Identifying Adria Rows: Chip Comparer

Keith A. Baggerly and Kevin R. Coombes

November 13, 2007

1 Introduction

In point 6 of our correspondence, we note an apparent reversal of the sensitive and resistant labels applied to adriamycin. In their response, Potti and Nevins address this, commenting on "the acute lymphocytic leukemia dataset in which the labels are accurate — full details are provided on our web page." (`http://data.cgt.duke.edu/NatureMedicine.php`.)

Of the 16 files available as of Nov 8, 2007, only one, "Adria_ALL(n = 122).txt" involves the adriamycin samples. This file contains processed data for 144 samples, 22 training samples and 122 validation samples. A note to the side (in position EP3 when the table is read into Excel) states that "Validation data is from GSE4698, GSE649, GSE650, GSE651, and others."

As with the table of processed data for Docetaxel, we'll try to mimic the processing steps involved for Adriamycin. we may be able to understand the approach better if we can reproduce their processed data. Thus, we'll load our numbers, then theirs, and try to align them.

2 Options and Libraries

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

```
> load(file.path("RDataObjects", "chemoPredictors.Rda"))
> adriaData <- predictors[, predictorsInfo$drugName == "Adria"]
> adriaInfo <- predictorsInfo[predictorsInfo$drugName == "Adria",
+     ]
> rm("predictors", "predictorsInfo", "predictorsSubset")
```

3 Loading the Duke Quantifications

The table of Duke data was initially in tab-delimited form, with an added comment in column EP, row 3 stating that "Validation data is from GSE4698, GSE649, GSE650, GSE651 and others." We trimmed off this comment to get a more consistently formatted table for easier loading we will return to it at the end). There are two header lines indicating (row 1) whether the column is training (Adria) or testing (Validation), and (row 2) whether the column is Sens or Resistant (training) or Resp or NR (testing).

```
> dukeHeader1 <- read.table(file.path("DukeWebSite", "Adria_ALL(n = 122).txt"),
+     sep = "\t", nrows = 1, header = FALSE)
> dukeHeader1 <- as.vector(t(dukeHeader1))
> dukeHeader2 <- read.table(file.path("DukeWebSite", "Adria_ALL(n = 122).txt"),
```

```
+     sep = "\t", skip = 1, nrows = 1, header = FALSE)
> dukeHeader2 <- as.vector(t(dukeHeader2))
> dukeAdria <- read.table(file.path("DukeWebSite", "Adria_ALL(n = 122).txt"),
+     sep = "\t", skip = 2, header = FALSE)
> table(dukeHeader1)

dukeHeader1
          0           1           2      Adria0      Adria1 Validation2
          9          11         120           1           1           2

> table(dukeHeader2)

dukeHeader2
       NR Resistant      Resp      Sens
       99        10        23        12

> dim(dukeAdria)

[1] 8958  144

> dukeAdria[1:3, 1:10]

    V1   V2   V3   V4   V5   V6   V7   V8   V9  V10
1 1.18 1.12 3.46 0.65 3.07 1.57 0.13 1.05 2.38 1.53
2 1.75 4.02 0.43 0.31 0.76 0.37 0.21 0.69 0.15 1.65
3 0.13 0.35 1.13 1.14 0.84 0.27 0.63 0.89 2.40 2.33

> range(dukeAdria)

[1]    0 2110

> range(dukeAdria[, 1:22])

[1]  0.02 72.30
```

As with the data for Docetaxel, the data have been processed (standardized) in what appears to be a similar fashion. There is, however, an additional difficulty. The data matrix has 8958 rows and 144 columns (22 training, 122 testing). This is a much smaller number of rows than we've worked with before. This comes about because the Adria testing data was run on U133Av2 arrays, and the training data on U95Av2 arrays, necessitating a mapping across platforms. As noted in Potti et al, this mapping was accomplished using "Chip Comparer", available at `http://tenero.duhs.duke.edu/genearray/perl/chip/chipcomparer.pl`. We ran chipComparer to get mappings going each way (the row orderings are different depending on which platform is named first), and found that there were 8958 distinct Locus Link clusters that the data were mapped to.

4 Identifying the Probesets

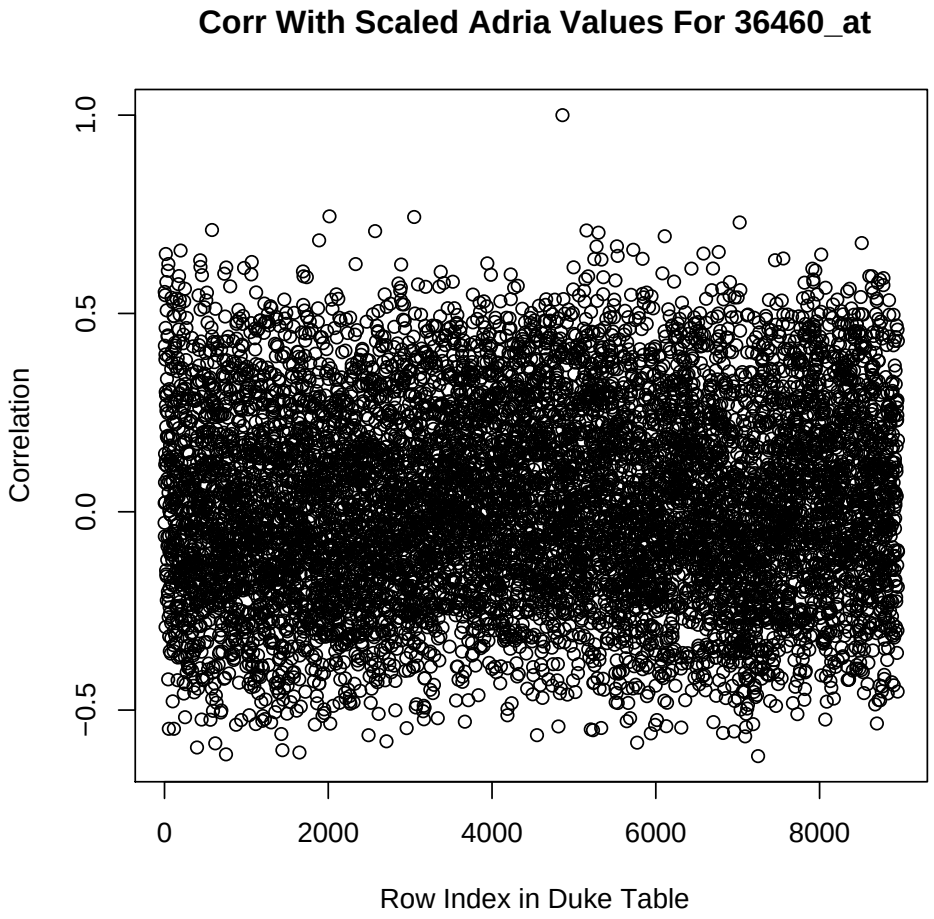
We're not sure what probesets the rows correspond to, but in some cases (when just one probeset maps to a given Locus Link id) they should correspond to individual Affy U95Av2 entries. While we don't know for certain what the samples are either, we're going to make an educated guess (based on experience with docetaxel, described in rep04) that the columns are the Adria columns from the chemo predictors, in the same order. If so, we should be able to establish some mappings.

```
> adriaScaled <- round(exp(t(scale(t(log(adriaData))))), 2)
> rownames(adriaScaled) <- rownames(adriaData)
> temp <- cor(adriaScaled[1, ], t(dukeAdria[, 1:22]))
> plot(t(temp), xlab = "Row Index in Duke Table", ylab = "Correlation",
+     main = paste("Corr With Scaled Adria Values For", rownames(adriaScaled)[1]))
> max(temp)

[1] 1

> which.max(temp)

[1] 4860
```



Here, there is a perfect match for the first probeset. This suggests that at least part of the problem is amenable to brute force. We can start with the rows in the Duke table and try to identify the best correlated rows from the chemo predictors used earlier.

```
> temp <- apply(dukeAdria[, 1:22], 1, function(x) {
+     max(cor(x, t(adriaScaled)))
+ })
> min(temp)
```

```
[1] 0.9999924
```

Actually, this worked better than we expected. The correlations nearly all perfect. This suggests that they use chip comparer by simply "picking one" of the probeset ids mapping to a given Locus Link. Given this approach, we can identify all of the probesets (and hence Locus Link values) involved.

