Course: DD2427 - Exercise Class 2

Questions with an asterix(*) are a bit more involved and are more to aid understanding as opposed to representing potentional exam questions.

Exercises 1: *Linear Separability*

Given a set of data points $\{\mathbf{x}_i\}$, we can define the *convex hull* to be the set of all points $\mathbf{x}$ given by

$$\mathbf{x} = \sum_i \alpha_i \, \mathbf{x}_i$$

where $\alpha_i \geq 0$ and $\sum_i \alpha_i = 1$. Consider a set of points $\{\mathbf{y}_i\}$ together with their corresponding convex hull. By definition, the two sets of points will be linearly separable if there exists a vector $\hat{\mathbf{w}}$ and a scalar ω_0 such that $\hat{\mathbf{w}}^t \, \mathbf{x}_i + \omega_0 > 0$ for all $\mathbf{x}_i$, and $\hat{\mathbf{w}}^t \, \mathbf{x}_i + \omega_0 < 0$ for all $\mathbf{y}_i$. Show that if their convex hulls intersect, the two sets of points cannot be linearly separable, and conversely that if they are linearly separable, their convex hulls do not intersect.

Solution:

Part 1

Assume the convex hulls intersect and thus there exists $\mathbf{z}$ such that

$$\mathbf{z} = \sum_i \eta_i \, \mathbf{x}_i = \sum_j \gamma_j \, \mathbf{y}_j$$

where for all i, j

$$\alpha_i, \gamma_j \geq 0, \quad \text{and} \quad \sum_i \eta_i = \sum_j \gamma_j = 1$$

If the two sets of points are linearly separable, there exists a separating hyper-plane $\mathbf{w}, w_0$ such that

$$\begin{aligned} \mathbf{w}^t \, \mathbf{x}_i + w_0 > 0 \quad \forall i \\ \mathbf{w}^t \, \mathbf{y}_j + w_0 < 0 \quad \forall j \end{aligned}$$

Then

$$\begin{aligned} \mathbf{w}^t \mathbf{z} + w_0 &= \mathbf{w}^t \sum_i \eta_i \, \mathbf{x}_i + w_0 \\ &= \sum_i \eta_i \, \mathbf{w}^t \mathbf{x}_i + w_0 \sum_i \eta_i, \quad \text{as } \textstyle\sum_i \eta_i = 1 \\ &= \sum_i \eta_i \, (\mathbf{w}^t \mathbf{x}_i + w_0) \end{aligned}$$

Similarly

$$\mathbf{w}^T\mathbf{z} + w_0 = \sum_j \gamma_j \left(\mathbf{w}^T\mathbf{y}_j + w_0\right)$$

Thus

$$\sum_i \eta_i \left(\mathbf{w}^T\mathbf{x}_i + w_0\right) = \sum_j \gamma_j \left(\mathbf{w}^T\mathbf{y}_j + w_0\right)$$

But this implies that

$$\eta_i = 0 \ \forall i \text{ and } \gamma_j = 0 \ \forall j$$

as $\mathbf{w}^T\,\mathbf{x}_i + w_0 > 0 \ \forall i$ and $\mathbf{w}^T\,\mathbf{y}_j + w_0 < 0 \ \forall j$ and the $\eta_i' s$ and γ_j's are non-negative. However, this is impossible as $\sum_i \eta_i = \sum_j \gamma_j = 1$. Thus assuming the two sets of points are linearly separable leads us to a contradiction. Therefore this assumption is false.

Thus if the convex hulls of two separate sets of points intersect $\implies$ the two sets are not linearly separable.

Part 2

Assume that the two sets of points are linearly separable.

Then imagine that there exists a point $\mathbf{z}$ that belongs to both convex hulls, that is

$$\mathbf{z} = \sum_i \eta_i\,\mathbf{x}_i = \sum_j \gamma_j\,\mathbf{y}_j$$

Therefore

$$\begin{aligned}\mathbf{w}^T\mathbf{z} + w_0 &= \mathbf{w}^T \sum_i \eta_i\,\mathbf{x}_i + w_0 \\ &= \sum_i \eta_i\,\mathbf{w}^T\mathbf{x}_i + w_0 \sum_i \eta_i, \quad \text{as } \textstyle\sum_i \eta_i = 1 \\ &= \sum_i \eta_i \left(\mathbf{w}^T\mathbf{x}_i + w_0\right) > 0\end{aligned}$$

as each $\mathbf{w}^T\mathbf{x}_i + w_0 > 0$ and the η_j's are non-negative and $\sum \eta_j = 1$. The latter means that there is at least one $\eta_j > 0$. But we also have

$$\mathbf{w}^T\mathbf{z} + w_0 = \sum_j \gamma_j \left(\mathbf{w}^T\mathbf{y}_j + w_0\right) < 0$$

as each $\mathbf{w}^T\mathbf{y}_j + w_0 < 0$ and the γ_j's are non-negative and $\sum \gamma_j = 1$. The latter means that there is at least one $\gamma_j > 0$.

Therefore we have

$$\mathbf{w}^T\mathbf{z} + w_0 > 0 \quad \textbf{and} \quad \mathbf{w}^T\mathbf{z} + w_0 < 0$$

This is not possible. Therefore the assumption that the convex hulls intersect is incorrect.

Thus if the two sets of points are linearly separable $\implies$ their convex hulls cannot intersect.

Exercises 2: *Boosting*

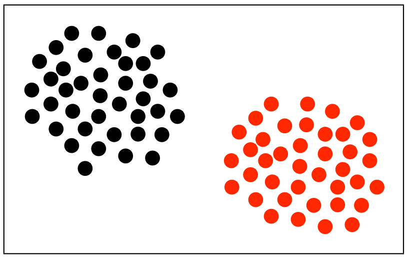
Imagine you have started your ex-jobb project and you have hired some first year students to label some training data into two classes for you. Unfortunately, the night before working for you the student spent the night partying until early in the morning. Thus he has created a labelled dataset with lots of labelling errors, say upto 20% of the data is misclassified. What consequences will this have for your project if you are building a classifier using a boosting mechanism? (Or indeed using a nearest neighbour classifier?)

What learning mechanism could I use instead to fit separate models to the two different classes which would ignore the labels?

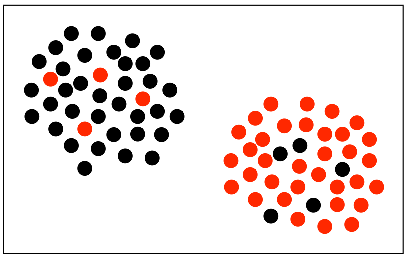
Solution:

Boosting puts increasing emphasis on the *hard* training examples. This is, of course, great if the data is correctly labelled. However, if training examples are incorrectly labelled then there can potentially be problems.

Consider the following example in the picture below. The left image shows the feature vectors from two classes where each example is correctly labelled. They are clearly linearly separable. Boosting with oriented lines as the weak classifiers could easily find a good strong classifier. However, in the right picture, it is the same feature data but some of them are now incorrectly labelled. Boosting will now devote many weak classifiers to correctly classifying the incorrectly labelled examples. This will undoubtedly lead to a strong classifier with a much more complicated strong classifier which will probably have an incorrect decision boundary.



