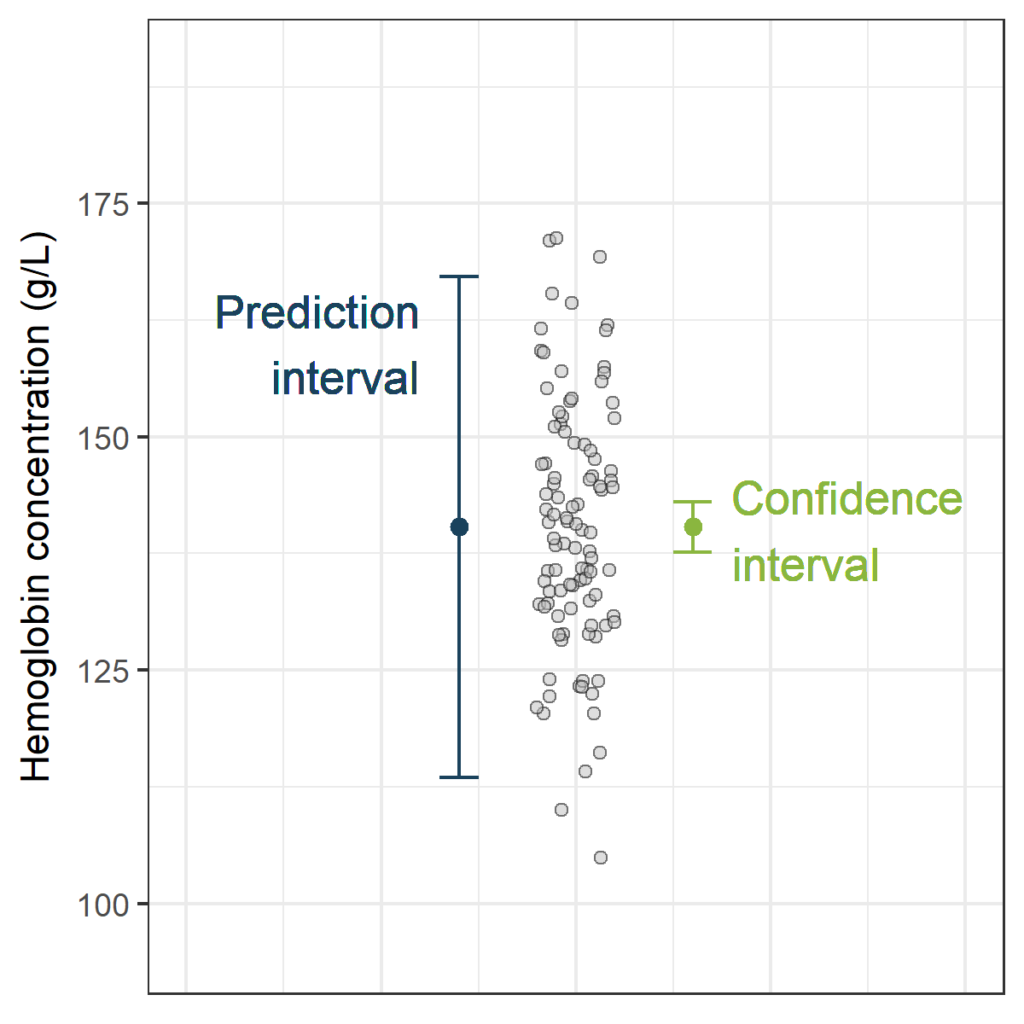
**Prediction interval versus Confidence interval**

Very often a confidence interval is misinterpreted as a prediction interval, leading to unrealistic “precise” predictions. As you will see, prediction intervals (PI) resemble confidence intervals (CI), but the width of the PI is by definition larger than the width of the CI.

Let’s assume that we measure the whole blood hemoglobin concentration in a random sample of 100 persons. We obtain the estimated mean (Est\_mean), limits of the confidence interval (CI\_Lower and CI\_Upper) and limits of the prediction interval (PI\_Lower and PI\_Upper): (The R-code to do this is in the next paragraph)

| **Est\_mean** | **CI\_Lower** | **CI\_Upper** | **PI\_Lower** | **PI\_Upper** |
| --- | --- | --- | --- | --- |
| 140 | 138 | 143 | 113 | 167 |

The graphical presentation:  
[](https://datascienceplus.com/wp-content/uploads/2018/06/PI_CI_n100-1.png)

A **Confidence interval** (CI) is an interval of good estimates of the unknown true **population parameter**. About a 95% confidence interval for the mean, we can state that if we would repeat our sampling process infinitely, 95% of the constructed confidence intervals would contain the true population mean. In other words, there is a 95% chance of selecting a sample such that the 95% confidence interval calculated from that sample contains the true population mean.

Interpretation of the 95% confidence interval in our example:

* The values contained in the interval [138g/L to 143g/L] are good estimates of the unknown **mean** whole blood hemoglobin concentration in the population. In general, if we would repeat our sampling process infinitely, 95% of such constructed confidence intervals would contain the true mean hemoglobin concentration.

A **Prediction interval** (PI) is an estimate of an interval in which **a future observation** will fall, with a certain confidence level, given the observations that were already observed. About a 95% prediction interval we can state that if we would repeat our sampling process infinitely, 95% of the constructed prediction intervals would contain the new observation.

Interpretation of the 95% prediction interval in the above example:

* Given the observed whole blood hemoglobin concentrations, the whole blood hemoglobin concentration of a **new sample** will be between 113g/L and 167g/L with a confidence of 95%. In general, if we would repeat our sampling process infinitely, 95% of the such constructed prediction intervals would contain the new hemoglobin concentration measurement.

Remark: Very often we will read the interpretation “The whole blood hemogblobin concentration of a new sample will be between 113g/L and 167g/L with a **probability** of 95%.” (for example on [wikipedia](https://en.wikipedia.org/wiki/Prediction_interval)). This interpretation is correct in the theoretical situation where the parameters (true mean and standard deviation) are known.

