Support Vector Machines

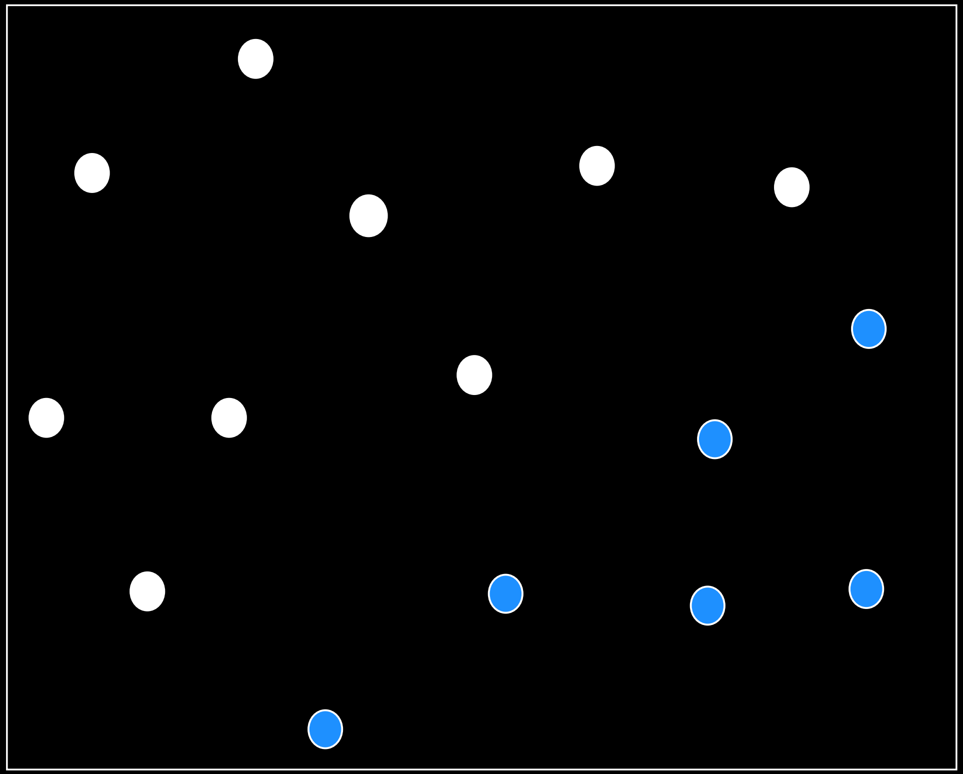
Presented by:

Stephen Sinclair

MUMT 611 Winter 2006, McGill University

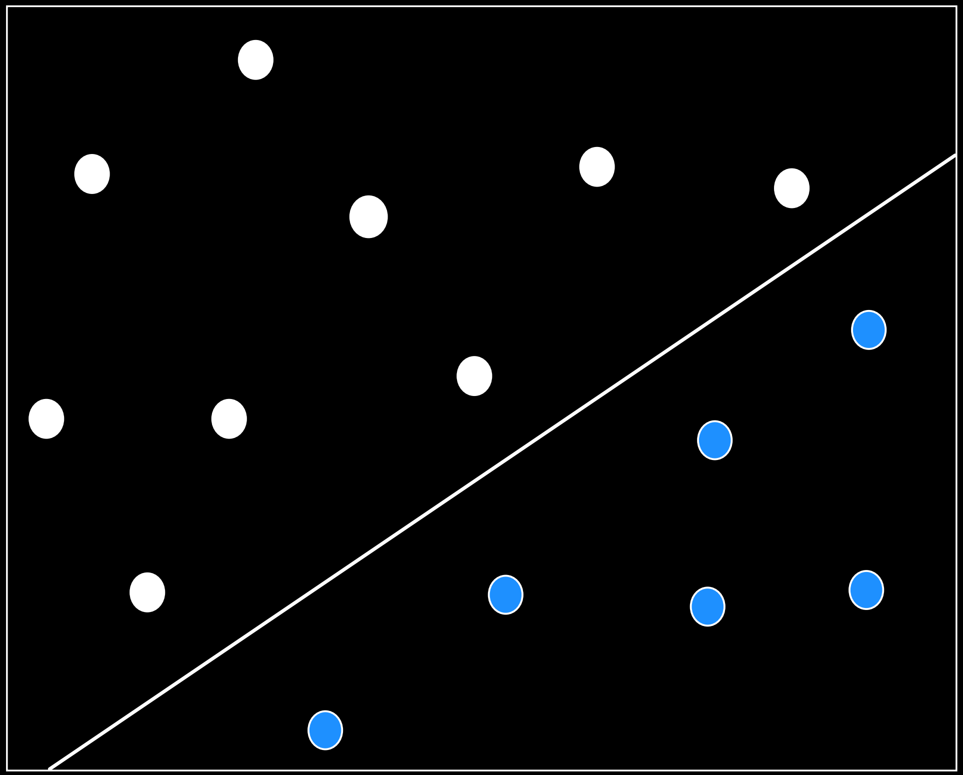
Support Vector Machines

- Allow classification of multi-dimensional data sets.
- Perform well on data sets with high dimensionality.
- Divide data into sections by defining a hyperplane.
  - Visualize as drawing a line on a graph.



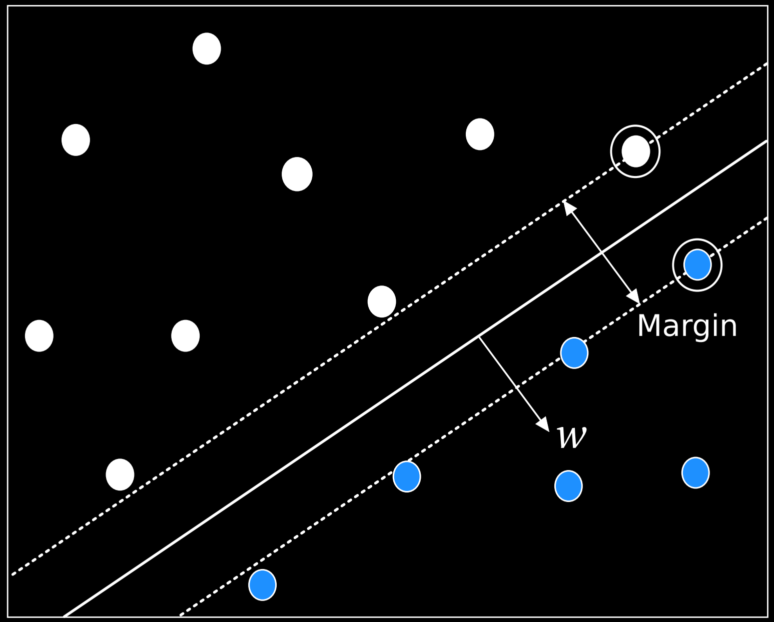
Support Vector Machines

- Allow classification of multi-dimensional data sets.
- Perform well on data sets with high dimensionality.
- Divide data into sections by defining a hyperplane.
  - Visualize as drawing a line on a graph.



Support Vectors

- Points which “hold up” the linear divider



Adapted from (Burges 1998)

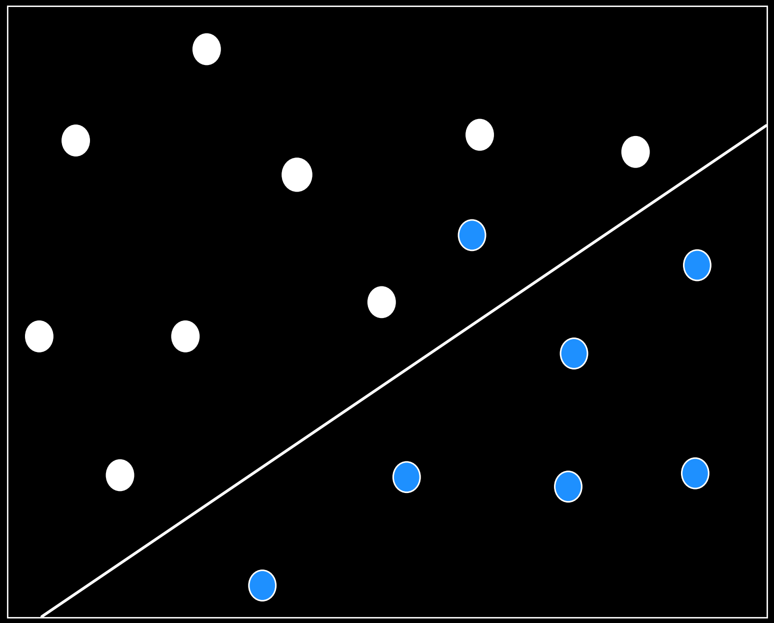
- Constraints:
    - $\mathbf{x}_i \cdot \mathbf{w} + b \geq +1$, for $y_i = +1$ (white dots)
    - $\mathbf{x}_i \cdot \mathbf{w} + b \leq -1$, for $y_i = -1$ (blue dots)

