MATH497-Best Draw?

Team Pbh

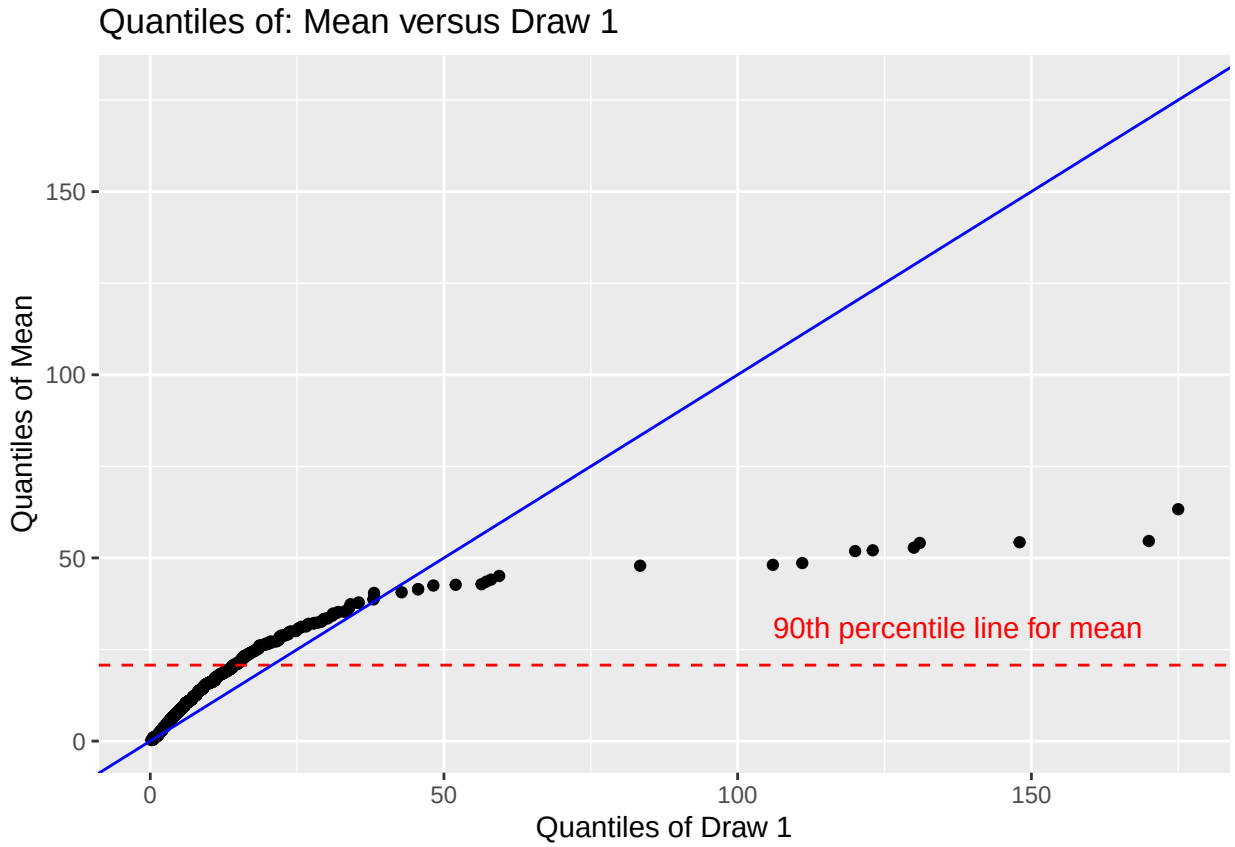
2023-03-30

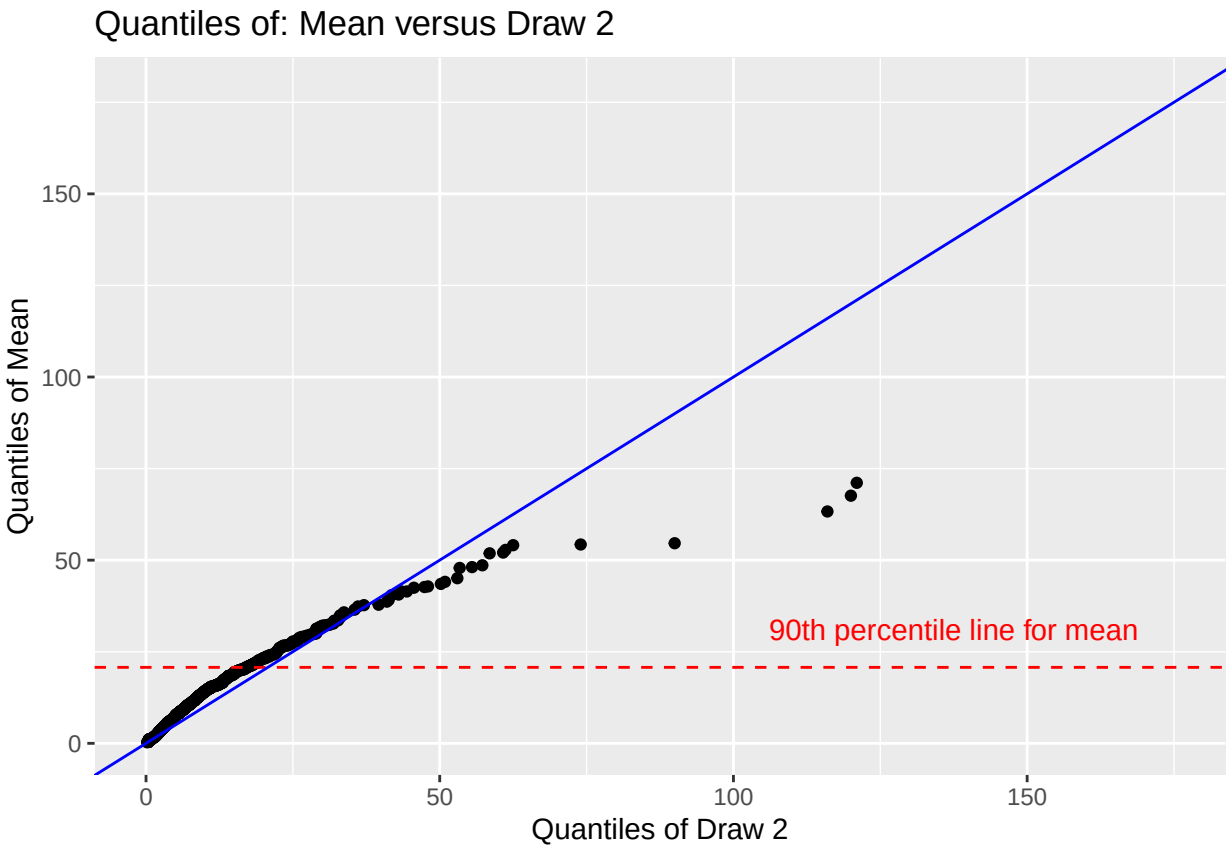
```
rm(list=ls())
df1 <- read.csv('C:/Users/krtfe/Downloads/497CleanedSequentialData.csv')[-1]

# copy of the dataframe
df <- df1

# df with just relevant columns
df <- df[1:12]

# computation of the means
df['Mean'] <- rowMeans(df[3:12])
df1['Mean'] <- rowMeans(df[3:12])

# df with removed na values
dfRMNA <- na.omit(df)

# df with removed duplicated addresses
dfRMNA <- dfRMNA[duplicated(dfRMNA$Address) == F,]
```

