Using the DART package: Denoising Algorithm based on Relevance network Topology

Katherine Lawler, Yan Jiao, Andrew E Teschendorff, Charles Shijie Zheng

October 30, 2018

Contents

1 Introduction

DART (Denoising Algorithm based on Relevance network Topology) estimates an activity score for a pathway or perturbation gene signature (the 'model signature') in independent samples of a gene expression matrix, following a denoising step. The model signature takes the form of a gene (or probe) list with associated weights, typically representing the log fold changes or statistics of differential expression in response to some experimental cue (i.e pathway activation). The denoising step removes prior information that is inconsistent with a training data set. The DART algorithm has been shown to be a robust method for estimating pathway and/or perturbation signature activity in many different cancer gene expression data sets [1,2].

See [1,2] for further details on relevance network construction, consistency score significance estimation and metrics for activation score computation.

We demonstrate the DART functions by estimating the activity of an ERBB2 perturbation signature [3], derived from a cell-line model, in primary breast cancer gene expression data sets. The goal is to show how the in-vitro derived ERBB2 perturbation signature reflecting artificial ERBB2 overexpression can be used to predict the naturally occuring perturbations affecting ERBB2 expression in primary cancers. A relevance correlation network of the model signature genes is trained using the Wang/Minn breast cancer data set [4,5]. It is important that this training data set is fairly large and representative of breast cancer demographics. The relevance network is then pruned to remove the edges (the edges represent statistically significant correlations across the training data), which are inconsistent with the prior information. The pruned network is then used to estimate activity scores in each sample of the training set as well as in the samples of the smaller "Mainz" breast cancer data set [6]. We note that since phenotypic sample labels are not used in the unsupervised inference of the pruned network, that it is valid to obtain activity scores in the training set itself.

```
> library(DART)
```

2 Load example data sets

For this example, the Wang/Minn and Mainz breast cancer data sets are loaded as ExpressionSets from existing Bioconductor Experiment Data packages [7,8]. Both data sets are based on the Affymetrix U133A platform. `Note:` DART does not deal with preprocessing of microarray gene expression data. If necessary, expression data set preprocessing such as normalization within data sets should be dealt with by the user before applying DART.

```
> library(Biobase)
> library(breastCancerVDX) # Wang et al./Minn et al.
> library(breastCancerMAINZ)
> data(vdx)
> data(mainz)
```

An in vitro-derived ERBB2 perturbation signature [3] is provided as example data in the DART package. The perturbation signature is given as a vector of differential gene expression following ERBB2 activation and annotated using Entrez Gene IDs. This model signature contains 431 genes, of which 255 genes are reported to be up-regulated and 176 down-regulated.

```
> data(dataDART)
> modelSig <- dataDART$sign
```

Mapping between probe identifiers and model signature

The DART functions assume that the same type of identifier is used to label gene expression rows and model signature values.

In general, the mappings between gene expression platforms and the model signatures or pathways should be handled by the user (for example, see the biomaRt package [9] for mapping microarray probe identifiers to Ensembl Gene IDs, Entrez Gene IDs, or other gene identifiers).

This example uses Entrez Gene IDs, so we extract a gene expression matrix from each ExpressionSet and annotate the rows using Entrez Gene IDs. The mapping between Affymetrix probe IDs (gene expression data) and Entrez Gene IDs (model signature) is handled using the Affymetrix U133A probeset annotation provided in the loaded ExpressionSet objects.