Solving for $\mathbf{w}$

- Lagrange equations:
  Minimize: $L_P \equiv \frac{1}{2}||w||^2 - \sum_{i=1}^{l} \alpha_i y_i(\mathbf{x}_i - \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i$
  Maximize: $L_D = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$
- This is a “convex quadratic programming problem”.
- Support vectors are those for which $\alpha_i > 0$
  - For all others, $\alpha_i = 0$
- The SVM can be solved by solving the associated Karush-Kuhn-Tucker (KKT) conditions, a property of convex optimization problems.
- In real world, numerical methods are needed. Only specific conditions allow for analytical solutions.

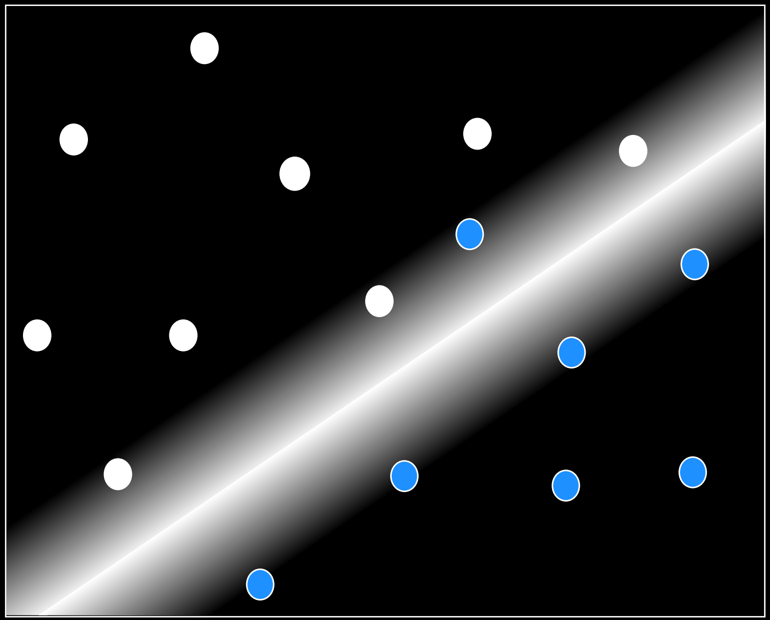
Non-separable data

- What do if there is no clear line?



Non-separable data

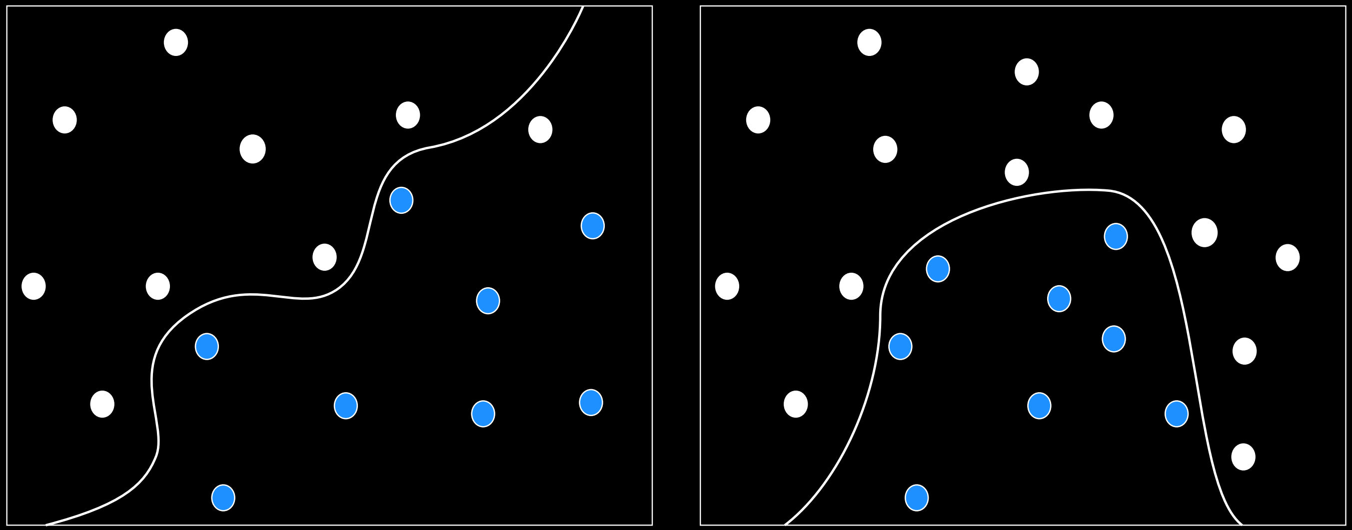
- What do if there is no clear line?



