

# Package ‘KnockoffTrio’

July 21, 2025

**Title** GWAS with Trio and Duo Data using Knockoff Statistics for FDR Control

**Version** 1.1.0

**Description** Identification of putative causal variants in genome-wide association studies with trio and duo families. The package calculates the W feature statistics from KnockoffTrio and p-values from the family-based association test (FBAT) using trio and/or duo data. Compared to previous versions, a significant improvement has been made in Version 1.1.0 to allow the package to be applied not only to trio families but also to duo families. The package implements the methods in the paper: ``Yang, Y., Wang, C., Liu, L., Buxbaum, J., He, Z., & Ionita-Laza, I. (2022). KnockoffTrio: A knockoff framework for the identification of putative causal variants in genome-wide association studies with trio design. The American Journal of Human Genetics, 109(10), 1761-1776."

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.2

**Depends** R (>= 2.10)

**NeedsCompilation** no

**Author** Yi Yang [aut, cre] (ORCID: <<https://orcid.org/0000-0002-1471-4026>>)

**Maintainer** Yi Yang <yi.yang@cityu.edu.hk>

**Repository** CRAN

**Date/Publication** 2025-01-23 09:40:06 UTC

## Contents

causal_loci . . . . .	2
create_knockoff . . . . .	3
KnockoffTrio . . . . .	4
KnockoffTrio.example . . . . .	7
meta_analysis . . . . .	7

<b>Index</b>	<b>9</b>
--------------	----------

---

causal_loci	<i>Identification of putative causal loci</i>
-------------	---

---

## Description

Identification of putative causal loci using KnockoffTrio's feature statistics

## Usage

```
causal_loci(window, M = 10, fdr = 0.1)
```

## Arguments

window	The result window from function KnockoffTrio. If there are multiple result windows (e.g., when you analyze multiple regions in the genome), please use rbind to combine all the windows before running causal_loci.
M	A positive integer for the number of knockoffs. The default is 10.
fdr	A real number in a range of (0,1) indicating the target FDR level. The default is 0.1. Use 0.2 for a more lenient FDR control.

## Value

A list that contains the following elements for claiming significance using knockoff statistics. The result window also contains FBAT p-values and ACAT-combined p-values, which can be used for claiming significance in addition to knockoff statistics. If p-values are used, Bonferroni correction is usually necessary to adjust for multiple testing for controlling the family-wise error rate - see examples below.

**window** A data frame for an updated result window that includes an extra column for KnockoffTrio's Q-values. A locus with a Q-value  $\leq$  the target FDR level, i.e., `window$q<=fdr`, is considered as putative causal at the target FDR.

**thr.w** A positive real number indicating the significance threshold for KnockoffTrio's feature statistics. A locus with a feature statistic  $\geq$  thr.w, i.e., `window$w>=thr.w` is considered as putative causal at the target FDR. The loci selected by `window$w>=thr.w` are equivalent to those by `window$q<=fdr`. No loci are selected at the target FDR level if thr.w=Inf.

## Examples

```
data(KnockoffTrio.example)
knockoff<-create_knockoff(trio.hap=KnockoffTrio.example$trio.hap,
  duo.hap=KnockoffTrio.example$duo.hap, pos=KnockoffTrio.example$pos, M=10)
window<-KnockoffTrio(trio=KnockoffTrio.example$trio, trio.ko=knockoff$trio.ko,
  duo=knockoff$duo, duo.ko=knockoff$duo.ko, pos=KnockoffTrio.example$pos)

#Identification of significant loci using KnockoffTrio's feature statistics (W or Q)
#at a target FDR
target_fdr<-0.1
result<-causal_loci(window,M=10,fdr=target_fdr)
```

```

sig_loci_by_w_index<-which(result$window$w>=result$thr.w)
sig_loci_by_q_index<-which(result$window$q<=target_fdr)

#Identification of significant loci using FBAT p-values with Bonferroni correction
#for controlling the family-wise error rate at 0.05
sig_loci_by_p_fbat_index<-which(window$p.burden<0.05/nrow(window))

#Identification of significant loci using ACAT p-values with Bonferroni correction
#for controlling the family-wise error rate at 0.05
sig_loci_by_p_acat_index<-which(window$p<0.05/nrow(window))

```

---

create_knockoff	<i>Create knockoff genotype data</i>
-----------------	--------------------------------------

---

## Description

Create knockoff genotype data using phased haplotype data.

