6.891 Fall 1999: Problem Set Grading Guide

For each part of each problem assign one of the following grades:

- $\surd+$: got correct answer and properly justified solution (i.e., "nailed the problem and answer completely correct").
- $\surd$: showed significant work and insight but may not have completely solved the part, or the answer was slightly off (i.e., "got pretty much everything right except made a small error or left off some justification").
- $\surd-$: showed significant work and insight but had a some gaps in justification or answer (i.e., "had the right idea, but got lost somewhere or had a major error").
- 0 : did not show significant work or insight, or answer was not applicable to the question.

6.891 Fall 1999: Problem Set #5 Solutions

Problem 1: Continue preparing for the final project.

Nothing to turn in here, but email us of your idea.

Problem 2: Principal Components Analysis

> **A:** Show that if one of the eigenvalues of the covariance matrix is zero, then all of the data lies in a hyperplane that is perpendicular to the associated eigenvector.

Any hyperplane (of any dimension) which includes all of the data must also include the mean of the data (since the mean is a linear combination of the data points). Thus, we may subtract off the mean from the data and this hyperplane will now pass through the origin. Hence, we will only consider the case of zero-mean data and hyperplanes which pass through the origin (subspaces).

Let's take v to be an eigenvector of the covariance matrix of the data. If the associated eigenvalue is 0, then we know that

$$\begin{aligned} \Sigma v &= 0v && (1) \\ v^T \Sigma v &= 0 && (2) \end{aligned}$$

Remembering that $\Sigma = \sum_i x_i x_i^T$, we have

$$\begin{aligned} v^T \sum_i x_i x_i^T v &= 0 && (3) \\ \sum_i v^T x_i x_i^T v &= 0 && (4) \\ \sum_i \left(v^T x_i\right)\left(x_i^T v\right) &= 0 && (5) \\ \sum_i \left(v^T x_i\right)^2 &= 0 && (6) \end{aligned}$$

Since squares can't be negative, this last line implies that $v^T x_i = 0$ *for all* i which in turn implies that all of the data (when the mean has been subtracted) lies perpendicularly to the direction of v. Thus, it all lies on a hyperplane perpendicular to v.

> **B:** Let D be the dimension of the data, and N the number of data points in a training set. Show that if D is larger than N the covariance matrix must have some zero eigenvectors.

If there are D dimensions in the space and we are considering N points, we know all of these points must lie on a $N-1$ dimensional hyperplane in our D dimensional space. Now let us consider the eigenvectors of the covariance matrix of this data. The maximal eigenvector must lie in this $N-1$ dimensional subspace (if it didn't, you get gain a larger variance by using the normalized projection of the vector onto the subspace). Similarly, the next largest eigenvector must also lie in the $N-1$ dimensional subspace. However, since each eigenvector must be orthogonal to all of the others, we can only get $N-1$ eigenvectors to lie in this subspace. Since $D > N$, there must be $D - N + 1$ other eigenvectors. These eigenvectors must be perpendicular to the subspace (since the first $N-1$ eigenvectors span that space) and therefore their eigenvalues must be 0 (since there is no variance of the data in that direction).

> **C:** Look at the quality of the reconstructions for different numbers of dimensions. Based on the appearance of the characters what is a good compromise between dimensionality and quality?