**Estimating a prediction interval in R**

First, let's simulate some data. The sample size in the plot above was (n=100). Now, to see the effect of the sample size on the width of the confidence interval and the prediction interval, let's take a “sample” of 400 hemoglobin measurements using the same parameters:

set.seed(123)

hemoglobin<-rnorm(400, mean = 139, sd = 14.75)

df<-data.frame(hemoglobin)

Although we don't need a linear regression yet, I'd like to use the lm() function, which makes it very easy to construct a confidence interval (CI) and a prediction interval (PI). We can estimate the mean by fitting a “regression model” with an intercept only (no slope). The default confidence level is 95%.

**Confidence interval:**

CI<-predict(lm(df$hemoglobin~ 1), interval="confidence")

CI[1,]

*## fit lwr upr*

*## 139.2474 137.8425 140.6524*

The CI object has a length of 400. But since there is no slope in our “model”, each row is exactly the same.

**Prediction interval:**

PI<-predict(lm(df$hemoglobin~ 1), interval="predict")

*## Warning in predict.lm(lm(df$hemoglobin ~ 1), interval = "predict"): predictions on current data refer to \_future\_ responses*

PI[1,]

*## fit lwr upr*

*## 139.2474 111.1134 167.3815*

We get a “warning” that “predictions on current data refer to future responses”. That's exactly what we want, so no worries there. As you see, the column names of the objects CI and PI are the same. Now, let's visualize the confidence and the prediction interval.

The code below is not very elegant but I like the result (tips are welcome :-))

library(ggplot2)

limits\_CI <- aes(x=1.3 , ymin=CI[1,2], ymax =CI[1,3])

limits\_PI <- aes(x=0.7 , ymin=PI[1,2], ymax =PI[1,3])

PI\_CI<-ggplot(df, aes(x=1, y=hemoglobin)) +

geom\_jitter(width=0.1, pch=21, fill="grey", alpha=0.5) +

geom\_errorbar (limits\_PI, width=0.1, col="#1A425C") +

geom\_point (aes(x=0.7, y=PI[1,1]), col="#1A425C", size=2) +

geom\_errorbar (limits\_CI, width=0.1, col="#8AB63F") +

geom\_point (aes(x=1.3, y=CI[1,1]), col="#8AB63F", size=2) +

scale\_x\_continuous(limits=c(0,2))+

scale\_y\_continuous(limits=c(95,190))+

theme\_bw()+ylab("Hemoglobin concentration (g/L)") +

xlab(NULL)+

geom\_text(aes(x=0.6, y=160, label="Prediction\ninterval",

hjust="right", cex=2), col="#1A425C")+

geom\_text(aes(x=1.4, y=140, label="Confidence\ninterval",

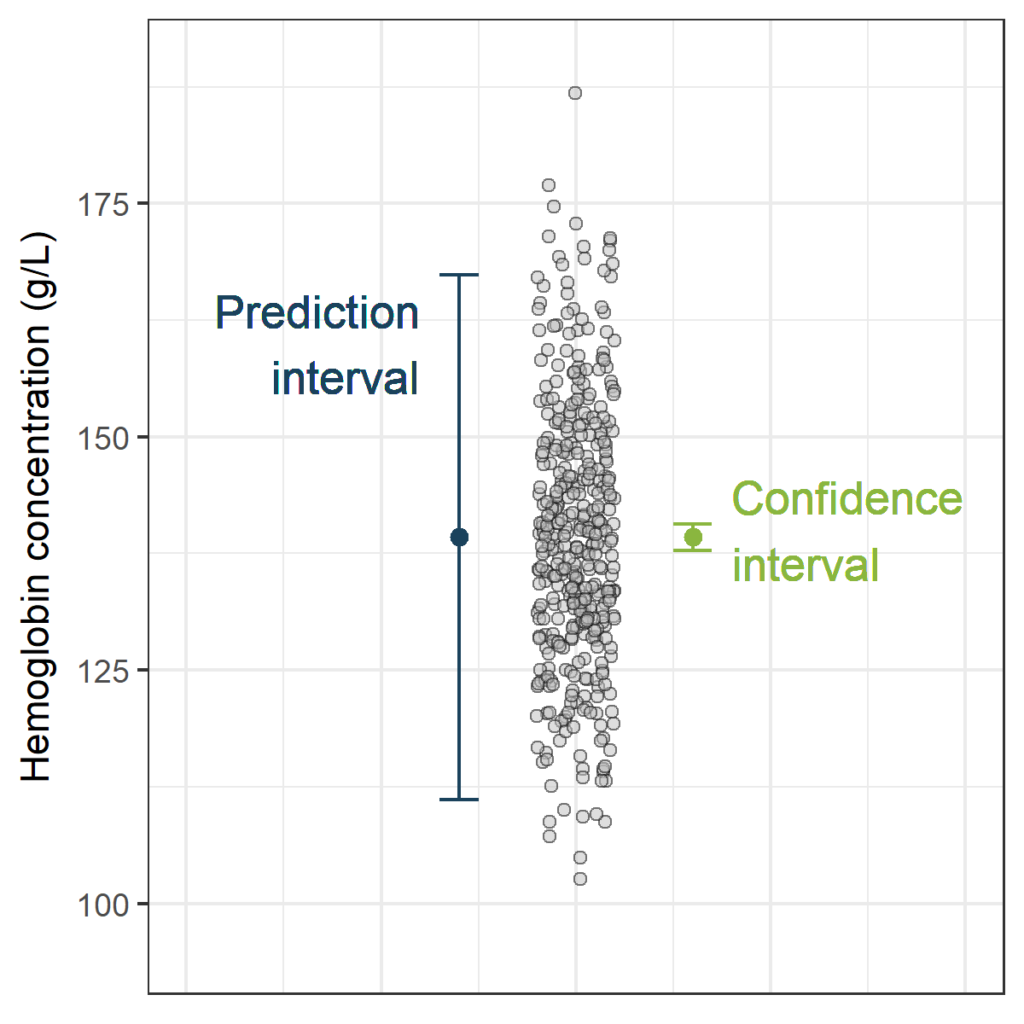
hjust="left", cex=2), col="#8AB63F")+

theme(legend.position="none",

axis.text.x = element\_blank(),

axis.ticks.x = element\_blank())

PI\_CI

Gives this plot:  
[](https://datascienceplus.com/wp-content/uploads/2018/06/PI_CI_n400.png)

The width of the confidence interval is very small, now that we have this large sample size (n=400). This is not surprising, as the estimated mean is the only source of uncertainty. In contrast, the width of the prediction interval is still substantial. The prediction interval has two sources of uncertainty: the estimated mean (just like the confidence interval) and the random variance of new observations.

