Universiti Malaysia PAHANG
Engineering • Technology • Creativity

CHAPTER 3
Predictive Modelling

DATA MINING (BSD3533)
DR. KU MUHAMMAD NA’IM KU KHALIF

Content

Chapter 3.1: Regression

Chapter 3.1.1: About Regression

Chapter 3.1.2: Multiple Linear Regression

Chapter 3.2: Classification

Chapter 3.2.1: About Classification

Chapter 3.2.2: Decision Tree

Chapter 3.2.3: Naïve Bayes

Chapter 3.3: Evaluation Metrics for Predictive Modelling

Chapter 3.3.1: Performance Measure/ Score

Chapter 3.3.2: Confusion Matrix

Chapter 3.3.3: ROC Curve

Chapter 3.3.4: Cross Validation

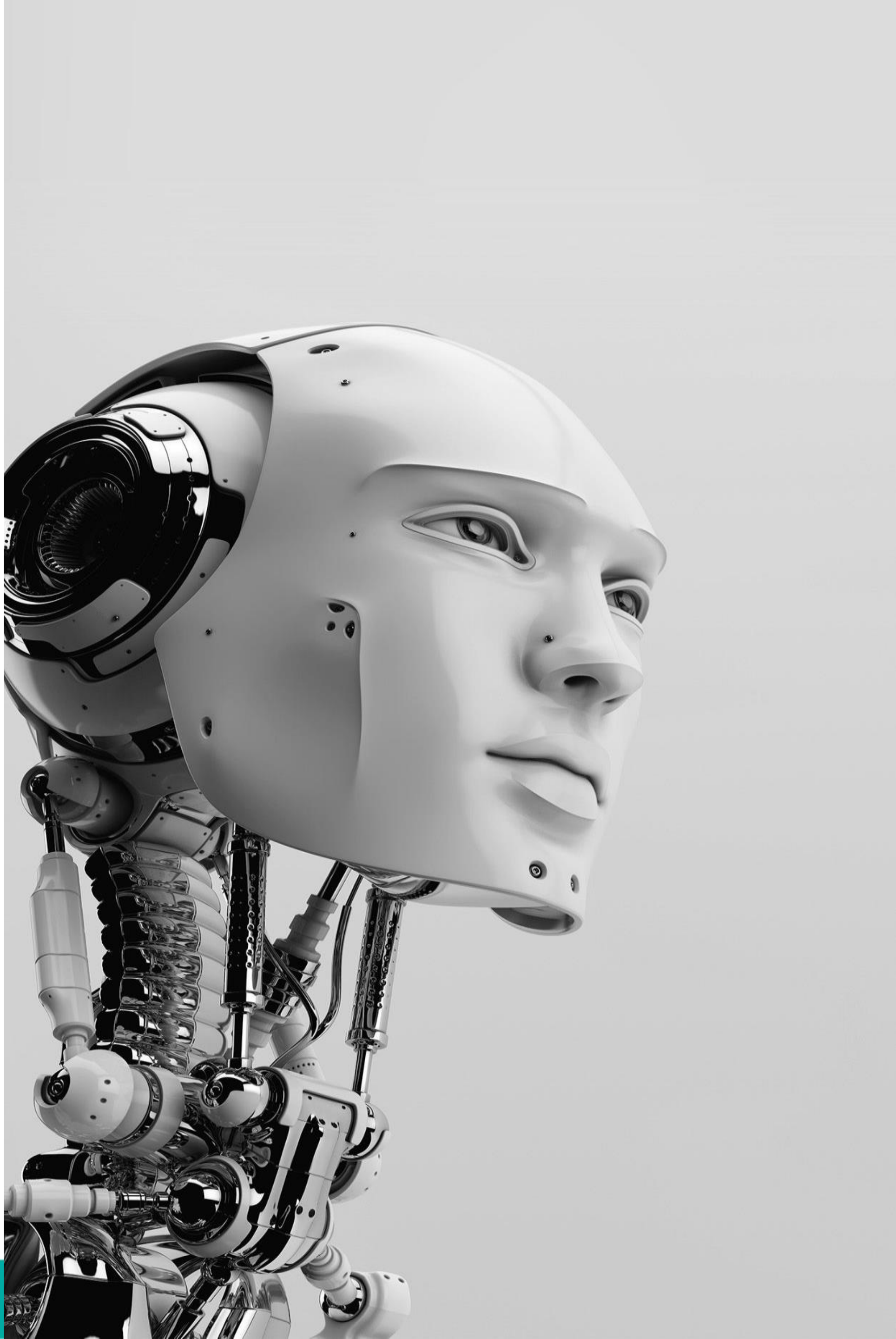
Predictive Modelling

- Predictive modelling is based on predicting the outcome of an event.
- It is designed on a pattern similar to the human learning experience in using observations to form a model of the important characteristics of some task.
- It is developed using a supervised learning approach, where we have some labelled data and we use this data to predict the outcome of unknown instances. It can be of two types, i.e., regression or classification.
- Some of the applications of predictive modelling are: predicting the outcome of an event, predicting the sale price of a property, predicting the placement of students, predicting the score of any team during a football match and so on.

Chapter 3.1: Regression

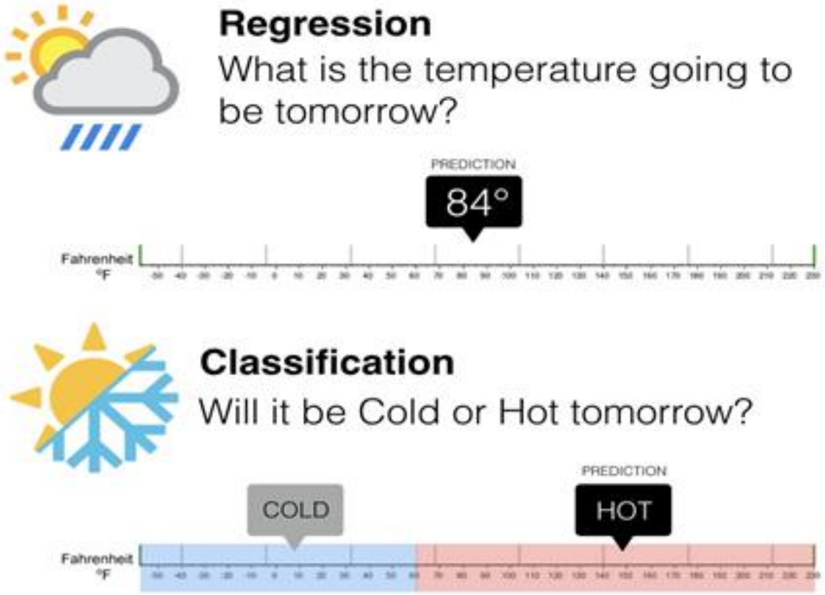
By the end of this topic, you should be able to:

- understand the concepts of regression analysis in data mining process.
- understand the applications of regression analysis in data science problems.

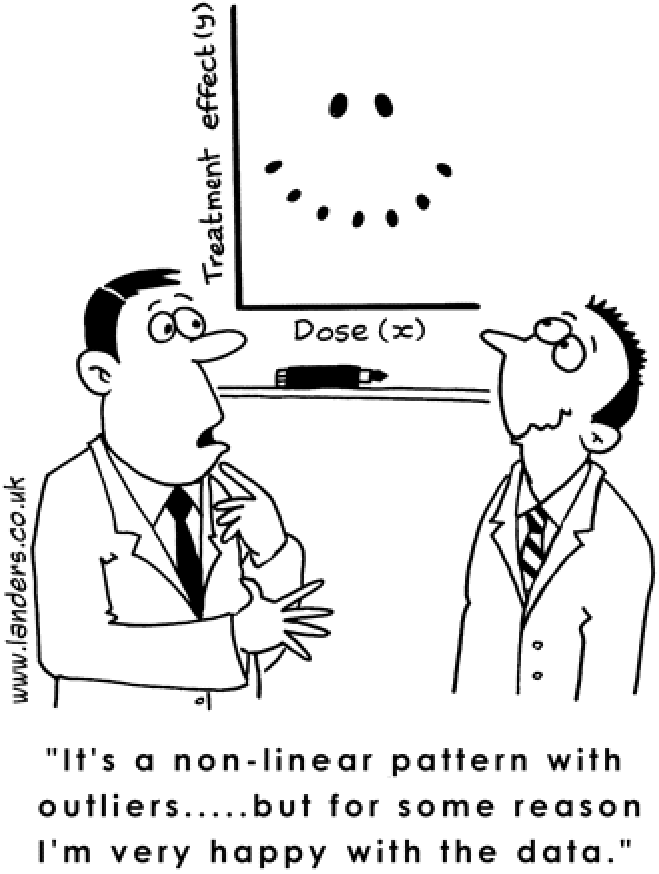


About Regression

- Regressions are one of the oldest self-learning methods used for predictive analytics, either to predict numerical values (linear and polynomial regression) or nominal classes (logistic regression).



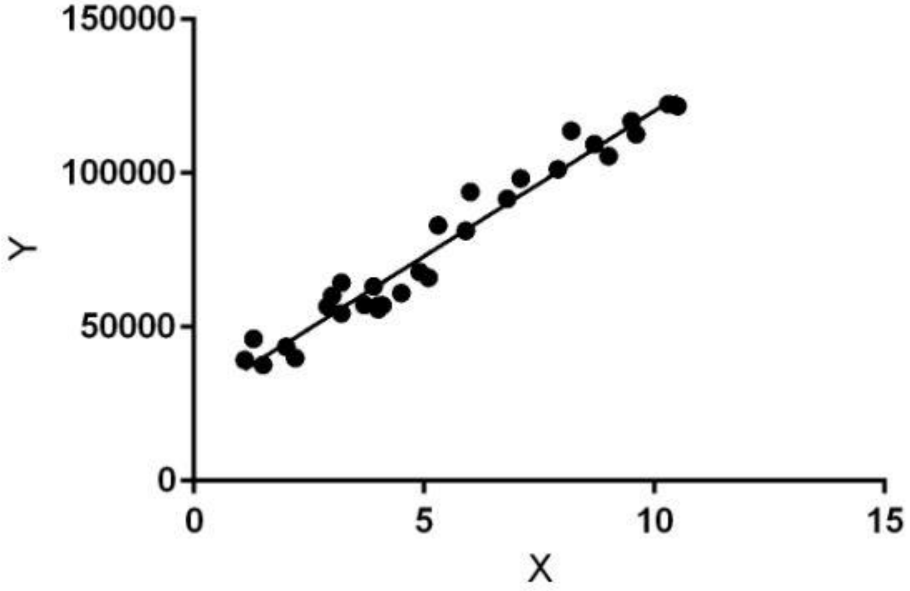
- Regression is a statistical measure that attempts to determine the strength of relationship between dependent and independent variable.
- Regression method describes how one variable depends on another.
- The latest development in regression algorithms consists of ensemble methods, such as regression trees, where a number of different regression models are trained to work together to predict the task at hand.


Treatment effect (y)
Dose (x)
www.landers.co.uk
"It's a non-linear pattern with outliers.....but for some reason I'm very happy with the data."

Linear Regression Analysis

- Linear regression analysis is used to predict the value of a variable based on the value of another variable.
- Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting.
- Different regression models differ based on – the kind of relationship between dependent and independent variables, they are considering and the number of independent variables being used.
- Linear approximate of a relationship between two or more variables.
- Model mathematically the relationship between two or more variables (dependent & independent variables).

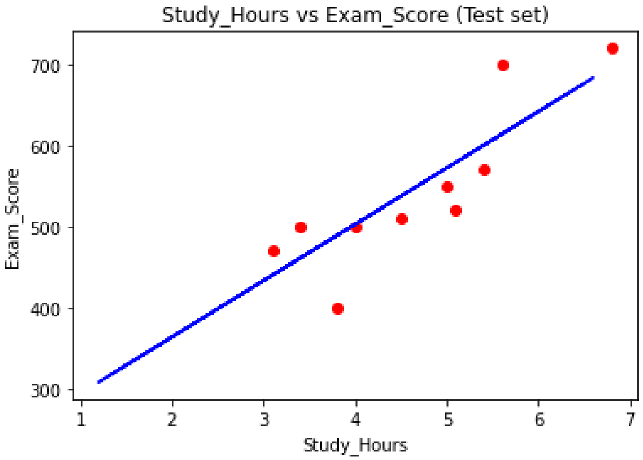
- The output response are numerical values.
- Simple linear regression model is a model with a single independent variable x that has a relationship with a response variable y and it can be represented by an equation of a straight line or known as line of best fit.
- If we have more than one independent variables then it becomes multiple linear regression.



- Linear regression performs the task of predicting a dependent variable value (y) based on a given independent variable (x). So, this regression technique finds out a linear relationship between x (input) and y(output). Hence, the name is Linear Regression.
- In the figure, X (input) is the study hour and Y (output) is exam score of students. The regression line is the best fit line for our model.
- Ordinary Least Square - Most common method to estimate the linear regression equation.

$$y_i = b_0 + b_1 x_i$$

y = dependent variable (output)
x_i = independent variable (predictor)
b_1 = coefficient of x_i
b_0 = constant/intercept



Linear Regression Applications

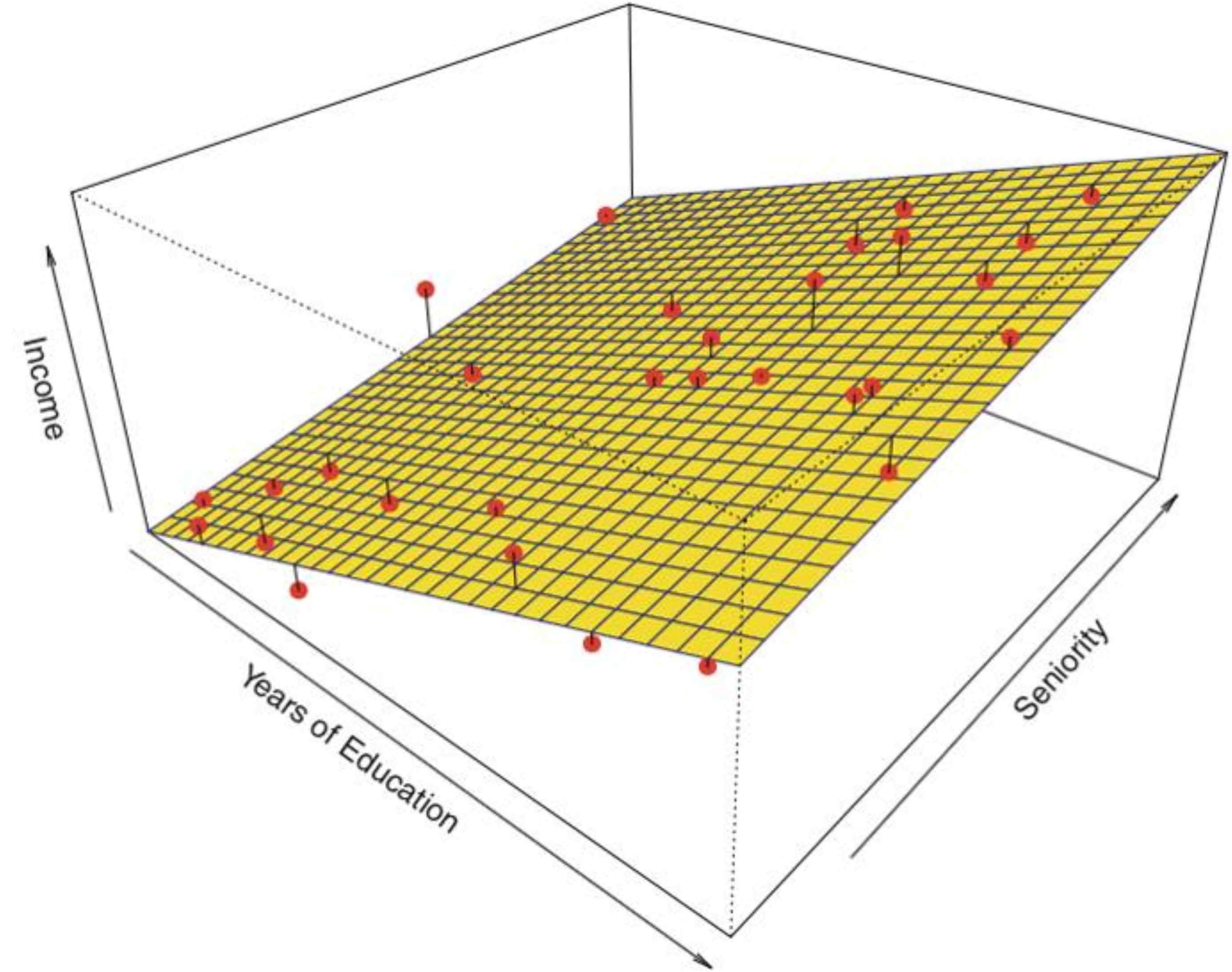
- Linear regression can be used to predict the sale of products in the future based on past buying behavior.
- Economists use linear regression to predict the economic growth of a country or state.
- Sports analysts use linear regression to predict the number of runs or goals a player would score in the coming matches based on previous performances.
- An organization can use linear regression to figure out how much they would pay to a new joiner based on the years of experience.
- Linear regression analysis can help a builder to predict how much houses it would sell in the coming months and at what price.

Multiple Linear Regression

Why do we use multiple linear regression?

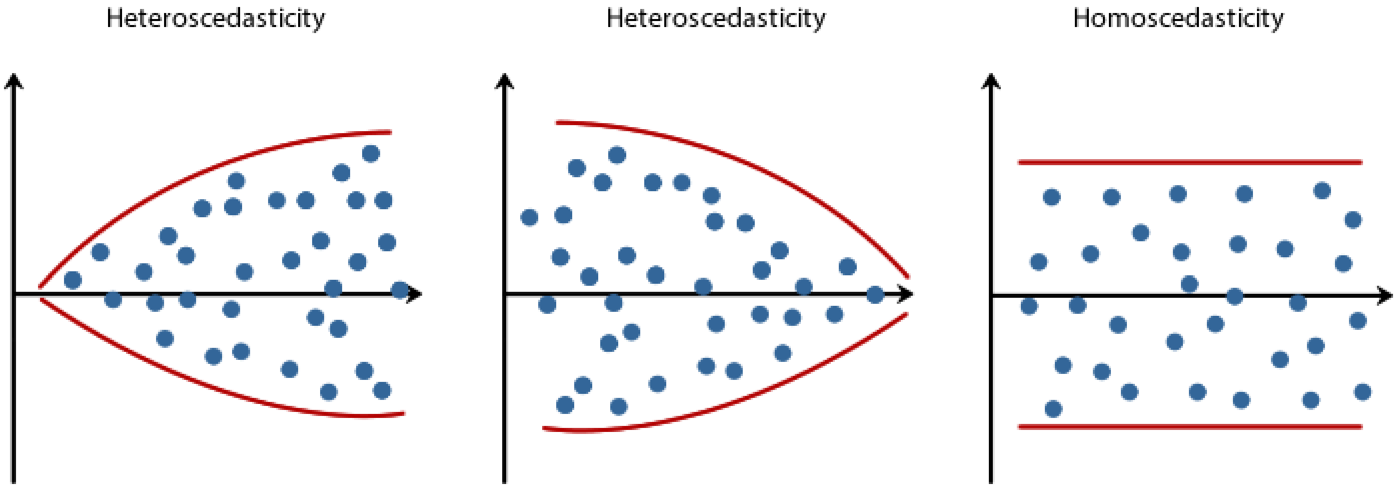
- Multiple linear regression is the most prevalent and influential form of regression analysis, and it is used to predict the result of a variable using two or more independent (or explanatory) variables.
- Why? Multiple regression allows us to determine the relative impact of one variable on another. Keeping all else constant, we can examine if the number of bathrooms in a home impacts its price. If only one variable influences the dependent variable (housing prices), a simple linear regression model would be constructed. But what if we felt that multiple factors influence home prices? (i.e. view, neighborhood, location to nearest city).

- Here, multivariate linear regression becomes an invaluable tool. Multiple regression can incorporate a large number of independent factors into a model to describe how these variables influence the dependent variable. If we desired a model that explained our dependent variable more clearly in the preceding example, we could include factors such as school district, crime rate, or another variable that may help predict your response variable (price).
- Before we get into our analysis of multiple linear regression, we must comprehend the data assumptions we make when developing a regression model and the meanings of our variables. We will then investigate how to comprehend and interpret our model.


Income
Years of Education
Seniority

Assumptions of multiple linear regression:

1. Homogeneity of variance or homoscedasticity: this assumes that the size of the error in our prediction doesn't change significantly across the values of the independent variable.



Copyright 2014. Laerd Statistics.

2. Independence of observations — This assumption states that the observations within our data are independent of one another and have been collected through statistically valid methods.
3. Multicollinearity: In multiple linear regression, it is possible that some of your explanatory variables are correlated with one another. This is known as multicollinearity. For example, height and weight are heavily correlated. If both these variables are used to predict sex, we should only use one of these independent variables in our model as this can create redundant information and skew the results in our regression model.
4. Normality: We assume our data is normally distributed.
5. Linearity: Multiple linear regression requires the relationship between the independent and dependent variables to be linear.

Simple Linear Regression

- Estimate the model parameter and the prediction

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

where

$\hat{y}$ = estimated dependent (response) variable

x = independent (or predictor/ regressor /explanatory) variable

$\hat{\beta}_0$ = estimate of y – intercept, the point which the line intersects the y-axis (regression constant)

$\hat{\beta}_1$ = estimate of slope, the amount of increase/decrease of y for each unit increase (or decrease) in x (regression coefficient)

Multiple Linear Regression

- Used to describe linear relationships involving a dependent variable (y) with more than two independent variables
- The general form of multiple linear regression model is given by

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k + \varepsilon$$

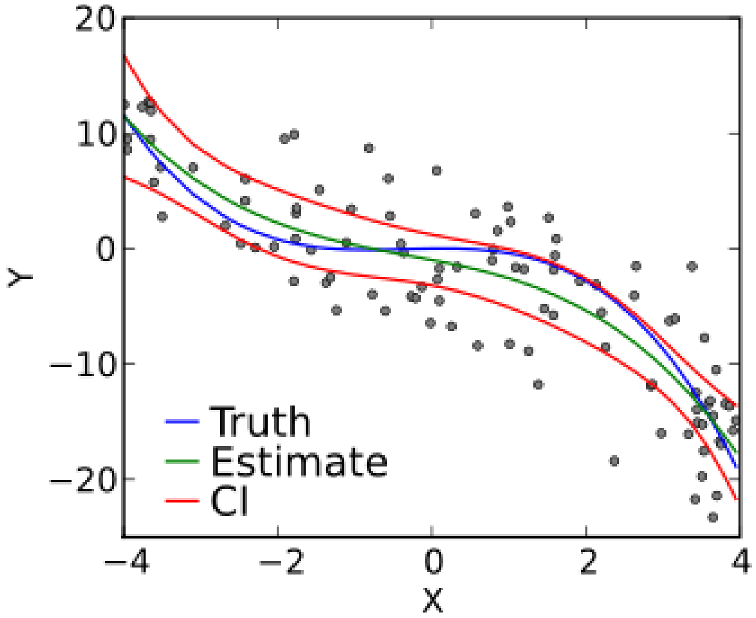
$$where$$

$\beta_0, \beta_1, \ldots, \beta_k$: unknown parameter (regression coefficient)

ϵ : error term

Polynomial Regression Analysis

- Polynomial regression is another type of regression analysis technique in machine learning, which is the same as multiple linear regression with a little modification.
- In polynomial regression, the relationship between independent and dependent variables, that is *X* and *Y*, is denoted by the n-th degree.
- It is a linear model as an estimator. The least Mean Squared Method is used in polynomial regression also.
- The best-fit line in polynomial regression that passes through all the data points is not a straight line, but a curved line, which depends upon the power of *X* or the value of n.

