Machine Learning

Classification

Hiroshi Shimodaira and Hao Tang

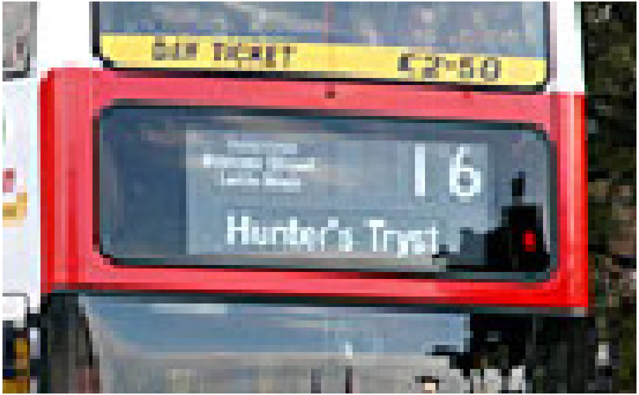
22, 26 January 2024

Ver. 1.0a

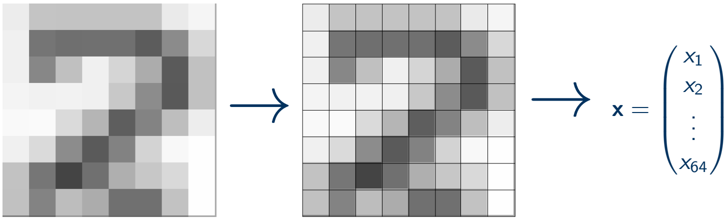
Topics - you should be able to explain after this week

- Data and data preprocessing
- Features and labels
- Linear classifiers
- Hyperplanes, decision boundaries, and decision regions
- Training of classifiers
- Loss and cost functions
- Logistic regression
- Extension of binary classification to multiclass classification
- Sigmoid and softmax functions

Image data



Pixel image to a feature vector



Turn each cell (pixel) into a number

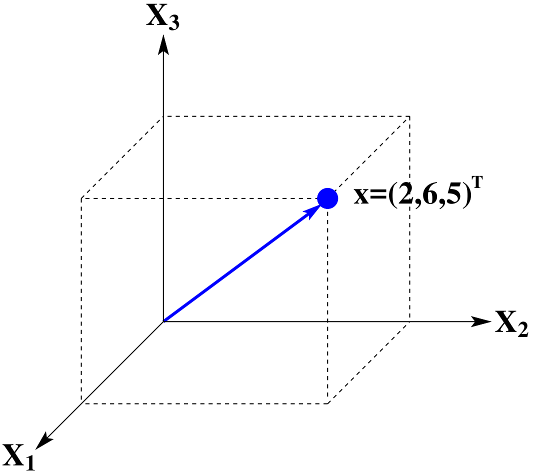
Unravel into a column vector, a feature vector

$\Rightarrow$ represented digit as point in $64D$

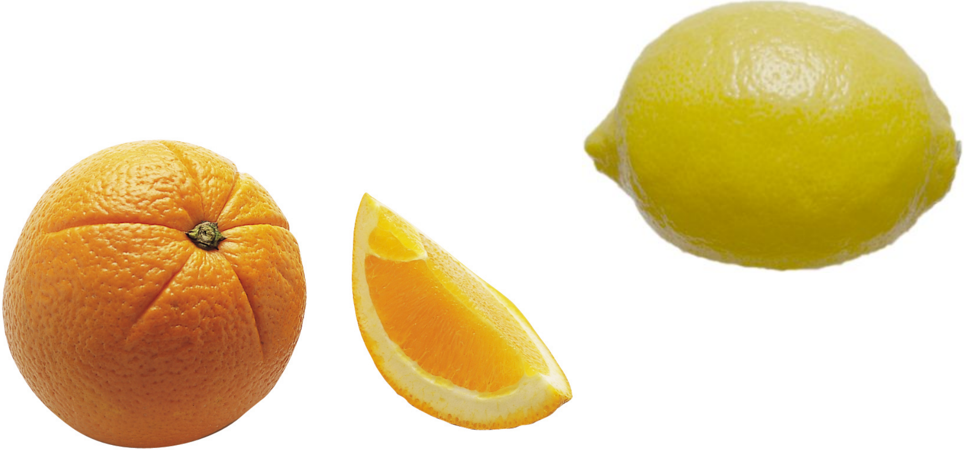
$$\mathbf{x} = (x_1, x_2, \ldots, x_{64})^T, \quad x_i \in [0, 127] \text{ or } x_i \in [0, 1]$$

