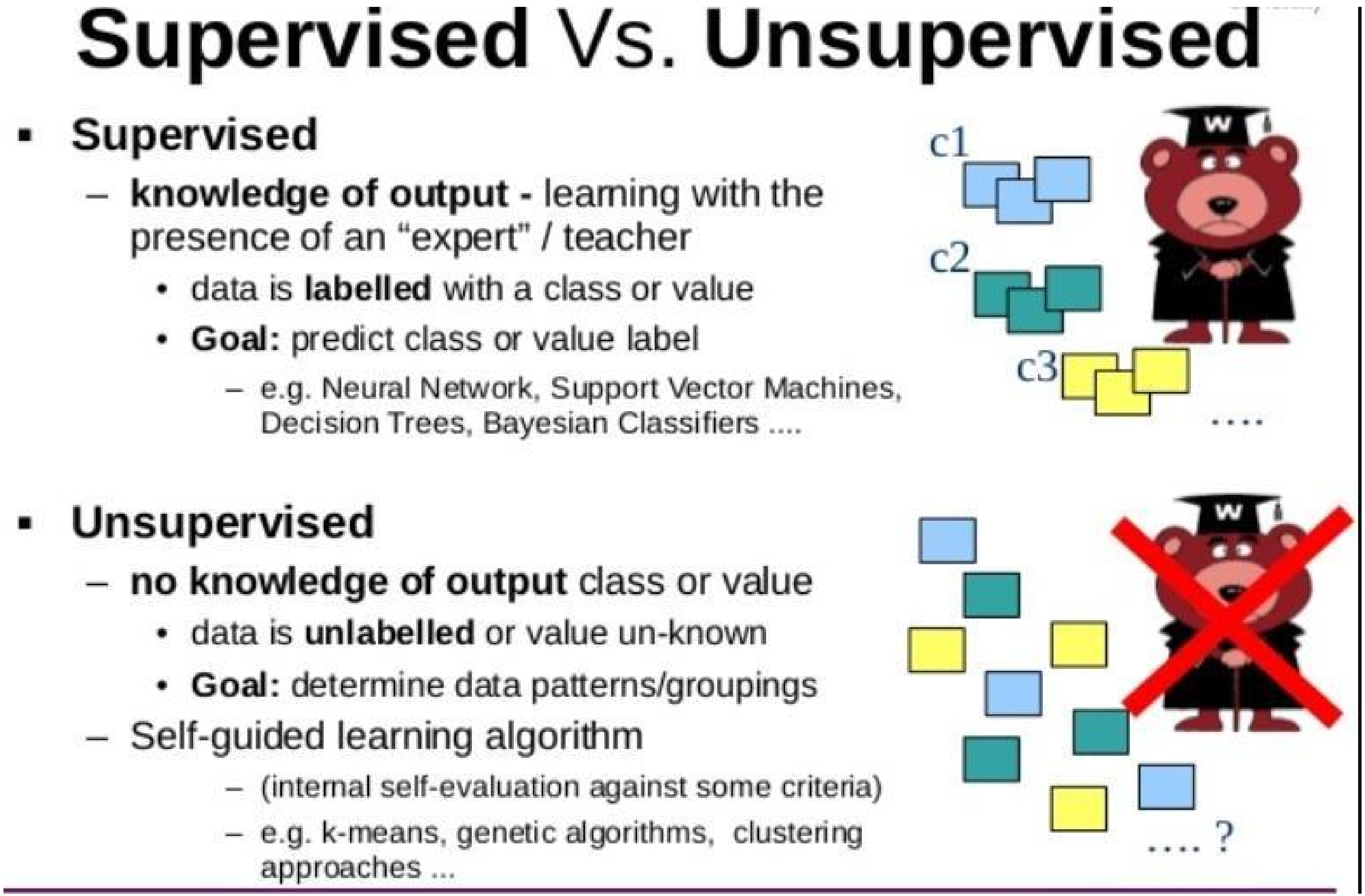
Artificial Intelligence

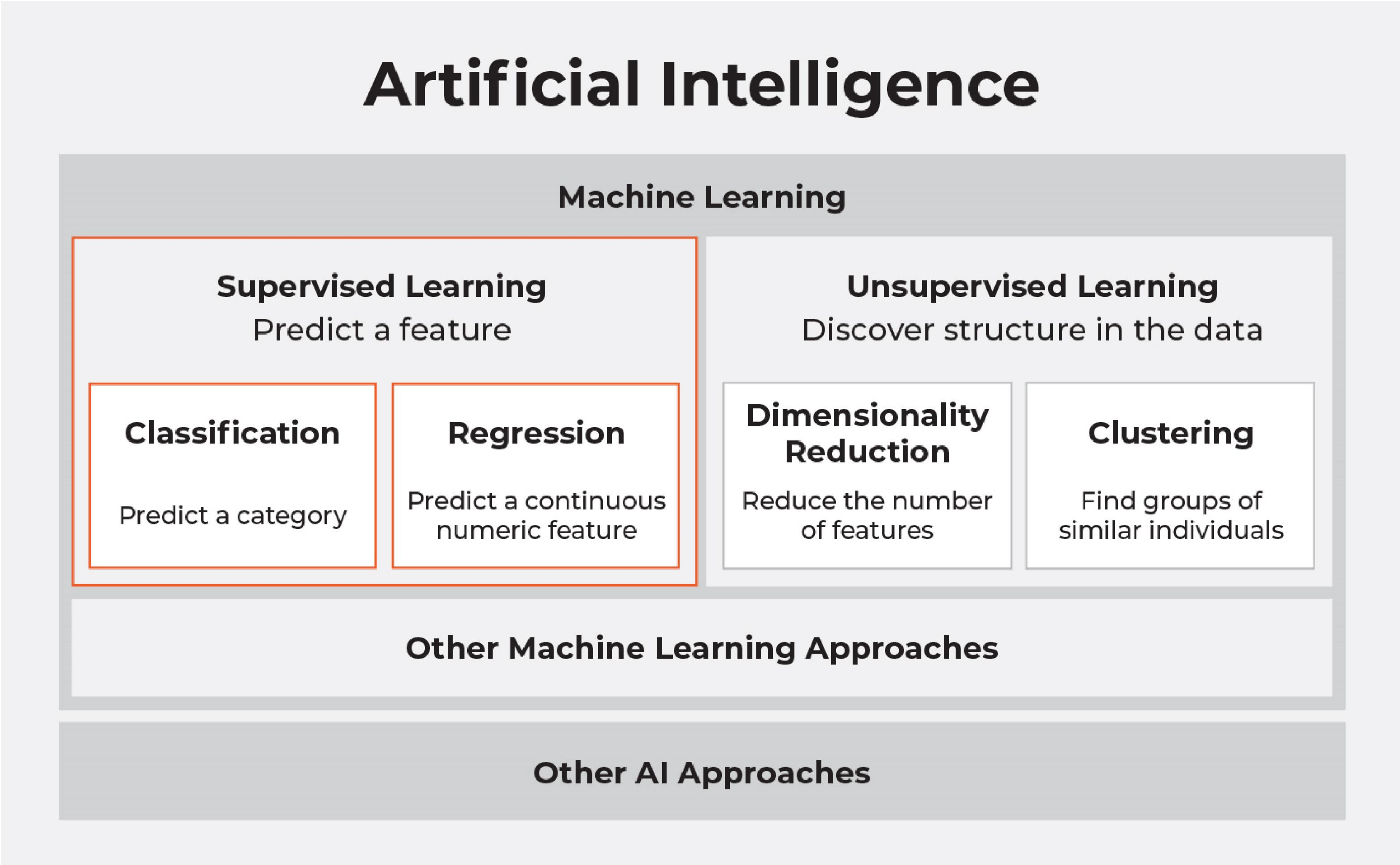
