

Transmission Risk Comparison

Remo Schmutz

2022-12-21

Libraries

```
library(tidyverse)
library(ggplot2)
require("knitr")
library(gridExtra)
library(grid)
library(lubridate)
library(dplyr)
library(hms)
library(truncdist)
library(crch)
library(tidyr)
library(stats)
library(LaplacesDemon)
library(ggstatsplot)
library(MASS)
library(fitdistrplus)
library(truncnorm)
library(tidybayes)
library(ggpubr)
library(psych)
library(ggpattern)
library(gridExtra)
library(GGally)
library(cowplot)
library(vcd)
library(boot)
library(magrittr)

text_size = 8
theme_bw2 <- function () {
  theme_bw(base_size = text_size, base_family = "sans") %+replace%
    theme(
      axis.text = element_text(size = text_size),
      axis.title = element_text(size = text_size),
      plot.title = element_text(size = text_size + 2, face = "bold", hjust = 0, margin = ggplot2::margin%)
    )
}
```

Data import

```
ch <- readRDS("../data-clean/palas.rds")
ch_old <- readRDS("../data-clean/co2-ch.rds") #swiss data
satz <- readRDS("../data-clean/co2-sa-tz.rds")

ch <- ch %>%
  rename(co2 = CO2) %>%
  filter(co2 < 4000) %>% #we assume these are outliers (measurement error)
  mutate(co2 = ifelse(co2 < 400, 400, co2)) %>% #replace co2 values less than 400 with 400
  mutate(time_posix = strptime(time, format = "%H:%M")) %>% #change format
  mutate(time_h = hour(time_posix)) %>% #extract hours
  filter(time_h >= 8 & time_h <= 17) %>% #filter hours when pupils ar present
  arrange(co2) #arrange by co2 for later

sa <- satz %>%
  filter(country == "South Africa") %>%
  filter(co2 < 4000) %>% #we assume these are outliers (measurement error)
  mutate(co2 = ifelse(co2 < 400, 400, co2)) %>% #replace co2 values less than 400 with 400
  arrange(co2)

tz <- satz %>%
  filter(country == "Tanzania") %>%
  filter(co2 < 4000) %>% #we assume these are outliers (measurement error)
  mutate(co2 = ifelse(co2 < 400, 400, co2)) %>% #replace co2 values less than 400 with 400
  mutate(time_h = hour(date)) %>% #extract hour
  filter(time_h >= 8) %>% #filter time when pupils are present (info from Lukas)
  arrange(co2) #tanzania data
```

Methods

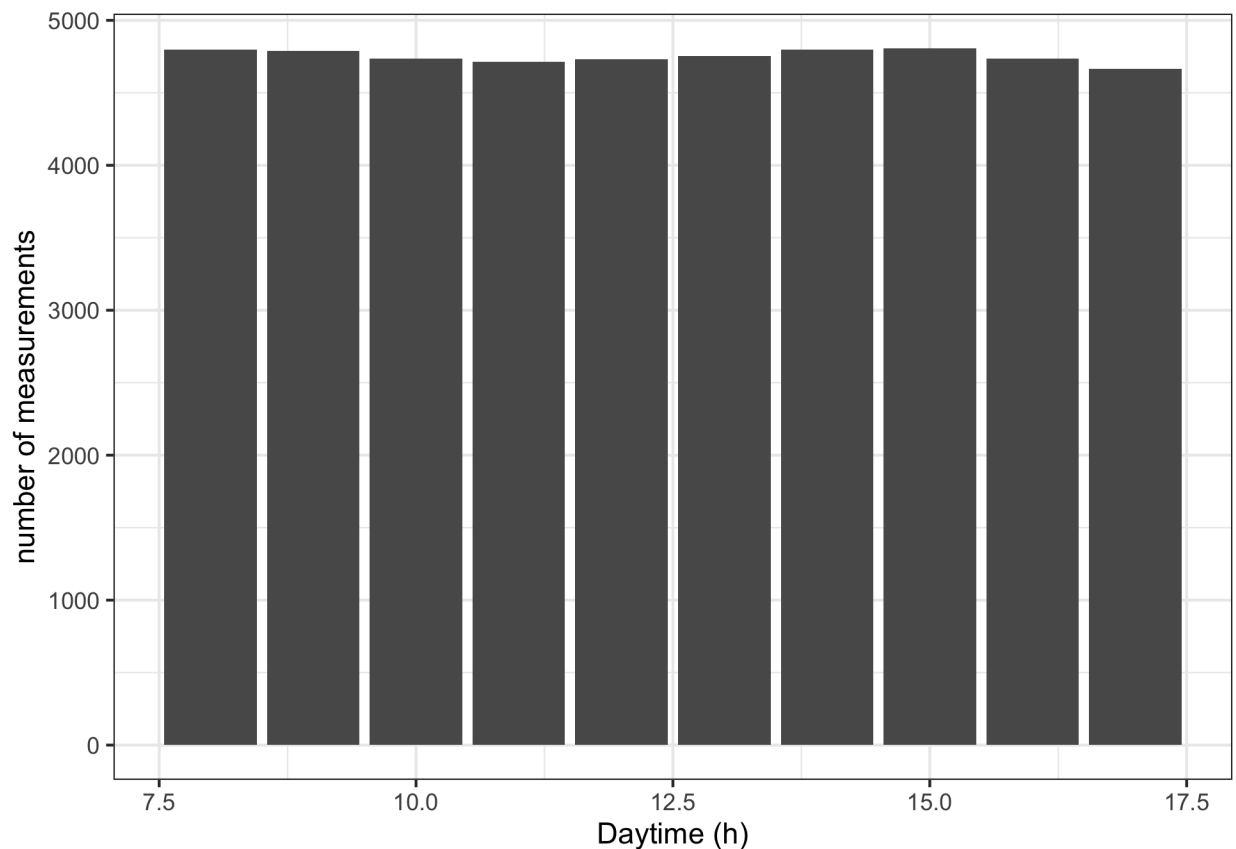
Indoor Co2 concentration

- * mean or distribution from data
- * C_o := Outdoor Co2 concentration * from literature https://www.fsis.usda.gov/sites/default/files/media_file/2020-08/Carbon-Dioxide.pdf
- * C_a := Volume fraction of CO2 added to exhaled breath during breathing
- * Persily and de Jonge [Table 3 and 4] doi: 10.1111/ina.12383
- * $\bar{f} := \int_{t=0}^{t=max} f dt$
- * integrating over f values from different times (2) or using a distribution based on the data
- * I := Number of infectors in the class
- * estimated using prevalence of the age group in the country
- * q := Quantum per hour
- * assuming a distribution from literature
- * t := time
- * changing this parameter to compare
- * n := number of people in the class
- * data (Switzerland) or assumption (South Africa, Tanzania)

Preprocess

```
ch_hourly <- ch %>%
  group_by(time_h) %>% #to analyze each hour seperately
  summarize(mean = mean(co2),
            lower = quantile(co2, 0.25),
            upper = quantile(co2, 0.75),
            n_data = n()) %>%
  ungroup()

ch_hourly %>%
  ggplot(aes(x = time_h, y = n_data)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  xlab("Daytime (h)") +
  ylab("number of measurements") +
  theme_bw() +
  theme(legend.position = c(0.9,0.9), legend.title = element_blank())+
  scale_color_brewer(palette = "Set2") +
  scale_fill_brewer(palette = "Set2") #there are sufficient measurements althrough the day (this informati
```



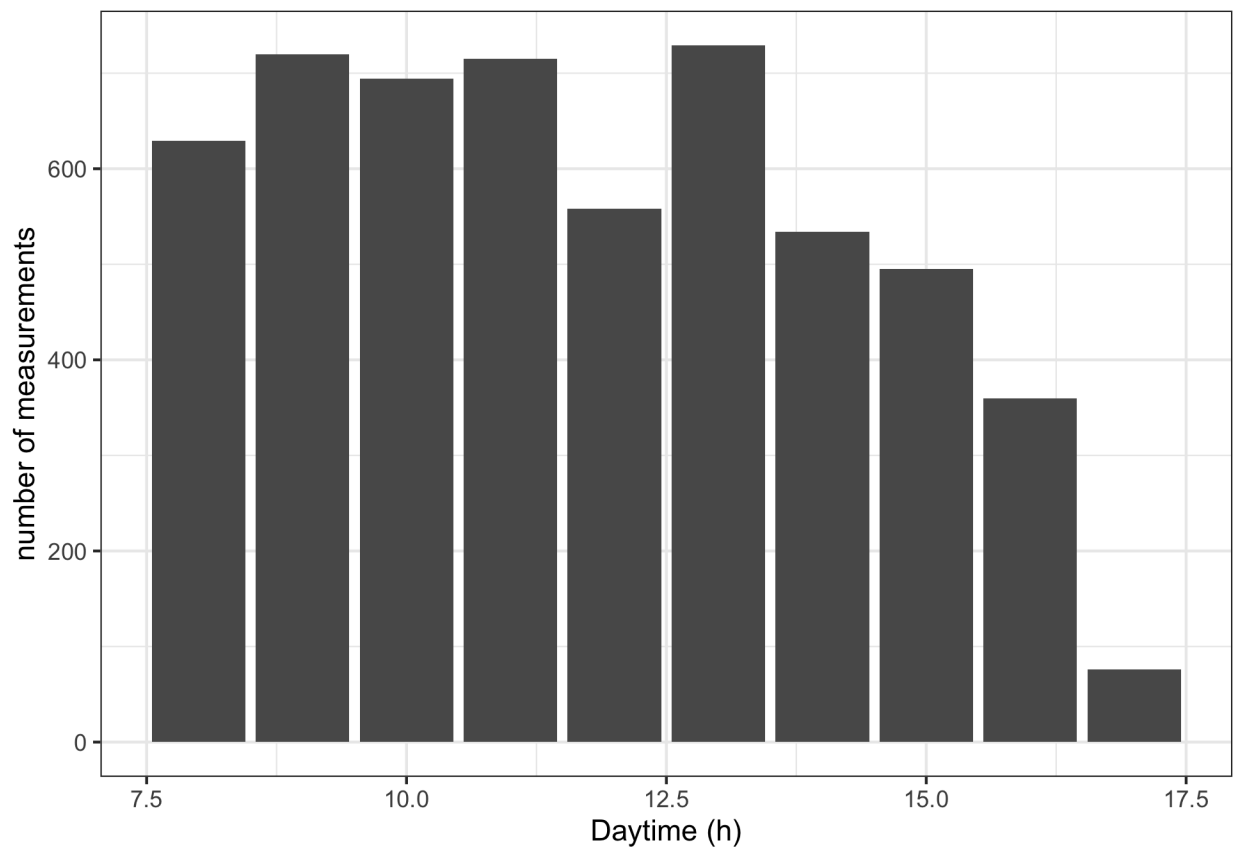
```
tz_hourly <- tz %>%
  mutate(time_h = hour(date)) %>%
  group_by(time_h) %>%
  summarize(mean = mean(co2),
```

```

    lower = quantile(co2, 0.25),
    upper = quantile(co2, 0.75),
    n_data = n()) %>%
  ungroup()

tz_hourly %>%
  ggplot(aes(x = time_h, y = n_data)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  xlab("Daytime (h)") +
  ylab("number of measurements") +
  theme_bw() +
  theme(legend.position = c(0.9,0.9), legend.title = element_blank())

```



data is measured throughout the day in south africa, there is no information about which time the mea
This is the reason, we won't use this information, as we don't have it for all three countries

Analysis

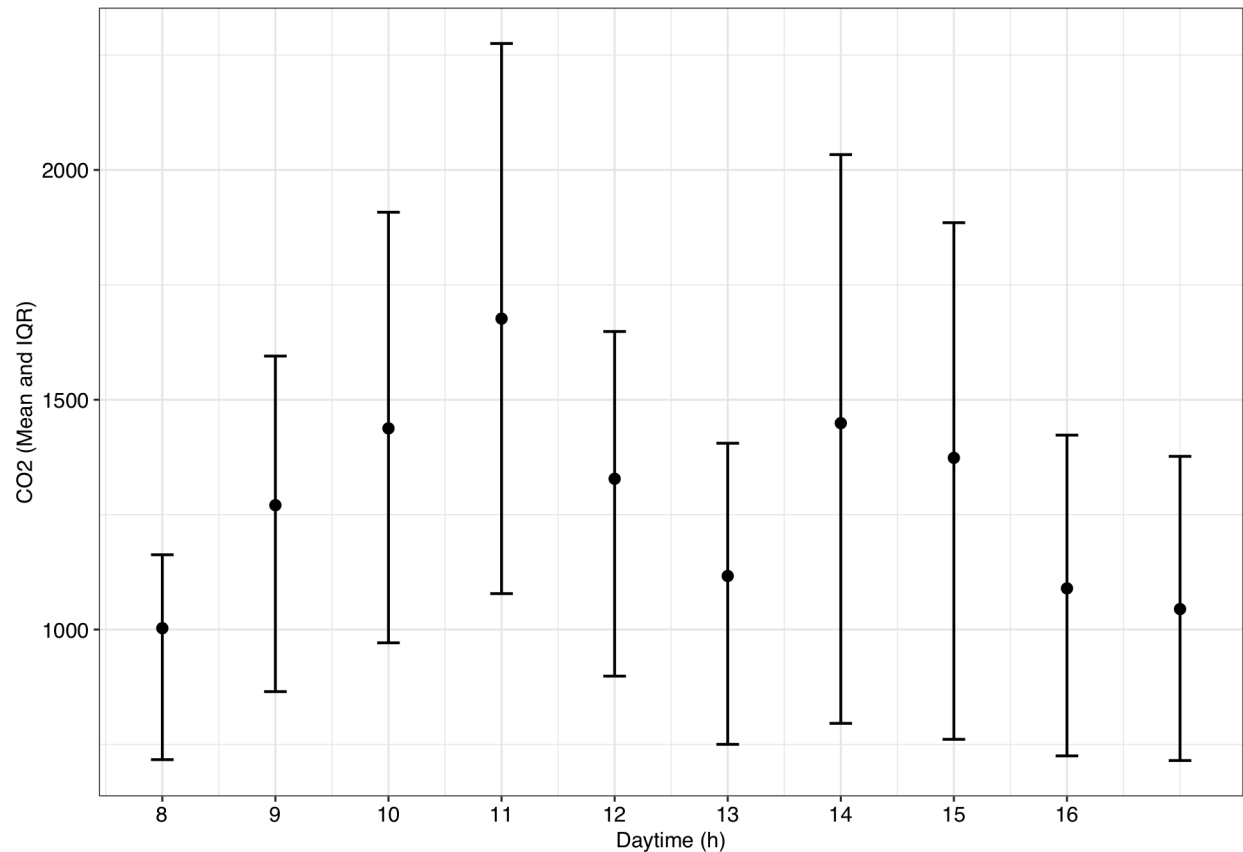
Co2 over time

```

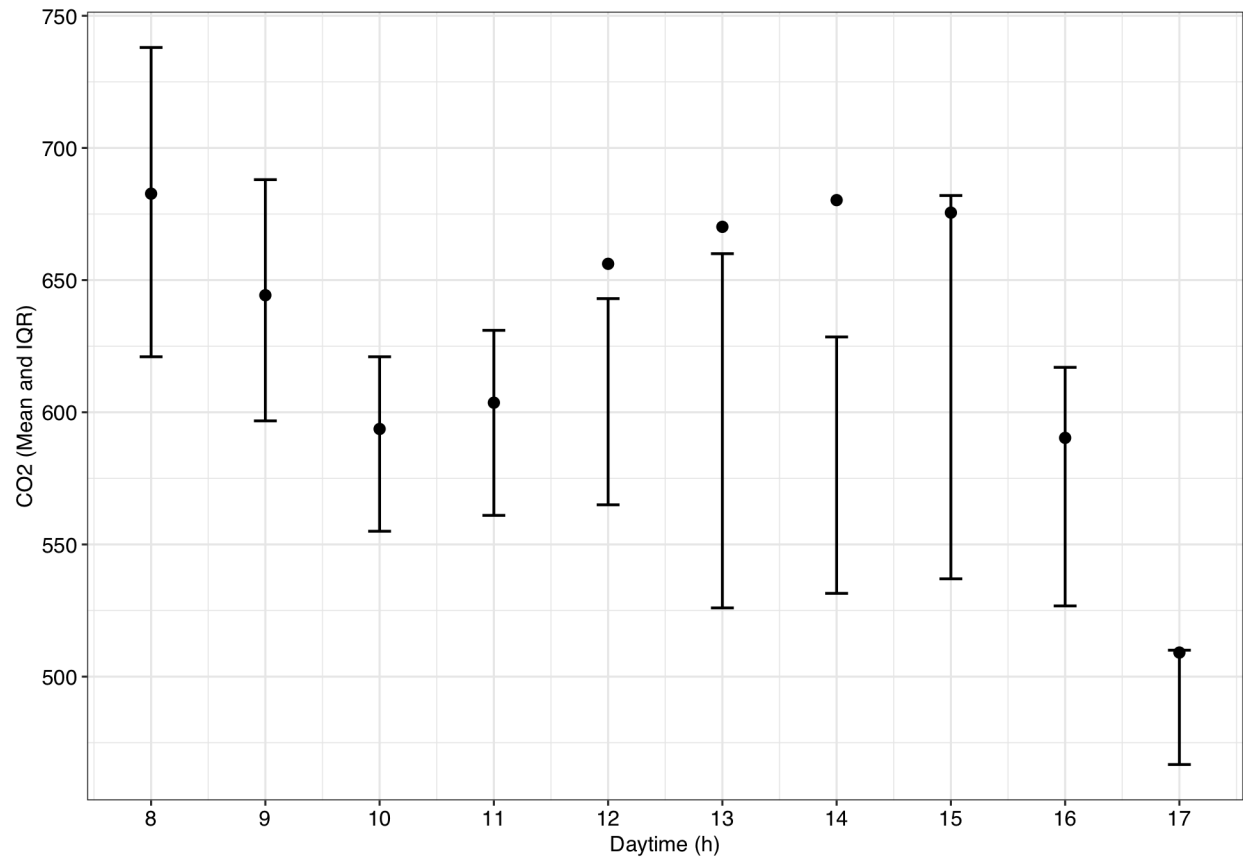
ch_hourly %>% #plot co2 during the day (ch)
  ggplot(aes(x = time_h)) +
  geom_errorbar(aes(ymin = lower, ymax = upper), width = .2, position = position_dodge2(width = .2))

