# Package 'panp'

November 12, 2026

**Version** 1.82.0

**Title** Presence-Absence Calls from Negative Strand Matching Probesets

**Author** Peter Warren

**Description** A function to make gene presence/absence calls based on distance from negative strand matching probesets (NSMP) which are derived from Affymetrix annotation. PANP is applied after gene expression values are created, and therefore can be used after any preprocessing method such as MAS5 or GCRMA, or PM-only methods like RMA. NSMP sets have been established for the HGU133A and HGU133-Plus-2.0 chipsets to date.

**Maintainer** Peter Warren <peter.warren@verizon.net>

**Depends** R (>= 2.10), affy (>= 1.23.4), Biobase (>= 2.5.5)

**Imports** Biobase, methods, stats, utils

**Suggests** gcrma

**License** GPL (>= 2)

**biocViews** Infrastructure

**git_url** https://git.bioconductor.org/packages/panp

**git_branch** RELEASE_3_23

**git_last_commit** 7255748

**git_last_commit_date** 2026-04-28

**Repository** Bioconductor 3.23

**Date/Publication** 2026-11-11

## Contents


gcrma.ExpressionSet . . . . . 2
NSMPnames.hgu133a . . . . . 2
NSMPnames.hgu133plus2 . . . . . 3
pa.calls . . . . . 3

**Index** **5**

---

`gcrma.ExpressionSet` *ExpressionSet resulting from gcrma processing of 3 HG-U133A chips*

---

## Description

An ExpressionSet object resulting from gcrma processing of three HG-U133A chips from the 28-chip affyCOMP II dataset, derived from the AffyMetrix Latin Squares dataset. The gcrma.ExpressionSet can be used as example input for panp's pa.calls() function, and it is used for this purpose in the panp vignette.

## Usage

```
data(gcrma.ExpressionSet)
```

## Format

An ExpressionSet object comprising three gcrma-processed HG-U133A chips

## Source

The 3 HG-U133A chips are the first three of affyCOMP II Latin Squares set

---

`NSMPnames.hgu133a` *Negative Strand Matching Probeset (NSMP) names for HG-U133A chip type*

---

## Description

This list of NSMP probeset names is used by pa.calls() in calculating the NSMP expression distribution. It is specific to the chip type; the correct version is automatically loaded for the detected chip type, if supported. The user does not load or use this data directly.

## Usage

```
data(NSMPnames.hgu133a)
```

## Format

A vector containing probeset names

## Source

October 2004 AffyMetrix chip annotation files, filtered, with outliers removed

## NSMPnames.hgu133plus2 *Negative Strand Matching Probeset (NSMP) names for HG-U133 Plus 2.0 chip type*

### Description

This list of NSMP probeset names is used by pa.calls() in calculating the NSMP expression distribution. It is specific to the chip type; the correct version is automatically loaded for the detected chip type, if supported. The user does not load or use this data directly.

### Usage

```
data(NSMPnames.hgu133plus2)
```

### Format

A vector containing probeset names

### Source

October 2004 AffyMetrix chip annotation files, filtered, with outliers removed

## pa.calls *Presence-Absence Calls from Negative Strand Matching Probesets*

### Description

Function to make gene presence/absence calls based on distance from empirical distribution of chip-specific negative strand matching probesets (NSMP).

### Usage

```
pa.calls(object, looseCutoff = 0.02, tightCutoff = 0.01, verbose = FALSE)
```

### Arguments

| | |
|---|---|
| object | an ExpressionSet object (result of running expression-generating function, like expresso(), rma(), mas5(), etc.) Currently, this must be of chip type HGU133A or HGU133 Plus 2.0 |
| looseCutoff | the larger P-value cutoff (see details) |
| tightCutoff | the smaller, more strict P-value cutoff |
| verbose | logical. If 'TRUE' detailed progress messages are reported. |

## Details

The function calculates a matrix of P-values for the expression values in the input ExpressionSet. P-values are calculated based on the empirical survivor function (1-CDF) of the set of negative probesets identified by Affymetrix as negative strand matching probesets (NSMP) with no cross hybridization. These probesets are therefore assumed to show nothing but background/machine noise plus some occasional non-specific binding. The P-value returned for any probeset expression value in ExpressionSet is the value of the NSMP survivor function for that expression level.

Presence/Absence calls are derived by applying the two cutoff values to the matrix of P-values for all genes in the ExpressionSet, as follows:

**Present ('P')** P-values <= tightCutoff

**Absent ('A')** P-values > looseCutoff

**Marginal ('M')** P-values between tightCutoff and looseCutoff

## Value

| | |
|---|---|
| `list` | a new list containing two matrices: Pcalls and Pvals, as follows: |
| `Pcalls` | a matrix of Presence (P), Marginal (M), Absent (A) indicators |
| `Pvals` | a matrix of P-values. Each data point is the P-value for the expr at the same x, y coordinates. |

## Note

NSMP sets have been established for the HGU133A and HGU133-Plus-2.0 chipsets to date. Hence only these two are currently supported by PANP.

## Author(s)

Peter Warren

## References

Warren, P., Bienkowska, J., Martini, P., Jackson, J., and Taylor, D., PANP - a New Method of Gene Detection on Oligonucleotide Expression Arrays (2007), in preparation

## Examples

```
## Load example ExpressionSet
data(gcrma.ExpressionSet)

## Generate Pvals and Pcalls matrices from ExpressionSet, using default cutoffs
PA <- pa.calls(gcrma.ExpressionSet)

## to access the Pcalls and Pvals:
myPcalls <- PA$Pcalls
myPvals <-  PA$Pvals
```

# Index

∗ **datasets**
gcrma.ExpressionSet, 2
NSMPnames.hgu133a, 2
NSMPnames.hgu133plus2, 3
∗ **manip**
pa.calls, 3

gcrma.ExpressionSet, 2

NSMPnames.hgu133a, 2
NSMPnames.hgu133plus2, 3

pa.calls, 3