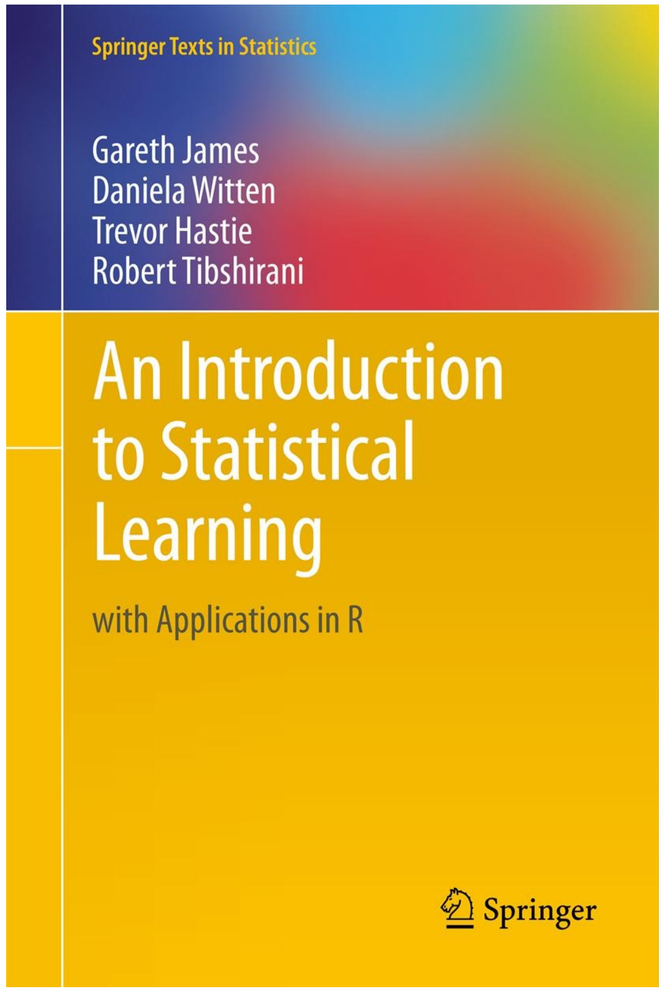
Introduction to Machine Learning

Piotr Januszewski

Great book

Most of the examples and content come from this book. It's amazing, I highly recommend you to give it a try.

And it's free! Here:
http://www-bcf.usc.edu/~gareth/ISL/



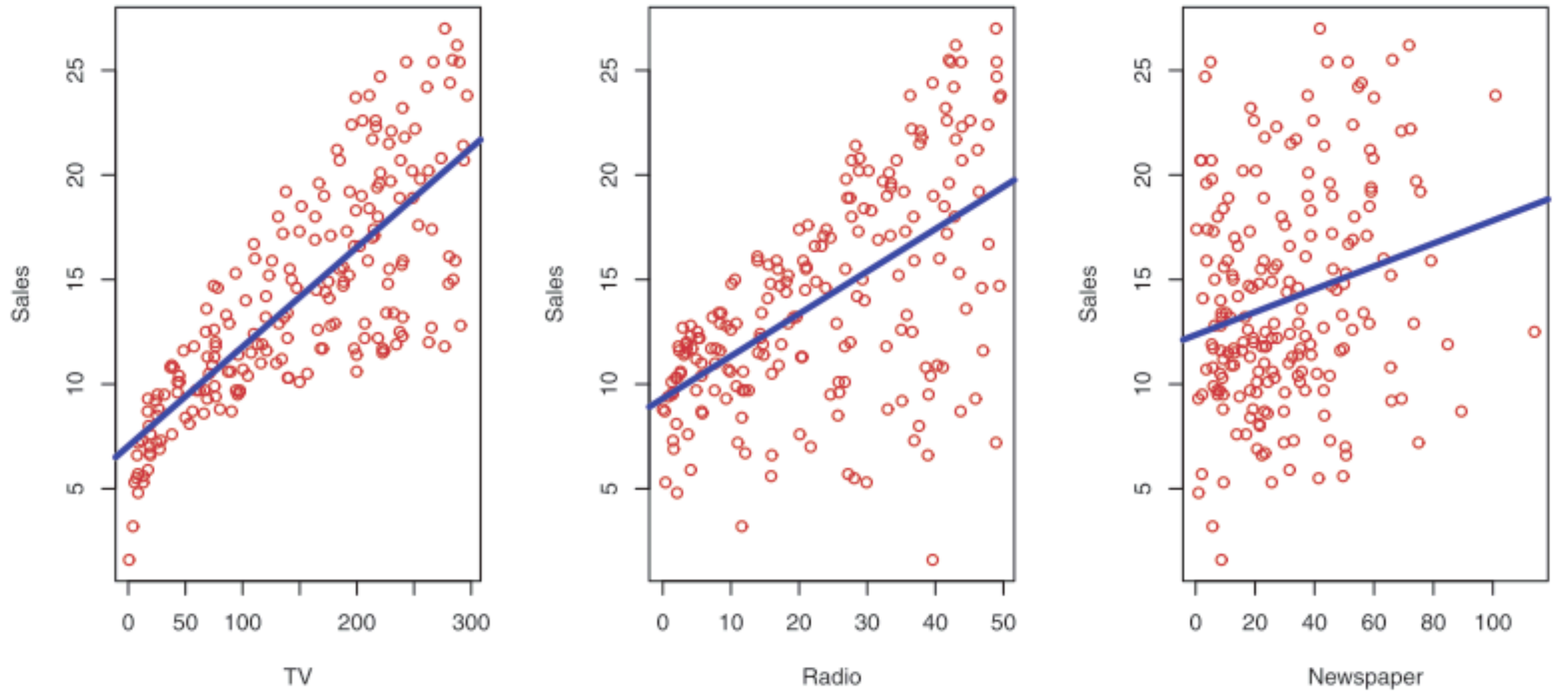
Roadmap

1. The Word of Introduction
    - What is machine learning?
    - Why do we estimate the model?
    - How do we estimate the model?
    - The trade-off between prediction accuracy and model interpretability.
    - Types of learning.
2. Assessing Model Accuracy
    - Measuring the quality of fit.
    - The bias-variance trade-off.
3. The Classification Setting
    - The Bayes classifier.
    - K-nearest neighbours.

