Creating heatmaps using package Heatplus

Alexander Ploner
Medical Epidemiology & Biostatistics
Karolinska Institutet, Stockholm
email: alexander.ploner@ki.se

October 24, 2023

Abstract

The package Heatplus offers several functions for producing heatmaps, specifically annotated heatmaps that display extra information about samples and/or features (variables) in panels beside the main plot and the dendrograms. This documents demonstrates some basic applications to gene expression data.

Contents

1 Setup

We use the example data from Biobase for demonstration:

```
> require(Biobase)
> data(sample.ExpressionSet)
> exdat = sample.ExpressionSet
```

We also generate a shortlist of genes that are associated with the phenotype type:

```
> require(limma)
> design1 = model.matrix( ~ type, data=pData(exdat))
> lm1 = lmFit(exprs(exdat), design1)
> lm1 = eBayes(lm1)
> geneID = rownames(topTable(lm1, coef=2, num=100, adjust="none",
+         p.value=0.05))
> length(geneID)
```

```
[1] 46
```

(Of course, this is only for the example's sake, as it does not account for multiple testing.)

```
> exdat2 = exdat[geneID,]
```

2 Regular heatmaps

The function regHeatmap generates heatmaps without annotation. Apart from some display defaults, this is very similar to standard heatmap, but allows you to add a simple legend.

Figure 1 shows the full example data with the default settings. Figure 2 shows the gene expression for the short list, with the legend moved to the right, for simpler breaks of the intensity scale and a different palette. Figure 3 shows how different distance- and clustering functions can be passed in. Figure 4 shows how the lists of arguments can be used to specify different settings for row- and column dendrograms.

```
> require(Heatplus)
> reg1 = regHeatmap(exprs(exdat))
> plot(reg1)
```

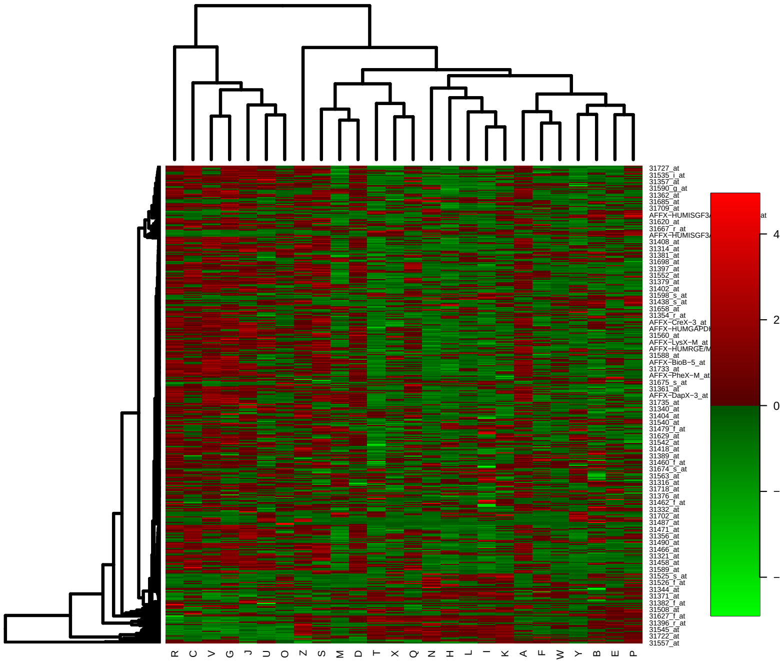

Figure 1: Heatmap with row- and column dendrograms and a legend for 500 genes and 26 samples.

```
> reg2 = regHeatmap(exprs(exdat2), legend=2, col=heat.colors, breaks=-3:3)
> plot(reg2)
```

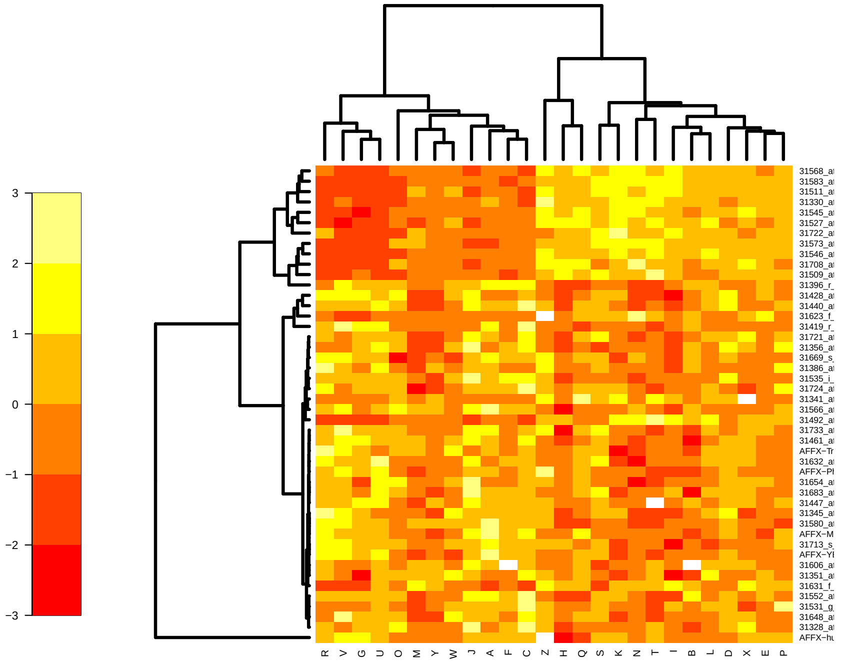

Figure 2: Heatmap with row- and column dendrograms and a legend for 46 genes and 26 samples. Legend placement, color scheme and intervals have been changed compared to the default

```
> corrdist = function(x) as.dist(1-cor(t(x)))
> hclust.avl = function(x) hclust(x, method="average")
> reg3 = regHeatmap(exprs(exdat2), legend=2, dendrogram =
+   list(clustfun=hclust.avl, distfun=corrdist))
> plot(reg3)
```

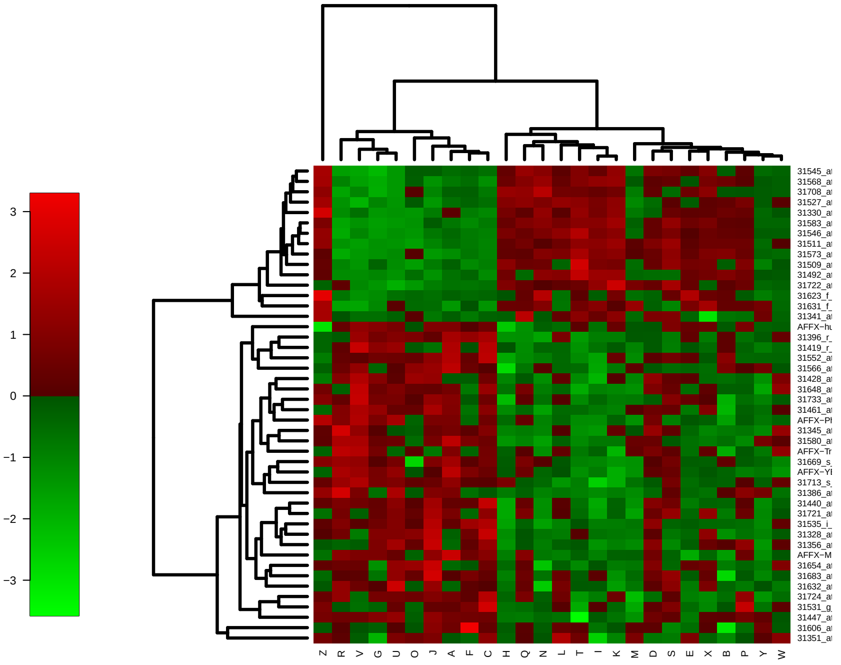

Figure 3: Heatmap with row- and column dendrograms and a legend for 46 genes and 26 samples. Distance measure (correlation distance instead of Euclidean) and clustering method (average instead of complete linkage) for the dendrogram are changed compared to the default.

```
> reg4 = regHeatmap(exprs(exdat2), legend=3,
+   dendrogram=list(Col=list(status="hide")))
> plot(reg4)
```

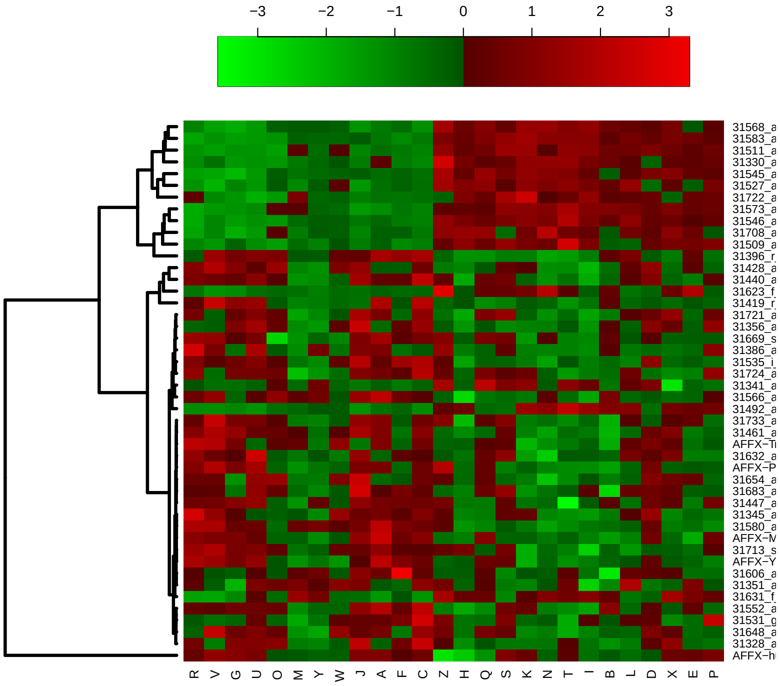

Figure 4: Heatmap with a row dendrogram and a legend, for 46 genes and 26 samples. The column dendrogram is not shown, though the samples are still arranged as in Figure 2, i.e. sorted according to the hidden column dendrogram.

3 Annotated heatmaps

Figure 5 shows the default annotated heatmap for the full data set. Figure 6 shows the default annotated heatmap for the smaller data set, with the column dendrogram cut at distance 5000.

Figure 7 shows a similar plot as Figure 6, but cut at a distance of 7500 (resulting in two instead of three clusters) and with costumised cluster labels.

4 Double-annotated heatmaps

We can also add annotation information about the features. These can be of all kinds: quality information, relationship with sample annotation, or membership in different pathways. Let's start with the median of the standard errors as a quality control measure:

```
> SE = apply(get("se.exprs", assayData(exdat2)), 1, median)
```

Then we look at the correlation of the features with the variable `score` in the phenotype data, plus t-statistic and p-value:

```
> CO = cor(t(exprs(exdat2)), pData(exdat2)$score)
> df = nrow(exdat2)-2
> TT = sqrt(df) * CO/sqrt(1-CO^2)
> PV = 2*pt(-abs(TT), df=df)
```

Finally, we want to see which of the features are associated with the GeneOntology category *translational elongation*:

```
> require(hgu95av2.db)
> allGO = as.list(mget(featureNames(exdat2), hgu95av2GO))
> isTransElong = sapply(allGO, function(x) "GO:0006414" %in% names(x))
```

Let's put this into an annotation data frame:

```
> annFeatures = data.frame(standard.errors=SE, sigCorScore=PV<0.05,
+ isTransElong)
```

```
> ann1 = annHeatmap(exprs(exdat), ann=pData(exdat))
> plot(ann1)
```

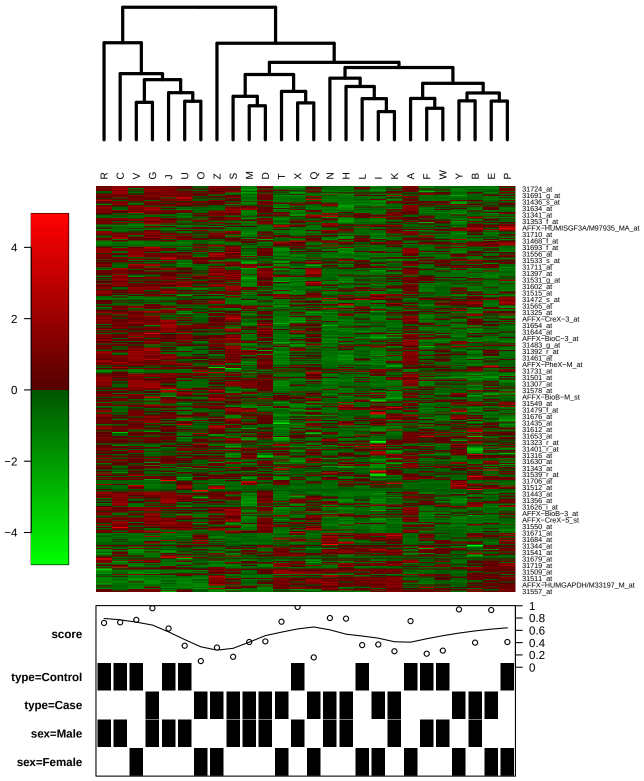

Figure 5: Annotated heatmap with column dendrogram and a legend, for 500 genes and 26 samples.

```
> ann2 = annHeatmap(exprs(exdat2), ann=pData(exdat2),
+          cluster=list(cuth=5000))
> plot(ann2)
```

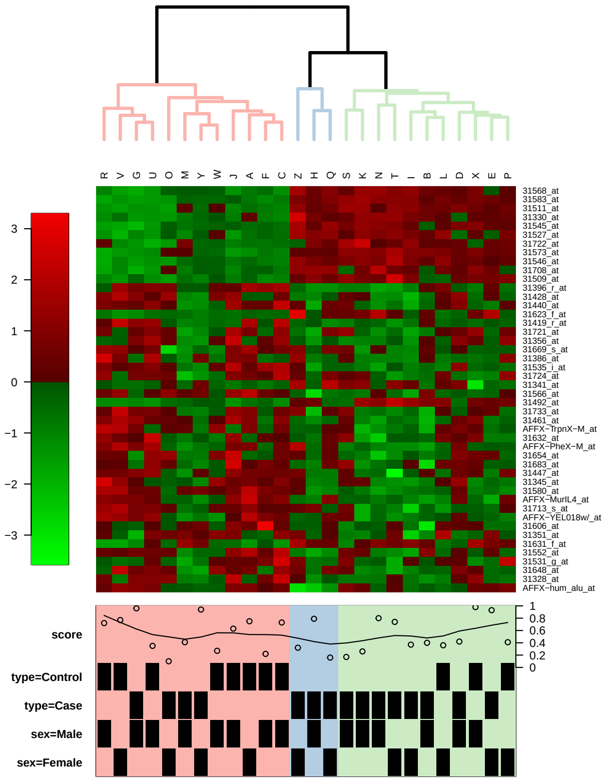

Figure 6: Annotated heatmap with column dendrogram and a legend, for 46 genes and 26 samples. The column dendrogram is cut at $h = 5000$

```
> ann3 = annHeatmap(exprs(exdat2), ann=pData(exdat2),
+   cluster=list(cuth=7500, label=c("Control-like", "Case-like")))
> plot(ann3)
```

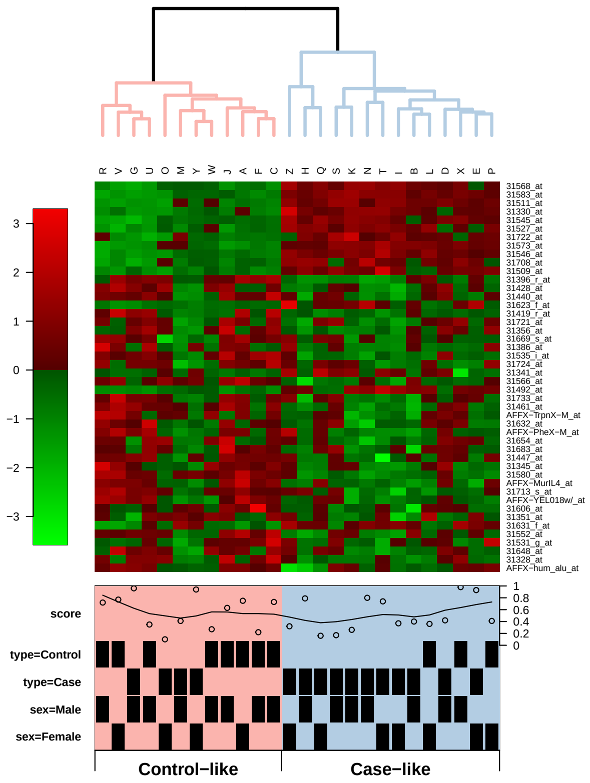

Figure 7: Annotated heatmap with column dendrogram and a legend, for 46 genes and 26 samples. The column dendrogram is cut at $h = 7500$, and we add cluster names, based on the annotation.

Figure 8 shows the smaller data set with default settings. Figure 9 changes the space alloted for the feature- and sample labels. Figure 10 additionally slims the annotation data frames by excluding the reference levels for the factor variables; it also shows how the plot method allows changing the proportions of the heatmap in display. Figure 11 shows how to construct an annotation data frame beforehand using `convAnnData` and passing it in unchanged using the `asIs` switch in the `annotation` argument.

5 Roadmap

The following improvements are planned for the next release:

- Improved precision for coloring the cutting height in dendrograms
- Specification of common clustering algorithms and distance measures via character codes (instead of costum wrapper functions for `hclust` and `dist`)

The medium term goal is of course to add more specific methods for common Bioconductor classes and relevant objects, as well as providing convenience functions for including biological annotation data more easily, but there is no detailed road map for this yet.

```
> ann4 = annHeatmap2(exprs(exdat2),
+   ann=list(Col=list(data=pData(exdat2)), Row=list(data=annFeatures)))
> plot(ann4)
```

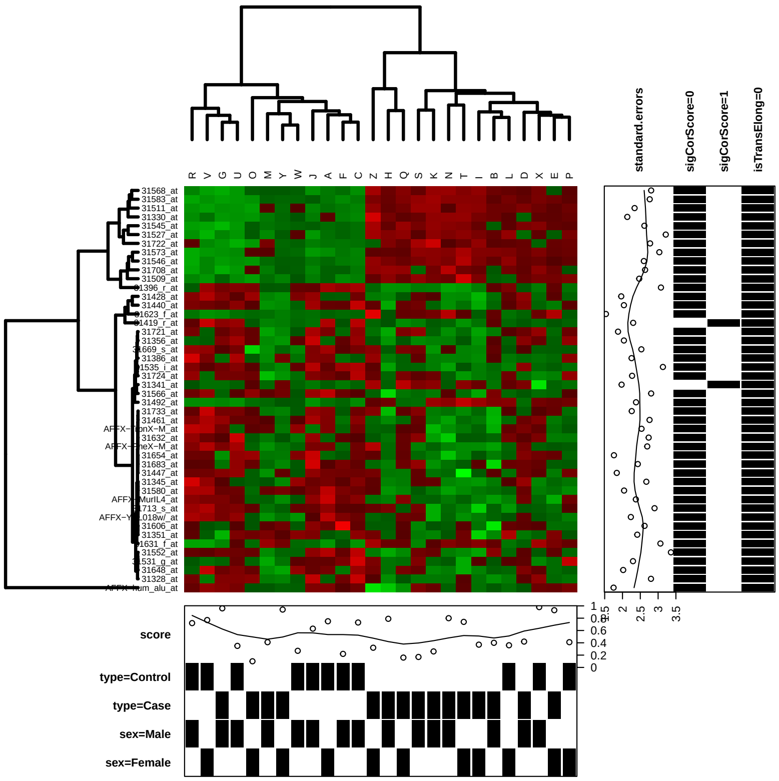

Figure 8: Double annotated heatmap with row- and column dendrograms and annotation, plus a legend, for 46 genes and 26 samples.

```
> ann4a = annHeatmap2(exprs(exdat2),
+   ann=list(Col=list(data=pData(exdat2)), Row=list(data=annFeatures)),
+   labels=list(Row=list(nrow=7), Col=list(nrow=2)))
> plot(ann4a)
```

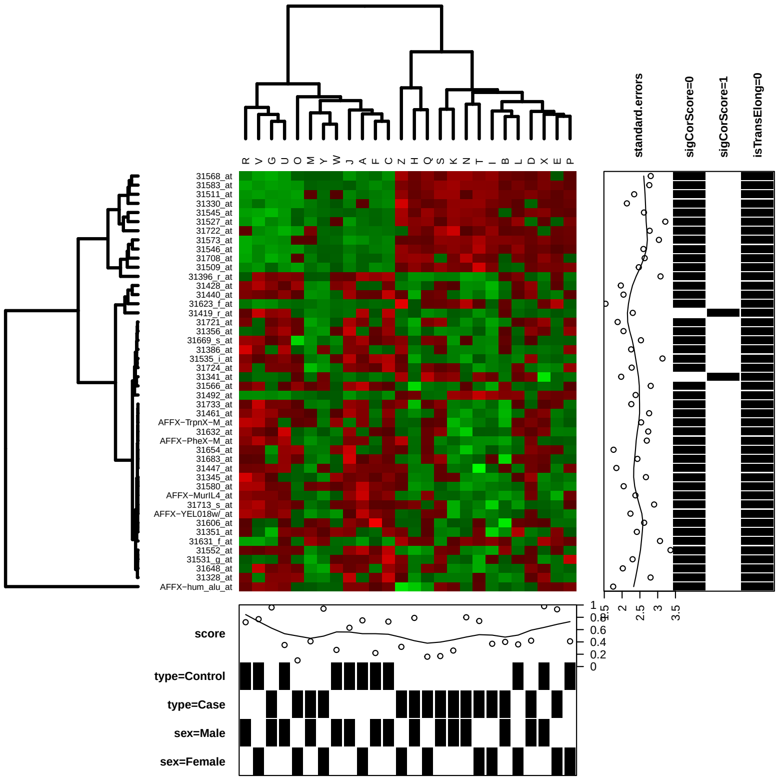

Figure 9: Double annotated heatmap with row- and column dendrograms and annotation, plus a legend, for 46 genes and 26 samples, with modified space for column/row labels.

```
> ann4b = annHeatmap2(exprs(exdat2),
+   ann=list( inclRef=FALSE,
+                   Col=list(data=pData(exdat2)),
+                   Row=list(data=annFeatures)   ),
+         labels=list(Row=list(nrow=7), Col=list(nrow=2)))
> plot(ann4b, widths=c(2,5,1), heights=c(2,5,1))
```

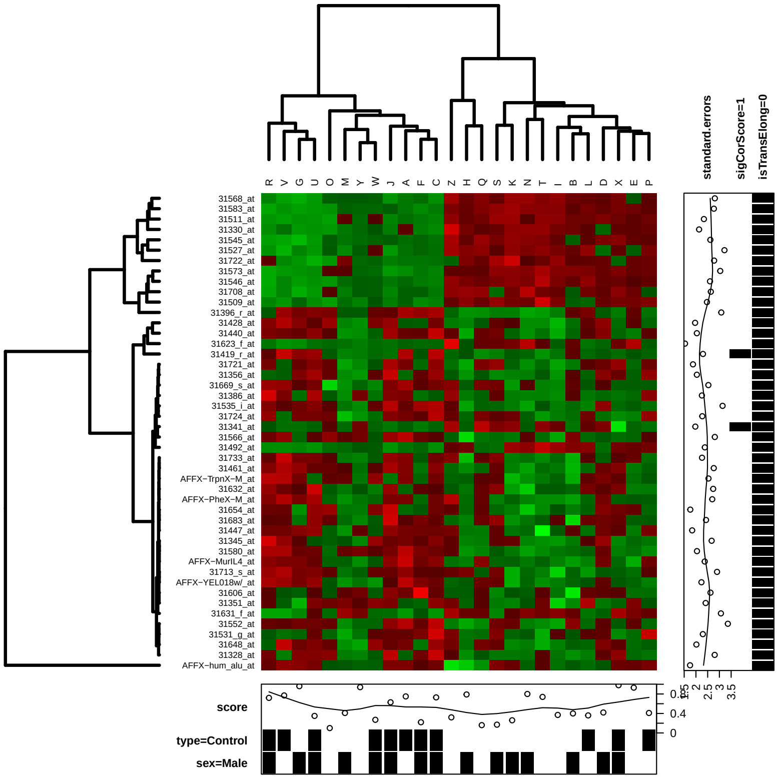

Figure 10: Double annotated heatmap with row- and column dendrograms and annotation, plus a legend, for 46 genes and 26 samples. Extra space for labels, no reference levels for annotation data, and changed proportions of the plot.

```
> ann1 = convAnnData(pData(exdat2), inclRef=FALSE)[, 3:1]
> colnames(ann1) = c("Score", "isControl", "isMale")
> ann2 = convAnnData(annFeatures, inclRef=FALSE)[, 3:1]
> colnames(ann2) = c("isTranslationalElongation", "isCorrelated", "SE")
> ann4c = annHeatmap2(exprs(exdat2),
+   ann=list(asIs=TRUE, Col=list(data=ann1), Row=list(data=ann2)),
+   labels=list(Row=list(nrow=7), Col=list(nrow=2)))
> plot(ann4c)
```

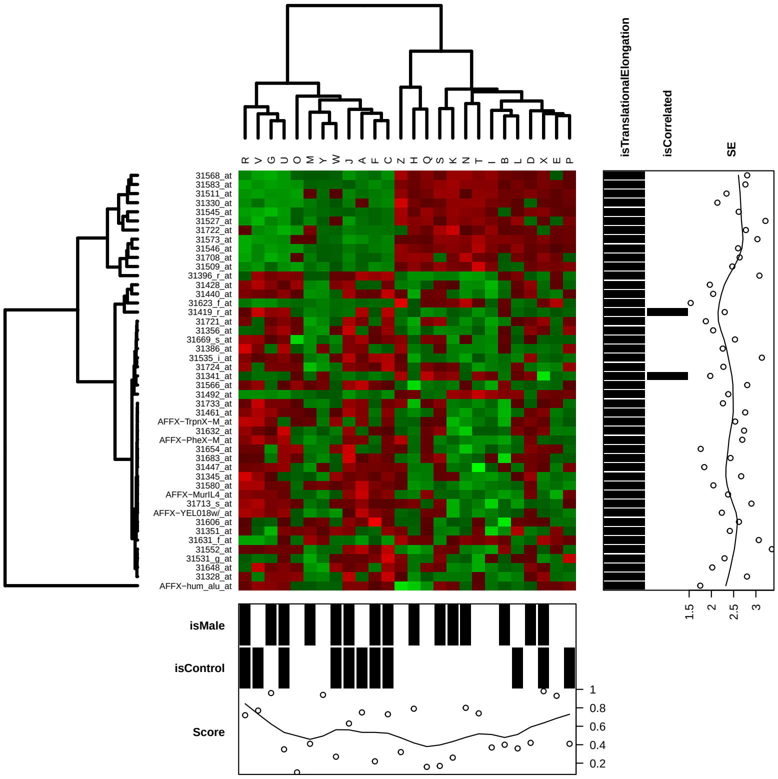

Figure 11: Double annotated heatmap with row- and column dendrograms and annotation, plus a legend, for 46 genes and 26 samples, with refined annotation data frames passed in and used unchanged.