CS-419m: Practice question set 6 (Bagging, Boosting, SVMs)

1. Consider two different methods of training a binary SVM:

$$\begin{aligned} &\min_{\mathbf{w},\xi} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N}\xi_i \\ &\text{s.t. } y^i(\mathbf{w}.\mathbf{x}^i + w_0) \geq 1 - \xi_i \quad \forall i: 1\dots N \\ &\xi_i \geq 0 \quad i: 1\dots N \end{aligned} \tag{1}$$

and

$$\begin{aligned} &\min_{\mathbf{w},\xi} \frac{1}{2}(||\mathbf{w}||^2 + w_0^2) + C\sum_{i=1}^{N}\xi_i \\ &\text{s.t. } y^i(\mathbf{w}.\mathbf{x}^i + w_0) \geq 1 - \xi_i \quad \forall i: 1\dots N \\ &\xi_i \geq 0 \quad i: 1\dots N \end{aligned} \tag{2}$$

(a) For objective 1 write the value of w_0 in terms of its dual variables $\alpha_1, \dots, \alpha_N$.
`Choose a support vector by picking any` i `such that` $0 < \alpha_i < c$. `For such` i, $\xi_i = 0$ `(KKT Conditions), and` $y^i(\mathbf{w}.\mathbf{x}^i + w_0) = 1$. `Hence,`

$$w_0 = \frac{1}{y^i} - \mathbf{w}.\mathbf{x}^i$$

..1

(b) For objective 2 write the value of w_0 in terms of its dual variables $\alpha_1, \dots, \alpha_N$.
w_0 `behaves like just another feature weight for a feature with constant value 1. Hence,`

$$w_0 = \sum_{i=1}^{N} \alpha_i \cdot y_i \cdot 1 = \sum_{i=1}^{N} \alpha_i \cdot y_i$$

..2

(c) Suppose we translate the training set $D = \{(\mathbf{x}^i, y_i) : i = 1, \dots, N\}$ by using a scalar λ so that each $\mathbf{x}^i = (x_1^i, \dots, x_d^i)^T$ is replaced with $(x_1^i + \lambda, \dots, x_d^i + \lambda)^T$. Call this new dataset D_λ. If we train both objectives using D_λ, for which of the two objectives will the optimal value of $\mathbf{w}$ remain the same as with data D. Justify. (No marks will be awarded for answers without proper justification.)
`For the first objective, optimal value of` $\mathbf{w}$ `will remain unchanged because we can just add` $-\lambda \cdot \sum_{j=1}^{N} w_j$ `to` w_0 `and keep the constraints and objective function unchanged.` ..4

(d) For the objective that you answered above, write the new optimal value of w_0 in terms of the optimal value w_0^* with D.

$$w_0 = w_0^* - \lambda \cdot \sum_{j=1}^{N} w_j$$

..1

(e) For the objective for which the optimal value of $\mathbf{w}$ changes as you move from D to D_λ, write the dual in terms of $\mathbf{x}^i, y_i, \lambda, \alpha_i, C$. You can assume a linear Kernel.

$$\max_{0<=\alpha<=c, \sum_{i=1}^{N}\alpha_i . y_i = 1} \left\{ \sum \alpha_i - \frac{1}{2} * \left(\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i.\alpha_j.y_i.y_j. < \mathbf{x}_i + \lambda, \mathbf{x}_j + \lambda > \right) \right\}$$

(f) Design a small dataset D and λ where the solution of w in the two objectives is the same on D but differs on D_λ.

Assume d=1 and c is large. $D = (x^1, y_1), (x^2, y_2) = ([-1], -1), ([1], 1)$ For both objectives, $w_0 = 0; w_1 = 1$.

Now, pick $\lambda = 100$.

In case of objective 1, w shifts as follows $w_0 = -100; w_1 = 1$

In case of objective 2, both w_0 and w_1 shift. For example, $w_1 = \frac{1}{100}$ and w_0 accordingly (depending on c). ..3

2. Consider a two class training dataset D with $d = 1$ attributes where a fraction ϵ of the points have x value -2 and are labeled positive, a fraction $(1 - \epsilon)/2$ are uniformly distributed between $0 and a$ where $\alpha > 0$ and are labeled negative, and the remaining $(1 - \epsilon)/2$ are uniformly distributed between α and 2α and are labeled positive. Suppose we run boosting on this dataset where each stage is a decision tree classifier restricted to have a single node.

