# Homework 4

36-350: Data Mining

Due at the start of class Friday, October 10, 2008

The following problems apply the methods we have developed so far to a medical problem, gene expression in cancer. Except in the last problem, you do not have to write any code, though you do have to interpret code output. In some questions there are calculations which you may find faster to do with R than by hand.

*Biological background*: A gene is a stretch of DNA inside the cell that tells the cell how to make a specific protein. All cells in the body contain the same genes[1], but they do not always make the same proteins in the same quantities; the genes have different **expression levels** in different cell types, and cells can **regulate** gene expression levels in response to their environment. Different types of cells thus have different expression profiles. Many diseases, including cancer, fundamentally involve breakdowns in the regulation of gene expression. The expression profile of cancer cells becomes abnormal, and different kinds of cancers have different expression profiles.[2]

Our data are gene expression measurements from cells drawn from 64 different tumors (from 64 different patients). In each case, a device called a **microarray** (or **gene chip**) measured the expression of each of 6830 distinct genes[3], essentially the logarithm of the chemical concentration of the gene's product. Thus, each record in the data set is a vector of length 6830. (The website for the data is `http://www-stat.stanford.edu/~tibs/ElemStatLearn/datasets/nci.info`.)

The cells mostly come from known cancer types, so there are classes, in addition to the measurements of the expression levels. The classes are BREAST, CNS (central nervous system), COLON, LEUKEMIA, MELANOMA, NSCLC (non-small-cell lung cancer), OVARIAN, PROSTATE, RENAL, K562A, K562B, MCF7A, MCF7D (those three are laboratory tumor cultures) and UNKNOWN.

[1] Except, oddly, red blood cells, which do not contain any DNA.

[2] A very good place to begin learning more is, in all seriousness, *The Cartoon Guide to Genetics*, by Larry Gonick and Mark Wheelis.

[3] Strictly speaking, genes are first "transcribed" into RNA sequences, which are then "translated" into proteins, and gene chips really measure the RNA level, not the protein level. The difference can be important, but we will not get into that here.

1. *Similarity searching for cells*

   (a) How many features are there in the raw data? Are they discrete or continuous? (If there are some of each, which is which?) If some are continuous, are they necessarily positive?

   (b) Describe a method for finding the $k$ cells whose expression profiles are most similar to that of a given cell. Carefully explain what you mean by "similar". How could you evaluate the success of the search algorithm?

   (c) Would it make sense to weight genes by "inverse cell frequency"? Explain.

At this point, I randomly selected 100 of the genes, and ran further analyses use only those features, ignoring the others. (This was done to speed up the calculations, and is not really a good idea.) Problems 2–6 refer to this restricted set of features.

2. The file `nci.kmeans` records the true cell type and the result of running the $k$-means algorithm on the data three different times, each time with $k = 14$. (The R command for this is `kmeans`.)

   (a) For each run, calculate the number of errors made by $k$-means, i.e., how many pairs of cells of the same type are in different clusters, and how many pairs of cells of different types are in the same cluster.

   (b) Is one kind of error more frequent in all three runs?

   (c) Are there any classes which seem particularly hard for $k$-means to pick up?

   (d) Are there any pairs of cells which are always clustered together, and if so, are they of the same class?

3. Figures 1 and 2 show hierarchical clusterings of the data. (The R command used was `hclust`, with `method` set to `ward` or `single`.)

   (a) Which cell classes seem to be best captured by each clustering method?

   (b) Figure 3 shows the merging-cost curve for Ward's method. How many clusters does this suggest?

   (c) Which method does a better job of capturing the cell classes? (Justify your answer.)

   (d) Suppose you did not know the cell classes. Can you think of any reason to prefer one clustering method over another here, based on their outputs and the rest of what you know about the problem?

4. Figure 4 shows the fraction of the total variance ($R^2$) described by the first $k$ principal components, as a function of $k$.

   (a) How (if at all) is this different from a scree plot?

(b) Roughly how much variance is captured by the first two principal components?

(c) Roughly how many components must be used to capture half of the variance? To capture nine-tenths?

(d) Is there any point to using more than 50 components? (Justify your answer.)

(e) How much confidence should you have in results from visualizing the first two principal components?

5. Figure 5 shows the projection of the 64 cells on to the first two principal components.

   (a) One tumor class (at least) forms a cluster in the projection. Say which, and explain your answer.

   (b) Identify a tumor class which does *not* form a compact cluster.

   (c) Of the two classes of tumors you have just named, which will be more easily classified with the prototype method? With the nearest neighbor method?

