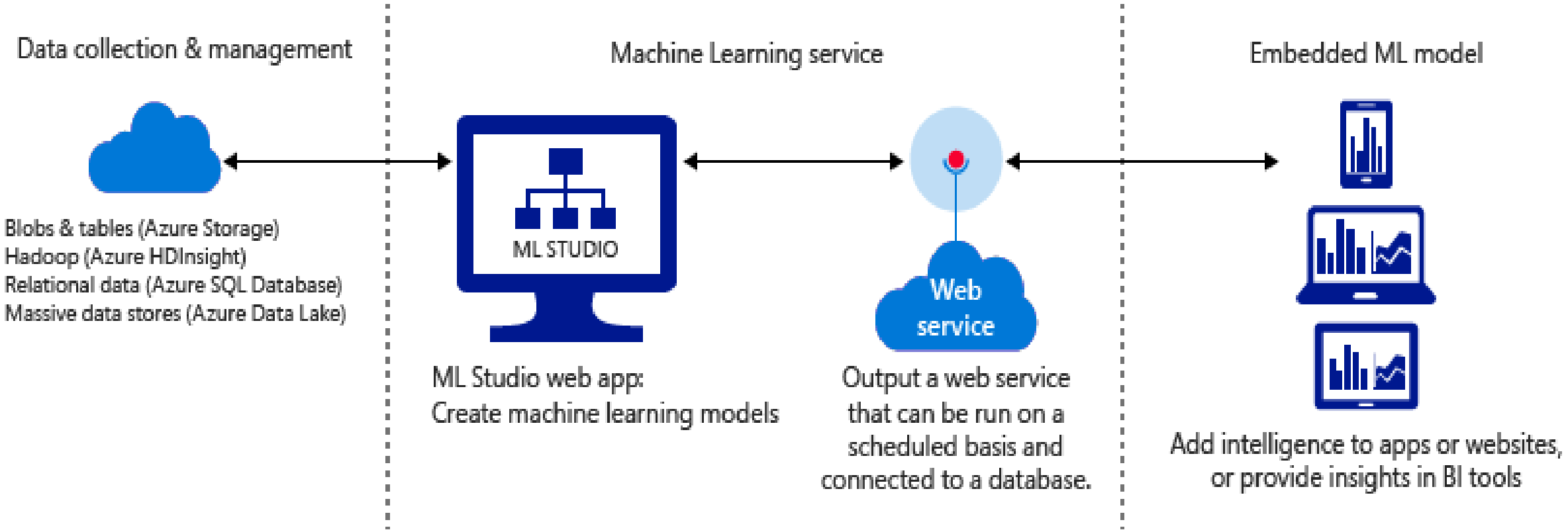
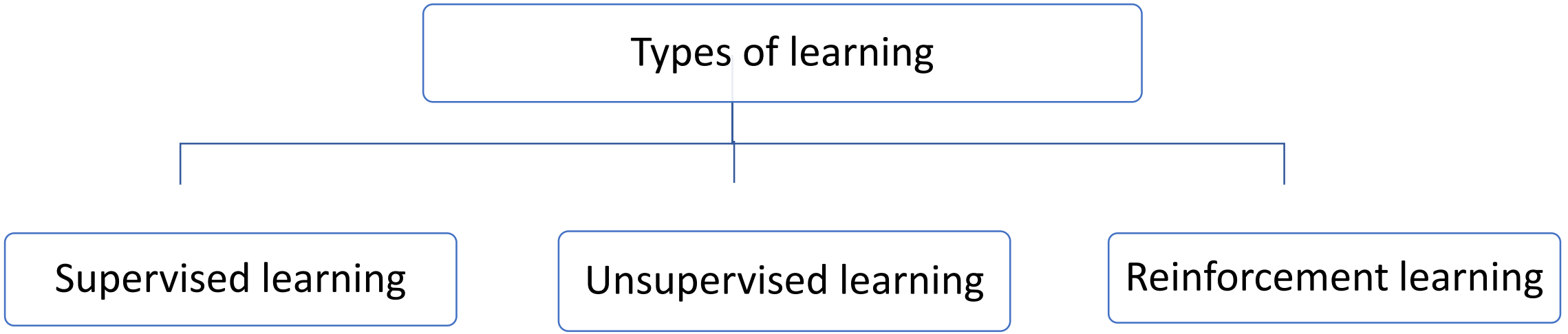
What is Machine Learning

- It is a type of Artificial Intelligence that makes the computers capable of learning on their own i.e. without explicitly being programmed. With machine learning, machines can update their own code, whenever they come across a new situation.



Categories of Algorithm



Categories of Algorithm

Supervised learning

- Supervised learning is a machine learning algorithm that makes use of known(train dataset) to make predictions.
- The training dataset includes input data and response values.
- Supervised learning algorithm seeks to build a model that can make predictions of the response values for a new dataset. A test dataset is often used to validate the model.

Unsupervised learning

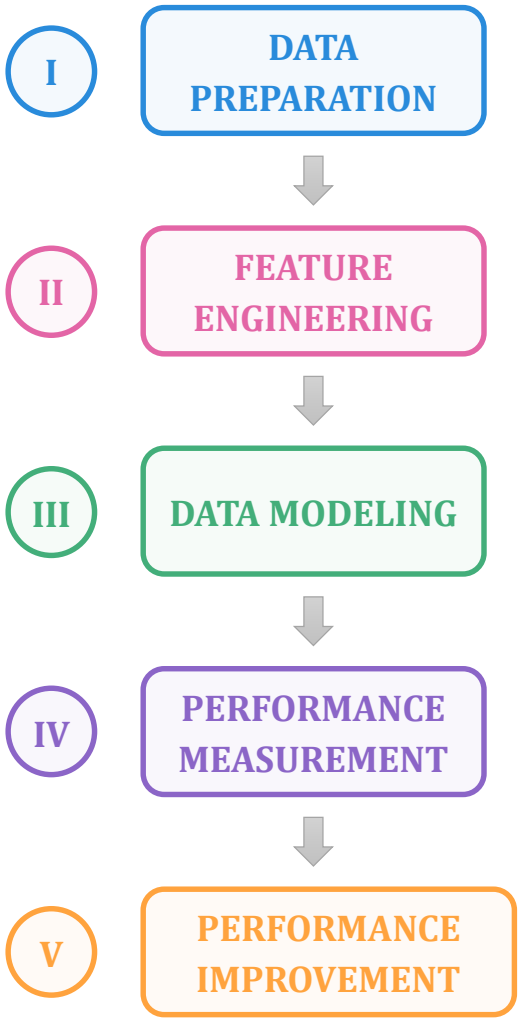
- A Type of Machine Learning algorithm used to draw inferences from datasets consisting of input data.

Reinforcement learning (RL) is an area of machine **learning** inspired by behaviorist psychology, concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward.

Applications

- Game playing,
- Robot in a maze

Steps in Supervised Learning



How can you import your raw data?
What are the most common data cleaning methods?

How do you turn raw data into relevant data, i.e. meaningful for a learning algorithm?
How can you make the difference between useful and useless data in a huge dataset?

What are the different types of machine learning algorithms? (with examples)
Which one should you choose to build your model?

What is the right method to assess the performance of your ML model?
Which indicator should you use?

What are the reasons why your ML model is not performing well?
What are the most common techniques to improve its performance?

Step 1 – Data Preparation

Data preparation typically requires 3 steps

1. Query your data

2. Clean your data
   - Deal with missing values
   - Remove outliers

3. Format your data

Clean your data

a Dealing with missing values

Some of your columns will certainly contain missing values, often as 'NaN'. You will not be able to use algorithms with NaN values.

Compute ratio $\boldsymbol{R_m} = \frac{\text{Number of missing values}}{\text{Total number of values}}$xa

If $\boldsymbol{R_m}$is high, you might want to remove the whole column

If $\boldsymbol{R_m}$ is reasonably low, to avoid losing data, you can impute the mean, the median or the most frequent value in place of the missing values.

b Remove Outliers

Outliers are values that lie at a significantly abnormal distance from other values within your sample set, so you will have to remove them arbitrarily or using robust methods.

Format your data

Your will have to modify your data so that they fit the constraints of algorithms.

The most common transformation is the encoding of categorical variables.

Gender	Purchase
Male	3
Female	1
Female	3
Female	2
Male	3

We replace strings with numbers.

Gender	Purchase
0	3
1	1
1	3
1	2
0	3

Because there is no hierarchical relationship between the 0 and 1, 'Gender' is a "dummy variable". We create a new column for each of the values in the document.

Purchase	Gender_M	Gender_F
3	1	0
1	0	1
3	0	1
2	0	1
3	1	0

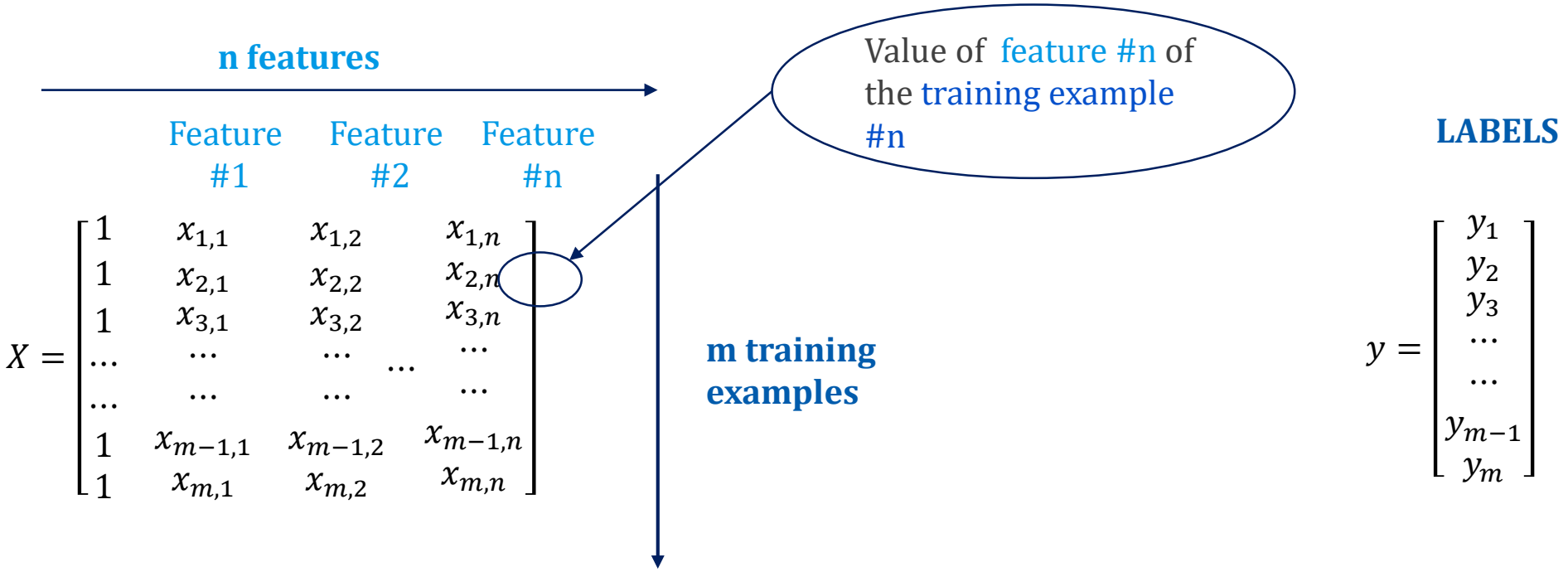
Step 2 - Feature Engineering

Feature engineering is the process of transforming raw data into relevant features, i.e. that are:

- Informative (it provides useful data for your model to correctly predict the label)
- Discriminative (it will help your model make differences among your training examples)
- Non-redundant (it does not say the same thing than another feature)

These features result in improved model performance on unseen data.

After feature engineering, your dataset will be a big matrix of numerical values.



Remember that behind "data" there are two very different notions, training examples and features.

Steps in Feature Engineering

Feature engineering usually includes,

1. Feature Construction
2. Feature Transformation
3. Dimension Reduction
   a. Feature Selection
   b. Feature Extraction

Feature Construction

Feature construction means turning raw data into informative features that best represent the underlying problem and that the algorithm can understand.

Example: Decompose a Date-Time

Same raw data		Different problems		Different features
2017-01-03 15:00:00	→	Predict how much hungry someone is	→	"Hours elapsed since last meal" : 2
2017-01-03 15:00:00	→	Predict the likelihood of a burglary	→	"Night": 0 (numerical value for "False") 1

Feature Transformation

Feature transformation is the process of transforming a feature into a new one with a specific function.

Examples of transformations:

NAME	TRANSFORMATION	OBJECTIVES
Scaling	$x_{new} = \frac{x_{old} - \mu}{\sigma}$	The most important. Many algorithms need feature scaling for faster computations and relevant results, e.g. in dimension reduction
Log	$x_{new} = \log(x_{old})$	Reduce heteroscedasticity, which can be an issue for some algorithms

Dimension Reduction

Dimension reduction is the process of reducing the number of features used to build the model, with the goal of keeping only informative, discriminative non-redundant features.

The main benefits are:

- Faster computations
- Less storage space required
- Increased model performance
- Data visualization (when reduced to 2D or 3D)

Dimension Reduction involves two steps:

- Feature Selection
- Feature Extraction

Dimension Reduction – Feature Selection

Feature selection is the process of selecting the most relevant features among your existing features.

To keep "relevant" features only, we will remove features that are:

i. Non informative
ii. Non discriminative
iii. Redundant

Methods:

1. Recursive Feature Elimination (RFE) (among others)

   Principle: We eliminate one feature at a time, run the model each time and note impact on the performance of the model.

2. Variance threshold filter

   Principle: We remove any feature whose values are close across all the different training examples (i.e. that have low variance)

Dimension Reduction – Feature Extraction

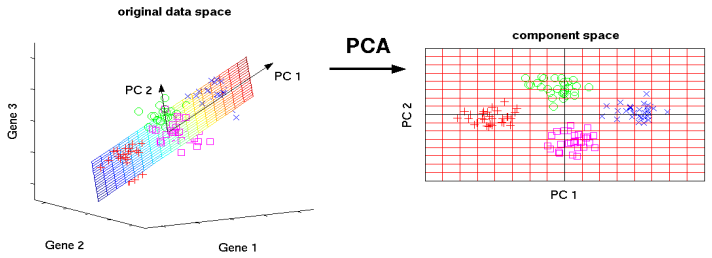
What happens when a data set has too many variables ?

-Most of the variables may be correlated.

-Build a model on whole data and that will result poor accuracy

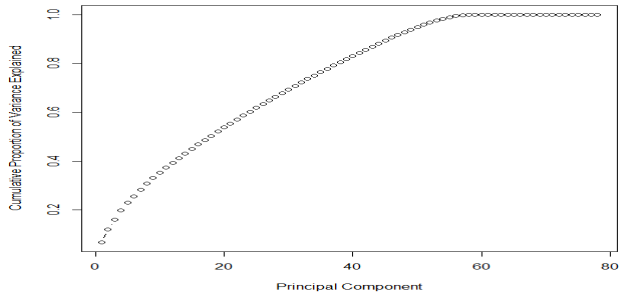
Principal Component Analysis(PCA) is a method of extracting important variables (in form of components) from a large set of variables available in a data set.

A **principal component** is a normalized linear combination of the original predictors in a data set.



PC1 and PC2 are the principal components.

- **First principal component(PC1)** is a linear combination of original predictor variables which captures the maximum variance in the data set.
- **Second principal component(PC2)** is also a linear combination of original predictors which captures the remaining variance in the data set.



The plot above shows that ~ 51 components explains around 98% variance in the data set. Using PCA, we have reduced 78 predictors to 51 without compromising on explained variance

Step 3 – Model Building

You are going to train a model on your data using machine learning algorithm

Remember

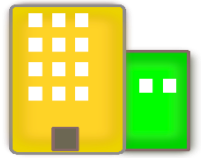
DATA **ALGORITHMS** **MODEL**

Supervised Learning

Supervised learning algorithms are used to build two different kind of models.

a **Regression**

When the label to predict is a continuous value
Example: Predict the price of an apartment



b **Classification**

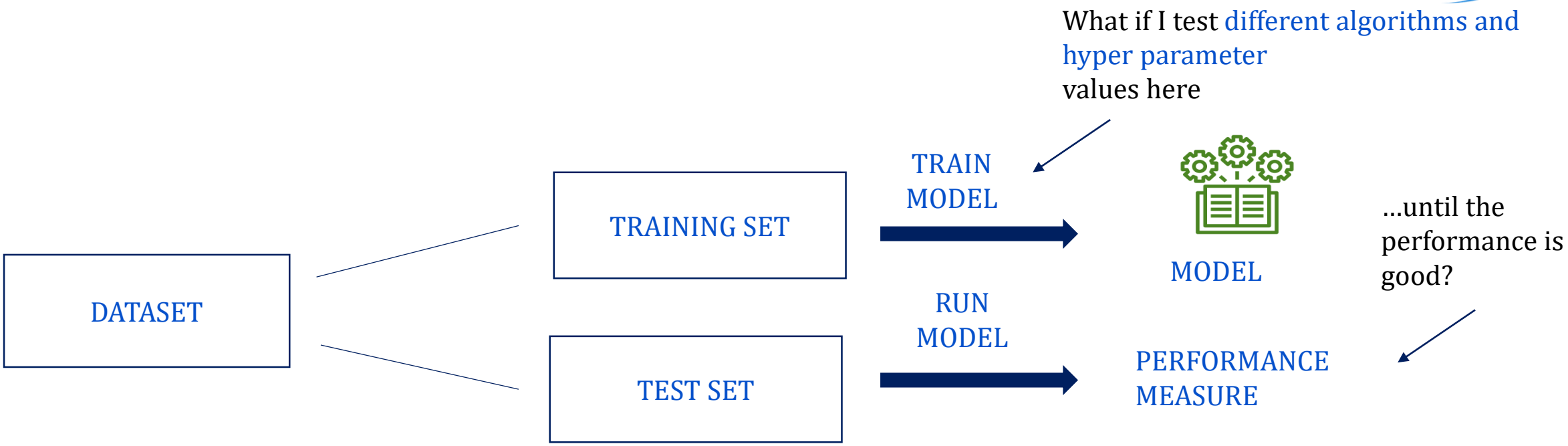
When the label to predict is a discrete value
Example: Predict how many stars I am going to rate a movie on Rotten tomatoes(0,1,2,3,4,5)

Step 4 – Performance Measurement

Assessing your model performance is a 2-step process

1. Use your model to predict the labels in your own dataset
    a. Training set and test set
    b. Choosing the right performance indicator
2. Use some indicator to compare the predicted values with the real values
    a. Regression
    b. Classification

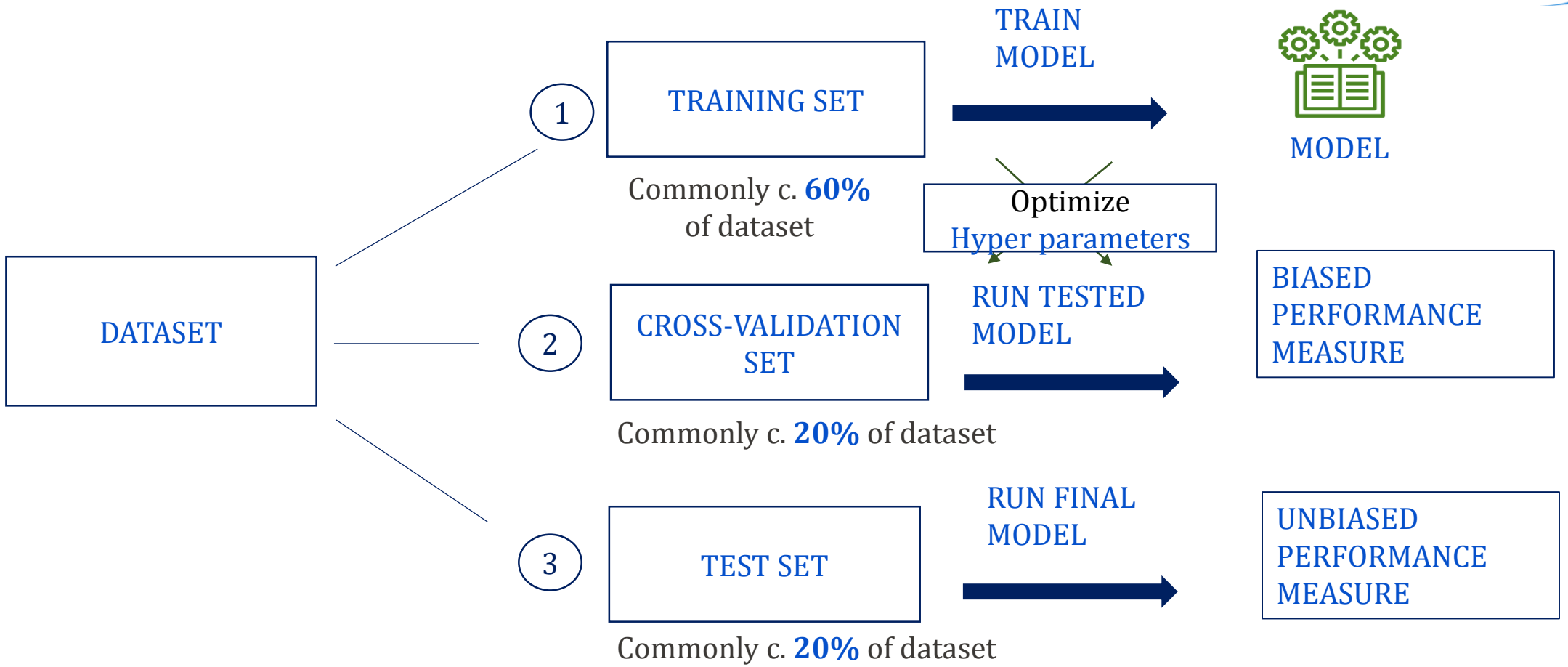
Training & Test Set



Such a method is often used to quickly test different algorithms at the beginning or to fine-tune hyper parameters

However, the performance measure will be biased again, because it highly depends on the data in the test set, which is why you will have to use cross-validation. 1 Training set and test seta

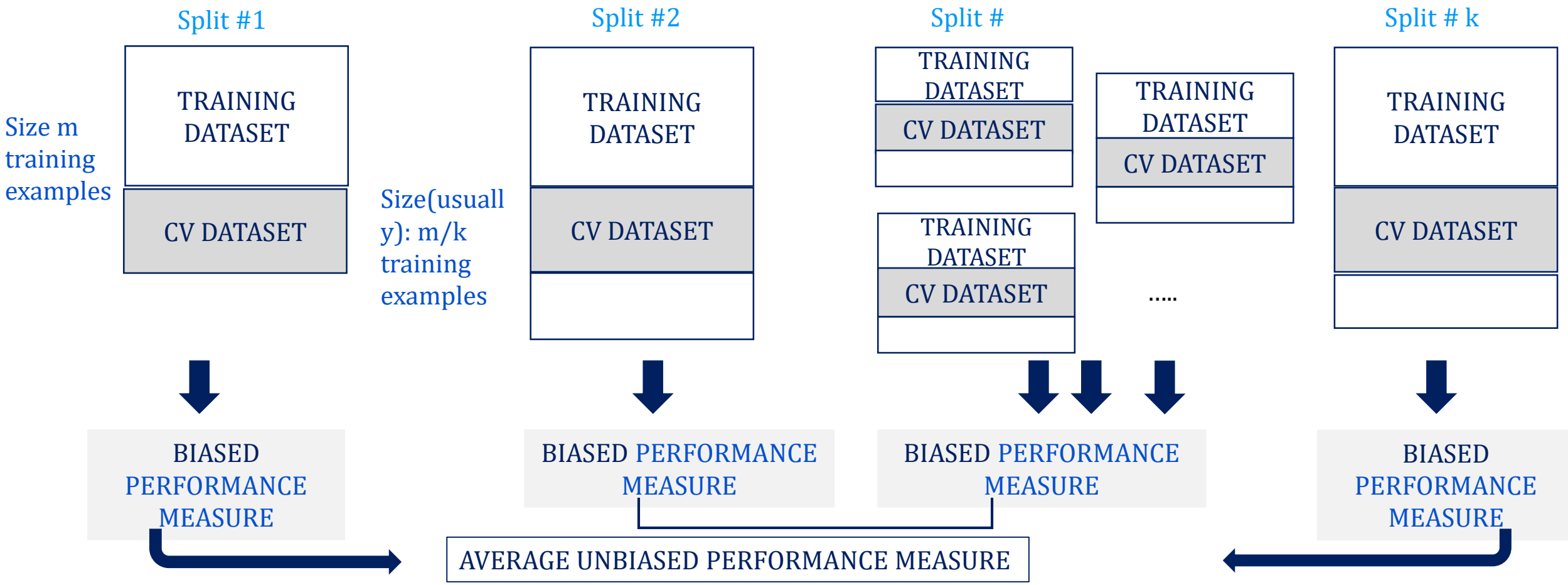
Cross Validation



PROBLEM: You will **lose c. 20% of the data** to train your algorithm. If you don't have lots of data, you might prefer KFold cross-validation

K fold Cross Validation

K Fold cross-validation consists in repeating the training / CV random splitting process K times to come up with an average performance measure.



Choosing the right Performance Indicator - Regression

y_i Is the true label for the i-th example in the test data

y_i Is the predicted label for the i-th example in the test data

$\bar{y}$ Is the average label for the i-th example in the test data

2 commonly used indicators	**Mean squared error**	**Coefficient of determination (R^2)**
Formula	$\mathrm{MSE}(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} (y_i - \hat{y_i})^2$	$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{n_{samples}-1} (y_i - \hat{y_i})^2}{\sum_{i=0}^{n_{samples}-1} (y_i - \bar{y})^2}$
Pros	Easy to understand	Absolute Value Very roughly, a model with $R^2 > 0.6$ is getting good (1 being the best), $R^2 < 0.6$ is not so good
Cons	Relative value You need the scale of your labels to interpret MSE	Difficult to explain

Choosing the right Performance Indicator - Classification

$$\text{Accuracy} = \frac{\text{Number ofcorrectly predicted labels}}{\text{Total no. of lables in test set}}$$

Accuracy is very easy to understand but often too simple to correctly interpret the performance of your model

Confusion matrix

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision is a % expressing the **precision with which the positive values where recalled** by your model.

ACTUAL LABELS	PREDICTED LABELS: POSITIVE	PREDICTED LABELS: NEGATIVE
ACTUAL LABELS: POSITIVE	✓ TRUE POSITIVE (TP)	✗ FALSE NEGATIVE (FN)
ACTUAL LABELS: NEGATIVE	✗ FALSE POSITIVE (FP)	✓ TRUE NEGATIVE (TN)

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall is a % expressing the **capacity of your model to recall positive values**.

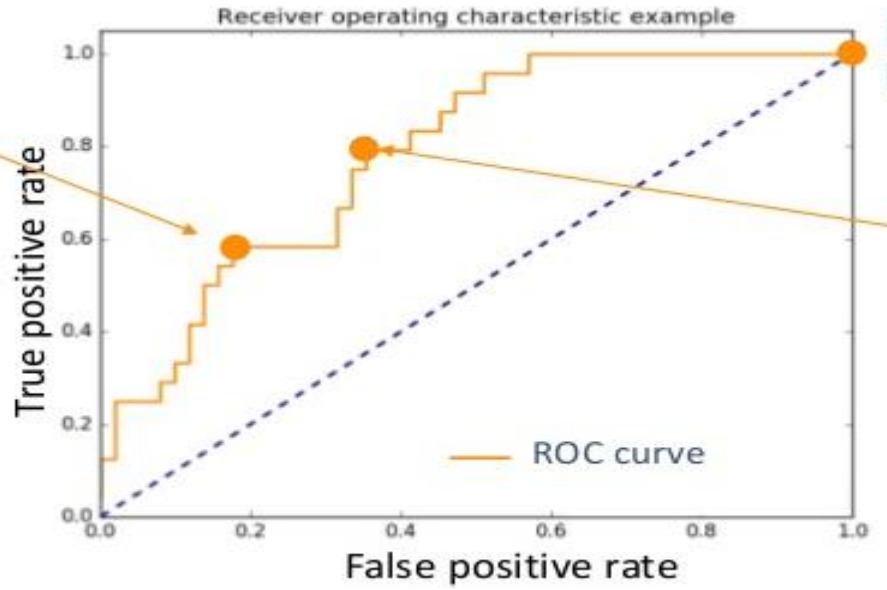
Choosing the right Performance Indicator - Classification

ROC curve

There is always a trade-off in function of whether you want to prefer precision or recall. You can visualize how the TP and FP rates evolve according to different discrimination thresholds of your model with the **ROC curve**.

Precision > Recall

If you are running a marketing campaign but don't have too much money, you might want to focus on a smaller target (low recall) where your probability to convert is high (high precision)



Recall = 100%
Precision = % of positive examples in your dataset

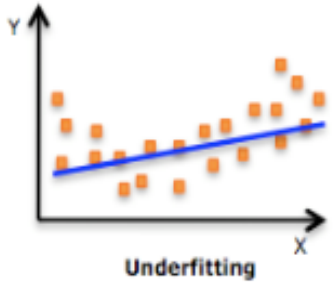
Recall > Precision

If you are running a marketing campaign and have lots of budget, you will rather focus on a large target where your probability to convert is lower (low precision), but on a greater number of people (high recall)

Step 5 – Performance Improvement

Reasons for underperformance

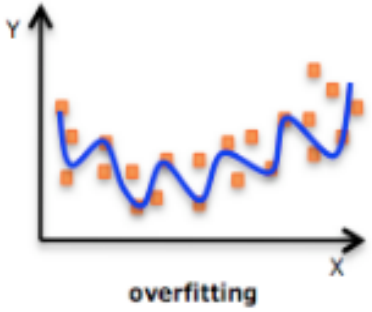
Under-fitting



Underfitting happens when your model is too simple to reproduce the underlying data structure.

When underfitting, a model is said to have high bias.

Over-fitting

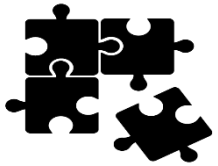
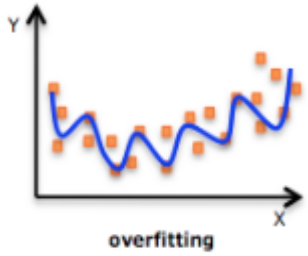
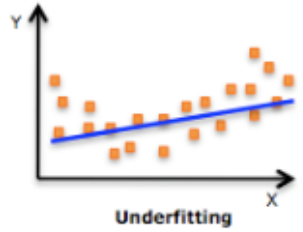


Overfitting happens when your model is too complex to reproduce the underlying data structure. It captures the random noise in the data, whereas it shouldn't. When overfitting, a model is said to have high variance.

Step 5 – Performance Improvement

Issue of the

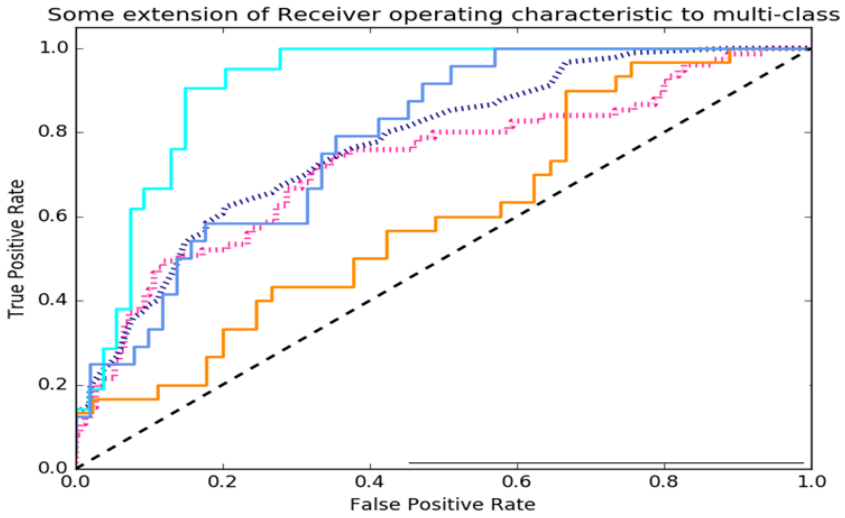
Potential solutions on

 MODEL			
 DATA	A	More training examples	
	B	a Less features	b More Features
 ALGORITHMS	C	Simple Algorithms	More complex Algorithm
	D	Regularization	
	E	Bagging	F Boosting

"ENSEMBLE METHODS"

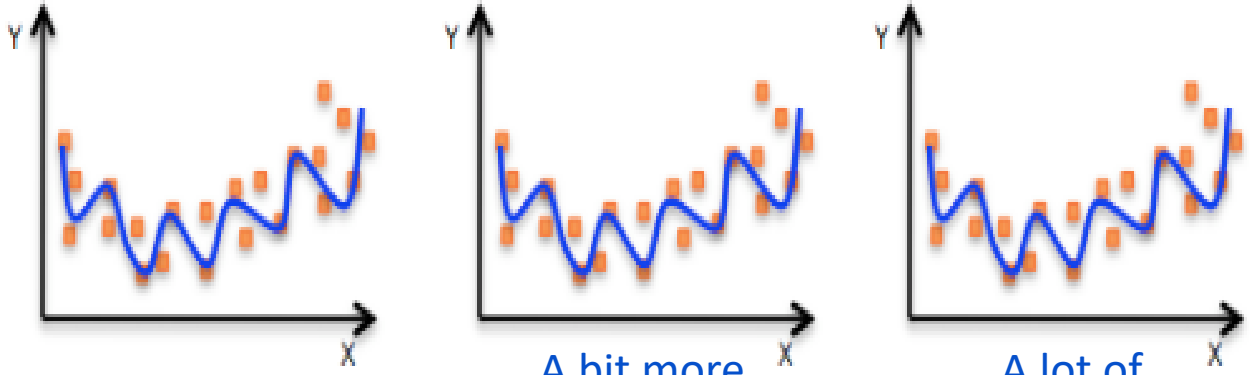
More Training Examples

The study below shows how different algorithms perform similarly for a given problem as the amount of training examples increases.



The more training examples there are, the more complex it is for an algorithm to fit the noise in the data.

Therefore, the fitted model will be less sensitive to noise and will better generalize.

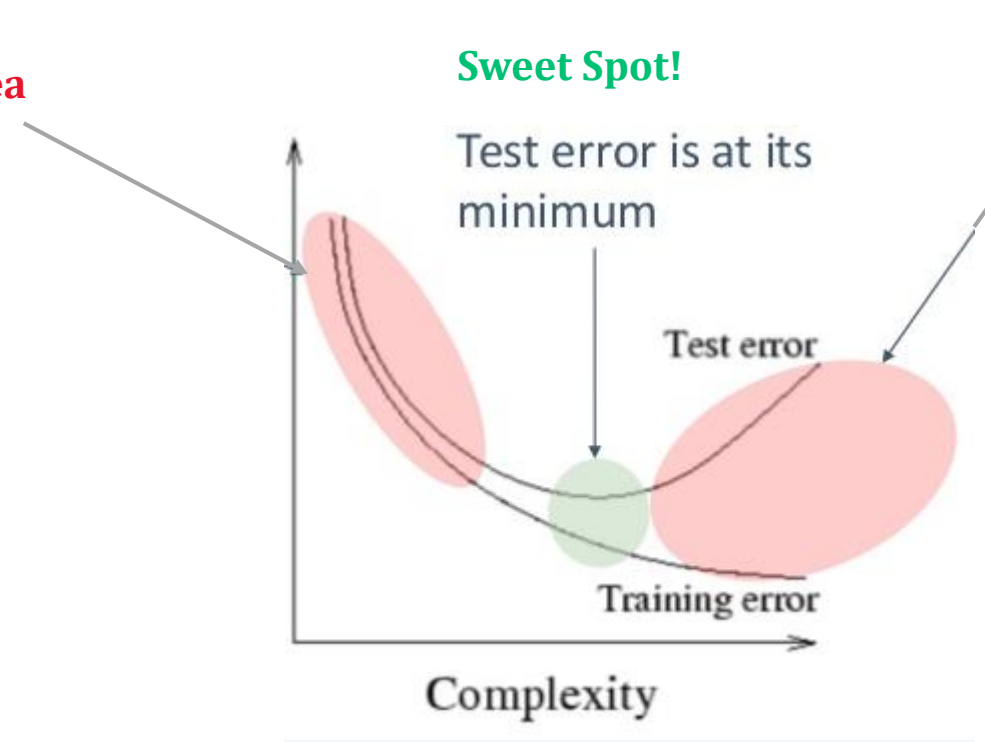


Algorithm Complexity

Below demonstrates how *a* model error will theoretically evolve as its complexity increases (i.e. more complex algorithms, more features).

Underfitting area

In addition/place of more features, you might need more complex algorithms (nonlinear, nonparametric) to model your data structure.



Overfitting area

In addition/place of less features, you might need simpler algorithms. Certain algorithms, such as deep neural networks, are very powerful and easily tend to over fit if you don't have millions of rows of data.

Solutions to Increase Performance – Get More Data

Data is key.

We must know how to make good use of these data with good feature engineering.

Data can help you to:

- Reduce variance (overfitting) with more training examples
- Reduce bias (underfitting) with more features More details

Introduction to Regression

Regression is a tool for finding a relationship between a dependent variable and one or more independent variables in a study.

- Linear Regression, also called Ordinary Least-Squares (OLS) Regression, is probably the most commonly used technique in Statistical Learning.
- It is also the oldest, dating back to the eighteenth century and the work of Carl Friedrich Gauss and Adrien-Marie Legendre
- The relationship can be linear or non-linear.

The basic function of regression is to identify statistically significant independent variables and estimate the model parameters.

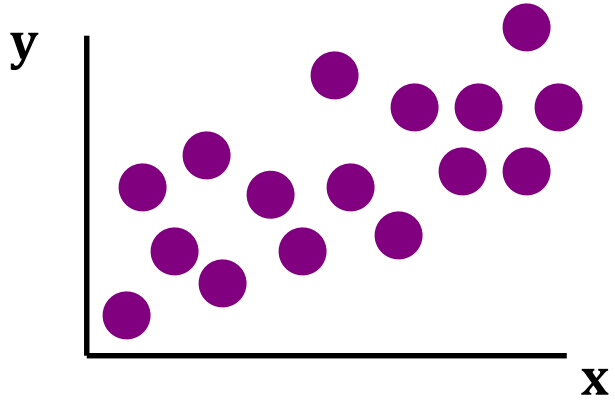
Scatter Plots and Correlation

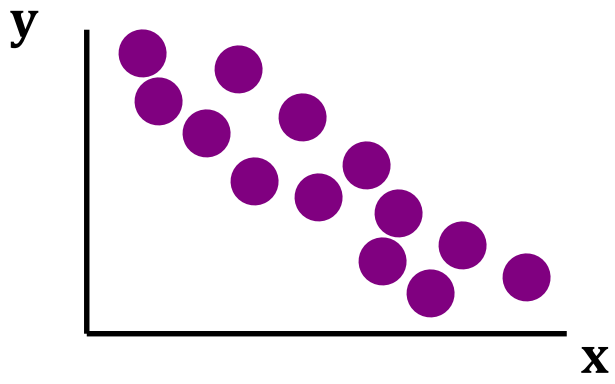
- Correlation analysis is used to measure strength of the association (linear relationship) between two variables
  - Only concerned with strength of the relationship
  - No causal effect is implied

- A scatter plot (or scatter diagram) is used to show the relationship between two variables

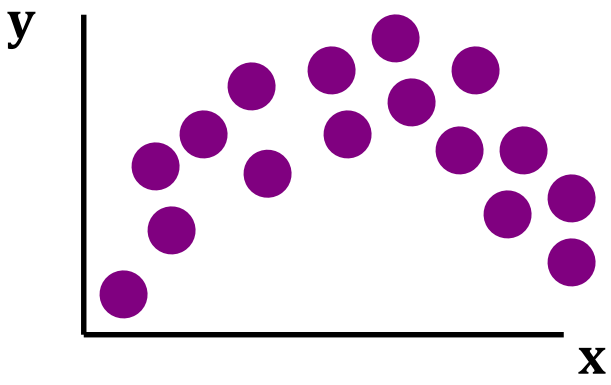
Scatter Plot Examples

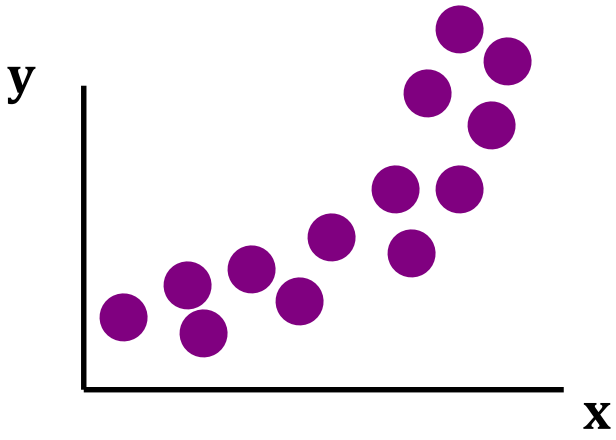
Linear relationships





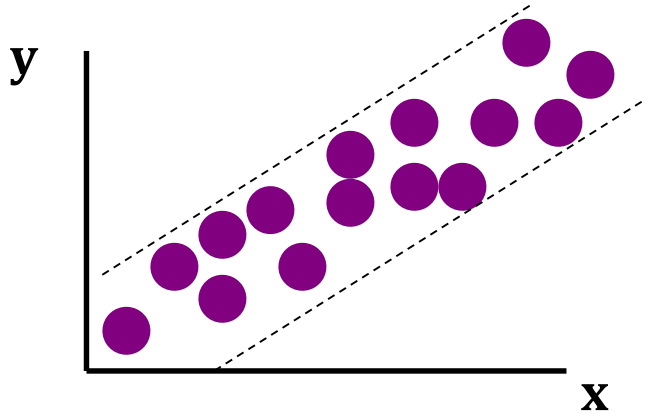
Curvilinear relationships

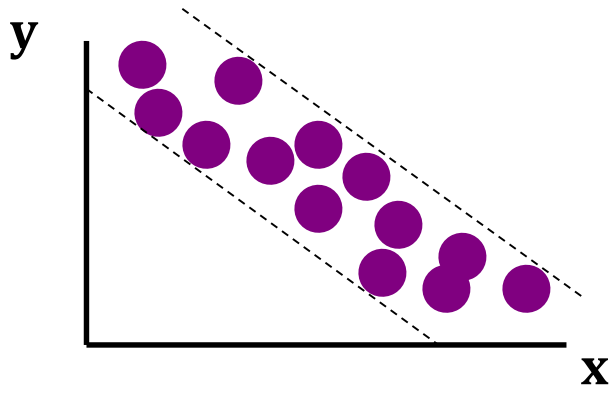




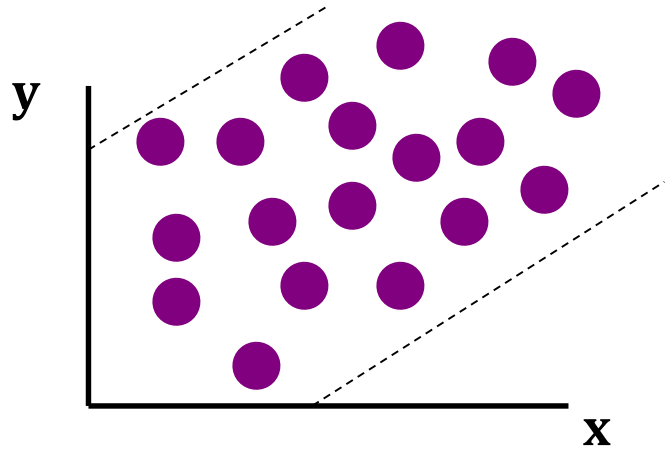
Scatter Plot Examples (Contd.)

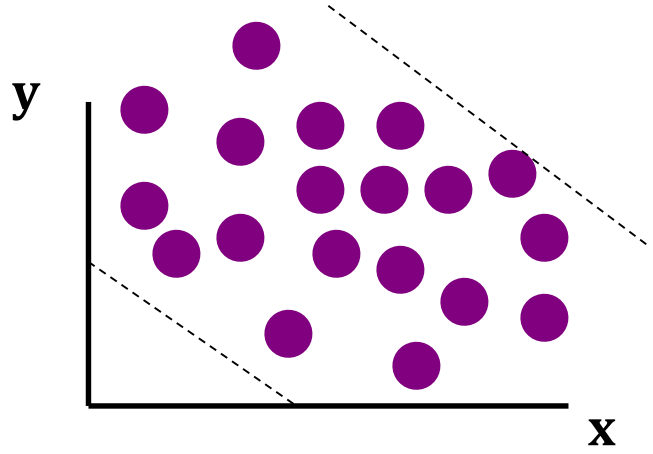
Strong relationships



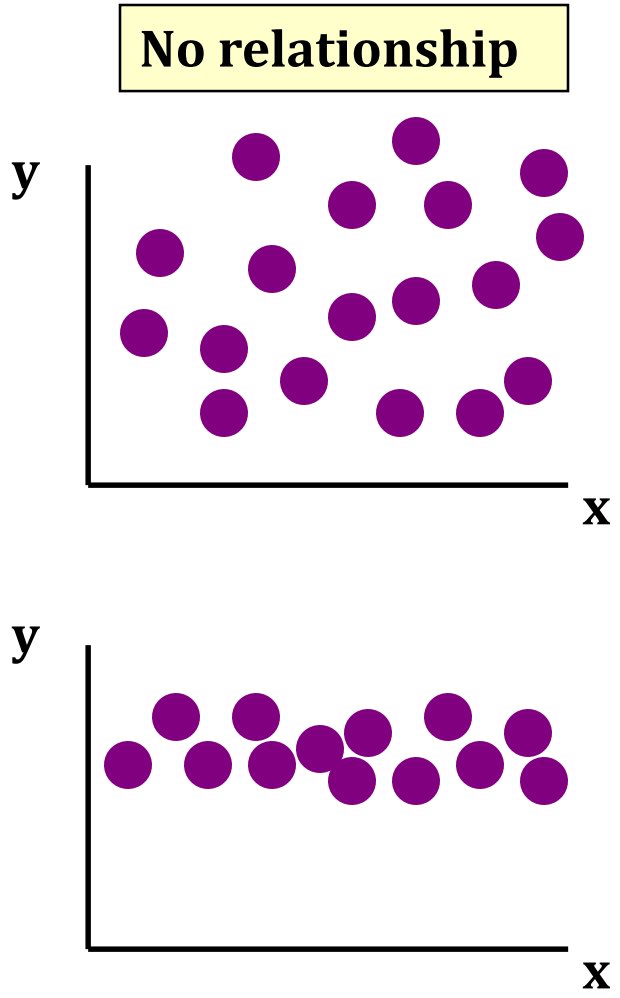


Weak relationships





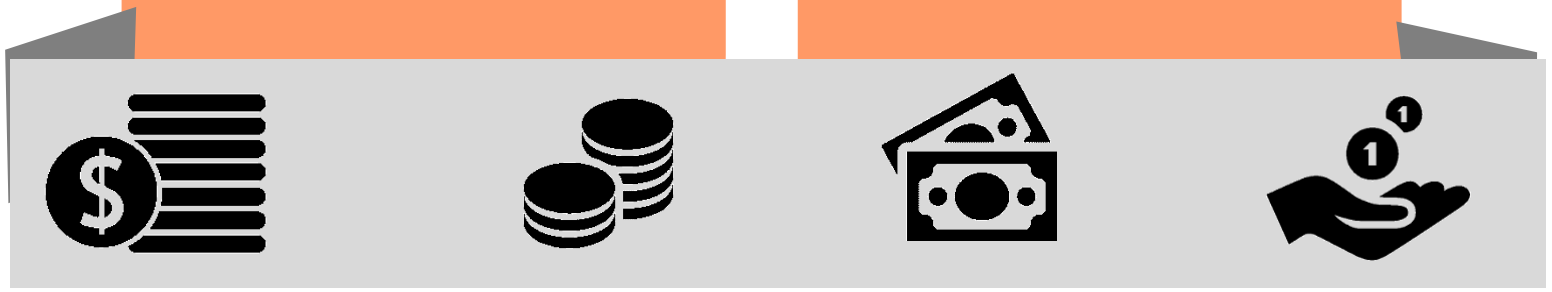
Scatter Plot Examples (Contd.)



Correlation Coefficient

The population correlation coefficient ρ (rho) measures the strength of the association between the variables.

The sample correlation coefficient r is an estimate of ρ and is used to measure the strength of the linear relationship in the sample observations.



