

MS4S10 Machine Learning and Decision Making

Week 3

Moizzah Asif

moizzah.asif@southwales.ac.uk

J418

1

Know thy module



Week 1 – 4
Moizzah Asif

- 27-11-2020 – Basics of Machine Learning; The machine learning process; Data collection & preprocessing
- 04-12-2020 – **Supervised Learning:** classification, regression, optimisation, model selection and generalisation, parametric and non-parametric learning, Decision Trees
- 11-12-2020 – **Supervised Learning:** Probabilistic learning, Bayes Learning, Naïve Bayes classifiers, Ensemble Learning, Random Forest, Support Vector Machines, Kernel functions, Hyper-parameter optimisation
- 08-01-2021 –

Course work 1 – 50% weightage

2

Probabilistic Learning

Certain learning problems can be best approached with probabilistic hypothesis formed by learning algorithms, such as: Naïve Bayes.

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 - i. Data Preparation
 - ii. Algorithm Selection

In ML, often the goal is to determine the best h , from H using the training data D .

One way to do that is to find the *most probable h given D* and any initial knowledge about prior probabilities of various other h in H

3

Probabilistic Learning

Bayes Theorem

Bayes theorem provides to calculate the probability of a hypothesis based on:

1. its prior probability,
2. The probabilities of observing various data given the hypothesis, and the data itself

Some notations

$P(h)$ – initial probability that h holds true also called prior probability of h

$P(D)$ – the probability that training data D will be observed

4

Probabilistic Learning

Some notations

$P(h)$ – initial probability that h holds true also called prior probability of h

$P(D)$ – the probability that training data D will be observed

$P(D|h)$ – probability of observing D , given h holds true

The algorithm wants to find $P(h|D)$

$P(h|D)$ - the posterior probability of h

It reflects

- the confidence that h holds after D has been observed
- The influence of the D , in contrast to the independent prior probability $P(h)$

5

Probabilistic Learning

Bayes theorem

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

Posterior probability increases with:

- $P(h)$
- $P(D|h)$

and decreases with:

- $P(D)$ - (the more probable it becomes that D can be observed independent of h , the less evidence it provides in support of h)

6

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes Classifier

Naïve Bayes classifier is one of the practical Bayesian learning method.

Applicable to tasks where:

- each training/new instance x is described by a conjunction of feature/attribute values, and
- assumes a target value from a set of possible outcomes.

Naïve Bayes classifies the new instance by assigning the most probable target value given the attribute values of that instance.

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Molizah Asif - Machine Learning and Decision Making © University of South Wales

7


University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes Classifier

There are two parts to this algorithm:

- Naïve - Assumes independence among predictors.



- Bayes – Applies Bayes theorem

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Molizah Asif - Machine Learning and Decision Making © University of South Wales

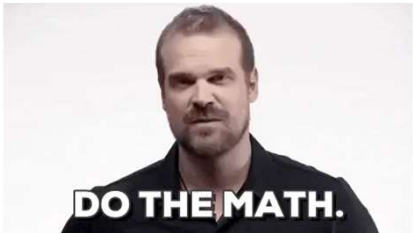
8

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier

The Bayesian approach classifies a new instance by assigning the most probable target value given the attribute values.



1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Molizah Asif - Machine Learning and Decision Making © University of South Wales

9

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier

The most probably target value is assigned using the most probable $h \in H$ given the observed D .

Such maximally probable h is called a *maximum a posteriori* (MAP) hypothesis

$$h_{MAP} \equiv \operatorname{argmax}_{h \in H} P(h|D)$$

Using Bayes theorem

$$\begin{aligned} h_{MAP} &= \operatorname{argmax}_{h \in H} \frac{P(D|h) P(h)}{P(D)} \\ &= \operatorname{argmax}_{h \in H} P(D|h) (Ph) \end{aligned}$$

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Molizah Asif - Machine Learning and Decision Making © University of South Wales

10

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier

The most probably target value is assigned using the most probable $h \in H$ given the observed D .

So,

If most probable target value is represented by $v_{MAP} \in V$, and

Attribute values are $\langle a_1, a_2, \dots, a_n \rangle$

$$v_{MAP} \equiv \operatorname{argmax}_{v_j \in V} P(v_j | a_1, a_2, \dots, a_n)$$

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Molizah Asif - Machine Learning and Decision Making © University of South Wales

11

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier

The most probably target value is assigned using the most probable $h \in H$ given the observed D .

