

Diagnostic plots for multilevel residuals

Richard Wilkinson

5 March 2019

Motivated by a question posted on MOLE, lets examine more closely how to use diagnostic plots. We'll create some fake data from the model

$$y_{ij} = \mu + b_i + \epsilon_{ij}$$

where we control the distribution of b_i and ϵ_{ij} . We will begin by generating b_i and ϵ_{ij} from a normal distribution so that we can see what **good** diagnostic plots look like. We will then simulate values from a t-distribution, which is heavy tailed distribution compared to the Gaussian distribution. We will then be able to see what **bad** diagnostic plots look like.

Let's begin by creating some fake data from the model where everything is Gaussian.

```
set.seed(1)
library(ggplot2)
library(lme4)

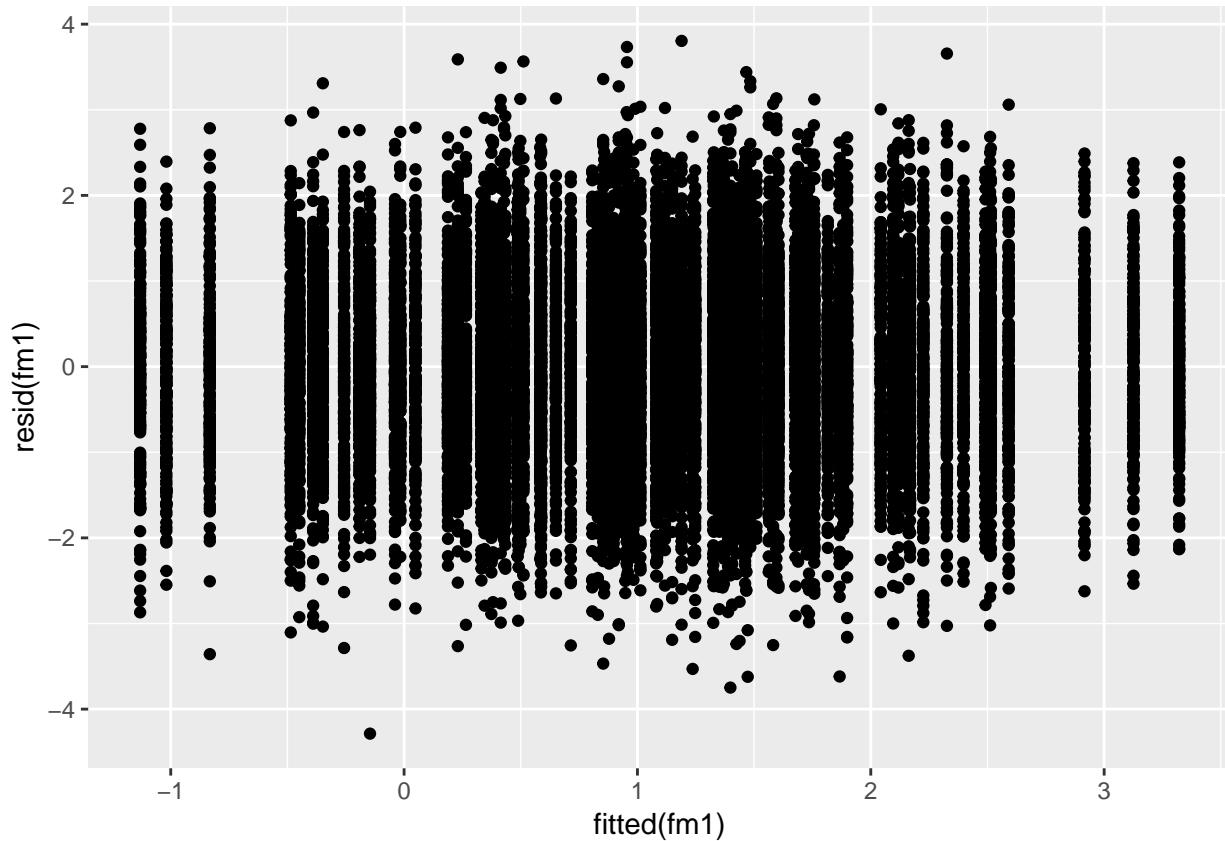
## Loading required package: Matrix
ngroup <- 100
groupsize <- 200
group <- gl(ngroup,groupsize)
b_i <- rnorm(ngroup) # random effects
epsilon <- rnorm(ngroup*groupsize)
y <- 1+b_i[group]+epsilon
data = data.frame(y, group = group)
```

Now lets fit the appropriate model, which is the true model, and look at the diagnostic plots.