6. The file `nci.pca2.kmeans` contains the results of running the $k$-means algorithm three times on the PCA-projected data, with $k = 14$. (As the name indicates, this used just the first two principal components.)

   (a) Calculate the number of errors, as with the $k$-means clustering based on all 100 genes.

   (b) Are there any pairs of cells which are always clustered together? If so, do they have the same cell type?

   (c) Does $k$-means find a cluster corresponding to the cell type you thought would be especially easy to identify in the previous problem? (Explain your answer.)

   (d) Does $k$-means find a cluster corresponding to the cell type you thought be be especially hard to identify?

7. Install the library `ElemStatLearn` from CRAN and call up the data set with the command `data(nci)`. This is set up so that each *column* is a different cell, and the column names are the cell classes. You will want to transpose this so that the cells are rows and the genes are columns.

   (a) Use `prcomp` to do a principal components analysis of the *complete* set of genes. What are the first eight eigenvalues? Should you set `scale.=TRUE`? Explain your reasoning.

   (b) Write a function to print cell class labels against the projections on to the first two components. How different does the output look from Figure 5? Are the same clusters visible in your plot as in the figure?

*Hints:* Running `prcomp` with the option `retx=TRUE` will include the projections of the data points in the return value (though it's not printed by default — see `help(prcomp)`). Also, look at the code in Lecture 14 for plotting states' names against their latitude and longitude. Alternately, you may be able to achieve the same effect by manipulating the options of the `biplot` command (see `help(biplot)`).

(c) Calculate the error rate of the prototype method using *all* the gene features, using the leave-one-out estimate. You may use the solution code from homework 2, or your own code. (Note: this may take several minutes to run. [Why so slow?])

(d) Calculate the error rate of the prototype method using the first two, ten and twenty principal components. Include the R commands you used to do this.

(e) (Extra credit.) Plot the error rate, as in the previous part, against the number $q$ of principal components used, for $q$ from 2 to 64. Include your code and comment on the graph. (*Hint*: Write a function to calculate the error rate for arbitrary $q$, and use `sapply`.)

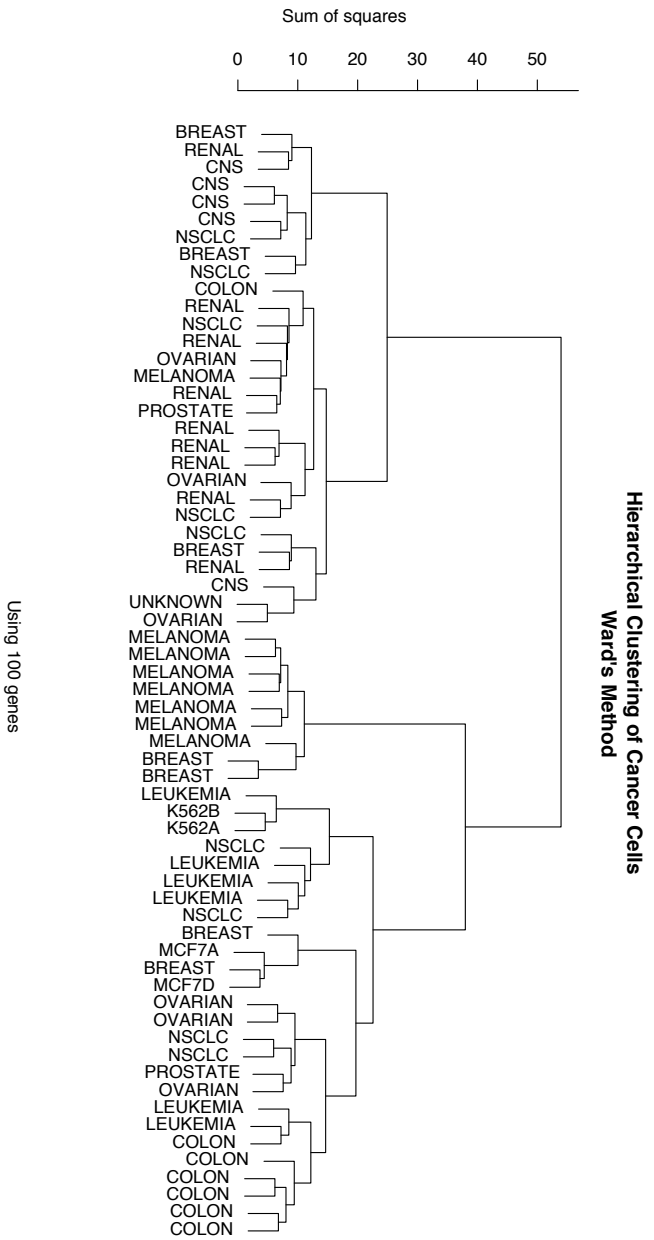
Sum of squares
0
10
20
30
40
50
BREAST
RENAL
CNS
CNS
CNS
CNS
NSCLC
BREAST
NSCLC
COLON
RENAL
NSCLC
RENAL
OVARIAN
MELANOMA
RENAL
PROSTATE
RENAL
RENAL
RENAL
OVARIAN
RENAL
NSCLC
NSCLC
BREAST
RENAL
CNS
UNKNOWN
OVARIAN
MELANOMA
MELANOMA
MELANOMA
MELANOMA
MELANOMA
MELANOMA
MELANOMA
BREAST
BREAST
LEUKEMIA
K562B
K562A
NSCLC
LEUKEMIA
LEUKEMIA
LEUKEMIA
NSCLC
BREAST
MCF7A
BREAST
MCF7D
OVARIAN
OVARIAN
NSCLC
NSCLC
PROSTATE
OVARIAN
LEUKEMIA
LEUKEMIA
COLON
COLON
COLON
COLON
COLON
COLON
Using 100 genes
Hierarchical Clustering of Cancer Cells
Ward's Method

