Introduction to Statistical Machine Learning

Cheng Soon Ong & Christian Walder

Machine Learning Research Group
Data61 | CSIRO
and
College of Engineering and Computer Science
The Australian National University

Canberra
February – June 2017

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

Part II

Sparse Kernel Machines

Sparse Kernel Machines - Motivation

- Nonlinear kernels extended our toolbox of methods considerably
  - Wherever an inner product was used in an algorithm, we can replace it with a kernel.
  - Kernels act as a kind of 'similarity' measure and can be defined over graphs, sets, strings, and documents.
- But the kernel matrix is a square matrix with dimensions equal to the number of data points N
  - In order to calculate it, the kernel function must be evaluated for all pairs of training inputs.
- Sparse Kernel Machines implement learning algorithms where, for prediction, the kernels are only evaluated at a subset of the training data.

Data61 | CSIRO
The Australian National University

Maximum Margin Classifiers

- Return to the two-class classification with a linear model

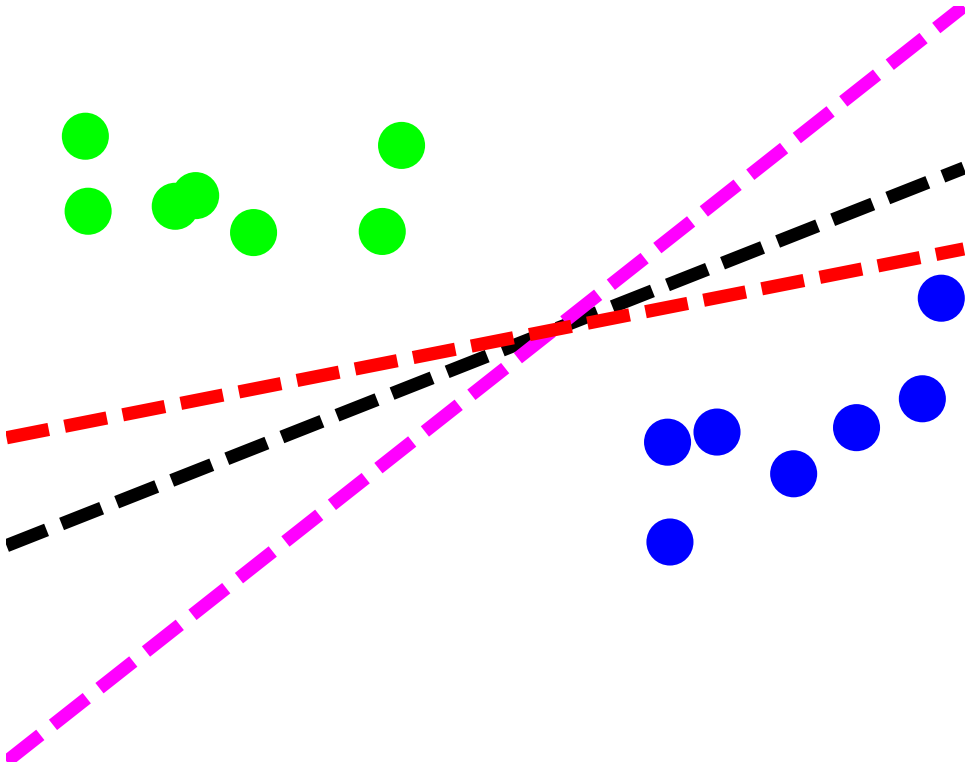
$$y(\mathbf{x}) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}) + b$$

  where $\phi(\mathbf{x})$ is some fixed feature space mapping, and b is the bias.
- Training data are N input vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$ with corresponding targets $t_1, \ldots, t_N$ where $t_n \in \{-1, +1\}$.
- The class of a new point is predicted as $\text{sign}\,(y(\mathbf{x}))$.
- Assume there exists a linear decision boundary. That means, there exist $\mathbf{w}$ and b such that

$$t_n \ \text{sign}\,(y(\mathbf{x}_n)) > 0 \qquad\qquad n = 1, \ldots, N.$$

The Multiple Separator problem

- There may exist many solutions $\mathbf{w}$ and b for which the linear classifier perfectly separates the two classes!



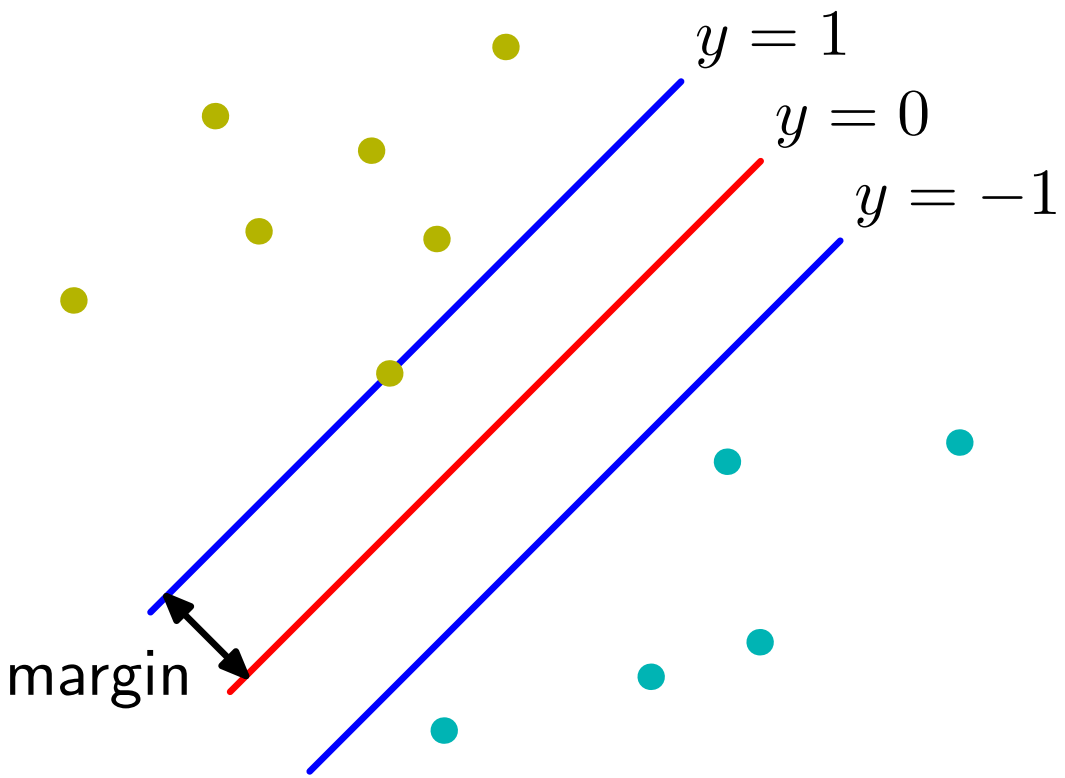
Maximum Margin Classifiers

- There may exist many solutions $\mathbf{w}$ and b for which the linear classifier perfectly separates the two classes.
- The perceptron algorithm can find a solution, but this depends on the initial choice of parameters.
- What is the decision boundary which results in the best generalisation (smallest generalisation error)?

