

CW_2 GY7702 R for Data Science

Student Number: 219031729

1/5/2022

Introduction

GY7702 R for Data Science Course Work 2

```
library(dplyr)
library(tidyverse)
library(knitr)
library(readr)
library(lubridate)
library(ggplot2)
library(psych)
library(Hmisc)
library(corrplot)
library(PerformanceAnalytics)
library(car)
library(magrittr)
library(lmtest)
```

1.0 Loading and selecting data needed for analysis

Note: The variables for this analysis are encoded, see appendix 2 for full description of the variables.

#Loading in data for analysis

```
OAC_Raw_uVariables_2011 <-
```

```
  read.csv("/home/kal41/Practical_204/CW1/GY7702_2021-22_Assignment_2_v1-1_datapack/2011_OAC_Raw_uVariables_2011.csv")
```

#Loading data that would be used to extract my Output Area

```
LAD_Allocation_data <- read.csv("/home/kal41/Practical_204/CW1/GY7702_2021-22_Assignment_2_v1-1_datapack/LAD_Allocation_data.csv")
```

#Filtering out my allocated LAD

```
LAD_Allocation_data <- LAD_Allocation_data %>%
  filter(LAD11CD == "E09000006")
```

#Joining the two data to select my allocated Output Area only

```
OwnLadd <- LAD_Allocation_data %>%
  left_join(
    OAC_Raw_uVariables_2011, by = c("OA11CD" = "OA")
  ) %>%
  select(- c(LSOA11CD, LSO11ANM, MSOA11CD, MSOA11NM, LAD11CD, LAD11NM, LAD11NMW))
```

#Selecting variables needed for Analysis

```
explorData <- OwnLadd %>%
  select( u104:u115, u159:u167)
```

1.1 Exploratory analysis of the data

```
describe(explorData,skew=TRUE, IQR = TRUE)
```

```
## explorData
##
## 21 Variables      1020 Observations
## -----
## u104
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      1016        1     82.87     31.17     45.69     51.67
##      .25      .50      .75      .90      .95
##    62.18     77.95     97.54    119.40    135.42
##
## lowest : 25.70986 27.55359 27.58909 28.14020 29.37061
## highest: 197.17663 206.23959 210.10168 221.69228 258.17275
## -----
## u105
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      195        1      152     45.73     87.95    102.00
##      .25      .50      .75      .90      .95
##   125.00    149.50    178.00    203.00    217.05
##
## lowest : 8 37 45 48 56, highest: 279 282 285 303 332
## -----
## u106
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      122        1     103.5     24.76     68.0     75.9
##      .25      .50      .75      .90      .95
##     89.0    103.5    117.0    130.0    139.0
##
## lowest : 35 36 43 44 46, highest: 176 180 184 188 208
## -----
## u107
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      64      0.999     35.39     12.07     19.95     23.00
##      .25      .50      .75      .90      .95
##    28.00     34.00     42.00     49.00     53.05
##
## lowest : 9 10 11 12 13, highest: 76 77 78 90 93
## -----
## u108
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      34     0.996     9.71     5.816      3      4
##      .25      .50      .75      .90      .95
##      6      9      13      17      20
##
## lowest : 0 1 2 3 4, highest: 30 31 32 36 37
## -----
## u109
```

```

##          n missing distinct      Info      Mean      Gmd      .05      .10
##      1020         0        20    0.969     2.763     2.454         0         0
##          .25      .50      .75      .90      .95
##           1         2         4         6         7
##
## lowest :  0  1  2  3  4, highest: 16 17 21 22 23
##
## Value          0         1         2         3         4         5         6         7         8         9        10
## Frequency      112      237      225      169      107       64       38       24       13         9         6
## Proportion 0.110 0.232 0.221 0.166 0.105 0.063 0.037 0.024 0.013 0.009 0.006
##
## Value          11         12         13         15         16         17         21         22         23
## Frequency         5         2         2         1         2         1         1         1         1
## Proportion 0.005 0.002 0.002 0.001 0.002 0.001 0.001 0.001 0.001
## -----
## u110
##          n missing distinct      Info      Mean      Gmd      .05      .10
##      1020         0         61    0.999     30.4     11.91        15        17
##          .25      .50      .75      .90      .95
##          23       30       38       44       48
##
## lowest :  3  4  6  7  8, highest: 61 62 65 68 70
## -----
## u111
##          n missing distinct      Info      Mean      Gmd      .05      .10
##      1020         0         95         1     43.7     21.22     18.95     22.00
##          .25      .50      .75      .90      .95
##      29.00     40.00     56.00     70.00     81.00
##
## lowest :   4   5   7   8   9, highest: 103 106 107 115 118
## -----
## u112
##          n missing distinct      Info      Mean      Gmd      .05      .10
##      1020         0        114         1     79.07     24.34     43.95     50.00
##          .25      .50      .75      .90      .95
##      66.00     79.00     93.00    106.00    116.05
##
## lowest :  19  23  24  25  30, highest: 138 140 143 144 155
## -----
## u113
##          n missing distinct      Info      Mean      Gmd      .05      .10
##      1020         0         49    0.998     28.08     9.239        15        18
##          .25      .50      .75      .90      .95
##          22       28       33       39       42
##
## lowest :   6   7   9  10  11, highest: 51 52 53 57 58
## -----
## u114
##          n missing distinct      Info      Mean      Gmd      .05      .10
##      1020         0        145         1     80.69     35.13     31.95     39.00
##          .25      .50      .75      .90      .95
##      58.00     80.00    100.00    121.00    134.00
##
## lowest :  15  17  18  19  20, highest: 176 179 194 204 221

```

```

## -----
## u115
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      47    0.998    15.33    7.732      5      7
##      .25      .50      .75      .90      .95
##      10      15      20      24      28
##
## lowest : 0 1 2 3 4, highest: 47 54 56 58 59
## -----
## u159
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      51    0.999    19.4    9.919      7      9
##      .25      .50      .75      .90      .95
##      13      19      25      30      35
##
## lowest : 0 2 3 4 5, highest: 51 52 56 59 61
## -----
## u160
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      74    0.999    31.76    14.99     12     15
##      .25      .50      .75      .90      .95
##      22      31      40      49      55
##
## lowest : 0 2 3 4 5, highest: 75 77 78 87 90
## -----
## u161
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      54    0.999    24.01    10.52      9     12
##      .25      .50      .75      .90      .95
##      17      23      30      36      40
##
## lowest : 0 3 4 5 6, highest: 52 57 58 60 65
## -----
## u162
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      45    0.998    22.63     8.352     11     13
##      .25      .50      .75      .90      .95
##      18      22      27      33      36
##
## lowest : 2 3 4 5 6, highest: 42 43 44 45 51
## -----
## u163
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      37    0.997    13.68     6.605      5      7
##      .25      .50      .75      .90      .95
##      9      13      17      21      24
##
## lowest : 1 2 3 4 5, highest: 34 35 37 38 43
## -----
## u164
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      31    0.996    11.32     5.796      4      5
##      .25      .50      .75      .90      .95
##      8      11      15      18      20

```

```
##
## lowest : 1 2 3 4 5, highest: 27 29 30 33 35
## -----
## u165
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      28    0.995    9.994    5.044    4.0    4.9
##      .25      .50      .75      .90      .95
##      7.0      9.0     13.0     16.0     18.0
##
## lowest : 0 1 2 3 4, highest: 23 24 25 27 33
## -----
## u166
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      19    0.992    6.174    3.86      1      2
##      .25      .50      .75      .90      .95
##      4       6       8       11      13
##
## lowest : 0 1 2 3 4, highest: 14 15 16 17 18
##
## Value      0      1      2      3      4      5      6      7      8      9     10
## Frequency   19     39     85    107    117    112    117    85    95    69    58
## Proportion 0.019 0.038 0.083 0.105 0.115 0.110 0.115 0.083 0.093 0.068 0.057
##
## Value      11     12     13     14     15     16     17     18
## Frequency   42     23     18     17      9      3      2      3
## Proportion 0.041 0.023 0.018 0.017 0.009 0.003 0.002 0.003
## -----
## u167
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    1020      0      33    0.996    9.441    6.086    2.00    3.00
##      .25      .50      .75      .90      .95
##      5.00     9.00    12.25    17.00    20.00
##
## lowest : 0 1 2 3 4, highest: 28 29 30 31 35
## -----
```

For each variable, the function `describe()` in 1.1 shows the number of observation in the `explorData` dataframe; the number of missing observations; the number of distinct observation; how continuous the data is; the mean; the Gini's mean difference (GMD) which shows the data variability and underlying distribution (the mean absolute difference between variables); the percentile (5th, 10th, 25th, 50th, 75th, 90th, 95th); the five lowest and five highest. The statistics shows that all the variables have no missing value, and non of the variables have 100% distinct observations.

1.11 Showing the structure of the data

```
str(explorData) %>%
  knitr::kable()
```

```
## 'data.frame': 1020 obs. of 21 variables:
## $ u104: num 77.7 52.4 155.3 115.1 103.5 ...
## $ u105: int 135 212 131 191 128 104 173 200 132 154 ...
## $ u106: int 70 119 107 111 76 71 146 105 98 94 ...
## $ u107: int 26 30 51 50 44 25 46 30 54 35 ...
## $ u108: int 4 4 14 14 14 6 15 4 14 9 ...
## $ u109: int 1 3 7 17 6 3 7 0 1 5 ...
```

```
## $ u110: int 23 32 39 32 41 26 38 27 26 34 ...
## $ u111: int 17 21 67 57 50 36 45 23 58 33 ...
## $ u112: int 43 72 70 94 64 61 82 70 76 72 ...
## $ u113: int 20 41 23 38 15 15 38 48 27 28 ...
## $ u114: int 104 156 90 108 80 51 134 119 112 105 ...
## $ u115: int 11 24 22 23 10 12 20 24 4 12 ...
## $ u159: int 20 45 18 22 19 14 35 49 19 42 ...
## $ u160: int 52 42 31 47 20 22 32 45 34 39 ...
## $ u161: int 11 38 23 28 25 13 34 32 42 17 ...
## $ u162: int 18 22 14 21 14 10 25 30 24 27 ...
## $ u163: int 12 6 5 11 9 16 6 5 20 10 ...
## $ u164: int 1 9 12 11 9 5 6 7 11 5 ...
## $ u165: int 5 10 8 9 4 8 9 2 9 6 ...
## $ u166: int 3 3 3 6 6 6 4 1 4 5 ...
## $ u167: int 5 4 3 10 7 5 2 4 12 2 ...
```

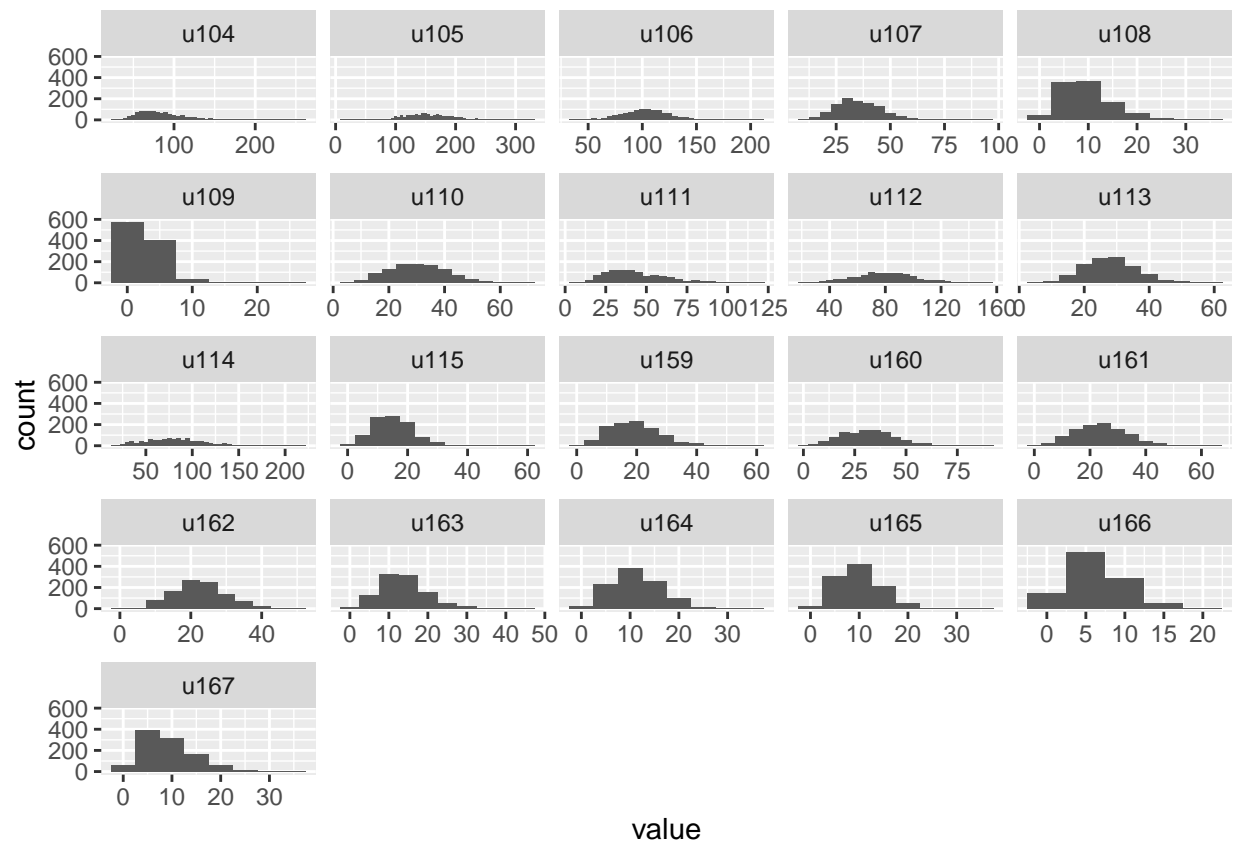
```
|| || || ||
```

This shows that the data is a dataframe which has **21 variables and 1020 observations** All the variables are of integer types except variable u104(Day-to-day activities).

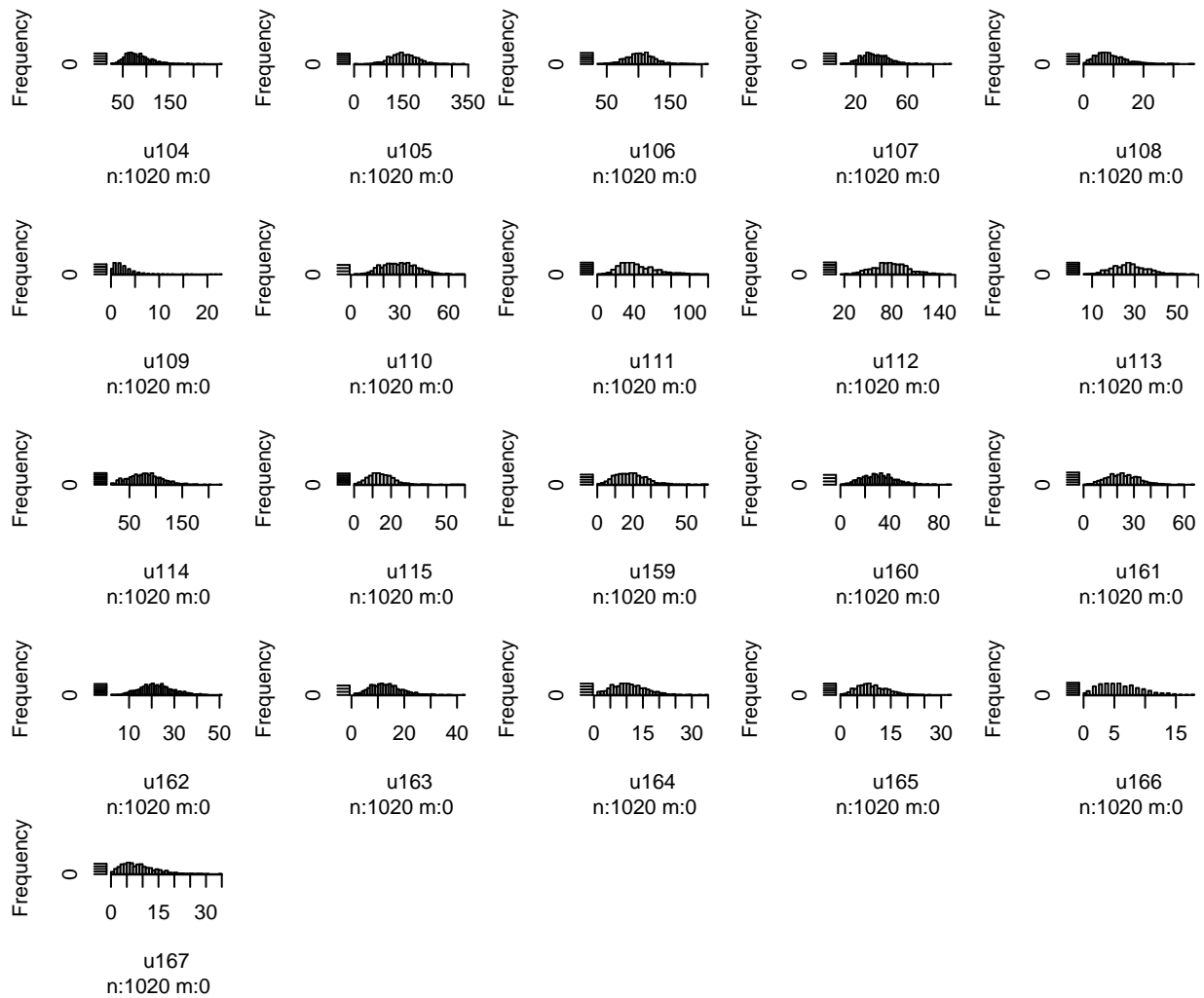
1.12 Visualizing the distribution of the data with Histogram and QQ plot

```
par(mar=c(5,5,3,0)) ##This margin command should do the trick

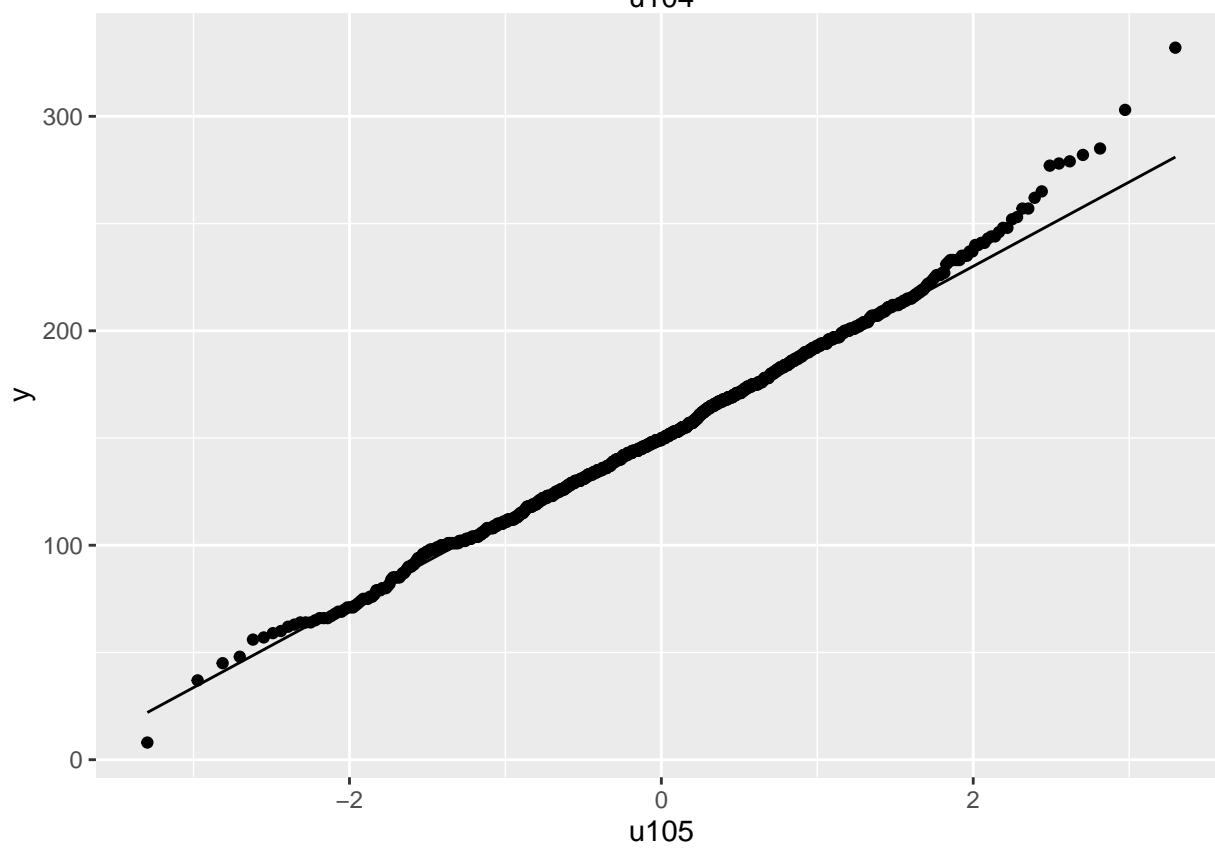
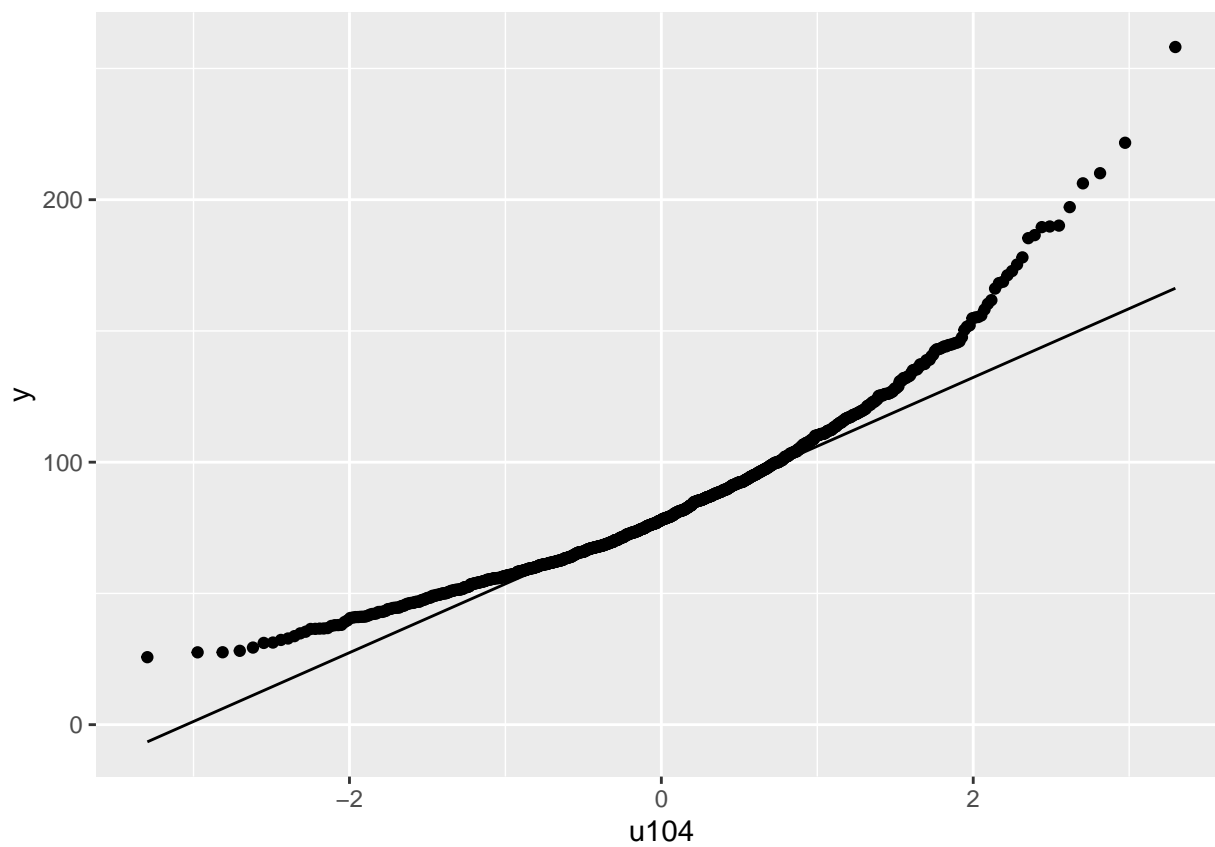
explorData %>% gather() %>%
  ggplot2::ggplot(
    aes(
      x = value
    )
  )+
  ggplot2:: geom_histogram(binwidth = 5) +
  facet_wrap(~key, scales = 'free_x')
```