- Add a “slack” variable
  - $\mathbf{x}_i \cdot \mathbf{w} + b \geq +1 - \xi_i$, for $y_i = +1$
  - $\mathbf{x}_i \cdot \mathbf{w} + b \leq -1 + \xi_i$, for $y_i = -1$

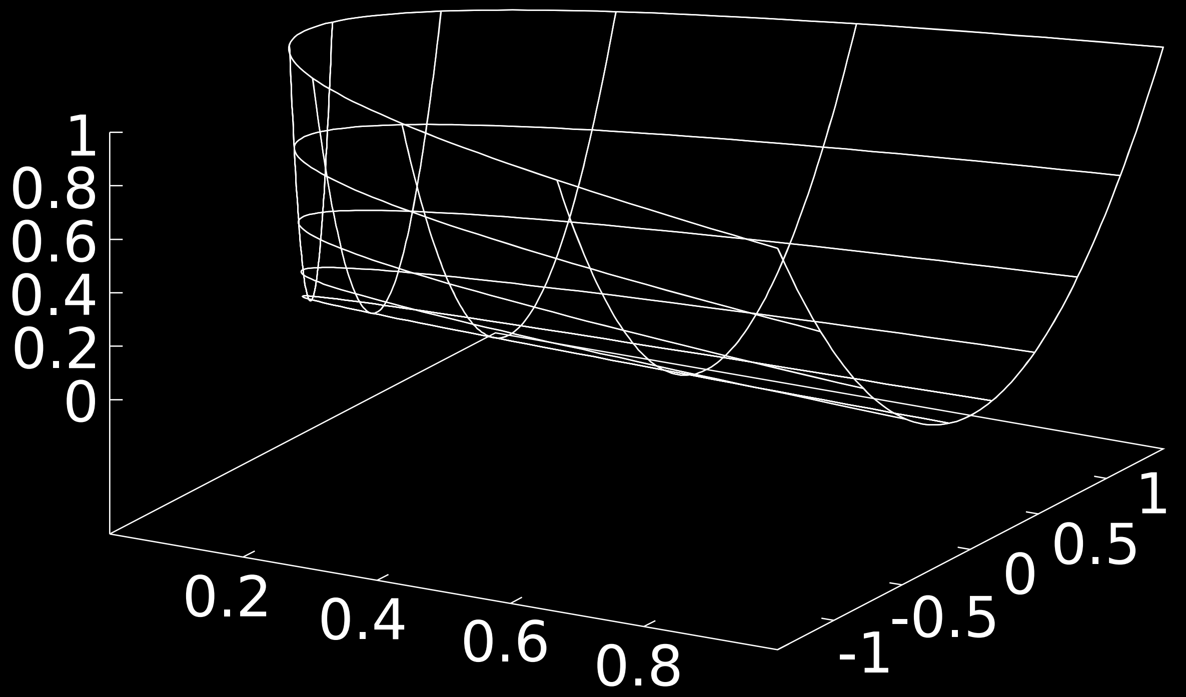
Non-linear classifiers

- The hyperplane classifier presented so far is *linear*.
- Some data can be separable in a non-linear way.
- To do so, we can use the “kernel trick”.



