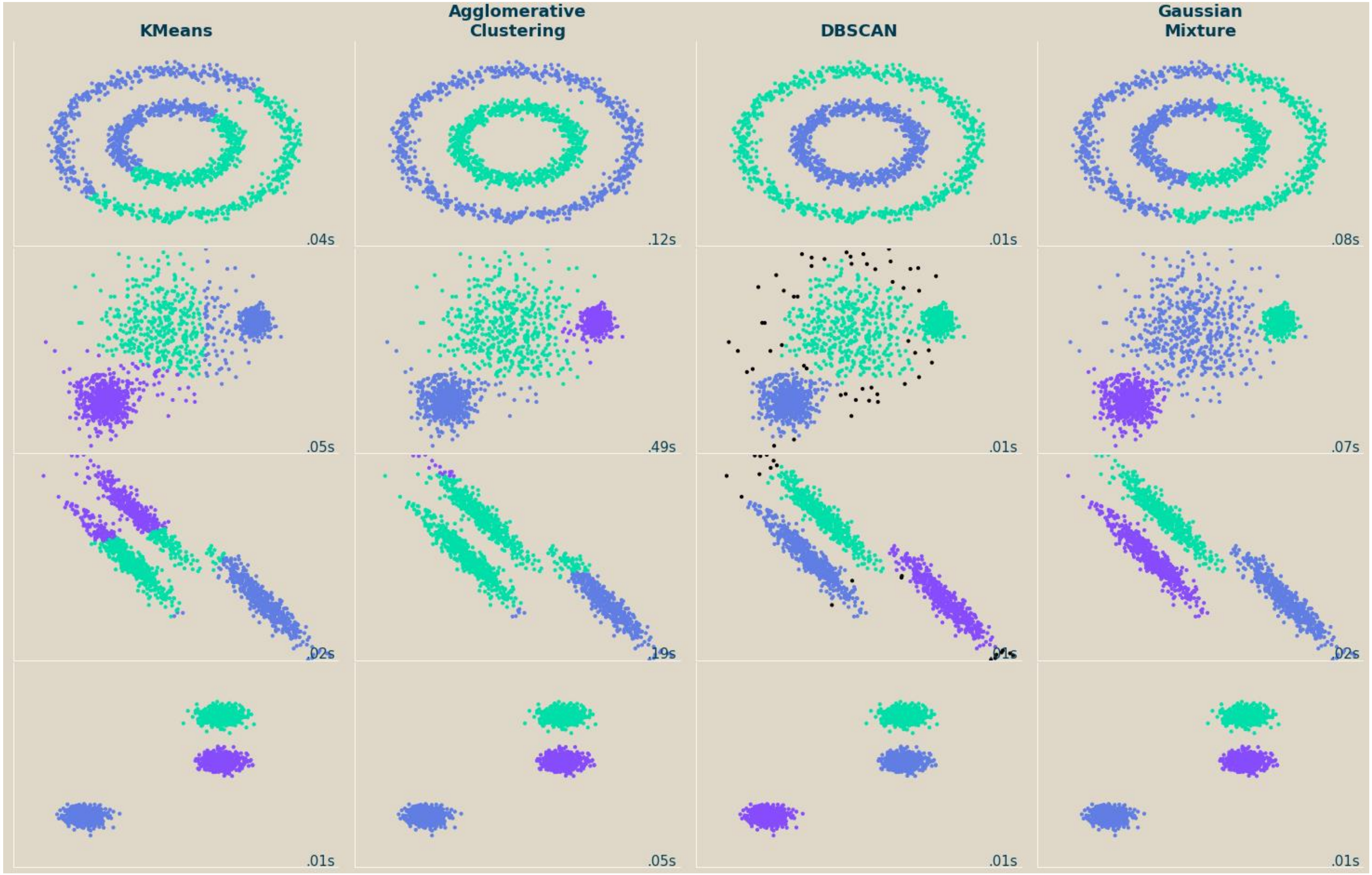
- A computer program interacts as an agent with a dynamic environment in which it must perform a certain goal.

