

MUH-1 Certification

TCM

Sunday, July 20, 2014

Automated analysing submitted data for MUH-1 based on defined outliers

Thomas Meisel (2014-08-03)

defining the RM and measurand to be analysed

```
refmat <- 'MUH' # defining the RM
```

opts_chunk\$set(dev="png", dev.args=list(type="cairo"), dpi=300) options(base64_images = 'inline') #####
general comments to the design

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 "traceability" sample and is here used for quality control purposes. It was the task of the labs to prepare two independent sample preparations (i.e. digestions) of each packet and analyse the preparations on two different days. Labs thus should have submitted 12 values (3x2x2 PacketxPrepxDay). The outliers have been selected based in Youden plots, Mandel's k and detection limit criteria. In this file the property values and the uncertainties are calculated for all analytes of a specific candidate MUH

```
'%p%' <- function(x, y) {as.character(paste(x, y, sep = ""))}  
df <- data.frame(cbind(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0))  
names(df) <- c("date", "RM", "measurand", "mean.before", "mean.after", "median.before", "median.after",  
df <- df[!1,]  
write.table(df, "df1.txt", row.names=FALSE) # needed only the first time
```

```
# Data for certification project was gathered and joined in Excel. The files  
# were exported from Excel as xxxx.csv files to make them universally  
# 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'
```

```
source("Makefile.R")
```

importing the data and assigning factors

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

```
## 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 calculate
meanGOM.packet <- ddply(GOM, c("Lab", "Packet"), numcolwise(meanGOM)) # calculated the mean for each Pa
## mean over mean of packets within lab
GOM.mean <- ddply(meanGOM.packet, c("Lab")), numcolwise(meanGOM))
GOM.mean <- merge(GOM.mean, OKUM.methods, by="Lab")
```

```
## median over packets within lab
medianGOM <- function(x) median(x, na.rm=TRUE)
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))
GOM.median <- merge(GOM.median, OKUM.methods, by="Lab")
```

initial calculations with complete data set

plots before outlier removal and outlier removal

```
sequence <- seq(from = 1, to = length(names(MUH.outlier)), by = 3)
col <- MUH.outlier[,c(sequence)]
col.names <- colnames(col)
for (m in col.names) {
  measurand.name <- m
  switch(
    reformat,
    GAS = rm1 <- 2,
    MUH = rm1 <- 1,
    OKUM = rm1 <- 0
  )
  if(rm1 > 0)
  {measurand <- measurand.name %p% '.' %p% rm1
  } else
  {
    measurand <- measurand.name
  }
  MorT <- grep(measurand.name, colnames(GOM), fixed=TRUE) # finding the position of the measurand.name
  ifelse(MorT[1]< 21, MorT <- 'M', MorT<-'T') # testing if measurand is a major or trace element/componu
  ifelse(MorT == "T", unit <- 'mg/kg', unit <- 'g/100g') # testing which unit is needed

print(plot_method(measurand))

# outlier removal
```

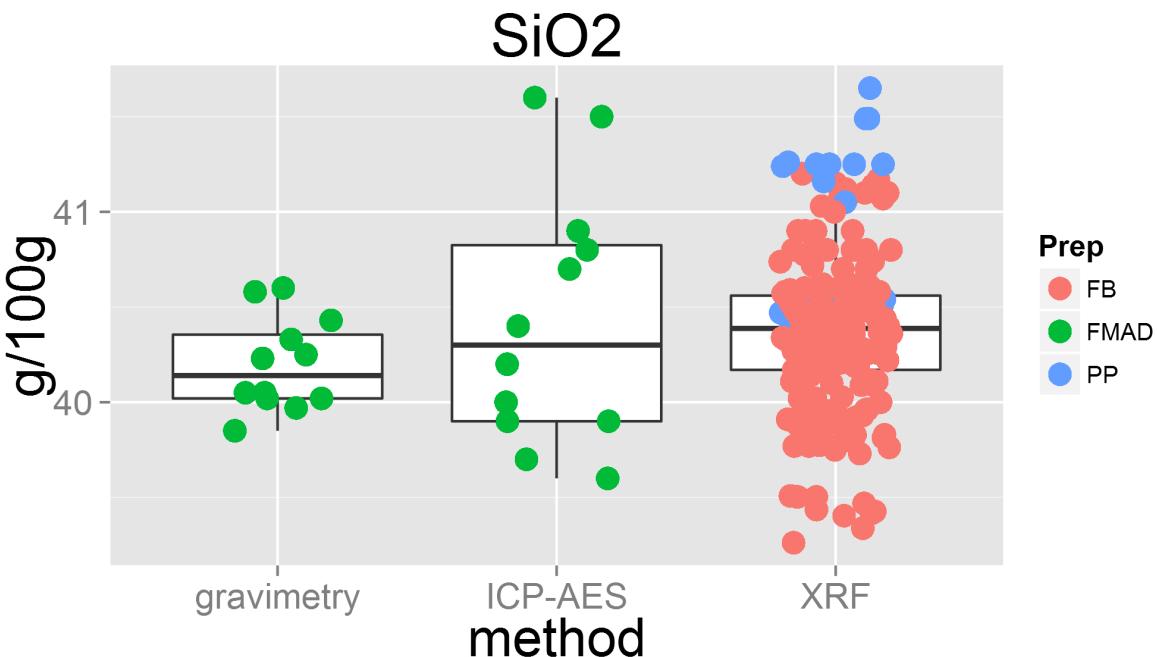
```

outlier <- MUH.outlier[[measurand.name]]
outlier <- na.omit(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
  message("Lab ", outlier[i], " was removed")
# print(summary(GOM[[measurand]]), na.rm=TRUE, digits=4))
}
}

```

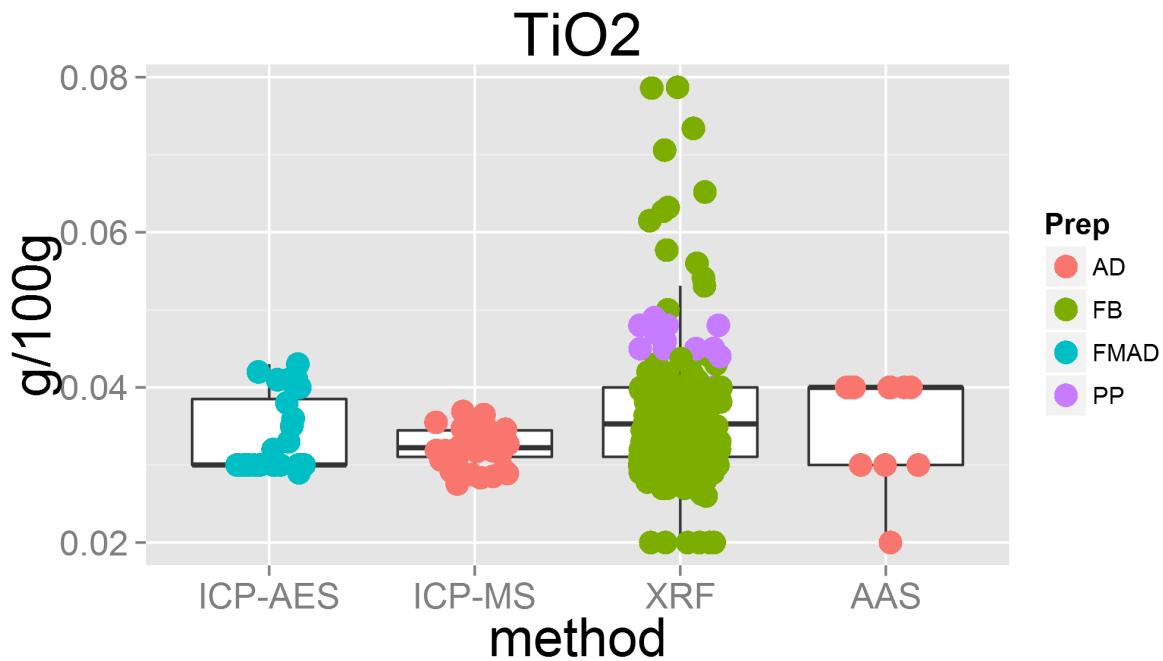
Warning: Removed 24 rows containing missing values (geom_point).

Lab 0 was removed



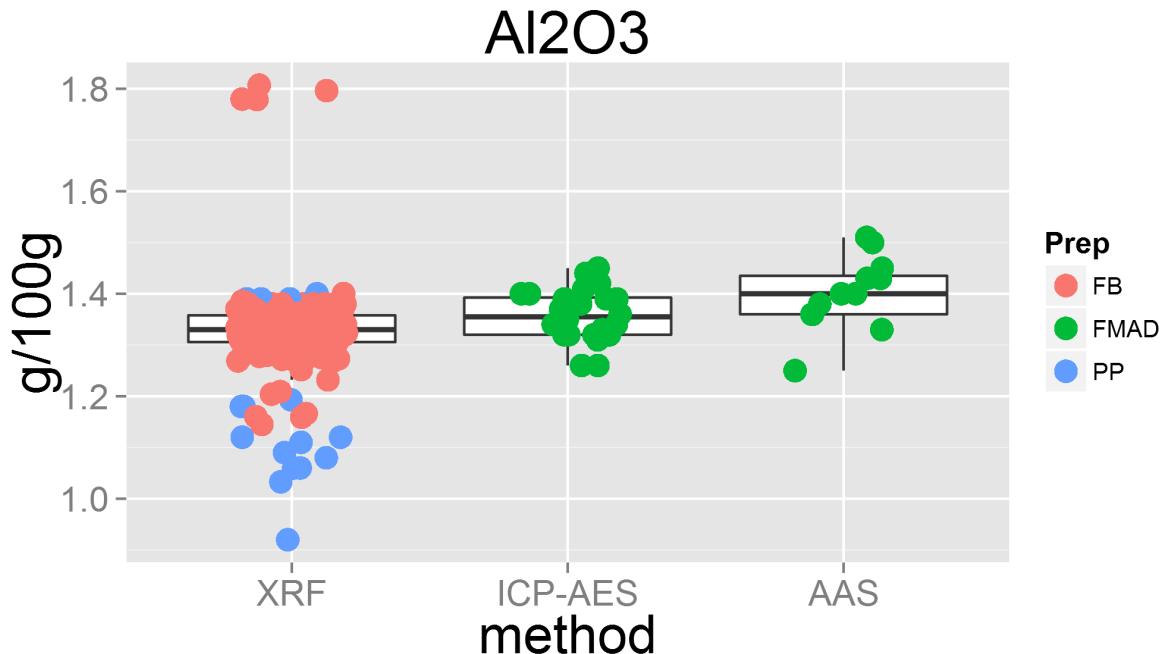
Warning: Removed 11 rows containing missing values (geom_point).

Lab 12 was removed
Lab 23 was removed



```
## Warning: Removed 22 rows containing missing values (geom_point).
```

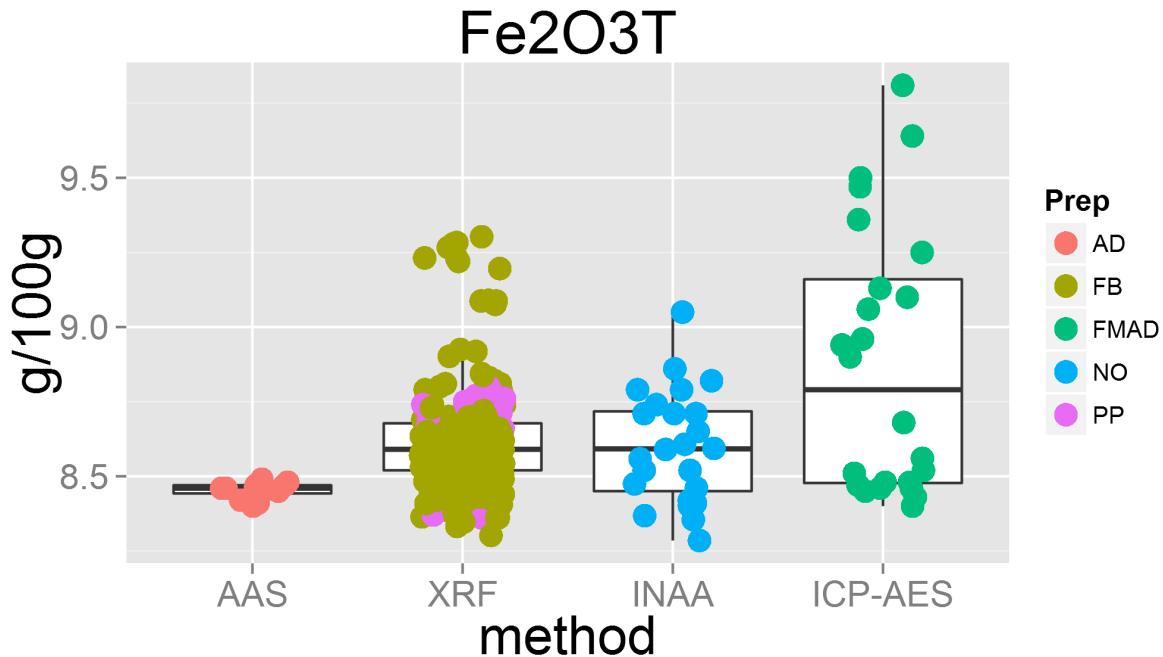
```
## Lab 12 was removed
## Lab 14 was removed
## Lab 31 was removed
```



```
## Warning: Removed 1 rows containing missing values (geom_point).
```

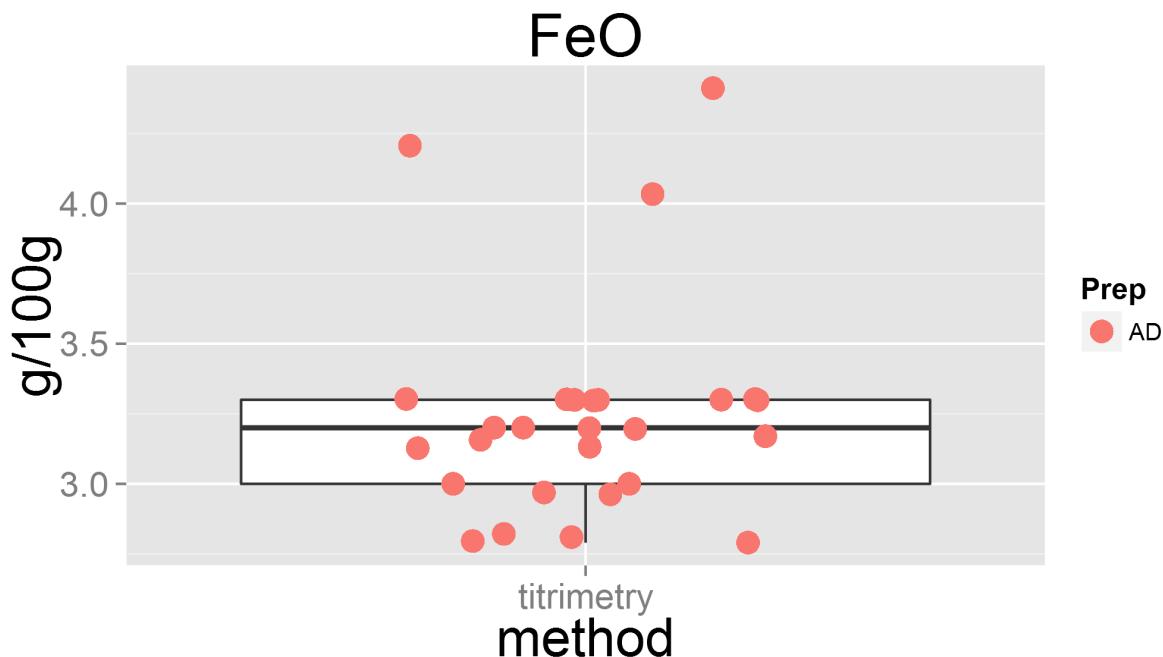
```
## Warning: Removed 14 rows containing missing values (geom_point).
```

```
## Lab 16 was removed  
## Lab 31 was removed
```



```
## Warning: Removed 3 rows containing missing values (geom_point).
```

```
## Lab 0 was removed
```

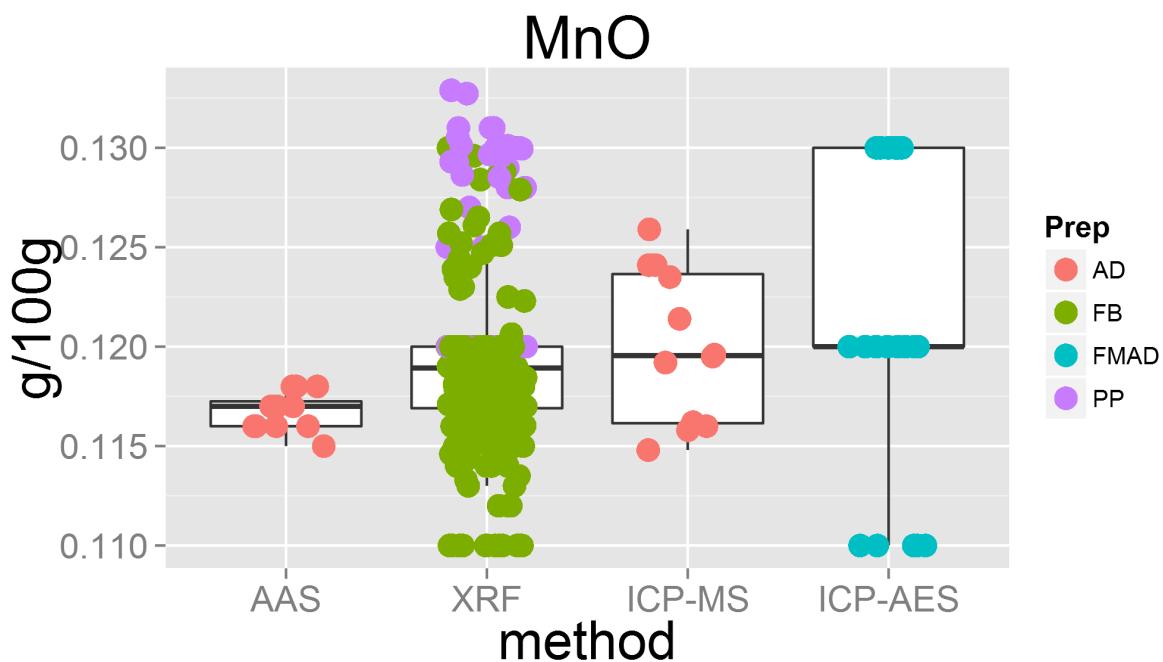


```

## Warning: Removed 60 rows containing missing values (geom_point).

## Lab 12 was removed
## Lab 16 was removed
## Lab 23 was removed
## Lab 24 was removed

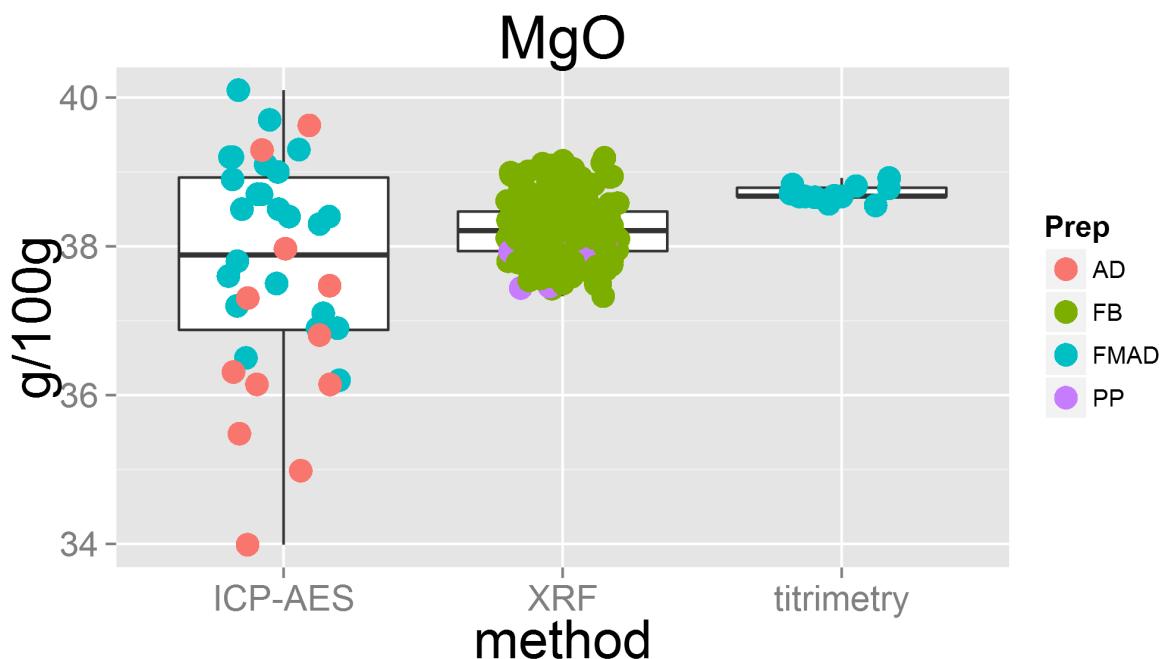
```



```

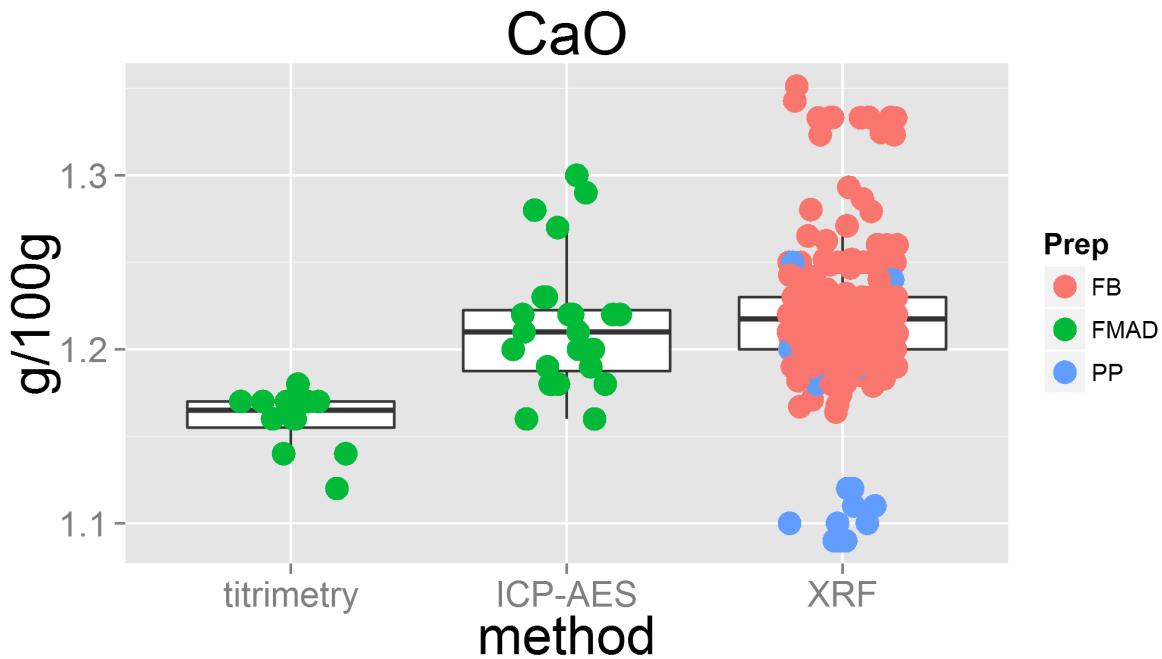
## Lab 32 was removed

```



```
## Warning: Removed 1 rows containing missing values (geom_point).  
## Warning: Removed 3 rows containing missing values (geom_point).  
## Warning: Removed 27 rows containing missing values (geom_point).
```

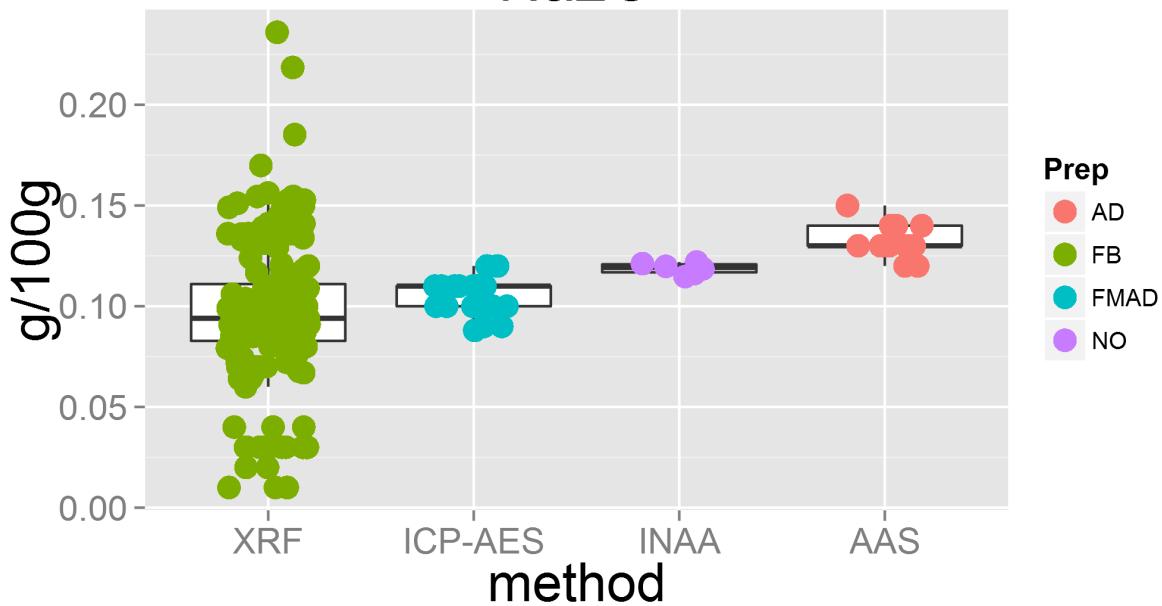
```
## Lab 12 was removed  
## Lab 30 was removed
```



```
## Warning: Removed 27 rows containing missing values (geom_point).
```

```
## Lab 3 was removed  
## Lab 22 was removed  
## Lab 33 was removed
```

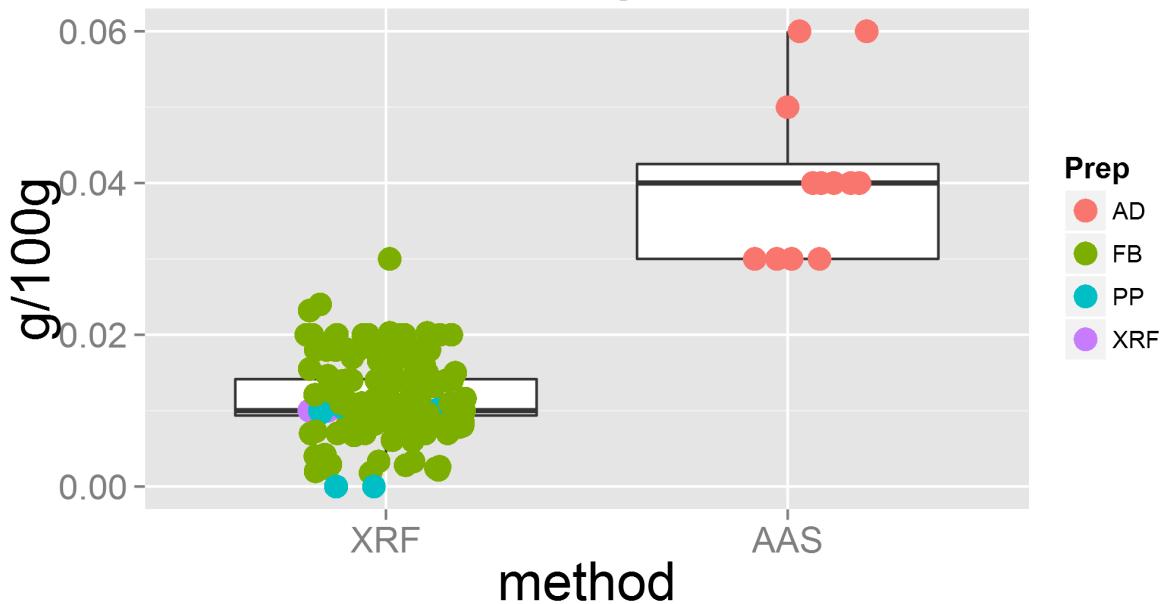
Na₂O



```
## Warning: Removed 8 rows containing missing values (geom_point).
```

```
## Lab 7 was removed  
## Lab 12 was removed
```

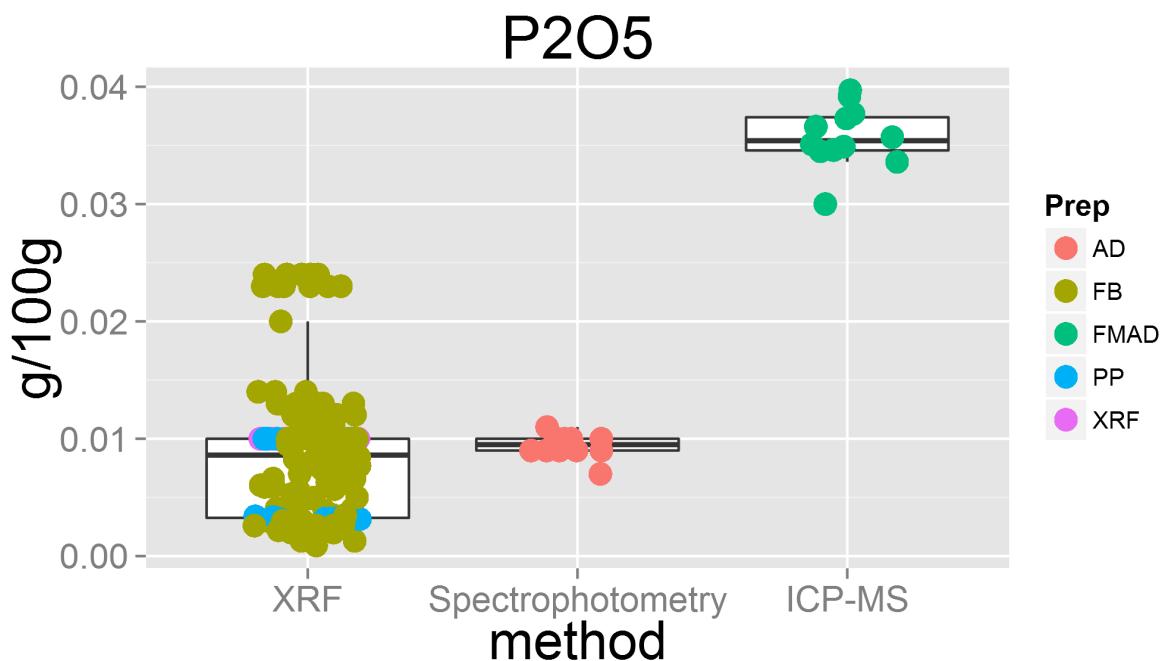
K₂O



```
## Warning: Removed 12 rows containing missing values (geom_point).
```

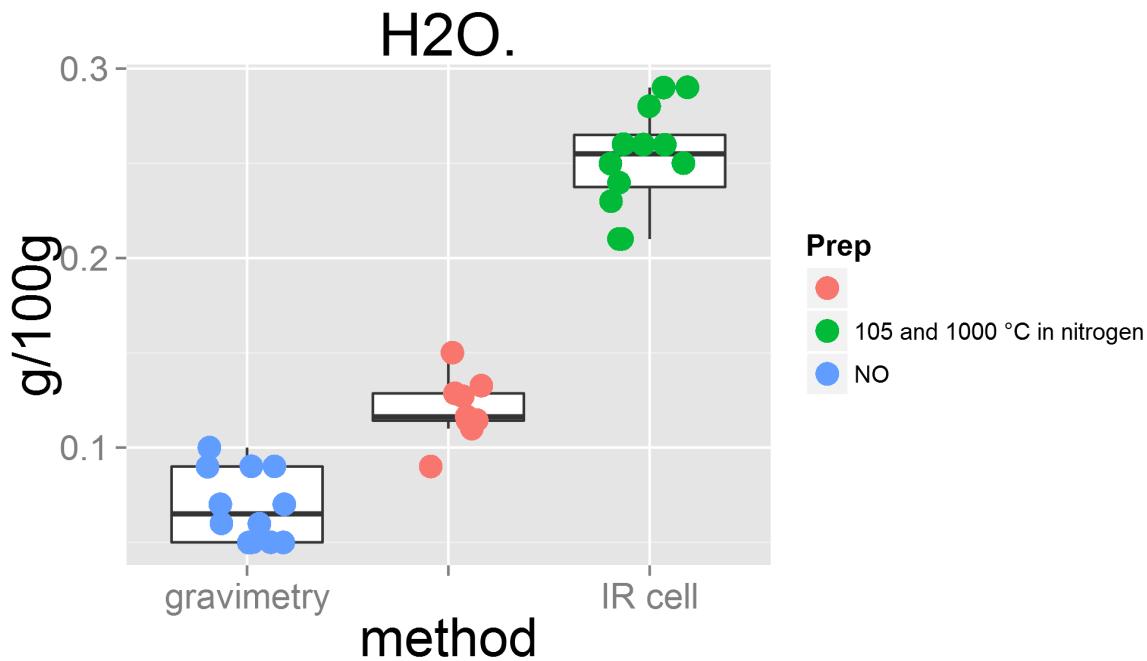
```
## Warning: Removed 1 rows containing missing values (geom_point).
## Warning: Removed 1 rows containing missing values (geom_point).

## Lab 6 was removed
## Lab 33 was removed
```

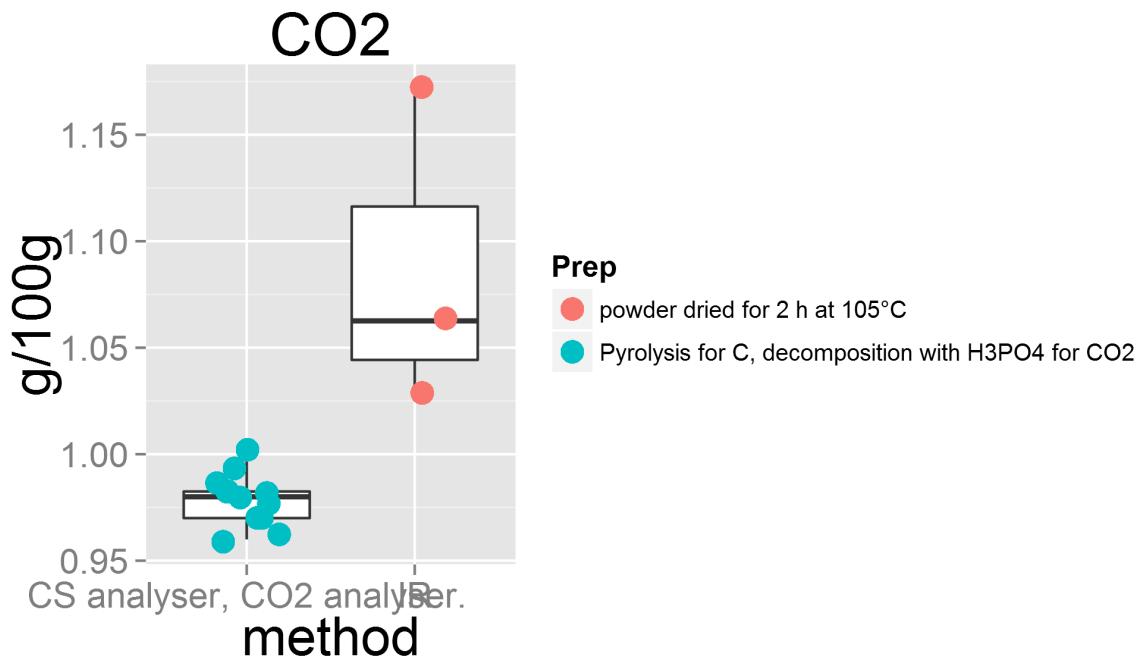


```
## Warning: Removed 1 rows containing missing values (geom_point).
```

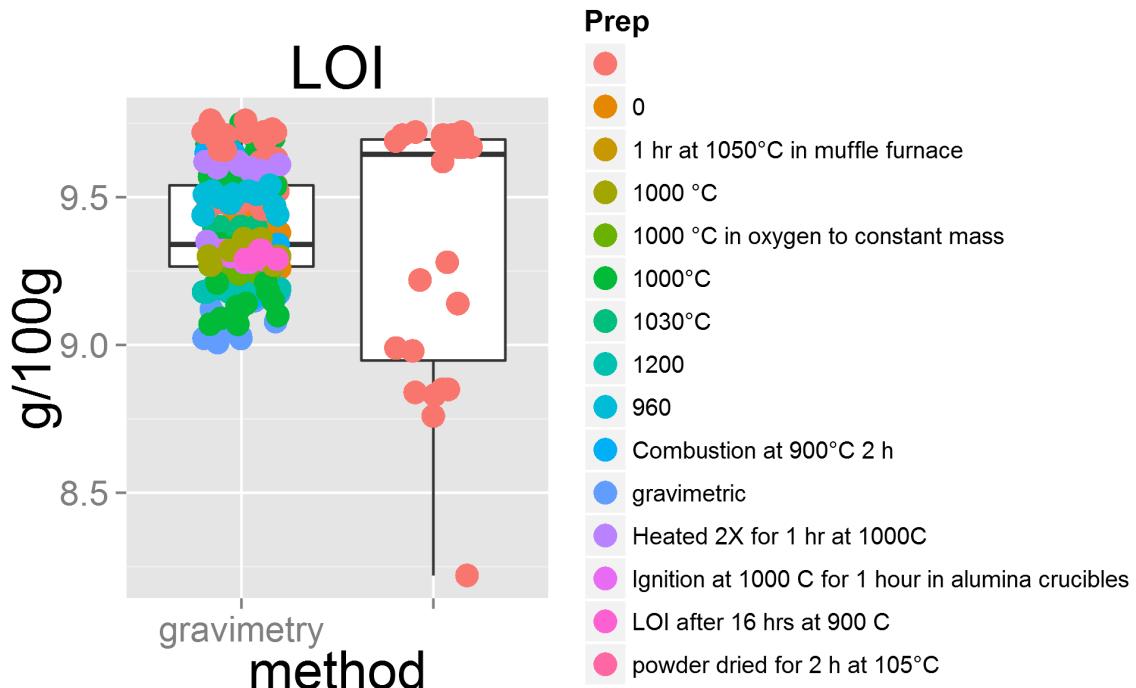
```
## Lab 0 was removed
```



```
## Lab 0 was removed
```

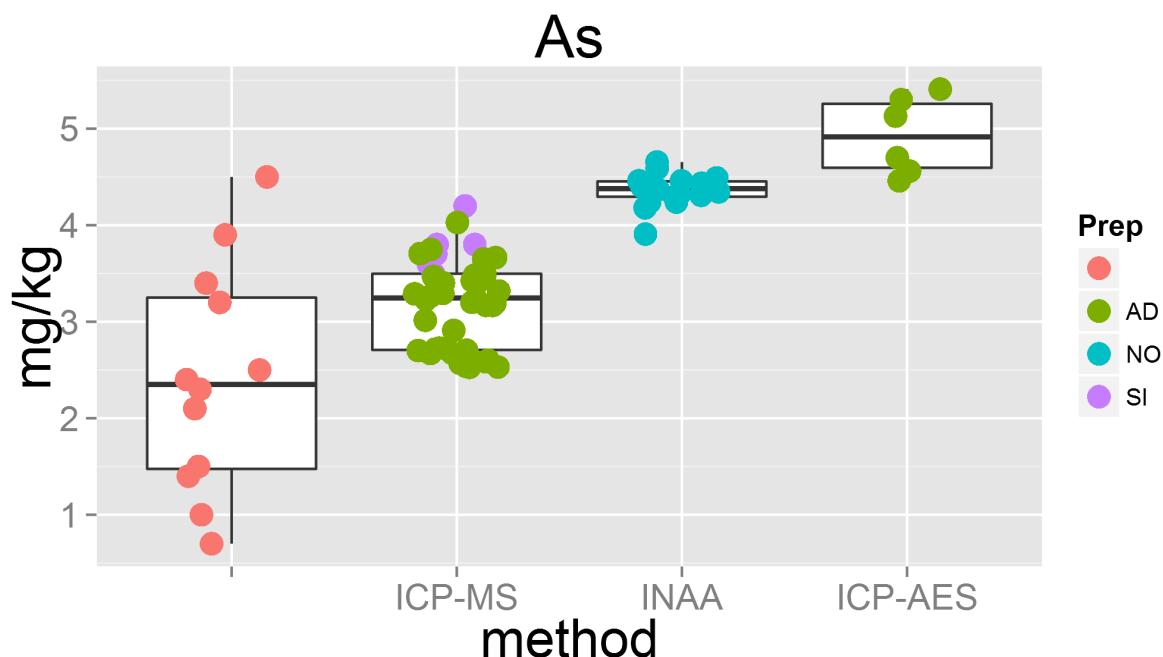


```
## Lab 0 was removed
```



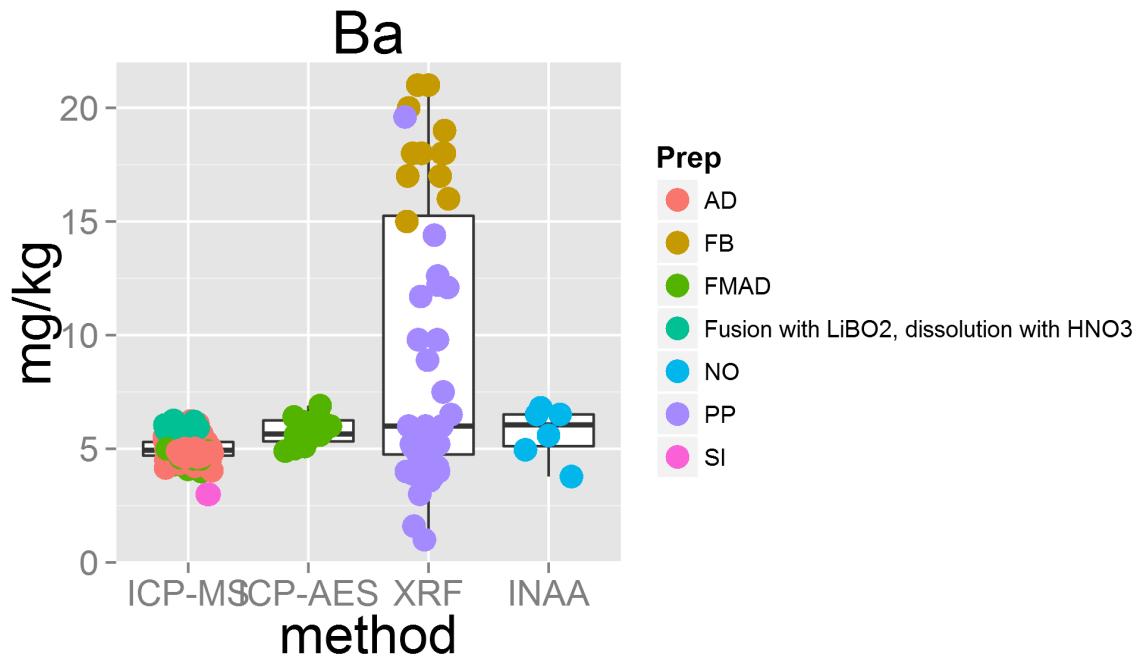
```
## Warning: Removed 1 rows containing missing values (geom_point).
```

```
## Lab 0 was removed
```



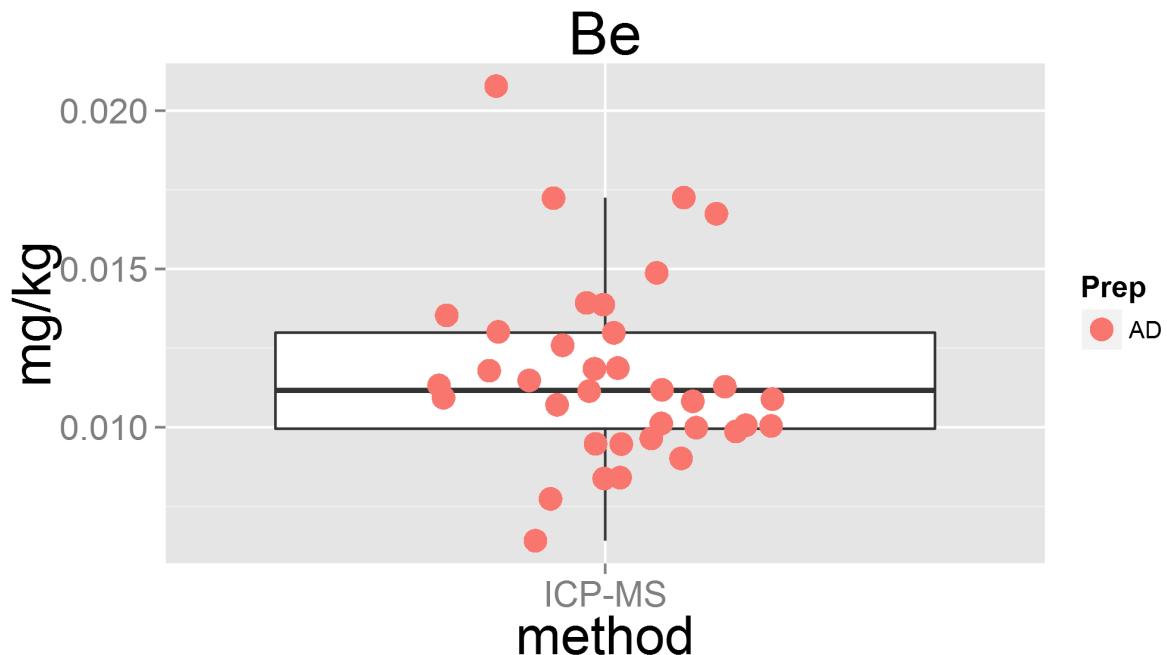
```
## Warning: Removed 4 rows containing missing values (geom_point).
```

```
## Lab 14 was removed  
## Lab 26 was removed
```



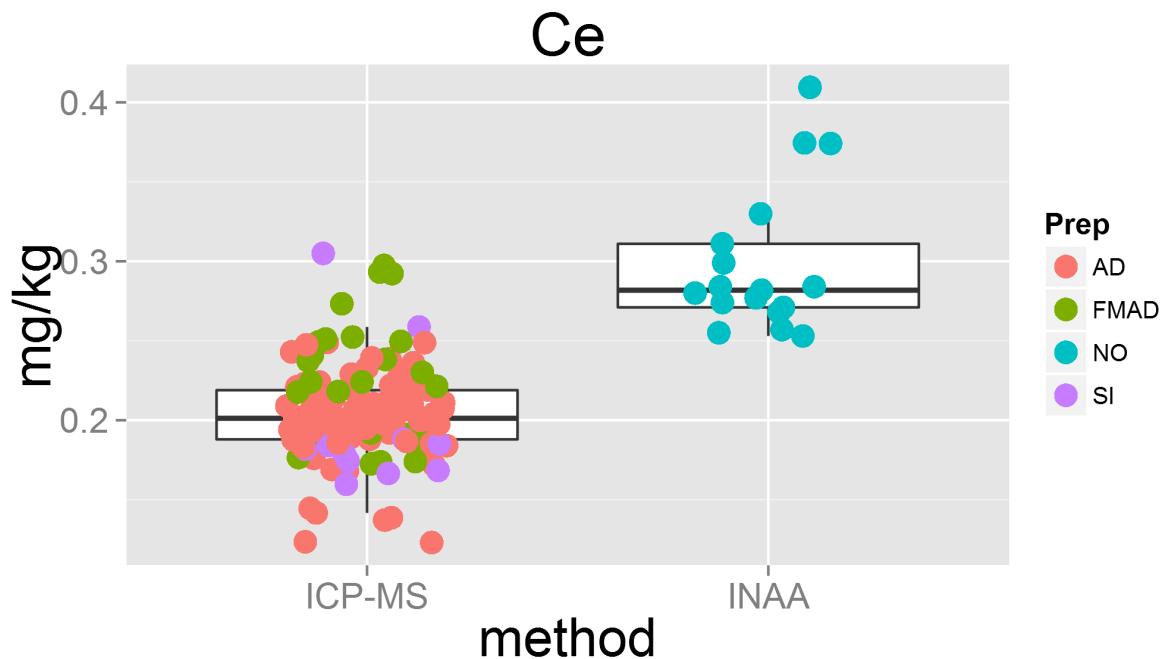
```
## Warning: Removed 1 rows containing missing values (geom_point).
```

```
## Lab 0 was removed
```



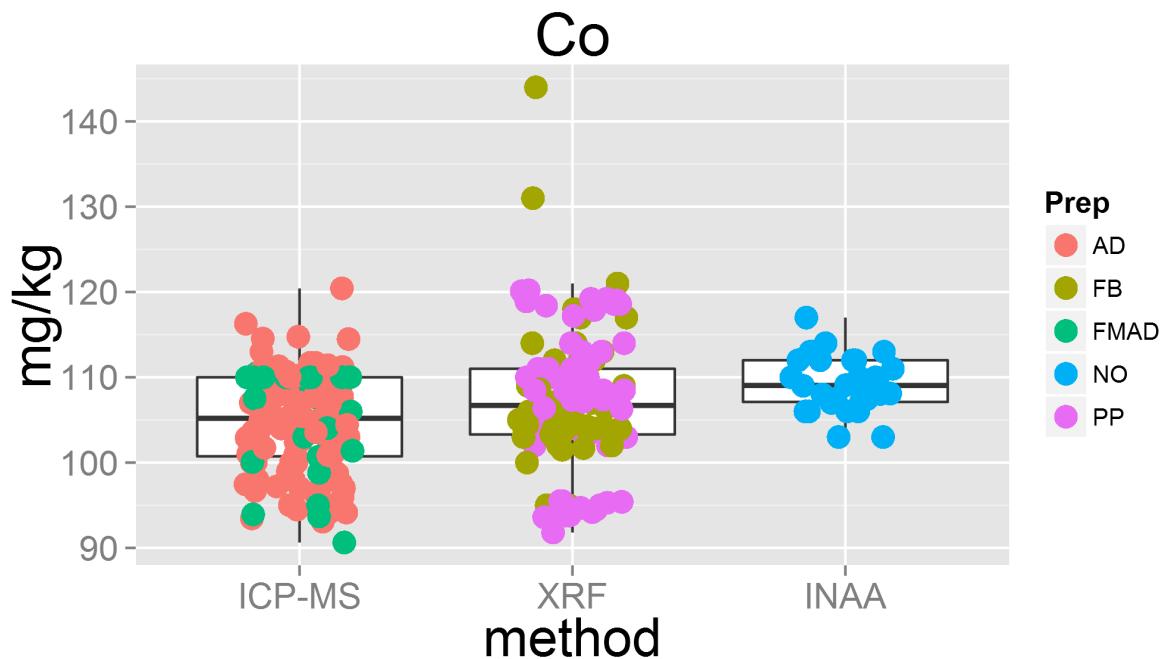
```
## Warning: Removed 9 rows containing missing values (geom_point).  
## Warning: Removed 3 rows containing missing values (geom_point).
```

```
## Lab 29 was removed
```