Correlation Coefficient

Sample correlation coefficient:

$$r = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sqrt{[\sum (x - \bar{x})^2][\sum (y - \bar{y})^2]}}$$

or the algebraic equivalent:

$$r = \frac{n\sum xy - \sum x \sum y}{\sqrt{[n(\sum x^2) - (\sum x)^2][n(\sum y^2) - (\sum y)^2]}}$$

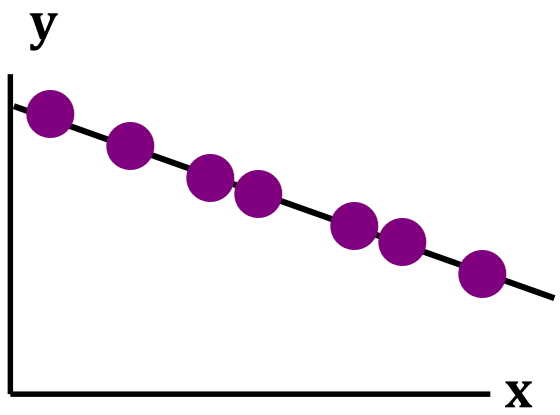
Where:

r = Sample correlation coefficient
n = Sample size
x = Value of the independent variable
y = Value of the dependent variable

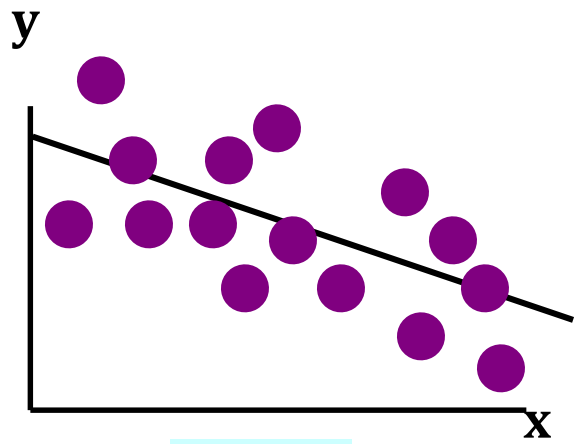
Features of ρ and r

- Unit free
- Range between -1 and 1
- The closer to -1, the stronger the negative linear relationship
- The closer to 1, the stronger the positive linear relationship
- The closer to 0, the weaker the linear relationship

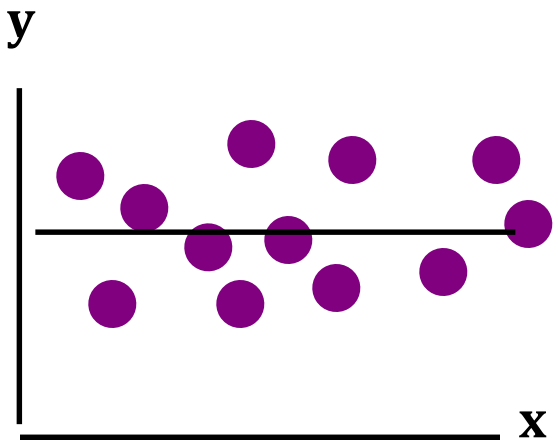
Examples of Approximate r Values



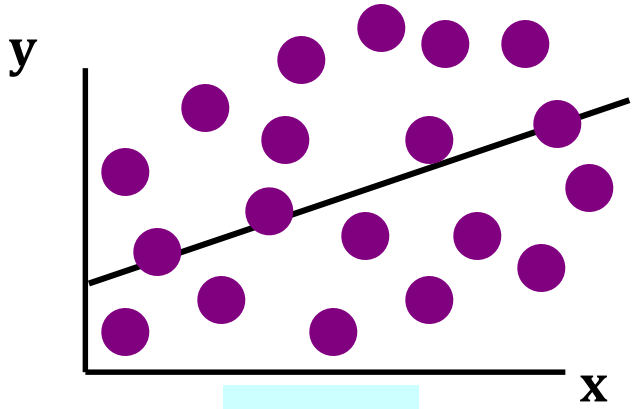
r = -1



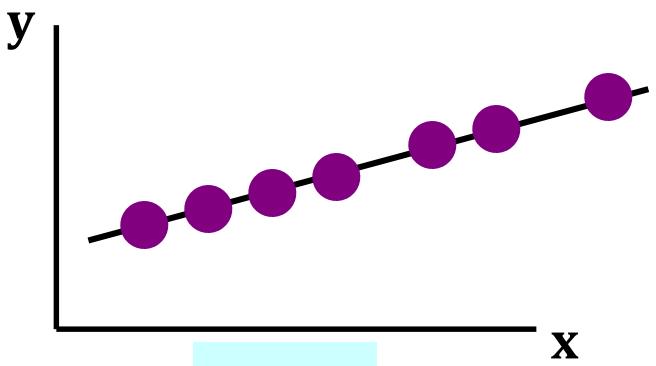
r = -.6



r = 0



r = +.3

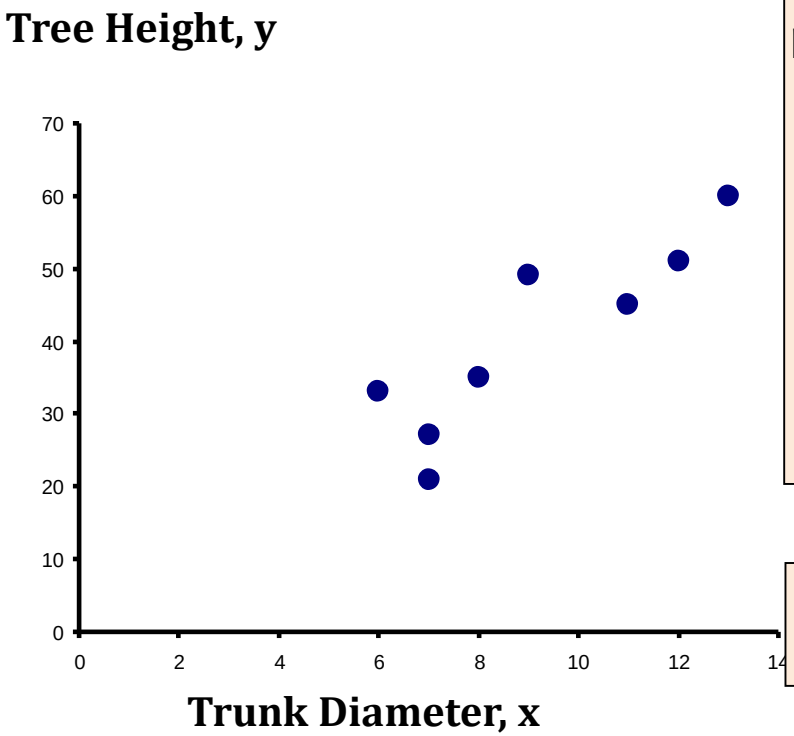


r = +1

Calculation Example

Tree Height	Trunk Diameter			
y	x	xy	y^2	x^2
35	8	280	1225	64
49	9	441	2401	81
27	7	189	729	49
33	6	198	1089	36
60	13	780	3600	169
21	7	147	441	49
45	11	495	2025	121
51	12	612	2601	144
Σ=321	Σ=73	Σ=3142	Σ=14111	Σ=713

Calculation Example



$$r = \frac{n\sum xy - \sum x \sum y}{\sqrt{[n(\sum x^2) - (\sum x)^2][n(\sum y^2) - (\sum y)^2]}}$$

$$= \frac{8(3142) - (73)(321)}{\sqrt{[8(713) - (73)^2][8(14111) - (321)^2]}}$$

$$= 0.886$$

r = 0.886 → relatively strong positive linear association between x and y

Output

Excel Correlation Output

Tools / data analysis / correlation

	Tree Height	Trunk Diameter
Tree Height	1	
Trunk Diameter	0.886231	1

Correlation between

Tree Height and Trunk Diameter

Significance Test for Correlation

- Hypotheses

$H_0: \rho = 0$ (no correlation)

$H_A: \rho \neq 0$ (correlation exists)

- Test statistic (with n – 2 degrees of freedom)

$$t = \frac{r}{\sqrt{\frac{1 - r^2}{n - 2}}}$$

Example

Is there evidence of a linear relationship between tree height and trunk diameter at the .05 level of significance?

H_0: $\rho = 0$ (No correlation)

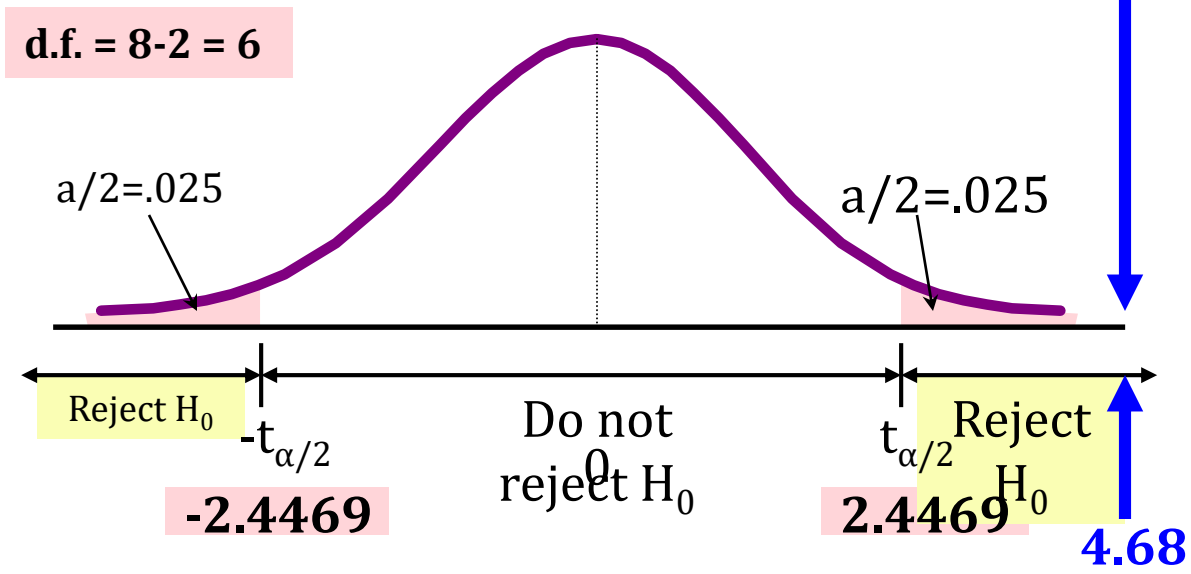
H_1: $\rho \neq 0$ (correlation exists)

$\alpha = .05$, df = 8 - 2 = 6

$$t = \frac{r}{\sqrt{\frac{1-r^2}{n-2}}} = \frac{.886}{\sqrt{\frac{1-.886^2}{8-2}}} = 4.68$$

Example: Test Solution

$$t = \frac{r}{\sqrt{\frac{1-r^2}{n-2}}} = \frac{.886}{\sqrt{\frac{1-.886^2}{8-2}}} = 4.68$$



Decision:
Reject H_0

Conclusion:
There **is evidence** of a linear relationship at the 5% level of significance

Why do Regression Analysis?

Impact of food label on purchase decision?

Which promotion is more effective?

What is the risk associated with a customer?

Which customer is likely to default?

What percentage of loans is likely to result in a loss?

How to identify the most profitable customer?

Regression Analysis

Example

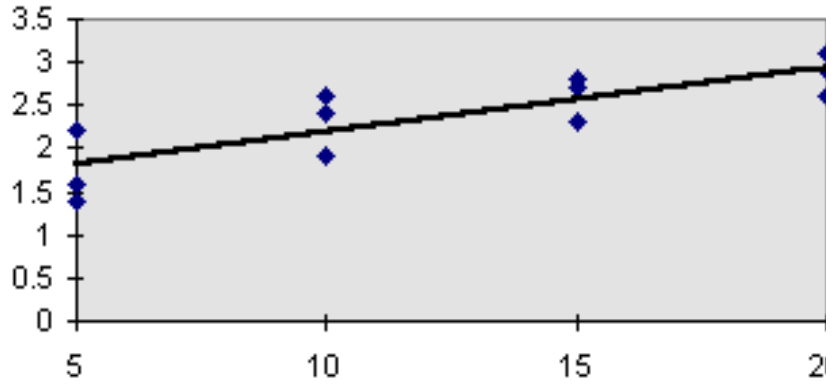
- What's the relationship between exercise duration and calories burned? Is it linear or curvilinear?
  For example, does exercise have less impact on the number of calories burned after a certain point?
- How does effort (the percentage of time at the target heart rate, the average walking speed) factor in?
- Are these relationships the same for young and old, male and female, heavy and slim?

Where is it used?

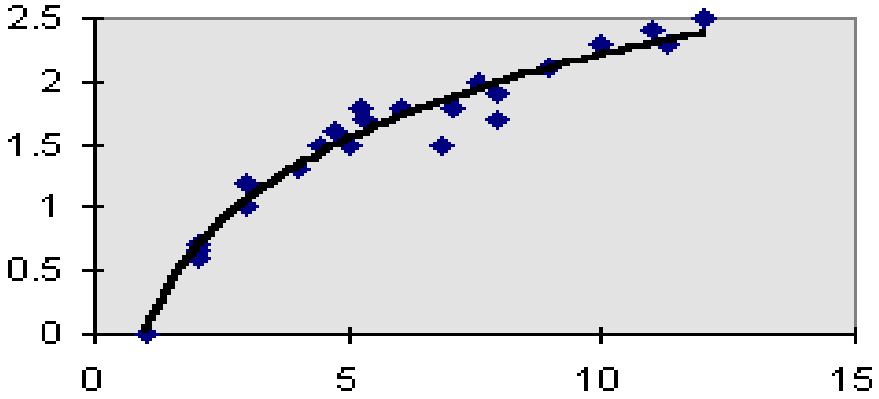
- Every functional area of management uses regression.
- Finance: CAPM, Chance of bankruptcy, credit risk.
- Marketing: Sales, market share, customer satisfaction, customer churn, customer retention, customer life time value.
- Operations: Inventory, productivity, efficiency.
- HR: Job satisfaction, attrition.

Types of Regression Models

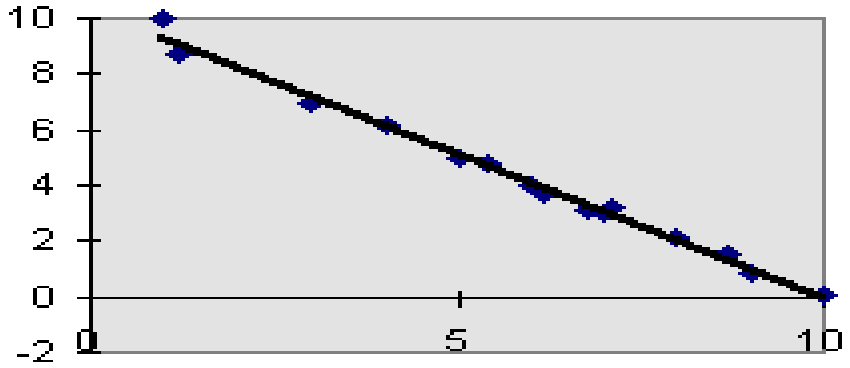
Positive Linear Relationship



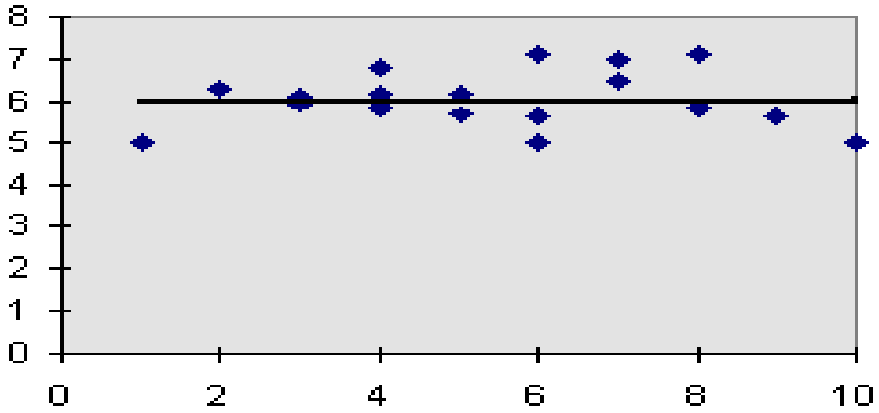
Relationship NOT Linear



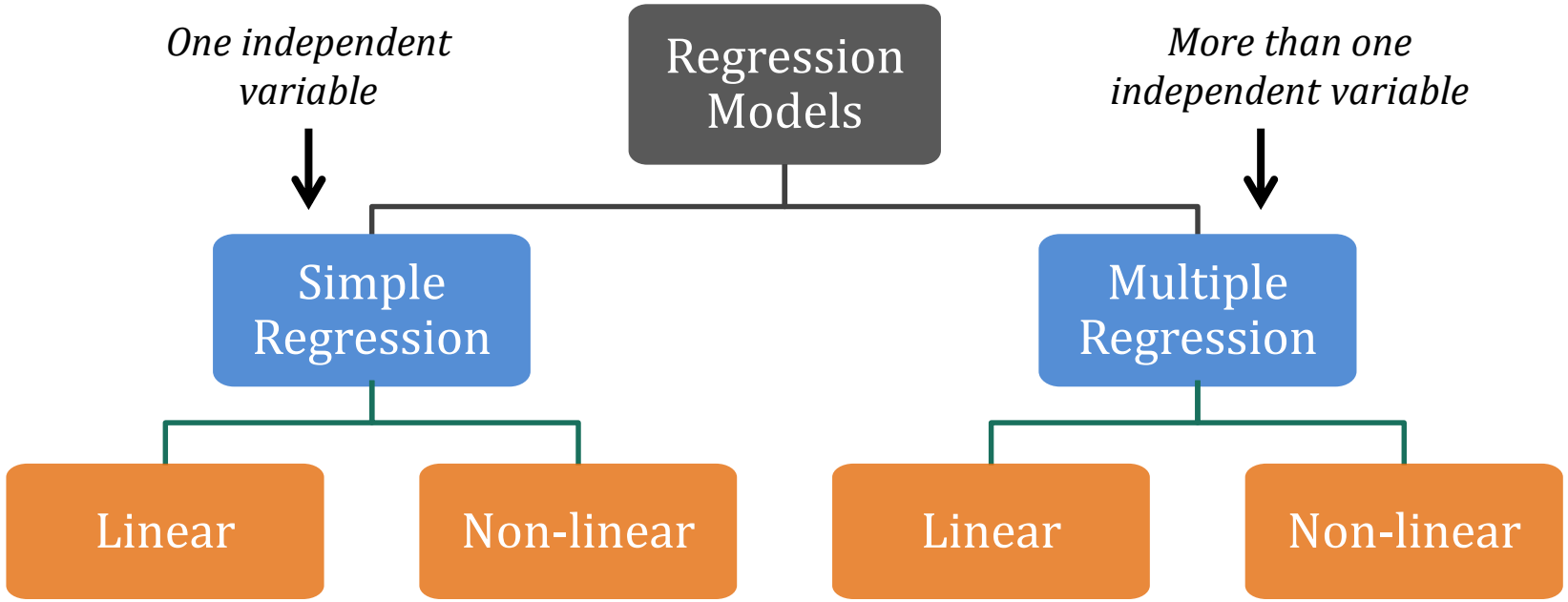
Negative Linear Relationship



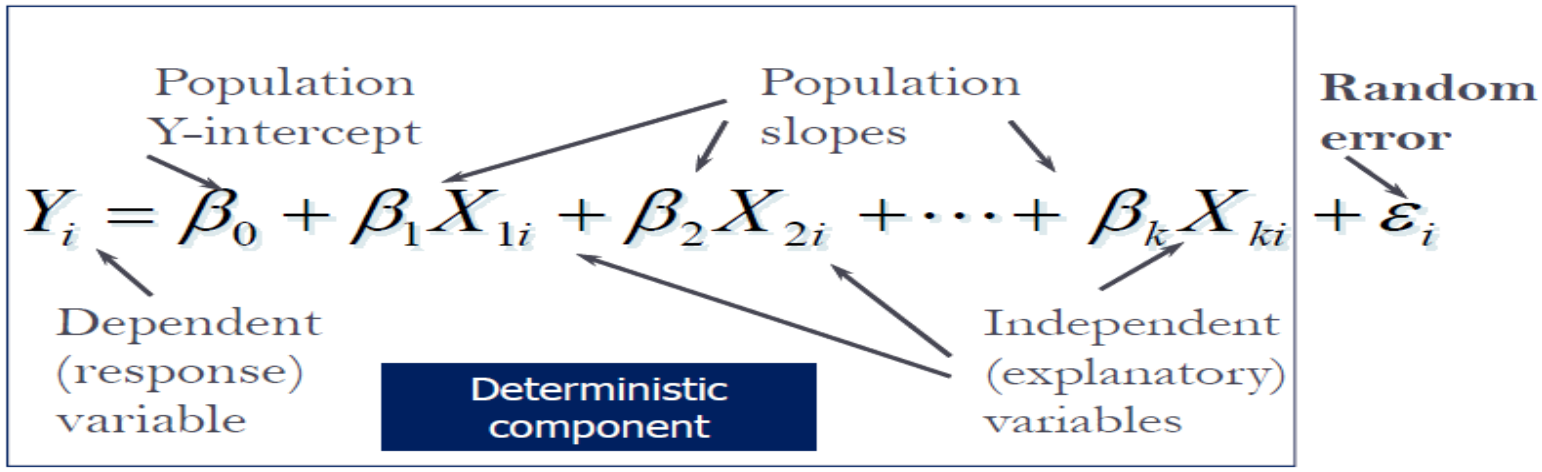
No Relationship



Types of Regression



Ordinary Least Squares (OLS) Regression



$\hat{Y}_i$ is the predicted value of the dependent variable for observation i (specifically, it's the estimated mean of the Y distribution, conditional on the set of predictor values)

X_{1i} is the lth predictor value for the ith observation

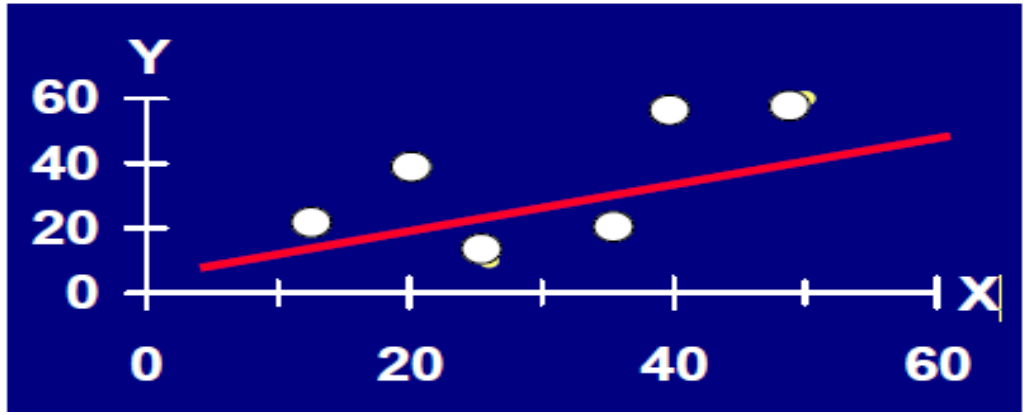
$\hat{\beta}_0$ is the intercept (the predicted value of Y when all the predictor variables equal zero)

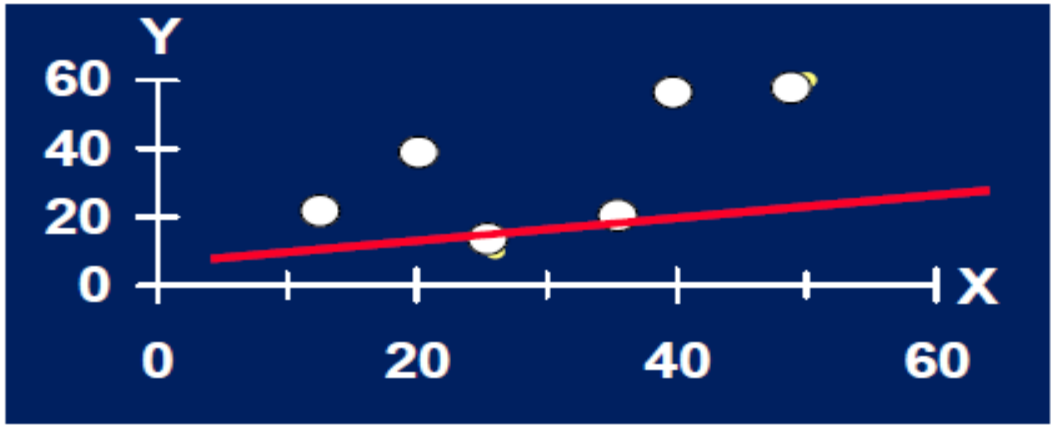
$\hat{\beta}_1$ is the regression coefficient for the lth predictor

Our goal is to select model parameters (intercept and slopes) that minimize the difference between actual response values and those predicted by the model. Specifically, model parameters are selected to minimize the sum of squared residuals

OLS Regression – What Is The Best Fit?

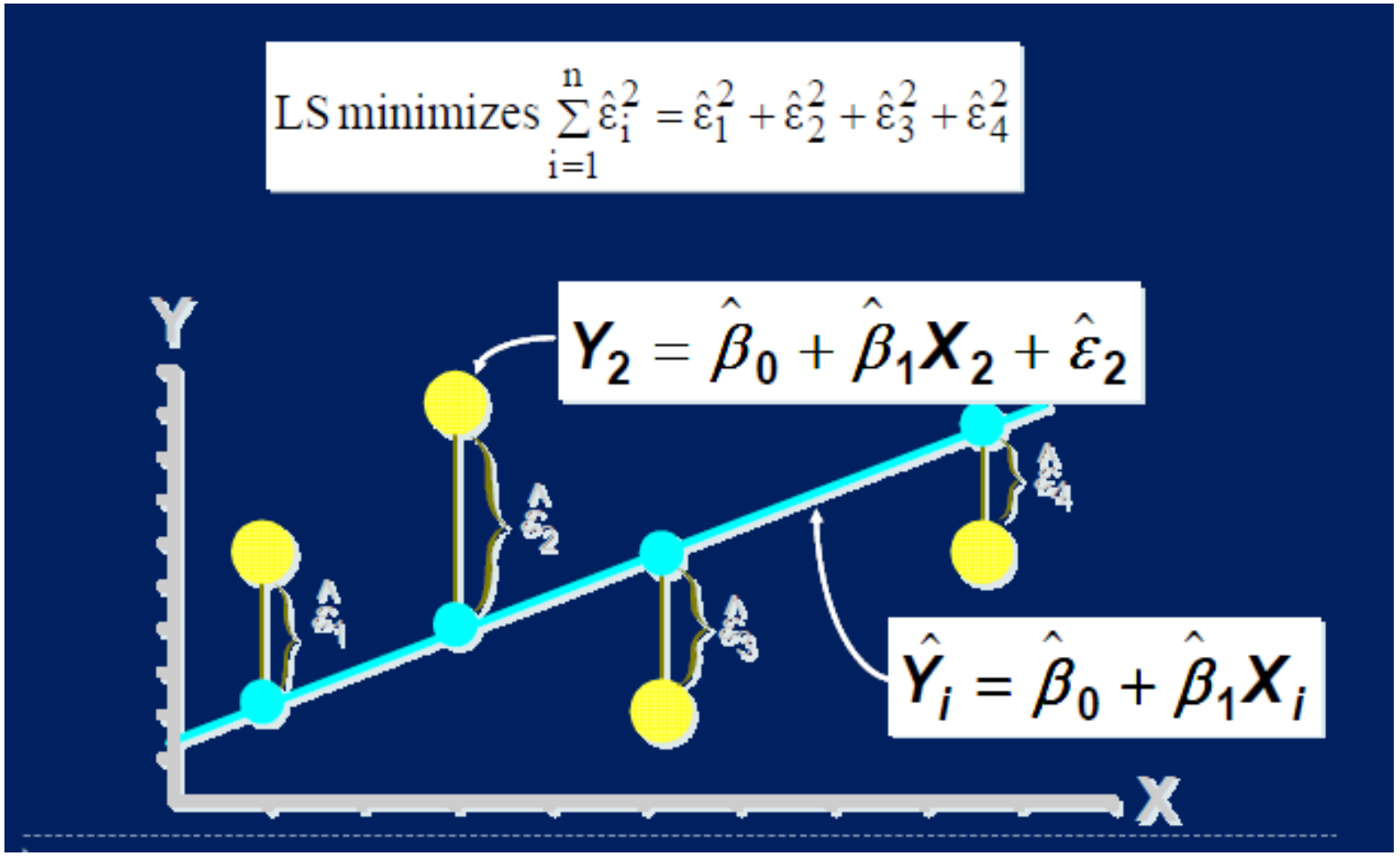
- How would you draw a line through the points?
- How do you determine which line fits best?





OLS Regression – Least Squares

Method of Least Squares



Dependent Variable(s) - Examples

- **Dependent variable** is also called the response variable, and is the output of a process or statistical analysis.
- The values that result from the independent variables.

Dependent variable can be continuous, discrete or categorical.

Example - I

Sales as a dependent variable can be looked at in many ways, such as sales of a specific doll, sales of a category like toy cars, overall sales at a particular store, or even sales for the entire company

Example - II

A credit card company receives thousands of applications for new cards. It has to decide whether an application should approved, or to classify applications into two categories, approved and denied

Independent Variable(s) - Examples

Independent variables

- The values that can be changed in a given model or equation.
- They provide the "input" which is modified by the model to change the "output.“

Example

When the dependent variable is sales revenue, the elements of the marketing mix -- product, price, promotion and place -- will definitely influence the dependent variable and can therefore be identified as independent variables

Example

Each credit card application contains information about an applicant, age, Marital status, annual salary, outstanding debts, credit rating etc.

Least Squares Criterion

- Let $e_i = (y_i - \hat{y}_i)$ be the prediction error for observation i.
- Sum of Squares of Errors, $SSE = \sum_{i=1}^{n} e_i^2$
- For good fit, SSE should be minimum, that is "Least Squares"

$$SSE = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$$\phi(\beta_o, \beta_1) = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y - (\beta_o + \beta_1 x))^2$$

Minimization for Least Squares Criterion

- b_0 and b_1 are obtained by finding the values of b_0 and b_1 that minimize the sum of the squared residuals
- From calculus we know,

For minimum $\phi(\beta_0, \beta_1)$

$$\frac{\partial \phi}{\partial \beta_0} = \frac{\partial \phi}{\partial \beta_1} = 0 \ \ and \ \ \frac{\partial^2 \phi}{\partial \beta_0^2} < 0, \frac{\partial^2 \phi}{\partial \beta_0^2} < 0$$

Minimization for Least Squares Criterion

- $\frac{\partial \phi}{\partial \beta_0} = 0$, results

$$\beta_0 = \overline{y} - \beta_1 \overline{x} \quad \ldots.(1)$$

$$\frac{\partial \phi}{\partial \beta_0} = \sum_{i=1}^{n} \frac{\partial}{\partial \beta_0} (y_i - (\beta_o + \beta_1 x_i))^2 = 0$$

$$\frac{\partial \phi}{\partial \beta_0} = 2 \sum_{i=1}^{n} (y_i - (\beta_o + \beta_1 x_i)) \frac{\partial}{\partial \beta_0} (y_i - (\beta_o + \beta_1 x_i)) \quad = 0$$

$$\frac{\partial \phi}{\partial \beta_0} = 2 \sum_{i=1}^{n} (y_i - (\beta_o + \beta_1 x_i)) \left(\frac{\partial}{\partial \beta_0} y_i - (\frac{\partial}{\partial \beta_0} \beta_o + \frac{\partial}{\partial \beta_0} \beta_1 x_i) \right) = 0$$

0 1 0

$$\frac{\partial \phi}{\partial \beta_0} = -2 \sum_{i=1}^{n} (y_i - (\beta_o + \beta_1 x_i)) = -2 \left(\sum_{i=1}^{n} y_i - \beta_0 \sum_{i=1}^{n} 1 - \beta_1 \sum_{i=1}^{n} x_i \right) = 0$$

n

$$\Rightarrow \quad \beta_0 = \overline{y} - \beta_1 \overline{x}$$

Minimization for Least Squares Criterion

- $\frac{\partial \phi}{\partial \beta_1} = 0$, results

$$\beta_1 = \frac{\sum_{i=1}^{n} x_i y_i - \beta_0 \sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} x_i^2} \ldots (2)$$

$$\frac{\partial \phi}{\partial \beta_1} = \sum_{i=1}^{n} \frac{\partial}{\partial \beta_1} (y_i - (\beta_o + \beta_1 x_i))^2 = 0$$

$$\frac{\partial \phi}{\partial \beta_1} = 2 \sum_{i=1}^{n} (y_i - (\beta_o + \beta_1 x_i)) \frac{\partial}{\partial \beta_1} (y_i - (\beta_o + \beta_1 x_i)) \quad = 0$$

$$\frac{\partial \phi}{\partial \beta_1} = 2 \sum_{i=1}^{n} (y_i - (\beta_o + \beta_1 x_i)) \left(\frac{\partial}{\partial \beta_1} y_i - (\frac{\partial}{\partial \beta_1} \beta_o + \frac{\partial}{\partial \beta_1} \beta_1 x_i) \right) = 0$$

$\frac{\partial}{\partial \beta_1} y_i \to 0$, $\frac{\partial}{\partial \beta_1} \beta_o \to 0$, $\frac{\partial}{\partial \beta_1} \beta_1 x_i \to x_i$

$$\frac{\partial \phi}{\partial \beta_1} = -2 \sum_{i=1}^{n} (y_i - (\beta_o + \beta_1 x_i) x_i) = -2 \left(\sum_{i=1}^{n} x_i y_i - \beta_0 \sum_{i=1}^{n} x_i - \beta_1 \sum_{i=1}^{n} x_i^2 \right) = 0$$

$\Rightarrow$

$$\beta_1 = \frac{\sum_{i=1}^{n} x_i y_i - \beta_0 \sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} x_i^2}$$

The Least Squares Equation

Substituting for β_1 from (2) in (1) and rearranging we get, the formula for β_0 and β_1:

$$\beta_1 = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sum (x - \bar{x})^2}, \beta_0 = \bar{y} - \beta_1 \bar{x}$$

Algebraic equivalent:

$$\beta_1 = \frac{\sum xy - \frac{\sum x \sum y}{n}}{\sum x^2 - \frac{(\sum x)^2}{n}}, \beta_0 = \bar{y} - \beta_1 \bar{x}$$

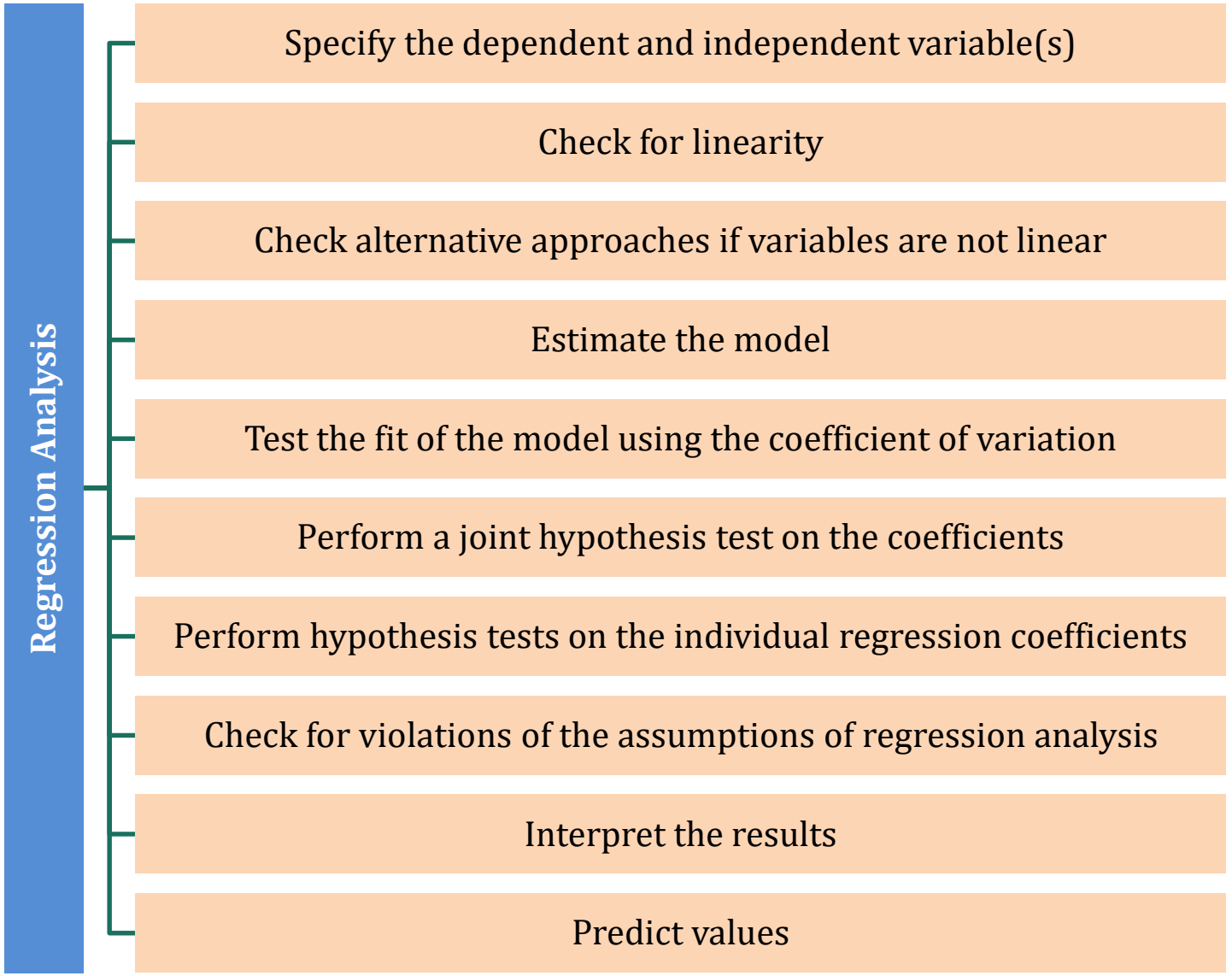
Interpretation

- β_0 is the estimated average value of y when the value of x is zero.
  - Traditionally it is the “bias” of the model
- β_1 is the estimated change in the average value of y as a result of a one-unit change in x.
  - A sensitivity measure, “Slope” or “rate” of the model

Initial Data Analysis

- Does the data look like as we expect?
- Prior to any analysis, the data should always be inspected for:
  - Data-entry errors
  - Missing values
  - Outliers
  - Unusual (e.g. asymmetric) distributions
  - Changes in variability
  - Clustering
  - Non-linear bivariate relatioships
  - Unexpected patterns

Steps Used to Implement a Regression Model



Simple Linear Regression - Example

A regression model based on a single independent variable is known as a simple regression model

Simple linear regression $Y = \beta_o + \beta_1 X$

Description

Data Science is a buzz word in the market. An engineering college (King's College") would like to introduce a Master's Program in Data Analytics. In order to plan the infrastructure for the course King's College would like to estimate the number of applications it can expect for the next year admission. King's College decided to examine the dataset.

- Year – Year
- APPLICATIONS – number of Applications when a new course was introduced in the college
- PLACE_RATE – Average Placement Rate (all the colleges combined)
- NO_GRAD_STUD – Number of under grad final year students (all courses combined) who would graduate from all the colleges in the current year

Simple Linear Regression - Example

RGui (64-bit) - [R Console]

File Edit View Misc Packages Windows Help

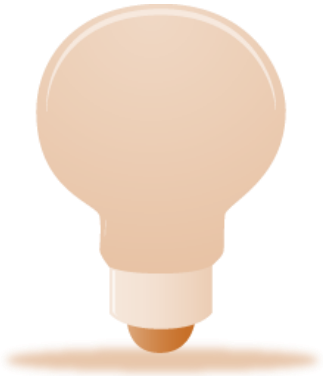
```
> data(longley)
> model <- lm( Employed ~ GNP, data=longley)
> summary(model)

Call:
lm(formula = Employed ~ GNP, data = longley)

Residuals:
     Min       1Q   Median       3Q      Max
-0.77958 -0.55440 -0.00944  0.34361  1.44594

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 51.843590   0.681372   76.09  < 2e-16 ***
GNP          0.034752   0.001706   20.37 8.36e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6566 on 14 degrees of freedom
Multiple R-squared:  0.9674,    Adjusted R-squared:  0.965
F-statistic: 415.1 on 1 and 14 DF,  p-value: 8.363e-12
```



Simple Regression Analysis

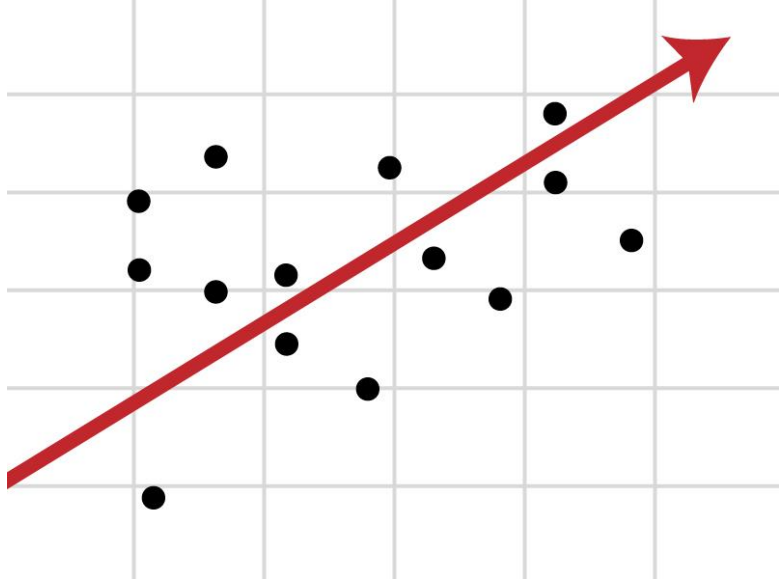
Output Interpretation

- From the output, it is determined that the intercept is 51.8436 and the coefficient for the placement rate is 0.0348

- Therefore, the complete regression equation is

Employed = 51.84 + 0.0348 * **GNP**

- Employed percentage goes up by 0.0348 when GNP goes up by 1 unit



Least Squares Regression Properties

The sum of the residuals from the least squares regression line is 0.

($\sum(y-\hat{y})=0$)

The sum of the squared residuals is a minimum .

(minimized $\sum(y-\hat{y})^2$)

The simple regression line always passes through the mean of the y variable and the mean of the x variable.

The least squares coefficients are unbiased estimates of β_0 and β_1.

Explained and Unexplained Variation

Total variation is made up of two parts:

$$SST = SSE + SSR$$

Total sum of Squares	Sum of Squares Error	Sum of Squares Regression
$SST = \sum (y - \bar{y})^2$	$SSE = \sum (y - \hat{y})^2$	$SSR = \sum (\hat{y} - \bar{y})^2$

Where:

$\bar{y}$ = Average value of the dependent variable

y = Observed values of the dependent variable

$\hat{y}$ = Estimated value of y for the given x value

Explained and Unexplained Variation (Contd.)

SST = Total sum of squares

Measures the variation of the y_i values around their mean y

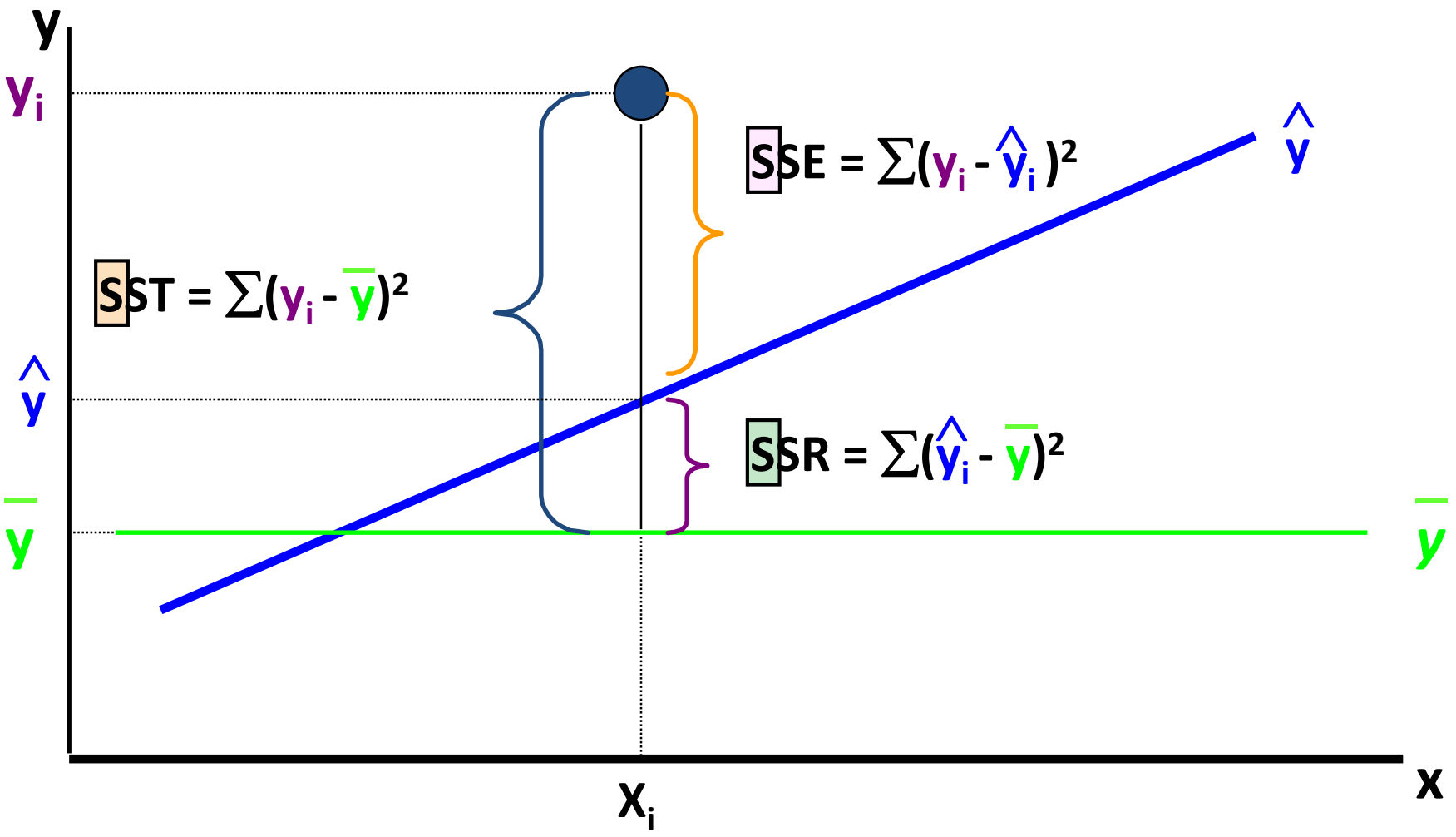
SSE = Error sum of squares

Variation attributable to factors other than the relationship between x and y

SSR = Regression sum of squares

Explained variation attributable to the relationship between x and y

Explained and Unexplained Variation



Coefficient of Determination, R^2

- The coefficient of determination is the portion of the total variation in the dependent variable that is explained by variation in the independent variable

- The coefficient of determination is also called R-squared and is denoted as R^2

$$R^2 = \frac{SSR}{SST}$$

Where $0 \leq R^2 \leq 1$

Coefficient of Determination, R^2

Coefficient of determination

$$R^2 = \frac{SSR}{SST} = \frac{\text{sum of squares explained by regression}}{\text{total sum of squares}}$$

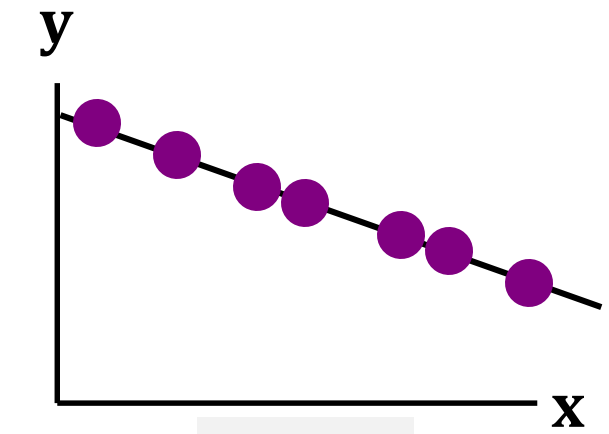
Note: In the single independent variable case, the coefficient of determination is:

$$R^2 = r^2$$

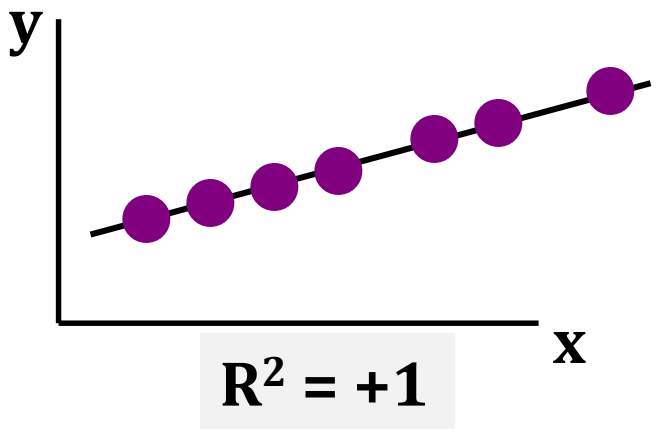
Where:

- R^2 = Coefficient of determination
- r = Simple correlation coefficient

Example of Approximate R^2 Values



$R^2 = 1$

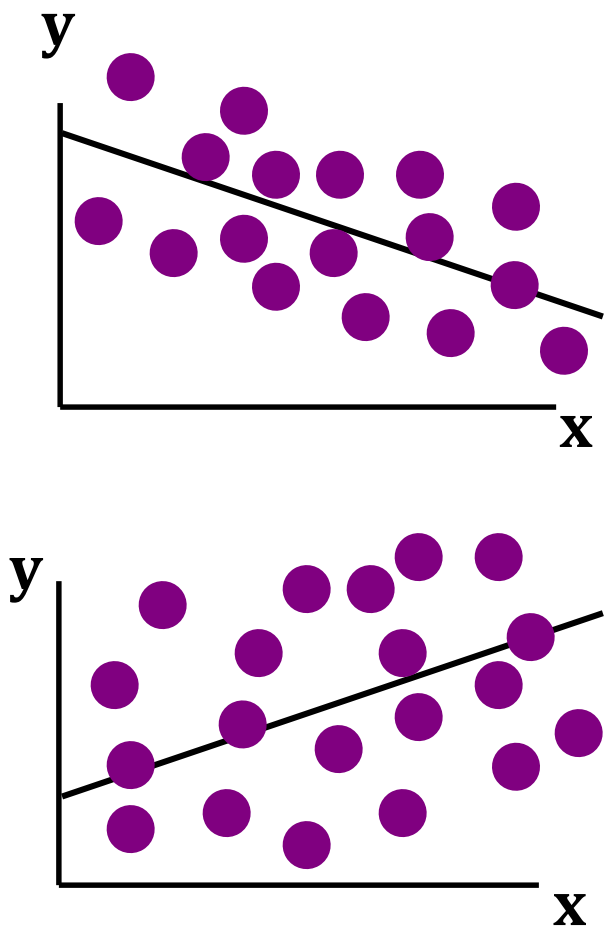


$R^2 = +1$

$R^2 = 1$

- Perfect linear relationship between x and y:
- 100% of the variation in y is explained by variation in x

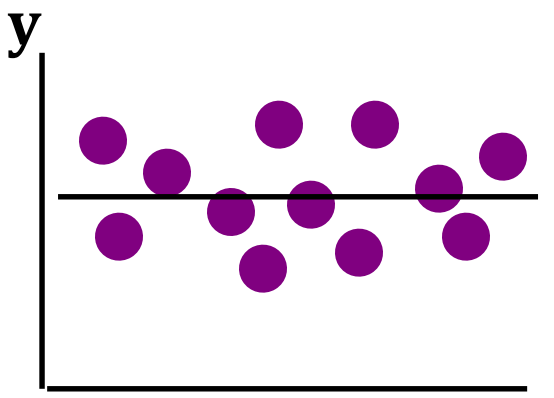
Example of Approximate R^2 Values



$0 < R^2 < 1$

- Weaker linear relationship between x and y:
- Some but not all of the variation in y is explained by variation in x

Example of Approximate R^2 Values



$R^2 = 0$

$R^2 = 0$

- No linear relationship between x and y:
- The value of Y does not depend on x. (None of the variation in y is explained by variation in x)

Standard Error of Estimate

The standard deviation of the variation of observations around the regression line is estimated by:

$$s_{\varepsilon} = \sqrt{\frac{SSE}{n - k - 1}}$$

Where,

- SSE = Sum of squares error
- n = Sample size
- k = number of independent variables in the model

The Standard Deviation of the Regression Slope

The standard error of the regression slope coefficient (b_1) is estimated by:

$$s_{b_1} = \frac{s_\varepsilon}{\sqrt{\sum (x - \bar{x})^2}} = \frac{s_\varepsilon}{\sqrt{\sum x^2 - \frac{(\sum x)^2}{n}}}$$

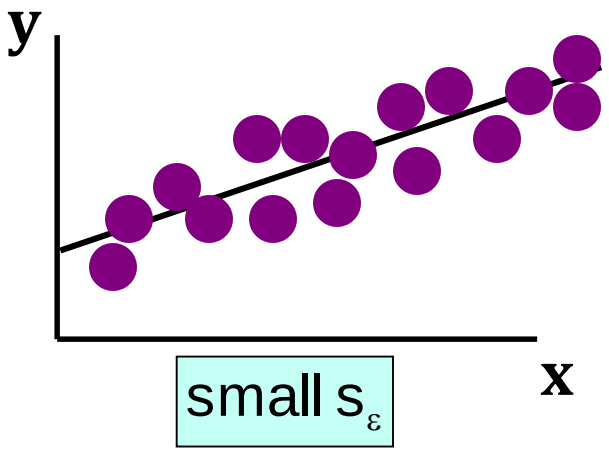
Where:

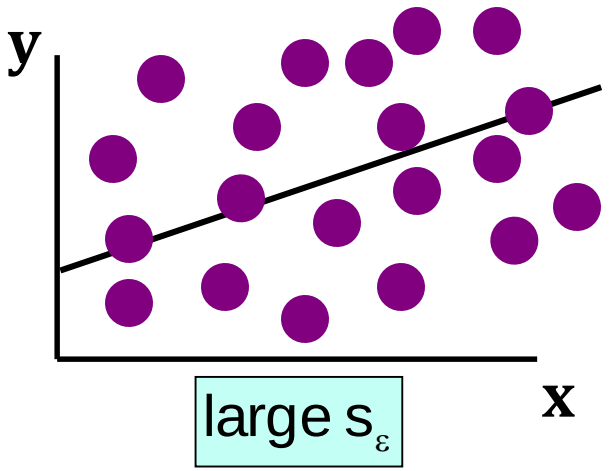
s_{b_1} = Estimate of the standard error of the least squares slope

$s_\varepsilon = \sqrt{\frac{SSE}{n-2}}$ = Sample standard error of the estimate

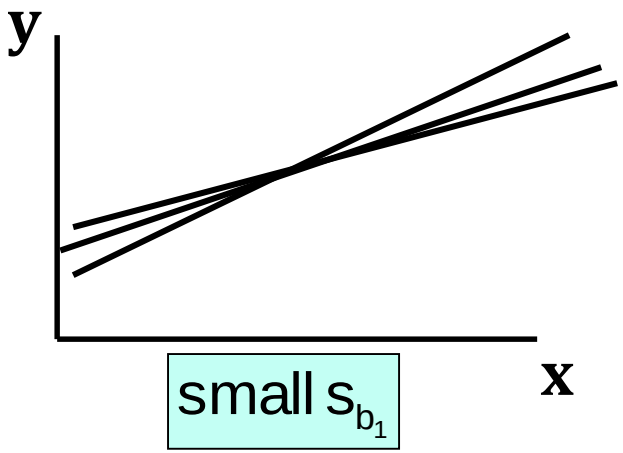
Comparing Standard Errors

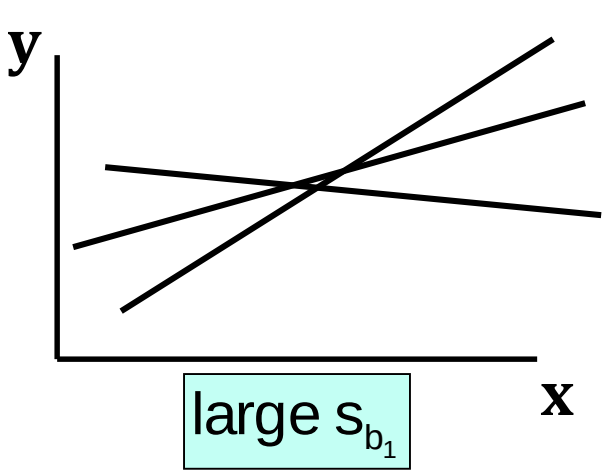
Variation of observed y values from the regression line





Variation in the slope of regression lines from different possible samples





Inference about the Slope: t Test

- T test for a population slope
  - Is there a linear relationship between x and y?

- Null and alternative hypotheses
  - H_0: $\beta_1 = 0$ (no linear relationship)
  - H_1: β_1 not equal to 0 (linear relationship does exist)
- Test statistic

$$t = \frac{b_1 - \beta_1}{s_{b_1}}$$

Where,

- b_1 = Sample regression slope coefficient
- β_1 = Hypothesized slope
- s_{b1} = Estimator of the standard error of the slope

$$d.f. = n - 2$$

Inference about the Slope: t Test

House Price in $1000s (y)	Square Feet (x)
245	1400
312	1600
279	1700
308	1875
199	1100
219	1550
405	2350
324	2450
319	1425
255	1700

Estimated Regression Equation:

$$\hat{y}\text{ House price} = 98.25 + 0.1098\,(\text{sq.ft.})$$

The slope of this model is 0.1098

Does square footage of the house affect its sales price?

Inference about the Slope: t Test Example

Test Statistic: **t = 3.329**

$H_0: \beta_1 = 0$

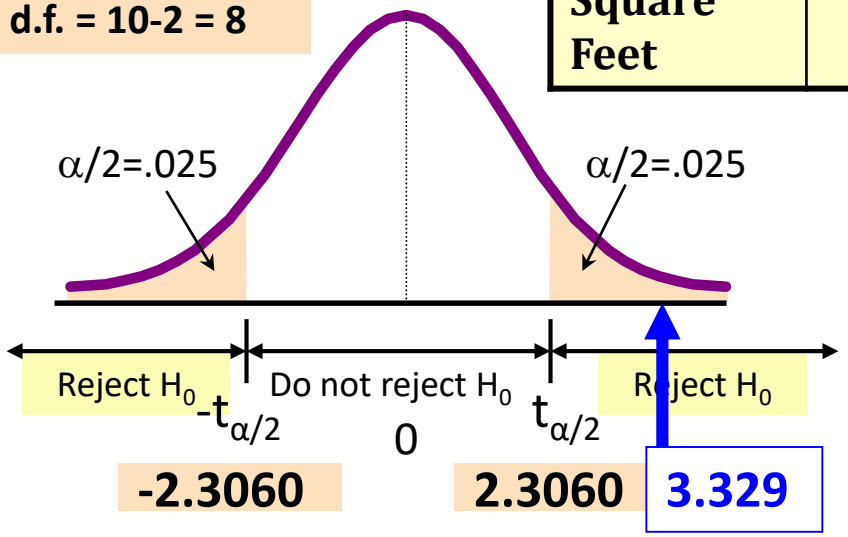
$H_A: \beta_1 \neq 0$

From output:

	Coefficients	*Standard Error*	*t Stat*	*P-value*
Intercept	**98.24833**	**58.03348**	**1.69296**	**0.12892**
Square Feet	**0.10977**	**0.03297**	**3.32938**	**0.01039**

b_1 S_{b_1} t

d.f. = 10-2 = 8



Decision:

Reject H_0

Conclusion:

There is sufficient evidence that square footage affects house price

Regression Analysis for Description

Confidence Interval Estimate of the Slope:

$$b_1 \pm t_{\alpha/2} s_{b_1}$$

d.f. = n - 2

Excel Printout for House Prices:

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	**98.24833**	**58.03348**	**1.69296**	**0.12892**	**-35.57720**	**232.07386**
Square Feet	**0.10977**	**0.03297**	**3.32938**	**0.01039**	**0.03374**	**0.18580**

At 95% level of confidence, the confidence interval for the slope is (0.0337, 0.1858)

Regression Analysis for Description

	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%
Intercept	98.24833	58.03348	1.69296	0.12892	-35.57720	232.07386
Square Feet	0.10977	0.03297	3.32938	0.01039	0.03374	0.18580

Since the units of the house price variable is $1000s, we are 95% confident that the average impact on sales price is between $33.70 and $185.80 per square foot of house size.

- This 95% confidence interval does not include 0.
- Conclusion: There is a significant relationship between house price and square feet at the .05 level of significance

Confidence Interval for the Average y, given x

Confidence interval estimate for the
mean of y given a particular x_p

Size of interval varies according to distance away from mean, x

$$\hat{y} \pm t_{\alpha/2} s_\varepsilon \sqrt{\frac{1}{n} + \frac{(x_p - \bar{x})^2}{\sum (x - \bar{x})^2}}$$

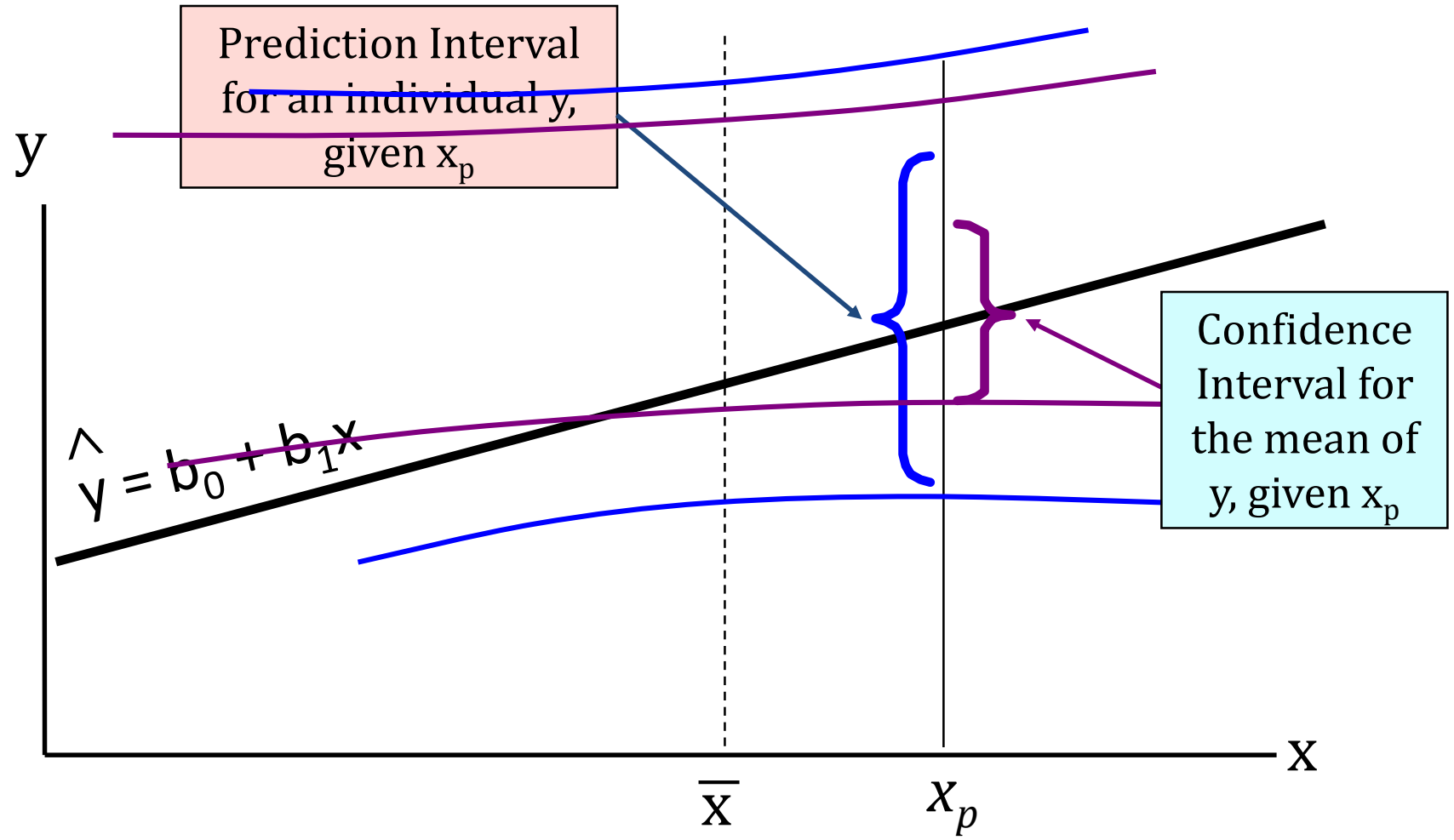
Confidence Interval for an Individual y, given x

Confidence interval estimate for an
Individual value of y given a particular x_p

$$\hat{y} \pm t_{\alpha/2} s_\varepsilon \sqrt{1 + \frac{1}{n} + \frac{(x_p - \bar{x})^2}{\sum (x - \bar{x})^2}}$$

This extra term adds to the interval width to reflect the added uncertainty for an individual case.

Interval Estimates for Different Values of x



Example

House Price in $1000s (y)	Square Feet (x)
245	1400
312	1600
279	1700
308	1875
199	1100
219	1550
405	2350
324	2450
319	1425
255	1700

Estimated Regression Equation:

$$\widehat{\text{House price}} = 98.25 + 0.1098\,(\text{sq.ft.})$$

Predict the price for a house with 2000 square feet

Example: House Prices

Predict the price for a house with 2000 square feet:

$$\begin{aligned}\widehat{\text{House price}} &= 98.25 + 0.1098\,(\text{sq.ft.}) \\ &= 98.25 + 0.1098(2000) \\ &= 317.85\end{aligned}$$

The predicted price for a house with 2000 square feet is 317.85($1,000s) = $317,850

Estimation of Mean Values: Example

Confidence Interval Estimate for $E(y)|x_p$

Find the 95% confidence interval for the average price of 2,000 square-foot houses.

Predicted Price $\hat{Y}_i = 317.85$ ($1,000s)

$$\hat{y} \pm t_{\alpha/2} s_\varepsilon \sqrt{\frac{1}{n} + \frac{(x_p - \bar{x})^2}{\sum (x - \bar{x})^2}} = 317.85 \pm 37.12$$

The confidence interval endpoints are 280.66 -- 354.90, or from $280,660 -- $354,900

Estimation of Individual Values: Example

Prediction Interval Estimate for $y|x_p$

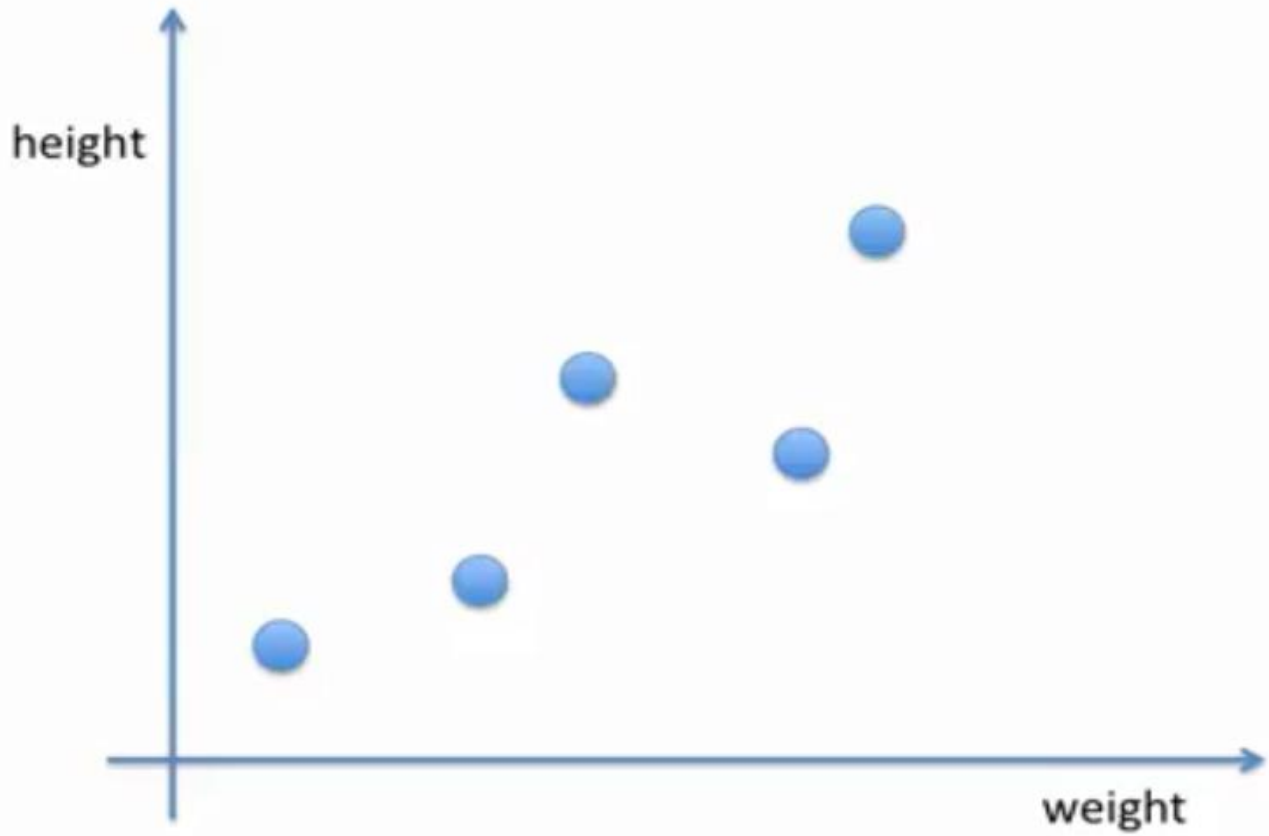
Find the 95% confidence interval for an individual house with 2,000 square feet.

Predicted Price $Y_i = 317.85$ ($1,000s)

$$\hat{y} \pm t_{\alpha/2} s_\varepsilon \sqrt{1 + \frac{1}{n} + \frac{(x_p - \bar{x})^2}{\sum (x - \bar{x})^2}} = 317.85 \pm 102.28$$

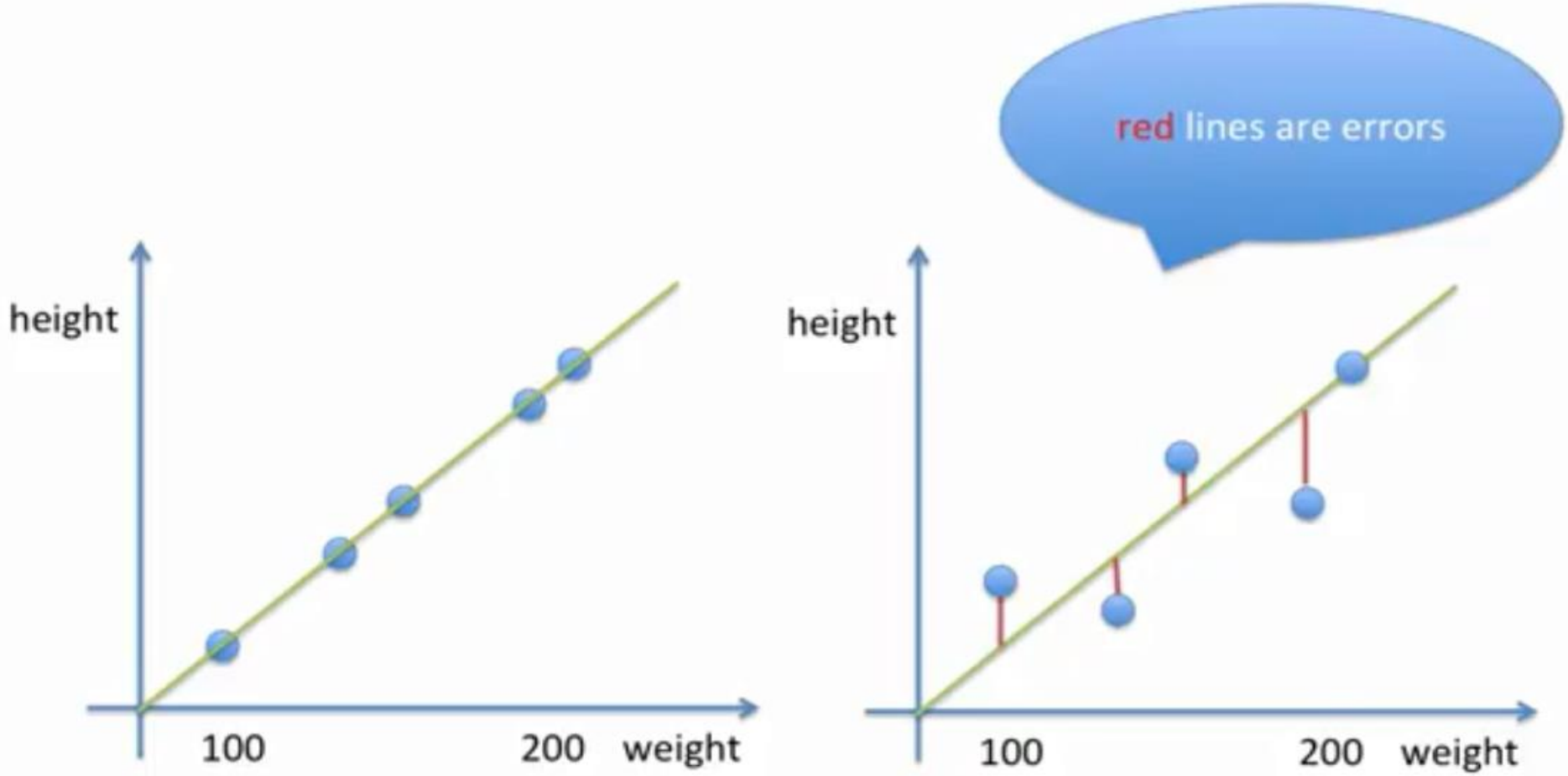
The prediction interval endpoints are 215.50 -- 420.07, or from $215,500 -- $420,070

Linear Regression

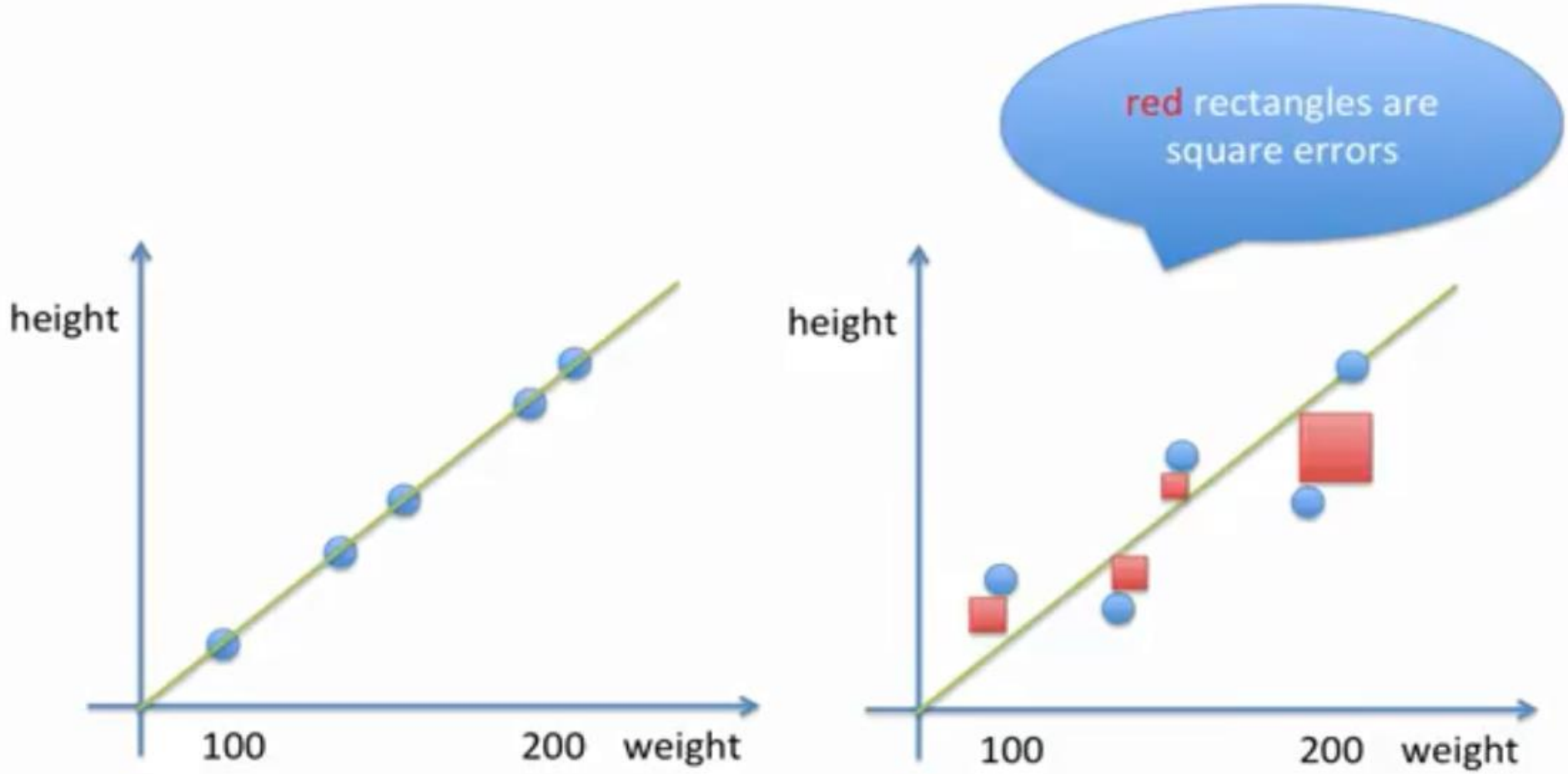


Linear regression predicts output by fitting a linear equation (y = ax + b) to observed data

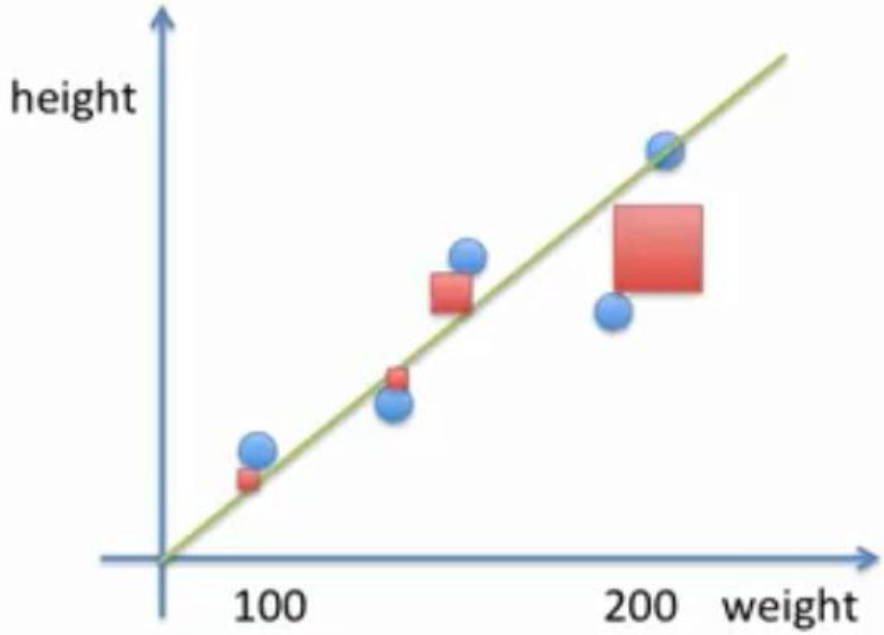
Error – difference between prediction and real value

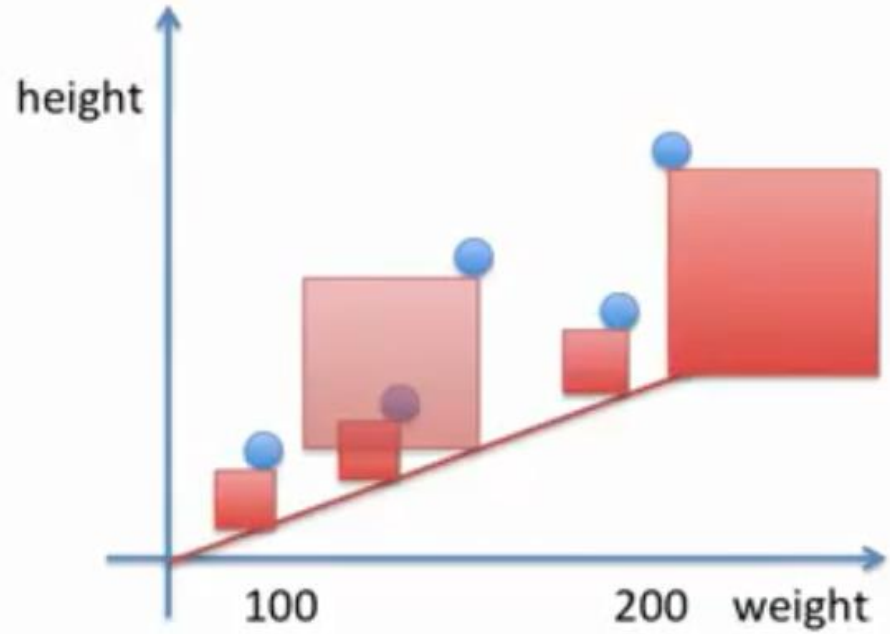


Square Error– (difference between prediction and real value)^2



Mean Square Error





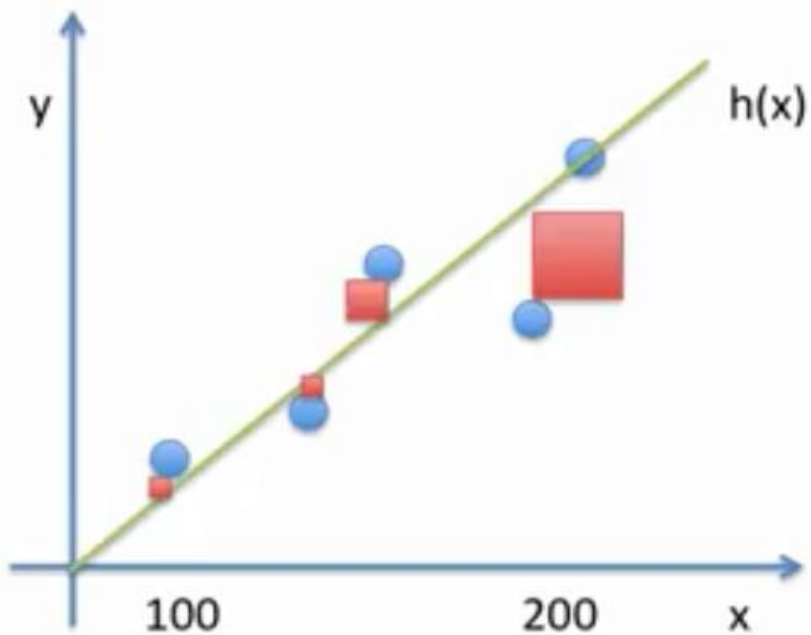
green line has less error, therefore prediction will be more likely to be true value (height)

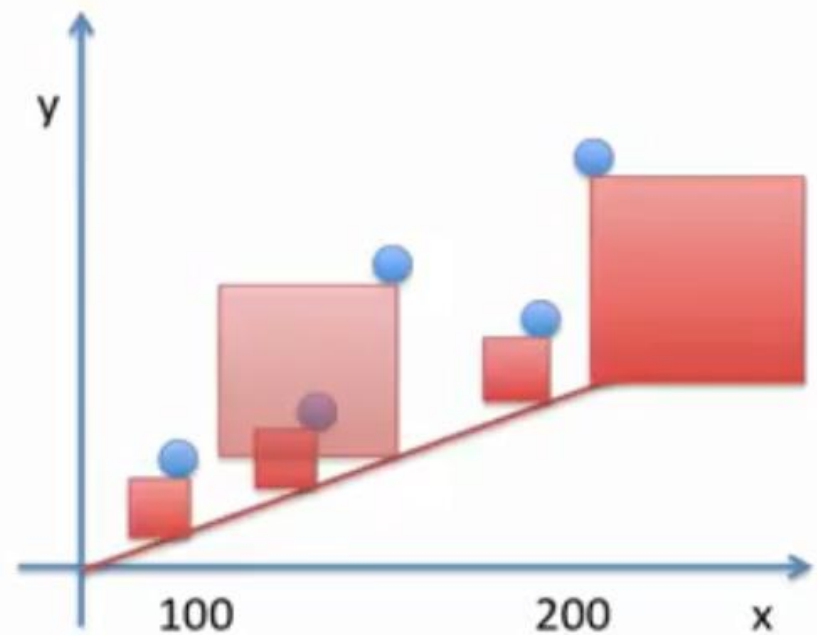
Find linear equation has Least Mean Square (LMS) Error

Error = h(x) - y

Square Error = $(h(x) - y)^2$

Mean Square Error = $1/n \sum(h(x) - y)^2$



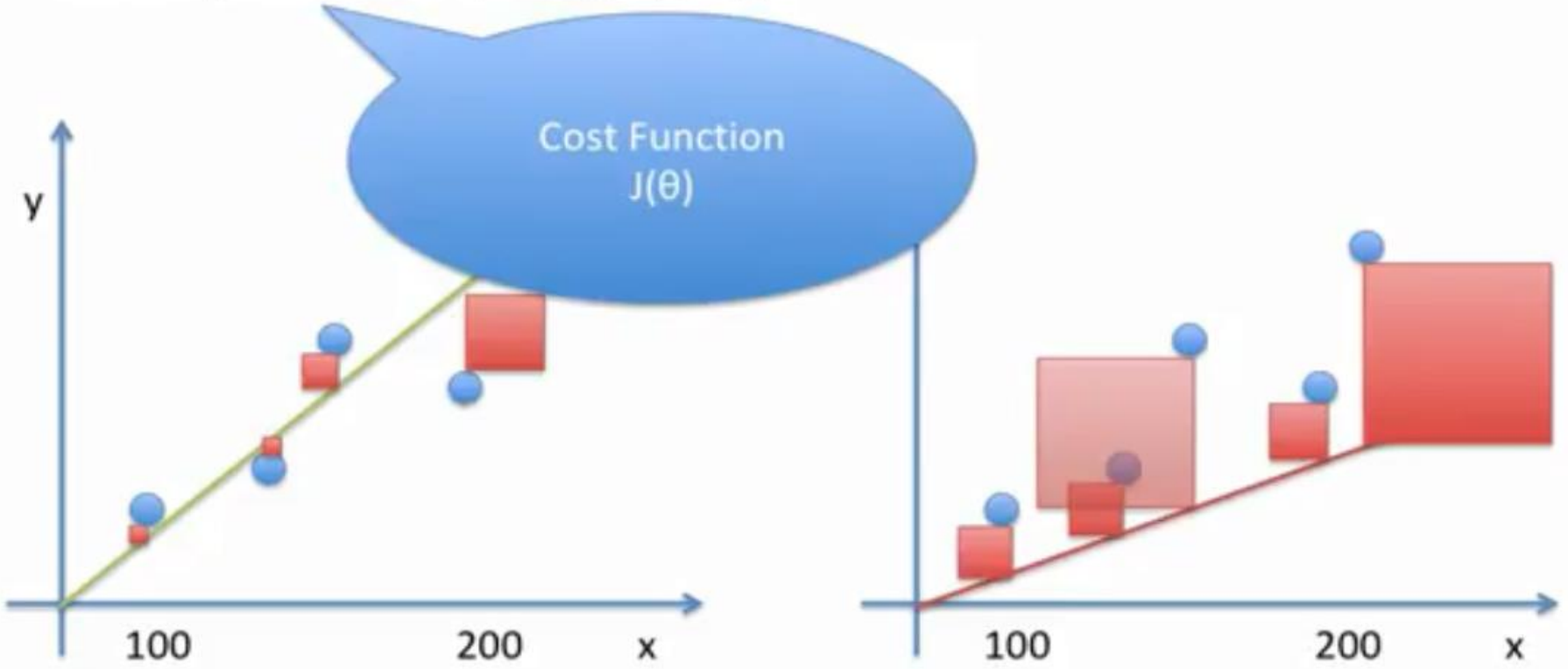


Find linear equation has Least Mean Square (LMS) Error

Error = h(x) - y

Square Error = $(h(x) - y)^2$

Mean Square Error = $1/n * \sum (h(x) - y)^2$



Gradient Descent

- To minimize the cost function, we use optimization algorithm. The most common optimization algorithm is Gradient Descent Algorithm.

Types:

- **Stochastic Gradient Descent** – Performs the parameters update on each example.

- **Mini batch Gradient Descent** - Sums up over lower number of examples based on the batch size.

- **Batch Gradient Descent** - Sum up each iteration when performing the updates to the parameters.

Stochastic Gradient Descent

- Gradient Descent is an algorithm which iterates through different combinations of weights in an optimal way to find the best combination such that errors are minimized.
- Sum of Squared Errors (SSE) = ½ Sum (Actual House Price – Predicted House Price)2

$$= \frac{1}{2} \text{Sum}(Y - Y_{pred})^2$$

HOUSING DATA	
House Size (X)	House Price (Y)
1,100	1,99,000
1,400	2,45,000
1,425	3,19,000
1,550	2,40,000
1,600	3,12,000
1,700	2,79,000
1,700	3,10,000
1,875	3,08,000
2,350	4,05,000
2,450	3,24,000

Min-Max Standardization	
X (X-Min/Max-min)	Y (Y-Min/Max-Min)
0.00	0.00
0.22	0.22
0.24	0.58
0.33	0.20
0.37	0.55
0.44	0.39
0.44	0.54
0.57	0.53
0.93	1.00
1.00	0.61

Stochastic Gradient Descent

Step 1: To fit a line Ypred = a + b X, start off with random values of a and b and calculate prediction error (SSE)

a	b	X	Y	YP=a+bX	SSE=1/2(Y-YP)^2
0.45	0.75	0.00	0.00	0.45	0.101
		0.22	0.22	0.62	0.077
		0.24	0.58	0.63	0.001
		0.33	0.20	0.70	0.125
		0.37	0.55	0.73	0.016
		0.44	0.39	0.78	0.078
		0.44	0.54	0.78	0.030
		0.57	0.53	0.88	0.062
		0.93	1.00	1.14	0.010
		1.00	0.61	1.20	0.176
				Total SSE	**0.677**

Stochastic Gradient Descent

Step 2: Calculate the error gradient w.r.t the weights

$\partial SSE/\partial a = -(Y-YP)$

$\partial SSE/\partial b = -(Y-YP)X$

a	b	X	Y	YP=a+bX		SSE		∂SSE/∂a = -(Y-YP)	∂SSE/∂b = -(Y-YP)X
0.45	0.75	0.00	0.00	0.45		0.101		0.45	0.00
		0.22	0.22	0.62		0.077		0.39	0.09
		0.24	0.58	0.63		0.001		0.05	0.01
		0.33	0.20	0.70		0.125		0.50	0.17
		0.37	0.55	0.73		0.016		0.18	0.07
		0.44	0.39	0.78		0.078		0.39	0.18
		0.44	0.54	0.78		0.030		0.24	0.11
		0.57	0.53	0.88		0.062		0.35	0.20
		0.93	1.00	1.14		0.010		0.14	0.13
		1.00	0.61	1.20		0.176		0.59	0.59
					Total SSE	**0.677**	**Sum**	**3.300**	**1.545**

Stochastic Gradient Descent

Step 3: Adjust the weights with the gradients to reach the optimal values where SSE is minimized

We need to update the random values of a, b so that we move in the direction of optimal a, b.

Update rules:

New a = a – r * ∂SSE/∂a = 0.45-0.01*3.300 = 0.42

New b = b – r * ∂SSE/∂b= 0.75-0.01*1.545 = 0.73

r is the learning rate = 0.01

Stochastic Gradient Descent

Step 4: Use new a and b for prediction and to calculate new Total SSE

a	b	X	Y	YP=a+bX		SSE		∂SSE/∂a	∂SSE/∂b
0.42	0.73	0.00	0.00	0.42		0.087		0.42	0.00
		0.22	0.22	0.58		0.064		0.36	0.08
		0.24	0.58	0.59		0.000		0.01	0.00
		0.33	0.20	0.66		0.107		0.46	0.15
		0.37	0.55	0.69		0.010		0.14	0.05
		0.44	0.39	0.74		0.063		0.36	0.16
		0.44	0.54	0.74		0.021		0.20	0.09
		0.57	0.53	0.84		0.048		0.31	0.18
		0.93	1.00	1.10		0.005		0.10	0.09
		1.00	0.61	1.15		0.148		0.54	0.54
					Total SSE	0.553	Sum	2.900	1.350

Step 5: Repeat step 3 and 4 till the time further adjustments to a, b doesn't significantly reduces the error.

Stochastic Gradient Descent

- **Step 1:** Initialize the weights(a & b) with random values and calculate Error (SSE)
- **Step 2:** Calculate the gradient i.e. change in SSE when the weights (a & b) are changed by a very small value from their original randomly initialized value. This helps us move the values of a & b in the direction in which SSE is minimized.
- **Step 3:** Adjust the weights with the gradients to reach the optimal values where SSE is minimized
- **Step 4:** Use the new weights for prediction and to calculate the new SSE
- **Step 5:** Repeat steps 2 and 3 till further adjustments to weights doesn’t significantly reduce the Error

Basic Gradient Descent

- Sum of Squared Errors (SSE) = ½ Sum (Actual House Price – Predicted House Price)2

  = ½ Sum$(Y - Y_{pred})^2$

- Cost Function: $J(\beta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\beta(x^{(i)}) - y^{(i)})^2$

- Gradient Descent Function: $\beta_j := \beta_j - \alpha\frac{\partial}{\partial\beta_j}J(\beta)$

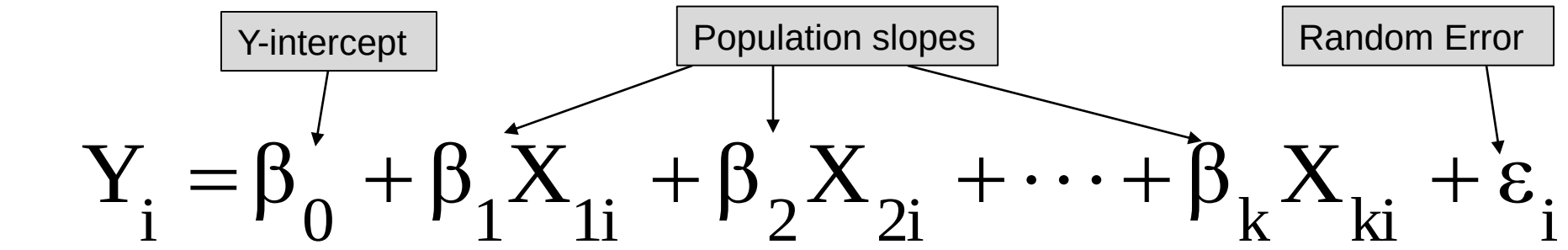
- Substituting $J(\beta)$: $\beta_j := \beta_j - \alpha\frac{1}{m}\sum_{i=1}^{m}(h_\beta(x^{(i)}) - y^{(i)})x_j^{(i)}$

Multiple Regression

Multiple Regression Model

Idea: Examine the linear relationship between
1 dependent (Y) & 2 or more independent variables (X_i)

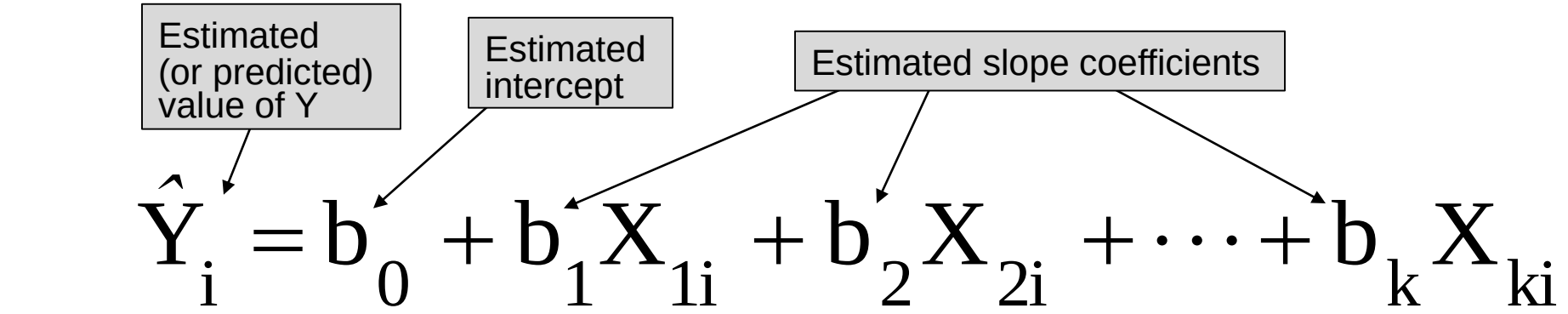
Multiple Regression Model with k Independent Variables:



Multiple Regression Model

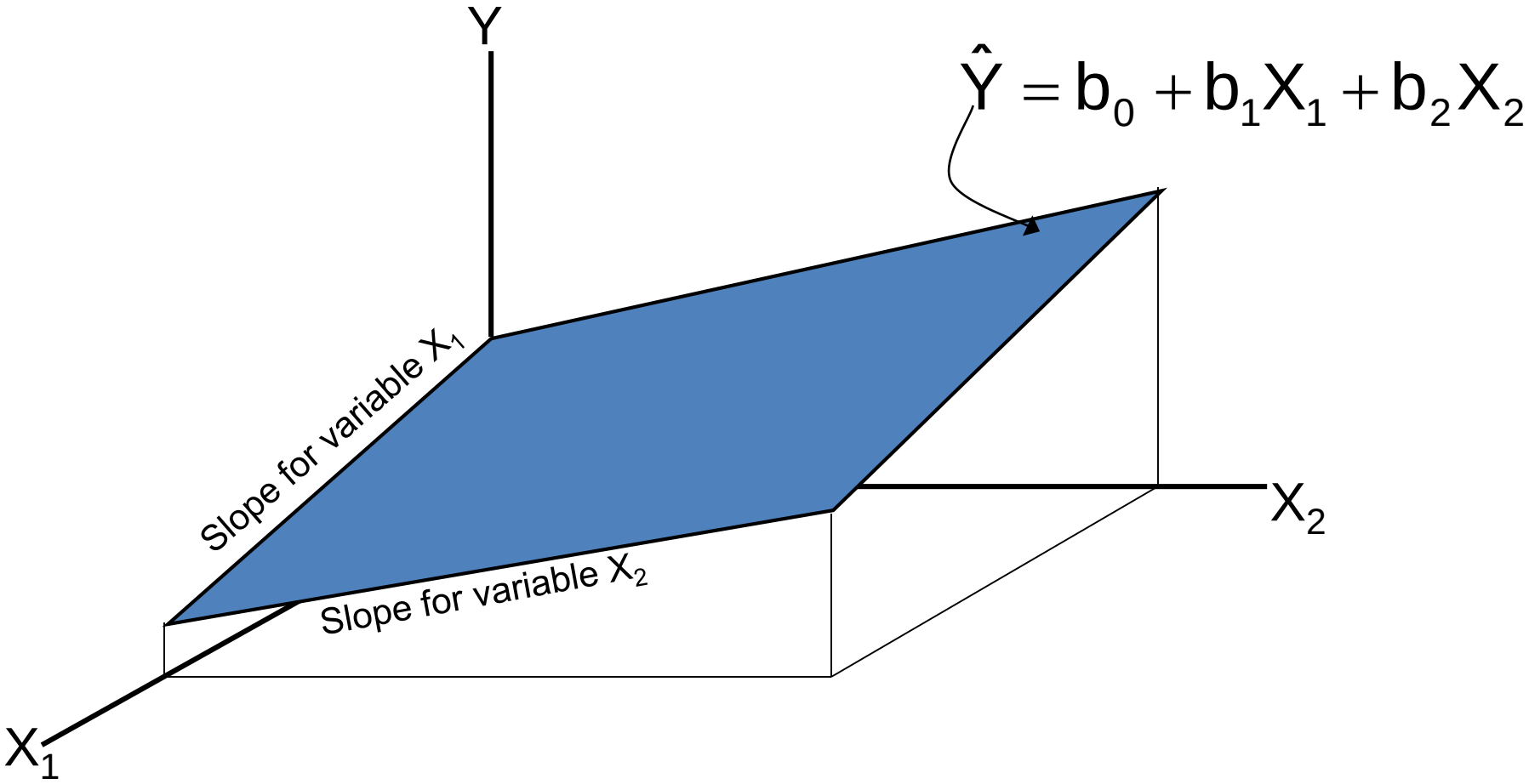
The coefficients of the multiple regression model are estimated using sample data

Multiple regression equation with k independent variables:



Multiple Regression Model

Two variable model



Multiple Regression Model

- A distributor of frozen dessert pies wants to evaluate factors that influence demand
  - Dependent variable : Pie sales (units per week)
  - Independent variables: Price (in $)
    Advertising ($100's)
- Data are collected for 15 weeks

Pie Sales Example

Week	Pie Sales	Price ($)	Advertising ($100s)
1	350	5.50	3.3
2	460	7.50	3.3
3	350	8.00	3.0
4	430	8.00	4.5
5	350	6.80	3.0
6	380	7.50	4.0
7	430	4.50	3.0
8	470	6.40	3.7
9	450	7.00	3.5
10	490	5.00	4.0
11	340	7.20	3.5
12	300	7.90	3.2
13	440	5.90	4.0
14	450	5.00	3.5
15	300	7.00	2.7

Multiple regression equation:

$$\widehat{Sales} = b_0 + b_1 (Price) + b_2 (Advertising)$$

The Multiple Regression Equation

$$\widehat{Sales} = 306.526 - 24.975(Price) + 74.131(Advertising)$$

where
- Sales is in number of pies per week
- Price is in $
- Advertising is in $100's.

b_1 = -24.975: sales will decrease, on average, by 24.975 pies per week for each $1 increase in selling price, net of the effects of changes due to advertising

b_2 = 74.131: sales will increase, on average, by 74.131 pies per week for each $100 increase in advertising, net of the effects of changes due to price

Using The Equation to Make Predictions

Predict sales for a week in which the selling price is $5.50 and advertising is $350:

$$\widehat{\text{Sales}} = 306.526 - 24.975(\text{Price}) + 74.131(\text{Advertising})$$

$$= 306.526 - 24.975(5.50) + 74.131(3.5)$$

$$= 428.62$$

Predicted sales is 428.62 pies

Note that Advertising is in $100's, so $350 means that $X_2 = 3.5$

Coefficient of Multiple Determination

- Reports the proportion of total variation in Y explained by all X variables taken together

$$r^2 = \frac{SSR}{SST} = \frac{\text{regression sum of squares}}{\text{total sum of squares}}$$

Adjusted r^2

- r^2 never decreases when a new X variable is added to the model
  - This can be a disadvantage when comparing models
- What is the net effect of adding a new variable?
  - We lose a degree of freedom when a new X variable is added
  - Did the new X variable add enough explanatory power to offset the loss of one degree of freedom?

Adjusted r^2

- Shows the proportion of variation in Y explained by all X variables adjusted for the number of X variables used

$$r^2_{adj} = 1 - \left[(1 - r^2) \left(\frac{n-1}{n-k-1} \right) \right]$$

(where n = sample size, k = number of independent variables)

  - Penalize excessive use of unimportant independent variables
  - Smaller than r^2
  - Useful in comparing among models

Is the Model Significant?

