

```

# file name = GDcorr---GDave(x,y) and GDslp for slope computation
# GDslp defined to obtain slope in SLR, see line slp1 to use it
# put x data in x1 and y data in x2
rguniq <- function(rky) {
n <- length(rky); n1 <- n - 1
ryr <- n +1 -rky
dy <- NULL; dyn <- NULL
for (i in 1:n1) {
dy <- c(dy,sum(rky[1:i]-i >0))
dyn <- c(dyn,sum(ryr[1:i]-i>0))}
mdyr <- max(dyn)
mdy <- max(dy)
corrg <- (mdyr - mdy)/(n%/%2)
corrg}
GDave <- function(x,y) {
n <- length(x)
xt <- x[order(y,x)] # x order by y with y ties ordered by x
rky <- 1:n
rky <- rky[order(xt,rky)] # ranks of y ordered by x
ccp <- rguniq(rky) # GD positive
# now for GD most negative
xrr <- n + 1 -rank(x) #reverse ranks on the x
xt <- x[order(y,xrr)] # x ordered by y with y ties ordered by rev(x)
rky <- order(xt,n:1) # ranks of y ordered by x with y ties ordered by rev(y)
ccn <- rguniq(rky)
GDcor <- (ccp+ccn)/2
GDcor }
x1 <- rnorm(20) #sample(10,20,T)
x2 <- x1*.90+sqrt(1-.9^2)*rnorm(20) #sample(10,20,T)
GDave(x1,x2)
GDslp <- function(b,x,y) { GDave(x,y-b*x) }
slp1 <- uniroot(GDslp, c(-10,10),x=x1,y=x2)$root #run this to get slope
int1 <- median(x2-slp1*x1)
c(int1,slp1)
plot(x1,x2) ;abline(int1,slp1)

```