

```

## -----
## Code Supplement for "ggplot2 Compatible Quantile-Quantile Plots in R"
## -----

## ---- Load the package -----
library(qqplotr)

## Loading required package: ggplot2

##
## Attaching package: 'qqplotr'

## The following objects are masked from 'package:ggplot2':
##
##   stat_qq_line, StatQqLine
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
library(gridExtra)

##
## Attaching package: 'gridExtra'

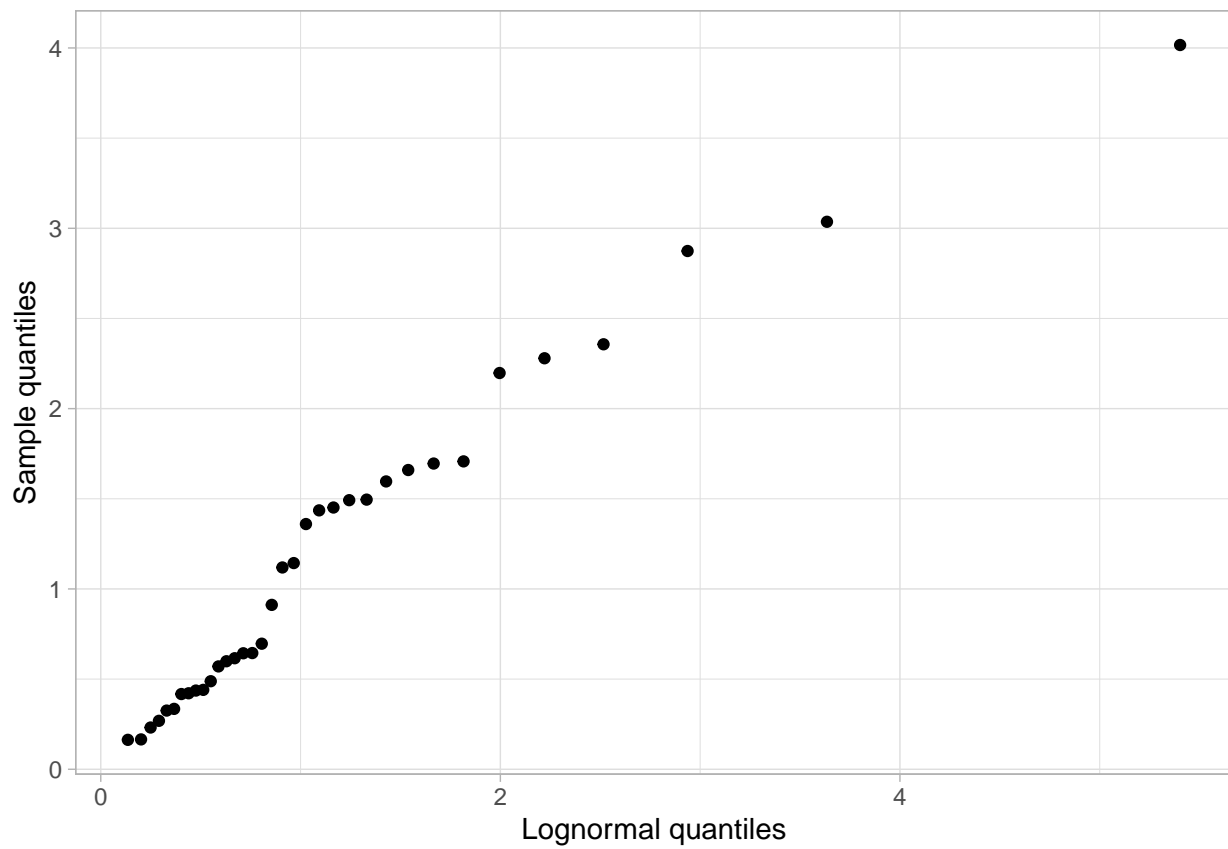
## The following object is masked from 'package:dplyr':
##
##   combine

## -----
## Background
## -----

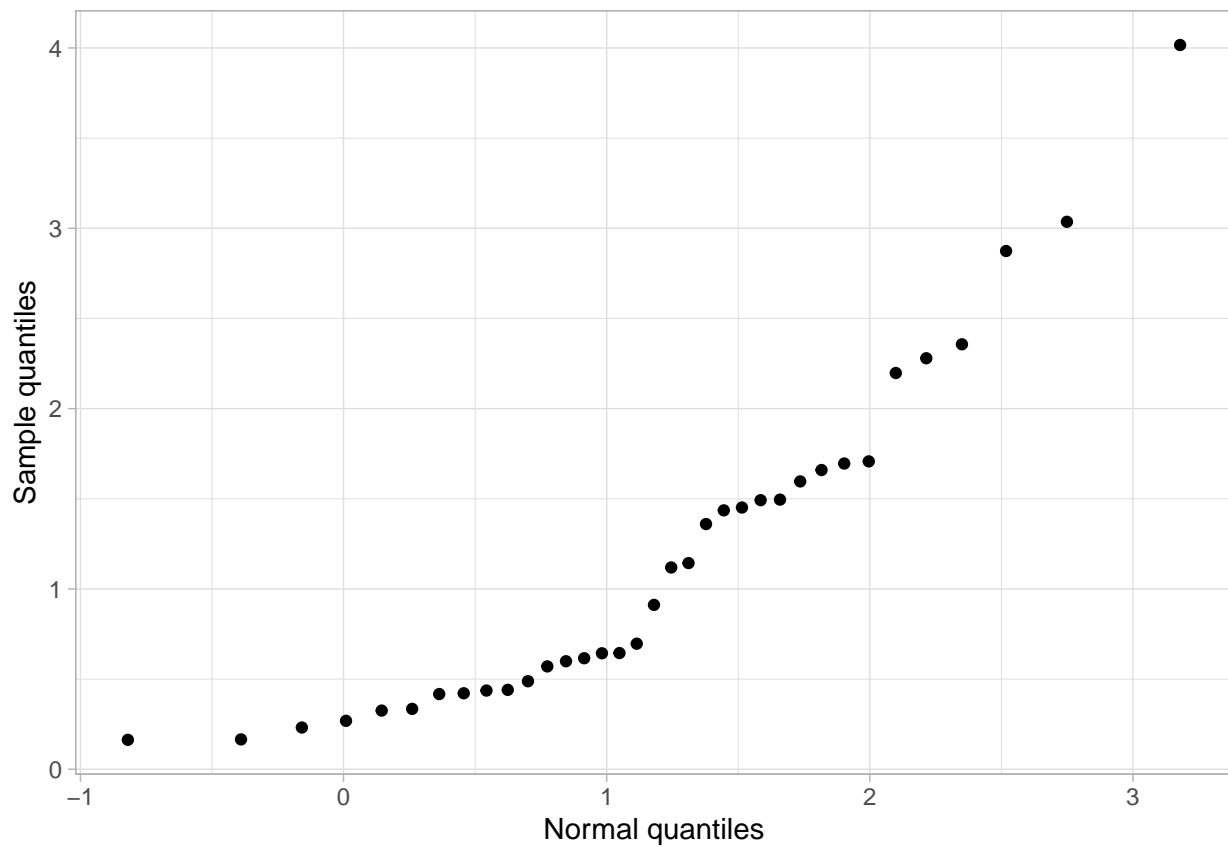
## ---- Code for Figure 1
set.seed(952017)
rand_sample <- data.frame(rs = rlnorm(35, meanlog = 0, sdlog = 1))

ggplot(data = rand_sample, mapping = aes(sample = rs)) +
  stat_qq_point(distribution = "lnorm") +
  ylab("Sample quantiles") +
  xlab("Lognormal quantiles") +
  theme_light()