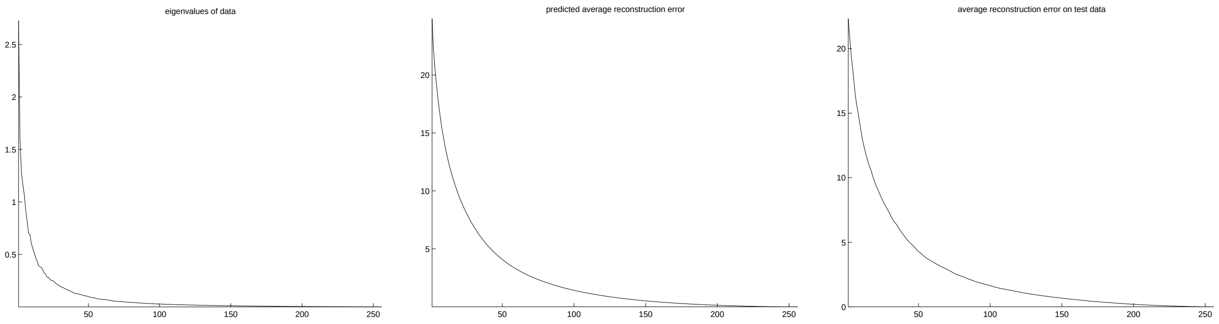

Figure 1: from left to right: the eigenvalues, the sum of the first n eigenvalues, the average reconstruction error on the test data

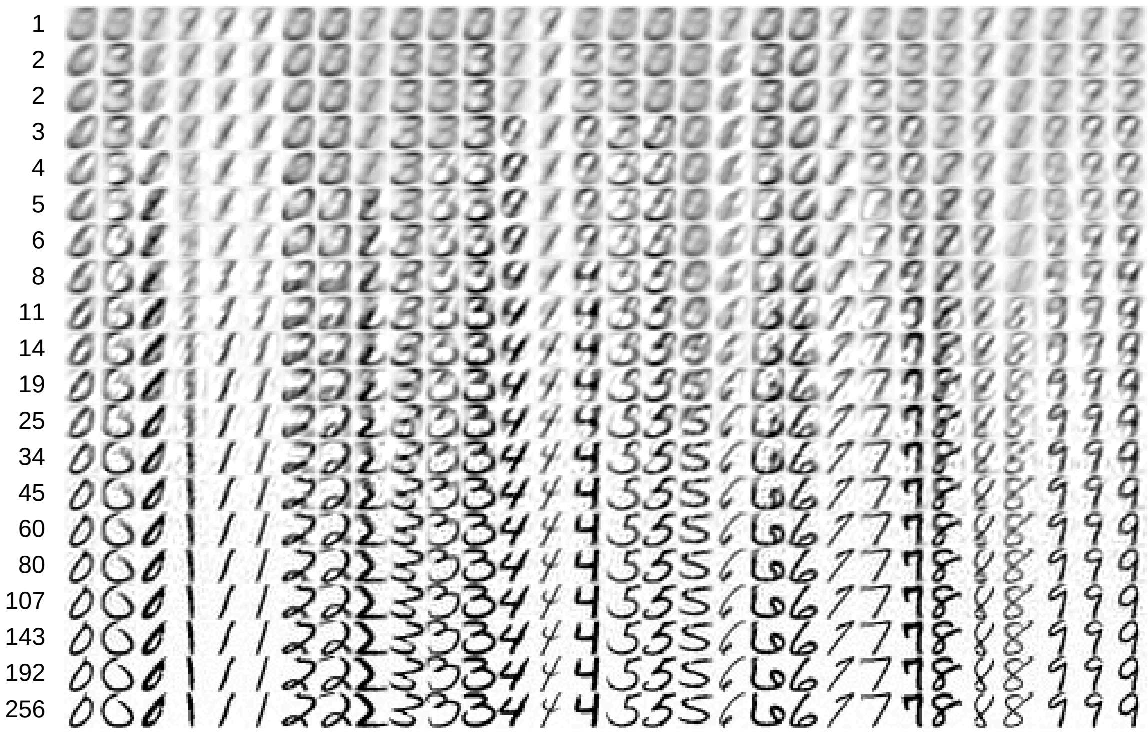

Figure 2: Reconstructions (from left to right) with varying numbers of eigenvectors (top to bottom)

D: Explore the relationship between the dimensionality of the training set and classifications performance. Choose your favorite classifier from problem set 2 or 3 (or your favorite from somewhere else – be sure to describe it clearly). For various numbers of dimensions build a classifier for the digit 2 (i.e. that distinguishes 2's from non-2's). You can use cross validation to pick the optimal number if you are adventurous. Did this number of dimensions agree with your intuitive selection?