```

```
geom_point(aes(y = mean), position = position_dodge2(width = .2)) +
scale_color_brewer(palette = "Set2") +
scale_x_continuous(breaks = seq(7, 16, 1)) +
labs(x = "Daytime (h)", y = "CO2 (Mean and IQR)") +
theme_bw2() +
theme(legend.position = c(0.9,0.9), legend.title = element_blank())
```



```
tz_hourly %>% #plot co2 during the day (tz)
ggplot(aes(x = time_h)) +
geom_errorbar(aes(ymin = lower, ymax = upper), width = .2, position = position_dodge2(width = .2)) +
geom_point(aes(y = mean), position = position_dodge2(width = .2)) +
scale_color_brewer(palette = "Set2") +
scale_x_continuous(breaks = seq(7, 17, 1)) +
labs(x = "Daytime (h)", y = "CO2 (Mean and IQR)") +
theme_bw2() +
theme(legend.position = c(0.9,0.9), legend.title = element_blank())
```



*#these plots show how the co2 concentration (mean, lower and upper quantil) changes during a day.
#no time data available for south africa*

Smoothing Data

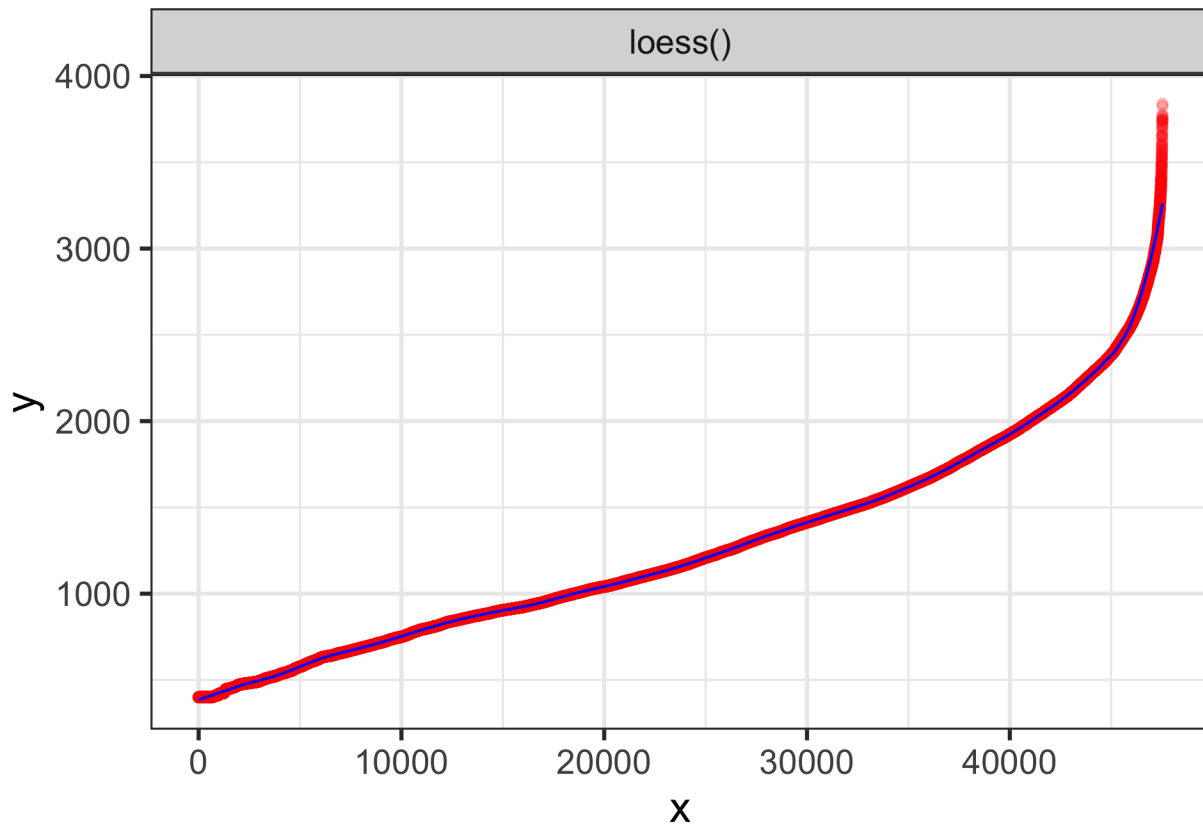
```
# Switzerland
set.seed(1343)

dat_ch <- data.frame(
  x = 1:length(ch$co2),
  y = ch$co2
)#preparing dataset for Loess method

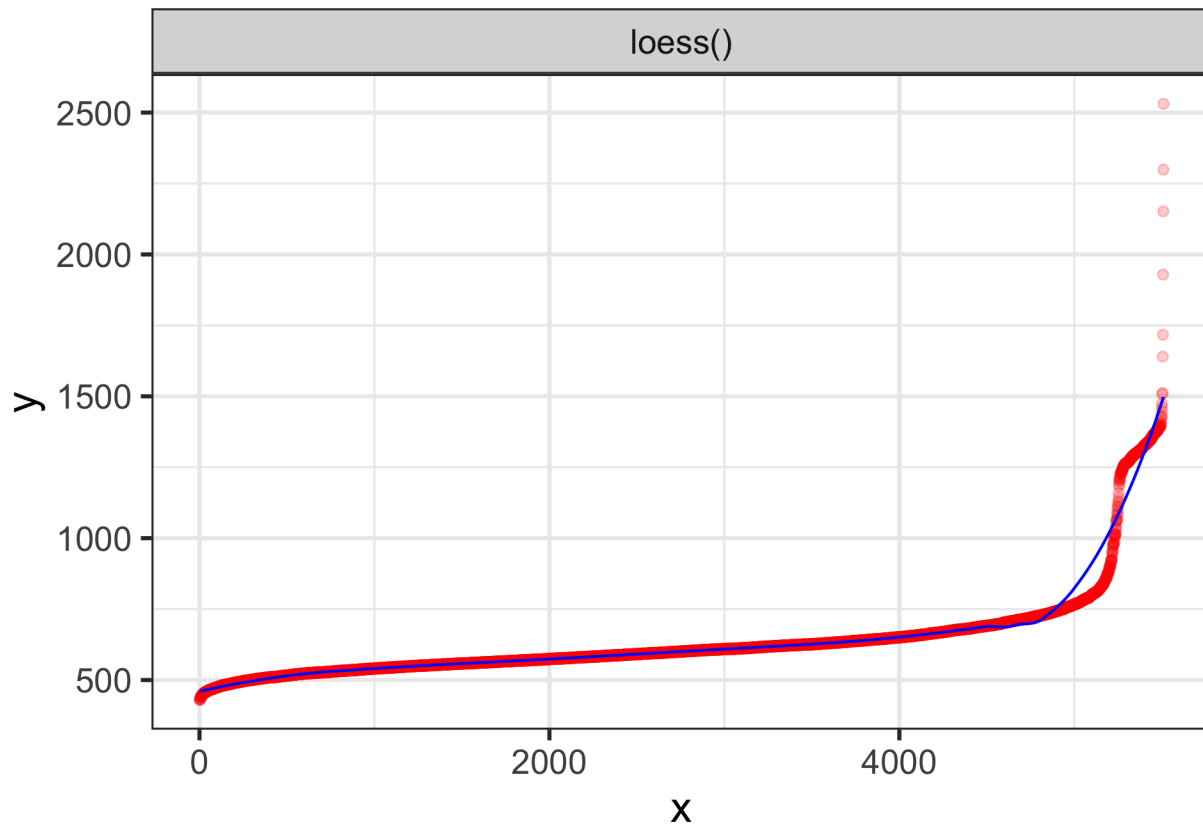
loessData_ch <- data.frame(
  x = 1:length(ch$co2),
  y = predict(loess(y~x, dat_ch, span = 0.1)),
  method = "loess()"
) %>%
  mutate(school = "Switzerland") #smooting the data using Loess

ggplot(loessData_ch, aes(x, y)) +
  geom_point(dat = dat_ch, aes(x, y), alpha = 0.2, col = "red") +
  geom_line(col = "blue") +
```

```
facet_wrap(~method) +  
theme_bw(16)
```



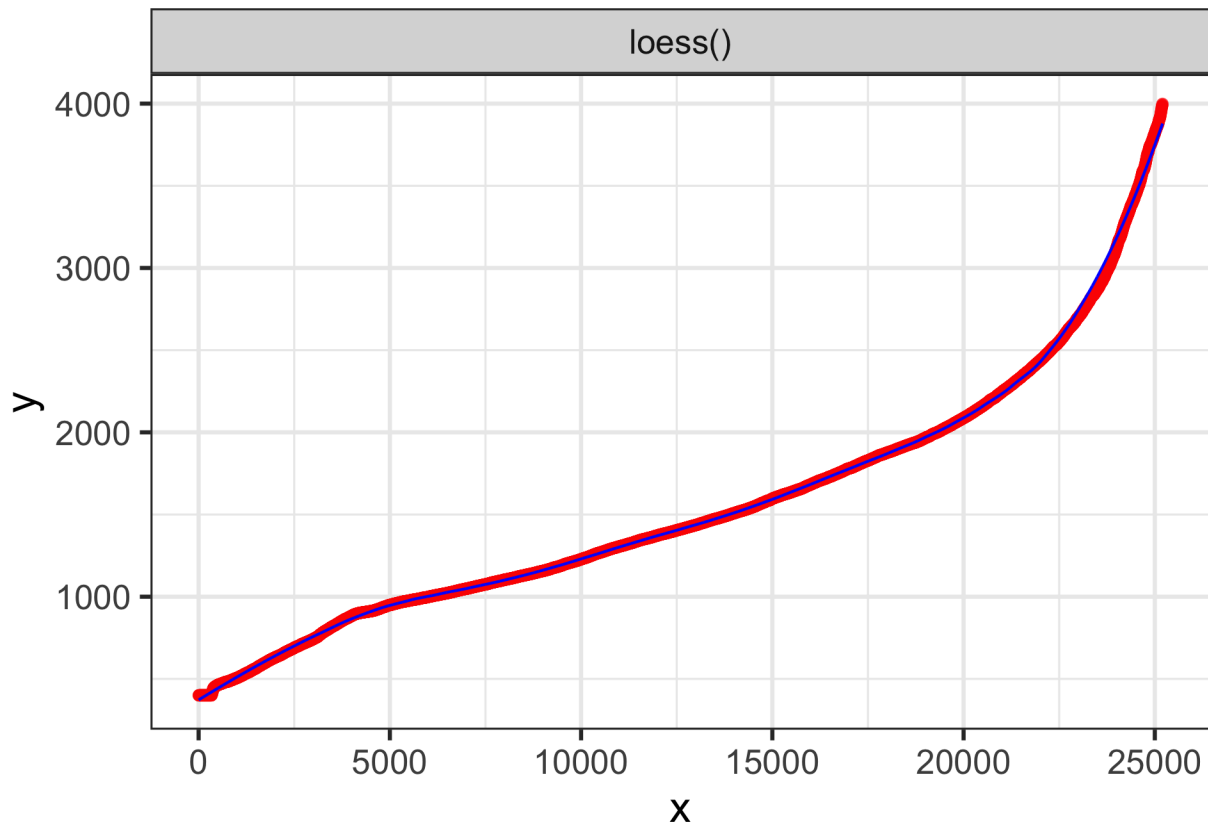
```
# TZ  
dat_tz <- data.frame(  
  x = 1:length(tz$co2),  
  y = tz$co2  
)  
  
loessData_tz <- data.frame(  
  x = 1:length(tz$co2),  
  y = predict(loess(y~x, dat_tz, span = 0.3)),  
  method = "loess()  
)  
  
ggplot(loessData_tz, aes(x, y)) +  
  geom_point(dat = dat_tz, aes(x, y), alpha = 0.2, col = "red") +  
  geom_line(col = "blue") +  
  facet_wrap(~method) +  
  theme_bw(16)
```



```
# SA
dat_sa <- data.frame(
  x = 1:length(sa$co2),
  y = sa$co2
)

loessData_sa <- data.frame(
  x = 1:length(sa$co2),
  y = predict(loess(y~x, dat_sa, span = 0.3)),
  method = "loess()"
)

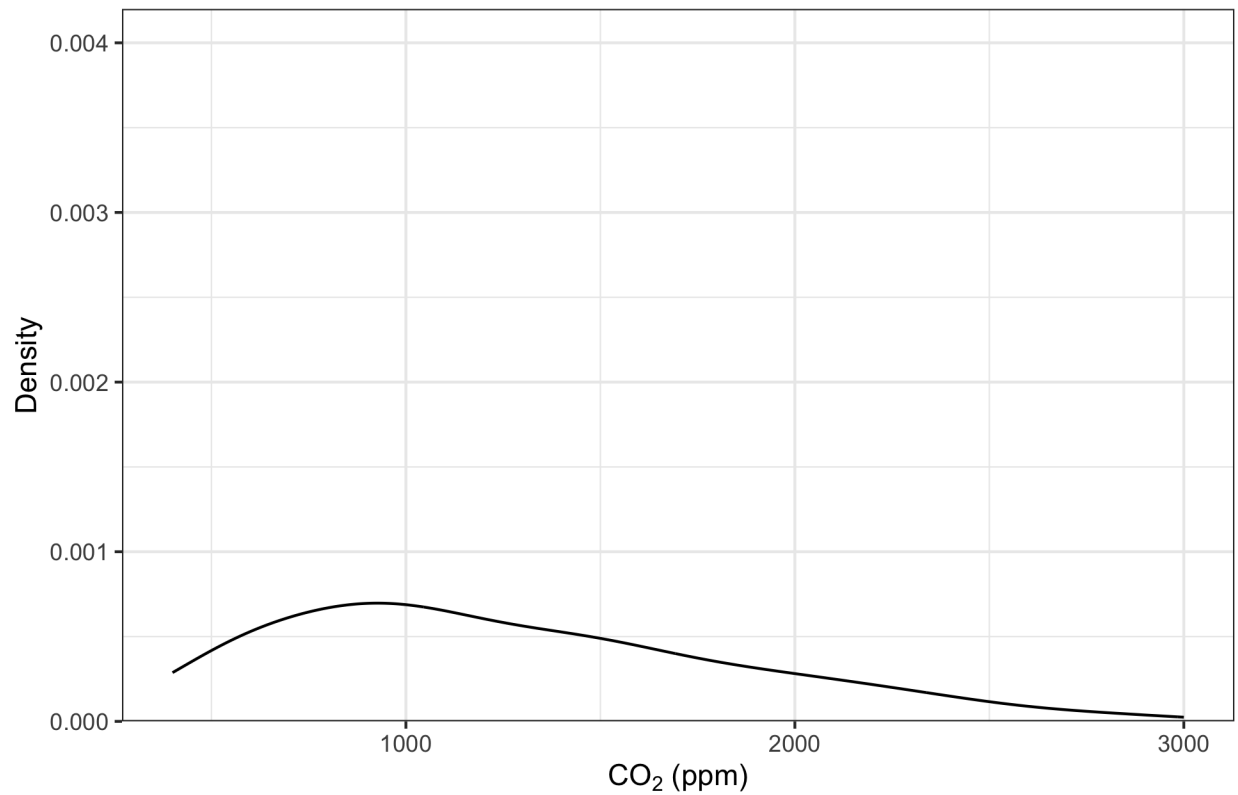
ggplot(loessData_sa, aes(x, y)) +
  geom_point(dat = dat_sa, aes(x, y), alpha = 0.2, col = "red") +
  geom_line(col = "blue") +
  facet_wrap(~method) +
  theme_bw(16)
```

