



# COS10022 – DATA SCIENCE PRINCIPLES

Dr Pei-Wei Tsai (Lecturer, Unit Convenor)  
ptsai@swin.edu.au, EN508d

SWIN  
BUR  
NE

SWINBURNE  
UNIVERSITY OF  
TECHNOLOGY





# WEEK 02 - REGRESSIONS

## Linear Regression

SWIN  
BUR  
NE

SWINBURNE  
UNIVERSITY OF  
TECHNOLOGY

# Key Questions



What are the distinctions between the model planning and model building phase?



What are some key considerations in model building?



What software tools (commercial, open source) are typically used at this phase?



What is Linear Regression model and in what situation is it appropriate?



How does the Linear Regression model work for predictive modelling tasks?



How do we prepare our data prior to applying the Linear Regression model?



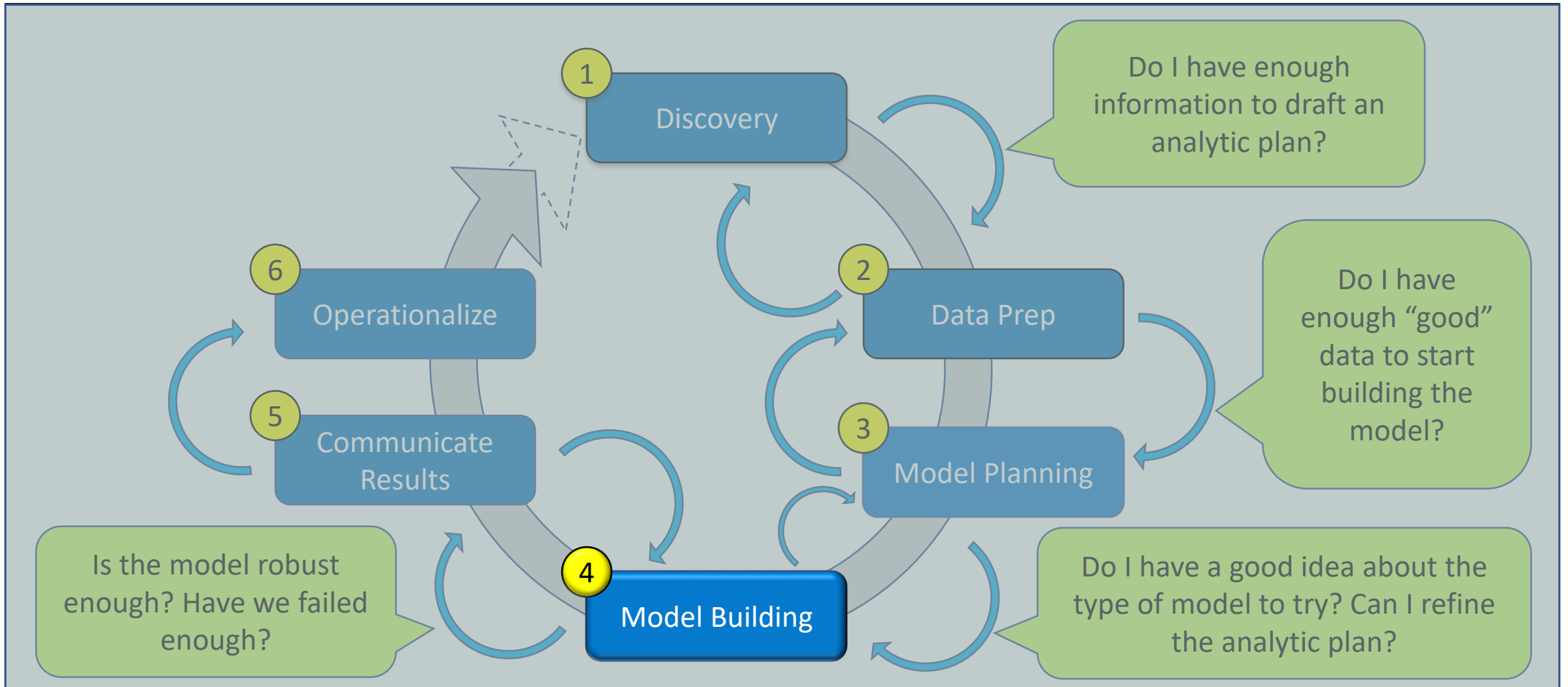
# Learning Outcomes

This lecture supports the achievement of the following learning outcomes:

3. **Describe the processes within the Data Analytics Lifecycle.**
4. **Analyse business and organisational problems and formulate them into data science tasks.**
5. **Evaluate suitable techniques and tools for specific data science tasks.**
6. **Develop analytics plan for a given business case study.**



# Data Analytics Lifecycle – Phase 4



# Phase 4 – Model Building



In the 4<sup>th</sup> phase of the Data Analytics Lifecycle (DAL), the data science team develops datasets for testing, training, and production purposes. The team builds and executes models based on the work done in the model planning phase.

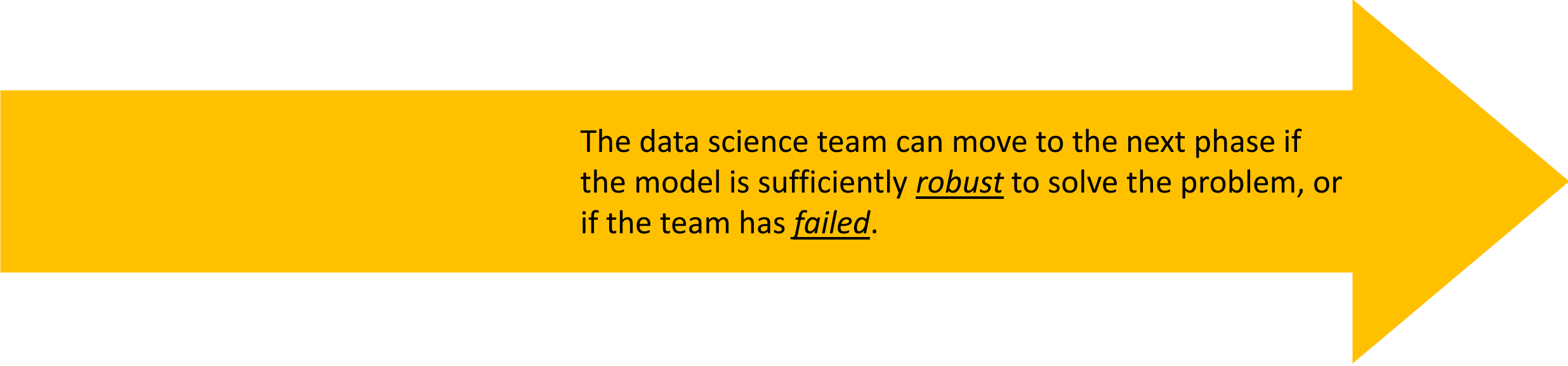


The team also considers the sufficiency of the existing tools to run the models, or whether a more robust environment for executing the models is needed (e.g., fast hardware, parallel processing, etc.).

# Phase 4 – Model Building

Key activities:

**Develop analytical model, fit it on the training data, and evaluate its performance on the test data.**



The data science team can move to the next phase if the model is sufficiently robust to solve the problem, or if the team has failed.

# Phase 4 – Model Building

In this phase, an analytical model is developed and fit on the training dataset, and subsequently evaluated against the test dataset. Both the model planning (Phase 3) and model building phases can overlap, where a data science team iterate back and forth between these two phases before settling on the final model.

- By '**developed**', we do not always mean coding an **entirely new analytics model** from scratch. Rather, this usually involves selecting and experimenting with **various models**, and where applicable, **fine tuning their parameters**.

Although some modelling techniques can be quite complex, the actual duration of model building can be short in comparison with the time spent for preparing data and planning the model.



# Phase 4 – Model Building

## **Documentation is important at this stage.**

When immersed in the details of building models and transforming data, many small decisions are often made about the data and the approach for modeling. These details can be easily forgotten once the project is completed.

