# PSCAN

# GENERAL INFORMATION

PSCAN package has the main PSCAN function that implements protein-structure-guided scan (PSCAN) methods for detecting gene-level associations and signal variants. PSCAN methods leverage the tendency of functional variants to cluster in 3D protein space. PSCAN methods are built upon flexibly shaped spatial scan statistics, with scan windows adaptively defined to accommodate diverse topologies of variant positions in protein space. PSCAN performs fast gene-level association tests by combining SNP-set-based testing p-values across windows using the Cauchy method. In addition, PSCAN implements an efficient search algorithm for the detection of multiple signal regions in protein space. The details are described in Tang et al. (2020, Submitted).

In this vignette, we will give examples to demonstrate how to use PSCAN.

# EXAMPLE

The PSCAN function takes variant-level score statistics from the association analysis on a gene, its covariance matrix, and the minor allele counts as required input. These information can be routinly generated from the software programs for SNP-set association analyses such as RAREMETALWORKER, MetaSKAT, seqMeta, SCORE-Seq.

```
# load example data from PSCAN package
data(one.gene)
U = one.gene$U
V = one.gene$V
MAC = one.gene$MAC
weight = one.gene$weight
```

The first few elements of the vector of score statistics U:

```
U[1:5]
```

```
## 1:59811928 1:59811930 1:59811954 1:59811969 1:59811989
##  1.2091128  0.1874159  0.5788166 -3.9812787  6.2168732
```

The corresponding covariance matrix V:

```
V[1:5,1:5]
```

```
##          [,1]    [,2]   [,3]        [,4]        [,5]
## [1,] 3.342603 0.00000 0.0000  0.00000000  0.00000000
## [2,] 0.000000 5.80427 0.0000  0.00000000  0.00000000
## [3,] 0.000000 0.00000 3.5139  0.00000000  0.00000000
## [4,] 0.000000 0.00000 0.0000  4.37972354 -0.01173339
## [5,] 0.000000 0.00000 0.0000 -0.01173339  4.33809225
```

The first few elements of the vector of minor allele counts (MAC):

```
MAC[1:5]
```

```
## [1] 20 14 17 28 26
```

PSCAN perform SNP-set association analysis within each scan window. One may want to give different weights to different SNPs, say, according to their minor allele frequencies. If users don't specify weight, the analysis will use flat weight (i.e. all variants have the same unit weight). The first few elements of the vector of weights for this example:

```
weight[1:5]
```

```
## [1] 87.08042 87.08042 87.08042 87.08042 87.08042
```

It is required that the name of the score statistic vector is the SNP ID with chr:position format. The other inputs are not reqired to have names but the order of the SNPs should the same as the score statistic vector.

PSCAN incoporats the 3D coordinates of variants in the protein space as the internal database. Therefore, users do not need to provide this information as input. If users are interested in knowing the coordinates we used in the analysis, this information can be output by setting details=T. Please see details below in the section "Signal Cluster and Variant Detection".

## Gene-level Association Tests

We can suppress the signal variant detection and only run the gene-level association tests through setting FWER=NULL. This is useful when performing the whole-genome scan as it will save a lot of time. To test the mean of the genetic effect, set type="mean".

```
PSCAN(type="mean", U=U, V=V, MAC=MAC, weight=weight, FWER=NULL)
```

```
$pscan.pval
[1] 1.519986e-08
```

If the signal cluster contains variants of both positive and negative effects and/or many neutral variants, testing the variance would be more powerful than testing the mean. To test the variance, set type="variance"

```
PSCAN(type="variance", U=U, V=V, MAC=MAC, weight=weight, FWER=NULL)
```

```
$pscan.pval
[1] 1.444214e-08
```

By default, the gene-level association p-value is obtained using the Cauchy method to combine p-values across windows. The Cauchy method is very fast and suitable for genome-wide association studies. Previous research has shown that the Cauchy method can properly control the type I error regardless of correlations of individual tests when p-value is small ($< 10^{-4}$). However, the Cauchy method can have a slight ination in the type I error for large p-values. Therefore, in the candidate gene analysis where only a handful of genes are tested, using Monte Carlo simulation to evaluate the significance of minimum p-value (minP) scan statistic is a more robust alternative. Users can set p.comb="minP" to perform minP gene-level test. It could be much slower than the default (p.comb="Cauchy"), especially when you need a large number of Monte Carlo simulations to obtain accurate estimates of small p-values.

## Signal Cluster and Variant Detection

For the known trait-associated genes or the genes that have already been identified as significant in the whole-genome scan. We may be interested to further identify the signal cluster/regions and the associated variants that live in the regions. The significance threshold is determined by the specified FWER.

```
PSCAN(type="mean", U=U, V=V, MAC=MAC, weight=weight, FWER=0.05)
```

```
$pscan.pval
[1] 1.519986e-08

$signal
$signal[[1]]
[1] "1:60223655"
```

```
$signal[[2]]
[1] "1:59844459" "1:59922718"
```

For this example, PSCAN identified two sginal regions on the protein space when we test the mean effects. One reigon contain one SNP and the other contains two SNPs. By setting details=TRUE, the function will output the p-values associated with these two signal regions, the 3D coordinates for all the variants in the gene, and whether the protein structure is experimentally determined ("PDB") or computationally predicted ("Modbase").

```
PSCAN(type="mean", U=U, V=V, MAC=MAC, weight=weight, FWER=0.05, details=TRUE)
```

to obtain

```
$pscan.pval
[1] 1.519986e-08

$signal
$signal[[1]]
[1] "1:60223655"

$signal[[2]]
[1] "1:59844459" "1:59922718"


$signal.pval
[1] 1.876524e-10 6.953743e-04

$structure.type
[1] "Modbase"

$structure.coord
$structure.coord$ENSP00000360262.4_1
                   x          y          z
1:59811928   8.291399 -29.627802 -11.176199
1:59811930   6.044636 -26.294092 -14.805182
1:59811954  -6.045200 -22.103601 -26.434101
1:59811969  -1.795222 -28.656002 -33.824776
1:59811989  -7.056112 -30.640778 -43.326447
1:59812068   5.122889 -31.334221 -33.507774
1:59844459   0.128455 -40.130184 -22.448273
1:59844470  -9.636000 -37.483856 -14.070143
1:59844503  -4.587714 -48.364429 -16.135000
1:59922641 -15.895167 -39.558002 -19.921165
1:59922649 -15.470284 -40.471859 -25.216429
1:59922692  -7.872334 -44.328671 -38.820835
1:59922718   2.875125 -43.610874 -31.384626
1:59978024  -7.880000 -46.939377  -1.882000
1:59978048  -6.077375 -39.611252   1.719250
1:60019834 -42.799835 -37.502167  -8.521167
1:60019840 -39.209000 -37.480000 -13.891749
1:60019853 -36.625713 -33.758572 -23.690287
1:60019880 -14.494429 -21.352999 -14.322715
1:60019891 -24.432199 -25.751699 -21.676601
1:60019898 -27.657751 -33.301750 -23.117500
1:60073482 -32.267776 -43.016331 -25.584110
1:60073515 -22.770430 -37.275856 -30.436571
```

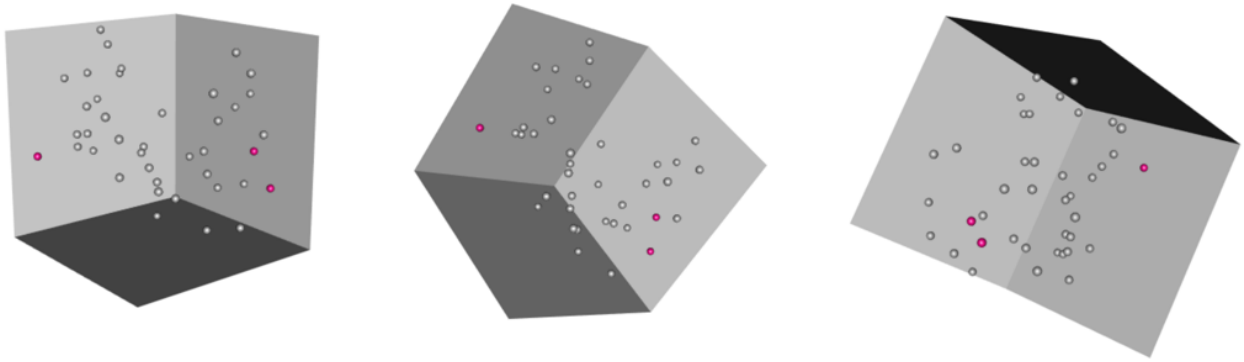
```
1:60073550 -25.523876 -33.582874 -26.836252
1:60073563 -20.143223 -20.916000 -21.878223
1:60091687 -12.974800  -0.541200  -7.677400
1:60091722 -21.369183  -4.202727   4.571818
1:60103905 -21.991663  -8.283443   1.497222
1:60103910 -18.187876  -8.753500  -4.763125
1:60103931 -23.722500  -1.025700  -6.266300
1:60103937 -31.405499  -2.040875  -6.597000
1:60103985 -24.386202  -4.087300 -25.599697
1:60125926 -18.634624   2.194125 -14.075500
1:60133064 -34.352600 -16.893002 -17.228800
1:60133066 -35.095875 -13.330500 -17.985750
1:60139785 -22.990499 -51.029747  -3.865000
1:60223637 -38.102573 -22.674143  -9.281285
1:60223655 -44.010555  -8.904111 -22.789333
1:60228180 -32.850002  -9.902083 -22.045500
1:60228246  -7.148111  -9.340556 -45.130890
```

