Introduction

Definition of machine learning

- Machine learning is programming computers to optimize a performance criterion using example data or past experience.
- We have a model defined up to some parameters, and learning is the execution of a computer program to optimize the parameters of the model using the training data or past experience.
- The model may be predictive to make predictions in the future, or descriptive to gain knowledge from data, or both.
- The field of study known as machine learning is concerned with the question of how to construct computer programs that automatically improve with experience.

Definition of learning

Definition

- A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks T, as measured by P, improves with experience E.

Examples

i) Handwriting recognition learning problem

- Task T: Recognising and classifying handwritten words within images
- Performance P: Percent of words correctly classified
- Training experience E: A dataset of handwritten words with given classifications

ii) A robot driving learning problem

- Task T: Driving on highways using vision sensors
- Performance measure P: Average distance traveled before an error
- Training experience: A sequence of images and steering commands recorded while observing a human driver

iii) A chess learning problem

- Task T: Playing chess
- Performance measure P: Percent of games won against opponents
- Training experience E: Playing practice games against itself

A computer program which learns from experience is called a machine learning program or simply a learning program. Such a program is sometimes also referred to as a learner.

How machines learn

- Basic components of learning process
- The learning process, whether by a human or a machine, can be divided into four components, namely, data storage, abstraction, generalization and evaluation. Figure 1.1 illustrates the various components and the steps involved in the learning process.

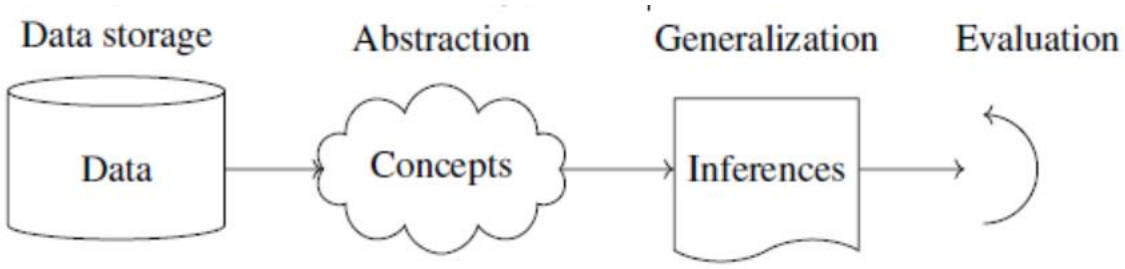


Figure 1.1: Components of learning process

1.Data storage

- Facilities for storing and retrieving huge amounts of data are an important component of the learning process. Humans and computers alike utilize data storage as a foundation for advanced reasoning.
- In a human being, the data is stored in the brain and data is retrieved using electrochemical signals.
- Computers use hard disk drives, flash memory, random access memory and similar devices to store data and use cables and other technology to retrieve data.

2. Abstraction

- The second component of the learning process is known as abstraction.
- Abstraction is the process of extracting knowledge about stored data. This involves creating general concepts about the data as a whole. The creation of knowledge involves application of known models and creation of new models.
- The process of fitting a model to a dataset is known as training.
- When the model has been trained, the data is transformed into an abstract form that summarizes the original information.

3. Generalization

- The third component of the learning process is known as generalization.
- The term generalization describes the process of turning the knowledge about stored data into a form that can be utilized for future action.
- These actions are to be carried out on tasks that are similar, but not identical, to those what have been seen before.
- In generalization, the goal is to discover those properties of the data that will be most relevant to future tasks.

4. **Evaluation**

- Evaluation is the last component of the learning process.
- It is the process of giving feedback to the user to measure the utility of the learned knowledge.
- This feedback is then utilised to effect improvements in the whole learning process.

Applications of machine learning

- Application of machine learning methods to large databases is called data mining.
- In data mining, a large volume of data is processed to construct a simple model with valuable use, for example, having high predictive accuracy.

The following is a list of some of the typical applications of machine learning.

- In retail business, machine learning is used to study consumer behaviour.
- In finance, banks analyze their past data to build models to use in credit applications, fraud detection, and the stock market.
- In manufacturing, learning models are used for optimization, control, and troubleshooting.
- In medicine, learning programs are used for medical diagnosis.
- In telecommunications, call patterns are analyzed for network optimization and maximizing the quality of service.
- In artificial intelligence, it is used to teach a system to learn and adapt to changes so that the system designer need not foresee and provide solutions for all possible situations.

Understanding data

Unit of observation

- It refers to the entity or object that is being measured or observed in a study.
- By a unit of observation we mean the smallest entity with measured properties of interest for a study.

Examples

- A person, an object or a thing
- A time point
- A geographic region
- A measurement
- Sometimes, units of observation are combined to form units such as person-years.

Examples and features

- Datasets that store the units of observation and their properties can be imagined as collections of data consisting of the following:

Examples

- An “example” is an instance of the unit of observation for which properties have been recorded.
- An “example” is also referred to as an “instance”, or “case” or “record.” (It may be noted that the word “example” has been used here in a technical sense.)

Features

- A “feature” is a recorded property or a characteristic of examples. It is also referred to as “attribute”, or “variable” or “feature.”

Examples for “examples” and “features”

Cancer detection

- Consider the problem of developing an algorithm for detecting cancer. In this study we note the following.

(a) The units of observation are the patients.

(b) The examples are members of a sample of cancer patients.

(c) The following attributes of the patients may be chosen as the features:

- gender
- age
- blood pressure
- the findings of the pathology report after a biopsy

Pet selection

- Suppose we want to predict the type of pet a person will choose.

(a) The units are the persons.

(b) The examples are members of a sample of persons who own pets.

(c) The features might include age, home region, family income, etc. of persons who own pets

features

year	model	price	mileage	color	transmission
2011	SEL	21992	7413	Yellow	AUTO
2011	SEL	20995	10926	Gray	AUTO
2011	SEL	19995	7351	Silver	AUTO
2011	SEL	17809	11613	Gray	AUTO
2012	SE	17500	8367	White	MANUAL
2010	SEL	17495	25125	Silver	AUTO
2011	SEL	17000	27393	Blue	AUTO
2010	SEL	16995	21026	Silver	AUTO
2011	SES	16995	32655	Silver	AUTO

examples

Figure 1.2: Example for "examples" and "features" collected in a matrix format (data relates to automobiles and their features)

General classes of machine learning problems

- Learning associations

Association rule learning

- Association rule learning is a machine learning method for discovering interesting relations, called “association rules”, between variables in large databases using some measures of “interestingness”.

Example

- Consider a supermarket chain. The management of the chain is interested in knowing whether there are any patterns in the purchases of products by customers like the following:
- “If a customer buys onions and potatoes together, then he/she is likely to also buy hamburger.”

- From the standpoint of customer behavior, this defines an association between the set of products {onion, potato} and the set {burger}.
- This association is represented in the form of a rule as follows:

  $\{onion, potato\} \Rightarrow \{burger\}$

- The measure of how likely a customer, who has bought onion and potato, to buy burger also is given by the conditional probability

  $P(\{onion, potato\}|\{burger\})$.

- If this conditional probability is 0.8, then the rule may be stated more precisely as follows:

  “80% of customers who buy onion and potato also buy burger.”

How association rules are made use of

- Consider an association rule of the form

  $X \Rightarrow Y,$

- that is, if people buy X then they are also likely to buy Y .
- Suppose there is a customer who buys X and does not buy Y . Then that customer is a potential Y customer.
- Once we find such customers, we can target them for cross-selling. A knowledge of such rules can be used for promotional pricing or product placements.

General case

- In finding an association rule $X \Rightarrow Y$, we are interested in learning a conditional probability of the form $P(Y \mid X)$ where Y is the product the customer may buy and X is the product or the set of products the customer has already purchased.
- If we may want to make a distinction among customers, we may estimate $P(Y \mid X, D)$ where D is a set of customer attributes, like gender, age, marital status, and so on, assuming that we have access to this information.

Algorithms

- There are several algorithms for generating association rules. Some of the well-known algorithms are listed below:

a) Apriori algorithm

b) Eclat algorithm

c) FP-Growth Algorithm (FP stands for Frequency Pattern)

Classification

Definition

- In machine learning, classification is the problem of identifying to which of a set of categories a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known.