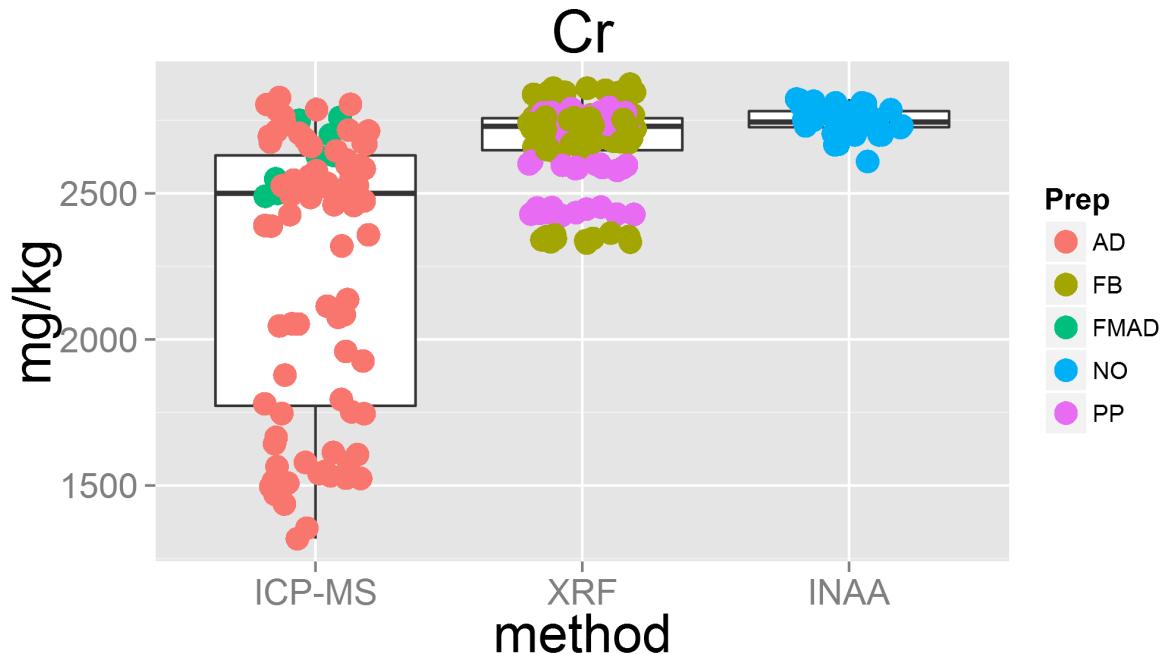
```
## Warning: Removed 2 rows containing missing values (geom_point).
```

```
## Lab 12 was removed  
## Lab 16 was removed  
## Lab 5 was removed
```



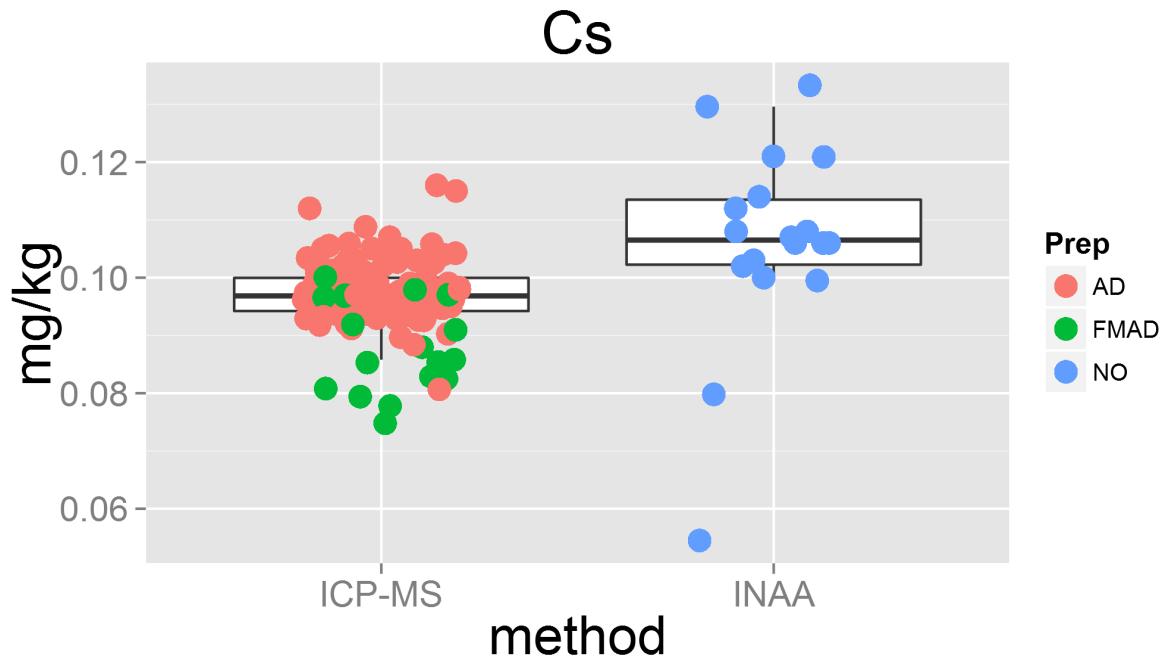
```
## Warning: Removed 24 rows containing missing values (geom_point).
## Warning: Removed 1 rows containing missing values (geom_point).
```

```
## Lab 8 was removed
## Lab 12 was removed
## Lab 18 was removed
```



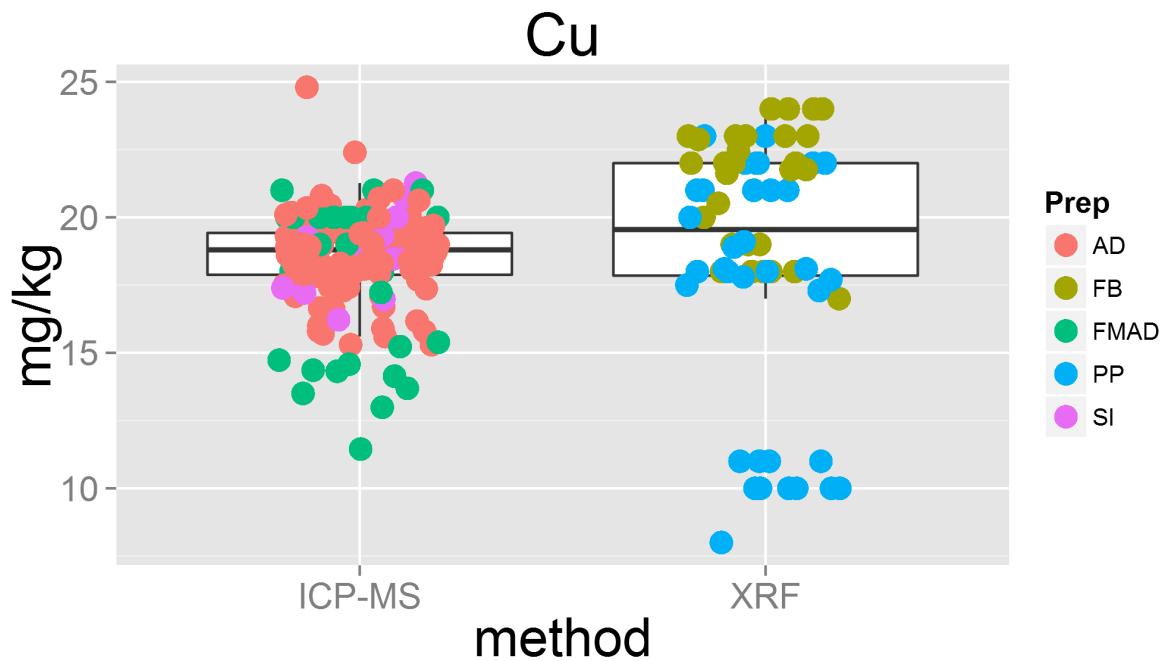
```
## Warning: Removed 13 rows containing missing values (geom_point).  
## Warning: Removed 3 rows containing missing values (geom_point).
```

```
## Lab 33 was removed
```



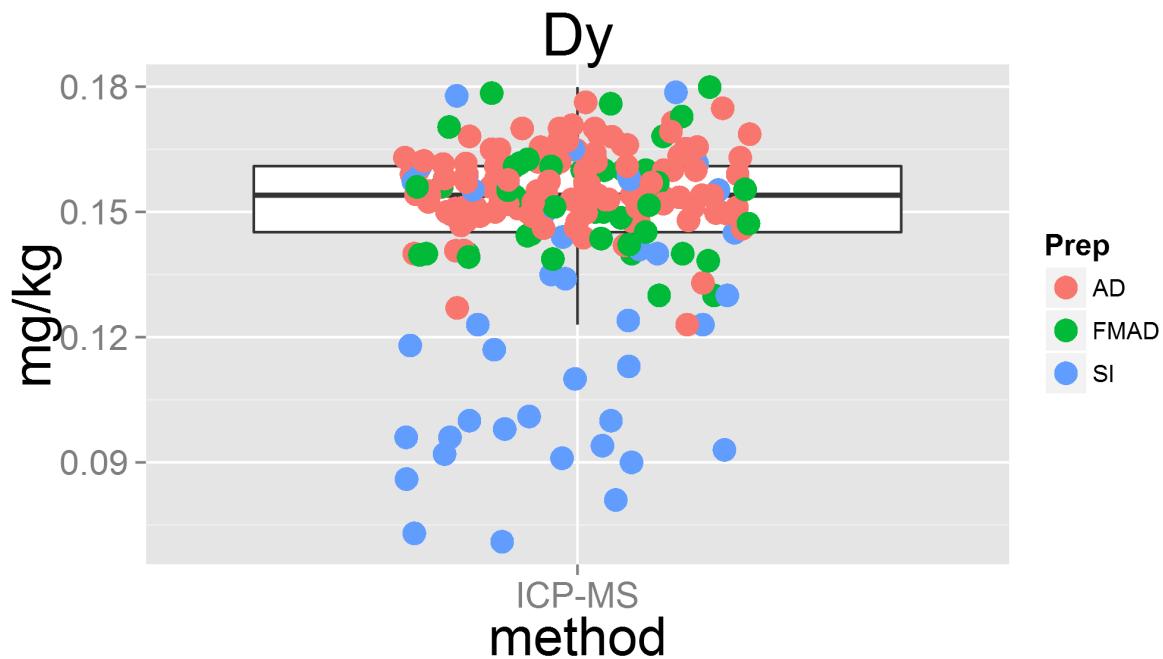
```
## Warning: Removed 15 rows containing missing values (geom_point).  
## Warning: Removed 12 rows containing missing values (geom_point).
```

```
## Lab 4 was removed  
## Lab 14 was removed  
## Lab 33 was removed
```



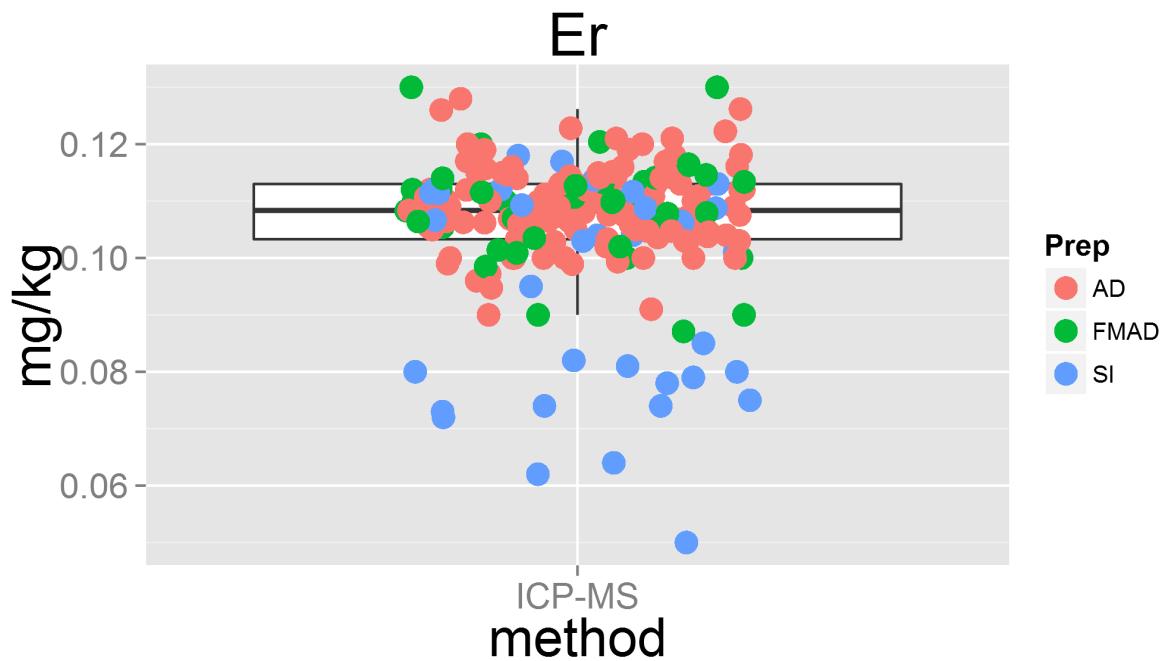
```
## Warning: Removed 19 rows containing missing values (geom_point).
```

```
## Lab 31 was removed
```



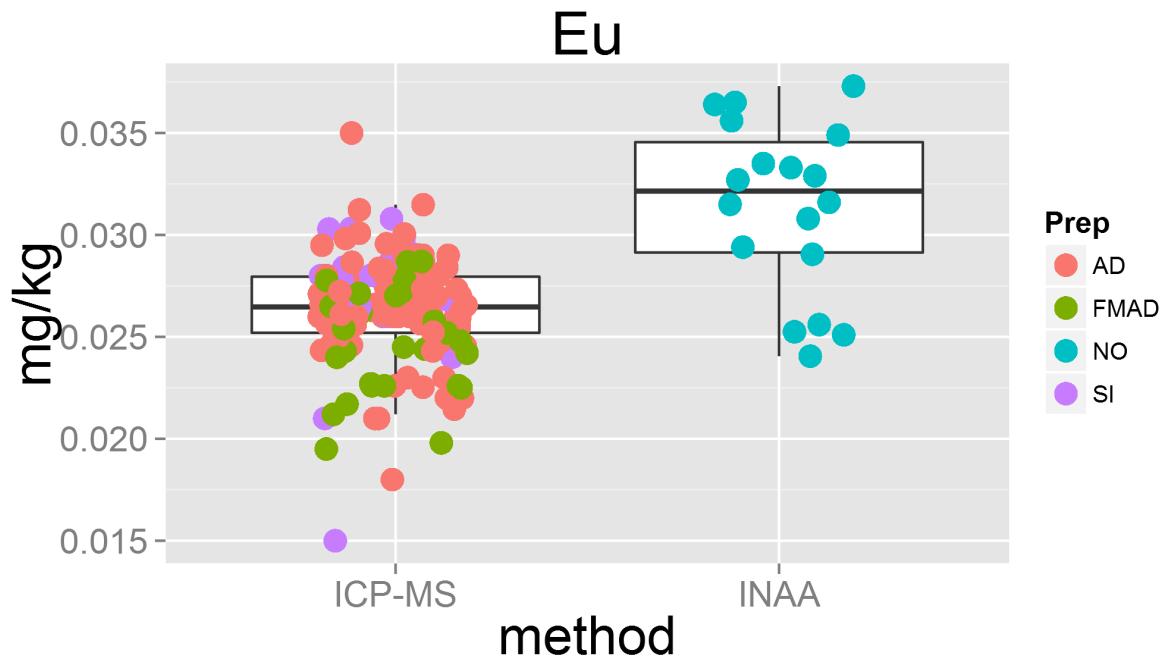
```
## Warning: Removed 20 rows containing missing values (geom_point).
```

```
## Lab 31 was removed
```

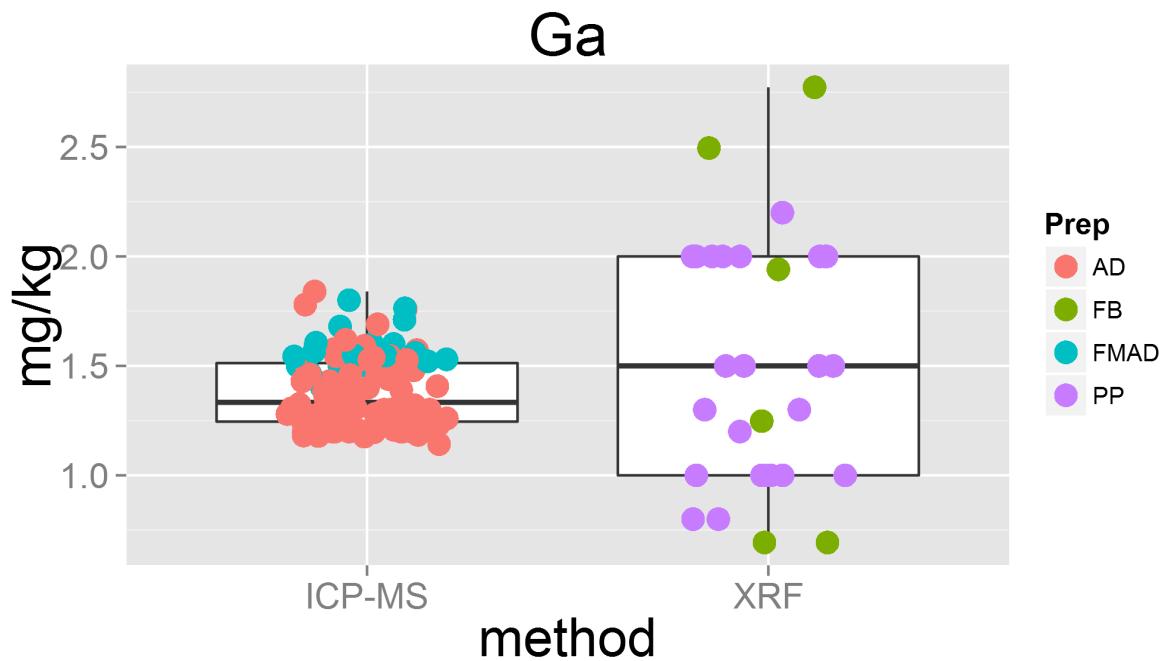


```
## Warning: Removed 8 rows containing missing values (geom_point).
```

```
## Lab 31 was removed
```

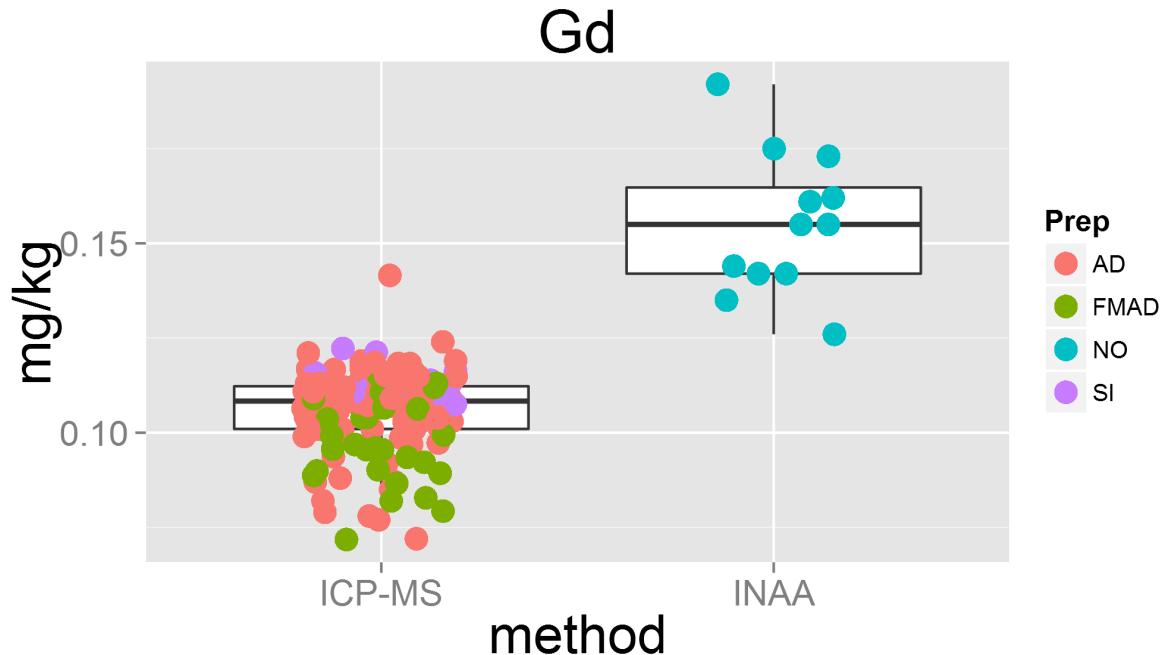


```
## Lab 4 was removed
## Lab 35 was removed
```



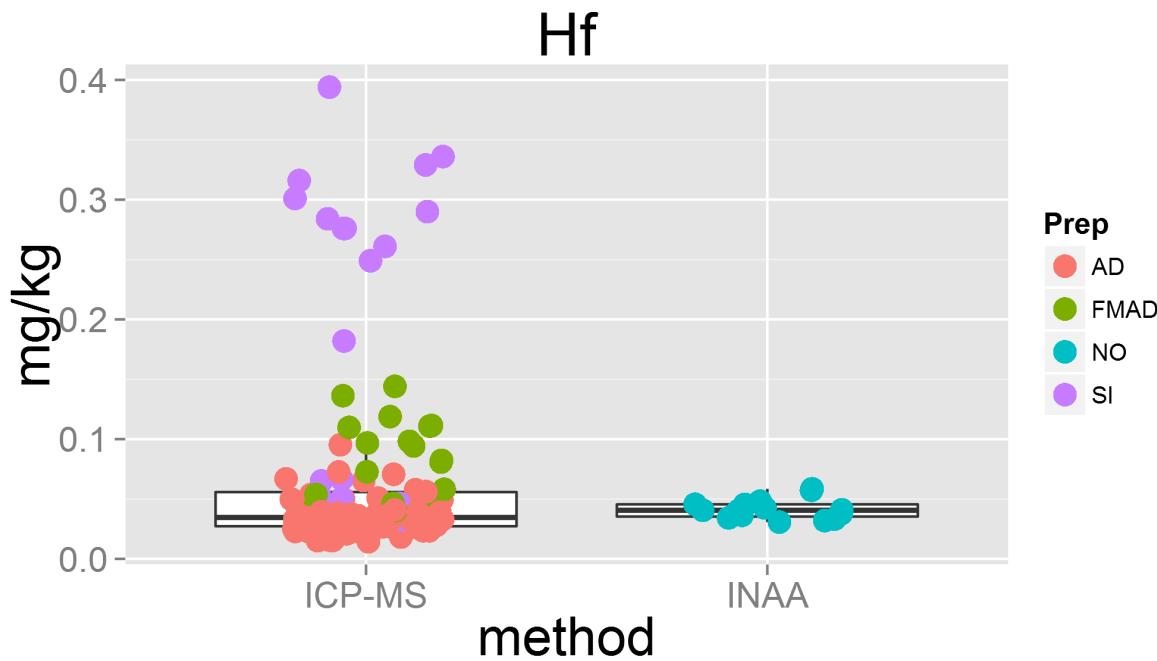
```
## Warning: Removed 10 rows containing missing values (geom_point).
```

```
## Lab 8 was removed
## Lab 31 was removed
```



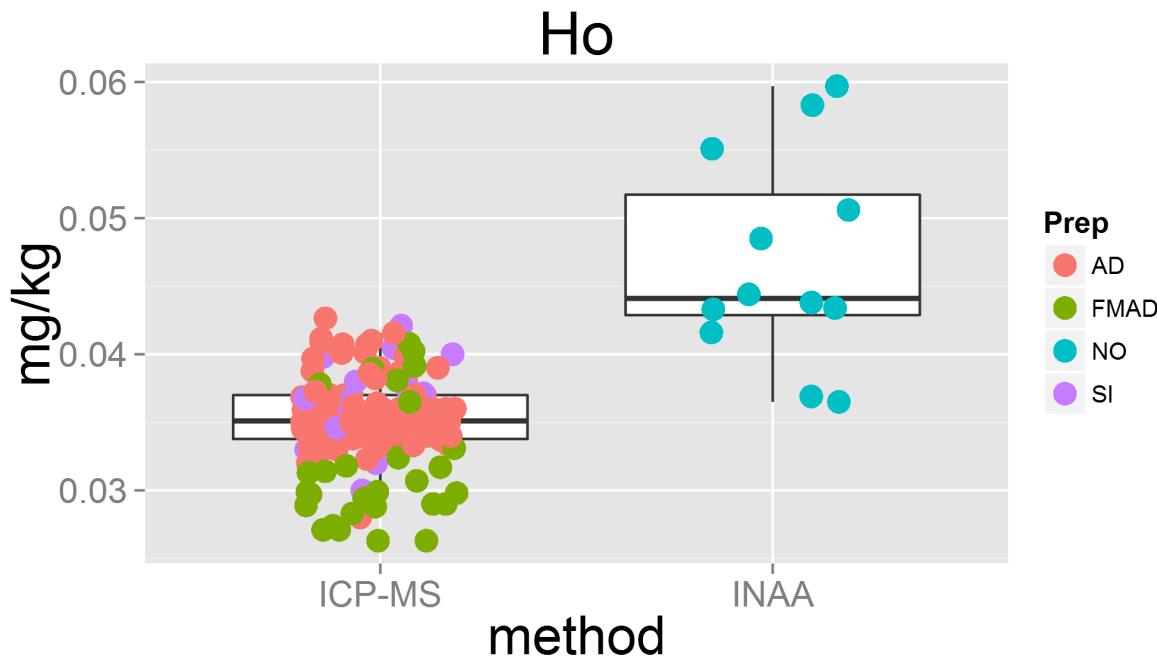
```
## Warning: Removed 18 rows containing missing values (geom_point).
```

```
## Lab 14 was removed
```



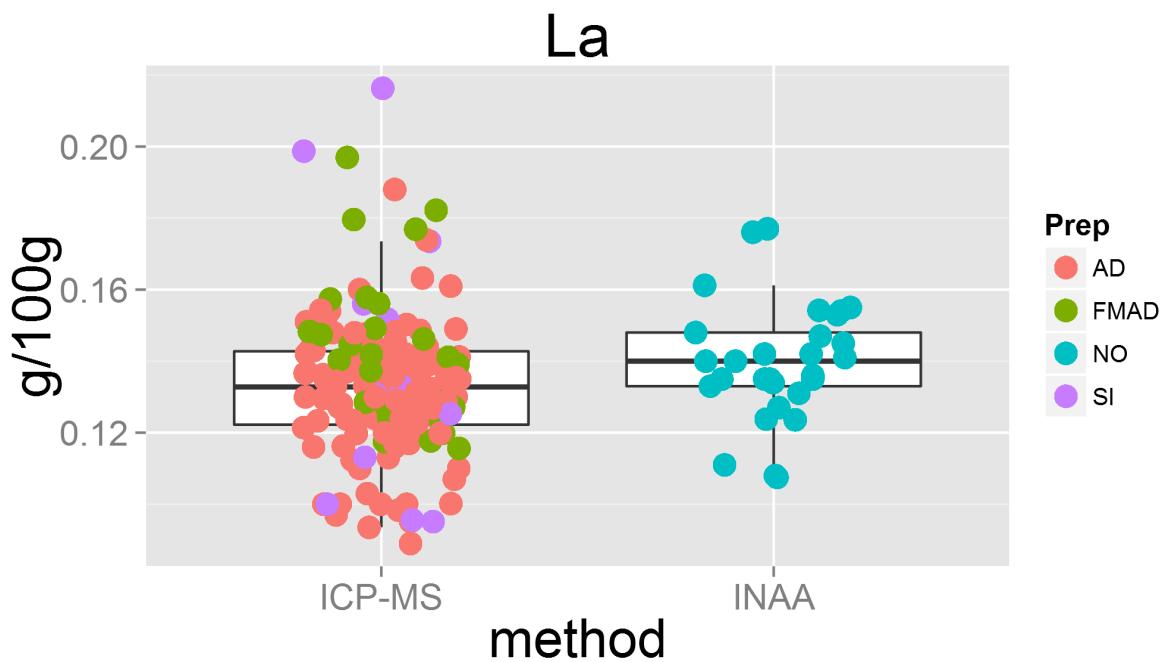
```
## Warning: Removed 11 rows containing missing values (geom_point).
```

```
## Lab 31 was removed  
## Lab 33 was removed
```



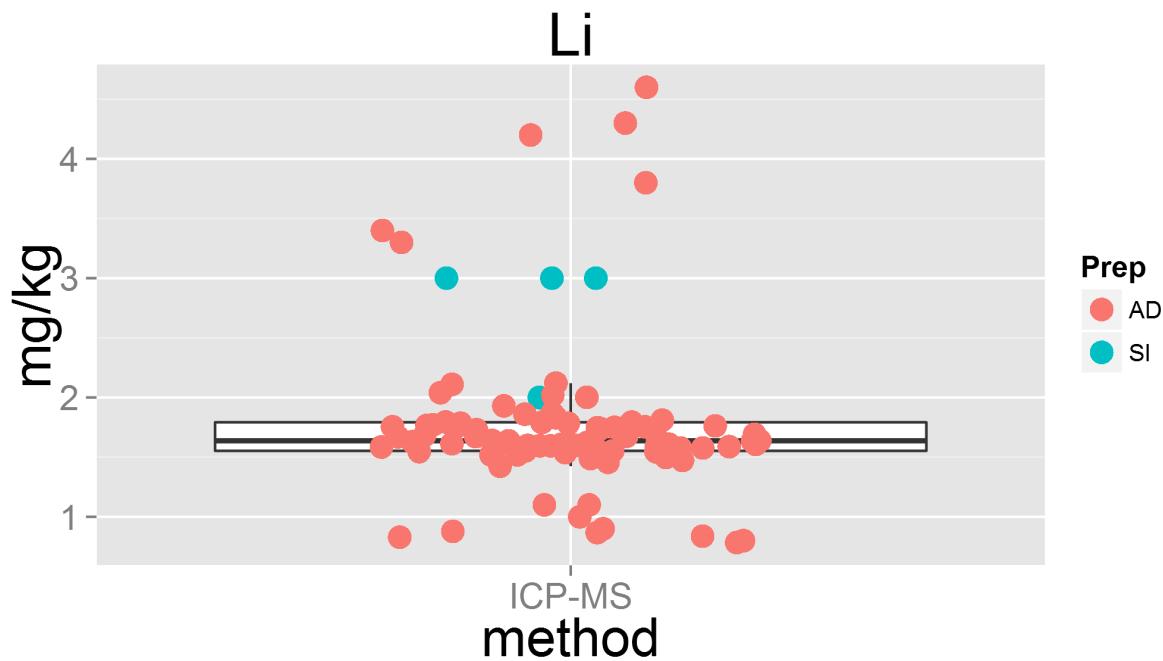
```
## Warning: Removed 9 rows containing missing values (geom_point).
## Warning: Removed 4 rows containing missing values (geom_point).

## Lab 16 was removed
## Lab 12 was removed
```



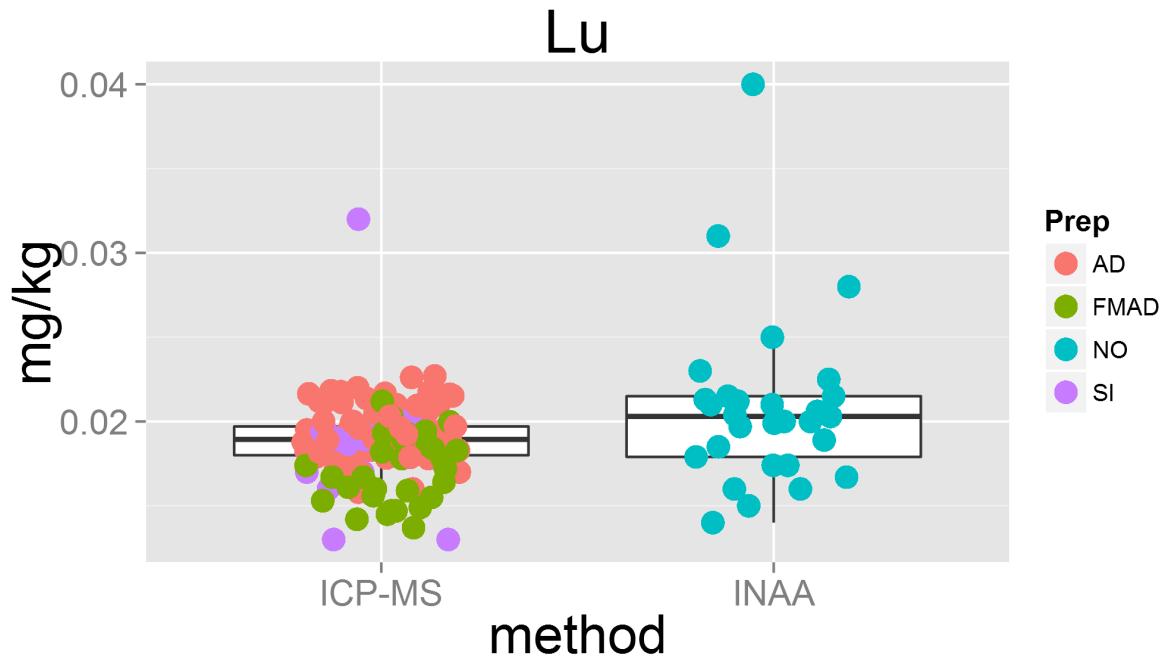
```
## Warning: Removed 19 rows containing missing values (geom_point).
```

```
## Lab 31 was removed
```



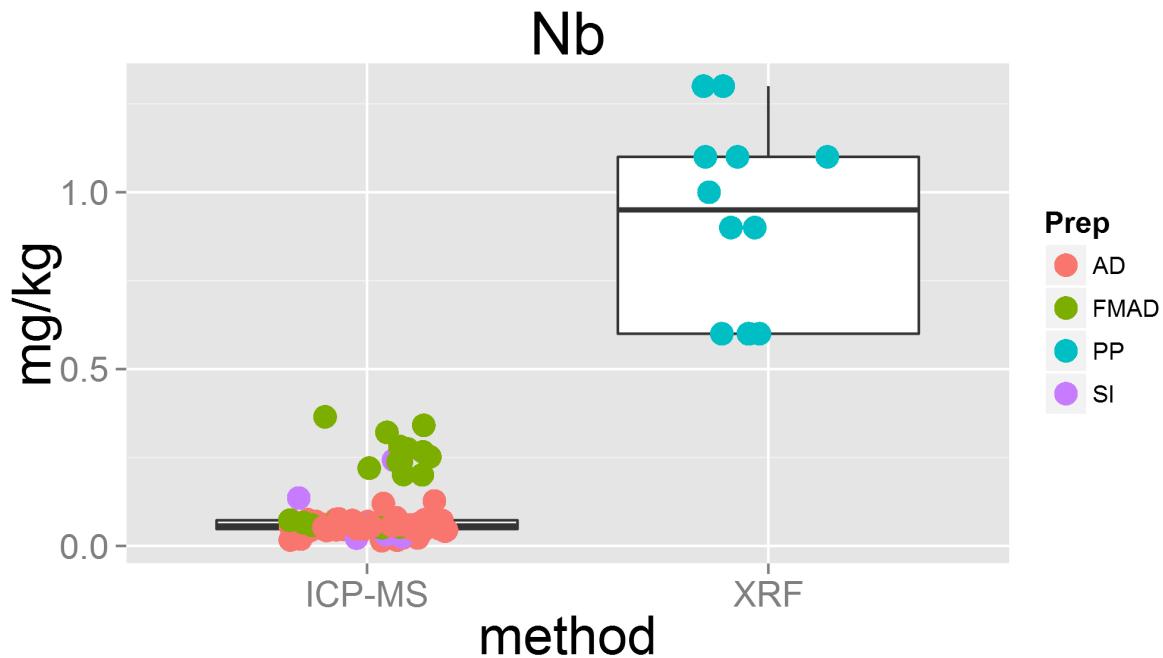
```
## Warning: Removed 12 rows containing missing values (geom_point).
## Warning: Removed 3 rows containing missing values (geom_point).
```

```
## Lab 4 was removed
```



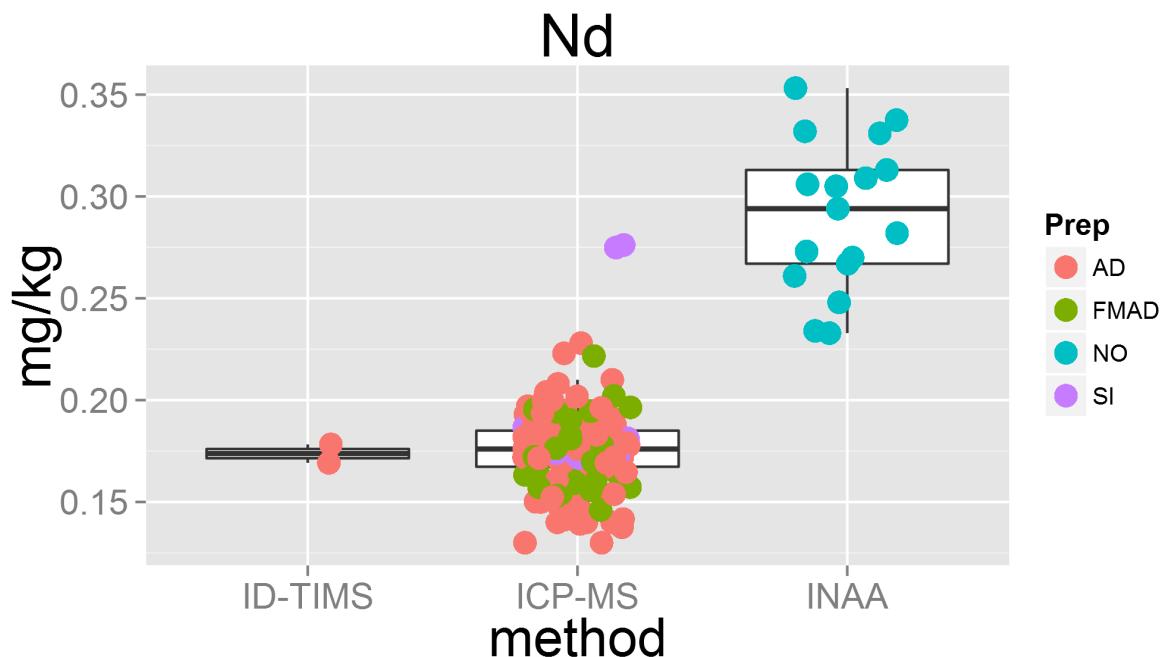
```
## Warning: Removed 17 rows containing missing values (geom_point).
```

```
## Lab 26 was removed  
## Lab 33 was removed
```



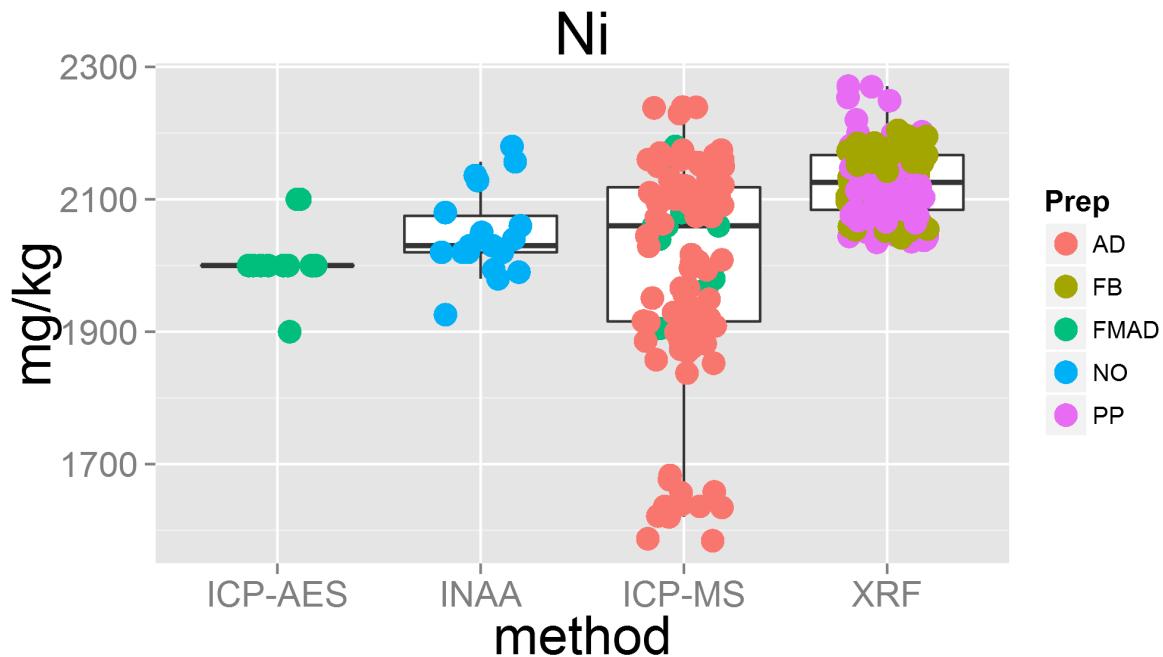
```
## Warning: Removed 12 rows containing missing values (geom_point).
```

```
## Lab 29 was removed  
## Lab 31 was removed
```



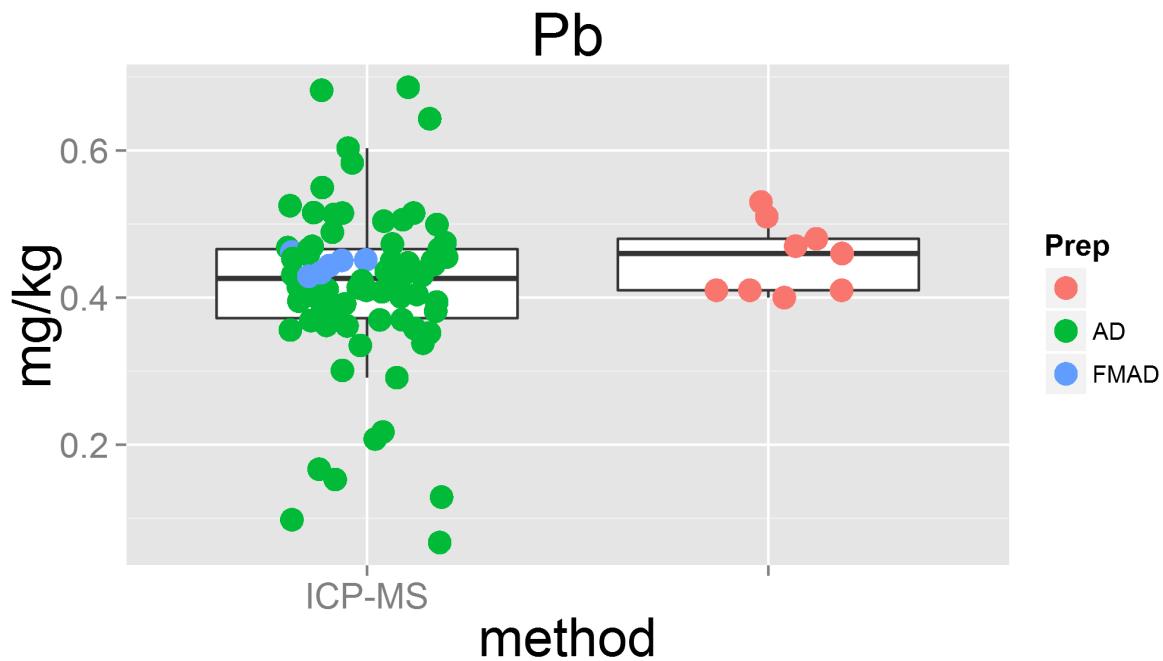
```
## Warning: Removed 3 rows containing missing values (geom_point).  
## Warning: Removed 2 rows containing missing values (geom_point).  
## Warning: Removed 2 rows containing missing values (geom_point).
```

```
## Lab 12 was removed  
## Lab 18 was removed  
## Lab 22 was removed
```



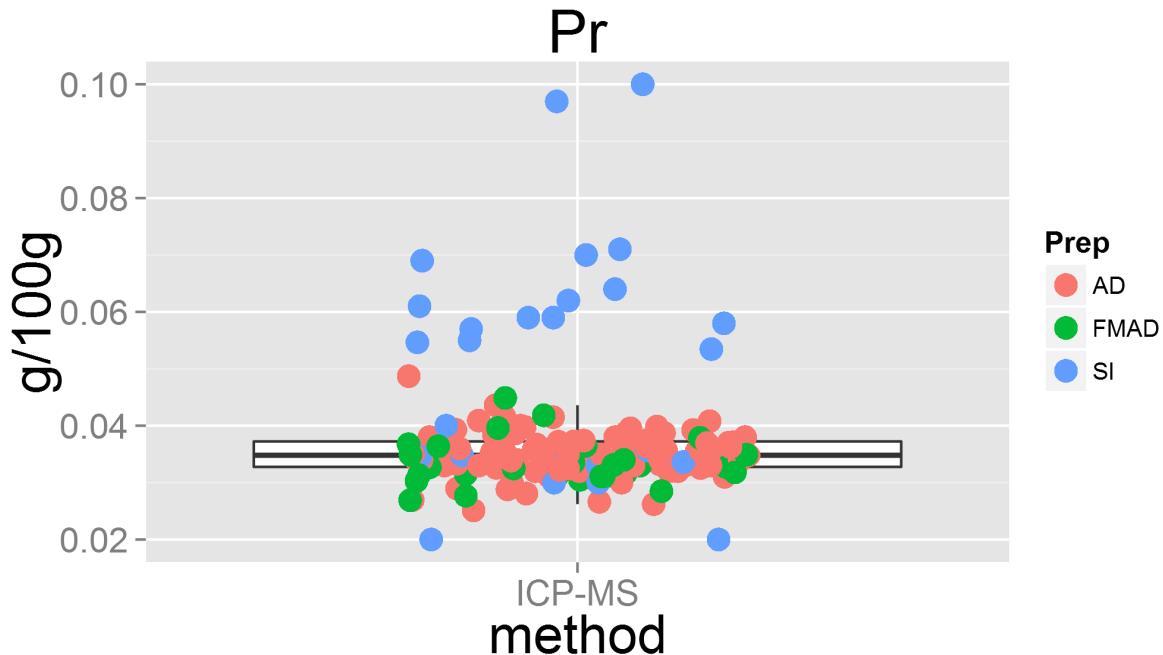
```
## Warning: Removed 10 rows containing missing values (geom_point).
```

```
## Lab 0 was removed
```



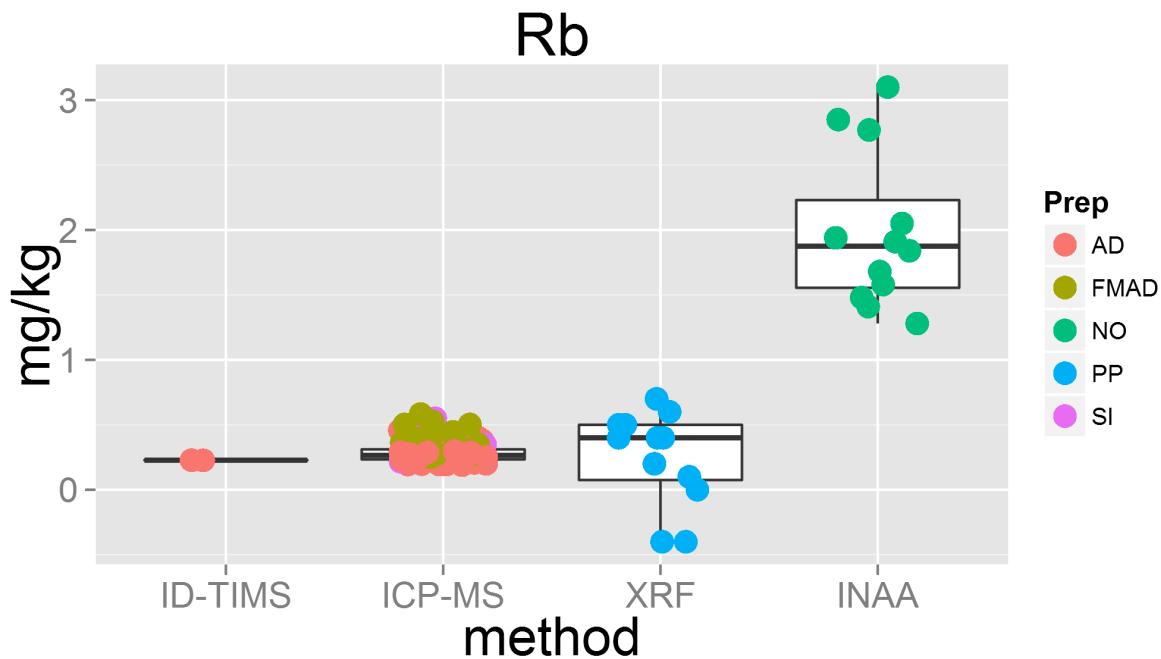
```
## Warning: Removed 20 rows containing missing values (geom_point).
```

```
## Lab 14 was removed
## Lab 31 was removed
```



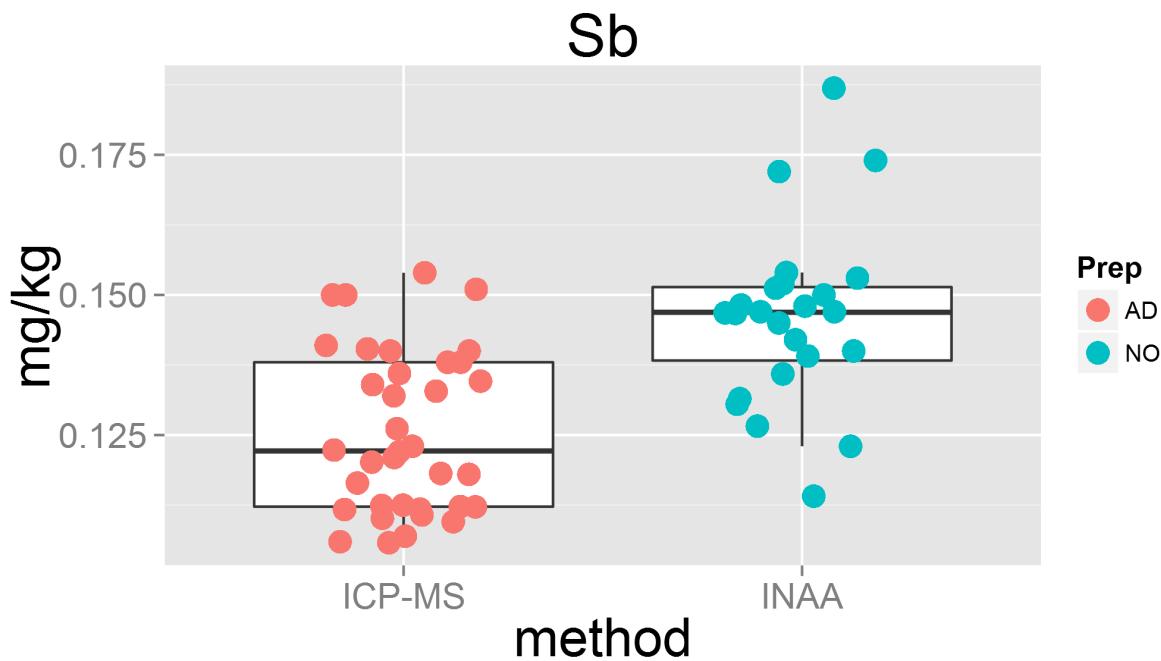
```
## Warning: Removed 11 rows containing missing values (geom_point).
```

```
## Lab 31 was removed  
## Lab 26 was removed  
## Lab 33 was removed
```