For this gene, we do not have experimentally determined protein structure availiable, therefore, the function automatically switch to use coordiantes from the computationally predicted structure ($structure.type is "Modbase").

Because of the dynamic nature of proteins and difficulties in elucidating structures of larger proteins, some regions of proteins may be missing in available structural models or broken up into fragments across multiple models. In the case of multiple fragments, there would be multiple matrix elements in the list $structure.coord, each for one fragment. For this example gene, there is one fragment. There are a total of 44 variants in this gene but the coordinates are only for 40 variants. This is because the 4 variants do not map to the fragment of the protein that has structural information. These 4 variants will still be included in the PSCAN analysis. The strategy to handle unmapped variants and multiple structures are explained in Tang et al. (2020). For the gene that have no protein structure, PSCAN gene-level tests simply reduce to regular gene-level tests (i.e. burden-type test if type="mean" ; SKAT-type test if type="variance" )

The variant positions on the protein space can be interativelly visualized by settting plot3D=TRUE

```
PSCAN(type="mean", U=U, V=V, MAC=MAC, weight=weight, FWER=0.05, details=TRUE, plot3D=TRUE)
```



The viewing angle can be adjusted. Above shows graphs from three different angles.

By default, we assume the signal regions are not overlapping with each other. Specifically, we first pick candidate signal regions as windows with p-values less than the significance threshold. We select a region that has the smallest p-value among all the candidate regions and remove regions that overlap with the selected region from the pool of candidate regions.

Alternatively, we may only removes windows that overlap by more than the pre-specified overlap fraction $f$, where $0 \leq f \leq 1$ (see Algorithm S1 in the Supplementary File of Tang et al. 2020). When $f = 0$, this

algorithm basically reduce to the default approach. When $f = 1$, this algorithm essentially keeps every region passing the significance threshold as the detected signal regions.

For this example gene, if we use mean-base test (type="mean"), we always get the same signal regions regarless of $f$. If we use variance-based test (type="variance"), we only identify one region with one variant if we set $f = 0$.

```
PSCAN(type="variance", U=U, V=V, MAC=MAC, weight=weight, FWER=0.05, f=0)
```

```
$pscan.pval
[1] 1.444214e-08

$signal
$signal[[1]]
[1] "1:60223655"
```

If we set $f = 0.5$, we identify multiple overlapping signal regions.

```
PSCAN(type="variance", U=U, V=V, MAC=MAC, weight=weight, FWER=0.05, f=0.5)
```

```
$pscan.pval
[1] 1.444214e-08

$signal
$signal[[1]]
[1] "1:60223655"

$signal[[2]]
[1] "1:60103985" "1:60133064" "1:60133066" "1:60223637" "1:60223655" "1:60228180"

$signal[[3]]
 [1] "1:59811954" "1:59811969" "1:59811989" "1:59812068" "1:59844459" "1:59844470"
 [7] "1:59844503" "1:59922641" "1:59922649" "1:59922692" "1:59922718" "1:60019834"
[13] "1:60019840" "1:60019853" "1:60019880" "1:60019891" "1:60019898" "1:60073482"
[19] "1:60073515" "1:60073550" "1:60073563" "1:60103985" "1:60133064" "1:60133066"
[25] "1:60223637" "1:60223655" "1:60228180"
```

# References

Tang ZZ, Sliwoski GR, Chen G, Jin B, Bush WS, Li B, and Capra JA. (2020). Spatial scan tests guided by protein structures improve complex disease gene discovery and signal variant detection. Genome Biology.