Pre-Final Practice Questions

Patrick Spieker and Ben Evans

March 2019

Questions inspired by Tom Mitchell of CMU, Carlos Guestrin of CMU/UW, Emily Fox of UW, Noah Smith of UW

Pre-midterm:

1. Overfitting and Generalization
2. K-means
3. Perceptron
4. Bayes optimal classification
5. PCA
6. Linear regression
7. L2 regularization
8. Loss function basics
9. Gradient Descent

Post-midterm:

1. SGD + Mini-batching + Multi-class classification
2. Back-propagation
3. Non-convex optimization: stationary/saddle points, etc.
4. Structured Neural Networks: CNNS
5. Auto-differentiation (Baur-Strassen, etc)
6. Run-time analysis of auto-diff + training NNs
7. Gaussian Mixture Models
8. Expectation Maximization
9. Generative Models

Neural Network Overfitting

For a neural network, which one of these structural assumptions is the one that most affects the trade-off between underfitting (i.e. a high bias model) and overfitting (i.e. a high variance model):
(i) The number of hidden nodes
(ii) The learning rate
(iii) The initial choice of weights
(iv) The use of a constant-term unit input

Solution.

The number of hidden nodes. 0 will result in a linear model, which many (with non-linear activation) significantly increases the variance of the model.

L* regression decision boundaries

Consider the dataset $X = [-2, -1, 0, 1]^\intercal, Y = [1, 1, -1, -1]$.
a. Plot the dataset
b. Draw the line that would result from running linear regression on the dataset
c. Draw the line that would result from running logistic regression on the dataset

Solution.

(Jupyter notebook)

Linear Regression vs Logistic Regression

True or False: Since classification is a special case of regression, logistic regression is a special case of linear regression [Tom Mitchell's Question]

Solution.

False. Logistic Regression uses an entirely different loss function, and has significantly different behavaior (hopefully highlighted in other questions).

Neural Network Learning

Which of the following logical structure can a 1 hidden layer neural network learn (assuming $\{0,1\}^2$ is our input): OR, AND, NOT, XOR?

Solution.

All of them! A 0-layer net with log-loss (logistic regression) can learn OR, AND, and NOT. 1 hidden layer allows for the learning of XOR.

Neural Network Optimization

Let f be a neural network with one hidden layer defined as $f(x) = W_2 \tanh(W_1 x)$.
Let our loss function be the squared loss: $\ell = \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2$.
Say we use gradient descent to update the weights and get to a point where $\frac{\partial \ell}{\partial W_1} = \frac{\partial \ell}{\partial W_2} = 0$.

Have we reached the **global** minimum of our loss function? Why or why not?

Solution.

No! When optimizing neural networks, our loss is non-convex, so we have no guarantee that we have reached the global minimum. In reality, we could even be at a saddle point.

Is this loss?

Why don't we try and minimize the 0/1 loss directly? What do we do instead?

Solution.

Minimizing the 0/1 loss is NP-hard. We use a relaxation (like square loss) and minimize that instead.

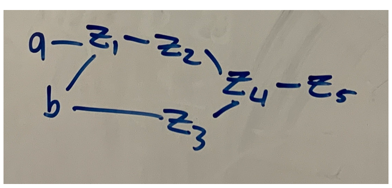
(Manual) Diff

Let $f(a, b) = \sin(e^{a+b} + b^2)$ and let

$$
\begin{aligned}
z_1 &= a + b \\
z_2 &= e^{z_1} \\
z_3 &= b^2 \\
z_4 &= z_2 + z_3 \\
z_5 &= \sin(z_4)
\end{aligned}
$$

a. Draw the computation graph of f.
b. Compute $\frac{df}{da}, \frac{df}{db}$ using the reverse mode.

Solution.



Solution.

$$\frac{dz_5}{dz_5} = \boxed{1}$$

$$\frac{dz_5}{dz_4} = \frac{dz_5}{dz_5} \cdot \frac{\partial z_5}{\partial z_4} = \boxed{\frac{dz_5}{dz_5} \cos(z_4)}$$

$$\frac{dz_5}{dz_3} = \frac{dz_5}{dz_4} \cdot \frac{\partial z_4}{\partial z_3} = \boxed{\frac{dz_5}{dz_4} \cdot 1}$$

$$\frac{dz_5}{dz_2} = \frac{dz_5}{dz_4} \cdot \frac{\partial z_4}{\partial z_2} = \boxed{\frac{dz_5}{dz_4} \cdot 1}$$

$$\frac{dz_5}{dz_1} = \frac{dz_5}{dz_2} \cdot \frac{\partial z_2}{\partial z_1} = \boxed{\frac{dz_5}{dz_2} \cdot e^{z_1}}$$

$$\frac{dz_5}{da} = \frac{dz_5}{dz_1} \cdot \frac{\partial z_1}{\partial a} = \boxed{\frac{dz_5}{dz_1} \cdot 1}$$

$$\frac{dz_5}{db} = \frac{dz_5}{dz_1} \cdot \frac{\partial z_1}{\partial b} + \frac{dz_5}{dz_3} \cdot \frac{\partial z_3}{\partial b}$$

$$= \boxed{\frac{dz_5}{dz_1} \cdot 1 + \frac{dz_5}{dz_3} \cdot 2b}$$

PCA

Consider the dataset $X = [[1, 1], [-1, -1], [0.5, -0.5], [-0.5, 0.5]]$.

a. Plot the dataset.

b. What is the first principal component?

Solution.

This problem can be done visually. By inspecting our plot, we can see the direction of highest variance is along the $x_1 = x_2$ direction. It follows that $v_1 = [\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}]^\mathsf{T}$