ELE 535: Machine Learning and Pattern Recognition
Homework 5

Zachary Hervieux-Moore

Monday 22$^{\text{nd}}$ October, 2018

Exercise 1: Derive the derivative, and if exists the gradient, of the following functions.

a) For $x \in \mathbb{R}^n$, $f(x) = \sum_{j=1}^{n} x_j$.

b) For $x \in \mathbb{R}^n$, $f(x) = e^{\sum_{j=1}^{n} x_j}$.

c) For $x \in \mathbb{R}^n$, $f(x) = x^T A x + a^T x + b$, where $b \in \mathbb{R}$, $a \in \mathbb{R}^n$, and $A \in \mathbb{R}^{n \times n}$.

d) For $M \in \mathbb{R}^{n \times n}$, $f(M) = \|M\|_F^2$.

e) For $x \in \mathbb{R}^n$, $f(x) = xx^T \in \mathbb{R}^{n \times n}$.

Answer:

a) The i^{th} component of the gradient is given by:

$$[\nabla f(x)]_i = 1$$

Thus, we have that the gradient is $\nabla f(x) = \mathbf{1}$ and the derivative is $Df(x)(v) = \mathbf{1}^T v$.

b) The i^{th} component of the gradient is given by:

$$[\nabla f(x)]_i = e^{\sum_{j=1}^{n} x_j}$$

Thus, we have that the gradient is $\nabla f(x) = f(x) \cdot \mathbf{1}$ and the derivative is $Df(x)(v) = f(x)\mathbf{1}^T v$.

c) Taking the derivative directly, have that the gradient is $\nabla f(x) = 2Ax + a$ and the derivative is $Df(x)(v) = (2x^T A^T + a^T)v$.

d) We have that $f(M) = \|M\|_F^2 = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij}^2$. This means that the gradient is

$$\begin{aligned}
[\nabla_M f(M)]_{ij} &= 2a_{ij} \\
\implies \nabla_M f(M) &= 2M \\
\implies Df(M)(V) &= 2a_{ij}v_{ij} = 2\text{trace}(M^T V)
\end{aligned}$$

Thus, we have that the gradient is $\nabla_M f(M) = 2M$ and the derivative is $Df(M)(V) = 2\text{trace}(M^T V)$.

e) The outer product is given by

$$f(x) = xx^T = \begin{bmatrix} x_1^2 & x_1x_2 & \cdots & x_1x_n \\ \vdots & & & \vdots \\ x_nx_1 & x_nx_2 & \cdots & x_n^2 \end{bmatrix}.$$

For illustrative purposes, the 1^{st} partial derivative is given by:

$$\frac{\partial f(x)}{\partial x_1} = \begin{bmatrix} 2x_1 & x_2 & \cdots & x_n \\ \vdots & & 0 & \\ x_n & & & \end{bmatrix}$$

The i^{th} partial has a similar form except the 0's occur on the entries not in the i^{th} row or column. This yields a derivative of

$$Df(x)(v) = \sum_{j=1}^{n} \frac{\partial f(x)}{\partial x_j} v_j = \begin{bmatrix} 2x_1v_1 & x_2v_1 & \cdots & x_nv_1 \\ \vdots & & 0 & \\ x_nv_1 & & & \end{bmatrix}$$

$$+ \cdots + \begin{bmatrix} & & & x_1v_n \\ & 0 & & \vdots \\ x_1v_n & x_2v_n & \cdots & 2x_nv_n \end{bmatrix}$$

$$Df(x)(v) = \begin{bmatrix} 2x_1v_1 & x_2v_1 + x_1v_2 & \cdots & x_nv_1 + x_1v_n \\ \vdots & & & \vdots \\ x_nv_1 + x_1v_n & x_nv_2 + x_2v_n & \cdots & 2x_nv_n \end{bmatrix}$$

$$Df(x)(v) = x \otimes v + v \otimes x$$

Exercise 2: Matrix Inversion Lemma. Let $M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ with $A \in \mathbb{R}^{p\times p}$, $D \in \mathbb{R}^{q\times q}$, $B \in \mathbb{R}^{p\times q}$, $C \in \mathbb{R}^{q\times p}$.

a) If A and D, and at least one of S_A or S_D are invertible, derive the equality (this was partially done in class):

$$(A - BD^{-1}C)^{-1} = A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}$$

b) Use part a) to show that if A and D, and at least one of $A + BDC$ or $D^{-1} + CA^{-1}B$ are invertible, then:

$$(A + BDC)^{-1} = A^{-1} - A^{-1}B(D^{-1} + CA^{-1}B)^{-1}CA^{-1}$$

Answer:

a) This can be shown directly by checking that multiplying the two results in the identity.

