582631 Introduction to Machine Learning (Fall 2016)
Grading criteria for the course examination on December 20th 2016
Teemu Roos

NB: Since Problem 3, which was worth 20 points (out of maximum 60), turned out to be harder than expected, the grading was adjusted by rescaling all problems to be worth 15 points.

1. [*10 points*] Explain briefly the following terms and concepts. Your explanation should include, when appropriate, both a precise definition and a brief description of how the concept is useful in machine learning. Your answer to each subproblem should fit to roughly one third of a page of normal handwriting or less.

   (a) *Gini index* and *entropy*
   The Gini index and entropy can be used as impurity measures. The formulas are $\texttt{Gini} = \sum_{c=1}^{K} \hat{p}_{mc}(1-\hat{p}_{mc})$ and $\texttt{entropy} = -\sum_{c=1}^{K} \hat{p}_{mc} \log \hat{p}_{mc}$. They measure how "pure" (or rather, how "impure", i.e., mixed) a segment of the data is with respect to the class variable. One point was awarded for the definitions (including explanations of the symbols), another point for the explanation of their interpretation.

   (b) *squared error* and *logarithmic loss* (*log-loss*)
   The formulas are $\texttt{squared} = (y - \hat{y})^2$ and $\texttt{log-loss} = -\log \hat{p}(y|x)$, where y is the actual label, $\hat{y}$ is the output of a classifier (a single label), and $\hat{p}(y|x)$ is the probability of the actual label given the test instance x according to a probabilistic classifier. One point for the formulas (incl. explanations of the symbols), another point for the interpretation. Getting one of the two terms correct (both formula and interpretation) was also counted as one point if the other term was incorrect in some ways.

   (c) *dimension reduction*
   Mapping high-dimensional data (large number of features) to a lower-dimensional representation in a way that retains as much of the "important" (defined in some way) properties or structure of the data. One point for the above definition, another for explaining that this is importance in order to facilitate various machine learning tasks such as classification.

   (d) *linear regression*
   Predicting a continuous outcome y from a set of features $\mathbf{x} = x_1, \ldots, x_p$ by a linear model $\hat{y} = A\mathbf{x}$. One point for giving the above or equivalent formula, another for explaining the setup (input features, continuous outcome).

   (e) *kernel trick*
   For algorithms that depend on the data only through dot products, $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$, the dot product can be replaced by a kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ which is mathematically equivalent to mapping the feature vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ into a (potentially infinite dimensional) feature space and taking dot products in the feature space. One point for explaining this, and another for explaining that the point is to enable more flexible classification and other methods (for example, support vector machines with a non-linear kernel function allow non-linear decision boundaries).

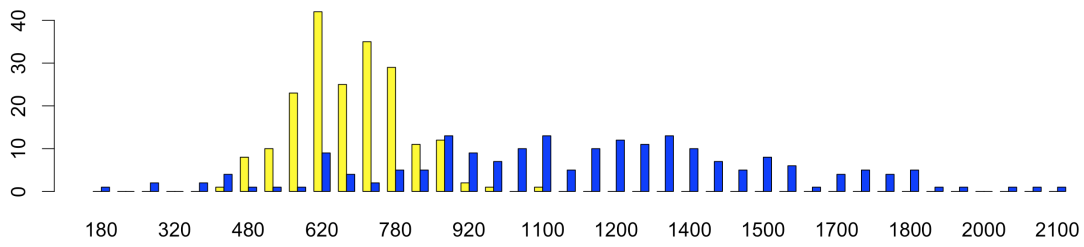
2. [*15 points*] Consider a data set with $n = 100$ observations. Imagine you learn a classification model and find that it classifies all the training examples correctly.

   (a) What can you say about the performance of your classifier on new test data? Explain what makes generalization hard. What properties of the classification method are most relevant?
   It is hard to say much about the performance on test data since the model was fitted to the training data and evaluated on the same data. The main problem is overfitting: the classifier may a) actually be good (also on the test data) or b) be selected from a large set of poor classifiers, some of which just happen to fit the training data well. The most relevant proporties of the classification method are *i*) complexity (the size of the space of possible classification rules) and *ii*) how well the possible classification rules fit the underlying data source. These are closely related to variance (complexity) and bias (fitting the source). 1–5 points depending on how many point from the ones made above were mentioned and how clearly they were expressed.

   (b) Explain cross-validation.
   Split the data into k subsets of as equal size as possible, evaluate performance of supervised classification methods on each subset while using the other $k-1$ subsets as training data, average results. 1–5 point determined case-by-case.

   (c) Now suppose that instead of classification, the task would have been to estimate, for example, the median of an unknown distribution from which we have $n = 100$ data points. How would you apply resampling to measure the accuracy of an estimate computed from the given n points?
   Bootstrap. Resample a number of bootstrap samples by drawing n points with replacement from the original sample, estimate median in each bootstrap sample, consider the variability of the median estimates (e.g., variance or histogram). 1 point for saying "bootstrap", 1 point for explaining the process of creating bootstrap samples by drawing n points with replacement, 1 point for estimating the median in each bootstrap sample, 1 point for summarizing the variability of the estimates, plus 1 point for overall correctness. 2–3 points were also awarded for reasonable altenative solutions instead of bootstrap.