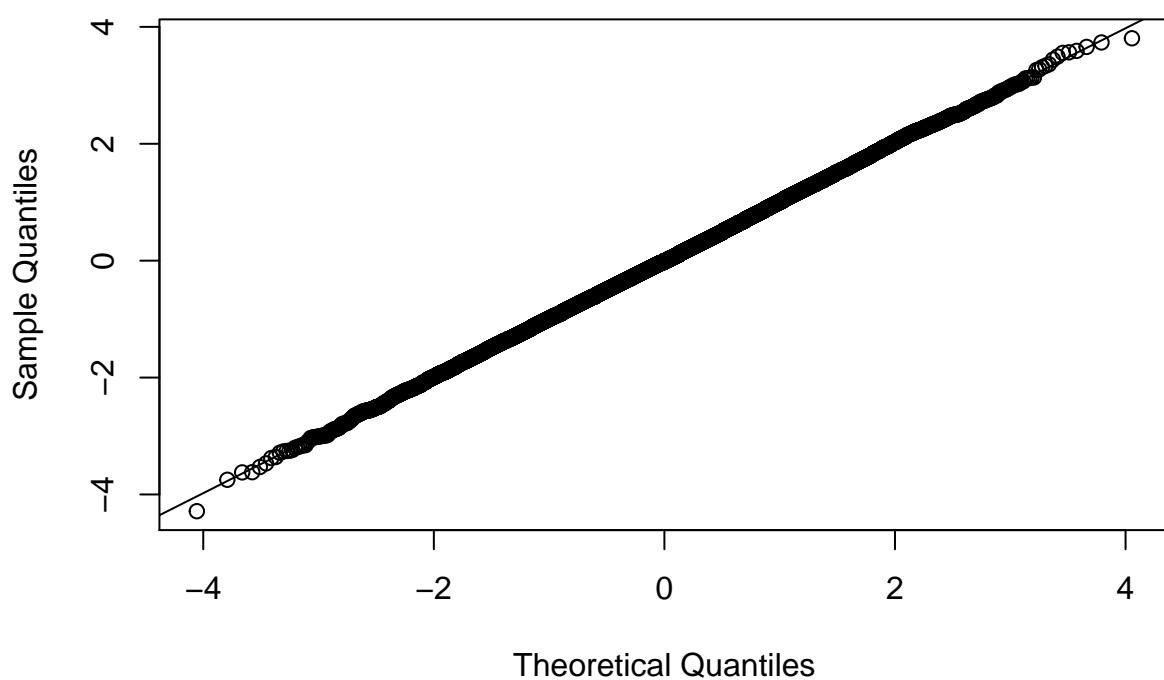
```
fm1<- lmer(y~1+(1|group), data = data)
```

Let's begin by looking at the level 1 residuals. We can plot the residuals against the fitted values. We hope to see no relationship. We can also look at the qq plot - where we hope to see a straight line.

```
qplot(fitted(fm1),resid(fm1))
```



