Mapping Chang To GEO

Keith A. Baggerly and Kevin R. Coombes

November 13, 2007

1 Introduction

We want to match the Chang array data from GEO with the clinical information supplied in Chang et al and the supplementary material. We do this by trying to match the expression values for the 92 genes Chang et al identify as "important" for separating Sensitive and Resistant samples. Chang et al give the expression values for these 92 probesets in their supplementary information table. In that table, the values are mapped to (a) the sample identifiers N1-N24 and (b) associated clinical information such as the percent residual disease. The quantification files from GEO (which we loaded in for the Coombes et al analysis) also give expression values for these probesets, and link them to the array identifiers GSM4901-GSM4924.

2 Options and Libraries

```
> options(width = 80)
```

```
> load(file.path("RDataObjects", "changData.Rda"))
```

3 Establishing the Mapping

The Chang probeset ordering was matched to the Novartis order when the data was initially read into R (supplementary report SR6 from Coombes et al). In addition, the samples were arranged so that the 11 Responders (as reported by Chang et al) are in columns 1-11, and the 13 nonresponders are in columns 12-24.

The clinical information comes from Chang et al, both from Table 1 in the initial paper and from a supplementary table available from the Lancet website, `http://image.thelancet.com/extras/01art11086webtable.pdf`. Unfortunately, there are some difficulties in mapping the values. The numbers in the Lancet table do not exactly match the numbers that we obtain from GEO. Some change has occurred, possibly in the version of dChip used or the size of the sample set used to define the models. We can still identify the samples, but the approach will be based upon high correlation rather than perfect identity.

We have parsed the pdf file to generate a csv file for easier loading. We surveyed the structure of this file in rep01-checkingDoceClinical, so we do not cover the parsing in much detail here.

