Classifiers & Classification

Forsyth & Ponce
“Computer Vision A Modern Approach”
chapter 22

Pattern Classification – Duda, Hart and Stork

graphics • vision • visualisation

School of Computer Science & Statistics
Trinity College Dublin
Dublin 2
Ireland
www.scss.tcd.ie

Lecture Overview

- An introduction to Classifiers
- Parametric and Non-parametric approaches
- Building Classifiers from Class Histograms
- Evaluation of Classifiers
- Support Vector Machines

Basic Framework

- Object defined by a set of features
  - Use a classifier to classify the set of extracted features i.e. the "feature vector"
- Training Set
  - Set of labeled examples
    - ASIDE: Supervised learning as opposed to clustering or unsupervised learning where there are no labels
  - Classifier builds up rules to label new examples
  - Training data-set (x_i, y_i) where x_i - feature measurements are mapped onto y_i labels

Loss Function - how costly is a mistake?

- Consider doctors diagnosing a patient
  - Cost to patient of a False Positive (FP)?
  - Cost to patient of a False Negative (FN)?
- The Loss Function

$$L(i \rightarrow i) = 0 \qquad L(i \rightarrow j) = loss$$

- The Risk Function

$$R(s) = Pr\{1 \rightarrow 2 \mid using\ s\} L(1 \rightarrow 2) + Pr\{2 \rightarrow 1 \mid using\ s\} L(2 \rightarrow 1)$$

  - We want to minimise total risk

2 Class Classifier that minimises total risk

- Choose between two classes
  - e.g. face & non-face, tumour & non-tumour
  - Boundary in feature space - *decision boundary*
  - Points on the decision boundary of optimal classifier both classes have the same expected loss

$$p(\mathbf{x} \mid 1)p(1)L(1 \to 2) = p(\mathbf{x} \mid 2)p(2)L(2 \to 1)$$

  - All other points choose the lowest expected loss
    - Class one if $p(1 \mid \mathbf{x})L(1 \to 2) > p(2 \mid \mathbf{x})L(2 \to 1)$
    - Class two if $p(1 \mid \mathbf{x})L(1 \to 2) < p(2 \mid \mathbf{x})L(2 \to 1)$

Multiple Classes

- let us assume $L(i \rightarrow j) = 0$ for $i=j$ and 1 otherwise
  - In some case you can make no decision (d) but this option also has some loss thus: $d<1$

$$L(i \rightarrow j) = \begin{cases} 1 & i \neq j \\ 0 & i = j \\ d < 1 & no\,decision \end{cases}$$

  - Choose class k if $P(k|x) > P(i|x)$ for all i, and $P(k|x) > 1-d$
  - If there are several classes where $P(k_p|x)=P(k_q|x)=\ldots$ choose randomly between the classes k
  - If $P(k|x) < 1-d$ don't make a decision

Methods for Building Classifiers

- At the outset we don't know $P(x|k)$ or $P(k)$ and we must determine these from a data-set

- Two main strategies:
  - Explicit Probability models
    - Parametric classifiers
  - Determine the Decision boundaries directly
    - Non-parametric classifiers

Explicit Probability Models

- Assume the distribution of the feature vectors has a well defined functional form, e.g., Gaussian distribution.
- From a training set where we have N classes
  - The *k*'th class has N_k examples in which the *i*'th feature vector is $x_{k,i}$
  - Estimate the mean μ and covariance Σ for each class k

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N_k} x_{k,i} \qquad \Sigma_k = \frac{1}{N_k - 1} \sum_{i=1}^{N_k} (x_{k,i} - \mu_k)(x_{k,i} - \mu_k)^T$$

Parameter Estimates

Estimates themselves are Random Vectors/variables

Judging how good your estimates are:

Let τ be an estimate of a parameter T.

Bias: $E(\tau) - T$, Variance: $V(\tau)$. E is the expectation.

Aim at "minimum variance unbiased estimates".

Let us consider the estimate of population mean:

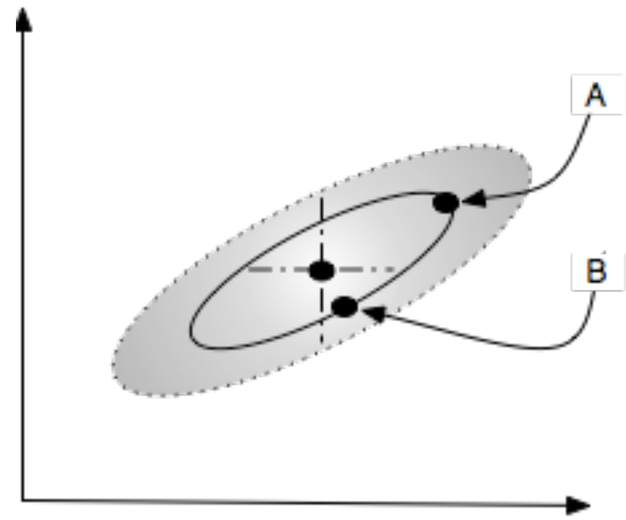
$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N_k} x_{k,i}$$

$\mathrm{Bias}(\mu_k) = 0$, $V(\mu_k) = \sigma^2/N_k$, Larger the sample size better the Estimate !!!!.

The Mahalanobis distance

- For data point x, choose the closest class, *taking the variance into account*
  - The shortest mahalanobis distance

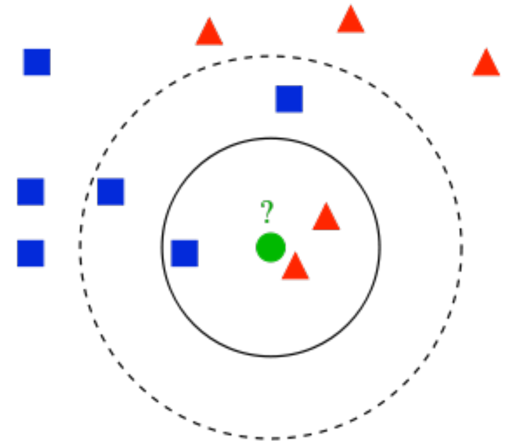
$$\delta(x;\mu_k,\Sigma_k) = \sqrt{(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)}$$



- Choose class K which has the smallest value of

$$\delta(x;\mu_k,\Sigma_k)^2 - P(k) + \frac{1}{2}log \mid \Sigma_k \mid$$

A non-parametric classifier- K-nearest neighbour



- Classify unknown point by using the nearest neighbours
- A (k, ℓ) nearest neighbour classifier - given a feature vector x
  - Class with most votes in k nearest examples
  - But if less than ℓ votes don't classify
  - What are the nearest neighbours? - search?
  - What should be the distance metric?
    - Feature Vector: length, colour, angle - mahalanobis?

Performance: estimation and improvement

- Can the classifier generalise beyond its training set? - *training set* vs. *test set*
- Overfitting / Selection Bias
  - Good on training set, but poor generalisation
  - Learned the quirks of training set, training set not fully representative?
- Performance estimation
  - Hold back some data for test set
  - Theoretical measures of performance

Cross Validation

- Labelled data sets are difficult to get
- Leave one out cross validation
  - Leave one example out and test the classification error on that one
  - Iterate through the data set
  - Compute the average classification error
- K-fold cross validation
  - Split the data set in to K sub-sets, leave one out
  - 10 fold cross validation common

Bootstrapping

- Not all examples are equally useful
  - Examples close to the decision boundary are key
- Very large training sets
  - Not efficient to use all points (e.g. KNN)
- Bootstrapping
  - Train on subset of data
  - Test on remainder
  - Put FP and FN into the training set and retrain
    - The FP and FN tell us most about the position of the decision boundary

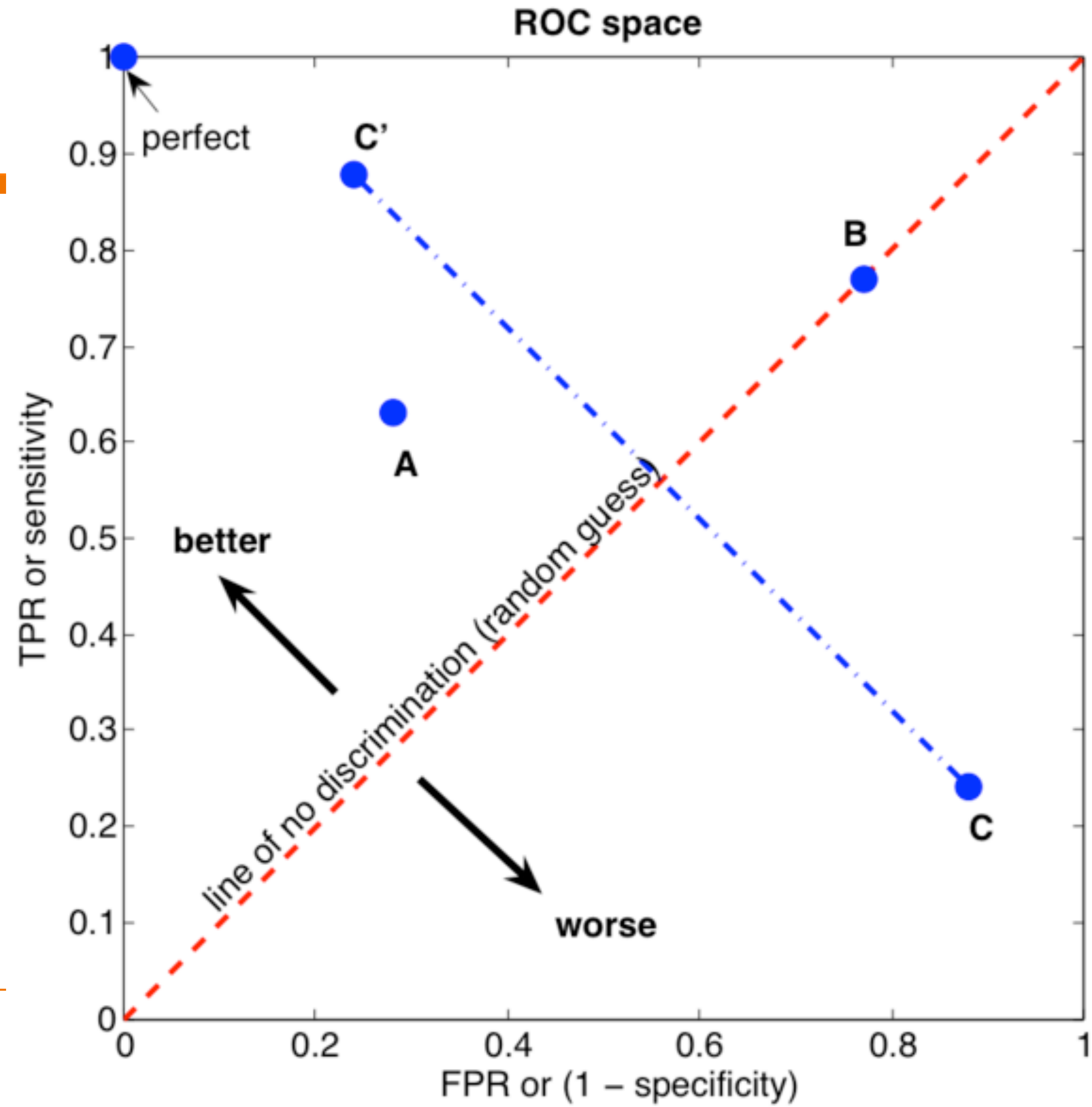
Building Classifiers from Histograms

- Use histograms of values to estimate PDFs
- Skin detection- Jones and Rehg
  - RGB histogram of skin pixels
  - RGB histogram of non-skin pixels
- Feature vector, x, the RBG values at a pixel
  - Histograms provide $P(x|skin)$ and $P(x|non\text{-}skin)$
  - If $P(skin|x)>\theta$ classify as skin
  - If $P(skin|x)<\theta$ classify as non-skin
  - If $P(skin|x)=\theta$ classify randomly
  - θ's encapsulate relative loss functions

Comparing Classifiers: The ROC curve

- Comparing performance of different classifiers
  - E.g. What is the right θ?
- Receiver Operator Characteristic(ROC) curve
  - Plot “True Positive Rate” vs. “False Positive Rate”
  - TPR = TP / (TP+FN)
    - Also called hit rate, recall, sensitivity
  - FPR = FP/(FP+TN)
    - Also called false alarm rate, fall-out, =1-specificity

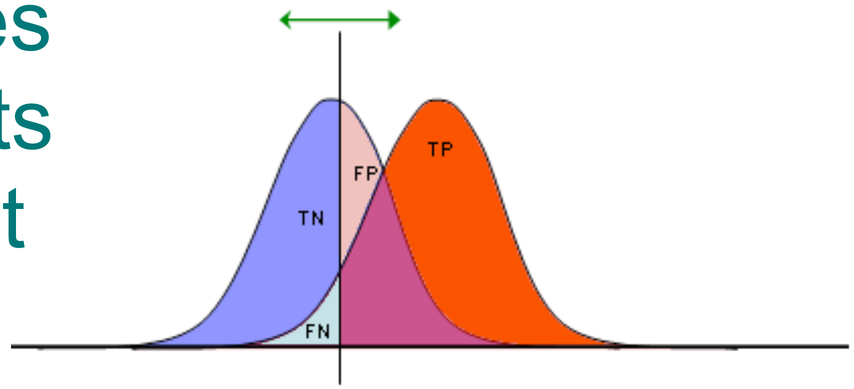
ROC Space



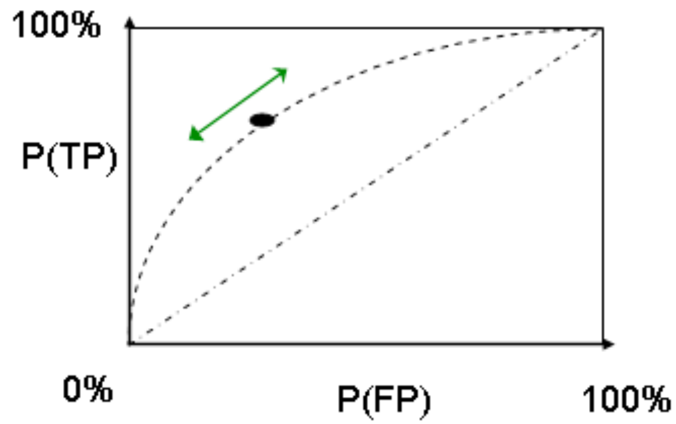
ROC space
perfect
C'
B
A
C
better
worse
line of no discrimination (random guess)
TPR or sensitivity
FPR or (1 – specificity)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1

ROC Curve

- Different values of θ yield points on a curve that can be plotted
- Compare classifiers using Area Under Curve (AUC)



TP	FP
FN	TN
1	1



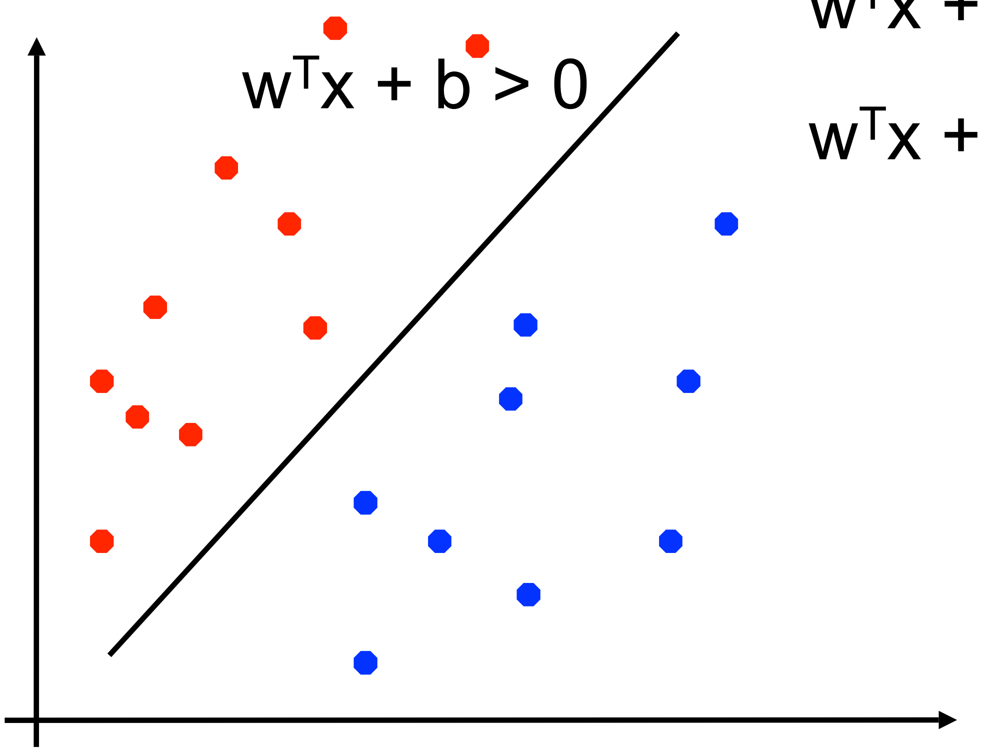
Support Vector Machines (SVM)

- Very popular classifier in vision for training on the basis of example data

- Consider a binary classification problem (-1,1)
  - Dataset with N data points of data x and class label y.
  - We want to predict the y_i for each x_i
  - Assume that the data are Linearly separable
    - “Linear SVM”

Linear Separators

- Binary classification can be viewed as the task of separating classes in feature space:



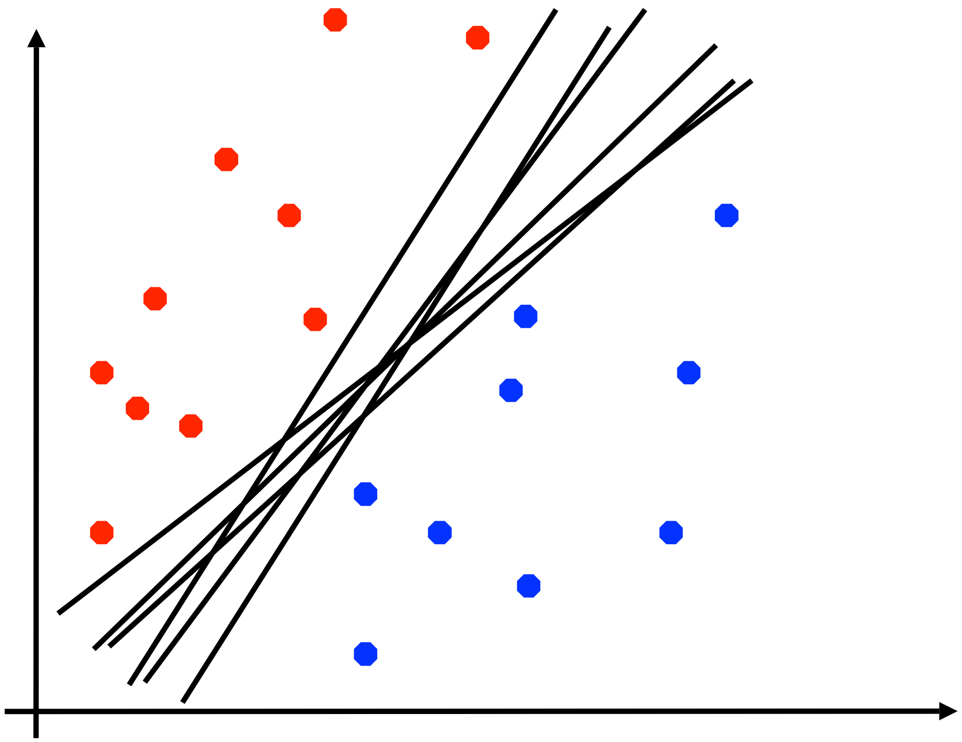
$w^Tx + b = 0$

$w^Tx + b < 0$

$$f(x) = \text{sign}(w^Tx + b)$$

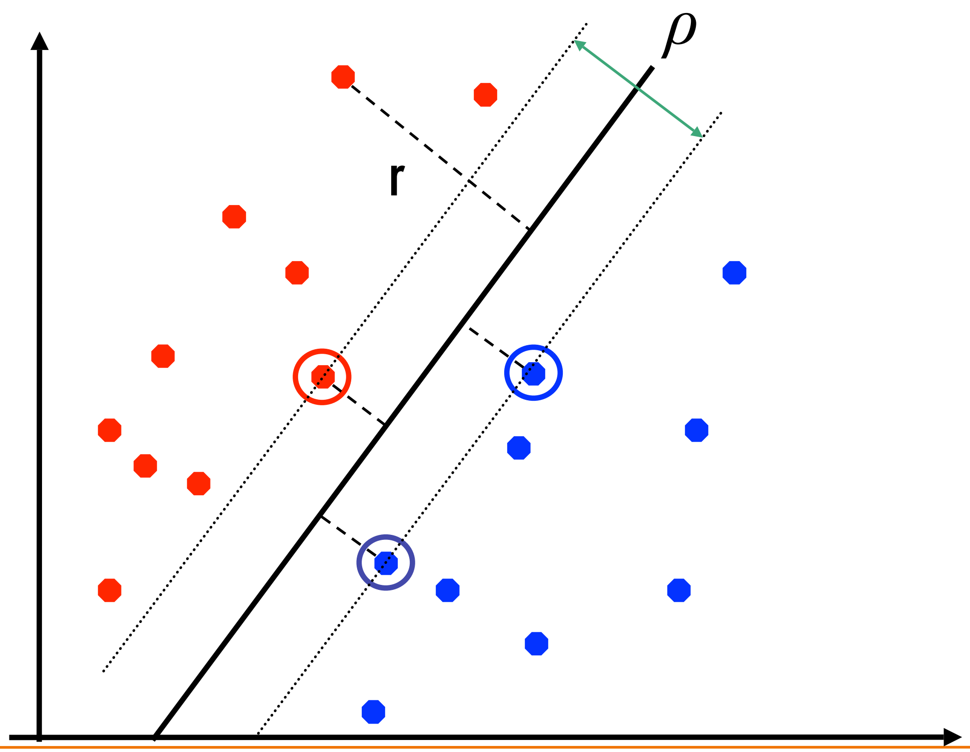
Linear Separators

- Which of the linear separators is optimal?



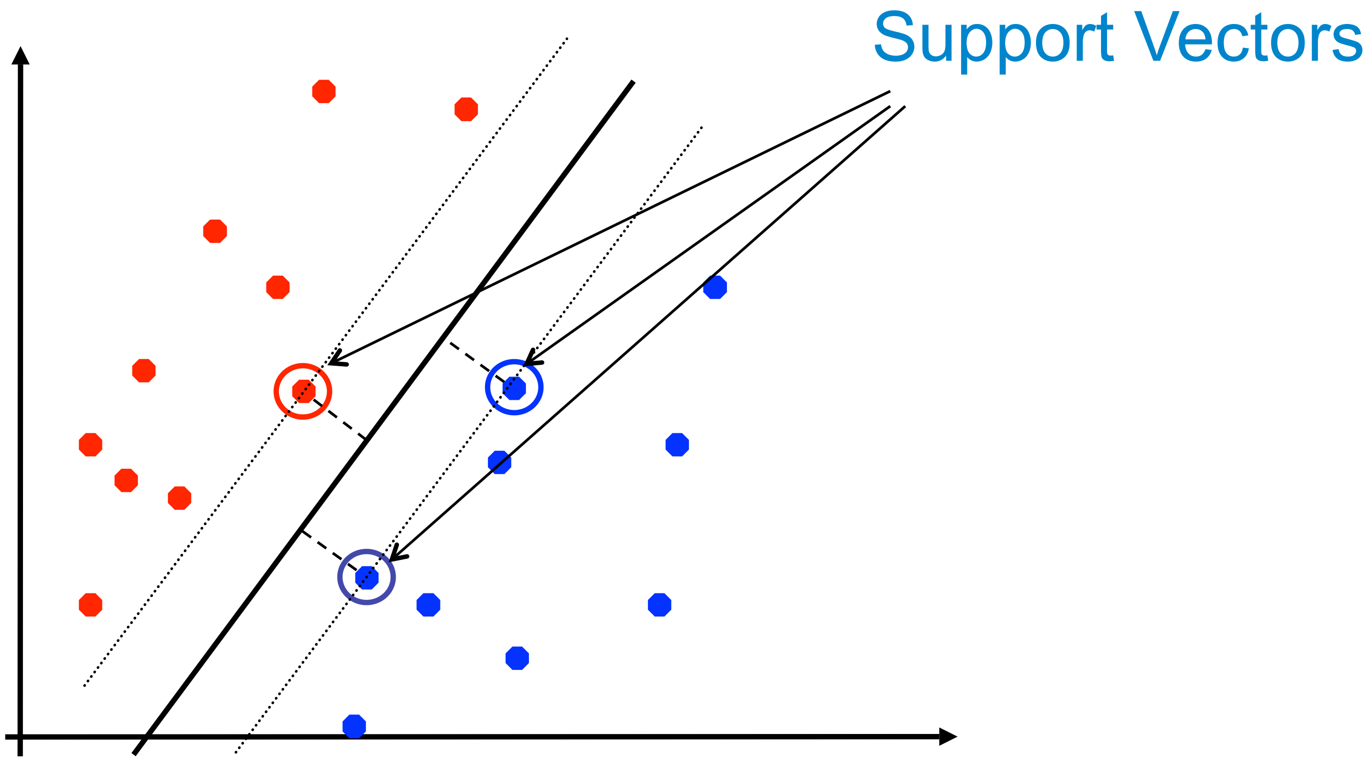
Classification Margin

- Distance from example $\mathbf{x}_i$ to the separator is $r = \frac{\mathbf{w}^T \mathbf{x}_i + b}{\|\mathbf{w}\|}$
- Examples closest to the hyperplane are ***support vectors***.
- ***Margin*** ρ of the separator is the distance between support vectors.



Maximum Margin Classification

- Place the linear boundary (line or hyperplane) such the margin is maximized.
- Implies that only support vectors matter; other training examples are ignorable.



Linear SVM Mathematically

- Let training set $\{(\mathbf{x}_i, y_i)\}_{i=1..n}$, $\mathbf{x}_i \in \mathbf{R}^d$, $y_i \in \{-1, 1\}$ be separated by a hyperplane with margin ρ. Then for each training example $(\mathbf{x}_i, y_i)$:

$$\begin{aligned} w^T x_i + b &\leq -\rho/2 \quad \text{if } y_i = -1 \\ w^T x_i + b &\geq \rho/2 \quad \text{if } y_i = 1 \end{aligned} \quad \Leftrightarrow \quad y_i(w^T x_i + b) \geq \rho/2$$

- For every support vector $\mathbf{x}_s$ the above inequality is an equality. After rescaling $\mathbf{w}$ and b by $\rho/2$ in the equality, we obtain that distance between each $\mathbf{x}_s$ and the hyperplane is

$$r = \frac{y_s(\mathbf{w}^T \mathbf{x}_s + b)}{\|\mathbf{w}\|} = \frac{1}{\|\mathbf{w}\|}$$

