MUH Certification

TCM

Tuesday, July 15, 2014

# Analysing submitted data for OKUM, GAS or MUH-1

Thomas Meisel (2014-08-04)  
The data for this interlaboratory comparison based certification of property values were analysed by 36 labs following the nested design approached as proposed the IAG certification protocol. Participating labs received 3 packages of OKUM and MUH-1 respectively and one package of GAS. The latter was supplied as a "traceablility" sample and is here used for quality control purposes. It was the task of the labs to prepare two independent sample preparations (i.e. digstions) of each packet and analyse the preparations on two different days. Labs thus should have submitted 12 values (3x2x2 PacketxPrepxDay). important notice: .RProfile was originally taken from <http://gforge.se/2014/01/fast-track-publishing-using-knitr-part-ii/> but does not work with the RStudio verion 0.98.976 as PDF option does not show up

#### importing the data and assigning factors

source("Makefile.R")

refmat <- 'MUH' # defining the RM  
measurand.name <- 'Zr'

# measurand selected Zr for candiate reference material MUH

# Data for certification project was gathered and joined in Excel. The files  
# were exported from Excel as xxxx.csv files to make them universially  
# readable. For this markdown the data is stored in the 'root/documents'  
# directory. Data is loaded ('GOMGather1.R') and merged ('GOMMerge.R') for  
# GAS, OKUM and MUH-1 are merged together with a methods file  
# ('OKUM.method') into a universal data.frame file named 'GOM'. All of this  
# happens in the 'Makefile.R'

MorT <- grep(measurand.name, colnames(GOM), fixed = TRUE) # finding the position of the measurand.name in the Columnheaders of dataframe  
ifelse(MorT[1] < 21, MorT <- "M", MorT <- "T") # testing if measurand is a major or trace element/compound (col:5-20 majors)

## [1] "T"

ifelse(MorT == "T", unit <- "mg/kg", unit <- "g/100g") # testing which unit is needed

## [1] "mg/kg"

#### defining the plotting theme

#### defining the function for plotting methods vs. measurand mass fraction. Sample preparation methods are also marked in the plot.

#### defining the function of Youden plots

#### initial calculations with complete data set

## means over packets within lab  
meanGOM <- function(x) mean(x, na.rm = TRUE) # defining a function for further calcuations  
sdGOM <- function(x) sd(x, na.rm = TRUE) # defining a function for further calcuations, here for calculating standard deviations neede for Youden plot  
meanGOM.packet <- ddply(GOM, c("Lab", "Packet"), numcolwise(meanGOM)) # calculated the mean for each Packet within each Lab by calculating the mean of days and preparations lumped together  
## mean over mean of packets within lab  
GOM.mean <- ddply(meanGOM.packet, c("Lab"), numcolwise(meanGOM))

## median over packets within lab   
medianGOM <- function(x) median(x, na.rm=TRUE) # defining a function for further calcuations  
medianGOM.packet <- ddply(GOM, c("Lab", "Packet"), numcolwise(medianGOM))  
GOM.sd <- ddply(medianGOM.packet, c("Lab"), numcolwise(sdGOM))  
## median over median of packets within lab  
GOM.median <- ddply(medianGOM.packet, c("Lab"), numcolwise(medianGOM))# calculated the median for each Packet within each Lab by calculating the median of days and preparations lumped together  
GOM.median <- merge(GOM.median, OKUM.methods, by="Lab")

summary(GOM[[measurand]], na.rm=TRUE, digits=4) # with values without outlier removal

