QMB Exercise 2 - Estimation and Testing

Karin Gryzlak & Dino Nienhold

10 May, 2015

Introduction

The following report is based on the QMB Exercise 2 - Estimation and Testing. The task description pdf file is bis_ex2-EstimationTesting-20150429.pdf

Requirements

Please make sure that you the following packages loaded in your workspace.

```
library("dplyr")
library("ggplot2")
library("ggExtra")
library("gridExtra")
library("moments")
```

Data Set

Please make sure you have the file housingrents.csv in the subdirectoy Data in your workspace.

```
housingrents <- read.csv("./Data/housingrents.csv",sep=";")
```

Data Processing

For analysis purposes it is necessary to convert the rooms and NRE variable to a factor. Furthermore a new variable rps (rent per square meter) is created. Additionally the 2 sub datasets are created for NRE respectively non-NRE appartments.

```
housingrents <- mutate(housingrents, rooms = factor(rooms),
  nre = factor(nre,levels=c(0,1),labels=c("no","yes")))
housingrents <- mutate(housingrents,rps = rent/area)
nrehousing <- filter(housingrents,nre=="yes")
nonnrehousing <- filter(housingrents,nre=="no")
```

Task 1

In this task sample mean, standard error of the mean, conf interval, t-test and probablity is calculated.

a)

For the dataset of 8 repair cases (2.6,12.2,8.3,28.6,0.5,19.0,16.3,5.7) the sample mean and standard error of the mean is calculated.

```
x <- c(2.6, 12.2, 8.3, 28.6, 0.5, 19.0, 16.3, 5.7)
```

```
me <- mean(x)
n <- length(x)
se <- sd(x)/sqrt(n)
```

The mean is 11.65 and the standard error of the mean is 3.316894

b)

In this sub task the normal plot is drawn for the dataset x. In figure 1 the points on this plot form a nearly linear pattern, which indicates that the normal distribution is a good model for this data set.

```
print(gg_qq(x))
```

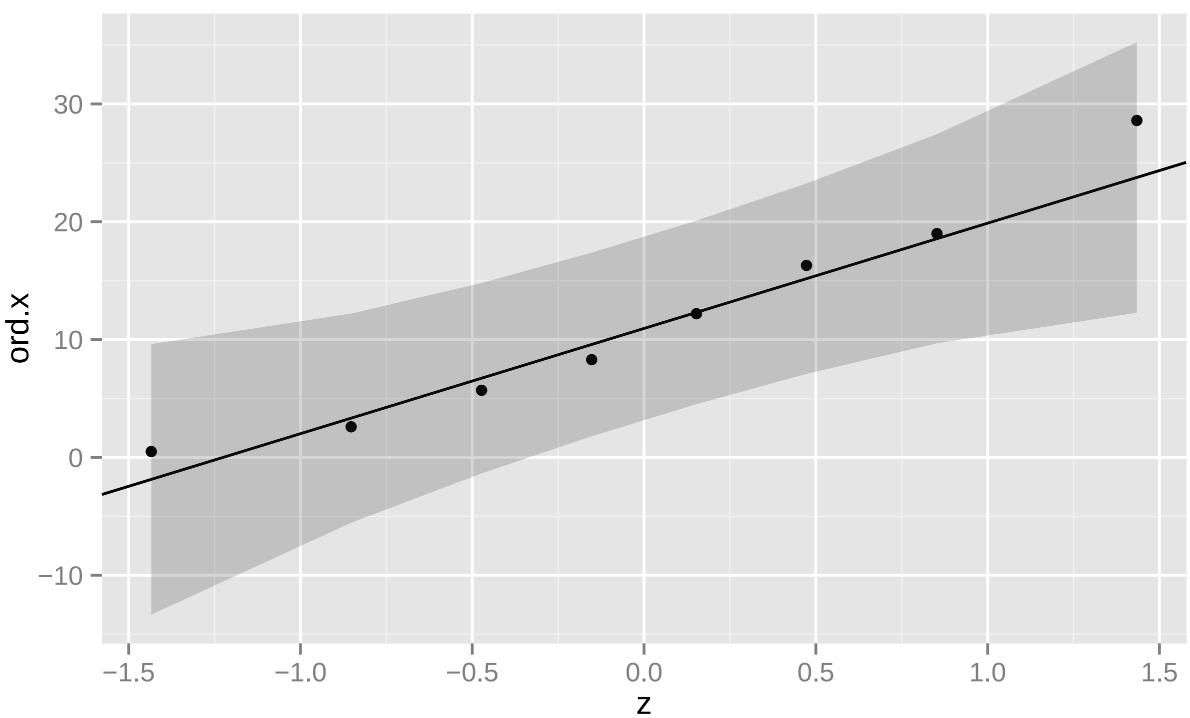

Figure 1: QQ Plot for dataset x

c)

In this sub task a 95% confidence interval is created:

```
me + c(-1,1) * qnorm(.975) * se
```

We are 95% confident that the serice times are between 5.1490072 and 18.1509928.

d)

A t-test is conducted to see if the mean of x is less than four hours.

