10-701 Machine Learning, Fall 2012: Homework 2 Solutions

1 Learning with L1 norm

Suppose you want to predict an unknown value $Y \in \mathbb{R}$, but you are only given a sequence of noisy observations $x_1, \ldots, x_n$ of Y with iid noise ($x_i = Y + \epsilon_i$).

We have seen in the last homework, that if we assume the noise is i.i.d. Gaussian($\epsilon_i \sim N(0, \sigma^2)$), finding the maximum likelihood estimate for Y is equivalent to finding the value $\hat{y}$ which minimizes the sum of the least square errors to the x's. That is to say

$$\hat{y} = \arg\min_y \sum_{i=1}^{n} (y - x_i)^2 \tag{1}$$

And there is a simple closed form solution:

$$\hat{y} = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{2}$$

It was also suggested that if we assume that the noise is i.i.d. Laplace ($\epsilon_i \sim \mathit{Laplace}(0, b)$) with pdf

$$f_{\epsilon_i}(x) = \frac{1}{2b} \exp\left(-\frac{|x|}{b}\right) \tag{3}$$

we end up with a maximum likelihood estimator that is in some sense more robust. We will show this more rigorously in this part.

1 [2 pts] Begin by showing that finding the MLE for Y, assuming Laplace noise, is equivalent to finding the value $\hat{y}$ that minimizes the sum of absolute error. That is

$$L(y) = \sum_{i=1}^{n} |y - x_i| \tag{4}$$

$$\hat{y} = \arg\min_y L(y) \tag{5}$$

Solution: Likelihood of Y is

$$P(X|Y) = \prod_{i=1}^{n} \left(-\frac{-|x_i - Y|}{b}\right) \tag{6}$$

$$\log(P(X|Y)) \propto \sum_{i}^{n} (-|x_i - Y|) \tag{7}$$

Maximizing $\log(P(X|Y))$ is equivalent to minimizing $-\log(P(X|Y))$ which gives me

$$L(y) = \sum_i^n |x_i - y|$$

$$\hat{y} = \arg\min_y L(y)$$

2 [4 pts] A standard way to minimize a loss function is to take the derivative and set it to zero. This loss function is not directly differentiable. However, it is easy to see that the function is not differentiable only where y has same value as any of the x's.

Assume that the x's are distinct and are sorted in ascending order ($\forall i, \forall j > i, x_j > x_i$).

Find an expression for the gradient $\frac{dL(y)}{dy}$ under the constraint that y lies between two consecutive values of x (That is to say $x_i < y < x_{i+1}$). (**Hint:** You may have to consider x's which are $> y$ separately from the x's which are $< y$)

Solution: Let X^+ be the values of x which are $> y$ and let X^- be the values of x which are $< y$. Then

$$\begin{aligned}
\frac{dL(y)}{dy} &= \frac{d\sum_i^n |x_i - y|}{dy} && (8)\\
&= \sum_i^{|X^+|} \frac{d|x_i^+ - y|}{dy} + \sum_i^{|X^-|} \frac{d|x_i^- - y|}{dy} && (9)\\
&= \sum_i^{|X^+|} \frac{d(x_i^+ - y)}{dy} + \sum_i^{|X^-|} \frac{d(y - x_i^-)}{dy} && (10)\\
&= |X^-| - |X^+| && (11)
\end{aligned}$$

In words the derivative is (the number of X's smaller than y) - (the number of X's larger than y).

3 [3 pts] Assuming that there are an even number of x's, what are all the values of y for which $\frac{dL(y)}{dy} = 0$?

Solution: The derivative is 0 when (the number of X's smaller than y) = (the number of X's larger than y). Let X_i and X_{i+1} be the middle 2 elements of X. Then when $X_i < y < X_{i+1}$ the derivative is 0.

4 [3 pts] If we have an odd number of x's, there is no value of y where $\frac{dL(y)}{dy} = 0$. However there is a y_0 such that $\frac{dL(y)}{dy} < 0$ if $y < y_0$ and $\frac{dL(y)}{dy} > 0$ if $y > y_0$. What is y_0?

Solution: Let X_i be the middle element of X. Then $y_0 = X_i$. Simply, if $y < y_0$, I will have $|X_+| > |X_-|$ and $\frac{dL(y)}{dy} < 0$ and if $y > y_0$, I will have $|X_+| < |X_-$ and $\frac{dL(y)}{dy} > 0$.

5 [4 pts] Your answer in the last two parts are therefore the solution to $\hat{y}$. Give an explanation of what the solution represents (Hint: Its either mean, median or mode). Give a brief explanation why this solution may be more robust against outlier in the data (as compared to least square errors).

Solution: A really far outlier will result in the mean being skewed. For instance if $X = (1, 1, 1, 1, 1, 1, 1, 1, 1, 10)$ then using least square the prediction will be $19/10 = 1.9$ where as using absolute error, I will still have $y = 1$.

Now we will see the how L1 penalty leads to feature selection. You are given a set of points and the corresponding outputs: $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, where $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$ for $i = 1, \ldots, n$. You want to use this data to train a linear predictor $y = w^T F(x)$ where $w^T = (w_1, \ldots, w_K)$ and $F(x)^T = (f_1(x), \ldots, f_K(x))$, where K is finite. Here f_i is the ith feature for our learning problem. Consider the following objective function:

$$J(w, \lambda) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2}(y_i - w^T F(x_i))^2 + \lambda ||w||_1 \tag{12}$$

where $||w||_1 = \sum_{i=1}^{n} |w_i|$ ($||w||_1$ is called the L1 norm of the vector w). We use our data to learn the vector w by minimizing $J(w, \lambda)$, i.e.

$$w^* = \arg\min_{w \in \mathbb{R}^k} J(w, \lambda) \tag{13}$$

The above optimization criterion typically leads to an effective feature selection by picking a large value for parameter λ. In other words if we pick a large value of λ many w_is will be 0. Therefore the corresponding feature function will be unimportant for our predictor.

6 [2 pts] Write the derivative of the first term ($\frac{1}{n} \sum_{i=1}^{n} \frac{1}{2}(y_i - w^T F(x_i))^2$) of the function wrt w_k as $a_k w_k - r_k$. (ie find a_k and r_k).

Solution:

$$\begin{aligned}
\text{Summation term} &= \frac{1}{2n} \sum_{p=1}^{n} (y_p - w^T F(x_p))^2 \\
&= \frac{1}{2n} \sum_{p=1}^{n} (y_p - \sum_{j=1}^{k} w_j f_j(x_p))^2 \\
&= \frac{1}{2n} \sum_{p=1}^{n} (y_p - \sum_{j=1, j \neq i}^{k} w_j f_j(x_p) - w_i f_i(x_p))^2 \\
&= \frac{1}{2n} \sum_{p=1}^{n} \left(w_i^2 f_i(x_p)^2 + 2(w_i f_i(x_p))(y_p - \sum_{j=1, j \neq i}^{k} w_j f_j(x_p)) + (y_p - \sum_{j=1, j \neq i}^{k} w_j f_j(x_p))^2 \right) \\
&= \frac{1}{2} a_i w_i^2 - r_i w_i + d_i
\end{aligned} \tag{14}$$

where $a_i = \frac{1}{n} \sum_{p=1}^{n} f_i(x_p)$, $r_i = \frac{p=1}{n} f_i(x_p)(y_p - \sum_{j=1, j \neq i}^{k} w_j f_j(x_p))$ and $d_i = (y_p - \sum_{j=1, j \neq i}^{k} w_j f_j(x_p))^2$. Thus the derivative wrt w_i will be equal to $a_i w_i - r_i$.

7 [7 pts] It can be shown that at optima the following conditions is satisfied:-

$$w_k^* = \begin{cases} \frac{r_i + \lambda}{a_i} & \text{if } r_k < -\lambda \\ 0 & \text{if } r_k \in [-\lambda, \lambda] \\ \frac{r_i - \lambda}{a_i} & \text{if } r_k > \lambda \end{cases} \tag{15}$$

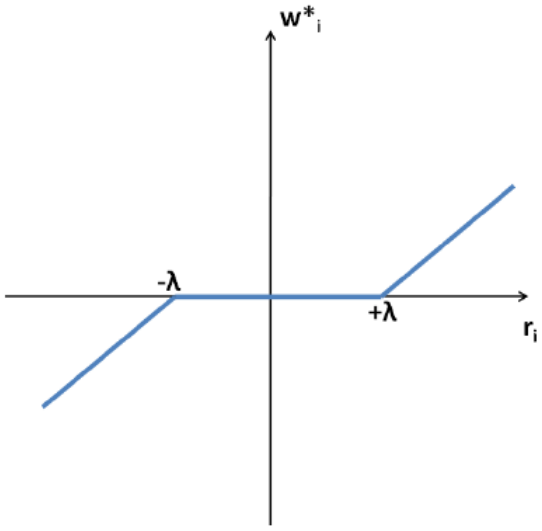


Figure 1: w_i^* vs r_i

Let's explore the relation between the regularization parameter λ, the weight of the kth parameter w_k and the quantity r_k What is the meaning of r_k (2 pt)? Provide a plot of w_i^* vs r_i. Where does λ appear in the plot (3 pt)? What can you say about when the value of w_k^* is zero (2 pts)?

Solution: The plot is shown in figure 1.

$$r_k = \frac{1}{n}\sum_{j=1}^{n} f_j(x_j)\left(y_j - \sum_{m=1,m\neq k}^{K} w_m f_m(x_j)\right) \tag{16}$$

This is the correlation between the i-th basis function and the prediction error when we use all basis function except the i-th one. The larger the r_i is the more "informative" it will be for our purposes. As we see from the plot λ is the regularization parameter that decides what we consider as a relevant / informative basis function/ feature. The plot also shows that when the correlation between the ith basis and the residual is not significant then the corresponding weight will be zero.

2 k Nearest Neighbor and Kernel Regression

2.1 (a)

$\mathbf{K}(x, x_i) = I(-k/2 \leq |x - x_i| < k/2)$

2.2 (b)

1. $w_i(x, x_i) = 1/k$

2. Kernel regression resembles n-NN weighted regression with weight function

$$w_i(x, x_i) = \frac{d^{-1}(x, x_i)}{\sum_{j=1}^{n} d^{-1}(x, x_j)}$$

where n is the number of data point.

2.3 (c)

Solution 1 (do not modify kernel regression):

Replace y with an equivalent vector of dummy variables. Note that a dummy variable is a variable taking 0 or 1, which can be written in the form $I(condition)$ where I is the indicator function. Suppose $y \in \{0, 1, ..., n-1\}$. Define $y_j^* = factor(y_j) = < I(y_j^{(1)} \neq 1), I(y_j^{(2)} \neq 1), ..., I(y_j^{(n-1)} \neq 1) >$ where y_j is the label of the j-th data point and y_j^* is the corresponding factorized label. Using kernel regression on x, we will end up with a vector $\hat{r}(x)$ of a sequence of values between [0, 1]. Find the first element that is less than 0.5 and assign its index to y; if all the elements in this vector are more than 0.5; assign 0 to y.

Solution 2 (change kernel regression into a voting function):

$\hat{r}(x) = \arg\max_y \sum_{i|x_i \in k-nn} w_i(x, x_i) I(y = y_i)$. This method is in fact equivalent to using dummy variables in someway.

One typical wrong answer is $y = round(\hat{r}(x))$. Consider the case where we only have data (x=0, y=0) and (x=1, y=2). If we have a new point x = 0.5, it should be categorized into 0 or 2 for 2-NN algorithm. But using $y = round(\hat{r}(x))$ will give us 1. Factorizing y is necessary since it is a quantitative variable with different meaning for different value.

2.4 (d)

Asking about training error and test error is equivalent to asking about training risk and true risk. Generally, a smoother kernel would generate more training risk; more bias and less variance in test. Thus, the correct order is

1. Training risk: $K2 < K1 < K3$
2. Bias: $K2 < K1 < K3$
3. Variance: $K3 < K1 < K2$
4. K2 is closer to 1-NN; K3 is closer to n-NN

3 Variance and Bias Tradeoff, Model Selection

In the following questions, assume zero Bayes error, i.e. zero noise variance.

1. **[12 points]** In class we define True Risk as Mean Squared Error between our model and the true model as:

$$R(f) = \mathbb{E}[(f(X) - Y)^2] \tag{17}$$

We also show this risk in terms of Bias-Variance Tradeoff, i.e:

$$R(f) = \mathbb{E}[(f(X) - Y)^2] = Variance + Bias^2 \tag{18}$$

where $Y = f^*(X)$ (the problem is assumed to be noise free).

We can also define risk in terms of our estimated parameter $\hat{\theta}$ (obtained using MLE or MAP or density estimator, etc) and the true parameter θ as:

$$R(\theta, \hat{\theta}) = \mathbb{E}_\theta(\hat{\theta} - \theta)^2 = Var_\theta(\hat{\theta}) + bias^2 \tag{19}$$

where $bias = \mathbb{E}_\theta[\hat{\theta}] - \theta$

Let $X_1, ..., X_n \sim Bernoulli(p)$, be our coin-flip example where each X_i is an independent flip where $X_i = 1$ indicates flipping a head, $X_i = 0$ indicates flipping a tail, and p is the probability of getting a head.

Consider two estimators for p, $\hat{p}_1 = \frac{1}{n}\sum_i X_i$ (the MLE estimate) and $\hat{p}_2 = \frac{\sum_i X_i + \alpha}{\alpha+\beta+n}$ (the mean of the posterior Beta distribution $P(p|D)$ when we use $Beta(\alpha, \beta)$ as prior).

(a) **[1 point]** Compute the risk of $\hat{p}_1$, i.e. $R(p, \hat{p}_1)$

Answer:

$$\begin{aligned}
R(p, \hat{p}_1) &= Var[\hat{p}_1] + (Bias[\hat{p}_1])^2 \\
&= Var[\frac{1}{n}\sum_i X_i] + (E[\frac{1}{n}\sum_i X_i] - p)^2 \\
&= \frac{1}{n^2} np(1-p) + (p-p)^2 \\
&= \frac{p(1-p)}{n}
\end{aligned}$$

(b) **[4 point]** Compute the risk of $\hat{p}_2$, i.e. $R(p, \hat{p}_2)$

Answer:

$$\begin{aligned}
R(p, \hat{p}_2) &= Var[\hat{p}_2] + (Bias[\hat{p}_2])^2 \\
&= Var[\frac{\alpha + \sum X_i}{\alpha+\beta+n}] + (E[\frac{\alpha + \sum X_i}{\alpha+\beta+n}] - p)^2 \\
&= \frac{Var[\sum X_i]}{(\alpha+\beta+n)^2} + (\frac{\alpha + np}{\alpha+\beta+n} - p)^2 \\
&= \frac{np(1-p)}{(\alpha+\beta+n)^2} + (\frac{\alpha + np}{\alpha+\beta+n} - p)^2 \\
&= \frac{np(1-p) + (\alpha - p(\alpha+\beta))^2}{(\alpha+\beta+n)^2}
\end{aligned}$$

(c) **[2 point]** Which estimator $\hat{p}_1$ or $\hat{p}_2$ that you will prefer when there is less data and which will you prefer when there is more data? (Hint: consider bias-variance tradeoff)
Answer:

When there is less data, depends on the prior (if we trust the prior) we should choose $\hat{p}_2$ that has lower variance. When there is more data, both estimators will typically

converge to the same value (looking at the equations), unless the prior is really bad that its effect is hard to 'wash out' with more data. Hence, when there is more data, it maybe more preferable to choose the smaller bias $\hat{p}_1$.

(d) **[3 point]** Given a particular n, find the value of α and β that will make the risk of $\hat{p}_2$ constant.

Answer:

$$\begin{aligned} R(p, \hat{p}_2) &= \frac{np(1-p) + (\alpha - p(\alpha+\beta))^2}{(\alpha+\beta+n)^2} \\ &= \frac{1}{(\alpha+\beta+n)^2}\left[\left[(\alpha+\beta)^2 - n\right]p^2 + \left[n - 2\alpha(\alpha+\beta)\right]p + \alpha^2\right] \end{aligned}$$

To make this constant (does not depend on p), we need to make the numerator a constant, thus setting:

$$\begin{aligned} (\alpha+\beta)^2 - n = 0 \\ n - 2\alpha(\alpha+\beta) = 0 \end{aligned}$$

Hence,

$$\begin{aligned} (\alpha+\beta)^2 - n &= 0 \\ \alpha+\beta &= \sqrt{n} \end{aligned}$$

Therefore,

$$\begin{aligned} n - 2\alpha(\alpha+\beta) &= 0 \\ n - 2\alpha(\sqrt{n}) &= 0 \\ \alpha &= \frac{\sqrt{n}}{2} \\ \beta &= \frac{\sqrt{n}}{2} \end{aligned}$$

(e) **[2 point]** Using Hoeffding's inequality, and knowing that $\mathbb{P}(0 \leq X_i \leq 1) = 1$, find an upper bound of $|\hat{p}_1 - p|$ with a probability of at least $1 - \gamma$.

Answer: Using Hoeffding's inequality, given $X_1, X_2, ..., X_n$, i.i.d. observations such that $E[X_i] = p$ and $a \leq X_i \leq b$; for any $\epsilon \geq 0$, $P(|\frac{1}{n}\sum_i X_i - p \mid \geq \epsilon) \leq 2e^{-2n\epsilon^2/(b-a)^2}$. Hence, in our case:

$$
\begin{aligned}
P(|\frac{1}{n}\sum_i X_i - p| \geq \epsilon) &\leq 2e^{-2n\epsilon^2} \\
P(|\frac{1}{n}\sum_i X_i - p| \leq \epsilon) &> 1 - 2e^{-2n\epsilon^2} \\
P(|\hat{p}_1 - p| \leq \epsilon) &> 1 - \gamma
\end{aligned}
$$

Hence set:

$$
\begin{aligned}
\gamma &= 2e^{-2n\epsilon^2} \\
log\ \frac{\gamma}{2} &= -2n\epsilon^2 \\
\epsilon &= \sqrt{\frac{1}{2n} log\ \frac{2}{\gamma}}
\end{aligned}
$$

Thus, the upper bound is $\epsilon = \sqrt{\frac{1}{2n} log\ \frac{2}{\gamma}}$.

Note: Whenever appropriate, give the answer in terms of α, β, γ, p, and n.

2. **[8 points]** Consider the case when X_i's have continuous value, and i.i.d according to a probability function g; i.e. $X_1, ..., X_n \sim g$.

   Let $\hat{g}$ denotes some estimator of g. The risk $R(g, \hat{g})$ in this case can be expressed as $R(g, \hat{g}) = \mathbb{E}[L(g, \hat{g})]$ where

$$
L(g, \hat{g}) = \int (\hat{g}(x) - g(x))^2 dx \tag{20}
$$

   (a) **[3 point]** Given $\tilde{R}(g, \hat{g}) = \mathbb{E}[\tilde{L}(g, \hat{g})]$ where

$$
\tilde{L}(g, \hat{g}) = \int (\hat{g}(x))^2 dx - 2\int \hat{g}(x) g(x) dx \tag{21}
$$

   Show that minimizing $\tilde{R}(g, \hat{g})$ over $\hat{g}$ is equivalent to minimizing $R(g, \hat{g})$.

