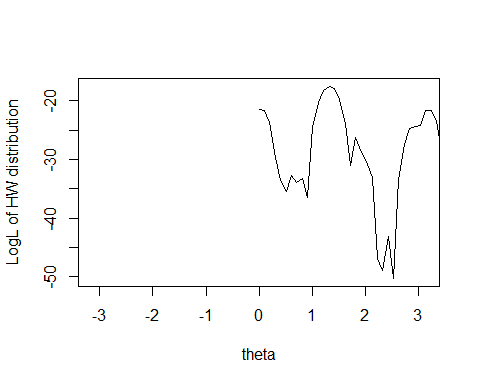
Prob6.R

Audrey McCombs

Fri Nov 10 22:03:13 2017

#Problem 6  
  
 #Problem 6.a  
x <-c(3.91, 4.85, 2.28, 4.06, 3.70, 4.04, 5.46, 3.53, 2.28, 1.96, 2.53, 3.88, 2.22, 3.47, 4.82, 2.46, 2.99, 2.54, 0.52, 2.50)  
theta <- seq(0,10,,100)  
  
hm <- function(theta = theta, x = 3.91) (1-cos(x - theta))/(2\*pi)  
loghm <- function(theta, x) (-log(2\*pi)+sum(log(1-cos(x-theta)^2)))  
plot(theta,sapply(theta,loghm,x),type="l",ylab="LogL of HW distribution", xlim = c(-3.1415927, 3.1415927))



plot(theta,sapply(theta,loghm,x),type="l",ylab="LogL of HW distribution", xlim = c(0, 10))  
  
 #Problem 6.b  
optimize(function(theta) sapply(theta, loghm, x), interval = c(1,2), maximum = T)

## $maximum  
## [1] 1.326029  
##   
## $objective  
## [1] -17.51277

#Problem 6.c  
ghm <- function(theta,x) sin(x-theta)/(1-cos(x-theta))  
derghm <- function(theta,x) 1/(1-cos(x-theta))  
  
loglik <- function(fun, derf, x0, eps, nlim,...) {  
 iter <- 0  
 repeat {  
 iter <- iter + 1  
 if(iter > nlim) {  
 cat(" Iteration Limit Exceeded: Current = ",iter, fill = T)  
 x1 <- NA  
 break  
 }  
 x1 <- x0 - fun(x0,...)/derf(x0,...)  
 if(abs(x0 - x1) < eps||abs(fun(x1,...))<1.0e-12)  
 break  
 x0 <- x1  
 cat("\n\*\*\*\*\*\* Iter. No: ", iter, " Current Iterate = ", x1, fill=T)  
 }  
 return(x1)  
}  
  
loglik(ghm, derghm, 0, 0.00001, 100, x)

##   
## \*\*\*\*\*\* Iter. No: 1 Current Iterate = 0.694991 0.9905465 -0.7588807   
## 0.7946357 0.5298361 0.7823359 0.7333152 0.3787149 -0.7588807 -0.9252115   
## -0.5741721 0.6731109 -0.7965655 0.3225359 0.9942155 -0.6300306 -0.1510127   
## -0.5659562 -0.4968801 -0.5984721  
##   
## \*\*\*\*\*\* Iter. No: 2 Current Iterate = 0.7683414 1.648321 -0.8614122   
## 0.9180916 0.5584035 0.8981469 1.733213 0.3884072 -0.8614122 -1.178793   
## -0.6115839 0.738361 -0.9212672 0.3284073 1.626262 -0.6815698 -0.1515927   
## -0.6015851 -1.347351 -0.6415793

## [1] 0.7684073 1.7083712 -0.8615927 0.9184073 0.5584073 0.8984073  
## [7] 2.2855744 0.3884073 -0.8615927 -1.1815926 -0.6115927 0.7384073  
## [13] -0.9215927 0.3284073 1.6783837 -0.6815927 -0.1515927 -0.6015927  
## [19] -2.3037001 -0.6415927

#This found the MLE value at -0.64: the local maximum near x = 0.  
  
