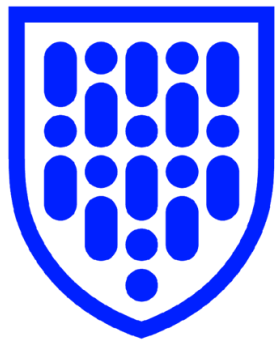


Institute of Data

Data Science and AI

Module 4

Introduction to Machine Learning/Regression

Institute of Data

Agenda: Module 4

- Introduction to **Machine Learning**
- The **Predictive** Modelling Process
- **Feature** Selection
- **Measuring** the Accuracy of Regression Models
- **Overfitting**

Institute of Data

Introduction to Machine Learning

- **Supervised** Learning
- **Unsupervised** Learning
- **Regression**
- **Classification**

"All models are **wrong**;

some are **useful**!"

George Edward Pelham Box (1919-2013)

Supervised Learning

- The main characteristic of **Supervised Learning** is to have previously known results
- Consider data collected from cases with known **input** and **output**
  - Cases can also be called observations, experiments, results, etc
  - Input can also be called variables; input, explanatory or independent variables; features; predictors
  - Output can also be called output or dependent variable; label; result; outcome; response
- Supervised learning relies on **induction**:
  - the inference of a general law from particular instances.

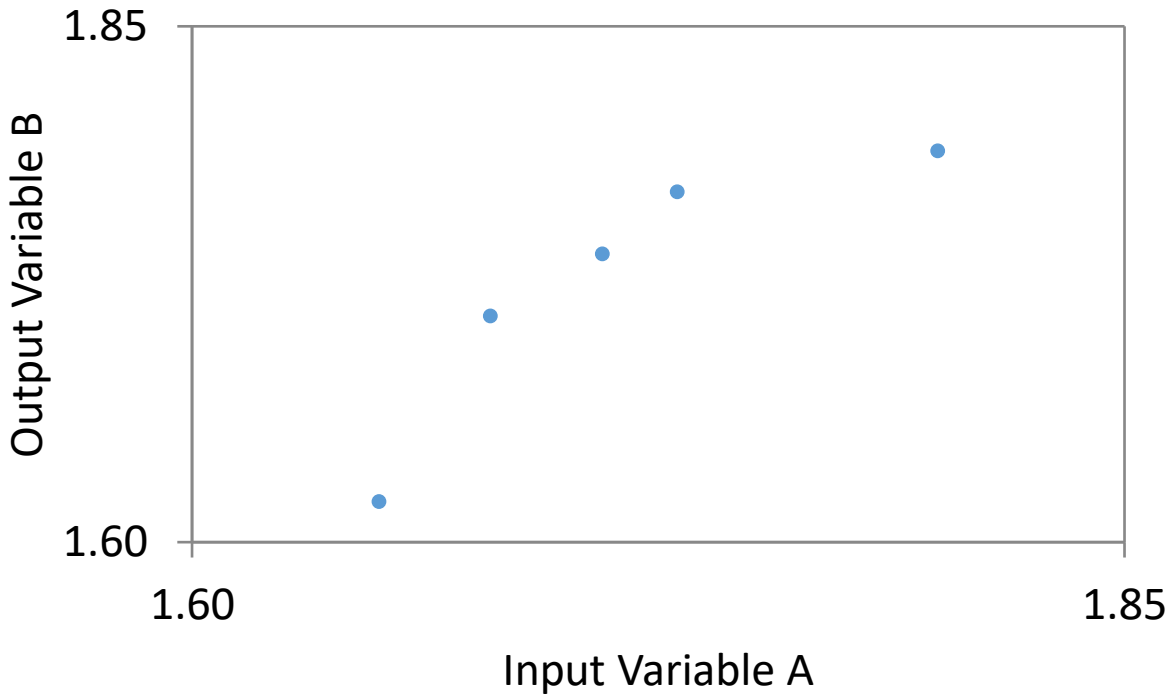
Supervised Learning

- Most **practical** Machine Learning uses Supervised Learning.
- Knowing the values of an input variable **A** and an output variable **B** can be the source of a model to predict future outcomes of **B** given a new input of **A**
- Definition:
  - Supervised learning is where you have input variables (x) and an output variable (Y) and you use an algorithm to learn the **mapping function** from the input to the output **Y = f(X)**

Supervised Learning

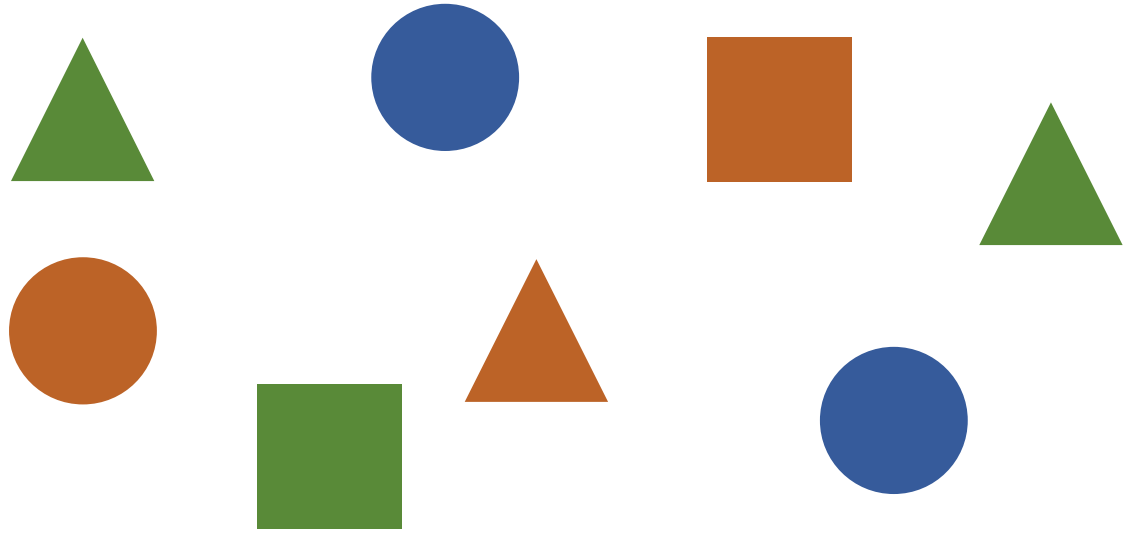
- The computer will find a function by the analysis of input-output pairs, so it can be used to predict future results in similar conditions

A	B
1.71	1.74
1.65	1.62
1.80	1.79
1.68	1.71
1.73	1.77



Unsupervised Learning

- The main characteristic of **Unsupervised Learning** is to have **unknown prior outcomes**
- Consider data collected from **observations** of which **only** the **input** is **known.** For example the collection of customers or transactions