It is **vital** to record the results and logic of the model during this phase. One must also take care to record any operating **assumptions** made concerning the data or the context during the modeling process.



# Phase 4 – Model Building

*(reproduced from Lecture 03)*

## Commercial tools used in this phase:

[SAS Enterprise Miner](#) – allows users to run predictive and descriptive models based on large volumes of data from across the enterprise.

[IBM SPSS Modeler](#) – offers methods to explore and analyze data through GUI.

[Matlab](#) – provides a high-level language for performing a variety of data analytics, algorithms, and data exploration.

[Chorus 6](#) – provides a GUI front end for users to develop analytic workflows and interact with Big Data tools and platforms on the back end.

... and many other well-regarded data mining tools e.g. [STATA](#), [STATISTICA](#) , [Mathematica](#)

# Phase 4 – Model Building

*(reproduced from Lecture 03)*

## Open-source tools:

[R](#) and [PL/R](#) – PL/R is a procedural language for PostgreSQL with R which allows R commands to be executed in database.

[Octave](#) – programming language for computational modeling, with some of the functionalities of Matlab.

[WEKA](#) – data mining package with an analytic workbench and rich Java API.

[Python](#) – offers rich machine learning and data visualization packages: scikit-learn, NumPy, SciPy, pandas and matplotlib.

[MADlib](#) – provides machine learning library of algorithms than can be executed in-database, for PostgreSQL or Greenplum.

[KNIME](#) – GUI ready software for easier data processing.

# Predictive Models

**Predictive models** are data analytics models/algorithms/techniques used for **predicting certain attributes of a given object**.

Examples:

1. A predictive model can be used for guessing whether a customer will “subscribe” or “not subscribe” to a certain product or service.
2. Alternatively, a predictive model may be used to predict whether a patient would “survive” or “not survive” a specific disease.

The goal of a **predictive model** greatly differs from the goal of **unsupervised models** (e.g. *k*-Means Clustering) which are limited to **finding specific patterns or structures** within the data (e.g. clusters or segments).



# Predictive Models

Predicting a **categorical** attribute of objects are usually solved as **classification** problems. In a classification problem, a model is presented with a set of data examples that are already labeled (**training dataset**). After learning from these examples, the model then attempts to label new, previously unseen set of data (**test dataset**).

Given the utilisation of training set, most classification models are categorised as **supervised** models.

Training set

Test set

input variables					output / class variable
#	job	marital	education	loan	subscribed
1	management	single	tertiary	no	no
2	entrepreneur	married	secondary	yes	no
3	services	divorced	tertiary	no	yes
4	management	married	tertiary	yes	no
5	management	single	secondary	yes	no
6	management	divorced	secondary	yes	yes
7	blue-collar	married	primary	yes	no
8	admin	married	tertiary	no	
9	management	single	tertiary	no	
10	blue-collar	single	secondary	yes	

Class labels ('yes', 'no')

Class labels to be predicted

# Training and Test sets

- **Training dataset:** the portion of data used to discover a **predictive relationship**.
- **Test dataset:** the portion of data used to assess the strength and utility of a **predictive relationship**.

The training and test datasets are usually **independent** from each other (non-overlapping).

In addition to splitting a dataset into training and test sets, it is also common to set aside a certain portion of the dataset as a **validation set** to improve the performance of a model.

**Validation dataset** is the portion of data that is used to minimize the possible overfitting of a model and select the optimal model parameters (*more of these in the next lecture*).



There is no general rule for how you should partition the data!

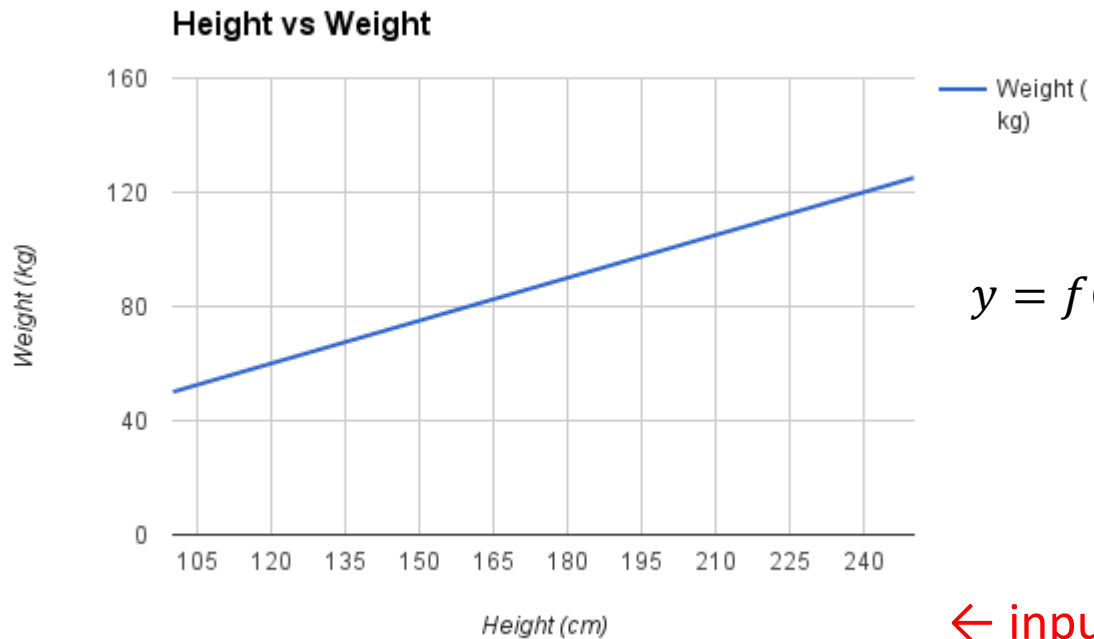
# Linear Regression Model

Linear Regression is considered as one of the **oldest supervised/predictive models** (more than 200 years old). Its goal is to understand the relationship between input and output variables. The model assumes that the output variable (i.e. predicted variable) is **numerical** and that **a linear relationship** exists between the input variables and the single output variable. The value of the output variable is calculated from a linear combination of the input variables.

- **Advantages:**
  - (a) simplicity;
  - (b) gives optimal results when the relationships between the input and output variables are linear.
- **Disadvantages:**
  - (a) limited to predicting numerical values;
  - (b) will not work for modeling non-linear relationships.

# Linear Regression Model

output (y) →

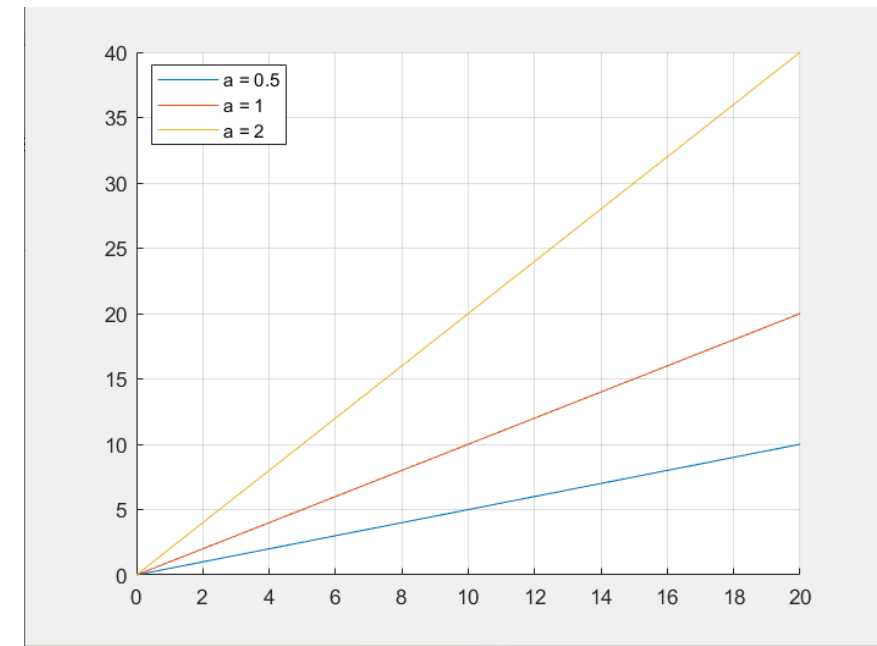


$$y = f(x) = a \cdot x + b$$

← input (x)

**Sample of a linear relationship between weight and height data.**

Source: <http://machinelearningmastery.com/linear-regression-for-machine-learning/>





# Linear Regression Model

Linear Regression belongs to what is called as **parametric learning** or **parameter modeling** approach.

