Data Science and Machine Learning in Python

Stephan Weyers

Fachhochschule Dortmund
University of Applied Sciences and Arts

Topics covered in the online lectures

Fachhochschule Dortmund
University of Applied Sciences and Arts

Part 1: Data Science

	Date	Topics covered
1	Apr 13th	Course introduction Data Science motivation How to use Jupyter Notebook Python types and lists Loops, if/else, functions
2	Apr 20th	Python tuples, lists, dictionaries Functions Numpy basics, operations Image processing
3	Apr 27th	Pandas Series, DataFrame Pandas basic operations Import/export files
4	May 4th	Principles of data visualization Data cleaning and preparation Join, combine and reshape data
5	May 11th	Volkswohl Bund dataset Data visualization in Python How to write Data Science reports Data aggregation and grouping

Part 2: Machine Learning

	Date	Topics covered
6	Jun 1st	Introduction to supervised learning Classification and regression scikit-learn k-Nearest Neighbors Linear regression (ridge and lasso)
7	Jun 8th	Linear classification models Decision trees Random forests and gradient boosting Support vector machines Neural networks
8	Jun 15th	Introduction to unsupervised learning Preprocessing and scaling Dimensionality reduction Principal component analysis
9	Jun 22nd	k-means clustering Hierarchical clustering DBSCAN
10	Jun 29th	Representing data Engineering features
11	Jul 6th	Model evaluation and improvement Text data analysis

Agenda for online lecture 6

Fachhochschule Dortmund
University of Applied Sciences and Arts

Session	Topic	Mode	Materials used	Minutes	End
14:30-16:00	Organizational questions	Q&A		10	14:40
	Supermarket exercise	Team work in break-out rooms	Lecture 06a notebook	40	15:20
	k-Nearest Neighbors	Lecture / Q&A	Lecture slides	15	15:35
	Linear regression	Lecture / Q&A	Lecture slides	20	15:55
16:10-17:40	Regression toy data	Lecture / Q&A	Lecture 06b notebook	20	16:30
	California housing data	Team work in break-out rooms	Lecture 06c notebook	45	17:15
	Recap W02 / W03	Lecture / Q&A		20	17:35
17:50-19:20	Teams W04	Lecture / Q&A		10	18:00
	Happiness data	Team work in break-out rooms	Lecture 06d notebook	80	19:20