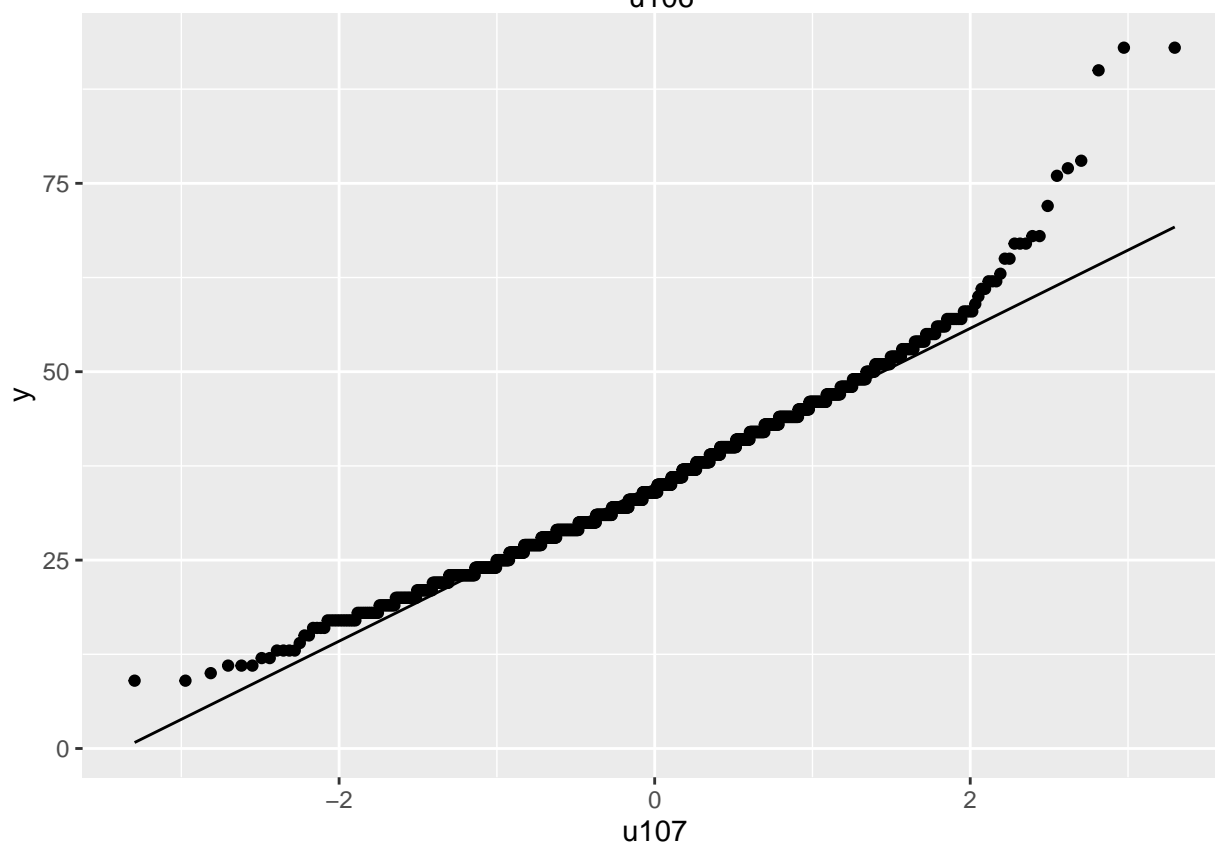
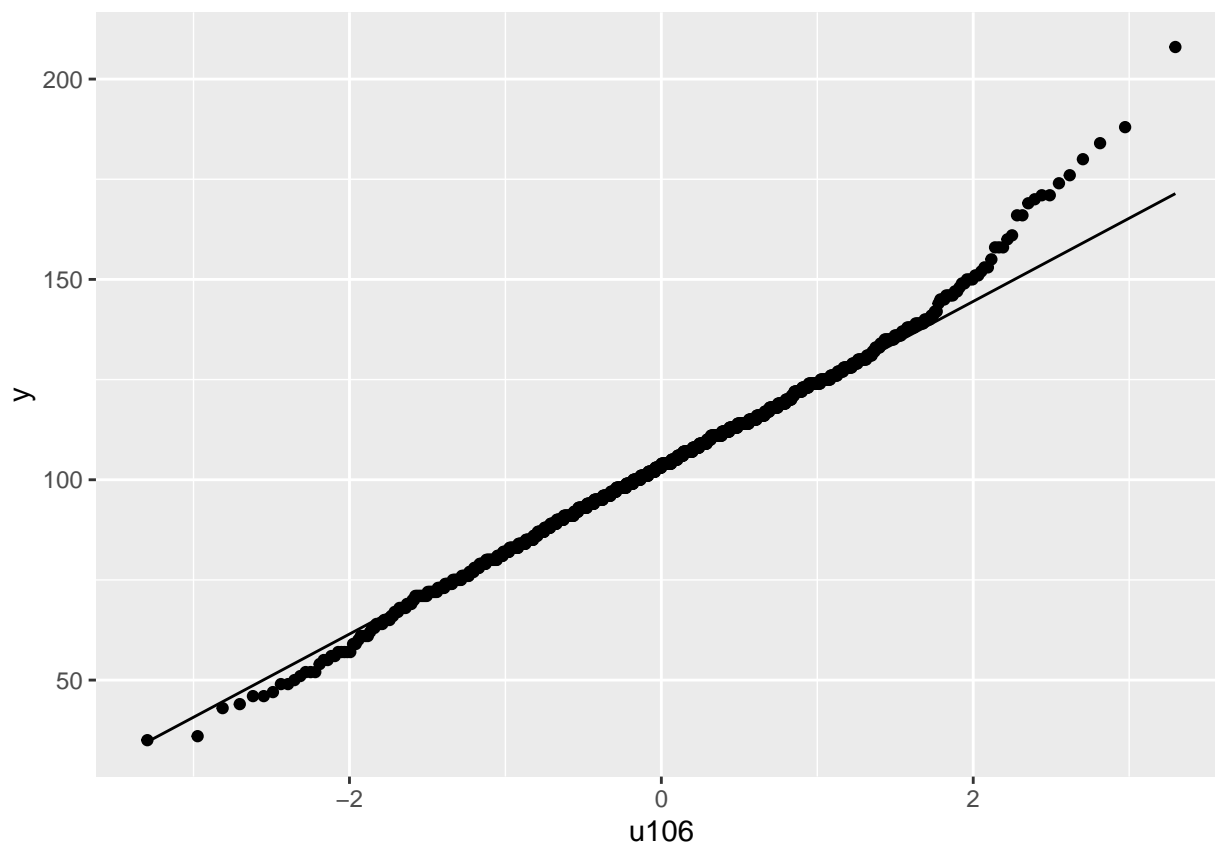


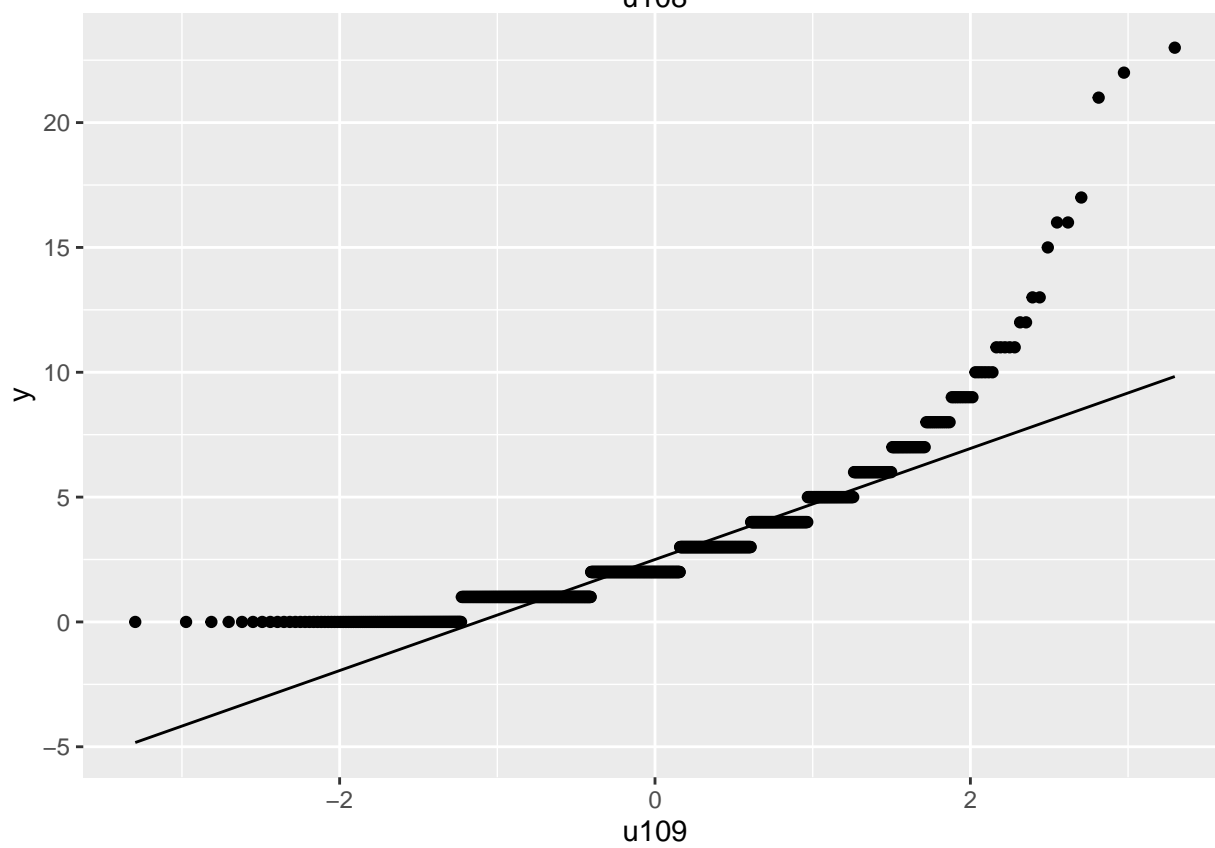
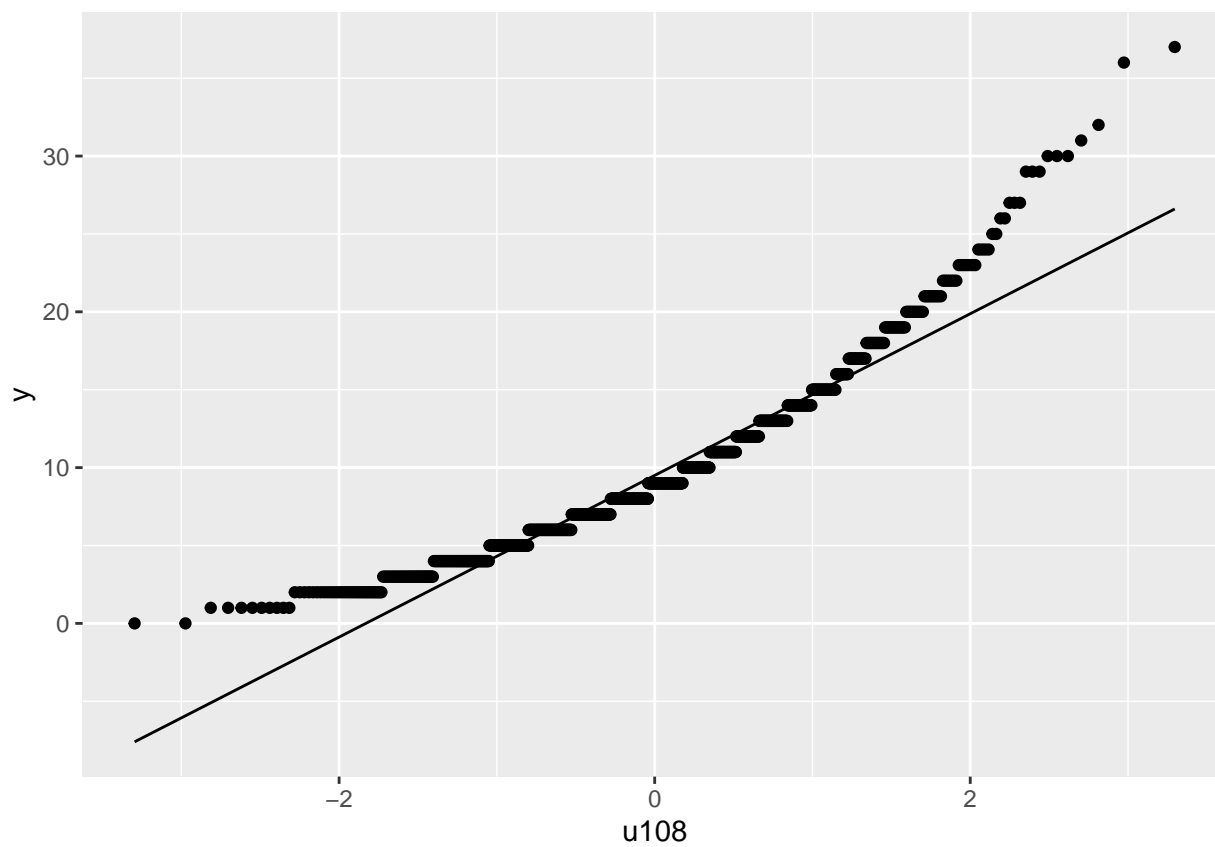
```
hist.data.frame(explorData)
```

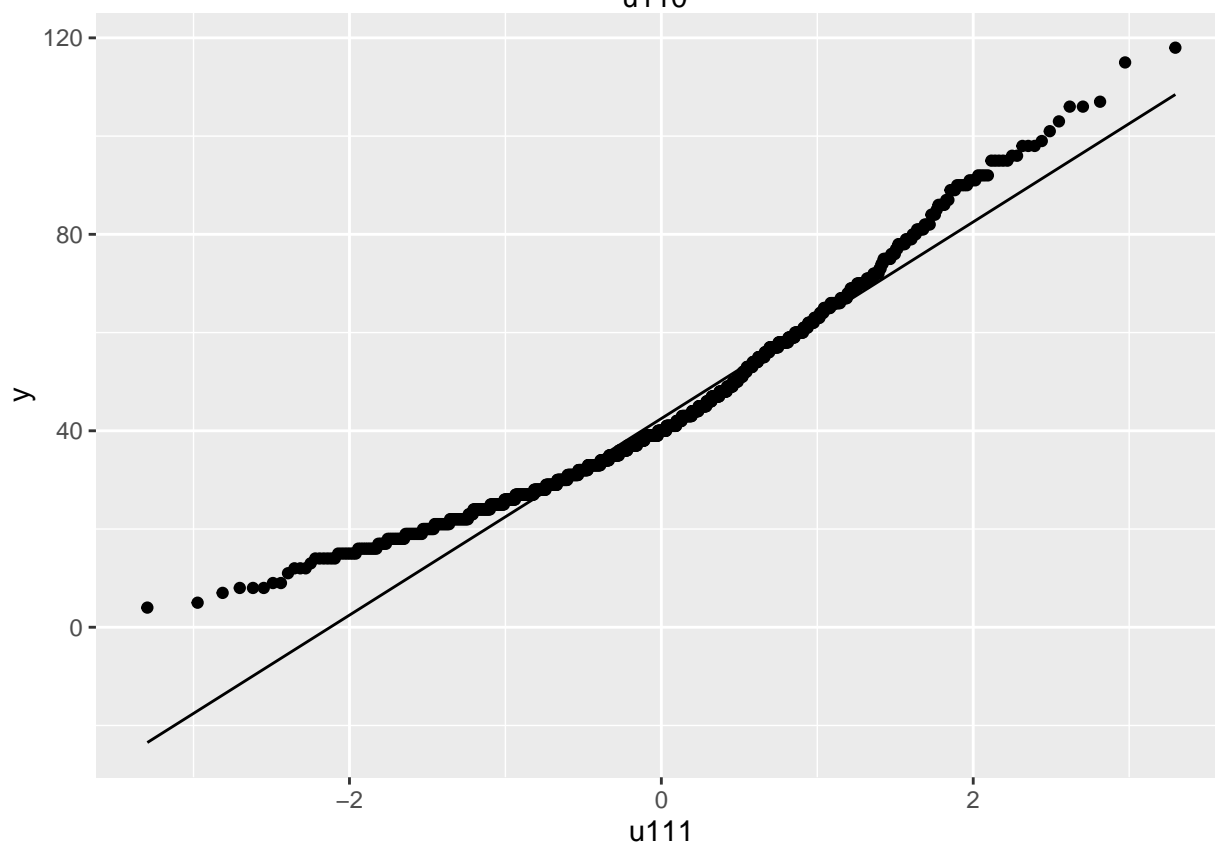
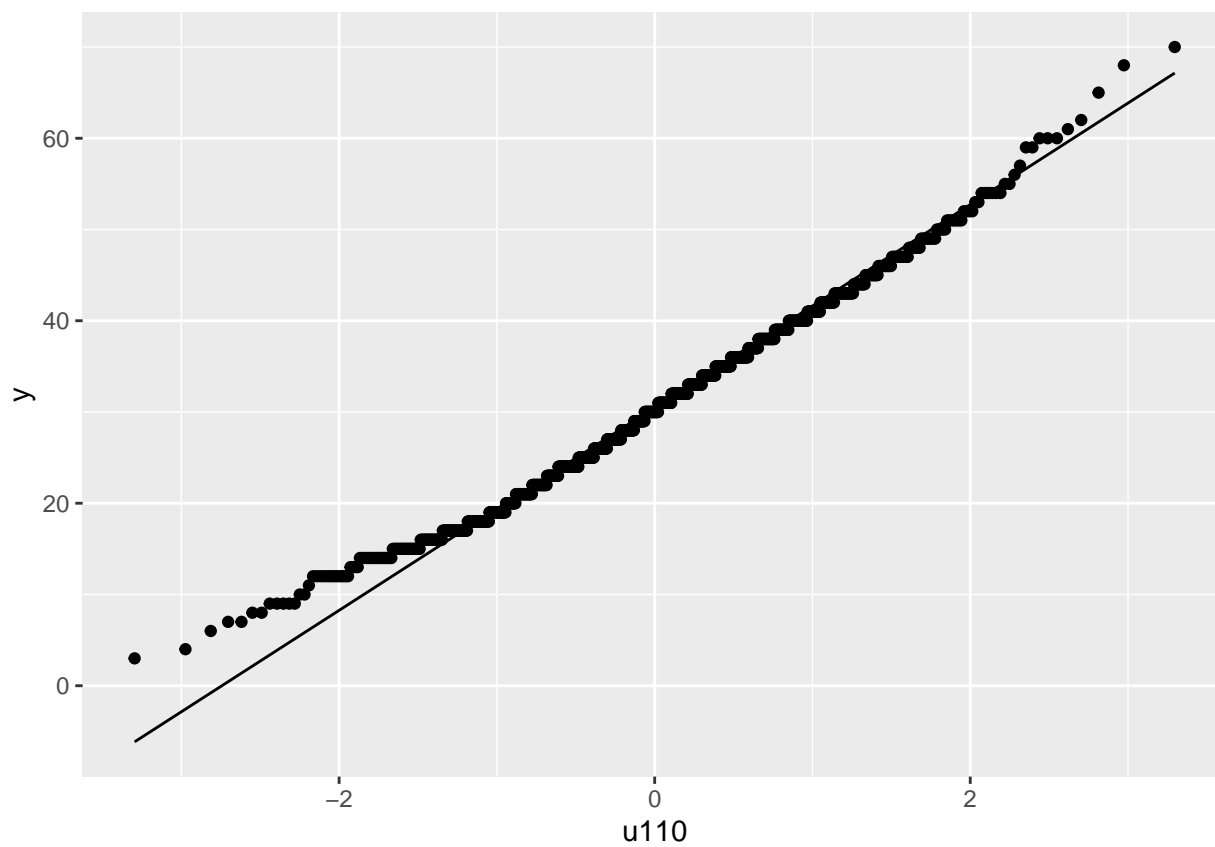


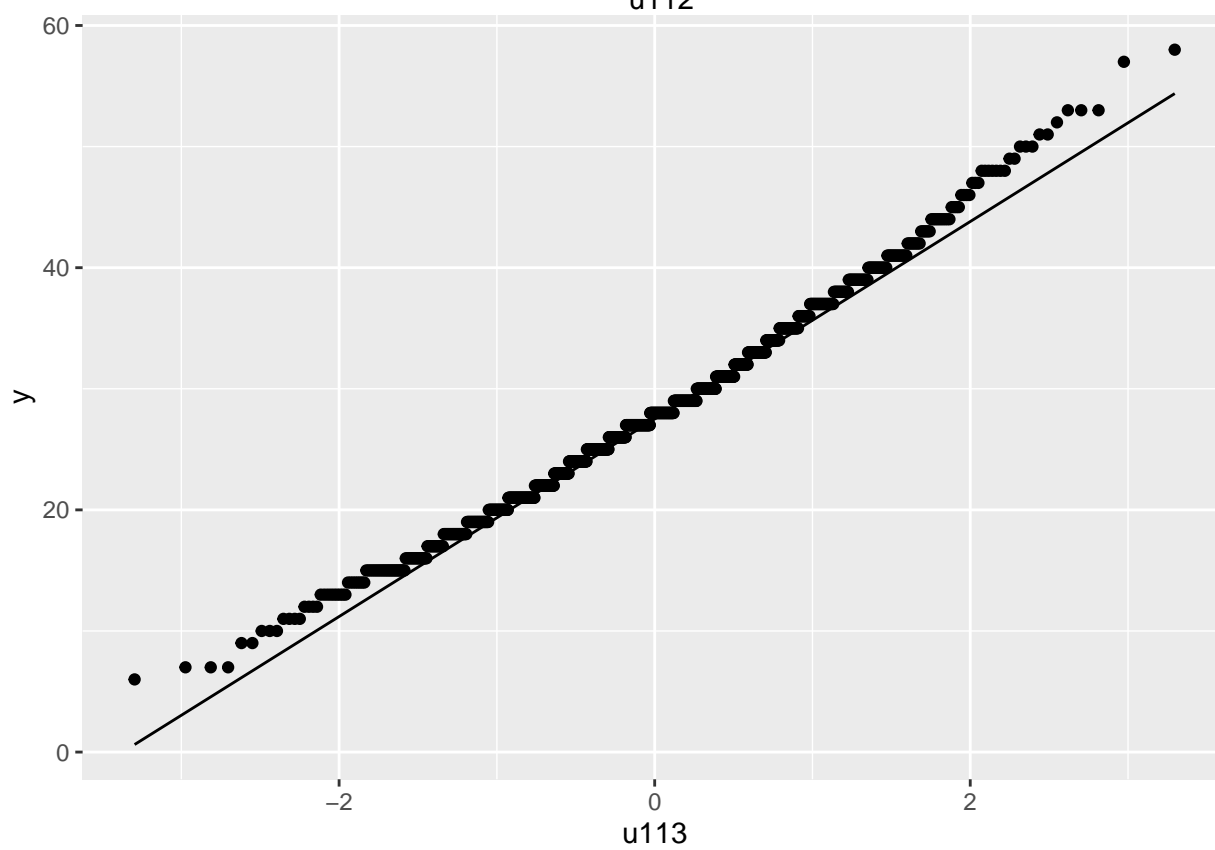
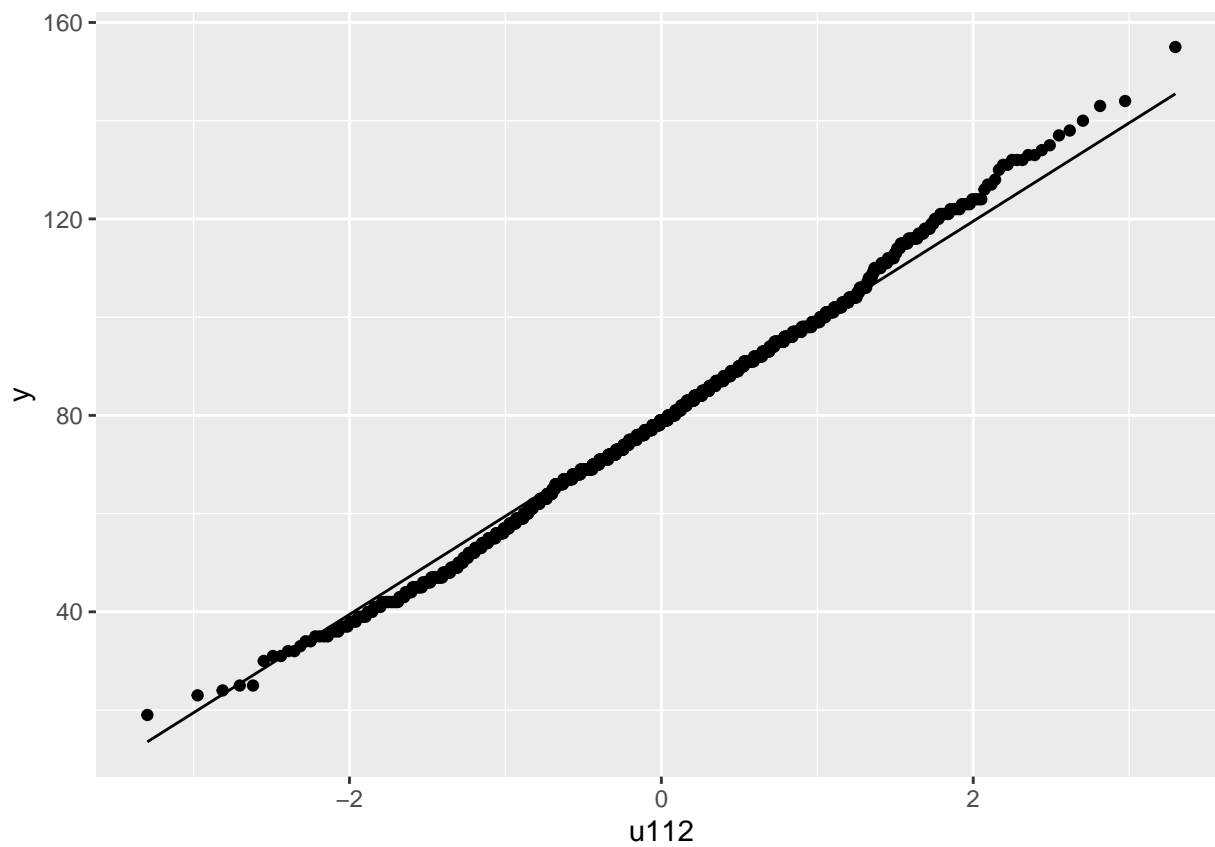
```
for (i in 1:ncol(explorData)) {
  plt <- ggplot2::ggplot(explorData,
    aes(
      sample = explorData[,i]
    )
  ) +
    ggplot2::stat_qq() +
    ggplot2::stat_qq_line()+
    ggplot2::xlab(colnames( explorData[i]))
  print(plt)
}
```