http://alex.seewald.at/digits/

Image data as a point in a vector space

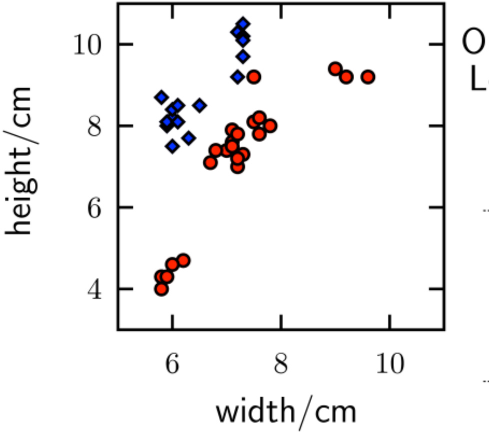


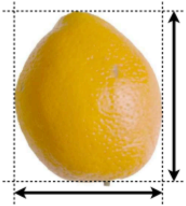
Classification of oranges and lemons



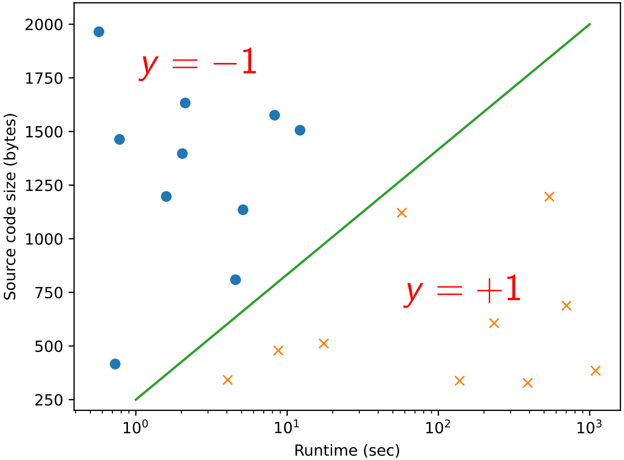
A two-dimensional space

Represent each sample as a point (w, h) in a 2D space

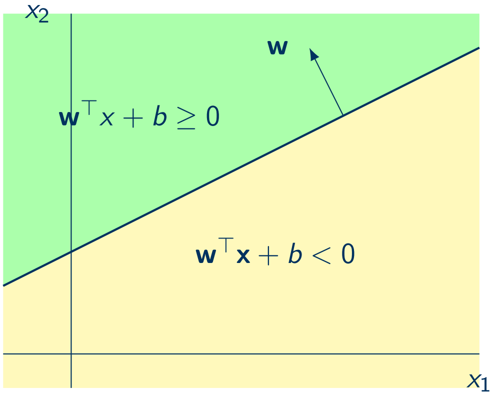




Classification



Geometry of linear classification



$$w_1 x_1 + w_2 x_2 + b = 0$$

$$\mathbf{w}^\top \mathbf{x} + b = 0 \quad \text{where } \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \ \mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$$

$\cdots$ **hyperplane**, **decision boundary**, splitting the space into **decision regions**

NB: $\boldsymbol{w}$ is a normal vector of the hyperplane. b is not the x_2 intercept.

Binary classification

$$h(\boldsymbol{x}) = \begin{cases} -1 & \text{if } \boldsymbol{w}^\top \boldsymbol{x} + b < 0 \\ +1 & \text{if } \boldsymbol{w}^\top x + b \geq 0 \end{cases} \tag{1}$$

- The hyperplane $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$ separates the two classes.
- The function h labels one class as -1 and the other class as $+1$.
- The task is called binary classification, because there are two classes.

Zero-one loss

$$\ell_{01}(\hat{y}, y) = \begin{cases} 1 & \text{if } \hat{y} \neq y \\ 0 & \text{otherwise} \end{cases} = \mathbb{1}_{\hat{y} \neq y} \tag{2}$$

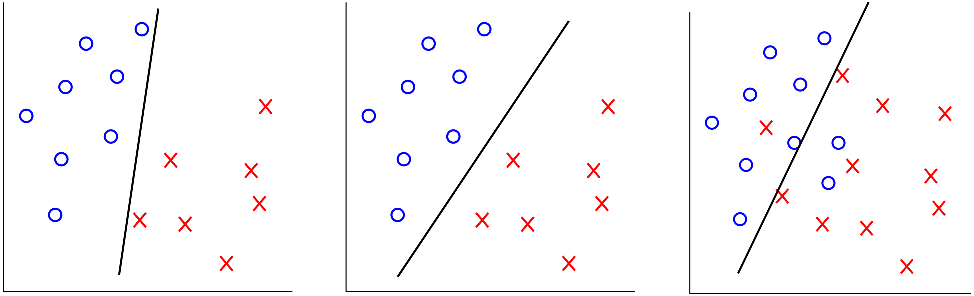
- Think $\hat{y}$ as the prediction and y as the label.
- We suffer a loss of 1 if we predict the label wrong.
- In the binary case, $\ell_{01}(\hat{y}, y) = \mathbb{1}_{\hat{y}y<0}$.

