Computational Molecular Medicine

Assignment 4

Due April 8

As discussed at length in class, one of the most intense research areas in computational molecular medicine over the past fifteen years is the prediction of cellular phenotypes, e.g., properties of cancerous growths, based on gene expression profiles. This assignment concerns a very important and largely unsolved problem in the treatment of breast cancer, namely predicting the time of relapse among patients diagnosed with a malignant tumor. In particular, we only consider patients with ER (estrogen receptor) positive disease, and who were treated with surgery or surgery plus radiation, and untreated with systemic hormonal therapy and/or chemotherapy. Of these, we consider two categories of patients, labeled "NoRelapse" ($Y = 0$) and "Relapse" ($Y = 1$). "NoRelapse" refers to patients who did not relapse at all and were in the study for at least 6.5 years. The "Relapse" patients are those who relapsed BEFORE 6.5 years. These are "extreme" cases; there were indeed patients relapsing AFTER 6.5 years, for example at 8 or 12 years, but they have been excluded.

This is an important problem since it is a snapshot of the natural evolution of tumors that are caught early, are small, and have not yet apparently seeded any metastasis. Some patients will do very well without any treatment, staying alive for a long time and eventually dying of something else, whereas some patients have an aggressive cancer and would definitely benefit from adding systemic hormonal therapy and/or chemotherapy to their treatment.

There are four TAB delimited text files:

1) brcaTrainExpr.txt
2) brcaTestExpr.txt
3) brcaTestPheno.txt
4) brcaTrainPheno.txt

Files 1) and 2) are gene expression matrices, each containing 22215 rows (genes) and 213 columns (212 samples + Gene Name Column). Files 3) and 4) are phenotype tables, each containing 212 rows (samples) and 2 columns (GEO-ACCESSION, RelapseGroup). The training set has 152 NoRelapse and 60 Relapse patients. The test set has 137 NoRelapse and 75 Relapse patients.

Apply any statistical learning technique to learn a predictor for distinguishing between "NoRelapse" and "Relapse" individuals. Your method can be one you learned in the course, or one described in the Statistical Learning chapter (but not covered in class), or one you learned anywhere else, or even a new method you invent yourself. Your effort will be evaluated by various criteria, including creativity, (mathematical) coherence, parsimony and proper validation. Obtaining high accuracy is also valuable, but is very difficult and not necessary to do a good job. You might want to see what happens if you filter the genes based on differential expression. If you are going to evaluate the effect of the level of filtering I suggest you just try orders of magnitude, like retaining 10 vs 100 vs 1000 genes. The number of genes used is one aspect of "parsimony." Think carefully about how you will evaluate your classifier. There are many possibilities. For instance, you could obviously train it on the training dataset and test it on the test dataset, and vice-versa. But feel free to organize the data in other ways. These data were collected from three different European studies and, as we saw in class, microarray data collected on the same phenotypes but by different labs can be surprisingly different, and results obtained on one dataset often do not generalize to the other. One possibility is to learn a classifier and estimate its error rate with cross-validation on each of the two datasets and compare the results.

Finally, in order to allow us to compare your results and frame them in a more clinically realistic setting, determine the specificity on the test dataset that can be achieved while maintaining 80% senstivity, where the classifier is trained on the training dataset. As usual, sensitivity (respectively, specificity) is the fraction of correctly classified “Relapse” patients (respectively, “NoRelapse” patients). We want to maintain a high sensitivity at the expense of reduced specificity since witholding treatment from a patient who would benefit (a false negative) is more serious than treating a patient who would not (a false positive). So the steps for this last part are the following:

- Define a (real-valued) discriminant function (“score”) $g(x)$ for your approach, where large values of $g(x)$ are associated with $Y = 1$. Whatever approach you use this has likely already been done, perhaps implicitly. For example, if you chose a k for the kNN classifier (or the $kTSP$ classifier), the natural

discriminant function is simply the number of the k nearest neighbors with class 1 (number of pairs voting for class 1). If you built a random forest, the discriminant function could be the number of trees which vote for class 1. Virtually any method has a natural discriminant.

- Construct an ROC curve for your g using the test set. That is, apply the classifier $F_t(x) = \delta\{g(x) \geq t\}$ to the samples x of the test set, compute the sensitivity $sens(t)$ and specificity $spec(t)$ for each t, and plot $(1 - spec(t), sens(t))$.

- Find the point on the ROC curve where $sens(t) \geq .80$, i.e., the largest value of t for which

$$\hat{P}(g(X) \geq t | Y = 1) \geq .8,$$

where $\hat{P}$ refers to the estimate based on the test set. Call this point t_{80}. Your specificity at 80% sensitivity is then $spec(t_{80}) = \hat{P}(g(X) < t_{80} | Y = 0)$.

This is actually an optimistic estimate of $spec(t_{80})$ because in practice you would need to estimate the threshold t_{80} *based on the training data.* But this still gives us a single number for each approach. (Again, you are not being "graded" based on the specificity you report.) Just show the ROC curve and report $spec(t_{80})$.