intcomp: Benchmarking pipeline for integrative cancer gene prioritization algorithms based on gene expression and copy number data

Leo Lahti*and Martin Schäfer

May 23, 2012

1 Introduction

Various algorithms have been suggested to discover cancer-associated genes and chromosomal regions based on integrative analysis of genome-wide gene expression and DNA copy number profiling data. However, quantitative comparison of these models has been missing. The *intcomp* R package provides a benchmarking pipeline for quantitative comparisons of the alternative methods.

This vignette is aimed to provide installation instructions, examples and algorithmic details of the *intcomp* benchmarking pipeline used in [1]. Each algorithm is used to prioritize the gene list, and the resulting order is compared to golden standard lists of known cancer genes on simulated and real data sets. For details, see [1].

2 Installation

2.1 Dependencies

You need to check dependencies are installed before the *intcomp* package can be installed. The benchmarking pipeline depends on various external R packages. Install the dependencies from within R:

```
> source("http://www.bioconductor.org/biocLite.R")
> biocLite(c("biomaRt", "DNAcopy", "DRI", "hgu133ahsentrezg.db",
+            "org.Hs.eg.db", "pint", "PMA", "SIM", "PREDA",
+            "CGHcall"))
> install.packages("XML")
> install.packages("RCurl")
```

*leo.lahti@iki.fi

You may need to install the following packages manually: curl library[1] and the R packages XML[2], RCurl[3], edira and ediraAMLdata[4], intCNGEan[5], and CNAmet[6].

2.2 Installing the intcomp benchmarking pipeline

After installing the dependencies, download the source package from http://intcomp.r-forge.r-project.org/archive/intcomp_latest.tar.gz

To install this package directly from within R:

```
> install.packages("intcomp_latest.tar.gz", type = "source")
```

Let us know if you encounter problems in the installation.

3 Using the benchmarking pipeline

The minimal input data for the test.geneorder.pipeline banchmarking function includes (i) gene expression data (ge), (ii) gene copy number data (cn.raw / cn.seg / cn.call / cghCall), (iii) a golden standard list of known cancer genes (cancerGenes), and (iv) the list of methods to compare (mehods). For practical examples, see below.

The gene expression and copy number data sets are lists containing *data* and *info* fields; the probes in gene expression and gene copy number need to be matched; *data* is a data matrix with gene expression (GE$data) or gene copy number (CN$data) data; *info* field is a data frame containing additional information about genes: *loc* indicates the genomic location of the probes in base pairs (numeric); *chr* and *arm* are factors indicating the chromosome and chromosomal arm of the probe, respectively. The user can provide the copy number data as raw (cn.raw), segmented (cn.seg) or called (cn.call) version. Certain methods require specific versions of the copy number. For instance, the CNAmet requires called copy number data. The intCNGEan algorithm requires copy number as a cghCall object from the *CGHcall* R package. It is advisable to provide all four versions - cn.raw, cn.seg, cn.call and cghCall - in the input to the test.geneorder.pipeline function when possible. The cn.raw, cn.seg and cn.call should follow the data + info format explained above, and the cghCall contains the raw, segmented and called data in the cghCall format. Finally, if multiple versions of copy number data are available, the user can specify (through the cn.default argument) which version is coupled with gene expression data unless otherwise specified by particular methods. By default, the associations between gene expression and segmented copy number data (ge + cn.seg) are investigated.

4 Benchmarking examples

The package contains a copy of the publicly available cancer data sets from [12] and [13] ('Hyman' and 'Pollack' data sets) and two simulated data sets from

[1] http://curl.haxx.se/download.html
[2] http://cran.r-project.org/web/packages/XML/index.html
[3] http://www.omegahat.org/RCurl/
[4] http://www.statistik.tu-dortmund.de/ schaefer/
[5] http://www.few.vu.nl/ wvanwie/software/intCNGEan/intCNGEan.html
[6] http://csbi.ltdk.helsinki.fi/CNAmet/

previous publications [10, 5] ('Ferrari' and 'Schaefer' data sets). For details, see [1] Showcases running the benchmarking pipeline on each data set are described below.

4.1 Hyman et al. (2002)

The Hyman et al. (2002) [12] breast cancer data set[7], and a golden standard list of known breast cancer genes from The Breast Cancer Gene Database [14] provide the first example data set for benchmarking the comparison algorithms. The cancer gene list was downloaded[8] and stored to the tgdb object. The gene symbols are converted into Entrez Gene IDs, the probes are matched between gene expression and copy number data, as detailed in the *read.hyman* function, and the known breast cancer genes from the TGDB golden standard list present in the ge/cn data are selected. Further details are detailed in the *read.hyman* and *the get.brca.genes* functions.