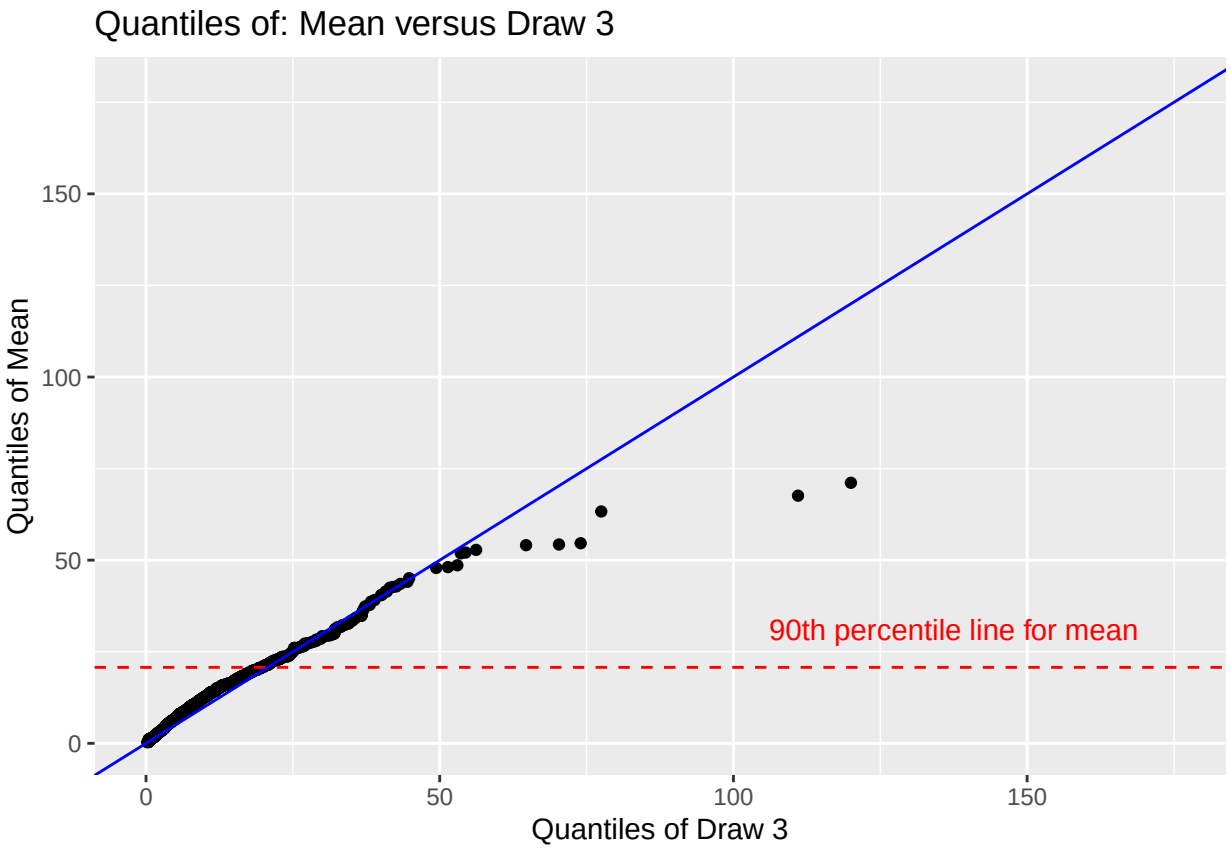
```
library(EnvStats)
library(ggpubr)

# empty dataframe
dfQuant <- data.frame(Mean = rep(NA, nrow(dfRMNA)))

# 90th percentile for the mean
perc90 <- quantile(dfRMNA$Mean, probs = seq(0, 1, 0.9), na.rm = TRUE)[2]

# plotting the quantile data against each other
for (i in 3:12) {
  plotQuan <- qqplot(dfRMNA[,i], dfRMNA$Mean, plot = FALSE)
  dfQuant[paste('Draw_', i-2, sep = '')] <- plotQuan$x

  #plot
  print(ggplot() +
          geom_point(aes(x=plotQuan$x, y=plotQuan$y)) +
          geom_hline(yintercept = perc90, linetype = 'dashed',
                     color = 'red') +
          geom_abline(intercept = 0, slope = 1, color = 'blue') +
```

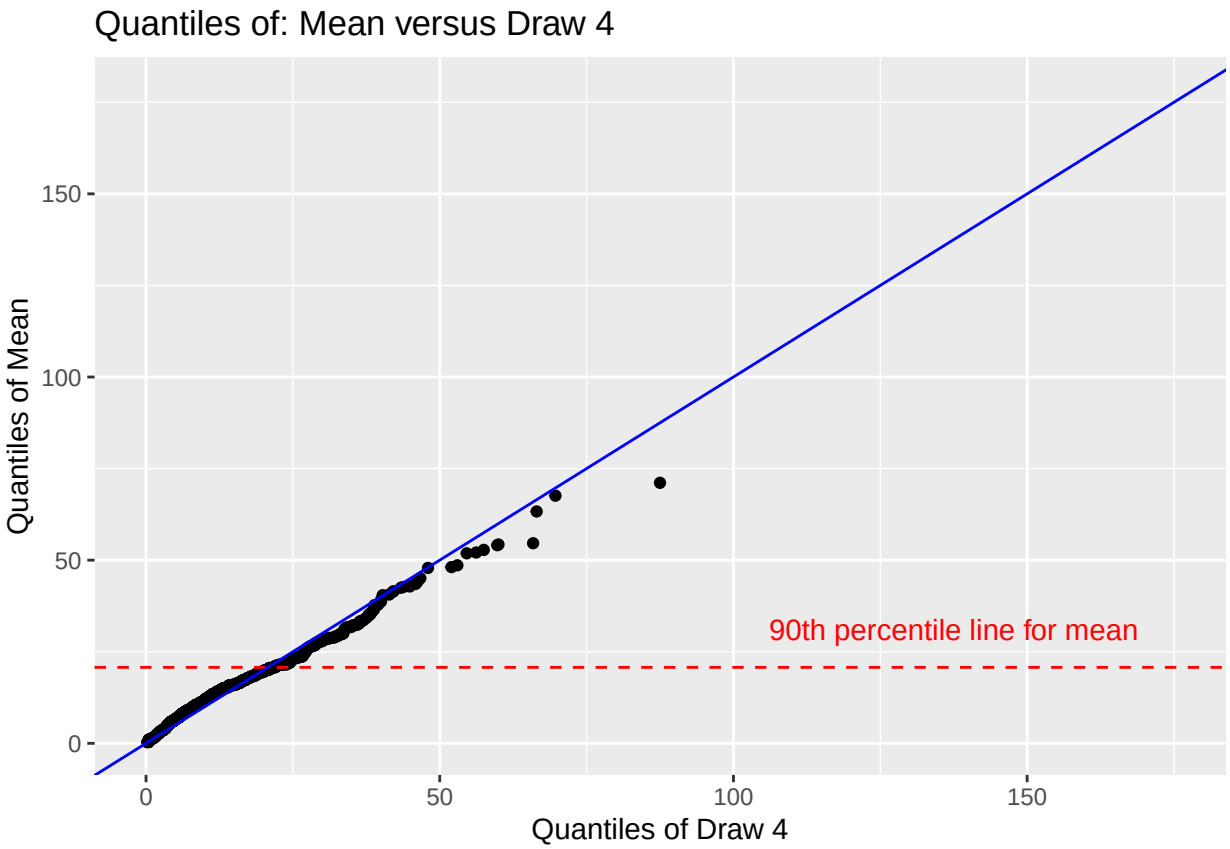
```
          ggtitle(paste('Quantiles of: Mean versus Draw', i-2)) +
          ylab('Quantiles of Mean') + xlab(paste('Quantiles of Draw',
                                                 i-2)) +
          annotate(geom = 'text',
                   label = '90th percentile line for mean',
                   color = 'red', x = 137.5, y = perc90*1.5) +
          xlim(c(0, 175)))
}
```

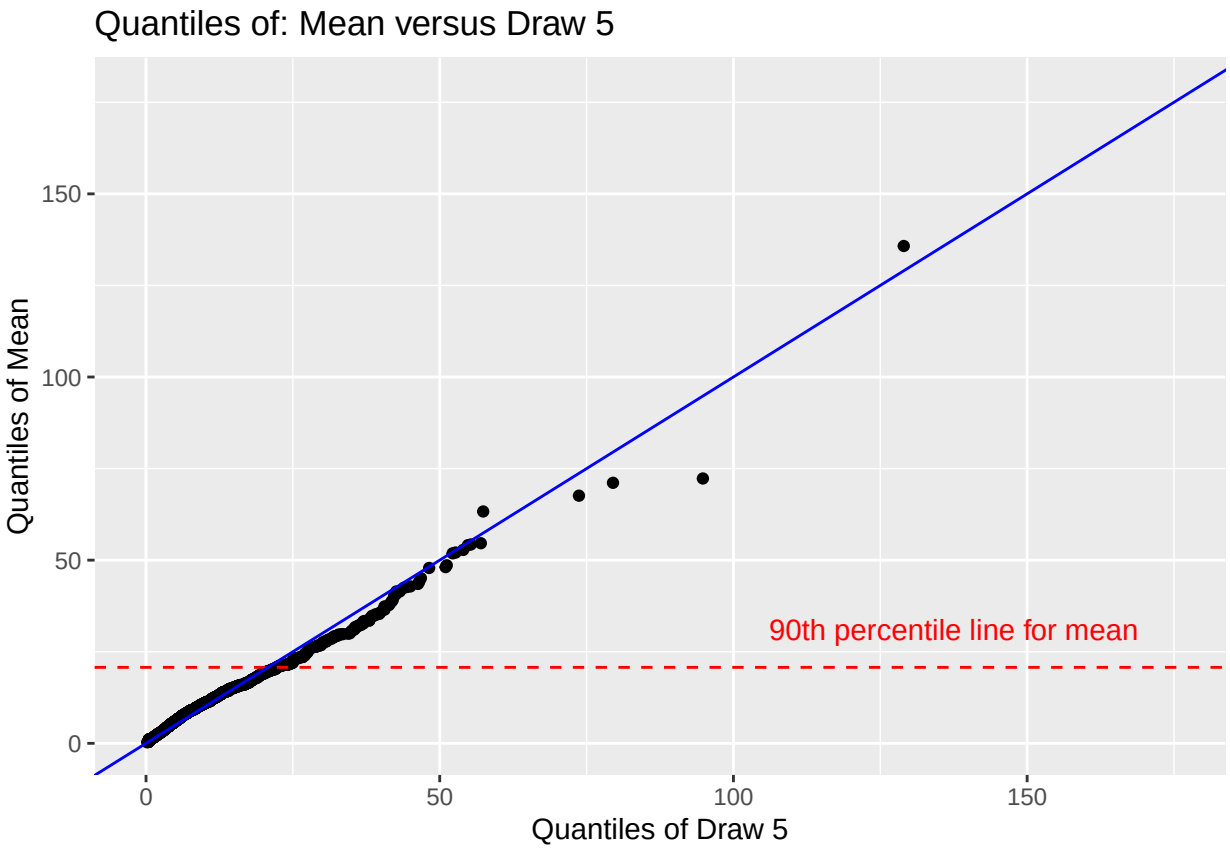
Quantile plots using qqplot() (comparing the quantiles of the plots vs. the quantiles of the mean):