H_0 = Mean is equal 4.

H_a = Mean is greater than 4.

The p-value is 0.7395827 and much higher than 0.05. The mean difference is 0.7852164 and lies in the 95% confidence interval of the estimated population mean of -2.8088392 and Infinity. Therefore we fail to reject H_0.

```
t.test(x,mu=4, alternative = "greater")
```

```
##
##  One Sample t-test
##
## data:  x
## t = 2.3064, df = 7, p-value = 0.02724
## alternative hypothesis: true mean is greater than 4
## 95 percent confidence interval:
##  5.365884      Inf
## sample estimates:
## mean of x
##     11.65
```

e)

In this sub task the probability the the repair time is larger than 24 hours is calculated.

```
pnorm(24, mean(x), sd(x), lower.tail=FALSE)
```

```
## [1] 0.09401864
```

The probability for a repair time of 24 hours or more is 9.40%.

Task 2

Task 2 checks normality of variable rent per square (rps). Additionally t-tests are conducted

a)

In this sub task the distribution of the variable rps for NRE respectively non-NRE appartments are checked for normality. Figure 2 shows that the distribution is right skewed with several outliers on the higher rps range. Skewness value is 1.9353555. Figure 3 shows a qq plot. The plot does also show the outliers and the non-normality of the distribution.

```
p1 <- qplot(x = 1, y = rps, data = nrehousing, xlab = "", geom = 'boxplot') +
        coord_flip(ylim=c(0,60))


p2 <- ggplot(nrehousing, aes(x = rps)) +
        geom_histogram(colour="black", fill="white") +
        coord_cartesian(xlim=c(0,60)) +
        scale_y_continuous(breaks=seq(0,10,2))

grid.arrange(p1, p2, widths = c(1, 2))
```

