BITS Pilani
Pilani Campus

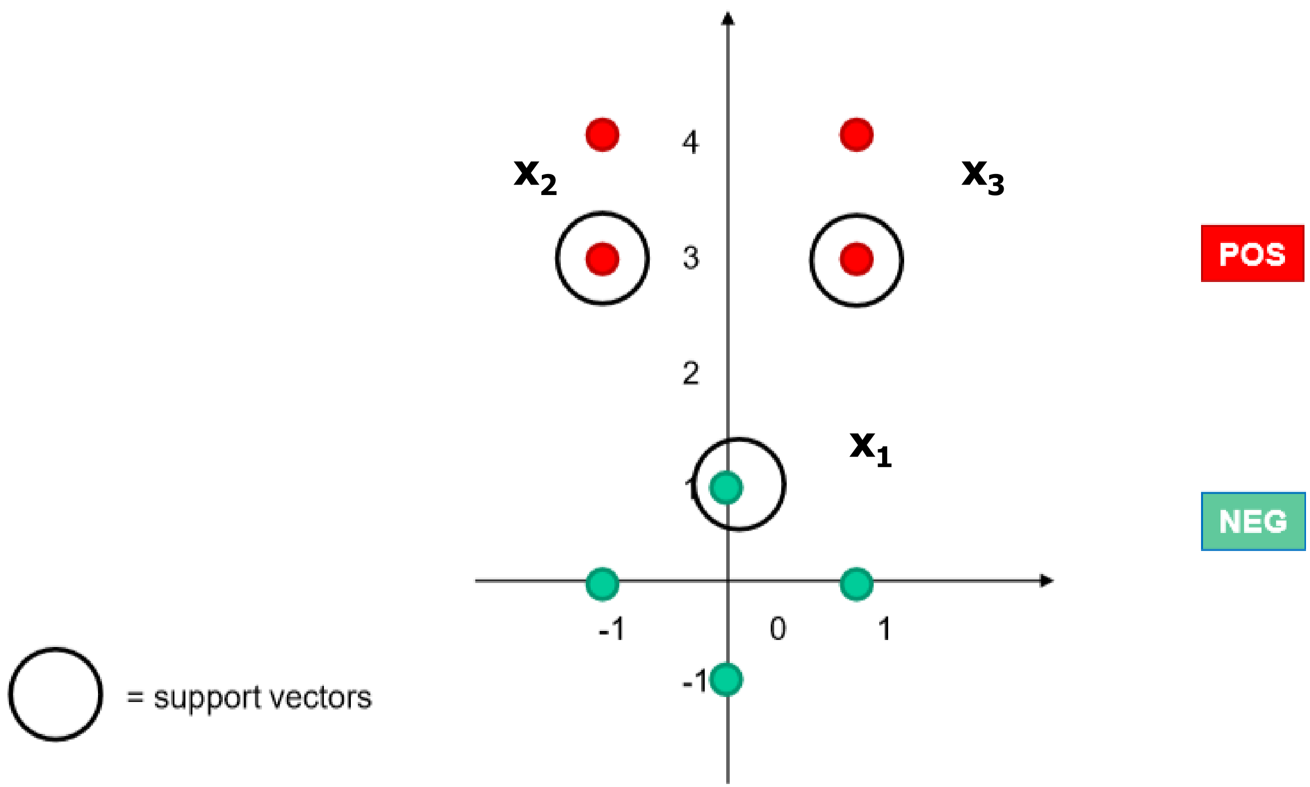
Machine Learning

**** CLZG565

Support Vector Machine

Raja vadhana P
Assistant Professor,
BITS - CSIS

Problem Type – 1 Linear SVM



Example adapted from Dan Ventura

Problem Type – 1 Linear SVM

Solving for α

- We know that for the support vectors, f(x) = 1 or -1 exactly
- Add a 1 in the feature representation for the bias

- The support vectors have coordinates and labels:
  - $\mathbf{x_1 = [0\ 1\ 1],\ y_1 = -1}$
  - $\mathbf{x_2 = [-1\ 3\ 1],\ y_2 = +1}$
  - $\mathbf{x_3 = [1\ 3\ 1],\ y_3 = +1}$

- Thus we can form the following system of linear equations:

Linear SVM Problem

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$

$$b = y_i - \mathbf{w} \cdot \mathbf{x}_i$$

$$f(x) = \sum_i \alpha_i y_i (\mathbf{x}_i^\top \mathbf{x}) + b$$

innovate achieve lead

$\alpha_i\,[-1\,(\mathbf{w}\,.\,\mathbf{x_i} + b)] = -1$

$\alpha_i\,[+1\,(\mathbf{w}\,.\,\mathbf{x_i} + b)] = 1$

I Select the support vectors:

$x_1 = [0\ 1\ 1],\ y_1 = -1$

$x_2 = [-1\ 3\ 1],\ y_2 = +1$

$x_3 = [1\ 3\ 1],\ y_3 = +1$

II Substitute in Lagrangian function: $L(w, b, \alpha_i) = \Sigma\, \alpha_i - \frac{1}{2}\left(\sum_i \sum_j \alpha_i \alpha_j y_i y_j\, xi\, .\, xj\right)$

$L(w, b, \alpha_i) = \alpha_1 + \alpha_2 + \alpha_3 - \frac{1}{2}(\alpha_1 \alpha_1 y_1 y_1 x_1 . x_1 + \alpha_2 \alpha_2 y_2 y_2 x_2 . x_2 + \alpha_3 \alpha_3 y_3 y_3 x_3 . x_3$
$+ 2\, \alpha_1 \alpha_2 y_1 y_2 x_1 . x_2 + 2\, \alpha_1 \alpha_3 y_1 y_3 x_1 . x_3 + 2\, \alpha_2 \alpha_3 y_2 y_3 x_2 . x_3)$

$= \alpha_1 + \alpha_2 + \alpha_3 - \frac{1}{2}(\alpha_1 \alpha_1 x_1 . x_1 + \alpha_2 \alpha_2 x_2 . x_2 + \alpha_3 \alpha_3 x_3 . x_3 - 2\, \alpha_1 \alpha_2 x_1 . x_2 - 2\, \alpha_1 \alpha_3 x_1 . x_3 + 2\, \alpha_2 \alpha_3 x_2 . x_3)$

$= \alpha_1 + \alpha_2 + \alpha_3 - \frac{1}{2}(\alpha_1 \alpha_1 [0\ 1\ 1] . [0\ 1\ 1] + \alpha_2 \alpha_2 [-1\ 3\ 1] . [-1\ 3\ 1] + \alpha_3 \alpha_3 [1\ 3\ 1] . [1\ 3\ 1]$
$- 2\, \alpha_1 \alpha_2 [0\ 1\ 1] . [-1\ 3\ 1] - 2\, \alpha_1 \alpha_3 [0\ 1\ 1] . [1\ 3\ 1] + 2\, \alpha_2 \alpha_3 [-1\ 3\ 1] . [1\ 3\ 1])$

III Find the Unconstrained Optimization Function:

$L(w, b, \alpha_i) = \alpha_1 + \alpha_2 + \alpha_3 - \frac{1}{2}(2\alpha_1 \alpha_1 + 11\, \alpha_2 \alpha_2 + 11\, \alpha_3 \alpha_3 - 8\, \alpha_1 \alpha_2 - 8\, \alpha_1 \alpha_3 + 18\, \alpha_2 \alpha_3)$

Linear SVM Problem

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$

$$b = y_i - \mathbf{w} \cdot \mathbf{x}_i$$

$$f(x) = \sum_i \alpha_i y_i (\mathbf{x}_i^\top \mathbf{x}) + b$$

IV Gradient of the Lagrangian:

α_i [-1 ($\mathbf{w}$. $\mathbf{x_i}$ + b)] = -1
α_i [+1 ($\mathbf{w}$. $\mathbf{x_i}$ + b)] = 1

$L(w, b, \alpha_i) = \alpha_1 + \alpha_2 + \alpha_3 - \frac{1}{2}(2\alpha_1 \alpha_1 + 11\, \alpha_2 \alpha_2 + 11\, \alpha_3 \alpha_3 - 8\, \alpha_1 \alpha_2 - 8\, \alpha_1 \alpha_3 + 18\, \alpha_2 \alpha_3)$

$\frac{\partial L}{\partial \alpha_1} = 0 \rightarrow -2\alpha_1 + 4\alpha_2 + 4\alpha_3 = -1$

$\frac{\partial L}{\partial \alpha_2} = 0 \rightarrow -4\alpha_1 + 11\alpha_2 + 9\alpha_3 = +1$

$\frac{\partial L}{\partial \alpha_3} = 0 \rightarrow -4\alpha_1 + 9\alpha_2 + 11\alpha_3 = +1$

V Solve the simultaneous linear equation and find the lagrange multiplier:

$(\alpha_1, \alpha_2, \alpha_3) = (3.5, 0.75, 0.75)$

Linear SVM Problem

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$
$$b = y_i - \mathbf{w} \cdot \mathbf{x}_i$$

$$f(x) = \sum_i \alpha_i y_i (\mathbf{x}_i^\top \mathbf{x}) + b$$

VI Substitute the Lagrange multiplier and obtain the weight's:

$\alpha_i [-1 (\mathbf{w} \ . \ \mathbf{x}_i + b)] = -1$
$\alpha_i [+1 (\mathbf{w} \ . \ \mathbf{x}_i + b)] = 1$

$W = -\alpha_1 \ [0\ 1\ 1] + \alpha_2 \ [-1\ 3\ 1] + \alpha_3 [1\ 3\ 1]$
$= -3.5 \ [0\ 1\ 1] + 0.75 \ [-1\ 3\ 1] + 0.75\ [1\ 3\ 1]$
$= [0\ 1\ -2]$

VII Find the bias with help of any one of the support vectors:

Note the Bias is found above as a part of weight vector!!

VIII Construct the equation of the LSVM hyperplane:

$W X + b = 0$
$Y - 2 = 0$
$Y = 2$

IX Optionally find the width of the margin: $\frac{2}{||W||}$ H.W

Problem Type – 1 Linear SVM Solving for α

$$f(x) = \sum_i \alpha_i y_i (\mathbf{x}_i{}^\top \mathbf{x}) + b$$

- System of linear equations:

α1 y1 dot(x1, x1) + α2 y2 dot(x1, x2) + α3 y3 dot(x1, x3) = y1

α1 y1 dot(x2, x1) + α2 y2 dot(x2, x2) + α3 y3 dot(x2, x3) = y2

α1 y1 dot(x3, x1) + α2 y2 dot(x3, x2) + α3 y3 dot(x3, x3) = y3

-2 * α1 + 4 * α2 + 4 * α3 = -1

-4 * α1 + 11 * α2 + 9 * α3 = +1

-4 * α1 + 9 * α2 + 11 * α3 = +1

$\alpha_i [-1 (\mathbf{w} \,.\, \mathbf{x_i} + b)] = -1$

$\alpha_i [+1 (\mathbf{w} \,.\, \mathbf{x_i} + b)] = 1$

- Solution: α1 = 3.5, α2 = 0.75, α3 = 0.75

Solving for w, b; plotting boundary

- We know $\mathbf{w} = \Sigma\, \alpha_i\, y_i\, \mathbf{x}_i$ **i.e** $\mathbf{w} = \alpha_1\, y_1\, \mathbf{x}_1 + \ldots + \alpha_N\, y_N\, \mathbf{x}_N$

  where N = No of SVs

- Thus w = -3.5 * [0 1 1] + 0.75 [-1 3 1] + 0.75 [1 3 1] = [0 1 -2]
- Separating out weights and bias, we have: **w = [0 1] and b = -2**

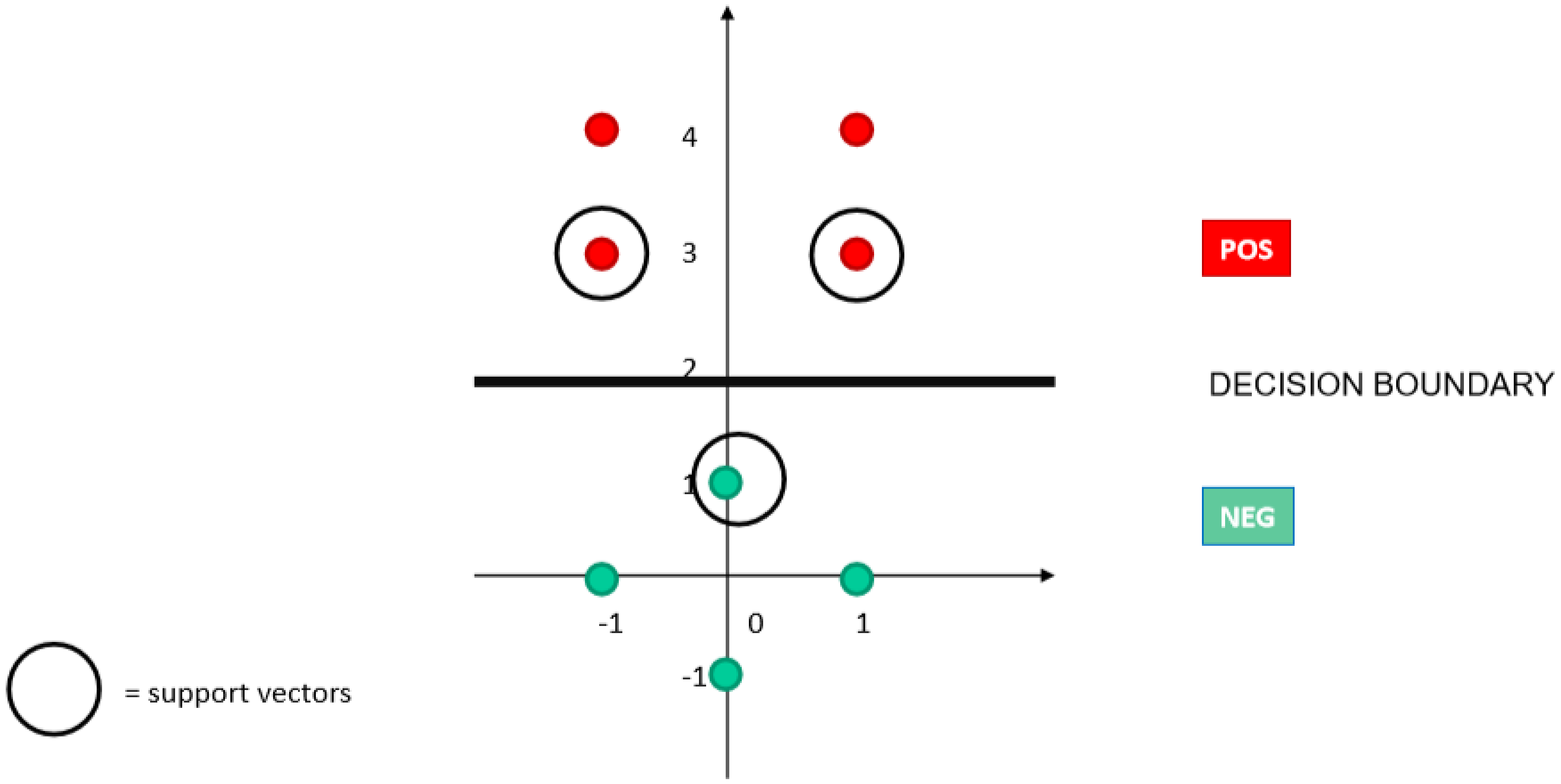
a=0, c=1

<u>Boundary:</u>

- For SVMs, we used this eq for a line: ax + cy + b = 0 where w = [a c]
- Thus ax + b = -cy ➔ y = (-a/c) x + (-b/c)
- Thus y-intercept is (-b/c) = -(-2)/1 = 2
- The decision boundary is perpendicular to w and it has slope

  =(-a/c) = -0/1 = 0

Problem Type – 1 Linear SVM Decision boundary



Linear SVM Problem

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$
$$b = y_i - \mathbf{w} \cdot \mathbf{x}_i$$

X Predict the class for unknown data:

$$f(x) = \sum_i \alpha_i y_i (\mathbf{x}_i{}^\top \mathbf{x}) + b$$

$\alpha_i\,[-1\,(\mathbf{w}\,.\,\mathbf{x}_i + b)] = -1$
$\alpha_i\,[+1\,(\mathbf{w}\,.\,\mathbf{x}_i + b)] = 1$

- Solution: $\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$

  $b = y_i - \mathbf{w} \cdot \mathbf{x}_i$ (for any support vector)
- Classification function:

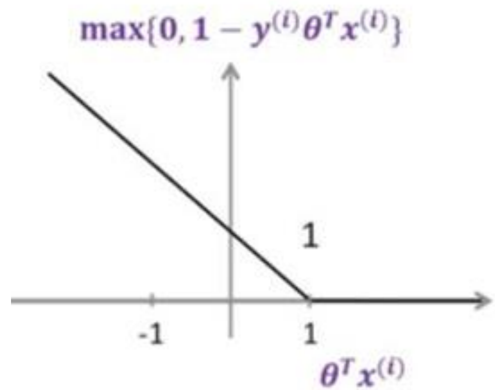
  $$f(x) = \mathrm{sign}\,(\mathbf{w} \cdot \mathbf{x} + \mathrm{b})$$
  $$= \mathrm{sign}\left(\sum_i \alpha_i y_i \boxed{\mathbf{x}_i \cdot \mathbf{x}} + b\right)$$

  If f(x) < 0, classify as negative, otherwise classify as positive.
- Notice that it relies on an *inner product* between the test point $\boldsymbol{x}$ and the support vectors $\boldsymbol{x}_i$
- (Solving the optimization problem also involves computing the inner products $\boldsymbol{x}_i \cdot \boldsymbol{x}_j$ between all pairs of training points)

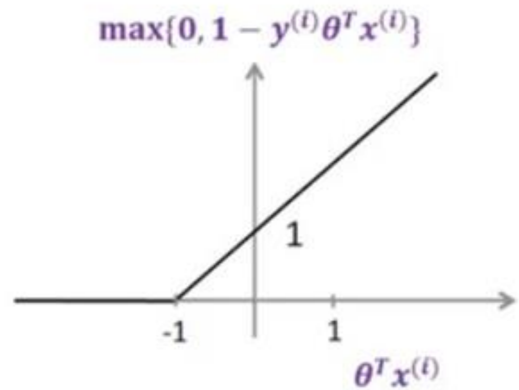
Slack variable-Hinge loss

$$\min_{\theta} \underbrace{\frac{1}{2}\|\hat{\theta}\|^2}_{\text{Margin}} + \underbrace{C}_{\substack{\text{Regularization}\\ \text{parameter}}} \sum_{i=1}^{n} \underbrace{\max\{0, 1 - y^{(i)}\,\theta^T x^{(i)}\}}_{\substack{\text{(hinge loss).}\\ \text{Cost/Loss of classifying one}\\ \text{data-point}}}$$

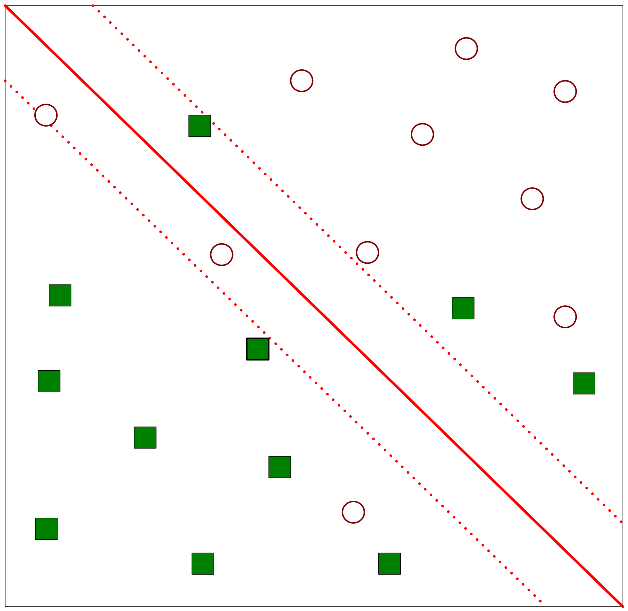
Case where $y^{(i)} = +1$



Case where $y^{(i)} = -1$



Soft Margin



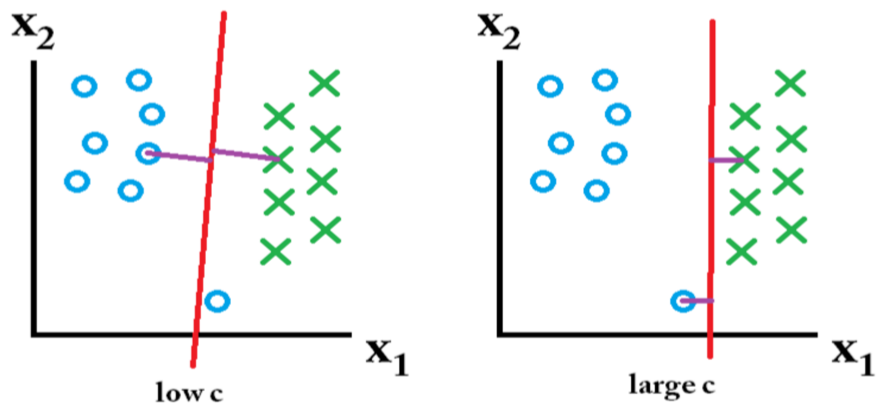
Noisy Training Set

Solution :

Slack variables ξ

Regularization C

Training set



Misclassification ok, want large margin

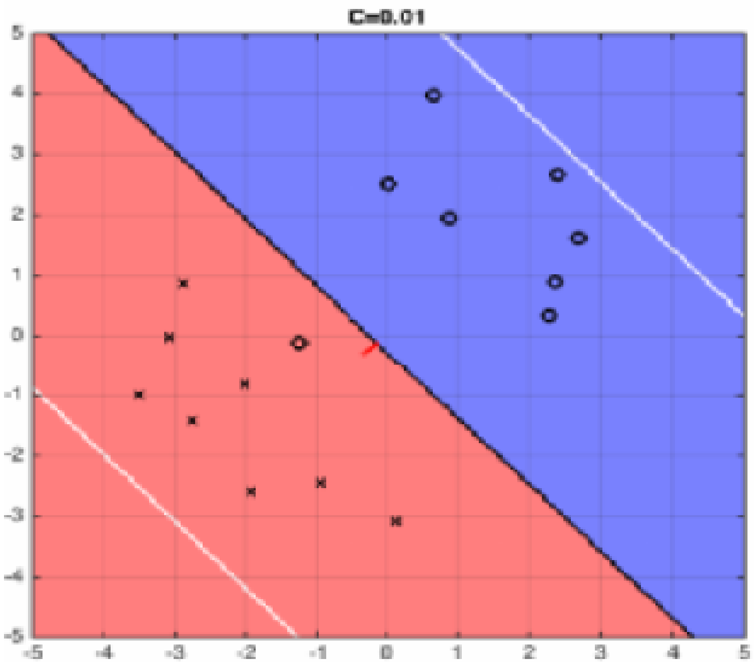
Misclassification not ok

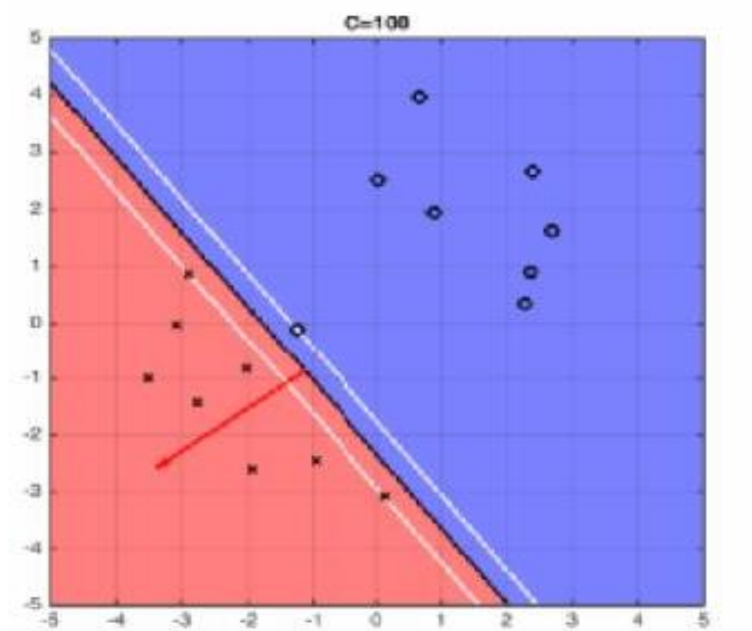
When C is large, larger slacks penalize the objective function of SVM's more than when C is small

For Large values of C, the optimization will choose a smaller-margin hyperplane

Effect of Margin size v/s misclassification cost

Effect of C





Small value of C will cause the optimizer to look for a larger-margin (small penalties) separating hyperplane, even if that hyperplane misclassifies more points.

