# Welcome to General Assembly

- **WiFi** GA Guest
- **Password** yellowpencil

GENERAL ASSEMBLY

# DATA SCIENCE

## SYD DAT 10

## Week 2 - Linear Regression

# AGENDA



**Key:** Interactive Paul Presenting

1. Course Plan revisited
2. Review – Git & GitHub setup
3. Recap last time & Homework Presentations
4. Visualisation Lab continued (Class 3)
5. Supervised Vs Unsupervised learning
6. What is Linear Regression?
7. How do Run a Linear Regression Model?
8. Multiple Linear Regression
9. Non-Linear Effects
10. Regression Lab
11. Discussion / Review / Homework

# Course Plan

## UNITS

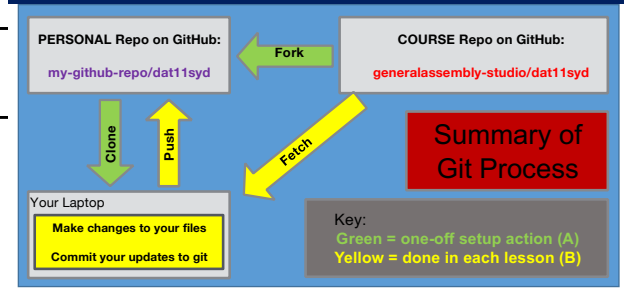
| | | |
|---|---|---|
| **UNIT 1: FOUNDATIONS OF DATA MODELING** | Introduction to Data Science | Lesson 1 |
| | Elements of Data Science | Lesson 2 |
| | Data Visualisation | Lesson 3 |
| | Linear Regression | Lesson 4 |
| | Logistic Regression | Lesson 5 |
| | Model Evaluation | Lesson 6 |
| | Regularisation | Lesson 7 |
| | Clustering | Lesson 8 |
| **UNIT 2: DATA SCIENCE IN THE REAL WORLD** | Recommendations | Lesson 9 |
| | SQL + Productivity | Lesson 10 |
| | Decision Trees | Lesson 11 |
| | Ensembles | Lesson 12 |
| | Natural Language Programming | Lesson 13 |
| | Cloud Computing | Lesson 14 |
| | Time Series | Lesson 15 |
| | Soft Skills | Lesson 16 |
| | Network Analysis | Lesson 17 |
| | Neural Networks | Lesson 18 |
| | Final Projects Presentations | Lesson 19 |
| | Final Projects Presentations | Lesson 20 |

GA

# Git & GitHub – 1 Pager Guide!

- Squash the GIT confusion from last class!
- 1-Pager follows through the steps clearly
- Run through this once together ➔ CRYSTAL CLEAR

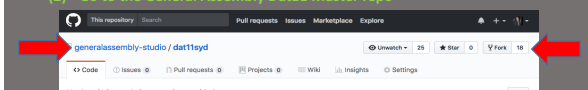
## 1-Pager - Git and GitHub




(Part A) - One-off setup:

(1) Login to GitHub as you

(2) Go to the General Assembly Dat11 Master repo




(3) Click "Fork" to send a copy of the COURSE repo to your own GitHub

(4) Clone your PERSONAL repo to your own workspace (laptop or server):

```
git clone https://github.com/yourgithubname/yourgithubrepo
```

(5) Set your upstream to the COURSE repo:

```
git remote add upstream https://github.com/generalassembly-studio/dat11syd
```

(Part B) EVERY CLASS:

At the START of the class, you'll need to sync the latest materials from the COURSE repo:

(1) Make sure you are in the dat11syd directory:

```
cd ~/workspace/dat11syd
```

(2) Make sure to select the "master" branch of your repo:

```
git checkout master
```

(3) Fetch the latest changes from the UPSTREAM repo (i.e the course repo)

```
git fetch upstream
```

(4) Merge the changes from the upstream repo to your master branch:

```
git merge upstream/master
```

DURING the class:

(5) Before editing, either copy files to your "students/" folder, or rename them

At the END of every class:

(6) Make sure you are in the dat11syd directory:

```
cd ~/workspace/dat11syd
```

(7) Add any files that you've updated to your git registry:

```
git add -A
```

(8) Commit the changes with a sensible comment:

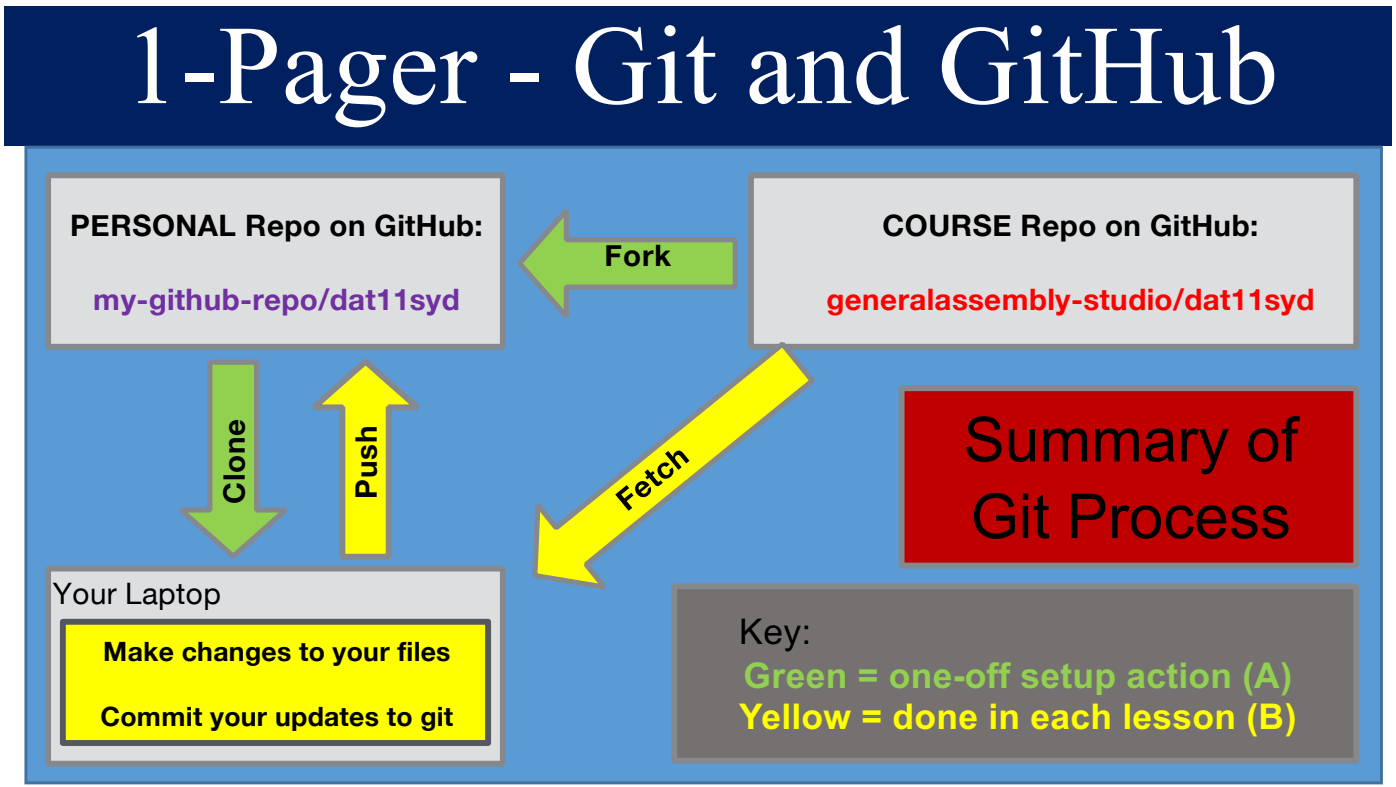
```
git commit -m "my updates for lesson 7"
```

(9) Push your changes to your PERSONAL repo:

```
git push origin master
```

DONE!!!!!

# Git & GitHub – 1 Pager Guide!

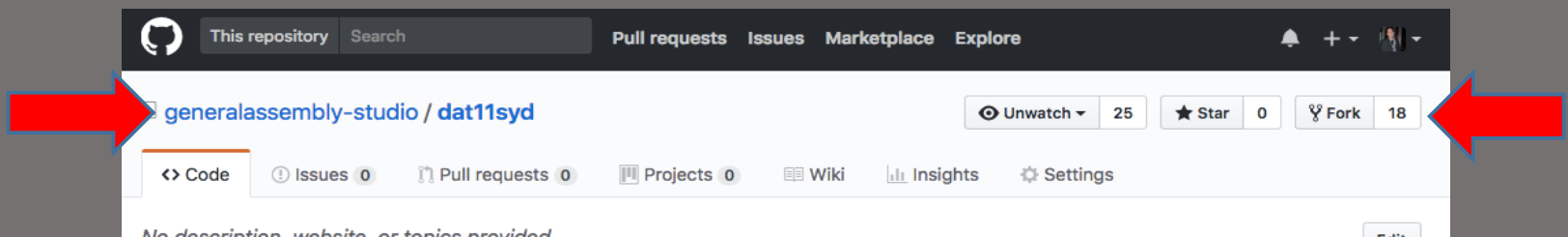


# Git & GitHub – 1 Pager Guide!

**(Part A) - One-off setup:**

(1) Login to GitHub as you

(2) Go to the General Assembly Dat11 Master repo




(3) Click "Fork" to send a copy of the COURSE repo to your own GitHub

(4) Clone your PERSONAL repo to your own workspace (laptop or server):

```
git clone https://github.com/yourgithubname/yourgithubrepo
```

(5) Set your upstream to the COURSE repo:

```
git remote add upstream https://github.com/generalassembly-studio/dat11syd
```

# Git & GitHub – 1 Pager Guide!

**(Part B) EVERY CLASS:**

**At the START of the class, you'll need to sync the latest materials from the COURSE repo:**

(1) **Make sure you are in the dat11syd directory:**

```
cd ~/workspace/dat11syd
```

(2) **Make sure to select the "master" branch of your repo:**

```
git checkout master
```

(3) **Fetch the latest changes from the UPSTREAM repo (i.e the course repo)**

```
git fetch upstream
```

(4) **Merge the changes from the upstream repo to your master branch:**

```
git merge upstream/master
```

**DURING the class:**

(5) **Before editing, either copy files to your "students/" folder, or rename them**

**At the END of every class:**

(6) **Make sure you are in the dat11syd directory:**

```
cd ~/workspace/dat11syd
```

(7) **Add any files that you've updated to your git registry:**

```
git add -A
```

(8) **Commit the changes with a sensible comment:**

```
git commit -m "my updates for lesson 7"
```

(9) **Push your changes to your PERSONAL repo:**

```
git push origin master
```

**DONE!!!!!**

# Recap – last lesson

**Homework Presentations ~ 2mins each**

## Data Science Homework 1

## DAT11 | Lesson 2 | Homework 1

### Investigating Data

*Instructions for the homework - time required: ~2 hours*

- 1. Browse through the list of Data Science problems on Kaggle: https://www.kaggle.com/datasets
- 2. Choose 3 data sets that interest you,
- 3. Download the data and read it into Python Notebook on Jupyter Hub
- 4. Have a look at the datasets you have downloaded, and consider:

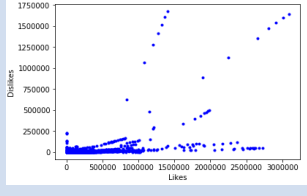
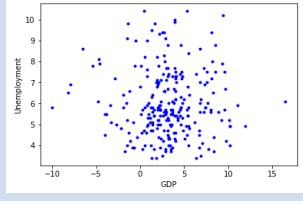
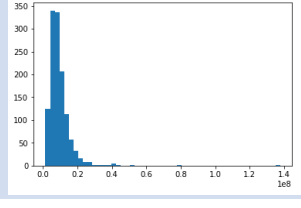
a. What is the data for?,
b. What are the data-types? (Numerics, Characteristics, etc?)
c. What alternative uses could you think of for the dataset?

- 5. Derive some simple, interesting facts from each dataset to report back to the class (plot a trend, or determine a relationship)

### For the next class (Monday 26th Feb), please prepare:

```
* 1 single slide about your investigations - plan to keep your presentation to <= 5mins *
```

# Tostee

| | YouTube Trending Video | Federal Reserve Interest Rate | S&P Stock Data |
|---|---|---|---|
| Data Purpose | This dataset is a daily record of the top trending YouTube videos.<br><br>• Sentiment analysis in a variety of forms<br>• Categorising YouTube videos based on their comments and statistics.<br>• Training ML algorithms like RNNs to generate their own YouTube comments.<br>• Analysing what factors affect how popular a YouTube video will be.<br>• Statistical analysis over time. | This dataset includes data on the economic conditions in the United States on a monthly basis since 1954. The federal funds rate is the interest rate at which depository institutions trade federal funds (balances held at Federal Reserve Banks) with each other overnight. The rate that the borrowing institution pays to the lending institution is determined between the two banks; the weighted average rate for all of these types of negotiations is called the effective federal funds rate.<br><br>How does economic growth, unemployment, and inflation impact the Federal Reserve's interest rates decisions? How has the interest rate policy changed over time? Can you predict the Federal Reserve's next decision? Will the target range set in March 2017 be increased, decreased, or remain the same? | The data is presented in a couple of formats to suit different individual's needs or computational limitations. I have included files containing 5 years of stock data<br><br>This dataset lends itself to a some very interesting visualizations. One can look at simple things like how prices change over time, graph an compare multiple stocks at once, or generate and graph new metrics from the data provided. From these data informative stock stats such as volatility and moving averages can be easily calculated. The million dollar question is: can you develop a model that can beat the market and allow you to make statistically informed trades! |
| Data Types | Data includes the video title, channel title, publish time, tags, views, likes and dislikes, description, and comment count.<br><br>Data is Strings, Numeric, DateTime | Federal Funds Rate, Real GDP, Unemployment Rate, Inflation Rate<br><br>Numeric, String | All the files have the following columns: Date - in format: yy-mm-dd<br>Open - price of the stock at market open (this is NYSE data so all in USD). High, Low, Close, Volume |
| Alternative Uses | Measurin | Identifying how changes in interest rates will impact GDP over the subsequent periods | Predict volumes based on historical patterns. Identify seasonality flows. |
| Interesting Fact | Max Likes 3,093,544<br>Max Dislikes 1,674,420<br>Dislikes Avg 3,042, Median 332<br>Likes Avg 9011, Median 332 | Max unemployment 10.8%<br>Highest Growth 16.5% | AAL Median Volume 8,111,323<br>AAL Max Volume 137,767,165<br><br>AAL American Airlines |
| Plot / Relationship |  |  |  |

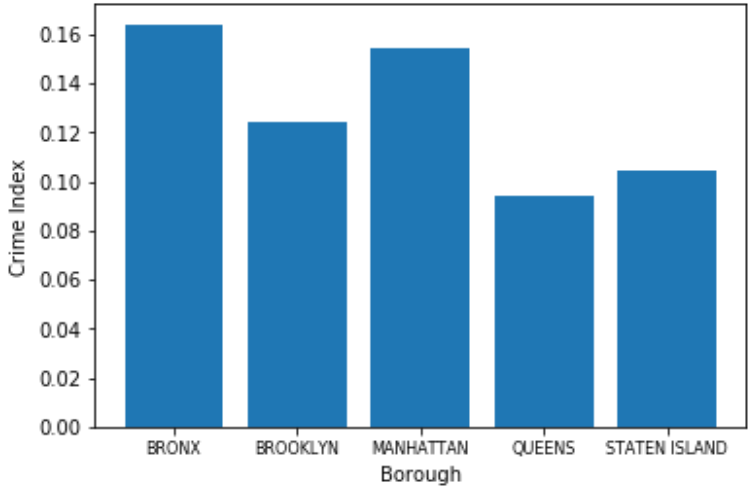
# Andrew

## Chess Game Dataset (Lichess)

Black Win Rate = 47.8%
White Win Rate = 52.2%

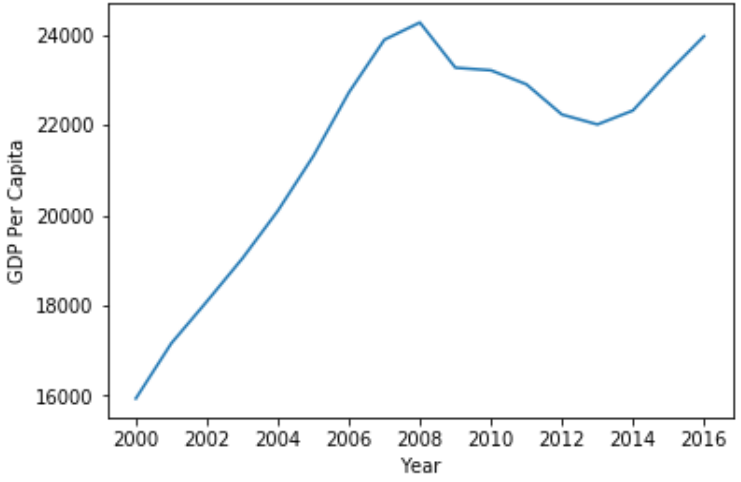
## New York City Crimes

Crime Index created from population and a count of crime occurrences.




## Nominal € GDP per capita of Spain (by regions)

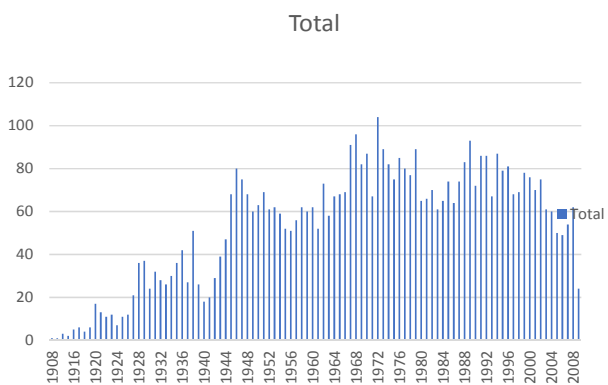
Aggregated to National Average



# Suren

## Airplance crashes since 1908

- Data: Air plane crashes since 1908
- 5270 rows up to year 2009
- Data issues
  - ❑ Location - Inconsistent
  - ❑ Flight No/Route - Blanks
- Data Types
  - ❑ Date
  - ❑ Time
  - ❑ Str – Operator, Summary
  - ❑ Int – Fatalities
- Interesting fact : Fatalities remain constant since 1960's




Alternate dataset – Fatalities by car

## Wine reviews by variety, region and taster

- Data: winemag-data_first150k
- 130 K rows of data
- Data Issues - Blanks
  - Price –
  - Location – country, province, region
- Data Types
  - ❑ Index
  - ❑ Int – Points
  - ❑ Currency - Cost
  - ❑ Str – locations, variety, taster
- Interesting fact: Wines with the most points are the most expensive wines

| Varierty | Max of price | Max of points |
|---|---|---|
| Portuguese Red | $450.00 | 100.00 |
| Sangiovese | $800.00 | 100.00 |
| Prugnolo Gentile | $237.00 | 100.00 |
| Muscat | $350.00 | 100.00 |
| Sangiovese Grosso | $900.00 | 100.00 |
| Syrah | $750.00 | 100.00 |
| Port | $1,000.00 | 100.00 |
| Chardonnay | $2,013.00 | 100.00 |
| Merlot | $625.00 | 100.00 |
| Cabernet Sauvignon | $625.00 | 100.00 |
| Bordeaux-style Red Blend | $3,300.00 | 100.00 |
| Champagne Blend | $600.00 | 100.00 |
| Bordeaux-style White Blend | $1,000.00 | 100.00 |

Alternate dataset – Craft Beer and Spirits

## Melbourne Housing Market

- Data: Melbourne_housing_FULL
- 31 K rows of data for year 2017
- Data Issues - Blanks
  - Price – undisclosed
  - Property –
- Data Types
  - ❑ Type, Method
  - ❑ float – distance
  - ❑ Currency - Cost
  - ❑ Str – Other
- Interesting fact: 1/3 of the total suburbs have an average property price above the average price of $1M for all property in Melb.

| Suburb | Average of Price | Max of Price2 |
|---|---|---|
| Canterbury | $2,379,609.76 | $8,000,000.00 |
| Deepdene | $2,191,666.67 | $3,680,000.00 |
| Middle Park | $2,130,400.00 | $6,400,000.00 |
| Malvern | $1,995,710.28 | $6,600,000.00 |
| Brighton | $1,976,054.40 | $11,200,000.00 |
| Albert Park | $1,950,105.56 | $4,735,000.00 |
| Balwyn | $1,867,150.72 | $5,650,000.00 |
| Camberwell | $1,850,031.05 | $5,450,000.00 |
| Ivanhoe East | $1,802,769.23 | $3,850,000.00 |
| Kew | $1,782,822.76 | $6,500,000.00 |

Alternate dataset – Sydney

**Enough talking…**

**Let's do some lab work:**

**dat11syd/lessons/lesson-03/lab/lesson-03-lab-02_Visualisation**

DATA SCIENCE PART TIME COURSE

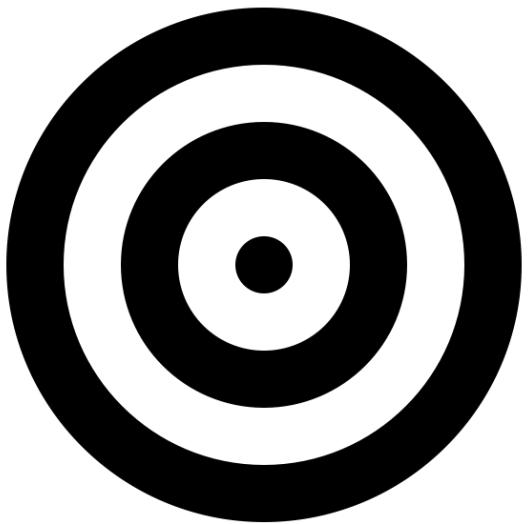
# SUPERVISED & UNSUPERVISED LEARNING

Learning by example.