```

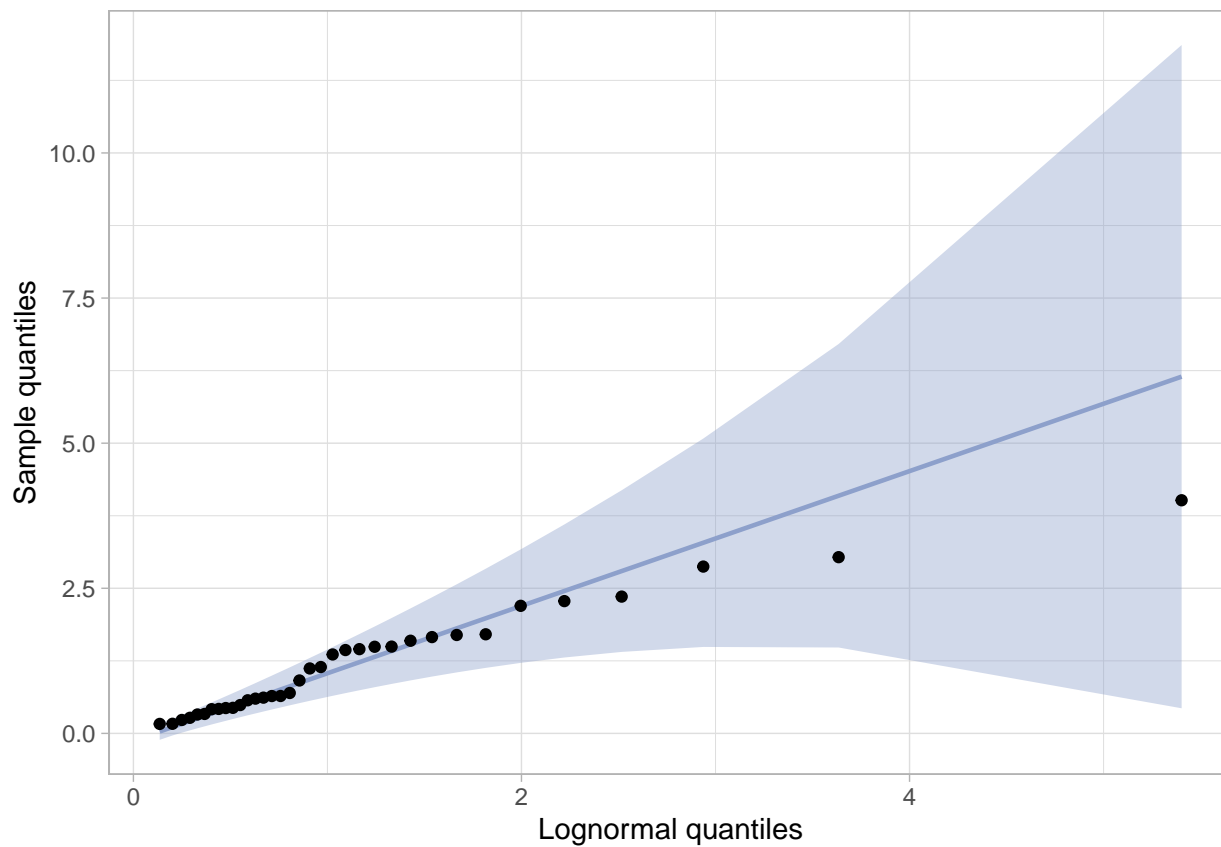


```
ggplot(data = rand_sample, mapping = aes(sample = rs)) +  
  stat_qq_point() +  
  ylab("Sample quantiles") +  
  xlab("Normal quantiles") +  
  theme_light()
```

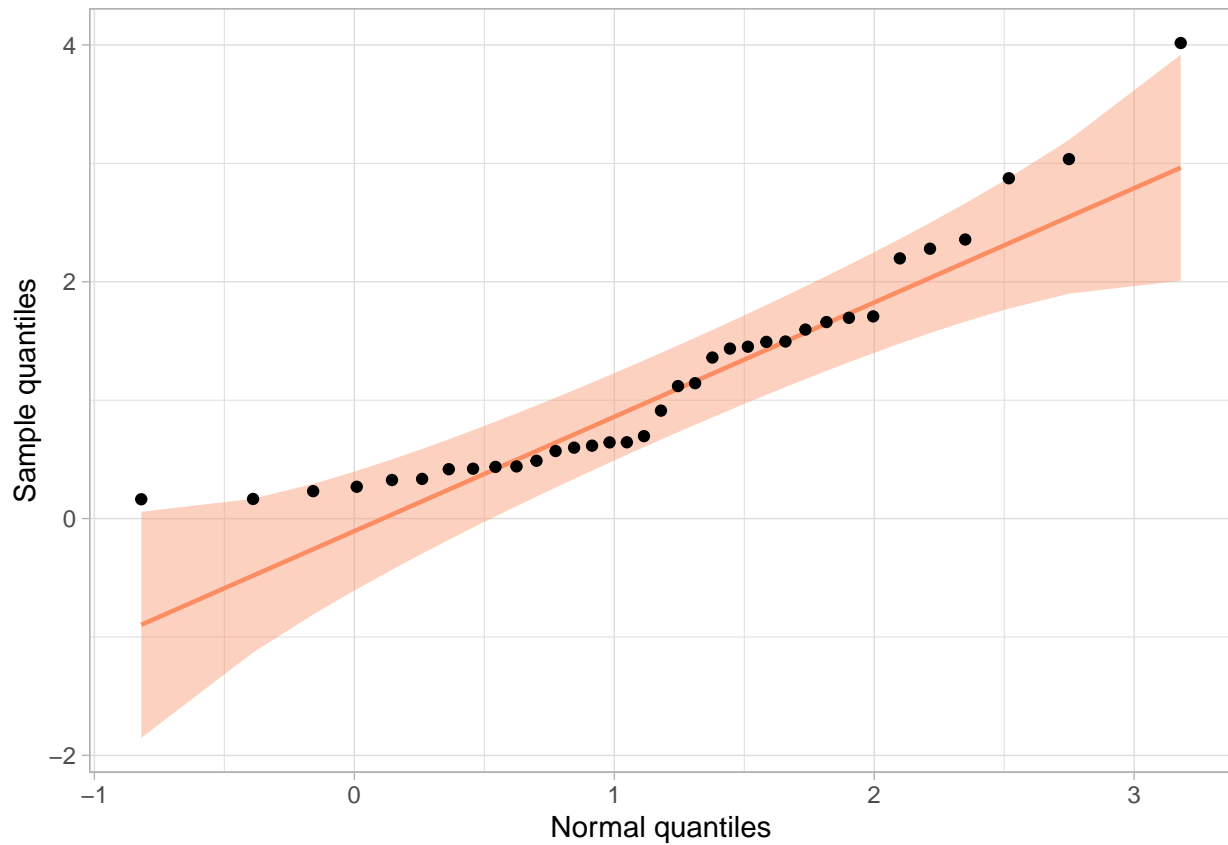


---- Code for Figure 2

```
ggplot(data = rand_sample, mapping = aes(sample = rs)) +  
  stat_qq_band(distribution = "lnorm", fill = "#8DA0CB", alpha = 0.4) +  
  stat_qq_line(distribution = "lnorm", colour = "#8DA0CB") +  
  stat_qq_point(distribution = "lnorm") +  
  ylab("Sample quantiles") +  
  xlab("Lognormal quantiles") +  
  theme_light()
```

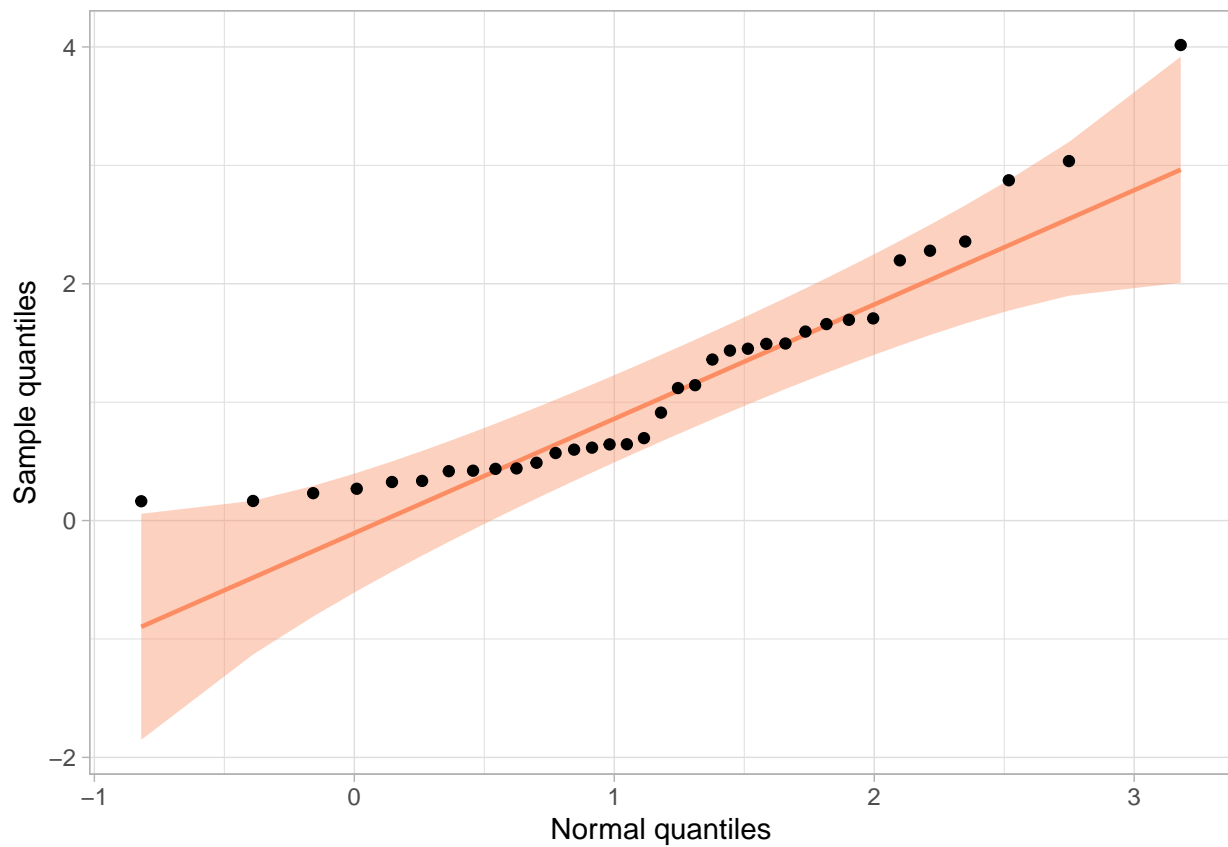


```
ggplot(data = rand_sample, mapping = aes(sample = rs)) +  
  stat_qq_band(fill = "#FC8D62", alpha = 0.4) +  
  stat_qq_line(colour = "#FC8D62") +  
  stat_qq_point() +  
  ylab("Sample quantiles") +  
  xlab("Normal quantiles") +  
  theme_light()
```