Types of problems

Fachhochschule Dortmund
University of Applied Sciences and Arts

Supervised Approaches

- Labeled data
- Target values known

Classification

- Predict category

Regression

- Predict numeric value

Unsupervised Approaches

- Unlabeled data
- No target value provided

Cluster Analysis

- Organize similar cases into segments

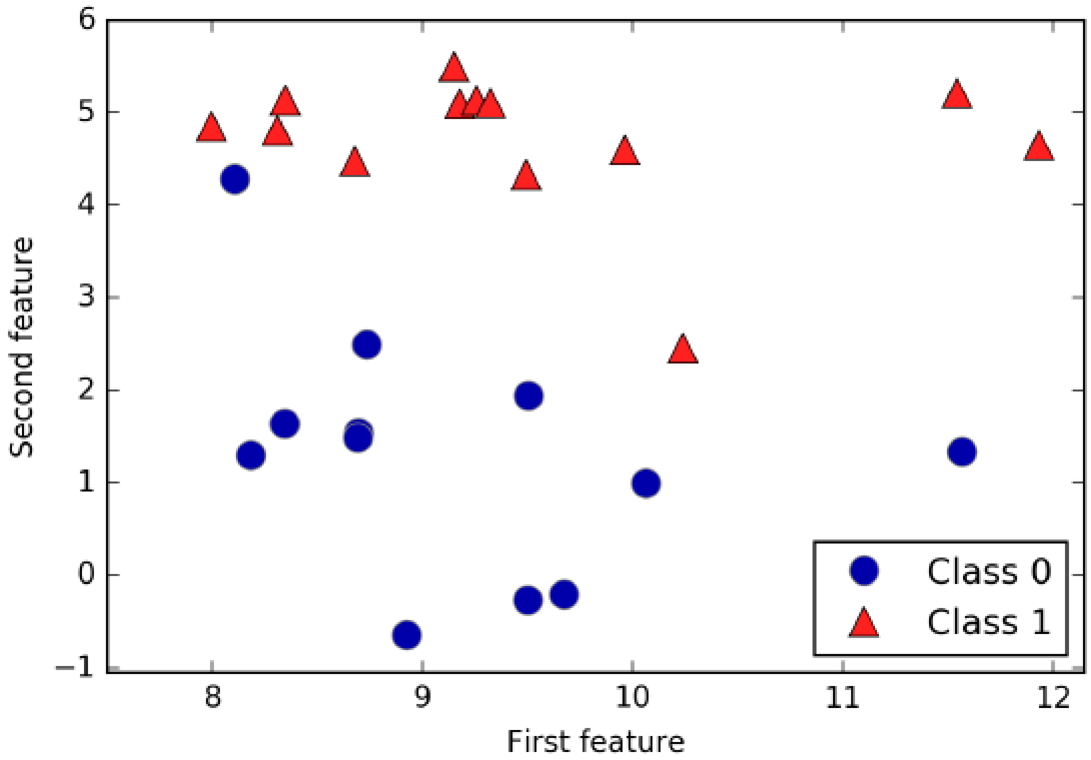
Dimensionality reduction

- Reduce number of features

Question for discussion

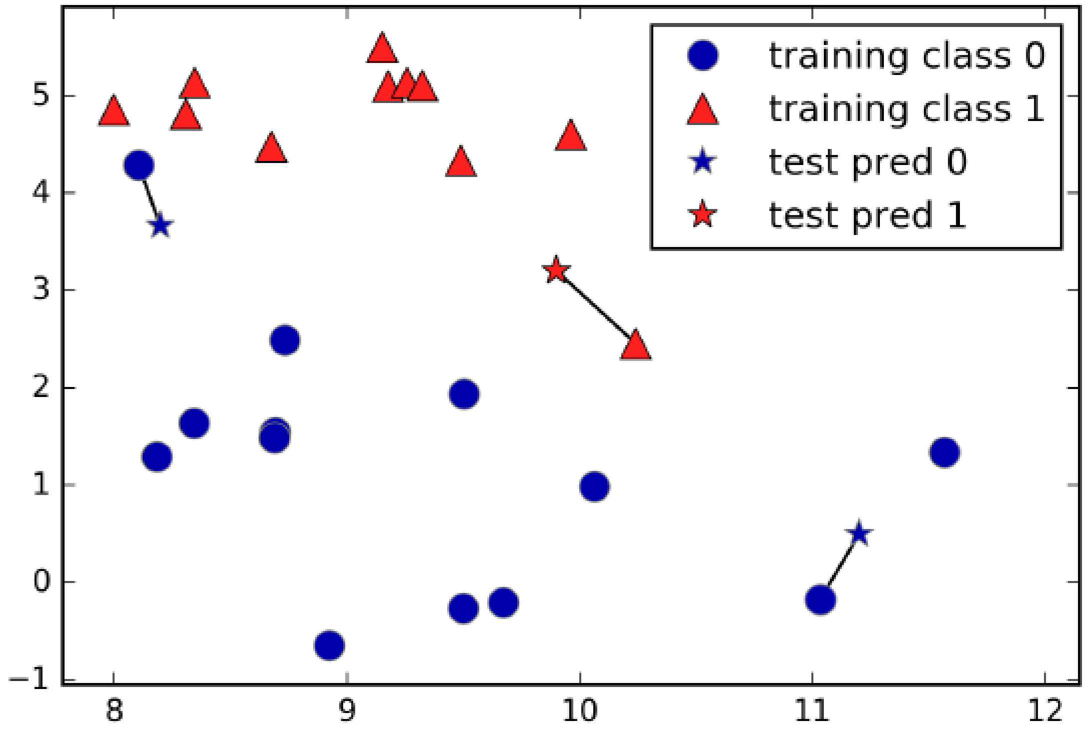
- Find examples for each of the 4 categories

