

# Package ‘GrafGen’

July 4, 2025

**Title** Classification of Helicobacter Pylori Genomes

**Version** 1.5.0

**Date** 2024-12-16

**Imports** stats, utils, graphics, ggplot2, plotly, scales, RColorBrewer,  
dplyr, grDevices, GenomicRanges, shiny, cowplot, ggpubr,  
stringr, rlang

**Depends** R (>= 4.3.0)

**Suggests** knitr, rmarkdown, RUnit, BiocManager, BiocGenerics,  
BiocStyle, devtools

**Description** To classify Helicobacter pylori genomes according to genetic distance from nine reference populations. The nine reference populations are hpgpAfrica, hpgpAfrica-distant, hpgpAfroamerica, hpgpEuroamerica, hpgpMediterranea, hpgpEurope, hpgpEurasia, hpgpAsia, and hpgpAklavik86-like.  
The vertex populations are Africa, Europe and Asia.

**License** GPL-2

**biocViews** Genetics, Software, GenomeAnnotation, Classification

**NeedsCompilation** yes

**BugReports** <https://github.com/wheelerb/GrafGen/issues>

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/GrafGen>

**git\_branch** devel

**git\_last\_commit** c26e7a8

**git\_last\_commit\_date** 2025-04-15

**Repository** Bioconductor 3.22

**Date/Publication** 2025-07-04

**Author** William Wheeler [aut, cre],  
Difei Wang [aut],  
Isaac Zhao [aut],  
Yumi Jin [aut],  
Charles Rabkin [aut]

**Maintainer** William Wheeler <[wheelerb@imsweb.com](mailto:wheelerb@imsweb.com)>

## Contents

|                                       |           |
|---------------------------------------|-----------|
| GrafGen-package . . . . .             | 2         |
| createApp . . . . .                   | 3         |
| example_metadata . . . . .            | 4         |
| grafGen . . . . .                     | 4         |
| grafGenPlot . . . . .                 | 5         |
| grafGen_example_results . . . . .     | 6         |
| grafGen_reference_dataframe . . . . . | 7         |
| grafGen_reference_results . . . . .   | 7         |
| HpyloriData . . . . .                 | 8         |
| interactivePlot . . . . .             | 8         |
| interactiveReferencePlot . . . . .    | 9         |
| S3 methods . . . . .                  | 10        |
| <b>Index</b>                          | <b>12</b> |

---

|                 |                                   |
|-----------------|-----------------------------------|
| GrafGen-package | <i>Classify H. pylori genomes</i> |
|-----------------|-----------------------------------|

---

## Description

To classify H. pylori genomes according to genetic distance from nine reference populations.

## Details

This package was modified from the GrafPop software (<https://www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/Software.cgi>) to be applied on H. pylori genomes. The three vertex populations are "Africa", "Europe" and "Asia". The nine reference populations are "hpgpAfrica", "hpgpAfrica-distant", "hpgpAfroamerica", "hpgpEuroamerica", "hpgpMediterranea", "hpgpEurope", "hpgpEurasia", "hpgpAsia", and "hpgpAklavik86-like". The training data is based on The Helicobacter pylori Genome Project (HpGP), see <https://www.ncbi.nlm.nih.gov/bioproject/?term=HpGP> or <https://zenodo.org/records/10048320>.

To use this package, the user must have a file of genotypes for H. pylori strains. The genotype file can be a binary PLINK file in SNP-major format, or a VCF file of genotypes. If a PLINK file, then the corresponding bim and fam files must also be present. If a VCF file, then the format should be genotypes:

```
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">.
```

Ideally, the genotype file will contain all the SNPs with positions given in the `HpyloriData` data frame, where the positions are based on the reference genome 26695 (NCBI GenBank Accession NC\_000915.1). However, the software has been shown to work well with only using a much smaller fraction of SNPs in `HpyloriData`. The main function in this package is `grafGen`.

## Author(s)

William Wheeler, Difei Wang, Isaac Zhao, Yumi Jin, Charles Rabkin

## References

Jin Y, Schaffer AA, Feolo M, Holmes JB and Kattman BL (2019). GRAF-pop: A Fast Distance-based Method to Infer Subject Ancestry from Multiple Genotype Datasets without Principal Components Analysis. *G3: Genes | Genomes | Genetics*. DOI: 10.1534/g3.118.200925.

Thorell K, Munoz-Ramirez ZY, Wang D, Sandoval-Motta S, Boscolo Agostini R, Ghirotto S, Torres RC, HpGP Research Network, Falush D, Camargo MC and Rabkin CS (2023). New insights into *Helicobacter pylori* population structure from analysis of a worldwide collection of complete genomes: the *H. pylori* genome project. *Nature Communications*. DOI: 10.1038/s41467-023-43562-y.

---

createApp

*R Shiny App*

---

## Description

To return an R Shiny app for the user's data.

## Usage

```
createApp(obj, metadata=NULL, id=NULL)
```

## Arguments

|          |                                                                                                    |
|----------|----------------------------------------------------------------------------------------------------|
| obj      | Return object from <a href="#">grafGen</a> .                                                       |
| metadata | NULL or data frame containing meta data for the plot. This data frame must contain an id variable. |
| id       | Name of the id column in metadata. If NULL, then the first column will be used.                    |

## Details

This R function returns an R Shiny app that can be launched by calling [runApp](#). The app allows the user to view and filter the plot using up to two variables.

## Value

A list containing an R Shiny app and data frames needed to run the app.

## See Also

[grafGen](#)

## Examples

```
library(GrafGen)
data(grafGen_example_results, package="GrafGen")
data(example_metadata, package="GrafGen")
tmp <- createApp(grafGen_example_results, metadata=example_metadata,
  id="Sample")
reference_results <- tmp$reference_results
user_results      <- tmp$user_results
```

```

user_metadata    <- tmp$user_metadata
if (interactive()) {
  shiny::runApp(tmp$app)
}

```

---

|                  |                              |
|------------------|------------------------------|
| example_metadata | <i>Metadata for examples</i> |
|------------------|------------------------------|

---

### Description

A data frame containing metadata used in examples.

### Details

The data frame contains the sample id, type (i.e. source country), and country abbreviation for the 206 genomes in [grafGen\\_example\\_results](#).

### Value

A data frame

### See Also

[grafGen\\_example\\_results](#)

### Examples

```

data(example_metadata, package="GrafGen")

# Display a few rows
example_metadata[seq_len(5), ]

```

---

|         |                                                   |
|---------|---------------------------------------------------|
| grafGen | <i>Reference population for H. pylori strains</i> |
|---------|---------------------------------------------------|

---

### Description

To determine the ancestry of H. pylori strains.

### Usage

```
grafGen(genoFile, print=1)
```

### Arguments

|          |                                                                                                                                                                                                                                                                                                                                      |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| genoFile | The complete path to the input genotype file. This file can only be a PLINK binary file (.bed) or a VCF file (.vcf, .vcf.gz). If it is a .bed file, then the corresponding .bim and .fam files must also exist. If a VCF file, then the format should be genotypes:<br>##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">. |
| print    | 0 or 1 to print information as the program runs.                                                                                                                                                                                                                                                                                     |

Details

See the references for complete details of the algorithm.

This function is more efficient if the input genotype file only contains the set (or subset) of SNPs defined in [HpyloriData](#). The SNPs can be extracted by utilizing the VCFtools software if the genotype file is a VCF file. For a binary PLINK file, the PLINK software can be used to extract the SNPs.

Value

A list of class "grafpop" containing a data frame (table) that includes the ancestry percents (F\_percent, E\_percent, A\_percent) for African, European and Asian respectively, normalized genetic distance scores (GD1\_x, GD2\_y, GD3\_z), the predicted reference population (Refpop), next nearest reference population (Nearest\_neighbor), separation to the next nearest reference population (Separation\_percent) defined as  $100 \cdot \text{abs}(d1 - d2) / d1$ , where d1 and d2 are the genetic distances to the sample's assigned reference population and next nearest reference population respectively, and the genetic distances to each reference population (hpgpAfrica, hpgpAfrica-distant, hpgpAfroamerica, hpgpEuroamerica, hpgpMediterranea, hpgpEurope, hpgpEurasia, hpgpAsia, and hpgpAklavik86-like) as defined by equation 2 in Jin (2019). The returned object also includes the list vertex which gives the x-y coordinates of the vertex populations.

