STAT 631 Homework 2

Jack Cunningham (jgavc@tamu.edu)

9/9/24

1)

We have function $f^\star(y|\xi)$ below:

$$f^\star(y|\xi) = \begin{cases} f(y\xi) & y < 0 \\ f(y/\xi) & y \geq 0 \end{cases}$$

We are tasked with integrating $f^\star(y|\xi)$:

$$\int_{-\infty}^{\infty} f^\star(y|\xi) = \int_{-\infty}^{0} f(y\xi)dy + \int_{0}^{\infty} f(y/\xi)dy$$

Let $u = y\xi$ so $du = \xi dy$. Let $w = y/\xi$ so $dw = \frac{1}{\xi}dy$. Then we have:

$$\xi \int_{-\infty}^{0} f(u)du + \xi^{-1} \int_{0}^{\infty} f(w)dw$$

$$\xi(F(0) - F(-\infty)) + \xi^{-1}(F(\infty) - F(0))$$

Since f is a symmetric pdf about 0 we have $F(0) = P(y < 0) = 1/2$. Also $F(-\infty) = 0$ and $F(\infty) = 1$ from basic rules about CDFs. Then we have our desired result:

$$\xi(1/2 - 0) + \xi^{-1}(1 - 1/2) = \frac{1}{2}(\xi + \xi^{-1})$$

2)

a)

The formula of Kurtosis for a random variable with a finite fourth moment is its fourth standardized moment.

$$Kur = \frac{E[(x-\mu)^4]}{\sigma^4}$$

We know that $Kur = 3$ for a normal distribution. This implies that $E[(x-\mu)^4] = 3\sigma^4$ for the normal distribution.

We have the below discrete density mixture:

$$f(x) = .95f_1(x) + .05f_2(x)$$

Where $f_1(x)$ is the density function of $N(0,1)$ and $f_2(x) = N(0,10)$.

A helpful fact for this problem is that $\mu^{(k)} = \sum_{i=1}^{n} p_i E_{f_i}[x_i^k]$.Essentially when we are computing a moment we can compute each distribution's moment, multiply them by their weight, and sum them together.

This leads to $f(x)$ having $\sigma^2 = p_1\sigma_1^2 + p_2\sigma_2^2 = .9(1) + .1(10) = 1.9$. And $E[(x-\mu)^4] = p_1E_{f_1}[(x-\mu_1)^4] + p_2Ef_2[(x-\mu_2)^4]$. So using what we know about the fourth central moment of the normal distribution we have:

$$E[(x-\mu)^4] = p_1(3\sigma_1^4) + p_2(3\sigma_2^4)$$

Then we can compute kurtosis with the below:

$$Kur = \frac{3(p_1\sigma_1^4 + p_2\sigma_2^4)}{(p_1\sigma_1^2 + p_2\sigma_2^2)^2}$$

```r
p_1 <- .9
sigma_1 <- 1
p_2 <- .1
sigma_2 <- sqrt(10)

Kur <- 3*(p_1*sigma_1^4+p_2*sigma_2^4)/(p_1*sigma_1^2+p_2*sigma_2^2)^2
Kur
```

```
[1] 9.058172
```

The kurtosis for this discrete mixture distribution is 9.058.

b)

Building off what I worked on in the previous question we have the below after substituting $\sigma_1 = 1, \sigma_2 = \sigma, p_1 = p, p_2 = (1 - p_1)$.

$$Kur(p, \sigma) = \frac{3(p + (1-p)\sigma^4)}{(p + (1-p)\sigma^2)^2}$$

3)

```
library(quantmod)
```

```
getSymbols("^GSPC",from = "2005-01-01", to = "2024-08-01")
```

```
[1] "GSPC"
```

```
x = weeklyReturn(Ad(GSPC),type = "log")*100
n = dim(x)[1]
```

