CDS503: Machine Learning

Topic 2

Supervised Learning

Assoc. Prof. DR UMI KALSOM YUSOF
SCHOOL OF COMPUTER SCIENCES
UNIVERSITI SAINS MALAYSIA (USM)

Contents

- Learning a Class from Examples
- Formalizing supervised learning
  - Instance space
  - Label space
  - Hypothesis space
- Noise
- Model Selection and Generalization
- Dimensions of a Supervised Machine Learning Algorithm

Some slides are based on Supervised Learning: The Setup, Machine Learning Fall 2018, The University of Utah

The Car Searching

Name	Label
Ferrari	-
Mazda 8	+
Mazda CX5	-
Buggati Chiron	-
Honda City	-
Toyota Vios	-
Toyota Avanza	+
Toyota Vellfire	+
Honda Odyssey	+
Mini Cooper R53	-
Kia Carnival	+

Searching for a family car

How are the labels generated?

The Car Searching

Name	Label
Ferrari	-
Mazda 8	+
Mazda CX5	-
Buggati Chiron	-
Honda City	-
Toyota Vios	-
Toyota Avanza	+
Toyota Vellfire	+
Honda Odyssey	+
Mini Cooper R53	-
Kia Carnival	+

What is the label for “Hyundai Startex”?

What about the label for “Honda Accord”?

The Car Searching

Name	Label
Ferrari	-
Mazda 8	+
Mazda CX5	-
Buggati Chiron	-
Honda City	-
Toyota Vios	-
Toyota Avanza	+
Toyota Vellfire	+
Honda Odyssey	+
Mini Cooper R53	-
Kia Carnival	+

How are the labels generated?

Is it depends on the price and engine power?

x_1: price, x_2 : engine power

Identifying a class label (20 minutes)

Task:

- Form a group of consist 5 members – same group as assigned
- Each group - discuss on the type of dataset assigned to you
  - List of the at least 5 items/instances that consist of:
    - 2 or 3 class labels
    - 3 features/characteristics that determine the class labels
  - Identity two (2) new members of the item and determine the class label.
  - Paste to respective Padlet in eLearn

Item Id	Feature 1	Feature 2	Feature 3	Label

No	Item	Group	Group	Group
1	Flowers	1	7	13
2	Houses	2	8	14
3	Sport Car	3	9	15
4	Foods	4	10	16
5	Phones	5	11	
6	Song	6	12	

The Car Searching

- ***Class C*** of a "family car"
  - Prediction: Is car *x* a family car?
  - Knowledge extraction:
    - What do people expect from a family car?
- **Output:**

  Positive (+) and negative (–) examples
- **Input representation:**

  x_1: price, x_2 : engine power

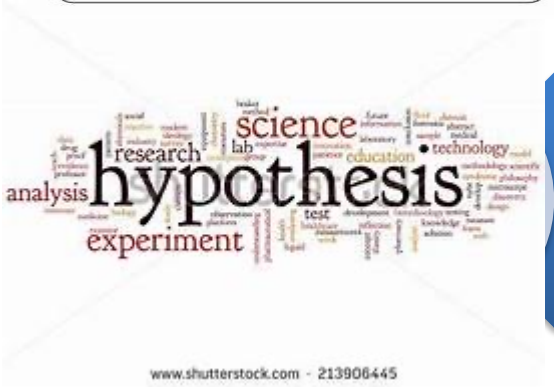
The Car Searching

- Questions:
  - Are you sure you got the correct function?
  - How did you arrive at it?
  - Learning issues:
    - Is this prediction or just modeling data?
    - What "learning algorithm" did you use?

Formalizing Supervised Learning

Instance space and feature

Label space



Hypothesis space

Instances and Labels

Example: Automatically tag news instance

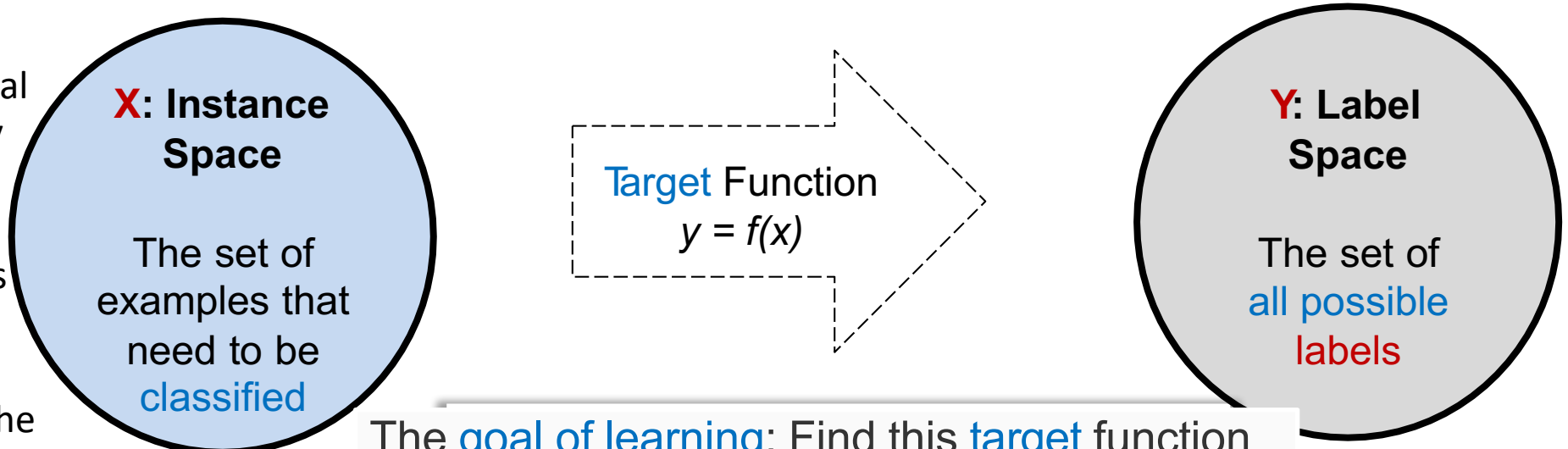
An **instance** of a news article need to be classified

label

Instances and Labels

Designing an appropriate feature representation of the instance space is crucial
Instances X are defined by features/attributes.

Example: Boolean features
- Does the email contain the word "free"?
- What is the height of the person?
- What was the stock price yesterday?



Example: The set of all possible documents, names, sentences, images, emails, etc.

