Exam 2 Solutions

April 28, 2015

Question 1: Short Answer

1. d
2. b
3. 
\[ \hat{p}_{\text{MLE}} = \frac{\text{#heads}}{\text{#number of tosses}} = \frac{3}{5}. \]
4. b and d
5. b
6. a
8. c
9. SVM
10. b
11. a
Question 2: Representation Learning, Dimensionality Reduction, Neural Networks

2.1 Principal Components Analysis

(a) In each figure, the first principle component is labeled $p_1$ and the second principle component is labeled $p_2$.

(b) It is not possible to draw a third principal component, since the principal components must be orthogonal to one another, but the two principal components already form a basis for the entire space.

2.2 Autoencoder Neural Networks

(a) Set $w_1, w_4, w_5,$ and $w_8$ to 1 and the rest to 0.

(b) (i) Yes. Since the nodes are linear units, the entire network outputs a linear function of the input. Therefore, training the network is equivalent to linear regression, which can be solved by gradient descent. Moreover, the weight assignment in part (a) shows that it is possible to achieve 0 error. Therefore, gradient descent will converge to weights with 0 error.

(ii) No. Since the resulting weights compromise between minimizing the reconstruction error and minimizing the regularization term, the solution will not generally have zero error.

(c) Yes. Set $(w_1, w_3) = p_1$ and $(w_2, w_4) = p_2$. Then the value of the internal nodes is exactly the inner product of the input with each of the principal components.
Question 3: Kernels, Margins, and SVM

3.1 True/False
(a) True
(b) False
(c) True

3.2 Composition of kernels
(a) Yes. Let $\Phi(x) = \sqrt{10}x$. Then we have that $\Phi(x)^\top \Phi(z) = \sqrt{10}x^\top \sqrt{10}z = 10x^\top z = K(x, z)$.

(b) Yes. For any kernels $k_1$ and $k_2$ and any positive numbers $\alpha \geq 0$ and $\beta \geq 0$, we have that $\alpha k_1 + \beta k_2$ and $k_1 \cdot k_2$ are both valid kernels. Repeated application of the multiplication closure property shows that $(x^\top z)^5$ is a valid kernel. Since both $x^\top z$ and 1 are valid kernels (the latter having the feature map $\Phi(x) = 1$), we have that $x^\top z + 1$ is a valid kernel, and therefore so is $(x^\top z + 1)^5$. Finally, the sum of the two kernels is again a kernel.

3.3 Margins, SVM
(a) The margin between the hyperplane $w^\top x = 0$ and the point $x_0 = [1, 1, 0, -1]$ is given by $\frac{w^\top x_0}{\|w\|} = 1/2$.

(b) The max margin linear separator is $h(x) = \text{sign}(w^\top x)$, where $w = [1, 1]$. All training points are equally distant from the decision boundary $w^\top x = 0$. The margin for the point $x_0 = [1, 0]$ is given by $\frac{w^\top x_0}{\|w\|} = 1/\sqrt{2}$.

(c) False.
Question 4: Expectation Maximization

4.1 Short Answer
(a) True
(b) False
(c) iii. Only the M step.

4.2 Deriving EM
(a)\[
\log P(T, R, A|\pi, \tau, \theta) = \log \left( \prod_{i=1}^{n} P(T_i, R_i, A_i|\pi, \tau, \theta) \right)
= \sum_{i=1}^{n} \log [P(T_i, R_i, A_i|\pi, \tau, \theta)]
= \sum_{i=1}^{n} \log \left[ P(T_i|\pi)P(A_i|\pi)P(R_i|A_i, T_i, \theta) \right]
\]

(b) For each \( a \in \{0, 1\} \), we have
\[
P(A_i = a|T_i, R_i, \pi, \tau, \theta)
= \frac{P(A_i = a, T_i, R_i|\pi, \tau, \theta)}{P(T_i, R_i|\pi, \tau, \theta)}
= \frac{P(A_i = a|\tau)P(T_i|\pi)P(R_i|A_i = a, T_i, \theta)}{P(A_i = 0|\tau)P(T_i|\pi)P(R_i|A_i = 0, T_i, \theta) + P(A_i = 1|\tau)P(T_i|\pi)P(R_i|A_i = 1, T_i, \theta)}
= \frac{P(A_i = a|\tau)P(R_i|A_i = a, T_i, \theta)}{P(A_i = 0|\tau)P(R_i|A_i = 0, T_i, \theta) + P(A_i = 1|\tau)P(R_i|A_i = 1, T_i, \theta)}.
\]

Using the fact that \( P(A_i = a|\tau) = \tau^a(1 - \tau)^{1-a} \) and \( P(R_i|A_i = a, T_i, \theta) = \theta_{T,i,a} R_i (1 - \theta_{T,i,a})^{1-R_i} \) gives
\[
P(A_i = a|T_i, R_i, \pi, \tau, \theta) = \frac{\tau^a(1 - \tau)^{1-a} \theta_{T,i,a} R_i (1 - \theta_{T,i,a})^{1-R_i}}{(1 - \tau)\theta_{T,0} R_i (1 - \theta_{T,0})^{1-R_i} + \tau\theta_{T,1} R_i (1 - \theta_{T,1})^{1-R_i}}.
\]

(c) For the first part:
\[
\propto C \prod_{i=1}^{n} \theta_{00} R_i (1-T_i)(1-A_i) \cdot (1 - \theta_{00})^{1-R_i}(1-T_i)(1-A_i)
\]
Question 5: Training Error, Testing Error, Underfitting, Overfitting, and Model Complexity

5.1 Drawing Error Curves

(a) As we increase the model complexity, we will be able to better fit the training data, so the training error should go to zero.

(b) Low complexity models require less data to train, but might not be able to fit the data. High complexity models can fit the data, but require more data to train (so for a fixed training set, we expect to overfit).

(c) We want to pick the complexity that minimizes our error on the testing data.

(d) The high testing error to the left of the optimal complexity is due to underfitting, and the high test error to the right is due to overfitting.

(e) Given more training data, I would expect the curve to get higher, but still tend to zero, since it is harder to fit more data with a model of the same complexity.

(f) I would expect the curve showing the testing error to move closer to the training error curve, since now the training error is a better estimate of the test error.

(g) The vertical line should move to the right because we now have enough data to accurately train a more complex model, which may better fit the data.
5.2 Error Curves for Boosting

(a) We saw in class that the training error of the boosted classifier goes to zero exponentially quickly. Since the bottom curve goes to zero quickly, it is likely the training error.

(b) In some situations, running additional rounds of boosting even after the training error has dropped to zero has the effect of increasing the margin of the classifier. We saw in class that large-margin classifiers have lower complexity, which explains why the training error continues to drop even after the training error is zero.