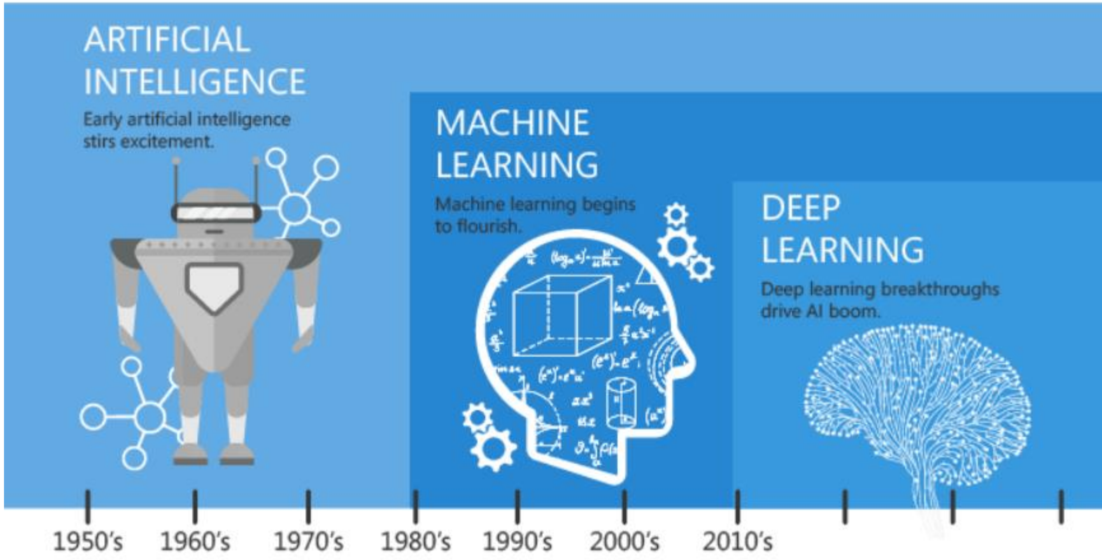
Lecture 15-16
Artificial Intelligence

Khola Naseem
khola.naseem@uet.edu.pk



Machine learning

➢Types of learning

➢ Machine learning implementations are classified into four major categories, depending on the nature of the learning

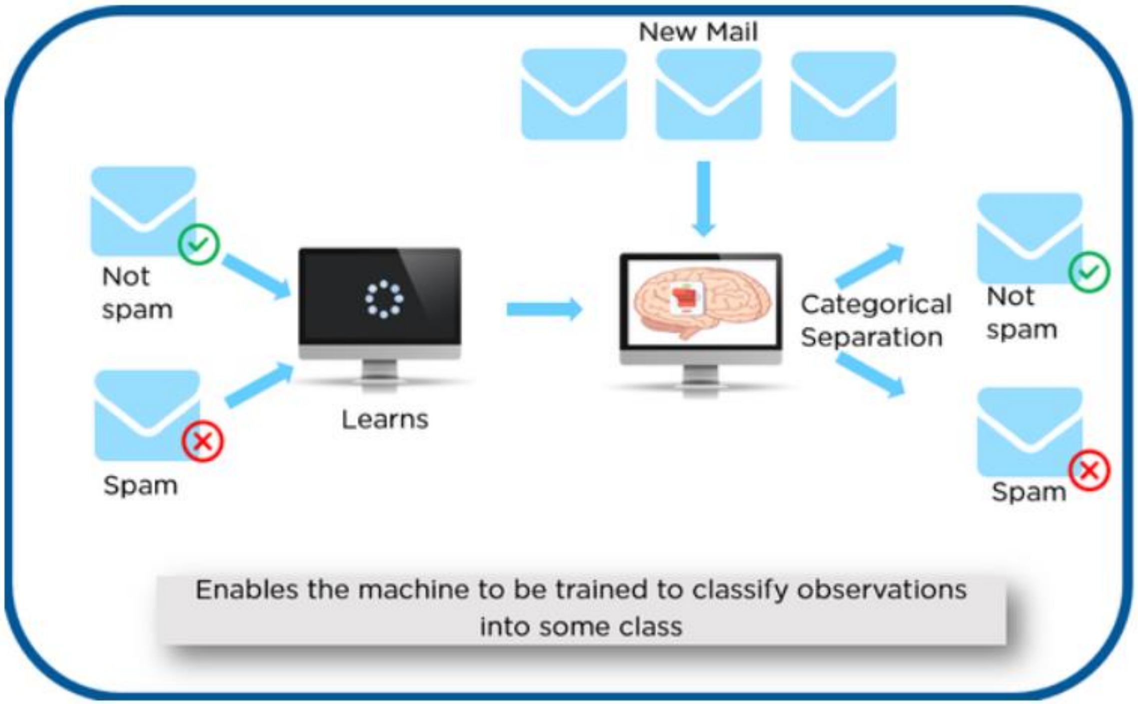
1. Supervised learning
2. Unsupervised learning
3. Reinforcement learning
4. Semi-supervised learning

Machine learning

- Supervised learning:
- Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs.
- The given data is labeled
- Both *classification* and *regression* problems are supervised learning problems
  - Example — Consider the following data regarding patients entering a clinic . The data consists of the gender and age of the patients and each patient is labeled as “healthy” or “sick”.

Machine learning

- Supervised learning:
- Example
- Spam Filter



Machine learning

- Supervised learning:
- Example
- Consider the following data regarding patients entering a clinic The data consists of the gender and age of the patients and each patient is labeled as “healthy” or “sick”.

gender	age	label
M	48	sick
M	67	sick
F	53	healthy
M	49	sick
F	32	healthy
M	34	healthy
M	21	healthy

Machine learning

- Unsupervised learning:
  - Unsupervised learning is a type of machine learning algorithm used to draw inferences from datasets consisting of input data without labeled responses.
  - In unsupervised learning algorithms, classification or categorization is not included in the observations.
  - Consider the following data regarding patients entering a clinic. The data consists of the gender and age of the patients.

gender	age
M	48
M	67
F	53
M	49
F	34
M	21

Credit: Khola Naseem

Machine learning

- Unsupervised learning:
  - As a kind of learning, it resembles the methods humans use to figure out that certain objects or events are from the same class, such as by observing the degree of similarity between objects.
  - Some recommendation systems that you find on the web in the form of marketing automation are based on this type of learning.