Classification

- $S = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$: data set
  - $\boldsymbol{x}_i = \begin{bmatrix} x_{i1} & \cdots & x_{id} \end{bmatrix}^\top$, $i = 1, \ldots, n$: input, feature vector, features
  - y_i: ground truth, label, gold reference, for $\boldsymbol{x}_i$.
- $f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$: linear separator, linear predictor
  - $\boldsymbol{w} = \begin{bmatrix} w_1 & \cdots & w_d \end{bmatrix}^\top$: weights, weight vector
  - $b \in \mathbb{R}$: bias
  - $\{\boldsymbol{w}, b\}$: parameters $\cdots$ $(\boldsymbol{\theta} = [b\ \boldsymbol{w}^\top]^\top)$
- $h(\boldsymbol{x}) = \text{sgn}(f(x))$, where $\text{sgn}(z) = \begin{cases} -1 & \text{if } z < 0 \\ +1 & \text{if } z \geq 0 \end{cases}$

NB: This is a non-standard definition of a sign function

Classification with a linear classifier



(a-1) (a-2) (b)

Linearly separable **Linearly non-separable**

Training of Classification

- Given $S = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$, find $\boldsymbol{\theta}$ such that the **zero-one loss**

$$L = \frac{1}{N}\sum_{i=1}^{N} \ell_{01}(h(\boldsymbol{x}_i), y_i) \tag{3}$$

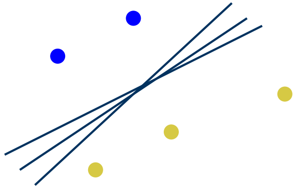
  is minimised. NB: L is called a **cost function**.

- The act of finding the model parameter $\boldsymbol{\theta}$ is called training.
  (We also say "fit the model on the training data" to mean the training)

- In the binary case,

$$L = \frac{1}{N}\sum_{i=1}^{N} \ell_{01}(\mathrm{sgn}(\boldsymbol{w}^\top \boldsymbol{x}_i + b), y_i) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{y_i(\mathrm{sgn}(\boldsymbol{w}^\top \boldsymbol{x}_i + b))<0} \tag{4}$$

Training based on the zero-one loss



- Slightly changing $\boldsymbol{w}$ and b does not change the loss.
- The loss value only changes when the hyperplane flips the sign of a data point, and it either increases by 1 or none at all.
- The loss function (with respect to $\boldsymbol{w}$ and b) is like step functions, flat everywhere with discontinuity when the value changes.
- Finding the optimal $\boldsymbol{w}$ and b is inherently combinatorial and hard.

Types of linear classifiers

- Template-based matching with Euclidean distance
- Fisher's linear discriminant
- Logistic regression
- Support Vector Machine (linear version)
- Perceptron (original version)
- Single-layer neural networks with no hidden nodes

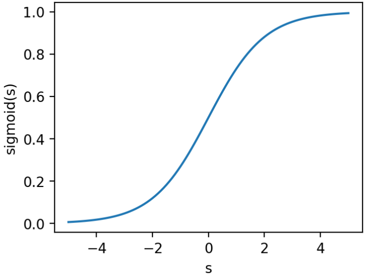
$\vdots$

A probabilistic approach

- The range of $f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$: $(-\infty, +\infty)$
- We want to squeeze the range into $[0, 1]$ with a function $g(s)$ so that it can be treated as a probability.
  $$g(f(\boldsymbol{x})) = g(\boldsymbol{w}^\top \boldsymbol{x} + b) \ \rightarrow \ p(y = +1 | \boldsymbol{x})$$
- A candidate for $g(s)$ is the **logistic (sigmoid) function**:
  $$g(s) = \frac{e^s}{1 + e^s} = \frac{1}{1 + e^{-s}} \tag{5}$$
- Logistic regression model:
  $$p(y = +1 | \boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{1 + \exp(-(\boldsymbol{w}^\top \boldsymbol{x} + b))} \tag{6}$$
  $$p(y = -1 | \boldsymbol{x}, \boldsymbol{\theta}) = 1 - p(y = +1 | \boldsymbol{x}) \tag{7}$$

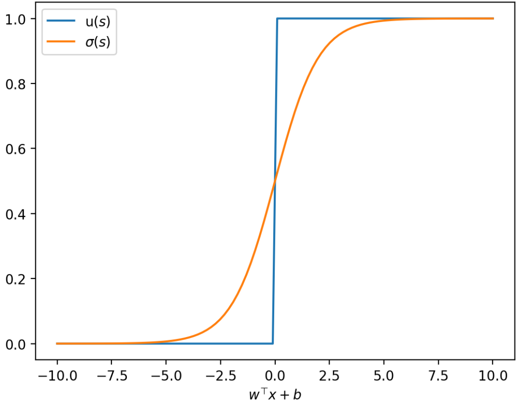
Sigmoid function

$$\sigma(s) = \frac{1}{1 + \exp(-s)}$$



- When $s \to \infty$, $\sigma(s) \to 1$.
- When $s \to -\infty$, $\sigma(s) \to 0$.