So,

If most probable target value is represented by $v_{MAP} \in V$, and

Attribute values are $\langle a_1, a_2, \dots, a_n \rangle$

$$v_{MAP} \equiv \operatorname{argmax}_{v_j \in V} P(v_j | a_1, a_2, \dots, a_n)$$

Using Bayes theorem

$$\begin{aligned} v_{MAP} &= \operatorname{argmax}_{h \in H} \frac{P(a_1, a_2, \dots, a_n | v_j) P(v_j)}{P(a_1, a_2, \dots, a_n)} \\ &= \operatorname{argmax}_{h \in H} P(a_1, a_2, \dots, a_n | v_j) P(v_j) \end{aligned}$$

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Molizah Asif - Machine Learning and Decision Making © University of South Wales

12

University of
South Wales
Prifysgol
De Cymru

Probabilistic Learning

Naïve Bayes classifier

The most probably target value is assigned using the most probable $h \in H$ given the observed D .

Using Bayes theorem

$$v_{MAP} = \underset{h \in H}{\operatorname{argmax}} \frac{P(a_1, a_2, \dots, a_n | v_j) P(v_j)}{P(a_1, a_2, \dots, a_n)}$$

$$= \underset{h \in H}{\operatorname{argmax}} P(a_1, a_2, \dots, a_n | v_j) P(v_j)$$

Estimating $P(v_j)$ is simple

$P(a_1, a_2, \dots, a_n)$ is not. Why?

It is not feasible unless we have a huge training dataset, WHY?

The problem is: the number of these terms is equal to the number of possible instances times the number of possible target values.

Therefore, have to see every instance in the instance space many times in order to obtain reliable estimates

13
Molizah Asif - Machine Learning and Decision Making
© University of South Wales

13

University of
South Wales
Prifysgol
De Cymru

Probabilistic Learning

Naïve Bayes classifier

The most probably target value is assigned using the most probable $h \in H$ given the observed D .

Using Bayes theorem


$$v_{MAP} = \underset{h \in H}{\operatorname{argmax}} \frac{P(a_1, a_2, \dots, a_n | v_j) P(v_j)}{P(a_1, a_2, \dots, a_n)}$$

$$= \underset{h \in H}{\operatorname{argmax}} P(a_1, a_2, \dots, a_n | v_j) P(v_j)$$

Estimating $P(v_j)$ is simple

$P(a_1, a_2, \dots, a_n)$ is not. Why?

That's where Naïve Bayes classifier comes to the rescue



14
Molizah Asif - Machine Learning and Decision Making
© University of South Wales

14

University of
South Wales
Prifysgol
De Cymru

Probabilistic Learning

Naïve Bayes classifier

The most probably target value is assigned using the most probable $h \in H$ given the observed D .

Using Bayes theorem

$$v_{MAP} = \underset{h \in H}{\operatorname{argmax}} \frac{P(a_1, a_2, \dots, a_n | v_j) P(v_j)}{P(a_1, a_2, \dots, a_n)}$$

$$= \underset{h \in H}{\operatorname{argmax}} P(a_1, a_2, \dots, a_n | v_j) P(v_j)$$

Estimating $P(v_j)$ is simple

$P(a_1, a_2, \dots, a_n)$ is not. Why?

Naïve Bayes is based on simplifying assumption that:

The attribute values are conditionally independent given the target value

15
Molizah Asif - Machine Learning and Decision Making
© University of South Wales

15

University of
South Wales
Prifysgol
De Cymru

Probabilistic Learning


Naïve Bayes classifier

Naïve Bayes is based on simplifying assumption that:

The attribute values are conditionally independent given the target value

In **simpler** words:

Given the target value of the instance, the probability of observing the conjunction a_1, a_2, \dots, a_n is just the product of the probabilities for the individual attributes.



16
Molizah Asif - Machine Learning and Decision Making
© University of South Wales

16

University of
South Wales
Prifysgol
De Cymru

Probabilistic Learning

Naïve Bayes classifier

Naïve Bayes is based on simplifying assumption that:

The attribute values are conditionally independent given the target value

In **simpler** words:

Given the target value of the instance, the probability of observing the conjunction a_1, a_2, \dots, a_n is just the product of the probabilities for the individual attributes.

