Data Mining: 36-462/36-662
Homework 3

Due Thursday March 7 2013
(at the beginning of lecture)

Append your R code to the end of your homework. In your solutions, you should just present your R output (e.g., numbers, table, figures) or snippets of R code as you deem it appropriate. Make sure to present your results (i.e., your R ouput) in a clear and readable fashion. Careless or confusing presentations will be penalized.

Problem 1

Using R, create an example of 100 points in 2 dimensions that obey a 1-dimensional structure, i.e., the points lie along a smooth curve. Color the points by their intrinsic order along this curve, e.g., with colors from `rainbow(100)`.

Your example should be a situation in which regular principal component analysis will fail—i.e., the first principal component score fails to properly unravel the points according to their order on the curve. (The coloring is used to visualize this ordering.)

By passing the appropriate distance matrix to multidimensional scaling (not just Euclidean distances! this will just give us back principal component analysis), show that this method can produce a 1-dimensional representation of your curve, such that the points are in the correct order. (Again, this ordering is demonsrated by the coloring).

Your write up should include, in addition to a short exlanation of what you did and why it worked, 3 plots: the original 2-dimensional data, the first principal component score, and the 1-dimensional representation returned by multidimensional scaling (applied to your custom distances).

(Hint: you have total control over your example, so create an example where it is obvious to you what the "right" distance metric is to pass to multidimensional scaling.)

Problem 2

Given $X, Y \in \mathbb{R}$, suppose that $Z = (X, Y) \in \mathbb{R}^2$ has a bivariate normal distribution, $Z \sim N(\mu, \Sigma)$, where $\mu \in \mathbb{R}^2$ and $\Sigma \in \mathbb{R}^{2\times 2}$. The density of $Z = (X, Y)$ is

$$f_{X,Y}(z) = \frac{1}{2\pi\sqrt{\det(\Sigma)}} \exp\Big(-\frac{1}{2}(z-\mu)^T\Sigma^{-1}(z-\mu)\Big).$$

(a) Argue that we can always write

$$\Sigma = \begin{pmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{pmatrix},$$

where $\sigma_X^2 = \mathrm{Var}(X), \sigma_Y^2 = \mathrm{Var}(Y)$, and $\rho = \mathrm{Cor}(X, Y)$.

(b) Express the density $f_{X,Y}$ at a point $z = (x, y)$ explicitly in terms of x and y, i.e., not involving any matrix-vector multiplications.

(Hint: for a 2×2 matrix, you can write out its determinant and inverse explicitly.)

(c) Show that when $\rho = 0$, the joint density factors as $f_{X,Y}(x, y) = f_X(x) f_Y(y)$ where f_X, f_Y are the marginal densities of X, Y, respectively, and hence X, Y are independent.

Problem 3

Download the file "smoother.R" from the course website, and load it into your R session using `source("smoother.R")`. Now you should have the function `smoother`. This function takes `x` and `y` as arguments, which are the vectors of independent and dependent observations, respectively. It smooths `y` on `x` and returns the vector of fitted values.

(a) You're going to write an R function to perform the alternative conditional expectations (ACE) algorithm. Your function should look like:

```
my.ace = function(x, y, tol=1e-6, maxiter=500) {
  fx = x-mean(x)
  fx = fx/sqrt(sum(fx^2))
  gy = y

  # Your code goes here, to build fx, gy, maxcor, iter

  return(list(fx=fx,gy=gy,maxcor=maxcor,iter=iter))
}
```

The function takes arguments:

- `x,y`: vectors of observations, whose maximal correlation we want to compute.
- `tol`: if the absolute difference in the correlation of `fx,gy` is smaller than `tol` across successive iterations, then we quit.
- `maxiter`: the maximum number of iterations before quitting.

The function returns a list with elements:

- `fx,gy`: the optimal transformations of `x,y`, respectively, as determined by your ACE algorithm.
- `maxcor`: the maximal correlation of `x,y`, i.e., the correlation of `fx,gy`.
- `iter`: the number of iterations performed by your ACE algorithm.