Following this approach, building a predictive model starts with specifying the structure of the model with certain numeric parameters left unspecified. The objective of the model building is to **estimate the best values for these parameters** from the training data.

Linear Regression's model involves a set of parameterized numerical attributes. These attributes can be chosen based on domain knowledge regarding which attributes are likely to be informative in predicting the target variable, or based on more objective methods such as attribute selection techniques.

# Linear Regression Model

The Linear Regression model:

$$y = \overset{\text{weight}}{w_0} + \overset{\text{height}}{w_1 x_1} + \overset{\text{age}}{w_2 x_2} + \dots$$

where:

$y$  is the predicted output variable;

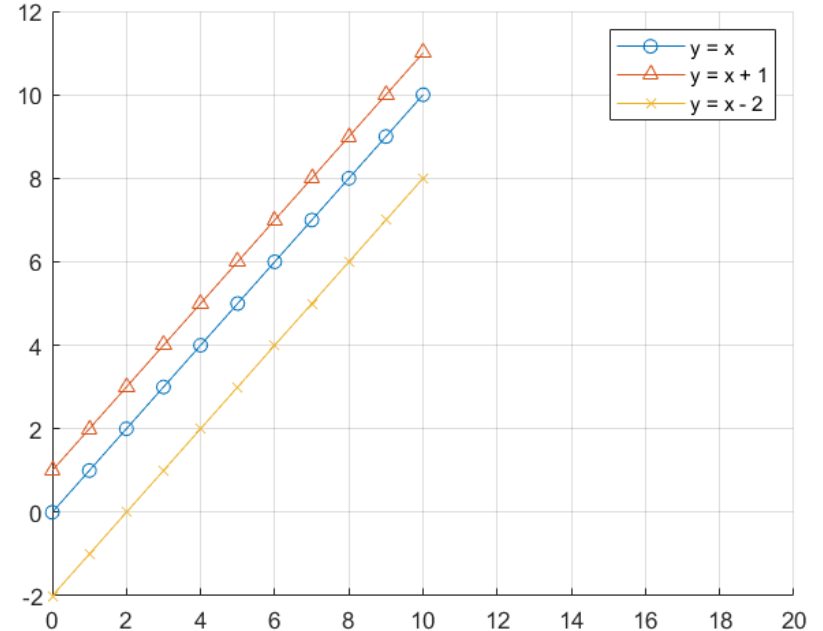
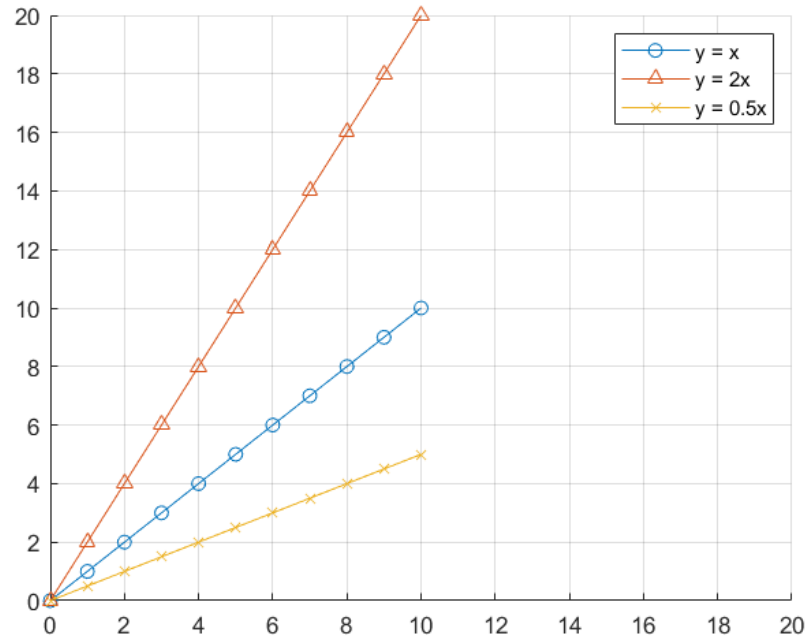
$w_0$  is the **bias coefficient** / intercept (the value of  $y$  when all input variables are zero);

$w_1, w_2, \dots$  are the parameters / weights / coefficients of the input values that need to be estimated from the training data; and

$x_1, x_2, \dots$  are values of the input variables.

# Linear Regression Model

The reason we need  $w_0$  and  $w_1$ :



# Linear Regression Model – Example

Source: <http://onlinestatbook.com/2/regression/intro.html>

Given the following data:

Table 1. Example data.

X	Y
1.00	1.00
2.00	2.00
3.00	1.30
4.00	3.75
5.00	2.25

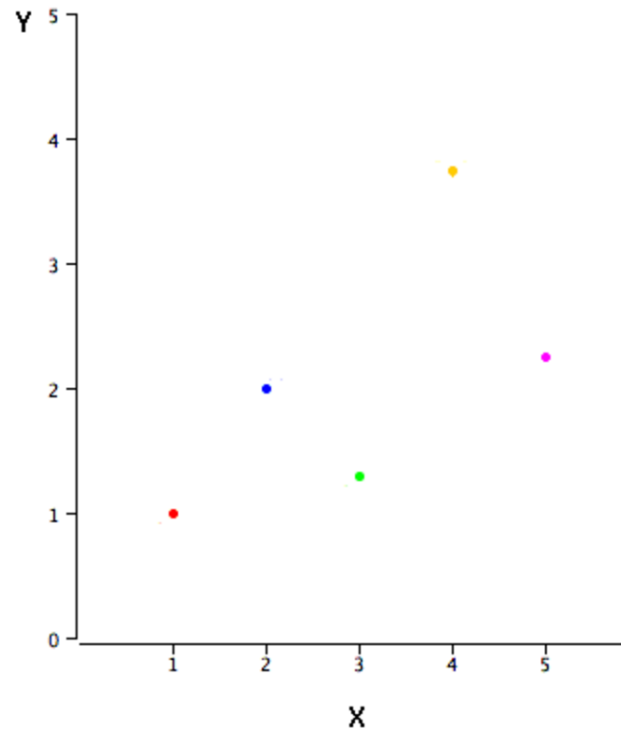


Figure 1. A scatter plot of the example data.

## Task.

Build a simple Linear Regression model that predicts the value of Y when the value of X is known.



# Linear Regression Model – Example

Source: <http://onlinestatbook.com/2/regression/intro.html>

Building a simple Linear Regression is similar to finding the best-fitting straight line (called **regression line**) through the existing data points.

In the diagram on the right, the straight black line is the resulting regression line. Points along this line represents the predicted values of Y given a value of X.