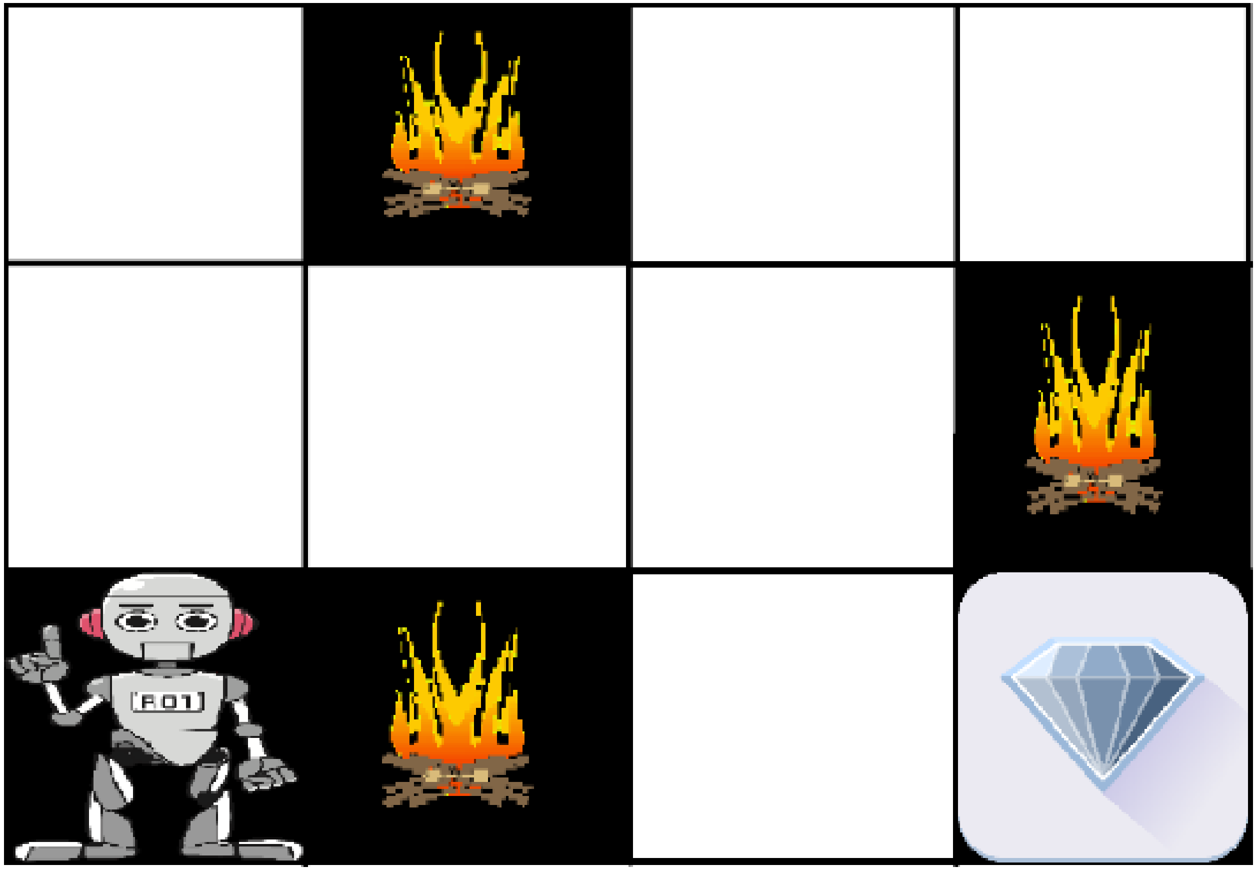
Machine learning

- Reinforcement Learning:
- Reinforcement learning is the problem of getting an agent to act in the world so as to maximize its rewards.
- A learner is not told what actions to take as in most forms of machine learning but instead must discover which actions yield the most reward by trying them.
- For example:
  - Consider teaching a dog a new trick: we cannot tell it what tell it to do what to do, but we can reward/punish it if it does the right/wrong thing.

Credit: Khola Naseem

Machine learning

- Reinforcement Learning:



Credit: Khola Naseem

Machine learning

- Sami-Supervised Learning:
  - Training set with some (often many) of the target outputs missing.
  - Semi-supervised learning is an approach to machine learning that combines small labeled data with a large amount of unlabeled data during training.
  - Semi-supervised learning falls between unsupervised learning and supervised learning.

Supervised Machine learning

Supervised Machine learning

- An example application
  - An emergency room in a hospital measures 17 variables (e.g., blood pressure, age, etc) of newly admitted patients.
  - A decision is needed: whether to put a new patient in an intensive-care unit.
  - Due to the high cost of ICU, those patients who may survive less than a month are given higher priority.
  - Problem: to predict high-risk patients and discriminate them from low-risk patients.

Supervised Machine learning

- An example application
  - A credit card company receives thousands of applications for new cards. Each application contains information about an applicant,
    - age
    - Marital status
    - annual salary
    - outstanding debts
    - credit rating , etc.
  - Problem: to decide whether an application should approved, or to classify applications into two categories, approved and not approved.

Supervised Machine learning

- Like human learning from past experiences.
- A computer does not have "experiences".
- A computer system learns from data, which represent some "past experiences" of an application domain.
- Our focus: learn a target function that can be used to predict the values of a discrete class attribute, e.g., approve or not-approved, and high-risk or low risk.
- The task is commonly called: Supervised learning, classification, or inductive learning.

The data and the goal

- Data: A set of data records (also called examples, instances or cases) described by
  - k attributes: A1, A2, … Ak.
  - a class: Each example is labelled with a pre-defined class.
- Goal: To learn a classification model from the data that can be used to predict the classes of new (future, or test) cases/instances.

The data and the goal

- Data:

Approved or not

ID	Age	Has_Job	Own_House	Credit_Rating	Class
1	young	false	false	fair	No
2	young	false	false	good	No
3	young	true	false	good	Yes
4	young	true	true	fair	Yes
5	young	false	false	fair	No
6	middle	false	false	fair	No
7	middle	false	false	good	No
8	middle	true	true	good	Yes
9	middle	false	true	excellent	Yes
10	middle	false	true	excellent	Yes
11	old	false	true	excellent	Yes
12	old	false	true	good	Yes
13	old	true	false	good	Yes
14	old	true	false	excellent	Yes
15	old	false	false	fair	No

Credit: Khola Naseem

The data and the goal

- Learn a classification model from the data
- Use the model to classify future loan applications into
  - Yes (approved) and
  - No (not approved)
- What is the class for following case/instance?

Age	Has_Job	Own_house	Credit-Rating	Class
young	false	false	good	?

The data and the goal

- Learn a classification model from the data
- What is the class for following case/instance?

Age	Has_Job	Own_house	Credit-Rating	Class
young	false	false	good	?

- No learning: classify all future applications (test data) to the majority class (i.e., Yes):

Accuracy = 9/15 = 60%.

- We can do better than 60% with learning.

Supervised machine learning:

- Data:
- Data: labeled instances <xi, y>, e.g. emails marked spam/not spam
  - Training Set
  - Held-out Set/validation set
  - Test Set
- Features: attribute-value pairs which characterize each x
- Learn parameters (e.g. model probabilities) on training set (Tune hyper-parameters on held-out set)
- Compute accuracy of test set
- Very important: never “peek” at the test set!