Glossary

- Input variables (X) = predictors, independent variables, features or variables.
- An output variable (y) = response or dependent variable.
- Quantitative variables take on numerical values.
- Qualitative variables (categorical) take on values in one of K different classes.

What is machine learning?



The Advertising data set. The plot displays sales, in thousands of units, as a function of TV, radio and newspaper budgets, in thousands of dollars, for 200 different markets. In each plot we show the simple least squares fit of sales to that variable, as described later. In other words, each blue line represents a simple model that can be used to predict sales using TV, radio, and newspaper, respectively.

What is machine learning?

We observe a quantitative response Y and p different predictors: $X_1, X_2, \ldots, X_p$. We assume that there is some relationship between Y and $X = (X_1, X_2, \ldots, X_p)$, which can be written in the very general form:

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

Here f is some fixed but unknown function of X and ϵ is a random error term, which is independent of X and has mean zero. In this formulation, f represents the systematic information that X provides about Y.

Our goal is to estimate f with something called a model.

Why do we estimate the model?

- **Prediction:**
  In many situations, a set of inputs *X* are readily available, but the output *Y* cannot be easily obtained. We want to predict it!
- **Inference:**
  All the cases when we want to get “more” from our data than just predicting from unseen data. When we want to infer some information like:
  - Is there a relationship between advertising budget and sales?
  - How strong is the relationship between advertising budget and sales?
  - Which media contribute to sales?
  - Is the relationship linear, or quadratic, or …?
  - Is there a synergy among the advertising media?
    Two media used together yield stronger effect on sales then separately

How do we estimate the model?

We use training dataset!

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$$

$$X = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$$

Then apply machine learning methods to the training data to estimate unknown *f:*

$$y \approx \hat{f}(x) \textbf{ for any } (x, y) \textbf{ from the dataset}$$

How to do this depends on a specific method, but there are two general approaches…

How do we estimate the model?

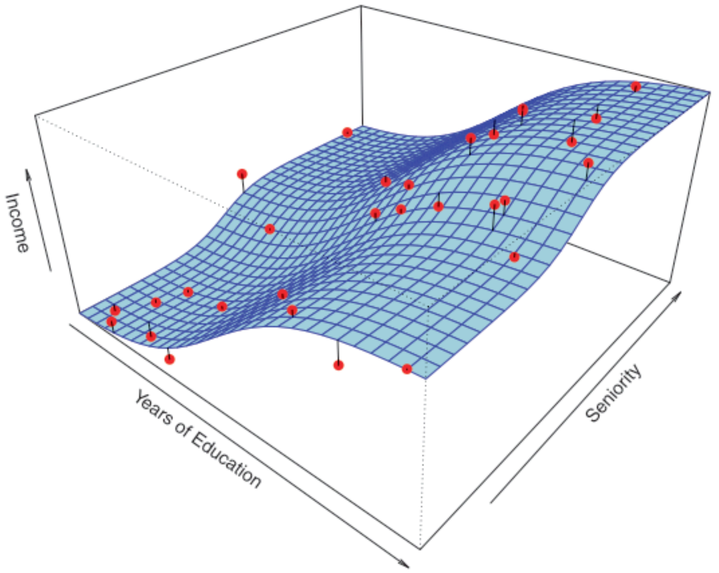
- **Parametric methods:**
  It's easier to estimate parameters of some model (e.g. linear function), than it is to fit arbitrary function (we say that problem is smaller).

  1. Make assumption about *f* form/shape, select model e.g. linear model.

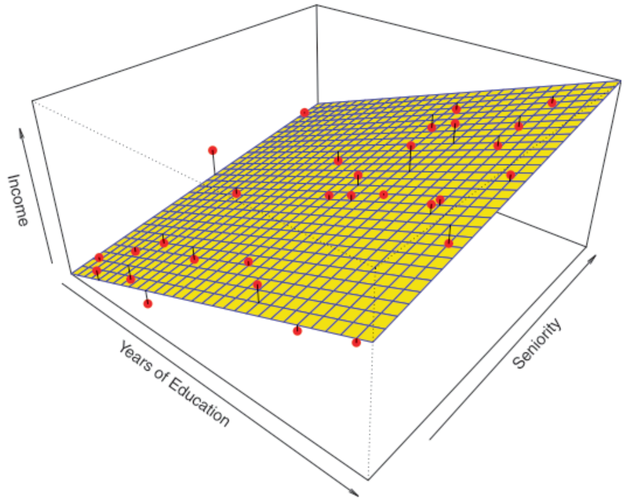
  2. Fit/train the model. Estimate its parameters such that: $y \approx \hat{f}(x)$

- **Non-parametric methods:**
  No explicite assumptions on *f* form (e.g. decision tree). It can fit wider range of *f*s. But it means the problem doesn't reduce to a small number of parameters and hence it might need larger number of observations (data points).

