

# Package ‘PopGenHelpR’

May 9, 2024

**Title** Streamline Population Genomic and Genetic Analyses

**Version** 1.3.0

**Description** Estimate commonly used population genomic statistics and generate publication quality figures. 'PopGenHelpR' uses vcf, 'geno' (012), and csv files to generate output.

**URL** <https://kfarleigh.github.io/PopGenHelpR/>

**BugReports** <https://github.com/kfarleigh/PopGenHelpR/issues>

**License** GPL (>= 3)

**Encoding** UTF-8

**RoxygenNote** 7.3.1

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Imports** adegenet, dartR, dplyr, ggplot2, gstat, hierfstat, magrittr, methods, poppr, raster, reshape2, rlang, scatterpie, StAMPP, stats, sp, spData, spdep, utils, vcfR

**Depends** R (>= 2.10)

**LazyData** true

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Keaka Farleigh [aut, cph, cre]  
(<https://orcid.org/0000-0002-9195-121X>),  
Mason Murphy [aut, cph, ctb] (<https://orcid.org/0000-0002-5317-1386>),  
Christopher Blair [aut, cph, ctb]  
(<https://orcid.org/0000-0002-4782-6788>),  
Tereza Jezkova [aut, cph, ctb]  
(<https://orcid.org/0000-0003-4114-4564>)

**Maintainer** Keaka Farleigh <keakafarleigh@gmail.com>

**Repository** CRAN

**Date/Publication** 2024-05-09 19:30:02 UTC

**R topics documented:**

Ancestry_barchart . . . . .	2
Differentiation . . . . .	3
Dif_stats . . . . .	5
Dif_Stats_Map . . . . .	6
Div_stats . . . . .	8
Div_Stats_Map . . . . .	9
Fst_dat . . . . .	10
Heterozygosity . . . . .	11
Het_dat . . . . .	13
HornedLizard_Pop . . . . .	14
HornedLizard_VCF . . . . .	15
Network_map . . . . .	15
Pairwise_heatmap . . . . .	17
PCA . . . . .	18
Piechart_map . . . . .	19
Plot_ancestry . . . . .	20
Plot_coordinates . . . . .	22
Point_map . . . . .	23
Private.alleles . . . . .	24
Q_dat . . . . .	25
<b>Index</b>	<b>27</b>

---

Ancestry_barchart	<i>Plot an ancestry matrix for individuals and(or) populations.</i>
-------------------	---

---

**Description**

Plot an ancestry matrix for individuals and(or) populations.

**Usage**

```
Ancestry_barchart(
  anc.mat,
  pops,
  K,
  plot.type = "all",
  col,
  ind.order = NULL,
  pop.order = NULL
)
```

**Arguments**

<code>anc.mat</code>	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The first column should be the names of each sample/population, followed by the estimated contribution of each cluster to that individual/pop.
<code>pops</code>	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The columns should be named Sample, containing the sample IDs; Population indicating the population assignment of the individual, population and sample names must be the same type (i.e., both numeric or both characters); Long, indicating the longitude of the sample; Lat, indicating the latitude of the sample.
<code>K</code>	Numeric. The number of genetic clusters in your data set, please contact the package authors if you need help doing this.
<code>plot.type</code>	Character string. Options are all, individual, and population. All is default and recommended, this will plot a barchart for both the individuals and populations.
<code>col</code>	Character vector indicating the colors you wish to use for plotting.
<code>ind.order</code>	Character vector indicating the order to plot the individuals in the individual ancestry bar chart.
<code>pop.order</code>	Character vector indicating the order to plot the populations in the population ancestry bar chart.

**Value**

A list containing your plots and the data frames used to generate the plots.

**Author(s)**

Keaka Farleigh

**Examples**

```
data(Q_dat)
Qmat <- Q_dat[[1]]
rownames(Qmat) <- Qmat[,1]
Loc <- Q_dat[[2]]
Test_all <- Ancestry_barchart(anc.mat = Qmat, pops = Loc, K = 5,
plot.type = 'all', col = c('#d73027', '#fc8d59', '#e0f3f8', '#91bfdb', '#4575b4'))
```

---

Differentiation

*A function to estimate three measures of genetic differentiation using geno files, vcf files, or vcfR objects. Data is assumed to be bi-allelic.*

---

**Description**

A function to estimate three measures of genetic differentiation using geno files, vcf files, or vcfR objects. Data is assumed to be bi-allelic.