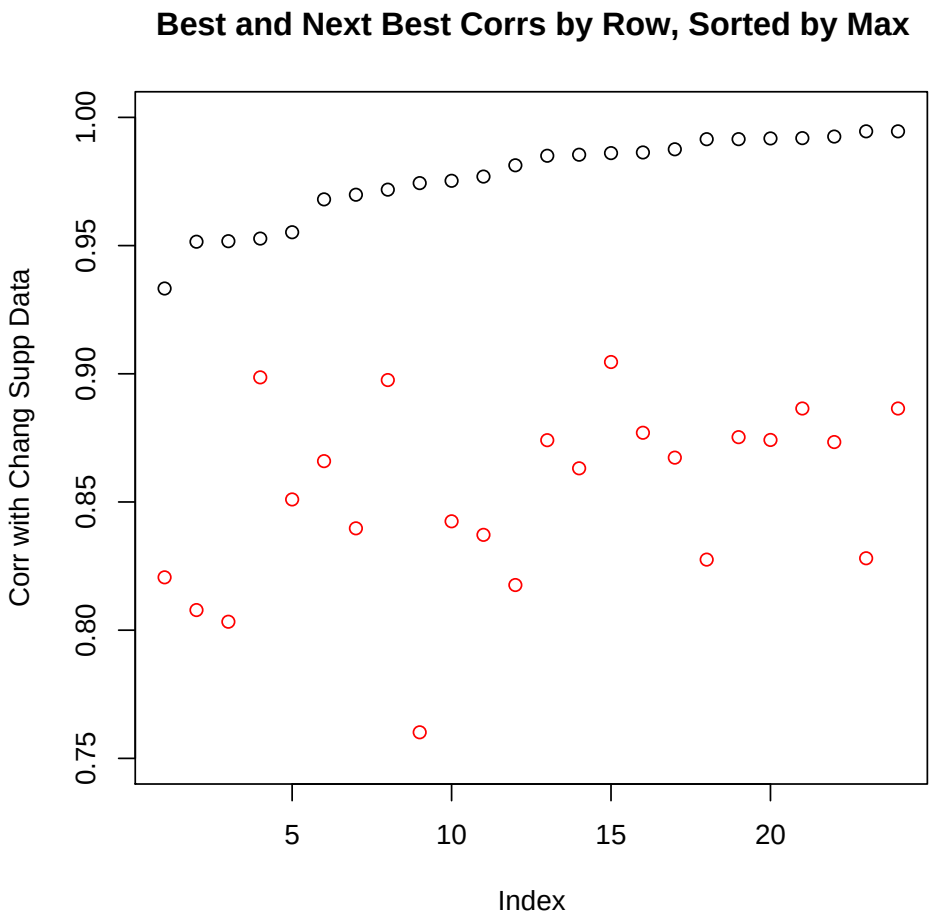
```
> changKeyTable <- read.table(file.path("OtherData", "01art11086webtable_rev.csv"),
+     skip = 4, header = FALSE, sep = ",")
> dim(changKeyTable)
```

```
[1] 92 29
```

```
> colnames(changKeyTable) <- c("ProbeSet", "GeneID", "LocusLink",
+     "GeneSymbol", "GeneDescription", paste("N", 1:24, sep = ""))
> changKeyTable[1:3, 1:5]

   ProbeSet  GeneID LocusLink GeneSymbol
1 1008_f_at U50648       5610      PRKR
2   1199_at D13748       1973    EIF4A1
3   1250_at U47077       5591     PRKDC
                                                    GeneDescription
1 protein kinase, interferon-inducible double stranded RNA dependent
2             eukaryotic translation initiation factor 4A, isoform 1
3               protein kinase, DNA-activated, catalytic polypeptide

> as.character(changKeyTable$ProbeSet[1:4])

[1] "1008_f_at" "1199_at"   "1250_at"   "1624_at"

> changKeyRows <- match(as.character(changKeyTable$ProbeSet), rownames(changSoft))
> changKeyRows[1:4]

[1] 6120 6328 6380 6808

> sampleCorrs <- cor(changKeyTable[, 6:29], changSoft[changKeyRows,
+     ], method = "spearman")
> sampleCorrs[1:4, 1:4]

     GSM4903   GSM4907   GSM4908   GSM4914
N1 0.7993157 0.7769215 0.7155341 0.3669532
N2 0.7553442 0.7381440 0.7494413 0.5128515
N3 0.8439377 0.8741152 0.9850461 0.5895507
N4 0.7427061 0.7508747 0.7539417 0.3579757
```

Looking at the subtable shown above, we see (for example) that N3 is very highly correlated with GSM4908. We can check to see how much the biggest correlations exceed the next biggest to see if there are "clear winners".

```
> rowMaxes <- apply(sampleCorrs, 1, max)
> rowSecondBiggest <- apply(sampleCorrs, 1, function(x) {
+     -sort(-x)[2]
+ })
> plot(sort(rowMaxes), ylim = c(0.75, 1), ylab = "Corr with Chang Supp Data",
+     main = "Best and Next Best Corrs by Row, Sorted by Max")
> points(rowSecondBiggest[order(rowMaxes)], col = "red")
```



There are clear winners in all 24 cases. After some experimentation, we find that 0.93 is the first 2 digit cutoff that passes 24 points in the matrix; 0.94 passes 23 and 0.98 passes 13. Let's look at where these high correlations are.

```
> sum(sampleCorrs > 0.93)
```

```
[1] 24
```

```
> sum(sampleCorrs > 0.94)
```