QQ plot for level 1 residuals

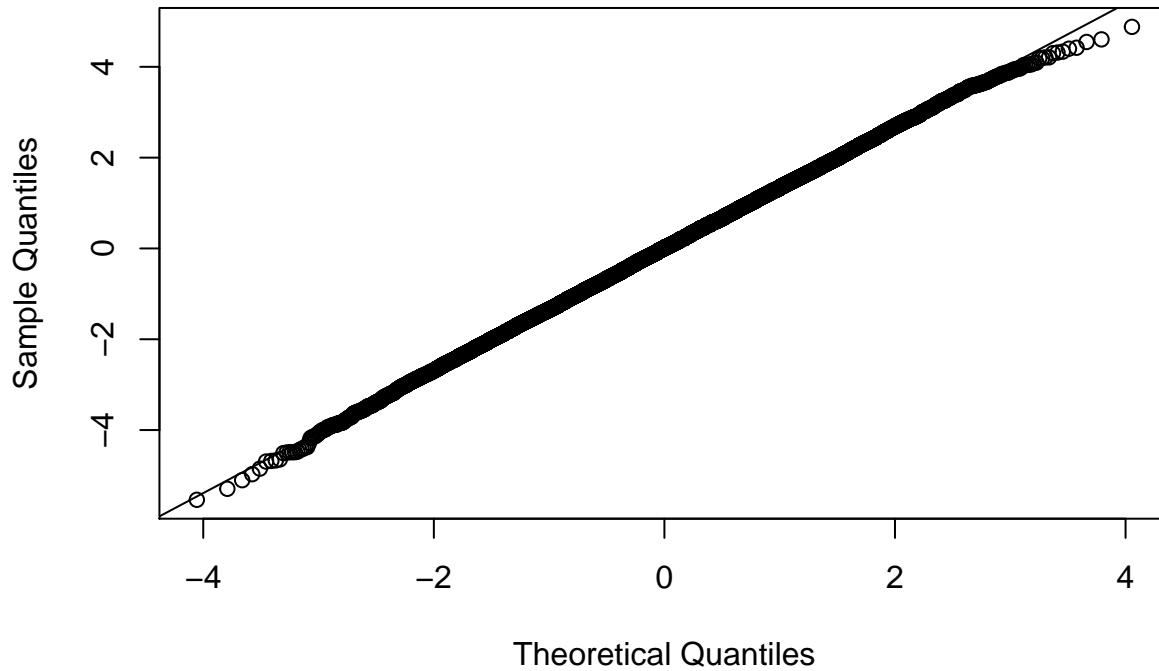


Both of these look fine, as they should as we know the model is true in this case. Now let's check the level 0 residuals. Note that there is no point looking at the residuals vs the fitted values now as all the fitted values are the same (this wouldn't be the case if we had a fixed effect that depended on a covariate).

```
fitted.level0<-fm1@pp$X %*% fixef(fm1)
resid.level0<-y-fitted.level0

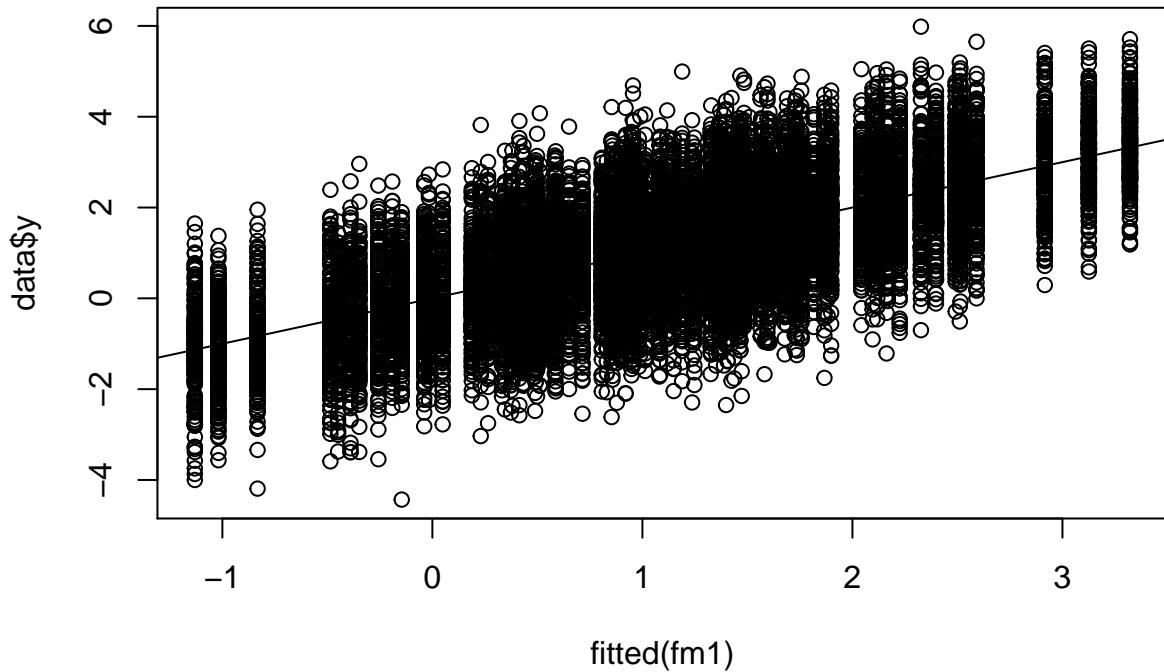
qqnorm(resid.level0,main='QQ plot for level 0 residuals')
qqline(resid.level0)
```

QQ plot for level 0 residuals



Finally, we can also assess the general fit of the model by plotting the true values against the fitted values. Ideally these point will lie close to the line $y = x$.

```
# Assess general fit of model
plot(fitted(fm1),data$y)
abline(0,1)
```



Again this looks fine.

Heavy tailed random errors

Now lets start again by generating some more data, but this time, instead of using a Gaussian distribution to generate the ϵ_{ij} , let's use a t-distribution with three degrees of freedom. This is heavy tailed compared to the Gaussian distribution.

```
b_i <- rnorm(n group) # random effects
epsilon <- rt(n group * group size, df=3)
y <- 1+b_i[group]+epsilon
data = data.frame(y, group = group)
fm1<- lmer(y~1+(1|group), data = data)
qqnorm(resid(fm1), main='QQ plot for level 1 residuals')
qqline(resid(fm1))

fitted.level0<-fm1@pp$X %*% fixef(fm1)
resid.level0<-y-fitted.level0
qqnorm(resid.level0, main='QQ plot for level 0 residuals')
qqline(resid.level0)
```

So in this case, we can clearly see that the level 1 and level 0 residuals are non-Gaussian. This suggests that the problem lies with the random errors. The random effects may also be non-Gaussian of course, but we can't assess this here as we know that if the random errors are non-Gaussian, then the residuals at all levels will appear non-Gaussian. We need to fix the problem with the random errors before we can assess the normality of the random effects.

Heavy tailed random effects

Now we will repeat the exercise. We will let the random errors be Gaussian, but use a t-distribution to generate the random effects.

QQ plot for level 1 residuals

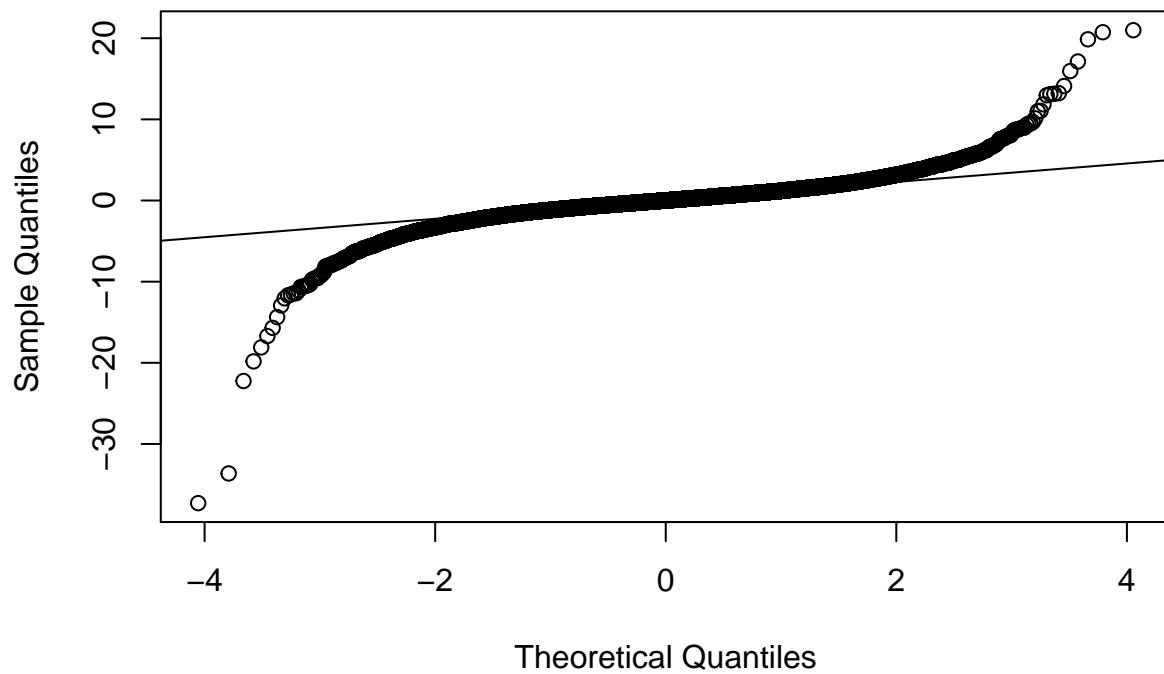


Figure 1:

QQ plot for level 0 residuals

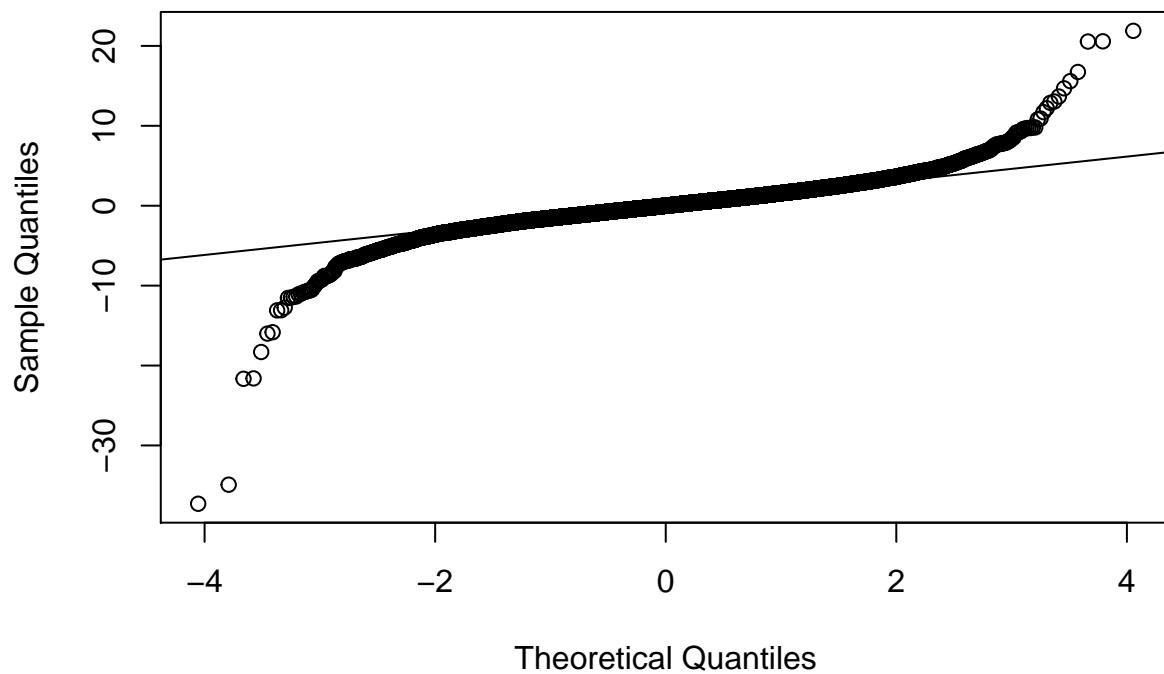


Figure 2:

QQ plot for level 1 residuals

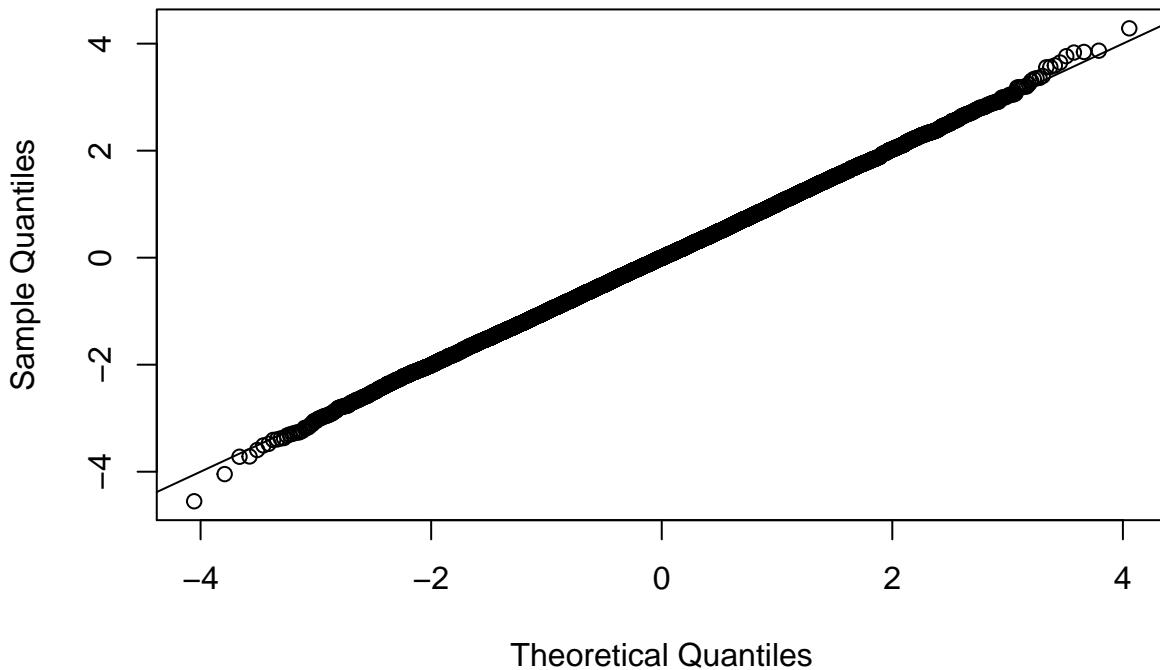


Figure 3:

```

b_i <- rt(ngroup, df=2) # random effects
epsilon <- rnorm(ngroup*groupsize)
y <- 1+b_i[group]+epsilon
data = data.frame(y, group = group)
fm1<- lmer(y~1+(1|group), data = data)
qqnorm(resid(fm1), main='QQ plot for level 1 residuals')
qqline(resid(fm1))

fitted.level0<-fm1@pp$X %*% fixef(fm1)
resid.level0<-y-fitted.level0
qqnorm(resid.level0, main='QQ plot for level 0 residuals')
qqline(resid.level0)

```

Now we see that the level 1 residual QQ plot looks okay, suggesting that the random errors are Gaussian (as indeed they are), but the level 0 residual QQ plot looks heavy tailed, suggesting that the random effects are heavy tailed (which they are).

Exercise

Repeat this exercise but using a different wrong model. Two options would be to make the variance depend on some covariate x , or to build correlation into the random effect terms.

Another exercise would be to try this again but using a smaller number of groups. Above, I used 100 groups in the data, which is much larger than in many examples. This makes deviations from normality much easier to spot. Try using 5 and 10 groups - are the errors still obvious?

QQ plot for level 0 residuals

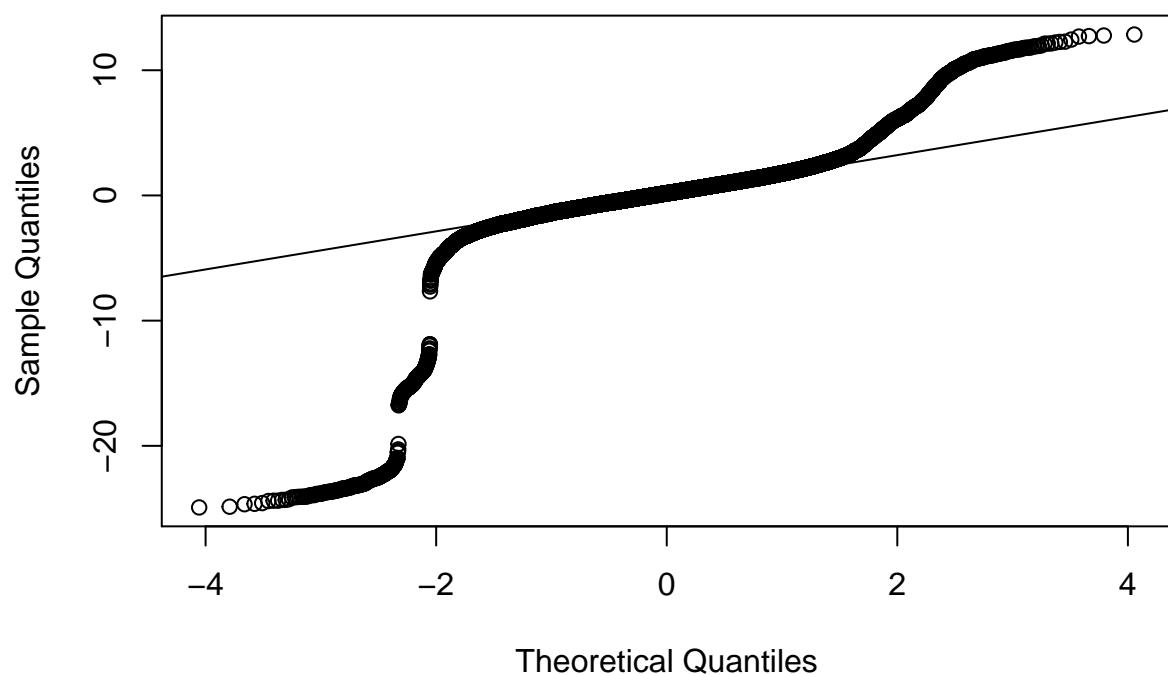


Figure 4: