NoiSeq

NoiSeq is a R Bioconductor package that can be used to analyse count data coming from next-generation sequencing technologies. [Noiseq.pdf] The NOISeq package can be used to Quality control the data, Normalize and filter low count data, and do differential expression analysis.

NOISeq vs NOISeqBio

Why select NOISeq?

\*\*

The NOISeq and NOISeqBIO methods are data-adaptive and nonparametric. Therefore, unlike other packages, no distributional assumptions must be made to analyse differential data.

**Methodology**

1. Input Data

To start the process input data will be read by the readData function and the expression data and the factors are two input parameters to this function. The expression data is provided through a matrix or as a R dataframe which contains all the rows as the number of features that need to be studied and all the columns as the number of experiment samples.

The variables indicating different experimental groups are called factors in each sample. Here, the elements' order for the factor must match the order of the sample columns in the expression data. [Noiseq.pdf]

\*sample factors and expression data\*

We are not provided with additional biological information such as length of features,

GC content, biological feature classification or chromosome position and therefore, in this analysis we are not going to use NOISeqBio.

1. Convert data into NOISeq object

After creating the expression data and factors, these two parameters will be used in the readData function to create the NOISeq object that contains count data ready to be analysed. This readData function returns a Biobase’s eSet class object containing read count information.

1. Quality control of count data

Analyse through visualization plots

Sequencing bias detection

Quality Control report

1. Normalization, Low-count filtering & Batch effect correction

Normalization

There are different normalization techniques implemented in NOISeq and they are RPKM, TMM and Upper Quartile.

Among these TPM (Transcripts Per Million) is considered as more accurate for gene expression comparisons. In this function, it normalizes by the sum of the effective lengths of transcripts instead of the total number of reads.

Low-count filtering

Excluding features with low counts improves, in general, di erential expression results, no matter the method

being used, since noise in the data is reduced. However, the best procedure to  lter these low count features has

not been yet decided nor implemented in the di erential expression packages. NOISeq includes three methods to

 lter out features with low counts:

Batch effect correction

1. Differential Expression

*The NOISeq package computes differential expression between two experimental conditions given the expression level of the considered features.*

*he package includes two non-parametric approaches for di erential expression*

*analysis: NOISeq [1] for technical replicates or no replication at all, and NOISeqBIO [2], which is optimized for the*

*use of biological replicates. Both methods take read counts from RNA-seq as the expression values, in addition*

*to previously normalized data and read counts from other NGS technologies.*

*In the previous section, we described how to use normalization and  ltering functions prior to perform*

*di erential expression analysis. However, when using NOISeq or NOISeqBIO to compute di erential expression,*

*it is not necessary to normalize or  lter low counts before applying these methods because they include these*

*options. Thus, normalization can be done automatically by choosing the corresponding value for the parameter*

*norm. Furthermore, they also accept expression values normalized with other packages or procedures. If the data*

*have been previously normalized, norm parameter must be set to  n . Regarding the low-count  ltering, it is not*

*necessary to  lter in NOISeq method. In contrast, it is recommended to do it in NOISeqBIO, which by default*

*lters out low-count features with CPM method (filter=1).*

NOISeq

*NOISeq method was designed to compute di erential expression on data with technical replicates (NOISeq-real)*

*or no replicates at all (NOISeq-sim). If there are technical replicates available, it summarizes them by summing*

*up them. It is also possible to apply this method on biological replicates, that are averaged instead of summed.*

*However, for biological replicates we strongly recommend NOISeqBIO. NOISeq computes the following di erential*

*expression statistics for each feature: M (which is the log2-ratio of the two conditions) and D (the value of the*

*di erence between conditions). Expression levels equal to 0 are replaced with the given constant k > 0, in order*

*to avoid in nite or undetermined M-values. If k = NULL, the 0 is replaced by the midpoint between 0 and*

*the next non-zero value in the expression matrix. A feature is considered to be di erentially expressed if its*

*corresponding M and D values are likely to be higher than in noise. Noise distribution is obtained by comparing*

*all pairs of replicates within the same condition. The corresponding M and D values are pooled together to*

*generate the distribution. Changes in expression between conditions with the same magnitude than changes*

*in expression between replicates within the same condition should not be considered as di erential expression.*

*Thus, by comparing the (M,D) values of a given feature against the noise distribution, NOISeq obtains the*

*probability of di erential expression  for this feature. If the odds Pr(di erential expression)/Pr(non-di erential*

*expression) are higher than a given threshold, the feature is considered to be di erentially expressed between*

*conditions. For instance, an odds value of 4:1 is equivalent to q = Pr(di erential expression) = 0.8 and it means*

*18*

*that the feature is 4 times more likely to be di erentially expressed than non-di erentially expressed.*

NOISeq parameters

*NOISeq returns an Output object containing the following elements:*

*. comparison: String indicating the two experimental conditions being compared and the sense of the comparison.*

*. factor: String indicating the factor chosen to compute the di erential expression.*

*. k: Value to replace zeros in order to avoid indetermination when computing logarithms.*

*. method: Normalization method chosen.*

Once executing the NOISeq on data using above paramters, the output result gives mean values for each conditions, M, D, prob and ranking values. These M and D values are the differential expression statistics for the NOISeq and prob is the probabaility of differential expression.

*The output myresults@results[[1]]$prob gives the estimated probability of differential expression for each*

*feature.* ***Note that when using NOISeq, these probabilities are not equivalent to p-values****. The higher the*

*probability, the more likely that the difference in expression is due to the change in the experimental condition*

*and not to chance.*

NOISeq uses the probability (*P**NOI*) to identify differentially expressed genes. As mentioned before, we can consider 1-*P**NOI*to be equivalent to *q*-value [[25](https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-14-S13-S7#ref-CR25)- https://pubs.acs.org/doi/epdf/10.1021/pr700739d]. [https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-14-S13-S7 ]

FDR = (Number of False Positives) / (Total Number of Differentially Expressed Genes)

Select differentially expressed features

*Once we have obtained the differential expression probability for each one of the features by using NOISeq or*

*NOISeqBIO function, we may want to select the differentially expressed features for a given threshold q. This can*

*be done with the degenes function on the  output" object using the parameter q. With the argument M we choose if we want all the differentially expressed features, only the differentially expressed features that are more expressed in condition 1 than in condition 2 (M =  up") or only the differentially expressed features that are under-expressed in condition 1 with regard to condition 2 (M =  down"):*

From the NOISeq “M = up” gives the *differentially expressed features that are up expressed in condition 1 than in condition 2. This can be taken similar to the differentially expressed features that are under expressed in condition 2 than in condition 1. So, according to the metadata provided differentially expressed features are given based on condition 2 compared to condition 1. Therefore, the output of the M=up(up expressed genes in condition 1 compared to condition 2) has been assigned to the gene count that is under-expressed in condition 2 compared to 1.*

**3\_500\_500 dataset**

q=0.8

> nrow(DEgenes) #total differentilly expressed genes identified by noiseq

[1] 188

> nrow(detected\_down) #total down regulated differentilly expressed genes identified by noiseq

[1] 83

> nrow(detected\_up) #total up regulated differentilly expressed genes identified by noiseq

[1] 105

> true\_positives

[1] 120

> false\_positives

[1] 68

> true\_negatives

[1] 9799

> false\_negatives

[1] 0

> accuracy

[1] 0.9931911

> precision

[1] 0.6382979

> recall

[1] 1

> f1\_score

[1] 0.7792208

> FDR

[1] 0.3617021

q=0.9

> nrow(DEgenes) #total differentilly expressed genes identified by noiseq

[1] 29

> true\_positives

[1] 16

> false\_positives

[1] 13

> true\_negatives

[1] 9958

> false\_negatives

[1] 0

> accuracy

[1] 0.9986983

> precision

[1] 0.5517241

> recall

[1] 1

> f1\_score

[1] 0.7111111

> FDR

[1] 0.4482759

q=0.7

> nrow(DEgenes) #total differentilly expressed genes identified by noiseq

[1] 458

> true\_positives

[1] 261

> false\_positives

[1] 197

> true\_negatives

[1] 9529

> false\_negatives

[1] 0

> accuracy

[1] 0.9802744

> precision

[1] 0.569869

> recall

[1] 1

> f1\_score

[1] 0.7260083

> FDR

[1] 0.430131

Minimum FDR was given when q=0.8 and therefore 0.8 was selected as the threshold to identify diferentially expressed genes.

**6\_750\_250**

q=0.8

> nrow(DEgenes) #total differentilly expressed genes identified by noiseq

[1] 191

> nrow(detected\_down) #total down regulated differentilly expressed genes identified by noiseq

[1] 45

> nrow(detected\_up) #total up regulated differentilly expressed genes identified by noiseq

[1] 146

> true\_positives

[1] 158

> false\_positives

[1] 33

> true\_negatives

[1] 9806

> false\_negatives

[1] 0

> accuracy

[1] 0.996699

> precision

[1] 0.8272251

> recall

[1] 1

> f1\_score

[1] 0.9054441

> FDR

[1] 0.1727749

**3\_750\_250**

> nrow(DEgenes) #total differentilly expressed genes identified by noiseq

[1] 207

> nrow(detected\_down) #total down regulated differentilly expressed genes identified by noiseq

[1] 62

> nrow(detected\_up) #total up regulated differentilly expressed genes identified by noiseq

[1] 145

> true\_positives

[1] 149

> false\_positives

[1] 58

> true\_negatives

[1] 9782

> false\_negatives

[1] 0

> accuracy

[1] 0.9941936

> precision

[1] 0.7198068

> recall

[1] 1

> f1\_score

[1] 0.8370787

> FDR

[1] 0.2801932

**9\_1000\_0**

> nrow(DEgenes) #total differentilly expressed genes identified by noiseq

[1] 189

> nrow(detected\_down) #total down regulated differentilly expressed genes identified by noiseq

[1] 10

> nrow(detected\_up) #total up regulated differentilly expressed genes identified by noiseq

[1] 179

> true\_positives

[1] 172

> false\_positives

[1] 17

> true\_negatives

[1] 9810

> false\_negatives

[1] 0

> accuracy

[1] 0.9982998

> precision

[1] 0.9100529

> recall

[1] 1

> f1\_score

[1] 0.9529086

> FDR

[1] 0.08994709

**9\_750\_250**

> nrow(DEgenes) #total differentilly expressed genes identified by noiseq

[1] 194

> nrow(detected\_down) #total down regulated differentilly expressed genes identified by noiseq

[1] 42

> nrow(detected\_up) #total up regulated differentilly expressed genes identified by noiseq

[1] 152

> true\_positives

[1] 178

> false\_positives

[1] 16

> true\_negatives

[1] 9805

> false\_negatives

[1] 0

> accuracy

[1] 0.9983998

> precision

[1] 0.9175258

> recall

[1] 1

> f1\_score

[1] 0.9569892

> FDR

[1] 0.08247423