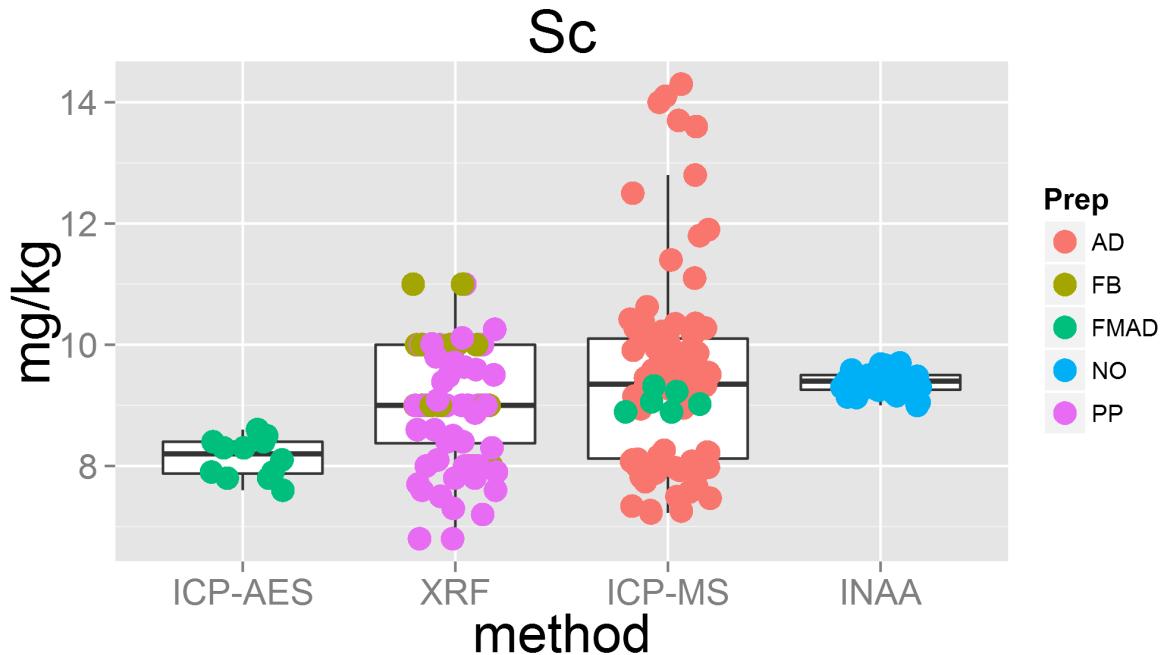
```
## Warning: Removed 4 rows containing missing values (geom_point).
```

```
## Lab 0 was removed
```



```
## Warning: Removed 5 rows containing missing values (geom_point).
```

```
## Lab 8 was removed
```

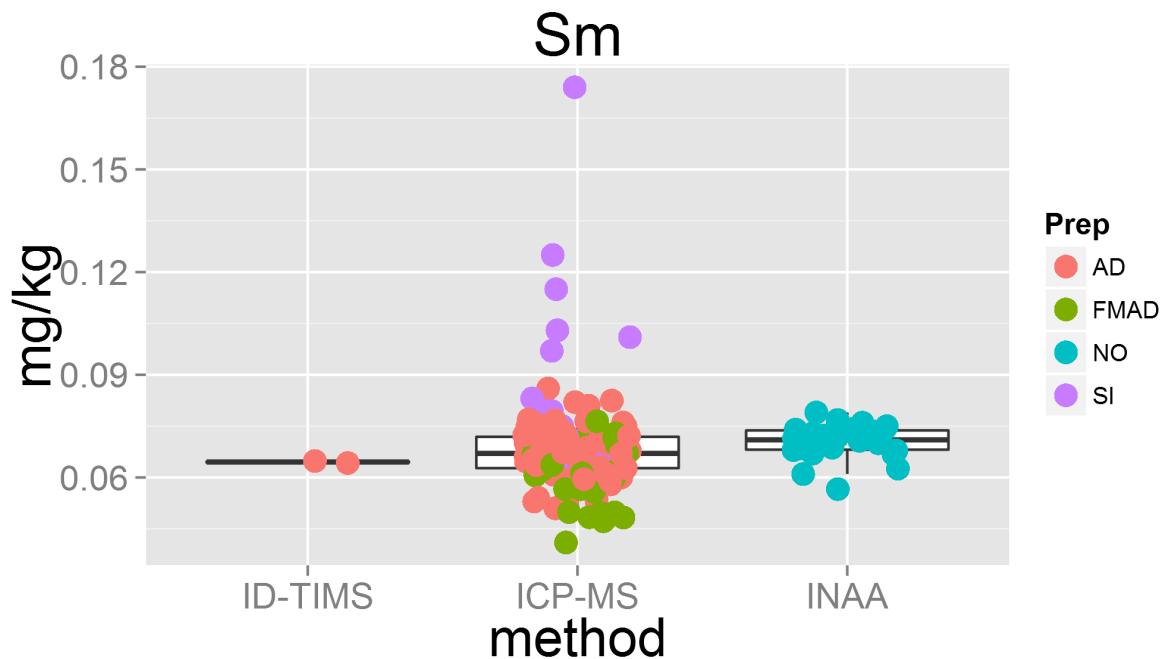


```
## Warning: Removed 12 rows containing missing values (geom_point).
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```

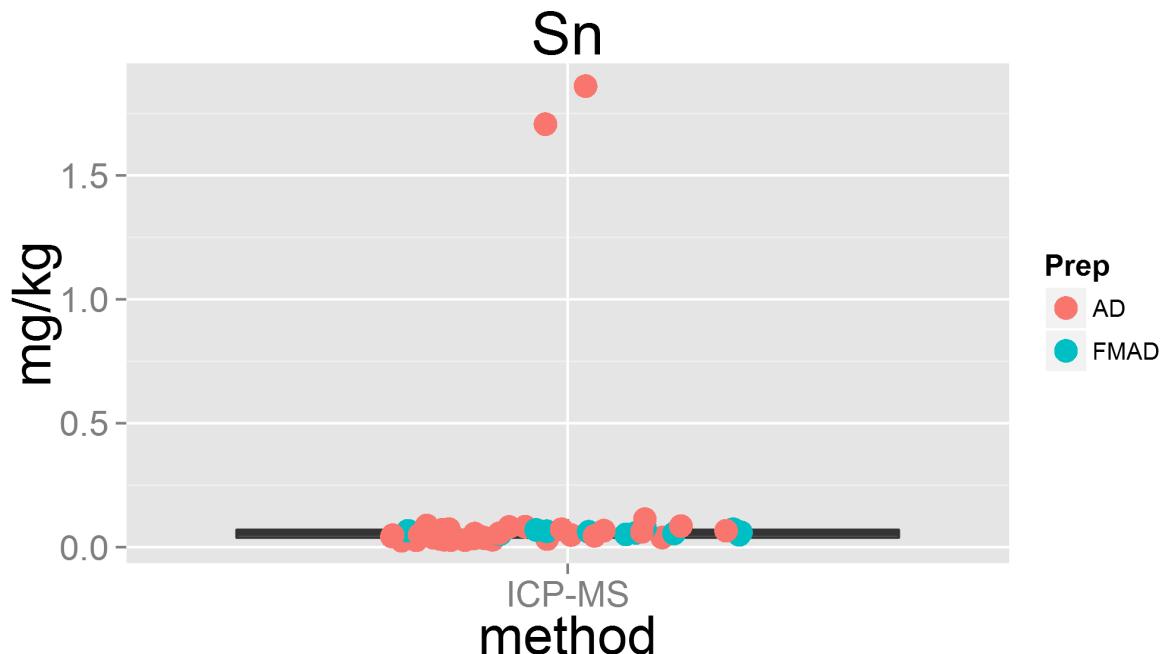
```
## Lab 14 was removed
```

```
## Lab 33 was removed
```



```
## Warning: Removed 2 rows containing missing values (geom_point).
```

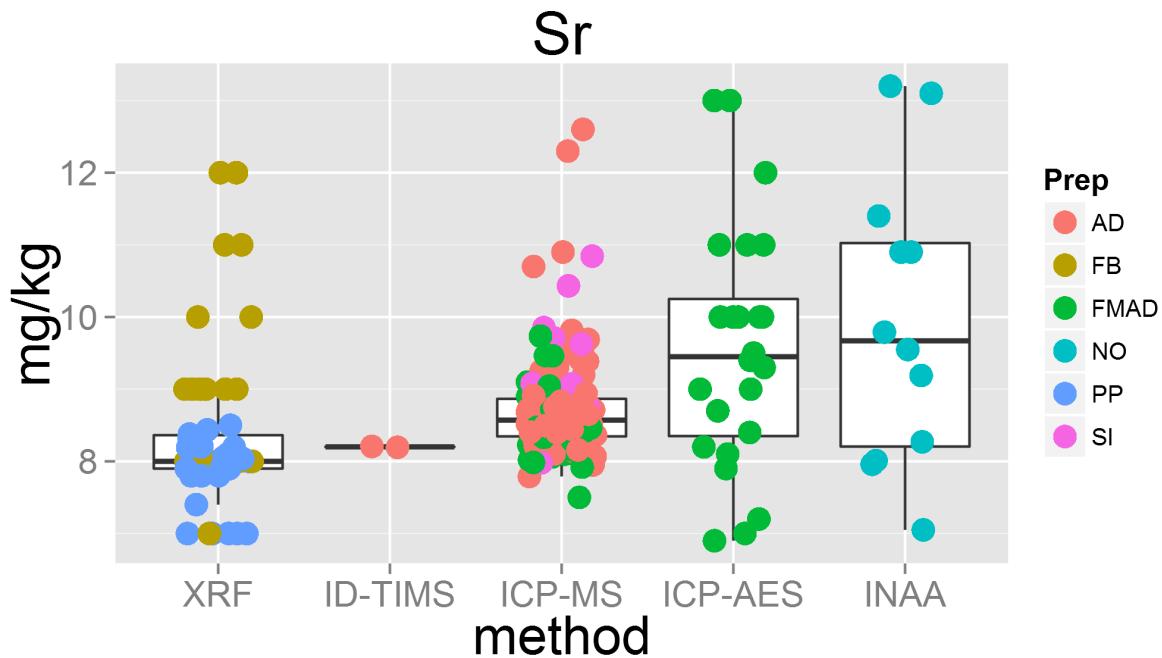
```
## Lab 0 was removed
```



```
## Warning: Removed 14 rows containing missing values (geom_point).
```

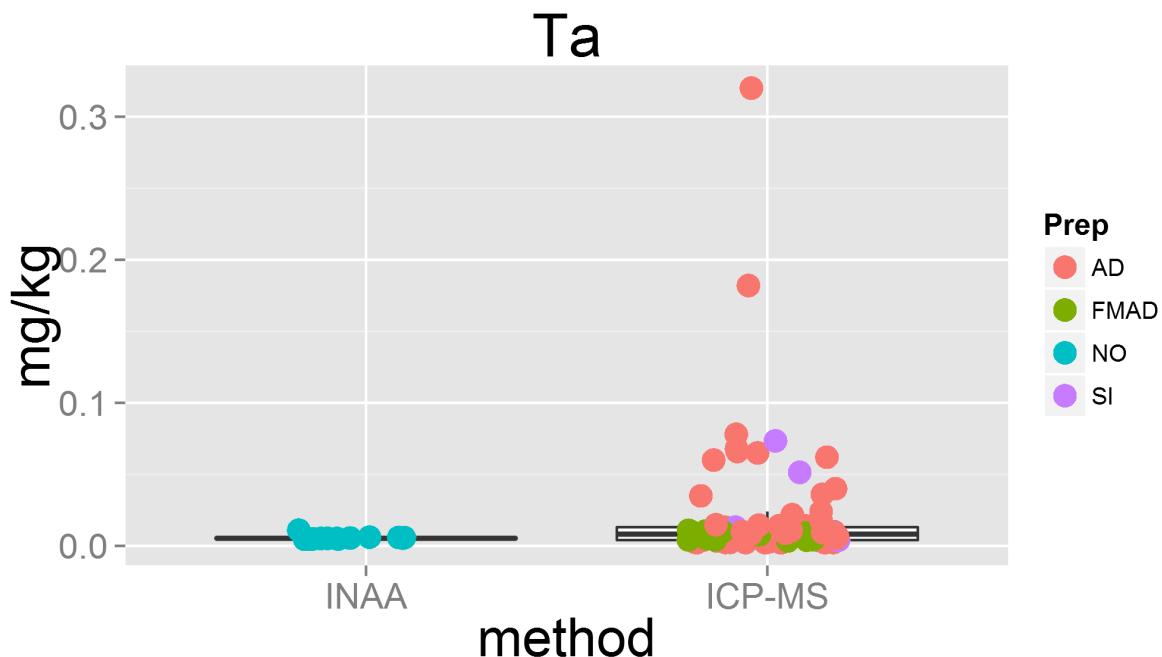
```
## Warning: Removed 12 rows containing missing values (geom_point).
```

```
## Lab 7 was removed  
## Lab 7 was removed
```



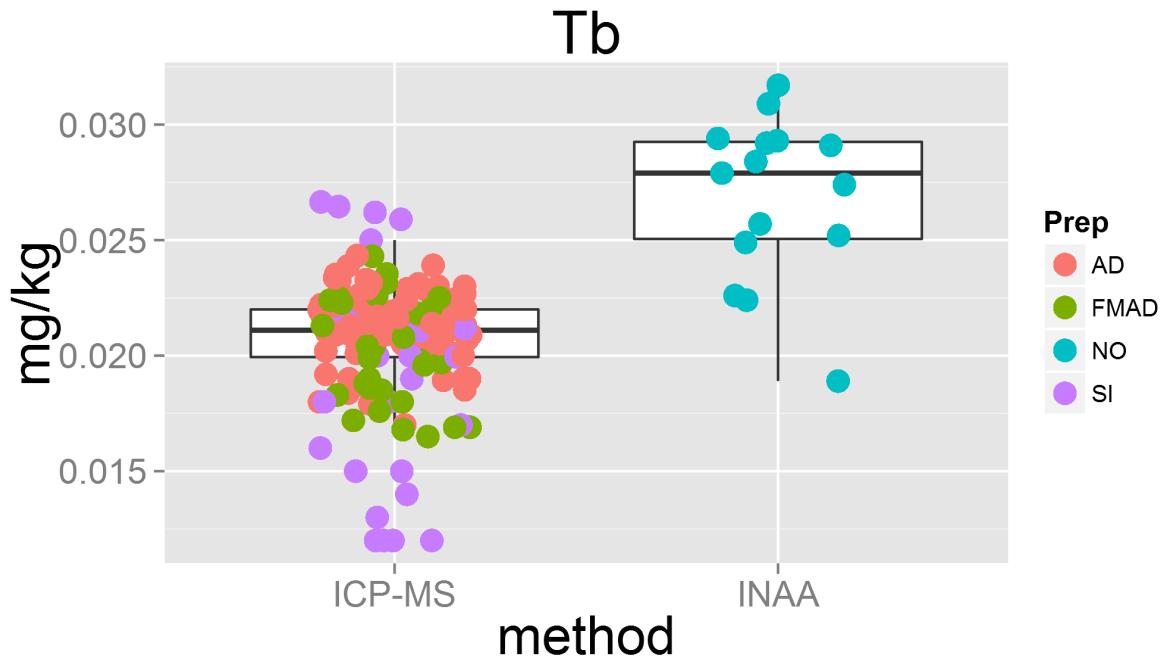
```
## Warning: Removed 1 rows containing missing values (geom_point).  
## Warning: Removed 13 rows containing missing values (geom_point).
```

```
## Lab 8 was removed
```



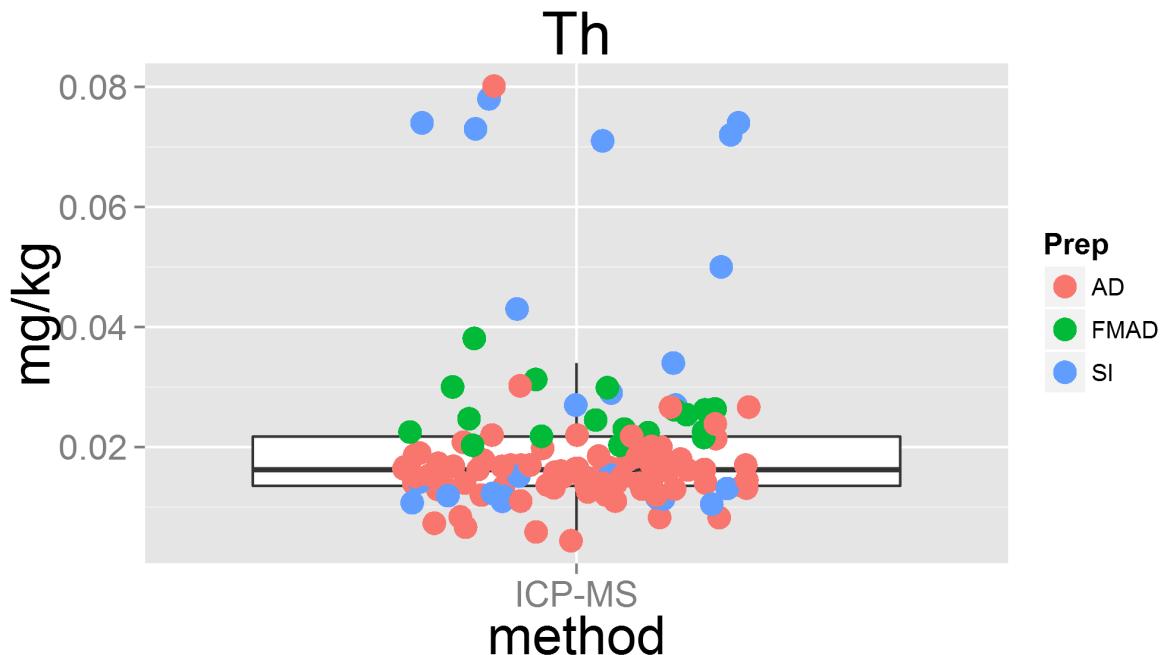
```
## Warning: Removed 15 rows containing missing values (geom_point).
```

```
## Lab 32 was removed  
## Lab 31 was removed
```



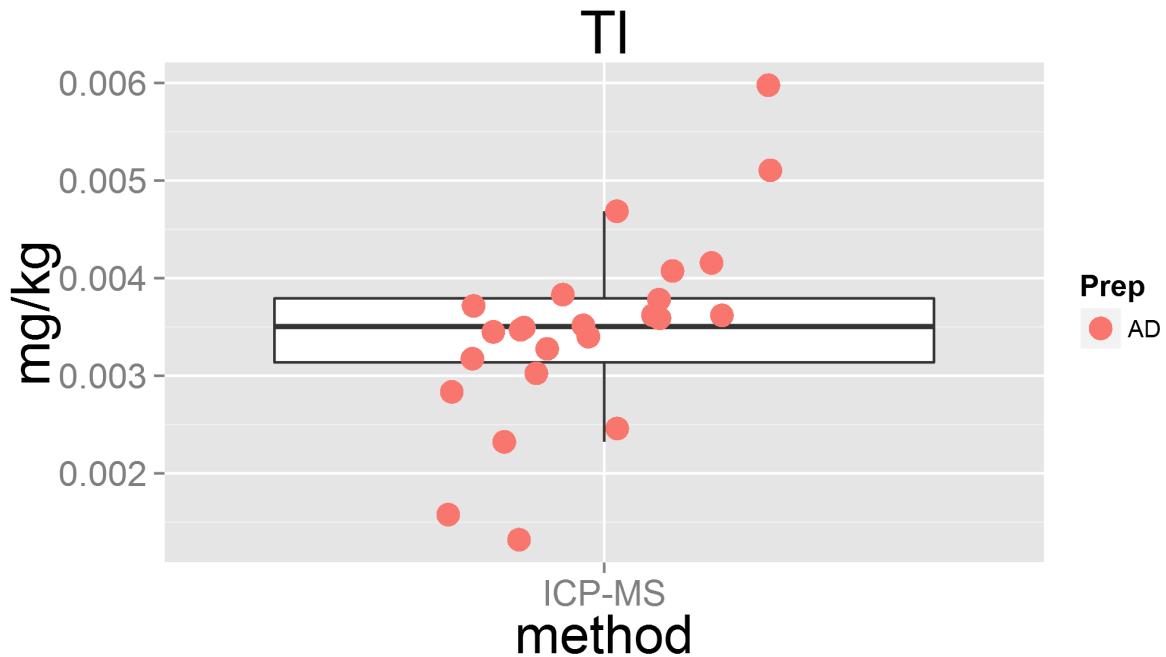
```
## Warning: Removed 10 rows containing missing values (geom_point).
```

```
## Lab 14 was removed
```



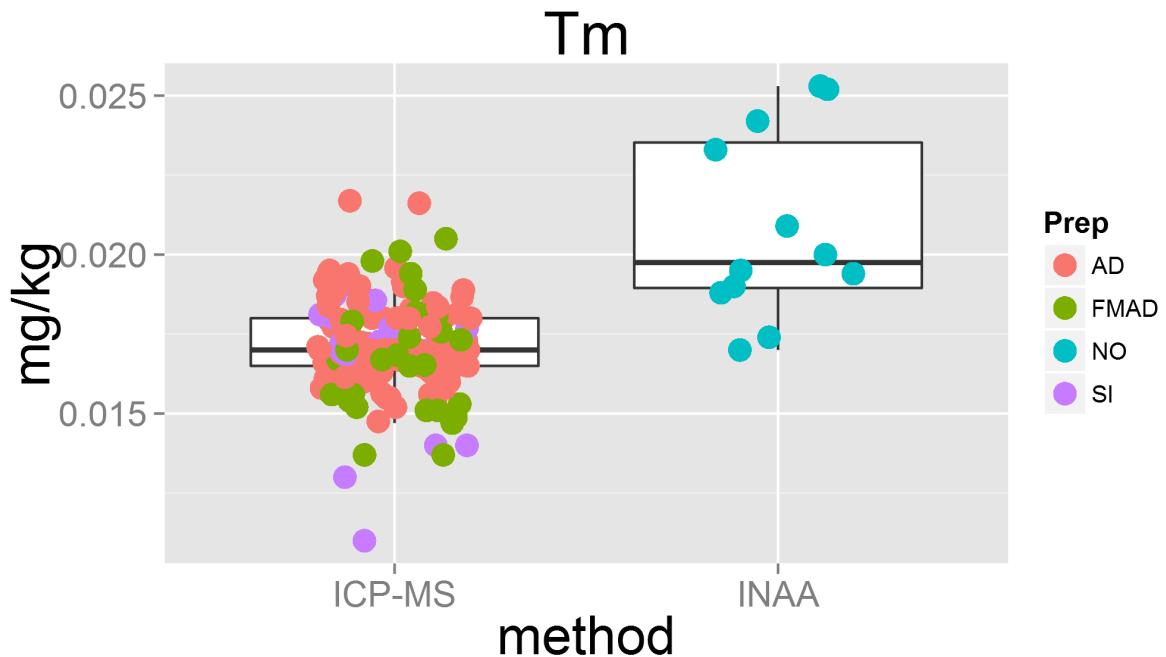
```
## Warning: Removed 4 rows containing missing values (geom_point).
```

```
## Lab 0 was removed
```



```
## Warning: Removed 9 rows containing missing values (geom_point).
```

```
## Lab 31 was removed
```

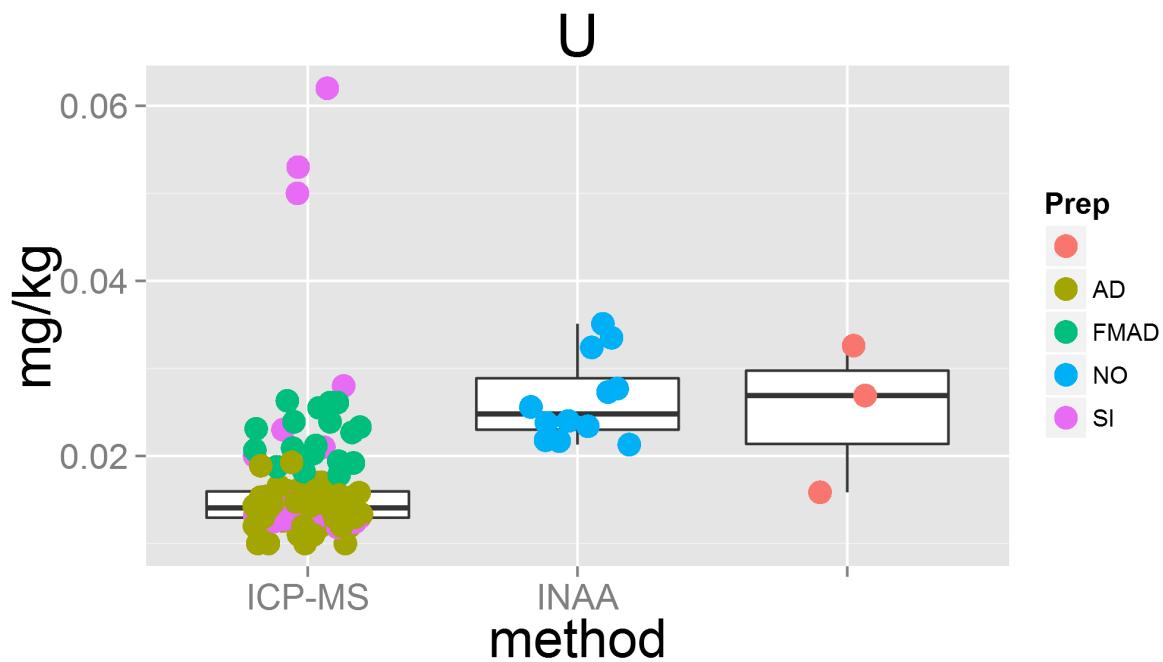


```

## Warning: Removed 18 rows containing missing values (geom_point).

## Lab 14 was removed
## Lab 29 was removed
## Lab 31 was removed

```

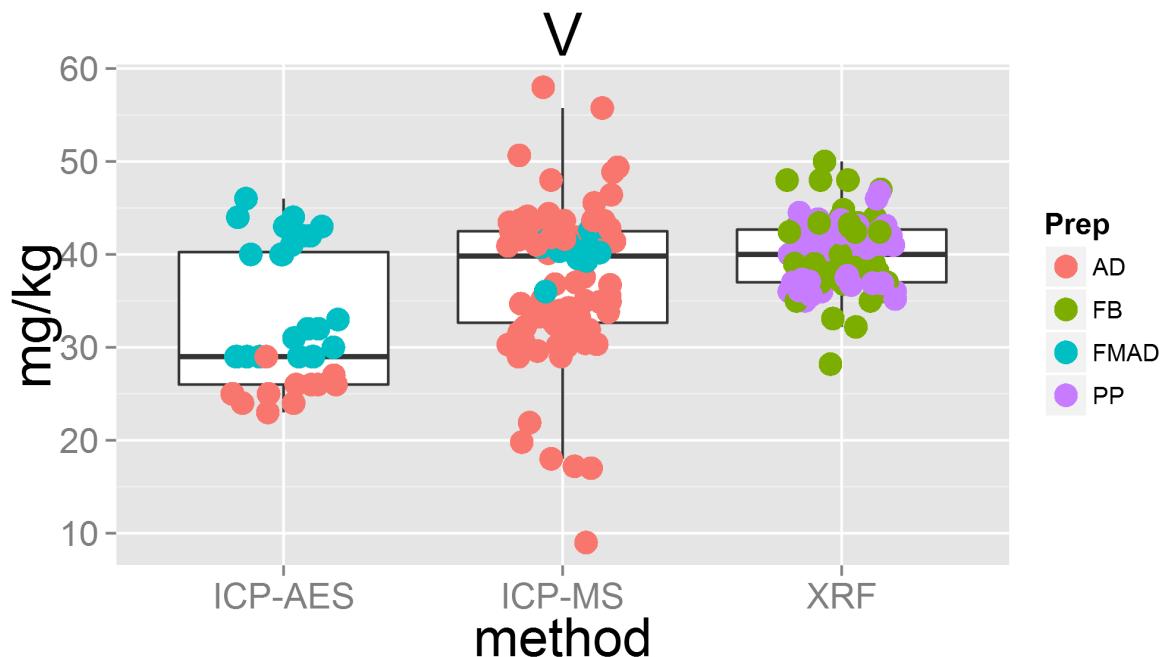


```

## Warning: Removed 4 rows containing missing values (geom_point).
## Warning: Removed 1 rows containing missing values (geom_point).

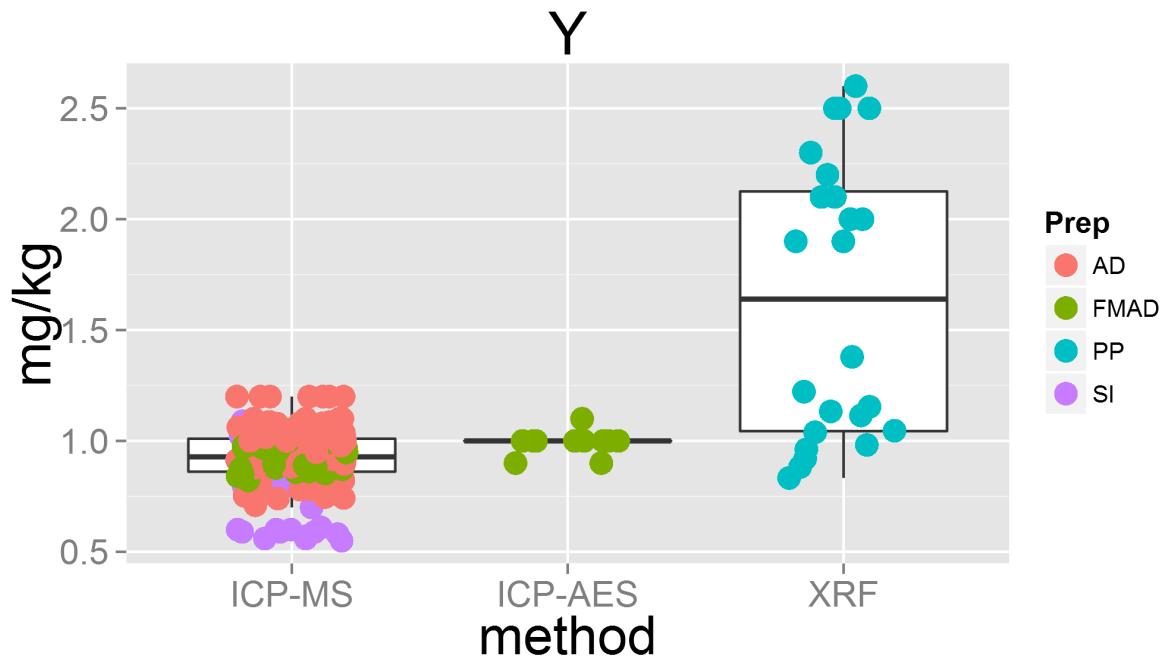
## Lab 31 was removed
## Lab 32 was removed

```



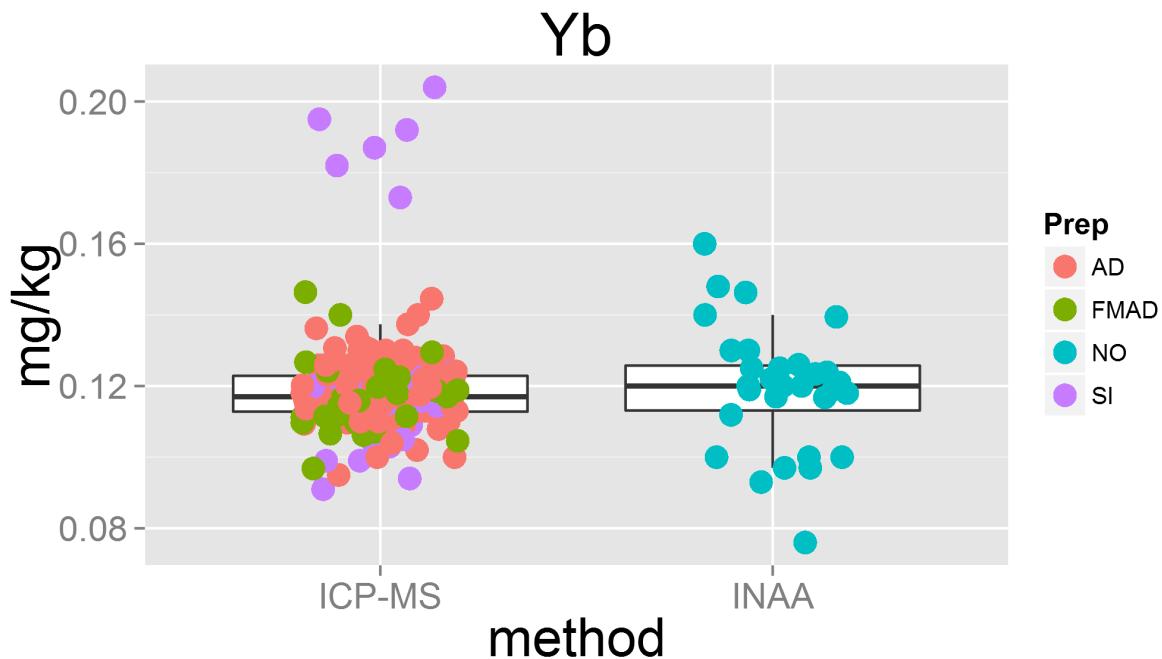
```
## Warning: Removed 12 rows containing missing values (geom_point).
## Warning: Removed 3 rows containing missing values (geom_point).
```

```
## Lab 26 was removed
## Lab 31 was removed
## Lab 32 was removed
```



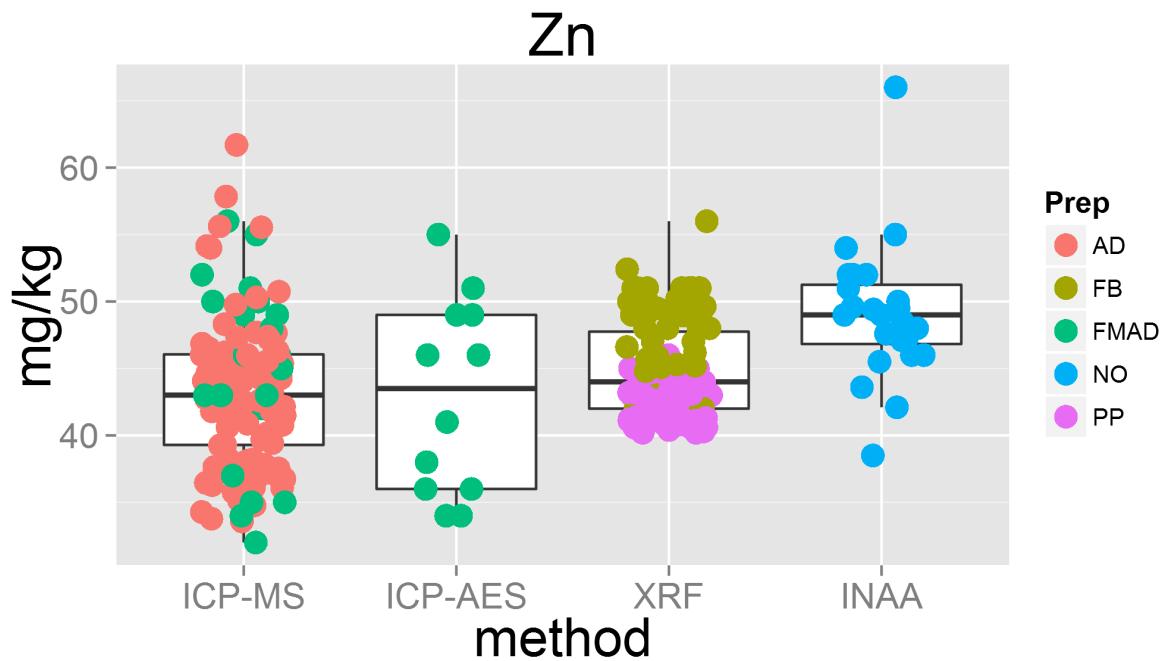
```
## Warning: Removed 14 rows containing missing values (geom_point).  
## Warning: Removed 5 rows containing missing values (geom_point).
```

```
## Lab 14 was removed
```



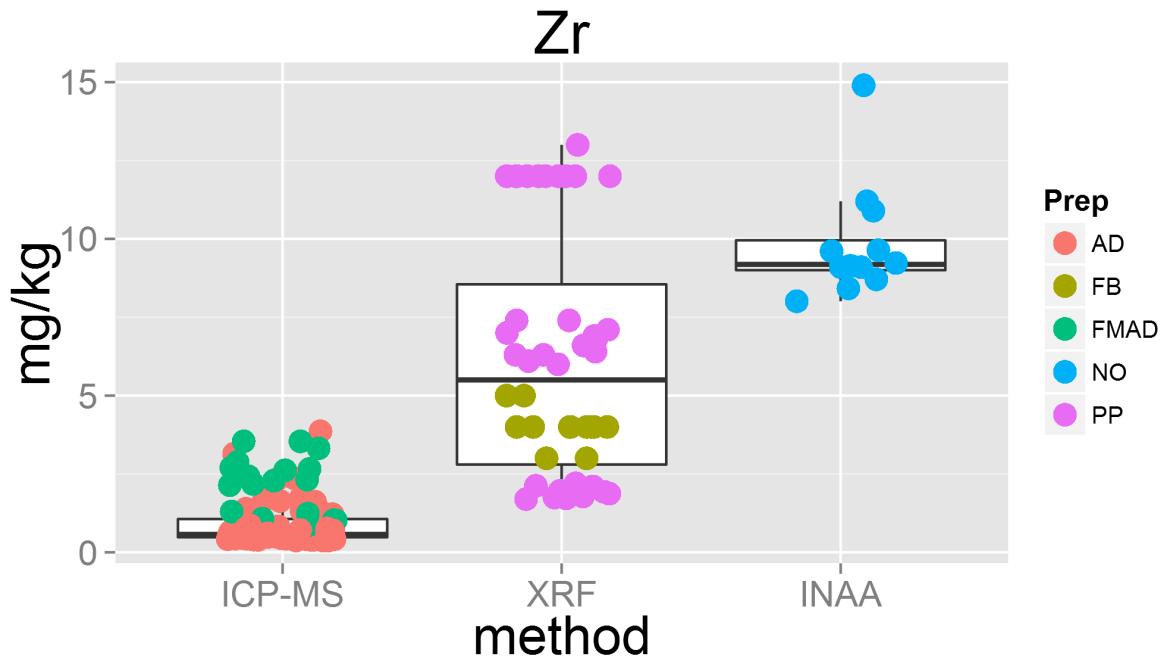
```
## Warning: Removed 2 rows containing missing values (geom_point).  
## Warning: Removed 2 rows containing missing values (geom_point).
```

```
## Lab 0 was removed
```



```
## Warning: Removed 16 rows containing missing values (geom_point).
## Warning: Removed 1 rows containing missing values (geom_point).
```

```
## Lab 10 was removed
## Lab 6 was removed
## Lab 26 was removed
## Lab 33 was removed
## Lab 31 was removed
## Lab 24 was removed
```



```
medianGOM.packet.after <- ddply(GOM, c("Lab", "Packet"), numcolwise(medianGOM)) # median Lab and Packet
## median over median of packets within lab
GOM.median.after <- ddply(medianGOM.packet.after, c("Lab"), numcolwise(medianGOM)) # creating a new table
GOM.median.after <- merge(GOM.median.after, OKUM.methods, by="Lab")
GOM.median.after.df <- data.frame(apply(GOM.median.after[2:56], 2, median, na.rm=TRUE)) # creating a new table
names(GOM.median.after.df) <- c("mass fraction")
meanGOM.packet <- ddply(GOM, c("Lab", "Packet"), numcolwise(meanGOM))
meanGOM.packet.after <- ddply(GOM, c("Lab", "Packet"), numcolwise(meanGOM)) # mean Lab and Packets after
## mean over mean of packets within lab
GOM.mean.after <- ddply(meanGOM.packet.after, c("Lab"), numcolwise(meanGOM)) # creating a new table of means
GOM.mean.after <- merge(GOM.mean.after, OKUM.methods, by="Lab")
GOM.mean.after.df <- data.frame(apply(GOM.mean.after[2:56], 2, mean, na.rm=TRUE)) # creating a new table
names(GOM.mean.after.df) <- c("mass fraction")
```

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

$$x_{ijk} = \mu + A_i + B_{ij} + \epsilon_{ijk}$$

where

x_{ijk} is the k th result of sample unit j reported from method/laboratory i ,

A_i is the error due to method/laboratory i ,

B_{ij} is the error due to the j th sample unit within method/laboratory i , ϵ_{ijk} is the measurement error.

The parameters to be estimated are the grand mean, the between-laboratory standard deviation s_L , the between-bottle standard deviation s_{bb} and the repeatability standard deviation s_r . The are related as follows

$$\begin{aligned}s_L &= \sqrt{\text{Var}(A_i)} \\s_{bb} &= \sqrt{\text{Var}(B_{ij})} \\s_r &= \sqrt{\text{Var}(\epsilon_{ijk})}\end{aligned}$$

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

$$\bar{x} = \frac{1}{\sum_{i=1}^p \sum_{j=1}^{b_i} n_{ij}} \sum_{i=1}^p \sum_{j=1}^{b_i} \sum_{k=1}^{n_{ij}} x_{ijk}$$

where p denotes the number of laboratories, b_i the number of bottles used by method/laboratory i , and n_{ij} is the number of replicates measured on bottle ij . The variances are computed as follows

$$\begin{aligned}\text{Var}(\epsilon_{ijk}) &= MS_{within} = s_r^2 \\ \text{Var}(B_{ij}) &= \frac{MS_{B \subset A} - MS_{within}}{n_0} = s_{bb}^2 \\ \text{Var}(A_i) &= \frac{MS_{among} - n'_0 \text{Var}(B_{ij}) - \text{Var}(\epsilon_{ijk})}{(nb)_0} = s_L^2\end{aligned}$$

where

$$\begin{aligned}MS_{among} &= \frac{\sum_{i=1}^p n_i (\bar{x}_A - \bar{x})^2}{p-1} \\MS_{B \subset A} &= \frac{\sum_{i=1}^p \sum_{j=1}^{b_i} n_{ij} (\bar{x}_B - \bar{x}_A)^2}{\sum_{i=1}^p b_i - p} \\MS_{within} &= \frac{\sum_{i=1}^p \sum_{j=1}^{b_i} \sum_{k=1}^{n_{ij}} (x_{ijk} - \bar{x}_B)^2}{\sum_{i=1}^p \sum_{j=1}^{b_i} n_{ij} - \sum_{i=1}^p b_i}\end{aligned}$$

and

$$\begin{aligned}n'_0 &= \frac{\sum_{i=1}^p \left(\frac{\sum_{j=1}^{b_i} n_{ij}^2}{\sum_{j=1}^{b_i} n_{ij}} \right) - \sum_{i=1}^p \sum_{j=1}^{b_i} n_{ij}^2}{p-1} \\n_0 &= \frac{\sum_{i=1}^p \sum_{j=1}^{b_i} n_{ij} - \sum_{i=1}^p \left(\frac{\sum_{j=1}^{b_i} n_{ij}^2}{\sum_{j=1}^{b_i} n_{ij}} \right)}{\sum_{i=1}^p b_i - p} \\(nb)_0 &= \frac{\sum_{i=1}^p \sum_{j=1}^{b_i} n_{ij} - \frac{\sum_{i=1}^p (\sum_{j=1}^{b_i} n_{ij})^2}{\sum_{i=1}^p \sum_{j=1}^{b_i} n_{ij}}}{p-1}\end{aligned}$$

solutions of the above equations in R

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 extracted with package ape (varcomp {ape})

$$\begin{aligned}s_L^2 &= sL2 <- varcomp(GOM.lme, FALSE)[[1]] \# between-laboratory variance \\s_{bb}^2 &= sbb2 <- varcomp(GOM.lme, FALSE)[[2]] \# between bottle variance \\s_r^2 &= sr2 <- varcomp(GOM.lme, FALSE)[[3]] \# repeatability standard deviation\end{aligned}$$

The characterisation uncertainty u_{char} is calculated

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

which is equivalent to

$$u_1 = \sqrt{\frac{s_L^2}{p} - \frac{s_r^2}{np} - \frac{s_{bb}^2}{npr}}$$

This approach is not completely 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 s_{bb}^2 component.

```
for (m in col.names) {
  measurand.name <- m
  switch(
    reformat,
    GAS = rm1 <- 2,
    MUH = rm1 <- 1,
    OKUM = rm1 <- 0
  )
  if(rm1 > 0)
    {measurand <- measurand.name %p% '.' %p% rm1
  } else
  {
    measurand <- measurand.name
  }
  MorT <- grep(measurand.name, colnames(GOM), fixed=TRUE) # finding the position of the measurand.name
  ifelse(MorT[1]< 21, MorT <- 'M', MorT<-'T') # testing if measurand is a major or trace element/compound
  ifelse(MorT == "T", unit <- 'mg/kg', unit <- 'g/100g') # testing which unit is needed
  ## calculating method parameters
# '%p%' <- function(x, y) {as.character(paste(x, y, sep = ""))}
mean <- mean(tapply(GOM[[measurand]], GOM$Lab, mean, na.rm=TRUE), na.rm=TRUE)
mean.before <- mean(GOM.mean[[measurand]], na.rm=TRUE)
median.before <- median(GOM.median[[measurand]], na.rm=TRUE)
median.after <- median(GOM.median.after[[measurand]], na.rm=TRUE) # median of the individual measurand
prep <- 'Prep.'
method <- 'Method.'
anal.prep <- prep %p% measurand.name
anal.method <- method %% measurand.name
anal <- GOM.median.after[[measurand]]
anal.prep <- GOM.median.after[[anal.prep]]
anal.method <- GOM.median.after[[anal.method]]
analyte <- data.frame(GOM.median.after$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))

meanGOM.packet.after$Lab <- as.factor(meanGOM.packet$Lab) # using only the median of the 3 packages per
meanGOM.packet.after$Packet <- as.factor(meanGOM.packet$Packet)
anal <- meanGOM.packet.after[[measurand]]
DF.lme <- data.frame(meanGOM.packet.after$Lab, meanGOM.packet.after$Packet, meanGOM.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
```

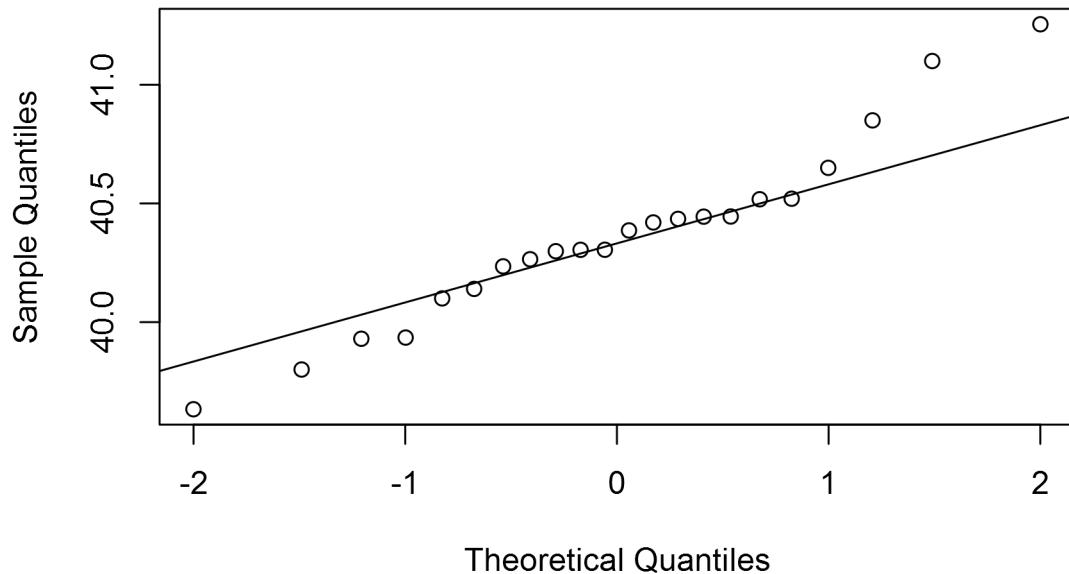
```

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 observations
p <- length(unique(DF.lme$Lab)) # haven't found a better way how to extract the number of labs (number
r <- length(unique(DF.lme$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$fixDF,"varFixFact") # gives the same results as u1, amazing!
# plot(DF.lme)
analyte.noPP <- subset(analyte, analyte$anal.prep!="PP") # Removing all PP preparations for comparison
median.after.noPP <- median(analyte.noPP$anal)
outlier <- ifelse(outlier=="0", "X", outlier)
print(measurand)
qqnorm(GOM.median.after[[measurand]])
qqline(GOM.median.after[[measurand]])
reference.line <- median.after
u.Ulim <- median.after + u2.a*t.value
l.Ulim <- median.after - u2.a*t.value
bymethod <- ggplot(bymethod.n, aes(x=anal.method, y=median))+geom_point(size=4)+geom_errorbar(aes(ymin=r
plot.lab <- plot_lab(measurand, MorT, horw = FALSE, u = TRUE)
# grid.arrange(bymethod, plot.lab, ncol=2)
print(plot.lab)
# print(bymethod.n)
outlier.type.name <- measurand.name %p% ".outlier.type" # defining if outlier is selected ("Y" or NA)
outlier.dist.type <- measurand.name %p% ".dist.type" # defining if outlier is based on Y = Youden plot,
out.measurand <- data.frame(MUH.outlier[[measurand.name]], MUH.outlier[[outlier.type.name]], MUH.outlier[[outlier.dist.type.name]])
# out.measurand <- cbind(MUH.outlier[[measurand.name]], MUH.outlier[[outlier.type.name]])
names(out.measurand) <- c("outlier.lab", "outlier.type", "outlier.dist" )
property.value.dist.type <- out.measurand[1,3]
property.value <- ifelse(property.value.dist.type == "median", median.after, mean)
df <- data.frame(Sys.Date(), refmat, measurand.name, signif(mean.before, 4), signif(mean,4), signif(med
write.table(df, "df1.txt", row.names=FALSE, append=TRUE, col.names=FALSE)
}