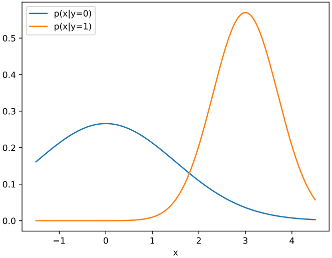
Sigmoid function vs step function



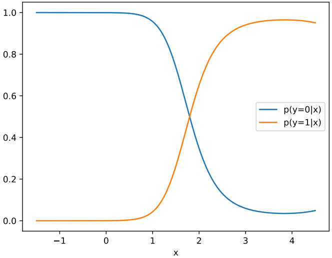
Step function: $u(s) = \begin{cases} 0 & \text{if } s < 0 \\ 1 & \text{if } s \geq 0 \end{cases}$

Interpretation of the logistic regression model

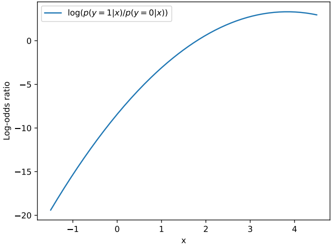
Data distributions $p(x|y)$



Posterior prob. $p(y|x)$



$\log \frac{p(y=1|x)}{p(y=0|x)}$



Model the log odds ratio with a line: $\log \frac{p(y=1|x)}{p(y=0|x)} = \boldsymbol{w}^\top \boldsymbol{x} + b$

Classification with the logistic regression model

For a test input $\boldsymbol{x}$,

1. calculate the posterior probability with the model.

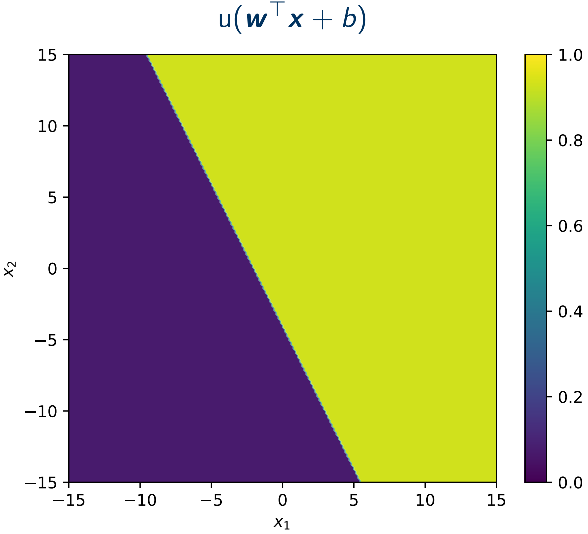
$$p(y=1|\boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{1 + \exp(-(\boldsymbol{w}^\top \boldsymbol{x} + b))}$$

2. make a prediction:

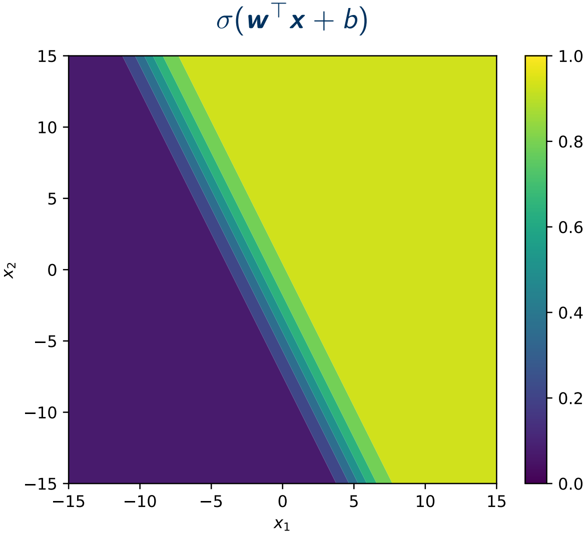
$$\hat{y} = \begin{cases} +1 & p(y=+1|\boldsymbol{x}, \boldsymbol{\theta}) > \text{threshold}, \\ -1 & p(y=+1|\boldsymbol{x}, \boldsymbol{\theta}) \leq \text{threshold} \end{cases} \tag{8}$$

NB: threshold $= 0.5$ normally – it gives a minimum misclassification rate.

