RchyOptimyx: Gating Hierarchy Optimization for Flow Cytometry

Nima Aghaeepour and Adrin Jalali

April 10, 2013

naghaeep@bccrc.ca

Contents

1 Licensing

2 Introduction

This document demonstrates the functionality of the RchyOptimyx package, a tool for cellular hieraRCHY OPTIMization for flow cytometry data (named after Archeopteryx).

RchyOptimyx models all possible gating strategies and marker panels that can be generated using a high-color assay, and uses dynamic programing and optimization tools from graph-theory to determine the minimal sets of markers that can identify a target population to a desired level of purity. A cellular

hierarchy is a directed acyclic graph (DAG), embedded in a plane as a top-down diagram, with one node on the top most level representing all cells (or a major component therefore, such as T-cells) and nodes further down showing more specific cell populations. All the intermediate cell populations are placed in the hierarchy using parent-child relationships. The graph starts from level 0 to level m including i-marker phenotypes on i^{th} level. The phenotype with 0 markers is the top most phenotype with all cells and the phenotype with m markers is the cell population of interest.

The required input phenotypes and their respective scores (target values of the optimization) can be generated either using manual gates or automated gating algorithms (see the *flowType* package in Bioconductor for examples).

3 First Example: Preparing Raw Data for RchyOptimyx

In this example, we start from a raw flowSet and generated the required materials to produce an RchyOptimyx graph. The dataset consists of a flowSet *HIVData* with 18 HIV$^+$ and 13 normals and a matrix *HIVMetaData* which consists of FCS filename, tube number, and patient label. In this example, we are interested in the second tube only. For more details please see the *flowType* package.

```
> library(flowType)
> data(HIVData)
> data(HIVMetaData)
> HIVMetaData <- HIVMetaData[which(HIVMetaData[,'Tube']==2),];
```

We convert the subject labels so that HIV$^+$ and normal subjects are labeled 2 and 1, respectively.

```
> Labels=(HIVMetaData[,2]=='+')+1;
```

3.1 Processing using flowType

We start by calculating the cell proportions using flowType:

```
> library(flowCore);
> library(RchyOptimyx);
> ##Markers for which cell proportions will be measured.
> PropMarkers <- 5:10
> ##Markers for which MFIs will be measured.
> MFIMarkers <- PropMarkers
> ##Marker Names
> MarkerNames <- c('Time', 'FSC-A','FSC-H','SSC-A',
+                  'IgG','CD38','CD19','CD3',
+                  'CD27','CD20', 'NA', 'NA')
```

```
> ##Apply flowType
> ResList <- fsApply(HIVData, 'flowType', PropMarkers,
+                    MFIMarkers, 'kmeans', MarkerNames);
> ##Extract phenotype names
> phenotype.names=names(ResList[[1]]@CellFreqs)
```

Then we extract all cell proportions from the list of flowType results and normalize them by the total number of cells in each sample to create the all.proportions matrix.

```
> all.proportions <- matrix(0,3^length(PropMarkers)
+                           -1,length(HIVMetaData[,1]))
> for (i in 1:length(ResList)){
+   all.proportions[,i] = ResList[[i]]@CellFreqs
+   all.proportions[,i] = all.proportions[,i] /
+     max(all.proportions
+         [which(names(ResList[[i]]@CellFreqs)==''),i])
+ }
```

We use a t-test to select the phenotypes that have a significantly different mean across the two groups of patients (FDR=0.05). Remember that in real world use-cases the assumptions of a t-test must be checked or a resampling-based alternative (e.g., a permutation test) should be used. Sensitivity analysis (e.g., bootstrapping) is also necessary. Eight phenotypes are selected as statistically significant.

```
> Pvals <- vector();
> EffectSize <- vector();
> for (i in 1:dim(all.proportions)[1]){
+
+   ##If all of the cell proportions are 1 (i.e., the phenotype
+   ##with no gates) the p-value is 1.
+   if (length(which(all.proportions[i,]!=1))==0){
+     Pvals[i]=1;
+     EffectSize[i]=0;
+     next;
+   }
+   temp=t.test(all.proportions[i, Labels==1],
+     all.proportions[i, Labels==2])
+   Pvals[i] <- temp$p.value
+   EffectSize[i] <- abs(temp$statistic)
+ }
> Pvals[is.nan(Pvals)]=1
> names(Pvals)=phenotype.names
> ##Bonferroni's correction
> selected <- which(p.adjust(Pvals)<0.1);
> print(names(selected))
```

```
[1] "IgG-CD38-CD19-CD27+CD20-" "IgG-CD38-CD19-CD27+"
[3] "IgG-CD38-CD27+CD20-"      "IgG-CD38-CD27+"
[5] "IgG-CD19-CD27+CD20-"      "IgG-CD19-CD27+"
[7] "IgG-CD27+CD20-"           "IgG-CD27+"
```

3.2 Basic RchyOptimyx Functionality

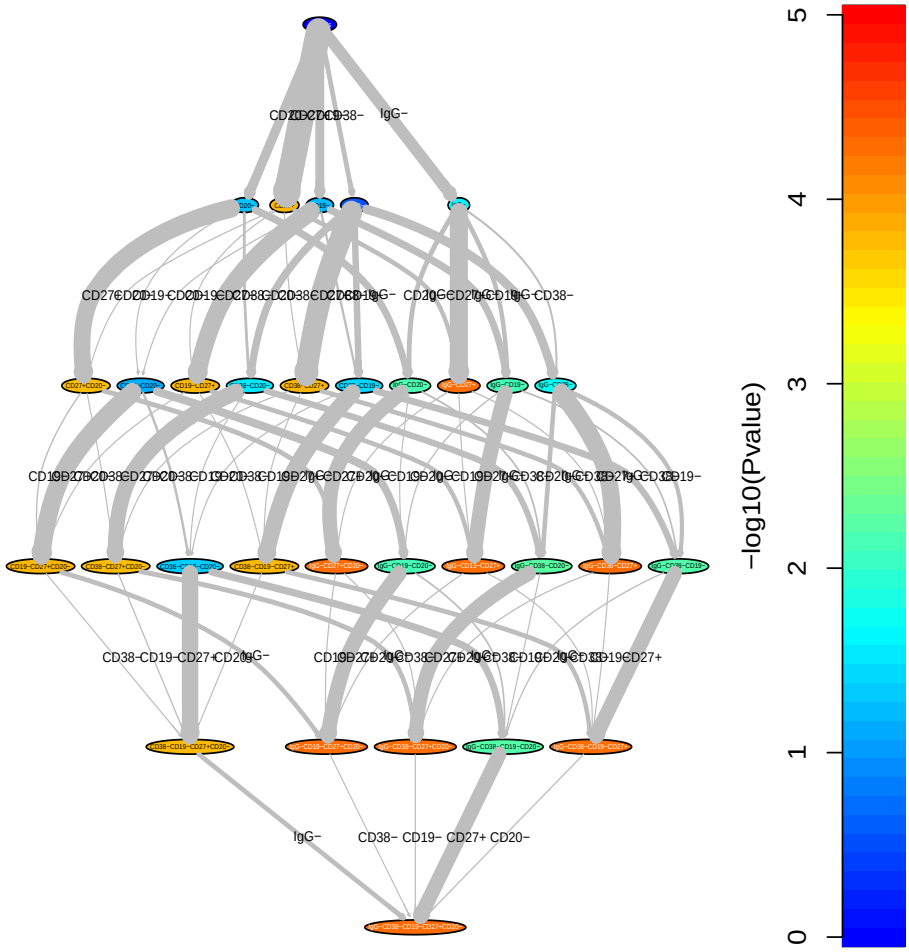
We select the longest one (*IgG-CD38-CD19-CD27+CD20-*) for further analysis using RchyOptimyx. First we need to create the *Signs* matrix. We use the *digitsBase* number to generate a matrix with $3^{length(PropMarkers)} - 1$ rows and $length(PropMarkers)$ columns. flowType produces it's results in the exact same order, making assigning row and column names easy.

```
> library(sfsmisc)
> Signs=t(digitsBase(1:(3^length(PropMarkers)-1),
+   3,ndigits=length(PropMarkers)))
> rownames(Signs)=phenotype.names;
> colnames(Signs)=MarkerNames[PropMarkers]
> head(Signs)
```

```
[1] 0 0 0 0 0 0
```

