CSCI446/946 Big Data Analytics

Week 5 – Lecture: Regression & Association Rules

School of Computing and Information Technology
University of Wollongong Australia

Spring 2024

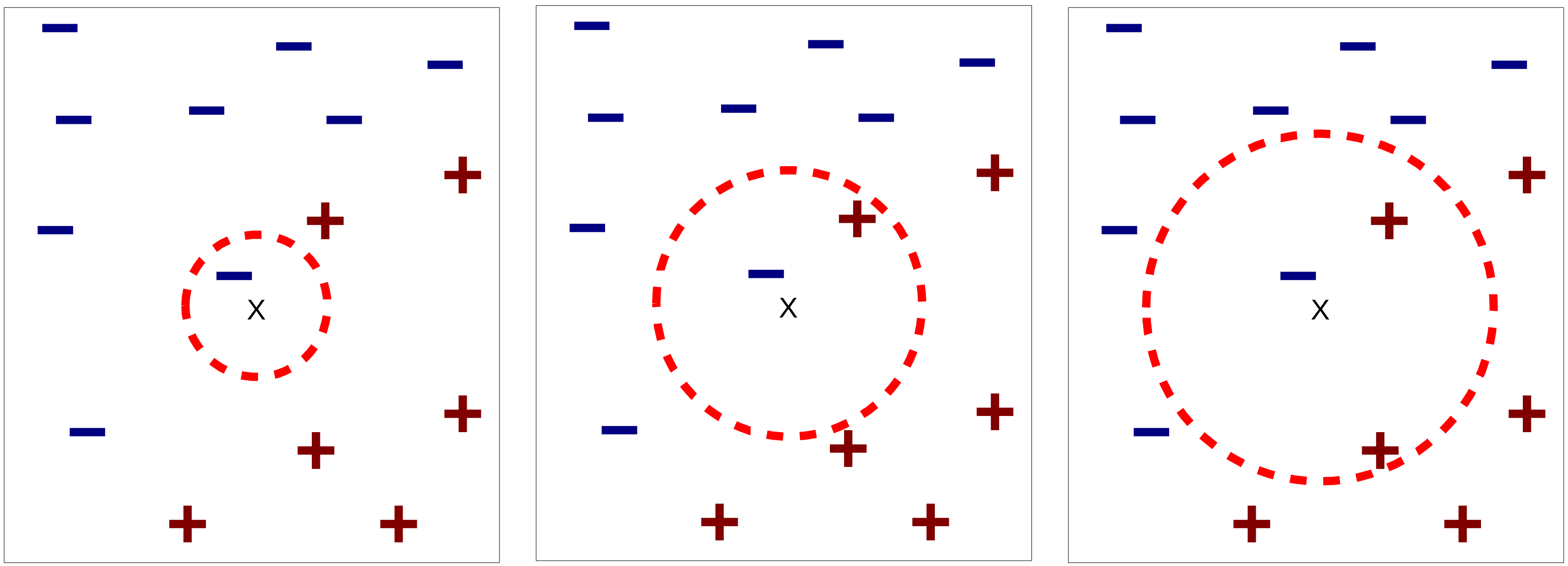
Content

- Brief Recap
  - Classification
  - Performance indicators
- Regression
  - Linear regression
  - Logistic regression
- Association Rules

Content

- Brief Recap
  - Classification
  - Performance indicators
- Regression
  - Linear regression
  - Logistic regression
- Association Rules

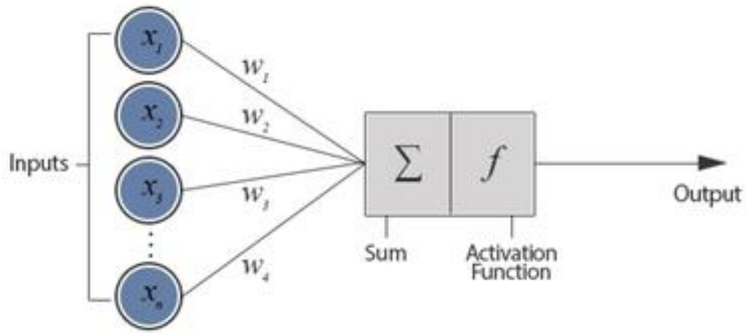
Nearest Neighbor Classifier (recap)

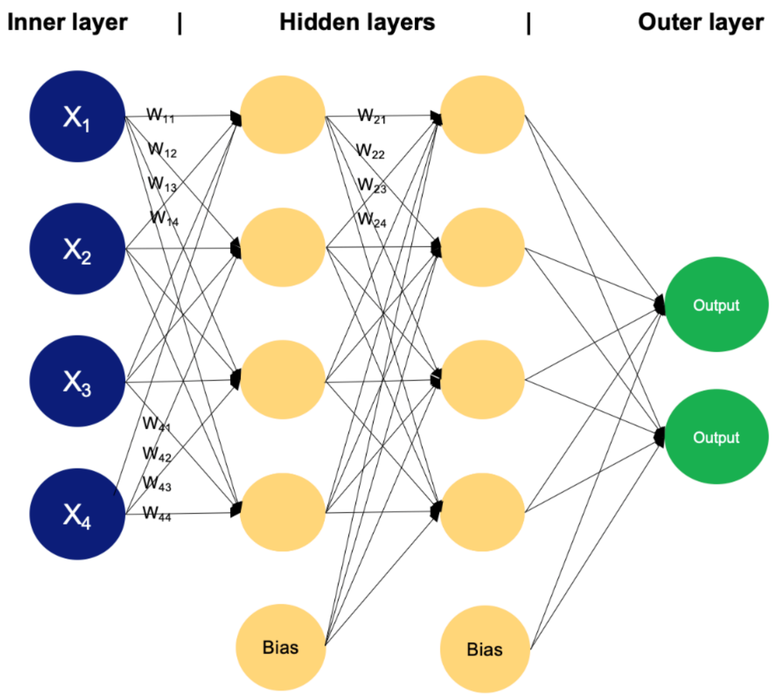


(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

K-nearest neighbors of a record x are data points that have the k smallest distance to x

Multi-Layer Perceptron (MLP) (recap)



