Result Visualization for P2C2M

Michael Gruenstaeudl

October 21, 2014

This vignette provides two examples of how the results of the R package **P2C2M** can be visualized using ggplot2 [2].

1 Visualization Examples

1.1 Ranked vs. Unranked Comparisons

In order to evaluate if the approach of comparing descriptive statistics in a pairwise fashion after sorting the values by rank is more likely to exclude false negative results than the approach employed by [1], a comparison of ranked versus unranked descriptive statistics values was conducted.

Example data is loaded into R. An alpha value is set. Several lists are initialized. Gene names are specified.

```
> library(P2C2M)
> data(viz_example_1)
> inp = viz_example_1
> alpha = 0.05
> inData = qnts = df = titles = list()
> df$lwr = df$upr = list()
> titles$sorted = sprintf("gene%02d_sorted", c(1:10))
> titles$unsorted = sprintf("gene%02d_unsorted", c(1:10))
```

The example data is converted into a stacked format. Upper and lower quantiles are calculated.

```
> colnames(inp$sorted) = titles$sorted
> inData$sorted = stack(as.data.frame(inp$sorted))
> colnames(inData$sorted) = c("value", "gene")
> qnts$sorted = apply(inp$sorted, 2, quantile, c(alpha, 1-alpha), na.rm=TRUE)
> df$lwr$sorted = data.frame(lwrQntl=qnts$sorted[1,], gene=names(qnts$sorted[1,]))
> df$upr$sorted = data.frame(uprQntl=qnts$sorted[2,], gene=names(qnts$sorted[2,]))
```

The same step as above is performed for data set that was generated without sorting.

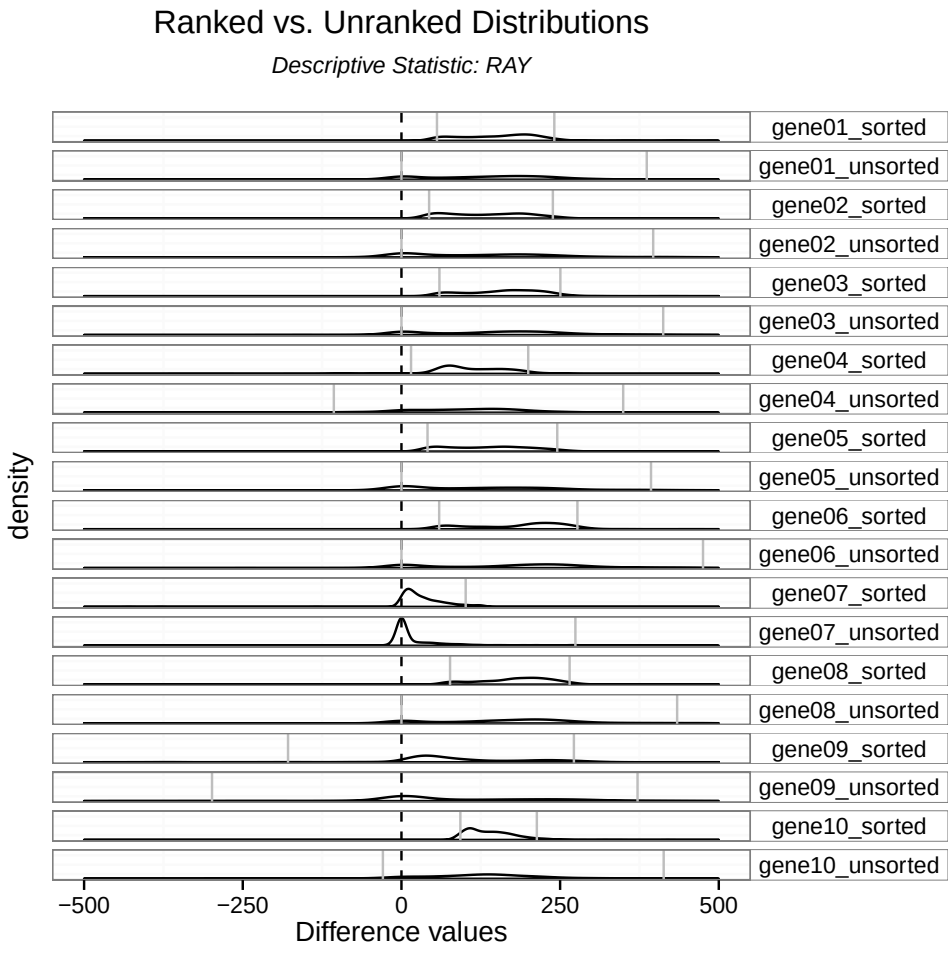
```
> colnames(inp$unsorted) = titles$unsorted
> inData$unsorted = stack(as.data.frame(inp$unsorted))
> colnames(inData$unsorted) = c("value", "gene")
> qnts$unsorted = apply(inp$unsorted, 2, quantile, c(alpha, 1-alpha), na.rm=TRUE)
> df$lwr$unsorted = data.frame(lwrQntl=qnts$unsorted[1,], gene=names(qnts$unsorted[1,]))
> df$upr$unsorted = data.frame(uprQntl=qnts$unsorted[2,], gene=names(qnts$unsorted[2,]))
```