**Example: comparing a new with a reference measurement method.**

A prediction interval can be useful in the case where a new method should replace a standard (or reference) method.  
If we can predict well enough what the measurement by the reference method would be, (given the new method) than the two methods give similar information and the new method can be used.

For example in (Tian, 2017) a new spectral method (Near-Infra-Red) to measure hemoglobin is compared with a Golden Standard. In contrast with the Golden Standard method, the new spectral method does not require reagents. Moreover, the new method is faster. We will investigate whether we can predict well enough, based on the measured concentration of the new method, what the measurement by the Golden Standard would be. (note: the measured concentrations presented below are fictive)

Hb<- read.table("http://rforbiostatistics.colmanstatistics.be/wp-content/uploads/2018/06/Hb.txt",

header = TRUE)

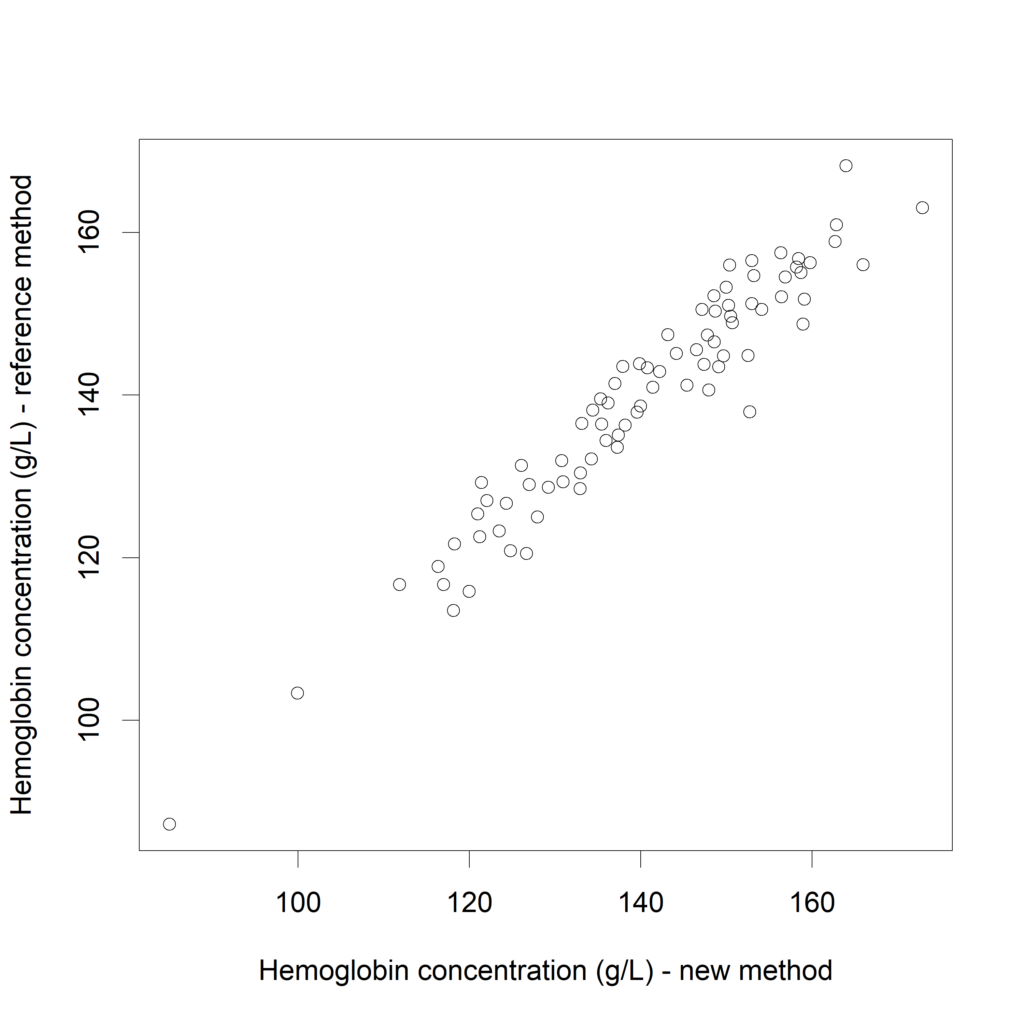
kable(head(Hb))

| **New** | **Reference** |
| --- | --- |
| 84.96576 | 87.24013 |
| 99.91483 | 103.38143 |
| 111.79984 | 116.71593 |
| 116.95961 | 116.72065 |
| 118.11140 | 113.51541 |
| 118.21411 | 121.70586 |

plot(Hb$New, Hb$Reference,

xlab="Hemoglobin concentration (g/L) - new method",

ylab="Hemoglobin concentration (g/L) - reference method")

Gives this plot:  
[](https://datascienceplus.com/wp-content/uploads/2018/06/plotHb.png)

**Prediction Interval based on linear regression**

Now, let's fit a linear regression model predicting the hemoglobin concentrations measured by the reference method, based on the concentrations measured with the new method.

