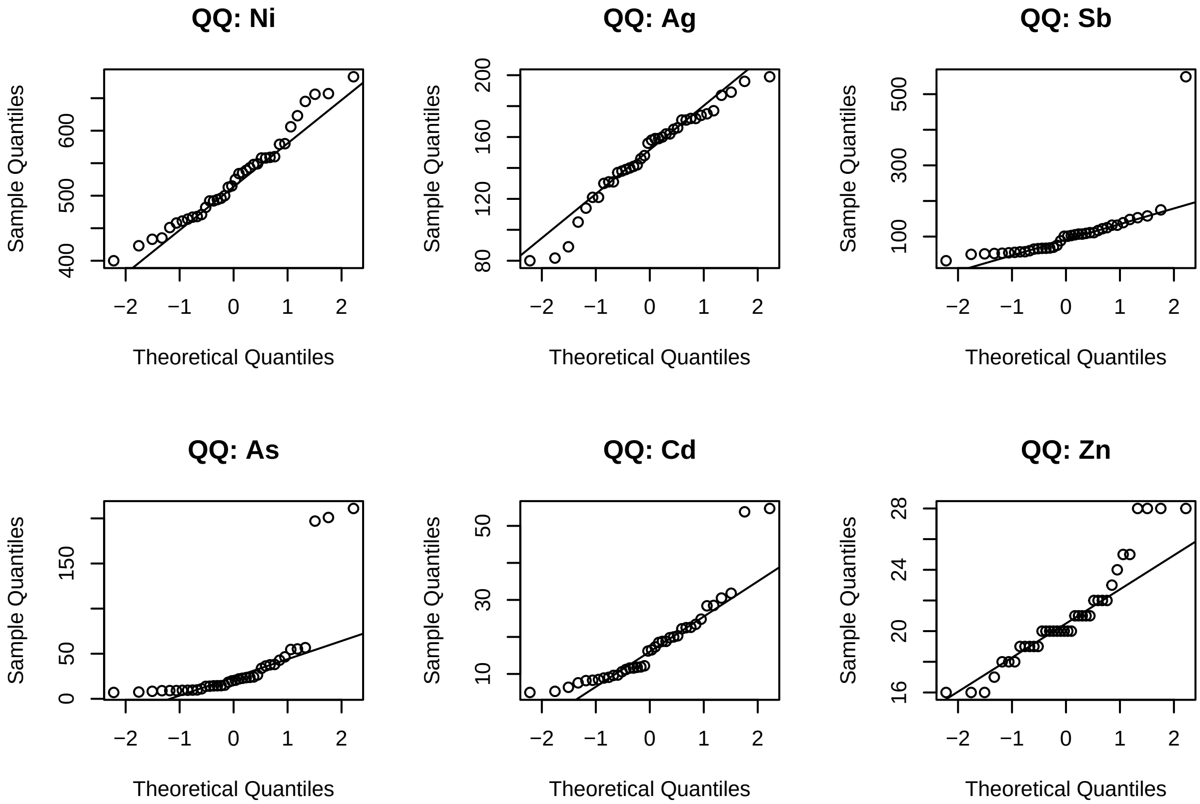
MAD scaling

2025-10-07

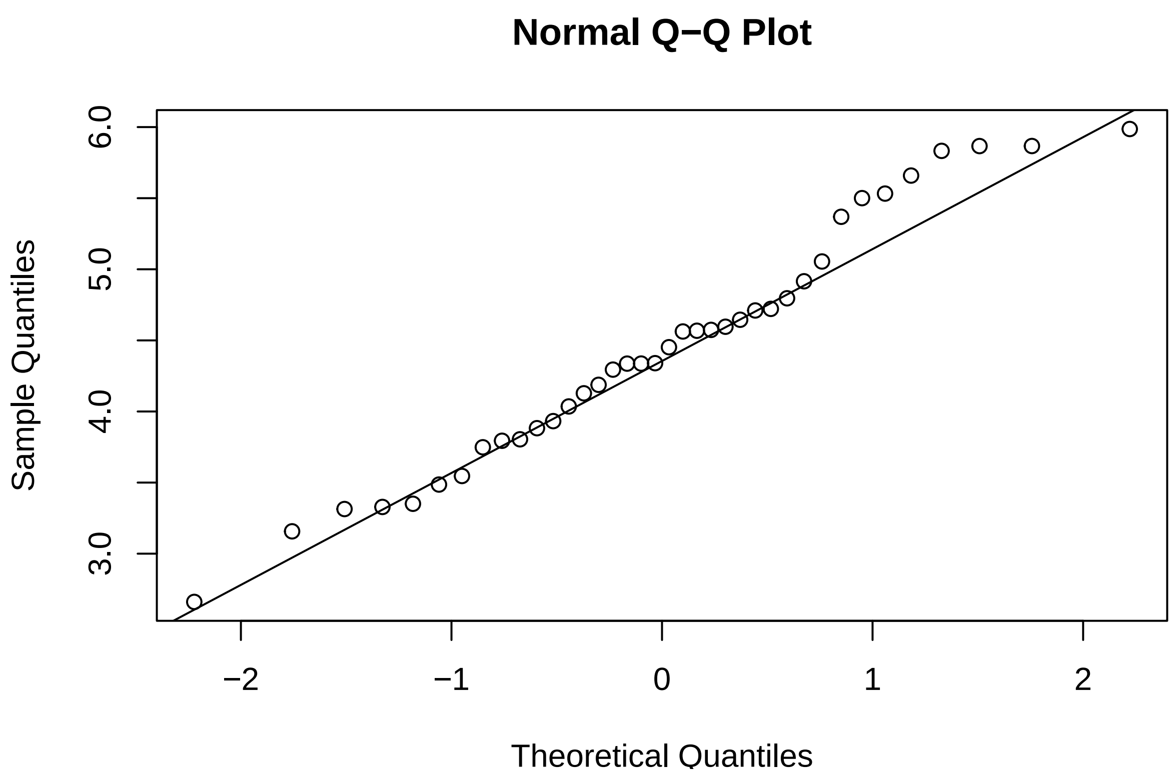
```r
df <- read.csv("orichalcum_ingots_dataset.csv", skip = 1, header = TRUE)

Ingots <- df[, 1:(ncol(df)-2)]
colnames(Ingots) <- c(
  "Sample","Cu","Zn","Pb","Fe","Ni","Ag","Sb","As","Co","Cd","Sn","Te",
  "Bi","Mn","Li","Al","V","Cr","Rb","Sr","Ba"
)

stopifnot(ncol(Ingots) == 22)  # 1 id + 21 variables

head(Ingots)
```

```
##   Sample Cu Zn   Pb   Fe  Ni  Ag    Sb    As   Co   Cd   Sn   Te   Bi   Mn   Li
## 1     S1 73 21 5.60 0.30 580 146  55.7  9.85 2.79 12.2 9.70 0.57 41.4 19.9 5.55
## 2     S2 87 16 4.29 0.53 534 199 132.0 36.40 8.44 20.0 4.80 1.00 72.8 20.8 2.17
## 3     S3 73 28 5.91 0.78 451 131 105.0 23.80 8.47 16.5 6.70 0.64 29.0 15.7 4.00
## 4     S4 80 20 6.17 0.39 558 171  70.0 11.10 4.83 11.9 9.34 0.82 60.7 23.2 6.16
## 5     S5 78 22 3.87 0.55 513 156  67.0 14.50 6.49  5.3 7.69 0.58 58.7 23.3 3.87
## 6     S6 81 22 6.51 0.19 579 159  60.3  8.32 1.88  9.6 8.54 0.96 61.4 15.7 2.12
##      Al    V   Cr   Rb   Sr   Ba
## 1 11.50 3.71 3.17 0.08 1.00 0.19
## 2  4.46 1.55 1.31 0.07 0.15 0.18
## 3  8.89 3.33 2.70 0.14 8.90 0.35
## 4 13.80 6.08 4.91 0.14 0.29 0.39
## 5 11.60 4.38 3.69 0.12 0.91 0.45
## 6  4.36 2.29 1.90 0.11 0.45 0.75
```

```r
#install.packages("tidyverse")
library(tibble)
library(ggplot2)

## Normality checks (raw)
X_raw <- as.matrix(Ingots[, -1])

# Univariate QQ for a few skewed variables
par(mfrow=c(2,3))
for (v in c("Ni","Ag","Sb","As","Cd","Zn")) { qqnorm(X_raw[,v], main=paste("QQ:", v)); qqline(X_raw[,v]
```



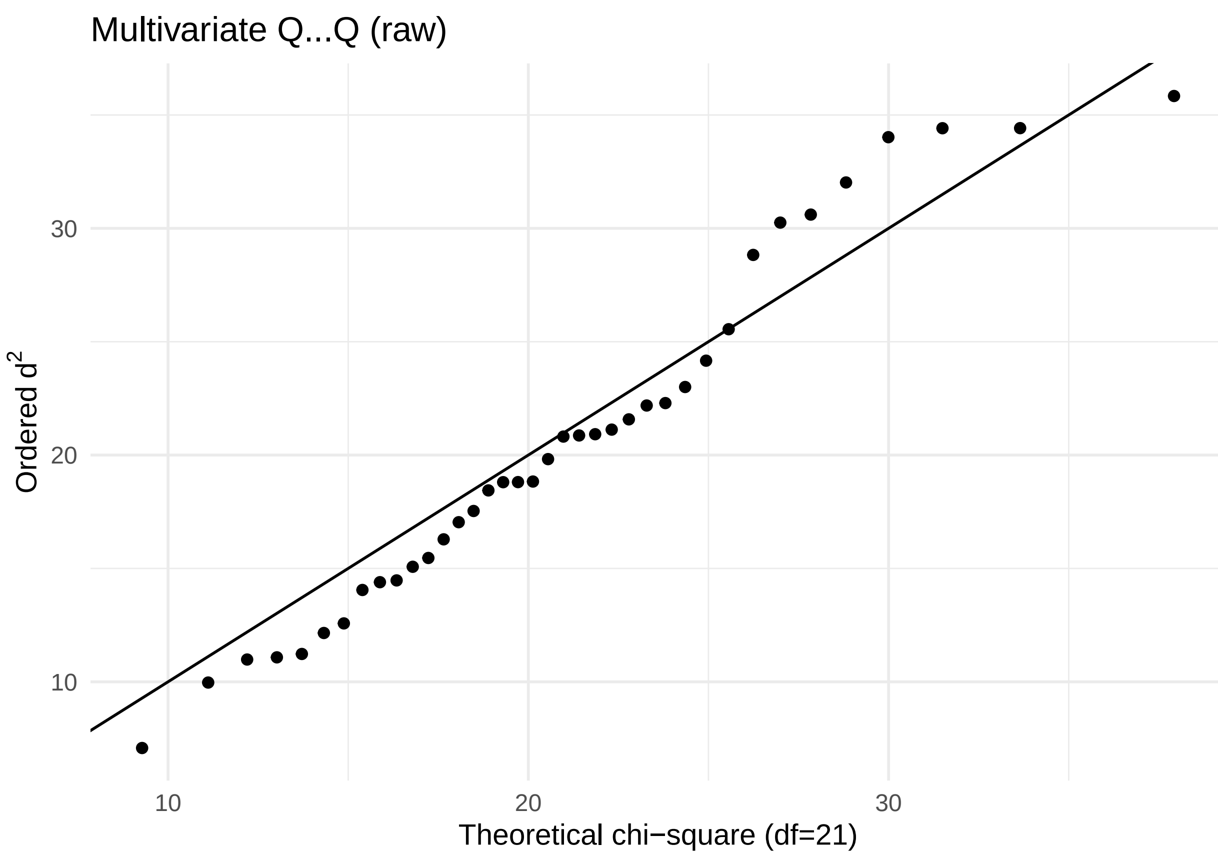

```r
par(mfrow=c(1,1))

# Mardia (raw)
psych::mardia(X_raw)
```



```
## Call: psych::mardia(x = X_raw)
##
## Mardia tests of multivariate skew and kurtosis
## Use describe(x) the to get univariate tests
## n.obs = 38   num.vars =  21
## b1p =  316.94   skew =  2007.28  with probability  <=  6.7e-05
##  small sample skew =  2181  with probability <=  6.4e-11
## b2p =  475.17   kurtosis =  -0.78  with probability <=  0.44
```

```r
n <- nrow(X_raw); p <- ncol(X_raw)
d2_raw <- mahalanobis(X_raw, center = colMeans(X_raw), cov = cov(X_raw))
th_raw <- qchisq(ppoints(n), df = p)

qq_raw_df <- tibble(theoretical = sort(th_raw), observed = sort(d2_raw))

ggplot(qq_raw_df, aes(theoretical, observed)) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1) +
  labs(title = "Multivariate Q-Q (raw)", x = paste0("Theoretical chi-square (df=", p, ")"),