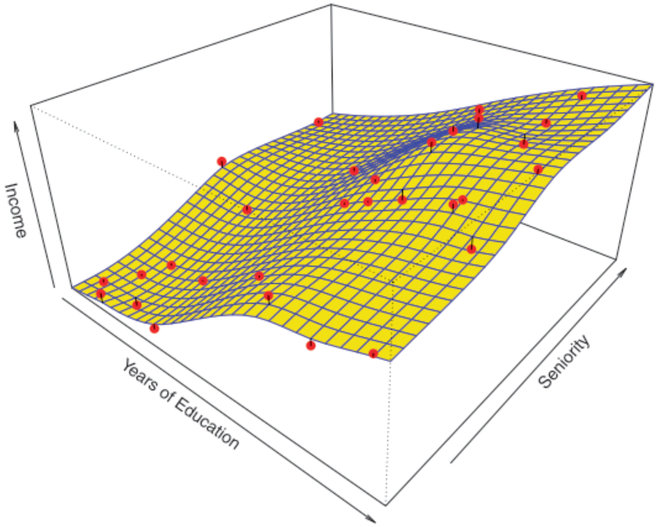
How do we estimate the model?



The plot displays income as a function of years of education and seniority in the Income data set. The blue surface represents the true underlying relationship between income and years of education and seniority, which is known since the data are simulated. The red dots indicate the observed values of these quantities for 30 individuals.



A linear model fit to the data.



A smooth thin-plate spline fit to the data.

The trade-off between prediction accuracy and model interpretability

- **Simple models**
  like linear regression
  - High interpretability,
  - Good for inference,
  - Can give inaccurate predictions.
- **Complex models**
  line neural network
  - Hard to interpret,
  - Bad for inference,
  - Great prediction power!

Types of learning

- **Supervised Learning**
  Dataset keeps associated responses for each predictor,

  - We want to model relationship between predictors *X* and responses *Y* to predict future or to better understand this relationship, inference.

- **Unsupervised Learning**
  We lack responses in the dataset. We can still do many useful things with such a dataset e.g.:

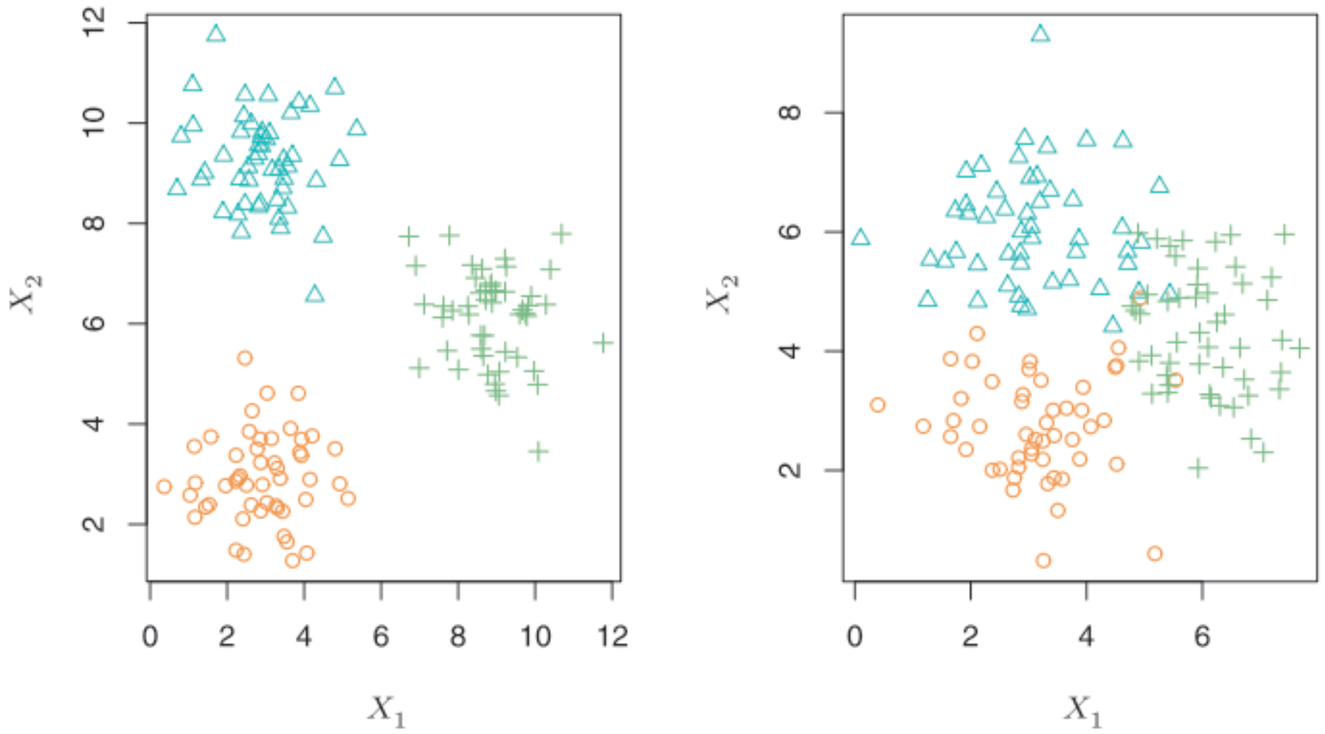
  - Finding patterns, clusters analysis,

  - Dimensionality reduction,

  - Latent features extraction,

  - Data modeling and generation.

Types of learning



A clustering data set involving three groups. Each group is shown using a different coloured symbol. Left: The three groups are well-separated. In this setting, a clustering approach should successfully identify the three groups. Right: There is some overlap among the groups. Now the clustering task is more challenging.

