S&DS 265 / 565

Introductory Machine Learning

Classification and Regression Concepts

Thursday, September 9

Yale

Logistics

- Recordings posted to Canvas under Media Library
- Assignment 1 posted on Tuesday
- Quiz 0 available on Canvas at noon today, for 24 hours
- Check Canvas / EdD for office hours; updated later today

Plan for Today

- Continue Python elements
- Pandas and linear regression example
- Basics of classification, regression, overfitting

Python elements (continued)

+ Code + Text

Python and Jupyter essentials for iML

This notebook was adapted from multiple resources including the Data8 curriculum, Yale EENG201, and Stanford CS231. It is intended to give you a quick "jumpstart" and introduction to the tools that we will use throughout the course, based on Python, Jupyter notebooks, and essential useful packages like `numpy` and `pandas`.

It's important to recognize that practice is crucial here---you need to write code and implement things, making mistakes along the way, to gain proficiency in this material.


Subtopics marked with the scream icon are a little more advanced, and can be skipped on a first reading.

Get Started

Different ways to run Python

1. Create a file using editor, then: `$ python myscript.py`
2. Run interpreter interactively `$ python`
3. Use a Python environment, e.g. Anaconda or Google Colab

We recommend Anaconda:

- easy to install
- easy to add additional packages
- allows creation of custom environments

But Google Colab is also a good option. We plan to create a video on how to use Google Colab.

Pandas example

+ Code + Text

The New York Times Covid-19 Database

The New York Times Covid-19 Database is a county-level database of confirmed cases and deaths, compiled from state and local governments and health departments across the United States. The initial release of the database was on Thursday, March 26, 2020, and it is updated daily.

These data have fueled many articles and graphics by The Times; these are updated regularly at https://www.nytimes.com/interactive/2020/us/coronavirus-us-cases.html. The Times has created many visualizations that are effective communications of important information about the pandemic.

The data are publically available via GitHub: https://github.com/nytimes/covid-19-data. In this illustration we will only use the data aggregated at the state level.

```
[ ] import pandas as pd
    import numpy as np

    %matplotlib inline
    import matplotlib.pyplot as plt
    plt.style.use('fivethirtyeight')
```

```
[ ] covid_table = pd.read_csv("https://raw.githubusercontent.com/nytimes/covid-19-data/master/us-states.csv")
    covid_table = covid_table.drop('fips', axis=1)
    covid_table.tail(20)
```

	date	state	cases	deaths
30464	2021-09-07	North Dakota	119995	1596
30465	2021-09-07	Northern Mariana Islands	248	2
30466	2021-09-07	Ohio	1262018	21020
30467	2021-09-07	Oklahoma	570923	8001

Some Terminology

- supervised vs. unsupervised
- classification vs. regression
- prediction vs. inference

Supervised Learning vs. Unsupervised Learning

Supervised learning:

- Given a set of (x, y), learn to predict y using x.
- e.g.
  - Predicting whether a loan will default based on customer characteristics

Supervised Learning vs. Unsupervised Learning

Supervised learning:

- Given a set of (x, y), learn to predict y using x.
- e.g.
  - Predicting whether a loan will default based on customer characteristics

Unsupervised learning:

- Given a set of x, learn underlying structure or relationships of x.
- e.g.
  - Identifying market segments with similar spending patterns.

Classification vs. Regression

The `Income` dataset:

Education	Seniority	Income
21.58621	113.1034	99.91717
18.27586	119.3103	92.57913
12.06897	100.6897	34.67873
17.03448	187.5862	78.70281
19.93103	20.0000	68.00992
18.27586	26.2069	71.50449

Information for 30 *simulated individuals*.

Classification vs. Regression

The `Income` dataset:

Education	Seniority	Income
21.58621	113.1034	99.91717
18.27586	119.3103	92.57913
12.06897	100.6897	34.67873
17.03448	187.5862	78.70281
19.93103	20.0000	68.00992
18.27586	26.2069	71.50449

Information for 30 *simulated individuals*.

Regression: Model income based on other characteristics.