See Also

[HpyloriData](#)

Examples

```
dir <- system.file("extdata", package="GrafGen", mustWork=TRUE)
file <- file.path(dir, "data.vcf.gz")
grafGen(file)
```

|             |              |
|-------------|--------------|
| grafGenPlot | Plot results |
|-------------|--------------|

Description

Plot results

Usage

```
grafGenPlot(obj, which=1, legend.pos=NULL,
            ylim=NULL, showRefData=TRUE,
            jitter=0)
```

Arguments

- obj

An object of class "grafpop" returned from [grafGen](#).
- which

A vector of integers in 1, 2, 3, 4, 5 to determine which plots are produced where:
- which

plot

1

GD1\_x vs GD2\_y

|             |                                                                                                                         |
|-------------|-------------------------------------------------------------------------------------------------------------------------|
| 2           | GD1_x vs GD3_z                                                                                                          |
| 3           | GD2_y vs GD3_z                                                                                                          |
| 4           | Distance from each strain to its predicted population                                                                   |
| 5           | Ancestry percents for each strain within each reference population                                                      |
| legend.pos  | The position of the legend. See <a href="#">legend</a> .                                                                |
| ylim        | NULL or the limits of the y-axis. See <a href="#">plot</a> .                                                            |
| showRefData | TRUE or FALSE to display the 95 percent confidence ellipses for the reference data results.                             |
| jitter      | Numeric value for the amount of jitter to add for the plot which = 5. Values less than 0.5 work well. The default is 0. |

### Details

The option `legend.pos` is only available for `which = 1-3`, option `ylim` is only available for `which = 4-5`, and option `jitter` is only available for `which = 5`.

### Value

NULL

### See Also

[grafGen](#)

### Examples

```
data(grafGen_example_results, package="GrafGen")
grafGenPlot(grafGen_example_results)
```

---

grafGen\_example\_results

*Example results*

---

### Description

The returned object from [grafGen](#) in the analysis of a subset of the reference data.

### Details

An object of class "grafpop" containing the [grafGen](#) results for a subset of 206 genomes and 35528 SNPs in the reference data. This subset of the reference data is included in the package (`/extdata/data.vcf.gz`).

### Value

An object of class "grafpop".

### See Also

[example\\_metadata](#)

**Examples**

```
data(grafGen_example_results, package="GrafGen")
grafGen_example_results
```

---

```
grafGen_reference_dataframe
```

*Reference data results for plots*

---

**Description**

A data frame of the reference data results used in creating plots.

**Details**

The data frame contains the results for each of the 1011 genomes in the reference data used in training the model along with some additional columns.

**Value**

A data frame

**See Also**

[grafGen\\_reference\\_results](#)

**Examples**

```
data(grafGen_reference_dataframe, package="GrafGen")

# Display a few rows
grafGen_reference_dataframe[seq_len(5), ]
```

---

```
grafGen_reference_results
```

*Reference data results*

---

**Description**

The returned object from [grafGen](#) in the analysis of the reference data.

**Details**

An object of class "grafpop" containing the [grafGen](#) results for each of the 1011 genomes in the reference data. The full set of reference data can be found at <https://github.com/wheelerb/GrafGen/tree/reference/data>.

**Value**

An object of class "grafpop".

See Also

[grafGen](#)

Examples

```
data(grafGen_reference_results, package="GrafGen")
grafGen_reference_results
```

---

|             |                       |
|-------------|-----------------------|
| HpyloriData | <i>H. pylori data</i> |
|-------------|-----------------------|

---

Description

SNP positions and allele frequencies for the reference data

Details

A [GPos](#) class object containing the vertex and reference population allele frequencies for the set of 143705 SNPs used in the analysis for *H. pylori*. The SNPs were created using 26695 (NCBI GenBank Accession NC\_000915.1) as the reference genome. The set of SNPs was selected using a MAF threshold of 0.01. The total sample size was from a set of 1011 *H. pylori* strains.

Value

An object of class [GPos](#).

Examples

```
# Load data and view the first few rows
data(HpyloriData, package="GrafGen")
HpyloriData
```

---

|                 |                                        |
|-----------------|----------------------------------------|
| interactivePlot | <i>Interactive plot of user's data</i> |
|-----------------|----------------------------------------|

---

Description

Create an interactive plot of user data

Usage

```
interactivePlot(obj, metadata=NULL, id=NULL, type=NULL, group=NULL)
```

**Arguments**

|          |                                                                                                                                                                                   |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| obj      | Return object from <a href="#">grafGen</a> .                                                                                                                                      |
| metadata | NULL or data frame containing meta data for the plot. This data frame must contain an id variable.                                                                                |
| id       | Name of the id column in metadata. If NULL, then the first column will be used.                                                                                                   |
| type     | Name of the type variable in metadata, which is rendered as different colors. This variable should be categorical. If NULL, then NA values will appear when hovering over points. |
| group    | Name of the group variable in metadata, which is rendered as different symbols. This variable should be categorical. If NULL, then group will be set to type.                     |

**Details**

This plot will all show the results of all samples in the user's data. Hovering over a point in the plot will display three lines of information. Line 1 contains the group, type and id of that sample. Line 2 contains the sample's assigned reference population, next nearest reference population, and separation to the next nearest reference population defined as  $100 * \text{abs}(d1 - d2) / d1$ , where  $d1$  and  $d2$  are the genetic distances to the sample's assigned reference population and next nearest reference population respectively. Line 3 contains the percent African, European and Asian ancestry for that sample. The legend shows the types for all samples, and clicking a type will add or remove those samples from the plot.

Note that printing the returned object from [grafGen](#) with the command `print(obj)` will display the frequency counts for each reference population.

**Value**

NULL

**See Also**

[grafGenPlot](#)

**Examples**

```
if (interactive()) {
  data(grafGen_example_results, package="GrafGen")
  data(example_metadata, package="GrafGen")
  interactivePlot(grafGen_example_results, metadata=example_metadata,
    id="Sample", type="Country")
}
```

---

interactiveReferencePlot

*Interactive plot of the reference data*

---

**Description**

Create an interactive plot of the reference data

**Usage**

```
interactiveReferencePlot()
```

**Details**

This plot will all show the results of all samples in the reference data. Hovering over a point in the plot will display three lines of information. Line 1 contains the type (i.e., the source country) and id of that sample. Line 2 contains the sample's assigned reference population, next nearest reference population, and separation to the next nearest reference population defined as  $100 \times \text{abs}(d1 - d2)/d1$ , where  $d1$  and  $d2$  are the genetic distances to the sample's assigned reference population and next nearest reference population respectively. Line 3 contains the percent African, European and Asian ancestry for that sample. The legend shows the abbreviated names of the source countries for all samples, and clicking a country will add or remove those samples from the plot.

**Value**

NULL

**See Also**

[grafGenPlot](#)

**Examples**

```
if (interactive()) {
  interactiveReferencePlot()
}
```

---

S3 methods

*Plot and Print*

---

**Description**

Plot or print an object of class "grafpop".

**Usage**

```
## S3 method for class 'grafpop'
plot(x, legend.pos="right", showRefData=TRUE,
     ...)
## S3 method for class 'grafpop'
print(x, ...)
```

**Arguments**

|                          |                                                                                             |
|--------------------------|---------------------------------------------------------------------------------------------|
| <code>x</code>           | An object of class "grafpop" returned from <a href="#">grafGen</a> .                        |
| <code>legend.pos</code>  | The position of the legend. The default is "topleft". See <a href="#">legend</a> .          |
| <code>showRefData</code> | TRUE or FALSE to display the 95 percent confidence ellipses for the reference data results. |
| <code>...</code>         | Additional arguments.                                                                       |

**Details**

Printing an object of class "grafpop" will display the frequency counts of the predicted reference populations.

**Value**

NULL

**See Also**

[grafGen](#), [grafGenPlot](#)

**Examples**

```
data(grafGen_example_results, package="GrafGen")
obj <- grafGen_example_results
print(obj)
plot(obj)
```

# Index

- \* **data**
  - example\_metadata, [4](#)
  - grafGen\_example\_results, [6](#)
  - grafGen\_reference\_dataframe, [7](#)
  - grafGen\_reference\_results, [7](#)
  - HpyloriData, [8](#)
- \* **package**
  - GrafGen-package, [2](#)
- createApp, [3](#)
- example\_metadata, [4](#), [6](#)
- GPos, [8](#)
- GrafGen (GrafGen-package), [2](#)
- grafGen, [2](#), [3](#), [4](#), [5–11](#)
- GrafGen-package, [2](#)
- grafGen\_example\_results, [4](#), [6](#)
- grafGen\_reference\_dataframe, [7](#)
- grafGen\_reference\_results, [7](#), [7](#)
- grafGenPlot, [5](#), [9–11](#)
- HpyloriData, [2](#), [5](#), [8](#)
- interactivePlot, [8](#)
- interactiveReferencePlot, [9](#)
- legend, [6](#), [10](#)
- plot, [6](#)
- plot.grafpop (S3 methods), [10](#)
- print.grafpop (S3 methods), [10](#)
- runApp, [3](#)
- S3 methods, [10](#)