Correctly labelled training data



Errors in labelling

Similarly, if you use a nearest neighbour classifier with data which has corrupted labels you can run into problems. Consider the decision boundary induced by a nearest neighbour classifier for the training data in the left figure.

You could, of course, use unsupervised learning! For instance k means cluster could be used to partition the feature vectors into two classes and then you could fit or you similarly you could use EM and fit a Gaussian mixture model to the training data.

Exercises 3: *Boosting II*

a) What does the *AdaBoost* algorithm produce from a set of weak classifiers and labelled training data?

b) Describe the steps of the *AdaBoost* algorithm.

c) In your opinion what is the critical step in the *AdaBoost* algorithm?

d) What quality must the weak classifiers possess in order for *AdaBoost* to run successfully?

e) What are the strengths/weakness of using boosting to solve classification problems?

Solution:

a) A strong classifier which is a weighted sum of weak classifiers which has a much better performance than each of the individual classifiers.

b) See lecture notes.

c) The re-weighting of the training examples. So that at the next round of boosting emphasis is put on *hard examples*.

d) They must perform better than chance.

e) **Strengths** Good generalization properties; it tends not to over-fit even when the training error has reached zero; **Weaknesses** Requires a lot of labelled training data; Training is slow; Not robust to errors in the labellings of the training data.

Exercises 4:

Imagine you have a face detector such that

$$P(\hat{f} = 1 \,|\, f = 1) = 1 - \epsilon \quad \textbf{and} \quad P(\hat{f} = 0 \,|\, f = 0) = 1 - \alpha$$

where $f \in \{0, 1\}$ indicates the ground truth of whether a face is present or not while $\hat{f} \in \{0, 1\}$ is the prediction of the face detector.

Imagine you have K *independent* face detectors each having the same true and false positive rate of the detector just described. If these detectors are applied to an image patch we get

$$\hat{\mathbf{f}} = (\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_K)$$

where $\hat{f}_i \in \{0, 1\}$ is the prediction of the ith detector. $k_{\hat{\mathbf{f}}} = \sum_i \hat{f}_i$ is equal to the number of detectors which predict a face while $K - k_{\hat{\mathbf{f}}}$ is the number which predict a non-face. Let γ be the prior probability that the patch contains a face. Given this information answer the following:

a) **(.2)** Write down the expression, remembering to exploit the independence, for

$$p(\hat{\mathbf{f}} \,|\, f = 1)$$

b) **(.2)** What is the posterior probability the patch contains a face given $\hat{\mathbf{f}}$?

c) **(.3)** Let $\gamma = \epsilon = \alpha = .01$ what is the constraint $k_{\hat{\mathbf{f}}}$ must fulfill such that

$$p(f = 1 \,|\, \hat{\mathbf{f}}) \geq .99$$

For $K = 4$, what is the minimal value of $k_{\hat{\mathbf{f}}}$ such that the above performance level is met? For $K = 10$? And as $K \to \infty$ what ratio of detections should correctly predict a face to ensure this level of performance.

d) **(.3)** Define a final classifier such that

$$F(\hat{\mathbf{f}}) = \begin{cases} 1 & \text{if } \sum_i \hat{f}_i \geq K_0 \\ 0 & \text{otherwise} \end{cases}$$

Continuing with the parameter settings just given, write down an expression for the $P(\text{error})$ of this classifier.

Solution:

a)

$$p(\hat{\mathbf{f}}\,|\,f=1) = \prod_{i=1}^{K} p(\hat{f}_i\,|\,f=1) = \prod_{i=1}^{K}(1-\epsilon)^{\hat{f}_i}\epsilon^{1-\hat{f}_i} = (1-\epsilon)^{\sum_i \hat{f}_i}\epsilon^{K-\sum_i \hat{f}_i}$$

$$= (1-\epsilon)^{k_{\hat{\mathbf{f}}}}\epsilon^{K-k_{\hat{\mathbf{f}}}}$$

b)

$$p(f=1\,|\,\hat{\mathbf{f}}) = \frac{p(\hat{\mathbf{f}}\,|\,f=1)\,p(f=1)}{p(\hat{\mathbf{f}})} = \frac{\gamma\,(1-\epsilon)^{k_{\hat{\mathbf{f}}}}\,\epsilon^{K-k_{\hat{\mathbf{f}}}}}{\gamma\,(1-\epsilon)^{k_{\hat{\mathbf{f}}}}\,\epsilon^{K-k_{\hat{\mathbf{f}}}} + (1-\gamma)\,\alpha^{k_{\hat{\mathbf{f}}}}\,(1-\alpha)^{K-k_{\hat{\mathbf{f}}}}}$$

c) As $\gamma = \alpha = \gamma$ then

$$\begin{aligned} p(f=1\,|\,\hat{\mathbf{f}}) &= \frac{\epsilon\,(1-\epsilon)^{k_{\hat{\mathbf{f}}}}\,\epsilon^{K-k_{\hat{\mathbf{f}}}}}{\epsilon\,(1-\epsilon)^{k_{\hat{\mathbf{f}}}}\,\epsilon^{K-k_{\hat{\mathbf{f}}}} + (1-\epsilon)\,\epsilon^{k_{\hat{\mathbf{f}}}}\,(1-\epsilon)^{K-k_{\hat{\mathbf{f}}}}} \\ &= \frac{(1-\epsilon)^{k_{\hat{\mathbf{f}}}}\,\epsilon^{K-k_{\hat{\mathbf{f}}}+1}}{(1-\epsilon)^{k_{\hat{\mathbf{f}}}}\,\epsilon^{K-k_{\hat{\mathbf{f}}}+1} + \epsilon^{k_{\hat{\mathbf{f}}}}\,(1-\epsilon)^{K-k_{\hat{\mathbf{f}}}+1}} \geq A \end{aligned}$$

This implies that

$$\begin{aligned} (1-\epsilon)^{k_{\hat{\mathbf{f}}}}\,\epsilon^{K-k_{\hat{\mathbf{f}}}+1}(1-A) &\geq A\,\epsilon^{k_{\hat{\mathbf{f}}}}\,(1-\epsilon)^{K-k_{\hat{\mathbf{f}}}+1} \\ k_{\hat{\mathbf{f}}}\log(1-\epsilon) + (K-k_{\hat{\mathbf{f}}}+1)\log\epsilon + \log(1-A) &\geq \log A + k_{\hat{\mathbf{f}}}\log\epsilon + (K-k_{\hat{\mathbf{f}}}+1)\log(1-\epsilon) \\ 2\,k_{\hat{\mathbf{f}}}(\log(1-\epsilon) - \log\epsilon) &\geq \log A - \log(1-A) + (K+1)(\log(1-\epsilon) - \log\epsilon) \end{aligned}$$

and finally

$$k_{\hat{\mathbf{f}}} \geq \frac{\log A - \log(1-A)}{2(\log(1-\epsilon) - \log\epsilon)} + \frac{1}{2}(K+1)$$

