User guide for *ISOP* package version 0.99.1

Trung Nghia Vu

February 24, 2017

1 Introduction

RNA-sequencing of single-cells enables characterization of transcriptional heterogeneity in seemingly homogenous cell populations. Single-cell gene-level expression variability has been characterized by RNA-sequencing in multitudes of biological context to date, but few studies have focused on heterogeneity at isoform-level expression. We propose a novel method ISOform-Patterns (ISOP) [1], based on mixture modeling, to characterize the expression patterns of pairs of isoform from the same genes and determine if isoform-level expression patterns are random or signify biological effects. The method allows to investigate single-cell isoform preference and commitment, and assess heterogeneity on the level of isoform expression. It also provides a novel way to assess biological effects in single-cell RNA-seq data through the isoform patterns, then discover differential-pattern genes (DP genes).

In this document, we introduce practical uses of the ISOP method for analyzing isoform patterns and discovering differential-pattern genes.

2 Detect isoform patterns

In this section, we use a small example dataset (read count) to show a practical use of ISOP for

- modelling expression differences of pairs of isoforms by the Gaussian mixture model
- detecting principal isoform patterns from the isoform pairs

```
#Load libaries
library(ISOP)
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
library(AnnotationDbi)
```

We prepare reference annotation (txdb) and load data for the experiment

```
#Load libaries
library(ISOP)
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
library(AnnotationDbi)
#The reference annotation (txdb) used here is loaded from UCSC hg19.
#However, to your real dataset, the txdb also can be imported from
#a gtf file by using the R package GenomicFeatures
txdb=TxDb.Hsapiens.UCSC.hg19.knownGene
#Load sample data
data(isoformDataSample)
#Preprocessing step: only read counts no less than 3 are consider as expressed
isoformDataSample=ifelse(isoformDataSample <= 3,0,isoformDataSample)
isoformDataSample=isoformDataSample[which(rowSums(isoformDataSample)>0),]
#Tranform the read counts to log scale
```

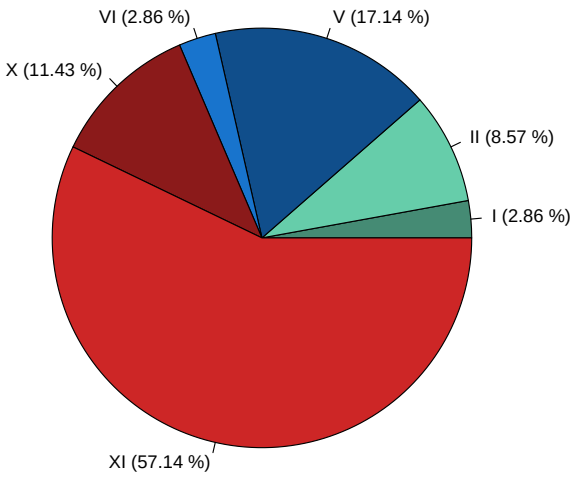
```
isoformDataSample=ifelse(isoformDataSample==0,0,log2(isoformDataSample))
#The data is now ready for ISOP
```

Now, the data and reference annotation are ready. We start to detect isoform patterns.

```
#Define the number of breaks
tbreak=round(sqrt(ncol(isoformDataSample)))
#Do mixture modelling
model.res=doMixtureModelMatrix(isoformLevel.data=isoformDataSample,txdb=txdb,tbreak=tbreak)

## 'select()' returned 1:1 mapping between keys and columns

#Detect patterns
pattern.res=detectPatternType(dmix.list=model.res$dmix.list,
        nq.list=model.res$nq.list,isoformLevel.data=isoformDataSample)
#Display in a pie chart
patternTable=table(pattern.res$pattern.type)
pie(patternTable,col=colors()[c(12,11,132,131,137,136)], labels=paste(names(patternTable),
                " (",round(patternTable/sum(patternTable)*100,2)," %)", sep=""))
```



