ACB spreadsheet verification: method/patient comparison

Ed Wilkes

This document describes the verification of the method/patient comparison spreadsheet, written by Prof Anders Kallner, that performs calculations for the assessment of differences between two sample sets (July 2018 version). Calculations performed by these spreadsheets were verified in an independent statistical software (the R statistical computing environment v3.4.1) by the author of this document. The R packages required to run this code are shown below. This code can be copied and pasted into an instance of R and, given the test data as input, reproduce the analysis in this document.

Required packages:

```
require(dplyr)
require(ggplot2)
require(knitr)
require(mcr)
require(moments)
require(outliers)
require(reshape2)
require(tidyr)
```

Reading data into R:

```
# Read in csv file: "2018-07 ACB Method comparison, patient samples - test
data.csv"
df <- read.csv(file.choose(), header = TRUE)
```

Check calculation of means, relative and absolute differences:

These are the data presented in columns G:J from row 13 onwards. Absolute bias is calculated as follows, where $\bar{x}$ represents the mean of the relevant measurements.

$$\text{Absolute bias} = \bar{x}_y - \bar{x}_x$$

Relative bias is calculated as follows:

$$\text{Relative bias} = \frac{\bar{x}_y - \bar{x}_x}{\bar{x}_x} \cdot 100$$

```
df <- df %>%
  group_by(sample) %>%
  mutate(mean_x = round(mean(c(method_1_rep_1, method_1_rep_2)), 2)
         ,mean_y = round(mean(c(method_2_rep_1, method_2_rep_2)), 2)
         ,relative_diff = round((mean_y - mean_x) / mean_x * 100, 1)
         ,absolute_diff = round(mean_y - mean_x, 2)
  )
```

```
kable(df)
```

method_1_rep_1	method_1_rep_2	method_2_rep_1	method_2_rep_2	sample	mean_x	mean_y	relative_diff	absolute_diff
18.60116	18.53197	20.15498	20.27766	1	18.57	20.22	8.9	1.65
19.38586	19.51303	21.26132	21.26753	2	19.45	21.26	9.3	1.81
21.48562	21.43538	23.18010	23.13701	3	21.46	23.16	7.9	1.70
19.68039	19.57123	21.09728	21.07627	4	19.63	21.09	7.4	1.46
19.94862	19.92508	22.09524	22.11352	5	19.94	22.10	10.8	2.16
20.36261	20.16845	22.41640	22.51573	6	20.27	22.47	10.9	2.20
19.88527	19.84320	21.89768	22.00686	7	19.86	21.95	10.5	2.09
21.28557	21.50132	23.66281	23.59001	8	21.39	23.63	10.5	2.24
18.86726	18.89286	20.93664	20.88221	9	18.88	20.91	10.8	2.03
21.47574	21.26134	23.66020	23.61715	10	21.37	23.64	10.6	2.27
21.22596	21.32730	23.28220	23.22750	11	21.28	23.25	9.3	1.97
18.87582	19.20725	20.73483	20.88164	12	19.04	20.81	9.3	1.77
19.57281	19.60013	21.59360	21.70506	13	19.59	21.65	10.5	2.06
19.30487	19.46057	21.85298	21.72839	14	19.38	21.79	12.4	2.41
20.85705	21.16725	23.10865	23.16631	15	21.01	23.14	10.1	2.13
20.10052	20.24209	21.89479	21.86125	16	20.17	21.88	8.5	1.71
19.30564	19.20717	21.31628	21.32139	17	19.26	21.32	10.7	2.06
19.91938	19.92743	22.06979	22.08042	18	19.92	22.08	10.8	2.16
20.76180	20.57981	22.57988	22.87126	19	20.67	22.73	10.0	2.06
17.97752	17.87576	20.12523	20.02294	20	17.93	20.07	11.9	2.14
17.90993	17.81860	19.90440	19.86442	21	17.86	19.88	11.3	2.02
19.77760	19.61176	21.68401	21.67320	22	19.69	21.68	10.1	1.99
18.89125	18.66982	20.20817	20.66214	23	18.78	20.44	8.8	1.66
17.17749	17.11119	19.14998	19.05155	24	17.14	19.10	11.4	1.96
21.63746	21.59639	23.63230	23.52005	25	21.62	23.58	9.1	1.96
19.84472	19.94806	21.78475	21.87926	26	19.90	21.83	9.7	1.93
21.49990	21.44791	23.47181	23.47261	27	21.47	23.47	9.3	2.00
20.65765	20.89596	22.71556	22.73426	28	20.78	22.72	9.3	1.94
19.51010	19.56230	21.65891	21.46737	29	19.54	21.56	10.3	2.02
20.74755	20.85127	22.48977	22.51080	30	20.80	22.50	8.2	1.70
21.35716	21.25436	23.91606	24.00512	31	21.31	23.96	12.4	2.65
19.32776	19.46531	21.60421	21.36150	32	19.40	21.48	10.7	2.08
19.44721	19.58336	21.68235	21.62503	33	19.52	21.65	10.9	2.13
20.83126	20.73788	22.84730	22.70082	34	20.78	22.77	9.6	1.99
19.46359	19.19397	21.33642	21.41477	35	19.33	21.38	10.6	2.05
20.33817	20.36140	22.34849	22.17548	36	20.35	22.26	9.4	1.91
20.68189	20.87643	22.80514	22.72556	37	20.78	22.77	9.6	1.99
21.07800	21.13281	22.86273	22.88037	38	21.11	22.87	8.3	1.76
20.44122	20.31547	22.67285	22.52374	39	20.38	22.60	10.9	2.22

