# FIT3152 WEEK 1-8

Data Analytics (Monash University)

# FIT3152

## WEEK 1 – INTRODUCTION TO DATA SCIENCE

What is data science?

- Data Science is making a product out of data and using graphics to communicate the result
- Complex problems of societal interest/utulity
- Large data sets, multiple data sets, messy data, incomplete data

Data scientist skills

- Understand problem from client's perspective
- Collect, cleanse and manage data
- Understand the data
- Anaylse and model the data using statistical and machine learning techniques
- Communicate the results simply and effectively

Examples

- Road shooter found via mass data collection
- Food network
- Climate change time series
- Customer analytics

## WEEK 2 – DATA VISUALISATION

Dimensions

- Graphics can show many dimensions
- Include colour, size, position, adjacency, connect, shape

Graphic types – Visualisation Zoo

- Time-series data
    - Index charts
    - Stacked graphs
    - Small multiples
    - Horizon Graphs
- Statistical distributions
    - Stem-and-leaf plots
    - Q-Q plots
    - SPLOM
    - Parallel coordinates
- Maps
    - Flow maps
    - Chloropleth maps
    - Graduated symbol maps
    - Cartograms
- Hierarchies
    - Node-link diagrams
    - Adjacency diagrams
    - Enclosure diagrams
- Networks
    - Force-directed layouts
    - Arc diagrams
    - Matrix views

# WEEK 3 – DATA MANIPULATION

Manipulation in R

- Making tables and summaries
- Applying functions and considering factors
- Transforming data
    - Aggregate
    - Cor
        - Correlation function only works on columns
    - By
        - By(*data frame*, *column*, function(df) *function*)
            - Df is a temporary data frame created for each factor
        - Applied to a df and returns list
    - As.table
        - List to a table
    - As.data.frame
        - Table to a df
    - Colnames
        - colnames(data frame) = c("name1", "name2")
        - Adds column name
    - Merge
    - Cbind
    - Rbind
    - Max
    - Which.max
        - Determines the index of the maximum numeric vector
    - Do.call
        - Calls a function on a list of arguments

# WEEK 4 – REGRESSION MODELS

Fitting the regression

- Simple least squares regression assumes that the relationship is approximately linear and errors are approximately normally distributed
- Line of best fit

Regression diagnostics

- Look at median and coefficients
- P-value for variables and for overall model
    - Small p-value = Is significant
- R-squared – Coefficient of determination
    - High r-squared = How close the data is to the fitted regression line

Multiple linear regression

Regression with qualitative variables

- 0, 1 column

# WEEK 5 – NETWORKS

Structure

- Nodes (vertices) and edges (lines/arcs)
    - Can be directed with arrows or undirected
    - Edges can be weighted
- Walk: Sequence of links

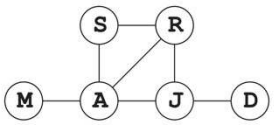
- Path: Walk with no repeated vertices
- Cycle: Walk that begins and ends at the same vertex
- Geodesic: Shortest path between two vertices
- Length: Number of links in walk/path
- Connected: Path between each pair of vertices
- Directed graphs: Travel on each edge is permitted in one direction only
- Link sequences
    - Loop: An edge from a vertex to itself
    - Complete: Graph where every vertex is joined to another vertex
    - Subgraph: Subset of a graph
    - Clique: Subgraph that is complete (super connected circle)
    - Simple: Graph with no loops or multi-edges
- Degree: Number of edges connected to a vertex
    - For directed, in-degree and out-degree
    - degree(graph object)
- Diameter: Longest geodesic
- Average path length: Average geodesic
    - Create distance matrix first
- Degree distribution: Probability distribution describing the magnitude of the vertices (weighted)

- *Diameter*: = *max(dist(u,v))* = 3
- *Degree distribution*: ⟶

| Degree | N |
|---|---|
| 0 | 0 |
| 1 | 2 |
| 2 | 1 |
| 3 | 2 |
| 4 | 1 |

- *Average path length*: 1.667

| | J | A | S | D | R | M |
|---|---|---|---|---|---|---|
| J | | 1 | 2 | 1 | 1 | 2 |
| A | | | 1 | 2 | 1 | 1 |
| S | | | | 3 | 1 | 2 |
| D | | | | | 2 | 3 |
| R | | | | | | 2 |
| M | Distance Matrix | | | | | |



