1 How to use EL-Twosample

Currently the program is available as a single R file. The program is currently in a work-in-progress state. There are plans, however, to finish the program and to add it to CRAN. For now to use it, simply load the file in to R, before using the functions described below.

```
source("el-twosample.R")
```

1.1 EL.means

Calculate the sample mean difference and construct confidence bands for it.

Usage

```
EL.means(X, Y, conf.level=0.95)
```

Parameters

`X,Y` the samples

`conf.level` the confidence level for the confidence interval.

Return value

`estim` the estimated sample mean difference

`conf.int` the confidence interval as a two-element vector.

Example

```
X <- rnorm(100, 1)
Y <- rnorm(100)
EL.means(X, Y)

# Compare confidence intervals with the t-test
t.test(X, Y)
```

1.2 EL.curve

Estimates various two-sample statistics and constructs their confidence bands.

Usage

```
EL.curve(X, Y, t, type="roc", conf.level=NULL, bw.fun=bw.nrd0,
         bw.fixed=NULL, sim.conf.level=NULL, sim.numboot=200)
```

Parameters

`X,Y` the samples

`t` a vector of parameter values at which to calculate the statistic

`type` the type of the statistic to calculate; possible values are

- **roc** constructs a ROC curve
- **pp** constructs a P-P plot
- **qq** constructs a Q-Q plot
- **fdiff** constructs CDF differences
- **qdiff** constructs quantile differences

`conf.level` the confidence level for the point-wise confidence bands; no confidence bands are constructed if this is `NULL`

`bw.fun` a bandwidth function taking a data sample and returning a bandwidth; see also `bw.fixed`

`bw.fixed` a vector with two values representing the values used as bandwidths for the respective samples `X` and `Y`; if this is not `NULL`, the value of `bw.fun` is ignored

`sim.conf.level` the confidence level for the simultaneous confidence bands; this function will considerably increase the running time of the program, since a critical value needs to be bootstrapped; the critical value calculated is significantly dependent upon the choice of the vector t; if this is left `NULL`, no simultaneous confidence bands will be calculated

`sim.numboot` the number of bootstrap samples for the simultaneous confidence interval

Return value

`estim` a vector of the estimated values of the statistic

`conf.int` a matrix containing the point-wise confidence bands, with column 1 containing the lower and column 2 the upper confidence band

`boot.crit.val` the bootstrapped critical value

`simult.conf.int` a matrix containing the simultaneous confidence bands, with column 1 containing the lower and column 2 the upper confidence band

Examples

```
# Showcases all types of statistic available
#
# This example may take several minutes to run,
# depending on the computer


p <- par(mfrow=c(2,3))
X <- rlnorm(100, 1)
Y <- rchisq(100, 2)

dx <- density(X)
dy <- density(Y)
plot(dx, xlab="", ylab="", xlim=c(min(dx$x, dy$x), max(dx$x, dy$x)),
     ylim=c(0, 1.1*max(dx$y, dy$y)), main="Densities")
lines(dy, lty="dashed")
legend("topright", c("X", "Y"), lty=c("solid", "dashed"))

tt <- seq(0.05, 0.95, length=50)
zz <- EL.curve(X, Y, tt, type="roc")
plot(tt, zz$estim, xlab="", ylab="", main="ROC curve", type='l')

tt <- seq(0.05, 0.95, length=50)
zz <- EL.curve(X, Y, tt, type="pp")
plot(tt, zz$estim, xlab="", ylab="", main="P-P plot", type='l')

tt <- seq(0.05, 0.95, length=50)
zz <- EL.curve(X, Y, tt, type="qdiff")
plot(tt, zz$estim, xlab="", ylab="", main="Quantile difference", type='l')

dd <- max(X) - min(X)
tt <- seq(min(X)+0.05*dd, max(X)-0.05*dd, length=50)
zz <- EL.curve(X, Y, tt, type="qq")
plot(tt, zz$estim, xlab="", ylab="", main="Q-Q plot", type='l')

dd <- min(max(X),max(Y)) - max(min(X),min(Y))
tt <- seq(min(max(X),max(Y))+0.05*dd,
          max(min(X),min(Y))-0.05*dd,
          length=50)
zz <- EL.curve(X, Y, tt, type="fdiff")
plot(tt, zz$estim, xlab="", ylab="", main="CDF difference", type='l')

par(p)

# Constructs confidence bands
#
```

```
# This example may take several minutes to run,
# depending on the computer


X <- rnorm(200, 1)
Y <- rnorm(200)

tt <- seq(0.05, 0.95, length=50)
zz <- EL.curve(X, Y, tt, type="roc", conf.level=0.95, sim.conf.level=0.95)
plot(tt, zz$estim, xlab="", ylab="", ylim=c(0,1), main="ROC curve", type='l')
lines(tt, zz$conf.int[1,], lty="dashed")
lines(tt, zz$conf.int[2,], lty="dashed")
lines(tt, zz$simult.conf.int[1,], lty="dotted")
lines(tt, zz$simult.conf.int[2,], lty="dotted")
abline(0,1)
```