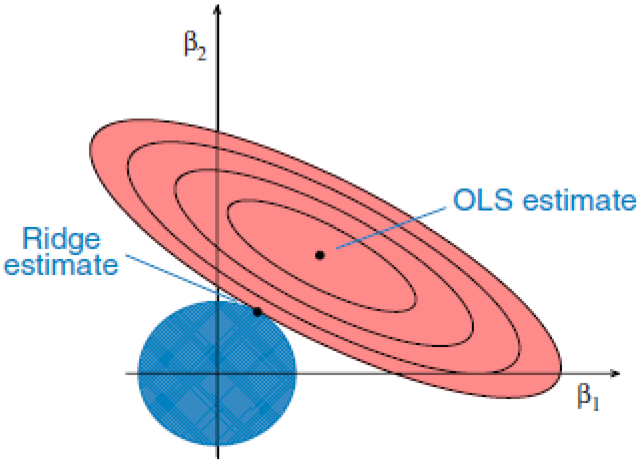


- While trying to reduce the mean squared error to a minimum and to get the best-fit line, the model can be prone to overfitting.
- It is recommended to analyze the curve towards the end as the higher Polynomials can give strange results on extrapolation.
- Below equation represents the Polynomial Regression:

$$Y = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \ldots + \theta_n x^n$$

Ridge Regression Analysis

- Ridge regression is a type of linear regression technique that is used to reduce the overfitting of linear models. Recall that Linear regression is a method of modelling data that represents relationships between a response variable and one or more predictor variables.
- Usually used when there is a **high correlation** between the independent variables.
- This is because, in the case of multi-collinear data, the least square estimates give unbiased values. But, if the collinearity is very high, there can be some bias value.
- Therefore, a bias matrix is introduced in the equation of ridge regression. This is a powerful regression method where the model is less susceptible to overfitting.



- Below is the equation used to denote the ridge regression, where the introduction of *λ* (lambda) solves the problem of multicollinearity:

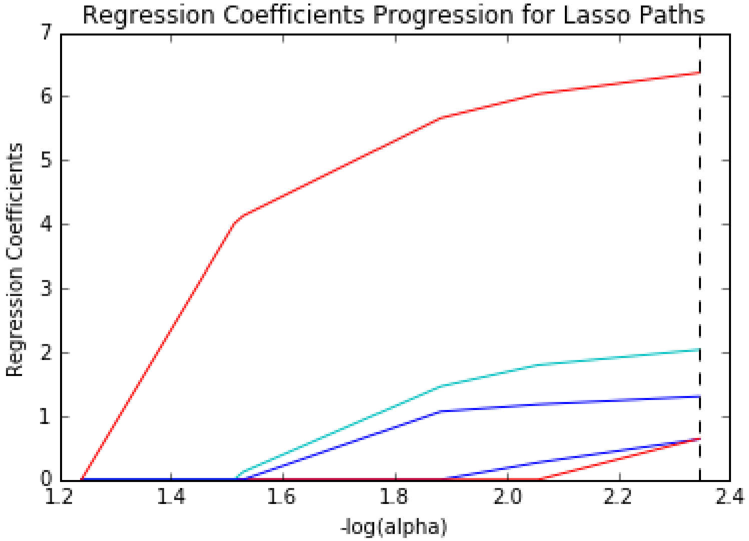
$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(h_\theta(x^{(i)}), y^{(i)}) + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$

$$RSS_{ridge}(w,b) = \underbrace{\sum_{i=1}^{n}(y_i - (w_i x_i + b))^2}_{\text{Fit training data well}} + \alpha \overbrace{\underbrace{\sum_{j=1}^{p} w_j^2}_{\text{Keep parameters small}}}^{\text{L2 penalty / Penalty Term / Regularisation Term}}$$

A trade-off between fitting the training data well and keeping parameters small

Lasso Regression Analysis

- Lasso regression is one of the types of regression that performs regularization along with feature selection.
- It prohibits the absolute size of the regression coefficient. As a result, the coefficient value gets nearer to zero, which does not happen in the case of ridge regression.
- Due to this, feature selection gets used in lasso regression, which allows the selection of a set of features from the dataset to build the model. In the case of lasso Regression, only the required features are used, and the others are made zero.
- This helps in avoiding overfitting in the model. In case the independent variables are **highly collinear**, then Lasso regression picks only one variable and makes other variables shrink to zero.



- Lasso is a modification of linear regression, where the model is penalized for the sum of absolute values of the weights. Thus, the absolute values of weight will be (in general) reduced, and many will tend to be zeros.
- Lasso introduced a new hyperparameter, alpha, and the coefficient to penalize weights. During training, the objective function becomes:

$$\frac{1}{2m}\sum_{i=1}^{m}(y - Xw)^2 + alpha\sum_{j=1}^{p}\left|w_j\right|$$

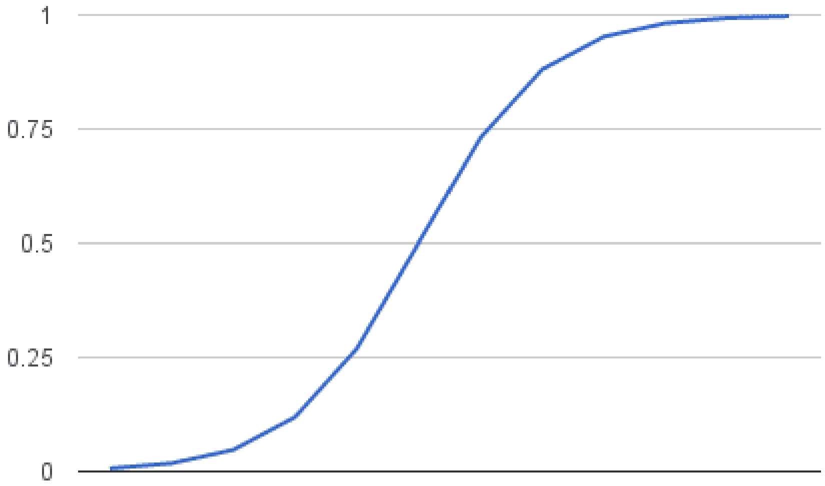
Logistic Regression Analysis

- Logistic regression uses an equation as the representation, very much like linear regression.
- Input values (x) are combined linearly using weights or coefficient values to predict an output value (y).
- Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis.
- Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

Logistic Function

- Logistic regression is named for the function used at the core of the method, the logistic function.
- The logistic function, also called the sigmoid function was developed by statisticians to describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment.
- It’s an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

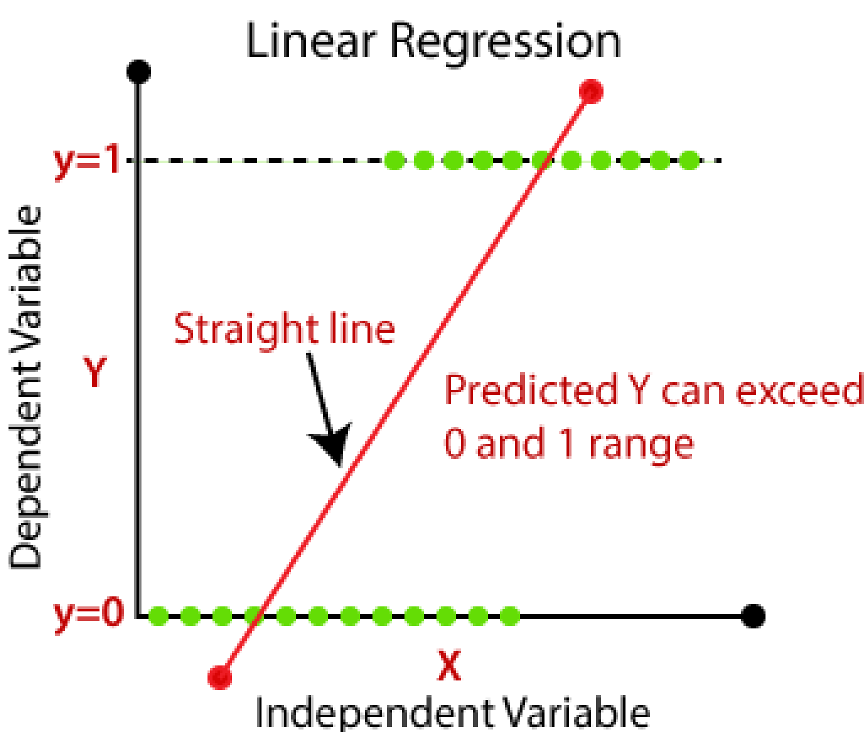
- Where e is the base of the natural logarithms (Euler’s number or the EXP() function in your spreadsheet) and value is the actual numerical value that you want to transform.
- Below is a plot of the numbers between -5 and 5 transformed into the range 0 and 1 using the logistic function.

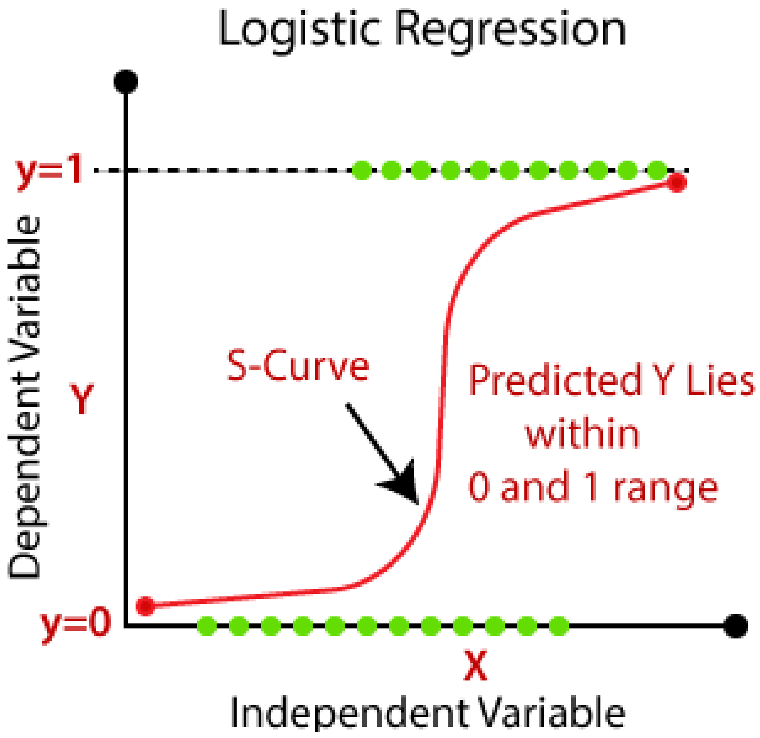


All regression models attempt to model the relationship f between a dependent variable y and a number of independent variable x_i.

Differences between linear and logistic regression

	Linear Regression	Logistic Regression
Dependent variable, y	Numeric	Nominal
Functional relationship between independent and..	..dependent variables $y = f(x_1,\ldots, x_n, \beta_o,\ldots, \beta_n)$ $y = \beta_o + \beta_1 x_1 + \ldots + \beta_n x_n)$	…class probability $P(y$= class $i)$ $P(y = c_i) = f(x_1,\ldots, x_n, \beta_0,\ldots, \beta_n)$


Linear Regression
y=1
y=0
Dependent Variable
Y
Straight line
Predicted Y can exceed
0 and 1 range
X
Independent Variable


Logistic Regression
y=1
y=0
Dependent Variable
Y
S-Curve
Predicted Y Lies
within
0 and 1 range
X
Independent Variable

Dependent variable has two possible outcomes, such as:

$\{y = white, y = black\}$ which can be coded as $\{y = 0, y = 1\}$

Function that describes the relationship between the probability of a class $y = 1$, and the independent variables

$$P(y = 1|x) = \frac{1}{1 + \exp^{(-z)}} = \frac{\exp^{(z)}}{1 + \exp^{(z)}} \quad \in [0,1]$$

With $z = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n = x\beta$, from linear regression model

Then $P(y = 1|x)$...aka as π

The probabilities of all classes have to sum up to 1:

$$\Rightarrow P(y=1|x)+P(y=0|x)=1$$

$$\Rightarrow P(y=0|x)=1-\pi=1-P(y=1|x)$$

we don’t need any coefficients for the second class.

Binary Logistic Regression

Calculating the regression coefficients

In order to calculate the coefficients, we maximize the likelihood function $L(\beta, y, X)$ to get the best approximation of the probability

$$L(\beta, y, X) = \prod_{i=1}^{m} \pi_i^{y_i} (1-\pi_i)^{1-y_i}$$

$y_i = 0$ if y_i is equal to the reference category

$y_i = 1$ if y_i isn’t equal to the reference category

.