Non-linear classifiers

- First, a non-linear projection onto a higher-dimensional space
  - $\Phi : R^d \mapsto H$



(Burges 1998)

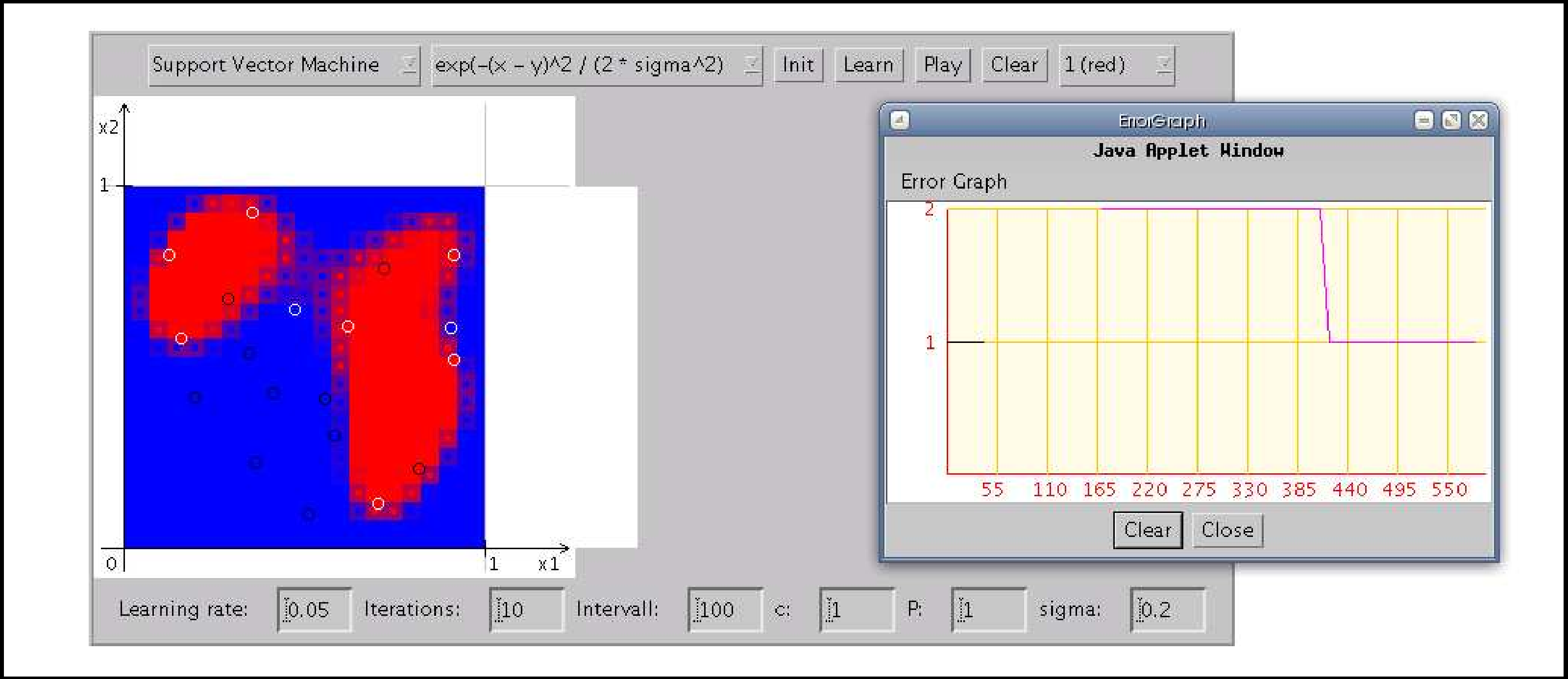
Non-linear classifiers

- Note that H is a Hilbert space, and may be of infinite dimensions.
- Now the training algorithm only depends on dot products in H.
  - $\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$
- Then, we can use a “kernel function” K, such that $K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$
- Kernels can be determined according to Mercer’s condition
- Some kernels:
  - $K(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y} + 1)^p$
  - $K(\mathbf{x}, \mathbf{y}) = e^{\frac{-||\mathbf{x}-\mathbf{y}||^2}{2\sigma^2}}$
    (radial basis function)
  - $K(\mathbf{x}, \mathbf{y}) = tanh(k\mathbf{x} \cdot \mathbf{y} - \delta)$
    (2-layer neural network)

Java Applet

- Play with two types of 2-dimensional SVMs.

  `http://diwww.epfl.ch/mantra/tutorial/english/svm/html/index.html`



References

Burges, C. J. C.. 1998. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery 2*(2), 121–167.