Decision surface - step function version



Decision surface - sigmoid function version



A logistic regression model

$$p(y=+1|\boldsymbol{x},\boldsymbol{\theta}) = \frac{1}{1+\exp(-(\boldsymbol{w}^\top\boldsymbol{x}+b))} \tag{9}$$

$$p(y=-1|\boldsymbol{x},\boldsymbol{\theta}) = 1 - \frac{1}{1+\exp(-(\boldsymbol{w}^\top\boldsymbol{x}+b))} = \frac{\exp(-(\boldsymbol{w}^\top\boldsymbol{x}+b))}{1+\exp(-(\boldsymbol{w}^\top\boldsymbol{x}+b))} \tag{10}$$

$$= \frac{1}{\exp(\boldsymbol{w}^\top\boldsymbol{x}+b)+1} \tag{11}$$

Thus,

$$p(y|\boldsymbol{x},\boldsymbol{\theta}) = \frac{1}{1+\exp(-{\color{red}y}(\boldsymbol{w}^\top\boldsymbol{x}+b))} \tag{12}$$

How to train the logistic regression model?

- Use MSE? $\min_{\boldsymbol{w},b} \sum_{i=1}^{n} (p(y=+1|\boldsymbol{x}_i, \boldsymbol{\theta}) - y_i)^2$ NB: the label y_i needs to be changed to $\{0,1\}$.
- Apply the maximum likelihood estimation (MLE):

  Given a data set $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$,
  maximise the likelihood L of $\boldsymbol{w}$ and b.

$$\max_{\boldsymbol{w},b} L \tag{13}$$

$$L = \log \prod_{i=1}^{N} p(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}) = \sum_{i=1}^{N} \log \frac{1}{1 + \exp(-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b))} \tag{14}$$

$$= \sum_{i=1}^{N} - \log \left(1 + \exp(-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)) \right) \tag{15}$$

How to find the optimal solutions w and b?

- The zero-one loss $\sum_{i=1}^{N} \mathbb{1}_{y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) < 0}$ is flat, and is hard to optimise.

- The log likelihood of the logistic regression model $L = \sum_{i=1}^{N} - \log(1 + \exp(-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b))$ is differentiable.

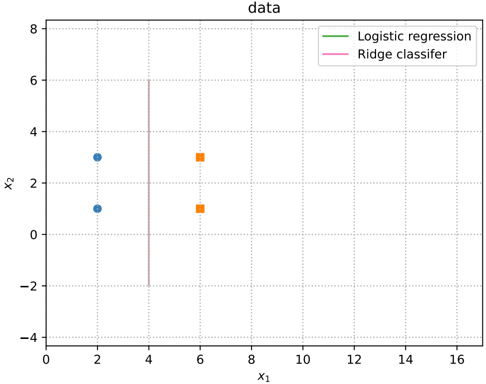
- However,

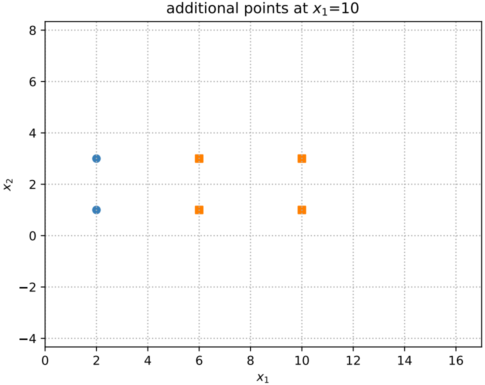
$$\frac{\partial L}{\partial w_i} = 0, \ \ i = 1, \dots, d \quad \text{and} \quad \frac{\partial L}{\partial b} = 0 \tag{16}$$

  do not have closed-form solutions.
  $\rightarrow$ employ gradient ascent.

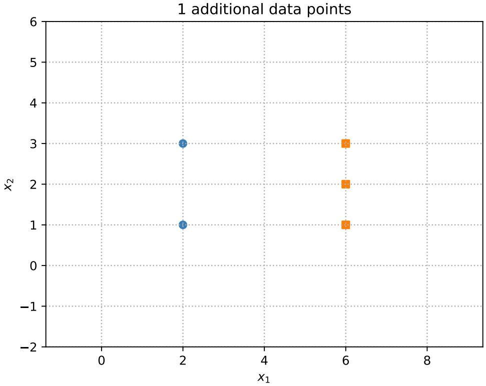
- We will come back to this in a lecture on optimisation.