Calculating the regression coefficients

The algorithm is a monotonically increasing function

=> Maximizing the algorithm of the Likelihood function $LL(\beta; y, X)$

$$\max_{\beta} LL(\beta; y, X) = \max_{\beta} \sum_{i=1}^{n} y_i \ln(\pi_i) + \ln(1 - y_i) \ln(1 - \pi_i)$$

is equivalent to maximizing the original Likelihood function

.

$$\max_{\beta} LL(\beta; y, X) = \max_{\beta} \prod_{i=1}^{m} \pi_i^{y_i} (1 - \pi_i)^{1 - y_i}$$

Meaning of the Regression Coefficients

Interpretation of the sign:

$\beta_i > 0$: Higher xi leads to higher probability

$\beta_i < 0$: Higher xi leads to smaller probability

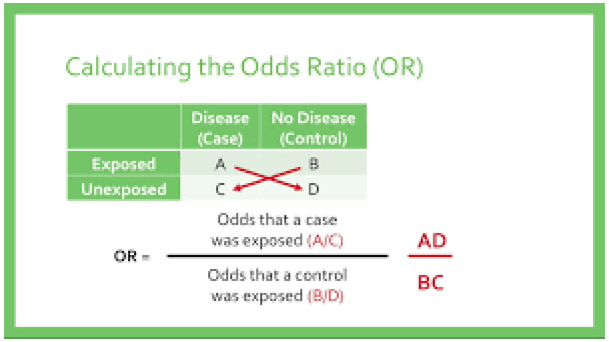
Interpretation of the p-value, which is the result of the Wald test (show whether a feature has significant impact)

Other advanced interpretation method: Odd ratio

.

$$OddsRatio(x_i) = \exp(\beta_i)$$

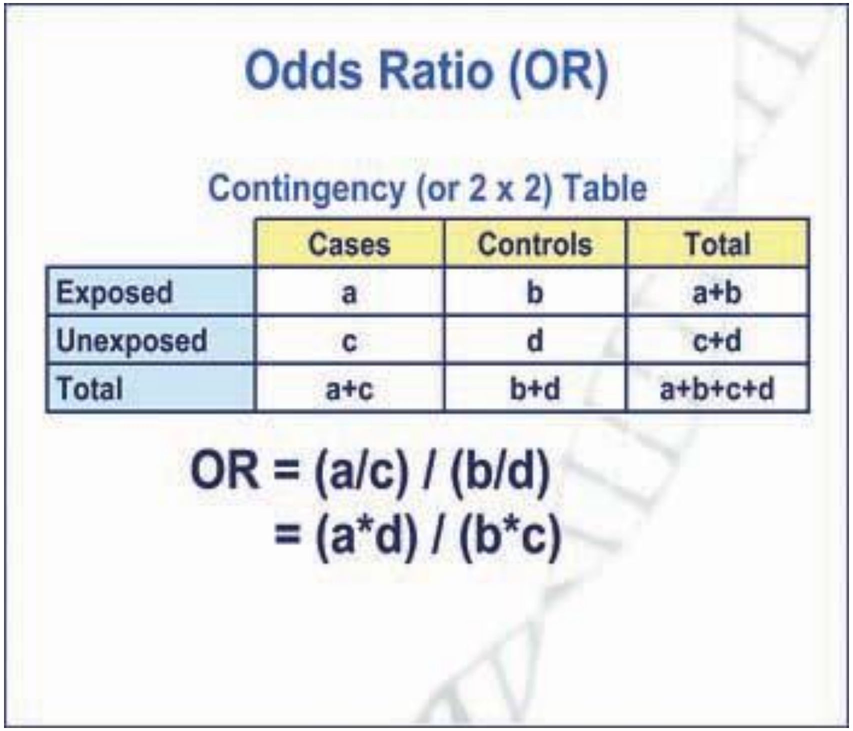
An odds ratio (OR) is a measure of association between a certain property A and a second property B in a population. Specifically, it tells you how the presence or absence of property A has an effect on the presence or absence of property B. The OR is also used to figure out if a particular exposure (like eating processed meat) is a risk factor for a particular outcome (such as colon cancer), and to compare the various risk factors for that outcome.



Calculating the Odds Ratio (OR)

	Disease (Case)	No Disease (Control)
Exposed	A	B
Unexposed	C	D

$$OR = \frac{\text{Odds that a case was exposed (A/C)}}{\text{Odds that a control was exposed (B/D)}} \quad \frac{AD}{BC}$$



Odds Ratio (OR)

Contingency (or 2 x 2) Table

	Cases	Controls	Total
Exposed	a	b	a+b
Unexposed	c	d	c+d
Total	a+c	b+d	a+b+c+d

$$OR = (a/c) / (b/d)$$
$$= (a*d) / (b*c)$$

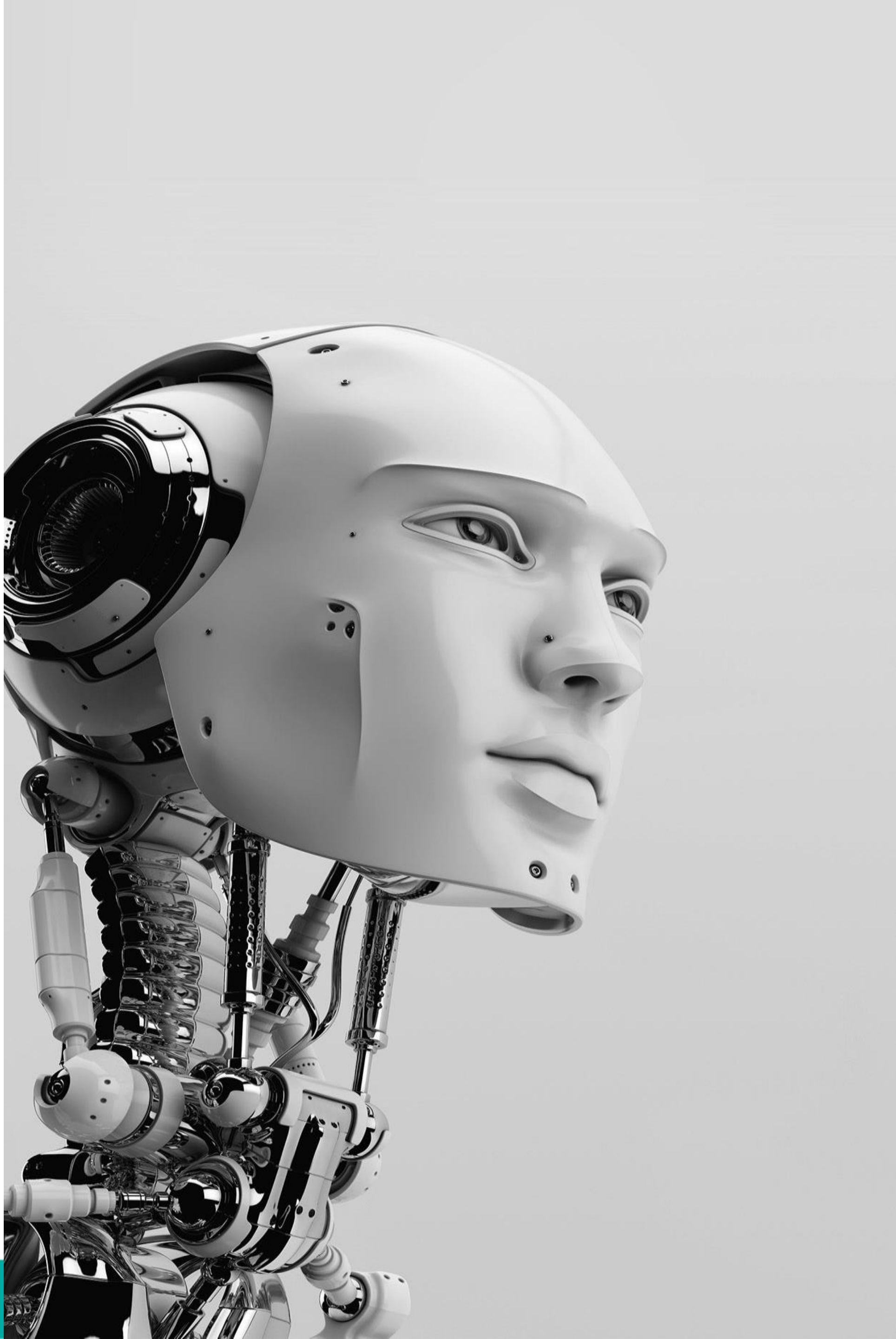
Logistic Regression Applications

- Credit scoring: ID Finance is a financial company that makes predictive models for credit scoring. They need their models to be easily interpretable. They can be asked by a regulator about a certain decision at any moment.
- Medicine: Medical information is gathered in such a way that when a research group studies a biological molecule and its properties, they publish a paper about it. Thus, there is a huge amount of medical data about various compounds, but they are not combined into a single database.
- Text editing: used to make some claim about a text fragment. Toxic speech detection, topic classification for questions to support, and email sorting are examples where logistic regression shows good results.

Chapter 3.2: Classification

By the end of this topic, you should be able to:

- understand the concepts of classification analysis in data mining process.
- understand the applications of classification analysis in data science problems.



Classification Analysis

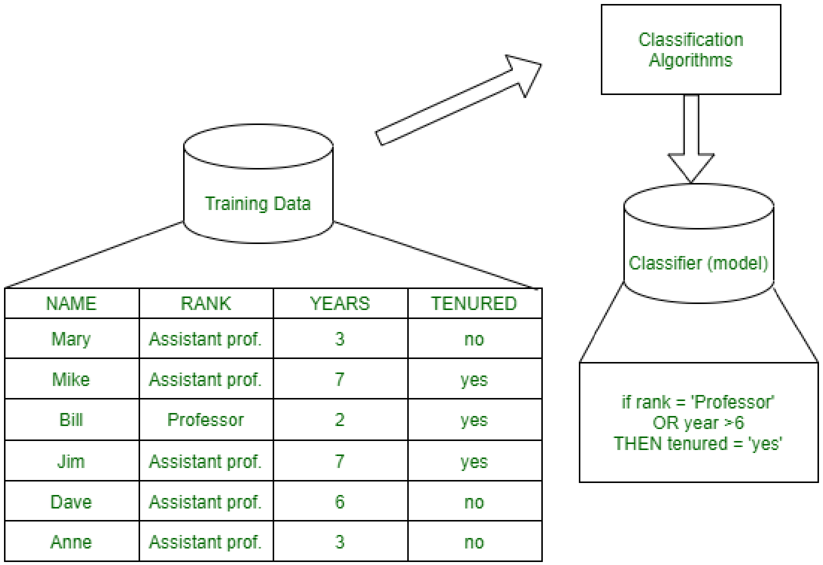
- Classification: It is a data analysis task, i.e. the process of finding a model that describes and distinguishes data classes and concepts. Classification is the problem of identifying to which of a set of categories (subpopulations), a new observation belongs to, on the basis of a training set of data containing observations and whose categories membership is known.
- Example: Before starting any project, we need to check its feasibility. In this case, a classifier is required to predict class labels such as ‘Safe’ and ‘Risky’ for adopting the Project and to further approve it. It is a two-step process such as:

Reference: https://www.geeksforgeeks.org/basic-concept-classification-data-mining/

Learning Step (Training Phase):

Construction of Classification Model

Different Algorithms are used to build a classifier by making the model learn using the training set available. The model has to be trained for the prediction of accurate results.



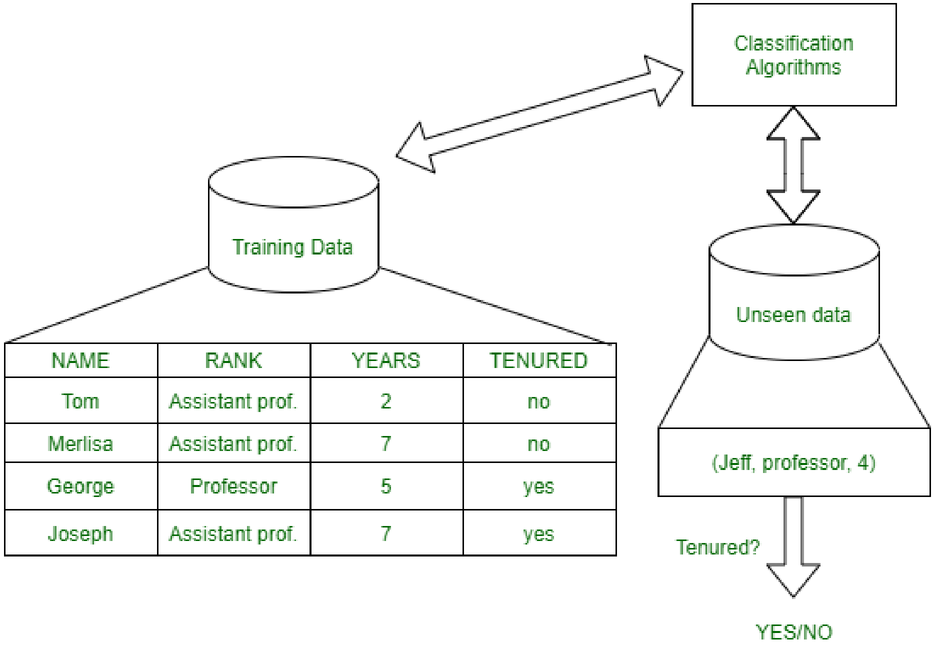
NAME	RANK	YEARS	TENURED
Mary	Assistant prof.	3	no
Mike	Assistant prof.	7	yes
Bill	Professor	2	yes
Jim	Assistant prof.	7	yes
Dave	Assistant prof.	6	no
Anne	Assistant prof.	3	no

if rank = 'Professor'
OR year >6
THEN tenured = 'yes'

Reference: https://www.geeksforgeeks.org/basic-concept-classification-data-mining/

Classification Step

Model used to predict class labels and testing the constructed model on test data and hence estimate the accuracy of the classification rules.