$$\begin{aligned}
&(A - BD^{-1}C)(A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}) \\
&= I + B(D - CA^{-1}B)^{-1}CA^{-1} - BD^{-1}CA^{-1} - BD^{-1}CA^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}) \\
&= I - BD^{-1}CA^{-1} + BD^{-1}(D - CA^{-1}B)^{-1}(D - CA^{-1}B)^{-1}CA^{-1} \\
&= I - BD^{-1}CA^{-1} + BD^{-1}CA^{-1} \\
&= I
\end{aligned}$$

b) Using part a), simply substitute $D' = -D^{-1}$ into the identity above which we can do because D is invertible by assumption.

$$\begin{aligned}
(A + BD'C)^{-1} &= A^{-1} + A^{-1}B(-D'^{-1} - CA^{-1}B)^{-1}CA^{-1} \\
&= A^{-1} - A^{-1}B(D'^{-1} + CA^{-1}B)^{-1}CA^{-1}
\end{aligned}$$

Exercise 3: On-line least squares with mini-batch updates. You want to solve a least squares regression problem by processing the data in small batches (mini-batches), yielding a new least squares solution after each update. assume each mini-batch contains k training examples. Group the examples in the t^{th} mini-batch into the columns of $X_t \in \mathbb{R}^{n\times k}$, and the corresponding targets into the rows of $y_t \in \mathbb{R}^k$. Let $P_{t-1} = \sum_{i=1}^{t-1} X_i X_i^T \in \mathbb{R}^{n\times n}$. Assume P_{t-1}^{-1} exists and is known. Similarly, let $s_{t-1} = \sum_{i=1}^{t-1} X_i y_i \mathbb{R}^n$. Derive the following equations for the t^{th} mini-batch update:

$$
\begin{aligned}
\hat{y}_t &\triangleq X_t^T w_{t-1}^* \text{ target prediction} \\
w_t^* &= w_{t-1}^* + P_{t-1}^{-1} X_t [I_k + X_t^T P_{t-1}^{-1} X_t]^{-1} (y_t - \hat{y}_t) \text{ update } w^* \\
P_t^{-1} &= P_{t-1}^{-1} - P_{t-1}^{-1} X_t [I_k + X_t^T P_{t-1}^{-1} X_t]^{-1} X_t^T P_{t-1}^{-1} \text{ update } P.
\end{aligned}
$$

How do these equations change if the mini-batches are not all the same size?

Answer: We start by listing the following equations that easily result from the definitions in the question:

$$
\begin{aligned}
P_t &= P_{t-1} + X_t X_t^T \\
s_t &= s_{t-1} + X_t y_t
\end{aligned}
$$

We also have the normal equation $P_t w_t^* = s_t$ which is equivalent to

$$
w_t^* = P_t^{-1}(s_{t-1} + X_t y_t)
$$

Now, using question 2b) with $A = P_{t-1}$, $B = X_t$, $C = X_t^T$, $D = I_k$. We have

$$
P_t^{-1} = (P_{t-1} + X_t X_t^T)^{-1} = P_{t-1}^{-1} - P_{t-1}^{-1} X_t [I_k + X_t^T P_{t-1}^{-1} X_t]^{-1} X_t^T P_{t-1}^{-1}
$$