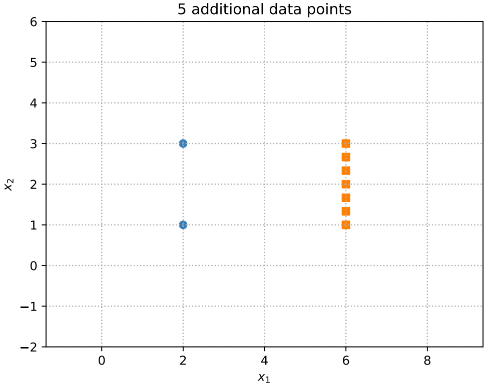
Effect of data distributions on decision regions





Effect of data distributions on decision regions *(cont.)*





Classification losses

Suppose we have a labelled data point $(\boldsymbol{x}, y)$.

- Zero-one loss

$$\mathbb{1}_{y(\boldsymbol{w}^\top \boldsymbol{x}+b)<0} \tag{17}$$

- Log loss (logistic loss)

$$-\log p(y|\boldsymbol{x}) = \log(1 + \exp(-y(\boldsymbol{w}^\top \boldsymbol{x} + b)) \tag{18}$$

Notation caveat

- The log loss notation $-\log p(y|\boldsymbol{x})$ can be misleading.
- Is y the ground truth or is it a free variable?
- What it really means is $-\log p(y = y^*|\boldsymbol{x})$ given a pair $(x, \boldsymbol{y}^*)$.
- Or $-\log p(y = y_i|\boldsymbol{x}_i)$ given a pair $(\boldsymbol{x}_i, y_i)$ in a data set.

How to resolve a linearly non-separable case?

Feature transformation

$$h(\boldsymbol{x}) = \begin{cases} -1 & \text{if } \boldsymbol{w}^\top \boldsymbol{x} + b < 0 \\ +1 & \text{if } \boldsymbol{w}^\top \boldsymbol{x} + b \geq 0 \end{cases} = \operatorname{sgn}(\boldsymbol{w}^\top \boldsymbol{x} + b) \tag{19}$$

$$\downarrow$$

$$h(\boldsymbol{x}) = \begin{cases} -1 & \text{if } \boldsymbol{w}^\top \phi(\boldsymbol{x}) < 0 \\ +1 & \text{if } \boldsymbol{w}^\top \phi(\boldsymbol{x}) \geq 0 \end{cases} = \operatorname{sgn}(\boldsymbol{w}^\top \phi(\boldsymbol{x})) \tag{20}$$

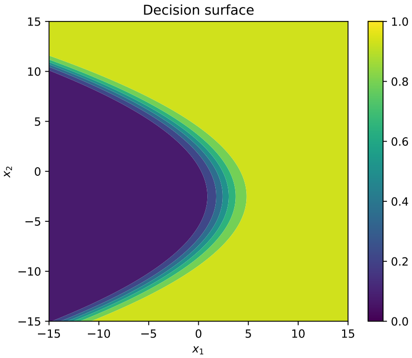
Feature transformation *(cont.)*

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{1 + \exp(-y(\boldsymbol{w}^\top \boldsymbol{x} + b))} \tag{21}$$

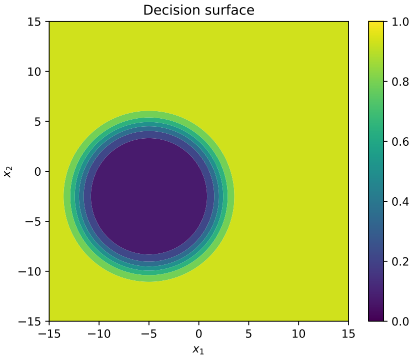
$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{1 + \exp(-y(\boldsymbol{w}^\top \phi(\boldsymbol{x})))} \tag{22}$$

Feature transformation - examples

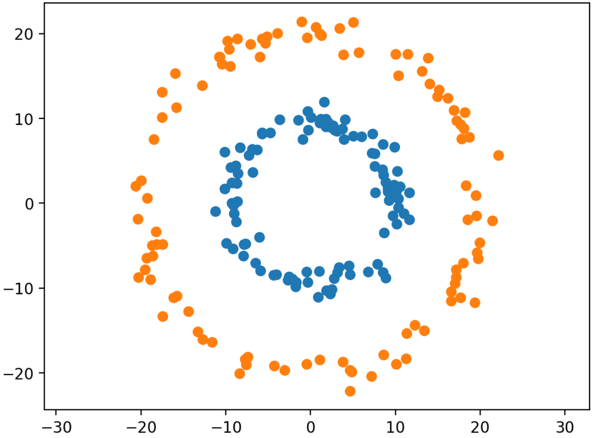
$$(x_1, x_2) \rightarrow (x_1, x_2, x_2^2)$$