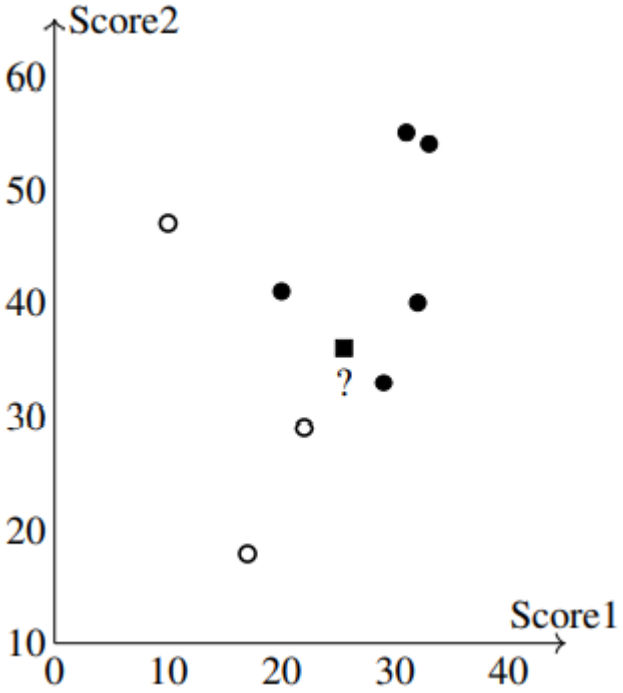
Example

- Consider the following data:

Score1	29	22	10	31	17	33	32	20
Score2	43	29	47	55	18	54	40	41
Result	Pass	Fail	Fail	Pass	Fail	Pass	Pass	Pass

Table 1.1: Example data for a classification problem

- Data in Table is the training set of data. There are two attributes "Score1" and "Score2". The class label is called "Result".
- The class label has two possible values "Pass" and "Fail". The data can be divided into two categories or classes: The set of data for which the class label is "Pass" and the set of data for which the class label is "Fail".
- Let us assume that we have no knowledge about the data other than what is given in the table.
- Now, the problem can be posed as follows:
- If we have some new data, say "Score1 = 25" and "Score2 = 36", what value should be assigned to "Result" corresponding to the new data; in other words, to which of the two categories or classes the new observation should be assigned?
- See Figure for a graphical representation of the problem.



Graphical representation of data in Table. Solid dots represent data in "Pass" class and hollow dots data in "Fail" class. The class label of the square dot is to be determined.

- To answer this question, using the given data alone we need to find the rule, or the formula, or the method that has been used in assigning the values to the class label "Result".
- The problem of finding this rule or formula or the method is the classification problem.
- In general, even the general form of the rule or function or method will not be known.
- So several different rules, etc. may have to be tested to obtain the correct rule or function or method.

Real life examples

- Optical character recognition: Optical character recognition problem, which is the problem of recognizing character codes from their images, is an example of classification problem.
- Face recognition: In the case of face recognition, the input is an image, the classes are people to be recognized, and the learning program should learn to associate the face images to identities changes in the image.
- Speech recognition: In speech recognition, the input is acoustic and the classes are words that can be uttered.
- Medical diagnosis: In medical diagnosis, the inputs are the relevant information we have about the patient and the classes are the illnesses.
- Knowledge extraction: Classification rules can also be used for knowledge extraction. The rule is a simple model that explains the data, and looking at this model we have an explanation about the process underlying the data.

Discriminant

- A discriminant of a classification problem is a rule or a function that is used to assign labels to new observations.
- Examples i) Consider the data given in Table and the associated classification problem.
- We may consider the following rules for the classification of the new data:
- IF Score1 + Score2 $\geq$ 60, THEN "Pass" ELSE "Fail".
- IF Score1 $\geq$ 20 AND Score2 $\geq$ 40 THEN "Pass" ELSE "Fail".
- Or, we may consider the following rules with unspecified values for M, m1, m2 and then by some method estimate their values.
- IF Score1 + Score2 $\geq$ M, THEN "Pass" ELSE "Fail". IF Score1 $\geq$ m1 AND Score2 $\geq$ m2 THEN "Pass" ELSE "Fail".

Algorithms

- There are several machine learning algorithms for classification. The following are some of the well-known algorithms.
- Logistic regression
- Naive Bayes algorithm
- k-NN algorithm
- Decision tree algorithm
- Support vector machine algorithm
- Random forest algorithm

Remarks

- A classification problem requires the examples that can be classified into one of two or more classes.
- A classification can have real-valued or discrete input variables.
- A problem with two classes is often called a two-class or binary classification problem.
- A problem with more than two classes is often called a multi-class classification problem.
- A problem where an example is assigned multiple classes is called a multi-label classification problem.

Regression

- In machine learning, a regression problem is the problem of predicting the value of a numeric variable based on observed values of the variable.
- The value of the output variable may be a number, such as an integer or a floating point value. These are often quantities, such as amounts and sizes.
- The input variables may be discrete or real-valued.

- Consider the data on car prices given in Table

Price (US$)	Age (years)	Distance (KM)	Weight (pounds)
13500	23	46986	1165
13750	23	72937	1165
13950	24	41711	1165
14950	26	48000	1165
13750	30	38500	1170
12950	32	61000	1170
16900	27	94612	1245
18600	30	75889	1245
21500	27	19700	1185
12950	23	71138	1105

Table 1.2: Prices of used cars: example data for regression

- Suppose we are required to estimate the price of a car aged 25 years with distance 53240 KM and weight 1200 pounds. This is an example of a regression problem because we have to predict the value of the numeric variable “Price”.

General approach

- Let x denote the set of input variables and y the output variable.
- In machine learning, the general approach to regression is to assume a model, that is, some mathematical relation between x and y, involving some parameters say, θ, in the following form:

$$\mathbf{y = f(x, \theta)}$$

- The function $f(x, \theta)$ is called the regression function.
- The machine learning algorithm optimizes the parameters in the set θ such that the approximation error is minimized; that is, the estimates of the values of the dependent variable y are as close as possible to the correct values given in the training set

Example

- For example, if the input variables are “Age”, “Distance” and “Weight” and the output variable is “Price”, the model may be

$$y = f(x,\theta)$$

$$\text{Price} = a0 + a1 \times (\text{Age}) + a2 \times (\text{Distance}) + a3 \times (\text{Weight})$$

- where $x = (\text{Age, Distance, Weight})$ denotes the set of input variables and $\theta = (a0; a1; a2; a3)$ denotes the set of parameters of the model.

Different regression models

- There are various types of regression techniques available to make predictions.
- These techniques mostly differ in three aspects, namely, the number and type of independent variables, the type of dependent variables and the shape of regression line.

Some of these are listed below.

- Simple linear regression: There is only one continuous independent variable x and the assumed relation between the independent variable and the dependent variable y is

$$y = a + bx$$

- Multivariate linear regression: There are more than one independent variable, say x1……., xn, and the assumed relation between the independent variables and the dependent variable is

$$y = a_0 + a_1 x_1 + \ldots\ldots + a_n x_n$$

- Polynomial regression: There is only one continuous independent variable x and the assumed model is