method_1_rep_1	method_1_rep_2	method_2_rep_1	method_2_rep_2	sample	mean_x	mean_y	relative_diff	absolute_diff
19.50336	19.45142	21.43274	21.48335	40	19.48	21.46	10.2	1.98

```
df %>%
  ungroup() %>%
  summarise(max_abs_diff = max(absolute_diff)
            ,min_abs_diff = min(absolute_diff)
            ,max_rel_diff = max(relative_diff)
            ,min_rel_diff = min(relative_diff)) %>%
  kable
```

max_abs_diff	min_abs_diff	max_rel_diff	min_rel_diff
2.65	1.46	12.4	7.4

The mean of both X and Y, and the absolute differences between each value, match those calculated in the spreadsheet.

Perform *t*- and Wilcoxon rank-sum tests:

These are the data presented in cells K4:L11.

```
t.test(x = df$mean_x
       ,y = df$mean_y
       ,alternative = "two.sided"
       ,paired = TRUE
       ,conf.level = 0.95)
##
##  Paired t-test
##
## data:  df$mean_x and df$mean_y
## t = -57.241, df = 39, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.07119 -1.92981
## sample estimates:
## mean of the differences
##                -2.0005
wilcox.test(x = df$mean_x
            ,y = df$mean_y
            ,alternative = "two.sided"
            ,paired = TRUE
            ,conf.level = 0.95)
## Warning in wilcox.test.default(x = df$mean_x, y = df$mean_y, alternative
=
## "two.sided", : cannot compute exact p-value with ties
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  df$mean_x and df$mean_y
## V = 0, p-value = 3.694e-08
## alternative hypothesis: true location shift is not equal to 0
```

The values obtained from these tests both reject the null hypothesis (H_0, i.e., that $\mu_1=\mu_2$) and accept the alternative hypothesis (H_1, i.e., that $\mu_1\neq\mu_2$) and match those calculated in the spreadsheet.

Check calculations of variable mean, SD, maximum, and minimum:

These are the data presented in cells K12:P13 and M16:P17.

```
df %>%
  ungroup() %>%
  summarise(total_mean_x = mean(mean_x)
            ,total_s_x = sd(mean_x)
            ,ci_lower_x = total_mean_x - (1.96 * total_s_x)
            ,ci_upper_x = total_mean_x + (1.96 * total_s_x)
            ,min_x = min(mean_x)
            ,max_x = max(mean_x)
            ,total_mean_y = mean(mean_y)
            ,total_s_y = sd(mean_y)
            ,ci_lower_y = total_mean_y - (1.96 * total_s_y)
            ,ci_upper_y = total_mean_y + (1.96 * total_s_y)
            ,min_y = min(mean_y)
            ,max_y = max(mean_y)
  ) %>%
  kable
```

total_mean_x	total_s_x	ci_lower_x	ci_upper_x	min_x	max_x	total_mean_y	total_s_y	ci_lower_y	ci_upper_y	min_y	max_y
19.97725	1.080653	17.85917	22.09533	17.14	21.62	21.97775	1.134461	19.75421	24.20129	19.1	23.96

The calculated values match those provided by the spreadsheet.

Check calculations of mean of differences and their SDs and SEMs:

These are the data presented in cells K14:P15.

```
df %>%
  ungroup() %>%
  summarise(mean_abs_diff = mean(absolute_diff)
            ,sd_abs_diff = sd(absolute_diff)
            ,sem_abs_diff = sd(absolute_diff) / sqrt(40)
            ,mean_rel_diff = mean(relative_diff)
            ,sd_rel_diff = sd(relative_diff)
            ,sem_rel_diff = sd(relative_diff) / sqrt(40)
  ) %>%
  kable
```

mean_abs_diff	sd_abs_diff	sem_abs_diff	mean_rel_diff	sd_rel_diff	sem_rel_diff
2.0005	0.2210343	0.0349486	10.03	1.153856	0.1824407

The calculated values match those provided by the spreadsheet.