The goal of learning: Find this target function

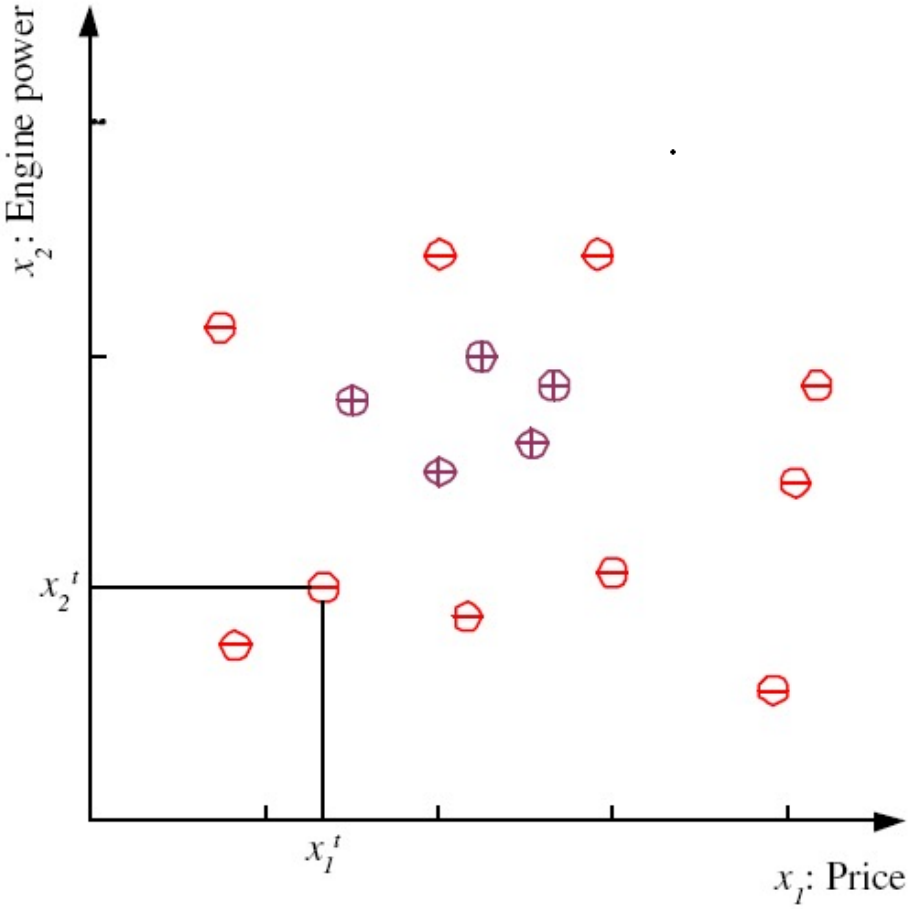
Learning is search over functions

Example: {Sick, Not Sick},

{+, -} etc.

Supervised Learning: Training

Training set X for a class of 'family car'



$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \begin{matrix} - \text{ Price} \\ - \text{ engine} \end{matrix}$$

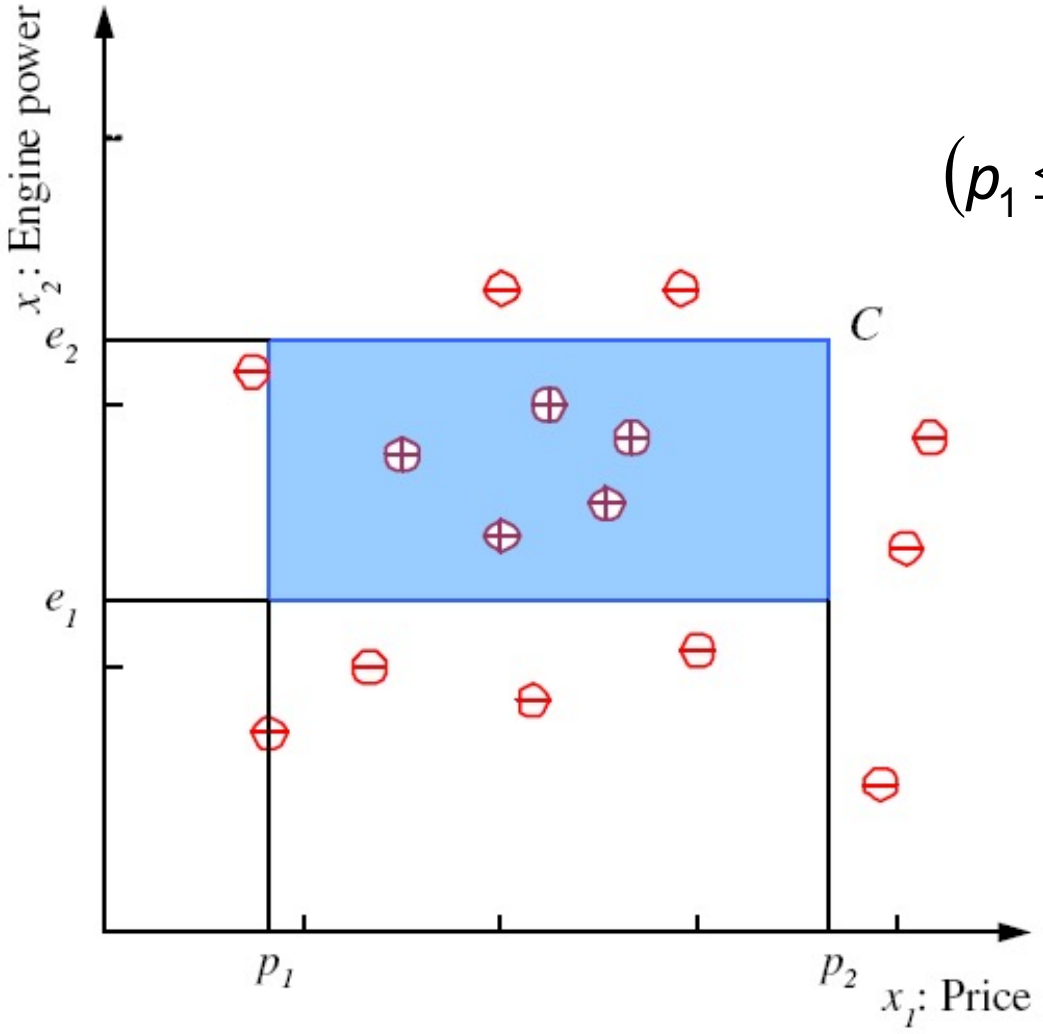
$$r = \begin{cases} 1 \text{ if } \mathbf{x} \text{ is positive} \\ 0 \text{ if } \mathbf{x} \text{ is negative} \end{cases}$$

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

Where t represents different example in the set

Formalizing Supervised Learning

Class C



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

On using supervised learning

- What is our **instance space**?
  - What are the inputs to the problem? What are the features?
- What is our **label space**?
  - What is the predictive task?
- What is our **hypothesis space**?
  - What functions should the learning algorithm search over?

