CS 189 Spring 2017 Introduction to Machine Learning Final

- Please do not open the exam before you are instructed to do so.
- The exam is closed book, closed notes except your two page cheat sheet.
- **Electronic devices are forbidden on your person**, including cell phones, iPods, headphones, and laptops. Turn your cell phone off and **leave all electronics at the front of the room**, or **risk getting a zero** on the exam.
- You have 3 hours.
- Please write your initials at the top right of each odd-numbered page (e.g., write "JS" if you are Jonathan Shewchuk). Finish this by the end of your 3 hours.
- Mark your answers on the exam itself in the space provided. Do **not** attach any extra sheets.
- The total number of points is 150. There are 26 multiple choice questions worth 3 points each, and 7 written questions worth a total of 72 points.
- For multiple answer questions, fill in the bubbles for **ALL correct choices**: there may be more than one correct choice, but there is always at least one correct choice. **NO partial credit** on multiple answer questions: the set of all correct answers must be checked.

First name	
Last name	
SID	
First and last name of student to your left	
First and last name of student to your right	

Q1. [78 pts] Multiple Answer

Fill in the bubbles for **ALL correct choices**: there may be more than one correct choice, but there is always at least one correct choice. **NO partial credit**: the set of all correct answers must be checked.

(1) [3 pts] Which of the following are NP-hard problems? Let $X \in \mathbb{R}^{n \times d}$ be a design matrix, let $y \in \mathbb{R}^n$ be a vector of labels, let L be the Laplacian matrix of some n-vertex graph, and let $\mathbf{1} = [1 \quad 1 \quad \ldots \quad 1]^\top$.

- ◯ $\min_{\mu,y} \sum_{i=1}^{k} \sum_{y_j=i} |X_j - \mu_i|^2$ where each μ_i is the mean of sample points assigned class i
- ◯ $\min_{y \in \mathbb{R}^n} \frac{1}{4} y^\top L y$ subject to $|y|^2 = n$; $\mathbf{1}^\top y = 0$
- ◯ $\min_{y} \sum_{i=1}^{k} \sum_{y_j=i} |X_j - \mu_i|^2$ with each μ_i fixed
- ◯ $\min_{y \in \mathbb{R}^n} \frac{1}{4} y^\top L y$ subject to $\forall j, y_j \in \{-1, +1\}$; $\mathbf{1}^\top y = 0$

(2) [3 pts] Which clustering algorithms permit you to decide the number of clusters after the clustering is done?

- ◯ k-means clustering
- ◯ a k-d tree used for divisive clustering
- ◯ agglomerative clustering with single linkage
- ◯ spectral graph clustering with 3 eigenvectors

(3) [3 pts] For which of the following does normalizing your input features influence the predictions?

- ◯ decision tree (with usual splitting method)
- ◯ neural network
- ◯ Lasso
- ◯ soft-margin support vector machine

(4) [3 pts] With the SVD, we write $X = UDV^\top$. For which of the following matrices are the eigenvectors the columns of U?

- ◯ $X^\top X$
- ◯ $X^\top X X^\top X$
- ◯ $XX^\top$
- ◯ $XX^\top XX^\top$

(5) [3 pts] Why is PCA sometimes used as a preprocessing step before regression?

- ◯ To reduce overfitting by removing poorly predictive dimensions.
- ◯ To make computation faster by reducing the dimensionality of the data.
- ◯ To expose information missing from the input data.
- ◯ For inference and scientific discovery, we prefer features that are not axis-aligned.

(6) [3 pts] Consider the matrix $X = \sum_{i=1}^{r} \alpha_i u_i v_i^\top$ where each α_i is a scalar and each u_i and v_i is a vector. It is possible that the rank of X might be

- ◯ $r + 1$
- ◯ $r - 1$
- ◯ r
- ◯ 0

(7) [3 pts] Why would we use a random forest instead of a decision tree?