↓

$$P(a_1, a_2, \dots, a_n | v_j) = \prod_i P(a_i | v_j)$$

17
Molizah Asif - Machine Learning and Decision Making
© University of South Wales

17

University of
South Wales
Prifysgol
De Cymru

Probabilistic Learning

Naïve Bayes classifier

Naïve Bayes is based on simplifying assumption that:

The attribute values are conditionally independent given the target value

$$v_{NB} = \underset{v_j \in V}{\operatorname{argmax}} P(v_j) \prod_i P(a_i | v_j)$$

Where v_{NB} denotes the target value output by the naïve Bayes classifier

18
Molizah Asif - Machine Learning and Decision Making
© University of South Wales

18

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier

The number of distinct $P(a_i|v_j)$ terms to be estimated from the training data is:

just the number of distinct attribute values times the number distinct target values

This is a much smaller number!

19 Moizah Asif - Machine Learning and Decision Making © University of South Wales

19

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier

The weather example again:

outlook = sunny, humidity = high, temperature = cool, wind = strong

Includes 9 positive and 5 negative examples [9+, 5-]

$$v_{NB} = \underset{v_j \in \{yes, no\}}{\operatorname{argmax}} P(v_j) \prod_i P(a_i|v_j)$$

$$v_{NB} = \underset{v_j \in \{yes, no\}}{\operatorname{argmax}} P(v_j) P(outlook = sunny|v_j) P(temperature = cool|v_j) P(humidity = high|v_j) P(wind = strong|v_j)$$

20 Moizah Asif - Machine Learning and Decision Making © University of South Wales

20

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier

The weather example again:

outlook = sunny, humidity = high, temperature = cool, wind = strong

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

21 Moizah Asif - Machine Learning and Decision Making © University of South Wales

21

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier – Bernoulli naïve Bayes

Can be applied when:

- target variable is binary - $v \in \{0,1\}$
- Predictor variables are discrete and binary - $x \in \{0,1\}^K$, where $|X| = K$

Possible application?

22 Moizah Asif - Machine Learning and Decision Making © University of South Wales

22

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier – Multinomial naïve Bayes

Can be applied when:

- Predictor variables are discrete and can assume more than two values

Possible application?

23 Moizah Asif - Machine Learning and Decision Making © University of South Wales

23

University of South Wales Prifysgol De Cymru

Probabilistic Learning

Naïve Bayes classifier – Gaussian naïve Bayes

Assumes features follow a normal distribution.

Can be applied when:

- Predictor variables are continuous

Possible application?

24 Moizah Asif - Machine Learning and Decision Making © University of South Wales

24

University of South Wales
Prifysgol De Cymru

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

Ensemble learning

Basic idea

Many learners, with slightly different results on the same dataset.

Put them together well enough, the generated results will be significantly better than individual results

How does it differ from cross validation?

25

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

25

University of South Wales
Prifysgol De Cymru

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

Ensemble learning

Boosting

A collection of weak learner, when put together to make an ensemble learner.

Pseudocode

1. Split the data into three
2. Train a classifier on one of thirds
3. Test it on another of the thirds
4. Extract the misclassified data from step 3 (test)
5. Extract equal and random instances of correctly classified data from 3

26

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

26

University of South Wales
Prifysgol De Cymru

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

Ensemble learning

Boosting

Pseudocode

1. Split the data into three
2. Train a classifier on one of thirds
3. Test it on another of the thirds
4. Extract the misclassified data from step 3 (test)
5. Extract equal and random instances of correctly classified data from 3
6. Merge 4 and 5 to create new dataset
7. Train a second classifier on the new dataset
8. Test classifier from 2 and 7 on the last third split of the thirds
9. Instances for which same output is produced from both classifiers are ignored, and add those with different outputs to a new dataset

27

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

27

University of South Wales
Prifysgol De Cymru

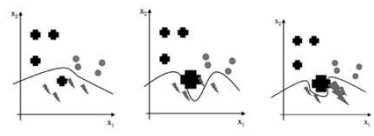
1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

Ensemble learning

AdaBoost – Adaptive Boost

Assigns weights to instances based on how difficult previous classifiers found classifying it.

