Statistical Machine Learning

Christian Walder

College of Engineering and Computer Science
The Australian National University

Canberra
Semester One, 2020.

(Many figures from C. M. Bishop, "Pattern Recognition and Machine Learning")

Outlines

Part V

Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

Strategy in this course

Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

- Estimate best predictor = training = learning

Giv

1.
2.
3.
4. Calculate the optimal parameter ($\mathbf{w}$)
5. Model uncertainty using the Bayesian appro
6. Implement and compute (the algorithm in py
7. Interpret and diagnose results

Classification

Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

- Goal : Given input data $\mathbf{x}$, assign it to one of K discrete classes $\mathcal{C}_k$ where $k = 1, \ldots, K$.
- Divide the input space into different regions.
- Eq

Length of petal [in cm] vs sepal [cm] for three types of flowers (Iris Setosa, Iris Versicolor, Iris Virginica).

How to represent binary class labels?

Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

- Class labels are no longer real values as in regression, but a dis
- Tw
(t
- Ca
with only two values possible for the probability
- Note: Other conventions to map classes into int
possible, check the setup.

1

How to represent multi-class labels?

- If there are more than two classes ($K > 2$), we call it a multi-class setup.
- Oft
  len
  co
- Exa
  class $\mathcal{C}_2$ will be encoded as the target vector

$$\mathbf{t} = (0, 1, 0, 0, 0)^T$$

- Note: Other conventions to map multi-classe
  possible, check the setup.

Linear Model

- Idea: Use again a Linear Model as in regression: $y(\mathbf{x}, \mathbf{w})$ is a linear function of the parameters $\mathbf{w}$

$$y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x})$$

- But
  Exa

Generalised Linear Model

- Apply a mapping $f : \mathbb{R} \to \mathbb{Z}$ to the linear model to get the discrete class labels.
- Generalised Linear Model

$$y(\mathbf{x}_n, \mathbf{w}) = f(\mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x}_n))$$

- Acti
- Lin

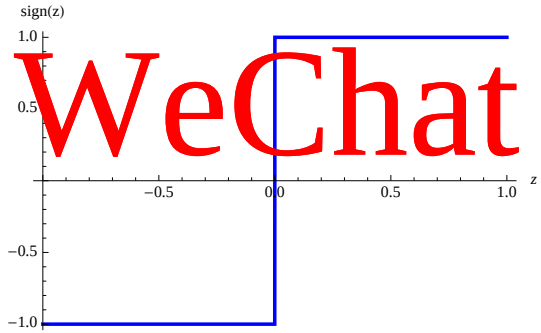


Figure: Example of an activation function $f(z) = \text{sign}(z)$.

©2020
Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

Three Models for Decision Problems

- Find a discriminant function $f(\mathbf{x})$ which maps each input directly onto a class label.
- Discriminative Models
  1.
  2.
- Generative Models
  1. Solve the inference problem of determining th class-conditional probabilities $p(\mathbf{x}|\mathcal{C}_k)$
  2. *Also infer the prior class probabilities*
  3. *Use Bayes' theorem to find the posterior* $p(\mathcal{C}_k|\mathbf{x})$
  4. *Alternatively model the joint distribution* $p(\mathbf{x}, \mathcal{C}_k)$ directly.
  5. Use decision theory to assign each new $\mathbf{x}$ to one of the classes.

©2020 Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

Definition

A discriminant is a function that maps from an input vector $\mathbf{x}$ to one of K classes, denoted by $\mathcal{C}_k$.

- Co
- Co

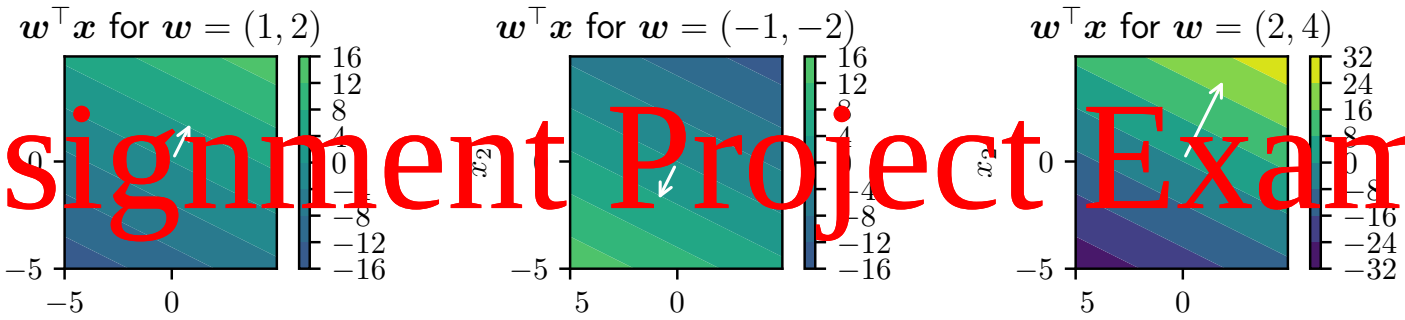
$$y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + w_0$$

such that $\mathbf{x}$ being assigned to class $\mathcal{C}_1$ class $\mathcal{C}_2$ otherwise.

- weight vector $\mathbf{w}$
- bias w_0 (sometimes $-w_0$ called threshold)

©2020 Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

Linear Functions



- Gra
- The $+ \quad_0 =$
- Projecting $\mathbf{x}$ on that hyper-plane means find $\arg\min_{\mathbf{x}_\perp} \|\mathbf{x} - \mathbf{x}_\perp\|$ subject to the constrain $\mathbf{w}^\top \mathbf{x}_\perp + w_0 = 0$. Geometrically: move in the d $\quad -$
- Rate of change of function value in that direction is $\frac{\mathrm{d}}{\mathrm{d}a}\left(a\frac{\mathbf{w}}{\|\mathbf{w}\|}\right)^\top \mathbf{w} = a\|\mathbf{w}\|$.
- The length $\left\|a\frac{\mathbf{w}}{\|\mathbf{w}\|}\right\| = \frac{a}{\|\mathbf{w}\|}\|\mathbf{w}\| = a$.
- For a fixed change in $\mathbf{w}^\top\left(a\frac{\mathbf{w}}{\|\mathbf{w}\|}\right)$, $a \propto \frac{1}{\|\mathbf{w}\|}$.