Example: Constructing a gene expression matrix

For each of the Wang/Minn and Mainz data sets, we retrieve a gene expression matrix from the ExpressionSet and label the rows using Entrez Gene IDs. Where multiple probesets map to one Entrez Gene ID, we retain only the most variable probeset.

```
> # Order vdx by std dev across samples, decreasing
> vdx.ord <- vdx[order(  apply(exprs(vdx), 1, sd,na.rm=T), decreasing=T ), ]
> # Reduce vdx to retain a single (most variable) probeset per Entrez Gene ID
> vdx.EntrezUniq <- vdx.ord[
+         match(unique(fData(vdx.ord)$EntrezGene.ID), fData(vdx.ord)$EntrezGene.ID),]
> # Get vdx data, labelled by Entrez Gene IDs
> vdx.data <- exprs(vdx.EntrezUniq)
> rownames(vdx.data) <- fData(vdx.EntrezUniq)$EntrezGene.ID[
+         match(rownames(vdx.data), rownames(fData(vdx.EntrezUniq))) ]
> colnames(vdx.data) <- sub('.CEL.gz$','',pData(vdx.EntrezUniq)$filename)
> #
> # Do the same for the Mainz data set
> mainz.ord <- mainz[order(  apply(exprs(mainz), 1, sd,na.rm=T), decreasing=T ), ]
> mainz.EntrezUniq <- mainz.ord[
+         match(unique(fData(mainz.ord)$EntrezGene.ID),fData(mainz.ord)$EntrezGene.ID),]
> mainz.data <- exprs(mainz.EntrezUniq)
> rownames(mainz.data) <- fData(mainz.EntrezUniq)$EntrezGene.ID[
+         match(rownames(mainz.data),rownames(fData(mainz.EntrezUniq))) ]
> colnames(mainz.data) <- sub('.CEL.gz$','',pData(mainz.EntrezUniq)$filename)
> #
> # Inspect the gene expression matrices
> dim(vdx.data)

[1] 13092   344

> dim(mainz.data)

[1] 13092   200
```

3 Relevance network and consistency score

A relevance network is constructed using the model signature genes as nodes. A pair of nodes is connected if the correlation/anti-correlation of the nodes is significant in the training data set. Significance is defined using a user-specified false discovery rate threshold (default value is 0.000001). This is stringent, but reflects a conservative Bonferroni threshold: since typical

model signatures consist on the order of 100 genes, this means estimation on the order of 10000 pairwise correlations, so a Bonferroni threshold would approximately be around 1e-6.

```
> rn.o <- BuildRN(vdx.data, modelSig, fdr=0.000001)

[1] "Found 97% of signature genes in data matrix"

> # How many nodes in the relevance network?
> print(dim(rn.o$adj))

[1] 420 420

> # How many edges in the relevance network? Look at the adjacency matrix.
> print(sum(rn.o$adj == 1)/2)

[1] 6400

> # The nodes in the relevance network are the genes in the model signature
> # which were found in the data matrix
> print(length(rn.o$rep.idx))

[1] 420
```

Reducing the false discovery rate even further to 0.0000001 results in a relevance network with reduced connectivity:

```
> rn.smallerFDR.o <- BuildRN(vdx.data, modelSig, fdr=0.0000001)

[1] "Found 97% of signature genes in data matrix"

> print(sum(rn.smallerFDR.o$adj == 1)/2)

[1] 4975
```

Before performing the pruning step, we check whether the relevance network is significantly consistent with the model signature when compared with randomly permuted node-node correlations. `Note:` If the consistency score is not significant compared to the random permutations, then this indicates that the directional expression changes encoded in the perturbation signature do not account for expression variation of these genes in the data set. Therefore, if the consistency score is not significant it is not recommended to use the signature to predict pathway activity.

```
> ### Evaluate Consistency
> evalNet.o <- EvalConsNet(rn.o);
> print(evalNet.o$netcons['fconsE'])

   fconsE
0.7190625
```

```
> ### The consistency score (i.e fraction of consistent edges) is 0.72.
>
> print(evalNet.o$netcons)

          nG            nE            fE        fconsE Pval(consist)
 4.200000e+02  6.400000e+03  7.273554e-02  7.190625e-01  0.000000e+00

> ### The p-value of the consistency score is significant, so proceed.
> ### Note that P-values may appear as zero because of the finite number
> ### of randomisations performed.
```