- F Test for Overall Significance of the Model
- Shows if there is a linear relationship between all of the X variables considered together and Y
- Use F-test statistic
- Hypotheses:

$H_0: \beta_1 = \beta_2 = \cdots = \beta_k = 0$ (no linear relationship)

H_1: at least one $\beta_i \neq 0$ (at least one independent variable affects Y)

F Test for Overall Significance

- Test statistic:

$$F_{STAT} = \frac{MSR}{MSE} = \frac{\frac{SSR}{k}}{\frac{SSE}{n-k-1}}$$

where F_{STAT} has numerator d.f. = k and

denominator d.f. = (n – k - 1)

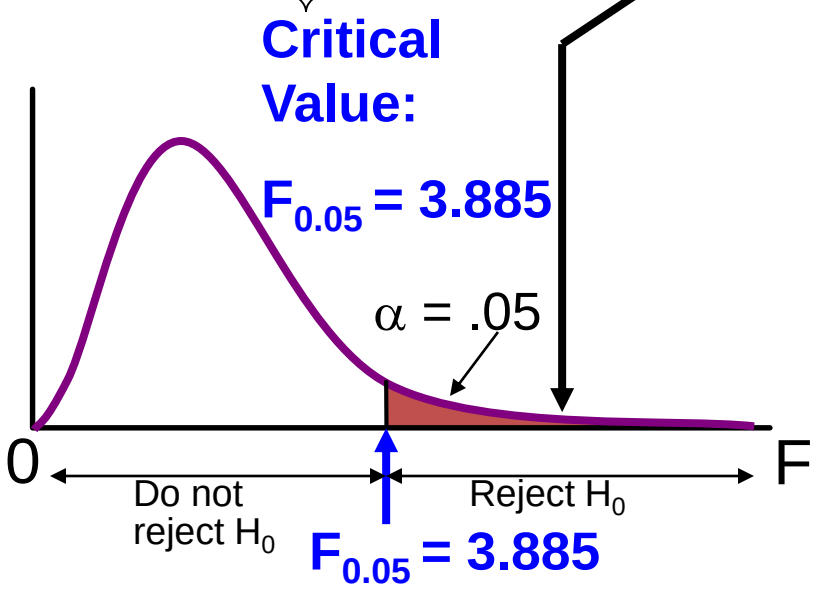
F Test for Overall Significance

$H_0: \beta_1 = \beta_2 = 0$

$H_1: \beta_1$ and β_2 not both zero

$\alpha = .05$

$df_1 = 2 \quad df_2 = 12$



Test Statistic:

$$F_{STAT} = \frac{MSR}{MSE} = 6.5386$$

Decision:

Since F_{STAT} test statistic is in the rejection region (p-value < .05), reject H_0

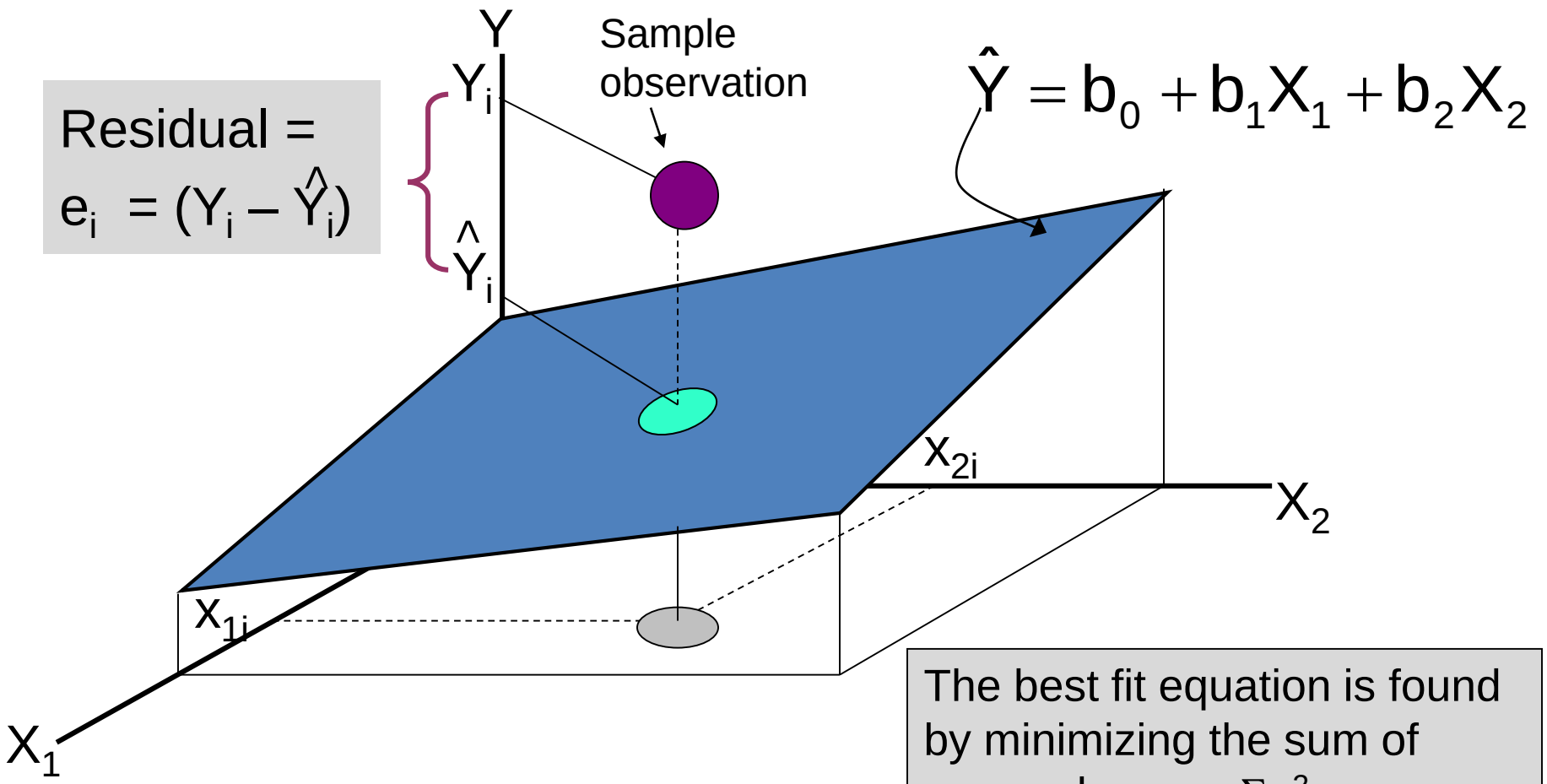
Conclusion:

There is evidence that at least one independent variable affects Y

Residuals in Multiple Regression

Two variable model

Residual = $e_i = (Y_i - \hat{Y}_i)$



The best fit equation is found by minimizing the sum of squared errors, Σe^2

Multiple Regression Assumptions

Errors (residuals) from the regression model:

$$e_i = (Y_i - \hat{Y}_i)$$

Assumptions:

- The errors are normally distributed
- Errors have a constant variance
- The model errors are independent

Are Individual Variables Significant?

- Use t tests of individual variable slopes
- Shows if there is a linear relationship between the variable X_j and Y holding constant the effects of other X variables
- Hypotheses:
  - $H_0: \beta_j = 0$ (no linear relationship)
  - $H_1: \beta_j \neq 0$ (linear relationship does exist between X_j and Y)

Are Individual Variables Significant?

H_0: $\beta_j = 0$ (no linear relationship)

H_1: $\beta_j \neq 0$ (linear relationship does exist between X_j and Y)

Test Statistic:

$$t_{STAT} = \frac{b_j - 0}{S_{b_j}}$$

(df = n – k – 1)

Confidence Interval Estimate for the Slope

Confidence interval for the population slope β_j

$$b_j \pm t_{\alpha/2} S_{b_j}$$

where t has
(n – k – 1) d.f.

	Coefficients	*Standard Error*
Intercept	**306.52619**	**114.25389**
Price	**-24.97509**	**10.83213**
Advertising	**74.13096**	**25.96732**

Here, t has
(15 – 2 – 1) = 12 d.f.

Example: Form a 95% confidence interval for the effect of changes in price (X_1) on pie sales:

-24.975 ± (2.1788)(10.832)

So the interval is (-48.576 , -1.374)

(This interval does not contain zero, so price has a significant effect on sales)

Testing Portions of the Multiple Regression Model

- Contribution of a Single Independent Variable X_j

$SSR(X_j \mid \text{all variables except } X_j)$
$= SSR(\text{all variables}) - SSR(\text{all variables except } X_j)$

  - Measures the contribution of X_j in explaining the total variation in Y (SST)

Testing Portions of the Multiple Regression Model

Contribution of a Single Independent Variable X_j, assuming all other variables are already included (consider here a 2-variable model):

$SSR(X_1 \mid X_2)$
= SSR (all variables) – $SSR(X_2$

From ANOVA section of regression for

$$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2$$

From ANOVA section of regression for

$$\hat{Y} = b_0 + b_2 X_2$$

Measures the contribution of X_1 in explaining SST

The Partial F-Test Statistic

- Consider the hypothesis test:

H_0: variable X_j does not significantly improve the model after all other variables are included

H_1: variable X_j significantly improves the model after all other variables are included

- Test using the F-test statistic:

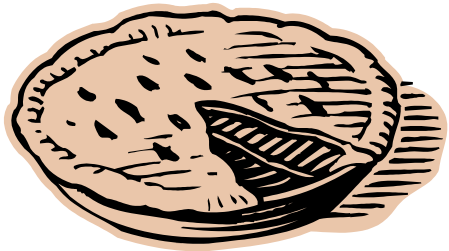
(with 1 and n-k-1 d.f.)

$$F_{STAT} = \frac{\text{SSR}\left(X_j \mid \text{all variables except j}\right)}{\text{MSE}}$$

Testing Portions of Model: Example

Example: Frozen dessert pies

Test at the $\alpha = .05$ level to determine whether the price variable significantly improves the model given that advertising is included



Testing Portions of Model: Example

H_0: X_1 (price) does not improve the model with X_2 (advertising) included

H_1: X_1 does improve model

$\alpha = .05$, df = 1 and 12

$F_{0.05} = 4.75$

(For X_1 and X_2)

ANOVA			
	df	*SS*	*MS*
Regression	2	**29460.02687**	14730.01343
Residual	12	27033.30647	**2252.775539**
Total	14	56493.33333	

(For X_2 only)

ANOVA		
	df	*SS*
Regression	1	**17484.22249**
Residual	13	39009.11085
Total	14	56493.33333

Testing Portions of Model: Example

(For X_1 and X_2)

ANOVA			
	df	*SS*	*MS*
Regression	2	**29460.02687**	14730.01343
Residual	12	27033.30647	**2252.775539**
Total	14	56493.33333	

(For X_2 only)

ANOVA		
	df	*SS*
Regression	1	**17484.22249**
Residual	13	39009.11085
Total	14	56493.33333

$$F_{STAT} = \frac{\text{SSR}\,(X_1 \mid X_2)}{\text{MSE(all)}} = \frac{29{,}460.03 - 17{,}484.22}{2252.78} = 5.316$$

Conclusion: Since $F_{STAT} = 5.316 > F_{0.05} = 4.75$ Reject H_0;
Adding X_1 does improve model

Dummy-Variable Example (with 2 Levels)

$$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2$$

Let:

Y = pie sales

X_1 = price

X_2 = holiday (X_2 = 1 if a holiday occurred during the week)
(X_2 = 0 if there was no holiday that week)

Using Dummy Variables

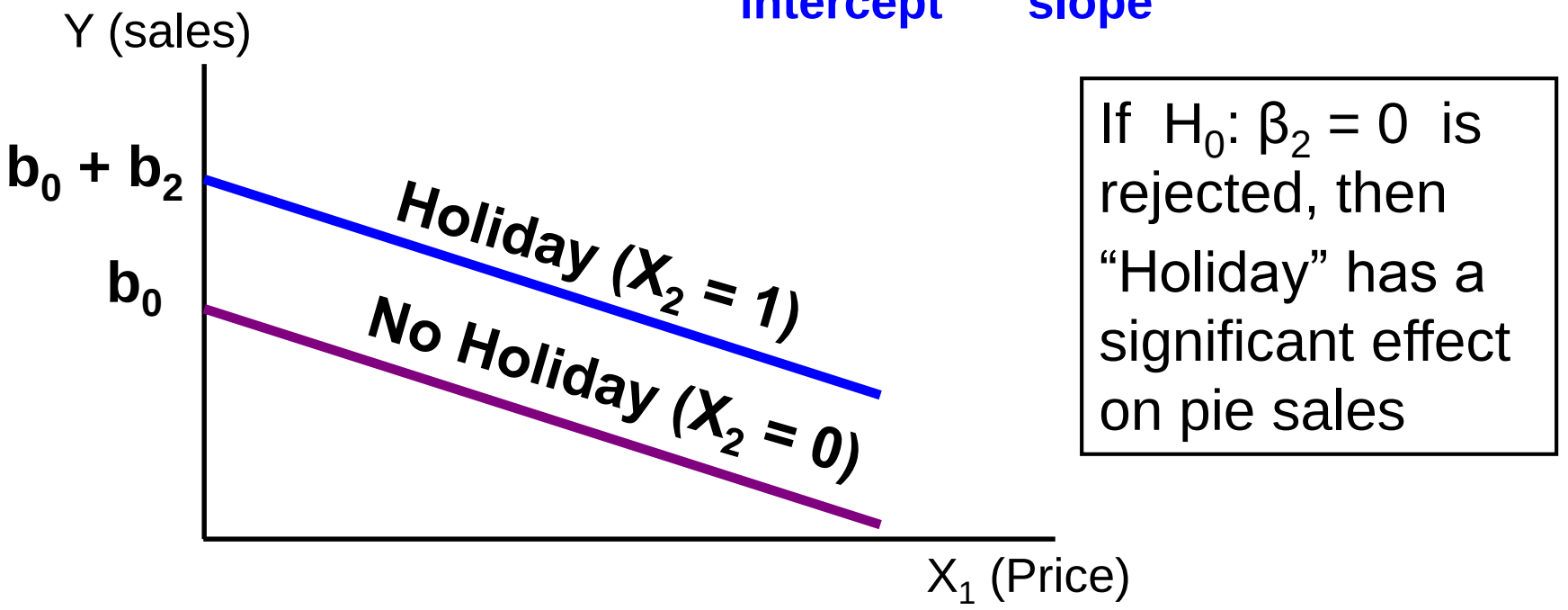
- A dummy variable is a categorical independent variable with two levels:
  - yes or no, on or off, male or female
  - coded as 0 or 1
- Assumes the slopes associated with numerical independent variables do not change with the value for the categorical variable
- If more than two levels, the number of dummy variables needed is (number of levels - 1)

Dummy-Variable Example (with 2 Levels)

$$\hat{Y} = b_0 + b_1X_1 + b_2(1) = (b_0 + b_2) + b_1X_1$$ **Holiday**

$$\hat{Y} = b_0 + b_1X_1 + b_2(0) = b_0 + b_1X_1$$ **No Holiday**

Different intercept **Same slope**



If $H_0: \beta_2 = 0$ is rejected, then "Holiday" has a significant effect on pie sales

Interpreting the Dummy Variable Coefficient(with 2 Levels)

Example: $\widehat{Sales} = 300 - 30(Price) + 15(Holiday)$

Sales: number of pies sold per week
Price: pie price in $

Holiday: $\begin{cases} 1 & \text{If a holiday occurred during the week} \\ 0 & \text{If no holiday occurred} \end{cases}$

$b_2 = 15$: On average, sales were 15 pies greater in weeks with a holiday than in weeks without a holiday, given the same price

Dummy-Variable Models (more than 2 Levels)

- The number of dummy variables is **one less than the number of levels**

- Example:

  Y = house price ; X_1 = square feet

- If style of the house is also thought to matter:

  Style = ranch, split level, colonial

Three levels, so two dummy variables are needed

Dummy-Variable Models (more than 2 Levels)

- Example: Let "colonial" be the default category, and let X_2 and X_3 be used for the other two categories:

  Y = house price

  X_1 = square feet

  X_2 = 1 if ranch, 0 otherwise

  X_3 = 1 if split level, 0 otherwise

The multiple regression equation is:

$$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2 + b_3 X_3$$

Interpreting the Dummy Variable Coefficients (3 Levels)

Consider the regression equation:

$$\hat{Y} = 20.43 + 0.045X_1 + 23.53X_2 + 18.84X_3$$

For a colonial: $X_2 = X_3 = 0$

$$\hat{Y} = 20.43 + 0.045X_1$$

For a ranch: $X_2 = 1; X_3 = 0$

$$\hat{Y} = 20.43 + 0.045X_1 + 23.53$$

With the same square feet, a ranch will have an estimated average price of 23.53 thousand dollars more than a colonial.

For a split level: $X_2 = 0; X_3 = 1$

$$\hat{Y} = 20.43 + 0.045X_1 + 18.84$$

With the same square feet, a split-level will have an estimated average price of 18.84 thousand dollars more than a colonial.

LOGISTIC REGRESSION

Multiple Linear Regression

We model the mean of a numeric response as linear combination of the predictors themselves or some functions based on the predictors, i.e.

$$E(Y|X) = \beta_0 + \beta_1 X_1 + \ldots. + \beta_n X_n + \epsilon$$

Here the terms in the model are the predictors

$$E(Y|X) = \beta_0 + \beta_1 f_1(X) + \ldots. + \beta_k f_k(X) + \epsilon$$

Here the terms in the model are k different functions of the n predictors.

Multiple Linear Regression

For the classic multiple regression model

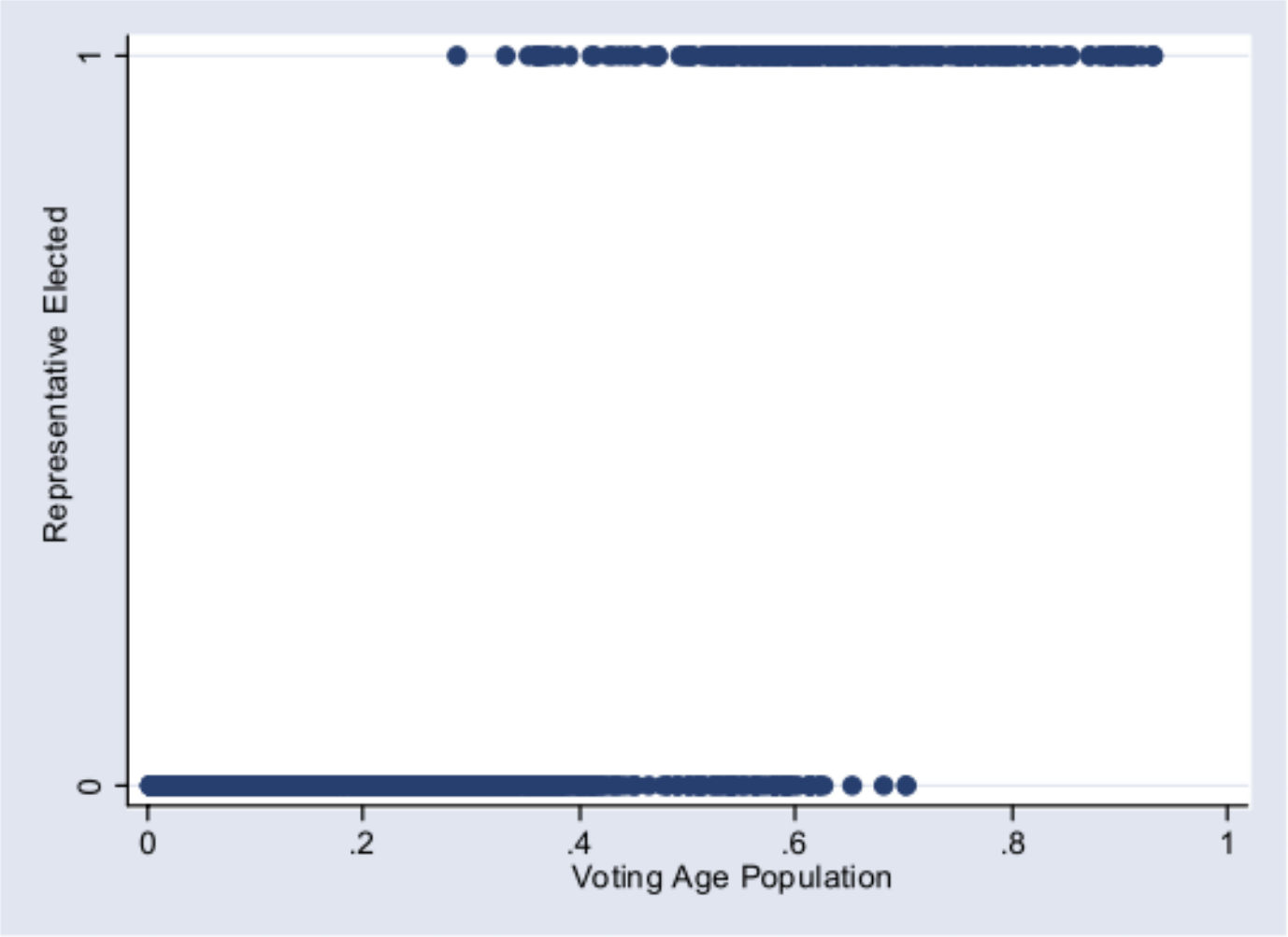
$$E(Y|X) = \beta_0 + \beta_1 X_1 + \ldots . + \beta_n X_n + \epsilon$$

- The regression coefficients β_i represent the estimated change in the mean of the response *Y* associated with a unit change in X_i while the other predictors are held constant.
- They measure the association between *Y* and X_i **adjusted** for the other predictors in the model.

Non-linear Estimation

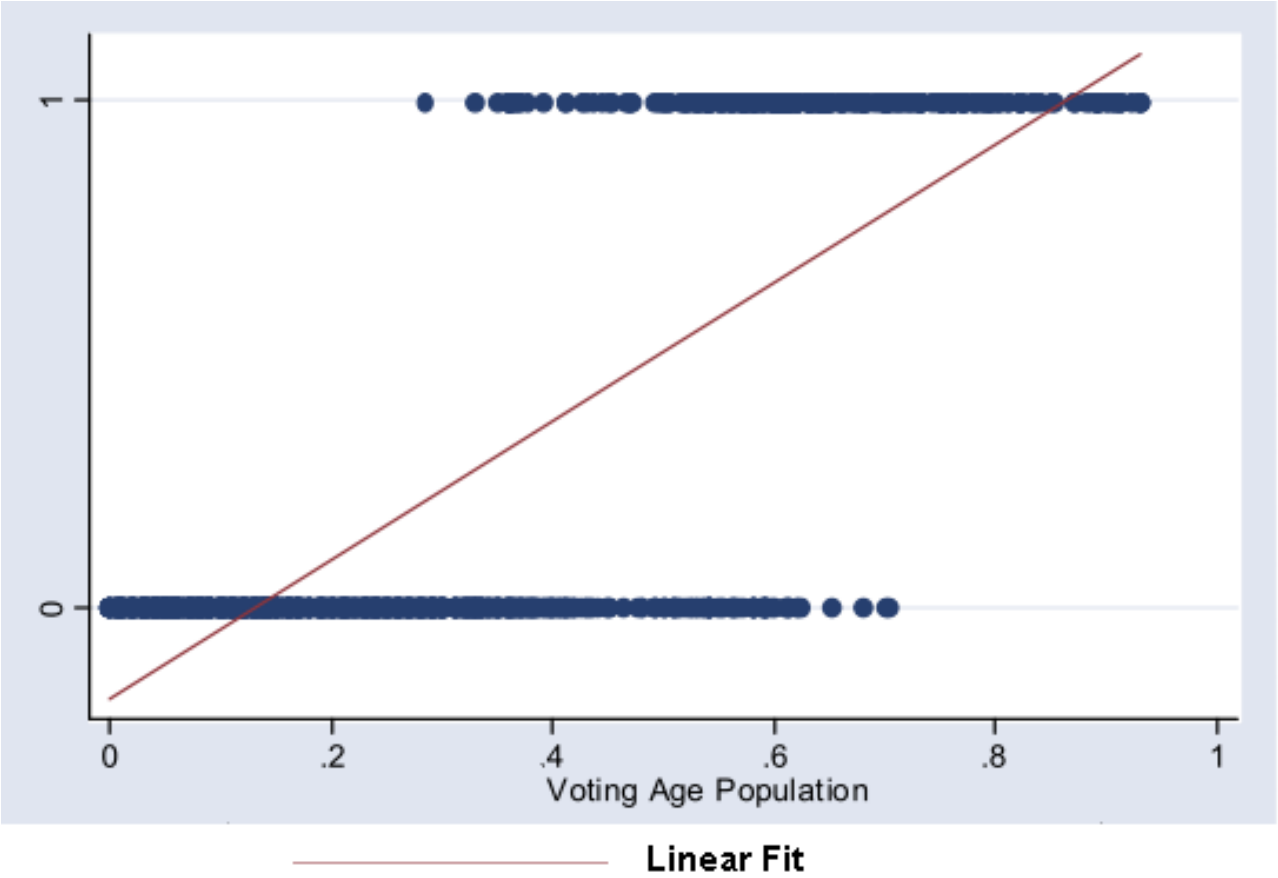
- In all these models Y, the dependent variable, was continuous.
  - Independent variables could be dichotomous (dummy variables), but not the dependent variable
- we will explore non-linear estimation with dichotomous Y variables.
- These arise in many data problems that we intend to model
  - Customer purchases item online: Yes/No
  - Credit scoring[Good customer]: YES/No
  - Students will pass mathematics courses: Yes/No
  - Involved in an Armed Conflict: Yes/No

Dichotomous Target Variables



Linear Fit: Dichotomous target Variable

- The line doesn't fit the data very well
- And if we take values of Y between 0 and 1 to be probabilities, this doesn't make sense



Logistic Regression

- Models relationship between set of variables X_i
  - Dichotomous (yes/no, smoker/nonsmoker)
  - Categorical (social class, race, ...)
  - Continuous (age, weight, gestational age)
  - Dichotomous categorical response variable *Y*

Example: Success/Failure, Remission/No Remission, Survived/Died, CHD/No CHD, Low Birth Weight/Normal Birth Weight, etc.

GLM model

- It is extension of the linear model framework, which includes dependent variables which are non-normal also.
- These models comprise a linear combination of input features.
- The mean of the response variable is related to the linear combination of input features via a link function.
- The response variable is considered to have an underlying probability distribution belonging to the family of exponential distributions such as binomial distribution, Poisson distribution, or Gaussian distribution.

LR assumption

- The response variable must follow a binomial distribution.
- Logistic Regression assumes a linear relationship between the independent variables and the link function (logit).
- The dependent variable should have mutually exclusive and exhaustive categories.

Logistic Regression Types

- **Multinomial Logistic Regression:**
  - target variable has K = 4 classes.
  - This technique handles the multi-class problem by fitting K-1 independent binary logistic classifier model.
- **Ordinal Logistic Regression:**
  - when the target variable is ordinal in nature.
  - Example predict years of work experience (1,2,3,4,5, etc). So, there exists an order in the value, i.e., 5>4>3>2>1.
  - when we train K -1 models, Ordinal Logistic Regression builds a single model with multiple threshold values.

What is binomial distribution?

- There must be a fixed number of trials denoted by **n**,
  - i.e. in the data set, there must be a fixed number of rows.
- Each trial can have only two outcomes;
  - i.e., the response variable can have only two unique categories.
- The outcome of each trial must be independent of each other
  - i.e., the unique levels of the response variable must be independent of each other.
- The probability of success (p) and failure (q) should be the same for each trial.

Logistic Regression derivation

probabilities always lie between 0 and 1. In other words, we can say:

- The response value must be positive.
- It should be lower than 1.

logistic function

$$P(Y=1|X) = \frac{e^{(\beta_o+\beta_1 x)}}{e^{(\beta_o+\beta_1 x)}+1}$$

$$\implies \quad p = \frac{e^{(\beta_o+\beta_1 x)}}{e^{(\beta_o+\beta_1 x)}+1}$$

$$\implies p(e^{(\beta_o+\beta_1 x)}+1) = e^{(\beta_o+\beta_1 x)}$$

$$\implies p.e^{(\beta_o+\beta_1 x)} + p = e^{(\beta_o+\beta_1 x)}$$

$$\implies p = e^{(\beta_o+\beta_1 x)} - p.e^{(\beta_o+\beta_1 x)}$$

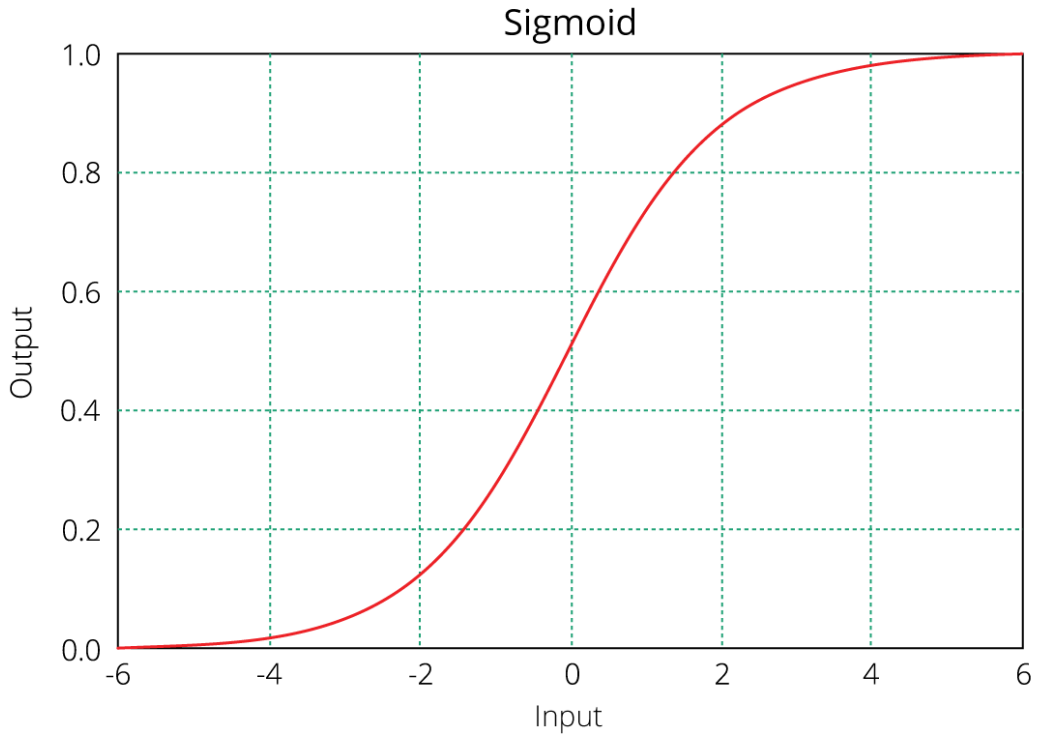
$$\implies p = e^{(\beta_o+\beta_1 x)}(1-p)$$

$$\implies \frac{p}{1-p} = e^{(\beta_o+\beta_1 x)}$$

$$\implies \ln(\tfrac{p}{1-p}) = \beta_0 + \beta_1 x$$

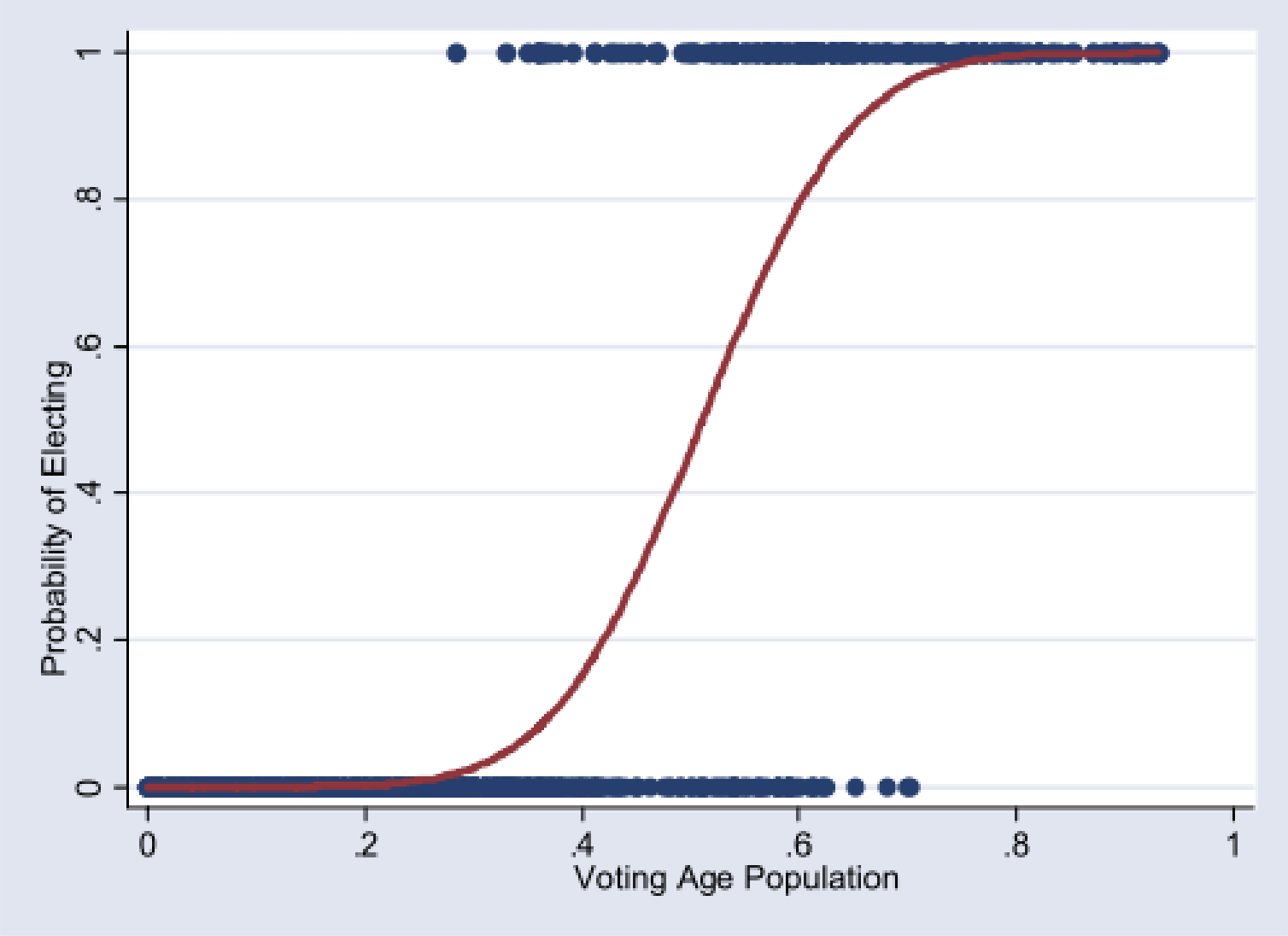
Sigmoid and Maximum likelihood

- This link function follows a sigmoid function which limits its range of probabilities between 0 and 1.
- the regression coefficients explain the change in log(odds) in the response for a unit change in predictor.
- To find the value of coefficients (βo,β1) the predicted probabilities are as close to the observed probabilities as possible. In other words, for a binary classification (1/0), maximum likelihood will try to find values of βo and β1 such that the resultant probabilities are closest to either 1 or 0.



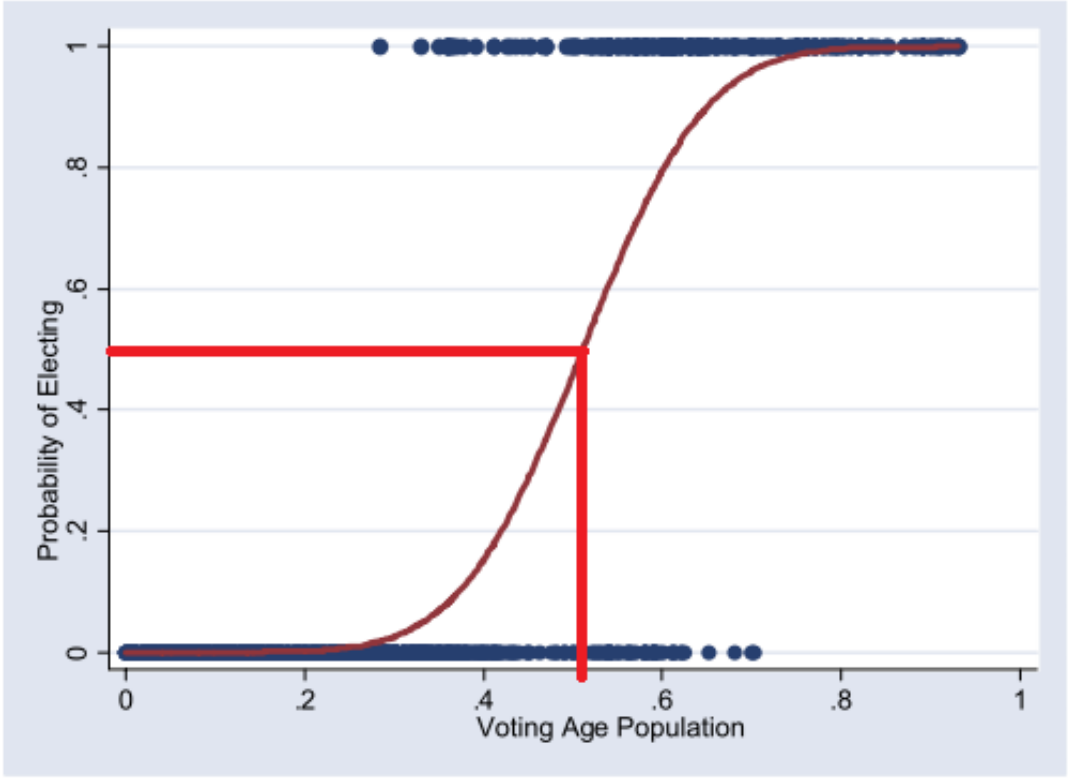
$$\ell(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'}))$$

Probit Estimation



Probit Estimation

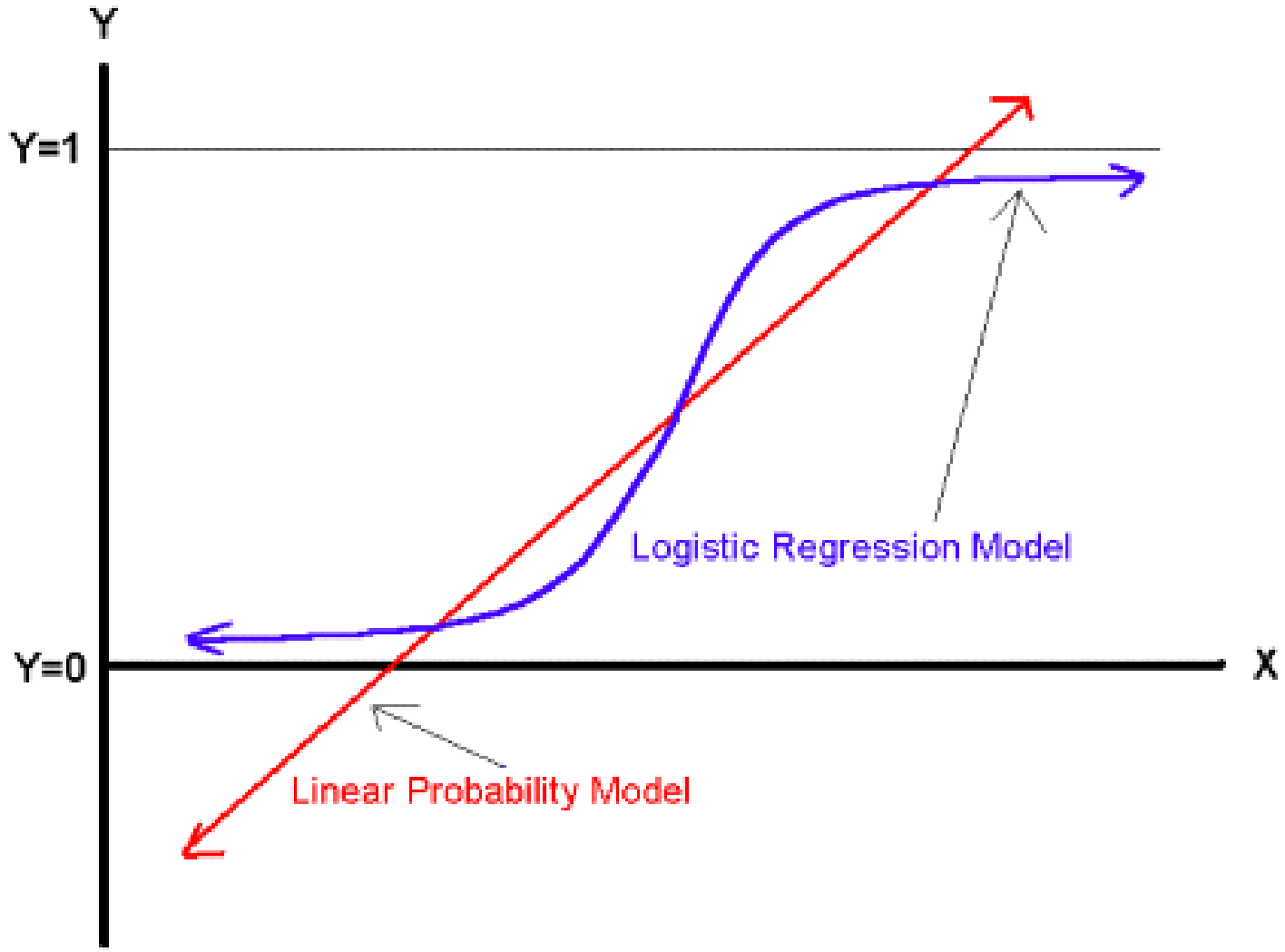
- This fits the data much better than the linear estimation
- Always lies between 0 and 1
- Can estimate, for instance, the BVAP at which Pr(Y=1) = 50%
- This is the “point of equal opportunity”



Dependent Variable

- Let's return to the problem of transforming Y from {0,1} to the real line
- If some event occurs with probability p, then the
- Odds of it happening are O(p) = p/(1-p)
  - p = 0 $\rightarrow$ O(p) = 0
  - p = ¼ $\rightarrow$ O(p) = 1/3 ("Odds are 1-to-3 against")
  - p = ½ $\rightarrow$ O(p) = 1 ("Even odds")
  - p = ¾ $\rightarrow$ O(p) = 3 ("Odds are 3-to-1 in favor")
  - p = 1 $\rightarrow$ O(p) = ∞

Comparing the Logistic & Linear Regression Models



Logistic Regression

$$Y = \text{logit}(Y) = X\,\beta$$

Note:

$$\Pr(Y = 1|X) = \frac{e^{X\,\beta}}{1+e^{X\,\beta}}$$

Where,

- exp or e is the exponential function (e=2.71828…)
- p is probability that the event y occurs given x, and can range between 0 and 1
- p/(1-p) is the **"odds ratio"**
- ln[p/(1-p)] is the log odds ratio, or "logit"
- all other components of the regression model are the same

Example: Prediction of Graduation Result

- Our target is to predict which female students graduated with honors or not.

	Female	Read	Write	math	hon	femalexmath
1	0	57	52	41	0	0
2	1	68	59	53	0	53
3	0	44	33	54	0	0
4	0	63	44	47	0	0
5	0	47	52	57	0	0
6	0	44	52	51	0	0

- **Understanding Data**:

  Data consists of **gender** (female=1 if female), **reading scores**, **writing scores**, **math scores**, **honors status** (hon=1 if graduated with honors) and **femalexmath** showing the math score for female alone.

Example: Prediction of Graduation Result

Analyzing data & Forming Crosstab:

The crosstab of the variable **hon** with female shows that there are 109 female and 91 male; 32 of those 109 females secured honors.

hon	female male	female	Total
0	74	77	151
1	17	32	49
Total	91	109	200

Calculating Probability:

Probability of females sec

= 32/109

$$= \frac{\text{\# females securing honours}}{\text{Total \#females}}$$

Example: Prediction of Graduation Result

Calculating Odds:

- Odds of an event is the probability of that event occurring (probability that y=1), divided by the probability that it does not occur.
- So odds of females securing honors:

$$= \frac{\text{probability of females securing honours}}{\text{probability of females not securing honours}}$$

$$= \frac{\text{[\# females securing honours] } \textbf{/} \text{ [Total \# females]}}{\text{[\# females not securing honours] } \textbf{/} \text{ [Total \# females]}}$$

$$= \frac{32 / 109}{77 / 109}$$

$$= 32 / 77$$

$$= 0.4155 = 0.42$$

Interpretation : For every 32 females that secure honors, there are 77 females that do not secure honors.

Example: Prediction of Graduation Result

Log odds:
The Logit or log-odds of an event is the log of the odds. (base 'e').

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$$

Log-odds of females securing honours: $=\log_e(0.4155) = -0.87$

Odds ratio:

- It is the ratio of 2 odds.
- These 2 odds are obtained at 2 different values of x,
- The odds obtained when x=0 and x=1 (where x=0 denotes male and x=1 denotes female).

Example: Prediction of Graduation Result

Find the odds ratio of graduating with honours for females and males.

$$OR = \frac{\text{Odds at } x = 1}{\text{Odds at } x = 0}$$

$$= \frac{\text{Odds of females securing honours}}{\text{Odds of males securing honours}}$$

$$= \frac{\text{[probability of females securing honours] / [probability of females not securing honours]}}{\text{[probability of males securing honours] / [probability of male not securing honours]}}$$

$$= \frac{\frac{32}{109}}{\frac{77}{109}} \Big/ \frac{\frac{17}{91}}{\frac{74}{91}}$$

$$= 1.82$$

Example: Prediction of Graduation Result

Probability Calculation:

To calculate the effect of being female on the probability of graduating with honours.

Where :

$$y = \text{logit}(p) = \log(p/(1-p)) = \beta_0 + \beta_1 * x1 + \ldots + \beta_k * xk = Bt.X$$

B0,B1,..Bk are estimated as the 'log-odds' of a unit change in the input feature it is associated with.

As B0 is the coefficient not associated with any input feature, B0= log-odds of the reference variable, x=0 (ie x=male). ie

Here, B0= log[odds(male graduating with honours)]

As B1 is the coefficient of the input feature 'female',

B1= log-odds obtained with a unit change in x= female.

B1= log-odds obtained when x=female and x=male.

$$B1 = \log \frac{\text{odds(females graduating with honours)}}{\text{odds(males graduating with honours)}}$$

Example: Prediction of Graduation Result

B0 = log[odds(males graduating with honours)]

$$= \log \frac{\text{probability of males securing honours}}{\text{probability of males not securing honours}}$$

$$= \log \frac{[\#\text{males securing honours}] \textbf{ / } [\text{Total \#males}]}{[\#\text{ males not securing honours}] \textbf{ / } [\text{Total \# males }]}$$

= log [(17/91) / (74/91)]

=log (0.23)

= -1.47

$$B1 = \log \frac{\text{odds(females graduating with honours)}}{\text{odds(males graduating with honours)}}$$

From the calculation of 'odds ratio(OR)',

B1= log (1.82)

B1= 0.593

Thus, LogR equation becomes

y= -1.47 + 0.593* female

where the value of female is substituted as 0 for male and 1 for female.

Example: Prediction of Graduation Result

Now, let us find out the probability of a female securing honours when there is only 1 input feature present- 'female'.

Substitute female=1 in: **y= -1.47 + 0.593* female**

y=log[odds(female)]= -1.47 + 0.593*1 = -0.877

Log-odds = -0.877.

Odds= e^ (log odds)= e^ (-0.877)= 0.416

Probability is $= \frac{e^{\wedge}(Bt.X)}{1+e^{\wedge}(Bt.X)}$

$= \frac{(-0.877)}{1+e^{\wedge}(-0.877)}$

$= 0.416/\ 1.416$

$= 0.29$

Interpretation: Probability of a female securing honours when there is only 1 input feature present-'female', is 0.29.

Confusion Matrix

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

True Positive (TP): These are cases in which we predicted yes and actually its yes.

True Negative (TN): These are cases in which we predicted no, and actually its no.

False Positive (FP): We predicted yes, but actually its no. (Also known as "Type I error.")

False Negative (FN): We predicted no, but actually its yes. (Also known as "Type II error.")

Confusion Matrix

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Accuracy: Overall, how often is the classifier correct?
(TP+TN)/total = (100+50)/165 = 0.91

Misclassification Rate: Overall, how often is it wrong?
(FP+FN)/total = (10+5)/165 = 0.09
Equivalent to 1 minus Accuracy. Also known as **"Error Rate."**

True Positive Rate: When it's actually yes, how often does it predict yes?
TP/actual yes =100/105 = 0.95
Also known as **"Sensitivity" or "Recall."**

False Positive Rate: When it's actually no, how often does it predict yes?
FP/actual no =10/60 = O.17

Specificity: When it's actually no, how often does it predict no?
TN/actual no 50/60 - 0.83
Equivalent to 1 minus False Positive Rate

Precision: When it predicts yes, how often is it correct?
TP/predicted yes = 100/110 = 0.91

Confusion Matrix

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Kappa = (Observed Accuracy - Expected Accuracy) / (1 - Expected Accuracy)

Observed Accuracy = (TP + TN)/Total
= (100+50)/165 = 0.91

Expected Accuracy =[(Actual Yes/Total * Predicted Yes /Total)] +
[(Actual No/Total * Predicted No /Total)]

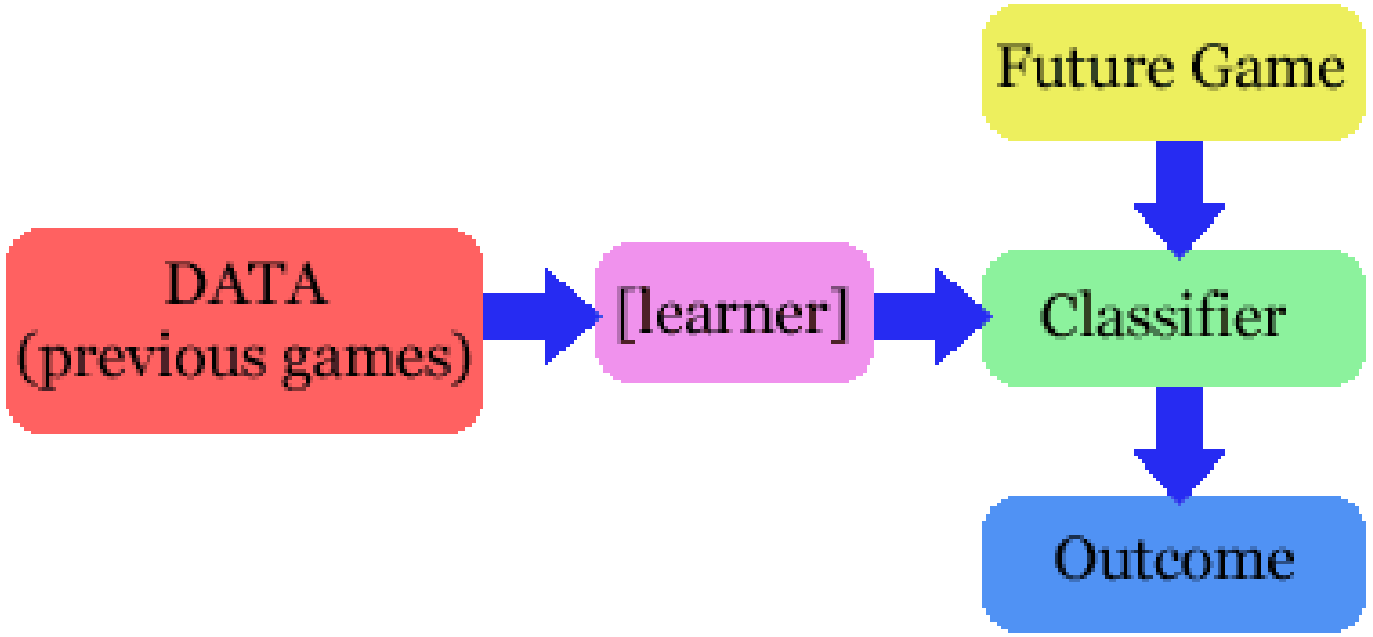
= [(105/165) * (110/165)] + [(60/165) * (55/165)]
= [0.64 * 0.67] + [0.34 * 0.33]
=0.43+0.11
=0.54

Kappa = (0.91-0.54)/(1-0.54)
= 0.37/0.46
= 0.81

Decision Tree Algorithms

The Problem

- Given a set of training cases/objects and their attribute values, try to determine the target attribute value of new examples.
  - Classification
  - Prediction



Example

- Bank - loan application
- Classify application
  - approved class
  - denied class
- Criteria - Target Class approved if 3 binary attributes have certain value:
  - (a) borrower has good credit history (credit rating in excess of some threshold)
  - (b) loan amount less than some percentage of collateral value (e.g., 80% home value)
  - (c) borrower has income to make payments on loan
- Possible scenarios = $3^2 = 8$
  - If the parameters for splitting the nodes can be adjusted, the number of scenarios grows exponentially.

How Decision Tree Works?