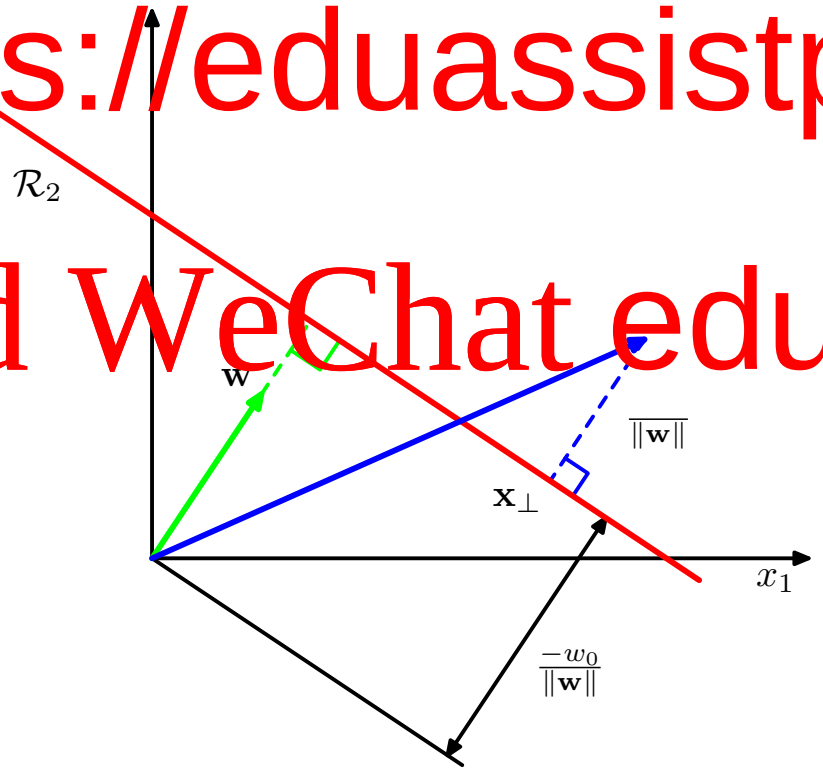
Two Classes

- Decision boundary $y(\mathbf{x}) = 0$ is a $(D-1)$-dimensional hyp
surf
- $\mathbf{w}$ is
- Pro
decision surface. Then,

$$0 = y(\mathbf{x}_A) - y(\mathbf{x}_B) = \mathbf{w}^\top(\mathbf{x}_A - \mathbf{x}_B)$$

- $y(\mathbf{x})$ gives a **signed** measure of the perpendicular distance r **from** the decision surface **to** $\mathbf{x}$, that is $r = y(\mathbf{x})/\|w\|$.

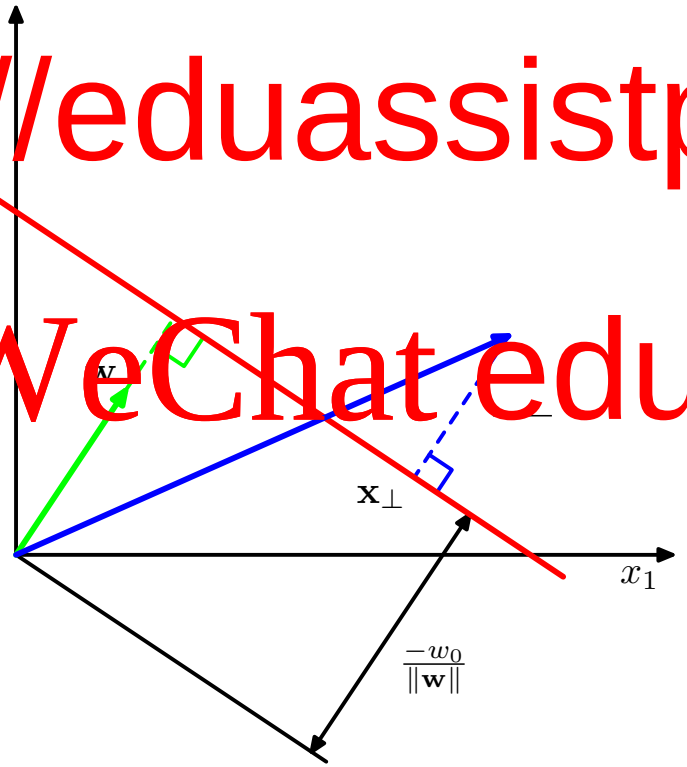
$$y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + r\frac{\mathbf{w}}{\|\mathbf{w}\|} + w_0 = r\frac{\mathbf{w}^\top \mathbf{w}}{\|\mathbf{w}\|} + \mathbf{w}^\top \mathbf{x} + w_0 = r\,\|w\| + 0$$