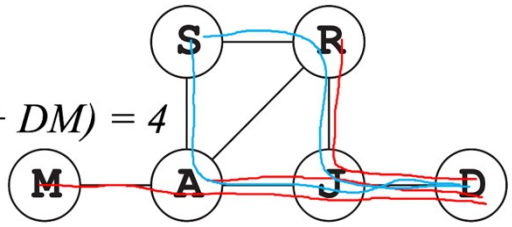

Vertex characteristics

- Importance of a vertex is based on degree and centrality
- Centrality of a vertex
    - Betweenness
        - Indicates the degree to which the vertex is between other vertices
        - Calculate by hand
        - Measures the hub potential of a node
        - betweenness(graph object)

$$c_B(v) = \sum_{s \neq t \neq v \in V} \frac{\sigma(s,t|v)}{\sigma(s,t)}$$

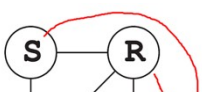
$$c_B(J) = (DR + DA + DS^* + DM) = 4$$

$$c_B(M) = 0$$



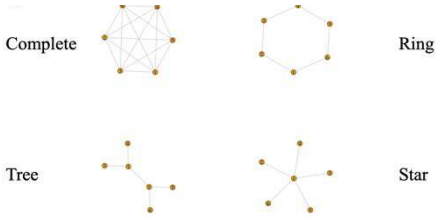

    - Closeness
        - Vertex is close if there is a small total distance between it and all the other vertices
        - Calculate by hand
        - Measures how well a node is connected locally
        - closeness(graph object)

$$c_{Cl}(v) = \frac{1}{\sum_{u \in V} dist(u,v)}$$



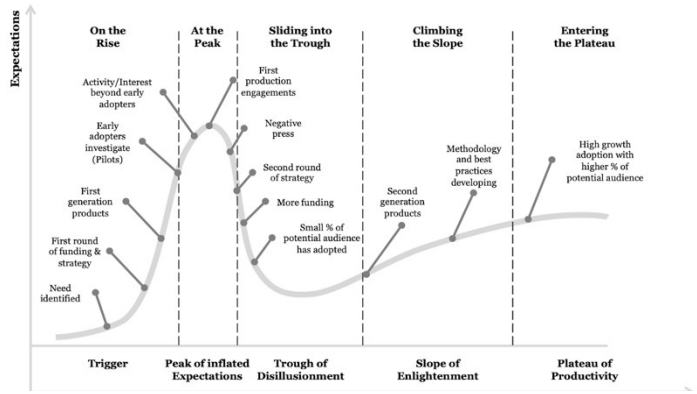
- Eigenvector
  - Higher weight to vertices with neighbours that are more central
  - Weights a node according to the quality of its connection where nodes connected to important nodes are ranked higher
  - evcent(graph object)

Four network topologies
- Complete (most robust)
- Ring
- Tree
- Star (most fragile)




