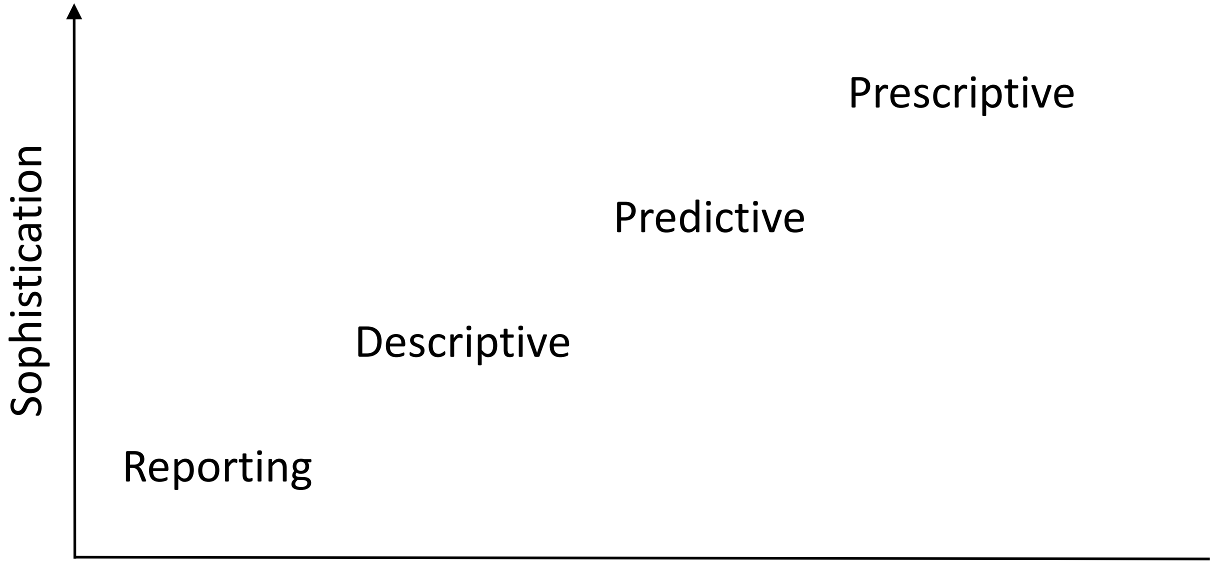
Machine Learning for Business Analytics

Lecture 01

What is Analytics?

- Gaining insights by understanding data



What is machine learning?

- AKA predictive analytics

- Machine learning is "field of study that gives computers the ability to learn without being explicitly programmed" (Arthur Samuel 1959).

- "A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P* if its performance at tasks in *T*, as measured by *P*, improves with experience *E*." (Tom Mitchell)

- For example,
  - T: detecting spam
  - P: percentage of spam messages correctly identified
  - E: labelled spam/non-spam email messages

Examples of machine learning problems

- Email spam detection
- Self-driving cars (e.g., predicting whether a pedestrian will cross the street)
- Medicine (e.g., radiology, reading scans and detecting disease)
- Language translation (e.g., Google translate)
- Fraud detection
- Weather prediction
- Recommender systems (e.g., Netflix, if you like X you might enjoy Y)

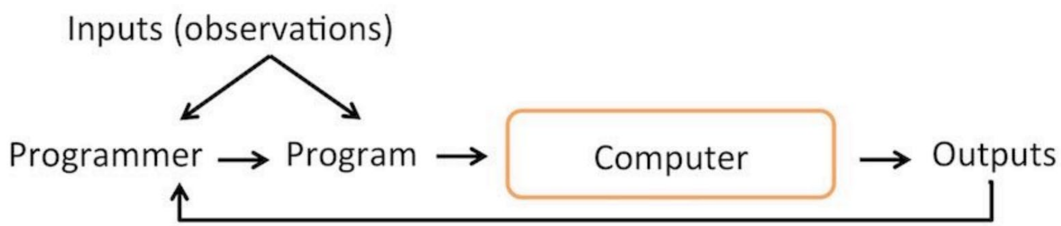
- Others from your experience?