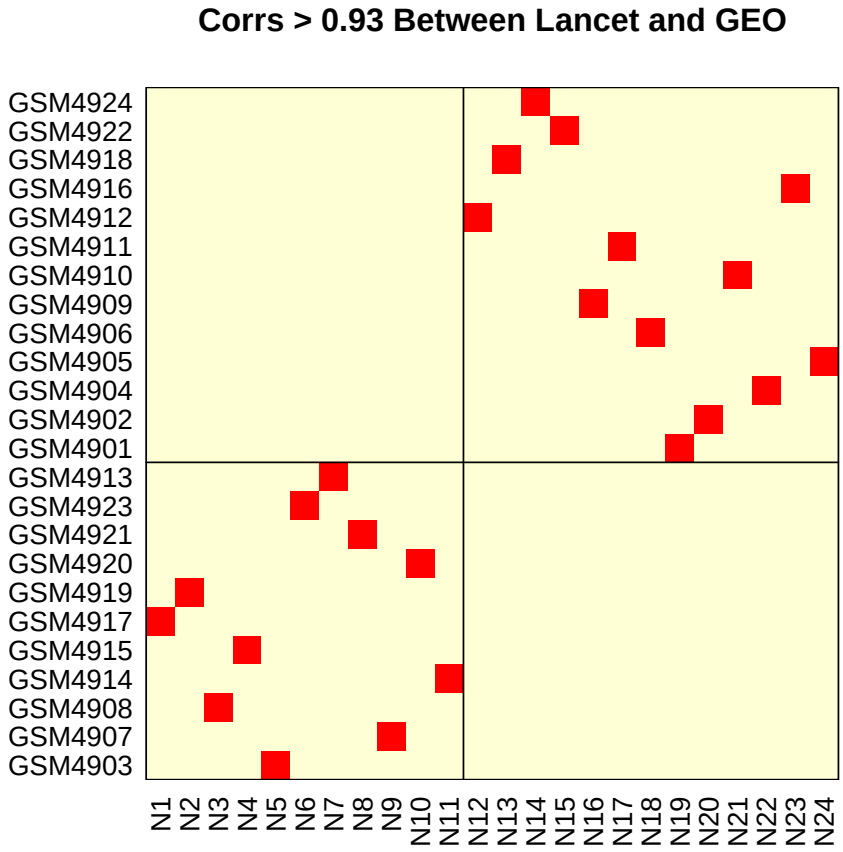
```
[1] 23
```

```
> sum(sampleCorrs > 0.98)
```

```
[1] 13
```

```
> image(1:24, 1:24, sampleCorrs < 0.93, axes = FALSE, xlab = "",
+     ylab = "", asp = 1, main = "Corrs > 0.93 Between Lancet and GEO")
```

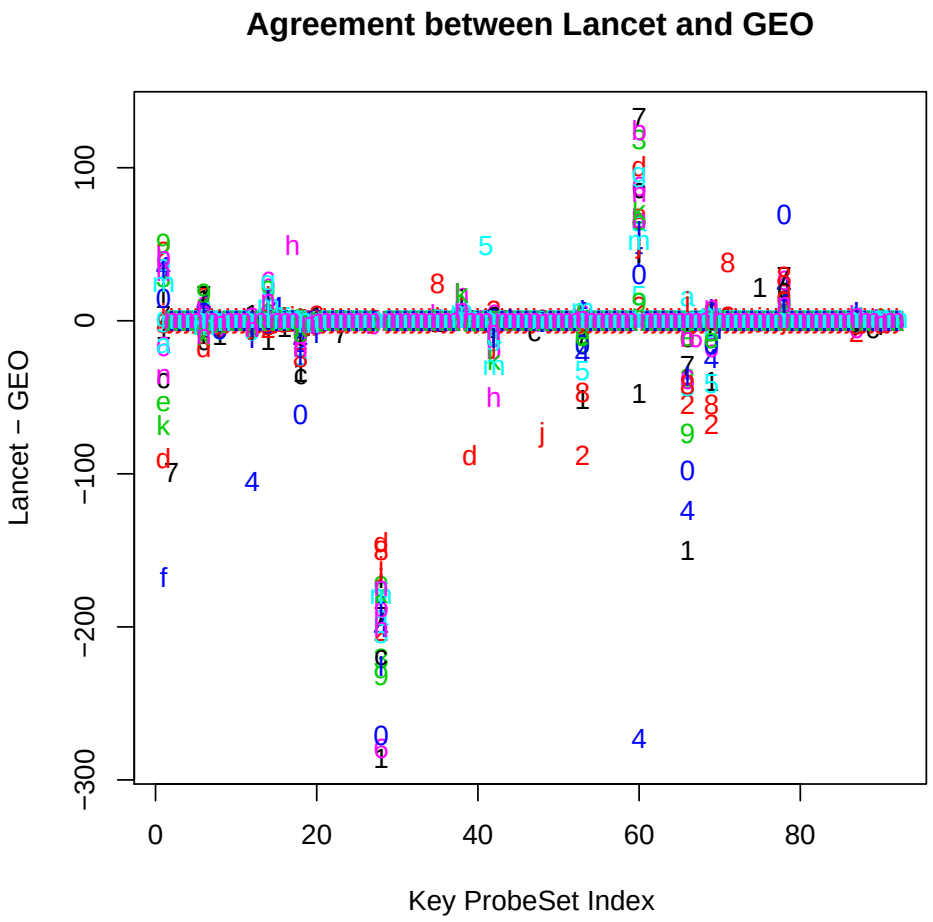
```
> lines(c(0.5, 24.5), c(11.5, 11.5))
> lines(c(11.5, 11.5), c(0.5, 24.5))
> rect(0.5, 0.5, 24.5, 24.5)
> axis(1, at = 1:24, labels = rownames(sampleCorrs), las = 2, line = -0.5,
+     tick = 0)
> axis(2, at = 1:24, labels = colnames(sampleCorrs), las = 2, line = -2.1,
+     tick = 0)
```