k-Nearest Neighbors – example



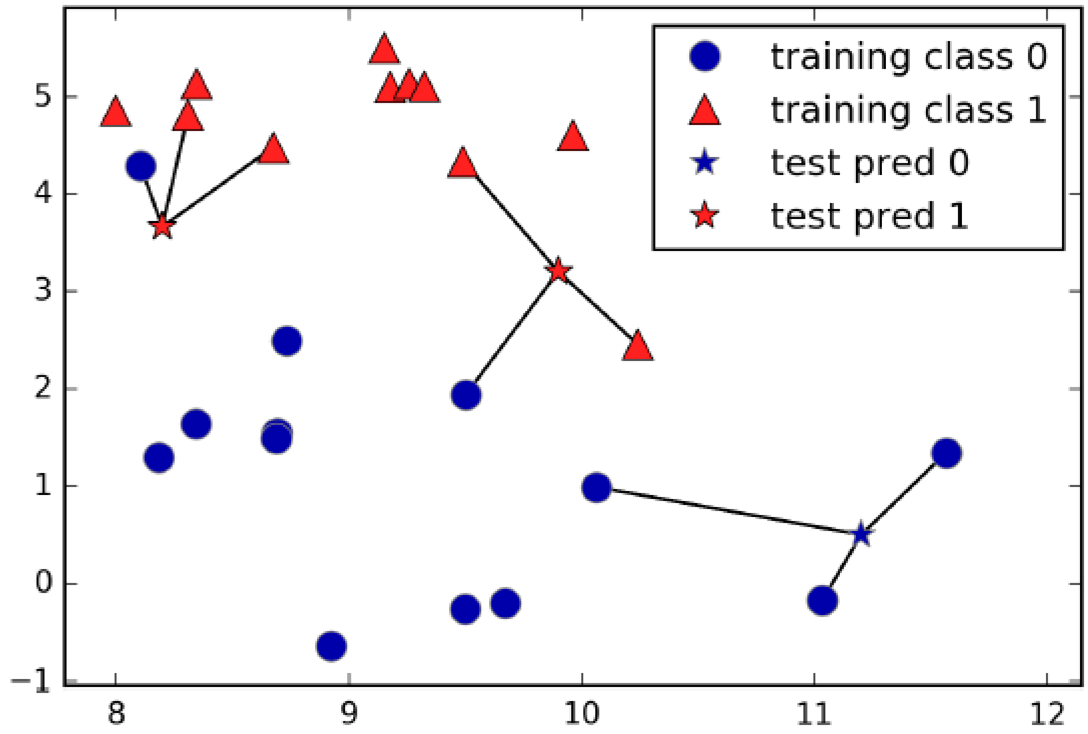
Source: Müller, A. C., Guido, S. (2016) Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media

k-Nearest Neighbors – predictions 1-NN



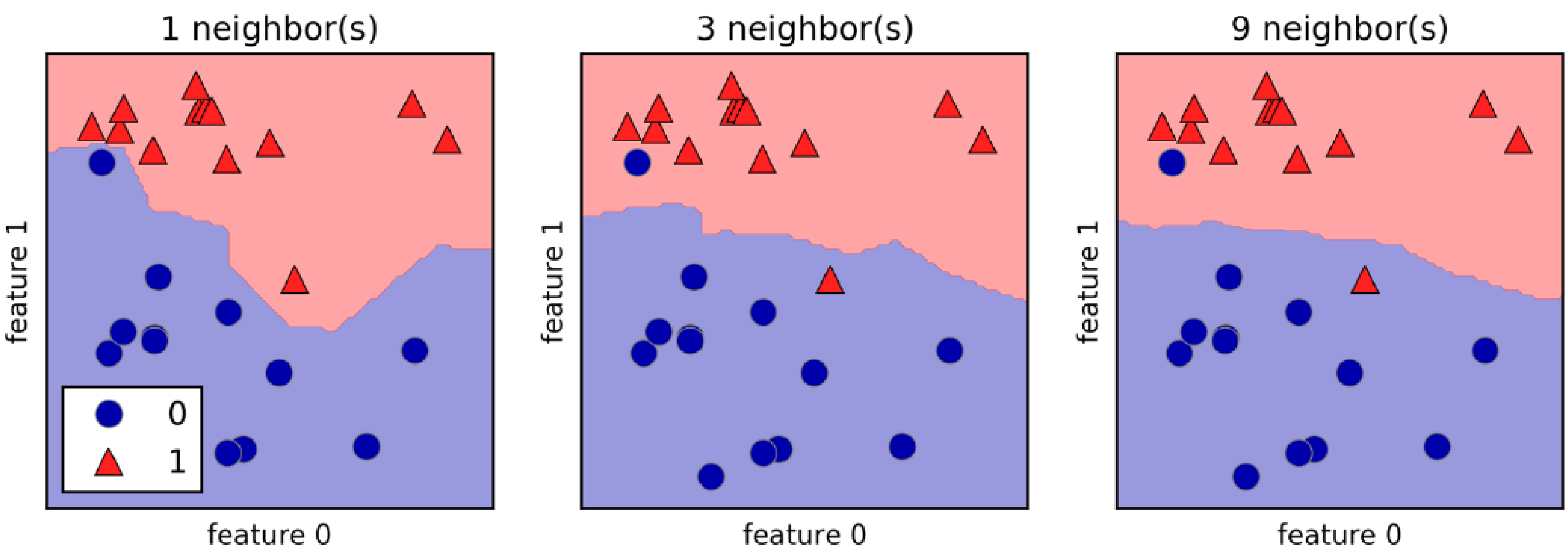
Source: Müller, A. C., Guido, S. (2016) Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media