NAME	RANK	YEARS	TENURED
Tom	Assistant prof.	2	no
Merlisa	Assistant prof.	7	no
George	Professor	5	yes
Joseph	Assistant prof.	7	yes

Reference: https://www.geeksforgeeks.org/basic-concept-classification-data-mining/

Types of Classifiers

Discriminative

It is a very basic classifier and determines just one class for each row of data. It tries to model just by depending on the observed data, depends heavily on the quality of data rather than on distributions.

- Example: Logistic Regression
- Acceptance of a student at a University (Test and Grades need to be considered)

Reference: https://www.geeksforgeeks.org/basic-concept-classification-data-mining/

Generative

- It models the distribution of individual classes and tries to learn the model that generates the data behind the scenes by estimating assumptions and distributions of the model. Used to predict the unseen data.
- Example: Naive Bayes Classifier
- Detecting Spam emails by looking at the previous data. Suppose 100 emails and that too divided in 1:4 i.e. Class A: 25%(Spam emails) and Class B: 75%(Non-Spam emails). Now if a user wants to check that if an email contains the word cheap, then that may be termed as Spam. It seems to be that in Class A(i.e. in 25% of data), 20 out of 25 emails are spam and rest not. And in Class B(i.e. in 75% of data), 70 out of 75 emails are not spam and rest are spam. So, if the email contains the word cheap, what is the probability of it being spam ?? (= 80%)

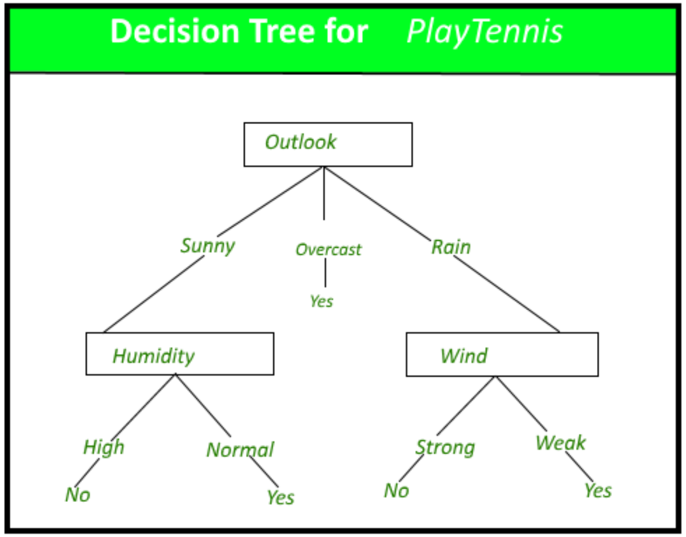
Reference: https://www.geeksforgeeks.org/basic-concept-classification-data-mining/

Classifiers

- Decision Trees
- Bayesian Classifiers
- K-Nearest Neighbors
- Support Vector Machines
- Logistic Regression

Decision Tree

- Decision tree is the most powerful and popular tool for classification and prediction. A decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.



Reference: https://www.geeksforgeeks.org/decision-tree/

Construction of Decision Tree:

A tree can be “learned” by splitting the source set into subsets based on an attribute value test. This process is repeated on each derived subset in a recursive manner called recursive partitioning. The recursion is completed when the subset at a node all has the same value of the target variable, or when splitting no longer adds value to the predictions. The construction of a decision tree classifier does not require any domain knowledge or parameter setting, and therefore is appropriate for exploratory knowledge discovery. Decision trees can handle high-dimensional data. In general decision tree classifier has good accuracy. Decision tree induction is a typical inductive approach to learning knowledge on classification.

Reference: https://www.geeksforgeeks.org/decision-tree/

Decision Tree Representation:

Decision trees classify instances by sorting them down the tree from the root to some leaf node, which provides the classification of the instance. An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute as shown in the above figure. This process is then repeated for the subtree rooted at the new node.

The decision tree in above figure classifies a particular morning according to whether it is suitable for playing tennis and returning the classification associated with the particular leaf.(Yes or No).

For example, the instance

(Outlook = Rain, Temperature = Hot, Humidity = High, Wind = Strong)

Reference: https://www.geeksforgeeks.org/decision-tree/

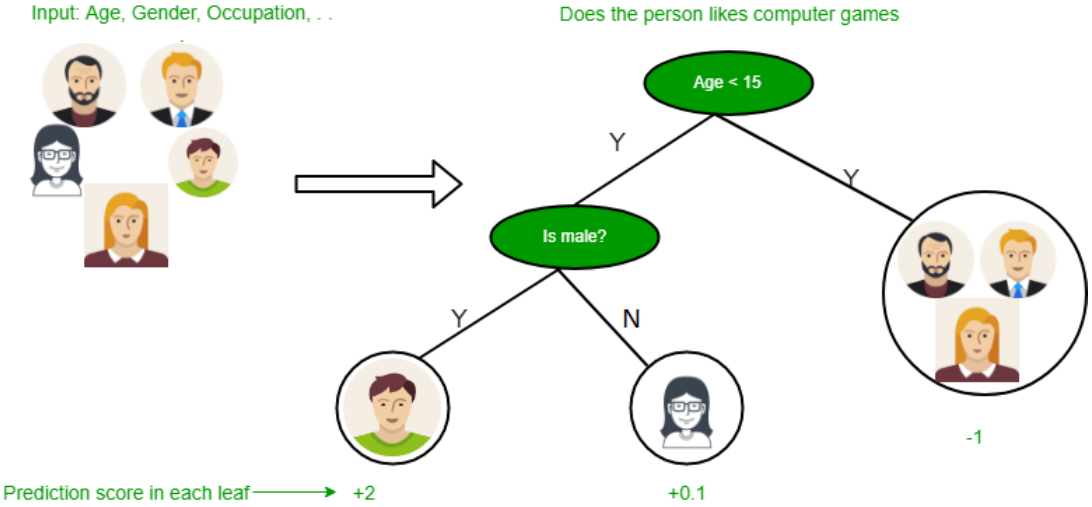
would be sorted down the leftmost branch of this decision tree and would therefore be classified as a negative instance.

In other words we can say that decision tree represent a disjunction of conjunctions of constraints on the attribute values of instances.

(Outlook = Sunny ^ Humidity = Normal) v (Outlook = Overcast) v (Outlook = Rain ^ Wind = Weak)

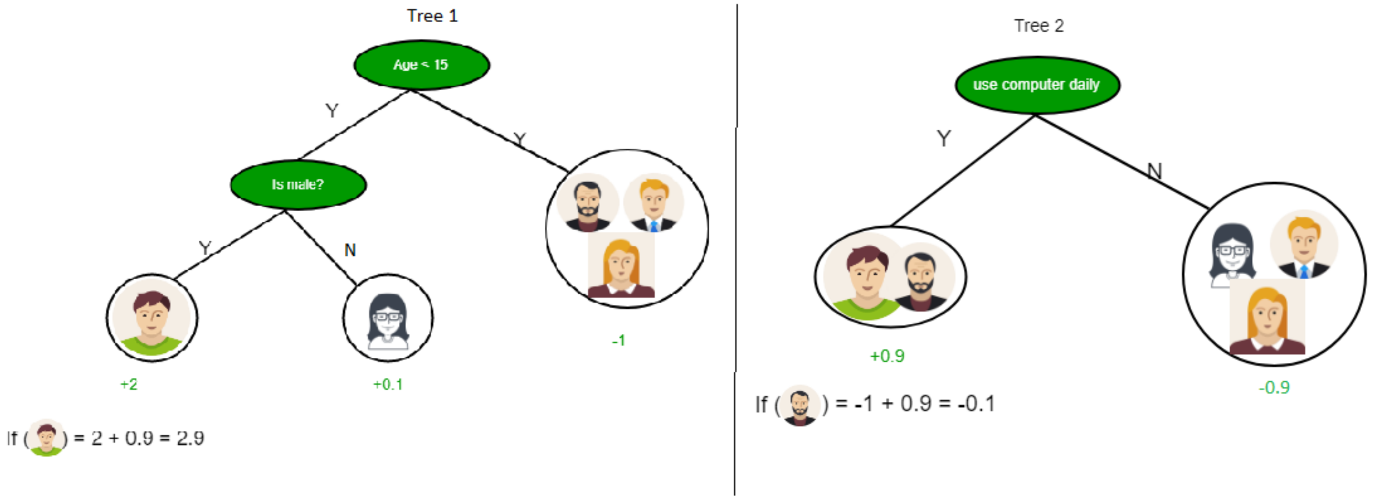
Reference: https://www.geeksforgeeks.org/decision-tree/

- Decision tree algorithm falls under the category of supervised learning. They can be used to solve both regression and classification problems.
- Decision tree uses the tree representation to solve the problem in which each leaf node corresponds to a class label and attributes are represented on the internal node of the tree.
- We can represent any Boolean function on discrete attributes using the decision tree.


Input: Age, Gender, Occupation, . .
Does the person likes computer games
Age < 15
Y
Y
Is male?
Y
N
-1
Prediction score in each leaf
+2
+0.1

Reference: https://www.geeksforgeeks.org/decision-tree-introduction-example/

- At the beginning, we consider the whole training set as the root.
- Feature values are preferred to be categorical. If the values are continuous then they are discretized prior to building the model.
- On the basis of attribute values records are distributed recursively.
- We use statistical methods for ordering attributes as root or the internal node.



Reference: https://www.geeksforgeeks.org/decision-tree-introduction-example/

- As you can see from the above image that Decision Tree works on the Sum of Product form which is also known as Disjunctive Normal Form. In the previous image, we are predicting the use of computer in the daily life of the people.
- In Decision Tree the major challenge is to identification of the attribute for the root node in each level. This process is known as attribute selection. We have two popular attribute selection measures:

1. Information Gain/ Gain Ratio
2. Gini Index

Reference: https://www.geeksforgeeks.org/decision-tree-introduction-example/

Information Gain/ Gain Ratio

- When we use a node in a decision tree to partition the training instances into smaller subsets the entropy changes. Information gain is a measure of this change in entropy.
- Definition: Suppose S is a set of instances, A is an attribute, Sv is the subset of S with A = v, and Values (A) is the set of all possible values of A, then

$$Gain(S,A) = Entropy(S) - \sum_{v \epsilon Values(A)} \frac{|S_v|}{|S|} . Entropy(S_v)$$

Entropy

- Entropy is the measure of uncertainty of a random variable, it characterizes the impurity of an arbitrary collection of examples. The higher the entropy more the information content.

Reference: https://www.geeksforgeeks.org/decision-tree-introduction-example/

Information Gain/ Gain Ratio

Definition: Suppose S is a set of instances, A is an attribute, Sv is the subset of S with A = v, and Values (A) is the set of all possible values of A, then

$$Gain(S, A) = Entropy(S) - \sum_{v \epsilon Values(A)} \frac{|S_v|}{|S|} . Entropy(S_v)$$

Example:

```
For the set X = {a,a,a,b,b,b,b,b}

Total intances: 8

Instances of b: 5

Instances of a: 3
```

$$Entropy H(X) = -\left[\left(\frac{3}{8}\right) log_2 \frac{3}{8} + \left(\frac{5}{8}\right) log_2 \frac{5}{8}\right]$$

```
= -[0.375 * (-1.415) + 0.625 * (-0.678)]

=-(-0.53-0.424)

= 0.954
```

Reference: https://www.geeksforgeeks.org/decision-tree-introduction-example/

Building Decision Tree using Information Gain

The essentials:

- Start with all training instances associated with the root node
- Use info gain to choose which attribute to label each node with
- Note: No root-to-leaf path should contain the same discrete attribute twice
- Recursively construct each subtree on the subset of training instances that would be classified down that path in the tree.