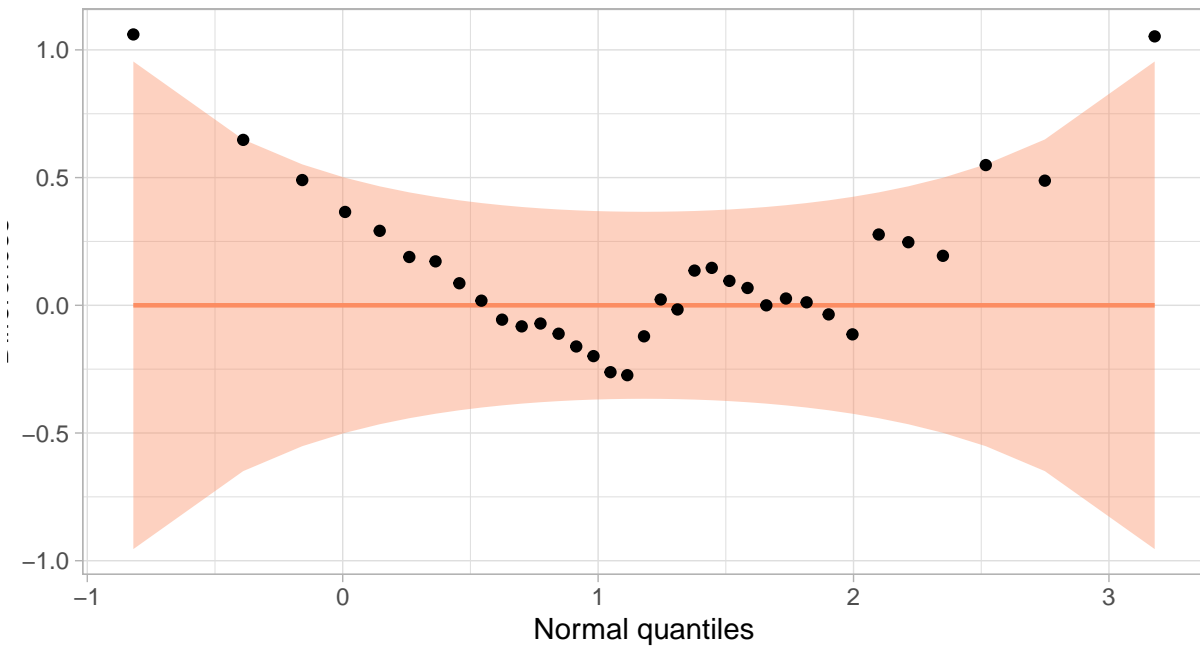


```
## ---- Code for Figure 3
set.seed(952017)
rand_sample <- data.frame(rs = rlnorm(35, meanlog = 0, sdlog = 1))

ggplot(data = rand_sample, mapping = aes(sample = rs)) +
  stat_qq_band(fill = "#FC8D62", alpha = 0.4) +
  stat_qq_line(colour = "#FC8D62") +
  stat_qq_point() +
  ylab("Sample quantiles") +
  xlab("Normal quantiles") +
  theme_light()
```



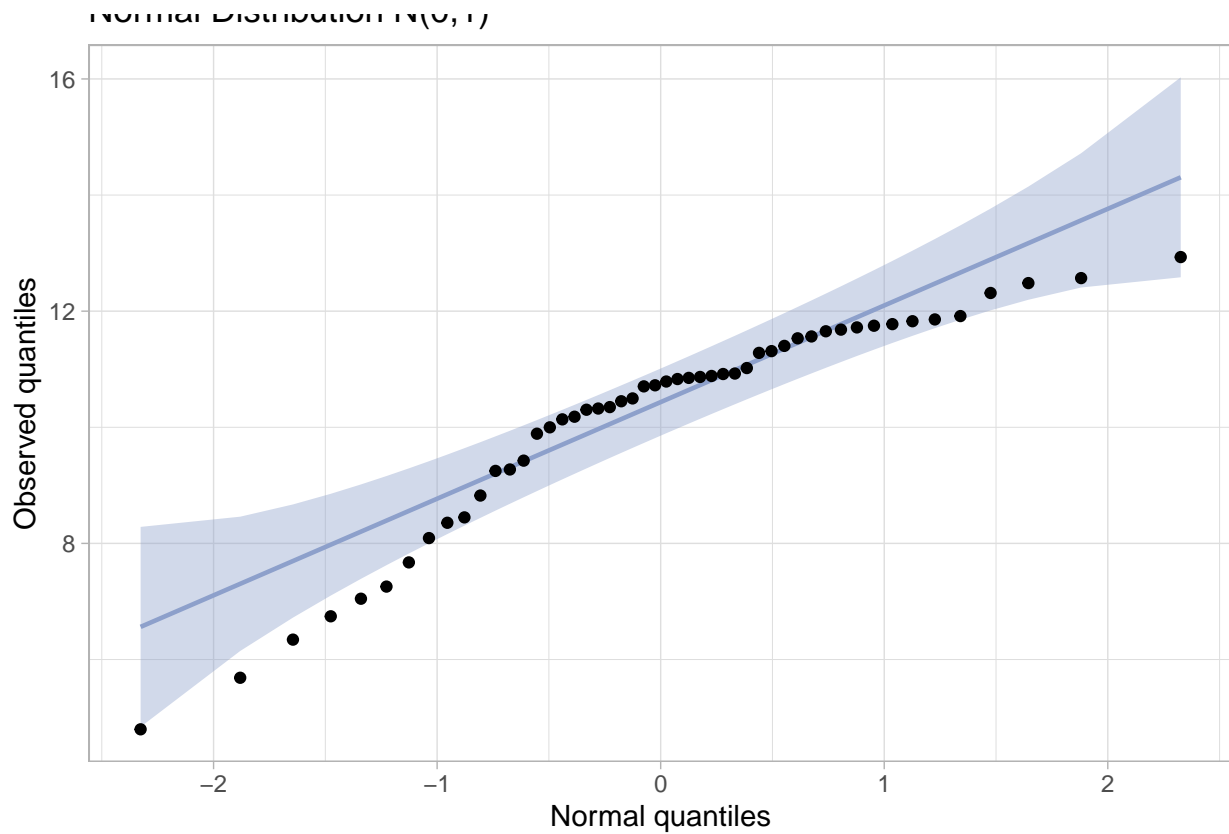
```
ggplot(data = rand_sample, mapping = aes(sample = rs)) +  
  stat_qq_band(detrend = TRUE, fill = "#FC8D62", alpha = 0.4) +  
  stat_qq_line(detrend = TRUE, colour = "#FC8D62") +  
  stat_qq_point(detrend = TRUE) +  
  ylab("Differences") +  
  xlab("Normal quantiles") +  
  theme_light() +  
  coord_fixed(ratio = 1)
```



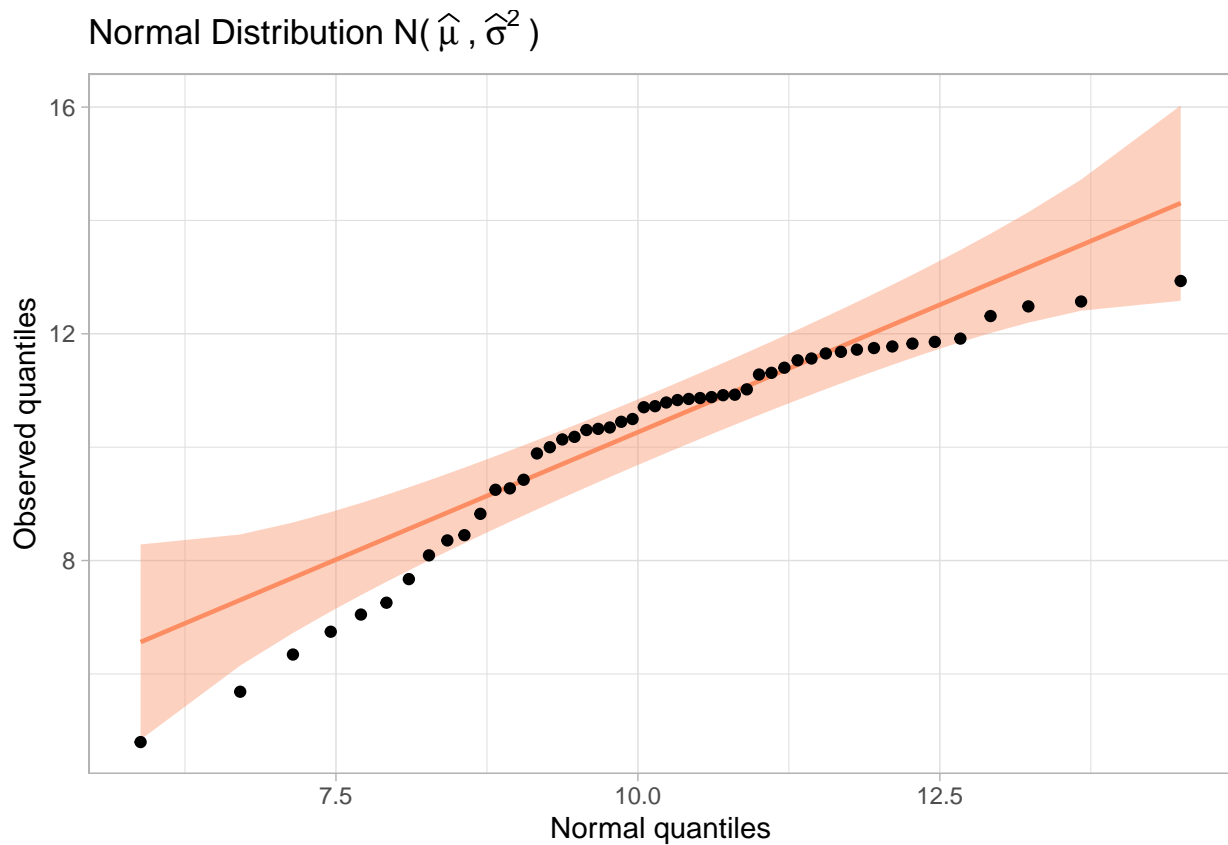
```
## -----
## Implementing Q-Q plots in the ggplot2 framework
## -----

## ---- Code for Figure 4
dframe <- data.frame(xnorm = rnorm(50, mean = 10, sd = 2),
                     xlnorm = rlnorm(50, mean = 2, sd = .5))

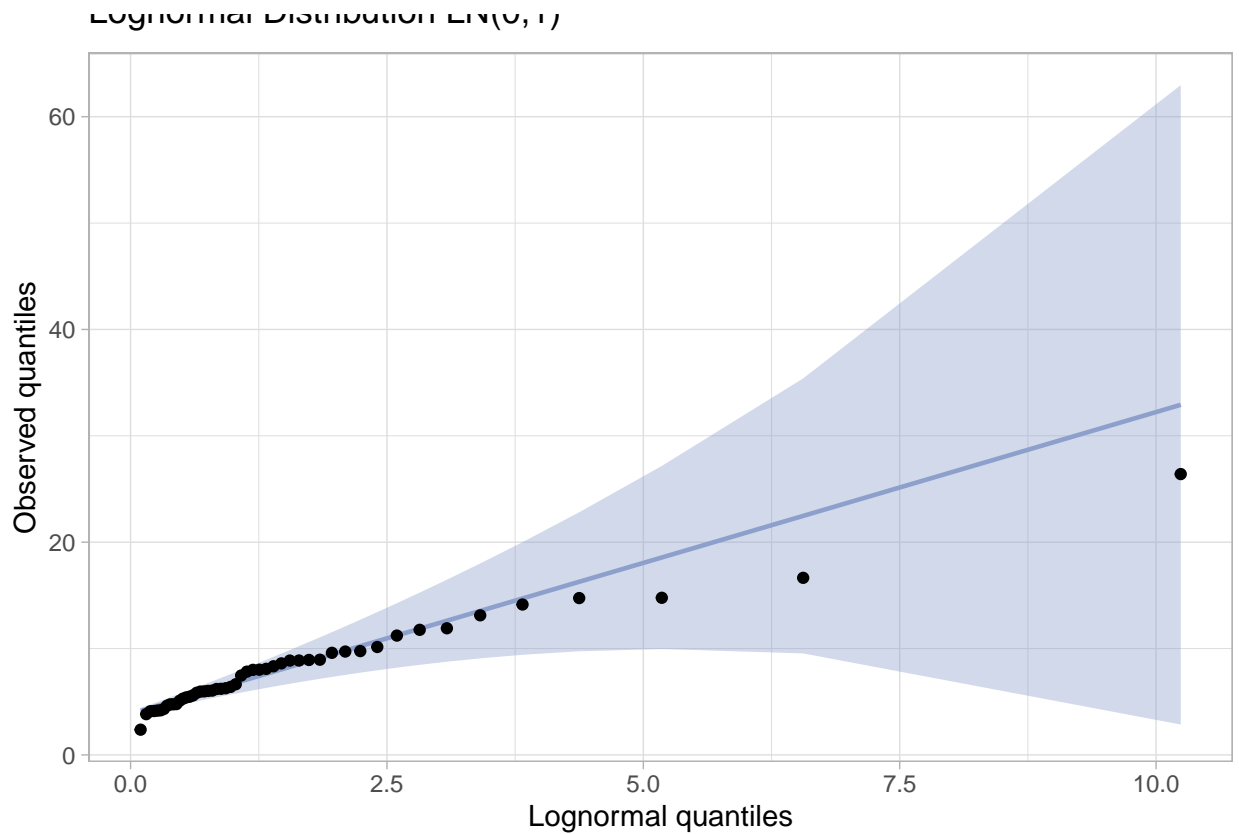
dframe %>% ggplot(aes(sample = xnorm)) +
  geom_qq_band(dparams = list(mean = 0, sd = 1), fill = "#8DA0CB", alpha = 0.4) +
  stat_qq_line(dparams = list(mean = 0, sd = 1), colour = "#8DA0CB") +
  stat_qq_point(dparams = list(mean = 0, sd = 1)) +
  theme_light() +
  xlab("Normal quantiles") +
  ylab("Observed quantiles") +
  ggtitle("Normal Distribution N(0,1)")
```



```
dframe %>% ggplot(aes(sample = xnorm)) +
  geom_qq_band(fill = "#FC8D62", alpha = 0.4) +
  stat_qq_line(colour = "#FC8D62") +
  stat_qq_point() +
  theme_light() +
  xlab("Normal quantiles") +
  ylab("Observed quantiles") +
  ggtitle(expression("Normal Distribution N(" ~ \hat{\mu} ~ ", " ~ \hat{\sigma}^2 ~ ")"))
```

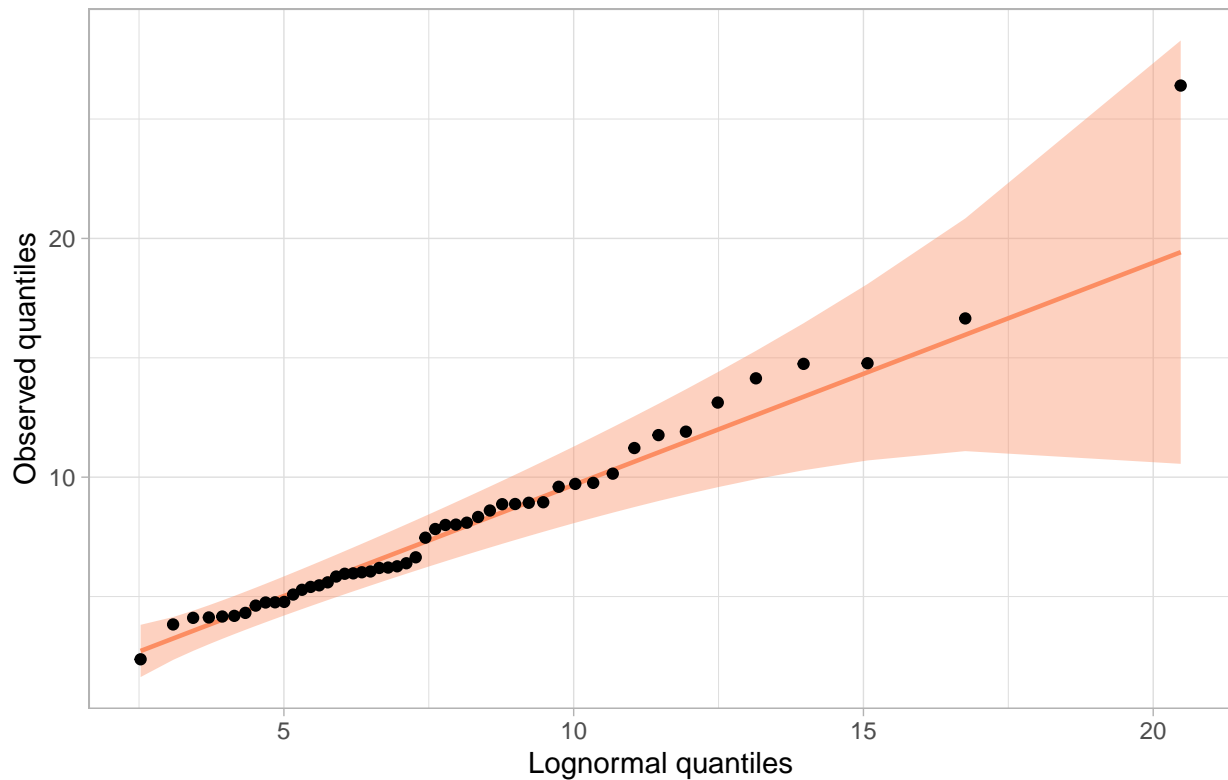



```
dframe %>% ggplot(aes(sample = xlnorm)) +
  geom_qq_band(distribution = "lnorm", dparams = list(mean = 0, sd = 1), fill = "#8DA0CB", alpha = 0.4)
  stat_qq_line(distribution = "lnorm", dparams = list(mean = 0, sd = 1), colour = "#8DA0CB") +
  stat_qq_point(distribution = "lnorm", dparams = list(mean = 0, sd = 1)) +
  theme_light() +
  xlab("Lognormal quantiles") +
  ylab("Observed quantiles") +
  ggtitle(expression("Lognormal Distribution LN(0,1)"))
```



```
dframe %>% ggplot(aes(sample = xlnorm)) +
  geom_qq_band(distribution = "lnorm", fill = "#FC8D62", alpha = 0.4) +
  stat_qq_line(distribution = "lnorm", colour = "#FC8D62") +
  stat_qq_point(distribution = "lnorm") +
  theme_light() +
  xlab("Lognormal quantiles") +
  ylab("Observed quantiles") +
  ggtitle(expression("Lognormal Distribution LN(" ~ \widehat{\mu} ~ "," ~ \widehat{\sigma}^2 ~ ")"))
```

Lognormal Distribution $\text{LN}(\hat{\mu}, \hat{\sigma}^2)$



```
## -----
## Constructing Q-Q plots with qqplotr
## -----

## ---- Code for Figure 5
data(urine, package = "boot")
library(boot)
set.seed(9)

p11 <- urine %>% ggplot(aes(sample = ph)) +
  stat_qq_band(bandType = "pointwise", fill = "#8DA0CB", alpha = 0.4) +
  stat_qq_line(colour = "#8DA0CB") +
  stat_qq_point() +
  ggtitle("Pointwise") +
  xlab("Normal quantiles") +
  ylab("pH measurements quantiles") +
  theme_light() +
  coord_cartesian(ylim = c(3.2, 8.7))

p12 <- urine %>% ggplot(aes(sample = ph)) +
  stat_qq_band(bandType = "ks", fill = "#FC8D62", alpha = 0.4) +
  stat_qq_line(colour = "#FC8D62") +
  stat_qq_point() +
  ggtitle("Kolmogorov-type") +
  xlab("Normal quantiles") +
  ylab("pH measurements quantiles") +
  theme_light() +
```

```

    theme(axis.text.y = element_blank(),
          axis.ticks.y = element_blank(),
          axis.title.y = element_blank(),
          plot.title = element_text(vjust = 0),
          plot.margin = margin(t = 3.5, r = 5.5, b = 5.5, l = 5.5, unit = "pt")) +
    coord_cartesian(ylim = c(3.2, 8.7))

p13 <- urine %>% ggplot(aes(sample = ph)) +
  stat_qq_band(bandType = "ts", fill = "#66C2A5", alpha = 0.4) +
  stat_qq_line(colour = "#66C2A5") +
  stat_qq_point() +
  ggtitle("Tail-sensitive") +
  xlab("Normal quantiles") +
  ylab("Observed quantiles of pH measurements") +
  theme_light() +
  theme(axis.text.y = element_blank(),
        axis.ticks.y = element_blank(),
        axis.title.y = element_blank()) +
  coord_cartesian(ylim = c(3.2, 8.7))

p21 <- urine %>% ggplot(aes(sample = ph)) +
  stat_qq_band(bandType = "pointwise", fill = "#8DA0CB", alpha = 0.4, detrend = TRUE) +
  stat_qq_line(colour = "#8DA0CB", detrend = TRUE) +
  stat_qq_point(detrend = TRUE) +
  ggtitle("Pointwise (Detrended)") +
  xlab("Normal quantiles") +
  ylab("Differences") +
  theme_light() +
  coord_fixed(ratio=1, ylim = c(-1.5, 1.5))

p22 <- urine %>% ggplot(aes(sample = ph)) +
  stat_qq_band(bandType = "ks", fill = "#FC8D62", alpha = 0.4, detrend = TRUE) +
  stat_qq_line(colour = "#FC8D62", detrend = TRUE) +
  stat_qq_point(detrend = TRUE) +
  ggtitle("Kolmogorov-type (Detrended)") +
  xlab("Normal quantiles") +
  ylab("Differences") +
  theme_light() +
  theme(axis.text.y = element_blank(),
        axis.ticks.y = element_blank(),
        axis.title.y = element_blank(),
        plot.title = element_text(vjust = 0),
        plot.margin = margin(t = 3.5, r = 5.5, b = 5.5, l = 5.5, unit = "pt")) +
  coord_fixed(ratio=1, ylim = c(-1.5, 1.5))

p23 <- urine %>% ggplot(aes(sample = ph)) +
  stat_qq_band(bandType = "ts", fill = "#66C2A5", alpha = 0.4, detrend = TRUE) +
  stat_qq_line(colour = "#66C2A5", detrend = TRUE) +
  stat_qq_point(detrend = TRUE) +
  ggtitle("Tail-sensitive (Detrended)") +
  xlab("Normal quantiles") +
  ylab("Differences") +
  theme_light() +

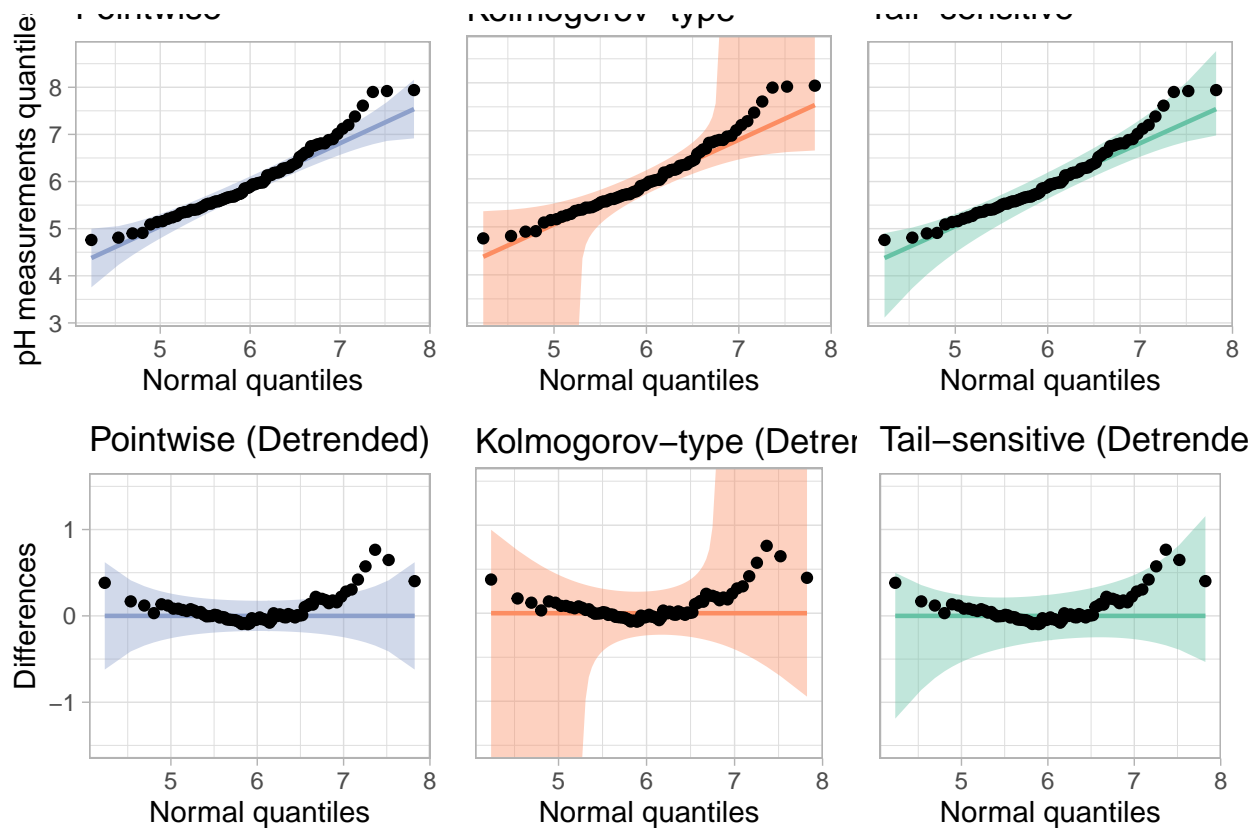
```

```

theme(axis.text.y = element_blank(),
      axis.ticks.y = element_blank(),
      axis.title.y = element_blank()) +
coord_fixed(ratio=1, ylim = c(-1.5, 1.5))

grid.arrange(p11, p12, p13,
             p21, p22, p23,
             nrow = 2,
             ncol = 3,
             widths = c(1.11, 1, 1))

