CSE 6740 - HW 3

Manvitha Kalicheti
gtID: 903838438

November 10, 2022

Collaborators

Keerthan Ramnath, Aishwarya Vijaykumar Sheelvant

References

Pattern Recognition and Machine Learning by Christopher Bishop, Class slides

1

(a)

$$\begin{aligned}
E[\hat{\theta}] &= E[(X^TX)^{-1}X^TY] \\
&= E[(X^TX)^{-1}X^T(\theta^TX+\epsilon)] \\
&= E[(X^TX)^{-1}X^T\theta^TX] \text{ (as } E[\epsilon]=0) \\
&= (X^TX)^{-1}X^TX\theta \\
E[\hat{\theta}] &= \theta
\end{aligned}$$

(b)

$$\begin{aligned}
Var[\hat{\theta}] &= Var[(X^TX)^{-1}X^TY] \\
&= (X^TX)^{-1}X^TVar[Y]((X^TX)^{-1}X^T)^T \\
&= (X^TX)^{-1}X^TX(XX^T)^{-1}\sigma^2 \text{ (as } Var[Y]=Var[\theta^TX+\epsilon]=Var[\epsilon]=0) \\
Var[\hat{\theta}] &= \sigma^2(XX^T)^{-1}
\end{aligned}$$

(c)

Yes. We know that $\hat{\theta} = (X^TX)^{-1}X^T(\theta^TX+\epsilon)$. $\hat{\theta}$ is thus a linear combination of θ and ϵ, and θ is a constant. As ϵ follows a Gaussian distribution, $\hat{\theta}$ also follows a Gaussian Distribution with mean θ and variance $\sigma^2(XX^T)^{-1}$ as shown in 1(a) and 1(b).

(d)

1.

False. The least squares cost function in linear regression is always convex. In convex functions, gradient descent converges to a global minimum.

2.

True. At global minimum, the derivative is 0. Therefore, gradient descent will not change θ.

3.

False. A large learning rate could cause an overshoot and oscillate near the minimum, thus increasing the number of steps required to reach the minimum.

2

(a)

$$\begin{aligned} \text{Prior: } & p(\theta) = \mathcal{N}(\theta|0, I\tau^2) \\ \text{Likelihood: } & p(y|\theta) = \mathcal{N}(y|X\theta, I\sigma^2) \\ \text{Normalising constant: } & p(y) = K \\ \text{Baye's Rule: } & p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \end{aligned}$$

$$\begin{aligned} p(\theta|y) &= \frac{1}{K}\mathcal{N}(y|X\theta, I\sigma^2)\mathcal{N}(\theta|0, I\tau^2) \\ &= \frac{1}{K}\left(\frac{1}{\sqrt{(2\pi)^n\tau^2}}exp\left(-\frac{\theta^T\theta}{2\tau^2}\right)\frac{1}{\sqrt{(2\pi)^n\sigma^2}}exp\left(-\frac{1}{2\sigma^2}(y-X\theta)^T(y-X\theta)\right)\right) \\ LL = \log(p(\theta|y)) &= -\log K - \frac{1}{2}\log((2\pi)^n\sigma^2) - \frac{1}{2}\log((2\pi)^n\tau^2) - \frac{1}{2\sigma^2}(y-X\theta)^T(y-X\theta) - \frac{1}{2\tau^2}\theta^T\theta \end{aligned}$$

$$\frac{\partial LL}{\partial \theta} = \frac{1}{-2}\left(\frac{2(-X^T)(y-X\theta)}{2} + \frac{2\theta}{\tau^2}\right)$$

Setting this to 0:

$$\begin{aligned} \theta\sigma^2 &= X^T(y-X\theta)\tau^2 \\ \theta\sigma^2 &= X^Ty\tau^2 - X^TX\theta\tau^2 \\ \theta &= \left(\frac{\sigma^2}{\tau^2}I + X^TX\right)^{-1}X^Ty \end{aligned}$$

Ridge regression estimate $= (X^T + \lambda I)^{-1}X^Ty$
Comparing above two expressions, $\lambda = \frac{\sigma^2}{\tau^2}$

Posterior: $P(\theta|y) = \mathcal{N}(\mu, \Sigma)$ (as it is a product of two Gaussians)
The exponent part of $p(\theta|y) = -\frac{1}{2}(\theta - \mu)^T\Sigma^{-1}(\theta - \mu)$.
The exponent part of $p(y|\theta)p(\theta) = -\frac{1}{2\sigma^2}(y - X\theta)^T(y - X\theta) - \frac{\theta^T\theta}{2\tau^2}$.
Both are equivalent, so we compare terms.

$$\theta^T\Sigma^{-1}\theta - \theta^T\Sigma^{-1}\mu - \mu^T\Sigma^{-1}\theta + \mu^T\Sigma^{-1}\mu = \frac{y^ty - \theta^TX^Ty - y^TX\theta + \theta^TX^TX\theta}{\sigma^2} + \frac{\theta^T\theta}{\tau^2}$$

Comparing second order θ terms,

$$\begin{aligned}\theta^T\Sigma^{-1}\theta &= \frac{\theta^TX^TX\theta}{\sigma^2} + \frac{\theta^T\theta}{\tau^2} \\ &= \theta^T\frac{1}{\sigma^2}\left(X^TX + \frac{\sigma^2}{\tau^2}I\right)\theta \\ \Sigma^{-1} &= \frac{1}{\sigma^2}\left(X^TX + \frac{\sigma^2}{\tau^2}I\right)\end{aligned}$$

Comparing first order θ terms,

$$\begin{aligned}\theta^T\Sigma^{-1}\mu &= \frac{\theta^TX^Ty}{\sigma^2} \\ \Sigma^{-1}\mu &= \frac{X^Ty}{\sigma^2} \\ \mu &= \Sigma\frac{X^Ty}{\sigma^2} \\ \mu &= \left(X^TX + \frac{\sigma^2}{\tau^2}I\right)^{-1}X^Ty\end{aligned}$$

Clearly, $\mu \equiv$ ridge regression estimate.

(b)

K fold cross-validation procedure:

$\longrightarrow$ Split the dataset {1,...,n} into K subsets of roughly equal size, $F_1, ..., F_k$.

$\longrightarrow$ For each of these subsets, train on all data points not in the considered subset, and validate on the data points in the considered subset. For each value of the regularization parameter $\lambda \in \{\lambda_1, ..., \lambda_n\}$, find the estimate $\hat{f}_\lambda^{-k}$ on the training set, and find total error from the validation set using $e_k(\lambda) = \sum_{i\in F_k}(y_i - \hat{f}_\lambda^{-k}(x_i))^2$.

$\longrightarrow$ For each regularization parameter value λ, find the average error over all subsets: $CV(\lambda) = \frac{1}{n}\sum_{k=1}^{K} e_k(\lambda) = \frac{1}{n}\sum_{k=1}^{K}\sum_{i\in F_k}(y_i - \hat{f}_\lambda^{-k}(x_i))^2$

$\longrightarrow$ Pick the λ with the least average error.

3.1

(a) Joint Bayes

(i)

$$
\begin{aligned}
P(K=1|a=1\wedge b=1\wedge c=0) &= \frac{P(a=1\wedge b=1\wedge c=0|K=1)P(K=1)}{P(a=1\wedge b=1\wedge c=0)}\\
&= \frac{\frac{1}{5}\frac{5}{9}}{\frac{2}{9}}\\
&= \boxed{\frac{1}{2}}
\end{aligned}
$$

(ii)

$$
\begin{aligned}
P(K=0|a=1\wedge b=1) &= \frac{P(a=1\wedge b=1|K=0)P(K=0)}{P(a=1\wedge b=1)}\\
&= \frac{\frac{1}{4}\frac{4}{9}}{\frac{3}{9}}\\
&= \boxed{\frac{1}{3}}
\end{aligned}
$$

(b) Naive Bayes

(i)

$P(K=1|a=1\wedge b=1\wedge c=0)$

$$
\begin{aligned}
&= \frac{P(K=1)P(a=1|K=1)P(b=1|K=1)P(c=0|K=1)}{P(K=1)P(a=1|K=1)P(b=1|K=1)P(c=0|K=1)+P(K=0)P(a=1|K=0)P(b=1|K=0)P(c=0|K=0)}\\
&= \frac{\frac{5}{9}\frac{4}{5}\frac{2}{5}\frac{2}{5}}{\frac{5}{9}\frac{4}{5}\frac{2}{5}\frac{2}{5}+\frac{4}{9}\frac{1}{4}\frac{3}{4}\frac{3}{4}}\\
&= \boxed{\frac{256}{481}=0.5322}
\end{aligned}
$$

(ii)

$$
\begin{aligned}
P(K=0|a=1\wedge b=1) &= \frac{P(K=0)P(a=1|K=0)P(b=1|K=0)}{P(K=0)P(a=1|K=0)P(b=1|K=0)+P(K=1)P(a=1|K=1)P(b=1|K=1)}\\
&= \frac{\frac{4}{9}\frac{1}{4}\frac{3}{4}}{\frac{4}{9}\frac{1}{4}\frac{3}{4}+\frac{5}{9}\frac{4}{5}\frac{2}{5}}\\
&= \boxed{\frac{15}{47}=0.3191}
\end{aligned}
$$

3.2

(a)

$$f(x) = sign\left(\log\left(\frac{p(x|y=1)p(y=1)}{p(x|y=-1)p(y=-1)}\right)\right)$$
$$f(x) = sign(h(X))$$
$$Gaussian \implies P(X=x|Y=y) = \frac{1}{\sqrt{2\pi^d det\Sigma_y}} exp\left(-\frac{1}{2}(x-\mu_y)^T\Sigma_y^{-1}(x-\mu_y)\right)$$

$$h(X) = \log\left(\frac{\frac{1}{\sqrt{2\pi^d det\Sigma_y}} exp\left(-\frac{1}{2}(x-\mu_y)^T\Sigma_y^{-1}(x-\mu_y)\right)}{\frac{1}{\sqrt{2\pi^d det\Sigma_y}} exp\left(-\frac{1}{2}(x-\mu_y)^T\Sigma_y^{-1}(x-\mu_y)\right)}\right) + \log\left(\frac{p(y=1)}{p(y=-1)}\right)$$
$$h(X) = \frac{1}{2}\log\left(\frac{\Sigma_{-1}}{\Sigma_1}\right) + \log\left(\frac{p(y=1)}{p(y=-1)}\right) - \frac{1}{2}(X-\mu_1)^T\Sigma_1^{-1}(X-\mu_1) + \frac{1}{2}(X-\mu_{-1})^T\Sigma_{-1}^{-1}(X-\mu_{-1})$$

If $\Sigma_1^{-1} \neq \Sigma_{-1}^{-1}$, we will have quadratic terms in X in $h(X) \implies$ shape of the decision boundary is a quadratically curved surface in d dimensions.

(b)

$$\Sigma_1^{-1} = \Sigma_{-1}^{-1}$$

$$h(X) = \log\left(\frac{p(y=1)}{p(y=-1)}\right) + (\mu_1 - \mu_{-1})^T\Sigma_1^{-1}X + \frac{1}{2}\mu_{-1}^T\Sigma_1^{-1}\mu_{-1} - \frac{1}{2}\mu_1^T\Sigma_1^{-1}\mu_1$$

This $h(X)$ is clearly only linearly dependent on $X \implies$ decision boundary is a d-dimensional plane.

(c)

$$\Sigma_1^{-1} = \Sigma_{-1}^{-1} = I$$

$$h(X) = \log\left(\frac{p(y=1)}{p(y=-1)}\right) + \frac{1}{2}\mu_{-1}^T\Sigma_1^{-1}\mu_{-1} - \frac{1}{2}\mu_1^T\Sigma_1^{-1}\mu_1 + (\mu_1 - \mu_{-1})^TX$$

Decision boundary is a d-dimensional plane orthogonal to $(\mu_1 - \mu_{-1})^T$.

4

(a)

$$L(z) = \log\left(1 + exp(-z)\right)$$
$$L'(z) = \frac{-exp(-z)}{1+exp(-z)}$$

$$\begin{aligned} L''(z) &= -exp(-z)\left(\frac{-1}{1+exp(-z))^2}\right)(-exp(-z)) - \frac{exp(-z)}{1+exp(-z)} \\ &= \frac{-exp(-2z)+(1+exp(-z))(exp(-z))}{(1+exp(-z))^2} \\ &= \frac{exp(-z)}{[1+exp(-z)]^2} > 0 \\ &\implies L(z) \text{ is a convex function.} \end{aligned}$$

(b)

f is a concave function.

We use induction to prove that $f(\sum_{i=1}^{m}\alpha_i x_i) \geq \sum_{i=1}^{m}\alpha_i f(x_i)$.