4 Pruning the relevance network

The pruning step removes edges which are inconsistent with the prior information contained in the model signature vector. An edge is removed from the relevance network if the nodes are correlated in the training data set but have opposite signs in the model signature, or the nodes are anti-correlated in the training set but have the same sign in the model signature vector.

```
> ### Prune (i.e. denoise) the network
> prNet.o <- PruneNet(evalNet.o)
> ### Print dimension of the maximally connected pruned network
> print(dim(prNet.o$pradjMC))

[1] 347 347

> ### Print number of edges in maximally connected pruned network
> print(sum(prNet.o$pradjMC)/2)

[1] 4601
```

5 Activity score prediction

Finally, an activity score for the model signature is estimated for each sample in the data set.

```
> ### Infer signature activation in the original data set
> pred.o <- PredActScore(prNet.o, vdx.data)

[1] "Found 100% of maximally connected pruned network genes in the data"
```

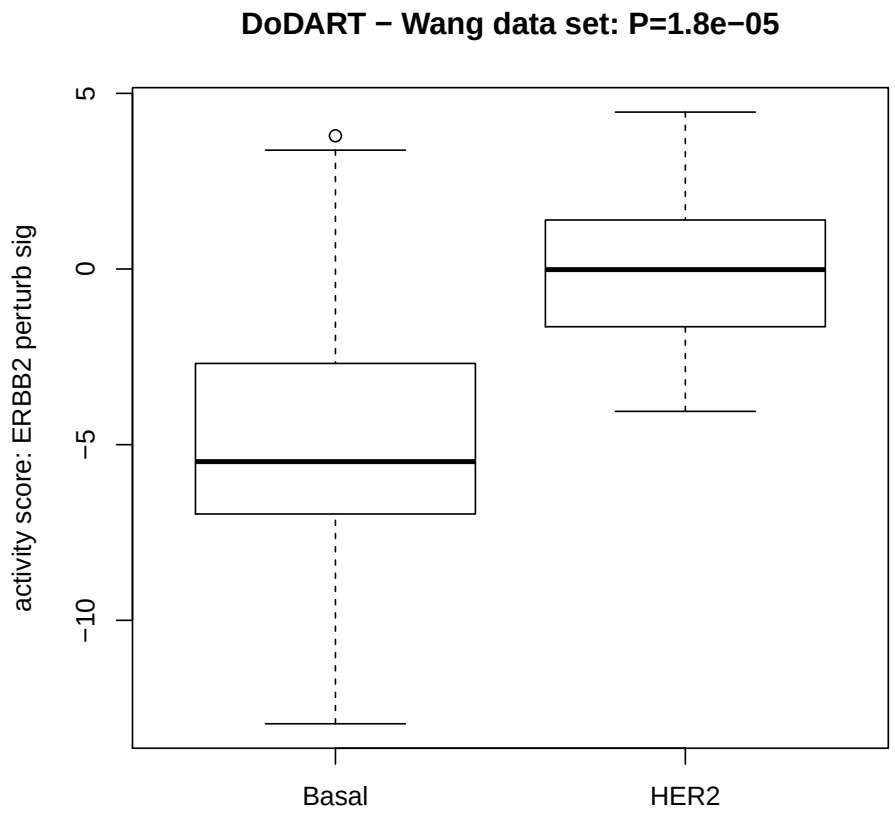
The activity score on a relevance network of N genes is defined for each sample (data set column) as

$$\frac{1}{\sqrt{\sum_{i \in N} k_i^2}} \sum_{i \in N} \sigma_i k_i \vec{z}_i$$

where k_i is the number of neighbours of gene i in the network, $\vec{z}$ is the row-standardized expression vector for the sample, and σ_i is ± 1 according to whether gene i is up- or down-regulated (this information comes from the model signature).

Comparing the estimated activation scores to predicted intrinsic breast cancer subtypes for the Wang et al. subset of samples shows a significant difference in ERBB2 permutation signature activation between basal-like and HER2 breast cancer subtypes. (Intrinsic subtype predictions were precomputed for this example using an intrinsic subtype classifier [10].)

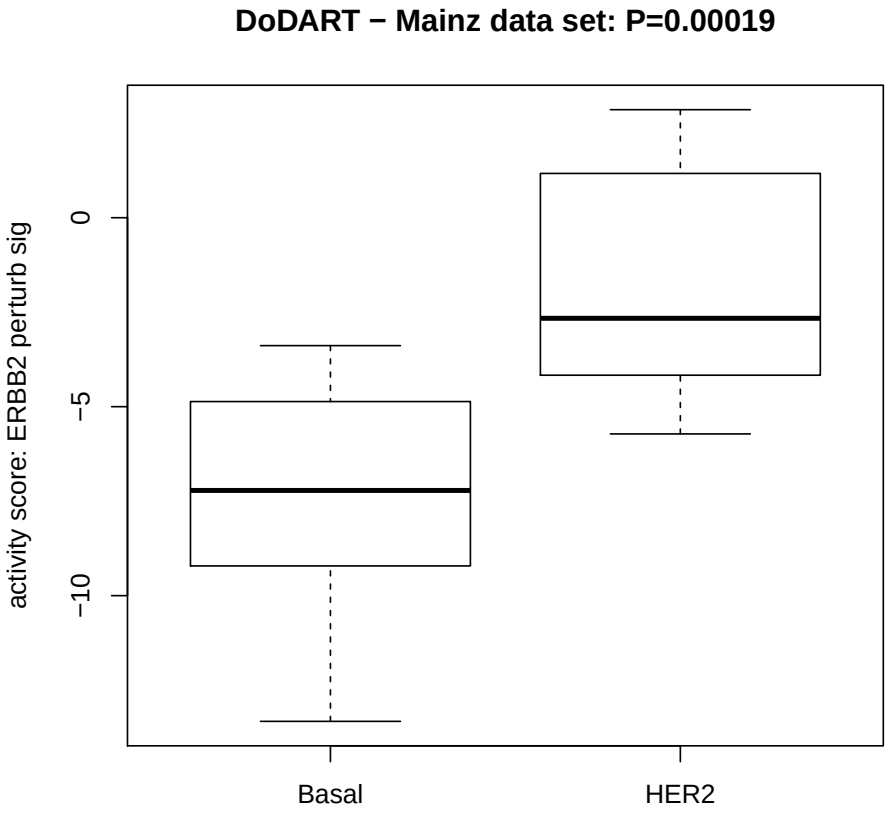
```
> ### Check that activation is higher in HER2+ compared to basals
> pred.o.score.report <- pred.o$score[match(names(dataDART$pheno),names(pred.o$score))]
> boxplot(pred.o.score.report ~ dataDART$pheno,ylab='activity score: ERBB2 perturb sig')
> pv <- wilcox.test(pred.o.score.report ~ dataDART$pheno)$p.value
> title(main=paste("DoDART - Wang data set: P=",signif(pv,2),sep=""))
```



6 Inferring signature activation in an independent data set

The pruned relevance network can be used to estimate robust model signature activity in other breast cancer data sets. Note that for the learning of the pruned network a relatively large data set is necessary, ideally one that represents a random sampling of the underlying disease demographics. Having inferred the pruned network in this large training set, predicting pathway activity in independent samples is then done for each sample separately.

```
> pred.mainz.o <- PredActScore(prNet.o, mainz.data)
[1] "Found 100% of maximally connected pruned network genes in the data"
> # Check that activation is higher in HER2+ compared to basals
> # in the smaller Mainz data set, after training on the larger Wang/Minn data set
>
> pred.mainz.o.score.report <- pred.mainz.o$score[
+     match(names(dataDART$phenoMAINZ),names(pred.mainz.o$score))]
> boxplot(pred.mainz.o.score.report ~ dataDART$phenoMAINZ,
+         ylab='activity score: ERBB2 perturb sig')
> pv <- wilcox.test(pred.mainz.o.score.report ~ dataDART$phenoMAINZ)$p.value
> title(main=paste("DoDART - Mainz data set: P=",signif(pv,2),sep=""))
```



