

CHAPTER 2: SUPERVISED LEARNING

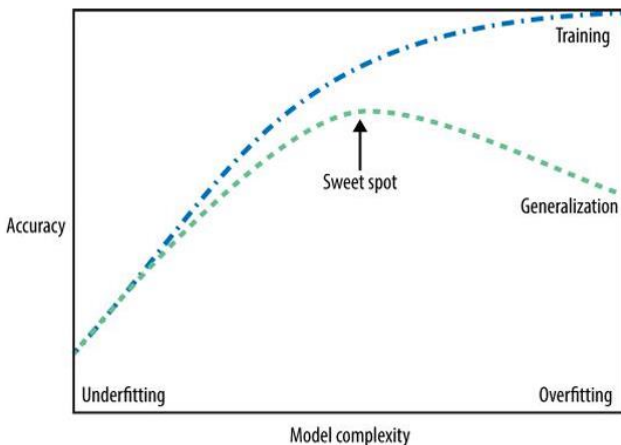
- Supervised learning is used whenever we want to predict a certain outcome from a given input, and we have examples of input/output pairs.

CLASSIFICATION AND REGRESSION

- Two major types of supervised learning: Classification and Regression.
- **Classification:**
Goal: Predict a class label, which is a choice from a predefined list of possibilities (labels).
Binary Classification: two classes. Yes/no questions.
Example: Detecting spam emails.
Multiclass Classification: more than two classes.
Example: Iris species classification.
- **Regression:**
Goal: Predict a real/continuous number.
Example: Predict a person's annual income from their education, age, where they live,...
Predict the yield of a corn farm from previous yields, weather, employees,...

GENERALIZATION, OVERFITTING, UNDERFITTING

- If a model is able to make accurate predictions on unseen data, it is able to *generalize* from the training set to the test set.
- **Overfitting** is when you fit a model too closely to the particularities of the training set and obtain a model that works well on the training set but is not able to generalize to new data.
- **Underfitting** is choosing too simple of a model.



RELATION OF MODEL COMPLEXITY TO DATASET SIZE

- Having more data and building appropriately more complex models can often work wonders for supervised learning tasks.

SUPERVISED LEARNING ALGORITHMS

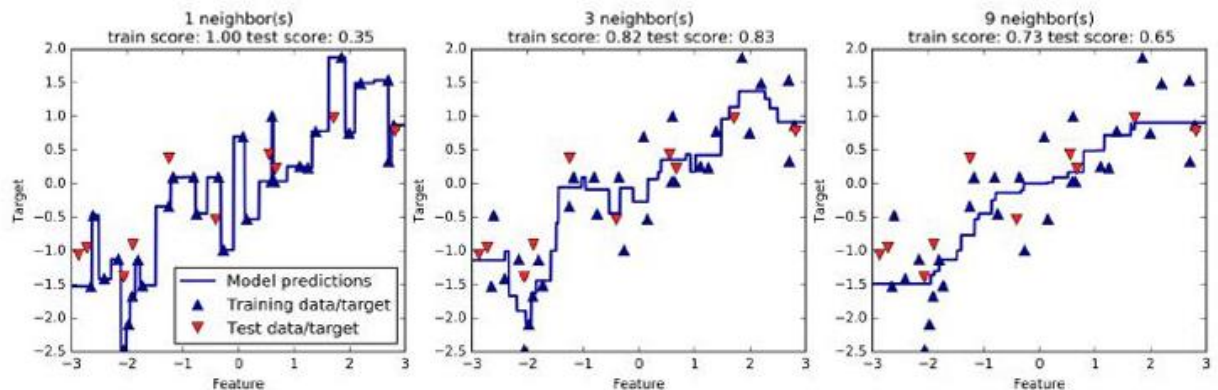
K-Nearest Neighbors

- To make a prediction for a new data point, the algorithm finds the closest data points in the training datasets – “nearest neighbors”.

K-Neighbors classification

- Using few neighbors corresponds to high model complexity, using many neighbors corresponds to low model complexity.

K-Neighbors regression



Strengths, weaknesses, and parameters

- Two important parameters to the KNeighbors classifier: the number of neighbors and how you measure distance between data points.
- **Strengths:**
 - + easy to understand
 - + reasonable performance without a lot of adjustments
 - + good baseline method to try before considering more advanced techniques.
 - + train fast
- **Weaknesses:**
 - + slow prediction for large training set.
 - + need to pre-process data.
 - + does not perform well on datasets with many features (hundreds >) or datasets where most features are 0 most of the time (sparse datasets).

Linear Models

- Linear models make a prediction using a linear function of the input features.

Linear models for regression

- General prediction formula: $\hat{y} = w[0] * x[0] + w[1] * x[1] + \dots + w[p] * x[p] + b$
 $x[0]$ to $x[p]$ denotes the features (the number of features is p).

w and b are parameters.

\hat{Y} is the prediction the model makes.

For datasets with a single features: $\hat{y} = w[0] * x[0] + b$ (equation for a line)

- Linear models for regression can be characterized as regression models for which the prediction is a line for a single feature, a plane when using two features, a hyper-plane in higher dimensions.

Linear regression (OLS)

- Linear regression finds the parameters w and b that minimize the mean squared error between predictions and the true regression targets, y , on the training set.
- The **mean squared error** (MSE) is the sum of the squared differences between the predictions and the true values.
- Linear regression has no parameters, which is a benefit, but also has no way to control model complexity.

Ridge Regression

- Same prediction formula used for Linear Regression.
- The coefficients w are not chosen not only so that they predict well on the training set but also to fit an additional constraint.
- All entries of w should be close to 0. \Rightarrow small slope \Rightarrow each features have as little effect on the outcome as possible \Rightarrow **Regularization**.
- **Regularization** means explicitly restricting a model to avoid overfitting.
- Choose the Ridge model over Linear Regression model for better generalization performance.