Name	Price	Engine	**Label**
Ferrari			-
Mazda 8			+
Mazda CX5			-
Buggati Chiron			-
Honda City			-
Toyota Vios			-
Toyota Avanza			+
Toyota Vellfire			+
Honda Odyssey			+
Mini Cooper R53			-
Kia Carnival			+

Good features are essential

- Good features decide how well a task can be learned
  - Eg: A bad feature function the badges game
    - "Is there a day of the week that begins with the last letter of the first name?"
- Much effort goes into designing features
  - Or maybe learning them
- Touch upon general principles for designing good features
  - But feature definition largely domain specific
  - Comes with experience

5MoT

Name	Price	Engine	**Label**
Ferrari			-
Mazda 8			+
Mazda CX5			-
Buggati Chiron			-
Honda City			-
Toyota Vios			-
Toyota Avanza			+
Toyota Vellfire			+
Honda Odyssey			+
Mini Cooper R53			-
Kia Carnival			+

Instances $x \in X$ are defined by features/attributes

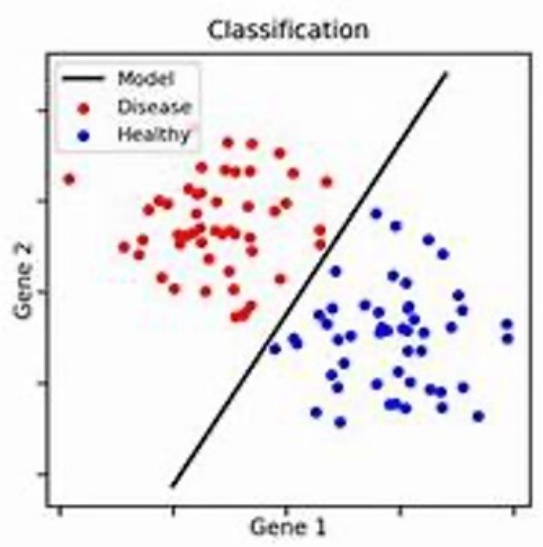
The choice of features is crucial to how well a task can be learned.

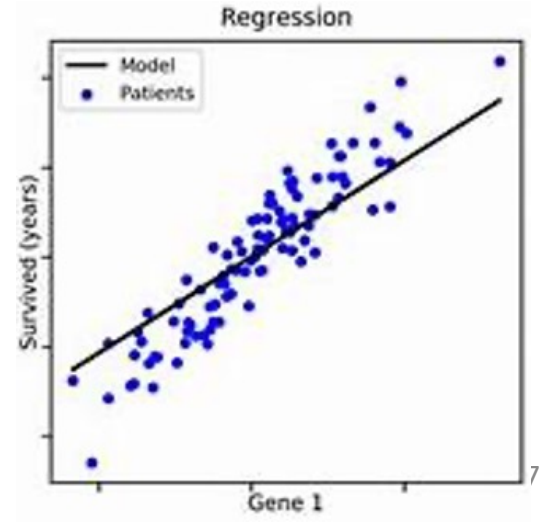
What are other possible features in the Car Searching?

- *Number of seats?*
- *The look?*
- *The comfortable level?*
- *??*

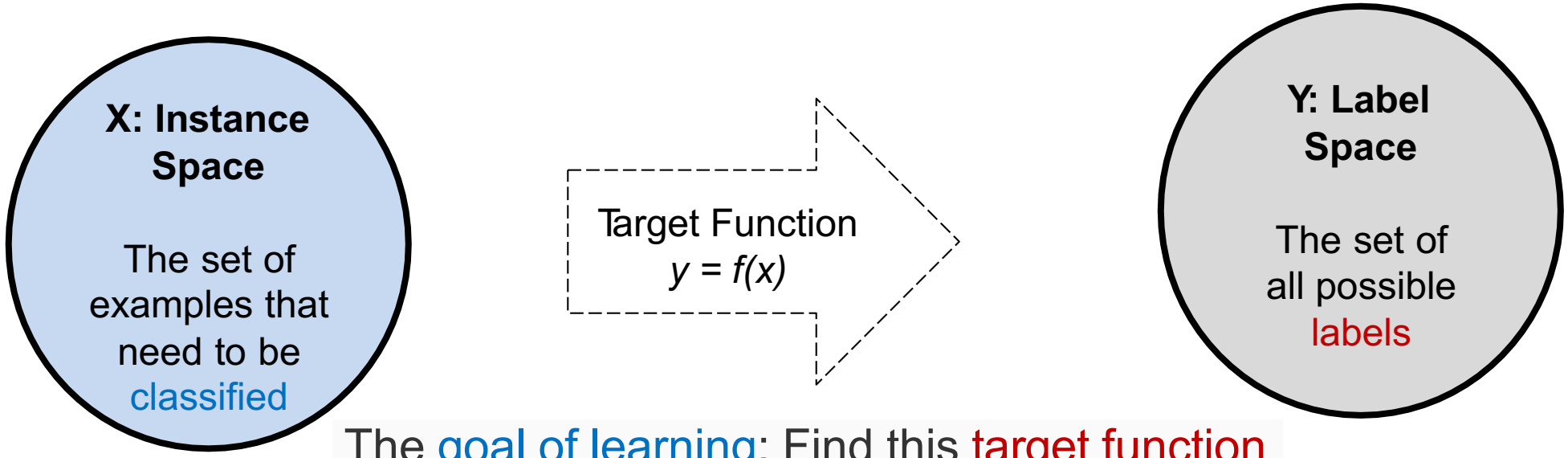
Label Space

- Determines what kind of supervised learning task we are dealing with
- **Classification**: Output is categorical
  - **Binary** classification: Two possible labels, $y \in \{-1, 1\}$ (e.g., [Yes, No], [Sick, Not Sick])
  - **Multi-class** classification: More than two possible labels [Like, Dislike, Neutral]
- **Regression**: Output is numerical (real numbers)





Hypothesis Space



The goal of learning: Find this target function

Learning is search over functions

The **hypothesis space** is the set of functions we consider for this search

Hypothesis Space

- Restrict the search space
- A hypothesis space is a set of possible functions we consider
  - Choose a hypothesis space that is smaller than the space of all functions
  - Make some extra assumptions to make learning possible
  - Set of assumptions = **Inductive bias**

Inductive Bias:
Every machine learning algorithm with any ability to generalize beyond the training data that it sees has some type of inductive bias. This is the assumptions made by the model to learn the target function and to generalize beyond training data.

Hypothesis Space

How do we pick a hypothesis space?

- Use some prior knowledge or by guessing

What if the hypothesis space is so small that nothing in it agrees with the data?