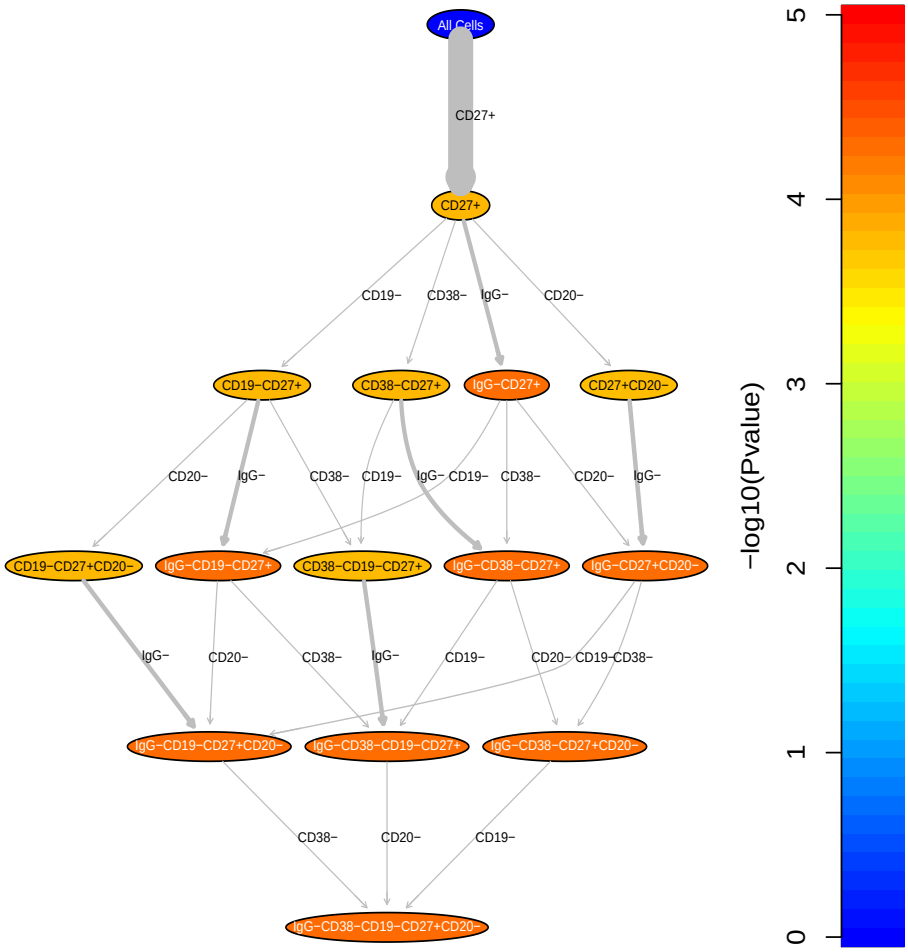
Now we can plot the complete hierarchy:

```
> res<-RchyOptimyx(Signs, -log10(Pvals),
+                  paste(Signs['IgG-CD38-CD19-CD27+CD20-',],
+                        collapse=''), factorial(6), FALSE)
> plot(res, phenotypeScores=-log10(Pvals), ylab='-log10(Pvalue)')
```



and an optimized hierarchy (with only the top 15 paths):

```
> res<-RchyOptimyx(Signs, -log10(Pvals),
+                  paste(Signs['IgG-CD38-CD19-CD27+CD20-',],
+                        collapse=''), 15, FALSE)
> plot(res, phenotypeScores=-log10(Pvals), ylab='-log10(Pvalue)')
```



4 Analysis of a Large Dataset

In this section we will describe two use-cases of RchyOptimyx using a real-world clinical dataset of 17-color assays of 466 HIV$^+$ subjects. We start by loading the library (for installation guidelines see the Bioconductor website).

```
> library(RchyOptimyx)
> data(HIVData)
```

The *HIVData* dataset consists of a matrix (*Signs*) and 2 numeric vectors *LogRankPvals* and *OverlapScores*). The *Signs* matrix consists of 10 columns (one per measured marker) and $3^{10} - 1 = 59048$ rows (one per immunophenotype) similar to the previous example. See [1] or the flowType package for more

details. For every phenotype (row), the entity corresponding to a given marker (column) can be 0, 1, and 2 for negative, neutral, and positive respectively. The row names and column names are set respectively.

LogRankPvals is a vector of log-rank test P-values to determine the correlation between HIV's progression and each of the measured immunophenotypes in 466 HIV positive patients (lower values represent a stronger correlation). For more details see [1]. The names of the vector match the names of the *Signs* matrix.

Ganesan et. al. define Naive T-cells as CD28+CD45RO-CD57-CCR5-CD27+ CCR7+ within the CD3+CD14- compartment [2]. The *OverlapScores* vector has the proportion of Naive T-cells (as defined above) to the total number of cells in any given immunophenotype (a higher value represents a larger overlap):

$$\frac{\sum_{All\ Samples} \frac{Number\ of\ CD28^{+}CD45RO^{-}CD57^{-}CCR5^{-}CD27^{+}CCR7^{+}\ cells}{Number\ of\ cells\ in\ the\ given\ population}}{(Number\ of\ Samples)} \tag{1}$$

The names of the vector match the names of the *Signs* matrix.

4.1 Second Example: Optimization against a Clinical Outcome

$KI67^{+}CD4^{-}CCR5^{+}CD127^{-}$ cells in HIV^{+} patients have a negative correlation with protection against HIV [1]. The P-value assigned to the immunophenotype confirms this:

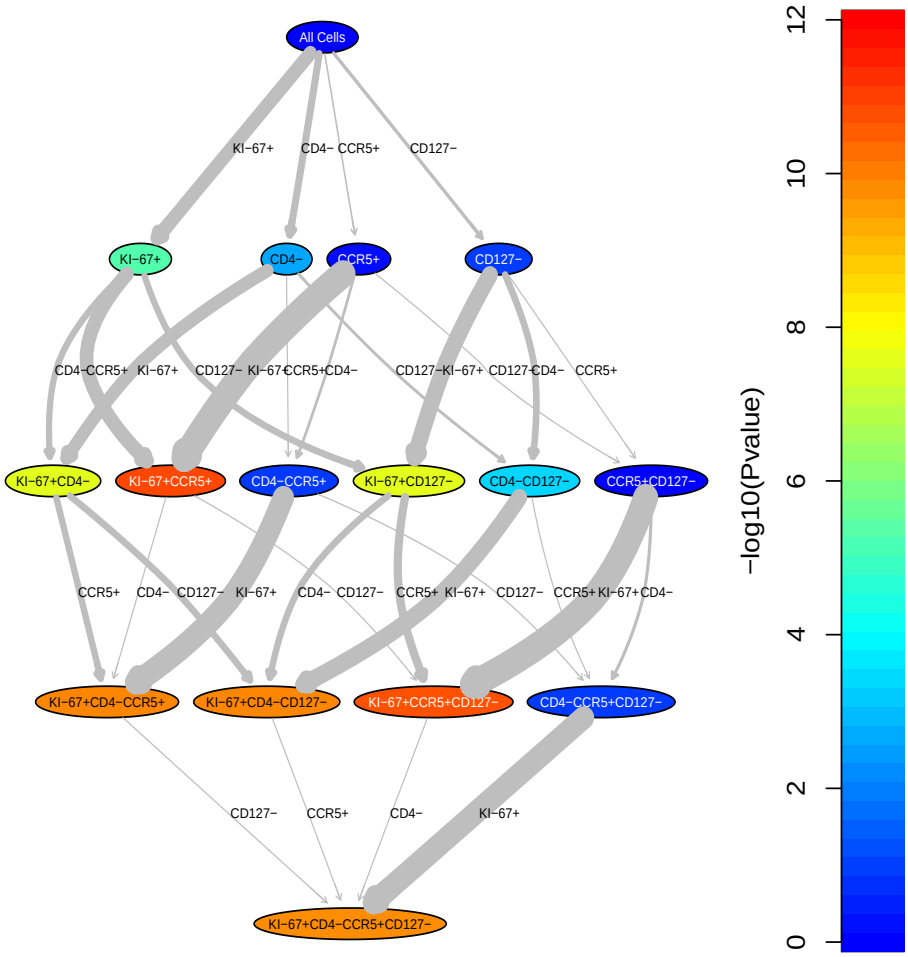
```
> LogRankPvals['KI-67+CD4-CCR5+CD127-']

KI-67+CD4-CCR5+CD127-
         1.702094e-10
```

We first need to calculate the base-3 code of the immunophenotype using the Signs matrix:

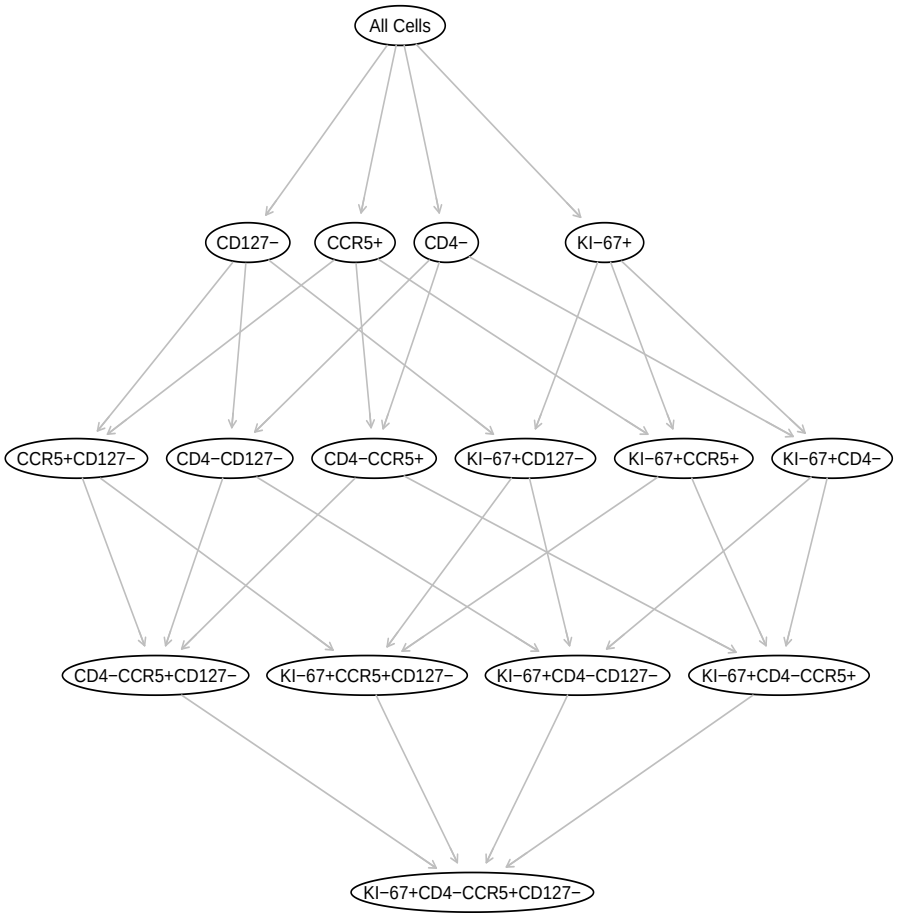
```
> paste(Signs['KI-67+CD4-CCR5+CD127-',], collapse='')

[1] "2111012110"
```

Since 4 markers are involved in this immunophenotype, there are $4! = 24$ paths to reach this population. RchyOptimyx can calculate and visualize all of these paths, alongside each intermediary population's predictive power:

```
> par(mar=c(1,4,1,1))
> res<-RchyOptimyx(Signs, -log10(LogRankPvals),
+                  '2111012110', 24,FALSE)
> plot(res, phenotypeScores=-log10(LogRankPvals),
+      ylab='-log10(Pvalue)')
```



The width of the edges demonstrates the amount of predictive power gained by moving from one node to another. The color of the nodes demonstrates the predictive power of the cell population. Node colors, edge weights, and node/edge labels can be removed from the graph as desired using the parameters of the plot



function:

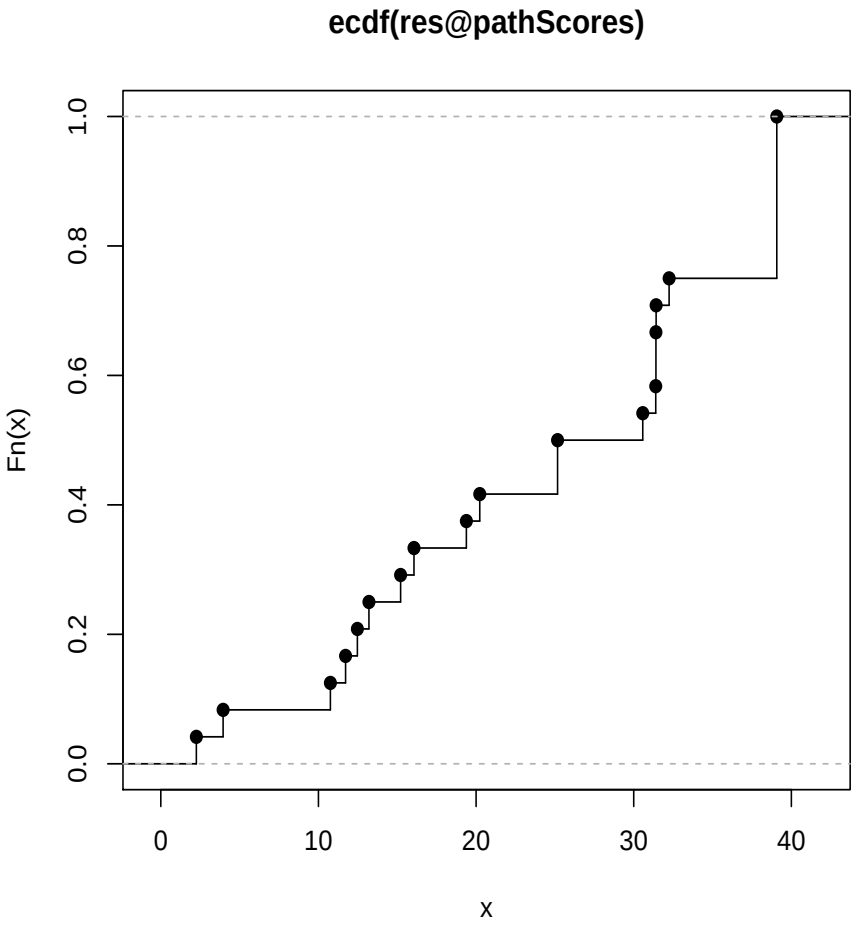
res is an *OptimizedHierarchy* object:

```
> summary(res)
```

```
An optimized hierarchy with 16 nodes, 32 edges, and 24 paths
```

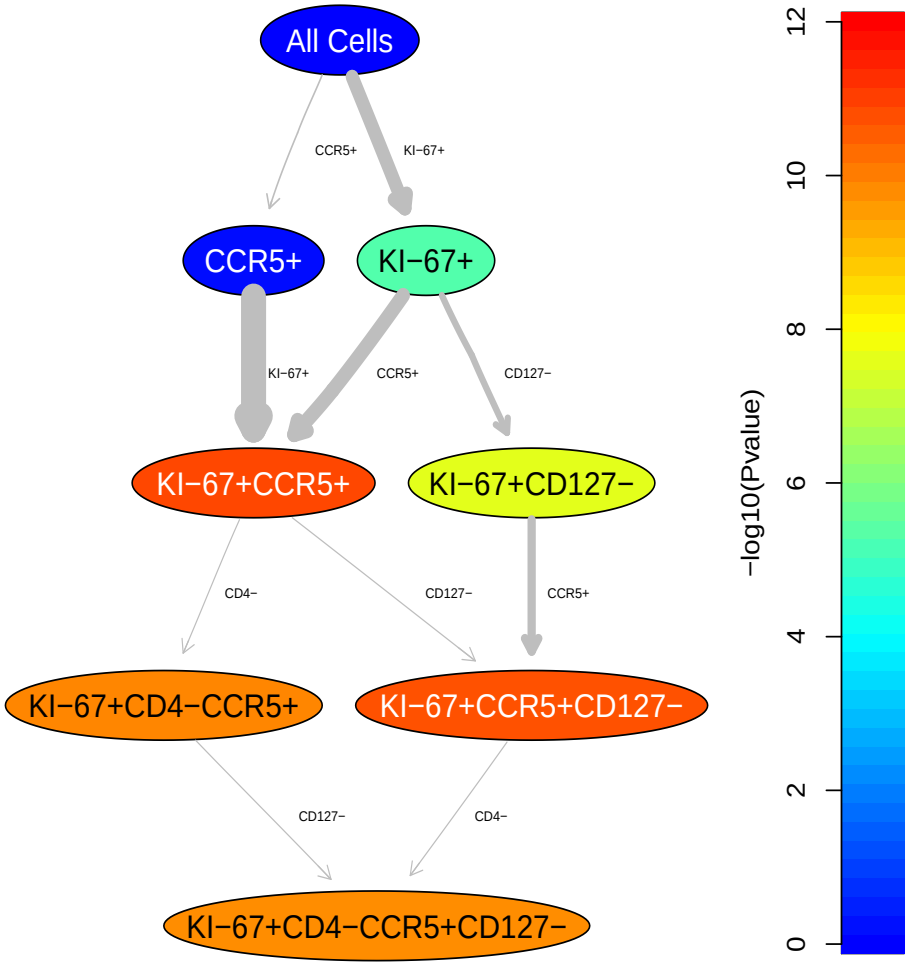
This object stores the scores assigned to everyone of the calculated paths:

```
> plot(ecdf(res@pathScores), verticals=TRUE)
```



We can re-run RchyOptimyx to limit the hierarchy to the top 4 paths:

```
> par(mar=c(1,4,1,1))
> res<-RchyOptimyx(Signs, -log10(LogRankPvals), '2111012110', 4,FALSE)
> plot(res, phenotypeScores=-log10(LogRankPvals), ylab='-log10(Pvalue)')
```



This suggests that the KI-67$^+$CCR5$^+$ cell population is also correlated with protection against HIV but uses only 2 markers. This can be confirmed by the log-rank test P-value:

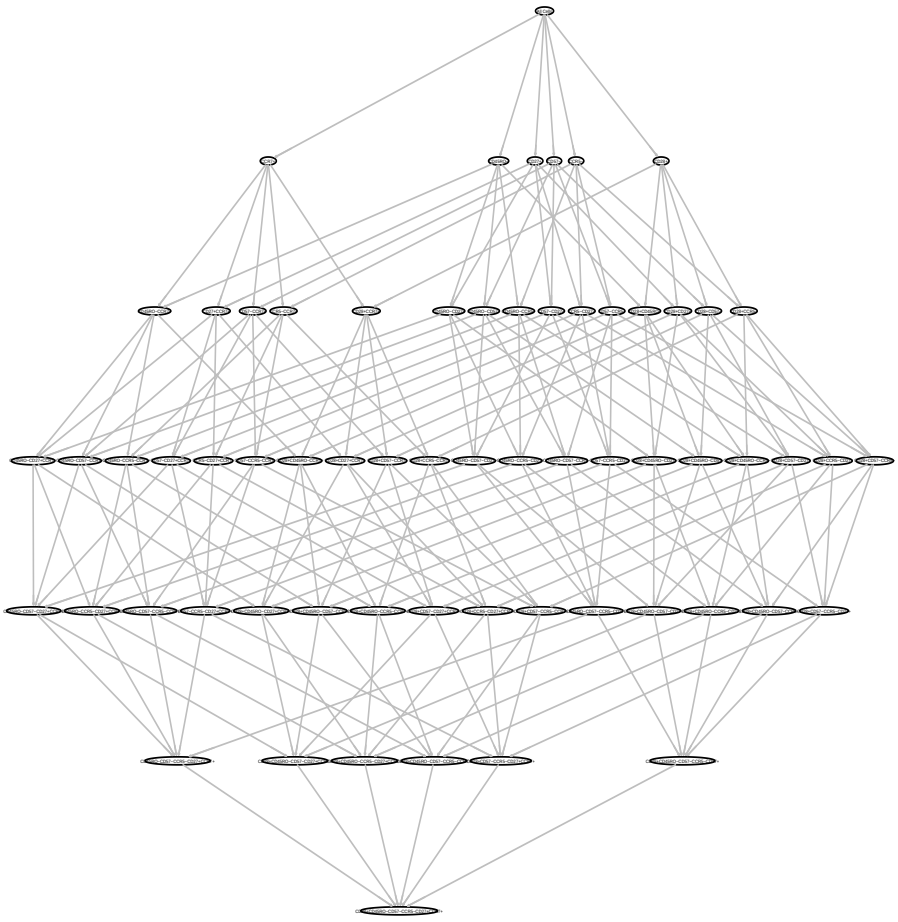
```
> LogRankPvals['KI-67+CCR5+']
```

```
 KI-67+CCR5+
1.317502e-11
```

4.2 Third Example: Optimization against Event Overlap

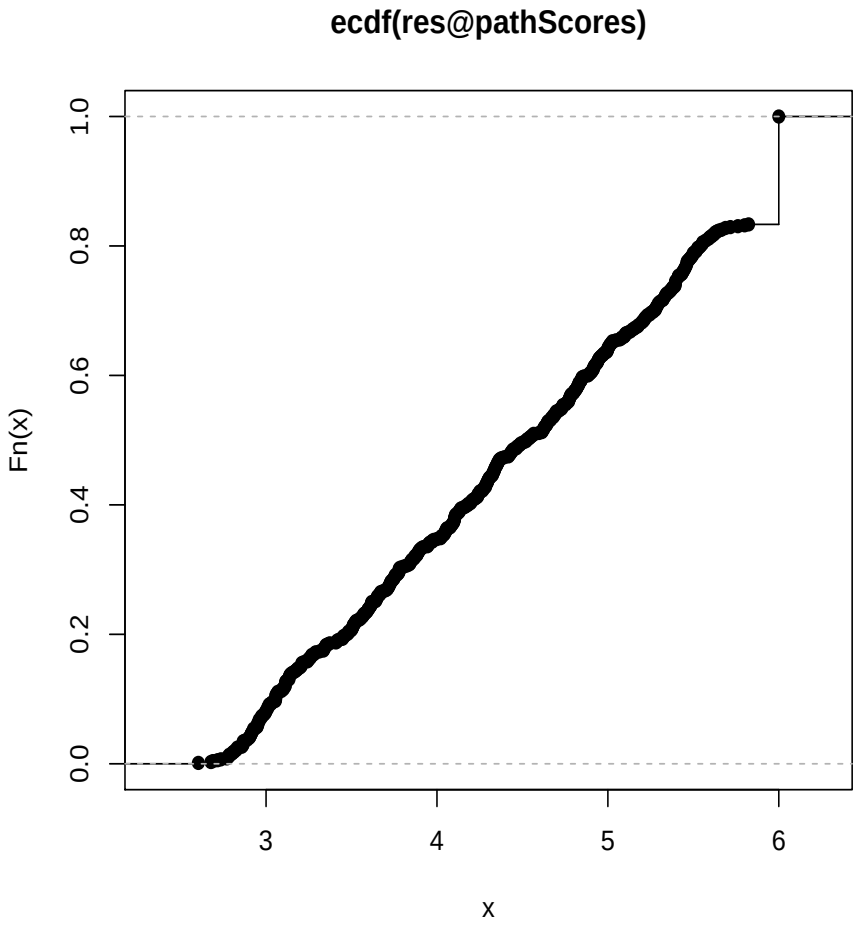
Ganesan et. al. use a strict but potentially redundant definition for naive T-cells, of CD28$^+$ CD45RO$^-$ CD57$^-$ CCR5$^-$ CD27$^+$ CCR7$^+$ within the CD3$^+$CD14$^-$ compartment [2]. Since 6 markers are involved, 720 paths can exist:

```
> res<-RchyOptimyx(Signs, OverlapScores,
+                  paste(Signs['CD28+CD45RO-CD57-CCR5-CD27+CCR7+',],
+                        collapse=''), 720, FALSE)
> par(mar=c(1,4,1,1))
> plot(res, phenotypeScores=OverlapScores, ylab='Purity',
+      uniformColors=TRUE, edgeWeights=FALSE, edgeLabels=FALSE,
+      nodeLabels=TRUE)
```



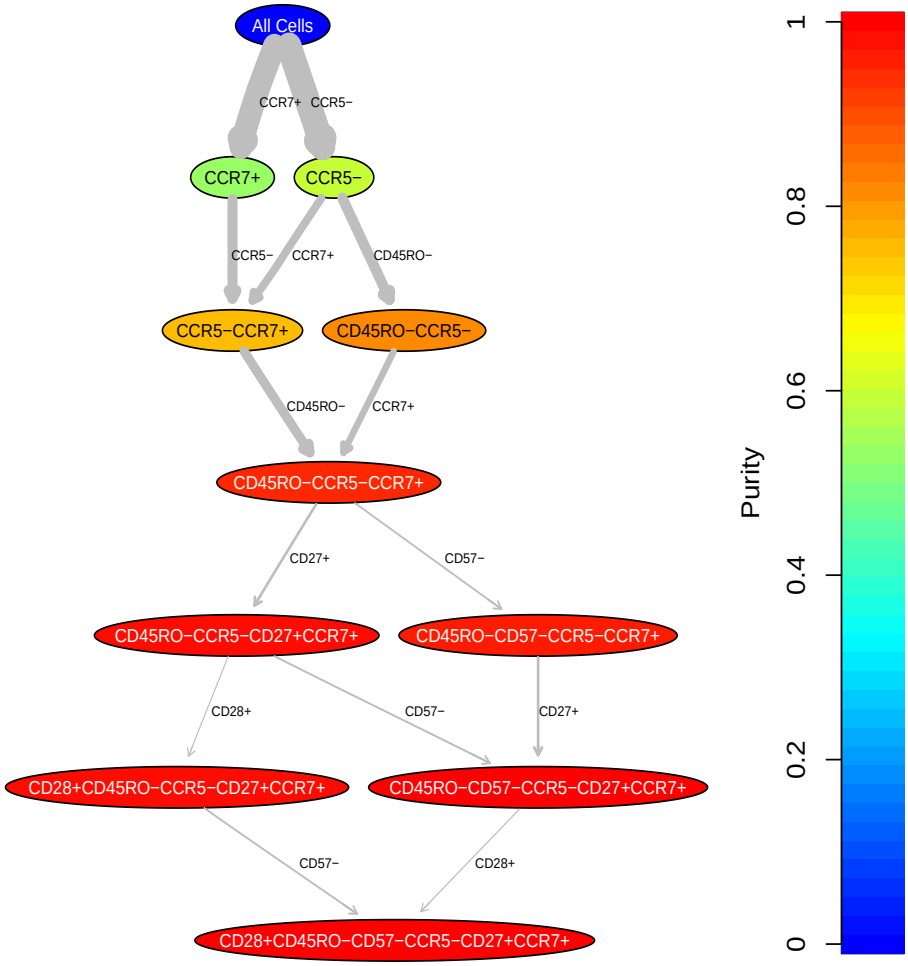
Here is the distribution of these paths:

```
> plot(ecdf(res@pathScores), verticals=TRUE)
```



And a cellular hierarchy with the top 5 paths:

```
> res<-RchyOptimyx(Signs, OverlapScores,
+                  paste(Signs['CD28+CD45RO-CD57-CCR5-CD27+CCR7+',],
+                        collapse=''), 5, FALSE)
> par(mar=c(1,4,1,1))
> plot(res, phenotypeScores=OverlapScores, ylab='Purity')
```



This shows that a 95% pure population of strictly naive T cells can be identified using only 3 markers ($CD45RO^{-}CCR5^{-}CCR7^{+}$).

```
> OverlapScores['CD45RO-CCR5-CCR7+']
```

```
CD45RO-CCR5-CCR7+
        0.9489143
```

References

[1] N. Aghaeepour, P. K. Chattopadhyay, A. Ganesan, K. O'Neill, H. Zare, A. Jalali, H. H. Hoos, M. Roederer, and R. R. Brinkman. Early Immunologic Correlates of HIV Protection can be Identified from Computational Analysis

of Complex Multivariate T-cell Flow Cytometry Assays. *Bioinformatics*, Feb 2012.

[2] A. Ganesan, P.K. Chattopadhyay, T.M. Brodie, J. Qin, W. Gu, J.R. Mascola, N.L. Michael, D.A. Follmann, and M. Roederer. Immunologic and virologic events in early hiv infection predict subsequent rate of progression. *Journal of Infectious Diseases*, 201(2):272, 2010.