HYD298 Homework #5

Andrew Brown

February 3, 2016

Q1: Gumbel

```
library(fitdistrplus)
foo=read.csv("..\\HW1\\sebou.csv")

xx=sort(foo$maxQ,decreasing=FALSE)

plot(density(xx),xlab="Discharge, cfs",ylab="Probability density",
     main="Empirical and modeled (Gumbel)\nprobability densities of Sebou R.\nmaximum annual flows",lty=

##### Method of Moments
alpha=sqrt(6*var(xx)/pi^2)
xii=mean(xx)-0.5772*alpha
gum_pdf=(1/alpha)*exp(-((xx-xii)/alpha)-exp(-(xx-xii)/alpha))
lines(xx,gum_pdf,col="BLUE",lty=2,lwd=2)
gum_qua=xii-alpha*log(-log(c(0.01,0.99)))

#PARAMETERS
xii
```

```
## [1] 585.5702
```

```
alpha
```

```
## [1] 725.7755
```

```
#QUANTILES
gum_qua
```

```
## [1] -522.8195 3924.2460
```

```
##### L-Moments
pwm_gum = function(x,r) {
  #calculates probability weightetd moments of order r
  # used for calculating betas and associated L-moments
  cs=0
  n=length(x)
  for(i in seq(1,n-r)) {
    cs=cs+(choose(n-i,r)*x[i])/choose(n-1,r)
  }
  return(cs/n)
}

lambda1=mean(xx)
```

```r
beta1=pwm_gum(sort(xx,decreasing=TRUE),1)
lambda2=2*beta1-lambda1
alpha=lambda2/log(2)
xii=mean(xx)-0.5772*alpha
gum_pdf=(1/alpha)*exp(-((xx-xii)/alpha)-exp(-(xx-xii)/alpha))
gum_qua=xii-alpha*log(-log(c(0.01,0.99)))
lines(xx,gum_pdf,col="RED",lty=2,lwd=2)

#PARAMETERS
xii
```

```
## [1] 656.7242
```

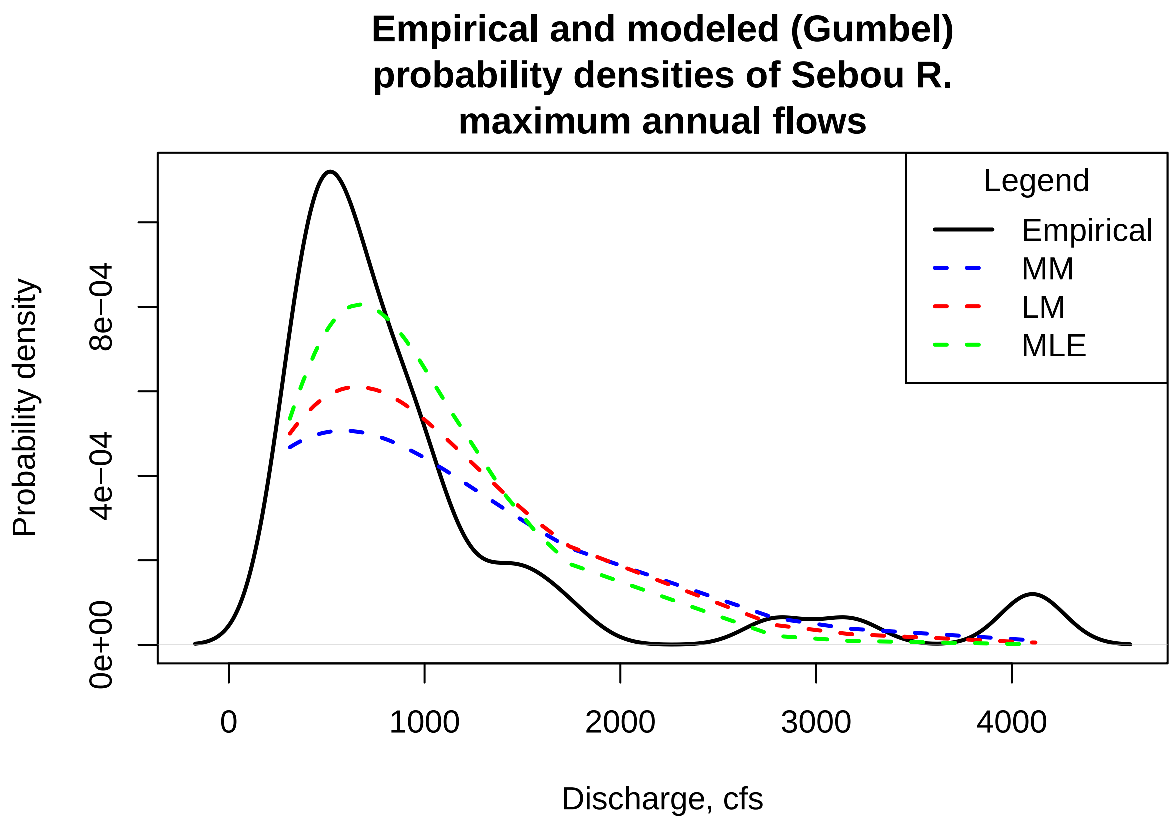
```r
alpha
```

```
## [1] 602.5011
```

```r
#QUANTILES
gum_qua
```

```
## [1] -263.4033 3428.3192
```

```r
##### Maximum Likelihood
dgumbel <- function(x, ps, al) {  exp((ps - x)/al - exp((ps - x)/al))/al }
pgumbel <- function(q, ps, al) {  exp(-exp(-((q - ps)/al))) }
qgumbel <- function(p, ps, al) {  ps-al*log(-log(p)) }
gumbel.fit <- fitdist(xx, "gumbel", start=list(ps=mean(xx), al=sd(xx)), method="mle")
xii=gumbel.fit$estimate['ps'][[1]]
alpha=gumbel.fit$estimate['al'][[1]]
gum_pdf=(1/alpha)*exp(-((xx-xii)/alpha)-exp(-(xx-xii)/alpha))
lines(xx,gum_pdf,col="GREEN",lty=2,lwd=2)
legend("topright", c("Empirical","MM","LM","MLE"), col=c('BLACK',"BLUE","RED","GREEN"), lty=c(1,2,2,2),
```




```r
#PARAMETERS:
xii
```

```
## [1] 671.0222
```