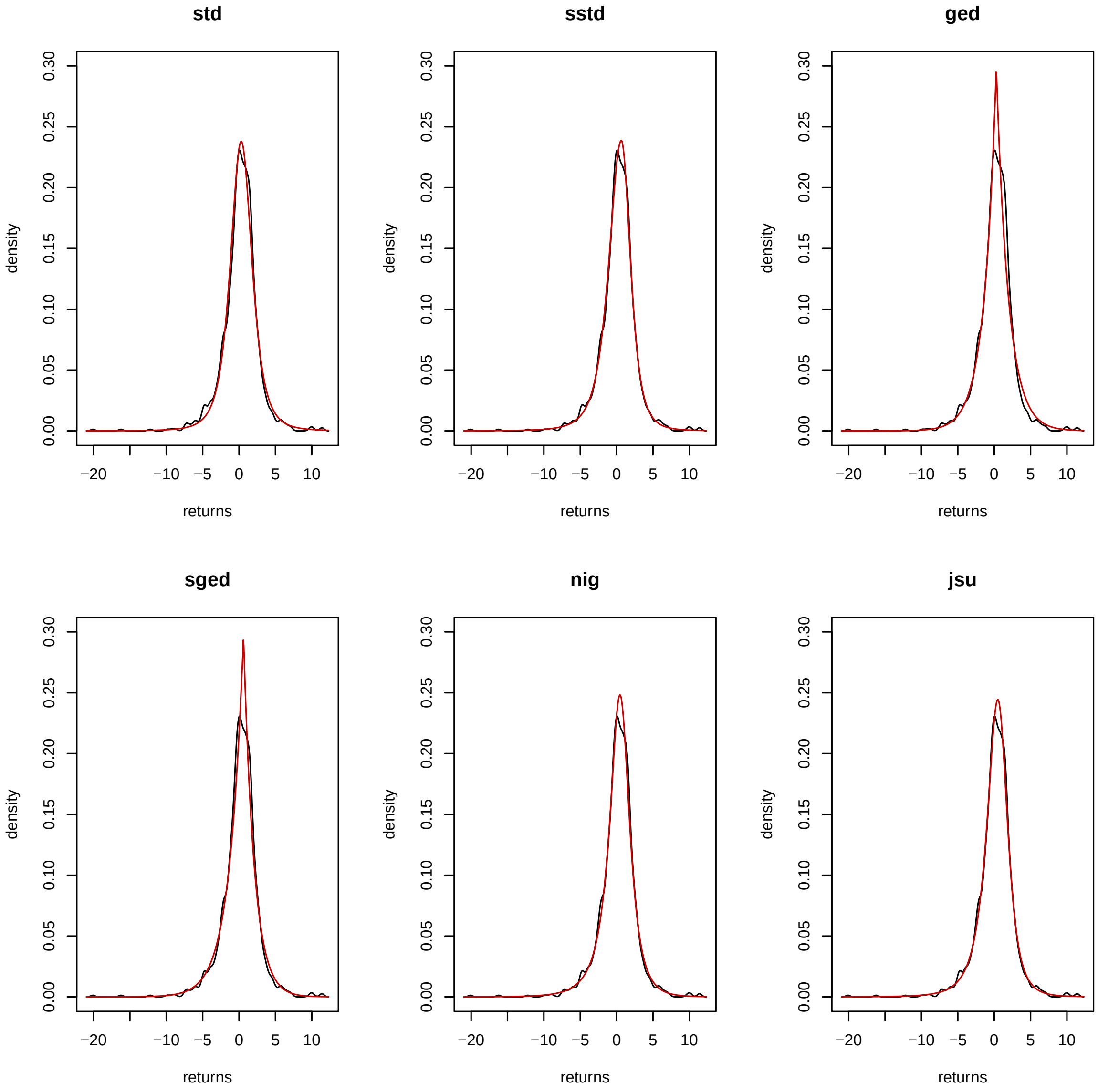
a)

```
library(rugarch)
```

```
dists = c("std", "sstd", "ged", "sged", "nig", "jsu")
fits = vector("list", 6)
for (i in 1:6) fits[[i]]  = fitdist(dists[i], x)
```

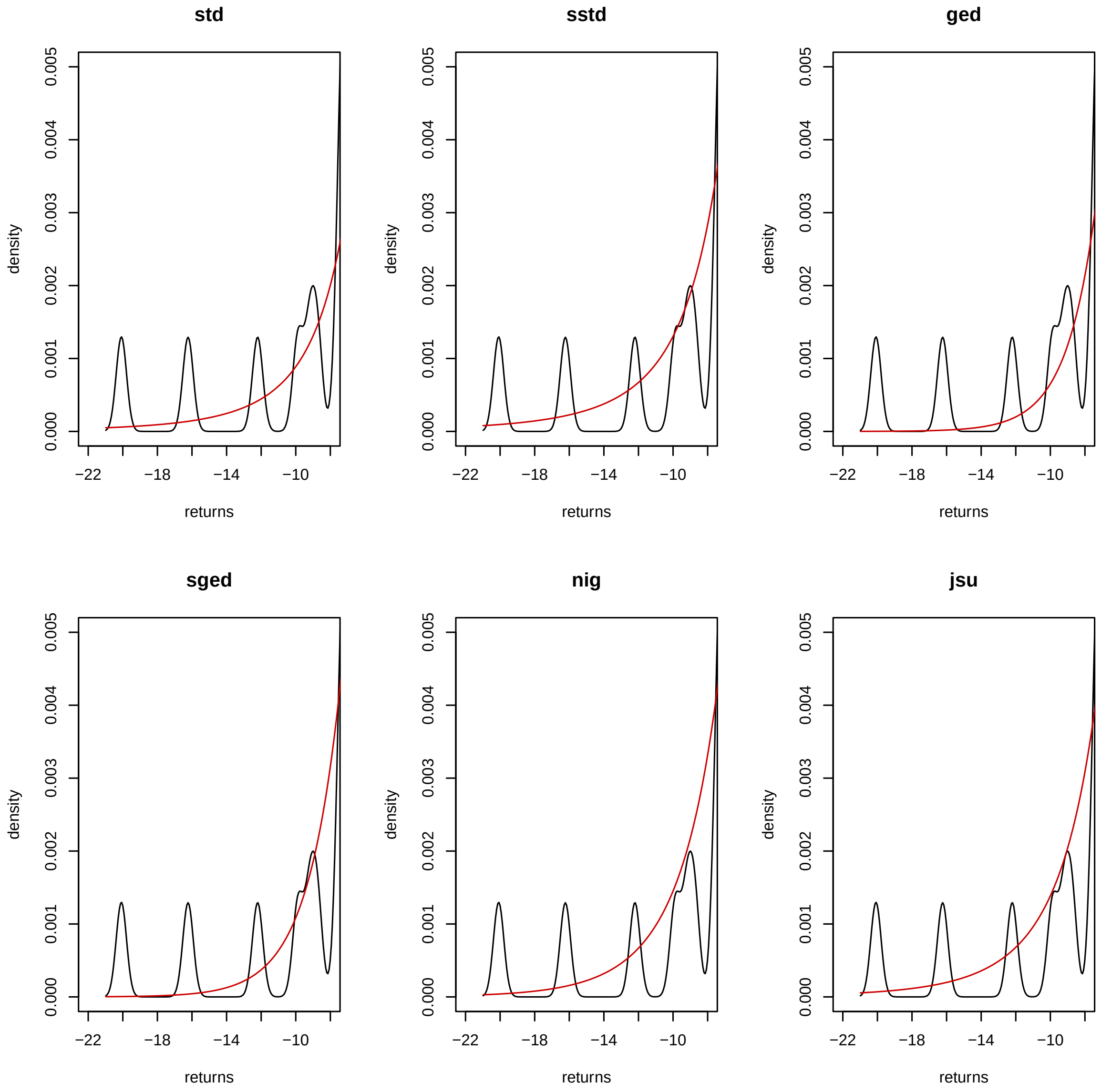
b)

```
den = density(x, adjust = 0.75)
x0 = den$x; y0 = den$y
par(mfrow = c(2,3))

for(i in 1:length(dists)) {
  plot(x0, y0, type = "l", main = dists[i], ylim = c(0,0.3), xlab = "returns",
  ylab= "density")
  est = fits[[i]]$pars
  yi = ddist(dists[i], x0, mu = est["mu"], sigma = est["sigma"], skew =
  est["skew"], shape = est["shape"])
lines(x0, yi, col = "red3")
}
```



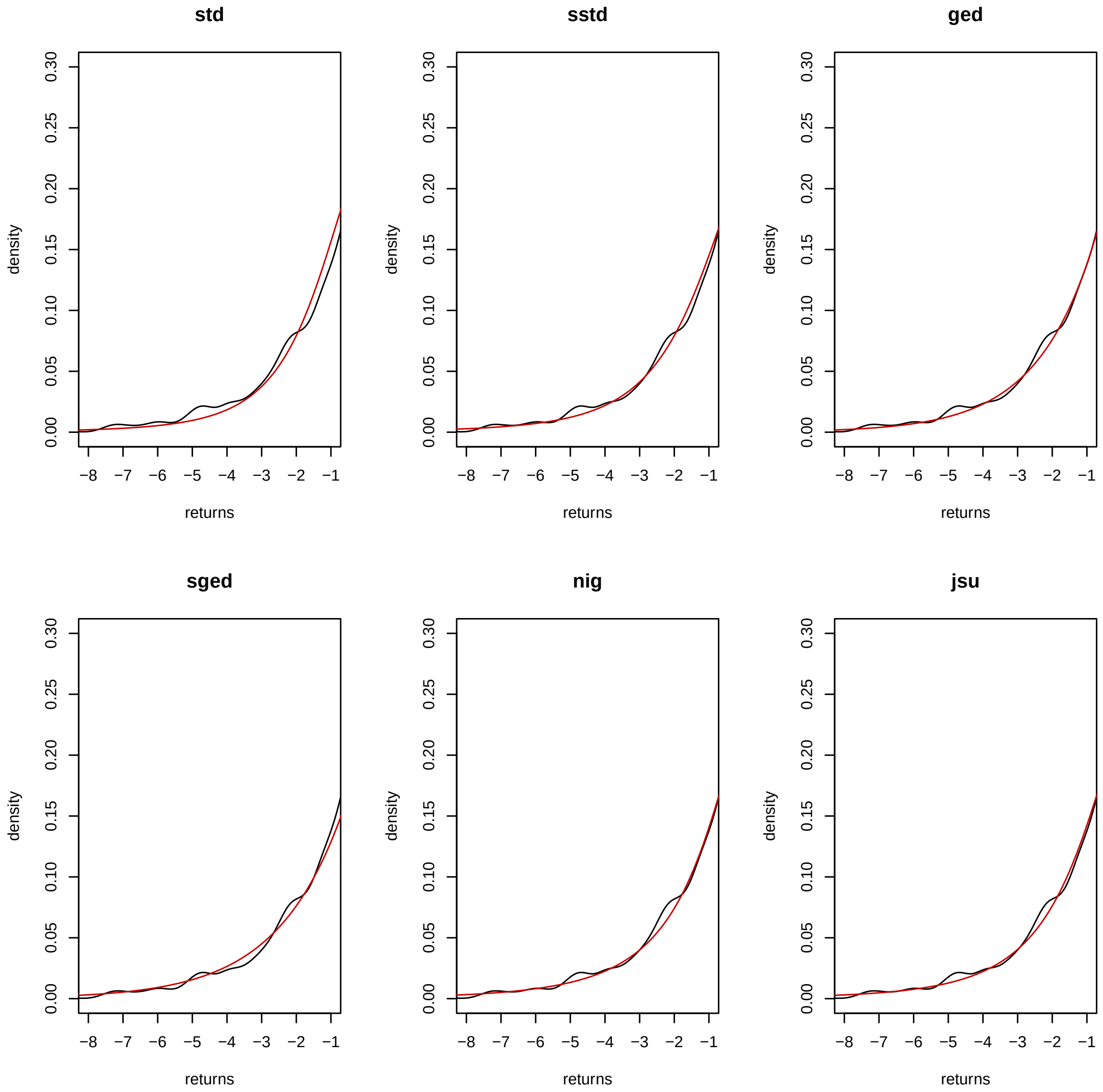
From this first plot we can see that the GED and skewed GED both seem to peak sharper than the kernel density estimate. The GED estimate also appears to struggle in the right shoulder region. The other four estimates fit well in the middle region. Next let's take a look at the left tail region.

```r
par(mfrow = c(2,3))

for(i in 1:length(dists)) {
  plot(x0, y0, type = "l", main = dists[i], ylim = c(0,0.005), xlim = c(-22,-8), xlab = "r
  ylab= "density")
  est = fits[[i]]$pars
  yi = ddist(dists[i], x0, mu = est["mu"], sigma = est["sigma"], skew =
  est["skew"], shape = est["shape"])
lines(x0, yi, col = "red3")
}
```



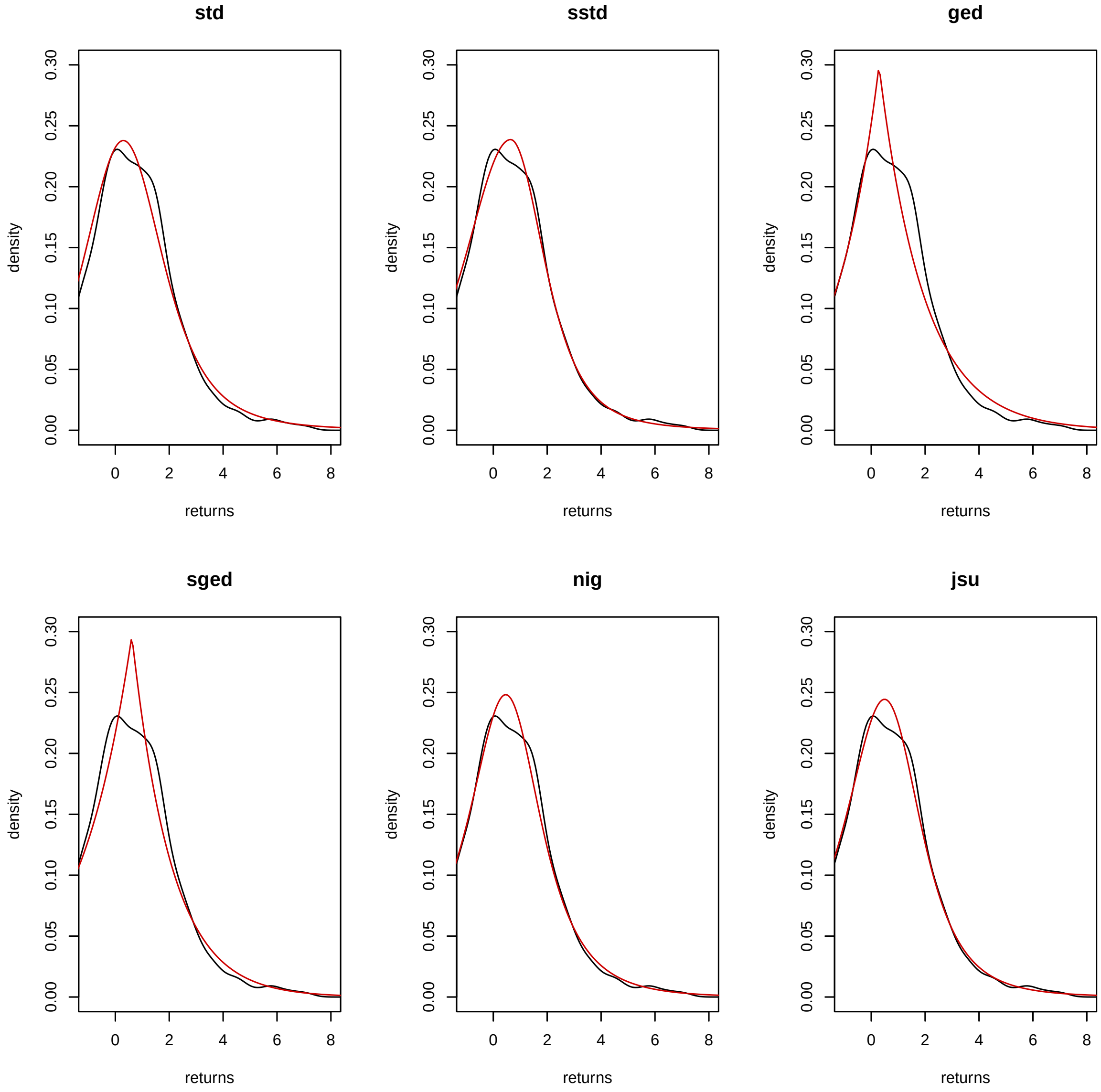
Looking at the extreme tail values we can get an idea of how quickly each distributions tail dissipates. We can see that the GED distribution appears to zero out around $x = 15$, too early compared to the kernel density estimator. The skewed GED distribution seems to taper a bit later around $x = 16$, still too early. The standardized t distribution does well here, it it intercepting around the middle of the kernel density estimator and tapers off completely at around $x = 19$. The other three distributions do well but seem to slightly overestimate the density in the $x = 8$ to $x = 11$ range.

```
par(mfrow = c(2,3))

for(i in 1:length(dists)) {
  plot(x0, y0, type = "l", main = dists[i], ylim = c(0,0.3), xlim = c(-8,-1), xlab = "retu
  ylab= "density")
  est = fits[[i]]$pars
  yi = ddist(dists[i], x0, mu = est["mu"], sigma = est["sigma"], skew =
  est["skew"], shape = est["shape"])
lines(x0, yi, col = "red3")
}
```



In the left shoulder region all densities fit the data reasonably.