Unsupervised Learning

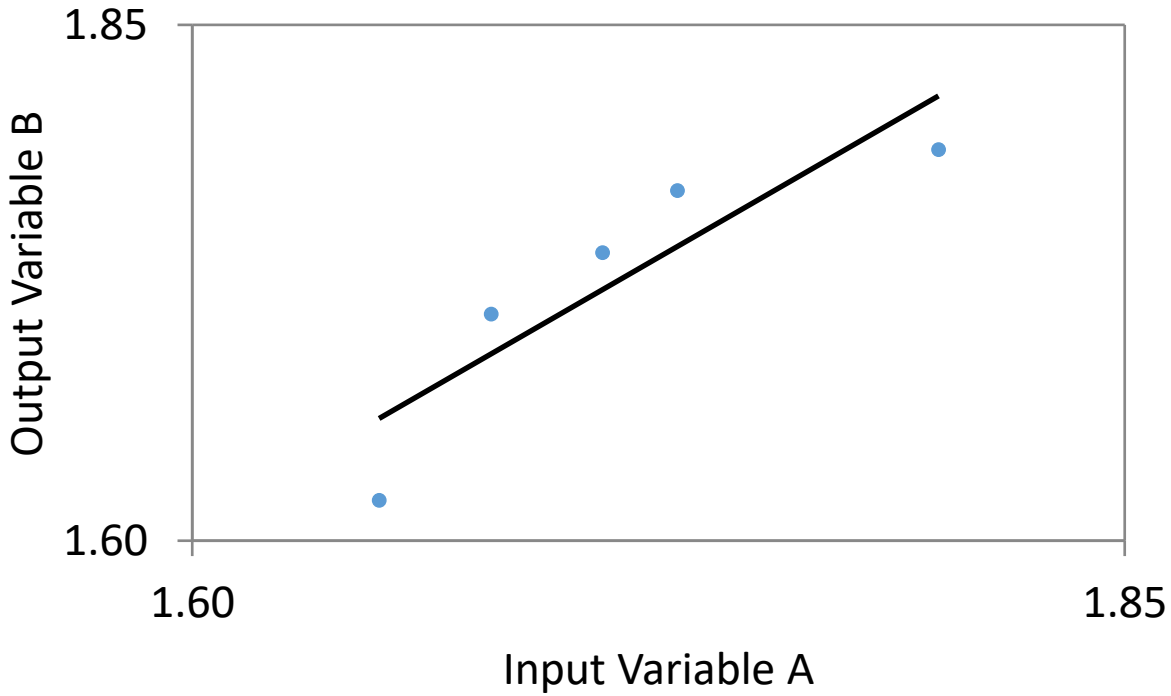
- The computer will infer a function by the analysis of **similar characteristics** or **proximity** of the input examples, so it can be used to predict future results in the same conditions

Regression

- What characterises **Regression** is to obtain a **scalar result**, a number
  - Regression results can have a value range from -∞ to ∞
- Tries to answer questions like: How Much / How Many?
  - What will the temperature be tomorrow?
  - How many new followers will be in the next week?
- Set of statistical processes for estimating the relationships among variables
  - One dependent variable: **univariate** analysis
  - More than one dependent variable: **multivariate** analysis

(Univariate) Regression

- A function of changes in any one independent variable (holding others fixed)



Classification

- What characterises **Classification** is to identify a **class** or **group**
  - Classification results can be a value within the domain set

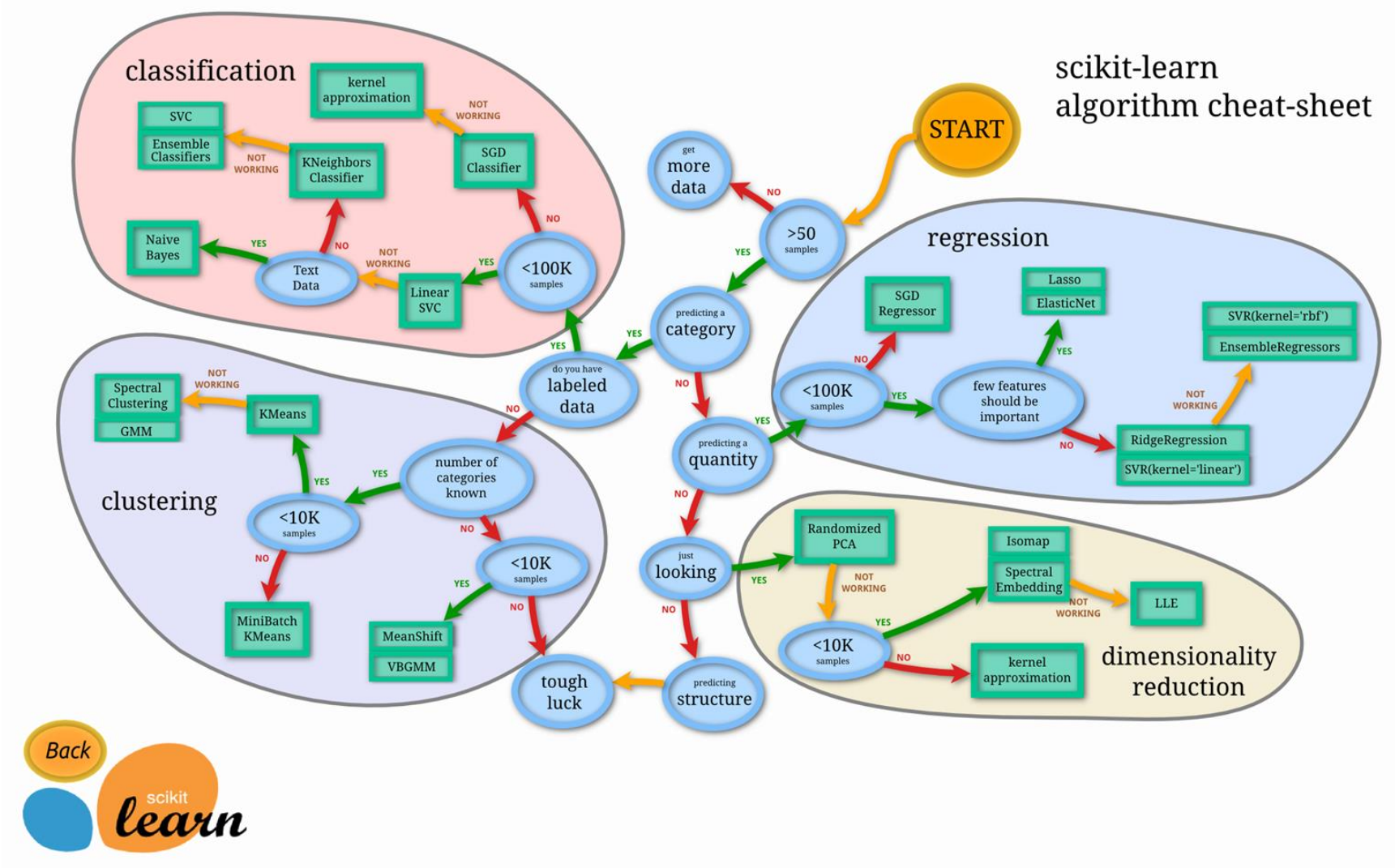
- Set of statistical processes to determine membership or similarity

Classification

- Two classes (Binary classification): (Yes, No), (Positive, Negative)
  - Answer questions like:
    - Will the customer renew their subscription?
    - Will the website visitor click the submit button?

- More than two classes: A set of possible categories, classes or groups
  - Answer questions like:
    - Which animal is in this image?
    - What is the genre of this movie?

