---

title: "Classical Single Case Analysis for AB Designs"

author: "Compiled by Brandon K. Schultz, Ed.D., NCSP"

date: "January 15, 2016"

output: word\_document

---

```{r data\_entry, echo=FALSE}

# Enter your data in the "scdata" line below, between the parentheses, placing a comma between each measurement occasion. The same metric (e.g., 0-100 scale) must be used at each occasion.

scdata <- c(87.5, 82.5, 53.4, 72.3, 94.2, 96.6, 57.4, 78.1, 47.2, 80.7, 82.1, 73.7, 49.3, 79.3, 73.3, 57.3, 31.7, 50.4, 77.8, 67, 40.5, 1.6, 38.6, 3.2, 24.1)

n1 = 10 # This is the number of data points that were collected at baseline (A phase)

n2 = 15 # This is the number of data points collected during/after intervention (B phase)

lonum = min(scdata)

hinum = max(scdata)

ypre = scdata[1:n1]

ypost = scdata[n1 + 1:n2]

ssd1 <- data.frame(myy = ypre, myphase = "A")

ssd2 <- data.frame(myy = ypost, myphase = "B")

ssdall = rbind(ssd1, ssd2)

ssdall["xrownum"] <- c(1:nrow(ssdall))

```

Below is an analysis of AB (two-phase) single-case data using some of the more common methods. The analyses are based on software produced by Auerbach and Zeitlen (2013). Users should select the best test depending on their sample and whether their data are consistent with the assumptions of the test. Some tests (e.g., outcomes relative to a goal line) cannot be produced here, but users can still use this code to explore their data prior to applying those tests.

We begin with a simple line graph of the data, with a vertical line separating the baseline and treatment data. Make sure that the data in the graph are accurate before proceding.

#####Figure 1: Overview of Single-Case Data

```{r plot\_data, dpi=300, echo=FALSE}

suppressPackageStartupMessages(library(SSDforR))

suppressPackageStartupMessages(require(SSDforR))

library(SSDforR)

require(SSDforR)

suppressPackageStartupMessages(library(ggplot2))

library(ggplot2)

suppressPackageStartupMessages(library(rmarkdown))

library(rmarkdown)

Trendplot <- ggplot(ssdall, aes(x=xrownum, y=myy, colour=myphase)) +

geom\_vline(linetype=2, xintercept=n1+.5)

Trendplot + geom\_line(size=1) + geom\_point() +

theme(legend.title=element\_blank()) +

labs(x = "Time", y = "Data")

```

## I. Checking Assumptions

### A. Is the Baseline Stable?

Assuming the data are entered correctly, we begin by checking the stability of the baseline. Using Figure 2, examine whether baseline data stay within a range we would expect to occur by chance. In the figure, purple lines show the range of expected values around the mean (black) of the baseline. If more than 5% if the baseline data are outside of this range -- referred to as "outliers" -- it suggests that the baseline was unstable. Make a note of that before continuing.

##### Figure 2: A Closer Look at the Baseline...

```{r graph\_baseline, dpi=300, echo=FALSE}

Trendplot <- ggplot(ssdall, aes(x=xrownum, y=myy, colour=myphase)) +

geom\_hline(yintercept=mean(ypre), color="black", size=0.5) +

geom\_hline(yintercept=mean(ypre)+2\*sd(ypre), color="purple", size=0.5, linetype="longdash") +

geom\_hline(yintercept=mean(ypre)-2\*sd(ypre), color="purple", size=0.5, linetype="longdash") +

geom\_vline(linetype=2, xintercept=n1+.5)

Trendplot + geom\_line(data=subset(ssdall, myphase=="A"), size=1) +

geom\_point(data=subset(ssdall, myphase=="A")) +

theme(legend.title=element\_blank()) +

labs(x = "Time", y = "Data")

```

It is also vital to check for a trends that would suggest increasing or decreasing values in either phase. If there is a trend, outcomes can be difficult to interpret because the subject's behavior was changing before the intervention began, or varied during the intervention (rather than responding consistently).