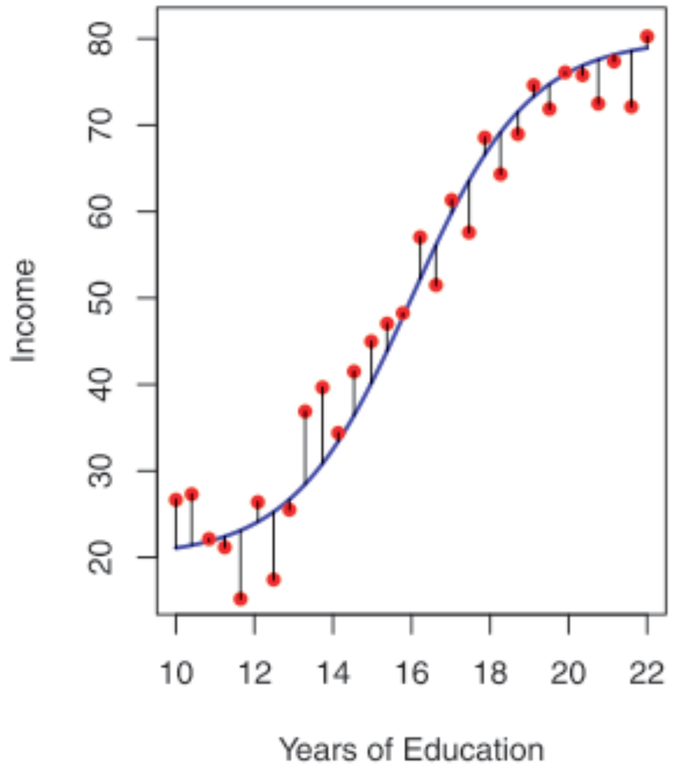
Assessing Model Accuracy

Measuring the quality of fit

Mean Squared Error can be used to asses a model fit to the dataset in the regression problem.

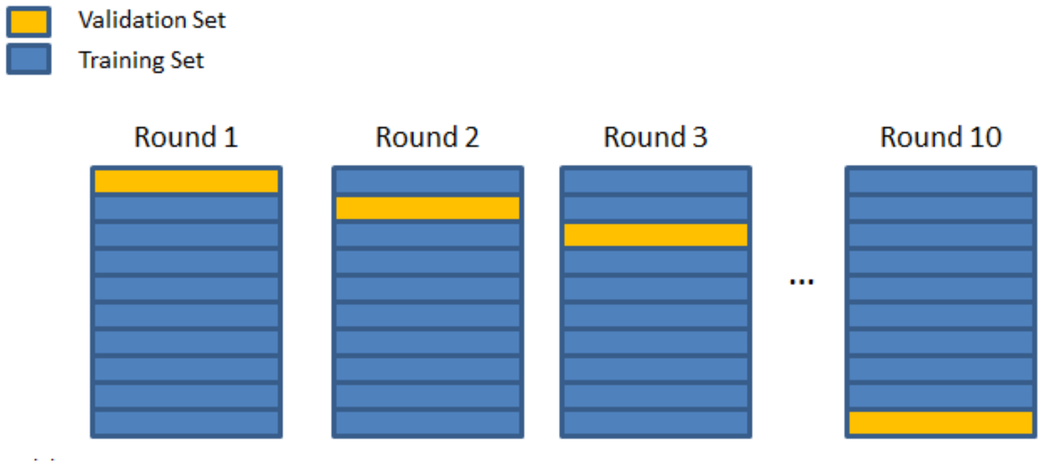
$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{f}(x_i))^2$$

What we really want is to find out how good is our model on unseen data. How do we do that?



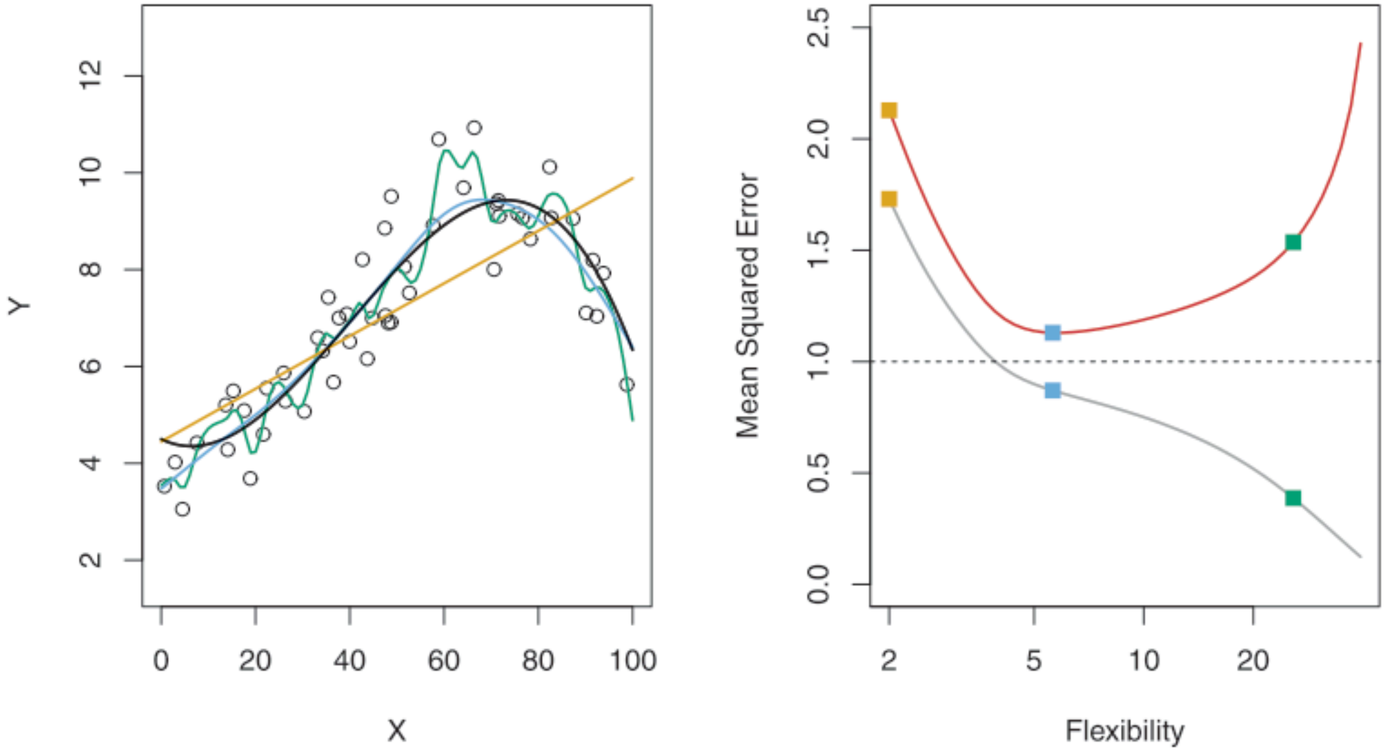
K-folds Cross Validation Method

1. Divide the dataset into k equal parts.
2. Use k-1 of the parts for training, your train set, and the part left for testing, your test set.
3. Repeat the procedure k times, rotating the test set.
4. Determine an expected performance metric like MSE by averaging results across the interactions.



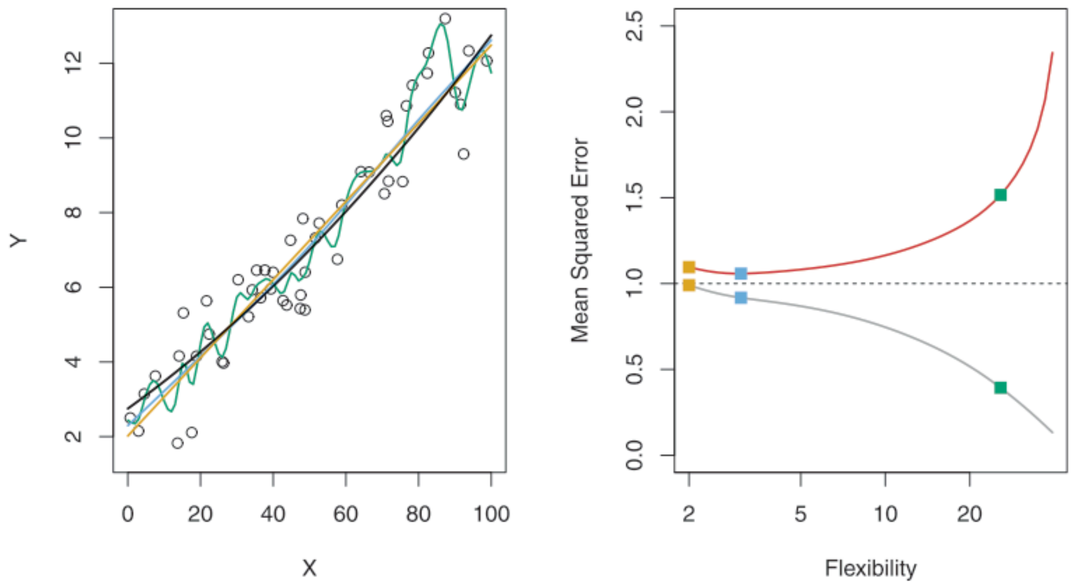
Measuring the quality of fit

- MSE_{test} will always be higher than irreducible error, Var(ϵ), coming from random noise in data.
- MSE_{test} will be always higher than MSE_{train}. It is because the former is minimised indirectly. There is also no guaranty that low train MSE means low test MSE! In deed, it's called overfitting.
- Overfitting is following noise to closely.
- When MSE_{test} >> MSE_{train} then model overfitted. You can see it on the right.

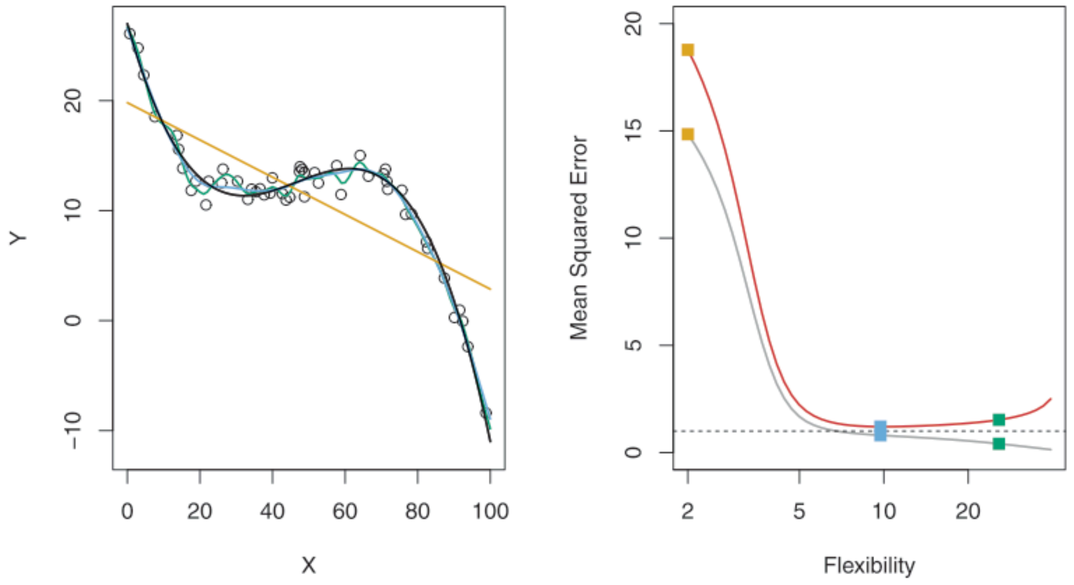


Left: Data simulated from f, shown in black. Three estimates of f are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves). Right: Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.

Measuring the quality of fit



Details are as before, using a different true f that is much closer to linear. In this setting, linear regression provides a very good fit to the data.

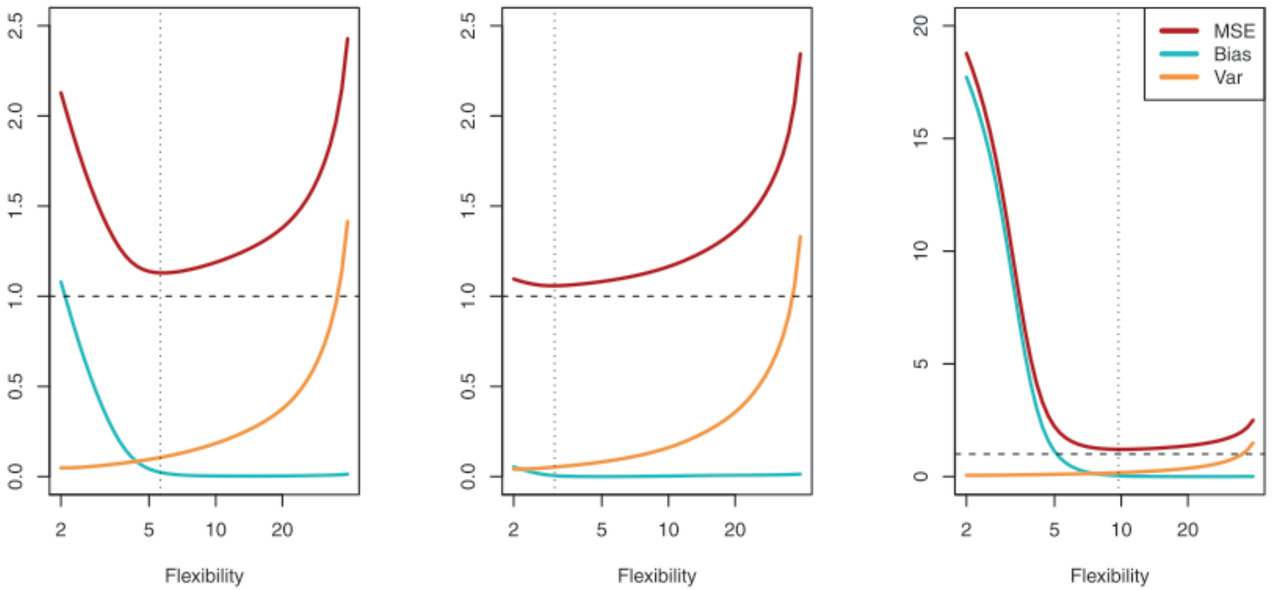


Details are as before, using a different true f that is far from linear. In this setting, linear regression provides a very poor fit to the data.

The bias-variance trade-off

The U-shape observed in the test MSE curves turns out to be the result of two competing properties of machine learning methods. Though the mathematical proof is beyond the scope of this class, it is possible to show that the expected test MSE, for a given data point, x_0, can always be decomposed into the sum of three fundamental quantities: the variance of *f*, the squared bias of *f* and the variance of the error terms:

$$E\left(y_0 - \hat{f}(x_0)\right)^2 = \mathrm{Var}(\hat{f}(x_0)) + [\mathrm{Bias}(\hat{f}(x_0))]^2 + \mathrm{Var}(\epsilon)$$



Squared bias (blue curve), variance (orange curve), error variance (dashed line), and test MSE (red curve) for the three data sets in figures before. The vertical dotted line indicates the flexibility level corresponding to the smallest test MSE.

