stackfuel

Focus Session

Data Science Overview

What is Machine Learning?

Machine Learning is the „Field of study that gives computers the ability to learn without being explicitly programmed "

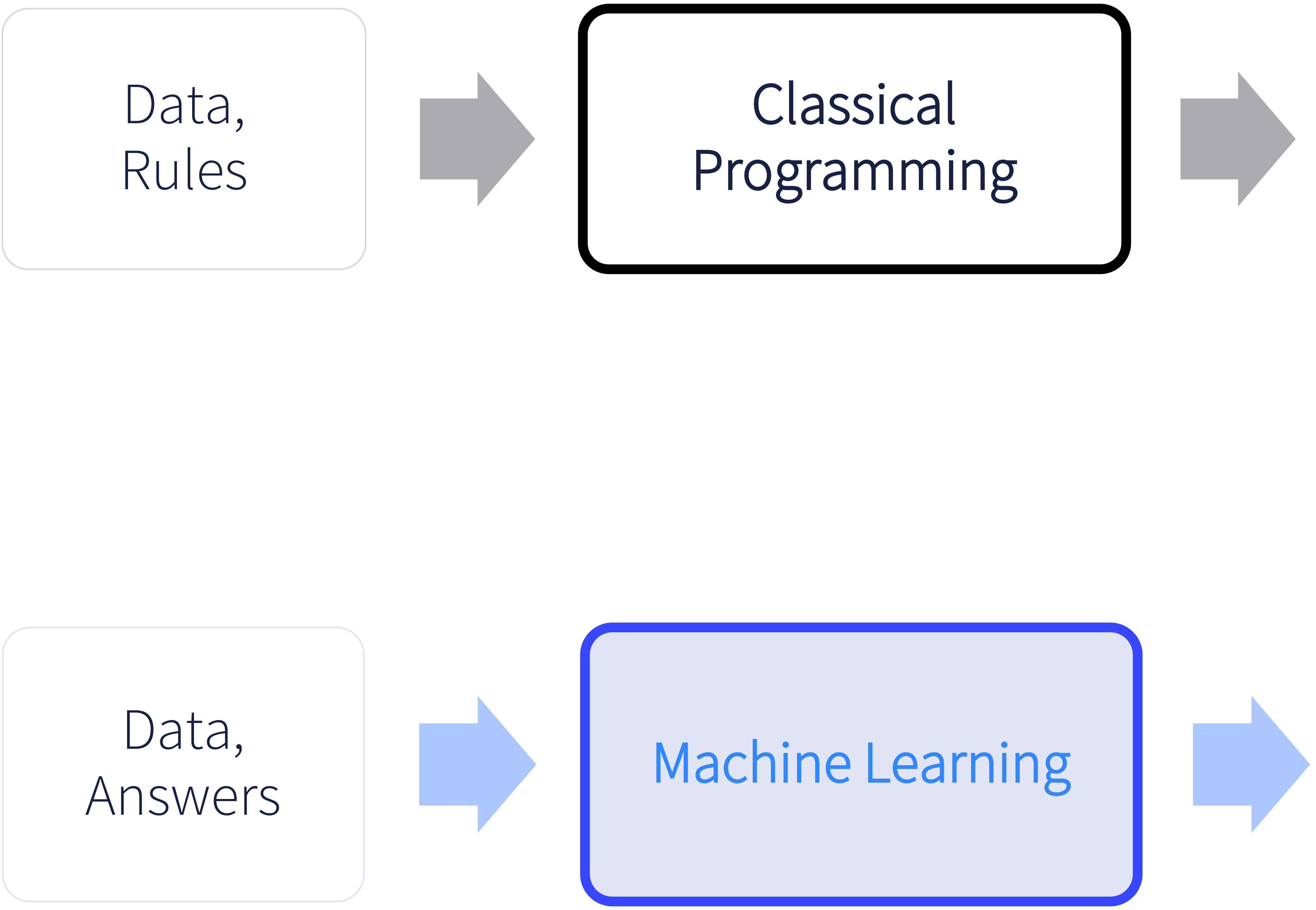
Arthur Samuel, 1959

„A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E. "

Tom Mitchell, 1997

Machine Learning is the art of programming computers such, that they can learn from data.

Programming paradigms



Data, Rules

Classical Programming

Answers

Data, Answers

Machine Learning

Rules

Example spam filter

If **in subject line (certain) character,** then **spam**

If **in subject line „credit card“ & „free“,** then **spam**

If **in subject line … ,** then **spam**

If **in text …,** then **spam**

If …

➡ Long list of increasingly complex rules

ML recognizes unusally frequent use patterns of words, phrases and characters → learns to predict spam in an automatic manner

Der Unterschied zwischen Machine Learning und „klassischer“ Software (neunetz.com)

Quelle: A.M. Turing, 1950, *Computational Machinery and Intelligence*

Why do we use Machine Learning?

ML techniques are applicable in a variety of fields:

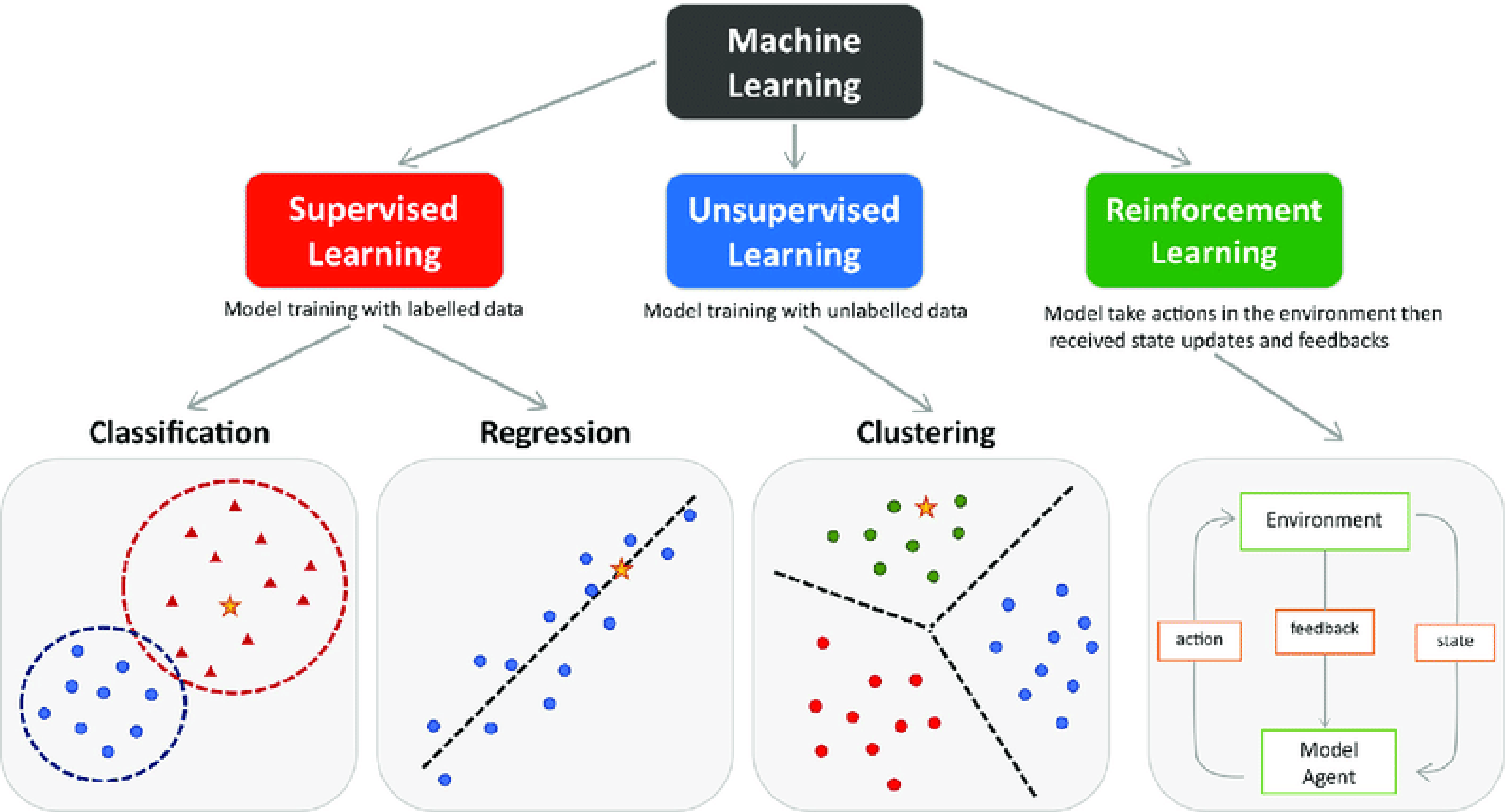
- Image recognition
- Speech recognition
- SemanticSpeech recognition
- Pattern recognition
- Process optimization

Why do we use Machine Learning?

Examples of different ML projects.:

- Diagnosis of hard-to-detect diseases
- prevention or detection of criminal behavior
- Prediction of house prices
- Product recommendation for customers
- FIlter/classification of texts (spam filter)
- Prediction of future revenue based on performance metrics
- Prediction of customer interest in certain products
- Customer segmentation
- Development for Intelligent Gaming Bots / intelligent NPCs