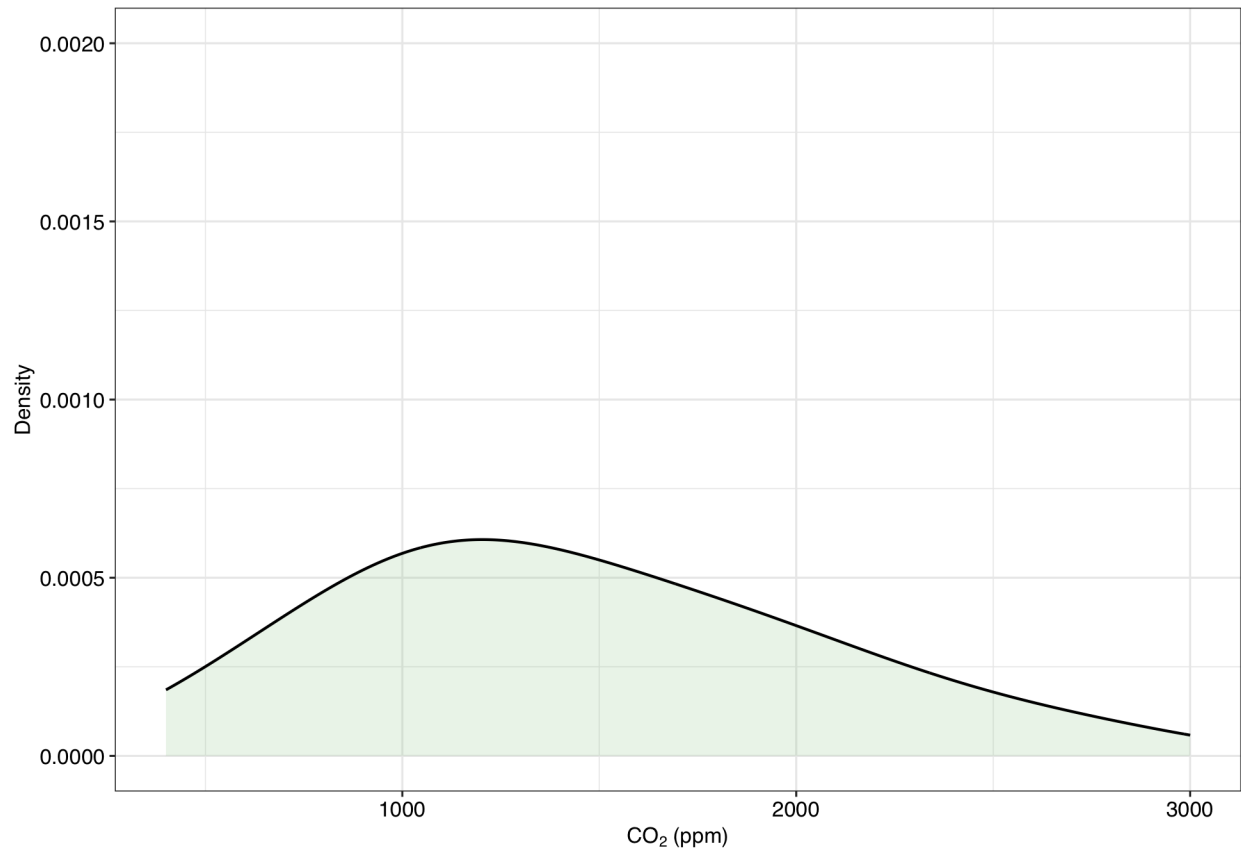
Co2 distribution of smoothed data

```
loessData_ch %>% #density plot ch
  ggplot(aes(x = y)) +
  geom_density(alpha = .2, kernel = "gaussian", adjust = 3) +
  scale_x_continuous(expand = c(0,0)) +
  scale_color_brewer(palette = "Set2") +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.05)), limits = c(0, 0.004)) +
  scale_x_continuous(limits = c(400, 3000)) +
  labs(x = expression(CO[2]*" (ppm)"), y = "Density") +
  theme_bw() +
  theme(legend.position = c(0.9,0.9), legend.title = element_blank()) +
  ggtitle("Probability distribution of observed Co2-values (Switzerland)")
```

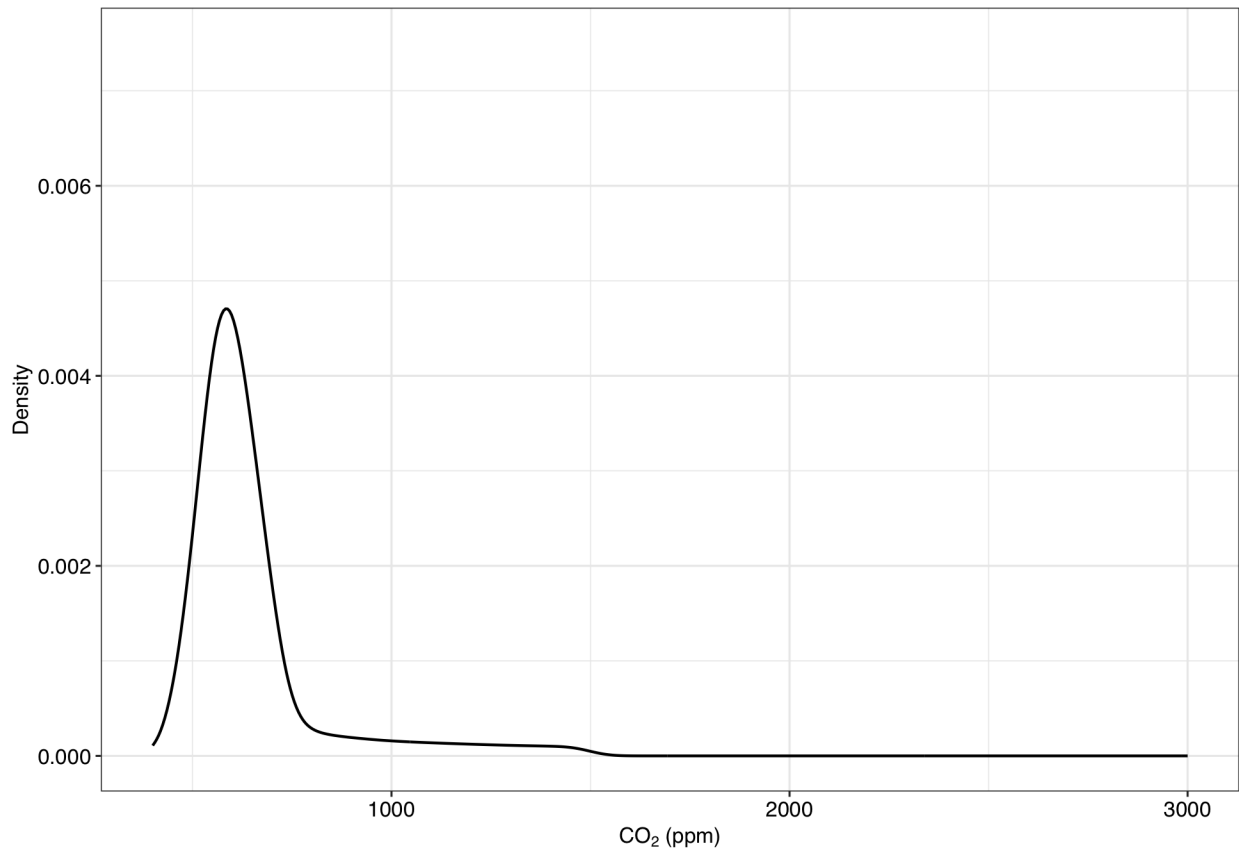
Probability distribution of observed Co2-values (Switzerland)



```
loessData_sa %>% #density plot sa smooth
  ggplot(aes(x = y)) +
  geom_density(alpha = .2, kernel = "gaussian", adjust = 4, fill = "darkseagreen3") +
  scale_x_continuous(expand = c(0,0)) +
  scale_color_brewer(palette = "Set2") +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(limits = c(0, 0.002)) +
  scale_x_continuous(limits = c(400, 3000)) +
  labs(x = expression(CO[2]*" (ppm)"), y = "Density") +
  theme(legend.position = "none") +
  theme_bw2()
```



```
loessData_tz %>% #density plot tz smooth
  ggplot(aes(x = y)) +
  geom_density(alpha = .2, kernel = "gaussian", adjust = 3.2) +
  scale_x_continuous(expand = c(0,0)) +
  scale_color_brewer(palette = "Set2") +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(limits = c(0, 0.0075)) +
  scale_x_continuous(limits = c(400, 3000)) +
  labs(x = expression(CO[2]*" (ppm)"), y = "Density") +
  theme_bw2() +
  theme(legend.position = c(0.9,0.9), legend.title = element_blank())
```



Modeling distribution for co2 and other CO2 parameters

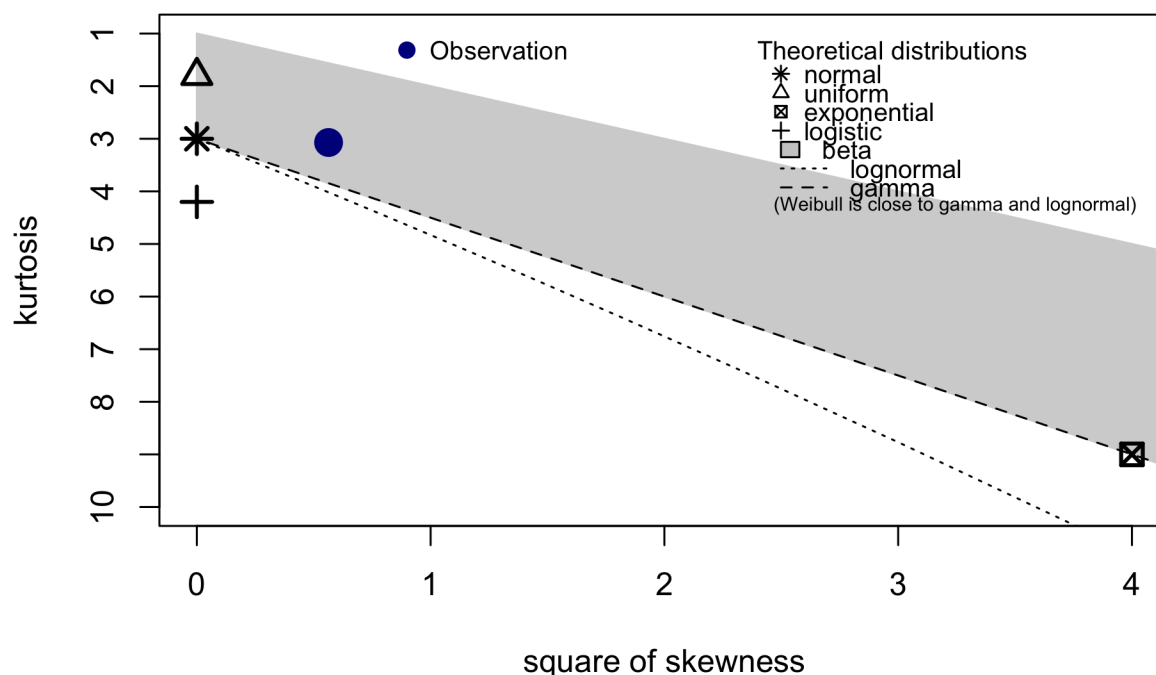
```
#C_a
C_a <- ((0.0042)*60)/8 #https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5666301/ mean girls (11-16 and 16-19)

#C_o
C_o <- 400 #p.p.m (taking a higher estimate because higher values ar possible when a lot of traffic ect
## all schools are directly on the side of a road (no info for tanzania), so i won't make a distinction

#Switzerland
x_ch <- loessData_ch %>%
  pull(y) #I now use the smoothed co2 data to fit a gamma distribution

descdist(x_ch, discrete = FALSE) #gamma distribution fits well
```

Cullen and Frey graph



```
## summary statistics
## -----
## min: 384.3828 max: 3262.327
## median: 1158.84
## mean: 1279.487
## estimated sd: 603.5674
## estimated skewness: 0.7506849
## estimated kurtosis: 3.071178
```

```
fitdistr(x_ch, "gamma") #get parameters for the gamma distribution
```

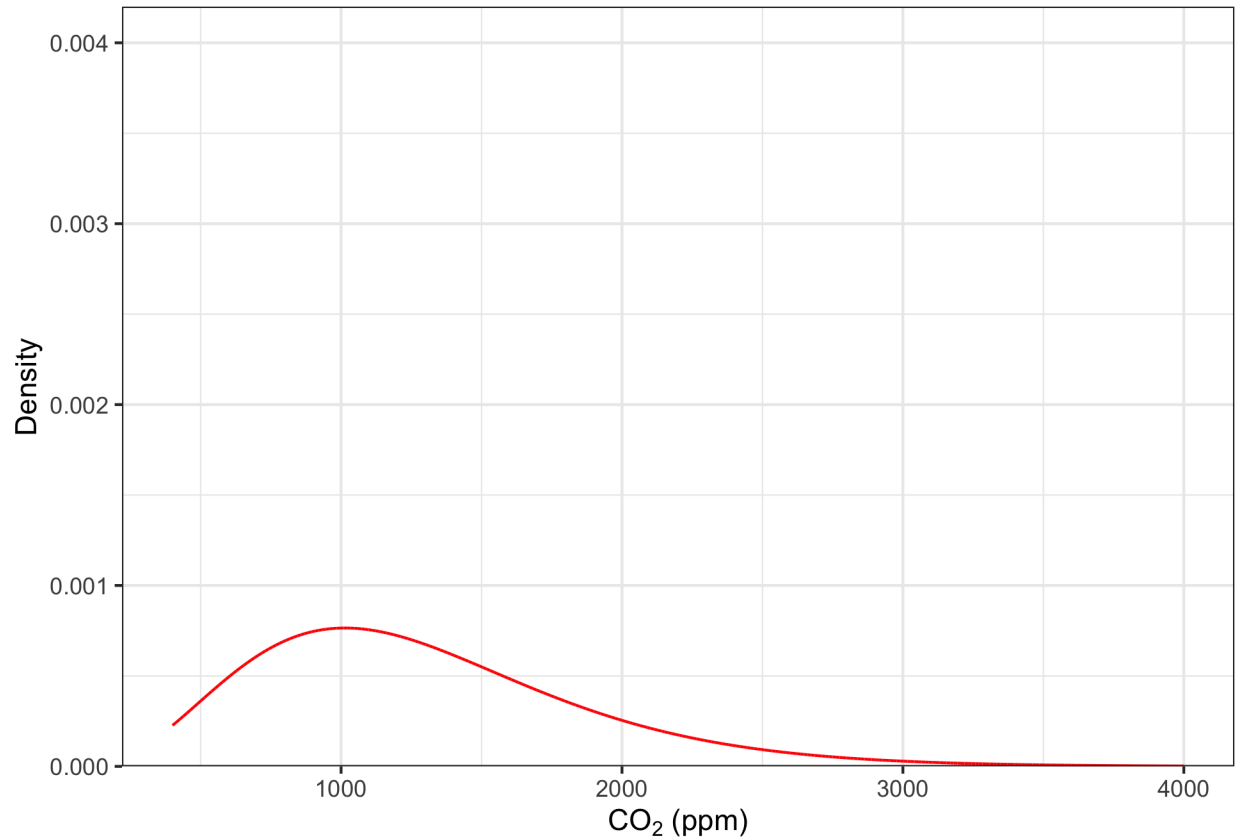
```
##          shape          rate
## 4.54709464550 0.00355384185
## (0.02362341620) (0.00001873289)
```

```
x <- seq(400, 4000, by = .1) #vector for the plot
```

```
co2_distr_ch <- data.frame(co2 = seq(400, 4000, .1)) %>%
  mutate(prob = dtrunc(co2, spec = "gamma", a = 400, b = 4000, shape = 4.751, rate = 0.0037)) %>%
  mutate(country = "Switzerland")
```

```
co2_distr_ch %>% #plot for the fitted distribution (not used in the final paper)
  ggplot(aes(x=co2,y=prob)) +
  geom_line(color= "red") +
```

```
labs(x = expression(CO[2]*" (ppm)"), y = "Density") +
theme_bw() +
scale_y_continuous(expand = expansion(mult = c(0, 0.05)), limits = c(0, 0.004)) +
scale_x_continuous(limits = c(400, 4000))
```