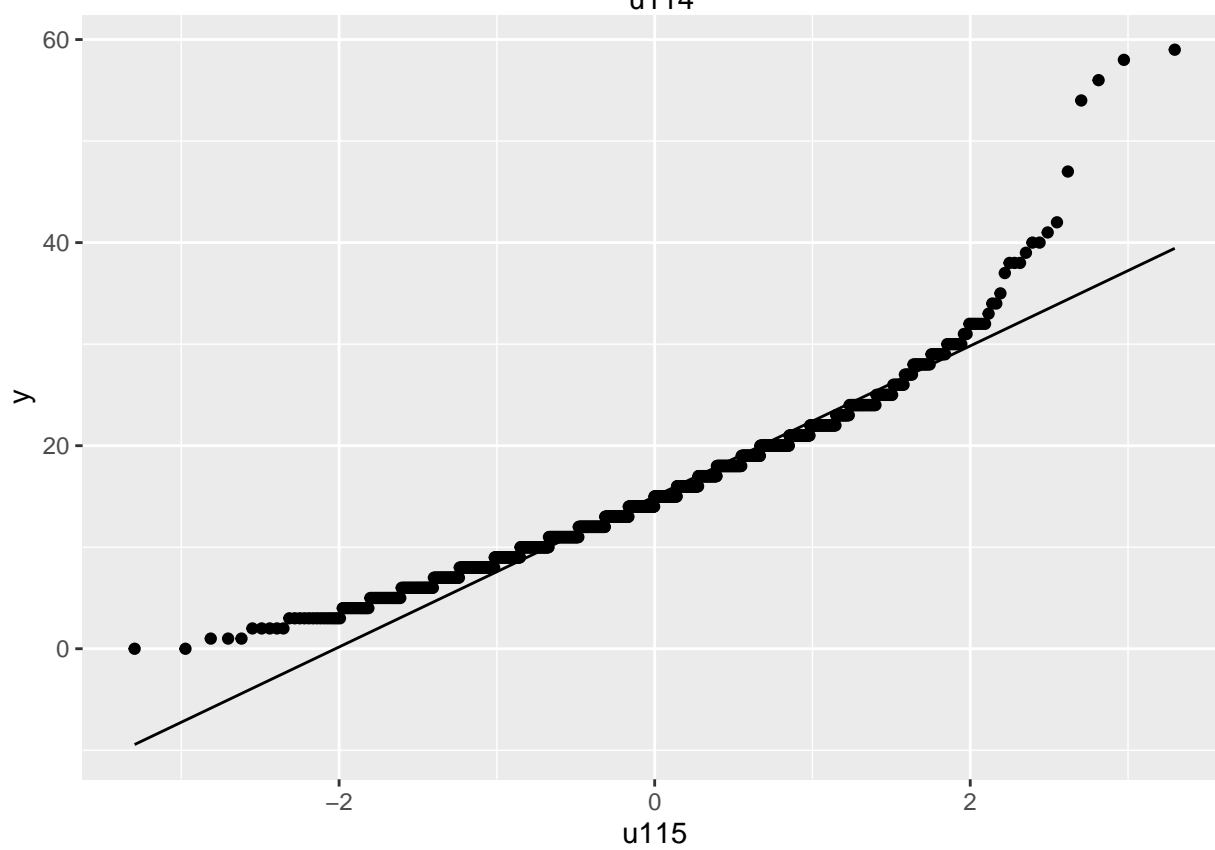
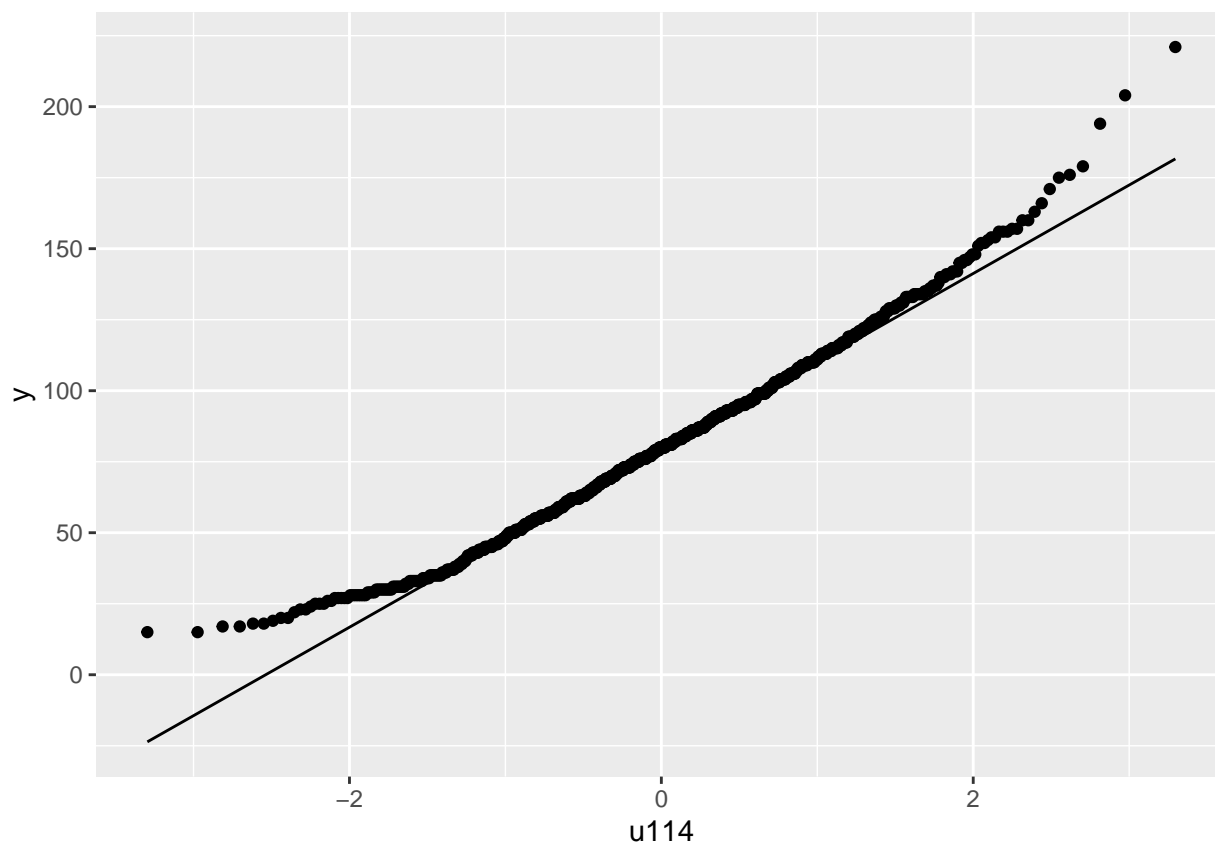



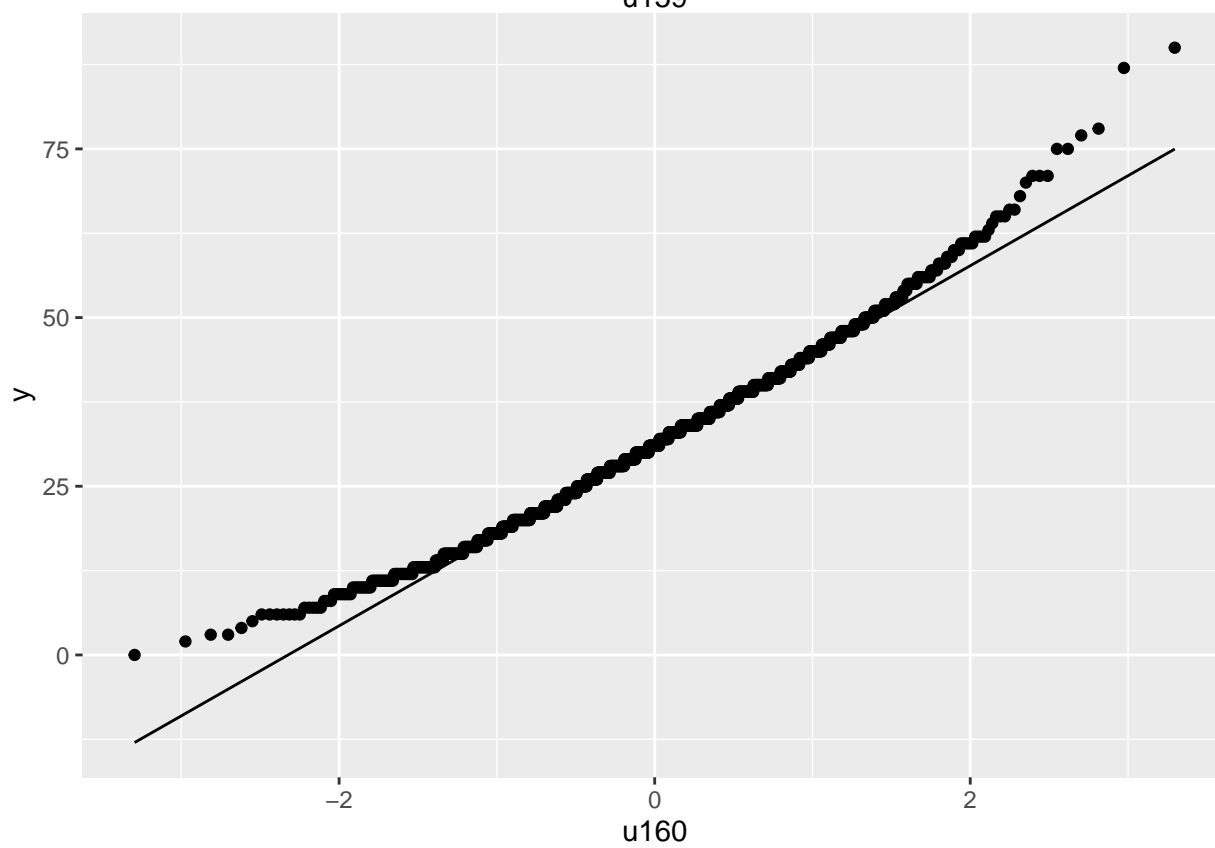
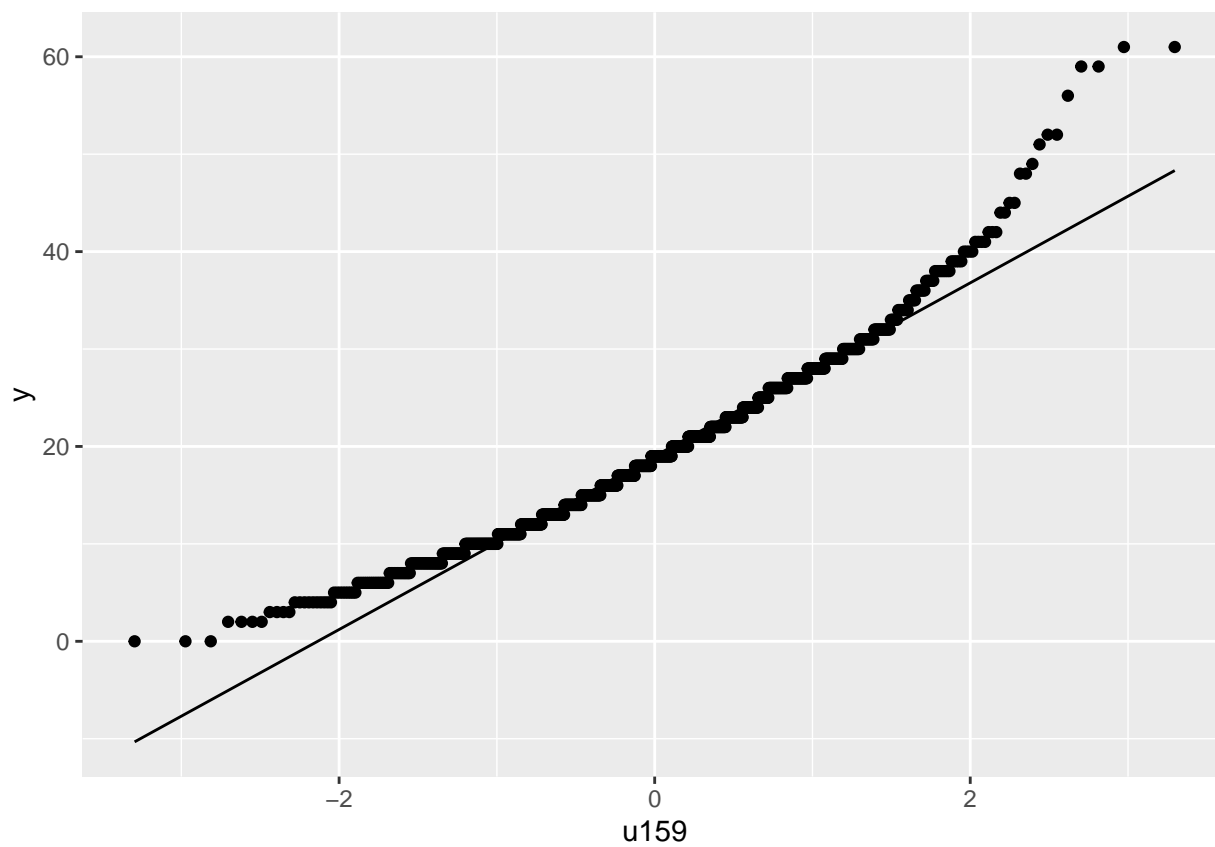


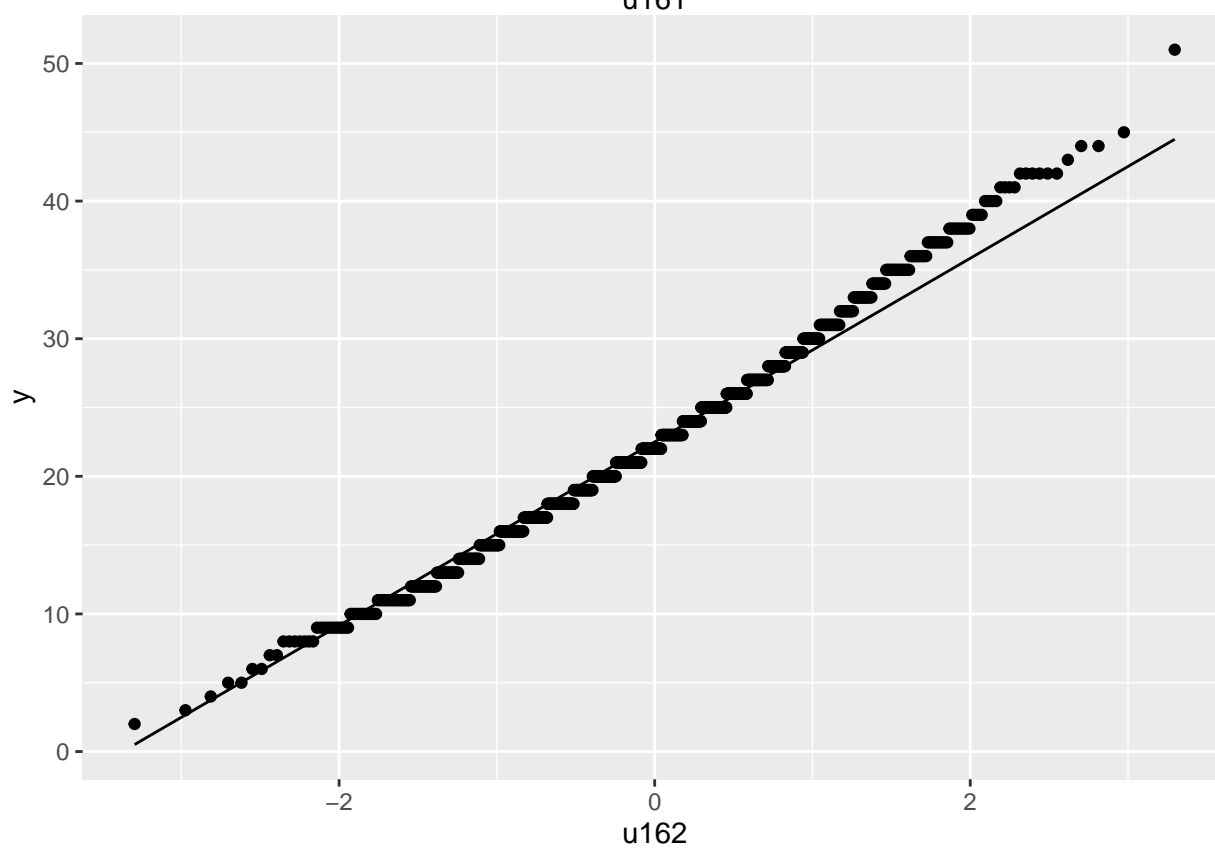
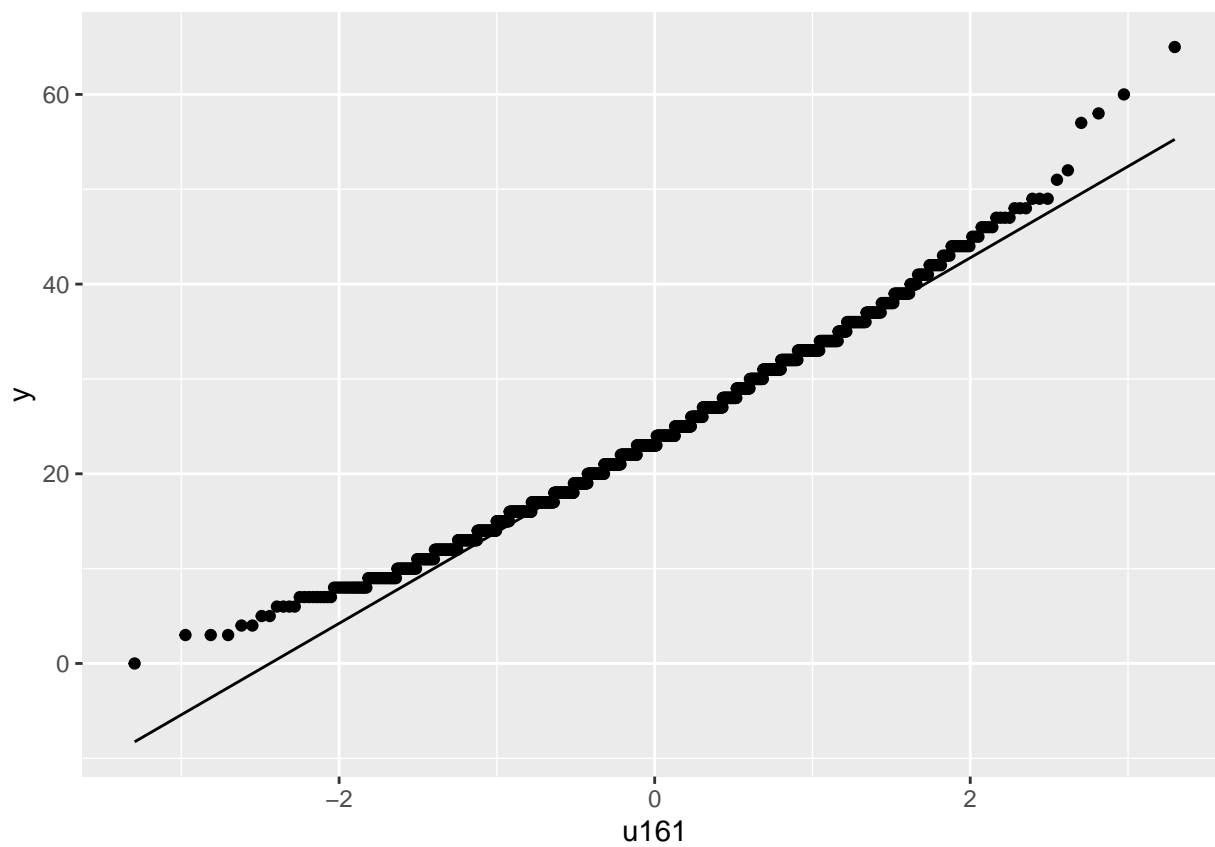