## What do we want to model?

Often we want to predict something.

- Who will win a match
- What a customer wants
- The value of a stock
- The time until an event

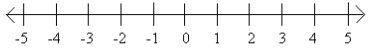


The target may also be called:

- response variable
- dependant variable
- label

**Regression:**

If the target variable is numeric then we have a regression problem - we are trying to predict a continuous number




Classification:

If the target variable is a category (for example trying to predict a type of flower) the we have a classification problem - we are trying to classify what group that y belongs to.

The data values that provide information to help guess the target

The features may also be called:

- predictor variables
- independent variable

The data values that provide information to help guess the target

Today's Air Pressure

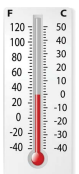
Today's Temperature

Today's Humidity

Today's Cloud Cover

~>

◎ Tomorrow's Temperature

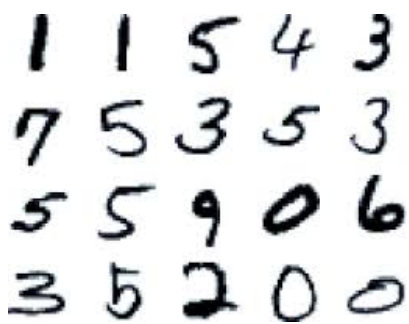


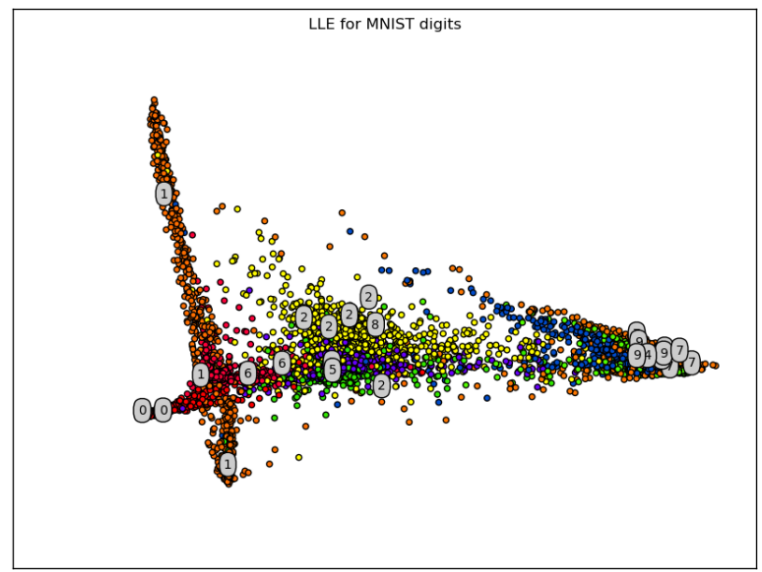

The features may also be called:

- predictor variables
- independent variable

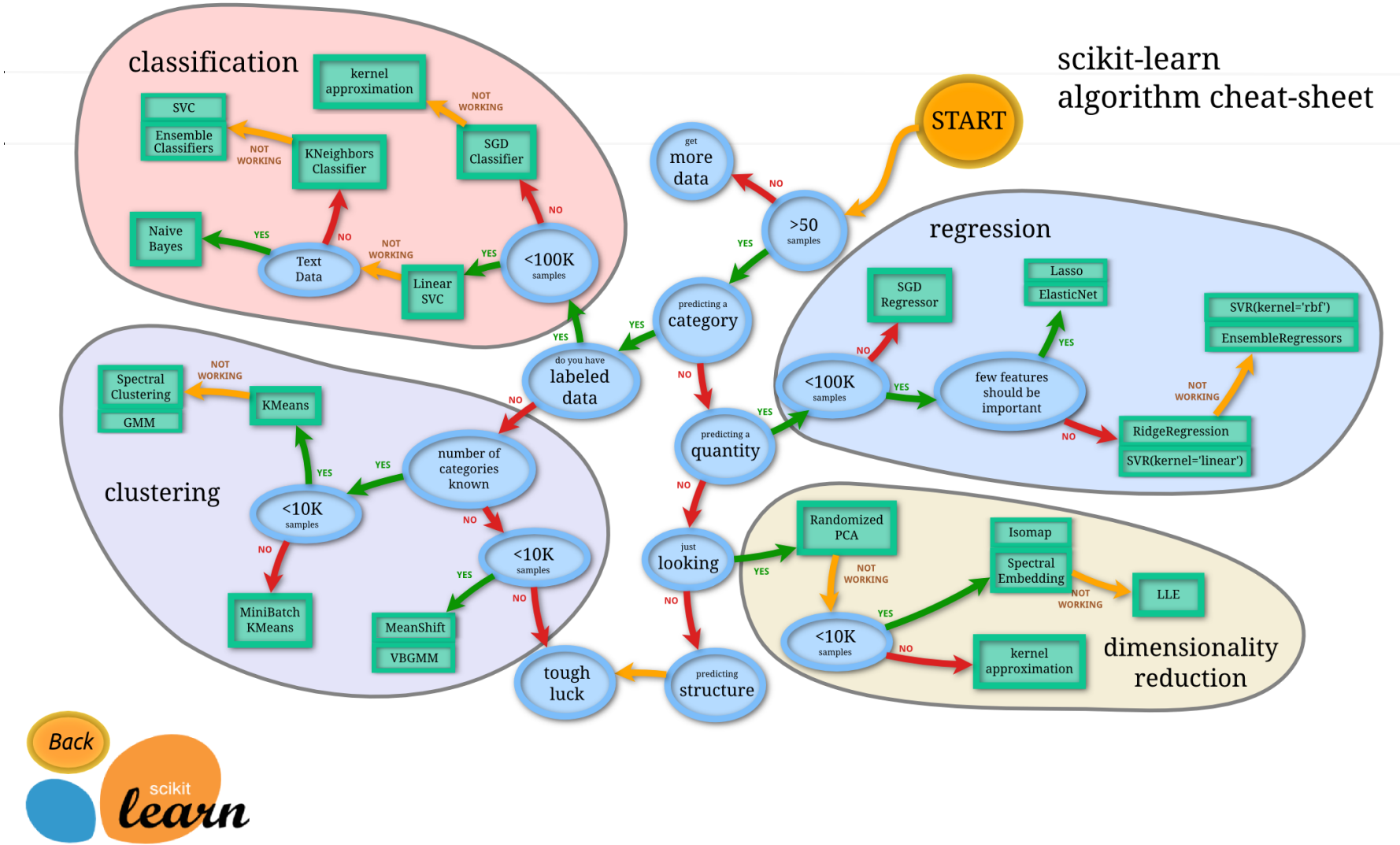
Find structure in data, such as clusters.

We want to find some underlying structure or patterns in the data but in this case we don't have any labeled data.






http://projector.tensorflow.org/


scikit-learn
algorithm cheat-sheet
START
get more data
NO
>50 samples
YES
predicting a category
YES
do you have labeled data
YES
<100K samples
YES
Linear SVC
NOT WORKING
Text Data
NO
Naive Bayes
YES
KNeighbors Classifier
NOT WORKING
Ensemble Classifiers
SVC
NOT WORKING
kernel approximation
NOT WORKING
SGD Classifier
NO
classification
NO
predicting a quantity
YES
<100K samples
NO
SGD Regressor
YES
few features should be important
YES
Lasso
ElasticNet
NO
RidgeRegression
SVR(kernel='linear')
NOT WORKING
SVR(kernel='rbf')
EnsembleRegressors
regression
NO
number of categories known
YES
<10K samples
YES
KMeans
NOT WORKING
Spectral Clustering
GMM
NO
MiniBatch KMeans
NO
<10K samples
YES
MeanShift
VBGMM
NO
clustering
NO
just looking
YES
Randomized PCA
NOT WORKING
<10K samples
YES
Isomap
Spectral Embedding
NOT WORKING
LLE
NO
kernel approximation
dimensionality reduction
NO
predicting structure
tough luck
Back
scikit learn

DATA SCIENCE PART TIME COURSE

# WHAT IS LINEAR REGRESSION?

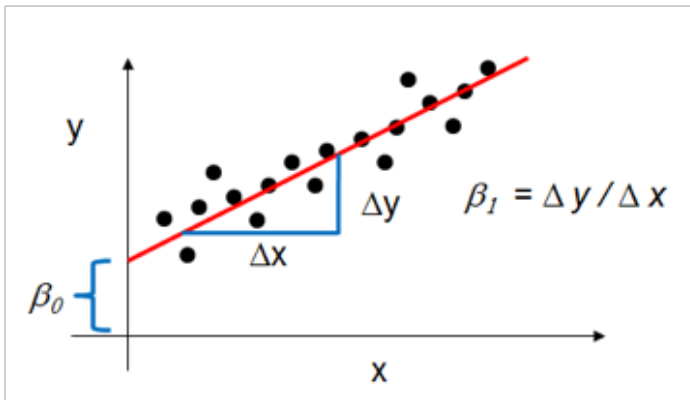
We want to model a linear relationship (think straight line) between our target variable y and our input variable x.

$$y=mx+b$$

- Def: Explanation of a continuous variable given a series of independent variables

- The simplest version is just a line of best fit: $y = mx + b$

- Explain the relationship between **x** and **y** using the starting point **b** and the power in explanation **m**.



$$y=X\beta+\epsilon$$

- y = target variable
- X = input variable
- β = coefficients
- ϵ = error term

Note, one of our input variables can be 1 so we have an intercept parameter

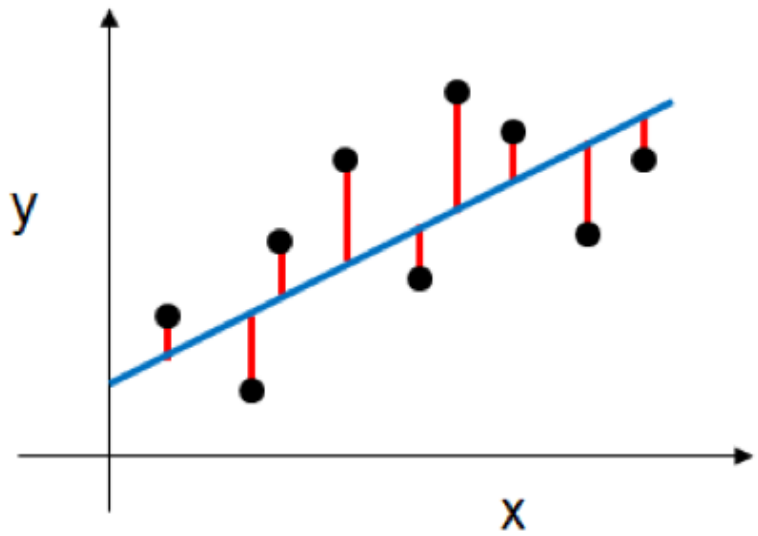
# LINEAR REGRESSION - EXAMPLE



We want to predict the price of a house, based on some observed data we have about the area, number of bedrooms, size of the house, and if it has a pool or not.

The goal is a function $y = f(X)$, to describe the house price based on observed data.

- y : house price $
- X's : the area (x1), number of bedrooms (x2), size of the house (x3), and if it has a pool or not (x4)



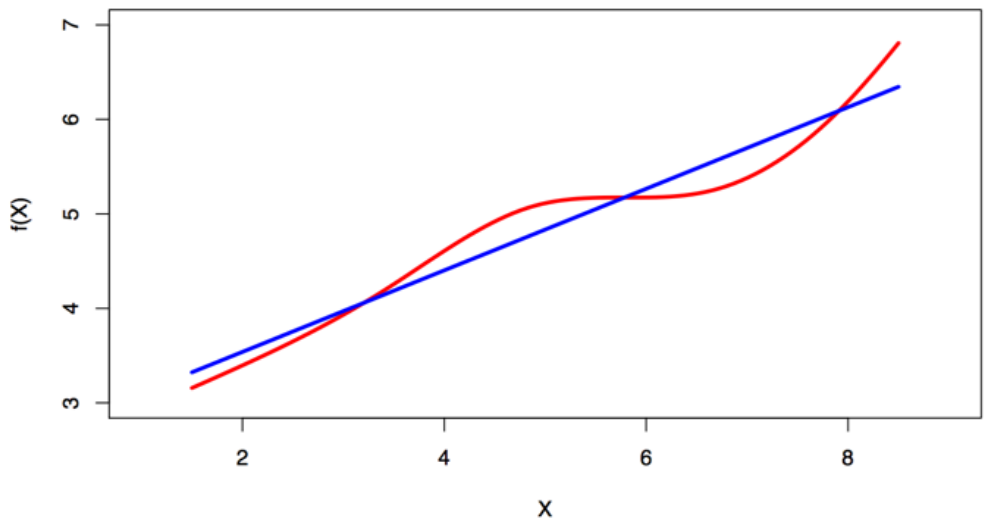

Model Prediction

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

Observed Result

Linear regression is a simple approach to supervised learning. It assumes that the dependence of Y on X1, X2 ,… Xp is linear.

True regression functions are never linear



- Linear relationship in the parameters, β, we can transform the actual values of the inputs if we want
- Variance of the error term, ϵ, is constant. This means there is no systematic pattern in the values of X and the variance of ϵ
- The mean of ϵ = 0
- ϵ has a normal distribution. If it does not, it could introduce *bias.*
- No perfect (or near perfect) co-linearity between any of the input variables. Otherwise the fitting procedure will break.

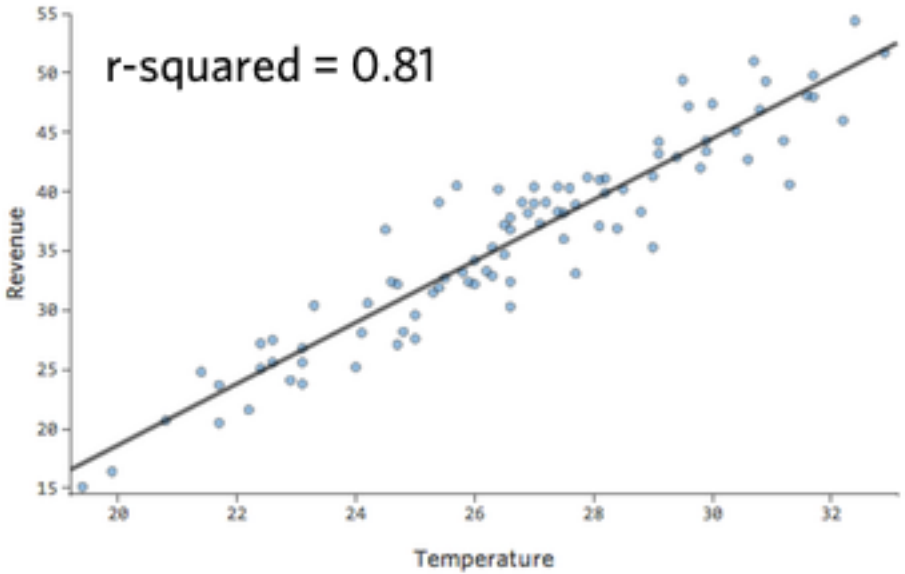
- R-squared, the central metric introduced for linear regression
- Which model performed better, one with an r-squared of 0.79 or 0.81?
- R-squared measures explain variance.
- But does it tell the magnitude or scale of error?
- We'll explore loss functions and find ways to refine our model.

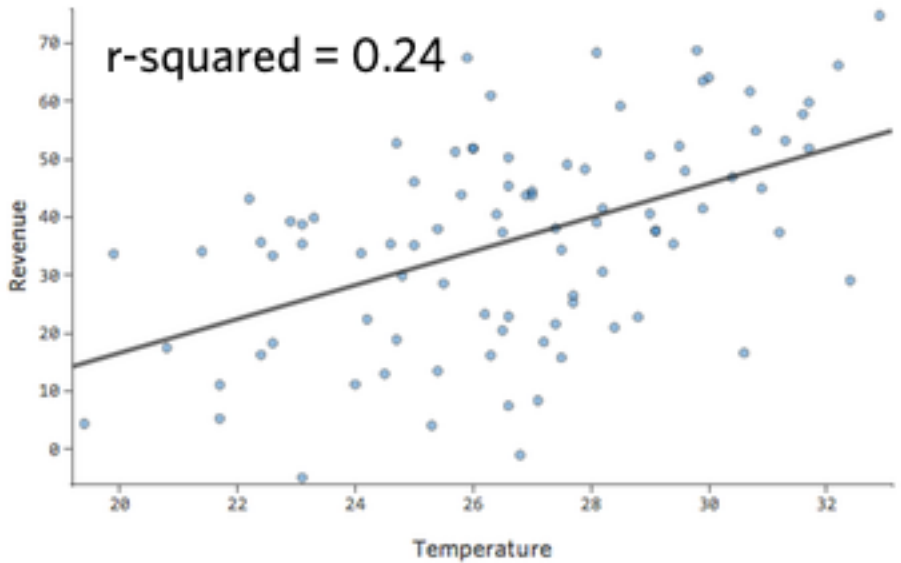
DATA SCIENCE PART TIME COURSE

# EVALUATING QUALITY OF FIT?

# LINEAR REGRESSION – MEASURING MODEL RESULTS 

**Perfect agreement between predicted and actual values: $R^2 = 1$**
**Awful / No agreement between predicted and actual values: $R^2$ = close to zero**



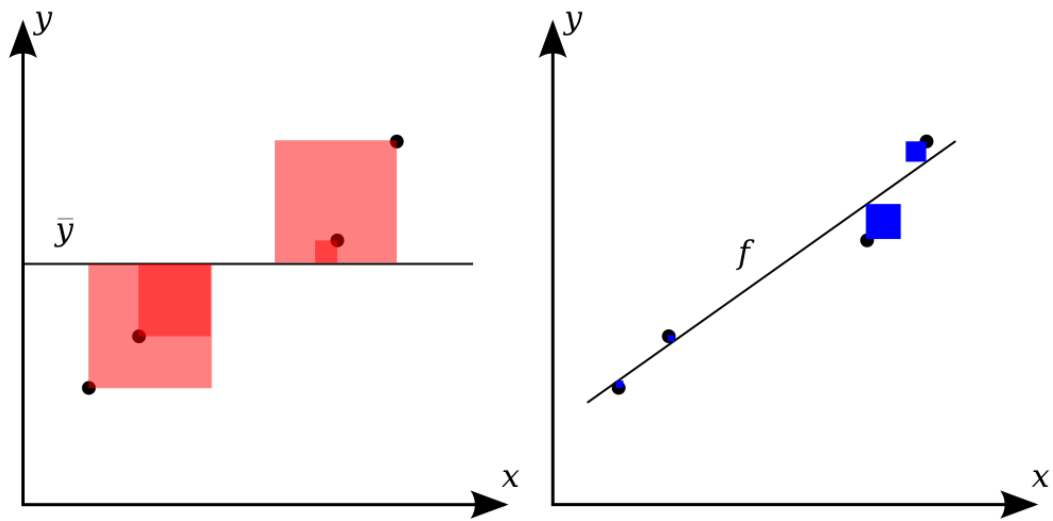

**http://docs.statwing.com/interpreting-residual-plots-to-improve-your-regression/**

R-Squared

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

$$SS_{\text{res}} = \sum_i (y_i - f_i)^2$$

$$SS_{\text{tot}} = \sum_i (y_i - \bar{y})^2$$



DATA SCIENCE PART TIME COURSE

# HOW TO RUN LINEAR REGRESSION?

$$SS_{res} = \sum_{i=1}^{n} (y_i - f(x_i))^2$$

Basically, what we are trying to do is minimise the Residual Sum of Squares. This is the Sum of the squared difference between our observed value and the value from the model

$$SS_{res} = \sum_{i=1}^{n} (y_i - f(x_i))^2$$

Basically, what we are trying to do is minimise the Residual Sum of Squares. This is the Sum of the squared difference between our observed value and the value from the model

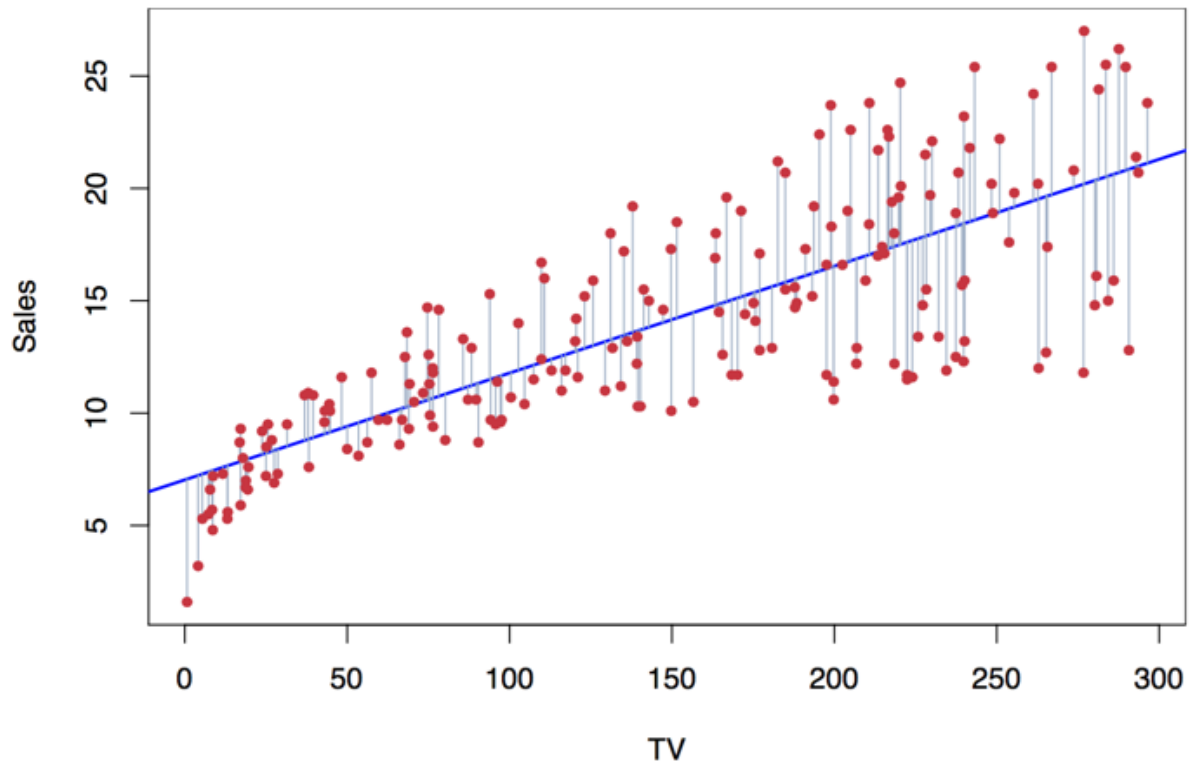
1 $$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$$

2 $$\text{RSS} = e_1^2 + e_2^2 + \cdots + e_n^2$$

3 $$\text{RSS} = (y_1 - \hat{\beta}_0 - \hat{\beta}_1 x_1)^2 + (y_2 - \hat{\beta}_0 - \hat{\beta}_1 x_2)^2 + \ldots + (y_n - \hat{\beta}_0 - \hat{\beta}_1 x_n)^2$$

4 $$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2},$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x},$$

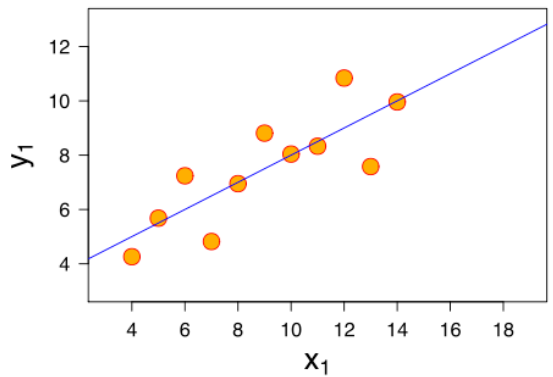

Sales
5
10
15
20
25
0
50
100
150
200
250
300
TV

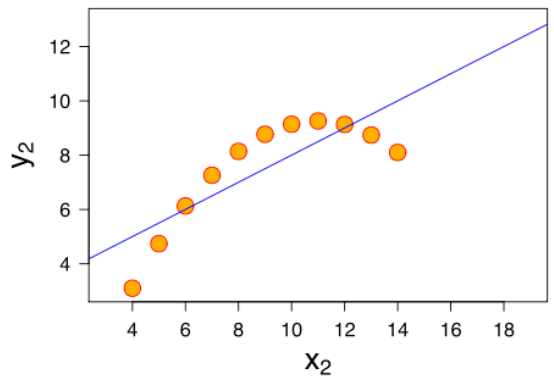
# LINEAR REGRESSION - IMPORTANT STEP

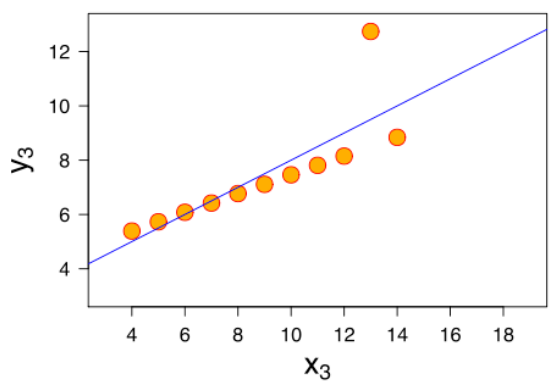


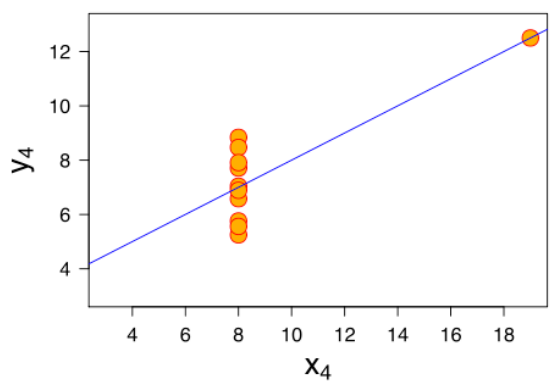
**Recognise whether the relationship between a data feature and target variable is actually something like linear.**

**If it is not, you will want to either transform the variable to make it linear, or you may decide to throw the feature away.**



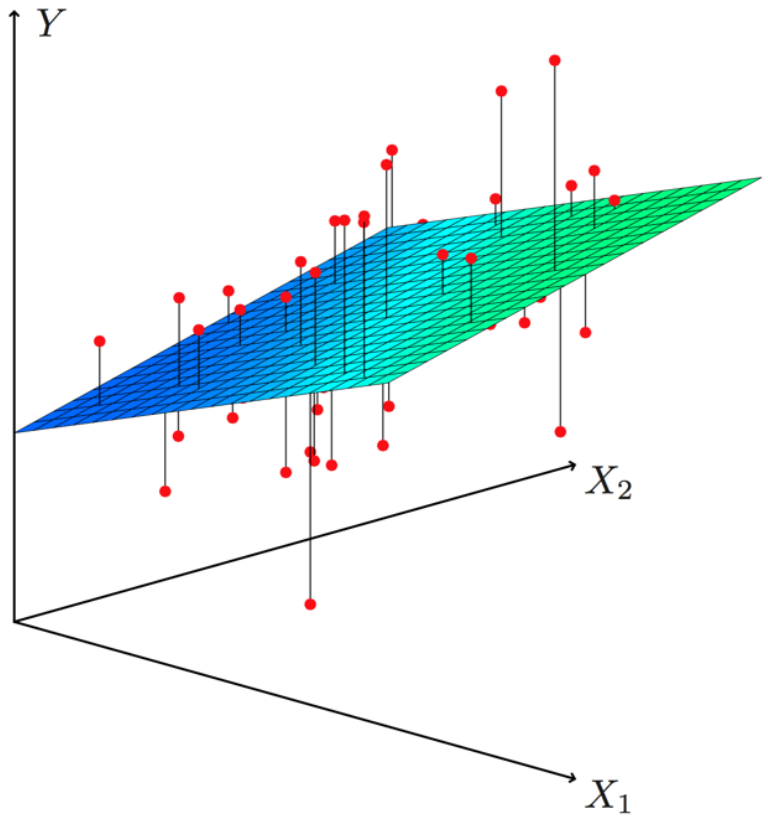



DATA SCIENCE PART TIME COURSE

# MULTIPLE LINEAR REGRESSION

- multi-dimensions
- allows for complex models even with linear components

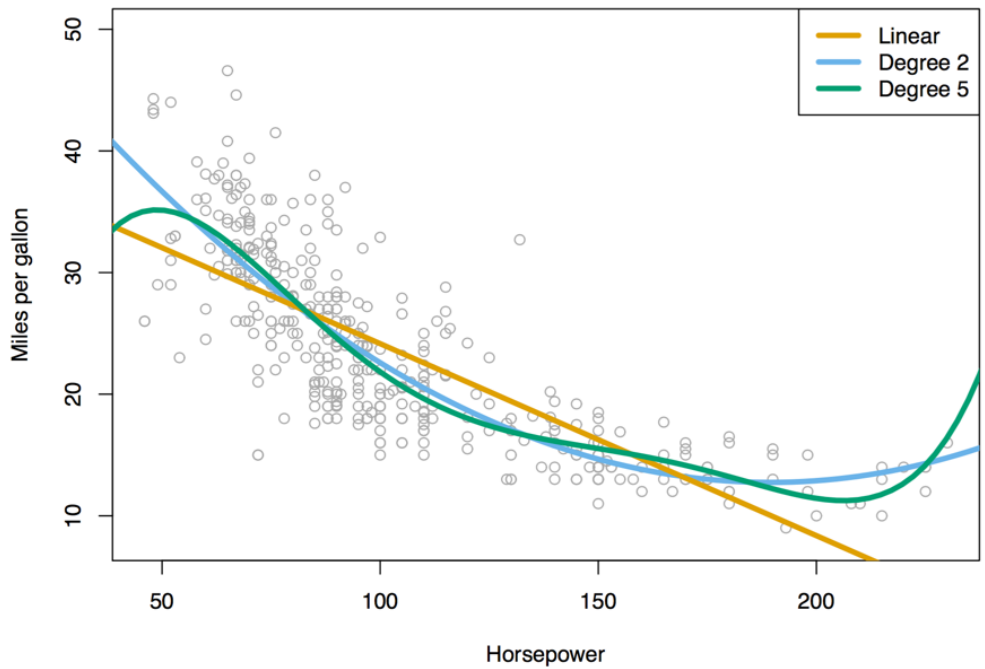
$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$


Y
X2
X1

- The ideal scenario is when the predictors are uncorrelated:
  - Interpretations can be made such as “a unit change in Xj is associated with a βj change in Y , while all the other variables stay fixed”
- Correlations amongst predictors cause problems
  - when Xj changes, everything else changes

DATA SCIENCE PART TIME COURSE

# NON-LINEAR EFFECTS USING LINEAR REGRESSION




$$\texttt{mpg} = \beta_0 + \beta_1 \times \texttt{horsepower} + \beta_2 \times \texttt{horsepower}^2 + \epsilon$$

# LAB

# DATA SCIENCE PART TIME COURSE

**Refer to the Git 1-pager at:**

**dat11syd / docs**

**(Part B) EVERY CLASS:**

**At the START of the class, you'll need to sync the latest materials from the COURSE repo:**

(1) **Make sure you are in the dat11syd directory:**
```
cd ~/workspace/dat11syd
```
(2) **Make sure to select the "master" branch of your repo:**
```
git checkout master
```
(3) **Fetch the latest changes from the UPSTREAM repo (i.e the course repo)**
```
git fetch upstream
```
(4) **Merge the changes from the upstream repo to your master branch:**
```
git merge upstream/master
```

**DURING the class:**

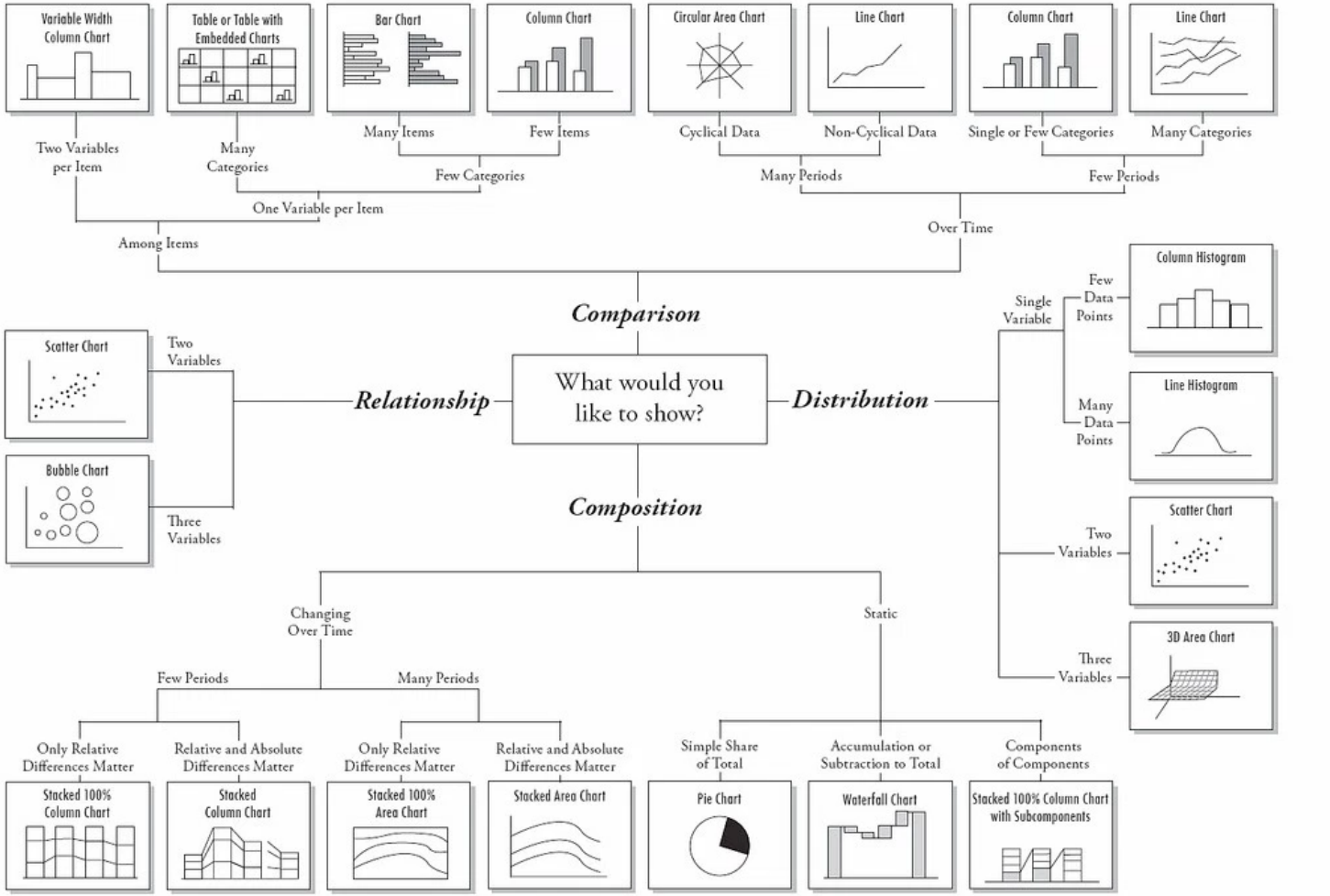
(5) **Before editing, either copy files to your "students/" folder, or rename them**

**At the END of every class:**

(6) **Make sure you are in the dat11syd directory:**
```
cd ~/workspace/dat11syd
```
(7) **Add any files that you've updated to your git registry:**
```
git add -A
```
(8) **Commit the changes with a sensible comment:**
```
git commit -m "my updates for lesson ?"
```
(9) **Push your changes to your PERSONAL repo:**
```
git push origin master
```

**DONE!!!!!**

# Chart Suggestions—A Thought-Starter



# DISCUSSION TIME

**Homework (Due Tue 28th Nov)**

Apply Linear Regression Example to your own project

- Make one plot and
- A linear regression
- Slack your Jupyter notebook to me or push to your repo and let me know on slack

Read Chapter 4 of Introduction to Statistical Learning - Classification

If your data does not lend itself to regression use this dataset

http://archive.ics.uci.edu/ml/datasets/Auto+MPG

To get data click “Data Folder” at top of page