Playing games

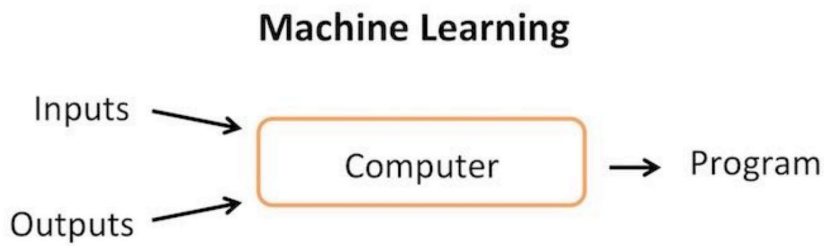
Today’s class

1. What is machine learning?
2. How do we measure prediction accuracy?
   Mean squared error
3. Improving prediction accuracy by understanding the bias-variance trade-off
   Regularization, Ridge, LASSO

Machine learning vs traditional programming



Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed
– Arthur Samuel (1959)



Sebastian Raschka, 2016

Notation

- We will refer to our response variable as Y and we will denote successive observations as $Y_1, Y_2, \dots, Y_n$

- We will refer to our predictors as X

- X can be a collection of predictors, e.g., $X = (X_1, X_2, X_3)$ where each X_k is a column vector

- We will write our models as $Y = f(X) + e$, where e is randomness that we cannot model

Different types of machine learning

1. Regression versus classification

2. Supervised versus unsupervised learning

3. Prediction versus inference

Supervised learning

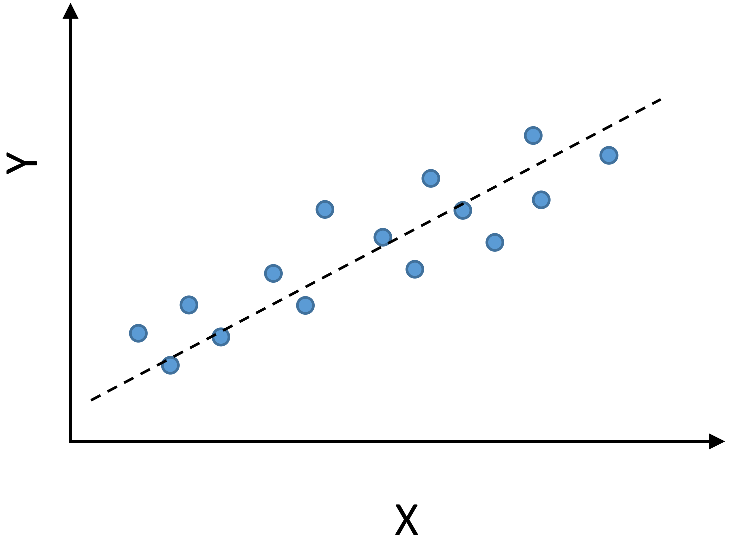
- We are given labelled data with an outcome variable $Y = (Y_1, Y_2, \dots, Y_n)$
  - For example, the outcome could be sales
  - Or the outcome could be a label (spam, non-spam)
  - Here, n is the number of observations in our dataset

- For each outcome Y_i, we also have a number of predictors X_i (aka predictors, regressors, covariates)
  - For example, $X_i = (\mathrm{X1_i}, \mathrm{X2_i}, \dots)$, where $\mathrm{X1}$ is ad spend on TV and $\mathrm{X2}$ is ad spend online
  - Or X could be the words included in an email message

- The goal is to predict Y given X

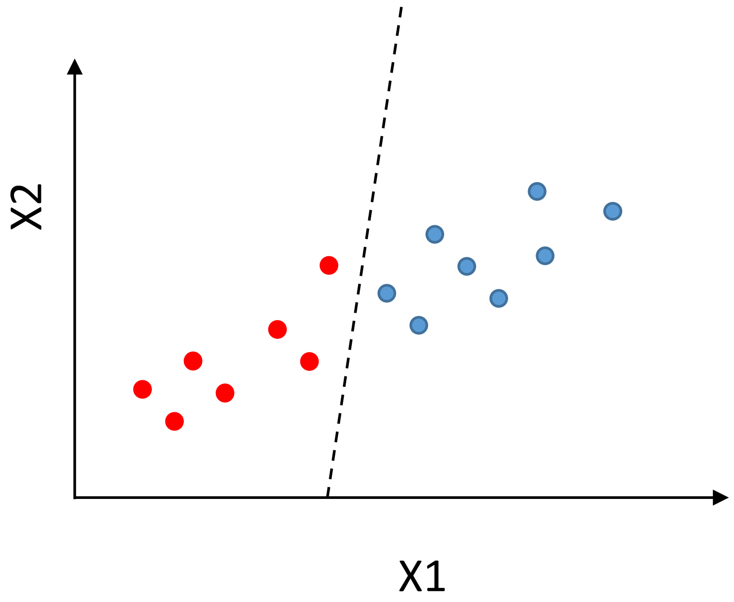
Supervised learning -- regression

- Regression means we are predicting a number



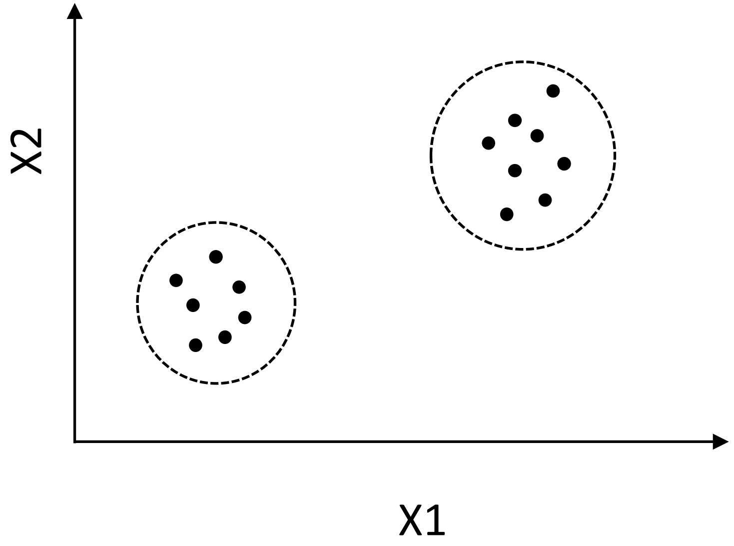
Supervised learning -- classification

- Classification means we are predicting a label

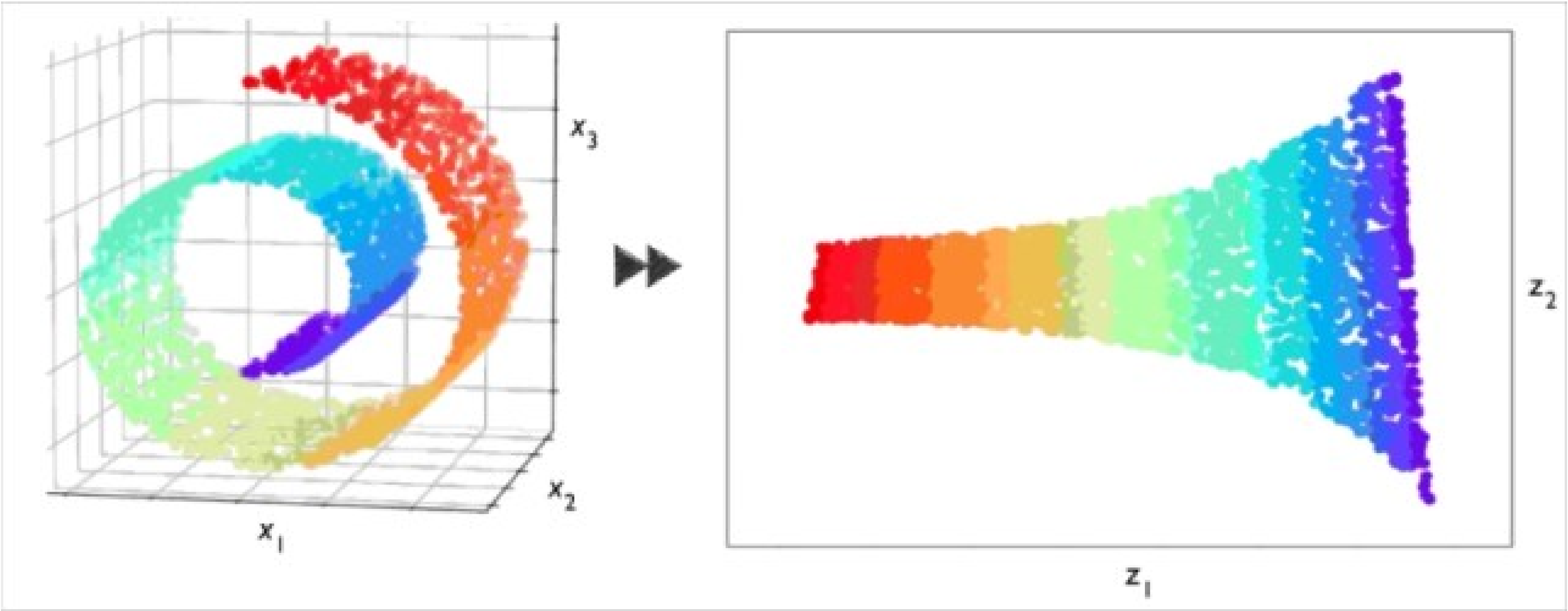


Unsupervised learning -- clustering

- There is no outcome Y in our data
- What can we do in this case?



Unsupervised learning – dimensionality reduction



Learning objectives: Prediction vs Inference

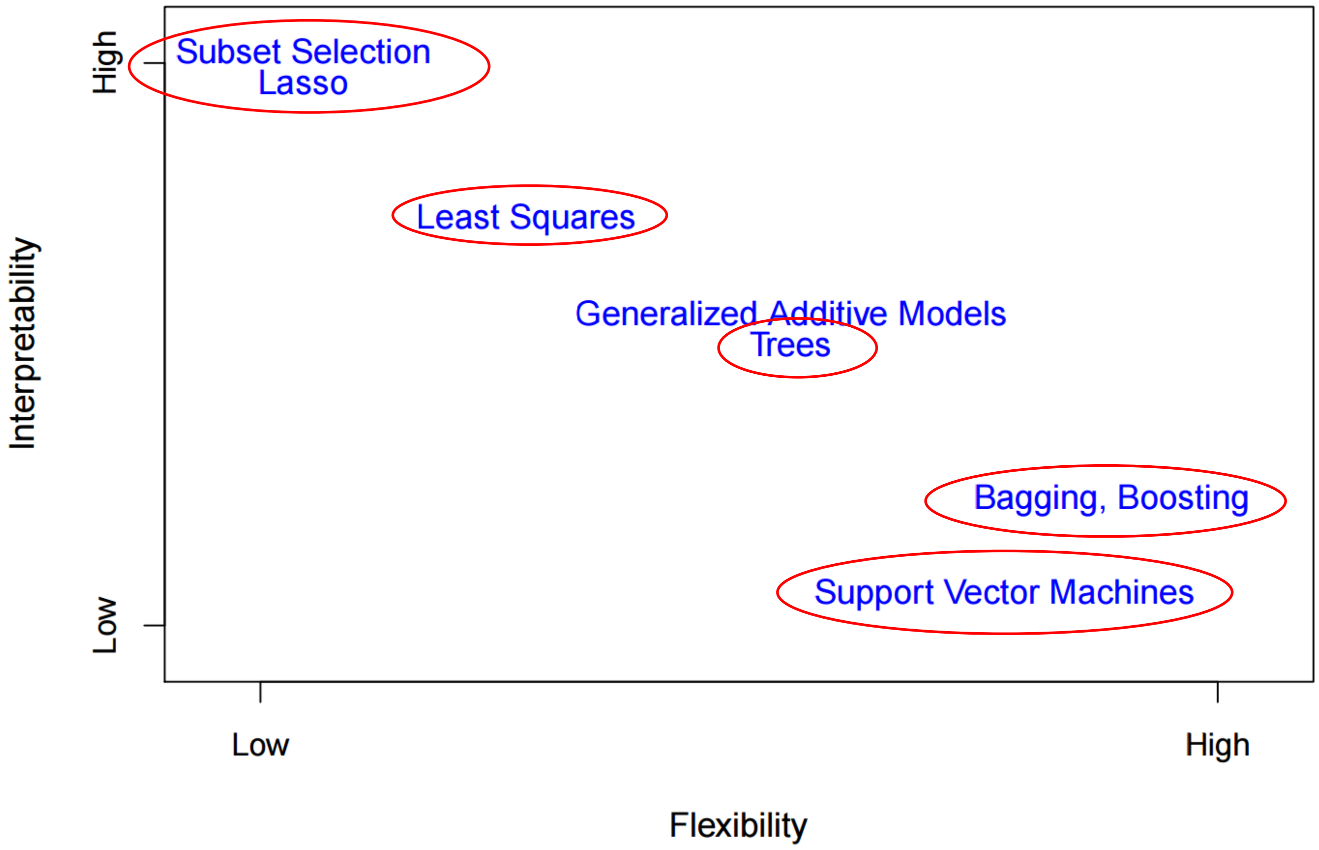
- **Prediction:** given new data point X predict a response Y
- $\hat{Y} = \hat{f}(X)$, where $\hat{f}$ is our estimate of f and $\hat{Y}$ is our prediction of Y
- We don’t care what $\hat{f}$ looks like – we treat it as a black box
- The main goal here is **accuracy**

Learning objectives: Prediction vs Inference

- **Inference:** find out the relationship between X and Y

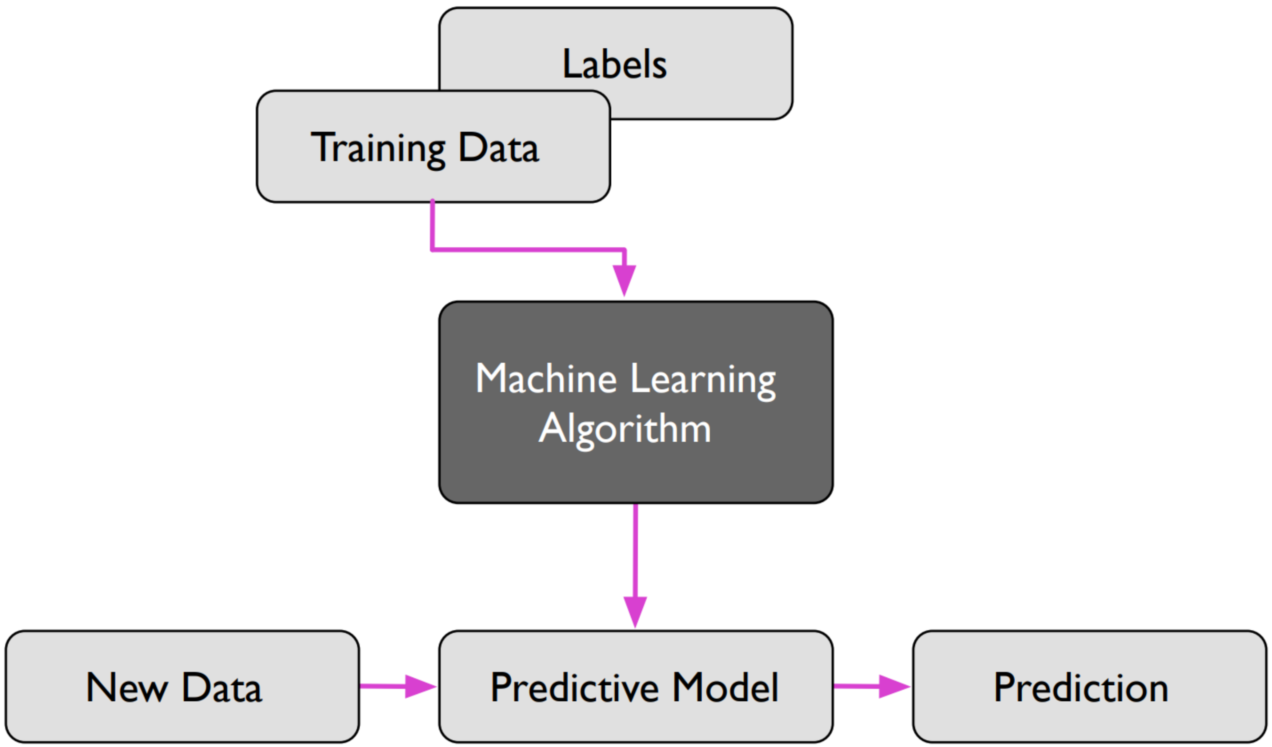
- Again, we start by estimating $\hat{Y} = \hat{f}(X)$
- But now, we care about the kind of relationship between Y and the various X's
    - $\hat{f}(X)$ is not a black box anymore
- For example…
    - What are the key determinants of credit card default?
- The main goal here is **interpretability**

Trade-offs between interpretability and flexibility



Source: Tibshirani et al.

Supervised learning workflow



Supervised learning workflow

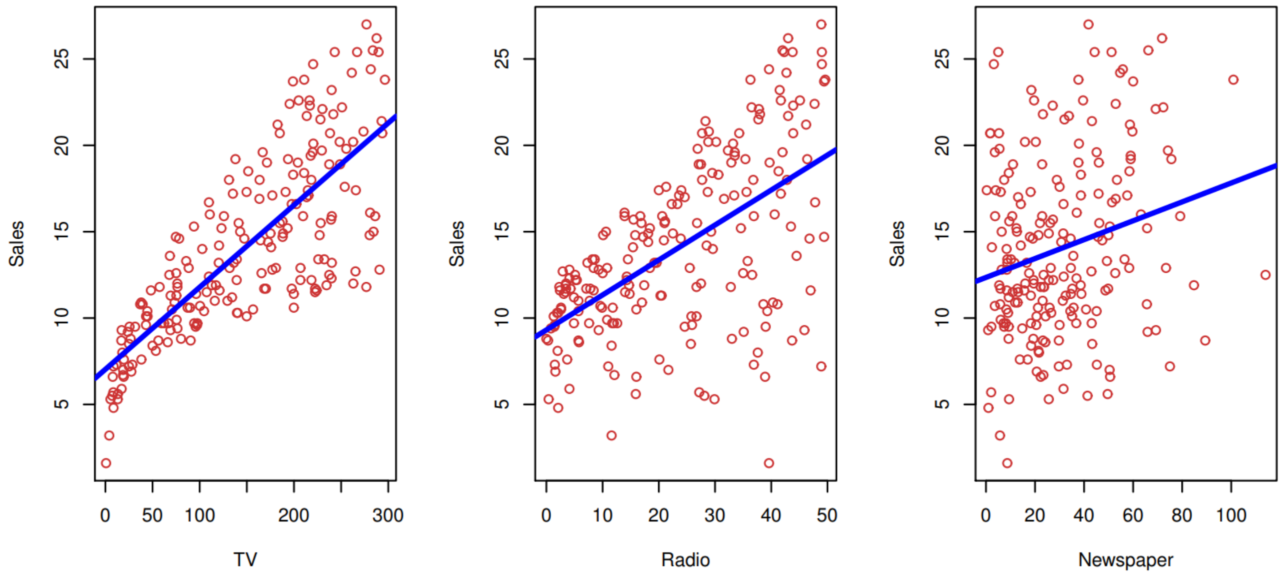
- Define the problem to be solved
- Collect labelled training data
- Choose an ML algorithm, and fit to the data
- Evaluate the model according to your chosen metric
- Use the model to predict out of sample

Supervised learning workflow

- Define the problem to be solved
- Collect **and clean** labelled training data
- Choose an ML algorithm, and fit to the data
- Evaluate the model according to your chosen metric
- Use the model to predict out of sample

Linear regression

- Simple approach to supervised learning
- Assumes linear relationship between predictors and outcome
- This assumption is almost never true but works well in practice



Linear regression

- A simple linear regression model $Y = f(X)$

$$Y = \beta_0 + \beta_1 X_1 + e$$

- β_0 is the intercept
- β_1 is the slope
- e is unobserved randomness we cannot model
- Our goal is to estimate the parameters of this model, β_0 and β_1
  - How do we do this?

Linear regression

- A simple linear regression model $Y = f(X)$

$$Y = \beta_0 + \beta_1 X_1 + e$$

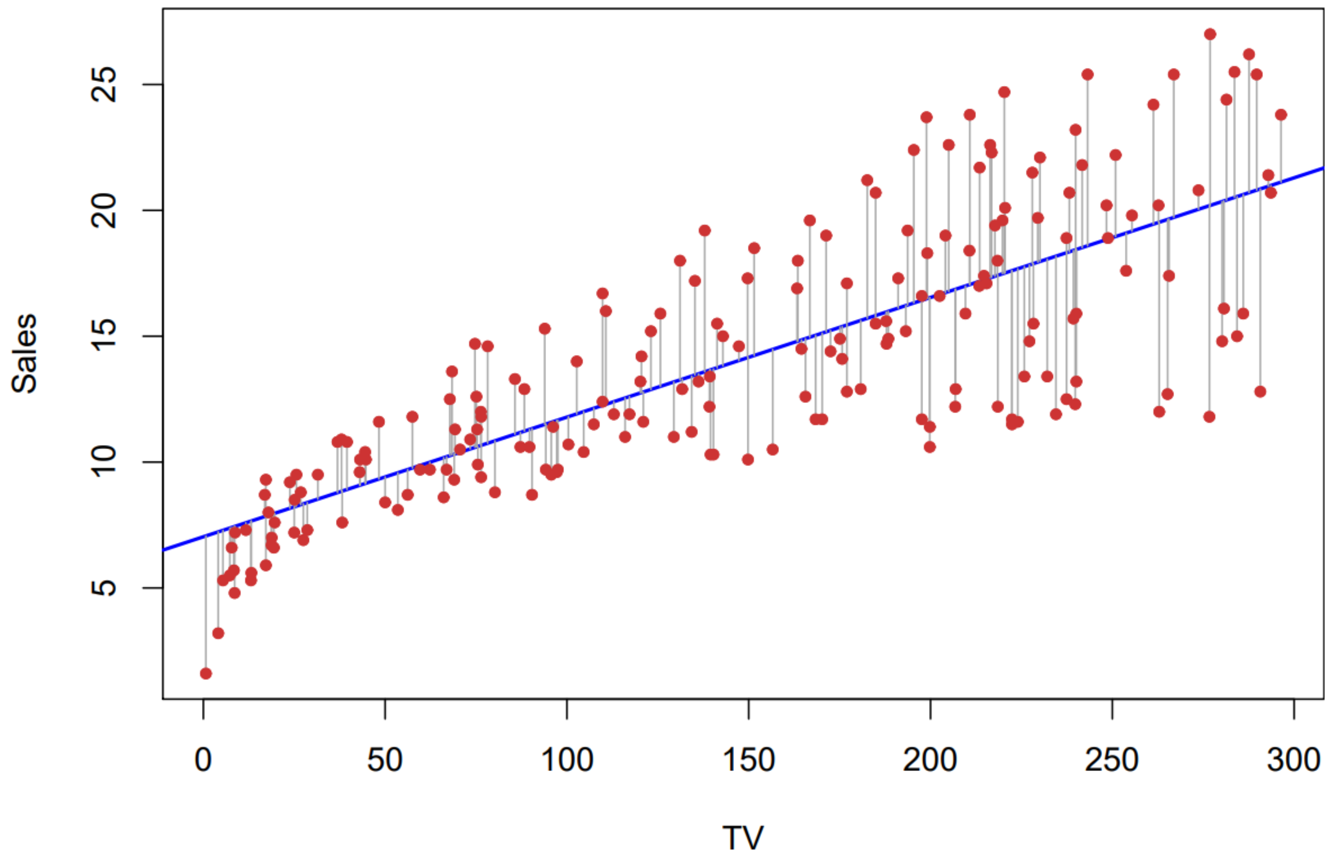
- Suppose we have some parameter estimates $\hat{\beta}_0$ and $\hat{\beta}_1$
  - Notice the hats – these are estimates parameters and might be different that the real ones
- Then we can form predictions $\hat{Y} = \hat{f}(X)$

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X_1$$

- We will call $\mathrm{res_i} = Y_i - \widehat{Y_i}$ the i^{th} residual

Measuring accuracy

$$\widehat{Sales} = \hat{\beta}_0 + \hat{\beta}_1 TV$$



Measuring accuracy

- One measure of accuracy is the average squared residual

$$Average\left(res_1^2, res_2^2, \dots, res_n^2\right)$$

- Why squared?
- Also known as the Mean Squared Error (MSE)
- Using every observation in your training dataset compute

$$MSE_{Train} = \frac{1}{n}\sum_{i=1}^{n} res_i^2 = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2$$

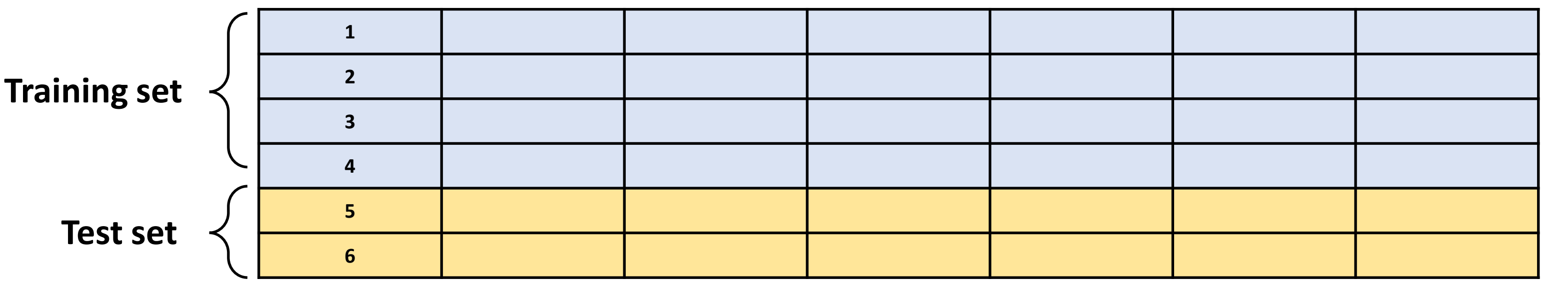
- Is this the right thing to do?

Measuring accuracy

- Is measuring accuracy on the training set the right thing to do?
- Not really – our estimate of the accuracy will be overly optimistic because future data will be different that the data we used to train on
- We want out model to "generalize" to unseen data
- How do we achieve this goal?

The train-test paradigm

- We are typically interested in out of sample (aka generalization) accuracy
- Instead, suppose we have a test dataset that was not used for training



- Compute MSE on test data
- For every observation y_i' in your test dataset compute
  - $res_i^2 = \left(\widehat{y_i'} - y_i'\right)^2$
  - Then compute MSE_{Test} by averaging up these test residuals
- Why is this better?
  - The data we will see in the future is unlikely to be the same as the data at hand
  - We want to avoid overfitting the specific draw of data we got
- What are some concerns with using a test set?

What is a “good” MSE

- Mean squared error easier to interpret if a take a square root
  - Why?

- We call this RMSE

- What is a good RMSE?

- Try fitting the simplest model possible: a linear regression just with an intercept term
  - What is the prediction $\hat{y}$ of this model for an observation X?
  - What is the RMSE of this simple model?
  - Use the RMSE of this very simple model as baseline to evaluate more sophisticated model.

- END