Types of Machine Learning



Machine Learning

Supervised Learning

Model training with labelled data

Unsupervised Learning

Model training with unlabelled data

Reinforcement Learning

Model take actions in the environment then received state updates and feedbacks

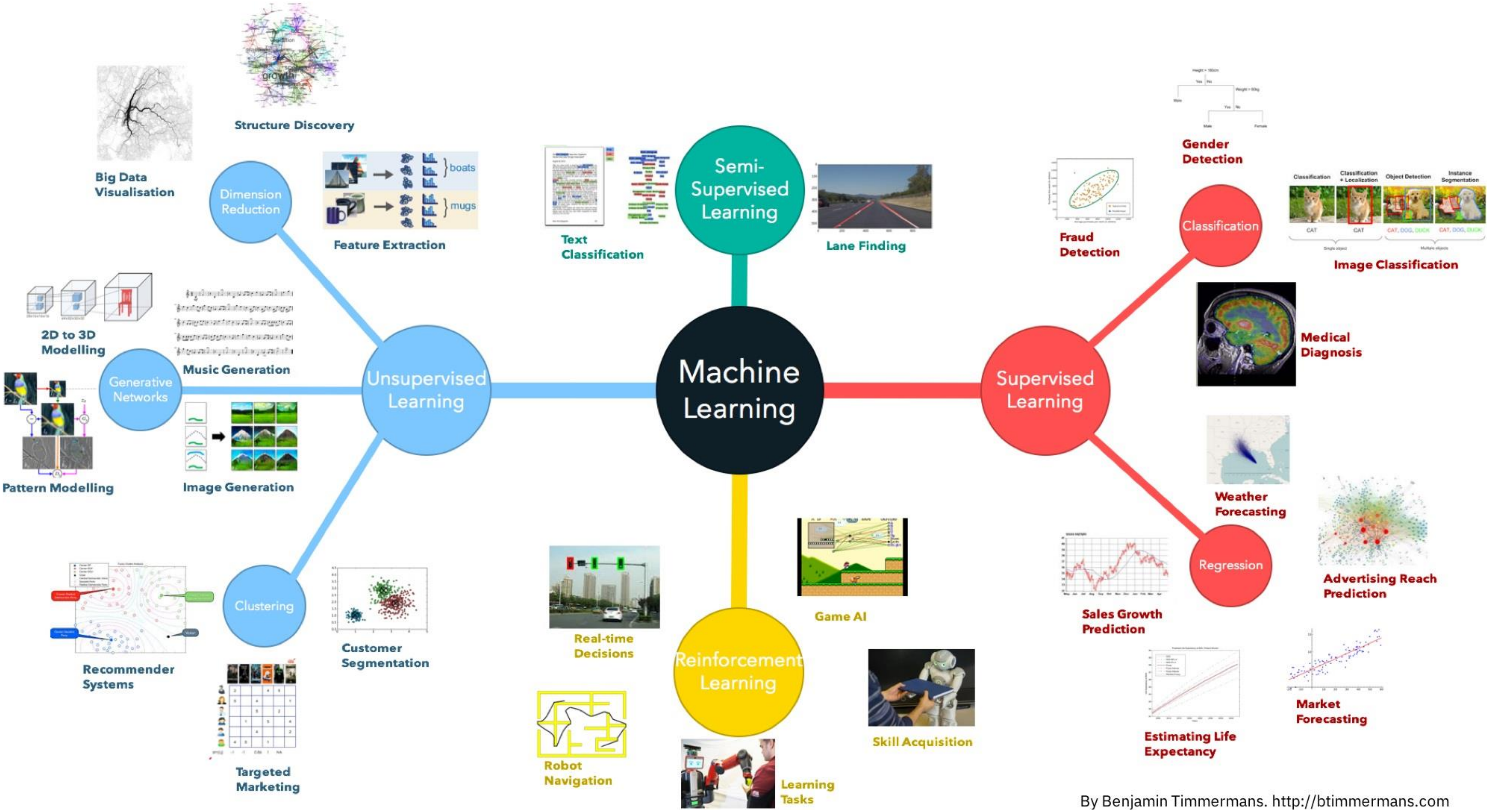
Classification

Regression

Clustering

Example: Logistic Regression Linear Regression DBSCAN

Quelle: Peng et al., 2021
https://www.researchgate.net/figure/The-main-types-of-machine-learning-Main-approaches-include-classification-and_fig1_354960266



Machine Learning

- Unsupervised Learning
  - Dimension Reduction
    - Big Data Visualisation
    - Structure Discovery
    - Feature Extraction
  - Generative Networks
    - 2D to 3D Modelling
    - Music Generation
    - Pattern Modelling
    - Image Generation
  - Clustering
    - Recommender Systems
    - Targeted Marketing
    - Customer Segmentation
- Semi-Supervised Learning
  - Text Classification
  - Lane Finding
- Supervised Learning
  - Classification
    - Fraud Detection
    - Gender Detection
    - Image Classification
    - Medical Diagnosis
  - Regression
    - Weather Forecasting
    - Advertising Reach Prediction
    - Sales Growth Prediction
    - Estimating Life Expectancy
    - Market Forecasting
- Reinforcement Learning
  - Real-time Decisions
  - Game AI
  - Robot Navigation
  - Learning Tasks
  - Skill Acquisition

By Benjamin Timmermans. http://btimmermans.com

Preconditions for Machine Learning

For predictions we need:

- Training data / predictors (*features*)
- Outcome variable (*target*)
- Data in a numerical format

The user needs:

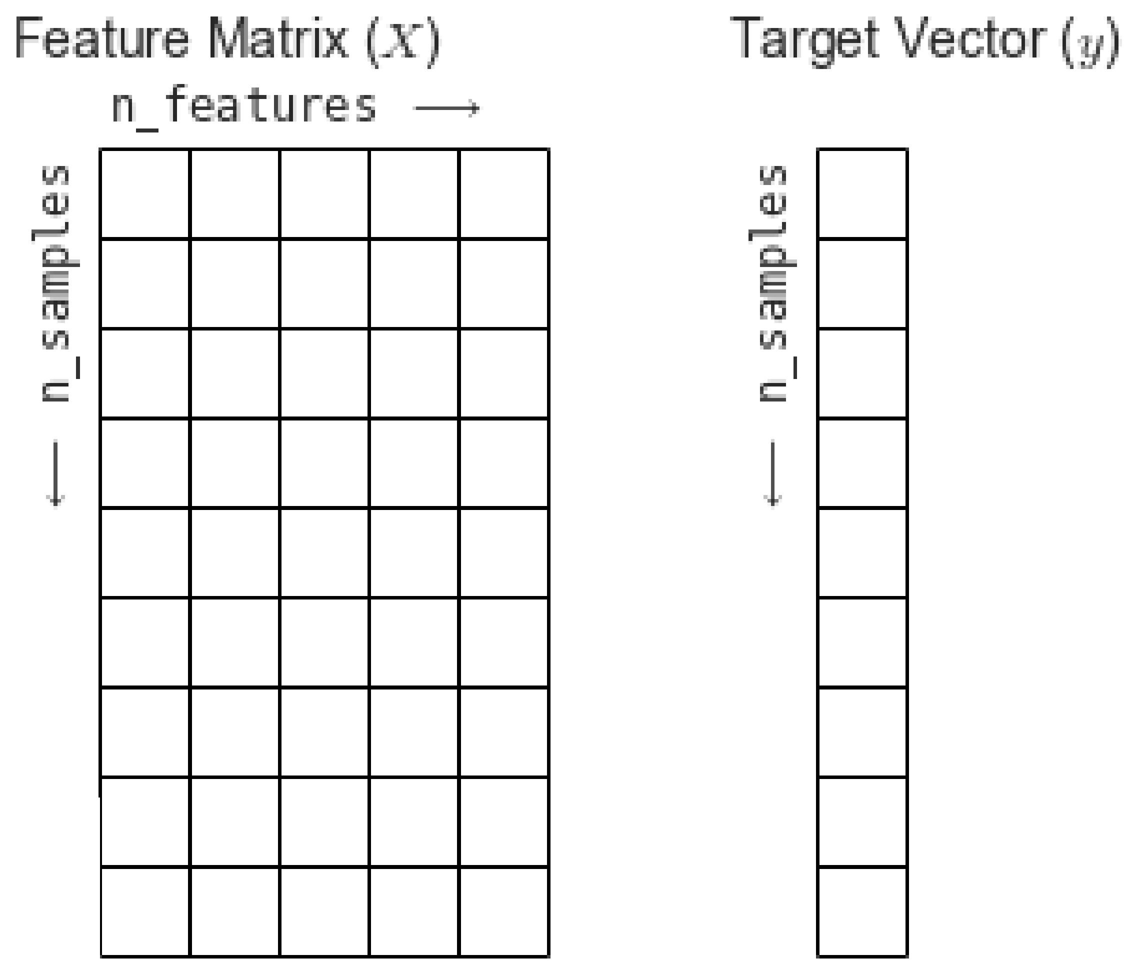
- Awareness of model limitations
- Awareness of data set limitations
- Awareness of problem statement

Example:

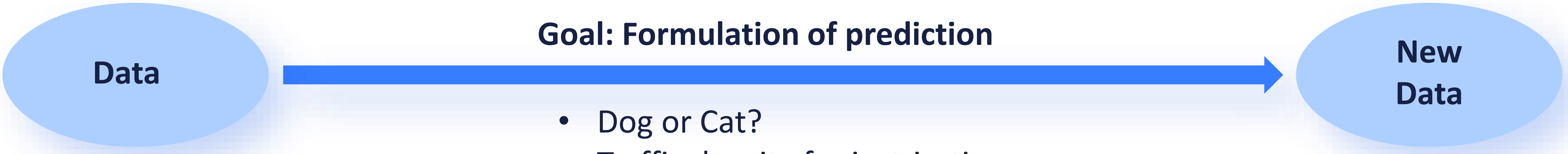
df_feature, X

df_target, y

0 … 1

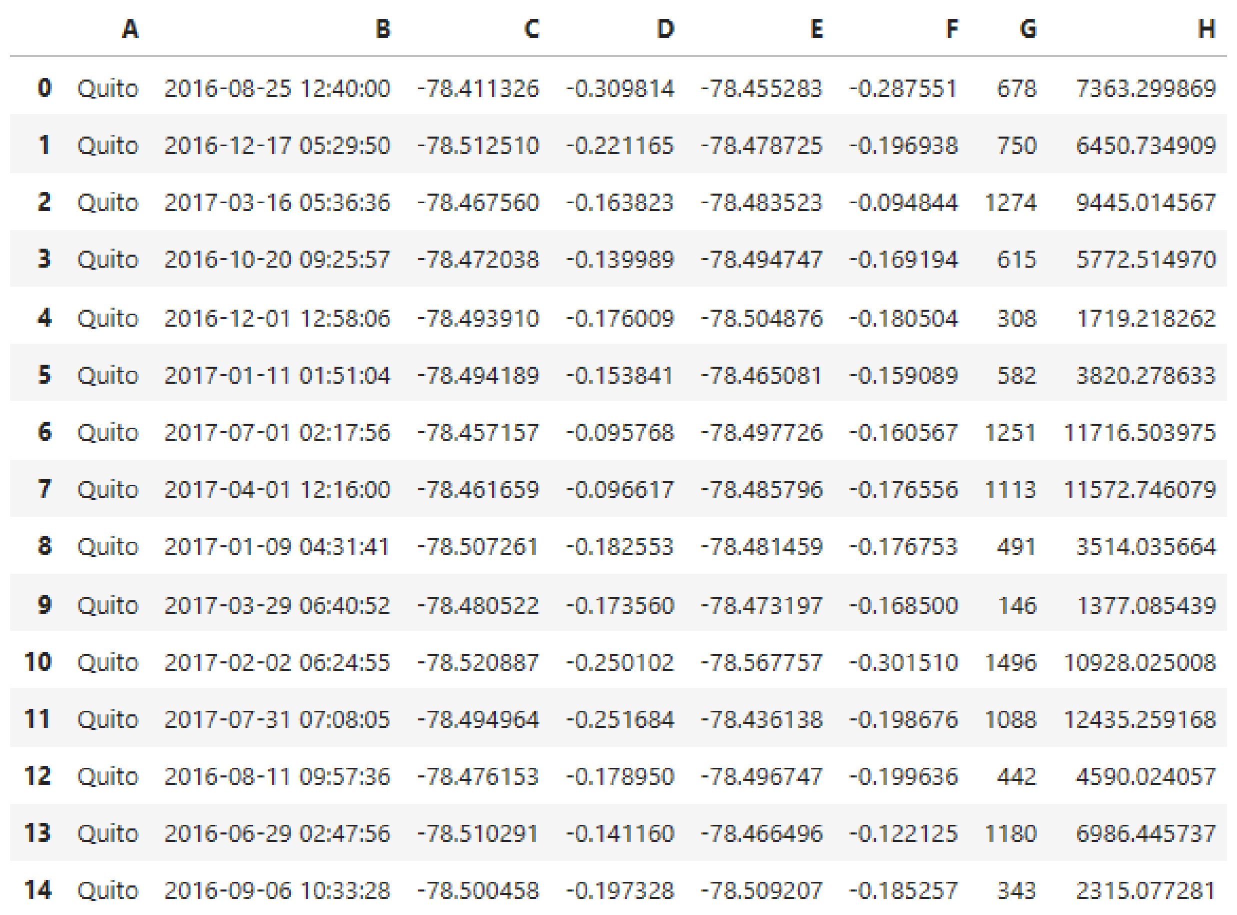


Basic approach in Machine Learning



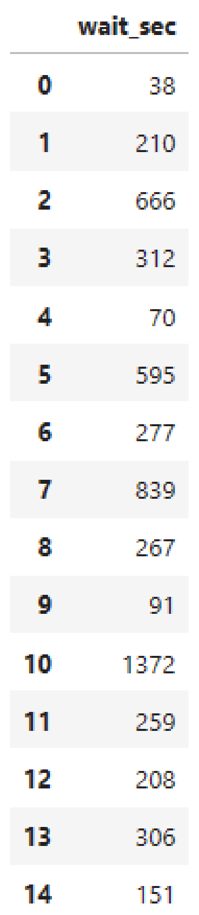
- Dog or Cat?
- Traffic density for just-in-time production (best route)
- Which machinery components need maintenance/replacement soon?
- Which employees will quit soon?
- Which customers might be interested in which product?
- Taxi waiting times
- …

features



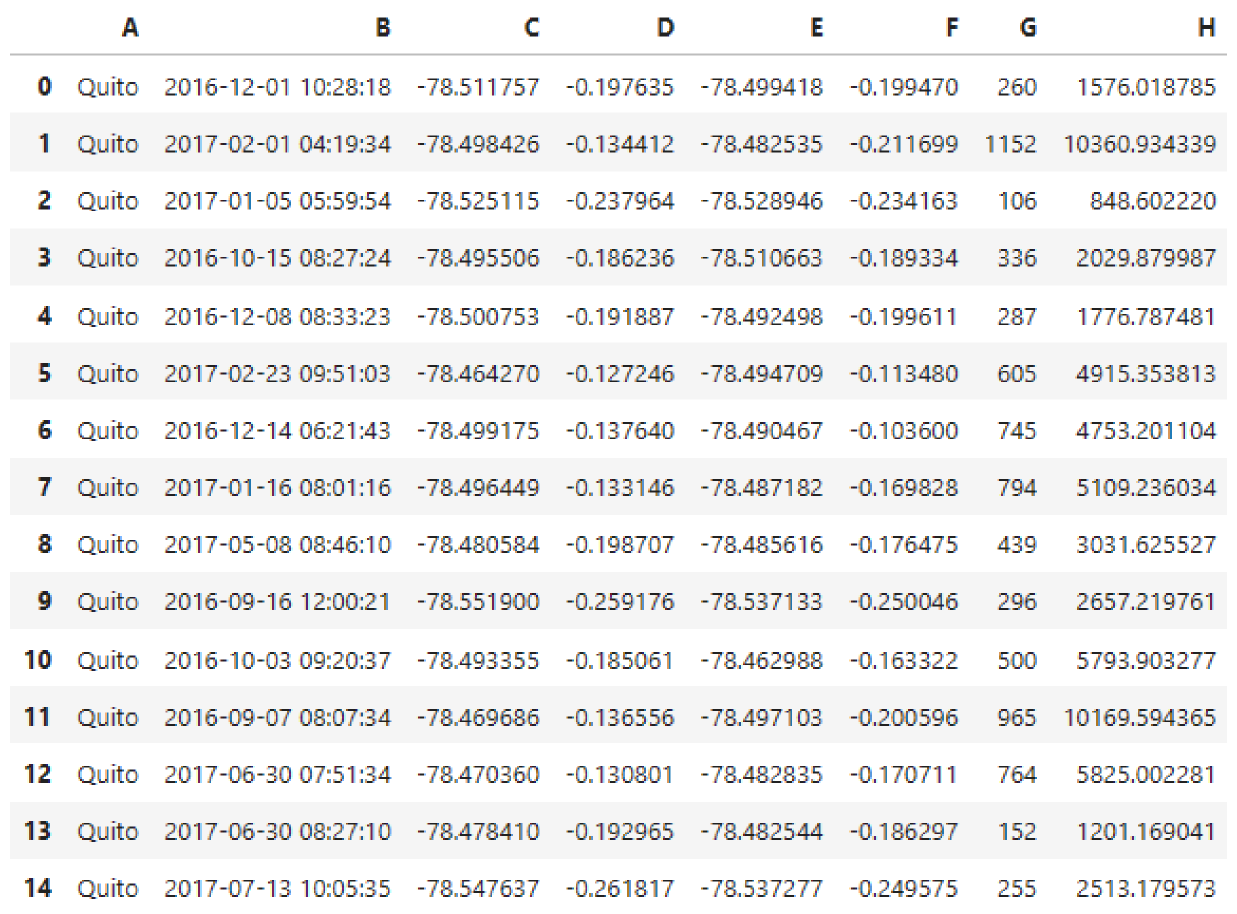
	A	B	C	D	E	F	G	H
0	Quito	2016-08-25 12:40:00	-78.411326	-0.309814	-78.455283	-0.287551	678	7363.299869
1	Quito	2016-12-17 05:29:50	-78.512510	-0.221165	-78.478725	-0.196938	750	6450.734909
2	Quito	2017-03-16 05:36:36	-78.467560	-0.163823	-78.483523	-0.094844	1274	9445.014567
3	Quito	2016-10-20 09:25:57	-78.472038	-0.139989	-78.494747	-0.169194	615	5772.514970
4	Quito	2016-12-01 12:58:06	-78.493910	-0.176009	-78.504876	-0.180504	308	1719.218262
5	Quito	2017-01-11 01:51:04	-78.494189	-0.153841	-78.465081	-0.159089	582	3820.278633
6	Quito	2017-07-01 02:17:56	-78.457157	-0.095768	-78.497726	-0.160567	1251	11716.503975
7	Quito	2017-04-01 12:16:00	-78.461659	-0.096617	-78.485796	-0.176556	1113	11572.746079
8	Quito	2017-01-09 04:31:41	-78.507261	-0.182553	-78.481459	-0.176753	491	3514.035664
9	Quito	2017-03-29 06:40:52	-78.480522	-0.173560	-78.473197	-0.168500	146	1377.085439
10	Quito	2017-02-02 06:24:55	-78.520887	-0.250102	-78.567757	-0.301510	1496	10928.025008
11	Quito	2017-07-31 07:08:05	-78.494964	-0.251684	-78.436138	-0.198676	1088	12435.259168
12	Quito	2016-08-11 09:57:36	-78.476153	-0.178950	-78.496747	-0.199636	442	4590.024057
13	Quito	2016-06-29 02:47:56	-78.510291	-0.141160	-78.466496	-0.122125	1180	6986.445737
14	Quito	2016-09-06 10:33:28	-78.500458	-0.197328	-78.509207	-0.185257	343	2315.077281

target



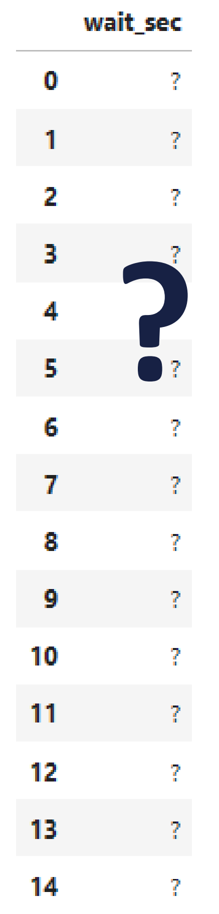
	wait_sec
0	38
1	210
2	666
3	312
4	70
5	595
6	277
7	839
8	267
9	91
10	1372
11	259
12	208
13	306
14	151

features_aim



	A	B	C	D	E	F	G	H
0	Quito	2016-12-01 10:28:18	-78.511757	-0.197635	-78.499418	-0.199470	260	1576.018785
1	Quito	2017-02-01 04:19:34	-78.498426	-0.134412	-78.482535	-0.211699	1152	10360.934339
2	Quito	2017-01-05 05:59:54	-78.525115	-0.237964	-78.528946	-0.234163	106	848.602220
3	Quito	2016-10-15 08:27:24	-78.495506	-0.186236	-78.510663	-0.189334	336	2029.879987
4	Quito	2016-12-08 08:33:23	-78.500753	-0.191887	-78.492498	-0.199611	287	1776.787481
5	Quito	2017-02-23 09:51:03	-78.464270	-0.127246	-78.494709	-0.113480	605	4915.353813
6	Quito	2016-12-14 06:21:43	-78.499175	-0.137640	-78.490467	-0.103600	745	4753.201104
7	Quito	2017-01-16 08:01:16	-78.496449	-0.133146	-78.487182	-0.169828	794	5109.236034
8	Quito	2017-05-08 08:46:10	-78.480584	-0.198707	-78.485616	-0.176475	439	3031.625527
9	Quito	2016-09-16 12:00:21	-78.551900	-0.259176	-78.537133	-0.250046	296	2657.219761
10	Quito	2016-10-03 09:20:37	-78.493355	-0.185061	-78.462988	-0.163322	500	5793.903277
11	Quito	2016-09-07 08:07:34	-78.469686	-0.136556	-78.497103	-0.200596	965	10169.594365
12	Quito	2017-06-30 07:51:34	-78.470360	-0.130801	-78.482835	-0.170711	764	5825.002281
13	Quito	2017-06-30 08:27:10	-78.478410	-0.192965	-78.482544	-0.186297	152	1201.169041
14	Quito	2017-07-13 10:05:35	-78.547637	-0.261817	-78.537277	-0.249575	255	2513.179573

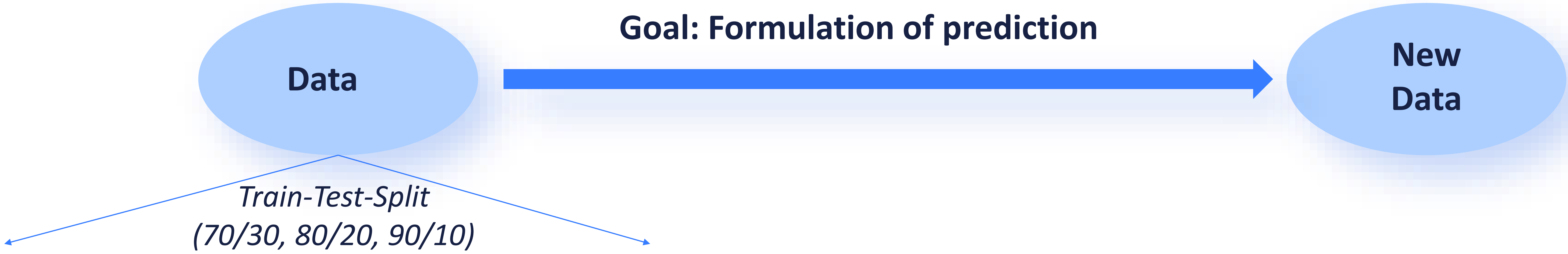
target_aim

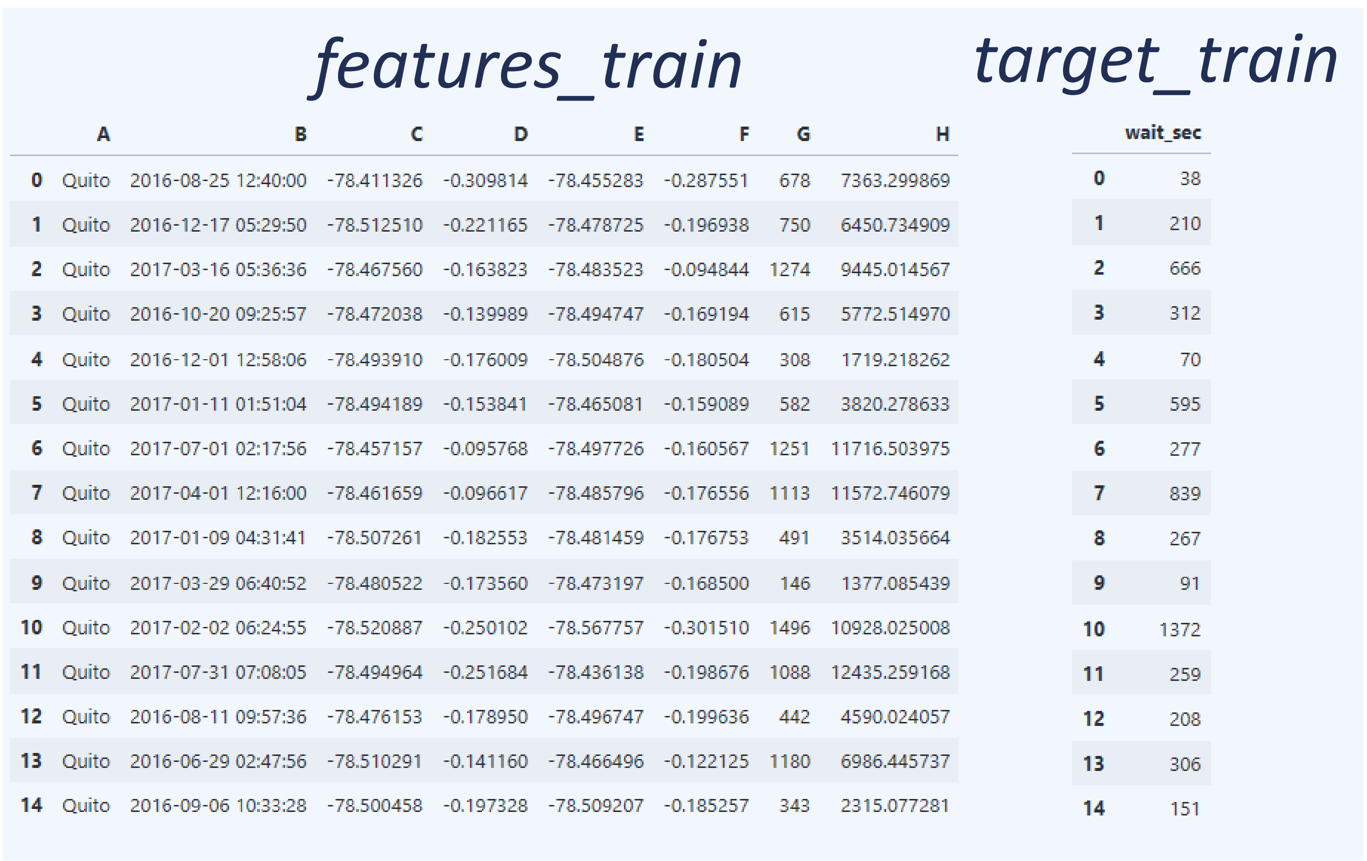


	wait_sec
0	?
1	?
2	?
3	?
4	?
5	?
6	?
7	?
8	?
9	?
10	?
11	?
12	?
13	?
14	?

How do we know how good our predictions are?

Basic approach in Machine Learning



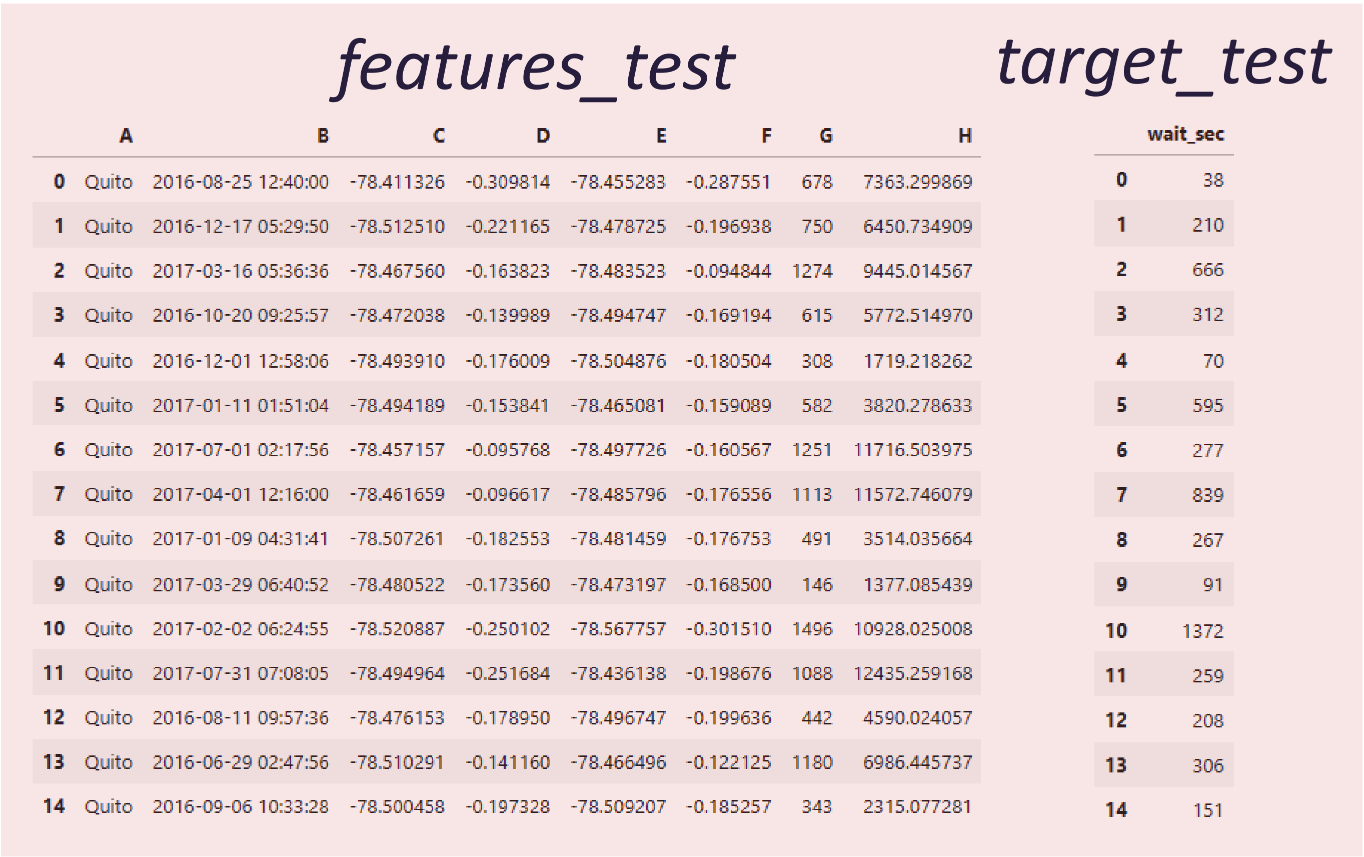


features_train

	A	B	C	D	E	F	G	H
0	Quito	2016-08-25 12:40:00	-78.411326	-0.309814	-78.455283	-0.287551	678	7363.299869
1	Quito	2016-12-17 05:29:50	-78.512510	-0.221165	-78.478725	-0.196938	750	6450.734909
2	Quito	2017-03-16 05:36:36	-78.467560	-0.163823	-78.483523	-0.094844	1274	9445.014567
3	Quito	2016-10-20 09:25:57	-78.472038	-0.139989	-78.494747	-0.169194	615	5772.514970
4	Quito	2016-12-01 12:58:06	-78.493910	-0.176009	-78.504876	-0.180504	308	1719.218262
5	Quito	2017-01-11 01:51:04	-78.494189	-0.153841	-78.465081	-0.159089	582	3820.278633
6	Quito	2017-07-01 02:17:56	-78.457157	-0.095768	-78.497726	-0.160567	1251	11716.503975
7	Quito	2017-04-01 12:16:00	-78.461659	-0.096617	-78.485796	-0.176556	1113	11572.746079
8	Quito	2017-01-09 04:31:41	-78.507261	-0.182553	-78.481459	-0.176753	491	3514.035664
9	Quito	2017-03-29 06:40:52	-78.480522	-0.173560	-78.473197	-0.168500	146	1377.085439
10	Quito	2017-02-02 06:24:55	-78.520887	-0.250102	-78.567757	-0.301510	1496	10928.025008
11	Quito	2017-07-31 07:08:05	-78.494964	-0.251684	-78.436138	-0.198676	1088	12435.259168
12	Quito	2016-08-11 09:57:36	-78.476153	-0.178950	-78.496747	-0.199636	442	4590.024057
13	Quito	2016-06-29 02:47:56	-78.510291	-0.141160	-78.466496	-0.122125	1180	6986.445737
14	Quito	2016-09-06 10:33:28	-78.500458	-0.197328	-78.509207	-0.185257	343	2315.077281

target_train

	wait_sec
0	38
1	210
2	666
3	312
4	70
5	595
6	277
7	839
8	267
9	91
10	1372
11	259
12	208
13	306
14	151

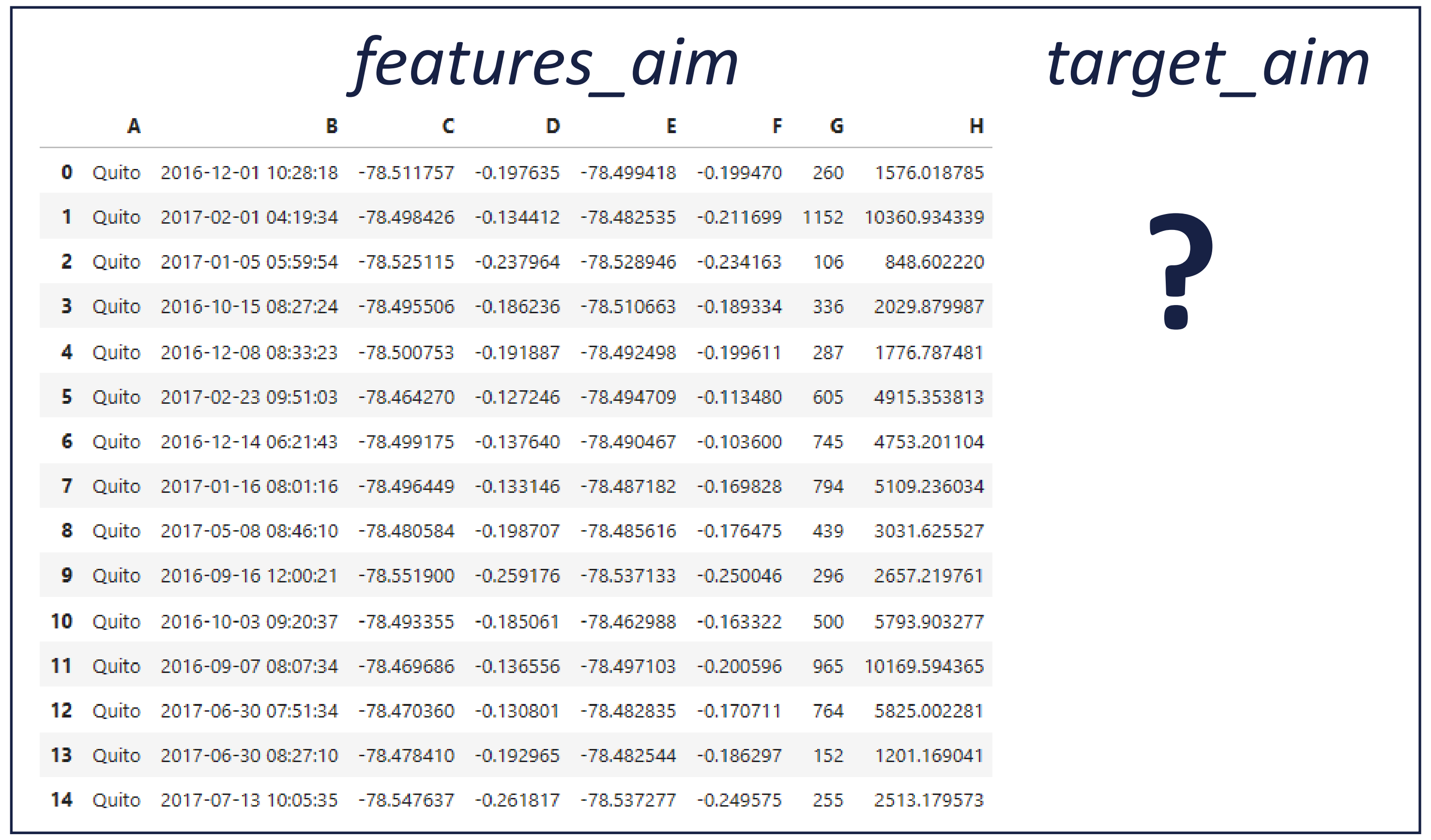


features_test

	A	B	C	D	E	F	G	H
0	Quito	2016-08-25 12:40:00	-78.411326	-0.309814	-78.455283	-0.287551	678	7363.299869
1	Quito	2016-12-17 05:29:50	-78.512510	-0.221165	-78.478725	-0.196938	750	6450.734909
2	Quito	2017-03-16 05:36:36	-78.467560	-0.163823	-78.483523	-0.094844	1274	9445.014567
3	Quito	2016-10-20 09:25:57	-78.472038	-0.139989	-78.494747	-0.169194	615	5772.514970
4	Quito	2016-12-01 12:58:06	-78.493910	-0.176009	-78.504876	-0.180504	308	1719.218262
5	Quito	2017-01-11 01:51:04	-78.494189	-0.153841	-78.465081	-0.159089	582	3820.278633
6	Quito	2017-07-01 02:17:56	-78.457157	-0.095768	-78.497726	-0.160567	1251	11716.503975
7	Quito	2017-04-01 12:16:00	-78.461659	-0.096617	-78.485796	-0.176556	1113	11572.746079
8	Quito	2017-01-09 04:31:41	-78.507261	-0.182553	-78.481459	-0.176753	491	3514.035664
9	Quito	2017-03-29 06:40:52	-78.480522	-0.173560	-78.473197	-0.168500	146	1377.085439
10	Quito	2017-02-02 06:24:55	-78.520887	-0.250102	-78.567757	-0.301510	1496	10928.025008
11	Quito	2017-07-31 07:08:05	-78.494964	-0.251684	-78.436138	-0.198676	1088	12435.259168
12	Quito	2016-08-11 09:57:36	-78.476153	-0.178950	-78.496747	-0.199636	442	4590.024057
13	Quito	2016-06-29 02:47:56	-78.510291	-0.141160	-78.466496	-0.122125	1180	6986.445737
14	Quito	2016-09-06 10:33:28	-78.500458	-0.197328	-78.509207	-0.185257	343	2315.077281

target_test

	wait_sec
0	38
1	210
2	666
3	312
4	70
5	595
6	277
7	839
8	267
9	91
10	1372
11	259
12	208
13	306
14	151



features_aim

	A	B	C	D	E	F	G	H
0	Quito	2016-12-01 10:28:18	-78.511757	-0.197635	-78.499418	-0.199470	260	1576.018785
1	Quito	2017-02-01 04:19:34	-78.498426	-0.134412	-78.482535	-0.211699	1152	10360.934339
2	Quito	2017-01-05 05:59:54	-78.525115	-0.237964	-78.528946	-0.234163	106	848.602220
3	Quito	2016-10-15 08:27:24	-78.495506	-0.186236	-78.510663	-0.189334	336	2029.879987
4	Quito	2016-12-08 08:33:23	-78.500753	-0.191887	-78.492498	-0.199611	287	1776.787481
5	Quito	2017-02-23 09:51:03	-78.464270	-0.127246	-78.494709	-0.113480	605	4915.353813
6	Quito	2016-12-14 06:21:43	-78.499175	-0.137640	-78.490467	-0.103600	745	4753.201104
7	Quito	2017-01-16 08:01:16	-78.496449	-0.133146	-78.487182	-0.169828	794	5109.236034
8	Quito	2017-05-08 08:46:10	-78.480584	-0.198707	-78.485616	-0.176475	439	3031.625527
9	Quito	2016-09-16 12:00:21	-78.551900	-0.259176	-78.537133	-0.250046	296	2657.219761
10	Quito	2016-10-03 09:20:37	-78.493355	-0.185061	-78.462988	-0.163322	500	5793.903277
11	Quito	2016-09-07 08:07:34	-78.469686	-0.136556	-78.497103	-0.200596	965	10169.594365
12	Quito	2017-06-30 07:51:34	-78.470360	-0.130801	-78.482835	-0.170711	764	5825.002281
13	Quito	2017-06-30 08:27:10	-78.478410	-0.192965	-78.482544	-0.186297	152	1201.169041
14	Quito	2017-07-13 10:05:35	-78.547637	-0.261817	-78.537277	-0.249575	255	2513.179573

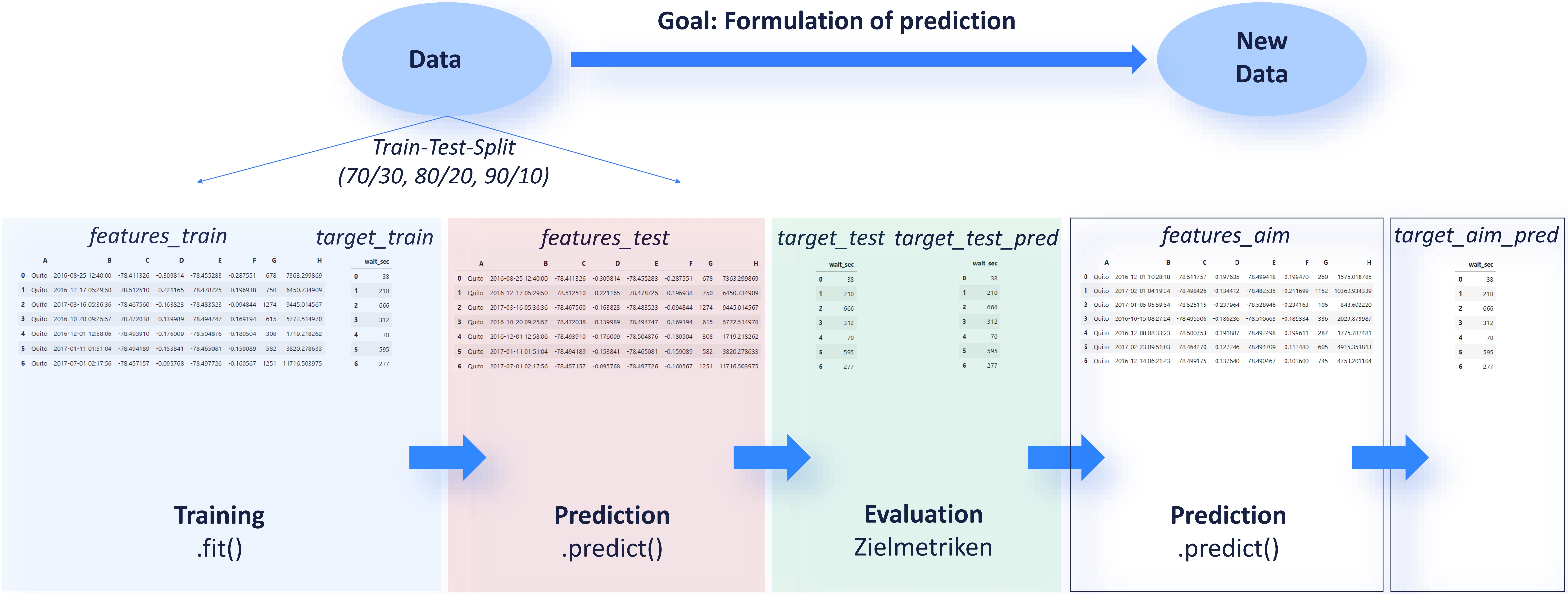
target_aim

?