- Decision rules - partition sample of data
- Terminal node (leaf) indicates the class assignment
- Tree partitions samples into mutually exclusive groups
- One group for each terminal node
- All paths
  - start at the root node
  - end at a leaf
- Each path represents a decision rule
  - joining (AND) of all the tests along that path
  - separate paths that result in the same class are disjunctions (ORs)
- All paths - mutually exclusive
  - for any one case - only one path will be followed
  - false decisions on the left branch
  - true decisions on the right branch

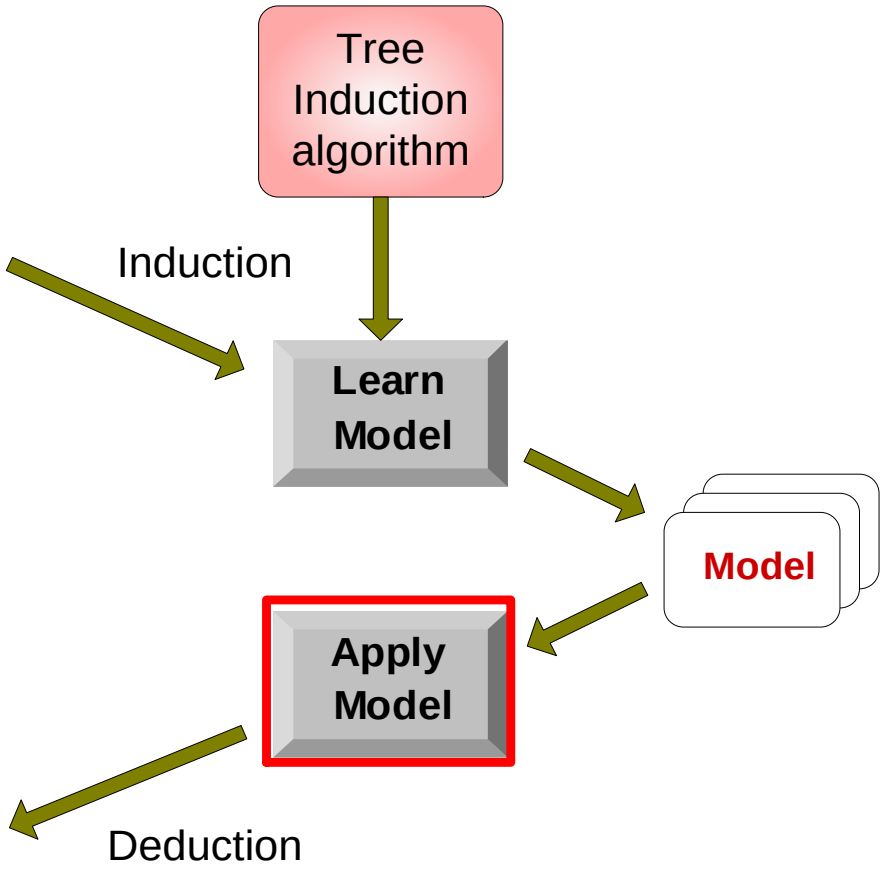
Classification Task using Decision Tree

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

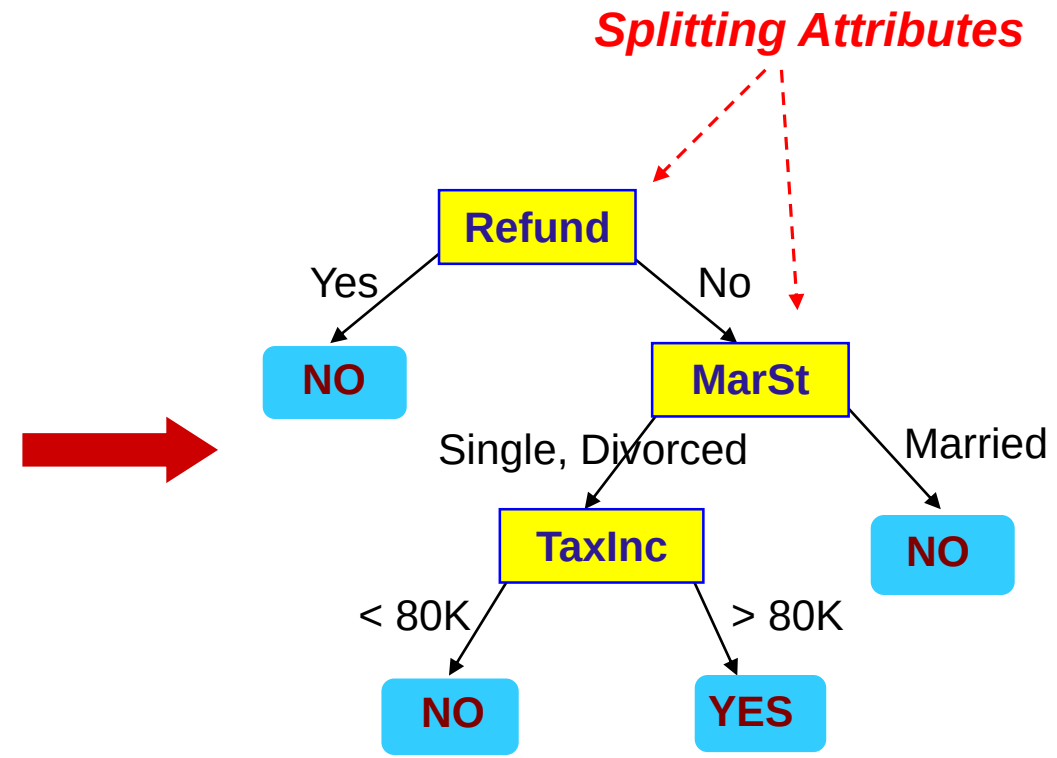
Test Set



Example Splitting Attributes

	categorical	categorical	continuous	class
Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

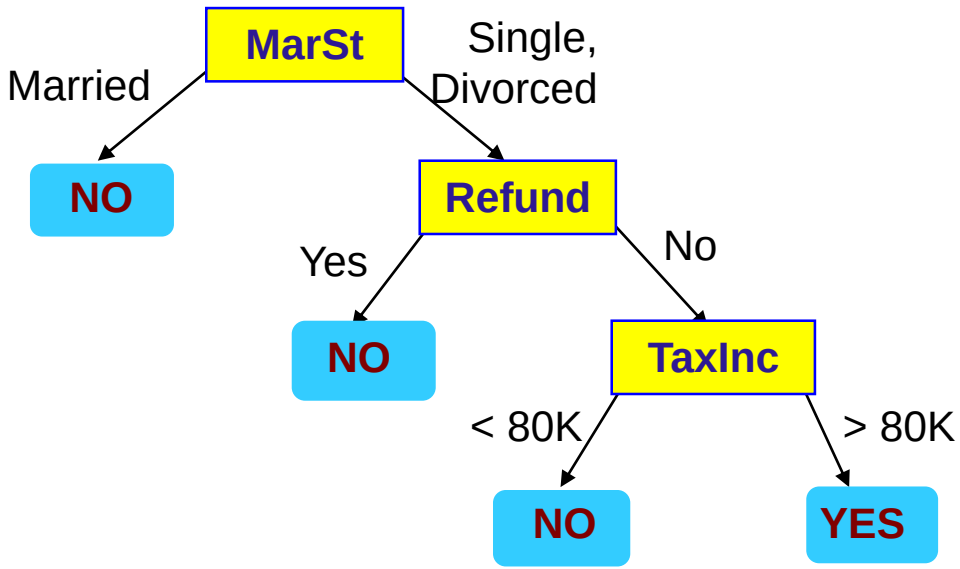
Training Data



Model: Decision Tree

Example Splitting Attributes

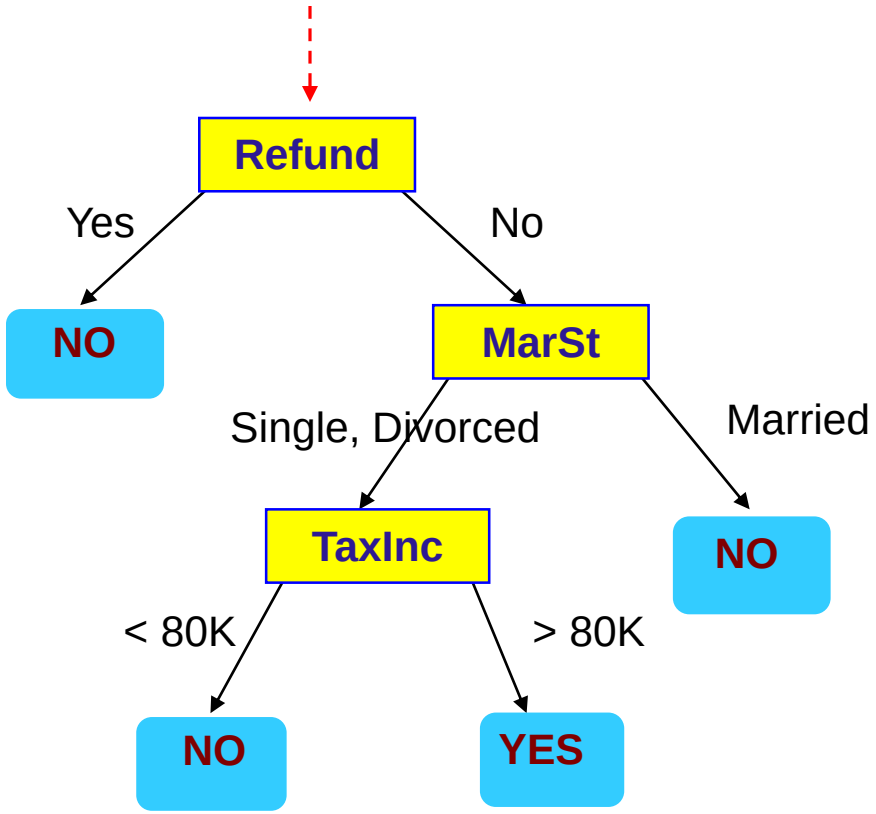
	categorical	categorical	continuous	class
Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**



There could be more than one tree that fits the same data!

Apply Model to Test Data

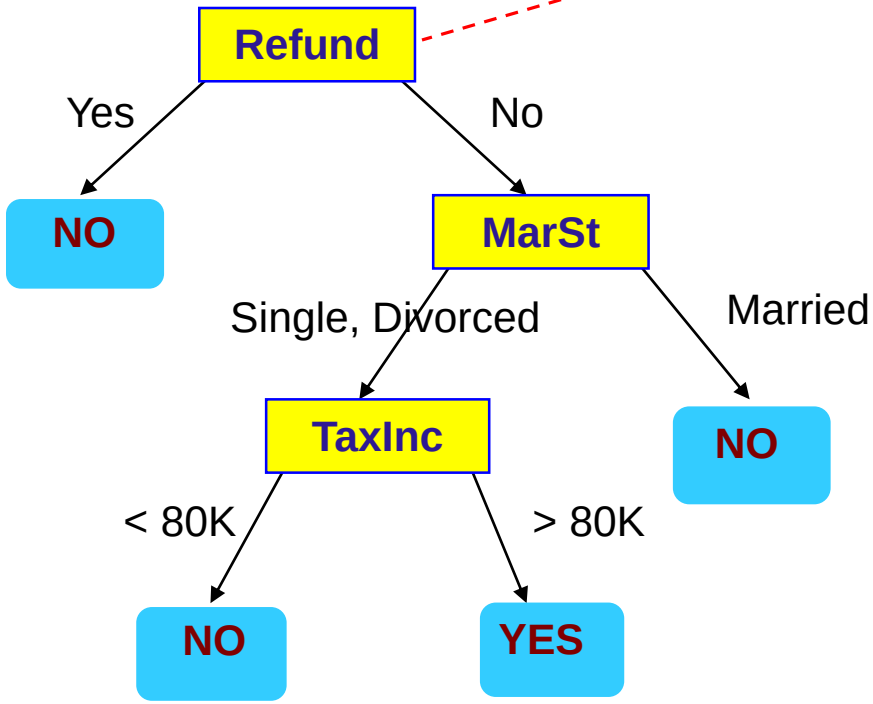
Start from the root of tree.



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

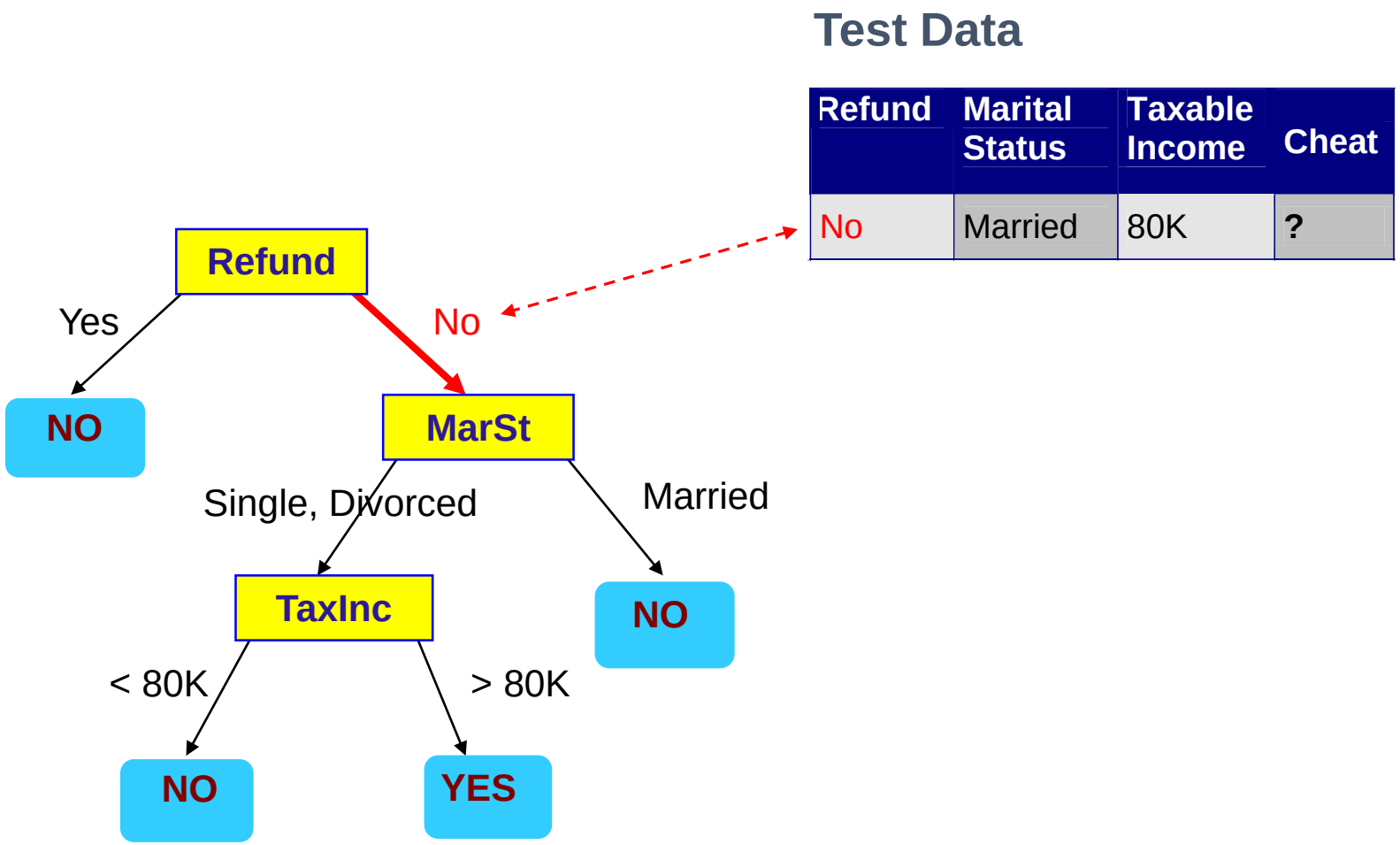
Apply Model to Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

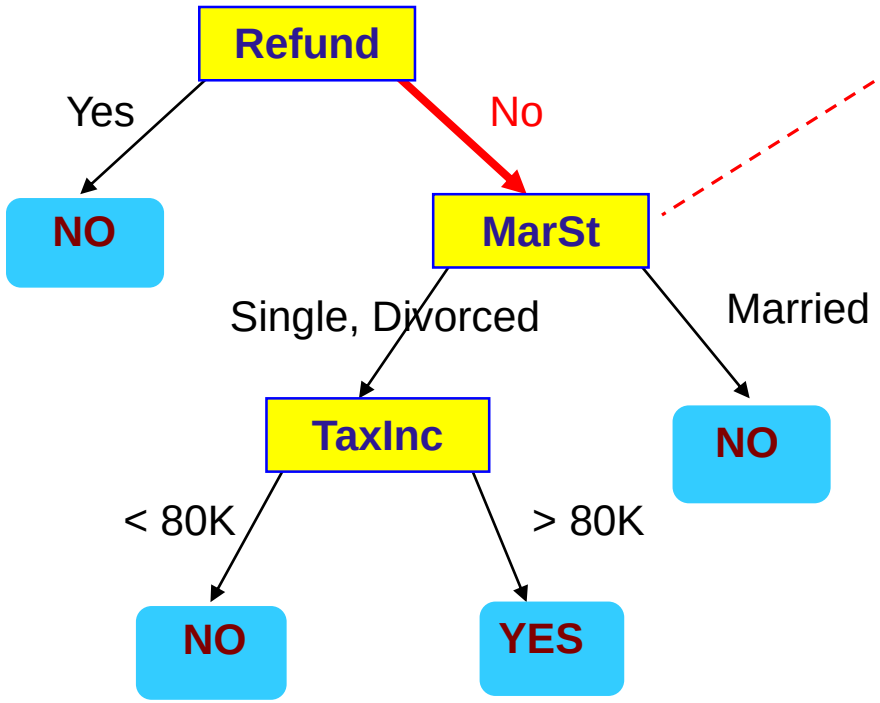
Apply Model to Test Data



Apply Model to Test Data

Test Data

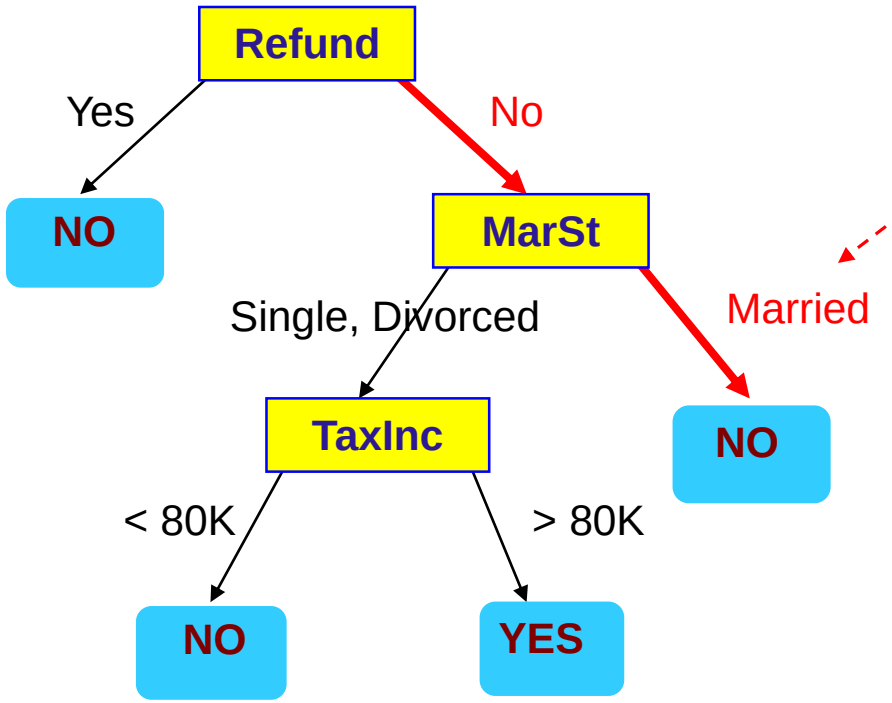
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Apply Model to Test Data

Test Data

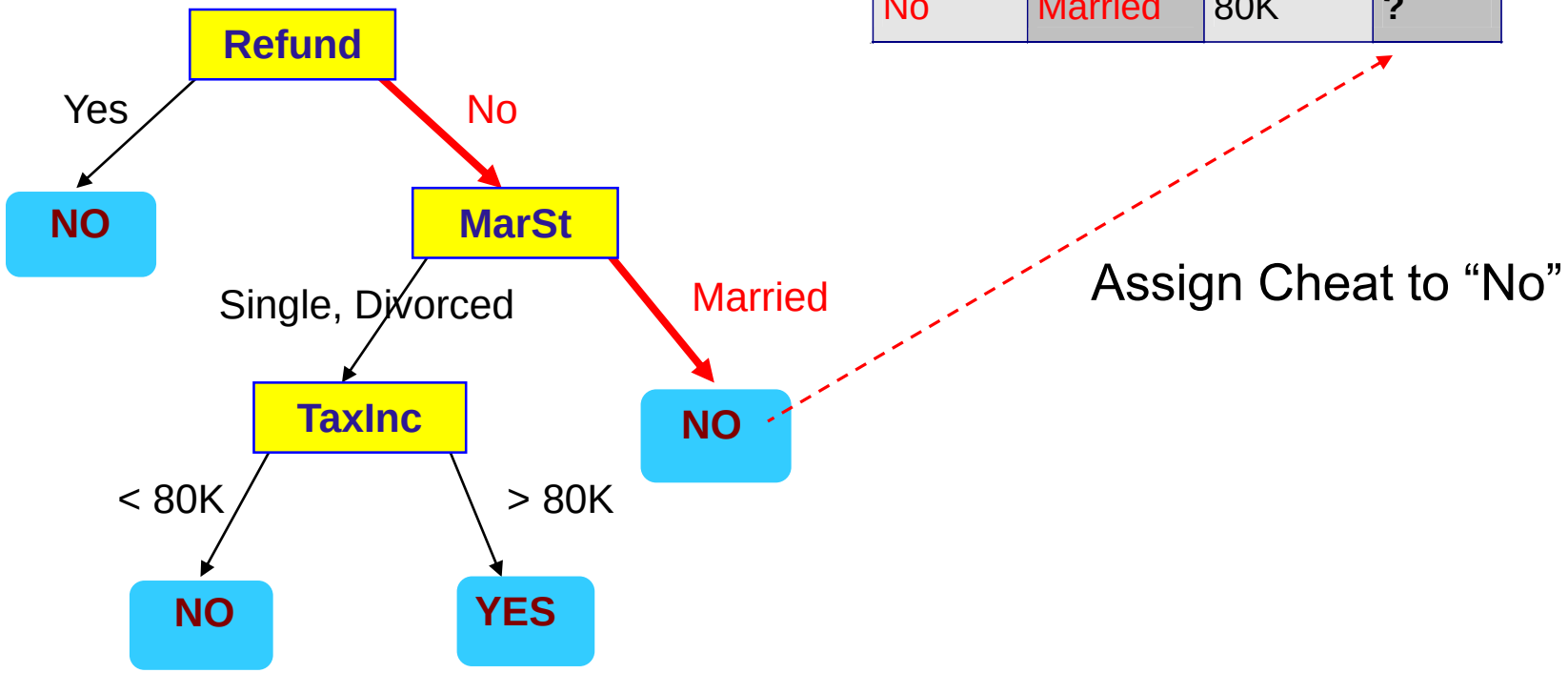
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



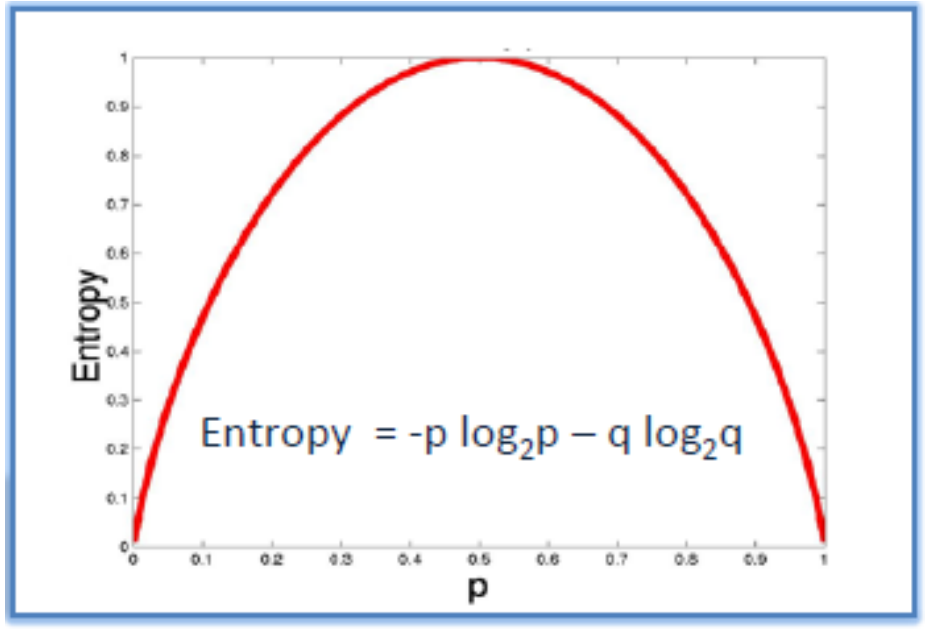
Assign Cheat to “No”

ENTROPY: MEASURE OF RANDOMNESS

- A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values (homogenous).
- ID3 algorithm uses entropy to calculate the homogeneity of a sample. If the sample is completely homogeneous the entropy is zero and if the sample is an equally divided it has entropy of one.

For a binary classification problem

- If all examples are positive or all are negative then entropy will be ***zero*** i.e, low.
- If half of the examples are of positive class and half are of negative class then entropy is **one** i.e, high.

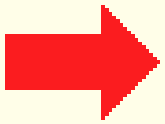


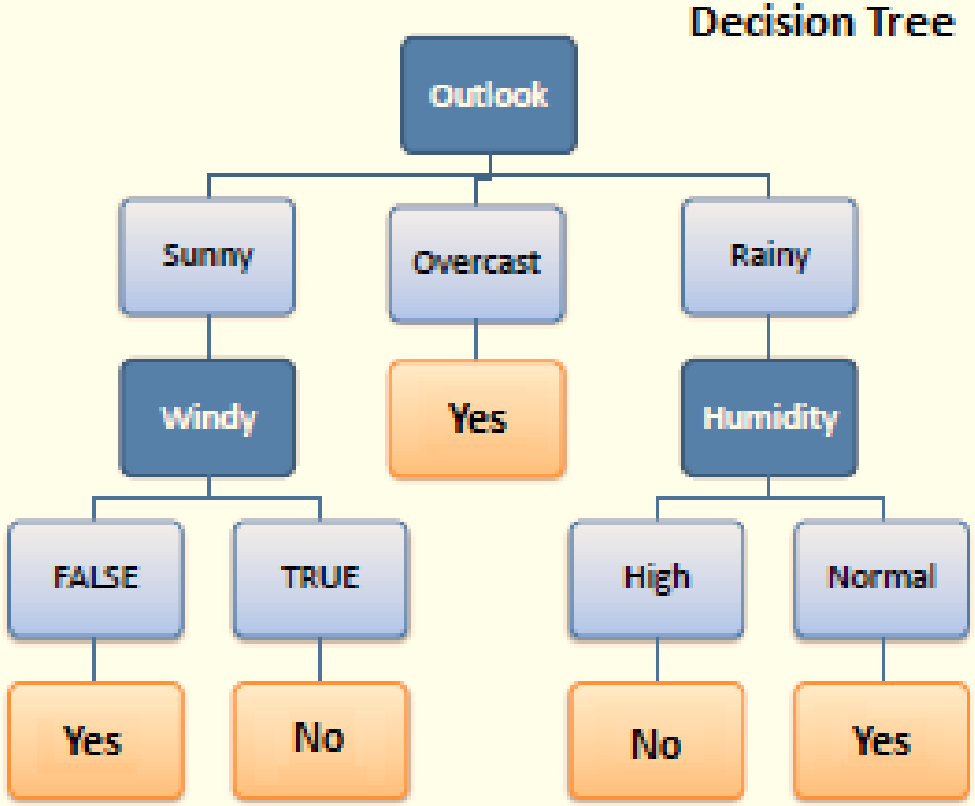
Entropy = $-0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$

Example Data

Predictors | Target

Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No





CALCULATION OF ENTROPY & IG

Entropy

Entropy is calculated based on proportion of Target Values. And formula is as follow

$$\text{Entropy} = \sum_{i=1}^{T} -p_i \log(p_i)$$

i is Level of Target Variable

Entropy

Entropy $H(S)$ is a measure of the amount of uncertainty in the (data) set S (i.e. entropy characterizes the (data) set S).

$$H(S) = \sum_{c \in C} -p(c) \log_2 p(c)$$

Where,

- S – The current (data) set for which entropy is being calculated (changes every iteration of the ID3 algorithm)
- C – Set of classes in S C={ yes, no }
- $p(c)$ – The proportion of the number of elements in class c to the number of elements in set S

When $H(S) = 0$, the set S is perfectly classified (i.e. all elements in S are of the same class).

In ID3, entropy is calculated for each remaining attribute. The attribute with the **smallest** entropy is used to split the set S on this iteration. The higher the entropy, the higher the potential to improve the classification here.

CALCULATION OF ENTROPY & IG

To build a decision tree, we need to calculate two types of entropy using frequency tables as follows:

a) Entropy using the frequency table of one attribute:

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Play Golf	
Yes	No
9	5

$$Entropy(S) = -\frac{p}{p+n}\log_2\left(\frac{p}{p+n}\right) - \frac{n}{p+n}\log_2\left(\frac{n}{p+n}\right)$$

Entropy(PlayGolf) = Entropy (5,9)
= Entropy (0.36, 0.64)
= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)
= 0.94

C ={yes,no}

Out of 14 instances, 9 are classified as yes, and 5 as no
pyes = −(9/14)*log2(9/14) = 0.41
pno = −(5/14)*log2(5/14) = 0.53

H (S) = pyes + pno = 0.94

In this target variable No has 36% values and Yes has 64%.
P (Target=Yes) = P1 = - 0.64 *log (0.64)

P1 = (-)0.64*-0.643856189774724 7

P1 = 0.41206796096

P (Target=No) = P2 = - 0.36 *log (0.36)

P2 = (-)0.36*-1.4739311883324124

P2=0.530615196

Now entropy for the node is calculated as
Entropy = P1+P2
0.41206796096+0.530615196=0.94

CALCULATION OF ENTROPY & IG

b) Entropy using the frequency table of two attributes:

$$E(T,X) = \sum_{c \in X} P(c)E(c)$$

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

E(PlayGolf, Outlook) = **P**(Sunny)*__E__(3,2) + **P**(Overcast)*__E__(4,0) + **P**(Rainy)*__E__(2,3)

= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971

= 0.693

Information Gain

The information gain is based on the decrease in entropy after a dataset is split on an attribute.
Constructing a decision tree is all about finding attribute that returns the highest information gain (i.e., the most homogeneous branches).

Step 1: Calculate entropy of the target.

Entropy(PlayGolf) = Entropy (5,9)

= Entropy (0.36, 0.64)

= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)

= 0.94

Information Gain

Step 2: The dataset is then split on the different attributes. The entropy for each branch is calculated. Then it is added proportionally, to get total entropy for the split. The resulting entropy is subtracted from the entropy before the split. The result is the Information Gain, or decrease in entropy.

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1
Gain = 0.029			

$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

G(PlayGolf, Outlook) = **E**(PlayGolf) – **E**(PlayGolf, Outlook)

= 0.940 – 0.693 = 0.247

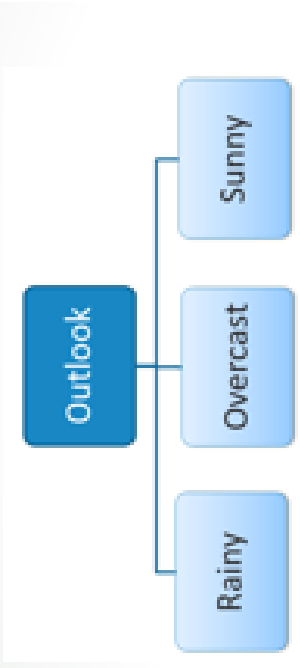
		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1
Gain = 0.152			

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3
Gain = 0.048			

Information Gain

Step 3: Choose attribute with the largest information gain as the decision node, divide the dataset by its branches and repeat the same process on every branch.

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

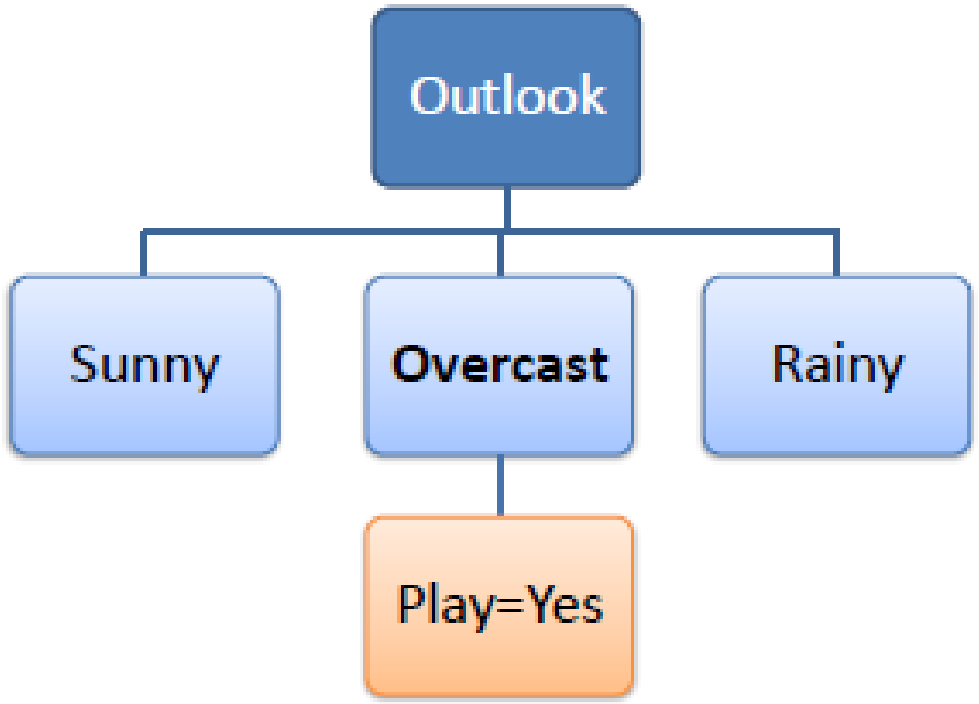


Outlook	Temp	Humidity	Windy	Play Golf
Sunny	Mild	High	FALSE	Yes
Sunny	Cool	Normal	FALSE	Yes
Sunny	Cool	Normal	TRUE	No
Sunny	Mild	Normal	FALSE	Yes
Sunny	Mild	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Overcast	Cool	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Rainy	Mild	High	FALSE	No
Rainy	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes

Information Gain

Step 4a: A branch with entropy of 0 is a leaf node.

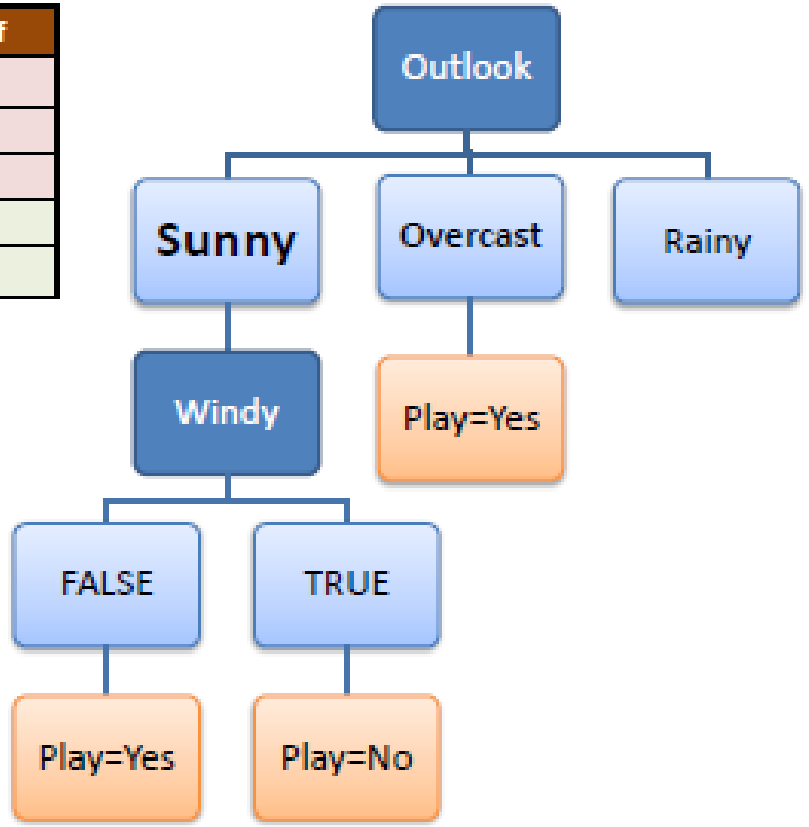
Temp	Humidity	Windy	Play Golf
Hot	High	FALSE	Yes
Cool	Normal	TRUE	Yes
Mild	High	TRUE	Yes
Hot	Normal	FALSE	Yes



Information Gain

Step 4b: A branch with entropy more than 0 needs further splitting.

Temp.	Humidity	Windy	Play Golf
Mild	High	FALSE	Yes
Cool	Normal	FALSE	Yes
Mild	Normal	FALSE	Yes
Cool	Normal	TRUE	No
Mild	High	TRUE	No



Decision Tree Rule

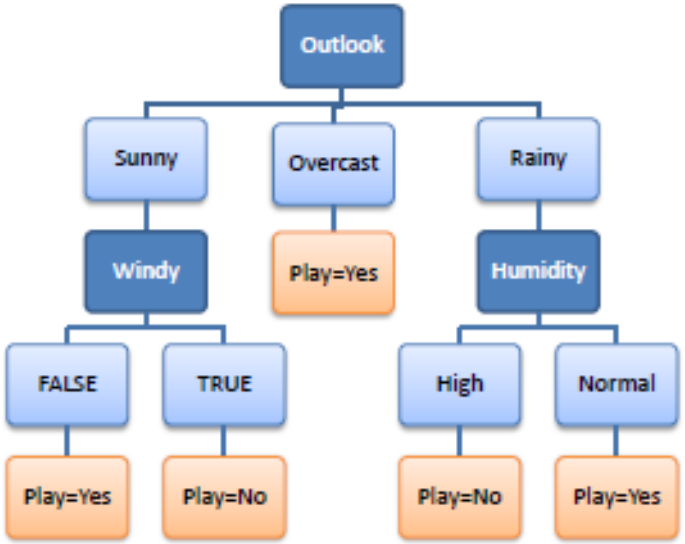
R_1: IF (Outlook=Sunny) AND (Windy=FALSE) THEN Play=Yes

R_2: IF (Outlook=Sunny) AND (Windy=TRUE) THEN Play=No

R_3: IF (Outlook=Overcast) THEN Play=Yes

R_4: IF (Outlook=Rainy) AND (Humidity=High) THEN Play=No

R_5: IF (Outlook=Rain) AND (Humidity=Normal) THEN Play=Yes



CART Decision Tree

GINI Index

$$Gini = \sum_{i \neq j} p(i)p(j)$$

i and j are levels of the target variable

Target variable is Binary variable which means it take two values (Yes and No). There can be 4 combinations.
|1|1|
|0|0|

- P(Target=1).P(Target=1) + P(Target=1).P(Target=0) + P(Target=0).P(Target=1) + P(Target=0).P(Target=0) = 1
- P(Target=1).P(Target=0) + P(Target=0).P(Target=1) = 1 – P^2(Target=0) – P^2(Target=1)

Gini Index for Binary Target variable is = 1 – P^2(Target=0) – P^2(Target=1)

$= 1 - \sum_{t=0}^{t=1} P_t^2$ p_t : Proportion of observations with target variable value t. In Binary t takes value 0 and 1.

CART Decision Tree

Target Variable Value	Count	%
Yes	346	74%
No	124	26%
Ovarall	470	100%

Gini Index = $1 - 0.74^2 - 0.26^2$
= 1-0.5476 -0.0676
= 0.3848

GINI of a split

GINI (s,t) = GINI (t) – P_L GINI (t_L) – P_R GINI (t_R)

Where

s : split

t : node

GINI (t) : Gini Index of input node t

P_L : Proportion of observation in Left Node after split, s

GINI (t_L) : Gini of Left Node after split, s

P_R : Proportion of observation in Right Node after split, s

GINI (t_R) : Gini of Right Node after split, s

CART Decision Tree

	Target Value		
Gender	**No**	**Yes**	**Total**
Female	6	2	8
Male	6	10	16
Total	**12**	**12**	**24**

Gini index for this node will be

= $1- (1/2)^2–(1/2)^2$

= 1- 0.25 -0.25

= 0.5

Now, let's calculate GINI index of the split using Gender variable.

GINI (s,t) = GINI (t) – P_L GINI (t_L) – P_R GINI (t_R)

GINI (t_L) = $1- (6/8)^2 – (2/8)^2$

= 1- 0.5625 -0.0625

= 0.375

GINI (t_R) = $1- (6/16)^2 – (10/16)^2$

= 1 - 0.140625 -0.390625

= 0.469

GINI (s,t) = 0.5 – (8/24)*0.375 – (16/24)*0.469

= 0.5 - 0.125 -0.313

= 0.0625

TERMINATION CRITERIA

- All the records at the node belong to one class
- A significant majority fraction of records belong to a single class
- The segment contains only one or very small number of records
- The improvement is not substantial enough to warrant making the split

PRUNING TREES

- The decision trees can be grown deeply enough to perfectly classify the training examples which leads to overfitting when there is noise in the data
- When the number of training examples is too small to produce a representative sample of the true target function.
- Practically, pruning is not important for classification

APPROACHES TO PRUNE TREE

- Three approaches
  - –Stop growing the tree earlier, before it reaches the point where it perfectly classifies the training data,
  - –Allow the tree to over fit the data, and then post-prune the tree.
  - –Allow the tree to over fit the data, transform the tree to rules and then post-prune the rules.

APPROACHES TO PRUNE TREE

- **Cost complexity pruning**

  J(Tree, S) = ErrorRate(Tree, S) + a |Tree|

  Play with several values a starting from 0

  Do a K-fold validation on all of them and find the best pruning α

Random Forest

Introduction

- Random Forest is an ensemble algorithm.
- **Ensemble algorithms** are those which combines more than one algorithms of same or different kind for classifying objects.
- In Random Forest, the method of **combining trees** is known as an **ensemble method**.
- Ensembling is nothing but a **combination of weak learners** (individual trees) to produce a **strong learner.**

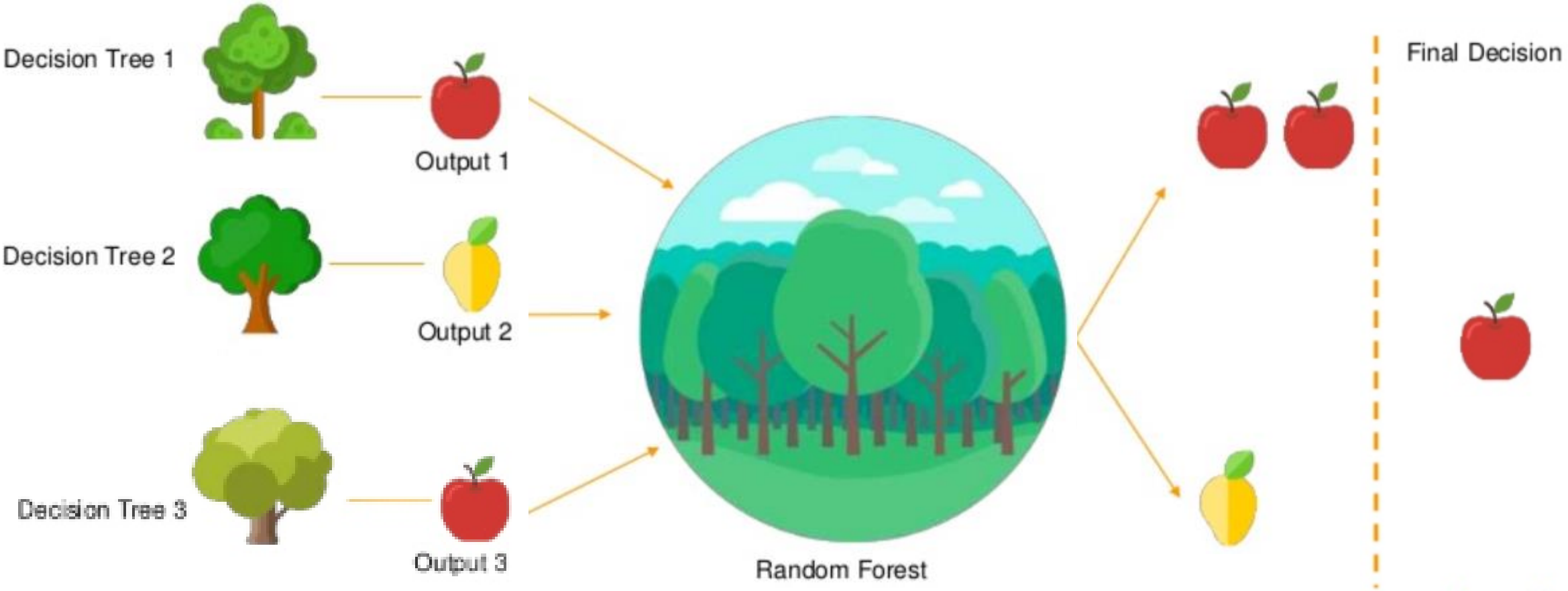
Random Forest Classifier:

- It is a method that operates by constructing multiple decision trees during training phase.
- It then aggregates votes from different decision trees to decide the final class of the test object.
- It uses randomness at 2 levels:

    i) In data selection

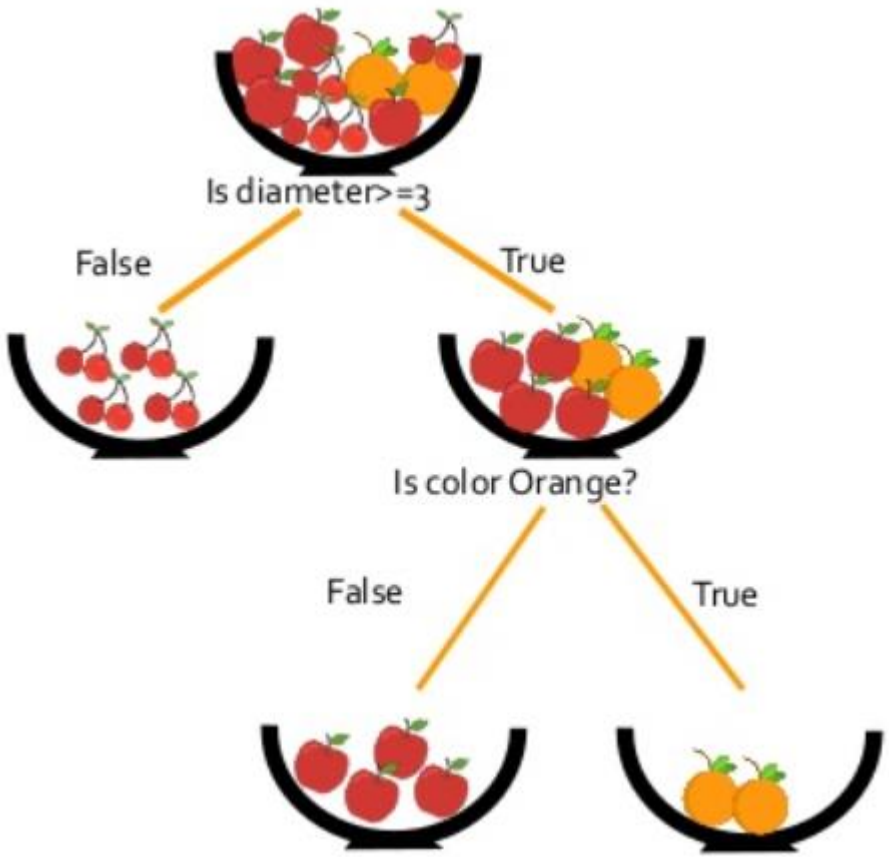
    ii) In attribute selection

Simple Example



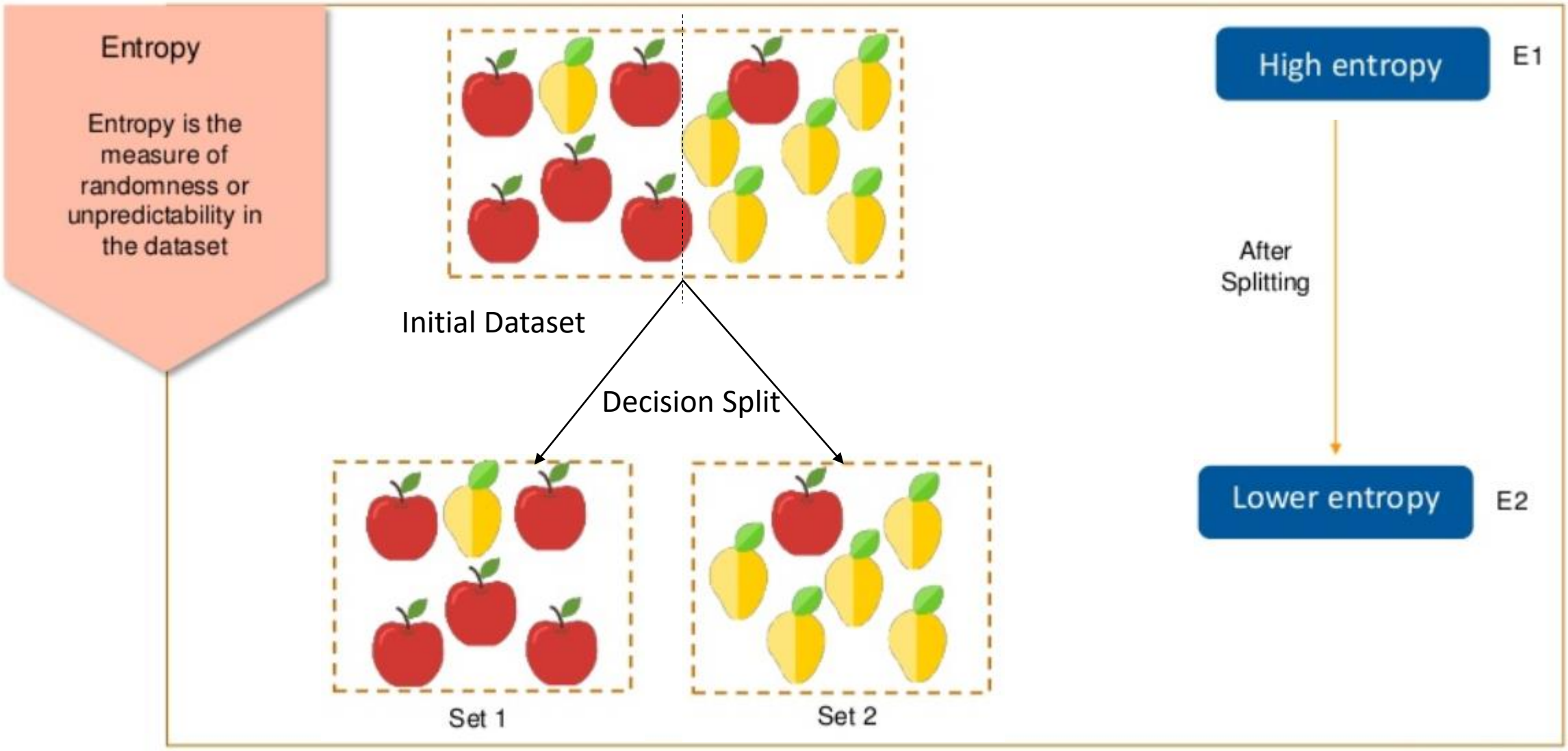
Now let us try to understand Decision tree vs Random Forest

Decision Tree

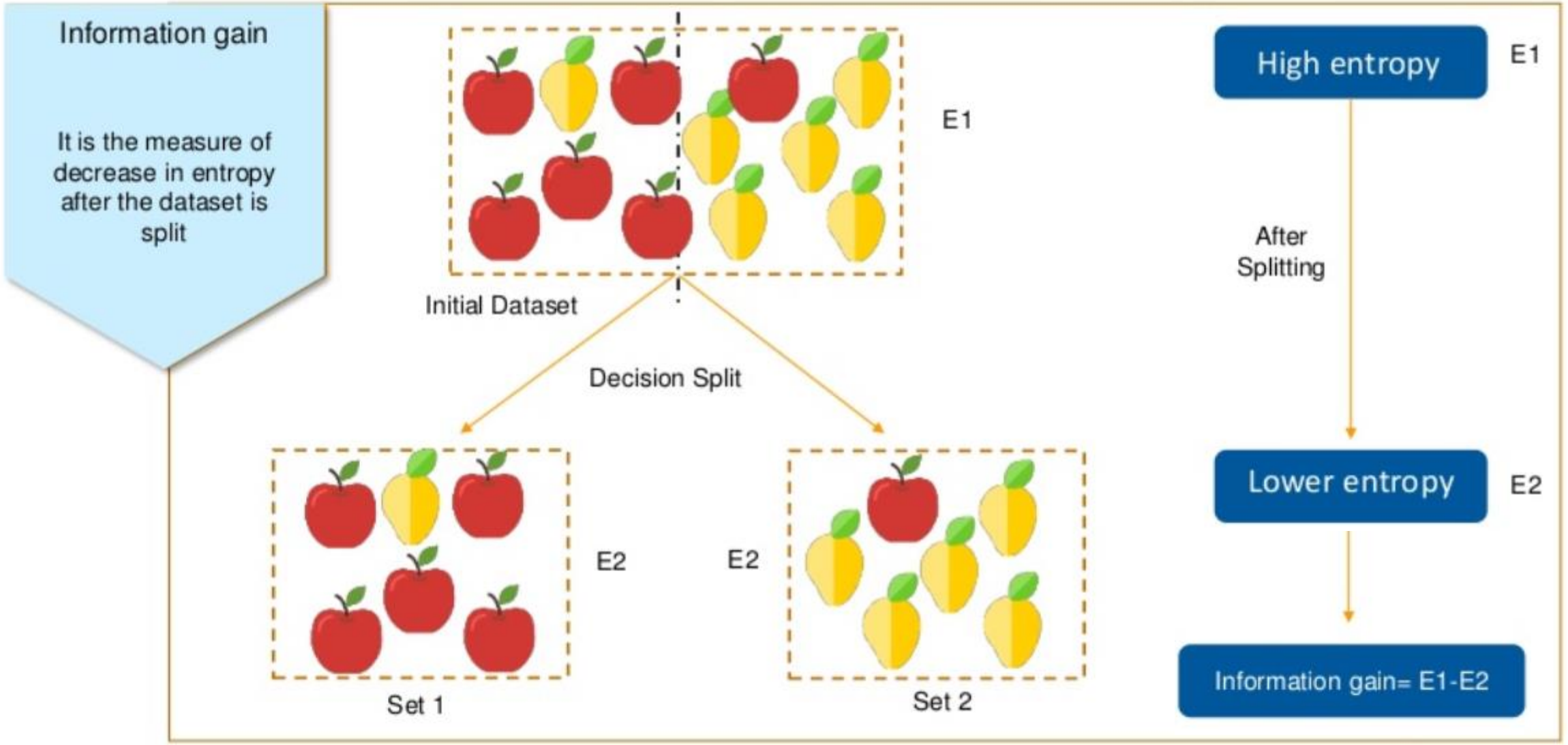


- Decision Tree is a tree shaped diagram used to determine course of action.
- Each branch represents a possible occurrence or relation.
- It is implemented using two methods:
  - Cart – Gini Index
  - Information Gain

Entropy


Entropy
Entropy is the measure of randomness or unpredictability in the dataset
Initial Dataset
Decision Split
Set 1
Set 2
High entropy
E1
After Splitting
Lower entropy
E2

Information Gain



How Decision Tree Works

Let us consider classifying different types of fruits in the bowl considering different features.