Scikit-Learn Cheat-Sheet



All Together

	Supervised	Unsupervised
Discrete	Classification	Clustering
Continuous	Regression	Dimensionality Reduction

Institute of Data

The Predictive Modelling Process

- Outline
- Data Science **Process**
- Defining the **Response** Variable
- Defining the **Cost Function**
- A Familiar Example: **Univariate** Linear Regression
- **Multiple** Linear Regression
- **Partitioning** Data
- Model **Training**
- Model **Testing**

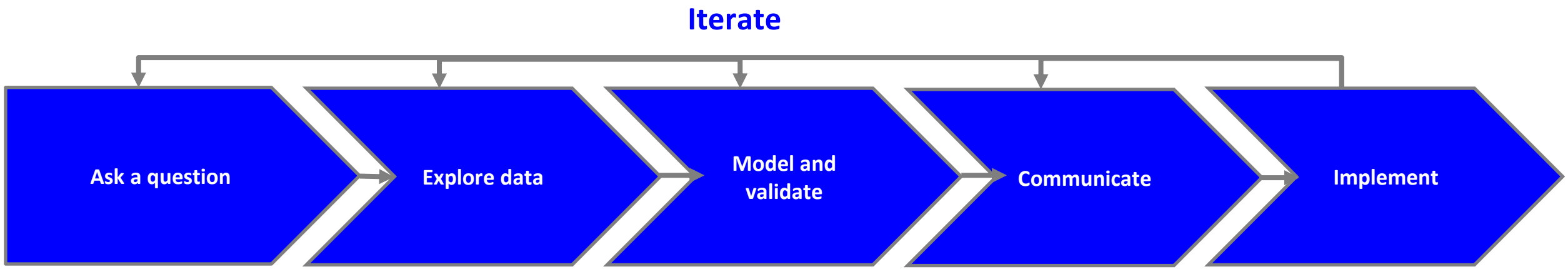
Outline

- **data** | ˈdeɪtə |, *noun*
  - **facts** and **statistics** collected together for reference or analysis
  - (Philosophy) things **known** or **assumed** as facts, making the basis of reasoning or calculation
  - ORIGIN: mid 17th century; from Latin, the plural of *datum*
- **science** | ˈsʌɪəns |, *noun*
  - the intellectual and practical activity encompassing the **systematic study** of the structure and behaviour of the physical and natural world through observation and experiment
  - ORIGIN: Middle English (denoting knowledge): from Old French, from Latin *scientia*, from *scire* ‘know’

Outline

- Data Science: **Systematic** use of data to identify insights and inform decisions
- Data science is an interdisciplinary field that uses scientific methods, processes, algorithms and systems to extract *information, knowledge and insights* from both structured and unstructured data.
- Data science is a "concept to unify statistics, data analysis, machine learning and their related methods" to "understand and analyse actual phenomena" **with data**. It employs techniques and theories drawn from many fields within the context of mathematics, statistics, information science and **computer science**.

Data Science Process



- Ask a (business) question
- Define objective and strategy

Most important step!

- Identify and collect, clean and transform data
- Explore data
- Ascertain quality and ability to answer questions

May consume large proportion of the total effort

- Determine target and feature variables
- Select model, apply and validate

Feature engineering requires deep domain knowledge

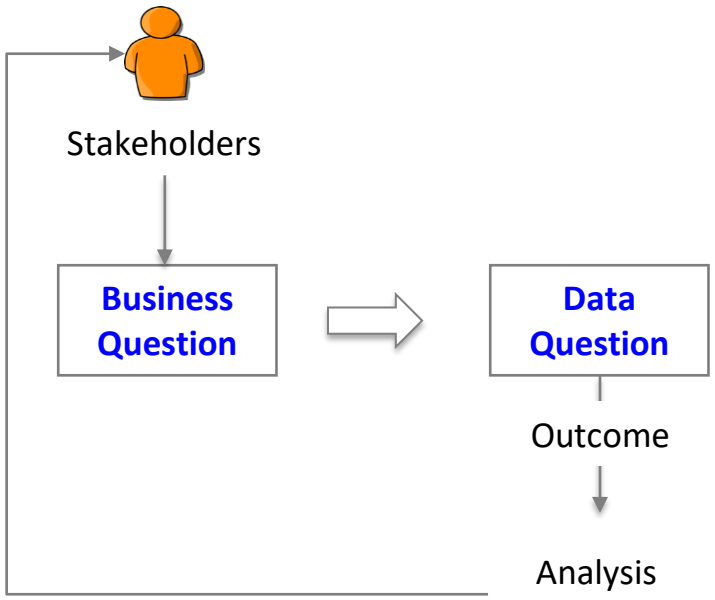
- Communicate to stakeholders and obtain buy in

Perform as early and as often as possible

- Develop end-to-end solution
- Build, test, deploy and monitor

Consider the total cost of implementation earlier in the process

Defining the Outcome/ Response Variable



- Your **question** leads to the selection of the outcome/ response variable:
  - To answer the question
  - You could frame it as a hypothesis that can be statistically tested

- The initial step to answer the question is to define which is the Response Variable and its characteristics, such as **type** and **range**

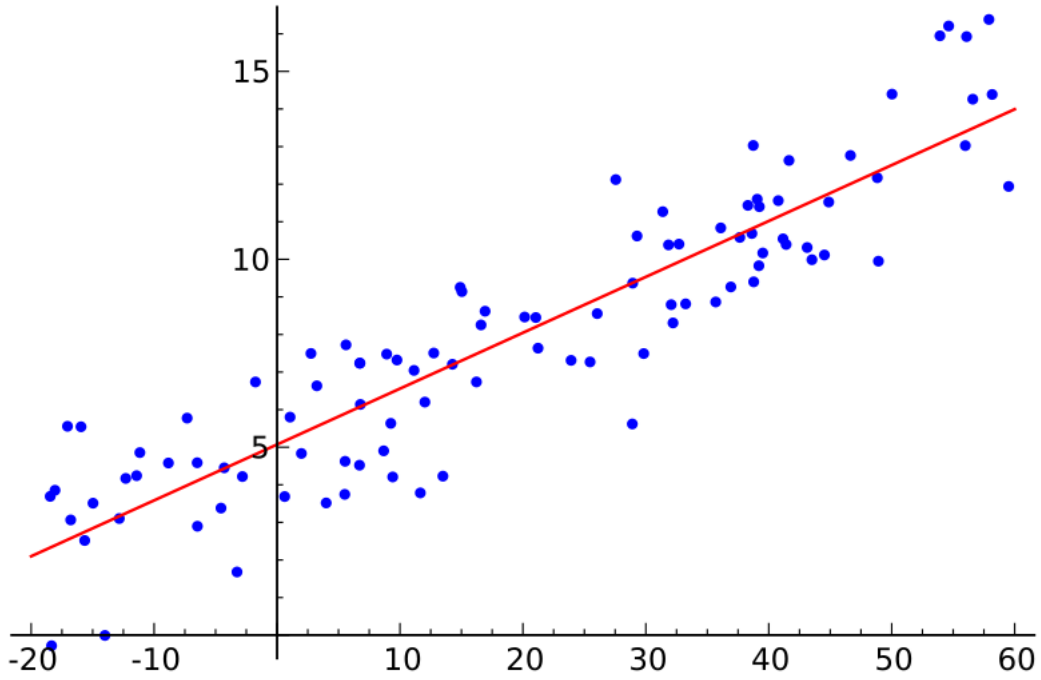
- Some **modelling techniques** that are only relevant to a particular type of variable, so knowing the nature of the response variable can reduce the number of approaches to consider

Defining the **Cost Function**

- A cost function calculates **how close the model can estimate the relationship** between the input variables (**X**) and output variable (**y**)
- In other words, it describes **how wrong the model is** by how far off its predictions are in comparison to reality
- Some traditional statistical methods such Linear Regression have error metrics associated with it, like MSE (Mean Squared Error)