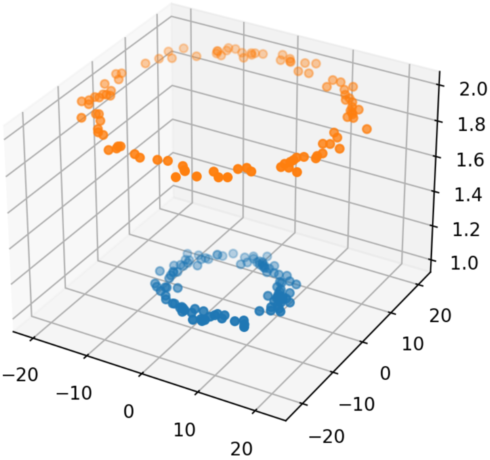
$$(x_1, x_2) \rightarrow (x_1, x_2, x_1^2, x_2^2)$$



Two-circle example



Two-circle example



What is it meant by linear classifiers?

- A linear classifier is linear in the parameters w, **not** in the features.
- A linear classifier can have arbitrary nonlinear features.

Should we consider very complex transformation?

- Not necessarily so.
- Complex models may **overfit** the training data and may not **generalise** very well.
- We will come back to this in some lectures later.

How to extend the model to multiclass classification?

- one-vs.-all (one-against-all)
- one-vs.-one

Multiclass classification with logistic regression

Replace the sigmoid with the **softmax function**

- w/o transformation

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \frac{\exp(\boldsymbol{w}_y^\top \boldsymbol{x})}{\sum_{y' \in \mathcal{Y}} \exp(\boldsymbol{w}_{y'}^\top \boldsymbol{x})} \tag{23}$$

- w transformation

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \frac{\exp(\boldsymbol{w}_y^\top \phi(\boldsymbol{x}))}{\sum_{y' \in \mathcal{Y}} \exp(\boldsymbol{w}_{y'}^\top \phi(\boldsymbol{x}))} \tag{24}$$

NB: we can just use and compare "$\boldsymbol{w}_y^\top \phi(\boldsymbol{x})$" for classification – the denominator is a constant for $y \in \mathcal{Y}$ and exp() is a monotonically increasing function.

Softmax for binary classification

$$p(y = +1 | \boldsymbol{x}, \boldsymbol{\theta}) = \frac{\exp(\boldsymbol{w}_{+1} 1^\top \boldsymbol{x})}{\exp(\boldsymbol{w}_{+1}^\top \boldsymbol{x}) + \exp(\boldsymbol{w}_{-1}^\top \boldsymbol{x})} \tag{25}$$

$$= \frac{1}{1 + \exp(-(\boldsymbol{w}_{+1} - \boldsymbol{w}_{+1})^\top \boldsymbol{x})} = \frac{1}{1 + \exp(-\boldsymbol{w}^\top \boldsymbol{x})} \tag{26}$$

$$p(y = -1 | \boldsymbol{x}, \boldsymbol{\theta}) = \frac{\exp(\boldsymbol{w}_{-1}^\top \boldsymbol{x})}{\exp(\boldsymbol{w}_{+1}^\top \boldsymbol{x}) + \exp(\boldsymbol{w}_{-1}^\top \boldsymbol{x})} \tag{27}$$

$$= \frac{\exp(-(\boldsymbol{w}_{+1} - \boldsymbol{w}_{-1})^\top \boldsymbol{x})}{1 + \exp(-(\boldsymbol{w}_{+1} - \boldsymbol{w}_{-1})^\top \boldsymbol{x})} = \frac{\exp(-\boldsymbol{w}^\top \boldsymbol{x})}{1 + \exp(-\boldsymbol{w})^\top \boldsymbol{x})} \tag{28}$$

where $\boldsymbol{w} = \boldsymbol{w}_{+1} - \boldsymbol{w}_{-1}$.

$\rightarrow$ the same as the sigmoid.

Summary

- Log loss in the binary case

$$\sum_{i=1}^{N} \log\left(1 + \exp(-y_i \boldsymbol{w}^\top \phi(\boldsymbol{x}_i))\right) \tag{29}$$

- Log loss in the multiclass case

$$\sum_{i=1}^{N} -\boldsymbol{w}_{y_i}^\top \phi(\boldsymbol{x}_i) + \log\left(\sum_{y' \in \mathcal{Y}} \exp(\boldsymbol{w}_{y'}^\top \phi(\boldsymbol{x}_i))\right) \tag{30}$$