Figure 1:

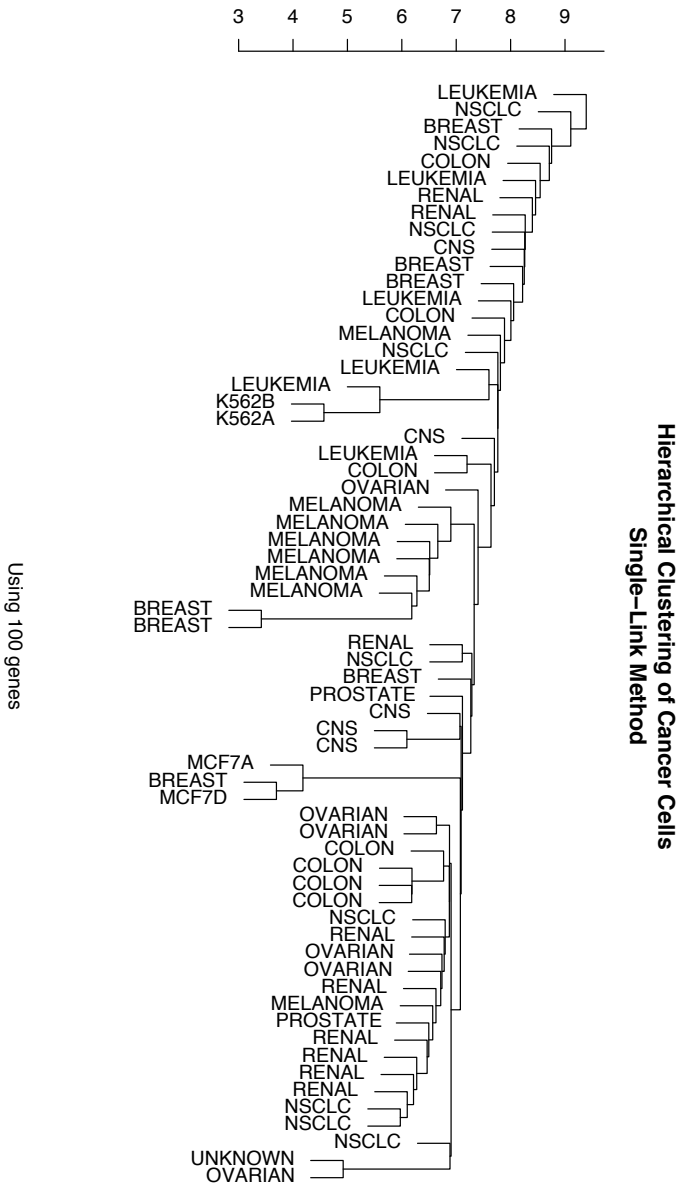
Hierarchical Clustering of Cancer Cells
Single−Link Method
Using 100 genes
3
4
5
6
7
8
9
LEUKEMIA
NSCLC
BREAST
NSCLC
COLON
LEUKEMIA
RENAL
RENAL
NSCLC
CNS
BREAST
BREAST
LEUKEMIA
COLON
MELANOMA
NSCLC
LEUKEMIA
LEUKEMIA
K562B
K562A
CNS
LEUKEMIA
COLON
OVARIAN
MELANOMA
MELANOMA
MELANOMA
MELANOMA
MELANOMA
MELANOMA
BREAST
BREAST
RENAL
NSCLC
BREAST
PROSTATE
CNS
CNS
CNS
MCF7A
BREAST
MCF7D
OVARIAN
OVARIAN
COLON
COLON
COLON
COLON
NSCLC
RENAL
OVARIAN
OVARIAN
RENAL
MELANOMA
PROSTATE
RENAL
RENAL
RENAL
RENAL
NSCLC
NSCLC
NSCLC
UNKNOWN
OVARIAN