$$y = a_0 + a1x + \ldots\ldots + a_n x^n:$$

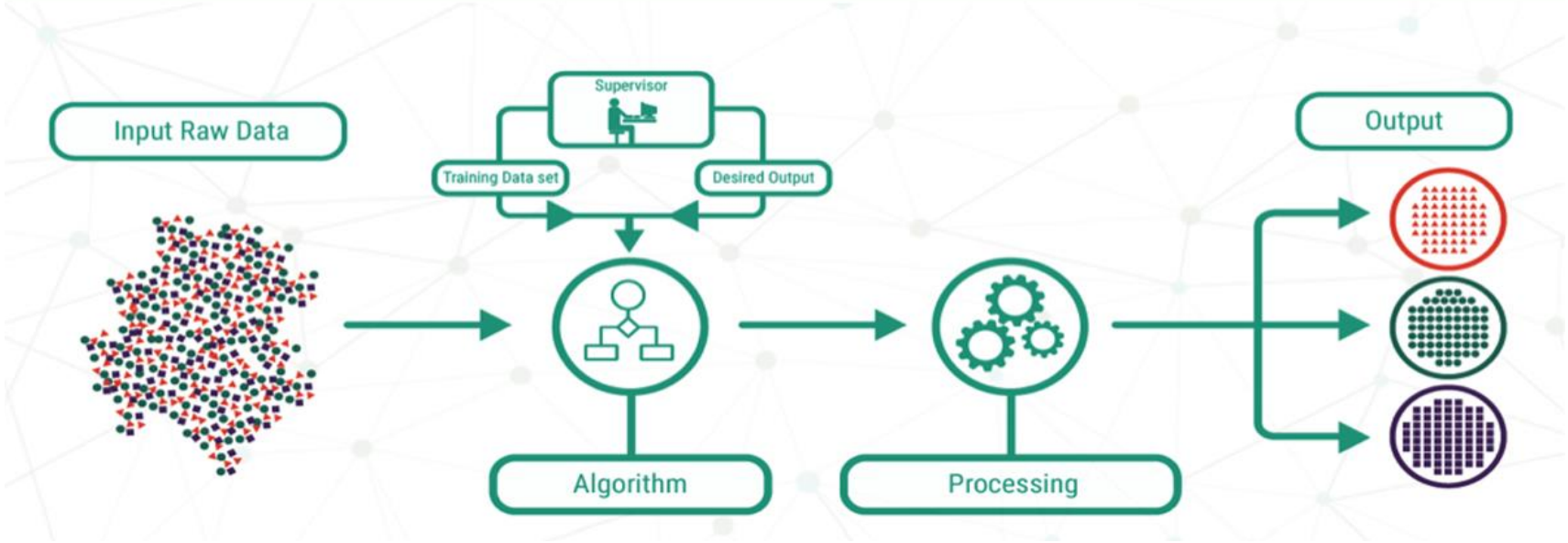
- Logistic regression: The dependent variable is binary, that is, a variable which takes only the values 0 and 1. The assumed model involves certain probability distributions.

Different types of 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.
- In supervised learning, each example in the training set is a pair consisting of an input object (typically a vector) and an output value.
- A supervised learning algorithm analyzes the training data and produces a function, which can be used for mapping new examples.
- In the optimal case, the function will correctly determine the class labels for unseen instances.
- Both classification and regression problems are supervised learning problems.

Supervised Learning



- A “supervised learning” is so called because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process.

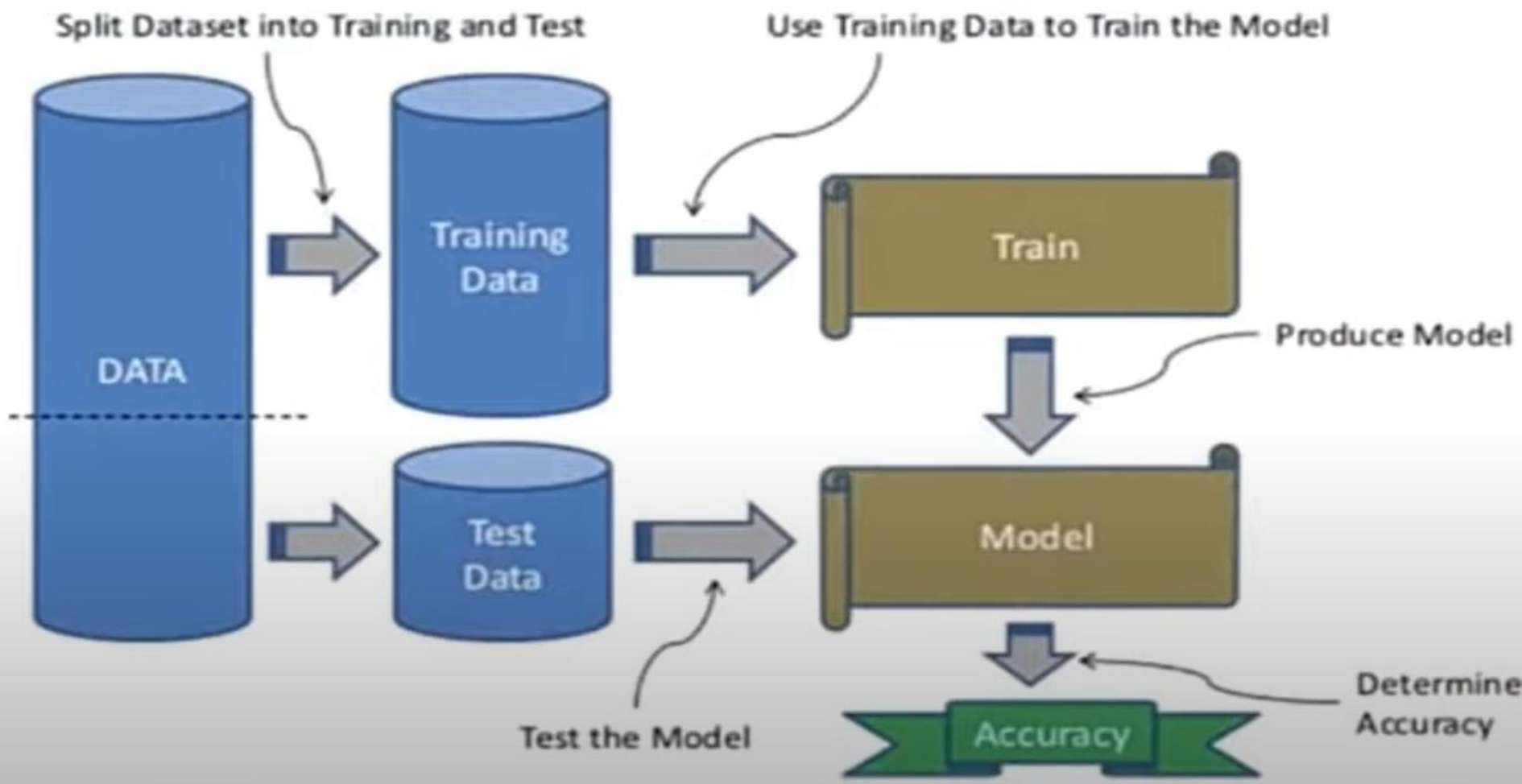
Example

- 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	healthy
F	34	sick
M	21	healthy

- Based on this data, when a new patient enters the clinic, how can one predict whether he/she is healthy or sick?

Supervised 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, a classification or categorization is not included in the observations.
- There are no output values and so there is no estimation of functions. Since the examples given to the learner are unlabeled, the accuracy of the structure that is output by the algorithm cannot be evaluated.
- The most common unsupervised learning method is cluster analysis, which is used for exploratory data analysis to find hidden patterns or grouping in data.

Difference between Regression and Classification

Regression Algorithm	Classification Algorithm
In Regression, the output variable must be of continuous nature or real value.	In Classification, the output variable must be a discrete value.
The task of the regression algorithm is to map the input value (x) with the continuous output variable(y).	The task of the classification algorithm is to map the input value(x) with the discrete output variable(y).
Regression Algorithms are used with continuous data.	Classification Algorithms are used with discrete data.
In Regression, we try to find the best fit line, which can predict the output more accurately.	In Classification, we try to find the decision boundary, which can divide the dataset into different classes.
Regression algorithms can be used to solve the regression problems such as Weather Prediction, House price prediction, etc.	Classification Algorithms can be used to solve classification problems such as Identification of spam emails, Speech Recognition, Identification of cancer cells, etc.
The regression Algorithm can be further divided into Linear and Non-linear Regression.	The Classification algorithms can be divided into Binary Classifier and Multi-class Classifier.

