Computing Response to Combination Therapy

Keith A. Baggerly

May 20, 2008

1 Introduction

Bonnefoi et al. [1] extend the gene signature approach introduced by Potti et al. [3] to predict which of two combination regimens (FEC or TET) women with ER- breast cancer were more likely to respond to. Here, we are trying to clarify how results for individual drugs were combined to produce results for the ensemble. Such combination was performed earlier by Potti et al. [3].

According to the statistical methods section of Potti et al. [3],

In instances where a combined probability of sensitivity to a combination chemotherapeutic regimen was required based on the individual drug sensitivity patterns, we used the probabilities of response to individual drugs and used the theorem for combined probabilities as described by William Feller to deduce a probability of response to a combination of the drugs being studied. The result was then mean-centered to give a probability between 0 and 1.

We need to clarify this. The supplementary website for Potti et al (http://data.cgt.duke.edu/NatureMedicine.php) supplies the file “MDACC data.zip”. This contains two files:

- MDA51ValidationExpression.xls, and
- MDA51ValidationClinicalData and predictions.xls.

Since this data from MD Anderson involved profiles and clinical information for 51 breast cancer patients treated with a combination chemotherapy (TFAC = taxol, 5-fluorouracil, adriamycin, and cyclophosphamide), examining this collection of files more closely may clarify the approach.

In this report, we attempt to identify the combination rule used, check to confirm that we can generate the corresponding plots from Potti et al. [3], and also check whether we can recreate values supplied for the components.

2 Options and Libraries

> options(width = 80)

3 Checking the TFAC Combo Predictions: Defining the Rules Used

We begin by looking closely at “MDA51ValidationClinicalData and predictions.xls”. Most of the this file involves clinical information assembled at MDA by Keith Anderson, but the prediction values that we want
are also there. Specifically, there are 22 relevant columns at the right end of the file, each with one number per sample. There is a block of four columns for each of the four drugs involved (16 columns total). After some experimentation, we believe these give

1. The assessed probability of resistance to the drug
2. The lower bound of a credible interval for the probability mentioned above
3. The upper bound of the credible interval
4. The sample “score”, on something like a probit scale.

After these 16 columns, they then give 4 more, reporting the probability of sensitivity to each drug, computed as 1 minus the probability of resistance listed above. We’ll use the shorthand \( P() \) for the probability of sensitivity, so \( P(T) \) indicates the probability that a cell line or patient is sensitive to taxol.

Then they give a column for the “Combined probability (T or F or A or C)”, where the title chosen suggests that they are computing the probability of a union. Checking the formula for cell BB7 shows that this is computed as

\[
P(T) + P(F) + P(A) + P(C) - P(T) \times P(F) \times P(A) \times P(C).
\]

The theorem from Feller that we suspect gave rise to this is

\[
P(A \cup B) = P(A) + P(B) - P(AB),
\]

where \( P(AB) \) is meant to indicate the probability of the intersection of \( A \) and \( B \), not necessarily the product \( P(A) \times P(B) \). If the responses to drugs \( A \) and \( B \) are independent, then \( P(AB) = P(A) \times P(B) \), but the result does not hold in general. We’re willing to work with the independence assumption here; it may not be great, but it’s a reasonable starting point. Now, the above theorem from Feller deals with two events. The proper extension (from set theory) from 2 to 4 would be

\[
P(D_1 \cup D_2 \cup D_3 \cup D_4) = P(D_1) + P(D_2) + P(D_3) + P(D_4) - P(D_1D_2) - P(D_1D_3) - P(D_1D_4) - P(D_2D_3) - P(D_2D_4) - P(D_3D_4) + P(D_1D_2D_3) + P(D_1D_2D_4) + P(D_1D_3D_4) + P(D_2D_3D_4) - P(D_1D_2D_3D_4)
\]

with the compressed notation again indicating intersection, not product. If we assume that the actions of all 4 drugs are independent, this does simplify to

\[
1 - (1 - P(D_1)) \times (1 - P(D_2)) \times (1 - P(D_3)) \times (1 - P(D_4))
\]

which is still quite different from what they have.