How Decision Tree Works

Training Dataset

Color	Diameter	Label
Red	3	Apple
Yellow	3	Lemon
Purple	1	Grapes
Red	3	Apple
Yellow	3	Lemon
Purple	1	Grapes

Splitting:

- From the data, we need to frame different conditions.
- Split the data for each condition and check which condition has highest information gain.
- The condition that has the **highest gain** is used to make the **first split.**
- The process is then repeated for left and right nodes in the same fashion.

How Decision Tree Works

Conditions

Color== purple?

Diameter=3

Color==Yellow?

Color== Red?

Diameter=1

Suppose if this condition has the highest gain

Training Dataset

Color	Diameter	Label
Red	3	Apple
Yellow	3	Lemon
purple	1	Grapes
Red	3	Apple
Yellow	3	Lemon
purple	1	Grapes

How Decision Tree Works

Is diameter >=3?

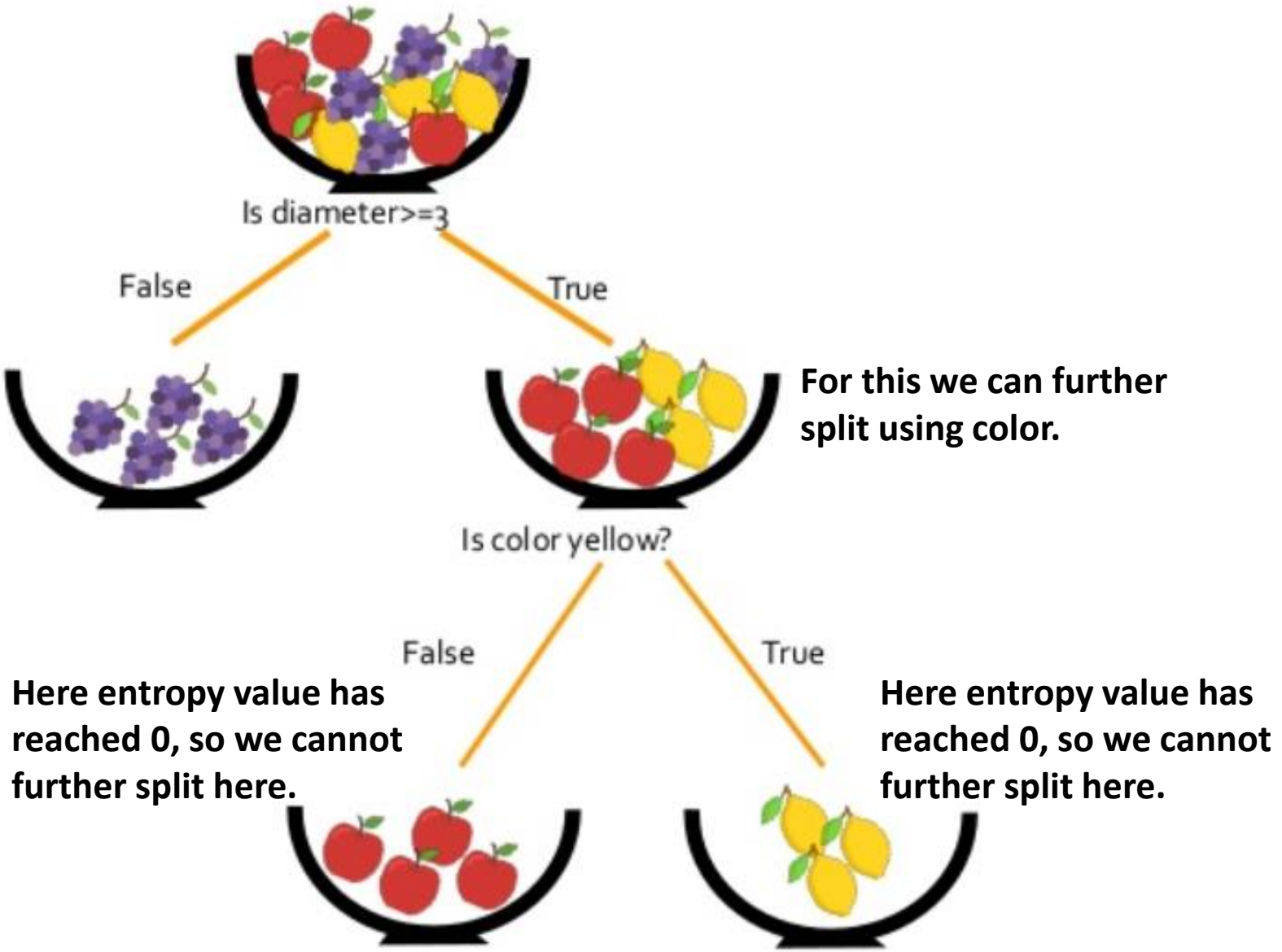
False

True

Here entropy value has reached 0, so we cannot further split here.

Here we need to further split.

How Decision Tree Works



For this we can further split using color.

Here entropy value has reached 0, so we cannot further split here.

Here entropy value has reached 0, so we cannot further split here.

How Random Forest Works

What are the problems with Decision tree algorithms?

Imagine in your dataset some green lemons and not ripe bananas and tomatoes. Now we have many green fruits to classify!

"Color" is not the attribute with the most information gain anymore, so it will not be the splitting attribute in the root node, the structure of the tree is going to change drastically.

Decision trees are very sensitive to changes in training examples.

If there is a non-informative features that happens to provide good information gain, coincidentally or because it is correlated with an informative feature, decision trees will wrongly use it as a splitting attribute. Decision trees are very sensitive to changes in the features.

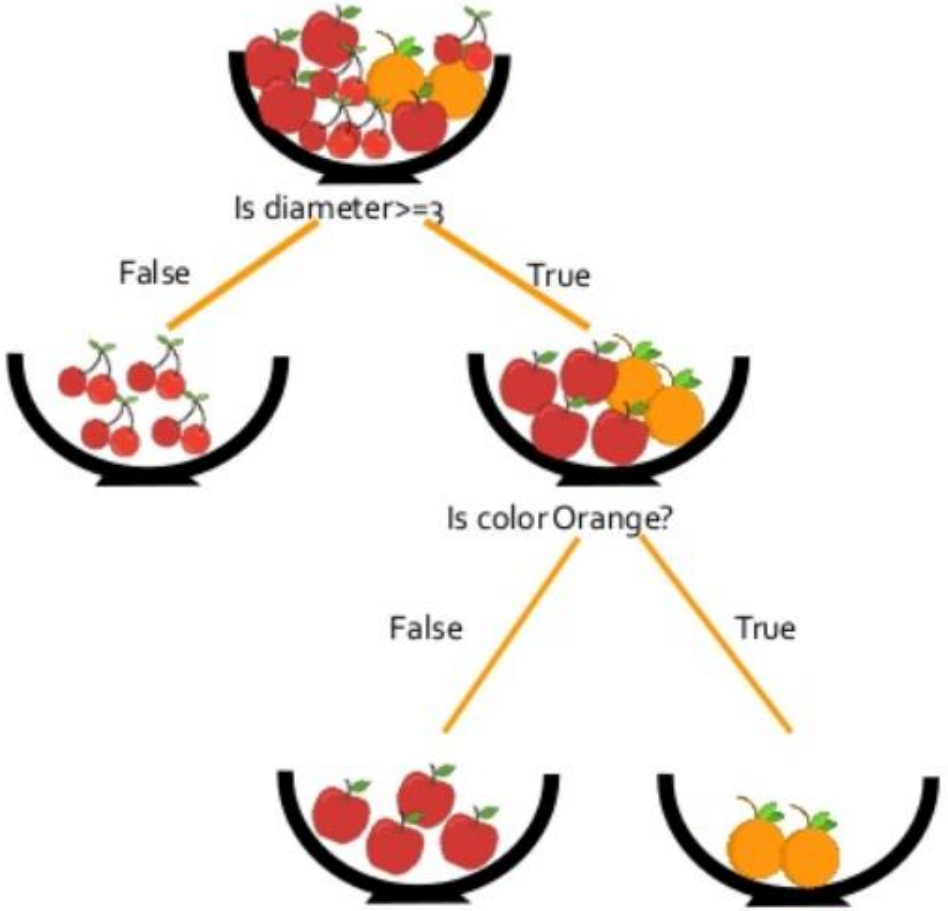
In a nutshell, decision trees are very sensitive to changes in the data and won't generalize well.

They are considered as weak learners

How Random Forest Works

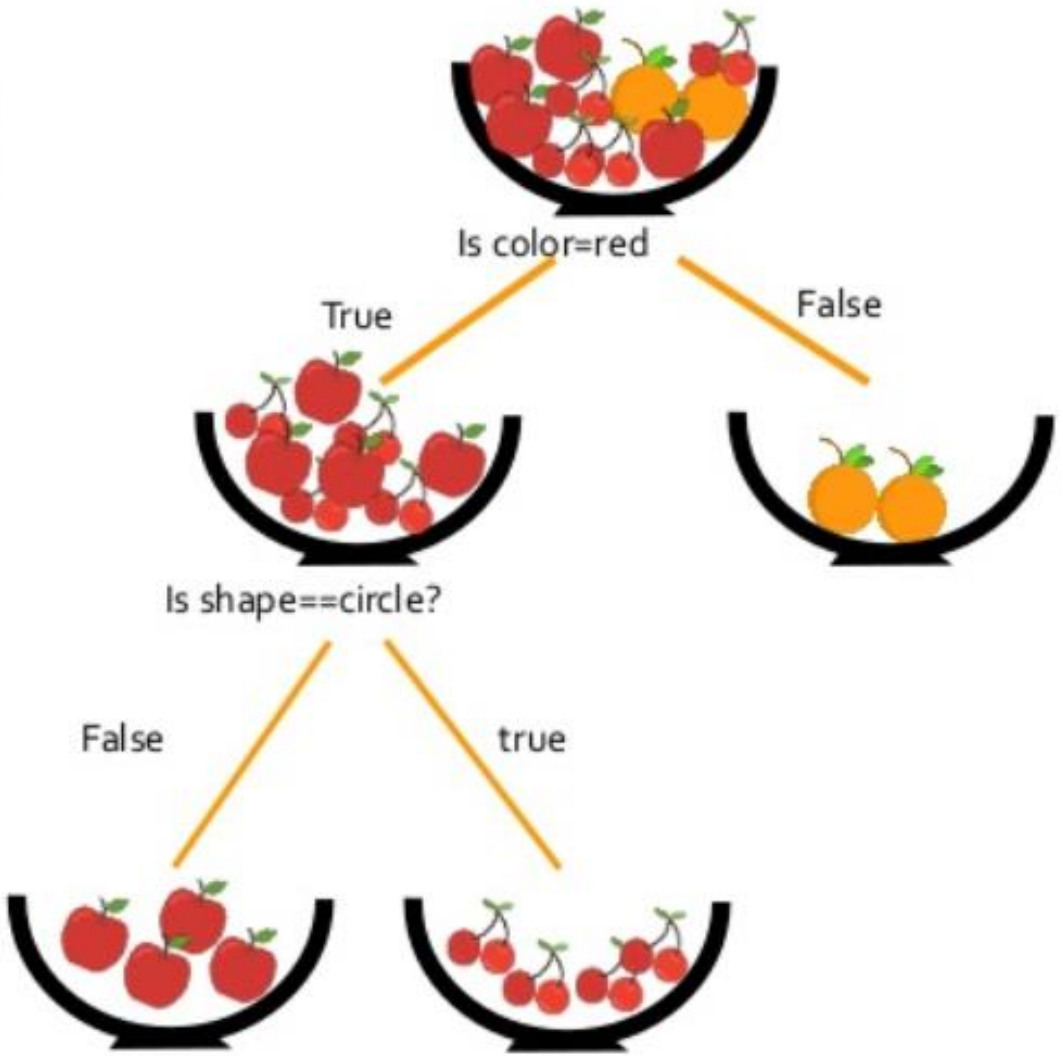
Now lets try to understand how the same problem can be solved using Random Forest.

Let this be Tree 1



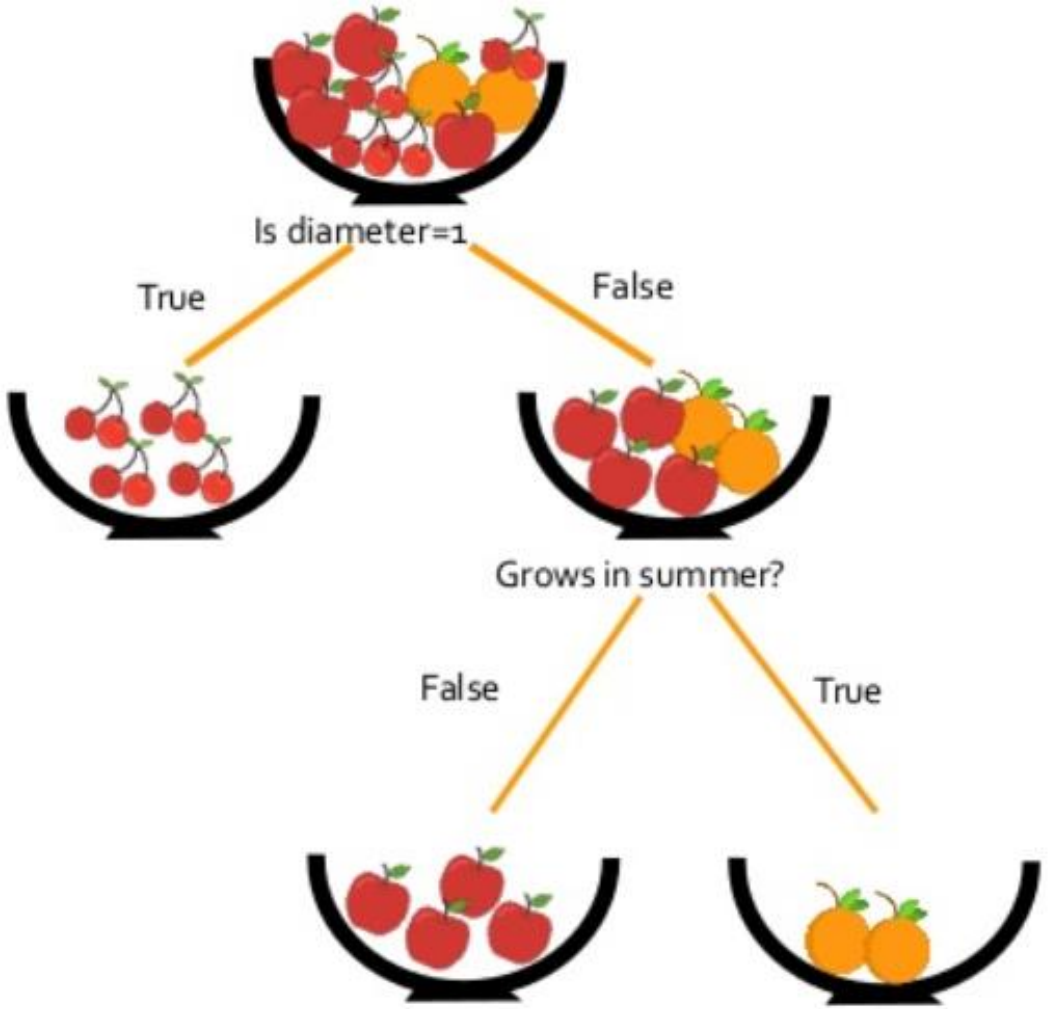
How Random Forest Works

Let this be Tree 2

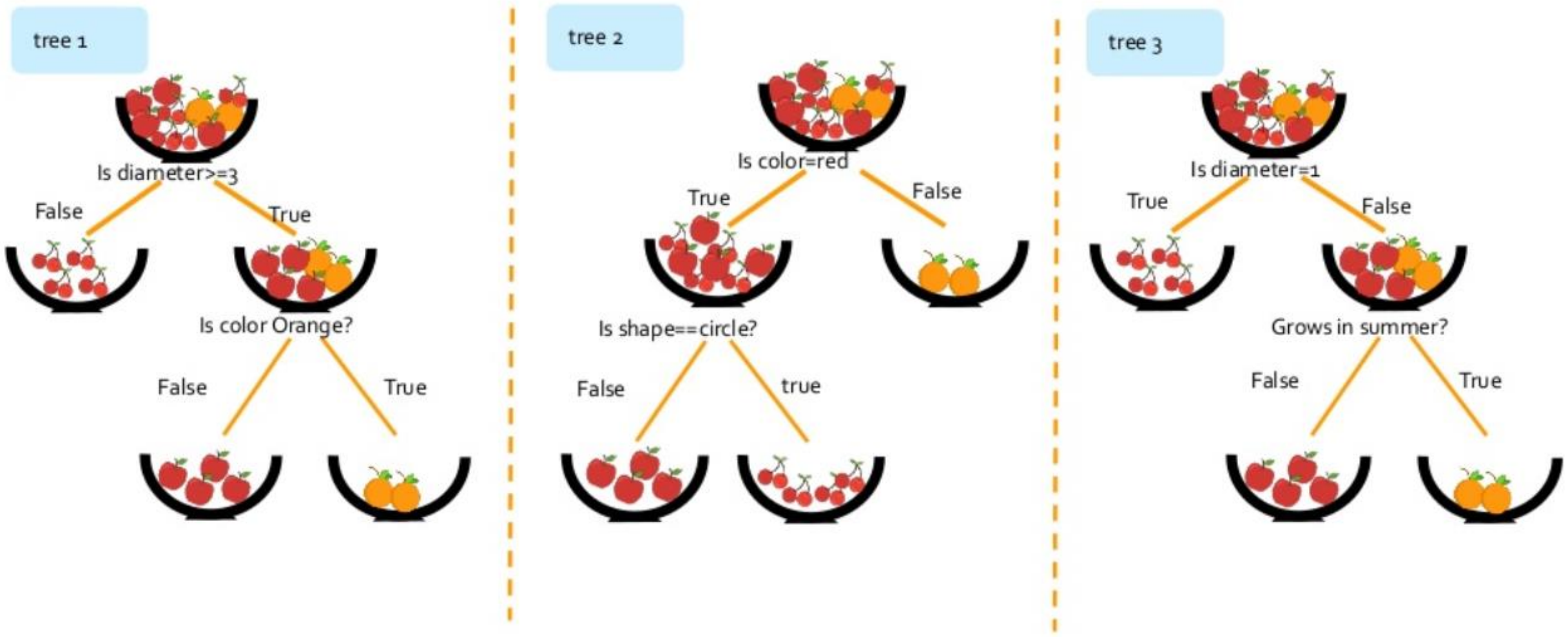


How Random Forest Works

Let this be Tree 3



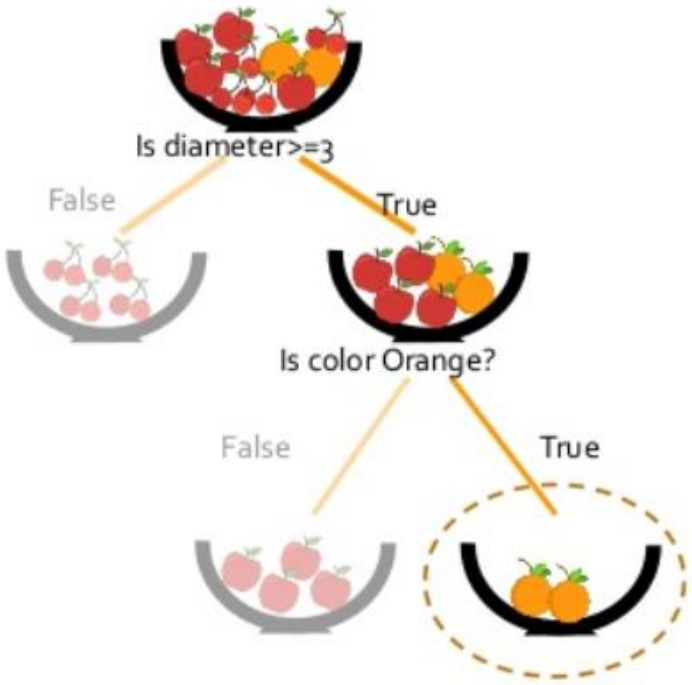
How Random Forest Works



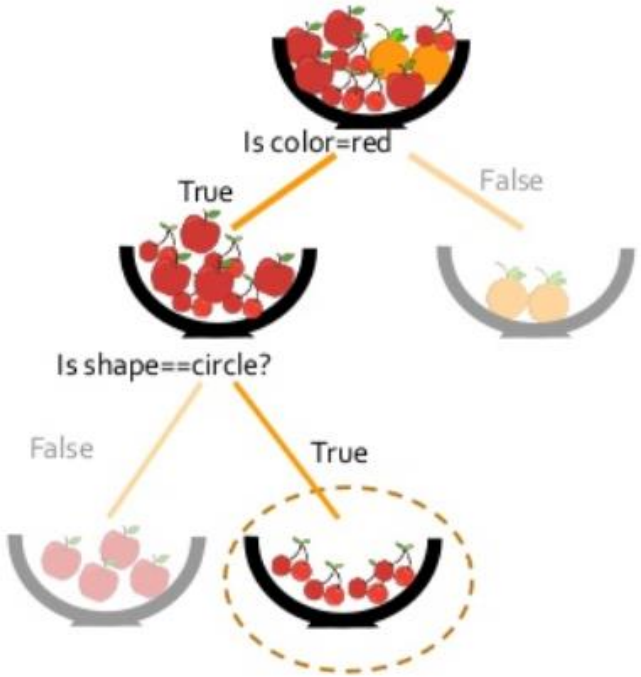
How Random Forest Works

Now let us try to identify what this fruit is which has no color.

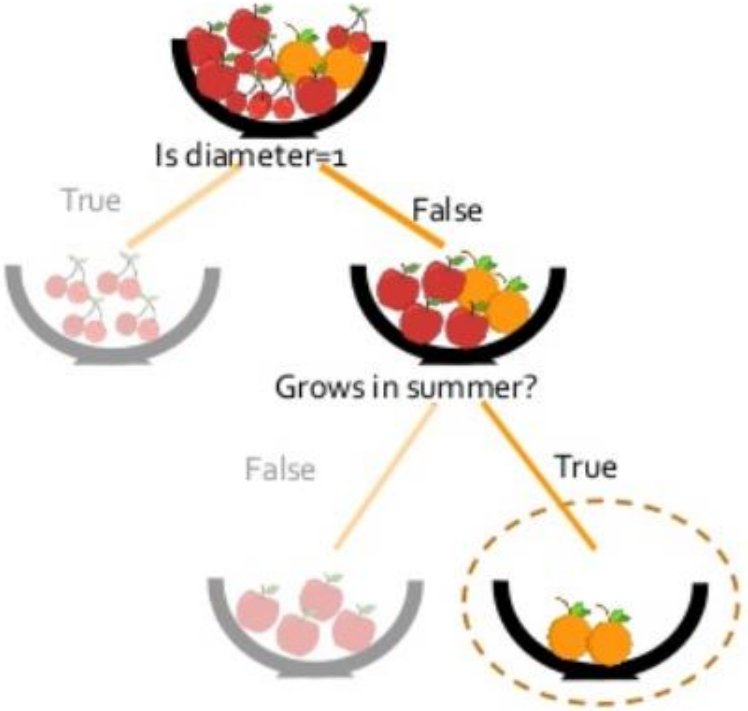
Tree 1



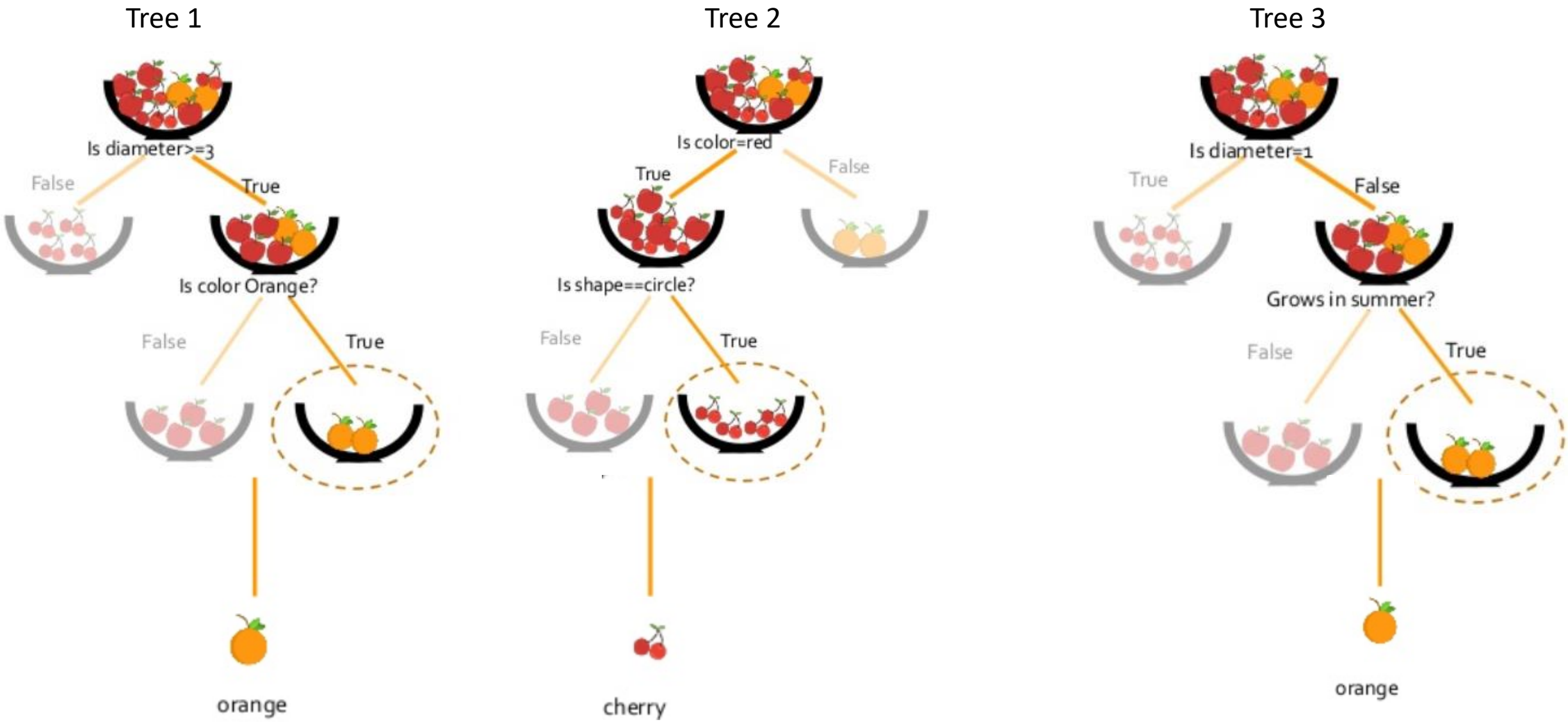
Tree 2



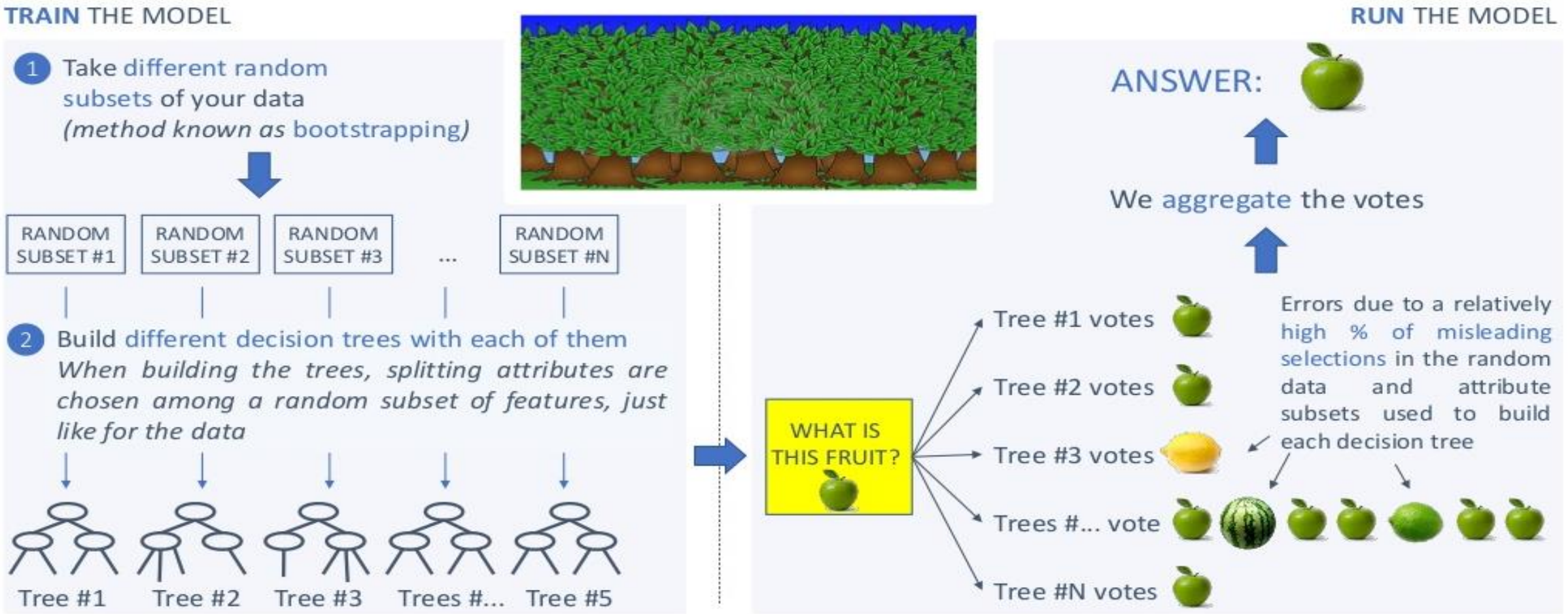
Tree 3



How Random Forest Works



In a nutshell



Bayesian Classification

What is it ?
Statistical method for classification.
Supervised Learning Method.
Assumes an underlying probabilistic model, the Bayes theorem.
Can solve problems involving both categorical and continuous valued attributes.

Naïve Bayes

Naive Bayes classifier works on the principles of conditional probability as given by the Bayes theorem

Before we move ahead, let us go through some of the simple concepts in probability that we will be using

Let us consider the following example of tossing two coins

Here, the sample space is:
{HH,HT,TH,TT}

1. P(Getting two heads) = ¼
2. P(At least one tail) = ¾
3. P(Second coin being head given first coin is tail) = ½
4. P(Getting two heads given first coin is a head) = 1/2

Naïve Bayes

Bayes Theorem gives the conditional probability of an event A given another event B has occurred

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Where,
P(A/B) = Conditional Probability of A given B
P(B/A) = Conditional Probability of A given B
P(A) = Probability of event A
P(B) = Probability of event B

Naïve Bayes

Let us apply Bayes theorem to our example

Here, the sample space is:

S={HH,HT,TH,TT}

P(getting two heads) = 1/4

P(At least one tail) = 3/4

P(Second coin being head given first coin is tail) = 1/2

P(Getting two heads given first coin is a head) = 1/2

These two use sample probabilities calculated directly from the sample

Naïve Bayes

Let us apply Bayes theorem to our example

Here, the sample space is:

S={HH,HT,TH,TT}

P(getting two heads) = 1/4

P(At least one tail) = 3/4

P(Second coin being head given first coin is tail) = 1/2

P(Getting two heads given first coin is a head) = 1/2

This uses conditional probability.
Let us understand this in detail

Naïve Bayes

In this sample space, let A be the event that second coin is head and B be the event that first coin is tail

Here, the sample space is:

S={HH,HT,TH,TT}

P(Second coin being head given first coin is tail)

=P(A/B)

=[P(B/A)*P(A)]/P(B)

=[P(First coin being tail given second coin is head)*P(Second coin being head)]/P(First coin being tail)

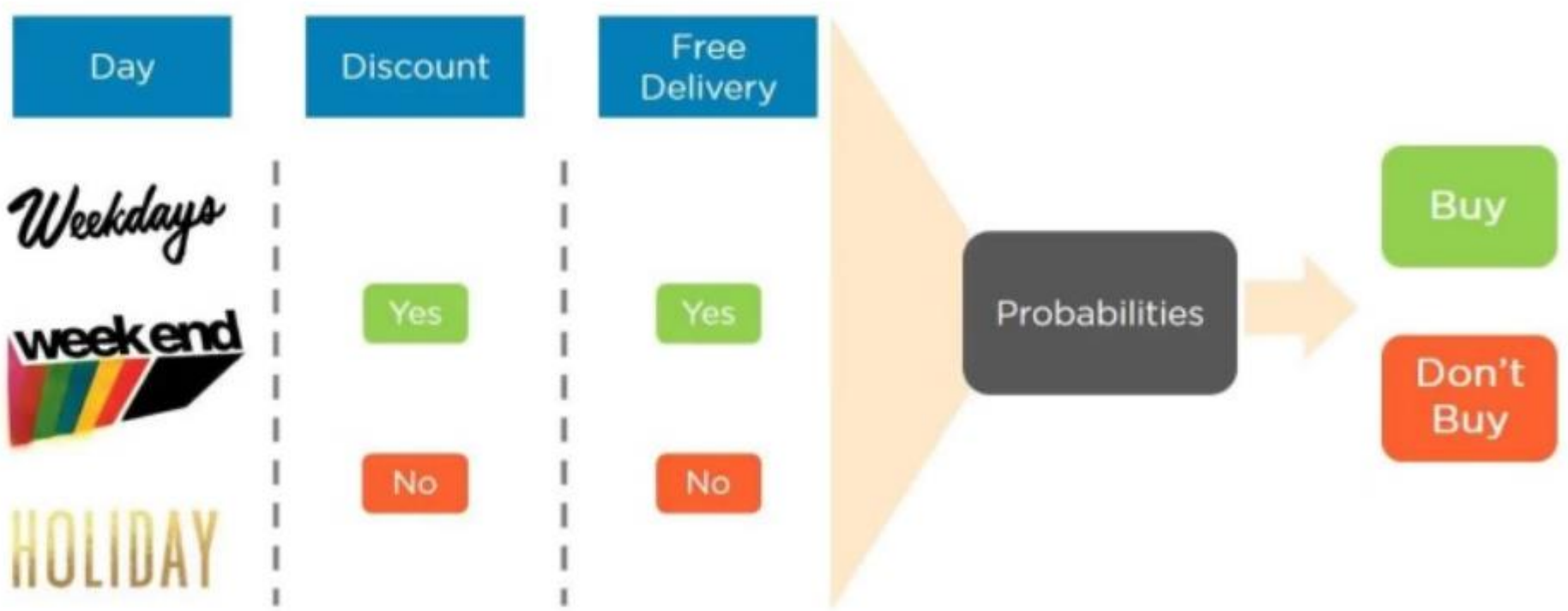
=[(1/2)*(1/2)]/(1/2)

=1/2 = 0.5

Bayes Theorem basically calculates the conditional probability of the occurrence of an event based on prior knowledge of conditions that might be related to the event

Example

To Predict Whether a person will purchase a product on specific combination of Day, Discount and free delivery using Naïve Bayes classifier



Example

We have a small space dataset of 30 rows for our demo

	A	B	C	D
1	Day	Discount	Free Delivery	Purchase
2	Weekday	Yes	Yes	Yes
3	Weekday	Yes	Yes	Yes
4	Weekday	No	No	No
5	Holiday	Yes	Yes	Yes
6	Weekend	Yes	Yes	Yes
7	Holiday	No	No	No
8	Weekend	Yes	No	Yes
9	Weekday	Yes	Yes	Yes
10	Weekend	Yes	Yes	Yes
11	Holiday	Yes	Yes	Yes
12	Holiday	No	Yes	Yes
13	Holiday	No	No	No
14	Weekend	Yes	Yes	Yes
15	Holiday	Yes	Yes	Yes

Naive_Bayes_Dataset

Example

Based on this dataset containing three input types of day, Discount and Free delivery, we will populate frequency tables for each attributes

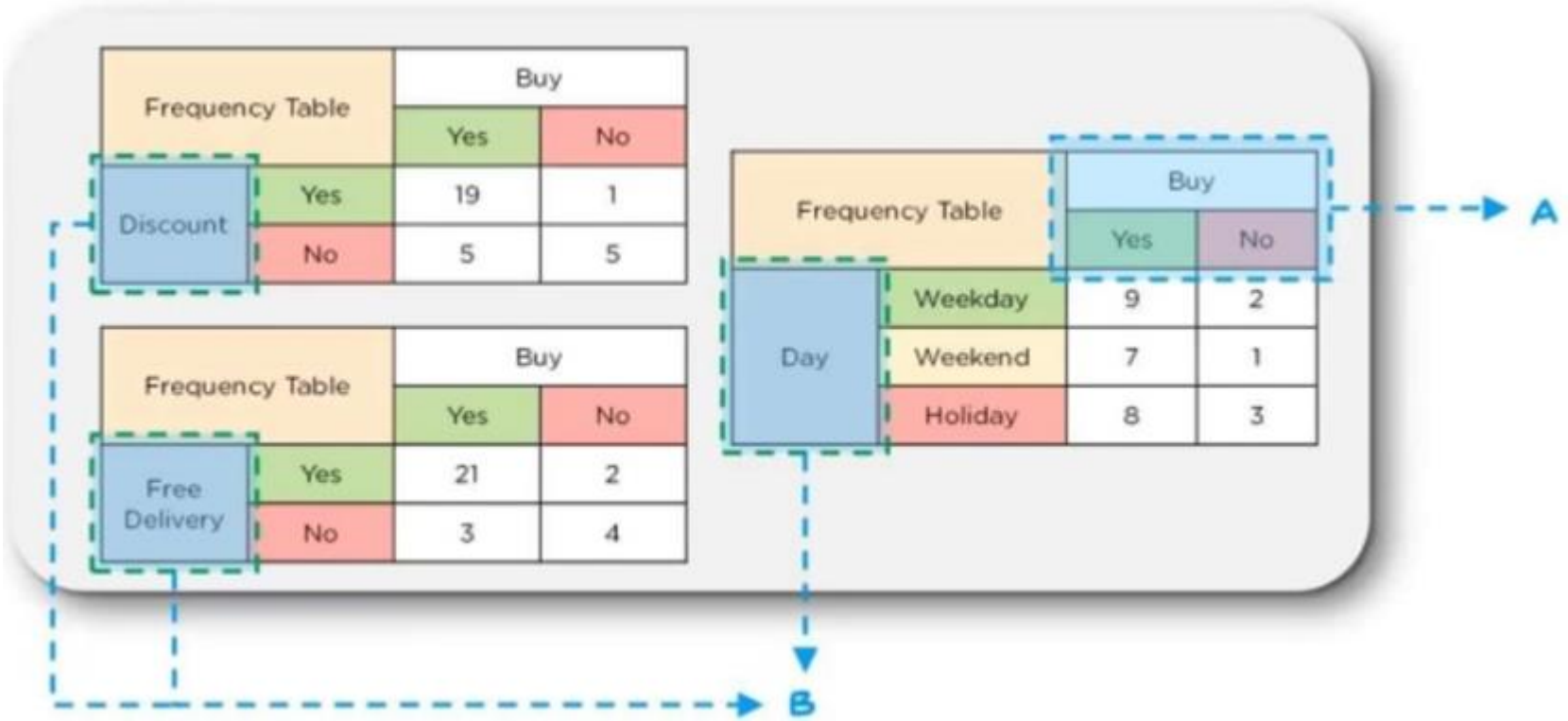
Frequency Table		Buy	
		Yes	No
Discount	Yes	19	1
	No	5	5

Frequency Table		Buy	
		Yes	No
Free Delivery	Yes	21	2
	No	3	4

Frequency Table		Buy	
		Yes	No
Day	Weekday	9	2
	Weekend	7	1
	Holiday	8	3

Example

Based on this dataset containing three input types of day, Discount and Free delivery, we will populate frequency tables for each attributes



For our Bayes theorem, let the event buy be A and the independent variables, discount, free delivery and day be B

Example

Now let us calculate the Likelihood table for one of the variable, Day which includes Weekday, Weekend and Holiday

Frequency Table		Buy		
		Yes	No	
Day	Weekday	9	2	11
	Weekend	7	1	8
	Holiday	8	3	11
		24	6	30

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Weekend	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	

P(B) = P(Weekday)
=11/30 = 0.37

P(A) = P(No Buy)
=6/30 = 0.2

P(B|A)
=P(Weekday | No Buy)
=2/6 = 0.33

Example

Based on this likelihood table, we will calculate conditional probabilities as below

Frequency Table		Buy		
		Yes	No	
Day	Weekday	9	2	11
	Weekend	7	1	8
	Holiday	8	3	11
		24	6	30

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Weekend	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	

P(B) = P(Weekday) = 11/30 = 0.367
P(A) = P(No Buy) = 6/30 = 0.2
P(B|A) = P(Weekday | No Buy) = 2/6 = 0.33

P(A|B) = P(No Buy) | Weekday)
= P(Weekday | No Buy) * P(No Buy) / P(Weekday)
= (0.33 * 0.2) / 0.367 = 0.179

Example

Based on this likelihood table, we will calculate conditional probabilities as below

Frequency Table		Buy		
		Yes	No	
Day	Weekday	9	2	11
	Weekend	7	1	8
	Holiday	8	3	11
		24	6	30

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Weekend	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	

P(B) = P(Weekday) = 11/30 = 0.367
P(A) = P(Buy) = 24/30 = 0.8
P(B|A) = P(Weekday | Buy) = 2/6 = 0.375
If A equals Buy, then
P(A|B) = P(Buy | Weekday)
=P(Weekday | Buy)*P(Buy)/P(Weekday)
=(0.375*0.8)/0.367
=0.817

As the Probability(Buy | Weekday) is more that Probability(No Buy | Weekday), we can conclude that a customer will most likely buy the product on a Weekday

Example

Similarly we can find the likelihood of occurrence of an event involving all three variables

We have the frequency tables of all the three independent variables. We will now construct likelihood tables for all the three

Frequency Table		Buy	
		Yes	No
Day	Weekday	3	7
	Weekend	8	2
	Holiday	9	1

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Weekend	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	

Frequency Table		Buy	
		Yes	No
Discount	Yes	19	1
	No	5	5

Frequency Table		Buy		
		Yes	No	
Discount	Yes	19/24	1/6	20/30
	No	5/24	5/6	10/30
		24/30	6/30	

Frequency Table		Buy	
		Yes	No
Free Delivery	Yes	21	2
	No	3	4

Frequency Table		Buy		
		Yes	No	
Free Delivery	Yes	21/24	2/6	23/30
	No	3/24	4/6	7/30
		24/30	6/30	

Example

Let us use these 3 likelihood tables to calculate whether a customer will purchase a product on a specific combination of day, Discount, and free delivery or not

Here let us take a combination of these factors:

Day = Holiday

Discount = yes

Free Delivery = yes

Example

Likelihood Table

Frequency Table		Buy	
		Yes	No
Day	Weekday	3	7
	Weekend	8	2
	Holiday	9	1

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Weekend	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	

Frequency Table		Buy	
		Yes	No
Discount	Yes	19	1
	No	5	5

Frequency Table		Buy		
		Yes	No	
Discount	Yes	19/24	1/6	20/30
	No	5/24	5/6	10/30
		24/30	6/30	

Frequency Table		Buy	
		Yes	No
Free Delivery	Yes	21	2
	No	3	4

Frequency Table		Buy		
		Yes	No	
Free Delivery	Yes	21/24	2/6	23/30
	No	3/24	4/6	7/30
		24/30	6/30	

Calculating Conditional Probability of purchase on the following combinations of day, discount and free delivery:

Where B equals:

- Day = Holiday
- Discount = Yes
- Free Delivery = Yes

Let A = No Buy

$P(A|B) = P(\text{No Buy} \mid \text{Discount = Yes, Free Delivery = Yes, Day = Holiday})$

$$= \frac{P(\text{Discount = Yes} \mid \text{No Buy}) * P(\text{Free Delivery = Yes} \mid \text{No Buy}) * P(\text{Day = Holiday} \mid \text{No Buy}) * P(\text{No Buy})}{P(\text{Discount = Yes}) * P(\text{Free Delivery = Yes}) * P(\text{Day = Holiday})}$$

$$= \frac{(1/6)*(2/6)*(3/6)*(6/30)}{(20/30)*(23/30)*(11/30)}$$

$$= 0.178$$

Example

Likelihood Table

Frequency Table		Buy	
		Yes	No
Day	Weekday	3	7
	Weekend	8	2
	Holiday	9	1

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Weekend	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	

Frequency Table		Buy	
		Yes	No
Discount	Yes	19	1
	No	5	5

Frequency Table		Buy		
		Yes	No	
Discount	Yes	19/24	1/6	20/30
	No	5/24	5/6	10/30
		24/30	6/30	

Frequency Table		Buy	
		Yes	No
Free Delivery	Yes	21	2
	No	3	4

Frequency Table		Buy		
		Yes	No	
Free Delivery	Yes	21/24	2/6	23/30
	No	3/24	4/6	7/30
		24/30	6/30	

Calculating Conditional Probability of purchase on the following combinations of day, discount and free delivery:

Where B equals:

- Day = Holiday
- Discount = Yes
- Free Delivery = Yes

Let A = Buy

$P(A|B) = P(\text{No Buy} \mid \text{Discount = Yes, Free Delivery = Yes, Day = Holiday})$

$$= \frac{P(\text{Discount = Yes} \mid \text{Buy}) * P(\text{Free Delivery = Yes} \mid \text{Buy}) * P(\text{Day = Holiday} \mid \text{Buy}) * P(\text{Buy})}{P(\text{Discount = Yes}) * P(\text{Free Delivery = Yes}) * P(\text{Day = Holiday})}$$

$$= \frac{(19/24)*(21/24)*(8/24)*(24/30)}{(20/30)*(23/30)*(11/30)}$$

$$= 0.986$$

Example

Probability of purchase = 0.986
Probability of purchase = 0.178

Finally, We have conditional probabilities of purchase on this day

Let us now normalize these probabilities to get the likelihood of the events.

Sum of probabilities
=0.986+0.178 = 1.164
Likelihood of purchase
=0.986/1.164 = 84.71%
Likelihood of no purchase
=0.178/1.164 = 15.29%

Probability of purchase = 0.986
Probability of no purchase = 0.178

As 84.71% is greater than 15.29% we can conclude that an average customer will buy on a holiday with discount and free delivery

Example

Sum of probabilities
=0.986+0.178 = 1.164
Likelihood of purchase
=0.986/1.164 = 84.71%
Likelihood of no purchase
=0.178/1.164 = 15.29%

Probability of purchase = 0.986
Probability of no purchase = 0.178

As 84.71% is greater than 15.29% we can conclude that an average customer will buy on a holiday with discount and free delivery

Probabilistic classification

- MAP classification rule
  - **MAP**: **M**aximum **A** **P**osterior
  - Assign x to c^* if

$$P(C = c^* \mid \mathbf{X} = \mathbf{x}) > P(C = c \mid \mathbf{X} = \mathbf{x}) \quad c \neq c^*, \; c = c_1, \cdots, c_L$$

- Generative classification with the MAP rule
  - Apply Bayesian rule to convert them into posterior probabilities

$$\begin{aligned} P(C = c_i \mid \mathbf{X} = \mathbf{x}) &= \frac{P(\mathbf{X} = \mathbf{x} \mid C = c_i) P(C = c_i)}{P(\mathbf{X} = \mathbf{x})} \\ &\propto P(\mathbf{X} = \mathbf{x} \mid C = c_i) P(C = c_i) \\ &\text{for } \; i = 1, 2, \cdots, L \end{aligned}$$

  - Then apply the MAP rule

Probabilistic classification

- Bayes classification

$$P(C \mid \mathbf{X}) \propto P(\mathbf{X} \mid C)P(C) = P(X_1, \cdots, X_n \mid C)P(C)$$

Difficulty: learning the joint probability $P(X_1, \cdots, X_n \mid C)$

- Naïve Bayes classification
  - Assumption that all input attributes are conditionally independent!

$$\begin{aligned} P(X_1, X_2, \cdots, X_n \mid C) &= P(X_1 \mid X_2, \cdots, X_n; C)P(X_2, \cdots, X_n \mid C) \\ &= P(X_1 \mid C)P(X_2, \cdots, X_n \mid C) \\ &= P(X_1 \mid C)P(X_2 \mid C) \cdots P(X_n \mid C) \end{aligned}$$

  - MAP classification rule: for $\mathbf{x} = (x_1, x_2, \cdots, x_n)$

$$[P(x_1 \mid c^*) \cdots P(x_n \mid c^*)]P(c^*) > [P(x_1 \mid c) \cdots P(x_n \mid c)]P(c), \quad c \neq c^*, c = c_1, \cdots, c_L$$

Probabilistic classification

- Naïve Bayes Algorithm
- Learning Phase: Given a training set **S**,

For each target value of c_i $(c_i = c_1, \cdots, c_L)$

$\hat{P}(C = c_i) \leftarrow$ estimate $P(C = c_i)$ with examples in **S**;

For every attribute value x_{jk} of each attribute X_j $(j = 1, \cdots, n; k = 1, \cdots, N_j)$

$\hat{P}(X_j = x_{jk} \mid C = c_i) \leftarrow$ estimate $P(X_j = x_{jk} \mid C = c_i)$ with examples in **S**;

Output: conditional probability tables; for $X_j, N_j \times L$ elements

- Test Phase: Given an unknown instance $\mathbf{X}' = (a_1', \cdots, a_n')$,

Look up tables to assign the label c^* to **X'** if

$$[\hat{P}(a_1' \mid c^*) \cdots \hat{P}(a_n' \mid c^*)]\hat{P}(c^*) > [\hat{P}(a_1' \mid c) \cdots \hat{P}(a_n' \mid c)]\hat{P}(c), \quad c \neq c^*, c = c_1, \cdots, c_L$$

Example

- Example: Play Tennis

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Example

- Learning Phase

Outlook	Play=*Yes*	Play=*No*
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Temperature	Play=*Yes*	Play=*No*
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

Humidity	Play=*Yes*	Play=No
High	3/9	4/5
Normal	6/9	1/5

Wind	Play=*Yes*	Play=*No*
Strong	3/9	3/5
Weak	6/9	2/5

$P(\text{Play}=Yes) = 9/14 \quad P(\text{Play}=No) = 5/14$

Example

- Test Phase
  - Given a new instance,

    $\mathbf{x}'$=(Outlook=*Sunny,* Temperature=*Cool,* Humidity=*High,* Wind=*Strong*)
  - Look up tables

P(Outlook=*Sunny*|Play=*Yes*) = 2/9
P(Temperature=*Cool*|Play=*Yes*) = 3/9
P(Huminity=*High*|Play=*Yes*) = 3/9
P(Wind=*Strong*|Play=*Yes*) = 3/9
P(Play=*Yes*) = 9/14

P(Outlook=S*unny*|Play=*No*) = 3/5
P(Temperature=*Cool*|Play==*No*) = 1/5
P(Huminity=*High*|Play=*No*) = 4/5
P(Wind=*Strong*|Play=*No*) = 3/5
P(Play=*No*) = 5/14

  - MAP rule

P(*Yes*|$\mathbf{x}'$): [P(*Sunny*|Yes)P(*Cool*|*Yes*)P(*High*|Yes)P(*Strong*|*Yes*)]P(Play=*Yes*) = 0.0053
P(*No*|$\mathbf{x}'$): [P(*Sunny*|No) P(*Cool*|No)P(*High*|*No*)P(*Strong*|*No*)]P(Play=*No*) = 0.0206

Given the fact P(*Yes*|$\mathbf{x}'$) < P(*No*|$\mathbf{x}'$), we label $\mathbf{x}'$ to be "*No*".