The border cases:

- If all positive or all negative training instances remain, label that node “yes” or “no” accordingly
- If no attributes remain, label with a majority vote of training instances left at that node
- If no instances remain, label with a majority vote of the parent’s training instances

Reference: https://www.geeksforgeeks.org/decision-tree-introduction-example/

Example:

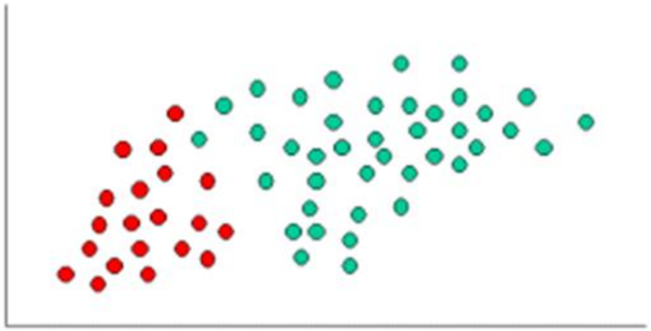
Now, let’s draw a Decision Tree for the following data using Information gain.

Training set: 3 features and 2 classes…

Reference: https://www.geeksforgeeks.org/decision-tree-introduction-example/

Naïve Bayes

- The Naive Bayes algorithm is based on the so-called Bayesian theorem and is particularly suited when the dimensionality of the inputs is high. Despite its simplicity, Naive Bayes can often outperform more sophisticated classification methods.



- To demonstrate the concept of Naïve Bayes, consider the example displayed in the illustration above. As indicated, the objects can be classified as either GREEN or RED. Our task is to classify new cases as they arrive, i.e., decide to which class label they belong, based on the currently exiting objects.

- Since there are twice as many GREEN objects as RED, it is reasonable to believe that a new case (which hasn't been observed yet) is twice as likely to have membership GREEN rather than RED.
- In the Bayesian analysis, this belief is known as the prior probability. Prior probabilities are based on previous experience, in this case the percentage of GREEN and RED objects, and often used to predict outcomes before they actually happen.
- Thus, we can write:

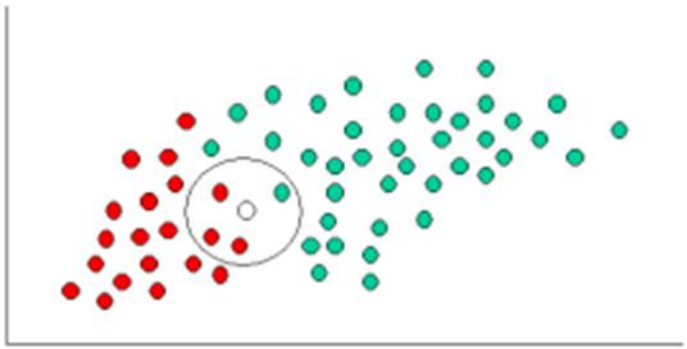
$$\text{Prior probability for GREEN} \propto \frac{\text{Number of GREEN objects}}{\text{Total number of objects}}$$

$$\text{Prior probability for RED} \propto \frac{\text{Number of RED objects}}{\text{Total number of objects}}$$

- Since there is a total of 60 objects, 40 of which are GREEN and 20 RED, our prior probabilities for class membership are:

$$\text{Prior probability for GREEN} \propto \frac{40}{60}$$

$$\text{Prior probability for RED} \propto \frac{20}{60}$$



- Having formulated our prior probability, we are now ready to classify a new object (WHITE circle). Since the objects are well clustered, it is reasonable to assume that the more GREEN (or RED) objects in the vicinity of X, the more likely that the new cases belong to that particular colour.
- To measure this likelihood, we draw a circle around X which encompasses a number (to be chosen a priori) of points irrespective of their class labels. Then we calculate the number of points in the circle belonging to each class label. From this we calculate the likelihood:

$$\text{Likelihood of } X \text{ given GREEN} \propto \frac{\text{Number of GREEN in the vicinity of } X}{\text{Total number of GREEN cases}}$$

$$\text{Likelihood of } X \text{ given RED} \propto \frac{\text{Number of RED in the vicinity of } X}{\text{Total number of RED cases}}$$

- From the illustration above, it is clear that Likelihood of X given GREEN is smaller than Likelihood of X given RED, since the circle encompasses 1 GREEN object and 3 RED ones. Thus:

$$\textit{Probability of } X \textit{ given GREEN} \propto \frac{1}{40}$$

$$\textit{Probability of } X \textit{ given RED} \propto \frac{3}{20}$$

- Although the prior probabilities indicate that X may belong to GREEN (given that there are twice as many GREEN compared to RED) the likelihood indicates otherwise; that the class membership of X is RED (given that there are more RED objects in the vicinity of X than GREEN). In the Bayesian analysis, the final classification is produced by combining both sources of information, i.e., the prior and the likelihood, to form a posterior probability using the so-called Bayes' rule (named after Rev. Thomas Bayes 1702-1761).

$$\textit{Posterior probability of } X \textit{ being GREEN} \propto$$

$$\textit{Prior probability of GREEN} \times \textit{Likelihood of } X \textit{ given GREEN}$$

$$= \frac{4}{6} \times \frac{1}{40} = \frac{1}{60}$$

$$\textit{Posterior probability of } X \textit{ being RED} \propto$$

$$\textit{Prior probability of RED} \times \textit{Likelihood of } X \textit{ given RED}$$

$$= \frac{2}{6} \times \frac{3}{20} = \frac{1}{20}$$

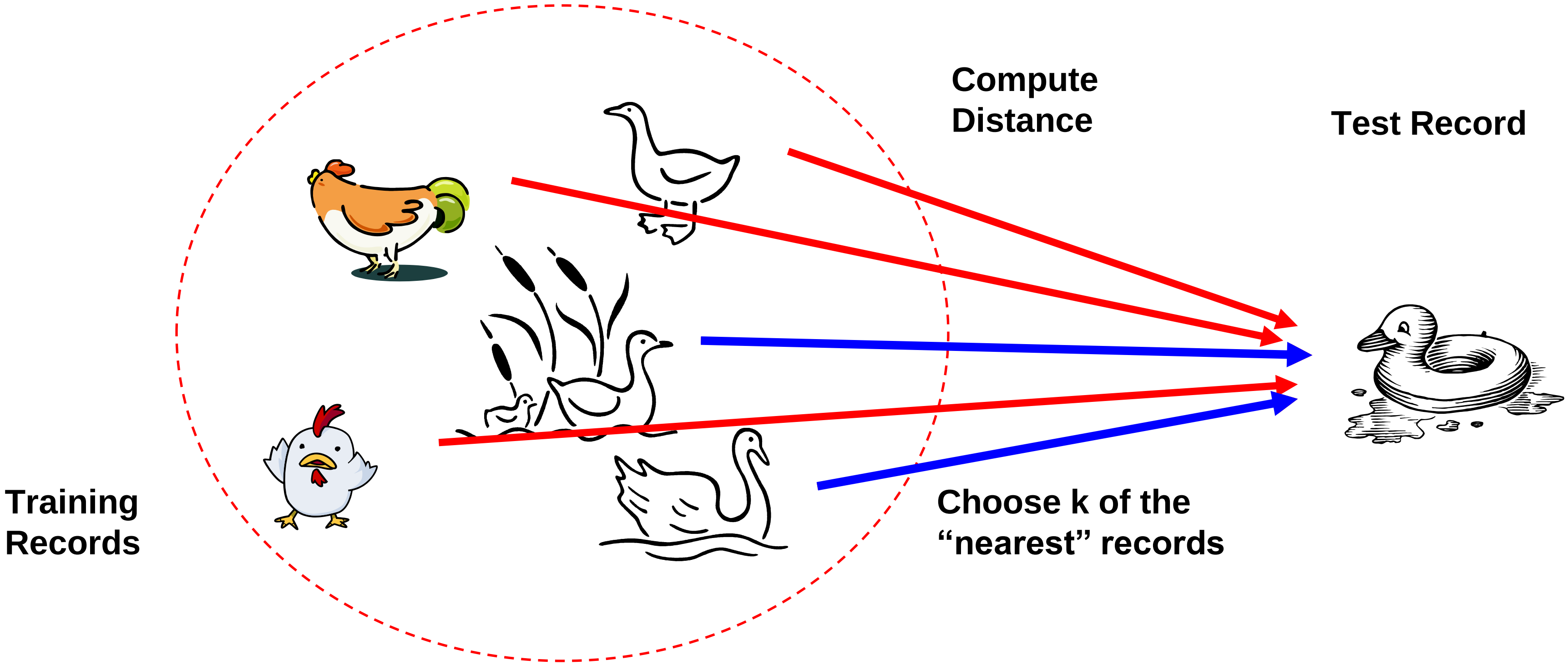
- Finally, we classify X as RED since its class membership achieves the largest posterior probability.

Note: The above probabilities are not normalized. However, this does not affect the classification outcome since their normalizing constants are the same.

K-Nearest Neighbors

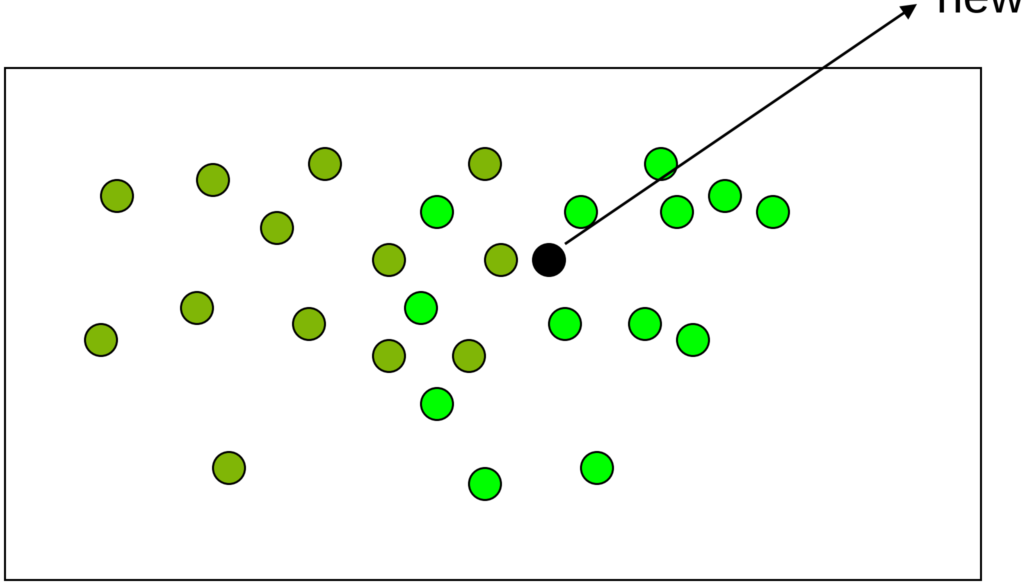
Basic idea:

- If it walks like a duck, quacks like a duck, then it's probably a duck.



- Majority vote within the k nearest neighbors.

$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i$$



K= 1: dark green

K= 3: green

Requires three things:

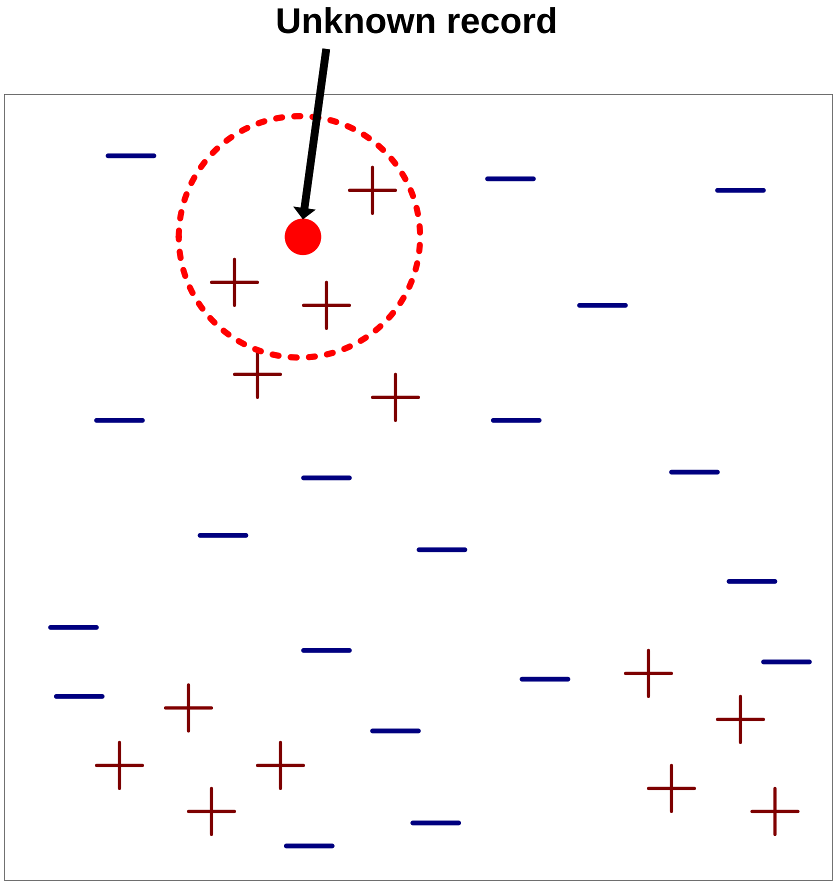
a. The set of stored records.