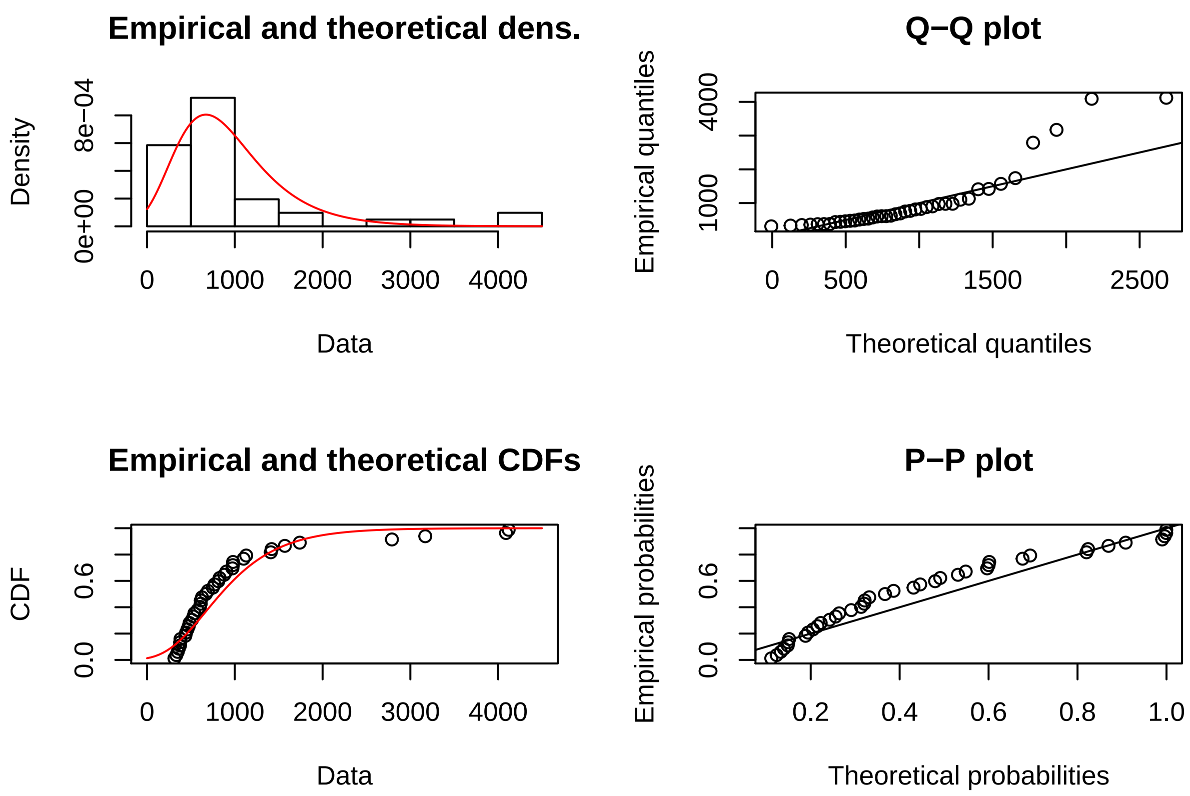
```r
alpha
```

```
## [1] 456.7768
```

```r
#QUANTILES:
quantile(gumbel.fit,probs=c(0.01,0.99))
```

```
## Estimated quantiles for each specified probability (non-censored data)
##             p=0.01   p=0.99
## estimate -26.55802 2772.264
```

```r
plot(gumbel.fit)
```



Q2: GEV

```
#gev density
dgev=function (x, xi = 1, mu = 0, sigma = 1) {
  tmp <- (1 + (xi * (x - mu))/sigma)
  (as.numeric(tmp > 0) * (tmp^(-1/xi - 1) * exp(-tmp^(-1/xi))))/sigma
}
#gev p fn
pgev=function (q, xi = 1, mu = 0, sigma = 1) {   exp(-(1 + (xi * (q - mu))/sigma)^(-1/xi)) }
#gev quantile fn
qgev=function (p, xi = 1, mu = 0, sigma = 1) {  mu + (sigma/xi) * ((-logb(p))^(-xi) - 1) }

plot(density(xx),xlab="Discharge, cfs",ylab="Probability density",
     main="Empirical and modeled (GEV)\nprobability densities of Sebou R.\nmaximum annual flows",lty=1,

##### L-Moments
pwm_gev=function(x,r) {
  rr=seq(r+1,length(x))
  sum(choose(rr-1,r)*x[rr]/choose(length(x),r+1))/(r+1)
}
lambda1=pwm_gev(xx,0) #=mean(xx)
beta1=pwm_gev(xx,1)
beta2=pwm_gev(xx,2)
lambda2=2*beta1-lambda1
```

```r
lambda3=6*beta2-6*beta1+lambda1
tau3=lambda3/lambda2
cc=2/(tau3+3)-log(2)/log(3)
#calculate kappa first
cc=lambda2/(3*beta2-lambda1)-(log(2)/log(3))
kappa=7.8590*cc+2.9554*(cc^2)
alpha=kappa*lambda2/(gamma(1+kappa)*(1-2^-kappa))
xii=lambda1+(alpha/kappa)*(gamma(1+kappa)-1)
gev_cdf=exp(-(1-(kappa*(xx-xii)/alpha))^(1/kappa))
gev_qua=xii+(alpha/kappa)*(1-(-log(c(0.01,0.99)))^kappa)
lines(xx,dgev(xx,xi=kappa,mu=xii,sigma=alpha),col="BLUE",lty=2,lwd=3)
 #Theoretical Upper bound
xii-(alpha/kappa) #note that density function is undefined above this value
```

```
## [1] 1214.454
```

```r
#PARAMETERS
xii
```

```
## [1] 567.5917
```

```r
alpha
```

```
## [1] 304.1845
```

```r
kappa
```

```
## [1] -0.4702465
```

```r
#QUANTILES
gev_qua
```

```
## [1]  236.1741 5547.7584
```

```r
##### Maximum Likelihood
gev.fit <- fitdist(xx, "gev", start=list(mu=0, sigma=1, xi=1), method="mle")
xii=gev.fit$estimate['mu'][[1]]
alpha=gev.fit$estimate['sigma'][[1]]
kappa=gev.fit$estimate['xi'][[1]]
lines(xx,dgev(xx,xi=kappa,mu=xii,sigma=alpha),col="GREEN",lty=2,lwd=2)
legend("topright", c("Empirical","LM","MLE","GMLE"), col=c('BLACK','BLUE','GREEN','RED'), lty=c(1,2,2,2

#PARAMETERS:
xii
```

```
## [1] 543.7068
```

```r
alpha
```

```
## [1] 258.0146
```

```r
kappa
```