 #Problem 6.d  
loglik(ghm, derghm, -2.0, 0.00001, 100, x)

##   
## \*\*\*\*\*\* Iter. No: 1 Current Iterate = -1.635417 -2.536948 -1.092033   
## -1.778663 -1.449314 -1.759205 -2.923388 -1.316034 -1.092033 -1.269942   
## -1.016587 -1.60765 -1.118794 -1.27352 -2.511401 -1.031681 -1.038287   
## -1.014822 -2.582331 -1.02247  
##   
## \*\*\*\*\*\* Iter. No: 2 Current Iterate = -0.9627782 -3.429855 -0.8636267   
## -1.348636 -0.5432575 -1.293898 -3.786495 -0.3249513 -0.8636267 -1.181708   
## -0.6225734 -0.8934112 -0.9228683 -0.2740048 -3.377936 -0.6887002   
## -0.2633001 -0.6132531 -2.621583 -0.6507349  
##   
## \*\*\*\*\*\* Iter. No: 3 Current Iterate = 0.02438699 -4.340533 -0.8615927   
## -0.5813819 0.3487038 -0.4808975 -3.963835 0.3294259 -0.8615927 -1.181593   
## -0.6115929 0.1047275 -0.9215927 0.2926269 -4.319364 -0.6815927 -0.1518248   
## -0.6015929 -2.621593 -0.6415928  
##   
## \*\*\*\*\*\* Iter. No: 4 Current Iterate = 0.7016384 -4.572642 -0.8615927   
## 0.4160982 0.5568737 0.5008239 -3.964778 0.3883732 -0.8615927 -1.181593   
## -0.6115927 0.6968417 -0.9215927 0.3283997 -4.600919 -0.6815927 -0.1515927   
## -0.6015927 -2.621593 -0.6415927  
##   
## \*\*\*\*\*\* Iter. No: 5 Current Iterate = 0.7683577 -4.574778 -0.8615927   
## 0.8975489 0.5584073 0.8880153 -3.964778 0.3884073 -0.8615927 -1.181593   
## -0.6115927 0.7383954 -0.9215927 0.3284073 -4.604778 -0.6815927 -0.1515927   
## -0.6015927 -2.621593 -0.6415927

## [1] 0.7684073 -4.5747780 -0.8615927 0.9184058 0.5584073 0.8984072  
## [7] -3.9647780 0.3884073 -0.8615927 -1.1815927 -0.6115927 0.7384073  
## [13] -0.9215927 0.3284073 -4.6047780 -0.6815927 -0.1515927 -0.6015927  
## [19] -2.6215927 -0.6415927

loglik(ghm, derghm, -2.7, 0.00001, 100, x)

##   
## \*\*\*\*\*\* Iter. No: 1 Current Iterate = -3.021028 -3.654152 -1.735595   
## -3.158951 -2.816549 -3.141092 -3.653541 -2.64684 -1.735595 -1.701372   
## -1.830996 -2.992476 -1.721474 -2.587056 -3.644745 -1.798516 -2.140995   
## -1.835988 -2.621673 -1.816545  
##   
## \*\*\*\*\*\* Iter. No: 2 Current Iterate = -3.624496 -4.450133 -0.9686912   
## -3.964005 -3.047801 -3.923115 -3.959777 -2.540695 -0.9686912 -1.204683   
## -0.892102 -3.548247 -1.004201 -2.362849 -4.463955 -0.8997605 -1.227339   
## -0.892039 -2.621593 -0.8938737  
##   
## \*\*\*\*\*\* Iter. No: 3 Current Iterate = -4.573892 -4.574455 -0.8617973   
## -4.949586 -3.49588 -4.917166 -3.964778 -2.329799 -0.8617973 -1.181595   
## -0.6152569 -4.458982 -0.9216866 -1.927581 -4.604313 -0.6833192 -0.3473938   
## -0.6056591 -2.621593 -0.6442603  
##   
## \*\*\*\*\*\* Iter. No: 4 Current Iterate = -5.381973 -4.574778 -0.8615927   
## -5.352952 -4.287034 -5.367922 -3.964778 -1.918949 -0.8615927 -1.181593   
## -0.6115927 -5.343657 -0.9215927 -1.153284 -4.604778 -0.6815927 -0.1528414   
## -0.6015927 -2.621593 -0.6415927  
##   
## \*\*\*\*\*\* Iter. No: 5 Current Iterate = -5.514388 -4.574778 -0.8615927   
## -5.364778 -5.278196 -5.384777 -3.964778 -1.178166 -0.8615927 -1.181593   
## -0.6115927 -5.543425 -0.9215927 -0.1572512 -4.604778 -0.6815927 -0.1515927   
## -0.6015927 -2.621593 -0.6415927  
##   
## \*\*\*\*\*\* Iter. No: 6 Current Iterate = -5.514778 -4.574778 -0.8615927   
## -5.364778 -5.710081 -5.384778 -3.964778 -0.1781749 -0.8615927 -1.181593   
## -0.6115927 -5.544778 -0.9215927 0.3095397 -4.604778 -0.6815927 -0.1515927   
## -0.6015927 -2.621593 -0.6415927