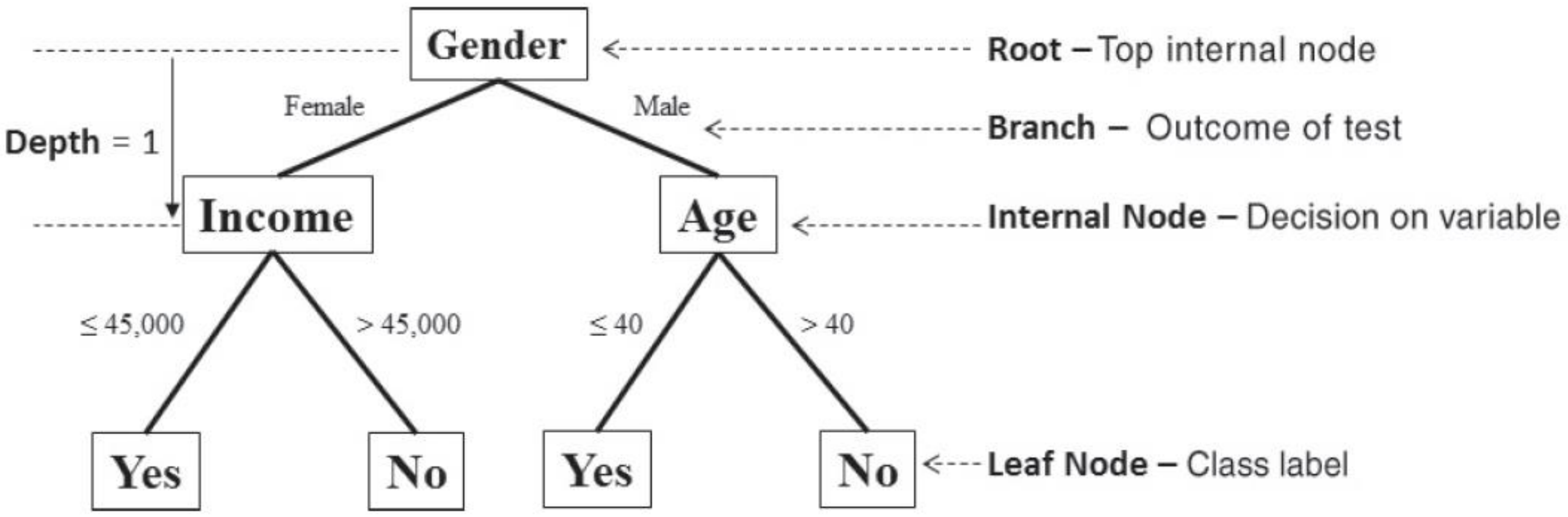


Decision Tree (recap)

- Each node tests a particular input variable
- Each branch represents the decision made

- Classifying a new observation is to traverse this decision tree.

$$InfoGain_A = H_S - H_{S|A}$$



Naïve Bayes Classifier (recap)

- An example
  - With the bank marketing dataset, use Naïve Bayes Classifier to predict if a client would subscribe to a term deposit
- Building a Naïve Bayes classifier requires to calculate some statistics from training dataset
  - $P(A|c_i)$ for each class i=1,2,...,n
  - $P(a_j|c_i)$ for each attribute j=1,2,...,m in each class

$$P(c_i|A) \propto P(c_i) \cdot \prod_{j=1}^{m} P(a_j|c_i) \qquad i = 1,2,\ldots n$$

Performance Indicators (recap)

A\P	C	¬C	
C	**TP**	**FN**	**P**
¬C	**FP**	**TN**	**N**
	P’	**N’**	**All**

- **Classifier Accuracy,** or recognition rate: percentage of test set tuples that are correctly classified

  Accuracy = (TP + TN)/All

- **Error rate:** *1 – accuracy*, or

  Error rate = (FP + FN)/All

- **Class Imbalance Problem**:
  - One class may be *rare*, e.g. fraud, or HIV-positive
  - Significant *majority of the negative class* and minority of the positive class
  - **Sensitivity**: True Positive recognition rate
    - **Sensitivity = TP/P**
  - **Specificity**: True Negative recognition rate
    - **Specificity = TN/N**

Performance Indicators (recap)

- **Precision**: exactness – what % that the classifier labeled as positive are actually positive

$$precision = \frac{TP}{TP + FP}$$

- **Recall:** completeness – what % of the positives did the classifier label as positive? (equals to sensitivity)

$$recall = \frac{TP}{TP + FN}$$

  - Perfect score is 1.0
  - In practice, inverse relationship between precision & recall

- ***F* measure (*F_1* or *F*-score)**: harmonic mean of precision and recall,

$$F = \frac{2 \times precision \times recall}{precision + recall}$$

Content

- Brief Recap
  - Classification
  - Performance indicators
- Regression
  - Linear regression
  - Logistic regression
- Association Rules

Regression

- Overview of Regression
- Linear Regression
- Logistic Regression
- Reasons to Choose and Cautions
- Additional Regression Models

All the figures, tables and codes are from the book "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data" unless indicated otherwise.

Regression

- Classification vs Regression predictive modelling problems.
  - Classification is the task of predicting a discrete class label.
  - Regression is the task of predicting a continuous quantity.
- Regression methods:
  - Linear Regression
  - Non-linear i.e. Logistic Regression
- What are used for regression analysis
  - Explain the influence that a set of variables has on the outcome of another variable of interest.
- Single layer MLPs with linear activation function are linear regression methods.

Linear Regression

- An analytical technique used to model the relationship between several input variables and a continuous outcome variable
- A key assumption
  - The relationship is linear (x_i to y)
- Non-deterministic nature
  - Accounts for the randomness in an outcome
  - Provides the expected value of the outcome

Use Cases

- Real estate
  - Home prices vs. {living area, number of bedrooms, school district rankings, crime statistics, etc.}
- Demand forecasting
  - Quantity of food that customers will consume vs. {weather, day of the week, discount, etc.}
- Medical
  - Effect of a treatment vs. {duration, dose, patient attributes, etc.}

Model Description

- Linear regression assumes
  - There is a linear relationship between the input variables and the outcome variable

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_{p-1} x_{p-1} + \varepsilon$$

where:

y is the outcome variable

x_j are the input variables, for $j = 1, 2, \ldots, p - 1$

β_0 is the value of y when each x_j equals zero

β_j is the change in y based on a unit change in x_j, for $j = 1, 2, \ldots, p - 1$

ε is a random error term that represents the difference in the linear model and a particular observed value for y

Model Description

- Key question
  - β_0 , β_1 ,..., β_{p-1} are the unknown model parameters.
  - How to obtain their values?

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_{p-1} x_{p-1} + \varepsilon$$

where:

y is the outcome variable

x_j are the input variables, for $j = 1, 2, \ldots, p - 1$

β_0 is the value of y when each x_j equals zero

β_j is the change in y based on a unit change in x_j, for $j = 1, 2, \ldots, p - 1$

ε is a random error term that represents the difference in the linear model and a particular observed value for y

Model Description

- Objective
  - The estimates of these unknown model parameters shall make the linear regression model provide a reasonable estimate of the outcome variable
  - In other words, they shall minimize the overall error between the following two:
    - The value predicted by the linear regression model
    - The actual observations collected