Here, the problem with the formulation used manifests in that all of the “combined probabilities” thus produced are greater than 1. This leads to the last of the columns we consider, titled “Centered”. If we check the formula for the first entry in this column (cell BC7), we find

\[
(BB7 - \text{MIN}(BB$7:BB$57))/\text{MAX}(BB$7:BB$57) - \text{MIN}(BB$7:BB$57))
\]

which gives the “mean-centering” they referred to in their initial description. They took the set of values produced, set the largest observed value to 1 (sensitive), the smallest observed value to 0 (resistant), and linearly interpolated in between. This is a decidedly nonstandard adjustment, particularly in that it forces the probabilities of sensitivity for some samples to 0 or 1.

To examine these numbers in more detail, we extracted the relevant columns from the full Excel file and saved them in csv format for easier loading. One extra column was left in to allow the formula to refer to the correct offset location; we excise this column after loading the data into R.
> tfac <- read.table(file.path("OtherData", "tfac_combo.csv"),
+   sep = ",", header = TRUE)
> dim(tfac)

[1] 51 25

> tfac[1, ]

idtext pCR TAXOL.resistance taxol2 taxol3 taxol4 X5.FU.resistance
1 M107 0 0.6509626 0.4155768 0.8720181 0.415958 0.4144488
   fu2 fu3 fu4 ADRIA.resistance adria2 adria3 adria4
1 0.3323356 0.4741328 -0.2331979 0.5736701 0.4356633 0.7321636 0.1199092
   CYTOXAN.resistance cyt2 cyt3 cyt4 Taxol.sensitivity
1 0.6884801 0.5333264 0.8897092 0.4724243 0.3490374
   X5.Fu.sensitivity Adria.sensitivity Cytoxan.sensitivity X
1 0.5855512 0.4263299 0.3115199 NA
   Combined.probability..T.or.F.or.A.or.C. Centered
1 1.645295 0.2755124

> tfac[1:3, 23]

[1] NA NA NA

> tfac <- tfac[, c(1:22, 24:25)]
> colnames(tfac) <- c("SampleID", "pCR", "TaxolProbResistant",
+   "5FUProbLB", "5FUProbUB", "5FUProbit", "AdriaProbResistant",
+   "5FUProbSensitive", "AdriaProbSensitive", "CytoxProbSensitive",
+   "ComboProbSensitive", "ComboProbCentered")
> tfac$pCR

[1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[39] 1 1 1 1 1 1 1 1 1 1 1 1 1

We note that the clinical information has been sorted so that the data for the 13 responders (pCR = 1) is in the last 13 rows. There is a second page to the initial Excel file that gives the interpretations for 0 and 1; it confirms that a pCR value of 1 indicates a responder.

4 Checking Figure 3A From Potti et al. [3]

Given the probabilities loaded above, we want to check that we can get the results shown in Figure 3A of Potti et al. [2], which shows their predictions for TFAC therapy. First, we check the results for the individual drugs.

> plot(c(0.5, 8.5), c(0, 1), type = "n", axes = FALSE, xlab = "Individual Drugs",
+   ylab = "Probability of Sensitivity", ylim = c(0, 1))
> points(jitter(rep(1, sum(tfac$pCR)), amount = 0.25), tfac$TaxolProbSensitive[tfac$pCR ==
```R
> points(jitter(rep(2, sum(tfac$pCR)), amount = 0.25), tfac$"5FUProbSensitive"[tfac$pCR == + 1], col = "blue")
> points(jitter(rep(3, sum(tfac$pCR)), amount = 0.25), tfac$AdriaProbSensitive[tfac$pCR == + 1], col = "red")
> points(jitter(rep(4, sum(tfac$pCR)), amount = 0.25), tfac$CytoxProbSensitive[tfac$pCR == + 1], col = "green")
> points(jitter(rep(5, sum(1 - tfac$pCR)), amount = 0.25), tfac$TaxolProbSensitive[tfac$pCR == + 0], col = "blue")
> points(jitter(rep(6, sum(1 - tfac$pCR)), amount = 0.25), tfac$"5FUProbSensitive"[tfac$pCR == + 0], col = "red")
> points(jitter(rep(7, sum(1 - tfac$pCR)), amount = 0.25), tfac$AdriaProbSensitive[tfac$pCR == + 0], col = "black")
> points(jitter(rep(8, sum(1 - tfac$pCR)), amount = 0.25), tfac$CytoxProbSensitive[tfac$pCR == + 0], col = "green")
> meanResp <- apply(tfac[tfac$pCR == 1, c("TaxolProbSensitive", + "5FUProbSensitive", "AdriaProbSensitive", "CytoxProbSensitive")], + 2, mean)
> meanNonResp <- apply(tfac[tfac$pCR == 0, c("TaxolProbSensitive", + "5FUProbSensitive", "AdriaProbSensitive", "CytoxProbSensitive")], + 2, mean)
> segments(c(1:8) - 0.4, c(meanResp, meanNonResp), c(1:8) + 0.4, + c(meanResp, meanNonResp), col = "black", lwd = 1)
> abline(v = 4.5, col = "black")
> axis(side = 1, at = c(1:8), labels = c("T", "F", "A", "C", "T", + "F", "A", "C")
> axis(side = 2, at = c(0, 0.25, 0.5, 0.75, 1), labels = TRUE)
> axis(side = 3, at = c(2.5, 6.5), labels = c("Responders", "Nonresponders"), + cex = 1.5, lty = "blank")
> box(bty = "l")
```
This figure is qualitatively identical to Figure 3A from Potti et al. [3] for individual drugs, essentially differing only in the manner of “jittering” the x-values to make the points appear distinct.