       y = expression(Ordered~d^2)) + theme_minimal()
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q (raw)' in 'mbcsToSbcs': dot substituted
## for <e2>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q (raw)' in 'mbcsToSbcs': dot substituted
## for <80>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q (raw)' in 'mbcsToSbcs': dot substituted
## for <93>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q (raw)' in 'mbcsToSbcs': dot substituted
## for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q (raw)' in 'mbcsToSbcs': dot substituted
## for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q (raw)' in 'mbcsToSbcs': dot substituted
## for <93>
```




```r
## Natural log (ln) + MAD scaling
# ln + MAD scaling (robust), stored for later reuse
eps <- 1e-6
Ingots_ln <- Ingots
Ingots_ln[, -1] <- log(Ingots_ln[, -1] + eps)  # <natural log

vars    <- names(Ingots_ln)[-1]
medians <- apply(Ingots_ln[, -1], 2, median, na.rm = TRUE)
mads    <- mapply(function(x, m) mad(x, center = m, constant = 1, na.rm = TRUE),
                  as.data.frame(Ingots_ln[, -1]), medians)
mads[mads == 0] <- 1e-9   # safety

Ingots_ln_mad <- Ingots
Ingots_ln_mad[, -1] <- sweep(Ingots_ln[, -1], 2, medians, FUN = "-")
Ingots_ln_mad[, -1] <- sweep(Ingots_ln_mad[, -1], 2, mads,    FUN = "/")

# matrix for analyses
X_ln_mad <- as.matrix(Ingots_ln_mad[, -1]); rownames(X_ln_mad) <- Ingots$Sample
X_ln_mad
```

```
##             Cu         Zn           Pb          Fe         Ni          Ag
## S1  -1.1804289  0.6656072 -0.458855752 -1.04339184  1.2502137 -0.65087158
## S2   2.7018635 -3.0441782 -2.693242914  0.46281191  0.3045641  2.12490393
## S3  -1.1804289  4.5902355 -0.007087767  1.48552983 -1.6286660 -1.62252681
## S4   0.8457512  0.0000000  0.353904194 -0.34900125  0.8076812  0.76576747
## S5   0.2855239  1.3002445 -3.557150113  0.56084789 -0.1545721 -0.05708893
## S6   1.1206341  1.3002445  0.803671438 -2.25227722  1.2304655  0.11363700
```

```
## S7  -0.5823440  0.0000000   0.726033607 -0.79113769  0.8281720 -1.69120791
## S8   1.3921442  2.4872747   0.299368549 -0.95660816  0.6007303  0.16983054
## S9   0.8457512 -1.4373536  -2.053731154 -0.02973909 -0.8679010  0.11363700
## S10 -1.4856447  4.5902355   0.007087767  1.41678128 -2.0420409 -3.60543845
## S11 -0.2892563 -0.6997555   0.021239375 -0.09058471  0.1100426 -1.15595395
## S12 -2.4273952 -2.2171225   1.007258397  0.30859342  2.6767851 -0.96319727
## S13  0.2855239  0.0000000  -4.893121350  2.29734389 -1.4524062  1.66279952
## S14 -0.8793659  3.0441782  -1.927911477 -0.02973909 -2.0947786 -5.85427195
## S15  1.9253699  0.0000000  -1.080238178 -3.06051786  0.6215947  0.76576747
## S16  1.3921442 -0.6997555  -1.892307069 -2.70710618  0.8486261  0.28117160
## S17  0.0000000  0.0000000   1.315785696  1.51924583 -0.4483156  1.07486156
## S18  1.6603632 -0.6997555  -2.163099992 -0.02973909 -0.1100426  0.05708893
## S19 -1.7951294  4.5902355  -1.000000000  1.27368317 -0.5864751 -2.86835151
## S20 -0.2892563 -0.6997555   0.542033179 -0.21664032  0.3259749 -1.09124002
## S21 -0.8793659  0.6656072  -1.612787004  0.25512355 -2.3621757 -6.04273688
## S22 -1.7951294  0.6656072   0.119637454  1.95105449  1.7520579 -0.89985538
## S23 -1.4856447  1.3002445   0.367482862 -0.79113769  2.0686757 -2.33423707
## S24 -0.2892563 -1.4373536   0.777871954 -1.22599283  2.4658285  0.81802908
## S25  0.2855239 -1.4373536 -14.494852401  1.38172553 -0.5402364  0.28117160
## S26 -0.5823440  0.0000000   0.595018037  0.02973909  0.4958344 -1.22113853
## S27 -1.4856447  0.0000000   0.764942381 -0.21664032  0.8076812 -1.02699000
## S28 -0.2892563  0.6656072  -0.007087767  1.64997218 -1.1320990  0.81802908
## S29 -2.4273952  4.5902355   0.880596986  1.58541604 -0.6329015 -2.33423707
## S30  2.4460480 -3.0441782  -2.090029156  0.65538184 -1.2297037  1.98875691
## S31  0.0000000  3.0441782  -1.000000000  1.31018955 -3.0019867 -5.08721086
## S32  1.3921442  1.3002445   1.671630864  1.27368317 -1.3776894 -0.52892630
## S33  1.3921442  0.0000000  -1.177548658  0.02973909 -0.6329015  0.49978830
## S34  0.8457512 -3.0441782  -2.329875497  0.14482975 -1.2052244  1.56744926
## S35  0.2855239 -0.6997555   0.944166858 -1.63397782  0.4112197  0.92164688
## S36  0.2855239  0.0000000   2.222205667 -2.39537475  3.1209392  0.97301000
## S37 -0.5823440  1.9066662   0.956823202 -1.74661876 -1.3034574 -1.62252681
## S38  3.4520301  0.6656072  -7.811494624 -0.41774971  2.6593531  0.44563200
##              Sb          As          Co          Cd         Sn          Te
## S1  -1.53050501 -1.14930195 -1.14587823 -0.73654260  0.7458198 -3.53390592
## S2   0.68838841  0.93560956  1.01752346  0.50707986 -1.1237987  1.04089733
## S3   0.09988326  0.25789118  1.02445805  0.02308271 -0.2375254 -2.59120931
## S4  -0.94283945 -0.95873314 -0.07329566 -0.79918347  0.6453120 -0.57419546
## S5  -1.05548537 -0.53252484  0.50406465 -2.83415926  0.1287213 -3.39236333
## S6  -1.32643791 -1.41856440 -1.91741504 -1.33954634  0.4073403  0.70866753
## S7   0.14840679  3.73860893 -0.77674905 -0.86342386  2.3291213 -0.77515624
## S8  -0.75173338 -1.18037041 -0.88402947 -1.92730847 -0.1287213 -0.47554579
## S9   0.19603168  0.28447629  0.10080131  1.56879469 -1.6620186  0.27334990
## S10  0.24279061  0.23765746  0.98247654 -0.02308271 -0.2574323 -2.98193210
## S11 -1.64860062 -0.46785099  0.04447473 -1.64572915  0.3283815 -0.18657466
## S12 -1.80299701 -0.61144898 -0.23344359 -1.70563543  0.2798550 -0.09248692
## S13  4.35378163  3.62910036 -2.23259692  0.14220296  4.3955083  1.04089733
## S14 -0.34849623 -0.52156212  0.02533259  0.54453879 -3.4771165 -1.74645951
## S15  0.14840679  0.02757185 -3.03728961  0.90209269 -1.4534758  0.62344686
## S16  0.05042653 -0.02757185 -2.89508904  0.76964394 -1.6083287  0.09248692
## S17  1.06805957  1.58527345  1.15606445  1.67380925  5.8336990 -0.67405559
## S18  0.24279061  0.13968378  0.02917604  1.38931334 -1.7446436  0.95910267
## S19  0.00000000  1.00841733  0.39571719  0.31093289  0.9309983 -1.63263253
## S20 -1.73211971 -0.58849794 -0.25109064 -1.73613211  0.2700425  0.79300507
## S21 -1.01738585 -0.54356343  0.24007755  0.80341552 -3.4637283  0.27334990
```

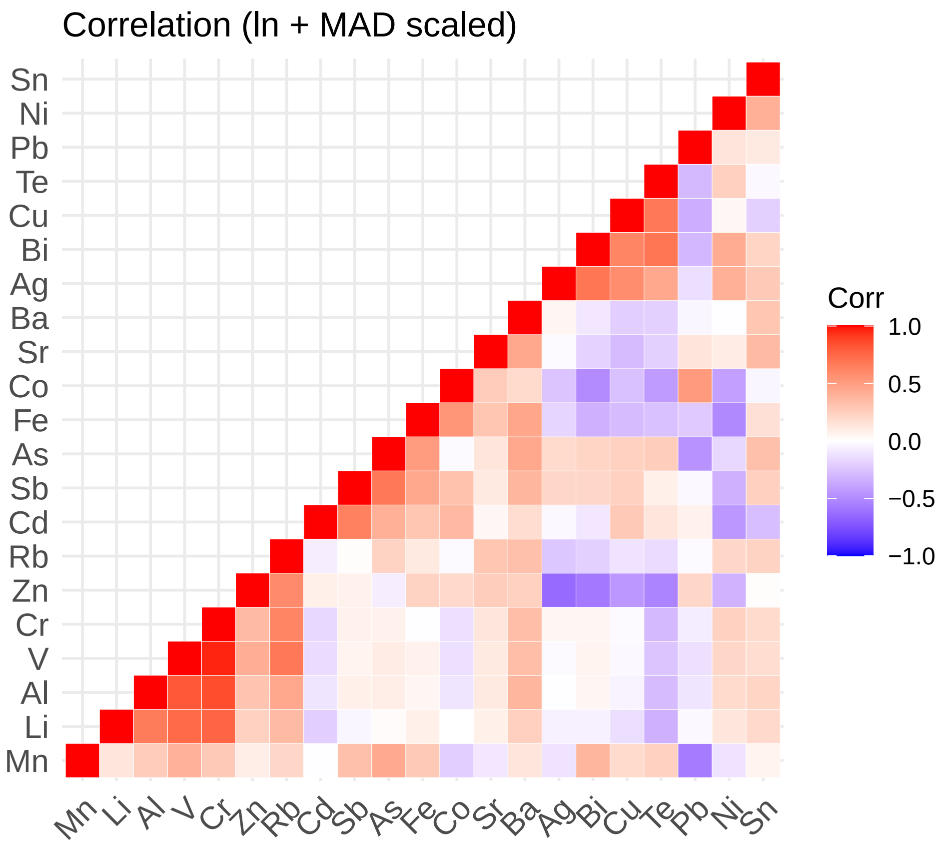
```
## S22  1.41355852  1.60267438  2.14533950 -0.82041520  4.6405876 -3.67795382
## S23 -1.04782017 -1.27212496 -1.27621964 -0.86342386  0.5673552 -3.39236333
## S24 -1.68266298 -1.21880769 -1.85601533 -1.47412094  0.7594834  1.04089733
## S25 -2.93981571  1.64255321 -5.25958921 -2.98076062 -1.5427067  0.27334990
## S26 -1.46666593 -1.20701103 -0.02533259 -0.95171169  0.3283815 -0.28176286
## S27 -1.58651457 -1.27562676 -0.28236166 -1.09023907  0.2961292  0.09248692
## S28  0.82127180  1.32621592  1.11684446  3.03845148 -3.6007461 -2.34077430
## S29  0.48578994  0.42326551  0.86089523  0.35140473  0.7265720 -1.74645951
## S30  1.15075707  0.99158267  1.42564753  0.48179376 -1.0690017  0.95910267
## S31  0.00000000 -0.15257599  1.34508197  0.81457273 -2.4721356 -2.09781608
## S32  0.98261415  1.19396196  1.15390609  2.99671134 -2.5185164  0.62344686
## S33  0.37817296  0.18942273  0.09338424  1.39815675 -1.8957228  1.04089733
## S34  0.68838841  0.81740216  0.82289538  0.35140473 -1.2835283  1.20206101
## S35 -1.45318990 -1.28971135 -1.88647408 -1.53003313  0.7676480  0.95910267
## S36 -1.14134578 -1.68502731 -1.47300511 -2.35967426 -0.2895968  1.04089733
## S37 -1.09415784 -1.58194227 -0.78838240 -1.31347414 -0.7046600 -2.46502856
## S38  0.54826274  3.66116315 -3.35708621  1.04828831  2.0255266  5.59533148
##              Bi          Mn          Li          Al            V          Cr
## S1  -0.91307352  0.04176508  3.39639114  1.61923030  2.052158908  2.32318321
## S2   0.73036983  0.22372827  1.91502942 -0.35104085 -0.906023997 -0.25546969
## S3  -1.94958115 -0.93342915  2.87976203  1.08376987  1.685902640  1.85490243
## S4   0.20111072  0.67294350  3.56088856  1.99847818  3.726426629  3.59993376
## S5   0.10355852  0.69063694  2.82764245  1.63723998  2.614855131  2.76641311
## S6   0.23449621 -0.93342915  1.87825675 -0.39821076  0.416844880  0.82952201
## S7  -0.30586199  0.51018173  1.57569642  0.08494758  0.218756647  0.27283036
## S8   0.09362111  0.76065939  1.77843179  1.92171284  1.902740017  1.46216781
## S9   1.38967121  0.76065939  1.52066262  0.96583764  0.975077552  0.63914993
## S10 -2.09365028 -0.04176508  1.01947599  0.51709546 -0.042094451 -0.04081389
## S11 -0.11260032 -1.74150099  0.56138388  0.45471884 -0.573055510 -0.32307124
## S12 -0.08600963 -1.58506697  0.98052401 -1.02867094 -0.694118062 -0.76529426
## S13  3.38225521  6.09131877  1.38425262  1.17307810  1.421385308  1.29683361
## S14 -2.25635520  1.68901063  0.88571155  0.43563518  0.700641674  0.23541954
## S15  1.41190986 -2.03900788  0.21793059 -0.89119663 -0.841049657 -0.81883710
## S16  1.08041567 -1.90411976  0.06117306  2.89153480  0.460959060  1.66513741
## S17 -2.06217256 -5.02988200 -0.75250856 -0.97132906 -2.637411396 -2.21200380
## S18  1.32517997  0.35989217 -0.33863598 -0.53627781 -0.612931160 -1.01437451
## S19 -1.52089668 -0.44017807  1.32197576  0.95590872  0.860171909  0.21653273
## S20 -0.01798931 -1.58506697 -0.06117306 -1.23923627 -1.399469527 -1.73124783
## S21 -2.34699787  1.18887243 -0.67554341 -1.56346282 -2.494679167 -2.58304267
## S22 -0.58737203  1.09393624  0.43654674  1.63723998  1.122465638  1.24401762
## S23 -1.00596873 -0.30304140  0.19646802  0.03906107  0.741233677  0.04081389
## S24  0.99403127 -1.23224636 -0.75250856 -0.70617661 -1.275302139 -1.48351111
## S25 -0.37519052  1.66121513 -0.08682361  0.01181890 -0.008368885 -0.34595766
## S26  0.09859405  0.70825461 -0.43237485  2.11569817 -0.612931160 -0.76529426
## S27  0.01798931 -0.28062325 -0.91871187 -1.14149529 -1.039833120 -1.51764017
## S28 -2.15764539 -1.87107766 -0.71355664  0.37402198  0.008368885  0.32806235
## S29 -1.54701040  0.47312095 -0.30858796 -0.01181890  