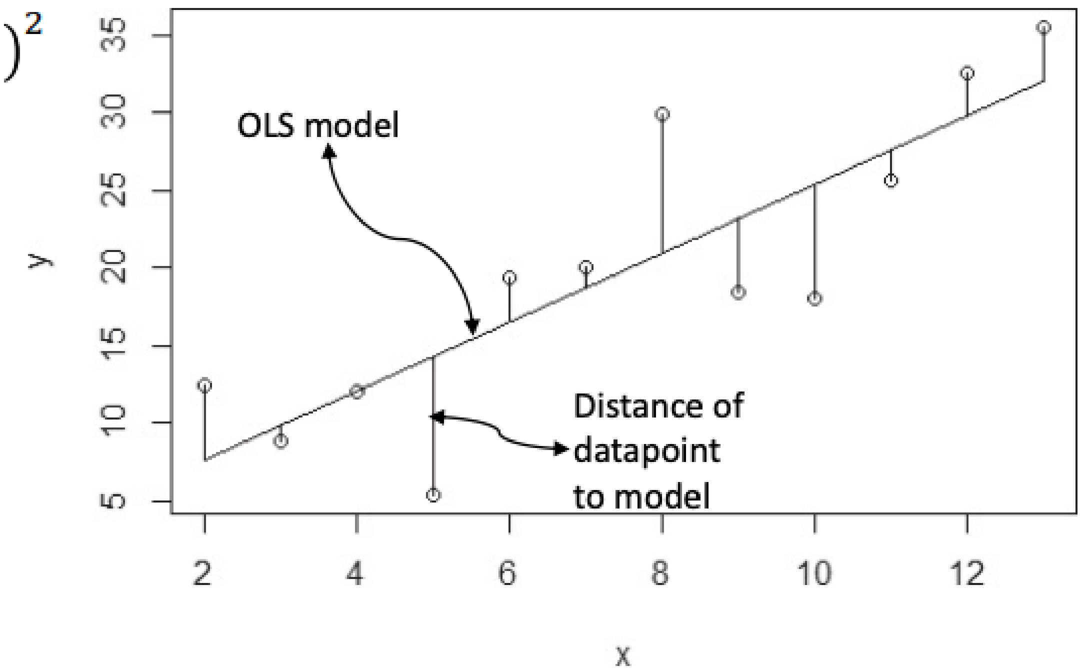
Model Description

- Ordinary Least Squares (OLS)
  - A common technique to estimate the parameters
  - Find the line best approximating the relationship

$$min \sum_{i=1}^{N} \left(y_i - (\beta_0 + \sum_{j=1}^{n} \beta_j x_{ij})\right)^2$$

For the example shown:

$$min \sum_{i=1}^{N} \left(y_i - (\beta_0 + \beta_1 x_i)\right)^2$$



Model Description

- OSL
  - Make no assumptions about the error term.
- Linear regression model
  - Making additional assumptions on top of the Ordinary Least Squares (OLS)
  - These additional assumptions provide further capabilities in utilising the linear regression model
  - These additional assumptions are almost always made

Model Description

- Linear regression model (with normally distributed errors)

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \ldots + \beta_{p-1} x_{p-1} + \varepsilon$$

where:

y is the outcome variable

x_j are the input variables, for j = 1, 2,. . ., p – 1

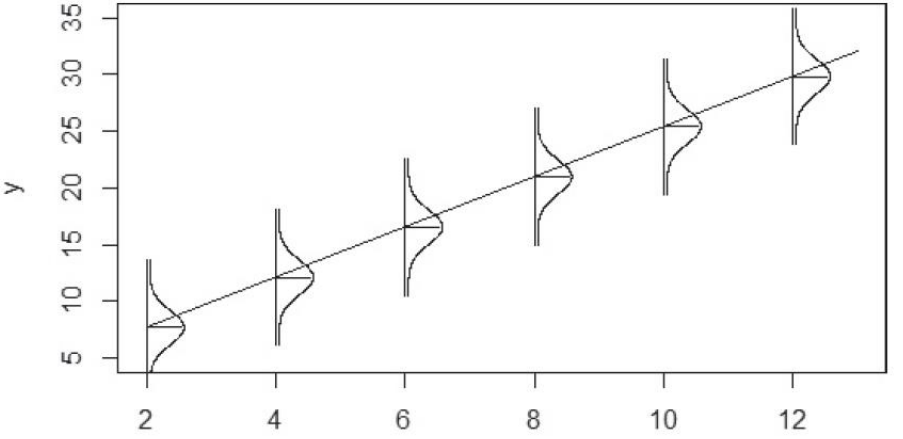
β_0 is the value of y when each x_j equals zero

β_j is the change in y based on a unit change in x_j, for j = 1, 2,. . ., p – 1

$\varepsilon \sim N(0, \sigma^2)$ **and the εs are independent of each other**

Model Description

- For given x_1, x_2, ... x_{p-1},



- So, the regression model estimates the expected value of y for the given value of x subject to a normal distributed error term.

Model Description

- An example

```
income_input = as.data.frame(  read.csv("c:/data/income.csv")   )
income_input[1:10,]

   ID Income Age Education Gender
1   1    113  69        12      1
2   2     91  52        18      0
3   3    121  65        14      0
4   4     81  58        12      0
5   5     68  31        16      1
6   6     92  51        15      1
7   7     75  53        15      0
8   8     76  56        13      0
9   9     56  42        15      1
10 10     53  33        11      1
```

Model Description

- The proposed linear regression model is

$$Income = \beta_0 + \beta_1 Age + \beta_2 Education + \beta_3 Gender + \varepsilon$$

- Implemented in R by `lm()` function

```
results <- lm(Income~Age + Education + Gender, income_input)
summary(results)
```

dataset

Model Description

```
results <- lm(Income~Age + Education + Gender, income_input)
summary(results)

Call:
lm(formula = Income ~ Age + Education + Gender, data = income_input)

Residuals:
    Min      1Q  Median      3Q     Max
-37.340  -8.101   0.139   7.885  37.271

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.26299    1.95575   3.714 0.000212 ***
Age          0.99520    0.02057  48.373  < 2e-16 ***
Education    1.75788    0.11581  15.179  < 2e-16 ***
Gender      -0.93433    0.62388  -1.498 0.134443
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.07 on 1496 degrees of freedom
Multiple R-squared:  0.6364,  Adjusted R-squared:  0.6357
F-statistic:   873 on 3 and 1496 DF,  p-value: < 2.2e-16
```

Model Description

- Hypothesis testing on coefficients
  - Coefficients are estimated based on the given observed sample only
  - There is some uncertainty for the estimates
  - `Std. Error` can be used to perform hypothesis testing to determine if each coefficient is statistically different from zero

$$H_0 : \beta_j = 0 \quad \text{versus} \quad H_A : \beta_j \neq 0$$

$$Income = \beta_0 + \beta_1 Age + \beta_2 Education + \beta_3 Gender + \varepsilon$$

Model Description

- Coefficients provide information on how influential an attribute is on the outcome.