Classification: Naïve Bayes

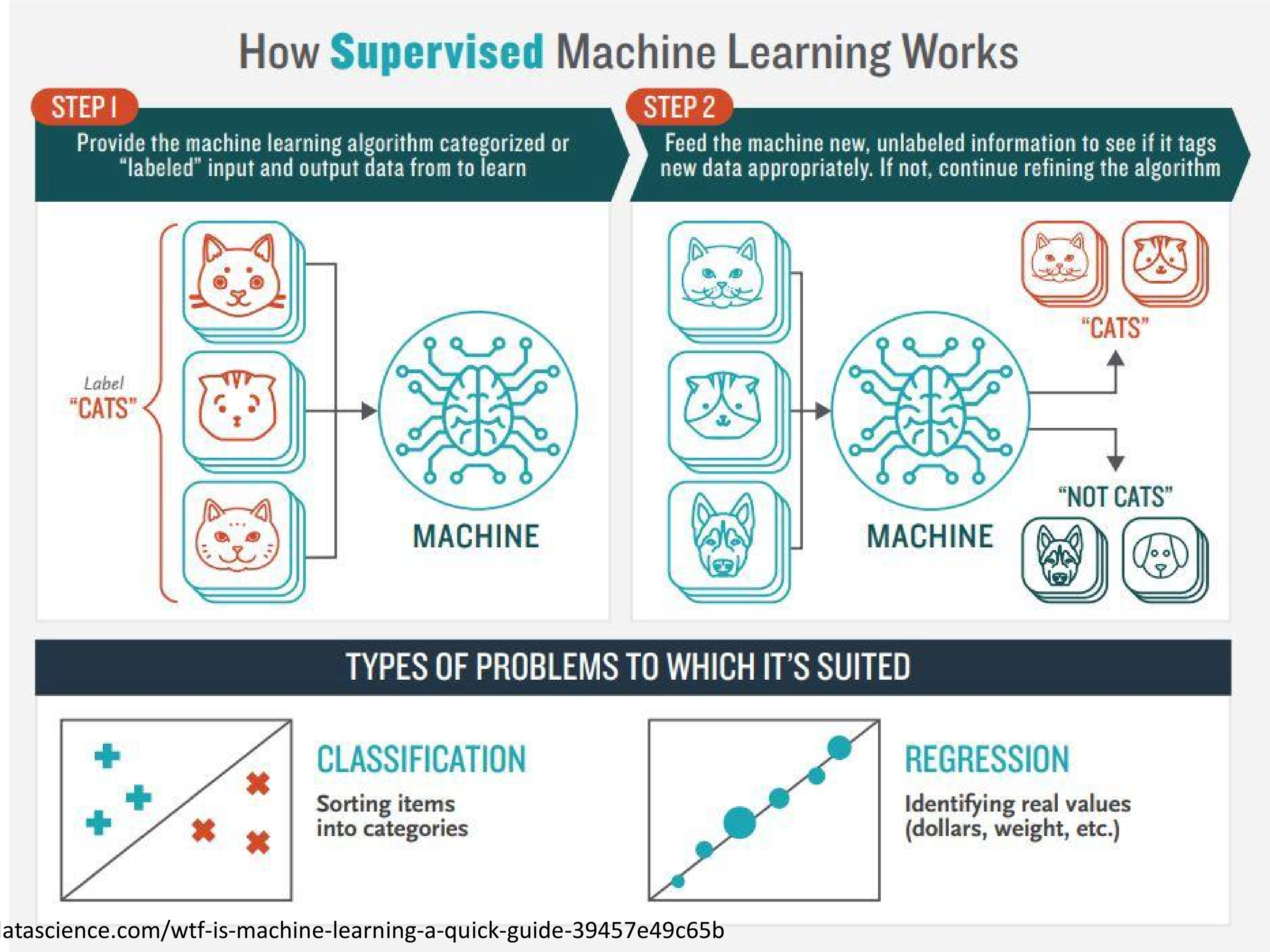
tita@pens.ac.id

Contents



https://www.researchgate.net/figure/Supervised-VS-Unsupervised-Learning_fig6_323108787


