CAncer bioMarker Prediction Pipeline (CAMPP) - User Manual.

Thilde Bagger Terkelsen
Danish Cancer Society & University of Copenhagen
thilde@cancer.com, thildebate@gmail.com

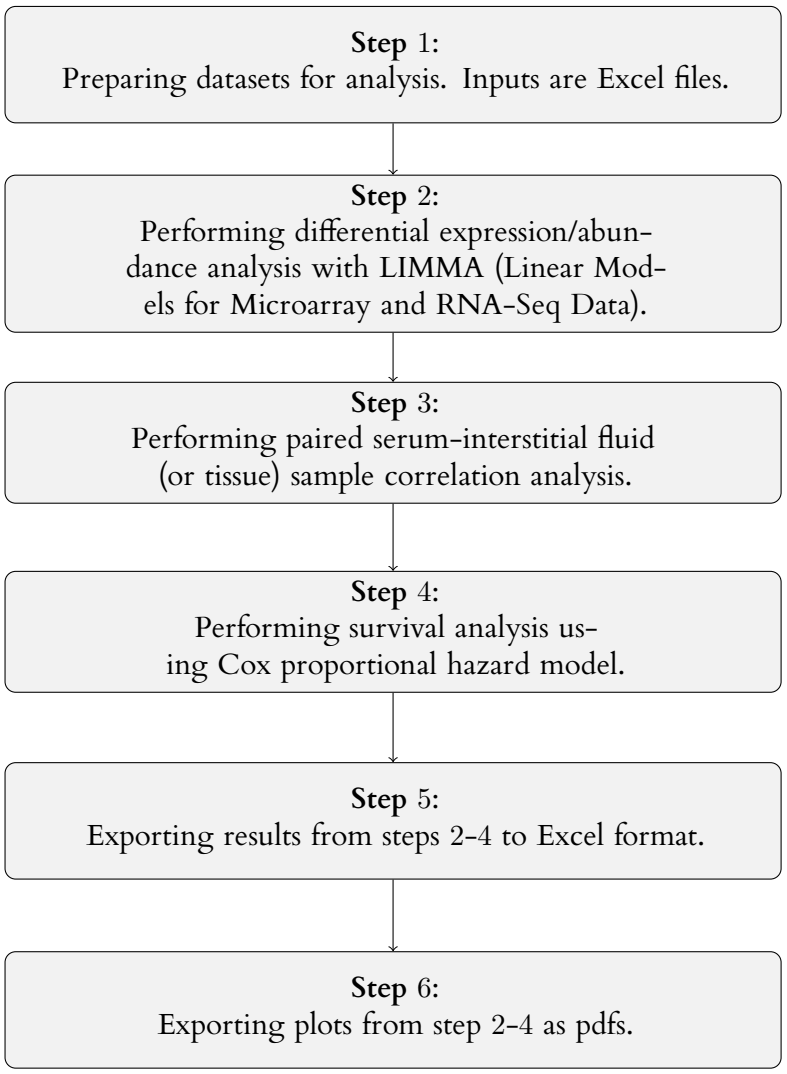
2018-11-06

Introduction

The CAncer bioMarker Prediction Pipeline (CAMPP) is a simple bioinformatics tools intended to aid researches with identification of diagnostic and prognostic cancer biomarkers targetable in serum. The pipeline is versitile and may be used for analysis of a variety of quantitative biological data from high throughput platforms, including miRNAs, mRNAs, proteins and glycans. CAMPP currently supports three types of analysis: differential expression/abundance analysis (limma), correlation analysis (Pearson/Spearman) and survival analysis (Cox proportional hazard regression). CAMPP is written in R [1] and runs using flags in the terminal command-line.

Contents

Workflow Overview



1 About CAMPP

The CAncer bioMarker Prediction Pipeline was developed for internal use at the Danish Cancer Society Research Center (DCRC) providing researchers with an easy way to identify biomarkers for cancer diagnostics and prognostics. In its original form the pipeline was implemented using interstitial fluid samples from patients with breast cancer but can be run with any type of cancer data and tissue sample types.
The pipeline consist of three parts (I) Differential Expression/Abundance Analysis (II) Pearsons/Spearman Correlation Analysis and (III) Survival analysis (overall survival) with cox proportional-hazard model. Differential expression/abundance analysis can be performed with as few as 3 biological replicates in each group, whereas serum correlation and survival analysis is only advisable with a decently large number of samples (a minimum of $\sim$ 100 samples is recommended). CAMPP

may be run with a variety of biological molecules (mRNAs, miRNAs, proteins and N-glycans) from various platforms (high-throughput sequencing, microarray data, liquid chromatography–mass spectrometry, ect.).
For differential expression/abundance analysis the limma package [12] for R is utilized. Though limma was originally designed for microarray data, and more recently RNAseq, a number of studies have shown the versatility of this software for the analysis of other -omics data [4, 9, 10]. LIMMA has few underlying statistical assumptions and is known to be powerful for small sample sizes as a result of shrinkage of feature-specific variances [14].
Survival (over-all, relapse-free) of patients based on the abundance/expression of a given marker is predicted usign cox proportional-hazard model. The model is corrected for age at diagnosis (surgery, entry into trail) by default, however, if correction for additional confounders is desired these should be specified by the user.