**Usage**

```
Differentiation(
  data,
  pops,
  statistic = "all",
  missing_value = NA,
  write = FALSE,
  prefix = NULL,
  population_col = NULL,
  individual_col = NULL
)
```

**Arguments**

<code>data</code>	Character. String indicating the name of the vcf file, geno file or vcfR object to be used in the analysis.
<code>pops</code>	Character. String indicating the name of the population assignment file or dataframe containing the population assignment information for each individual in the data. This file must be in the same order as the vcf file and include columns specifying the individual and the population that individual belongs to. The first column should contain individual names and the second column should indicate the population assignment of each individual. Alternatively, you can indicate the column containing the individual and population information using the <code>individual_col</code> and <code>population_col</code> arguments.
<code>statistic</code>	Character. String or vector indicating the statistic to calculate. Options are any of: <code>all</code> ; all of the statistics; <code>Fst</code> , Weir and Cockerham (1984) <code>Fst</code> ; <code>NeisD</code> , Nei's D statistic; <code>JostsD</code> , Jost's D.
<code>missing_value</code>	Character. String indicating missing data in the input data. It is assumed to be <code>NA</code> , but that may not be true (is likely not) in the case of geno files.
<code>write</code>	Boolean. Whether or not to write the output to files in the current working directory. There will be one or two files for each statistic. Files will be named based on their statistic such as <code>Fst_perpop.csv</code> .
<code>prefix</code>	Character. Optional argument. String that will be appended to file output. Please provide a prefix if <code>write</code> is set to <code>TRUE</code> .
<code>population_col</code>	Numeric. Optional argument (a number) indicating the column that contains the population assignment information.
<code>individual_col</code>	Numeric. Optional argument (a number) indicating the column that contains the individuals (i.e., sample name) in the data.

**Value**

A list containing the estimated heterozygosity statistics. The per pop values are calculated by taking the average of the per locus estimates.

**Author(s)**

Keaka Farleigh

## References

### Fst:

Pembleton, L. W., Cogan, N. O., & Forster, J. W. (2013). StAMPP: An R package for calculation of genetic differentiation and structure of mixed-ploidy level populations. *Molecular ecology resources*, 13(5), 946-952.[doi:10.1111/17550998.12129](https://doi.org/10.1111/17550998.12129)

Weir, B. S., & Cockerham, C. C. (1984). Estimating F-statistics for the analysis of population structure. *evolution*, 1358-1370.

### Nei's D:

Nei, M. (1972). Genetic distance between populations. *The American Naturalist*, 106(949), 283-292.[doi:10.1086/282771](https://doi.org/10.1086/282771)

[doi:10.1111/17550998.12129](https://doi.org/10.1111/17550998.12129) Pembleton, L. W., Cogan, N. O., & Forster, J. W. (2013). StAMPP: An R package for calculation of genetic differentiation and structure of mixed-ploidy level populations. *Molecular ecology resources*, 13(5), 946-952.

### Jost's D:

Jost L (2008). GST and its relatives do not measure differentiation. *Molecular Ecology*, 17, 4015–4026.[doi:10.1111/j.1365294X.2008.03887.x](https://doi.org/10.1111/j.1365294X.2008.03887.x)

## Examples

```
data("HornedLizard_Pop")
data("HornedLizard_VCF")
Test <- Differentiation(data = HornedLizard_VCF, pops = HornedLizard_Pop, write = FALSE)
```

---

Dif\_stats

***WARNING! This function has been deprecated and is no longer supported. Please use the Differentiation function. A function to calculate differentiation statistics and perform significance testing with a vcf file.***

---

## Description

**WARNING! This function has been deprecated and is no longer supported. Please use the Differentiation function.** A function to calculate differentiation statistics and perform significance testing with a vcf file.

## Usage

```
Dif_stats(
  VCF,
  pops,
  ploidy,
  statistic = "both",
  boots,
  write = FALSE,
```

```

    prefix = NULL
  )

```

### Arguments

VCF	Character string indicating the name of the vcf file to be used in analysis.
pops	Character string indicating the name of the population assignment file. This file should have four columns and be in the same order as your vcf file. The first column named Sample indicates the sample name. The second column named Population indicates the population assignment of each individual. The third column named Long indicates the longitude of the sample. The fourth column named Lat indicates the latitude of the sample.
ploidy	Numeric. The ploidy of the data.
statistic	Character string. Options are both, FST, and NeisD.
boots	Numeric. The number of bootstraps to use to evaluate statistical significance. Only relevant for FST estimation.
write	Boolean. Whether or not to write the output to a file in the current working directory.
prefix	Character string that will be appended to file output.

### Value

A list containing data frames for the requested statistic.

### Examples

```

data("HornedLizard_Pop")
data("HornedLizard_VCF")
Test <- Dif_stats(VCF = HornedLizard_VCF, pops = HornedLizard_Pop,
ploidy = 2, statistic = "both", boots = 10, write = FALSE)

```

---

Dif\_Stats\_Map

***WARNING! This function has been deprecated and is no longer supported. Please use the Network\_map function. A function to map differentiation statistics.***

---

### Description

**WARNING! This function has been deprecated and is no longer supported. Please use the Network\_map function.** A function to map differentiation statistics.

