# Review Outline

## W1

### What is data science

The study of data and extract the knowledge of data (extract meaningful insights from data.)

### Drew Conway’s Venn diagram



### Usefulness of machine learning

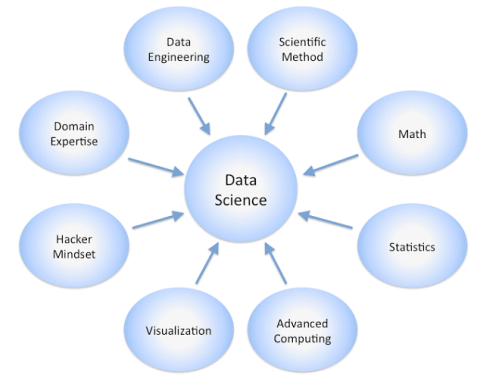
* Human expertise is not available
* Many solutions need to be adapted automatically
* Humans are expensive for the work
* Situation changes overtime
* large amounts of data

### Different components of a data science process

Standard Value Chain:

* Collection: getting the data
* Engineering: storage and computational resources across full lifecycle
* Governance: overall management of data across full lifecycle
* Wrangling: data pre-processing, cleaning
* Analysis: discovery (learning, visualisation, etc.)
* Presentation: arguing the case that the results are significant and useful
* Operationalisation: putting the results to work, so as to gain benefits or value

### Differentiate data science from other related disciplines



## W2

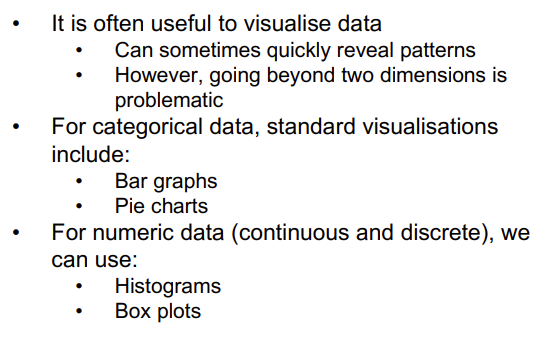
### Essentials for coding in Python for data science

### Interpret given Python codes

### Why we study Python and its importance for data science

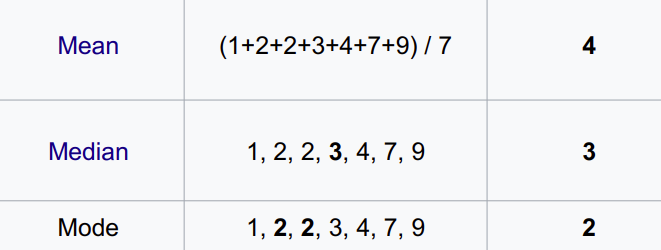
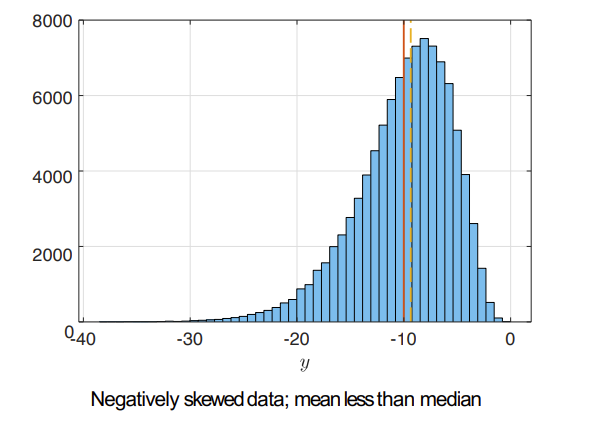
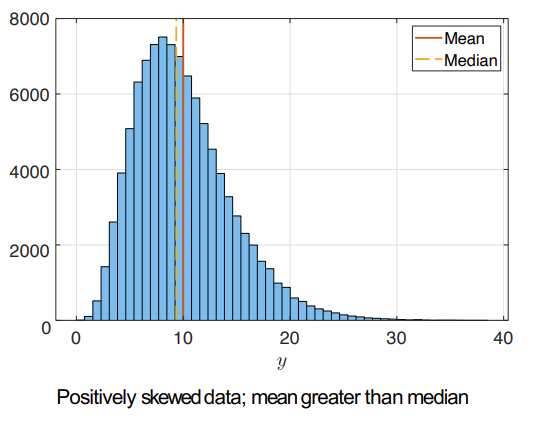
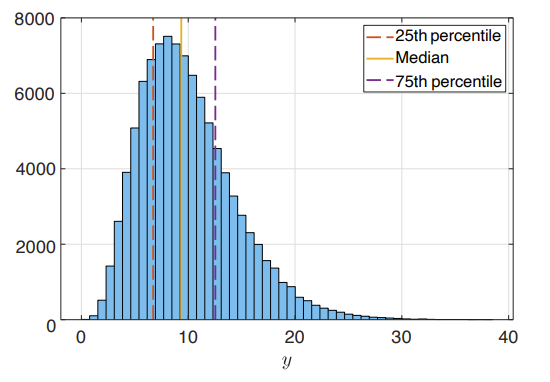
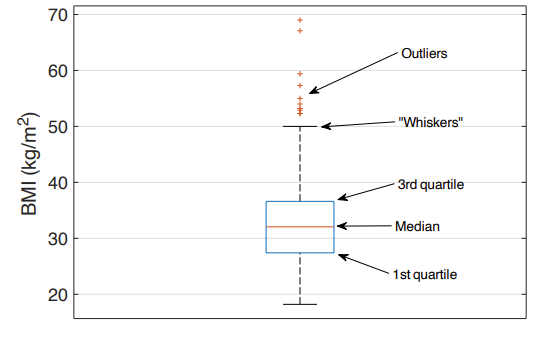
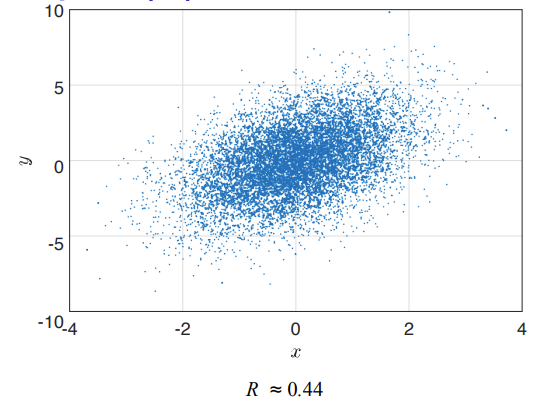
## W3

### The importance/power of data visualization



### Approaches for data visualisation, explain where each approach is appropriate to be used

### Concepts in descriptive statistics

* summarise aspects of the data
* Usually lose information
* gain easy comprehension
* describe properties of the data
* Mean vs Median
* 
  + The mean uses all the values of the sample
  + median uses at most two of the values of the sample
  + Negatively Skewed Data
    - 
  + Positively Skewed Data
    - 
* Percentiles
  + 25th and 75th percentiles- 1st and 3rd quartiles
  + 
  + In boxplot:
  + 
* Correlation/Scatter Plot
* 

### More sophisticated group-by operations in Python

* Categorical-Nominal:
  + Discrete numbers of values, no inherent ordering
  + E.g., country of birth, sex
* Categorical-Ordinal:
  + Discrete number of states, but with an ordering
  + E.g., Education status, State of disease progression
* Numeric-Discrete:
  + Numeric, but the values are enumerable
  + E.g., Number of live births, Age (in whole years)
* Numeric-Continuous:
  + Numeric, not enumerable (i.e., real numbers)
  + E.g., Weight, Height, Distance from CBD

## W4

### Open data and linked open data

Open Data

* Publicly available
  + government allow sharing
* Machine readable
* But not always usable and need the right skills
* common format for open data is “Linked Open Data (LOD)”
  + Triples: subject, verb and object
  + Enables data from different sources to be connected and queried.