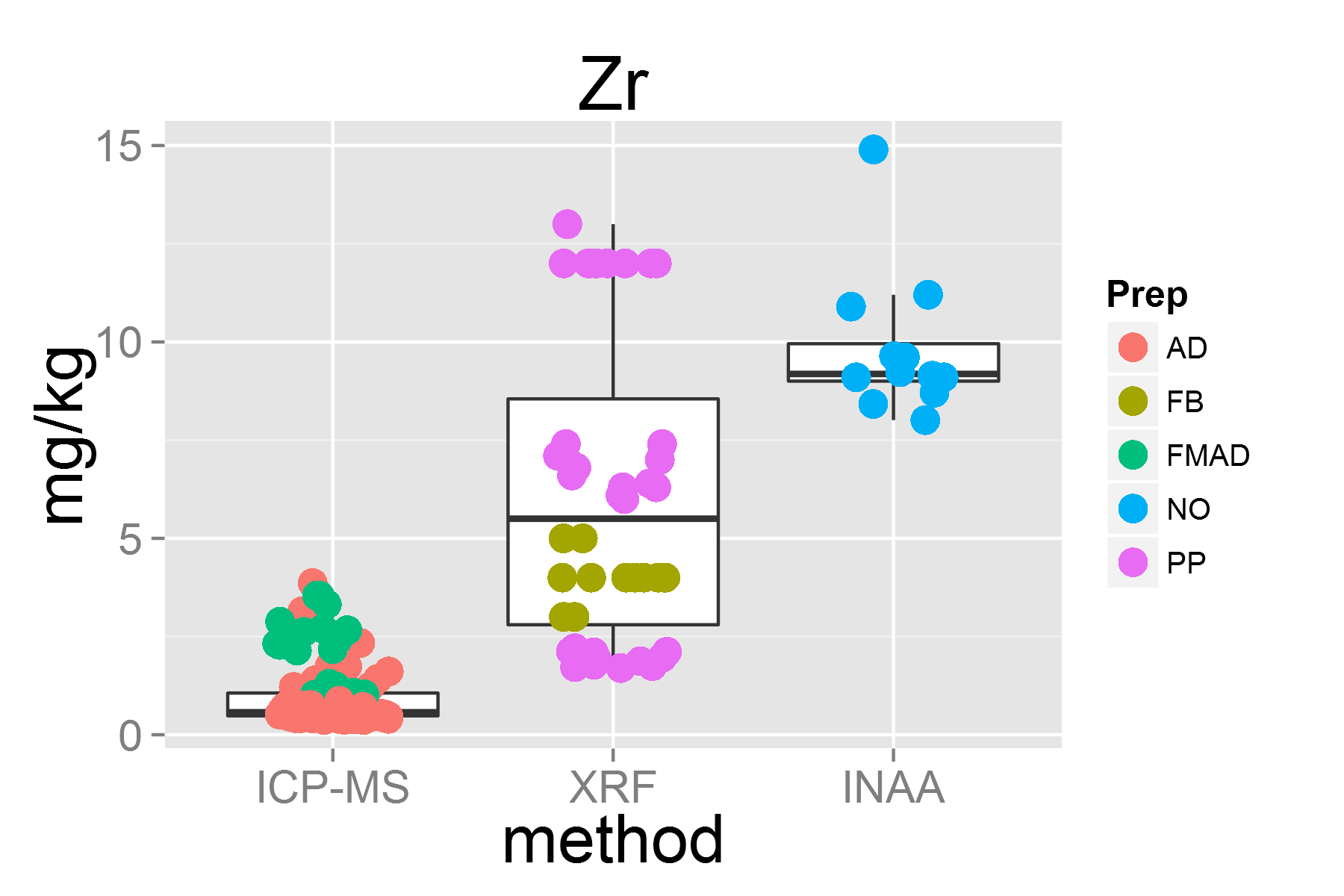
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.38 0.52 1.22 3.01 4.00 14.90 258

mean.before <- mean(GOM.mean[[measurand]], na.rm=TRUE)  
median.before <- median(GOM.median[[measurand]], na.rm=TRUE)

The Zr.1 mean of the Lab means is **2.9485** g/100 g  
The Zr.1 median of the Lab+Package medians is **1.1535** g/100 g