   Answer:

$$
\begin{aligned}
min_{\hat{g}}\ R(g, \hat{g}) &= min_{\hat{g}}\ \mathbb{E}[\int (\hat{g}(x))^2 dx - 2\int \hat{g}(x) g(x) dx + \int (g(x))^2 dx] \\
min_{\hat{g}}\ R(g, \hat{g}) &= min_{\hat{g}}\ \mathbb{E}[\int (\hat{g}(x))^2 dx] - 2\mathbb{E}[\int \hat{g}(x) g(x) dx] + \mathbb{E}[\int (g(x))^2 dx]
\end{aligned}
$$

   where the last term is constant in terms of $\hat{g}$ (does not depend on $\hat{g}$. Hence,

$$
\begin{aligned}
min_{\hat{g}}\ R(g,\hat{g}) &= min_{\hat{g}}\ \mathbb{E}[\int(\hat{g}(x))^2dx] - 2\mathbb{E}[\int \hat{g}(x)g(x)dx] \\
min_{\hat{g}}\ R(g,\hat{g}) &= min_{\hat{g}}\ \mathbb{E}[\int(\hat{g}(x))^2dx - 2\int \hat{g}(x)g(x)dx] \\
min_{\hat{g}}\ R(g,\hat{g}) &= min_{\hat{g}}\ \tilde{R}(g,\hat{g})
\end{aligned}
$$

(b) **[5 point]** Given a second sample used as validation set, $V_1, ..., V_n \sim g$, we define the risk on this validation set as

$$\hat{R}(g,\hat{g}) = \int(\hat{g}(x))^2dx - \frac{2}{n}\sum_{i=1}^{n}\hat{g}(V_i) \tag{22}$$

where $\hat{g}$ is still based on $X_1, ..., X_n$.

Show that $\mathbb{E}[\hat{R}(g,\hat{g})] = \tilde{R}(g,\hat{g})$. Hence, $\hat{R}(g,\hat{g})$ can be used as an estimate of the risk.

Answer:

$$
\begin{aligned}
\mathbb{E}[\hat{R}(g,\hat{g})] &= \mathbb{E}[\int(\hat{g}(x))^2dx] - \mathbb{E}[\frac{2}{n}\sum_{i=1}^{n}\hat{g}(V_i)] \\
\tilde{R}(g,\hat{g}) &= \mathbb{E}[\int(\hat{g}(x))^2dx] - 2\mathbb{E}[\int \hat{g}(x)g(x)dx]
\end{aligned}
$$

We need to prove that the second terms are equal in these two expressions:

One possible answer, using Law of Large Numbers, as $n \to \infty$, $\frac{1}{n}\sum_i X_i \to \mathbb{E}[X]$. Hence,

$$
\begin{aligned}
\mathbb{E}[\frac{2}{n}\sum_{i=1}^{n}\hat{g}(V_i)] &= 2\,\mathbb{E}[\frac{1}{n}\sum_{i=1}^{n}\hat{g}(V_i)] \\
&= 2\,\mathbb{E}[\mathbb{E}[\hat{g}(V)]] \\
&= 2\,\mathbb{E}[\int \hat{g}(v)g(v)dv] \\
&= 2\,\mathbb{E}[\int \hat{g}(x)g(x)dx]
\end{aligned}
$$

Hence,

$$\mathbb{E}[\hat{R}(g,\hat{g})] = \tilde{R}(g,\hat{g})$$

Another possible answer, using Law of iterated expectation, $\mathbb{E}[V] = \mathbb{E}_X(\mathbb{E}_V(V|X))$:

$$
\begin{aligned}
\mathbb{E}\,[\frac{2}{n}\sum_{i=1}^{n}\hat{g}(V_i)] &= 2\,\mathbb{E}[\frac{1}{n}\sum_{i=1}^{n}\hat{g}(V_i)] \\
&= 2\,\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\hat{g}(V_i)] \\
&= 2\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\hat{g}(V)] \quad (\textit{since } V_i'\textit{s are i.i.d.}) \\
&= 2\,\mathbb{E}[\hat{g}(V)] \\
&= 2\,\mathbb{E}_X[\mathbb{E}_V(\hat{g}(V)|X)] \\
&= 2\,\mathbb{E}_X[\int \hat{g}(v)g(v|x)dv] \\
&= 2\,\mathbb{E}[\int \hat{g}(v)g(v)dv] \quad (\textit{since } V_i'\textit{s and } X_i'\textit{s are independent samples } \sim\ g) \\
&= 2\,\mathbb{E}[\int \hat{g}(x)g(x)dx]
\end{aligned}
$$

Hence,

$$\mathbb{E}[\hat{R}(g,\hat{g})] \;=\; \tilde{R}(g,\hat{g})$$

3. **[5 points]** In this problem you will implement L1 regularization to logistic regression. Use a step size around .0001.