## Usage

```

create_knockoff(
  trio.hap = NULL,
  duo.hap = NULL,
  pos,
  M = 10,
  maxcor = 0.7,
  maxbp = 80000,
  phasing.dad = NULL,
  phasing.mom = NULL,
  seed = 100
)

```

## Arguments

trio.hap	A $6n \times p$ matrix for trio haplotype data, in which $n$ is the number of trios and $p$ is the number of variants. Each trio must consist of father, mother, and offspring (in this order). The haplotypes must be coded as 0 or 1. Missing haplotypes are not allowed.
duo.hap	A $4m \times p$ matrix for duo haplotype data, in which $m$ is the number of duos and $p$ is the number of variants. Each duo must consist of a single parent and offspring (in this order). The haplotypes must be coded as 0 or 1. Missing haplotypes are not allowed.
pos	A numeric vector of length $p$ for the position of $p$ variants.
M	A positive integer for the number of knockoffs. The default is 10.
maxcor	A real number in a range of $[0,1]$ for hierarchical clustering of neighboring variants used to generate knockoff parents. The default is 0.7.

maxbp	A positive integer for the size of neighboring base pairs used to generate knock-off parents. The default is 80000.
phasing.dad	A numeric vector of length n that contains 1 or 2 to indicate which paternal haplotype was transmitted to offspring in each trio. If NULL, the function will calculate the phasing information based on the input trio haplotype matrix.
phasing.mom	A numeric vector of length n that contains 1 or 2 to indicate which maternal haplotype was transmitted to offspring in each trio. If NULL, the function will calculate the phasing information based on the input trio haplotype matrix.
seed	An integer for the random seed used for knockoff generation.

### Value

A list that contains:

**trio.ko** A  $3n \times p \times M$  array for knockoff trio genotype data if trio.hap is provided.

**duo.ko** A  $3m \times p \times M$  array for knockoff duo genotype data if duo.hap is provided.

**duo** A  $3m \times p$  matrix for duo genotype data if duo.hap is provided.

### Examples

```
data(KnockoffTrio.example)
knockoff<-create_knockoff(trio.hap=KnockoffTrio.example$trio.hap,
                        duo.hap=KnockoffTrio.example$duo.hap, pos=KnockoffTrio.example$pos, M=10)
```

---

KnockoffTrio

---

*Calculate KnockoffTrio's feature statistics*


---

### Description

Calculate KnockoffTrio's feature statistics and FBAT statistics using original and knockoff genotype data.

### Usage

```
KnockoffTrio(
  trio,
  trio.ko = NULL,
  duo = NULL,
  duo.ko = NULL,
  pos,
  start = NULL,
  end = NULL,
  size = c(1, 1000, 5000, 10000, 20000, 50000),
  p_value_only = FALSE,
  adjust_for_cov = FALSE,
  y = NULL,
```

```

    chr = "1",
    xchr = FALSE,
    sex = NULL
)

```

## Arguments

<code>trio</code>	A $3n \times p$ matrix for the trio genotype data, in which $n$ is the number of trios and $p$ is the number of variants. Each trio must consist of father, mother, and offspring (in this order). The genotypes must be coded as 0, 1, or 2. Missing genotypes are not allowed.
<code>trio.ko</code>	A $3n \times p \times M$ array for the knockoff trio genotype data created by function <code>create_knockoff</code> . $M$ is the number of knockoffs.
<code>duo</code>	A $3m \times p$ matrix for the duo genotype data created by function <code>create_knockoff</code> , in which $m$ is the number of duos and $p$ is the number of variants. Please do not use the original $2m \times p$ duo genotype matrix.
<code>duo.ko</code>	A $3m \times p \times M$ array for the knockoff duo genotype data created by function <code>create_knockoff</code> . $M$ is the number of knockoffs.
<code>pos</code>	A numeric vector of length $p$ for the position of $p$ variants.
<code>start</code>	An integer for the first position of sliding windows. If <code>NULL</code> , <code>start=min(pos)</code> . Only used if you would like to use the same starting position for different cohorts/analyses.
<code>end</code>	An integer for the last position of sliding windows. If <code>NULL</code> , <code>end=max(pos)</code> . Only used if you would like to use the same ending position for different cohorts/analyses.
<code>size</code>	A numeric vector for the size(s) of sliding windows when scanning the genome.
<code>p_value_only</code>	A logical value indicating whether to perform the knockoff analysis. When <code>p_value_only</code> is <code>TRUE</code> , only the ACAT-combined p-values are to be calculated for each window. When <code>p_value_only</code> is <code>FALSE</code> , <code>trio.ko</code> or <code>duo.ko</code> is required and KnockoffTrio's feature statistics are to be calculated for each window in addition to the p-values.
<code>adjust_for_cov</code>	A logical value indicating whether to adjust for covariates. When <code>adjust_for_cov</code> is <code>TRUE</code> , <code>y</code> is required.
<code>y</code>	A numeric vector of length $n$ for the residual $Y - \hat{Y}$ . $\hat{Y}$ is the predicted value from the regression model in which the quantitative trait $Y$ is regressed on the covariates. If $Y$ is dichotomous, you may treat $Y$ as quantitative when applying the regression model.
<code>chr</code>	A character for the name of the chromosome, e.g., "1", "2", ..., "22", and "X".
<code>xchr</code>	A logical value indicating whether the analysis is for the X chromosome. When <code>xchr</code> is <code>TRUE</code> , the analysis is for the X chromosome and sex is required. When <code>xchr</code> is <code>FALSE</code> , the analysis is for the autosomes. The default is <code>FALSE</code> .
<code>sex</code>	A numeric vector of length $n$ for the sex of offspring. 0s indicate females and 1s indicate males. Sex is required when <code>xchr</code> is <code>TRUE</code> .