Credit: Khola Naseem

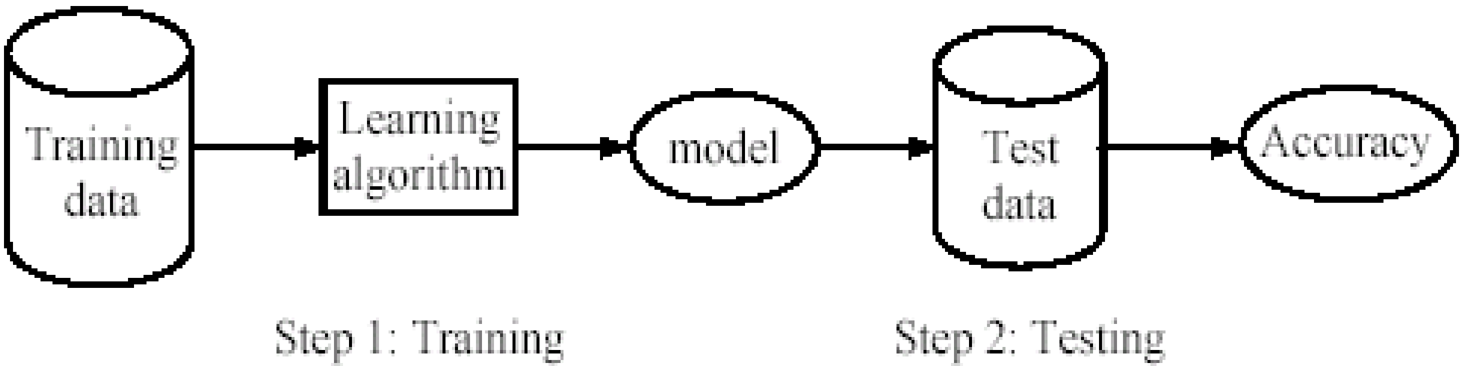
Fundamental assumption of learning:

- Data:
- Assumption: The distribution of training examples is identical to the distribution of test examples (including future unseen examples).
- In practice, this assumption is often violated to certain degree.
- Strong violations will clearly result in poor classification accuracy.
- To achieve good accuracy on the test data, training examples must be sufficiently representative of the test data.

Supervised machine learning:

- Two steps:
  - Learning (training): Learn a model using the training data
  - Testing: Test the model using unseen test data to assess the model accuracy

$$Accuracy = \frac{\text{Number of correct classifications}}{\text{Total number of test cases}},$$



Credit: Khola Naseem

Supervised machine learning:

- Like all machine learning algorithms, supervised learning is based on training.
- During its training phase, the system is fed with labeled data sets, which instruct the system what output is related to each specific input value.
- The trained model is then presented with test data: This is data that has been labeled, but the labels have not been revealed to the algorithm.
- The aim of the testing data is to measure how accurately the algorithm will perform on unlabeled data.

Supervised machine learning:

- Supervised learning:

Given a **training set** of N example input–output pairs

$$(x_1, y_1), (x_2, y_2), \ldots (x_N, y_N) ,$$

where each y_j was generated by an unknown function $y = f(x)$,

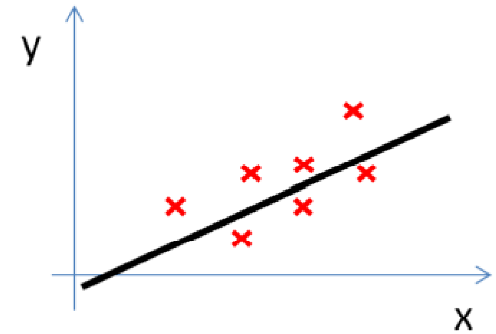
- Here x and y can be any value; they need not be numbers.
- The function is a hypothesis.
- Learning is a search through the space of possible hypothesis for one that will perform well, even on new examples beyond the training set

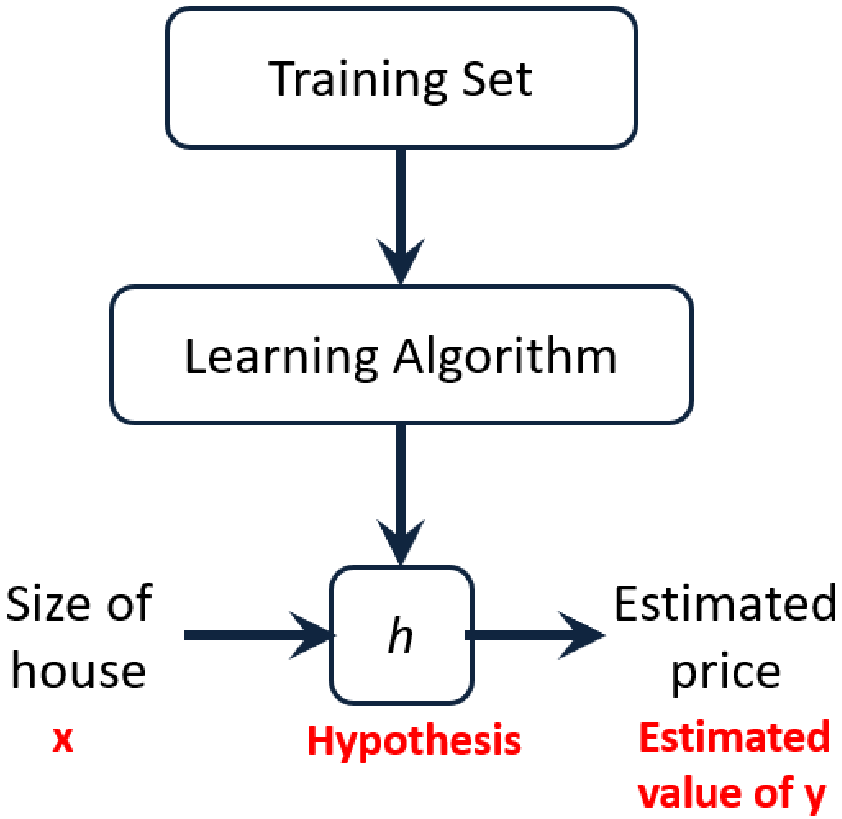
Supervised learning:

- h is a function
- h maps from x's to y's

univariate linear eq

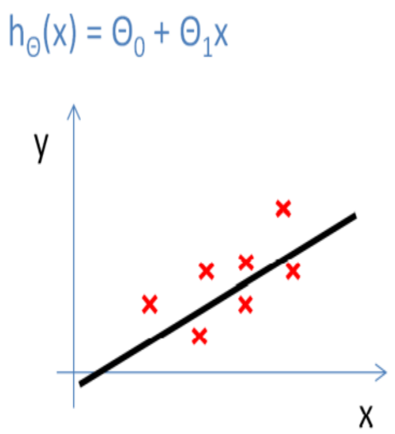
$h_\Theta(x) = \Theta_0 + \Theta_1 x$



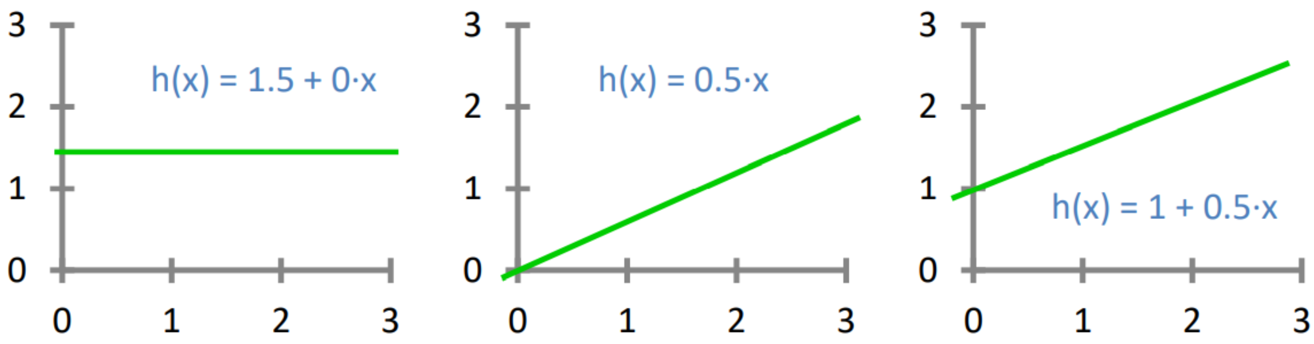


Supervised learning:

- h



$$h_\theta(x) = \theta_0 + \theta_1 x$$



$\theta_0 = 1.5$ $\theta_1 = 0$ | $\theta_0 = 0$ $\theta_1 = 0.5$ | $\theta_0 = 1$ $\theta_1 = 0.5$

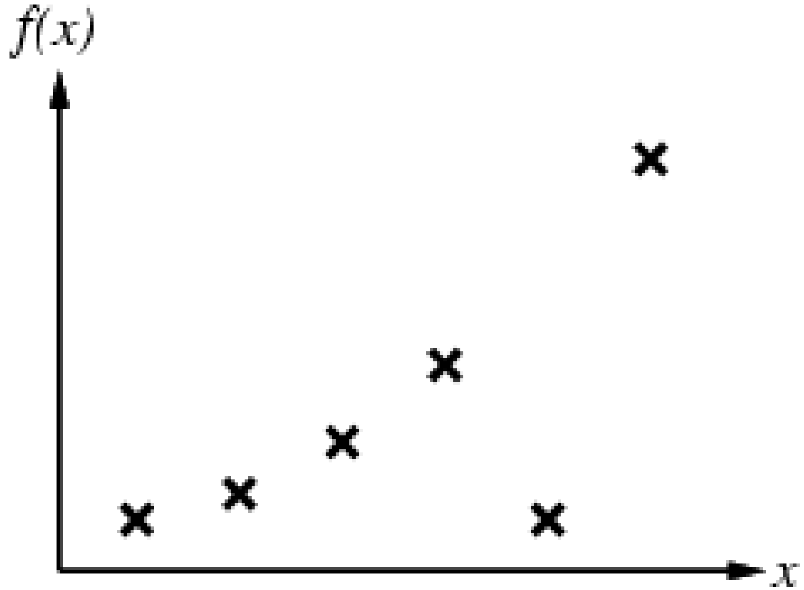
$$\theta_0 = 1.5 \quad \theta_1 = 0$$

$$\theta_0 = 0 \quad \theta_1 = 0.5$$

$$\theta_0 = 1 \quad \theta_1 = 0.5$$

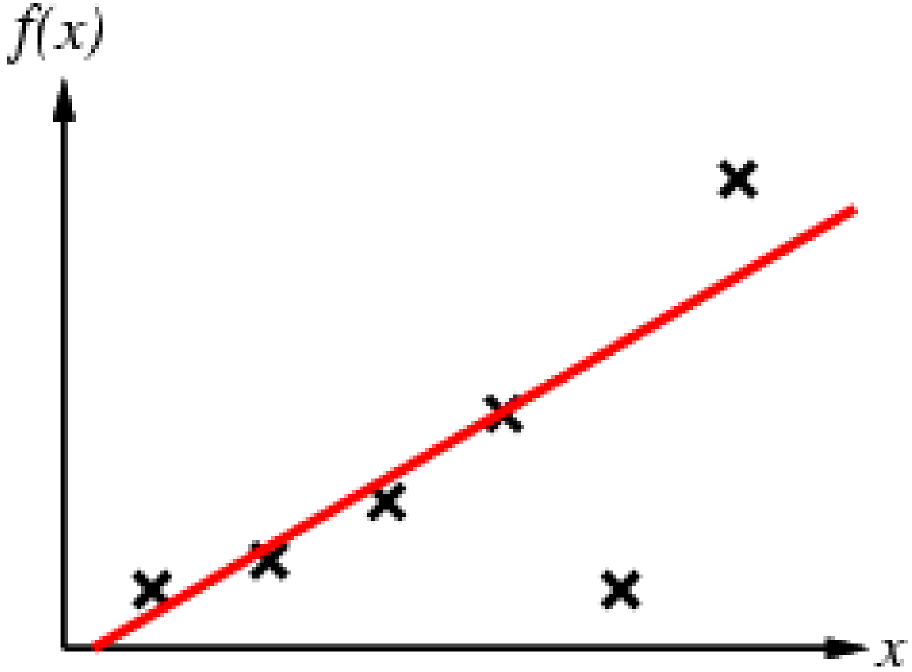
Supervised machine learning:

- Inductive learning method:
- Construct/adjust h to agree with f on training set
- (h is consistent if it agrees with f on all examples)
- E.g., curve fitting:



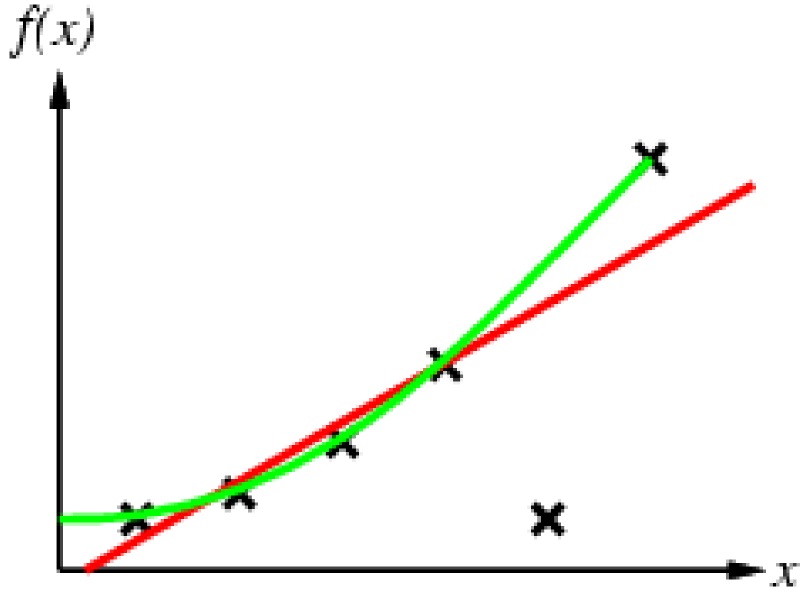
Supervised machine learning:

- Inductive learning method:
- Construct/adjust h to agree with f on training set
- (h is consistent if it agrees with f on all examples)
- E.g., curve fitting:



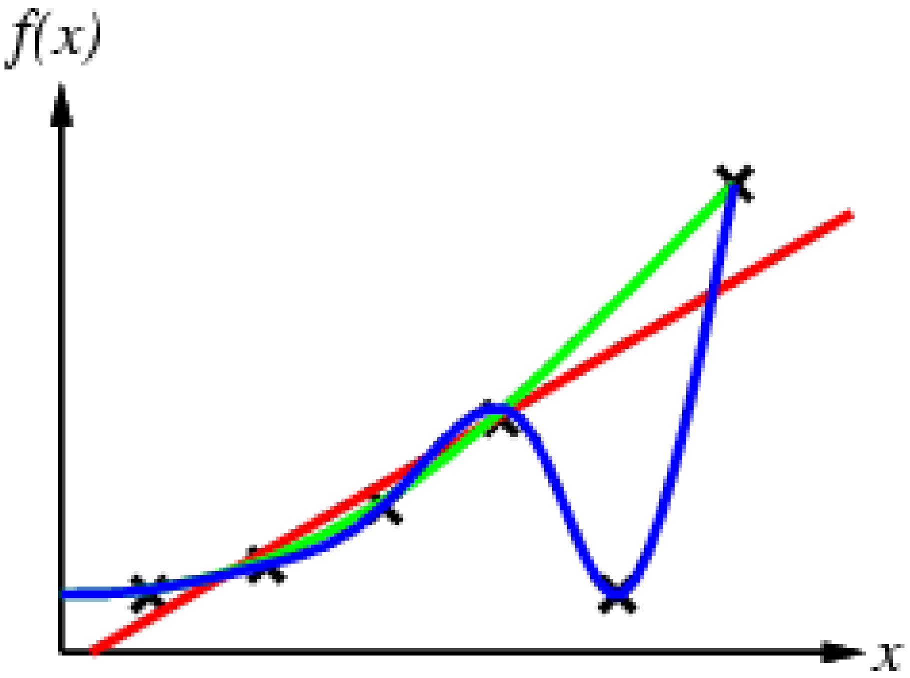
Supervised machine learning:

- Inductive learning method:
- Construct/adjust h to agree with f on training set
- (h is consistent if it agrees with f on all examples)
- E.g., curve fitting:



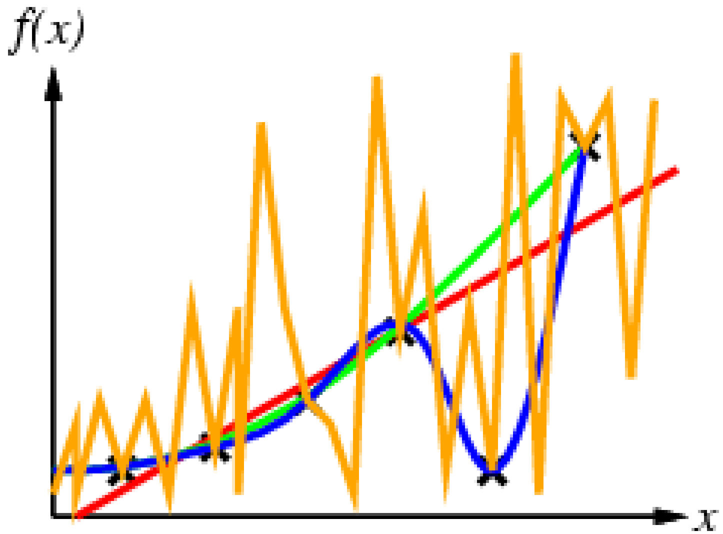
Supervised machine learning:

- Inductive learning method:
- Construct/adjust h to agree with f on training set
- (h is consistent if it agrees with f on all examples)
- E.g., curve fitting:



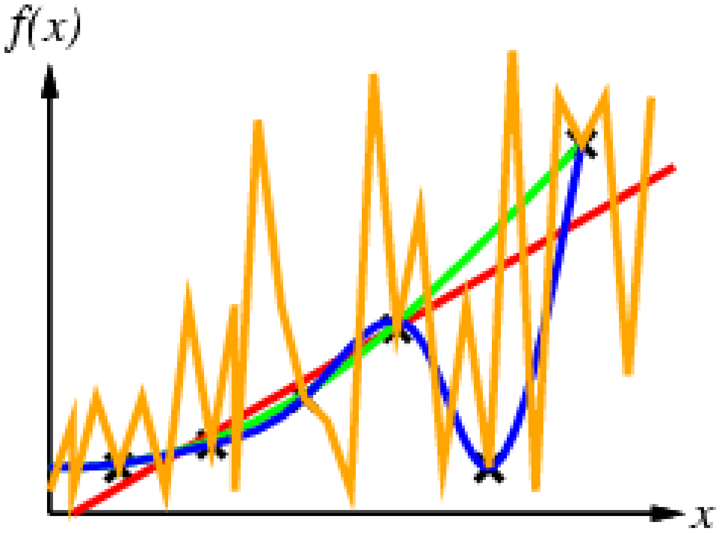
Supervised machine learning:

- Inductive learning method:
- Construct/adjust h to agree with f on training set
- (h is consistent if it agrees with f on all examples)
- E.g., curve fitting:



Supervised machine learning:

- Inductive learning method:
- Construct/adjust h to agree with f on training set
- (h is consistent if it agrees with f on all examples)
- E.g., curve fitting:



- Ockham's razor: prefer the simplest hypothesis consistent with data

Credit: Khola Naseem

Generalization:

- Hypotheses must generalize to correctly classify instances not in the training data.
- Simply memorizing training examples is a consistent hypothesis that does not generalize.
- Occam's razor:
  - Finding a simple hypothesis helps ensure generalization.

