EECS 553 Homework 3 Solution (FA24)

1. **OSM Hyperplane Classifier (3 points each)**

   (a) Grading rubrics

   - 3 pts for fully correct answer.
   - 1 pt for pointing out $\xi_i \geq 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)$.
   - 1 pt for stating $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \leq 0$ when $\boldsymbol{x}_i$ is misclassified.

   For any feasible points, $\xi_i \geq 0$ and $\xi_i \geq 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)$. So $\xi_i^* \geq \max(0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b))$. When $\boldsymbol{x}_i$ is misclassified, $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \leq 0$, making $\xi_i^* \geq \max(0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)) \geq 1$.

   (b) Grading rubrics

   - 3 pts for fully correct answer.
   - 1.5 pt for pointing out $\xi_i^* \geq \max(0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b))$.

   Recall $\xi_i^* \geq \max(0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b))$. At the optimum, equality must be achieved in the inequality $\xi_i \geq \max(0, 1 - y_i(w^T x_i + b))$, otherwise the objective could be reduced. Let $y_i = 1$, then $0 < 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) = 1 - (\boldsymbol{w}^T\boldsymbol{x}_i + b) = |\boldsymbol{w}^T\boldsymbol{x}_i + b - y_i|$. Likewise, if $y_i = 1$, then $0 < 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) = 1 + (\boldsymbol{w}^T\boldsymbol{x}_i + b) = |\boldsymbol{w}^T\boldsymbol{x}_i + b - y_i|$. $\xi_i^* > 0 \implies \xi_i^* = 1 - y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) = ||\boldsymbol{w}||_2 \frac{|\boldsymbol{w}^T\boldsymbol{x}_i + b - y_i|}{||\boldsymbol{w}||_2}$.

2. **Optimal soft-margin hyperplane**

   Grading rubrics

   - 1 pt for combining the constraints into $\xi_i \geq \max(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b))$
   - 1 pt for arguing that the optimal $\xi_i^* = \max(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b))$ (otherwise we could reduce the objective value without violating the constraints)
   - 1 pt for substituting $\xi_i^* = \max(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b))$ in the objective, eliminating the constraints, and recognizing the cost-sensitive optimal soft-margin hyperplane corresponds to doing ERM with a *weighted* hinge loss. As long as the weights are correct, it does not matter how the student wrote the weights.
   - 0 pt if no effort or completely wrong.

   To get some intuition for this problem and the properties of the optimal classifier, consider the following.

   For both $y_i = 1$ and $y_i = -1$, if $\mathbf{x}_i$ is misclassified, then

   $$y_i(\mathbf{w}^{*T}\mathbf{x}_i + b^*) \leq 0$$

   Combined with the constraint $y_i(\mathbf{w}^{*T}\mathbf{x}_i + b^*) \geq 1 - \xi_i^*$, we have

   $$1 - \xi_i^* \leq y_i(\mathbf{w}^{*T}\mathbf{x}_i + b^*) \leq 0,$$

   which implies $\xi_i^* \geq 1$ for all misclassified $\boldsymbol{x}_i$.

Therefore,

$$\frac{1-\alpha}{n}\sum_{i:y_i=1}\boldsymbol{\xi}^* \geq (1-\alpha)\frac{\text{number of false negatives}}{n} = \frac{(1-\alpha)n}{\sum_{i=1}^n \mathbf{1}_{\{y_i=-1\}}}\cdot\text{FNR} = \frac{1-\alpha}{\hat{\pi}_{-1}}\cdot\text{FNR},$$

where $\hat{\pi}_{-1}$ is the sample proportion of class $y=-1$, and FNR stands for False Negative Rate. To see the second step above note that FNR= $\frac{\text{number of false negatives}}{\text{number of training data where } y=-1}$ Similarly,

$$\frac{\alpha}{n}\sum_{i:y_i=-1}\boldsymbol{\xi}^* \geq \alpha\frac{\text{number of false positives}}{n} = \frac{\alpha n}{\sum_{i=1}^n \mathbf{1}_{\{y_i=1\}}}\cdot\text{FPR} = \frac{\alpha}{\hat{\pi}_1}\cdot\text{FPR},$$

where $\hat{\pi}_1$ is the sample proportion of class $y=1$, and FPR stands for False Positive Rate.

I've connected the above to the FNR and FPR, which are the statistics typically considered to evaluate cost-sensitive classifiers. However, it is enough to know that these terms are weighted upper bounds on the number of false positives and false negatives. The above is just for intuition and is not required.

We follow the same approach from the lecture notes for the standard soft-margin hyperplane. First, notice that we can divide by C without changing the optimal solution. Now we consider the constraints.

The first constraint $y_i(\mathbf{w}^T\mathbf{x}_i+b)\geq 1-\xi_i$ can be rewritten as $\xi_i \geq 1-y_i(\mathbf{w}^T\mathbf{x}_i+b)$. The second constraint is $\xi_i\geq 0$. These are just two lower bounds for ξ_i that must *both* be satisfied. If we want to satisfy two lower bounds simultaneously, we can just pick the larger lower bound and make sure it is satisfied. Therefore, the two constraints can be reduced to a single constraint

$$\xi_i \geq \max(0, 1-y_i(\mathbf{w}^T\mathbf{x}_i+b)),$$

and we can equivalently write the optimization problem

$$\begin{aligned}\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{n}[(1-\alpha)\sum_{i:y_i=1}\xi_i + \alpha\sum_{i:y_i=-1}\xi_i] \\ \text{s.t. } \xi_i \geq \max(0, 1-y_i(\mathbf{w}^T\mathbf{x}_i+b)), \quad \forall i\end{aligned}$$

where $\lambda = \frac{1}{C}$. If the constraint is not satisfied with equality, the objective can always be decreased. Therefore, it must be the case that

$$\xi_i^* = \max(0, 1-y_i((\boldsymbol{w}^*)^T\mathbf{x}_i+b^*)), \quad \forall i.$$

So we can equivalently write the optimization problem

$$\begin{aligned}\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{n}[(1-\alpha)\sum_{i:y_i=1}\xi_i + \alpha\sum_{i:y_i=-1}\xi_i] \\ \text{s.t. } \xi_i = \max(0, 1-y_i(\mathbf{w}^T\mathbf{x}_i+b)), \quad \forall i\end{aligned}$$

However, we now have an explicit formula for ξ_i through the equality constraint. We can just plug the formula into the objective and eliminate the constraint. Doing this yields

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{n}[(1-\alpha)\sum_{i:y_i=1}\max(0, 1-y_i(\mathbf{w}^T\mathbf{x}_i+b)) + \alpha\sum_{i:y_i=-1}\max(0, 1-y_i(\mathbf{w}^T\mathbf{x}_i+b))].$$

We can simplify the above to make it more concise by writing

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{n}\sum_{i=1}^{n} c_i \max(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b)),$$

where $c_i = \begin{cases} 1-\alpha, & y_i = 1 \\ \alpha, & y_i = -1 \end{cases}$, or we can even be a bit more slick by noting that an equivalent form for c_i is $c_i = (1-\alpha)\mathbb{1}_{y_i=1} + \alpha\mathbb{1}_{y_i=-1}$. This yields

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{n}\sum_{i=1}^{n}((1-\alpha)\mathbb{1}_{y_i=1} + \alpha\mathbb{1}_{y_i=-1}) \max(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b)).$$

In the end, the loss we have derived is a weighted hinge loss of the form

$$L(\alpha, y_i, t) = ((1-\alpha)\mathbb{1}_{y_i=1} + \alpha\mathbb{1}_{y_i=-1}) \max(0, 1 - y_i t),$$

and our regularization parameter is $\lambda = \frac{1}{C}$.

3. **Naïve Bayes for Document Classification: Cars or Motorcycles?** (3 points each)

   (a) **Grading Rubrics**

   1. 3 points for any answer related to the independence of the word occurrence
   2. 0 if no effort

   Answer: the additional assumption is that the occurrence of each word in a document is independent. This is stronger than Naive Bayes which only requires *features* to be independent (conditioned on class).

   Note that our features x_j are the number of times word j appears in a given document. When we compute $(p_{kj})^{x_j}$ we are treating each occurrence of the word j as an independent event with likelihood p_{kj}. Then $P(X_j = \ell | Y = k) = P(X_j = 1 | Y = k)^\ell = (p_{kj})^\ell$.

   (b) **Grading Rubrics**

   1. 3 points if fully correct
   2. 2 points if minor errors (e.g., did not substitute the expression for p_{kj}.)
   3. 1 point if mostly wrong (e.g., wrong log operations.)
   4. 0 if no effort

   Given

$$\hat{y_i} = \arg\max_{k\in\{0,1\}} \log\left(\pi_k \Pi_{j=1}^{d} {p_{kj}}^{x_{ij}}\right)$$

   Distribute the log:

$$\log\left(\pi_k \Pi_{j=1}^{d} {p_{kj}}^{x_{ij}}\right) = \log \pi_k + \sum_{j=1}^{d} x_{ij} \log p_{kj}$$

   Substituting the definition of p_{kj} gives:

$$\log\left(\pi_k \Pi_{j=1}^{d} {p_{kj}}^{x_{ij}}\right) = \log \pi_k + \sum_{j=1}^{d} x_{ij} \left(\log(n_{kj} + \alpha) - \log(n_k + \alpha d)\right)$$

Also correct: the definition of $\pi_k = n_k/n$ may be substituted

$$\log \left(\pi_k \Pi_{j=1}^{d} {p_{kj}}^{x_{ij}} \right) = \log n_k - \log n + \sum_{j=1}^{d} x_{ij} \left(\log(n_{kj} + \alpha) - \log(n_k + \alpha d) \right)$$

Optional: the above can be further simplified with vector notation, resulting in a linear classifier of the form

$$\hat{y_i} = \arg \max_{k \in \{0,1\}} b_k + \mathbf{w}_k^T x_i$$

Where $b_k = \log \pi_k$, and $w_{kj} = \log p_{kj} = \log(n_{kj} + \alpha) - \log(n_k + \alpha d)$

(c) **Grading Rubrics**

1. 1.5 points for each correct prior probability or log prior probability.
2. 0 if no effort or wrong probabilities/estimates

$\hat{\pi_0} = 0.4983$, and $\hat{\pi_1} = 0.5017$
$\log \hat{\pi_0} = -0.6966$, and $\log \hat{\pi_1} = -0.6898$

(d) **Grading Rubrics**

1. 3 points for correct test error or accuracy (give full credits for answers within ±0.5%)
2. 0 if no effort or wrong numbers

The correct test error is 12.5945%, (or an accuracy of 87.41%).

(e) **Grading Rubrics**

1. 3 points for correct test error or accuracy based on majority vote
2. 0 if no effort or wrong numbers

The correct majority-vote predictor always chooses class 1 over class 0, resulting in a test error of 49.8741%, (or an accuracy of 50.13%).

(Note that the answer here is *not* $\hat{\pi_0}$, as $\hat{\pi_0}$ is computed on the training data. The answer 49.8741% is equivalent to the estimate of the class 0 prior on the test data).

(f) **Grading Rubrics**

1. 3 points for submitting code (as long as students are not submitting only print functions full credits will be given)
2. 0 if no effort