Substituting in the values of $\epsilon = .01$ and $A = .99$ get

$$k_{\hat{\mathbf{f}}} \geq .5\,K + 1$$

d) There are two types of error we can make. The first is to classify a face patch as a non-face patch. The probability of this occurring is:

$$\begin{aligned} P(F(\hat{\mathbf{f}}) = 0|f=1) = \sum_{\text{all } \hat{\mathbf{f}} \text{ s.t. } \sum \hat{f}_i < K_0} p(\hat{\mathbf{f}}|f=1) &= \sum_{k=0}^{K_0-1}\ \sum_{\text{all } \hat{\mathbf{f}} \text{ s.t. } \sum \hat{f}_i = k} p(\hat{\mathbf{f}}|f=1) \\ &= \sum_{k=0}^{K_0-1}\ \sum_{\text{all } \hat{\mathbf{f}} \text{ s.t. } \sum \hat{f}_i = k} (1-\epsilon)^k\epsilon^{K-k} \\ &= \sum_{k=0}^{K_0-1}\binom{K}{k}(1-\epsilon)^k\epsilon^{K-k} \end{aligned}$$

While the second is to classify a non-face patch as a face patch. The probability of this occurring is:

$$p(F(\hat{\mathbf{f}}) = 1|f = 0) = \sum_{\text{all } \hat{\mathbf{f}} \text{ s.t. } \sum \hat{f}_i \geq K_0} p(\hat{\mathbf{f}}|f = 0) = \sum_{k=K_0}^{K} \binom{K}{k} \alpha^k (1-\alpha)^{K-k}$$

Therefore the probability of error

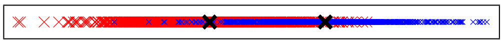
$$\begin{aligned} P(\text{error}) &= P(f = 0)\, P(F(\hat{\mathbf{f}}) = 1 \,|\, f = 0) + P(f = 1)\, P(F(\hat{\mathbf{f}}) = 0 \,|\, f = 1) \\ &= (1-\gamma) \sum_{k=K_0}^{K} \binom{K}{k} \alpha^k (1-\alpha)^{K-k} + \gamma \sum_{k=0}^{K_0-1} \binom{K}{k} (1-\epsilon)^k \epsilon^{K-k} \end{aligned}$$

Exercises 5: *k means clustering*

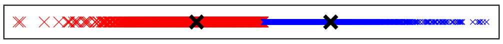
Consider the k-means algorithm applied to a large amount of one-dimensional data that comes from either of one of two classes with equal prior probability. The class conditional distribution for each class is Gaussian with true means $\mu = \pm 1$ and both have standard deviation $\sigma = 1$. What happens when you apply the k-means algorithm with $k = 2$ to this data? What can you say about the means of the two clusters found and the mean of the class-conditional distributions.

Solution:

The left image shows training examples randomly generated from each class. If k-means clustering is applied to this data then the clusters found are shown in the right image. Note the means of the distributions and the clusters are shown by black crosses.



Training examples plus labels

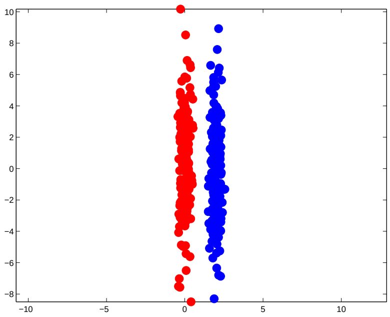


Labels after k-means clustering

The means of the clusters found by k-means are -1.2058 and 1.1488. Thus, the distance between the cluster centres is larger than the distance between the means of the class-conditional distributions.

Exercises 6: *k means clustering II*

Consider these two clusters



a) Will the standard k-means algorithm have a problem with finding the two clusters even if good initial guesses of the cluster centres are given?

b) Why?

c) How could we quickly fix this problem?

Solution:

a) Yes.

b) In the $k-$means algorithm each point is assigned to the cluster such that

$$i^* = \min_i \ \|\mathbf{x} - \boldsymbol{\mu}_i\|^2$$

However, if a cluster has more variation in one direction over another then a measure of cluster membership using the Mahalobnis distance would be a better bet.

c) One could adapt the $k-$means algorithm such that this cost score is minimized

$$\sum_{i=1}^{K} \sum_{\mathbf{x} \in \mathcal{C}_i} (\mathbf{x} - \boldsymbol{\mu}_i)^t \ \Sigma_i^{-1} \ (\mathbf{x} - \boldsymbol{\mu}_i)$$

where Σ_i is the estimated covariance matrix of the ith cluster which would be updated after data point was assigned to a cluster using this measure. This measure allows one to model elongated and rotated cluster blobs.

Exercises 7: *Decision stumps*

The type of weak binary classifier for data $\mathbf{x} \in \mathbb{R}^d$ you have been using in the face lab has a name. It is called a *decision stump*. As you know the classification rule has parameters $q \in \{-1, 1\}, j \in \{1, 2, \ldots, d\}$ and θ and takes the form:

$$h(\mathbf{x}; j, q, \theta) = q \times \operatorname{sgn}(x_j - \theta)$$

Decision stumps classify example $\mathbf{x}$ based only on the value of its jth coordinate. θ is a threshold value in $\mathbb{R}$ and q is the parity.

Decision stumps, by themselves, are not very powerful classifiers. For instance, a single vertical or horizontal decision stump can only shatter 2 points in $\mathbb{R}^2$. However, combining multiple decision stumps can give rise to more complex classifiers, as you do in the boosting algorithm. In this part, we calculate the VC dimensions of some combinations of decision stumps.

For points in R^2, calculate the VC-dimension of the following sets of classifiers:

a) Convex combinations (i.e. coefficients must be non-negative and sum to 1) of two vertical decision stumps.

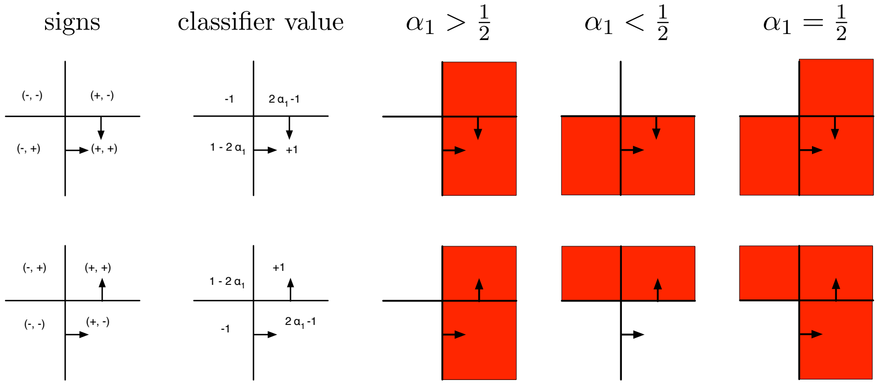
b) Convex combinations of one vertical and one horizontal decision stump.

Solution:

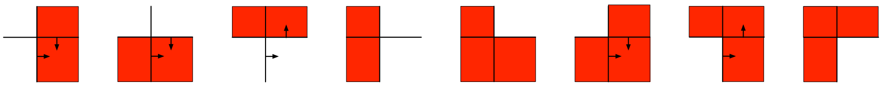
a) Given two decision stump classifiers $h_1(\mathbf{x})$ and $h_2(\mathbf{x})$, the classifier obtained as a convex combination is given by $\operatorname{sgn}(\alpha_1 h_1(\mathbf{x}) + \alpha_2 h_2(\mathbf{x}))$. As stated in the problem set, a single vertical decision stump (and hence also a convex combination of two vertical decision stumps) can shatter 2 points in $\mathbb{R}^2$. However, no set of 3 points can be shattered. For the purpose of labelling points using vertical decision stumps, we need only consider the horizontal coordinates of the points. Let these be x_1, x_2 and x_3. Further, let $\operatorname{sgn}(0)$ be equal to $+1$. In this case, the labelling $x_1 = -1, x_2 = +1, x_3 = -1$ is not possible. Changing the definition of $\operatorname{sgn}(0)$ does not help, as then the case $x_1 = +1, x_2 = -1, x_3 = +1$ is not possible.

b) Consider 3 points that form an equilateral triangle, one of whose sides is parallel to the horizontal axis. Any required labelling of these 3 points can be obtained by using either a single horizontal or a single vertical decision stump. Thus, the set of convex combinations of a horizontal and a vertical decision stump (in particular, the subset where one of the two weights in the combination is unity and the other zero) can shatter 3 points.

This convex combination divides the $xy-$plane into four quadrants. Let the red color denote region the quadrants which generate a positive value. We assume that sgn (0) is equal to +1 (it is easy to show that the opposite assumption leads to equivalent results.)



Enumerating all the possible combination of the two base classifiers we can split the $xy-$plane in the following ways:



No set of 4 points can be shattered by the given set of classifiers. To see this consider the two cases in which 4 points can be arranged (ignoring the cases of 3 or more collinear points, which can clearly not be shattered) with the above pictures to help you.

- The convex hull of the 4 points is a triangle, with one point lying strictly inside this convex hull: in this case, labelling the points at the vertices of the triangle as -1 and the interior point as +1 is not possible. This is because the interior point cannot lie in a half-space that does not contain any of the other three points.
- The convex hull of the 4 points is a quadrilateral: in this case, one of the two labellings of non-adjacent vertices – both+1, remaining vertices -1, or both-1, remaining vertices +1 – is not possible.

Exercises 8: *VC-dimension I*

Remember that in order to prove that a class of functions $\mathcal{H}$ has VC-dimension d you need to show that

- There exists a set of d points which can be **shattered** by $\mathcal{H}$.
- There exists **no** set of $d + 1$ points that can be shattered by $\mathcal{H}$

a) When does a class of functions $\mathcal{H}$ *shatter* a set of points $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$?

b) Show with appropriate diagrams that there exists 3 points in $\mathbb{R}^2$ that can be shattered by a line.

c) What is the VC-dimension of intervals in $\mathbb{R}$? In this case $\mathcal{H}$ is defined such that each $h \in \mathcal{H}$ is associated with an interval $[a, b]$ and $x \in \mathbb{R}$ has $h(x) = 1$ if and only if $x \in [a, b]$.

d) What is the VC-dimension of the union of k intervals on the real line? In other words each $h \in \mathcal{H}$ is associated with k closed intervals $[a_i, b_i]$, $i = 1, 2, \ldots, k$ and $h(x) = 1$ if and only if $x \in \cup_{i=1}^{n} [a_i, b_i]$.

e) What is the VC-dimension of axis parallel rectangles in $\mathbb{R}^2$? In other words $h \in \mathcal{H}$ is associated with 2 closed intervals $[a_i, b_i]$ for $i = 1, 2$ and then for any $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$, $h(\mathbf{x}) = 1$ if and only if $x_i \in [a_i, b_i]$ for $i = 1, 2$.

f) Show that the VC-dimension of the class $\mathcal{H}$ of hyperplanes in $\mathbb{R}^2$ is 3?

g) Show that the VC-dimension of the class $\mathcal{H}$ of hyperplanes in $\mathbb{R}^d$ is $\geq d + 1$?

Solution:

a) See lecture notes.

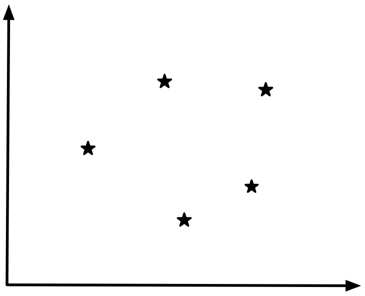
b) See lecture notes.

c) 2. Obviously can't shatter 3 points as the x o x cannot be classified correctly by a single interval.

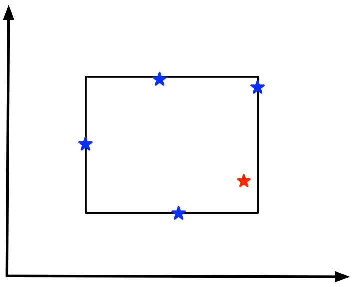
d) The hardest case to correctly classify is o x o x $o \ldots$ x o x. k intervals can be used to define class x. Thus one can correctly $2k$ points.

e) Can shatter 4 points but cannot shatter 5 points.

For any set of 5 points, choose the 4 points that have the max and min in the first and the second coordinates. Then by definition, (assuming no ties) the fifth point must be inside a rectangle given by the max and min points. In this arrangement we cannot assign 1 on the edges and 0 inside. Thus, this proves that we cannot shatter 5 or more points with rectangles.



(a) 5 points



(b) bounding box of the points

f) See lecture notes.

g) See lecture notes.

Exercises 9: *VC-dimension II**

Prove that a oriented hyper-plane cannot shatter $d+2$ points in $\mathbb{R}^d$.

Solution:

Before we prove the above we need the following result:

Radon's Theorem Let $\mathcal{S}$ be set of $d+2$ points in d dimensions. Then $\mathcal{S}$ can be partitioned into two (disjoint) subsets $\mathcal{S}_1$ and $\mathcal{S}_2$ whose convex hulls intersect.

Prove of Radon's Theorem

Assume we have $d+2$ points $\mathbf{x}_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,d})^T$ and construct the matrix B of size $(d+1) \times (d+2)$:

$$B = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_{1,1} & x_{2,1} & x_{3,1} & \cdots & x_{d+2,1} \\ x_{1,2} & x_{2,2} & x_{3,2} & \cdots & x_{d+2,2} \\ & & & & \\ & & & & \\ x_{1,d} & x_{2,d} & x_{3,d} & \cdots & x_{d+2,d} \end{pmatrix}$$

Clearly, since the rank of this matrix is at most $d+1$, the columns are linearly dependent. Let $\boldsymbol{\lambda} = (\lambda_1, x_1, \ldots, \lambda_{d+2})$ be a non-zero vector such that $B\boldsymbol{\lambda} = \mathbf{0}$. This means that

$$\lambda_1 \mathbf{x}_1 + \lambda_2 \mathbf{x}_2 + \cdots + \lambda_{d+2} \mathbf{x}_{d+2} = \mathbf{0} \tag{1}$$

$$\lambda_1 + \cdots + \lambda_{d+2} = 0. \tag{2}$$

Let $\mathcal{S}_1 = \{\mathbf{x}_i : \lambda_i > 0\}$ and let $\mathcal{S}_2 = \{\mathbf{x}_i : \lambda_i \leq 0\}$ and then from equation (1)

$$\sum_{i:\mathbf{x}_i \in \mathcal{S}_1} \lambda_i \mathbf{x}_i = - \sum_{i:\mathbf{x}_i \in \mathcal{S}_2} \lambda_i \mathbf{x}_i$$

Let $L = \sum_{i:\mathbf{x}_i \in \mathcal{S}_1} \lambda_i$ and note that this implies $\sum_{i:\mathbf{x}_i \in \mathcal{S}_2} \lambda_i = -L$ via equation (2). Then

$$p = \sum_{i:\mathbf{x}_i \in \mathcal{S}_1} \frac{\lambda_i}{L} \mathbf{x}_i$$

is in the convex hull of $\mathcal{S}_1$, using the fact that the λ_i's and L are non-negative. But p is also in the convex hull of $\mathcal{S}_2$ since

$$- \sum_{i:\mathbf{x}_i \in \mathcal{S}_2} \frac{\lambda_i}{L} \mathbf{x}_i = \sum_{i:\mathbf{x}_i \in \mathcal{S}_1} \frac{\lambda_i}{L} \mathbf{x}_i = p$$

So the convex hulls intersect.

Proof that a oriented hyper-plane cannot shatter $d + 2$ points in $\mathbb{R}^d$.

If $\mathcal{S}$ is a set of $d + 2$ points, then by Radon's theorem we may partition $\mathcal{S}$ into sets $\mathcal{S}_1$ and $\mathcal{S}_2$ whose convex hulls intersect. Let p be a point in that intersection. No hyperplane can have $\mathcal{S}_1$ on one side and $\mathcal{S}_2$ on the other since that would imply that the convex hull of $\mathcal{S}_1$ is on one side and the convex hull of $\mathcal{S}_2$ is on the other. which means that p is on both sizes. So, no set of $d + 2$ points can be shattered.

Exercises 10: *Lagrange Multipliers*

Minimize

$$(x_1 - \frac{3}{2})^2 + (x_2 - \frac{1}{8})^2$$

subject to

$$x_1^2 + x_2^2 \leq 1$$

Exercises 11: *SVM*

i) Consider the degree-two polynomial kernel defined by $K(\mathbf{x}, \mathbf{z}) = (1 + \mathbf{x}^T\mathbf{z})^2$. Expand this out completely for the three-dimensional case (i.e., $\mathbf{x} = \langle x_1, x_2, x_3\rangle$ and $\mathbf{z} = \langle z_1, z_2, z_3\rangle$. Verify that this has the same form as the quadratic expansion, although with different coefficients on the terms.

ii) Continuing from the previous question, what is the form of Φ so that $K(\mathbf{x}, \mathbf{z}) = \Phi(\mathbf{x})^T\Phi(\mathbf{z})$? (You need only consider the three-dimensional data case.) How does this differ from the expansion

$$\Phi(\mathbf{x}) = \langle x_1, x_2, x_3, x_1^2, x_2^2, x_3^2, x_1x_2, x_1x_3, x_2x_3\rangle?$$

iii) Consider optimizing an SVM with *squared* loss on the ξ variables. That is, an optimization problem of the form:

$$\begin{aligned}
&\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + \lambda\sum_n \xi_n^2 \text{ s.t.} \\
&\quad y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 - \xi_n \qquad (\forall n) \\
&\quad \xi_n \geq 0 \qquad (\forall n)
\end{aligned}$$

Construct the dual formulation for this problem. In particular, construct the Lagrangian, optimize it with respect to $\mathbf{w}$ and b, plug these solutions back in and get an optimization problem just in terms of the dual (Lagrange) variables $\boldsymbol{\alpha}$. How does this compare to the dual formulation for the standard SVM?

iv) For D dimensional data, consider using the degree d polynomial kernel defined by $K(\mathbf{x}, \mathbf{z}) = (1 + \mathbf{x}^T\mathbf{z})^d$. What is the general form of the expansion? What are the coefficients on all the different forms in the expansion?

Solution:

i)

$$\begin{aligned}
K(\mathbf{x}, \mathbf{z}) &= (1 + \mathbf{x}^T\mathbf{z})^2 \\
&= (1 + (x_1, x_2, x_3)\cdot(z_1, z_2, z_3))^2 \\
&= (1 + x_1\,z_1 + x_2\,z_2 + x_3\,z_3)^2 \\
&= 1 + 2\sum_{i=1}^{3} x_iz_i + \sum_{i=1}^{3} x_i^2z_i^2 + 2(x_1z_1x_2z_2 + x_1z_1x_3z_3 + x_3z_3x_2z_2)
\end{aligned}$$

ii) Want

$$\Phi(\mathbf{x})^T\Phi(\mathbf{z}) = 1 + 2\sum_{i=1}^{3} x_i z_i + \sum_{i=1}^{3} x_i^2 z_i^2 + 2(x_1 z_1 x_2 z_2 + x_1 z_1 x_3 z_3 + x_3 z_3 x_2 z_2)$$

Thus

$$\Phi(\mathbf{x}) = (1, \sqrt{2}\,x_1, \sqrt{2}\,x_2, \sqrt{2}\,x_3, x_1^2, x_2^2, x_3^2, \sqrt{2}\,x_1\,x_2, \sqrt{2}\,x_1\,x_3, \sqrt{2}\,x_2\,x_3)^T$$

This differs from

$$\Phi_0(\mathbf{x}) = (x_1, x_2, x_3, x_1^2, x_2^2, x_3^2, x_1 x_2, x_1 x_3, x_2 x_3)$$

in that in this latter transformation there is no constant term and also in some of the weightings of the terms.

iii) This is the constrained optimization problem we want to solve

$$\begin{aligned} &\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i^2 \text{ s.t.} \\ &y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i \qquad (\forall i) \\ &\xi_i \geq 0 \qquad (\forall i) \end{aligned}$$

The Lagrangian is:

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \mathbf{r}) = \frac{1}{2}\mathbf{w}^T\,\mathbf{w} + C\sum_i \xi_i^2 + \sum_{i=1}^{n} \lambda_i\left[1 - \xi_i - y_i(\mathbf{w}^T\mathbf{x}_i + b)\right] - \sum_{i=1}^{n} r_i\,\xi_i$$

Taking the derivative of $\mathcal{L}$ w.r.t. $\mathbf{w}, b$ and $\boldsymbol{\xi}$ we get:

$$\begin{aligned} \frac{\partial\mathcal{L}}{\partial\mathbf{w}} &= \mathbf{w} - \sum \lambda_i y_i \mathbf{x}_i, \\ \frac{\partial\mathcal{L}}{\partial b} &= -\sum \lambda_i y_i \\ \frac{\partial\mathcal{L}}{\partial\xi_j} &= 2\,C\,\xi_j - \lambda_j - r_j \quad \text{for } j = 1, 2, \ldots, n \end{aligned}$$

and setting these equal to zero get:

$$\mathbf{w}^* = \sum \lambda_i y_i \mathbf{x}_i, \quad \sum \lambda_i y_i = 0, \quad \lambda_j + r_j = 2\,C\,\xi_j$$

Plugging these back into the Lagrangian and after some algebra get:

$$\Theta(\boldsymbol{\lambda}, \mathbf{r}) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2C}\sum(\lambda_i + r_i)^2 - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i\,\lambda_j\,y_i\,y_j\,\mathbf{x}_i^T\,\mathbf{x}_j$$

The dual formulation of the problem is then

$$\boxed{\max_{\boldsymbol{\lambda},\mathbf{r}} \left\{ \sum_{i=1}^{n} \lambda_i - \frac{1}{2C} \sum (\lambda_i + r_i)^2 - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \, \lambda_j \, y_i \, y_j \, \mathbf{x}_i^T \, \mathbf{x}_j \right\}}$$

subject to

$$\boxed{r_j \geq 0, \ \ \lambda_j \geq 0 \ \text{ for } j = 1, \ldots, n \quad \textbf{and} \quad \sum_{i=1}^{n} \lambda_i y_i = 0}$$