A nonsignificant trend test suggests that the baseline and intervention response was stable, so p-values ABOVE .10 in the lines labelled "F-statistic" in Box 1 are desirable. Note that the baseline is tested first, followed by the intervention phase. Also note that highly inconsistent data collection can impact these estimates (read the section on the G-index below for more information).

##### Box 1. Trend Analysis

```{r, echo=FALSE, message=FALSE}

attach(ssdall)

ABregres(myy, myphase, "A", "B")

```

If the baseline was brief (<8 data points), we also want to make sure that the estimates of "x1" and "x2" in Box 1 are not too large. These values are the "slopes" in each phase, which tell us how much the behavior changed on average from time to time. Ideally, x1 and x2 should be a small fraction of the total range of data observed, particularly in the baseline (x1). If the baseline slope leads to unrealistic predictions in Figure 5, for example, then there is a meaningful trend, even if the test in Box 1 is nonsignificant. Read the section on the G-index test below for more information. Make a note of this outcome before continuing.

#### A.1. What if there are too many outliers?

If the baseline has a high proportion of outliers (> 5%), according to Figure 2, the baseline is unstable and practitioners should continue to collect data prior to starting an intervention, if possible. If the intervention has already started, it is vital to use estimates of treatment outcome that do not assume a normal distribution in the data, In other words, use "nonparametric" tests.

#### A.2. What if there appears to be a trend in the baseline or intervention data?

If a trend test is significant (p < .10) for baseline or intervention, then it is advisable to use nonparametric tests of the effect or the Bayesian estimates below. If there is some indication of a trend in the baseline that is nonsignificant, use conservative estimates of effect (Hedge's g), nonparametric tests, or possibly the Bayesian analysis.

### B. Are the Data Autocorrelated?

Next, it is important to test the data for "autocorrelation," which can impact any test that assumes normal distributions (i.e., parametric tests). Although autocorrelation -- also called serial dependency -- is a complicated topic, the test is not. The test for "lag-1" autocorrelation for the baseline data is reported in Box 2. If the result of the test is significant (p < .05), then autocorrelation is a concern. The p-value can be found on the line labelled "sig of rf2."

##### Box 2: Test of Autocorrelation in the Baseline Data

```{r, echo=FALSE}

detach(ssdall)

attach(ssd1)

ABrf2(ypre, myphase, "A")

```

Similarly, it is important to test autocorrelation in the data collected in the intervention phase. This test is in Box 3 below (interpreted the same as above):

##### Box 3: Test of Autocorrelation in the Intervention Data

```{r, echo=FALSE}

detach(ssd1)

attach(ssd2)

ABrf2(ypost, myphase, "B")

```

#### B.1. What if there is significant autocorrelation?

Autocorrelation can bias single case analyses when using parametric tests, regardless of whether it occurs in the baseline or intervention phases. Most researchers reccommend transforming the data when there is autocorrelation, using either "moving average"" or "differencing"" strategies. Unfortunately, these transformations generally require 35 or more time points for accuracy and are difficult to interpret. The alternative is to use non-parametric tests (described below) or Bayesian analysis (provided in a separate Markdown file).

## II. Descriptive Statistics

A boxplot is a good way to visually summarize the disributions in the two phases, and it provides another chance to scan for potential outliers.

##### Figure 3: Boxplot of AB Phases

```{r, dpi=300, echo=FALSE}

detach(ssd2)

attach(ssdall)

descBoxplot <- ggplot(ssdall, aes(myphase, myy))

descBoxplot + geom\_boxplot() +

theme(legend.title=element\_blank()) +

labs(x = "Phases", y = "Data")

```

The descriptive statistics, including the mean, median, and SD are provided in Box 4 below. The "10% Trim Mean" removes the influence of extreme scores and, as a result, provides the best summary of the two phases if there are concerns with outliers, trends, or autocorrelation. The interquartile range, or "IQR," provides the range from the 25%ile to the 75%ile in each distribution, as displayed in Figure 3 above, but this can be influenced by unstable, trending data.