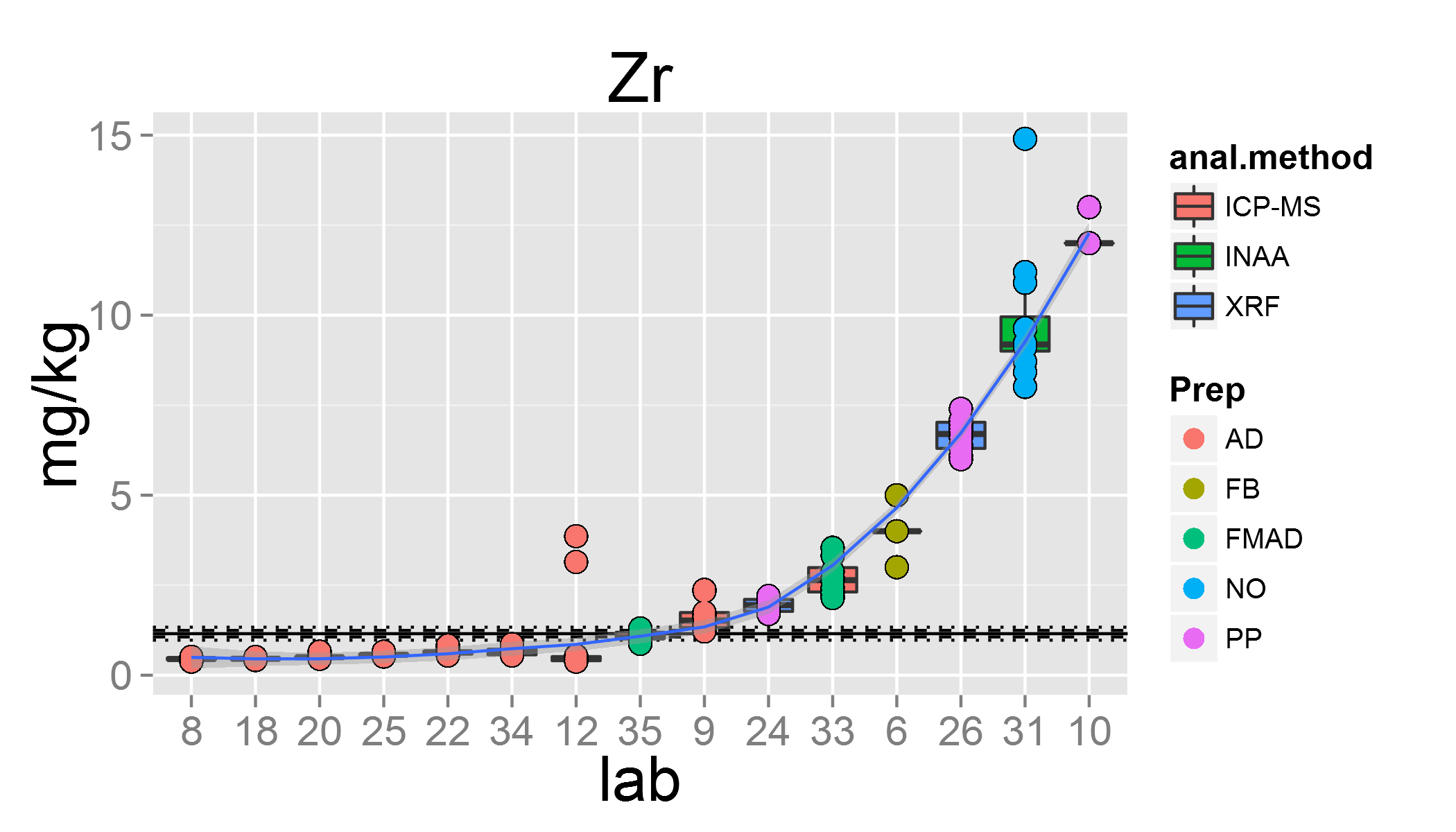
plot\_method(measurand)

## Warning: Removed 16 rows containing missing values (geom\_point).  
## Warning: Removed 1 rows containing missing values (geom\_point).



plot\_lab(measurand, MorT, horw=TRUE, u =FALSE)

## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.



plot\_youd(measurand.name, 'MUH', 'GAS')

## Warning: Removed 21 rows containing missing values (geom\_point).  
## Warning: Removed 21 rows containing missing values (geom\_text).

plot\_youd(measurand.name, 'MUH', 'OKUM')

## Warning: Removed 21 rows containing missing values (geom\_point).  
## Warning: Removed 21 rows containing missing values (geom\_text).

### Mandel k barplot displays the within lab performance relative to all participating labs using the median over packages

k <- with(medianGOM.packet, mandel.kh(medianGOM.packet[[measurand]], g=Lab, type="k"))   
barplot(k, las=2, col=1:4)

plot of chunk mandel k

### Outlier rejection

#### Removing outliers lab based on Youden plot and Mandel's k (lab performance)

outlier <- MUH.outlier[[measurand.name]]  
#outlier <- ifelse(MUH.outlier[[measurand.name]][1]=="NA", "0", MUH.outlier[[measurand.name]]) # does not work for FeO  
outlier <- na.omit(outlier)  
# outlier<- c(16) ## defining the outlying lab here with lab# directly. Enter 0 when no outlier  
leng <- length(outlier) ## counting the number of outliers for loop  
for(i in seq(leng)) ## looping  
{  
 GOM[[measurand]] <- ifelse(GOM$Lab==outlier[i], NA, GOM[[measurand]]) ## replacing values of outlying lab with "NA" and defining new GOM  
 message("Lab ", outlier[i], " was removed")  
 print(summary(GOM[[measurand]], na.rm=TRUE, digits=4))  
 }