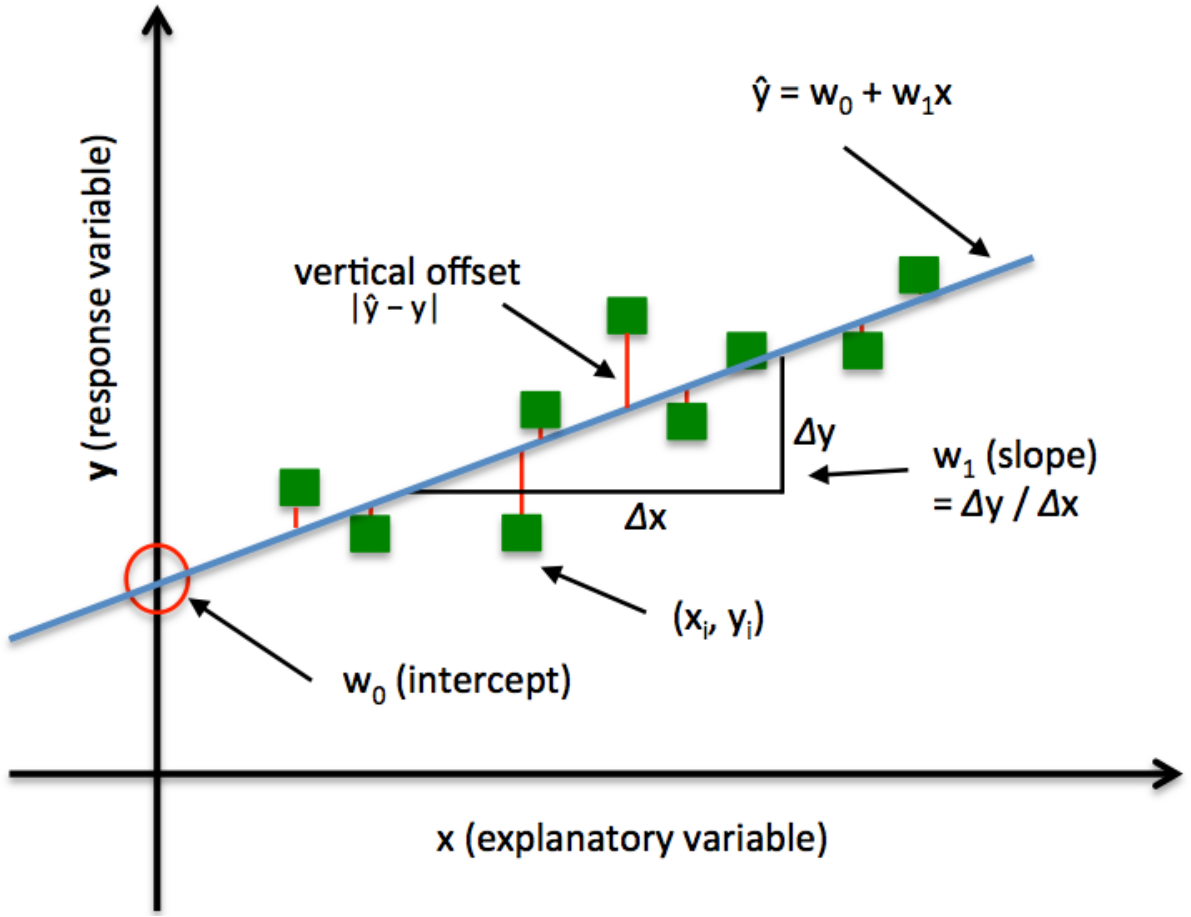
A Familiar Example:
Univariate Linear Regression

- The relationship between a **continuous variable** and other independent variables
- Explain the relationship between ***x*** and ***y*** using the starting point ***a*** and the power in explanation ***b***
- The simplest version is just a **line** of best fit



Source: Wikipedia

Example:
Univariate Linear Regression



Source: MLxtend

Multiple Linear Regression

- However, linear regression uses linear algebra to explain the relationship between ***y*** and multiple ***x***

- Explains the relationship
  - Between a Matrix ***X*** and a Dependent Vector ***y***
  - By using a y-intercept **alpha** and the relative coefficients **beta**

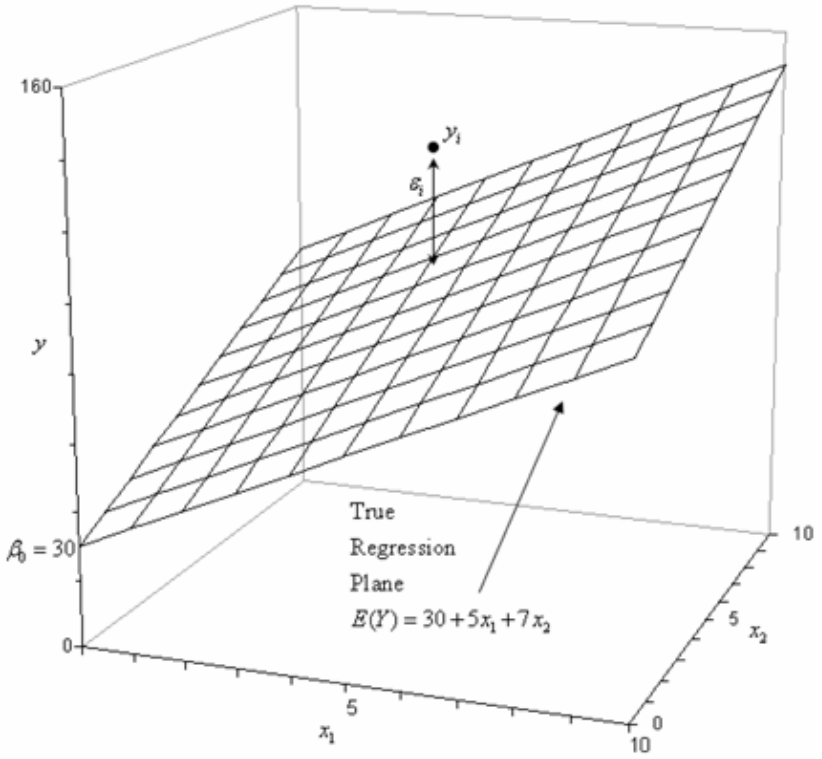
Multiple Linear Regression

$$\boldsymbol{y} = \boldsymbol{\alpha} + \boldsymbol{\beta}\,.\boldsymbol{X} + \boldsymbol{\varepsilon}$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \cdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & x_{1,3} & \cdots & x_{1,m} \\ 1 & x_{2,1} & x_{2,2} & x_{2,3} & \cdots & x_{2,m} \\ 1 & x_{3,1} & x_{3,2} & x_{3,3} & \cdots & x_{3,m} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & x_{n,1} & x_{n,2} & x_{n,3} & \cdots & x_{n,m} \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta_1 \\ \beta_2 \\ \beta_3 \\ \cdots \\ \beta_m \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \cdots \\ \epsilon_n \end{bmatrix}$$

Multiple Linear Regression

$$y = \alpha + \beta . X + \varepsilon$$



Source: Reliawiki

Source: MathWorks

Linear Regression

- Linear Regression works best when
  - The data is **normally** distributed (but does not have to be)
  - Xs significantly **explain** y (have low p-values)
  - Xs are **independent** from one another (low multicollinearity/ correlation)

Model Training, evaluation and testing

- Application of one or more modelling techniques on the training data aiming for a model with either or both **better performance** and **lower cost**
- Collect **evaluation** results for each technique, each set of variables
- **Refine** and **select** the best model for testing
- Optionally explore a **combination** of techniques
- **Test** and compare metrics with previous results

Discussion

What do you think about the terms **“Data Science” and “Machine Learning”?**

What do they mean?