### How to access to new data sources through APIs

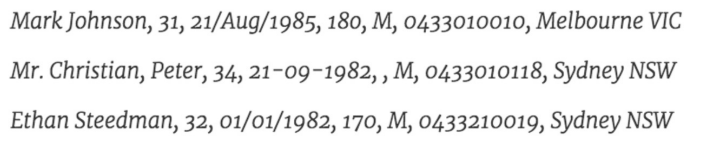
API (Application Programme Interface)

* Routines providing programmatic access to an application
* Computers talk to each other
* For example, if you go to the track order interface, that's an API

### How different APIs work

The user initiates an API call that tells the application to do something, then the application will use an API to ask the web server to do something

### Data quality problems in datasets

* Interpretability issue
  + Example: lack of column header
* Data format issue
  + Data from different sources have different data formats
* Inconsistent and faulty data
  + Mistyped
  + inconsistent entry
  + 
  + Integrity Constraint Violation and an Irregularity

irregularity in terms of the format of data.

For example, dates.

If it is supposed to be DD/MM/YYYY but some data uses MM/DD/YYYY format, then that format is valid, but it is an irregularity.

In terms of integrity constraints, it really depends on the logic of the data. For example, if the date that a loan is obtained is 24th of Oct 2020, then the date that loan is paid is 20th of Dec 2004, this is an integrity constraint violation. Why? because it is not logical to pay off a loan before you even get the loan.

Data wrangling cleans these.

* Missing and incomplete data
* Outliers
* Duplicates

### Data wrangling commands in Python

Process of transforming “raw” data into data that can be analyzed to generate valid actionable results and insights

Including:

* Data pre-processing
* Data preparation
* Data cleansing
* Data transformation

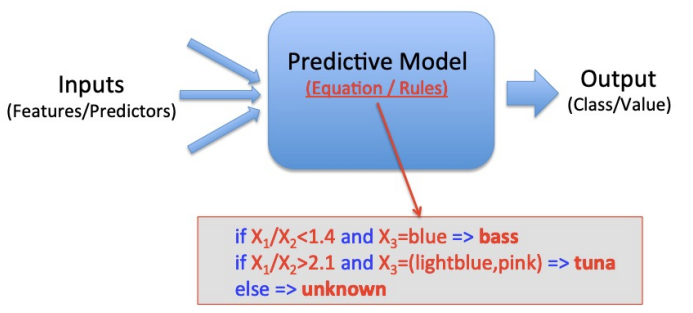
The auditing of data in wrangling is similar to get information from data: eg correlation

## W5

### What are models and predictive models

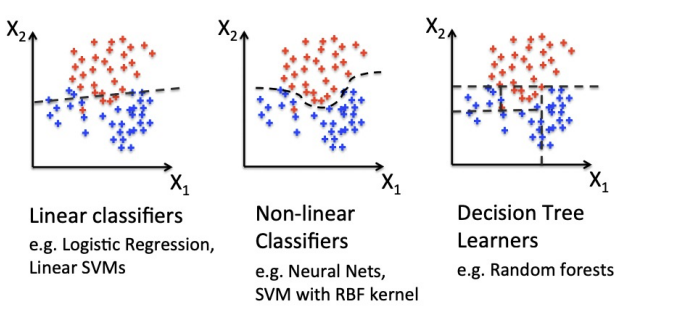
machine learning model is a file that has been trained to recognize certain types of patterns

Predictive Models- uses equations/rules to map the input features to output values

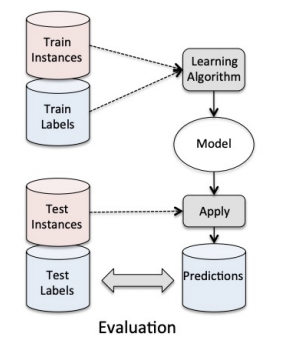


* based on a set of features describing an object
  + binary/categorical classifier
  + real values regression

### Analyse predictive models in different examples



### How to evaluate predictive models

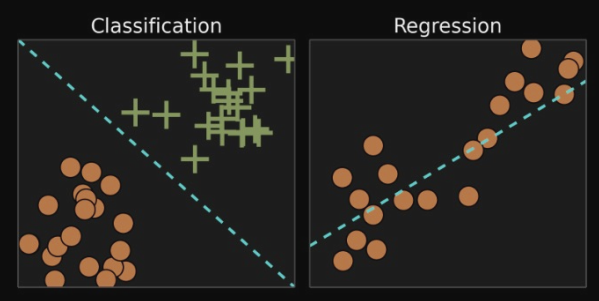


* The more training data the better performance
* the more features the better performance

### Supervised vs Unsupervised Machin Learning



#### Supervised



* All data is labelled and the algorithms learn to predict the output
* The goal is input data (x) we can get predict output Y
  + Classification output is a category
  + Regression output real value e.g., dollars
    - Linear regression for regression
    - Random forest for classification and regression
    - Support vector machines for classification

#### Unsupervised

* All data is unlabelled and the algorithms learn to inherent structure from the input data
* The goal is finding underlying structure or distribution in the data to learn more about the data
  + Clustering finds inherent groupings in the data
  + Association Discover rules that describe large portions of your data (e.g. people that buy X also tend to buy Y)
    - k-means for clustering
    - Apriori algorithm for association

### How to estimate linear regression model

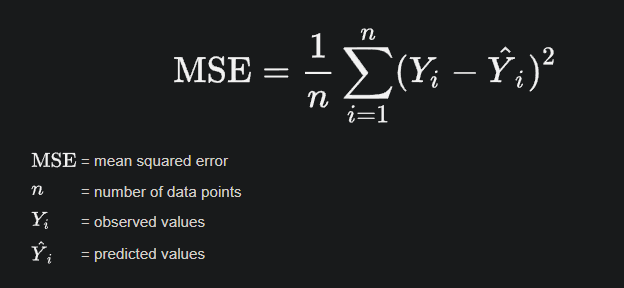
### linear regression and polynomial regression in Python

#### Linear Regression

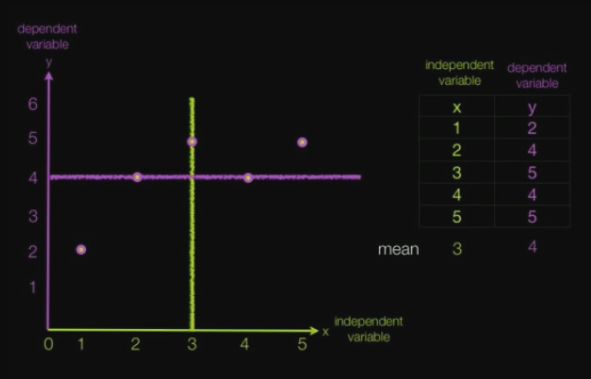
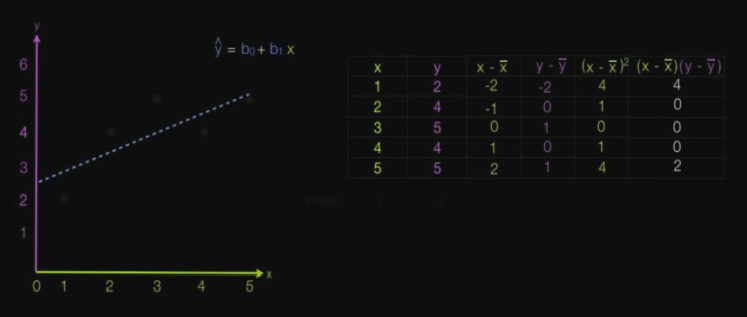
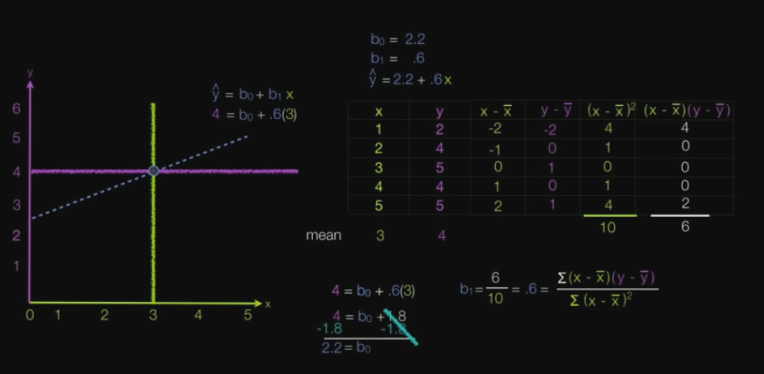
Regression fits a very simple equation to the data 