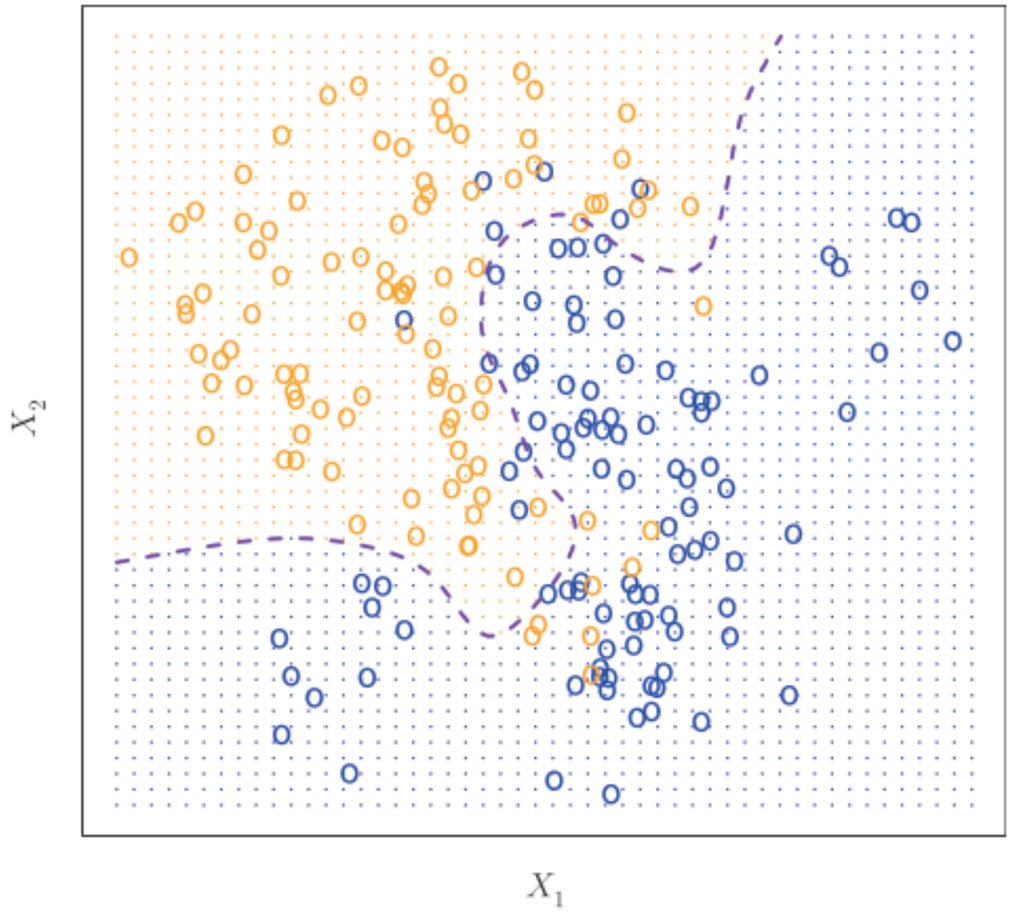
The Classification Setting

The Bayes classifier

Pretty much everything translates from quantitative herring to qualitative setting (e.g. Bias-Variance tradeoff).

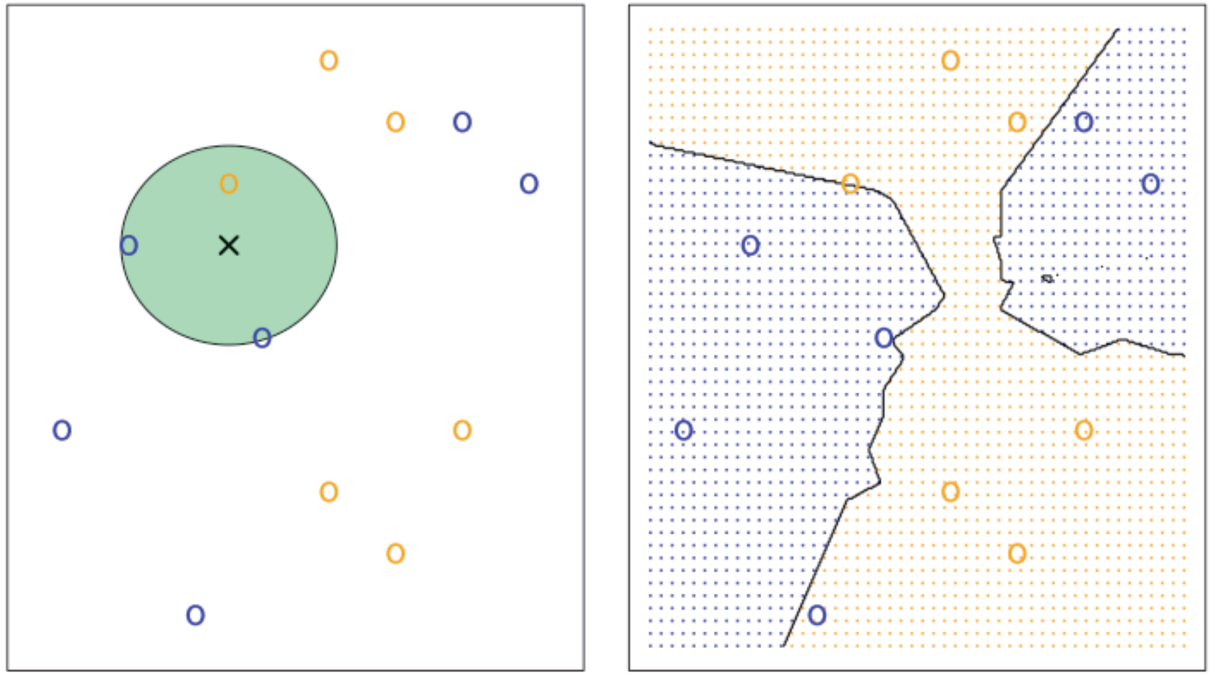
- Targets/responses are now classes.
- We measure accuracy with error rate: $\frac{1}{N}\sum_{i=1}^{N} I(y_i \neq \hat{f}(x_i))$