Ong & Walder
Data61 | CSIRO
The Australian National University

Maximum Margin Classifiers

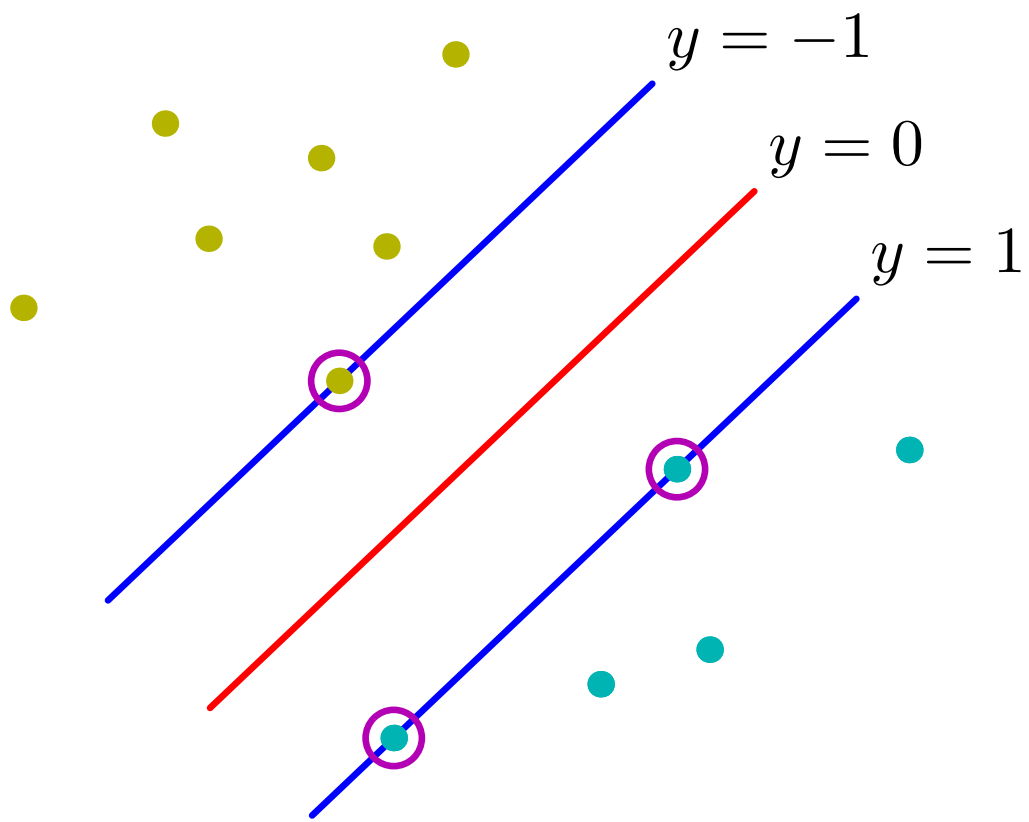
- The margin is the smallest distance between the decision boundary and any of the samples. (We will see later why $y = \pm 1$ on the margin.)



Ong & Walder
Data61 | CSIRO
The Australian National University

Maximum Margin Classifiers

- Support Vector Machines (SVMs) choose the decision boundary which maximises the margin.



Ong & Walder
Data61 | CSIRO
The Australian National University

Reminder - Linear Classification

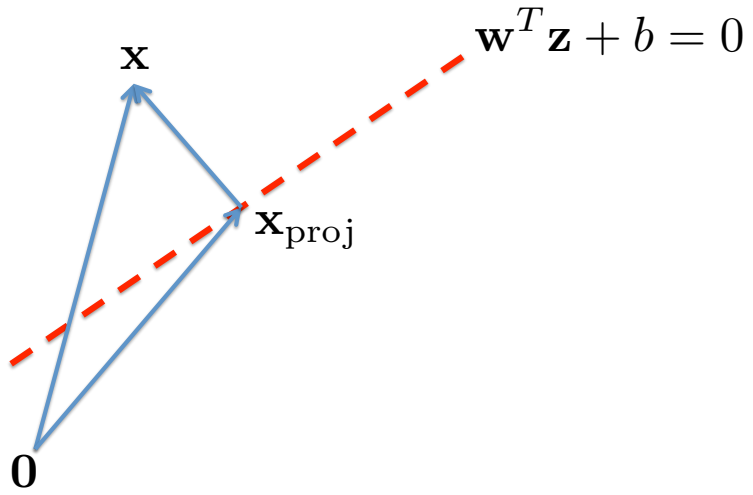
- For a linear model with weights $\mathbf{w}$ and bias b, the decision boundary is $\mathbf{w}^T\mathbf{z} + b = 0$
- Any $\mathbf{x}$ has projection $\mathbf{x}_{\text{proj}}$ on the boundary, and

$$\mathbf{x} - \mathbf{x}_{\text{proj}} = \lambda \cdot \mathbf{w}$$

since $\mathbf{w}$ is orthogonal to the decision boundary

- But we also have

$$\mathbf{w}^T\mathbf{x} - \mathbf{w}^T\mathbf{x}_{\text{proj}} = \mathbf{w}^T\mathbf{x} + b = \lambda \cdot \mathbf{w}^T\mathbf{w}$$



