Advanced Data Analysis

DATA 71200

Class 7

Supervised Learning

- **Learning paradigm where we have example input/output pairs to learn from**
- **Classification predicts a class label**
  - Binary example: is this email spam? yes/no
  - Multiclass example: what is the species of this flower?
- **Regression predicts a *continuous* number**
  - Example: what is the value of a house given a set of features?

Generalization

- **Generalization - a model's ability to make accurate predictions on unseen data**
- **Typically we want to find the simplest effective model**
- **Model complexity**
    - **Underfitting - when a model is too simple to represent the training data**
    - **Overfitting - when a model is too specific to the training data to generalize to new data**

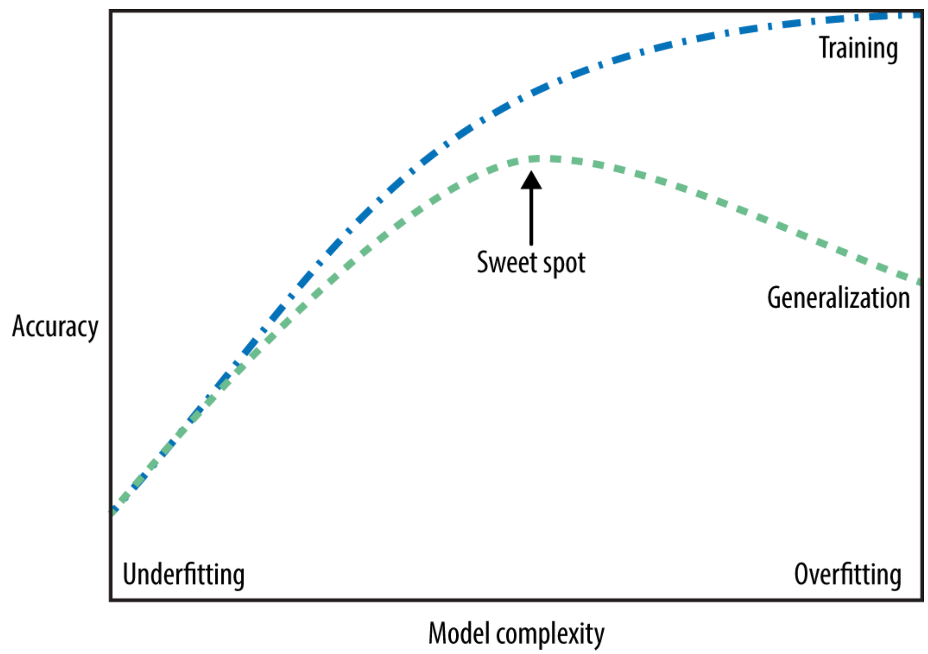

Figure 2-1. Trade-off of model complexity against training and test accuracy

Guido, Sarah and Andreas C. Muller. (2016). Introduction to Machine Learning with Python, O'Reilly Media, Inc.

k-Nearest Neighbors Classification

- Classification of unlabeled points based on the closest labeled point(s)
- *k* is the number of points considered to determine the class of an unlabeled point
- When *k* = 1 the class of the nearest point