Two Classes

- The normal distance from the origin to the decision surface is therefore

$$-\frac{y(\mathbf{0})}{\|w\|} = -\frac{w_0}{\|w\|}$$



Two Classes

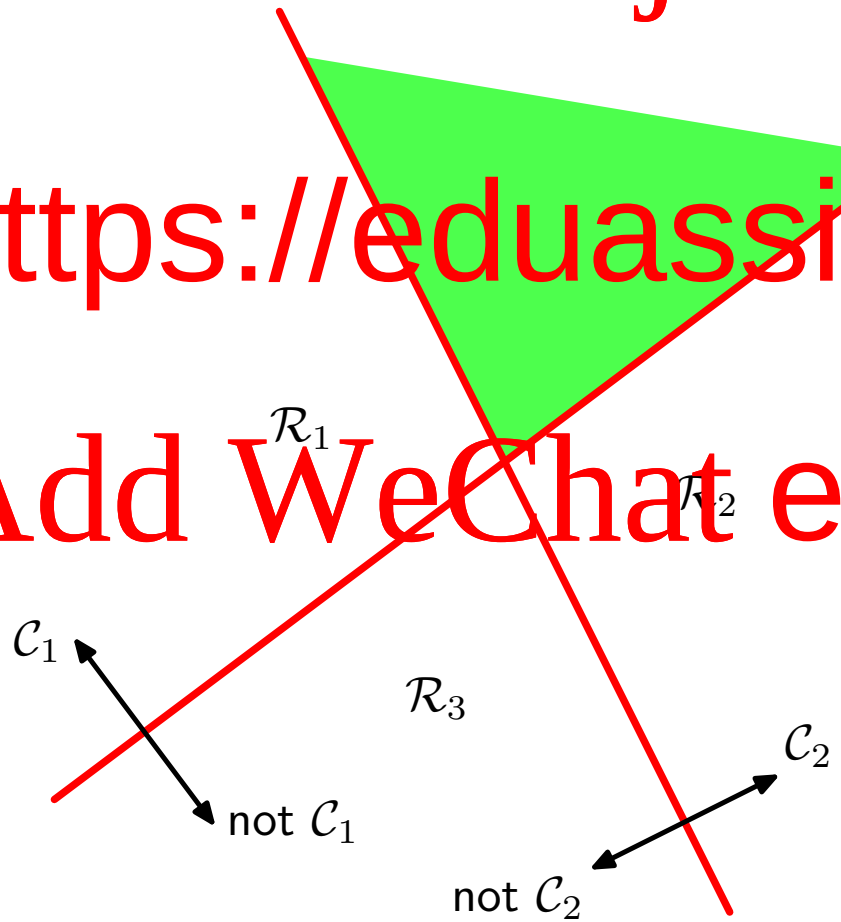
- More compact notation : Add an extra dimension to the inp
- Als

(if it helps, you may think of $\widetilde{\mathbf{w}}^\top$ as a func

- Decision surface is now a D-dimension $D+1$-dimensional expanded input space.

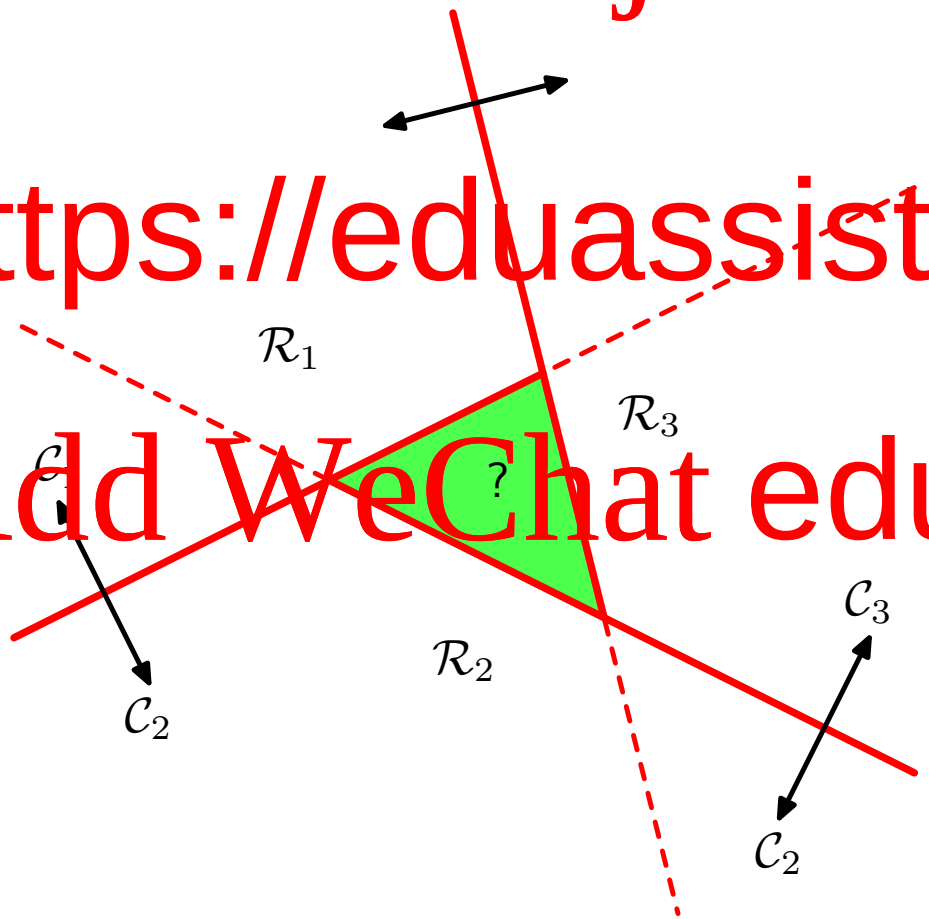
Multi-Class

- Number of classes $K > 2$
- Can we combine a number of two-class discriminant functions using $K - 1$ one-versus-the-rest classifiers?



Multi-Class

- Number of classes $K > 2$
- Can we combine a number of two-class discriminant functions using $K(K-1)/2$ one versus one classifiers?