# WEEK 6 – TIDY AND DIRTY DATA

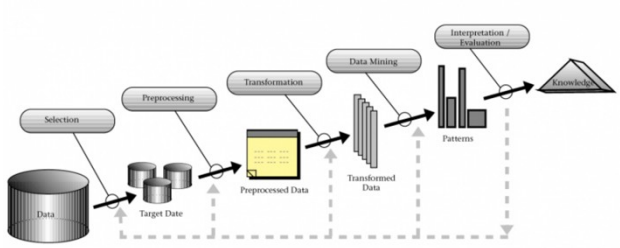
Gartner Hype Cycle



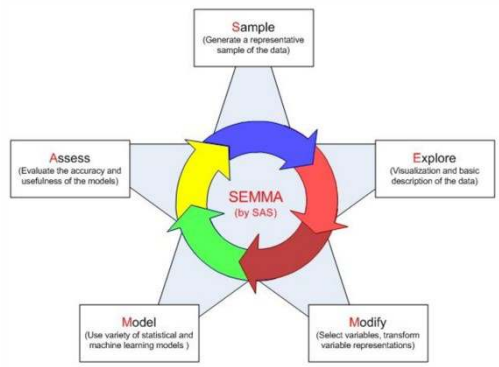

Five stages of understanding
- Descriptive
  - What happened?
  - Condense data into smaller, more useful pieces of information
- Diagnostic
  - Why did it happen?
  - Data analysis by employing predefined criteria
- Explorative
  - What might be interesting?
  - Uncover underlying structure, patterns, and anomalies
  - Manual discovery and automated discovery (machine learning & clustering)
- Predictive
  - What is likely to happen?
  - Predictive modelling on historical data ot produce future likelihood of events
  - E.g. Random forest, bagging, & boosting
- Prescriptive
  - What can we do about it?
  - Suggests the best option for handling a future scenario

Data science methodologies (NE)

- KDD (Knowledge Discovery in Databases)
  - Broad process of finding knowledge in data
  - Emphasises the high-level use of data-mining methods
  - Selection -> preprocessing -> transformation -> data mining -> interpretation/evaluation

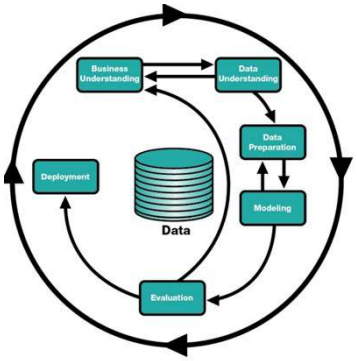



- SEMMA (Sample, Explore, Modify, Model, and Assess)




  - Methodology for data mining processes

- CRISP-DM (Cross-Industry Standard Process for DM)
  - Preferred due to inclusion of business aspects
  - Business understanding
    - Understanding the project objectives and requirements from a business perspective
  - Data understanding
    - Initial data collection and activities to get familiar with the data
  - Data preparation
    - Activities required to construct the final dataset from the initial raw data
    - Performed repeatedly and not in any prescribed order
  - Modelling
    - Modelling techniques are selected and applied
  - Evaluation
    - Evaluate model and review steps taken to ensure it properly achieves the business objectives
  - Deployment
    - Knowledge will need to be organised and presented in a way that the customer can use




| Business Understanding | Data Understanding | Data Preparation | Modeling | Evaluation | Deployment |
|---|---|---|---|---|---|
| **Determine Business Objectives**<br>*Background*<br>*Business Objectives*<br>*Business Success Criteria* | **Collect Initial Data**<br>*Initial Data Collection Report* | **Select Data**<br>*Rationale for Inclusion/ Exclusion* | **Select Modeling Techniques**<br>*Modeling Technique*<br>*Modeling Assumptions* | **Evaluate Results**<br>*Assessment of Data Mining Results w.r.t. Business Success Criteria*<br>*Approved Models* | **Plan Deployment**<br>*Deployment Plan* |
| **Assess Situation**<br>*Inventory of Resources*<br>*Requirements, Assumptions, and Constraints*<br>*Risks and Contingencies*<br>*Terminology*<br>*Costs and Benefits* | **Describe Data**<br>*Data Description Report* | **Clean Data**<br>*Data Cleaning Report* | **Generate Test Design**<br>*Test Design* | **Review Process**<br>*Review of Process* | **Plan Monitoring and Maintenance**<br>*Monitoring and Maintenance Plan* |
| **Determine Data Mining Goals**<br>*Data Mining Goals*<br>*Data Mining Success Criteria* | **Explore Data**<br>*Data Exploration Report* | **Construct Data**<br>*Derived Attributes*<br>*Generated Records* | **Build Model**<br>*Parameter Settings*<br>*Models*<br>*Model Descriptions* | **Determine Next Steps**<br>*List of Possible Actions*<br>*Decision* | **Produce Final Report**<br>*Final Report*<br>*Final Presentation* |
| **Produce Project Plan**<br>*Project Plan*<br>*Initial Assessment of Tools and Techniques* | **Verify Data Quality**<br>*Data Quality Report* | **Integrate Data**<br>*Merged Data* | **Assess Model**<br>*Model Assessment*<br>*Revised Parameter Settings* | | **Review Project**<br>*Experience Documentation* |
| | | **Format Data**<br>*Reformatted Data* | | | |
| | | *Dataset*<br>*Dataset Description* | | | |