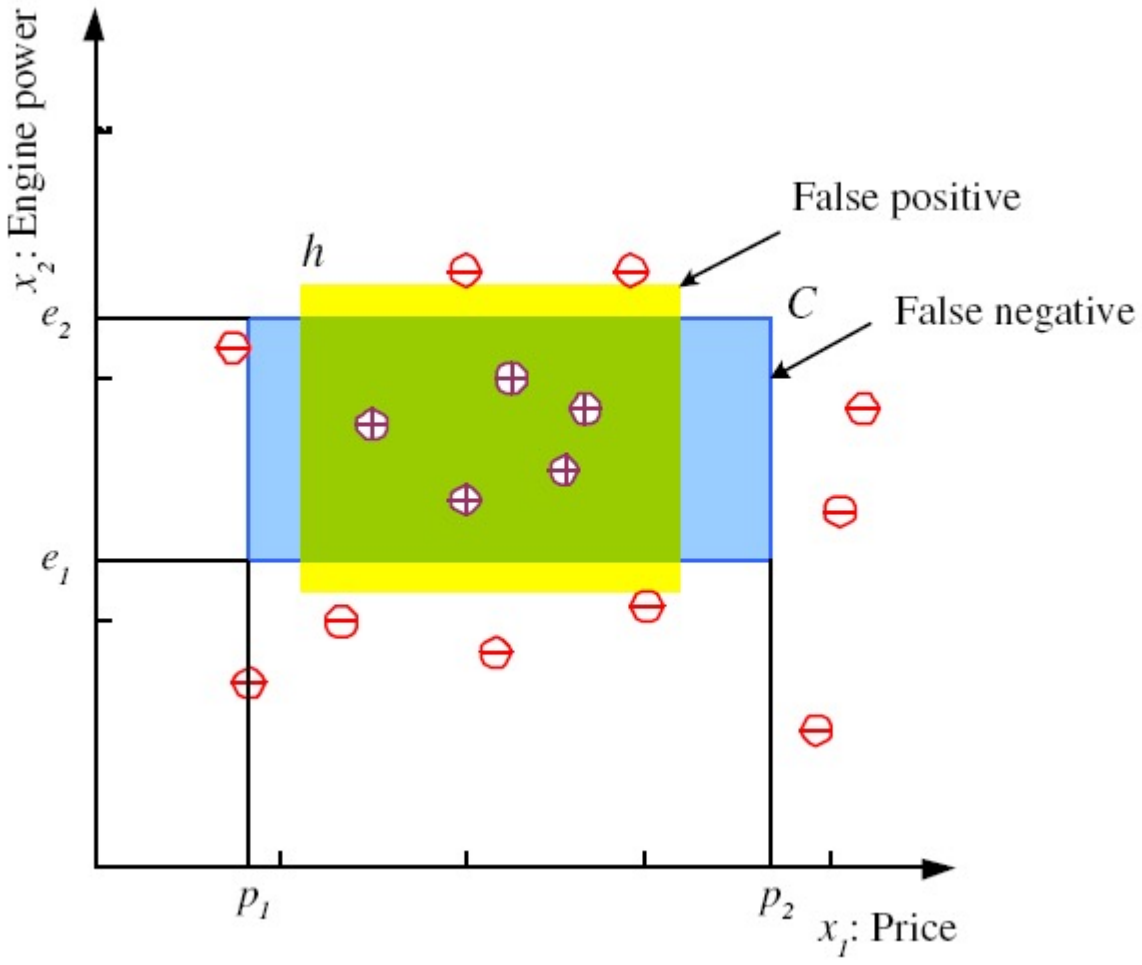
- Need a hypothesis space that is flexible enough

Views of learning

- Learning is the removal of the remaining uncertainty
- Learning requires a good, small hypothesis space

- Learning algorithms **find a hypothesis in our space** and find the way to guarantee that it generalizes well

Hypothesis *Class H*

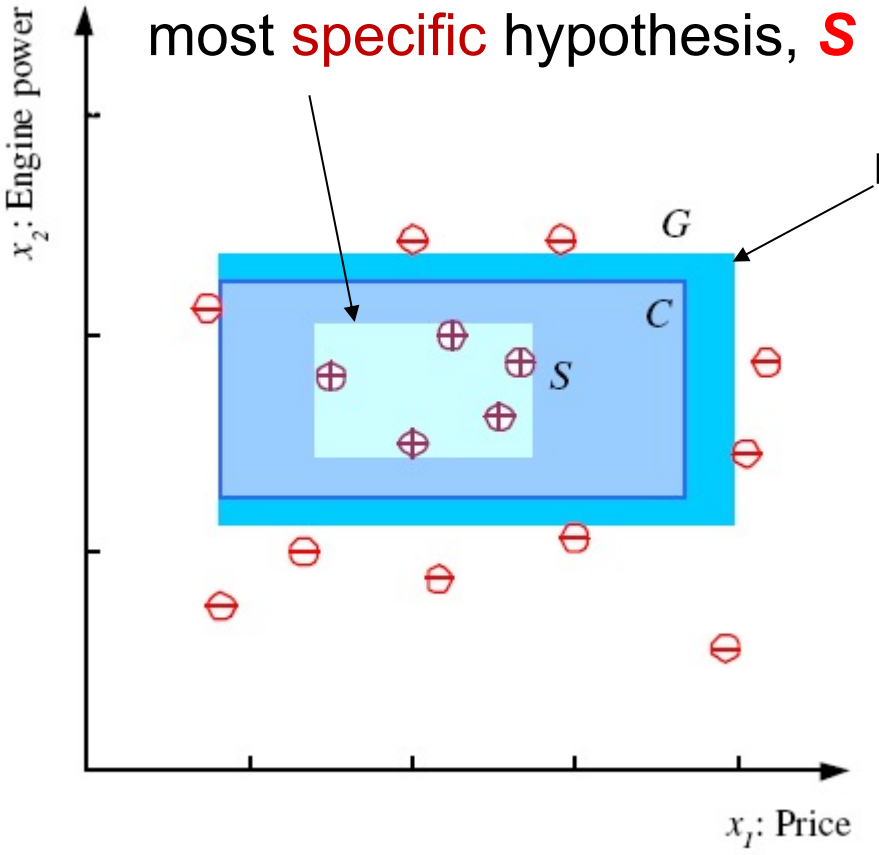


$$h(\mathbf{x}) = \begin{cases} 1 \text{ if } h \text{ says } \mathbf{x} \text{ is positive} \\ 0 \text{ if } h \text{ says } \mathbf{x} \text{ is negative} \end{cases}$$

Error of h on H

$$E(h \mid \mathcal{X}) = \sum_{t=1}^{N} 1\left(h\left(\mathbf{x}^t\right) \neq r^t\right)$$