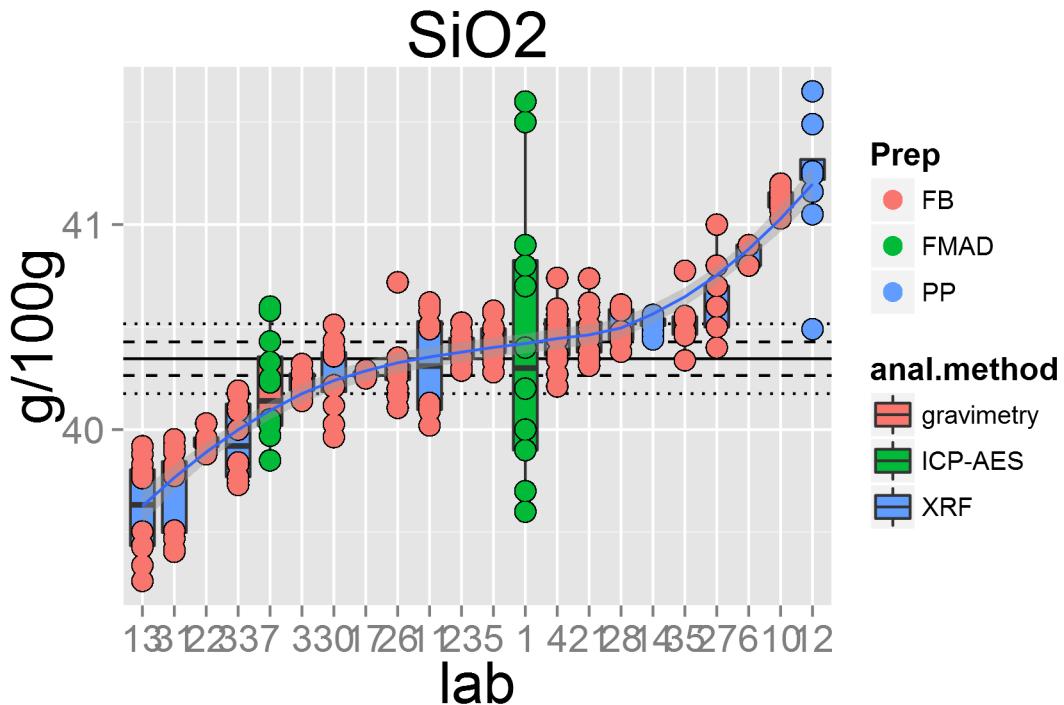
## [1] "SiO2.1"

```

Normal Q-Q Plot

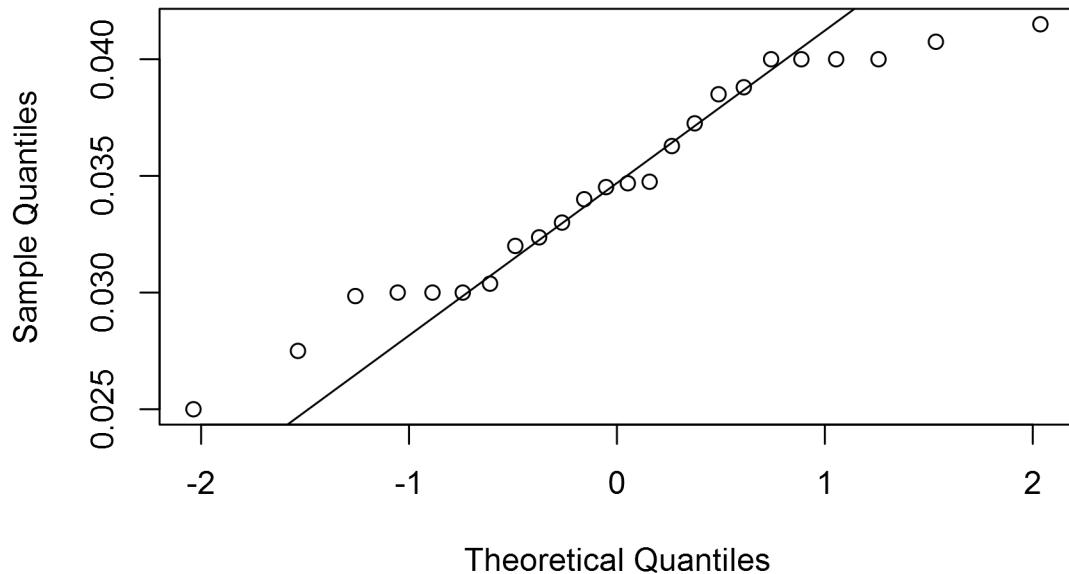


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to c
```

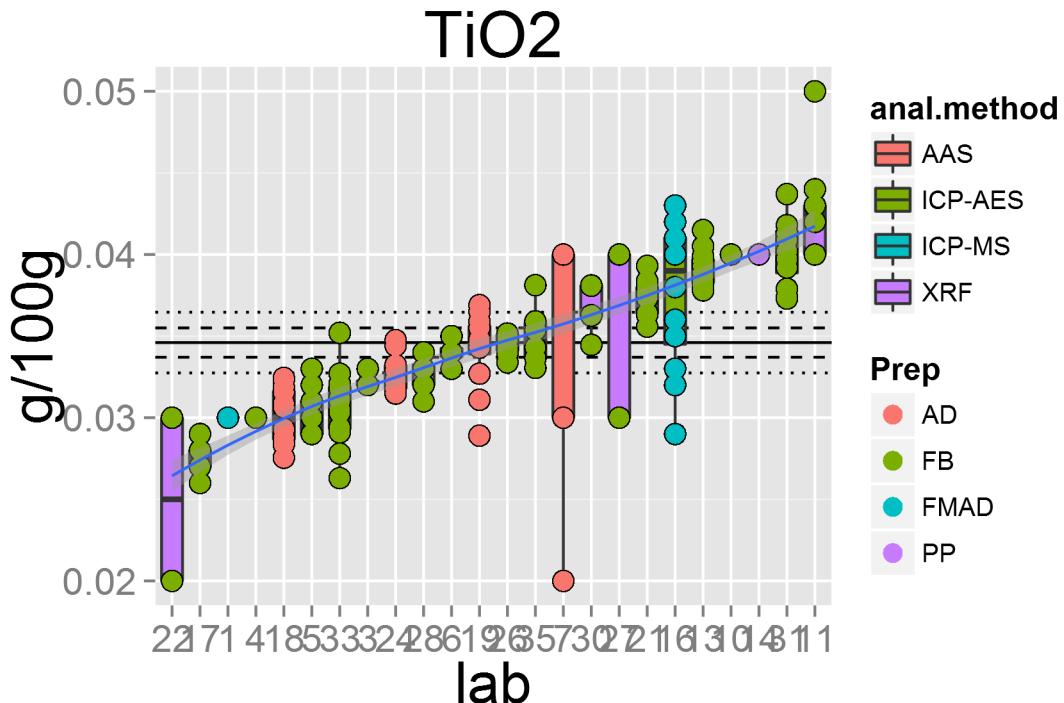


```
## [1] "Ti02.1"
```

Normal Q-Q Plot

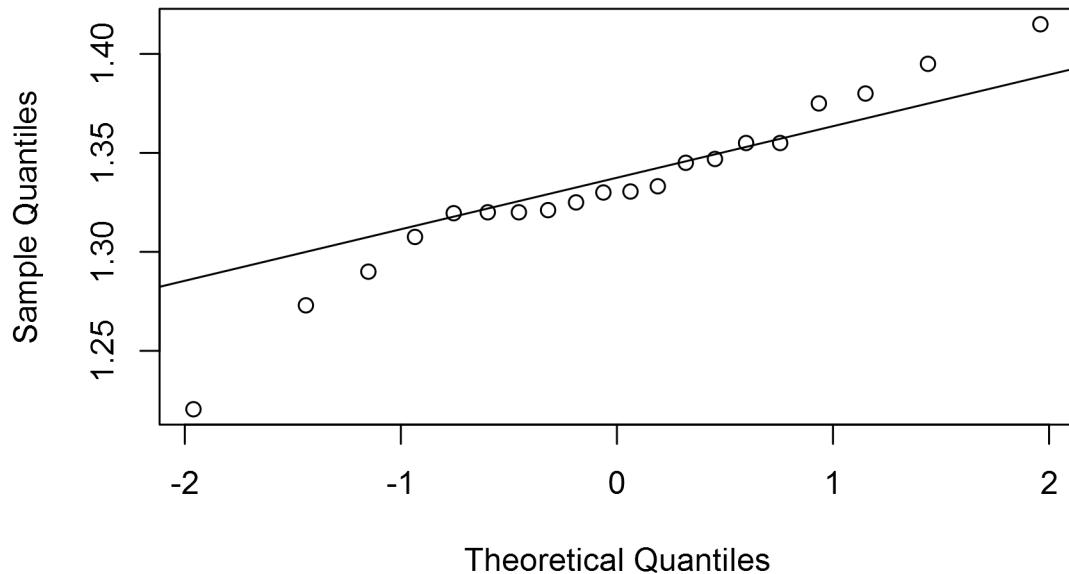


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to c
```

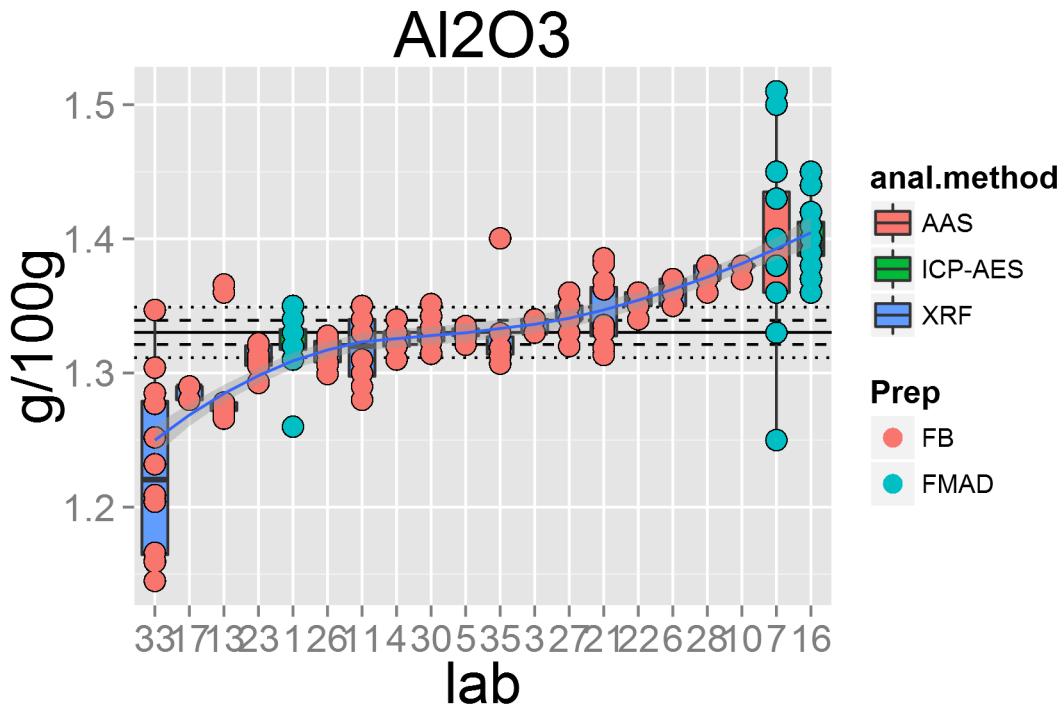


```
## [1] "Al203.1"
```

Normal Q-Q Plot

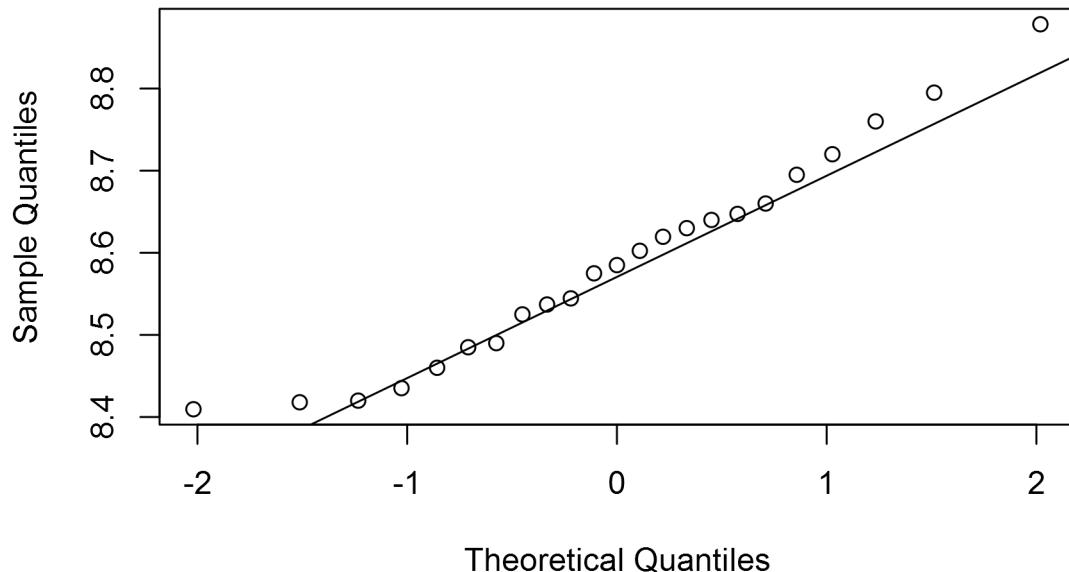


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to c
```

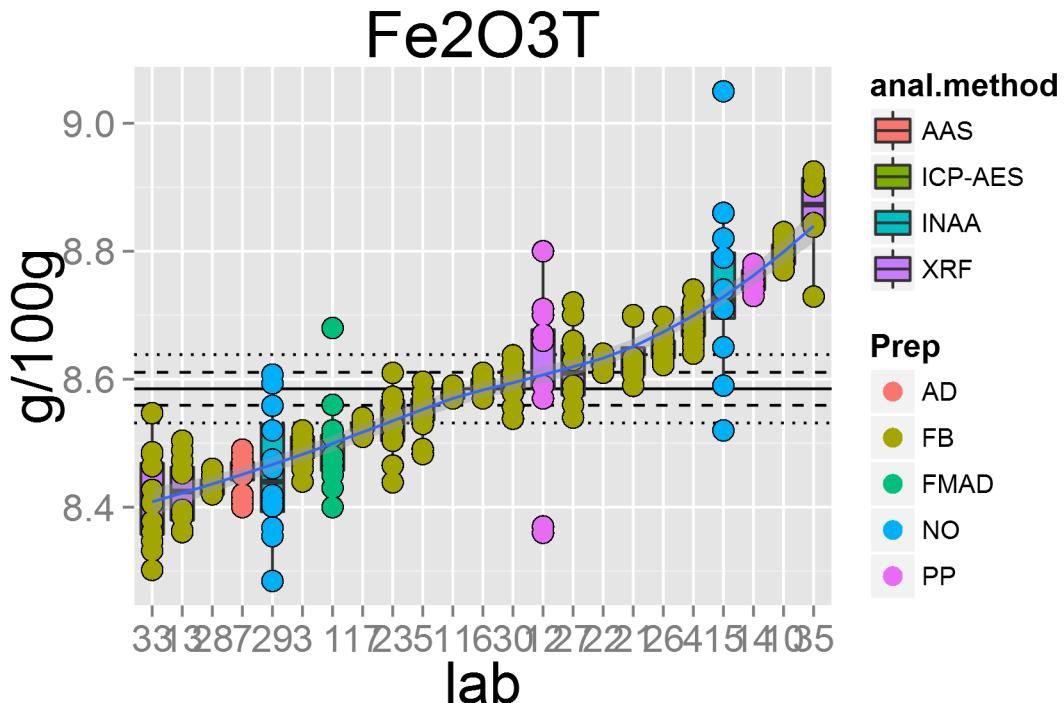


```
## [1] "Fe203T.1"
```

Normal Q-Q Plot

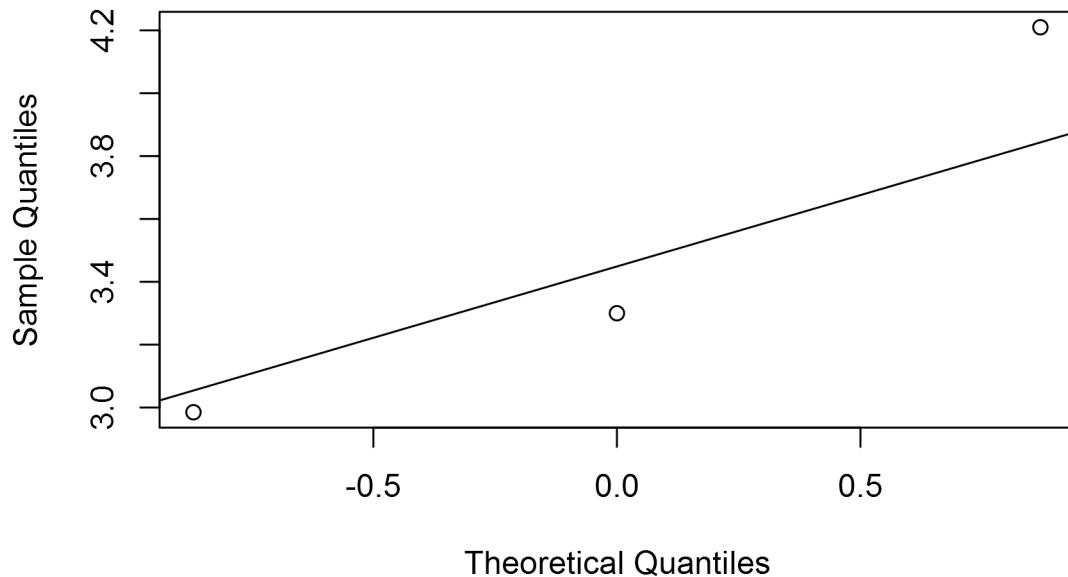


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```

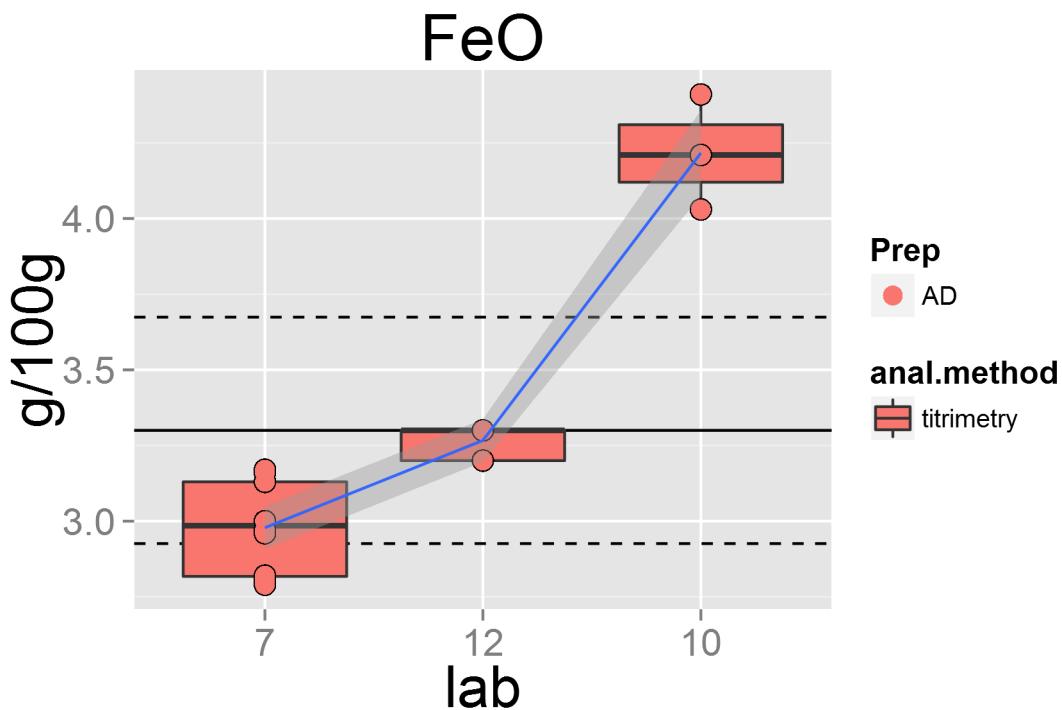


```
## [1] "Fe0.1"
```

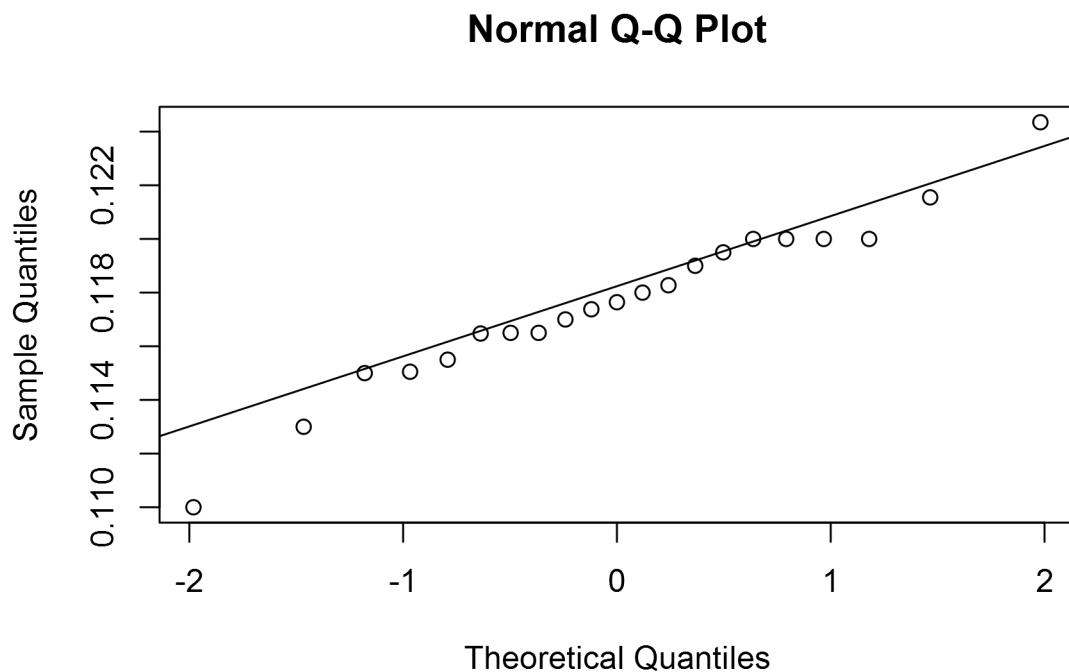
Normal Q-Q Plot



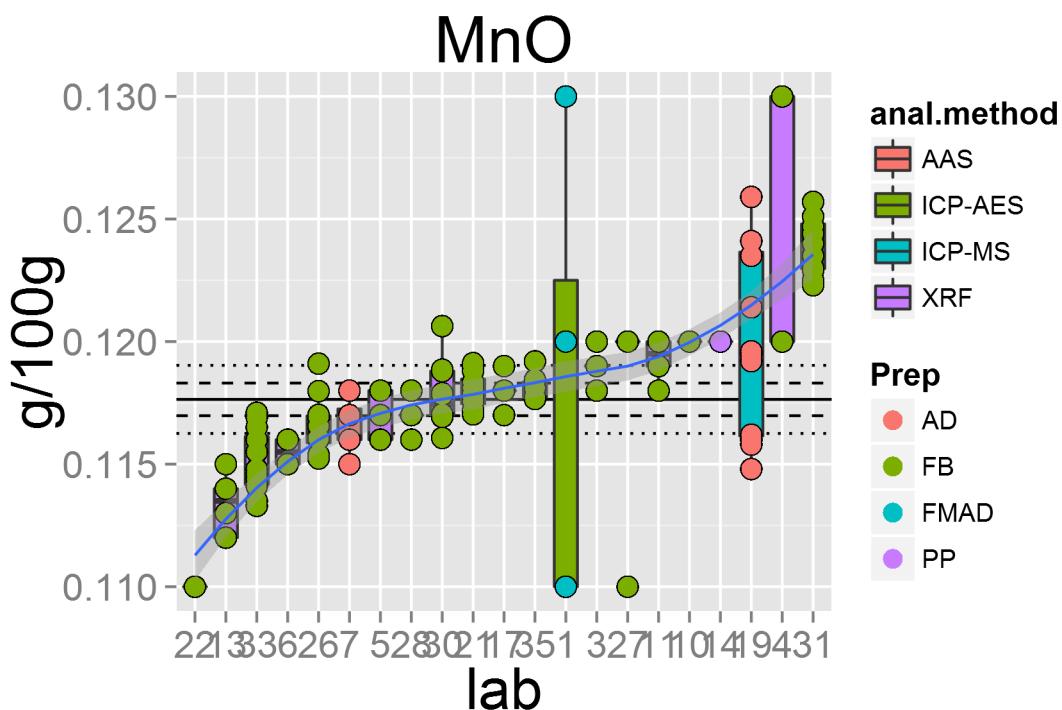
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to c
## Warning: pseudoinverse used at 0.99
## Warning: neighborhood radius 1.01
## Warning: reciprocal condition number  0
## Warning: There are other near singularities as well. 4.0401
## Warning: pseudoinverse used at 0.99
## Warning: neighborhood radius 1.01
## Warning: reciprocal condition number  0
## Warning: There are other near singularities as well. 4.0401
```



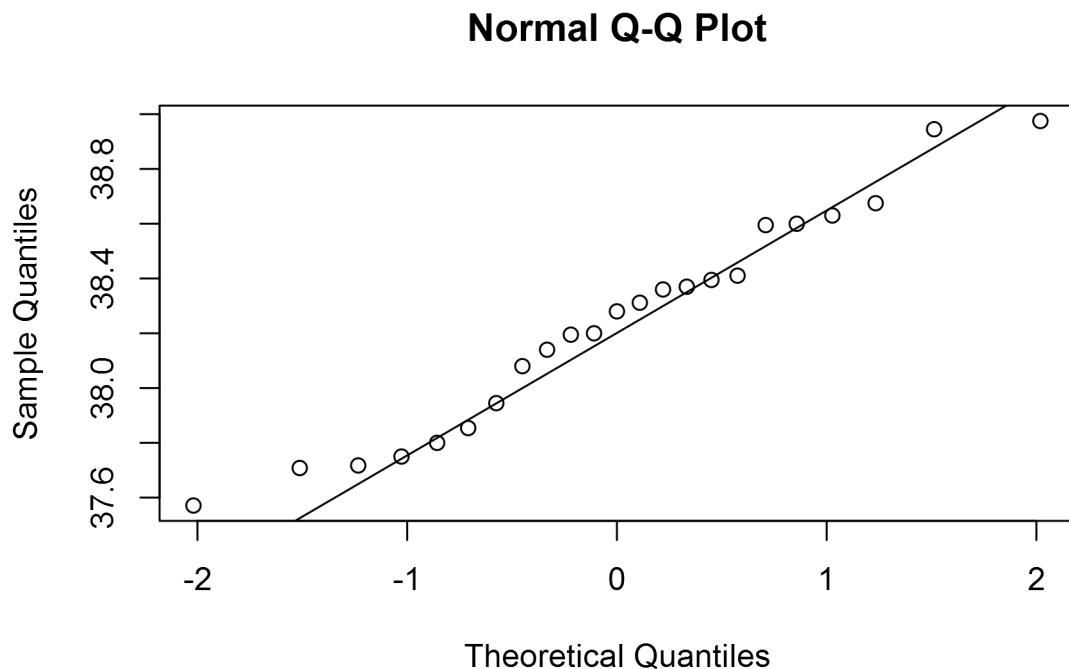
```
## [1] "Mn0.1"
```



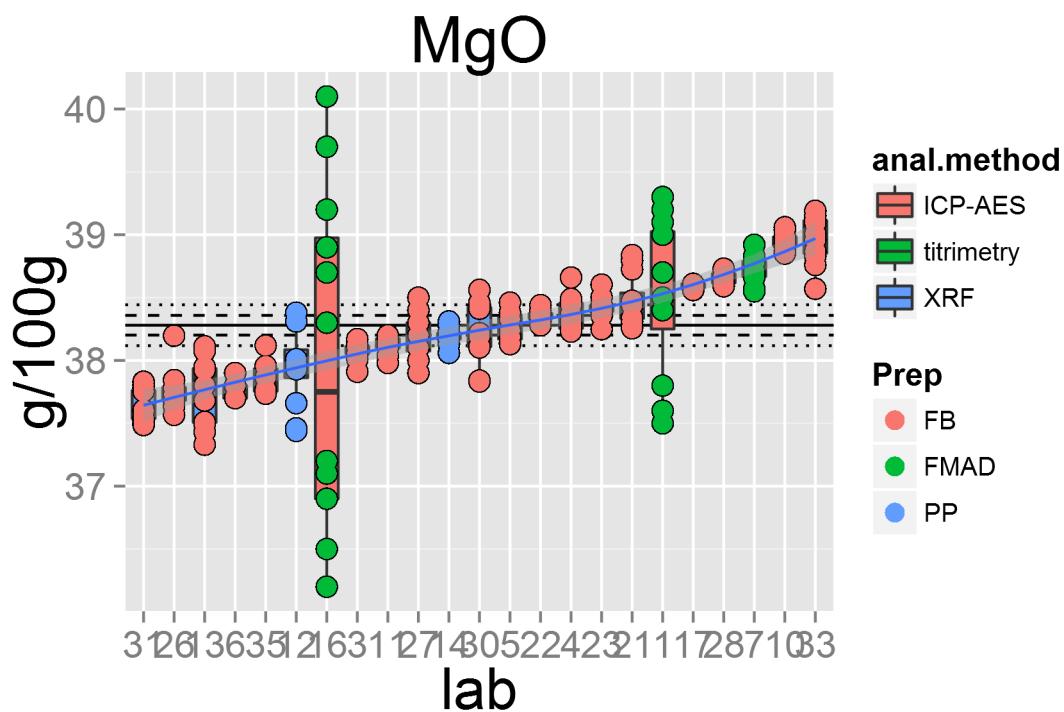
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



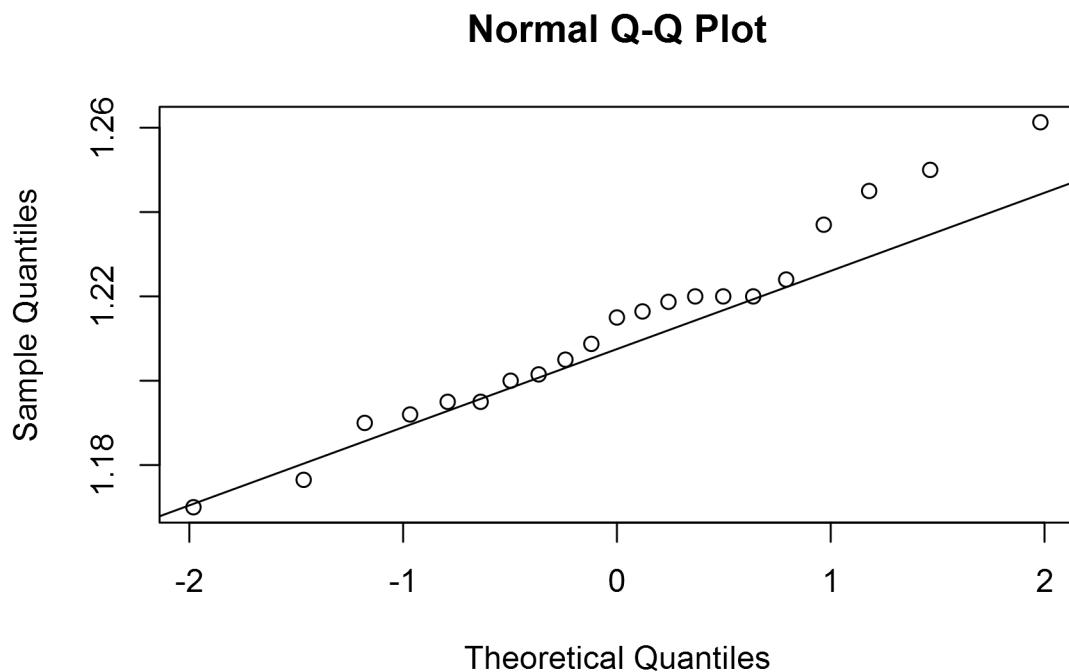
```
## [1] "Mg0.1"
```



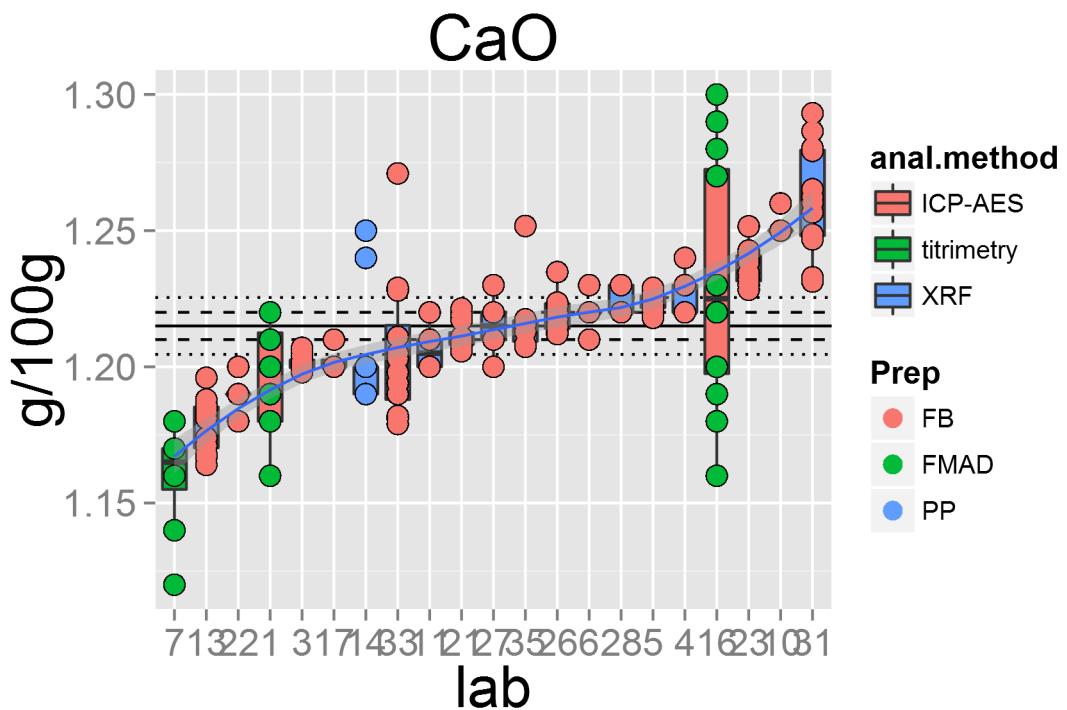
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



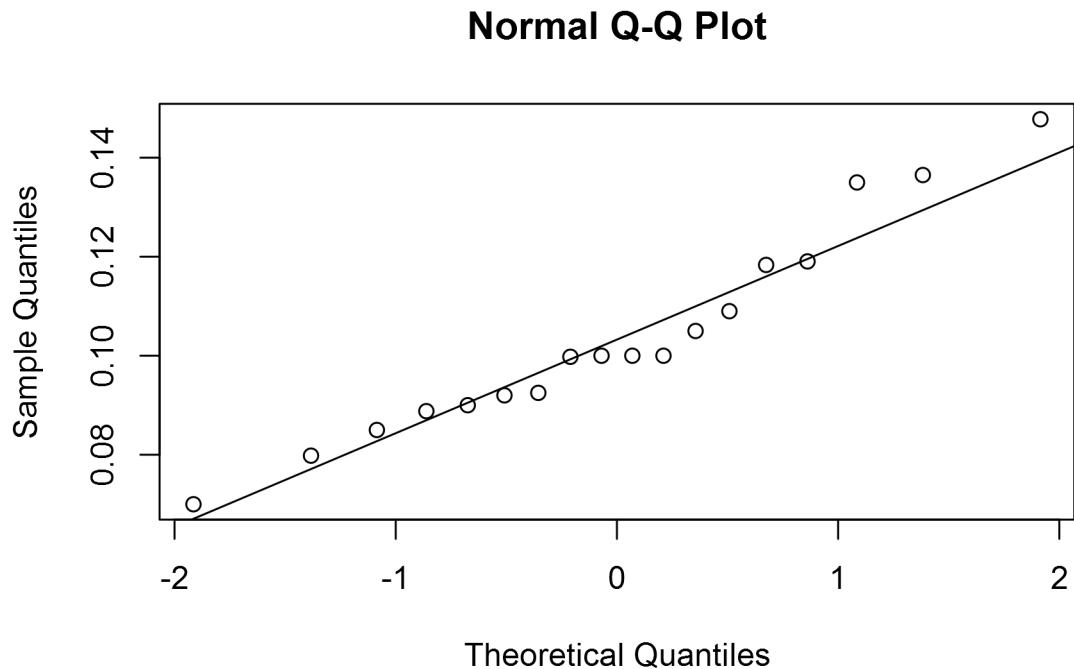
```
## [1] "Ca0.1"
```



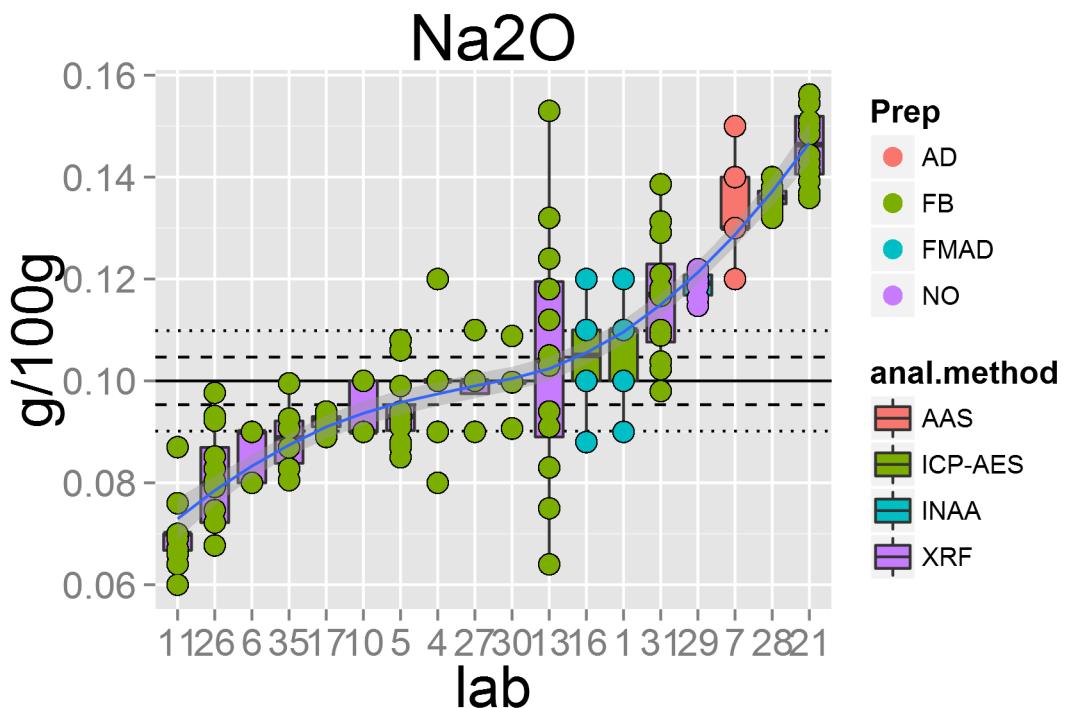
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



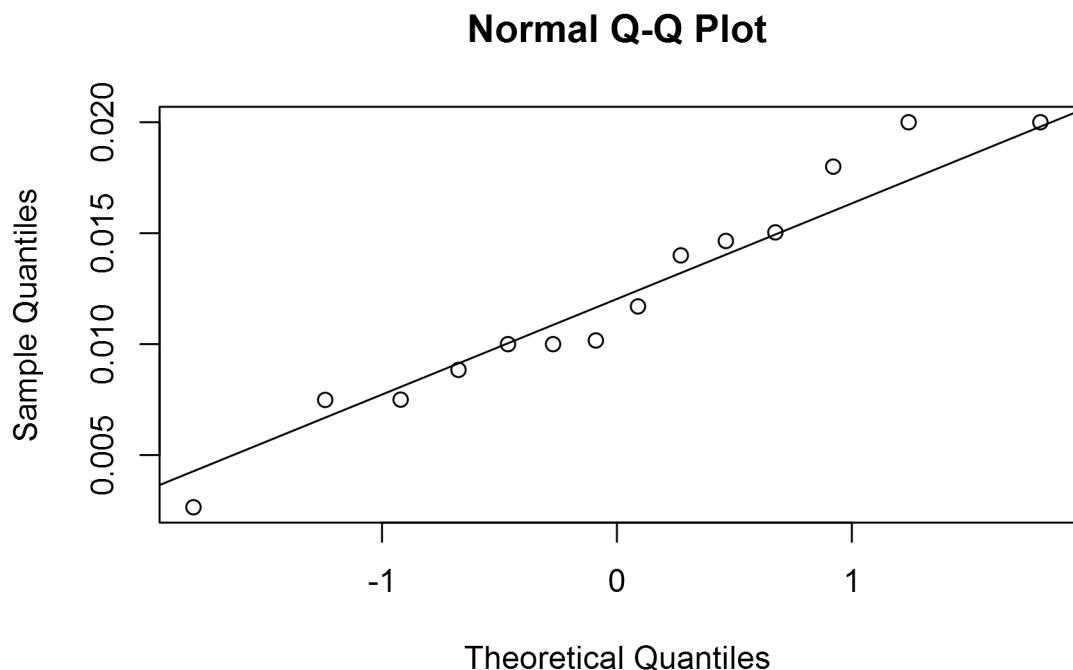
```
## [1] "Na20.1"
```



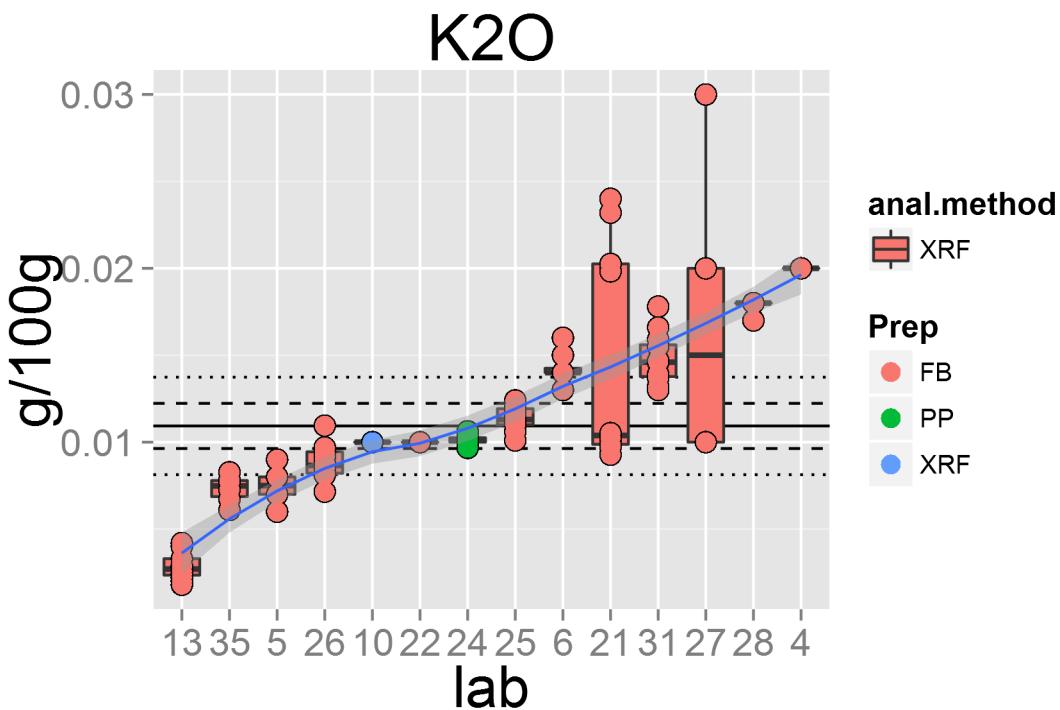
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



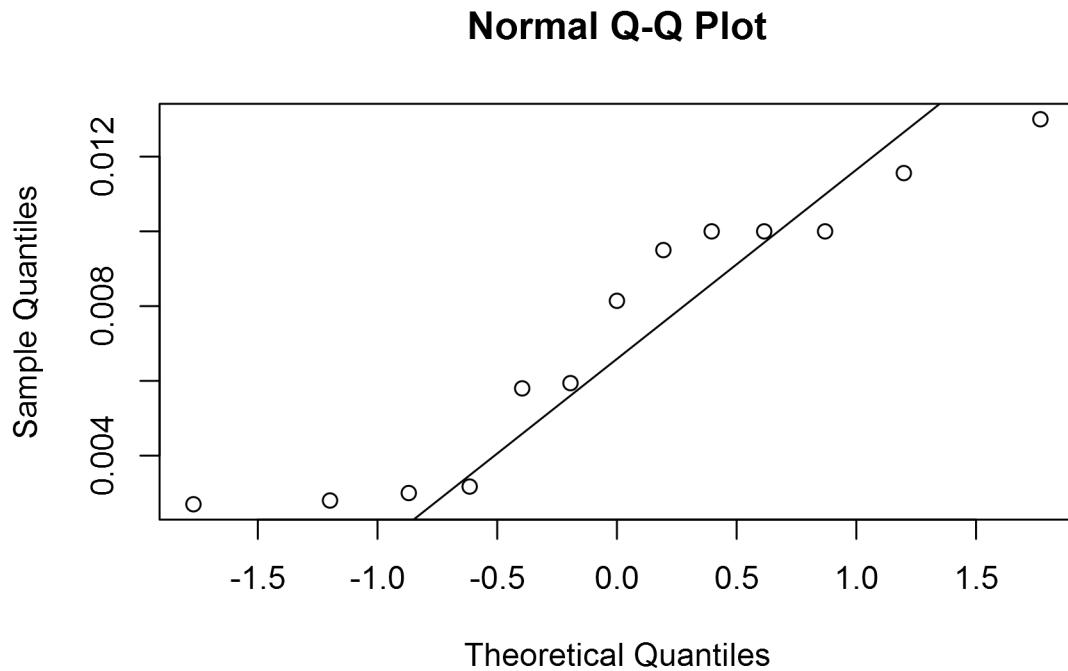
```
## [1] "K20.1"
```



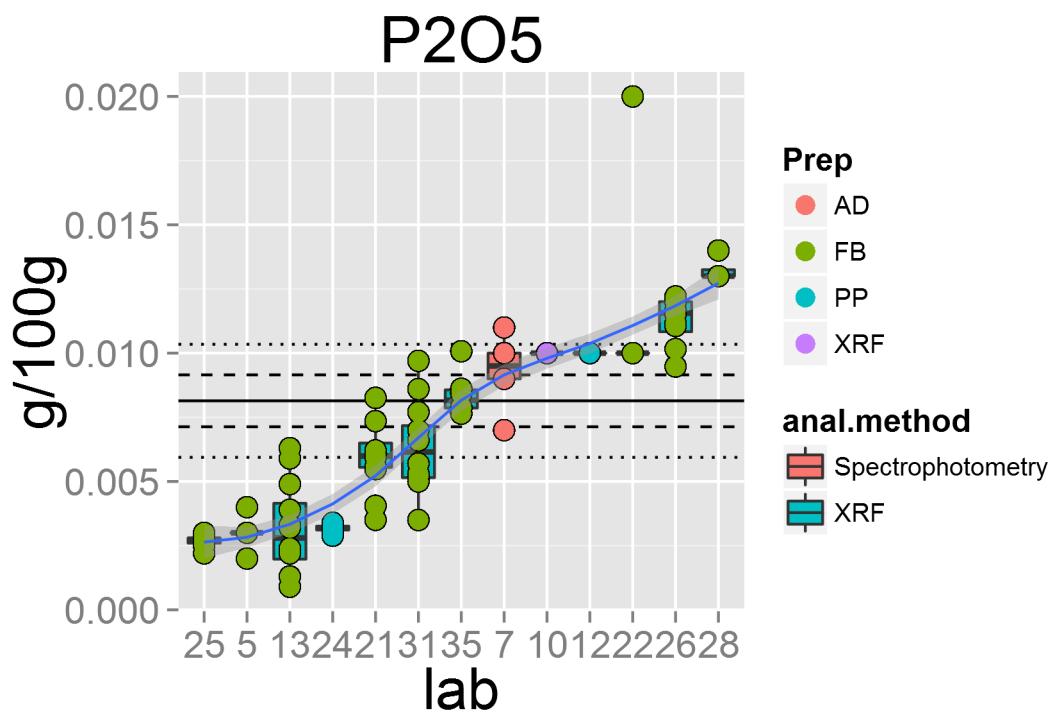
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



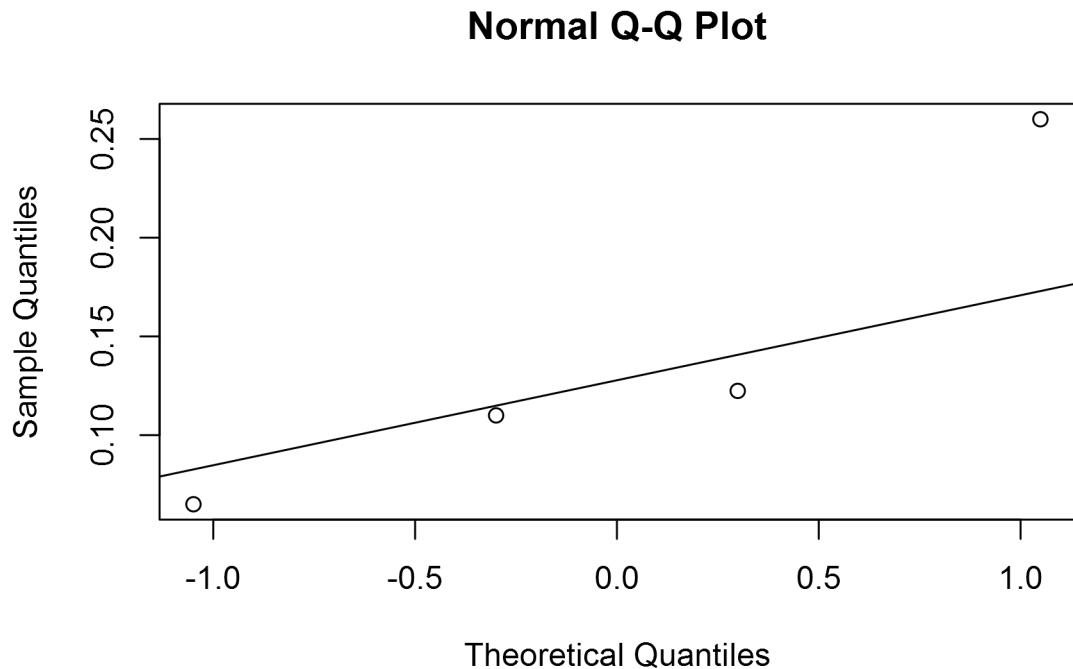
```
## [1] "P205.1"
```



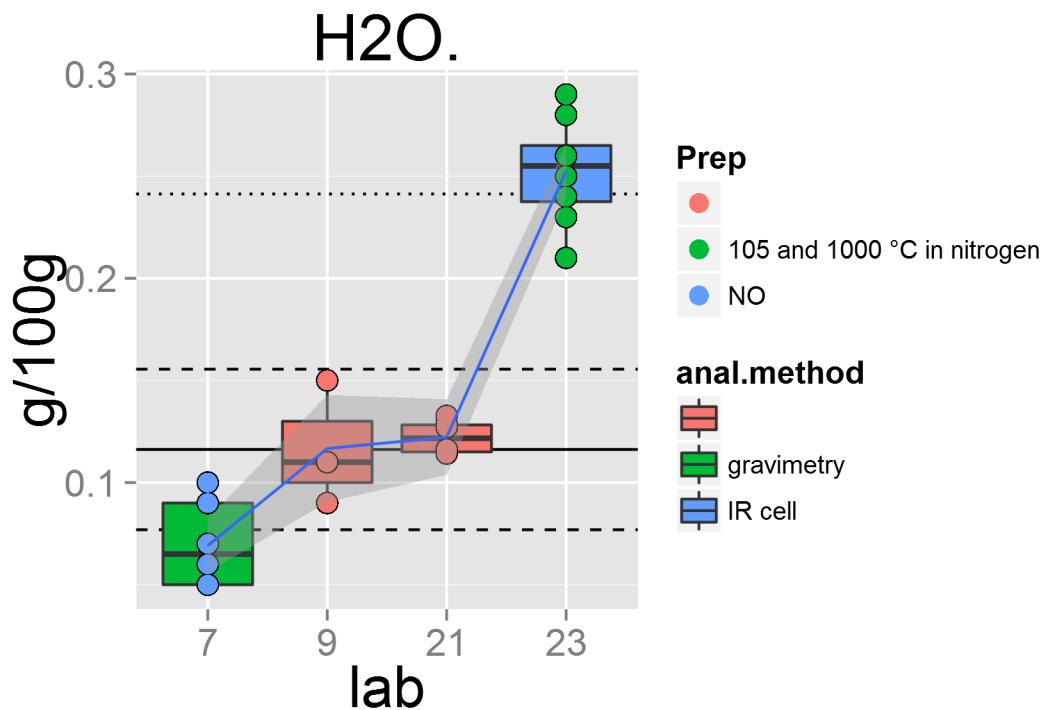
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



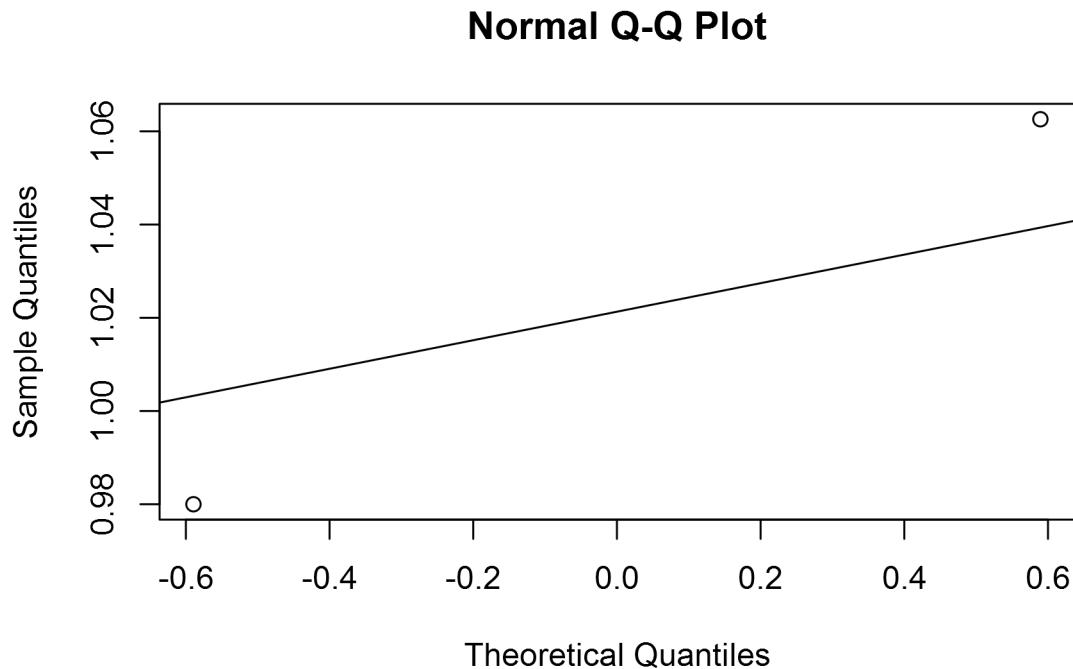
```
## [1] "H20..1"
```