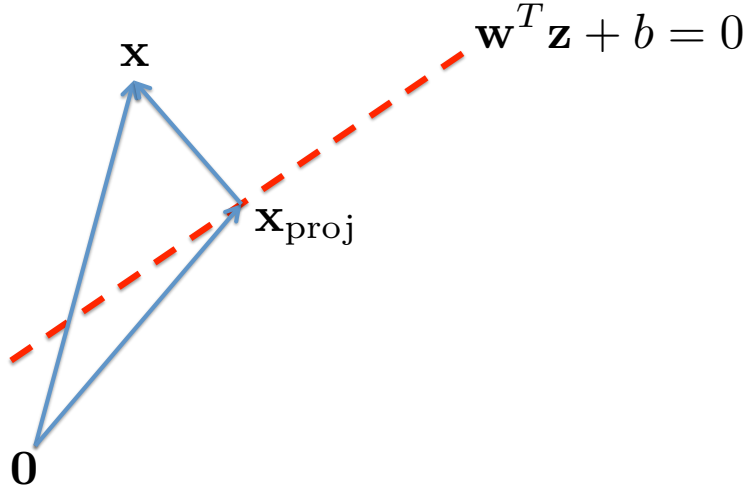
Ong & Walder
Data61 | CSIRO
The Australian National University

Reminder - Linear Classification

- So,

$$\begin{aligned}\|\mathbf{x} - \mathbf{x}_{\text{proj}}\| &= |\lambda| \cdot \|\mathbf{w}\| \\ &= \frac{|\mathbf{w}^T\mathbf{x} + b|}{\|\mathbf{w}\|}\end{aligned}$$

- The distance of $\mathbf{x}$ from the decision boundary is therefore $\frac{|\mathbf{w}^T\mathbf{x}+b|}{\|\mathbf{w}\|} = \frac{|y(\mathbf{x})|}{\|\mathbf{w}\|}$.

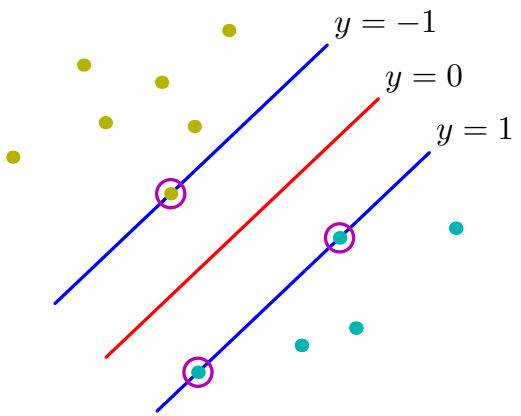


Ong & Walder
Data61 | CSIRO
The Australian National University

Maximum Margin Classifiers

- Support Vector Machines choose the decision boundary which maximises the smallest distance to samples in both classes.
- We calculated the distance of a point $\mathbf{x}$ from the hyperplane $y(\mathbf{x}) = 0$ as $|y(\mathbf{x})|/\|\mathbf{w}\|$.
- Perfect classification means $t_n y(\mathbf{x}_n) > 0$ for all n.
- Thus, the distance of $\mathbf{x}_n$ from the decision boundary is

$$\frac{t_n\, y(\mathbf{x}_n)}{\|\mathbf{w}\|} = \frac{t_n\, (\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n) + b)}{\|\mathbf{w}\|}$$



Ong & Walder
Data61 | CSIRO
The Australian National University

Maximum Margin Classifiers

- Support Vector Machines choose the decision boundary which maximises the smallest distance to samples in both classes.
- For the maximum margin solution, solve

$$\arg\max_{\mathbf{w},b} \left\{ \frac{1}{\|\mathbf{w}\|} \min_n \left[t_n \left(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n) + b \right) \right] \right\}.$$

- How can we solve this?

Maximum Margin Classifiers

- Maximum margin solution

$$\arg\max_{\mathbf{w},b}\left\{\frac{1}{\|\mathbf{w}\|}\min_{n}\left[t_n\left(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n)+b\right)\right]\right\}$$

or

$$\arg\max_{\mathbf{w},b,\gamma}\left\{\frac{\gamma}{\|\mathbf{w}\|}\right\} \text{ s.t. } t_n\left(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n)+b\right)\geq\gamma \qquad n=1,\ldots,N.$$

- By rescaling any $(\mathbf{w},b)$ by $1/\gamma$, we don't change the answer!
  - We can assume that $\gamma=1$
- The canonical representation for the decision hyperplane is therefore

$$t_n\left(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n)+b\right)\geq 1 \qquad n=1,\ldots,N.$$

©2016 Ong & Walder Data61 | CSIRO The Australian National University

Maximum Margin Classifiers

- Maximum margin solution

$$\arg\max_{\mathbf{w},b} \left\{ \frac{1}{\|\mathbf{w}\|} \min_n \left[t_n \left(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n) + b \right) \right] \right\}$$

- Transformed once

$$\arg\max_{\mathbf{w},b} \left\{ \frac{1}{\|\mathbf{w}\|} \right\} \quad \text{s.t. } t_n \left(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n) + b \right) \geq 1.$$

- Transformed again

$$\arg\min_{\mathbf{w},b} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 \right\} \quad \text{s.t. } t_n \left(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n) + b \right) \geq 1.$$

Lagrange Multipliers

Lagrange Multipliers

- Find the stationary points for a function $f(x_1, x_2)$ subject to one or more constraints on the variables x_1 and x_2 written in the form $g(x_1, x_2) = 0$.
- Direct approach
  1. Solve $g(x_1, x_2) = 0$ for one of the variables to get $x_2 = h(x_1)$.
  2. Insert the result into $f(x_1, x_2)$ to get a function of one variable $f(x_1, h(x_1))$.
  3. Find the stationary point(s) $x_1^\star$ of $f(x_1, h(x_1))$ with corresponding value $x_2^\star = h(x_1^\star)$.
- Finding $x_2 = h(x_1)$ may be hard.
- Symmetry in the variables x_1 and x_2 is lost.