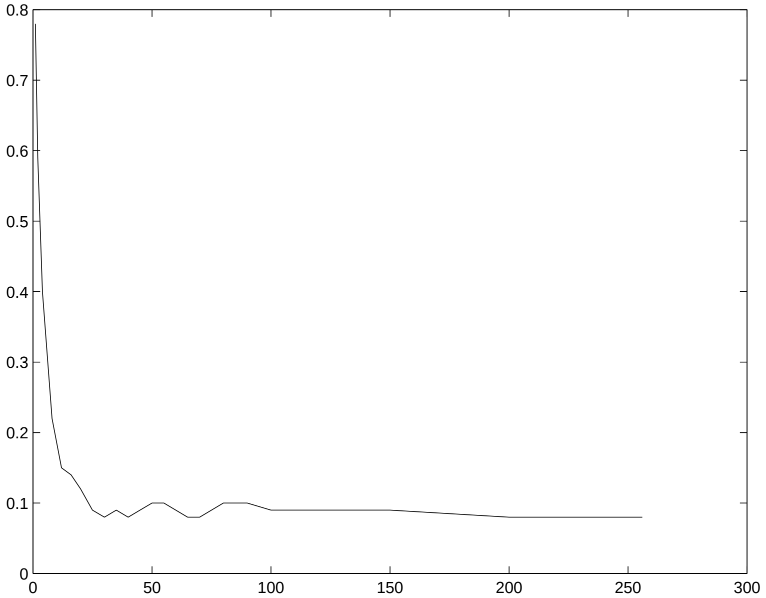

Figure 3: Testing error using a nearest neighbor classifier on the points projected into the first n eigenvectors (where n is plotted across the x-axis).

Problem 3: Independent Components Analysis

A: Run this code on the images provided. Show the resulting images. Note there is a noticeable problem with the result even after finding the optimal unmixing matrix. What is it? Can you modify the original images so that the final unmixed images are improved?

One setting eta=0.01 and batchsize=100 for 100 iterations seems to work. In general eta inversely proportional to the batchsize so that the batch steps taken in the hill-climbing are not too large.

Careful examination of the images shows that the pixels in each image do not quite correspond. It seems that the camera was probably moved slightly between

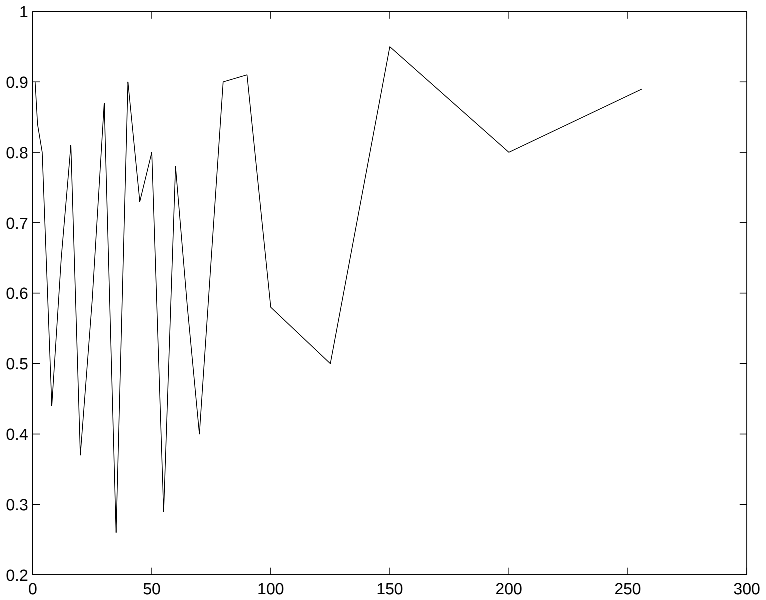

Figure 4: Testing error using a multi-layer perceptron (15 hidden units) on the points projected into the first n eigenvectors. (n is plotted across the x-axis.) The network tended to find rather poor local minima (these are the high points on the curve). If we look at the lower points (presumably where the network didn't converge to a degenerate minimum), we can see a trend of better performance until a point and then worse performance.

exposures. Unfortunately, if you consider what happens when a camera looking at a reflection moves, you notice that if the camera moves to the left, one of the two mixture components moves to the right and the other to the left. However, a slight shift in the pixels of one of the store images does help to align the two images better. Running the same ICA algorithm on the shifted store images produces the results in figure 6. These results are clearly better as they remove the edge artifacts in the previous trial.