for prediction for y at the point x using the model 

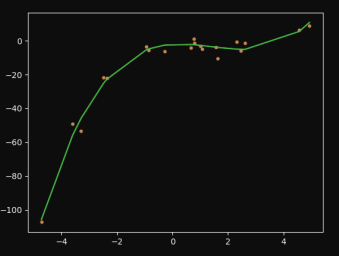
LOSS model mean square error



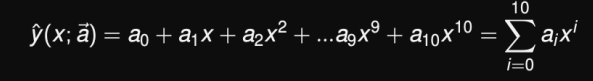
Step:

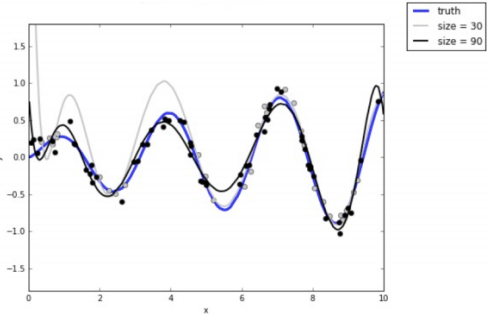
Polynomial Regression



Polynomial regression uses the same linear regression infrastructure to fit a higher order polynomial

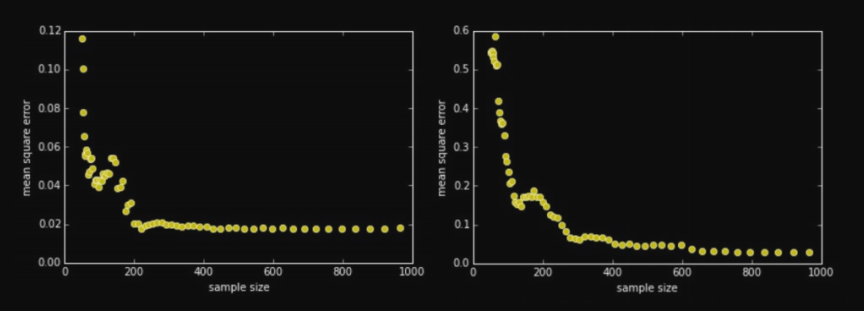
 

#### More Data Improves the Fit



* Blue line is true model that generated the data
* Grey curve is model fit to 30 data points
* Black curve is model fit to 90 data points

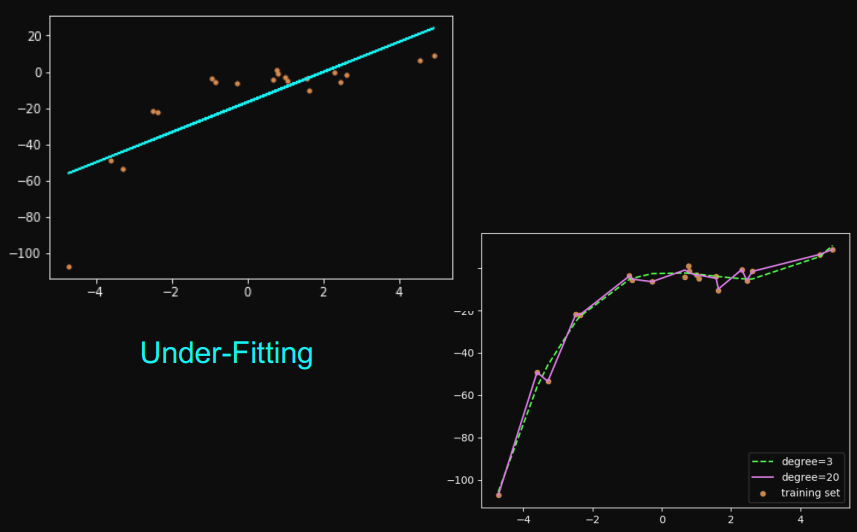
#### Loss decreases with Training Data



* MSE decreases as the amount of training data grows
* Different learning algorithms exhibit different behaviour

## W6

### Overfitting and underfitting of different models



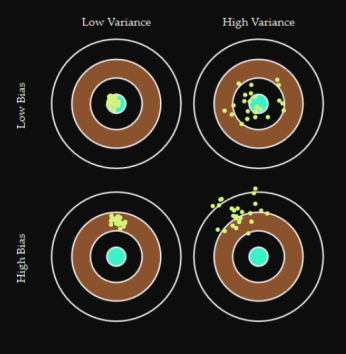
#### Overfitting

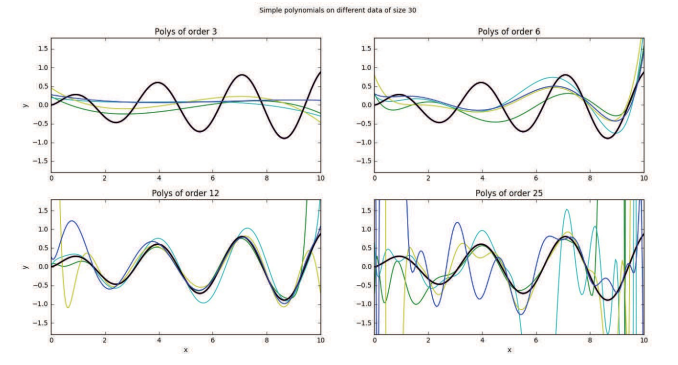
The more parameters a model has, the more complicated a curve it can fit.

If we don’t have very much data and we try to fit a complicated model to it, the model will make wild predictions.

* Small polynomial; cannot fit the data well; said to have high bias
* Large polynomial; can fit the data well; fits the data too well (follow noise); said to have small bias
* Poor fit due to high bias called underfitting
* Poor fit due to low bias called overfitting

### Bias and variance trade-off



* Bias: measures how much the prediction differs from the desired regression function.
* Variance: measures how much the predictions for individual data sets vary around their average.
* 
* The "bias-variance" trade-off is minimizing both bias and variance. This can be done by taking an ensemble then choosing the one with the lowest variance and bias
* The trade-offs are:
  + The higher the complexity of your fit, the higher the variance (bad) but the lower the bias (good). So, you're trading lower bias for higher variance.
  + The lower the complexity of your fit, the higher the bias (bad) but the lower the variance (good). So, you're trading lower variance for higher bias.