Example

- 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

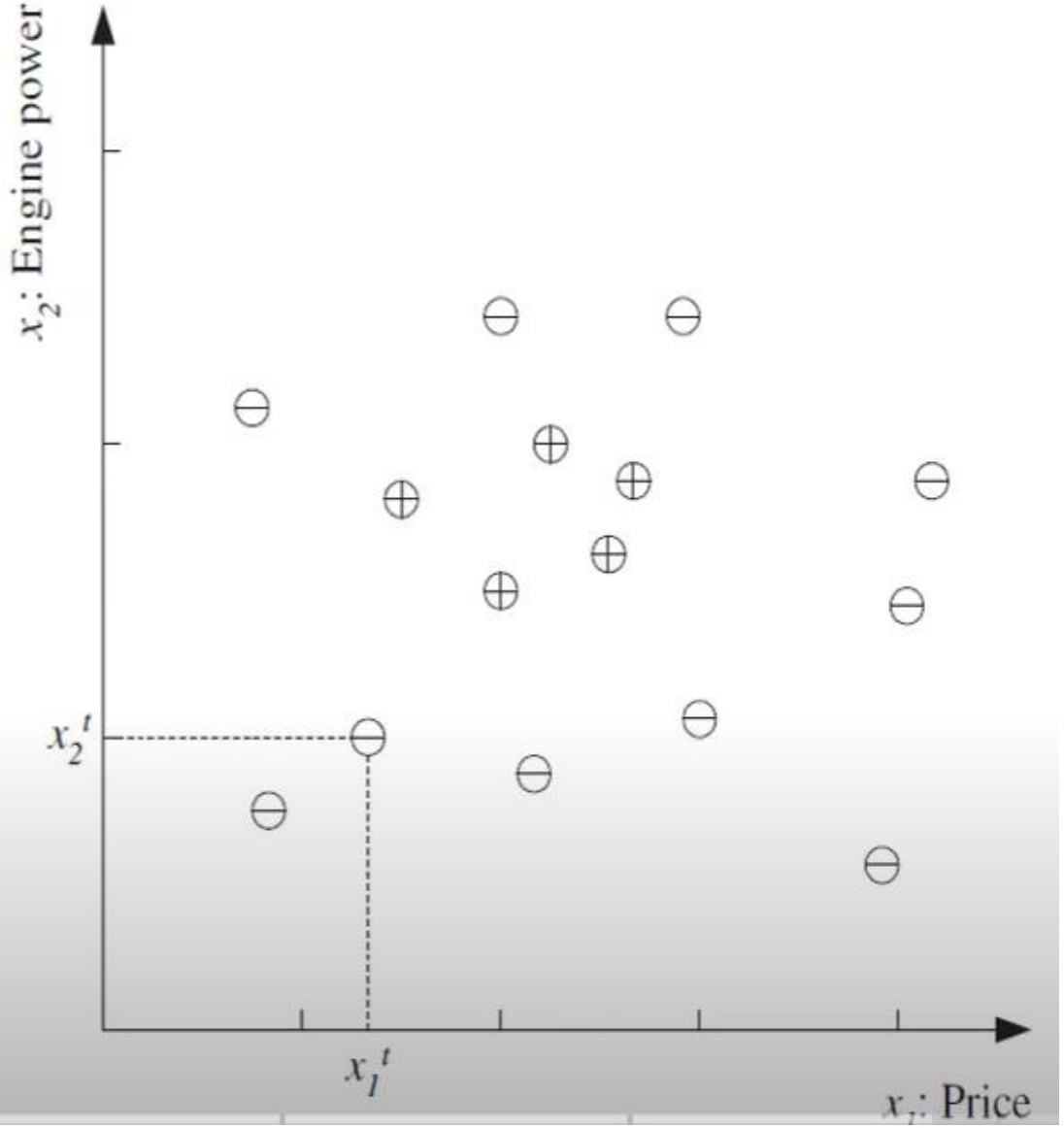
- Based on this data, can we infer anything regarding the patients entering the clinic?

Supervised Learning: Learning a Class from Examples…

- Set of cars "Class-C : Family of Cars"
- A group of people look at the cars and label them; family car or not with two attributes the price and engine power.
- The cars that they believe are family cars are positive examples, and the other cars are negative examples.
- People ignore other attributes such as seating capacity and color and consider those of irrelevant.

Training set-Family Car

- The data point corresponds to one sample car
- Coordinates: price and engine power
- '+': positive example of class (a family car),
- '–': negative example (not a family car)



Variables 'x' and 'r'

- Price as the first input attribute x1 (e.g., in Rupees)
- Engine power as the second attribute x2 (e.g., engine volume in cubic centi-meters).
- Label denotes its type

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$r = \begin{cases} 1 & \text{if } \mathbf{x} \text{ is a positive example} \\ 0 & \text{if } \mathbf{x} \text{ is a negative example} \end{cases}$$

- Each car is represented by such an ordered pair (x, r) and the training set contains N such examples

$$\mathcal{X} = \{\mathbf{x}^t, r^t\}_{t=1}^{N}$$

- where t indexes the training set.

Input representation

- The general classification problem is concerned with assigning a class label to an unknown instance from instances of known assignments of labels.
- In a real world problem, a given situation or an object will have large number of features which may contribute to the assignment of the labels.
- But in practice, not all these features may be equally relevant or important. Only those which are significant need be considered as inputs for assigning the class labels.
- These features are referred to as the "input features" for the problem.
- They are also said to constitute an "input representation" for the problem.

Example

- Consider the problem of assigning the label “family car” or “not family car” to cars.
- Let us assume that the features that separate a family car from other cars are the price and engine power.
- These attributes or features constitute the input representation for the problem.
- While deciding on this input representation, we are ignoring various other attributes like seating capacity or color as irrelevant.

Hypothesis space

- In the following discussions we consider only "binary classification" problems; that is, classification problems with only two class labels.
- The class labels are usually taken as "1" and "0".
- The label "1" may indicate "True", or "Yes", or "Pass", or any such label. The label "0" may indicate "False", or "No" or "Fail", or any such label.
- The examples with class labels 1 are called "positive examples" and examples with labels "0" are called "negative examples".

Definition

Hypothesis

- In a binary classification problem, a hypothesis is a statement or a proposition purporting to explain a given set of facts or observations.

Hypothesis space

- The hypothesis space for a binary classification problem is a set of hypotheses for the problem that might possibly be returned by it.

Consistency and satisfying

- Let x be an example in a binary classification problem and let c(x) denote the class label assigned to x (c(x) is 1 or 0).
- Let D be a set of training examples for the problem. Let h be a hypothesis for the problem and h(x) be the class label assigned to x by the hypothesis h.

(a) We say that the hypothesis h is consistent with the set of training examples D if $h(x) = c(x)$ for all $x > D$.

(b) We say that an example x satisfies the hypothesis h if $h(x) = 1$.

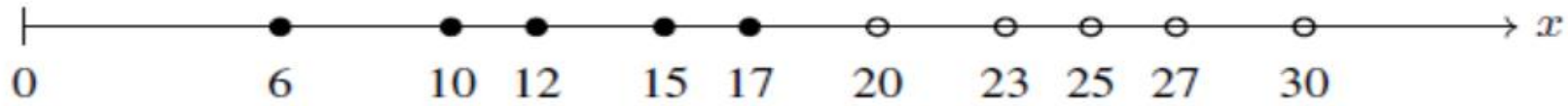
Examples

- 1. Consider the set of observations of a variable x with the associated class labels given in Table

x	27	15	23	20	25	17	12	30	6	10
Class	1	0	1	1	1	0	0	1	0	0

Table 2.1: Sample data to illustrate the concept of hypotheses

Figure 2.1 shows the data plotted on the x-axis.



- Data in Table with hollow dots representing positive examples and solid dots representing negative examples

- Looking at Figure 2.1, it appears that the class labeling has been done based on the following rule.

h' : IF x ≥ 20 THEN "1" ELSE "0".

Note that h' is consistent with the training examples in Table 2.1. For example, we have:

$$h'(27) = 1, \quad c(27) = 1, \quad h'(27) = c(27)$$
$$h'(15) = 0, \quad c(15) = 0, \quad h'(15) = c(15)$$

Note also that, for $x = 5$ and $x = 28$ (not in training data),

$$h'(5) = 0, \quad h'(28) = 1.$$

The hypothesis h' explains the data. The following proposition also explains the data:

$$h'' \quad : \quad \text{IF } x \geq 19 \text{ THEN "0" ELSE "1".}$$

- It is not enough that the hypothesis explains the given data; it must also predict correctly the class label of future observations.
- So we consider a set of such hypotheses and choose the “best” one.
- The set of hypotheses can be defined using a parameter, say m, as given below:

  h_m : IF x ≥ m THEN “1” ELSE ”0”.

- The set of all hypotheses obtained by assigning different values to m constitutes the hypothesis space H; that is,

  H = $\{h_m$: m is a real number$\}$:

- For the same data, we can have different hypothesis spaces. For example, for the data in Table, we may also consider the hypothesis space defined by the following proposition:

  h'_m : IF x ≤ m THEN “0” ELSE “1”.

- Consider the problem of assigning the label “family car” or “not family car” to cars.
- For convenience, we shall replace the label “family car” by “1” and “not family car” by “0”.
- Suppose we choose the features “price (’000 $)” and “power (hp)” as the input representation for the problem.
- Further, suppose that there is some reason to believe that for a car to be a family car, its price and power should be in certain ranges.
- This supposition can be formulated in the form of the following proposition:

$$\text{IF } (p_1 < \text{price} < p_2) \text{ AND } (e_1 < \text{power} < e_2) \text{ THEN “1” ELSE ”0”}$$

for suitable values of p1, p2, e1 and e2

- Since a solution to the problem is a proposition of the form Eq.(2.5) with specific values for p1, p2, e1 and e2.
- The hypothesis space for the problem is the set of all such propositions obtained by assigning all possible values for p1, p2, e1 and e2.