## Lab 10 was removed

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.38 0.51 0.85 2.34 2.97 14.90 270

## Lab 6 was removed

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.38 0.51 0.70 2.21 2.27 14.90 282

## Lab 26 was removed

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.38 0.50 0.65 1.82 1.92 14.90 294

## Lab 33 was removed

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.38 0.49 0.61 1.73 1.56 14.90 306

## Lab 31 was removed

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.4 0.5 0.6 0.9 1.1 3.9 318

## Lab 24 was removed

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.4 0.5 0.6 0.8 0.7 3.9 330

# plot\_lab(measurand, MorT, horw =TRUE, u = FALSE) # replotting without outlier, M for majors and T for traces

### Calculating mean and median of property value

summary(GOM[[measurand]], na.rm=TRUE, digits=4)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.4 0.5 0.6 0.8 0.7 3.9 330

mean <- mean(tapply(GOM[[measurand]], GOM$Lab, mean, na.rm=TRUE), na.rm=TRUE) # mean of lab means after outlier removal  
median.before <- median(GOM.median[[measurand]], na.rm=TRUE) # median of measurand w/o outlier removal  
medianGOM.packet.after <- ddply(GOM, c("Lab", "Packet"), numcolwise(medianGOM)) # median Lab and Packets after outlier removal  
## median over median of packets within lab  
GOM.median.after <- ddply(medianGOM.packet.after, c("Lab"), numcolwise(medianGOM)) # creating a new table of median of labs after outlier removal for measurand but for entire table (all measurands)  
GOM.median.after <- merge(GOM.median.after, OKUM.methods, by="Lab")  
median.after <- median(GOM.median.after[[measurand]], na.rm=TRUE) # median of the individual measurand after outlier removal  
## calculating method parameters  
# '%p%' <- function(x, y) {as.character(paste (x, y, sep =""))}  
prep <- 'Prep.'  
method <- 'Method.'  
anal.prep <- prep %p% measurand.name  
anal.method <- method %p% measurand.name  
anal <- GOM.median.after[[measurand]]  
anal.prep <- GOM.median.after[[anal.prep]]  
anal.method <- GOM.median.after[[anal.method]]  
analyte <- data.frame(GOM.median.after$Lab, GOM.median.after$names, anal, anal.prep, anal.method )  
analyte <- na.omit(analyte)  
  
 bymethod.n <- ddply(analyte, c("anal.method"), summarise,   
 N=length(anal),   
 mean = round(mean(anal), 3),   
 median = round(median(anal), 3),   
 sd = round(sd(anal),3),   
 se = round(sd/sqrt(N),3))

## Measurement uncertainty estimations of property value

The the mean of means is   
The standard deviation is   
The combined standard uncertainty equals then

### Nested random effects in data analysis: two way ANOVA

This model can be used when the results of the interlaboratory study are used to confirm the homogeneity of the material as well as to characterise it. The experimental scheme is illustrated in Fig X for the particular case of the IAG protocol. When the ILC consists of different methos, the result can be expressed by the equation

where

is the th result of sample unit j reported from method/laboratory ,  
 is the error due to method/laboratory ,  
 is the error due to the th sample unit within method/laboratory , is the measurement error.

The parameters to be estimated are the grand mean, the between-laboratory standard deviation , the between-bottle standard deviation and the repeatability standard deviation . The are related as follows

The formulae for computing the above-mentioned estimates read as follows. The grand mean is computed using

where denotes the number of laboratories, the number of bottles used by method/laboratory , and is the number of replicates measured on bottle . The variances are computed as follows

where

and

### solutions of the above equations in

ANOVA is calculated based on a linear model using the using "linear mixed effects models" of package nlme (lme {nlme})

GOM.lme <- lme(measurand ~ 1, random = ~ 1|Lab/Packet, data=DF.lme) # linear model with random effects

the variance components are extrated with package ape (varcomp {ape})

= sL2 <- varcomp(GOM.lme, FALSE, FALSE)[[1]] # between-laboratory variance  
 = sbb2 <- varcomp(GOM.lme, FALSE, FALSE)[[2]] # between bottle variance  
 = sr2 <- varcomp(GOM.lme, FALSE, FALSE)[[3]] # repeatability standard deviation

The characterisation uncertainty is calculated

u1 <- sqrt(sL2/p+sbb2/p/r+sr2/p/r/4)

which is equivalent to

This approach is not completly correct as it assumes 4 replicates per bottle

u2 <- attr(GOM.lme$fixDF,"varFixFact") # this approach takes unbalanced data into account and is used for all further calculations.