k-Nearest Neighbors – predictions 3-NN



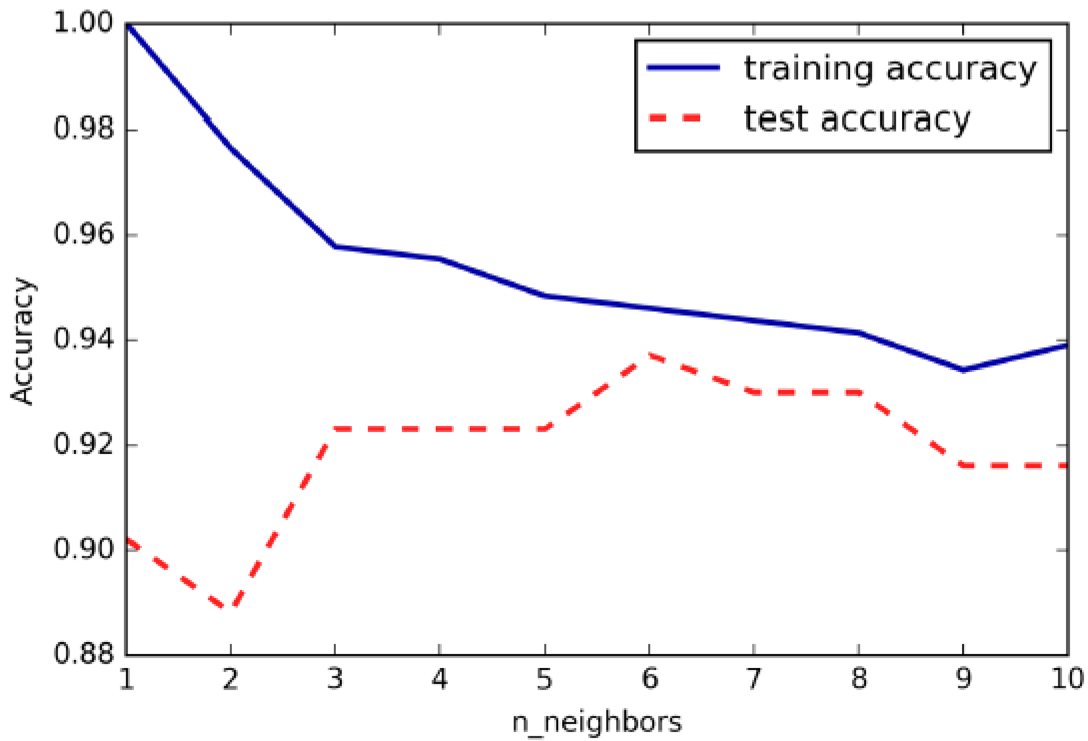
Source: Müller, A. C., Guido, S. (2016) Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media

k-Nearest Neighbors – Decision Boundary



Source: Müller, A. C., Guido, S. (2016) Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media

k-Nearest Neighbors – Accuracy



Source: Müller, A. C., Guido, S. (2016) Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media

k-Nearest Neighbors – Summary

Parameters

- Number of neighbors k
- Distance metric (Euklidean as default)