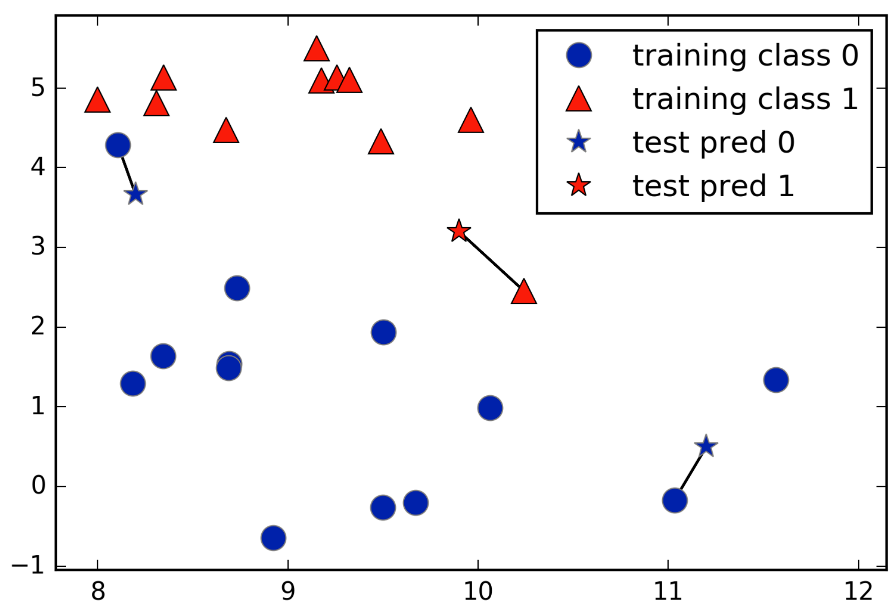

Figure 2-4. Predictions made by the one-nearest-neighbor model on the forge dataset

Jupyter Notebook 02-supervised-learning .ipynb [10]

Guido, Sarah and Andreas C. Muller. (2016). Introduction to Machine Learning with Python, O'Reilly Media, Inc.

k-Nearest Neighbors Classification

- **When k > 1 the algorithm uses a simple voting paradigm**
  - Assigned class label is the most frequent label among the neighbors
  - *k* should be larger than the number of classes
  - *k* should not be a multiple of the number of classes

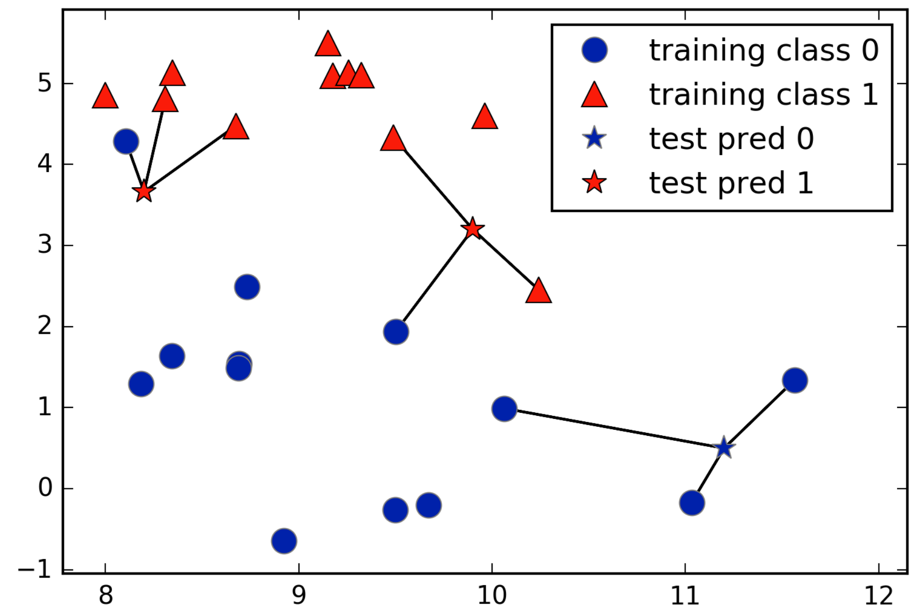

Figure 2-5. Predictions made by the three-nearest-neighbors model on the forge dataset

Jupyter Notebook 02-supervised-learning .ipynb [11–16]

Guido, Sarah and Andreas C. Muller. (2016). Introduction to Machine Learning with Python, O'Reilly Media, Inc.

Size of *k*

- When *k* = 1 the decision boundary follows the the training data
- As *k* increases the decision boundary becomes smoother (but does not fit the training data as well)

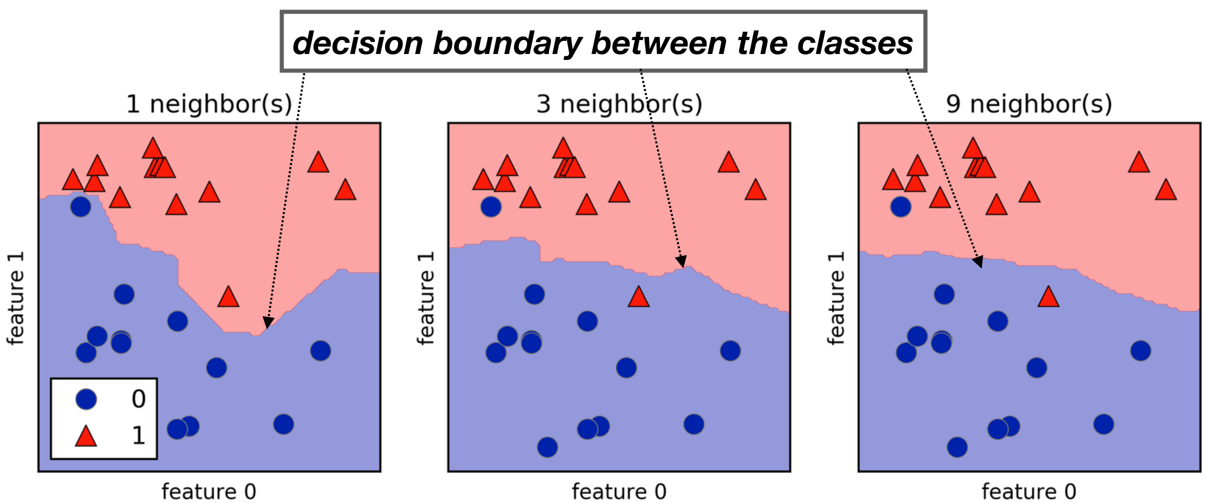

Figure 2-6. Decision boundaries created by the nearest neighbors model for different values of n_neighbors

Jupyter Notebook 02-supervised-learning .ipynb [17]

Guido, Sarah and Andreas C. Muller. (2016). Introduction to Machine Learning with Python, O'Reilly Media, Inc.

Size of *k*

- **When *k* is too small the model is too complex and tends to overfit**
- **When *k* is too large the model is too simple and tends to underfit**

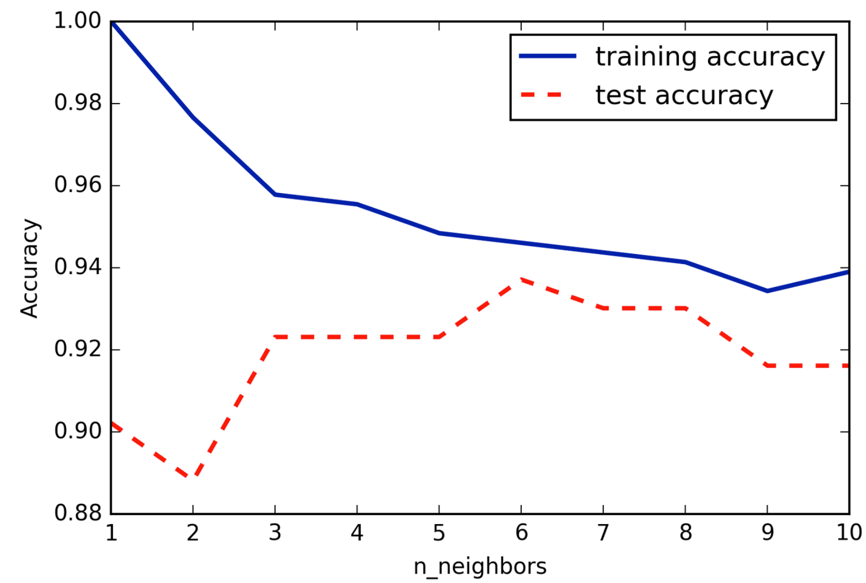

Figure 2-7. Comparison of training and test accuracy as a function of n_neighbors

Jupyter Notebook 02-supervised-learning .ipynb [18]

Guido, Sarah and Andreas C. Muller. (2016). Introduction to Machine Learning with Python, O'Reilly Media, Inc.

k-Nearest Neighbors Regression

- When k = 1 the value of the nearest point determines the value of the unlabeled point