1.024922448  0.68791993
## S30  0.78583171  0.54691161 -1.92684165 -1.85920049 -1.977197830 -2.21200380
## S31 -2.23412868  2.07226629 -1.15459398 -0.59837508 -0.399202470 -0.90103979
## S32 -1.07949942 -1.40487528 -0.79244671 -0.22438670  0.727756953  0.23541954
## S33  1.26256120  0.41689612 -2.40857109 -1.99858971 -2.391441044 -2.58304267
## S34  0.63276313 -0.21409265 -2.40857109 -1.84002889 -2.786419540 -2.68367334
## S35  1.01586883 -1.28898821 -2.10229651 -1.44967878 -1.476213724 -2.00198505
## S36  0.26281080 -1.34652368 -1.20631899 -0.01578478  0.265507617  0.93509439
```

```
## S37 -0.96988821 -1.06657085 -2.93934367 -1.41050429 -1.689789275 -0.98562549
## S38  2.85461224  5.71866952 -0.75250856  1.10240475  2.264618562  1.43733359
##             Rb          Sr          Ba
## S1  -0.4686684  0.95914260 -1.02786767
## S2  -1.0000000 -1.13838463 -1.17823738
## S3   1.7580939  3.37612938  0.67117200
## S4   1.7580939 -0.40949924  0.97213233
## S5   1.1447131  0.85486897  1.37012017
## S6   0.7984866  0.07628117  2.79081423
## S7   1.4632109 -0.23363136  1.66314581
## S8   0.7984866  1.36995813 -0.15553439
## S9  -1.0000000 -1.48130193  1.24368154
## S10  1.1447131 -0.76636926  0.24245298
## S11  0.4192387  0.31807241 -0.37814644
## S12  0.4192387  2.22071464 -0.62013944
## S13 -1.0000000  0.23613576  4.02756293
## S14  0.4192387 -1.00000000 -1.02786767
## S15 -0.4686684 -1.83339364 -1.33720414
## S16 -0.4686684 -0.33576277  0.33364699
## S17 -0.4686684  3.15246379  0.67117200
## S18  0.0000000  3.68797344 -0.05057225
## S19  2.2894290  2.71400649  6.96979367
## S20 -1.6133760  0.00000000  1.04254546
## S21 -1.0000000  0.56478866 -0.49651176
## S22  1.1447131  3.96061918  7.16112478
## S23  0.7984866 -1.83339364 -1.87718290
## S24 -0.4686684 -1.13838463  0.05057225
## S25  0.0000000 -1.00000000  0.42194546
## S26 -1.0000000 -0.87702507 -0.26461371
## S27 -1.0000000  3.06712917 -0.15553439
## S28  0.0000000  2.49188555  7.82908618
## S29  1.4632109  0.87890556  0.89989009
## S30 -1.0000000 -0.44829750 -0.88521267
## S31  0.4192387 -1.21466544 -0.37814644
## S32  1.1447131 -0.93680368  0.33364699
## S33 -1.6133760 -0.26663792 -1.50581093
## S34 -1.6133760 -1.70316943 -1.02786767
## S35 -0.4686684 -0.61873214 -1.33720414
## S36  0.7984866  0.00000000 -0.49651176
## S37  0.0000000  2.12867812 -2.30589993
## S38  3.5565909  1.83089272  1.17823922
```

```
#install.packages("ggcorrplot")
#install.packages("tidyverse")