Problem 4: Bayes Net Warmup

A: After looking at Figure 1, we can already tell that the joint distribution $P(A, B, C, D)$ has a simplified form. Write out that simplified form as a product of marginal and conditional distributions. It is possible to express this joint distribution using only *marginal distributions*. What is this expression?

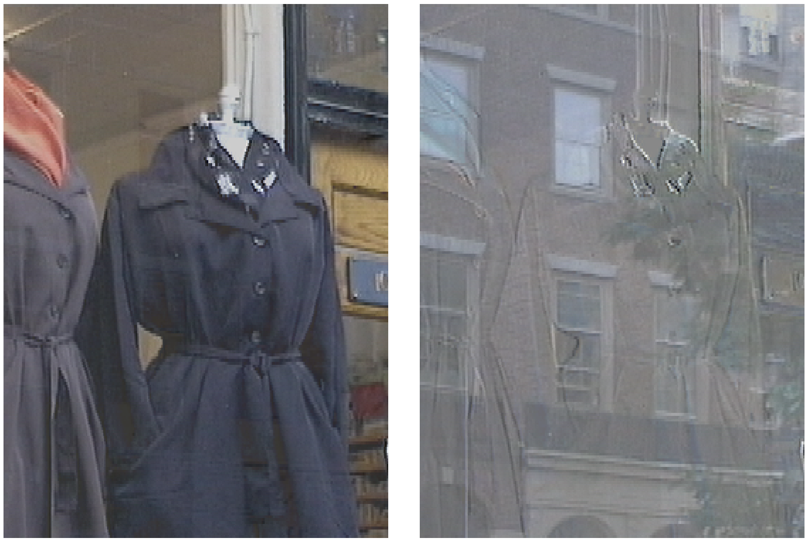

Figure 5: The unmixed storefront images

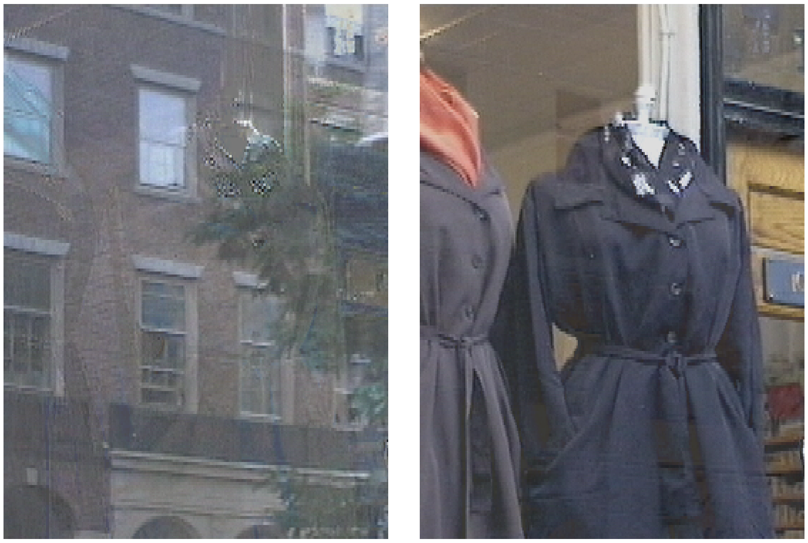

Figure 6: The unmixed storefront images after shifting

The simplified form is $P(A,B,C,D) = P(A)P(C|A)P(D|A,B)P(B)$ according to the construction of the bayes net. We can remove all the conditionals by rewriting them as joints divided by marginals so:

$$\begin{aligned} P(A,B,C,D) &= P(A)P(C|A)P(D|A,B)P(B) \tag{7} \\ &= P(A)\frac{P(C,A)}{P(A)}\frac{P(D,A,B)}{P(A,B)}P(B) \tag{8} \\ &= \frac{P(C,A)P(D,A,B)P(B)}{P(A,B)} \tag{9} \\ & \tag{10} \end{aligned}$$

B: Prove that nodes A and B are independent if we have no knowledge of either C or D.