Base case: $f(x) = f(x)$

Inductive Hypothesis: Assume that $f(\sum_{i=1}^{m-1}\alpha_i x_i) \geq \sum_{i=1}^{m-1}\alpha_i f(x_i)$

Proof: $f(\sum_{i=1}^{m}\alpha_i x_i) = f(\sum_{i=1}^{m-1}\alpha_i x_i + \alpha_m x_m)$

Let $\sum_{i=1}^{m-1}\alpha_i x_i = (1-\alpha_m)y$.

From the definition of a concave function,

$f((1-\alpha_m)y + \alpha_m x_m) \geq (1-\alpha_m)f(y) + \alpha_m f(x_m)$

$f((1-\alpha_m)y + \alpha_m x_m) \geq (1-\alpha_m)f(\frac{\sum_{i=1}^{m-1}\alpha_i x_i}{(1-\alpha_m)}) + \alpha_m f(x_m)$

From the inductive hypothesis,

$f((1-\alpha_m)y + \alpha_m x_m) \geq (1-\alpha_m)\frac{\sum_{i=1}^{m-1}\alpha_i}{(1-\alpha_m)}f(x_i) + \alpha_m f(x_m)$

$f((1-\alpha_m)y + \alpha_m x_m) \geq \sum_{i=1}^{m-1}\alpha_i f(x_i) + \alpha_m f(x_m)$

$$\boxed{f(\sum_{i=1}^{m}\alpha_i x_i) \geq \sum_{i=1}^{m}\alpha_i f(x_i)}$$

5

(a)

$$P[Y=1 \mid X=x] = \frac{exp(w_0 + w^T x)}{1+exp(w_0+w^T x)}$$

$$P[Y=0 \mid X=x] = 1 - P[Y=1 \mid X=x] = 1 - \frac{exp(w_0+w^T x)}{1+exp(w_0+w^T x)} = \frac{1}{1+exp(w_0+w^T x)}$$

$$\text{log-odds of success} = \ln\left(\tfrac{P[Y=1|X=x]}{P[Y=0|X=x]}\right)$$

$$\frac{P[Y=1 \mid X=x]}{P[Y=0 \mid X=x]} = exp(w_0 + w^T x)$$

$$\text{log-odds of success} = w_0 + w^T x$$

This is clearly a linear function of x.

(b)

$p(y^i = c|x^i, \theta_1, ..., \theta_c) = \frac{exp(\theta_c^T x^i}{\sum_{c'=1}^C exp(\theta_{c'}^T x^i)}$

$L = \Pi_{i=1}^n p(y^i = c|x^i, \theta_1, ..., \theta_c) = \Pi_{i=1}^n \frac{exp(\theta_c^T x^i}{\sum_{c'=1}^C exp(\theta_{c'}^T x^i)}$

$LL = \sum_{i=1}^n (exp(\theta_c^T x^i) - \sum_{i=1}^n \ln(\sum_{c'=1}^C exp(\theta_{c'}^T x^i))$

Log likelihood: $\boxed{LL = (\sum_{i=1}^n x^i)\theta_c^T - \sum_{i=1}^n \ln(\sum_{c'=1}^C exp(\theta_{c'}^T x^i))}$

$\frac{\partial LL}{\partial \theta_c} = \sum_{i=1}^n x^i I(y^i = c) - \sum_{i=1}^n \frac{1}{\sum_{c'=1} exp(\theta_{c'}^T x^i)} exp(\theta_c^T x^i) x^i I(y^i = c)$

$\frac{\partial LL}{\partial \theta_c} = \sum_{i=1}^n x^i I(y^i = c) - \sum_{i=1}^n x^i p(y^i = c|x^i, \theta_1, ..., \theta_c) I(y^i = c)$

Derivative of LL: $\boxed{\frac{\partial LL}{\partial \theta_c} = \sum_{i=1}^n I(y^i = c) x^i (1 - p(y^i = c|x^i, \theta_1, ..., \theta_c)}$