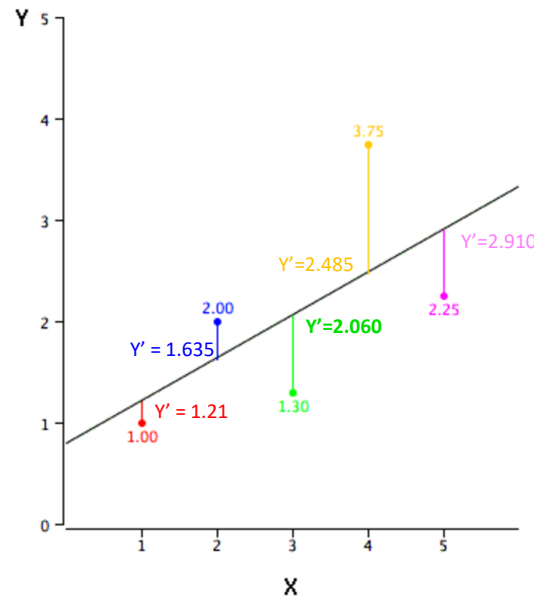


Figure 2. A scatter plot of the example data. The black line consists of the predictions, the points are the actual data, and the vertical lines between the points and the black line represent errors of prediction.

Observe.

The regression line represents our best estimation of the actual values of Y (the coloured data points) and does not need to cross exactly over all of the actual points on the scatterplot. Otherwise, we might end up with an overfitting problem.

Note that the line passes quite closely to the **red data point**; in contrast, it is situated quite far from the **yellow data point**.

# Linear Regression Model – Example

Source: <http://onlinestatbook.com/2/regression/intro.html>

Since there are many possibilities for drawing a regression line through the coloured data points, there must be a way to decide on the **best-fitting regression line**.

Linear Regression solves this by finding the regression line that minimizes the prediction error (hence, an optimization problem). A common measure of such error is the **sum of the squared errors (SSE)**.

Table 2. Example data.

X	Y	Y'	Y-Y'	(Y-Y') <sup>2</sup>
1.00	1.00	1.210	-0.210	0.044
2.00	2.00	1.635	0.365	0.133
3.00	1.30	2.060	-0.760	0.578
4.00	3.75	2.485	1.265	1.600
5.00	2.25	2.910	-0.660	0.436

**SSE = 2.791**

SSE indicates how much of the variation in the dependent variable (Y) is **not explained** by the model.

r indicates how well the model **fits** the data.

Observe.

Table 2 shows the predicted Y values (Y') based on the previous regression line, given each value of X.

Y-Y' is the **absolute error value**.

(Y-Y')<sup>2</sup> is the **squared error value**. Adding up these values for the five data points gives the **sum of the squared errors**.

# Linear Regression Model – Example

Source: <http://onlinestatbook.com/2/regression/intro.html>

The previous regression line is modelled using the following equation:

$$y = 0.785 + 0.425 x$$

For example:

$$\text{for } x = 1, \quad y = 0.785 + 0.425 (1) = 1.21$$

$$\text{for } x = 2, \quad y = 0.785 + 0.425 (2) = 1.64$$


# Linear Regression Model – Example

Source: <http://onlinestatbook.com/2/regression/intro.html>

**How did we calculate the previous Linear Regression equation in the first place?**

Five statistics are required:

- mean of X:  $\mu_x$
- mean of Y:  $\mu_y$
- standard deviation of X:  $s_x$
- standard deviation of Y:  $s_y$
- Pearson's correlation coefficient:  $r_{xy}$


$$\mu_x = \frac{\sum_{i=1}^N x_i}{N}, \text{ where}$$

$x_i$  : an input value

N : the total number of values in a given input variable  $x$



# Linear Regression Model – Example

Source: <http://onlinestatbook.com/2/regression/intro.html>

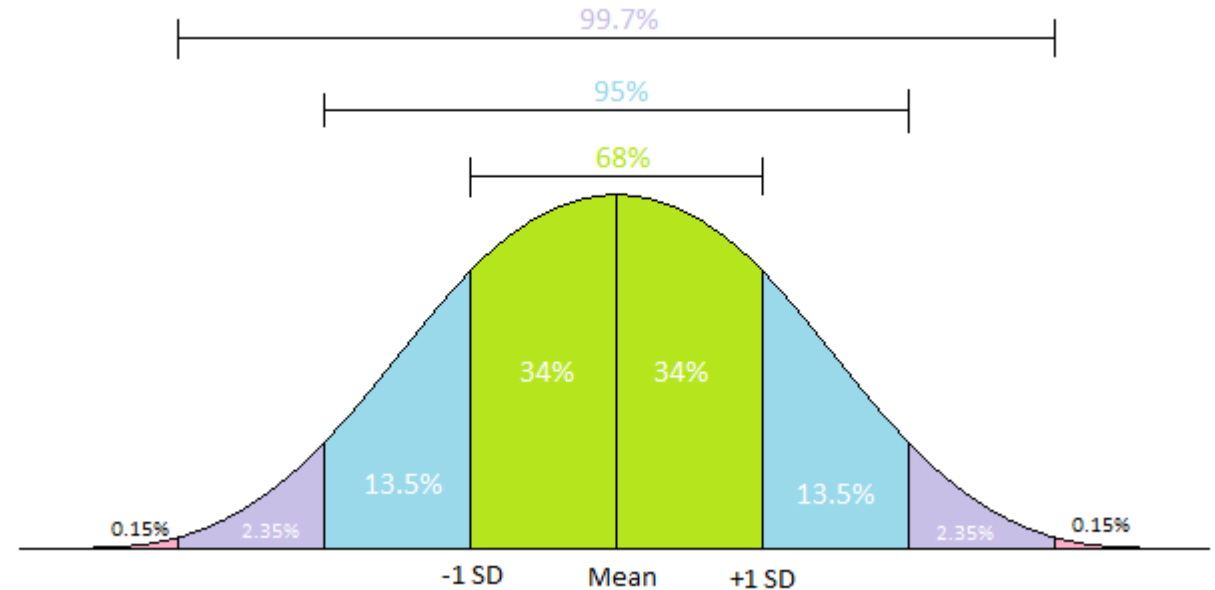
## Standard deviation.

$$s_x = \sqrt{\frac{\sum_{i=1}^N (x_i - \mu_x)^2}{N - 1}}$$

$$s_y = \sqrt{\frac{\sum_{i=1}^N (y_i - \mu_y)^2}{N - 1}}$$

this part of the equation is called the 'sample variance'.

**Standard deviation** measures how far a set of random numbers are spread out from their average value (mean).



Source: <https://www.biologyforlife.com/standard-deviation.html>

# Linear Regression Model – Example

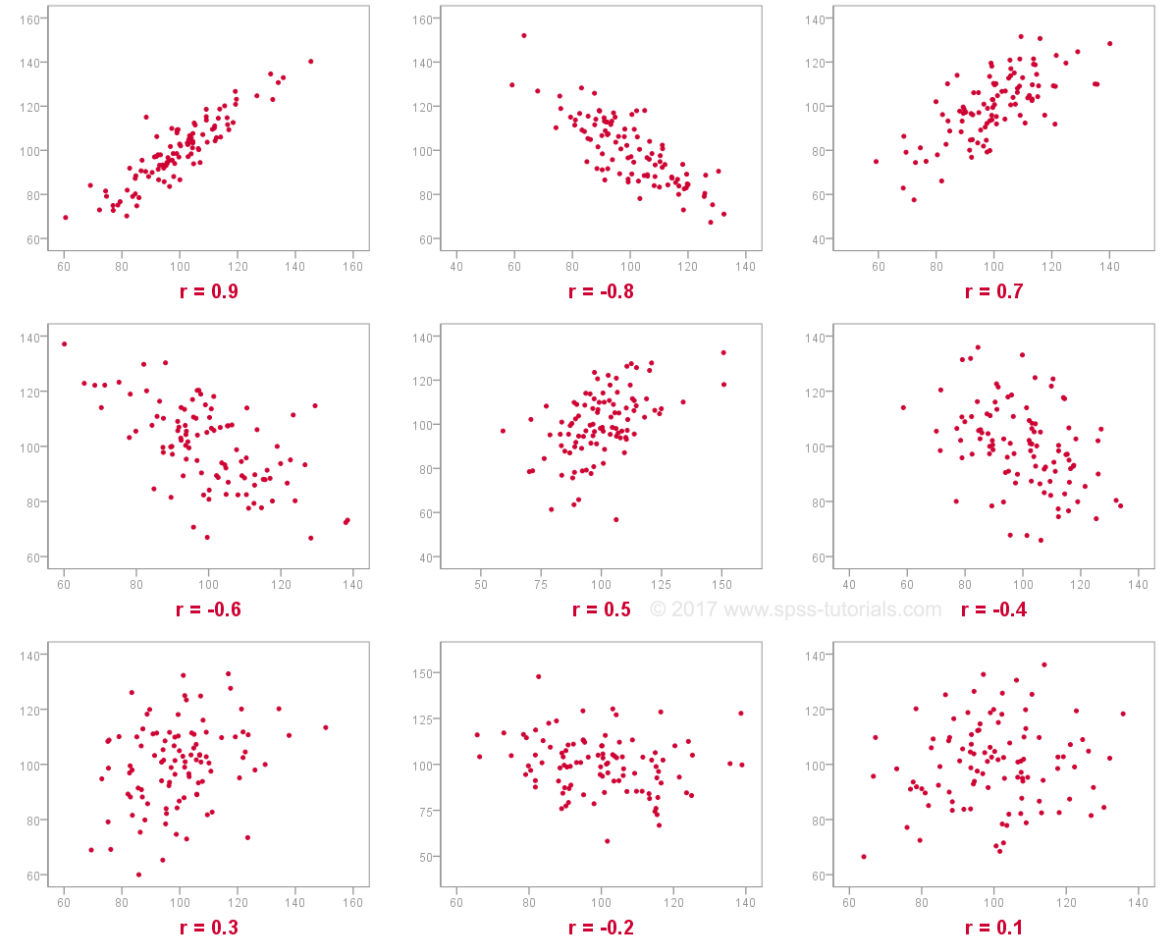
Source: <http://onlinestatbook.com/2/regression/intro.html>