Artificial Intelligence
Machine Learning
Supervised Learning
Predict a feature
Classification
Predict a category
Regression
Predict a continuous numeric feature
Unsupervised Learning
Discover structure in the data
Dimensionality Reduction
Reduce the number of features
Clustering
Find groups of similar individuals
Other Machine Learning Approaches
Other AI Approaches



https://towardsdatascience.com/wtf-is-machine-learning-a-quick-guide-39457e49c65b

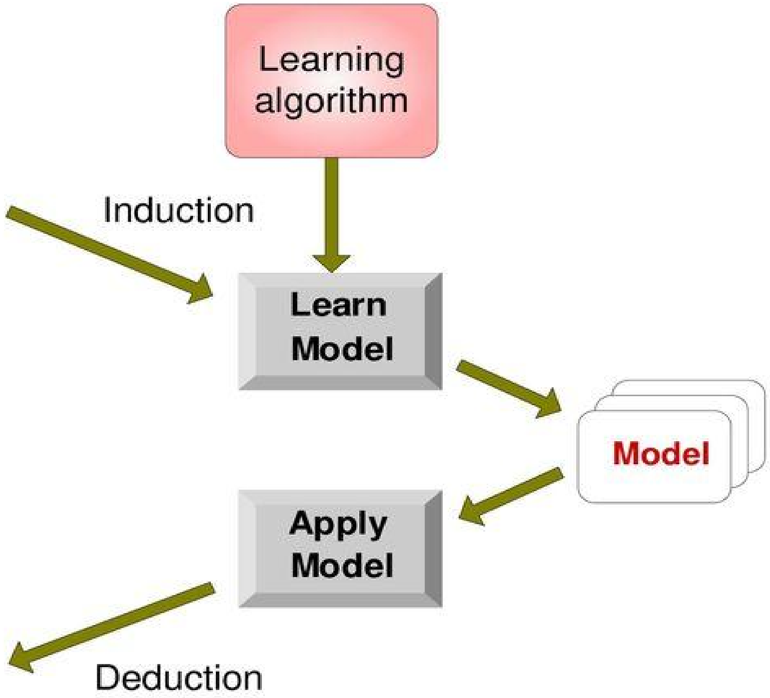
Illustration of classification

https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623

Illustrating Classification Task

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

Algoritma Naive Bayes

- **Algoritma Naive Bayes** merupakan sebuah metoda klasifikasi menggunakan metode probabilitas dan statistic.
- **Algoritma Naive Bayes** memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya.
- Ciri utama dari Naïve Bayes Classifier adalah asumsi yang sangat kuat (naïf) akan independensi dari masing-masing kondisi /atribut.

Probability Basics

We have two six-sided dice. When they are tolled, it could end up with the following occurrence:

(A) dice 1 lands on side "3",

(B) dice 2 lands on side "1", and

(C) Two dice sum to eight.

Answer the following questions:

1) $P(A) = ?$

2) $P(B) = ?$

3) $P(C) = ?$

4) $P(A \mid B) = ?$

5) $P(C \mid A) = ?$

6) $P(A, B) = ?$

7) $P(A, C) = ?$

8) Is $P(A, C)$ equals $P(A) * P(C)$?

A very simple dataset – one field / one class (1)

P34 level	Prostate cancer
High	Y
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	Y
High	N
Low	N
Medium	Y

A very simple dataset – one field / one class (2)

- A new patient has a blood test – his **P34 level is HIGH**.
- What is our best guess for prostate cancer?

P(cancer = Y)

P34 level	Prostate cancer
High	Y
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	Y
High	N
Low	N
Medium	Y

A very simple dataset – one field / one class (3)

- It's useful to know:

  P(cancer = Y)

- On basis of this tiny dataset,

  P(c = Y) is 5/10 = 0.5

P34 level	Prostate cancer
High	Y
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	Y
High	N
Low	N
Medium	Y

A very simple dataset – one field / one class (4)

- It's useful to know:
  P(cancer = Y)
- On basis of this tiny dataset,
  P(c = Y) is 5/10 = 0.5

P34 level	Prostate cancer
High	Y
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	Y
High	N
Low	N
Medium	Y

So, with **no other info** you'd expect P(cancer=Y) to be 0.5

A very simple dataset – one field / one class (5)