Are they meaningful/ practical term?

Any Questions?

Partitioning Data

- Models are **simplified representations of reality** created from data that can be used to estimate future outcomes better

- To validate the models, it is necessary **to simulate existing data and future data**, hence the need to use **existing** data to play those roles

- The **Quality** of data is more critical than the **Quantity** of data
  - Data must be **representative**
  - Large quantities may incur in higher storage and computation costs without improving the model's performance

Partitioning Data

- Existing data can be **split** into (percentages shown are common, not a rule):
  - Training (70%) and Testing (30%), or
  - Training (65%), Validation (20%) and Testing (15%)

- Training (Learning) Data is used during the development of the model
- Validation Data can be used to improve the performance of the model
- Test Data is used to check the performance of the model with unseen data

- Data split must be **random**

Lab 4.1.1: Linear Regression from scratch

- Purpose
  - To develop familiarity with the Data Science Process
- Resources
  - Sample data
- Materials
  - Jupyter Notebook (Lab-4.1.1)

Lab 4.1.2: Splitting Data

- Purpose
  - To develop familiarity with the Data Science Process
  - Split data for model development and evaluation
- Resources
  - Sample data from SciKit-Learn
- Materials
  - Jupyter Notebook (Lab-4.1.2)

Institute of Data

Feature Selection

- Forward and Backward Stepwise Feature Selection
- Feature Engineering

Forward and Backward Stepwise Feature Selection

- **Occam's razor** (also **Ockham's razor** or "law of parsimony") is the problem-solving principle that the simplest solution tends to be the right one.
  - **When presented with competing hypotheses to solve a problem, one should select the solution with the fewest assumptions**
  - The idea is attributed to William of Ockham (c. 1287–1347), who was an English Franciscan friar, scholastic philosopher, and theologian

(*Wikipedia*)

Forward and Backward Stepwise Feature Selection

- Forward Feature Selection
  - Take **each** of the predictors **individually** and select the variable from the model that explains the output variable the best
  - Repeat the process by **adding** the remaining predictors **individually** until there is **no** significant improvement

- Backward Feature Selection
  - Take **all** of the predictors and create a model
  - Remove each of the predictors individually and keep the model that explains the output variable the best
  - Repeat the process by removing the remaining predictors individually until there is a significant reduction in performance

Feature Engineering

- **Feature Engineering** can be described as the process of **transforming** the variables into forms or domains **suitable** both to the **underlying problem** and to computation
- Modelling techniques manipulate **numbers** to optimise parameters and coefficients used in functions
- Not all techniques can manipulate nominal and ordinal variables (classes, categories)

Feature Engineering

- There are **no specific rules** in cases when the data manipulation relates to **particular problems**, industries or businesses

- Some data manipulations are mode specific to the chosen modelling approaches, such:
  - Variables transformation from nominal/ordinal to numeric
    - A common practice is to use one hot encoding
  - **Change of scale** or range for numeric variables
    - Variables with **different orders of magnitude** (A:0-10, B:300-9000) can be **normalised** to be closer to each other (A:0-1, B:0-1)

Lab 4.2: Feature Selection

- Purpose
  - To understand Forward Feature Selection
- Resources
  - Sample data from SciKit-Learn
- Materials
  - Jupyter Notebook (Lab-4_2_1, Lab-4_2_2)

Institute of Data

Measuring the Accuracy of Regression Models

- Introduction
- **R-Squared**
- **Mean Squared Error** (MSE)

Introduction

- It is possible to compare the known results with the ones predicted by the model when using **Supervised modelling**
- When comparing known and predict results it is possible to make calculations and create metrics on how close or similar there outcomes are
- Such metrics can both tell how **“good”** or how **“efficient”** a model is and how it compares to other models as most metrics have standard results

R-Squared

- Is the central metric introduced for Linear Regression
- It determines how much of the variation in ***y*** is **explained** by the change in ***X***
  - But does it tell the magnitude or scale of error?
- The value range is between:
  - 0: the model does not model explain any variability in ***y***
  - 1: the model **explains full variability** in the target variable
- Exploring cost or loss functions can be used to **refine** the models

R-Squared and Sum of Squares (SS)

Total SS

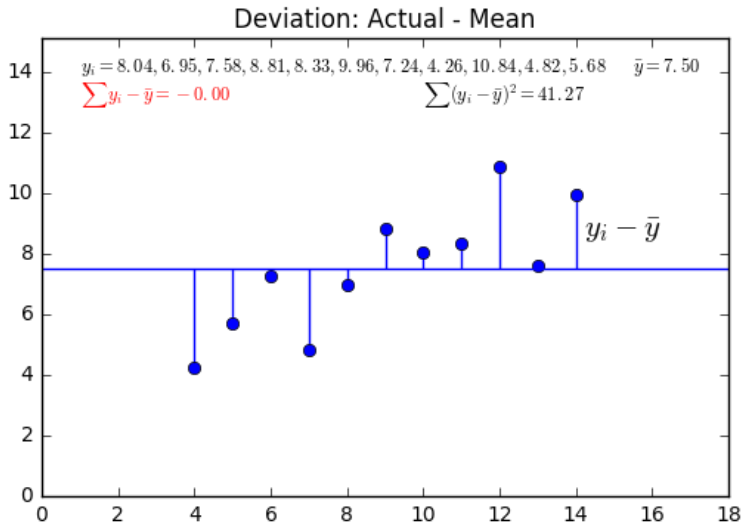
$$SS_{tot} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