Lagrange Multipliers

- To solve

$$\max_{\mathbf{x}} f(\mathbf{x}) \text{ s.t. } g(\mathbf{x}) = 0$$

introduce the Lagrangian function

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})$$

from which we get the constraint stationary conditions

$$\nabla_{\mathbf{x}} L(\mathbf{x}, \lambda) = \nabla f(\mathbf{x}) + \lambda \nabla g(\mathbf{x}) = 0$$

and the constraint itself

$$\frac{\partial L(\mathbf{x}, \lambda)}{\partial \lambda} = g(\mathbf{x}) = 0.$$

- This are D equations resulting from $\nabla_{\mathbf{x}} L(\mathbf{x}, \lambda)$ and one equation from $\frac{\partial L(\mathbf{x}, \lambda)}{\partial \lambda}$, together determining $\mathbf{x}^\star$ and λ.

Lagrange Multipliers - Example

- Given $f(x_1, x_2) = 1 - x_1^2 - x_2^2$ subject to the constraint $g(x_1, x_2) = x_1 + x_2 - 1 = 0$.
- Define the Lagrangian function

$$L(\mathbf{x}, \lambda) = 1 - x_1^2 - x_2^2 + \lambda(x_1 + x_2 - 1).$$

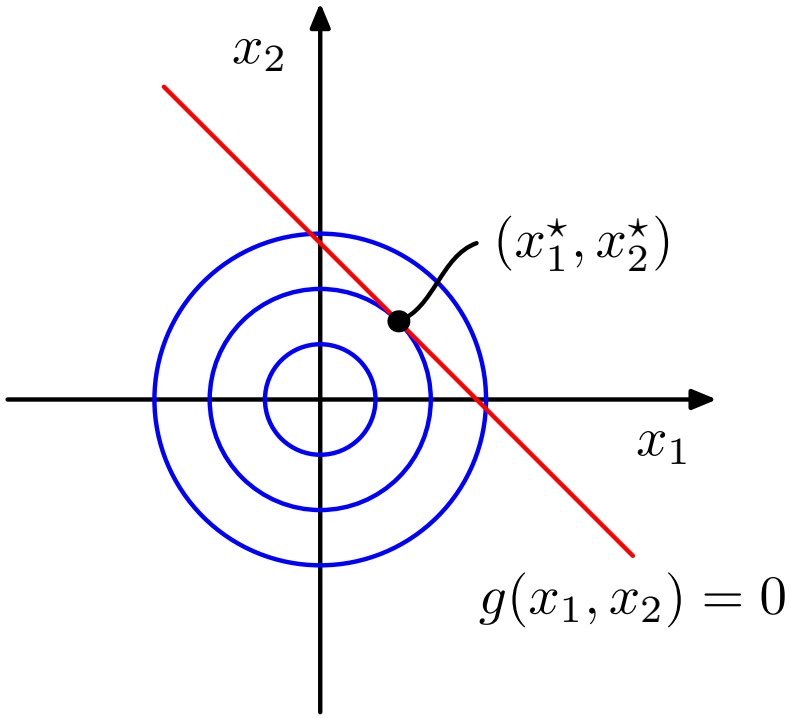
- A stationary solution with respect to x_1, x_2, and λ must satisfy

$$\begin{aligned} -2x_1 + \lambda &= 0 \\ -2x_2 + \lambda &= 0 \\ x_1 + x_2 - 1 &= 0. \end{aligned}$$

- Therefore $(x_1^\star, x_2^\star) = (\frac{1}{2}, \frac{1}{2})$ and $\lambda = 1$.

Lagrange Multipliers - Example

- Given $f(x_1, x_2) = 1 - x_1^2 - x_2^2$ subject to the constraint $g(x_1, x_2) = x_1 + x_2 - 1 = 0$.
- Lagrangian $L(\mathbf{x}, \lambda) = 1 - x_1^2 - x_2^2 + \lambda(x_1 + x_2 - 1)$
- $(x_1^\star, x_2^\star) = (\frac{1}{2}, \frac{1}{2})$.



Ong & Walder
Data61 | CSIRO
The Australian National University

Lagrange Multipliers for Inequality Constraints

- To solve

$$\max_{\mathbf{x}} f(\mathbf{x}) \text{ s.t. } g(\mathbf{x}) \geq 0$$

  define the Lagrangian

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})$$

- Solve for $\mathbf{x}$ and λ subject to the constraints (Karush-Kuhn-Tucker or KKT conditions)

$$g(\mathbf{x}) \geq 0$$
$$\lambda \geq 0$$
$$\lambda g(\mathbf{x}) = 0$$

Lagrange Multipliers for General Case

- Maximise $f(\mathbf{x})$ subject to the constraints $g_j(\mathbf{x}) = 0$ for $j = 1, \ldots, J$, and $h_k(\mathbf{x}) \geq 0$ for $k = 1, \ldots, K$.
- Define the Lagrange multipliers $\{\lambda_j\}$ and $\{\mu_k\}$, and the Lagrangian

$$L(\mathbf{x}, \{\lambda_j\}, \{\mu_k\}) = f(\mathbf{x}) + \sum_{j=1}^{J} \lambda_j \, g_j(\mathbf{x}) + \sum_{k=1}^{K} \mu_k \, h_k(\mathbf{x}).$$

- Solve for $\mathbf{x}$, $\{\lambda_j\}$, and $\{\mu_k\}$ subject to the constraints (Karush-Kuhn-Tucker or KKT conditions)

$$\mu_k \geq 0$$
$$\mu_k \, h_k(\mathbf{x}) = 0$$

  for $k = 1, \ldots, K$.
- For minimisation of $f(\mathbf{x})$, change the sign in front of the Lagrange multipliers.

