# Package ‘tropicalHAR’

April 10, 2023

**Type** Package

**Title** Functions for Statistical Inference in Tropical Geometry

**Version** 0.1.0

**Author** David Barnhill, Ruriko Yoshida, Keiji Miura

**Maintainer** David Barnhill <david.barnhill@nps.edu>

**Description** This package provides functions for use in statistical inference for data existing in the tropical projective torus. The package focuses on the use of Markov chain Monte Carlo Hit-and-Run methods to sample points over sample spaces defined by tropical polytopes. Additionally, this package contains supervised and unsupervised learning methods to be applied to data over the tropical projective torus.

**License** What license is it under?

**Encoding** UTF-8

**Imports** Rdpack
MASS
ape
phangorn
lpSolve
lpSolveAPI
miscTools
Matrix
rcdd
rgl
magick
gtools
combinat
pracma
RcppAlgos
Rfast

**RdMacros** Rdpack

**RoxygenNote** 7.2.3

**NeedsCompilation** no

## R topics documented:

---

`bw.nn` *Nearest neighbor bandwidth calculation used adapted from the package kdetrees.*

---

## Description

Bandwidth calculation based on nearest neighbor methods for use in non-parametric kernel density estimation over the space of phylogenetic trees.

## Usage

```
bw.nn(x,prop=0.2,tol=1e-6)
```

## Arguments

| | |
|---|---|
| `x` | |
| `prop` | Proportion of the set of trees used to define the neighborhood around x. |
| `tol` | Tolerance: stopping condition to determine bandwidth. Default as shown. |

## Value

Bandwidth value used for kernel density estimation over phylogenetic trees.

## Note

The package kdetrees has been archived. We therefore include it here.

## References

G. Weyenberg, P. Huggins, C. Schardl, D.K. Howe, and R. Yoshida. kdetrees: Nonparametric estimation of phylogenetic tree distributions. Bioinformatics, 2014.

## Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

## `Check.onto.Tpoly` *Check whether a tropical point is in a tropical polytope.*

### Description

Check whether a tropical point is in a tropical polytope.

### Usage

```
Check.onto.Tpoly(D_s, x)
```

### Arguments

| | |
|---|---|
| `D_s` | Matrix of points defining a tropical polytope. |
| `D` | Point in tropical projective space. |

### Value

Logical indicating whether a point in tropical projective space is in a tropical polytope (1=yes or 0=no).

### Note

Point of interest does not need to be normalized.

### Author(s)

Ruriko Yoshida

### Examples

```
D_s<-matrix(c(0,0,0,0,2,5,0,3,1),3,3,TRUE)
D<-c(0,6,2)
Check.onto.Tpoly(D_s,D)
```

## `draw.tpolytope.3d` *Draw a 3D tropical polytope.*

### Description

Draw a 3D tropical polytope.

### Usage

```
draw.tpolytope.3d(D, c)
```

### Arguments

| | |
|---|---|
| `c` | Color of polytope. |
| `D_s` | Matrix of points defining a tropical polytope. |

### Value

Rendering of a 3D tropical polytope.

### Author(s)

Ruriko Yoshida

### Examples

```
D_s<-matrix(c(0,0,0,0,0,1,2,5,0,1,3,1,0,2,5,10),4,4,TRUE)
c<-'red'
draw.tpolytope.3d(D_s,c)
```

---

## HAR.TLineSeg *Hit-and-Run sampler over a tropical line segment.*

---

### Description

Hit-and-Run sampler over a tropical line segment.

### Usage

```
HAR.TLineSeg(D1, D2)
```

### Arguments

| | |
|---|---|
| D1 | First tropical point. |
| D2 | Second tropical point. |

### Value

Single unnormalized sample point on line segment between D1 and D2.

### Author(s)

Ruriko Yoshida

### References

Yoshida, R., Miura, K., & Barnhill, D. (2022). *Hit and Run Sampling from Tropically Convex Sets*.

### Examples

```
D1<-c(0,0,0)
D2<-c(0,3,5)
x<-HAR.TLineSeg(D1,D2)
```

HAR.TLineSeg.Norm *Gaussian Sampling Over Tropical Line Segment*

## Description

This function samples a point on a line segment normally about a center of mass point mu.