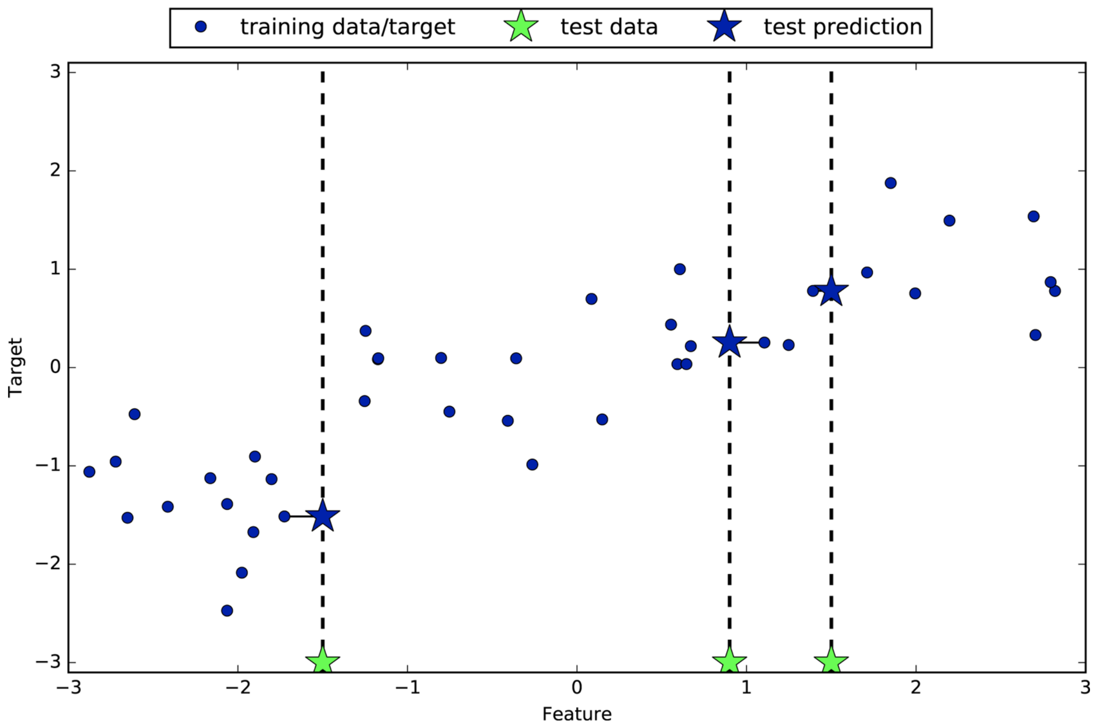

Figure 2-8. Predictions made by one-nearest-neighbor regression on the wave dataset

Jupyter Notebook 02-supervised-learning .ipynb [19]

Guido, Sarah and Andreas C. Muller. (2016). Introduction to Machine Learning with Python, O'Reilly Media, Inc.

k-Nearest Neighbors Regression

- **When *k* > 1 the mean of the values of the *k* nearest points determines the value of the unlabeled point**