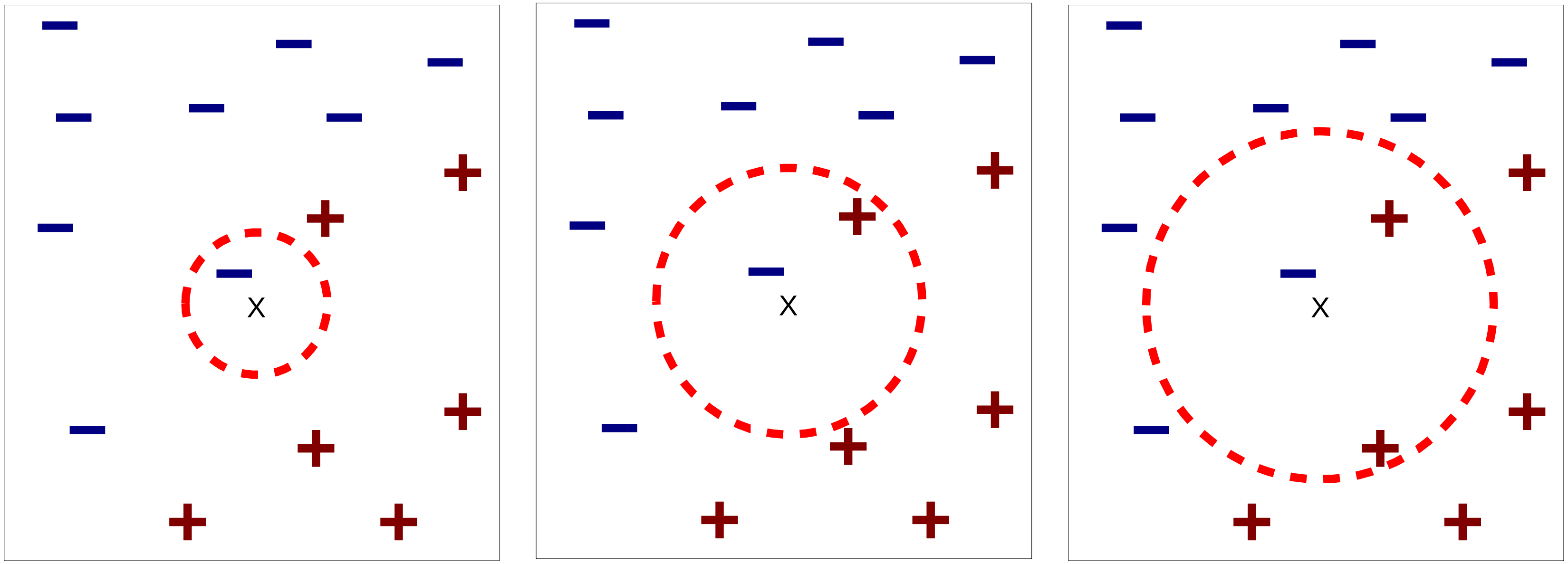
b. Distance Metric to compute distance between records.

c. The value of *k*, the number of nearest neighbors to retrieve.

To classify an unknown record:

- ✓ Compute distance to other training records
- ✓ Identify k nearest neighbors
- ✓ Use class labels of nearest neighbors to determine the class label of unknown record (e.g., by taking majority vote.





(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*.

Compute distance between two points:
Euclidean distance

$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}$$

Determine the class from nearest neighbor list:

Take the majority vote of class labels among the k-nearest neighbors.

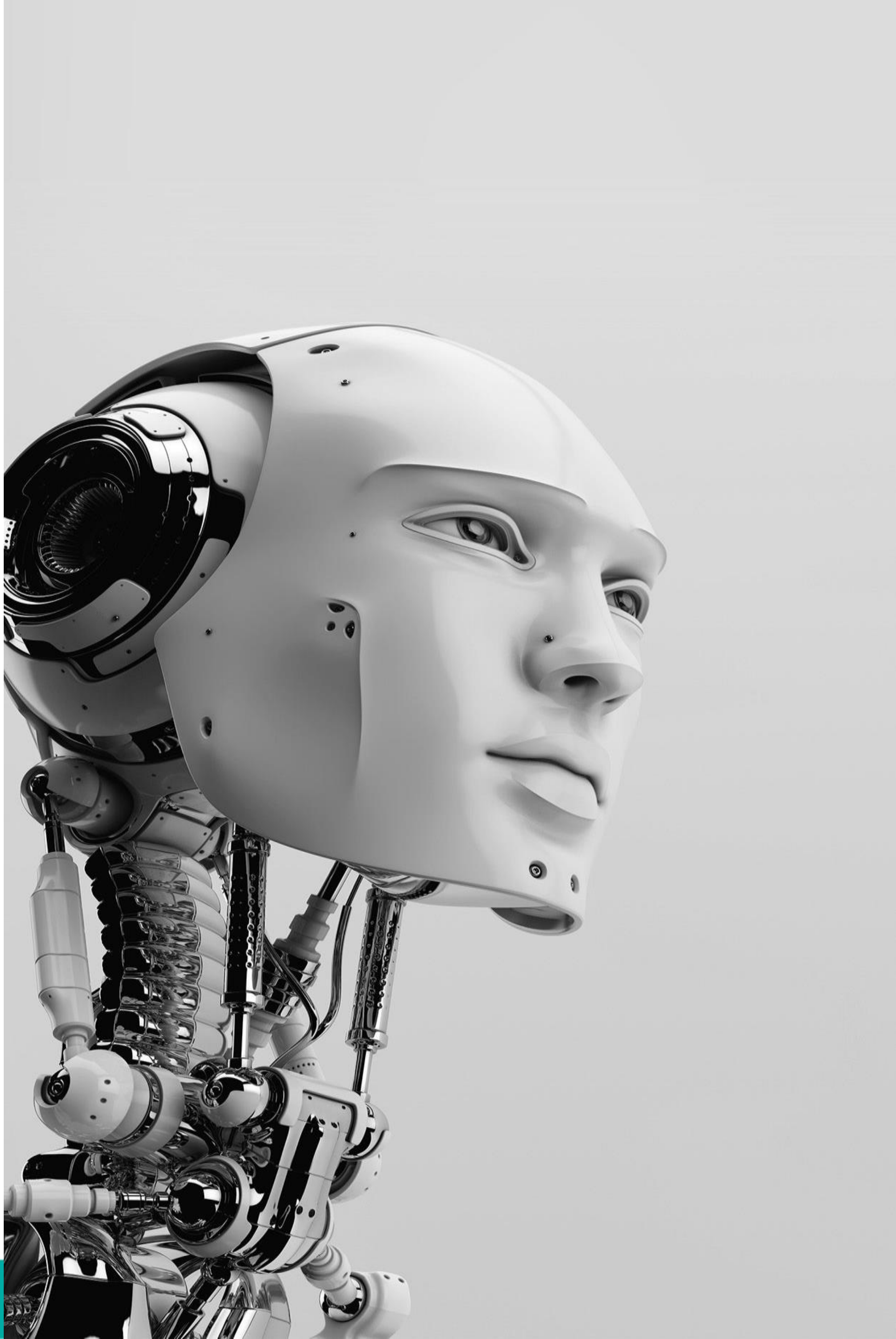
Weight the vote according to distance using

weight factor, $w = 1/d^2$

Chapter 3.3: Evaluation Metrics for Predictive Modelling

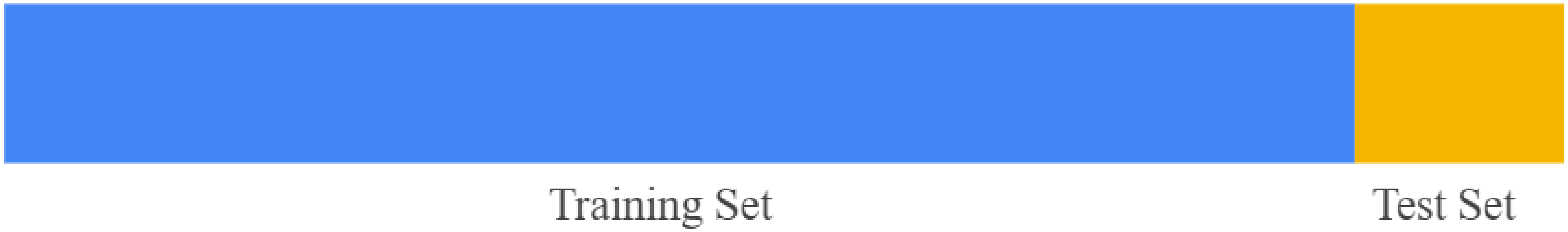
By the end of this topic, you should be able to:

- understand and apply the data partitioning in data mining process.
- understand the need of model evaluation in data mining algorithms.
- apply the regression metrics and classification metrics.



Training and Testing Dataset

- Training set — a subset to train a model.
- Test set — a subset to test the trained model.
- You could imagine slicing the single data set as follows:



- From the prepared data, basically we need to split our dataset into a training and testing set.
- An algorithm will be trained on the training dataset and will be evaluated against the test set.

- Make sure that your test set meets the following two conditions:
    a. Is large enough to yield statistically meaningful results.
    b. Is representative of the data set as a whole. In other words, don't pick a test set with different characteristics than the training set.
- Assuming that your test set meets the preceding two conditions, your goal is to create a model that generalizes well to new data.
- Our test set serves as a proxy for new data. For example, consider the following figure. Notice that the model learned for the training data is very simple.
- This model doesn't do a perfect job—a few predictions are wrong. However, this model does about as well on the test data as it does on the training data. In other words, this simple model does not overfit the training data.