- But we know that P34 =H, so actually we want:
  ***P*(cancer=Y | P34 = H)**
- The probability that cancer is Y, ***given that* P34 is high**

P34 level	Prostate cancer
High	Y
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	Y
High	N
Low	N
Medium	Y

A very simple dataset – one field / one class (6)

***P*(cancer=Y | P34 = H)**

- The probability that cancer is Y, ***given that* P34 is high**
- This seems to be
  2/3 =~ 0.67

P34 level	Prostate cancer
High	Y
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	Y
High	N
Low	N
Medium	Y

A very simple dataset – one field / one class (7)

So we have:

$P(c=Y \mid \mathbf{P34 = H}) = 2/3 = 0.67$

$P(c=N \mid \mathbf{P34 = H}) = 1/3 = 0.33$

The class value with the highest probability is our best guess

P34 level	Prostate cancer
High	**Y**
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	**Y**
High	**N**
Low	N
Medium	Y

In general we may have any number of class values

Suppose again we know that P34 is High;
Here we have:

P(c=Y | P34=H) = 2/4 = 0.5
P(c=N | P34=H) = 1/4 = 0.25
P(c=Maybe | P34=H) = 1/4 = 0.25

... and again, Y is the winner

P34 level	Prostate cancer
High	**Y**
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	**Y**
High	N
High	**Maybe**
Medium	Y

That is the essence of Naive Bayes

but:

the probability calculations are much trickier when there are >1 fields

so we make a ‘Naive’ assumption that makes it simpler

Bayes' theorem

As we saw, on the right we are illustrating:

$$P(\text{cancer} = \text{Y} \mid \mathbf{P34 = H}) = 2/3 = 0.67$$

P34 level	Prostate cancer
High	Y
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	Y
High	N
Low	N
Medium	Y

Bayes' theorem

And now we are illustrating

$$P(\text{P34} = \text{H} \mid \textbf{cancer} = \textbf{Y})$$

This is a different thing, that turns out as 2/5 = 0.4

P34 level	Prostate cancer
High	Y
Medium	Y
Low	Y
Low	N
Low	N
Medium	N
High	Y
High	N
Low	N
Medium	Y

Bayes, Thomas (1763) An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London,* **53:370-418**

Bayes' theorem is this:

Likelihood

Class Prior probability

$$P(c \mid x) = \frac{P(x|c)\, P(c)}{P(x)}$$

Posterior probability

Predictor Prior probability

- $P(c|x)$ is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*).
- $P(c)$ is the prior probability of *class*.
- $P(x|c)$ is the likelihood which is the probability of *predictor* given *class*.
- $P(x)$ is the prior probability of *predictor*.

Bayes' theorem for multiple attributes:

Likelihood

Class Prior probability

$$P(Y|X_1,\ldots,X_n) = \frac{P(X_1,\ldots,X_n|Y)P(Y)}{P(X_1,\ldots,X_n)}$$

Posterior probability

Predictor Prior probability

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \cdots \times P(x_n|c) \times P(c)$$

"The posterior probability equals the prior probability times the likelihood ratio."

Bayes' theorem in 1-non-class-field context:

$$P(\text{Class=X} \mid \text{Fieldval=F}) = \frac{P(\text{Fieldval=F} \mid \text{Class=X}) \times P(\text{Class=X})}{P(\text{Fieldval=F})}$$

We want to check this for each class and choose the class that gives the highest value.

… and that was *Exactly* how we do Naïve Bayes for a 1-field dataset

Naïve-Bayes with Many-fields

P34 level	P61 level	BMI	Prostate cancer
High	Low	Medium	Y
Medium	Low	Medium	Y
Low	Low	High	Y
Low	High	Low	N
Low	Low	Low	N
Medium	Medium	Low	N
High	Low	Medium	Y
High	Medium	Low	N
Low	Low	High	N
Medium	High	High	Y

Naïve-Bayes with Many-fields

New patient:
P34=M, P61=M, BMI = H

Best guess at cancer field ?