- Hypothesis testing on coefficients
  - If a coefficient is NOT statistically different from zero, the coefficient and the associated variable in the model shall be excluded

Caution

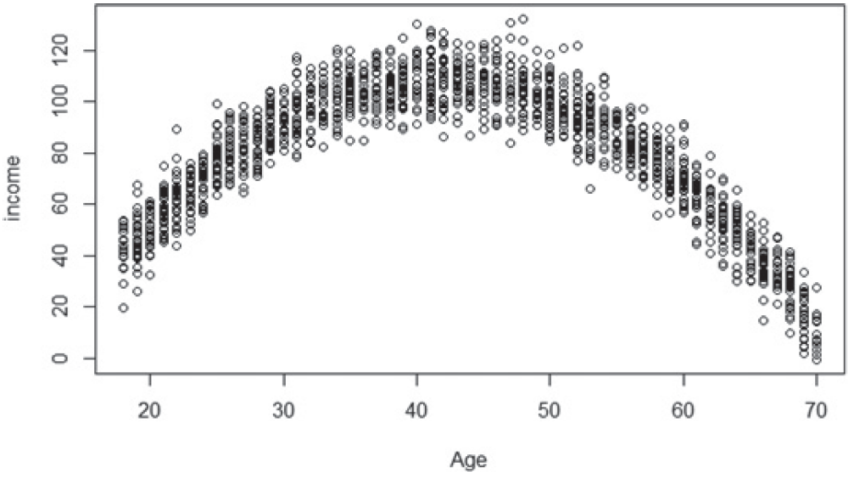
- Categorical variables
  - Gender, ZIP codes, nationality, …
  - An incorrect approach is to assign a number to each of them based on an alphabetical ordering
- A proper way
  - For a categorical variable can take m different values, we shall add m-1 binary variables to the regression model

Diagnostics

- Recall that linear regression models depend on assumptions
- We need to validate a fitted regression model
  - Evaluate the linearity assumption
  - Evaluate the residuals
  - Evaluate the normality assumption

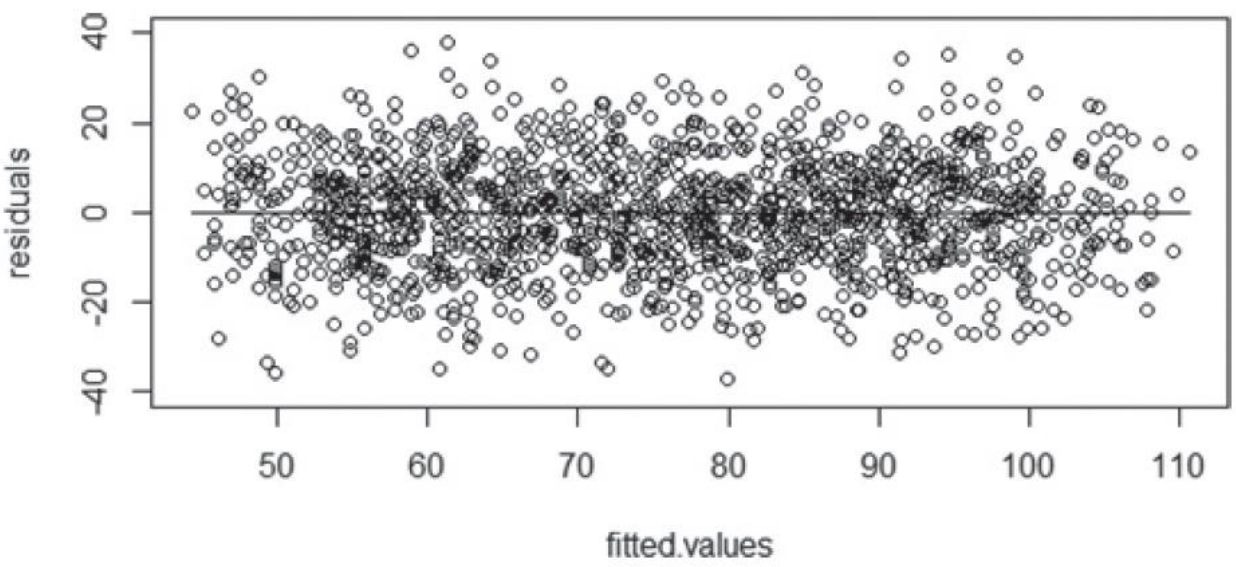
Diagnostics

- Evaluate the linearity assumption
  - Plot the outcome variable against each input variable
  - If not linear
    - Transform the outcome or input variables
    - Add extra input variables , e.g., age squared



Diagnostics

- Evaluate the Residuals (**Residual = Observed – Predicted**)
  - Recall $\varepsilon \sim N(0, \sigma^2)$ **and the εs are independent of each other**
  - If this assumption is violated, the various inferences are suspect

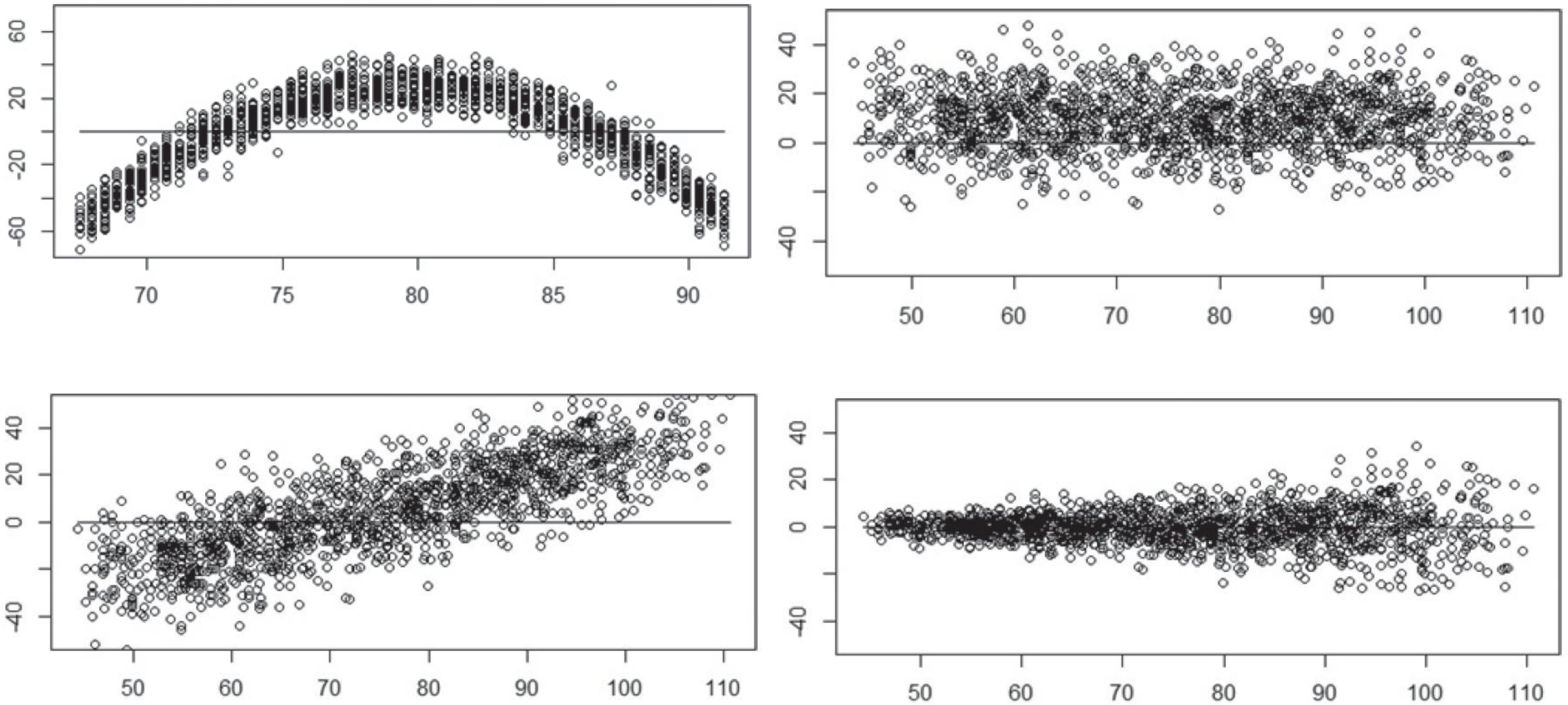


Residuals have zero mean and a constant variance

Diagnostics

- Evaluate the Residuals (zero mean and constant variance)