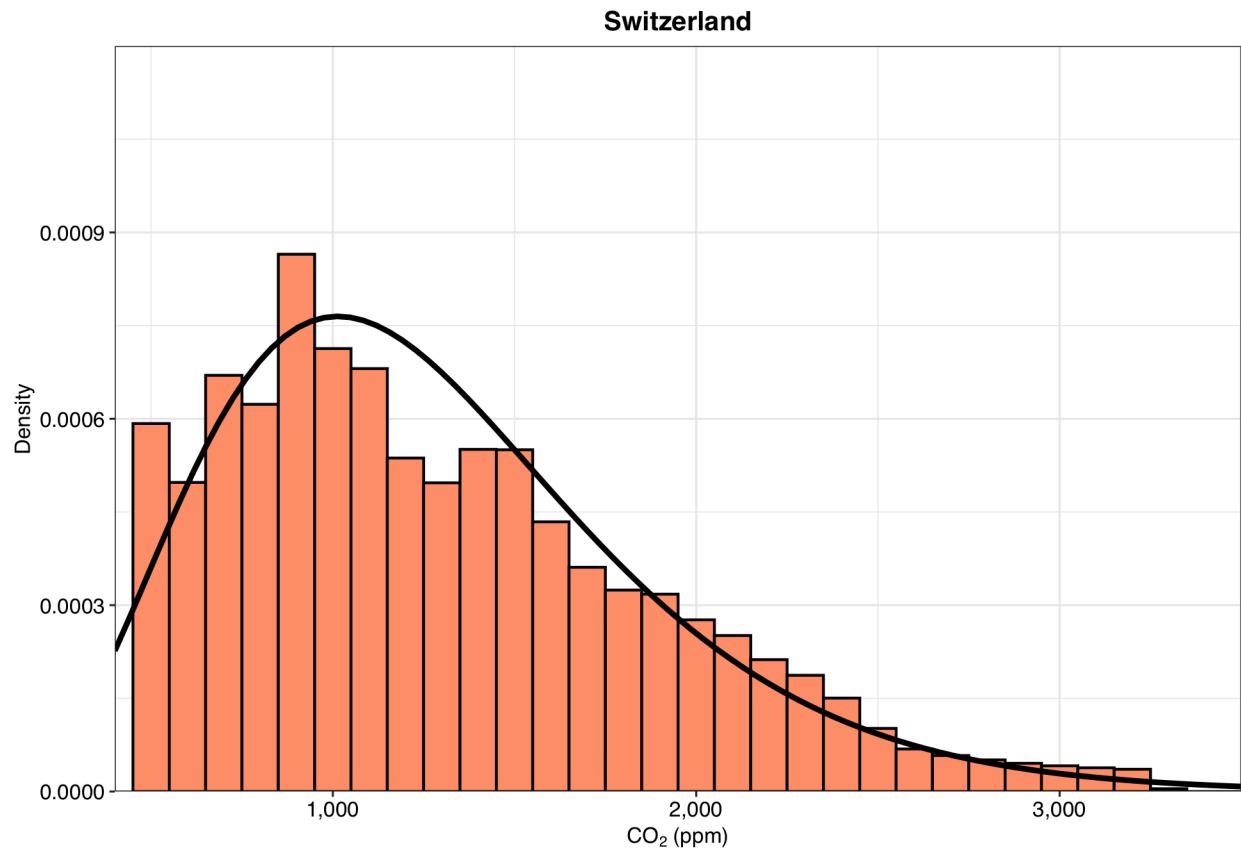


```
sample_co2_ch <- sample(co2_distr_ch$co2, 3000, replace = TRUE, prob = co2_distr_ch$prob) #sample co2 f
sample_f_ch <- tibble(co2 = sample_co2_ch, f = ((co2-C_o)/C_a)/1000000) %>% #sample f for dataframe lat
  dplyr::select(-co2)
```

Plot for paper

```
plot_co2_ch <- loessData_ch %>%
  ggplot(aes(x = y)) +
  geom_histogram(binwidth = 100, color = "black", fill = "#FFA07A", aes(y = ..density..)) +
  scale_y_continuous(limits = c(0, 0.0012), expand = c(0,0)) +
  scale_x_continuous(limits = c(400, 4000), expand = c(0,0), labels = scales::comma) +
  labs(x = expression(CO[2]*" (ppm)"), y = "Density") +
  coord_cartesian(xlim = c(400,3500))+
  theme_bw2()+
  stat_function(fun = function(x) dtrunc(x, spec = "gamma", a = 400, b = 4000, shape = 4.751, rate = 0.000001))
  ggtitle("Switzerland") +
  theme(plot.title = element_text(hjust = 0.5))

plot_co2_ch
```

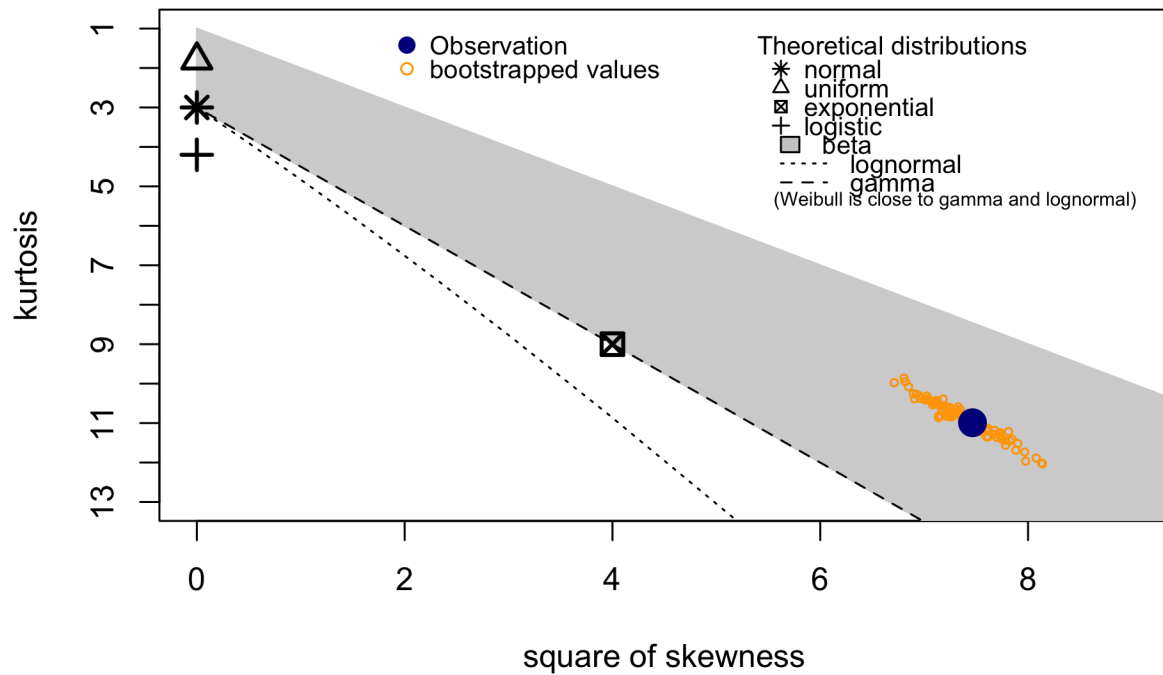


```
#tanzania
x_tz <- loessData_tz %>%
  pull(y)

x_tz <- as.numeric(x_tz)

descdist(x_tz, discrete = FALSE, boot = 100) #gamma fits ok
```

Cullen and Frey graph



```
## summary statistics
## -----
## min: 460.1963 max: 1497.719
## median: 600.4938
## mean: 644.2189
## estimated sd: 173.3067
## estimated skewness: 2.732294
## estimated kurtosis: 10.99144
```

```
fitdistr(x_tz, "t") #get parameters
```

```
##          m          s          df
## 593.94973711 54.89374964 1.55018887
## ( 1.02944941) ( 1.01107831) ( 0.04266281)
```

```
co2_distr_tz <- data.frame(co2 = seq(400, 4000, .1)) %>%
  mutate(prob = dtrunc(x, spec = "st", a = 400, b = 4000, mu = 593.95, sigma = 80, nu = 1.55)) %>% mutate

plot_co2_tz <- co2_distr_tz %>%
  ggplot(aes(x=x,y=prob)) +
  geom_line(color= "red") +
  labs(x = expression(CO[2]*" (ppm)"), y = "Density") +
  ggtitle("Tanzania") +
  theme_bw2() +
```



```

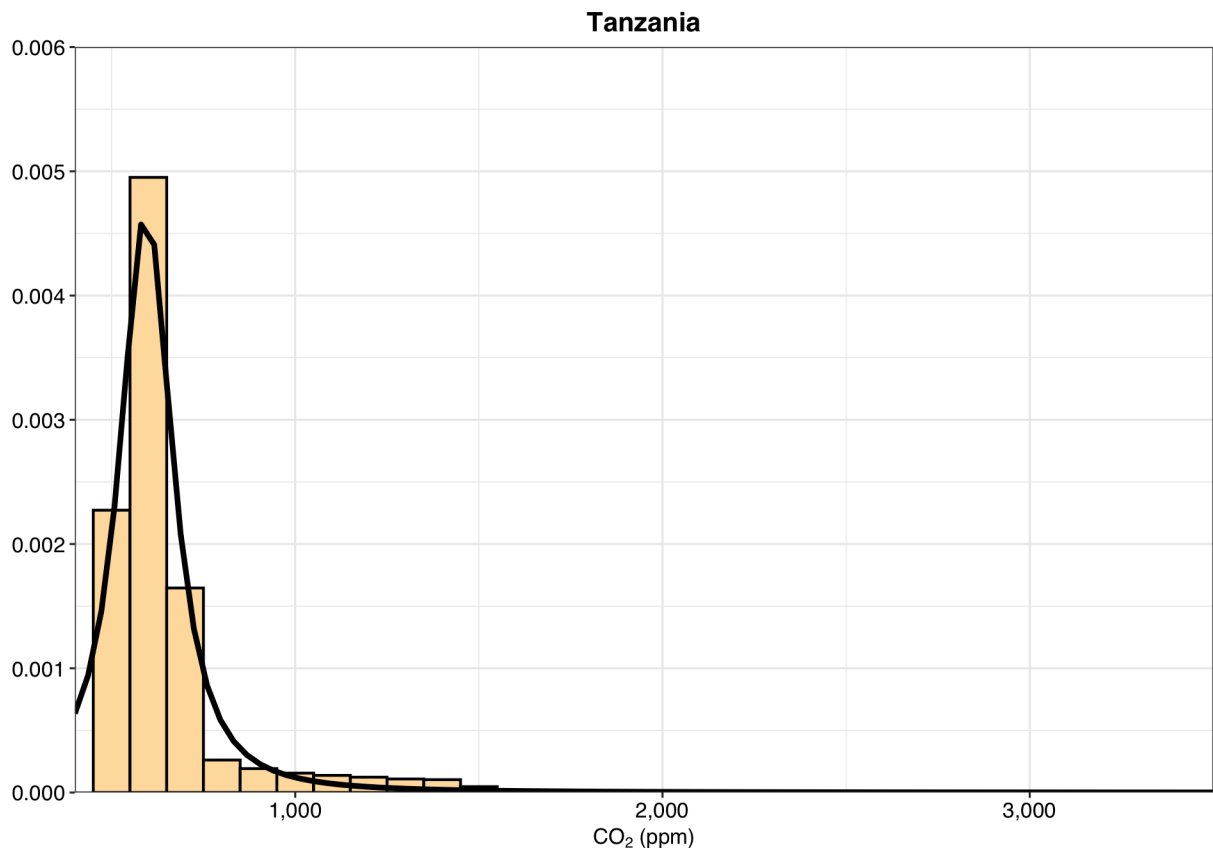
scale_y_continuous(limits = c(0, 0.0075)) +
scale_x_continuous(limits = c(400, 3000))

sample_co2_tz <- sample(co2_distr_tz$co2, 3000, replace = TRUE, prob = co2_distr_tz$prob) #sample
sample_f_tz <- tibble(co2 = sample_co2_tz, f = ((co2-C_o)/C_a)/1000000) %>% #sample f for df_tz
  dplyr::select(-co2)

####histogram / density comparison (plot in supplements)
plot_co2_tz <- loessData_tz %>%
  ggplot(aes(x = y)) +
  geom_histogram(binwidth = 100, color = "black", fill = "#FFDEAD", aes(y = ..density..)) +
  scale_y_continuous(limits = c(0, 0.006), expand = c(0,0)) +
  scale_x_continuous(limits = c(400, 4000), expand = c(0,0), labels = scales::comma) +
  labs(x = expression(CO[2]*" (ppm)"), y = "") +
  coord_cartesian(xlim = c(400,3500))+
  theme_bw2() +
  stat_function(fun = function(x) dtrunc(x, spec = "st", a = 400, b = 4000, mu = 593.95, sigma = 80, nu
  ggtitle("Tanzania") +
  theme(plot.title = element_text(hjust = 0.5))

plot_co2_tz

```



```

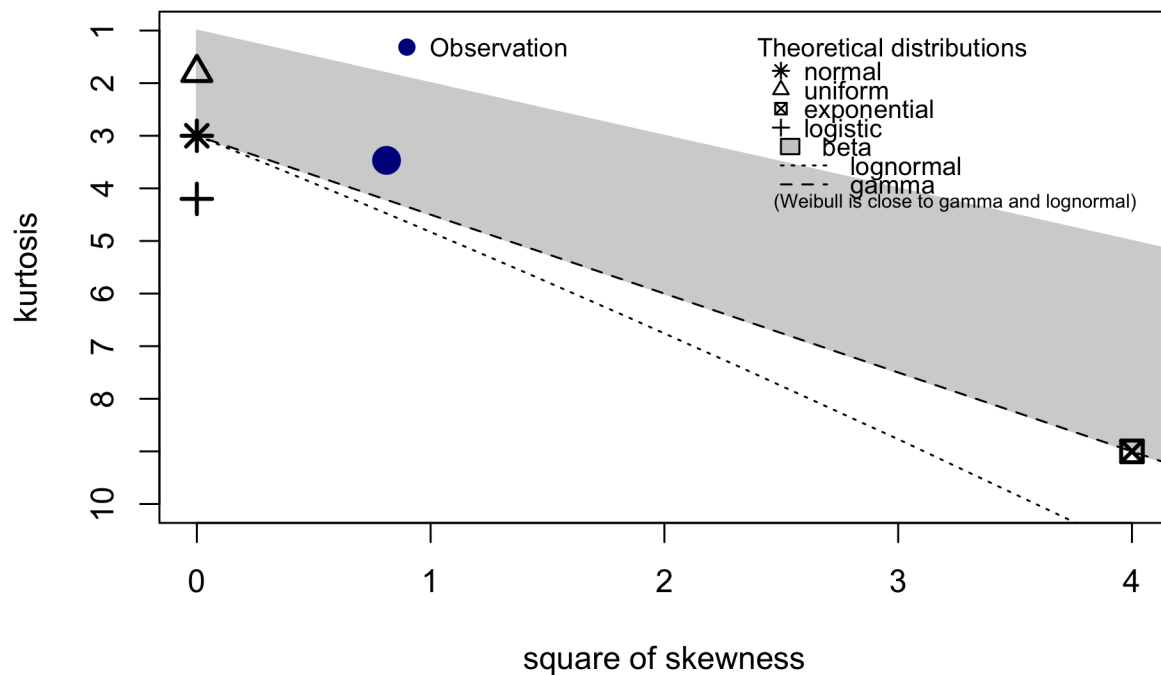
#south africa
x_sa <- loessData_sa %>%
  pull(y)

```

```
x_sa <- as.numeric(x_sa)
```

```
descdist(x_sa, discrete = FALSE) #normal/gamma fits ok -> after comparison --> gamma is better
```