   The training set for this task is given at `http://www.cs.cmu.edu/~epxing/Class/10701/hw2-train.csv`. The test set is given at `http://www.cs.cmu.edu/~epxing/Class/10701/hw2-test.csv`. The data is comma-separated (no header), with the first column being the class name. There are 2 classes: 0 and 1. Each feature can take a value: 1, 2, or 3.

   We will use cross validation to select the model class (i.e., the appropriate weight (λ) for the regularizer) which has the smallest empirical error on the validation set.

   Use **Leave-One-Out** cross validation on the training data to pick appropriate weight (λ) **between 0 and 50** for the regularizer.

   Answer: The update rule:

$$\theta_j \leftarrow \theta_j + \eta\,(\sum_i (y_i - P(y_i = 1|x_i;\theta))x_i^j - 2\lambda\theta_j)$$

   (a) **[2 points]** If there are more than one values of λ that minimizes the empirical error on the validation set, i.e.

$$\hat{\lambda} = argmin_\lambda \frac{1}{K}\sum_{k=1}^{K}\hat{R}_{V_k}(\hat{f}_{k,\lambda}) \tag{23}$$

Which value of λ will you pick (the largest? the smallest?) and why?

Answer: The largest, to cause more sparsity in learned weights and possibly make the model simpler

Using step size, $\eta = 0.0001$ and ϵ for convergence $= 0.01$:

(b) **[1 points]** Based on your answer to the previous question and your experiment result, what is the value of λ you will select for the test set? **Answer:** $\lambda = 8$

(c) **[1 points]** What is the empirical error on the validation set?

Answer: 0.06

(d) **[1 points]** What is the empirical error on the test set?

Answer: 0.15

Submit your code (zipped and named with your andrew ID or your email, in case you do not have an andrew ID) via email to dwijaya@andrew.cmu.edu.

4 SVMs

1. **[6 points]** The Radial Basis Function (a.k.a. Gaussian) kernel with inverse width $\kappa > 0$ is defined as

$$K(\mathbf{u}, \mathbf{v}) = e^{-\kappa\|\mathbf{u}-\mathbf{v}\|^2}.$$

In Figure 2 we have plotted the decision boundaries and margins for SVM learned on the same data set using the following parameters (not in the same order as the figures):

i. Linear kernel, $C = 10$;

ii. Linear kernel, $C = 1$;

iii. Linear kernel, $C = 0.1$;

iv. RBF kernel with $\kappa = 1$, $C = 3$;

v. RBF kernel with $\kappa = 10$, $C = 1$;

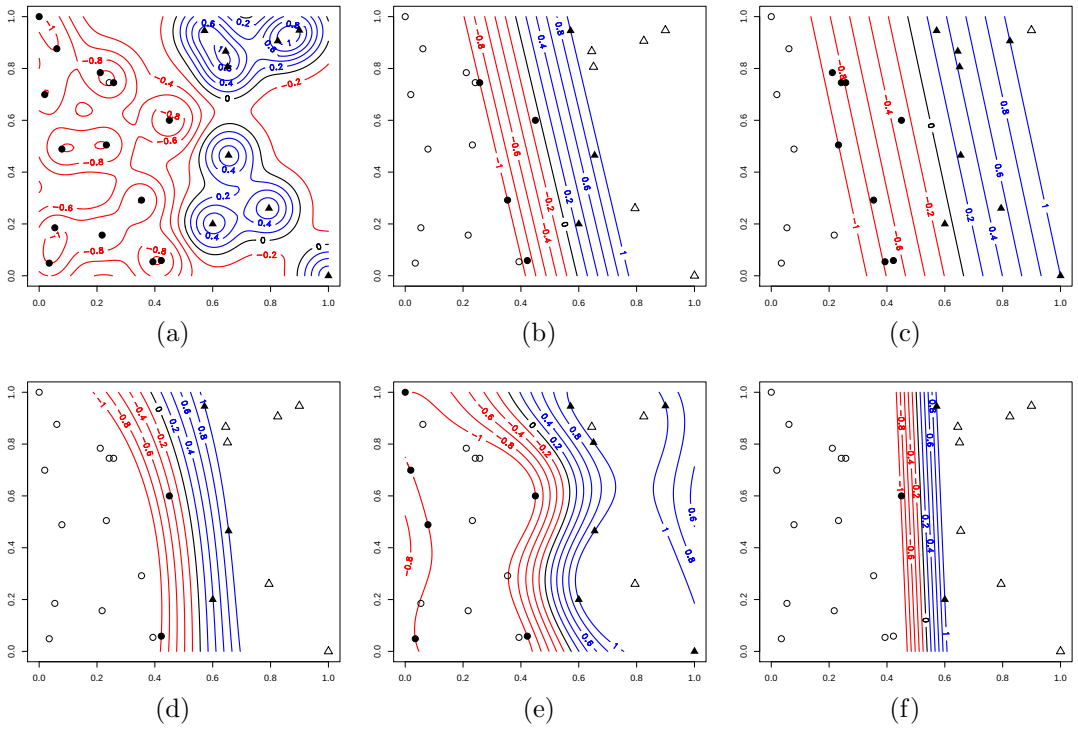
vi. RBF kernel with $\kappa = 0.1$, $C = 15$.

Match each one of the figures with one of these parameter settings. Explain your matchings in one or two words each.

Answer: Figures (b), (d), and (f) are clearly based on the linear kernel. Of these, (f) has the largest penalty on the slack variables, and (c) the smallest. This can be seen by observing either the magnitude of $w \cdot w$ in the three figures, or the number of support vectors and their positions with respect to the decision margin. So we conclude that i, ii, and iii match (f), (b), and (c) respectively.

Since Figure (a) is clearly the result of using a kernel with significantly smaller bandwidth than (d) and (e), it must correspond to v. Figures (d) and (e) can be differentiated using either the bandwidth or the penalty, both of which suggest (d) corresponds to vi and (e) to iv.

Figure 2: SVM decision boundaries and margins learned on the same data set for several parameter settings. Circles and triangles denote Class 1 and 2 respectively, solid points are support vectors.



2. **[3 points]** In Figure 2e, some of the support vectors for each class are far away from any points from the other class. For example, there are four support vectors from Class 1 near the leftmost edge of the plot (corresponding to small values for the coordinate plotted on the horizontal axis), even though there are no points from Class 2 nearby. Explain why.

   Answer: There are two ways to think about this. One is to simply state that transforming the data to the implicit feature space of the Gaussian kernel results in a very different configuration of the points, making it irrelevant that in the original space the points near the left edge of figure (e) are "far" from points of the other label.

   A possibly more intuitively satisfying view is the following. Solving the kernelized SVM gives us a linear function in the implicit feature space given by $f(x) = w^T\Phi(x) + b$, where w and $\Phi(x)$ are infinite dimensional, which we then threshold at 0 to classify a new point. We know that we can compute the same quantity in terms of *some* numbers $\beta_1, ..., \beta_n$ and b' as $f(x) = \sum_{i=1}^{n} \beta_i K(x_i, x) + b'$ (note that β_i are not necessarily positive). This is now a function of x in the original feature space that we can plot (This is exactly how Figure 2 was obtained). Note that for the Gaussian kernel, this function is simply the weighted sum of spherical Gaussians (with identical variance) centered at the training points. The decision boundary is still the set $f(x) = 0$, and the decision margin is still the set $f(x) \in [-1, 1]$. In light of this, it is obvious that for sufficiently high C, a point in a low density region must be a support vector in order to keep it outside the decision margin.

3. **[16 points]** Recall that we can restate the non-kernelized version of SVM as minimizing the sum of hinge losses per sample, defined as $\text{Loss}_{\text{SVM}}(f(x_i), y_i) = (1-(w\cdot x_i+b)y_i)_+$, regularized

by $w \cdot w$:

$$\min_{w,b} \; w \cdot w + C \sum_i \text{Loss}_{\text{SVM}}(f(x_i), y_i)$$

where (x_i, y_i) are pairs of points and labels, with $y_i \in \{-1, +1\}$.

Suppose you are given a data set $(x_1, y_1), ..., (x_n, y_n)$ with $n = 2000000$, where for $i = 1, ..., n/2$ we have $x_i = 0$ and $y_i = -1$, and for $i = n/2+1, ..., 2n$ we have $x_i = 2$ and $y_i = +1$. In other words, you are given 1 million copies of the point 0 (in one dimension, of course), all labeled -1, and 1 million copies of the point 2, all labeled $+1$.

(a) [7 points] Find w and b to minimize the SVM objective for this data, assuming $C = 1$.

What is the decision boundary and the margin of the resulting SVM? (How would the answer change if we changed C to be progressively smaller, tending to 0?)

Answer: (This is not the shortest possible solution!) Define $D = 1000000C = 10^6$. We can rewrite the objective as

$$\min_{w,b} \; w^2 + D(1 + b)_+ + D(1 - 2w - b)_+.$$

We'd like to get rid of b first. The second term is exactly 0 for any $b \leq -1$. The third term is exactly 0 for any $b \geq 1 - 2w$. If $1 - 2w \leq -1$ (i.e. $w \geq 1$), then the last two terms are both 0 for $b = -1$, which must be a minimum since both of those terms are non-negative, and the objective is w^2. Of course this is minimized at $w = 1$, with loss equal to 1.

If $1 - 2w \geq -1$ (i.e. $w \leq 1$), then the sum of the last two terms is minimized for any $-1 \leq b \leq 1 - 2w$, and is equal to $D(2 - 2w)$. Thus we must minimize $w^2 - 2Dw + 2D$ subject to $w \leq 1$. This is a convex function, and it is decreasing at $w = 1$ (the derivative is $2w - 2D$, which at $w = 1$ is $2 - 2D = 2 - 2 \cdot 10^6 < 0$), so the optimal value must be at $w = 1$. Indeed, for $w = 1$ it is equal to 1, which, luckily, matches the result we had when considering $w \geq 1$ (a different answer would indicate an error on my part).

So we have that the optimal values are $b = -1$ and $w = 1$, which corresponds to a decision boundary at $x = 1$ and a decision margin between 0 and 2.

We can also easily obtain the solution for any other value of C. In particular, we see that a larger value of C would leave the answer unchanged (since the derivative for $w \leq 1$ would still be negative at $w = 1$). In fact, the answer remains unchanged as long as $2 - 2 \cdot 10^6 C \leq 0$, i.e. $C \geq 10^{-6}$. For smaller C, the optimal value of w becomes $w = 10^6 C$.

(b) [7 points] Now suppose in addition to those $n = 2000000$ data points, we were also given an $n + 1$'th point:

$$x_{n+1} = 100, \quad y_{n+1} = -1.$$

What are the new optimal values of w and b (still using $C = 1$)?

Answer: (Again, there are much shorter solutions.) Now we need to find

$$\min_{w,b} \; w^2 + D(1 + b)_+ + D(1 - 2w - b)_+ + (1 + 100w + b)_+.$$

There are three lines where the objective is not differentiable – $b = -1$, $2w + b = 1$, and $100w + b = -1$. Let's start by considering the case $100w + b \leq -1$. Then the third

term is 0, and the objective is $w^2 + D(1+b)_+ + D(1-2w-b)_+$. If $b \le -1$, then we must minimize $w^2 + D(1-2w-b)_+$ subject to $b \le \min(-1, -1-100w)$. Since that function is decreasing in b, we set $b = \min(-1, -1-100w) = -1-100w_+$, and minimize $w^2 + D(2+100w_+ - 2w)$ (note second term is always positive, so we can omit the "+" from the subscript). This function is decreasing for negative w and increasing for positive w, so the minimum occurs at $w = 0$ and is $2D$.

For $b \ge -1$, the objective is

$$\begin{aligned} w^2 + D(1+b) + D(1-2w-b) &= w^2 + D(1+b) + D\max(1-2w-b, 0) \\ &\ge w^2 + D(1+b) + D\max(1-2w-b, -b) \\ &= w^2 + D(1+b) + D\max(1-2w, 0) - Db \\ &= w^2 + D + D(1-2w)_+ \\ &\ge D. \end{aligned}$$

In other words, subject to $100w + b \le -1$ the objective is at least D.

Now consider $100w + b \ge -1$. We must minimize $w^2 + D(1+b)_+ + D(1-2w-b)_+ + 1 + 100w + b$.

If $b \le \min(-1, 1-2w)$, the objective is $w^2 + D(1-2w-b) + 1 + 100w + b = w^2 - (2D-100)w + D + 1 - (D-1)b$. The minimum is $w^2 - (2D-100)w + D + 1 - (D-1)\min(-1, 1-2w)$. For $w \le 1$, this is $w^2 - (2D-100)w + 2D$, and has derivative $2 - 2D + 100 < 0$ at $w = 1$, so $w = 1$ is the minimum and the objective is 101. Likewise for $w \ge 1$, we must minimize $w^2 + 98w + 2$, which is minimized at $w = 1$ with value 101. We already see that the solutions subject to $100w + b \le -1$ need not be considered.

If $b \ge \max(-1, 1-2w)$, the objective is $w^2 + D(1+b) + 1 + 100w + b$, which is increasing with b so we set $b = \max(-1, 1-2w)$ and minimize $w^2 - (D+1)\min(1, 2w-1) + 100w + D + 1$. It is easy to check that this is minimized at $w = 1$, with value 101.

Finally we have to check the cases when b is between -1 and $1-2w$. If $w \ge 1$, this means $b \in [1-2w, -1]$. The objective is $w^2 + 1 + 100w + b$. The best value for b is $b = 1-2w$, the new objective is $w^2 + 98w + 2$, and the minimum is 101 at $w = 1$. If $w \le 1$, $b \in [-1, 1-2w]$, objective is $w^2 - (2D-100)w + 2D + 1 + b$, minimized at $b = -1$ and equal to $w^2 - (2D-100)w + 2D$, which in turn is minimized at $w = 1$ and equal to 101.

So the optimal solution is unchanged – $b = -1$ and $w = 1$.

(c) [2 points] Intuitively, how is the behavior of the SVM in part (b) different from what would happen if we used logistic regression instead?

Answer: The SVM solution is exactly the same in parts (a) and (b). The (unregularized) logistic regression answer in part (b) would shift (slightly) due to the extra point. This demonstrates the robustness of SVMs to certain types of outliers.