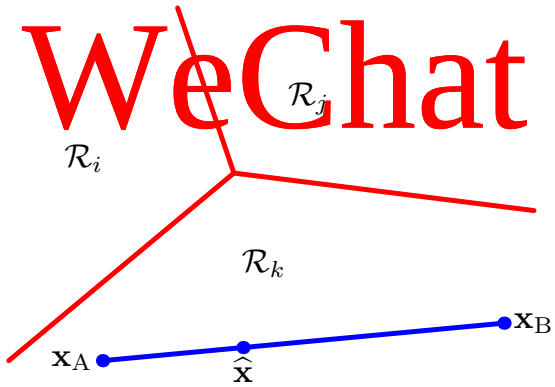


- Number of classes $K > 2$
- Solution: Use K linear functions

$$y_k(\mathbf{x}) = \mathbf{w}_k^\top \mathbf{x} + w_{k0}$$

- Ass
- De

$y_k(\mathbf{x}$ j



Least Squares for Classification

- Regression with a linear function of the model parameters and minimisation of sum-of-squares error function resulted in a cl
- Is thi
- Giv
- Use 1-of-K binary coding scheme.
- Each class is described by its own linear model

$$y_k(\mathbf{x}) = \mathbf{w}_k^\top \mathbf{x} + w_{k0} \qquad k =$$

- With the conventions

$$\widetilde{\mathbf{w}}_k = \begin{bmatrix} w_{k0} \\ \mathbf{w}_k \end{bmatrix} \in \mathbb{R}^{D+1}$$

- we get for the (vector valued) discriminant func

$$\mathbf{y}(\mathbf{x}) = \widetilde{\mathbf{W}}^\top \widetilde{\mathbf{x}}$$

(if it helps, you may think of $\widetilde{\mathbf{W}}^\top$ as a **vector-valued** function).

- For a new input $\mathbf{x}$, the class is then defined by the index of the largest value in the row vector $\mathbf{y}(\mathbf{x})$

Determine $\widetilde{\mathbf{W}}$

- Given a training set $\{\mathbf{x}_n, \mathbf{t}_n\}$ where $n = 1, \ldots, N$, and $\mathbf{t}_n$ is the class in the 1-of-K coding scheme.
- Define a matrix $\mathbf{T}$ where row n corresponds to $\mathbf{t}_n^\top$.
- The
(ch

$$E_D(\widetilde{\ }) = \frac{}{2}\,\mathrm{tr}\ \ (\widetilde{\ }\ \widetilde{\ } - \ \ )\ (\qquad\qquad)$$

- The minimum of $E_D(\widetilde{\mathbf{W}})$ will be reached for

$$\widetilde{\mathbf{W}} = (\widetilde{\mathbf{X}}^\top \widetilde{\mathbf{X}})^{-1} \widetilde{\mathbf{X}}^\top \mathbf{T} = \widetilde{\mathbf{X}}^\dagger \mathbf{T}$$

where $\widetilde{\mathbf{X}}^\dagger$ is the pseudo-inverse of $\widetilde{\mathbf{X}}$.

Discriminant Function for Multi-Class

- The discriminant function $\mathbf{y}(\mathbf{x})$ is therefore

$$\mathbf{y}(\mathbf{x}) = \mathbf{W}^\top \mathbf{x} = \mathbf{T}^\top (\mathbf{X}^\top)^\top \mathbf{x}$$

wh
inp

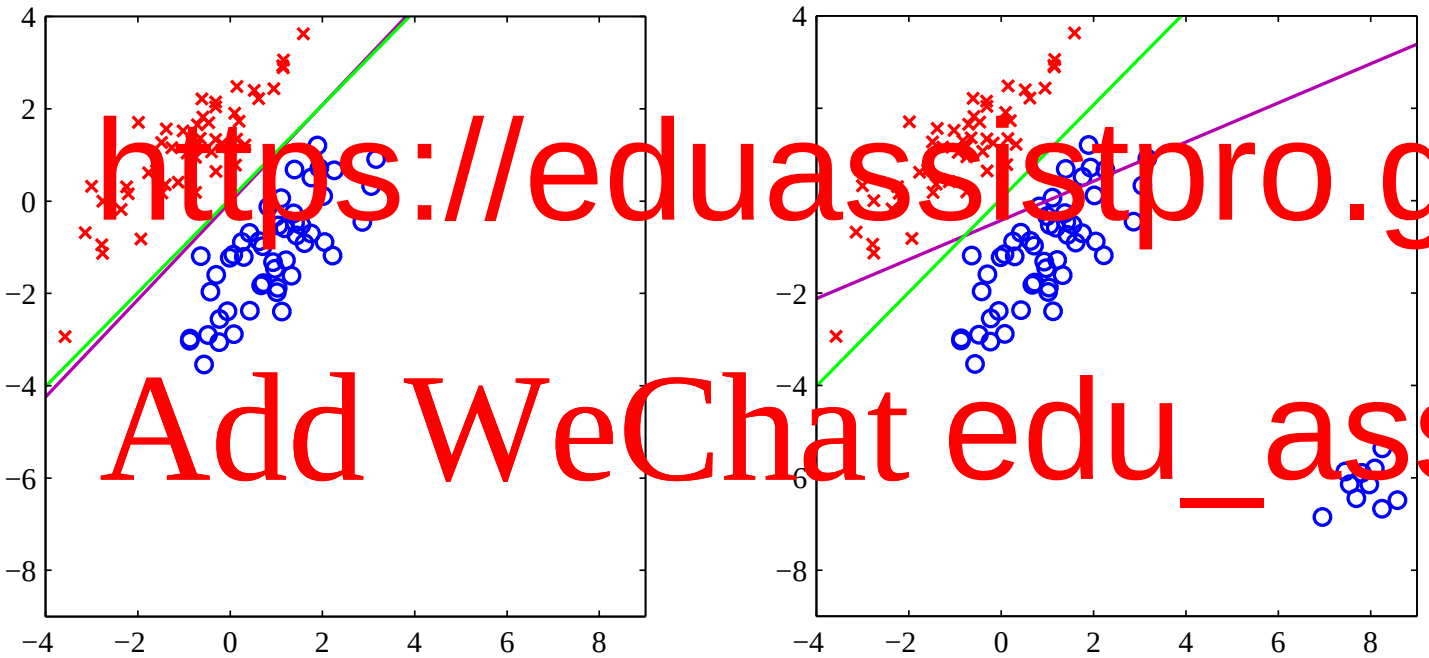
- Inte
con
also obey the same constraint

$$\mathbf{a}^\top \mathbf{y}(\mathbf{x}) + b = \mathbf{0}.$$

- For the 1-of-K coding scheme, the sum of all components in $\mathbf{t}_n$ is one, and therefore all components of $\mathbf{y}(\mathbf{x})$ will sum to one. BUT: the components are not probabilities, as they are not constraint to the interval $(0, 1)$.