The weights are fed as part of the input when training classifiers



28

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

28

University of South Wales
Prifysgol De Cymru

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

Ensemble learning

AdaBoost – conceptually

At each iteration, a new classifier is trained on the training set.

The weights are used as part of input at each iteration.

Initial weights are set to $\frac{1}{N}$, where N is the number of training instances.

At subsequent iteration, the error ϵ is computed as: sum of the weights of the misclassified instances.

For next iteration:
the weights of incorrectly classified examples are updated by multiplying with α (where $\alpha = \frac{1-\epsilon}{\epsilon}$)

29

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

29

University of South Wales
Prifysgol De Cymru

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

Ensemble learning

AdaBoost – conceptually

At each iteration, a new classifier is trained on the training set.

The weights are used as part of input at each iteration.

Initial weights are set to $\frac{1}{N}$, where N is the number of training instances.

At subsequent iteration, the error ϵ is computed as: sum of the weights of the misclassified instances.

For next iteration:
the weights of incorrectly classified examples are updated by multiplying with α (where $\alpha = \frac{1-\epsilon}{\epsilon}$)

the correctly classified examples weight remain the same

the whole set of weights is normalised so that it sums to 1

Training terminates after either:
a set number of iterations,
or until all training instances are classified correctly,
or one point carries more than half of the available weight

30

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

30

University of South Wales Prifysgol De Cymru

Ensemble learning

AdaBoost – Formally

- Given: N samples $\{x_n, y_n\}$, where $y_n \in \{+1, -1\}$, and some way of constructing weak (or base) classifiers
- Initialize weights $w_1(n) = \frac{1}{N}$ for every training sample
- For $t = 1$ to T
 - Train a weak classifier $h_t(x)$ using current weights $w_t(n)$, by minimising

$$\varepsilon_t = \sum_n w_t(n) I[y_n \neq h_t(x_n)]$$

where $I[y_n \neq h_t(x_n)] = 1$ if $h_t(x_n) \neq y_n$ or 0 otherwise
 - Compute contribution for this classifier:

$$\beta_t = \frac{1}{2} \log \frac{1 - \varepsilon_t}{\varepsilon_t}$$

31 Moizah Asif - Machine Learning and Decision Making © University of South Wales

31

University of South Wales Prifysgol De Cymru

Ensemble learning

AdaBoost – Formally

- Given: N samples $\{x_n, y_n\}$, where $y_n \in \{+1, -1\}$, and some way of constructing weak (or base) classifiers
- Initialize weights $w_1(n) = \frac{1}{N}$ for every training sample
- For $t = 1$ to T
 - Train a weak classifier $h_t(x)$ using current weights $w_t(n)$, by minimising

$$\varepsilon_t = \sum_n w_t(n) I[y_n \neq h_t(x_n)]$$

where $I[y_n \neq h_t(x_n)] = 1$ if $h_t(x_n) \neq y_n$ or 0 otherwise
 - Compute contribution for this classifier: $\beta_t = \frac{1}{2} \log \frac{1 - \varepsilon_t}{\varepsilon_t}$
 - Update weight on training points

$$w_{t+1}(n) := w_t(n) e^{-\beta_t y_n h_t(x)}$$

And normalise them such that $\sum_n w_{t+1}(n) = 1$
- Output the final classifier

$$h[x] = \text{sign} \left[\sum_{t=1}^T \beta_t h_t(x) \right]$$

32 Moizah Asif - Machine Learning and Decision Making © University of South Wales

32

University of South Wales Prifysgol De Cymru

Ensemble learning

Bagging

Another method for aggregating classifiers – bootstrap aggregating.


Bootstrap

A bootstrap sample is a sample taken from the original dataset with replacement, so that some data can be extracted several times and others not at all

The sample is the the same size as the original

Lots of sample are taken.

What's in it for learning??