## Usage

```
HAR.TLineSeg.Norm(D1, D2,mu,stdev)
```

## Arguments

| | |
|---|---|
| D1 | End point of a tropical line segment. |
| D2 | End point of a tropical line segment. |
| mu | Center of mass point on the line segment |
| stdev | Standard deviation to sample points around mu. |

## Value

Point sampled normally about the center of mass point on the line segment.

## Author(s)

David Barnhill

## Examples

```
D1<-c(0,0,0)
D2<-c(0,3,7)
mu<-c(0,1,4)
s<-.5
har_line<-matrix(0,500,3,TRUE)
for (i in 1:500){
  x<-HAR.TLineSeg.Norm(D1, D2,mu,s)
  har_line[i,]<-x
}
plot(c(0,3),c(0,7),asp=1)
points(har_line[,2],har_line[,3],col='blue')
hist(apply(har_line,1,function(x) trop.dist(x,c(0,0,4))))
```

HAR.TLineSeg.Norm_Poly

*Gaussian Sampling About a Projection of a Center of Mass Point onto a Tropical Line Segment.*

## Description

This Hit-and-Run sampler samples a point according to a Gaussian distribution about a center of mass point on a tropical line segment. The center of mass point on the line segment is selected uniformly on the interval of projections of the centers of mass of a tropical polytope. The resulting sampled point is selected according to a Gaussian distribution about this point. This is a helper function for the normal extrapolation sampler.

## Usage

```
HAR.TLineSeg.Norm_Poly(D1,D2,mu,stdev)
```

## Arguments

| | |
|---|---|
| D1 | End point of a tropical line segment. |
| D2 | End point of a tropical line segment. |
| mu | Center of mass point of the tropical polytope. |
| stdev | Standard deviation to sample points around mu. |

## Value

Point sampled according to a Gaussian distribution according to the tropical metric.

## Author(s)

David Barnhill

## See Also

TropicalPolytope.extrapolation.HAR_NORM

## Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

`hyper3d_max` *Max-plus Tropical Hyperplane*

## Description

2D/3D rendering of max-plus hyperplanes for a given set of tropical points.

## Usage

```
hyper3d_max(D,di,mi,ma,plt=FALSE)
```

## Arguments

| | |
|---|---|
| D | Matrix of points defining the polytope P. Points are row vectors. |
| di | Length of sheet emanating from hyperplane apex. |
| mi | Minmium axis value for plotting. |
| ma | Maximum axis value for plotting. |
| plt | Logical used to plot hyperplane(s) on coordinate system. |

## Value

Rendering of a 2-dimensional or 3-dimensional collection of tropical hyperplanes.

## Author(s)

David Barnhill

## Examples

```
2D Example
D<-matrix(c(0,0,0,0,3,1,0,2,5),3,3,TRUE)
di<-1
mi<--0.5
ma<-5.5
hyper3d_max(D,di,mi,ma,plt=TRUE)

3D Example
D<-matrix(c(0,0,0,0,0,1,3,1,0,1,2,5,0,2,5,10),4,4,TRUE)
di<-3
mi<--0.5
ma<-10.5
hyper3d_max(D,di,mi,ma,plt=TRUE)
```

hyper3d_min *2D/3D rendering of min-plus hyperplanes for a given set of tropical points.*

## Description

2D/3D rendering of min-plus hyperplanes for a given set of tropical points.

## Usage

```
hyper3d_min(D, di, mi, ma, plt = FALSE)
```

## Arguments

| | |
|---|---|
| D | Matrix of points defining the polytope P. Points are row vectors. |
| di | Length of sheet emanating from hyperplane apex. |
| mi | Minmium axis value for plotting. |
| ma | Maximum axis value for plotting. |
| plt | Logical used to plot hyperplane(s) on coordinate system. |

## Value

Rendering of 2D/3D min-plus hyperplanes for the given set of points in tropical projective space.

## Examples

```
2D Example
D<-matrix(c(0,0,0,0,3,1,0,2,5),3,3,TRUE)
di<-1
mi<--0.5
ma<-5.5
hypers3d_min(D,di,mi,ma,plt=TRUE)

3D Example
D<-matrix(c(0,0,0,0,0,1,3,1,0,1,2,5,0,2,5,10),4,4,TRUE)
di<-3
mi<--0.5
ma<-10.5
hyper3d_min(D,di,mi,ma,plt=TRUE)
```

kernel.ultrametric *Kernel density estimation function for ultrametrics.*

## Description

A non-parametric density estimator with the tropical metric over the space of ultrametrics that mimics a classical kernel density estimator (KDE) formulated using a normal density kernel.

### Usage

```
kernel.ultrametric(x, mu, s)
```

### Arguments

| | |
|---|---|
| `x` | A sample ultrametric. |
| `mu` | Center point about which the sample is obtained. |
| `s` | Bandwidth associated with KDE |

### Value

The ultrametric kernel result.

### References

Yoshida, R., Barnhill, D., Miura, K., & Howe, D. (2022). *Tropical Density Estimation of Phylogenetic Trees*

Ruriko Yoshida

---

## `max_ins_ball` *Defining the maximum inscribed ball of a tropical polytope.*

---

### Description

Defining the maximum inscribed ball of a tropical polytope.

### Usage

```
max_ins_ball(A)
```

### Arguments

| | |
|---|---|
| `A` | A matrix of points defining the polytope, P. |

### Details

This matters because.

### Value

A list consisting of the following components

| | |
|---|---|
| `rad` | The radius of the maximum inscribed ball inside of P. |
| `cent` | The center of the maximum inscribed ball inside of P. |

### References

Yoshida, R., Miura, K., & Barnhill, D. (2023). *Maximum Inscribed and Minimum Enclosing Tropical Balls of Tropical Polytopes and Applications to Volume Estimation and Uniform Sampling*.

## Examples

```
P<-matrix(c(0,0,0,0,2,5,0,3,1),3,3,TRUE)
L<-max_ins_ball(P)
```

---

`min_enc_ball` *Defining the minimum encompassing ball of a tropical polytope.*

---

## Description

Defining the minimum encompassing ball of a tropical polytope.

## Usage

```
min_enc_ball(A)
```

## Arguments

| | |
|---|---|
| `A` | A matrix of points defining the polytope, P. |

## Details

This matters because.

## Value

A list consisting of the following components

| | |
|---|---|
| `cent` | The center of the minimum encompassing ball surrounding P. |
| `rad` | The radius of the minimum encompassing ball surrounding P. |

## References

Yoshida, R., Miura, K., & Barnhill, D. (2023). *Maximum Inscribed and Minimum Enclosing Tropical Balls of Tropical Polytopes and Applications to Volume Estimation and Uniform Sampling*.

## Examples

```
P<-matrix(c(0,0,0,0,2,5,0,3,1),3,3,TRUE)
L<-min_enc_ball(P)
```

normaliz.polytope *Normalize points defining a tropical polytope.*

## Description

Normalize points defining a tropical polytope.

## Usage

```
normaliz.polytope(DD)
```

## Arguments

| | |
|---|---|
| DD | Matrix of tropical points defining a tropical polytope. |

## Value

Matrix of normalized (first coordinate equals zero) tropical points.

## Author(s)

Ruriko Yoshida

## Examples

```
DD<-matrix(c(1,1,1,2,4,7,3,6,4),3,3,TRUE)
DD<-normaliz.polytope(DD)
```

normaliz.tree *Normalize a phylogenetic tree.*

## Usage

```
normaliz.tree(D, h = 1)
```

## Arguments

| | |
|---|---|
| D | First tropical point. |
| h | Height of equidistant tree |

## Author(s)

Ruriko Yoshida

---

`normaliz.vector` *Normalize a tropical vector.*

---

## Description

Normalize a tropical vector.

## Usage

```
normaliz.vector(D1)
```

## Arguments

| | |
|---|---|
| `D1` | Tropical point. |

## Value

Normalized tropical point.

## Author(s)

Ruriko Yoshida

## Examples

```
D1<-c(3,4,2)
D<-normaliz.vector(D1)
```

---

`normaliz.vectors` *Normalize a matrix of tropical vector.*

---

## Description

Normalize a matrix of tropical vector.

## Usage

```
normaliz.vectors(DD)
```

## Arguments

| | |
|---|---|
| `DD` | Matrix of tropical points with points being row vectors. |

## Value

Matrix of normalized (first coordinate equals zero) tropical points.

## Author(s)

Ruriko Yoshida

### Examples

```
DD<-matrix(c(1,1,1,2,4,7,3,6,4),3,3,TRUE)
DD<-normaliz.vectors(DD)
```

---

## normalize.ultrametrices

*Normalize a matrix of ultrametrics*

---

### Description

This function normalizes a set of ultrametrics by subtracting the first coordinate of each row vector from every coordinate in the associated row vector.

### Usage

```
normalize.ultrametrices(D)
```

### Arguments

| | |
|---|---|
| D | Matrix of ultrametrics (row vectors). |

### Value

Matrix of normalized ultrametrics.

### Author(s)

Ruriko Yoshida

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

## plot.tree.PCA *Plot of PCA tree.*

### Usage

```
plot.tree.PCA(tree.list, freq, file="Tree_Topology.pdf")
```

### Arguments

`tree.list`

`freq`

`file` String; Output file for the plot. Defaults to PDF.

### Value

Rendering of PCA tree saved directly to file.

### Author(s)

Ruriko Yoshida

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

## plot.trop.triangle *Plot a tropical triangle.*

### Description

Plots a tropical triangle and saves to pdf.

### Usage

```
plot.trop.triangle(D)
```

### Arguments

`D` Matrix of points (row vectors) representing the points defining a tropical triangle.

### Value

Rendering of a tropical triangle saved to file.

### Author(s)

Ruriko Yoshida

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

---

## plot.trop.triangle.w.points

*Plot points of a tropical triangle.*

---

### Description

Plot the points of a tropical triangle.

### Usage

```
plot.trop.triangle.w.points(S, D, file="Tropical_Triangle.pdf")
```

### Arguments

`S`

`D`

`file`

### Value

Rendering of a the points of a tropical triangle.

### Author(s)

Ruriko Yoshida

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

`plot.trop.triangle.w.top` *Plot a Tropical Triangle with the Top.*

## Usage

```
plot.trop.triangle.w.top(S, D, tree.type, file="Tropical_Triangle_Trees.pdf")
```

## Author(s)

Ruriko Yoshida

`Points.TLineSeg` *Extract k equally distanced points on a tropical line segment.*

## Description

Extract k equally distanced points on a tropical line segment.

## Usage

```
Points.TLineSeg(D1, D2, k = 20)
```

## Arguments

| | |
|---|---|
| D1 | First tropical point. |
| D2 | Second tropical point. |
| k | Number of points. |

## Value

Matrix of k tropical points with points being row vectors.

## Author(s)

Ruriko Yoshida

## Examples

```
D1<-c(0,0,0)
D2<-c(0,3,5)
N<-Points.TLineSeg(D1,D2,k=20)
```

## polytope_iso

### Usage

```
polytope_iso(D, P)
```

### Arguments

D

P

### Author(s)

Ruriko Yoshida

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

## pre.draw.tpolytope.3d *Pre-draw function to display a 3D tropical polytope.*

### Description

Pre-draw function to display a 3D tropical polytope.

### Usage

```
pre.draw.tpolytope.3d(D, v, c)
```

### Arguments

| | |
|---|---|
| D | Tropical point. |
| v | Scalar indicating number of tropical points to evaluate. |
| c | Color for displayed polytope. |

### Note

This function is used in conjunction with draw.tpolytope.3d.

### Author(s)

Ruriko Yoshida

`pre.pplot.pro` *Helper function for PCA plotting.*

### Description

Helper function for tropical triangle plotting.

### Usage

```
pre.pplot.pro(S, D)
```

### Arguments

`S`

`D`

### Author(s)

Ruriko Yoshida

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

`project_pi` *Project a point onto a tropical polytope.*

### Description

Project a point onto a tropical polytope.

### Usage

```
project_pi(D_s, D)
```

### Arguments

| | |
|---|---|
| `D_s` | Matrix of points defining a tropical polytope. |
| `D` | Tropical point. |

### Value

Tropical point representing the projection of D onto D_s.

## Author(s)

Ruriko Yoshida

## Examples

```
D_s<-matrix(c(0,0,0,0,2,5,0,3,1),3,3,TRUE)
D<-c(0,5,9)
pi<-project_pi(D_s,D)
```

---

# pw.trop.dist

---

## Usage

```
pw.trop.dist(D1, D2)
```

## Arguments

D1

D2

## Author(s)

Ruriko Yoshida

## References

Yoshida, R., Barnhill, D., Miura, K., & Howe, D. (2022). *Tropical Density Estimation of Phylogenetic Trees.*

## Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

`rounding` *Rounding a tropical polytope.*

## Description

Rounding a tropical polytope.