The 24 identified links do establish a one-to-one mapping between the names used in Chang et al and the names assigned at GEO. Somewhat reassuringly, the first 11 GEO samples (labeled sensitive) are indeed the 11 samples with the smallest amount of residual disease (lines indicate the sensitive/resistant border).

Of course, there's still a question of why the correlations aren't higher than they are. We can first check the absolute agreement.

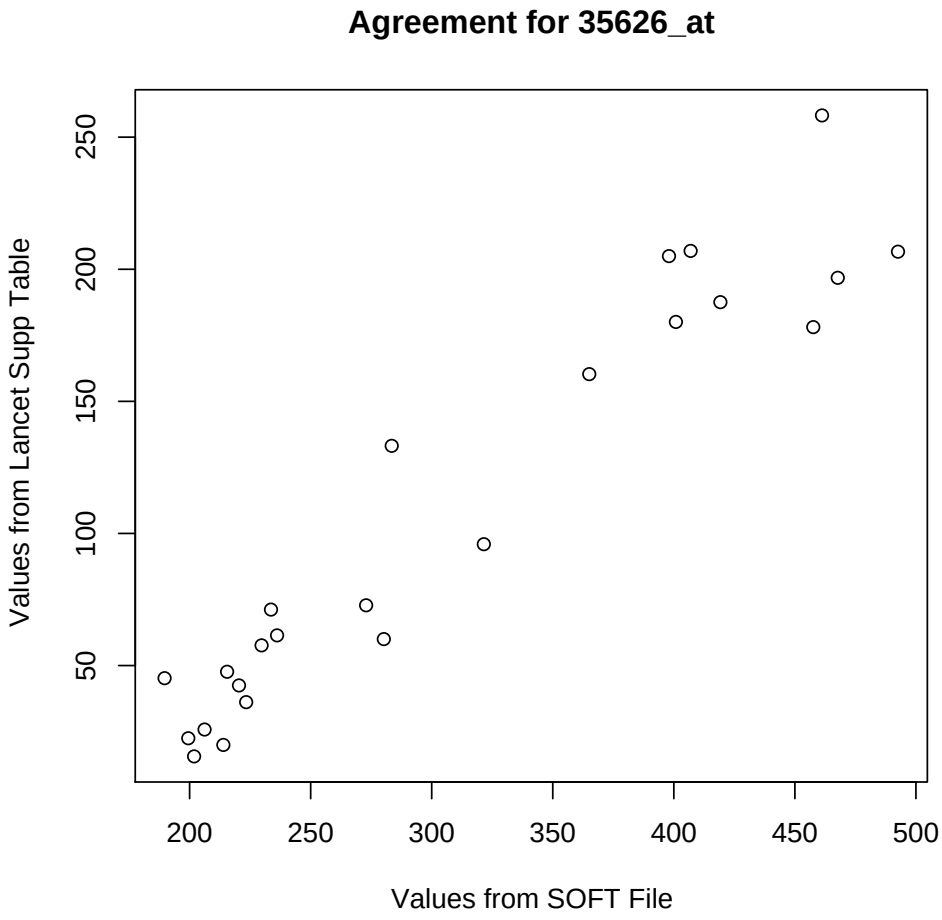
```
> tumorSize <- which(t(sampleCorrs > 0.93), arr.ind = TRUE)
> matplot(changKeyTable[, 6:29] - changSoft[changKeyRows, rownames(tumorSize)],
+     xlab = "Key ProbeSet Index", ylab = "Lancet - GEO", main = "Agreement between Lancet and GEO")
```



While the agreement is good for most of the probesets reported, there are a small number which are consistently off. Of course, some of the discrepancy may be due to slightly different normalizations. It may be more pertinent to see what the correlations are like, gene by gene, assuming that the ordering used here is correct.

The most problematic gene (giving the biggest absolute differences) occurs in row 28 of the above table: 35626_at. Let's take a look at this one.

```
> softGuess <- changSoft[changKeyRows, rownames(tumorSize)]
> plot(t(softGuess[28, ]), t(changKeyTable[28, 6:29]), xlab = "Values from SOFT File",
+     ylab = "Values from Lancet Supp Table", main = "Agreement for 35626_at")
```



Here, we can see that the correlation is actually quite high; for some reason the data simply has a different center and scale.

There is another way to check the stability of this ordering. Specifically, for every pair of samples, we have 92 "votes" as to how they should be ordered.

```
> matchDirections <- matrix(0, 24, 24)
> for (i1 in 1:24) {
+     for (i2 in 1:24) {
+         matchDirections[i1, i2] <- sum(sign(softGuess[, i1] -
+             softGuess[, i2]) == sign(changKeyTable[, i1 + 5] -
+             changKeyTable[, i2 + 5]))
+     }
+ }
> table(as.vector(matchDirections))/2

 84  85  86  87  88  89  90  91  92
  1   1   3  13  16  21  25  67 141
```

```
> which(matchDirections == 84, arr.ind = TRUE)

     row col
[1,]  17  16
[2,]  16  17
```

This tells us that for every pair of samples, there are at least 84 of the 92 possible votes in agreement with the ordering we have found.

At the end of the day, the mapping is

```
> changGEOMapping <- colnames(softGuess)
> names(changGEOMapping) <- colnames(changKeyTable)[6:29]
> changGEOMapping

       N1        N2        N3        N4        N5        N6        N7        N8
"GSM4917" "GSM4919" "GSM4908" "GSM4915" "GSM4903" "GSM4923" "GSM4913" "GSM4921"
       N9       N10       N11       N12       N13       N14       N15       N16
"GSM4907" "GSM4920" "GSM4914" "GSM4912" "GSM4918" "GSM4924" "GSM4922" "GSM4909"
      N17       N18       N19       N20       N21       N22       N23       N24
"GSM4911" "GSM4906" "GSM4901" "GSM4902" "GSM4910" "GSM4904" "GSM4916" "GSM4905"
```

4 Appendix

4.1 Saves

```
> save(changGEOMapping, file = file.path("RDataObjects", "changGEOMapping.Rda"))
```

4.2 SessionInfo

```
> sessionInfo()

R version 2.5.1 (2007-06-27)
i386-pc-mingw32

locale:
LC_COLLATE=English_United States.1252;LC_CTYPE=English_United States.1252;LC_MONETARY=English_United Sta

attached base packages:
[1] "stats"     "graphics"  "grDevices" "utils"     "datasets"  "methods"
[7] "base"

other attached packages:
R.matlab     R.oo
 "1.1.3"  "1.3.0"
```