Deficiencies of the Least Squares Approach

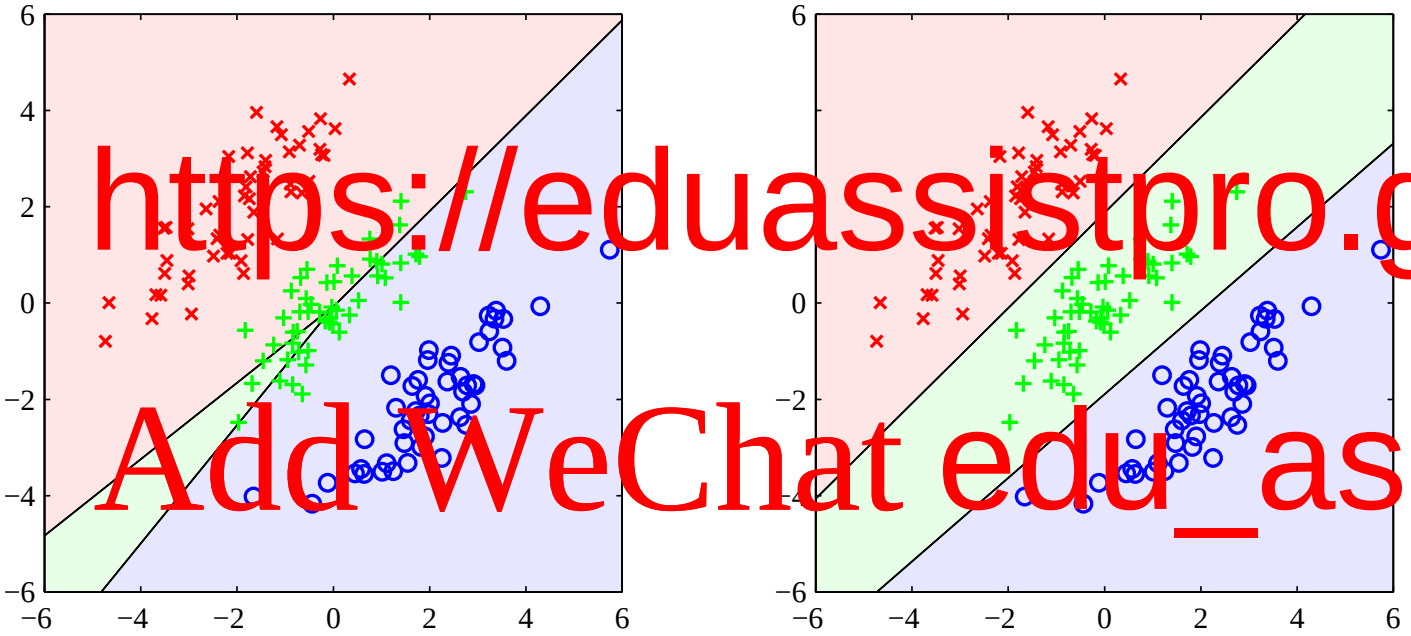
Magenta curve :
Decision Boundary for the least squares approach
Green curve :
Decision boundary for the logistic regression (described later)



(Imagine heat-maps of the quadratic penalty function, similarly to those of the linear functions earlier in the slides.)

Deficiencies of the Least Squares Approach

Left plot : Decision Boundary for least squares
Right plot : Boundaries for logistic regression (described later)



Fisher's Linear Discriminant

Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

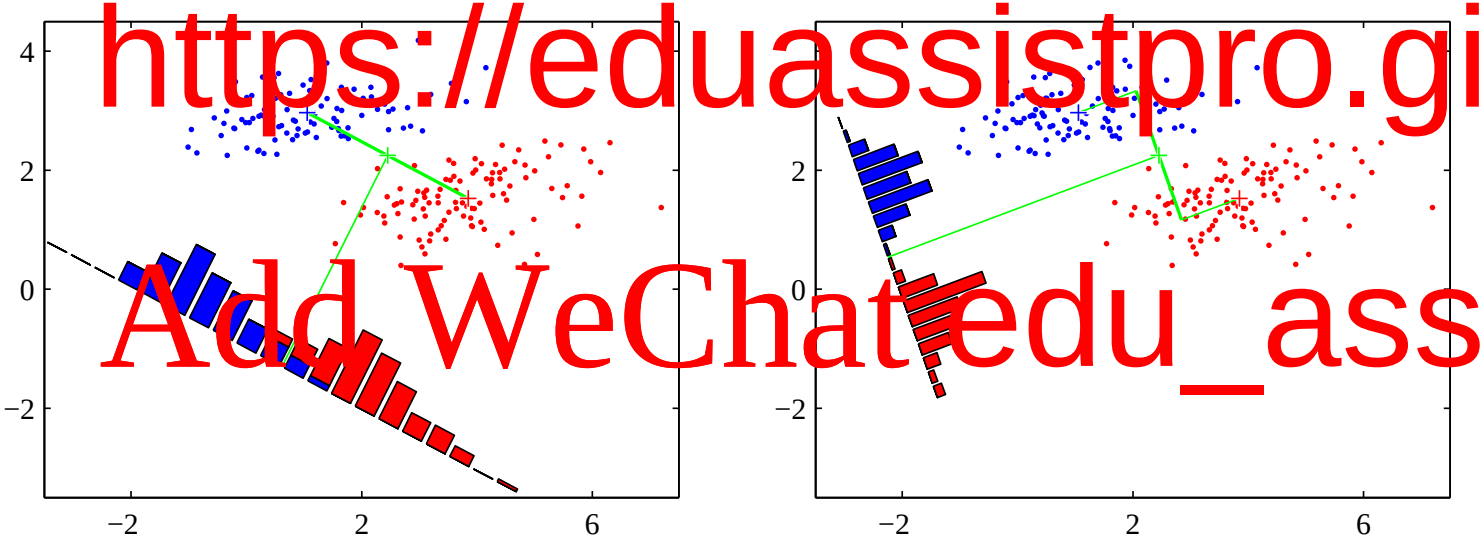
- View linear classification as dimensionality reduction.

$$y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$$

- If y
- But
  spa
- Projection always means loss of information.
- For classification we want to preserve the class in one dimension.
- Can we find a projection which maximally preserves the class separation ?