7 DoDART function

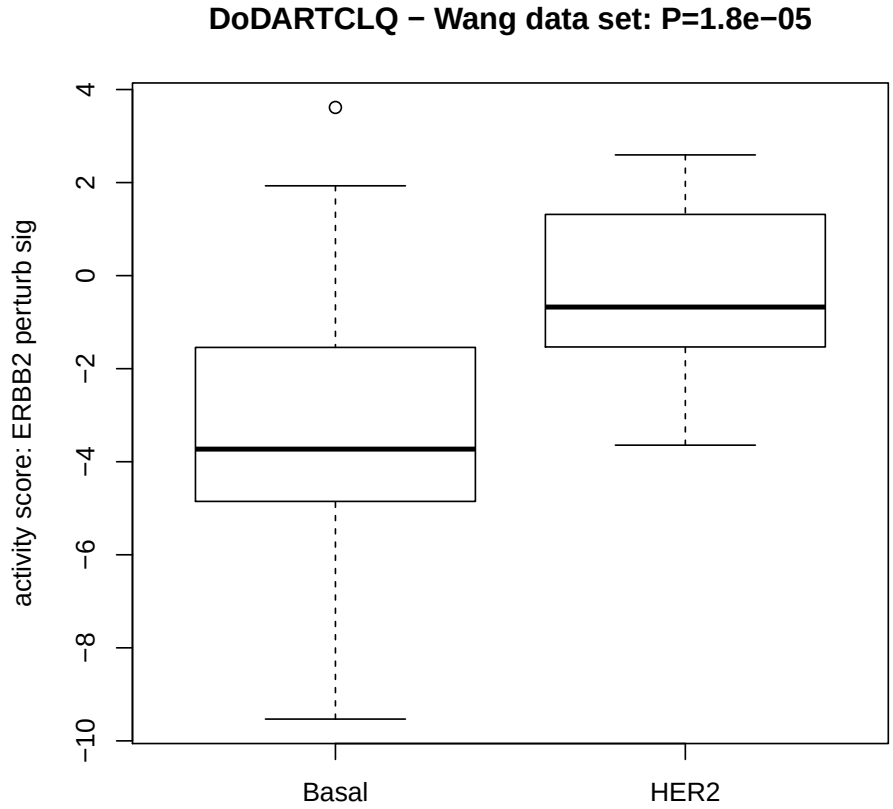
The wrapper function `DoDART` builds a relevance network, evaluates the consistency of correlations compared to the model signature, removes noise from the network (pruning step), and estimates an activity score for each sample in the data set [2].

```
> res.vdx <- DoDART(vdx.data, modelSig, fdr=0.000001)

[1] "Found 97% of signature genes in data matrix"
[1] "Found 100% of maximally connected pruned network genes in the data"

> # View the activity scores for the first five Wang/Minn samples
> print(res.vdx$score[1:5])

  GSM36793   GSM36796   GSM36797   GSM36798   GSM36800
 3.1162895  0.8782771 -6.9537040 -4.0296276  3.7383374
```