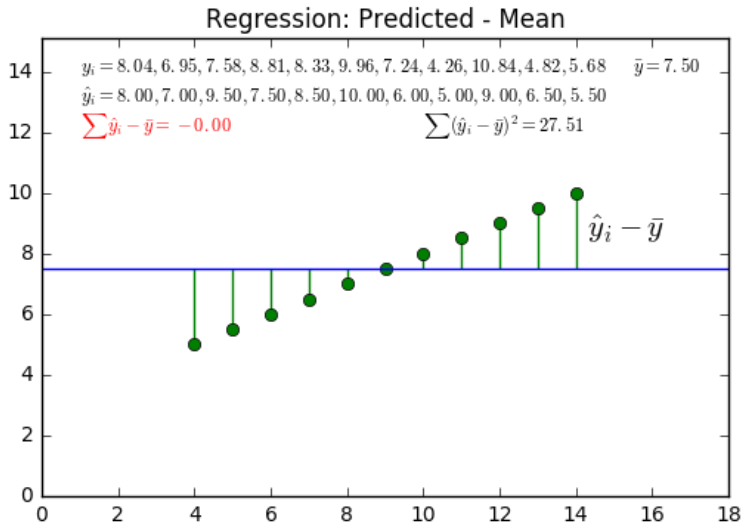
Regression SS

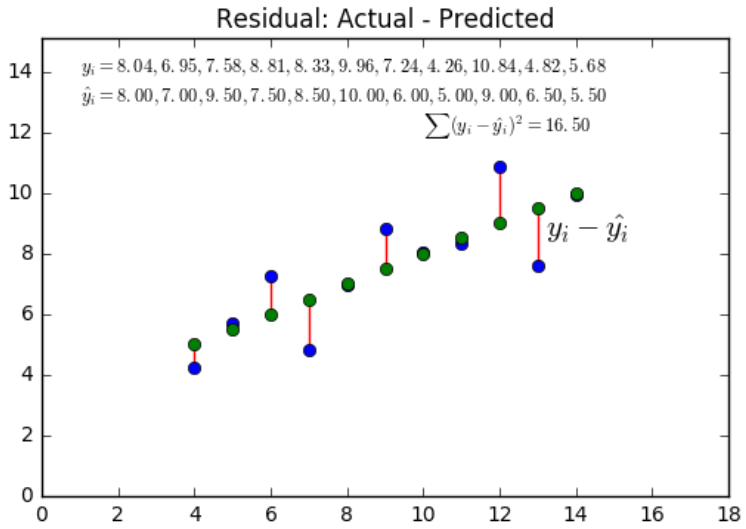
$$SS_{reg} = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$$

Residual SS

$$SS_{res} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$







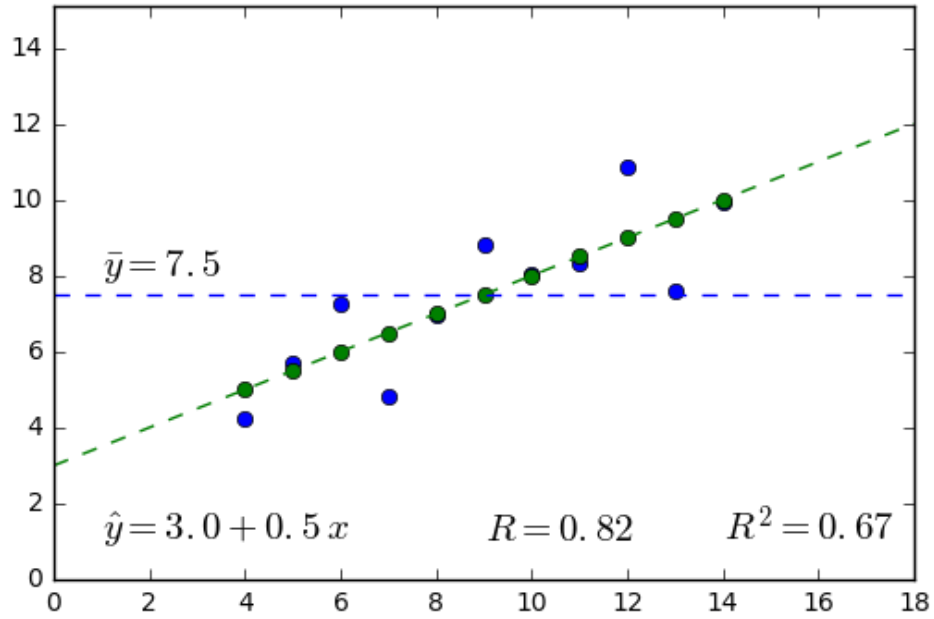
R-Squared and Residuals

- In a data set with **n** cases where the outputs are: $y_i = y_1, \ldots, y_n$
- And the predicted values are (note the *y-hat*): $\hat{y}_i = \hat{y}_1, \ldots, \hat{y}_n$
- SS: Sum of Squares
- The **mean of y** is (note the *y-bar*):

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

$$SS_{tot} = SS_{reg} + SS_{res}$$

$$R^2 = \frac{SS_{reg}}{SS_{tot}}$$



Adjusted R-Squared

- The Adjusted R-Squared is the R-Squared adjusted for the number of predictors in the model
- It incorporates a model's degrees of freedom
- Adjusted R-Squared only increases if the new term improves the model accuracy

$$Adjusted\ R^2 = 1 - (1 - R^2)\left[\frac{N-1}{N-(p+1)}\right]$$

- where

  R2 = Sample R-Squared

  N = Total sample size

  p = Number of predictors

Residual Error

- In linear models, the **residual error** must be normal with a median close to zero
- Individual residuals are useful to see the error of specific points, but it does not provide an overall picture for optimisation
- It is necessary a metric to summarise the error in the model into one value
  - **Mean Squared Error**: the mean residual error in our model

Mean Squared Error (MSE)

- Calculate the difference between each target ***y*** and the model's predicted value ***y-hat*** (i.e. the residual)
- Take the mean of the squared residual errors

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

- The MSE is always greater or equal to zero, with smaller errors being better
- An MSE of 0 means that actual and predicted results are identical

Minimising the Error, MSE as Cost Function

- The linear regression method with MSE is also known as:
  - Ordinary Least Squares
- Meaning that given a matrix ***X***
  - Solve for the least amount of square error for ***y***
- Assuming that ***X*** is unbiased
  - Saying it is representative of the population

Lab 4.3: Measurements

- Purpose
  - To compare model complexity
- Resources
  - Sample data from SciKit-Learn
- Materials
  - Jupyter Notebook (Lab-4_3)

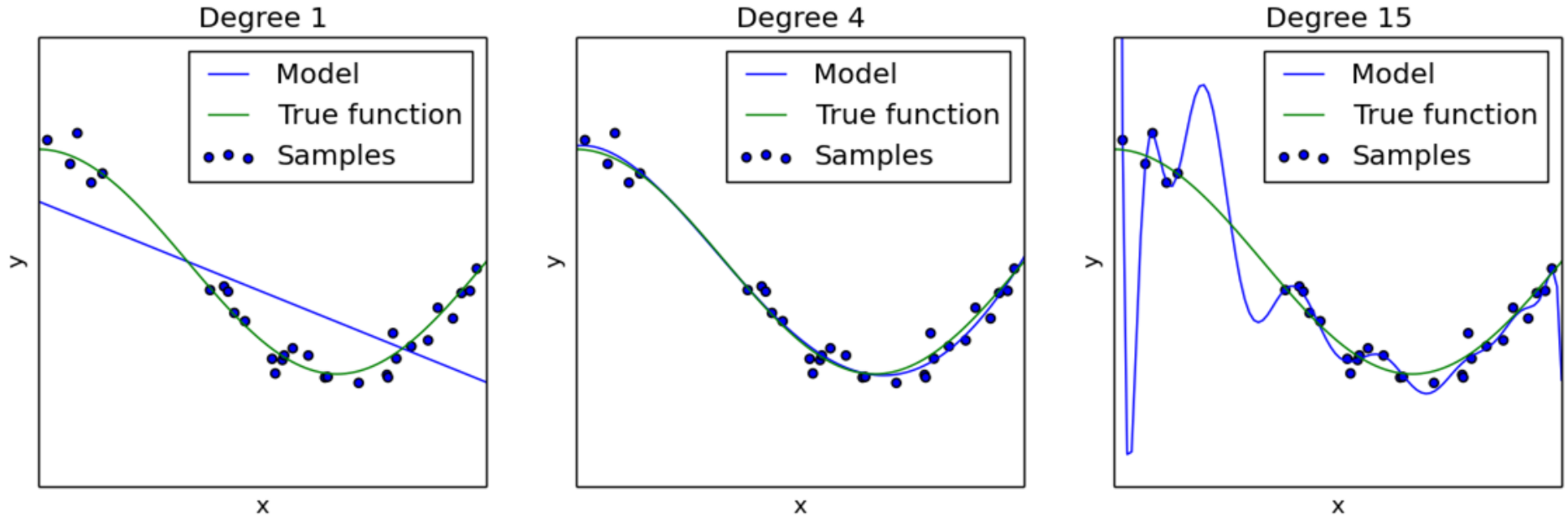
Institute of Data

Overfitting

- **Detecting** Overfitting
- **Avoiding** Overfitting
- **Cross-Validation**
- **Regularisation**