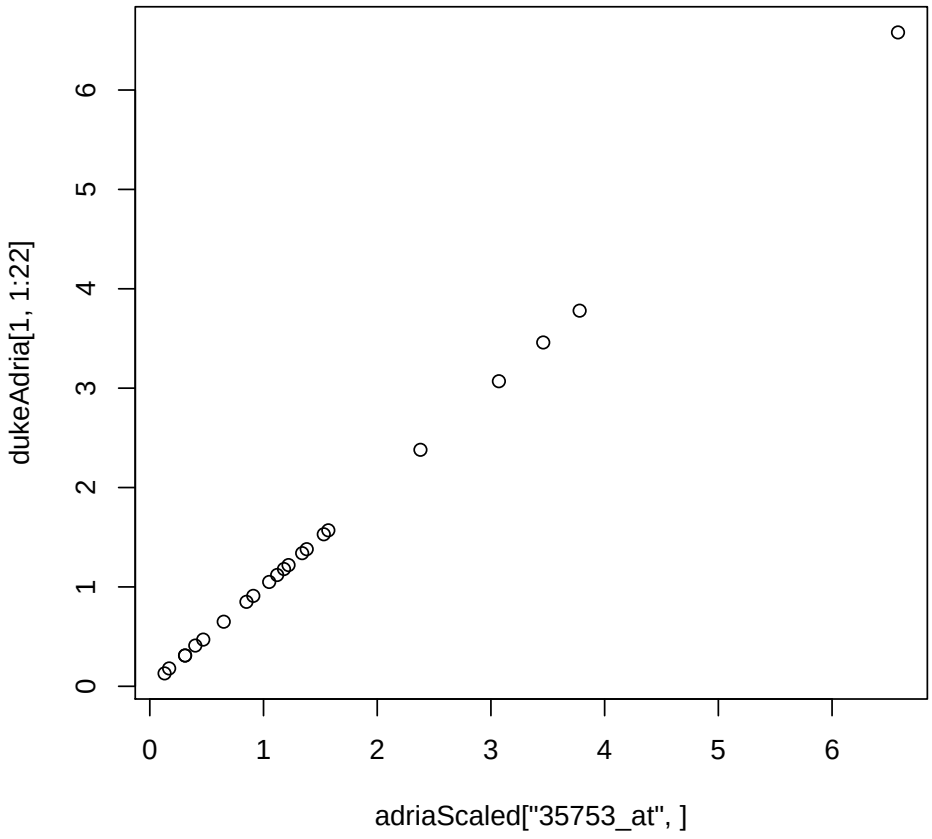
```
> dukeRowIds <- apply(dukeAdria[, 1:22], 1, function(x) {
+     which.max(cor(x, t(adriaScaled)))
+ })
> dukeProbesets <- rownames(adriaScaled)[dukeRowIds]
```

Let's double-check just to be sure.

```
> dukeProbesets[1:4]
```

```
[1] "35753_at" "36138_at" "41765_at" "35298_at"
```

```
> plot(adriaScaled["35753_at", ], dukeAdria[1, 1:22])
```



This works. Let's attach this to the data so we don't forget.

```
> rownames(dukeAdria) <- dukeProbesets
```

5 Conclusions and Observations

At this point, we understand part of chip comparer, and we have identified the U95Av2 probesets used. Unfortunately, we do not know what the corresponding U133Av2 probesets are because we don't know where the validation data came from. With that in mind, we now return to the side comment present in the Adria file.

"Validation data is from GSE4698, GSE649, GSE650, GSE651, and others."

Potti et al's initial Nature Medicine paper named only GSE650 and GSE651, as we noted in our correspondence. These two datasets involve 94 and 28 samples labeled respectively as sensitive and resistant to daunorubicin (adriamycin = doxorubicin), adding up to 122 samples in all. GSE649 gives measurements for 36 samples labeled as resistant to vincristine, not adriamycin. These three GEO datasets are all linked to the paper by Holleman et al (NEJM, 2004, 351:533-542). GSE4698, however, involves 60 samples from

children with relapsed ALL, described in Kirschner-Schwabe et al (Clin Canc Res 2006, 12:4553-4561); this last paper was not cited by Potti et al.

The columns of validation data are not identified by name. We are only told (a) that they are validation samples, and (b) whether they are Resp or NR. Thus, we have no idea what specific samples are involved, which full datasets they come from (assuming that they do not come from the the catchall "and others"), or what their associated clinical information is. We confess that we find these "full details" inadequate.

6 Appendix

6.1 Saves

```
> save(dukeAdria, dukeHeader1, dukeHeader2, file = file.path("RDataObjects",
+     "dukeAdria.Rda"))
```

6.2 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"
```