Continued value features

Algorithm: Continuous-valued Features

- Numberless values for a feature

- Conditional probability often modeled with the normal distribution

$$\hat{P}(X_j \mid C = c_i) = \frac{1}{\sqrt{2\pi}\sigma_{ji}} \exp\left(-\frac{(X_j - \mu_{ji})^2}{2\sigma_{ji}^2}\right)$$

μ_{ji} : mean (avearage) of feature values X_j of examples for which $C = c_i$

σ_{ji} : standard deviation of feature values X_j of examples for which $C = c_i$

- **Learning Phase:** for $\mathbf{X} = (X_1, \cdots, X_n)$, $C = c_1, \cdots, c_L$

  Output: $n \times L$ normal distributions and $P(C = c_i)$ $i = 1, \cdots, L$

- **Test Phase:** Given an unknown instance $\mathbf{X}' = (a_1', \cdots, a_n')$

  -Instead of looking-up tables, calculate conditional probabilities with all the normal distributions achieved in the learning phrase

  -Apply the MAP rule to make a decision

Example: Continuous-valued Features

-Temperature is naturally of continuous value.

Yes: 25.2, 19.3, 18.5, 21.7, 20.1, 24.3, 22.8, 23.1, 19.8

No: 27.3, 30.1, 17.4, 29.5, 15.1

-Estimate mean and variance for each class

$$\mu = \frac{1}{N}\sum_{n=1}^{N} x_n, \quad \sigma^2 = \frac{1}{N}\sum_{n=1}^{N}(x_n - \mu)^2$$

$$\mu_{Yes} = 21.64, \ \sigma_{Yes} = 2.35$$

$$\mu_{No} = 23.88, \ \sigma_{No} = 7.09$$

-Learning Phase: output two Gaussian models for **P(temp|C)**

$$\hat{P}(x \mid Yes) = \frac{1}{2.35\sqrt{2\pi}} \exp\left(-\frac{(x - 21.64)^2}{11.09}\right)$$

$$\hat{P}(x \mid No) = \frac{1}{7.09\sqrt{2\pi}} \exp\left(-\frac{(x - 23.88)^2}{50.25}\right)$$

Support Vector Machine

Introduction

Last week, my son and I visited a fruit shop

There he found a fruit which was similar to both

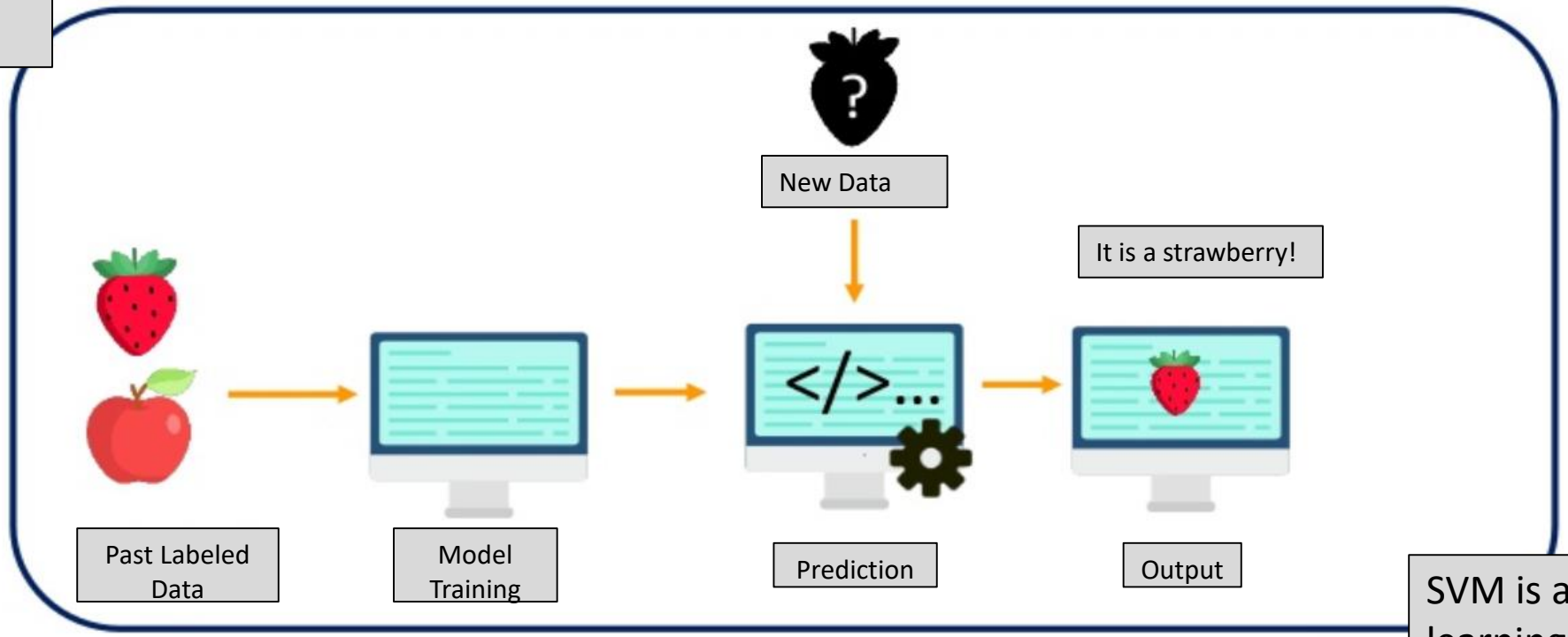
Dad, is that an apple or a strawberry?

After a couple of seconds, he could figure out that it was a strawberry

It is a strawberry!

Why Support Vector Machine?

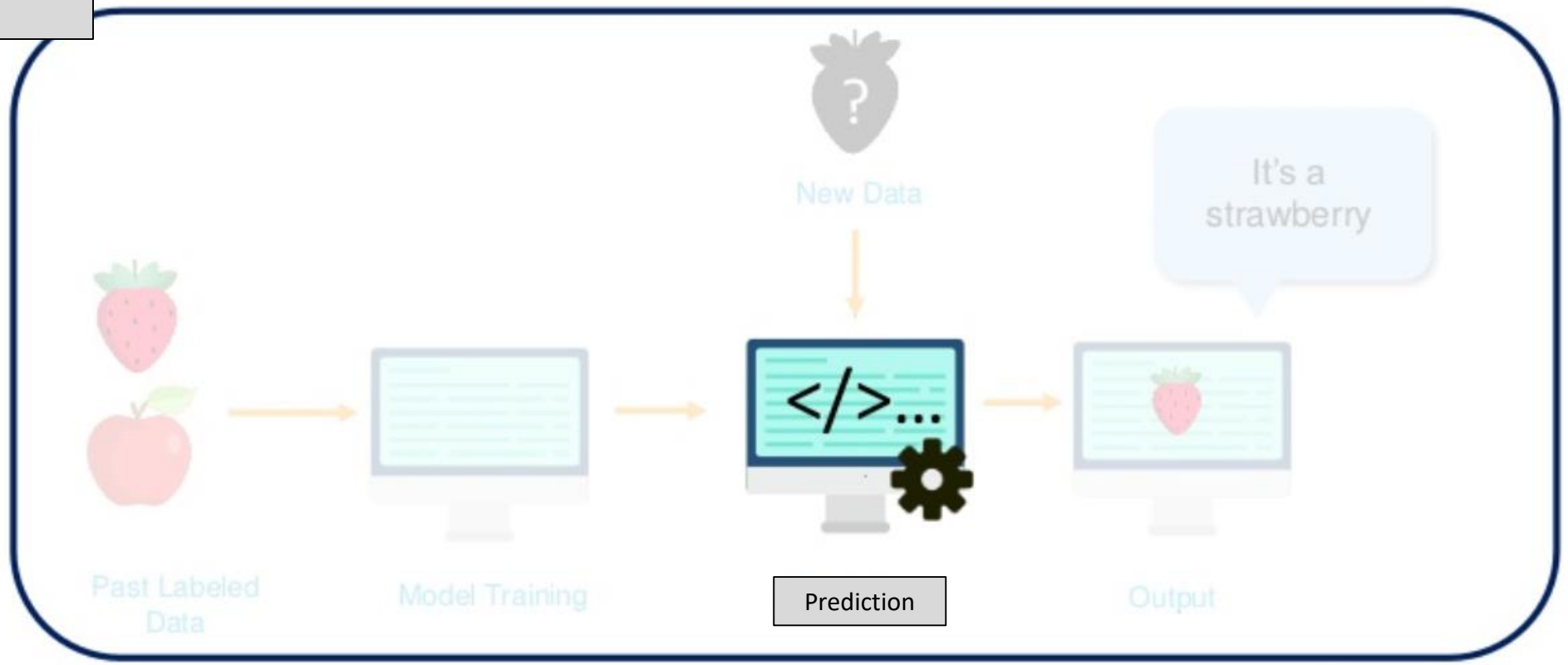
Why not build a model which can predict an unknown data?



SVM is a supervised learning method that looks at the data and sort it into one of the two categories

Why Support Vector Machine?

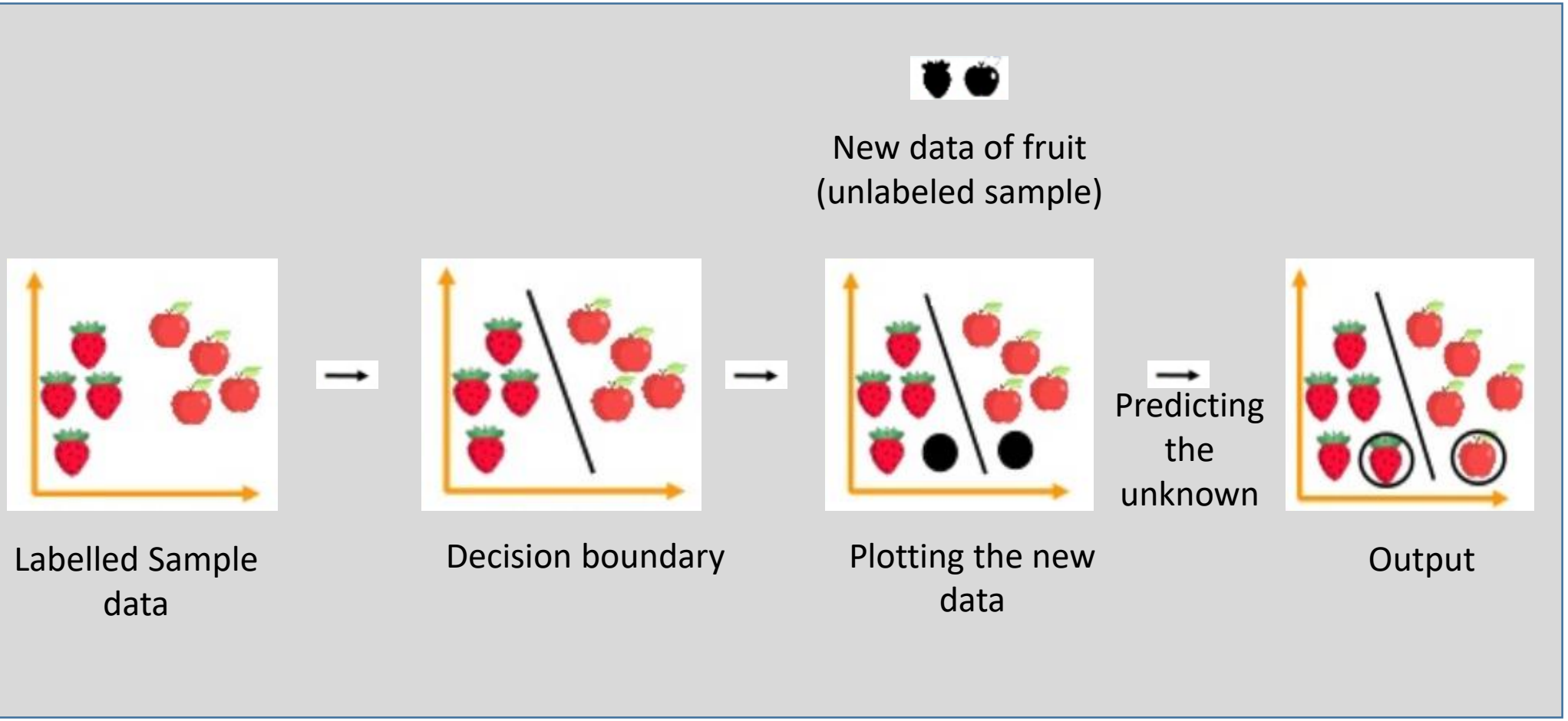
But how does prediction work?



Why Support Vector Machine?

SVM

Support Vector Machine



What is Support Vector Machine?

Example:

We are given a set people with different

- Height and
- Weight

Sample Dataset: Male

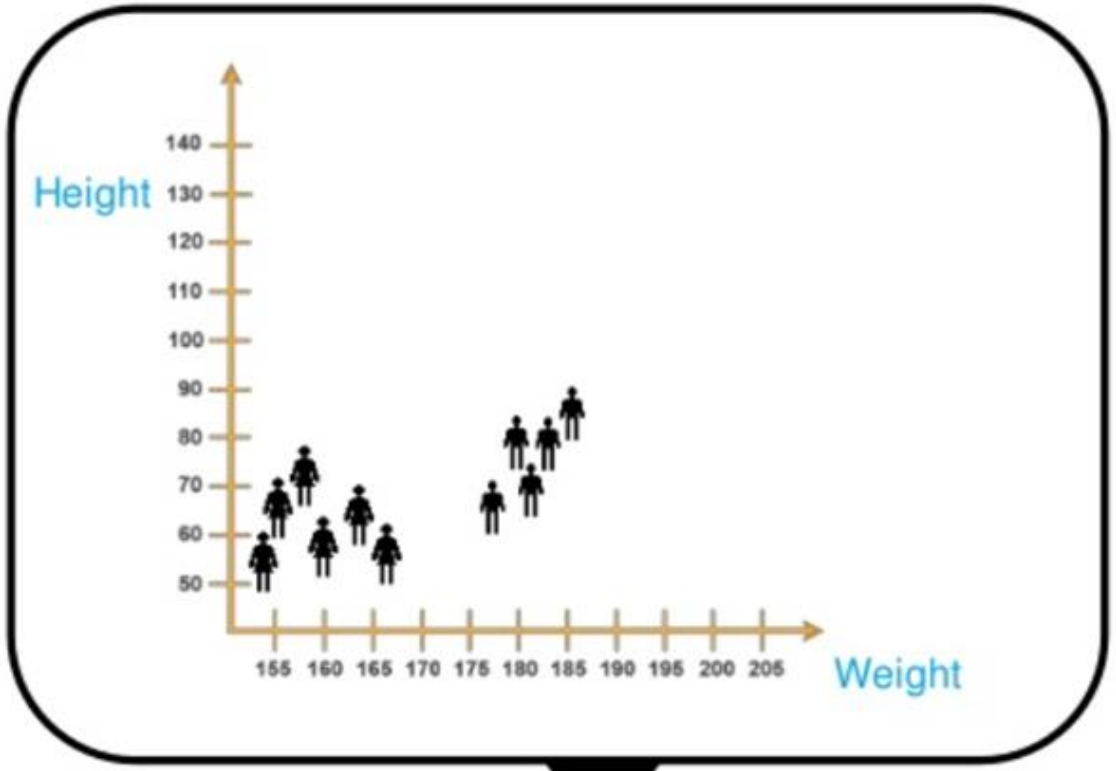
Height	Weight
179	90
180	80
183	80
187	85
182	72

Sample Dataset: Female

Height	Weight
174	65
174	88
175	75
180	65
185	80

What is Support Vector Machine?

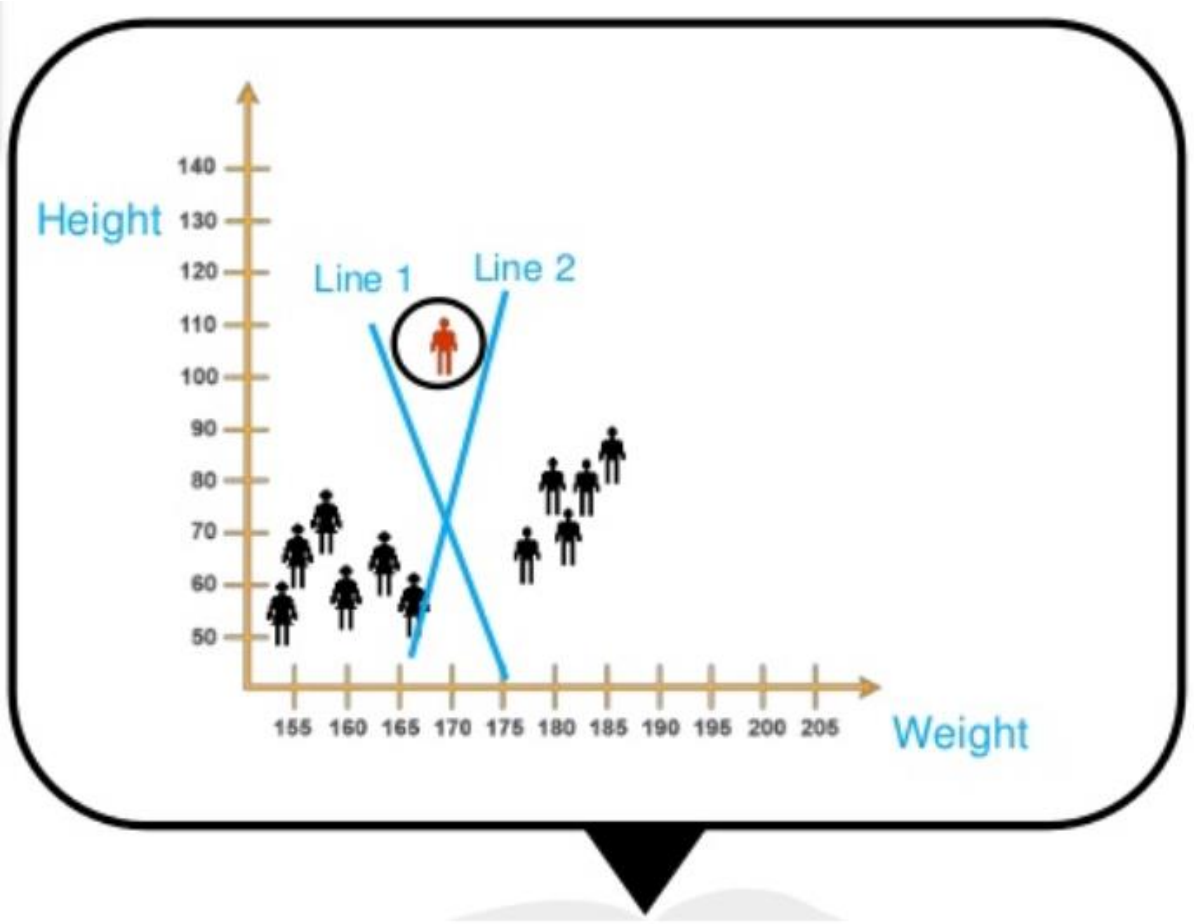
Lets add a new data point and figure out if it's a male or a female?



For this task, we need to split our data first

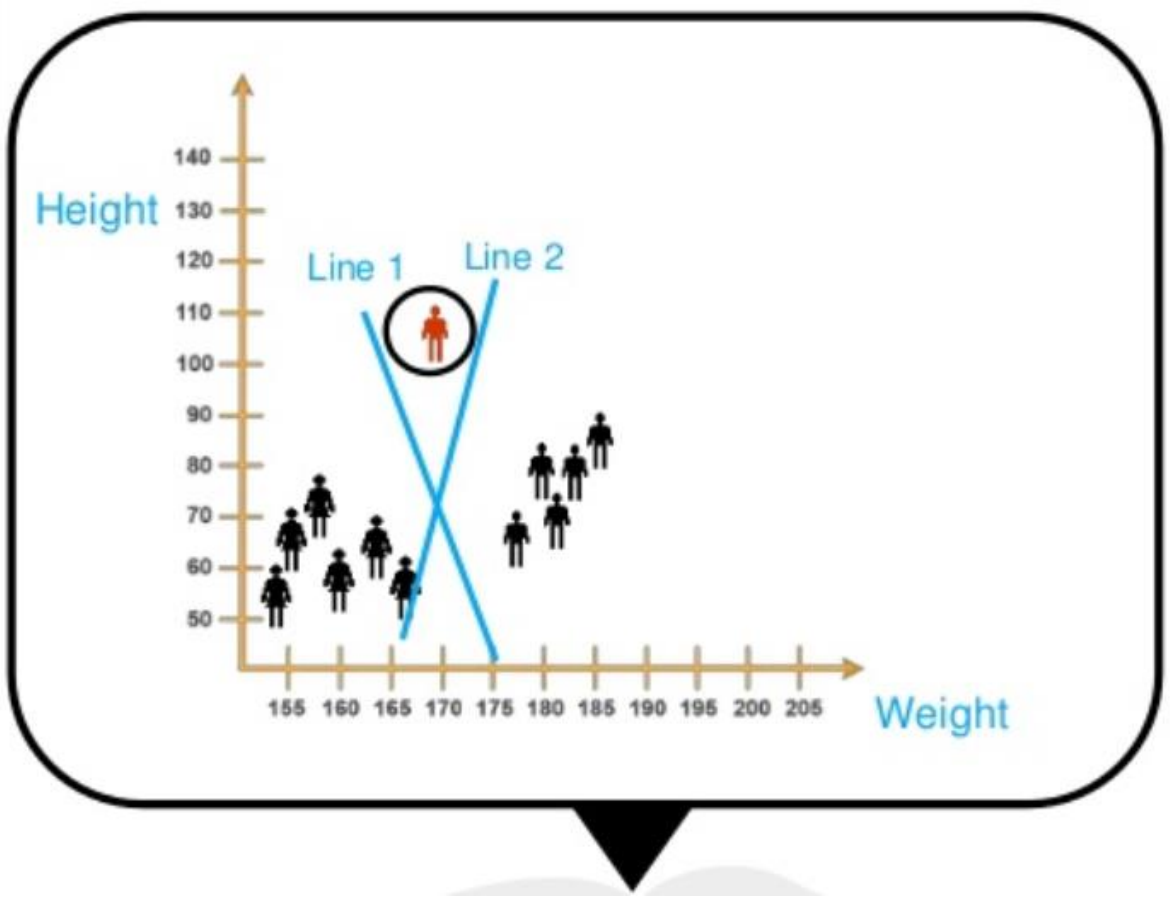
What is Support Vector Machine?

We can split our data by choosing any of these lines



What is Support Vector Machine?

But to predict the gender of a new data point we should split the data in the best possible way



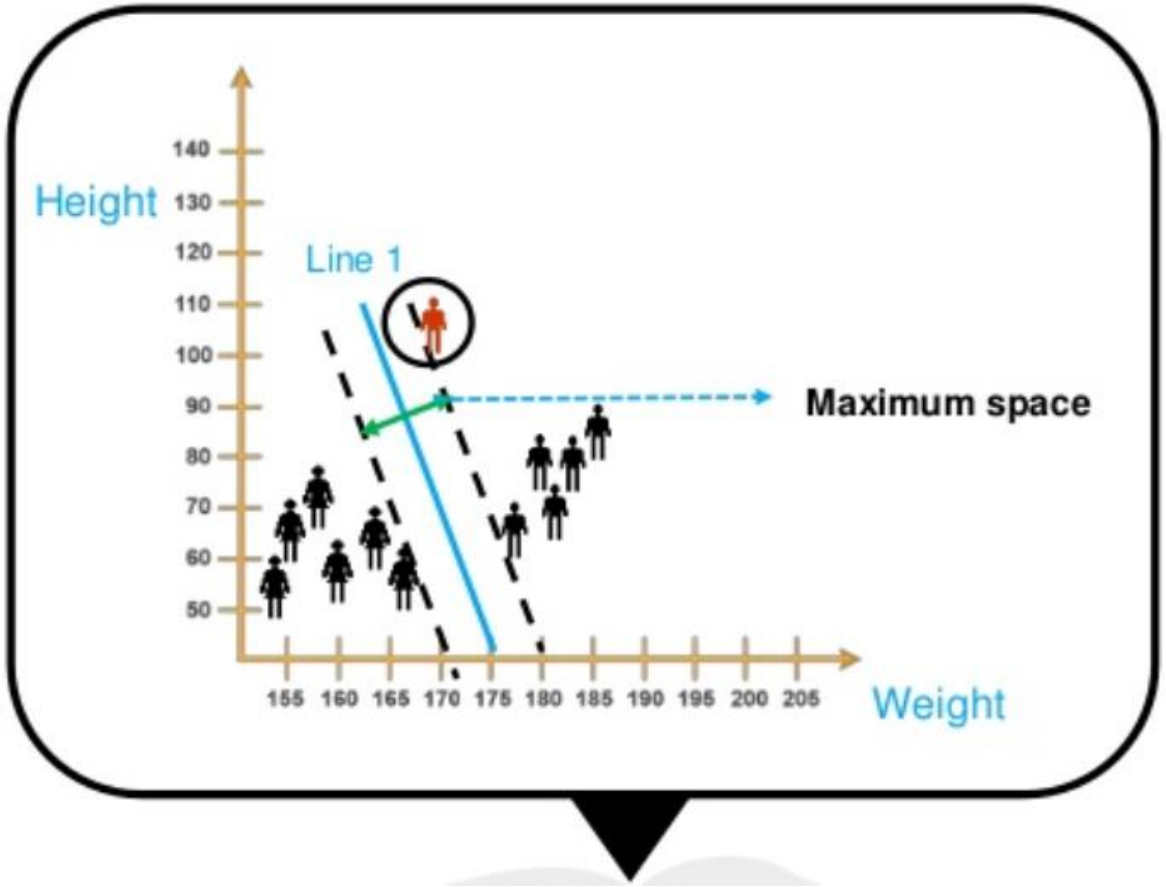
What is Support Vector Machine?

Then I would say, **Line 1** best splits the data

How do you say it is the best split?

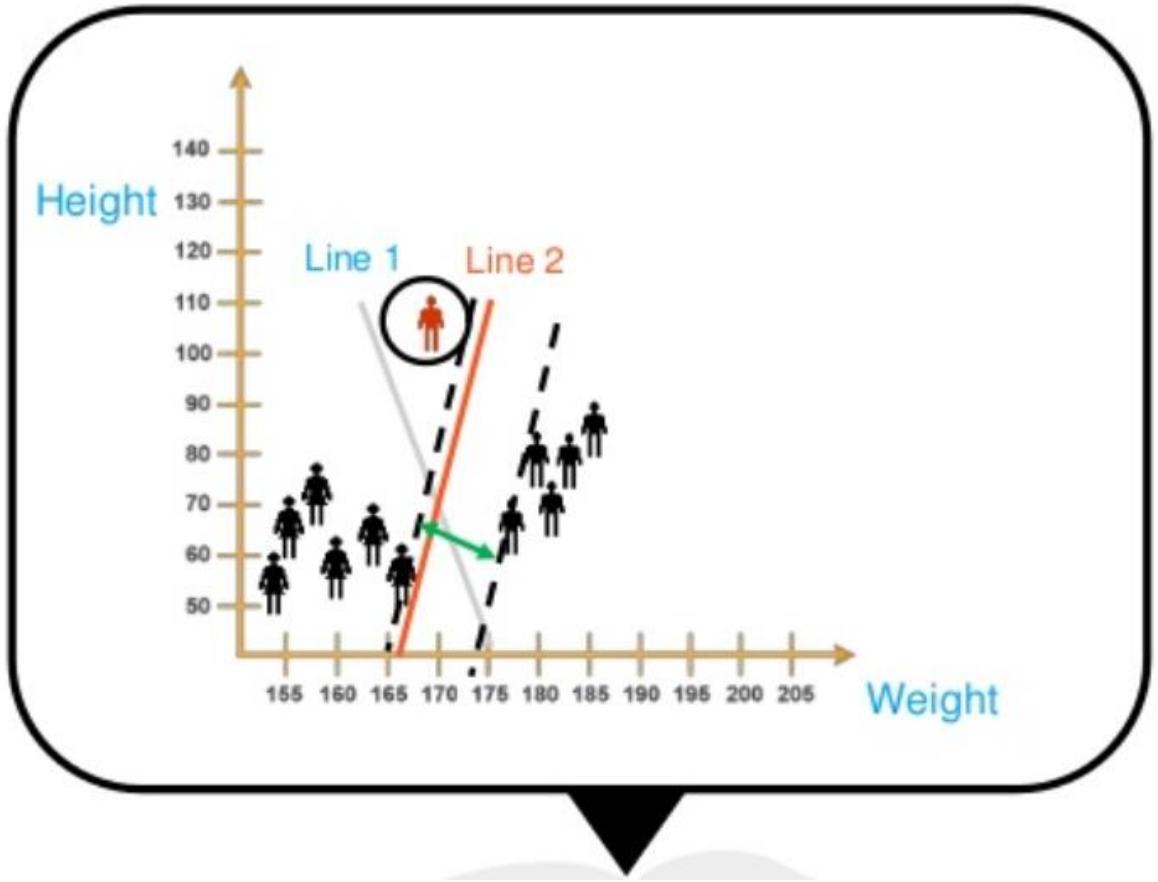
What is Support Vector Machine?

Because this line has the maximum space that separates the two classes



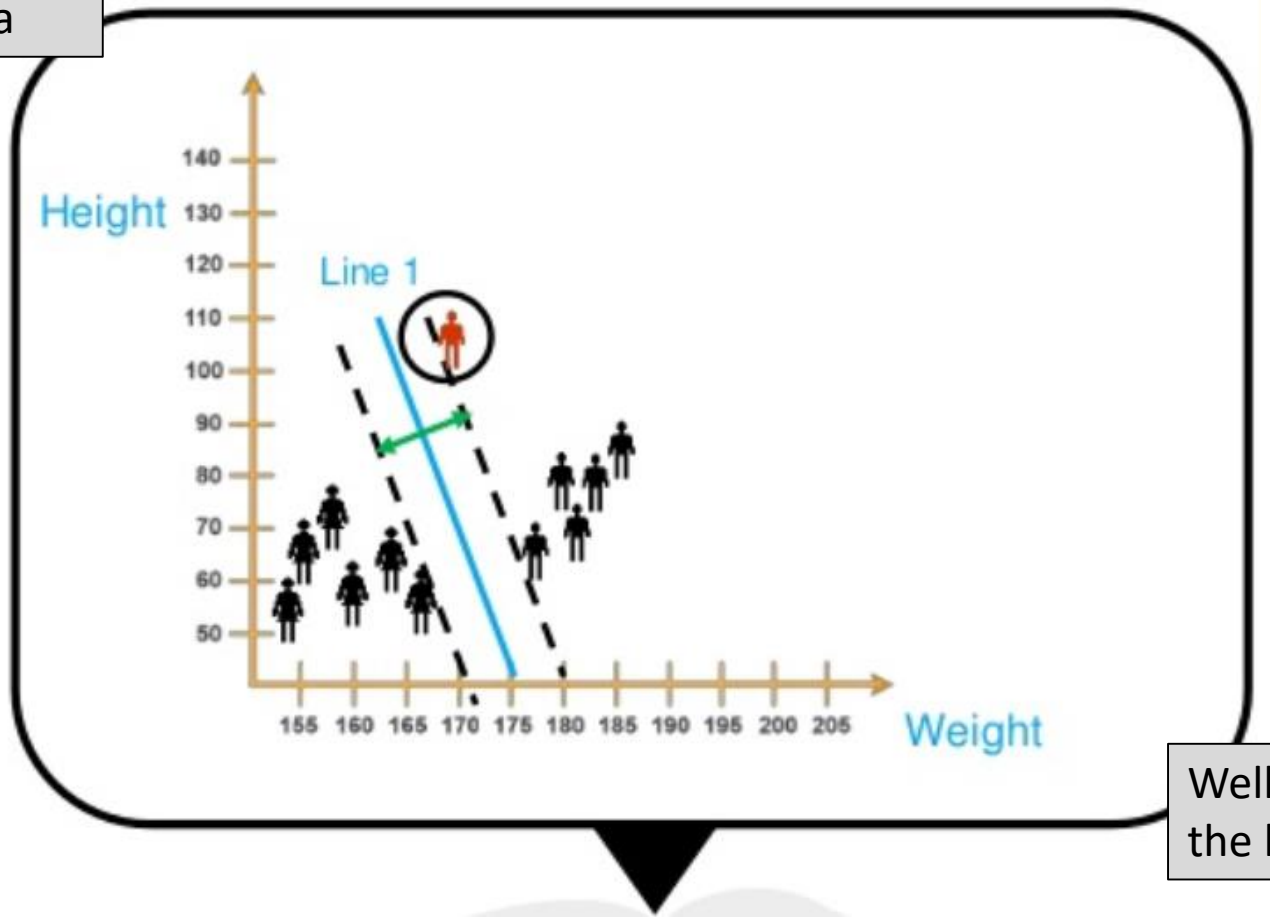
What is Support Vector Machine?

While the other line doesn't have maximum space that separates the two classes



What is Support Vector Machine?

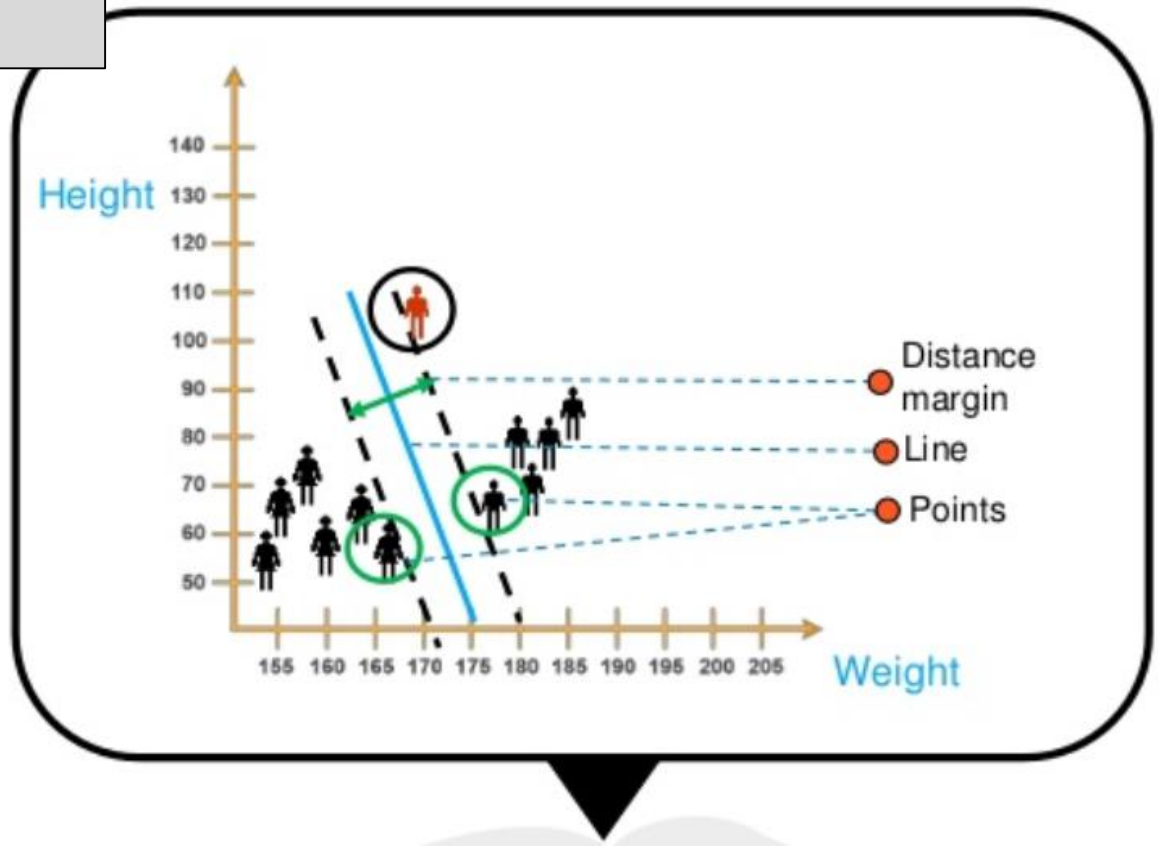
That is why this line best splits the data



Well yes… This is the best split

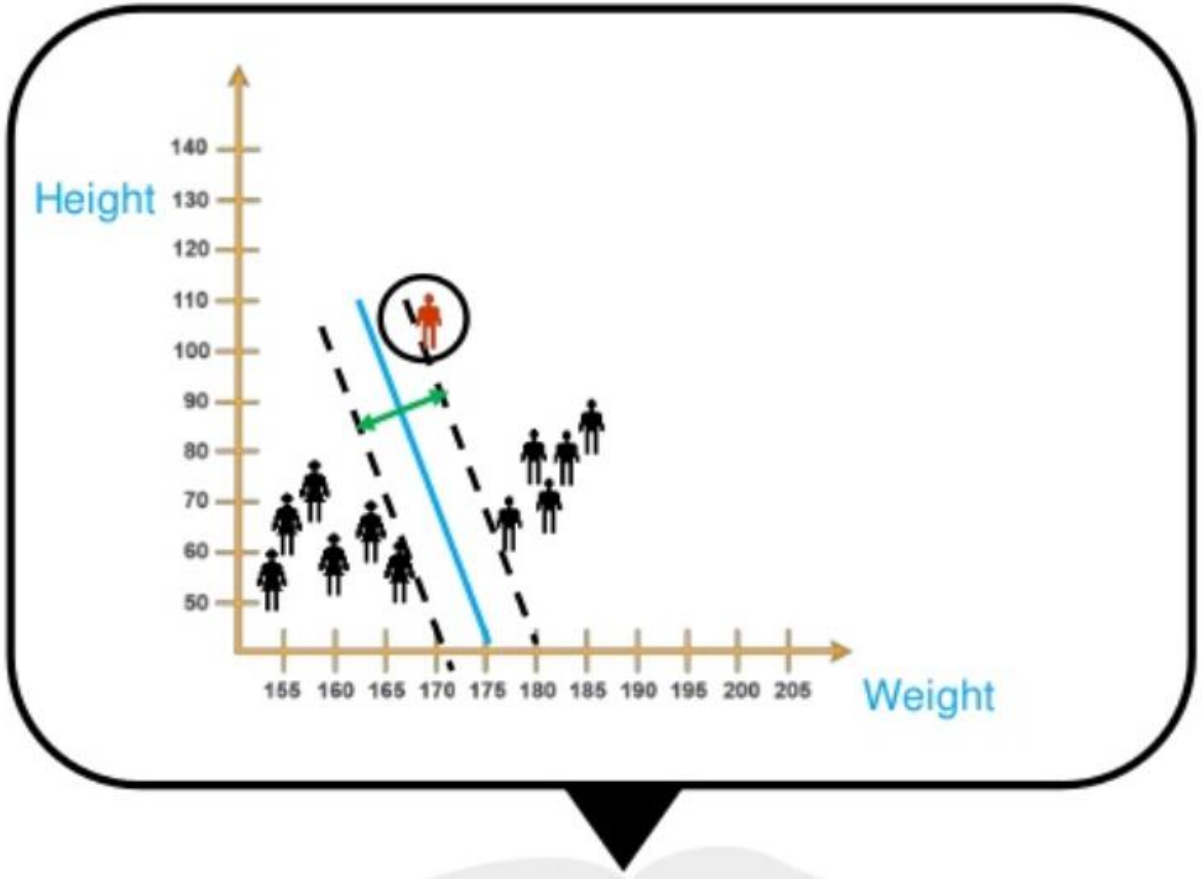
What is Support Vector Machine?

We can also say that the distance between points and the line should be far as possible



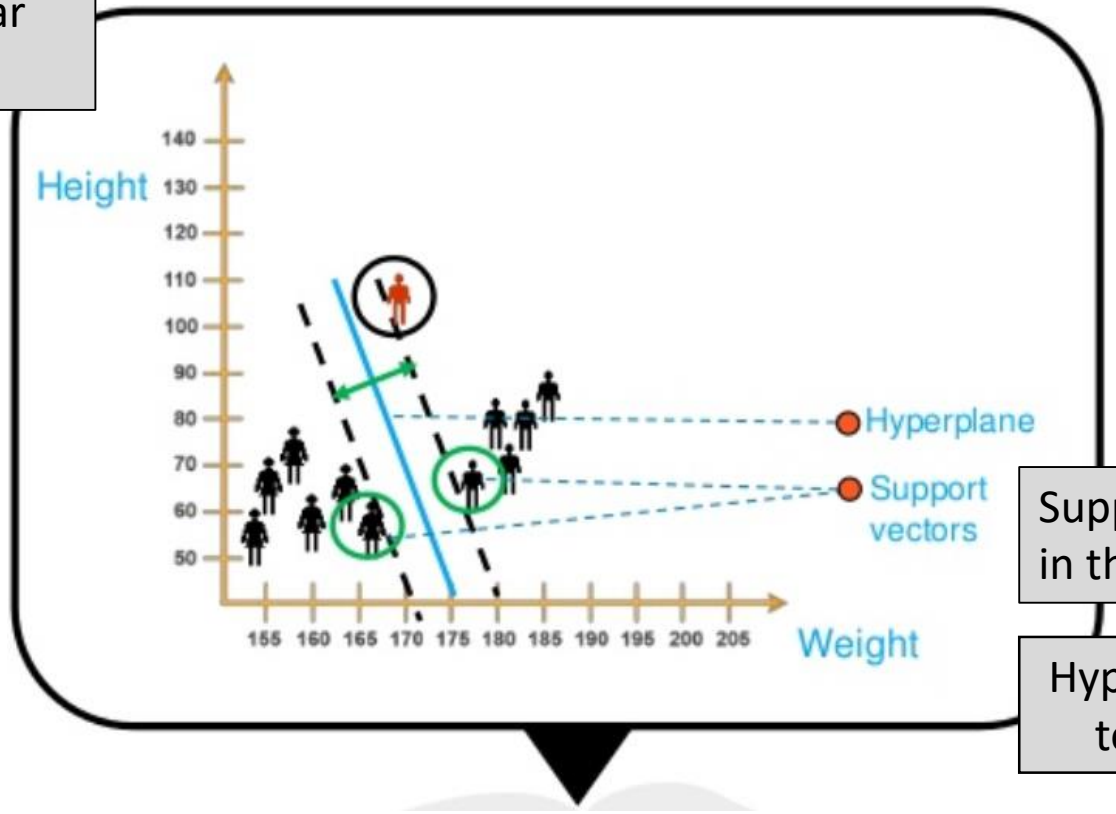
What is Support Vector Machine?

Now, let us look at it technically



What is Support Vector Machine?

In technical terms we can say, the distance between the support vector and the hyperplane should be far as possible

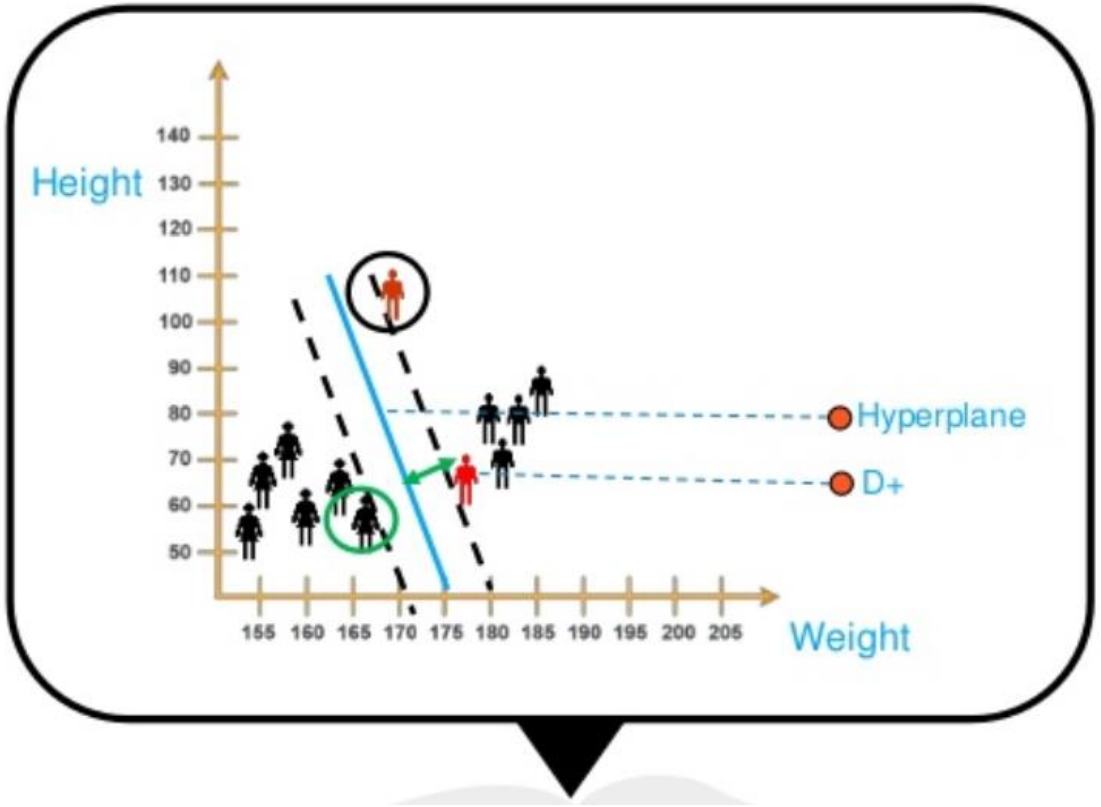


Support vectors are the extreme points in the datasets

Hyperplane has the maximum distance to the support vectors of any class

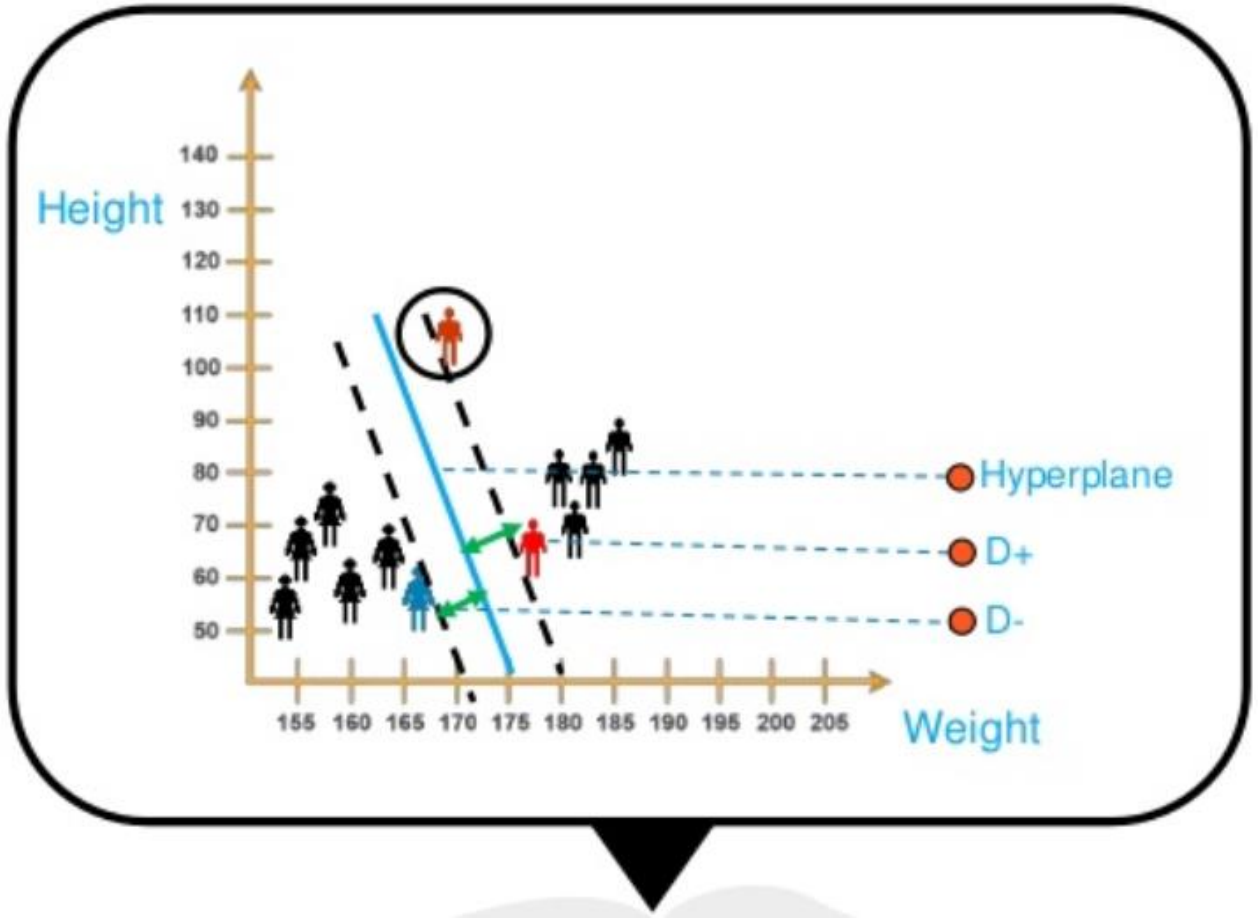
What is Support Vector Machine?

Here D+ is the shortest distance to the closest positive point



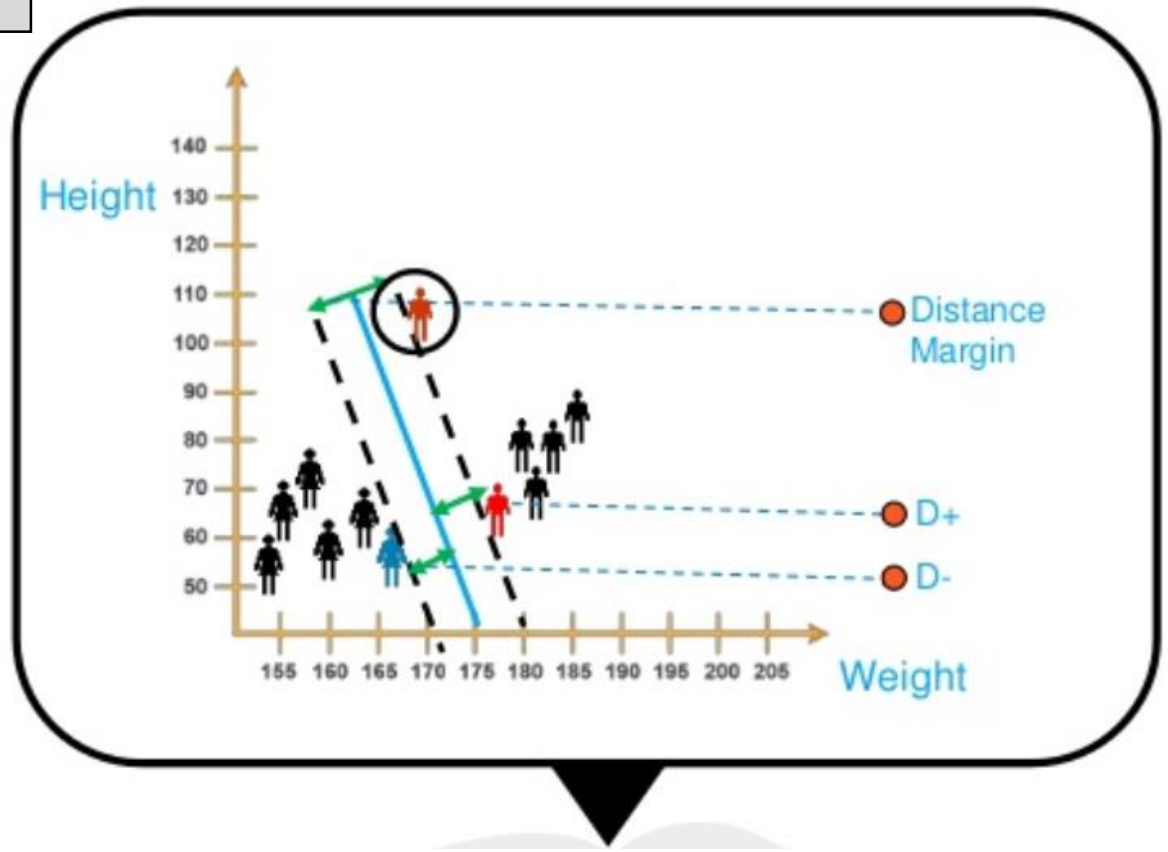
What is Support Vector Machine?