**Pearson's correlation coefficient.**

$$r_{xy} = \frac{\sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y)}{\sqrt{\sum_{i=1}^N (x_i - \mu_x)^2 \cdot \sum_{i=1}^N (y_i - \mu_y)^2}}, \text{ where}$$

N : the total number of data points

**Person's correlation coefficient** measures the strength of association between two variables.



Source: <https://www.spss-tutorials.com/pearson-correlation-coefficient/>

# Linear Regression Model – Example

Source: <http://onlinestatbook.com/2/regression/intro.html>

The resulting statistics:

$$\mu_x = 3.00$$

$$\mu_y = 2.06$$

$$s_x = 1.581$$

$$s_y = 1.072$$

$$r_{xy} = 0.627$$

Linear Regression formula.

$$y = 0.785 + 0.425 x$$

$$w_x = r_{xy} \cdot \frac{s_y}{s_x} = 0.425$$

$$w_0 = \mu_y - w_x \mu_x = 2.06 - (0.425)(3)$$

# Ordinary Least Squares Regression

The previous example illustrates a **simple linear regression** where we only have a single input variable.

When there are more than one input variables, the **ordinary least squares regression** is used to estimate the parameter value (i.e. the coefficients / weights) of each input variable. Similar to finding the best-fitting regression line, the goal here is to finetune these parameters such that they **minimize the sum of the squared error** of each data point.

As this is an optimisation problem, in practice you hardly need to do this manually. Most data science software packages include linear regression functionality to solve this optimization task easily.

# Preparing Data for Linear Regression

Source: <http://machinelearningmastery.com/linear-regression-for-machine-learning/>

- **Linear assumption.** Linear Regression assumes that the relationships between the input and output variables are linear. For non-linear data, e.g., exponential relationship, data transformation technique such as the log transform is needed.
- **Remove noise and outliers.** Linear Regression assumes that the data is clean. Apply appropriate data cleaning techniques to remove possible noise and outliers.
  - Examples of Noisy Data: Incorrect attribute values due to faulty data collection instruments, data entry problems, inconsistency in naming convention, etc.
- **Remove collinearity.** Collinearity is caused by having too many variables trying to do the same job. The **Occam's Razor** principle states that among several possible explanations for an event, the simplest explanation is the best. Consequently, the simpler our model is, the better. Consider calculating pairwise correlations for your input data and remove the most correlated ones.

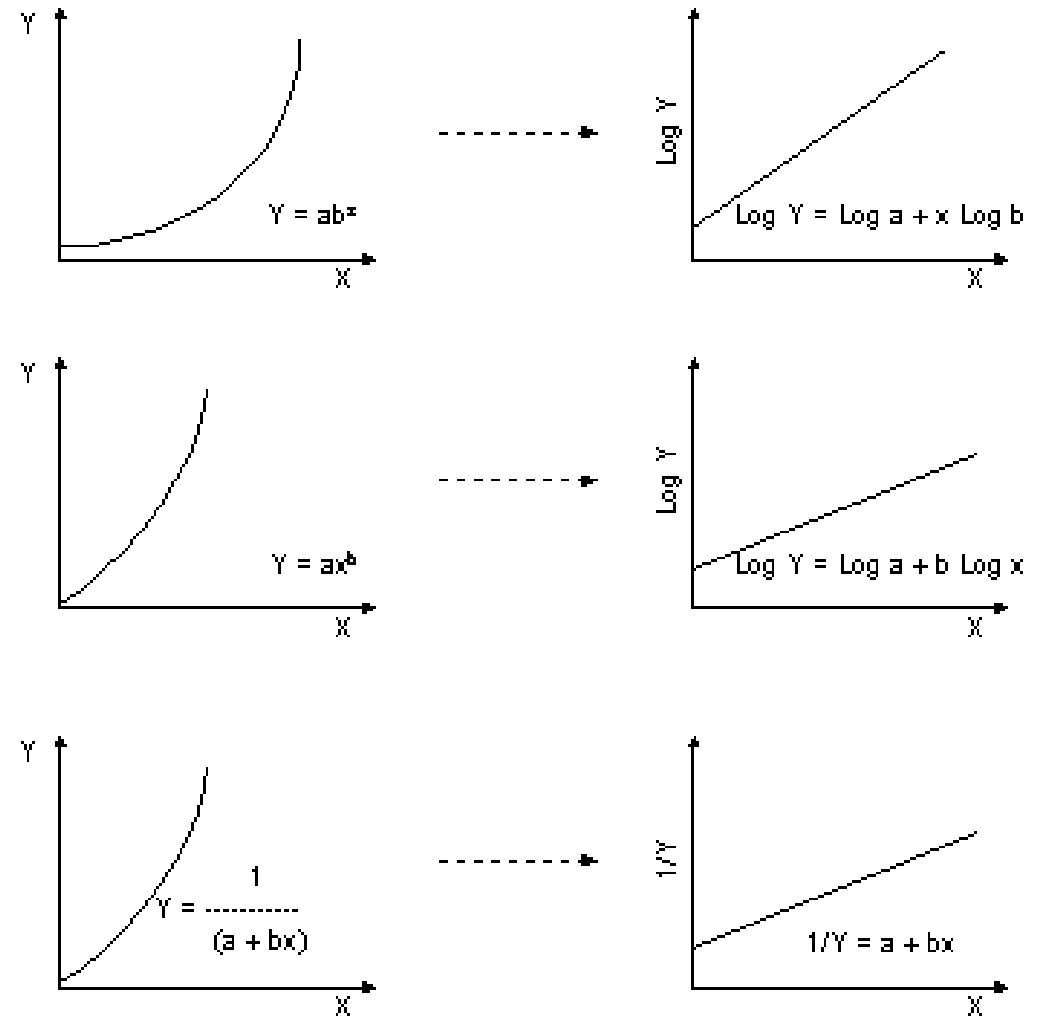
## Non-Linear Transformation:

Linear curve has straight line relationship.

Using non-linear transformation, non-linear problem can be solved as a linear (straight-line) problem.

Source:

<https://people.revoledu.com/kardi/tutorial/Regression/nonlinear/NonLinearTransformation.htm>





## Outliers:

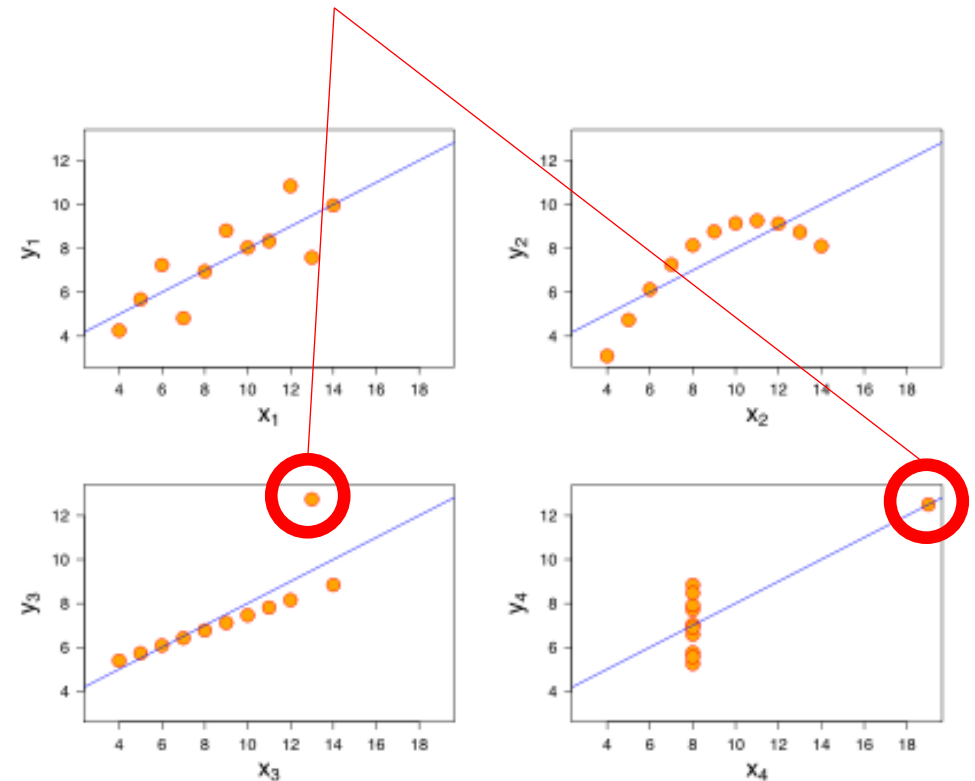
A data point that differs significantly from other data points.

## Anscombe's Quartet

Four (4) datasets with nearly identical descriptive statistics (*mean, variance*) but strikingly different shapes when graphed.

Property	Value	Accuracy
Mean of $x$	9	exact
Sample variance of $x$	11	exact
Mean of $y$	7.50	to 2 decimal places
Sample variance of $y$	4.125	plus/minus 0.003
Correlation between $x$ and $y$	0.816	to 3 decimal places
Linear regression line	$y = 3.00 + 0.500x$	to 2 and 3 decimal places, respectively

Source: [https://en.wikipedia.org/wiki/Anscombe's\\_quartet](https://en.wikipedia.org/wiki/Anscombe's_quartet)



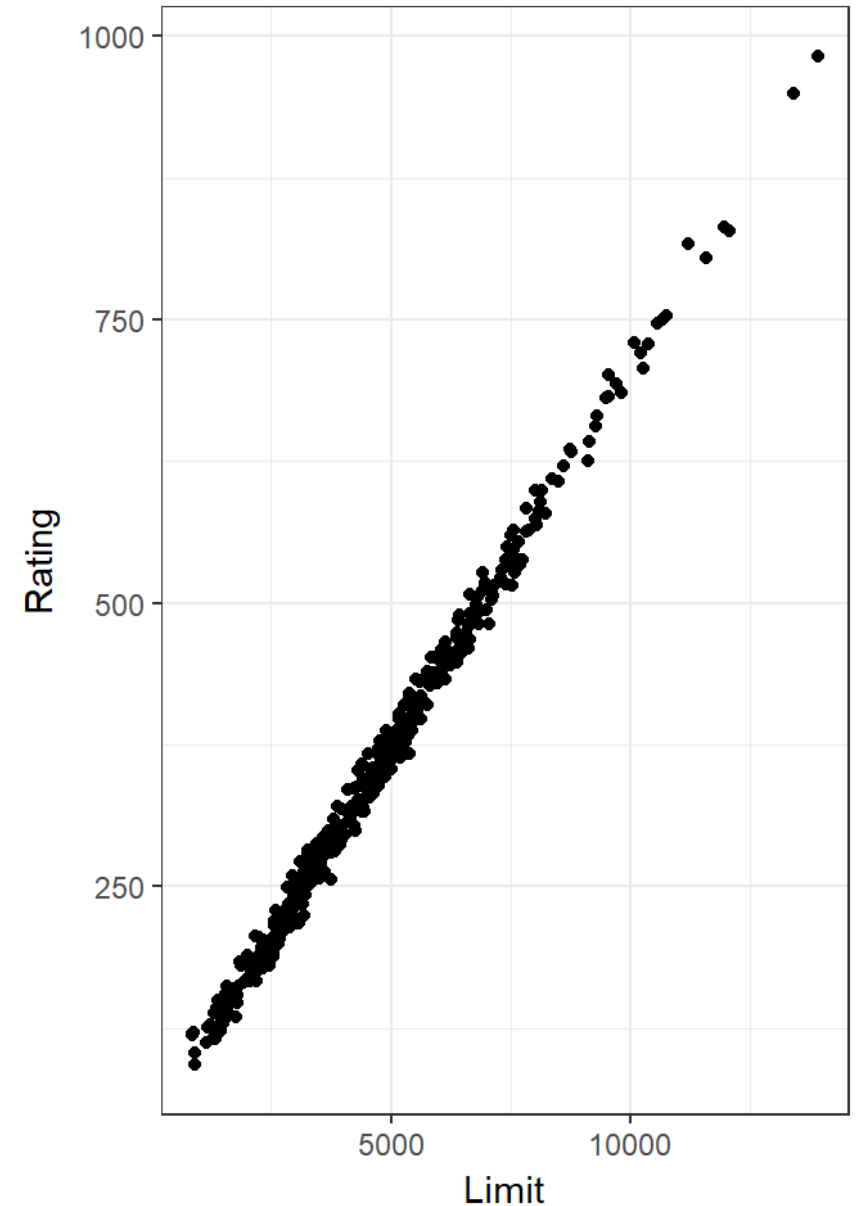
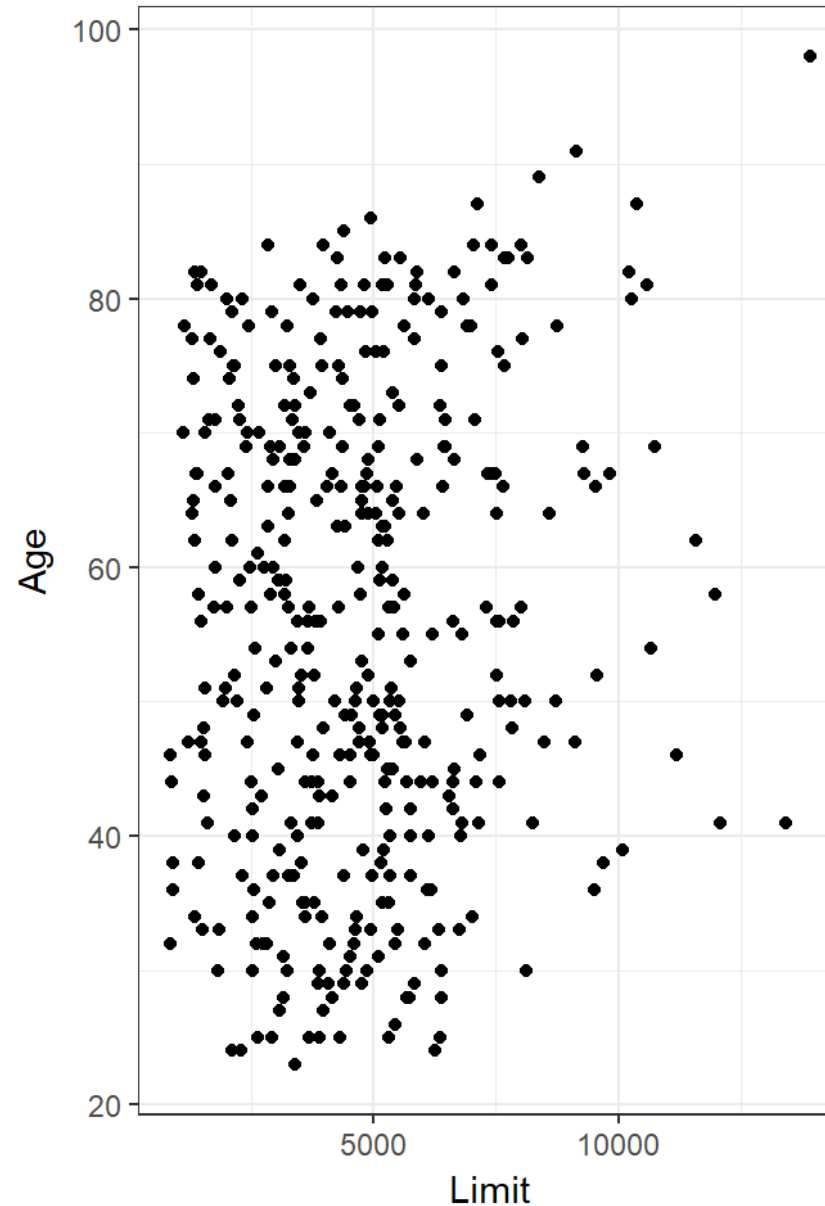
### Collinearity:

Two or more predictors are closely related.

- Problematic in regression because it is difficult to check how much each predictor influences the output separately.

Source:

<https://yetanotheriteration.netlify.com/2018/01/high-collinearity-effect-in-regressions/>



# Preparing Data for Linear Regression

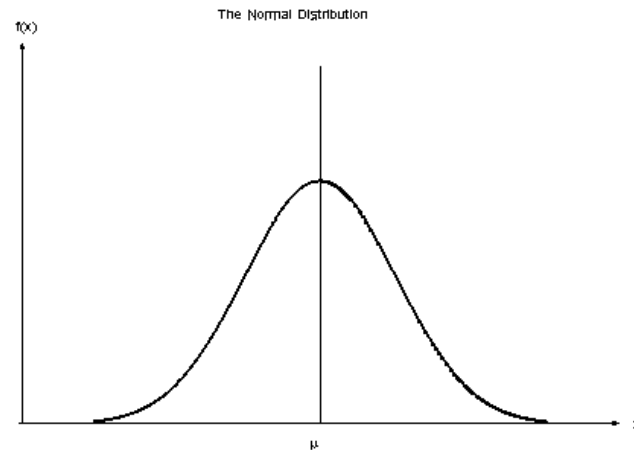
Source: <http://machinelearningmastery.com/linear-regression-for-machine-learning/>

- **Gaussian (normal) distributions.** Linear Regression will produce more reliable predictions if your input and output variables have a Gaussian distribution. Certain data transformation techniques can be used to create a distribution that is more Gaussian looking.

## Gaussian Distribution

Source:

<http://www.itl.nist.gov/diiv898/handbook/pmc/section5/gifs/normal.gif>



- **Rescale input.** Linear regression will often make more reliable predictions if you rescale input variables using standardization or normalization.

## Normalisation:

To change the observations so that they can be described as a normal distribution (also known as the bell curve).

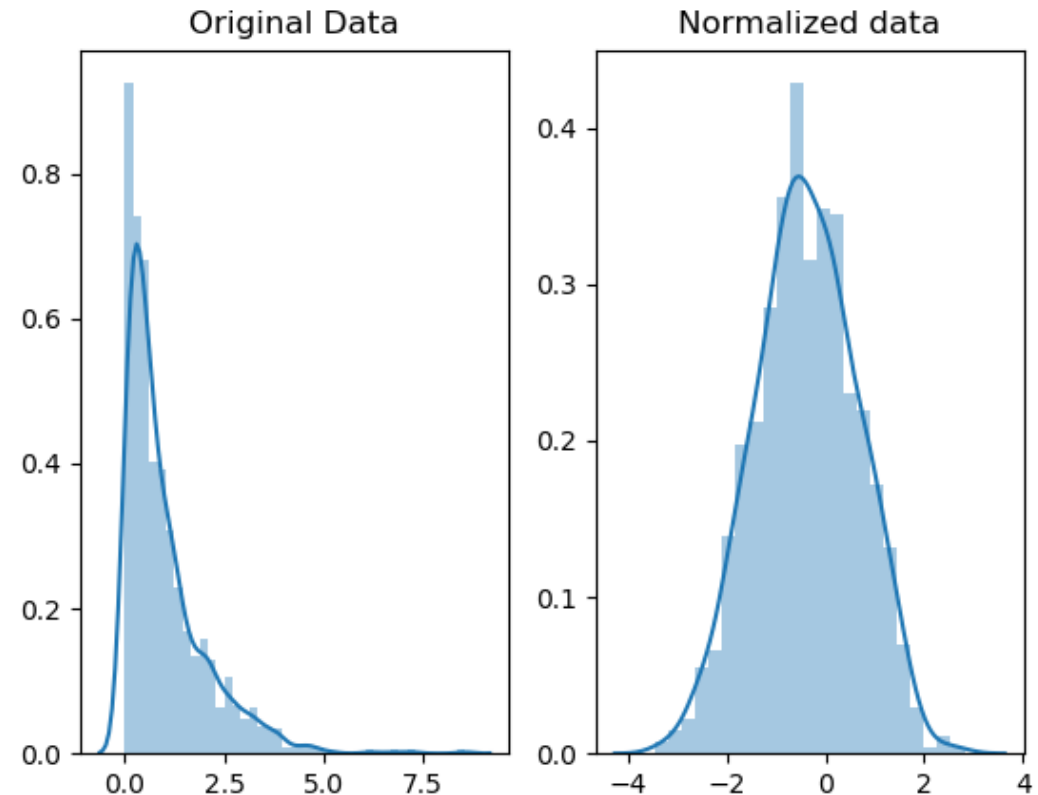
- A Specific statistical distribution where a roughly equal observations falls above and below the mean, the mean and the median are the same, and there are more observations closer to the mean.

$$x' = \frac{x - x_{mean}}{x_{max} - x_{min}}$$

## Standardisation:

Also called z-score normalisation, which transforms data so that the resulting distribution has a mean of 0 and a standard deviation of 1.