6

(a)

$$E(U,V) = \sum_{(v,j)\in M} \left(M_{v,j} - \sum_{k=1}^r U_{v,k} V_{j,k} \right)^2$$

$$\frac{\partial E(U,V)}{\partial U_{v,k}} = 2 \sum_j \left(M_{v,j} - \sum_{k=1}^r U_{v,k} V_{j,k} \right) (-V_{j,k})$$

$$\frac{\partial E(U,V)}{\partial V_{j,k}} = 2 \sum_v \left(M_{v,j} - \sum_{k=1}^r U_{v,k} V_{j,k} \right) (-U_{v,k})$$

$$U_{v,k} \longleftarrow U_{v,k} - \mu \frac{\partial E(U,V)}{\partial U_{v,k}}$$

$$\boxed{U_{v,k} \longleftarrow U_{v,k} + 2\mu \sum_j (V_{j,k}) \left(M_{v,j} - \sum_{k=1}^r U_{v,k} V_{j,k} \right)}$$

$$V_{j,k} \longleftarrow V_{j,k} - \mu \frac{\partial E(U,V)}{\partial V_{j,k}}$$

$$\boxed{V_{j,k} \longleftarrow V_{j,k} + 2\mu \sum_v (U_{v,k}) \left(M_{v,j} - \sum_{k=1}^r U_{v,k} V_{j,k} \right)}$$

(b)

$$E(U,V) = \sum_{(v,j)\in M} \left(M_{v,j} - \sum_{k=1}^{r} U_{v,k} V_{j,k} \right)^2 + \lambda \sum_{v,k} U_{v,k}^2 + \lambda \sum_{i,k} V_{i,k}^2$$

$$\frac{\partial E(U,V)}{\partial U_{v,k}} = 2 \sum_{j} \left(M_{v,j} - \sum_{k=1}^{r} U_{v,k} V_{j,k} \right) (-V_{j,k}) + 2\lambda U_{v,k}$$

$$\frac{\partial E(U,V)}{\partial V_{j,k}} = 2 \sum_{v} \left(M_{v,j} - \sum_{k=1}^{r} U_{v,k} V_{j,k} \right) (-U_{v,k}) + 2\lambda V_{j,k}$$

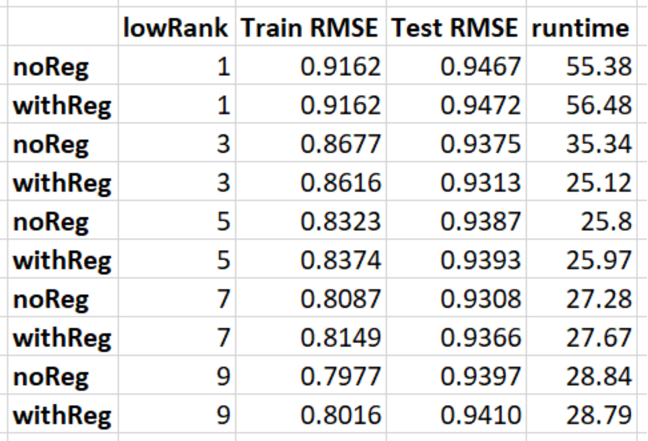
$$U_{v,k} \longleftarrow U_{v,k} - \mu \frac{\partial E(U,V)}{\partial U_{v,k}}$$

$$\boxed{U_{v,k} \longleftarrow U_{v,k} + \mu \left(2 \sum_{j} (V_{j,k}) \left(M_{v,j} - \sum_{k=1}^{r} U_{v,k} V_{j,k} \right) - 2\lambda U_{v,k} \right)}$$

$$V_{j,k} \longleftarrow V_{j,k} - \mu \frac{\partial E(U,V)}{\partial V_{j,k}}$$

$$\boxed{V_{j,k} \longleftarrow V_{j,k} + \mu \left(2 \sum_{v} (U_{v,k}) \left(M_{v,j} - \sum_{k=1}^{r} U_{v,k} V_{j,k} \right) - 2\lambda V_{j,k} \right)}$$

(c)



	lowRank	Train RMSE	Test RMSE	runtime
noReg	1	0.9162	0.9467	55.38
withReg	1	0.9162	0.9472	56.48
noReg	3	0.8677	0.9375	35.34
withReg	3	0.8616	0.9313	25.12
noReg	5	0.8323	0.9387	25.8
withReg	5	0.8374	0.9393	25.97
noReg	7	0.8087	0.9308	27.28
withReg	7	0.8149	0.9366	27.67
noReg	9	0.7977	0.9397	28.84
withReg	9	0.8016	0.9410	28.79

Figure 1: Experiment Output

The hyperparameters $\lambda = 0.02$ and $\mu = 2e - 4$ were picked by grid search. A very low learning rate μ caused runtime to increase too much.

We see that although the test RMSE varies randomly with lowRank, train RMSE reduces. This makes sense as we increase the number of features used, the error reduces.