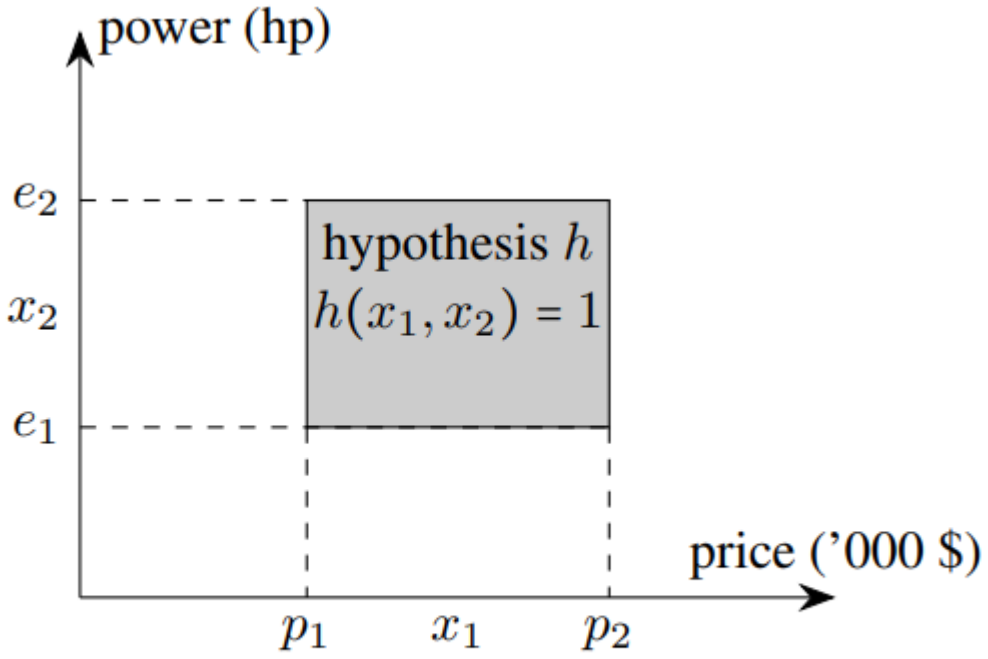


Figure 2.2: An example hypothesis defined by Eq.

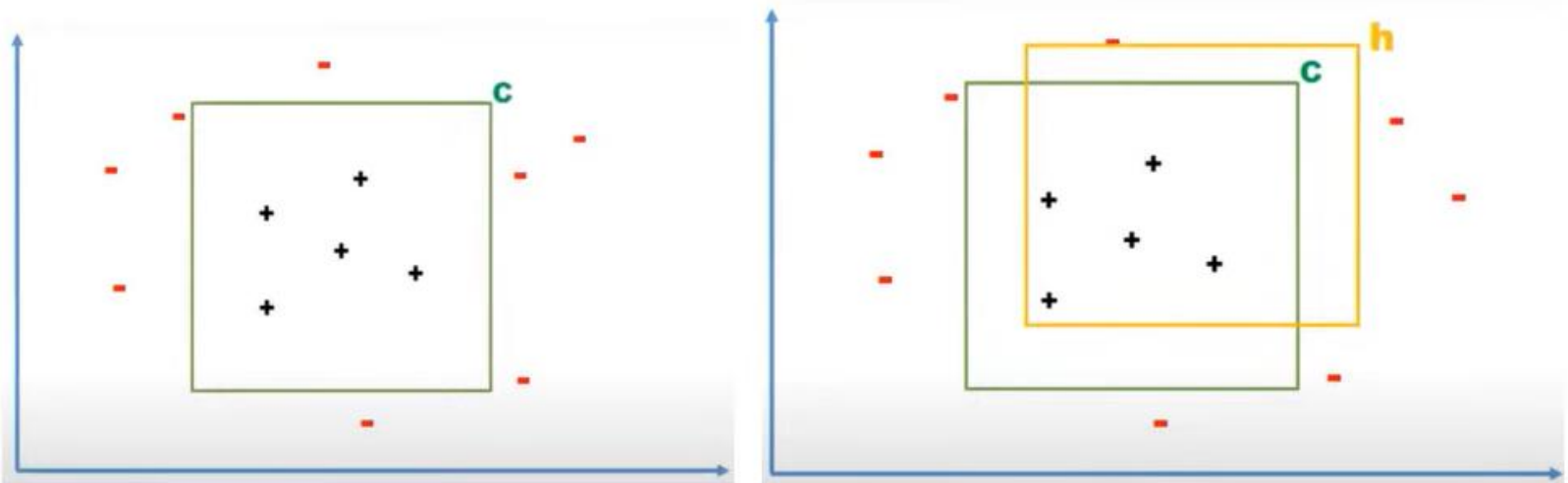
- It is interesting to observe that the set of points in the power–price plane which satisfies the condition

$$(p_1 < \text{price} < p_2) \text{ AND } (e_1 < \text{power} < e_2)$$

defines a rectangular region (minus the boundary) in the price–power space as shown in Figure.

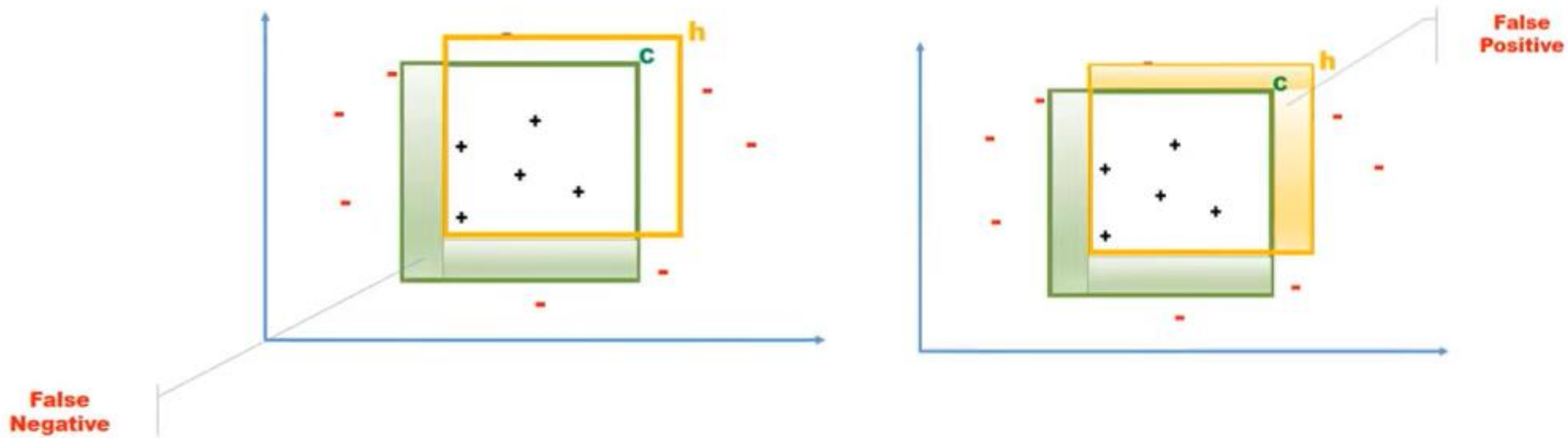
- The sides of this rectangular region are parallel to the coordinate axes. Such a rectangle is called an axis-aligned rectangle.
- If h is the hypothesis and (x1, x2) is any point in the price–power plane, then h(x1, x2) = 1 if and only if (x1, x2) is within the rectangular region.

- In real life we do not know C(x), so we cannot evaluate how well h(x) matches C(x).
- C – Target function
- Instances within rectangle represents family cars and outside are not family cars
- Hypothesis h – closely approximate c, and there may be error region.



False Positive and False Negative

- C is the actual class and h is our induced hypothesis.
- The point where C is 1 but h is 0 is a false negative, and
- the point where C is 0 but h is 1 is a false positive.
- true positives and true negatives—are correctly classified.