> N.B For every parameter added to the model more events are needed for appropriate statistical power. A rule of thumb is $\sim$ 10 events for each parameter, however, this will somewhat depend on the number of levels within a given parameter. Generally results of the survival analysis should be interpreted with caution as the pipeline is unable to account for all possible options and assumptions related to this type of analysis.

2 Requirements

To run CAMPP, a working R version 3.3.2 (or newer) is required [1]. The Pipeline relies on a variety of R-packages, see list below. CAMPPInstall.R script checks whether these packages are installed and helps the user download, install and load them (see section on 4).

Table 1: R-packages

Analysis	Packages
DE/DA Analysis	limma, sva
Excel formatting	openxlsx, xlsx
Plotting	ggplot2, heatmap.plus, RColorBrewer, squash
Data management	plyr, data.table, scales, stackoverflow
Survival analysis	survcomp, survminer

> To obtain information on R-packages above, e.g. developers and release information (articles) go to `R` and type `citation("package_name")`.

3 Download

The CAncer bioMarker Prediction Pipeline is easily installed by following the few outlined steps below.

1. (1) Download the GitHub repository locally from the address below. (2) If you wish to work externally on a server you can upload the repository from your computer to the server using scp from the command-line or (3) use git for a direct download:

Box 3.1: Download

(1) `https://github.com/ELELAB/CAncer-bioMarker-Prediction-Pipeline-CAMPP`

(2) `scp CAncer-bioMarker-Prediction-Pipeline-CAMPP-master.zip name@login:/path-to-dir.`

(3) `git clone git://github.com/ELELAB/CAncer-bioMarker-Prediction-Pipeline-CAMPP.git`

> If you are using the "git clone" option, make sure that the Git software has been installed.

2. Unzip the repository. From the command-line use "unzip" to decompress.

Box 3.2: Decompress

```
unzip CAncer-bioMarker-Prediction-Pipeline-CAMPP-master.zip
```

> The unzipped repository should contain the following: `Readme.md LICENSE.md CAMPPInstall.R, CAMPPFunctions.R CAMPP.R` and a folder named `DataExamples`.

4 Installation of R-packages

The CAncer bioMarker Prediction Pipeline consists of three R-scripts; CAMPPInstall.R, CAMPPFunctions.R and CAMPP.R. The CAMPPInstall.R only needs to be used the first time the pipeline is run - this script ensures that all required R-packages will be installed. The CAMPPFunctions is a script containing custom functions used in the analysis and must therefore be located in the directory from where the pipeline is run.

1. In order to ensure that all R-packages needed have been installed, run the CAMPPInstall.R script from the command-line.
2. Running `CAMPPInstall.R` will generate a script named `CAMPP-missingpackages.R`. This script contains the names of any packages not installed. If no `CAMPPmissingpackages.R` script is generated it means that the required R-packages already exist.

Box 4.1: Install R-packages

```
Rscript CAMPPInstall.R
```

3. To require any missing packages open R in the command-line and source the `CAMPPmissingpackages.R` script. R will suggest a library to place the packages in and will query about a "CRAN-mirror" - pick the mirror closest to your geographical location.

Box 4.3: Install R-packages

```
R
source("CAMPPmissingpackages.R")
```

4. Although the CAMPPmissingpackages.R will aid the user in installing the required R-packages from both CRAN and Bioconductor, there may be instances where a package is not updated and compatible with the running version of R.
   `Warning message:`
   `In install.packages("name_of_package") : package 'name_of_package' is not available...`
   In this case the package may need to be installed directly from the github repository. In order to do this check that the R-package `devtools` is installed and loaded. Next, use `install.github()` to install:

Box 4.4: Install R-packages

```
install.packages("devtools")
library(devtools)
install_github(Bioconductor/name_of_package)
install_github(Bioconductor-mirror/name_of_package)
```

5 Running CAMPP

After installation of all R-packages the CAncer bioMarker Prediction Pipeline is ready be used.

> It is essential that the script `CAMPPFunctions.R` is located in the directory from which the pipeline is run.