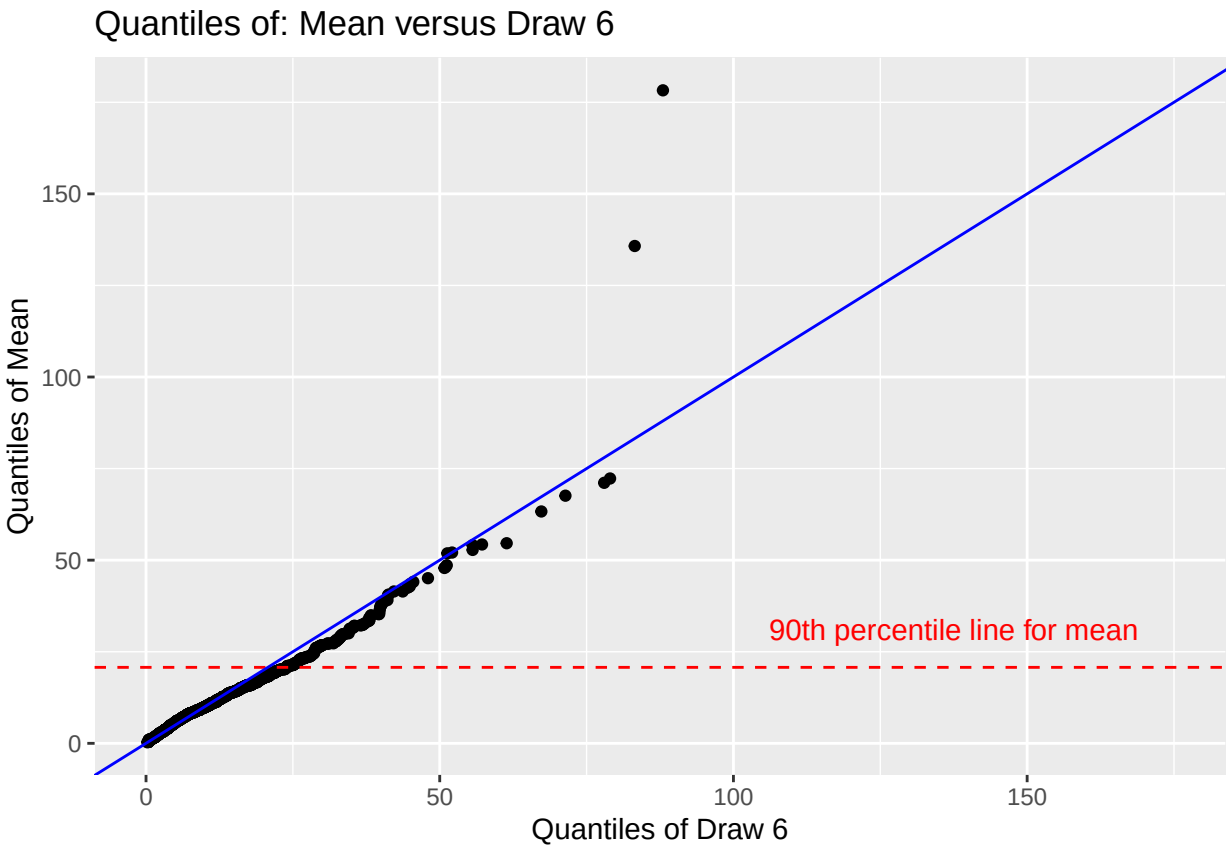


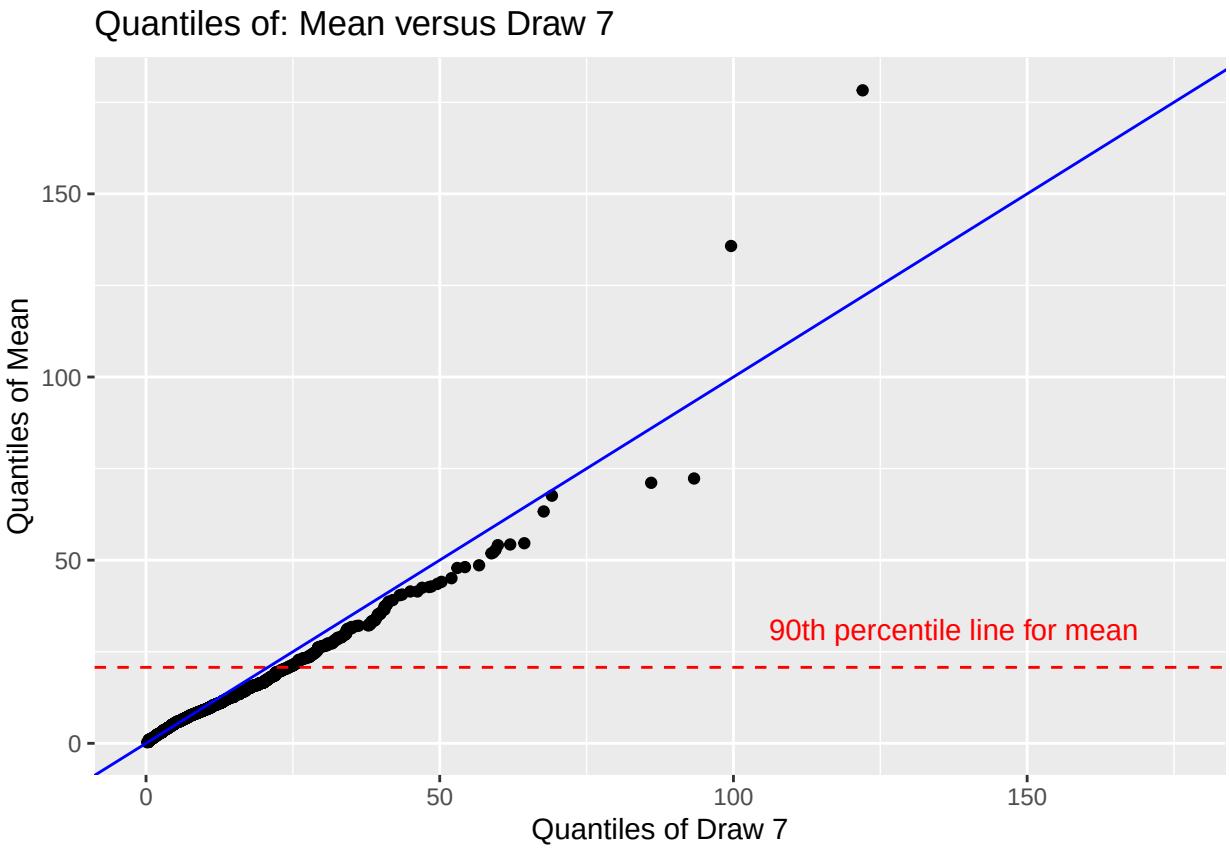

Quantiles of: Mean versus Draw 2
Quantiles of Mean
150
100
50
0
0
50
100
150
Quantiles of Draw 2
90th percentile line for mean

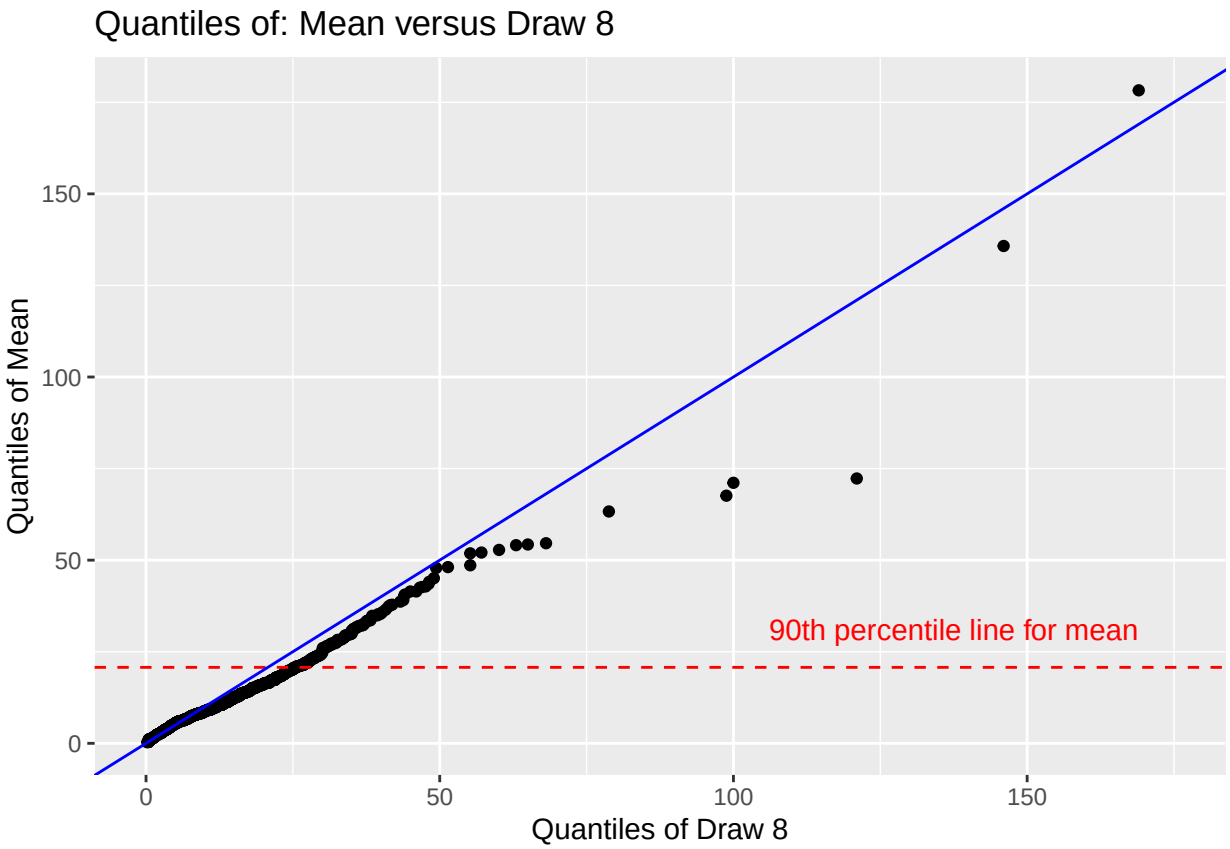

Quantiles of: Mean versus Draw 3
Quantiles of Mean
150
100
50
0
0
50
100
150
Quantiles of Draw 3
90th percentile line for mean

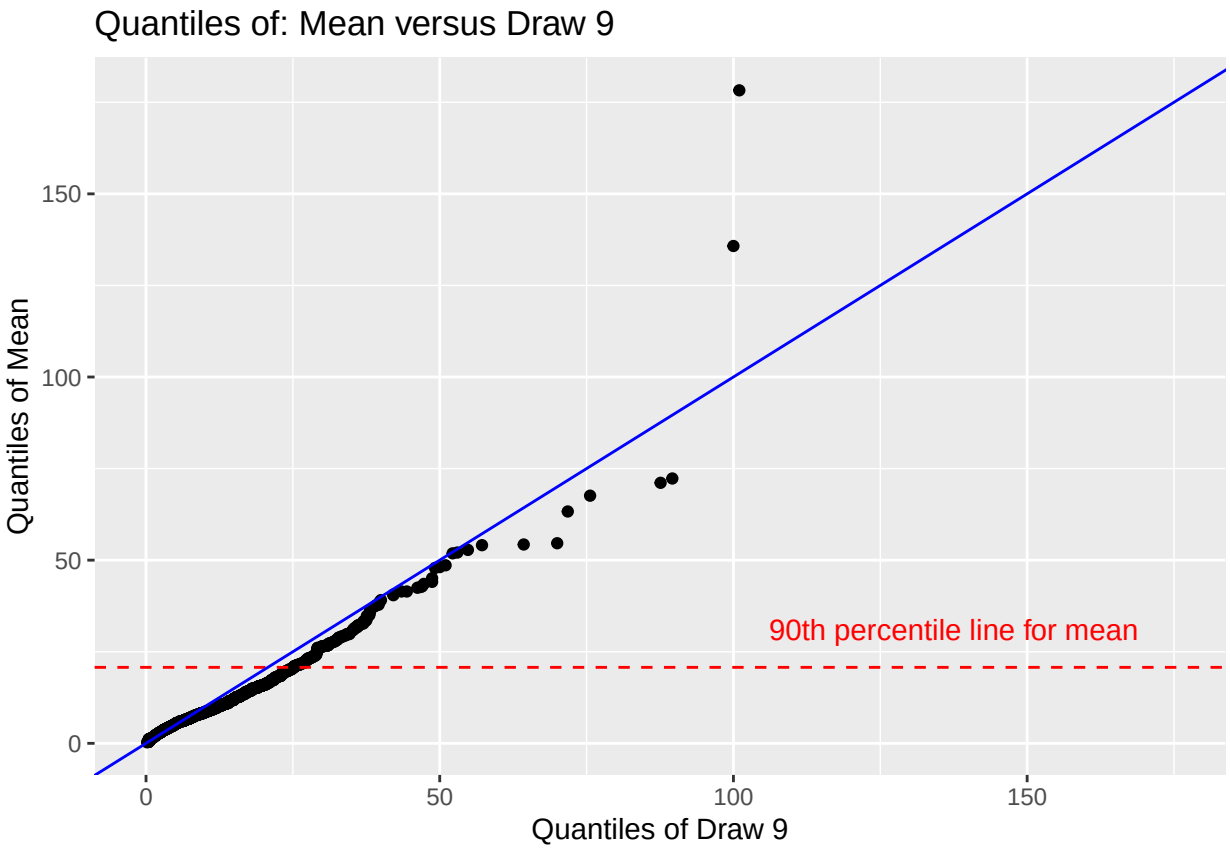

Quantiles of: Mean versus Draw 4
Quantiles of Mean
150
100
50
0
0
50
100
150
Quantiles of Draw 4
90th percentile line for mean

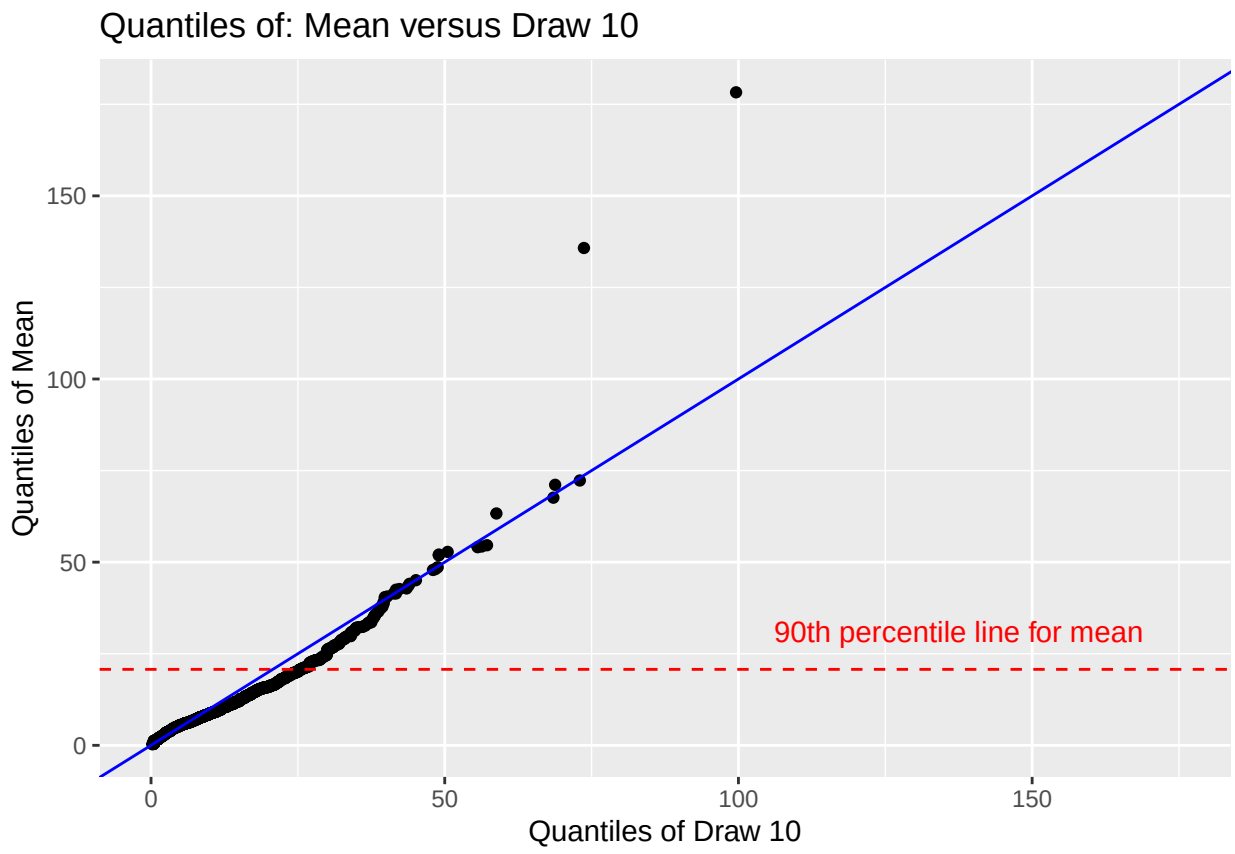

Quantiles of: Mean versus Draw 5
Quantiles of Mean
150
100
50
0
0
50
100
150
Quantiles of Draw 5
90th percentile line for mean


Quantiles of: Mean versus Draw 6
Quantiles of Mean
150
100
50
0
0
50
100
150
Quantiles of Draw 6
90th percentile line for mean


Quantiles of: Mean versus Draw 7
Quantiles of Mean
150
100
50
0
0
50
100
150
Quantiles of Draw 7
90th percentile line for mean


Quantiles of: Mean versus Draw 8
Quantiles of Mean
150
100
50
0
0
50
100
150
Quantiles of Draw 8
90th percentile line for mean


Quantiles of: Mean versus Draw 9
Quantiles of Mean
150
100
50
0
0
50
100
150
Quantiles of Draw 9
90th percentile line for mean




```
# just getting the mean values
dfQuant$Mean <- plotQuan$y
```

Comparison of the quantiles of the mean versus the quantiles of the draws.

```
for (i in 3:12) {
  b <- qqplot(df[,i], df$Mean, plot = FALSE)
  x <- b$x
  y <- b$y

  lg <- lm(y~x)
  summ <- summary(lg)$r.squared
  cat(paste('R^2 for draw ', i-2, ' vs. the mean: ', round(summ, 5), '\n', sep = ''))
}
```

Finding the R^2 values to determine the best relationship for the quantiles of the draws vs. the quantiles of the mean:

```
## R^2 for draw 1 vs. the mean: 0.74519
## R^2 for draw 2 vs. the mean: 0.7357
## R^2 for draw 3 vs. the mean: 0.81908
## R^2 for draw 4 vs. the mean: 0.90905
```

```
## R^2 for draw 5 vs. the mean: 0.97636
## R^2 for draw 6 vs. the mean: 0.92047
## R^2 for draw 7 vs. the mean: 0.95459
## R^2 for draw 8 vs. the mean: 0.97874
## R^2 for draw 9 vs. the mean: 0.92997
## R^2 for draw 10 vs. the mean: 0.90933
```

From this, the 8th draw gives the highest R^2 for an R^2 value of 0.97874

```
for (i in 3:12) {
  b <- qqplot(df[,i], df$Mean, plot = FALSE)
  x <- b$x
  y <- b$y

  lg <- lm(y~x)
  slope <- summary(lg)$coefficients[2]
  intercept <- summary(lg)$coefficients[1]
  cat(paste('Slope, intercept for draw ', i-2, ' vs. the mean: ',
            round(slope, 5), ', ', round(intercept, 5),
            '\n', sep = ''))
}
```

Looking at the slope and intercept of each of the above

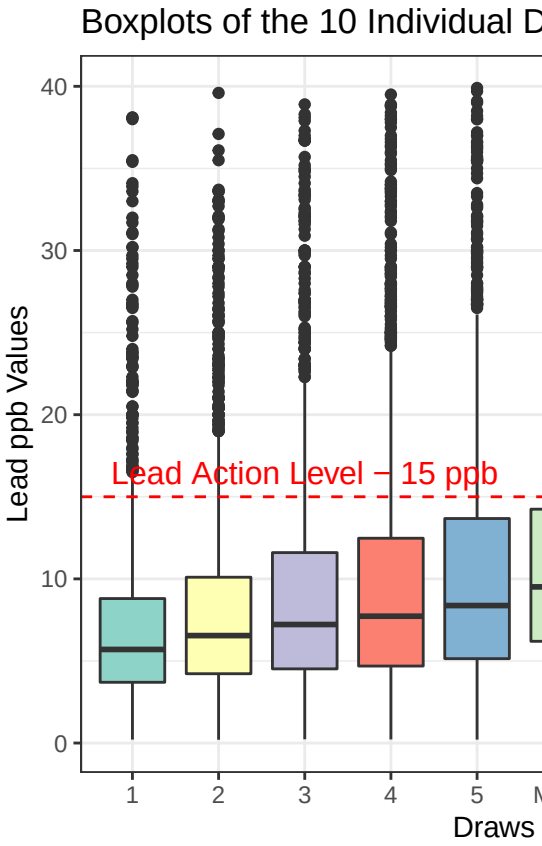
```
## Slope, intercept for draw 1 vs. the mean: 0.46071, 7.64377
## Slope, intercept for draw 2 vs. the mean: 0.44392, 7.48997
## Slope, intercept for draw 3 vs. the mean: 0.54694, 6.06729
## Slope, intercept for draw 4 vs. the mean: 0.70988, 4.01654
## Slope, intercept for draw 5 vs. the mean: 0.82597, 2.29406
## Slope, intercept for draw 6 vs. the mean: 0.94762, 0.27459
## Slope, intercept for draw 7 vs. the mean: 0.91514, -0.05556
## Slope, intercept for draw 8 vs. the mean: 0.83828, 0.33146
## Slope, intercept for draw 9 vs. the mean: 0.90907, -0.68655
## Slope, intercept for draw 10 vs. the mean: 0.93766, -0.87855
```

By this, the 6th draw gives the closest value to one for the slope. Since the 6th draw also gives one of the smallest intercept values, the 6th draw likely gives the closest quantiles to the mean's quantiles.

Overall, the 8th draw has the highest R^2 value, but the 6th draw gives the closest slope to one for the quantiles.

```
library(ggplot2)
library(reshape2)

restdf <- dfRMNA[3:13]
colnames(restdf) <- c('1', '2', '3', '4', '5',
                      '6', '7', '8', '9', '10', 'Mean')
a <- melt(restdf, na.rm = TRUE)
```

```
aplot <- ggplot(a[a$value < 40,],
       aes(x = variable, y = value, fill = variable),
       ) + geom_boxplot()

aplot +
  scale_fill_brewer(palette="Set3") + ylab('Lead ppb Values') +
  ggtitle('Boxplots of the 10 Individual Draws and the Mean') +
  scale_x_discrete(name = 'Draws and Mean',
                   limits = c('1', '2', '3', '4', '5', 'Mean', '6',
                              '7', '8', '9', '10') ) +
  geom_hline(yintercept = 15, linetype = 'dashed', color = 'red') +
  annotate(geom = 'text',
           label = 'Lead Action Level - 15 ppb', size = 4.2,
           color = 'red', x = 3, y = 16.5) +
  theme_bw()
```

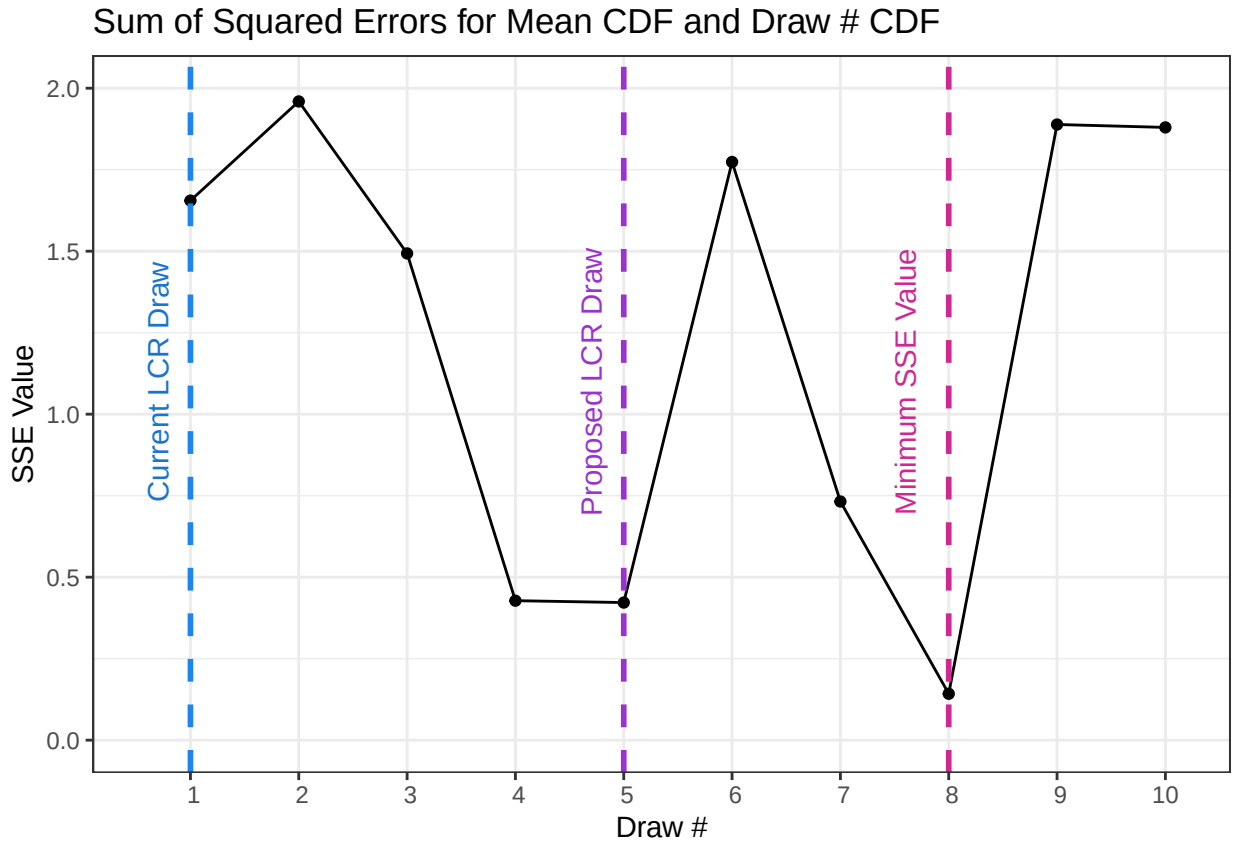


Boxplots of the mean versus the draws, just general visualizations:

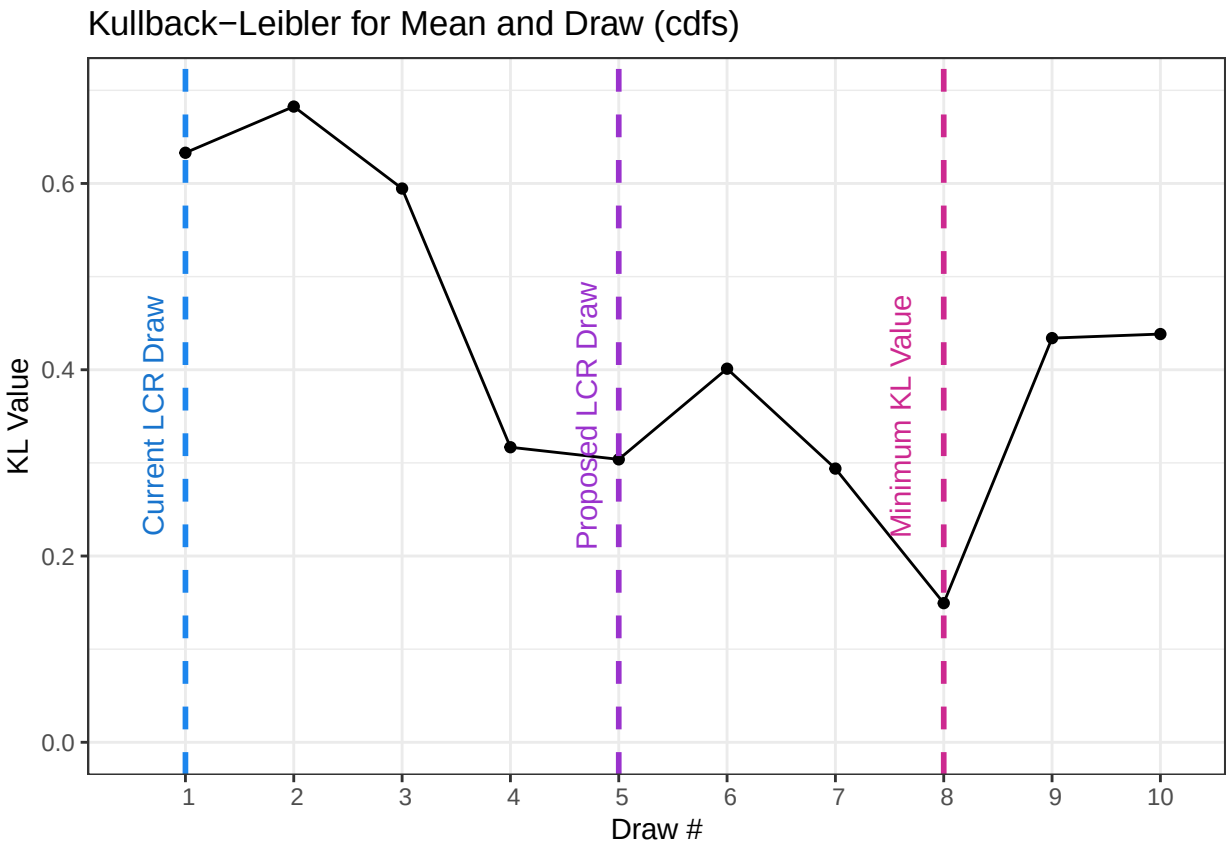
```
# pdf of the mean
pM <- hist(dfRMNA$Mean, breaks = seq(0, max(dfRMNA$Mean), length.out = 100), plot = FALSE)$counts / nrow

# cdf of the mean
cdfM <- cumsum(pM)
```

```r
# empty lists for computation
KLPlist <- c()
absvalL <- c()
ssevalL <- c()

# computing the differences for each type and each draw
for (i in 3:12) {
  # pdf of the draw
  p1 <- hist(unlist(dfRMNA[i]),
                    breaks = seq(0, max(dfRMNA[i]),
                                 length.out=100),
             plot = FALSE)$counts / nrow(dfRMNA)

  # cdf of the draw
  cdf1 <- cumsum(p1)

  # differences
  absval <- sum(abs(cdfM-cdf1))
  sseval <- sum((abs(cdfM-cdf1))^2)
  KLval <- max(abs(cdfM-cdf1))

  # add to list
  KLPlist <- c(KLPlist, KLval)
  absvalL <- c(absvalL, absval)
  ssevalL <- c(ssevalL, sseval)
}

# plot of the SSE for the draws vs the mean
ggplot() +
  geom_point(aes(x=1:10, y=ssevalL)) +
  geom_line(aes(x=1:10, y=ssevalL)) +
  ggtitle('Sum of Squared Errors for Mean CDF and Draw # CDF') +
  ylab('SSE Value') +
  ylim(c(0, 2)) +
  scale_x_discrete(name = 'Draw #', limits = c(1:10)) +
  geom_vline(xintercept = 8, size = 1,
             linetype = 'dashed', color = 'maroon3') +
  geom_vline(xintercept = 5, size = 1,
             linetype = 'dashed', color = 'darkorchid3') +
  geom_vline(xintercept = 1, size = 1,
             linetype = 'dashed', color = 'dodgerblue2') +
  annotate(geom = 'text',
           label = 'Minimum SSE Value', angle = 90,
           color = 'maroon3', x = 8*0.95, y = 1.1) +
  annotate(geom = 'text',
           label = 'Proposed LCR Draw', angle = 90,
           color = 'darkorchid3', x = 4.7, y = 1.1) +
  annotate(geom = 'text',
           label = 'Current LCR Draw', angle = 90,
           color = 'dodgerblue3', x = 0.7, y = 1.1) +
  theme_bw()
```

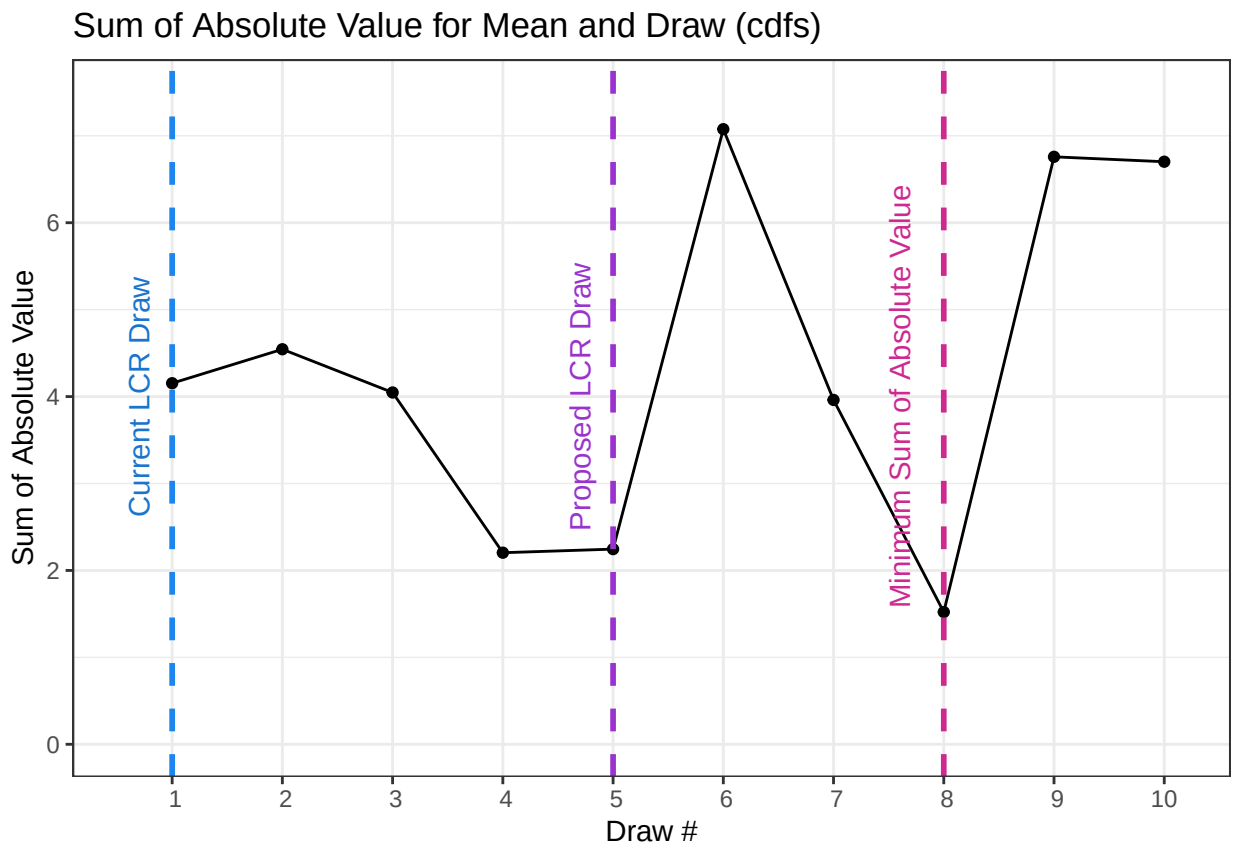
KL Divergence, Sum of Absolute Value, and SSE of the CDFs (plots and dataframe)




```
# plot of the KL-divergence for the draws vs. the mean
ggplot() +
  geom_point(aes(x=1:10, y=KLPlist)) +
  geom_line(aes(x=1:10, y=KLPlist)) +
  ggtitle('Kullback-Leibler for Mean and Draw (cdfs)') +
  ylab('KL Value') +
  ylim(c(0, 0.7)) +
  scale_x_discrete(name = 'Draw #', limits = c(1:10)) +
  geom_vline(xintercept = 8, size = 1,
             linetype = 'dashed', color = 'maroon3') +
  geom_vline(xintercept = 5, size = 1,
             linetype = 'dashed', color = 'darkorchid3') +
  geom_vline(xintercept = 1, size = 1,
             linetype = 'dashed', color = 'dodgerblue2') +
  annotate(geom = 'text',
           label = 'Minimum KL Value', angle = 90,
           color = 'maroon3', x = 8*0.95, y = 0.35) +
  annotate(geom = 'text',
           label = 'Proposed LCR Draw', angle = 90,
           color = 'darkorchid3', x = 4.7, y = 0.35) +
  annotate(geom = 'text',
           label = 'Current LCR Draw', angle = 90,
           color = 'dodgerblue3', x = 0.7, y = 0.35) +
  theme_bw()
```




```
# plot for the sum of absolute error for the draws vs the mean
ggplot() +
  geom_point(aes(x=1:10, y=absvalL)) +
  geom_line(aes(x=1:10, y=absvalL)) +
  ggtitle('Sum of Absolute Value for Mean and Draw (cdfs)') +
  ylab('Sum of Absolute Value') +
  ylim(c(0, 7.5)) +
  scale_x_discrete(name = 'Draw #', limits = c(1:10)) +
  geom_vline(xintercept = 8, size = 1,
             linetype = 'dashed', color = 'maroon3') +
  geom_vline(xintercept = 5, size = 1,
             linetype = 'dashed', color = 'darkorchid3') +
  geom_vline(xintercept = 1, size = 1,
             linetype = 'dashed', color = 'dodgerblue2') +
  annotate(geom = 'text',
           label = 'Minimum Sum of Absolute Value', angle = 90,
           color = 'maroon3', x = 8*0.95, y = 4) +
  annotate(geom = 'text',
           label = 'Proposed LCR Draw', angle = 90,
           color = 'darkorchid3', x = 4.7, y = 4) +
  annotate(geom = 'text',
           label = 'Current LCR Draw', angle = 90,
           color = 'dodgerblue3', x = 0.7, y = 4) +
  theme_bw()
```

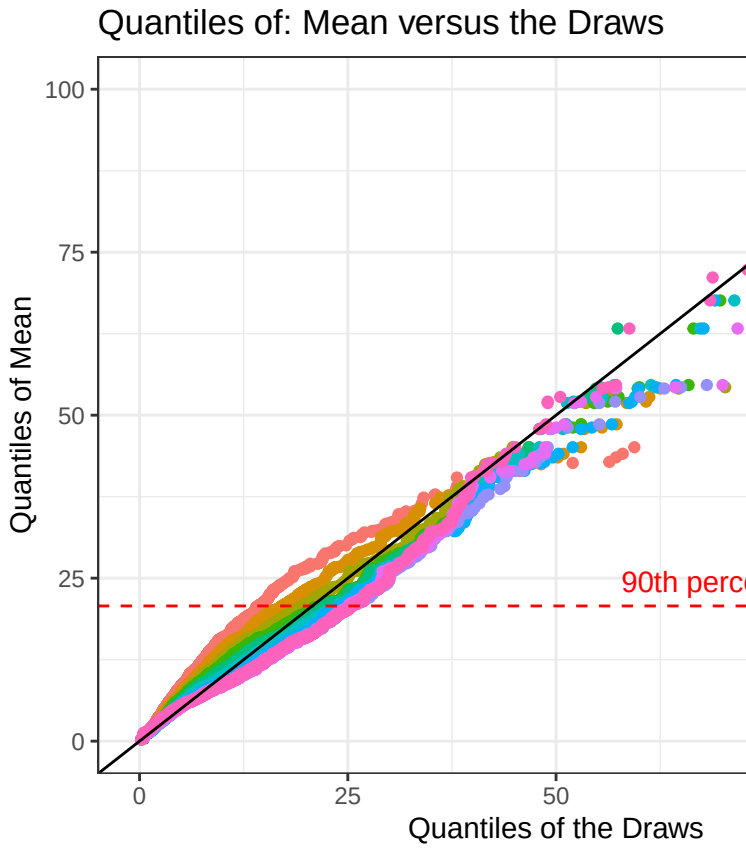



```
# data frame for direct comparison
lnormVal <- data.frame('Absolute Error' = absvalL,
                       'Sum of Squared Errors' = ssevalL,
                       'Kullback Leibler' = KLPlist)
print(lnormVal)
```

```
##    Absolute.Error Sum.of.Squared.Errors Kullback.Leibler
## 1        4.153751              1.6551800        0.6329820
## 2        4.544327              1.9593884        0.6825790
## 3        4.047117              1.4931748        0.5945443
## 4        2.203968              0.4278508        0.3168010
## 5        2.245505              0.4221097        0.3037818
## 6        7.076255              1.7736594        0.4011159
## 7        3.961562              0.7321380        0.2938624
## 8        1.521389              0.1423155        0.1494110
## 9        6.758215              1.8886046        0.4339740
## 10       6.701798              1.8797195        0.4383137
```

Here, the 8th draw's distribution gives the lowest KL number. Ideally, the closer the number is to 0, the more similar the two distributions are to each other. This suggests that the 8th draw best matches the mean distribution.

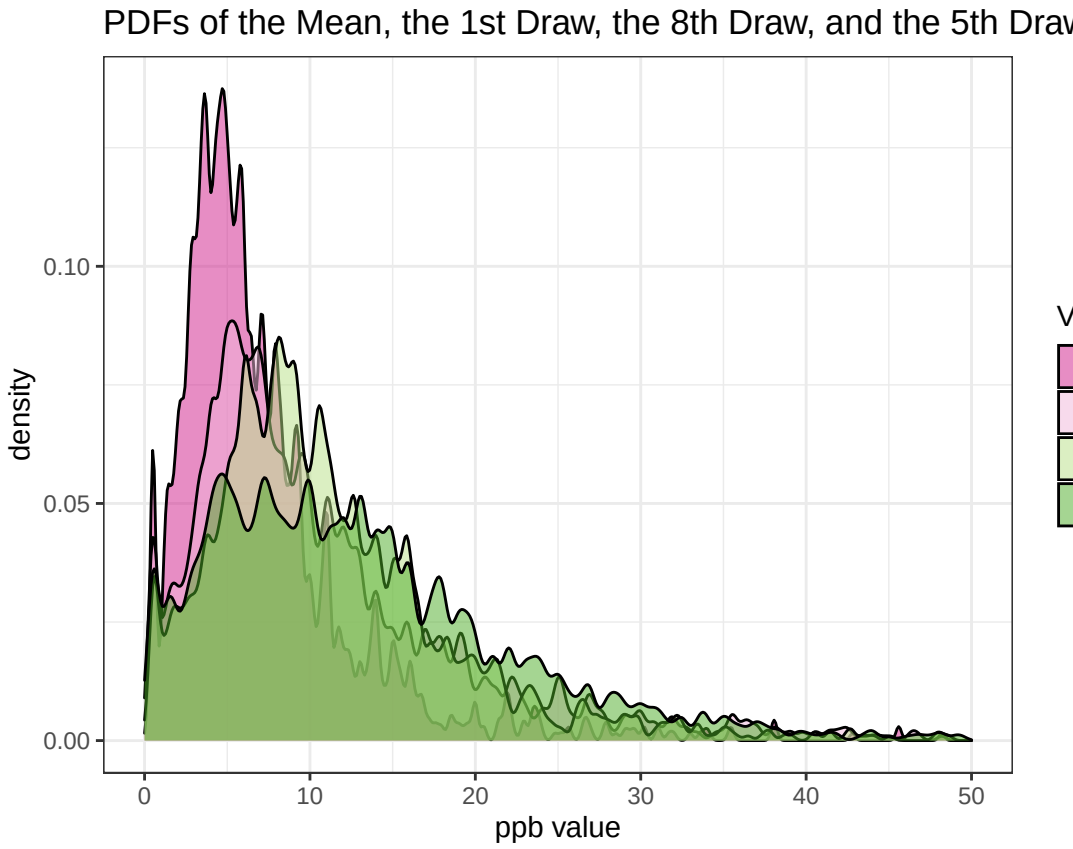
```
dfM <- melt(dfQuant, id.vars = 'Mean')
ggplot(dfM, aes(x = value, y = Mean, colour = variable)) +
  geom_point() +
  xlim(c(0, 100)) +
  ylim(c(0, 100)) +
  geom_hline(yintercept = perc90, linetype = 'dashed', color = 'red') +
  annotate(geom = 'text',
           label = '90th percentile line for mean',
           color = 'red', x = 80, y = perc90*1.1825) +
  geom_abline(intercept = 0, slope = 1) +
  ggtitle(paste('Quantiles of: Mean versus the Draws')) +
  ylab('Quantiles of Mean') +
  xlab('Quantiles of the Draws') +
  theme_bw()
```



The quantiles, all put together in one singular plot:

```
library(sm)

meltedDens <- melt(dfRMNA[c('X1st.Draw', 'X5th.Draw', 'Mean', 'X8th.Draw')])
ggplot(meltedDens, aes(x = value, fill = variable)) +
  geom_density(alpha = 0.5, adjust = 1/5) +
  xlim(c(0, 50)) +
```

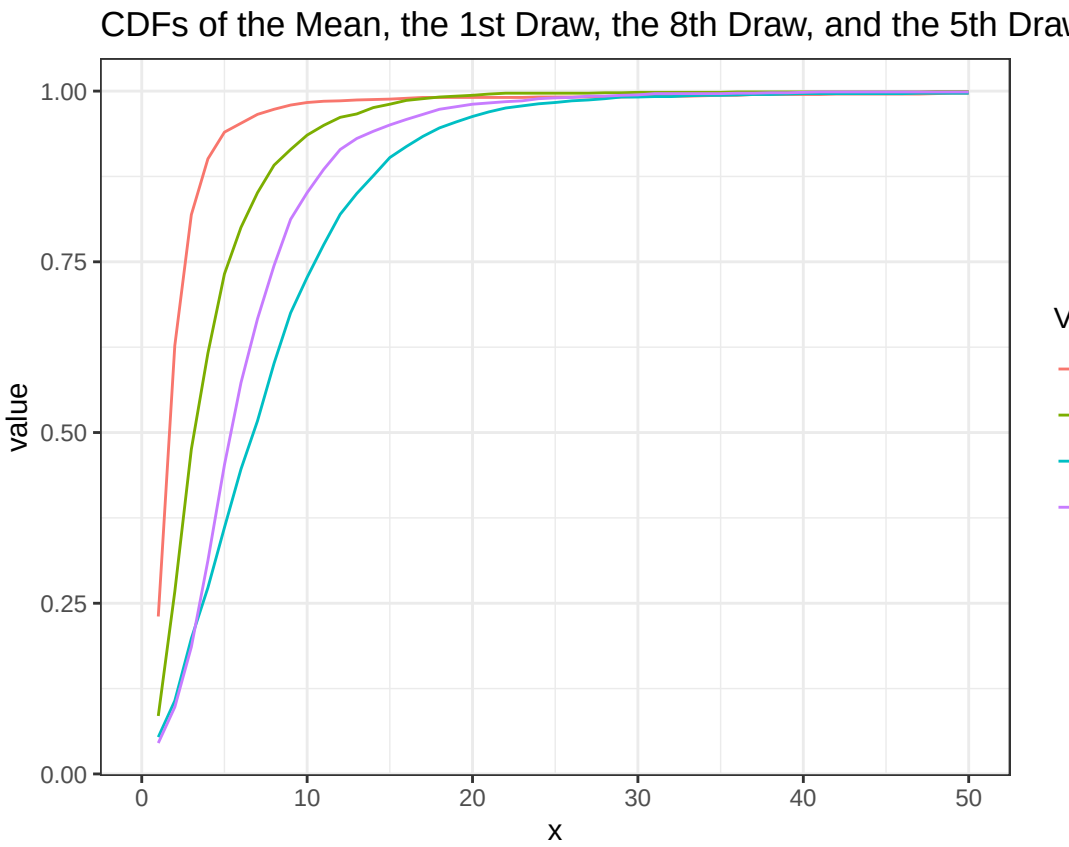
```
  xlab('ppb value') +
  scale_fill_brewer('Variable', palette = 'PiYG',
                    labels = c('1st Draw', '5th Draw', 'Mean', '8th Draw')) +
  theme_bw() +
  ggtitle('PDFs of the Mean, the 1st Draw, the 8th Draw, and the 5th Draw')
```



Visualization of the PDFs

```
p1 <- hist(dfRMNA$X1st.Draw,
           breaks = seq(0, max(dfRMNA$X1st.Draw),
                        length.out=100),
           plot = FALSE)$counts / nrow(dfRMNA)
cdf1 <- cumsum(p1)

p5 <- hist(dfRMNA$X5th.Draw,
                  breaks = seq(0, max(dfRMNA$X5th.Draw),
                               length.out=100),
             plot = FALSE)$counts / nrow(dfRMNA)
cdf5 <- cumsum(p5)

p8 <- hist(dfRMNA$X8th.Draw,
           breaks = seq(0, max(dfRMNA$X8th.Draw),
                        length.out=100),
           plot = FALSE)$counts / nrow(dfRMNA)
```

```
cdf8 <- cumsum(p8)

meltedDens <- melt(data.frame(cdf1, cdf5, cdf8, cdfM, 'x' = c(1:99)),
                   id.vars = 'x')
ggplot(meltedDens, aes(y = value, x = x, color = variable)) +
  geom_line() +
  ggtitle('CDFs of the Mean, the 1st Draw, the 8th Draw, and the 5th Draw') +
  # geom_area(aes(fill = variable, group = variable),
  #           alpha = 0.25, position = 'identity') +
  scale_color_hue(name="Variables",
                  labels=c('1st Draw', '5th Draw', '8th Draw', 'Mean',
                           '6th Draw')) +
  xlim(c(0, 50)) +
  theme_bw()
```



Visualization of the CDFs