Next, we check the results for the combination therapy.

> plot(c(0.5, 2.5), c(0, 1), type = "n", axes = FALSE, xlab = "Combination Therapy", +       ylab = "Probability of Combo Sensitivity", ylim = c(0, 1))
> points(jitter(rep(1, sum(tfac$pCR)), amount = 0.25), tfac$ComboProbCentered[tfac$pCR == +       1], col = "blue")
> points(jitter(rep(2, sum(1 - tfac$pCR)), amount = 0.25), tfac$ComboProbCentered[tfac$pCR == +       0], col = "red")
> meanResp <- mean(tfac[tfac$pCR == 1, "ComboProbCentered"])
> meanNonResp <- mean(tfac[tfac$pCR == 0, "ComboProbCentered"])
> segments(c(1:2) - 0.4, c(meanResp, meanNonResp), c(1:2) + 0.4, +       c(meanResp, meanNonResp), col = "black", lwd = 1)
> axis(side = 1, at = c(1, 2), labels = c("T+F+A+C", "T+F+A+C"))
> axis(side = 2, at = c(0, 0.25, 0.5, 0.75, 1), labels = TRUE)
Again, this plot is qualitatively identical to Figure 3A from Potti et al. for combination therapy, essentially differing only in the manner of “jittering” the x-values to make the points appear distinct. These values are the ones used to produce the figure.

We’d also like to try one more variant: looking at the combination results when we use the correct method for calculating the probability of the union of 4 independent events. First we calculate the union probabilities.

```r
> rightCombo <- rep(0, 51)
> for (i1 in 1:51) {
+   rightCombo[i1] <- 1 - prod(1 - tfac[i1, c("TaxolProbSensitive",
+                               "5FUProbSensitive", "AdriaProbSensitive",
+                               "CytoxProbSensitive")])
+ }
```
Now we revisit the combination plot.

```r
> plot(c(0.5, 2.5), c(0, 1), type = "n", axes = FALSE, xlab = "Combination Therapy", + ylab = "Probability of Combo Sensitivity", ylim = c(0, 1))
> points(jitter(rep(1, sum(tfac$pCR)), amount = 0.25), rightCombo[tfac$pCR == + 1], col = "blue")
> points(jitter(rep(2, sum(1 - tfac$pCR)), amount = 0.25), rightCombo[tfac$pCR == + 0], col = "red")
> meanResp <- mean(rightCombo[tfac$pCR == 1])
> meanNonResp <- mean(rightCombo[tfac$pCR == 0])
> segments(c(1:2) - 0.4, c(meanResp, meanNonResp), c(1:2) + 0.4, + c(meanResp, meanNonResp), col = "black", lwd = 1)
> axis(side = 1, at = c(1, 2), labels = c("T+F+A+C", "T+F+A+C"))
> axis(side = 2, at = c(0, 0.25, 0.5, 0.75, 1), labels = TRUE)
> axis(side = 3, at = c(1, 2), labels = c("Responders", "Nonresponders"), + cex = 1.5, lty = "blank")
> box(bty = "l")
```
This shows one of the limitations of the simple union — if we apply enough drugs, then almost all of the samples will eventually be “sensitive”. Since this does not match what is observed clinically, the “centering” step represents an ad hoc adjustment to bring predictions more into line with observations.

Even here, though, there is a significant difference between the responder and nonresponder probabilities. If the probabilities reported for the individual drugs are correct, they may have something. So, let’s check the probabilities for one of individual drugs where a difference should be visible.

5 Checking Taxol

Of the four drugs in the TFAC combo, taxol (T) shows the biggest division between responders and nonresponders at the individual drug level, so we’ll check the results for that.

There is a difficulty here in that the cell lines used for training were run on U95Av2 arrays, and the MDA samples were run on U133Av2s. Potti et al. [3] combine results across platforms using “chip comparer”, but we’re not completely sure how this was used. Specifically, chip comparer matches probesets from two different platforms to a common set of Locus Link ids. One problem is that the rule for how to deal with
many-to-one or many-to-many mappings is not specified. Another is that the processing steps used before combining the numbers to feed them to the “binreg” software are not specified. Here, then, we shall pursue a more qualitative strategy.

We have a list of genes identified as “important” for taxol based on the cell line data. For each one of these, we can clearly identify whether it is higher in the sensitive or the resistant cell lines. To first order, this directionality captures the main information present in the results after standardizing. Then, for each of these U95Av2 probesets, we can check the corresponding probeset entries for the MDACC data, using all possible chip comparer values. What we’re seeking is whether the responder/nonresponder separation is in the same direction for the named gene.

Assuming that there are matching U133 probesets, we extract and rank the intensities. If the gene is higher in sensitive cell lines, we rank the intensities directly. Since larger ranks go to larger values, a large rank means that we would predict that sample to be sensitive. If the gene is higher in resistant cell lines, we rank the intensities after multiplying them by -1; this leaves the interpretation given above intact. If there are no matching U133Av2 probesets, there is no contribution from the initial U95Av2 probeset. If there are many matching U133Av2 probesets, the ranks for all of them are averaged to give the “sample ranks” associated with that U95Av2 probeset. Similar approaches are used to combine results for multiple U95Av2 probesets mapping to the same locus, assuming that they point the same direction. The driving rule here is: one locus, one set of rank scores. Once we have computed these values for all of the U95Av2 probesets, we sum over probesets to give a single score for each sample. Again, a higher score means we view the sample as more sensitive.

We begin by loading the MAS5.0 expression values for the MDACC data. We have saved the Excel data provided in csv format for easier loading.

```r
> tfacExpression <- read.table(file.path("OtherData", "MDA51ValidationExpression.csv"),
+   sep = ",", header = TRUE, row.names = 1)
> dim(tfacExpression)
[1] 22215   51
> tfacExpression[1:3, 1:4]

M107   M120   M129   M141
200000_s_at  537.878  491.552  342.257  739.596
200001_at   1302.000  725.140 1078.780  763.945
200002_at   4326.570  3259.680  4381.190  4101.370

> match(as.character(tfac$SampleID), colnames(tfacExpression))
[1]  1  2  3  5  7 10 11 13 14 15 16 18 19 23 24 25 26 27 29 30 32 33 34 35
[26] 36 38 39 40 41 43 44 46 47 48 49 50 51  4  6  8 12 17 20 21 22 28 31 37 42
[51] 45
```

In passing, we reordered the columns to match the clinical information. The clinical information table puts the 13 responders last, so the last 13 columns of expression values now correspond to the responders.

Next, we load the initial data on the chemopredictors. This uses the Rda file constructed as part of the Coombes et al. [2] analysis of the Potti et al. [3] data; see [http://bioinformatics.mdanderson.org/Supplements/ReproRsch-Chemo](http://bioinformatics.mdanderson.org/Supplements/ReproRsch-Chemo).
> load(file.path("RDataObjects", "chemoPredictors.Rda"))

Let’s focus on just the Taxol data.

> taxol <- predictors[, predictorsInfo$drugName == "Taxol"]
> taxolMat <- as.matrix(taxol)
> rownames(taxolMat) <- rownames(taxol)
> taxolInfo <- predictorsInfo[predictorsInfo$drugName == "Taxol",
 + ]
> taxolInfo

<table>
<thead>
<tr>
<th>index</th>
<th>drugName</th>
<th>responseStatus</th>
<th>Source</th>
<th>NovartisName</th>
</tr>
</thead>
<tbody>
<tr>
<td>91</td>
<td>Taxol</td>
<td>Resistant</td>
<td>SF-295</td>
<td>A.SF-295</td>
</tr>
<tr>
<td>92</td>
<td>Taxol</td>
<td>Resistant</td>
<td>SF-539</td>
<td>A.SF-539</td>
</tr>
<tr>
<td>93</td>
<td>Taxol</td>
<td>Resistant</td>
<td>HS 578T</td>
<td>A.HS 578T</td>
</tr>
<tr>
<td>94</td>
<td>Taxol</td>
<td>Resistant</td>
<td>MDA-MB-435</td>
<td>A.MDA-MB-435</td>
</tr>
<tr>
<td>95</td>
<td>Taxol</td>
<td>Resistant</td>
<td>COLO 205</td>
<td>A.COLO 205</td>
</tr>
<tr>
<td>96</td>
<td>Taxol</td>
<td>Resistant</td>
<td>HCC-2998</td>
<td>A.HCC-2998</td>
</tr>
<tr>
<td>97</td>
<td>Taxol</td>
<td>Resistant</td>
<td>HT29</td>
<td>A.HT29</td>
</tr>
<tr>
<td>98</td>
<td>Taxol</td>
<td>Resistant</td>
<td>OVCAR-3</td>
<td>A.OVCAR-3</td>
</tr>
<tr>
<td>99</td>
<td>Taxol</td>
<td>Resistant</td>
<td>DU-145</td>
<td>A.DU-145</td>
</tr>
<tr>
<td>100</td>
<td>Taxol</td>
<td>Sensitive</td>
<td>CCRF-CEM</td>
<td>A.CCRF-CEM</td>
</tr>
<tr>
<td>101</td>
<td>Taxol</td>
<td>Sensitive</td>
<td>SW-620</td>
<td>A.SW-620</td>
</tr>
<tr>
<td>102</td>
<td>Taxol</td>
<td>Sensitive</td>
<td>A549/ATCC</td>
<td>A.A549/ATCC</td>
</tr>
<tr>
<td>103</td>
<td>Taxol</td>
<td>Sensitive</td>
<td>EKVX</td>
<td>A.EKVX</td>
</tr>
<tr>
<td>104</td>
<td>Taxol</td>
<td>Sensitive</td>
<td>MALME-3M</td>
<td>A.MALME-3M</td>
</tr>
<tr>
<td>105</td>
<td>Taxol</td>
<td>Sensitive</td>
<td>SK-MEL-28</td>
<td>A.SK-MEL-28</td>
</tr>
<tr>
<td>106</td>
<td>Taxol</td>
<td>Sensitive</td>
<td>OVCAR-8</td>
<td>A.OVCAR-8</td>
</tr>
<tr>
<td>107</td>
<td>Taxol</td>
<td>Sensitive</td>
<td>786-0</td>
<td>A.786-0</td>
</tr>
</tbody>
</table>

We note in passing that the list of cell lines given above differs slightly from that given at [http://data.genome.duke.edu/NatureMedicine.php](http://data.genome.duke.edu/NatureMedicine.php) in “Cell lines in each chemo predictor.xls”. We have DU-145 as the last “resistant” entry in our list, whereas their file now has NCI-H522. (This point is discussed in more detail in L04-getTrainingCellLines.pdf.) Using the cell lines we list allows us to perfectly match the heatmap shown in Figure 2A of Potti et al. [3] (although the heatmap for taxol/Paclitaxel is actually mislabeled as coming from Cyclophosphamide there). In terms of simply getting the direction of change, altering one sample out of the set of 17 will not revise matters.

Now that we have the expression data, we need the list of genes to focus on. These lists have been revised over time; we’re using the ones that were posted on their web site as of Nov 8, 2007.

> taxolGenes <- read.table(file.path("OtherData", "GeneLists",
+ "Taxol(final).txt"), sep = "\t", header = TRUE, quote = "")
> dim(taxolGenes)

[1] 40 8

> colnames(taxolGenes)

[7] "Pathway"  "Pathway.Hyperlink"
> taxolProbesets <- taxolGenes[, "Probe.Set.ID"]
> taxolProbesets <- as.character(taxolProbesets)

Now we need to load the chip comparer files to establish the mapping. The format of the chip comparer file differs depending on which of the two chip platforms is specified first, so we ran it both ways. We also trimmed the file names returned to clarify the meaning.

> u95ToU133 <- read.table(file.path("OtherData", "U95Av2ToU133Av2.csv"),
+     sep = ",", skip = 2)
> colnames(u95ToU133) <- c("U95Av2.probeset", "U95Av2.Locus", "U95Av2.Gene",
+     "U133Av2.probeset", "U133Av2.Locus", "U133Av2.Gene")

> u133ToU95 <- read.table(file.path("OtherData", "U133Av2ToU95Av2.csv"),
+     sep = ",", skip = 2)
> colnames(u133ToU95) <- c("U133Av2.probeset", "U133Av2.Locus",
+     "U133Av2.Gene", "U95Av2.probeset", "U95Av2.Locus", "U95Av2.Gene")

We want to establish the possibly many-to-one mappings involved. We can do this by grabbing the locus ids of the U95Av2 probesets involved.

> match(taxolProbesets, as.character(u133ToU95$U95Av2.probeset))

[1] 1042 19829 3013 2195 NA NA 9455 8193 13862 9852 15666
[13] NA 2890 641 11795 6043 1250 22981 20816 15984 3433 7433 4683
[25] 4367 4179 NA 16233 5077 4358 22390 3272 10004 22608 19006
[37] 959 5979 1377 NA

> taxolLoci <- as.character(u133ToU95$U95Av2.Locus[match(taxolProbesets,
+     u133ToU95$U95Av2.probeset)])
> names(taxolLoci) <- taxolProbesets
> unique(taxolLoci)

[1] "2063" "7155" "5916" "5111" "5781" NA
[7] "472" "4776" "158927" "786" "25770" "5296"
[13] "A1810767" "4626" "51097" "57157" "23096" "7222"
[19] "10906" "9843" "11063" "2628" "4000" "3064"
[25] "25844" "23549" "5101" "116138" "8273" "9355"
[31] "1164" "6398" "7536" "8986" "8445" "2665"

> as.character(u133ToU95$U133Av2.probeset[u133ToU95$U133Av2.Locus ==
+     taxolLoci[1]])

[1] "209261_s_at" "213364_s_at" "209262_s_at"

As it happens, we can only find matches for 35 of the 40 genes that they now report; the match returns 5 “NA”s. We then grab the Locus IDs for all 40. Checking the list, we see that all of the non-NA Loci are distinct, so we don’t need to worry about averaging results from multiple U95Av2 probesets. Given the loci, we can extract the matching U133Av2 probesets as illustrated above for the case of the first locus, 2063, which links to 3 U133A probesets.

At this point, we want to loop through the taxol probesets. For each, we first want to establish whether the mean expression level is higher in the sensitive cell lines or in the resistant cell lines.
> taxolDirection <- rep("NULL", length(taxolProbesets))
> names(taxolDirection) <- taxolProbesets
> for (i1 in 1:length(taxolProbesets)) {
+   if (mean(log2(taxolMat[taxolProbesets[i1], taxolInfo$responseStatus == "Sensitive"])) > mean(log2(taxolMat[taxolProbesets[i1], taxolInfo$responseStatus == "Resistant")))) {
+     taxolDirection[i1] <- "HigherInSensitive"
+   }
+   else {
+     taxolDirection[i1] <- "HigherInResistant"
+   }
+ }

Then, assuming that there are matching U133A probesets, we extract and rank the intensities as described above.

> taxolRanks <- matrix(0, nrow = length(taxolProbesets), ncol = dim(tfacExpression)[2])
> rownames(taxolRanks) <- taxolProbesets
> colnames(taxolRanks) <- colnames(tfacExpression)
> for (i1 in 1:length(taxolProbesets)) {
+   if (!is.na(taxolLoci[taxolProbesets[i1]])) {
+     tempProbesets <- as.character(u133ToU95$U133Av2.probeset[u133ToU95$U133Av2.Locus == taxolLoci[taxolProbesets[i1]]])
+     tempData <- tfacExpression[tempProbesets, ]
+     if (taxolDirection[taxolProbesets[i1]] == "HigherInResistant") {
+       tempData <- -tempData
+     }
+     tempRanks <- t(apply(tempData, 1, rank))
+     taxolRanks[taxolProbesets[i1], ] <- apply(tempRanks, 2, mean)
+   }
+ }
> taxolScores <- apply(taxolRanks, 2, sum)

We now want to see (a) whether these taxol scores are different for responders than for nonresponders (we expect them to be higher for responders based on the data from Potti et al. [3]), and (b) how these scores relate to the final probabilities that they report.

We check for stratification by status first. There was a clear split in the Potti et al. [3] data; responders were more sensitive.

> plot(c(0.5, 2.5), c(600, 1250), type = "n", axes = FALSE, xlab = "Taxol: Responders and Nonresponders", ylab = "Taxol Rank Score", ylim = c(600, 1250), main = "Taxol Scores by Response Status")
> points(jitter(rep(1, sum(tfac$pCR)), amount = 0.25), taxolScores[tfac$pCR == 1], col = "blue")
> points(jitter(rep(2, sum(1 - tfac$pCR)), amount = 0.25), taxolScores[tfac$pCR == 0], col = "red")
> meanResp <- mean(taxolScores[tfac$pCR == 1])
> meanNonResp <- mean(taxolScores[tfac$pCR == 0])
> segments(c(1:2) - 0.4, c(meanResp, meanNonResp), c(1:2) + 0.4,
```r
+ c(meanResp, meanNonResp), col = "black", lwd = 1)
> axis(side = 1, at = c(1, 2), labels = c("Responders", "Nonresponders"))
> axis(side = 2, at = seq(from = 600, to = 1250, by = 100), labels = TRUE)
> box(bty = "l")
> wilcox.test(taxolScores[tfac$pCR == 1], taxolScores[tfac$pCR == 0])

Wilcoxon rank sum test

data:  taxolScores[tfac$pCR == 1] and taxolScores[tfac$pCR == 0]
W = 251, p-value = 0.9405
alternative hypothesis: true location shift is not equal to 0

> t.test(taxolScores[tfac$pCR == 1], taxolScores[tfac$pCR == 0])$p.value

[1] 0.9738112
```

**Taxol Scores by Response Status**
Our scores do not differ as a function of response status, either visually or according to \( t \) or Wilcoxon tests.

Let’s take a look at how well their taxol probabilities agree with our scores.

```r
> plot(tfac$TaxolProbSensitive, taxolScores, xlab = "Reported Prob of Taxol Sensitivity", + ylab = "Our Taxol Score: Higher is More Sensitive")
```

The two sets of scores do not agree at all; there is no concentration near the main diagonal or even loose monotonicity. Several samples are placed in the off-diagonal corners, indicating a high value from one and a low value from the other.

Now let’s check our taxol scores against their overall combined probabilities.

```r
> plot(tfac$ComboProbCentered, taxolScores, xlab = "Reported Prob of Combo Sensitivity", + ylab = "Our Taxol Score: Higher is More Sensitive")
```
Again, there is no association.

6 Summary

The rule used by Potti et al. [3] for combining probabilities of four events,

\[ P(T) + P(F) + P(A) + P(C) - P(T) \times P(F) \times P(A) \times P(C), \]

followed by interpolating to the [0,1] range, is an incorrect generalization of the rule for combining results from two. This can distort predictions substantially.

Wholly aside from reproducing their results for combination therapy, we are unable to reproduce their results for the most important single agent. The scores we compute for taxol fail to separate responders from nonresponders.
7 Appendix

7.1 Saves

7.2 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 States.1252;LC_NUMERIC=C;LC_TIME=English_United States.1252

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

other attached packages:

XML affy affyio Biobase ROC "1.9-0" "1.14.2" "1.4.1" "1.14.1" "1.10.0"

References