Both sets of data are combined and ordered via factors.

```
> inData = rbind(inData$sorted, inData$unsorted)
> dfLwr = rbind(df$lwr$sorted, df$lwr$unsorted)
> dfUpr = rbind(df$upr$sorted, df$upr$unsorted)
> inData$gene = factor(inData$gene, levels = sort(c(titles$sorted, titles$unsorted)))
```

The distributions of differences of the descriptive statistic 'RAY' are visualized as stacked density distributions.

```
> library(ggplot2)
> ggplot(data=inData, aes(x=value)) +
+   geom_density() +
+   facet_grid(gene~.) +
+   labs(x="Difference values") +
+   ggtitle(expression(atop("Ranked vs. Unranked Distributions",
+                       atop(italic("Descriptive Statistic: RAY"), "")))) +
+
+   theme_bw() +
+   theme(axis.text.y=element_blank(),
+         axis.ticks.y=element_blank(),
+         strip.text.y=element_text(angle=0),
+         panel.grid.major.x=element_blank(),
+         panel.grid.major.y=element_blank(),
+         strip.background=element_rect(fill="white")) +
+   # Limits on the x-axis improve the visualization
+   xlim(-500, 500) +
+   geom_vline(xintercept=0, linetype = "dashed") +
+   geom_vline(aes(xintercept=lwrQntl), dfLwr, color="grey") +
+   geom_vline(aes(xintercept=uprQntl), dfUpr, color="grey")
```



1.2 Distribution of False Positives

In order to visualize the sensitivity to false positive results of the different descriptive statistics implemented in P2C2M, a graphical comparison is generated.

Example data is loaded into R.

```
> library(P2C2M)
> data(viz_example_2)
> inp = viz_example_2
```

A custom function is specified which converts results matrices into presence/absence matrices, stacks the matrix columns and adds identifier information.