Supervised learning:

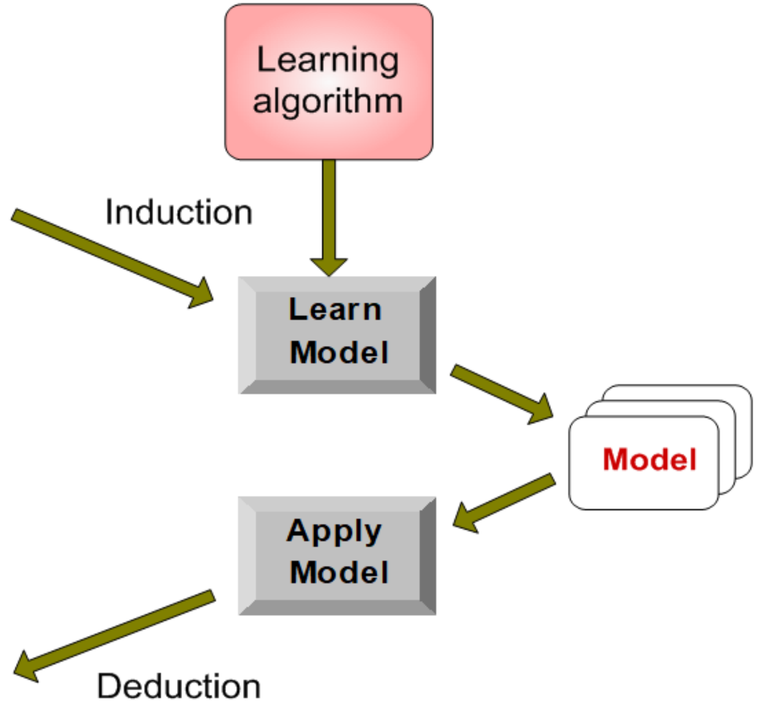
- the algorithm measures its accuracy through the loss function, adjusting until the error has been sufficiently minimized.
- Supervised learning can be separated into two types of problems:
  - Classification
  - Regression

Supervised learning:

- Supervised learning can be separated into two types of problems:
  - Classification

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	?

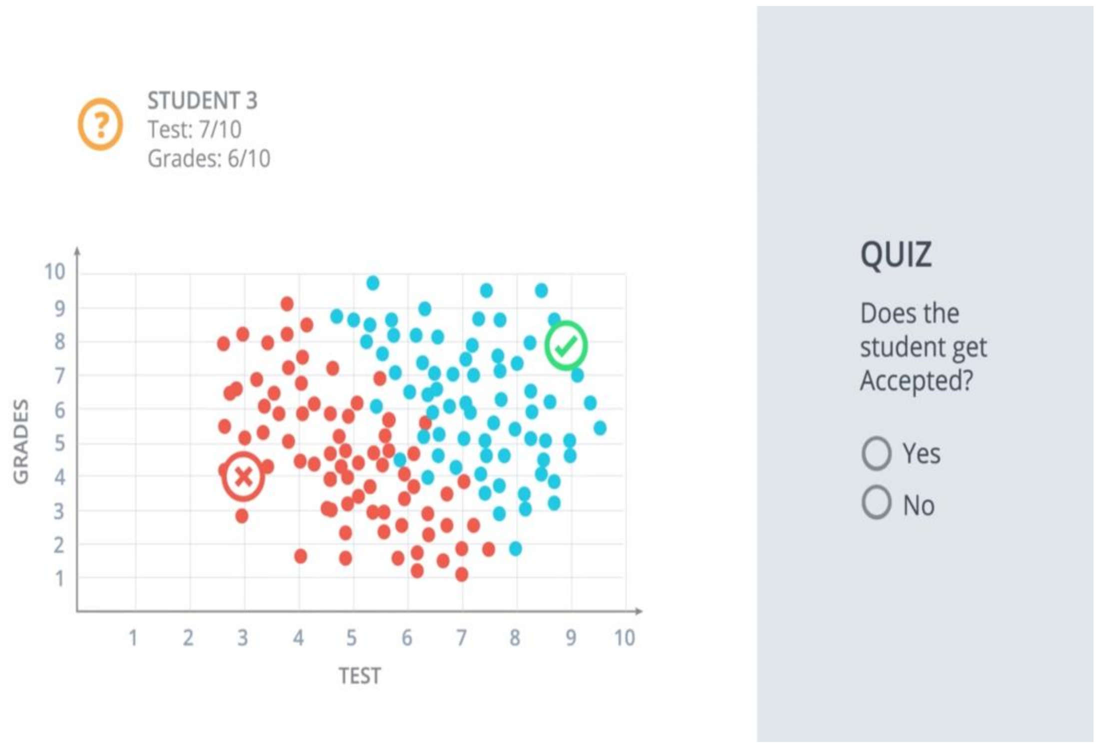
Test Set

21

Credit: Khola Naseem

Supervised learning:

- Supervised learning can be separated into two types of problems:
  - Classification



The data and the goal

Feature vector

Target Label Y

Day	*Sunny*	*Temp.*	*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

Credit: Khola Naseem

Classification

Movie review data example

From text to vectors

$$V = [v_1, v_2, v_3, \cdots, v_n, class]$$

review$_1$="great movie"

review$_2$="excellent film"

review$_3$="worst film ever"

review$_4$="sucks"

Features

Target Label Y

$$\begin{array}{l|ccccccc|c}
 & \text{ever} & \text{excellent} & \text{film} & \text{great} & \text{movie} & \text{sucks} & \text{worst} & \\
V_1 = [& 0, & 0, & 0, & 1, & 1, & 0, & 0, & +] \\
V_2 = [& 0, & 1, & 1, & 0, & 0, & 0, & 0, & +] \\
V_3 = [& 1, & 0, & 1, & 0, & 0, & 0, & 1, & -] \\
V_4 = [& 0, & 0, & 0, & 0, & 0, & 1, & 0, & -]
\end{array}$$

Credit: Khola Naseem

Supervised learning:

Regression:

- Regression tasks are different, as they expect the model to produce a numerical relationship between the input and output data.
- Examples of regression models include
  - predicting real estate prices based on zip code, or
  - predicting click rates in online ads in relation to time of day,
  - determining how much customers would be willing to pay for a certain product based on their age.

Supervised learning:

Regression:

Features

Target Label Y

Size ($feet^2$) x1	Number of bedrooms x2	Number of floors x3	Age of home (years) x4	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…

m

- Notation:
  - m = Number of training examples
  - x's = "input" variable / features
  - y's = "output" variable / "target" variable
  - (x, y) – one training example
  - (x(i) , y(i)) – ith training example

$$x^{(1)} = 2104$$
$$x^{(2)} = 1416$$
$$y^{(1)} = 460$$

Credit: Khola Naseem

Supervised learning:

In classification, we predict labels y (classes) for inputs x

Examples:

- OCR (input: images, classes: characters)
- Medical diagnosis (input: symptoms, classes: diseases)
- Automatic essay grader (input: document, classes: grades)
- Fraud detection (input: account activity, classes: fraud / no fraud)
- Recommended articles in a newspaper, recommended books
- DNA and protein sequence identification
- Categorization and identification of astronomical images
- Financial investments
- … many more

Credit: Khola Naseem

Supervised learning:

Pros:

- With the help of supervised learning, the model can predict the output on the basis of prior experiences.
- In supervised learning, we can have an exact idea about the classes of objects.
- Supervised learning model helps us to solve various real-world problems such as fraud detection, spam filtering, etc

Supervised learning:

Cons:

- Supervised learning cannot predict the correct output if the test data is different from the training dataset.
- Training required lots of computation times.
- In supervised learning, we need enough knowledge about the classes of object.