Classification vs. Regression

The `Income` dataset:

Education	Seniority	Income
21.58621	113.1034	99.91717
18.27586	119.3103	92.57913
12.06897	100.6897	34.67873
17.03448	187.5862	78.70281
19.93103	20.0000	68.00992
18.27586	26.2069	71.50449

Information for 30 *simulated individuals*.

Regression: Model income based on other characteristics.

Classification: Model whether someone will earn above the median income based on other characteristics.

Inference vs. Prediction

The `Income` dataset:

Education	Seniority	Income
21.58621	113.1034	99.91717
18.27586	119.3103	92.57913
12.06897	100.6897	34.67873
17.03448	187.5862	78.70281
19.93103	20.0000	68.00992
18.27586	26.2069	71.50449

Information for 30 *simulated individuals*.

Prediction: accurately predict Y for new observations

Inference vs. Prediction

The `Income` dataset:

Education	Seniority	Income
21.58621	113.1034	99.91717
18.27586	119.3103	92.57913
12.06897	100.6897	34.67873
17.03448	187.5862	78.70281
19.93103	20.0000	68.00992
18.27586	26.2069	71.50449

Information for 30 *simulated individuals*.

Prediction: accurately predict Y for new observations

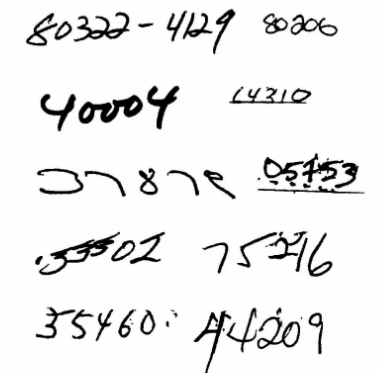
Inference: explain the underlying relationship between Y and X

Example: Handwritten Digit Recognition

- Data: images of handwritten digits (grayscale pixel values)
- Classify images as digits 0 to 9.

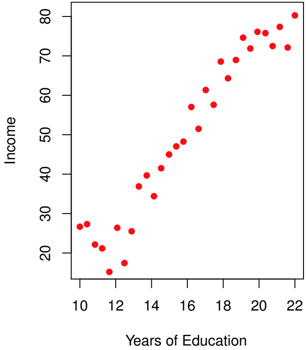
Example: Handwritten Digit Recognition

- Data: images of handwritten digits (grayscale pixel values)
- Classify images as digits 0 to 9.



Regression Example

The `Income` dataset:



Quantitative response Y

Predictors $X = (X_1, \ldots, X_p)$

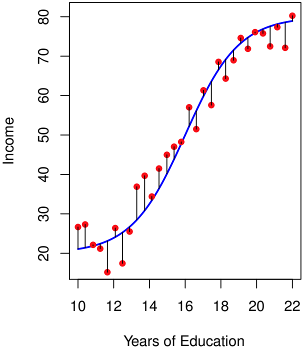
Assume the relationship can be expressed by:

$$Y = f(X) + \epsilon,$$

where f is a fixed, unknown function and ϵ is error term.

Regression Example

The `Income` dataset:



Quantitative response Y

Predictors $X = (X_1, \ldots, X_p)$

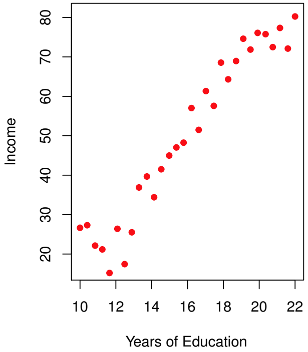
Assume the relationship can be expressed by:

$$Y = f(X) + \epsilon,$$

where f is a fixed, unknown function and ϵ is error term.

Regression Example

Back to regression with $p = 1$:



$$Y = f(X) + \epsilon$$

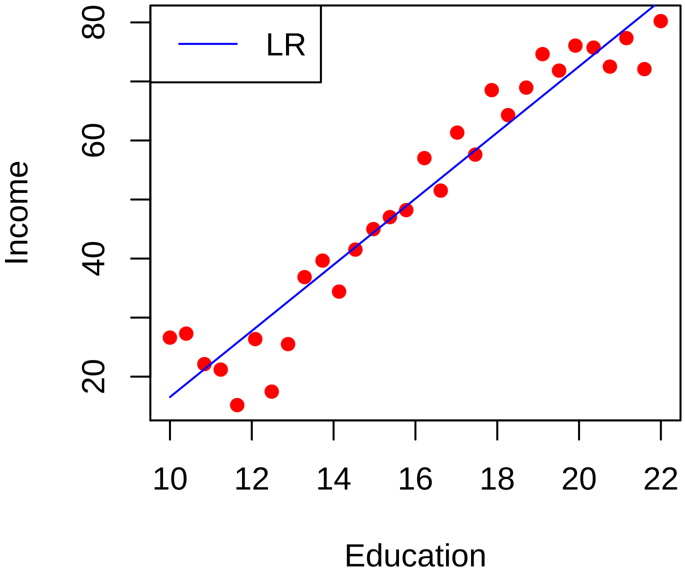
Modeling:

Use a procedure to get $\widehat{f}$. Derive estimates $\widehat{Y} = \widehat{f}(X)$.

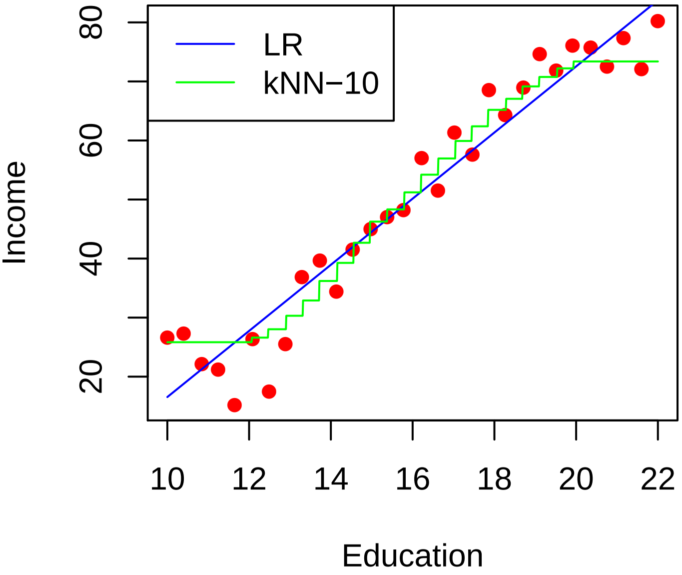
Possible Regression Approaches

- linear regression
  - Fitting a straight line through the data.
- k-nearest neighbors regression
  - Average together the y_i for x_i close to x