**Usage**

```
Dif_Stats_Map(
  dat,
  pops,
  neighbors,
  col,
  breaks = NULL,
  Lat_buffer,
  Long_buffer
)
```

**Arguments**

dat	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. If it is a csv, the 1st row should contain the individual/population names. The columns should also be named in this fashion.
pops	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The columns should be named Sample, containing the sample IDs; Population indicating the population assignment of the individual; Long, indicating the longitude of the sample; Lat, indicating the latitude of the sample.
neighbors	Numeric. The number of neighbors to plot connections with.
col	Character vector indicating the colors you wish to use for plotting.
breaks	Numeric. The breaks used to generate the color ramp when plotting. Users should supply 3 values if custom breaks are desired.
Lat_buffer	Numeric. A buffer to customize visualization.
Long_buffer	Numeric. A buffer to customize visualization.

**Value**

A list containing the map and the matrix used to plot the map.

**Author(s)**

Keaka Farleigh

**Examples**

```
data(Fst_dat)
Fst <- Fst_dat[[1]]
Loc <- Fst_dat[[2]]
Test <- Dif_Stats_Map(dat = Fst, pops = Loc,
  neighbors = 2,
  col = c('#fd8d3c', '#fc4e2a', '#e31a1c', '#bd0026', '#800026'), Lat_buffer = 1, Long_buffer = 1)
```

---

Div_stats	<b><i>WARNING! This function has been deprecated and is no longer supported. Please use the Heterozygosity and Private.alleles functions. A function to estimate heterozygosity and the number of private alleles from a vcf file.</i></b>
-----------	--

---

### Description

**WARNING! This function has been deprecated and is no longer supported. Please use the Heterozygosity and Private.alleles functions.** A function to estimate heterozygosity and the number of private alleles from a vcf file.

### Usage

```
Div_stats(VCF, pops, ploidy, write = FALSE, prefix)
```

### Arguments

VCF	Character string indicating the name of the vcf file to be used in analysis.
pops	Character string indicating the name of the population assignment file. This file should have four columns and be in the same order as your vcf file. The first column named Sample indicates the sample name. The second column named Population indicates the population assignment of each individual. The third column named Longitude indicates the longitude of the sample. The fourth column named Latitude indicates the latitude of the sample.
ploidy	Numeric. The ploidy of the data.
write	Boolean. Whether or not to write the output to a file in the current working directory.
prefix	Character string that will be appended to file output.

### Value

A list containing the estimated diversity statistics, model output from linear regression of these statistics against latitude, and model plots.

### Examples

```
data("HornedLizard_Pop")
data("HornedLizard_VCF")
Test <- Div_stats(VCF = HornedLizard_VCF, pops = HornedLizard_Pop,
ploidy = 2, write = FALSE)
```



---

Div_Stats_Map	<b><i>WARNING! This function has been deprecated and is no longer supported. Please use the Point_map function instead. A function to map diversity statistics.</i></b>
---------------	---

---

### Description

**WARNING! This function has been deprecated and is no longer supported. Please use the Point\_map function instead.** A function to map diversity statistics.

### Usage

```
Div_Stats_Map(
  dat,
  plot.type = "all",
  statistic,
  breaks = NULL,
  col,
  Lat_buffer = 1,
  Long_buffer = 1,
  write = FALSE,
  prefix = NULL
)
```

### Arguments

dat	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The first column should be the statistic to be plotted and named the same as the statistic argument. The second column is Population indicating which population each row belongs to. The third column is the standard deviation, the fourth column is Long indicating the longitude, and the fifth column is Lat, indicating the latitude.
plot.type	Character string. Options are all, point, or interpolated. All is recommended and will generate a map with points colored according to heterozygosity as well as a raster of interpolated heterozygosity values.
statistic	Character string. The statistic to be plotted.
breaks	Numeric. The breaks used to generate the color ramp when plotting. Users should supply 3 values if custom breaks are desired.
col	Character vector indicating the colors you wish to use for plotting, three colors are allowed (low, mid, high). The first color will be the low color, the second the middle, the third the high.
Lat_buffer	Numeric. A buffer to customize visualization.
Long_buffer	Numeric. A buffer to customize visualization.
write	Boolean. Whether or not to write the output to a file in the current working directory.
prefix	Character string that will be appended to file output.

**Value**

A list containing maps and the data frames used to generate them.

**Examples**

```
data(Het_dat)
Test_het <- Div_Stats_Map(dat = Het_dat, plot.type = 'all',
  statistic = "Heterozygosity",
  Lat_buffer = 1, Long_buffer = 1, write = FALSE, prefix = 'Test_het')
```

---

Fst_dat	<i>A genetic differentiation matrix and locality information for each population. This data was generated by subsetting data of Farleigh et al., 2021.</i>
---------	--

---

**Description**

A symmetric matrix with estimated genetic differentiation (Fst) between 3 populations.