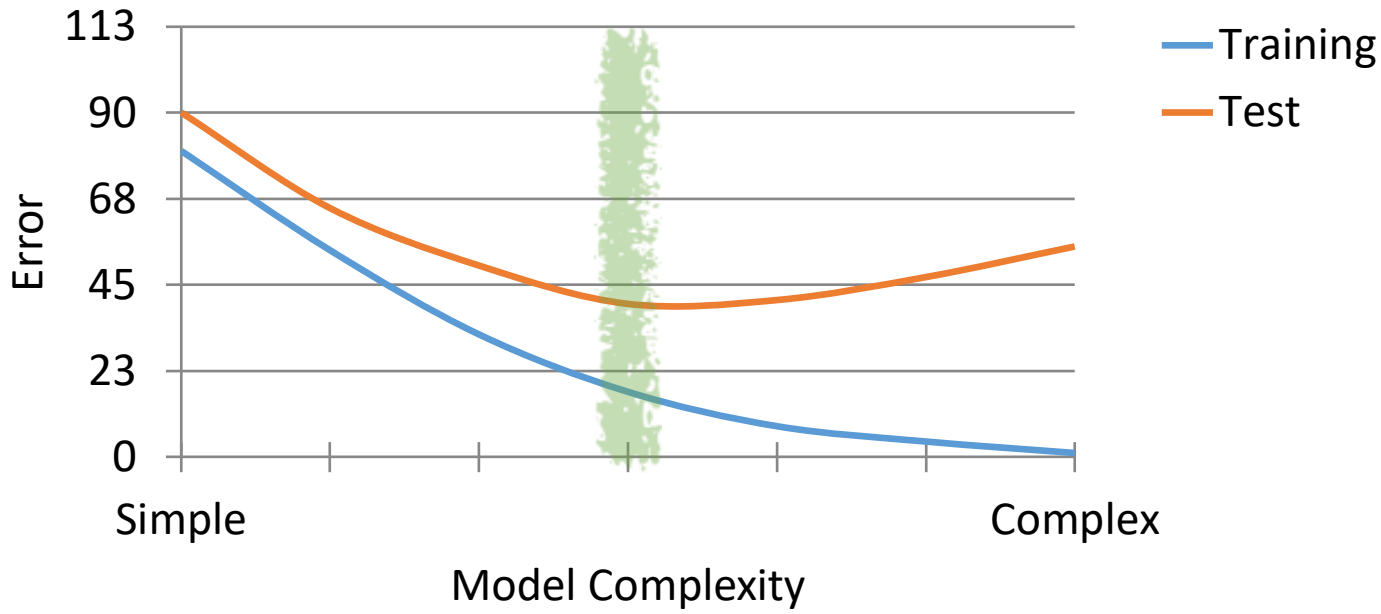
Overfitting

- The first model **poorly explains** the data (underfitting)
- The second model describes **the general curve** of the data
- The third model drastically **overfits** the model, bending to every point

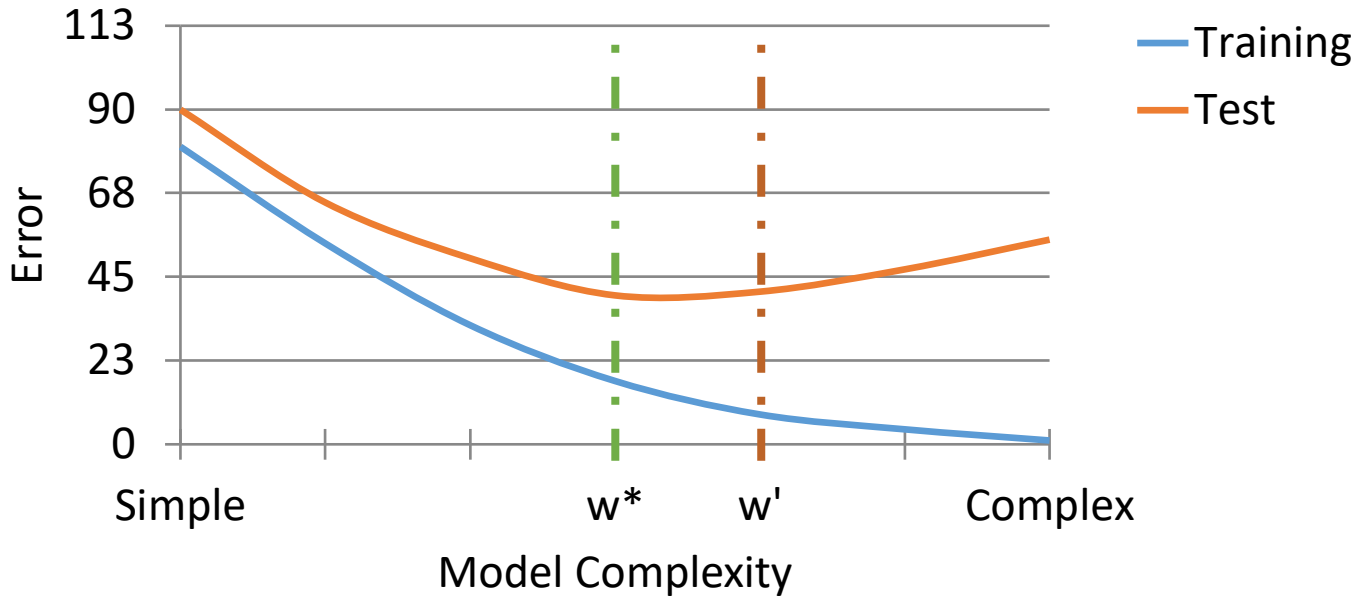


Detecting Overfitting

- Mathematically, modelling is to **fit** a curve to a set of known data points, so overfitting is to find a curve that fits the known data points very closely but does **not** generalise well to **unknown** data points



Detecting Overfitting



- Overfitting:
  - Training Error(w*) > Training Error(w')
  - Test Error(w*) < Test Error(w')

Avoiding Overfitting

- Have the data split into **a training** set and a **test** set
  - Works well only with the availability of large datasets
- To find out the best model (training error) the **residual sum of squares** is used
- Varies model complexity
  - Includes varying the set of **variable**s and data manipulation
- Continuously updated the parameters/coefficients/weights
- Use techniques such as **Cross-Validation** and **Regularisation**

Cross-Validation

- The goal of cross-validation is to test a model's ability to predict new data that was not used in estimating it
  - Used to flag problems like overfitting and to give an insight on how the model will **generalise** to an independent dataset

- General Process
  - Generate several models on different **cross-sections** of the data
    - Partition the data in separate data sets **alternating their roles** as Training and Test
  - Measure the **performance** of each iteration
  - Take the **overall performance**
    - Usually the mean of the collected results

Common types of Cross-Validation
Exhaustive cross-validation

- All distinct ways to divide the original sample are used to learn and test

- Leave-p-out
  - Use ***p*** observations as the test set and the remaining as the training set
  - This is repeated in all ways to cut the original sample on a validation set of ***p*** observations and a training set

Common types of Cross-Validation
Non-exhaustive cross-validation

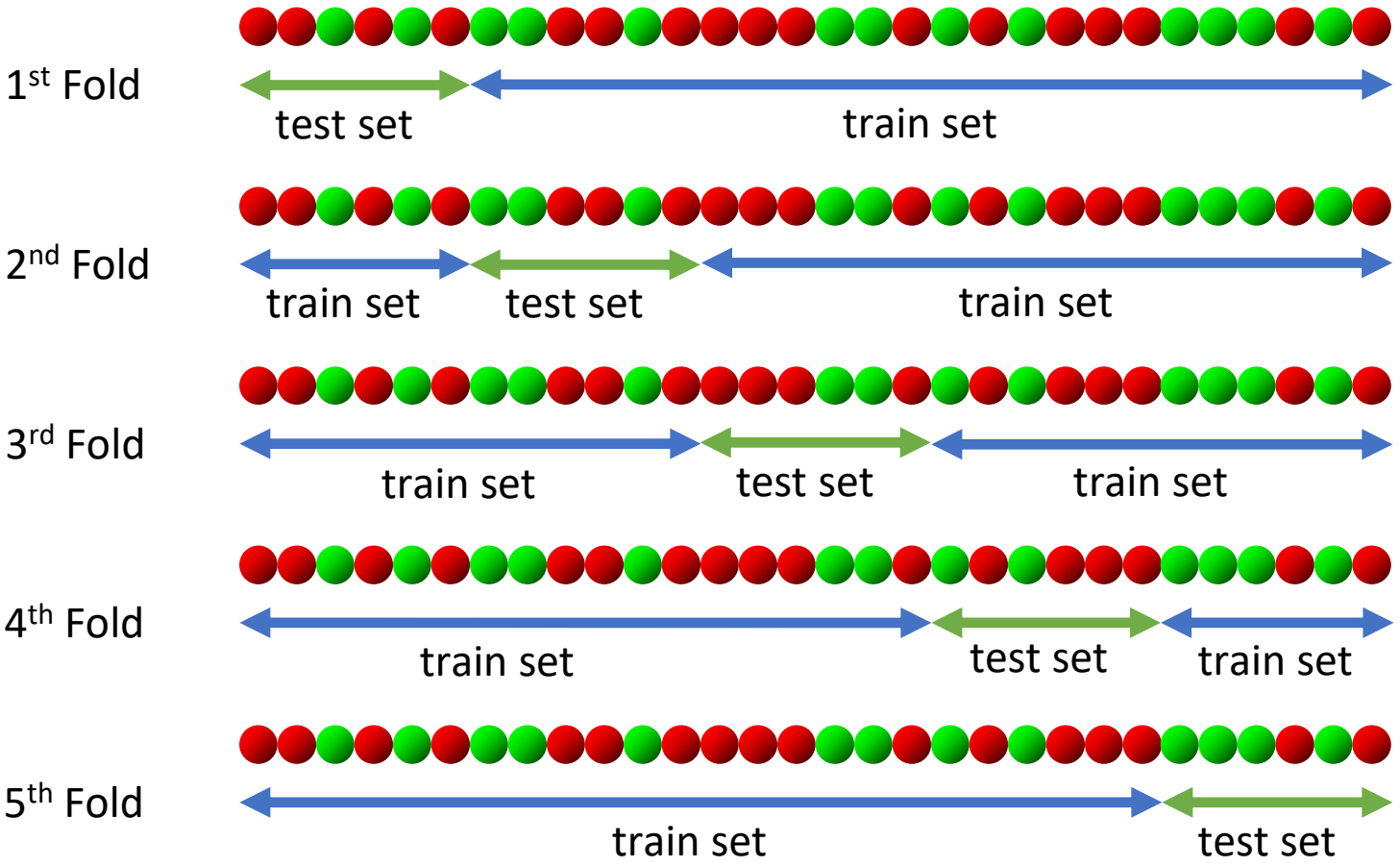
- Do not compute all ways of splitting the original sample; these methods are **approximations** of leave-p-out cross-validation
- k-fold
  - Randomly partition the original sample into **k equal sized subsamples**
  - In stratified k-fold cross-validation, select folds so that the **mean response value is approximately equal** in all the folds
- Holdout method
  - **Randomly** assign data points to two sets ***d0*** and ***d1***, usually called the training and the test sets, respectively
  - The test set is usually smaller than the training set

k-Fold Cross-Validation

- k-fold cross-validation
  - Split the data into **k group**
  - **Train** the model in all segments **except one**
  - **Test** model performance on the **remaining set**

- If k = 5, split the data into five segments and generate five models

Cross-Validation



Cross-Validation

- Computational issues
  - Straightforward to implement if the **prediction method** being studied is available
  - If the prediction method is **expensive** to train, cross-validation can be very slow since the training must be carried out repeatedly

- Limitations
  - Cross-validation only yields meaningful results if the validation set and training set are drawn from the same population and only if human biases are controlled

- Cross validation for time-series models
  - Since the order of the data is important, cross-validation might be problematic for time-series models
  - A more appropriate approach might be to use **forward chaining**

Regularisation

- Form of regression, that **constrains or shrinks** the coefficient estimates towards zero
  - This technique discourages a more complex model to avoid the risk of overfitting

- Given a set of ***p*** coefficient (β_p), the fitting procedure involves a loss function (**RSS** or $\mathbf{SS_{res}}$)
  - The coefficients are chosen, such that they minimise the loss function by adding an “extra cost”

- The estimated coefficients do not generalise well with unseen data if there is noise in the training data

Regularisation

- The are multiple forms or regularisation, the most common being:
  - **Ridge Regression**
    - Adds cost by the inclusion of a new parameter called lambda (***λ***)
  - **Lasso Regression**
    - Adds cost by the addition of a new parameter called lambda (***λ***)
    - Can force coefficients to be zero, hence effectively removing them from the model

Regularisation - Ridge Regression

- The RSS is added a shrinkage quantity $\boldsymbol{\lambda}$

$$RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$

- The coefficients are estimated by minimising the new function
- The coefficient $\boldsymbol{\lambda}$ is the parameter that controls how much to penalise the model
- The Ridge Regression force coefficients to stay low

Regularisation - Ridge Regression

- When $\lambda=0$, the penalty term has no effect and the estimates produced by Ridge regression will be equal to Least Squares

- It is necessary to standardise the predictors or bring the predictors to the same scale before performing Ridge regression

Regularisation - Lasso Regression

- It penalises the high coefficients by using $|\beta_j|$

$$RSS + \lambda \sum_{j=1}^{p} |\beta_j|$$

- Ridge regression will shrink the coefficients for least important predictors, very close to zero, but never make them precisely zero
  - The final model will include all predictors

- Lasso regression can force some of the coefficient estimates to be equal to zero when the tuning parameter λ is sufficiently large
  - The lasso method also performs **variable selection** and is said to yield sparse models

Lab 4.4: Regularisation

- Purpose
  - To compare some regularisation approaches
- Resources
  - Sample data from SciKit-Learn
- Materials
  - Jupyter Notebook (Lab-4_4)

Questions?

End of Presentation!