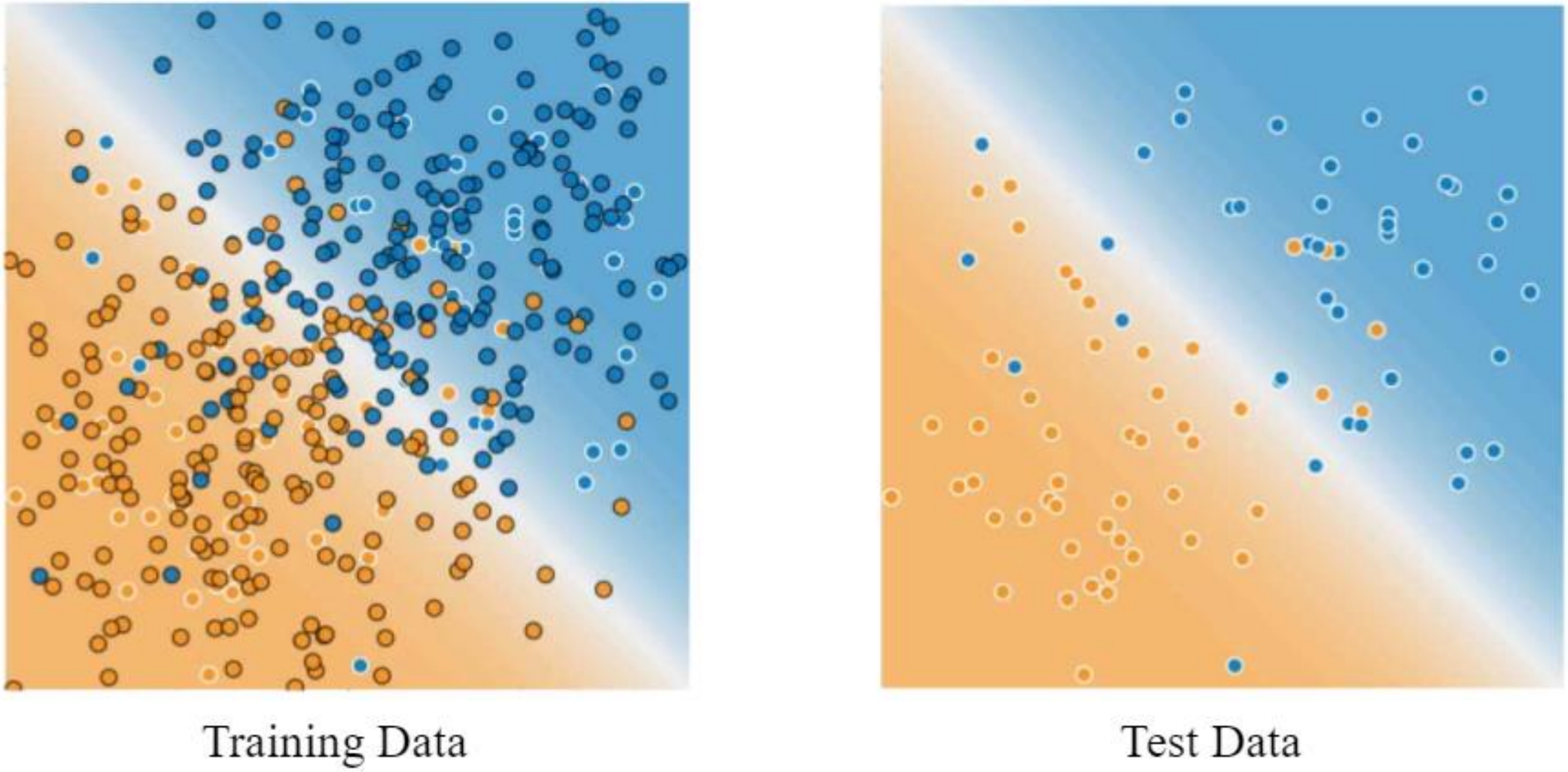
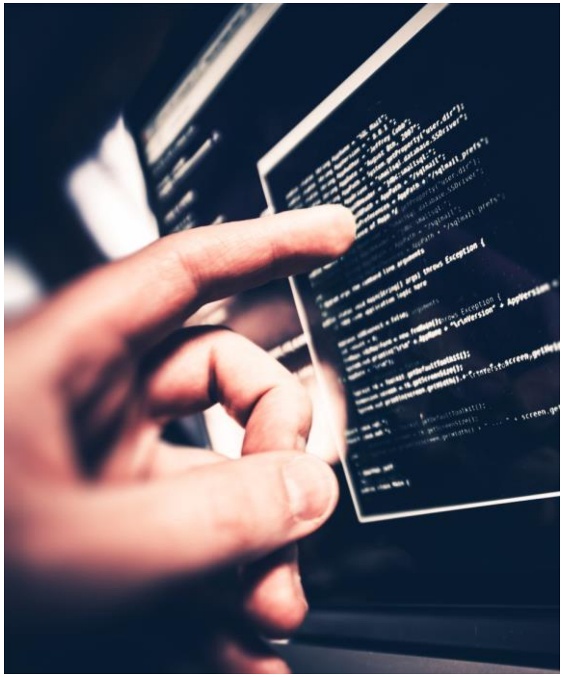


Figure 4.1: Training and testing data distribution.

Model Evaluation

- The idea of building machine learning models works on a constructive feedback principle.
- You build a model, get feedback from metrics, make improvements and continue until you achieve a desirable accuracy.
- Evaluation metrics explain the performance of a model. An important aspect of evaluation metrics is their capability to discriminate among model results.
- Some evaluation metrics utilized: regression and classification metrics.



(1) Regression Metrics

- Evaluation metrics for regression models are quite different than the classification models because we are now predicting in a continuous range instead of a discrete number of classes.
- If your regression model predicts the price of a house to be $400K and it sells for $405K, that's a pretty good prediction. However, in the classification examples we were only concerned with whether or not a prediction was correct or incorrect, there was no ability to say a prediction was "pretty good".
- Thus, here we cover R-squared and some error terms evaluation metrics for regression models.

R-Squared

- To evaluate the overall fit of a linear regression model, we use the R-squared value.
- R-squared is the proportion of variance explained.
- It is the proportion of variance in the observed data that is explained by the model, or the reduction in error over the null model.
- The null model just predicts the mean of the observed response, and thus it has an intercept and no slope.
- R-squared is between 0 and 1.
- Higher values are better because it means that more variance is explained by the model.

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \overline{y})^2}$$

Error Terms

Root Mean Squared Error (RMSE)

- RMSE (ranges from 0 to infinity, lower is better), also called Root Mean Square Deviation (RMSD), is a quadratic-based rule to measure the absolute average magnitude of the error.
- Technically it is produce by taking residuals (the difference between the regression model and the actual data), squaring it, averaging all the results and then taking the square root of the average. Because of this the product will always be a positive number.

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

- Because values are squared before averages, the affect of larger errors (think of the result of 3^2 compared to 8^2) is greatly amplified and should be used of those kinds of errors are important to identify.

- RMSE can be normalized by mean or range in order to compared between models of different scale. It will usually be expressed as a percentage and notated as NRMSD or NRMSE.

Mean Absolute Error (MAE)

- MAE (ranges from 0 to infinity, lower is better) is much like RMSE, but instead of squaring the difference of the residuals and taking the square root of the result, it just averages the absolute difference of the residuals.
- This produces a positive only numbers, and is less reactive to large errors but can show nuance a bit better. It has also fallen out of favor over time.

$$\mathrm{MAE} = \frac{1}{n}\sum_{j=1}^{n} |y_j - \hat{y}_j|$$

(2) Classification Metrics

When performing classification predictions, there's four types of outcomes that could occur.

a. **True positives**: are when you predict an observation belongs to a class and it actually does belong to that class.

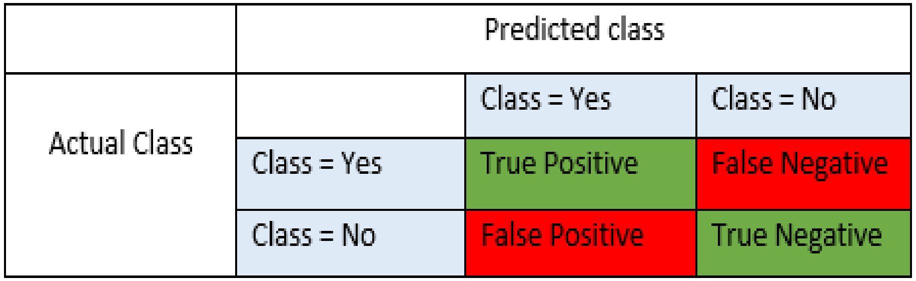
b. **True negatives**: are when you predict an observation does not belong to a class and it actually does not belong to that class.

c. **False positives**: occur when you predict an observation belongs to a class when in reality it does not.

d. **False negatives**: occur when you predict an observation does not belong to a class when in fact it does.

Confusion Matrix

A confusion matrix - describe the performance of a classification model on a set of test data for which the true values are known.



		Predicted class	
Actual Class		Class = Yes	Class = No
	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

True positive and **true negatives** are the observations that are correctly predicted.
We want to minimize false positives and false negatives

True Positives (TP) - These are the correctly predicted positive values which means that the value of actual class is yes and the value of predicted class is also yes.

True Negatives (TN) - These are the correctly predicted negative values which means that the value of actual class is no and value of predicted class is also no.

False positives and **false negatives,** these values occur when actual class contradicts with the predicted class.
False Positives (FP) – When actual class is no and predicted class is yes.
False Negatives (FN) – When actual class is yes but predicted class in no.

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.
From the model, we have got 0.803 which means our model is approximate 80% accurate.

Accuracy = TP+TN/TP+FP+FN+TN

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

Precision = TP/TP+FP

Recall (Sensitivity) is the ratio of correctly predicted positive observations to the all observations in actual class - yes.

Recall = TP/TP+FN

F1 Score is the weighted average of Precision and Recall.

F1 Score = 2*(Recall * Precision) / (Recall + Precision)

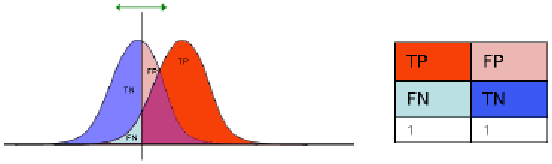
Receiver Operating Characteristics (ROC)

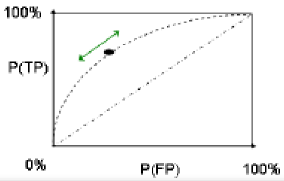
- ROC is a measure of the quality of the goodness of a prediction algorithm.
- The ROC curve is a graphical representation of the performance of a classification model at all thresholds. It has two thresholds: true positive rate & false positive rate.
- ROC exists as a curve because:

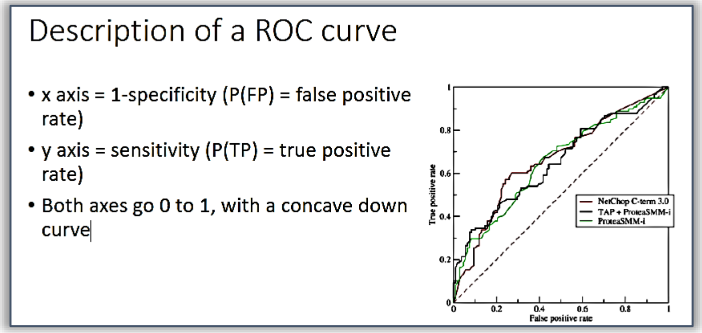
a. in binary classification, you are predicting one of two categories: alive/dead, click on ad/do not click.

b. but predictions are often quantitative: P(alive), prediction on a scale from 1-10.

c. The cut off you choose gives different results.






Description of a ROC curve

- x axis = 1-specificity (P(FP) = false positive rate)
- y axis = sensitivity (P(TP) = true positive rate)
- Both axes go 0 to 1, with a concave down curve

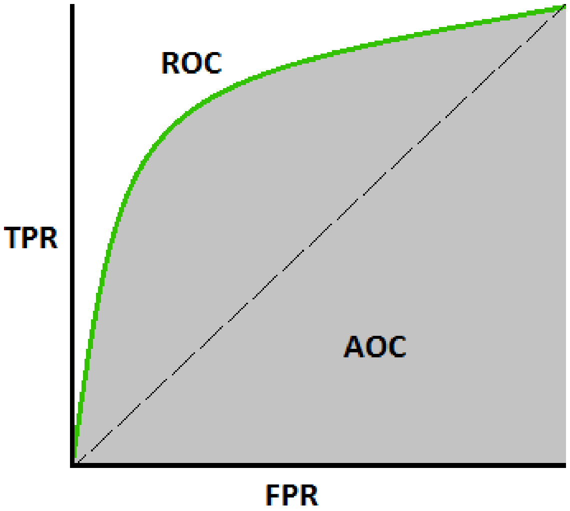
Evaluating Area Under ROC Curve (AUC)

- AUC (Area Under the ROC Curve) is, simply the area under the ROC curve.
- AUC measures the two-dimensional area underneath the ROC curve from (0,0) to (1,1).
- It used as a performance metric for evaluating binary classification models.
- The measurement can be scaled as:

  a. AUC = 0.5: random guessing (45 degree line basically)

  b. AUC < 0.5 means you did worse

  c. AUC = 1: perfect classifier
- In general, AUC above 0.8 is considered ‘good’.



Links to Read

Understanding Multiple Linear Regression

https://medium.com/swlh/understanding-multiple-linear-regression-e0a93327e960

Understanding Logistic Regression!!!

https://medium.com/analytics-vidhya/understanding-logistic-regression-b3c672deac04

Chapter 4: Decision Trees Algorithms

https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1

Naïve Bayes Algorithm

https://medium.com/analytics-vidhya/na%C3%AFve-bayes-algorithm-5bf31e9032a2

Links to Read

Decision Tree Introduction with Example

https://www.geeksforgeeks.org/decision-tree-introduction-example/

Confusion matrix and other metrics in machine learning

https://medium.com/hugo-ferreiras-blog/confusion-matrix-and-other-metrics-in-machine-learning-894688cb1c0a

A Beginner’s Guide to ROC Curves and AUC Metrics

https://medium.com/swlh/a-beginners-guide-to-roc-and-auc-curves-d279c1a5e0e6