Problem Set 1: Supervised Learning

Eitan Joseph Caroline Wang

July 1, 2020

1 Problem 1

Newton's method for computing least squares

In this problem, we will prove that if we use Newton's method solve the least squares optimization problem, then we only need one iteration to converge to θ.

(a) Find the Hessian of the cost function $J(\theta) = \frac{1}{2}\sum_{i=1}^{m}\left(\theta^T x^{(i)} - y^{(i)}\right)^2$.
answer:

$$H_{kj} = \frac{\partial^2}{\partial\theta_k\partial\theta_j}J(\theta) = \frac{\partial^2}{\partial\theta_k\partial\theta_j}\frac{1}{2}\sum_{i=1}^{m}\left(\theta^T x^{(i)} - y^{(i)}\right)^2 = \frac{\partial}{\partial\theta_k}\sum_{i=1}^{m}\left(\theta^T x^{(i)} - y^{(i)}\right)(x_j^{(i)}) = \sum_{i=1}^{m} x_k^{(i)} x_j^{(i)}$$

The sum can be understood as $x_k^{(i)} \cdot x_j^{(i)} \quad \forall i$

$$H = X^T X$$

(b) Show that the first iteration of Newton's method gives us $\theta^* = \left(X^T X\right)^{-1} X^T\vec{y}$, the solution to our least squares problem.
answer:

The first iteration of Newton's Method is given by $\theta^* = \theta^{^{(0)}} - H^{-1}\nabla_{\theta^{(0)}} J(\theta^{^{(0)}})$

Via lecture 2 we know that $\nabla_\theta J(\theta) = X^T X\theta - X^T\vec{y}$

Which means we need to solve $\theta^* = \theta^{^{(0)}} - H^{-1}\left(X^T X\theta^{^{(0)}} - X^T\vec{y}\right)$

We can substitute our result from part (a) to get $\theta^* = \theta^{^{(0)}} - \left(X^T X\right)^{-1}\left(X^T X\theta^{^{(0)}} - X^T\vec{y}\right)$

This reduces to $\theta^* = \left(X^T X\right)^{-1} X^T\vec{y}$

2 Problem 3

Multivariate least squares

So far in class, we have only considered cases where our target variable y is a scalar value. Suppose that instead of trying to predict a single output, we have a training set with multiple outputs for each example:

$$\{(x^{^{(i)}} y^{^{(i)}}), i = 1, ..., m\},\ x^{^{(i)}} \in \mathbb{R}^n,\ y^{^{(i)}} \in \mathbb{R}^p$$

Thus for each training example, $y^{(i)}$ is vector-valued, with p entries. We wish to use a linear model to predict the outputs, as in least squares, by specifying the parameter matrix Θ in

$$y = \Theta^T x, \text{ where } \Theta \in \mathbb{R}^{n\times p}$$

(a) The cost function for this case is

$$J(\Theta) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{p}\left((\Theta^T x^{(i)})_j - y_j^{(i)}\right)^2$$

Write $J(\Theta)$ in matrix-vector notation (i.e., without using any summations).

$$X = \begin{bmatrix} — & (x^{(1)})^T & — \\ — & (x^{(2)})^T & — \\ & \vdots & \\ — & (x^{(m)})^T & — \end{bmatrix}$$

and the $m \times p$ target matrix

$$Y = \begin{bmatrix} — & (y^{(1)})^T & — \\ — & (y^{(2)})^T & — \\ & \vdots & \\ — & (y^{(m)})^T & — \end{bmatrix}$$

answer:

$$\begin{aligned} J(\Theta) &= \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{p}\left((\Theta^T x^{(i)})_j - y_j^{(i)}\right)^2 \\ &= \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{p}(X\Theta - Y)_{i,j}^2 \\ &= \frac{1}{2}\sum_{j=1}^{p}(X\Theta - Y)^T(X\Theta - Y)_{j,j} \\ &= \frac{1}{2}\operatorname{tr}\left((X\Theta - Y)^T(X\Theta - Y)\right) \end{aligned}$$