Application:

- Go-playing agents

Popularity/ business usage in the market

Overview Clustering Algorithms



Evaluation metric for clustering

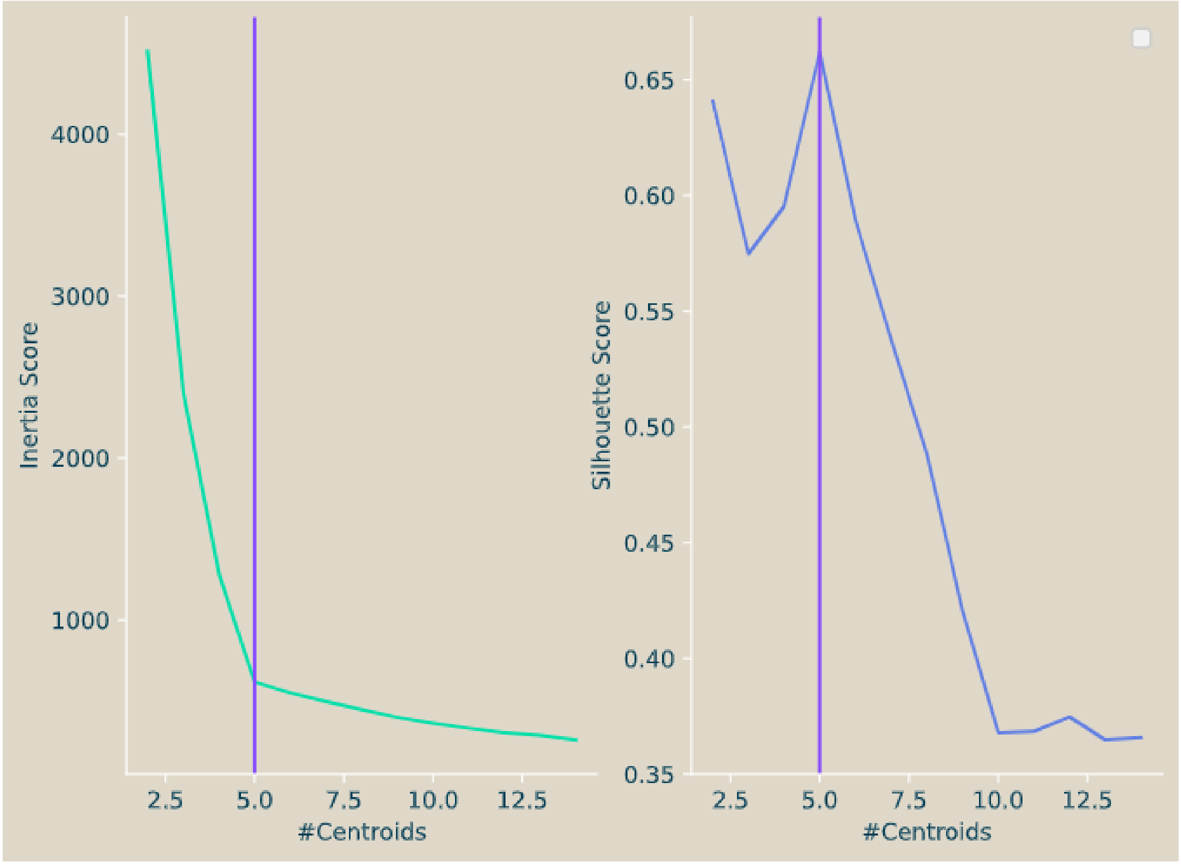
Within-cluster sum-of-squares criterion (inertia)

Measures the **sum of squared distances** between each datapoint and its assigned cluster center

Increasing the number of clusters decreases the within-cluster sum-of-squares criterion