The pipeline in run from the command-line using flags. In the folder `DataExamples` the user will find examples of files needed for running CAMPP.
The pipeline currently support three types of analysis. Differential

expression/abundance with Limma, paired interstitial fluid (or solid tissue) and serum sample correlation and survival analysis with cox proportional hazard model (Flowchart on page 2).

5.1 Mandatory Data Input

CAMPP needs as minimum two excel files to run.

1. **Data:** An excel file containing feature expression/abundance. With rows as features, and columns as samples, e.g. columns are N-glycan, protein, (mi)RNA identifiers and rows are sample IDs. The folder `DataExamples` contains an example with N-glycans named: `glycandata.xlsx`.
   - **Normalization:** If the pipeline is run with proteins or N-glycans then values should preferably be normalized (ex: median centered).
   - **Transformation:** The data can be transformed (log10, log2, logit, voom) or not. The pipeline has an option for doing transformation if desired.
2. **Metadata:** An excel file containing metadata. This file must contain at least two columns named "ids" with identifiers matching the column names in the data file and a column named "group" specifying a group for DE/DA analysis e.g. diagnosis (tumour or normal), tumour stage (1,2 or 3), drug treatment (A, B C) ect. The folder `DataExamples` contains an example with N-glycans named: `glycanmetadata.xlsx`.
   - **Batch:** If the data comes from experimental batches and the user wants to correct for this, a column named "batch" specifying which batch each sample belongs to (A,B or C, batch1, batch2, batch3 or batch4, ect.) should also be included in the metadata. Batch type must be denoted as a character, meaning numbers alone are not allowed.
   - **Matched Serum:** If the user is interested in performing correlation analysis a column named "serum" must be included in the metadata, specifying (in a binary way) which samples have a matched serum samples (denoted by 1) and which that do not (denoted by 0).

     N.B. if paired samples are avalibe for analysis the column 'serum' should only have the value 1 for those samples (either tumours or normals, A or B ect.) you choose to test for - not both.

   - **Survival Information:** If the user is interested in performing survival analysis a column named 'survival' must be included, specifying (in a binary way) which samples have survival information (denoted by 1) and which do not (denoted by 0).

N.B. if you have (paired) cancer and normal samples the column 'survival' should only have the value 1/0 for tumour samples (NA or other character values should be used for normal samples).

5.2 Arguments

1. **Simple Run:** To run the pipeline, arguments (flags) must be set to specify the input. In the simplest case we are interested in performing differential abundance/expression analysis and for this we only need to specify the two mandatory inputs.

Box 5.2.1: Mandatory Arguments

`-d`: Quantitative data
`-m`: Metadata

The box below shows an example of the simplest possible CAMPP run.

Box 5.2.2: First Run Example

```
Rscript CAMPP.R -d glycandata.xlsx -m glycanmetadata.xlsx
-a TRUE
```

2. The output of the command above will be an excel file with the identifiers, test-statistics, fold changes (logFC) and p-values of any differentially expressed/abundant features, as well as a heatmap visualization in pdf format specified by the option **-a**. For large datasets with thousands of variables heatmap plotting is NOT advisable!

Table 2: Example of Excel Output

logFC	t	P.Val	adj.P.Val	B	name	dir.	comp.
2.1	11.6	7.7e-22	4.9e-20	38.9	pA	up	T-N
1.3	8.3	8.4e-14	1.1e-12	20.6	pK	up	T-N
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
-1.2	-10.2	4.1-18	1.3e-16	30.5	pH	down	T-N

N.B. The pipeline logs while running and produces a text file, `CAMPPlog.txt`, with any errors or comments accumulated during analyses - this file should be viewed carefully.

3. **Help:** In addition to the two mandatory arguments above there are a range of optional arguments which may be utilized. The -h (help) option allows the user to obtain information about all available flags.

Box 5.2.3: Title.

```
Rscript CAMPP.R -h
```

4. **Other Arguments:** The table below show a some useful arguments.

Box 5.2.4: Optional Arguments

`-s`:	Serum Analysis
`-u`:	Survival Analysis
`-b`:	Batch Correction
`-t`:	Data Transformation
`-o`:	Multidimensional Scaling Plot