- ◯ For lower training error.
- ◯ To better approximate posterior probabilities.
- ◯ To reduce the variance of the model.
- ◯ For a model that is easier for a human to interpret.

(8) [3 pts] What tends to be true about increasing the k in k-nearest neighbors?

- ◯ The decision boundary tends to get smoother.
- ◯ The bias tends to increase.
- ◯ The variance tends to increase.
- ◯ As the number of sample points approaches infinity (with $n/k \to \infty$), the error rate approaches less than twice the Bayes risk (assuming training and test points are drawn independently from the same distribution).

(9) [3 pts] Which of the following statements are true about the entropy of a discrete probability distribution?

- ◯ It is a useful criterion for picking splits in decision trees.
- ◯ It is a convex function of the class probabilities.
- ◯ It is maximized when the probability distribution is uniform.
- ◯ It is minimized when the probability distribution is uniform.

(10) [3 pts] A low-rank approximation of a matrix can be useful for

- ◯ removing noise.
- ◯ discovering latent categories in the data.
- ◯ filling in unknown values.
- ◯ matrix compression.

(11) [3 pts] Let L be the Laplacian matrix of a graph with n vertices. Let

$$\beta = \min_{\substack{y \in \mathbb{R}^n \\ \forall i, y_i \in \{-1,+1\} \\ \mathbf{1}^\top y = 0}} y^\top L y \qquad \text{and} \qquad \gamma = \min_{\substack{y \in \mathbb{R}^n \\ |y|^2 = n \\ \mathbf{1}^\top y = 0}} y^\top L y.$$

Which of the following are true for every Laplacian matrix L?

- ◯ $\beta \geq \gamma$
- ◯ $\beta \leq \gamma$
- ◯ $\beta > \gamma$
- ◯ $\beta < \gamma$

(12) [3 pts] Which of the following are true about decision trees?

- ◯ They can be used only for classification.
- ◯ The tree depth never exceeds $O(\log n)$ for n sample points.
- ◯ All the leaves must be pure.
- ◯ Pruning usually achieves better test accuracy than stopping early.

(13) [3 pts] Which of the following is an effective way of reducing overfitting in neural networks?

- ◯ Augmenting the training data with similar synthetic examples
- ◯ Weight decay (i.e., ℓ_2 regularization)
- ◯ Increasing the number of layers
- ◯ Dropout

(14) [3 pts] If the VC dimension of a hypothesis class $\mathcal{H}$ is an integer $D < \infty$ (i.e., $\text{VC}(\mathcal{H}) = D$), this means

- ◯ there exists some set of D points shattered by $\mathcal{H}$.
- ◯ all sets of D points are shattered by H.
- ◯ no set of $D + 1$ points is shattered by $\mathcal{H}$.
- ◯ $\Pi_{\mathcal{H}}(D) = 2^D$.

(15) [3 pts] Consider the minimizer w^* of the ℓ_2-regularized least squares objective $J(w) = |Xw - y|^2 + \lambda|w|^2$ with $\lambda > 0$. Which of the following are true?

- ◯ $Xw^* = y$
- ◯ w^* exists if and only if $X^\top X$ is nonsingular
- ◯ $w^* = X^+ y$, where X^+ is the pseudoinverse of X
- ◯ The minimizer w^* is unique

(16) [3 pts] You are training a neural network, but the training error is high. Which of the following, if done in isolation, has a better-than-tiny chance of reducing the training error?

- ◯ Adding another hidden layer
- ◯ Adding more units to hidden layers
- ◯ Normalizing the input data
- ◯ Training on more data

(17) [3 pts] Filters in the **late** layers of a convolutional neural network designed to classify objects in photographs likely represent

- ◯ edge detectors.
- ◯ concepts such as "this image contains wheels."
- ◯ concepts such as "there is an animal."
- ◯ concepts such as "Jen is flirting with Dan."

(18) [3 pts] Which of the following techniques usually speeds up the training of a sigmoid-based neural network on a classification task?