Fisher's Linear Discriminant

Samples from two classes in a two-dimensional input space and their histogram when projected to two different one-dimensional spaces.



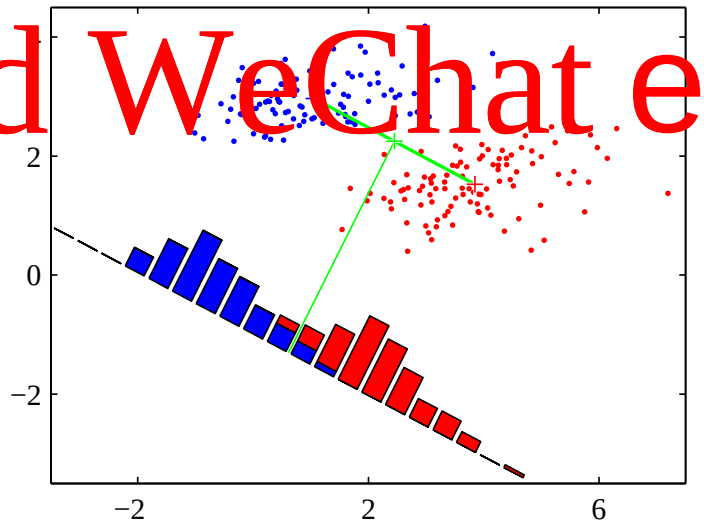
Fisher's Linear Discriminant - First Try

- Given N_1 input data of class $\mathcal{C}_1$, and N_2 input data of class $\mathcal{C}_2$, calculate the centres of the two classes

$$\mathbf{m}_1 = \frac{1}{N_1} \sum_{n \in \mathcal{C}_1} \mathbf{x}_n, \qquad \mathbf{m}_2 = \frac{1}{N_2} \sum_{n \in \mathcal{C}_2} \mathbf{x}_n$$

- Ch
proj

- Problem with non-uniform covariance



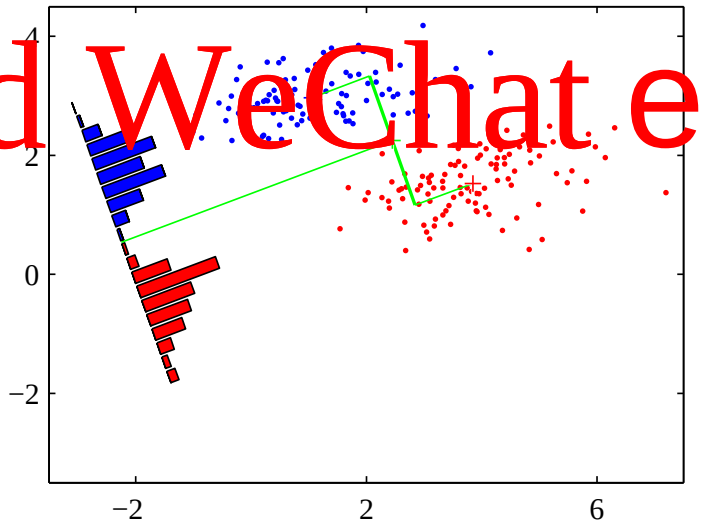
Fisher's Linear Discriminant

- Measure also the within-class variance for each class

$$s_k^2 = \sum_{n \in \mathcal{C}_k} (y_n - m_k)^2$$

where $y_n = \mathbf{w}^\top \mathbf{x}_n$.

- Ma



Primer: Bilinear form with a Covariance Matrix

Let

$$\begin{aligned}\boldsymbol{\mu} &= \mathbb{E}[\mathbf{x}] \\ \Sigma &= \mathrm{cov}[\mathbf{x}] \\ &= \mathbb{E}[(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^\top].\end{aligned}$$

Then

$$\begin{aligned}&= \mathbb{E}\left[(\mathbf{w}^\top\mathbf{x} - \mathbf{w}^\top\mathbb{E}[\mathbf{x}])^2\right] \\ &= \mathbb{E}\left[(\mathbf{w}^\top\mathbf{x} - \mathbf{w}^\top\boldsymbol{\mu})^2\right] \\ &= \mathbb{E}\left[(\mathbf{w}^\top\mathbf{x} - \mathbf{w}^\top\boldsymbol{\mu})(\mathbf{w}^\top\mathbf{x} - \mathbf{w}^\top\boldsymbol{\mu})^\top\right] \\ &= \mathbb{E}\left[(\mathbf{w}^\top\mathbf{x} - \mathbf{w}^\top\boldsymbol{\mu})(\mathbf{x}^\top\mathbf{w} - \boldsymbol{\mu}^\top\mathbf{w})\right] \\ &= \mathbb{E}\left[\mathbf{w}^\top(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^\top\mathbf{w}\right] \\ &= \mathbf{w}^\top\mathbb{E}\left[(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^\top\right]\mathbf{w} \\ &= \mathbf{w}^\top\Sigma\mathbf{w}.\end{aligned}$$

Fisher's Linear Discriminant

- The Fisher criterion can be rewritten as

$$J(\mathbf{w}) = \frac{\mathbf{w}^\top \mathbf{S}_B \mathbf{w}}{\mathbf{w}^\top \mathbf{S}_W \mathbf{w}}$$

