

FlowR

FlowR consists of a series of workflows, each representing a form of analysis. To date, these are; CutFlow, SurvFlow and CompFlow.

Installation

You can install the released version of FlowR from GITHUB with:

```
install_github("HHayman/FlowR")
```

You can load the FlowR package with:

```
library(FlowR)
```

CutFlow

Arrange your data to meet the following criteria;

- Carry out any exclusions
- Include an identifier variable
- Your status variable needs to be coded as 0 and 1, 0 for no event, 1 for an event
- Time variable must be included as a continuous variable
- There is no limit on the number of variables you wish to generate a cutpoint for
- If coding multiple datasets, the respective variables must have the exact same name in all datasets
 - Additional datasets do not need to contain all variables from your training dataset, just those you wish to be coded in that dataset
- Save your dataset(s) as a CSV file
- Create a subdirectory in your R directory, and place your dataset files inside

To run CutFlow, simply fill in the blanks, as in the example below;

- Subdirectory is the name of the folder you placed your datasets in. It must be within your current directory
- TrainingData is the name of the dataset you wish to be used to generate the cutpoint
 - This cutpoint is then applied to all datasets within the subdirectory
- CutPointStatus is the status variable to be used (such as CSS)
- CutPointTime is the time variable to be used (such as Survival_2021)
- minprop is the minimum proportion of cases to be include either side of the cutpoint
 - Default is 0.1, exclude the argument if you don't want to change this
- Greyscale is an optional toggle to produce a greyscale variant of all plots
 - The default is colour, exclude the argument if you don't want to change this
- Variables is a list of your variables to generate cutpoints for

```

CutFlow(Subdirectory = "YourSubdirectory", TrainingData = "TrainingDataset",
        CutPointStatus = "StatusVariable", CutPointTime = "TimeVariable", minprop = 0.1,
        Greyscale = TRUE, Variables = c( "Variable1",
                                         "Variable2",
                                         "Variable3",
                                         "Variable4",
                                         "Variable5",
                                         "Variable6",
                                         "Variable7",
                                         "Variable8",
                                         "Variable9",
                                         "Variable10",
                                         "Variable11",
                                         "Variable12"))

```

A new folder will be created in your R directory;

- Folder name format is CutFlow_SystemData_Number
- Three folders are contained within;
 - 0 - A copy of all datafiles fed into CutFlow, for record keeping
 - 1 - A copy of all cutpoint data, including a pdf list of cutpoints
 - 2 - A copy of all datasets, newly coded

SurvFlow

SurvFlow consists of modules. Currently there are two, SurvBase and SurvFacet

- SurvBase generates KM plots per variable, and provides x-year survival, logrank, hazard ratio, case numbers and event numbers
- SurvFacet runs SurvBase per level of each provided facet, such as TNM stage or MSI status
- SurvFacet is currently disabled but will be enabled again in the near future

Arrange your data to meet the following criteria;

- Carry out any exclusions
- Include an identifier variable
- Your status variables needs to be coded as 0 and 1, 0 for no event, 1 for an event
- Time variables must be included as a continuous variable
- There is no limit on the number of variables you wish to generate survival output for
- Save your dataset as a CSV file

To run SurvFlow, simply fill in the blanks, as in the example below;

- Data is a Coded dataset in csv format. If using CutFlow, simply use the produced coded dataset

- Variables is a list of variables (coded as 0 and 1) for analysis
- LegendLabels are optional labels for your legends. Default is the value of the level (0 and 1)
- Identifier is an identifier variable for cases
- PlotTitles are optional plot titles. Default is variable name
- SurvivalStatus are status variables (coded as 0 and 1)
 - Must have the same number of elements as the SurvivalTime variable
- SurvivalTime are survival time variables - continuous
 - Must have the same number of elements as the SurvivalStatus variable
- SurvivalTimeUnit is the unit of time for survival time
- xYearSurvivalVar is the number of years to be used to calculate 'X' years survival. Default = 5
- SurvBase takes TRUE or FALSE. Toggle to activate base survival analysis.