$$x' = \frac{x - x_{mean}}{\sigma \text{ (Standard deviation)}}$$



Source: <https://kharshit.github.io/blog/2018/03/23/scaling-vs-normalization>





# WEEK 02 - REGRESSIONS

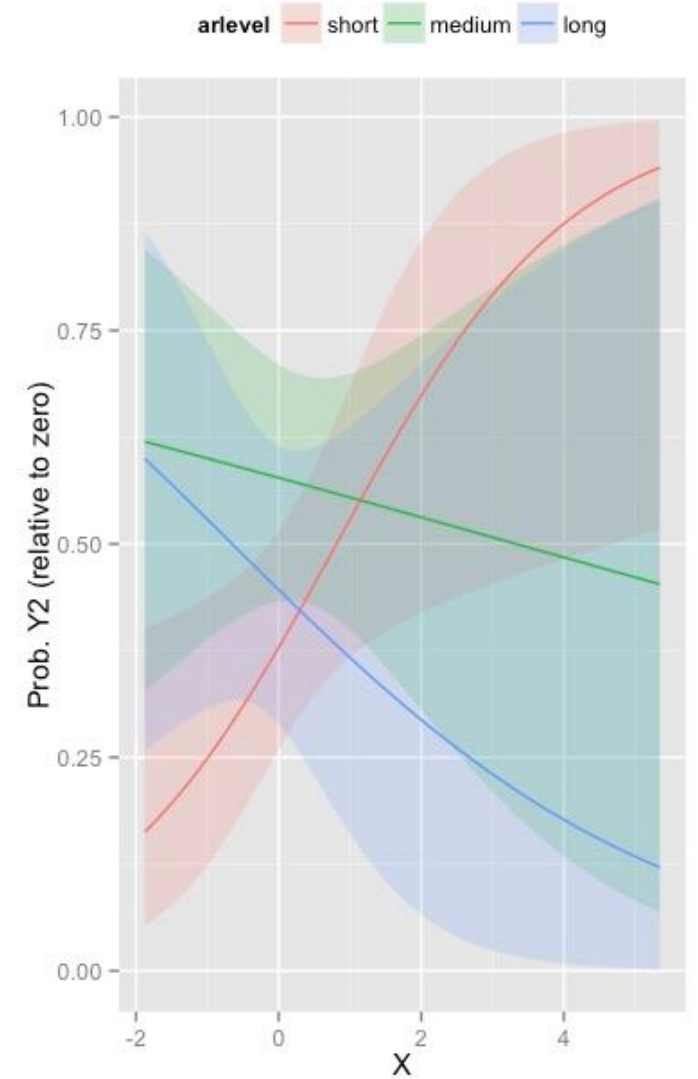
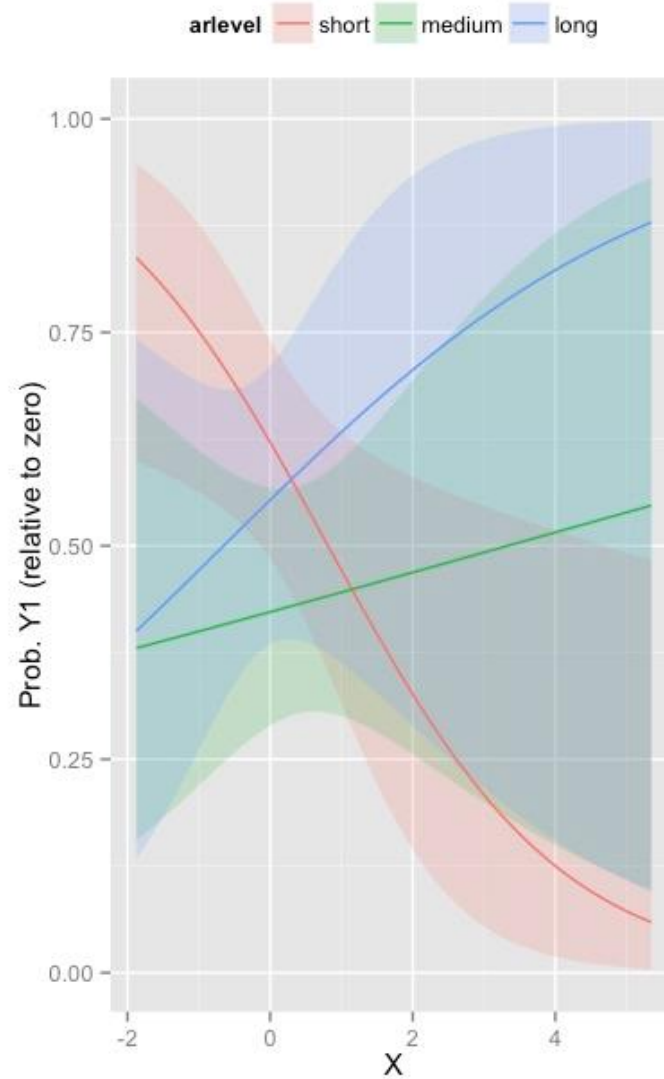
## Logistic Regression

SWIN  
BUR  
NE

SWINBURNE  
UNIVERSITY OF  
TECHNOLOGY

# Logistic Regression

- Logistic regression is a special case of regression analysis and is calculated when the dependent variable is nominally or ordinally scaled.

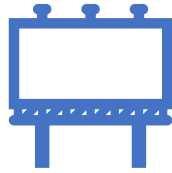




# Examples



Disease  
diagnosis.



Product of  
interest.



Credit  
evaluation.



Election  
predictions.

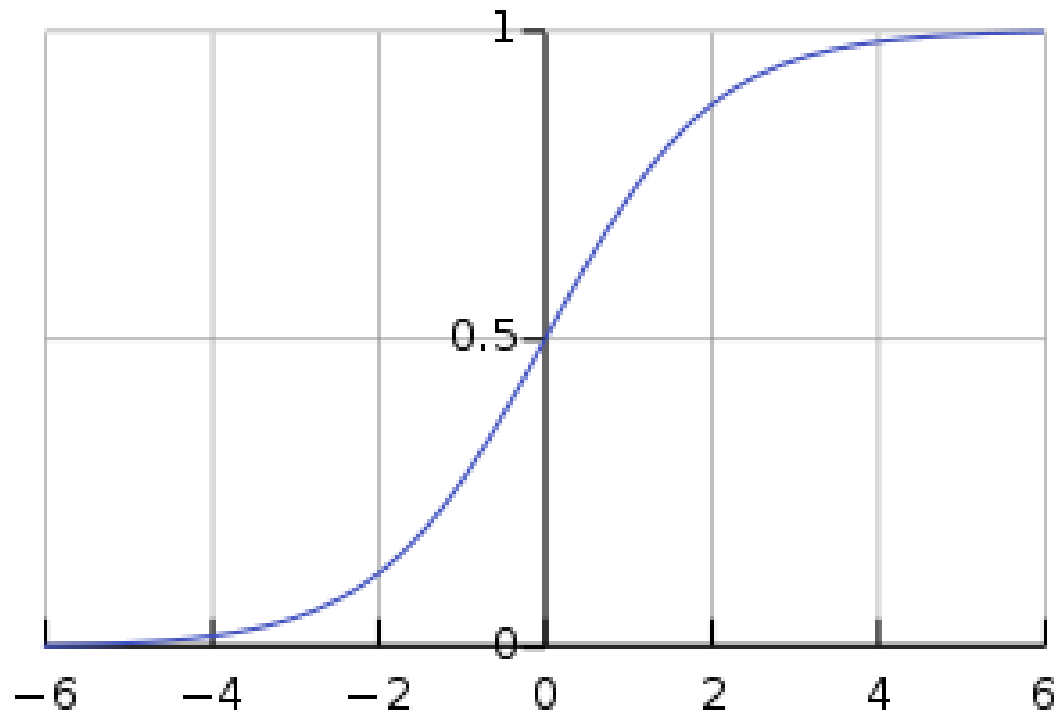
Common point of these examples: the output is a categorical data.



# Revisit Linear Regression

- The regression model uses a linear regression formula to describe the data.
- Ex:

$$y = a_1x_1 + a_2x_2 + \cdots + b$$



# Sigmoid Function

---

- A sigmoid function only produces output values between 0 and 1.
- $S(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{e^x+1} = 1 - S(-x)$

# Logistic Regression Model

- Let's replace the “x” in the sigmoid function with our linear regression function, then the logistic regression model is obtained.
- A logistic function or logistic curve is a common S-shaped curve (sigmoid curve) with equation

$$f(z) = \frac{1}{1 + e^{-z}}$$

where  $z$  stands for the linear regression function.

- Thus, a logistic regression function for the model given in the revisit is:

$$f(X) = \frac{1}{1 + e^{-(a_1x_1 + a_2x_2 + \dots + b)}}$$

# Texts and Resources

Unless stated otherwise, the materials presented in this lecture are taken from:

- Dietrich, D. ed., 2015. *Data Science & Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data*. EMC Education Services.