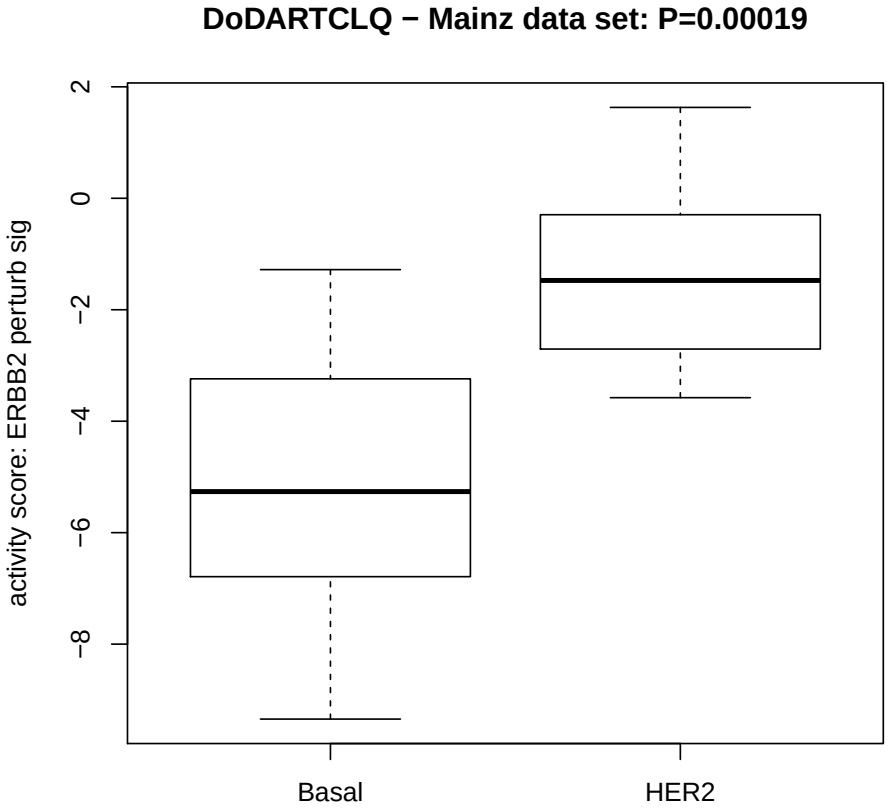
8 Enhanced DoDART function - DoDARTCLQ

The enhanced wrapper function `DoDARTCLQ` builds on `DoDART`. `DoDARTCLQ` uses a smaller and more compact network to estimate pathway/perturbation activity compared to `DoDART`. Whereas `DoDART` uses the whole pruned correlation network, `DoDARTCLQ` infers all maximal cliques within the pruned correlation network and then estimates activity using only genes in the union set of these maximal cliques. Although the largest cliques may not be unique, they typically exhibit very strong overlaps, justifying the use of the merged or union set.

```
> res2.vdx <- DoDARTCLQ(vdx.data, modelSig, fdr=0.000001)

[1] "Found 97% of signature genes in data matrix"
[1] "Found 100% of maximally connected pruned network genes in the data"

> # View the activity scores generated by DoDARTCLQ for the first five Wang/Minn samples
> print(res2.vdx$pred[1:5])

   GSM36793    GSM36796    GSM36797    GSM36798    GSM36800
 2.18991558 -0.01577221 -4.71891616 -1.49101043  1.12086416

> ### Check that activation generated by DoDARTCLQ is higher in HER2+ compared to basals
> pred.o.score.report2 <-res2.vdx$pred[match(names(dataDART$pheno),names(pred.o$score))]
> boxplot(pred.o.score.report2 ~ dataDART$pheno,ylab='activity score: ERBB2 perturb sig')
> pv <- wilcox.test(pred.o.score.report ~ dataDART$pheno)$p.value
> title(main=paste("DoDARTCLQ - Wang data set: P=",signif(pv,2),sep=""))
```




```
> # Check that activation is higher in HER2+ compared to basals
> # in the smaller Mainz data set, after training on the larger Wang/Minn data
> # set by DoDARTCLQ
>
> pred.mainz2.o <- PredActScore(res2.vdx$clq, mainz.data)

[1] "Found 100% of maximally connected pruned network genes in the data"

> pred.mainz.o.score.report2 <- pred.mainz2.o$score[
+     match(names(dataDART$phenoMAINZ),names(pred.mainz.o$score))]
> boxplot(pred.mainz.o.score.report2 ~ dataDART$phenoMAINZ,
+         ylab='activity score: ERBB2 perturb sig')
> pv <- wilcox.test(pred.mainz.o.score.report ~ dataDART$phenoMAINZ)$p.value
> title(main=paste("DoDARTCLQ - Mainz data set: P=",signif(pv,2),sep=""))
```