Supervised learning:

Navie Bayesian algorithm:

- A statistical classifier performs probabilistic prediction,
- No learning model, just need to extract probabilities
- Example
  - Let imagine we want to develop a Software for automatically classifying fruit types through SmartPhone camera
- i.e., predicts class membership probabilities, such as the probability that a given instance belongs to a particular class
- A simple Bayesian classifier, Naive Bayesian classifier, exhibits
  - high accuracy and speed when applied to large databases comparable to decision tree and selected neural network classifiers

Supervised learning:

Navie Bayesian algorithm:

- Naïve Bayes Classifier which is used for classification problem and it's supervised machine learning algorithm.
- Naive Bayes Classification works on concept of conditional probability.
- It would answer the question such as what is the probability that a given tuple of a data set belongs to a particular class label.

Supervised learning:

Navie Bayesian algorithm:

- **Probability:**
- Probability is the formal study of the laws of chance. Probability allows us to manage uncertainty.
- The sample space is the set of all outcomes. For example, for a dice we have 6 outcomes : {1,2,3,4,5,6}
- Probability allows us to measure many events. The events are subsets of the sample space. For example, for a dice we may consider the following events: Even: {2,4,6}; odd: {1,3,5}, greater than four {5,6}
- We assign probabilities to the events P(Even) = 3/6

Supervised learning:

Navie Bayesian algorithm:

- Bayesian classifiers use Bayes theorem:
- $P(C \mid F) = (\boldsymbol{P}(\boldsymbol{F}|\boldsymbol{C}) \times \boldsymbol{P}(\boldsymbol{C})) / \boldsymbol{P}(\boldsymbol{F})$
  - $\boldsymbol{P}(\boldsymbol{F}|\boldsymbol{C})$**:** Probability of generating instance F given class C,
    - We can imagine that being in class C, causes you to have feature F with some probability
  - **P(C):** probability of occurrence of class C,
    - This is just how frequent the class C, is in our dataset
  - **P(F):** probability of instance F occurring
    - This can actually be ignored, since it is the same for all classes

Supervised learning:

Navie Bayesian algorithm:

- Assume that we have two classes
  - c1 = male, and c2 = female, and only one feature F: name
- We have a person whose gender we do not know, say "drew" or F.
- Classifying drew as male or female is
  - equivalent to asking: is it more probable that drew is male or female, i.e. which is greater?
    - p(male | drew) or p(female | drew)
- What is the probability of being called "drew" given that you are a male?

$$p(\text{male} \mid drew) = \frac{p(drew \mid \text{male})\, p(\text{male})}{p(drew)}$$

What is the probability of being a male?

What is the probability of being named "*drew*"?

Drew Barrymore

Drew Carey

Credit: Khola Naseem

Supervised learning:

Navie Bayesian algorithm:

- This is Officer Drew. Is Officer Drew a Male or Female?
- Luckily, we have a small database with names and gender.
- We can use it to apply Bayes rule…

Officer Drew

$$P(C|F) = \frac{\boldsymbol{P(F|C) \times P(C)}}{\boldsymbol{P(F)}}$$

Drew	Male
Claudia	Female
Drew	Female
Drew	Female
Alberto	Male
Karin	Female
Nina	Female
Sergio	Male

Credit: Khola Naseem

Supervised learning:

Navie Bayesian algorithm:

- This is Officer Drew. Is Officer Drew a Male or Female?
- Luckily, we have a small database with names and gender.
- We can use it to apply Bayes rule…

Officer Drew

$$P(C|F) = \frac{P(F|C) \times P(C)}{P(F)}$$

Drew	Male
Claudia	Female
Drew	Female
Drew	Female
Alberto	Male
Karin	Female
Nina	Female
Sergio	Male

$$P(male|drew) = \frac{P(drew|m).P(m)}{P(drew)} = \frac{1/3 \times 3/8}{3/8} = \frac{1}{3}$$

$$\mathrm{P}(\mathrm{female}|\mathrm{drew}) = \frac{P(drew|f).P(f)}{P(drew)} = \frac{2/5 \times 5/8}{3/8} = \frac{2}{3}$$

Credit: Khola Naseem

Supervised learning:

Navie Bayesian algorithm:

- This is Officer Drew. Is Officer Drew a Male or Female?
- Luckily, we have a small database with names and gender.
- We can use it to apply Bayes rule…

Officer Drew

$$P(C|F) = \frac{P(F|C) \times P(C)}{P(F)}$$

$$P(male|drew) = \frac{P(drew|m).P(m)}{P(drew)} = \frac{1/3 \times 3/8}{3/8} = \frac{1}{3}$$

$$\mathrm{P(female|drew)} = \frac{P(drew|f).P(f)}{P(drew)} = \frac{2/5 \times 5/8}{3/8} = \frac{2}{3}$$

Officer Drew is more likely to be a Female.

Supervised learning:

Navie Bayesian algorithm:

- Example. Suppose customers described by attributes age and income
- F: a 35-year-old customer with an income of $40,000.
- C: the hypothesis that the customer will buy a computer.
- P(C|F): the probability that customer F will buy a computer given that we know the customer's age and income.
- P(C): the probability that any given customer will buy a computer, regardless of age and income
- P(F): the probability that a person from our set of customers is 35 years old and earns $40,000.
- P(F|C): the probability that a customer, X, is 35 years old and

earns $40,000, given that we know the customer will buy a computer

$$P(C|F) = \frac{P(F|C) \times P(C)}{P(F)}$$

Supervised learning:

Navie Bayesian algorithm:

- So far we have only considered Bayes Classification when we have one attribute (i.e. “name”). But we may have many features.
- How to calculate $P(C|F)=\boldsymbol{P(F|C)}\times\boldsymbol{P(C)}$
- P(C): prior probability P(C) : the probability that any given data sample is an apple, regardless of how the data sample looks
  - We need to calculate the probabilities for all fruit types
    - apple 2/8 = ...
    - banana 2/8 = ...
    - tomato 2/8 = ...
    - cherry 2/8 =...

shape	color	fruit
round	red	**apple**
round	green	**apple**
long	green	**banana**
long	yellow	**banana**
round	red	**cherry**
round	green	**tomato**
round	red	**tomato**
round	red	**cherry**