Strengths

- Easy to understand
- Building model is fast

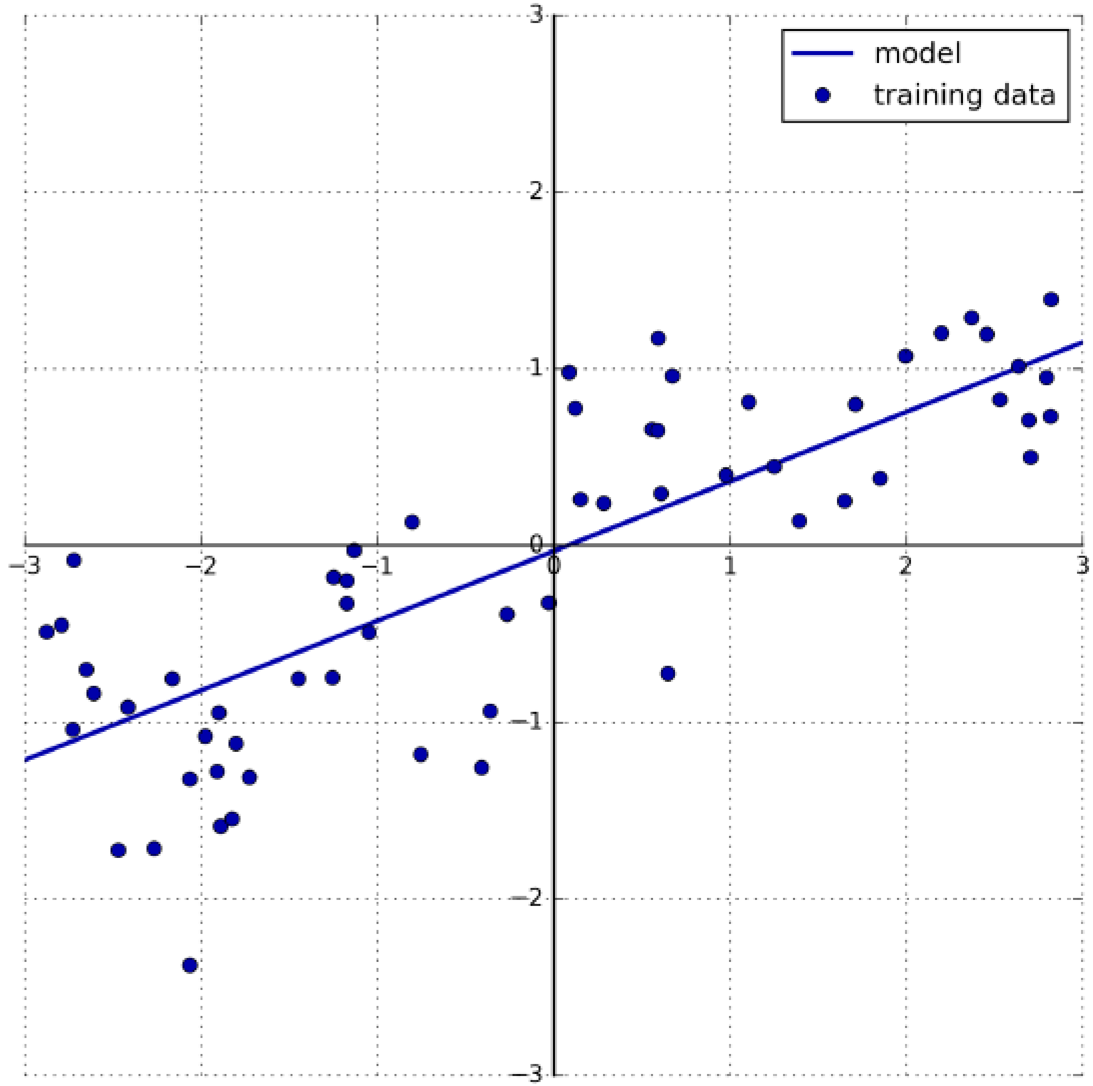
Weaknesses

- Making predictions is very slow on large datasets
- Usually not good with many features (hundreds or more)
- Particularly bad with sparse datasets (many zeros)
- Not robust if features are on different scales

Source: Müller, A. C., Guido, S. (2016) Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media