9 Citing the DART package

To cite the DART package and methods:

Jiao Y, Lawler K, Patel GS, Purushotham A, Jones AF et al. (2011) DART: Denoising Algorithm based on Relevance network Topology improves molecular pathway activity inference. BMC Bioinformatics, 12:403.

Teschendorff AE, Gomez S, Arenas A, El-Ashry D, Schmidt M, et al. (2010) Improved prognostic classification of breast cancer defined by antagonistic activation patterns of immune response pathway modules. BMC Cancer 10:604.

Teschendorff AE, Li L, Yang Z. (2015) Denoising perturbation signatures reveals an actionable AKT-signaling gene module underlying a poor clinical outcome in endocrine treated ER+ breast cancer. Genome Biology 16:61.

10 References

1. Jiao, Y. et al. DART: Denoising Algorithm based on Relevance network Topology improves molecular pathway activity inference. BMC Bioinformatics 12, 403 (2011).

2. Teschendorff, AE. et al. Denoising perturbation signatures reveals an actionable AKT-signaling gene module underlying a poor clinical outcome in endocrine treated ER+ breast cancer. Genome Biology 16:61 (2015).

3. Creighton, C.J. et al. Activation of mitogen-activated protein kinase in estrogen receptor alpha-positive breast cancer cells in vitro induces an in vivo molecular phenotype of estrogen receptor alpha-negative human breast tumors. Cancer Research 66, 3903-11 (2006).

4. Wang, Y. et al. Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. Lancet 365, 671-9 (2005).

5. Minn, A.J. et al. Lung metastasis genes couple breast tumor size and metastatic spread. PNAS 104, 6740-5 (2007).

6. Schmidt, M. et al. The humoral immune system has a key prognostic impact in node-negative breast cancer. Cancer Research 68, 5405-13 (2008).

7. Schroeder, M. et al. breastCancerVDX: Gene expression datasets published by Wang et al. [2005] and Minn et al. [2007] (VDX). (2011). <http://compbio.dfci.harvard.edu/>

8. Schroeder, M. et al. breastCancerMAINZ: Gene expression dataset published by Schmidt et al. [2008] (MAINZ). (2011). <http://compbio.dfci.harvard.edu/>

9. Durinck, S., Huber, W. biomaRt: Interface to BioMart databases (e.g. Ensembl, COSMIC, Wormbase and Gramene).

10. Hu, Z. et al. The molecular portraits of breast tumors are conserved across microarray platforms. BMC Genomics 7, 96 (2006).

11 Session information

```
> print(sessionInfo())

R version 3.5.1 Patched (2018-07-12 r74967)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 16.04.5 LTS

Matrix products: default
BLAS: /home/biocbuild/bbs-3.8-bioc/R/lib/libRblas.so
LAPACK: /home/biocbuild/bbs-3.8-bioc/R/lib/libRlapack.so

locale:
```

```
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=en_US.UTF-8        LC_COLLATE=C
 [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] parallel  stats     graphics  grDevices utils     datasets  methods
[8] base

other attached packages:
[1] breastCancerMAINZ_1.19.0 breastCancerVDX_1.19.0   Biobase_2.42.0
[4] BiocGenerics_0.28.0      DART_1.30.0              igraph_1.2.2

loaded via a namespace (and not attached):
[1] compiler_3.5.1  magrittr_1.5    tools_3.5.1     pkgconfig_2.0.2
```