Analysis of Lead Groups

Yunting Chiu

10/25/2020

```
library(mosaic)
```

```
## Registered S3 method overwritten by 'mosaic':
##   method                           from
##   fortify.SpatialPolygonsDataFrame ggplot2
```

```
##
## The 'mosaic' package masks several functions from core packages in order to add
## additional features.  The original behavior of these functions should not be affected by this.
```

```
##
## Attaching package: 'mosaic'
```

```
## The following objects are masked from 'package:dplyr':
##
##     count, do, tally
```

```
## The following object is masked from 'package:Matrix':
##
##     mean
```

```
## The following object is masked from 'package:ggplot2':
##
##     stat
```

```
## The following objects are masked from 'package:stats':
##
##     binom.test, cor, cor.test, cov, fivenum, IQR, median, prop.test,
##     quantile, sd, t.test, var
```

```
## The following objects are masked from 'package:base':
##
##     max, mean, min, prod, range, sample, sum
```

```
library(beanplot)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------- tidyverse 1.3.0 --
```

```
## v tibble  3.0.3     v purrr   0.3.4
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------ tidyverse_conflicts() --
## x mosaic::count()            masks dplyr::count()
## x purrr::cross()             masks mosaic::cross()
## x mosaic::do()               masks dplyr::do()
## x tidyr::expand()            masks Matrix::expand()
```

```
## x dplyr::filter()            masks stats::filter()
## x ggstance::geom_errorbarh() masks ggplot2::geom_errorbarh()
## x dplyr::lag()               masks stats::lag()
## x tidyr::pack()              masks Matrix::pack()
## x mosaic::stat()             masks ggplot2::stat()
## x mosaic::tally()            masks dplyr::tally()
## x tidyr::unpack()            masks Matrix::unpack()
```

```
lead <- read_csv("lead.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

```
lead
```

```
## # A tibble: 124 x 40
##       id  area   age   sex iqv_inf iqv_comp iqv_ar iqv_ds iqv_raw iqp_pc iqp_bd
##    <dbl> <dbl> <dbl> <dbl>   <dbl>    <dbl>  <dbl>  <dbl>   <dbl>  <dbl>  <dbl>
##  1   101     3  1101     1       3        4      3      5      15     10      8
##  2   102     3   905     1       7        9      7      6      29      8      7
##  3   103     3  1101     1       4        9      5      3      21     10      7
##  4   104     2   611     1       4        6      6      6      22      5      8
##  5   105     1  1103     1       5        4      8      5      22      5     10
##  6   106     2   606     1       5       12     11      9      37     14      7
##  7   107     3   611     1       7        9     10      7      33     10      8
##  8   108     1  1500     2       3        1      3      6      13      6      2
##  9   109     2   702     2      13       10     14     13      50      8     15
## 10   110     2   703     1       7        9     12      9      37      6      9
## # ... with 114 more rows, and 29 more variables: iqp_oa <dbl>, iqp_cod <dbl>,
## #   iqp_raw <dbl>, hh_index <dbl>, iqv <dbl>, iqp <dbl>, iqf <dbl>,
## #   iq_type <dbl>, lead_typ <dbl>, ld72 <dbl>, ld73 <dbl>, fst2yrs <dbl>,
## #   totyrs <dbl>, pica <dbl>, colic <dbl>, clumsi <dbl>, irrit <dbl>,
## #   convul <dbl>, `@2plat_r` <dbl>, `@2plat_l` <dbl>, visrea_r <dbl>,
## #   visrea_l <dbl>, audrea_r <dbl>, audrea_l <dbl>, fwt_r <dbl>, fwt_l <dbl>,
## #   hyperact <dbl>, MAXFT <dbl>, GROUP <dbl>
```

- EDA

```
summary(lead$lead_typ)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   1.000   1.548   2.000   3.000
```

```
favstats(MAXFT ~ lead_typ, data = lead)
```

```
##   lead_typ min   Q1 median    Q3 max    mean       sd  n missing
## 1        1  13 49.0   53.5 61.25  84 54.4375 12.05658 64      14
## 2        2  13 40.5   48.0 53.00  58 44.0000 12.65350 19       5
## 3        3  35 41.5   51.0 57.50  83 51.5000 12.94604 16       6
```

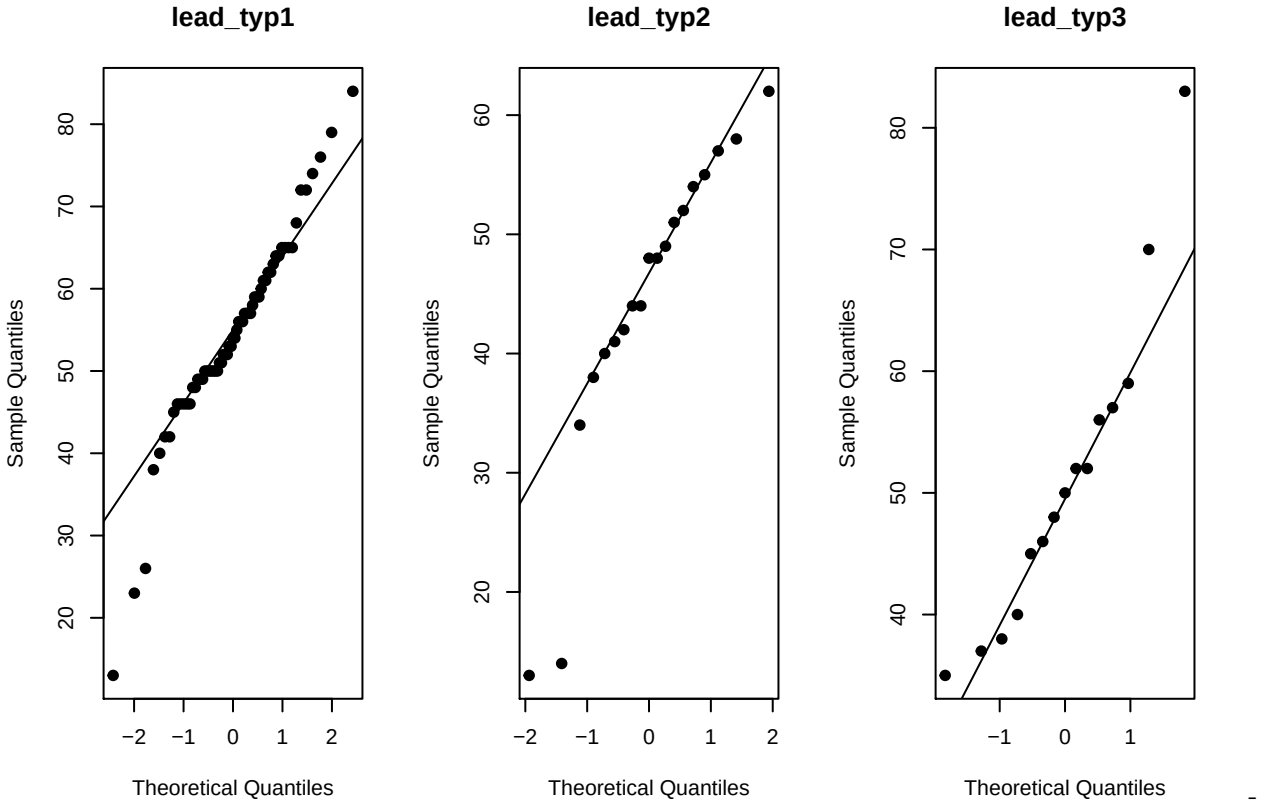
- remove NAs of MAXFT

```
lead %>%
  filter(!is.na(MAXFT)) -> leadRealMAXFT
favstats(MAXFT ~ lead_typ, data = leadRealMAXFT)
```

```
##   lead_typ min   Q1 median    Q3 max    mean       sd  n missing
## 1        1  13 49.0   53.5 61.25  84 54.4375 12.05658 64       0
## 2        2  13 40.5   48.0 53.00  58 44.0000 12.65350 19       0
## 3        3  35 41.5   51.0 57.50  83 51.5000 12.94604 16       0
```

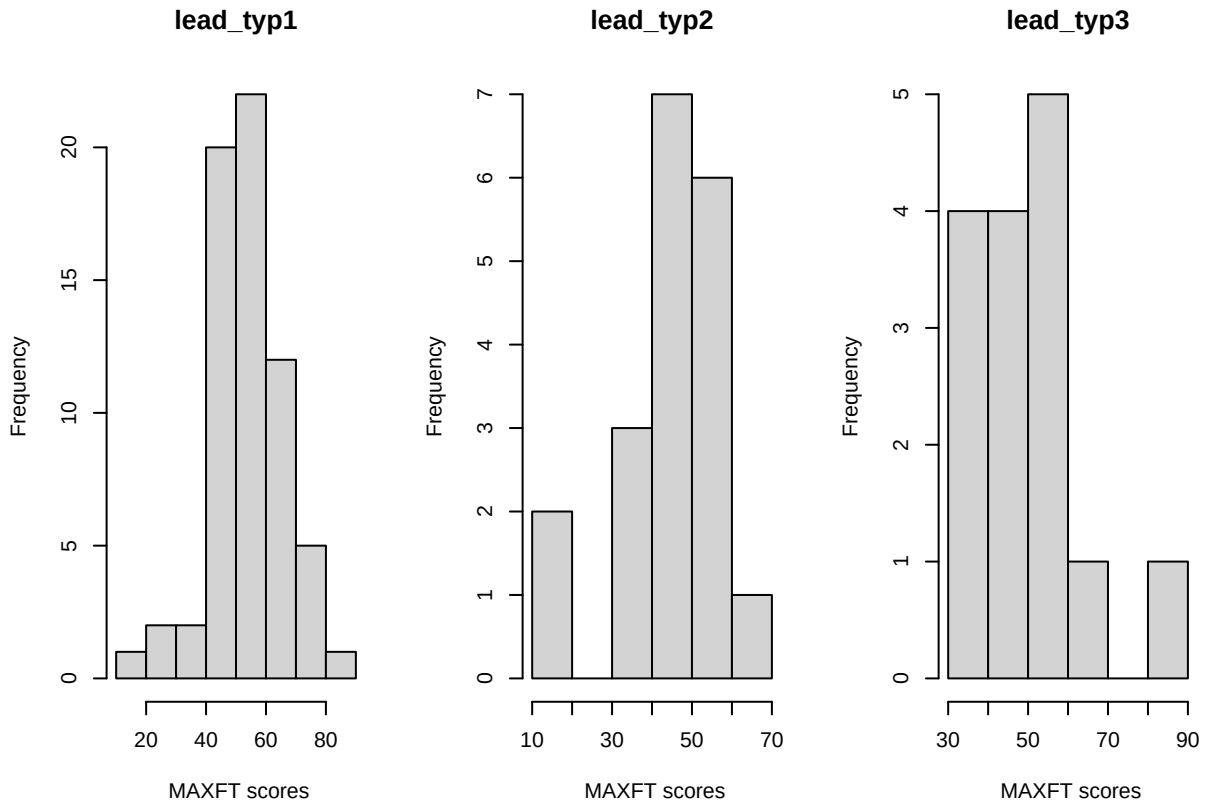
- qqplot

```
# lead_typ1
par(mfrow=c(1,3))
qqnorm(leadRealMAXFT$MAXFT[lead$lead_typ == 1],main="lead_typ1", pch = 19)
qqline(leadRealMAXFT$MAXFT[lead$lead_typ == 1])

# lead_typ2
qqnorm(leadRealMAXFT$MAXFT[lead$lead_typ == 2],main="lead_typ2", pch = 19)
qqline(leadRealMAXFT$MAXFT[lead$lead_typ == 2])

# lead_typ3
qqnorm(leadRealMAXFT$MAXFT[lead$lead_typ == 3],main="lead_typ3", pch = 19)
qqline(leadRealMAXFT$MAXFT[lead$lead_typ == 3])
```



histogram

```
par(mfrow=c(1,3)) # creates a single 1 by 3 grid of our three histograms
hist(leadRealMAXFT$MAXFT[lead$lead_typ == 1],main="lead_typ1", xlab = ' MAXFT scores')
hist(leadRealMAXFT$MAXFT[lead$lead_typ == 2],main="lead_typ2", xlab = ' MAXFT scores')
hist(leadRealMAXFT$MAXFT[lead$lead_typ == 3],main="lead_typ3", xlab = ' MAXFT scores')
```



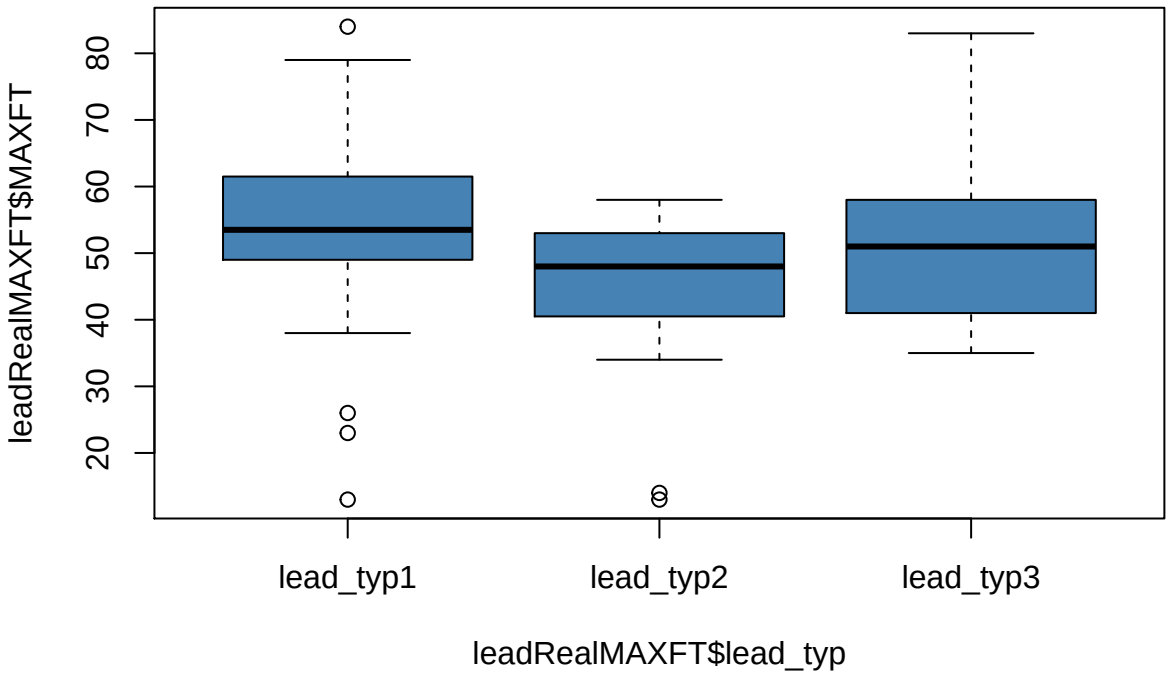
- Shapiro-Wilk Normality Test

```
shapiro.test(leadRealMAXFT$MAXFT)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  leadRealMAXFT$MAXFT
## W = 0.9556, p-value = 0.002108
```

- boxplots

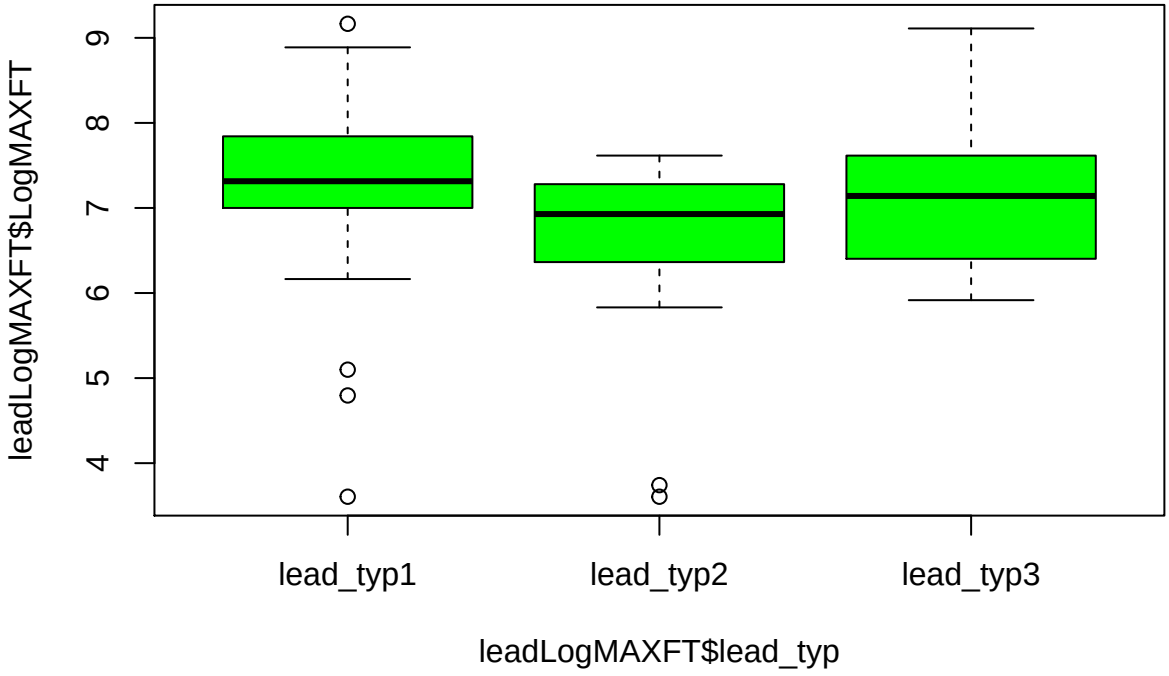
```
boxplot(leadRealMAXFT$MAXFT ~ leadRealMAXFT$lead_typ,
        names= c("lead_typ1","lead_typ2","lead_typ3"),col = "steelblue")
```



- Data transformation (log)

```
leadRealMAXFT %>%
  mutate(LogMAXFT = sqrt(MAXFT)) -> leadLogMAXFT
```

```
boxplot(leadLogMAXFT$LogMAXFT ~ leadLogMAXFT$lead_typ,
        names= c("lead_typ1","lead_typ2","lead_typ3"),col = "green")
```



- extract the lead_type

```
lead %>%
  filter(lead_typ == 1) -> lead_typ1
lead_typ1 %>%
```

```
summarize(across(everything(), ~sum(is.na(.))))
```

```
## # A tibble: 1 x 40
##      id  area   age   sex iqv_inf iqv_comp iqv_ar iqv_ds iqv_raw iqp_pc iqp_bd
##   <int> <int> <int> <int>   <int>    <int>  <int>  <int>   <int>  <int>  <int>
## 1     0     0     0     0       0        0      0      0       0      0      0
## # ... with 29 more variables: iqp_oa <int>, iqp_cod <int>, iqp_raw <int>,
## #   hh_index <int>, iqv <int>, iqp <int>, iqf <int>, iq_type <int>,
## #   lead_typ <int>, ld72 <int>, ld73 <int>, fst2yrs <int>, totyrs <int>,
## #   pica <int>, colic <int>, clumsi <int>, irrit <int>, convul <int>,
## #   `@2plat_r` <int>, `@2plat_l` <int>, visrea_r <int>, visrea_l <int>,
## #   audrea_r <int>, audrea_l <int>, fwt_r <int>, fwt_l <int>, hyperact <int>,
## #   MAXFT <int>, GROUP <int>
```

```
lead %>%
  filter(lead_typ == 2) -> lead_typ2
lead_typ2 %>%
summarize(across(everything(), ~sum(is.na(.))))
```

```
## # A tibble: 1 x 40
##      id  area   age   sex iqv_inf iqv_comp iqv_ar iqv_ds iqv_raw iqp_pc iqp_bd
##   <int> <int> <int> <int>   <int>    <int>  <int>  <int>   <int>  <int>  <int>
## 1     0     0     0     0       0        0      0      0       0      0      0
## # ... with 29 more variables: iqp_oa <int>, iqp_cod <int>, iqp_raw <int>,
## #   hh_index <int>, iqv <int>, iqp <int>, iqf <int>, iq_type <int>,
## #   lead_typ <int>, ld72 <int>, ld73 <int>, fst2yrs <int>, totyrs <int>,
## #   pica <int>, colic <int>, clumsi <int>, irrit <int>, convul <int>,
## #   `@2plat_r` <int>, `@2plat_l` <int>, visrea_r <int>, visrea_l <int>,
## #   audrea_r <int>, audrea_l <int>, fwt_r <int>, fwt_l <int>, hyperact <int>,
## #   MAXFT <int>, GROUP <int>
```

```
lead %>%
  filter(lead_typ == 3) -> lead_typ3
lead_typ3 %>%
summarize(across(everything(), ~sum(is.na(.))))
```

```
## # A tibble: 1 x 40
##      id  area   age   sex iqv_inf iqv_comp iqv_ar iqv_ds iqv_raw iqp_pc iqp_bd
##   <int> <int> <int> <int>   <int>    <int>  <int>  <int>   <int>  <int>  <int>
## 1     0     0     0     0       0        0      0      0       0      0      0
## # ... with 29 more variables: iqp_oa <int>, iqp_cod <int>, iqp_raw <int>,
## #   hh_index <int>, iqv <int>, iqp <int>, iqf <int>, iq_type <int>,
## #   lead_typ <int>, ld72 <int>, ld73 <int>, fst2yrs <int>, totyrs <int>,
## #   pica <int>, colic <int>, clumsi <int>, irrit <int>, convul <int>,
## #   `@2plat_r` <int>, `@2plat_l` <int>, visrea_r <int>, visrea_l <int>,
## #   audrea_r <int>, audrea_l <int>, fwt_r <int>, fwt_l <int>, hyperact <int>,
## #   MAXFT <int>, GROUP <int>
```

- remove outliers based on mean

```
lead %>%
  filter(!is.na(MAXFT)) -> leadRealMAXFT
favstats(MAXFT ~ lead_typ, data = leadRealMAXFT)
```

```
##   lead_typ min   Q1 median    Q3 max    mean       sd  n missing
## 1        1  13 49.0   53.5 61.25  84 54.4375 12.05658 64       0
```

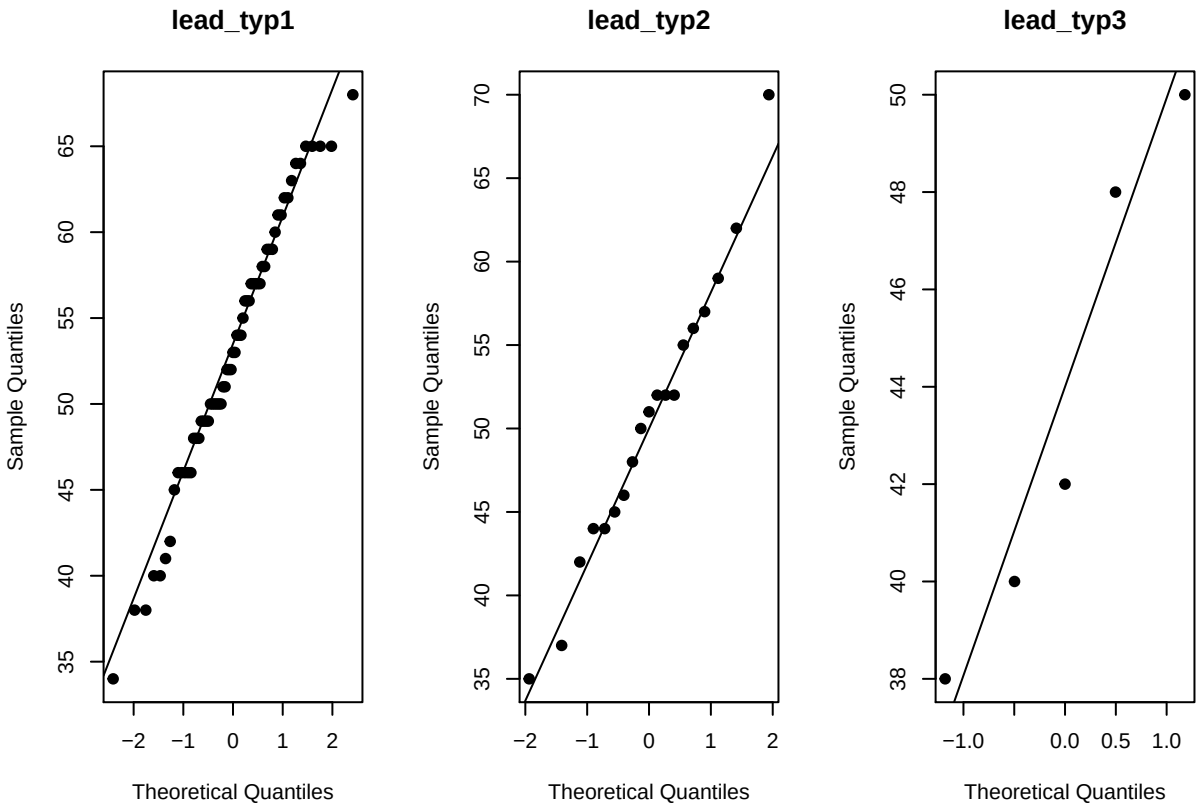
```
## 2        2  13 40.5   48.0 53.00  58 44.0000 12.65350 19       0
## 3        3  35 41.5   51.0 57.50  83 51.5000 12.94604 16       0
```

```
leadRealMAXFT %>%
  filter(MAXFT >= 30 & MAXFT <= 70) -> leadRmOutliers
favstats(MAXFT ~ lead_typ, data = leadRmOutliers)
```

```
##   lead_typ min Q1 median   Q3 max     mean        sd  n missing
## 1        1  38 49     53 59.0  68 53.90909  7.053428 55       0
## 2        2  34 42     48 54.0  58 47.58824  7.080420 17       0
## 3        3  35 41     50 56.5  70 49.40000 10.196638 15       0
```

- re-run the analyses: qqplot

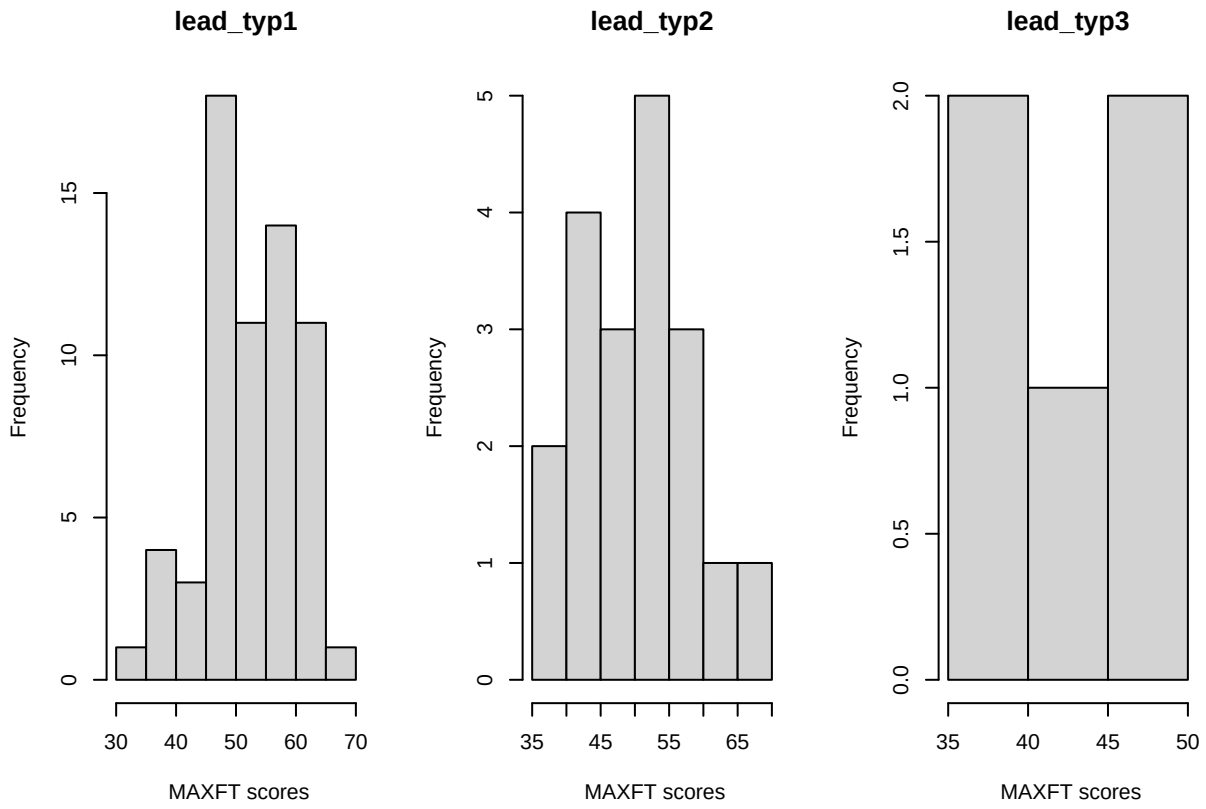
```
# lead_typ1
par(mfrow=c(1,3))
qqnorm(leadRmOutliers$MAXFT[lead$lead_typ == 1],main="lead_typ1", pch = 19)
qqline(leadRmOutliers$MAXFT[lead$lead_typ == 1])

# lead_typ2
qqnorm(leadRmOutliers$MAXFT[lead$lead_typ == 2],main="lead_typ2", pch = 19)
qqline(leadRmOutliers$MAXFT[lead$lead_typ == 2])

# lead_typ3
qqnorm(leadRmOutliers$MAXFT[lead$lead_typ == 3],main="lead_typ3", pch = 19)
qqline(leadRmOutliers$MAXFT[lead$lead_typ == 3])
```



- histogram

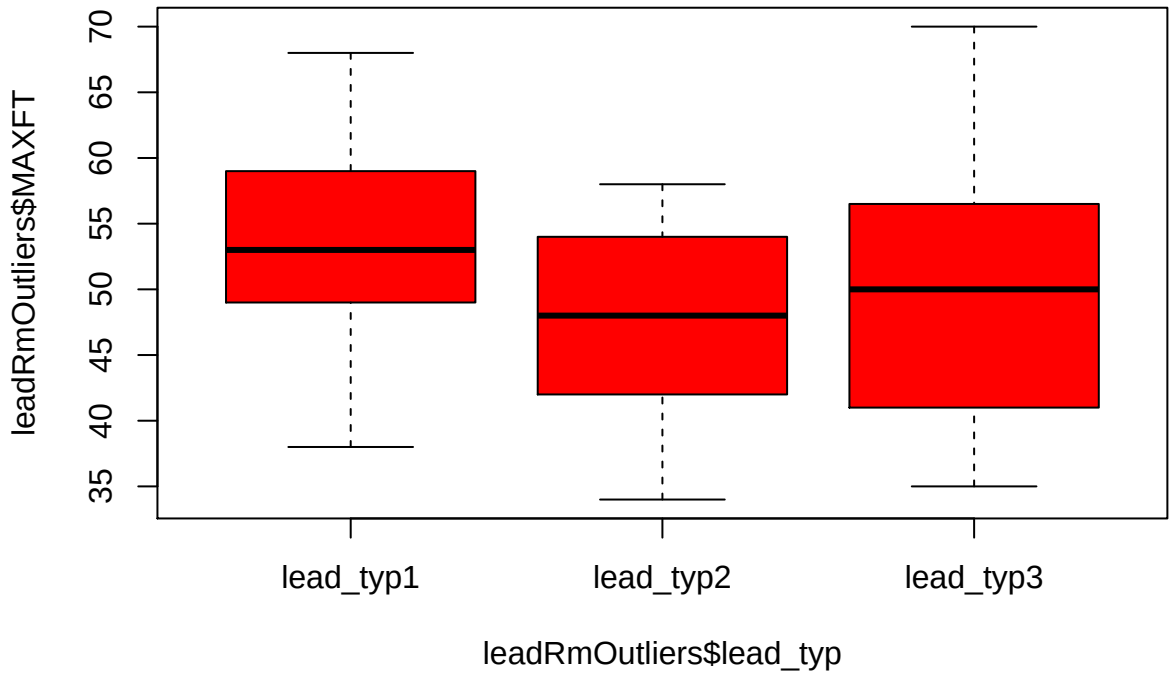
```
par(mfrow=c(1,3)) # creates a single 1 by 3 grid of our three histograms
hist(leadRmOutliers$MAXFT[lead$lead_typ == 1],main="lead_typ1", xlab = ' MAXFT scores')
```

```
hist(leadRmOutliers$MAXFT[lead$lead_typ == 2],main="lead_typ2", xlab = ' MAXFT scores')
hist(leadRmOutliers$MAXFT[lead$lead_typ == 3],main="lead_typ3", xlab = ' MAXFT scores')
```



- – boxplots

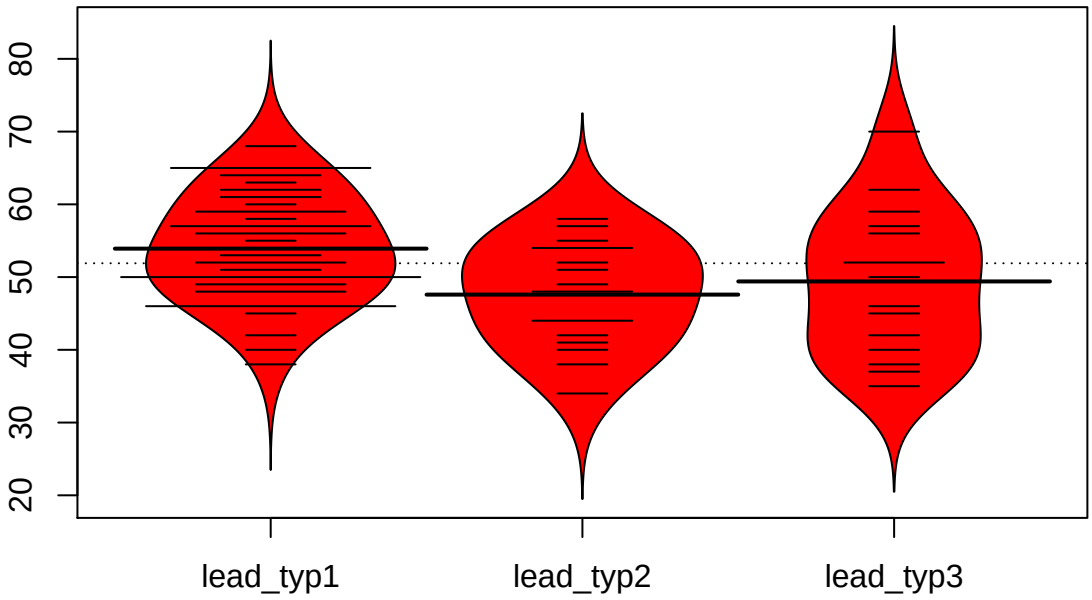
```
boxplot(leadRmOutliers$MAXFT ~ leadRmOutliers$lead_typ,
        names= c("lead_typ1","lead_typ2","lead_typ3"),col = "red")
```



- bean-

plot

```
beanplot(leadRmOutliers$MAXFT ~ leadRmOutliers$lead_typ, names = c ("lead_typ1","lead_typ2","lead_typ3")
```



- Shapiro-Wilk Normality Test

```
shapiro.test(leadRmOutliers$MAXFT)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  leadRmOutliers$MAXFT
## W = 0.9887, p-value = 0.6581
```

- ANOVA model

```
leadRmOutliers %>%
  mutate(lead_typ = as.factor(lead_typ)) -> leadRmOutliers
model.fit <- aov(MAXFT ~ lead_typ, data = leadRmOutliers)  # aov = anova model
anova(model.fit) # anova table
```

```
## Analysis of Variance Table
##
## Response: MAXFT
##           Df Sum Sq Mean Sq F value   Pr(>F)
## lead_typ   2  631.8  315.90   5.367 0.006404 **
## Residuals 84 4944.3   58.86
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#model.fit <- lm(MAXFT ~ lead_typ, data = leadRmOutliers)  # aov = anova model
#anova(model.fit) # anova table
```

- RQ2

```
tout <- t.test(leadRmOutliers$MAXFT[leadRmOutliers$lead_typ == 2],
               leadRmOutliers$MAXFT[leadRmOutliers$lead_typ == 3])
tout
```

```
##
##  Welch Two Sample t-test
##
## data:  leadRmOutliers$MAXFT[leadRmOutliers$lead_typ == 2] and leadRmOutliers$MAXFT[leadRmOutliers$lea
## t = -0.57639, df = 24.557, p-value = 0.5696
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -8.291454  4.667925
## sample estimates:
## mean of x mean of y
##  47.58824  49.40000
```

- RQ2

```
toutRQ <- t.test(leadRmOutliers$MAXFT[leadRmOutliers$lead_typ == 2],
                 leadRmOutliers$MAXFT[leadRmOutliers$lead_typ == 3],
                 alternative = "greater")
toutRQ
```

```
##
##  Welch Two Sample t-test
##
## data:  leadRmOutliers$MAXFT[leadRmOutliers$lead_typ == 2] and leadRmOutliers$MAXFT[leadRmOutliers$lea
## t = -0.57639, df = 24.557, p-value = 0.7152
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  -7.184704       Inf
## sample estimates:
## mean of x mean of y
##  47.58824  49.40000
```

```
toutRQ$stderr
```

```
## [1] 3.143309
```

- All pairwise comparisons

```
library(DescTools)
```

```
##
## Attaching package: 'DescTools'
```

```
## The following object is masked from 'package:mosaic':
##
##     MAD
```

```
PostHocTest(model.fit, method = "lsd")
```

```
##
##   Posthoc multiple comparisons of means : Fisher LSD
##     95% family-wise confidence level
##
## $lead_typ
##          diff     lwr.ci      upr.ci   pval
## 2-1 -6.320856 -10.554566 -2.08714536 0.0039 **
## 3-1 -4.509091  -8.953180 -0.06500204 0.0468 *
## 3-2  1.811765  -3.592861  7.21638999 0.5068
##
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- All pairwise comparisons for orginal data

```
leadRealMAXFT %>%
  mutate(lead_typ = as.factor(lead_typ)) -> leadRealMAXFT
model.fit01 <- aov(MAXFT ~ lead_typ, data = leadRealMAXFT)  # aov = anova model
# anova(model.fit01) # anova table
PostHocTest(model.fit01, method = "lsd")
```

```
##
##   Posthoc multiple comparisons of means : Fisher LSD
##     95% family-wise confidence level
##
## $lead_typ
##         diff      lwr.ci    upr.ci   pval
## 2-1 -10.4375 -16.8227981 -4.052202 0.0016 **
## 3-1  -2.9375  -9.7688090  3.893809 0.3955
## 3-2   7.5000  -0.7928946 15.792895 0.0758 .
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```