(b) Find the closed form solution for which minimizes $J(\Theta)$. This is the equivalent to the normal equations for the multivariate case.

answer:
We now have an optimization problem of the form:

$$\min_{\Theta}\ J(\Theta)$$

We can solve this by setting the gradient of $J(\Theta)$ to 0 and solving for Θ.

$$
\begin{aligned}
\nabla_\Theta J(\Theta) &= \frac{1}{2}\nabla_\Theta \operatorname{tr}\left(\left(X\Theta - Y\right)^T\left(X\Theta - Y\right)\right) \\
&= \frac{1}{2}\nabla_\Theta \operatorname{tr}\left(\Theta^T X^T X\Theta - \Theta^T X^T Y - Y^T X\Theta + Y^T Y\right) \\
&= \frac{1}{2}\left(\nabla_\Theta \operatorname{tr}\Theta^T X^T X\Theta - 2\nabla_\Theta \operatorname{tr}\Theta^T X^T Y + \nabla_\Theta \operatorname{tr} Y^T Y\right) \\
&= \frac{1}{2}\left(X^T X\Theta + X^T X\Theta - 2\nabla_\Theta \operatorname{tr}\Theta^T X^T Y\right) \\
&= X^T X\Theta - X^T Y
\end{aligned}
$$

Now after setting this result to 0 we get

$$
\begin{aligned}
X^T X\Theta - X^T Y = 0 \\
\Theta = \left(X^T X\right)^{-1} X^T Y
\end{aligned}
$$

(c) Suppose instead of considering the multivariate vectors $y^{(i)}$ all at once, we instead compute each variable $y_j^{(i)}$ separately for each $j = 1, \ldots, p$. In this case, we have a p individual linear models, of the form

$$
y_j^{(i)} = \theta_j^T x^{(i)},\ j = 1, \ldots, p.
$$

How do the parameters from these p independent least squares problems compare to the multivariate solution?

answer:
We first realize that Θ can be written in terms of each θ_js as

$$
\sum_{i=1}^{p}\begin{bmatrix}
\theta_i^{(1)}\mathbb{1}\{i=1\} & \theta_i^{(1)}\mathbb{1}\{i=2\} & \ldots & \theta_i^{(1)}\mathbb{1}\{i=p\} \\
\theta_i^{(2)}\mathbb{1}\{i=1\} & \theta_i^{(2)}\mathbb{1}\{i=2\} & \ldots & \theta_i^{(2)}\mathbb{1}\{i=p\} \\
\vdots & \vdots & \ddots & \vdots \\
\theta_i^{(n)}\mathbb{1}\{i=1\} & \theta_i^{(n)}\mathbb{1}\{i=2\} & \ldots & \theta_i^{(n)}\mathbb{1}\{i=p\}
\end{bmatrix}_{n\times p} = \begin{bmatrix}\theta_1 & \theta_2 & \ldots & \theta_p\end{bmatrix}
$$

Combining this with the original result of part (b) gives us

$$
\begin{aligned}
\begin{bmatrix}\theta_1 & \theta_2 & \ldots & \theta_p\end{bmatrix} &= \begin{bmatrix}\left(X^T X\right)^{-1} X^T \vec{y_1} & \left(X^T X\right)^{-1} X^T \vec{y_2} & \ldots & \left(X^T X\right)^{-1} X^T \vec{y_p}\end{bmatrix} \\
&= \left(X^T X\right)^{-1} X^T \begin{bmatrix}\vec{y_1} & \vec{y_2} & \ldots & \vec{y_p}\end{bmatrix} \\
&= \left(X^T X\right)^{-1} X^T Y \\
&= \Theta
\end{aligned}
$$

This result implies that evaluating the parameters separately yields the same result as evaluating them together.