Credit: Khola Naseem

Supervised learning:

Navie Bayesian algorithm:

- So far we have only considered Bayes Classification when we have one attribute (i.e. “name”). But we may have many features.
- F = (red and round)
- C: Classes in our data set(apple, banana, cherry, tomato)
- P(F/C)=p(round,red | apple)
- P(F/C)=p(round,red | banana)
- P(F/C)=p(round,red | cherry)
- P(F/C)=p(round,red | tomato)

shape	color	fruit
round	red	**apple**
round	green	**apple**
long	green	**banana**
long	yellow	**banana**
round	red	**cherry**
round	green	**tomato**
round	red	**tomato**
round	red	**cherry**

- The probability that X is an apple given that X is round and red
  - P(round,red|apple)=p(round|apple)*p(red|apple)=2/2 * 1/2= 0.5

Credit: Khola Naseem

Supervised learning:

Navie Bayesian algorithm:

- So far we have only considered Bayes Classification when we have one attribute (i.e. “name”). But we may have many features.
- The probability that X is an apple given that X is round and red
  - P(round,red|apple)=p(round|apple)*p(red|apple)=2/2 * 1/2= 0.5
  - P(round,red|banana)=p(round|banana)*p(red|banana)=0/2 * 0/2= 0
  - P(round,red|tomato)=p(round|tomato)*p(red|tomato)=2/2 * 1/2= 0.5
  - P(round,red|cherry)=p(round|cherry)*p(red|cherry)=2/2 * 2/2=1

shape	color	fruit
round	red	**apple**
round	green	**apple**
long	green	**banana**
long	yellow	**banana**
round	red	**cherry**
round	green	**tomato**
round	red	**tomato**
round	red	**cherry**

Credit: Khola Naseem

Supervised learning:

Navie Bayesian algorithm: Example 3:

Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

- We need to classify the following new instance?

Outlook	Temperature	Humidity	Wind	Play
Sunny	Cool	High	True	?

Supervised learning:

Navie Bayesian algorithm: Example 3:

- First, we find the Probability for yes and no in the target attribute which is play.
- We first estimate prior probabilities
- P(C) or P(Class)
  - For Yes: Total 14 and 9 Yes

$$\text{Probability} = \frac{\text{Favorable}}{\text{Total}}$$

$$P(\text{yes}) = \frac{9}{14}$$

  - For No: Total 14 and 5 No

$$P(\text{no}) = \frac{5}{14}$$

Supervised learning:

Navie Bayesian algorithm: Example 3:

- Now I can also calculate the individual property for Outlook,
- In the first attribute which is outlook we have three categorical values which are (sunny, overcast, rainy) and our target is that we find each of the probability for Sunny, Overcast and Rainy based on target attribute which is play (yes, no).

Outlook	yes	no	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Supervised learning:

Navie Bayesian algorithm: Example 3:

- Temperature:
- In the second attribute which is Temperature we have three categorical values which are (Hot, mild, cool) and our target is that we find each of the probability for Hot, mild and cool based on target attribute which is play (yes, no).

Temperature	yes	no	P(yes)	P(no)
Hot	2	2	2/9	2/5
mild	4	2	4/9	2/5
cool	3	1	3/9	1/5
Total	9	5	100%	100%

Supervised learning:

Navie Bayesian algorithm: Example 3:

- Humidity:

Humidity	yes	no	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

- Wind:

Wind	yes	no	P(yes)	P(no)
False	6	2	6/9	2/5
True	3	3	3/9	3/5
Total	9	5	100%	100%

Supervised learning:

Navie Bayesian algorithm: Example 3:

- P(X) = P(Outlook = Sunny) *P(Temp = Cool)* P(Humidity = High) *P(Wind = Strong)

  P(X) = P(5/14)*(4/14)*(7/14)*(6/14)

  P(X) = 0.02186
- For Yes:
- P(x/yes)=p(sunny/yes)*p(cool/yes)*p(high/yes)*p(True/yes)
- P(yes)=9/14

$$P(\text{yes} / X) = \frac{P(\frac{X}{yes}).\ P(yes)}{P(X)}$$

$$= \frac{0.0053}{0.02186}$$

$$= 0.2424$$

Supervised learning:

Navie Bayesian algorithm: Example 3:

$$P(\text{ yes }/X) = \frac{P(\frac{X}{yes}).\ P(yes)}{P(X)}$$

$$= \frac{0.0053}{0.02186}$$

$$= 0.2424$$

- For No
- P(x/No)=p(sunny/No)*p(cool/No)*p(high/No)*p(True/No)
- P(No)=5/14

$$P(\text{ No }/X) = \frac{P(\frac{X}{no}).\ P(no)}{P(X)}$$

$$= \frac{0.0206}{0.02186}$$

$$= 0.9421$$

- 0.9421 > 0.2424
- So the probability for no is highest as compare to yes.

Supervised learning:

Navie Bayesian algorithm: Example 3:

- In the End
  - 0.9421 > 0.2424
  - So the probability for no is highest as compare to yes.
  - Our prediction:

Outlook	Temperature	Humidity	Wind	Play
Sunny	Cool	High	True	No

Supervised learning:

Navie Bayesian algorithm: Example 4:

- Suppose that data items consists of the attributes x, y, z
- x, y, and z are each integers in the range 1 to 4.
- The available classifications are A, B, and C.

x	y	z	Classification
2	3	2	A
4	1	4	B
1	3	2	A
2	4	3	A
4	2	4	B
2	1	3	C
1	2	4	A
2	3	3	B
2	2	4	A
3	3	3	C
3	2	1	A
1	2	1	B
2	1	4	A
4	3	4	C
2	2	4	A

Supervised learning:

Navie Bayesian algorithm:

- Example 4:
- 15 pieces of training data, each of which has been classified.
- Eight of the training data are classified as A, four as B, and three as C.
- Suppose that we are presented with a new piece of data
  - $x = 2, y = 3, z = 4$

Supervised learning:

- **Why Naive bayes is naive:**
- Naive Bayes (NB) is 'naive' because it makes the assumption that features of a measurement are independent of each other.
- In other words the effect of an attribute value on a given class is independent of the values of the other attributes.
- This assumption is made to reduce computational costs, and therefore it is considered naïve.
- This is naive because it is (almost) never true
  - Stock Market Prediction: In financial markets, various economic indicators and stock prices are interrelated. The independence assumption is unlikely to hold, as market movements are influenced by multiple factors

Supervised learning:

- **Naive bayes :**
- Pros
  - It’s relatively simple to understand and build
  - It’s easily trained, even with a small dataset
  - It’s fast!
  - It’s not sensitive to irrelevant features

Supervised learning:

- **Naive bayes :**
- Pros
  - It assumes every feature is independent, which isn’t always the case