The Bayes classifier



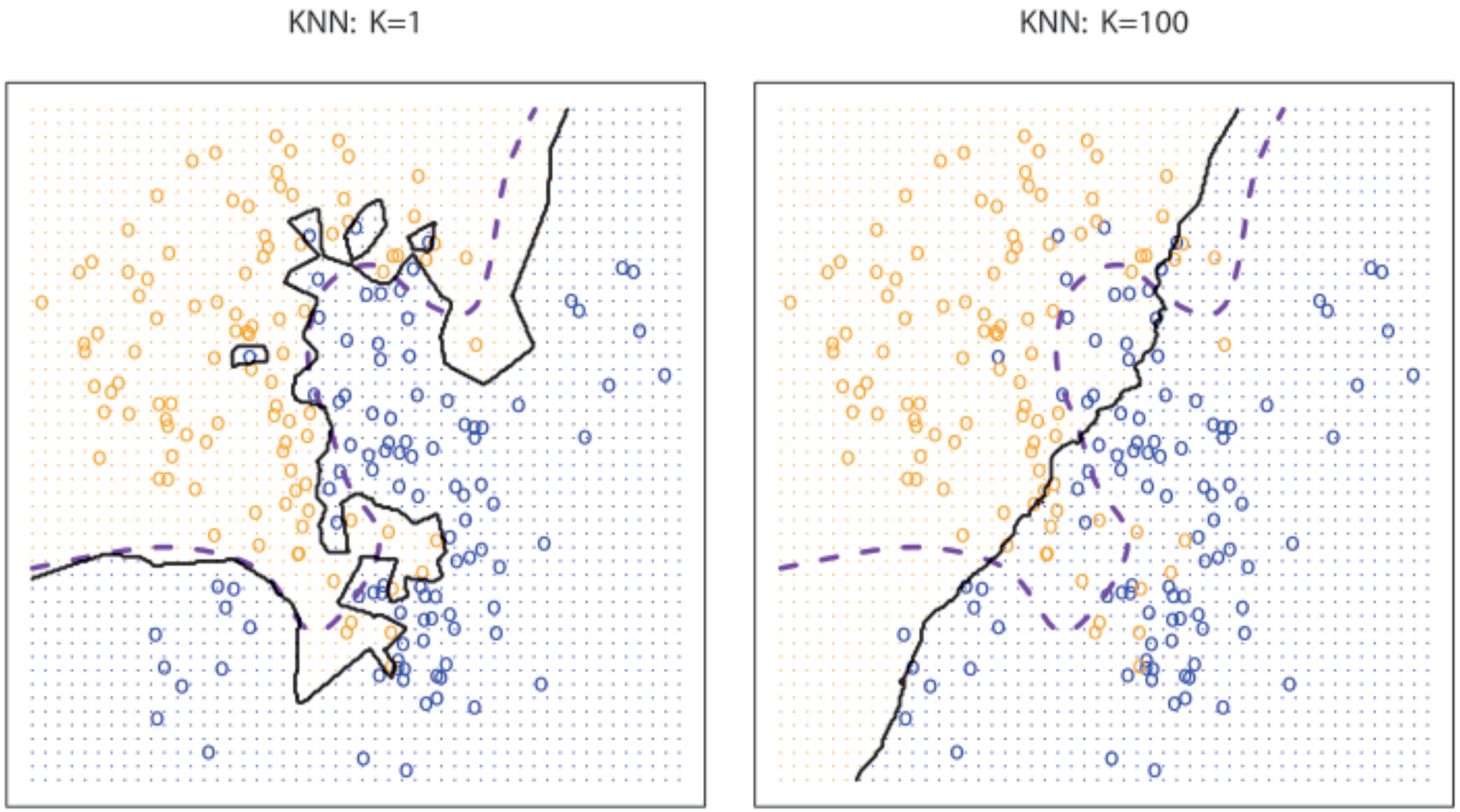
It turns out, that very simple classifier can be used to minimise (on average) test error rate. Just always pick the most probable class! Purple dotted line is a decision boundary, the probability of each class is equal on it, which means that each class is equally likely.

K-nearest neighbours



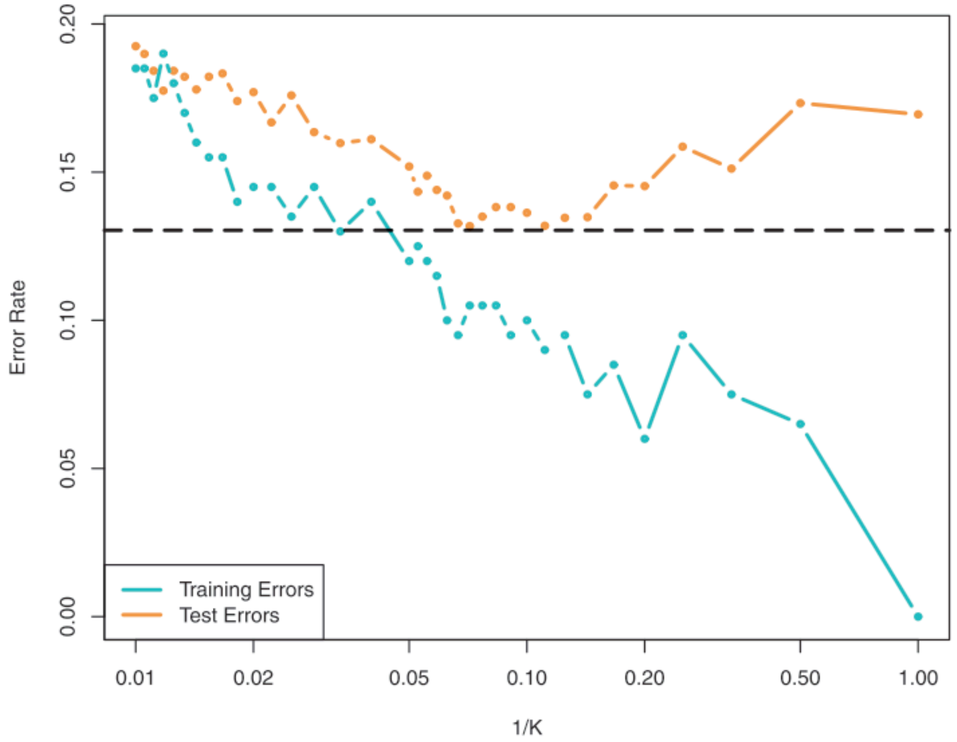
The KNN approach, using K =3, is illustrated in a simple situation with six blue observations and six orange observations. Left: a test observation at which a predicted class label is desired is shown as a black cross. The three closest points to the test observation are identified, and it is predicted that the test observation belongs to the most commonly-occurring class, in this case blue. Right: The KNN decision boundary for this example is shown in black. The blue grid indicates the region in which a test observation will be assigned to the blue class, and the orange grid indicates the region in which it will be assigned to the orange class.

K-nearest neighbours



A comparison of the KNN decision boundaries (solid black curves) obtained using K =1 and K = 100 on the data from Figure 2.13. With K =1, the decision boundary is overly flexible, while with K = 100 it is not sufficiently flexible. The Bayes decision boundary is shown as a purple dashed line.

K-nearest neighbours



The KNN training error rate (blue, 200 observations) and test error rate (orange, 5,000 observations) on the data from previous figure, as the level of flexibility (assessed using 1/K) increases, or equivalently as the number of neighbours K decreases. The black dashed line indicates the Bayes error rate. The jumpiness of the curves is due to the small size of the training data set.

To sum up…

Thank you for your attention 👨🏻‍🎓👩🏼‍🎓

Questions?