Silhouette Score

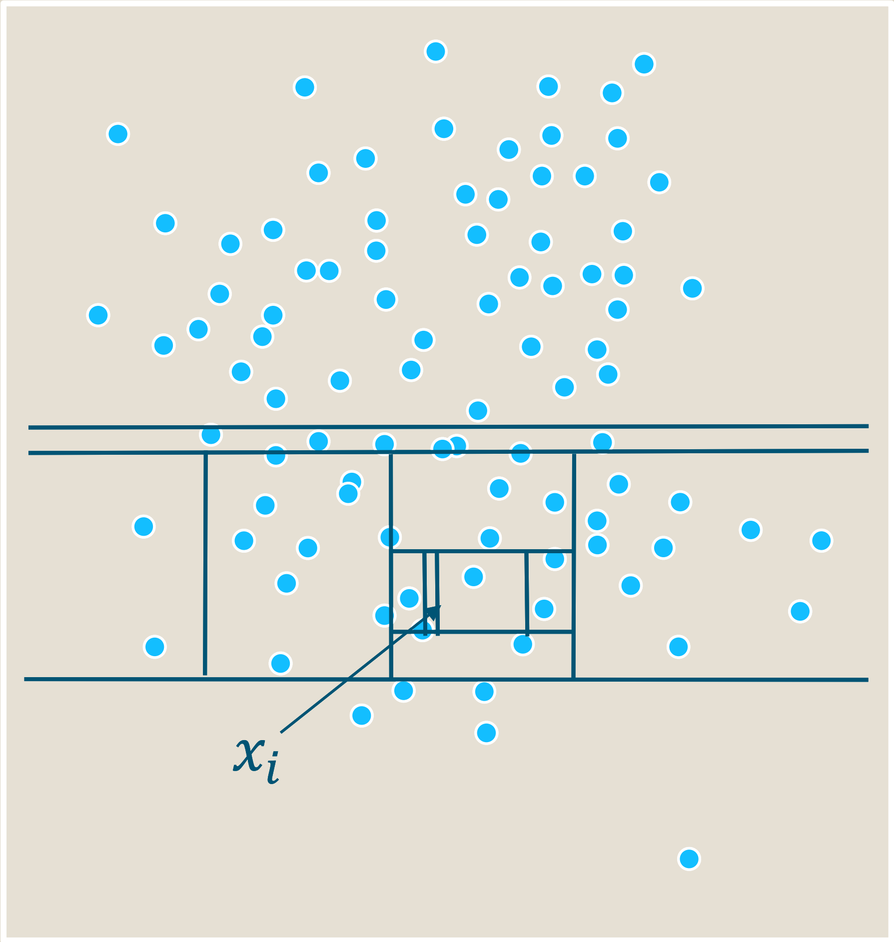
- Calculates the **difference between the average distance** of samples **within a cluster** and the average distance between a sample and the **nearest neighboring cluster**
- A value of **1 is the best possible value**. A value of **0** indicates **overlapping clusters**. A value **-1** means points were assigned to the **wrong cluster**



Isolation Forest

Create a random forest as follows:

- Create a lot of random trees as follows:
  - Repeat until single sample in node or set depth reached:
    - Create split by randomly select a feature and a random split value between minimum and maximum of that feature
- For each sample: Compute a score based on the average path length (i.e., number of splits in the tree until we end up a that sample)
  - Outlier x_0 will have a lower score since the average path length smaller (x_0 is easier to "isolate")



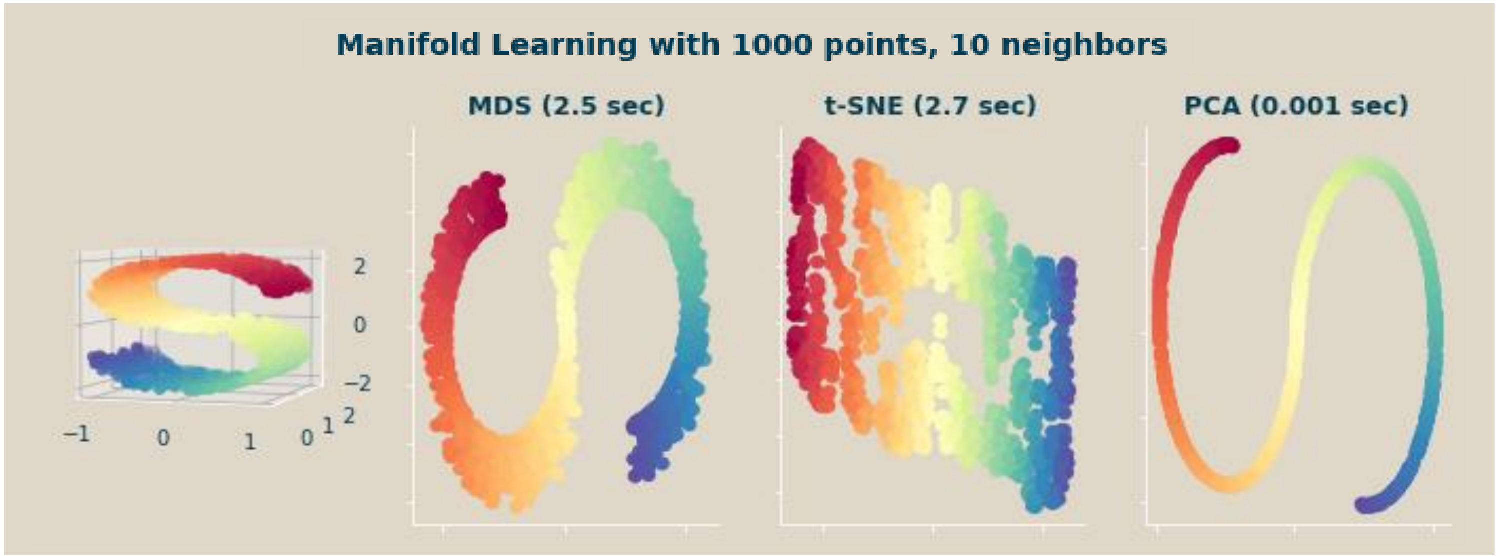
Dimensionality Reduction

Goal

- **Reducing the number of features**, i.e. the dimensionality of the data set
- Feature selection: only use a subset of the available features
- Feature extraction/projection: Transform data from high-dimensional space to a space of fewer dimensions

Why to do it?

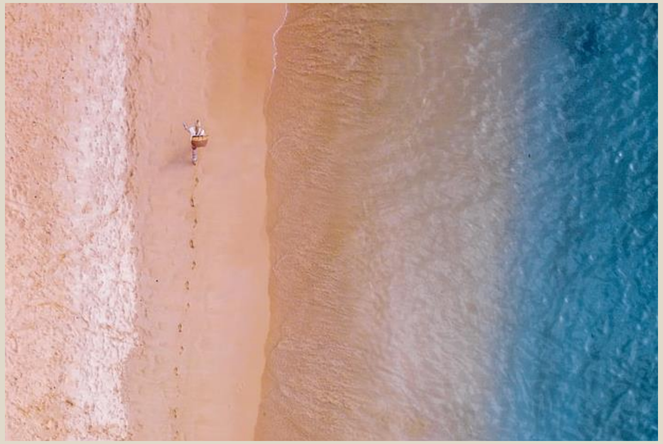
- High-dimensional data sets are hard to understand for humans and some models (this is sometimes called the curse of dimensionality)
- Speed up model training and evaluation
- Reduces noise and redundant/linear correlated feature in data (especially PCA) might increase model performance

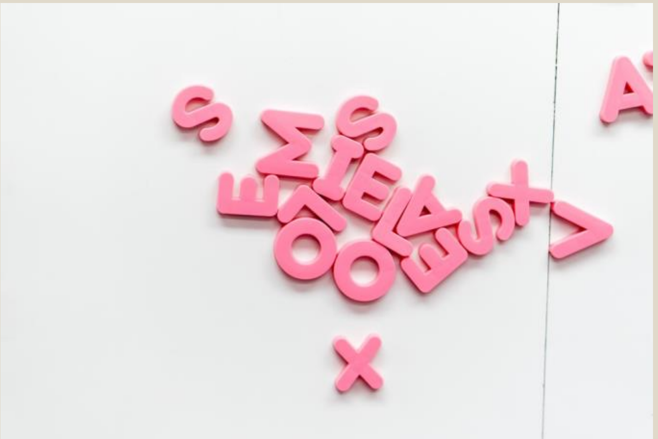


Try it yourself!

In the following exercises

Applications and achievements of deep learning





More mature and industrial applications **In active development and research**

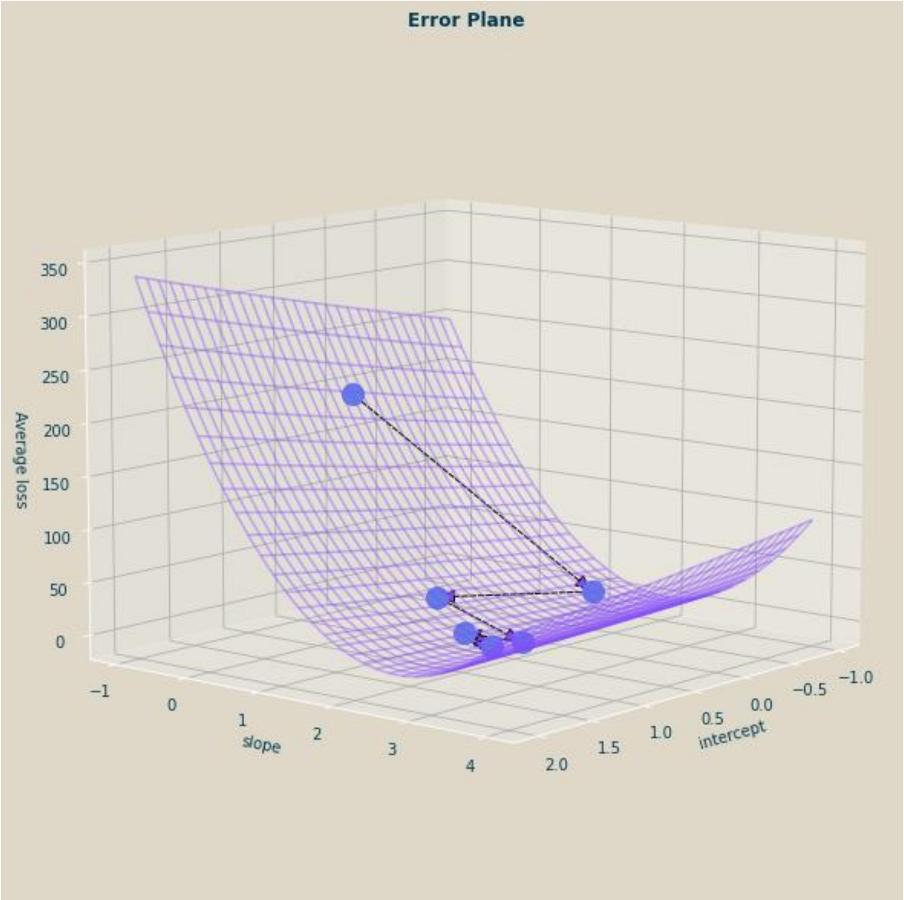
Near-human-level image classification Improved machine learning	Near-human-level speech recognition Near-human-level handwriting transcription	Autonomous driving

Gradient Descent Algorithm

Minimize $L(\theta)$

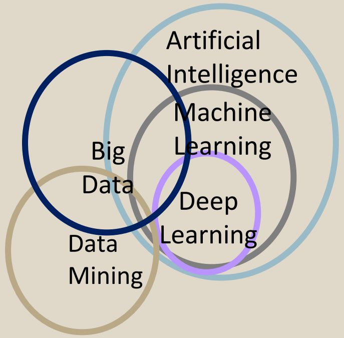
Input (decide on): A function $f(x;\theta)$, input x, target y, loss $L(\theta)$

1. **Initialize** θ **randomly**
2. **While** θ **still changes**
   1. Build the model $f(x;\theta)$
   2. Evaluate the fit with $L(\theta) = \sum_{i=1}^{n} l(y_i, f(x_i;\theta))$
   3. Update the parameters based on the negative gradient: $\theta = \theta \; - L'(\theta)$
3. **Return fitted model** $f(x;\theta)$

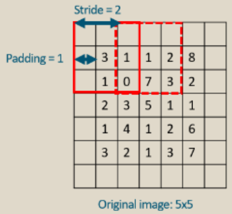


The gradient gives us the direction of the update.