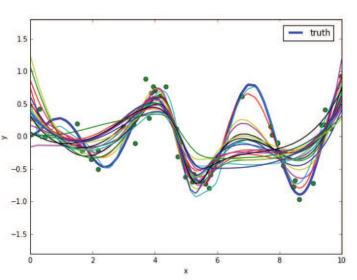
### “No Free Lunch Theorem”

*if an algorithm performs well on a certain class of problems, then it degraded performance on the set of all remaining problems*

eg:

* Naive Bayesian classification performs well for text classification with smaller data sets
* there is no universally good machine learning algorithm

### What are ensemble models



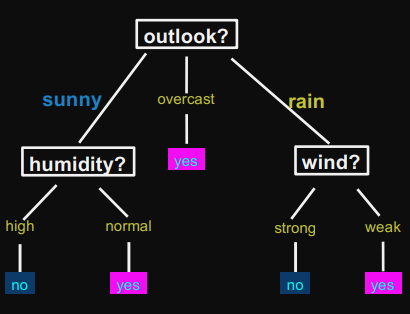
* ensemble is a collection of possible/reasonable models
* and get the most common one
* generating an ensemble is a whole statistical subject in itself

## W7

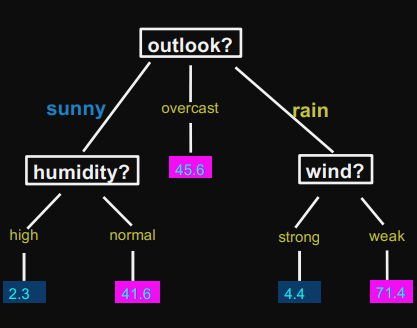
### Differentiate between classification and regression models

### How decision trees and regression trees work

Decision Trees

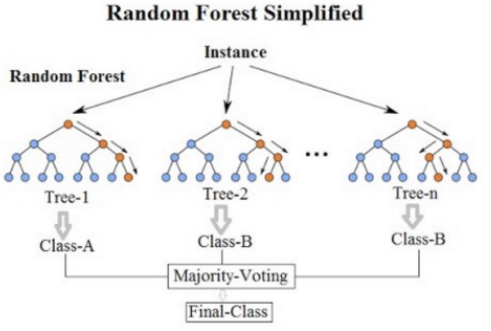
* Predict binary (or categorical) outcomes
* Prediction is the most common values
* 

Regression Trees

* Predict continuous (i.e. real) values
* Prediction is usually the average value
* 

### How random forest works

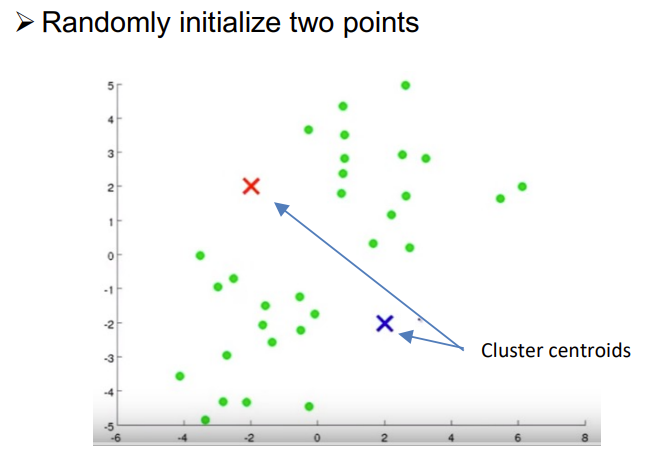
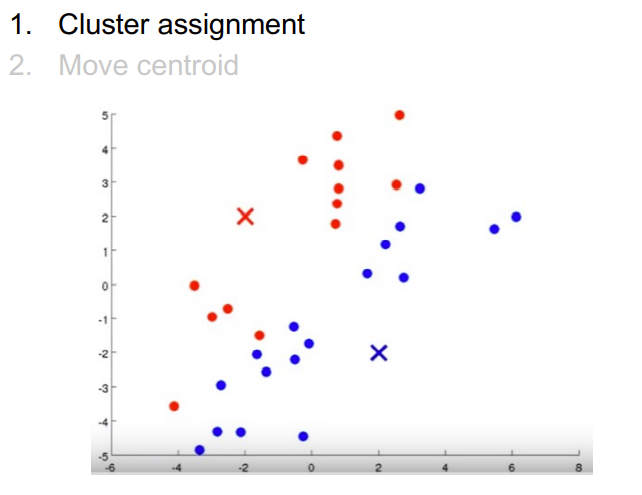
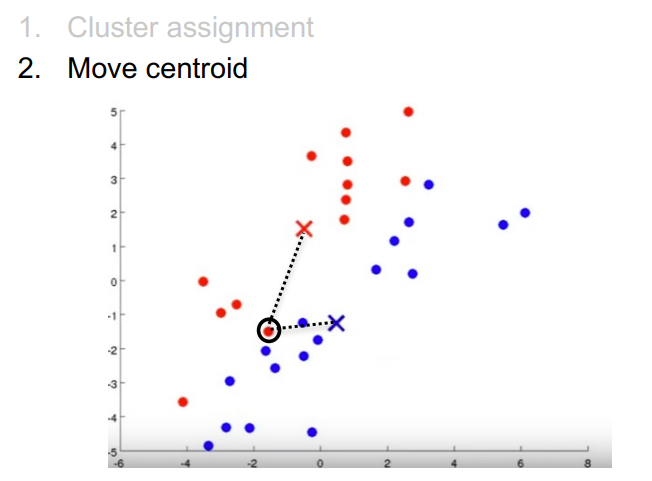
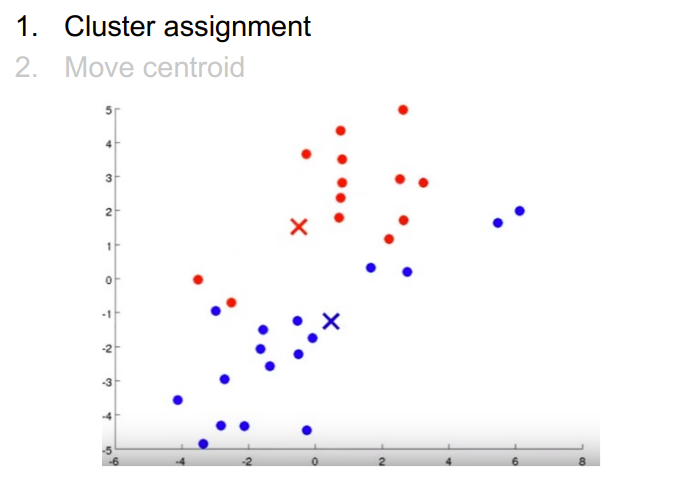
Ensemble learning method that operate by constructing a number of decision trees



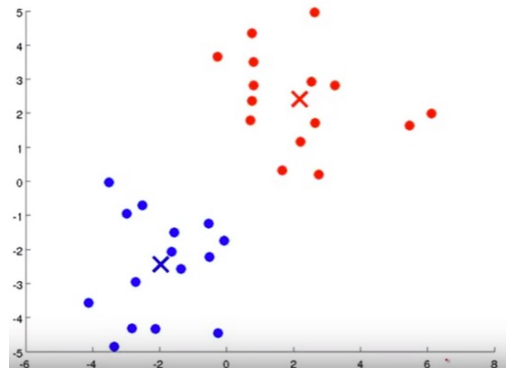
### How k-means clustering works

* K=the number of clusters
* Cluster centroid= The mean (average) of the location of all data points in a cluster

#### Step:

End like



#### K-means Algorithm

Input:

- A set of data points

- The number of clusters (K)