33 Moizah Asif - Machine Learning and Decision Making © University of South Wales

33

University of South Wales Prifysgol De Cymru

Ensemble learning

Bagging

Having taken a set of bootstrap samples,

the bagging method simply requires that we fit a model to each dataset, and

then combine them by taking the output to be the majority vote of all the classifiers

34 Moizah Asif - Machine Learning and Decision Making © University of South Wales

34

University of South Wales Prifysgol De Cymru

Random Forest

The idea

if one tree is good then many are better (given there is enough variety)

Uses bagging (this adds randomness in one way)*

Randomness is also ensured by limiting the choices a decision tree in the forest can make

each node is provided with a random subset of feature and can only make choices from them**

Results in a new parameter – how many features to consider

empirically, random forest is not sensitive to this parameter

35 Moizah Asif - Machine Learning and Decision Making © University of South Wales

35

University of South Wales Prifysgol De Cymru

Random Forest

The idea - benefit of randomness

Gets rid of the pruning trees

Question then – How many trees to build?
can be built until the error stops decreasing

The output of the forest is the majority vote for the classification or the mean response for regression

36 Moizah Asif - Machine Learning and Decision Making © University of South Wales

36

University of South Wales Prifysgol De Cymru

Random Forest

Basic pseudo code

For each of N trees:

- create a new bootstrap sample of the training set
- use this bootstrap sample to train a decision tree

at each node of the decision tree,

- randomly select m features, and
- compute the information gain (or Gini impurity) only on that set of features, selecting the optimal one

repeat until the tree is complete

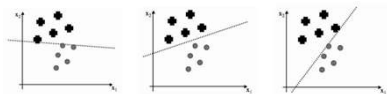
37 Molazzah Asif - Machine Learning and Decision Making © University of South Wales

37

University of South Wales Prifysgol De Cymru

Support Vector Machines

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection



All three of the lines that are drawn separate out the two classes, so in some sense they are 'correct'

Which one is the best and why?

38 Molazzah Asif - Machine Learning and Decision Making © University of South Wales

38

University of South Wales Prifysgol De Cymru

Support Vector Machines

Quantifying the characteristics of a good classification line

put a no man's land around the line
any point that lies in the no man's region has to be considered too close

Make this region symmetric about the line (i.e. mirror image from either side)
a cylinder in 3D?

The radius of the cylinder is measured from the line to the circumference (in 3D)/edge(2D) from either sides of the line

Let's call the largest possible radius, the Margin.

39 Molazzah Asif - Machine Learning and Decision Making © University of South Wales

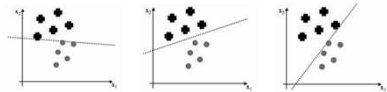
39

University of South Wales Prifysgol De Cymru

Support Vector Machines

Quantifying the characteristics of a good classification line

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection



So the line/classifier in the middle has largest margin

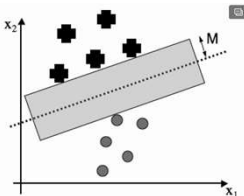
40 Molazzah Asif - Machine Learning and Decision Making © University of South Wales

40

University of South Wales Prifysgol De Cymru

Support Vector Machines

Quantifying the characteristics of a good classification line



The margin is the largest region that separates the classes without there being any points inside, where the box is made from two lines that are parallel to the decision boundary.

41 Molazzah Asif - Machine Learning and Decision Making © University of South Wales

41

University of South Wales Prifysgol De Cymru

Support Vector Machines

Support vectors

The datapoints in each class that lie closest to the classification line have a name as well. They are called **support vectors**.

What has been established so far

1. The best classifier has the largest margin
2. Support vector are the most important datapoints – Caution though – they may might be wrongly identified

42 Molazzah Asif - Machine Learning and Decision Making © University of South Wales

42


University of South Wales Prifysgol De Cymru

Support Vector Machines

Hence,

after training we can throw away all of the data except for the support vectors

Think of memory and data storage!



1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

43

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

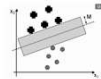
43

University of South Wales Prifysgol De Cymru

Support Vector Machines

The maths!

The equation of the classifier line
 $y = wx + b$
 where b is the bias weight/y intercept in 2D



So,
 Any value of x that gives y above the classifier, is classified the instance as $+$ class and 0 other wise

But what about the margin region?

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

44

Molizah Asif - Machine Learning and Decision Making

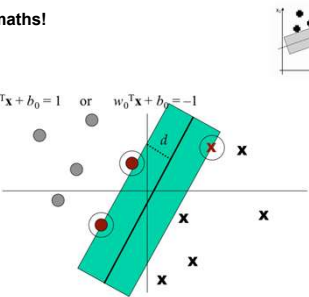
© University of South Wales

44

University of South Wales Prifysgol De Cymru

Support Vector Machines

The maths!



