CS229 Machine Learning Problem Set #2 Solution

Roy C.K. Chan

Problem 1(a). The training on dataset A converges, whereas that on dataset B does not appear to converge.

Problem 1(b). Dataset B has the problem of complete separation, i.e., linearly separable on the $x_1 - x_2$ plane. See the file "Q1.ipynb" for data visualization and calculations.

In terms of math, logistic regression formulates the following probability

$$\mathbb{P}(y = 1|x; \theta) \quad = \quad \frac{1}{1 + \exp(-\theta^T x)} \, . \tag{1}$$

Intuitively, maximum likelihood estimation attempts to make expression (1) close to 1 when $y = 1$ and close to 0 when $y = 0$. With complete separation, if all the examples are classified correctly at some $\theta = \theta_0$, this still holds at $\theta = M \cdot \theta_0$ for some $M > 1$, but the log-likelihood attains a higher value at this new value of θ. Therefore, the log-likelihood does not reach a maximum, and the parameters grows to infinity (in absolute value).

On the other hand, dataset A clearly does not have such a problem.

Problem 1(c). Modifications (i), (ii) and (iv) do not solve the complete separation problem, so they would not lead to convergence of training algorithm on dataset B. Modification (iii) prevents the parameters from getting too large (in absolute value), so it would lead to convergence. Modification (v) may/may not solve the complete separation problem, but that would depend on randomness and properties of Gaussian noise (such as its standard deviation).

Remark: In part (a), it is observed that the quantity "Diff theta" decreases with more iterations. Indeed, I think it may drop below the given threshold, and make the training algorithm "converging". Therefore, modifications (i) and (ii) would probably lead to "convergence" in this sense, but the resulting parameters would be very large in magnitude, and probably not what we want.

Problem 1(d). No. Geometrically, logistic regression searches for a hyperplane that maximizes the log-likelihood, and scaling of parameters may affect the value of log-likelihood; on the other hand, SVM (with linear kernel) attempts to find a hyperplane that maximizes the geometric margin, which is not affected by scaling as explained in the lectures.

Problem 2(a). Given that $x^{(i)} \in \mathbb{R}^{n+1}$ and $\theta \in \mathbb{R}^{n+1}$ are finite, $0 < h_\theta(x^{(i)}) < 1$ for all $i = 1, 2, \cdots, m$. Therefore, when $(a, b) = (0, 1)$, we have $I_{0,1} = \{1, 2, \cdots, m\}$, the summation sign $\sum_{i \in I_{0,1}}$ is the same as $\sum_{i=1}^{m}$ and $|\{i \in I_{0,1}\}| = m$.

Consider the usual log-likelihood

$$l(\theta) = \sum_{i=1}^{m} y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) \,. \tag{2}$$

It can be easily shown that (or by Problem 1(a) of PS1)

$$\begin{aligned} \nabla h_\theta(x^{(i)}) &= h_\theta(x^{(i)})(1 - h_\theta(x^{(i)})) \cdot x^{(i)} \\ \implies \frac{\partial h_\theta(x^{(i)})}{\partial \theta_0} &= h_\theta(x^{(i)})(1 - h_\theta(x^{(i)})) \cdot x_0^{(i)} \\ &= h_\theta(x^{(i)})(1 - h_\theta(x^{(i)})) \,, \end{aligned} \tag{3}$$

because $x_0^{(i)} = 1$ for all i. Differentiate (2) w.r.t. θ_0, by (3), we have

$$\begin{aligned} \frac{\partial l}{\partial \theta_0} &= \sum_{i=1}^{m} y^{(i)} \cdot \frac{1}{h_\theta(x^{(i)})} \cdot h_\theta(x^{(i)})(1 - h_\theta(x^{(i)})) + (1 - y^{(i)}) \cdot \frac{1}{1 - h_\theta(x^{(i)})} \cdot \big(- h_\theta(x^{(i)})(1 - h_\theta(x^{(i)}))\big) \\ &= \sum_{i=1}^{m} y^{(i)}(1 - h_\theta(x^{(i)})) - (1 - y^{(i)}) h_\theta(x^{(i)}) \\ &= \sum_{i=1}^{m} y^{(i)} - h_\theta(x^{(i)}) \\ &= \sum_{i=1}^{m} 1\{y^{(i)} = 1\} - \mathbb{P}(y^{(i)} = 1 | x^{(i)}; \theta) \,, \end{aligned}$$

where the last equality is by the definitions of $y^{(i)}$ and $h_\theta(x^{(i)})$. Set it to zero for maximization, rearrange terms and divide both sides by m,

$$\frac{\sum_{i \in I_{0,1}} \mathbb{P}(y^{(i)} = 1 | x^{(i)}; \theta)}{|\{i \in I_{0,1}\}|} = \frac{\sum_{i \in I_{0,1}} 1\{y^{(i)} = 1\}}{|\{i \in I_{0,1}\}|} \,. \tag{4}$$