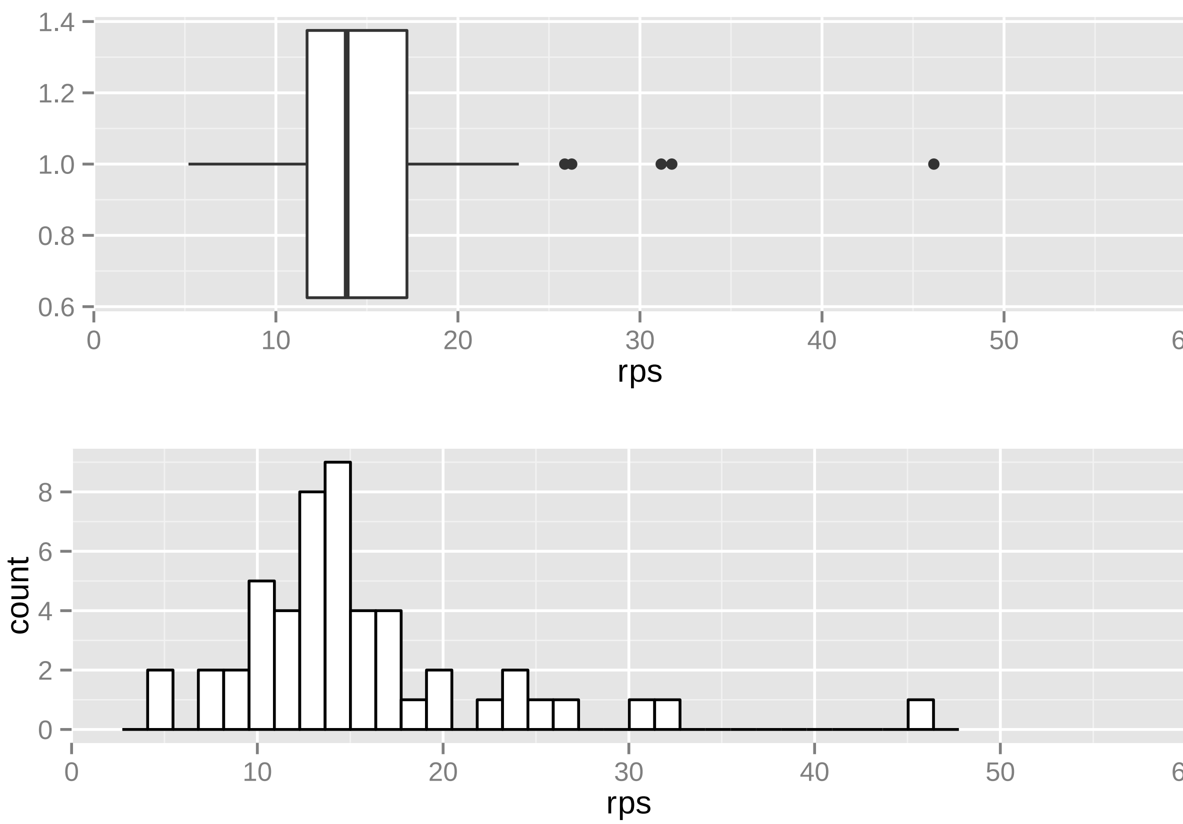

Figure 2: Boxplot and Histogramm for variable rps for NRE appartments

```
print(gg_qq(nrehousing$rps))
```

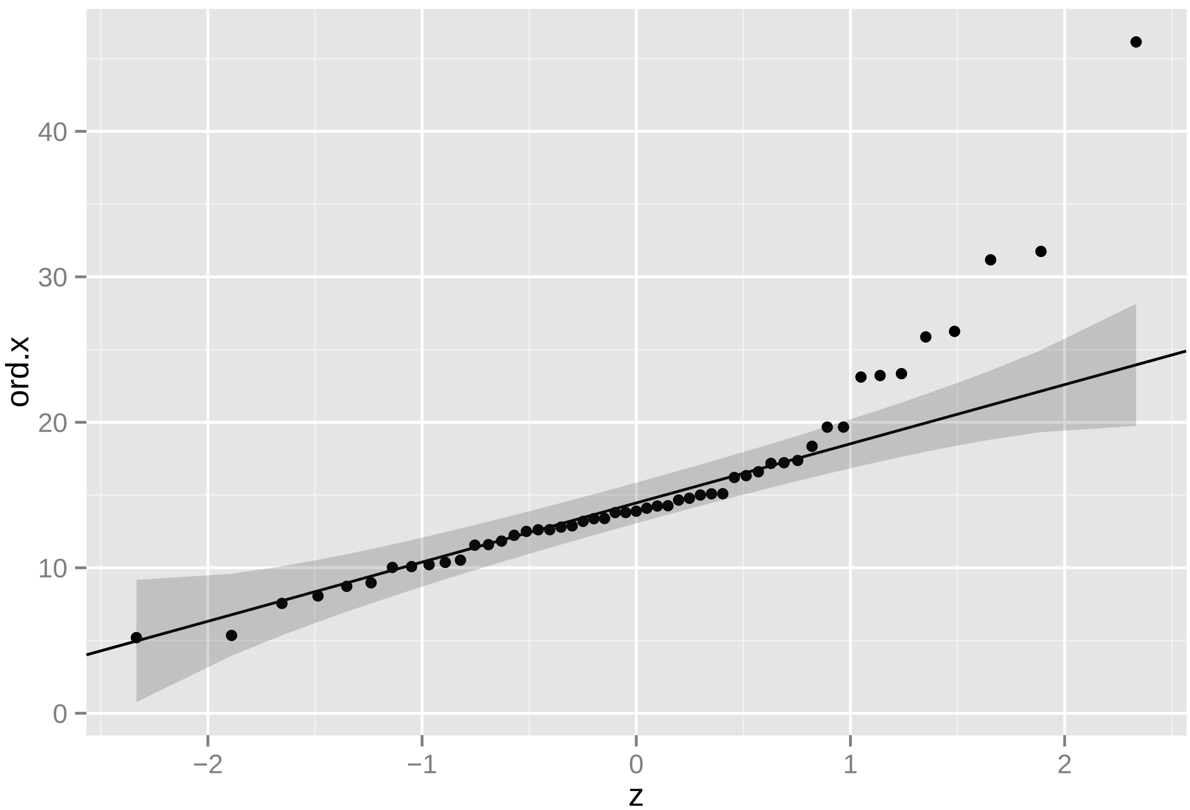

Figure 3: QQ Plot for variable rps for NRE appartments

Check normal distribution of rps variable for non-NRE appartments: Figure 4 shows that the distribution is right skewed with several outliers on the higher rps range. Skewness value is 1.9708836. Figure 5 shows a qq plot. The plot does also show the outliers and the non-normality of the distribution.

```
p1 <- qplot(x = 1, y = rps, data = nonnrehousing, xlab = "", geom = 'boxplot') +
        coord_flip(ylim=c(0,60))

p2 <- ggplot(nonnrehousing, aes(x = rps)) +
        geom_histogram(colour="black", fill="white") +
        coord_cartesian(xlim=c(0,60)) +
        scale_y_continuous(breaks=seq(0,10,2))

grid.arrange(p1, p2, widths = c(1, 2))
```

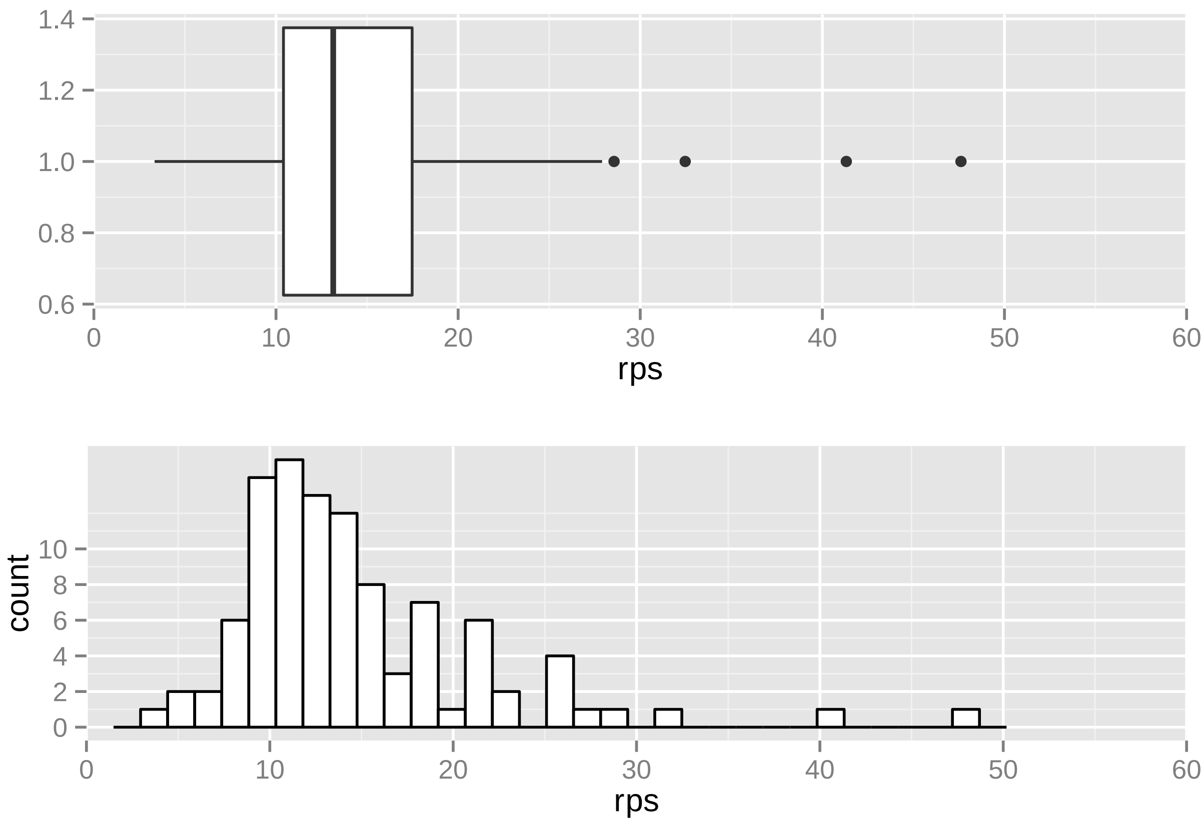

Figure 4: Boxplot and Histogramm for variable rps for non-NRE appartments

```
print(gg_qq(nonnrehousing$rps))
```

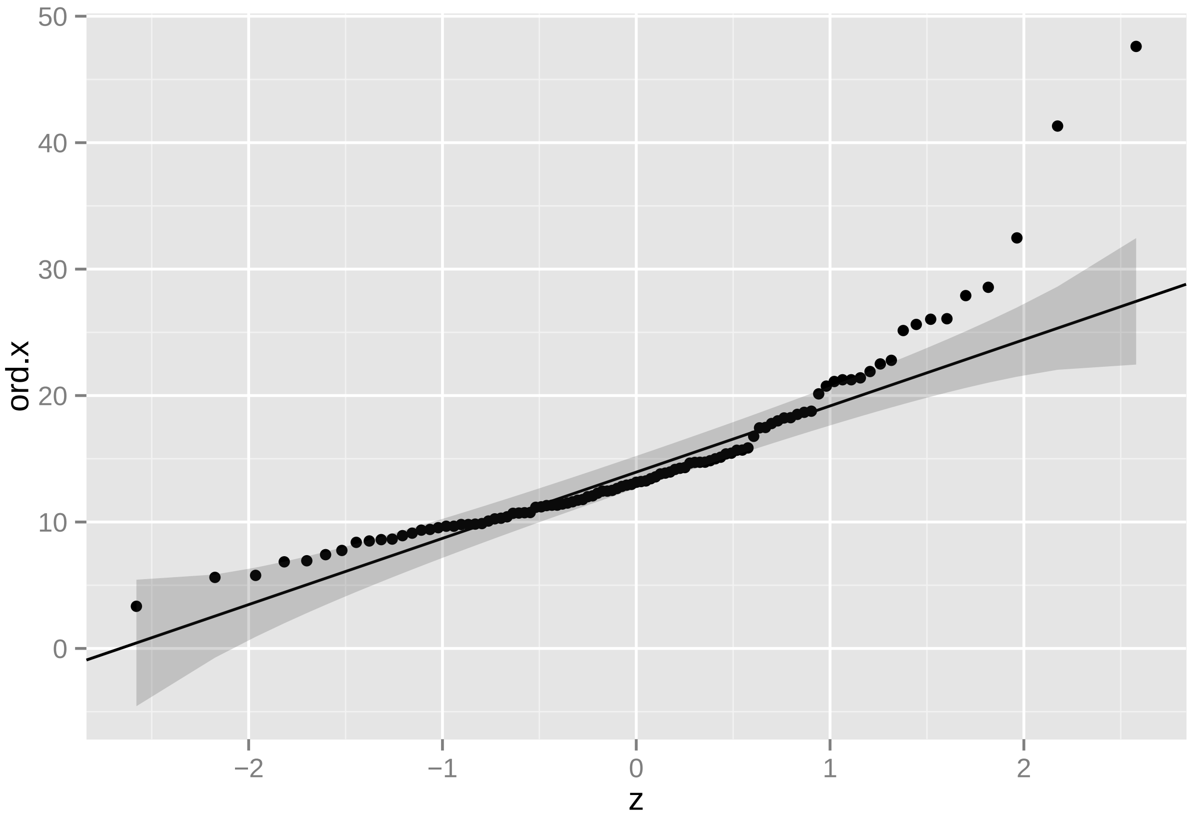

Figure 5: QQ Plot for variable rps for non-NRE appartments

b)

In this section a two-sided Student's t-test is conducted in order to check the following hypothesis:

H_0 = Mean diference of the variable rps for NRE and non-NRE appartments is equal 0.

H_a = Mean diference of the variable rps for NRE and non-NRE appartmentsis not equal 0.

In the first t-test equal variance is not assumed. The p-value is 0.5208345 and much higher than 0.05. The mean difference is 0.7852164 and lies in the 95% confidence interval of the estimated population mean of -3.2036613 and 1.6332286 . Therefore we fail to reject H_0.

```
t.test(housingrents$rps~housingrents$nre,alternative = "two.sided", mu=0, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  housingrents$rps by housingrents$nre
## t = -0.64441, df = 96.874, p-value = 0.5208
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.203661  1.633229
## sample estimates:
##  mean in group no mean in group yes
##          14.77912          15.56434
```

In the second t-test equal variance is assumed. The p-value is 0.5146434 and much higher than 0.05. The mean difference is 0.7852164 and lies in the 95% confidence interval of the estimated population mean of -3.160557 and 1.5901243 . Therefore we fail to reject H_0.

```
t.test(housingrents$rps~housingrents$nre,alternative = "two.sided", mu=0, var.equal = TRUE)
```

```
##
##  Two Sample t-test
##
## data:  housingrents$rps by housingrents$nre
## t = -0.65318, df = 150, p-value = 0.5146
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.160557  1.590124
## sample estimates:
##  mean in group no mean in group yes
##          14.77912          15.56434
```

c)

In this section a one-sided Student's t-test is conducted in order to check the following hypothesis:

H_0 = Mean diference of the variable rps for NRE and non-NRE appartments is equal 0.

H_a = Mean diference of the variable rps for NRE and non-NRE appartments isgreater 0.

In the second t-test equal variance is assumed. The p-value is 0.7395827 and much higher than 0.05. The mean difference is 0.7852164 and lies in the 95% confidence interval of the estimated population mean of -2.8088392 and Infinity. Therefore we fail to reject H_0.

```
t.test(housingrents$rps~housingrents$nre,alternative = "greater", mu=0, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  housingrents$rps by housingrents$nre
## t = -0.64441, df = 96.874, p-value = 0.7396
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  -2.808839       Inf
## sample estimates:
##  mean in group no mean in group yes
##          14.77912          15.56434
```

Task 3

In this task we conduct chi-square test of indepedence to test if the variable rooms is indepent from the variable nr.

a)

In this subtask the chi-square test is conducted.

```
housingrentTbl <- xtabs(~rooms+nre, data=housingrents)
```

```
chisq.test(housingrentTbl)
```

```
## Warning in chisq.test(housingrentTbl): Chi-squared approximation may be
## incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  housingrentTbl
## X-squared = 26.749, df = 5, p-value = 6.384e-05
```

As the expected chi-square test value for the six room appartments is below 5. The chi-square test computation of p values is done by Monte Carlo simulation.

```
chisq.test(housingrentTbl,simulate.p.value=TRUE)
```

```
##
##  Pearson's Chi-squared test with simulated p-value (based on 2000
##  replicates)
##
## data:  housingrentTbl
## X-squared = 26.749, df = NA, p-value = 0.0004998
```

The p value 0.000499 is and below 0.01. Therefore we reject the H_0 and there is a strong evidence that the variables are not independent.

b)

The residuals for the chi-square test are evaluated in this sub task.

```
resid(chisq.test(housingrentTbl,simulate.p.value=TRUE))
```

```
##      nre
## rooms         no        yes
##     1  0.6950302 -0.9780910
##     2  0.8497704 -1.1958513
##     3  1.4953421 -2.1043413
##     4 -1.5751126  2.2165994
##     5 -1.7397058  2.4482255
##     6  0.1616756 -0.2275202
```

The 3, 4 and 5 room appartments belonging to NRE have the lowest respectively lowest residuals. This means that they have the highest impact on the test.

Appendix

Functions

Plot qqplot with ggplot

```
gg_qq <- function(x, distribution = "norm", ..., line.estimate = NULL, conf = 0.95,
                  labels = names(x)){
        q.function <- eval(parse(text = paste0("q", distribution)))
        d.function <- eval(parse(text = paste0("d", distribution)))
        x <- na.omit(x)
        ord <- order(x)
        n <- length(x)
        P <- ppoints(length(x))
        df <- data.frame(ord.x = x[ord], z = q.function(P, ...))

        if(is.null(line.estimate)){
                Q.x <- quantile(df$ord.x, c(0.25, 0.75))
                Q.z <- q.function(c(0.25, 0.75), ...)
                b <- diff(Q.x)/diff(Q.z)
                coef <- c(Q.x[1] - b * Q.z[1], b)
        } else {
                coef <- coef(line.estimate(ord.x ~ z))
        }

        zz <- qnorm(1 - (1 - conf)/2)
        SE <- (coef[2]/d.function(df$z)) * sqrt(P * (1 - P)/n)
        fit.value <- coef[1] + coef[2] * df$z
        df$upper <- fit.value + zz * SE
        df$lower <- fit.value - zz * SE

        if(!is.null(labels)){
                df$label <- ifelse(df$ord.x > df$upper |
                                           df$ord.x < df$lower, labels[ord],"")
        }

        p <- ggplot(df, aes(x=z, y=ord.x)) +
                geom_point() +
                geom_abline(intercept = coef[1], slope = coef[2]) +
                geom_ribbon(aes(ymin = lower, ymax = upper), alpha=0.2)
        if(!is.null(labels)) p <- p + geom_text( aes(label = label))
        return(p)
       }
```