Cullen and Frey graph



```
## summary statistics
## -----
## min: 372.0888 max: 3878.796
## median: 1411.002
## mean: 1557.554
## estimated sd: 754.5649
## estimated skewness: 0.9008012
## estimated kurtosis: 3.467689
```

```
fitdistr(x_sa, "gamma") #get parameters
```

```
##          shape          rate
## 4.37959821196 0.00281184362
## (0.02884306566) (0.00001833917)
```

```
fitdistr(x_sa, "weibull") #get parameters
```

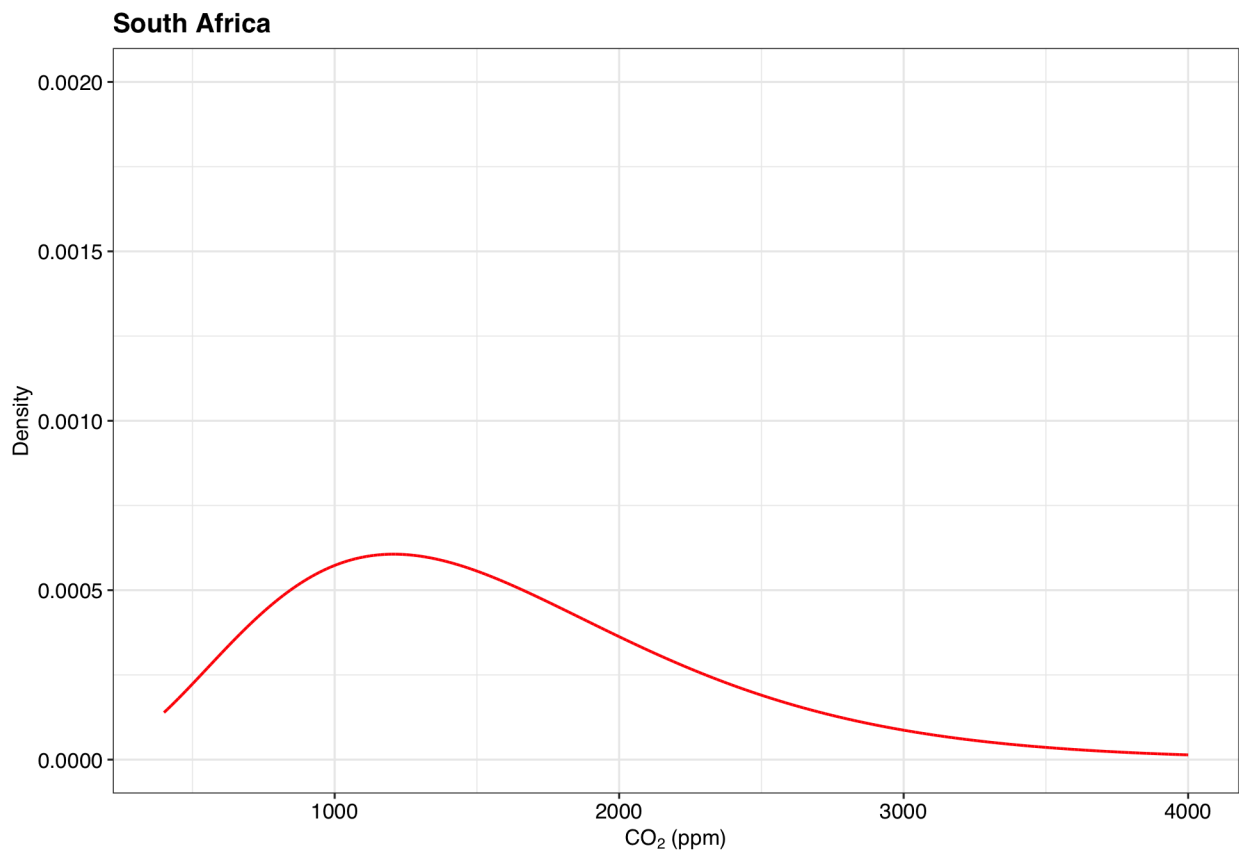
```
##          shape          scale
## 2.20053383 1764.14930033
## ( 0.01036866) ( 5.26269485)
```

```
fitdistr(x_sa, "lognormal") #get parameters
```

```
##      meanlog      sdlog  
## 7.232383759 0.499031464  
## (0.003143478) (0.002222775)
```

```
co2_distr_sa <- data.frame(co2 = seq(400, 4000, .1)) %>%  
  mutate(prob = dtrunc(x, spec = "gamma", a = 400, b = 4000, shape = 4.38, rate = 0.0028)) %>%  
  mutate(country = "South Africa")
```

```
co2_distr_sa %>%  
  ggplot(aes(x=x, y=prob)) +  
  geom_line(color = "red") +  
  labs(x = expression(CO[2]*" (ppm)"), y = "Density") +  
  ggtitle("South Africa") +  
  theme_bw2() +  
  scale_y_continuous(limits = c(0, 0.002)) +  
  scale_x_continuous(limits = c(400, 4000))
```



```
sample_co2_sa <- sample(co2_distr_sa$co2, 3000, replace = TRUE, prob = co2_distr_sa$prob) #sample  
sample_f_sa <- tibble(co2 = sample_co2_sa, f = ((co2-C_o)/C_a)/1000000) %>% #sample f for df_sa  
  dplyr::select(-co2)
```

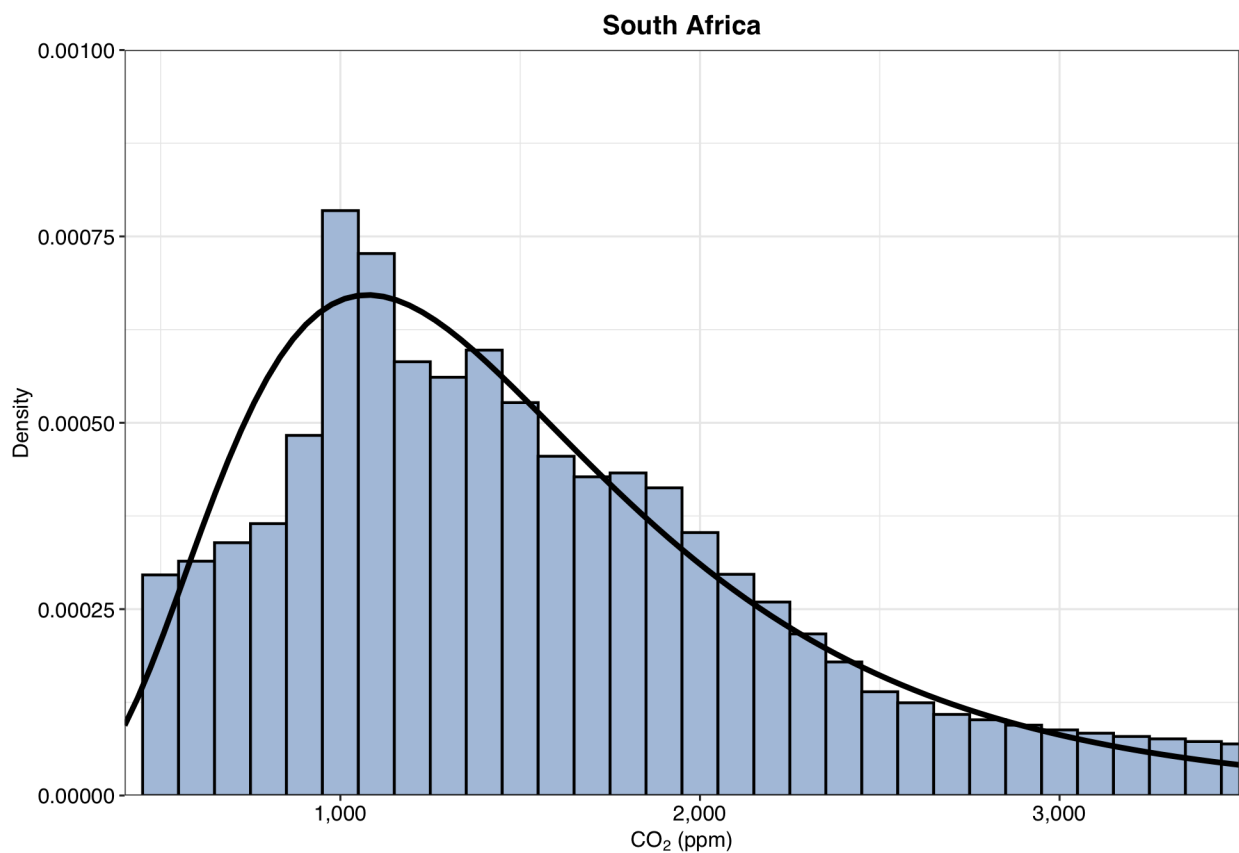
```
#plot
```

```

plot_co2_sa <- loessData_sa %>%
  ggplot(aes(x = y)) +
  geom_histogram(binwidth = 100, color = "black", fill = "#B0C4DE", aes(y = ..density..)) +
  scale_y_continuous(limits = c(0, 0.001), expand = c(0,0)) +
  scale_x_continuous(limits = c(400, 4000), expand = c(0,0), labels = scales::comma) +
  labs(x = expression(CO2" (ppm)"), y = "Density") +
  coord_cartesian(xlim = c(400,3500))+
  theme_bw2()+
  stat_function(fun = function(x) dtrunc(x, spec = "lnorm", a = 400, b = 4000, meanlog = 7.23, sdlog = 0.5))
  ggtitle("South Africa") +
  theme(plot.title = element_text(hjust = 0.5))

```

plot_co2_sa



#Plot for the paper

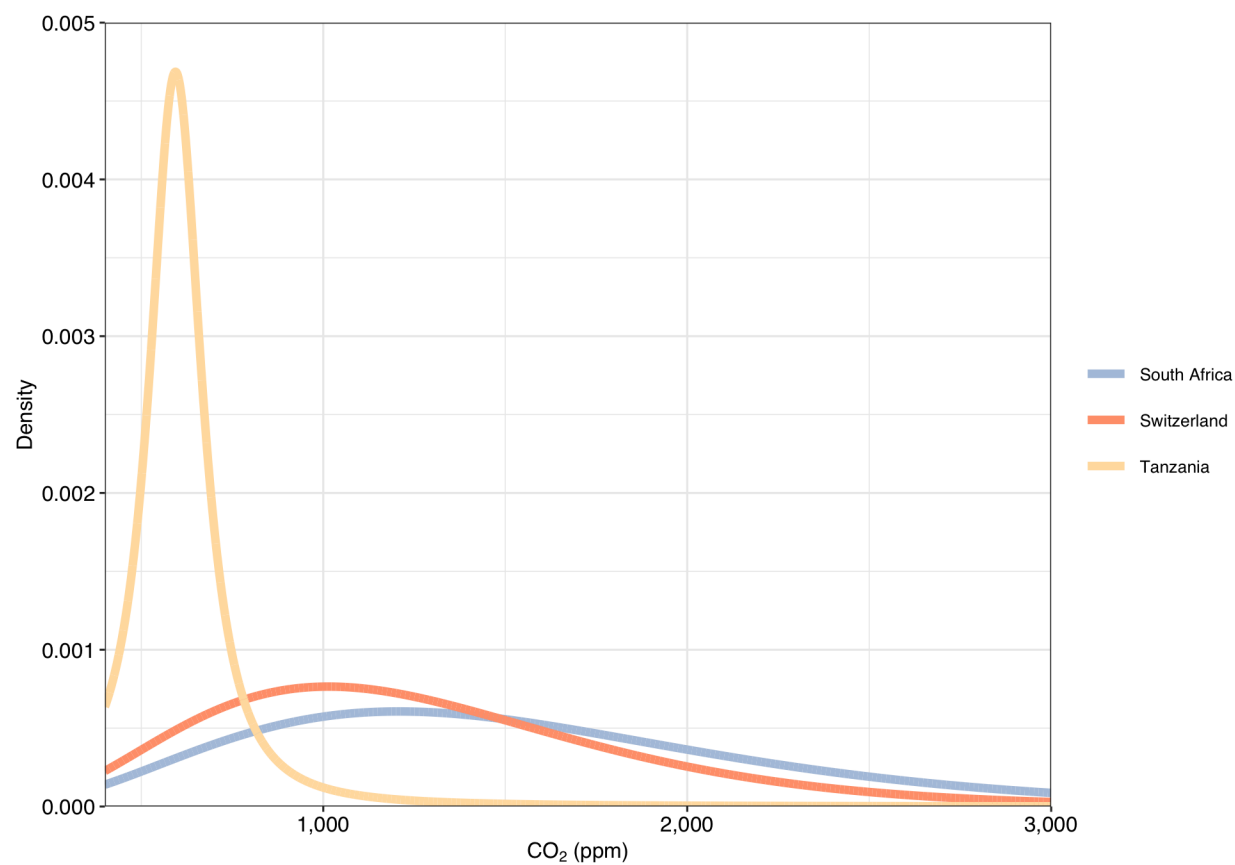
```

co2_data <- rbind(co2_distr_sa,co2_distr_ch,co2_distr_tz)

co2_data %>% #Comparison of distributions
  ggplot(aes(x=co2, y = prob, color = country)) +
  geom_line(size = 1.5) +
  labs(x = expression(CO2" (ppm)"), y = "Density") +
  theme_bw2() +
  scale_y_continuous(limits = c(0, 0.005), expand = c(0,0)) +
  scale_x_continuous(limits = c(400, 3000), expand = c(0,0), labels = scales::comma) +

```

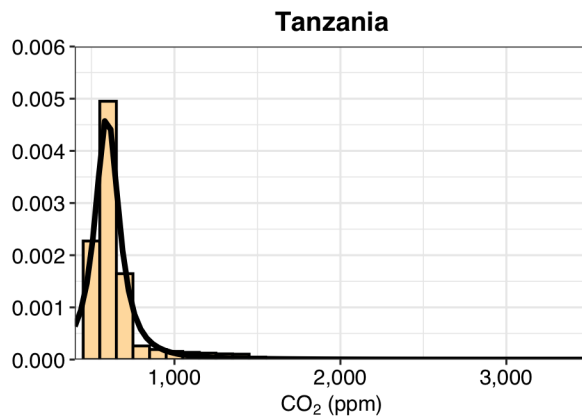
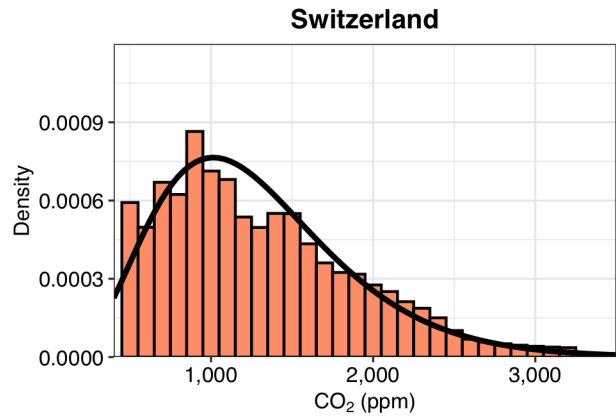
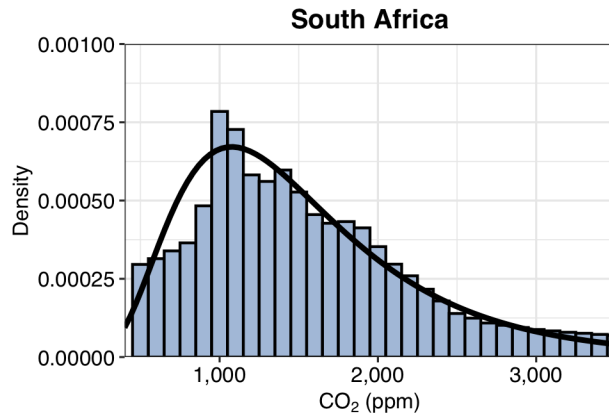
```
scale_color_manual(values = c("#BOC4DE", "#FFA07A", "#FFDEAD"), name = NULL)
```



```
text_size = 8
```

```
#Plot for supplements
```

```
plot_grid(plot_co2_sa, plot_co2_ch, plot_co2_tz, ncol = 2) #plot in the paper
```



Quanta

I'll use the following studies for calculating the mean parameter:

Riley (1962): 130 patients, q : 1.25 Escombe (2008): 117 patients, q : 8.2 Nardell (1991) : 1 patients, q : 12.5
 Andrews (2014) : 571 patients, q : 0.89

```
#Calculate the weighted mean (number of persons in the study as weighting) to get a guideline in which
q <- (1.25*130+8.2*117+12.5+0.89*571)/(130+117+1+571) #weighted mean from different studies

#Consider Table 2 from Escombe as it reports quanta estimates for individuals. I use the columns "number
mean_one_inf <- mean(c(12,3,5.5,1.8,18,12)) #mean quanta of pers. which infected one pig
mean_two_inf <- mean(c(2.9,40)) #mean quanta of pers. which infected two pigs
q_inf_persons <- c(12,3,2.9,5.5,1.8,18,40,12,226,52,mean_two_inf,rep(mean_one_inf,11)) #In the table tw

#I now assume that the rest of the persons who have infected pigs have only infected one pig each. Ther
#Vector with the captured + the rest I model with truncated normal distribution (M=1,SD=1) --> paramet
q_sample_total_norm <- c(q_inf_persons, rtruncnorm(117-length(q_inf_persons), a = 0, mean = 1, sd = 1))

#I fit a student-t-distribution on this vector (q_sample_total_norm) to get the parameters.
fitdistr(q_sample_total_norm, "t") #get parameters
```

```
##          m          s          df
##  1.29086768  0.54606585  0.82409533
## (0.07835933) (0.08135344) (0.11735146)
```

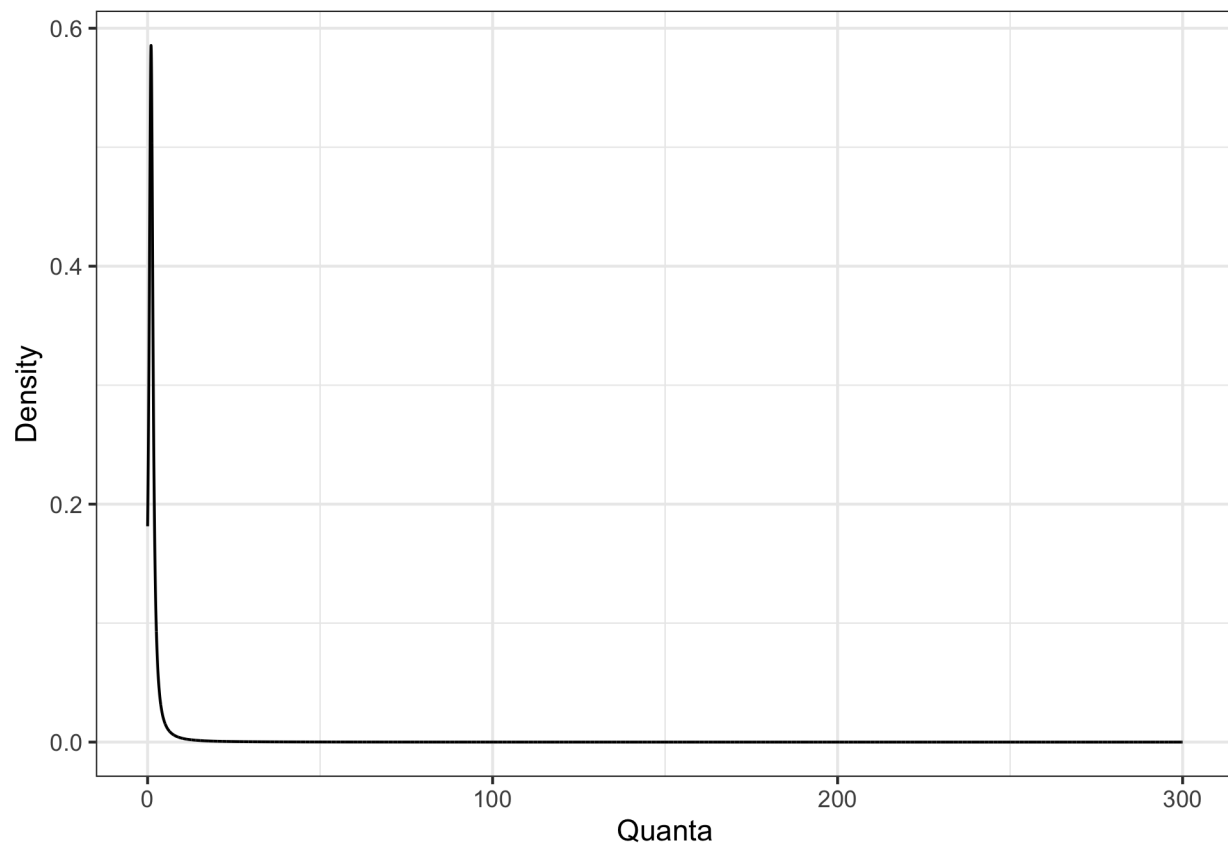
```

#Now I still adjust minimally.
dq <- function(x) {
  dtrunc(x, spec = "st", a = 0, b = 300, mu = 1, sigma = 0.67, nu = 1) #parameters from fitdistr
}

#dataframe, with density for each "point" --> mathematically not perfect but okay for the plot
rq_distr <- data.frame(quanta = seq(0, 300, .01)) %>%
  mutate(prob = dq(quanta)) %>%
  mutate(type = "Mtb")

rq_distr %>% #plot for quanta distribution
  ggplot(aes(x = quanta, y = prob)) +
  geom_line()+
  theme_bw() +
  xlab("Quanta") +
  ylab("Density")

```



```

sample_q <- sample(rq_distr$quanta, 3000, replace = TRUE, prob = rq_distr$prob) #sample q for dataframe

tb_sample <- tibble(sample = sample_q) %>%
  count(sample > 10) %>%
  mutate(prob = n/3000) #what is the ratio of datapoints over 10?

q_under_10 <- tb_sample[1,3, drop = TRUE]
q_under_10 # approx. 96% of data points are below 10 --> good

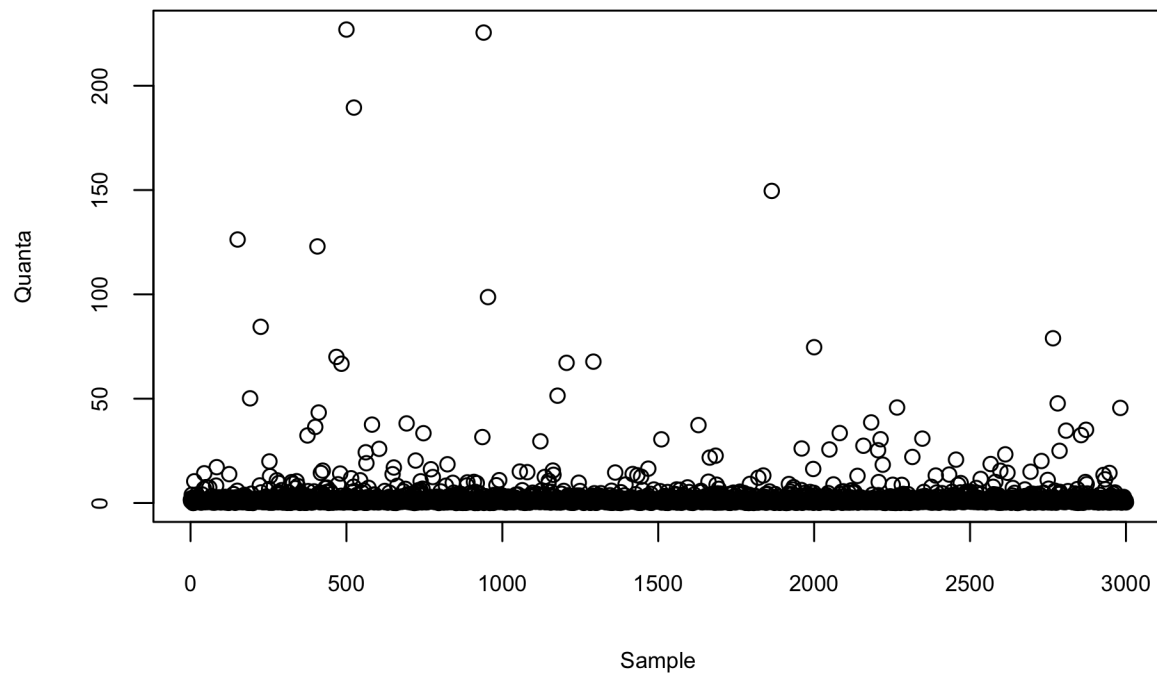
```

```
## [1] 0.9653333
```

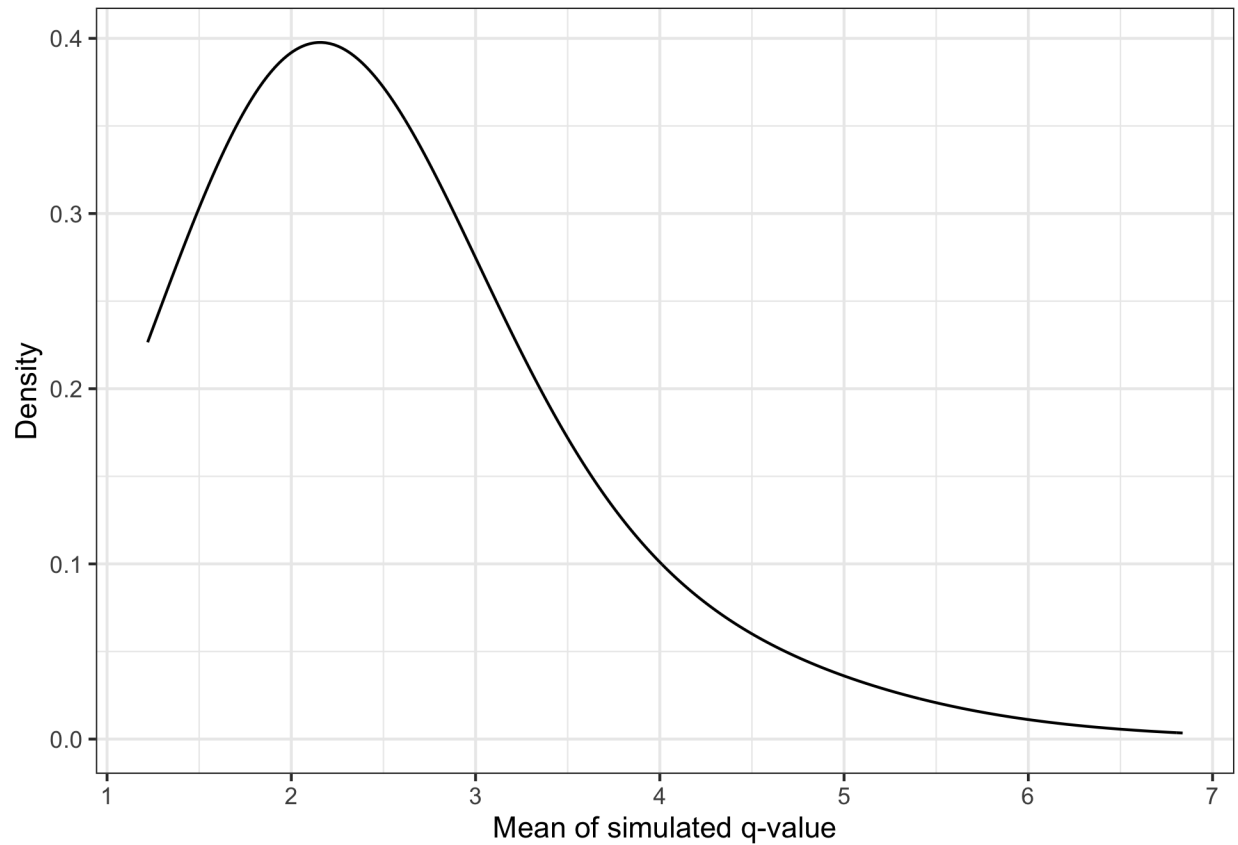
```
mean(sample_q) #mean of the sample should be between 2 and 3 --> good
```

```
## [1] 2.664323
```

```
plot(sample_q, ylab="Quanta", xlab = "Sample", cex.axis=.7, cex.lab=.7)
```



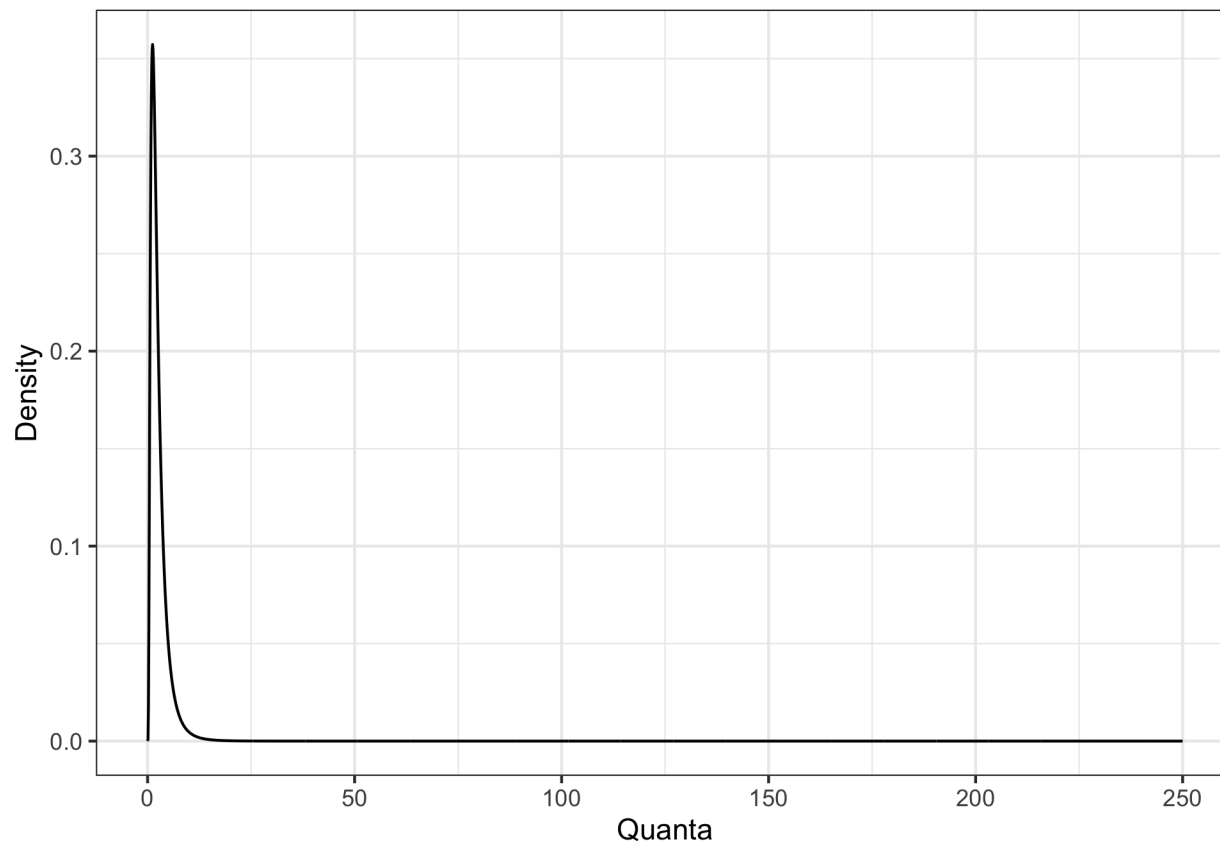
```
thousand_sample <- replicate(1000, mean(sample(rq_distr$quanta, 100, replace = TRUE, prob = rq_distr$pr  
#for calculating the distribution of the mean over lots of simulations  
ggplot() + geom_density(mapping = aes(thousand_sample), alpha = .2, kernel = "gaussian", adjust = 5) +  
  xlab("Mean of simulated q-value") +  
  ylab("Density") +  
  theme_bw()
```

```
## Buonanno et al. provide an estimation for the quanta (also used in MCID paper)
#"light activity, speaking"
cov_q <- function(x) {
  dtrunc(x, spec = "lnorm", a = 0, meanlog = 0.698, sdlog = 0.720) #defining function
}

cov_q_distr <- data.frame(quanta = seq(0, 250, .1)) %>% #density for each datapoint (mathematically not
  mutate(prob = cov_q(quanta)) %>%
  mutate(type = "SARS-CoV-2")

cov_q_distr %>% #plot of the distribution
  ggplot(aes(x = quanta, y = prob)) +
  geom_line() +
  theme_bw() +
  ylab("Density") +
  xlab("Quanta")
```



```
sample_q_cov <- sample(cov_q_distr$quanta, 3000, replace = TRUE, prob = cov_q_distr$prob) #sample for d
```

```
q_data <- rbind(rq_distr,cov_q_distr)
```

```
#calculate expected value to integrate in the plot
```

```
expected_q_TB <- integrate(function(x) x * dq(x), lower = 0, upper = 300)$value #e
```

```
expected_q_SARS <- integrate(function(x) x * cov_q(x), lower = 0, upper = 300)$value
```

```
#calculate sd for the paper
```

```
variance_q_TB <- integrate(function(x) (x - expected_q_TB)^2 * dq(x), lower = 0, upper = 300)$value
```

```
sd_q_TB <- sqrt(variance_q_TB)
```

```
variance_q_SARS <- integrate(function(x) (x - expected_q_SARS)^2 * cov_q(x), lower = 0, upper = 300)$value
```

```
sd_q_SARS <- sqrt(variance_q_SARS)
```

```
q_data %>% #plot quanta distribution for both (Supplements)
```

```
ggplot(aes(x=quanta, y = prob, color = type)) +
```

```
geom_line(width = 1) +
```

```
labs(x = "Quanta (q/h)", y = "Density") +
```

```
scale_y_continuous(limits = c(0, 0.700), expand= c(0,0)) +
```

```
scale_x_continuous(limits = c(0, 30), expand = c(0,0)) +
```