   (a) Draw the decision tree of the first three stages where the split condition is specified in terms of α and ϵ. Write down also the weight of each tree. ..4

   Call instances at -2, group A instances, negative instances group B, and rest group C. Assume $(1 - \epsilon)/2 > \epsilon$

   - First tree: weight of all instances 1, split at $x_1 < \alpha$, group A instances misclassified, error = ϵ.,
   - Second tree: weight of group A = $(1 - \epsilon)/\epsilon$.
     Split at $x_1 < 0$.
     Group C misclassified. Error = $\frac{(1-\epsilon)/2}{(1-\epsilon)+\epsilon*(1-\epsilon)/\epsilon} = 1/4$.
     (In this case, a second tree with the same error is a trivial tree that marks everything as positive)
   - Third tree: weight of group A = $(1-\epsilon)/\epsilon$. weight of group B = 1. weight of group C = (1-1/4)/(1/4) = 3.
     Sum of weight of group A and group C = $(1-\epsilon)+3(1-\epsilon)/2$ is greater than the weight of the negatives in group B. Therefore, third tree will mark everything as positive.
     Group B misclassified. Error = $\frac{(1-\epsilon)/2}{(1-\epsilon)/2+5*(1-\epsilon)/2}$=1/6.

   (b) Estimate the number of stages of the boosting algorithm in terms of α and ϵ. ..4

   The dataset is such that it is always possible to construct a single node tree where the error is less than 1/2. This is because of the three groups: A,B and C exactly one of them will be incorrectly labeled by an optimal tree with any weighting of the groups. If all groups have weight less than 1/2, the error of the tree will be less than 1/2. If any group has weight more than 1/2, you can always construct a tree to correctly classify that group as long as $\alpha > 0$.

   Thus, the boosting algorithm will never terminate.

3. Suppose we use Bagging to generate a set of B independent regression functions $f_1(\mathbf{x}), \dots, f_B(\mathbf{x})$. The prediction of the bagged regression function $F(\mathbf{x}) = \frac{1}{B}\sum_j f_j(\mathbf{x})$.

   (a) If σ^2 denotes the expected square error $E((y - f_j(\mathbf{x}))^2)$ of each function $f_j(\mathbf{x})$, what is the square error of $F(\mathbf{x})$ assuming that the errors of different classifiers are independent of each other. ..3

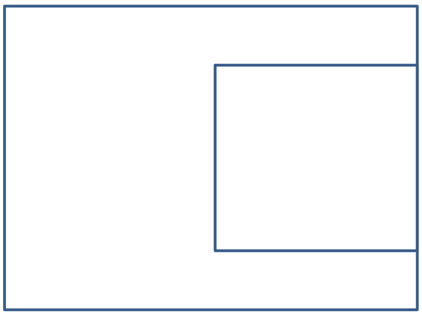
   $E((y - F(\mathbf{x}))^2) = ((y - \frac{1}{B}\sum_j f_j(\mathbf{x}))^2) = E(\frac{1}{B^2}(\sum_j (y - f_j(\mathbf{x}))^2 - 2\sum_{j,k}(y - f_j(\mathbf{x}))(y - f_k(\mathbf{x}))))$

   The first term in the expectation is $\frac{\sigma^2}{B}$. The second term equals zero because the different bags are independent of each other and have zero correlation.

(b) Suppose in the training set there is a severe outlier $(\mathbf{x}^r, y_y)$ that causes all functions that includes it to make correlated mistakes. What is the probability δ that a given component f_j will encounter this outlier in its training set? (Recall that the training set of f_j will be obtained by sampling with replacement N instances randomly from the original training set D)? ..3 `The probability` δ `that the instance will be included in the sample of bag b is` $1-(1-\frac{1}{N})^N$

(c) Let S be the set of component functions that include the outlier in their training sample. All functions in S have a higher square error of β^2. Also, there is a correlation c in the errors made by any two component functions from the set S. For other pairs of functions there is no correlation. Now, what is the expected error of $F(\mathbf{x})$. ..3

4. Consider this dataset with $d = 2$ attributes and two classes. Each point in this figure is an instance where the value of its two attributes are its coordinates in this 2D space. For example instance $\mathbf{x}^1$ is this figure is $[0.5 \;\; 0.8]$. All points within the inner rectangle are from class $y = 1$ and the points outside are from $y = -1$. For these questions, assume that there are lots of instances so that the space is uniformly and densely packed.