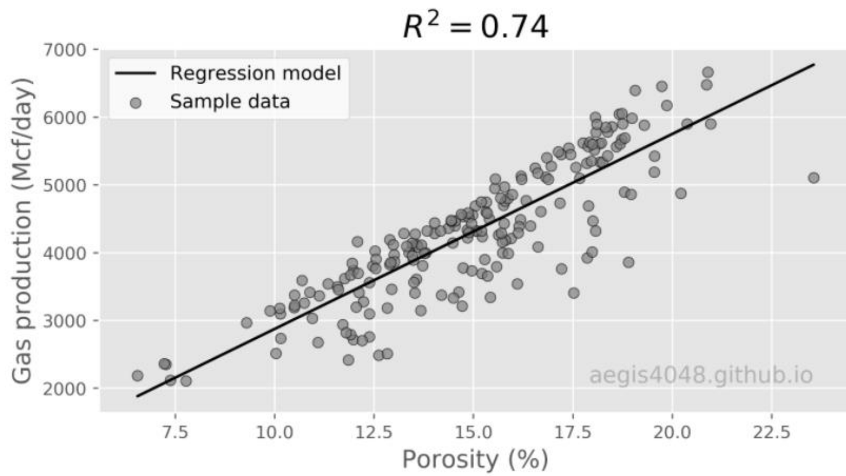
Linear Regression – one input variable

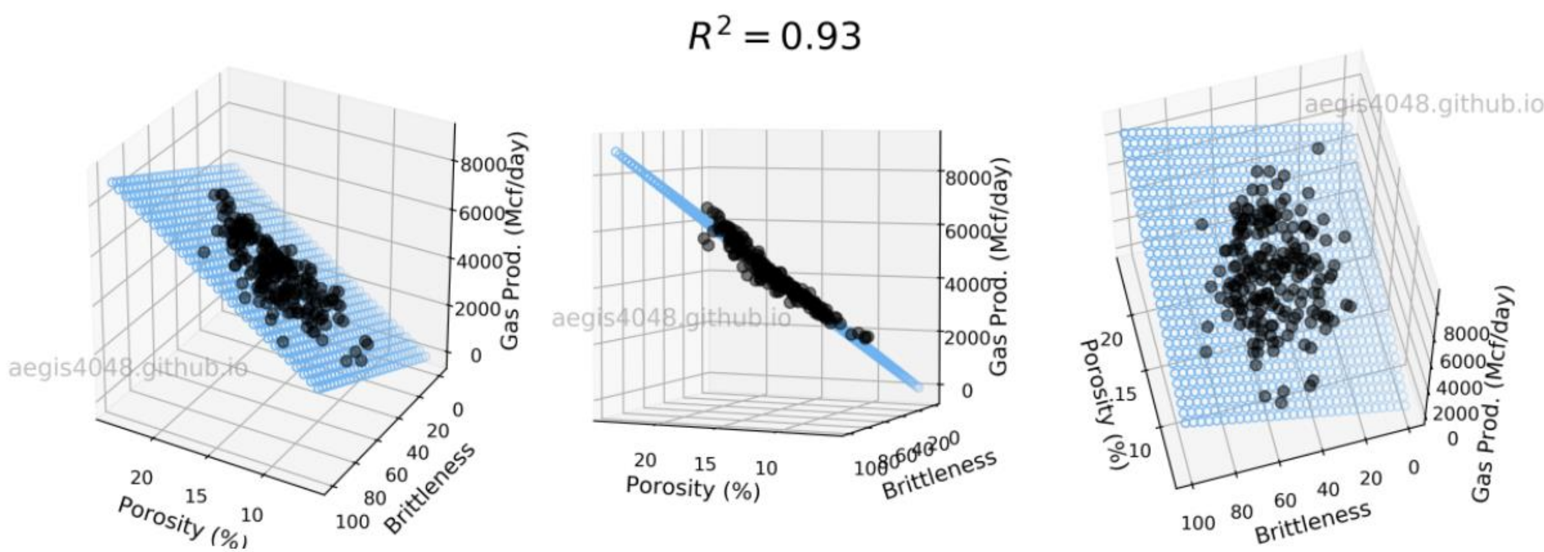
Fachhochschule Dortmund
University of Applied Sciences and Arts



Source: Müller, A. C., Guido, S. (2016) Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media

Linear Regression – two input variables





Source: https://aegis4048.github.io/mutiple_linear_regression_and_visualization_in_python

Linear Regression – Formulas

Given training input data $\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(n)} \in \mathbb{R}^{\boldsymbol{d}}$ with labels $\boldsymbol{y}^{(1)}, \dots, \boldsymbol{y}^{(n)} \in \mathbb{R}$.