3. [*20 points*] Consider a classification task with one real-valued feature (e.g., some medical test result). Below are two histograms of the feature, X, showing $n = 200$ data point from two classes $Y = 0$ (yellow) and $Y = 1$ (blue).



   (a) Just by looking at the data distributions, how would you classify three test data points with $X = 250$, $X = 500$, and $X = 1000$? What additional information

about the two classes, not contained in the above histograms, would be useful?
$X = 250, 500, 1000$ would be classified as `blue`, `yellow`, tt blue respectively because yellow points are concentrated around $X = 450 - -950$, and blue points are spread further in the tails. Additional information which is not directly available in the class-specific histograms would be the relative frequencies (or probabilities) of the classes themselves: how likely is blue/yellow a prior.

(b) Explain how you would apply Quadratic Discriminant Analysis (QDA) to this task. Draw a diagram to explain the learned model. How would the resulting classifier classify the three test data points in item (a)?
Fitting a QDA classifier would amount to *i*) estimating the class probabilities $p(Y = c)$ for $c \in \{\texttt{blue}, \texttt{yellow}\}$, and *ii*) estimating the one-dimensional feature distributions $p(x|Y = c)$ for each class by fitting a Gaussian distribution to the training data (the histograms in the diagram). The classification decisions would be obtained by comparing the probabilities $p(Y = c)P(x|Y = c)$ for each class, where x is the observed feature value, and picking the maximizing class c. The diagram should show two Gaussian (normal) distributions that match the general shape of the histogram. The classifications for the points $X = 250, 500, 1000$ are probably the same as in the previous item but if the drawing looks otherwise sensible, other classifications were also accepted. A few (but not many) points were awarded to those who persistently presented bivariate Gaussians instead of realizing that the distribution should be one-dimensional since there is only one feature variable.

(c) Compare the QDA classifier in this task (one-dimensional feature X) to the naive Bayes classifier. Also, compare QDA and naive Bayes in the multidimensional case where there are multiple features $X_1, \ldots, X_p$.
Naive Bayes assumes that the feature variables are conditionally independent of each other given the class variable. In this case, as there is only one feature variable, this assumption amounts to nothing at all, and NB = QDA. (Saying that they are "similar" was not accepted as a correct answer.) In the multidimensional case, the NB classifier is easier to train since it has fewer parameters, whereas QDA tends to be better whenever there is plenty of data and/or the feature variables exhibit strong (partial) correlations. Points were awarded on a case-by-case basis: roughly 3 points for stating that NB = QDA in 1D, 1–2 point for mentioning that NB means conditional independence, and 2 points for the comparison in the multidimensional case. (However, note the remark about rescaling above.)

4. [*15 points*]

(a) For what kind of tasks can we use the *K-means algorithm*? Explain carefully what the inputs and outputs of the algorithm are, and give a very brief intuitive explanation of how the results are to be interpreted.
K-means is used for clustering. Input is a pairwise distance matrix (or data and a function to compute distances), and output is a partitioning of the data plus cluster centroids. (One point was deducted if the centroids were not mentioned, which was quite common.) Clusters are sets of point that tend to be more similar to each other and to points in other clusters. Max 3 points.

(b) Describe the actual K-means algorithm (Lloyd's algorithm). The description should be brief and on a high level.

See the course material for a description. Max 4 points. Grading on a case-by-base basis.

(c) Define formally the objective (or cost) function that the K-means algorithm tries to minimise. Comment on how the objective function changes in the two stages of each iteration of Lloyd's algorithm.
The cost function is the sum

$$\sum_{j=1}^{K} \sum_{\mathbf{x} \in D_j} \|\mathbf{x} - \mu_j\|_2^2 = \sum_{i=1}^{n} \|\mathbf{x}_i - \mu_{j(i)}\|_2^2 ,$$

where μ_j is the cluster centroid of the jth cluster, and $j(i)$ indicates the assigned cluster for the ith data point. (There were several solutions suggesting that the cost function is the sum of pairwise distances between each pair of points that are assigned to the same cluster, but this is not correct.) Each stage of Lloyd'd algorithm decreases the cost function or keeps it the same. Points: 2 points for defining the cost function (either one of the above equivalent formulations was sufficient), 2 more for knowing that the two stages of Lloyd's algorithm never increase the cost.

(d) Consider the following set of data points.

Let $K = 3$ and take the three right-most points as initial cluster means (exemplars/prototypes). Simulate the algorithm for a couple of iterations. Draw the cluster assignments and the cluster means after each iteration.
The fisrt iteration groups the five top-most points to a cluster whose centroid is initialized as the top-left point of the three right-most points, and the four bottom-most points to a cluster whose centroid is initialized as the bottom-left point of the three right-most points, and the single right-most point as its own cluster whose centroid is the point itself. The second iteration moves the cluster centroids to the center of the said cluster, and groups the four top-left points as a cluster, the three bottom-left points as another, and the three right-most points as a third. Finally, the centroids are shifted to the centers of these cluster, after which the algorithm terminates. Max 4 points: 1 point for the iterative process, 1 point for assigning points to clusters correctly (or almost correctly), 1 point for moving the centroids (at least almost) correctly, 1 point for overall correctness (determined case by case).