

# Exploring new weights

Kristoffer Hellton

September 23, 2016

## Forslag til ny vektfunksjon

Basert på figurene som viser at konturene av fordelingen til  $\lambda_{x_0}$  er rette linjer som går gjennom origo, forslår jeg en ny vektfunksjon som måler *vinkelen* til fokus-kovariaten. All observasjoner langs linjen som følger denne vinkelen vektet opp.

Itillegg ble det klart at det er stor skille i tuning parameter langs  $x^T \beta_{OLS} = 0$  OLS null prediksjonen. Hvert fokus transformes til en vinkel fra  $x^T \beta_{OLS} = 0$  og kun på en øvre halvplanet, slik at vinkelen blir mellom 0 og  $\pi = 180$  deg.

Vekten for et tilfeldig  $x_0$  ser da slik ut:

```
beta0 <-rep(1,2)
p <- length(beta0)
sigma0 = 1
n <- 50

X <- scale(matrix(runif(p*n,min=-1,max=1),ncol=p),center = TRUE, scale = FALSE)
Y <- X%*%beta0 + scale(rnorm(n,mean=0,sd=sigma0),center = TRUE, scale = FALSE)

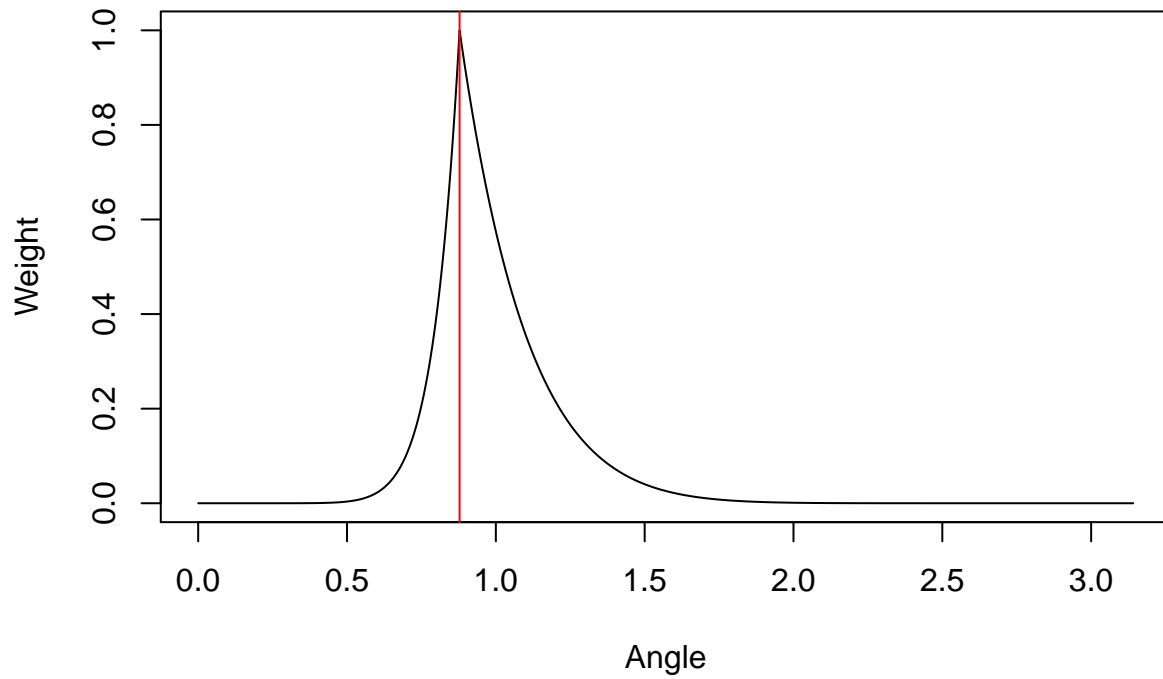
# Plot of observations (o) and focus (+), colored according to weight
x0 <- runif(p,min=-1,max=1)
betaOLS <- solve(crossprod(X,X))%*% t(X)%*%Y
x0.angle <- atan2(x0[2],x0[1]) - atan2(betaOLS[1],betaOLS[2])
if(x0.angle < (-pi)){x0.angle <- x0.angle+2*pi}
if(x0.angle > (pi)){x0.angle <- x0.angle-2*pi}
if(x0.angle < 0){x0.angle <- x0.angle + pi}
x0.angle

## [1] 0.8781662
```

```
weight <- function(x,x0.angle,alpha){
  x.angle <- atan2(x[2],x[1]) - atan2(betaOLS[1],betaOLS[2])
  if(x.angle < (-pi)){x.angle <- x.angle+2*pi}
  if(x.angle > (pi)){x.angle <- x.angle-2*pi}
  if(x.angle < 0){x.angle <- x.angle + pi}
  return((x.angle<x0.angle)*(x.angle/x0.angle)^alpha+
         (x.angle>=x0.angle)*((pi - x.angle)/(pi-x0.angle))^alpha)
}

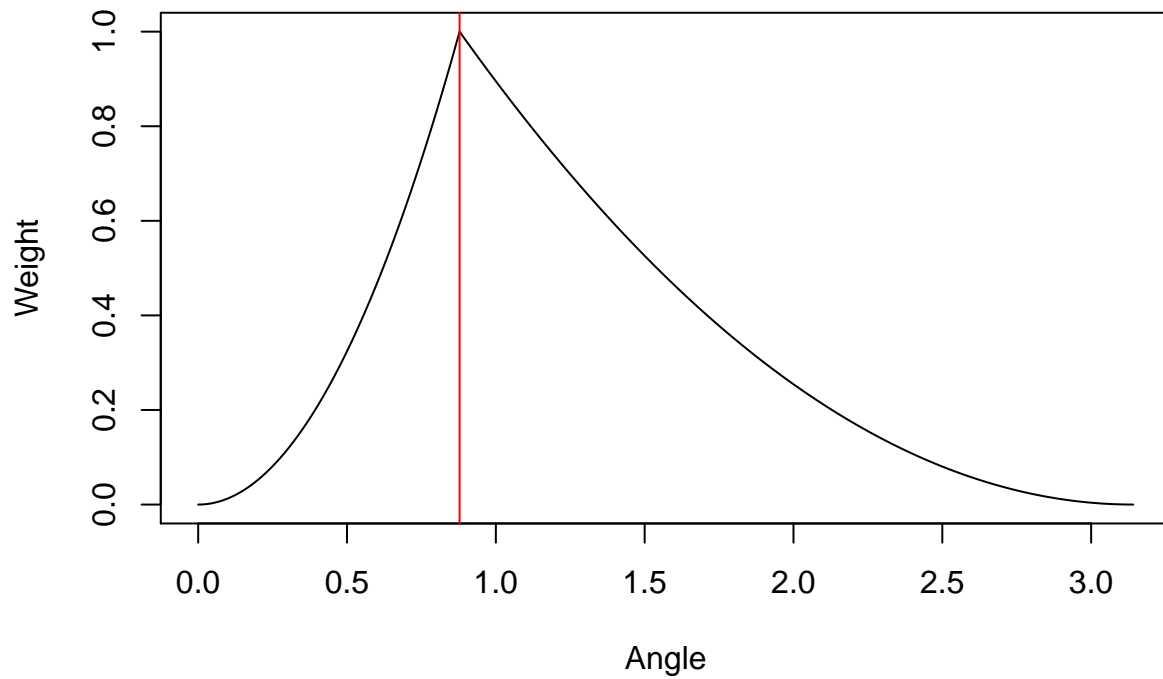
curve(sapply(x,function(x,alpha=10){(x<=x0.angle)*(x/x0.angle)^alpha+
  (x>x0.angle)*((pi - x)/(pi-x0.angle))^alpha}),0,pi,n=10^4,
  ylab='Weight',xlab='Angle',main='Weigth function with parameter 1')
abline(v=x0.angle,col=2)
```

### Weigth function with parameter 1

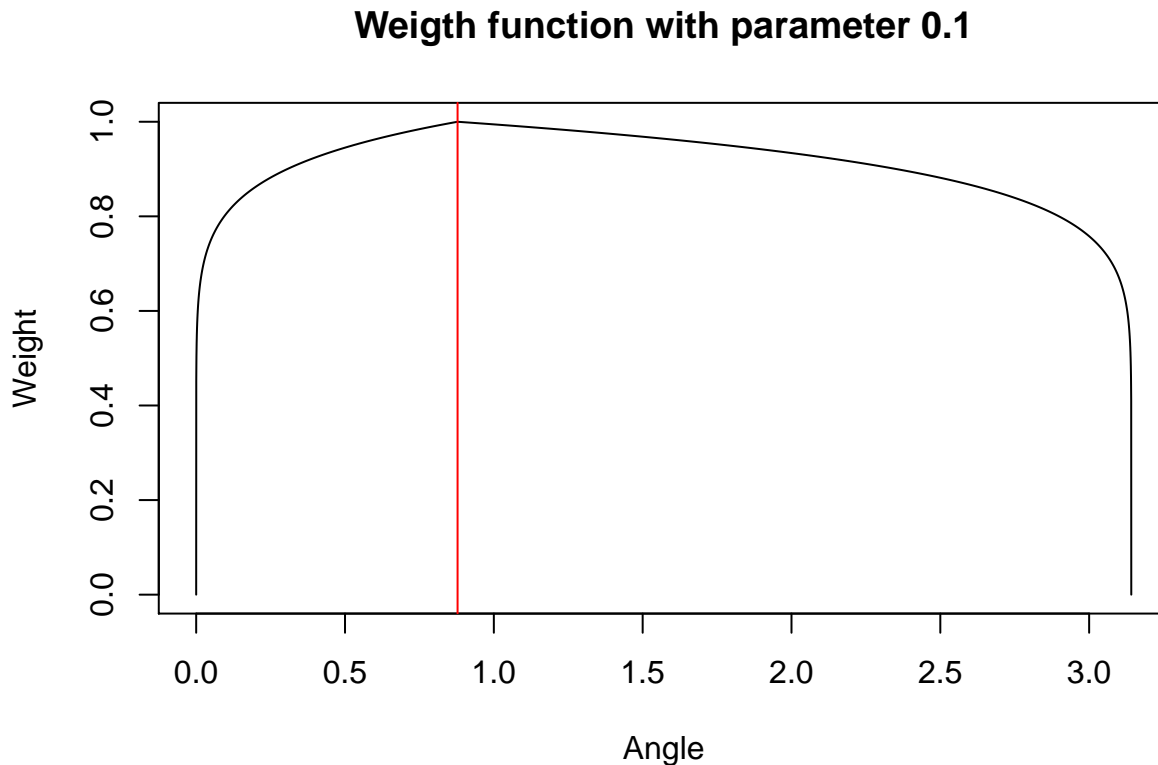


```
curve(sapply(x,function(x,alpha=2){(x<=x0.angle)*(x/x0.angle)^alpha+  
  (x>x0.angle)*((pi - x)/(pi-x0.angle))^alpha}),0,pi,n=10^4,  
  ylab='Weight',xlab='Angle',main='Weigth function with parameter 2')  
abline(v=x0.angle,col=2)
```