And D- is the shortest distance to the closest negative point



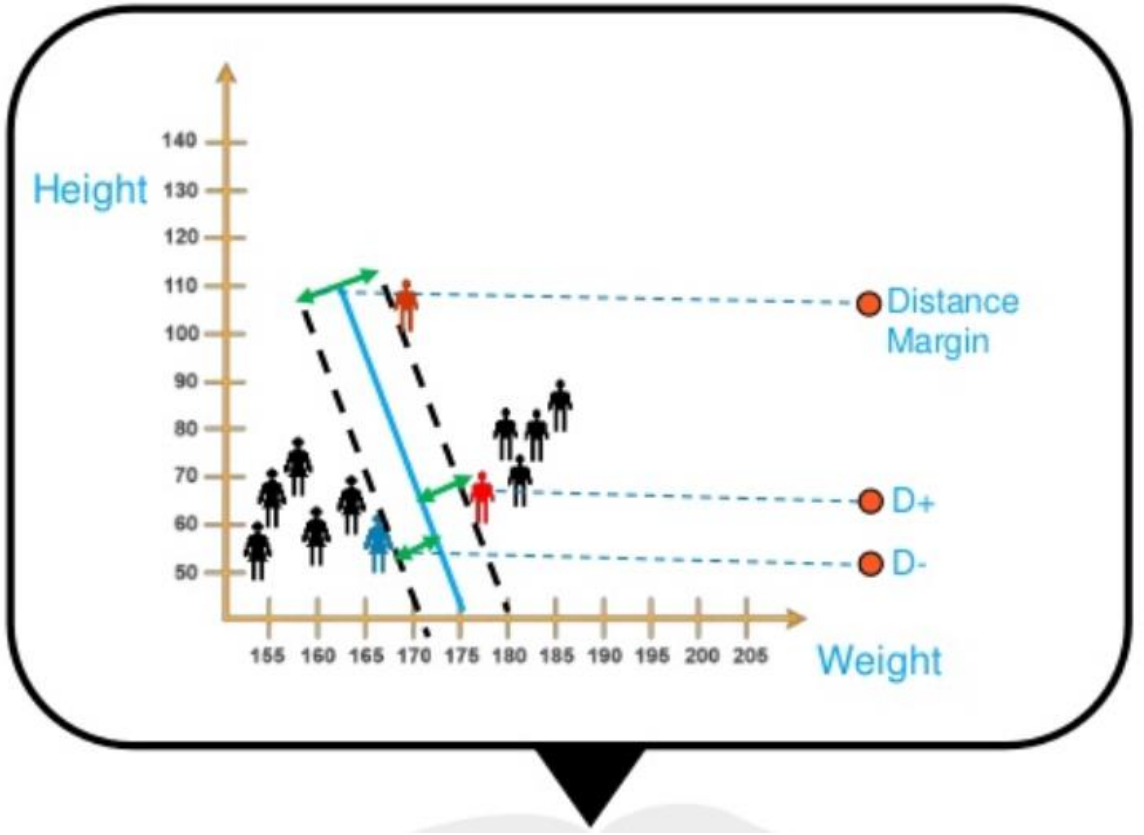
What is Support Vector Machine?

Sum of D+ and D- is called the distance margin



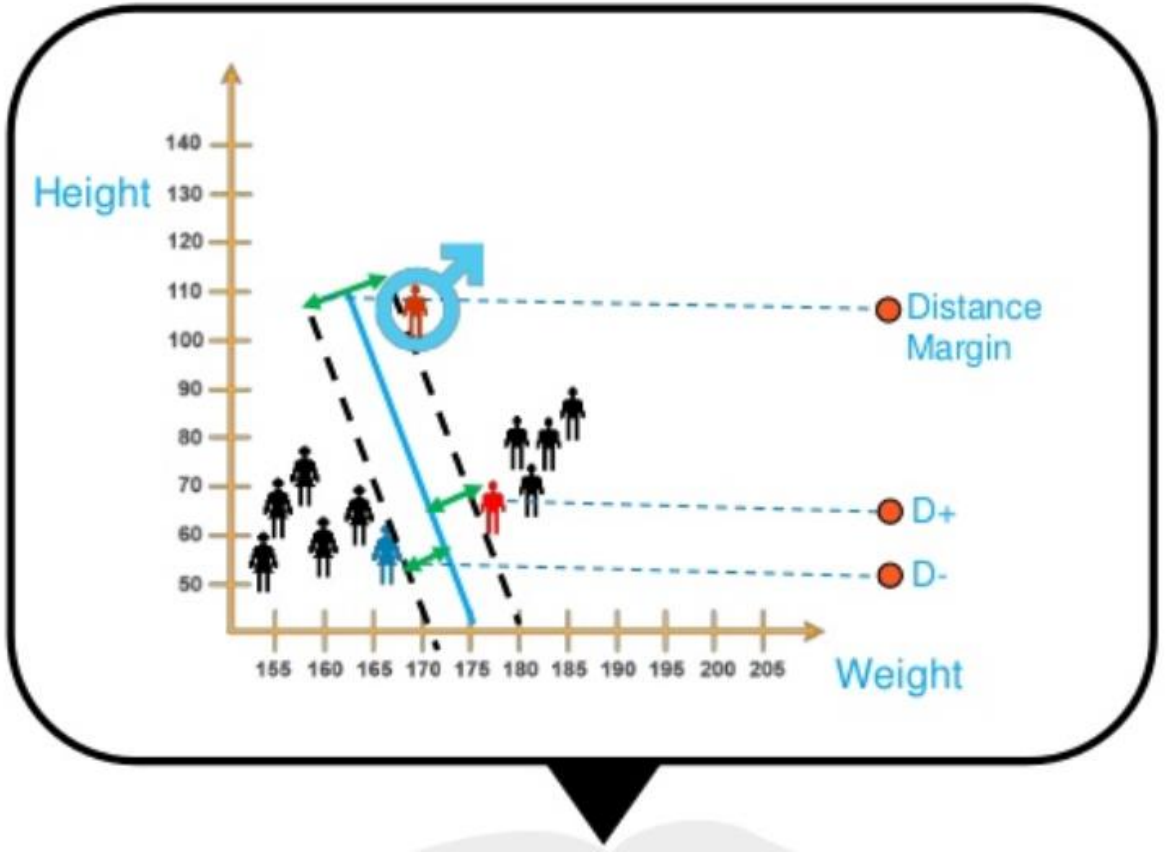
What is Support Vector Machine?

From the distance margin, we get an optimal hyperplane



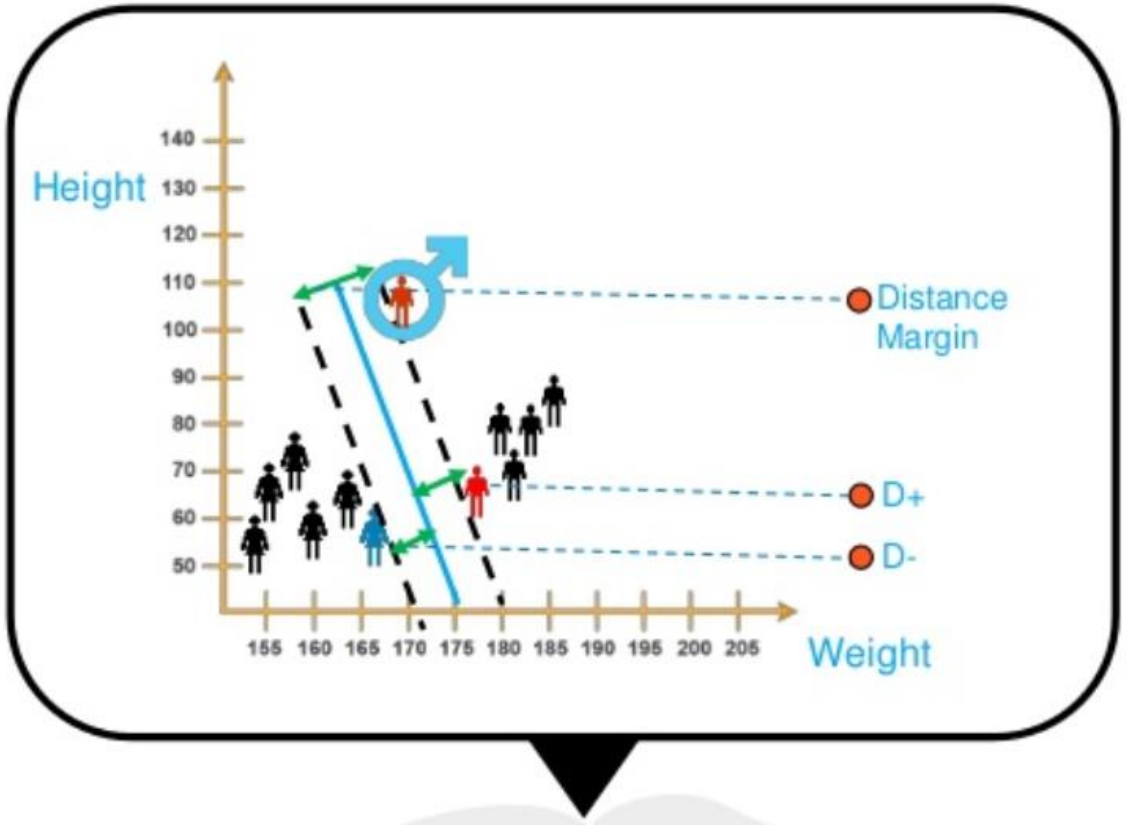
What is Support Vector Machine?

Based on the hyperplane, we can say the new data point belongs to male gender



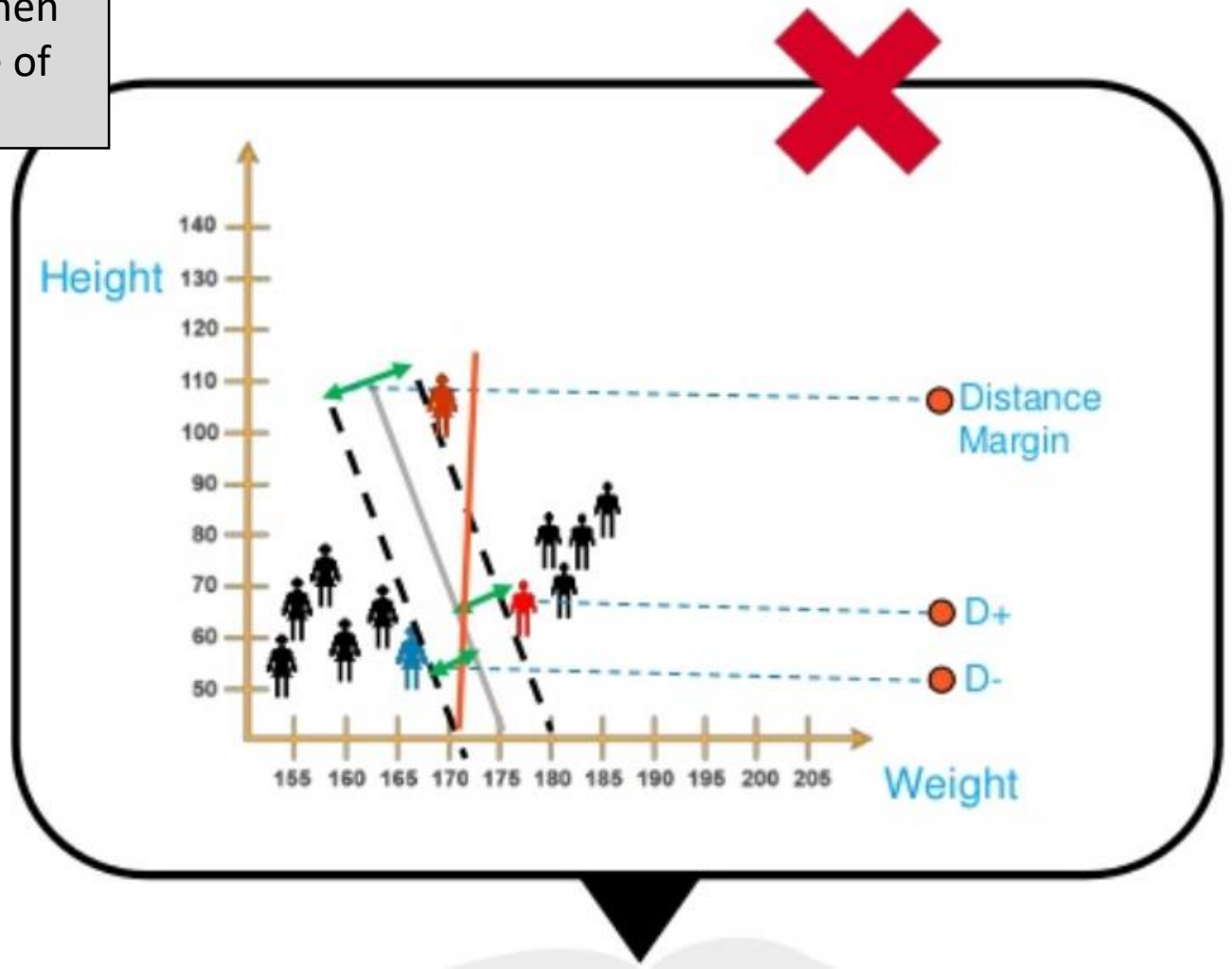
What is Support Vector Machine?

But what happens if a hyperplane is not optimal?



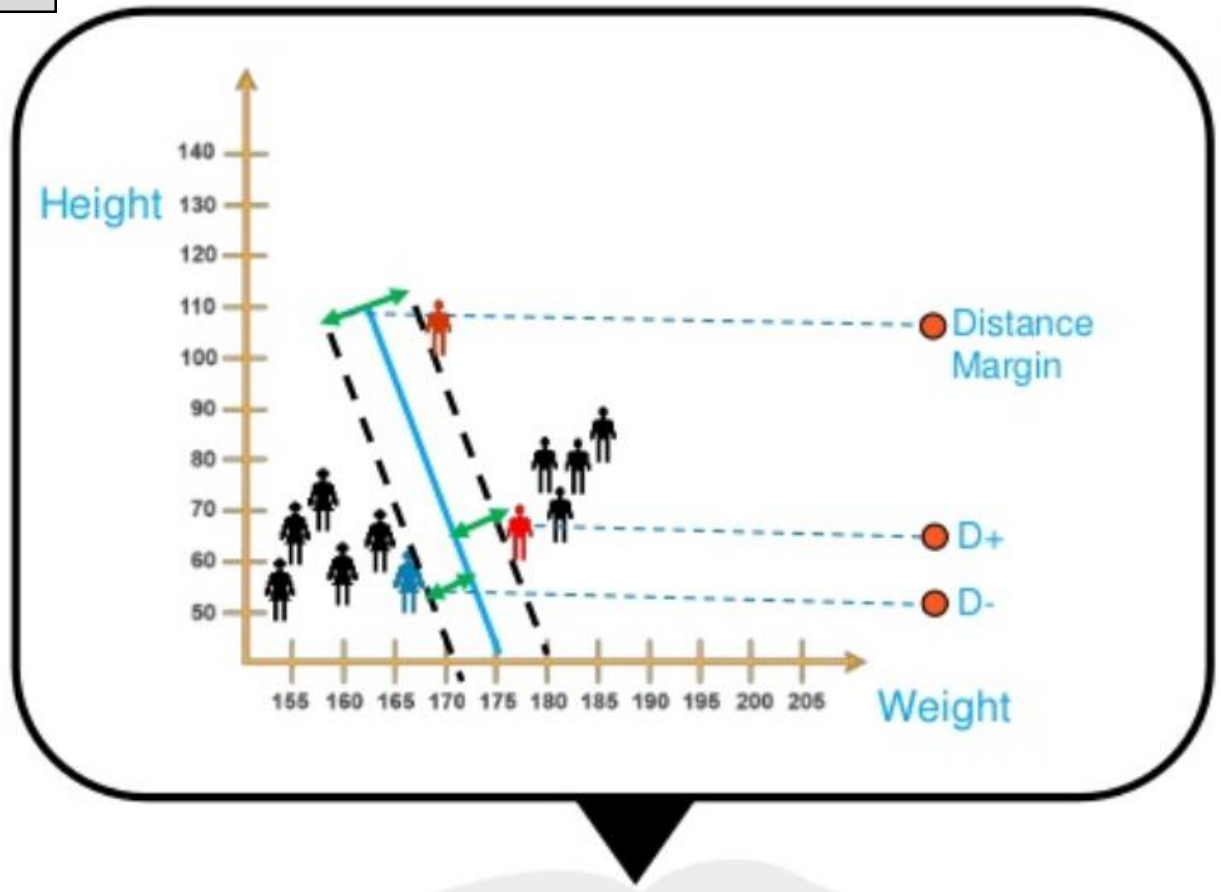
What is Support Vector Machine?

If we select a hyperplane having low margin then there is high chance of misclassification



What is Support Vector Machine?

What we discussed so far, is also called as **LSVM**



What is Support Vector Machine?

if $w^T x + b = 1$ y_+

if $w^T x + b = -1$ y_-

$(w^T x + b)y_+ = (1)y_+$

$(w^T x + b)y_- = (-1)y_-$

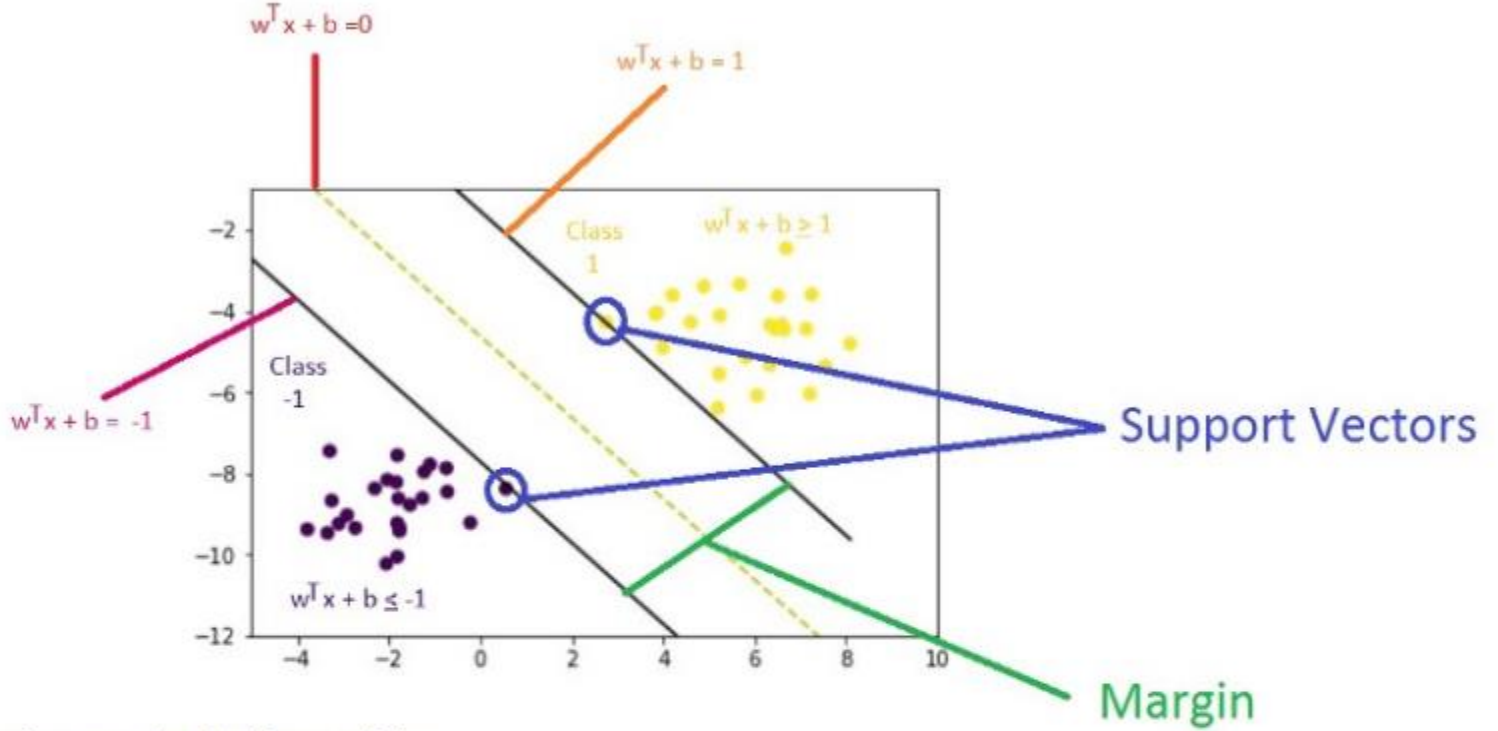
$y_+ = 1$, $y_- = -1$

$(w^T x + b)y_+ = (1)(1)$

$(w^T x + b)y_- = (-1)(-1)$

$$y_i\,(w^T x + b) = 1$$

$y_i = \{y_+ , y_-\}$



for all points we check this condition

if a point(x) yi(w.x+b)=1:*

point=support vector

classified correctly save parameters

else if > 1 :

classified correctly save parameters

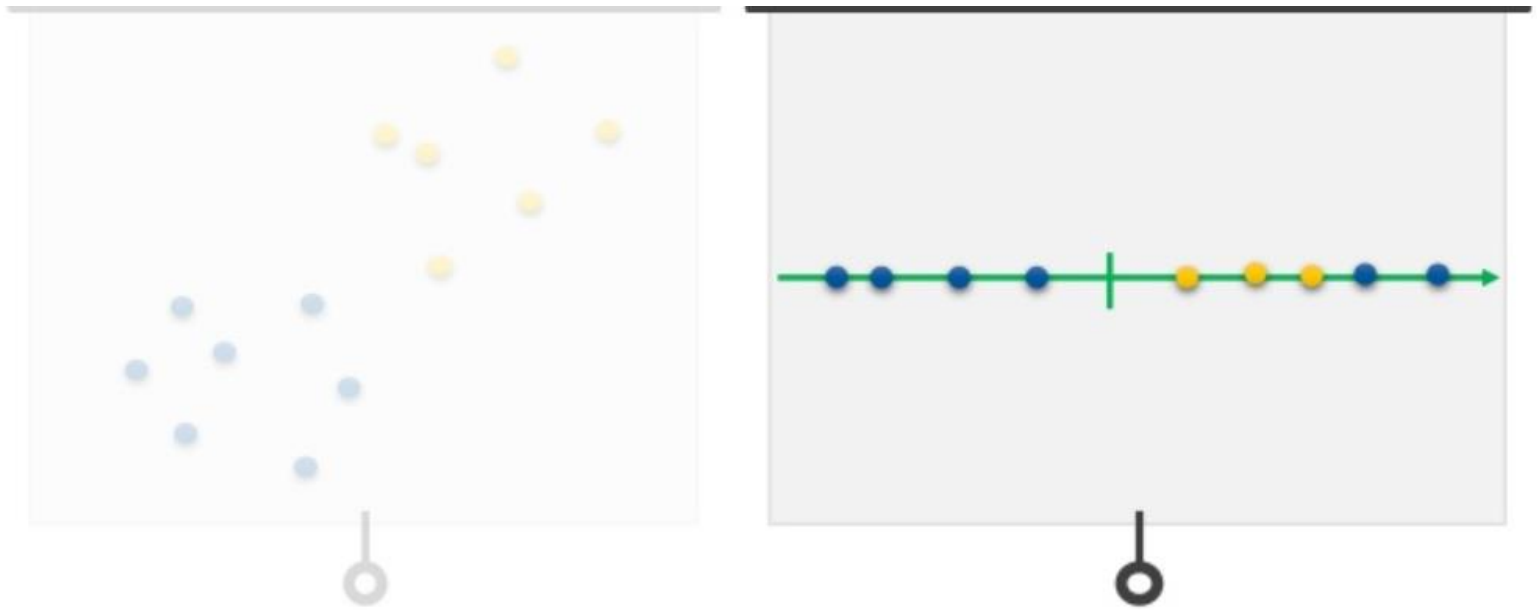
else :

classified incorrectly adjust parameters

(+) support vector -(-) support vector

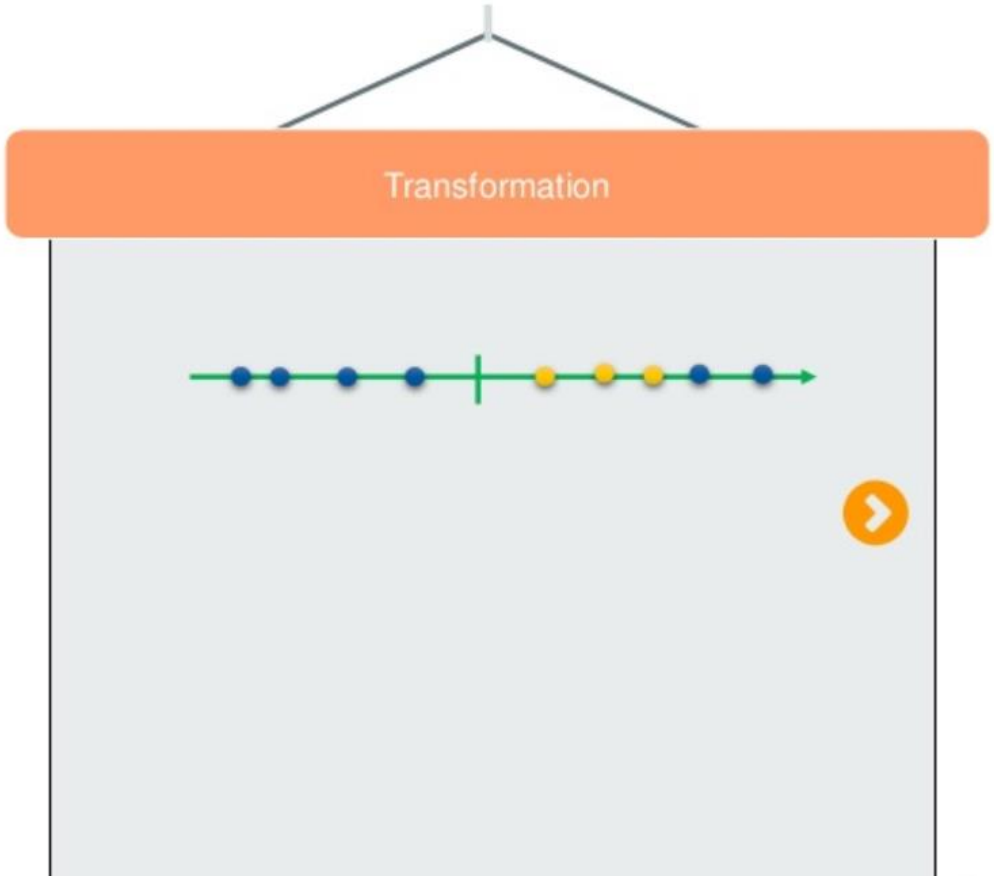
Understanding Support Vector Machine?

What if my data was like this?



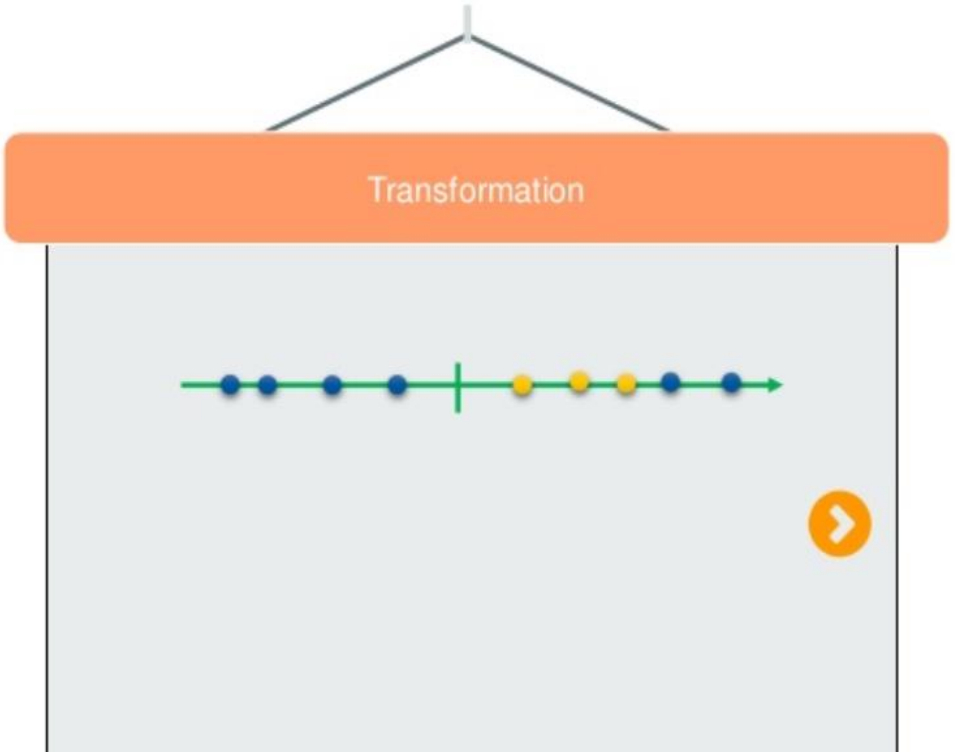
Understanding Support Vector Machine?

Here we cant use a hyperplane



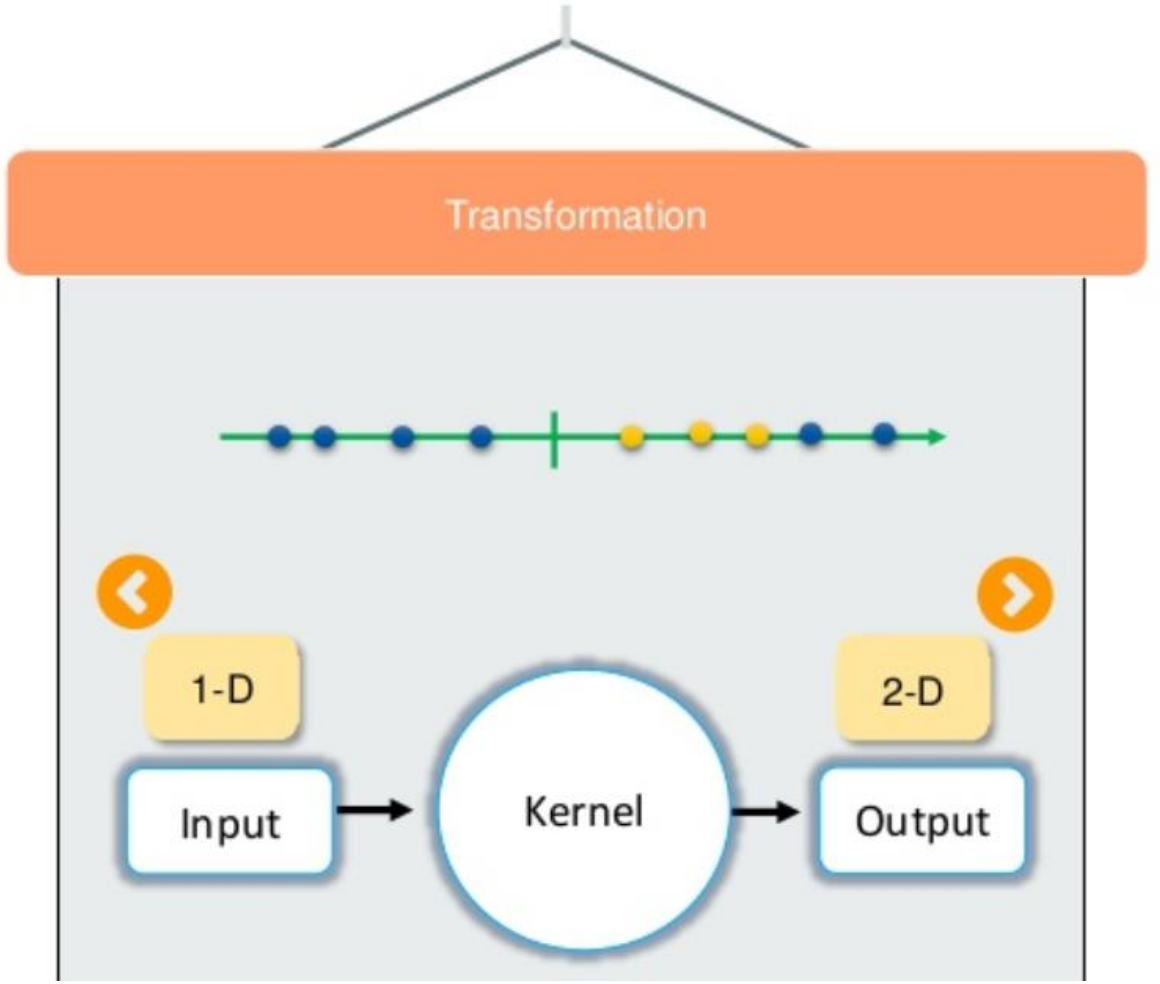
Understanding Support Vector Machine?

So it is necessary to move away from 1D view of data to 2D view data



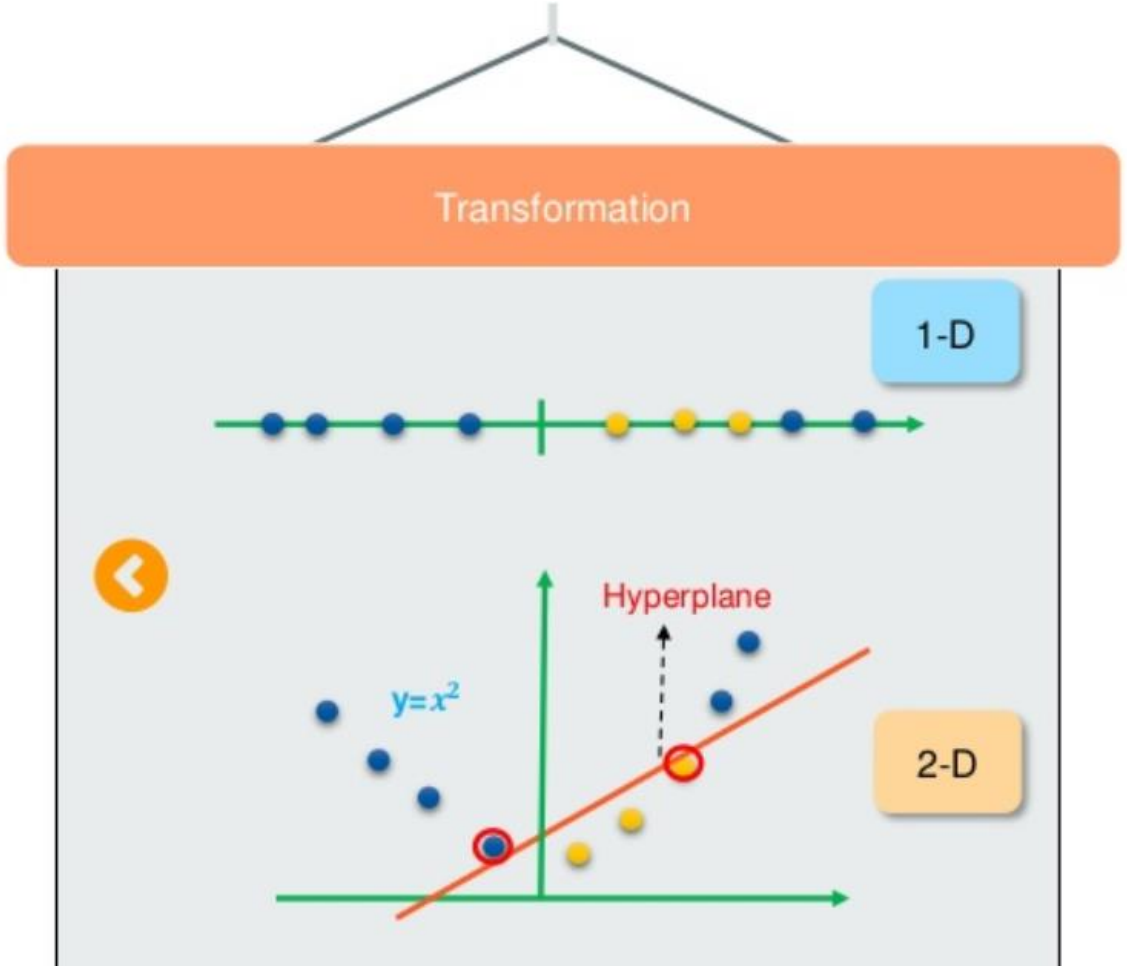
Understanding Support Vector Machine?

For the transformation we use kernel function which will take the 1-D input and transfer it to 2-D



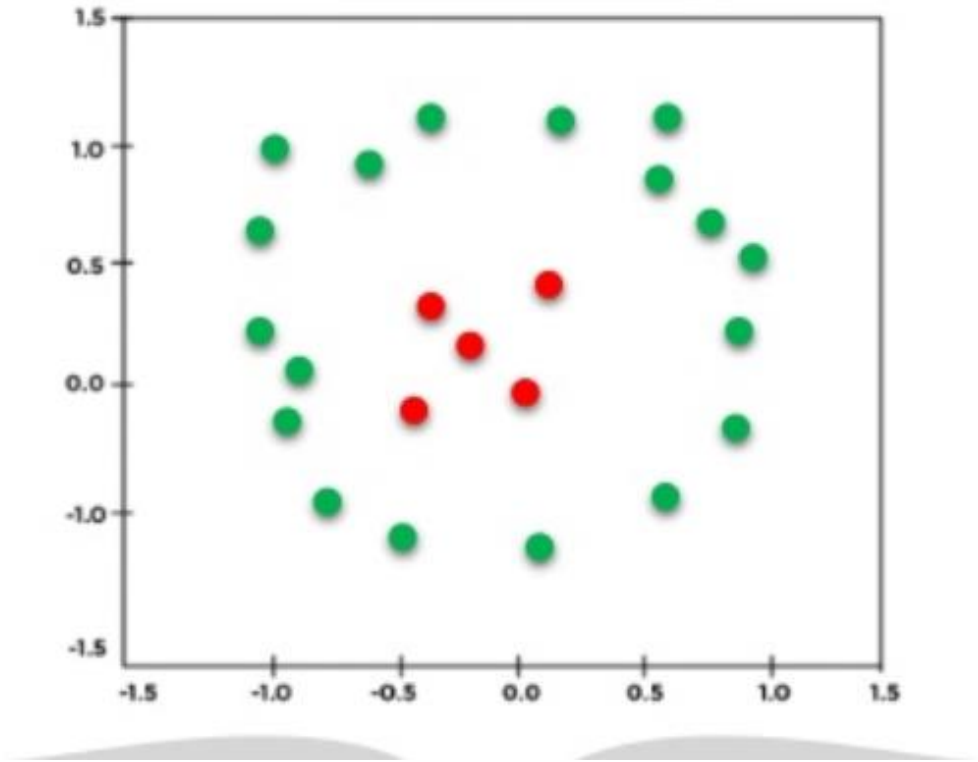
Understanding Support Vector Machine?

Now we got the result



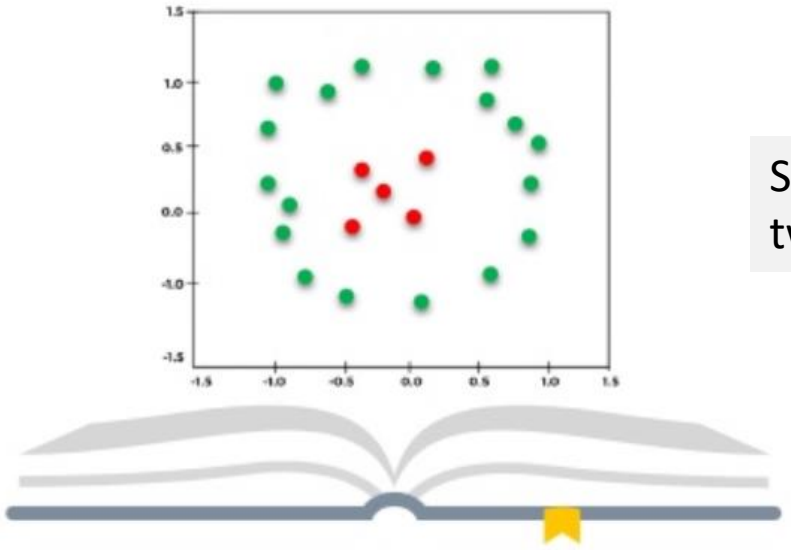
Understanding Support Vector Machine?

How to perform SVM for this type of dataset

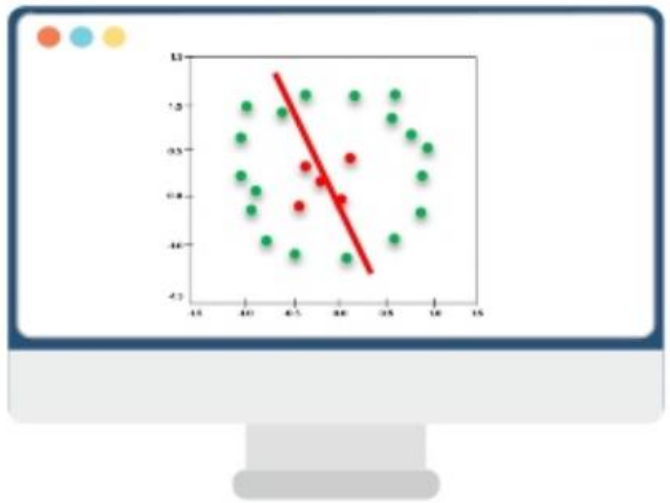


Understanding Support Vector Machine?

How to perform SVM for this type of dataset



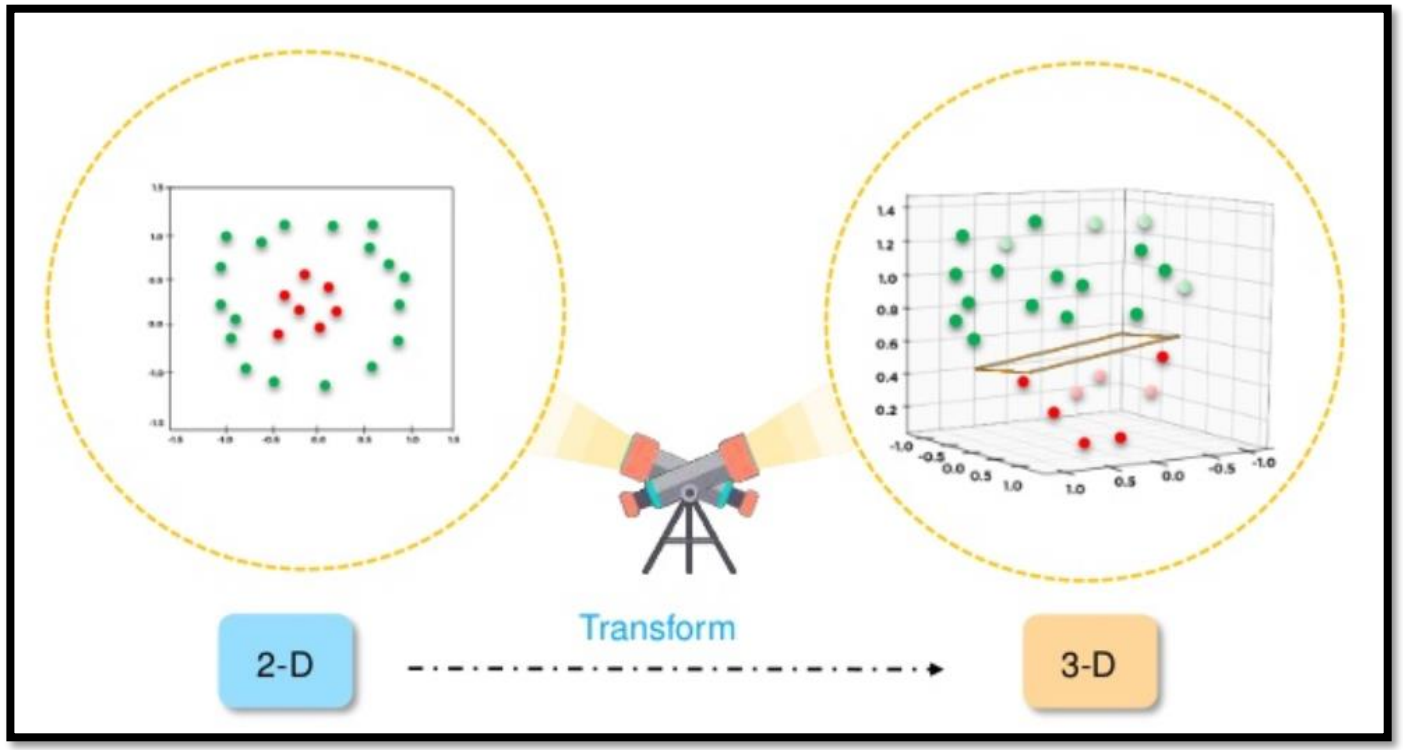
Segregate the two classes



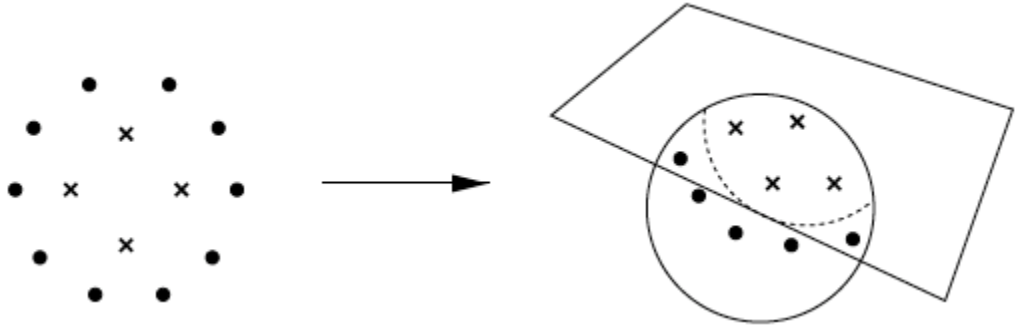
Not an optimal hyperplane

Understanding Support Vector Machine?

Kernel



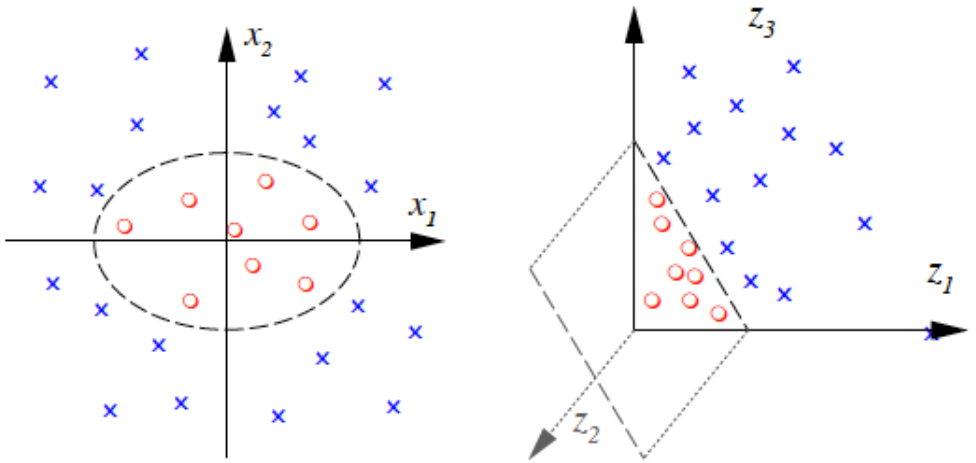
Polynomial kernel



$$k(x, y) = (\alpha x^T y + c)^d$$

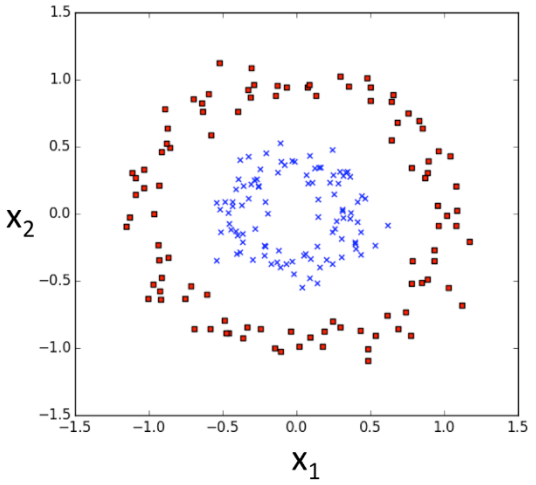
- The slope is alpha and c is the constant.
- These parameters are adjustable.
- The d is the polynomial degree

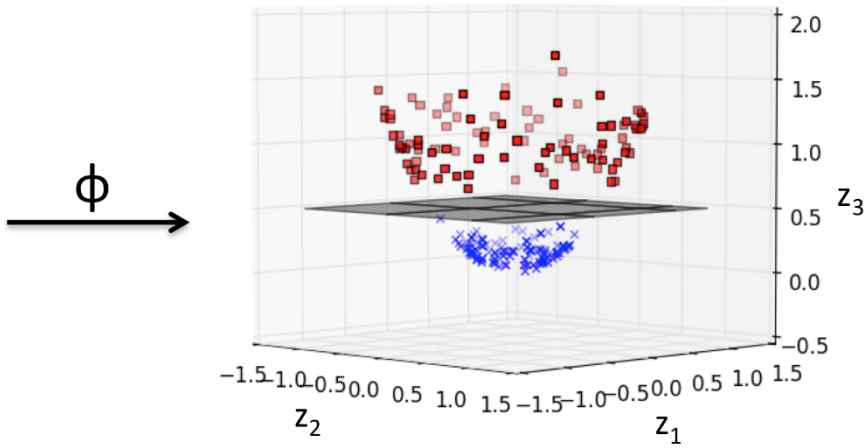
$$\Phi : R^2 \to R^3$$
$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{(2)} x_1 x_2, x_2^2)$$

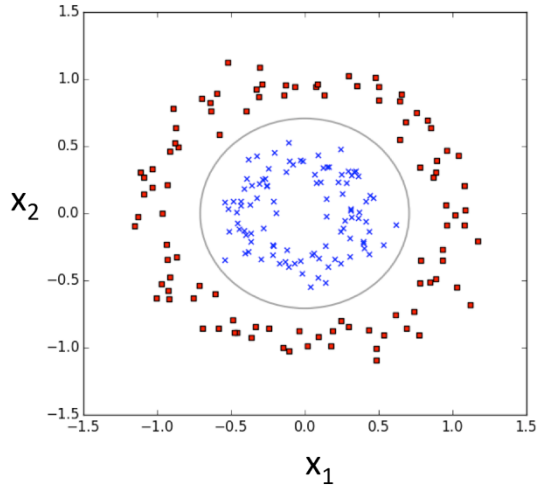


RBF kernel

$$K(x, y) = exp(-\gamma \sum_{j=1}^{p} (x_{ij} - y_{ij})^2)$$







γ here is a tuning parameter,

- which accounts for the smoothness of the decision boundary controls
- the variance of the model

Thank you