**Usage**

```
data(Fst_dat)
```

**Format**

A list with two elements:

**Fst\_dat** Data frame with three rows and three columns

**Loc\_dat** Data frame containing the locality information for each population

...

**Source**

Farleigh, K., Vladimirova, S. A., Blair, C., Bracken, J. T., Koochekian, N., Schield, D. R., ... & Jezkova, T. (2021). The effects of climate and demographic history in shaping genomic variation across populations of the Desert Horned Lizard (*Phrynosoma platyrhinos*). *Molecular Ecology*, 30(18), 4481-4496.

**Examples**

```
data(Fst_dat)
Fst <- Fst_dat[[1]]
Loc <- Fst_dat[[2]]

Test <- Dif_Stats_Map(dat = Fst, pops = Loc,
  neighbors = 2,
  col = c('#fd8d3c', '#fc4e2a', '#e31a1c', '#bd0026', '#800026'), Lat_buffer = 1, Long_buffer = 1)
```

```
Fstat_plot <- Pairwise_heatmap(dat = Fst, statistic = 'FST')
```

---

Heterozygosity      *A function to estimate seven measures of heterozygosity using geno files, vcf files, or vcfR objects. Data is assumed to be bi-allelic.*

---

### Description

A function to estimate seven measures of heterozygosity using geno files, vcf files, or vcfR objects. Data is assumed to be bi-allelic.

### Usage

```
Heterozygosity(
  data,
  pops,
  statistic = "all",
  missing_value = NA,
  write = FALSE,
  prefix = NULL,
  population_col = NULL,
  individual_col = NULL
)
```

### Arguments

<code>data</code>	Character. String indicating the name of the vcf file, geno file or vcfR object to be used in the analysis.
<code>pops</code>	Character. String indicating the name of the population assignment file or dataframe containing the population assignment information for each individual in the data. This file must be in the same order as the vcf file and include columns specifying the individual and the population that individual belongs to. The first column should contain individual names and the second column should indicate the population assignment of each individual. Alternatively, you can indicate the column containing the individual and population information using the <code>individual_col</code> and <code>population_col</code> arguments.
<code>statistic</code>	Character. String or vector indicating the statistic to calculate. Options are any of: <code>all</code> ; <code>all</code> of the statistics; <code>Ho</code> , observed heterozygosity; <code>He</code> , expected heterozygosity; <code>PHt</code> , proportion of heterozygous loci; <code>Hs_exp</code> , heterozygosity standardized by the average expected heterozygosity; <code>Hs_obs</code> , heterozygosity standardized by the average observed heterozygosity; <code>IR</code> , internal relatedness; <code>HL</code> , homozygosity by locus.
<code>missing_value</code>	Character. String indicating missing data in the input data. It is assumed to be <code>NA</code> , but that may not be true (is likely not) in the case of geno files.

write	Boolean. Whether or not to write the output to files in the current working directory. There will be one or two files for each statistic. Files will be named based on their statistic such as Ho_perpop.csv or Ho_perloc.csv.
prefix	Character. Optional argument. String that will be appended to file output. Please provide a prefix if write is set to TRUE.
population_col	Numeric. Optional argument (a number) indicating the column that contains the population assignment information.
individual_col	Numeric. Optional argument (a number) indicating the column that contains the individuals (i.e., sample name) in the data.

### Value

A list containing the estimated heterozygosity statistics. The per pop values are calculated by taking the average of the per locus estimates.

### Author(s)

Keaka Farleigh

### References

#### Expected (He) and observed heterozygosity (Ho):

Nei, M. (1987) *Molecular Evolutionary Genetics*. Columbia University Press

#### Homozygosity by locus (HL) and internal relatedness (IR):

Alho, J. S., Välimäki, K., & Merilä, J. (2010). Rhh: an R extension for estimating multilocus heterozygosity and heterozygosity–heterozygosity correlation. *Molecular ecology resources*, 10(4), 720-722.

Amos, W., Worthington Wilmer, J., Fullard, K., Burg, T. M., Croxall, J. P., Bloch, D., & Coulson, T. (2001). The influence of parental relatedness on reproductive success. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 268(1480), 2021-2027. doi:10.1098/rspb.2001.1751

Aparicio, J. M., Ortego, J., & Cordero, P. J. (2006). What should we weigh to estimate heterozygosity, alleles or loci?. *Molecular Ecology*, 15(14), 4659-4665.

#### Heterozygosity standardized by expected (Hs\_exp) and observed heterozygosity (Hs\_obs):

Coltman, D. W., Pilkington, J. G., Smith, J. A., & Pemberton, J. M. (1999). Parasite-mediated selection against Inbred Soay sheep in a free-living island population. *Evolution*, 53(4), 1259-1267.