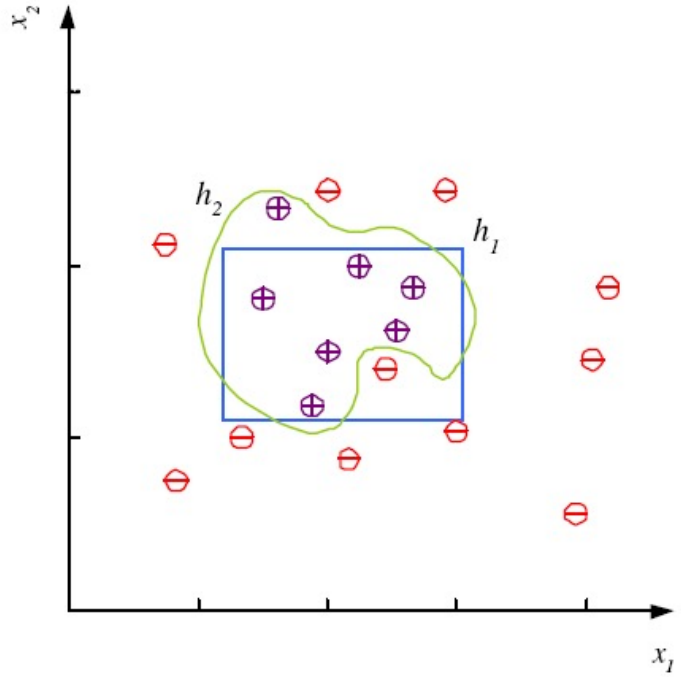
S, G, and the Version Space



between *S* and *G* is consistent and make up the version space

Noise and Model Complexity

- Unwanted anomaly in the data - Class may difficult to learn
- May due to:
  - Impression in recording the input attributes
  - Errors in labeling the data points; positive instance as negative and vice versa
  - Maybe additional attributes that not taken into account
- When there is noise, there is not a simple boundary between the positive and negative instances
- A rectangle is a simple hypothesis with four parameters defining the corners. An arbitrary closed form can be drawn by piecewise functions with a larger number of control points



anomaly detection (also **outlier detection**) is the identification of rare items, events or observations which raise suspicions by differing significantly from the majority of the data

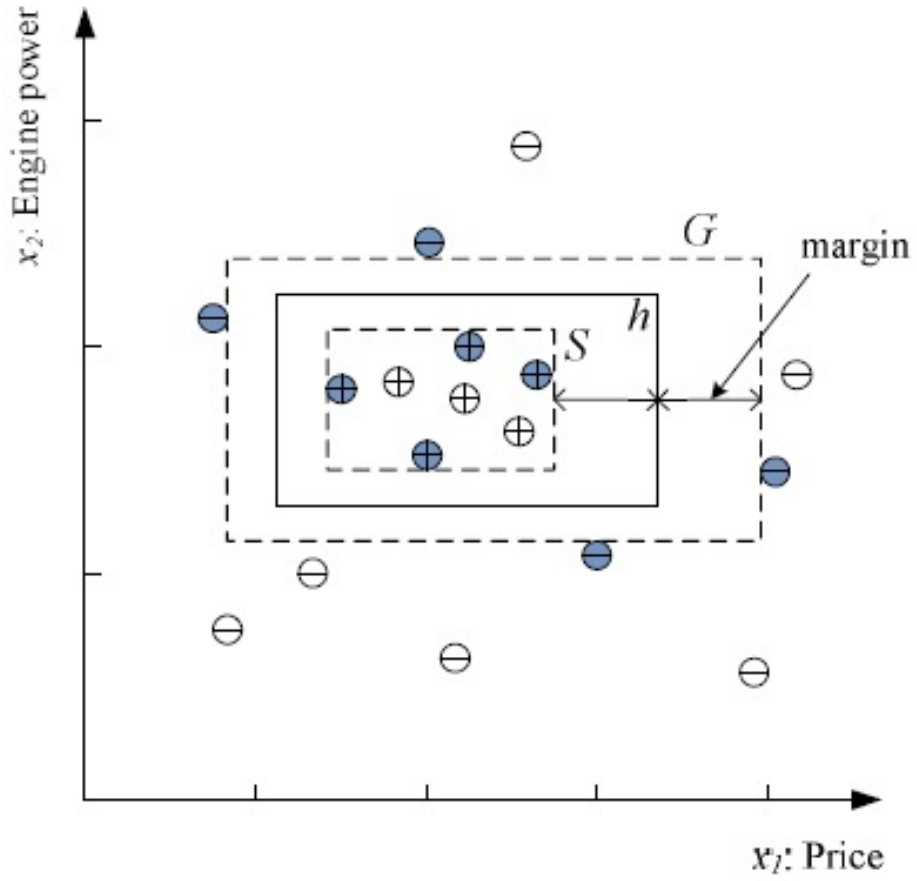
Noise and Model Complexity

Use the simpler Model one because

- Simpler to use (lower computational complexity)
- Easier to train (lower space complexity)
- Easier to explain (more interpretable)
- Generalizes better (lower variance - Occam's razor)

Margin

- Choose *h* with largest margin



- Doubt
  - Some application – wrong decision may be very costly
  - Any instance falls in between **S** and **G** is a doubt
  - System rejects the instance and defers the decision to human expert.

Generalization

- Learning is not possible without inductive bias,
  - how to choose the right bias.
- This is called *model selection*, which is choosing between possible H .
- Aim of machine learning is rarely to replicate the training data but the prediction for new cases.
  - able to generate the right output for an input instance outside the training set, one for which the correct output is not given in the training set.
  - ***Generalization -*** How well a model trained on the training set predicts the right output for new instances.

