

# Package ‘CompensAID’

January 22, 2026

**Type** Package

**Title** Automated detection tool for spillover errors

**Version** 0.99.6

**Description** The CompensAID is an automated quality control tool, which determines for each marker combination in the FCS file, whether there a potential presence of reference errors. Such reference errors, which represent themselves in the form of skewed populations, are detected by integrating the Secondary Stain Index (SSI) score. Marker combinations with an SSI  $< 1$  are flagged by CompensAID.

**URL** <https://github.com/Olsman/CompensAID>

**BugReports** <https://github.com/Olsman/CompensAID/issues>

**License** GPL ( $\geq 3$ )

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Imports** checkmate, dplyr, flowCore, flowDensity, ggcyto, ggplot2 ( $\geq 3.5.2$ ), methods, ParallelLogger, reshape2, rlang, stats, tibble, tidyr, utils

**biocViews** FlowCytometry, QualityControl, Preprocessing

**Suggests** knitr, rmarkdown, testthat ( $\geq 3.0.0$ ), BiocStyle

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**git\_url** <https://git.bioconductor.org/packages/CompensAID>

**git\_branch** devel

**git\_last\_commit** 68496b8

**git\_last\_commit\_date** 2026-01-12

**Repository** Bioconductor 3.23

**Date/Publication** 2026-01-22

**Author** Rosan Olsman [aut, cre] (ORCID:

[<https://orcid.org/0000-0001-5070-0476>](https://orcid.org/0000-0001-5070-0476)),

Sarah Bonte [aut] (ORCID: [<https://orcid.org/0000-0002-0637-4893>](https://orcid.org/0000-0002-0637-4893)),

Mattias Hofmans [aut] (ORCID: <<https://orcid.org/0000-0003-2061-7617>>),  
 Malicorne Buysse [aut],  
 Katrien Van der Borcht [aut] (ORCID:  
 <<https://orcid.org/0000-0003-1635-4359>>),  
 Yvan Saeys [aut] (ORCID: <<https://orcid.org/0000-0002-0415-1506>>),  
 Vincent van der Velden [aut] (ORCID:  
 <<https://orcid.org/0000-0001-9457-3763>>),  
 Sofie Van Gassen [aut] (ORCID: <<https://orcid.org/0000-0002-7119-5330>>)

**Maintainer** Rosan Olsman <[r.olsman@erasmusmc.nl](mailto:r.olsman@erasmusmc.nl)>

## Contents

|                      |          |
|----------------------|----------|
| CompensAID . . . . . | 2        |
| PlotDotSSI . . . . . | 3        |
| PlotMatrix . . . . . | 4        |
| <b>Index</b>         | <b>5</b> |

---

CompensAID

*CompensAID*

---

## Description

Run CompensAID to assess the potential presence of reference errors within the FCS file. The outcome is a list which contains an SSI matrix. The matrix quickly visualizes the marker combinations for each marker combinations. Additional information regarding the number of events per segments, which segments were merged, and the corresponding SSI values can be found in the Info dataframe.

## Usage

```
CompensAID(ff, segment.value = 4, events.value = 50)
```

## Arguments

|                            |                                                                                            |
|----------------------------|--------------------------------------------------------------------------------------------|
| <code>ff</code>            | (FlowFrame): FlowFrame containing the expression matrix, channel names, and marker names.  |
| <code>segment.value</code> | (numerical): Numerical value of the number of segments in the primary positive population. |
| <code>events.value</code>  | (numerical): Numerical value of the minimum number of events per population/segment.       |

## Value

(list) Returns a list containing the full SSI output and SSI matrix

**Examples**

```
# Import FCS file
file <- flowCore::read.FCS(system.file("extdata", "68983.fcs", package = "CompensAID"))

# Run CompensAID tool
compensAID.res <- CompensAID(ff = file)
```

PlotDotSSI

*Plot dot plot of Secondary Stain Index scores***Description**

This function plots the marker combinations of interest. Additionally, the gating, segmentation, and SSI values can be visualized by setting showScore to TRUE.

**Usage**

```
PlotDotSSI(output, og, primary, secondary, showScores = TRUE)
```

**Arguments**

|            |                                                                                           |
|------------|-------------------------------------------------------------------------------------------|
| output     | (matrix): Matrix containing the SSI output from the CompensAID tool.                      |
| og         | (flowFrame): FlowFrame containing the expression matrix, channel names, and marker names. |
| primary    | (character): Primary marker (x-axis)                                                      |
| secondary  | (character): Secondary marker (y-axis)                                                    |
| showScores | (boolean): Boolean variable determining if scores and gating should be visualized.        |

**Value**

(ggplot2) Returns figure containing a dot plot of the CompensAID output.

**See Also**

[CompensAID](#)

**Examples**

```
# Import FCS file
file <- flowCore::read.FCS(system.file("extdata", "68983.fcs", package = "CompensAID"))

# Run compensAID tool
compensAID.res <- CompensAID(ff = file)

# Marker names
```

```

primary.marker <- "CD19"
secondary.marker <- "CD3"

# Plot matrix
figure <- PlotDotSSI(output = compensAID.res,
                     og = file,
                     primary = primary.marker,
                     secondary = secondary.marker,
                     showScores = TRUE)

plot(figure)

```

---

PlotMatrix

*Plot Secondary Stain Index matrix*


---

### Description

This function plot the SSI matrix, quickly allowing the identification of potential reference errors if a SSI < -1 is obtained.

### Usage

```
PlotMatrix(output)
```

### Arguments

output (matrix): Matrix containing the SSI output.

### Value

(ggplot2) Returns figure containing a matrix of the total compensAID output.

### See Also

[CompensAID](#)

### Examples

```

# Import FCS file
file <- flowCore::read.FCS(system.file("extdata", "68983.fcs", package = "CompensAID"))

# Run compensAID tool
compensAID.res <- CompensAID(ff = file)

# Plot matrix
figure <- PlotMatrix(output = compensAID.res)
plot(figure)

```

# Index

CompensAID, [2](#), [3](#), [4](#)

PlotDotSSI, [3](#)

PlotMatrix, [4](#)