## Usage

```
rounding(P)
```

## Arguments

| | |
|---|---|
| P | A square matrix of points defining the polytope, P. |

## Details

This matters because.

## Value

Matrix of points where points are rows that define the pseudo-vertices of a full-dimensional polytope.

## References

Yoshida, R., Miura, K., & Barnhill, D. (2023). *Maximum Inscribed and Minimum Enclosing Tropical Balls of Tropical Polytopes and Applications to Volume Estimation and Uniform Sampling*.

Cousins, B.R., & Vempala, S.S. (2016). *A practical volume algorithm.* Mathematical Programming Computation, 8, 133-160.

## Examples

```
P<-matrix(c(0,0,0,0,1,0,0,0,1),3,3,TRUE)
P0<-rounding(P)
```

`Sum.Residuals`

## Description

Calculates the sum of the residuals for tropical PCA.

## Usage

```
Sum.Residuals(S, D)
```

## Arguments

| | |
|---|---|
| S | Set of vertices of a tropical polytope. |
| D | Data in matrix form. |

### Value

Value of the sum of the residuals where each residual is the tropical distance between a data point in D and the projection of that point on the tropical polytope S.

### Author(s)

Ruriko Yoshida

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

---

T.SVM *Tropical Support Vector Machine*

---

### Description

This sampler uses the tropical HAR extrapolation sampler to find a tropical hyperplane to find the best fit tropical hyperplane that separates data based on classifcation labels.

### Usage

```
T.SVM(V,clses,L=1000,tst=.8,st=100)
```

### Arguments

| | |
|---|---|
| V | Matrix of data without the dependent variable. |
| clses | Vector of the original data representing the dependent variable (classes) in the data. |
| L | Number of points to sample. |
| tst | Train set division. Expressed in proportions. |

### Value

Point representing the tropical hyperplane that minimizes the cost function.

### Author(s)

David Barnhill

## Examples

```
set.seed(135)
e <- 100
n <- 10
N <- 100
s <- 10
x <- rbind(
  rmvnorm(n, mu = c(5, -5, rep(0, e - 2)), sigma = diag(s, e)),
  rmvnorm(n, mu = c(-5, 5, rep(0, e - 2)), sigma = diag(s, e))
)
yy <- as.factor(c(rep(1, n), rep(2, n)))
newx <- rbind(
  rmvnorm(N, mu = c(5, -5, rep(0, e - 2)), sigma = diag(s, e)),
  rmvnorm(N, mu = c(-5, 5, rep(0, e - 2)), sigma = diag(s, e))
)
newy <- as.factor(rep(c(1, 2), each = N))

Hyp<-T.SVM(iris3,iris3[,ncol(iris3)])

data(iris)
specs<-as.vector(unique(iris$Species))
iris2<-iris
iris2$Species<-as.numeric(iris$Species)
iris3<-cbind(replicate(kk,rep(0,nrow(wine1))),iris2)

Hyp<-T.SVM(iris3,iris3[,ncol(iris3)])
```

---

# tball_volume — *Calculating the volume of a tropical ball.*

---

## Description

Calculate the volume of a ball in the tropical projective torus.

## Usage

```
tball_volume(r, d)
```

## Arguments

| | |
|---|---|
| r | Radius of a tropical ball. |
| d | Dimension of tropical ball. |

## Details

This matters because.

## Value

Volume of a tropical ball with radius r, and dimension, d.

## References

Yoshida, R., Miura, K., & Barnhill, D. (2023). *Maximum Inscribed and Minimum Enclosing Tropical Balls of Tropical Polytopes and Applications to Volume Estimation and Uniform Sampling*.

## Examples

```
r<-1.5
d<-3
V<-tball_volume(r,d)
```

---

`tdets` *Calculating the tropical determinant of a set of points in tropical projective space.*

---

## Description

Calculating the tropical determinant of a set of points in tropical projective space.

## Usage

```
tdets(P)
```

## Arguments

| | |
|---|---|
| `P` | Matrix of points defining the polytope, P. Points are row vectors. |

## Details

This matters because.

## Value

List with the following components

| | |
|---|---|
| `tdet` | Value of tropical determinant. |
| `P` | Matrix of permuted points (by row) from the original matrix P such that each point's contribution to the tropical determinant is on the diagonal. |

## References

Joswig, M. (2010). *Essentials of Tropical Combinatorics*.

## Examples

```
P<-matrix(c(0,0,0,0,2,5,0,3,1),3,3,TRUE)
L<-tdets(P)
```

## TLineSeg — *Construct a max-plus line segment between two tropical points.*

### Description

Construct a max-plus line segment between two tropical points.

### Usage

```
TLineSeg(D1, D2)
```

### Arguments

| | |
|---|---|
| D1 | First tropical point. |
| D2 | Second tropical point. |

### Value

List of unnormalized line segment bendpoints.

### Author(s)

Ruriko Yoshida

### Examples

```
D1<-c(0,0,0)
D2<-c(0,3,5)
L<-TLineSeg(D1,D2)
```

## TLineSeg_min — *Construct a min-plus line segment between two tropical points.*

### Description

Construct a min-plus line segment between two tropical points.

### Usage

```
TLineSeg_min(D1, D2)
```

### Arguments

| | |
|---|---|
| D1 | First tropical point. |
| D2 | Second tropical point. |

### Value

List of unnormalized line segment bendpoints.

### Author(s)

David Barnhill

### Examples

```
D1<-c(0,0,0)
D2<-c(0,3,5)
L<-TLineSeg_min(D1,D2)
```

---

## tree.topology.type

---

### Usage

```
tree.topology.type(D, n, label)
```

### Arguments

| | |
|---|---|
| D | Matrix where rows are observations. |
| n | Number of leaves. |
| label | Set of labels for leaves. |

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

---

## trop.dist *Calculate distance between two tropical points.*

---

### Description

Calculate distance between two tropical points.

### Usage

```
trop.dist(D1, D2)
```

### Arguments

| | |
|---|---|
| D1 | First tropical point. |
| D2 | Second tropical point. |

## Value

Tropical distance between two points.

## Author(s)

Ruriko Yoshida

## Examples

```
D1<-c(0,1,1)
D2<-c(0,2,2)
trop.dist(D1, D2)
```

---

`trop.dist.hyp_max` *Calculate distance from a point to a max-plus tropical hyperplane.*

---

## Description

Calculate distance from a point to a max-plus tropical hyperplane.

## Usage

```
trop.dist.hyp_max(D, v)
```

## Arguments

| | |
|---|---|
| `D` | Tropical point representing apex of max-plus hyperplane. |
| `v` | Point in tropical projective space. |

## Value

Distance from point to max-plus hyperplane.

## Author(s)

David Barnhill

## Examples

```
D<-c(0,0,0)
v<-c(0,2,1)
d<-trop.dist.hyp_max(D, v)
```

## trop.dist.hyp_min — *Calculate distance from a point to a min-plus tropical hyperplane.*

### Description

Calculate distance from a point to a min-plus tropical hyperplane.

### Usage

```
trop.dist.hyp_min(D, v)
```

### Arguments

| | |
|---|---|
| D | Tropical point representing apex of min-plus hyperplane. |
| v | Point in tropical projective space. |

### Value

Distance from point to min-plus hyperplane.

### Author(s)

David Barnhill

### Examples

```
D<-c(0,0,0)
v<-c(0,2,1)
d<-trop.dist.hyp_min(D, v)
```

## trop.Gaussian — *Hit-and-Run sampling around a center of mass.*

### Description

Hit-and-Run sampling around a center of mass.

### Usage

```
trop.Gaussian(D_s, x0, mu, s, n, I, k)
```

### Arguments

| | |
|---|---|
| D_s | Matrix of points defining a tropical polytope. |
| x0 | Starting point for the hit-and-run algorithm. |
| mu | Tropical point representing center of mass. |
| s | Distance parameter used in kernel function. |
| n | Sample size of points to be collected. |
| I | Number of iterations used for hit-and-run algorithm. |
| k | Cardinality of subset used in TropicalPolytope.extrapolation.HAR_v4 algorithm. |

## Value

Matrix of n points. Points are row vectors.

## Note

Point of interest does not need to be normalized.