Underfitting and Overfitting

Overfitting

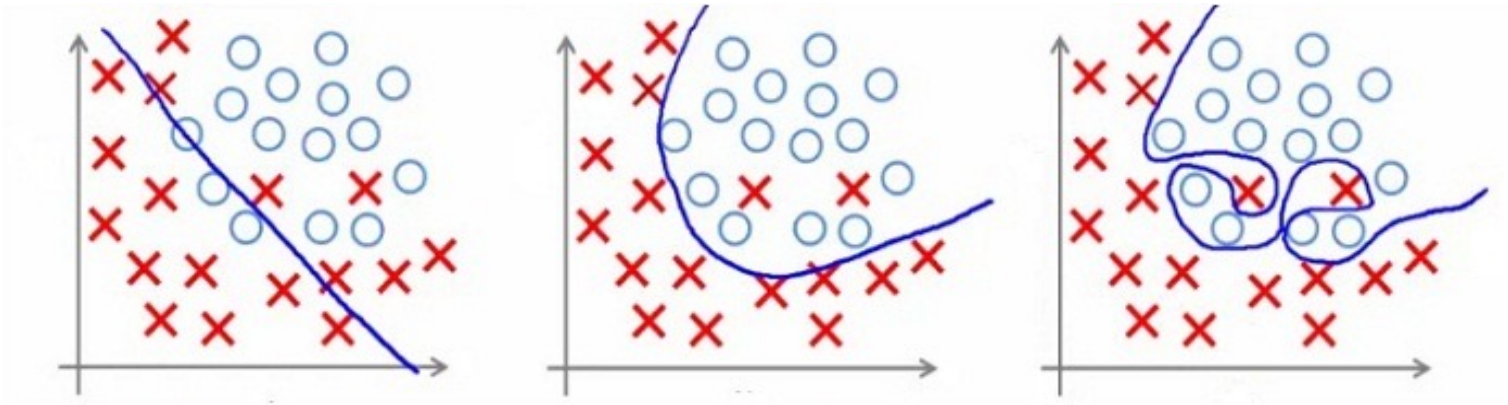
- Refers to a model that models the training data **too well**.
  - when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data.
- Overfitting is more likely with nonparametric and nonlinear models that have more flexibility when learning a target function.

Underfitting

- Refers to a model that can neither model the training data nor generalize to new data.
- An underfit machine learning model is not a suitable model and will be obvious as it will have poor performance on the training data.
- Often not discussed as it is easy to detect given a good performance metric.

Model Selection and Generalization

Example: Classification



Under-fitting

(too simple to explain the variance)

Appropriate-fitting

Over-fitting

(forcefitting -- too good to be true)

May be caused by noise (unwanted anomaly in the data)

Example in human

A

Not interested in learning

Class test ~50%
Test ~47%

Under-fit/ biased learning

B

Memorizing the lessons

Class test ~98%
Test ~69%

Over-fit/ Memorizing

C

Conceptual Learning

Class test ~92%
Test ~89%

Best-fit

Summarize of selecting Model

- Choosing between possible hypothesis or choosing the right bias
- Aim of Machine Learning:
  - Not to replicate the training data
  - **Generalization**: How well a model trained on the training set predicts the correct output for new instances?
  - For best generalization, complexity of the learned function, *g(x)*, must match the complexity of the function underlying the data, *f(x)*

Model Selection

- Triple trade-off in all learning algorithms trained from example data
  - Complexity of the hypothesis H, c (H)
  - Amount of training data, N
  - Generalization error, E, on new examples

As N , $E \downarrow$
As c ($\mathcal{H}$) $\uparrow$, first $E \downarrow$ and then $E \uparrow$

- As the amount of training data increases, the generalization error decreases.
- As the complexity of the model class H increases, the generalization error decreases first and then starts to increase.
- The generalization error of an over complex H can be kept in check by increasing the amount of training data but only up to a point.
- We can measure the generalization ability of a hypothesis, namely, the quality of its inductive bias, if we have access to data outside the training set.

Machine Learning Algorithm Dimensions

- **Model** used in learning g(x|θ)
  - g(·) is the model, x is the input, θ are the parameters
- **Loss function**
  - Compute the difference between the desired output and our approximation, g(·), given the current value of the parameters θ (approximation error)
- **Optimization procedure**
  - Find θ that minimizes the total error

A story of fruits ….

- You have a basket full with some fresh fruits - apple, banana, cherry, grape
  - Your task is to arrange the same type fruits at one place.
  - You already know from your previous work that, the shape of each and every fruit so it is easy to arrange the same type of fruits at one place.
  - Your previous work is called as **training data** in data mining.
- So you already learn the things from your training data.
  - This is because of you have a **response variable** which says you that if some fruit have so and so features it is grape, like that for each and every fruit.
  - In ML, these are called features.
- This type of data you will get from the **training data.**
- This type of learning is called as **supervised learning**.
- This type solving problem come under **Classification.**

- So you already learn the things so you can do your job confidently.

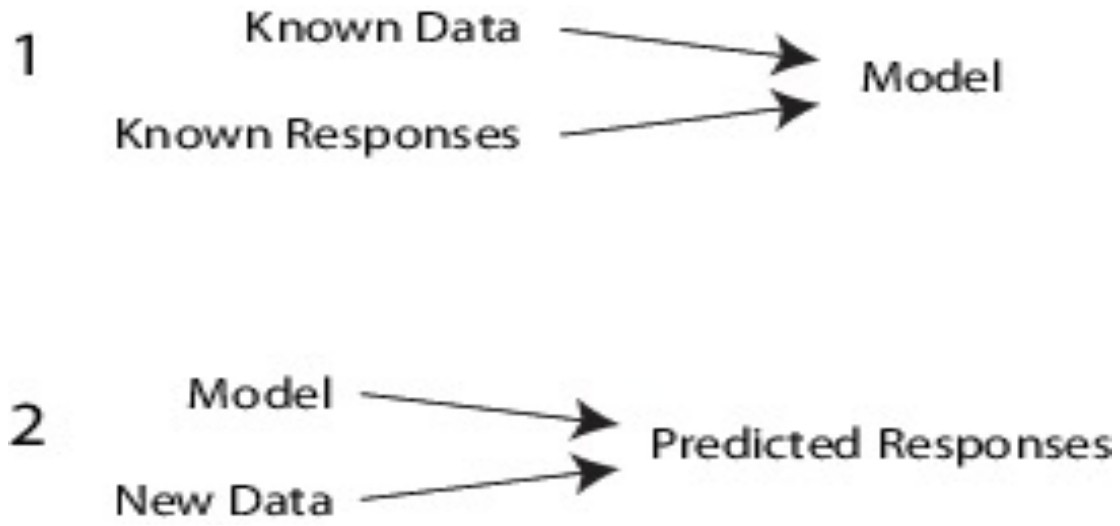
A story of fruits ….

- Suppose you had a basket and it is fulled with some fresh fruits your task is to arrange the same type fruits at one place.
- This time you don't know any thing about that fruits, you are first time seeing these fruits so how will you arrange the same type of fruits.
- What you will do first you take on fruit and you will select any physical character of that particular fruit. Suppose you taken colours.
- Then the groups will be some thing like this:
  - RED COLOUR: apples & cherry
  - GREEN COLOUR: grapes & pear

This type of learning is know **unsupervised learning.**

Aim Of Supervised Learning

- The aim of supervised machine learning is to build a model that makes predictions based on evidence in the presence of uncertainty.
- As adaptive algorithms identify patterns in data, a computer "learns" from the observations.
  - When exposed to more observations, the computer improves its predictive performance.



Example

Suppose you want to predict whether someone will have a heart attack within a year.

- You have a set of data on **previous patients**, including age, weight, height, blood pressure, etc.
- You know whether the previous patients had heart attacks within a year of their measurements.
- So, the problem is combining all the existing data into a model that can predict whether a new person will have a heart attack within a year.