Try to find $\boldsymbol{w} \in \mathbb{R}^{\boldsymbol{d}}$ and $\boldsymbol{b} \in \mathbb{R}$ in order to minimize

Linear regression (ordinary least squares)

$$\boldsymbol{L}(\boldsymbol{w}, \boldsymbol{b}) = \sum_{\boldsymbol{i}=1}^{\boldsymbol{n}} \left(\boldsymbol{y}^{(i)} - \left(\boldsymbol{w} \cdot \boldsymbol{x}^{(i)} + \boldsymbol{b}\right)\right)^2$$

Ridge regression

$$\boldsymbol{L}_2(\boldsymbol{w}, \boldsymbol{b}) = \sum_{\boldsymbol{i}=1}^{\boldsymbol{n}} \left(\boldsymbol{y}^{(i)} - \left(\boldsymbol{w} \cdot \boldsymbol{x}^{(i)} + \boldsymbol{b}\right)\right)^2 + \boldsymbol{C}\|\boldsymbol{w}\|_2^2$$

Lasso regression

$$\boldsymbol{L}_1(\boldsymbol{w}, \boldsymbol{b}) = \sum_{\boldsymbol{i}=1}^{\boldsymbol{n}} \left(\boldsymbol{y}^{(i)} - \left(\boldsymbol{w} \cdot \boldsymbol{x}^{(i)} + \boldsymbol{b}\right)\right)^2 + \boldsymbol{C}\|\boldsymbol{w}\|_1$$

Linear Models – Summary

Parameters

- Regularization parameter alpha and C
- Model type lasso vs. ridge for regression / logistic vs. SVM for classification

Strengths

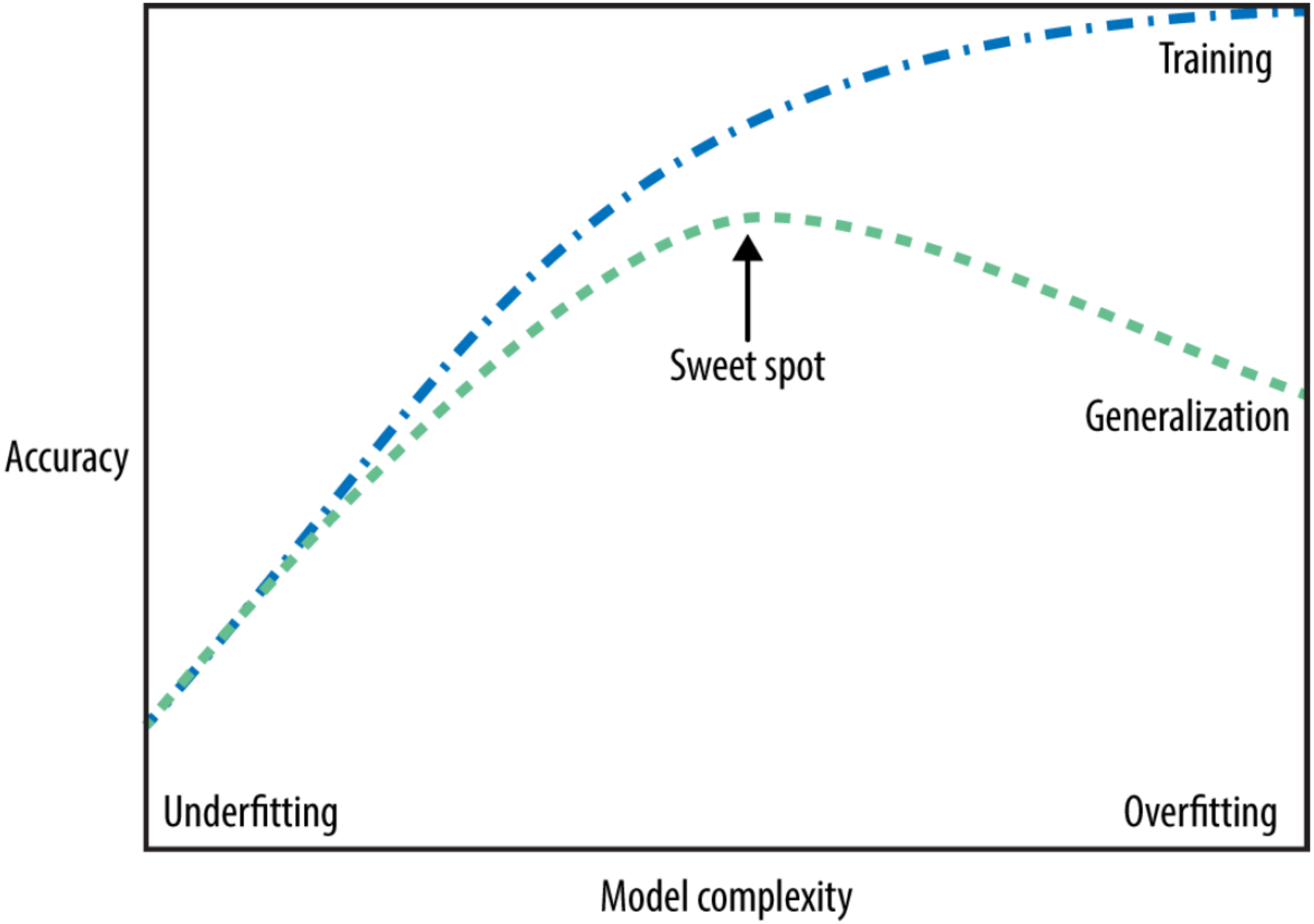
- Fast to train, fast to predict
- Work well with sparse data
- Relatively easy to understand how predictions are made
- Work well with large number of features