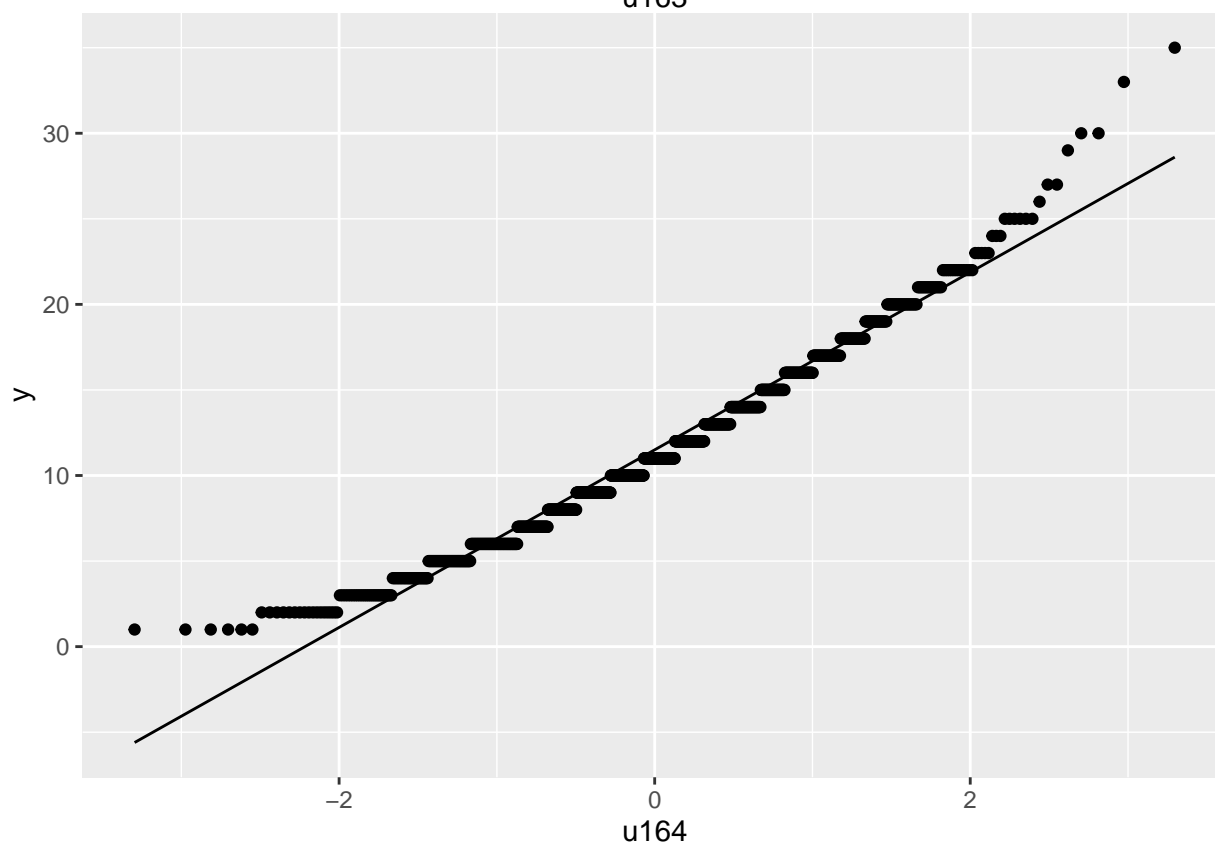
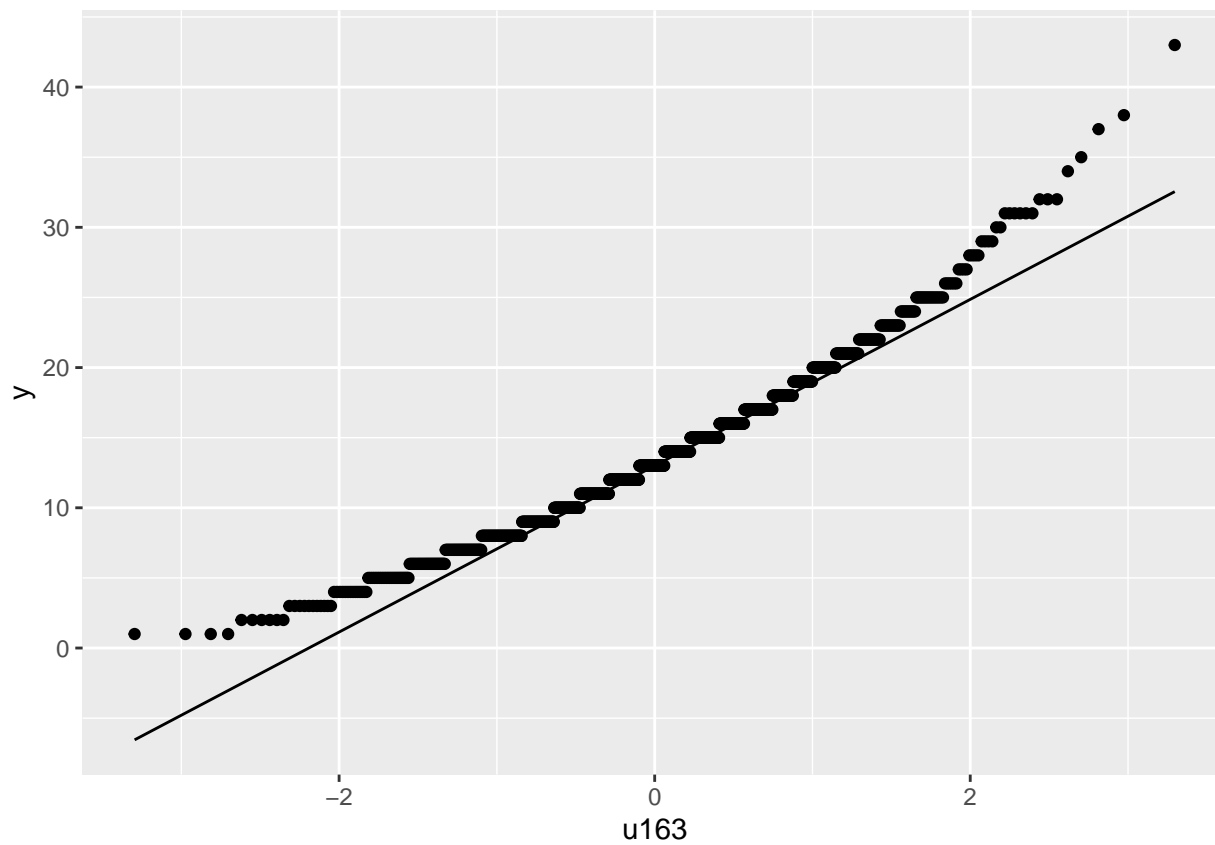


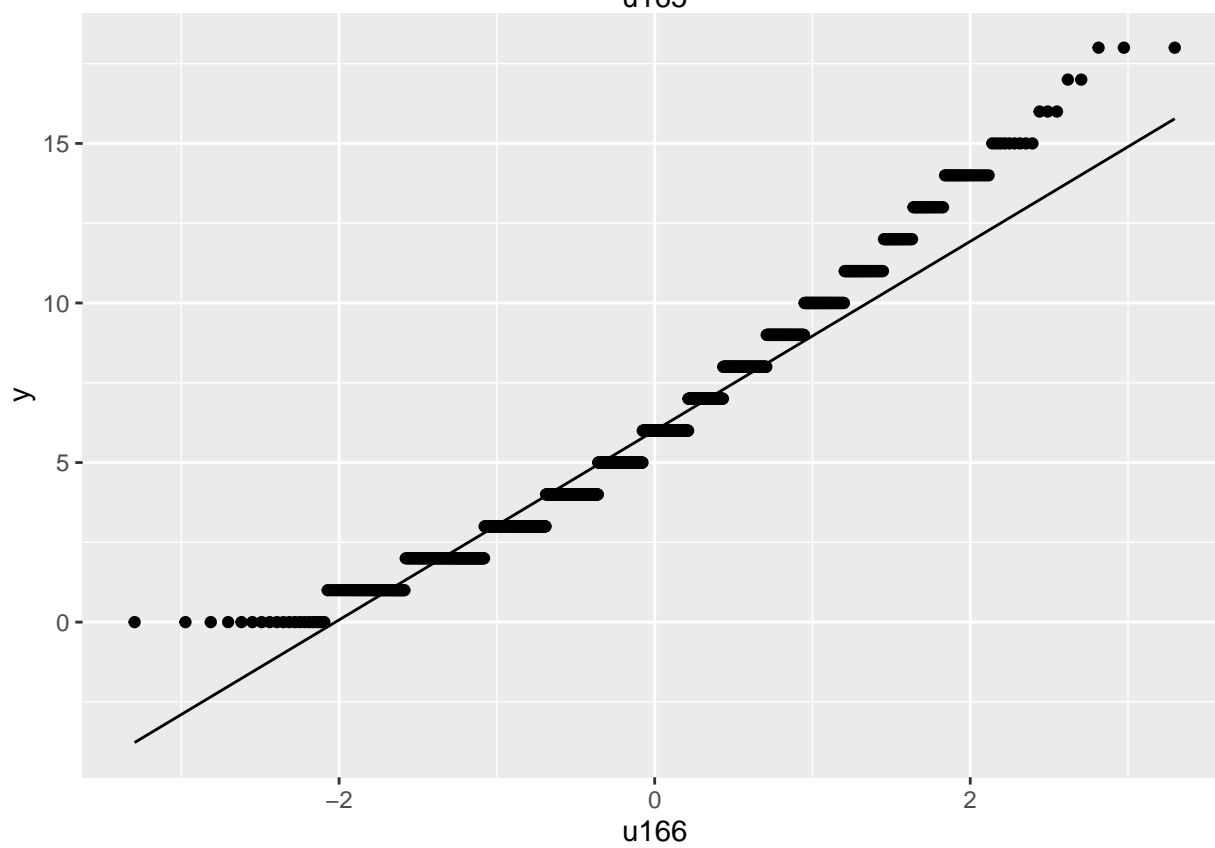
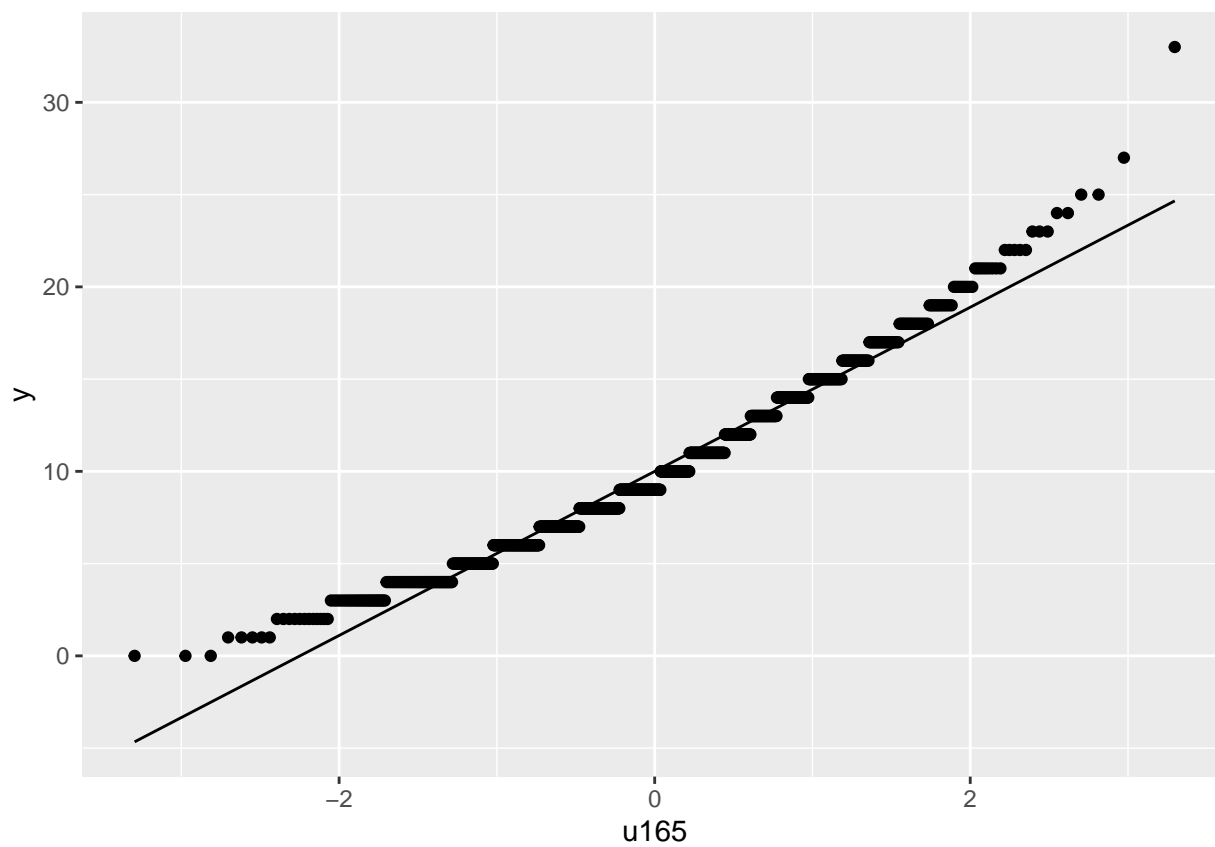


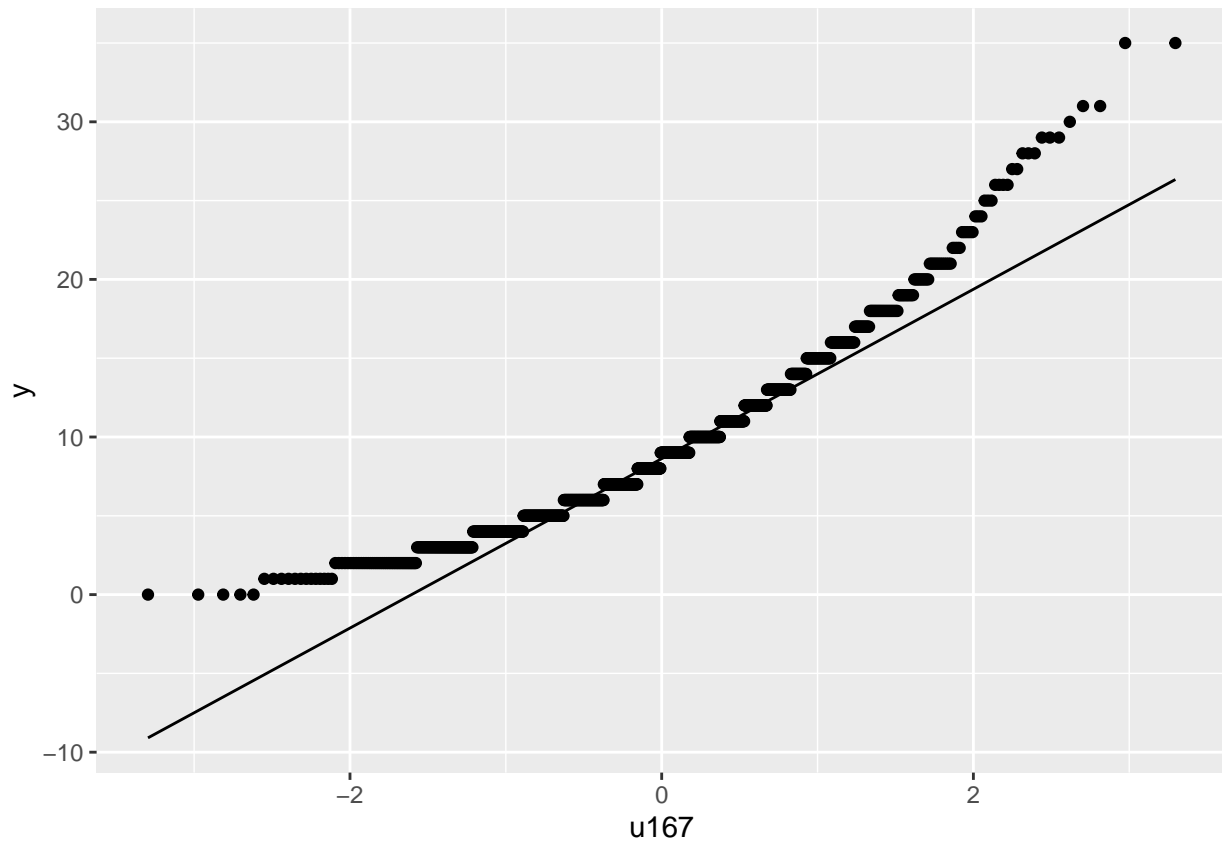








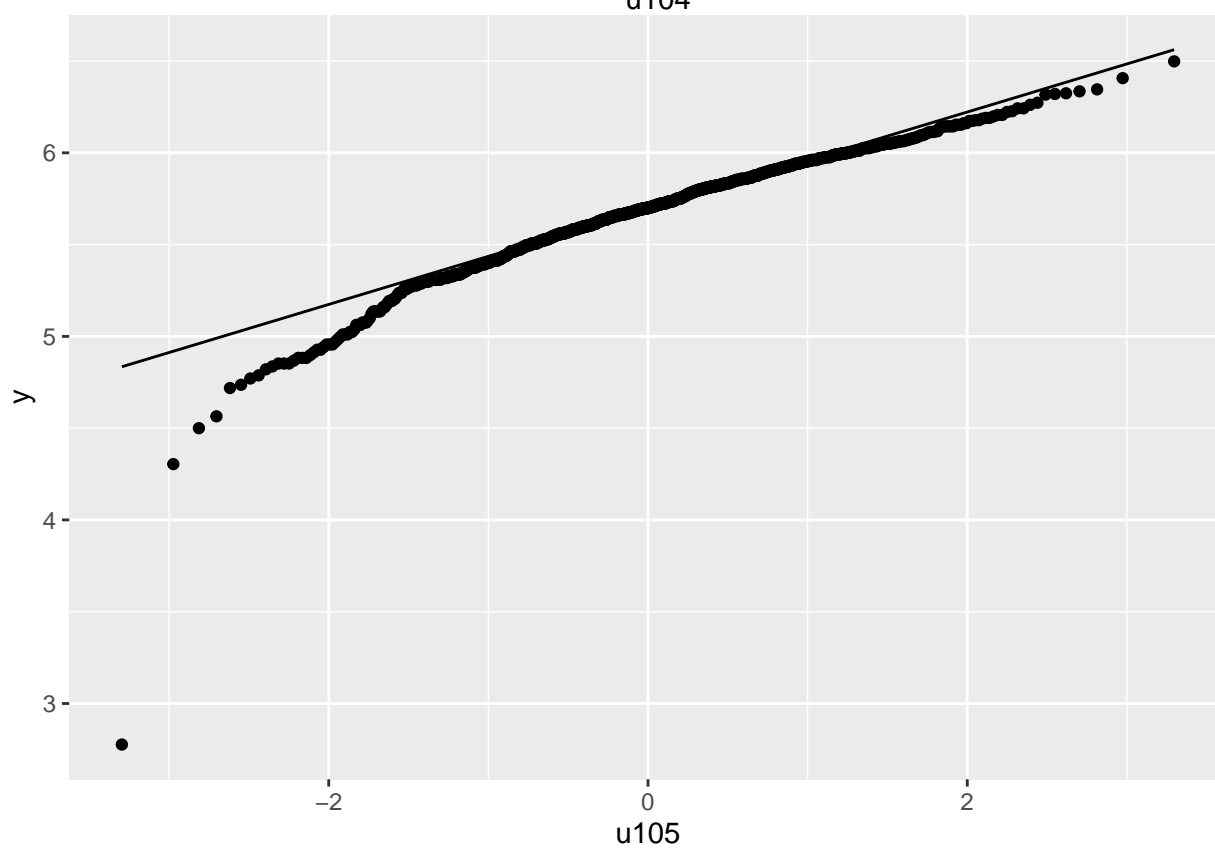
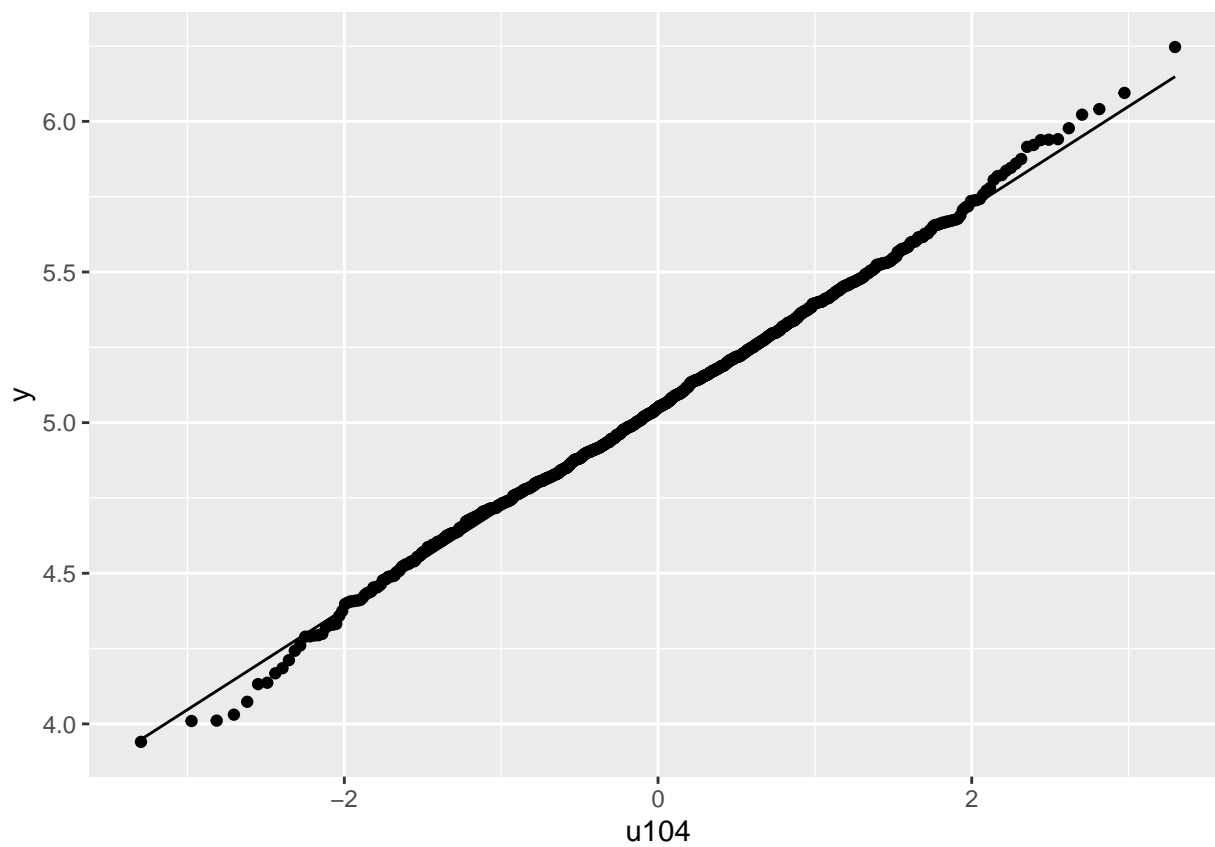


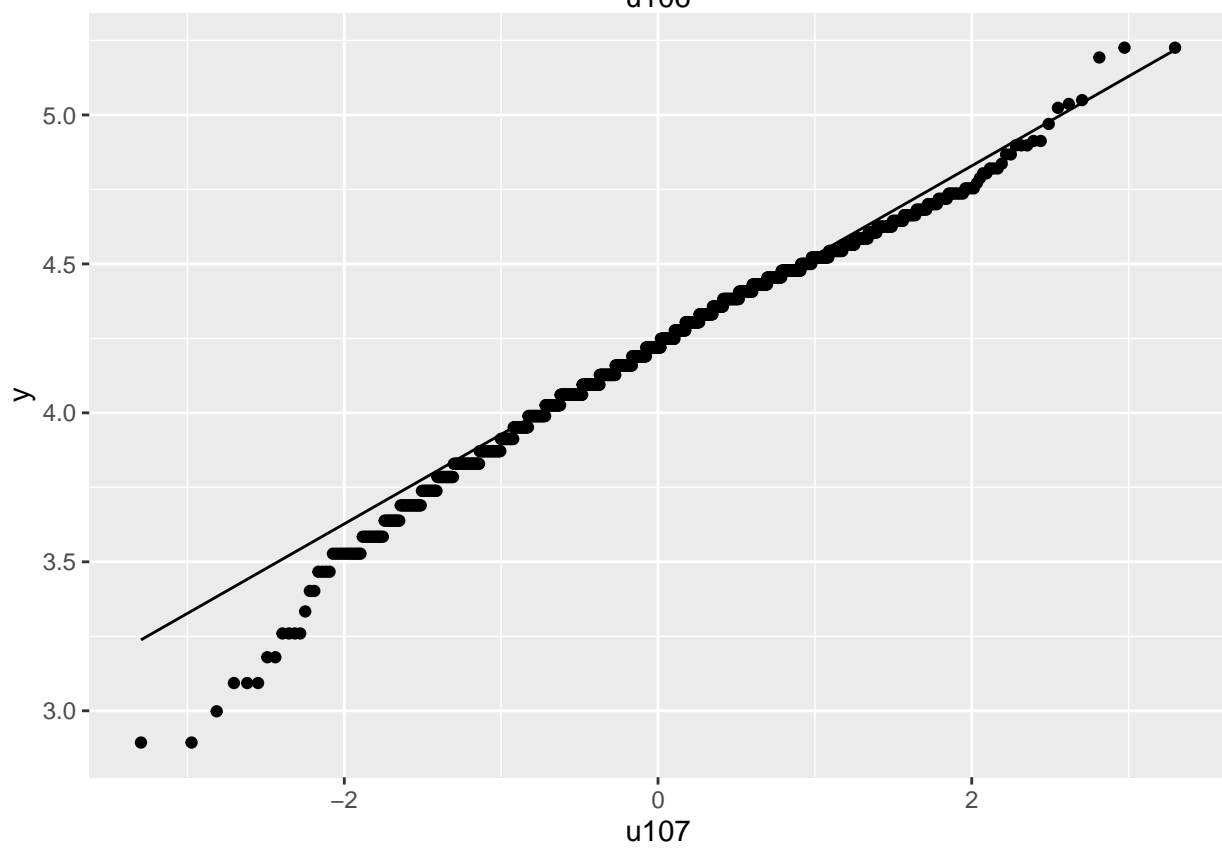
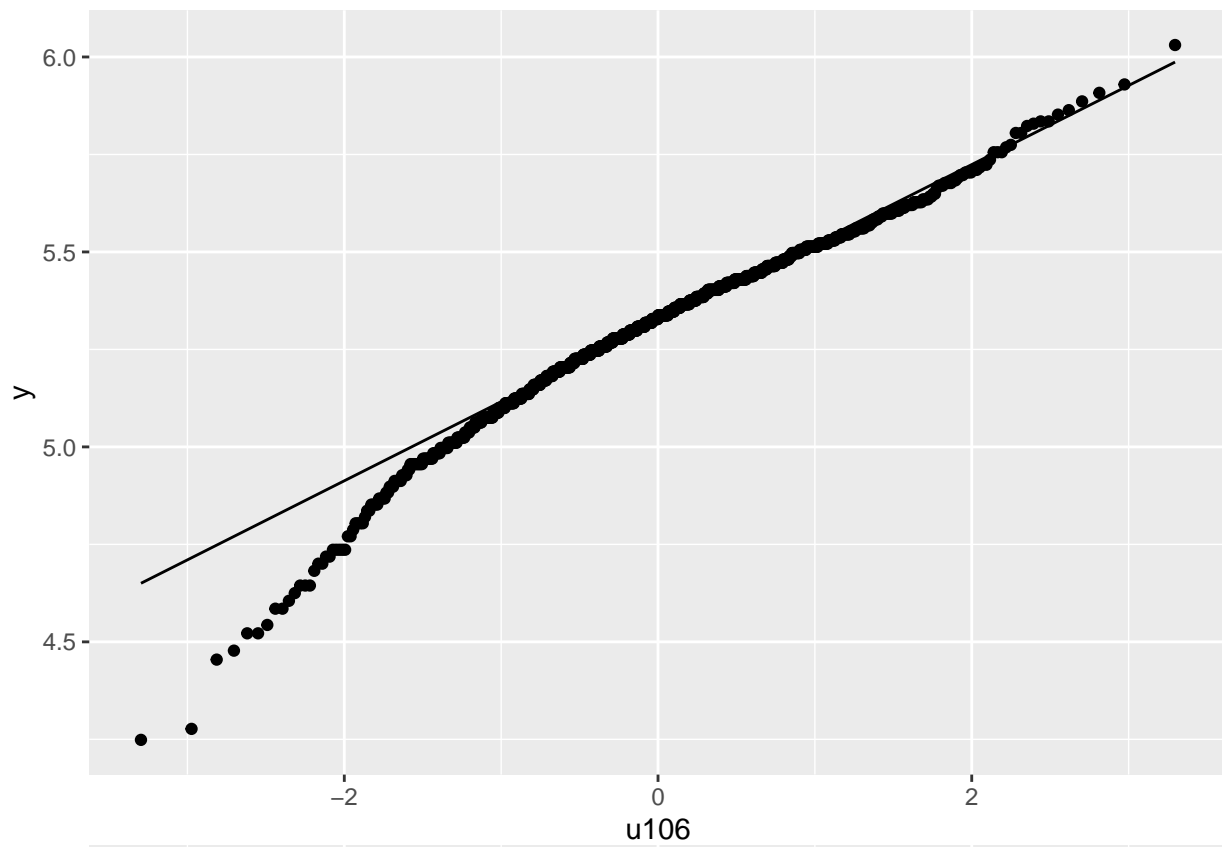


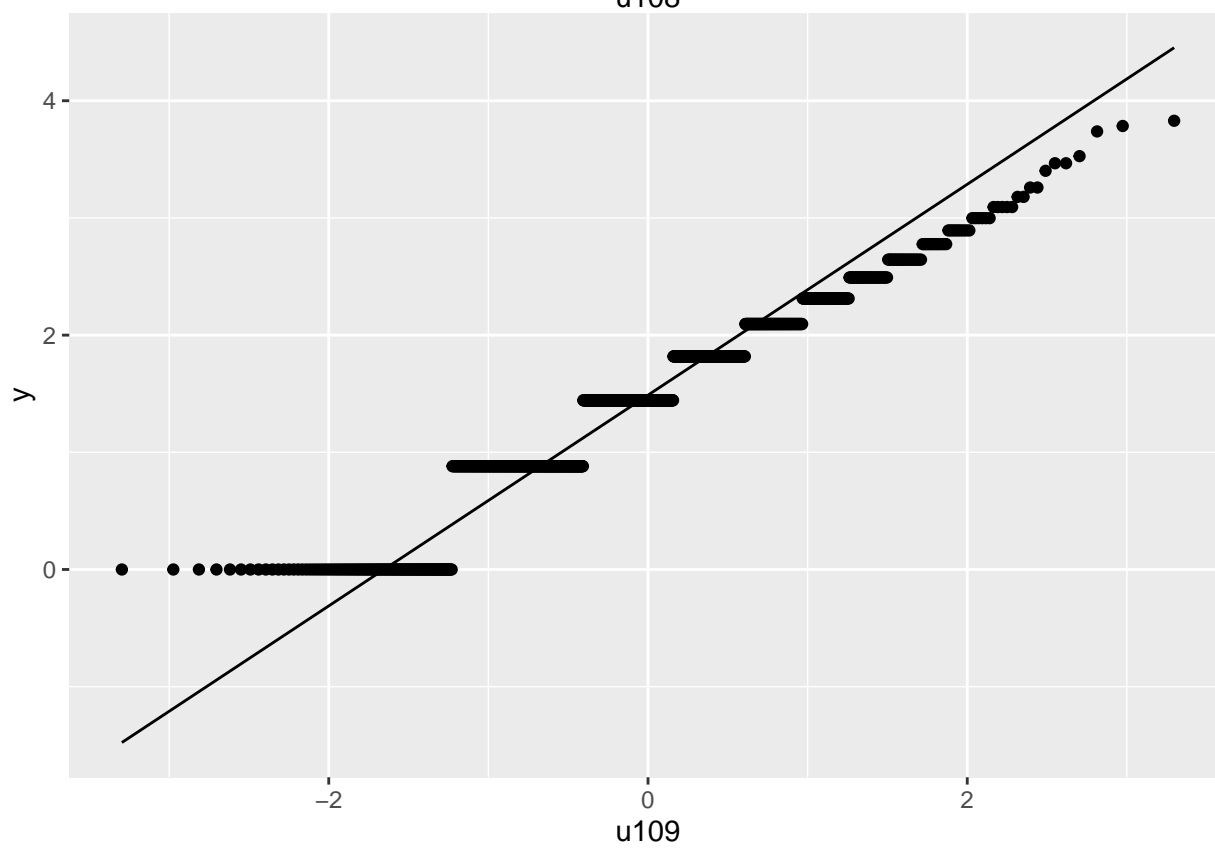
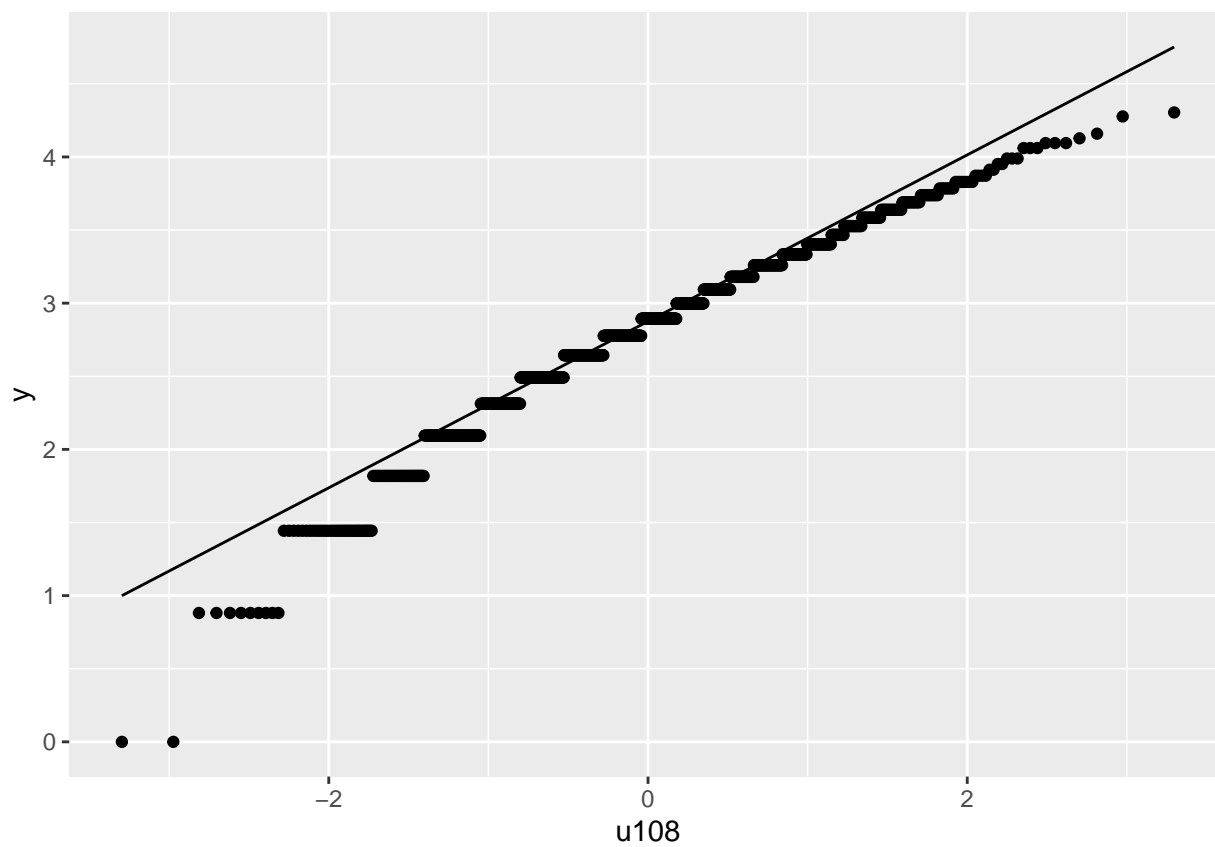
The result from the histogram shows that non of the variables are perfectly normally distributed. However, variables u112, u105, u106, u162 are close to being normally distributed.

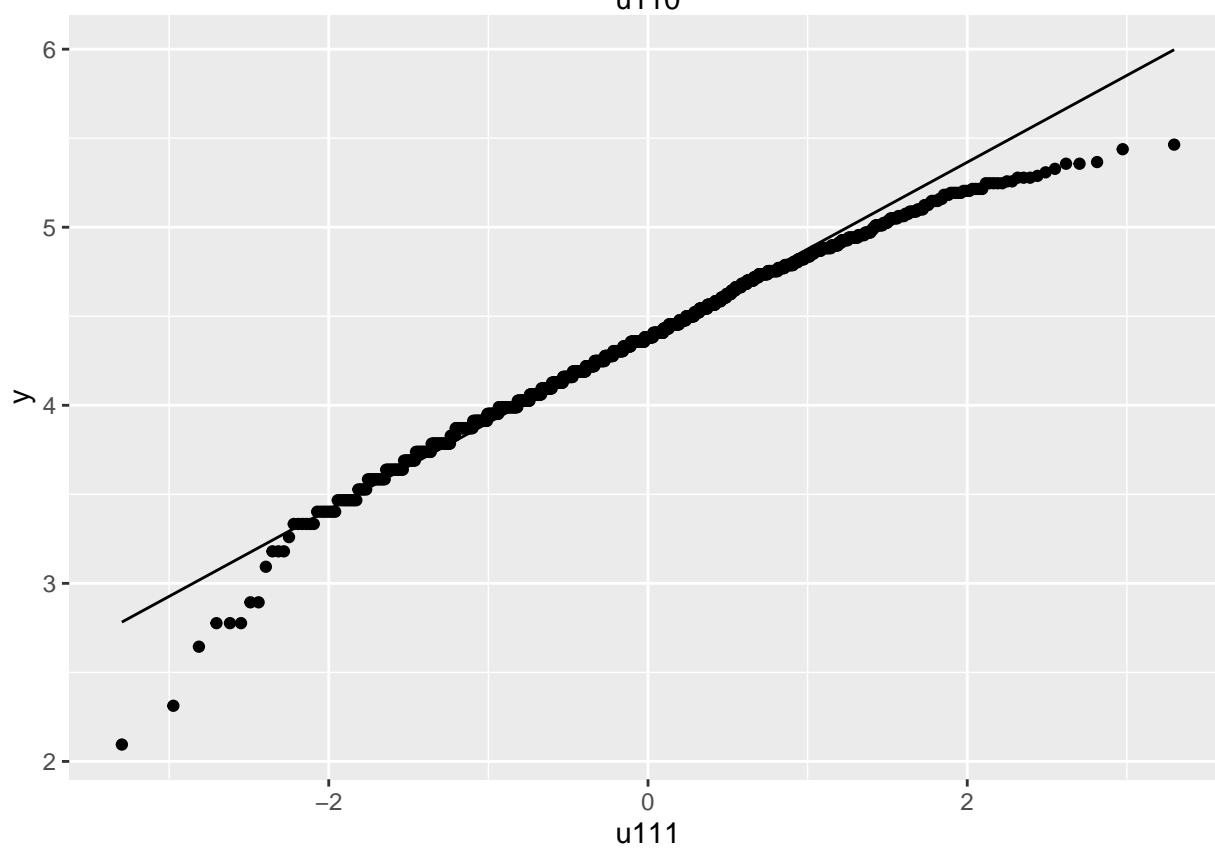
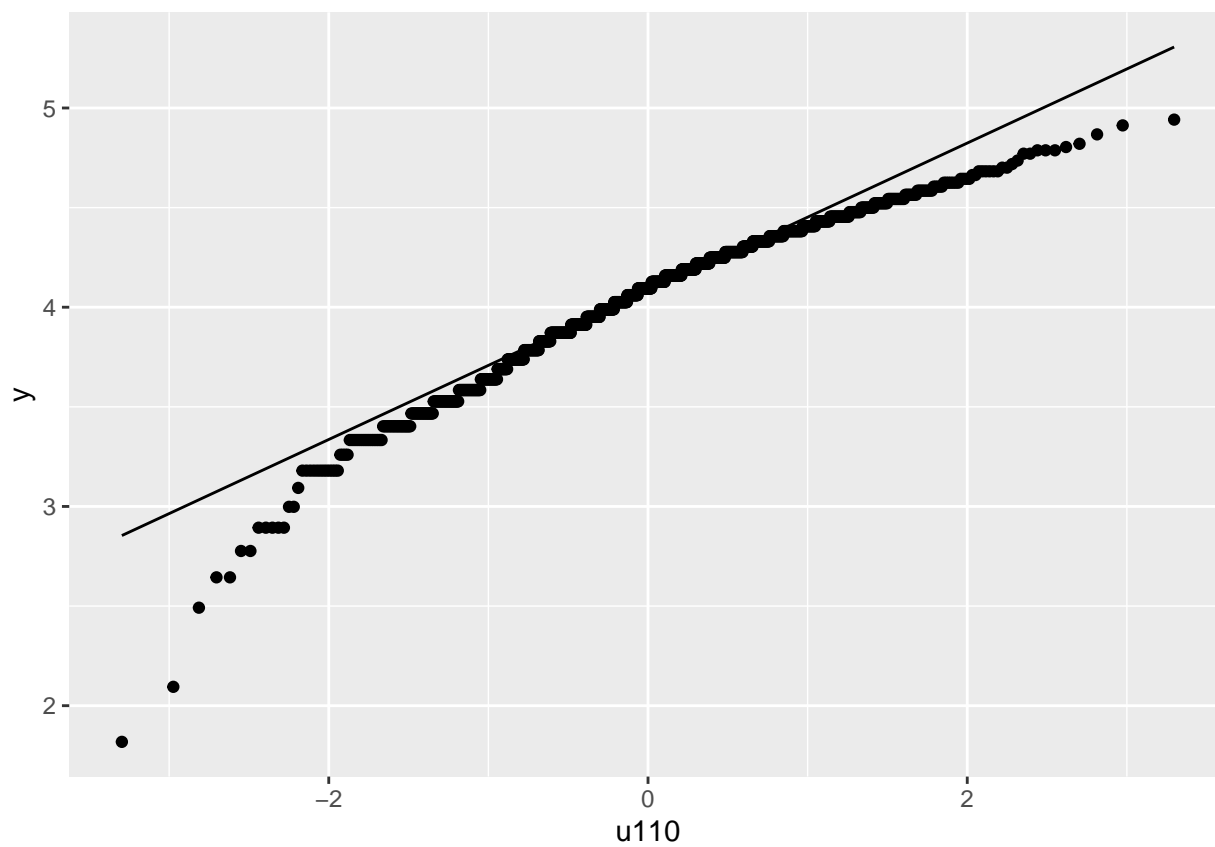
1.13 Tranforming the data with Inverse hyperbolic sine function

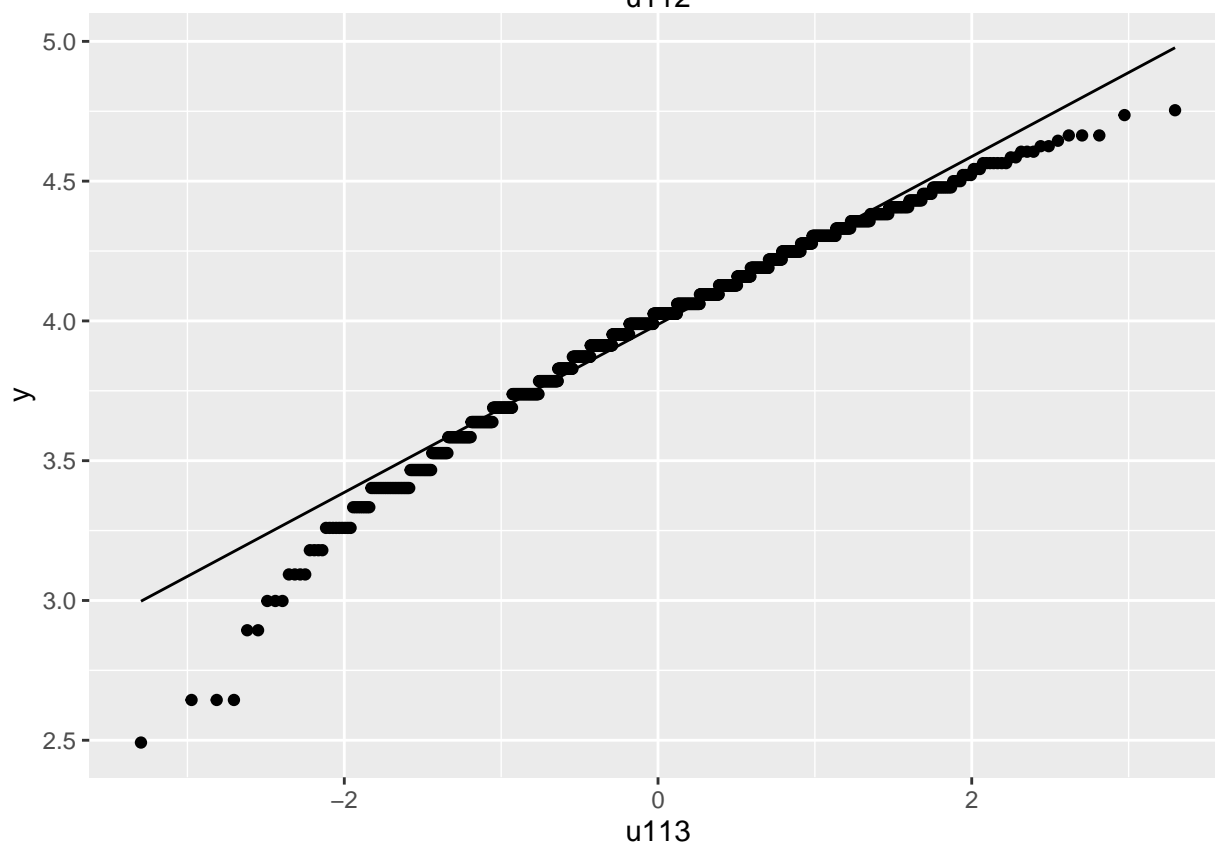
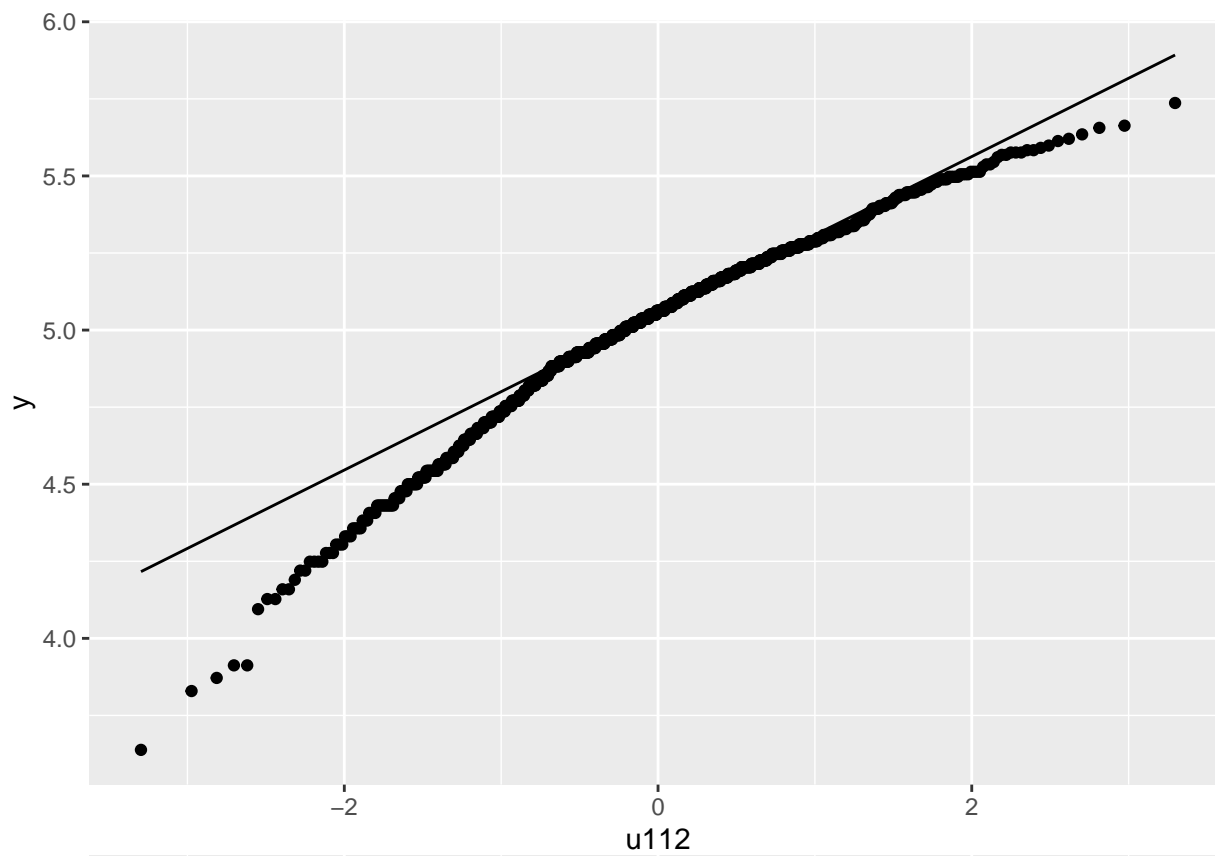
```
for (i in 1:ncol(explorData)) {
  plt <- ggplot2::ggplot(explorData,
    aes(
      #adding inverse hyperbolic sine function
      sample = asinh(explorData[,i])
    )
  ) +
  ggplot2::stat_qq() +
  ggplot2::stat_qq_line()+
  ggplot2::xlab(colnames( explorData[i]))
  print(plt)
}
```