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(ggcorrplot)

corr_mat <- cor(X_ln_mad)
ggcorrplot(corr_mat, hc.order = TRUE, type = "lower",
           lab = FALSE, show.diag = TRUE, outline.col = "white") +
  labs(title = "Correlation (ln + MAD scaled)")
```




```r
mad_long <- Ingots_ln_mad |>
  tidyr::pivot_longer(-Sample, names_to = "Element", values_to = "value")

p_hist <- ggplot(mad_long, aes(value)) +
  geom_histogram(bins = nclass.FD(mad_long$value), fill = "steelblue", color = "white") +
  facet_wrap(~Element, scales = "free_x", ncol = 3) +
  labs(title = "Histograms – ln + MAD scaled data",
       x = "Robust z-score (median/MAD)", y = "Count") +
  theme_minimal(base_size = 12) +
  theme(strip.text = element_text(face = "bold"))

p_hist
```

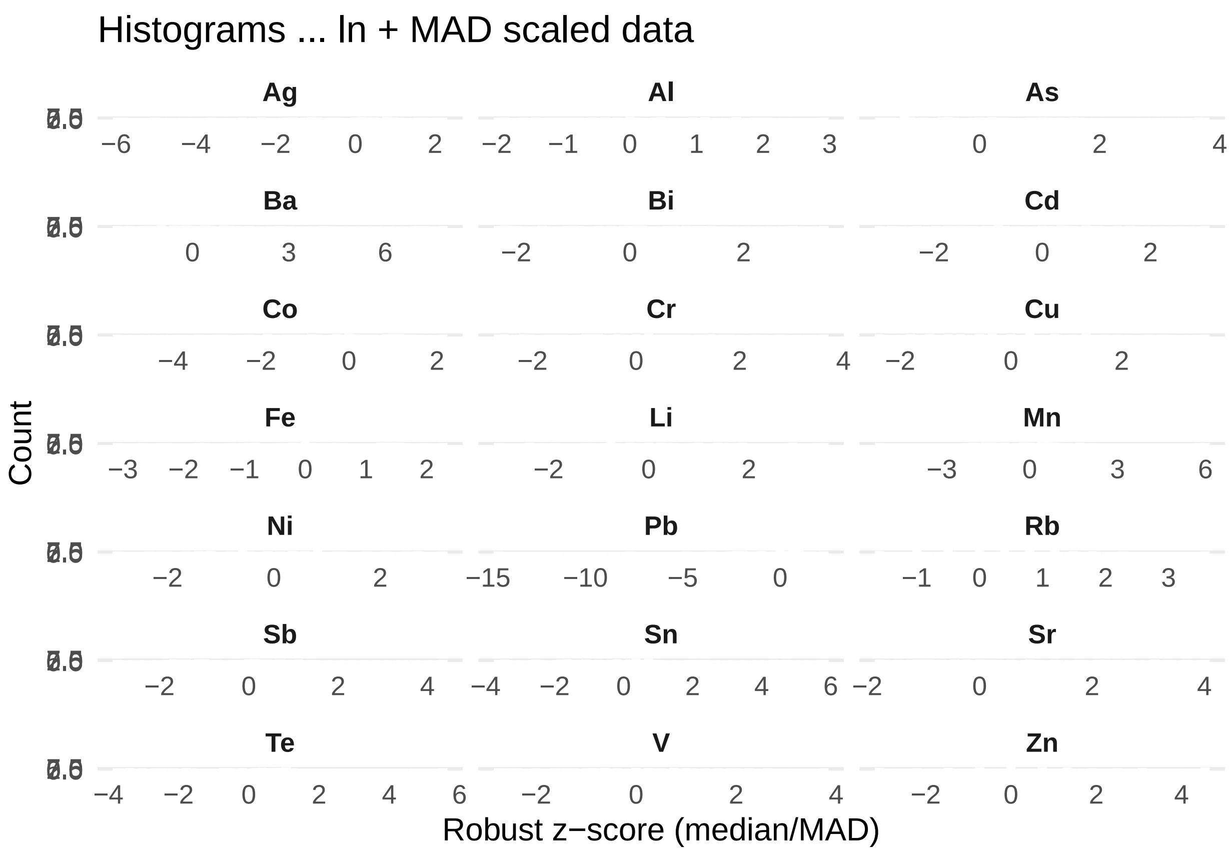
```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Histograms – ln + MAD scaled data' in 'mbcsToSbcs': dot
## substituted for <e2>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Histograms – ln + MAD scaled data' in 'mbcsToSbcs': dot
## substituted for <80>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Histograms – ln + MAD scaled data' in 'mbcsToSbcs': dot
## substituted for <94>
```

```
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Histograms – ln + MAD scaled data' in 'mbcsToSbcs': dot
## substituted for <e2>
```

```
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Histograms – ln + MAD scaled data' in 'mbcsToSbcs': dot
## substituted for <80>
```

```
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Histograms – ln + MAD scaled data' in 'mbcsToSbcs': dot
## substituted for <94>
```




```r
ggsave("histograms_ln_mad.png", plot = p_hist, width = 12, height = 8, dpi = 300)
```

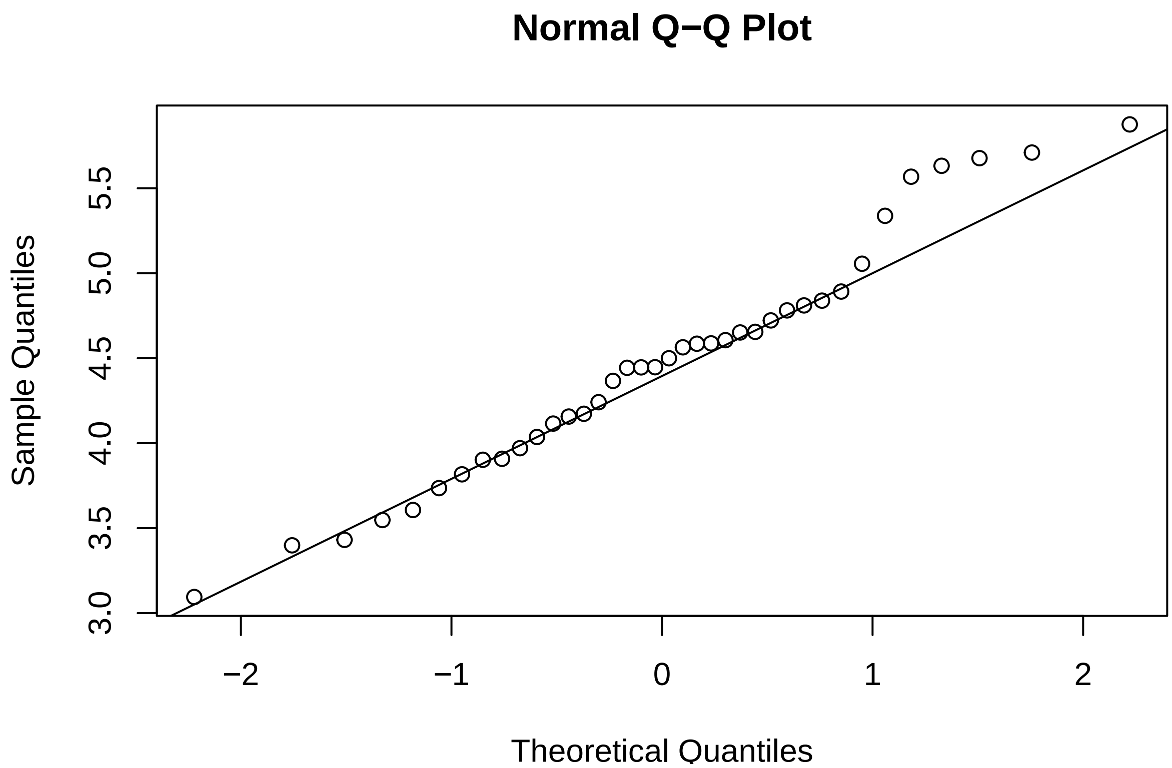
```r
library(psych)   # for mardia()
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```r
library(ggplot2)

X_log_mad <- as.matrix(Ingots_ln_mad[, -1])
```

```r
rownames(X_log_mad) <- Ingots_ln_mad$Sample

# 1) Mardia skewness & kurtosis
mardia_result <- psych::mardia(X_ln_mad)
```



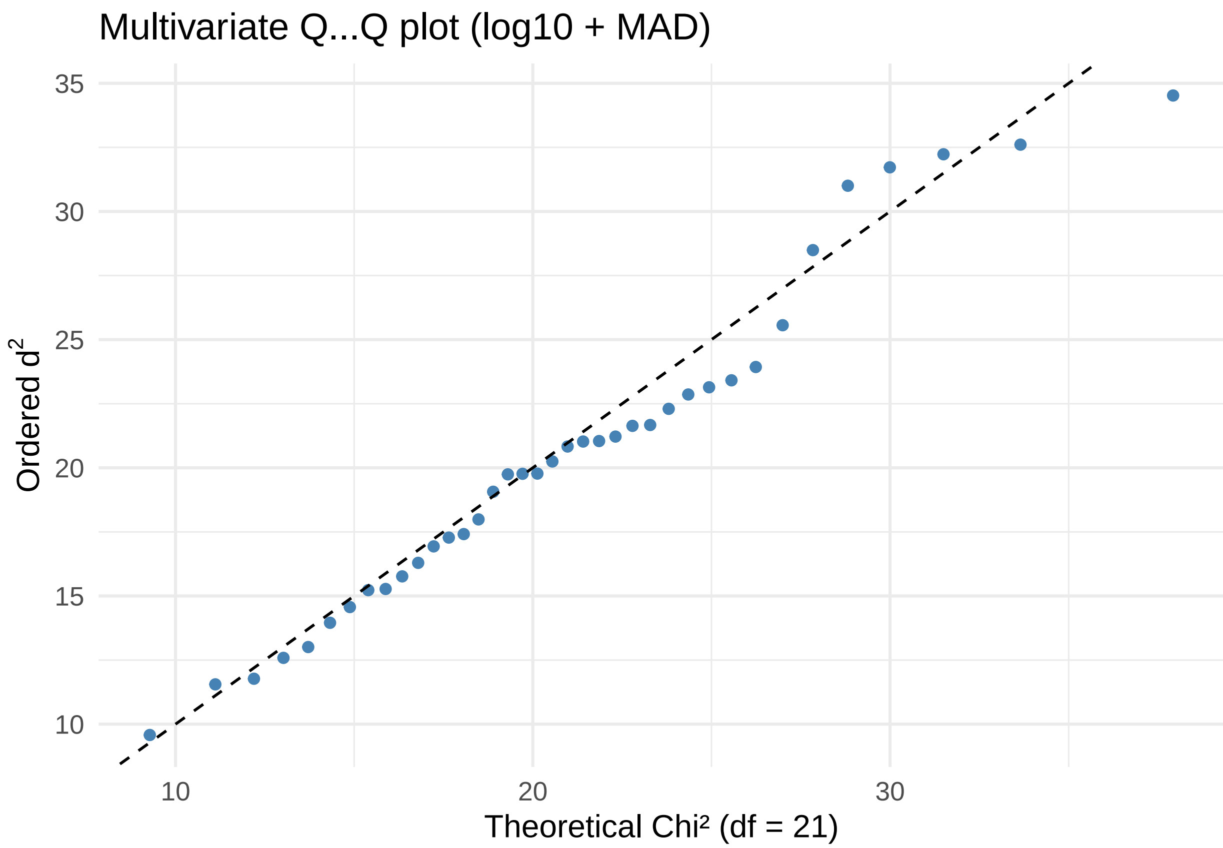

```r
mardia_result
```

```
## Call: psych::mardia(x = X_ln_mad)
##
## Mardia tests of multivariate skew and kurtosis
## Use describe(x) the to get univariate tests
## n.obs = 38   num.vars =  21
## b1p =  281.28   skew =  1781.42  with probability  <=  0.43
##  small sample skew =  1935.59  with probability <=  0.0035
## b2p =  456.43   kurtosis =  -2.63  with probability <=  0.0084
```

```r
# -> Check $skew and $kurtosis for p-values; significant = not MVN.

# 2) Multivariate Q-Q plot (Mahalanobis vs chi-square)
n  <- nrow(X_ln_mad)
p  <- ncol(X_ln_mad)
d2 <- mahalanobis(X_ln_mad, center = colMeans(X_ln_mad), cov = cov(X_ln_mad))
th <- qchisq(ppoints(n), df = p)

qq_df <- tibble(theoretical = sort(th), observed = sort(d2))

p_mvn <- ggplot(qq_df, aes(theoretical, observed)) +
  geom_point(color = "steelblue") +
  geom_abline(intercept = 0, slope = 1, linetype = "dashed") +
  labs(
```

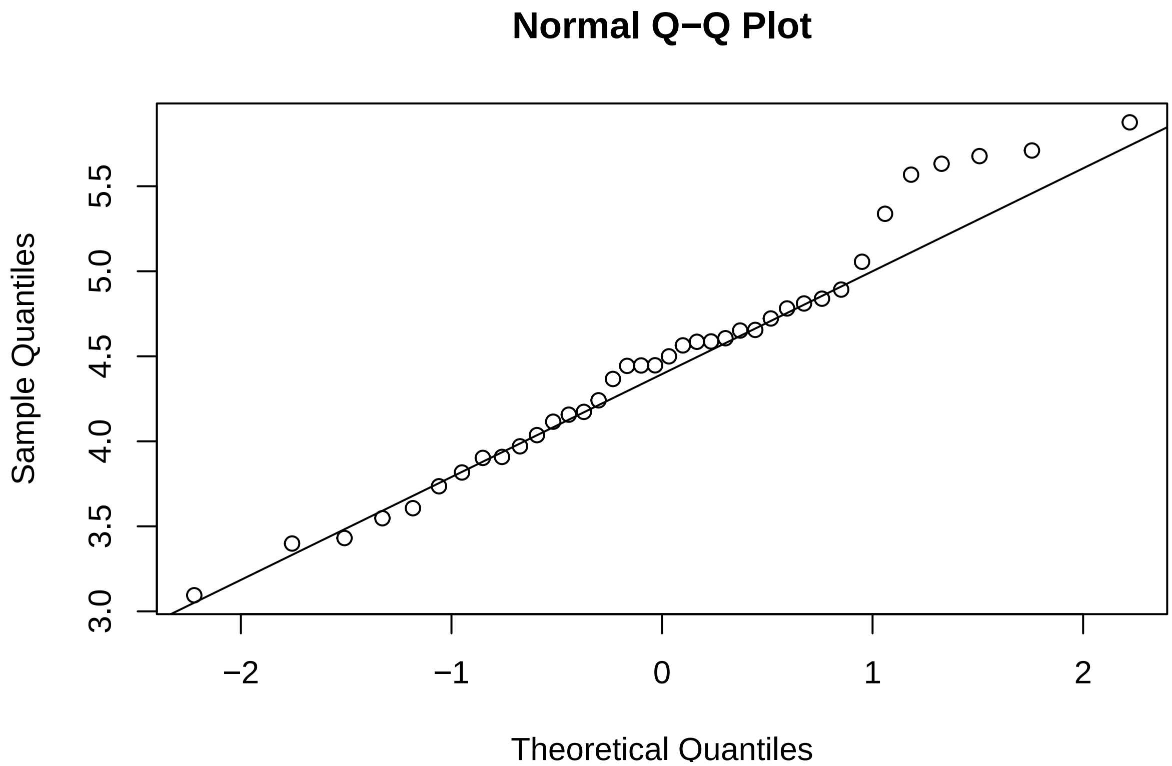
```r
    title = "Multivariate Q-Q plot (log10 + MAD)",
    x = paste0("Theoretical Chi² (df = ", p, ")"),
    y = expression(Ordered~d^2)
  ) +
  theme_minimal(base_size = 12)

p_mvn
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q plot (log10 + MAD)' in 'mbcsToSbcs':
## dot substituted for <e2>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q plot (log10 + MAD)' in 'mbcsToSbcs':
## dot substituted for <80>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q plot (log10 + MAD)' in 'mbcsToSbcs':
## dot substituted for <93>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q plot (log10 + MAD)' in 'mbcsToSbcs':
## dot substituted for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q plot (log10 + MAD)' in 'mbcsToSbcs':
## dot substituted for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Multivariate Q-Q plot (log10 + MAD)' in 'mbcsToSbcs':
## dot substituted for <93>
```




```r
# Save as PNG for slides
ggsave("qqplot_log_mad.png", plot = p_mvn, width = 8, height = 6, dpi = 300)
```

```r
## Natural log (ln) + standardisation (mean/SD) and Mardia

Ingots_ln_std <- Ingots
Ingots_ln_std[, -1] <- log(Ingots_ln_std[, -1] + eps)      # <--- natural log
Ingots_ln_std[, -1] <- scale(Ingots_ln_std[, -1], center = TRUE, scale = TRUE)

X_ln_std <- as.matrix(Ingots_ln_std[, -1]); rownames(X_ln_std) <- Ingots$Sample

mardia_ln_std <- psych::mardia(X_ln_std)
```




```
mardia_ln_std
```

```
## Call: psych::mardia(x = X_ln_std)
##
## Mardia tests of multivariate skew and kurtosis
## Use describe(x) the to get univariate tests
## n.obs = 38   num.vars =  21
## b1p =  281.28   skew =  1781.42  with probability  <=  0.43
##  small sample skew =  1935.59  with probability <=  0.0035
## b2p =  456.43   kurtosis =  -2.63  with probability <=  0.0084
```