```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



```
## [1] "CO2.1"
```

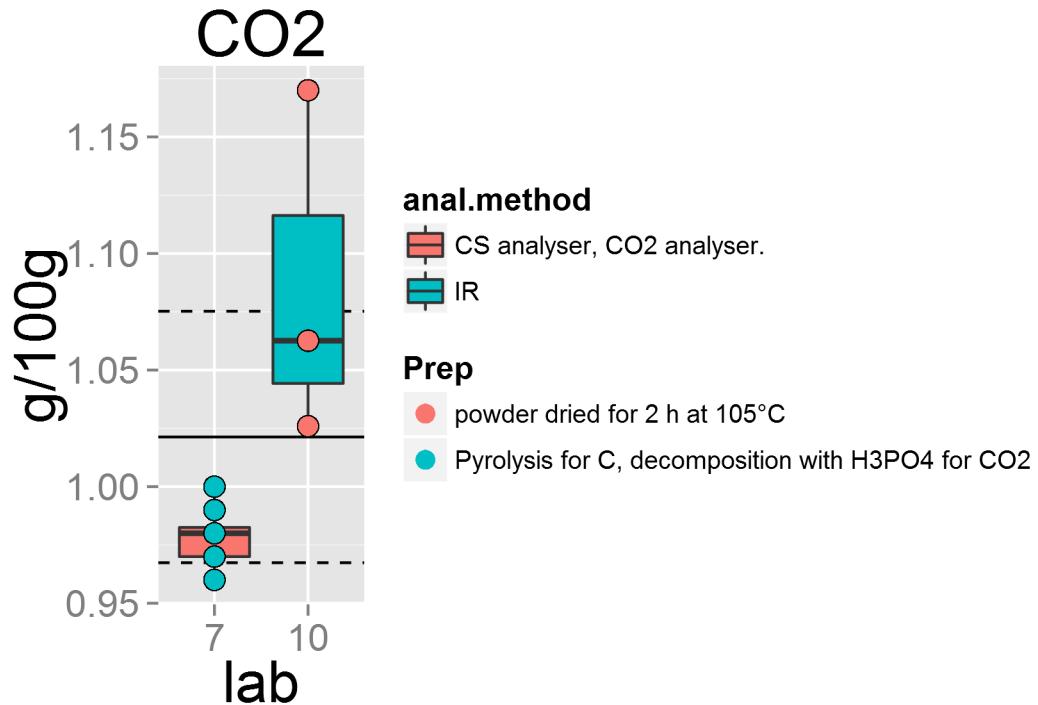


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```

```

## Warning: at  0.995
## Warning: radius  2.5e-005
## Warning: all data on boundary of neighborhood. make span bigger
## Warning: pseudoinverse used at 0.995
## Warning: neighborhood radius 0.005
## Warning: reciprocal condition number  1
## Warning: There are other near singularities as well. 1.01
## Warning: zero-width neighborhood. make span bigger

```

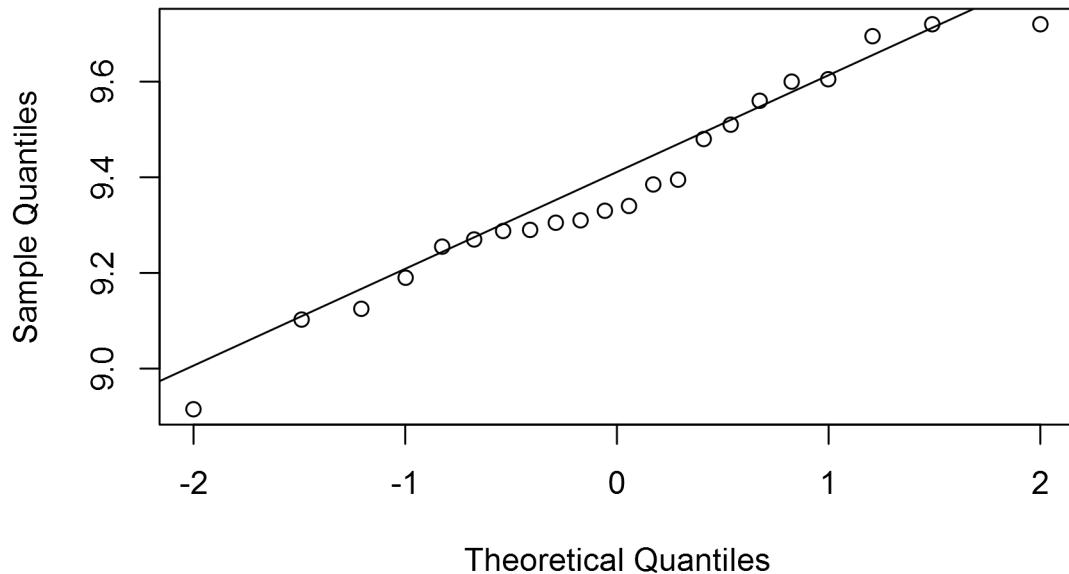


```

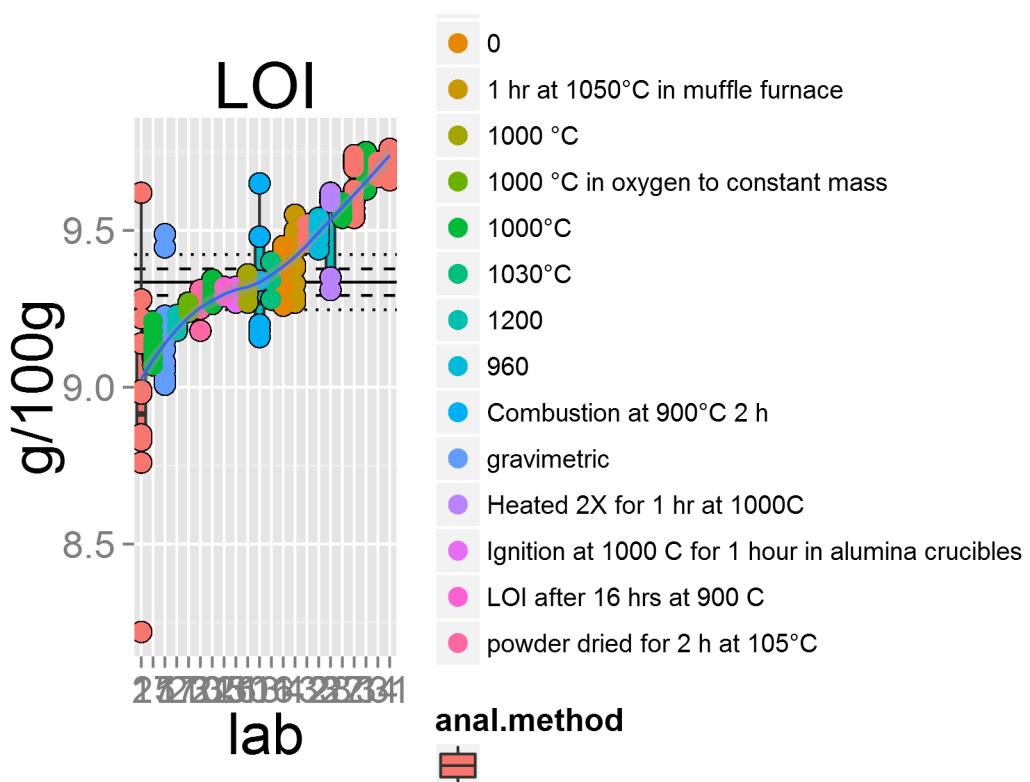
## [1] "LOI.1"

```

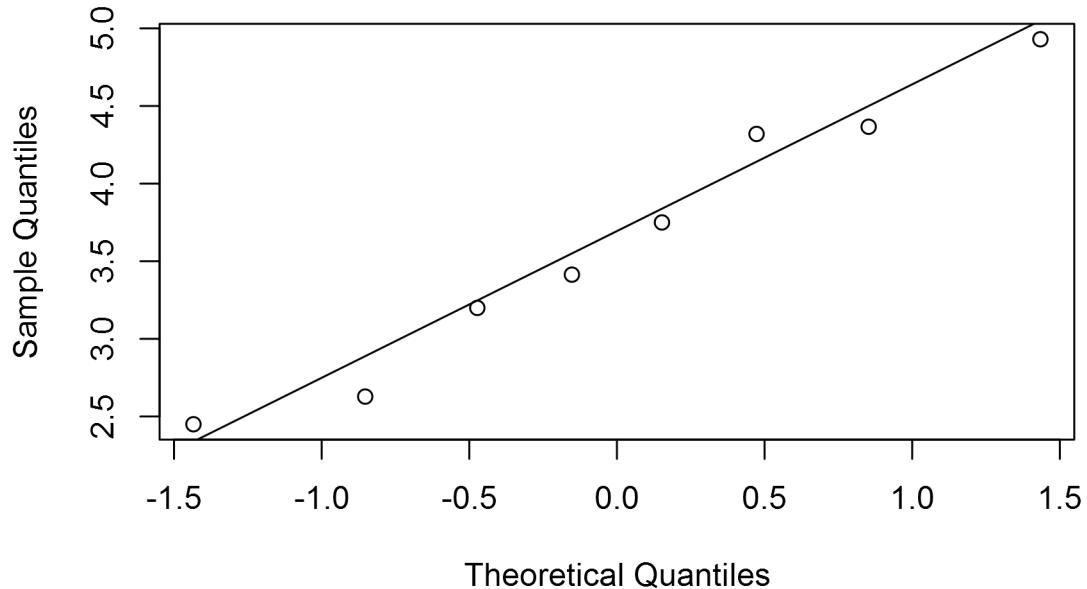
Normal Q-Q Plot



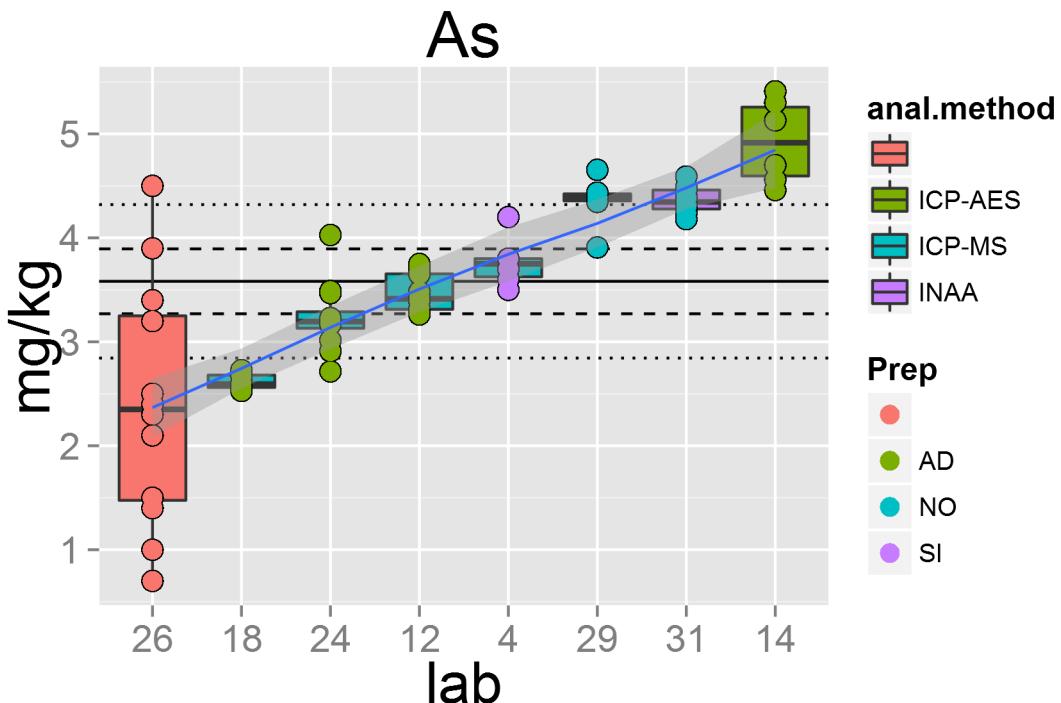
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to c
```



Normal Q-Q Plot

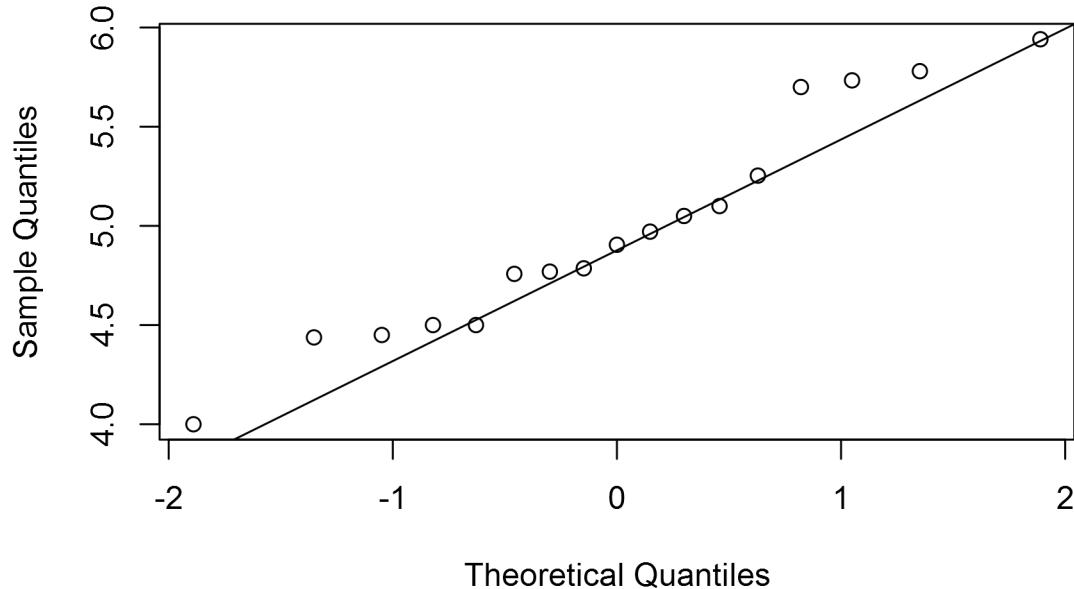


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```

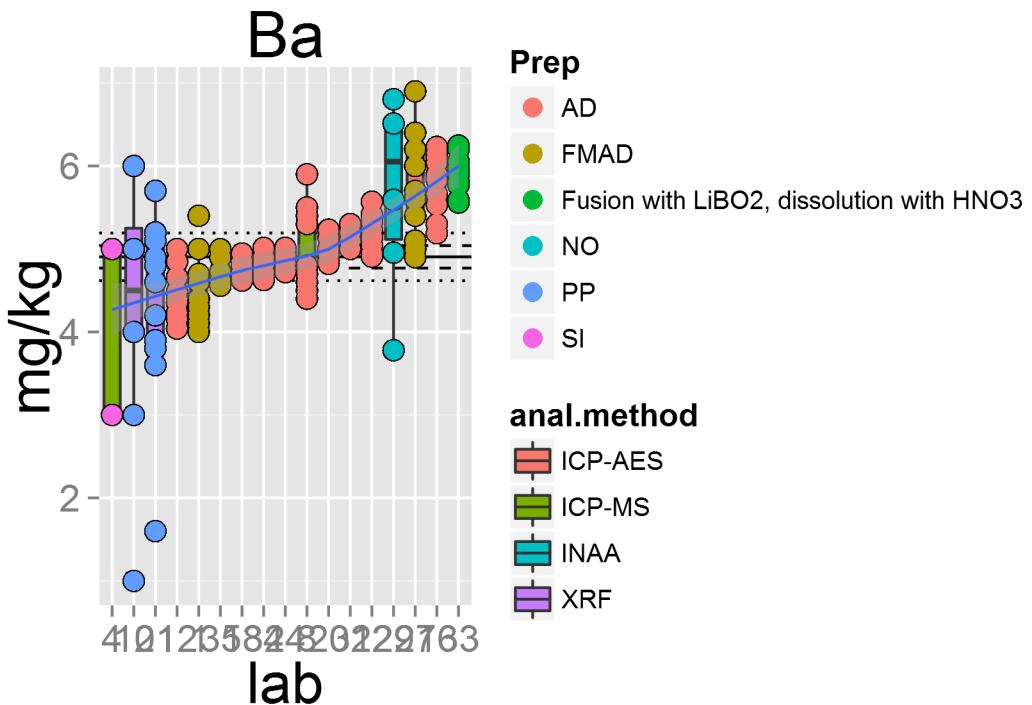


```
## [1] "Ba.1"
```

Normal Q-Q Plot

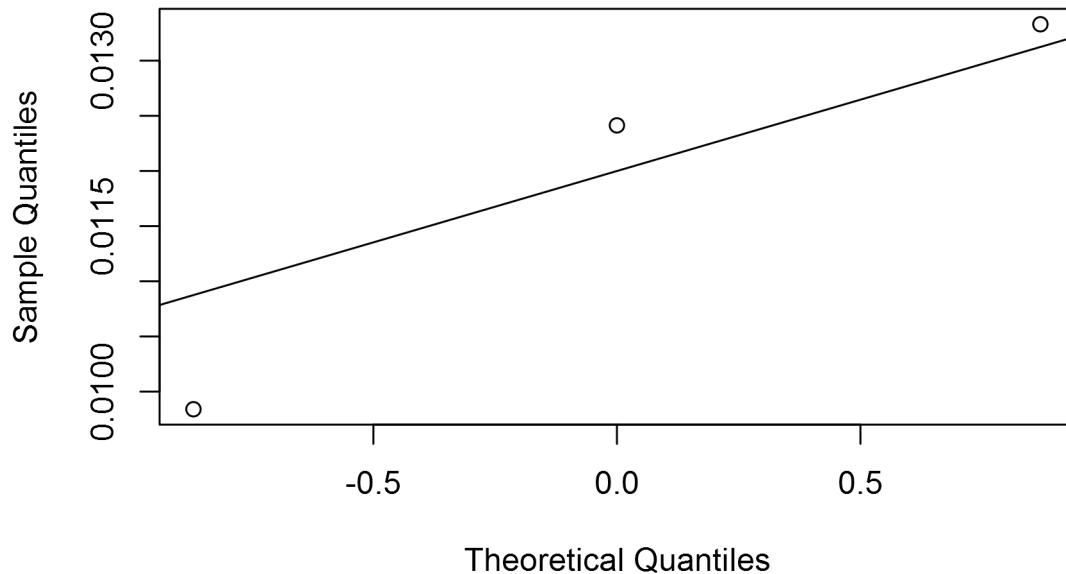


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```

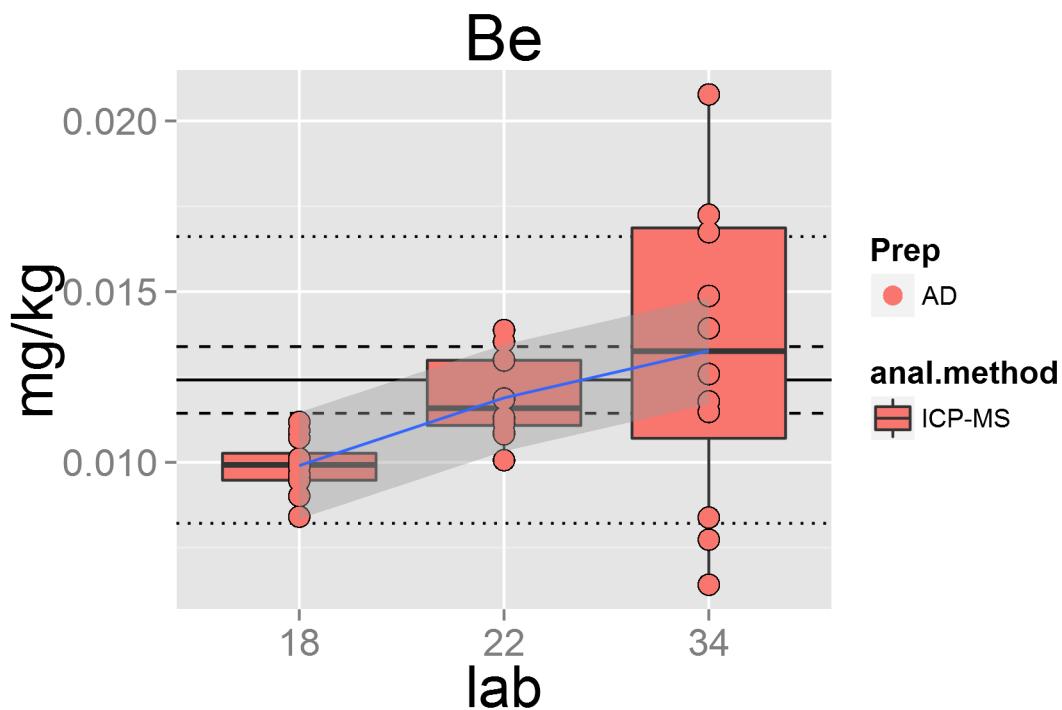


```
## [1] "Be.1"
```

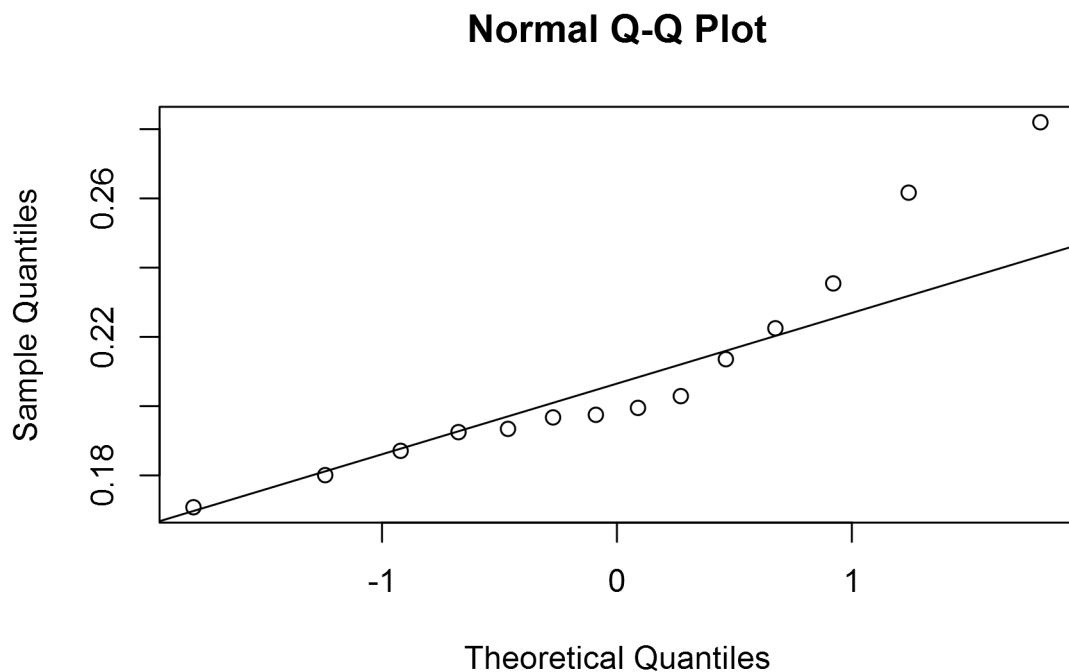
Normal Q-Q Plot



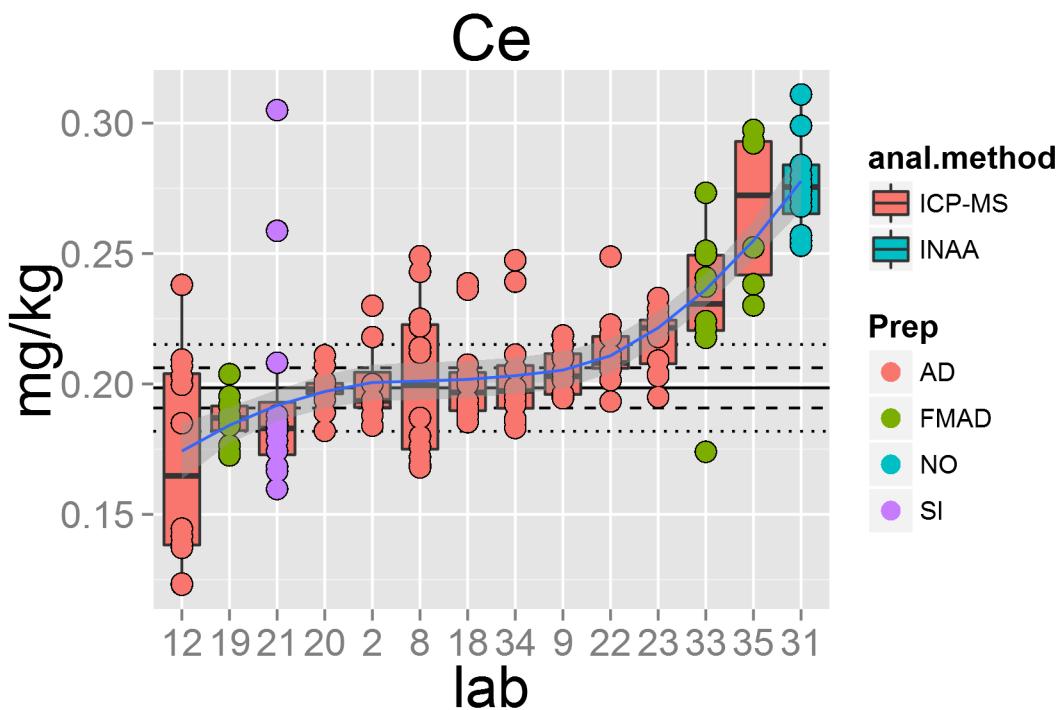
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to c
## Warning: pseudoinverse used at 0.99
## Warning: neighborhood radius 2.01
## Warning: reciprocal condition number  2.581e-016
## Warning: There are other near singularities as well. 4.0401
## Warning: pseudoinverse used at 0.99
## Warning: neighborhood radius 2.01
## Warning: reciprocal condition number  2.581e-016
## Warning: There are other near singularities as well. 4.0401
```



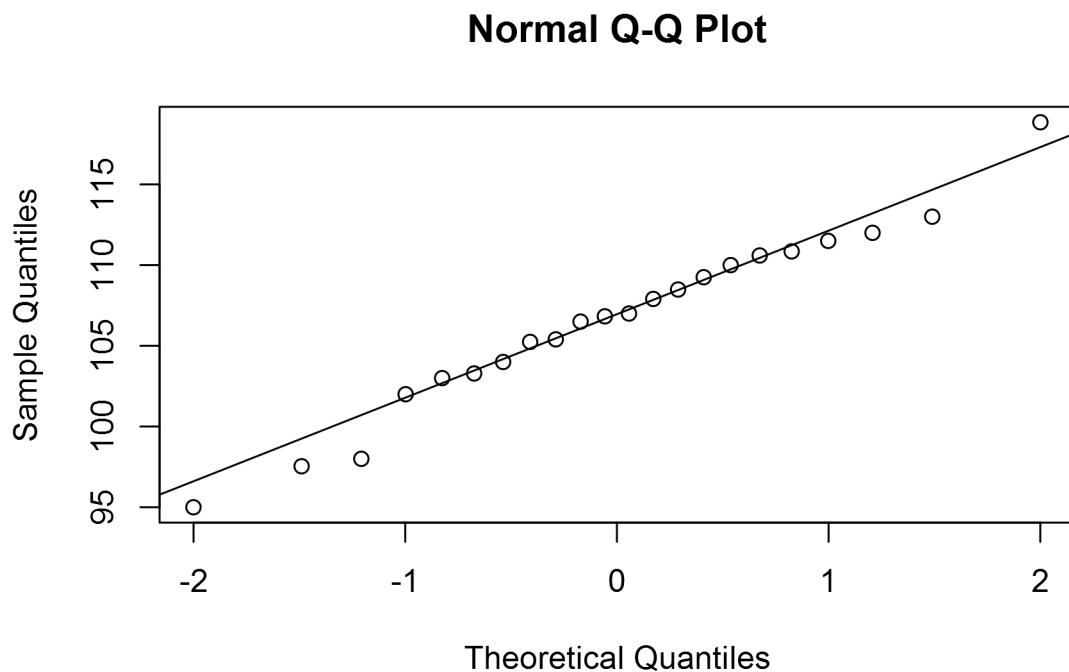
```
## [1] "Ce.1"
```



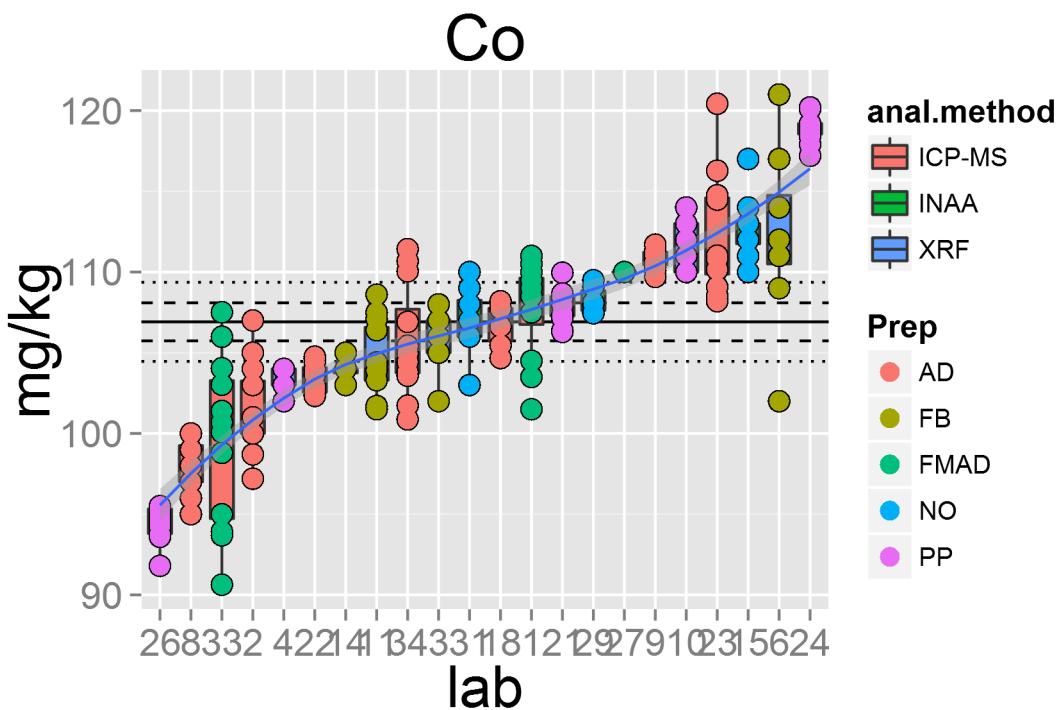
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



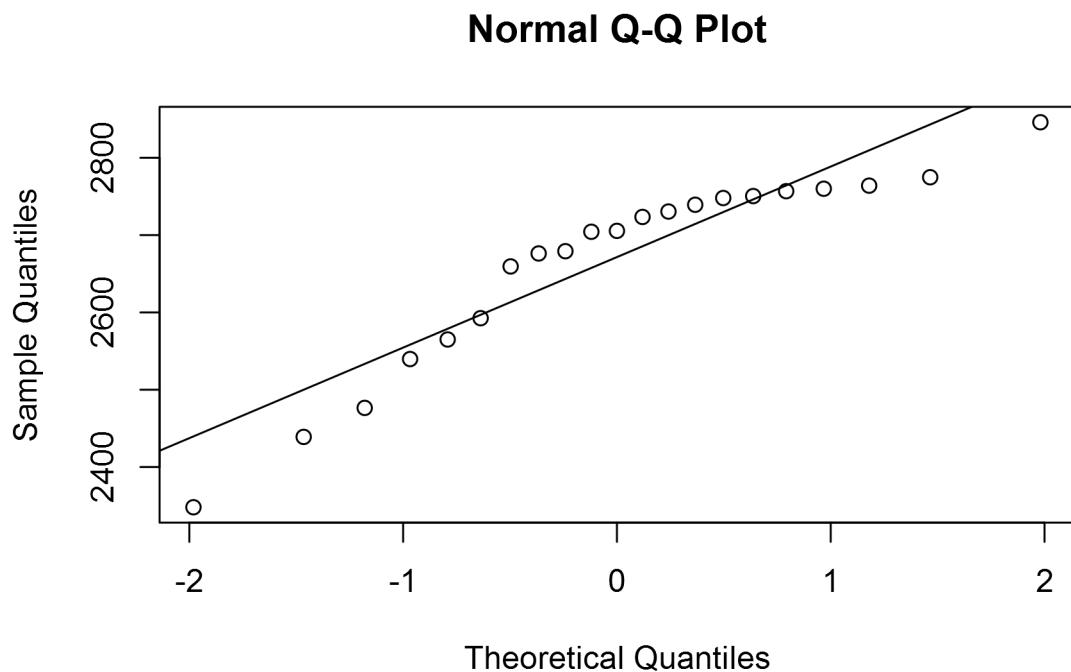
```
## [1] "Co.1"
```



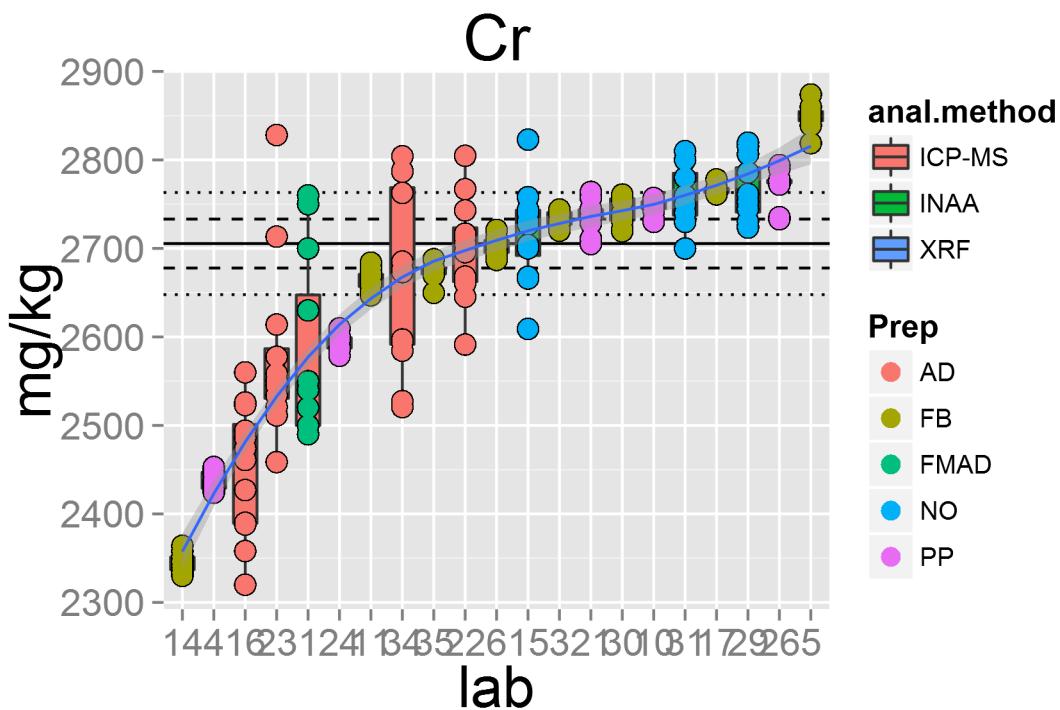
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



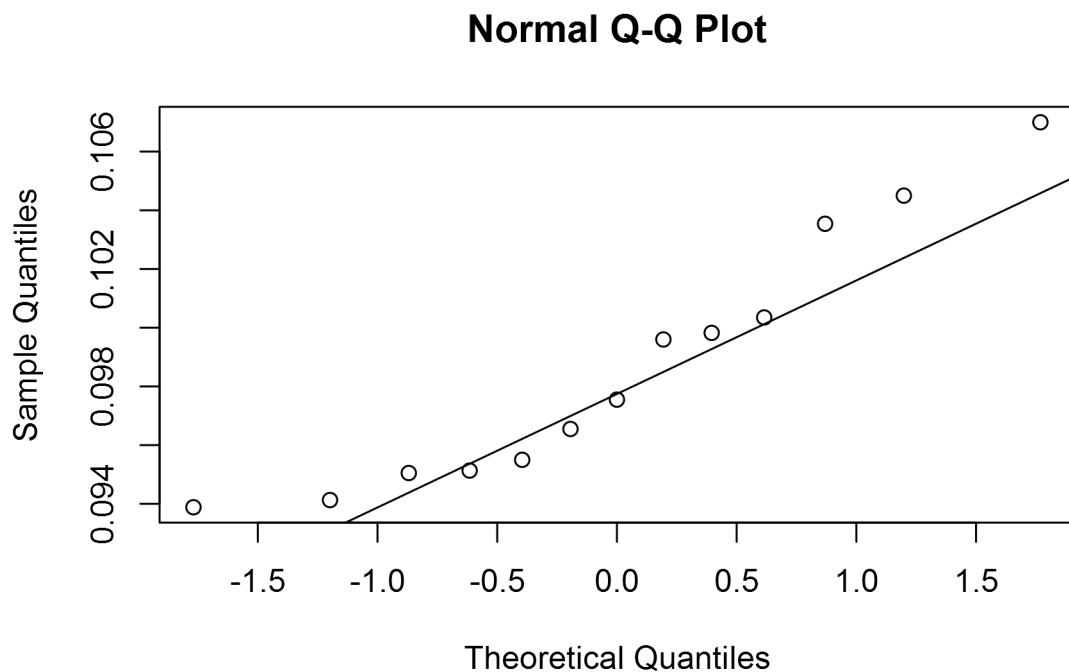
```
## [1] "Cr.1"
```



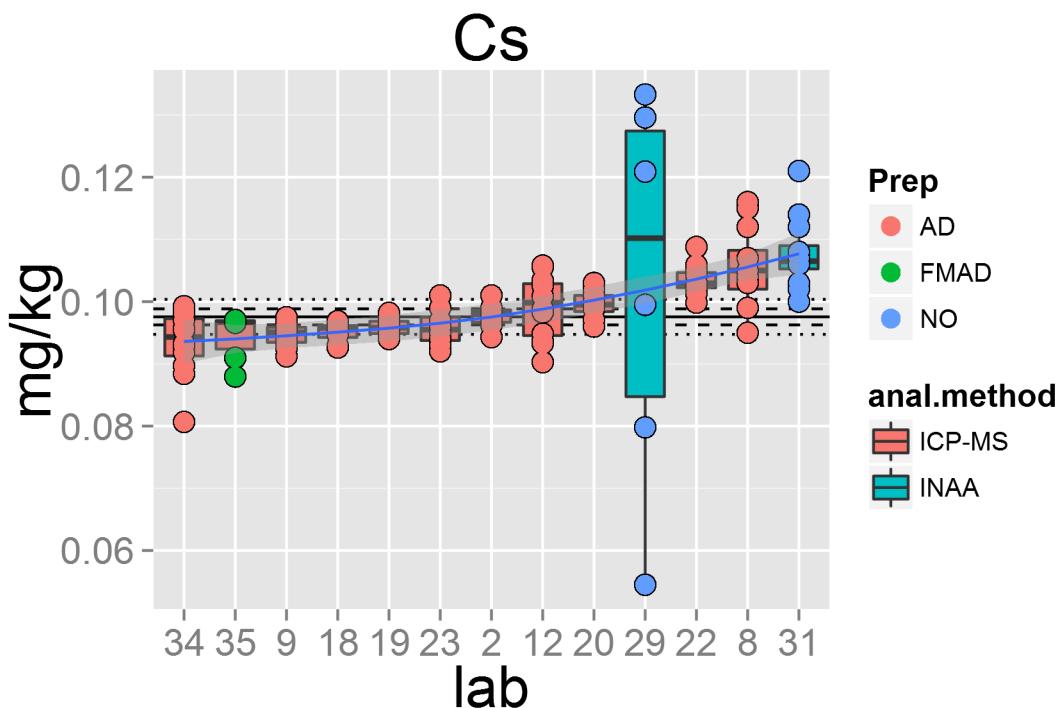
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



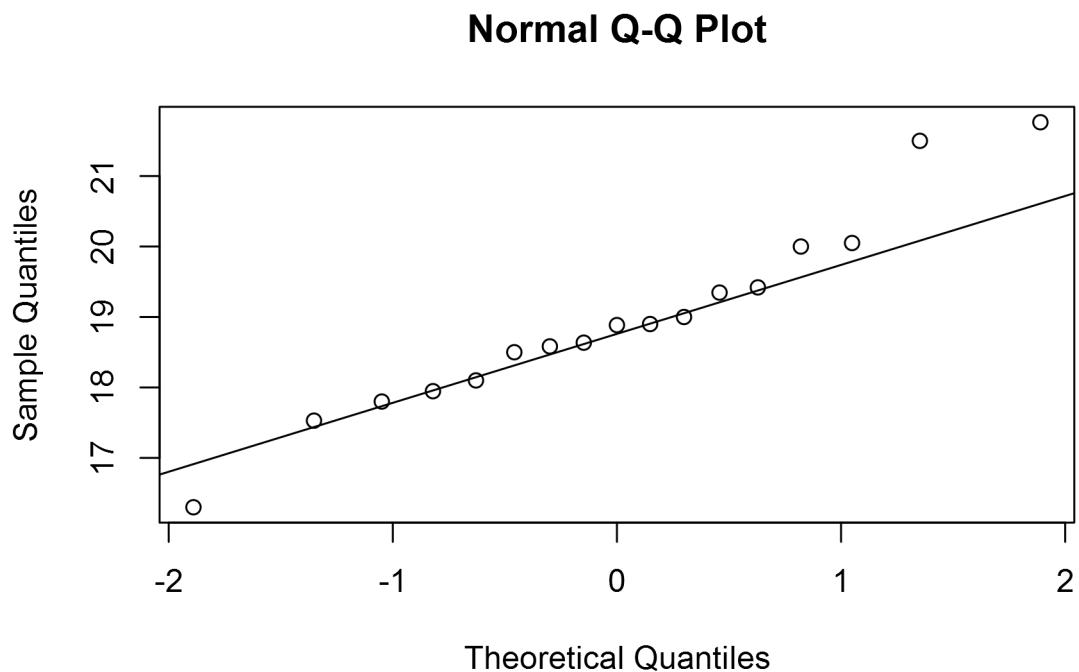
```
## [1] "Cs.1"
```



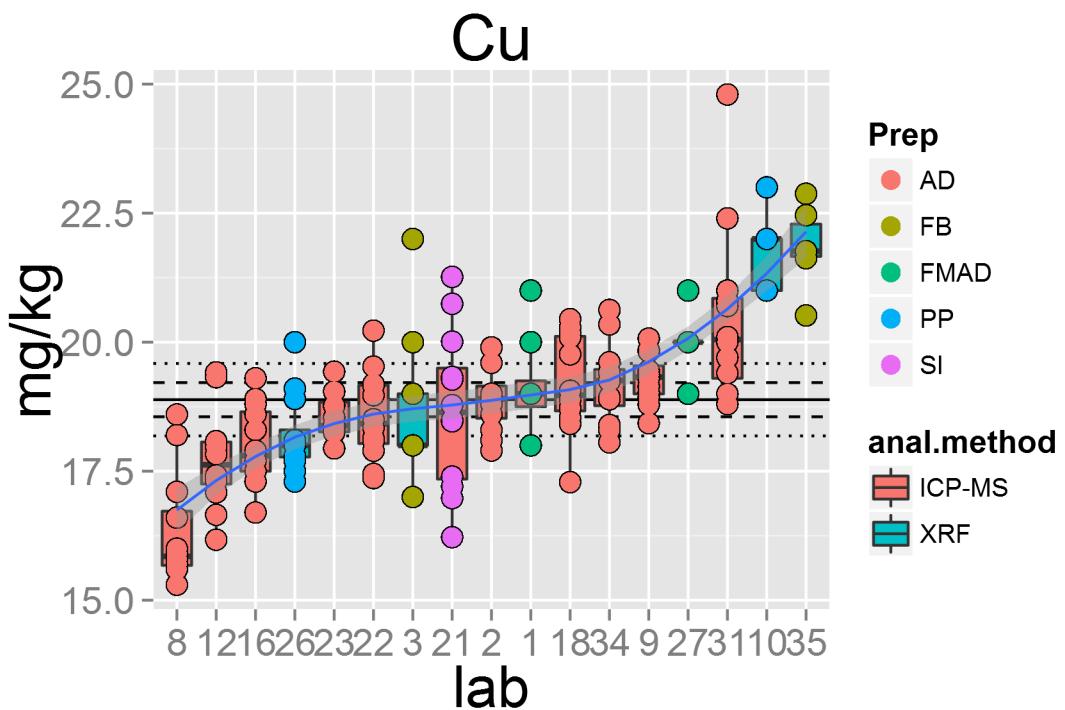
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



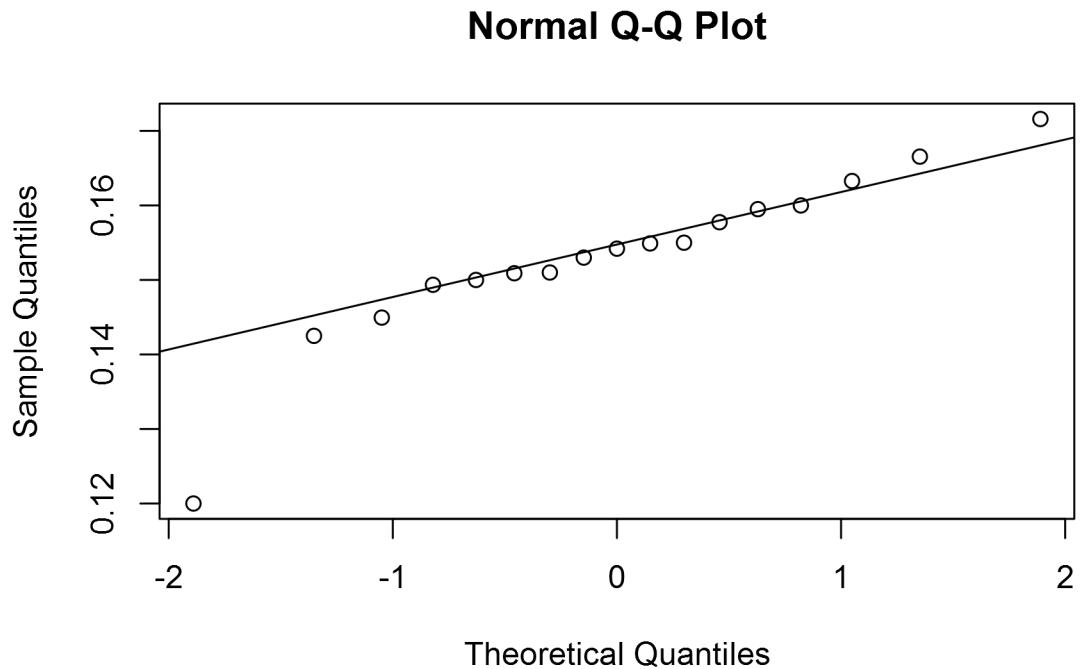
```
## [1] "Cu.1"
```



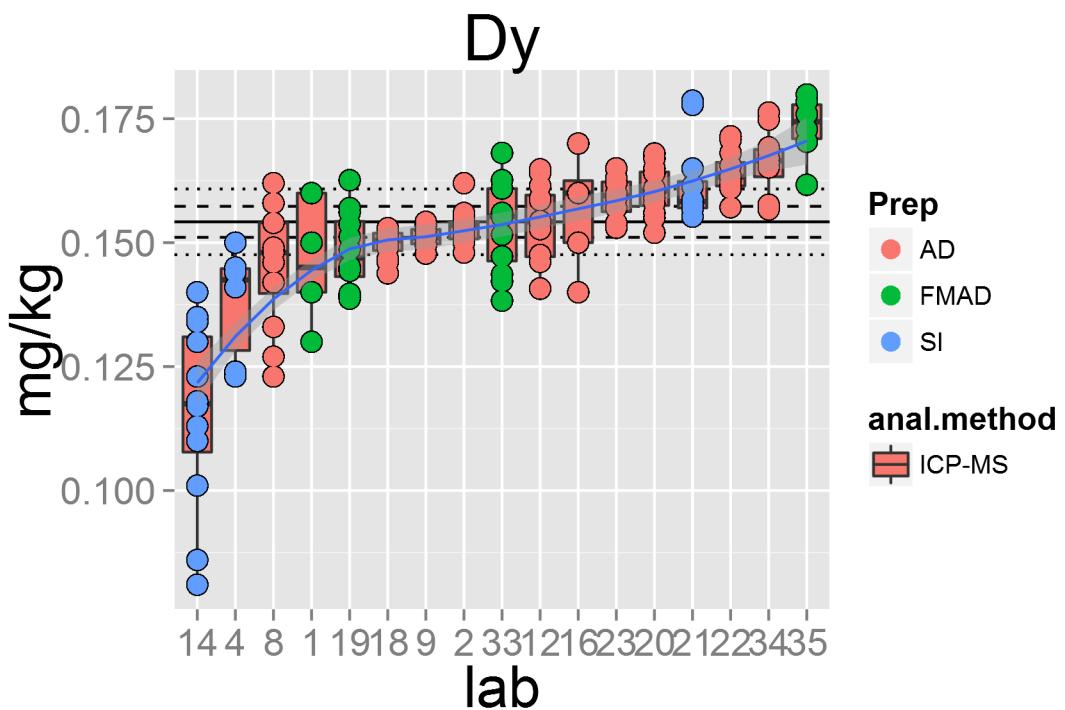
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```