Remember from lecture that the functions `fx,gy` should be centered and scaled at each iteration, i.e., these vectors should be centered to have mean zero and scale to have sum of squares equal to one.

Because writing this function is the point of this part of the problem, you should give your R code for the `my.ace` function as your solution (i.e., don't just append it at the end of your homework).

Download the file "hw3prob3.Rdata" from the course website and load it into your R session using `load("hw3prob3.Rdata")`. Now you should have a list `ace.data`, that contains 8 elements, each of which is a data set. These are `perf.lin`, `perf.quad`, `perf.cubic`, `perf.circle`, `noisy.lin`, `noisy.indep`, `noisy.pwcubic`, `noisy.checker`. Each of these is in turn a list with two elements, `x` and `y`. So, e.g., the data for the perfect linear dat set can be accessed using `ace.data$perf.lin$x` and `ace.data$perf.lin$y`.

(b) Run `my.ace` on each of the perfect data sets (first 4 data sets in `ace.data`). For each data set, report the maximal correlation. Also for each data set, produce a figure of 4 plots (using `par(mfrow=c(2,2))`), where the top left shows the data (i.e., `x` vs `y`), the top right shows the transformed data (i.e., `fx` vs `gy`), the bottom left shows the transformation of `x` (i.e., `x` vs `fx`), and the bottom right shows the transformation of `y` (i.e., `y` vs `gy`). Briefly comment on the transformations for each data set. Do they make sense?

(c) Repeat (b) for the noisy data sets (last 4 data sets in `ace.data`). What in particular do you notice about the transformations for the `noisy.lin` data set? Also, what is the reported maximal correlation for the `noisy.checker` data set? Looking at the transformations from the ACE algorithm, explain why this happened. Is this a desirable outcome?

Problem 4

In this problem you're going to prove the Gauss-Markov theorem. Gauss apparently proved this when we he was 18.[1] You're probably older than 18, but still, you can someday brag to your grandkids that at a young age you proved a fairly fundamental result in statistics.

Recall that the theorem assumes that we observe a vector $y \in \mathbb{R}^n$ of observations from the model

$$y = X\beta^* + \epsilon,$$

where $X \in \mathbb{R}^{n \times p}$ is a fixed matrix of predictor variables, $\beta^* \in \mathbb{R}^p$ are the true coefficients, and $\epsilon \in \mathbb{R}^n$ are random errors, with

$$\mathrm{E}[\epsilon] = 0, \quad \mathrm{Cov}(\epsilon) = \sigma^2 I.$$

Given a vector $a \in \mathbb{R}^p$, we consider linear unbiased estimates of $a^T\beta^*$. That is, we consider estimates of the form $c^T y$ such that $\mathrm{E}[c^T y] = a^T\beta^*$. Note that the regression estimate $a^T\hat{\beta}$ is both linear and unbiased, as

$$a^T\hat{\beta} = a^T(X^TX)^{-1}X^Ty = \big(X(X^TX)^{-1}a\big)^Ty = b^Ty,$$

[1] Although, he proved a slightly weaker version of the result in which the errors are assumed to be normally distributed.

and $\mathrm{E}[a^T\hat{\beta}] = a^T\mathrm{E}[\hat{\beta}] = a^T\beta^*$. As our criterion we use mean squared error; for any estimate c^Ty of $a^T\beta^*$, its mean squared error is

$$\mathrm{MSE}(c^Ty) = \mathrm{E}[(c^Ty - a^T\beta^*)^2].$$

The Gauss-Markov theorem states that $a^T\hat{\beta}$ is the best linear unbiased estimate (BLUE) of $a^T\beta^*$ in terms of mean squared error, that is,

$$\mathrm{MSE}(a^T\hat{\beta}) \leq \mathrm{MSE}(c^Ty)$$

for any other linear unbiased estimate c^Ty of $a^T\beta^*$. You will prove this in several steps.

(a) Prove that $\mathrm{E}[y] = X\beta^*$ and $\mathrm{Cov}(y) = \sigma^2 I$.

(b) Prove that if $\mathrm{E}[c^Ty] = a^T\beta^*$ holds for any vector $\beta^* \in \mathbb{R}^p$, then we must have $X^Tc = a$.

(c) Let c^Ty be an unbiased estimator of $a^T\beta^*$. Prove that $\mathrm{MSE}(c^Ty) = \sigma^2\|c\|_2^2$, and hence

$$\mathrm{MSE}(c^Ty) = \sigma^2(\|c^*\|_2^2 + \|c - c^*\|_2^2) \geq \sigma^2\|c^*\|_2^2,$$

with equality if and only if $c = c^*$, where $c^* = P_{\mathrm{col}(X)}\, c$, the projection of c onto the column space of X.

(Hint: If c^Ty is unbiased, then $\mathrm{MSE}(c^Ty) = \mathrm{Var}(c^Ty)$.)

(d) Prove that $(c^*)^Ty = a^T\hat{\beta}$, and conclude that $\mathrm{MSE}(a^T\hat{\beta}) \leq \mathrm{MSE}(c^Ty)$.

(Hint: Start with $(c^*)^Ty = (c^*)^T(\hat{y} + r)$ where $\hat{y}$ is the linear regression fit and $r = y - \hat{y}$ is the residual. Then apply the result of part (b) to c^*.)

Bonus problem

Given $x, y \in \mathbb{R}^n$, let $A, B \in \mathbb{R}^{n\times n}$ denote the pairwise distances matrices:

$$A_{ij} = |x_i - x_j|, \quad B_{ij} = |y_i - y_j|, \quad i, j = 1, \ldots n.$$

Also let $\tilde{A}, \tilde{B}$ denote the double-centered (i.e., both row- and column-centered) versions of A, B,

$$\tilde{A} = (I - M)A(I - M), \quad \tilde{B} = (I - M)B(I - M),$$

where $M \in \mathbb{R}^{n\times n}$ is the matrix of all $(1/n)$s. In lecture 12, we defined the sample version of the (squared) distance covariance between x and y as

$$\mathrm{dcov}^2(x, y) = \frac{1}{n^2}\sum_{i,j=1}^n \tilde{A}_{ij}\tilde{B}_{ij},$$

and we claimed that

$$\frac{1}{n^2}\sum_{i,j=1}^n \tilde{A}_{ij}\tilde{B}_{ij} = \frac{1}{n^2}\sum_{i,j=1}^n A_{ij}B_{ij} - \frac{1}{n}\sum_{j=1}^n A_{\cdot j}B_{\cdot j} - \frac{1}{n}\sum_{i=1}^n A_{i\cdot}B_{i\cdot} + A_{\cdot\cdot}B_{\cdot\cdot}, \tag{1}$$

where $\cdot$ denotes a sum over the appropriate component, i.e.,

$$A_{i\cdot} = \sum_{j=1}^{n} A_{ij}, \quad A_{\cdot j} = \sum_{i=1}^{n} A_{ij}, \quad A_{\cdot\cdot} = \sum_{i,j=1}^{n} A_{ij},$$

and similarly for B. In this problem you will prove this claim.

(a) Show that $\sum_{i,j=1}^{n} \tilde{A}_{ij}\tilde{B}_{ij} = \text{trace}(\tilde{A}^T\tilde{B})$, where recall trace gives the sum of the diagonal elements.

(b) Show that M and $I-M$ are both symmetric and idempotent, i.e., $M^T = M$, $(I-M)^T = I-M$, $M = M^2$, $I-M = (I-M)^2$.

(c) Plug in $\tilde{A} = (I-M)A(I-M)$ and $\tilde{B} = (I-M)B(I-M)$ to show that

$$\begin{aligned}\text{trace}(\tilde{A}^T\tilde{B}) = &\text{trace}(A^TB) - \text{trace}(MA^TBM) - \text{trace}(A^TMMB) \\ &+ \text{trace}(MA^TMMBM).\end{aligned}$$

(Hint: use part (b), and the fact that you can commute the product of two matrices under the trace operation.)

(d) Divide the result in part (c) by n^2, and expand the right-hand side to prove the claim in (1).