### Examples

```
data("HornedLizard_Pop")
data("HornedLizard_VCF")
Test <- Heterozygosity(data = HornedLizard_VCF, pops = HornedLizard_Pop, write = FALSE)
```

---

Het_dat	<i>A data frame of hypothetical heterozygosity data produced by Div_Stats.</i>
---------	--

---

## Description

Data frame containing 5 columns and 3 rows

## Usage

```
data(Het_dat)
```

## Format

A data frame with 5 columns and 3 rows:

**Heterozygosity** Estimated heterozygosity

**Pop** Population assignment

**Standard.Deviation** standard deviation

**Longitude** Longitude

**Latitude** Latitude

...

## Source

Coordinates and population names taken from Farleigh, K., Vladimirova, S. A., Blair, C., Bracken, J. T., Koochekian, N., Schield, D. R., ... & Jezkova, T. (2021). The effects of climate and demographic history in shaping genomic variation across populations of the Desert Horned Lizard (*Phrynosoma platyrhinos*). *Molecular Ecology*, 30(18), 4481-4496.

## Examples

```
data(Het_dat)
Test_het <- Div_Stats_Map(dat = Het_dat, plot.type = 'all',
  statistic = "Heterozygosity",
  Lat_buffer = 1, Long_buffer = 1, write = FALSE, prefix = 'Test_het')
```

---

HornedLizard_Pop	<i>A population assignment data frame to be used in Div_stats and Dif_stats.</i>
------------------	--

---

### Description

Data frame containing 4 columns and 72 rows

### Usage

```
data(HornedLizard_Pop)
```

### Format

A data frame with 4 columns and 72 rows:

**Sample** Sample Name

**Population** Population assignment according to sNMF results (see citation)

**Longitude** Longitude

**Latitude** Latitude

...

### Source

Coordinates and population names taken from Farleigh, K., Vladimirova, S. A., Blair, C., Bracken, J. T., Koochekian, N., Schield, D. R., ... & Jezkova, T. (2021). The effects of climate and demographic history in shaping genomic variation across populations of the Desert Horned Lizard (*Phrynosoma platyrhinos*). *Molecular Ecology*, 30(18), 4481-4496.

### Examples

```
data("HornedLizard_Pop")
data("HornedLizard_VCF")
Test <- Div_stats(VCF = HornedLizard_VCF, pops = HornedLizard_Pop,
ploidy = 2, write = FALSE)
```

---

HornedLizard_VCF	<i>A vcfR object to be used in Div_stats and Dif_stats.</i>
------------------	---

---

### Description

Data frame containing 4 columns and 72 rows

### Usage

```
data(HornedLizard_Pop)
```

### Format

A vcfR object

**vcfR object** A vcfR object containing genotype and sample information for 72 individuals.

...

### Source

Farleigh, K., Vladimirova, S. A., Blair, C., Bracken, J. T., Koochekian, N., Schield, D. R., ... & Jezkova, T. (2021). The effects of climate and demographic history in shaping genomic variation across populations of the Desert Horned Lizard (*Phrynosoma platyrhinos*). *Molecular Ecology*, 30(18), 4481-4496.

### Examples

```
data("HornedLizard_Pop")
data("HornedLizard_VCF")
Test <- Div_stats(VCF = HornedLizard_VCF, pops = HornedLizard_Pop,
ploidy = 2, write = FALSE)
```

---

Network_map	<i>A function to map statistics (i.e., genetic differentiation) between points as a network on a map.</i>
-------------	---

---

### Description

A function to map statistics (i.e., genetic differentiation) between points as a network on a map.

**Usage**

```

Network_map(
  dat,
  pops,
  neighbors,
  col,
  statistic = NULL,
  breaks = NULL,
  Lat_buffer = 1,
  Long_buffer = 1,
  Latitude_col = NULL,
  Longitude_col = NULL
)

```

**Arguments**

dat	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. If it is a csv, the 1st row should contain the individual/population names. The columns should also be named in this fashion.
pops	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The columns should be named Sample, containing the sample IDs; Population indicating the population assignment of the individual; Long, indicating the longitude of the sample; Lat, indicating the latitude of the sample. Alternatively, see the Longitude_col and Latitude_col arguments.
neighbors	Numeric or character. The number of neighbors to plot connections with, or the specific relationship that you want to visualize. Names should match those in the population assignment file and be separated by an underscore. If I want to visualize the relationship between East and West, for example, I would set neighbors = "East_West".
col	Character vector indicating the colors you wish to use for plotting.
statistic	Character indicating the statistic being plotted. This will be used to title the legend. The legend title will be blank if left as NULL.
breaks	Numeric. The breaks used to generate the color ramp when plotting. Users should supply 3 values if custom breaks are desired.
Lat_buffer	Numeric. A buffer to customize visualization.
Long_buffer	Numeric. A buffer to customize visualization.
Latitude_col	Numeric. The number of the column indicating the latitude for each sample. If this is not null, PopGenHelpR will use this column instead of looking for the Lat column.
Longitude_col	Numeric. The number of the column indicating the longitude for each sample. If this is not null, PopGenHelpR will use this column instead of looking for the Long column.