For Hyman, the original non-segmented data set from the publication is used (cn.seg = cn.raw) in the experiments (except with intCNGEan and CNAmet that require segmented and called data, respectively). To run the intcomp benchmarking pipeline on Hyman data set, use

```
> # Load the benchmarking pipeline
> library(intcomp)
> # Load and preprocess the Hyman data
> data(hyman)
> library("org.Hs.eg.db")
> symbol2entrezid <- as.list(org.Hs.egALIAS2EG)
> hyman <- read.hyman(cdna, cgh, genenames, xx = symbol2entrezid)
> # Load the known breast cancer genes
> data(tgdb)
> # Convert to EntrezIDs and select genes common with the ge/cn data
> cancerGenes <- get.brca.genes(rownames(hyman$ge$data), symbol2entrezid, tgdb)
> # Set comparison methods
> methods <- c("OrtizEstevez","intcngean", "edira", "pint",
+              "SIM.window",  "SIM.full", "DRI.cp", "DRI.cs",
+              "PMA.raw", "CNAmet", "DRI.ct")
> # Run the benchmarking pipeline
> set.seed(244441)
> res.hyman <- test.geneorder.pipeline(
+                     ge = hyman$ge,
+                 cn.raw = hyman$cn.raw,
+                cghCall = hyman$cghCall,
+            cancerGenes = cancerGenes,
+                methods = methods,
+             cn.default = "raw",
+                  nperm = 1e3)
> # Check the AUC values for the methods
> auc.ordered <- sort(unlist(res.hyman$auc))
```

[7] HymancdnaDataA.tab, HymancghDataA.tab and HymanAcc.mat obtained from http://www.ece.ucsb.edu/pubs/ieee/index.shtml accessed June 2, 2010.

[8] http://www.tumor-gene.org/cgi-bin/TGDB/tgdb_by_name.cgi accessed 5.6.2010; 'tgdb_by_name.cgi.html' and 'tgdb.txt'

4.2 Pollack et al. (2002)

The Pollack et al. (2002) [13] data set[9] is also used in combination with the golden standard list from the TGDB (See Hyman data set). The gene identifiers in the Pollack data are converted into Entrez Gene IDs. To run the benchmarking tests on Pollack data set, use

```
> # Load and preprocess the GE/CN data
> library(intcomp)
> data(pollack)
> pollack <- read.pollack(dat = CopyNoGeneDataset4719,
+                         clone2geneid = clone2geneid)
> # Load the known breast cancer genes
> library("org.Hs.eg.db")
> data(tgdb)
> # Convert to EntrezIDs and select genes common with the ge/cn data
> cancerGenes <- get.brca.genes(rownames(pollack$ge$data),
+                           as.list(org.Hs.egALIAS2EG), tgdb)
> # Set comparison methods for benchmarking
> methods <- c("OrtizEstevez","intcngean", "edira", "pint", "SIM.window",
+              "SIM.full", "DRI.cp", "DRI.cs", "DRI.ct", "PMA.raw", "CNAmet")
> # Run the benchmarking pipeline
> res.pollack <- test.geneorder.pipeline(ge = pollack$ge,
+                                        cn.raw = pollack$cn.raw,
+                                        cghCall = pollack$cghCall,
+                                        cancerGenes = cancerGenes,
+                                        methods = methods,
+                                        cn.default = "raw",
+                                        nperm = 1e3)
> # Check the AUC values for the methods
> auc.ordered <- sort(unlist(res.pollack$auc))
```

4.3 Ferrari data set (2009)

The first simulated data set, where the exact ground truth is known, is provided by the simulation approach given in [10]:

```
> library(intcomp)
> # Generate simulated data
> ferrari <- test.simulation(GE, CN, method = "ferrari")
> # Set comparison methods
> methods <- c("edira" , "DRI.cp" , "DRI.cs" , "SIM.full" , "SIM.window",
+              "CNAmet" , "intcngean" , "PMA.raw", "pint" , "OrtizEstevez",
+              "PREDA")
> #Run benchmarking pipeline
> set.seed(335)
> res.ferrari  <- test.geneorder.pipeline(
+             ge = ferrari$ge,
```

[9] http://www.pnas.org/content/suppl/2002/09/23/162471999.DC1/4719CopyNoGeneDatsetLegend.html accessed June 2, 2010.

```
+        cn.raw  = ferrari$cn.raw,
+        cn.seg  = ferrari$cn.seg,
+        cn.call  = ferrari$cn.call,
+        cghCall = ferrari$cn.cghCall,
+        cancerGenes = ferrari$cancerGenes,
+        methods = methods,
+        nperm = 1e3)
> # Check the AUC values from comparison methods
> auc.ordered <- sort(unlist(res.ferrari$auc))
```

