36-350: Data Mining

Lab 7
Date: October 10, 2003 **Due: end of lab**

Interspersed throughout this lab are questions that you will have to answer at check-off.

1. Download the files for this lab from the course web page to the desktop:

   `http://www.stat.cmu.edu/~minka/courses/36-350/lab/`

2. Open a Word or Notepad document to record your work.

Start R

3. `Start -> All Programs -> Class software -> R 1.7.0`

4. Load the special functions for this lab:

   `File -> Source R code...`

   Browse to the desktop and pick `lab7.r` (it may have been renamed to `lab7.r.txt` when you downloaded it). Another window will immediately pop up for you to pick the `mining.zip` file you downloaded.

The dataset

5. The dataset is is 506 neighborhoods in Boston, each described by 11 characteristics:

Crime	per capita crime rate
Industry	proportion of non-retail business acres
Pollution	nitrogen oxides concentration (parts per 10 million)
Rooms	average number of rooms per dwelling
Old	proportion of owner-occupied units built prior to 1940
Distance	weighted mean of distances to five Boston employment centers
Highway	index of accessibility to radial highways
Tax	full-value property-tax rate per $10,000
Student.Teacher.Ratio	student-teacher ratio
Low.Status	percent of the population which is 'lower status'
Price	median value of owner-occupied homes in $1000

   Load it via

   ```
   data(Housing)
   ```

   This defines a matrix called `Housing`. Look at the first few rows via `Housing[1:3,]`.

Standardizing

6. `Crime`, `Distance`, `Low.Status`, and `Pollution` all have skewed distributions and need to be transformed. Three require logarithm and one requires square root. *Which one?*

7. Transform these attributes (as in lab 5) and standardize to zero mean and unit variance. Make histograms to check that it worked. Let `x` be the transformed data.

Contour plot

8. Make scatterplots with trend line showing how `Price` depends on `Distance` and `Low.Status`. Scale the window so that the plots are not too stretched out. Keep a copy of this, and all later plots, for the homework.

9. Make a contour plot which shows how `Price` depends on `Distance` and `Low.Status`. Use 8 contour lines. (A slice plot may also be helpful.)

Projections

10. Project the data into two dimensions using PCA (lab 5). Plot the projected data as black dots and include axis arrows.

11. Project the data into two dimensions using m-projection (lab 6) to separate `Price` groups. That is, let `f = x[,"Price"]`. You should compute the projection `w` with the non-`Price` attributes only. A matrix excluding `Price` can be constructed via

    ```
    x.pred = not(x,"Price")
    ```

    Save the projection matrix `w`, rounded for easy reading:

    ```
    round(w,1)
    ```

12. Make a contour plot showing how `h1` and `h2` (in the m-projection) predict `Price`. Include axis arrows and make sure the aspect ratio is 1.

13. You can now get checked off.

Scatterplots with trend line If `x` is a matrix with columns `r`, `p1`, and `p2`, this is a convenient command for making multiple scatterplots with trend lines (also see lab 5):

```
predict.plot(r ~ p1 + p2,x)
```

Contour plot If `x` is a matrix with columns `r`, `p1`, and `p2`:

```
fit = smooth(r ~ p1 + p2, x, span=.5)
color.plot(fit,n=8)
```

The first line fits a regression surface (recall lab 5). `span=.5` controls the smoothness. The second line makes the plot, using `n=8` colors.

Slice plot

```
slices(r ~ p1 | p2, x, n=2)
```

Projection The `project` function has the useful feature that columns not named in `w` will not be altered. Thus if you project a matrix including `Price` but `Price` is not named in `w`, `Price` will remain as a column in `px`.