This can be simplified to

$$\boxed{\max_{\boldsymbol{\lambda}} \left\{ \sum_{i=1}^{n} \lambda_i - \frac{1}{2C} \sum \lambda_i^2 - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \, \lambda_j \, y_i \, y_j \, \mathbf{x}_i^T \, \mathbf{x}_j \right\}}$$

subject to

$$\boxed{\lambda_j \geq 0 \ \text{ for } j = 1, \ldots, n \quad \textbf{and} \quad \sum_{i=1}^{n} \lambda_i y_i = 0}$$

as the terms involving the r_i's are always negative and are maximized when each $r_i = 0$.

iv) A very useful result is the **binomial theorem** it states:

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^{n-k} \, y^k$$

Therefore

$$(1+x)^n = \sum_{k=0}^{n} \binom{n}{k} x^k$$

and the multinomial theorem which states that

$$(x_1 + x_2 + \cdots + x_n)^s = \sum_{\substack{j_1, j_2, \ldots, j_n \\ 0 \leq j_i \leq s \text{ for each } i \\ \text{and } j_1 + \cdots + j_k = s}} \frac{s\,!}{j_1! \, j_2! \cdots j_n!} \, x_1^{j_1} x_2^{j_2} \ldots x_n^{j_n}$$

Let $v_i = x_i z_i$ then

$$K(\mathbf{x}, \mathbf{z}) = (1 + \mathbf{x}^T \mathbf{z})^d = \sum_{s=0}^{d} \binom{d}{s} (\mathbf{x}^T \mathbf{z})^s = \sum_{s=0}^{d} \binom{d}{s} \left(\sum_{i=1}^{n} v_i \right)^s$$

$$= \sum_{s=0}^{d} \binom{d}{s} \sum_{\substack{j_1, j_2, \ldots, j_n \\ 0 \leq j_i \leq s \text{ for each } i \\ \text{and } j_1 + \cdots + j_k = s}} \frac{s\,!}{j_1! \, j_2! \cdots j_n!} \, v_1^{j_1} v_2^{j_2} \ldots v_n^{j_n}$$

Exercises 12: *Mixture Models*

a) Sketch this one dimensional probability distribution

$$p(x) = \pi_1 \mathcal{N}(0, .5) + (1 - \pi_1) \mathcal{N}(5, 1)$$

when $\pi_1 = \frac{1}{2}$; $\pi_1 = .1$; **and** $\pi_1 = .9$.

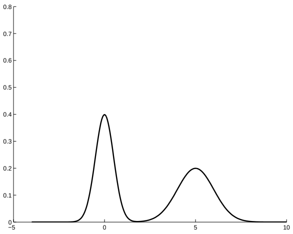
b) If you have n points generated from $p(x)$ when $\pi_1 = \frac{1}{2}$ and you fit a Gaussian distribution to this data. Sketch what this distribution will look like. What's the problem here?

c) This issue highlights a problem with parametric methods. What is it?

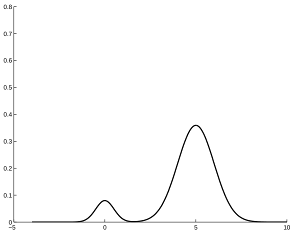
d) What method is used to find the parameters of a Gaussian mixture model from training examples generated from the distribution?

Solution:

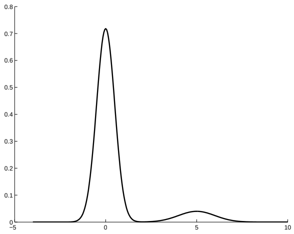
a)



$\pi_1 = .5$

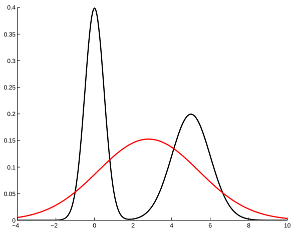


$\pi_1 = .1$



$\pi_1 = .9$

b) In this case the mean of the n sampled points will be between the two modes of $p(x)$. Therefore the peak of the estimated distribution will occur aver a region where $p(x)$ is close to zero.



c) If the possible shape of your parametric curve does not match that of the true distribution then you will not get a could estimate of this distribution.

d) If one knew the number of clusters one could use the EM algorithm.

Exercises 13: *Kernel SVM**

Show that the radial basis function is a valid kernel function

$$k(\mathbf{x},\mathbf{z}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{z}\|^2}{2\sigma^2}\right)$$

Solution:

To show the above we will need the following results - Say $k_1(.,.)$ and $k_2(.,.)$ are valid kernels then the $k(.,.)$'s given by

$$\begin{aligned}
&\mathbf{R1}: \quad k(\mathbf{x},\mathbf{z}) = k_1(\mathbf{x},\mathbf{z}) + k_2(\mathbf{x},\mathbf{z}) && (3)\\
&\mathbf{R2}: \quad k(\mathbf{x},\mathbf{z}) = k_1(\mathbf{x},\mathbf{z})\, k_2(\mathbf{x},\mathbf{z}) && (4)\\
&\mathbf{R3}: \quad k(\mathbf{x},\mathbf{z}) = a\, k_1(\mathbf{x},\mathbf{z}) \text{ for all } a \in \mathcal{R}^+ && (5)\\
&\mathbf{R4}: \quad k(\mathbf{x},\mathbf{z}) = k_1(\mathbf{x},\mathbf{z}) + c \text{ for all } c \in \mathcal{R}^+ && (6)\\
&\mathbf{R5}: \quad k(\mathbf{x},\mathbf{z}) = f(\mathbf{x})\, f(\mathbf{z}) \text{ for any } f : \mathcal{X} \to \mathcal{R} && (7)
\end{aligned}$$

are also valid kernels

There are several parts to this proof.

Result 1

First we will show that if $k_1(\mathbf{x},\mathbf{z})$ is a valid kernel then so is

$$k(\mathbf{x},\mathbf{z}) = \exp\left(k_1(\mathbf{x},\mathbf{z})/\sigma^2\right)$$

Now remember

$$e^x = \sum_{i=0}^{\infty} \frac{x^j}{j!}$$

Therefore

$$k(\mathbf{x},\mathbf{z}) = 1 + \sum_{i=1}^{\infty} \frac{1}{j!\,\sigma^{2j}} \left(k_1(\mathbf{x},\mathbf{z})\right)^j$$

Part a

First we will show that each

$$\frac{1}{j!\,\sigma^{2j}} \left(k_1(\mathbf{x},\mathbf{z})\right)^j$$

is a valid kernel.

Note that

$$\left(k_1(\mathbf{x},\mathbf{z})\right)^2 = k_1(\mathbf{x},\mathbf{z})\, k_1(\mathbf{x},\mathbf{z})$$

therefore $(k_1(\mathbf{x},\mathbf{z}))^2$ is a valid kernel as $k_1(\mathbf{x},\mathbf{z})$ is a valid kernel from **R2**. If $(k_1(\mathbf{x},\mathbf{z}))^{j-1}$ is a valid kernel, then

$$(k_1(\mathbf{x},\mathbf{z}))^j = \underbrace{k_1(\mathbf{x},\mathbf{z})}_{\text{valid kernel}}\ \underbrace{(k_1(\mathbf{x},\mathbf{z}))^{j-1}}_{\text{valid kernel}}$$

is a valid kernel by **R2**. Thus by induction if $k_1(\mathbf{x},\mathbf{z})$ is a valid kernel then $(k_1(\mathbf{x},\mathbf{z}))^j$ is also valid kernel (where j a positive integer).

Next let $k_2(\mathbf{x},\mathbf{z}) = (k_1(\mathbf{x},\mathbf{z}))^j$ and $a = \frac{1}{j!\,\sigma^{2j}}$. By definition $a > 0$ therefore $a\,k_2(\mathbf{x},\mathbf{z})$ is a valid kernel from **R3**.

Thus

$$\frac{1}{j!\,\sigma^{2j}}\ (k_1(\mathbf{x},\mathbf{z}))^j$$

is a valid kernel.

Part b

Using induction and **R1** it is obvious that

$$\sum_{j=1}^{n} k_j(\mathbf{x},\mathbf{z})$$

is a valid kernel if each $k_j(\mathbf{x},\mathbf{z})$ is a valid kernel. Therefore

$$\sum_{i=1}^{\infty} \frac{1}{j!\,\sigma^{2j}}\ (k_1(\mathbf{x},\mathbf{z}))^j$$

is a valid kernel and then by **R4**

$$\exp(k_1(\mathbf{x},bz)/\sigma^2) = 1 + \sum_{i=1}^{\infty} \frac{1}{j!\,\sigma^{2j}}\ (k_1(\mathbf{x},\mathbf{z}))^j$$

is a valid kernel.

Result 2

Next we will show that

$$k(\mathbf{x},\mathbf{z}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{z}\|^2}{2\sigma^2}\right)$$

is a valid kernel.

Firstly

$$\|\mathbf{x}-\mathbf{z}\|^2 = (\mathbf{x}-\mathbf{z})^t(\mathbf{x}-\mathbf{z}) = \mathbf{x}^t\mathbf{x} + \mathbf{z}^t\mathbf{z} - 2\mathbf{x}^t\mathbf{z}$$

thus

$$
\begin{aligned}
k(\mathbf{x},\mathbf{z}) &= \exp\left(-\frac{\mathbf{x}^t\mathbf{x}}{2\sigma^2}\right)\exp\left(-\frac{\mathbf{z}^t\mathbf{z}}{2\sigma^2}\right)\exp\left(\frac{\mathbf{x}^t\mathbf{z}}{\sigma^2}\right)\\
&= \qquad \underbrace{f(\mathbf{x})\, f(\mathbf{z})}_{\text{valid kernel by }\mathbf{R5}} \qquad\qquad \underbrace{\exp\left(\frac{\mathbf{x}^t\mathbf{z}}{\sigma^2}\right)}_{\text{valid kernel by the previous result}}
\end{aligned}
$$

where $f(\mathbf{x}) = \exp(-.5\,\mathbf{x}^t\mathbf{x}/\sigma^2)$ and set $k_1(\mathbf{x},\mathbf{z}) = \mathbf{x}^t\mathbf{z}$ is obviously a valid kernel function! Therefore $k(\mathbf{x},\mathbf{z})$ is the product of two valid kernels therefore it is a valid kernel.

Exercises 14: *Kernel magic**

Assume you are given m one dimensional training examples and their associated labels, that is $\{(x_i, y_i)\}_{i=1}^m$ where each $x_i \in \mathbb{R}^1$ and $y_i \in \{-1, +1\}$.

a) Draw a case where you have $m = 3$ training examples which are not linearly separable.

b) You know if you transform your one-dimensional data to a higher dimensional space then there is a higher likelihood that they will be linearly separable. Thus you define a feature transformation $\phi_n : \mathbb{R}^1 \to \mathbb{R}^n$ where

$$\phi_n(x) = \left(e^{-\frac{x^2}{2}}, x\, e^{-\frac{x^2}{2}}, \frac{x^2}{\sqrt{2}}\, e^{-\frac{x^2}{2}}, \ldots, \frac{x^n}{\sqrt{n!}}\, e^{-\frac{x^2}{2}}\right)$$

Explain why any set of 3 points (with no duplicates) can be linearly separated when transformed via ϕ_2. Similarly explain why any set of $n+1$ points (with no duplicates) can be linearly separated when transformed by ϕ_n.

c) Consider the case when $n \to \infty$ and ϕ_n becomes

$$\phi_\infty(x) = \left\{e^{-\frac{x^2}{2}}, x\, e^{-\frac{x^2}{2}}, \frac{x^2}{\sqrt{2}}\, e^{-\frac{x^2}{2}}, \ldots, \frac{x^j}{\sqrt{j!}}\, e^{-\frac{x^2}{2}}, \ldots\right\}$$

Can you explicitly construct $\phi_\infty(x)$? (Not a trick question)

d) Is there a finite set of points, containing no duplicates, that cannot be linearly separated after applying ϕ_∞?

e) A linear classifier can be expressed using only the inner products of support vectors in the transformed feature space. The Kernel trick, exploited by the SVM, is to define a function $K(\cdot, \cdot)$ such that

$$K(x, y) = \phi_\infty(x) \cdot \phi_\infty(y)$$

where the inner product between two infinite vectors $\mathbf{a} = (a_1, a_2, \ldots)$ and $\mathbf{c} = (c_1, c_2, \ldots)$ is defined as

$$\mathbf{a} \cdot \mathbf{c} = \sum_{i=1}^{\infty} a_i\, b_i$$

Given the definition of ϕ_∞ compute the form of $K(x, y)$. Hint you may want to use the Taylor series expansion of e^x:

$$e^x = \lim_{n\to\infty} \sum_{j=0}^{n} \frac{x^j}{j!}$$

f) With such a high dimensional feature space should we be concerned about over-fitting?

Solution:

Remember Given a set of $n+1$ data points (x_i, y_i) where no two x_i are the same, then one can always fit a polynomial of degree n

$$f(x) = w_0 + w_1 x + w_2 x^2 + \cdots + w_n x^n$$

s.t.

$$\boxed{f(x_i)} = w_0 + w_1 x_i + w_2 x_i^2 + \cdots + w_n x_i^n = \boxed{y_i}$$

for $i = 1, 2, \ldots, n+1$.

a) Draw a case where you have $m = 3$ training examples which are not linearly separable.

b) **Case I: 3 points** We have 3 points x_1, x_2, x_3 with labels y_1, y_2, y_3 and want to find a $\mathbf{w}, w_0$ such that for $i = 1, 2, 3$:

$$\mathbf{w}^T \phi_2(x_i) + w_0 > 0 \quad \text{or} \quad \mathbf{w}^T \phi_2(x_i) + w_0 < 0$$

depending on the y_i's. Now

$$\phi_2(x_i) = e^{-\frac{x_i^2}{2}} (1, x_i, \tfrac{x_i^2}{\sqrt{2}}) \quad \text{for } i = 1, 2, 3$$

From the polynomial result know that there exists (w_0', w_1', w_2') such that

$$w_0' + w_1'\, x_i + w_2'\, x_i^2 = y_i \quad \text{for } i = 1, 2, 3$$

This implies that for $i = 1, 2, 3$:

$$\begin{aligned}
e^{-\frac{x_i^2}{2}} (w_0' + w_1'\, x_i + w_2'\, x_i^2) &= e^{-\frac{x_i^2}{2}}\, y_i \\
\implies e^{-\frac{x_i^2}{2}} (w_0', w_1', \sqrt{2}\, w_2') \begin{pmatrix} 1 \\ x_i \\ \frac{x_i^2}{\sqrt{2}} \end{pmatrix} &= e^{-\frac{x_i^2}{2}}\, y_i \\
\implies (w_0', w_1', \sqrt{2}\, w_2')\, \phi_2(\mathbf{x}_i) &= e^{-\frac{x_i^2}{2}}\, y_i \\
\implies \mathbf{w}^T\, \phi_2(\mathbf{x}_i) &= e^{-\frac{x_i^2}{2}}\, y_i, \qquad \text{where } \mathbf{w}^T = (w_0', w_1', \sqrt{2}\, w_2')
\end{aligned}$$

As $e^{-\frac{x_i^2}{2}} > 0$ this implies:

$$\mathbf{w}^T\, \phi_2(\mathbf{x}_i) > 0 \text{ if } y_i = 1 \quad \textbf{and} \quad \mathbf{w}^T\, \phi_2(\mathbf{x}_i) < 0 \text{ if } y_i = -1$$

Thus $\mathbf{w}^T = (w'_0, w'_1, \sqrt{2}\, w'_2)$ and $w_0 = 0$ is a hyperplane which linearly separates the points $\phi_2(x_1), \phi_2(x_2), \phi_2(x_3)$. Thus the points $\phi_2(x_1), \phi_2(x_2), \phi_2(x_3)$ are linearly separable.

Case II: (n+1) points Basically the same argument works for $n+1$ points and $\phi_n(\cdot)$. Can construct a polynomial of degree n such that

$$w'_0 + w'_1\, x_i + w'_2\, x_i^2 + \cdots + w'_n\, x_i^n = y_i \;\text{ for } i = 1, 2, \ldots, n+1$$

From this one can define a hyperplane such that $\mathbf{w}, w_0$ such that for $i = 1, \ldots, n+1$

$$\mathbf{w}^T \phi_n(x_i) + w_0 > 0 \;\text{ if } y_i = 1 \quad \textbf{and} \quad \mathbf{w}^T \phi_n(x_i) + w_0 < 0 \;\text{ if } y_i = -1$$

c) No.

d) No. Given m points one can project them into an m dimensional space via $\phi_{m-1}(\cdot)$ and in this space the points are linearly separable. So obviously any finite set of points, without duplicates, can be separated by $\phi_\infty(\cdot)$.

e) If we substitute in the expressions for ϕ_∞ and also use the Taylor series expansion for e^x we can show that:

$$\begin{aligned}
K(x, y) &= \phi_\infty(x) \cdot \phi_\infty(y) \\
&= \sum_{j=0}^{\infty} \frac{x^j}{\sqrt{j!}}\, e^{-\frac{x^2}{2}}\, \frac{y^j}{\sqrt{j!}}\, e^{-\frac{y^2}{2}} \\
&= e^{-\frac{(x^2+y^2)}{2}} \sum_{j=0}^{\infty} \frac{x^j}{\sqrt{j!}} \frac{y^j}{\sqrt{j!}} \\
&= e^{-\frac{(x^2+y^2)}{2}} \sum_{j=0}^{\infty} \frac{(x\, y)^j}{j!} \\
&= e^{-\frac{(x^2+y^2)}{2}}\, e^{xy} = e^{-\frac{(x-y)^2}{2}}
\end{aligned}$$

f) If we are using an SVM to find the separating hyperplane then the practice of finding the one with largest margin should protect us from overfitting. However, there is of course the trade-off to be made between finding a hyperplane which correctly classifies the training data and having the width of the margin. This is controlled by the value C in the notes and cross-validation should be used to find a value of C which defines a good trade-off.

Exercises 15: *EM**

We have two coins. The first is a fair coin while the second is not necessarily fair. In summary:

$$P(H|\text{coin }1) = \frac{1}{2} \quad P(H|\text{coin }2) = \alpha$$

This procedure is as follows:

> Coin 1 is tossed. If this results in a head then coin 1 is tossed again otherwise coin 2 is tossed.

a) What is the probability that the 2nd toss results in a head ?

b) The above process is repeated N independent times and n_2 times a head is obtained on the 2nd toss. What is the maximum likelihood estimate for α ?

c) Say we're told that the process was repeated N times and in total M heads were obtained (this includes the first and second toss). What two update equations can we repeatedly apply to obtain an estimate for α ?

Solution:

a)

$$\begin{aligned} P(\text{2nd coin toss is a head}) &= P(HH) + P(TH) \\ &= P(H|\text{coin }1)\,P(H|\text{coin }1) + P(T|\text{coin }1)\,P(H|\text{coin }2) \\ &= \frac{1}{2}\times\frac{1}{2} + \frac{1}{2}\times\alpha = \frac{1}{4} + \frac{\alpha}{2} \end{aligned}$$

b)

$$P(\text{2nd coin toss results in } n_2 \text{ heads and } N - n_2 \text{ tails}) = K\left(\frac{1}{4}+\frac{\alpha}{2}\right)^{n_2}\left(\frac{3}{4}-\frac{\alpha}{2}\right)^{N-n_2}$$

$$\log p_1 = \log K + n_2 \log\left(\frac{1}{4}+\frac{\alpha}{2}\right) + (N-n_2)\log\left(\frac{3}{4}-\frac{\alpha}{2}\right)$$

Want to maximize this value wrt α. So

$$\begin{aligned} \frac{\partial \log p_1}{\partial \alpha} =& n_2\frac{2}{1+2\alpha} - (N-n_2)\frac{2}{3-2\alpha} = 0 \\ \implies& \frac{n_2}{1+2\alpha} = \frac{N-n_2}{3-2\alpha} \\ \implies& 4n_2 - N = 2N\alpha \\ \implies& \hat{\alpha} = \frac{4n_2 - N}{2N} \end{aligned}$$

c) The hidden data is n_2 the number of times the second toss was a head. Let n_1 be the number of times the first toss was a head. So $M = n_1 + n_2$.

If we have an estimate $\hat{n}_2$ for n_2 then from part **b** of this question we know we can estimate α by:

$$\hat{\alpha} = \frac{4\hat{n}_2 - N}{2N}$$

Say we have an estimate $\hat{\alpha}$ for α, then from this and M we want to estimate n_2. Given $\hat{\alpha}$ and M the expected value of n_2 is

$$\hat{n}_2 = \frac{\frac{1}{4} + \frac{\hat{\alpha}}{2}}{\frac{3}{4} + \frac{\hat{\alpha}}{2}} M = \frac{1 + 2\hat{\alpha}}{3 + 2\hat{\alpha}} M$$

as the ratio $n_1 : n_2$ should be the same as $\frac{1}{2} : (\frac{1}{4} + \frac{\alpha}{2})$.

α can then be estimated as follows. Initialize by setting $\alpha^{(0)} = \frac{1}{2}$. Then iterate between

$$n_2^{(t)} = \frac{1 + 2\alpha^{(t)}}{3 + 2\alpha^{(t)}} M$$

and

$$\alpha^{(t+1)} = \frac{4n_2^{(t)} - N}{2N}$$

until convergence.