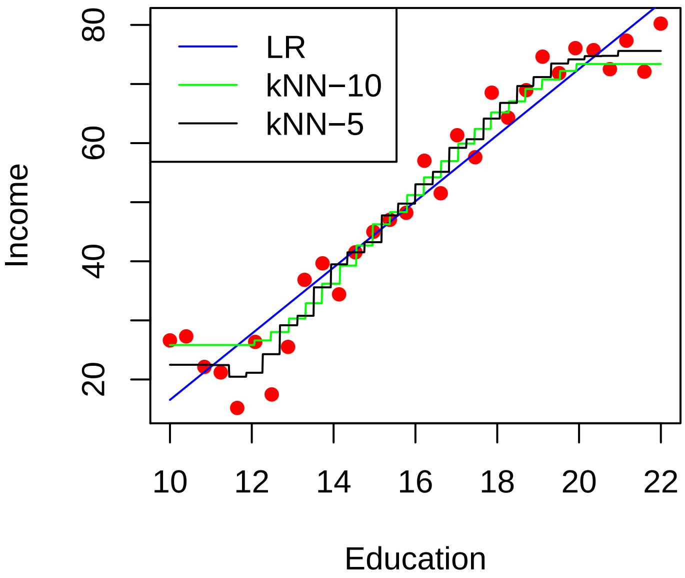
Possible Regression Approaches



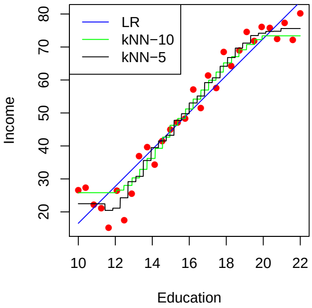
Possible Regression Approaches



Possible Regression Approaches



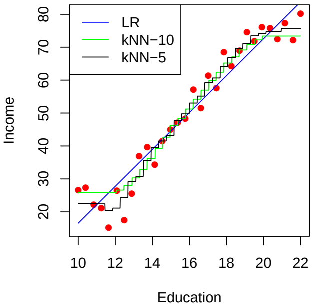
Possible Regression Approaches



Measuring performance via **Mean Squared Error**

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

Possible Regression Approaches



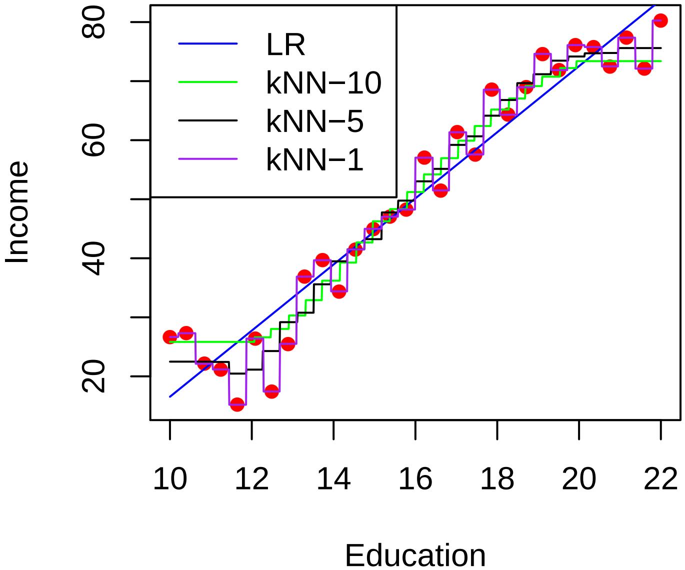
Measuring performance via **Mean Squared Error**

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{f}(x_i))^2$$

MSEs for three methods:

Linear Regression	29.829
k-Nearest Neighbors (k=10)	23.519
k-Nearest Neighbors (k=5)	16.21

A k-nearest neighbors model with $k = 5$ achieves lowest error. Is it the best?


LR
kNN−10
kNN−5
kNN−1
Income
Education
80
60
40
20
10
12
14
16
18
20
22

Training MSE vs. Test MSE

MSE in the previous table, **training MSE**, was computed based on data used in fitting the model.

We are more interested in **test MSE** computed on *unseen data*.

Training MSE vs. Test MSE

MSE in the previous table, **training MSE**, was computed based on data used in fitting the model.

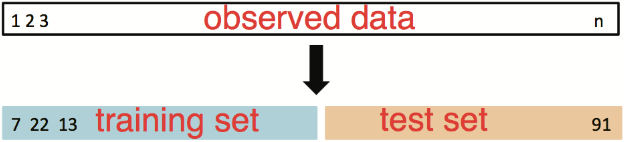
We are more interested in **test MSE** computed on *unseen data*.What if we don't have other data?

Training MSE vs. Test MSE

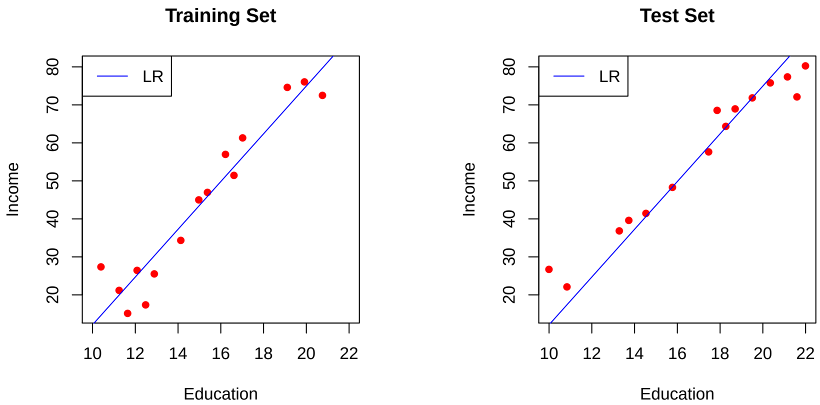
MSE in the previous table, **training MSE**, was computed based on data used in fitting the model.

We are more interested in **test MSE** computed on *unseen data*.What if we don't have other data?

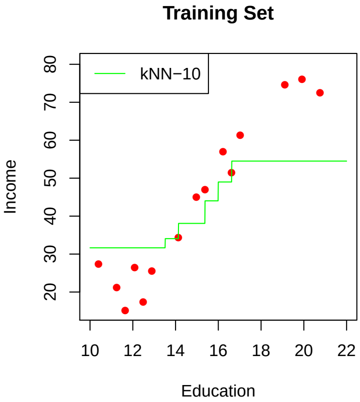
We can randomly split our data into a test set and a training set.

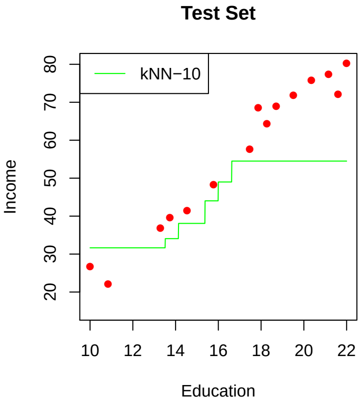


Regression Approaches Revisited

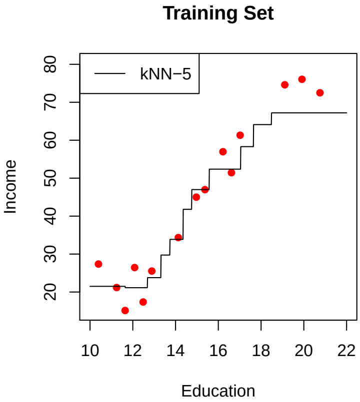


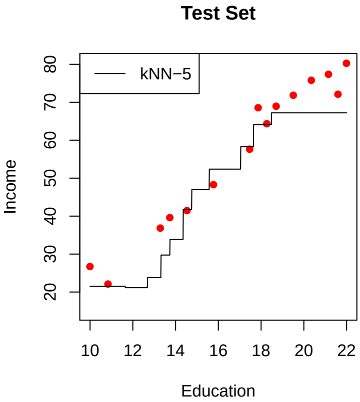
Regression Approaches Revisited



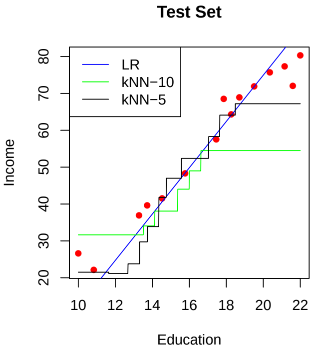


Regression Approaches Revisited





Regression Approaches Revisited

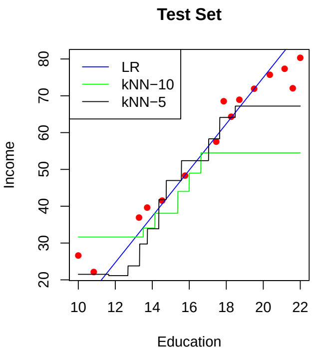


Compute MSE on the test set:

$$MSE = \frac{1}{n}\sum(y_i - \widehat{f}(x_i))^2$$

Linear Regression	37.807
k-Nearest Neighbors (k=10)	197.809
k-Nearest Neighbors (k=5)	48.682

Regression Approaches Revisited



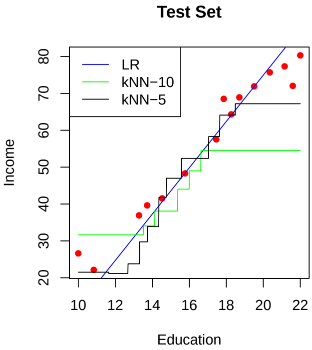
Compute MSE on the test set:

$$MSE = \frac{1}{n} \sum (y_i - \widehat{f}(x_i))^2$$

Linear Regression	37.807
k-Nearest Neighbors (k=10)	197.809
k-Nearest Neighbors (k=5)	48.682

So it appears that linear regression wins.

Regression Approaches Revisited



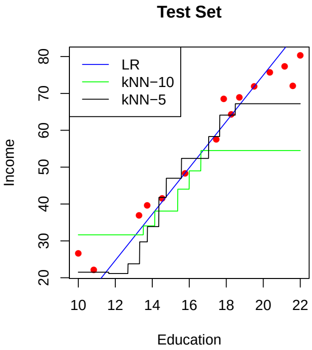
Compute MSE on the test set:

$$MSE = \frac{1}{n} \sum (y_i - \widehat{f}(x_i))^2$$

Linear Regression	37.807
k-Nearest Neighbors (k=10)	197.809
k-Nearest Neighbors (k=5)	48.682

So it appears that linear regression wins. Does it?

Regression Approaches Revisited



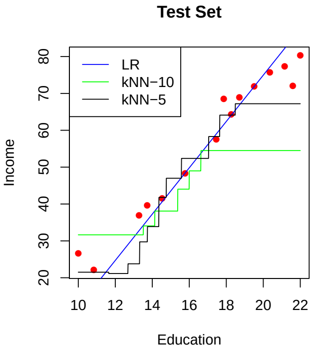
Compute MSE on the test set:

$$MSE = \frac{1}{n} \sum (y_i - \widehat{f}(x_i))^2$$

Linear Regression	37.807
k-Nearest Neighbors (k=10)	197.809
k-Nearest Neighbors (k=5)	48.682

So it appears that linear regression wins. Does it? With different random splits of test vs. training, we could have gotten different results.

Regression Approaches Revisited



Compute MSE on the test set:

$$MSE = \frac{1}{n} \sum (y_i - \widehat{f}(x_i))^2$$

Linear Regression	37.807
k-Nearest Neighbors (k=10)	197.809
k-Nearest Neighbors (k=5)	48.682

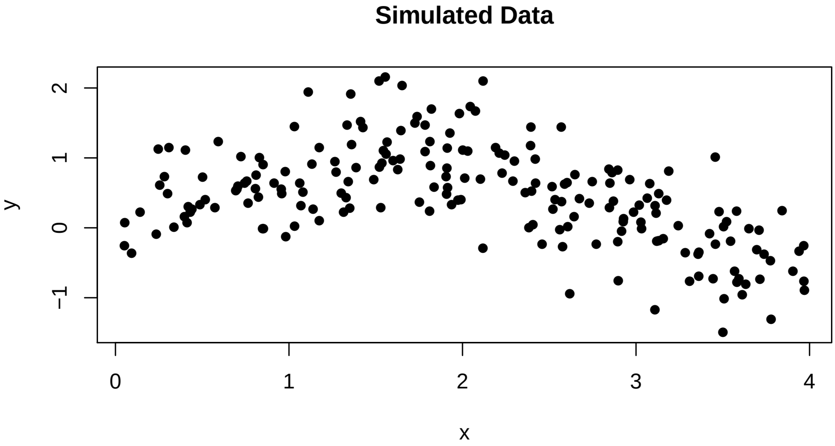
So it appears that linear regression wins. Does it? With different random splits of test vs. training, we could have gotten different results. We'll talk about ways around this later.

Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.

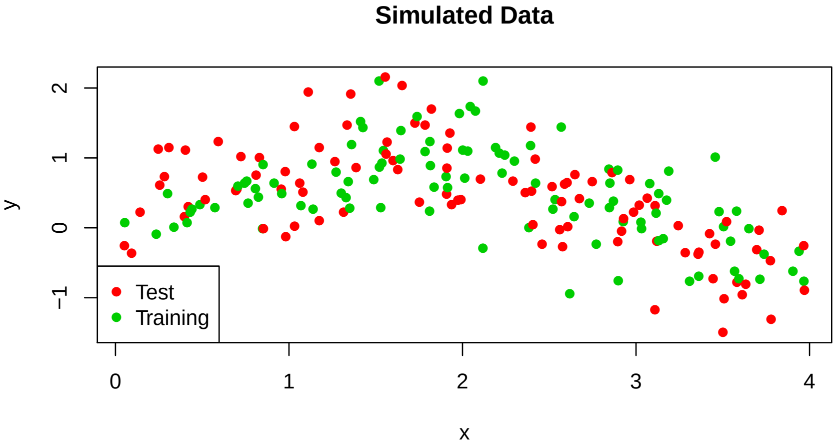
Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.
Let's examine this phenomenon using a bigger dataset:



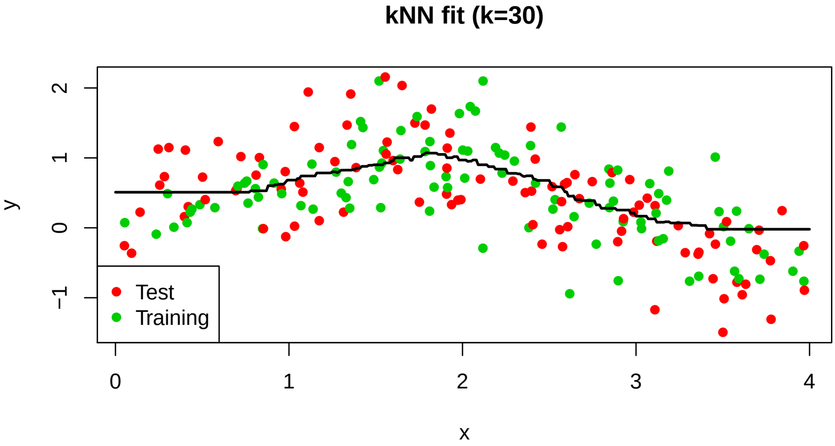
Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.
Let's examine this phenomenon using a bigger dataset:



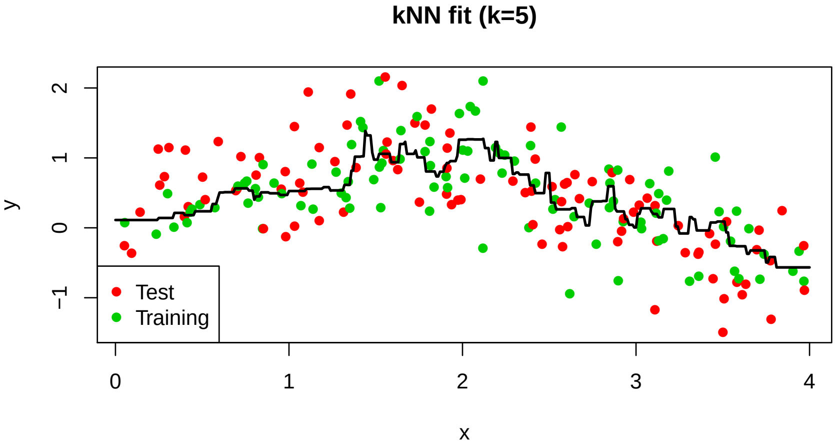
Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.
Let's examine this phenomenon using a bigger dataset:

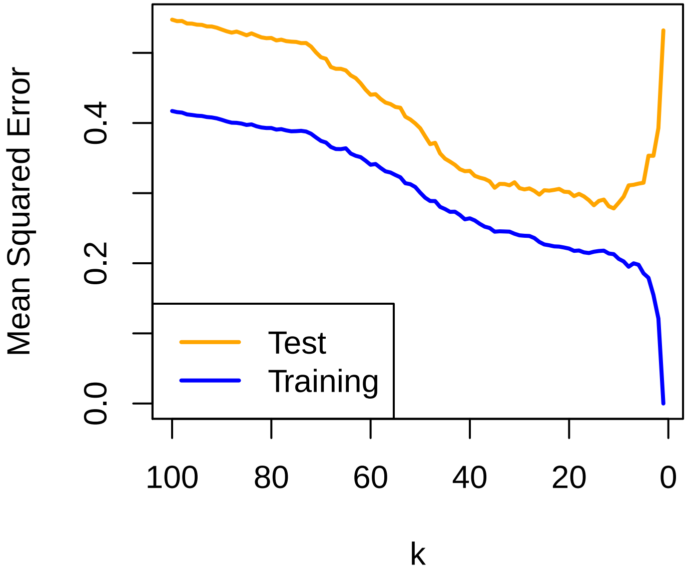


Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.
Let's examine this phenomenon using a bigger dataset:



Overfitting via k-Nearest Neighbors



Summary

- Two cultures: model based and prediction based
- Prediction based approaches are sometimes not interpretable
- Overfitting is easy with very flexible models and algorithms

Next week: Linear regression and classification