fit.lm <- lm(Hb$Reference ~ Hb$New)

plot(Hb$New, Hb$Reference,

xlab="Hemoglobin concentration (g/L) - new method",

ylab="Hemoglobin concentration (g/L) - reference method")

cat ("Adding the regression line:")

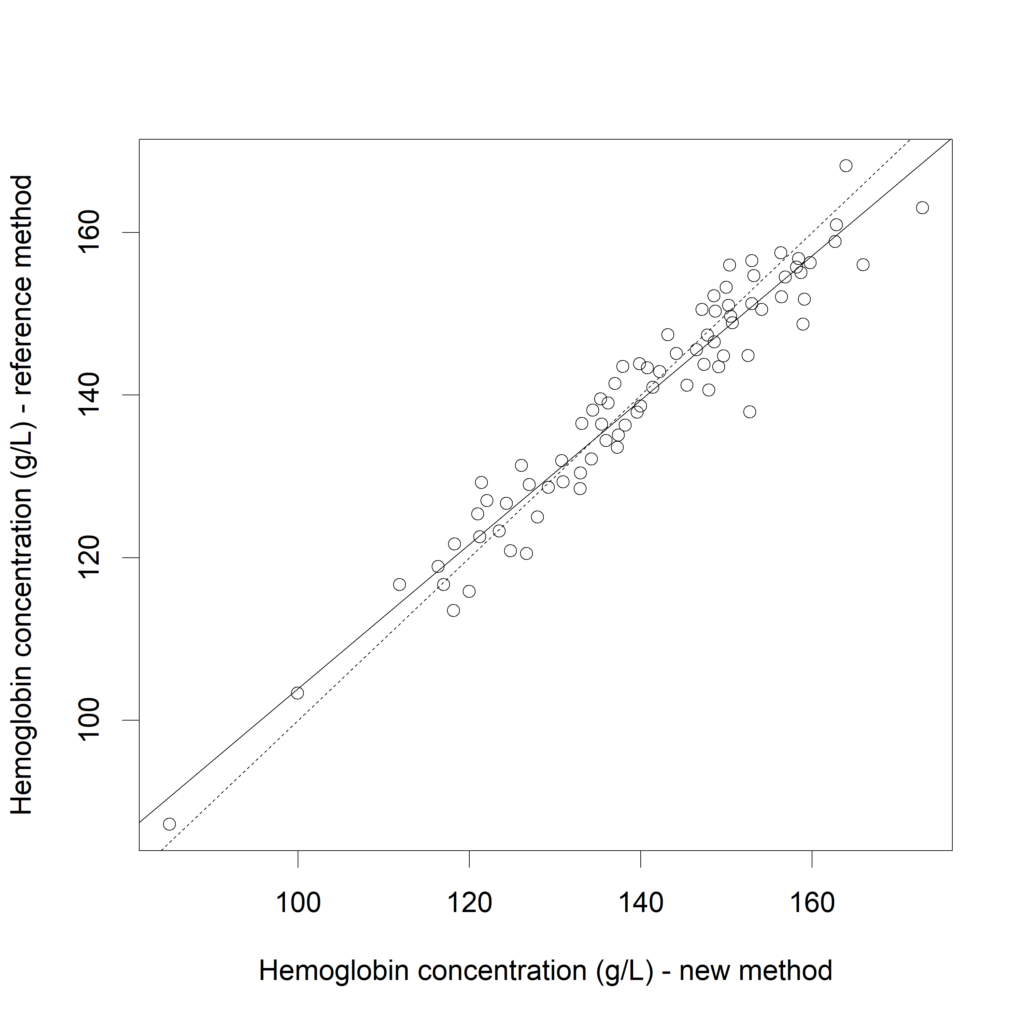
Adding the regression line:

abline (a=fit.lm$coefficients[1], b=fit.lm$coefficients[2] )

cat ("Adding the identity line:")

Adding the identity line:

abline (a=0, b=1, lty=2)

Th plot:  
[](https://datascienceplus.com/wp-content/uploads/2018/06/plot1.png)

If both measurement methods would exactly correspond, the intercept would be zero and the slope would be one (=“identity line”, dotted line). Now, let's calculated the confidence interval for this linear regression.

CI\_ex <- predict(fit.lm, interval="confidence")

colnames(CI\_ex)<- c("fit\_CI", "lwr\_CI", "upr\_CI")

And the prediction interval:

PI\_ex <- predict(fit.lm, interval="prediction")

*## Warning in predict.lm(fit.lm, interval = "prediction"): predictions on current data refer to \_future\_ responses*

colnames(PI\_ex)<- c("fit\_PI", "lwr\_PI", "upr\_PI")

We can combine those results in one data frame and plot both the confidence interval and the prediction interval.

Hb\_results<-cbind(Hb, CI\_ex, PI\_ex)

kable(head(round(Hb\_results),1))

| **New** | **Reference** | **fit\_CI** | **lwr\_CI** | **upr\_CI** | **fit\_PI** | **lwr\_PI** | **upr\_PI** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 85 | 87 | 91 | 87 | 94 | 91 | 82 | 99 |

Visualizing the CI (in green) and the PI (in blue):

plot(Hb$New, Hb$Reference,

xlab="Hemoglobin concentration (g/L) - new method",

ylab="Hemoglobin concentration (g/L) - reference method")

Hb\_results\_s <- Hb\_results[order(Hb\_results$New),]

lines (x=Hb\_results\_s$New, y=Hb\_results\_s$fit\_CI)

lines (x=Hb\_results\_s$New, y=Hb\_results\_s$lwr\_CI,

col="#8AB63F", lwd=1.2)

lines (x=Hb\_results\_s$New, y=Hb\_results\_s$upr\_CI,

col="#8AB63F", lwd=1.2)

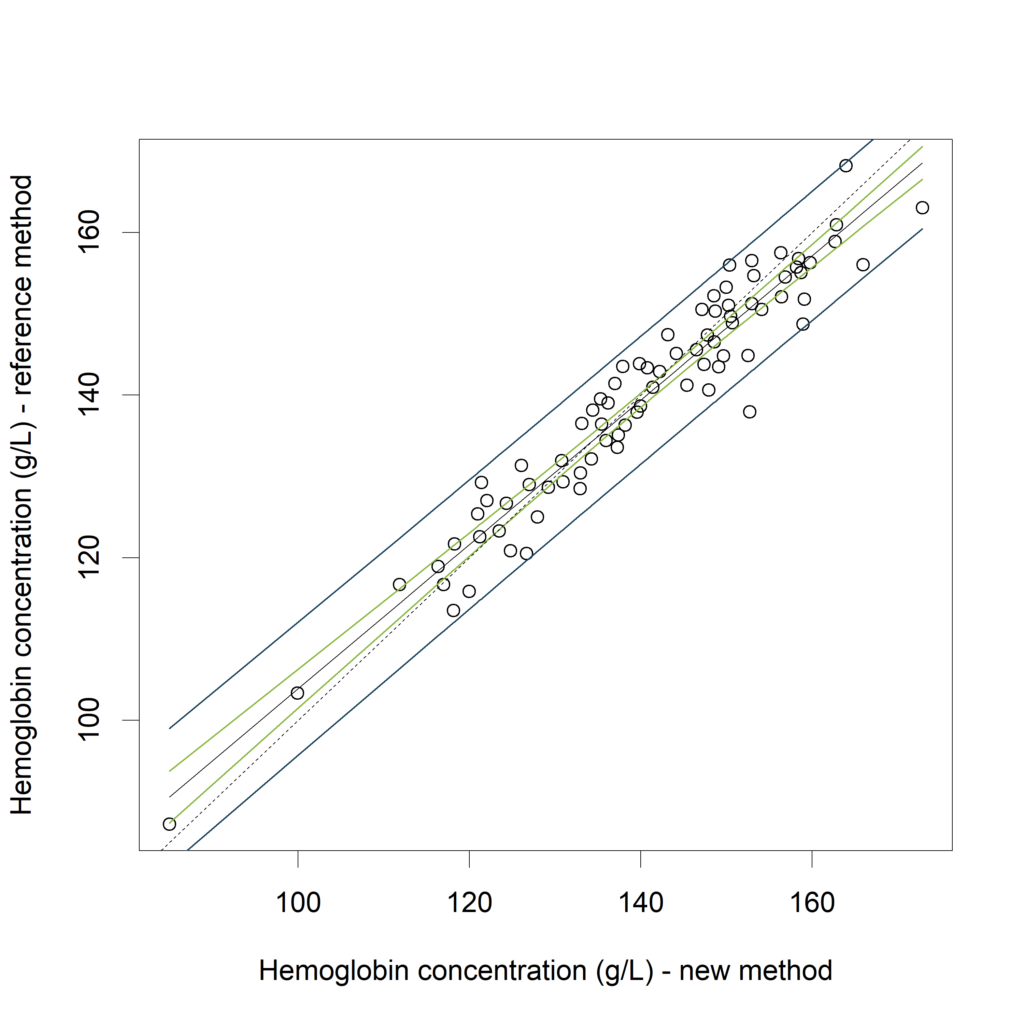
lines (x=Hb\_results\_s$New, y=Hb\_results\_s$lwr\_PI,

col="#1A425C", lwd=1.2)

lines (x=Hb\_results\_s$New, y=Hb\_results\_s$upr\_PI,

col="#1A425C", lwd=1.2)

abline (a=0, b=1, lty=2)

Gives this plot:  
[](https://datascienceplus.com/wp-content/uploads/2018/06/plots.png)

In (Bland, Altman 2003) it is proposed to calculate the average width of this prediction interval, and see whether this is acceptable. Another approach is to compare the calculated PI with an “acceptance interval”. If the PI is inside the acceptance interval for the measurement range of interest (see Francq, 2016).

In the above example, both methods do have the same measurement scale (g/L), but the linear regression with prediction interval is particularly useful when the two methods of measurement have different units.

However, the method has some disadvantages:

* Predictions intervals are very sensitive to deviations from the normal distribution.
* In “standard” linear regression (or Ordinary Least Squares (OLS) regression),the presence of measurement error is allowed for the Y-variable (here, the reference method) but not for the X-variable (the new method). The absence of errors on the x-axis is one of the assumptions. Since we can expect some measurement error for the new method, this assumption is violated here.

**Taking into account errors on both axes**

In contrast to Ordinary Least Square (OLS) regression, Bivariate Least Square (BLS) regression takes into account the measurement errors of both methods (the New method and the Reference method). Interestingly, prediction intervals calculated with BLS are not affected when the axes are switched (del Rio, 2001).

In 2017, a new R-package BivRegBLS was released. It offers several methods to assess the agreement in method comparison studies, including Bivariate Least Square (BLS) regression.

If the data are unreplicated but the variance of the measurement error of the methods is known, The BLS() and XY.plot() functions can be used to fit a bivariate Least Square regression line and corresponding confidence and prediction intervals.

library (BivRegBLS)

Hb.BLS = BLS (data = Hb, xcol = c("New"),

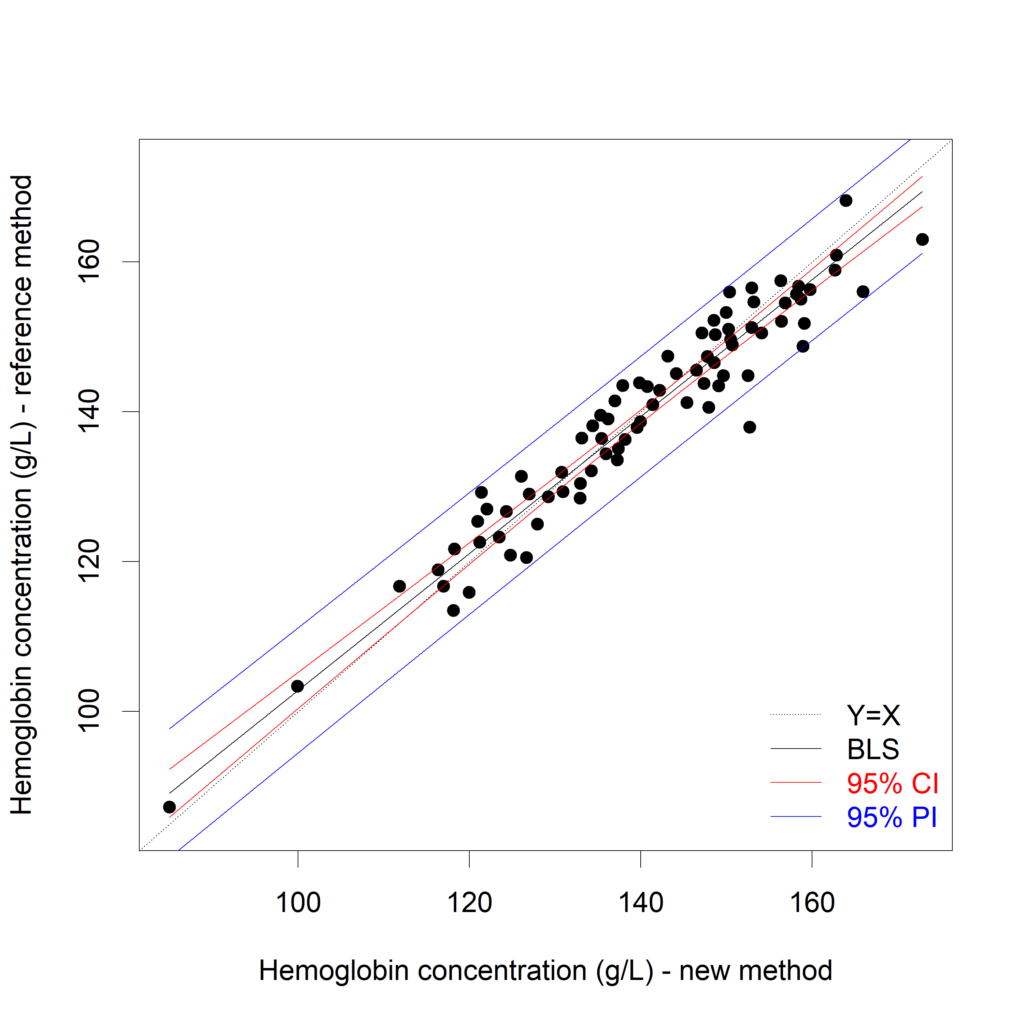
ycol = c("Reference"), var.y=10, var.x=8, conf.level=0.95)

XY.plot (Hb.BLS,

yname = "Hemoglobin concentration (g/L) - reference method",

xname = "Hemoglobin concentration (g/L) - new method",

graph.int = c("CI","PI"))

Gives this plot:  
[](https://datascienceplus.com/wp-content/uploads/2018/06/plot2.png)

Now we would like to decide whether the new method can replace the reference method. We allow the methods to differ up to a given threshold, which is not clinically relevant. Based on this threshold an “acceptance interval” is created. Suppose that differences up to 10 g/L (=threshold) are not clinically relevant, then the acceptance interval can be defined as Y=X±??, with ?? equal to 10. If the PI is inside the acceptance interval for the measurement range of interest then the two measurement methods can be considered to be interchangeable (see Francq, 2016).

The accept.int argument of the XY.plot() function allows for a visualization of this acceptance interval

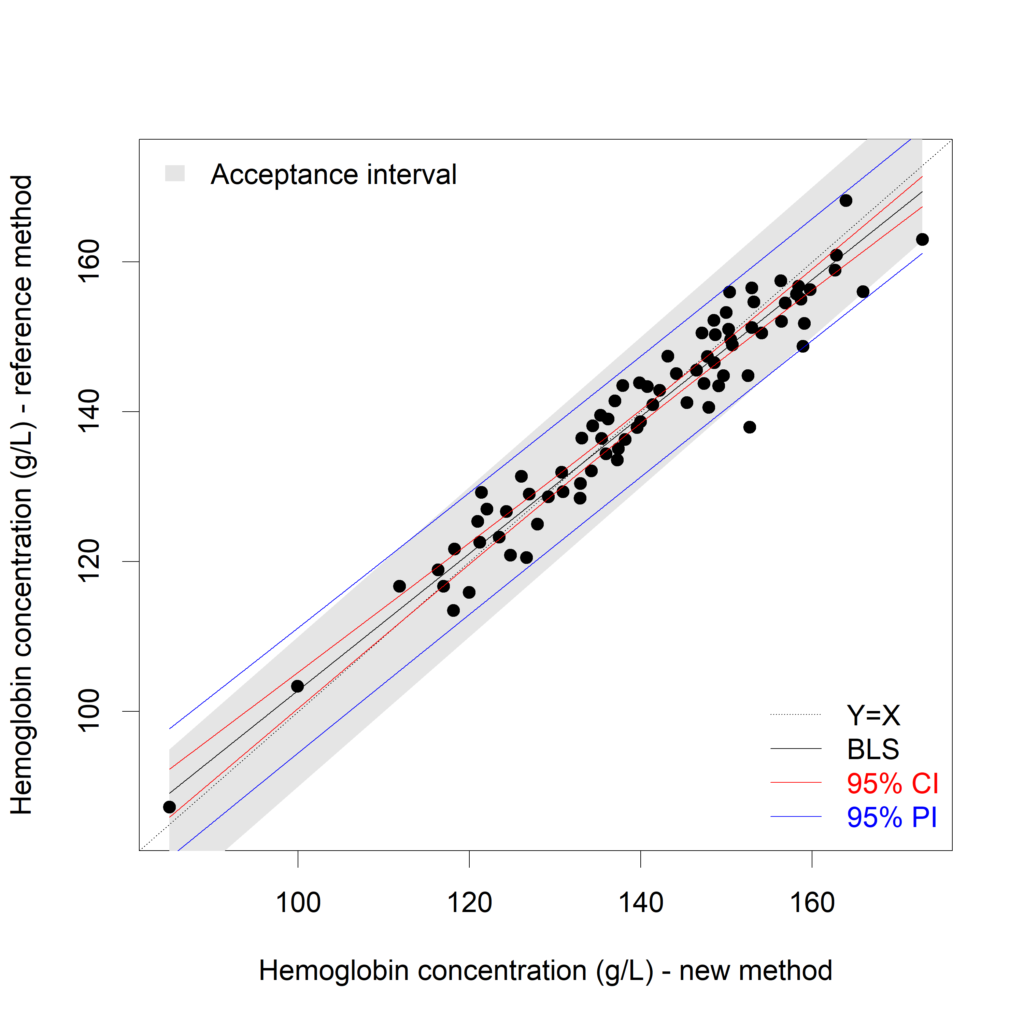
XY.plot (Hb.BLS,

yname = "Hemoglobin concentration (g/L) - reference method",

xname = "Hemoglobin concentration (g/L) - new method",

graph.int = c("CI","PI"),

accept.int=10)

Givbes this plot:  
[](https://datascienceplus.com/wp-content/uploads/2018/06/plot2b.png)

For the measurement region 120g/L to 150 g/L, we can conclude that the difference between both methods is acceptable. If the measurement regions below 120g/l and above 150g/L are important, the new method cannot replace the reference method.

**Regression on replicated data**

In method comparison studies, it is advised to create replicates (2 or more measurements of the same sample with the same method). An example of such a dataset:

Hb\_rep <- read.table("http://rforbiostatistics.colmanstatistics.be/wp-content/uploads/2018/06/Hb\_rep.txt",

header = TRUE)

kable(head(round(Hb\_rep),1))

| **New\_rep1** | **New\_rep2** | **Ref\_rep1** | **Ref\_rep2** |
| --- | --- | --- | --- |
| 88 | 95 | 90 | 84 |

When replicates are available, the variance of the measurement errors are calculated for both the new and the reference method and used to estimate the prediction interval. Again, the BLS() function and the XY.plot() function are used to estimate and plot the BLS regression line, the corresponding CI and PI.

Hb\_rep.BLS = BLS (data = Hb\_rep,

xcol = c("New\_rep1", "New\_rep2"),

ycol = c("Ref\_rep1", "Ref\_rep2"),

qx = 1, qy = 1,

conf.level=0.95, pred.level=0.95)

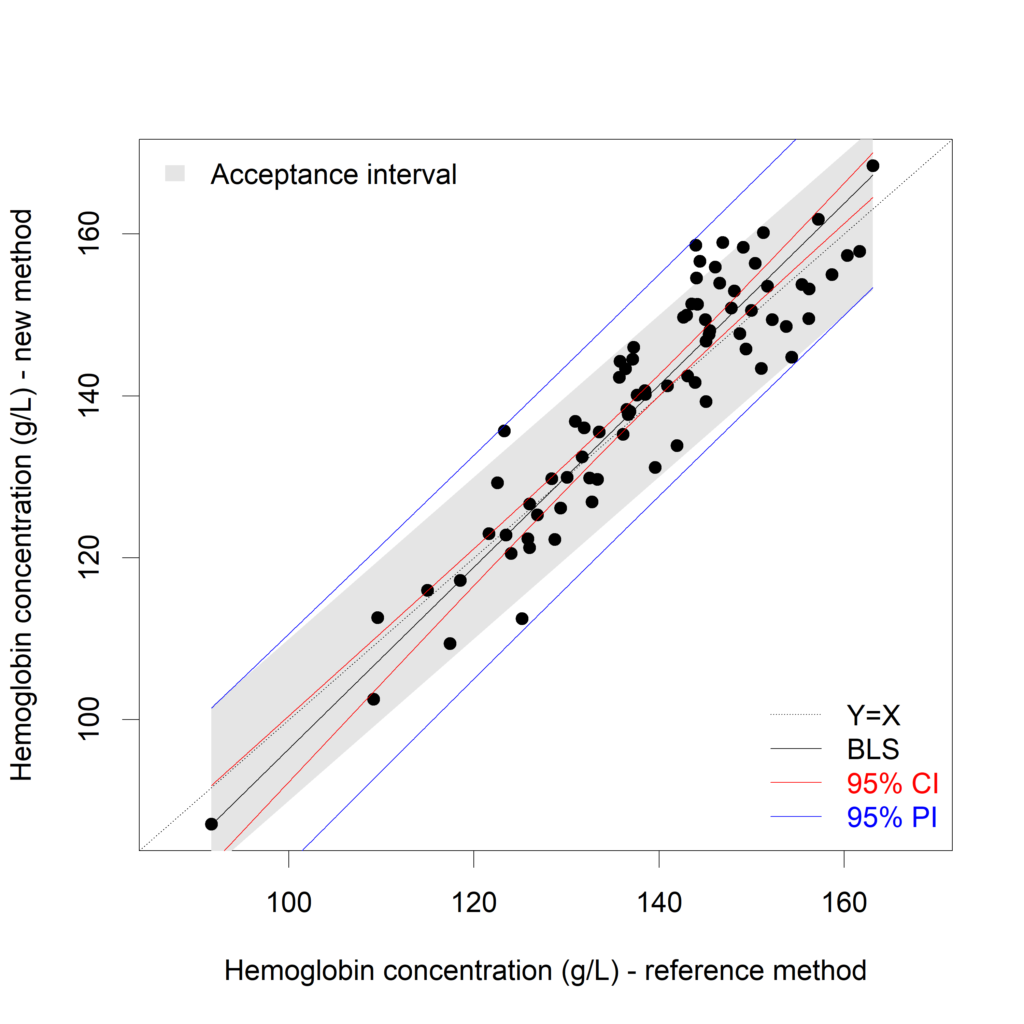
XY.plot (Hb\_rep.BLS,

yname = "Hemoglobin concentration (g/L) - reference method",

xname = "Hemoglobin concentration (g/L) - new method",

graph.int = c("CI","PI"),

accept.int=10)

Giuves this plot:  
[](https://datascienceplus.com/wp-content/uploads/2018/06/plot3.png)

It is clear that the prediction interval is not inside the acceptance interval here. The new method cannot replace the reference method. A possible solution is to average the repeats. The BivRegBLS package can create prediction intervals for the mean of (2 or more) future values, too!

In the plot above, averages of the two replicates are calculated and plotted. I'd like to see the individual measurements:

plot(x=c(Hb\_rep$New\_rep1, Hb\_rep$New\_rep2),

y=c(Hb\_rep$Ref\_rep1, Hb\_rep$Ref\_rep2),

xlab="Hemoglobin concentration (g/L) - new method",

ylab="Hemoglobin concentration (g/L) - reference method")

lines (x=as.numeric(Hb\_rep.BLS$Pred.BLS[,1]),

y=as.numeric(Hb\_rep.BLS$Pred.BLS[,2]),

lwd=2)

lines (x=as.numeric(Hb\_rep.BLS$Pred.BLS[,1]),

y=as.numeric(Hb\_rep.BLS$Pred.BLS[,3]),

col="#8AB63F", lwd=2)

lines (x=as.numeric(Hb\_rep.BLS$Pred.BLS[,1]),

y=as.numeric(Hb\_rep.BLS$Pred.BLS[,4]),

col="#8AB63F", lwd=2)

lines (x=as.numeric(Hb\_rep.BLS$Pred.BLS[,1]),

y=as.numeric(Hb\_rep.BLS$Pred.BLS[,5]),

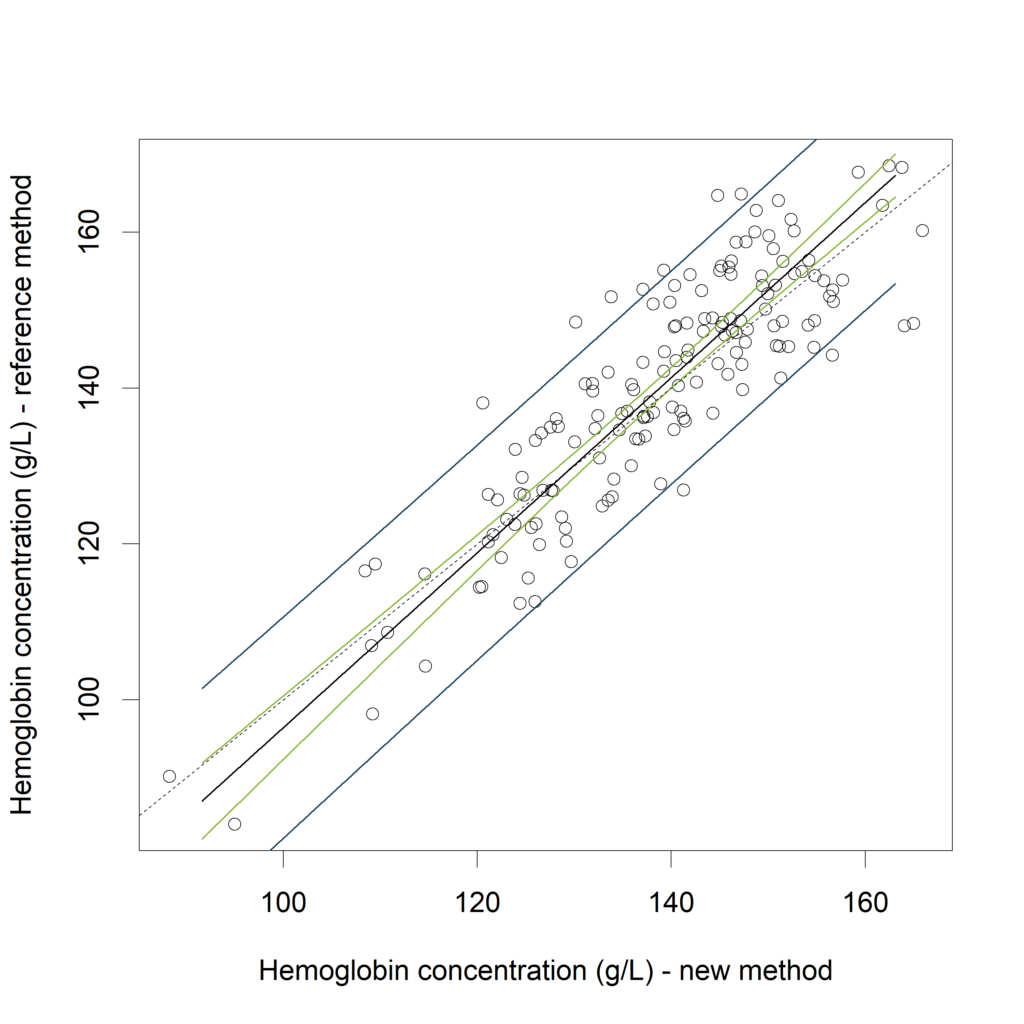
col="#1A425C", lwd=2)

lines (x=as.numeric(Hb\_rep.BLS$Pred.BLS[,1]),

y=as.numeric(Hb\_rep.BLS$Pred.BLS[,6]),

col="#1A425C", lwd=2)

abline (a=0, b=1, lty=2)

Gives this plot:  
[](https://datascienceplus.com/wp-content/uploads/2018/06/plot4.png)

**Remarks**

* Although not appropriate in the context of method comparison studies, Pearson correlation is still frequently used. See Bland & Altman (2003) for an explanation on why correlations are not adviced.
* Methods presented in this blogpost are not applicable to time-series