Check calculations of partitions:

These are the data presented in cells P21:Y24.

```
# Partition data and calculate statistics
df %>%
  mutate(partition = ifelse(mean_x > 17.0 && mean_x < 19.0
                            ,yes = 1
                            ,no = ifelse(mean_x >= 19.0 && mean_x < 21.0
                                         ,yes = 2
                                         ,no = 3)
                     )
  ) %>%
  group_by(partition) %>%
  summarise(n = n()
            ,mean_abs_bias = mean(absolute_diff)
            ,mean_rel_bias = mean(relative_diff)
            ,sd_x = sd(mean_x)
            ,sd_y = sd(mean_y)
            ,t_stat = t.test(x = mean_x, y = mean_y, paired = TRUE,
conf.level = 0.95)$statistic
            ,p_value = t.test(x = mean_x, y = mean_y, paired = TRUE,
conf.level = 0.95)$p.value
            ,pearson_r = cor(x = mean_x, y = mean_y, method = "pearson") ^
2
            ,slope = lm(mean_y ~ mean_x)$coefficients[2]
            ,intercept = lm(mean_y ~ mean_x)$coefficients[1]
  ) %>%
  kable
```

partition	n	mean_abs_bias	mean_rel_bias	sd_x	sd_y	t_stat	p_value	pearson_r	slope	intercept
1	6	1.910000	10.516667	0.6704526	0.6058603	-22.72093	3.1e-06	0.9081035	0.8611367	4.436387
2	25	1.995200	10.024000	0.5487173	0.5696540	-50.83624	0.0e+00	0.8819122	0.9749339	2.494426
3	9	2.075556	9.722222	0.1868229	0.3333333	-21.53824	0.0e+00	0.2513010	0.8944290	4.327972

Calculated values generally match those in the spreadsheet. There are some small differences in the regression coefficients and r^2r2 values; however, this is most likely to be as a result of rounding errors.

Check calculations of skewness:

These are the data presented in cells O27:P28.

```
# Skewness for absolute differences
skewness(df$absolute_diff)
## [1] 0.1392153
# Skewness for relative differences
skewness(df$relative_diff)
## [1] -0.1150158
```

Calculated values for the skewness of the relative and absolute difference distributions differ slightly to those presented in the spreadsheet.

Perform regression analyses (Deming and OLR):

These are the data presented in cells S30:W38.

Deming:

```
# Fit Deming regression and get parameters
mcreg(x = df$mean_x
      ,y = df$mean_y
      ,error.ratio = 1
      ,alpha = 0.05
      ,method.reg = "Deming"
      ,method.ci = "analytical")@para
##                 EST         SE        LCI      UCI
## Intercept 0.9863172 0.66972726 -0.3694747 2.342109
## Slope     1.0507669 0.03347678  0.9829967 1.118537
# EST = estimate of parameter
# SE = standard error of parameter
# LCI = lower confidence interval
# UCI = upper confidence interval
```

OLR:

```
# Fit OLR model
lm(df$mean_y ~ df$mean_x)
##
## Call:
## lm(formula = df$mean_y ~ df$mean_x)
##
## Coefficients:
## (Intercept)    df$mean_x
##       1.399        1.030
# Get confidence intervals for above model
confint(lm(df$mean_y ~ df$mean_x))
##                   2.5 %   97.5 %
## (Intercept) 0.06974965 2.728039
## df$mean_x   0.96367635 1.096553
```

Correlation coefficients:

```
# Calculate Pearson's r and coefficient of determination (r^2)
cor.test(x = df$mean_x
         ,y = df$mean_y
         ,method = "pearson"
         ,conf.level = 0.95)
```

```
##
##  Pearson's product-moment correlation
##
## data:  df$mean_x and df$mean_y
## t = 31.388, df = 38, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9645932 0.9901157
## sample estimates:
##       cor
## 0.9812552
```

The calculated regression coefficients and their confidence intervals matched those in the spreadsheet.

Check calculations in "Histogram" sheet

```
df %>%
  select(sample, mean_x, mean_y) %>%
  melt(id.vars = "sample") %>%
  group_by(variable) %>%
  summarise(max = max(value)
            ,min = min(value)
            ,mean = mean(value)
            ,sd = sd(value)
            ,skew = skewness(value)
            ,median = median(value)
            ,percent_2.5 = quantile(value, 0.025)
            ,percent_25 = quantile(value, 0.25)
            ,percent_75 = quantile(value, 0.75)
            ,percent_97.5 = quantile(value, 0.975)
  ) %>%
  kable
```

variable	max	min	mean	sd	skew	median	percent_2.5	percent_25	percent_75	percent_97.5
mean_x	21.62	17.14	19.97725	1.080653	-0.4715045	19.910	17.8420	19.395	20.785	21.47375
mean_y	23.96	19.10	21.97775	1.134461	-0.3773299	21.915	19.8605	21.365	22.770	23.64800

Conclusions:

1. Calculations of means, absolute and relative biases match those calculated in the spreadsheet
2. Calculations of the *t*-tests and Wilcoxon rank sum tests matched those in the spreadsheet
3. Calculations of the overall mean of X and Y variables and measures of their variation matched those provided in the spreadsheet

4. Calculations of the mean differences and measures of their variation matched those in the spreadsheet
5. Statistics calculated for the partitions matched those in the spreadsheet
6. Skewness for the relative and absolute differences distributions provided similar, but different values to those presented in the spreadsheet
7. Calculated Deming and OLR regression coefficients and their confidence intervals were identical to those shown in the spreadsheet
8. Calculation of means, medians, percentiles, etc. for the X and Y variables matched those shown in the “Histogram” spreadsheet. Some differences in the skewness values were observed; however, this may be due to rounding errors