

# Package ‘RankGWASK’

June 18, 2026

**Type** Package

**Title** Ranked Set Sampling Genome-Wide Association Studies Toolkit

**Version** 0.1.1

**Description** Provides methods for genome-wide association studies (GWAS) using ranked set sampling (RSS) designs. The package includes tools for ranked set sample selection, standard and RSS-based association analyses, simulation of genotype and phenotype data, statistical comparison of RSS and simple random sampling (SRS) approaches, visualization of GWAS results, and power analysis under alternative sampling schemes.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 4.3.0)

**Imports** qqman

**Config/roxygen2/version** 8.0.0

**NeedsCompilation** no

**Author** Khalid Ul Islam Rather [aut, cre]

**Maintainer** Khalid Ul Islam Rather <drkhalidulislam@gmail.com>

**Repository** CRAN

**Date/Publication** 2026-06-18 14:00:10 UTC

## Contents

compareRSSvsSRS . . . . .	2
gwas_data . . . . .	2
rssGWAS . . . . .	3
rssManhattan . . . . .	4
rssQQ . . . . .	4
rssSample . . . . .	5
simulateGWAS . . . . .	6
srsSample . . . . .	6
standardGWAS . . . . .	7

**Index****8**

---

compareRSSvsSRS	<i>Compare RSS and SRS GWAS</i>
-----------------	---------------------------------

---

**Description**

Compare top SNPs identified using RSS and SRS.

**Usage**

```
compareRSSvsSRS(  
  data,  
  phenotype = "Yield",  
  ranking = "PlantHeight",  
  set.size = 5  
)
```

**Arguments**

data	Dataset.
phenotype	Phenotype variable.
ranking	Ranking variable.
set.size	RSS set size.

**Value**

A list of GWAS results.

**Examples**

```
data(gwas_data)  
comp <- compareRSSvsSRS(gwas_data)
```

---

gwas_data	<i>Example GWAS Dataset</i>
-----------	-----------------------------

---

**Description**

Simulated genome-wide association study (GWAS) dataset containing 500 individuals, 100 SNP markers, a Yield phenotype, and a PlantHeight ranking variable.

**Usage**

```
gwas_data
```

**Format**

A data frame with 500 rows and 103 variables.

The dataset contains:

- **ID**: Individual identifier.
- **SNP1-SNP100**: Genotype markers coded as 0, 1, or 2.
- **Yield**: Quantitative phenotypic trait.
- **PlantHeight**: Ranking variable used in Ranked Set Sampling (RSS).

**Details**

This dataset was generated for illustrating Genome-Wide Association Studies (GWAS) under Ranked Set Sampling (RSS) and Simple Random Sampling (SRS) designs. SNP markers are coded as 0, 1, or 2, representing genotype classes.

**Source**

Simulated data generated using `simulateGWAS()`.

**Examples**

```
data(gwas_data)
dim(gwas_data)
names(gwas_data)[1:10]
```

---

rssGWAS

*RSS-Based GWAS*

---

**Description**

Perform GWAS after Ranked Set Sampling.

**Usage**

```
rssGWAS(data, phenotype = "Yield", ranking = "PlantHeight", set.size = 5)
```

**Arguments**

<code>data</code>	Dataset.
<code>phenotype</code>	Phenotype column.
<code>ranking</code>	Ranking variable.
<code>set.size</code>	RSS set size.

**Value**

GWAS results.

**Examples**

```
data(gwas_data)
res <- rssGWAS(
  gwas_data,
  phenotype = "Yield",
  ranking = "PlantHeight"
)
```

---

**rssManhattan***Manhattan Plot*

---

**Description**

Create Manhattan plot from GWAS results.

**Usage**

```
rssManhattan(results)
```

**Arguments**

**results**            Output from standardGWAS() or rssGWAS().

**Value**

Manhattan plot.

**Examples**

```
data(gwas_data)
res <- standardGWAS(gwas_data)
rssManhattan(res)
```

---

**rssQQ***QQ Plot for GWAS*

---

**Description**

Create QQ plot from GWAS results.

**Usage**

```
rssQQ(results)
```

**Arguments**

results            GWAS results.

**Value**

QQ plot.

**Examples**

```
data(gwas_data)
res <- standardGWAS(gwas_data)
rssQQ(res)
```

---

rssSample            *Ranked Set Sampling (RSS)*

---

**Description**

Performs Ranked Set Sampling (RSS) by dividing data into sets, ranking within sets, and selecting order statistics.

**Usage**

```
rssSample(data, ranking, set.size = 5, m = 1, r = 1)
```

**Arguments**

data                A data frame.  
ranking             Character string specifying ranking variable.  
set.size            Integer: number of units per set.  
m                    Integer: number of cycles (default 1).  
r                    Integer: number of repetitions (default 1).

**Value**

A data frame containing RSS selected observations.

**Examples**

```
data(gwas_data)
rssSample(gwas_data, "PlantHeight", set.size = 5, m = 2, r = 1)
```

simulateGWAS                      *Simulate GWAS Data*

---

**Description**

Generate simulated GWAS genotype and phenotype data.

**Usage**

```
simulateGWAS(n = 500, nSNP = 100, seed = 123)
```

**Arguments**

n	Number of individuals.
nSNP	Number of SNP markers.
seed	Random seed.

**Value**

A data frame containing simulated SNP markers, phenotype values and ranking variable.

**Examples**

```
sim <- simulateGWAS()  
head(sim)
```

---

srsSample                      *Simple Random Sampling*

---

**Description**

Draw a simple random sample.

**Usage**

```
srsSample(data, n)
```

**Arguments**

data	Data frame.
n	Sample size.

**Value**

A sampled data frame.

---

`standardGWAS`*Standard GWAS Analysis*

---

**Description**

Perform single-marker genome-wide association analysis using linear regression.

**Usage**

```
standardGWAS(data, phenotype = "Yield")
```

**Arguments**

<code>data</code>	A data frame containing SNP markers and phenotype.
<code>phenotype</code>	Character string specifying the phenotype column.

**Value**

A data frame containing SNP effects, standard errors, and p-values sorted by significance.

**Examples**

```
data(gwas_data)
res <- standardGWAS(
  gwas_data,
  phenotype = "Yield"
)

head(res)
```

# Index

## \* datasets

- gwas\_data, [2](#)
- compareRSSvsSRS, [2](#)
- gwas\_data, [2](#)
- rssGWAS, [3](#)
- rssManhattan, [4](#)
- rssQQ, [4](#)
- rssSample, [5](#)
- simulateGWAS, [6](#)
- srsSample, [6](#)
- standardGWAS, [7](#)