- $\mathbf{S}_B$ is the between-class covariance

so b

the variance of the projection of the means

- $\mathbf{S}_W$ is the within-class covariance

$$\mathbf{S}_W = \sum_{n \in \mathcal{C}_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^T + \sum_{n \in \mathcal{C}_2} ($$

so so by the previous slide and $\mathbf{w}^\top (A + B)\mathbf{w} = \mathbf{w}^\top A\mathbf{w} + \mathbf{w}^\top B\mathbf{w}$, the denominator of $J(\mathbf{w})$ is:
(the variance of the projection of the points in class $\mathcal{C}_1$) +
(the variance of the projection of the points in class $\mathcal{C}_2$)

Fisher's Linear Discriminant

- The Fisher criterion

$$\frac{\mathbf{w}^\top \mathbf{S}_B \mathbf{w}}{}$$

has

$$\mathbf{w} \propto \mathbf{S}_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

- Fisher's linear discriminant is NOT a discriminant but it can be used to construct one by choosing a threshold y_0 in the projection space.

Fisher's Discriminant For Multi-Class

- Assume that the dimensionality of the input space D is greater than the number of classes K.
- Use $D' > 1$ linear 'features' $y_k = \mathbf{w}^\top \mathbf{x}$ and write everything in vector form (with no bias term)

$$\mathbf{y} = \mathbf{W}^\top \mathbf{x}.$$

- The cov

$$\mathbf{S}_W = \sum_{k=1}^{K} \mathbf{S}_k$$

where

$$\mathbf{S}_k = \sum_{n \in \mathcal{C}_k} (\mathbf{x}_n - \mathbf{m}_k)(\mathbf{x}_n - \mathbf{m}_k)^\top$$

$$\mathbf{m}_k = \frac{1}{N_k} \sum_{n \in \mathcal{C}_k} \mathbf{x}_n$$

Fisher's Discriminant For Multi-Class

- Between-class covariance

$$\mathbf{S}_B = \sum_{k=1}^{K} N_k (\mathbf{m_k} - \mathbf{m})(\mathbf{m_k} - \mathbf{m})^T$$

wh

$$n=1$$

- One possible way to define a function of when the between-class covariance is large a within-class covariance is small is given by

$$J(\mathbf{W}) = \mathrm{tr}\left\{(\mathbf{W}^\top \mathbf{S}_W \mathbf{W})^{-1}(\mathbf{W}^\top \mathbf{S}_B \mathbf{W})\right\}$$

- The maximum of $J(\mathbf{W})$ is determined by the D' eigenvectors of $\mathbf{S}_W^{-1}\mathbf{S}_B$ with the largest eigenvalues.

The Perceptron Algorithm

- Frank Rosenblatt (1928 - 1969)
- "Principles of neurodynamics: Perceptrons and the theory of brain mechanisms" (Spartan Books, 1962)

©2020 Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

The Perceptron Algorithm

- Perceptron ("MARK 1") was the first computer which could learn new skills by trial and error

©2020 Ong & Walder & Webers
Data61 | CSIRO
The Australian National University

The Perceptron Algorithm

- Two class model
- Create feature vector $\phi(\mathbf{x})$ by a fixed nonlinear transformation of the input $\mathbf{x}$.
- Generalised linear model

wit

- non

$$f(a) = \begin{cases} +1, & a \geqslant \\ -1, & a < \end{cases}$$

- Target coding for perceptron

$$t = \begin{cases} +1, & \text{if } \mathcal{C}_1 \\ -1, & \text{if } \mathcal{C}_2 \end{cases}$$

The Perceptron Algorithm - Error Function

- Idea : Minimise total number of misclassified patterns.
- Problem : As a function of $\mathbf{w}$, this is piecewise constant and t
- Bet wa
- Perceptron Criterion : Add the errors for all patterns belonging to the set of misclassified patterns

$$E_P(\mathbf{w}) = -\sum_{n \in \mathcal{M}} \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}$$

Perceptron - Stochastic Gradient Descent

- Perceptron Criterion (with notation $\phi_n = \phi(\mathbf{x}_n)$)

$$E_P(\mathbf{w}) = -\sum_{n \in \mathcal{M}} \mathbf{w}^\top \phi_n t_n$$
$$\equiv E_P^{(n)}(\mathbf{w})$$

