98101469 میلاد حیدری

تمرین سری 2

سوال ۱-آ) بسط تیلور تابع را حول w^* می نویسیم. فرض کنیم w^* نقطه ای است که gradient desent به آن رسیده است:

$$J(w) = J(w^*) + \underbrace{\nabla J(w^*)^T (w - w^*)}_{0} + \frac{1}{2}(w-w^*)^T H (w-w^*)$$

حال فرض کنیم $\tilde{w}$ همان w^* است که فقط یکی از درایه هایش صفر شده است. فرض کنیم درایه k ام صفر شده:

$$w^* = \begin{bmatrix} w_1^* \\ \vdots \\ w_n^* \end{bmatrix} \longrightarrow \tilde{w} = \begin{bmatrix} w_1^* \\ \vdots \\ 0 \\ \vdots \\ w_n^* \end{bmatrix} \xleftarrow{\text{درایه } k \text{ ام}}$$

حال $J(\tilde{w})$ را حساب کنیم:

$$J(\tilde{w}) = J(w^*) + \frac{1}{2}(\tilde{w}-w^*)^T H(\tilde{w}-w^*)$$

$\tilde{w}-w^*$ برداری است که فقط درایه k ام آن ناصفر و برابر w_k^* است. درنتیجه اگر H به صورت زیر باشد:

$$H = [h_1, \dots, h_n] \longrightarrow H(\tilde{w}-w^*) = -w_k^* h_k \Rightarrow (\tilde{w}-w^*)^T(-w_k^* h_k)$$

$$= w_k^{*2} H_{KK} \Rightarrow J(\tilde{w}) = J(w^*) + H_{KK} w_k^{*2}$$

یعنی اگر w_k^* را حذف کنیم، تابع هزینه به اندازه $w_k^{*2} H_{KK}$ زیاد می شود. درنتیجه باید درایه ای را انتخاب کنیم که مقدار $w_k^{*2} H_{KK}$ کمترین باشد. توجه شود که فرض می کنیم الگوریتم به نقطه بهینه محلی local minimum رسیده است درنتیجه $H_{KK} \geq 0$ است یعنی هزینه بعد از حذف w_k^* افزایش می یابد.

ب) اگر H همانی باشد، $H_{ii} = 1$ برای همه i ها درنتیجه خواهیم داشت:

$$J(\tilde{w}) = J(w^*) + w_k^{*2}$$

درنتیجه باید درایه ای را حذف کنیم که کمترین اندازه را دارد. برای این کار اگر بخواهیم مثلا k درایه حذف کنیم، k درایه ای که کوچکترین اندازه را دارند را حذف می کنیم.

سوال 2 -

$$X = \begin{bmatrix} \text{—} & x_1^T & \text{—} \\ & \vdots & \\ \text{—} & x_N^T & \text{—} \end{bmatrix} \quad , \quad x_i \in \mathbb{R}^d \qquad y_i = b^T x_i + \varepsilon_i \quad , \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2)$$

آ) حدسی که ما از y می‌زنیم به صورت $\hat{y} = Xb$ خواهد بود. درنتیجه:

$$C = \| y - \hat{y} \|_2^2 = \| Xb - y \|_2^2 \Rightarrow \hat{b} = \underset{b}{\operatorname{argmin}} \| Xb - y \|_2^2$$

می‌دانیم که جواب این مسئله از pseudo inverse ماتریس X بدست می‌آید:

$$\hat{b} = X^{+} y$$

حال چون X رتبه کامل ستونی است ($N > d$) درنتیجه:

$$\hat{b} = (X^T X)^{-1} X^T y$$

دلیل معکوس پذیر بودن $X^T X$ این است که فضای سطری X و $X^T X$ برابر است و فضای سطری X هم دارای رنک d است. چون $X^T X$ ماتریس $d \times d$ است و فضای سطری آن دارای d پایه است یعنی رتبه کامل است و معکوس پذیر می‌باشد.

ب) اگر $A = X(X^T X)^{-1} X^T$ باشد:

$$A^T = X (X^T X)^{-1} X^T$$

$$A^2 = X \underbrace{(X^T X)^{-1} X^T X}_{I} (X^T X)^{-1} X^T = X (X^T X)^{-1} X^T$$

درنتیجه $A^T = A$ و A متقارن است.

درنتیجه $A^2 = A$ است و A متقارن است. چون A متقارن است، می‌توان تجزیه به مقادیر ویژه انجام داد:

$$A = Q \Lambda Q^T \longrightarrow A^2 = Q \Lambda^2 Q^T$$

درنتیجه مقادیر ویژه A^2، توان ۲ مقادیر ویژه A هستند، اگر آن‌ها را با λ نشان دهیم:

$$\lambda_i = \lambda_i^2 \longrightarrow \lambda_i (\lambda_i - 1) = 0 \longrightarrow \begin{cases} \lambda_i = 0 \\ \lambda_i = 1 \end{cases}$$

$A = X(X^TX)^{-1}X^T$ یک ماتریس $N \times N$ است. ثابت می‌کنیم فضای سطری آن با فضای ستونی X^T برابر است.

هر $x \neq 0$ را اگر در نظر بگیریم که $X^Tx = 0$ باشد:

$$Ax = X(X^TX)^{-1}X^Tx = 0 \longrightarrow \text{Null space of } X^T \subseteq \text{Null space of } A$$

برعکس هر x ای که $Ax = 0$ باشد، $x \neq 0$ داریم:

$$Ax = 0 \longrightarrow X(X^TX)^{-1}X^Tx = 0$$

چون $(X^TX)^{-1}$ رتبه کامل است هیچ $y \neq 0$ ای موجود نیست که $(X^TX)^{-1}y = 0$ باشد. همچنین چون X رتبه کامل ستونی است، هیچ $y \neq 0$ ای موجود نیست که $Xy = 0$ باشد. در نتیجه حتما $X^Tx = 0$ است.

در نتیجه، $N(A) \subseteq N(X^T)$. از این در نتیجه می‌گیریم فضای سطری X با فضای ستونی A برابر است. در نتیجه rank ماتریس A برابر d است. در نتیجه A، d مقدار ویژه ۱ و $N-d$ مقدار ویژه برابر با ۰ دارد. ماتریس $A-I$ نیز متقارن است و d مقدار ویژه ۰ و $N-d$ مقدار ویژه ۱- دارد.

Scanned with CamScanner

$$\mathbb{E}\left[\frac{1}{N}\|(I-A)y\|_2^2\right] = \frac{1}{N}\mathbb{E}\left[\|(I-A)y\|_2^2\right]$$ درنتیجه :

اگر تجزیه به مقادیر ویژه را برای $I-A$ انجام دهیم :

$$\Rightarrow \frac{1}{N}\mathbb{E}\left[\|Q\Lambda Q^T y\|_2^2\right] = \frac{1}{N}\mathbb{E}\left[\left\|\sum_{i=1}^{N-d} q_i q_i^T y\right\|_2^2\right]$$

چون Q ماتریس unitary است ستون های آن متعامد یکه هستند، داریم :

$$\Rightarrow \frac{1}{N}\mathbb{E}\left[\sum_{i=1}^{N-d}(q_i^T y)^2\right] = \frac{1}{N}\mathbb{E}\left[\sum_{i=1}^{N-d}(y^T q_i)^2\right]$$

q_i بردار ویژه متناظر مقدار ویژه 1 در $I-A$ است یعنی :

$$(I-A)q_i = q_i \longrightarrow y^T q_i = q_i^T y = q_i^T(I-A)y = q_i^T(y-\underbrace{Ay}_{\hat{y}}) = q_i^T\varepsilon$$

$$\text{where } \varepsilon = \begin{bmatrix}\varepsilon_1\\ \vdots \\ \varepsilon_n\end{bmatrix} \Rightarrow \quad \Rightarrow \mathbb{E}\left[(q_i^T\varepsilon)^2\right] = \mathbb{E}\left[\left(\sum_j q_{ij}\varepsilon_j\right)^2\right] = \sum_j q_{ij}^2\,\underbrace{\mathbb{E}[\varepsilon_j]^2}_{\sigma^2}$$

$$= \sigma^2\underbrace{\sum_j q_{ij}^2}_{1} = \sigma^2$$

درنتیجه :

$$\frac{1}{N}\mathbb{E}\left[\|(I-A)y\|_2^2\right] = \frac{1}{N}(N-d)\sigma^2 = \frac{N-d}{N}\sigma^2$$

ج) با توجه به رابطه‌ای که بالا بدست آمد، در صورت رابطه $N-d$ موجود است. اگر N به d نزدیک باشد خطای training بسیار کم می‌شود. درنتیجه باید تعداد example ها بیشتر از تعداد feature ها باشد که overfitting رخ ندهد.

سوال ۴ - آ)

$$J=\left(y_d-\sum_{k=1}^{n}\delta_k w_k x_k\right)^2 , \quad \delta_k\sim N(1,\sigma^2)$$

$$\frac{\partial J}{\partial w_j}=2\delta_j x_j\left(\sum_{k=1}^{n}\delta_k w_k x_k-y_d\right)=-2\delta_j x_j y_d+2\delta_j x_j\sum_{k=1}^{n}\delta_k w_k x_k$$

$$=-2\delta_j x_j y_d+2\delta_j^2 x_j^2 w_j+2\delta_j x_j\sum_{\substack{k=1\\k\neq j}}^{n}\delta_k w_k x_k$$

$$\Rightarrow \mathbb{E}\left[\frac{\partial J}{\partial w_j}\right]=-2x_j y_d+2x_j^2 w_j\,\mathbb{E}[\delta_j^2]+2x_j\sum_{\substack{k=1\\k\neq j}}^{n}\mathbb{E}[\delta_j\delta_k]\,w_k x_k$$

چون δ_j و δ_k مستقل فرض می‌شوند، $\mathbb{E}[\delta_j\delta_k]=\mathbb{E}[\delta_j]\,\mathbb{E}[\delta_k]=1$ ، همچنین $\mathbb{E}[\delta_j]^2=\sigma^2+1$

در نتیجه،

$$\mathbb{E}\left[\frac{\partial J}{\partial w_j}\right]=-2x_j y_d+2x_j^2 w_j\sigma^2+2x_j\sum_{\substack{k=1\\k\neq j}}^{n}w_k x_k+2x_j^2 w_j$$

ب) اگر تابع هزینه بدون dropout را محاسبه کنیم،

$$\tilde{J}=\left(y_d-\sum_{k=1}^{n}w_k x_k\right)^2\Rightarrow\frac{\partial\tilde{J}}{\partial w_j}=2x_j\left(\sum_{k=1}^{n}w_k x_k-y_d\right)=-2x_j y_d+2x_j\sum_{k=1}^{n}w_k x_k$$

$$=-2x_j y_d+2x_j^2 w_j+2x_j\sum_{\substack{k=1\\k\neq j}}^{n}w_k x_k$$

در نتیجه،

$$\frac{\partial J}{\partial w_j}=\frac{\partial\tilde{J}}{\partial w_j}+2x_j^2\sigma^2 w_j\longrightarrow J=\tilde{J}+\sum_{j=1}^{n}2x_j^2\sigma^2 w_j^2$$

مشاهده می‌شود که یک عبارت به فرم $\sum k w_j^2$ به تابع هزینه اضافه شده است، این مانند یک regularization نرم 2 (L_2) عمل می‌کند و باعث جلوگیری از overfitting می‌شود.