SVMs and Maximum Margin Classifiers: Redux

Ong & Walder
Data61 | CSIRO
The Australian National University

Maximum Margin Classifiers

- Quadratic Programming (QP) problem: minimise a quadratic function subject to linear constraints.

$$\arg\min_{\mathbf{w},b} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 \right\} \quad \text{s.t. } t_n \left(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n) + b\right) \geq 1.$$

- Introduce Lagrange multipliers $\{a_n \geq 0\}$ to get the Lagrangian

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N} a_n \left\{ t_n \left(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n) + b\right) - 1 \right\}$$

  - note negative sign since minimisation problem

Maximum Margin Classifiers

- Lagrangian

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N} a_n \left\{ t_n \left(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n) + b \right) - 1 \right\}.$$

- Derivatives with respect to $\mathbf{w}$ and b are

$$\mathbf{w} = \sum_{n=1}^{N} a_n \, t_n \, \boldsymbol{\phi}(\mathbf{x}_n) \qquad\qquad 0 = \sum_{n=1}^{N} a_n \, t_n$$

- Dual representation (again QP problem): maximise

$$\widetilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a_n \, a_m \, t_n \, t_m \, k(\mathbf{x}_n, \mathbf{x}_m)$$

$$\sum_{n=1}^{N} a_n \, t_n = 0$$

$$a_n \geq 0 \qquad\qquad n = 1, \ldots, N.$$

Maximum Margin Classifiers

- Dual representation (again QP problem): maximise

$$\widetilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n\, a_m\, t_n\, t_m\, k(\mathbf{x}_n, \mathbf{x}_m)$$

$$\sum_{n=1}^{N} a_n\, t_n = 0$$

$$a_n \geq 0 \qquad\qquad n = 1, \ldots, N.$$

- Vector form: maximise

$$\widetilde{L}(\mathbf{a}) = \mathbf{1}^T\mathbf{a} - \frac{1}{2}\mathbf{a}^T\mathbf{Q}\mathbf{a}$$

$$\mathbf{a}^T\mathbf{t} = 0$$

$$a_n \geq 0 \qquad\qquad n = 1, \ldots, N.$$

where

$$\mathbf{Q}_{nm} = t_n \cdot t_m \cdot k(\mathbf{x}_n, \mathbf{x}_m)$$

Maximum Margin Classifiers - Prediction

- Predict via

$$y(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b = \sum_{n=1}^{N} a_n \, t_n \, k(\mathbf{x}, \mathbf{x}_n) + b$$

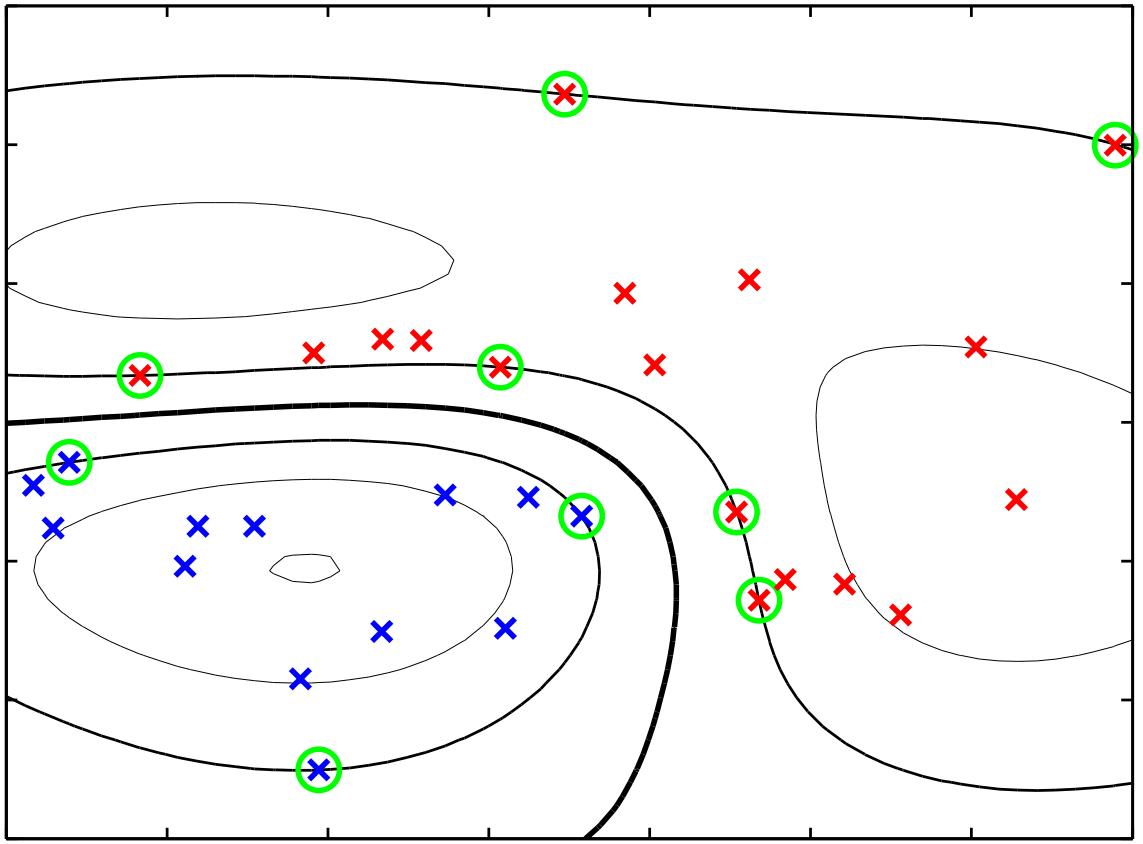
- The Karush-Kuhn-Tucker (KKT) conditions are

$$\begin{aligned} a_n &\geq 0 \\ t_n \, y(\mathbf{x}_n) - 1 &\geq 0 \\ a_n \left\{ t_n \, y(\mathbf{x}_n) - 1 \right\} &= 0. \end{aligned}$$

  - Therefore, either $a_n = 0$ or $t_n \, y(\mathbf{x}_n) - 1 = 0$.
  - If $a_n = 0$, no contribution to the prediction!
- After training, use only the set $\mathcal{S}$ of points for which $a_n > 0$ (and therefore $t_n \, y(\mathbf{x}_n) = 1$) holds.
- $\mathcal{S}$ contains only the support vectors.

Maximum Margin Classifiers - Support Vectors

- $\mathcal{S}$ contains only the support vectors.



Decision and margin boundaries for a two-class problem using Gaussian kernel functions.

Maximum Margin Classifiers - Support Vectors

- Now for the bias b.
- Observe that for the support vectors, $t_n\, y(\mathbf{x}_n) = 1$.
- Therefore, we find

$$t_n \left(\sum_{m \in \mathcal{S}} a_m\, t_m\, k(\mathbf{x}_n, \mathbf{x}_m) + b \right) = 1$$

  and can use any support vector to calculate b.
- Numerically more stable: Multiply by t_n, observe $t_n^2 = 1$, and average over all support vectors.

$$b = \frac{1}{N_{\mathcal{S}}} \sum_{n \in \mathcal{S}} \left(t_n - \sum_{m \in \mathcal{S}} a_m\, t_m\, k(\mathbf{x}_n, \mathbf{x}_m) \right)$$

SVMs: Summary

- Maximise the dual objective

$$\widetilde{L}(\mathbf{a}) = \mathbf{1}^T\mathbf{a} - \frac{1}{2}\mathbf{a}^T\mathbf{Q}\mathbf{a}$$
$$\mathbf{a}^T\mathbf{t} = 0$$
$$a_n \geq 0 \qquad\qquad n = 1, \ldots, N.$$

- Make predictions via

$$y(\mathbf{x}) = \sum_{n=1}^{N} a_n\, t_n\, k(\mathbf{x}, \mathbf{x}_n) + b$$

  - Dependence only on data points with $a_n > 0$

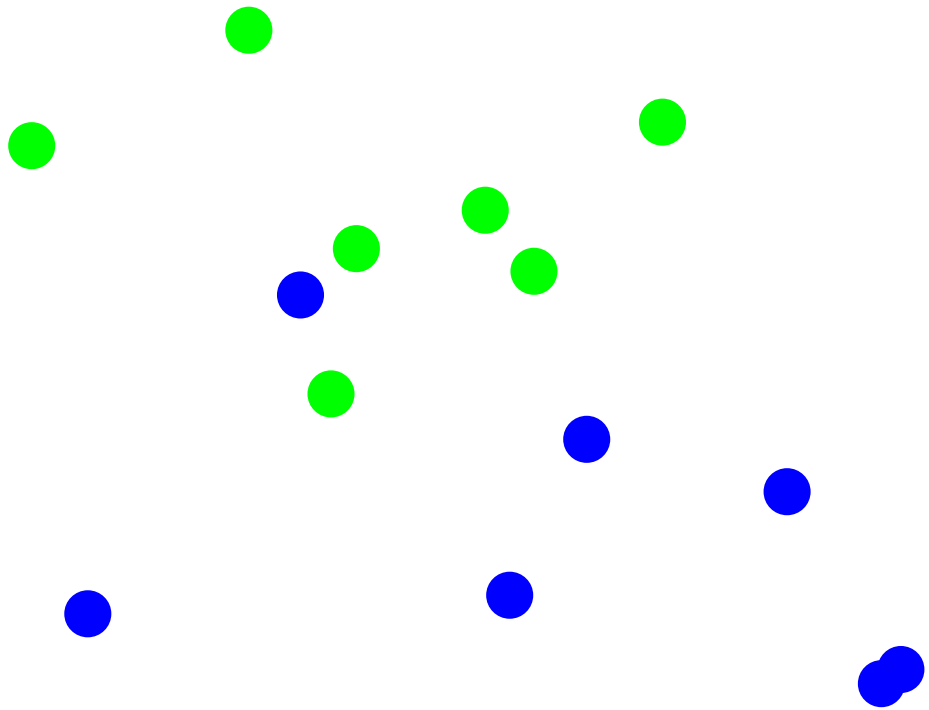
Ong & Walder
Data61 | CSIRO
The Australian National University

Soft Margin SVMs: Non-Separable Case

Ong & Walder
Data61 | CSIRO
The Australian National University

Non-Separable Data

- What if the training data in the feature space are not linearly separable?
  - Can we find a separator that makes fewest possible errors?



Non-Separable Data

- What if the training data in the feature space are not linearly separable?
  - Allow some data points to be on the 'wrong side' of the decision boundary.
  - Increase a penalty with distance from the decision boundary.
- Assume, the penalty increases linearly with distance.
- Introduce slack variable $\xi_n \geq 0$ for each data point n .

$$\xi_n \begin{cases} = 0, & \text{data point is correctly classified and} \\ & \quad \text{on margin boundary or beyond} \\ < 1, & \text{data point is in correct margin} \\ = 1, & \text{data point is on the decision boundary} \\ > 1, & \text{data point is misclassified} \end{cases}$$

Data61 | CSIRO
The Australian National University

Non-Separable Data

- Constraints of the separable classification

$$t_n\, y(\mathbf{x}_n) \geq 1, \qquad n = 1, \ldots, N$$

change now to

$$t_n\, y(\mathbf{x}_n) \geq 1 - \xi_n, \qquad n = 1, \ldots, N$$

- In sum, we have the constraints

$$\xi_n \geq 0$$
$$\xi_n \geq 1 - t_n\, y(\mathbf{x}_n).$$

Ong & Walder
Data61 | CSIRO
The Australian National University

Non-Separable Data

- Find now

$$\min_{\mathbf{w},\xi} C \sum_{n=1}^{N} \xi_n + \frac{1}{2}\|\mathbf{w}\|^2 \text{ s.t. } \xi_n \geq 0$$

$$\xi_n \geq 1 - t_n\, y(\mathbf{x}_n).$$

  where C controls the trade-off between the slack variable penalty and the margin.
  - Minimise the total slack associated with all data points
- Any misclassified point contributes $\xi_n > 0$
  - therefore $\sum_{n=1}^{N} \xi_n$ is an upper bound on the number of misclassified points.

Ong & Walder
Data61 | CSIRO
The Australian National University

Non-Separable Data

- The Lagrangian with the Lagrange multipliers $\mathbf{a} = (a_1, \ldots, a_N)^T$ and $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_N)^T$ is now

$$L(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}, \boldsymbol{\mu}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{N}\xi_n - \sum_{n=1}^{N} a_n\{t_n\, y(\mathbf{x}_n) - 1 + \xi_n\} - \sum_{n=1}^{N}\mu_n\xi_n.$$

- The KKT conditions for $n = 1, \ldots, N$ are

$$\begin{aligned} a_n &\geq 0 \\ t_n\, y(\mathbf{x}_n) - 1 + \xi_n &\geq 0 \\ a_n(t_n\, y(\mathbf{x}_n) - 1 + \xi_n) &= 0 \\ \mu_n &\geq 0 \\ \xi_n &\geq 0 \\ \mu_n\xi_n &= 0. \end{aligned}$$

Ong & Walder
Data61 | CSIRO
The Australian National University

Non-Separable Data

- The dual Lagrangian after eliminating $\mathbf{w}$, b, $\boldsymbol{\xi}$, and $\boldsymbol{\mu}$ is then

$$\widetilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a_n \, a_m \, t_n \, t_m \, k(\mathbf{x}_n, \mathbf{x}_m)$$

$$\sum_{n=1}^{N} a_n \, t_n = 0$$

$$0 \leq a_n \leq C \qquad\qquad n = 1, \ldots, N.$$

- The only change from the separable case, is the box constraint via the parameter C.