- Then the margin can be expressed through (rescaled) $\mathbf{w}$ and b as:

$$\rho = 2r = \frac{2}{\|\mathbf{w}\|}$$

SVM

- Maximising the distance is the same as minimising

$$\frac{1}{2} w \cdot w$$

- Subject to

$$y_i(w \cdot x_i + b) \geq 1$$

- If we introduce Lagrange multipliers the problem becomes

$$\frac{1}{2} w \cdot w - \sum_{1}^{N} \alpha_i (y_i(w \cdot x_i + b) - 1)$$

- Minimise wrt w and b
- Maximise wrt α_i
- Some math gymnastics gives

$$\sum_{1}^{N} \alpha_i y_i x_i = w \qquad \sum_{1}^{N} \alpha_i y_i = 0$$

SVM

- The hyperplane is determined by very few data points i.e. Most of the α_i are zero
- To classify a new data point:

$$f(x) = sign(w \cdot x + b)$$

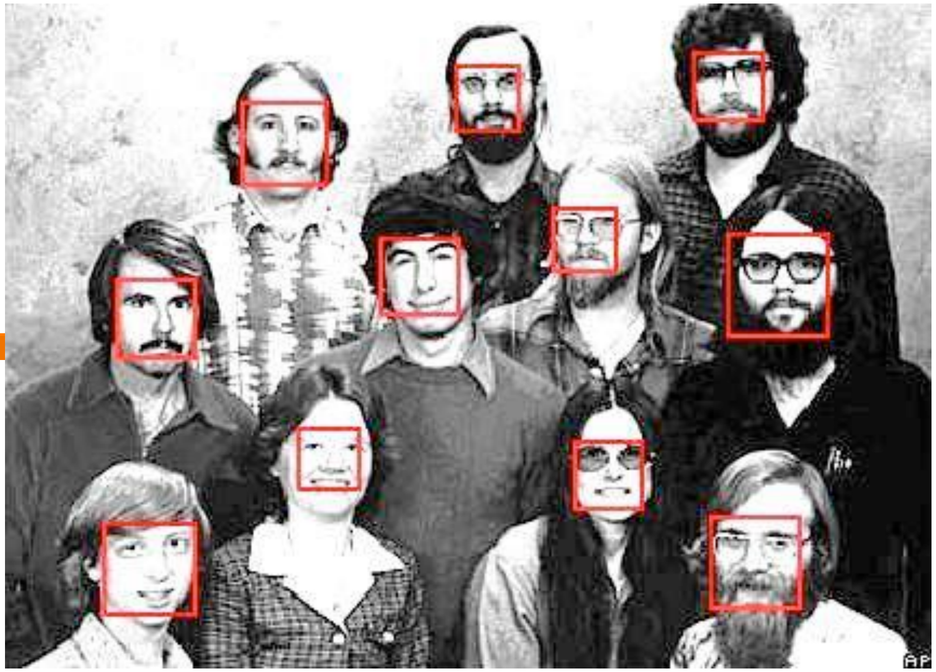
$$f(x) = sign(\sum_1^N (\alpha_i y_i x \cdot x_i + b))$$

  - Where the α_i are non-zero
  - Only have to calculate the support vectors
- More complexity in non-linear cases….

Using SVM to find people

- Papegeorgiou et al 1999
- Extract 1326 Harr wavelet features from sub images
- Build an SVM classifier
- Feature Selection
  - Reduce 1326 features to 29
  - ROC curves to compare performance
- Trade off accuracy vs. speedup in feature extraction

Feature Selection



- Consider a classification problem:
- What features?
  - Harr wavelets, raw pixels, HOG, GLCM entropy…..
  - How do we know which are useful?
  - Sometimes the vectors lie in a very high dimensional space, e.g., Raw Pixels from an image of size 256x256 – Feature Vector size is 65536
  - We need to prune the feature vectors
  - More on this tomorrow

Classifiers in Vision

- Classifiers are a means to an end in vision
- Trained with example images
- High dimensional problems
- Iterative path toward solution
  - Try lots of features
  - Perform Feature selection
  - Empirical comparison of performance
    - Accuracy vs speed
  - Performance tuning but beware of over fitting