4.4 Schaefer data set (2009)

The second simulated data set is provided by the simulation approach given in [5] with added flexibility. The quantile grid to be simulated can be defined by the user, as well as the mixing weight, the number of different variances to be considered and the call probabilities.

```
> library(intcomp)
> library(ediraAMLdata)
> # Generate simulated data
> data(AMLdata, package="ediraAMLdata")
> schaefer <- test.simulation(GE, CN, method = "schaefer")
> # Set the methods for benchmarking
> methods <- c("edira" , "DRI.cp" , "DRI.cs" , "DRI.ct", "SIM.full",
+              "SIM.window", "CNAmet" , "intcngean" , "PMA.raw", "pint" ,
+              "OrtizEstevez", "PREDA")
> # Run the benchmarking pipeline
> res.schaefer  <- test.geneorder.pipeline(
+        ge = schaefer$ge,
+        cn.raw = schaefer$cn.raw,
+        cghCall = schaefer$cn.cghCall,
+        cancerGenes = schaefer$cancerGenes,
+        methods = methods,
+        callprobs = schaefer$callprobs,
+        nperm = 1e3)
> # Check the AUC values from comparison methods
> auc.ordered <- sort(unlist(res.schaefer$auc))
```

5 Comparison methods

The following implementations included in the *intcomp* benchmarking pipeline: *CNAmet* [2, 3], variants of *DRI* [4], *edira* [5], *intCNGEan* [6], *Ortiz-Estevez* [7], *pint* [8], variants of *SIM* [11], *PMA* [9], *PREDA/SODEGIR* [10, 15]. The list of available methods in the pipeline is retrieved with:

```
> library(intcomp)
```

```
> list.methods()
```

```
 [1] "edira"        "DRI.cp"       "DRI.cs"       "DRI.ct"       "SIM.full"
 [6] "SIM.window"   "CNAmet"       "intcngean"    "PMA.raw"      "pint"
[11] "OrtizEstevez" "PREDA"
```

6 Benchmarking results

The prioritized cancer gene list provided by each method is compared to the golden standard list of known cancer genes; the result contains running times of the algorithms and the AUC values from ROC analysis. The AUC values provide quantitative estimates of model performance in cancer gene detection and provide the basis for the comparisons as reported in [1].

6.1 Version details

The following package versions were used to produce this vignette:

```
> sessionInfo()

R version 2.15.0 (2012-03-30)
Platform: x86_64-pc-linux-gnu (64-bit)

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

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

other attached packages:
 [1] intcomp_0.3.28       intCNGEan_0.54       PREDA_1.2.0
 [4] annotate_1.34.0      multtest_2.12.0      lokern_1.1-3
 [7] ediraAMLdata_1.0.5   CNAmet_1.2           CGHcall_2.16.0
[10] CGHbase_1.14.0       marray_1.34.0        limma_3.12.0
[13] SIM_1.26.0           quantreg_4.79        SparseM_0.96
[16] PMA_1.0.8            plyr_1.7.1           pint_1.8.0
[19] dmt_0.7.06           MASS_7.3-18          Matrix_1.0-6
[22] lattice_0.20-6       mvtnorm_0.9-9992     org.Hs.eg.db_2.7.1
[25] RSQLite_0.11.1       DBI_0.2-5            AnnotationDbi_1.18.0
[28] edira_1.1.9          tilingArray_1.34.0   pixmap_0.4-11
[31] DRI_1.1              cghFLasso_0.2-1      impute_1.30.0
[34] DNAcopy_1.30.0       biomaRt_2.12.0       affy_1.34.0
[37] Biobase_2.16.0       BiocGenerics_0.2.0

loaded via a namespace (and not attached):
 [1] affyio_1.24.0         BiocInstaller_1.4.4   genefilter_1.38.0
 [4] globaltest_5.10.0     grid_2.15.0           IRanges_1.14.3
```

```
 [7] preprocessCore_1.18.0 quantsmooth_1.22.0    RColorBrewer_1.0-5
[10] RCurl_1.91-1          sfsmisc_1.0-20        splines_2.15.0
[13] stats4_2.15.0         strucchange_1.4-6     survival_2.36-14
[16] tools_2.15.0          vsn_3.24.0            XML_3.9-4
[19] xtable_1.7-0          zlibbioc_1.2.0
```


Acknowledgements

This work has been supported by EuGESMA COST Action BM0801: European Genetic and Epigenetic Study on AML and MDS. We would also like to thank Fransesco Ferrari for providing simulated data for the study.

References

[1] L. Lahti, M. Schäfer, H.-U. Klein, S. Bicciato, and M. Dugas, "Cancer gene prioritization by multi-platform data integration: a comparative review. upcoming,"

[2] S. Hautaniemi, M. Ringnér, P. Kauraniemi, R. Autio, H. Edgren, O. Yli-Harja, J. Astola, A. Kallioniemi, and O. Kallioniemi, "A strategy for identifying putative causes of gene expression variation in human cancers," *Journal of the Franklin Institute*, vol. 341, no. 1-2, pp. 77–88, 2004.

[3] R. Louhimo and S. Hautaniemi, "CNAmet: an R package for integrating copy number, methylation and expression data," *Bioinformatics*, vol. 27, no. 6, pp. 887–888, 2011.

[4] K. Salari, R. Tibshirani, and J. R. Pollack, "DR-Integrator: a new analytic tool for integrating DNA copy number and gene expression data.," *Bioinformatics*, vol. 26, no. 3, pp. 414–6, 2010.

[5] M. Schäfer, H. Schwender, S. Merk, C. Haferlach, K. Ickstadt, and M. Dugas, "Integrated analysis of copy number alterations and gene expression: a bivariate assessment of equally directed abnormalities," *Bioinformatics*, vol. 25, no. 24, pp. 3228–3235, 2009.

[6] W. N. van Wieringen and M. A. van de Wiel, "Nonparametric testing for DNA copy number induced differential mRNA gene expression," *Biometrics*, vol. 65, pp. 19–29, 2009.

[7] M. Ortiz-Estevez, J. De Las Rivas, C. Fontanillo, and A. Rubio, "Segmentation of genomic and transcriptomic microarrays data reveals major correlation between DNA copy number aberrations and gene-loci expression," *Genomics*, vol. 97, pp. 86–93, 2011.

[8] L. Lahti, S. Myllykangas, S. Knuutila, and S. Kaski, "Dependency detection with similarity constraints," in *Proceedings MLSP'09 IEEE International Workshop on Machine Learning for Signal Processing XIX*, (Piscataway, NJ), pp. 89–94, IEEE Signal Processing Society, September 2-4 2009. Implementation available in pint package of R/BioConductor http://www.bioconductor.org/packages/release/bioc/html/pint.html.

[9] D. M. Witten, R. Tibshirani, and T. Hastie, "A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis," *Biostatistics*, vol. 10, no. 3, pp. 515–534, 2009.

[10] S. Bicciato, R. Spinelli, M. Zampieri, E. Mangano, F. Ferrari, L. Beltrame, I. Cifola, C. Peano, A. Solari, and C. Battaglia, "A computational procedure to identify significant overlap of differentially expressed and genomic imbalanced regions in cancer datasets," *Nucleic Acids Research*, vol. 37, pp. 5057–5070, 2009.

[11] R. X. Menezes, M. Boetzer, M. Sieswerda, G.-J. B. van Ommen, and J. M. Boer, "Integrated analysis of DNA copy number and gene expression microarray data using gene sets.," *BMC bioinformatics*, vol. 10, no. 1, p. 203, 2009.

[12] E. Hyman, P. Kauraniemi, S. Hautaniemi, M. Wolf, S. Mousses, E. Rozenblum, M. Ringner, G. Sauter, O. Monni, A. Elkahloun, O.-P. Kallioniemi, and A. Kallioniemi, "Impact of DNA Amplification on Gene Expression Patterns in Breast Cancer," *Cancer Res.*, vol. 62, pp. 6240–6245, Nov. 2002.

[13] J. R. Pollack, T. Sø rlie, C. M. Perou, C. A. Rees, S. S. Jeffrey, P. E. Lonning, R. Tibshirani, D. Botstein, A.-L. Bø rresen Dale, and P. O. Brown, "Microarray analysis reveals a major direct role of DNA copy number alteration in the transcriptional program of human breast tumors.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, pp. 12963–8, Oct. 2002.

[14] R. Baasiri, S. Glasser, D. Steffen, and D. Wheeler, "The Breast Cancer Gene Database: a collaborative information resource," *Oncogene*, vol. 18, no. 56, pp. 7958–7965, 1999.

[15] Ferrari F, Solari A, Battaglia C, Bicciato S. PREDA: an R-package to identify regional variations in genomic data. *Bioinformatics* online July 7, 2011.