```



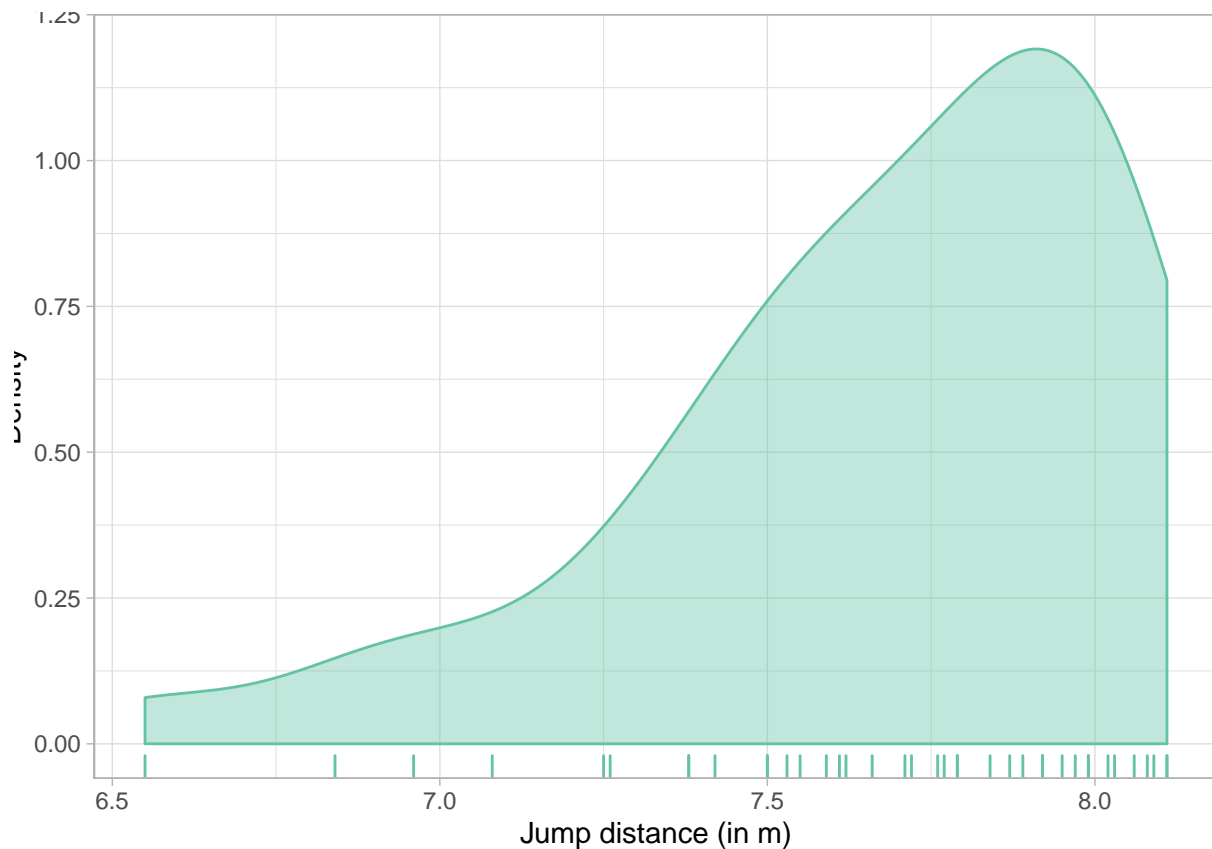
```

## -----
## User-provided distributions
## -----

## ---- Loading longjump data
data("longjump", package = "qqplotr")
longjump <- na.omit(longjump)

## ---- Code for Figure 6
ggplot(longjump, aes(x = distance)) +
  geom_density(fill = "#66C2A5", colour = "#66C2A5", alpha = 0.4) +
  geom_rug(colour = "#66C2A5") +
  xlab("Jump distance (in m)") +
  ylab("Density") +
  theme_light()

```



```
## ---- Defining SEV distribution function
# CDF
psev <- function(q, mu = 0, sigma = 1) {
  z <- (q - mu) / sigma
  1 - exp(-exp(z))
}

# PDF
dsev <- function(x, mu = 0, sigma = 1) {
  z <- (x - mu) / sigma
  (1 / sigma) * exp(z - exp(z))
}

# Quantile function
qsev <- function(p, mu = 0, sigma = 1) {
  mu + log(-log(1 - p)) * sigma
}

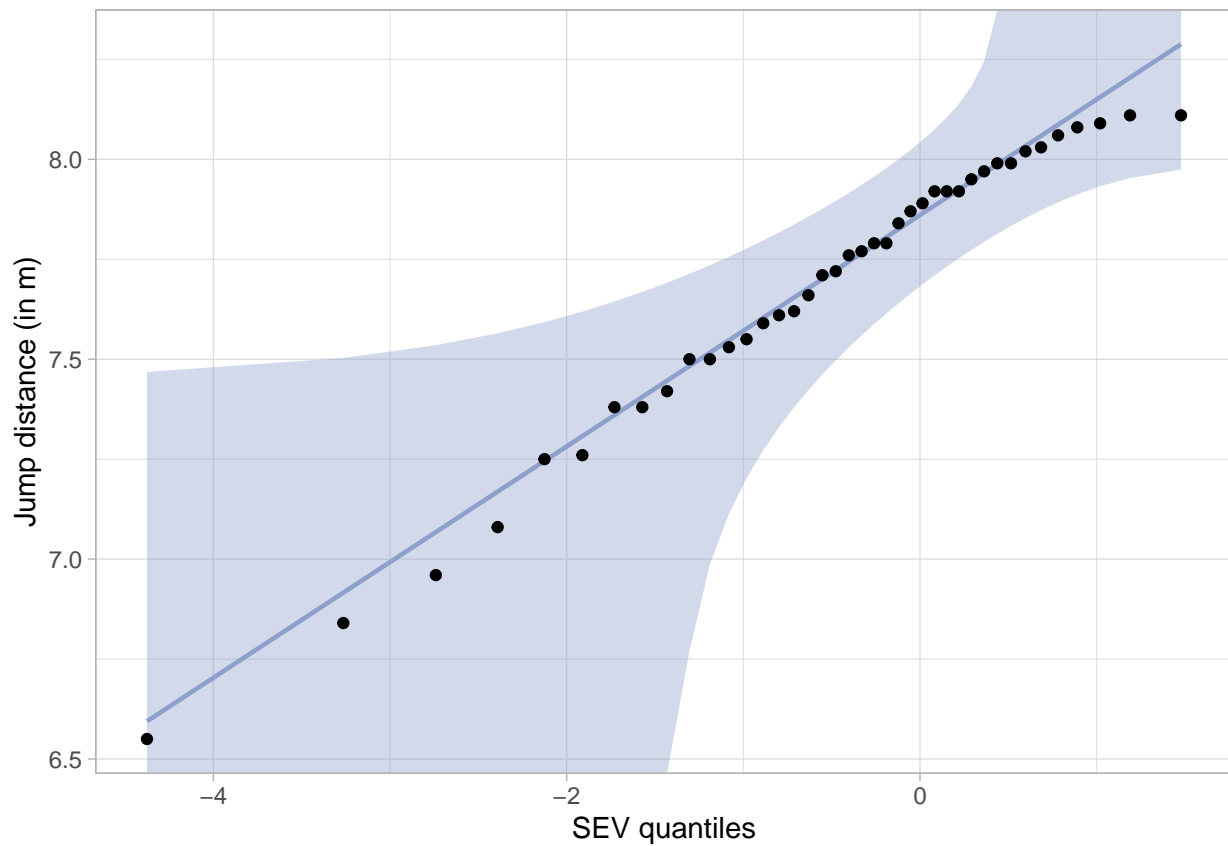
# Simulation function
rsev <- function(n, mu = 0, sigma = 1) {
  qsev(runif(n), mu, sigma)
}

## ---- Code for Figure 7
ggplot(longjump, aes(sample = distance)) +
  stat_qq_band(distribution = "sev",
              bandType = "ks",
```

```

dparams = list(mu = 0, sigma = 1),
fill = "#8DA0CB",
alpha = 0.4) +
stat_qq_line(distribution = "sev",
colour = "#8DA0CB",
dparams = list(mu = 0, sigma = 1)) +
stat_qq_point(distribution = "sev",
dparams = list(mu = 0, sigma = 1)) +
xlab("SEV quantiles") +
ylab("Jump distance (in m)") +
theme_light()