- ◯ Using batch descent instead of stochastic
- ◯ Having a good initialization of the weights
- ◯ Increasing the learning rate with every iteration
- ◯ Using the cross-entropy loss instead of the mean squared error

(19) [3 pts] In a soft-margin support vector machine, decreasing the slack penalty term C causes

- ◯ more overfitting.
- ◯ a smaller margin.
- ◯ less overfitting.
- ◯ less sensitivity to outliers.

(20) [3 pts] The shortest distance from a point z to a hyperplane $w^\top x = 0$ is

- ◯ $w^\top z$
- ◯ $\frac{w^\top z}{|w|^2}$
- ◯ $\frac{w^\top z}{|w|}$
- ◯ $|w| \cdot |z|$

(21) [3 pts] The Bayes decision rule

- ◯ does the best a classifier can do, in expectation
- ◯ chooses the class with the greatest posterior probability, if we use the 0-1 risk function
- ◯ can be computed exactly from a large sample
- ◯ minimizes the risk functional

(22) [3 pts] Which of the following are techniques commonly used in training neural nets?

- ◯ linear programming
- ◯ Newton's method
- ◯ backpropagation
- ◯ cross-validation

(23) [3 pts] Which of these statements about learning theory are correct?

◯ The VC dimension of halfplanes is 3.

◯ The VC dimension of halfspaces in 3D is ∞.

◯ For a fixed set of training points, the more dichotomies Π we have, the higher the probability that the training error is close to the true risk.

◯ For a fixed hypothesis class $\mathcal{H}$, the more training points we have, the higher the probability that the training error is close to the true risk.

(24) [3 pts] Which of the following statements are true for a design matrix $X \in \mathbb{R}^{n \times d}$ with $d > n$? (The rows are n sample points and the columns represent d features.)

◯ Least-squares linear regression computes the weights $w = (X^\top X)^{-1} X^\top y$.

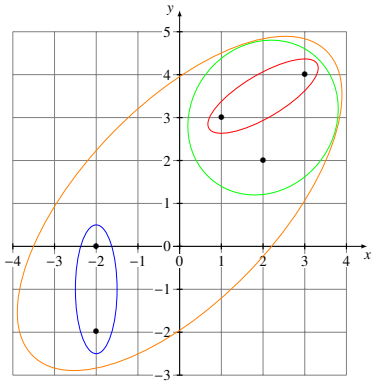
◯ The sample points are linearly separable.

◯ X has exactly $d - n$ eigenvectors with eigenvalue zero.

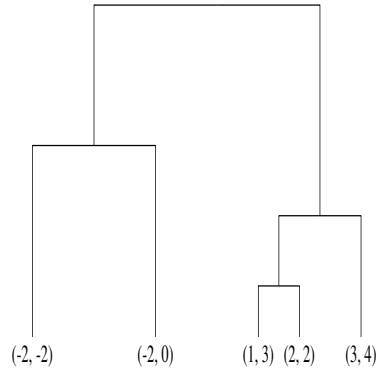
◯ At least one principal component direction is orthogonal to a hyperplane that contains all the sample points.

(25) [3 pts] Which of the following visuals accurately represent the clustering produced by greedy agglomerative hierarchical clustering with centroid linkage on the set of feature vectors {(-2, -2), (-2, 0), (1, 3), (2, 2), (3, 4)}?

