
O
G
D
O
O
O
D
O

CS229 Summer 2022 Final jachan7.

Start time 9PM AEST.
+4hr 1AM

I attest that I have not given or received aid in this examination, and that I have done my share and taken an active part in seeing to it that others as well as myself uphold the spirit and letter of the Honor Code.

[signature] 13 August 2022
~9PM AEST

Section		Fin.
1	10 pts	10pm
2	10 pts	11pm
3	10 pts	12am
4	10 pts	1am
	40 pts	

Question 1

1. FALSE. EM algorithm could potentially overfit because this algorithm attempts to approximate Maximum likelihood estimation (MLE) in the face of latent variables, and MLE itself can overfit data.

2. TRUE. It is possible for model A to have lower generalization error than model B + regularization if the hypothesis class of Model A more accurately models the test data than model B + regularization.

3. False, SVMs don't output class probabilities

4. True Generative models model $p(x, y)$ whereas discriminative models model $P(y|x)$ directly.

5. False The final cluster assignments in k-means are sensitive to the initial cluster centroids, which are typically randomized. Thus you may get different number of datapoints assigned to each cluster.

6. TRUE

While Newton's method (NM) may take fewer steps than Stochastic Gradient Descent (SGD), SGD will converge faster than NM when the Hessian is intractable or slow to calculate, which is required for NM because it's a second order operation vs. first order operation for SGD.

7. TRUE

Bellman equation using the value iteration algorithms guarantees finding an optimal policy for any arbitrary MDP.

8. TRUE

Given the same state $s \in S$, taking the greedy action as compared to arbitrary policy π_0 while in this same state means that $V^{\pi_1}(s) \geq V^{\pi_0}(s)$.

Note: greedy policies aren't necessarily globally optimal but given this local case of same state s it is locally optimal.

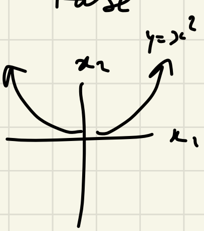
9. FALSE

A unique solution will exist regardless of K & d. Since the optimation function is a quadratic there will always exist a solution for θ.

$$0 = \sum_{i=1}^{n} \cancel{2}\left(y^{(i)} - \theta^T x^{(i)}\right) \cancel{x^{(i)}}$$

$$\theta^T = \sum_{i=1}^{n} y^{(i)} / x^{(i)}$$

10. False



PCA, in this case won't capture all or almost all the variance because we have a non-linear distribution and PCA can't handle non-linear data.

Question 2.

1. Need to show A is PSD.
Which can be done $u^T A u \geqslant 0$ or show eig values $\geqslant 0$.

$$(A - I\lambda) = \begin{bmatrix} 1-\lambda & a & a \\ a & 1-\lambda & a \\ a & a & 1-\lambda \end{bmatrix} \xrightarrow{\text{let } 1-\lambda = b} \begin{bmatrix} b & a & a \\ a & b & a \\ a & a & b \end{bmatrix}$$

$$\xrightarrow[R_2 - R_3]{R_1 - R_3} \begin{bmatrix} b-a & 0 & a-b \\ 0 & b-a & a-b \\ a & a & b \end{bmatrix} = b-a \begin{vmatrix} b-a & a-b \\ a & b \end{vmatrix} + (a-b) \begin{vmatrix} 0 & b-a \\ a & a \end{vmatrix}$$

$$0 \leq \cancel{(b-a)}\left((b-a)b - a(a-b)\right) + \underbrace{\cancel{(a-b)}}_{-1}\left(-a(b-a)\right)$$

$$= (b-a)b - a(a-b) + a(b-a)$$

$$= (b-a)(b+a+a)$$

$$= (b-a)(b+2a)$$ two cases!!

$1-\lambda + a = 0$

$1 - a = \lambda$

$1 - a \geqslant 0$

$a \leq 1$

$b + 2a = 0$

$(1-\lambda) + 2a = 0$

$1 + 2a = \lambda$

$1 + 2a \geqslant 0$

$a \geqslant -\frac{1}{2}$

$\therefore \boxed{-\frac{1}{2} \leq a \leq 1}$

2. $x = As.$

$x = AAAs.$

3. Drawback when n is very large for gaussian processes vs. bayesian linear regression is that is slower to run. because it consumes more computation since it is $O(n^3)$ where n is the number of datapoints

$x \in R^d$

4. $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_d \end{pmatrix} \Rightarrow \phi(x)$ has all monomial of deg 1 & deg 2

formula for num dimensions of all monomials degree D given $x \in R^d$ is

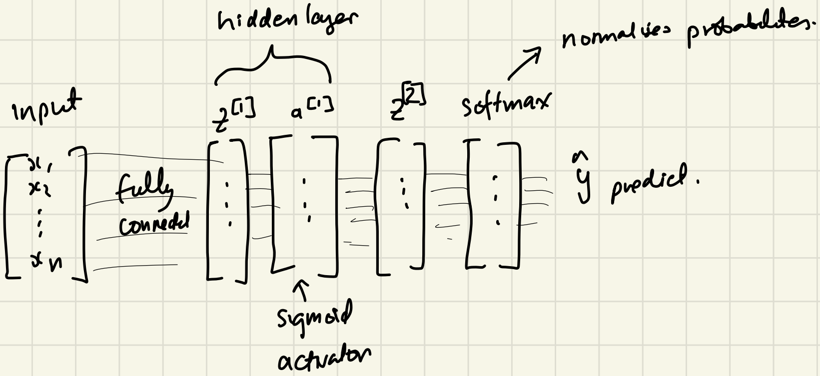
$$\frac{(D+d-1)!}{D!(d-1)!}$$

$$\Rightarrow \quad D=1: \quad \frac{(1+d-1)!}{1!(d-1)!} = \frac{d!}{(d-1)!} = d$$

$$\Rightarrow \quad D=2: \quad \frac{(2+d-1)!}{2!(d-1)!} = \frac{(d+1)!}{2\,(d-1)!} = \frac{(d+1)(d)}{2}$$

$$\boxed{\therefore \text{ dimension of } \phi(x) = d + \frac{d(d+1)}{2}}$$

5. Describe how this neural network can be viewed as learning a feature mapping for softmax regression



<u>Assumptions</u> · fully connected layers

Neural Networks (NN) with hidden layers perform the function of automatic feature detection. In this case, this NN has one hidden layer with a sigmoid activation, which learns the features to then pass to the next layer

The second layer evaluates some linear combination (+ bias) to formulate a classification outcome, which the final softmax layer normalises so that the probabilities for each predicted class sum to 1 for each input sample.

Question 3

1. a. Continuous real values

b. discrete bernoulli

c. discrete multinomial.

2. a $x = x_s + x_{s\perp}$

- S of R^d, $S\perp$ R^d as well

$u^T v = 0$

$\theta = \theta_s + \theta_{s\perp}$

- $\theta \in R^d$

$= \text{proj}_{S_x} \theta + \text{proj}_{S_x^\perp} \theta$

$S_x = \text{span}\{x^{(i)} \mid i \in 1,2,\ldots n\} = \text{span}\{\bar{u}_1\}$

$\Rightarrow \text{proj}_{S_x} \theta = \text{proj}_u \theta = \left(\frac{u\theta}{u \cdot u}\right) u$

b

c

Question 4

1. 1st layer

$W^{[1]}$: $p \times n$

$b^{[1]}$ $\quad p$

2nd layer

$W^{[2]}$ $\quad K \times p$

$b^{[2]}$ $\quad K$

Shortcut layer

W $\quad K \times n$

· Softmax · K classes ∴ $y \in \mathbb{R}^K$.

· m training examples

$$\boxed{\begin{aligned} &\text{total} = \\ &p \times n + p + K \times p + K + K \times n \\ &= p(n+1) + K(p+1+n) \end{aligned}}$$

2. Goal: Show $\nabla_{z^{[2]}} CE(y,\hat{y}) = \hat{y} - y$.

Given:

$$CE(y,\hat{y}) = -\sum_{k=1}^{K} y_k \log \hat{y}_k$$

Use hint: rewrite y_k as index in l^{th} position

$$= -\cancel{y_\ell} \log \hat{y}_\ell$$

Recall $\hat{y}_\ell := \text{softmax}(z^{[2]})_\ell$ ∵ its the final layer.

so
$$CE(y,\hat{y}) = -\log\left(\text{softmax}\left(z_\ell^{[2]}\right)\right)$$

$$= -\log\left(\frac{e^{z_\ell^{[2]}}}{\sum_{k=1}^{k} e^{z_k^{[2]}}}\right)$$

$$= -\log e^{z_\ell^{[2]}} + \log \sum_{k=1}^{k} e^{z_k^{[2]}}$$

$$= -z_\ell^{[2]} + \log \sum_{k=1}^{k} e^{z_k^{[2]}}$$

Like our PS2 assignment, we know there are 2 cases when finding partial derivative of softmax

Case 1

$$\frac{\partial CE(y,\hat{y})}{\partial z_l^{[2]}} = -1 + \underbrace{\frac{z_l^{[2]}}{\sum_{k=1}^{K} e^{z_k^{[2]}}}}_{\hat{y}_l}$$

$$= \hat{y}_l - y_l$$

Case 2 when $j \neq l$

$$\frac{\partial CE(y,\hat{y})}{\partial z_j^{[2]}} = 0 + \underbrace{\frac{e^{z_j^{[2]}}}{\sum_{k=1}^{K} e^{z_k^{[2]}}}}_{\hat{y}_j}$$

$$= \hat{y}_j - y_j$$

finally when both cases combined:

$$\nabla_{z^{[2]}} CE(y,\hat{y}) = \hat{y} - y.$$

3. $$\nabla_{W^{[2]}} CE(y,\hat{y}) = \delta_{z^{[2]}} \circ \nabla_{W^{[2]}} z^{[2]}$$

$$= (\hat{y}-y) \circ \nabla_{W^{[2]}} \left(W^{[2]} a^{[1]} + b^{[2]} + Wx\right)$$

$$= (\hat{y}-y)\, a^{[1]T}$$

$$\nabla_{b^{[1]}} CE(y,\hat{y}) = \delta_{z^{[2]}} \circ \nabla_{a^{[1]}} z^{[2]} \circ \nabla_{b^{[1]}} a^{[1]}$$

$$= W^{[2]T}(\hat{y}-y) \circ \nabla_{b^{[1]}} a^{[1]}$$

$$= W^{[2]T}(\hat{y}-y) \circ a^{[1]} \circ (1-a^{[1]})$$

$$\nabla_{W^{[1]}} CE(y,\hat{y}) = \delta_{z^{[2]}} \circ \nabla_{a^{[1]}} z^{[2]} \circ \nabla_{W^{[1]}} a^{[1]}$$

$$= W^{[2]T}(\hat{y}-y) \circ \nabla_{W^{[1]}} a^{[1]}$$

$$= \left(W^{[2]T}(\hat{y}-y) \circ a^{[1]} \circ (1-a^{[1]})\right) x$$

$$\nabla_{x} CE(y,\hat{y}) = \delta_{z^{[2]}} \circ \nabla_{a^{[1]}} z^{[2]} \circ \nabla_{x} a^{[1]} + \delta_{z^{[2]}} \circ \nabla_{x}(Wx)$$

$$= W^{[2]T}(\hat{y}-y) \circ \nabla_{x} a^{[1]} + W^{T} \delta_{z^{[2]}}$$

$$= W^{[1]T}\left(W^{[2]T}(\hat{y}-y) \circ a^{[1]} \circ (1-a^{[1]})\right)$$

$$+ W^{T}(\hat{y}-y)$$