An hypothesis h is **consistent** with a set of training examples D iff $h(x) = c(x)$ for each example in D

$$Consistent(h, D) \equiv (\forall \langle x, c(x) \rangle \in D)\ h(x) = c(x))$$

Example	Citations	Size	InLibrary	Price	Editions	Buy
1	Some	Small	No	Affordable	One	No
2	Many	Big	No	Expensive	Many	Yes

h1 = (?, ?, No, ?, Many) – Consistent

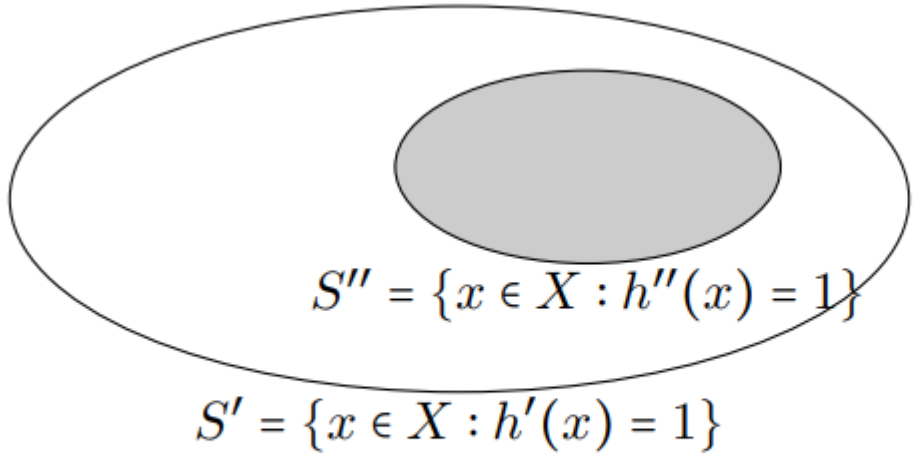
h2 = (?, ?, No, ?, ?) – Not Consistent

- A **consistent hypothesis** matches all positive examples (classified as "Yes") and avoids matching negative examples (classified as "No").
- An **inconsistent hypothesis** fails to correctly classify at least one example, either by wrongly classifying a positive example as "No" or misclassifying a negative example as "Yes."

Ordering of hypotheses

Definition

- Let X be the set of all possible examples for a binary classification problem and let h ′ and h ′′ be two hypotheses for the problem.



Hypothesis h' is more general than hypothesis h'' if and only if $S'' \subseteq S'$

1. We say that h ′ is more general than h ″ if and only if for every x ∈ X, if x satisfies h ″ then x satisfies h ′ also; that is, if h ″(x) = 1 then h ′ (x) = 1 also. The relation “is more general than” defines a partial ordering relation in hypothesis space.
2. We say that h′ is more specific than h″, if h″ is more general than h′.
3. We say that h′ is strictly more general than h″ if h′ is more general than h ″ and h″ is not more general than h ′ .
4. We say that h ′ is strictly more specific than h″ if h′ is more specific than h″ and h″ is not more specific than h ′

Example

- Consider the hypotheses h′ and h″ . defined in Eqs. Then it is easy to check that if h′(x) = 1 then h″ (x) = 1 also. So, h″ is more general than h′ .
- But, h′ is not more general than h″ and so h″ is strictly more general than h′ .

1. **Ordering of Hypotheses:** This refers to organizing the hypotheses in a general-to-specific or specific-to-general manner. The ordering can be done in two ways:
2. **General-to-Specific Ordering**: Hypotheses are ordered from the most general (least specific) to the most specific hypothesis.
3. **Specific-to-General Ordering:** Hypotheses are ordered from the most specific (least general) to the most general hypothesis.

- Let's say we are trying to classify a set of shapes as "Acceptable" or "Not Acceptable" based on three features:
- Color: {Red, Green}
- Shape: {Circle, Square}
- Size: {Large, Small}

- **Ordering the Hypotheses**
- **General-to-Specific Ordering:**
- H3: <Red, Any, Any> (most general, accepts all shapes with red color)
- H1: <Red, Circle, Large>
- H2: <Red, Circle, Small> (most specific, accepts only red circles of small size)
- **Specific-to-General Ordering:**
- H2: <Red, Circle, Small> (most specific)
- H1: <Red, Circle, Large>
- H3: <Red, Any, Any> (most general)

Version space

Definition

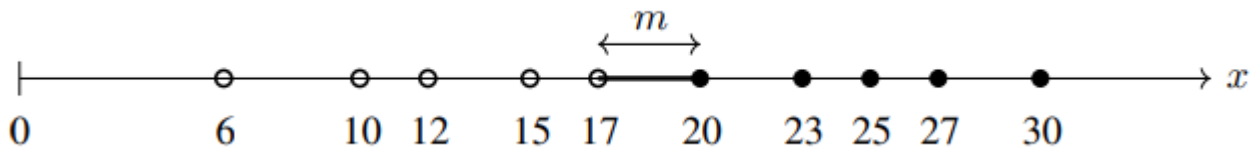
- Consider a binary classification problem. Let D be a set of training examples and H a hypothesis space for the problem.
- The version space for the problem with respect to the set D and the space H is the set of hypotheses from H consistent with D; that is, it is the set

$$\mathrm{VS}_{D,H} = \{h \in H | : h(x) = c(x) \text{ for all } x \in D\}.$$

- The version space $VS_{H,D}$ is the subset of the hypothesis from H consistent with the training example in D.
- In general terms, say that $VS_{H,D}$ contains all hypothesis which are consistent with the training examples.

Example

- Consider the data D given in Table and the hypothesis space defined by Equations



- Values of m which define the version space with data in Table and hypothesis space defined by above Equations.
- we can easily see that the hypothesis space with respect this dataset D and hypothesis space H is as given below:

$$\mathbf{VS}_{D,H} = \{h_m : 17 < m \leq 20\}.$$

Example

- Consider the problem of assigning the label “family car” (indicated by “1”) or “not family car” (indicated by “0”) to cars.
- Given the following examples for the problem and assuming that the hypothesis space is as defined by Eq.

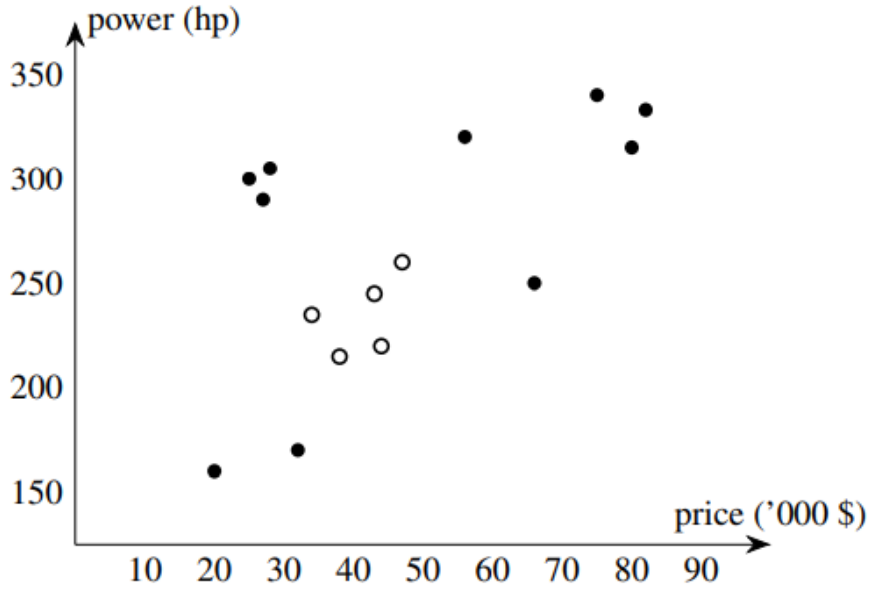
$$\text{IF } (p_1 < \text{price} < p_2) \text{ AND } (e_1 < \text{power} < e_2) \text{ THEN “1” ELSE ”0”}$$