- *Test data is trested like new, unknown data behandelt*
- *Information from test data is not allowed to inform the training process (Data Leakage)*

Data Analytics & Data Science Intensivkurs

Basic approach in Machine Learning



Data

Goal: Formulation of prediction

New Data

Train-Test-Split
(70/30, 80/20, 90/10)

features_train

	A	B	C	D	E	F	G	H
0	Quito	2016-08-25 12:40:00	-78.411326	-0.309814	-78.455283	-0.287551	678	7363.299869
1	Quito	2016-12-17 05:29:50	-78.512510	-0.221165	-78.478725	-0.196938	750	6450.734909
2	Quito	2017-03-16 05:36:36	-78.467560	-0.163823	-78.483523	-0.094844	1274	9445.014567
3	Quito	2016-10-20 09:25:57	-78.472038	-0.139989	-78.494747	-0.169194	615	5772.514970
4	Quito	2016-12-01 12:58:06	-78.493910	-0.176009	-78.504876	-0.180504	308	1719.218262
5	Quito	2017-01-11 01:51:04	-78.494189	-0.153841	-78.465081	-0.159089	582	3820.278633
6	Quito	2017-07-01 02:17:56	-78.457157	-0.095768	-78.497726	-0.160567	1251	11716.503975

target_train

	wait_sec
0	38
1	210
2	666
3	312
4	70
5	595
6	277

Training
.fit()

features_test

	A	B	C	D	E	F	G	H
0	Quito	2016-08-25 12:40:00	-78.411326	-0.309814	-78.455283	-0.287551	678	7363.299869
1	Quito	2016-12-17 05:29:50	-78.512510	-0.221165	-78.478725	-0.196938	750	6450.734909
2	Quito	2017-03-16 05:36:36	-78.467560	-0.163823	-78.483523	-0.094844	1274	9445.014567
3	Quito	2016-10-20 09:25:57	-78.472038	-0.139989	-78.494747	-0.169194	615	5772.514970
4	Quito	2016-12-01 12:58:06	-78.493910	-0.176009	-78.504876	-0.180504	308	1719.218262
5	Quito	2017-01-11 01:51:04	-78.494189	-0.153841	-78.465081	-0.159089	582	3820.278633
6	Quito	2017-07-01 02:17:56	-78.457157	-0.095768	-78.497726	-0.160567	1251	11716.503975

Prediction
.predict()

target_test

	wait_sec
0	38
1	210
2	666
3	312
4	70
5	595
6	277

target_test_pred

	wait_sec
0	38
1	210
2	666
3	312
4	70
5	595
6	277

Evaluation
Zielmetriken

features_aim

	A	B	C	D	E	F	G	H
0	Quito	2016-12-01 10:28:18	-78.511757	-0.197635	-78.499418	-0.199470	260	1576.018785
1	Quito	2017-02-01 04:19:34	-78.498426	-0.134412	-78.482535	-0.211699	1152	10360.934339
2	Quito	2017-01-05 05:59:54	-78.525115	-0.237964	-78.528946	-0.234163	106	848.602220
3	Quito	2016-10-15 08:27:24	-78.495506	-0.186236	-78.510663	-0.189334	336	2029.879987
4	Quito	2016-12-08 08:33:23	-78.500753	-0.191887	-78.492498	-0.199611	287	1776.787481
5	Quito	2017-02-23 09:51:03	-78.464270	-0.127246	-78.494709	-0.113480	605	4915.353813
6	Quito	2016-12-14 06:21:43	-78.499175	-0.137640	-78.490467	-0.103600	745	4753.201104

Prediction
.predict()

target_aim_pred

	wait_sec
0	38
1	210
2	666
3	312
4	70
5	595
6	277

Important challenges for Machine Learning

- Not enough volume of training data

  ML techniques need a great nummer of data points in order to work well. Even for simple tasks, thousands of examples are needed to learn. More complex tasks like image recognition even use millions of labeled data points.

- Training data is not representative

  If the training data is not representative for the abundance of real life situations the system might be confronted with, the model will make „bad“ preditions/decisions

- Data of inferior quality

  A lot of missing data, outliers, erronous data, noise (random and irrelevant shape of data, that are useless for prediction) impede pattern recognition

- Irrelevant features

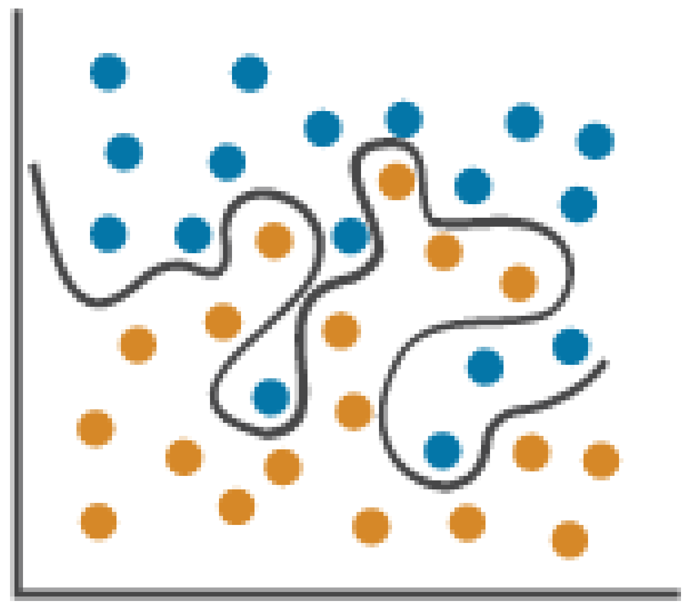
  Training data should be comprised of enough relevant and not too many irrelevant features. This is why feature selection and feature engineering are very important steps in the workflow.

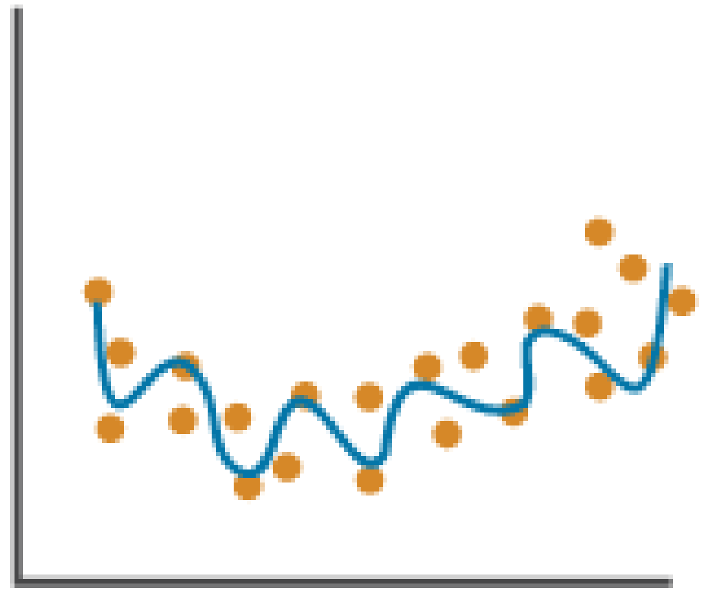
Important challenges for Machine Learning

- Overfitting

Model works **well on training data**, but **not on test or validation data**. This means that model is bad at generalization, too closley fitted to training data, not abstract enough.

Classification





Regression

Reasons might be:

- Not enough training data volume
- Too much noise/irrelavant information in training data
- Model complexity might be too high, learns noise in training data as if it were pattern

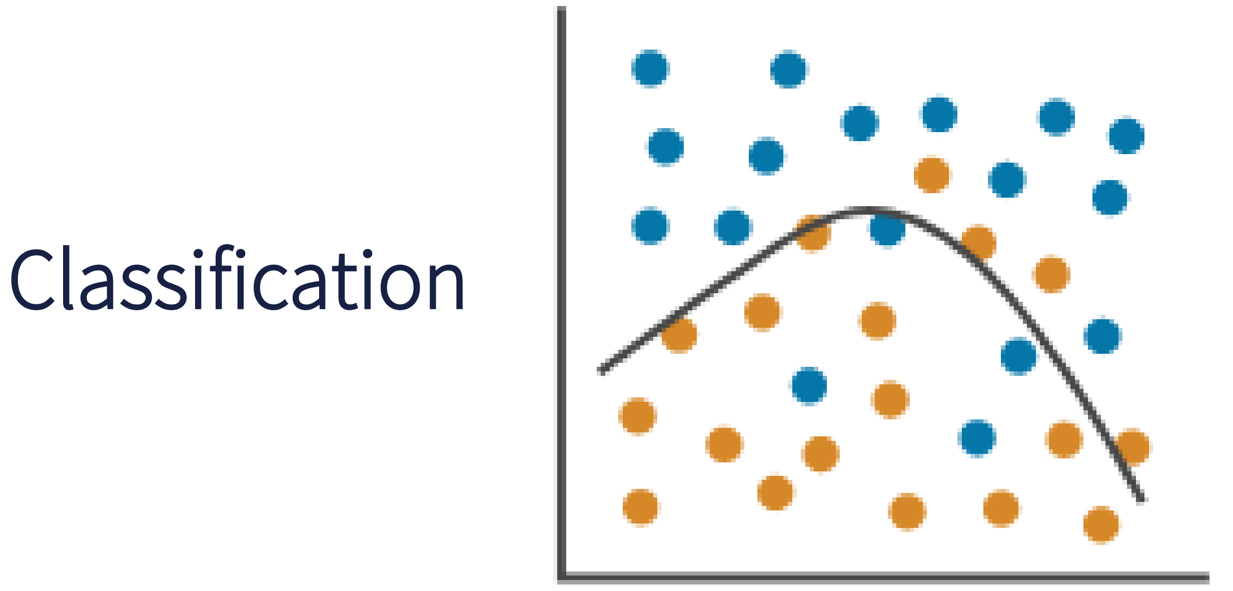
Important challenges for Machine Learning