```



```

## -----
## Detrending Q-Q plots
## -----

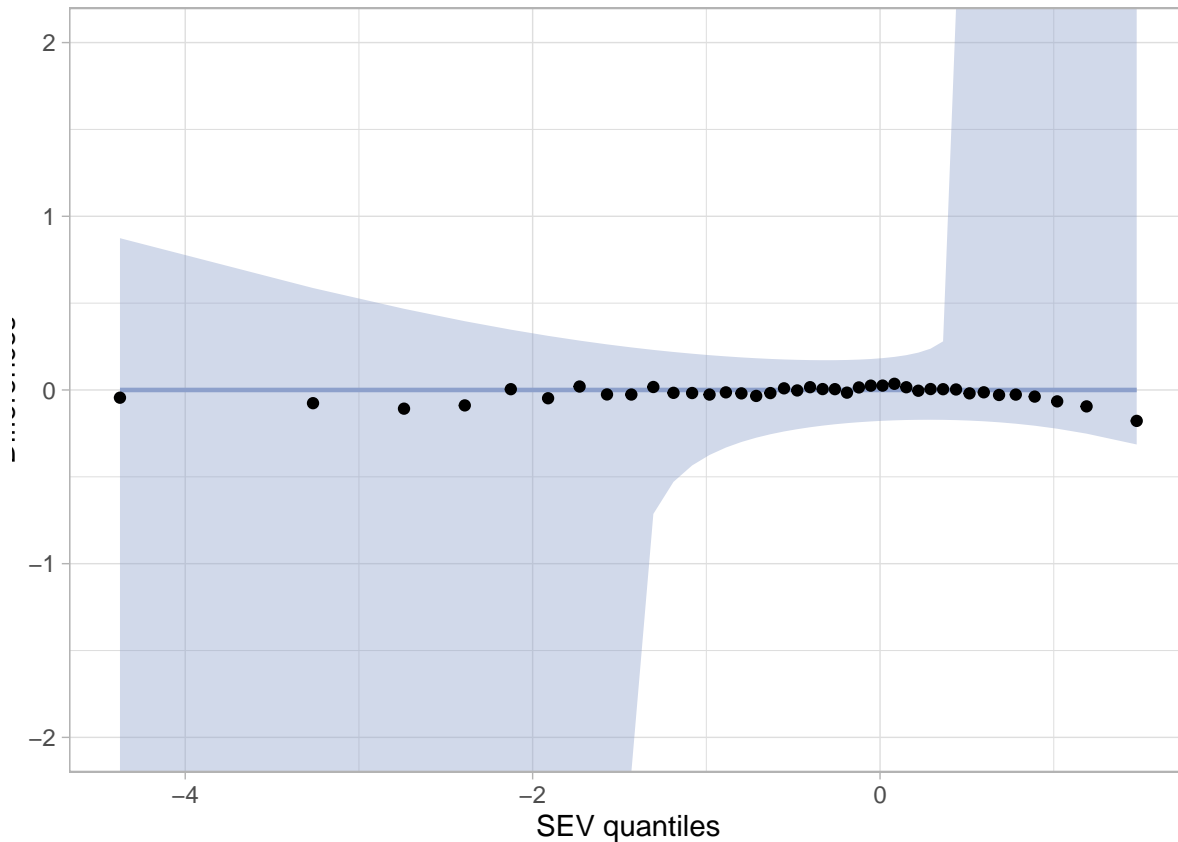
## ---- Code for Figure 8
ggplot(longjump, aes(sample = distance)) +
  stat_qq_band(distribution = "sev",
    bandType = "ks",
    detrend = TRUE,
    dparams = list(mu = 0, sigma = 1),
    fill = "#8DA0CB",
    alpha = 0.4) +
  stat_qq_line(distribution = "sev",
    detrend = TRUE,

```

```

dparams = list(mu = 0, sigma = 1),
colour = "#8DA0CB") +
stat_qq_point(distribution = "sev",
detrend = TRUE,
dparams = list(mu = 0, sigma = 1)) +
xlab("SEV quantiles") +
ylab("Differences") +
theme_light() +
coord_fixed(ratio = 1, ylim = c(-2, 2))

```



```

## -----
## BRFSS Example
## -----

data("iowa", package = "qqplotr")
set.seed(3145)

## ---- Data preprocessing
sample_ia <- iowa %>%
  tidyr::nest(-SEX) %>%
  mutate(
    data = data %>%
      purrr::map(.f = function(x) sample_n(x, size = 200))
  ) %>%
  tidyr::unnest(data) %>%
  dplyr::select(SEX, WTKG3, HTIN4) %>%

```



```

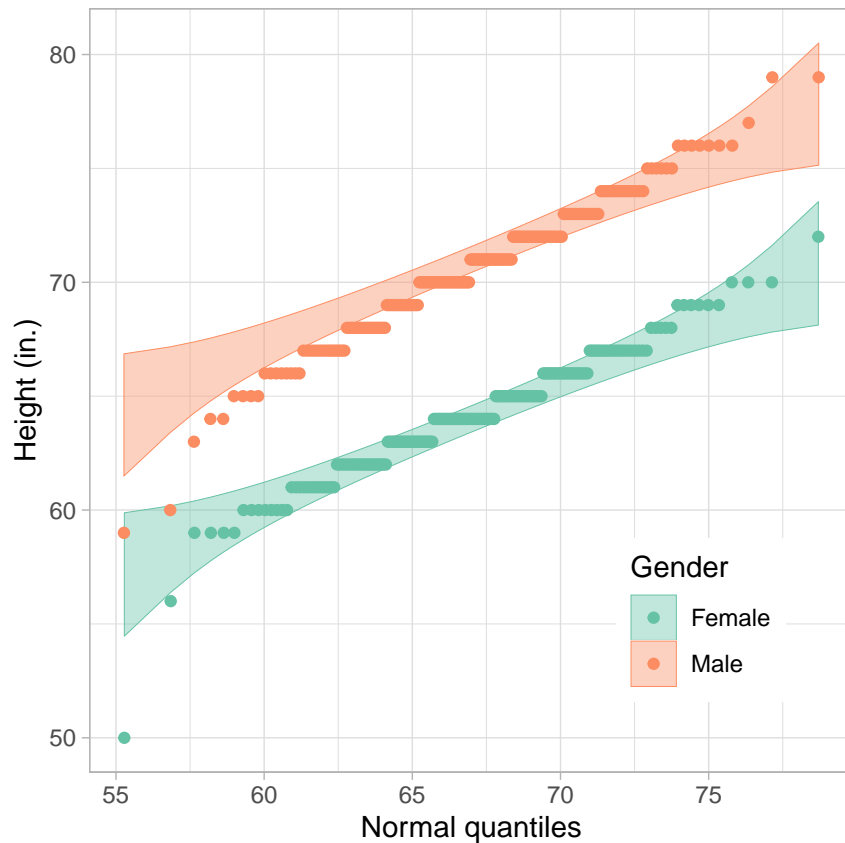
mutate(Gender = c("Male", "Female")[SEX])

## ---- Code for figure 9
params <- iowa %>%
  filter(!is.na(HTIN4)) %>%
  summarize(m = mean(HTIN4), s = sd(HTIN4))

customization <- list(scale_fill_brewer(palette = "Set2"),
  scale_colour_brewer(palette = "Set2"),
  xlab("Normal quantiles"),
  ylab("Height (in.)"),
  coord_equal(),
  theme_light(),
  theme(legend.position = c(0.8, 0.2), aspect.ratio = 1))

sample_ia %>%
  ggplot(aes(sample = HTIN4, colour=Gender, fill=Gender)) +
  stat_qq_band(bandType = "ts",
    alpha = 0.4,
    dparams = list(mean = params$m, sd = params$s)) +
  stat_qq_point(dparams = list(mean = params$m, sd = params$s)) +
  customization