Transform, additional input variable

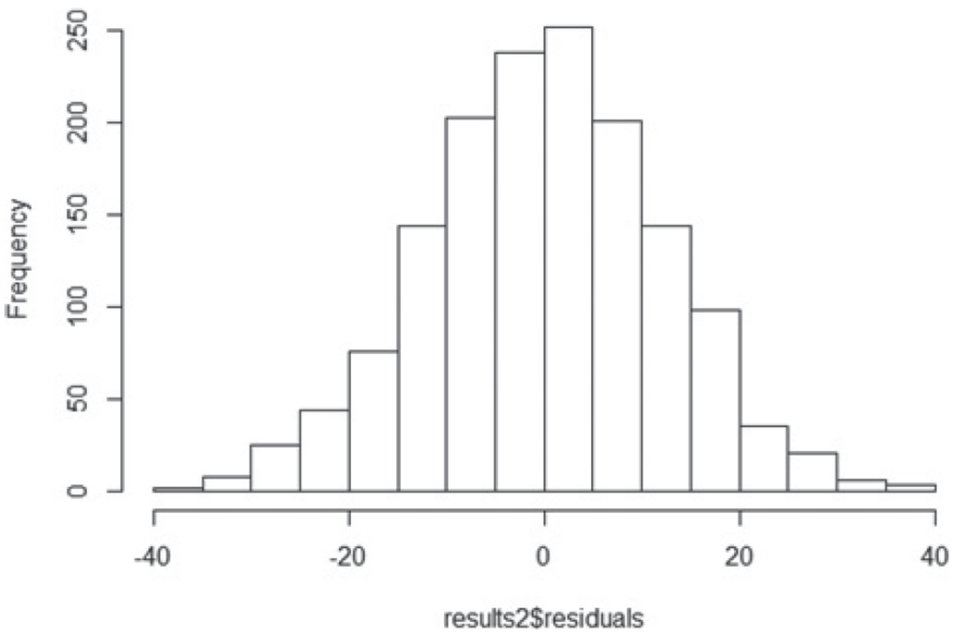


How about these residual plots?

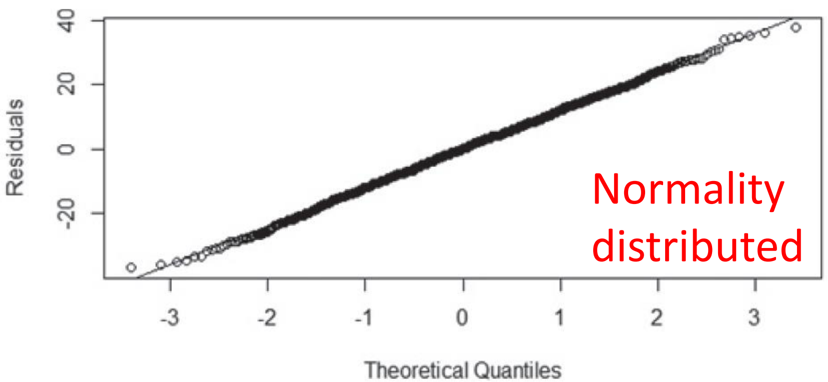
Diagnostics

- Evaluate the Residuals (normality assumption)

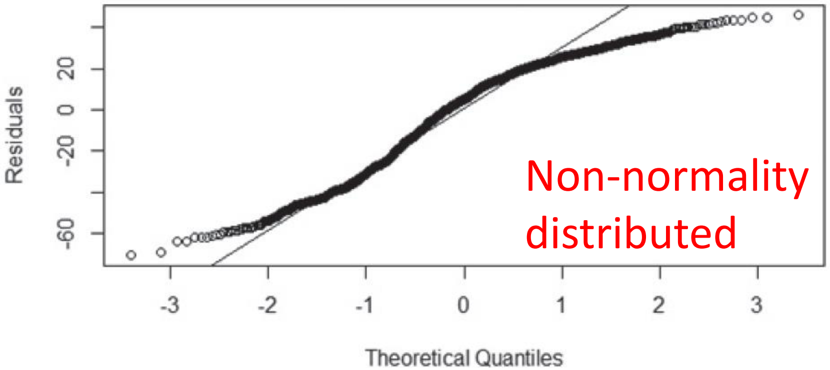
Option 1



Option 2



Normality distributed



Non-normality distributed

```
qqnorm(results2$residuals, ylab="Residuals", main="")
qqline(results2$residuals)
```

Diagnostics

- Other considerations
  - Consider all possible input variables early in the analytic process
  - Be careful when adding more variables
    - The R^2 value may decrease because of the increased input dimension.
  - Linear regression is sensitive to outliers
    - Examine any outliers, observed points that are markedly different from the majority of the points
  - Examine if the magnitudes and signs of the estimated parameters make sense

Logistic Regression

- In linear regression, the outcome variable is a continuous variable
- When the outcome variable is categorical in nature, logistic regression can be used
  - To predict the probability of an outcome based on the input variables

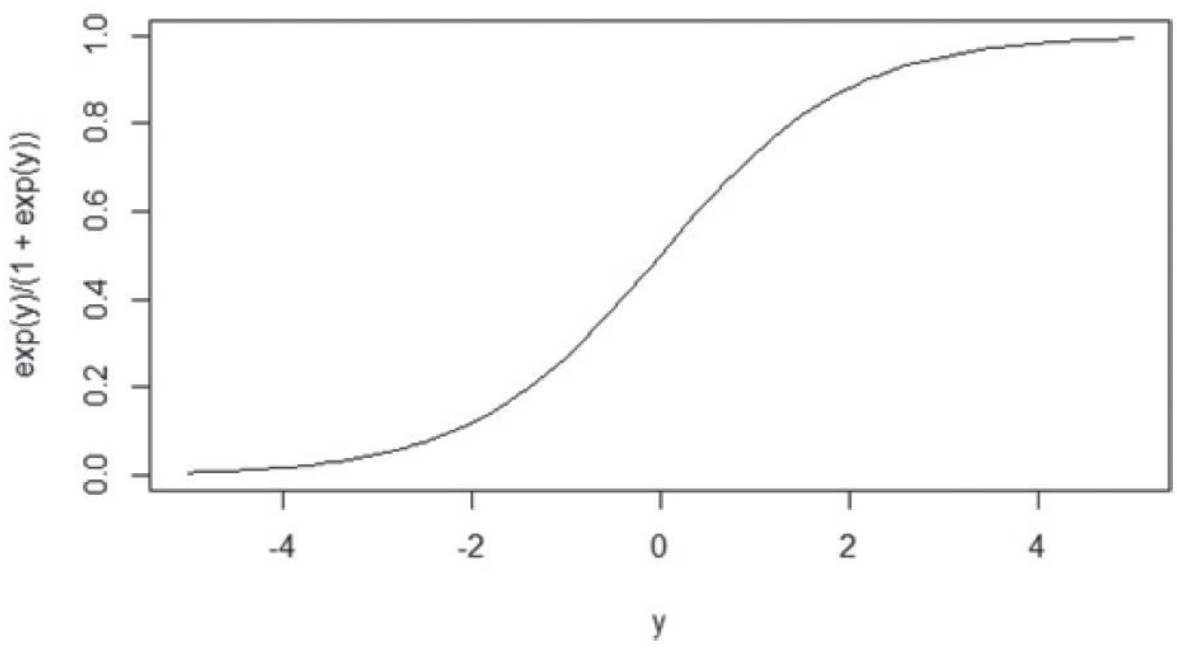
Logistic Regression

- Use Cases
  - Medical: determine the probability of a patient's response to a medical treatment
  - Finance: determine the probability that an applicant will default on the loan
  - Marketing: Determine the probability for a customer to switch carriers (churning)
  - Engineering: Determine the probability of a mechanical part to fail

Model Description

- Logistic function

$$f(y) = \frac{e^y}{1+e^y} \qquad \text{for } -\infty < y < \infty$$



Model Description

- In logistic regression, *y* is expressed as a linear function of the input variables (but y is not observed! Only f(y) is observed!)

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \ldots + \beta_{p-1} x_{p-1}$$

- The probability of an event is

$$P(C|x_1, x_2, \cdots, x_p) = f(y) = \frac{e^y}{1+e^y} \quad for \; -\infty < y < \infty$$

$$P = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_{p-1} x_{p-1}}}{1 + e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_{p-1} x_{p-1}}}$$

Model Description

- Rewriting the equation can give us the log odd ratio (the logit of P)

$$\ln\left(\frac{P}{1-P}\right) = y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_{p-1}$$

- Maximum Likelihood Estimation (MLE) is often used to estimate the model parameters
  - It finds the parameter values that maximize the chances of observing the given dataset

Customer Churn Example

- Input variables: Age (years), Married (true/false), Duration (years), Churned_contacts (count)
- Outcome variable: Churned (true/false)

$$y = 3.50 - 0.16 * Age + 0.38 * Churned\_contacts$$

Customer	Age (Years)	Churned_Contacts	y	Prob. of Churning
1	50	1	−4.12	0.016
2	50	3	−3.36	0.034
3	50	6	−2.22	0.098
4	30	1	−0.92	0.285
5	30	3	−0.16	0.460
6	30	6	0.98	0.727
7	20	1	0.68	0.664
8	20	3	1.44	0.808
9	20	6	2.58	0.930

ROC Curve

- Logistic regression is often used as a classifier to assign class labels to a data example
  - Based on the predicted probability
- Commonly, 0.5 is used as the default probability threshold
- However, any threshold value can be used depending on the preference to avoid false positives

Diagnosis (review)

- **True Positive (TP):** model predicts C, when actually C
- **True Negative (TN):** model predicts ¬ C, when actually ¬ C
- **False Positive (FP):** model predicts C, when actually ¬ C
- **False Negative (FN):** model predicts ¬ C, when actually C

$$\text{Accuracy (ACC)} = \frac{\#\text{TP} + \#\text{TN}}{\#\text{TP} + \#\text{FP} + \#\text{TN} + \#\text{FN}}$$

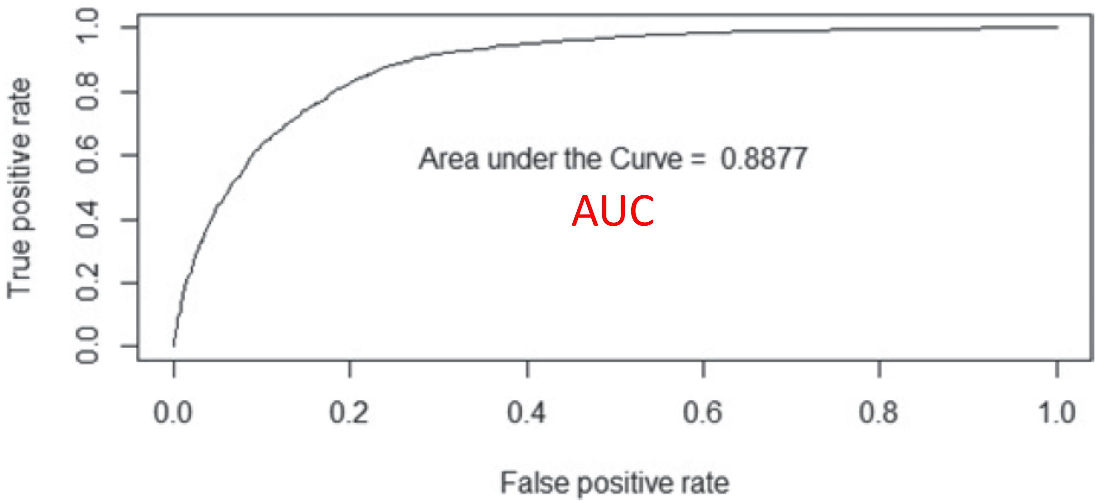
$$\text{False Positive Rate (FPR)} = \frac{\#\text{FP}}{\#\text{TN} + \#\text{FN}}$$

$$\text{True Positive Rate (TPR)} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FP}}$$

#="number of"

ROC Curve

- Receiver Operating Characteristic (ROC) curve
  - The plot of the True Positive Rate (TPR) against the False Positive Rate (FPR)
  - A classifier shall have a low FPR and a high TPR
  - A metric: the area under the ROC curve (AUC)



Reasons to Choose and Cautions

- Linear regression
  - Input variables are continuous or discrete
  - Outcome variable is continuous
- Logistic regression
  - A better choice if outcome variable is categorical
- Both models assume a linear additive function of the input variables

Reasons to Choose and Cautions

- Correlation does not imply causation
  - We shall NOT infer that the input variables directly cause an outcome
- Generalization issue
  - Use caution when applying an already fitted model to data that falls outside the dataset used to train the model
- Multicolinearity issue
  - Ridge regression and Lasso regression

Summary

- Linear regression and logistic regression
  - Model observed data to predict future outcomes
- Care must be taken in performing and interpreting a regression analysis
  - Determine the best input variables and their relationship to outcome variables
  - Understand and validate underlying assumptions
  - Transform variables when necessary

Content

- Brief Recap
  - Classification
  - Performance indicators
- Regression
  - Linear regression
  - Logistic regression
- Association Rules

Association Rules

- Overview of Association Rules
- Apriori Algorithm
- Evaluation of Candidate Rules
- An example of rule generation
- Validation and Testing
- Diagnostics

All the figures, tables and codes are from the book "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data" unless indicated otherwise.

Association Rules

- Association rule discovery:
  - An unsupervised learning method
  - Descriptive, not predictive
  - Discover interesting, hidden relationship
    - Represented as rules or frequent itemsets
  - Commonly used for mining transactions in databases

All the figures, tables and codes are from the book "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data" unless indicated otherwise.

Association Rules

- It can usually answer the questions like
  - Which products tend to be purchased together?
  - Of those customers who are similar to this person, what products do they tend to buy?
  - Of those customers who have purchased this product, what other products do they tend to view or purchase?

Overview of Association Rules

- Each transaction consists of one or more items
- What items are frequently purchased together
- Goal: discover "interesting" relationships among the items

Overview of Association Rules

- Uncovered rule is in the form X → Y
  - meaning that when item X is observed, item Y is also observed
  - X: left-hand side (lhs); Y: right-hand side (rhs)
  - What does "Cereal → Milk (90%)" mean?
    - When cereal is purchased, 90% of the time milk is also purchased.

Overview of Association Rules

- Also known as "market basket analysis"
  - Each transaction – shopping basket
- Itemset
  - A collection of items or individual entities that contain some kind of relationship
- k-itemset
  - An itemset containing k items
  - $\{item_1, item_2, \ldots, item_k\}$

Overview of Association Rules

- How to discover relationships between items?
  - Exhaustively check all possible itemsets?
    - No! The size is exponentially large…
- Apriori algorithm
  - One of the earliest and the most fundamental algorithms for generating association rules
- Key concept: support
  - For pruning itemsets and controlling the exponential growth of candidate itemsets

Overview of Association Rules

- Support
  - Given an itemset X, the support of X is the percentage of transactions that contain X
  - Denoted by support(X)
- Frequent itemset
  - Contains items that appear together often enough
  - Formally, its support >= a minimum support

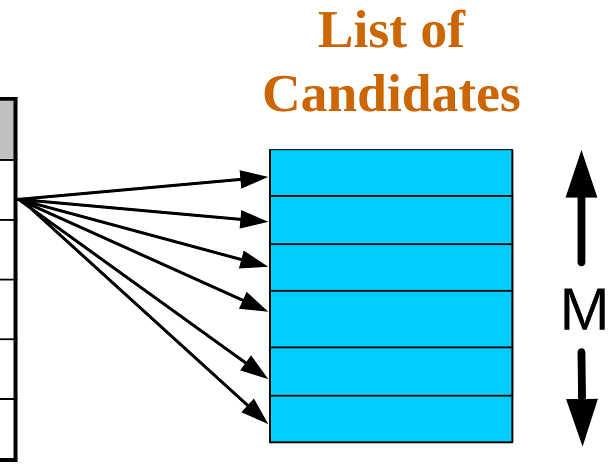
Frequent Itemset Generation

- Brute-force approach:
  - Each itemset in the lattice is a candidate frequent itemset
  - Count the support of each candidate by scanning the database

Transactions

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

N (rows), w (width)



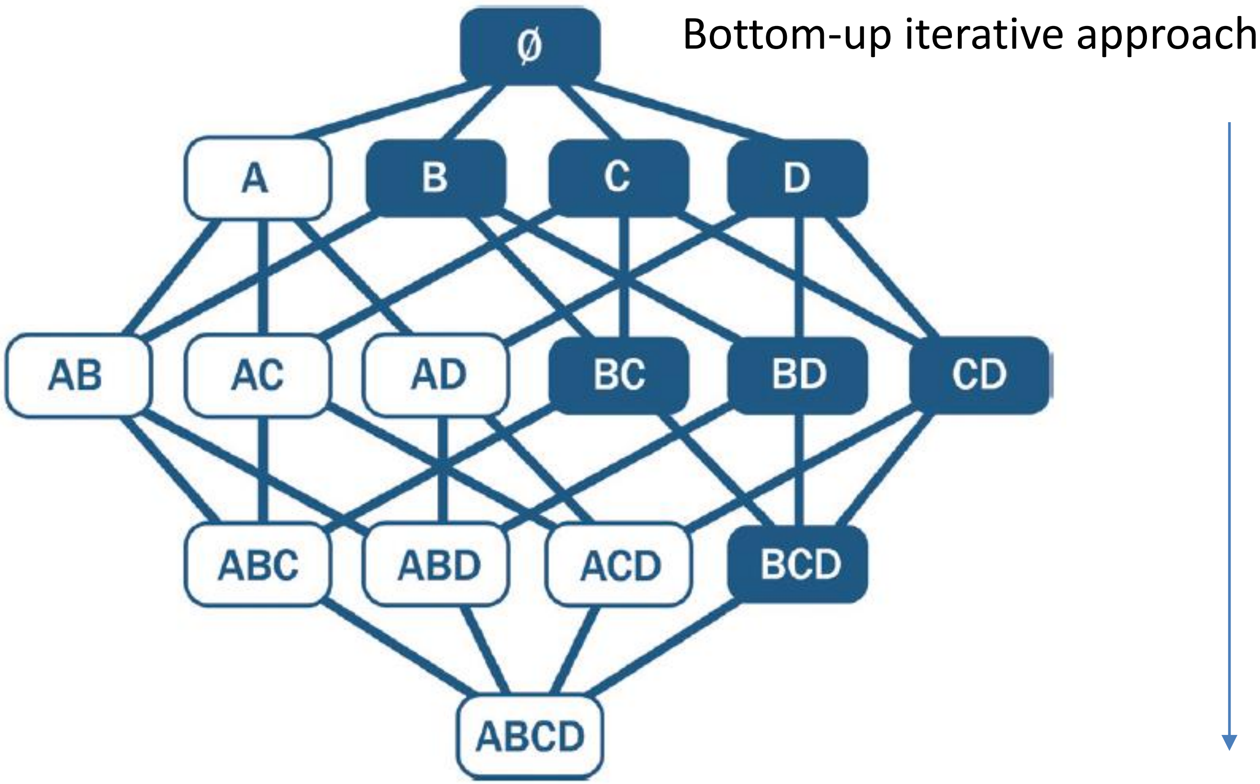
  - Match each transaction against every candidate
  - Complexity ~ O(NMw) => Expensive since $M = 2^d$!!!

Overview of Association Rules

- Apriori property (downward closure property)
  - If an itemset is frequent, then any subset of this itemset must also be frequent
  - It provides the basis for the Apriori algorithm
- An example: If support({bread, jam}) = 0.6 ➔ support({bread}}) >= 0.6 and support({jam}}) >= 0.6
- Therefore, if X is infrequent then all supersets that contain X must also be infrequent.

Overview of Association Rules

- Apriori property (downward closure property)



Itemset {A, B, C, D} and its subsets

Apriori Algorithm

- It takes a bottom-up iterative approach to uncovering frequent itemsets
  - First, identify all frequent items (or 1-itemsets)
  - The identified frequent 1-itemsets are paired into 2-itemsets to identify frequent 2-itemsets
  - Grow the size of identified frequent itemsets and identify again
  - Repeat this process until 1) it runs out of support or 2) the itemsets reach a predefined length

Apriori Algorithm

Input

- A transaction database D
- A minimum support threshold δ
- An optional parameter N indicating the maximum length an itemset could reach

```
Apriori (D, δ, N)
  k ← 1
  L_k ← {1-itemsets that satisfy minimum support δ}
  while L_k ≠ ∅
    if ∄N ∨ (∃N ∧ k < N)
      C_{k+1} ← candidate itemsets generated from L_k
      for each transaction t in database D do
        increment the counts of C_{k+1} contained in t
      L_{k+1} ← candidates in C_{k+1} that satisfy minimum support δ
      k ← k + 1
  return ⋃_k L_k
```

Apriori Algorithm

- Output of the Apriori algorithm
  - The collection of all the frequent k-itemsets
- A collection of candidate rules is formed based on the frequent itemsets uncovered
  - {milk, eggs} may suggest candidate rules
    - {milk} → {eggs} and {eggs} → {milk}
- Implemented by `apriori()` function in R

```
itemsets <- apriori(Groceries, parameter=list(minlen=1, support=0.02,
                                          target="frequent itemsets"))
```

Illustrating Apriori Principle

Item	Count
Bread	**4**
Coke	**2**
Milk	**4**
Beer	**3**
Diaper	**4**
Eggs	**1**

Items (1-itemsets)

Itemset	Count
{Bread,Milk}	**3**
{Bread,Beer}	**2**
{Bread,Diaper}	**3**
{Milk,Beer}	**2**
{Milk,Diaper}	**3**
{Beer,Diaper}	**3**

Pairs (2-itemsets)

(No need to generate candidates involving Coke or Eggs)

Triplets (3-itemsets)

Itemset	Count
{Bread,Milk,Diaper}	**3**

Minimum Support = 3

If every subset is considered,
$^6C_1 + {}^6C_2 + {}^6C_3 = 41$
With support-based pruning,
$6 + 6 + 1 = 13$

Rule Generation

- Given a frequent itemset L, find all non-empty subsets $f \subset L$ such that $f \rightarrow L - f$ satisfies the minimum confidence requirement
  - If {A,B,C,D} is a frequent itemset, candidate rules:

    $ABC \rightarrow D$, $ABD \rightarrow C$, $ACD \rightarrow B$, $BCD \rightarrow A$,
    $A \rightarrow BCD$, $B \rightarrow ACD$, $C \rightarrow ABD$, $D \rightarrow ABC$
    $AB \rightarrow CD$, $AC \rightarrow BD$, $AD \rightarrow BC$, $BC \rightarrow AD$,
    $BD \rightarrow AC$, $CD \rightarrow AB$,

- If $|L| = k$, then there are $2^k - 2$ candidate association rules (ignoring $L \rightarrow \emptyset$ and $\emptyset \rightarrow L$)

Evaluation of Candidate Rules

- How to evaluate the appropriateness of these candidate rules?
  - Many measures!
  - Measure: Confidence, lift, and leverage
- Confidence
  - The measure of certainty or trustworthiness associated with each rule

$$Confidence(X \rightarrow Y) = \frac{Support(X \wedge Y)}{Support(X)}$$

Evaluation of Candidate Rules

- Minimum Confidence
  - A predefined threshold to indicate a relationship is "interesting"
  - A higher confidence could indicates that the rule (X→Y) is more interesting (be careful…)
  - All the rules can be ranked based on support or confidence

$$Confidence(X \rightarrow Y) = \frac{Support(X \wedge Y)}{Support(X)}$$

Evaluation of Candidate Rules

- Issue with "Confidence"
  - In what cases, we will obtain high confidence?
  - Confidence does NOT consider the rule (Y)!
  - It cannot tell
    - if a rule contains true implication of the relationship
    - If the rule is purely coincidental

$$Confidence(X \rightarrow Y) = \frac{Support(X \wedge Y)}{Support(X)}$$

Evaluation of Candidate Rules

- Lift
  - Measures how many times more often X and Y occur together than expected if they are statistically independent of each other
  - Measures how X and Y are really related rather than coincidentally happening together

$$Lift(X \rightarrow Y) = \frac{Support(X \wedge Y)}{Support(X) * Support(Y)}$$

Evaluation of Candidate Rules

- Lift
  - Lift is 1 if X and Y are statistically independent of each other
  - A lift of X → Y greater than 1 indicates some usefulness of the rule
  - A larger lift suggests a greater strength of the association between X and Y

$$Lift(X \rightarrow Y) = \frac{Support(X \wedge Y)}{Support(X) * Support(Y)}$$

Evaluation of Candidate Rules

- Leverage (Pitetsky-Shapiro's)
  - Measures the difference in the probability of X and Y appearing together compared to what would be expected if X and Y were statistically independent of each other

$$Lift(X \rightarrow Y) = \frac{Support(X \wedge Y)}{Support(X) * Support(Y)}$$

$$Leverage(X \rightarrow Y) = Support(X \wedge Y) - Support(X) * Support(Y)$$

Evaluation of Candidate Rules

- Leverage
  - Its value will be zero when X and Y are statistically independent of each other
  - If X and Y have some kind of relationship, the leverage would be greater than zero.

$$Lift(X \rightarrow Y) = \frac{Support(X \wedge Y)}{Support(X) * Support(Y)}$$

$$Leverage(X \rightarrow Y) = Support(X \wedge Y) - Support(X) * Support(Y)$$

Evaluation of Candidate Rules

- Four measures
  - Support, Confidence, Lift, and Leverage
  - A high-confidence rule can sometimes be misleading
  - Lift and leverage not only ensure interesting rules but also filter out coincidental rules

$$Confidence(X \rightarrow Y) = \frac{Support(X \wedge Y)}{Support(X)} \quad Lift(X \rightarrow Y) = \frac{Support(X \wedge Y)}{Support(X) * Support(Y)}$$

$$Leverage(X \rightarrow Y) = Support(X \wedge Y) - Support(X) * Support(Y)$$

Evaluation of Candidate Rules

- Combination of Measures
  - Measures are often used in combination.
  - Example: Find all rules with a minimum level of confidence then, of those rules, sort rules in descending order by lift or leverage.
- Problem: These measures do not reflect novelty of rules i.e. differentiate between known rules and rules that are new to an observer.
  - Novelty and value of rules need to be evaluated by a human observer.

Applications of Association Rules

- Market basket analysis
  - Better merchandising, Placement of products, and Promotion plan
- Recommender system
  - Discover related products or similar customers
- Clickstream analysis
  - Analyse data of web browsing and use clicks
- Much more…

Validation and Testing

- Uninteresting rules
  - Involve mutually independent items
  - Cover few transactions
- Some rules could be purely coincidental
  - If 95% of customers buy X and 90% of them buy Y, then X and Y would occur together at least 85% of the time, even if there is no relationship between them
- Subjective criteria
  - Rules don’t reveal unexpected profitable actions

Diagnostics

- Measures like confidence, lift, and leverage shall be used along with human insights
- Properly specify the minimum support
- Apriori algorithm can be computationally expensive!
  - Various methods to improve Apriori's efficiency

Association Rules - Summary

- Apriori Algorithm
  - Unsupervised analysis technique
  - Uncovers relationships among items
- A wide range of applications
- Several measures to help validation
- Interesting rules
  - Do not seem obvious
  - Provide valuable insights

Q&A

Images Courtesy of Google Image