Method:

- Select K initial random points

- Repeat

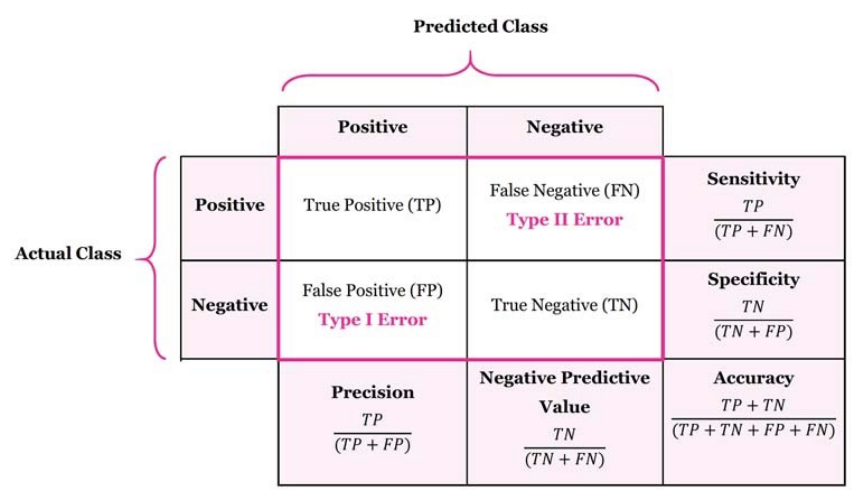
– Cluster assignment

– Move the cluster centroids to the mean value of data points in the cluster

- Until no change

### Confusion matrix and prediction accuracy

A tool to measure performance for classification



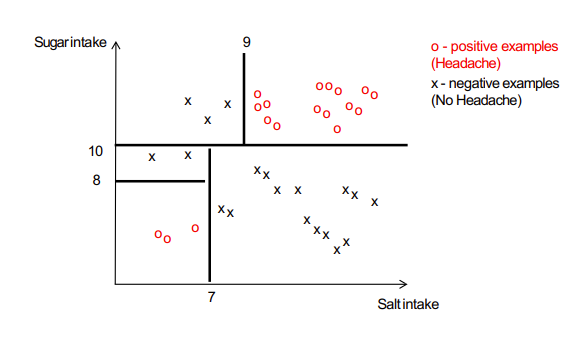
* Accuracy: Overall, how often is the prediction correct?
* Sensitivity (Recall): When the actual value is positive, how often is the prediction correct?
* Specificity: When the actual value is negative, how often is the prediction correct?
* False Positive Rate: When the actual value is negative, how often is the prediction incorrect?
* Precision: When a positive value is predicted, how often is the prediction correct?

### Different classification metrics

* Spam filter: Optimise precision or specificity
  + False negatives (spam goes to the inbox) are more acceptable than false positives (non-spam is caught by the spam filter)
* Fraudulent transaction detector: Optimise sensitivity
  + False positives (normal transactions that are flagged as possible fraud) are more acceptable than false negatives (fraudulent transactions that are not detected)

### Recursive Partitioning

* At each iteration, we divide the data to group similar instances together

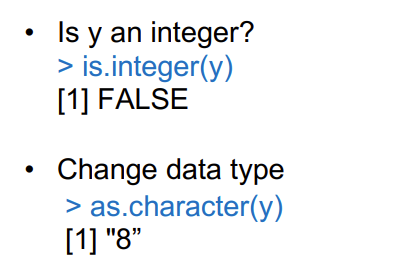
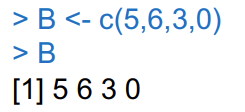


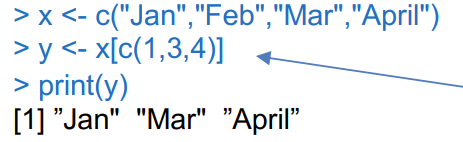
## W8

### Essentials for coding in R for data science

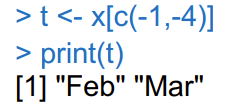
A language for analysing and visualising data

### Explain and interpret given R commands

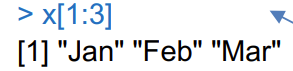
* 
* Define a vector
* 
* Accessing vector elements using position

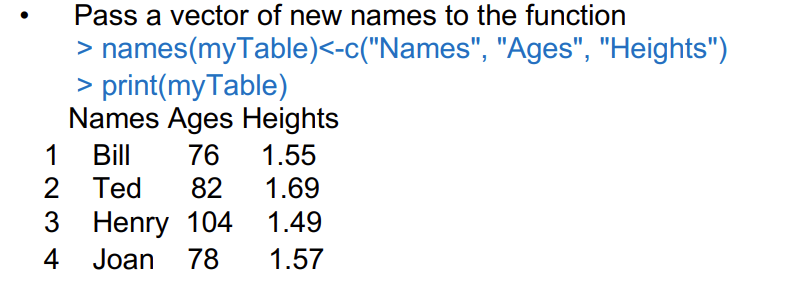
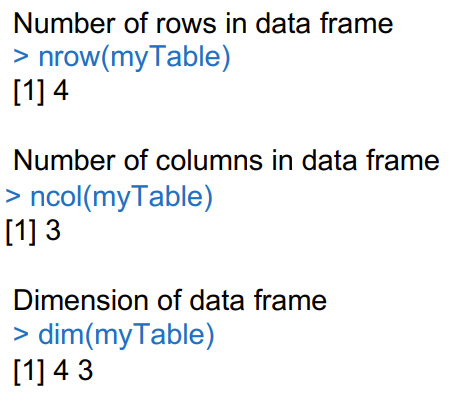
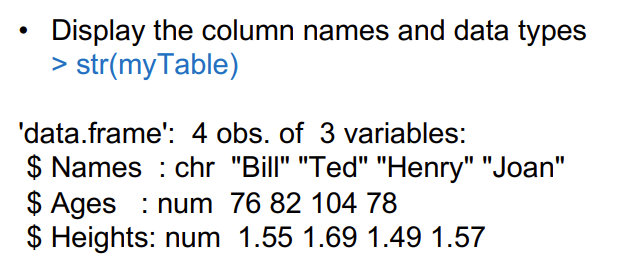