For large values of C, the optimization will choose a smaller-margin (large penalties) hyperplane if that hyperplane does a better job of getting all the training points classified correctly.

C=infinity ->hard margin SVM

SVM Problem - Summary

SVM: Optimization

$$\min \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_i \xi_i$$

$$\text{Subject to:} \quad y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i$$

SVM: Training

Input: (X,y), C

Output: alpha for support vectors, b

Hyper parameter : C

SVM: Classification

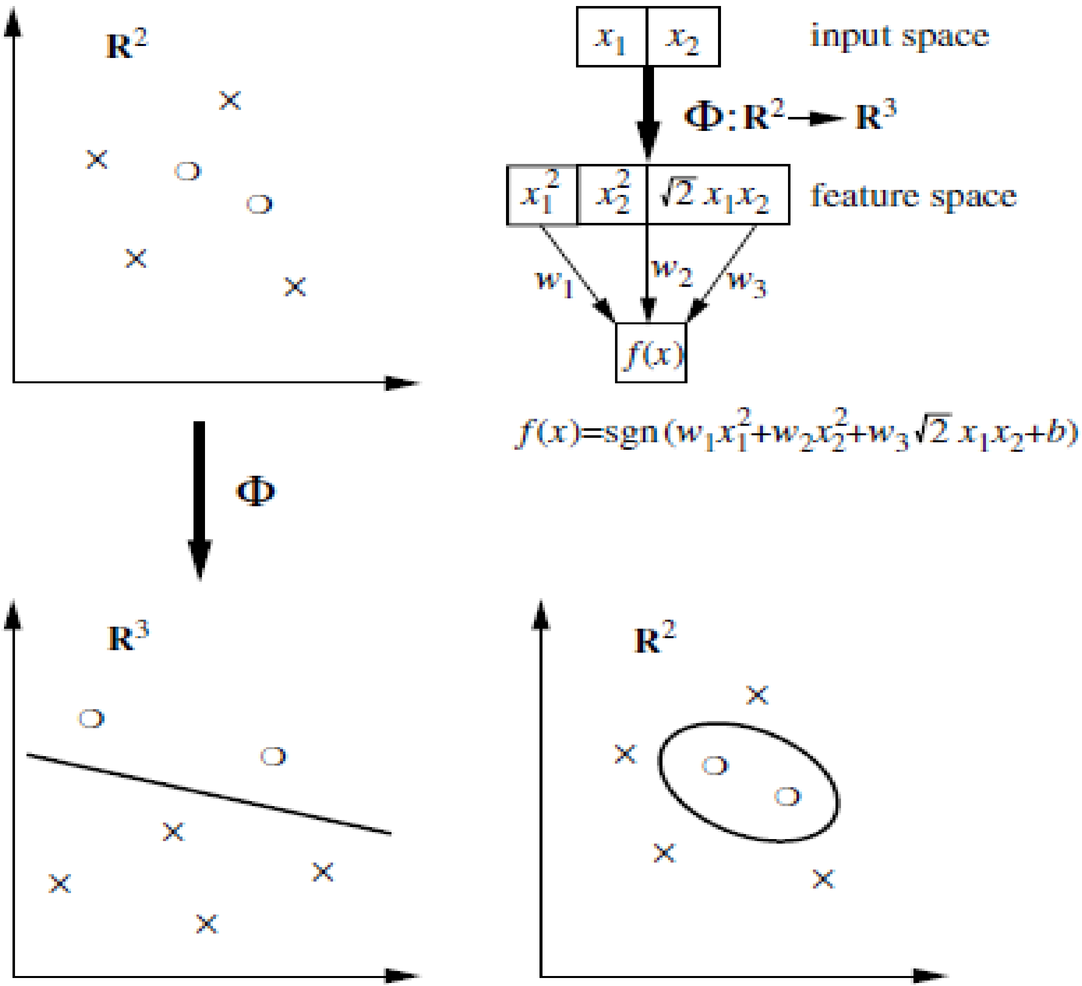
$$sign(\mathbf{w}^T\mathbf{x} + b)$$

$$= sign(\sum_i \alpha_i y_i \mathbf{x}_i^T\mathbf{x} + b)$$

C parameter tells the SVM optimization how much you want to avoid misclassifying each training example and *C* can be viewed as a way to control overfitting

Nonlinear SVM - kernels

Mapping into a New Feature Space



- Rather than run SVM on x_i, run it on $\Phi(x_i)$
- Find non-linear separator in input space
- What if $\Phi(x_i)$ is really big?
- Use kernels to compute it implicitly!

Kernel Trick (SVM)

- E.g. remember the hypothesis function of the original simplified SVM:

$$h_\theta(x) = \theta^T x = \theta_0 + \sum_{i=1}^{n} \alpha_i \, y^{(i)} \boldsymbol{x^T x^{(i)}}$$

  – It involves a dot product between the test data-point x and the support vectors ${x^{(i)}}^T$

- Instead of explicitly mapping the data to a higher dimensional space, we can just use a kernel function, and the hypothesis function would have the same form:

$$h_\theta(x) = \theta^T x = \theta_0 + \sum_{i=1}^{n} \alpha_i \, y^{(i)} \underbrace{\boldsymbol{k(x^T x^{(i)})}}_{z^T z^{(i)}}$$

Because since k is a kernel function, we know that $k(x, x^{(i)}) = \underbrace{\Phi(x)^T}_{z^T} \underbrace{\Phi(x^{(i)})}_{z^{(i)}}$

So we can use the dot product between the higher dimensional vectors, without explicitly knowing them (i.e. a trick).

Problem Type - 2

Example:

Let $x^{(i)} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \end{bmatrix}$, $x^{(j)} = \begin{bmatrix} x_1^{(j)} \\ x_2^{(j)} \end{bmatrix}$, $k(x^{(i)}, x^{(j)}) = \left(1 + x^{(i)^T} x^{(j)}\right)^2$

- Is this a kernel function ?
- We need to show that $k(x^{(i)}, x^{(j)}) = \Phi(x^{(i)})^T \Phi(x^{(j)})$

Problem Type - 2

Example:

Let $x^{(i)} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \end{bmatrix}$, $x^{(j)} = \begin{bmatrix} x_1^{(j)} \\ x_2^{(j)} \end{bmatrix}$, $k(x^{(i)}, x^{(j)}) = \left(1 + x^{(i)^T} x^{(j)}\right)^2$

$$k(x^{(i)}, x^{(j)}) = 1 + x_1^{(i)^2} x_1^{(j)^2} + 2x_1^{(i)} x_1^{(j)} x_2^{(i)} x_2^{(j)} + x_2^{(i)^2} x_2^{(j)^2} + 2x_1^{(i)} x_1^{(j)} + 2x_2^{(i)} x_2^{(j)}$$

$$= \underbrace{\begin{bmatrix} 1 & x_1^{(i)^2} & \sqrt{2}x_1^{(i)} x_2^{(i)} & x_2^{(i)^2} & \sqrt{2}x_1^{(i)} & \sqrt{2}x_2^{(i)} \end{bmatrix}}_{\Phi\left(x^{(i)}\right)^T} \underbrace{\begin{bmatrix} 1 \\ x_1^{(j)^2} \\ \sqrt{2}x_1^{(j)} x_2^{(j)} \\ x_2^{(j)^2} \\ \sqrt{2}x_1^{(j)} \\ \sqrt{2}x_2^{(j)} \end{bmatrix}}_{\Phi(x^{(j)})}$$

So, yes, this is a kernel function.

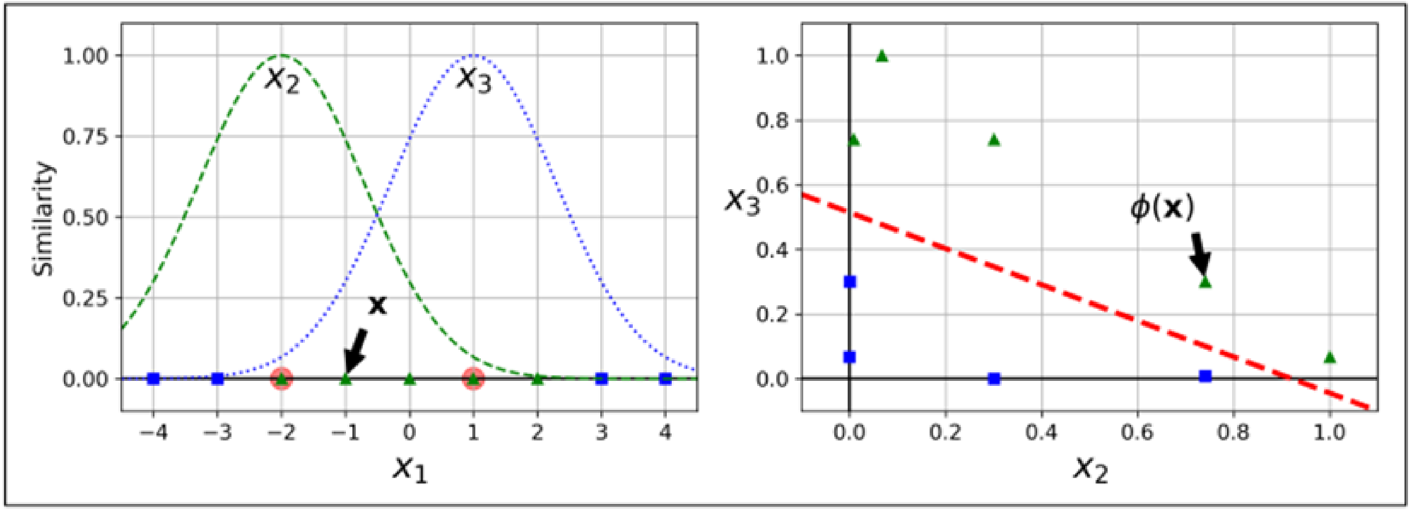
```python
from sklearn.datasets import make_moons
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import PolynomialFeatures

polynomial_svm_clf = Pipeline([
        ("poly_features", PolynomialFeatures(degree=3)),
        ("scaler", StandardScaler()),
        ("svm_clf", LinearSVC(C=10, loss="hinge"))
    ])

polynomial_svm_clf.fit(X, y)
```

The hyper parameter coef0 controls how much the model is influenced by the polynomial

```python
from sklearn.svm import SVC
poly_kernel_svm_clf = Pipeline([
        ("scaler", StandardScaler()),
        ("svm_clf", SVC(kernel="poly", degree=3, coef0=1, C=5))
    ])
poly_kernel_svm_clf.fit(X, y)
```

Non-Linear SVM- Idea

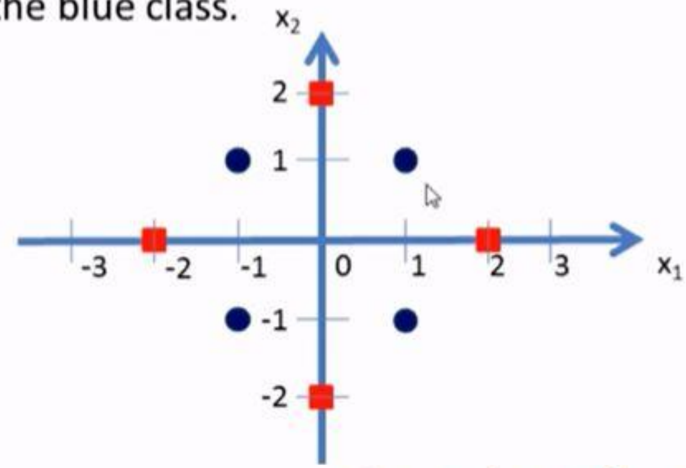
```
rbf_kernel_svm_clf = Pipeline([
        ("scaler", StandardScaler()),
        ("svm_clf", SVC(kernel="rbf", gamma=5, C=0.001))
    ])
rbf_kernel_svm_clf.fit(X, y)
```



Increasing gamma makes the bell-shape curve narrower, and as a result each instance's range of influence is smaller: the decision boundary ends up being more irregular

Problem Type - 3

- Obviously there is no clear separating hyperplane between the red class and the blue class.



- Blue class vectors are: $\begin{pmatrix}1\\1\end{pmatrix}, \begin{pmatrix}-1\\1\end{pmatrix}, \begin{pmatrix}-1\\-1\end{pmatrix}, \begin{pmatrix}1\\-1\end{pmatrix}$

- Red class vectors are: $\begin{pmatrix}2\\0\end{pmatrix}, \begin{pmatrix}0\\2\end{pmatrix}, \begin{pmatrix}-2\\0\end{pmatrix}, \begin{pmatrix}0\\-2\end{pmatrix}$

Non-linear SVM using kernel steps

1. Select a kernel function.
2. Compute pairwise kernel values between labeled examples.
3. Use this “kernel matrix” to solve for SVM support vectors & alpha weights.
4. To classify a new example: compute kernel values between new input and support vectors, apply alpha weights, check sign of output.

Problem Type - 3

- Here we need to find a non-linear mapping function Φ which can transform these data in to a new feature space where a seperating hyperplane can be found.
- Let us consider the following mapping function.
- $$\Phi\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{cases} \begin{pmatrix} 6 - x_1 + (x_1 - x_2)^2 \\ 6 - x_2 + (x_1 - x_2)^2 \end{pmatrix} & \text{if } \sqrt{x_1^2 + x_2^2} \geq 2 \\ \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} & \text{otherwise} \end{cases}$$

Problem Type - 3

- Now let us transform the blue and red calss vectors using the non-linear mapping function Φ .

- $$\Phi\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{cases} \begin{pmatrix} 6 - x_1 + (x_1 - x_2)^2 \\ 6 - x_2 + (x_1 - x_2)^2 \end{pmatrix} & \text{if } \sqrt{x_1^2 + x_2^2} \geq 2 \\ \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} & \text{otherwise} \end{cases}$$

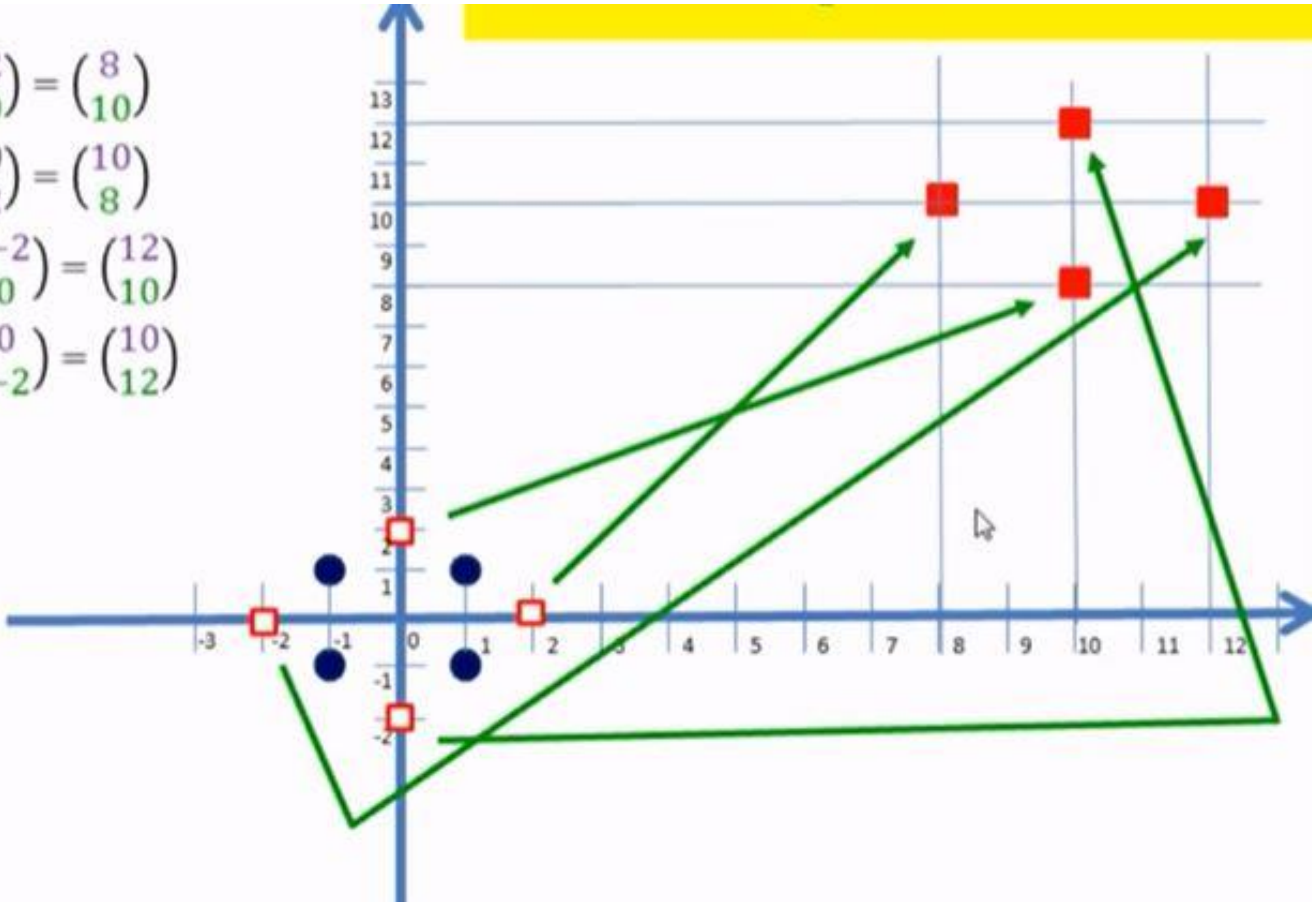
- Blue class vectors are: $\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ no change since $\sqrt{x_1^2 + x_2^2} < 2$ for all the vectors

Problem Type - 3

- $$\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \begin{cases}\begin{pmatrix}6 - x_1 + (x_1 - x_2)^2\\6 - x_2 + (x_1 - x_2)^2\end{pmatrix} & \text{if } \sqrt{x_1^2 + x_2^2} \geq 2\\ \begin{pmatrix}x_1\\x_2\end{pmatrix} & \text{otherwise}\end{cases}$$
- Let us take Red class vectors : $\begin{pmatrix}2\\0\end{pmatrix}, \begin{pmatrix}0\\2\end{pmatrix}, \begin{pmatrix}-2\\0\end{pmatrix}, \begin{pmatrix}0\\-2\end{pmatrix}$
- $\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \Phi\begin{pmatrix}2\\0\end{pmatrix} = \begin{pmatrix}6 - 2 + (2 - 0)^2\\6 - 0 + (2 - 0)^2\end{pmatrix} = \begin{pmatrix}8\\10\end{pmatrix}$
- $\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \Phi\begin{pmatrix}0\\2\end{pmatrix} = \begin{pmatrix}6 - 0 + (0 - 2)^2\\6 - 2 + (0 - 2)^2\end{pmatrix} = \begin{pmatrix}10\\8\end{pmatrix}$
- $\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \Phi\begin{pmatrix}-2\\0\end{pmatrix} = \begin{pmatrix}6 + 2 + (-2 - 0)^2\\6 - 0 + (-2 - 0)^2\end{pmatrix} = \begin{pmatrix}12\\10\end{pmatrix}$
- $\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \Phi\begin{pmatrix}0\\-2\end{pmatrix} = \begin{pmatrix}6 - 0 + (0 + 2)^2\\6 + 2 + (0 + 2)^2\end{pmatrix} = \begin{pmatrix}10\\12\end{pmatrix}$

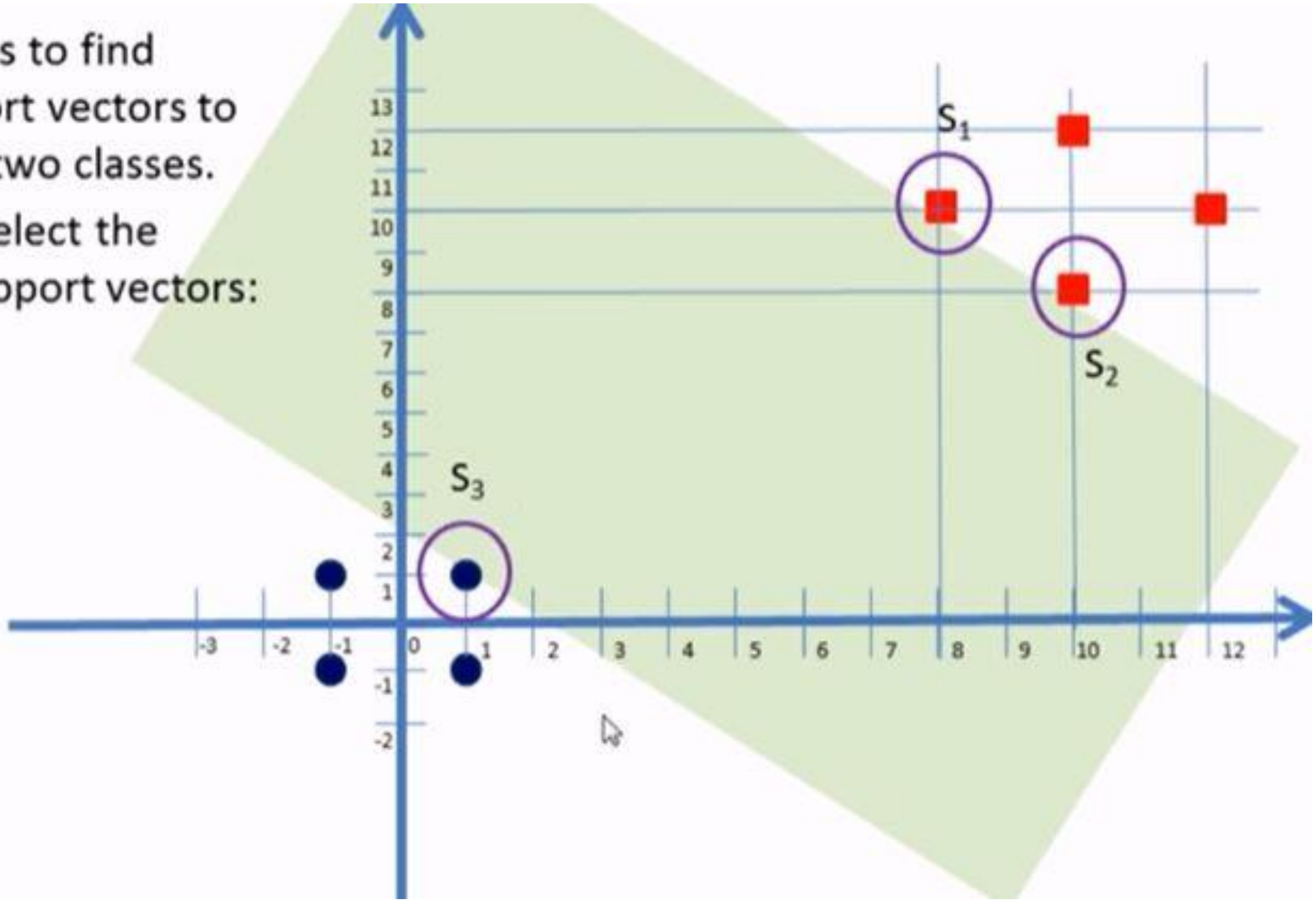
Problem Type - 3

- $\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \Phi\begin{pmatrix}2\\0\end{pmatrix} = \begin{pmatrix}8\\10\end{pmatrix}$
- $\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \Phi\begin{pmatrix}0\\2\end{pmatrix} = \begin{pmatrix}10\\8\end{pmatrix}$
- $\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \Phi\begin{pmatrix}-2\\0\end{pmatrix} = \begin{pmatrix}12\\10\end{pmatrix}$
- $\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \Phi\begin{pmatrix}0\\-2\end{pmatrix} = \begin{pmatrix}10\\12\end{pmatrix}$



Problem Type - 3

- Now our task is to find suitable support vectors to classify these two classes.
- Here we will select the following 3 support vectors:
- $S_1 = \begin{pmatrix} 8 \\ 10 \end{pmatrix}$,
- $S_2 = \begin{pmatrix} 10 \\ 8 \end{pmatrix}$,
- and $S_3 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$



Problem Type - 3

- Here we will use vectors augmented with a 1 as a bias input, and for clarity we will differentiate these with an over-tilde. That is:

$$S_1 = \begin{pmatrix} 8 \\ 10 \end{pmatrix} \qquad \widetilde{S_1} = \begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix}$$

$$S_2 = \begin{pmatrix} 10 \\ 8 \end{pmatrix} \qquad \widetilde{S_2} = \begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix}$$

$$S_3 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \qquad \widetilde{S_3} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

Problem Type - 3

$$f(x) = \sum_i \alpha_i y_i (\mathbf{x}_i{}^\top \mathbf{x}) + b$$

- Now we need to find 3 parameters α_1, α_2, and α_3 based on the following 3 linear equations:

$$\alpha_1 \widetilde{S_1}.\widetilde{S_1} + \alpha_2 \widetilde{S_2}.\widetilde{S_1} \overset{-}{\cancel{+}} \alpha_3 \widetilde{S_3}.\widetilde{S_1} = +1 \ (+ve\ class)$$

$$\alpha_1 \widetilde{S_1}.\widetilde{S_2} + \alpha_2 \widetilde{S_2}.\widetilde{S_2} \overset{-}{\cancel{+}} \alpha_3 \widetilde{S_3}.\widetilde{S_2} = +1 \ (+ve\ class)$$

$$\alpha_1 \widetilde{S_1}.\widetilde{S_3} + \alpha_2 \widetilde{S_2}.\widetilde{S_3} \overset{-}{\cancel{+}} \alpha_3 \widetilde{S_3}.\widetilde{S_3} = -1 \ (-ve\ class)$$

- Let's substitute the values for S_1, S_2 and S_3 in the above equations.

$$\widetilde{S_1} = \begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix} \quad \widetilde{S_2} = \begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix} \quad \widetilde{S_3} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\alpha_1 \begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix}.\begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix}.\begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix} \overset{-}{\cancel{+}} \alpha_3 \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.\begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix} = +1$$

$$\alpha_1 \begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix}.\begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix}.\begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix} \overset{-}{\cancel{+}} \alpha_3 \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.\begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix} = +1$$

$$\alpha_1 \begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix}.\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix}.\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \overset{-}{\cancel{+}} \alpha_3 \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = -1$$

Problem Type - 3

- After multiplication we get:

$$165\,\alpha_1 + 161\,\alpha_2 + 19\,\alpha_3 = +1$$

$$161\,\alpha_1 + 165\,\alpha_2 + 19\,\alpha_3 = +1$$

$$19\,\alpha_1 + 19\,\alpha_2 + 3\,\alpha_3 = -1$$

- Simplifying the above 3 simultaneous equations we get: α_1=α_2= 0.859 and α_3=-1.4219.

Problem Type - 3

- The hyper plane that discriminates the positive class from the negative class is given by:

$$\mathbf{w} = \Sigma\, \alpha_i y_i \mathbf{x_i}$$

- Substituting the values we get:

$$\tilde{w} = \alpha_1 \begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix} - \alpha_3 \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\tilde{w} = (0.0859).\begin{pmatrix} 8 \\ 10 \\ 1 \end{pmatrix} + (0.0859).\begin{pmatrix} 10 \\ 8 \\ 1 \end{pmatrix} - (+1.4219).\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0.1243 \\ 0.1243 \\ -1.2501 \end{pmatrix}$$

Problem Type - 3

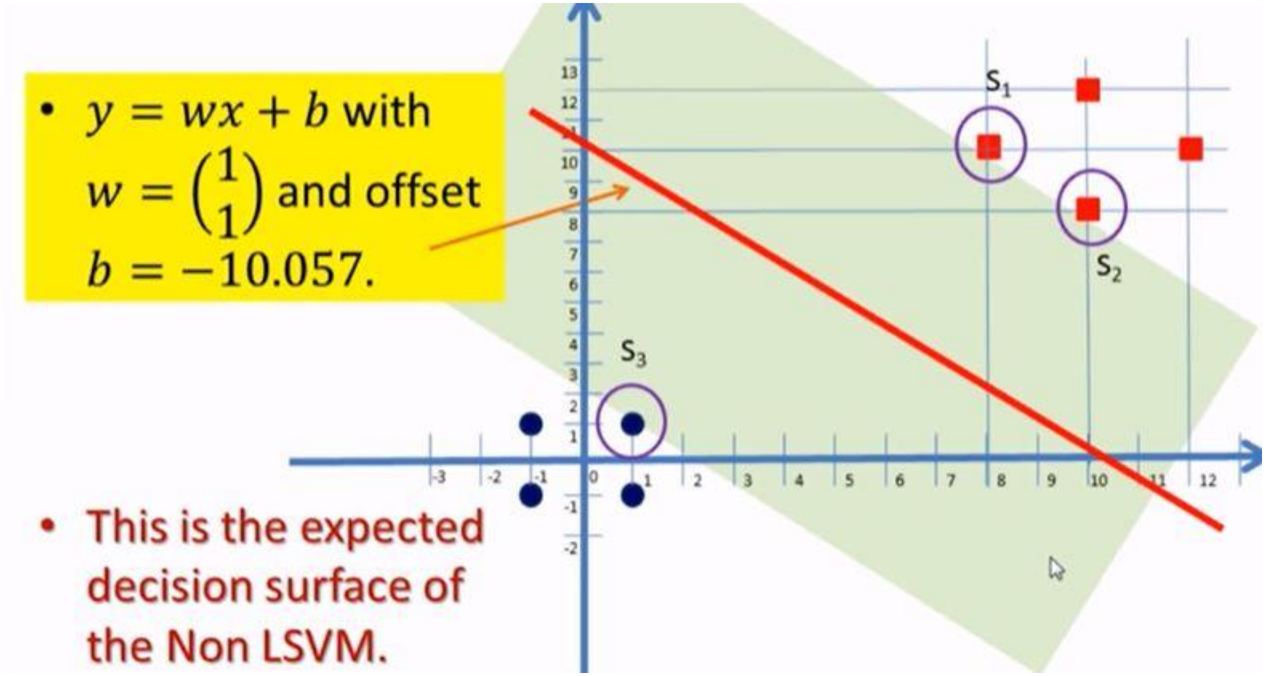
For SVMs, we used this eq for a line: ax + cy + b = 0 where w = [a c]
Thus ax + b = -cy ➔ y = (-a/c) x + (-b/c)
Thus y-intercept is (-b/c)
The decision boundary is perpendicular to w and it has slope =(-a/c)

- Our vectors are augmented with a bias.
- Hence we can equate the entry in $\widetilde{w}$ as the hyper plane with an offset b.
- Therefore the separating hyper plane equation

$y = wx + b$ with $w = \begin{pmatrix} 0.1243/0.1243 \\ 0.1243/0.1243 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

and an offset $b = -\frac{1.2501}{0.1243} = -10.057.$ ← **Y intercept**

Problem Type - 3



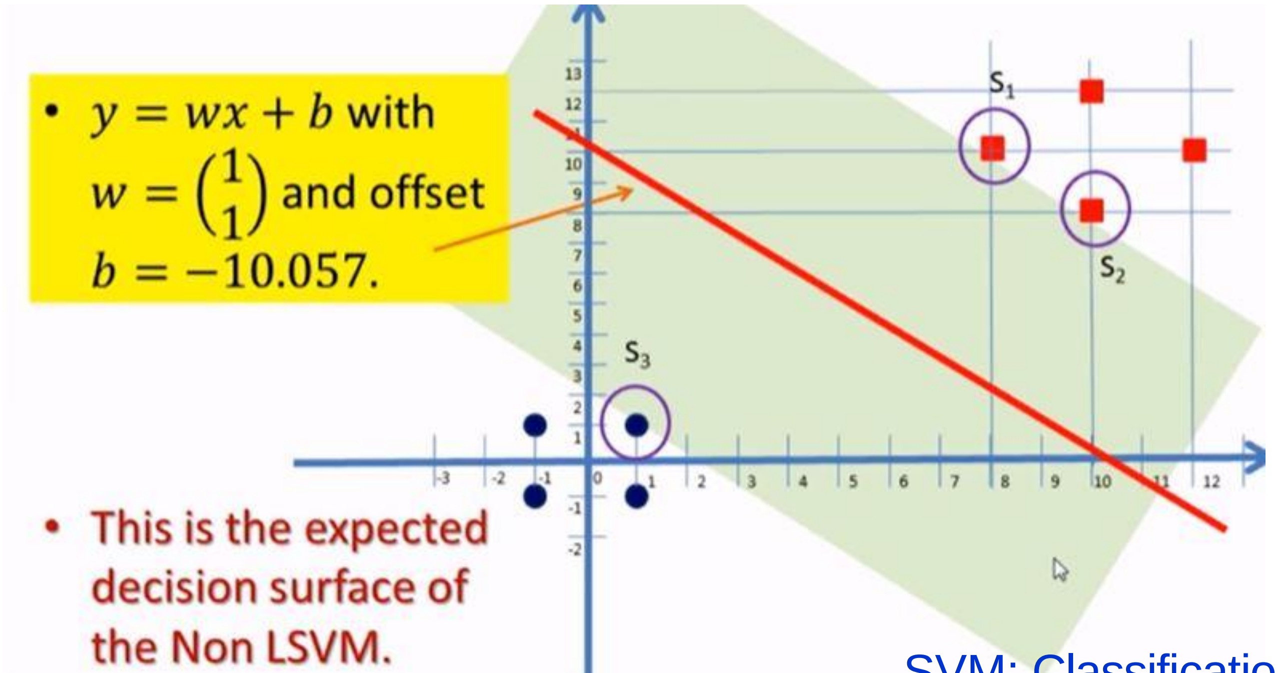
Problem Type - 3

New $X_t = [0 \;\; 1.5]^T$

$\Phi(X_t) = [4.5 \;\; 4.5]$

Prediction: y=-1
sign(-1.057)

$$\Phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \begin{cases}\begin{pmatrix}6 - x_1 + (x_1 - x_2)^2\\6 - x_2 + (x_1 - x_2)^2\end{pmatrix} & \text{if } \sqrt{x_1^2 + x_2^2} \geq 2\\ \begin{pmatrix}x_1\\x_2\end{pmatrix} & \text{otherwise}\end{cases}$$

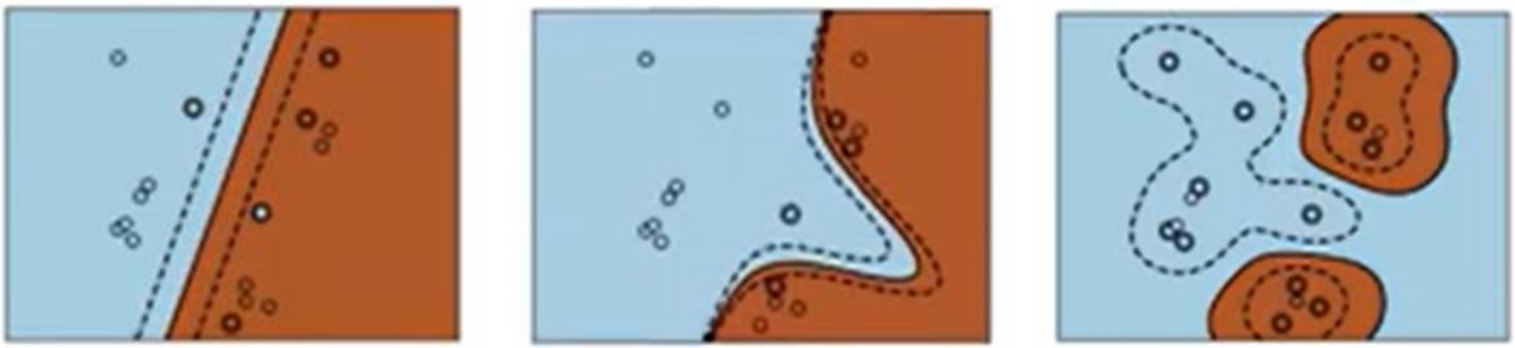


SVM: Classification

$$sign(\mathbf{w}^T\mathbf{x} + b)$$
$$= sign(\sum_i \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b)$$

$$f(\mathbf{z}) = sign(\mathbf{w} \cdot \Phi(\mathbf{z}) + b) = sign\left(\sum_{i=1}^{n} \lambda_i y_i \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{z}) + b\right)$$

SVM Kernels

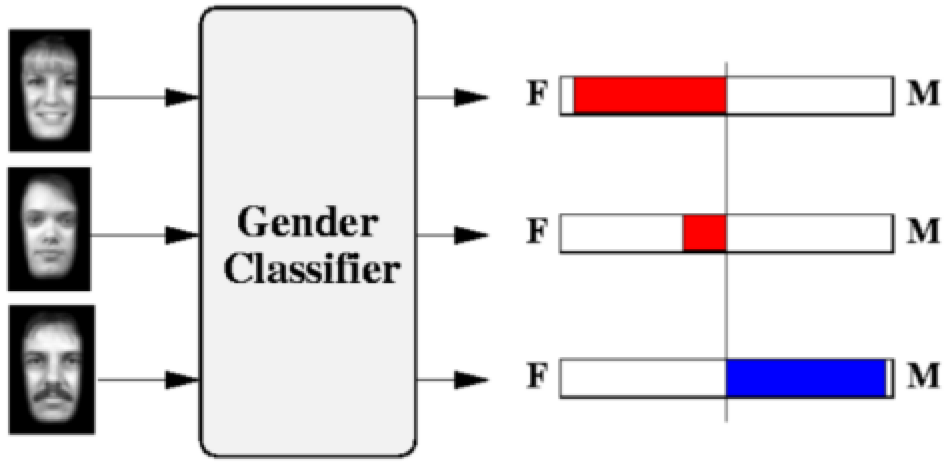


Name of Kernel Function	Definition
Linear	$K(\mathbf{u},\mathbf{v}) = \mathbf{u}^T\mathbf{v}$
Polynomial of degree d	$K(\mathbf{u},\mathbf{v}) = (\mathbf{u}^T\mathbf{v}+1)^d$
Gaussian Radial Basis Function (RBF)	$K(\mathbf{u},\mathbf{v}) = e^{-\frac{1}{2}[(\mathbf{u}-\mathbf{v})^T\Sigma^{-1}(\mathbf{u}-\mathbf{v})]}$
Sigmoid	$K(\mathbf{u},\mathbf{v}) = \tanh[\mathbf{u}^T\mathbf{v}+b]$

Linear Kernal	Large Data	Text
Polynomial Kernal	Normalized Data	Image Processing
Gaussian Kernal	EDA not clear	Computing Power

SVM Application – Observations

Learning Gender from Images



Moghaddam and Yang, Learning Gender with Support Faces, TPAMI 2002

Moghaddam and Yang, Face & Gesture 2000

SVM Application – Observations

Image Analysis

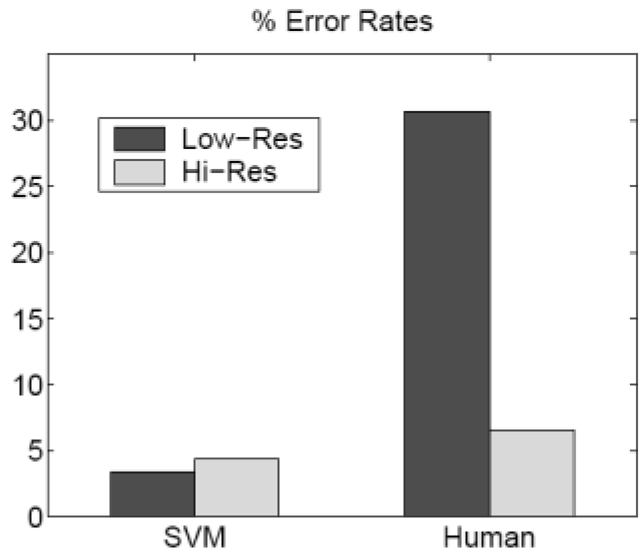


Figure 6. SVM vs. Human performance

- SVMs performed better than humans, at either resolution

Properties of SVM

- Flexibility in choosing a similarity function
- Sparseness of solution when dealing with large data sets
  - Only support vectors are used to specify the separating hyperplane
  - Therefore SVM also called sparse kernel machine.
- Ability to handle large feature spaces
  - complexity does not depend on the dimensionality of the feature space
- Overfitting can be controlled by soft margin approach
- Nice math property: a simple convex optimization problem which is guaranteed to converge to a single global solution