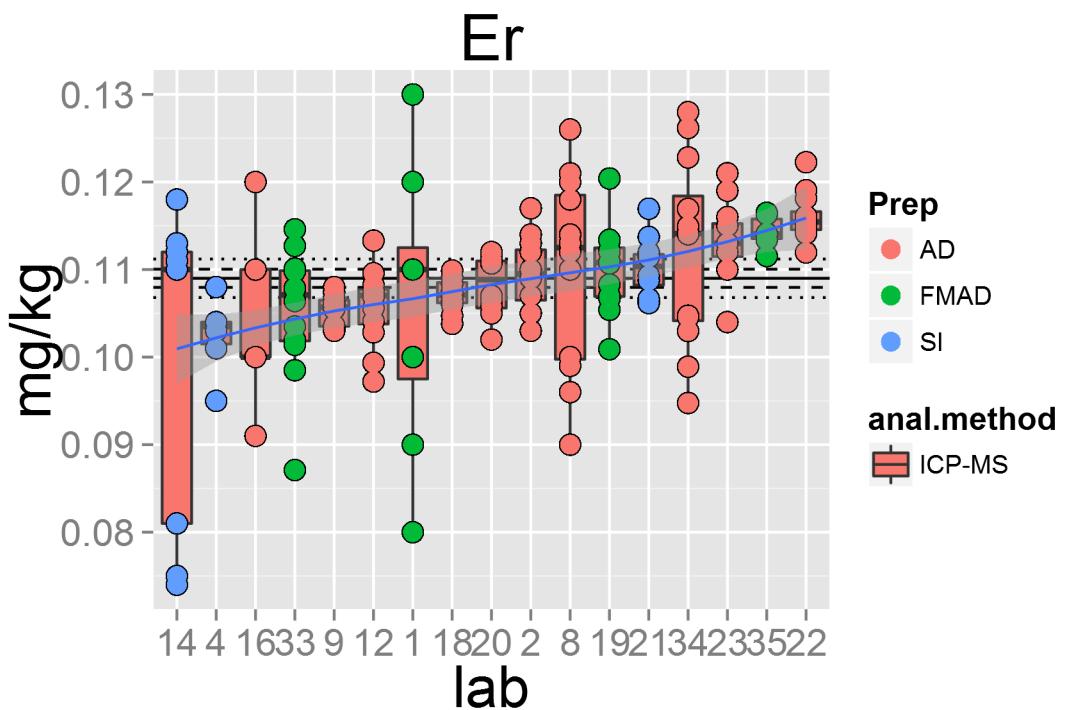


```
## [1] "Dy.1"
```

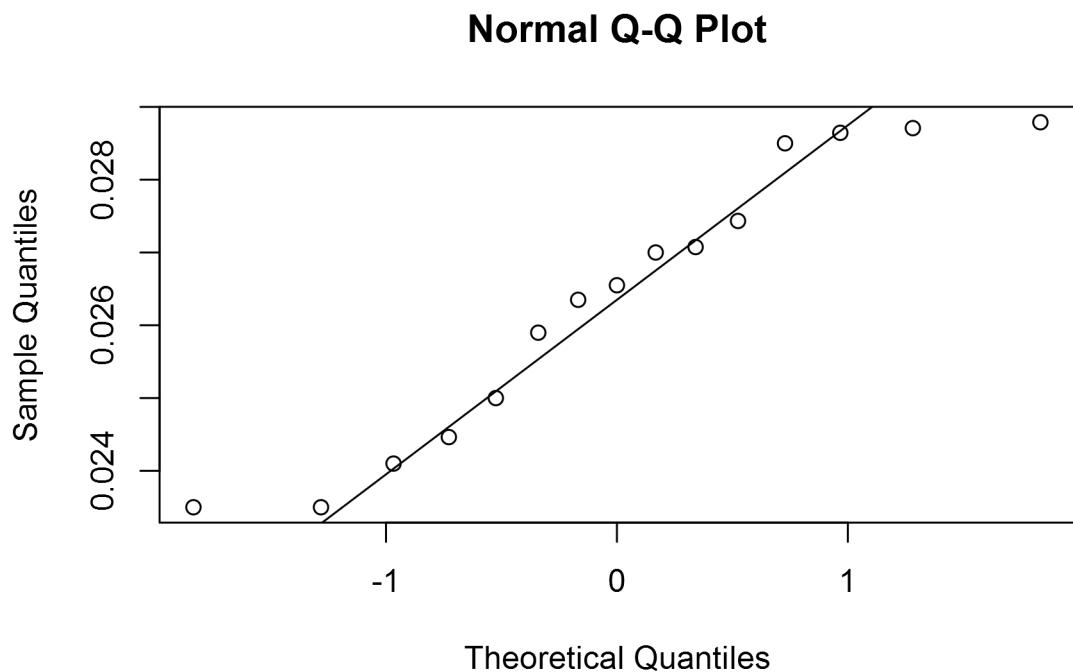


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```

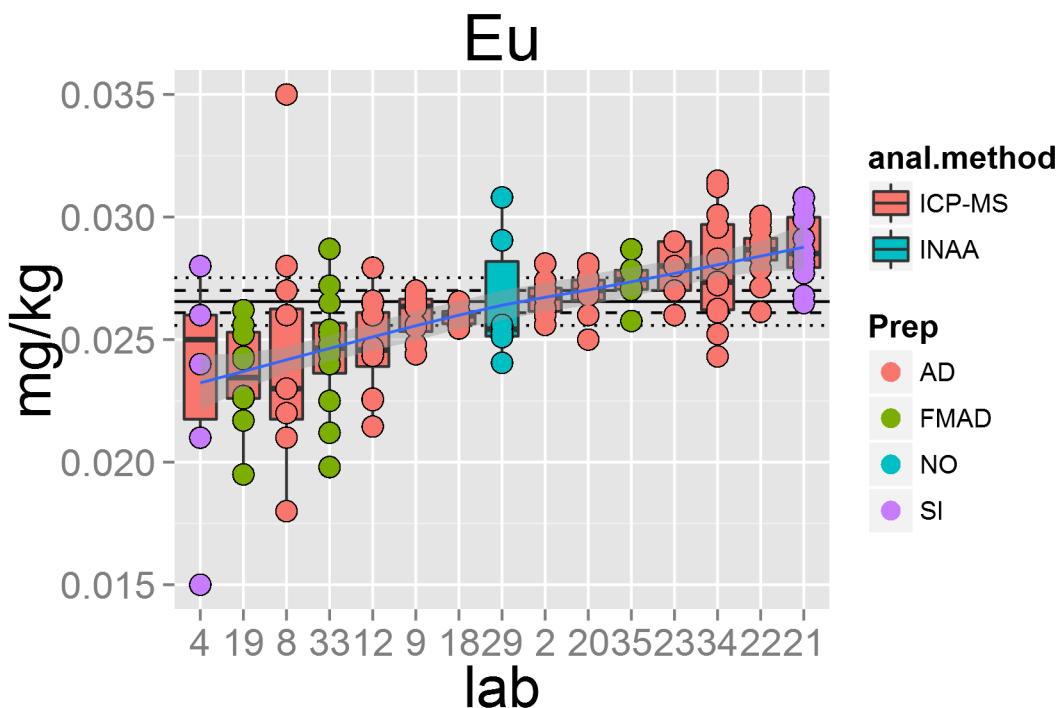




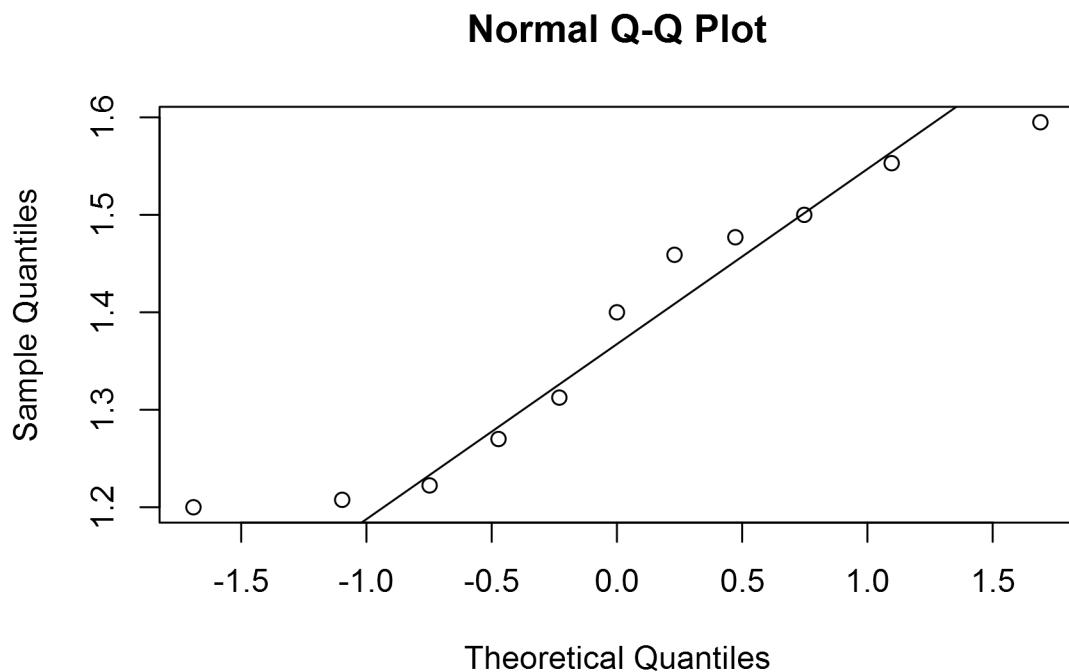
```
## [1] "Eu.1"
```



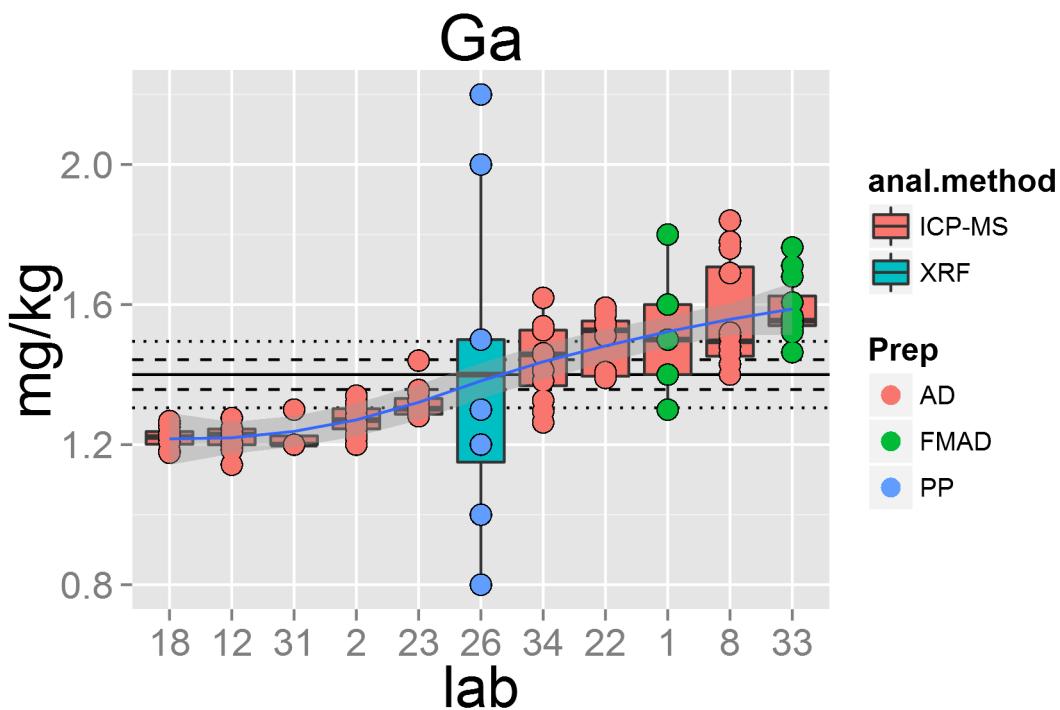
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



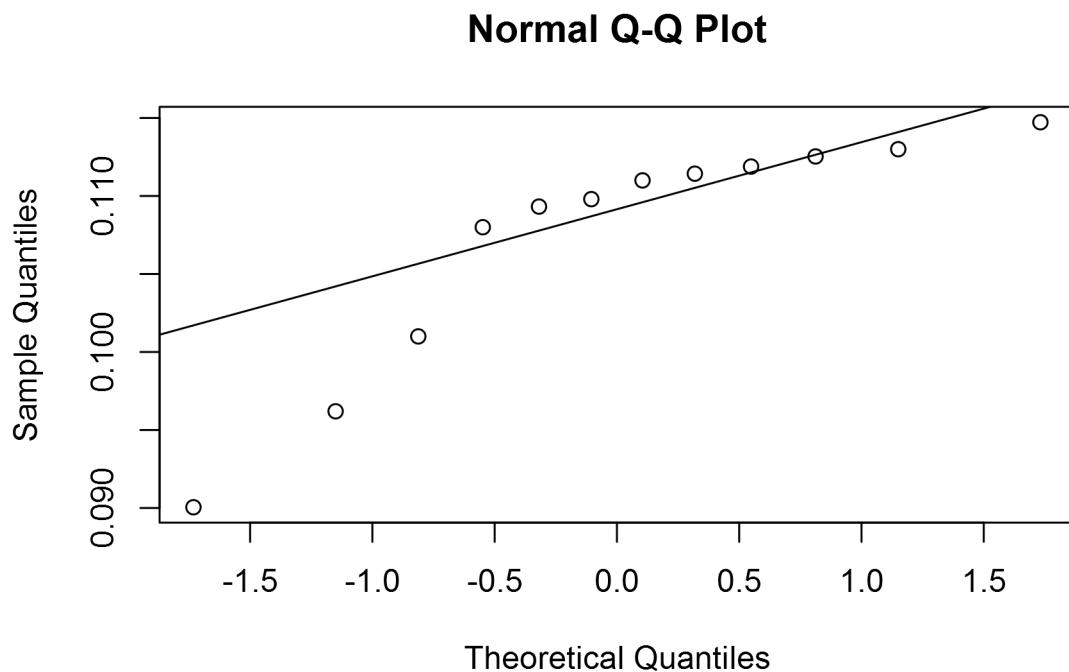
```
## [1] "Ga.1"
```



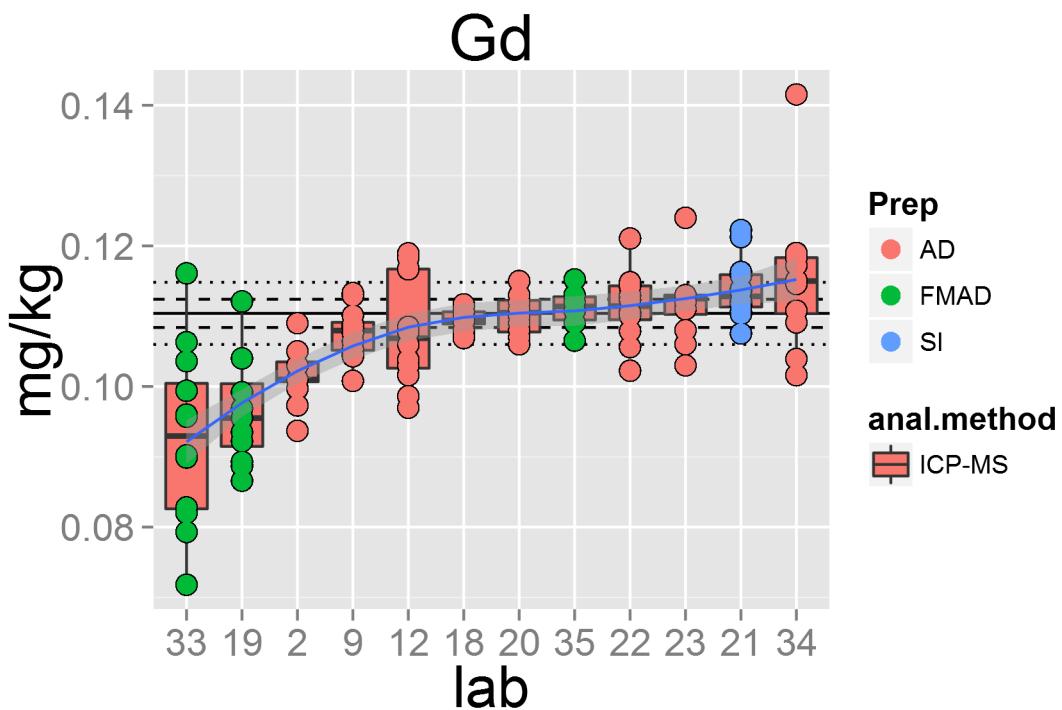
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



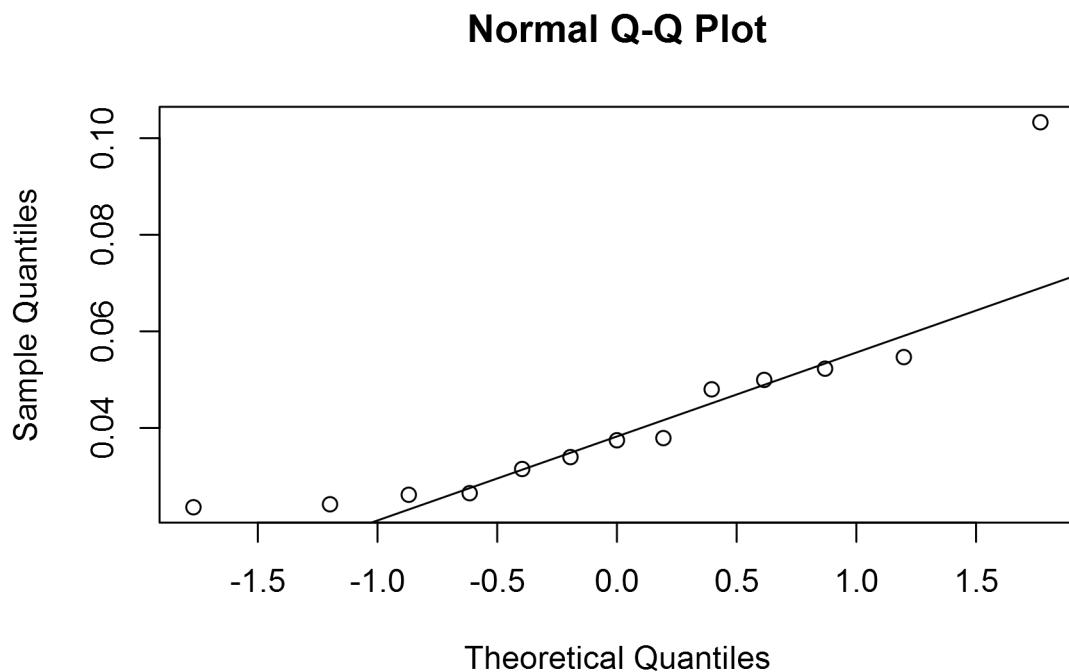
```
## [1] "Gd.1"
```



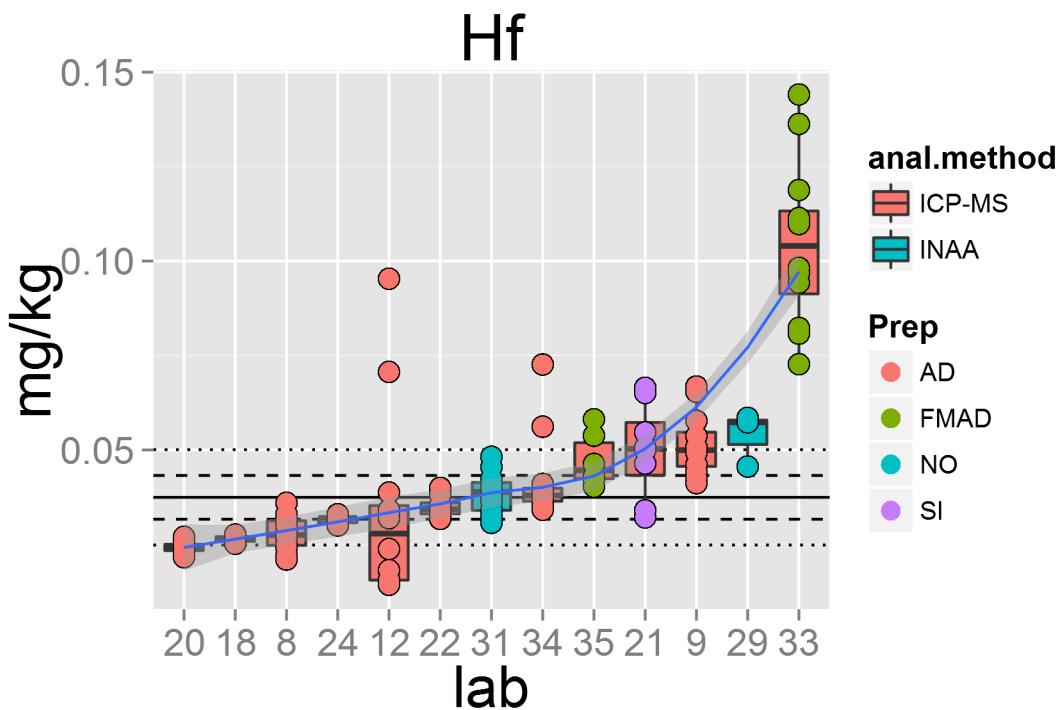
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



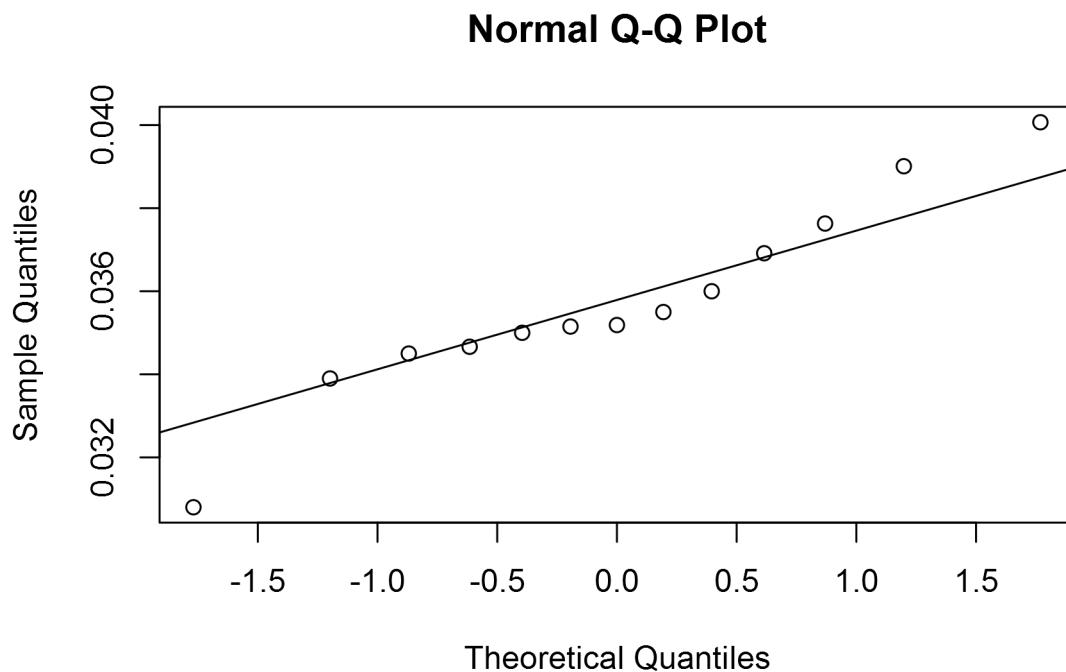
```
## [1] "Hf.1"
```



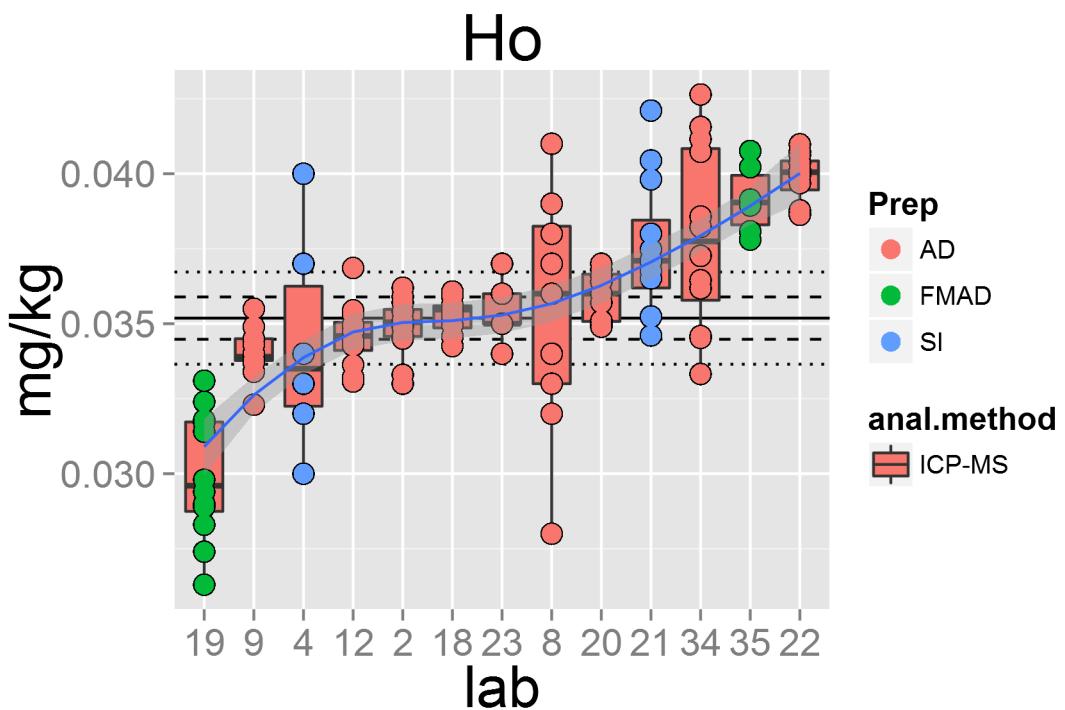
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



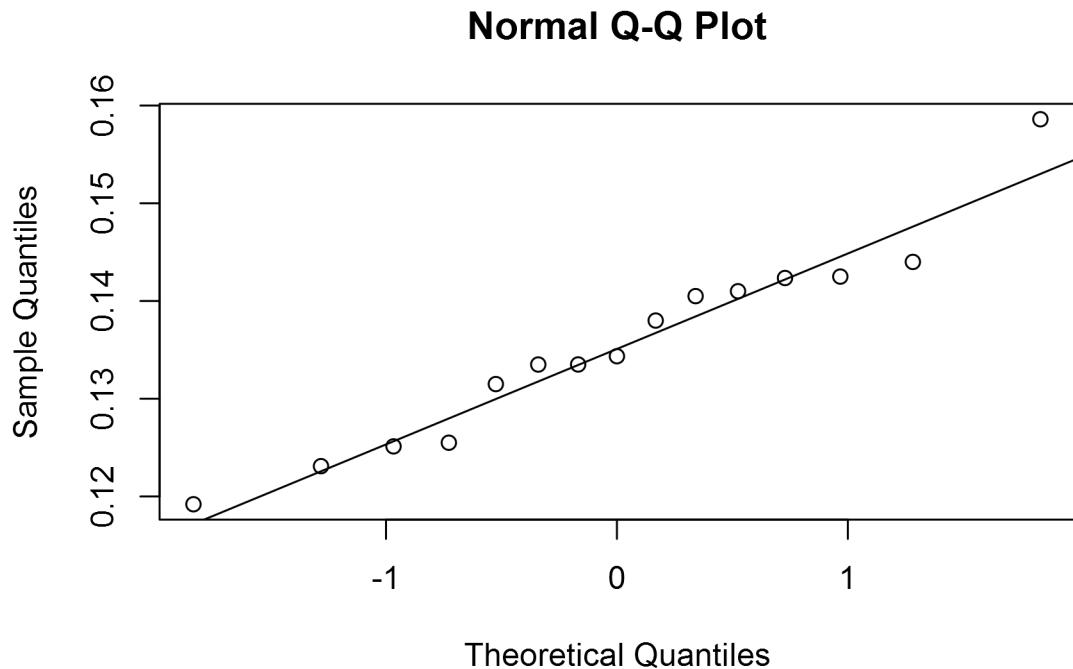
```
## [1] "Ho.1"
```



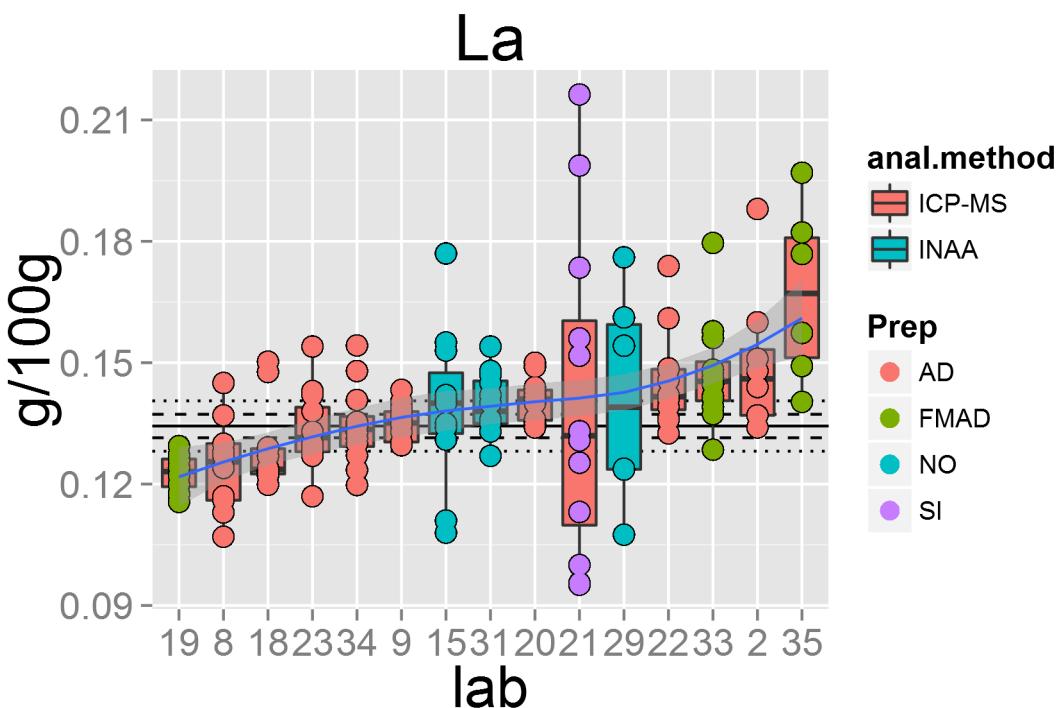
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



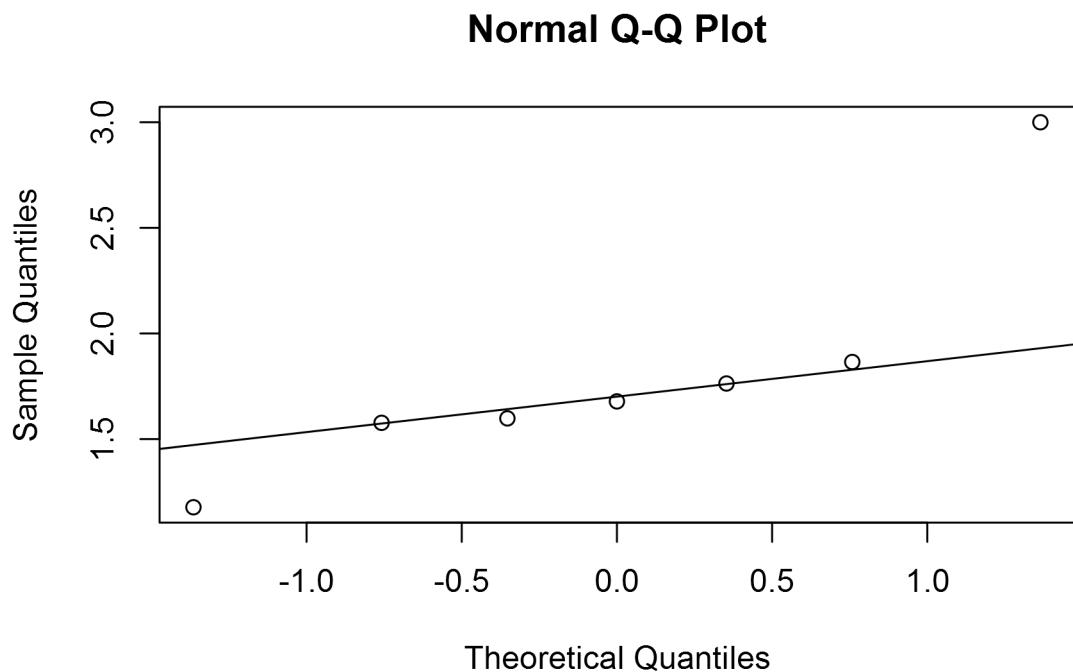
```
## [1] "La.1"
```



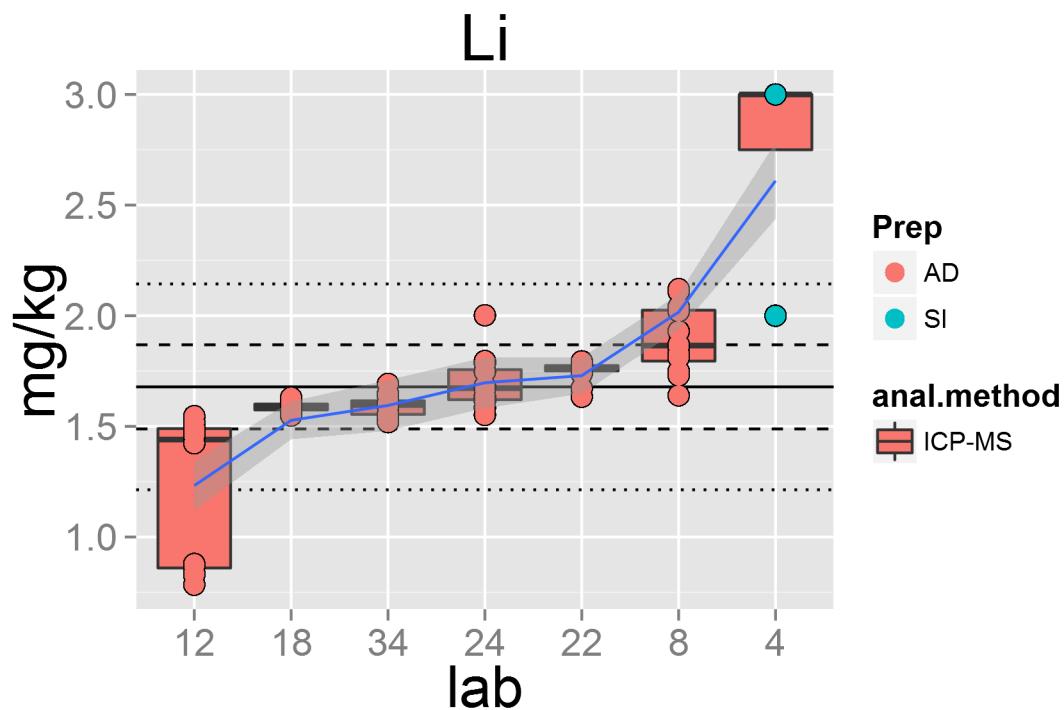
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



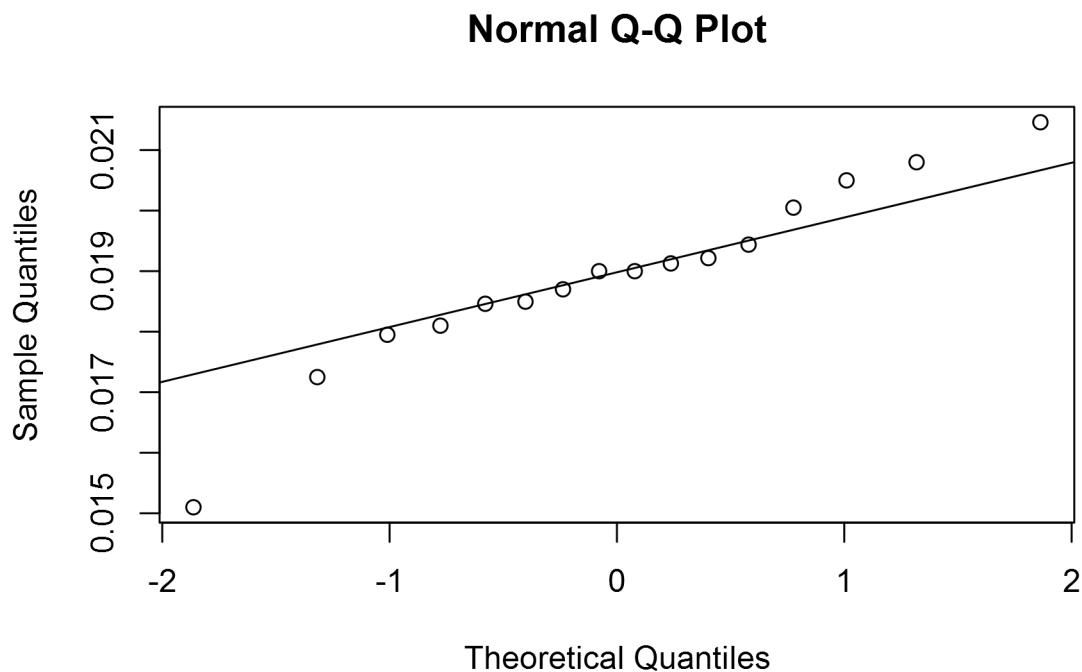
```
## [1] "Li.1"
```



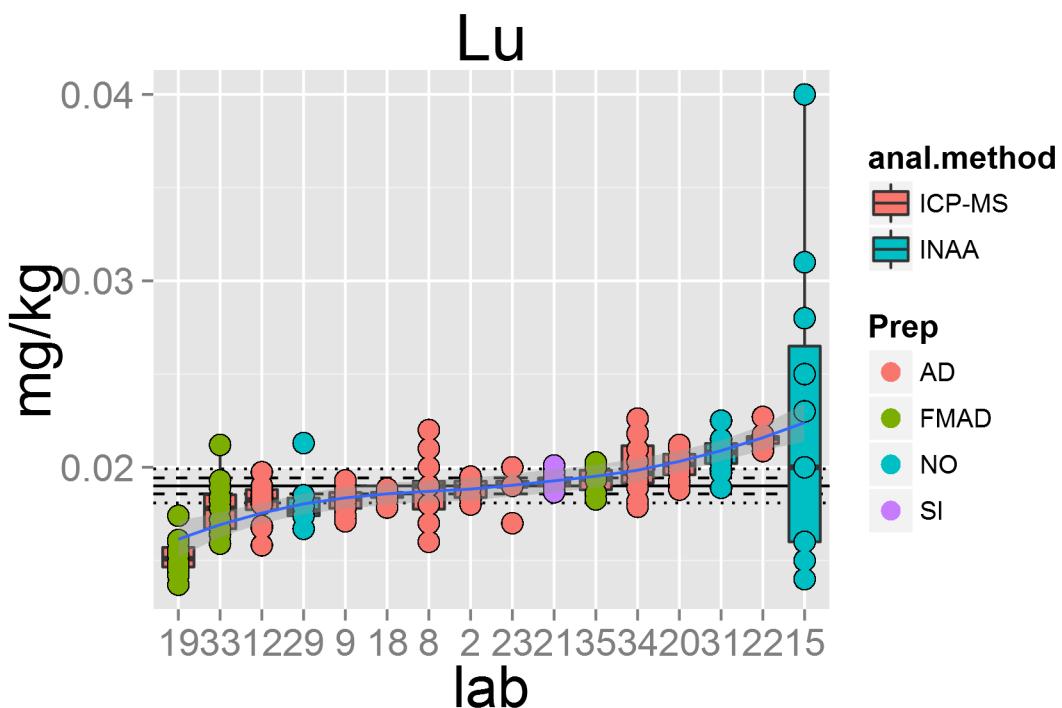
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



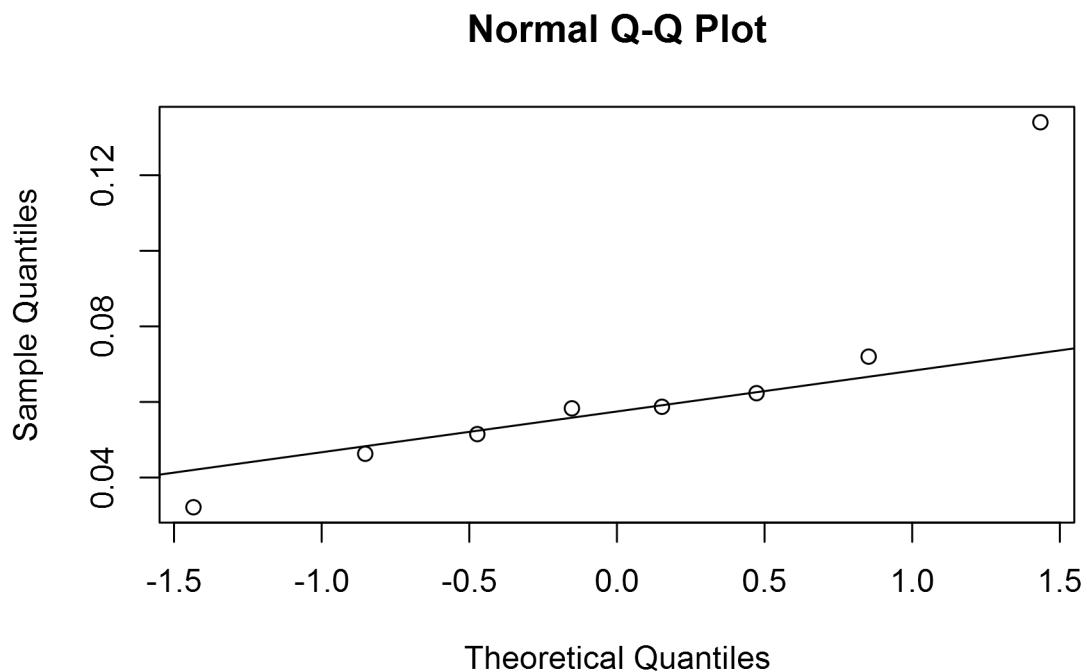
```
## [1] "Lu.1"
```



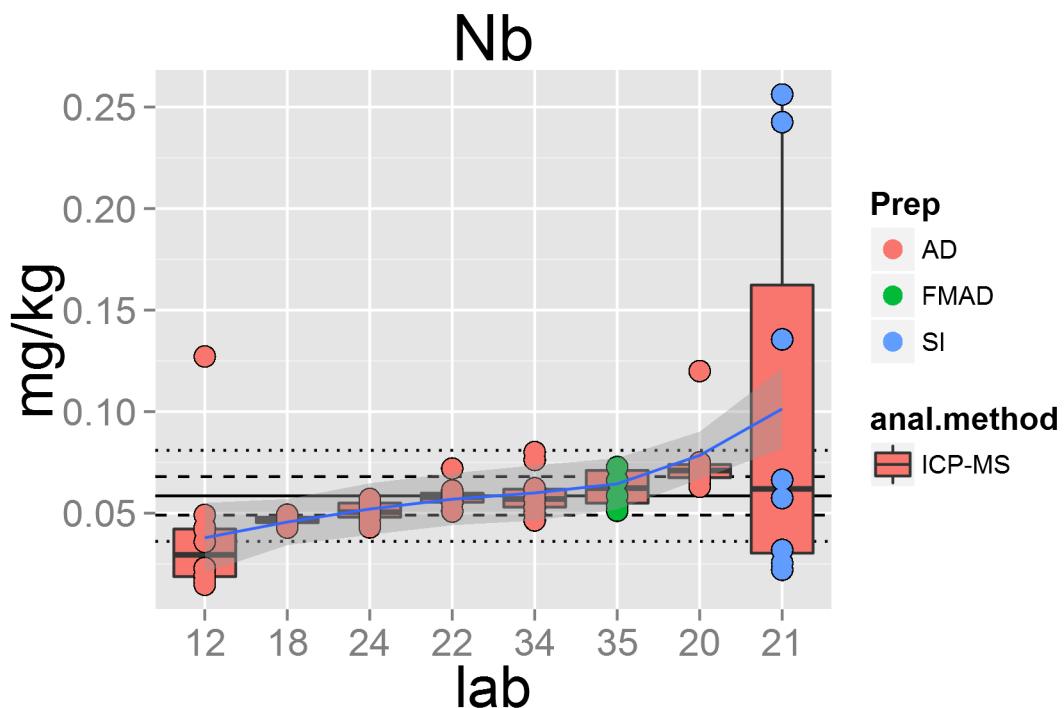
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



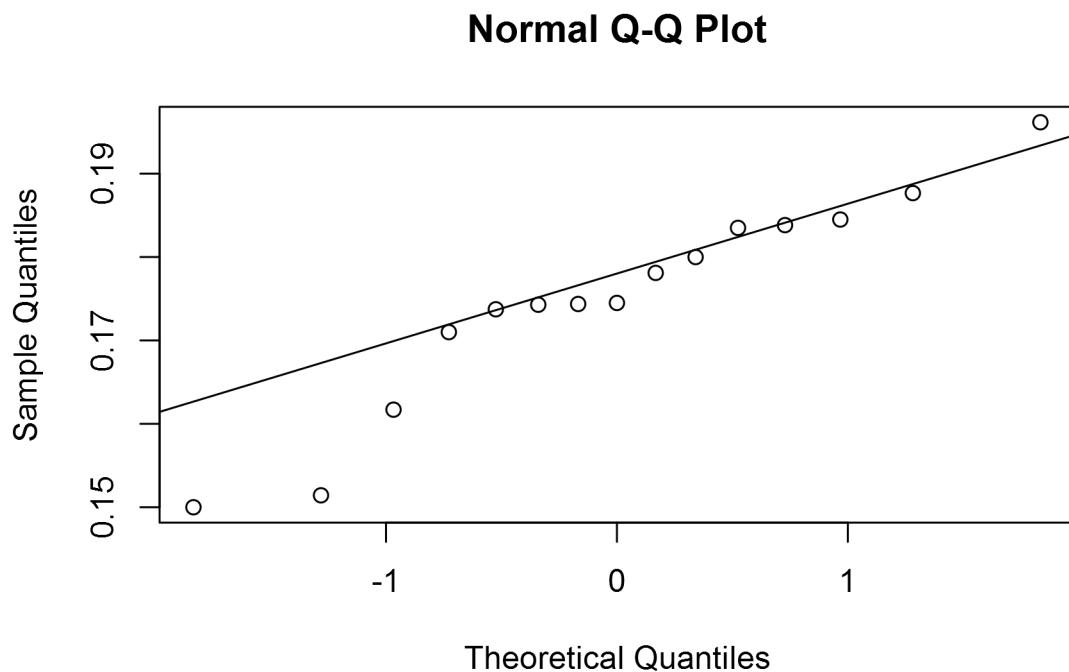
```
## [1] "Nb.1"
```



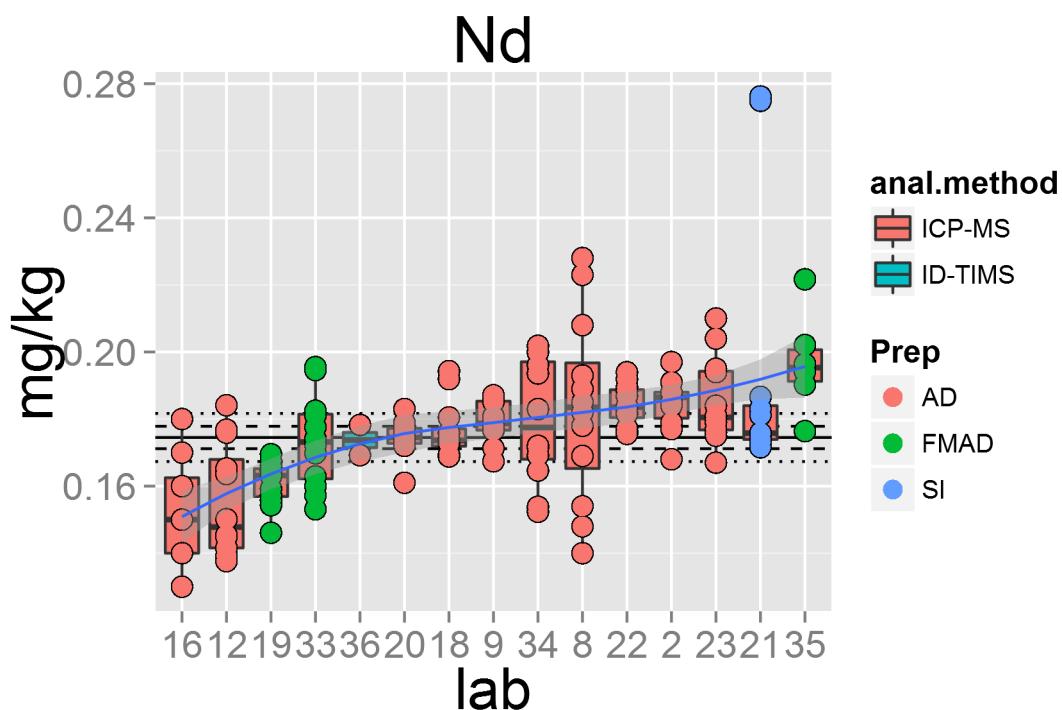
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