**Value**

A list containing the map and the matrix used to plot the map.



**Author(s)**

Keaka Farleigh

**Examples**

```
data(Fst_dat)
Fst <- Fst_dat[[1]]
Loc <- Fst_dat[[2]]
Test <- Network_map(dat = Fst, pops = Loc,
neighbors = 2,col = c('#4575b4', '#91bfd8', '#e0f3f8','#fd8d3c','#fc4e2a'),
statistic = "Fst", Lat_buffer = 1, Long_buffer = 1)
```

---

Pairwise\_heatmap      *A function to plot a heatmap from a symmetric matrix.*

---

**Description**

A function to plot a heatmap from a symmetric matrix.

**Usage**

```
Pairwise_heatmap(dat, statistic, col = NULL)
```

**Arguments**

dat	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. If it is a csv, the 1st row should contain the individual/population names. The columns should also be named in this fashion.
statistic	Character indicating the statistic represented in the matrix, this will be used to label the plot.
col	Character vector indicating the colors to be used in plotting. The vector should contain two colors, the first will be the low value, the second will be the high value.

**Value**

A heatmap plot

**Examples**

```
#' data(Fst_dat)
Fst <- Fst_dat[[1]]
Fstat_plot <- Pairwise_heatmap(dat = Fst, statistic = 'FST')
```

---

PCA *A function to perform principal component analysis (PCA) on genetic data. Loci with missing data will be removed prior to PCA.*

---

### Description

A function to perform principal component analysis (PCA) on genetic data. Loci with missing data will be removed prior to PCA.

### Usage

```
PCA(  
  data,  
  center = TRUE,  
  scale = FALSE,  
  missing_value = NA,  
  write = FALSE,  
  prefix = NULL  
)
```

### Arguments

data	Character. String indicating the name of the vcf file, geno file or vcfR object to be used in the analysis.
center	Boolean. Whether or not to center the data before principal component analysis.
scale	Boolean. Whether or not to scale the data before principal component analysis.
missing_value	Character. String indicating missing data in the input data. It is assumed to be NA, but that may not be true (is likely not) in the case of geno files.
write	Boolean. Whether or not to write the output to files in the current working directory. There will be two files, one for the individual loadings and the other for the percent variance explained by each axis.
prefix	Character. Optional argument. String that will be appended to file output. Please provide a prefix if write is set to TRUE.

### Value

A list containing two elements: the loadings of individuals on each principal component and the variance explained by each principal component.

### Author(s)

Keaka Farleigh

**Examples**

```
data("HornedLizard_VCF")
Test <- PCA(data = HornedLizard_VCF)
```

---

Piechart\_map                      *Plot a map of ancestry pie charts.*

---

**Description**

Plot a map of ancestry pie charts.

**Usage**

```
Piechart_map(
  anc.mat,
  pops,
  K,
  plot.type = "all",
  col,
  piesize = 0.35,
  Lat_buffer,
  Long_buffer,
  Latitude_col = NULL,
  Longitude_col = NULL
)
```

**Arguments**

anc.mat	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The first column should be the names of each sample/population, followed by the estimated contribution of each cluster to that individual/pop.
pops	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The columns should be named Sample, containing the sample IDs; Population indicating the population assignment of the individual, population and sample names must be the same type (i.e., both numeric or both characters); Long, indicating the longitude of the sample; Lat, indicating the latitude of the sample. Alternatively, see the Longitude_col and Latitude_col arguments.
K	Numeric. The number of genetic clusters in your data set, please contact the package authors if you need help doing this.
plot.type	Character string. Options are all, individual, and population. All is default and recommended, this will plot a piechart map for both the individuals and populations.
col	Character vector indicating the colors you wish to use for plotting.

piesize	Numeric. The radius of the pie chart for ancestry mapping.
Lat_buffer	Numeric. A buffer to customize visualization.
Long_buffer	Numeric. A buffer to customize visualization.
Latitude_col	Numeric. The number of the column indicating the latitude for each sample. If this is not null, PopGenHelpR will use this column instead of looking for the Lat column.
Longitude_col	Numeric. The number of the column indicating the longitude for each sample. If this is not null, PopGenHelpR will use this column instead of looking for the Long column.

**Value**

A list containing your plots and the data frames used to generate the plots.