Figure 2:

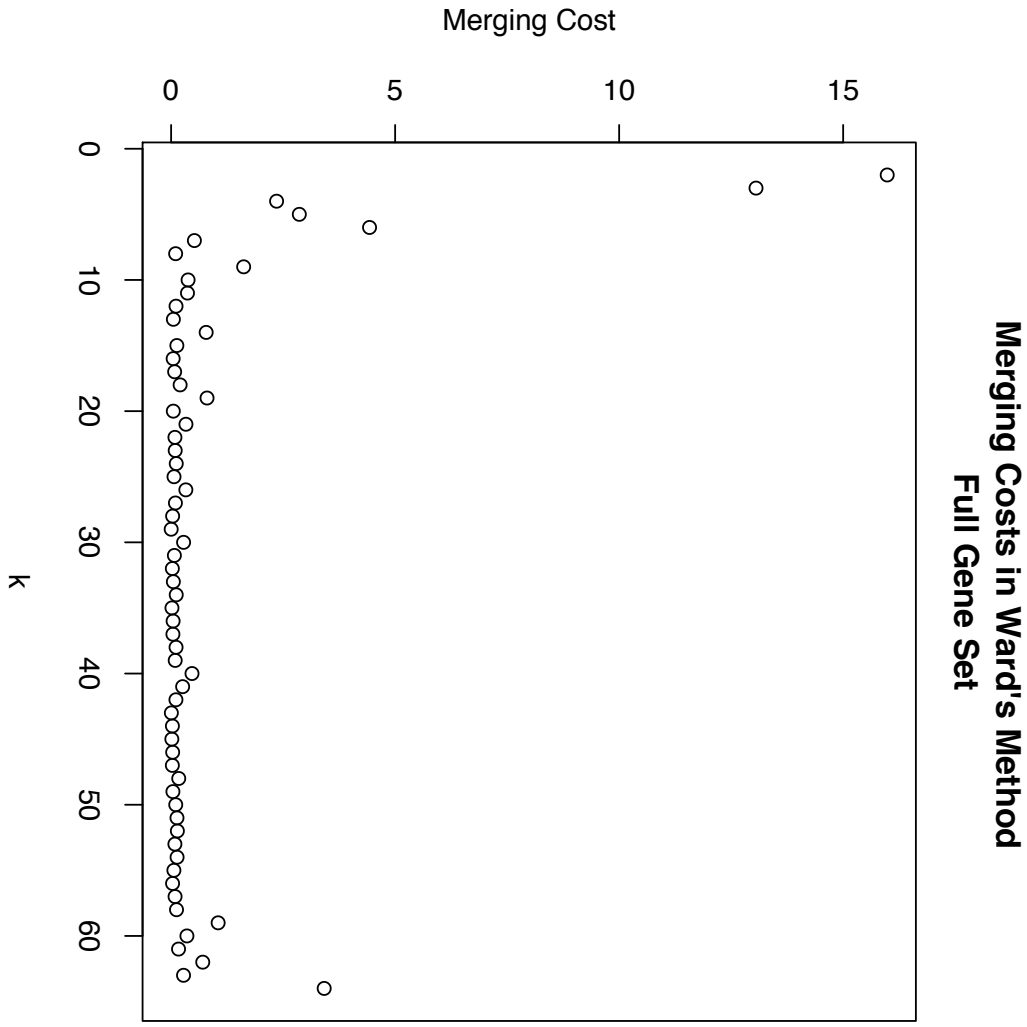
Merging Cost
0
5
10
15
k
0
10
20
30
40
50
60
Merging Costs in Ward's Method
Full Gene Set

Figure 3:

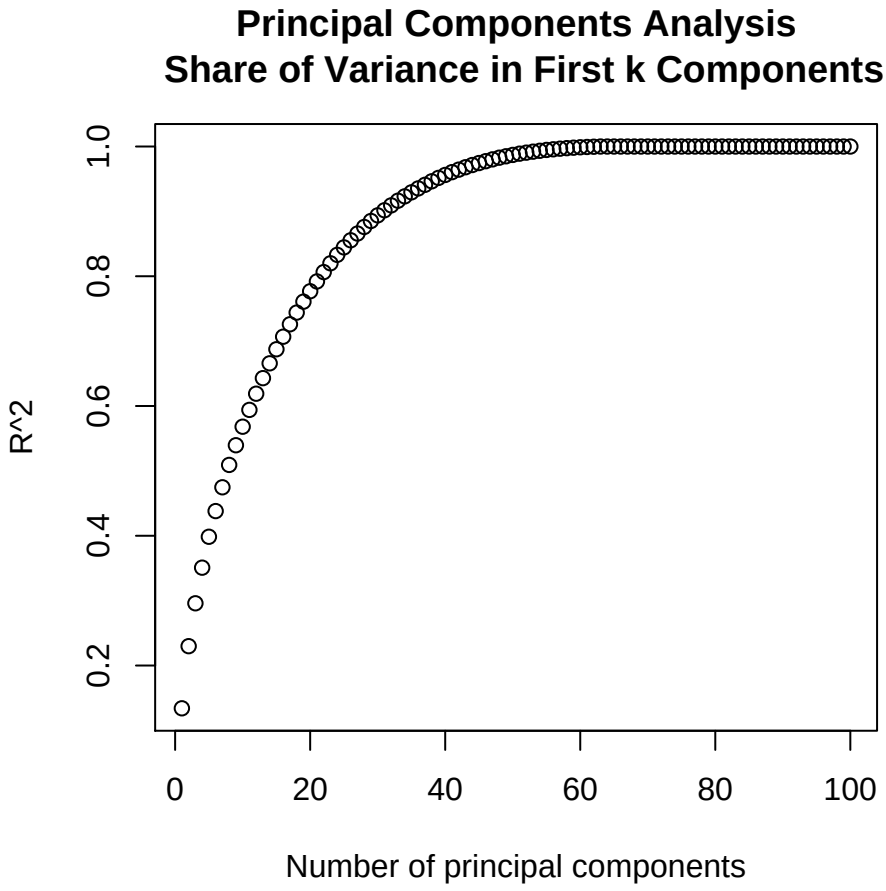
Principal Components Analysis
Share of Variance in First k Components
R^2
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
Number of principal components

Figure 4:

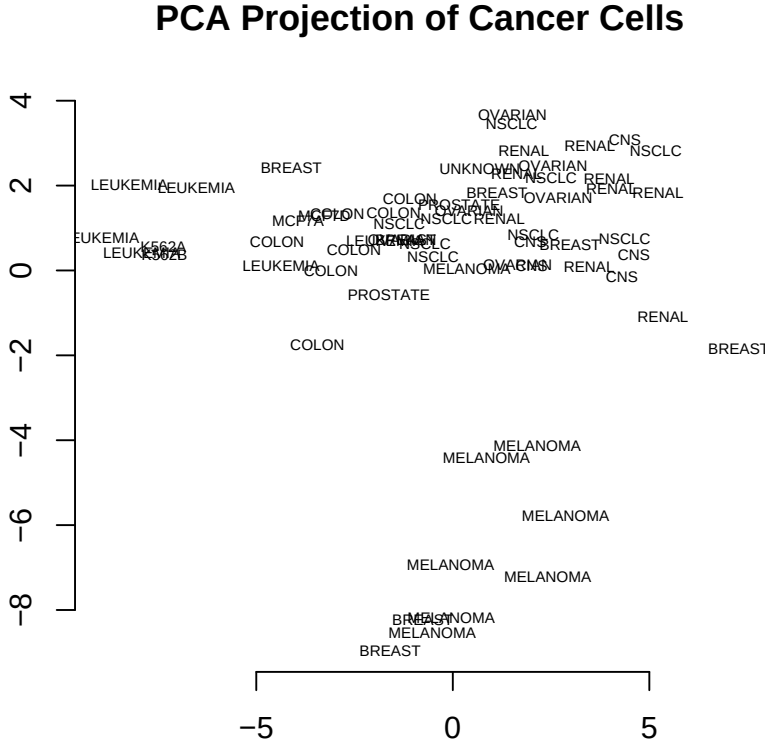
PCA Projection of Cancer Cells
4
2
0
−2
−4
−6
−8
−5
0
5
OVARIAN
NSCLC
RENAL
CNS
BREAST
LEUKEMIA
COLON
PROSTATE
MELANOMA

Figure 5: