U88387934

Ziqi Tan

Q1

a) No. A discriminative classifier does not depend on any priors or probability distribution functions. It can only classify samples but not ~~generate~~ samples.
generate

b) Anomaly detection, because the number of failing engines are too small.

c) The prior can change the position of the decision boundary of LDA.

d) 1. ~~Normalize~~ Normalize the image data.
2. Compute the covariance matrix of the data.
3. Perform SVD on this matrix.
4. Project the data onto top K eigenvectors to get the compression image

The K will control the degree of compression.

e) $\min\{\frac{1}{2}\|w\|^2 + \lambda \sum_{i=1}^{n} \xi_i\}$

s.t. $(w^T x_i + b) y_i \geq 1 - \xi_i \quad \forall i$

$\xi_i \geq 0$

U88387934

Ziqi Tan

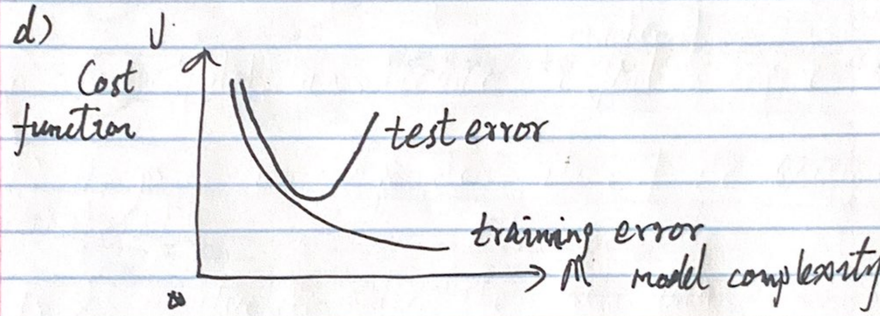
Q2

A) A: $\lambda = 10^2$ B: $\lambda = 10^0$ C: $\lambda = 10^{-2}$

b) Model A has highest bias.

c) Model C has higher variance.

d)



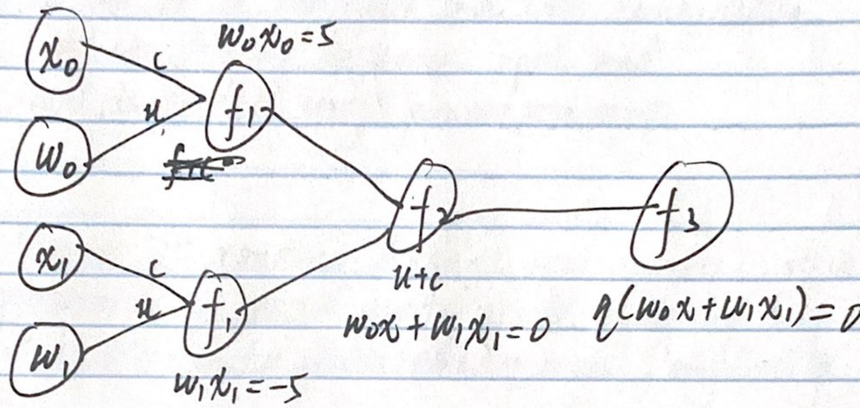
Focus on the test error curve, when the curve goes up, overfitting is happening although training error is still going down.

U 89387934

Ziyi Tan

Q3

a)



b) $\frac{\partial f_1}{\partial u} = \cancel{x_0}\, c \quad \frac{\partial f_2}{\partial u} = 1 \quad \frac{\partial f_3}{\partial u} = 1 - u^2$

c) $wx^T = 1\times5 - 5\times1 = 0$

$q(wx^T) = 0$ is the forward pass output.

d) $\cancel{h(x) = q(w_0x + w_1x_1)}$

$h(x) = q(u) \quad u = w_0x_0 + w_1x_1$

$$\frac{\partial h}{\partial w_0} = \frac{\partial h}{\partial u}\frac{\partial u}{\partial w_0} = [1 - (w_0x_0 + w_1x_1)^2]\,x_0$$

$$\frac{\partial h}{\partial x_0} = \frac{\partial h}{\partial u}\frac{\partial u}{\partial x_0} = [1 - (w_0x_0 + w_1x_1)^2]\,w_0$$

$$\frac{\partial h}{\partial w_1} = \frac{\partial h}{\partial u}\frac{\partial u}{\partial w_1} = [1 - (w_0x_0 + w_1x_1)^2]\,x_1$$

$$\frac{\partial h}{\partial x_1} = \frac{\partial h}{\partial u}\frac{\partial u}{\partial x_1} = [1 - (w_0x_0 + w_1x_1)^2]\,w_1$$

U88387934

Ziqi Tan

Q4

a) It can be a ~~CN~~ RNN and CNN combination.
RNN can handle sequence input and
CNN is good at image ~~pro~~ ~~do~~ processing.

b) Dropout. It randomly disables some neurons at a certain probability to prevent overfitting. ~~When~~ ~~By~~ By doing that, it is similar to ~~decre~~ reducing model complexity.

c) No.
Deeping learning itself takes care of feature engineering. Other machine learning algorithms require ~~use~~ manually feature engineering.

d) No.
The function of activation is to ~~apply~~ make the model to be able to learn non-linear features of the data. If you use a linear activation function, it is hard for a model ~~the~~ to form non-linear features by performing linear transformation.

e) Use sythetic images to augment the ~~image~~ data set, such as cropping, coloring, rotation and ~~so~~ so on, which can tell the model that these are also the features you should learn.

f) CNN has covolutional layler and pooling layer.
CNN can better handle high-dimension input like 2D image.

g)

U88387934

Zyi Tan

Q5

a) $p(D|\mu) = \prod_{i=1}^{m} \mu^{x(i)} (1-\mu)^{1-x(i)}$

b) ~~$p(\mu|D) = \frac{p(D|\mu)\, p(\mu)}{p(D)}$~~

$\arg\max_{\mu} p(\mu|D) = p(D|\mu)\, p(\mu)$

$=$

U88387934

Ziqi Tan

Q6

A) i) That means the prediction is correct.

ii) The prediction ~~is~~ lies on the decision boundary.

iii) The prediction is wrong.

b) ~~The loss function is not convex~~
~~It is hard to find a minimum loss.~~

~~Cannot produce a reasonable solution.~~
~~It increases monotonically.~~

It can be a reasonable loss function,
but not a good loss function,
because we expect z to be greater to classify ~~it~~ correctly.

c) i)

$$\cancel{J = -\exp(-2z-1) + \frac{1}{2}\lambda\|w\|^2}$$

$$J = \exp(-2z-1) + \frac{1}{2}\lambda\|w\|^2$$

ii) It should be minimized.

GD:

Initialize w.

Repeat:

$$\theta_{t+1} = \theta_t - \alpha\left[\frac{dJ}{dw} + \lambda\|w\|\right]$$

Until it converges.