P34 level	P61 level	BMI	Prostate cancer
High	Low	Medium	Y
Medium	Low	Medium	Y
Low	Low	High	Y
Low	High	Low	N
Low	Low	Low	N
Medium	Medium	Low	N
High	Low	Medium	Y
High	Medium	Low	N
Low	Low	High	N
Medium	High	High	Y

Naïve-Bayes with Many-fields

New patient:
P34=M, P61=M, BMI = H

Best guess at cancer field ?

which of these gives the highest value?

P34 level	P61 level	BMI	Prostate cancer
High	Low	Medium	Y
Medium	Low	Medium	Y
Low	Low	High	Y
Low	High	Low	N
Low	Low	Low	N
Medium	Medium	Low	N
High	Low	Medium	Y
High	Medium	Low	N
Low	Low	High	N
Medium	High	High	Y

$$P(\text{p34=M} \mid \text{Y}) \times P(\text{p61=M} \mid \text{Y}) \times P(\text{BMI=H} \mid \text{Y}) \times P(\text{cancer} = \text{Y})$$
$$P(\text{p34=M} \mid \text{N}) \times P(\text{p61=M} \mid \text{N}) \times P(\text{BMI=H} \mid \text{N}) \times P(\text{cancer} = \text{N})$$

Naïve-Bayes with Many-fields

New patient:
P34=M, P61=M, BMI = H

Best guess at cancer field ?

which of these gives the highest value?

P34 level	P61 level	BMI	Prostate cancer
High	Low	Medium	Y
Medium	Low	Medium	Y
Low	Low	High	Y
Low	High	Low	N
Low	Low	Low	N
Medium	Medium	Low	N
High	Low	Medium	Y
High	Medium	Low	N
Low	Low	High	N
Medium	High	High	Y

$\boldsymbol{P(\text{p34=M} \mid \text{Y})} \times P(\text{p61=M} \mid \text{Y}) \times P(\text{BMI=H} \mid \text{Y}) \times P(\text{cancer} = \text{Y})$

$P(\text{p34=M} \mid \text{N}) \times P(\text{p61=M} \mid \text{N}) \times P(\text{BMI=H} \mid \text{N}) \times P(\text{cancer} = \text{N})$

Naïve-Bayes with Many-fields

New patient:
P34=M, P61=M, BMI = H

Best guess at cancer field ?

which of these gives the highest value?

P34 level	P61 level	BMI	Prostate cancer
High	Low	Medium	Y
Medium	Low	Medium	Y
Low	Low	High	Y
Low	High	Low	N
Low	Low	Low	N
Medium	Medium	Low	N
High	Low	Medium	Y
High	Medium	Low	N
Low	Low	High	N
Medium	High	High	Y

P(p34=M | Y) × ***P*(p61=M | Y)** × P(BMI=H |Y) × P(cancer = Y)
P(p34=M | N) × P(p61=M | N) × P(BMI=H |N) × P(cancer = N)

Naïve -Bayes with Many-fields

New patient:
P34=M, P61=M, BMI = H

Best guess at cancer field ?

which of these gives the highest value?

P34 level	P61 level	BMI	Prostate cancer
High	Low	Medium	Y
Medium	Low	Medium	Y
Low	Low	High	Y
Low	High	Low	N
Low	Low	Low	N
Medium	Medium	Low	N
High	Low	Medium	Y
High	Medium	Low	N
Low	Low	High	N
Medium	High	High	Y

$P(\text{p34=M} \mid \text{Y}) \times P(\text{p61=M} \mid \text{Y}) \times \boldsymbol{P(\text{BMI=H} \mid \text{Y})} \times P(\text{cancer} = \text{Y})$

$P(\text{p34=M} \mid \text{N}) \times P(\text{p61=M} \mid \text{N}) \times P(\text{BMI=H} \mid \text{N}) \times P(\text{cancer} = \text{N})$

Naïve-Bayes with Many-fields

New patient:
P34=M, P61=M, BMI = H

Best guess at cancer field ?

which of these gives the highest value?

P34 level	P61 level	BMI	Prostate cancer
High	Low	Medium	Y
Medium	Low	Medium	Y
Low	Low	High	Y
Low	High	Low	N
Low	Low	Low	N
Medium	Medium	Low	N
High	Low	Medium	Y
High	Medium	Low	N
Low	Low	High	N
Medium	High	High	Y

$P(\text{p34=M} \mid \text{Y}) \times P(\text{p61=M} \mid \text{Y}) \times P(\text{BMI=H} \mid \text{Y}) \times \boldsymbol{P(\text{cancer} = \text{Y})}$

$P(\text{p34=M} \mid \text{N}) \times P(\text{p61=M} \mid \text{N}) \times P(\text{BMI=H} \mid \text{N}) \times P(\text{cancer} = \text{N})$

Naïve-Bayes with Many-fields

New patient:
P34=M, P61=M, BMI = H

Best guess at cancer field ?

which of these gives the highest value?

P34 level	P61 level	BMI	Prostate cancer
High	Low	Medium	Y
Medium	Low	Medium	Y
Low	Low	High	Y
Low	High	Low	N
Low	Low	Low	N
Medium	Medium	Low	N
High	Low	Medium	Y
High	Medium	Low	N
Low	Low	High	N
Medium	High	High	Y

$0.4 \times 0 \times 0.4 \times 0.5 = 0$

$0.2 \times 0.4 \times 0.2 \times 0.5 = 0.008$

Given the fact P(Yes|x′) < P(No|x′), we label x′ to be "No".

Tennis Example

- Example: Play Tennis

PlayTennis: training examples

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

The **learning phase** for tennis example

PlayTennis: training examples

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

$P(\text{Play}=Yes) = 9/14$

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

We have four variables, we calculate for each we calculate the conditional probability table

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

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

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

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

The *test phase* for the tennis example

- Test Phase
  - Given a new instance of variable values,
    x′=(Outlook=*Sunny*, Temperature=*Cool*, Humidity=*High*, Wind=*Strong*)
  - Given calculated Look up tables

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

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

  - **To calculate Yes or No**

P(*Yes*|x′): [P(*Sunny*|Yes)P(*Cool*|Yes)P(*High*|Yes)P(*Strong*|Yes)]P(Play=*Yes*) = 0.0053
P(*No*|x′): [P(*Sunny*|No) P(*Cool*|No)P(*High*|No)P(*Strong*|No)]P(Play=*No*) = 0.0206

Given the fact P(*Yes*|x′) < P(*No*|x′), we label x′ to be "*No*".

4 Applications of Naive Bayes Algorithms

- **Real time Prediction:** Naive Bayes is an eager learning classifier and it is sure fast. Thus, it could be used for making predictions in real time.
- **Multi class Prediction:** This algorithm is also well known for multi class prediction feature. Here we can predict the probability of multiple classes of target variable.
- **Text classification/ Spam Filtering/ Sentiment Analysis:** Naive Bayes classifiers mostly used in text classification (due to better result in multi class problems and independence rule) have higher success rate as compared to other algorithms. As a result, it is widely used in Spam filtering (identify spam e-mail) and Sentiment Analysis (in social media analysis, to identify positive and negative customer sentiments)
- **Recommendation System:** Naive Bayes Classifier and Collaborative Filtering together builds a Recommendation System that uses machine learning and data mining techniques to filter unseen information and predict whether a user would like a given resource or not

Conclusions

- Naïve Bayes based on the independence assumption
- Training is very easy and fast
  - just requiring considering each attribute in each class separately
- Test is straightforward;
  - just looking up tables or calculating conditional probabilities with normal distributions

References

- David Corne, Data Mining and Machine Learning, Naive Bayes, Heriot Watt University, http://www.macs.hw.ac.uk/~dwcorne/Teaching/DMML/DMML-NB.ppt
- Marek Andrzej Perkowski, Naïve Bayes Classifier, http://web.cecs.pdx.edu/~mperkows, Portland State University, 2014.
- Sunil Ray, 6 Easy Steps to Learn Naive Bayes Algorithm with codes in Python and R, https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/, 2017