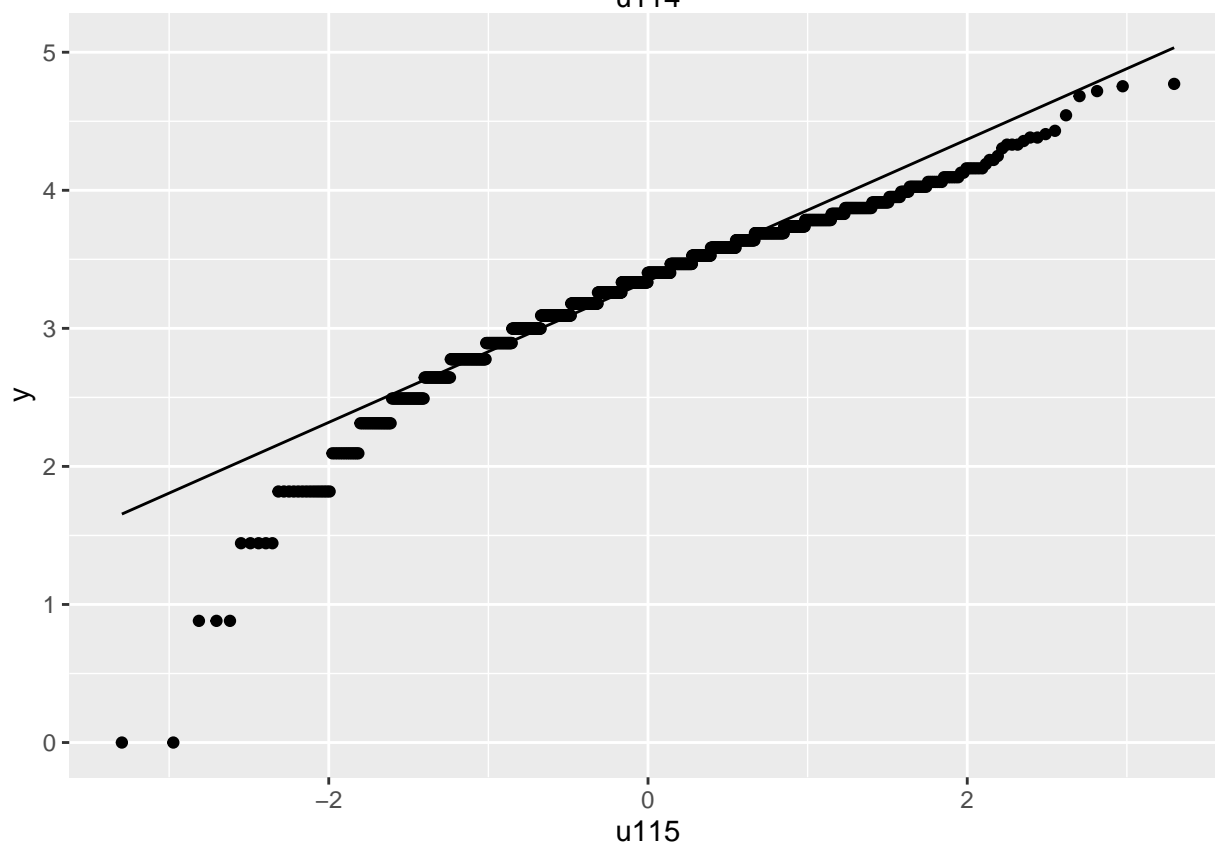
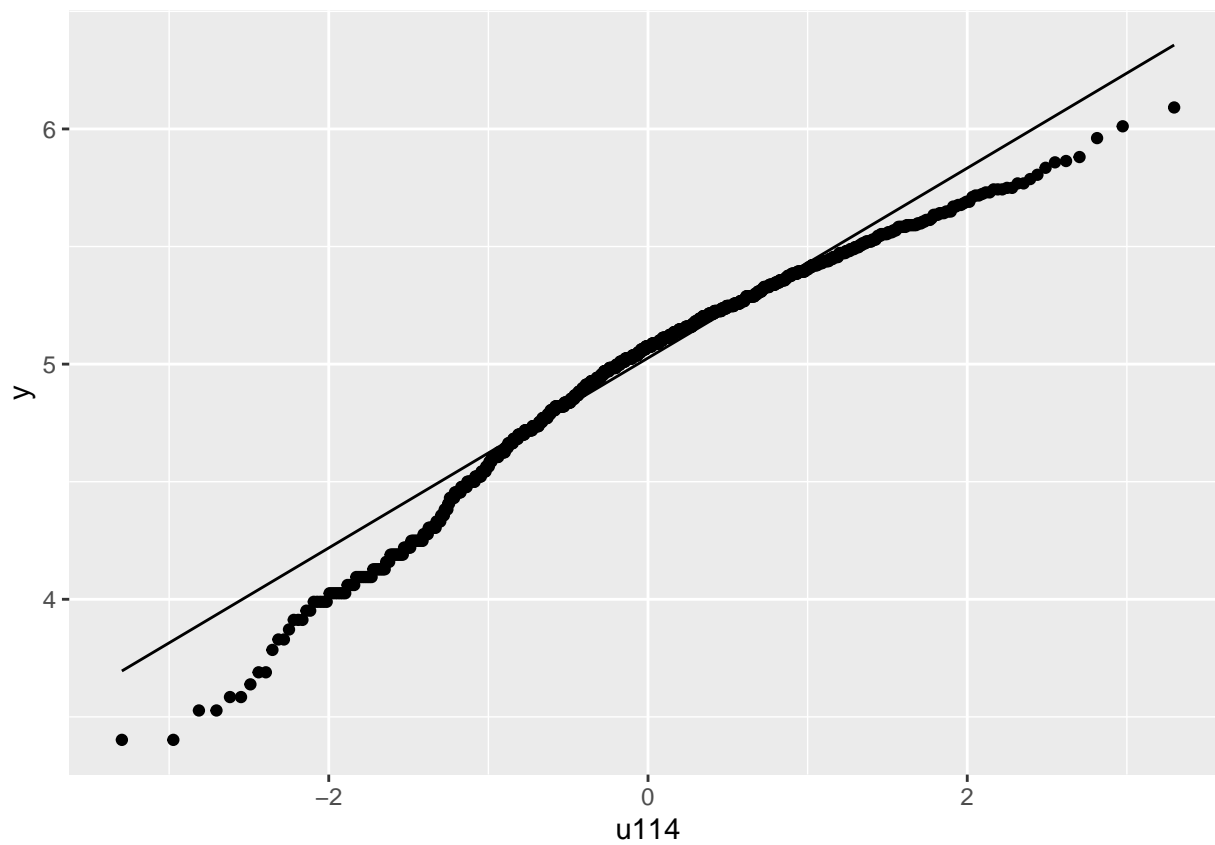


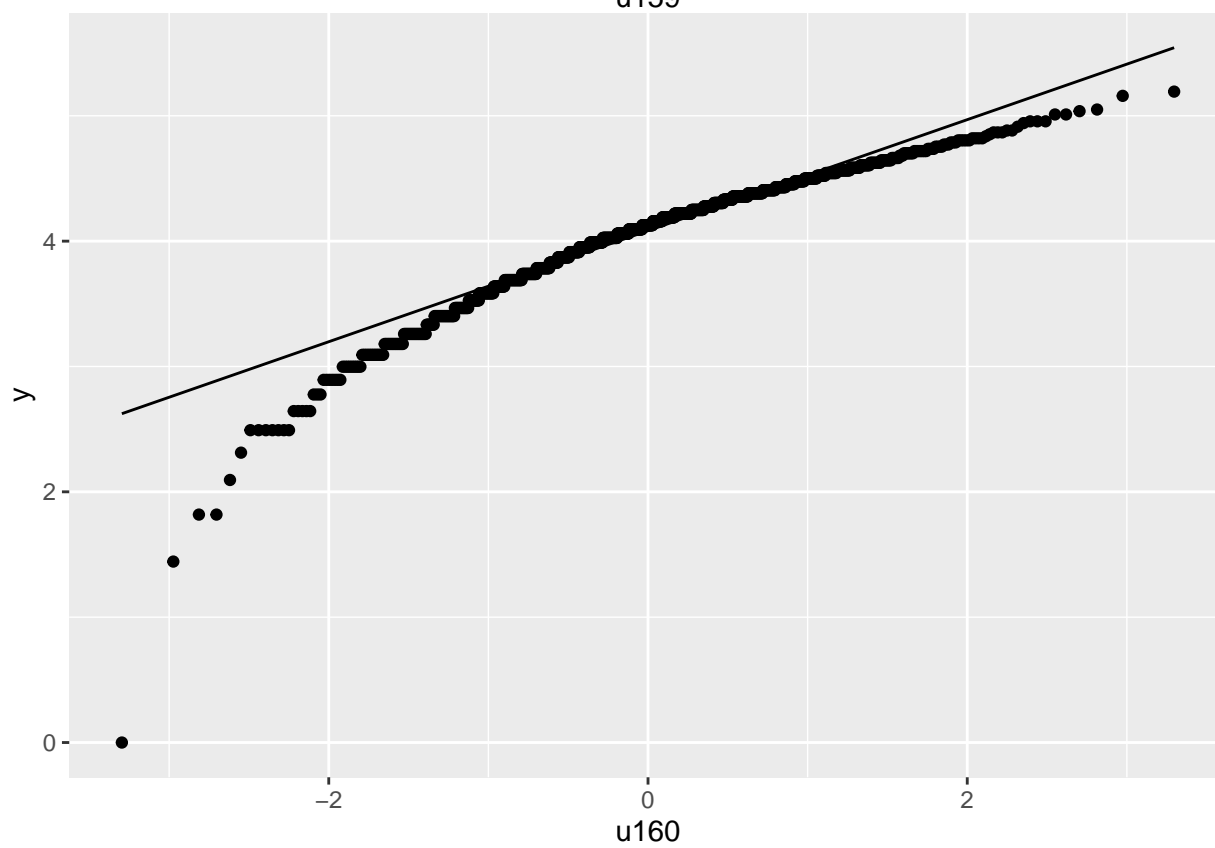
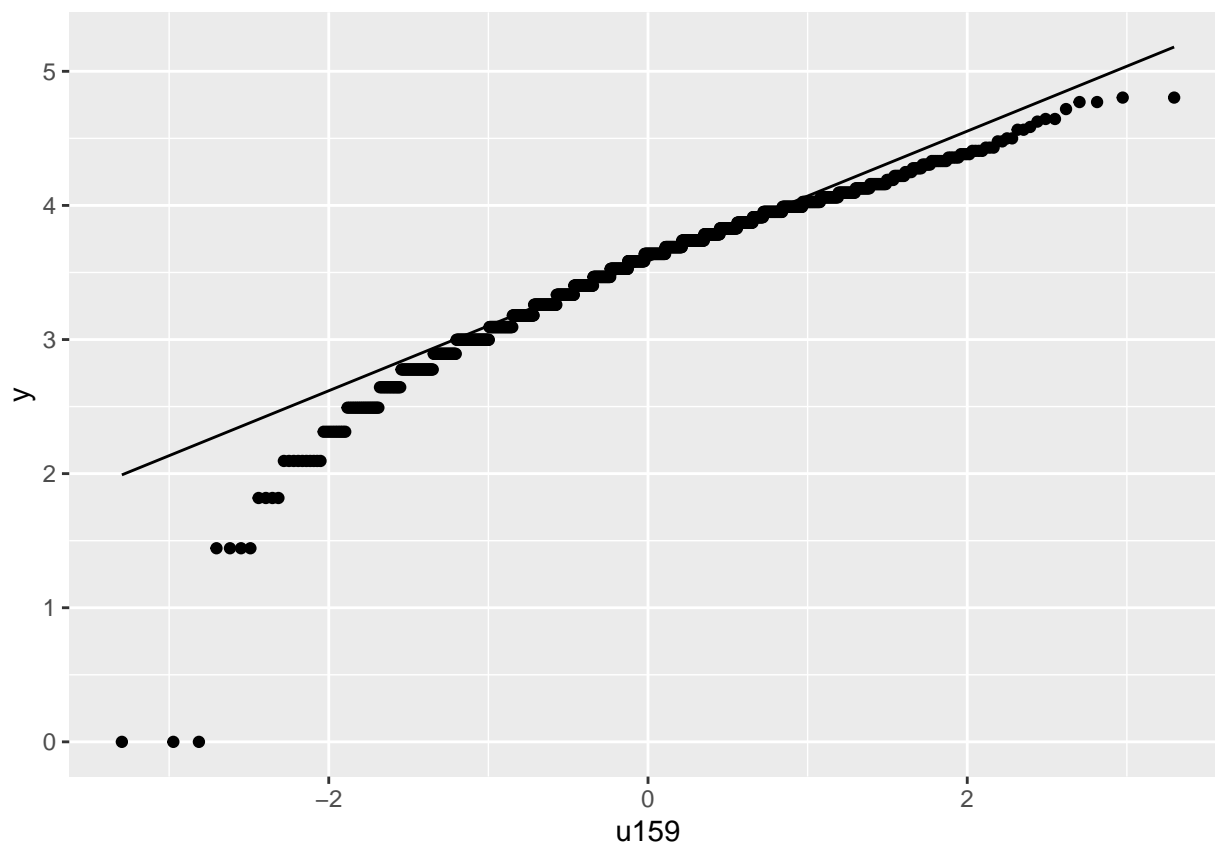


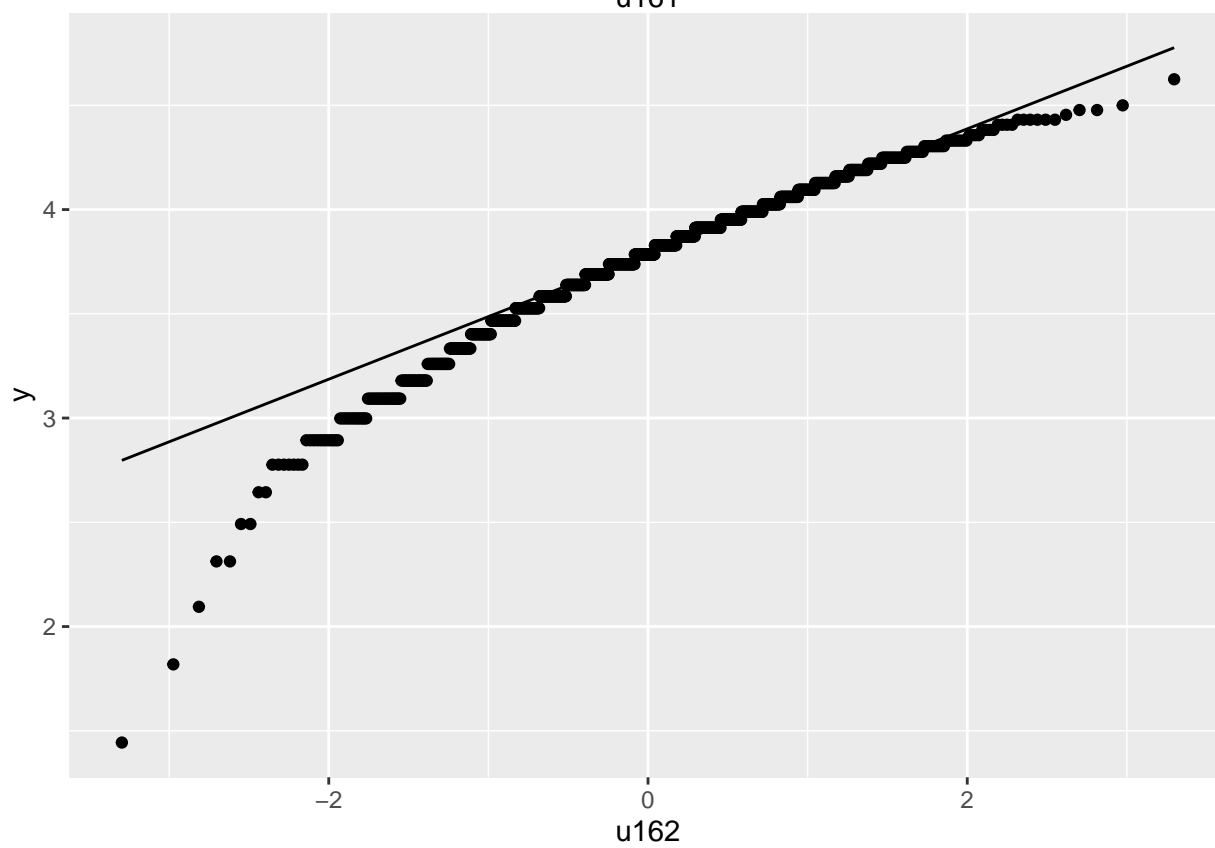
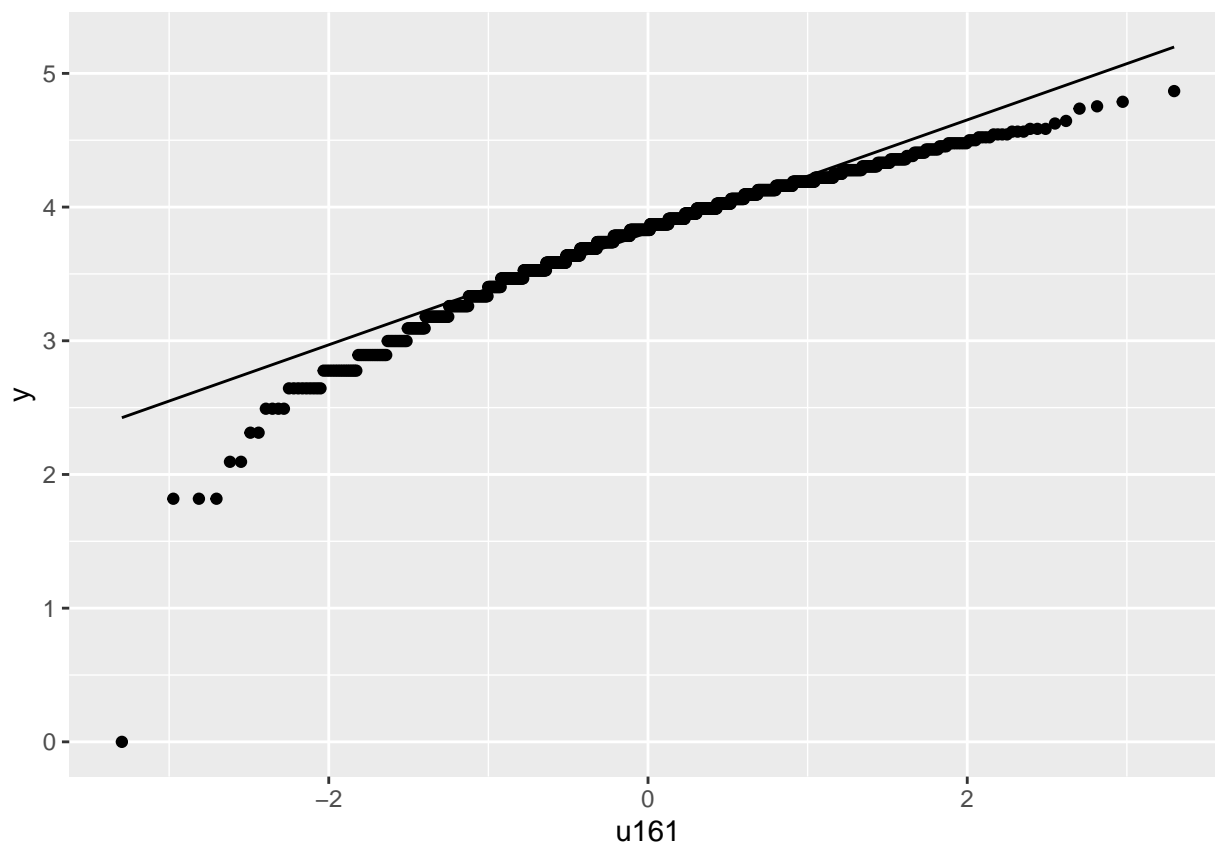


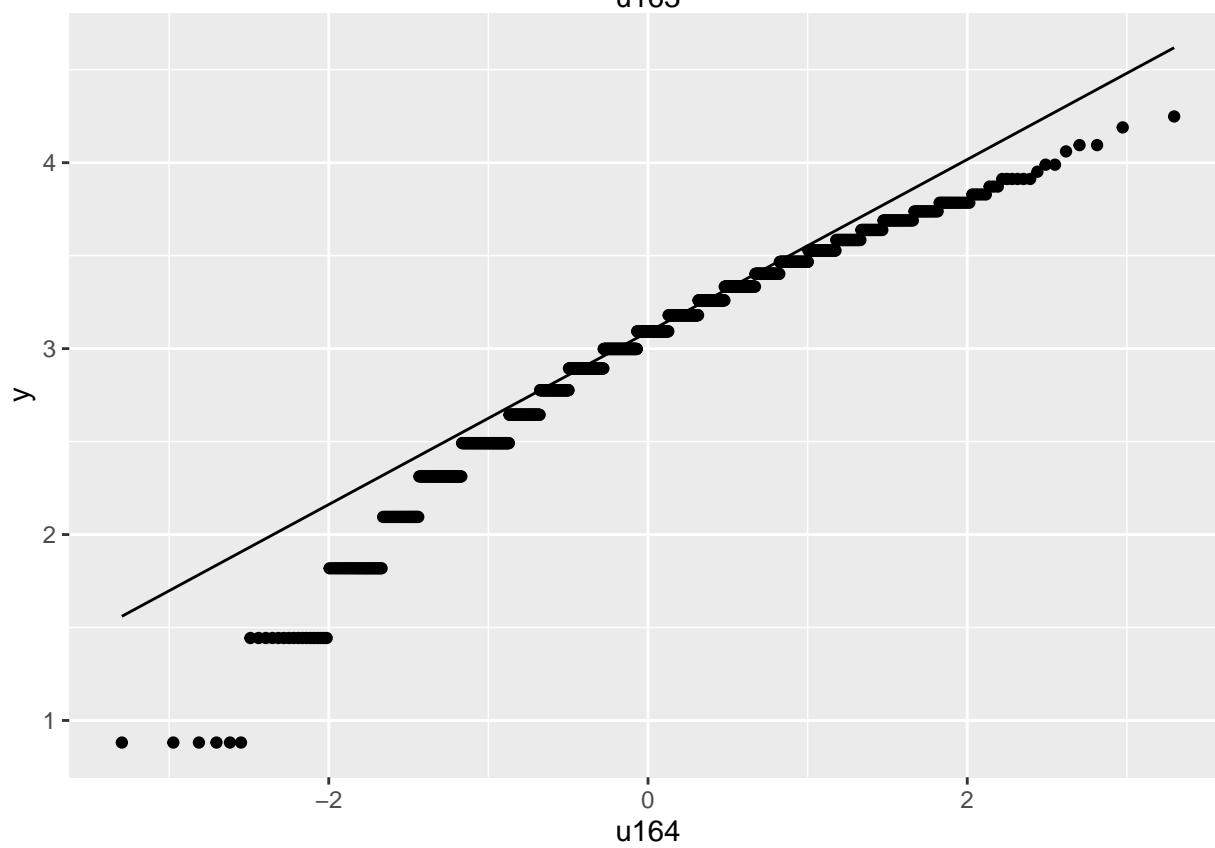
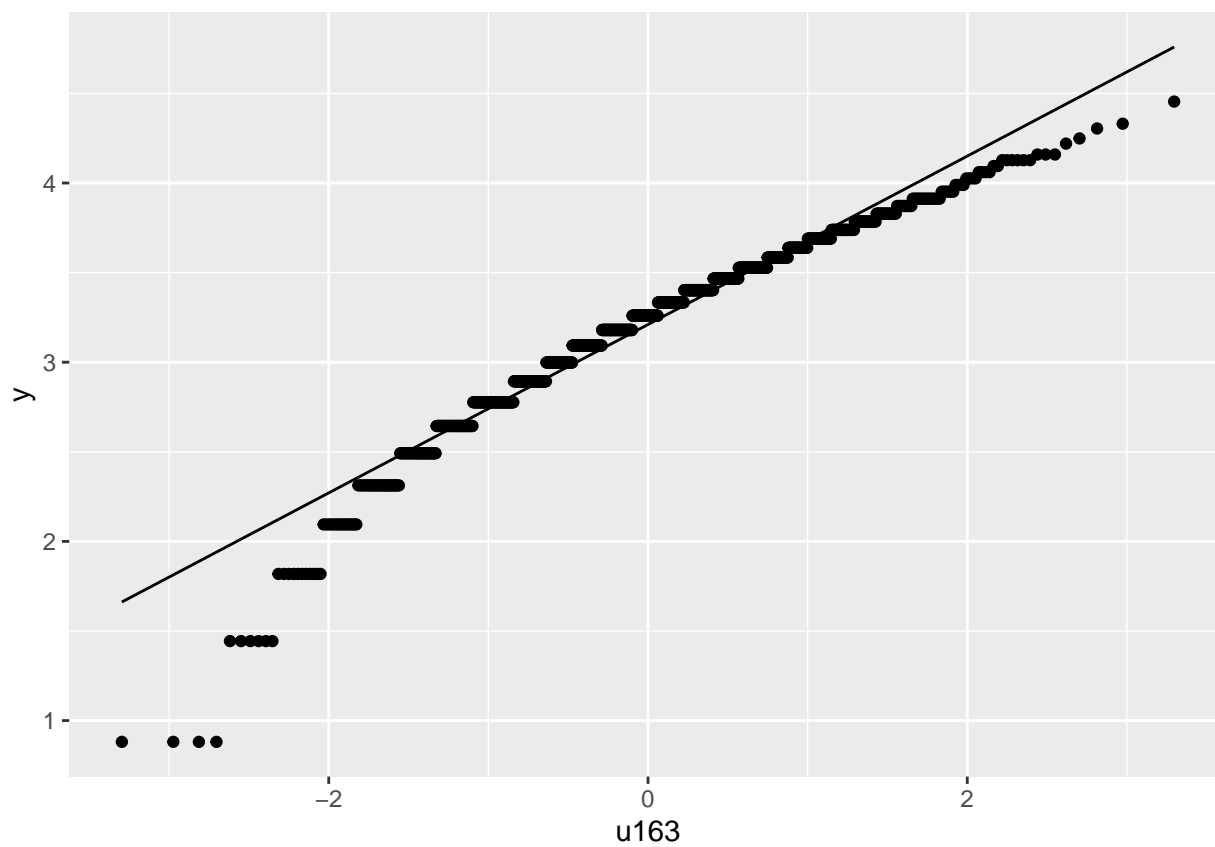


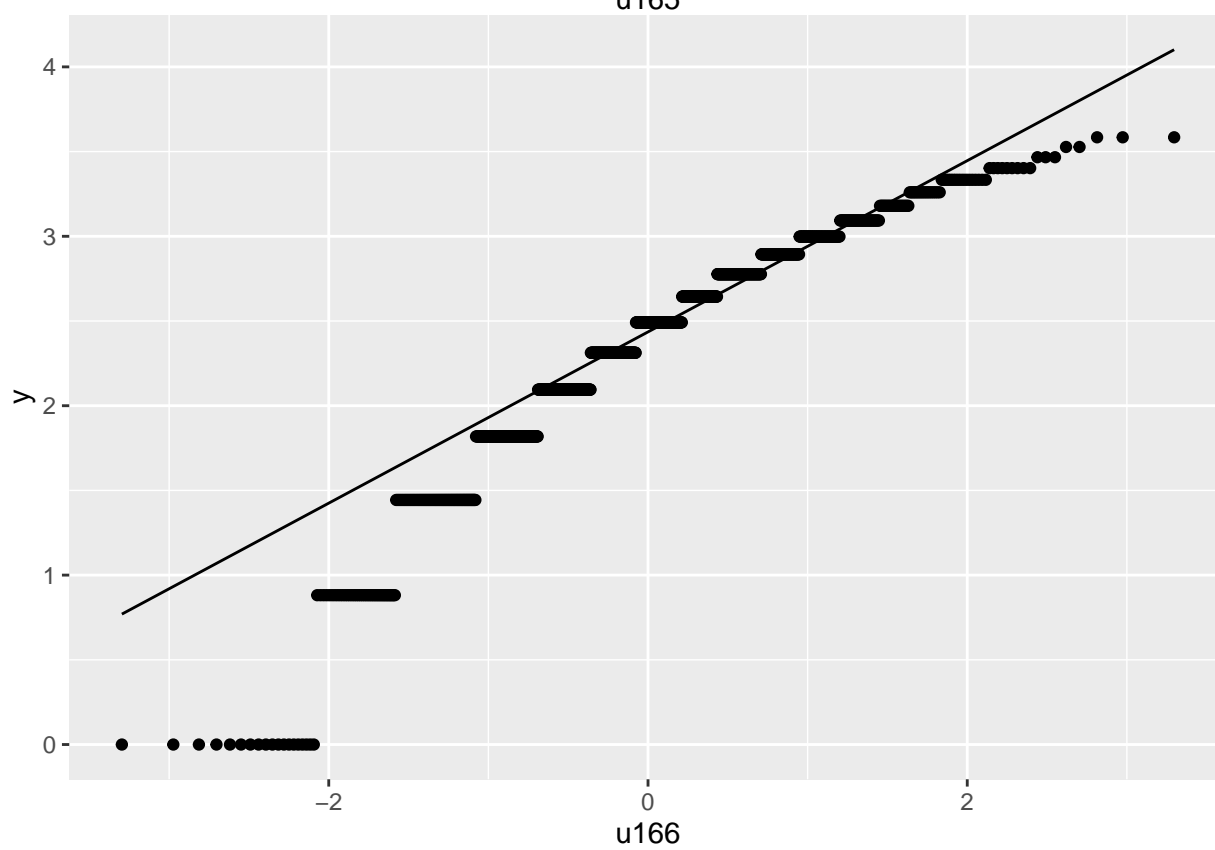
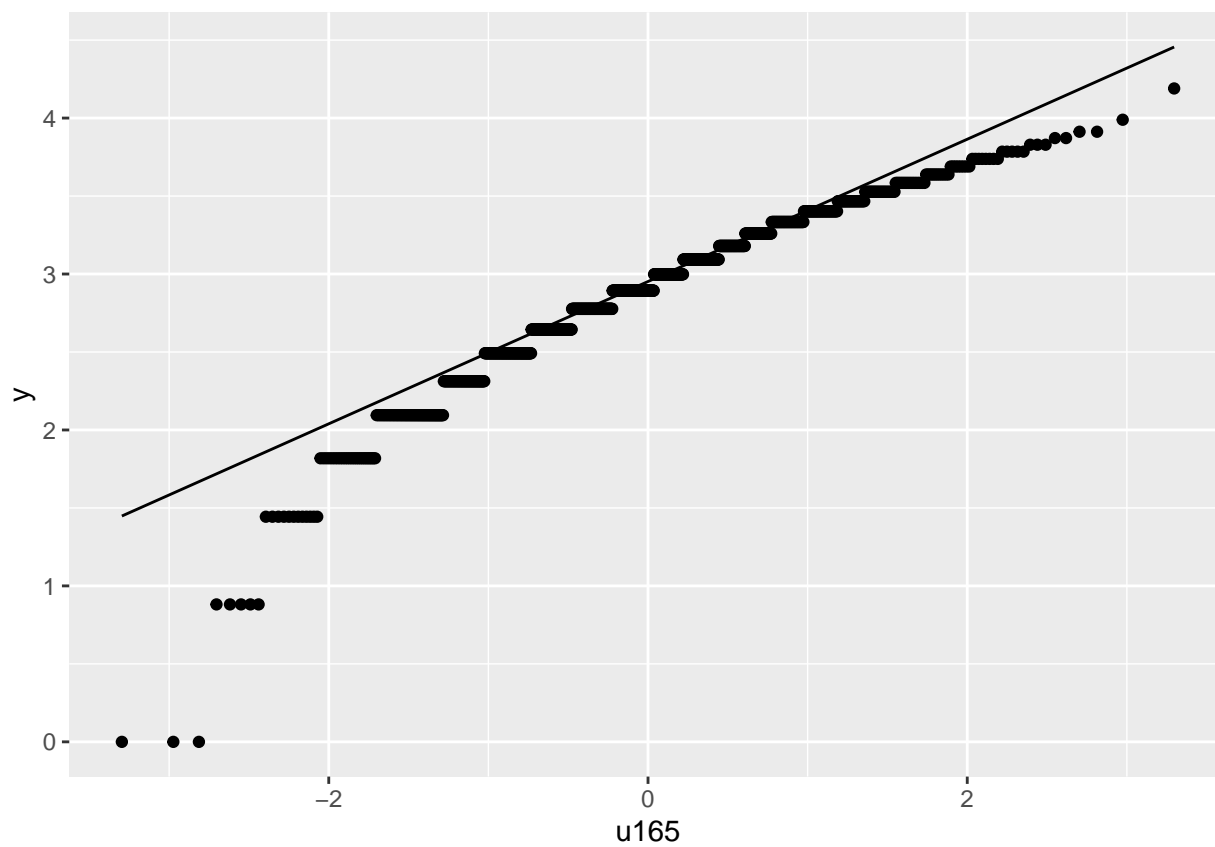


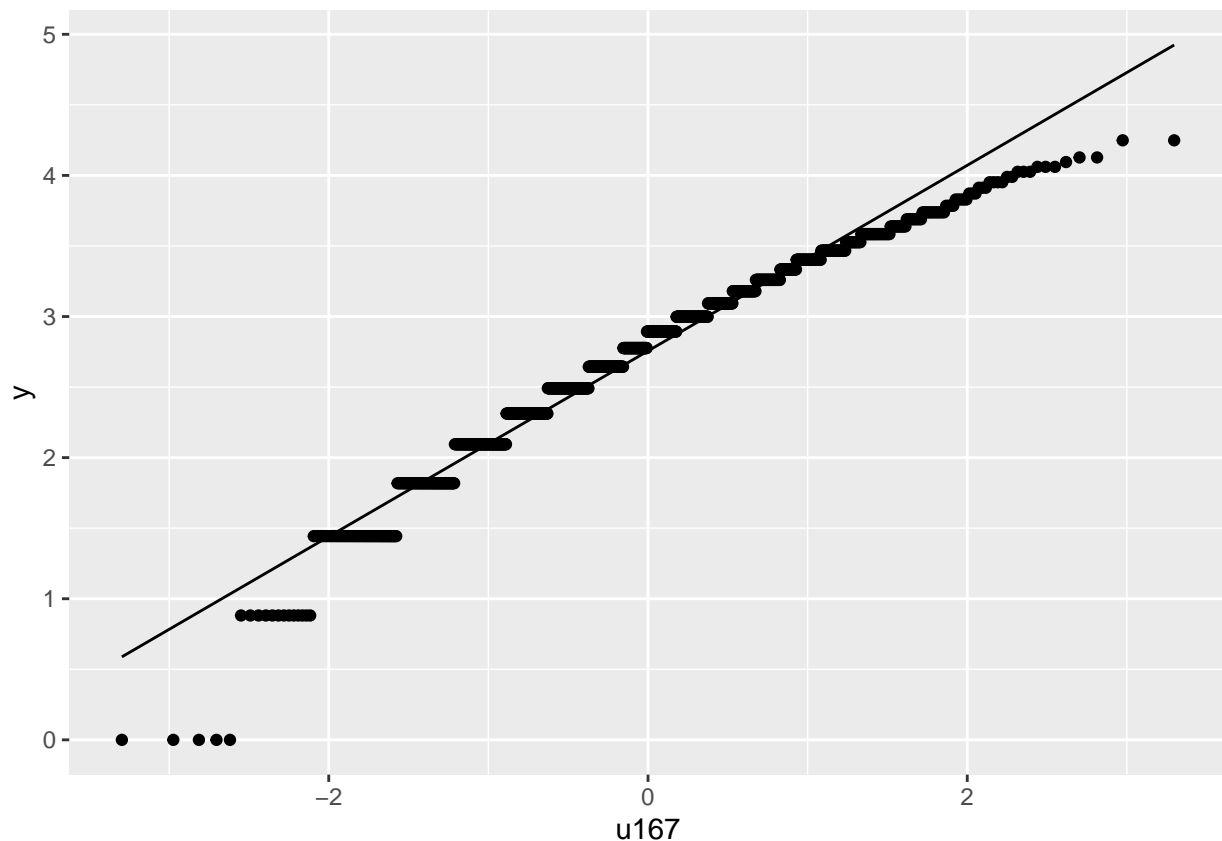












After transforming the transformed data (with inverse hyperbolic sine), the situation did not improve. Considering that plots are not the most accurate way of checking the normality of variables, we opted for statistical methods.

1.21 Statistical approach to exploratory analysis and descriptive statistics of the data

```
stat_view <- explorData %>%
  paste::stat.desc(norm = TRUE) %>%
  round(5)

print(stat_view)
```

##	u104	u105	u106	u107	u108
## nbr.val	1020.00000	1020.00000	1020.00000	1020.00000	1020.00000
## nbr.null	0.00000	0.00000	0.00000	0.00000	2.00000
## nbr.na	0.00000	0.00000	0.00000	0.00000	0.00000
## min	25.70986	8.00000	35.00000	9.00000	0.00000
## max	258.17275	332.00000	208.00000	93.00000	37.00000
## range	232.46290	324.00000	173.00000	84.00000	37.00000
## sum	84522.47699	154998.00000	105574.00000	36098.00000	9904.00000
## median	77.94665	149.50000	103.50000	34.00000	9.00000
## mean	82.86517	151.95882	103.50392	35.39020	9.70980
## SE.mean	0.90878	1.27974	0.69948	0.34379	0.16941
## CI.mean.0.95	1.78329	2.51122	1.37259	0.67461	0.33243
## var	842.39926	1670.47917	499.06083	120.55221	29.27390
## std.dev	29.02412	40.87150	22.33967	10.97963	5.41054
## coef.var	0.35026	0.26896	0.21583	0.31024	0.55722

## skewness	1.22876	0.25581	0.26070	0.77408	1.18682
## skew.2SE	8.02227	1.67011	1.70207	5.05380	7.74847
## kurtosis	2.87710	0.51392	0.94996	2.03134	2.14862
## kurt.2SE	9.40112	1.67926	3.10406	6.63753	7.02077
## normtest.W	0.93275	0.99460	0.99108	0.96913	0.92668
## normtest.p	0.00000	0.00102	0.00001	0.00000	0.00000
##	u109	u110	u111	u112	u113
## nbr.val	1020.00000	1020.00000	1020.00000	1020.00000	1020.00000
## nbr.null	112.00000	0.00000	0.00000	0.00000	0.00000
## nbr.na	0.00000	0.00000	0.00000	0.00000	0.00000
## min	0.00000	3.00000	4.00000	19.00000	6.00000
## max	23.00000	70.00000	118.00000	155.00000	58.00000
## range	23.00000	67.00000	114.00000	136.00000	52.00000
## sum	2818.00000	31012.00000	44572.00000	80656.00000	28644.00000
## median	2.00000	30.00000	40.00000	79.00000	28.00000
## mean	2.76275	30.40392	43.69804	79.07451	28.08235
## SE.mean	0.07969	0.32952	0.60058	0.67624	0.25776
## CI.mean.0.95	0.15638	0.64661	1.17852	1.32697	0.50580
## var	6.47751	110.75327	367.91266	466.43998	67.76750
## std.dev	2.54510	10.52394	19.18105	21.59722	8.23210
## coef.var	0.92122	0.34614	0.43895	0.27312	0.29314
## skewness	2.61363	0.30229	0.81435	0.14423	0.33283
## skew.2SE	17.06377	1.97356	5.31668	0.94164	2.17295
## kurtosis	12.47496	-0.06134	0.46615	-0.01207	0.16342
## kurt.2SE	40.76273	-0.20043	1.52318	-0.03943	0.53398
## normtest.W	0.78697	0.99086	0.95556	0.99684	0.99153
## normtest.p	0.00000	0.00001	0.00000	0.03971	0.00001
##	u114	u115	u159	u160	u161
## nbr.val	1020.00000	1020.00000	1020.00000	1020.00000	1020.00000
## nbr.null	0.00000	2.00000	3.00000	1.00000	1.00000
## nbr.na	0.00000	0.00000	0.00000	0.00000	0.00000
## min	15.00000	0.00000	0.00000	0.00000	0.00000
## max	221.00000	59.00000	61.00000	90.00000	65.00000
## range	206.00000	59.00000	61.00000	90.00000	65.00000
## sum	82304.00000	15640.00000	19784.00000	32394.00000	24488.00000
## median	80.00000	15.00000	19.00000	31.00000	23.00000
## mean	80.69020	15.33333	19.39608	31.75882	24.00784
## SE.mean	0.97809	0.22604	0.28281	0.41917	0.29354
## CI.mean.0.95	1.91930	0.44356	0.55495	0.82253	0.57600
## var	975.79205	52.11645	81.57996	179.21361	87.88610
## std.dev	31.23767	7.21917	9.03216	13.38707	9.37476
## coef.var	0.38713	0.47082	0.46567	0.42152	0.39049
## skewness	0.43096	1.24588	0.86535	0.49587	0.44075
## skew.2SE	2.81365	8.13410	5.64968	3.23741	2.87754
## kurtosis	0.32769	4.35593	1.64779	0.49176	0.35088
## kurt.2SE	1.07075	14.23329	5.38425	1.60685	1.14652
## normtest.W	0.98623	0.93589	0.96110	0.98474	0.98720
## normtest.p	0.00000	0.00000	0.00000	0.00000	0.00000
##	u162	u163	u164	u165	u166
## nbr.val	1020.00000	1020.00000	1020.00000	1020.00000	1020.00000
## nbr.null	0.00000	0.00000	0.00000	3.00000	19.00000
## nbr.na	0.00000	0.00000	0.00000	0.00000	0.00000
## min	2.00000	1.00000	1.00000	0.00000	0.00000
## max	51.00000	43.00000	35.00000	33.00000	18.00000

```

## range      49.00000    42.00000    34.00000    33.00000    18.00000
## sum        23079.00000 13956.00000 11546.00000 10194.00000 6297.00000
## median     22.00000    13.00000    11.00000     9.00000     6.00000
## mean       22.62647    13.68235    11.31961     9.99412     6.17353
## SE.mean    0.23240     0.18688     0.16242     0.14180     0.10788
## CI.mean.0.95 0.45603     0.36672     0.31871     0.27826     0.21169
## var        55.08801    35.62422    26.90658    20.51027    11.87074
## std.dev     7.42213     5.96860     5.18716     4.52883     3.44539
## coef.var    0.32803     0.43623     0.45825     0.45315     0.55809
## skewness    0.30842     0.71611     0.58252     0.64661     0.58164
## skew.2SE    2.01358     4.67532     3.80316     4.22155     3.79740
## kurtosis    0.02644     1.00322     0.54622     0.62081     0.01859
## kurt.2SE    0.08638     3.27808     1.78480     2.02853     0.06074
## normtest.W  0.99213     0.97082     0.97629     0.97154     0.96626
## normtest.p  0.00003     0.00000     0.00000     0.00000     0.00000
##            u167
## nbr.val     1020.00000
## nbr.null     5.00000
## nbr.na       0.00000
## min         0.00000
## max         35.00000
## range       35.00000
## sum        9630.00000
## median      9.00000
## mean       9.44118
## SE.mean     0.17533
## CI.mean.0.95 0.34404
## var        31.35375
## std.dev     5.59944
## coef.var    0.59309
## skewness    1.07157
## skew.2SE    6.99601
## kurtosis    1.51475
## kurt.2SE    4.94956
## normtest.W  0.93272
## normtest.p  0.00000

```

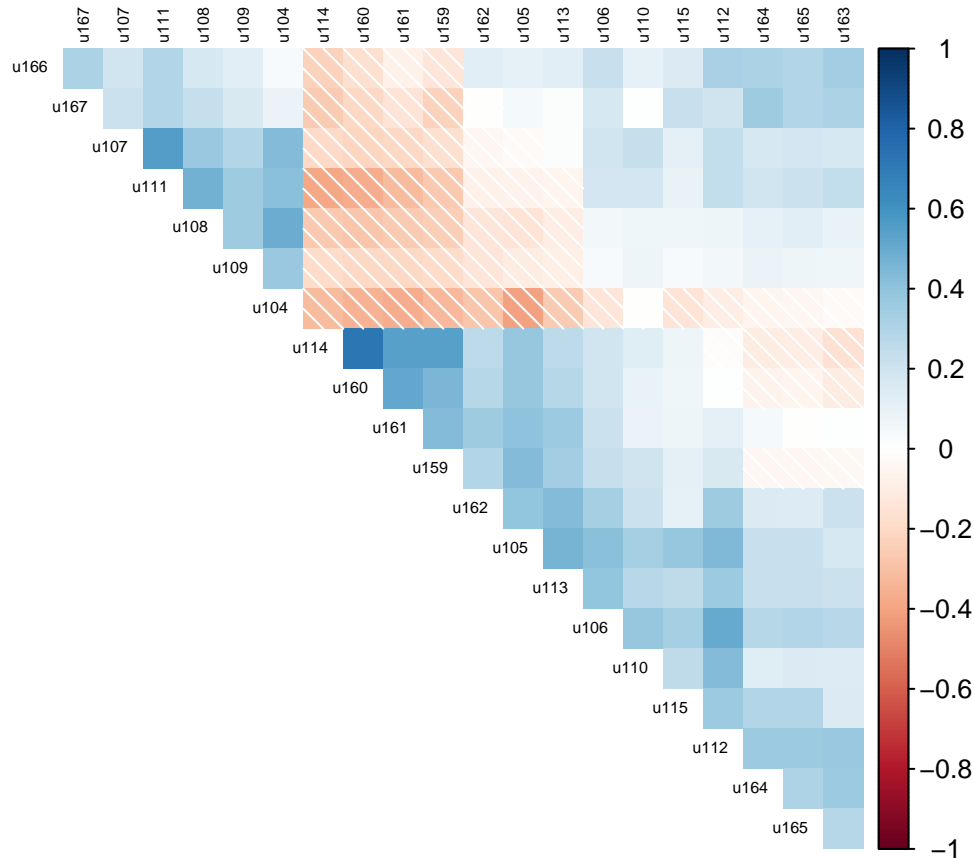
Since the number of observation for each variable is greater than 1000, I chose 0.01 and the significance level.

1.21 various statistics including the *mean*, *median*, *standard error*, and *more*. We are more interested in the *Skewness*, *skew.2SE*, *Kurtosis*, *kurt.2SE* and *normtest.p* value for each of the variables. According to the result of the statistics, the skewness and kurtosis for all the variables are not equal to zero; hence, they are not normally distributed. All the variables are positively skewed, hence, skewed to the right. Meanwhile, two variable u110 (Provides unpaid care) and u112 (Level 1, Level 2 or Apprenticeship) have flat distribution according to the kurtosis result while the remaining are heavy tailed.

According to the result from the `stat.desc` statistics, all the variables have `skew.2SE` values greater than **1.29 and less than -1.29**; hence, all the results of the skewness are significant. Nevertheless, not all all variables have `kurt.2SE` result greater than **1.29 or less than -1.29**, thus making kurtosis result for variables u110, u112, u113, u114, u162, and u166 not significant. Meanwhile, the p-value for the Shapiro test for all the variables is less than 0.01 except variable u112. This means than we will reject the null hypothesis that the distribution if normal for all the variables expect variable u112; hence, variable u112 is normally distributed.

1.3 Kendall's regression correlation plot

```
corrplot(cor(explorData, method = "kendall"), type = "upper",  
         tl.cex=0.5, method = 'shade', order = 'AOE',  
         diag = FALSE, tl.col="black")
```



Since we know that non of the variables are normally distributed and all the variables have duplication, according to 1.1 result, we cannot use Pearson and Spearman's correlation, hence; hence we used the Kendall's correlation The correlation plot for all the 21 variables. It was ordered in such a way to create cluster based on the correlation value. The negatively correlated variables are at the top right corner of the chart, (brown shaded color) while the uncorrelated are in the middle (white color), and the highly correlated variables fill the remaining place.

2.0 Part 2

2.11 Selecting the data needed for the regression analysis

```
regression_data <- OwnLadd %>%  
  select( Total_Population, u104:u115, u159:u167) %>%  
  #converting each column to represent percentage of population  
  mutate(  
    across(u104:u167,  
           function(x){  
             (x/Total_Population)*100  
           })  
  ) %>%  
  #renaming the variables
```

```

rename_with(
  function(x){paste('perc', x, sep = "_")},
  u104:u167
)

```

2.12 Checking the normality of the variables after normalizing them with percentage population

```

stat_view2 <- regression_data %>%
  pastecs::stat.desc(norm = TRUE) %>%
  round(5)
print(stat_view2)

```

##	Total_Population	perc_u104	perc_u105	perc_u106	perc_u107
## nbr.val	1020.00000	1020.00000	1020.00000	1020.00000	1020.00000
## nbr.null	0.00000	0.00000	0.00000	0.00000	0.00000
## nbr.na	0.00000	0.00000	0.00000	0.00000	0.00000
## min	112.00000	7.39814	5.19481	21.38728	4.50450
## max	603.00000	167.64464	70.02519	45.78947	40.90909
## range	491.00000	160.24651	64.83038	24.40219	36.40459
## sum	309392.00000	29788.69613	50669.77192	34889.40043	12132.08690
## median	302.00000	25.94649	50.17012	34.00693	11.19796
## mean	303.32549	29.20460	49.67625	34.20529	11.89420
## SE.mean	1.89171	0.47653	0.21412	0.11934	0.12123
## CI.mean.0.95	3.71209	0.93509	0.42017	0.23418	0.23789
## var	3650.13144	231.62116	46.76486	14.52671	14.99033
## std.dev	60.41632	15.21910	6.83848	3.81139	3.87173
## coef.var	0.19918	0.52112	0.13766	0.11143	0.32551
## skewness	0.31654	2.67054	-0.64267	0.14847	1.48854
## skew.2SE	2.06664	17.43537	-4.19588	0.96931	9.71835
## kurtosis	1.64467	13.12420	1.79828	-0.06276	5.22833
## kurt.2SE	5.37406	42.88419	5.87598	-0.20507	17.08390
## normtest.W	0.98472	0.80166	0.97910	0.99700	0.91652
## normtest.p	0.00000	0.00000	0.00000	0.05181	0.00000
##	perc_u108	perc_u109	perc_u110	perc_u111	perc_u112
## nbr.val	1020.00000	1020.00000	1020.00000	1020.00000	1020.00000
## nbr.null	2.00000	112.00000	0.00000	0.00000	0.00000
## nbr.na	0.00000	0.00000	0.00000	0.00000	0.00000
## min	0.00000	0.00000	1.66667	3.40136	12.33766
## max	15.62500	7.14286	20.07299	69.48052	39.09348
## range	15.62500	7.14286	18.40633	66.07916	26.75582
## sum	3361.30067	947.44008	10176.50313	14962.90809	26473.80726
## median	2.88787	0.73801	9.81161	13.07403	26.00143
## mean	3.29539	0.92886	9.97696	14.66952	25.95471
## SE.mean	0.06036	0.02600	0.08725	0.20927	0.14173
## CI.mean.0.95	0.11844	0.05102	0.17122	0.41065	0.27812
## var	3.71595	0.68942	7.76568	44.67059	20.48995
## std.dev	1.92768	0.83031	2.78670	6.68361	4.52658
## coef.var	0.58496	0.89390	0.27931	0.45561	0.17440
## skewness	1.47157	2.17525	0.29734	1.42929	0.00103
## skew.2SE	9.60758	14.20170	1.94129	9.33150	0.00670
## kurtosis	4.02607	8.33497	0.16815	5.01967	-0.15503
## kurt.2SE	13.15543	27.23504	0.54945	16.40208	-0.50657
## normtest.W	0.90764	0.83133	0.99420	0.91537	0.99849

```

## normtest.p      0.00000    0.00000    0.00056    0.00000    0.52810
##                perc_u113  perc_u114  perc_u115  perc_u159  perc_u160
## nbr.val         1020.00000  1020.00000  1020.00000  1020.00000  1020.00000
## nbr.null        0.00000    0.00000    2.00000    3.00000    1.00000
## nbr.na          0.00000    0.00000    0.00000    0.00000    0.00000
## min             2.50000    5.57377    0.00000    0.00000    0.00000
## max             16.98113    55.10204    15.88235    17.13483    33.33333
## range           14.48113    49.52827    15.88235    17.13483    33.33333
## sum             9463.19070  27339.88020  5069.33276  6492.33324  10753.54681
## median          9.22088    26.58735    4.82121    6.28657    10.42069
## mean            9.27764    26.80380    4.96993    6.36503    10.54269
## SE.mean         0.06613    0.29332    0.06061    0.08101    0.13028
## CI.mean.0.95    0.12977    0.57558    0.11894    0.15897    0.25565
## var             4.46083    87.75823    3.74753    6.69404    17.31302
## std.dev         2.11207    9.36794    1.93585    2.58728    4.16089
## coef.var        0.22765    0.34950    0.38951    0.40648    0.39467
## skewness        0.16368    0.08193    0.84964    0.48965    0.45260
## skew.2SE        1.06862    0.53490    5.54712    3.19682    2.95494
## kurtosis        0.18777    -0.53207    2.31945    0.63539    0.88780
## kurt.2SE        0.61356    -1.73859    7.57897    2.07617    2.90095
## normtest.W      0.99683    0.99211    0.96501    0.98592    0.98680
## normtest.p      0.03888    0.00003    0.00000    0.00000    0.00000
##                perc_u161  perc_u162  perc_u163  perc_u164  perc_u165  perc_u166
## nbr.val         1020.00000  1020.00000  1020.00000  1020.00000  1020.00000  1020.00000
## nbr.null        1.00000    0.00000    0.00000    0.00000    3.00000    19.00000
## nbr.na          0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
## min             0.00000    1.19760    0.29412    0.28011    0.00000    0.00000
## max             18.93204    15.18987    11.32075    10.87866    8.22785    6.25000
## range           18.93204    13.99227    11.02664    10.59855    8.22785    6.25000
## sum             8133.15776  7644.60336  4621.90516  3790.27954  3353.48500  2082.86707
## median          7.74110    7.47815    4.39375    3.62720    3.15789    1.91235
## mean            7.97368    7.49471    4.53128    3.71596    3.28773    2.04203
## SE.mean         0.09174    0.06631    0.05636    0.04782    0.04188    0.03431
## CI.mean.0.95    0.18002    0.13012    0.11060    0.09385    0.08218    0.06732
## var             8.58441    4.48521    3.24006    2.33291    1.78900    1.20058
## std.dev         2.92992    2.11783    1.80002    1.52739    1.33753    1.09571
## coef.var        0.36745    0.28258    0.39724    0.41103    0.40683    0.53658
## skewness        0.51751    0.18721    0.50625    0.45797    0.49683    0.54326
## skew.2SE        3.37870    1.22227    3.30518    2.98995    3.24367    3.54681
## kurtosis        0.51649    0.16297    0.48052    0.47915    0.18568    0.08331
## kurt.2SE        1.68765    0.53252    1.57012    1.56566    0.60671    0.27224
## normtest.W      0.98444    0.99700    0.98456    0.98619    0.98432    0.97840
## normtest.p      0.00000    0.05166    0.00000    0.00000    0.00000    0.00000
##                perc_u167
## nbr.val         1020.00000
## nbr.null        5.00000
## nbr.na          0.00000
## min             0.00000
## max             12.13389
## range           12.13389
## sum             3197.66947
## median          2.79070
## mean            3.13497
## SE.mean         0.05561

```

```
## CI.mean.0.95    0.10911
## var            3.15380
## std.dev        1.77589
## coef.var       0.56648
## skewness       0.89427
## skew.2SE       5.83847
## kurtosis       0.81193
## kurt.2SE       2.65303
## normtest.W     0.94818
## normtest.p     0.00000
```

Variables `u162`, `u112`, `u113`, and `u106` are normally distributed after normalizing the data. The p-value of the aforementioned four variables is greater than 0.01; hence we can reject the null hypothesis that they are not normally distributed. Nevertheless, the other variables have p-values less than 0.01; hence we accept the null hypothesis -normally distributed.

2.13 Selecting the variable to be used for the regression analysis

How main focus is to check the relationship between variable `perc_u106` (percentage of people with good Health), `perc_u112` (percentage of people with Level 1, Level 2 or Apprenticeship qualifications) and `perc_u162` (percentage of people with Administrative and secretarial occupations). The two independent variables `perc_u112` and `perc_u162` were chosen because they are normally distributed and they are likely uncorrelated.

```
forregression <- regression_data %>%
  select(perc_u106, perc_u112, perc_u162)
```

Since the three variables `perc_u106` (percentage of people with good health), `perc_u112` (percentage of people with Level 1, Level 2 or Apprenticeship qualifications) and `perc_u162` () meet the assumptions of Pearson correlation, we will run a pearson correlation

2.21 Pearson correlation between variable `perc_u106` and `perc_u112`

```
forregression %$%
cor.test(perc_u106, perc_u112)
```

```
##
## Pearson's product-moment correlation
##
## data: perc_u106 and perc_u112
## t = 3.6591, df = 1018, p-value = 0.0002661
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.05292495 0.17410127
## sample estimates:
##      cor
## 0.1139368
```

2.21 shows the result of the correlation between the percentage of people with good health and the percentage of people with Level 1, Level 2 or Apprenticeship qualifications. According to the result, we reject the null hypothesis that there is no correlation between the two variables `perc_u106` and `perc_u112` since the p-value is less than 0.01; hence, there is relationship between the two variables. The correlation is positive since the `cor` value is **0.1139368**. However, the correlation is very weak as the two variables share only **1.2% variability**.

2.22 Pearson regression between variables perc_u106 and perc_u162

```
forregression %$$
cor.test( perc_u106, perc_u162)

##
## Pearson's product-moment correlation
##
## data:  perc_u106 and perc_u162
## t = 3.5233, df = 1018, p-value = 0.0004451
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.04870619 0.16999679
## sample estimates:
##      cor
## 0.1097601
```

2.22 shows the result of the correlation between the percentage of people with good health and the percentage of people with Administrative and secretarial occupations. According to the result, we reject the null hypothesis that there is no correlation between the two variables **perc_u106** and **perc_u162** since the p-value is less than 0.01; hence, there is relationship between the two variable. The correlation is positive since the **cor** value is **0.1097601**. However, the correlation is very weak as the two variables share only **1.2% variability**.

2.31 Regression analysis between variable perc_u106 (dependent) ~ perc_u114 + perc_u165(Independent)

```
health_model <- forregression %$$
lm(perc_u106 ~ perc_u112 + perc_u162)
```

Percentage Population with good Health = (Percentage with level 4 qualification + Percentage doing customer service occupation) + error.

#2.32 Summary of the model

```
summary(health_model)

##
## Call:
## lm(formula = perc_u106 ~ perc_u112 + perc_u162)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.8891  -2.5656  -0.2614   2.5220  11.5524
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  30.79670    0.75456  40.814  < 2e-16 ***
## perc_u112     0.08282    0.02649   3.126  0.00182 **
## perc_u162     0.16799    0.05662   2.967  0.00308 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.774 on 1017 degrees of freedom
## Multiple R-squared:  0.02145,    Adjusted R-squared:  0.01953
## F-statistic: 11.15 on 2 and 1017 DF,  p-value: 1.625e-05
```

The summary of the model indicates that:

- The p-value is **0.00001625**: **p-value < 0.01**; Hence the model is significant. We can reject the null hypothesis that none of the predictors have relationship with the response variable.
 - This result is gotten by comparing the F-statistic to F distribution **11.15** where the degrees of freedom is **(2, 1017)**
 - $F(2, 1017) = 11.15$
 - Adjusted R-squared = **0.01953**
- Coefficient
 - the coefficient = **30.79670 (significant)**
 - The coefficient of slope for % of people with with Level 1, Level 2 or Apprenticeship qualifications is estimated as **0.08282 (significant)**
 - The coefficient of slope for % of people with with Administrative and secretarial occupations is estimated as **0.16799 (Significant)**

2.4 Test for normality, homoscedasticity, independence and multicollinearity

2.41 Test for normality

```
health_model %>%  
  stats::rstandard() %>%  
  stats::shapiro.test()
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  .  
## W = 0.99633, p-value = 0.01678
```

2.42 Test for homoscedasticity

```
health_model %>%  
  lmtest::bptest()
```

```
##  
##  studentized Breusch-Pagan test  
##  
## data:  .  
## BP = 1.7153, df = 2, p-value = 0.4242
```

2.43 Test for independence

```
health_model %>%  
  lmtest::dwtest()
```

```
##  
##  Durbin-Watson test  
##  
## data:  .  
## DW = 1.9789, p-value = 0.3613  
## alternative hypothesis: true autocorrelation is greater than 0
```

2.43 Test for multicollinearity

```
health_model %>%  
  vif()
```

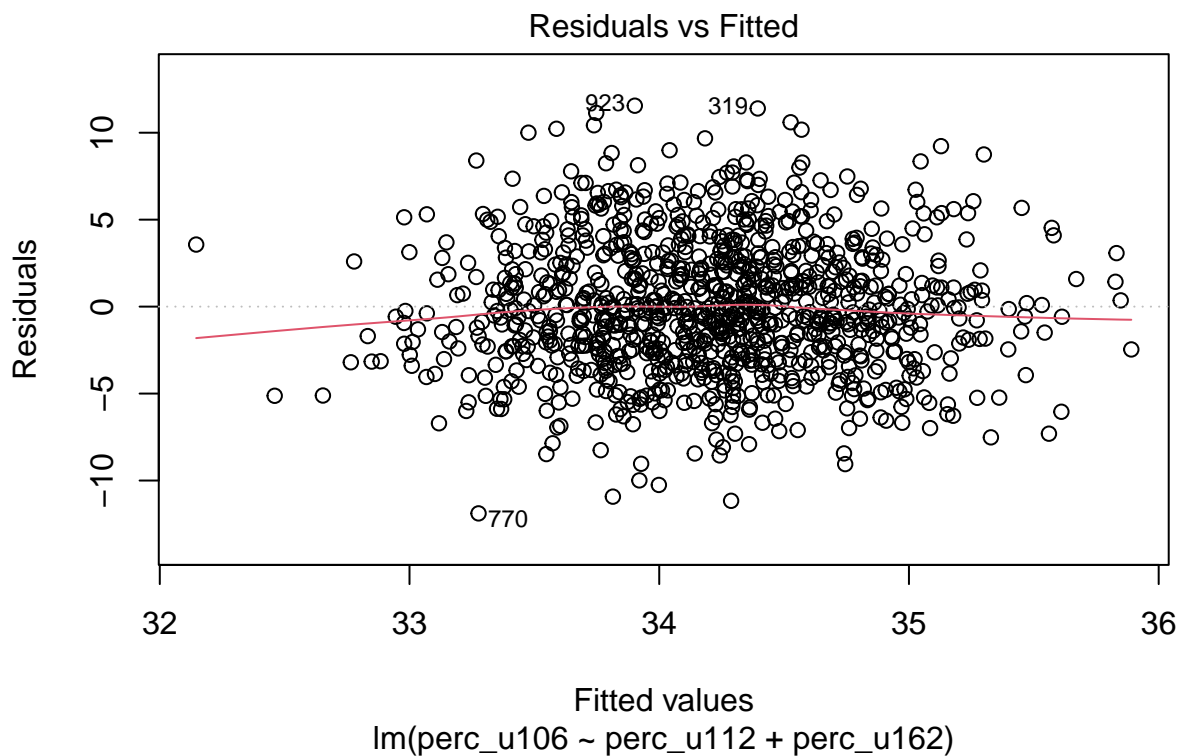
```
## perc_u112 perc_u162  
## 1.028645 1.028645
```

The output of the model indicates that the model fits ($F(2, 1017) = 11.15$), **p-Value < 0.01**. However, the model on the percentage of people with Level 1, Level 2 or Apprenticeship qualifications and

people with Administrative and secretarial occupations can only predict 2% of people with good health. The model have normally distributed residuals (**Shapiro-Wilk test**, $W=0.99$, $p=0.01678$), no multicollinearity with average **VIF 1.028645**, the residuals satisfy the assumption of homoscedasticity (**Breusch-Pagan test**, $BP = 1.7153$, $p\text{-value} = 0.4242$) and assumptions of independence (**Durbin-Watson test**, $DW = 1.9789$, $p\text{-value} = 0.3613$), However, we can say that the model is partially robust because of the low adjusted R-squared value. Based on the result, the model indicates that for every one percent increase in the percentage of people with with Level 1, Level 2 or Apprenticeship qualifications, there will be 0.08282 increase in the percentage of people with good health. Similarly, for every one percent increase in the percentage of people with Administrative and secretarial occupations, there will be 0.16799 increase in the percentage of people with good health.

2.51 Residual vs Fitted plot

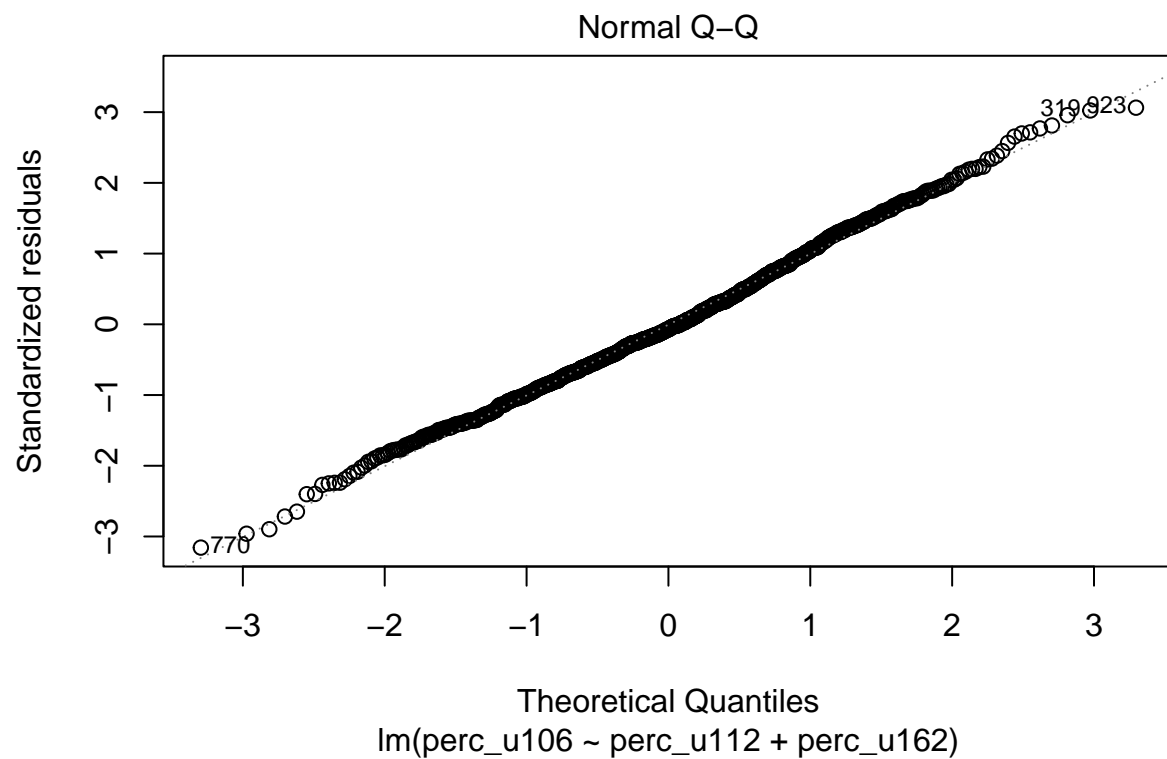
```
health_model %>%
  plot(which = c(1))
```



2.51 gives an insight into the homoskedasticity of the residual. Since the red line is close to the dash line, the linearity of the model seems to hold well, the model is homoskedastic as the variance is not increasing, and point 770, 923 and 319 are outliers.

2.52 Normal Q-Q plot

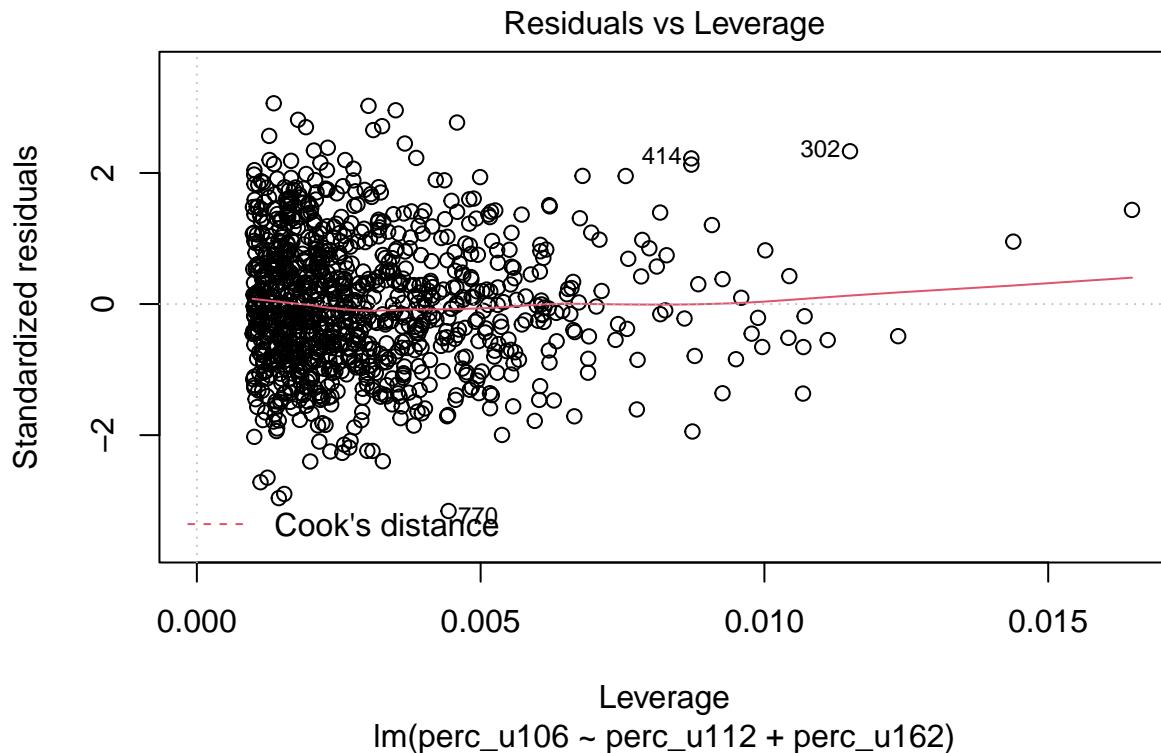
```
health_model %>%
  plot(which = c(2))
```



2.52 shows the normality of the residuals. The fact that the qq plot lies on the line shows that it is normally distributed.

2.53 Residual vs Leverage plot

```
health_model %>%
  plot(which = c(5))
```

2.53 gives insight about the Cook's distance. No point fall outside the Cook's Distance, indicating that there is no influential point in the regression model.

Part 3

3.1

Thanks to the classes we had in the first and second part of this course, I was able to achieve this task with not so much difficulty. As suggested in the course, I first observed and visualize my data to understand the types of data I have using key functions like `describe()`, `str()`, `stat.desc()`, `histogram`, `ggplot` and `qqplot`. I also tried to observe the relationship between the variables with correlation analysis before running a regression model. Afterwards, I tried to check the robustness of my model with functions like `shapiro.test()`, `vif()`, `bptest()`, `dwtest()` and more. All these helped me to achieve the task. We with known I had in the first part, the data predictability was quite straight forward with libraries like `dplyr`, `magrittr`, `tidyverse` and more.

Reference

1. Sthda.com. 2022. Correlation matrix : A quick start guide to analyze, format and visualize a correlation matrix using R software - Easy Guides - Wiki - STHDA. [online] Available at: <http://www.sthda.com/english/wiki/correlation-matrix-a-quick-start-guide-to-analyze-format-and-visualize-a-correlation-matrix-using-r-software> [Accessed 6 January 2022].
2. Rdocumentation.org. 2022. describe function - RDocumentation. [online] Available at: <https://www.rdocumentation.org/packages/Hmisc/versions/4.6-0/topics/describe> [Accessed 6 January 2022].
3. Medium. 2022. How to Create a Correlation Matrix with Too Many Variables in R. [online] Available at: <https://towardsdatascience.com/how-to-create-a-correlation-matrix-with-too-many-variables-309cc0c0a57> [Accessed 6 January 2022].
4. Boostedml. 2022. Linear Regression Plots: Fitted vs Residuals - Boostedml. [online] Available at: <https://boostedml.com/2019/03/linear-regression-plots-fitted-vs-residuals.html> [Accessed 6 January 2022].

5. Sabbata, S., 2022. Chapter 8 Regression analysis | R for Geographic Data Science. [online] Sdesabbata.github.io. Available at: <https://sdesabbata.github.io/r-for-geographic-data-science/regression-analysis.html> [Accessed 6 January 2022].

Appendix

1. This document includes information from public sector licensed under the **Open Government Licence v3.0** from the **Office for National Statistics**.

2. **VariableCode | VariableDescription**

u104:	Day-to-day activities limited a lot or a little Standardised Illness Ratio
u105:	Very good health
u106:	Good health
u107:	Fair health
u108:	Bad health
u109:	Very bad health
u110:	Provides unpaid care
u111:	No qualifications
u112:	Highest level of qualification: Level 1, Level 2 or Apprenticeship
u113:	Highest level of qualification: Level 3 qualifications
u114:	Highest level of qualification: Level 4 qualifications and above
u115:	Schoolchildren and full-time students: Age 16 and over
u159:	Managers, directors and senior officials
u160:	Professional occupations
u161:	Associate professional and technical occupations
u162:	Administrative and secretarial occupations
u163:	Skilled trades occupations
u164:	Caring, leisure and other service occupations
u165:	Sales and customer service occupations
u166:	Process, plant and machine operatives
u167:	Elementary occupations