Weaknesses

- Coefficients hard to interpret, especially if features are highly correlated
- Sometimes fail with small datasets
- Perform bad with non-linear features and datasets that are not linearly separable

Source: Müller, A. C., Guido, S. (2016) Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media

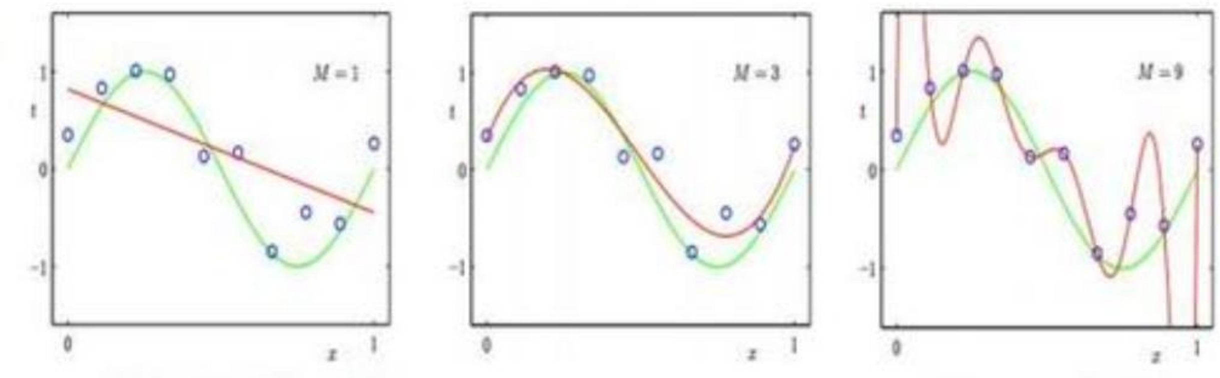
Machine Learning – Training and Test Error



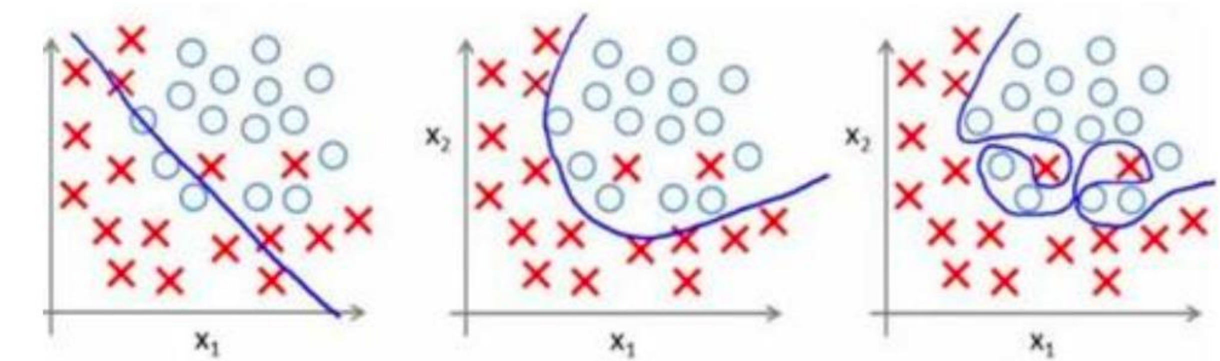
Source: Müller, A. C., Guido, S. (2016) Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media

Under- and Overfitting

Regression



Classi-
fication



Low model complexity (Underfitting)

High model complexity (Overfitting)

Source: Prof. Dr. Michael Bücker – Data Mining