Data61 | CSIRO
The Australian National University

Non-Separable Data

- Equivalent formulation by Schölkopf et. al. (2000)

$$\widetilde{L}(\mathbf{a}) = -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n\, a_m\, t_n\, t_m\, k(\mathbf{x}_n, \mathbf{x}_m)$$

$$\sum_{n=1}^{N} a_n\, t_n = 0$$

$$0 \leq a_n \leq 1/N$$

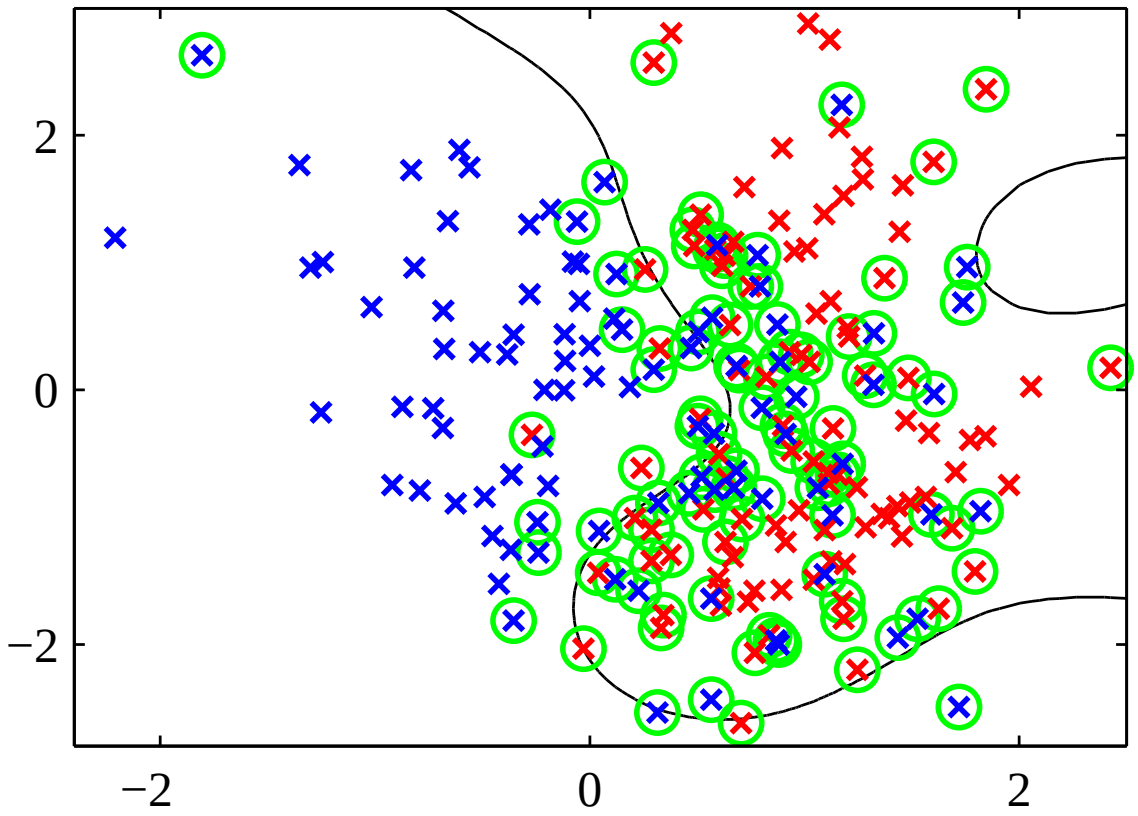
$$\sum_{n=1}^{N} a_n \geq \nu \qquad\qquad n = 1, \ldots, N.$$

where ν is both

1. an upper bound on the fraction of margin errors, and,
2. a lower bound on the fraction of support vectors.

Ong & Walder
Data61 | CSIRO
The Australian National University

Non-Separable Data



The ν-SVM algorithm using Gaussian kernels $\exp(-\gamma\|\mathbf{x}-\mathbf{x}'\|^2)$ with $\gamma = 0.45$ applied to a nonseparable data set in two dimensions. Support vectors are indicated by circles.

Ong & Walder
Data61 | CSIRO
The Australian National University

Support Vector Machines - Limitations

- Output are decisions, not posterior probabilities.
- Extension to classification with more than two classes is problematic.
- There is a complexity parameter C (or ν) which must be found (e.g. via cross-validation).
- Predictions are expressed as linear combinations of kernel functions that are centered on the training points.
- Kernel matrix is required to be positive definite.

Ong & Walder
Data61 | CSIRO
The Australian National University

Relevance Vector Machines

Ong & Walder
Data61 | CSIRO
The Australian National University

Relevance Vector Machines

- Assume a Bayesian model as in Bayesian Linear Regression with the probability of the target t given by

$$p(t \mid \mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(t \mid y(\mathbf{x}), \beta^{-1}),$$

and the linear model

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

with fixed nonlinear basis functions $\boldsymbol{\phi}(\mathbf{x})$ (including a constant term to accomodate for the bias).

- But now the prior over $\mathbf{w}$ includes a different precision α_i for each of the components of $\mathbf{w}$

$$p(\mathbf{w} \mid \boldsymbol{\alpha}) = \prod_{i=1}^{M} \mathcal{N}(w_i \mid 0, \alpha_i^{-1}).$$

Relevance Vector Machines

- When using this prior for the weights $\mathbf{w}$

$$p(\mathbf{w} \mid \boldsymbol{\alpha}) = \prod_{i=1}^{M} \mathcal{N}(w_i \mid 0, \alpha_i^{-1})$$

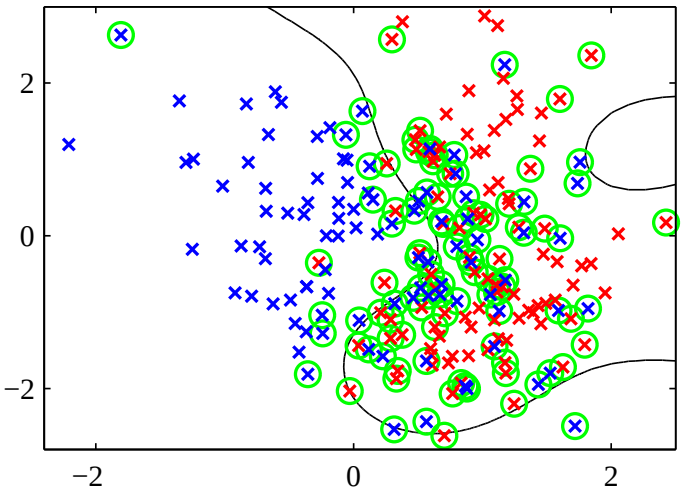
  and maximising the evidence with respect to the hyperparameters α_i, a significant portion of the α_i will go to infinity.
- The corresponding w_i will therefore be concentrated at zero, and play no role for the prediction.

Ong & Walder
Data61 | CSIRO
The Australian National University

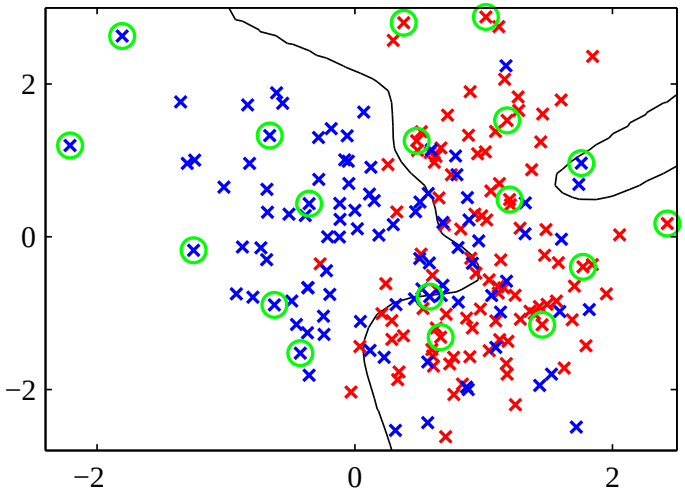
Relevance Vector Machines

- The Relevance Vector Machine iteratively calculates the α_i and β from the training data by optimising a nonconvex function.
- Disadvantages
  1. Nonconvex function means that the algorithm can get stuck in some local minimum.
  2. Training times can be longer than for SVM.
- Advantages
  1. As a Bayesian method, the parameters governing complexity and noise are determined from the input data. (Compare to SVM where we needed cross-validation).
  2. Number of relevance vectors is much smaller than the number of support vectors in a SVM.
  3. Therefore, prediction is much faster than with SVM.
  4. Extends well to multiclass learning.

Relevance Vector Machines



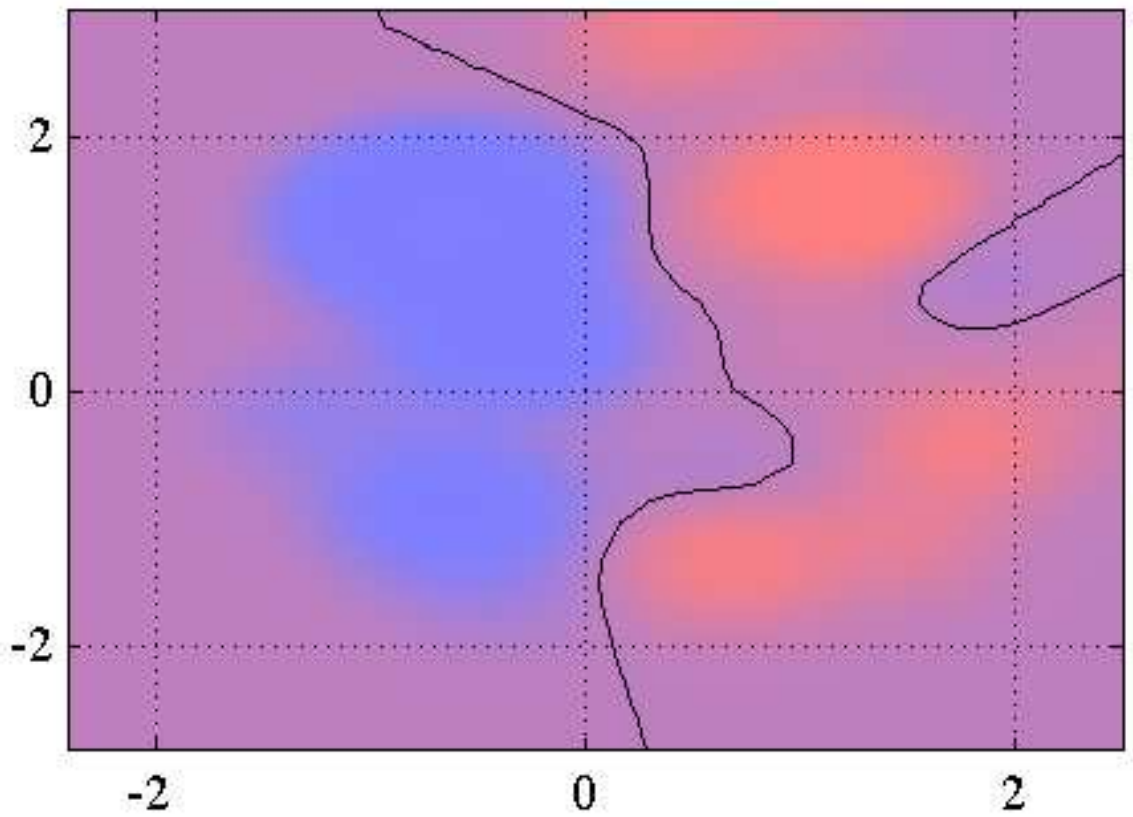
Support Vector Machine



Relevance Vector Machine

Ong & Walder
Data61 | CSIRO
The Australian National University

Relevance Vector Machines



Decision boundaries and posterior probability for a two-class problem found by a Relevance Vector Machine.