The between day variance is neglected here but the uncertainty component due to inhomogeneity is taken into account through the compenent.

### before outlier removal

medianGOM.packet$Lab <- as.factor(medianGOM.packet$Lab) # using only the median of the 3 packages per lab  
medianGOM.packet$Packet <- as.factor(medianGOM.packet$Packet)  
anal <- medianGOM.packet[[measurand]]  
DF.lme <- data.frame(medianGOM.packet$Lab, medianGOM.packet$Packet, medianGOM.packet[[measurand]])  
DF.lme <- na.omit(DF.lme)  
names(DF.lme) <- c("Lab", "Packet", "measurand")  
GOM.lme <- lme(measurand ~ 1, random = ~ 1|Lab/Packet, data=DF.lme) # linear model with random effects  
sL2 <- varcomp(GOM.lme, FALSE, FALSE)[[1]] # between-laboratory variance  
sbb2 <- varcomp(GOM.lme, FALSE, FALSE)[[2]] # between bottle variance  
sr2 <- varcomp(GOM.lme, FALSE, FALSE)[[3]] # repeatability standard deviation  
n.p <- dim(DF.lme)[1] # number of obervations  
p <- length(unique(DF.lme$Lab)) # haven't found a better way how to extract the number of labs (number of groups)  
r <- length(unique(DF.lme$Packet))  
u1 <- sqrt(sL2/p+sbb2/p/r+sr2/p/r/4) # calculating the standard uncertainty of characterization  
u2 <- attr(GOM.lme$fixDF,"varFixFact") # gives the same results as u, amazing!  
# print(plot(DF.lme))

### after outlier removal

medianGOM.packet.after$Lab <- as.factor(medianGOM.packet$Lab) # using only the median of the 3 packages per lab  
medianGOM.packet.after$Packet <- as.factor(medianGOM.packet$Packet)  
anal <- medianGOM.packet.after[[measurand]]  
DF.lme <- data.frame(medianGOM.packet.after$Lab, medianGOM.packet.after$Packet, medianGOM.packet.after[[measurand]])  
DF.lme <- na.omit(DF.lme)  
names(DF.lme) <- c("Lab", "Packet", "measurand")  
GOM.lme <- lme(measurand ~ 1, random = ~ 1|Lab/Packet, data=DF.lme) # linear model with random effects  
sL2.a <- varcomp(GOM.lme, FALSE, FALSE)[[1]] # between-laboratory variance  
sbb2.a <- varcomp(GOM.lme, FALSE, FALSE)[[2]] # between bottle standard deviation  
sr2.a <- varcomp(GOM.lme, FALSE, FALSE)[[3]] # repeatability standard deviation  
n.p <- dim(DF.lme)[1] # number of obervations  
p <- length(unique(DF.lme$Lab)) # haven't found a better way how to extract the number of labs (number of groups)  
r <- length(unique(DF.lme$Packet))  
t.value <- qt(0.975,df=p-1)  
u1.a <- sqrt(sL2.a/p+sbb2.a/p/r+sr2.a/p/r/4) # calculating the standard uncertainty of characterization  
u2.a <- attr(GOM.lme$fixDF,"varFixFact") # gives the same results as u1, amazing!  
plot(DF.lme)

plot of chunk measurement uncertainty of data after outlier rejection

### tests for normal distribution

qqnorm(GOM.median.after[[measurand]])  
qqline(GOM.median.after[[measurand]])

plot of chunk QQ plot

analyte.noPP <- subset(analyte, analyte$anal.prep!="PP") # Removing all PP preparations for comparison reasons  
median.after.noPP <- median(analyte.noPP$anal)

## Comparisons of property value calculations for MUH and measurand Zr

### before and after removal of outlying lab(s) # 10, 6, 26, 33, 31, 24

|  |  |  |  |
| --- | --- | --- | --- |
| type | before | after | unit |
| **mean** | 2.9485 | 0.7731 | mg/kg |
| **median** | 1.1535 | **0.579** | mg/kg |
| **median w/o PP** | - | **0.579** | mg/kg |