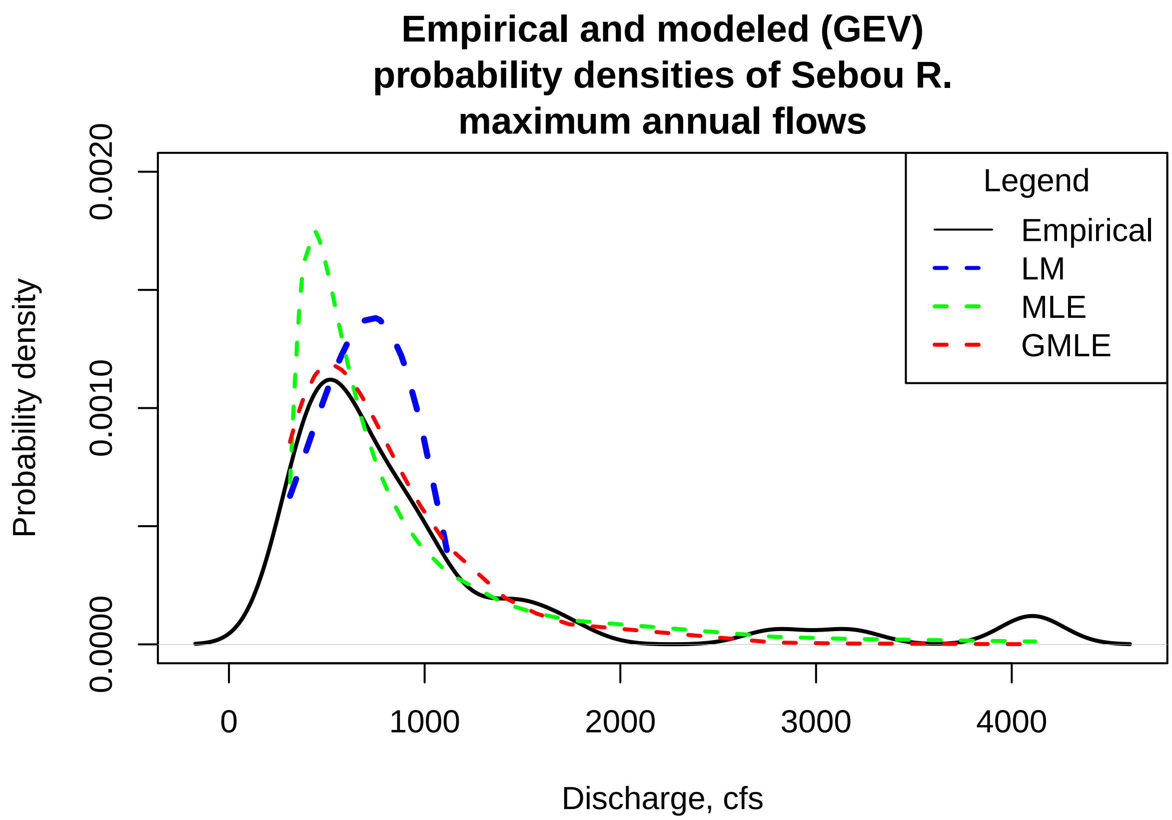
```
## [1] 0.7100908
```

```r
#QUANTILES:
quantile(gev.fit,probs=c(0.01,0.99))
```

```
## Estimated quantiles for each specified probability (non-censored data)
##            p=0.01   p=0.99
## estimate 303.1998 9707.516
```

```r
#plot(gev.fit)

##### GMLE
#A prior distribution that reflects general world-wide geophysical experience and physical realism is i
priork=function(kappa) {
  p=6
  q=9
  gamma(p)*gamma(q)*((0.5+kappa)^(p-1))*((0.5-kappa)^(q-1))/gamma(p+q)
}
#Likelihood function to be optimized
gmlf=function(p,x) {
  xii=p[1]
  alpha=p[2]
  kappa=p[3]
  yy=(1-(kappa/alpha)*(x-xii))
  -length(x)*log(alpha)+sum((1/kappa-1)*log(yy)-yy^(1/kappa))+log(priork(kappa))
}
minoo=optim(fn=gmlf,par=c(xii,alpha,-0.47),x=xx)
lines(xx,dgev(xx,mu=minoo$par[1],sigma=minoo$par[2],xi=minoo$par[3]),col="RED",lty=2,lwd=2)
```




```r
#QUANTILES
qgev(p=c(0.01,0.99),mu=minoo$par[1],sigma=minoo$par[2],xi=minoo$par[3])
```

```
## [1]   97.07712 2310.76745
```

```r
# xi
minoo$par[1]
```

```
## [1] 543.4494
```

```r
# alpha
minoo$par[2]
```

```
## [1] 312.2002
```

```r
# kappa
minoo$par[3]
```

```
## [1] 0.08729087
```

The GLME provides an excellent fit of the low-medium flows but the quantile corresponding to the 100yr flood is likely in error given the size of the dataset and the number of flows observed over that value. Here a prior distribution for κ from Loucks and van Beek (2006) is employed to demonstrate how prior information can be implemented to improve on MLE estimates. It is recognized that this prior distribution has little/nothing to do with the Sebou River dataset.

The starting points for the optimization are chosen based on the MLE parameters except in the case of κ. Optimizations using the MLE κ were unstable. A value of -0.47 (estimated from l-moments) was chosen as the starting point. The optimum value for κ was found to be stable from several starting points.