3 Problem 4

Naive Bayes

In this problem, we look at maximum likelihood parameter estimation using the naive Bayes assumption. Here, the input features $x_j, j = 1, \ldots, n$ to our model are discrete, binary-valued variables, so $x_j \in \{0,1\}$. We call $x = [x_1 x_2 \ldots x_n]^T$ to be the input vector. For each training example, our output targets are a single binary-value $y \in \{0,1\}$. Our model is then parameterized by $\phi_{j|y=0} = p(x_j = 1|y = 0)$, $\phi_{j|y=1} = p(x_j = 1|y = 1)$, and $\phi_y = p(y = 1)$. We model the joint distribution of (x, y) according to

$$
\begin{aligned}
p(y) &= (\phi_y)^y (1-\phi_y)^{1-y} \\
p(x|y=0) &= \prod_{j=1}^{n} p(x_j|y=0) \\
&= \prod_{j=1}^{n} (\phi_{j|y=0})^{x_j} (1-\phi_{j|y=0})^{1-x_j} \\
p(x|y=1) &= \prod_{j=1}^{n} p(x_j|y=1) \\
&= \prod_{j=1}^{n} (\phi_{j|y=1})^{x_j} (1-\phi_{j|y=1})^{1-x_j}
\end{aligned}
$$

(a) Find the joint likelihood function $\ell(\varphi) = \log \prod_{i=1}^{m} p(x^{(i)}, y^{(i)}; \varphi)$ in terms of the model parameters given above. Here, φ represents the entire set of parameters $\{\phi_y, \phi_{j|y=0}, \phi_{j|y=1} | j = 1, \ldots, n\}$.

answer:

$$
\begin{aligned}
\ell(\varphi) &= \log \prod_{i=1}^{m} p(x^{(i)}, y^{(i)}; \varphi) \\
&= \log \prod_{i=1}^{m} \left(\prod_{j=1}^{n_i} p(x_j^{(i)} | y^{(i)}; \varphi) \right) p(y^{(i)}; \varphi) \\
&= \sum_{i=1}^{m} \left(\log p(y^{(i)}; \varphi) + \log \prod_{j=1}^{n_i} p(x_j^{(i)} | y^{(i)}; \varphi) \right) \\
&= \sum_{i=1}^{m} \left(\log p(y^{(i)}; \varphi) + \sum_{j=1}^{n_i} \log p(x_j^{(i)} | y^{(i)}; \varphi) \right) \\
&= \sum_{i=1}^{m} \left(\log \left((\phi_y)^{y^{(i)}} (1-\phi_y)^{1-y^{(i)}} \right) + \sum_{j=1}^{n_i} \log \left((\phi_{j|y})^{x_j} (1-\phi_{j|y})^{1-x_j} \right) \right) \\
&= \sum_{i=1}^{m} \left(y^{(i)} \log(\phi_y) + (1-y^{(i)}) \log(1-\phi_y) + \sum_{j=1}^{n_i} x_j \log(\phi_{j|y}) + (1-x_j) \log(1-\phi_{j|y}) \right)
\end{aligned}
$$

(b) Show that the parameters which maximize the likelihood function are the same as those given in the lecture notes.

answer:

To find the maximum ϕ_y that maximizes the likelihood function, we will set the gradient with respect to ϕ_y of the result above to zero.

$$
\begin{aligned}
& \nabla_{\phi_y} \sum_{i=1}^{m} \left(y^{(i)} \log(\phi_y) + (1 - y^{(i)}) \log(1 - \phi_y) + \sum_{j=1}^{n_i} x_j \log(\phi_{j|y}) + (1 - x_j) \log(1 - \phi_{j|y}) \right) \stackrel{set}{=} 0 \\
= & \quad \nabla_{\phi_y} \sum_{i=1}^{m} y^{(i)} \log(\phi_y) + (1 - y^{(i)}) \log(1 - \phi_y) \\
= & \quad \sum_{i=1}^{m} \frac{y^{(i)}}{\phi_y} - \frac{1 - y^{(i)}}{1 - \phi_y} \\
= & \quad \sum_{i=1}^{m} y^{(i)} (1 - \phi_y) - \phi_y (1 - y^{(i)}) \\
= & \quad \sum_{i=1}^{m} y^{(i)} - \phi_y = 0 \\
\implies & \quad \sum_{i=1}^{m} y^{(i)} = \sum_{i=1}^{m} \phi_y \\
\implies & \quad \sum_{i=1}^{m} y^{(i)} = m\phi_y \\
\implies & \quad \phi_y = \frac{\sum_{i=1}^{m} \mathbb{1}\{y^{(i)} = 1\}}{m}
\end{aligned}
$$

Similarly for $\phi_{j|y}$ we can do the same thing and solve

$$
\begin{aligned}
& \nabla_{\phi_{j|y}} \sum_{i=1}^{m} \left(y^{(i)} \log(\phi_y) + (1 - y^{(i)}) \log(1 - \phi_y) + \sum_{j=1}^{n_i} x_j \log(\phi_{j|y}) + (1 - x_j) \log(1 - \phi_{j|y}) \right) \stackrel{set}{=} 0 \\
= & \quad \sum_{i=1}^{m} \frac{x_j^{(i)}}{\phi_{j|y}} - \frac{1 - x_j^{(i)}}{1 - \phi_{j|y}} = 0 \\
= & \quad \sum_{i=1}^{m} x_j^{(i)} (1 - \phi_{j|y}) - \phi_{j|y} (1 - x_j^{(i)}) = 0 \\
= & \quad \sum_{i=1}^{m} x_j^{(i)} - \phi_{j|y} = 0
\end{aligned}
$$

We can now separate this into two cases.

$\phi_{j|y=0}$:

$$\begin{aligned}
&\sum_{i=1}^{m}(x_j^{(i)} - \phi_{j|y=0})\mathbb{1}\{y^{(i)} = 0\} = 0 \\
\implies &\sum_{i=1}^{m} x_j^{(i)}\mathbb{1}\{y^{(i)} = 0\} = \sum_{i=1}^{m}\phi_{j|y=0}\mathbb{1}\{y^{(i)} = 0\} \\
\implies &\phi_{j|y=0} = \frac{\sum_{i=1}^{m} x_j^{(i)}\mathbb{1}\{y^{(i)} = 0\}}{\sum_{i=1}^{m}\mathbb{1}\{y^{(i)} = 0\}} \\
\implies &\phi_{j|y=0} = \frac{\sum_{i=1}^{m}\mathbb{1}\{x_j^{(i)} = 0 \wedge y^{(i)} = 0\}}{\sum_{i=1}^{m}\mathbb{1}\{y^{(i)} = 0\}}
\end{aligned}$$

$\phi_{j|y=1}$:

$$\begin{aligned}
&\sum_{i=1}^{m}(x_j^{(i)} - \phi_{j|y=1})\mathbb{1}\{y^{(i)} = 1\} = 0 \\
\implies &\sum_{i=1}^{m} x_j^{(i)}\mathbb{1}\{y^{(i)} = 1\} = \sum_{i=1}^{m}\phi_{j|y=1}\mathbb{1}\{y^{(i)} = 1\} \\
\implies &\phi_{j|y=1} = \frac{\sum_{i=1}^{m} x_j^{(i)}\mathbb{1}\{y^{(i)} = 1\}}{\sum_{i=1}^{m}\mathbb{1}\{y^{(i)} = 1\}} \\
\implies &\phi_{j|y=1} = \frac{\sum_{i=1}^{m}\mathbb{1}\{x_j^{(i)} = 1 \wedge y^{(i)} = 1\}}{\sum_{i=1}^{m}\mathbb{1}\{y^{(i)} = 1\}}
\end{aligned}$$

4 Problem 5

Exponential family and the geometric distribution

(a) Consider the geometric distribution parameterized by ϕ:

$$p(y;\phi) = (1-\phi)^{y-1}\phi, \quad j = 1, 2, 3\dots$$

Show that the geometric distribution is in the exponential family, and give $b(y)$, η, $T(y)$ and $a(\eta)$.

answer:

Recall that to be a member of the exponential family distribution, a distribution's PDF must be in the form $p(y;\phi) = b(y)\exp[T(y)\cdot\eta - a(\eta)]$.

Here we have

$$\begin{aligned}
p(y;\phi) =& (1-\phi)^{y-1}\phi \\
=& \exp[\log(1-\phi)^{y-1} + \log(\phi)] \\
=& \exp[(y-1)\log(1-\phi) + \log(\phi)] \\
=& \exp[y\log(1-\phi) - \log(1-\phi) + \log(\phi)]
\end{aligned}$$

Which can be decomposed as

$$
\begin{aligned}
&b(y) = 1 \\
&\eta = \log(1-\phi) \\
&T(y) = y \\
&a(\eta) = -\eta + \log(1-e^{\eta})
\end{aligned}
$$

(b) Consider performing regression using a GLM model with a geometric response variable. What is the canonical response function for the family? You may use the fact that the mean of a geometric distribution is given by $1/\phi$.

answer:

$$g(\eta) = E[y;\phi] = \frac{1}{\phi} = \frac{1}{1-e^{\eta}}$$

(c) For a training set $\{(x^{(i)}, y^{(i)}); i = 1, \ldots m\}$, let the log-likelihood of an example be $\log p(y^{(i)}|x^{(i)};\theta)$. By taking the derivative of the log-likelihood with respect to, derive the stochastic gradient ascent rule for learning using a GLM model with geometric responses y and the canonical response function.

answer:

The log-likelihood of θ with respect to a training example $(x^{(i)}, y^{(i)})$ is defined as $l_i(\theta) = \log p(y^{(i)}|x^{(i)};\theta)$. We use the GLM assumption that $\eta = \theta^T x$. Therefore, we obtain

$$
\begin{aligned}
l_i(\theta) =& \log[\exp(\theta^T x^{(i)} \cdot y^{(i)} - \theta^T x^{(i)} + \log(1-e^{\theta^T x^{(i)}}))] \\
=& \log[\exp(\theta^T x^{(i)} \cdot y^{(i)} - \log(e^{\theta^T x^{(i)}}) + \log(1-e^{\theta^T x^{(i)}}))] \\
=& \log\left[\exp\left(\theta^T x^{(i)} \cdot y^{(i)} - \log\left(\frac{e^{\theta^T x^{(i)}}}{1-e^{\theta^T x^{(i)}}}\right)\right)\right] \\
=& \log\left[\exp\left(\theta^T x^{(i)} \cdot y^{(i)} - \log\left(\frac{1}{e^{-\theta^T x^{(i)}}-1}\right)\right)\right] \\
=& \theta^T x^{(i)} \cdot y^{(i)} + \log\left(e^{-\theta^T x^{(i)}} - 1\right)
\end{aligned}
$$

Then we want to take the gradient respect to θ_j

$$
\begin{aligned}
\nabla_{\theta_j} =& {x^{(i)}}_j \cdot y^{(i)} + \frac{e^{-\theta^T x^{(i)}}}{e^{-\theta^T x^{(i)}} - 1}(-{x^{(i)}}_j) \\
=& {x^{(i)}}_j \left(y^{(i)} - \frac{e^{-\theta^T x^{(i)}}}{e^{-\theta^T x^{(i)}} - 1}\right) \\
=& {x^{(i)}}_j \left(y^{(i)} - \frac{1}{1-e^{\theta^T x^{(i)}}}\right)
\end{aligned}
$$

Finally, we can derive the stochastic gradient ascent update rule

$$\theta_j := \theta_j + \alpha\left({x^{(i)}}_j\left(y^{(i)} - \frac{1}{1-e^{\theta^T x^{(i)}}}\right)\right)$$