Summary *(cont.)*

binary classification	multiclass classification
$h(\boldsymbol{x}) = \begin{cases} -1 & \text{if } \boldsymbol{w}^\top \phi(\boldsymbol{x}) < 0 \\ +1 & \text{if } \boldsymbol{w}^\top \phi(\boldsymbol{x}) \geq 0 \end{cases}$	$h(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} \boldsymbol{w}_y^\top \phi(\boldsymbol{x})$
$p(y \mid \boldsymbol{x}, \boldsymbol{\theta}) = \dfrac{1}{1 + \exp(-y\boldsymbol{w}^\top \phi(\boldsymbol{x}))}$	$p(y \mid \boldsymbol{x}, \boldsymbol{\theta}) = \dfrac{\exp(\boldsymbol{w}_y^\top \phi(\boldsymbol{x}))}{\sum_{y' \in \mathcal{Y}} \exp(\boldsymbol{w}_{y'}^\top \phi(\boldsymbol{x}))}$

Appendix – softmax

$$\operatorname{softmax}\left(\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}\right) = \begin{bmatrix} \dfrac{\exp(a_1)}{\sum_{i=1}^{n} \exp(a_i)} \\ \dfrac{\exp(a_2)}{\sum_{i=1}^{n} \exp(a_i)} \\ \vdots \\ \dfrac{\exp(a_n)}{\sum_{i=1}^{n} \exp(a_i)} \end{bmatrix} \tag{31}$$

Appendix – softmax *(cont.)*

- $\mathrm{softmax}(\begin{bmatrix}1 & 2 & 3\end{bmatrix}^\top) = \begin{bmatrix}0.09 & 0.24 & 0.67\end{bmatrix}^\top$
- $\mathrm{softmax}(\begin{bmatrix}100 & 200 & 300\end{bmatrix}^\top) = \begin{bmatrix}10^{-87} & 10^{-44} & 1.0\end{bmatrix}^\top$
- Softmax always returns a probability distribution.
- When the dynamic range of the input is large, the result of softmax becomes "sharp."

Appendix – softmax *(cont.)*

- Claim: $\frac{\exp(a_{\max}/\tau)}{\sum_{i=1}^{n}\exp(a_i/\tau)} \to 1$ when $\tau \to 0$.
- That means $\frac{\exp(a_j/\tau)}{\sum_{i=1}^{n}\exp(a_i/\tau)} \to 0$ when $\tau \to 0$ for any a_j that is not the max.
- We have

$$\frac{\exp(a_m/\tau)}{\sum_{i=1}^{n}\exp(a_i/\tau)} = \frac{\exp(a_m/\tau)}{\exp(a_m/\tau) + \sum_{i\neq m}\exp(a_i/\tau)} \tag{32}$$

$$= \frac{1}{1 + \sum_{i\neq m}\exp((a_i - a_m)/\tau)} \to 1 \tag{33}$$

when $\tau \to 0$ because a_m is the largest and $a_i - a_m < 0$.

Quizzes

1. Consider two column vectors such that $\mathbf{a} = (1, 2, 3)^T$ and $\mathbf{b} = (-3, 3, -1)^T$.
   - Find $\mathbf{a} + \mathbf{b}$.
   - Find $\mathbf{a} - \mathbf{b}$.
   - Find $\|\mathbf{a}\|, \|\mathbf{b}\|$, and $\|\mathbf{a} - \mathbf{b}\|$.
   - Find $\mathbf{a}^T\mathbf{b}$.
   - Find $\mathbf{a}\mathbf{b}^T$.
   - What is the geometric relationship between $\mathbf{a}$ and $\mathbf{b}$?
2. Considering a classification problem of two classes, whose discriminant function takes the form, $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$.
   - Show that the decision boundary is a straight line when $D = 2$.
   - Show that the weight vector $\mathbf{w}$ is a normal vector to the decision boundary.
3. Derive a formula for the Euclidean distance between the origin $(0, 0)$ and a line $y = ax + b$, where a and b are arbitrary constants.

Quizzes *(cont.)*

4. Considering a linear classifier of binary classification in a two-dimensional vector space, such that the points $(-2, -3)$ and $(4, 1)$ are on the decision boundary, and the point $(2, -3)$ lies in the -1 class region.
   - Find the parameters $(\boldsymbol{w}, b)$ of the classifier.
   - Find the unit normal vector of $\boldsymbol{w}$.
5. Consider the following logistic regression model:
   $$p(y = +1|x) = \frac{1}{1 + \exp(-(wx + b))}$$
   Plot $p(y = +1|x)$ for each of the following cases, where you use a fixed plotting range or show all the plots on a single graph for comparison, and report your findings.
   - $w = 1, b = 0$
   - $w = 1, b = 1$
   - $w = -1, b = 1$
   - $w = 0.5, b = 1$
   - $w = 2, b = 1$

Quizzes *(cont.)*

6. Consider the logistic sigmoid function.

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$

   - Based on the graph of $\sigma(x)$, make an educated guess about the shape of the derivative $\sigma'(x)$ without performing any calculations and illustrate it by hand.
   - Find the derivative of $\sigma(x)$.
   - Plot the derivative on a graph.