Remark: In ordinary least squares (OLS), if we include the intercept/bias term, we can apply the same technique to show that the sum of residuals is zero.

Problem 2(b). Neither of them is true.

First, we show that that perfect calibration does not imply perfect accuracy. Suppose the training set has only two examples, $(x^{(1)}, y^{(1)})$ and $(x^{(2)}, y^{(2)})$, where $x^{(1)} = x^{(2)} = [1, -1]^T$ but $y^{(1)} = 0$ and $y^{(2)} = 1$. Moreover, let the binary classification model be a logistic regression with $\theta = [1, 1]^T$, and it classifies an example as positive if and only if $h_\theta(x) \geq \frac{1}{2}$. Note that

$$\mathbb{P}(y^{(i)} = 1 | x^{(i)}; \theta) = \frac{1}{1 + \exp(-0)} = \frac{1}{2}$$

for all $i = 1, 2$. Such a model is perfectly calibrated over any range $(a, b) \subset [0, 1]$ because

$$\frac{\sum_{i \in I_{a,b}} \mathbb{P}(y^{(i)} = 1 | x^{(i)}; \theta)}{|\{i \in I_{a,b}\}|} = \frac{1}{2} = \frac{\sum_{i \in I_{a,b}} 1\{y^{(i)} = 1\}}{|\{i \in I_{a,b}\}|},$$

for $0 \leq a < \frac{1}{2} < b \leq 1$. For $a \geq \frac{1}{2}$ or $b \leq \frac{1}{2}$, the index set $I_{a,b}$ is empty, so the property holds (vacuously). However, the model classifies the first training example as positive wrongly because $y^{(1)} = 0$.

Second, we show that perfect accuracy does not imply perfect calibration. In Q1(b), the logistic regression achieves perfect accuracy (after a certain number of iterations). But over the range $(a, b) = (\frac{1}{2}, 1)$, the model is not well-calibrated because

$$\mathbb{P}(y^{(i)} = 1 | x^{(i)}; \theta) < 1 = 1\{y^{(i)} = 1\}$$

for all $i \in I_{a,b}$.

Remark: These counter-examples are two extremes: the first one is "completely inseparable", while the second one is "completely separable".

Problem 2(c). If we regularize all the parameters except θ_0, then the expression for $\frac{\partial l}{\partial \theta_0}$ remains the same. As a result, relationship (4), i.e., the property in part (a), still holds true. On the other hand, if we regularize all the parameters including θ_0, consider the L_2 regularized log-likelihood

$$\begin{aligned} l(\theta) &= \sum_{i=1}^{m} y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) - \lambda \sum_{j=0}^{n} \theta_j^2 \\ \implies \frac{\partial l}{\partial \theta_0} &= \sum_{i=1}^{m} 1\{y^{(i)} = 1\} - \mathbb{P}(y^{(i)} = 1 | x^{(i)}; \theta) - 2\lambda\theta_0 \,. \end{aligned}$$

Relationship (4) no longer holds, because $\lambda > 0$ and it is unlikely that $\theta_0 = 0$. (L_2 regularization usually cannot set coefficients to exactly zero, whereas L_1 regularization can.)

Problem 3. For maximum likelihood estimation of logistic regression,

$$\theta_{ML} = \arg\max_{\theta} \sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) \,. \tag{5}$$

For maximum a posteriori estimation of Bayesian logistic regression, the prior $\theta \sim \mathcal{N}(0, \tau^2 I)$, so we have

$$\begin{aligned} p(\theta) &= \frac{1}{(2\pi)^{\frac{n+1}{2}} \tau^{n+1}} \exp\big(-\frac{||\theta||_2^2}{2\tau^2}\big) \\ &\propto \exp\big(-\frac{||\theta||_2^2}{2\tau^2}\big) \,, \end{aligned}$$

hence,

$$\theta_{MAP} = \arg\max_{\theta} \sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) - \frac{||\theta||_2^2}{2\tau^2} \,. \tag{6}$$

Now, we show the desired result using a proof by contradiction. Assume the contrary that $||\theta_{MAP}||_2 > ||\theta_{ML}||_2$,

$$\begin{aligned} &\sum_{i=1}^{m} y^{(i)} \log h_{\theta_{MAP}}(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_{\theta_{MAP}}(x^{(i)})) - \frac{||\theta_{MAP}||_2^2}{2\tau^2} \\ \geq &\sum_{i=1}^{m} y^{(i)} \log h_{\theta_{ML}}(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_{\theta_{ML}}(x^{(i)})) - \frac{||\theta_{ML}||_2^2}{2\tau^2} \\ > &\sum_{i=1}^{m} y^{(i)} \log h_{\theta_{ML}}(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_{\theta_{ML}}(x^{(i)})) - \frac{||\theta_{MAP}||_2^2}{2\tau^2} \\ \geq &\sum_{i=1}^{m} y^{(i)} \log h_{\theta_{MAP}}(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_{\theta_{MAP}}(x^{(i)})) - \frac{||\theta_{MAP}||_2^2}{2\tau^2} \,, \end{aligned}$$

where the first inequality holds because θ_{MAP} solves the maximization problem in (6), the second (strict) inequality comes from the assumption that $||\theta_{MAP}||_2 > ||\theta_{ML}||_2$, and the third inequality is true because θ_{ML} solves the maximization problem in (5). However, this is clearly a contradiction because the first and the last expressions are the same.

Remark: In this question, we do not make use of the properties of the sigmoid function in logistic regression, so it can be easily seen that the same arguments also apply to other generalized linear models, such as linear regression.

Problem 4(a). Yes. By Mercer's theorem, both K_1 and K_2 are symmetric and positive semidefinite. So, the matrix $K = K_1 + K_2$ is obviously symmetric, and also positive semidefinite because

$$y^T K y = y^T K_1 y + y^T K_2 y \geq 0 + 0 = 0\,,$$

for any $y \in \mathbb{R}^m$.

Problem 4(b). No. Choose $K_2 = 2K_1$, and by part (c), K_2 is a valid kernel. Therefore, the matrix $K = -K_1$ is negative semidefinite. More explicitly, a counter-example is given by $K_1(x, z) = xz$, where $x, z \in \mathbb{R}$, and $K_2 = 2K_1$; therefore, with the finite set $\{x^{(1)}\}$ where $x^{(1)} = 1$, and choose $y = 1$, we have $y^T K y = -1 < 0$, i.e., K is not positive semidefinite.

Problem 4(c). Yes. By Mercer's theorem, K_1 is symmetric and positive semidefinite. So, the matrix $K = aK_1$ is obviously symmetric, and also positive semidefinite because

$$y^T K y = y^T (aK_1) y = a(y^T K_1 y) \geq 0\,,$$

for any $a \in \mathbb{R}^+$, $y \in \mathbb{R}^m$.

Problem 4(d). No. The matrix K is negative semidefinite. Choose $a = 1$ and the same counter-example in part (b) applies.

Problem 4(e). Yes. Both K_1 and K_2 are valid kernels, so

$$K_1(x, z) = \phi_1(x)^T \phi_1(z)\,,$$
$$K_2(x, z) = \phi_2(x)^T \phi_2(z)\,,$$

for some $\phi_1 : \mathbb{R}^n \to \mathbb{R}^{n_1}, \phi_2 : \mathbb{R}^n \to \mathbb{R}^{n_2}$. Therefore,

$$\begin{aligned}
K(x, z) &= K_1(x, z) K_2(x, z) \\
&= \phi_1(x)^T \phi_1(z) \phi_2(x)^T \phi_2(z) \\
&= \sum_{i=1}^{n_1} \phi_{1i}(x)\phi_{1i}(z) \sum_{j=1}^{n_2} \phi_{2j}(x)\phi_{2j}(z) \\
&= \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \big(\phi_{1i}(x)\phi_{2j}(x)\big)\big(\phi_{1i}(z)\phi_{2j}(z)\big) \\
&= \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \Phi_{ij}(x)\Phi_{ij}(z)\,,
\end{aligned}$$

where the last equality holds by defining a function $\Phi : \mathbb{R}^n \to \mathbb{R}^{n_1 \times n_2}$ such that $\Phi_{ij} = \phi_{1i}\phi_{2j}$ for all $i = 1, 2, \cdots, n_1$ and $j = 1, 2, \cdots, n_2$, so the kernel K corresponds to a feature mapping to a $(n_1 n_2)$-dimensional space.