(a) Draw the smallest decision tree for these instances. ..2

(b) Write the equation of the best SVM classifier with a linear kernel (Assume C is large). ..2

(c) Write the equation of the best SVM classifier with a quadratic Kernel and show geometrically the boundary (Assume C is large). ..3

(d) The predicted class using k nearest neighbor classifier with Euclidean distance for the point $\mathbf{x}^1 = [0.5 \;\; 0.8]$ and $k = 10$ ..2

(e) How does the decision boundary of the nearest neighbor classifier differ from the boundary of the inner rectangle in the diagram? ..2

(f) Write the following parameters of the naive Bayes classifier that uses a Gaussian distribution along each attribute.

i. μ of attribute x_1, class $y = 1$, ..1

ii. σ of attribute x_1, class $y = 1$, ..2

iii. μ of attribute x_2, class $y = 1$, ..1

iv. μ of attribute x_1, class $y = -1$, ..1

v. μ of attribute x_2, class $y = -1$, ..1

vi. Class prior $\Pr(y = 1)$ ..1

(g) If you run boosting on these points with each stage restricted to be a decision tree of one node, draw the nodes of successive boosting stages. ..4

5. Suppose the true class label of a dataset with two real attributes x_1 and x_2 ($d = 2$), and $k = 2$ is generated using the following function $y = \text{sign}(x_1^2 + 9x_2^2 - 25)$. Assume you have lots of training examples uniformly distributed within the Euclidean space. Which of the family of classifiers below can perfectly classify this dataset.

   (a) Naive Bayes classifier: State the family from which the two distributions $\Pr(x_1|y), \Pr(x_2|y)$ have to be drawn to correctly classify this dataset. Specify the estimated maximum likelihood values of the parameters of your chosen family. ..5 `Gaussian distribution for both` $\Pr(x_1|y), \Pr(x_2|y)$.

   `The maximum likelihood parameter values are as follows: (Note the ML estimates may not correctly classify this dataset, but some other parameters will).`

   $\mu_{jc} = 0$ `for` $j = 1, 2$ `class` $c = +1, -1$.

   `The variance for class +1 is infinite in both classes because the Euclidean space is unbounded.`

   `The variance for class -1 is the variance of the uniform distribution U(-5,5) for attribute 1 and U(-5/3,5/3) for attribute 2. The variance of a uniform distribution` $U(a,b) = (a+b)^2/12$

   (b) LDA classifier: Justify. ..1 `No, since it can only support linear decision boundaries.`

   (c) SVM classifier with kernel. If yes, list the kernel types for which perfect classification is possible. ..2 `Any polynomial kernel of degree` ≥ 2.

   (d) For one of the kernels K above, provide an embedding $\phi(\mathbf{x})$ of $\mathbf{x} = (x_1, x_2)$ such that $K(\mathbf{x}^i, \mathbf{x}^j) = \phi(\mathbf{x}^i).\phi(\mathbf{x}^j)$. ..2 $K(\mathbf{x}^i, \mathbf{x}^j) = (\mathbf{x}^i.\mathbf{x}^j)^2$ $\phi(\mathbf{x}) = [x_1^2, x_2^2, \sqrt{2}x_1x_2]$

   (e) Boosting with a base classifier that can be any arbitrary classifier but on exactly one of the two attributes at a time. Justify your answer. ..3 `If base classifier is SVM with degree 2 polynomial kernel, then the first stage could be sign`$(x_1^2 - 25)$. `This is the best one can do with a single attribute. Assuming you are using Adaboost, this will increase the weight of instances in the` $x_1 \in [-5, 5]$ `and` $x_2 \notin [-5/3, 5/3]$ `so that the next classifier will be sign`$(9x_2^2 - 25)$. `However, it is unclear if one can get perfect separation by adding more boosting stages.`