Another Example

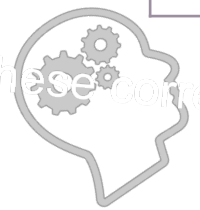
How Supervised Learning Solve this?

- First, we need LARGE amount of appropriate data on people in this campus.
- Second, we need to extract FEATURES from each person.
- Third, we select one algorithm and train it.

↓Training data for supervised learning

Id	Young	Hair	Fashion	Tired
1	young	blond	t shirts	not tired
2	young	black	suits	tired
3	not young	white	suits	tired
4	young	black	t shirts	not tired

Label



How do it learn a rule to classify?

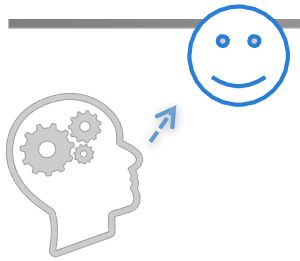
- It is like polls.

Id	Young	Hair	Fashion	Tired
1	young	blond	t shirts	not tired
2	young	black	suits	tired
3	Not young	white	suits	tired
4	young	black	t shirts	not tired

Label
Undergrad.
Undergrad.
Teacher
Undergrad.

Feature	Undergrad.	Teacher
Tired	+1	+1
Not tired	+2	
Young	+3	
Not young		+1
Blond hair	+1	
Black hair	+1	
White hair		+1
suits	+1	+1
t shirts	+2	

When a new person come up, how will it classify this person?



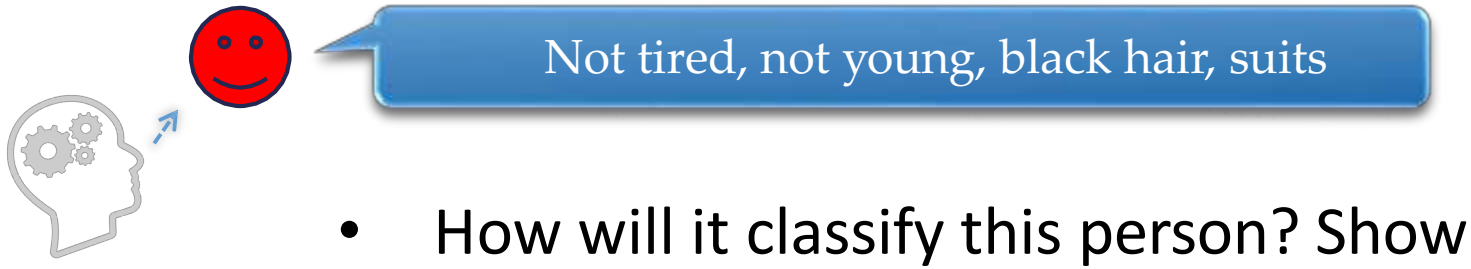
Tired, young, black hair, suits

Feature	Undergrad.	Teacher
Tired	**+1**	**+1**
Not tired	+2	
Young	**+3**	
Not young		+1
Blond hair	+1	
Black hair	**+1**	
White hair		+1
suits	**+1**	**+1**
t shirts	+2	

Undergrad.	Teacher
6 points	2 points

Undergrad. !!

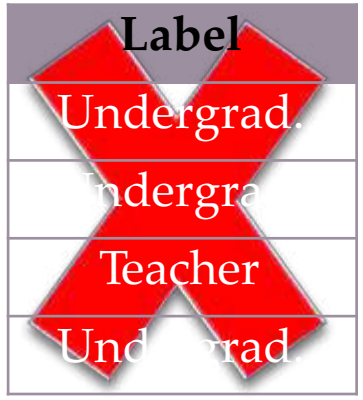
Exercise 1



- How will it classify this person? Show your workings.

How **Unsupervised** Learning Solve this?

Id	Young	Hair	Fashion	Tired
1	young	blond	t shirts	not tired
2	young	black	suits	tired
3	not young	white	suits	tired
4	young	black	t shirts	not tired

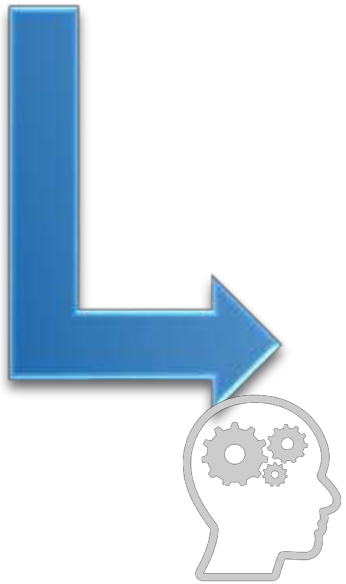


Unsupervised learning
don't need labels

Id	young	not young	blond	black	white	t shirts	suits	tired	not tired
1	1		1			1			1
2	1			1			1	1	
3		1			1		1	1	
4	1			1		1			1

Make a distance matrix

Id	young	not young	bold	black	white	t shirts	suits	tired	not tired
1	1		1			1			1
2	1			1			1	1	
3		1			1		1	1	
4	1			1		1			1



Calculate **distance** between every pair.

There are many types of distance.

- Euclidean distance
- Correlation coefficient
- Cosine similarity, etc.

Calculate Euclidean Distance

Id	young	not young	bold	black	white	t shirts	suits	tired	not tired	Distance
1	1		1			1			1	
2	1			1			1	1		
	$(1-1)^2$	$(0-0)^2$	$(1-0)^2$	$(0-1)^2$	$(0-0)^2$	$(1-0)^2$	$(0-1)^2$	$(0-1)^2$	$(1-0)^2$	$\sqrt{6}$
1	1		1			1			1	
3		1			1		1	1		
	$(1-0)^2$	$(0-1)^2$	$(1-0)^2$	$(0-0)^2$	$(0-1)^2$	$(1-0)^2$	$(0-1)^2$	$(0-1)^2$	$(1-0)^2$	$2\sqrt{2}$
1	1		1			1			1	
4	1			1		1			1	
	$(1-1)^2$	$(0-0)^2$	$(1-0)^2$	$(0-1)^2$	$(0-0)^2$	$(1-1)^2$	$(0-0)^2$	$(0-0)^2$	$(1-1)^2$	$\sqrt{2}$
2	1			1			1	1		
3		1			1		1	1		
	$(1-0)^2$	$(0-1)^2$	$(0-0)^2$	$(1-0)^2$	$(0-1)^2$	$(0-0)^2$	$(1-1)^2$	$(1-1)^2$	$(0-0)^2$	2
3		1			1		1	1		

Find the most similar people

↓ Distance Matrix

	1	2	3	4
1	1	√6	2√2	√2
2	√6	1	2	2
3	2√2	2	1	2√2
4	√2	2	2√2	1

The nearest!