## Author(s)

Ruriko Yoshida and David Barnhill

## Examples

```
D_s<-matrix(c(0,0,0,0,2,5,0,3,1),3,3,TRUE)
x0<-c(0,1,0)
mu<-c(0,1.5,1.5)
s<-.5
n<-1000
I<-50
k<-2
N<-trop.Gaussian(D_s, x0, mu, s, n, I, k)
```

---

# tropical.geodesic.dim.2

---

## Usage

```
tropical.geodesic.dim.2(D1, D2, flag = 0)
```

## Arguments

| | |
|---|---|
| D1 | |
| D2 | |
| flag | Logical |

## Author(s)

Ruriko Yoshida

## Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

---

`tropical.KDE` *Kernel density estimation over a space of phylogenetic trees.*

---

## Description

Kernel density estimation function over the space of phylogenetic trees for a given data set.

## Usage

```
tropical.KDE(D, n, sigma, h = 2)
```

## Arguments

| | |
|---|---|
| `D` | Data set with each row as an observation. |
| `n` | The number of leaves. |
| `h` | The height of the tree. |

## Value

Density estimate for a given data set.

## Author(s)

Ruriko Yoshida

## References

Yoshida, R., Barnhill, D., Miura, K., & Howe, D. (2022). *Tropical Density Estimation of Phylogenetic Trees.*

---

`tropical.PCA` *Tropical Principal Component Analysis*

---

## Usage

```
tropical.PCA(S, D, n, h, I = 1)
```

## Arguments

`S`

`D`

`n`

`h`

`I`

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

---

## tropical.PCA.Polytope

---

### Usage

```
tropical.PCA.Polytope(S, D, V, I = 1)
```

### Arguments

| | |
|---|---|
| S | |
| D | Set of vertices defining a tropical triangle. |
| V | Matrix of vertices of a tropical polytope larger than the tropical polytope defined by D. |
| I | Iterations. |

### Value

Returns a list

`Sum.Residuals(S.star, D):`
Sum of the residuals.

`S.star:`

`D1:`

### Author(s)

Ruriko Yoshida

### Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

---

TropicalPolytope.extrapolation
*Uniform Hit-and-run sampler for a tropical polytope.*

---

## Description

Hit-and-run sampler for a tropical polytope.

## Usage

```
TropicalPolytope.extrapolation.HAR_v4(D_s, x0, I = 1, k = dim(D_s)[2])
```

## Arguments

| | |
|---|---|
| D_s | Matrix of vertices representing the tropical convex hull of of a polytope, P. |
| x0 | Starting point in tropical projective space. |
| I | Number of iterations for the hit-and-run algorithm |
| k | Size of subset. Defaults to # of columns of D_s. |

## Value

A point sampled from a the polytope P.

## References

Yoshida, R., Miura, K., & Barnhill, D. (2022). *Hit and Run Sampling from Tropically Convex Sets*.

## Examples

```
D_s<-matrix(c(0,0,0,0,2,5,0,3,1),3,3,TRUE)
x_0<-c(0,2,1)
I=50
x=TropicalPolytope.extrapolation.HAR_v4(D_s,x0,I=I)
```

---

TropicalPolytope.extrapolation.HAR_NORM
*Gaussian HAR Sampler from a Tropical Polytope*

---

## Description

This function is a Hit-and-Run sampler that samples points from a tropical polytope in the tropical projective torus according to a Gaussian distribution based on the tropical metric. The original tropical polytope is defined by e points. Line segments are generated where end points are determined by the projection of a point onto the tropical polytope defined by e-1 vertices and the projection of the same point onto the remaining vertex. If the line segment intersects the boundary of the tropical polytope at more than two points, the line segment is truncated to only the portion of the line segment intersects the boundary at two points or, if the line segment follows the boundary of the tropical polytope, the line segment is truncated to the first bend point.

### Usage

```
TropicalPolytope.extrapolation.HAR_NORM(D_s, x0, I = 1,k=(ncol(D_s)-1),M,S)
```

### Arguments

| | |
|---|---|
| D_s | Matrix of vertices representing the tropical convex hull of of a polytope, P. |
| x0 | Starting point in tropical projective space. |
| I | Number of iterations for the hit-and-run algorithm |
| k | Size of subset. Defaults to one less than the # of columns of D_s. |
| M | Center of mass point in the tropical polytope. |
| S | Standard deviation to be used to sample from a tropical line segment. |

### Value

Point sampled from a tropical polytope according to a Gaussian distribution .

### Author(s)

David Barnhill

### Examples

```
library(rgl)
P1<-matrix(c(0,0,0,0,0,10,0,0,0,0,10,0,0,0,0,10),4,4,TRUE)
x0<-c(0,0,0,0)
mu<-c(0,5,5,5)
s<-1
har_norms<-matrix(0,15000,4,TRUE)
for (i in 1:1000){
  print(i)
  x<-TropicalPolytope.extrapolation.HAR_v4_NORM(P1,x0,I=50,k=3,mu,s)
  har_norms[i,]<-x
  x0<-x
  }
```

---

## trop_bal.vert *Calculate the tropical convex hull of a tropical ball.*

---

### Description

Calculate the tropical convex hull of a tropical ball.

### Usage

```
trop_bal.vert(x, d, dm)
```

### Arguments

| | |
|---|---|
| x | Center point for a tropical ball. |
| d | Radius of tropical ball. |
| dm | Dimension of tropical projective space. |

## Value

Square matrix of vertices defining convex hull of a tropical ball of radius d and center x.

## References

Yoshida, R., Miura, K., & Barnhill, D. (2022). *Hit and Run Sampling from Tropically Convex Sets.*

## Examples

```
x<-c(0,0,0)
d<-3
dm<-length(x)
V=trop_bal.vert(x,d,dm)
```

---

Trop_ball *Render a tropical ball in 2D or 3D.*

---

## Description

Render a tropical ball in 2D or 3D.

## Usage

```
Trop_ball(v, d, a = 1, cls = "black", fil = TRUE, plt = TRUE, bord = "black")
```

## Arguments

| | |
|---|---|
| v | Tropical point representing center of tropical ball. |
| d | Radius of tropical ball. |
| a | Opacity of 3D tropical ball. Unused for 2D rendering. |
| cls | Color of tropical ball. |
| fil | Logical indicating whether to fill the interior of 3D tropical ball or not. |
| plt | Logical indicating whether to plot the rendering with coordinates or not. |
| bord | Border color of tropical ball. |

## Value

Rendering of 2D or 3D tropical ball.

## Note

Point of interest does not need to be normalized.

## Author(s)

David Barnhill

### Examples

```
2D example
v<-c(0,1,0)
d<-1.5
cls<-'white'
Trop_ball(v, d=d, cls=cls)
3D example
v<-c(0,1,3,1)
a<-.1
d<-1.5
cls<-'blue'
Trop_ball(v, a=a,d=d,cls=cls)
```

---

## Trop_Volume *Estimate the volume of a tropical polytope.*

---

### Description

Estimate the volume of a tropical polytope.

### Usage

```
Trop_Volume(B, P, x0, S, i, d = ncol(P), R)
```

### Arguments

| | |
|---|---|
| B | Matrix of points defining the encompassing ball for a polytope, P. Points are row vectors. |
| P | Matrix of points defining the polytope P. Points are row vectors. |
| x0 | Starting point for use in HAR sampler. |
| S | Sample size to be taken. |
| i | Iterations used for HAR sampler. |
| d | Dimension of polytope for use to define subset size in HAR sampler. |
| R | Radius of tropical ball, B. |

### Details

This matters.

### Value

Returns a list with components

| | |
|---|---|
| r | Ratio of points falling in P to all sampled points in B. |
| VolB | Volume of minimum encompassing ball. |
| VolP | Volume estimate of tropical polytope, P. |
| har_points | Points sampled from B. |
| har_points1 | Sampled points that fall inside P. |

## References

Yoshida, R., Miura, K., & Barnhill, D. (2023). *Maximum Inscribed and Minimum Enclosing Tropical Balls of Tropical Polytopes and Applications to Volume Estimation and Uniform Sampling*.

## Examples

```
P<-matrix(c(0,0,0,0,3,1,0,2,5),3,3,TRUE)
cr<-max_enc_ball(P)
r<-cr[[2]]
cent<-cr[[1]]
B<-trop_ball.vert(cent,r,length(cent))
S<-1000
i<-50
V=Trop_Volume(B,P,x0,S,i,R=r)
```

# Index