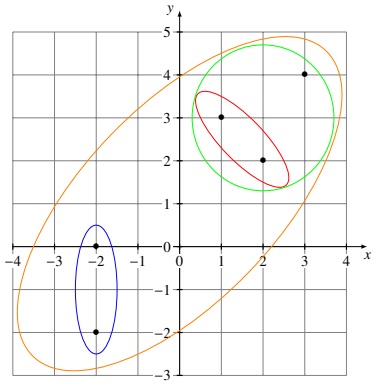
◯



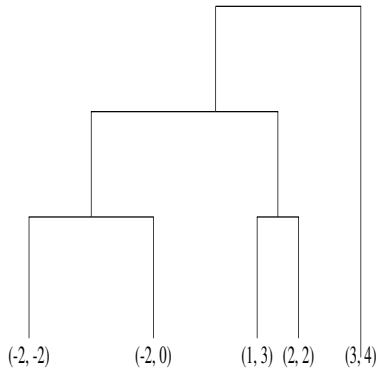
◯



◯



◯



(26) [3 pts] Which of the following statements is true about the standard k-means clustering algorithm?

◯ The random partition initialization method usually outperforms the Forgy method.

◯ It is computationally infeasible to find the optimal clustering of $n = 15$ points in $k = 3$ clusters.

◯ After a sufficiently large number of iterations, the clusters will stop changing.

◯ You can use the metric $d(x, y) = \frac{x \cdot y}{|x| \cdot |y|}$.

Q2. [9 pts] A Miscellany

(a) [3 pts] Consider a convolutional neural network for reading the handwritten MNIST letters, which are 28×28 images. Suppose the first hidden layer is a convolutional layer with 20 different 5×5 filters, applied to the input image with a stride of 1 (i.e., every filter is applied to *every* 5×5 patch of the image, with patches allowed to overlap). Each filter has a bias weight. How many weights (parameters) does this layer use?

(b) [3 pts] Let X be an $n \times d$ design matrix representing n sample points in $\mathbb{R}^d$. Let $X = UDV^\top$ be the singular value decomposition of X. We stated in lecture that row i of the matrix UD gives the coordinates of sample point X_i in principal coordinates space, i.e., $X_i \cdot v_j$ for each j, where X_i is the ith row of X and v_j is the jth column of V. Show that this is true.

(c) [3 pts] Let $x, y \in \mathbb{R}^d$ be two points (e.g., sample or test points). Consider the function $k(x, y) = x^\top \text{rev}(y)$ where $\text{rev}(y)$ reverses the order of the components in y. For example, $\text{rev}\left(\begin{bmatrix}1\\2\\3\end{bmatrix}\right) = \begin{bmatrix}3\\2\\1\end{bmatrix}$. Show that k *cannot* be a valid kernel function.

Hint: remember how the kernel function is defined, and show a simple two-dimensional counterexample.

Q3. [10 pts] Maximum Likelihood Estimation for Reliability Testing

Suppose we are reliability testing n units taken randomly from a population of identical appliances. We want to estimate the mean failure time of the population. We assume the failure times come from an exponential distribution with parameter $\lambda > 0$, whose probability density function is $f(x) = \lambda e^{-\lambda x}$ (on the domain $x \geq 0$) and whose cumulative distribution function is $F(x) = \int_0^x f(x)\,\mathrm{d}x = 1 - e^{-\lambda x}$.

(a) [6 pts] In an ideal (but impractical) scenario, we run the units until they all fail. The failure times are $t_1, t_2, \ldots, t_n$.

Formulate the likelihood function $\mathcal{L}(\lambda; t_1, \ldots, t_n)$ for our data. Then find the maximum likelihood estimate $\hat{\lambda}$ for the distribution's parameter.

(b) [4 pts] In a more realistic scenario, we run the units for a fixed time T. We observe r unit failures, where $0 \leq r \leq n$, and there are $n - r$ units that survive the entire time T without failing. The failure times are $t_1, t_2, \ldots, t_r$.

Formulate the likelihood function $\mathcal{L}(\lambda; n, r, t_1, \ldots, t_r)$ for our data. Then find the maximum likelihood estimate $\hat{\lambda}$ for the distribution's parameter.

Hint 1: What is the probability that a unit will not fail during time T? *Hint 2:* It is okay to define $\mathcal{L}(\lambda)$ in a way that includes contributions (densities and probability masses) that are not commensurate with each other. Then the constant of proportionality of $\mathcal{L}(\lambda)$ is meaningless, but that constant is irrelevant for finding the best-fit parameter $\hat{\lambda}$. *Hint 3:* If you're confused, for part marks write down the likelihood that r units fail and $n-r$ units survive; then try the full problem. *Hint 4:* If you do it right, $\hat{\lambda}$ will be the number of observed failures divided by the sum of unit test times.

Q4. [10 pts] Decision Trees

Consider the design matrix

$$\begin{bmatrix} 4 & 6 & 9 & 1 & 7 & 5 \\ 1 & 6 & 5 & 2 & 3 & 4 \end{bmatrix}^\top$$

representing 6 sample points, each with two features f_1 and f_2.

The labels for the data are

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}^\top$$

In this question, we build a decision tree of depth 2 by hand to classify the data.

(a) [2 pts] What is the entropy at the root of the tree?

(b) [3 pts] What is the rule for the first split? Write your answer in a form like $f_1 \geq 4$ or $f_2 \geq 3$. Hint: you should be able to eyeball the best split without calculating the entropies.

(c) [3 pts] For each of the two treenodes after the first split, what is the rule for the second split?

(d) [2 pts] Let's return to the root of the tree, and suppose we're incompetent tree builders. Is there a (not trivial) split at the root that would have given us an information gain of zero? Explain your answer.

Q5. [11 pts] Bagging and Random Forests

We are building a random forest for a 2-class classification problem with t decision trees and bagging. The input is a $n \times d$ design matrix X representing n sample points in $\mathbb{R}^d$ (quantitative real-valued features only). For the ith decision tree we create an n-point training set $X^{(i)}$ through standard bagging. At each node of each tree, we randomly select k of the features (this random subset is selected independently for each treenode) and choose the single-feature split that maximizes the information gain, compared to all possible single-feature splits on those k features. Assume that we can radix sort real numbers in linear time, and we can randomly select an item from a set in constant time.

(a) [3 pts] Remind us how bagging works. How do we generate the data sets $X^{(i)}$? What do we do with duplicate points?

(b) [3 pts] Fill in the blanks to derive the overall running time to construct a random forest with bagging and random subset selection. Let h be the height/depth (they're the same thing) of the tallest/deepest tree in the forest. You must use the tightest bounds possible with respect to n, d, t, k, h, and n'.

Consider choosing the split at a treenode whose box contains n' sample points. We can choose the best split for these n' sample points in $O($_____$)$ time. Therefore, the running time **per sample point in that node** is $O($_____$)$.

Each sample point in $X^{(i)}$ participates in at most $O($_____$)$ treenodes, so each sample point contributes at most $O($_____$)$ to the time. Therefore, the total running time for one tree is $O($_____$)$.

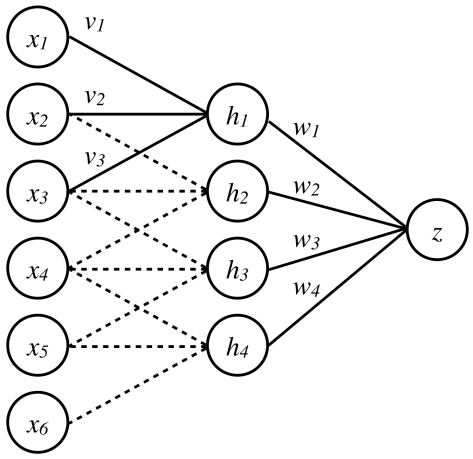
We have t trees, so the total running time to create the random forest is $O($_____$)$.

(c) [2 pts] If we instead use a support vector machine to choose the split in each treenode, how does that change the asymptotic **query** time to classify a test point?

(d) [3 pts] Why does bagging by itself (without random subset selection) tend **not** to improve the performance of decision trees as much as we might expect?

Q6. [11 pts] One-Dimensional ConvNet Backprop

Consider this convolutional neural network architecture.