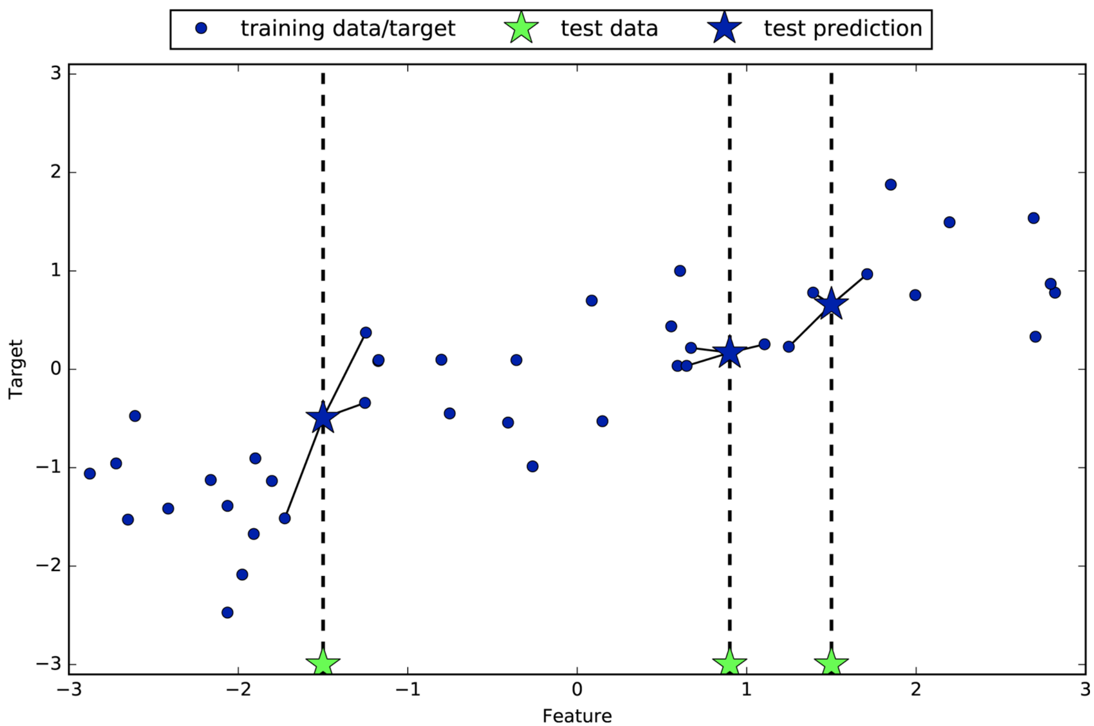

Figure 2-9. Predictions made by three-nearest-neighbors regression on the wave dataset

Jupyter Notebook 02-supervised-learning .ipynb [20–23]

Guido, Sarah and Andreas C. Muller. (2016). Introduction to Machine Learning with Python, O'Reilly Media, Inc.

k-Nearest Neighbors Regression

- When *k* = 1 the prediction is heavily influenced by the value of the nearest neighbor
- As *k* increases the predictions become smoother (but do not fit the training data as well)

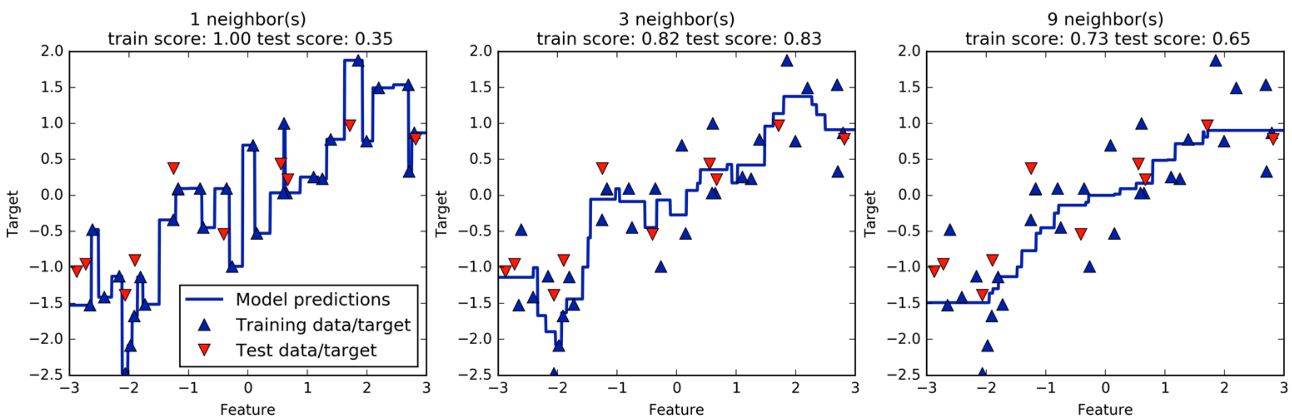

Figure 2-10. Comparing predictions made by nearest neighbors regression for different values of `n_neighbors`

Jupyter Notebook 02-supervised-learning.ipynb [24]

Guido, Sarah and Andreas C. Muller. (2016). Introduction to Machine Learning with Python, O'Reilly Media, Inc.

k-Nearest Neighbors Overview

- **Assumptions**
  - *k*NNs do not make any assumptions about the distribution of the data
- **Parameters**
  - Number of neighbors
  - Distance measure (Euclidean, Manhattan, etc)
- **Strengths**
  - Easy to understand
  - Reasonable performance (at most twice as bad as the optimal classifier)
- **Weaknesses**
  - Prediction can be slow on large dataset
  - Often requires pre-processing
  - Performs poorly on datasets with a large number of features
  - Performs poorly on sparse datasets (where many values are 0)