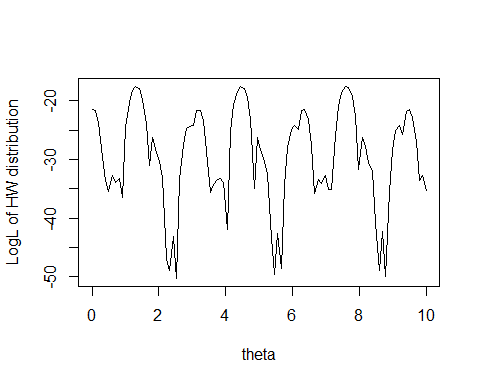
## [1] -5.5147780 -4.5747780 -0.8615927 -5.3647780 -5.7247774 -5.3847780  
## [7] -3.9647780 0.3585766 -0.8615927 -1.1815927 -0.6115927 -5.5447780  
## [13] -0.9215927 0.3284062 -4.6047780 -0.6815927 -0.1515927 -0.6015927  
## [19] -2.6215927 -0.6415927

# At a starting value of -2.0, the first iteration found an MLE of -1.02, which is a local maximum near -2.0. Later iterations found the same MLE as with starting point of 0 (i.e., -0.64). At a starting value of 2.7, however, the first iteration found the MLE at -1.8165, and it took more iterations to find the MLE of -0.64. This function bounces around a lot, so it's not surprising that the function finds local maxima and takes a while to settle.  
  
#Problem 7  
  
 #Problem 7.a  
men <- rnorm(n = 100, mean = 125, sd = 25)  
women <- rnorm(n = 100, mean = 125, sd = 15)  
t0 <- data.frame(M = men, W = women)  
head(t0)

## M W  
## 1 94.18540 142.2488  
## 2 98.84936 112.6996  
## 3 103.38450 121.5094  
## 4 185.52787 119.4877  
## 5 145.68981 119.1958  
## 6 93.34521 139.0001

#Problem 7.b  
permute <- function(t0, iter) {  
 t <- as.list(rep(NA, iter))  
 output <- as.list(rep(NA, iter))  
 ttemp <- t0  
 for (i in 1:iter) {  
 t[[i]] <- data.frame(M = sample(x = ttemp$M, 100), W = ttemp$W)  
 output[[i]] <- apply(t[[i]], 1, mean)  
 ttemp <- data.frame(M = output[[i]], W = output[[i]])  
 }  
 return(output)  
}  
  
 #Problem 7.c  
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.2.5



library(reshape2)

## Warning: package 'reshape2' was built under R version 3.2.5

heights <- permute(t0, 9)  
heights <- as.data.frame(heights)  
names(heights) <- paste("G", 1:9, sep = "")  
head(heights)

## G1 G2 G3 G4 G5 G6 G7 G8  
## 1 135.8820 136.1448 137.6520 129.1269 125.2955 125.6155 124.9099 124.7827  
## 2 116.8171 115.3775 112.4821 118.9764 118.8376 123.0381 124.2781 124.9299  
## 3 118.8127 127.7004 129.5418 126.7541 127.3588 124.2797 124.1995 124.2551  
## 4 112.0907 122.3789 119.9758 120.4831 128.1408 128.5821 127.7421 127.9181  
## 5 123.5298 125.8862 121.2932 123.3369 122.2617 121.9554 123.9541 124.9214  
## 6 121.1923 121.1923 120.8602 126.5974 125.4531 126.4668 126.6820 126.0116  
## G9  
## 1 126.1103  
## 2 125.1944  
## 3 125.1173  
## 4 125.3302  
## 5 126.4198  
## 6 125.7347

heights <- melt(heights)

## No id variables; using all as measure variables

names(heights) <- c("gen", "height")  
head(heights)

## gen height  
## 1 G1 135.8820  
## 2 G1 116.8171  
## 3 G1 118.8127  
## 4 G1 112.0907  
## 5 G1 123.5298  
## 6 G1 121.1923

ggplot(heights, aes(x=height)) + geom\_histogram(binwidth = 1) + facet\_wrap(~ gen)