2nd nearest!

Id	Young	Hair	Fashion	Tired
1	young	bold	t shirts	not tired
2	young	black	suits	tired
3	not young	white	suits	tired
4	young	black	t shirts	not tired

The nearest!

2nd nearest

Show the Result in a form of Dendrogram

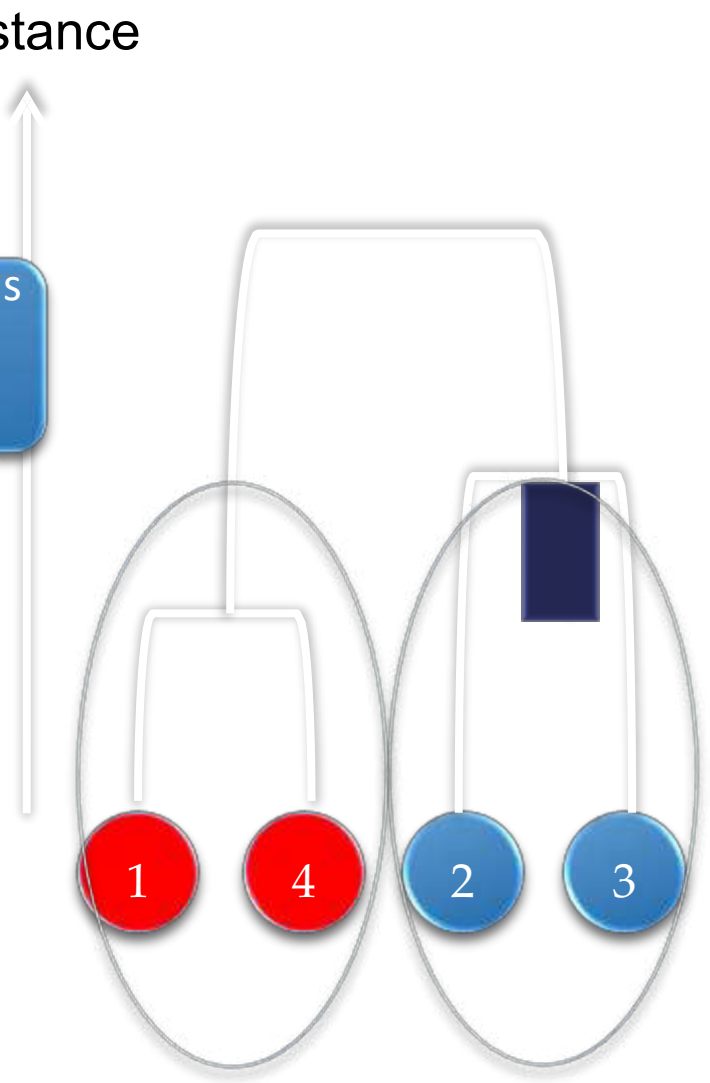
↓ Distance Matrix

	1	2	3	4
1	1	√6	2√2	√2
2	√6	1	2	2
3	2√2	**2**	1	2√2
4	**√2**	2	2√2	1

Use **furthest** one as distance between clusters

↓ Gathered Data

Id	Young	Hair	Fashion	Tired
1	young	bold	t shirts	not tired
2	young	black	suits	tired
3	not young	white	suits	tired
4	young	black	t shirts	not tired



Name clusters

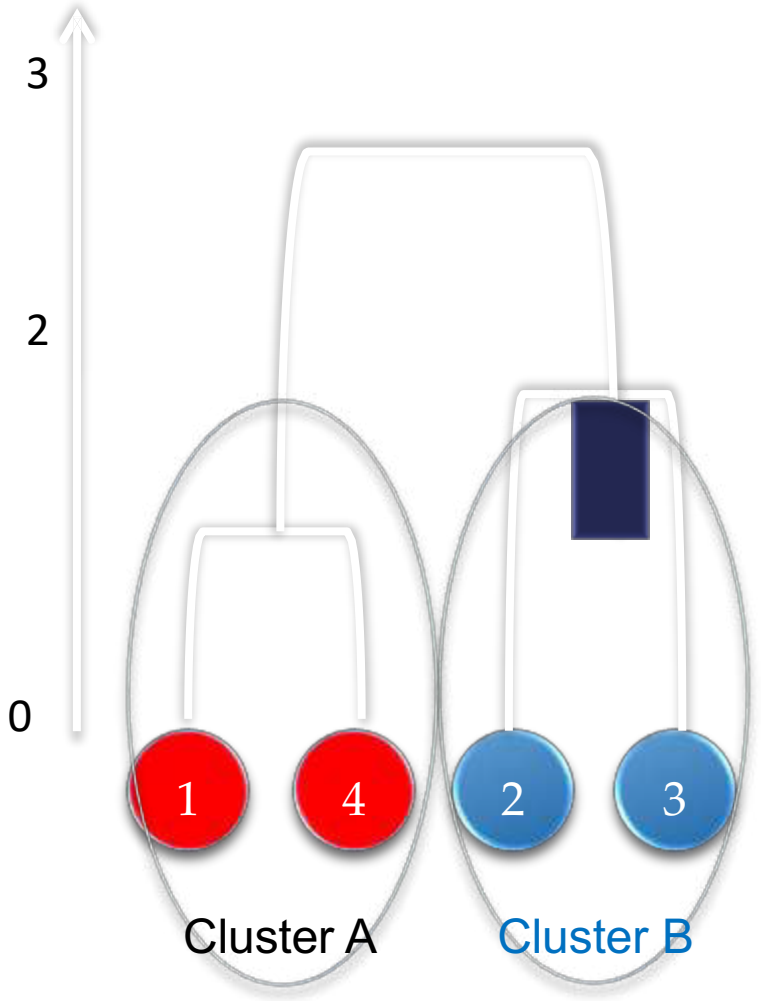
↓ Gathered Data

Id	Young	Hair	Fashion	Tired
1	young	bold	t shirts	not tired
2	young	black	suits	tired
3	not young	white	suits	tired
4	young	black	t shirts	not tired

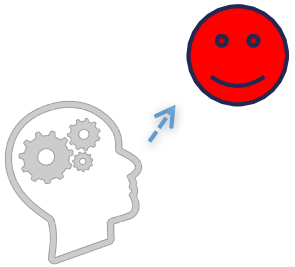
Cluster A 1 4 Not tired young cluster?

Cluster B 2 3 Tired suits cluster?

Are there better names?



Exercise 2



Not tired, not young, black hair, suits

- Make a new distance matrix and a new dendrogram
- Name each cluster
- What cluster will this person belong to?

Show your workings.

Thank you