```



```

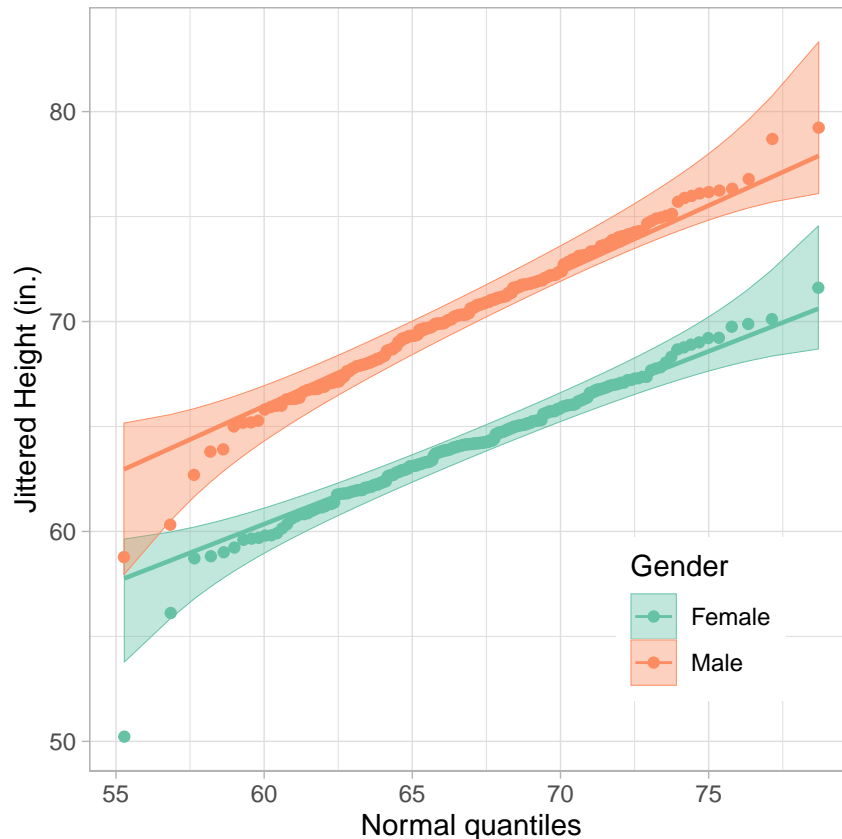
sample_ia %>%
  mutate(HTIN4.jitter = jitter(HTIN4, factor = 2)) %>%
  ggplot(aes(sample = HTIN4.jitter, colour = Gender, fill = Gender)) +

```

```

stat_qq_band(bandType = "ts",
             alpha = 0.4,
             dparams = list(mean = params$m, sd = params$s)) +
stat_qq_line(dparams = list(mean = params$m, sd = params$s)) +
stat_qq_point(dparams = list(mean = params$m, sd = params$s)) +
customization +
ylab("Jittered Height (in.)")

```



```

## ---- Code for table 1
bygender <- iowa %>%
  group_by(SEX) %>%
  summarize(mean.height = mean(HTIN4, na.rm = TRUE),
            sd.height = sd(HTIN4, na.rm = TRUE),
            mean.weight = mean(log(WTKG3), na.rm = TRUE),
            sd.weight = sd(log(WTKG3), na.rm = TRUE))
bygender$SEX <- c("Male", "Female")

total <- iowa %>%
  summarize(mean.height = mean(HTIN4, na.rm = TRUE),
            sd.height = sd(HTIN4, na.rm = TRUE),
            mean.weight = mean(log(WTKG3), na.rm = TRUE),
            sd.weight = sd(log(WTKG3), na.rm = TRUE))

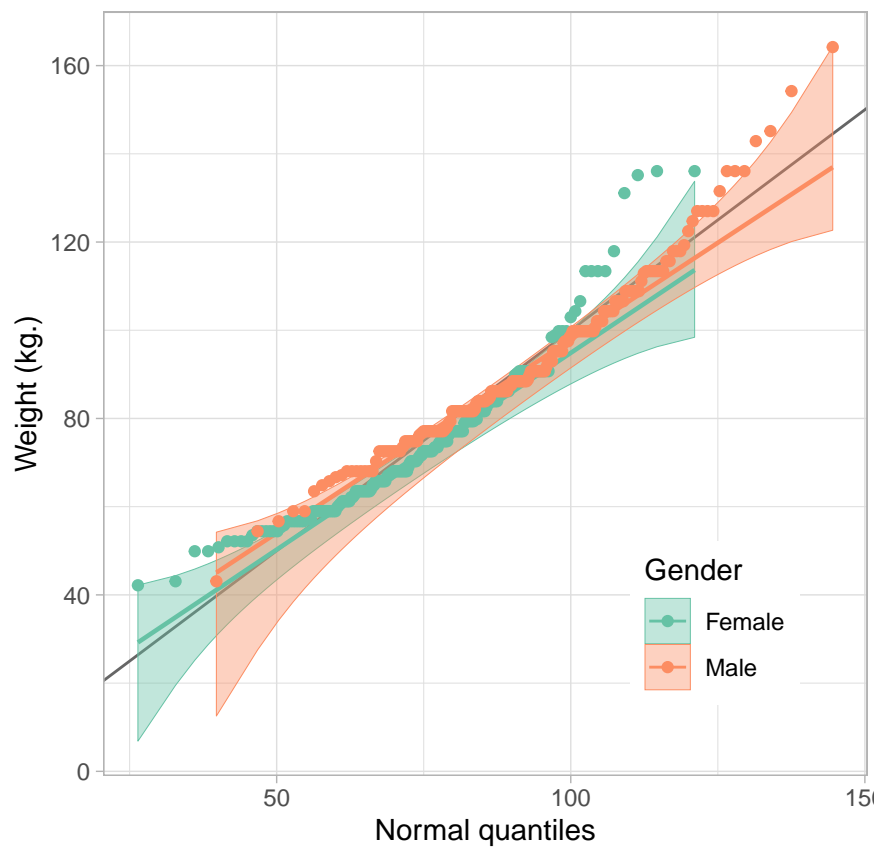
bygender[3,] <- c("Total", total)
names(bygender) <- c("SEX", "mean height (in)", "sd (in)", "mean log weight (kg)", "sd (kg)")
knitr::kable(bygender)

```

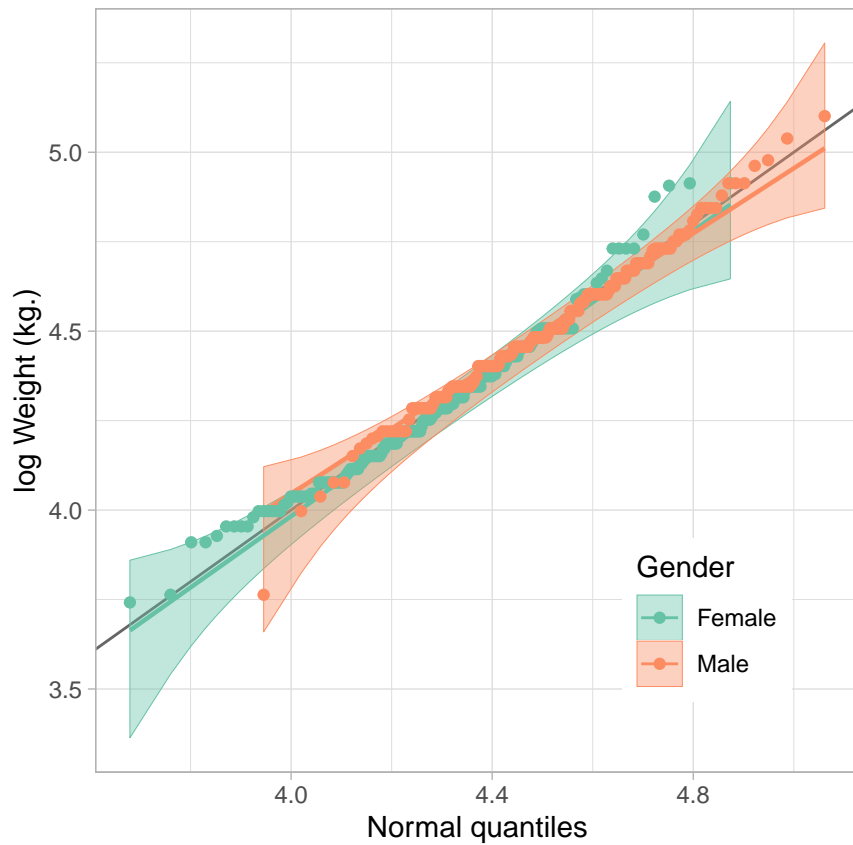
SEX	mean height (in)	sd (in)	mean log weight (kg)	sd (kg)
Male	70.55043	2.966460	9.100936	0.2011216
Female	64.50972	2.911454	8.887950	0.2254663
Total	66.99109	4.176295	8.978768	0.2397852

---- Code for figure 10

```
sample_ia %>%
  ggplot(aes(sample = WTKG3 / 100, colour = Gender, fill = Gender)) +
  geom_abline(colour = "grey40") +
  stat_qq_band(bandType = "ts", alpha = 0.4) +
  stat_qq_line() +
  stat_qq_point() +
  customization +
  ylab("Weight (kg.)")
```



```
sample_ia %>%
  ggplot(aes(sample = log(WTKG3/100), colour=Gender, fill=Gender)) +
  geom_abline(colour = "grey40") +
  stat_qq_band(bandType = "ts", alpha = 0.4) +
  stat_qq_line() +
  stat_qq_point() +
  customization +
  ylab("log Weight (kg.)")
```



---- Code for figure 11

```
sample_ia %>%
  ggplot(aes(sample = WTKG3 / 100, colour = Gender, fill = Gender)) +
  geom_abline(colour = "grey40") +
  stat_qq_band(bandType = "ks", distribution = "lnorm", alpha = 0.4) +
  stat_qq_line(distribution = "lnorm") +
  stat_qq_point(distribution = "lnorm") +
  customization +
  facet_grid(. ~ Gender) +
  xlab("Lognormal quantiles") +
  ylab("Weight (kg.)") +
  theme(legend.position = c(0.9, 0.2))
```