## Value

A data frame for analysis results from KnockoffTrio and FBAT. The data frame contains the following columns if `p_value_only` is FALSE:

**chr** The chromosome number.

**start, end** The start and end position of a window.

**actual\_start, actual\_end** The position of the first and last variant in a window.

**n** The number of variants in a window.

**dir** The direction of effect of the most significant variant in a window.

**w** The W knockoff feature statistic for a window. Please use function `causal_loci` to obtain the significance threshold for w at target FDRs.

**p** The ACAT-combined p-value for a window. If a window contains multiple variants (i.e.,  $n > 1$ ), ACAT combines FBAT p-values for each variant and a burden FBAT p-value for all variants in the window. If a window contains only one variant (i.e.,  $n = 1$ ), the ACAT-combined p-value is equivalent to the FBAT p-value for this variant.

**z** The FBAT z-score for a window. If a window contains multiple variants (i.e.,  $n > 1$ ), z is the burden FBAT z-score for all variants in the window. If a window contains only one variant (i.e.,  $n = 1$ ), z is the FBAT z-score for this variant.

**p.burden** The FBAT p-value for a window. If a window contains multiple variants (i.e.,  $n > 1$ ), p.burden is the burden FBAT p-value for all variants in the window. If a window contains only one variant (i.e.,  $n = 1$ ), p.burden is the FBAT p-value for this variant.

**kappa, tau** The two columns are used by function `causal_loci` for knockoff inference.

**p\_1, ..., p\_M** The ACAT-combined p-values for M knockoffs.

**z\_1, ..., z\_M** The FBAT z-scores for M knockoffs.

## Examples

```
data(KnockoffTrio.example)
knockoff<-create_knockoff(trio.hap=KnockoffTrio.example$trio.hap,
                        duo.hap=KnockoffTrio.example$duo.hap, pos=KnockoffTrio.example$pos, M=10)

#Analysis for both trios and duos
window<-KnockoffTrio(trio=KnockoffTrio.example$trio, trio.ko=knockoff$trio.ko,
                    duo=knockoff$duo, duo.ko=knockoff$duo.ko, pos=KnockoffTrio.example$pos)

#Analysis for trios only
window<-KnockoffTrio(trio=KnockoffTrio.example$trio, trio.ko=knockoff$trio.ko,
                    duo=NULL, duo.ko=NULL, pos=KnockoffTrio.example$pos)

#Analysis for duos only
window<-KnockoffTrio(trio=NULL, trio.ko=NULL,
                    duo=knockoff$duo, duo.ko=knockoff$duo.ko, pos=KnockoffTrio.example$pos)
```

---

KnockoffTrio.example    *Example data for KnockoffTrio*

---

### Description

A toy example of haplotype and genotype data for trios and duos.

### Usage

KnockoffTrio.example

### Format

KnockoffTrio.example contains the following items:

**trio** A 9\*5 numeric genotype matrix of 3 trios and 5 variants. Each trio contains 3 rows in the order of father, mother and offspring. Each column represents a variant.

**trio.hap** A 18\*5 numeric haplotype matrix of 3 trios and 5 variants. Each trio contains 6 rows in the order of father, mother and offspring. Each column represents a variant.

**duo.hap** A 12\*5 numeric haplotype matrix of 3 duos and 5 variants. Each duo contains 4 rows in the order of a single parent and offspring. Each column represents a variant.

**pos** A numeric vector of length 5 for the position of 5 variants.

---

meta\_analysis    *Meta-analysis for KnockoffTrio*

---

### Description

Meta-analysis for KnockoffTrio

### Usage

```
meta_analysis(window, n = NA, M = 10)
```

### Arguments

window	A list of windows for the analysis results from different cohorts/studies.
n	A positive integer vector for the number of families in each cohort/study. For weighted meta-analysis, a study's weight is based on the number of families. The default is NA for unweighted meta-analysis.
M	A positive integer for the number of knockoffs. The default is 10.

### Value

A data frame for the meta-analysis results.

**Examples**

```
data(KnockoffTrio.example)
knockoff<-create_knockoff(trio.hap=KnockoffTrio.example$trio.hap,
  duo.hap=KnockoffTrio.example$duo.hap, pos=KnockoffTrio.example$pos, M=10)
window<-KnockoffTrio(trio=KnockoffTrio.example$trio, trio.ko=knockoff$trio.ko,
  duo=knockoff$duo, duo.ko=knockoff$duo.ko, pos=KnockoffTrio.example$pos)
window.list<-list(window,window)
window.meta<-meta_analysis(window.list,M=10)
result<-causal_loci(window.meta,M=10,fdr=0.1)
```



# Index

## \* **datasets**

KnockoffTrio.example, [7](#)

causal\_loci, [2](#)

create\_knockoff, [3](#)

KnockoffTrio, [4](#)

KnockoffTrio.example, [7](#)

meta\_analysis, [7](#)