Then, subbing this into our definition of w_t^*,

$$
\begin{aligned}
w_t^* &= (P_{t-1}^{-1} - P_{t-1}^{-1}X_t[I_k + X_t^T P_{t-1}^{-1}X_t]^{-1}X_t^T P_{t-1}^{-1})(s_{t-1} + X_t y_t) \\
&= (P_{t-1}^{-1} - P_{t-1}^{-1}X_t[I_k + X_t^T P_{t-1}^{-1}X_t]^{-1}X_t^T P_{t-1}^{-1})((s_{t-2} + X_{t-1}y_{t-1}) + X_t y_t) \\
&= P_{t-1}^{-1}(s_{t-2} + X_{t-1}y_{t-1}) + P_{t-1}^{-1}X_t y_t \\
&\quad - P_{t-1}^{-1}X_t[I_k + X_t^T P_{t-1}^{-1}X_t]^{-1}X_t^T P_{t-1}^{-1}(s_{t-2} + X_{t-1}y_{t-1}) \\
&\quad - P_{t-1}^{-1}X_t[I_k - (I_k + X_t^T P_{t-1}^{-1}X_t)^{-1}X_t^T P_{t-1}^{-1}X_t]y_t \\
&= w_{t-1}^* + P_{t-1}^{-1}X_t(I_k - X_t^T P_{t-1}^{-1}X_t)^{-1}X_t^T P_{t-1}X_t)y_t \\
&\quad - P_{t-1}^{-1}X_t[I_k + X_t^T P_{t-1}^{-1}X_t]^{-1}X_t^T w_{t-1}^* \\
&= w_{t-1}^* + P_{t-1}^{-1}X_t(I_k - X_t^T P_{t-1}^{-1}X_t)^{-1}(I_k + X_t^T P_{t-1}^{-1}X_t - X_t^T P_{t-1}^{-1}X_t)y_t \\
&\quad - P_{t-1}^{-1}X_t[I_k + X_t^T P_{t-1}^{-1}X_t]^{-1}\hat{y}_t \\
&= w_{t-1}^* + P_{t-1}^{-1}X_t[I_k + X_t^T P_{t-1}^{-1}X_t]^{-1}(y_t - \hat{y}_t)
\end{aligned}
$$

Exercise 4: Linear regression with vector targets. We are given training data $\{(x_i, z_i)_{i=1}^m\}$ with input examples $x_i \in \mathbb{R}^n$ and vector targets $z_i \in \mathbb{R}^d$. Place the input examples into the columns of $X \in \mathbb{R}^{n\times m}$ and the targets into the columns of $Z \in \mathbb{R}^{d\times m}$. We want to learn a linear predictor of the vector targets $z \in \mathbb{R}^d$ of test inputs $x \in \mathbb{R}^n$. To do so, first use the training data to find:

$$W^* = \underset{W\in\mathbb{R}^{n\times d}}{\arg\min} \|Y - FW\|_F^2 + \lambda\|W\|_F^2,$$

where we have set $Y = Z^T$ and $F = X^T$, and we require $\lambda \leq 0$ ($\lambda = 0$ removes the ridge regularizer).

a) Show that the above separates into d standard ridge regression problems, each solvable separately.

b) Without using the property in a), set the derivative of the objective function w.r.t. W equal to zero, and find an expression for the solution W^*. Is the separation property evident from this expression?

Answer:

a) Using the definition of the Frobenius norm

$$\begin{aligned} W^* &= \underset{W\in\mathbb{R}^{n\times d}}{\arg\min} \|Y - FW\|_F^2 + \lambda\|W\|_F^2 \\ &= \underset{W\in\mathbb{R}^{n\times d}}{\arg\min} \sum_{i=1}^{n}\sum_{j=1}^{d} (Y_{ij} - [FW]_{ij})^2 + \lambda W_{ij}^2 \end{aligned}$$

But we have that $[FW]_{ij} = F_{i:}W_{:j}$. That is, the inner product of the i^{th} row of F and the j^{th} column of W. This gives

$$\begin{aligned} W^* &= \underset{W\in\mathbb{R}^{n\times d}}{\arg\min} \sum_{j=1}^{d}\sum_{i=1}^{n} (Y_{ij} - F_{i:}W_{:j})^2 + \lambda W_{ij}^2 \\ &= \underset{W\in\mathbb{R}^{n\times d}}{\arg\min} \sum_{j=1}^{d} \|Y_{:j} - FW_{:j}\|_2^2 + \lambda\|W_{:j}\|_2^2 \end{aligned}$$

That is, by splitting up the summation along the columns of W and Y, you can turn the original problem into d separate standard ridge regression.

b) Taking the derivative yields

$$\nabla_W = 2F^T(FW - Y) + \lambda 2W = 0$$
$$\implies W^* = (F^TF + \lambda I_n)^{-1}F^TY$$

The separation property is evident from this equation because the entries of W^*, say W^*_{ij}, only requires the column of $Y_{:j}$ to perform the calculation for all $i \in [n]$ and fixed j.

Exercise 5: The softmax function. This function maps $x \in \mathbb{R}^n$ to a probability mass function $s(x)$ on n outcomes. It can be written as the composition of two functions $s(x) = q(p(x))$, where $p : \mathbb{R}^n \to \mathbb{R}^n_+$ and $q : \mathbb{R}^n_+ \to \mathbb{R}^n_+$ are defined by

$$p(x) = [e^{x_i}] \quad q(z) = z/(\mathbf{1}^T z)$$

Here $\mathbb{R}^n_+$ denotes the positive cone $\{x \in \mathbb{R}^n : x_i > 0\}$. The function $p(\cdot)$ maps $x \in \mathbb{R}^n$ into the positive cone $\mathbb{R}^n_+$, and for $z \in \mathbb{R}^n_+$, $q(\cdot)$ normalizes z to a probability mass function in $\mathbb{R}^n_+$.

a) Determine the derivative of $p(x)$ at x.

b) Determine the derivative of $q(z)$ at z.

c) Determine the derivative of the softmax function at x.

Answer:

a) The i^{th} partial derivative of $p(x)$ is $[\nabla p(x)]_i = e^{x_i}$ and so $\nabla p(x) = p(x)$ and so the derivative is

$$Dp(x)(v) = p(x) \otimes v \in \mathbb{R}^n$$

$$\text{or } Dp(x)(v) = \begin{bmatrix} e^{x_1} & 0 & \cdots & 0 \\ 0 & e^{x_2} & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & e^{x_n} \end{bmatrix} v$$

b) We have that the i^{th} component of $q(z)$ is $[q(z)]_i = \frac{z_i}{\sum_{j=1}^n z_j}$. Thus, the partial derivatives are

$$\frac{\partial[q(z)]_i}{\partial z_j} = \frac{\sum_{j=1}^n z_j - z_i}{(\sum_{j=1}^n z_j)^2} \text{ if } i = j$$

$$\frac{\partial[q(z)]_i}{\partial z_j} = \frac{-z_i}{(\sum_{j=1}^n z_j)^2} \text{ if } i \neq j$$

$$\implies Dg(z)(v) = \begin{bmatrix} \frac{\sum_{j=1}^n z_j - z_1}{(\sum_{j=1}^n z_j)^2} & \frac{-z_1}{(\sum_{j=1}^n z_j)^2} & \cdots & \frac{-z_1}{(\sum_{j=1}^n z_j)^2} \\ \frac{-z_2}{(\sum_{j=1}^n z_j)^2} & \frac{\sum_{j=1}^n z_j - z_2}{(\sum_{j=1}^n z_j)^2} & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{-z_n}{(\sum_{j=1}^n z_j)^2} & \cdots & \cdots & \frac{\sum_{j=1}^n z_j - z_n}{(\sum_{j=1}^n z_j)^2} \end{bmatrix} v$$

c) Applying the chain rule

$$Ds(x)(v) = Dg(p(x)) \circ Dp(x)(v)$$

$$= \begin{bmatrix} \frac{\sum_{j=1}^n e^{x_j} - e^{x_1}}{(\sum_{j=1}^n e^{x_j})^2} & \frac{-e^{x_1}}{(\sum_{j=1}^n e^{x_j})^2} & \cdots & \frac{-e^{x_1}}{(\sum_{j=1}^n e^{x_j})^2} \\ \frac{-e^{x_2}}{(\sum_{j=1}^n e^{x_j})^2} & \frac{\sum_{j=1}^n e^{x_j} - e^{x_2}}{(\sum_{j=1}^n e^{x_j})^2} & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{-e^{x_n}}{(\sum_{j=1}^n e^{x_j})^2} & \cdots & \cdots & \frac{\sum_{j=1}^n e^{x_j} - e^{x_n}}{(\sum_{j=1}^n e^{x_j})^2} \end{bmatrix} \begin{bmatrix} e^{x_1} & 0 & \cdots & 0 \\ 0 & e^{x_2} & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & e^{x_n} \end{bmatrix} v$$

$$= \begin{bmatrix} e^{x_1} \frac{\sum_{j=1}^n e^{x_j} - e^{x_1 + x_2}}{(\sum_{j=1}^n e^{x_j})^2} & \frac{-e^{x_1 + x_2}}{(\sum_{j=1}^n e^{x_j})^2} & \cdots & \frac{-e^{x_1 + x_n}}{(\sum_{j=1}^n e^{x_j})^2} \\ \frac{-e^{x_2 + x_1}}{(\sum_{j=1}^n e^{x_j})^2} & e^{x_2} \frac{\sum_{j=1}^n e^{x_j} - e^{x_2}}{(\sum_{j=1}^n e^{x_j})^2} & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{-e^{x_n + x_1}}{(\sum_{j=1}^n e^{x_j})^2} & \cdots & \cdots & e^{x_n} \frac{\sum_{j=1}^n e^{x_j} - e^{x_n}}{(\sum_{j=1}^n e^{x_j})^2} \end{bmatrix} v$$