1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

45

Molizah Asif - Machine Learning and Decision Making

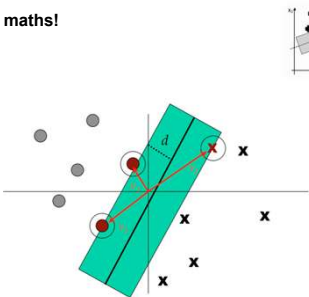
© University of South Wales

45

University of South Wales Prifysgol De Cymru

Support Vector Machines

The maths!



1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

46

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

46

University of South Wales Prifysgol De Cymru

Support Vector Machines

The maths!

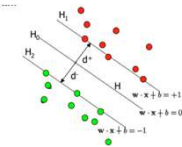
We need to find/define the hyperplanes such that:
 when $y_i = +1 \rightarrow wx_i + b \geq +1$
 when $y_i = -1 \rightarrow wx_i + b \leq -1$

So, if H_1 and H_2 are the planes, then there equations in 2D

$$H_1: wx_i + b = +1$$

$$H_2: wx_i + b = -1$$

And then the decision boundary's equation is
 $H_0: wx_i + b = 0$



1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

47

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

47

University of South Wales Prifysgol De Cymru

Support Vector Machines

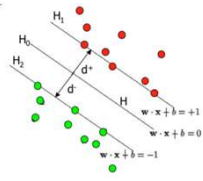
The maths!

The points on the planes H_1 and H_2 are the tips of the Support Vectors

d^+ = shortest distance to the closest positive point

d^- = shortest distance to the closest negative point

The margin (gutter) of a separating hyperplane is $d^+ + d^-$



1. Basics of Machine Learning (ML)
2. The Machine Learning Process
 i. Data Preparation
 ii. Algorithm Selection

48

Molizah Asif - Machine Learning and Decision Making

© University of South Wales

48

University of South Wales
Prifysgol De Cymru

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Support Vector Machines

The maths!

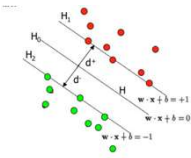
w = weight vector
 x = input vector
 b = bias term

So the decision hyper plane's equation can be re written as

$$H_0: w^T x + b = 0$$

And also

$$H_1: w^T x + b \geq 0 \text{ for } d_i = +1$$

$$H_2: w^T x + b \leq 0 \text{ for } d_i = -1$$


49 Moizah Asif - Machine Learning and Decision Making © University of South Wales

49

University of South Wales
Prifysgol De Cymru

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Support Vector Machines

The maths!

The distance from a point (x_0, y_0) to a line: $Ax + By + c = 0$

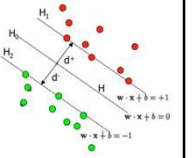
$$\frac{|Ax_0 + By_0 + c|}{\sqrt{A^2 + B^2}}, \text{ so,}$$

Distance between H_0 and H_1

$$= \frac{|wx + b|}{\|w\|}$$

$$= \frac{1}{\|w\|}$$

Then, the total distance b/w H_1 & H_2

$$\frac{2}{\|w\|}$$


50 Moizah Asif - Machine Learning and Decision Making © University of South Wales

50

University of South Wales
Prifysgol De Cymru

1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Support Vector Machines

The maths!

Then, the total distance b/w H_1 & H_2

$$\frac{2}{\|w\|}$$

So, to maximise the margin, minimise the $\|w\|$. Which can be done by posing it as a quadratic function:

$$\min f: \frac{\|w\|^2}{2}$$

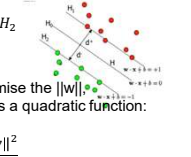
However there are constraints to finding the value:

- Making the margin as large as possible

$$x_i w + b \geq +1 \text{ when } y_i = +1$$

$$x_i w + b \leq -1 \text{ when } y_i = -1$$
 Or

$$y_i(x_i w) \geq 1$$
- Separate +v and -v classes



51 Moizah Asif - Machine Learning and Decision Making © University of South Wales

51

University of South Wales
Prifysgol De Cymru


1. Basics of Machine Learning (ML)
2. The Machine Learning Process
i. Data Preparation
ii. Algorithm Selection

Support Vector Machines

Let's get back to intuitive understanding

What about misclassification?

There is a parameter called regularisation parameter C .



Left: low regularization value, right: high regularization value

52 Moizah Asif - Machine Learning and Decision Making © University of South Wales

52