### performance characteristics for MUH and measurand Zr

|  |  |  |
| --- | --- | --- |
| type | variance before | variance after outlier rejection |
| between-laboratory variance | 13.0652 | 0.1202 |
| between-bottle variance | 0.0497 | 0.0624 |
| repeatability variance | 0.0536 | 0.0136 |

### mesasurment uncertainty

|  |  |  |  |
| --- | --- | --- | --- |
| type | before | after outlier rejection | alternative calculation |
| standard uncertainty | 0.9345 | 0.1257 | 0.1272 |
| expanded standard uncertainty | - | - | **0.2932** *k* = 2.31 |

outlier <- ifelse(outlier=="0", "X", outlier)  
df <- data.frame(Sys.Date(), refmat, measurand.name, mean.before, mean, median.before, median.after, median.after.noPP, unit, sL2.a, sbb2.a, sr2.a, u1.a, u2.a, u2.a\*t.value, t.value, outlier, p)  
#names(df) <- c("date", "RM", "measurand", "mean.before", "mean", "median.before", "median.after", "median.after.noPP", "unit", "sL2.a", "sbb2.a", "sr2.a", "u", "u.alternative", "U", "t.value", "outlier", "labs remaining") # needed only the first time  
#write.table(df, "dfMUH.txt", row.names=FALSE) # needed only the first time  
write.table(df, "dfMUH.txt", row.names=FALSE, append=TRUE, col.names=FALSE)

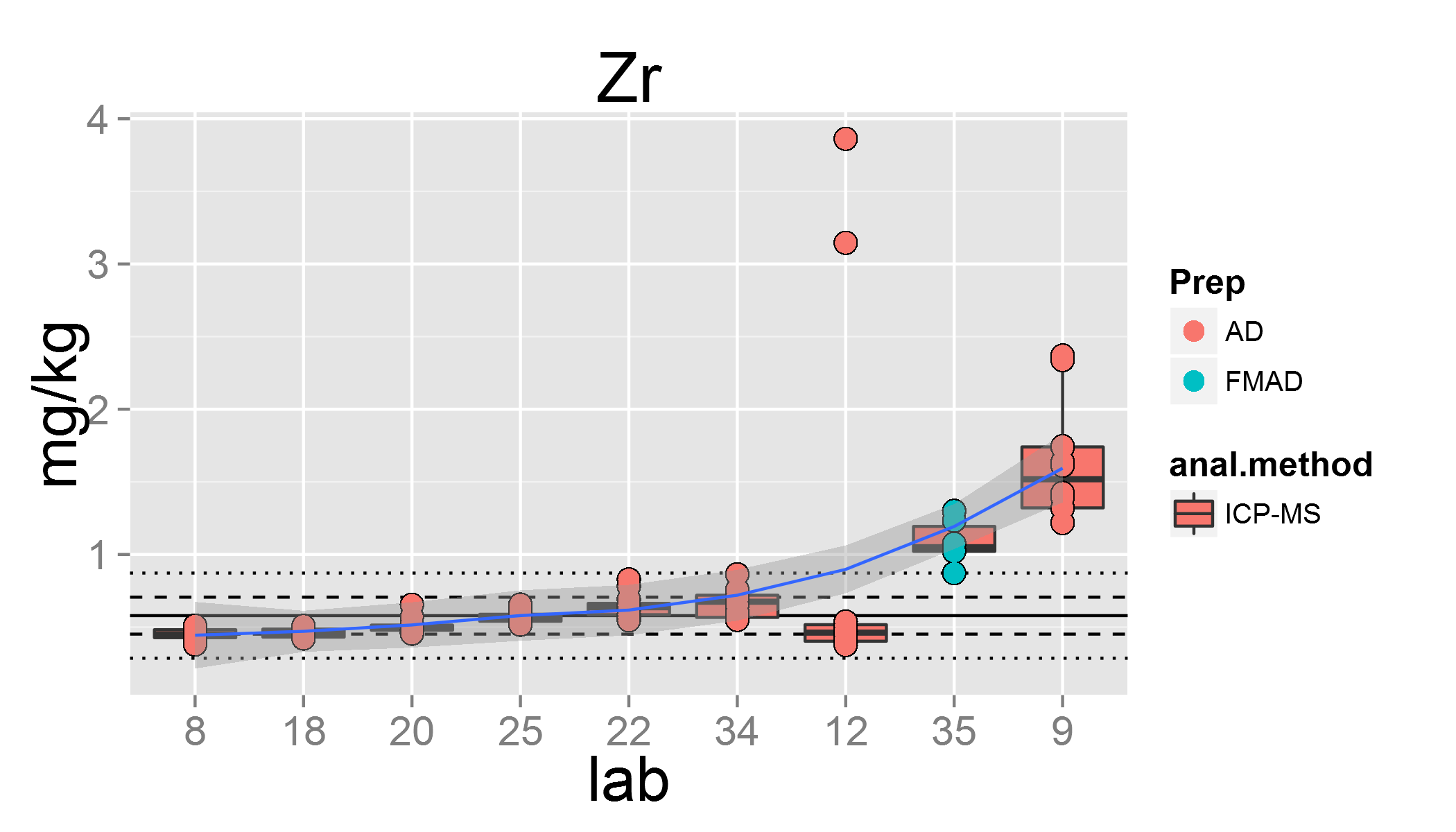
reference.line <- median.after  
u.Ulim <- median.after + u2.a\*t.value  
l.Ulim <- median.after - u2.a\*t.value  
ggplot(bymethod.n, aes(x=anal.method, y=median))+geom\_point(size=4)+geom\_errorbar(aes(ymin=median-se, ymax=median+se), width=0.05)+ geom\_abline(intercept = reference.line, slope = 0) + geom\_abline(intercept = u.Ulim, slope = 0, linetype ="dotted") + geom\_abline(intercept = l.Ulim, slope = 0, linetype = "dotted") + mytheme

plot of chunk plotting the method medians

### Final plot without outliers with u and U

plot\_lab(measurand, MorT, horw = FALSE, u = TRUE)

## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.

 ## Comparisons of method parameters based on median of the Lab+packet medians after outlier removal

print(bymethod.n)

## anal.method N mean median sd se  
## 1 ICP-MS 9 0.712 0.579 0.372 0.124

### final result based on median for property value and measurement uncertainty based on variance components

The **Zr** median of the Lab+packet median after outlier removal is **0.579** mg/kg  
The expanded standard uncertainty for the assigned value of Zr is **0.2932** with k = 2.31  
excluded labs for Zr is/are 10, 6, 26, 33, 31, 24  
labs remaining for calculations 9

### comments

outlier type | lab# :----| :------- based on Youden plots: |10, 26, 33, 31  
based on Mandel's k performance:|  
based on PP-XRF:|24  
based on detection limit:| 6

### comments

|  |  |
| --- | --- |
| outlier type | lab# |
| based on Youden plots: | 10, 26, 33, 31 |
| based on Mandel's k performance: | 10, 26, 33, 31 |
| based on PP-XRF: | 10, 26, 33, 31 |

sessionInfo()

## R version 3.1.1 (2014-07-10)  
## Platform: x86\_64-w64-mingw32/x64 (64-bit)  
##   
## locale:  
## [1] LC\_COLLATE=German\_Austria.1252 LC\_CTYPE=German\_Austria.1252   
## [3] LC\_MONETARY=German\_Austria.1252 LC\_NUMERIC=C   
## [5] LC\_TIME=German\_Austria.1252   
##   
## attached base packages:  
## [1] grid stats graphics grDevices utils datasets methods   
## [8] base   
##   
## other attached packages:  
## [1] knitr\_1.6 plyr\_1.8.1 nlme\_3.1-117 ape\_3.1-4   
## [5] metRology\_0.9-17 plotflow\_1.0 gridExtra\_0.9.1 ggplot2\_1.0.0   
##   
## loaded via a namespace (and not attached):  
## [1] colorspace\_1.2-4 digest\_0.6.4 evaluate\_0.5.5   
## [4] formatR\_0.10 gtable\_0.1.2 htmltools\_0.2.4   
## [7] labeling\_0.2 lattice\_0.20-29 MASS\_7.3-33   
## [10] munsell\_0.4.2 numDeriv\_2012.9-1 proto\_0.3-10   
## [13] Rcpp\_0.11.2 reshape2\_1.4 rmarkdown\_0.2.50   
## [16] scales\_0.2.4 stringr\_0.6.2 tools\_3.1.1   
## [19] yaml\_2.1.13