**Author(s)**

Keaka Farleigh

**Examples**

```
data(Q_dat)
Qmat <- Q_dat[[1]]
rownames(Qmat) <- Qmat[,1]
Loc <- Q_dat[[2]]
Test_all <- Piechart_map(anc.mat = Qmat, pops = Loc, K = 5,
plot.type = 'all', col = c('#d73027', '#fc8d59', '#e0f3f8', '#91bfdb', '#4575b4'), piesize = 0.35,
Lat_buffer = 1, Long_buffer = 1)
```

---

Plot\_ancestry

***WARNING! This function has been deprecated and is no longer supported. Please use the Ancestry\_barchart and Piechart\_map functions. Plot an ancestry matrix and map of ancestry pie charts.***

---

**Description**

**WARNING! This function has been deprecated and is no longer supported. Please use the Ancestry\_barchart and Piechart\_map functions.** Plot an ancestry matrix and map of ancestry pie charts.

**Usage**

```
Plot_ancestry(
  anc.mat,
  pops,
  K,
  plot.type = "all",
```

```

    col,
    piesize = 0.35,
    Lat_buffer,
    Long_buffer
  )

```

### Arguments

anc.mat	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The first column should be the names of each sample/population, followed by the estimated contribution of each cluster to that individual/pop.
pops	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The columns should be named Sample, containing the sample IDs; Population indicating the population assignment of the individual, population and sample names must be the same type (i.e., both numeric or both characters); Long, indicating the longitude of the sample; Lat, indicating the latitude of the sample.
K	Numeric. The number of genetic clusters in your data set, please contact the package authors if you need help doing this.
plot.type	Character string. Options are all, individual, and population. All is default and recommended, this will plot a barchart and piechart map for both the individuals and populations.
col	Character vector indicating the colors you wish to use for plotting.
piesize	Numeric. The radius of the pie chart for ancestry mapping.
Lat_buffer	Numeric. A buffer to customize visualization.
Long_buffer	Numeric. A buffer to customize visualization.

### Value

A list containing your plots and the data frames used to generate the plots.

### Examples

```

data(Q_dat)
Qmat <- Q_dat[[1]]
rownames(Qmat) <- Qmat[,1]
Loc <- Q_dat[[2]]
Test_all <- Plot_ancestry(anc.mat = Qmat, pops = Loc, K = 5,
plot.type = 'all', col <- c('red', 'maroon', 'navy', 'cyan', 'blue'), piesize = 0.35,
Lat_buffer = 1, Long_buffer = 1)

```

---

Plot\_coordinates      *A function to plot coordinates on a map.*

---

### Description

A function to plot coordinates on a map.

### Usage

```
Plot_coordinates(
  dat,
  col = c("#A9A9A9", "#000000"),
  size = 3,
  Lat_buffer = 1,
  Long_buffer = 1,
  Latitude_col = NULL,
  Longitude_col = NULL
)
```

### Arguments

dat	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The coordinates of each row should be indicated by columns named Longitude and Latitude. Alternatively, see the Latitude_col and Longitude_col arguments.
col	Character vector indicating the colors you wish to use for plotting, two colors are allowed. The first color will be the fill color, the second is the outline color. For example, if I want red points with a black outline I would set col = c("#FF0000", "#000000").
size	Numeric. The size of the points to plot.
Lat_buffer	Numeric. A buffer to customize visualization.
Long_buffer	Numeric. A buffer to customize visualization.
Latitude_col	Numeric. The number of the column indicating the latitude for each sample. If this is not null, PopGenHelpR will use this column instead of looking for the Latitude column.
Longitude_col	Numeric. The number of the column indicating the longitude for each sample. If this is not null, PopGenHelpR will use this column instead of looking for the Longitude column.

### Value

A ggplot object.

### Author(s)

Keaka Farleigh

**Examples**

```
data("HornedLizard_Pop")
Test <- Plot_coordinates(HornedLizard_Pop)
```

---

Point\_map

*A function to map statistics as colored points on a map.*


---

**Description**

A function to map statistics as colored points on a map.

**Usage**

```
Point_map(
  dat,
  statistic,
  size = 3,
  breaks = NULL,
  col,
  out.col = NULL,
  Lat_buffer = 1,
  Long_buffer = 1,
  Latitude_col = NULL,
  Longitude_col = NULL
)
```

**Arguments**

dat	Data frame or character string that supplies the input data. If it is a character string, the file should be a csv. The first column should be the statistic to be plotted. The coordinates of each row should be indicated by columns named Longitude and Latitude. Alternatively, see the Longitude_col and Latitude_col arguments.
statistic	Character string. The statistic to be plotted.
size	Numeric. The size of the points to plot.
breaks	Numeric. The breaks used to generate the color ramp when plotting. Users should supply 3 values if custom breaks are desired.
col	Character vector indicating the colors you wish to use for plotting, three colors are allowed (low, mid, high). The first color will be the low color, the second the middle, the third the high.
out.col	Character. A color for outlining points on the map. There will be no visible outline if left as NULL.
Lat_buffer	Numeric. A buffer to customize visualization.
Long_buffer	Numeric. A buffer to customize visualization.