- **Serum Analysis (`-s`):** The flag `-s` must be set to perform correlation analysis between interstitial fluid samples (or tissue) and matched serum. The user must provide an excel sheet with the expression/abundance values in serum with rows as features and columns as samples - with the same sample order as the matched interstitial fluid (tissue) samples from the data excel file. In addition to this, a column named 'serum' must be added to `metadata.xlsx`, specifying (in a binary way) whether a given sample has a match serum sample (1) or not (0) - this allows for cases of missing samples. The folder `DataExamples` contains an example with N-glycans named, `glycanserum.xlsx`.
- **Survival Analysis (`-u`):** The flag `-u` must be set to TRUE in order to perform survival analysis using cox proportional hazard model. For survival analysis the `metadata.xlsx` sheet must contain at least four columns in addition to the sample IDs named; 'age' (age in years at diagnosis, surgery or entry into trail), 'outcome.time' (time until end of follow-up, censuring or death in weeks, months or years), 'outcome' (numeric 0 = censuring, 1=dead) and 'survival' (numeric 0 = no survival info, 1=survival info available). If the user wishes to correct for potential confounders (e.g. tumor grade, hormone levels, drug-treatment ect.) these should also be included in the `metadata.xlsx`.
- **Batch Correction (`-b`):** The flag `-b` may be added if the user has experimental batches in the expression/abundance data and wants to correct for this. The flag is boolean, e.g. it is set to either TRUE or FALSE (default = FALSE). In order to perform

batch correction the metadata file must contain a column named 'batch' specifying which sample belongs to which batch (A, B or C, batch1, batch2, batch3 or batch4, ect.). Batch type must be denoted as a character, meaning numbers alone are not allowed.

- **Data Transformation(-t):** The flag `-t` may be set if the user desires the expression/abundance data to be transformed before analysis. A logarithmic transformation is recommended as the variance of measurements, from most platforms, depend on the expression/abundance level itself. The log-transformation reduces this dependency and additionally pushes the negative binomial distribution, displayed by count data (miRNA, mRNA), towards a normal distribution. The user may choose between log2, log10, logit or voom tranformation. For RNA-seq the voom transformation is recommended due to the variable library sizes assocated with this type of data [12], while a log tranformation may be more appropriate for proteomics and N-glycan abundances.
- **Multidimensional Scaling plot (-o):** The flag `-o` may be used to generate a preliminary multidimensional scaling (MDS) plot. Multidimensional scaling (MDS) reduces high-dimensional data to two (or more) "observable" dimensions (M1 and M2) in such a way that the inter-sample distance relationship in the simplified dimensions mimick the distance relationship between the original values of samples in n-dimensional space (n = number of features). MDS was performed with the euclidean distances as the distance metric. A MDS plot may help to determine whether there appears to be any "grouping" of data in a desired way (tumor samples together and normal samples together) or in an undesired way (experimental batches) before performing the DE/DA analysis.

6 N-glycan Serum Markers for BC Diagnostics

This section contains an example of how CAMPP is run using different flags in the command-line and what output the user can expect. The files used for running this example may be found in the repository folder `DataExamples`.
The case below uses N-glycans abundances measured using high resolution quantitative Ultra-Performance Liquid Chromatography (UPLC) [13] from interstitial samples [6] and matched serum. Tumor interstitial fluid (TIF), normal interstitial fluid (NIF) and serum samples were collected from $\sim$ 90 women diagnosed with breast cancer (BC). A total of 165 N-glycan groups were identified [15].

Briefly, the involvement of N-glycosylation in development and progression of BC has been documented by both in vitro and in vivo studies [3, 5, 7]. Several circulating N-glycan patterns with altered glycan structures, possibly originating from a primary tumor or from

other organs in response to a neoplastic process, have recently been described in a number of studies by using high-throughput N-glycan profiling [2, 8, 11, 13].

6.1 Differential Abundance Analysis

N-glycan abundances used in this example are relative (fractional) meaning that the sum of all values within one sample yields 100. Differential Abundance Analysis with CAMPP was performed with correction for experimental batch (b1, b2, b3). The data were log2 transformed (flag `-t`) and a preliminary MDS plot was generated (flag `-o`.) The cut-off for a significant hit was an FDR <= 0.05 (default) - no logFC cut-off was set to define N-glycan differential abundance (flag `-f`). Files used for this example may be found in the `DataExamples` folder.

The box below shows an example of DE/DA analysis with CAMPP.

Box 6.1: Differential Abundance Analysis.

```
Rscript CAMPP.R -d glycandata.xlsx -m glycanmetadata.xlsx
-t log2 -b TRUE -o TRUE -n FirstRunOutput
```

Output:
The run above generates three outputs:

1. A multidimensional scaling plot (`FirstRunOutput_MDSplot.pdf`) using the abundance/expression of features (in this case N-glycans). The components M1 and M2 in the plot below are those which best retained the distance relationship between samples in two dimensions [ref]. - See **Figure 1.**
2. An excel file (`FirstRunOutput_DE.xlsx`) with the stats for significant differentially abundant/expressed features (in this case N-glycans) - See example of format in **Table 2.**
3. A heatmap (`FirstRunOutput_heatmap.pdf`) visualizing the power of the identified DA/DE features (in this case N-glycans) for partitioning of the samples in groups/condition/treatment (in this case NIF and TIF samples)- See **Figure 2.**

The multidimensional scaling plot in **Figure 1** indicates that N-glycan abundance patterns may confer a separation of tumour interstitial fluid and normal interstitial fluid samples from patients with breast cancer - at least when all N-glycans features are retained. The MDS plot may be used as an indicator of whether the differential expression analysis is likely to yield any results, e.g. no clustering of samples from different groups/conditions/treatments implies that the biological data does NOT reflect the sample classification.

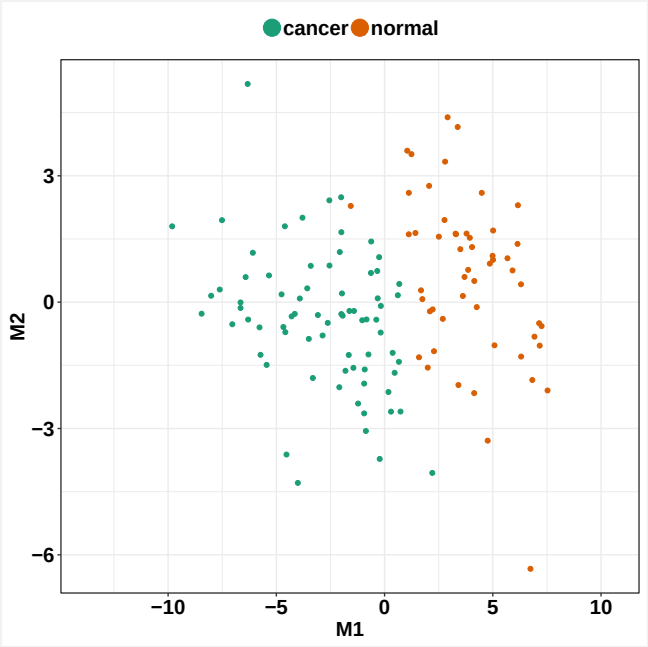


Figure 1: Multidimensional Scaling Plot showing the separtion of tumor interstitial and normal interstitial fluid samples based on N-glycan abundances. The data was batch corrected before plotting.

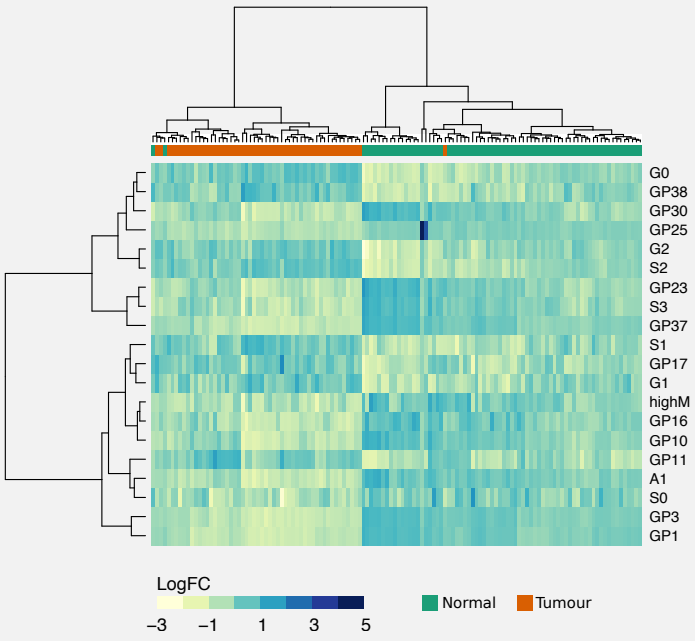


Figure 2: Heatmap showing the partitioning of TIF and NIF samples based on 20 N-glycan peaks/groups identified as differentially abundant.

6.2 Correlation of abundances in TIF and serum

In order to determine whether N-glycans with differential abundances in tumour -and normal interstitial fluids displayed corresponding patterns in serum, correlation was performed using TIF samples and matched serum.

For correlation analysis with CAMPP, the user must provide:

1. An excel sheet with serum abundances (rows as features and columns as samples). See section 5.2.4 for specifications. An example of such a file may be found in the folder `DataExamples`.
2. The file `metadata.xlsx` must contain a column named 'serum'. See section 5.1.2 for specifications.
3. If the serum samples are produced in batches the option `-e` may be used to correct for this before analysis. Serum batches should be included (like other sample batches) in the `metadata.xlsx` in a column named 'sbatch'.

Files used in the example below, can be found in the `DataExamples` folder.
Command-line box 6.2.3 shows an example of a CAMPP correlation analysis run. TIF and serum N-glycan abundances were corrected for experimental batch and log2 transformed before correlation.

Box 6.2.3: Correlation Analysis.

```
Rscript CAMPP.R -d glycandata.xlsx -m glycanmetadata.xlsx
-t log2 -b TRUE -s glycanserum.xlsx -e TRUE -n SecondRunOutput
```

Output:
The run above generates three outputs:

1. An excel file (`SecondRunOut_corr_serum_.xlsx`) with the stats for features (in this case N-glycans) displaying significant correlation of abundance/expression in tumor sample (interstitial fluid) and matched serum.

> **N.B** Only features found to be DA/DE between the tested groups/condition/treatment are included in the correlation analysis.

2. A correlation plot (`SecondRunOutput_corrplot.pdf`) visualizing the correlation coefficients and adjusted p-values of all differentially expressed/abundant features - in this case N-glycans found to partition TIF and NIF. See **Figure 3**.
3. Individual scatter plots (`SecondRunOutput_individual_corrplots.pdf`) for the features displaying a significant correlation between expression/abundance in TIF/tissue and serum (FDR <= 0.05 is default). See **Figure 4**.

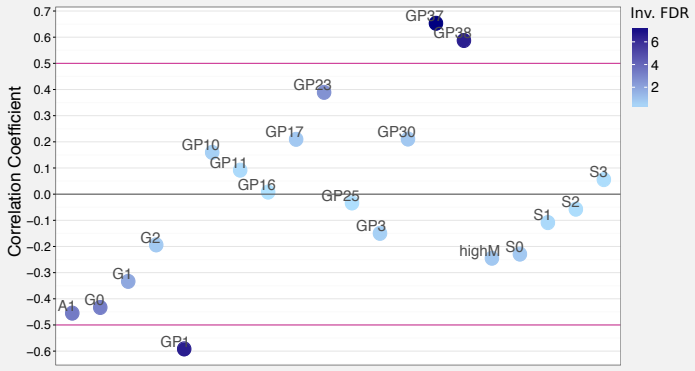


Figure 3: Correlation plot depicting correlation coefficients associated with abundances of N-glycans in TIF and mathed serum. Correlation was performed with the 20 N-glycans identified as DA in normal and tumour fluids. Dots are colored in accordance with inverse (scaled) FDRs, e.g. darker shade indicates smaller adjust p-value and vice versa.

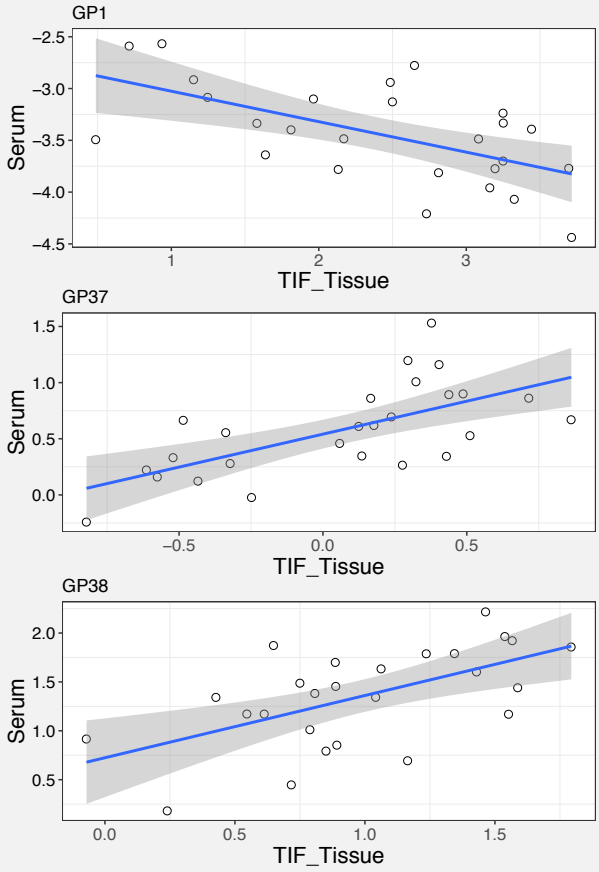


Figure 4: Scatter plots of the 3 N-glycans (GP1, GP37 and GP38) with significant correlations between abundances in TIF and serum. Shaded area indicates confidence of regression line.

Based on the excel sheet with coefficients and adjusted p-values (not shown), three N-glycans, GP1, GP37 and GP38 where found to display significant correlations of abundances in TIF and serum (see Figure 3). It is worth noting that levels of GP1 in TIF and serum are negatively correlated indicating an accumulation of this N-glycan within the tumor interstitium, perhaps as a result of primary tumor response. GP37 and GP38 have positive correlation coefficients and may be considered as potential diagnostic serum markers.
Figure 4 shows the scatter plots (TIF abundance plotted against serum abundance) of GP1, GP37 and GP28, with confidence shading of regression lines.

6.3 Survival Analysis

Cox proportional hazard model was used to assess whether any differentially abundant N-glycans had potential as prognostic biomarkers for the survival of breast cancer patients.
CAMPP only provides cox proportional hazard models for overall survival and simple relapse-free survival, e.g. one time entry per patient/sample.
CAMPP checks two underlying assumptions of the cox model before performing survival analysis (I) a linear relationship of continuous covariates with log hazards and (II) proportional hazards of categorical and continuous covariates, e.i. constant relative hazard. If the requirement of linearity is not fulfilled, cubic splines will be added to the covariates(s) in question, and analysis is continued. If the proportional hazard assumption is violated for a categorical variable the user may apply stratification and re-run the pipeline. However, the pipeline does not currently handle continuous variables with non-proportional hazards, e.i. that the user should ignore any cox models where this is the case.

In order to perform cox proportional hazard regression with CAMPP, the user must provide:

1. An excel sheet with patient survival information, including patient ID, patient age, time to outcome and outcome (dead or censured). Requirements for this excel sheet are specified in section 5.2.4 under the heading 'Survival Analysis'. An example of such a file may be found in the folder `DataExamples`.

Files used in the example below, can be found in the `DataExamples` folder.
Command-line box 6.3.2 shows an example of a CAMPP survival analysis run. TIF N-glycan abundances were corrected for experimental batch and log2 transformed before regression analysis.

Box 6.3.2.1: Survival Analysis (basic).

```
Rscript CAMPP.R -d glycandata.xlsx -m glycanmetadata.xlsx
-t log2 -b TRUE -u TRUE -n ThirdRunOutput
```

In the example above the cox model only contains patient age at diagnosis (default), however, the user may specify other confounders to include. Covariates are included with the flag **-p**. If multiple confounders are added these should be separated by commas and their names should match those of the desired columns within `metadata.xlsx`. In the run below, survival analysis was performed with tumor infiltrating lymphocyte status (name: TILS, type: categorical) and tumor size (name: TSize, type: continuous) as covariates – See the `metadata.xlsx` sheet in the `DataExampels` folder.

N.B It is important to note that each time a covariate is added to the model we increase the degrees of freedom. It is therefore essential to have a dataset of an appropriate size for the number of covariates added. The good rule of thumb is 10 events per parameter/variable.

Box 6.3.2.2: Survival Analysis (confounders).

```
Rscript CAMPP.R -d glycandata.xlsx -m glycanmetadata.xlsx
-t log2 -b TRUE -u TRUE -p TILS,TP -n ThirdRunOutput
```

If a CAMPP run produces the warning message below, it means that some of the specified covariates violate the proportional hazard assumption:

```
WARNING: The following features and/or covariates failed the test of
proportional hazard:  TILS.
IF the covariates that failed are categorical you may use strata by
re-running the pipline adding flag -y followed by the names of the
categorical covariates to stratify (if multiple, separate by comma).
N.B, this pipeline does not handle continuous variables that violate
the proportional hazard assumption, if any of these failed PH test,
the hazard ratios of these should NOT be evaluated.
```

If the covariates are categorical we can re-run the pipeline with stratification using the flag **-y**:

Box 6.3.2.2: Survival Analysis (confounders and stratification).

```
Rscript CAMPP.R -d glycandata.xlsx -m glycanmetadata.xlsx
-t log2 -b TRUE -u TRUE -p TILS,TP -y TILS -n ThirdRunOutput
```

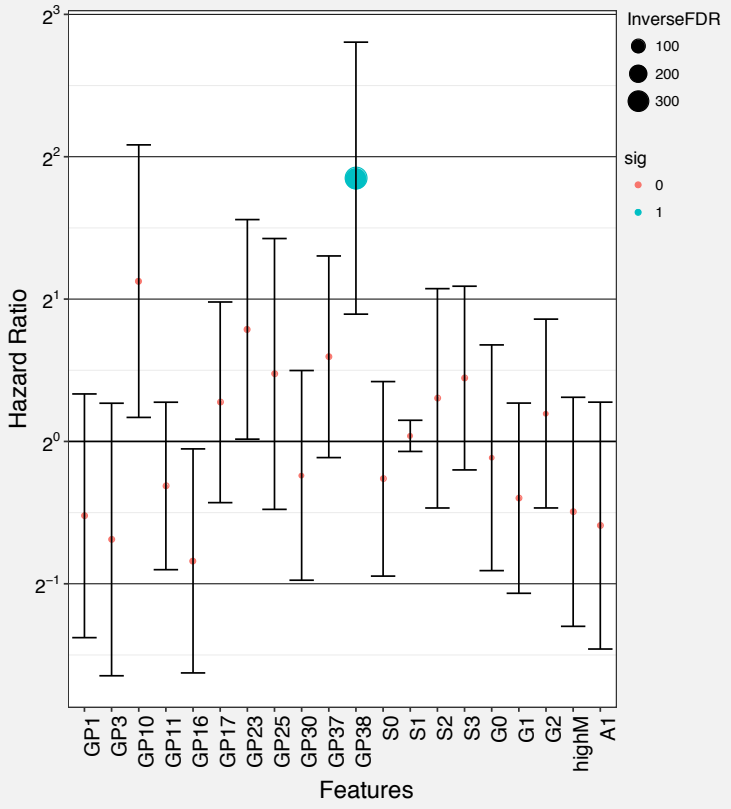


Figure 5: Summary plot of hazard ratios (and confidence intervals) for cox regression with each of the 20 N-glycans found to be differentially abundant between TIF and NIF. A hazard ratio (HR) < 1 indicates that a high level of a given N-glycan is associated with a positive outcome (longer overall survival after diagnosis), while a HR > 1 suggests that a high level of a given N-glycan predicts poorer overall survival. The dot size denotes inverse FDR, e.g. a larger dot is associated with a smaller FDR. Significant N-glycans (FDR <= 0.05) have blue dots, while non-significant N-glycans have orange dots.

Output:
The runs above generate two outputs:

1. An excel file (`ThirdRunOut_survival.xlsx`) with the stats for features (in this case N-glycans) displaying significant associations between abundance/expression in tumor sample (interstitial fluid) and patient outcome.

2. A plot of the hazard ratios for features (in this case N-glycans) with confidence intervals and significance (`ThirdRunOut_survivalplot.pdf`). See **Figure 5**.

Figure 5 shows that one N-glycan, GP38, was found to be significantly associated with survival, e.i. a high level of this N-glycan was predictive for poor overall survival. GP38 was one of the three N-glycans displaying a correlation between abundances in TIF and serum, suggestion that this glycan structure may have potential as a prognostic serum biomarker.

References

[1] R Core Team. *R: A language and environment for statistical computing.* R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/. (2014).

[2] Abd Hamid, U.M et al. *A strategy to reveal potential glycan markers from serum glycoproteins associated with breast cancer progression.* Glycobiology 18, (2008): 1105–1118.

[3] Abbott, K.L. et al. *Targeted glycoproteomic identification of biomarkers for human breast carcinoma.* Journal of Proteome Research 7, (2008):1470–1480.

[4] Castello, A., et al. *Insights into RNA biology from an atlas of mammalian mRNA-binding proteins.* Cell 149.6 (2012): 1393-1406.

[5] Christiansen M.N et al. *Cell surface protein glycosylation in cancer.* Proteomics 14, (2014):525–546.

[6] Gromov, P. et al. *Tumor interstitial fluid—a treasure trove of cancer biomarkers.* Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics 1834.11 (2013): 2259-2270.

[7] Guo, H.B. et al. *Specific posttranslational modification regulates early events in mammary carcinoma formation.* Proceedings of the National Academy of Sciences 107, (2010): 21116–21121.

[8] Haakensen, V.D. et al. *Serum N-glycan analysis in breast cancer patients – Relation to tumour biology and clinical outcome.* Journal of Molecular Oncology, (2015):1–14.

[9] Kammers, K., et al. *Detecting significant changes in protein abundance.* EuPA open proteomics 7 (2015): 11-19.

[10] Pineda, A.L., et al. *On Predicting lung cancer subtypes using 'omic' data from tumor and tumor-adjacent histologically-normal tissue.* BMC cancer 16.1 (2016): 184.

[11] Potapenko, I.O. et al. *Glycan-related gene expression signatures in breast cancer subtypes; relation to survival.* Journal of Molecular Oncology 9, (2013): 861–876.

[12] Ritchie, M.E., et al. *limma powers differential expression analyses for RNA-sequencing and microarray studies.* Nucleic Acids Research 43(7), e47.

[13] Saldova, R., et al. *Association of N-glycosylation with breast carcinoma and systemic features using high-resolution quantitative UPLC.*. Journal of proteome research 13.5 (2014): 2314-2327.

[14] Soneson, C., and Delorenzi, M. et al. *A comparison of methods for differential expression analysis of RNA-seq data.* BMC bioinformatics 14.1 (2013): 91.

[15] Terkelsen, T. and Haakansen, V.D. et al. *N-glycan signature identified in tumor interstitial fluid and serum of breast cancer patients - association with tumor biology and clinical outcome*. In revision for molecular oncology.