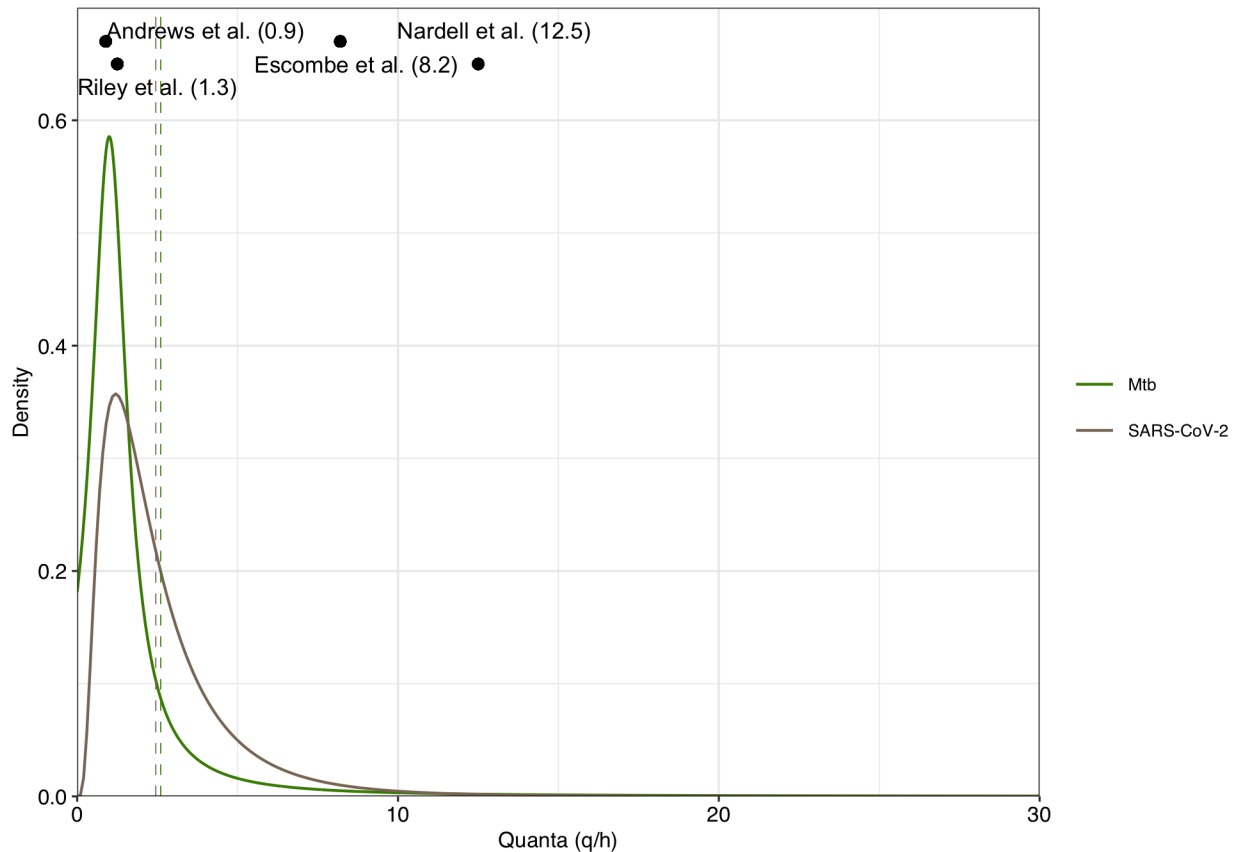
```
theme_bw2() +
```

```
geom_point(aes(x=1.25, y= 0.65), color = "black") +
```

```
geom_point(aes(x=12.5, y= 0.65), color = "black") +
```

```
geom_point(aes(x=8.2, y= 0.67), color = "black") +
```

```
geom_point(aes(x=0.89, y= 0.67), color = "black") +
  annotate("text", x=2.5, y=0.63, label="Riley et al. (1.3)", size = 3) +
  annotate("text", x=13, y=0.68, label="Nardell et al. (12.5)", size = 3) +
  annotate("text", x=8.7, y=0.65, label="Escombe et al. (8.2)", size = 3) +
  annotate("text", x=4, y= 0.68, label="Andrews et al. (0.9)", size = 3) +
  geom_vline(xintercept = expected_q_TB, linetype = "dashed", linewidth = 0.2, color = "peachpuff4") +
  geom_vline(xintercept = expected_q_SARS, linetype = "dashed", linewidth = 0.2, color = "chartreuse4") +
  scale_color_manual(values = c("chartreuse4", "peachpuff4"), name = NULL)
```



Rest of the parameters

```
#n
n_ch <- 20
n_sa <- 30 #Powerpoint
n_tz <- 50 #Powerpoint

#t
month <- 7*5*4
year <- 7*5*4*10

#I tuberculosis
inc_ch <- 4.1/100000
#https://www.bag.admin.ch/bag/de/home/zahlen-und-statistiken/zahlen-zu-infektionskrankheiten.exturl.htm

inc_sa <- 513/100000
# tendenziell zu hoch da 15-25
```

```

#https://worldhealthorg.shinyapps.io/tb_profiles/?_inputs_entity_type=%22country%22&lan=%22EN%22&iso2
inc_tz <- 208/100000
#https://data.worldbank.org/indicator/SH.TBS.INCD?locations=TZ

I_ch <- inc_ch*n_ch #prevalence per class (per year)
I_sa <- inc_sa*n_sa
I_tz <- inc_tz*n_tz

# Fixing prevalence for the second analysis
I_ch_fix <- inc_sa*n_ch
I_sa_fix <- inc_sa*n_sa
I_tz_fix <- inc_sa*n_tz

# I SARS
## SARS Estimate based on excess mortality

excess <- read.csv("../data-raw/excess.CSV") #https://ghdx.healthdata.org/record/ihme-data/covid_19_ex
IFR <- 0.0068 #IFR: https://doi.org/10.1016/j.ijid.2020.09.1464
IFR_SD <- (0.0082-0.0053)/3.92 #https://handbook-5-1.cochrane.org/chapter_7/7_7_7_2_obtaining_standard_
sim_IFR <- rnorm(3000, IFR, IFR_SD) #I estimate a normal distribution from the published IFR which accou

excess_mortality <- excess %>%
  filter(location_name %in% c("Switzerland", "Tanzania", "South Africa")) %>%
  rename(country = location_name) %>%
  filter(measure_name %in% c("excess_death_rate"))
  #https://handbook-5-1.cochrane.org/chapter_7/7_7_3_3_obtaining_standard_deviations_from_standard_err

#Switzerland
mean_ex_ch <- excess_mortality[excess_mortality$country == "Switzerland", "mean_value"]
sd_ex_ch <- (excess_mortality[excess_mortality$country == "Switzerland", "upper_ci"]-excess_mortality[exce

sim_excess_ch <- rnorm(3000, mean_ex_ch, sd_ex_ch) #form a normal distribution from the data for simula

#Tanzania
mean_ex_tz <- excess_mortality[excess_mortality$country == "Tanzania", "mean_value"]
sd_ex_tz <- (excess_mortality[excess_mortality$country == "Tanzania", "upper_ci"]-excess_mortality[exce

sim_excess_tz <- rnorm(3000, mean_ex_tz, sd_ex_tz)

#South Africa
mean_ex_sa <- excess_mortality[excess_mortality$country == "South Africa", "mean_value"]
sd_ex_sa <- (excess_mortality[excess_mortality$country == "South Africa", "upper_ci"]-excess_mortality[exce

sim_excess_sa <- rnorm(3000, mean_ex_sa, sd_ex_sa)

sim_excess <- excess_mortality %>%
  slice(rep(1:n(), each = 3000)) %>%
  mutate(IFR = rep(sim_IFR,3)) %>%
  mutate(Excess_total = c(sim_excess_ch,sim_excess_tz,sim_excess_sa)) %>% #per 100,000
  mutate(Infected_total = Excess_total/IFR) %>% #per 100,000
  mutate(Infected_week = Infected_total/104)
  #i use IFR before the simulation to retain some variability, if not, i would just get values 0,100,200

```

```

I_covEX_ch <- sim_excess %>%
  filter(country == "Switzerland") %>%
  mutate(I = Infected_week/100000*n_ch)

I_covEX_tz <- sim_excess %>%
  filter(country == "Tanzania") %>%
  mutate(I = Infected_week/100000*n_tz)

I_covEX_sa <- sim_excess %>%
  filter(country == "South Africa") %>%
  mutate(I = Infected_week/100000*n_sa)

##fixed on South Africa level
I_covEX_ch_fix <- sim_excess %>%
  filter(country == "South Africa") %>%
  mutate(I = Infected_week/100000*n_ch)

I_covEX_tz_fix <- sim_excess %>%
  filter(country == "South Africa") %>%
  mutate(I = Infected_week/100000*n_tz)

I_covEX_sa_fix <- sim_excess %>%
  filter(country == "South Africa") %>%
  mutate(I = Infected_week/100000*n_sa)

```

```

#Switzerland
##ACH
Vol_ch <- 233000

```

```

ACH_ch <- ch %>%
  filter(co2 >= 405) %>%
  mutate(Q = (0.13*38000)/(co2-400)) %>%
  mutate(ACH = (3600*Q*n_ch)/Vol_ch) %>%
  summarise(mean=mean(ACH), sd=sd(ACH), median = median(ACH), lower = quantile(ACH, probs = 0.25), upper = quantile(ACH, probs = 0.75))

```

```

##f
f_ch <- ch %>%
  mutate(f = ((co2-C_o)/C_a)/1000000) %>%
  summarise(mean=mean(f), sd=sd(f), median = median(f), lower = quantile(f, probs = 0.25), upper = quantile(f, probs = 0.75))

```

```

#South Africa

```

```

##ACHF

```

```

Vol_sa <- 180000

```

```

ACH_sa <- sa %>%
  filter(co2 >= 401) %>%
  mutate(Q = (0.13*38000)/(co2-400)) %>%
  mutate(ACH = (3600*Q*n_sa)/Vol_sa) %>%
  summarise(mean=mean(ACH), sd=sd(ACH), median = median(ACH), lower = quantile(ACH, probs = 0.25), upper = quantile(ACH, probs = 0.75))

```

```

##f
f_sa <- sa %>%
  mutate(f = ((co2-C_o)/C_a)/1000000) %>%
  summarise(mean=mean(f), sd=sd(f), median = median(f), lower = quantile(f, probs = 0.25), upper = quantile(f, probs = 0.75))

```

```

#Tanzania
##ACH
Vol_tz <- 162000

ACH_tz <- tz %>%
  filter(co2 >= 401) %>%
  mutate(Q = (0.13*38000)/(co2-400)) %>%
  mutate(ACH = (3600*Q*n_sa)/Vol_tz) %>%
  summarise(mean=mean(ACH), sd=sd(ACH), median = median(ACH), lower = quantile(ACH, probs = 0.25), upper = quantile(ACH, probs = 0.75))

##f
f_tz <- tz %>%
  mutate(f = ((co2-C_o)/C_a)/1000000) %>%
  summarise(mean=mean(f), sd=sd(f), median = median(f), lower = quantile(f, probs = 0.25), upper = quantile(f, probs = 0.75))

ch %>%
  summarise(mean = mean(co2), sd = sd(co2), median = median(co2), lower = quantile(co2, probs = 0.25), upper = quantile(co2, probs = 0.75))

## # A tibble: 1 x 5
##   mean    sd median lower upper
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 1279.  603.  1158.  818. 1651.

sa %>%
  summarise(mean = mean(co2), sd = sd(co2), median = median(co2), lower = quantile(co2, probs = 0.25), upper = quantile(co2, probs = 0.75))

## # A tibble: 1 x 5
##   mean    sd median lower upper
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 1556.  753.  1410  1013 1955.

tz %>%
  summarise(mean = mean(co2), sd = sd(co2), median = median(co2), lower = quantile(co2, probs = 0.25), upper = quantile(co2, probs = 0.75))

## # A tibble: 1 x 5
##   mean    sd median lower upper
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  643.  177.   601   554   660

#preparing datasets for plotting
#Switzerland
df_ch_TB <- tibble(school = c(rep("Switzerland", 3000)), f = sample_f_ch, q_tb = sample_q) %>%
  mutate("Prevalence" = 1 - exp(-(f*I_ch*q_tb*year)/n_ch)) %>%
  mutate("Fixed I/n" = 1 - exp(-(f*I_ch_fix*q_tb*year)/n_ch)) %>%
  pivot_longer(
    cols = c("Prevalence", "Fixed I/n"),
    names_to = "Assumption",
    values_to = "risk")

df_ch_COV_EX <- tibble(school = c(rep("Switzerland", 3000)),

```

```

      f = sample_f_ch,
      q_cov = sample_q_cov,
      I = I_covEX_ch$I) %>%
mutate(risk = 1 - exp(-(f*I*q_cov*month)/n_ch)) %>%
mutate(Assumption = c(rep("Prevalence",3000)))

df_ch_COV_FIX <- tibble(school = c(rep("Switzerland", 3000)),
      f = sample_f_ch,
      q_cov = sample_q_cov,
      I = I_covEX_ch_fix$I) %>%
mutate(risk = 1 - exp(-(f*I*q_cov*month)/n_ch)) %>%
mutate(Assumption = c(rep("Fixed I/n",3000)))

df_ch_COV <- rbind(df_ch_COV_EX,df_ch_COV_FIX)

#Tanzania
df_tz_TB <- tibble(school = c(rep("Tanzania", 3000)), f = sample_f_tz, q_tb = sample_q, q_cov = sample_q_cov,
      mutate("Prevalence" = 1 - exp(-(f*I_tz*q_tb*year)/n_tz)) %>%
      mutate("Fixed I/n" = 1 - exp(-(f*I_tz_fix*q_tb*year)/n_tz)) %>%
      pivot_longer(
        cols = c("Prevalence", "Fixed I/n"),
        names_to = "Assumption",
        values_to = "risk")

df_tz_COV_EX <- tibble(school = c(rep("Tanzania", 3000)),
      f = sample_f_tz,
      q_cov = sample_q_cov,
      I = I_covEX_tz$I) %>%
mutate(risk = 1 - exp(-(f*I*q_cov*month)/n_tz)) %>%
mutate(Assumption = c(rep("Prevalence",3000)))

df_tz_COV_FIX <- tibble(school = c(rep("Tanzania", 3000)),
      f = sample_f_tz,
      q_cov = sample_q_cov,
      I = I_covEX_tz_fix$I) %>%
mutate(risk = 1 - exp(-(f*I*q_cov*month)/n_tz)) %>%
mutate(Assumption = c(rep("Fixed I/n",3000)))

df_tz_COV <- rbind(df_tz_COV_EX,df_tz_COV_FIX)

#South Africa
df_sa_TB <- tibble(school = c(rep("South Africa", 3000)), f = sample_f_sa, q_tb = sample_q, q_cov = sample_q_cov,
      mutate("Prevalence" = 1 - exp(-(f*I_sa*q_tb*year)/n_sa)) %>%
      mutate("Fixed I/n" = 1 - exp(-(f*I_sa_fix*q_tb*year)/n_sa)) %>%
      pivot_longer(
        cols = c("Prevalence", "Fixed I/n"),
        names_to = "Assumption",
        values_to = "risk")

df_sa_COV_EX <- tibble(school = c(rep("South Africa", 3000)),
      f = sample_f_sa,
      q_cov = sample_q_cov,
      I = I_covEX_sa$I) %>%
mutate(risk = 1 - exp(-(f*I*q_cov*month)/n_sa)) %>%

```

```

mutate(Assumption = c(rep("Prevalence",3000)))

df_sa_COV_FIX <- tibble(school = c(rep("South Africa", 3000)),
                        f = sample_f_sa,
                        q_cov = sample_q_cov,
                        I = I_covEX_sa_fix$I) %>%
mutate(risk = 1 - exp(-(f*I*q_cov*month)/n_sa)) %>%
mutate(Assumption = c(rep("Fixed I/n",3000)))

df_sa_COV <- rbind(df_sa_COV_EX,df_sa_COV_FIX)

##

df_complet_TB <- bind_rows(df_ch_TB, df_sa_TB, df_tz_TB)
df_complet_COV <- bind_rows(df_ch_COV, df_tz_COV, df_sa_COV)

# mean+/-sd for Paper
##TB
median_fun <- function(data, indices) { #for bootstrapping (to estimate CI)
  median(data[indices])
}

# Perform bootstrapping with 1000 replications
df_ch_TB_PREV <- df_ch_TB %>%
  filter(Assumption == "Prevalence")

df_ch_TB_FIX <- df_ch_TB %>%
  filter(Assumption == "Fixed I/n")

boot_TB_ch_PREV <- boot(df_ch_TB_PREV$risk$f, median_fun, R = 1000)
boot_TB_ch_FIX <- boot(df_ch_TB_FIX$risk$f, median_fun, R = 1000)

# Calculate the 95% CI for the median
median(df_ch_TB_PREV$risk$f)

## [1] 0.001708731

med_TB_ch_PREV <- quantile(boot_TB_ch_PREV$t, c(0.025, 0.975))

median(df_ch_TB_FIX$risk$f)

## [1] 0.1926376

med_TB_ch_FIX <- quantile(boot_TB_ch_FIX$t, c(0.025, 0.975))

# Print the results
print(paste("95% CI for median:", med_TB_ch_PREV))

## [1] "95% CI for median: 0.00162621748914351"
## [2] "95% CI for median: 0.00177072374889531"

```



```

print(paste("95% CI for median:", med_TB_ch_FIX))

## [1] "95% CI for median: 0.182667695751964"
## [2] "95% CI for median: 0.198338794651666"

###Tanzania
df_tz_TB_PREV <- df_tz_TB %>%
  filter(Assumption == "Prevalence")

df_tz_TB_FIX <- df_tz_TB %>%
  filter(Assumption == "Fixed I/n")

boot_TB_tz_PREV <- boot(df_tz_TB_PREV$risk$f, median_fun, R = 1000)
boot_TB_tz_FIX <- boot(df_tz_TB_FIX$risk$f, median_fun, R = 1000)

# Calculate the 95% CI for the median
median(df_tz_TB_PREV$risk$f)

## [1] 0.02177995

med_TB_tz_PREV <- quantile(boot_TB_tz_PREV$t, c(0.025, 0.975))

median(df_tz_TB_FIX$risk$f)

## [1] 0.05286204

med_TB_tz_FIX <- quantile(boot_TB_tz_FIX$t, c(0.025, 0.975))

# Print the results
print(paste("95% CI for median:", med_TB_tz_PREV))

## [1] "95% CI for median: 0.0211083618985477"
## [2] "95% CI for median: 0.0226179246265176"

print(paste("95% CI for median:", med_TB_tz_FIX))

## [1] "95% CI for median: 0.0512108443525198"
## [2] "95% CI for median: 0.054860710586647"

###South Africa
df_sa_TB_PREV <- df_sa_TB %>%
  filter(Assumption == "Prevalence")

df_sa_TB_FIX <- df_sa_TB %>%
  filter(Assumption == "Fixed I/n")

boot_TB_sa_PREV <- boot(df_sa_TB_PREV$risk$f, median_fun, R = 1000)
boot_TB_sa_FIX <- boot(df_sa_TB_FIX$risk$f, median_fun, R = 1000)

# Calculate the 95% CI for the median
median(df_sa_TB_PREV$risk$f)

```

```
## [1] 0.2437176

med_TB_sa_PREV <- quantile(boot_TB_sa_PREV$t, c(0.025, 0.975))

median(df_sa_TB_FIX$risk$f)
```

```
## [1] 0.2437176

med_TB_sa_FIX <- quantile(boot_TB_sa_FIX$t, c(0.025, 0.975))

# Print the results
print(paste("95% CI for median:", med_TB_sa_PREV))
```

```
## [1] "95% CI for median: 0.234365390502962"
## [2] "95% CI for median: 0.255612726703808"

print(paste("95% CI for median:", med_TB_sa_FIX))
```

```
## [1] "95% CI for median: 0.234169336800885"
## [2] "95% CI for median: 0.255831655310328"
```

```
#COV

boot_COV_ch_PREV <- boot(df_ch_COV_EX$risk$f, median_fun, R = 1000)
boot_COV_ch_FIX <- boot(df_ch_COV_FIX$risk$f, median_fun, R = 1000)

# Calculate the 95% CI for the median
median(df_ch_COV_EX$risk$f)
```

```
## [1] 0.008650671

med_COV_ch_PREV <- quantile(boot_COV_ch_PREV$t, c(0.025, 0.975))

median(df_ch_COV_FIX$risk$f)
```

```
## [1] 0.02703613

med_COV_ch_FIX <- quantile(boot_COV_ch_FIX$t, c(0.025, 0.975))

# Print the results
print(paste("95% CI for median:", med_COV_ch_PREV))
```

```
## [1] "95% CI for median: 0.00825578952775848"
## [2] "95% CI for median: 0.00902990092502701"

print(paste("95% CI for median:", med_COV_ch_FIX))
```

```
## [1] "95% CI for median: 0.0259191030604106"
## [2] "95% CI for median: 0.0281989654559248"
```

```
###Tanzania
boot_COV_tz_PREV <- boot(df_tz_COV_EX$risk$f, median_fun, R = 1000)
boot_COV_tz_FIX <- boot(df_tz_COV_FIX$risk$f, median_fun, R = 1000)
```

```
# Calculate the 95% CI for the median
median(df_tz_COV_EX$risk$f)
```

```
## [1] 0.003118016
```

```
med_COV_tz_PREV <- quantile(boot_COV_tz_PREV$t, c(0.025, 0.975))
median(df_tz_COV_FIX$risk$f)
```

```
## [1] 0.007184675
```

```
med_COV_tz_FIX <- quantile(boot_COV_tz_FIX$t, c(0.025, 0.975))
```

```
# Print the results
print(paste("95% CI for median:", med_COV_tz_PREV))
```

```
## [1] "95% CI for median: 0.00300586913675099"
## [2] "95% CI for median: 0.0032416450377133"
```

```
print(paste("95% CI for median:", med_COV_tz_FIX))
```

```
## [1] "95% CI for median: 0.00691056122897866"
## [2] "95% CI for median: 0.00748110058987533"
```

```
###South Africa
boot_COV_sa_PREV <- boot(df_sa_COV_EX$risk$f, median_fun, R = 1000)
boot_COV_sa_FIX <- boot(df_sa_COV_FIX$risk$f, median_fun, R = 1000)
```

```
# Calculate the 95% CI for the median
median(df_sa_COV_EX$risk$f)
```

```
## [1] 0.0357451
```

```
med_COV_sa_PREV <- quantile(boot_COV_sa_PREV$t, c(0.025, 0.975))
median(df_sa_COV_FIX$risk$f)
```

```
## [1] 0.0357451
```

```
med_COV_sa_FIX <- quantile(boot_COV_sa_FIX$t, c(0.025, 0.975))
```

```
# Print the results
print(paste("95% CI for median:", med_COV_sa_PREV))
```

```
## [1] "95% CI for median: 0.0342379034686539"
## [2] "95% CI for median: 0.0373817282361993"
```

```
print(paste("95% CI for median:", med_COV_sa_FIX))
```

```
## [1] "95% CI for median: 0.0342981913952018"
## [2] "95% CI for median: 0.0374670771408825"
```

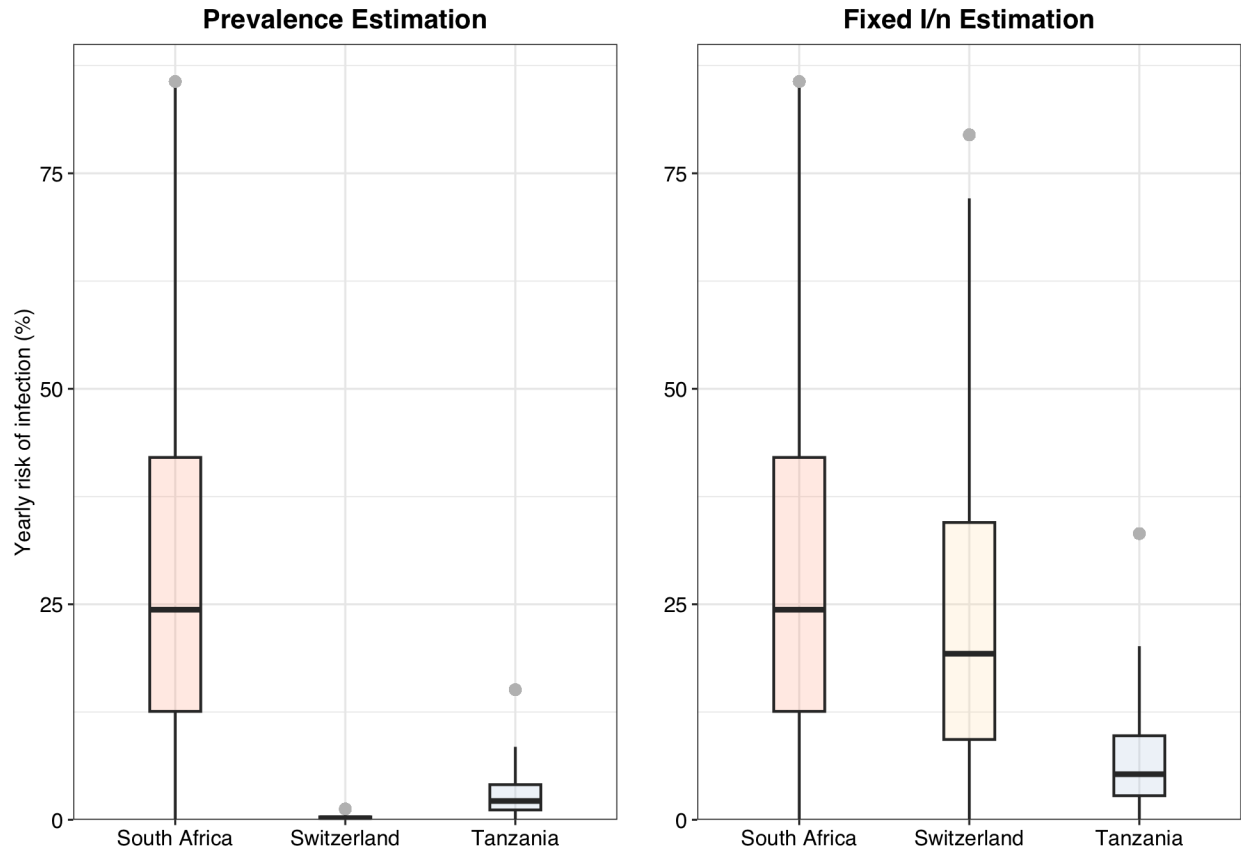
Plots of the transmission risk

```
school_order <- c("South Africa", "Switzerland", "Tanzania")

p2 <- df_complet_TB %>%
  filter(Assumption == "Fixed I/n") %>% #two panel
  mutate(school = factor(school, levels = school_order)) %>%
  ggplot(aes(x = school, y=risk$f, fill = school)) +
  scale_y_continuous(labels = function(x) paste0(x*100), expand = c(0,0))+
  xlab("Country") +
  ylab("") +
  theme_bw2() +
  theme(axis.title.x = element_blank())+
  guides(fill = "none") +
  geom_boxplot(width=0.3, alpha=0.2, outlier.shape = NA) +
  geom_point(aes(x=2,y=quantile(df_ch_TB$risk$f[df_ch_TB$Assumption == "Fixed I/n"], probs = 0.95)), col = "red", size=10) +
  geom_point(aes(x=1,y=quantile(df_sa_TB$risk$f[df_sa_TB$Assumption == "Fixed I/n"], probs = 0.95)), col = "red", size=10) +
  geom_point(aes(x=3,y=quantile(df_tz_TB$risk$f[df_tz_TB$Assumption == "Fixed I/n"], probs = 0.95)), col = "red", size=10) +
  scale_fill_manual(values = c("#FFA07A", "#FFDEAD", "#BOC4DE")) +
  ggtitle("Fixed I/n Estimation") +
  theme(plot.title = element_text(hjust = 0.5)) +
  coord_cartesian(ylim = c(0, 0.9))

p1 <- df_complet_TB %>%
  filter(Assumption == "Prevalence") %>% #two panel
  mutate(school = factor(school, levels = school_order)) %>%
  ggplot(aes(x = school, y=risk$f, fill = school)) +
  scale_y_continuous(labels = function(x) paste0(x*100), expand = c(0,0))+
  xlab("Country") +
  ylab("Yearly risk of infection (%)") +
  theme_bw2() +
  theme(axis.title.x = element_blank())+
  guides(fill = "none") +
  geom_boxplot(width=0.3, alpha=0.2, outlier.shape = NA) +
  geom_point(aes(x=2,y=quantile(df_ch_TB$risk$f[df_ch_TB$Assumption == "Prevalence"], probs = 0.95)), col = "red", size=10) +
  geom_point(aes(x=1,y=quantile(df_sa_TB$risk$f[df_sa_TB$Assumption == "Prevalence"], probs = 0.95)), col = "red", size=10) +
  geom_point(aes(x=3,y=quantile(df_tz_TB$risk$f[df_tz_TB$Assumption == "Prevalence"], probs = 0.95)), col = "red", size=10) +
  scale_fill_manual(values = c("#FFA07A", "#FFDEAD", "#BOC4DE")) +
  ggtitle("Prevalence Estimation") +
  theme(plot.title = element_text(hjust = 0.5)) +
  coord_cartesian(ylim = c(0, 0.90))

plot_grid(p1, p2, ncol = 2) #plot in the paper
```



```

plot_sars1 <- df_complet_COV%>%
  filter(Assumption == "Prevalence") %>% #two panel
  ggplot(aes(x = school, y=risk$f, fill = school)) +
  scale_y_continuous(labels = function(x) paste0(x*100), expand = c(0,0))+
  xlab("Country") +
  ylab("Monthly risk of infection (%)") +
  theme_bw2() +
  theme(axis.title.x = element_blank()) +
  geom_boxplot(width=0.3, alpha=0.2, outlier.shape = NA) +
  guides(fill="none")+
  geom_point(aes(x = 2, y = quantile(df_ch_COV_EX$risk$f, probs = 0.95)), colour = "grey") +
  geom_point(aes(x = 1, y = quantile(df_sa_COV_EX$risk$f, probs = 0.95)), colour = "grey") +
  geom_point(aes(x = 3, y = quantile(df_tz_COV_EX$risk$f, probs = 0.95)), colour = "grey") +
  scale_fill_manual(values = c("#BOC4DE", "#FFA07A", "#FFDEAD")) +
  ggtitle("Prevalence Estimation") +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_color_manual(values = c("#DDA0DD", "steelblue")) +
  coord_cartesian(ylim=c(0,0.20))

plot_sars2 <- df_complet_COV%>%
  filter(Assumption == "Fixed I/n") %>% #two panel
  ggplot(aes(x = school, y=risk$f, fill = school)) +
  scale_y_continuous(labels = function(x) paste0(x*100), expand = c(0,0))+
  xlab("Country") +
  ylab("") +
  theme_bw2() +

```

```

theme(axis.title.x = element_blank()) +
geom_boxplot(width=0.3, alpha=0.2, outlier.shape = NA) +
guides(fill="none")+
geom_point(aes(x = 2, y = quantile(df_ch_COV_FIX$risk$f, probs = 0.95)), colour = "grey") +
geom_point(aes(x = 1, y = quantile(df_sa_COV_FIX$risk$f, probs = 0.95)), colour = "grey") +
geom_point(aes(x = 3, y = quantile(df_tz_COV_FIX$risk$f, probs = 0.95)), colour = "grey") +
scale_fill_manual(values = c("#BOC4DE", "#FFA07A", "#FFDEAD")) +
ggtitle("Fixed I/n Estimation") +
theme(plot.title = element_text(hjust = 0.5)) +
scale_color_manual(values = c("#DDA0DD", "steelblue")) +
coord_cartesian(ylim=c(0,0.20))

plot_grid(plot_sars1, plot_sars2, ncol = 2) #plot in the paper

```