Remark: We can also use the Mercer's theorem to prove the result as follows. Both K_1 and K_2 are symmetric and positive semidefinite, so the square matrix $K = K_1 \odot K_2$ (where $\odot$ denotes elementwise multiplication) is clearly symmetric, and is positive semidefinite by the Schur product theorem.

Problem 4(f). Yes. By choosing $\Phi = f$, we have $K(x, z) = f(x)f(z) = \Phi(x)^T \Phi(z)$.

Problem 4(g). Yes. K_3 is a valid kernel, so $K_3(x,z) = \phi_1(x)^T\phi_1(z)$ for some $\phi_1 : \mathbb{R}^d \to \mathbb{R}^{n_3}$. Therefore,

$$\begin{aligned} K(x,z) &= K_3(\phi(x),\phi(z)) \\ &= \phi_1(\phi(x))^T\phi_1(\phi(z)) \\ &= \Phi(x)\Phi(z)\,, \end{aligned}$$

where the last equality holds by defining a function $\Phi = \phi_1 \circ \phi$. The kernel K corresponds to a feature mapping to a n_3-dimensional space.

Problem 4(h). Yes. $K(x,z) = p(K_1(x,z))$ can be constructed by a finite sequence of the following operations: (a) sum, (c) positive scalar multiplication, (e) power, and (f) constant.

Problem 5. Define the kernel $K(x,z) = \langle\phi(x),\phi(z)\rangle$. Observe the update rule

$$\theta^{(i+1)} := \theta^{(i)} + \alpha 1\{g(\langle\theta^{(i)},\phi(x^{(i+1)})\rangle)y^{(i+1)} < 0\}y^{(i+1)}\phi(x^{(i+1)})\,,$$

which updates the parameter vector $\theta^{(i)}$, by adding $\phi(x^{(i+1)})$ multiplied by some "weight", and the "weight" is expressed in terms of the inner product $\langle\theta^{(i)},\phi(x^{(i+1)})\rangle$ (and also $y^{(i+1)}$).

Using such a recursive definition, it can be easily shown by induction that (I) $\theta^{(i)} = \sum_{t=1}^{i}\beta_t\phi(x^{(t)})$, i.e., a linear combination of $\phi(x^{(t)})$, $t = 1,\cdots,i$, for all $i = 1,2,\cdots,m$; and (II) the "weights" β_t's can be computed efficiently using the kernel trick (in a sequential manner). We can store these coefficients so that each parameter vector $\theta^{(i)}$ is represented by a list $[\beta_1,\beta_2,\cdots,\beta_i]$, and the initial parameter vector $\theta^{(0)}$ is represented by an empty list [].

For $i \geq 1$, using the above representation of $\theta^{(i)}$, we can make a prediction on a new input $x^{(i+1)}$ by $g(\langle\theta^{(i)},\phi(x^{(i+1)})\rangle) = g(\sum_{t=1}^{i}\beta_t\langle\phi(x^{(t)}),\phi(x^{(i+1)})\rangle) = g(\sum_{t=1}^{i}\beta_t K(x^{(t)},x^{(i+1)}))$, which can be computed efficiently. For $i = 0$, the predicted value is $g(\langle\theta^{(0)},\phi(x^{(1)})\rangle) = g(0) = 1$.

Given that we compute the prediction on $x^{(i+1)}$ efficiently as stated above, we just need to compute the "weight" $\beta_{i+1} = \alpha 1\{g(\langle\theta^{(i)},\phi(x^{(i+1)})\rangle)y^{(i+1)} < 0\}y^{(i+1)}$, and append it to the end of the list $[\beta_1,\beta_2,\cdots,\beta_i]$.

Problem 6. See the files "nb.py" and "Q6.ipynb", which are located in the directory */ps2/Q6/spam_data*.

Remark: I modified the "evaluation" function in both "nb.py" and "svm.py", so that it returns the error useful for plotting in part (c) and (d).