##### Box 4: Descriptive Statistics

```{r, echo=FALSE}

ABdescrip(myy, myphase)

```

## III. Effect Size Calculations

### A. Parametric Tests: Is there a Change in Levels from Phases A to B?

If there are no significant trends or outliers in the baseline and autocorrelation is not a concern in either phase, traditional effect size estimates can be used to determine if there are meaningful changes from A to B. However, there is ongoing debate as to how to interpret effect sizes in single-case analyses, so always exercise caution by not "over-interpreting" these estimates.

The suggestions for effect size cutoffs provided in Box 5 are based on Bloom et al. (2009), who found that single-case effects are typically larger than those found for group comparisons. Among the options provided, the ES and d-index calculations are perhaps the most widely used, but the d-index is generally more conservative (and more defensible).

Hedge's g is the best effect size option if there is a slight trend in the data (see Box 1). In those cases, Hedge's g offers a more conservative effect size estimate that is preferable to both ES or d-index.

##### Box 5. Effect Size Estimates

```{r, echo=FALSE}

Effectsize(myy, myphase, "A", "B")

```

If you report ES, d-index, or Hedge's g, I recommend using the graph in Figure 4 because it visually demonstrates the comparison of levels between the two phases, consistent with the parametric nature of these calculations.

##### Figure 4: Single-Case Results as Measured by Traditional Effect Size Estimates

```{r, echo=FALSE, dpi=300, figure.width=6}

Trendplot <- ggplot(ssdall, aes(x=xrownum, y=myy, colour=myphase)) +

geom\_hline(yintercept=mean(ssd1$myy), color="tomato", size=2\*sd(ssd1$myy), alpha=0.1) +

geom\_hline(yintercept=mean(ssd2$myy), color="turquoise3", size=2\*sd(ssd2$myy), alpha=0.1) +

geom\_vline(linetype=2, xintercept=n1+.5)

Trendplot + geom\_line(size=1) + geom\_point() +

theme(legend.title=element\_blank()) +

labs(x = "Time", y = "Data") +

geom\_segment(aes(x=1, y=mean(ssd1$myy), xend=n1, yend=mean(ssd1$myy)), linetype="longdash", color="tomato") +

geom\_segment(aes(x=n1+1, y=mean(ssd2$myy), xend=n1+n2, yend=mean(ssd2$myy)), linetype="longdash", color="turquoise3")

```

You can also assess the intervention phase relative to what would have been predicted from the baseline data. This concept, called the G-index, is illustrated in Figure 5.

#####Figure 5: Single-Case Results as Measured by the G-index

```{r, dpi=300, echo=FALSE}

Trendplot <- ggplot(ssdall, aes(x=xrownum, y=myy, colour=myphase)) +

geom\_vline(linetype=2, xintercept=n1+.5)

Trendplot + geom\_line(size=1) + geom\_point() +

geom\_smooth(data=subset(ssdall, myphase=="A"), method=lm, se=FALSE, linetype="longdash", fullrange=TRUE, size=0.5, alpha=0.5) +

theme(legend.title=element\_blank()) +

labs(x = "Time", y = "Data")

```

Note, however, that to get an accurate prediction over time, the data need to be collected on a regular basis, so that the time lag from 1 to 2 is roughly equivalent to 2 to 3, etc., along the x-axis (Time). If data collection was inconsistent, you must collapse your data into standard intervals (e.g., days, weeks) in order for the G-index to be computed accurately. Thus, you may need to expand your intervals (e.g., two-week periods), re-enter your data, and rerun this code.

Assuming that data collection was relatively consistent over time, you can estimate a G-index statistic using the results in Box 6. If the intention is to LOWER scores over time, you would look in the "Below Lines" section of Box 6 and get the estimate for "G Regression Line" (if the goal is to raise scores, use the "Above Lines" estimate). Note that positive values of the G-index suggest that an improvement was noted relative to prediction, and negative values suggest that outcomes were worse than predicted. Use the scale at the top of Box 6 to interpret the G-index, provided by Cohen (1988).