* Accessing vector elements using negative indexing



* Access range of values in vector



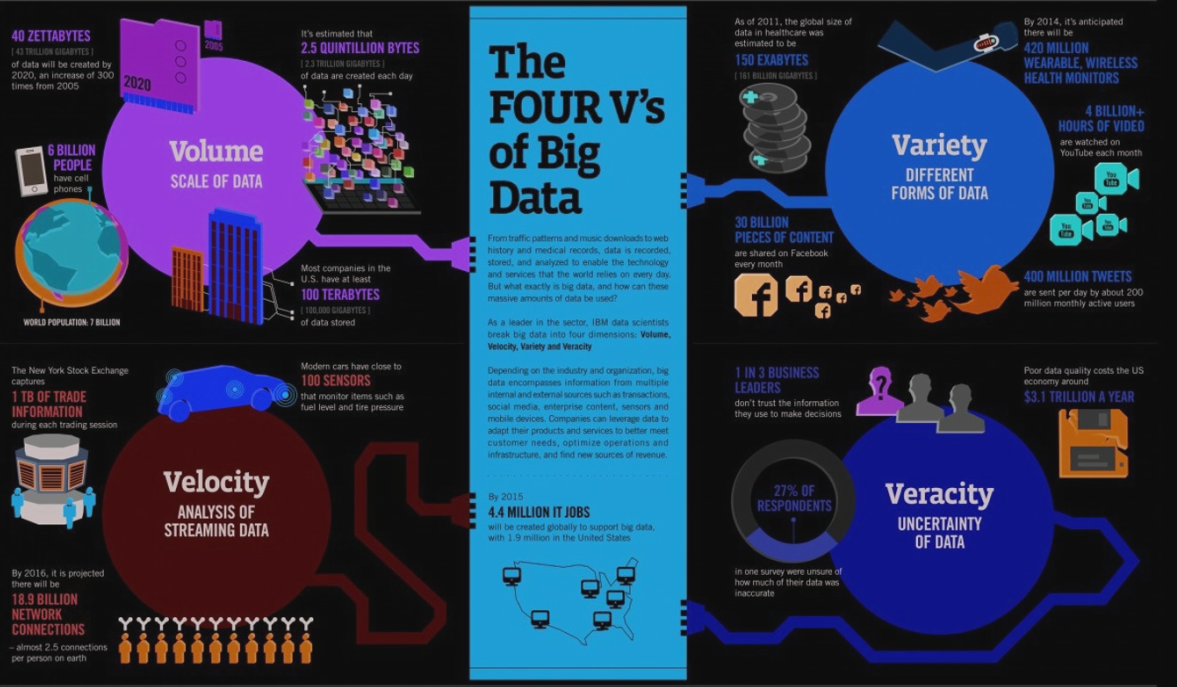
* Rename The Columns
  + 
* 
* 

### Apply R commands for data wrangling, visualisation, exploration and analysis

## W9

### Characterising big data: Volume, Velocity, Variety, Veracity

The first charactisations of big data were by someone with a penchant for alliteration ... others followed



Big Data- includes data sets with sizes beyond the ability of commonly used software tools

BIG DATA is ANY attribute that challenges CONSTRAINTS of a system CAPABILITY or BUSINESS NEED

* These characterise bigness, adequately
* 3V: Volume, Velocity and Variety
* Other V’s characterise problems with analysis and understanding
  + Veracity: correctness, truth
  + Variability: change in meaning over time, e.g., natural language
* Other V’s characterise aspirations
  + Visualisation: one method for analysis
  + Value: what we want to get out of the data

● Volume is size of data.

● Velocity is the frequency/Pace of incoming data that needs to be processed.

● Variety refers to different types of data.

● Veracity refers to the fact that how accurate or truthful a data set may be. More specifically, how accurate and reliable the data is?

### What is metadata? different types of metadata

#### Metadata

Is:

* Data about data is critical to understanding
* Structured so that a computer can process

Metadata types:

* Descriptive- Describes content for identification e.g. title, author of a book
* Structural- Documents relationships and links e.g. chapters in a book
* Administrative- Helps to manage information e.g. version number

==: Structured information that describes, explains, locates, or otherwise makes it easier to retrieve, use or manage an information resource

#### Why Use Metadata?

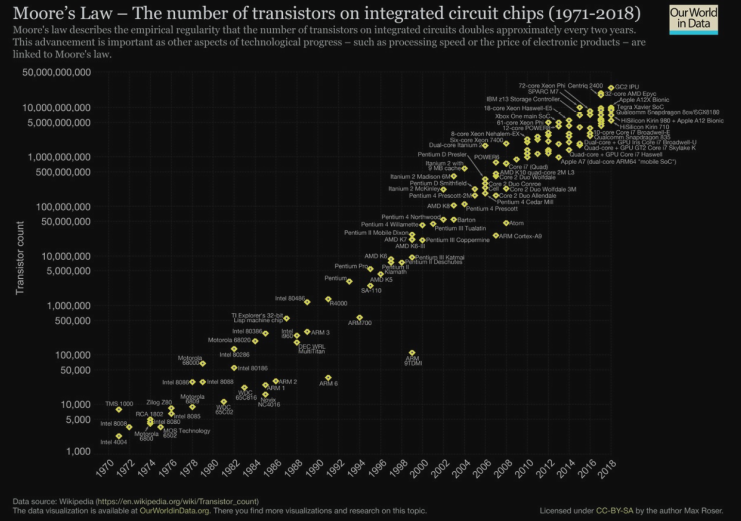
* Facilitate data discovery (find more discovery)
* Help users determine the applicability of the data
* Enable interpretation and reuse
* Clarify ownership and restrictions on reuse

#### Dimensions of data

Infographics on data dimensions

### Growth laws related to big data: Moore’s law, Koomey’s law, Bell’s Law and Zimmerman’s Law

#### Moore’s Law



Number of transistors per chip doubles every 2 years

o More memory

o Bigger CPUs

o Faster memory, CPUs

#### Koomey’s



Corollary of Moores Law exponential growth（指数性）

Amount of battery needed will fall by a factor of 100 every decade

Leads to ubiquitous computing

#### Bell’s Law

Roughly every decade a new, lower priced computer class forms based on a new programming platform, network, and interface resulting in new usage and the establishment of a new industry

* Yes: PCs, mobile computing, cloud, internet-of things
* No: Java, big data, Hadoop, flash memory

#### Zimmerman’s Law

* Surveillance is constantly increasing
* Privacy constantly decreasing

#### Unix Shell

Command line interface to a Unix computer

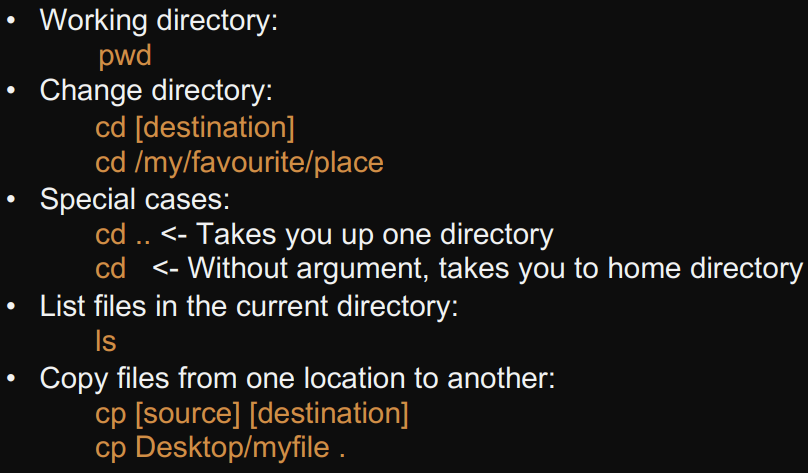
Why are shells interesting for Data Scientists?

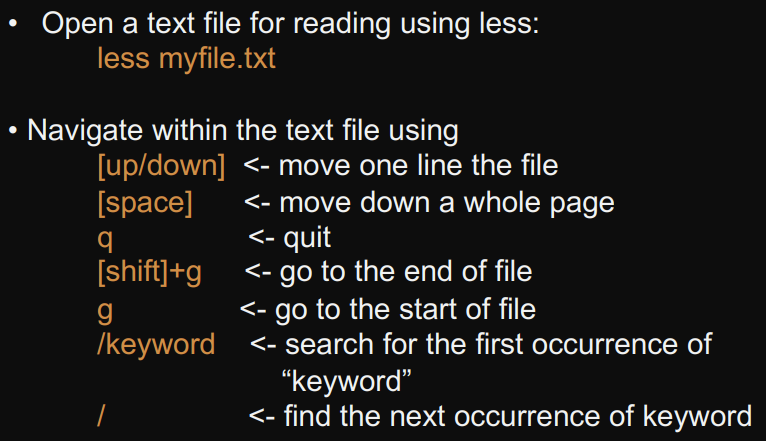
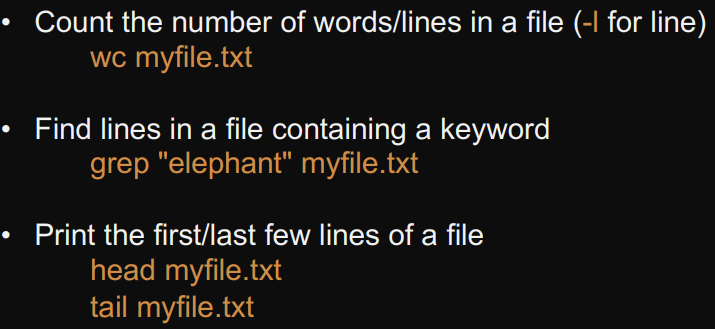
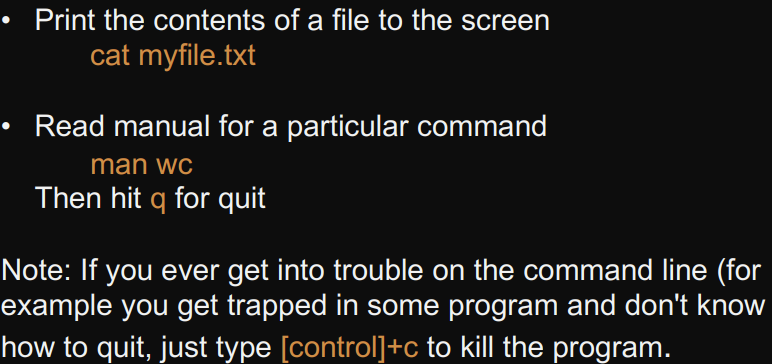
* Provide powerful & easy way to manipulate large data files
* And move data around a network

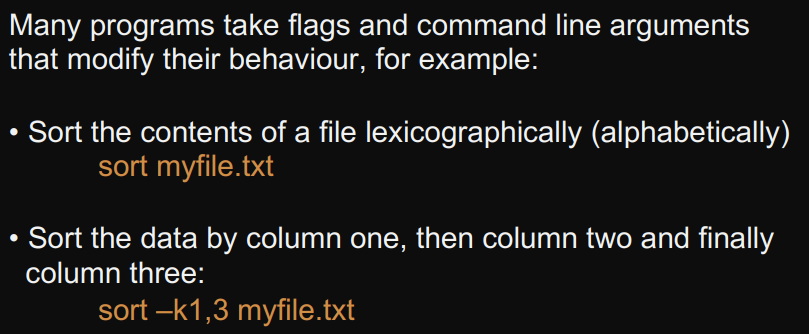
Super-computers are typically UNIX based

* Easier to manipulate and wrangle Big Data
  + Simple and easy to learn.
  + Ideal for textual data, e.g. unstructured data for social networks, life sciences, system logs, etc.
  + Quick to sort, search, match, replace and clean your data.
* Explore data before you use it in Python or R

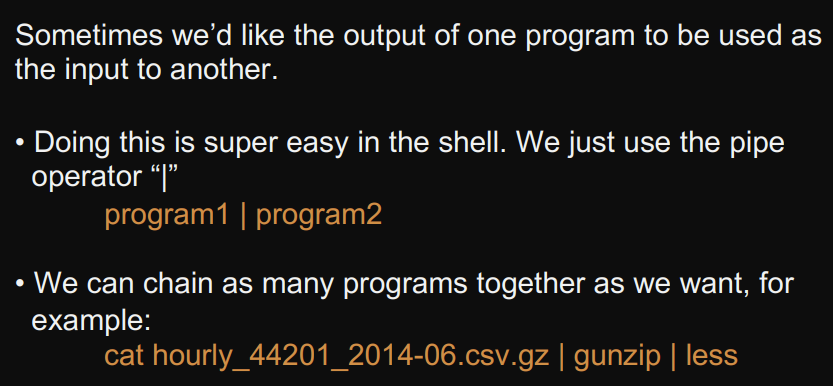
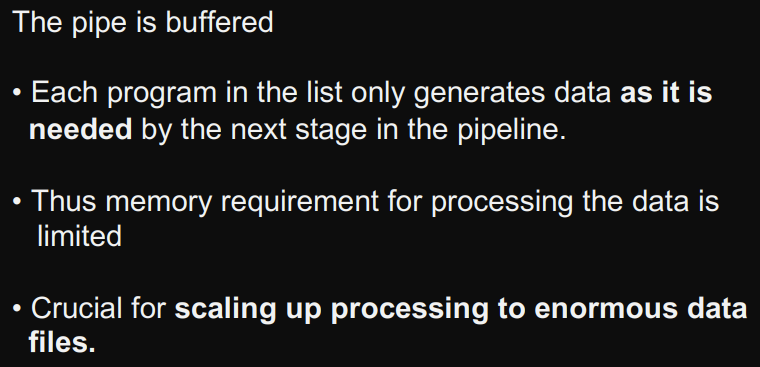
Navigating the Filesystem



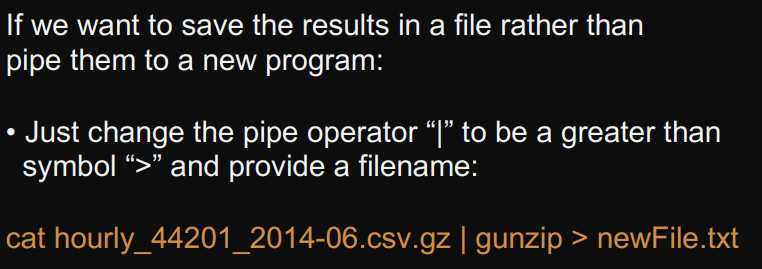
Reading a Text File  

Flags and Arguments

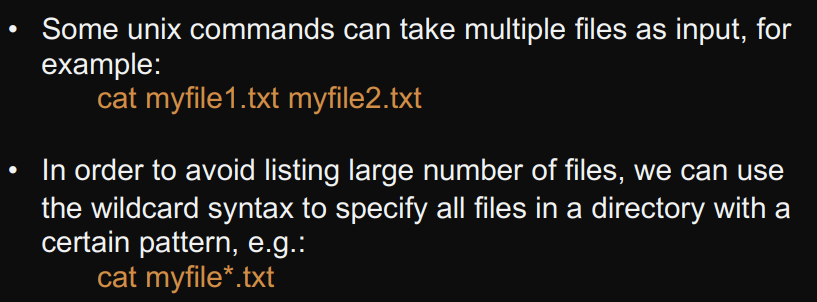
Pipes

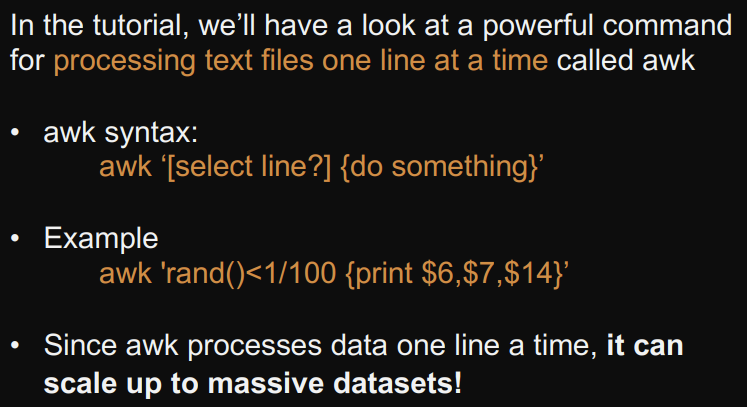
Redirects



Wildcards

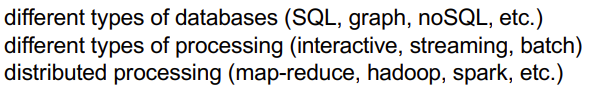


awk



## W10

### Processing big data



#### Unix Shell

### Useful for managing and manipulating large files

* without ever loading them fully into memory
* using pipes allow us to process files as a stream
* allows us to deal with files that are too big for applications and/or don’t fit into memory
  + less to view large files
  + grep to search large files
  + awk to process them one line at a time

### Databases

* in-database analytics: the analytics is done within the DB
* key-value: value accessible by key, e.g., hash table
* information silo: an insular information system can’t operation with other information systems, e.g.,
  + if two big banks merge, then initially their RDBMSs will be siloed
  + insurance company, auto and home insurance customer RDBMSs may be siloed
* Many NoSQL and SQL DB offer:
  + large scale, distributed processing
  + robustness achieved
  + general query languages
  + notion of consistency

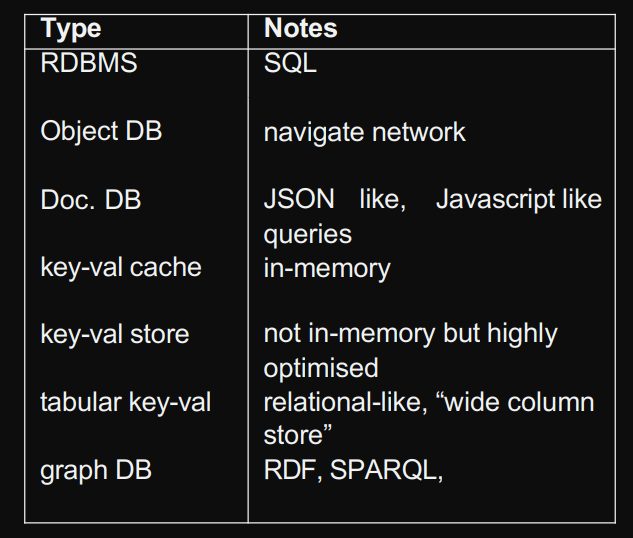
#### SQL Review

* Relational Database Management Systems (RDBMS)
* ACID: atomicity, consistency, isolation, and durability

#### JSON

* no fixed format
* key-value pairs, hierarchical
* “friendly” alternative to XML
* self-documenting structure

#### Beyond SQL Databases



JSON and graph database is better because: doc.db, give more flexibility

Use SQL database when data is structured and unchanging

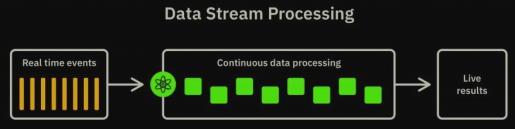
Use NoSQL database when

* Storing large volume of data with little to no structure
* Data changes rapidly

### Processing

Batch: data stored and analysed in large blocks 

Streaming: massive data streaming through system with little storage



Interactive: bringing humans into the loop

#### Batch vs Stream

Batch:

* where you don’t need real-time analytics results
* when it is more important to process large volumes of data to get more detailed insights than speed

Streaming:

* massive data streaming through system with little storage
* Sampling can be a solution to process massive datasets

批处理一般是解决离线计算数据量大，计算时间慢的问题，流处理相反是为了解决实时计算或是近实时计算问题，当然有了实时的要求就会使处理的数据量变少，但是计算速度要求更快

#### Background Concepts

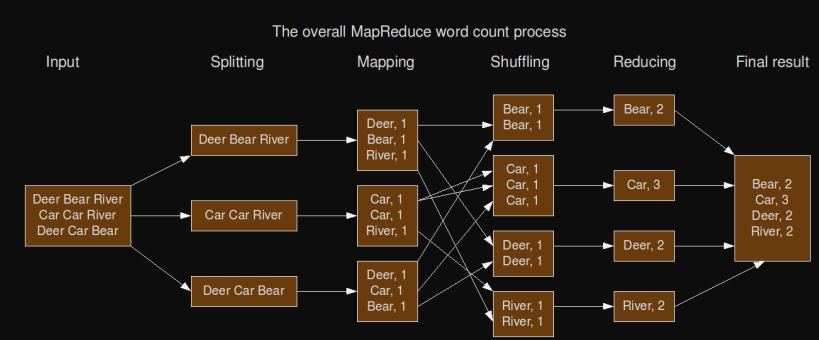
* in-memory: in RAM, i.e., not going to disk
* parallel processing: performing tasks in parallel
* distributed computing: across multiple machines
* scalability: to handle a growing amount of work; to be enlarged to accommodate growth (not just “big”)
* data parallel: processing can be done independently on separate chunks of data
  + yes: process all documents in a collection to extract names
  + no: convert a wiring diagram into a physical design (optimisation)

### Distributed Analytics

legacy systems provide powerful statistical tools on the desktop but often-times without distributed or multi-processor support

## Map-Reduce

* Simple distributed processing framework developed at Google
* To run on commodity hardware; so has fault-tolerant infrastructure
* quite simple

Example 

* for a simple word-count task:

(1) divide data across machines

(2) map() to key-value pairs

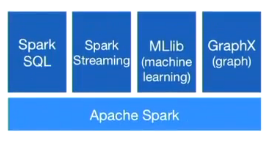
(3) sort and merge() identical keys

* requires simple data parallelism followed by reducing process
* stopped using

## Hadoop

* Open-source Java implementation of Map-Reduce
* provides an inexpensive and open-source platform for parallelising processing
* not suited to streaming (suitable for offline processing)
* This curve represents the maturity, adoption, and social application

## Spark

* interfaces in Java, Scala, Python, R
* provides in-memory analytics
* work with some Hadoop ecosystem
* 
* It is real time data processing
* includes Map-Reduce capabilities, provides real-time, in-memory processing, much faster than Hadoop

### What is deep learning(Neutral Network)



* machine learning subfield of learning representations of data
* algorithms attempt to learn representation by using a hierarchy of multiple layers
* If provide the system tons of information, it begins to understand it and respond in useful ways

## W12

### Confidentiality and privacy

Privacy is (for our purposes) having control over how one shares oneself with others.

Security as the protection of data, preventing it from being improperly used

Implicit data: data not explicitly stored but precision from available data

Confidentiality is information privacy, how information about an individual is treated and shared

* For many apps or services, you must accept their data sharing policies or you can’t use their services fully
* There could be an agent to interact in a narrative form with individual consumers
  + Are you willing to share your health data with company X?

### Regulatory compliance

* Ethics: the moral handling of data
* regulations to ensure that confidentiality is protected
* The process of ensuring you meet regulations is called compliance
* PCI (Payment Card Industry) standard
  + reduce credit card fraud
    - By placing specific regulations- information in encrypted
  + Companies who handle credit card have to with PCI standards
  + Audit (validation of compliance) is done annually

### Data management

#### Why Manage Data

* data is very valuable; data collection is usually time consuming and hard
* Large amount of data and documents are being generated with high growth rate
* Multiple sources of data

#### What is Data Management?

* Data management is the development, execution and supervision of plans, policies, programs and practices that *control, protect, deliver and enhance the value of data and information assets*

#### Issues in Data Management

Data management Deals with issues:

* replication and persistence
* security

#### Data Management and Data Science

* internet advertising: what implicit and explicit data is stored about a user
  + implicit: facebook like motorcycle, it will push motorcycle
  + explicit: something that is stated plainly-ur like in facebook
* medical informatics: for predicting fungal infections from nursing notes, the team needs to abide by confidentiality and security