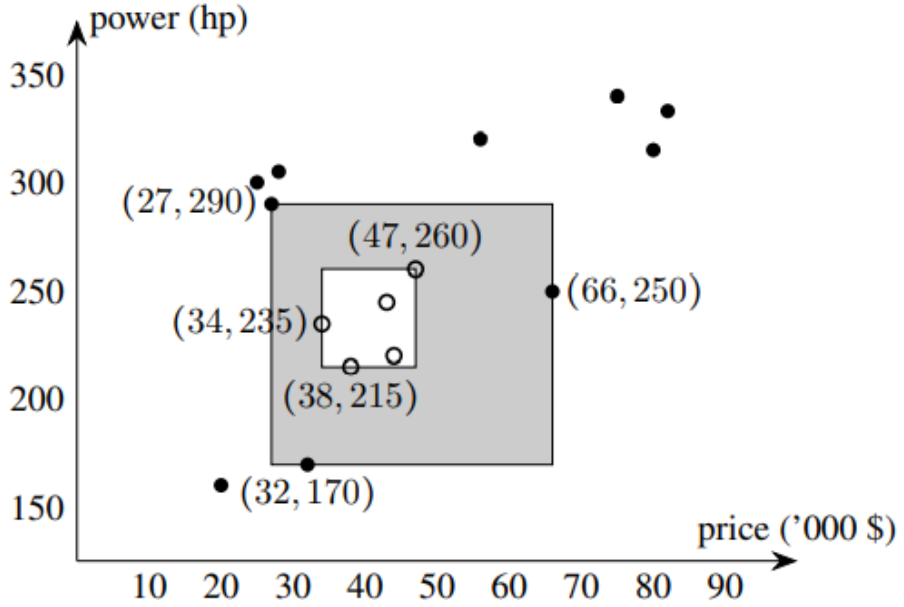
the version space for the problem.

x_1: Price in ’000 ($)	32	82	44	34	43	80	38
x_2: Power (hp)	170	333	220	235	245	315	215
Class	0	0	1	1	1	0	1

x_1	47	27	56	28	20	25	66	75
x_2	260	290	320	305	160	300	250	340
Class	1	0	0	0	0	0	0	0

- A hypothesis as given by above equation with specific values for the parameters p1, p2, e1 and e2 specifies an axis-aligned rectangle as shown in Figure.
- So the hypothesis space for the problem can be thought as the set of axis-aligned rectangles in the price-power plane





- The version space consists of hypotheses corresponding to axis-aligned rectangles contained in the shaded region.
- The version space consists of all hypotheses specified by axis-aligned rectangles contained in the shaded region.
- The inner rectangle is defined by

$$(34 < \text{price} < 47) \text{ AND } (215 < \text{power} < 260)$$

- The outer rectangle is defined by

$$(27 < \text{price} < 66) \text{ AND } (170 < \text{power} < 290).$$

Example 3

- Consider the problem of finding a rule for determining days on which one can enjoy water sport.
- The rule is to depend on a few attributes like “temp”, ”humidity”, etc. Suppose we have the following data to help us devise the rule.
- In the data, a value of “1” for “enjoy” means “yes” and a value of “0” indicates ”no”.

Example	sky	temp	humidity	wind	water	forecast	enjoy
1	sunny	warm	normal	strong	warm	same	1
2	sunny	warm	high	strong	warm	same	1
3	rainy	cold	high	strong	warm	change	0
4	sunny	warm	high	strong	cool	change	1

- Find the hypothesis space and the version space for the problem.

- We are required to find a rule of the following form, consistent with the data, as a solution of the problem.

$$(\text{sky} = x_1) \wedge (\text{temp} = x_2) \wedge (\text{humidity} = x_3) \wedge$$
$$(\text{wind} = x_4) \wedge (\text{water} = x_5) \wedge (\text{forecast} = x_6) \leftrightarrow \text{yes}$$

where

- x1 = sunny, warm, *
- x2 = warm, cold, *
- x3 = normal, high, *
- x4 = strong, *
- x5 = warm, cool, *
- x6 = same, change, *

- (Here a “*” indicates other possible values of the attributes.) The hypothesis may be represented compactly as a vector

$$(a1, a2, a3, a4, a5, a6)$$

- where, in the positions of a1, . . . , a6, we write
  - a “?” to indicate that any value is acceptable for the corresponding attribute,
  - a ”Ø” to indicate that no value is acceptable for the corresponding attribute,
  - some specific single required value for the corresponding attribute

- For example, the vector

(?, cold, high, ?, ?, ?)

indicates the hypothesis that one enjoys the sport only if “temp” is “cold” and “humidity” is “high” whatever be the values of the other attributes.

- It can be shown that the version space for the problem consists of the following six hypotheses only:
- (sunny, warm, ?, strong, ?, ?)
- (sunny, ?, ?, strong, ?, ?)
- (sunny, warm, ?, ?, ?, ?)
- (?, warm, ?, strong, ?, ?)
- (sunny, ?, ?, ?, ?, ?)
- (?, warm, ?, ?, ?, ?)

Example of Ordering of Hypothesis and Version Space

- Let's say we are trying to classify a set of shapes as "Acceptable" or "Not Acceptable" based on three features:
- Color: {Red, Green}
- Shape: {Circle, Square}
- Size: {Large, Small}

Step 1: Define Hypothesis Space (H)

- The hypothesis space will consist of all possible combinations of the features:
- H1: <Red, Circle, Large>
- H2: <Red, Circle, Small>
- H3: <Green, Square, Large>
- ...
- HN: (more combinations)

Step 2: Given Training Examples

- Assume we have the following training examples:
- <Red, Circle, Large> → Acceptable
- <Green, Square, Small> → Not Acceptable
- <Red, Circle, Small> → Acceptable

Step 3: Construct the Version Space

- We eliminate any hypotheses from the hypothesis space that are inconsistent with the training examples. After filtering, the remaining hypotheses form the version space.
- Let's say the consistent hypotheses after filtering are:
- H1: <Red, Circle, Large>
- H2: <Red, Circle, Small>
- H3: <Red, Any, Any>

Step 4: Ordering the Hypotheses

- General-to-Specific Ordering:
- H3: <Red, Any, Any> (most general, accepts all shapes with red color)
- H1: <Red, Circle, Large>
- H2: <Red, Circle, Small> (most specific, accepts only red circles of small size)
- Specific-to-General Ordering:
- H2: <Red, Circle, Small> (most specific)
- H1: <Red, Circle, Large>
- H3: <Red, Any, Any> (most general)

Key Points

- General Hypotheses: Cover a wider range of instances and are less specific.
- Specific Hypotheses: Cover a narrower range of instances and are more precise.
- Version Space: Shrinks as more training examples are observed because inconsistent hypotheses are eliminated.

Learning multiple classes

- In a general case there may be more than two classes. Two methods are generally used to handle such cases. These methods are known by the names “one-against-all" and “one-against-one”.

Procedures for learning multiple classes

“One-against all” method

- Consider the case where there are K classes denoted by $c_1 \ldots\ldots, c_k$. Each input instance belongs to exactly one of them.
- We view a K-class classification problem as K two-class problems. In the i-th two-class problem, the training examples belonging to Ci are taken as the positive examples and the examples of all other classes are taken as the negative examples.
- So, we have to find K hypotheses $h_1 \ldots\ldots, h_k$ where h_i is defined by

$$h_i(x) = \begin{cases} 1 & \text{if } x \text{ is in class } C_i \\ 0 & \text{otherwise} \end{cases}$$

- For a given x, ideally only one of h_i (x) is 1 and then we assign the class c_i to x. But, when no, or, two or more, h_i (x) is 1, we cannot choose a class. In such a case, we say that the classifier rejects such cases.

"One-against-one" method

- In the one-against-one (OAO) (also called one-vs-one (OVO)) strategy, a classifier is constructed for each pair of classes. If there are K different class labels, a total of $K(K-1)/2$ classifiers are constructed. An unknown instance is classified with the class getting the most votes. Ties are broken arbitrarily.

Generalisation

- How well a model trained on the training set predicts the right output for new instances is called generalization.
- Generalization refers to how well the concepts learned by a machine learning model apply to specific examples not seen by the model when it was learning.
- The goal of a good machine learning model is to generalize well from the training data to any data from the problem domain. This allows us to make predictions in the future on data the model has never seen.
- Overfitting and underfitting are the two biggest causes for poor performance of machine learning algorithms.
- The model should be selected having the best generalisation. This is said to be the case if these problems are avoided.