سوال 5 -

$$J = w^T H w \quad , \quad H = Q\Lambda Q^T$$

الف)

$$\nabla J = 2Hw \longrightarrow w^t = w^{t-1} - \varepsilon \nabla J(w^{t-1}) \Rightarrow w^t = w^{t-1} - \varepsilon\, 2Hw^{t-1}$$

$$\Rightarrow w^t = w^{t-1} - 2\varepsilon H w^{t-1}$$

ب)

$$w^t = (I-2\varepsilon H)w^{t-1} \longrightarrow w^1 = (I-2\varepsilon H)w^0 \longrightarrow w^t = (I-2\varepsilon H)^t w^0$$

ج) با توجه به اینکه H متقارن است، $I-2\varepsilon H$ نیز متقارن است. اگر مقادیر ویژه ماتریس H، چنانچه $\{\lambda_i\}_i$ باشند، مقادیر ویژه $I-2\varepsilon H$ برابر $1-2\varepsilon\lambda_i$ هستند. اگر A یک ماتریس متقارن باشد که به فرم $A=Q\Lambda Q^T$ نوشته باشیم، $A^n = Q\Lambda^n Q^T$ است. حال اگر همه مقادیر ویژه ماتریس A اندازه کمتر از 1 داشته باشند، A^n به یک ماتریس همگرا می‌شود. درنتیجه شرط همگرایی بالا این است که $|1-2\varepsilon\lambda_i| \leq 1$ باشد.

د) الگوریتم نیوتن بسط مرتبه 2 را حساب می‌کند و براساس آن نقطه بعدی را انتخاب می‌کند:

$$J(w) = J(w^{t-1}) + \nabla J(w^{t-1})^T (w-w^{t-1}) + \frac{1}{2}(w-w^{t-1})^T \tilde{H} (w-w^{t-1})$$

$$\nabla J(w) = \nabla J(w^{t-1}) + \tilde{H}(w-w^{t-1}) = 0 \longrightarrow \tilde{H}w - \tilde{H}w^{t-1} = -\nabla J(w^{t-1})$$

$$\Rightarrow \tilde{H}w - \tilde{H}w^{t-1} = -\nabla J(w^{t-1}) \longrightarrow w = w^{t-1} - \tilde{H}^{-1}\nabla J(w^{t-1})$$

که $\tilde{H}$ ماتریس هسین J است. با توجه به تابع هزینه بالا:

$$\nabla J(w^{t-1}) = 2Hw^{t-1}, \quad \tilde{H} = 2H$$

درنتیجه در الگوریتم نیوتن خواهیم داشت:

$$w^t = w^{t-1} - 2^{-1}H^{-1}2Hw^{t-1} = w^{t-1} - w^{t-1} = 0 \;!$$

اگر از نقطه دلخواه شروع کنیم:

$$w^1 = w^0 - w^0 = 0$$

درنتیجه الگوریتم نیوتن در یک گام جواب مسئله را پیدا می‌کند. با توجه به اینکه تابع هزینه یک تابع quadratic است، بسط مرتبه 2 تابع همان خود تابع است درنتیجه الگوریتم نیوتن به صورت تحلیلی در یک گام، مینیمم را پیدا می‌کند (فرض براین است که تابع convex باشد یعنی $H \succeq 0$).

ه) الگوریتم نیوتن نیازمند محاسبه ماتریس Hessian، معکوس کردن آن دارد که معکوس گرفتن از ماتریس از $O(n^3)$ است اگر ماتریس $n \times n$ باشد. درنتیجه هزینه محاسباتی را درصورتی که n بزرگ باشد، بسیار بالا می‌برد.