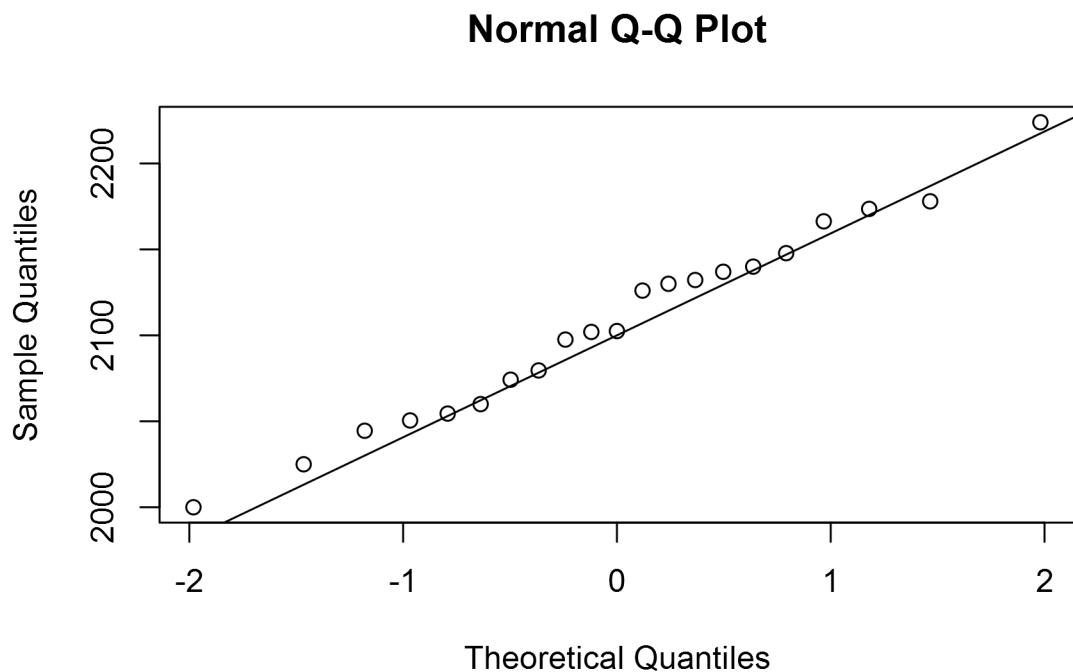
```
## [1] "Nd.1"
```



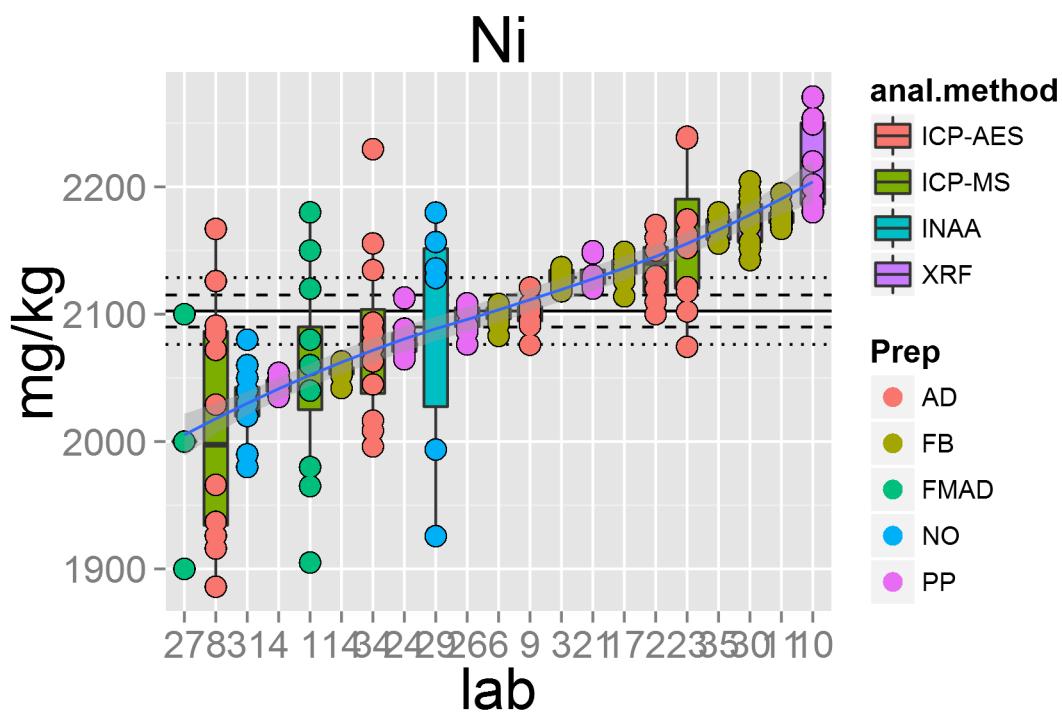
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



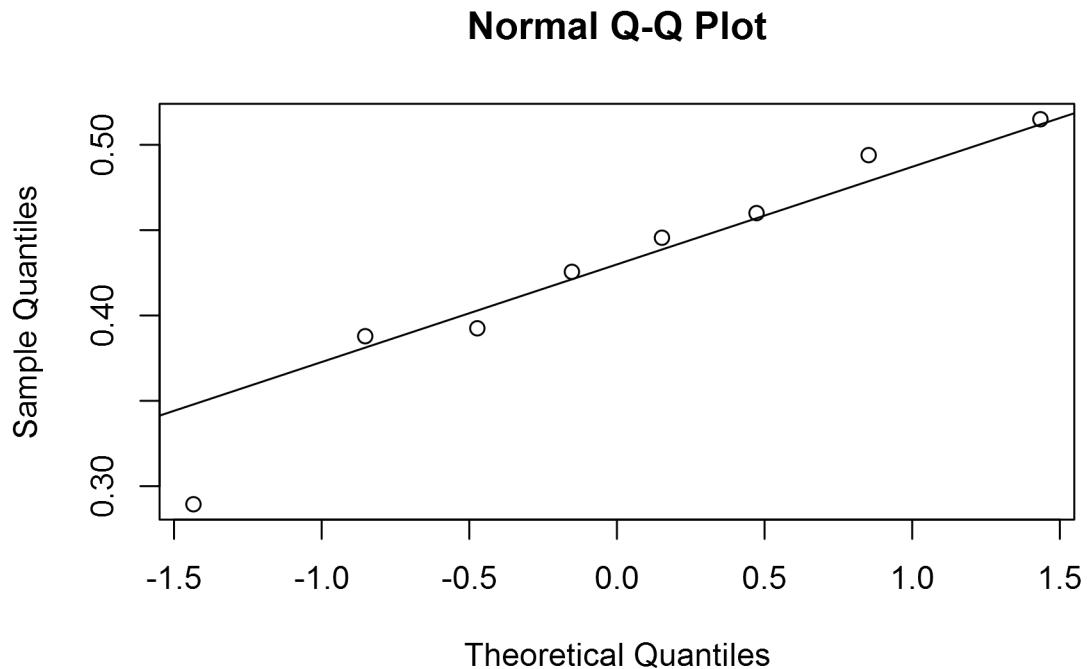
```
## [1] "Ni.1"
```



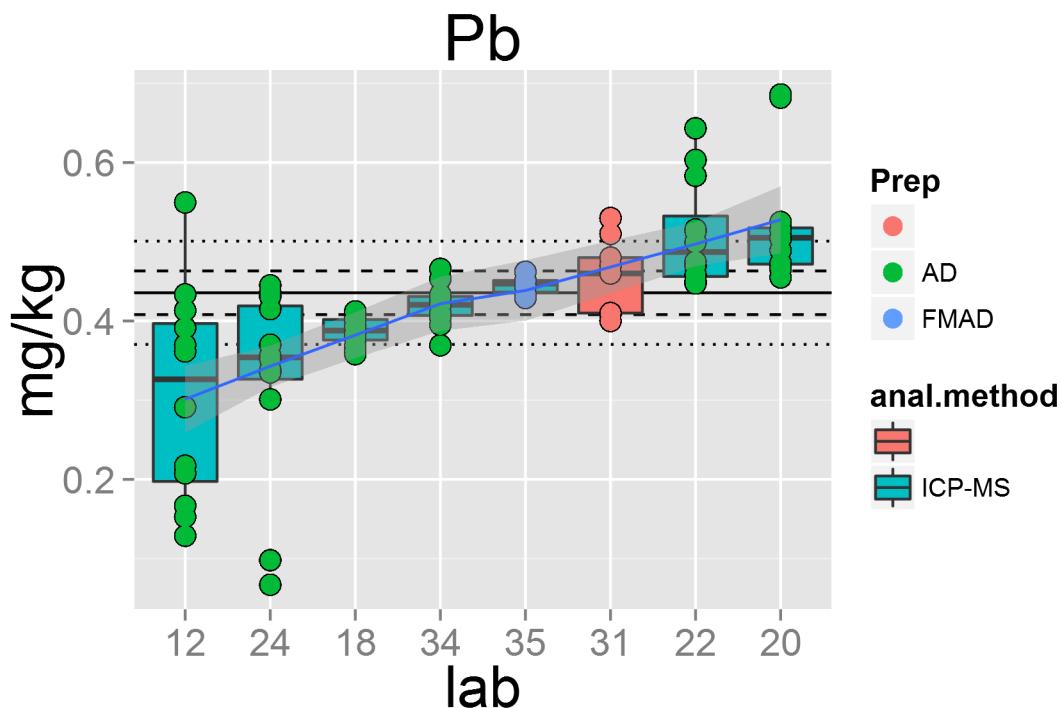
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



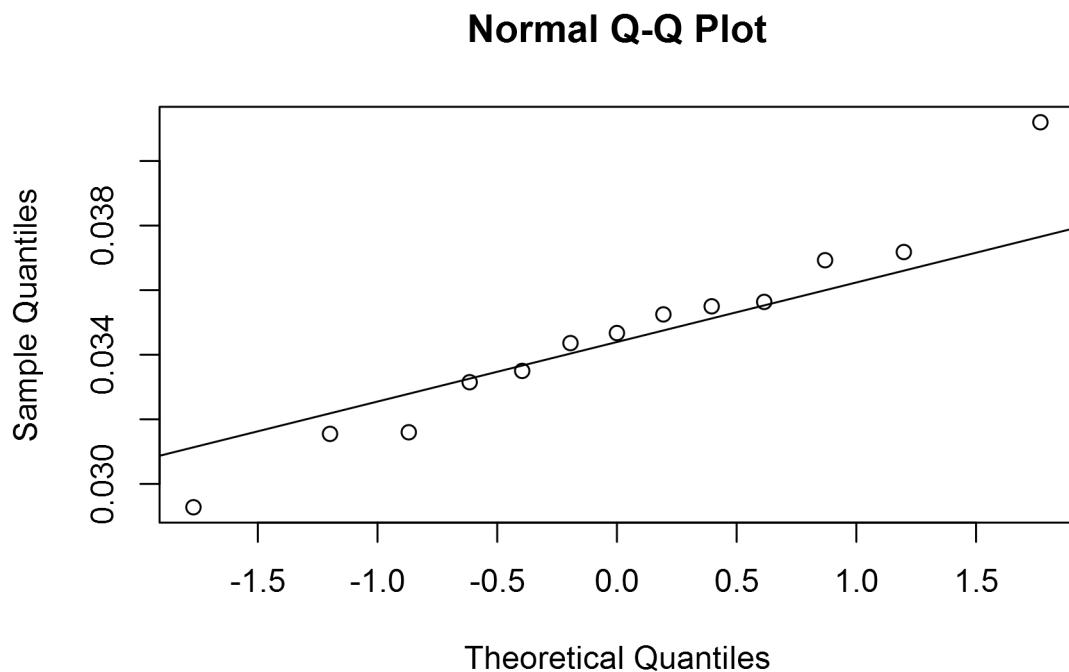
```
## [1] "Pb.1"
```



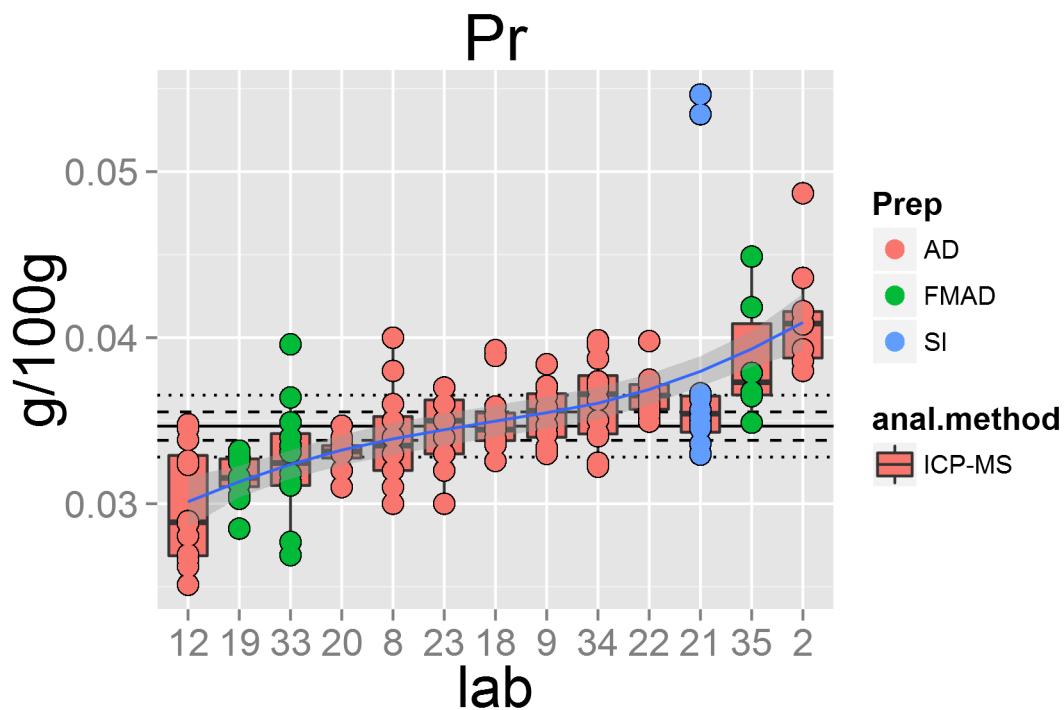
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



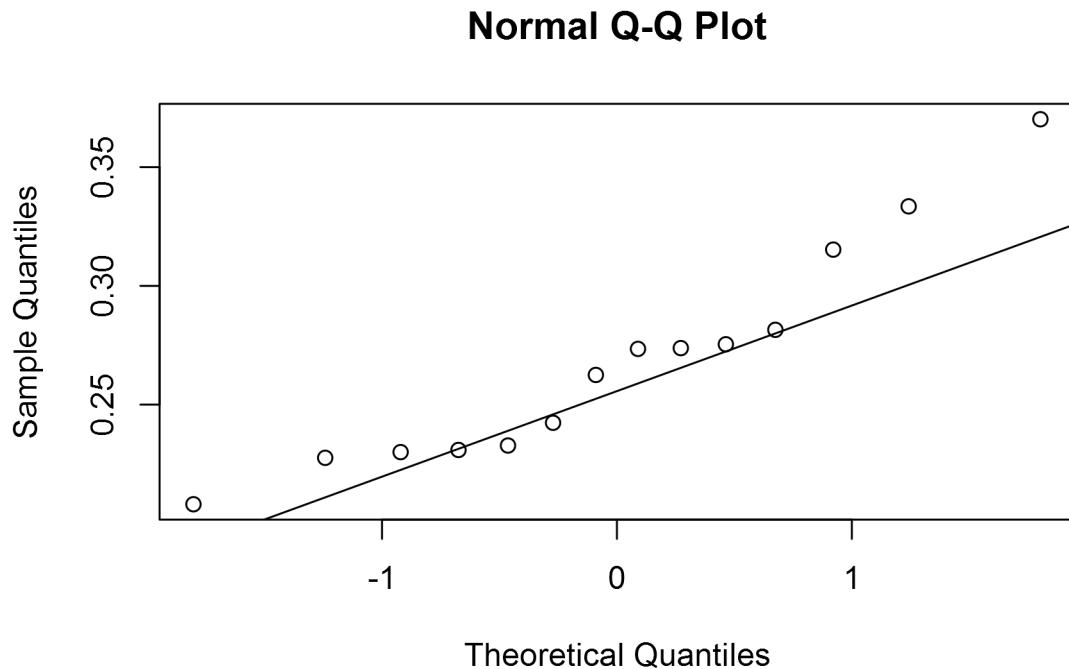
```
## [1] "Pr.1"
```



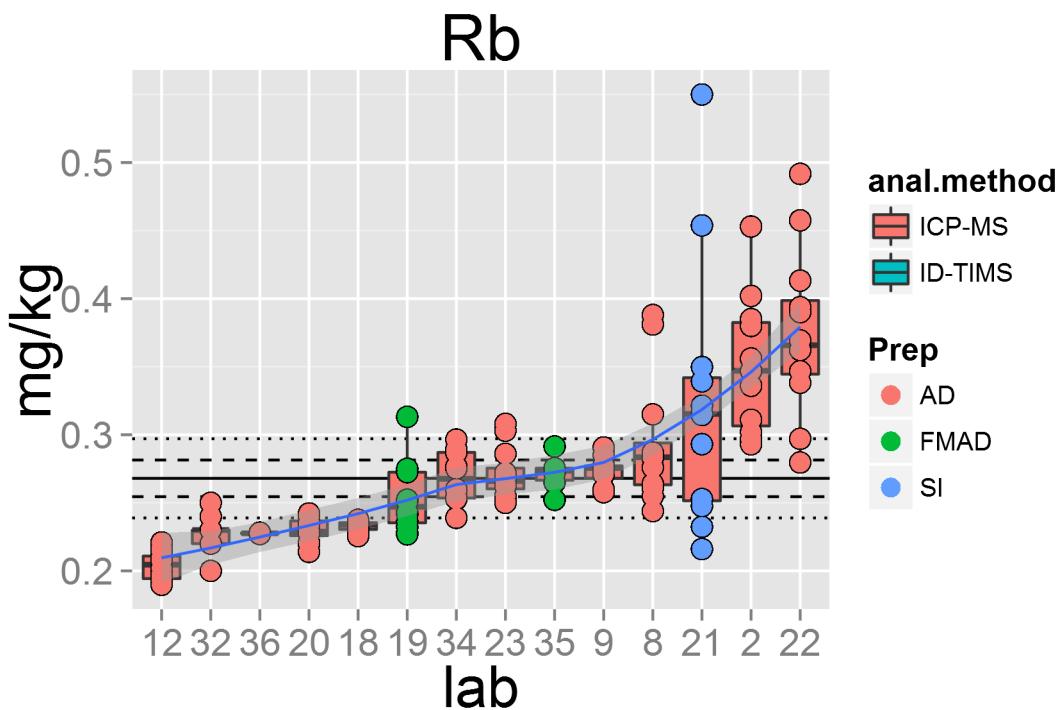
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



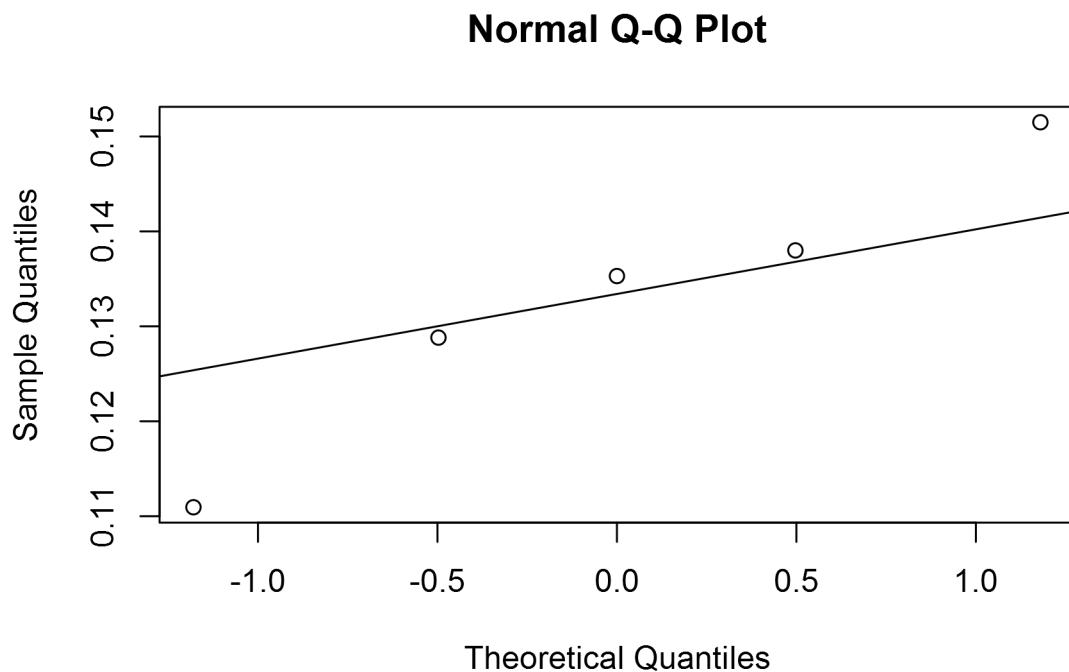
```
## [1] "Rb.1"
```



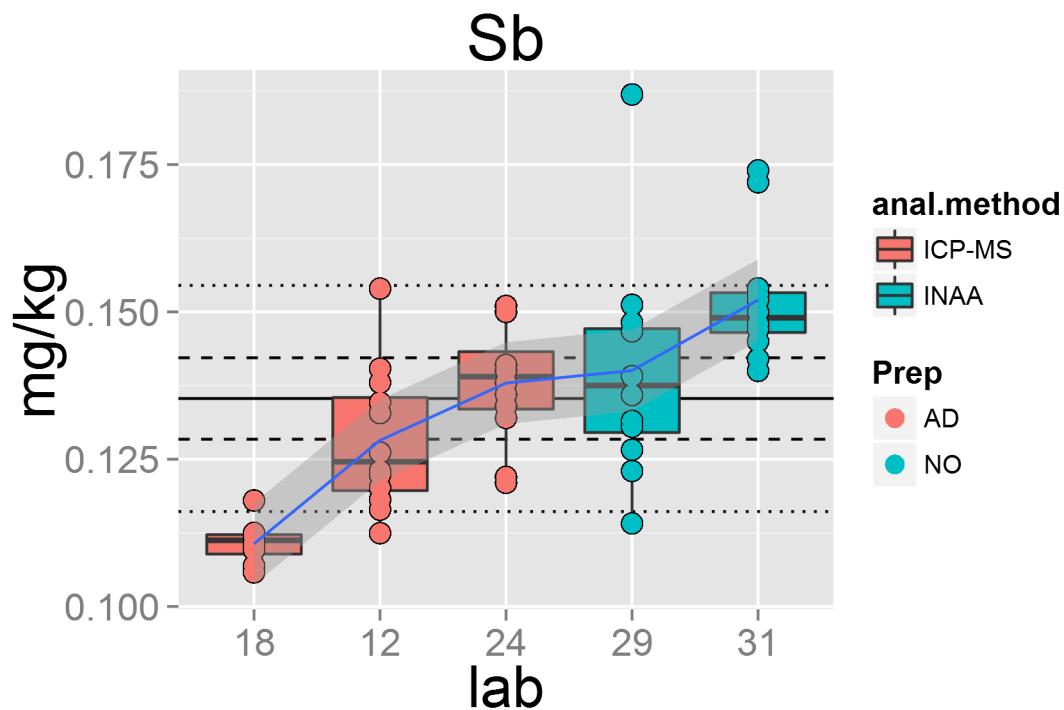
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



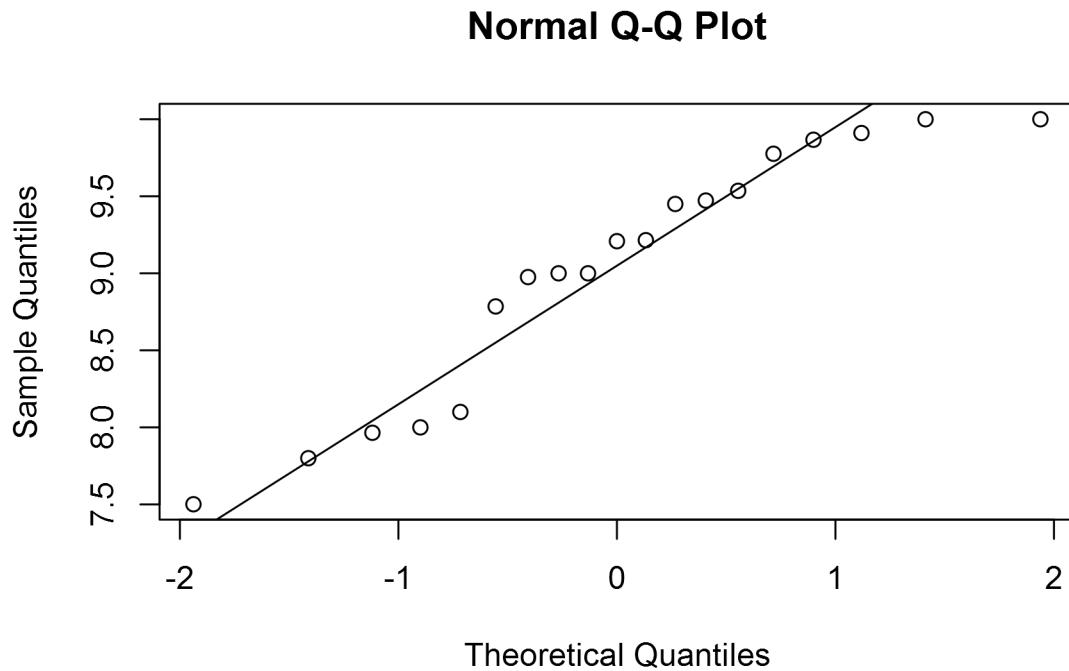
```
## [1] "Sb.1"
```



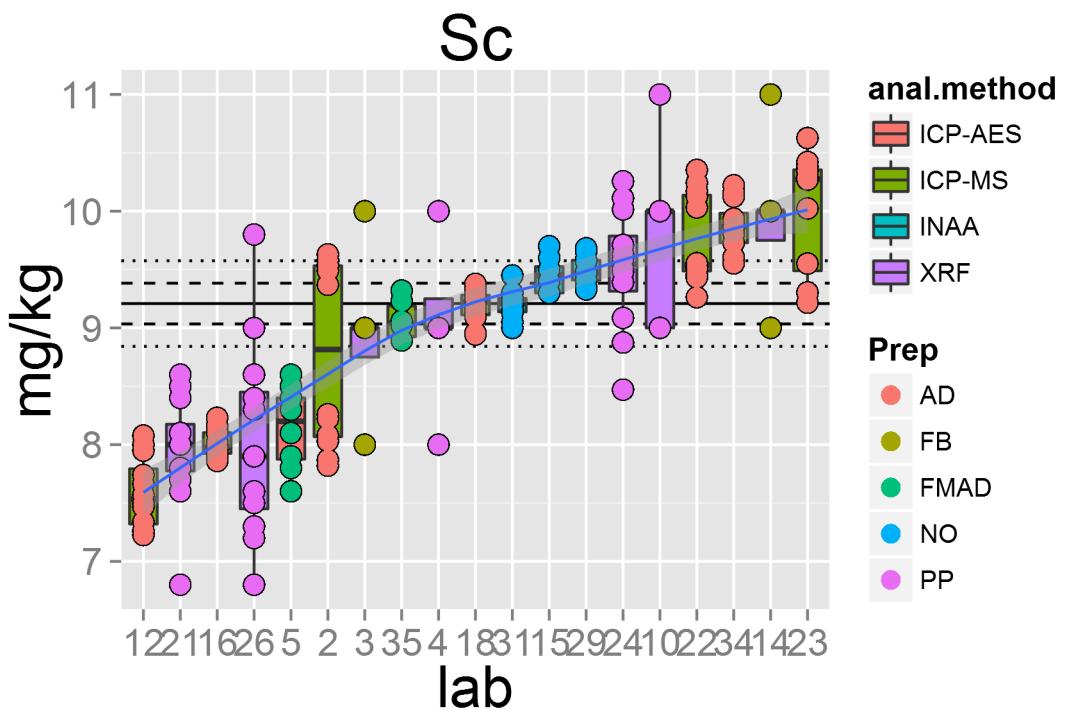
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



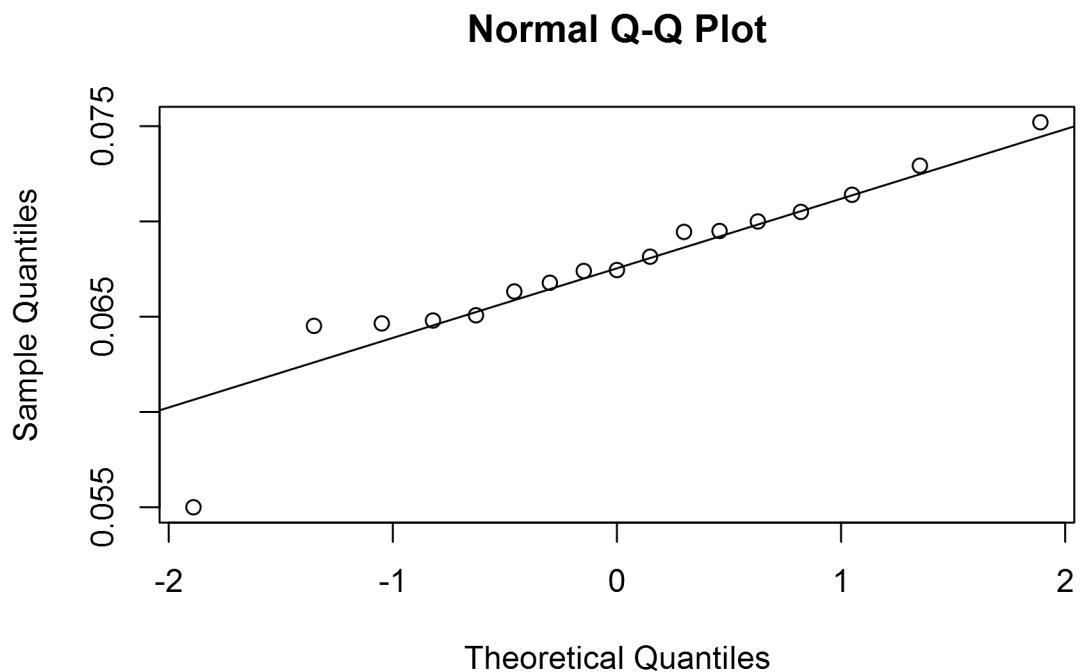
```
## [1] "Sc.1"
```



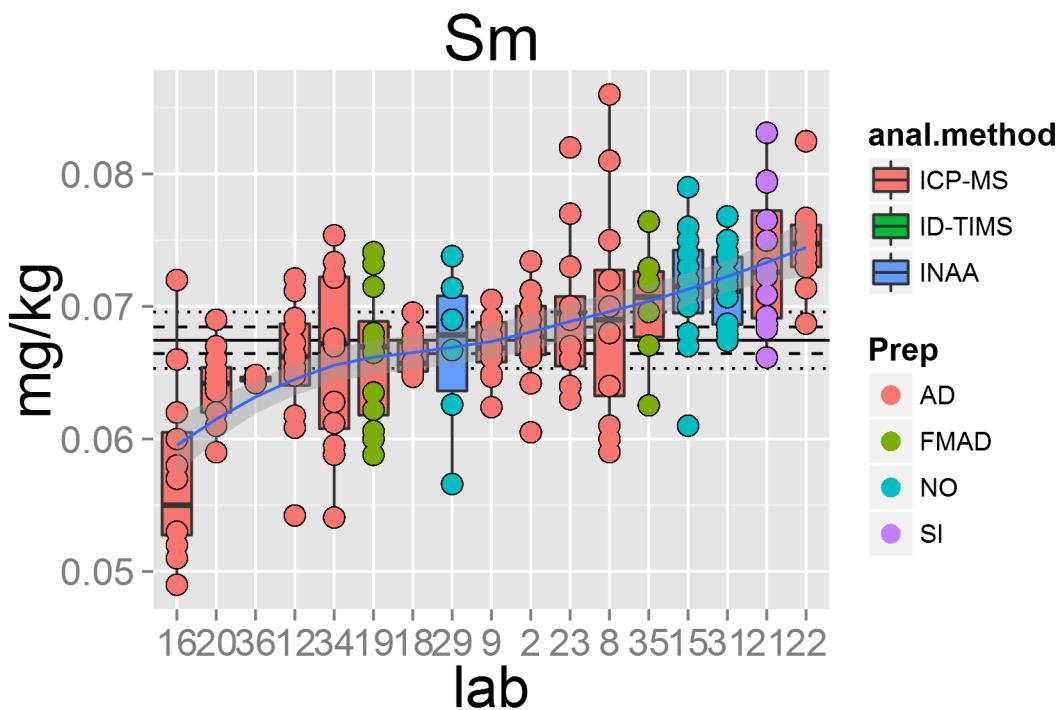
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



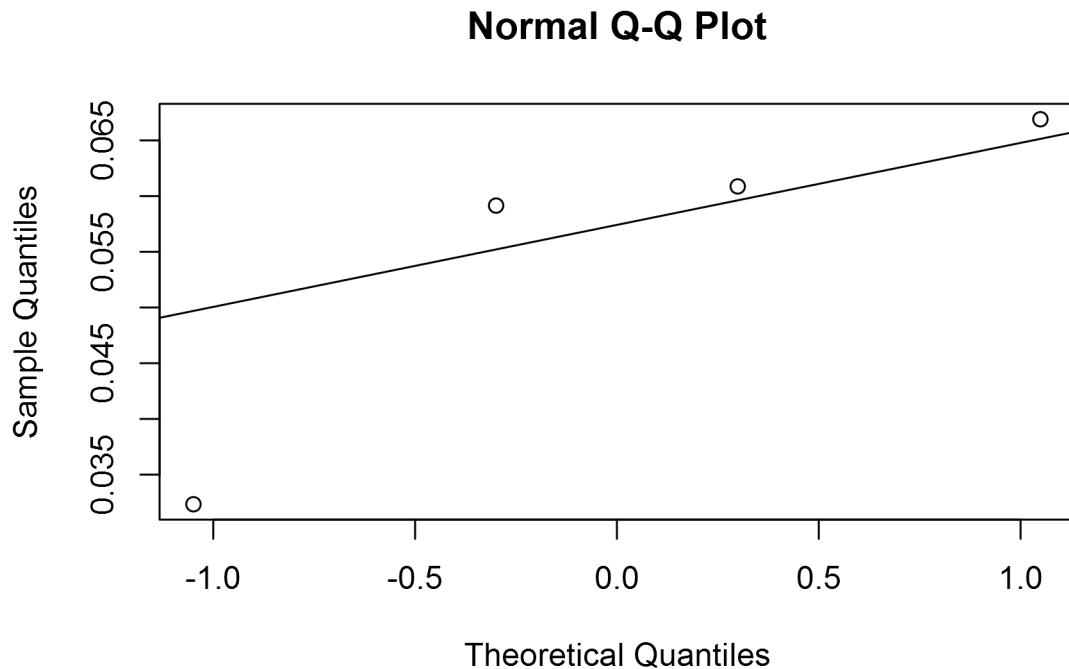
```
## [1] "Sm.1"
```



```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



```
## [1] "Sn.1"
```

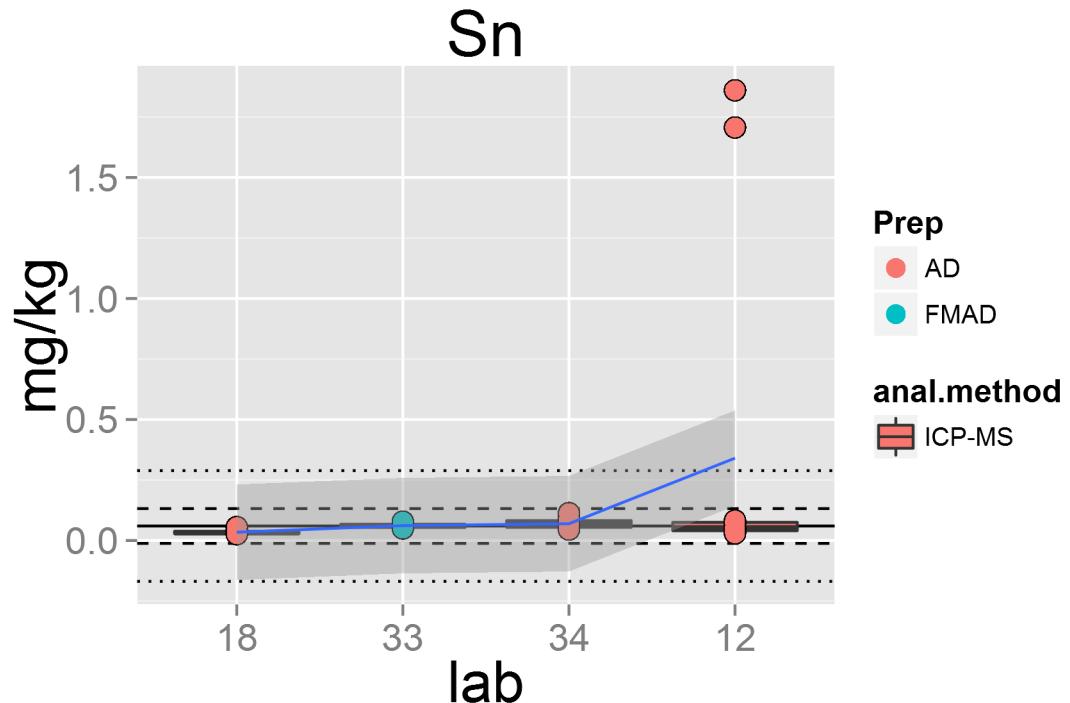


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```

```

## Warning: pseudoinverse used at 0.985
## Warning: neighborhood radius 2.015
## Warning: reciprocal condition number 3.0447e-016
## Warning: There are other near singularities as well. 4.0602
## Warning: pseudoinverse used at 0.985
## Warning: neighborhood radius 2.015
## Warning: reciprocal condition number 3.0447e-016
## Warning: There are other near singularities as well. 4.0602

```

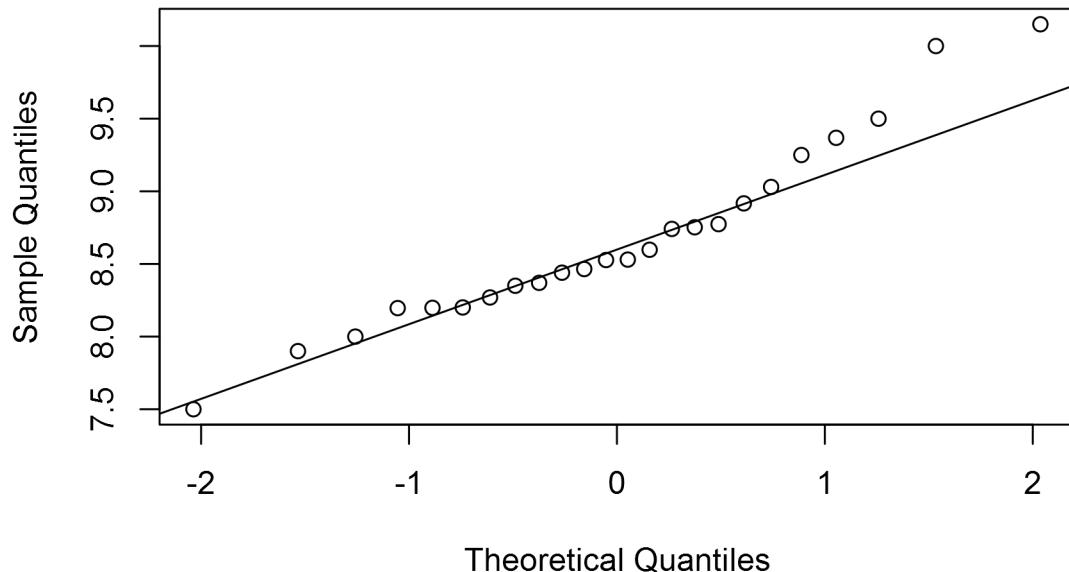


```

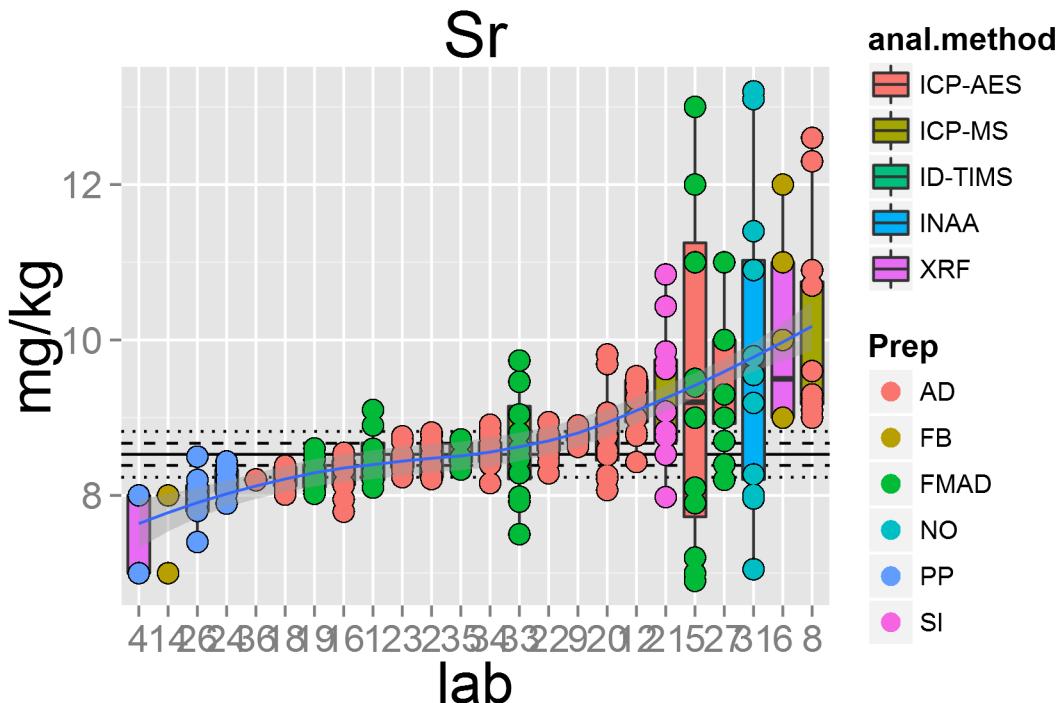
## [1] "Sr.1"

```

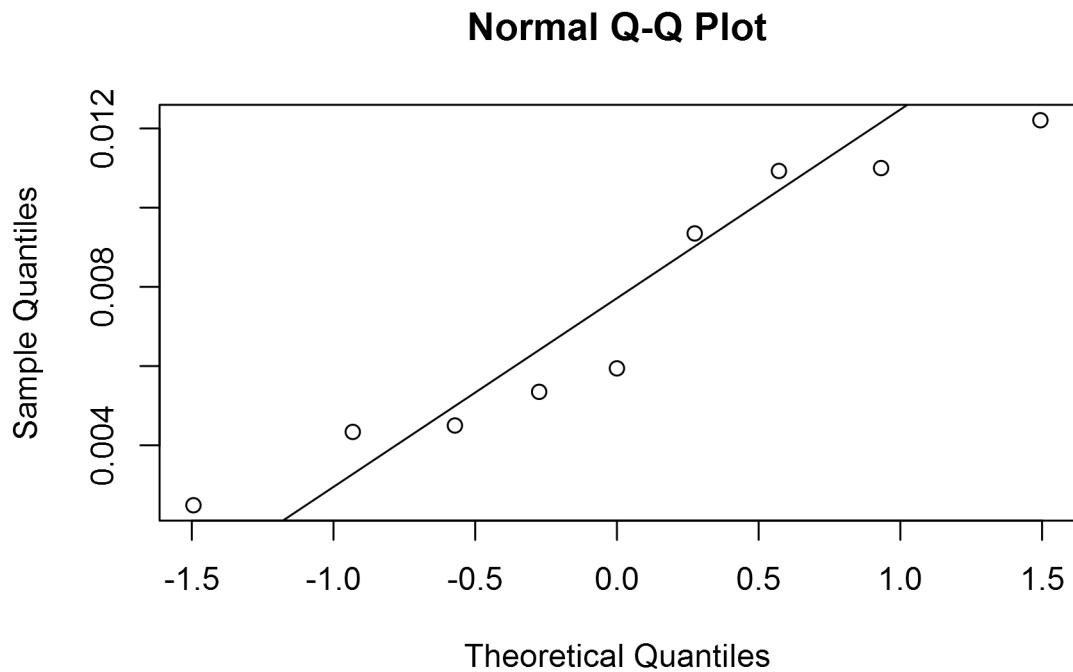
Normal Q-Q Plot



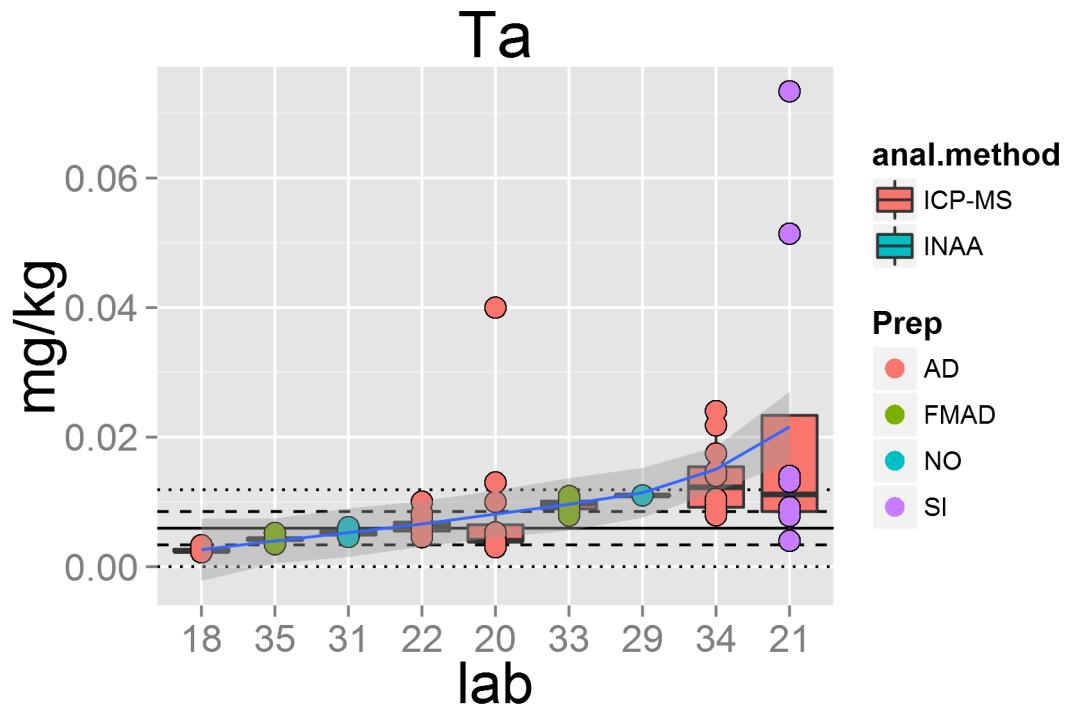
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



```
## [1] "Ta.1"
```

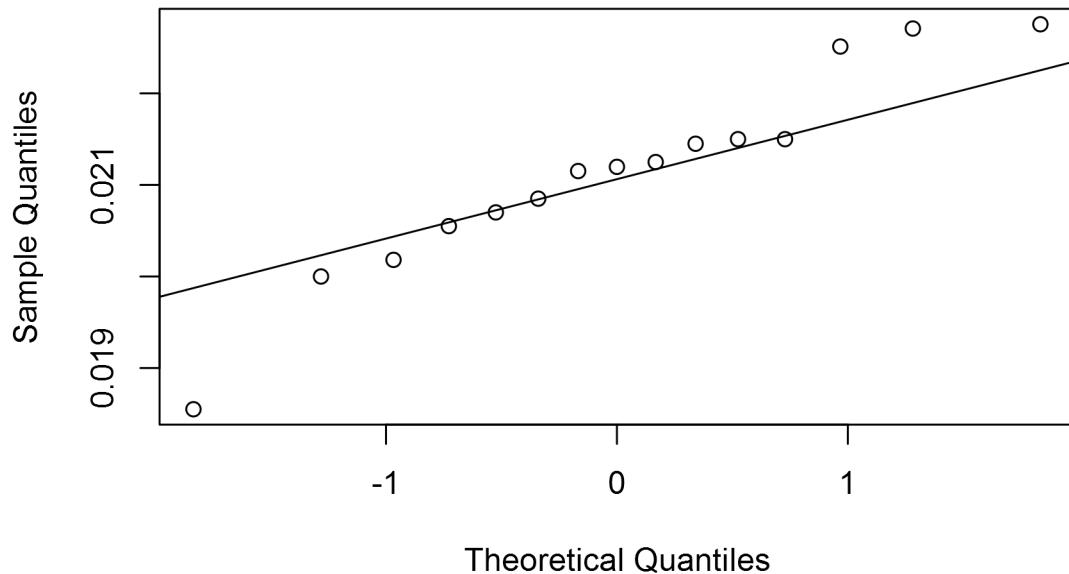


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```

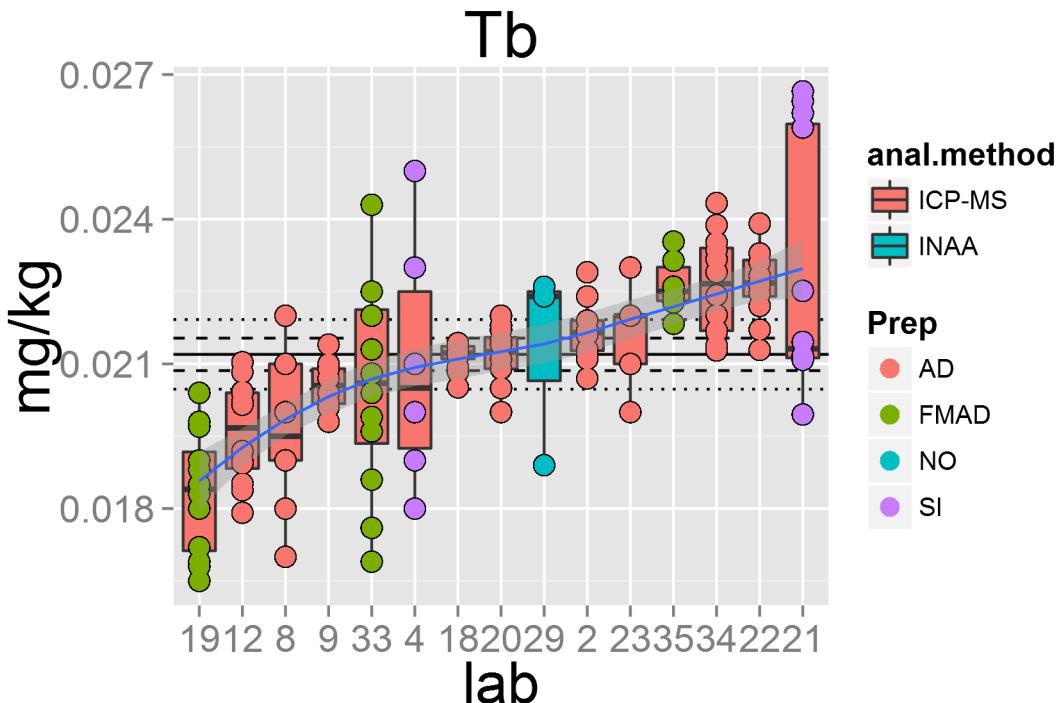