Latitude_col	Numeric. The number of the column indicating the latitude for each sample. If this is not null, PopGenHelpR will use this column instead of looking for the Latitude column.
Longitude_col	Numeric. The number of the column indicating the longitude for each sample. If this is not null, PopGenHelpR will use this column instead of looking for the Longitude column.

**Value**

A list containing maps and the data frames used to generate them.

**Author(s)**

Keaka Farleigh

**Examples**

```
data(Het_dat)
Test <- Point_map(Het_dat, statistic = "Heterozygosity")
```

---

Private.alleles      *A function to estimate the number of private alleles in each population.*

---

**Description**

A function to estimate the number of private alleles in each population.

**Usage**

```
Private.alleles(
  data,
  pops,
  write = FALSE,
  prefix = NULL,
  population_col = NULL,
  individual_col = NULL
)
```

**Arguments**

data	Character. String indicating the name of the vcf file or vcfR object to be used in the analysis.
pops	Character. String indicating the name of the population assignment file or dataframe containing the population assignment information for each individual in the data. This file must be in the same order as the vcf file and include columns specifying the individual and the population that individual belongs to. The first column should contain individual names and the second column should indicate



the population assignment of each individual. Alternatively, you can indicate the column containing the individual and population information using the `individual_col` and `population_col` arguments.

<code>write</code>	Boolean. Optional argument indicating Whether or not to write the output to a file in the current working directory. This will output to files; 1) the table of private allele counts per population (named <code>prefix_PrivateAlleles_countperpop</code> ) and 2) metadata associated with the private alleles (named <code>prefix_PrivateAlleles_metadata</code> ). Please supply a prefix if you write files to your working directory as a best practice.
<code>prefix</code>	Character. Optional argument indicating a string that will be appended to file output. Please set a prefix if <code>write</code> is TRUE.
<code>population_col</code>	Numeric. Optional argument (a number) indicating the column that contains the population assignment information.
<code>individual_col</code>	Numeric. Optional argument (a number) indicating the column that contains the individuals (i.e., sample name) in the data.

### Value

A list containing the count of private alleles in each population and the metadata for those alleles. The metadata is a list that contains the private allele and locus name for each population.

### Author(s)

Keaka Farleigh

### Examples

```
data("HornedLizard_Pop")
data("HornedLizard_VCF")
Test <- Private.alleles(data = HornedLizard_VCF, pops = HornedLizard_Pop, write = FALSE)
```

---

Q_dat	<i>A list representing a q-matrix and the locality information associated with the qmatrix</i>
-------	--

---

### Description

List with two elements

### Usage

```
data(Q_dat)
```

**Format**

A list with two elements:

**Qmat** A q-matrix with 6 columns and 30 rows, the first column lists the sample name and the remaining 5 represent the contribution a genetic cluster to that individuals ancestry

**Loc\_dat** The locality information for each individual in the q-matrix

...

**Source**

Data was generated by package authors.

**Examples**

```
data(Q_dat)
Qmat <- Q_dat[[1]]
rownames(Qmat) <- Qmat[,1]
Loc <- Q_dat[[2]]
Test_all <- Plot_ancestry(anc.mat = Qmat, pops = Loc, K = 5,
plot.type = 'all', col <- c('red', 'maroon', 'navy', 'cyan', 'blue'), piesize = 0.35,
Lat_buffer = 1, Long_buffer = 1)
```

# Index

## \* datasets

Fst\_dat, [10](#)  
Het\_dat, [13](#)  
HornedLizard\_Pop, [14](#)  
HornedLizard\_VCF, [15](#)  
Q\_dat, [25](#)

Ancestry\_barchart, [2](#)

Dif\_stats, [5](#)  
Dif\_Stats\_Map, [6](#)  
Differentiation, [3](#)  
Div\_stats, [8](#)  
Div\_Stats\_Map, [9](#)

Fst\_dat, [10](#)

Het\_dat, [13](#)  
Heterozygosity, [11](#)  
HornedLizard\_Pop, [14](#)  
HornedLizard\_VCF, [15](#)

Network\_map, [15](#)

Pairwise\_heatmap, [17](#)  
PCA, [18](#)  
Piechart\_map, [19](#)  
Plot\_ancestry, [20](#)  
Plot\_coordinates, [22](#)  
Point\_map, [23](#)  
Private.alleles, [24](#)

Q\_dat, [25](#)