- On
  1.
  2. Update the weight vector $\mathbf{w}$ by

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E$$

where

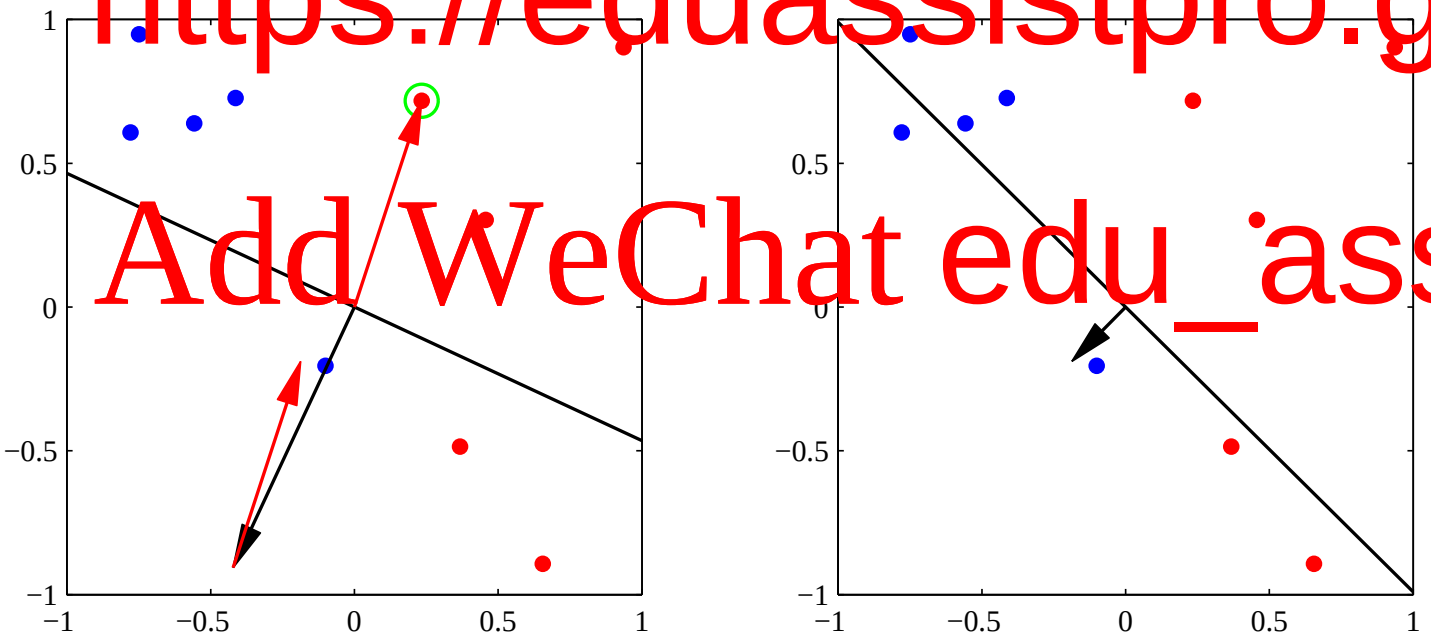
$$\nabla E_P^{(n)}(\mathbf{w}) = \begin{cases} -\phi_n t_n & \text{if } \left(\mathbf{w}^{(\tau)\top} \phi(\mathbf{x}_n) \cdot t_n\right) \leq 0 \\ 0 & \text{otherwise.} \end{cases}$$

- As $y(\mathbf{x}, \mathbf{w})$ is invariant to the norm of $\mathbf{w}$, we may set $\eta = 1$.

The Perceptron Algorithm - Update 1

Update of the perceptron weights from a misclassified pattern (green)

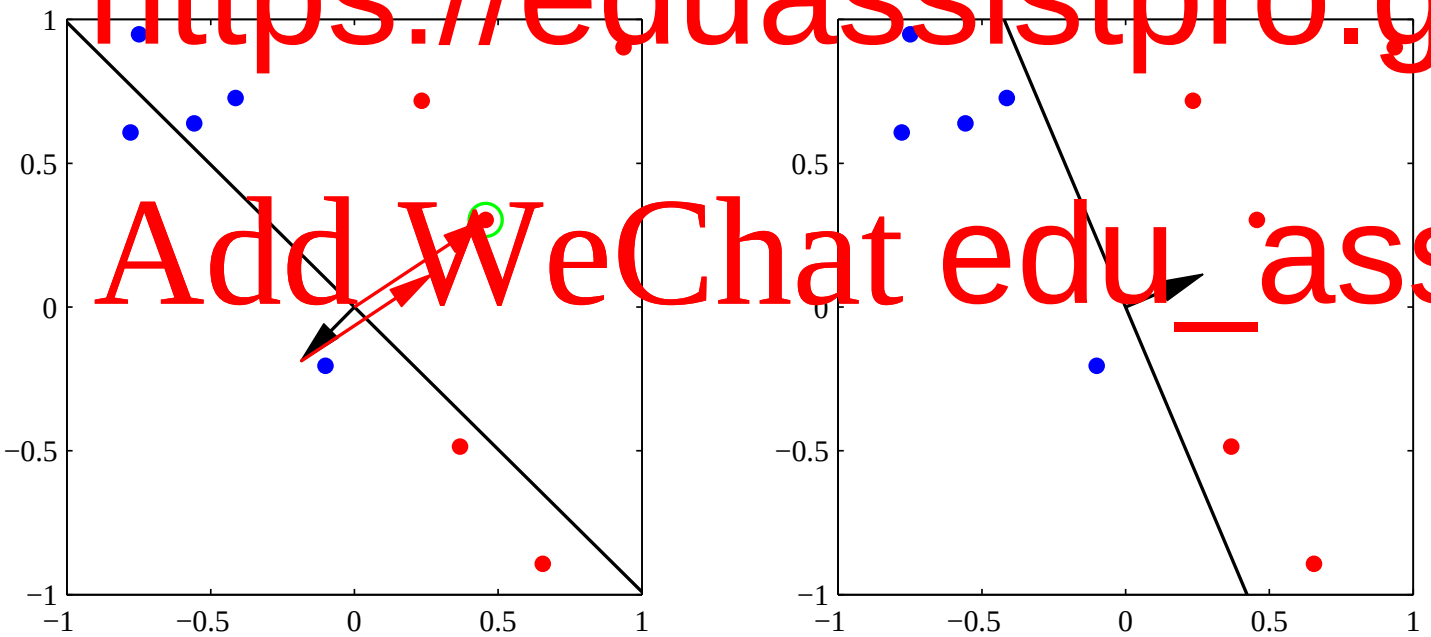
$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} + \boldsymbol{\varphi}_n t_n$$



The Perceptron Algorithm - Update 2

Update of the perceptron weights from a misclassified pattern (green)

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} + \varphi_n t_n$$



The Perceptron Algorithm - Convergence

- Does the algorithm converge ?
- For a single update step, letting $\eta = 1$, and considering the error from a single point,

$$\mathbf{w}^{(\tau+1)T} = \mathbf{w}^{(\tau)T} - \nabla E_n(\mathbf{w}^{(\tau)T})$$

  bec
  gra
  function.

- BUT: contributions to the error from the other mi
  patterns might have increased.

- AND: some correctly classified patterns migh
  misclassified.
- Perceptron Convergence Theorem : If the training set is linearly separable, the perceptron algorithm is guaranteed to find a solution in a finite number of steps.