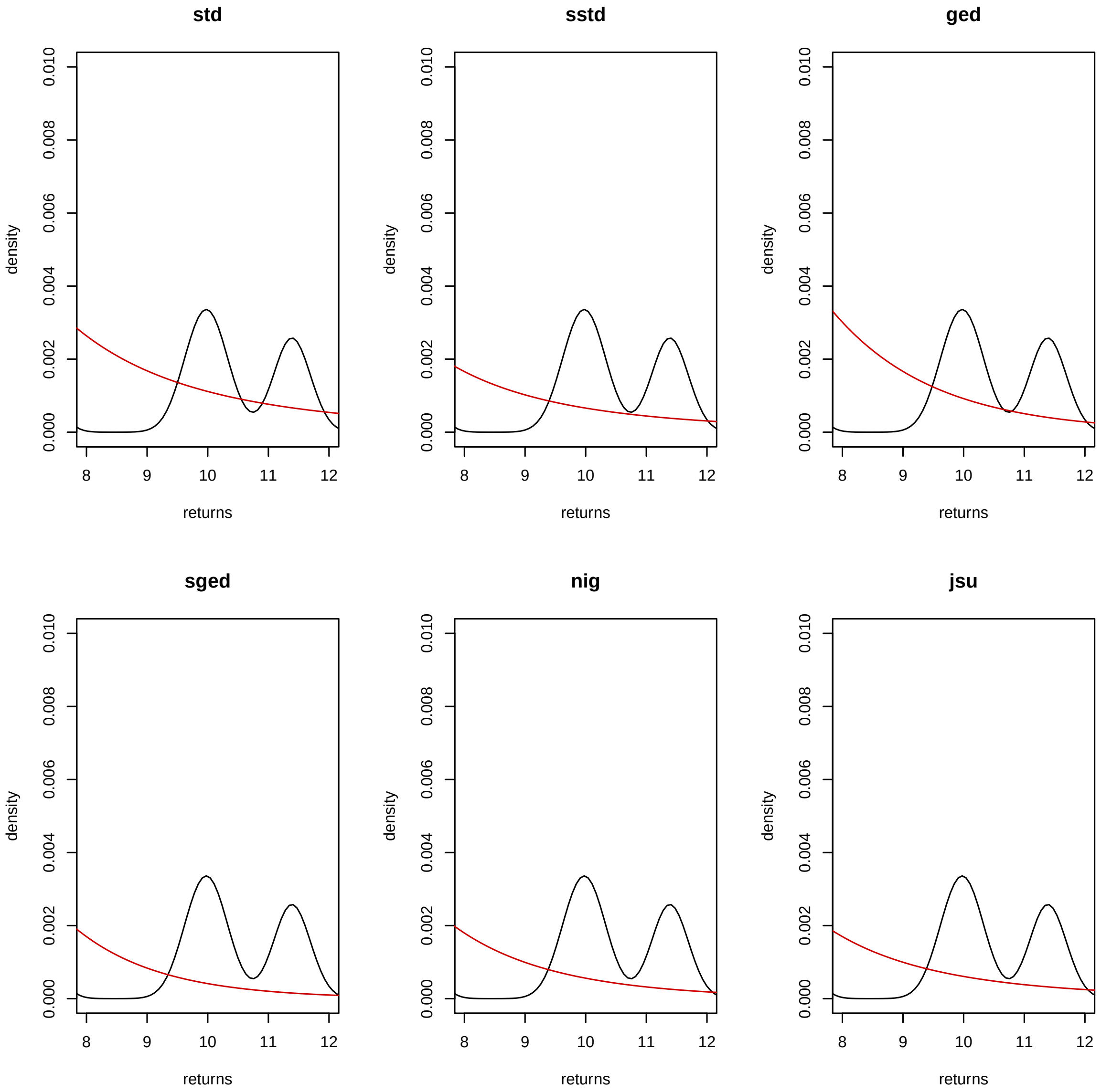
```
par(mfrow = c(2,3))

for(i in 1:length(dists)) {
  plot(x0, y0, type = "l", main = dists[i], ylim = c(0,0.3), xlim = c(-1,8), xlab = "retur
  ylab= "density")
  est = fits[[i]]$pars
  yi = ddist(dists[i], x0, mu = est["mu"], sigma = est["sigma"], skew =
  est["skew"], shape = est["shape"])
lines(x0, yi, col = "red3")
}
```



In this plot we can compare the peaks around $x = 1$. The GED and skewed GED models are unacceptable in the peak region. The standardized t and skewed t distributions perform similarly and follow the data closer than the other models in the peak region. The Normal Inverse Gaussian and Johnson S_U distributions overestimate the peak area a bit but not nearly as poorly as the GED based distributions.

In the shoulder region the skewed t distribution follows the data remarkably well. The Johnson S_U model tracks the data similarly well.

```
par(mfrow = c(2,3))

for(i in 1:length(dists)) {
  plot(x0, y0, type = "l", main = dists[i], ylim = c(0,0.01), xlim = c(8,12), xlab = "retu
  ylab= "density")
  est = fits[[i]]$pars
  yi = ddist(dists[i], x0, mu = est["mu"], sigma = est["sigma"], skew =
  est["skew"], shape = est["shape"])
lines(x0, yi, col = "red3")
}
```

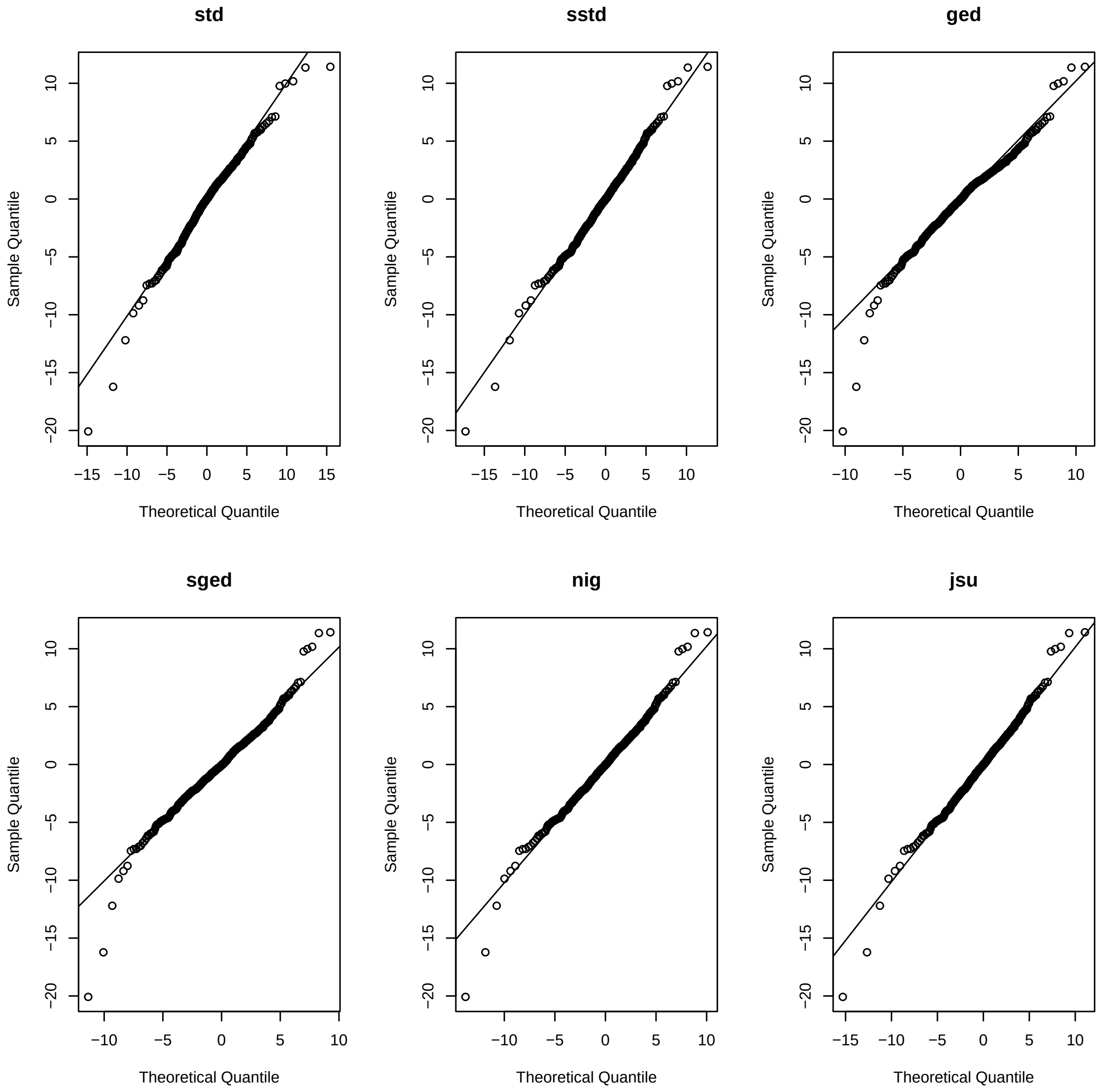


In the right tail region all the distributions perform similarly.

Overall after looking at these plots I'd say the best distribution is the skewed t distribution. It performs extremely well in the peaked and shoulder regions. In the left tail region is performs acceptably, tapering at a similar pace as the data.

c)

```
par(mfrow = c(2,3))
q_grid = (1:n) / (n + 1)
for(i in 1:length(dists)) {
  est = fits[[i]]$pars
  theoretical_quantile <- qdist(distribution = dists[i],p = q_grid, mu = est["mu"],
        sigma = est["sigma"], skew = est["skew"], shape = est["shape"])

  qqplot(x = theoretical_quantile, y = as.numeric(x), main = dists[i],
         xlab = "Theoretical Quantile",ylab= "Sample Quantile")
    abline(lm(quantile(x, c(0.25, 0.75)) ~
             qdist(distribution = dists[i], p = c(0.25, 0.75), mu = est["mu"],
                   sigma = est["sigma"], skew = est["skew"], shape = est["shape"])))
}
```



Immediately it appears that the differentiator between the good and poor fits is the performance in the left tail. The GED and skewed GED distributions underestimate the frequency outlier left tail values significantly, the left tail in the observed data is longer than what those distributions would suggest. The standardized t distribution does a bit better but still struggles in that region. The two distributions that do the best are the skewed t, Johnson S_U and Normal Inverse Gaussian . I would choose the skewed t distribution as the favored model, the three perform similarly but the skewed t distribution does slightly better length wise in the left tail. The extreme loss values of $x = -15, 20$ in particular are closer to the fitted line.

d)

The two information criterion that we will look at are the AIC (Akaike's information criterion) and BIC (Bayesian information criterion). Definitions below:

$$\text{AIC} = -2\log L(\hat{\theta}_{ML}) + 2p$$

$$\text{BIC} = -2\log L(\hat{\theta}_{ML}) + p\log(n)$$

```r
loglik = c();p = c()
for(i in 1:length(dists)){
  loglik[i] = -tail(fits[[i]]$values, 1)
  p[i] = length(fits[[i]]$pars)
}
names(loglik) = names(p) = dists
aic = -2*loglik + 2*p
bic = -2*loglik +2*log(n)

rbind(loglik,p,aic,bic)
```

```
             std      sstd       ged      sged       nig       jsu
loglik -2246.861 -2240.528 -2258.300 -2252.390 -2241.905 -2240.556
p          3.000     4.000     3.000     4.000     4.000     4.000
aic     4499.722  4489.057  4522.599  4512.779  4491.809  4489.113
bic     4507.581  4494.916  4530.458  4518.638  4497.668  4494.972
```

```r
cat("Model selected by AIC:", dists[which.min(aic)],
    "\nModel selected by BIC:", dists[which.min(bic)])
```

```
Model selected by AIC: sstd
Model selected by BIC: sstd
```

Both the AIC and BIC choose the skewed t distribution.

After analyzing the density and histogram plots I was between the skewed t distribution and the Johnson S_U distribution mainly. The skewed t distribution appeared to do a better job with extreme values in the left tail, which is arguably the most important aspect of the distribution to get correct in financial return data. It also tracked the observed return data very well in the peak and shoulder regions. With the added confidence that the AIC and BIC agrees on the skewed t distribution being the best fit, I would choose the skewed t distribution.

e)

The two skewed distributions are skewed-t and skewed-GED. The estimate of the skew parameters are below.

```
skew_param_sstd <- fits[[2]]$pars["skew"]
skew_param_sged <- fits[[4]]$pars["skew"]
names(skew_param_sstd) = dists[2]
names(skew_param_sged) = dists[4]

c(skew_param_sstd, skew_param_sged)
```

```
     sstd      sged 
0.8581450 0.8692741 
```

Next we need to get the standard errors for each skew parameter. This can be found from the hessian matrix, the skew parameter in each distribution is parameter 3.

```
skew_param_sstd_se <- sqrt(diag(solve(fits[[2]]$hessian)))[3]
skew_param_sged_se <- sqrt(diag(solve(fits[[4]]$hessian)))[3]
```

Then, using the fact that MLEs converge to a normal distribution with $\mu = \theta$ and $\sigma^2 = I^{-1}(\theta)$ as $n \to \infty$, we can compute the below confidence intervals.

```
sstd_conf_int_skew <- cbind(skew_param_sstd, skew_param_sstd-1*skew_param_sstd_se*qnorm(.9
sged_conf_int_skew <- cbind(skew_param_sged, skew_param_sged-1*skew_param_sged_se*qnorm(.9

skew_conf_ints <- rbind(sstd_conf_int_skew,sged_conf_int_skew)
colnames(skew_conf_ints) <- c("skew est","lower.95%","upper.95%")
rownames(skew_conf_ints) <- c("SSTD","SGED")
skew_conf_ints
```

```
      skew est lower.95% upper.95%
SSTD 0.8581450 0.7876853 0.9286048
SGED 0.8692741 0.8319127 0.9066356
```

f)

We are testing for distribution symmetry, if skew = 0 then a distribution is symmetric. Then the hypothesis test is:

$$H_0 : \text{skew} = 0, H_1 : \text{skew} \neq 0$$

We reject H_0 if:

$$2\{\log(\hat{\theta}_{ML}) - \log(\hat{\theta}_{skew=0,ML})\} \geq \chi_{.95,1}$$

We have already fitted the distribution of the reduced model with skew = 0, it is the standardized t distribution from earlier. So:

```
cat("log-likelihoods: \n");c(full = loglik[2], reduced=loglik[1])
```

```
log-likelihoods:


 full.sstd reduced.std
 -2240.528   -2246.861
```

```
LRT = as.numeric(2*(loglik[2]-loglik[1]))
critical_value = qchisq(.95,df = 1)
p_value = 1 - pchisq(LRT,df = 1)
c(LRT.statistic = round(LRT,5), Critical.Value = round(critical_value,5), P.Value = round(
```

```
LRT.statistic Critical.Value        P.Value
     12.66483        3.84146        0.00037
```

With the LRT statistic being greater than the critical value $\chi_{.95,1}$ we can reject the null hypothesis of no skewness in the distribution. The evidence suggests that the distribution is skewed.