

# Package: GNapi (via r-universe)

June 7, 2026

**Version** 0.3-8

**Date** 2023-07-26

**Title** Connection to the GeneNetwork API

**Description** Tools for connecting to the GeneNetwork API.

**Author** Karl W Broman [aut, cre]

(<https://orcid.org/0000-0002-4914-6671>)

**Maintainer** Karl W Broman <broman@wisc.edu>

**Depends** R (>= 3.1.0)

**Imports** utils, jsonlite, httr

**Suggests** testthat, devtools, roxygen2, knitr, rmarkdown

**License** MIT + file LICENSE

**URL** <https://github.com/rqtl/GNapi>

**BugReports** <https://github.com/rqtl/GNapi/issues>

**LazyData** true

**Encoding** UTF-8

**ByteCompile** true

**VignetteBuilder** knitr

**Roxygen** list(markdown=TRUE)

**Config/roxygen2/version** 8.0.0

**Config/pak/sysreqs** libssl-dev

**Repository** <https://rqtl.r-universe.dev>

**Date/Publication** 2026-06-07 20:49:05 UTC

**RemoteUrl** <https://github.com/rqtl/GNapi>

**RemoteRef** HEAD

**RemoteSha** b6150c2c1d34141641617f176b26dc9a17944c79

## Contents

|                           |   |
|---------------------------|---|
| check_gn . . . . .        | 2 |
| get_geno . . . . .        | 3 |
| get_pheno . . . . .       | 3 |
| gnapi_url . . . . .       | 4 |
| info_pheno . . . . .      | 4 |
| list_datasets . . . . .   | 5 |
| list_groups . . . . .     | 6 |
| list_species . . . . .    | 7 |
| run_correlation . . . . . | 7 |
| run_gemma . . . . .       | 8 |
| run_rqtl . . . . .        | 9 |

|              |           |
|--------------|-----------|
| <b>Index</b> | <b>10</b> |
|--------------|-----------|

---

|          |                                     |
|----------|-------------------------------------|
| check_gn | <i>Check if GeneNetwork is live</i> |
|----------|-------------------------------------|

---

### Description

Check if GeneNetwork is live

### Usage

```
check_gn(url = gnapi_url())
```

### Arguments

|     |                     |
|-----|---------------------|
| url | URL for GeneNetwork |
|-----|---------------------|

### Value

Character string.

### Examples

```
check_gn()
```

---

|          |                          |
|----------|--------------------------|
| get_geno | <i>Get genotype data</i> |
|----------|--------------------------|

---

**Description**

Get genotype data

**Usage**

```
get_geno(group, url = gnapi_url(), format = "geno")
```

**Arguments**

|        |                                 |
|--------|---------------------------------|
| group  | Name of group                   |
| url    | The URL for the GeneNetwork API |
| format | The group's genotypes format    |

**Value**

A data frame

**See Also**

[get\\_pheno\(\)](#), [list\\_groups\(\)](#)

**Examples**

```
g <- get_geno("QSM")
```

---

|           |                           |
|-----------|---------------------------|
| get_pheno | <i>Get phenotype data</i> |
|-----------|---------------------------|

---

**Description**

Get phenotype data

**Usage**

```
get_pheno(dataset, trait = NULL, url = gnapi_url())
```

**Arguments**

|         |  |
|---------|--|
| dataset | Short abbreviation for a dataset, or name of a group |
| trait   | Optional trait name                                  |
| url     | The URL for the GeneNetwork API                      |

**Value**

A data frame, with samples as rows and either traits as columns or with the columns being detailed information for a single trait

**See Also**

[get\\_genotype\(\)](#)

**Examples**

```
ph1 <- get_pheno("BXDPublish")
ph2 <- get_pheno("BXD", "10002")
ph3 <- get_pheno("HC_M2_0606_P", "1436869_at")
```

---

gnapi\_url

*The URL for the GeneNetwork API*

---

**Description**

The current URL for the GeneNetwork API

**Usage**

```
gnapi_url()
```

**Value**

The URL, as character string

**Examples**

```
gnapi_url()
```

---

info\_pheno

*Get summary information about a phenotype*

---

**Description**

Get summary information about a phenotype

**Usage**

```
info_pheno(group, trait, url = gnapi_url())
```

**Arguments**

|       |                                    |
|-------|------------------------------------|
| group | Name of a group of datasets        |
| trait | Trait identifier (can be a vector) |
| url   | The URL for the GeneNetwork API    |

**Value**

A data frame

**Examples**

```
info_pheno("BXD", "10002")  
info_pheno("HC_M2_0606_P", "1436869_at")
```

---

|               |  |
|---------------|--|
| list_datasets | <i>List available datasets for a group</i> |
|---------------|--|

---

**Description**

List available datasets for a group.

**Usage**

```
list_datasets(group = NULL, dataset = NULL, url = gnapi_url())
```

**Arguments**

|         |   |
|---------|---|
| group   | Name of group, as single character string |
| dataset | Optional name of a specific dataset       |
| url     | URL for GeneNetwork API                   |

**Value**

A data frame with dataset ID, name, and description

**See Also**

[list\\_species\(\)](#), [list\\_groups\(\)](#)

**Examples**

```
list_datasets("BXD")
list_datasets("BXD", "HC_M2_0606_P")
list_datasets(dataset="HC_M2_0606_P")

g <- list_groups()
d <- list_datasets(g$Name[8])
list_datasets(dataset=d$Short_Abbreviation[1])

list_datasets("bxd", "10001")
```

---

|             |                           |
|-------------|---------------------------|
| list_groups | <i>Get list of groups</i> |
|-------------|---------------------------|

---

**Description**

Get list of available groups

**Usage**

```
list_groups(species = NULL, url = gnapi_url())
```

**Arguments**

|         |  |
|---------|--|
| species | Optional species name, for just the groups for that species. |
| url     | URL for GeneNetwork API                                      |

**Value**

Data frame with columns FullName, Id, Name, and TaxonomyId.

**See Also**

[list\\_species\(\)](#), [list\\_datasets\(\)](#)

**Examples**

```
list_groups()
list_groups("barley")
```

---

|              |                            |
|--------------|----------------------------|
| list_species | <i>Get list of species</i> |
|--------------|----------------------------|

---

**Description**

Get list of available species

**Usage**

```
list_species(species = NULL, url = gnapi_url())
```

**Arguments**

|         |   |
|---------|---|
| species | Optional character string, to obtain a particular, single species |
| url     | URL for GeneNetwork API   |

**Value**

Data frame with columns FullName, Id, Name, and TaxonomyId.

**See Also**

[list\\_groups\(\)](#), [list\\_datasets\(\)](#)

**Examples**

```
list_species()
```

---

|                 |                               |
|-----------------|-------------------------------|
| run_correlation | <i>Calculate correlations</i> |
|-----------------|-------------------------------|

---

**Description**

Find correlated traits in GeneNetwork

**Usage**

```
run_correlation(  
  dataset,  
  group,  
  trait,  
  type = c("sample", "tissue"),  
  method = c("pearson", "spearman"),  
  n_results = 500,  
  url = gnapi_url()  
)
```

**Arguments**

|           |   |
|-----------|---|
| dataset   | Name of database for the trait (Short_Abbreviation) |
| group     | Target database name to be correlated against       |
| trait     | ID for trait used for correlation                   |
| type      | Sample or tissue correlation                        |
| method    | Pearson or Spearman correlation                     |
| n_results | Number of results to return                         |
| url       | The URL for the GeneNetwork API                     |

**Value**

Data frame

**Examples**

```
out <- run_correlation("HC_M2_0606_P", "BXDPublish", "1427571_at")
```

---

run\_gemma

*Run gemma*

---

**Description**

Perform a genome scan using gemma

**Usage**

```
run_gemma(dataset, trait, use_loco = FALSE, maf = 0.01, url = gnapi_url())
```

**Arguments**

|          |  |
|----------|--|
| dataset  | ID for dataset being used (Short_Abbreviation) |
| trait    | ID for trait being mapped                      |
| use_loco | Whether to use LOCO (leave one chromosome out) |
| maf      | Filter on minor allele frequency               |
| url      | The URL for the GeneNetwork API                |

**Value**

A data frame

**Examples**

```
out <- run_gemma("HC_M2_0606_P", "1418701_at")
```

---

|          |                  |
|----------|------------------|
| run_rqtl | <i>Run R/qtl</i> |
|----------|------------------|

---

### Description

Perform a genome scan using R/qtl

### Usage

```
run_rqtl(
  dataset,
  trait,
  method = c("hk", "ehk", "em", "imp", "mr", "mr-imp", "mr-argmax"),
  model = c("normal", "binary", "2part", "np"),
  n_perm = 0,
  control_marker = NULL,
  interval_mapping = FALSE,
  url = gnapi_url()
)
```

### Arguments

|                  |   |
|------------------|---|
| dataset          | ID for dataset being used (Short_Abbreviation)  |
| trait            | ID for trait being mapped   |
| method           | Indicates whether to use the EM algorithm, imputation, Haley-Knott regression, the extended Haley-Knott method, or marker regression. |
| model            | The phenotype model: the usual normal model, a model for binary traits, a two-part model, or non-parametric analysis                  |
| n_perm           | Number of permutations  |
| control_marker   | Name of marker to use as a covariate  |
| interval_mapping | Whether to use interval mapping   |
| url              | The URL for the GeneNetwork API   |

### Value

A data frame

### Examples

```
out <- run_rqtl("HC_M2_0606_P", "1418701_at")
```

# Index

`check_gn`, 2

`get_geno`, 3

`get_geno()`, 4

`get_pheno`, 3

`get_pheno()`, 3

`gnapi_url`, 4

`info_pheno`, 4

`list_datasets`, 5

`list_datasets()`, 6, 7

`list_groups`, 6

`list_groups()`, 3, 5, 7

`list_species`, 7

`list_species()`, 5, 6

`run_correlation`, 7

`run_gemma`, 8

`run_rqt1`, 9