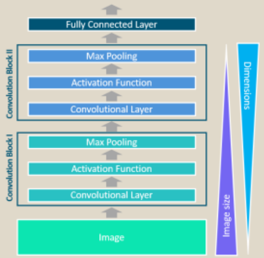
Deep Learning takeaways



Deep Learning is a subfield of Artificial Intelligence and is used for Problem with large data, like
- Image recognition
- Natural language processing

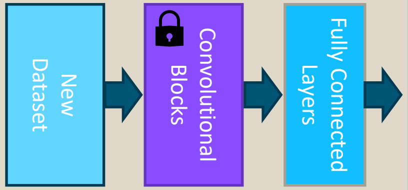


Filter layers are used as edge detectors. By combining the output of the edge detectors, a prediction about the content of the image is made. Parameters controlling the filter process are **filter size, padding and stride**.

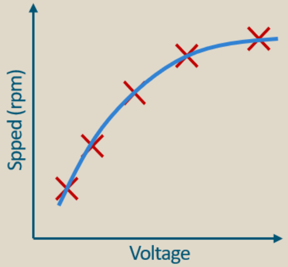


Convolutional Neural Networks (CNN) automate the feature engineering steps normally involved in image recognition. A CNN consists of layers of **Convolutional** and **Max-pooling layers**.

Computer Vision takeaways



We can reuse state-of-the-art image recognition models by retraining them on a different dataset.
- Retraining reduces the data required
- Retraining reduces the compute power required



With a reduced number of training samples it is of high importance to choose the samples carefully and correct any inconsistencies in the labeling process. This is called **data-centric view**.

For many advanced use cases in image recognition, we can use pre-trained filters. The advanced cases of this session were:
- R-CNN
- U-Net
- Face Recognition

In most practical cases we reuse state-of-the-art models and architectures and make only small changes to it

Setting up a NLP pipeline is not trivial

Tokenization
Break sentences into words, sylables or letters.

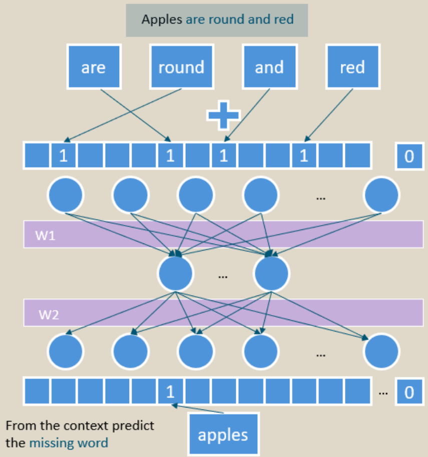
The Hunger games wasn’t a good movie

The	Hunger	games	was	n’t	a	good	movie

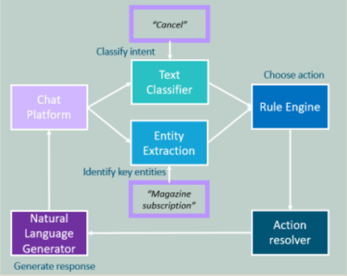
Stemming/Normalization
Merge Tokens of the same stem.

$\underbrace{\text{am} \quad \text{are} \quad \text{is}}_{\text{be}}$

Numericalization
Create word embeddings



NLP Use Cases



Chatbots:
- Use text classification and entity extraction to create natural language understanding
- Based on the understanding use predefined rules to perform the desired actions
- A natural language generator creates a response for the user

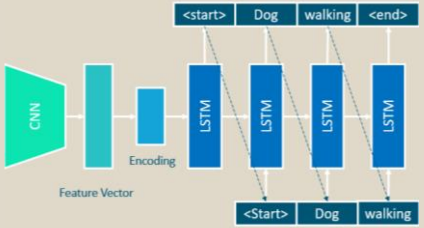
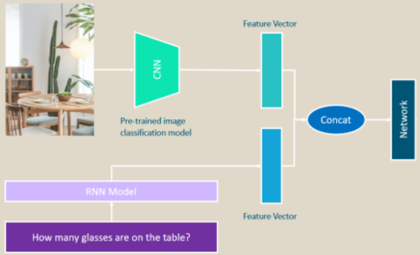


Image captioning:
- Combining the output of a convolutional neural network with a NLP model like an LSTM, we are able to describe the entities and actions depicted in the image



Visual question answering
- Multi-modal models combine two different modalities into a single model
- Like in visual question answering where text and image information is combined

Try it yourself!

In the following exercises

Agenda

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>