#####Box 6. G-Index Statistic

```{r, echo=FALSE}

Gindex(myy, myphase, "A", "B")

```

### B. Non-Parametic Tests: How Much Overlap is there from Phase A to Phase B?

In the vast majority of cases there are problems of outliers, autocorrelation, and trending data, so an alternative is to measure non-overlap of data across the phases. Such tests make no assumptions about distributions, so they are referred to as "nonparametric."

The first nonparametric test is the Percentage of Nonoverlapping Data (PND), which is perhaps the most commonly used form of effect size in the SCD literature. Depending on the intended direction of the intervention effect, PND is calculated as the proportion of B-phase data that exceed either the highest or lowest datum at baseline. Note that the PND is highly sensitive to outliers in the baseline, so if the baseline is unstable, the PND is not your best option. The PND is illustrated in Figure 6.

#####Figure 6. Percentage of Non-Overlapping Data (PND)

```{r, dpi=300, echo=FALSE}

Trendplot <- ggplot(ssdall, aes(x=xrownum, y=myy, colour=myphase)) +

geom\_vline(linetype=2, xintercept=n1+.5)

Trendplot + geom\_line(size=1) + geom\_point() +

geom\_hline(yintercept=max(ssd1$myy), color="tomato", size=0.5, linetype="longdash") +

geom\_hline(yintercept=min(ssd1$myy), color="tomato", size=0.5, linetype="longdash") +

theme(legend.title=element\_blank()) +

labs(x = "Time", y = "Data")

```

If the intent of the intervention is to reduce scores, the results of PND are in Box 7.

#####Box 7. PND Results for Interventions Intended to Lower Scores

```{r, echo=FALSE}

PNDbelow(myy, myphase, "A", "B")

```

If the intent of the intervention is to raise scores, the results of PND are in Box 8.

#####Box 8. PND Results for Interventions Intended to Raise Scores

```{r, echo=FALSE}

PNDabove(myy, myphase, "A", "B")

```

The second nonparametric option for assessing non-overlap is to examine the Percentage of Data Exceeding the Median (PEM). PEM is much less susceptible to outliers in the baseline than is PND, but can be misleading if there is an extreme trend in the baseline. The concept of PEM is illustrated in Figure 7.

##### Figure 7: Percentage of Data Exceeding the Median (PEM)

```{r, dpi=300, echo=FALSE}

Trendplot <- ggplot(ssdall, aes(x=xrownum, y=myy, colour=myphase)) +

geom\_vline(linetype=2, xintercept=n1+.5)

Trendplot + geom\_line(size=1) + geom\_point() +

geom\_hline(yintercept=median(ssd1$myy), color="tomato", size=0.5, linetype="longdash") +

theme(legend.title=element\_blank()) +

labs(x = "Time", y = "Data")

```

If the intention of intervention to lower scores, PEM results are in Box 9.

##### Box 9. PEM Results for Interventions Intended to Lower Scores

```{r, echo=FALSE}

PEMbelow(myy, myphase, "A", "B")

```

If the intention of intervention is raise scores, PEM results are in Box 10.

##### Box 10. PEM Results for Interventions Intended to Raise Scores

```{r, echo=FALSE}

PEMabove(myy, myphase, "A", "B")

```

Two other non-overlapping tests -- the Percentage of All Non-Overlapping Data (PAND) and the Improvement Rate Difference (IRD) -- can offer options in some cases, but those analyses require the removal of data points and that must be done by hand. Readers interested in those procedures are referred to Riley-Tillman and Burns (2009).

### References

Auerbach, C., & Zeitlin, W. (2014). \*SSD for R: An R Package for Analyzing Single-Subject Data\*. New York: Oxford.

Riley-Tillman, T.C., & Burns, M.K. (2009). \*Evaluating Educational Interventions: Single-Case Design for Measuring Response to Intervention.\* New York: Guilford Press.