We can interpret the bayes net as a causal network and use the definitions of d-separation to show that A and B are d-separated if we do not know D. This is because the links are converging and evidence cannot pass through unless we know D. We can also show this by marginalizing the joint expression above over C and D:

$$\begin{aligned} \sum_{C,D} P(A,B,C,D) &= \sum_{C,D} \frac{P(C,A)P(D,A,B)P(B)}{P(A,B)} \tag{11} \\ P(A,B) &= P(A)P(B)\sum_{D}\frac{P(D,A,B)}{P(A,B)} \tag{12} \\ &= P(A)P(B) \tag{13} \\ & \tag{14} \end{aligned}$$

C: Under what conditions are A and B dependent? Why?

Again, interpreting the bayes net as a causal network, A and B are conditionally dependent if we know D. The converging link between A and B through D allows evidence to explain both A and B.

D: Give two conditions under which C and D are independent.

C and D are independent when A is known as this blocks any evidence from C to D in the diverging connection. Also, degenerately, if we observe C and D, then they are also independent.

E: Let G be independent of E given F. Given P(E,F) and P(F,G) what is P(E,F,G)?

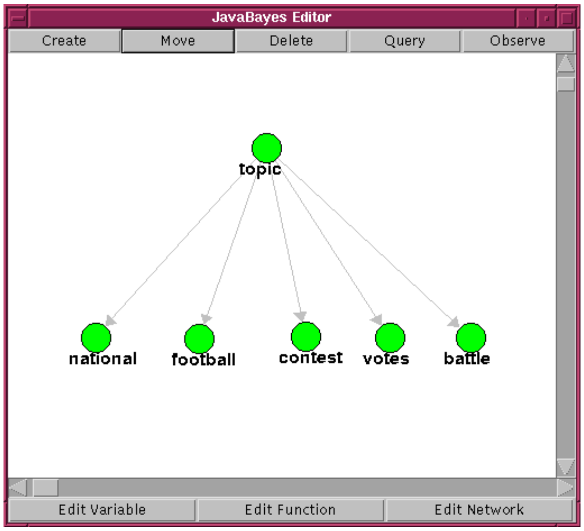

Figure 7: The bayes net for our documents world.

$$
\begin{aligned}
P(E,F,G) &= P(E,F)P(G|F) && (15) \\
&= \frac{P(E,F)P(F,G)}{P(F)} && (16) \\
& && (17)
\end{aligned}
$$

Note that we can get $P(F)$ by marginalizing $P(F,G)$ over G.

Problem 5: Bayes Nets for Text

A: Setup this problem up as a Bayes net in either JavaBayes or the Bayes Net Toolkit. Either take a screen dump of JavaBayes or show us your code from the Bayes Net Toolkit.

For our two-document world, we have one node for the "topic" of politics or sports with priors 2/3 and 1/3 respectively. This cause node then generates the rest of the words. Figure 7 shows the bayes net.

B: What is the most likely set of words to occur in a political document? What about sports?

For a political document, each word appears less than half of the time, so it's more likely that none of the words appear. For a sports document, we have a similar situation except "national" should occur since it happens more than half of the time.

> **C:** What set of words is most likely to be a sports document? What is the probability of this document being a sports document?

For the set of words most likely to be about sports, we want words that are most probable for sports but least probably for political documents because this will maximize the likelihood ratio. We see that "national", "football", and "battle" are more likely for sports than politics. Observing those nodes as occuring and the other words as not occuring in the bayes net and querying the topic node gives $P(sports) = 0.97$.

> **D:** Given that the words national and football appear in a document, what is the probability that the word votes will appear?

By observing the "national" and "football" nodes in the bayes net and querying the "votes" note, we get $P(votes) = 0.0575$.