3 Validate the non-randomness of the patterns

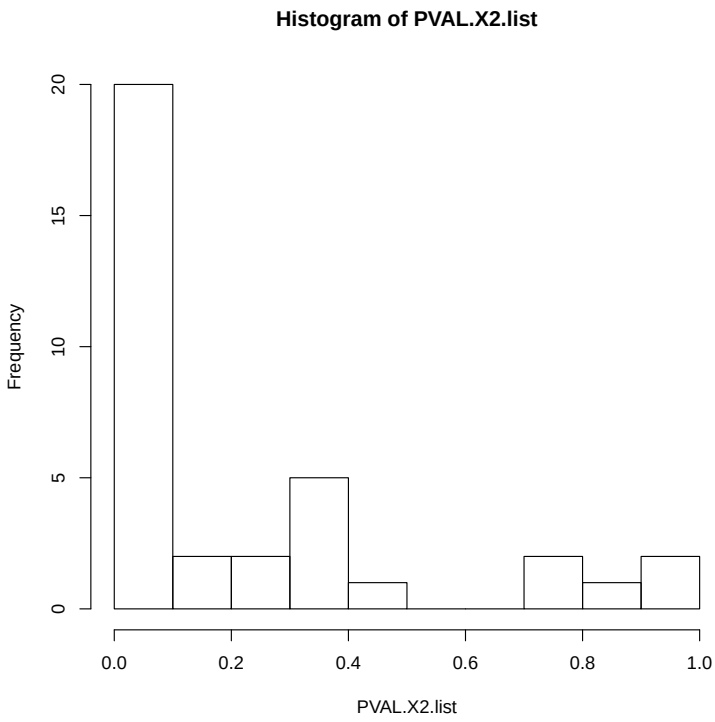
In order to test if isoform pair patterns are significant (non-random) we applied chi-squared goodness-of-fit test combined with a permutation-based approach. It should use 10000 permutations for the test, however due to time limit, we use only 100 permuations in this example.

```
##Register the number of cores for parallel computing if we apply useParallel=TRUE
#library(doParallel)
#registerDoParallel(cores=4)
set.seed(2015)
```

```
iso.pair.names=names(model.res$nq.list)
res=validateIsoformPair(iso.pair.names=iso.pair.names,
   isoformLevel.data=isoformDataSample,per.num=100,tbreak=tbreak,useParallel=FALSE)
#Extract p-values
PVAL.X2.list=unlist(res[names(res)=="pval"])
fdr.val=p.adjust(PVAL.X2.list,method="BH")
#Number of significant isoform patterns
length(which(fdr.val <= 0.05))

## [1] 15

hist(PVAL.X2.list, breaks=10)
```



4 Discover differential-pattern (DP) genes

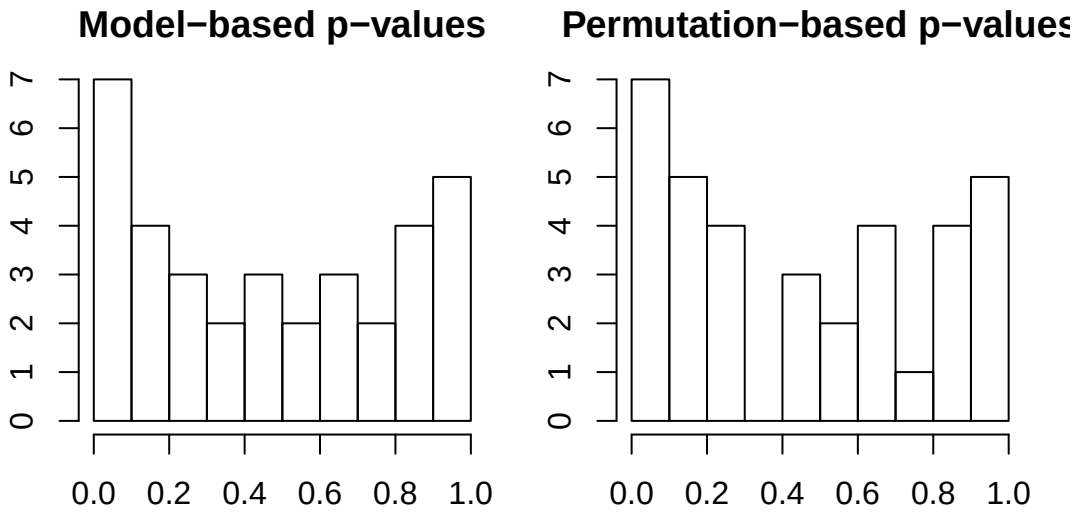
The sample dataset contains cells in treated group and control group that are indicated in column names of the data matrix. Next, we do differential pattern analysis through two functions:

- assignCellComp(): assigning cells to components of the mixture models to create cluster labels using maximum a posteriori probability (MAP) estimation
- getDPIP():computing the association of the cluster labels and the true group labels via Chi-squared test

In this example we also run permutation test for the DP analysis but limit only 100 permutations to keep the running fast.

```
##Register the number of cores for parallel computing if we apply useParallel=TRUE
#library(doParallel)
#registerDoParallel(cores=4)
set.seed(2015)
#Extract group labels of cells
```

```
group.label=unlist(lapply(colnames(isoformDataSample),
                          function(x) unlist(strsplit(x,"_"))[1]))
#Do DP analysis
iso.pair.names=names(model.res$nq.list)
map.res=assignCellComp(iso.pair.names,model.res$dmix.list,
       model.res$nq.list, isoformLevel.data=isoformDataSample)
res=getDPIP(map.res$cellCompMat.prob,group.label,usePermutation=TRUE,
            per.num=100,useParallel=FALSE)
#Extract DP isoform pairs from permutation-based p-values
FDR=p.adjust(res$e.PVAL, method="BH")
dp.pattern.id=which(FDR <= 0.05)
#Plot model-based p-values (res$t.PVAL) and permutation-based p-values (res$e.PVAL)
par(mfrow=c(1,2),mar=c(1,2,2,1)+0.2,oma=c(1,1,1,1));
hist(res$t.PVAL,breaks=10, main = "Model-based p-values")
hist(res$e.PVAL,breaks=10, main = "Permutation-based p-values")
```

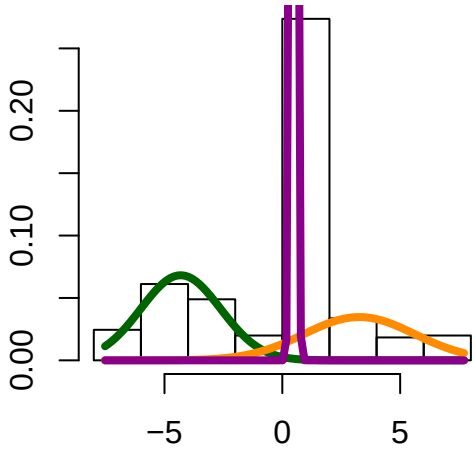


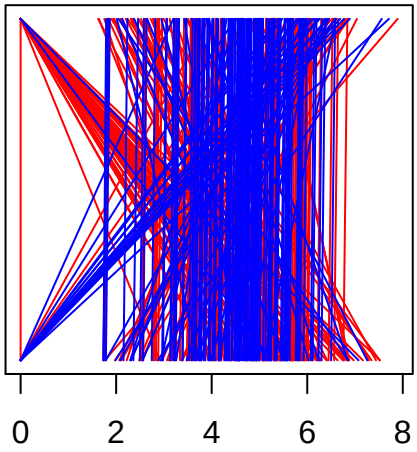
We select a DP isoform pair and draw the mixture model plot and pair-line plot of the pattern.

```
#Select the first DP pattern: dp.pattern.id[1]
pattern.name=unlist(strsplit(names(dp.pattern.id[1]),"\\^"))
#The name of pattern includes: gene name, isoform a and isoform b
cat(pattern.name)

## 128178 uc001hxv.1 uc001hxu.1

#Get expression of isoform a an isoform b
iso_a=isoformDataSample[which(rownames(isoformDataSample)==pattern.name[2]),]
iso_b=isoformDataSample[which(rownames(isoformDataSample)==pattern.name[3]),]
deltaVal=iso_a-iso_b
#Choose the best model of this isoform pair indicated by nq.list in the model.res
compNum=model.res$nq.list[[names(dp.pattern.id)[1]]]
#Extract the mixture model
mydmix=model.res$dmix.list[[names(dp.pattern.id)[1]]][[compNum]]
#Display the model and pair-line plot
par(mfrow=c(1,2),mar=c(1,2,2,1)+0.2,oma=c(1,1,1,1));
plotHistModels(deltaVal,mydmix,plot.title="",fit.line=FALSE,lwd=4)
plotPairFeatures(iso_a,iso_b,group.label=group.label)
```





5 References

1. Vu, Trung Nghia, Quin F. Wills, Krishna R. Kalari, Nifang Niu, Liewei Wang, Yudi Pawitan, and Mattias Rantalainen. "Isoform-Level Gene Expression Patterns in Single-Cell RNA-Sequencing Data." bioRxiv, January 16, 2016, 36988. doi:10.1101/036988.