In the first layer, we have a one-dimensional convolution with a single filter of size 3 such that $h_i = s\left(\sum_{j=1}^{3} v_j x_{i+j-1}\right)$. The second layer is fully connected, such that $z = \sum_{i=1}^{4} w_i h_i$. The hidden units' activation function $s(x)$ is the logistic (sigmoid) function with derivative $s'(x) = s(x)\,(1 - s(x))$. The output unit is linear (no activation function). We perform gradient descent on the loss function $R = (y - z)^2$, where y is the training label for x.

(a) [1 pt] What is the total number of parameters in this neural network? Recall that convolutional layers share weights. There are no bias terms.

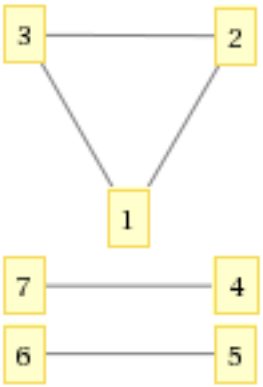
(b) [4 pts] Compute $\partial R / \partial w_i$.

(c) [1 pt] Vectorize the previous expression—that is, write $\partial R / \partial w$.

(d) [5 pts] Compute $\partial R / \partial v_j$.

Q7. [11 pts] Spectral Graph Partitioning

(a) [3 pts] Write down the Laplacian matrix L_G of the following graph G. Every edge has weight 1.



(b) [2 pts] Find three orthogonal eigenvectors of L_G, all having eigenvalue 0.

(c) [2 pts] Use two of those three eigenvectors (it doesn't matter which two) to assign each vertex of G a *spectral vector* in $\mathbb{R}^2$. Draw these vectors in the plane, and explain how they partition G into three clusters. (Optional alternative: if you can draw 3D figures well, you are welcome to use all three eigenvectors and assign each vertex a spectral vector in $\mathbb{R}^3$.)

(d) [3 pts] Let K_n be the complete graph on n vertices (every pair of vertices is connected by an edge of weight 1) and let L_{K_n} be its Laplacian matrix. The eigenvectors of L_{K_n} are $v_1 = \mathbf{1} = \begin{bmatrix} 1 & \dots & 1 \end{bmatrix}^\top$ and *every* vector that is orthogonal to $\mathbf{1}$. What are the eigenvalues of L_{K_n}?

(e) [1 pt] What property of these eigenvalues gives us a hint that the complete graph does not have any good partitions?

Q8. [10 pts] We Hope You Learned This

Consider learning closed intervals on the real line. Our hypothesis class $\mathcal{H}$ consists of all intervals of the form $[a, b]$ where $a < b$ and $a, b \in \mathbb{R}$. We interpret an interval (hypothesis) $[a, b] \in \mathcal{H}$ as a classifier that identifies a point x as being in class C if $a \leq x \leq b$, and identifies x as not being in class C if $x < a$ or $x > b$.

(a) [2 pts] Consider a set containing two distinct points on the real line. Which such sets can be shattered by $\mathcal{H}$?

(b) [2 pts] Show that no three points can be shattered by $\mathcal{H}$.

(c) [2 pts] Write down the shatter function $\Pi_{\mathcal{H}}(n)$. Explain your answer.

(d) [2 pts] Consider another hypothesis class $\mathcal{H}_2$. Each hypothesis in $\mathcal{H}_2$ is a union of two intervals. $\mathcal{H}_2$ is the set of all such hypotheses (i.e., every union of two intervals on the number line). For example, $[3, 7] \cup [8.5, 10] \in \mathcal{H}_2$; that's the set of all points x such that $3 \leq x \leq 7$ or $8.5 \leq x \leq 10$.

What is the largest number of distinct points that $\mathcal{H}_2$ can shatter? Explain why no larger number can be shattered.

(e) [2 pts] Which hypothesis class has a greater sample complexity, $\mathcal{H}$ or $\mathcal{H}_2$? Explain why.