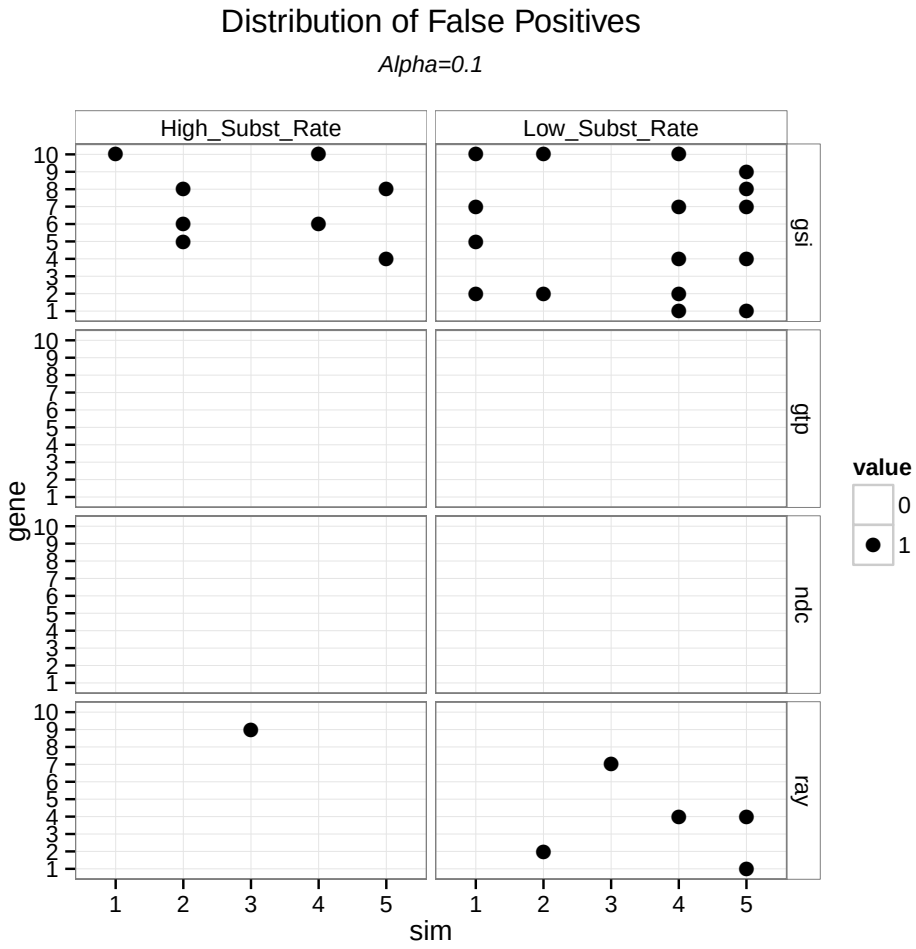
```
> myfunc = function(inData, simNum){
+   handle = inData
+   colnames(handle) = c("gtp", "ray", "ndc", "gsi")
+   # Convert results into presence/absence matrix
+   handle[!grepl("n.s.", handle)] = 1
+   handle[grepl("n.s.", handle)] = 0
+   # Stack the individual descriptive statistics
+   handle = stack(data.frame(handle, stringsAsFactors=FALSE))
+   colnames(handle)[1] = "value"
+   colnames(handle)[2] = "stat"
+   # Add gene identifiers (under the assumption that there are 10 genes)
+   handle[,3] = rep(c(1:10), 4)
+   colnames(handle)[3] = "gene"
+   handle[,4] = simNum
+   colnames(handle)[4] = "sim"
+ return(handle)
+ }
```

The custom function is executed on the example data, which consists of two subsets that are characterized by different substitution rates.

```
> highL = list()
> sims = as.numeric(names(inp$High))
> for (i in 1:length(inp$High)) {highL[[i]] = myfunc(inp$High[[i]], sims[i])}
> High = do.call("rbind", highL)
> High[,ncol(High)+1] = "High_Subst_Rate"
> colnames(High)[ncol(High)] = "ratetype"
> lowL = list()
> sims = as.numeric(names(inp$Low))
> for (i in 1:length(inp$Low)) {lowL[[i]] = myfunc(inp$Low[[i]], sims[i])}
> Low = do.call("rbind", lowL)
> Low[,ncol(Low)+1] = "Low_Subst_Rate"
> colnames(Low)[ncol(Low)] = "ratetype"
> inData = rbind(High, Low)
```

The distribution of false positive result values is visualized as presence/absence plot.

```
> library(ggplot2)
> ggplot(data=inData, aes(x=sim,y=gene)) +
+   geom_point(aes(colour=value), size = 3) +
+   scale_colour_manual(values = c(NA,'black')) +
+   facet_grid(stat~ratetype) +
+   ggtitle(expression(atop("Distribution of False Positives",
+           atop(italic("Alpha=0.1") , "")))) +
+   theme_bw() +
+   scale_x_discrete(breaks=c(1:5), labels=c(1:5)) +
+   scale_y_discrete(breaks=c(10:1), labels=c(10:1)) +
+   theme(strip.background = element_rect(fill="white")
+   )
```



Acknowledgements

I would like to thank Paul D. Blischak from the Ohio State University and Teofil Nakov from the University of Arkansas for help with testing sections of the above code.

References

[1] N M Reid, J M Brown, J D Satler, T A Pelletier, J D McVay, S M Hird, and B C Carstens. Poor fit to the multi-species coalescent model is widely detectable in empirical data. *Systematic Biology*, 63:322–333, 2014.

[2] H Wickham. *ggplot2: elegant graphics for data analysis*. Springer, New York, 2009.