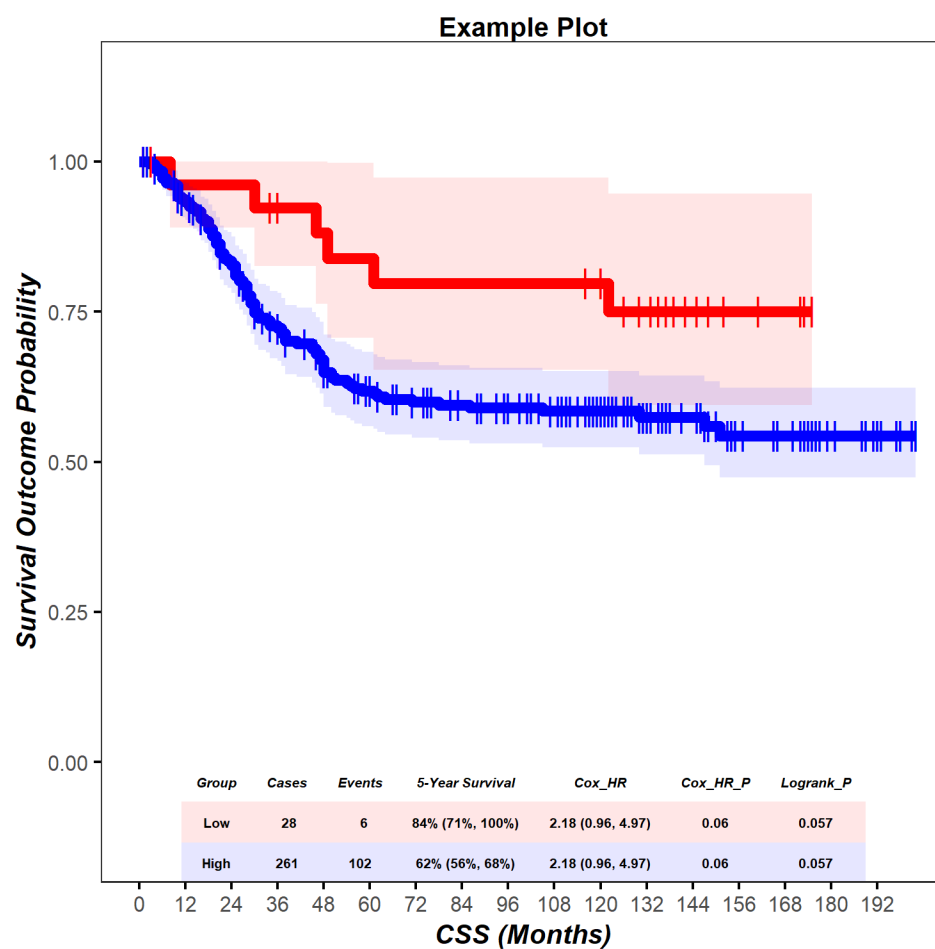
```
Data <- read.csv(file.choose(), fileEncoding = 'UTF-8-BOM')
```

```
SurvFlow(
  Data,
  Variables = c(
    "Variable1",
    "Variable2",
    "Variable3",
    "Variable4",
    "Variable5",
    "Variable6",
    "Variable7",
    "Variable8",
    "Variable9",
    "Variable10",
    "Variable11",
    "Variable12"
  ),
  LegendLabels = c("Low", "High"),
  Identifier = "Identifier",
  PlotTitles = c(
    "Title1",
    "Title2",
    "Title3",
    "Title4",
    "Title5",
    "Title6",
    "Title7",
    "Title8",
    "Title9",
    "Title10",
    "Title11",
    "Title12"
  ),
  SurvivalStatus = c("Status1", "Status2", "Status3", "Status4"),
  SurvivalTime = c("Time1", "Time1", "Time2", "Time2"),
  xYearSurvivalVar = 5,
  SurvivalTimeUnit = "Months",
```

```
SurvBase = TRUE
)
```

A new folder will be created in your R directory;

- Folder name format is SurvFlow_Filename_SystemData_Number
- Inside is a folder per SurvFlow module, for example BaseSurv
 - At the next level is a folder per survival status/time pair, containing the survival plots
- The plot can be seen as below;



CompFlow

CompFlow compares two sets of measurements and outputs;

- Scatterplot
- Bland-Altman plot

- % of points on the Bland-Altman which fall outside of confidence limits
- ICC value
- A quantification of proportional bias (Pearson) - only if your data meet assumptions for Pearson

Arrange your data to meet the following criteria;

- 2 columns of equal length

To run CompFlow, simply fill in the blanks, as in the example below;

- Data is a CSV file containing your two columns of data
- ICCModel - must be 'oneway' or 'twoway'. See '(A) "Model" Selection' in ICC reference
- ICCType - must be 'consistency' or 'agreement'. See '(C) "Definition" Selection' in ICC reference
- ICCUnit - must be 'single' or 'average'. See '(B) "Type" Selection' in ICC reference
- Density - must be 'Yes', 'No'. Density is 'No' by default. This adds density onto your plots.

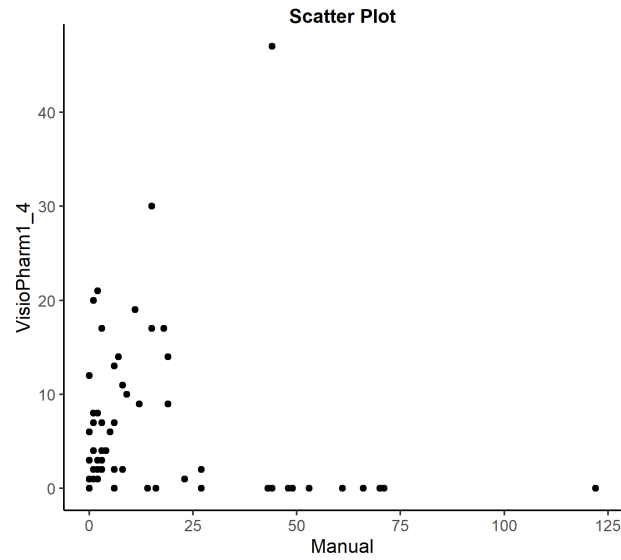
ICC reference

```
Data <- read.csv(file.choose(), fileEncoding = 'UTF-8-BOM')

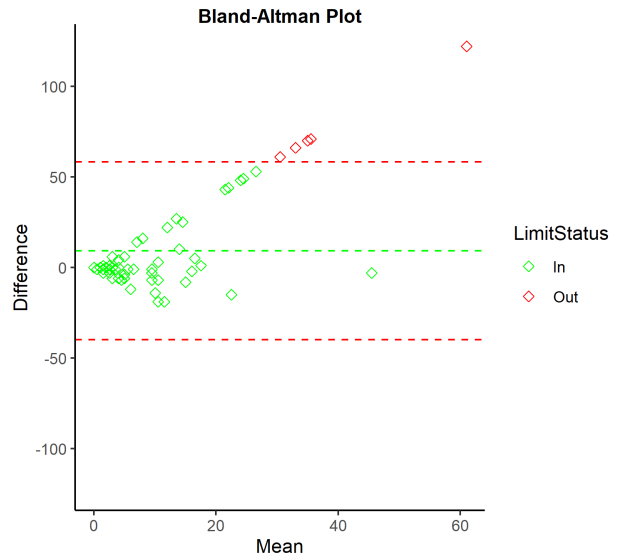
CompFlow(Data, ICCModel = "twoway", ICCType = "agreement", ICCUnit = "single", Density = "Yes")
```

A new folder will be created in your R directory;

- Folder name format is CompFlow_SystemData_Number
- Inside are three folders
 - 0.OriginalData contains your input file for record keeping
 - 1.ConfiguredData contains records of data configurations for record keeping
 - 2.Plots contains your output
- The plots can be seen as below;

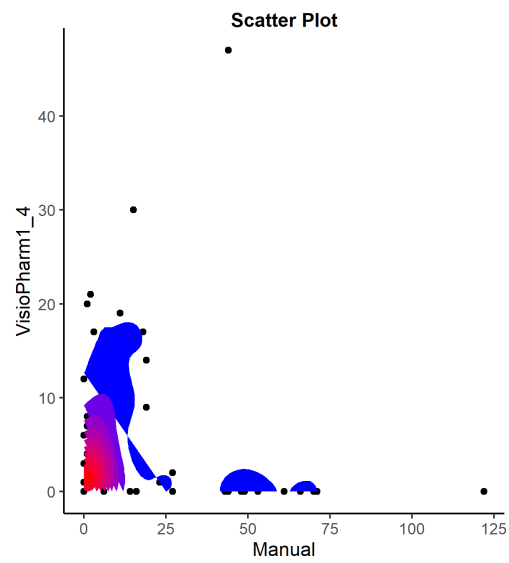


6.8% of points outside of confidence limits

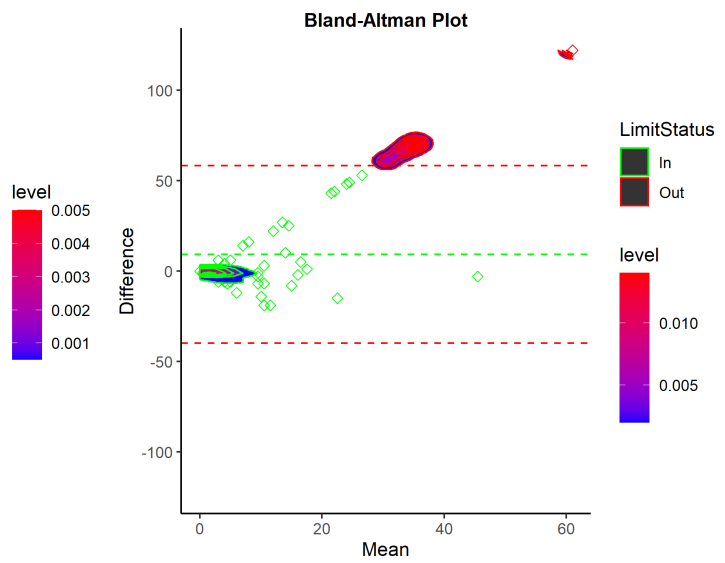


ICCC = 0.461

Proportional bias (Pearson) - p = NA



6.8% of points outside of confidence limits



ICCC = 0.461

Proportional bias (Pearson) - p = NA