### Weigth function with parameter 2



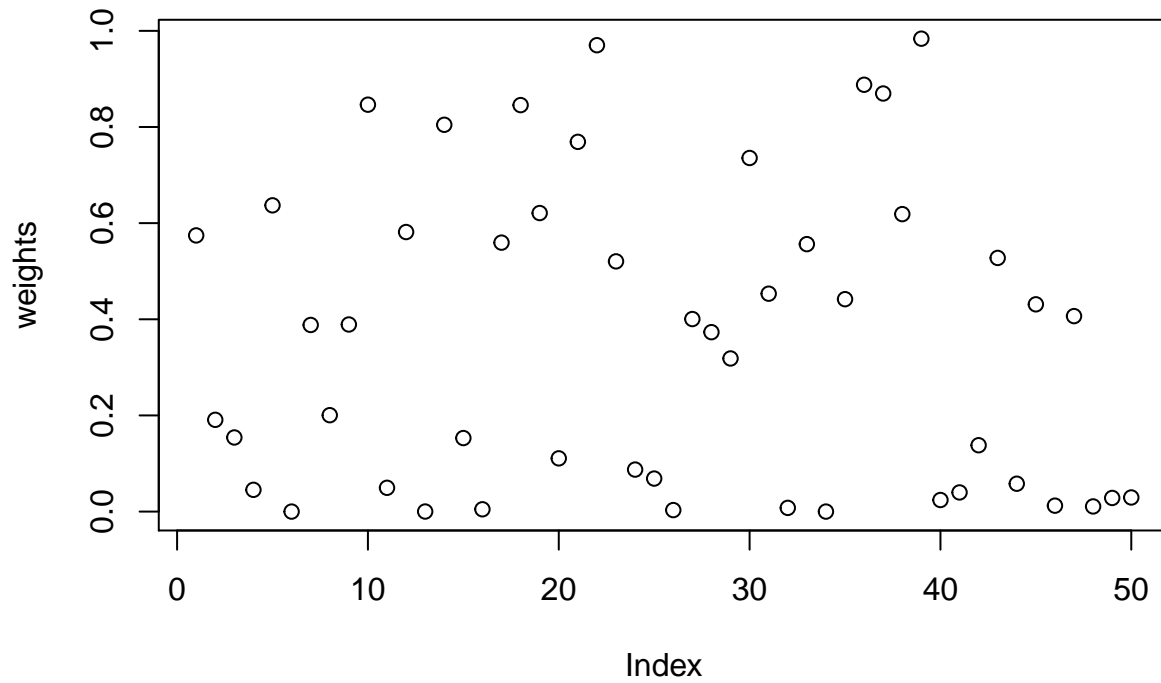
```
curve(sapply(x,function(x,alpha=0.1){(x<=x0.angle)*(x/x0.angle)^alpha+
  (x>x0.angle)*((pi - x)/(pi-x0.angle))^alpha}),0,pi,n=10^4,
  ylab='Weight',xlab='Angle',main='Weighth function with parameter 0.1')
abline(v=x0.angle,col=2)
```



## Hvordan ulike observasjoner blir vektet

Vi ser på hvordan ulike observasjon vektes forskjellig, først plassert etterhvert andre og så i et to dimensjonalt plot der observasjonen farges etter vektverdien (hvit er 1 og svart er 0), fokusverdien er avmerket med pluss og vinkelen gjennom fokuset er en rød linje.

```
weights <- apply(X,1,weight,x0.angle=x0.angle,alpha=2)
plot(weights)
```



```
valcol <- (weights - min(weights))/(max(weights) - min(weights))
plot(X,bg = gray(valcol),pch=21,ylim=c(-1.1,1.1),xlim=c(-1.1,1.1),type='p')
betaOLS <- solve(crossprod(X,X))%*% t(X)%*%Y
abline(a=0,b=betaOLS[1]/betaOLS[2])
abline(a=0,b=x0[2]/x0[1],col=2,lty=2)
abline(h=0,lty=2)
abline(v=0,lty=2)
points(x0[1],x0[2],pch=3)
```