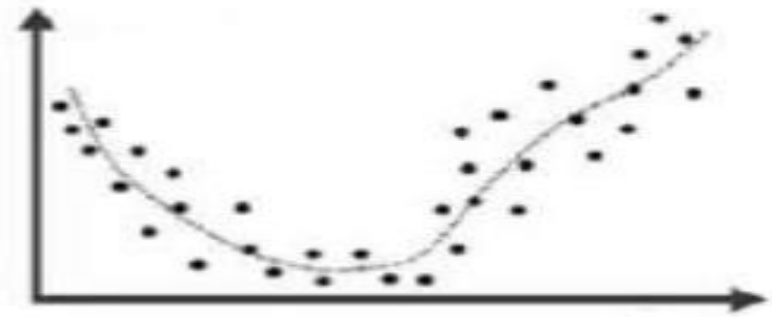
Underfitting

- Underfitting is the production of a machine learning model that is not complex enough to accurately capture relationships between a dataset A-Z features and a target variable.

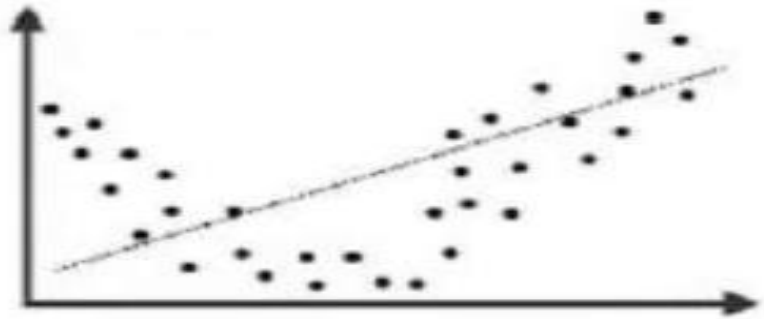
Overfitting

- Overfitting is the production of an analysis which corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably.

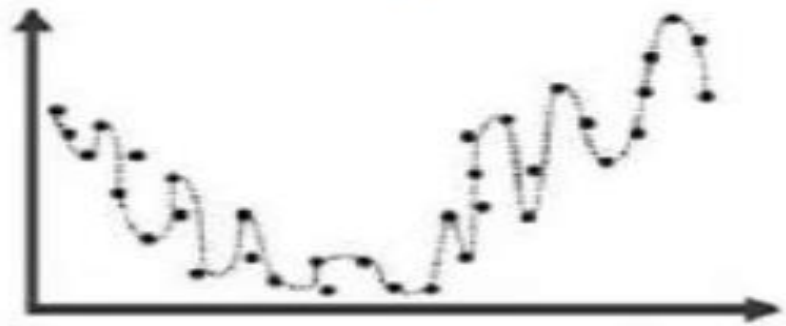
(a) Given dataset



(b) "Just right" model

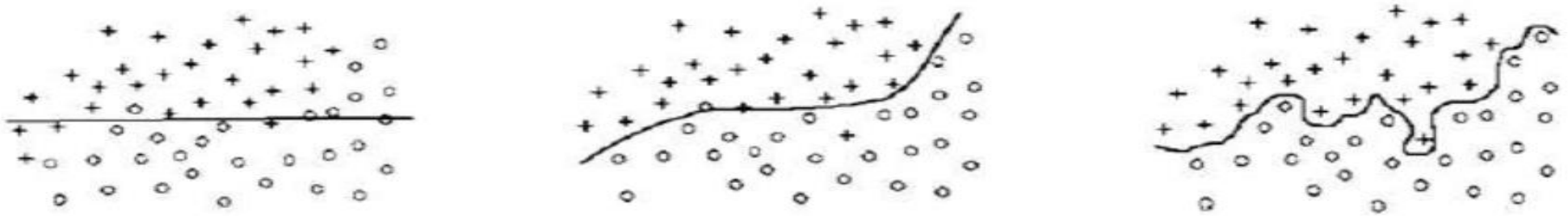


(c) Underfitting model



(d) Overfitting model

Figure 2.7: Examples for overfitting and overfitting models



(a) Underfitting (b) Right fitting (c) Overfitting

Figure 2.8: Fitting a classification boundary

- Suppose we have to determine the classification boundary for a dataset two class labels. An example situation is shown in Figure where the curved line is the classification boundary.
- The three figures illustrate the cases of underfitting, right fitting and overfitting.

VC dimension and PAC learning

- The concepts of Vapnik-Chervonenkis dimension (VC dimension) and probably approximate correct (PAC) learning are two important concepts in the mathematical theory of learnability and hence are mathematically oriented.

Vapnik-Chervonenkis dimension.

- Let H be the hypothesis space for some machine learning problem. The Vapnik-Chervonenkis dimension of H, also called the VC dimension of H, and denoted by V C(H), is a measure of the complexity (or, capacity, expressive power, richness, or flexibility) of the space H.
- To define the VC dimension we require the notion of the shattering of a set of instances.

Shattering of a set

- Let D be a dataset containing N examples for a binary classification problem with class labels 0 and 1.
- Let H be a hypothesis space for the problem. Each hypothesis h in H partitions D into two disjoint subsets as follows:

$$\{x \in D \,|\, h(x) = 0\} \text{ and } \{x \in D \,|\, h(x) = 1\}.$$

- Such a partition of S is called a "dichotomy" in D. It can be shown that there are 2^N possible dichotomies in D.
- To each dichotomy of D there is a unique assignment of the labels "1" and "0" to the elements of D. Conversely, if S is any subset of D then, S defines a unique hypothesis h as follows

$$h(x) = \begin{cases} 1 & \text{if } x \in S \\ 0 & \text{otherwise} \end{cases}$$

- Figure shows all possible dichotomies of D if D has three elements.
- In the figure, we have shown only one of the two sets in a dichotomy, namely the set $\{x \in D \mid h(x) = 1\}$. The circles and ellipses represent such sets.

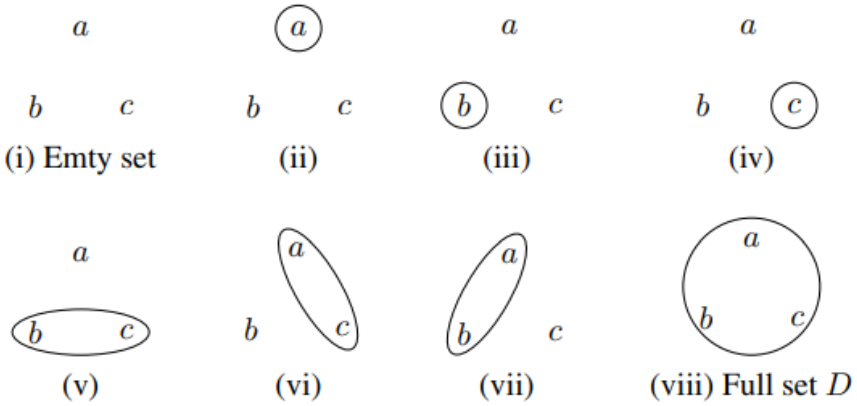


Figure 3.1: Different forms of the set $\{x \in S : h(x) = 1\}$ for $D = \{a, b, c\}$

- Definition A set of examples D is said to be shattered by a hypothesis space H if and only if for every dichotomy of D there exists some hypothesis in H consistent with the dichotomy of D.

Vapnik-Chervonenkis dimension

- Example Let the instance space X be the set of all real numbers. Consider the hypothesis space defined by Equations

$$H = \{h_m : m \text{ is a real number}\},$$

Where,

$$h_m \quad : \quad \text{IF } x \geq m \text{ THEN "1" ELSE "0".}$$

- Let D be a subset of X containing only a single number, say, D = {3.5}. There are 2 dichotomies for this set. These correspond to the following assignment of class labels

x	3.25
Label	0

x	3.25
Label	1

- h4 ∈ H is consistent with the former dichotomy and h3 ∈ H is consistent with the latter. So, to every dichotomy in D there is a hypothesis in H consistent with the dichotomy.
- Therefore, the set D is shattered by the hypothesis space H.

- Let D be a subset of X containing two elements, say, D = {3.25, 4.75}. There are 4 dichotomies in D and they correspond to the assignment of class labels shown in Table

x	3.25	4.75
Label	0	0

(a)

x	3.25	4.75
Label	0	1

(b)

x	3.25	4.75
Label	1	0

(c)

x	3.25	4.75
Label	1	1

(d)

- In these dichotomies, h5 is consistent with (a), h4 is consistent with (b) and h3 is consistent with (d).
- But there is no hypothesis $h_m \in$ H consistent with (c).
- Thus the two-element set D is not shattered by H.
- In a similar way it can be shown that there is no two-element subset of X which is shattered by H
- It follows that the size of the largest finite subset of X shattered by H is 1. This number is the VC dimension of H.