Dirty data

- Can be:
  - Incorrect
    - Valid data (e.g. dates)
  - Inaccurate
    - Correct but not accurage (e.g. Sydney, VIC)
  - Incomplete
    - Missing/empty fields
  - Duplicate
  - Violates business rules
    - E.g. date should precede expiration date
  - Inconsistent
    - Due to uncontrolled data redundancy (e.g. Customer listed three times)
  - Non-integrated
    - Data is stored redudantly and inconsistently across many systems
    - Primary keys don't match or are not unique

Tidy data

- Consistent format that has:
  - Each variable in its own column
  - Each observation in its own row
  - Each value in its own cell
- Actions:
  - Replace missing values
  - Standardisation
  - Normalisation

# WEEK 7 – DECISION TREES

Machine Learning

- Automated learning of a concept given some examples of data
- How can a model learn a concept?
  - Descriptive – captures training data
  - Predictive – generalises to unseen data
  - Explanatory – describes the concept to be learned

Classification

- Find a model to predict class as a function of the other attributes
- Goal is for unseen records to be assigned a class as accurately as possible
- Test set is used to determine the accuracy
- Training set is used to build the model

Decision trees

- Can be used to profile existing data and classify new instances
- Robust to noise and missing values
- Leaf nodes (class) and non-leaf nodes
- Branches corresponding to the values of the decision attributes
- Applications
    - Profiling
    - Classifying unseen data
    - Example areas:
        - Consumer credit evaluation
        - Spam filtering
        - Predicting customer churn
- Pros
    - Easy to interpret
    - Can handle discrete and continuous input
    - Robust to outliers and missing values
- Cons
    - Unstable
    - Can become overly complex (overfitting)

Top-down induction: ID3

- Greedy approach
    - At each step, determine the best decision for the decision attribute
    - May not be the best decision overall but stick with it
    - May not result in best overall decision tree
    - Goal is to increase the homogenity (same) of the resulting datasets with respect to the class variable
- Information gain
    - ID3 uses information gain as splitting criteria
    - Measures how well a given attribute separates the training set into homogenous groups
    - Expected reduction in entropy
- Entropy
    - Measures the uncertainty in a random variable
    - Ordered data = low entropy
    - Entropy is 0 if all members belong to the same class
    - For a two class problem

$$\begin{aligned} \text{Entropy}(S) &= -P_{c1}\log_2(P_{c1}) - P_{c2}\log_2(P_{c2}) \\ &= -\frac{N_{c1}}{N}\log_2(\frac{N_{c1}}{N}) - \frac{N_{c2}}{N}\log_2(\frac{N_{c2}}{N}) \end{aligned}$$

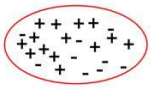
    - For a multi-class problem

$$\text{Entropy(S)} = -\sum_{i=1}^{C} \text{P}_i \log_2(\text{P}_i)$$

$$= -\sum_{i=1}^{C} \frac{\text{N}_i}{\text{N}} \log_2(\frac{\text{N}_i}{\text{N}})$$

- Example:

E.g. Suppose S is a collection of 14 examples, 9 positive and 5 negative → [9+,5-]



$$\text{Entropy(S)} = -\text{P}_{c1}\log_2(\text{P}_{c1}) - \text{P}_{c2}\log_2(\text{P}_{c2})$$
$$= -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14})$$
$$= 0.940$$

Entropy

E.g. suppose S has all positive or all negative

Metrics for performance evaluation

- Calculate accuracy using confusion matrix
- Accuracy is trues/everything

| | | PREDICTED CLASS | |
|---|---|---|---|
| | | Class=Yes | Class=No |
| ACTUAL CLASS | Class=Yes | a | b |
| | Class=No | c | d |

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

| | | PREDICTED CLASS | |
|---|---|---|---|
| | | Class=Yes | Class=No |
| ACTUAL CLASS | Class=Yes | a (TP) | b (FN) |
| | Class=No | c (FP) | d (TN) |

Most widely-used metric:

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Also:

Precision = tp/(tp + fp)

Sensitivity = tp/(tp+fn)

# WEEK 8 – OTHER CLASSIFIERS

Metrics

- Precision and recall

Precision is:

$$\frac{True\ Positives}{True\ Positives + False\ Positives}$$

Recall is:

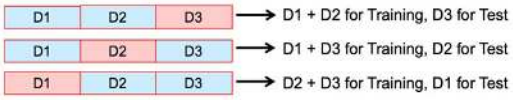
$$\frac{True\ Positives}{True\ Positives + False\ Negatives}$$

Overfitting/underfitting

- Overfitting
  - Model does not generalise well and is excessively complex
  - Performs well on training data but not on testing data
  - Low recall
  - To avoid
    - Pre-pruning: Stop growing tree earlier
    - Post-pruning: Allow tree to overfit and then post-prune
      - More effective in practice
      - Generally, remove rules if they have little effect on the error rate
- Underfitting
  - Model is too simple to give accurate labels
  - Performs poorly on training and testing data
  - Low precision

Cross validation

- K-fold cross-validation steps
  - Partition data into k disjoint subsets
  - Train on k-1 partitions and test remaining one




- Make sure that all folds have the same distribution of classes
- Stratification ensures that classes are properly represented across partitions
- Leave-one-out cross-validation
  - Each case is left out and the model is trained on all of the remaining instances

Bayesian classifiers

- Bayes' Theorem is conditional probability
- Naïve Bayes classifier
  - Robust to isolated noise points
  - Can adapt quickly to new data
  - Handles missing values by ignoring them
  - Robust to irrelevant attributes

Classifier model evaluation

- Confusion matrix and performance measures
- ROC (Receiver operating characteristics) for binary classifiers
  - Tradeoff between positive hits and false alarms
  - TPR on y axis and FPR on x axis

$$True\ Positive\ Rate:\ TPR = \frac{TP}{TP + FN}$$

$$False\ Positive\ Rate:\ TPR = \frac{FP}{FP + TN}$$

  - Where FPR is 0 on the x-axis is the perfect classification
  - On y = x is 50/50, no better than random
- AUC
  - Calculate area under ROC curve
- Lift charts
  - Measure of effectiveness of a predictive model
  - Lift factor = success rate with model/success rate without model
    - The probability of any chocolate bar chosen at random being "good" is 3/7.
    - If we just sample the top 2 (28%) we're most confident of then the probability of "good" is 2/2.

$$Lift = \frac{\left(\frac{2}{2}\right)}{\left(\frac{3}{7}\right)} = \frac{7}{3} = 2.33\ or\ 233\%$$

| Brand | Actual Class | Conf |
|---|---|---|
| Aero | 0 | 0.3 |
| Cherry Ripe | 0 | 0.4 |
| Kitkat | 0 | 0.4 |
| Bounty | 1 | 0.6 |
| Snickers | 0 | 0.7 |
| Flake | 1 | 0.8 |
| Violet Crumble | 1 | 1.0 |

    - For top 3 (42%), P(good) = 2/3

$$Lift = \frac{\left(\frac{2}{3}\right)}{\left(\frac{3}{7}\right)} = \frac{14}{9} = 1.56\ or\ 156\%$$

# WEEK 9 – ENSEMBLE METHODS AND ANN

## Ensemble methods

- Original training data – Create multiple data sets – Build multiple classifiers – Combine classifiers

- Works best when:
    - Individual classifiers are moderately accurate
    - Individual classifiers are not correlated
    - Decision trees work well as individual classifiers
- Bagging
    - Multiple replicates of original data by sampling, with replacing, from the training set
    - Combine the classifiers by taking a majority vote to produce the final version