```
## [1] "Tb.1"
```

Normal Q-Q Plot

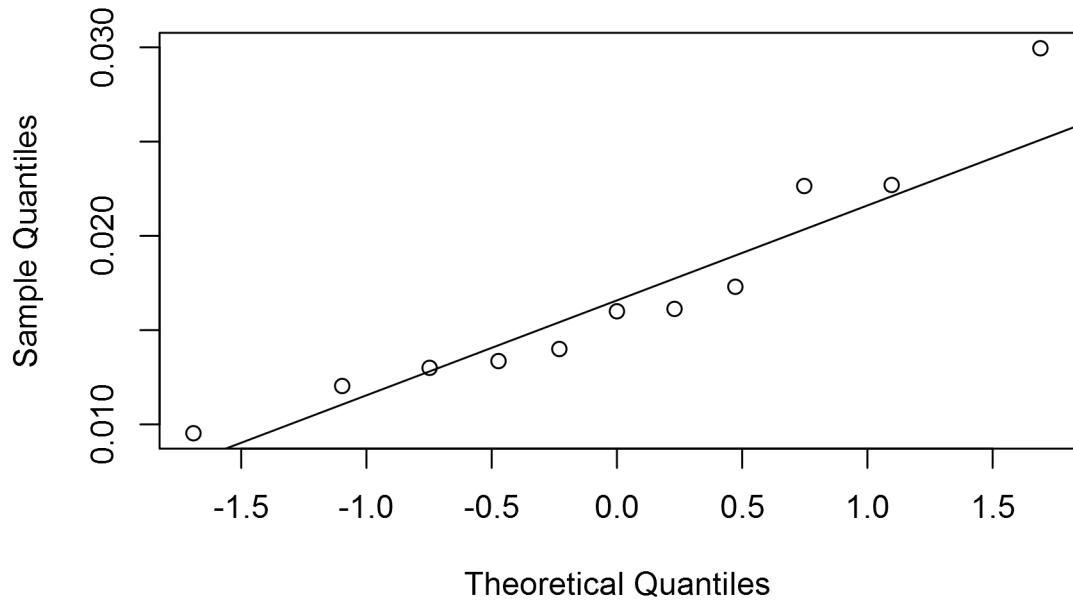


```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```

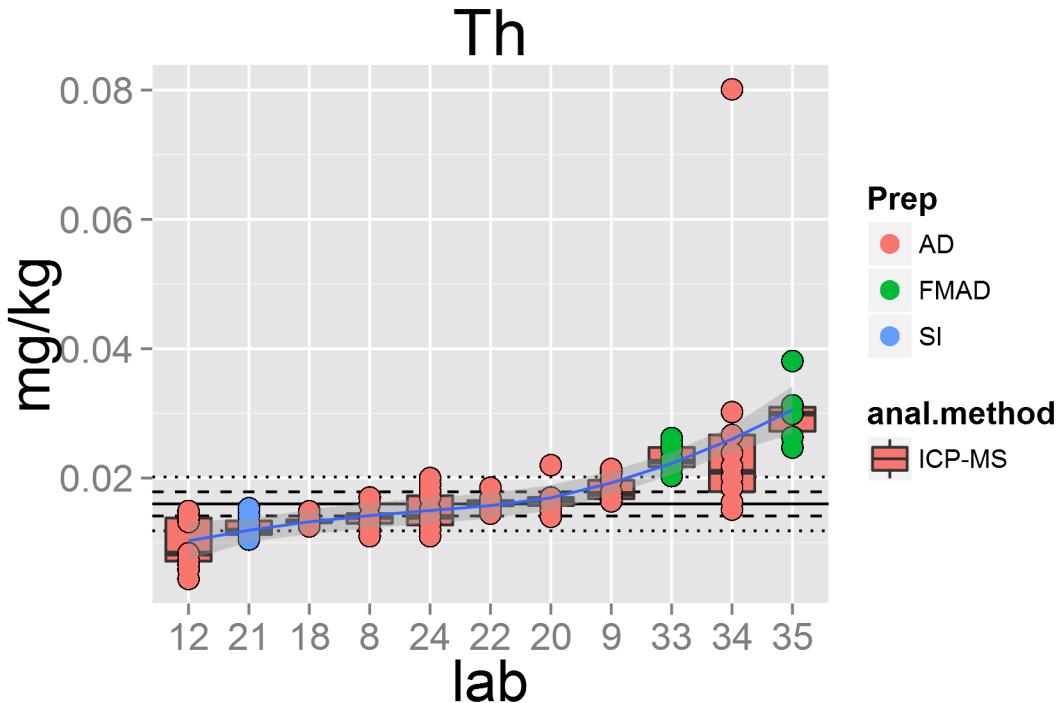


```
## [1] "Th.1"
```

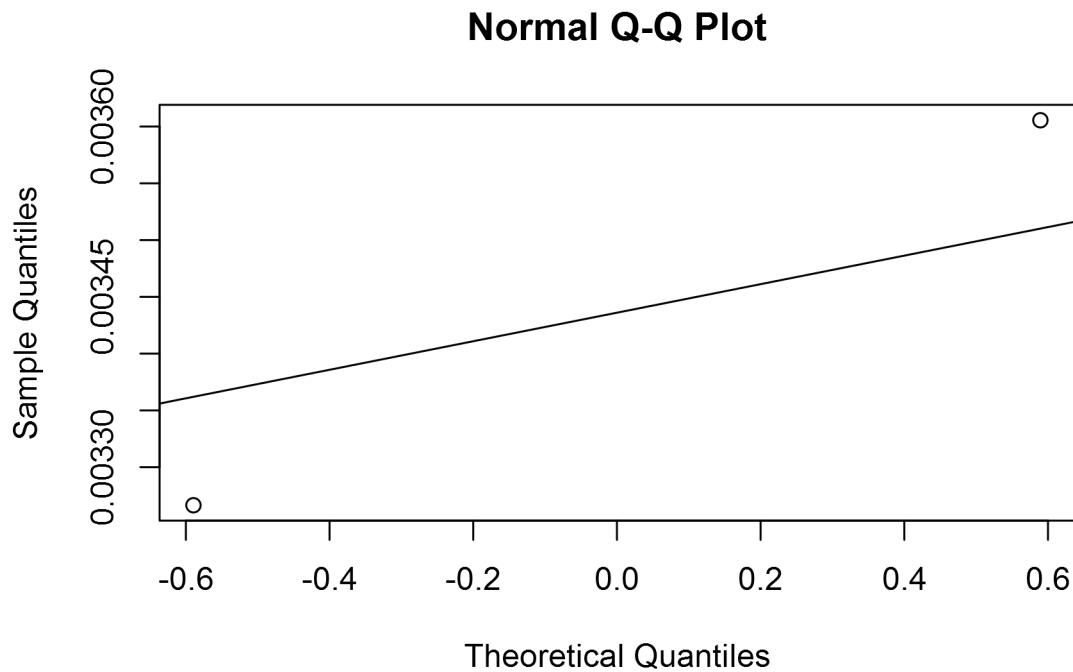
Normal Q-Q Plot



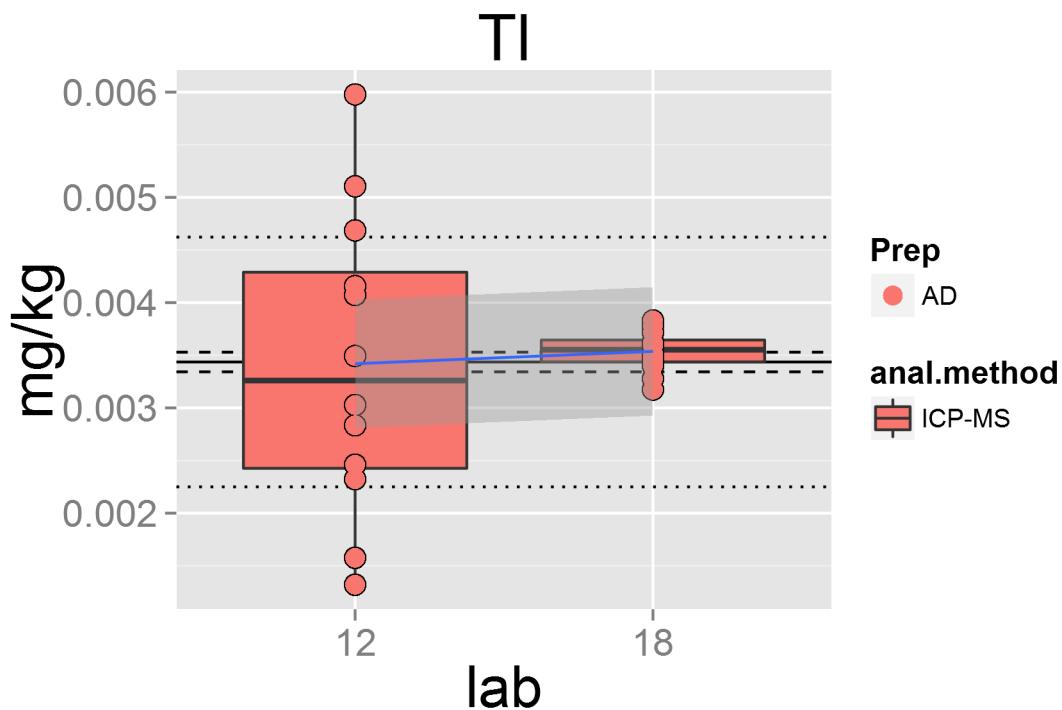
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to c
```



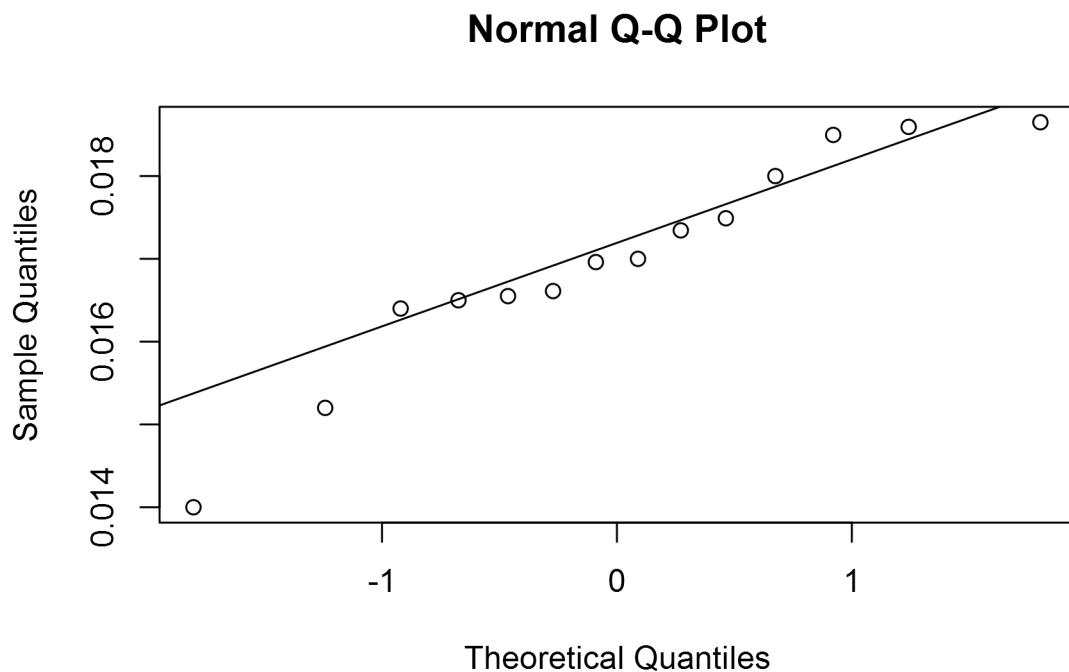
```
## [1] "T1.1"
```



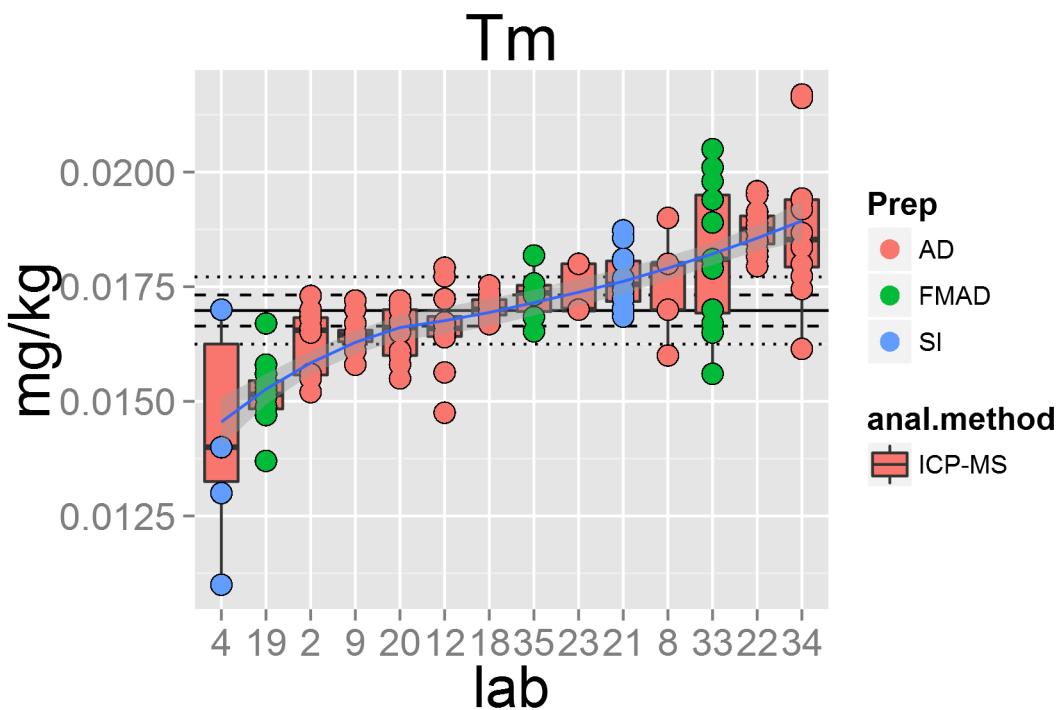
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to c  
## Warning: pseudoinverse used at 0.995  
## Warning: neighborhood radius 1.005  
## Warning: reciprocal condition number  0  
## Warning: There are other near singularities as well. 1.01  
## Warning: pseudoinverse used at 0.995  
## Warning: neighborhood radius 1.005  
## Warning: reciprocal condition number  0  
## Warning: There are other near singularities as well. 1.01
```



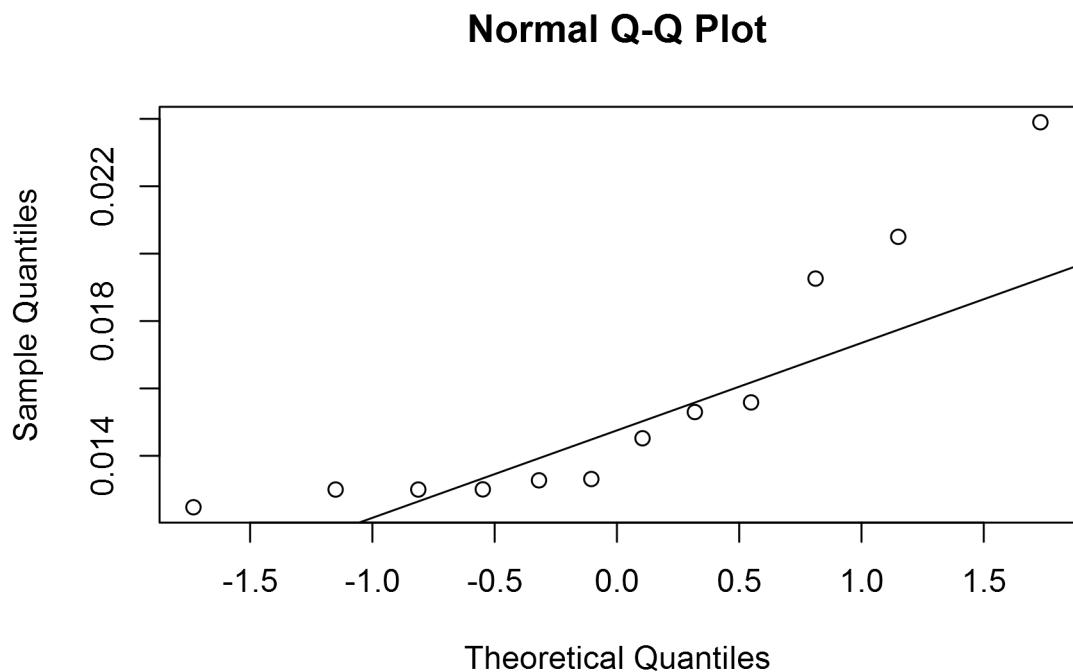
```
## [1] "Tm.1"
```



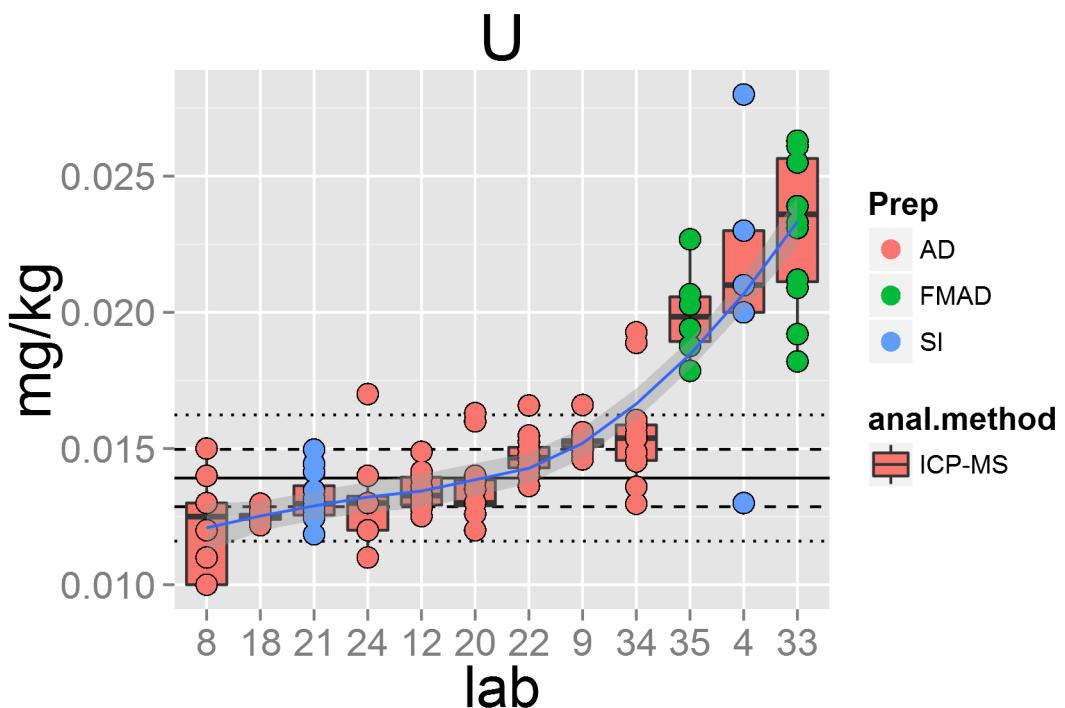
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



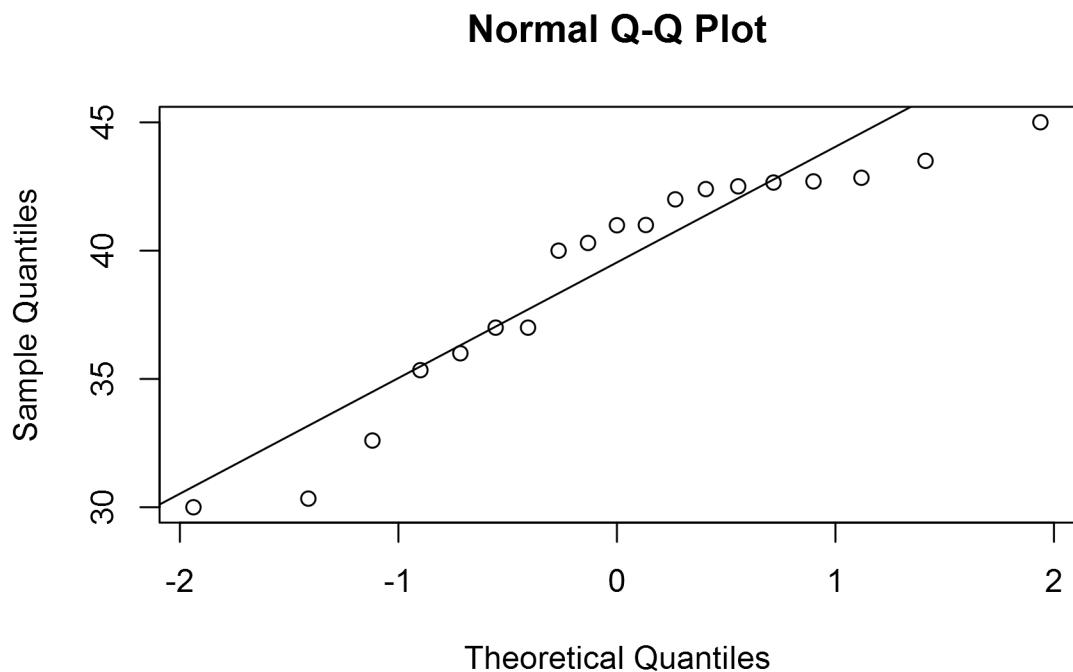
```
## [1] "U.1"
```



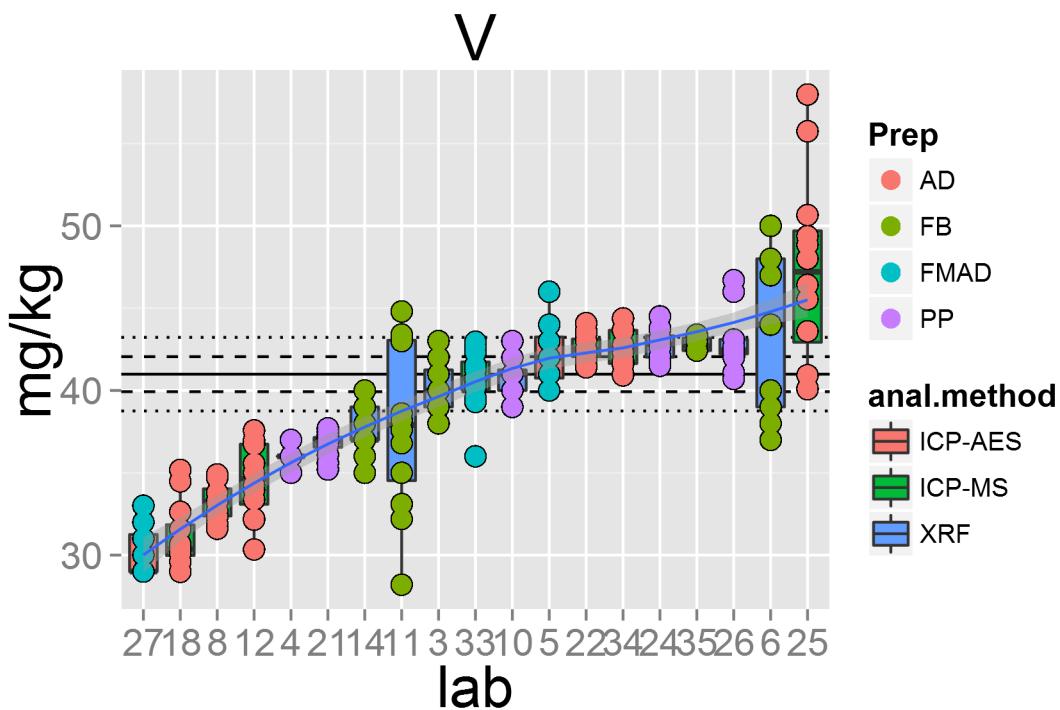
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



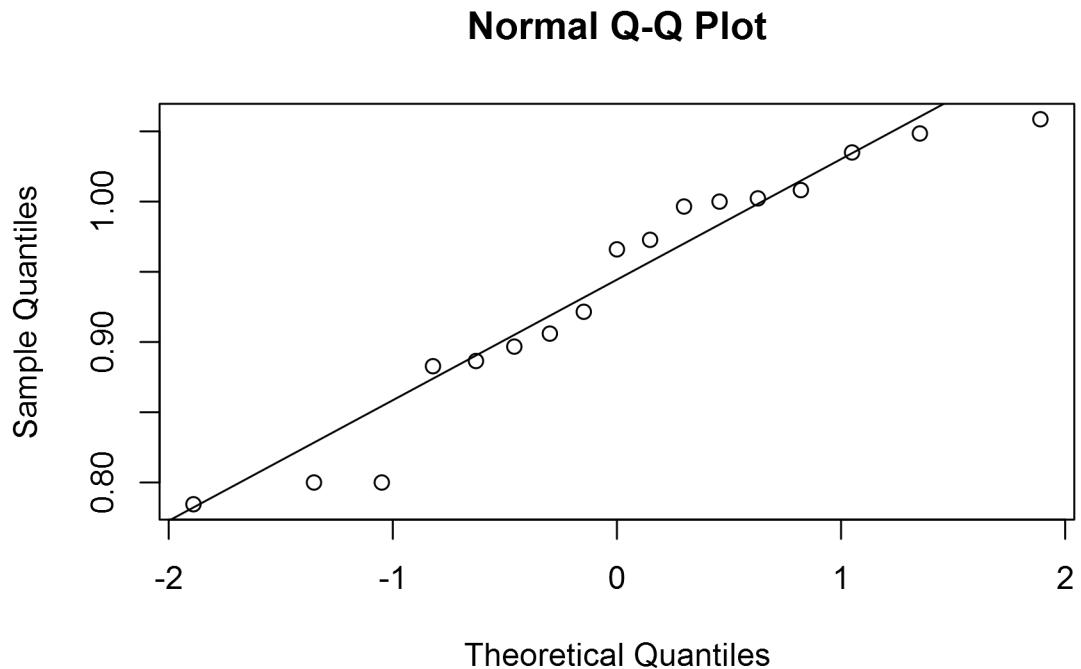
```
## [1] "V.1"
```



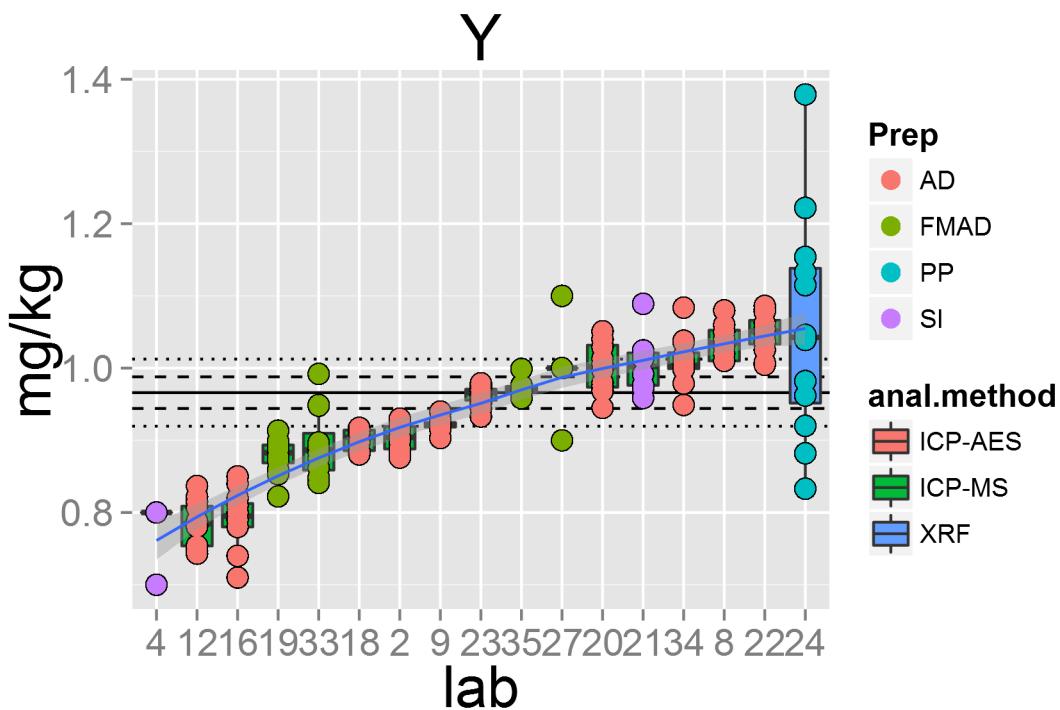
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



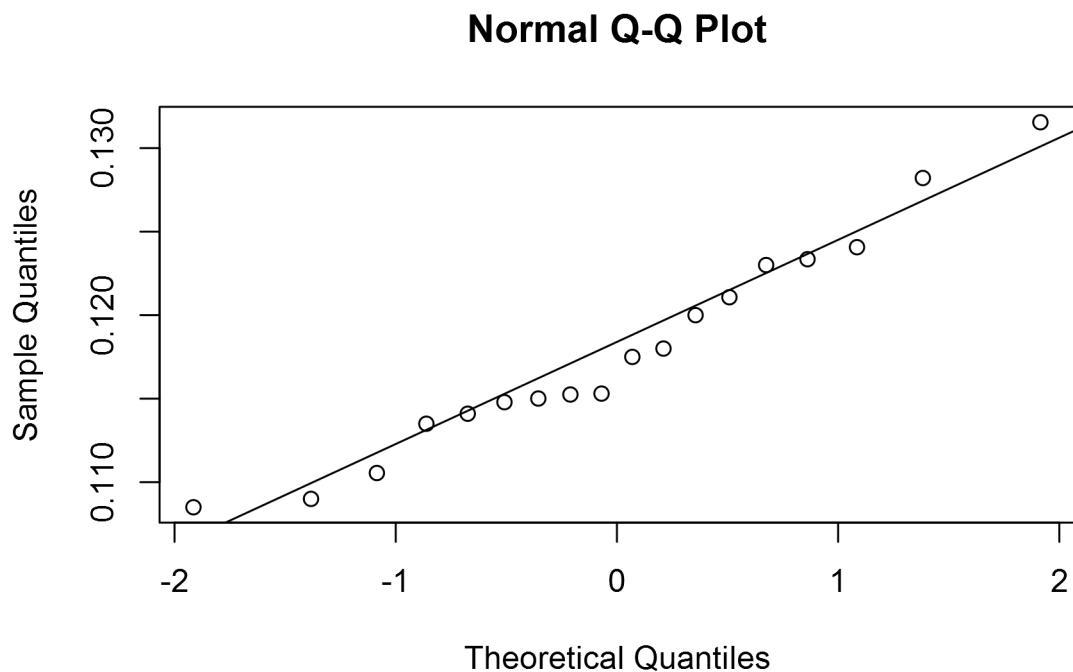
```
## [1] "Y.1"
```



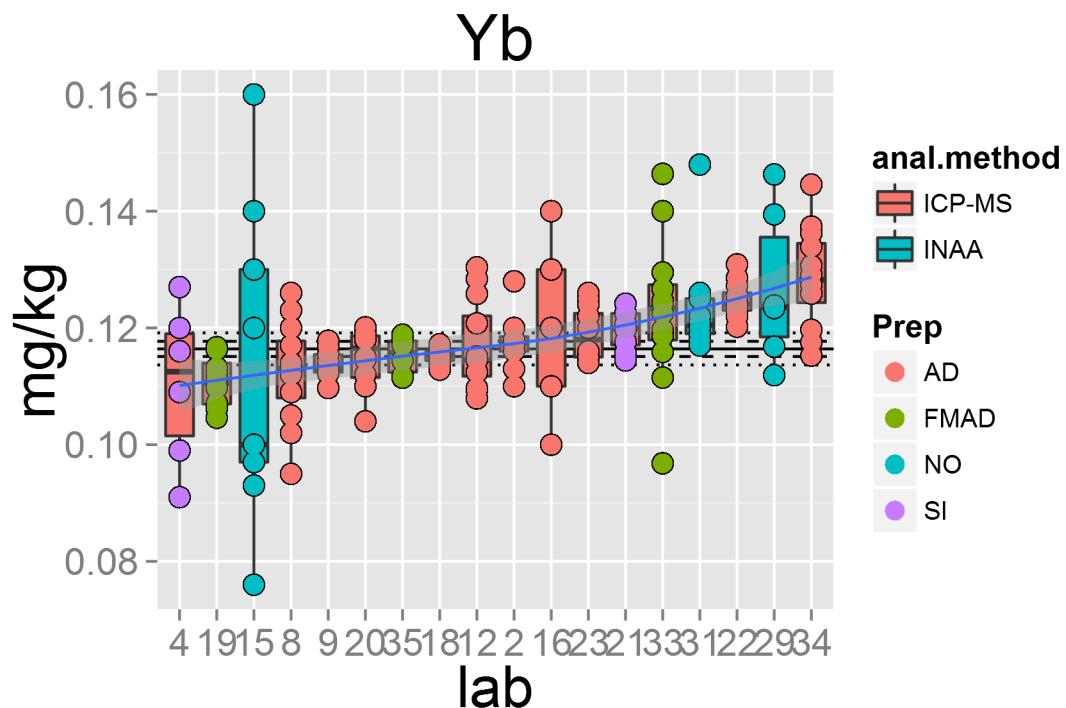
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



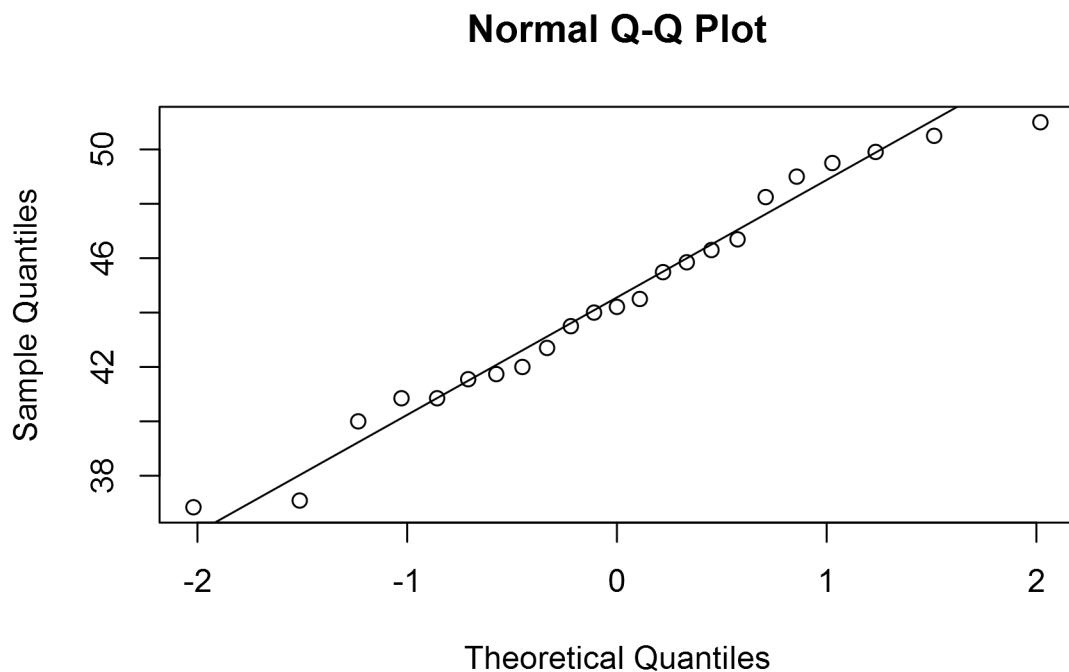
```
## [1] "Yb.1"
```



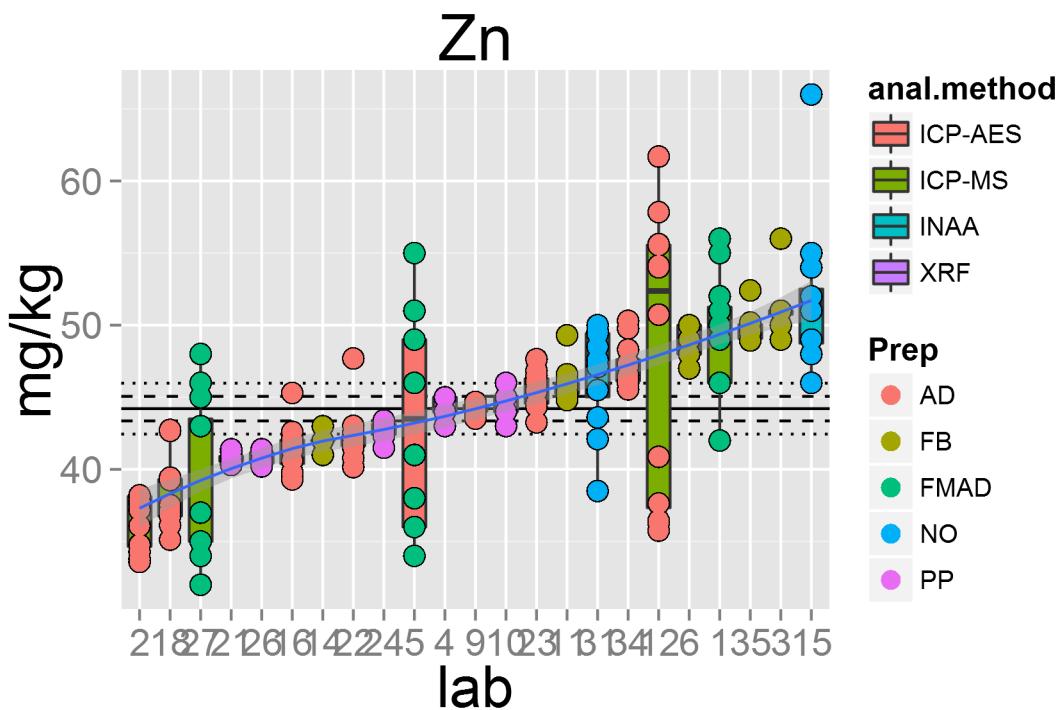
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



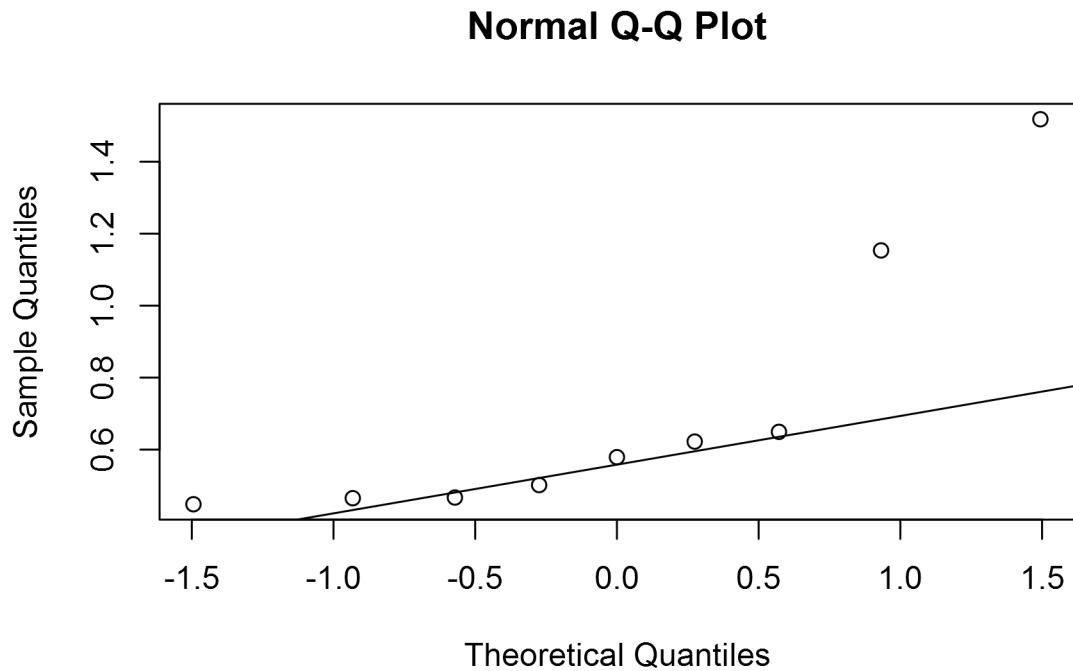
```
## [1] "Zn.1"
```



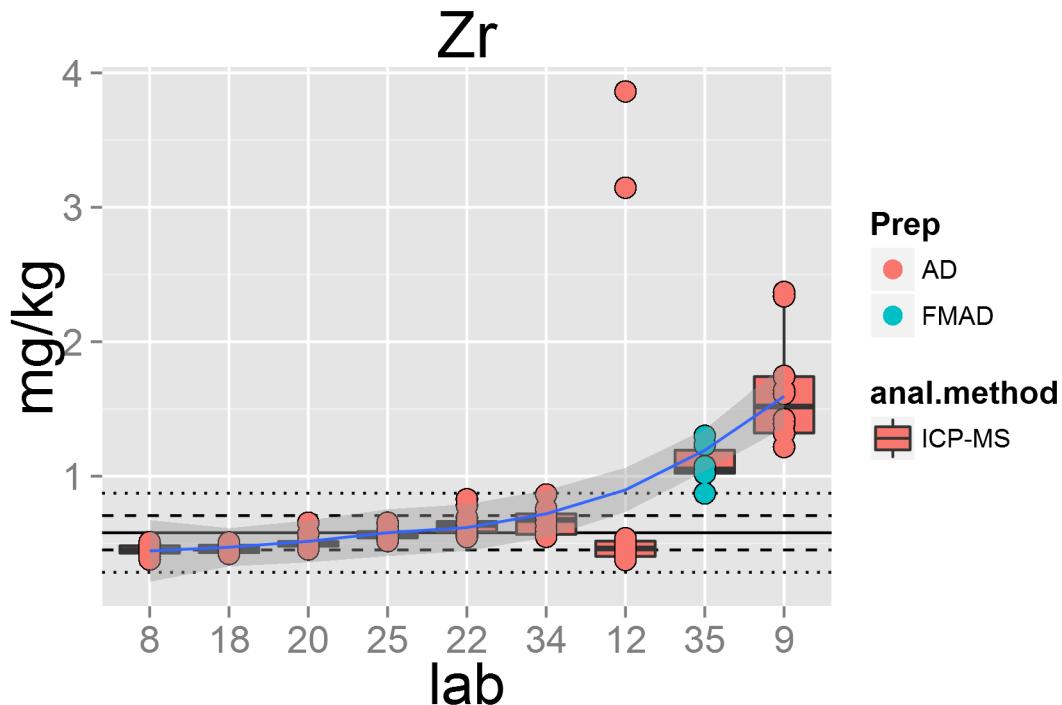
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



```
## [1] "Zr.1"
```



```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change this.
```



```
df1 <- read.table("~/GitHub/GOMcertification/df1.txt", header=T, quote="")
final <- ddply(df1, c("date", "RM", "measurand", "based.on", "unit"), numcolwise(meanGOM))
write.csv(final, "MUHall.csv")
```

```
# kable(final, format = "markdown", padding=0, digits=c(0,0,0,0,3,3,3,3,3,3,3,4,4,4,0,0,3,3,0))
#library(xtable)
#xt <- xtable(final, digits=c(0,0,0,0,0,3,3,3,3,3,3,3,3,3,3,2,0,0,3,3,0))
#print(xt, type="html")
#library(Gmisc)
# htmlTable(final)
```

```
certified.values <- data.frame(final$date, final$RM, final$measurand, final$t.value, final$lab.remaining)
names(certified.values) <- c("date", "RM", "measurand", "t.value", "n", "PV", "U", "unit")
write.table(certified.values, "CV1.txt", row.names=FALSE)
CV2 <- subset.data.frame(certified.values, n >= 10) # CV based on IAG protocol with n >= 10
CV3 <- subset.data.frame(certified.values, n < 10) # Information value based on IAG protocol with n < 10
write.table(CV2, "CV2.txt", row.names=FALSE) # just CV
write.table(CV3, "CV2.txt", row.names=FALSE, append=TRUE, col.names=FALSE) # CV and IV
CV <- read.table("~/GitHub/GOMcertification/CV2.txt", header=TRUE, quote="")
#xtCV <- xtable(CV, digits=c(0,0,0,0,2,0,4,4,0))
#print(xtCV, type="html")
kable(CV, digits=c(0,0,0,2,2,4,4,0), padding=1)
```

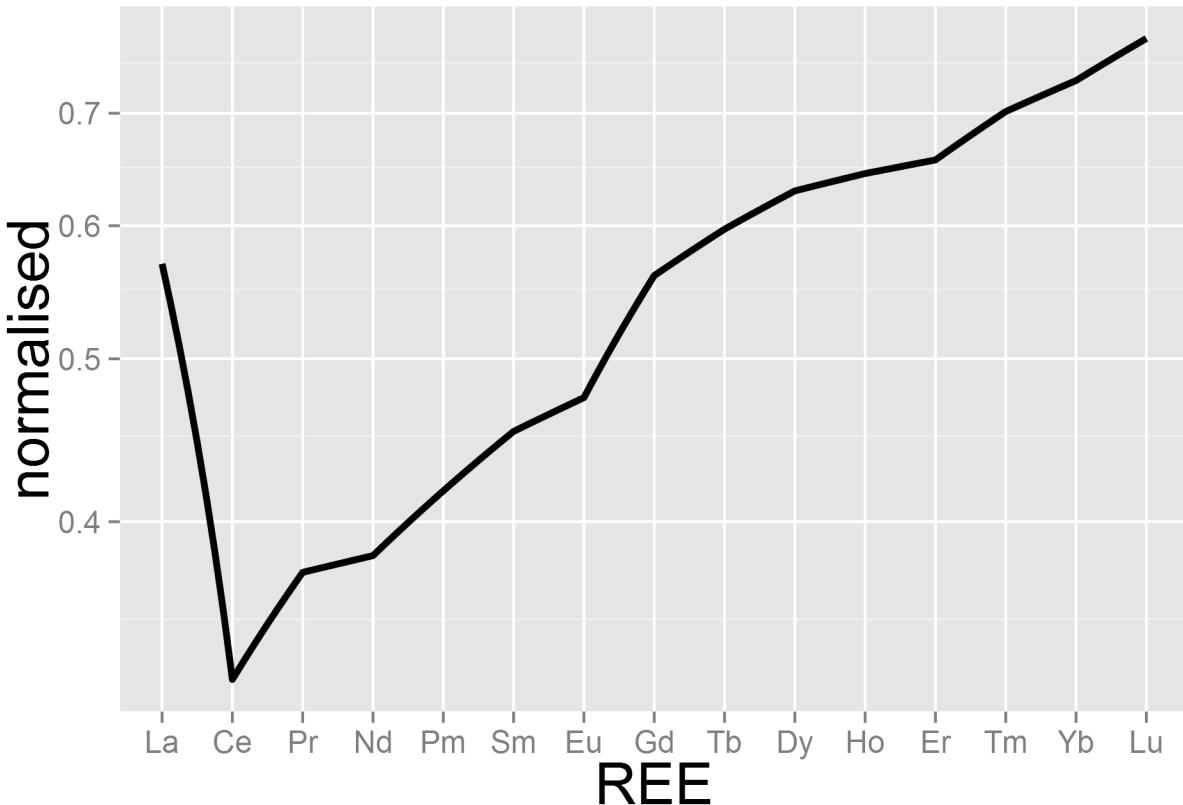
date	RM	measurand	t.value	n	PV	U	unit
2014-08-03	MUH	Al2O3	2.09	20	1.3300	0.0188	g/100g
2014-08-03	MUH	Ba	2.12	17	4.9050	0.2870	mg/kg
2014-08-03	MUH	CaO	2.09	21	1.2150	0.0104	g/100g
2014-08-03	MUH	Ce	2.16	14	0.1985	0.0166	mg/kg
2014-08-03	MUH	Co	2.08	22	106.9000	2.4500	mg/kg
2014-08-03	MUH	Cr	2.09	21	2706.0000	57.7000	mg/kg
2014-08-03	MUH	Cs	2.18	13	0.0976	0.0028	mg/kg
2014-08-03	MUH	Cu	2.12	17	18.8900	0.7020	mg/kg
2014-08-03	MUH	Dy	2.12	17	0.1542	0.0066	mg/kg
2014-08-03	MUH	Er	2.12	17	0.1090	0.0022	mg/kg
2014-08-03	MUH	Eu	2.14	15	0.0266	0.0010	mg/kg
2014-08-03	MUH	Fe2O3T	2.07	23	8.5850	0.0534	g/100g
2014-08-03	MUH	Ga	2.23	11	1.4000	0.0949	mg/kg
2014-08-03	MUH	Gd	2.20	12	0.1104	0.0044	mg/kg
2014-08-03	MUH	Hf	2.18	13	0.0374	0.0126	mg/kg
2014-08-03	MUH	Ho	2.18	13	0.0352	0.0015	mg/kg
2014-08-03	MUH	K2O	2.16	14	0.0109	0.0028	g/100g
2014-08-03	MUH	La	2.14	15	0.1344	0.0062	g/100g
2014-08-03	MUH	LOI	2.08	22	9.3350	0.0880	g/100g
2014-08-03	MUH	Lu	2.13	16	0.0190	0.0009	mg/kg
2014-08-03	MUH	MgO	2.07	23	38.2800	0.1630	g/100g
2014-08-03	MUH	MnO	2.09	21	0.1176	0.0014	g/100g
2014-08-03	MUH	Na2O	2.11	18	0.1000	0.0099	g/100g
2014-08-03	MUH	Nd	2.14	15	0.1745	0.0072	mg/kg
2014-08-03	MUH	Ni	2.09	21	2102.0000	26.2000	mg/kg
2014-08-03	MUH	P2O5	2.18	13	0.0081	0.0022	g/100g
2014-08-03	MUH	Pr	2.18	13	0.0347	0.0019	g/100g
2014-08-03	MUH	Rb	2.16	14	0.2680	0.0291	mg/kg
2014-08-03	MUH	Sc	2.10	19	9.2090	0.3670	mg/kg
2014-08-03	MUH	SiO2	2.08	22	40.3500	0.1700	g/100g
2014-08-03	MUH	Sm	2.12	17	0.0674	0.0021	mg/kg
2014-08-03	MUH	Sr	2.07	24	8.5290	0.2950	mg/kg
2014-08-03	MUH	Tb	2.14	15	0.0212	0.0007	mg/kg
2014-08-03	MUH	Th	2.23	11	0.0160	0.0042	mg/kg
2014-08-03	MUH	TiO2	2.07	24	0.0346	0.0019	g/100g
2014-08-03	MUH	Tm	2.16	14	0.0170	0.0007	mg/kg
2014-08-03	MUH	U	2.20	12	0.0139	0.0023	mg/kg

date	RM	measurand	t.value	n	PV	U	unit
2014-08-03	MUH	V	2.10	19	40.9900	2.2400	mg/kg
2014-08-03	MUH	Y	2.12	17	0.9660	0.0465	mg/kg
2014-08-03	MUH	Yb	2.11	18	0.1164	0.0028	mg/kg
2014-08-03	MUH	Zn	2.07	23	44.2100	1.7700	mg/kg
2014-08-03	MUH	As	2.36	8	3.5820	0.7380	mg/kg
2014-08-03	MUH	Be	4.30	3	0.0124	0.0042	mg/kg
2014-08-03	MUH	CO2	12.71	2	1.0210	0.6850	g/100g
2014-08-03	MUH	FeO	4.30	3	3.3000	1.6100	g/100g
2014-08-03	MUH	H2O.	3.18	4	0.1162	0.1250	g/100g
2014-08-03	MUH	Li	2.45	7	1.6780	0.4650	mg/kg
2014-08-03	MUH	Nb	2.36	8	0.0585	0.0224	mg/kg
2014-08-03	MUH	Pb	2.36	8	0.4356	0.0652	mg/kg
2014-08-03	MUH	Sb	2.78	5	0.1353	0.0192	mg/kg
2014-08-03	MUH	Sn	3.18	4	0.0600	0.2290	mg/kg
2014-08-03	MUH	Ta	2.31	9	0.0059	0.0059	mg/kg
2014-08-03	MUH	Tl	12.71	2	0.0034	0.0012	mg/kg
2014-08-03	MUH	Zr	2.31	9	0.5790	0.2940	mg/kg

```

REE.chondrites <- read.csv("~/GitHub/REE/REE.chondrites.csv", sep=";") # reading chondrite normalising values
REE.chondrites <- rename(REE.chondrites, c("norm"= "REE")) # renaming the column to make it suitable for merging
REE <- c("La", "Ce", "Pr", "Nd", "Pm", "Sm", "Eu", "Gd", "Tb", "Dy", "Ho", "Er", "Tm", "Yb", "Lu") # needed elements
RM <- read.table("~/GitHub/GOMcertification/CV2.txt", header=TRUE, quote="\") # reading all the finalised REE values
RM.REE <- subset(RM, measurand %in% REE, select=c(RM,measurand,PV)) # extracting only the REE values
RM.REE <- rename(RM.REE, c("measurand"="REE")) # renaming the column head suitable for merging
RM.REE <- merge(REE.chondrites, RM.REE, by = "REE", all.x=TRUE) # merging the data set so that element names are common
RM.normalised <- ddply(RM.REE, c("RM"), transform, normalised = PV/chondrite) # adding a chondrite normalised value
RM.normalised[15,5] <- (RM.normalised[9,5]+RM.normalised[11,5])/2
REEtheme <- theme_grey() + theme(plot.title = element_text(colour = "black", size = rel(2)))+ theme(axis.ticks=element_line(size=1))
p <- ggplot(data=RM.normalised, aes(x=REE, y=normalised, group=1)) + geom_line(size= 1.2)
p + scale_x_discrete(limits=c("La", "Ce", "Pr", "Nd", "Pm", "Sm", "Eu", "Gd", "Tb", "Dy", "Ho", "Er", "Tm"))

```



final

```
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
```