سوال 3- آ)

$$Y = X\beta + \varepsilon$$

تخمین ما از Y به صورت $\hat{Y} = X\hat{\beta}$ است. درنتیجه اگر بخواهیم خطا را کمینه کنیم:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \| Y - \hat{Y} \|_2^2 = \underset{\beta}{\operatorname{argmin}} \| X\beta - Y \|_2^2$$

با فرض اینکه X، رتبه کامل ستونی باشد (که در اکثر مواردی که تعداد نمونه‌ها بسیار بیشتر از feature ها باشد همین است) خواهیم داشت:

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

ب)

$$J = \| X\beta - Y \|_2^2 + \lambda \| \beta \|_2^2$$

این عبارت را می‌توان به صورت زیر نیز نوشت:

$$J = \left\| \begin{bmatrix} X \\ \sqrt{\lambda} I \end{bmatrix} \beta - \begin{bmatrix} Y \\ 0 \end{bmatrix} \right\|_2^2$$

با استفاده از حداقل مربعات، جواب بهینه برای β برابر خواهد بود با:

$$\hat{\beta} = \left(\begin{bmatrix} X^T & \sqrt{\lambda} I \end{bmatrix} \begin{bmatrix} X \\ \sqrt{\lambda} I \end{bmatrix} \right)^{-1} \begin{bmatrix} X^T & \sqrt{\lambda} I \end{bmatrix} \begin{bmatrix} Y \\ 0 \end{bmatrix} = (X^T X + \lambda I)^{-1} X^T Y$$

سوال 3 _ د)

با توجه به بخش ب) اگر $J = \|X\beta - Y\|_2^2 + \lambda\|\beta\|_2^2$ باشد، می توان

آن را به صورت زیر نوشت:

$$J = \left\| \underbrace{\begin{bmatrix} X \\ \sqrt{\lambda} I \end{bmatrix}}_{\tilde{X}} \beta - \underbrace{\begin{bmatrix} Y \\ 0 \end{bmatrix}}_{\tilde{Y}} \right\|_2^2 = \|\tilde{X}\beta - \tilde{Y}\|_2^2$$

با توجه به این فرم می توان در نظر گرفت

که $\tilde{X}$ همان ماتریس داده ها است که تعدادی داده به آن اضافه شده، $\tilde{Y}$ هم

بردار label ها است که تعدادی داده به آن اضافه شده است.

داده های اضافه شده در $\tilde{X}$ به فرم $\sqrt{\lambda} e_i$ هستند که e_i ستون i ام ماتریس

همانی است و label متناظر آن ها هم $\underline{0}$ می باشد. در نتیجه اگر

تابع هزینه ما به فرم روبرو باشد، می توان گفت که هم شکل این است که تعدادی

داده اضافه شده و $\lambda_2\|\beta\|_1$ نیز نشان دهنده L1 regularization است:

$$L(\beta, \lambda_1, \lambda_2) = \|y - X\beta\|_2^2 + \lambda_1\|\beta\|_2^2 + \lambda_2\|\beta\|_1$$

سوال ۶ - الف) ترم $\|W\|_F^2$ موجود که regularization است، باعث می‌شود که مدل ما overfit نکند. ترم $\|h_1-h_2\|_2^2$ نیز سعی می‌کند که تا حد امکان خروجی ها شبیه هم باشند. در واقع تبدیل $=\tanh(Wx+b)$ یک تبدیل غیرخطی است که x ها را به h ها map می‌کند، دوست داریم در فضایی که x ها به آن map می‌شوند، تا حد امکان به هم نزدیک باشند. این network ، تبدیل را یاد می‌گیرد. در واقع زیرفضای h را پیدا می‌کند. از این کار می‌توان برای فشرده سازی استفاده کرد. در واقع ورودی $x \in \mathbb{R}^n$ است، خروجی $h \in \mathbb{R}^m$

اگر $n >> m$ باشد، این مدل عمل فشرده سازی را انجام می‌دهد.

ب)

$$x_1, x_2 \in \mathbb{R}^n \qquad h_1 = \tanh(Wx_1+b) \quad , \quad h_2 = \tanh(Wx_2+b)$$

$$J = \|h_1-h_2\|_2^2 + \|W\|_F^2$$

$$\frac{\partial J}{\partial b} = \frac{\partial \|h_1-h_2\|_2^2}{\partial b} = \underbrace{\frac{\partial \|h_1-h_2\|_2^2}{\partial (h_1-h_2)}}_{2(h_1-h_2)} \cdot \frac{\partial (h_1-h_2)}{\partial b} = 2(h_1-h_2)\frac{\partial(h_1-h_2)}{\partial b}$$

$$\frac{\partial(h_1-h_2)}{\partial b} = \frac{\partial h_1}{\partial b} - \frac{\partial h_2}{\partial b} \Rightarrow \frac{\partial h_1}{\partial b} = \frac{\partial \tanh(Wx_1+b)}{\partial b} = \underbrace{\frac{\partial \tanh(Wx_1+b)}{\partial (Wx_1+b)}}_{diag(\tanh'(Wx_1+b))} \cdot \underbrace{\frac{\partial W_1x+b}{\partial b}}_{I}$$

$$\Rightarrow \frac{\partial(h_1-h_2)}{\partial b} = \odot \tanh'(Wx_1+b) - \odot \tanh'(Wx_2+b)$$

$$\Rightarrow \frac{\partial J}{\partial b} = 2(h_1-h_2)^T\left[\odot \tanh'(Wx_1+b) - \odot \tanh'(Wx_2+b)\right]$$

$$\frac{\partial J}{\partial W} = \frac{\partial \|h_1-h_2\|_2^2}{\partial W} + \underbrace{\frac{\partial \|W\|_F^2}{\partial W}}_{2W} \Rightarrow \frac{\partial \|h_1-h_2\|_2^2}{\partial W} = \underbrace{\frac{\partial \|h_1-h_2\|_2^2}{\partial (h_1-h_2)}}_{2(h_1-h_2)^T} \cdot \frac{\partial(h_1-h_2)}{\partial W}$$

$$\Rightarrow 2(h_1-h_2)^T \cdot \frac{\partial h_1}{\partial W} - 2(h_1-h_2)^T \cdot \frac{\partial h_2}{\partial W}$$

$$\Rightarrow 2(h_1-h_2)^T \odot \tanh'(Wx_1+b) \cdot \frac{\partial(Wx_1+b)}{\partial W} - 2(h_1-h_2)^T \odot \tanh'(Wx_2+b) \cdot \frac{\partial(Wx_2+b)}{\partial W}$$

$$\Rightarrow \left(\frac{\partial \|h_1-h_2\|_2^2}{\partial W}\right)_{ij} = \delta_i^1 x_j - \delta_i^2 x_j \quad \text{where} \begin{cases} \delta_i^1 = \left[2(h_1-h_2)^T \odot \tanh'(Wx_1+b)\right]_i \\ \delta_i^2 = \left[2(h_1-h_2)^T \odot \tanh'(Wx_2+b)\right]_i \end{cases}$$

$$\Rightarrow \frac{\partial \|h_1-h_2\|_2^2}{\partial W} = (\delta^1)^T x^T - (\delta^2)^T x^T \quad \text{where} \begin{cases} \delta^1 = 2(h_1-h_2)^T \odot \tanh'(Wx_1+b) \\ \delta^2 = 2(h_1-h_2)^T \odot \tanh'(Wx_2+b) \end{cases}$$

$$= \left(\left[2(h_1-h_2)^T \odot \tanh'(Wx_1+b)\right]^T - \left[2(h_1-h_2)^T \odot \tanh'(Wx_2+b)\right]^T\right) x^T$$

الگوریتم آپدیت به صورت زیر می شود: (فرض کنیم batch size برابر K باشد) :

$$b^t = b^{t-1} - \frac{\varepsilon^t}{K}\sum_{i=1}^{K} \frac{\partial \|h_1^i - h_2^i\|_2^2}{\partial b}$$

$$W^t = W^{t-1} - \frac{\varepsilon^t}{K}\sum_{i=1}^{K} \frac{\partial \|h_1^i - h_2^i\|_2^2}{\partial W}$$