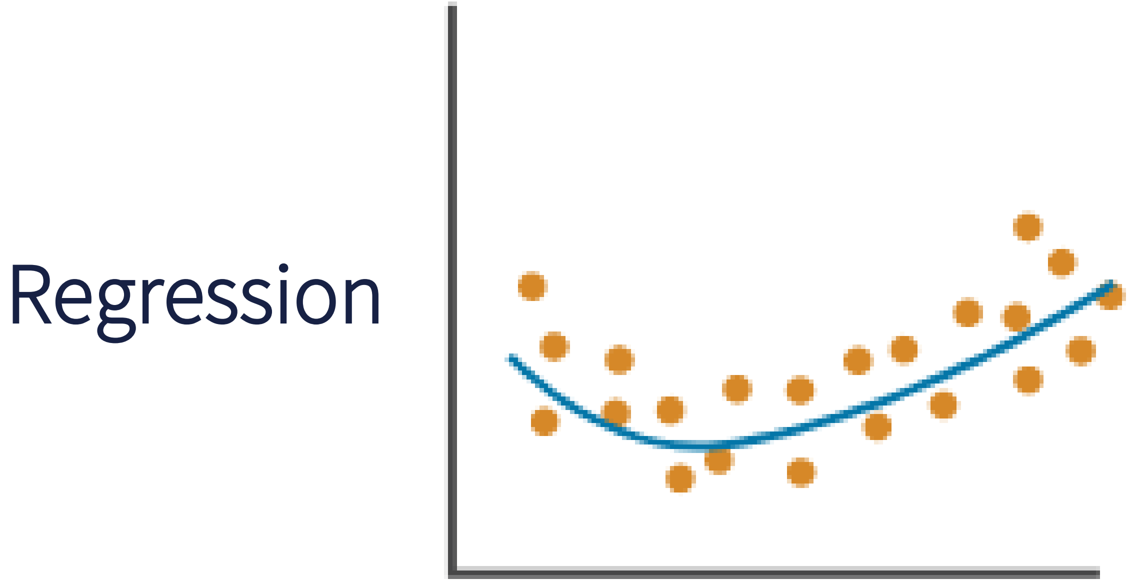
- Overfitting

How to avoid overfitting:

- Cross validation
- Higher training data volume
- Reduce noise in training data (errors, outliers, irrelevant features)
- Simplification of model:
  - Choice of simpler model (e.g. Linear instead of polynomial model)
  - Reduction/Selection of features
- Regularization (= restriction of model parameters)
  - Ridge-/ Lasso Regression
  - Early Stopping (ANN, Gradient-Boosted Decision Tress models like LightGBM, XGBoost, AdaBoost or CatBoost)
  - Pruning (identifying functions and parameters with (strong) effect on prediction)
- Ensembling: combination of different predictive models to get more accurate results (Bagging, Boosting, Voting, Stacking, Blending)

Appropriate fitting



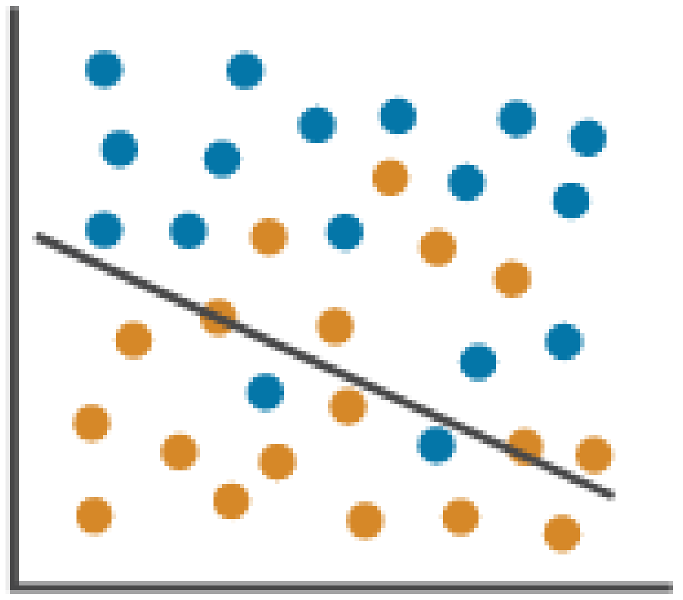


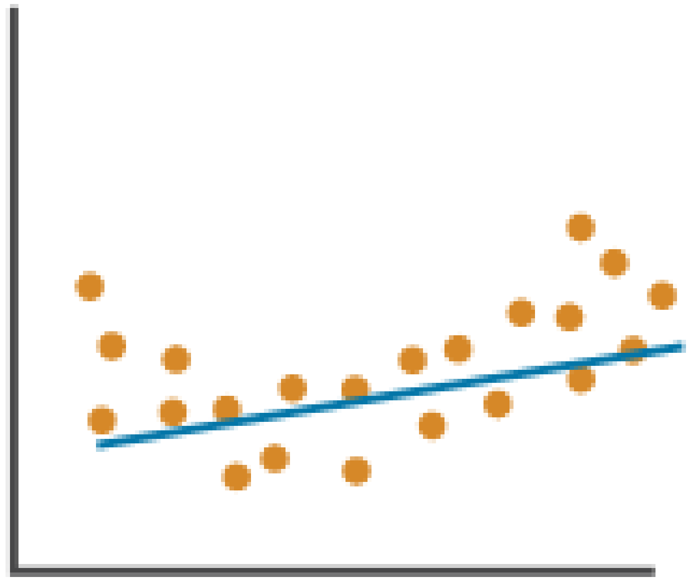
Important challenges for Machine Learning

- Underfitting

The modell **doesn‘t work well on either training nor test/ validation data**. This means that the model is not able to learn the given data structure/ recognize inherent patterns.

Classification





Regression

Reasons might be:

- Problem is represented too superficially (degree of simplification is too high)
- Too much noise or errors
- Model too strongly distorted, relation between features and target cannot be captures appropriately
- Model too simple (e.g. Linear model trained for complex scenarios)

Important challenges for Machine Learning

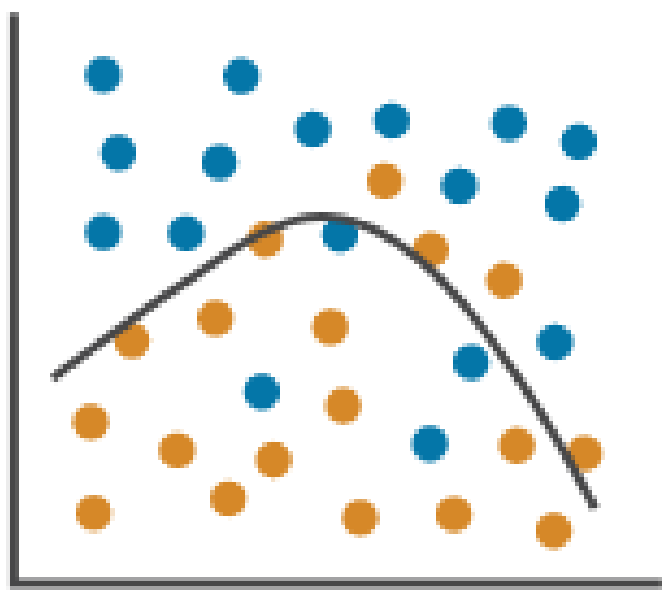
- Underfitting

How to avoid underfitting:

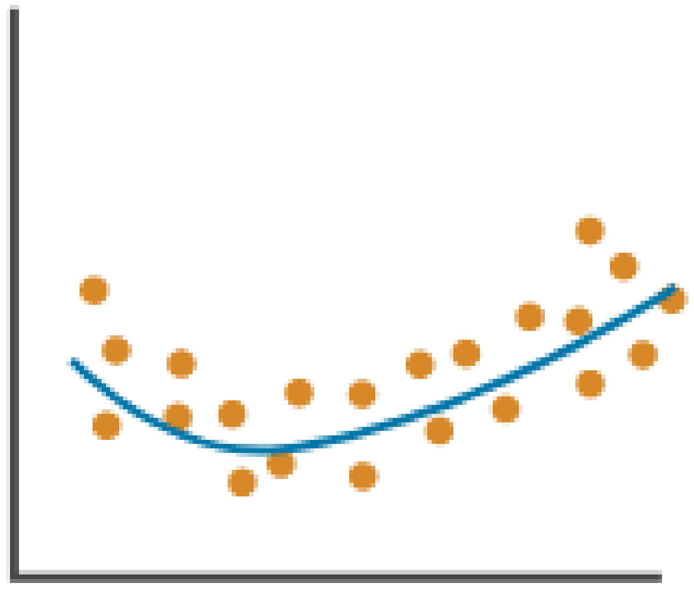
- Choose more complex model with more parameters
- Create more meaningful features (Feature Engineering)
- Reduce restrictions (e.g. Increase parameter space)

Appropriate fitting

Classification



Regression

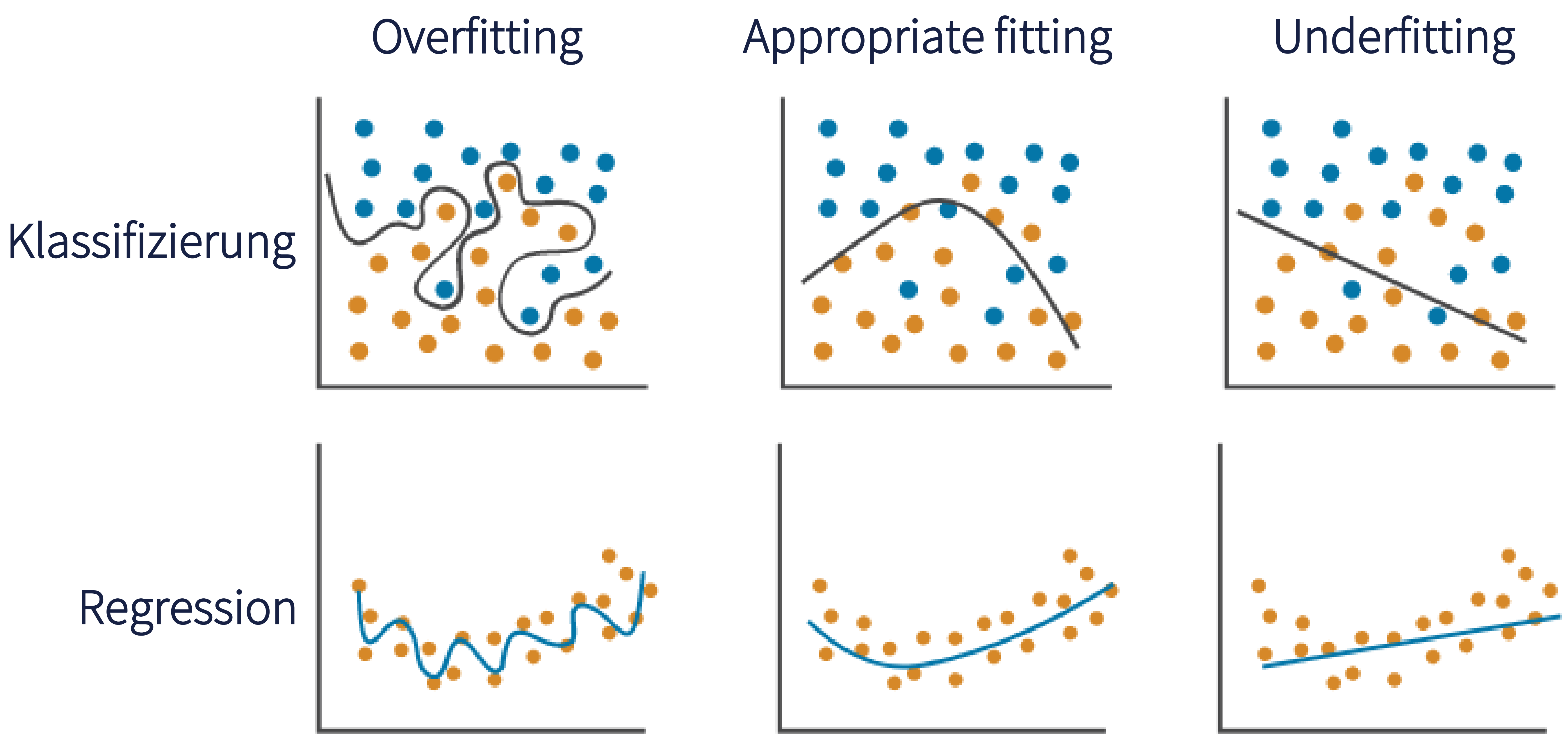


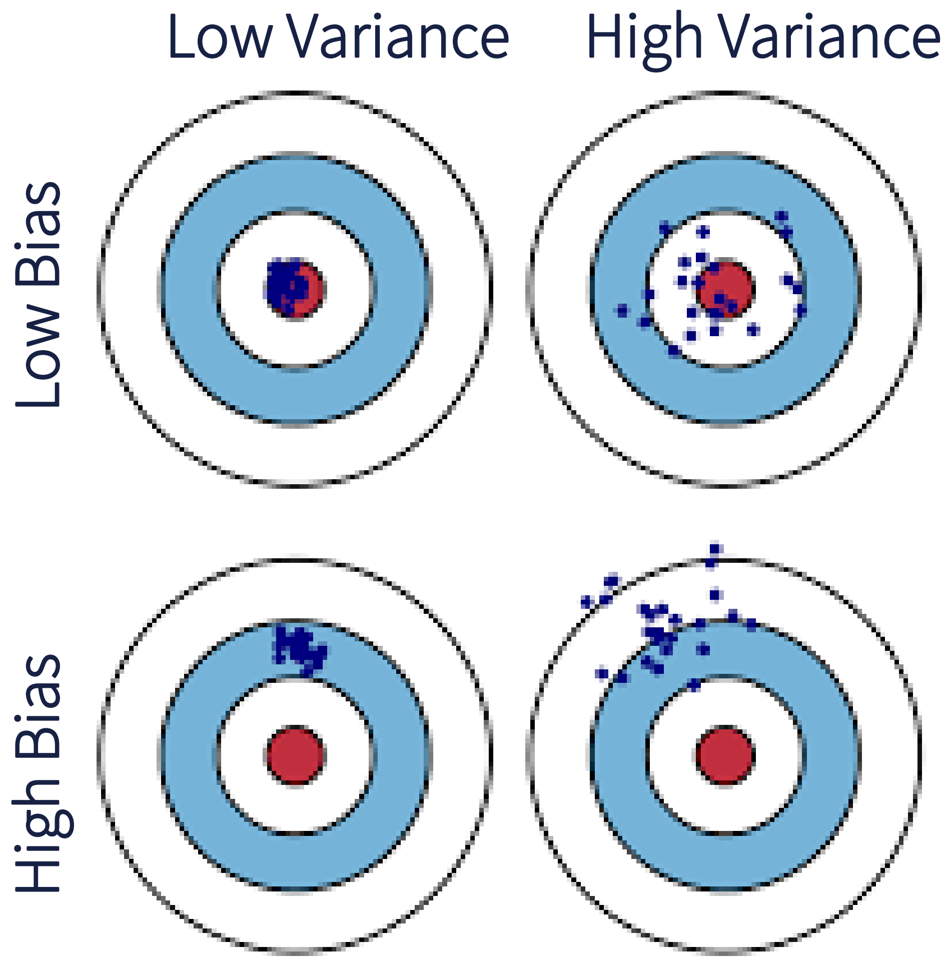
Bias-variance tradeoff

Underfitted models show a high bias – they produce inaccurate results for training data as well as test- and validation data.

Overfitted moels show high variance – they produce accurate results for training data, but not for test- and validation data..

Goal: balance between under – and overfitting + minimization of variance and bias.

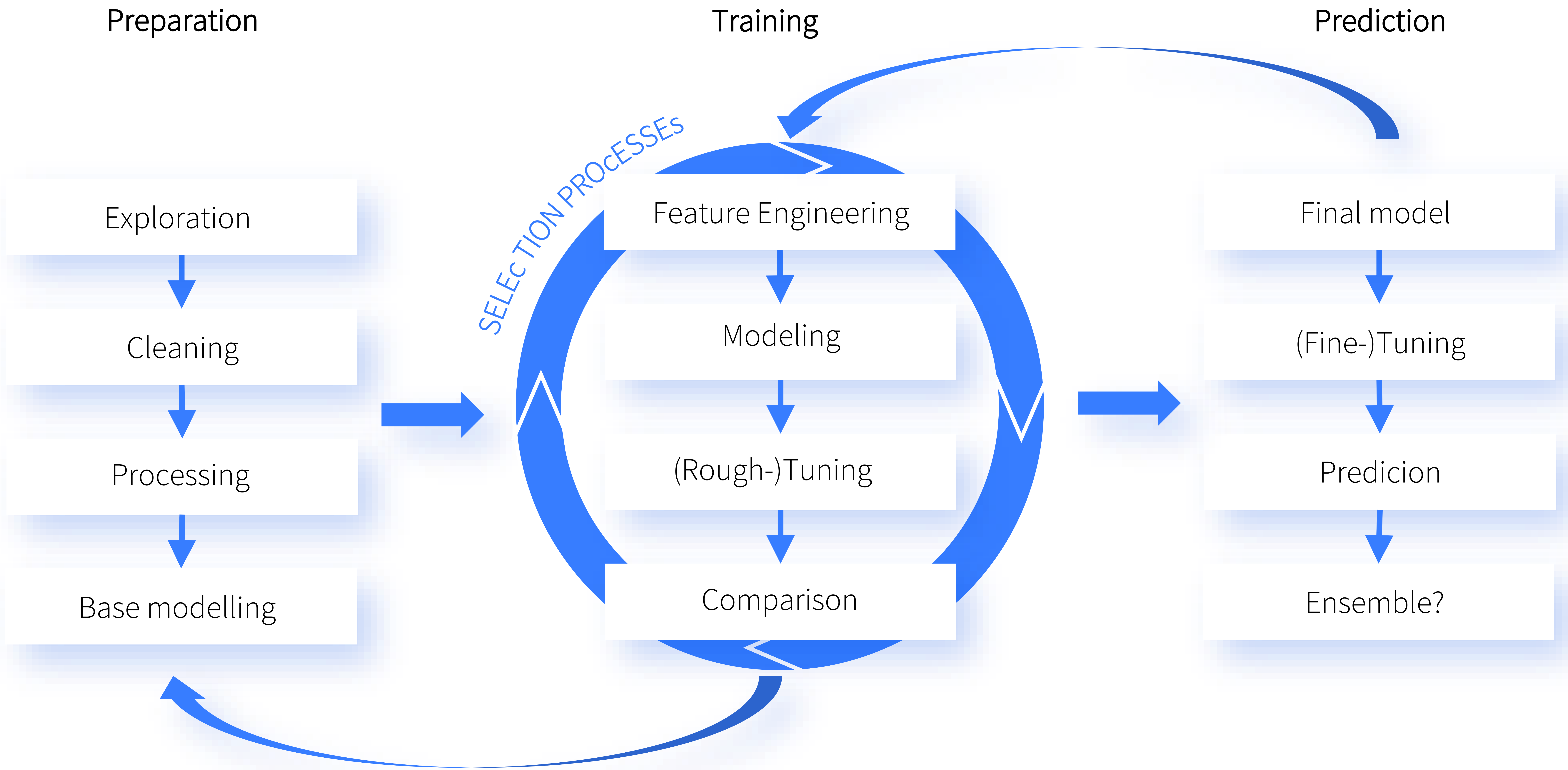




source:
https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote12.html

Fehler	Overfitting	Appropriate fitting	Underfitting
Training	Low	Low	High
Test	High	Low	High

ML Training is an iterative process



Summary

- Data Science is applicable in a great variation of fields
- Domain knowledge is vitally important for successful Data Science projects
- We split data into train and test sets, in order to be able to evaluate and improve our predictions
- Data cleaning is a big challenge
- Creation of prediction is iterative, not linear
- There are several possibilities of performance problems (over- or underfitting)

Literature recommendation

Data Science and Python:

- Vanderplas, J. (2016): Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly.
  -> Online verfügbar (inkl. Notebooks) auf dem GitHub-Account des Autors: https://jakevdp.github.io/PythonDataScienceHandbook/
- Müller, A. C./ Guido, S. (2016): Introduction to Machine Learning with Python. O'Reilly.
- Géron, A. (2020): Praxiseinstieg Machine-Learning mit Scikit-Learn, Keras und TensorFLow. O'Reilly.
- Gallatin, K./ Albon, C. (2023): Machine Learning with Python Cookbook: Practical Solutions from Preprocessing to Deep Learning. O'Reilly.
  -> gibt auch Auflage von 2018 mit Albon, C. als alleinigen Autor.
- James, G. et al. (2023): An Introduction to Statistical Learning with Applications in Python. Springer.
  -> verfügbar unter: https://www.statlearning.com/
- Ng, A. (2018): Machine Learning Yearning. Technical Strategy for AI Engineers in the Era of Deep Learning. Verfügbar unter: https://github.com/ajaymache/machine-learning-yearning/blob/master/full%20book/machine-learning-yearning.pdf

Literature recommendation

Platforms for ML with Python:

- stackoverflow.com
- machinelearningmastery.com
- analyticsvidhya.com

specialist journals:

- towardsdatascience.com
- medium.com

Advanced text books for mathematical/ statistical Basics:

- Grus, J. (2019): Data Science from Scratch. O'Reilly.
- Russel, S./ Norvig, P. (2021): Artificial Intelligence: A Modern Approach. Pearson Series.