

Cheat Sheet: dependent t Test

Measurement and Evaluation of HCC Systems

Scenario

Use the t test if you want to test the difference in continuous outcome variable $varY$ between levels A and B of a dichotomous variable $varX$. In this scenario $varX$ is usually an experimental manipulation with two conditions. If each participant receives both conditions (in random order), use a dependent t test. We assume that the data is in the “wide” format, with the two measurements saved as a single row in columns $varY.A$ and $varY.B$.

Power analysis

- Use Test family “ t tests”, “Means: Difference between two dependent means (two groups)”.
- A power analysis has four variables: Effect size, α (usually .05), power (usually .85), and N . If you know three of these, G*Power will calculate the fourth. Select the correct type of power analysis, based on the information you have, and what you want to find out.
- “Tail(s)” depends on your hypothesis. If you have a non-directional hypothesis (i.e. you hypothesize that $A \neq B$, without hypothesizing which one is larger), you should choose Two. Otherwise (i.e. you hypothesize a specific direction), you should choose One.
- By clicking on “Determine”, you can compute the effect size d from the expected mean and standard deviation of the difference.
- Click on “Calculate” to calculate the missing parameter.

(optional) Reshape the data into wide format

- If your data is in the long format (with columns id , $varX$ and $varY$), you have to reshape it into wide format (add additional repeated variables to $v.names$ if needed):

```
data <- reshape(dataLong, idvar="id", timevar="varX", direction="wide",  
v.names=c("varY"))
```

Plotting a bar chart and a box plot

- First remove the between-subjects differences from the data:

```
dataAdjusted <- data  
dataAdjusted$varY.A <- data$varY.A - (data$varY.A + data$varY.B)/2 +  
mean((data$varY.A + data$varY.B)/2)  
dataAdjusted$varY.B <- data$varY.B - (data$varY.A + data$varY.B)/2 +  
mean((data$varY.A + data$varY.B)/2)
```

- Stack the data:

```
dataStack <- stack(dataAdjusted)
names(dataStack) <- c("varY", "varX")
```
- Use the `ggplot2` package to plot a bar chart with error bars.

```
ggplot(dataStack, aes(varX, varY)) + stat_summary(fun.y=mean, geom="bar",
fill="white", color="black") + stat_summary(fun.data=mean_cl_normal,
geom="errorbar", width=0.2)
```
- Plot a boxplot:

```
ggplot(dataStack, aes(varX, varY)) + geom_boxplot()
```

Pre-testing assumptions

- In a dependent t test, the difference in Y should be normally distributed, continuous, and unbounded.
- If your N is small:
 - o Test for significant skewness, kurtosis, and Shapiro-Wilk test on the difference using `stat.desc` in the `pastecs` package and the `by` function.

```
stat.desc(data$varY.B-data$varY.A, basic=F, norm=T)
```
 - o Multiply `skew.2SE` and `kurt.2SE` by 2 to get the Z-scores of skewness and kurtosis. Compare these values to typical cut-off values ($Z > \pm 1.96$: $p < .05$, $Z > \pm 2.58$: $p < .01$, $Z > \pm 3.29$: $p < .001$). The significance of the Shapiro-Wilk test is listed under `normtest.p`.
- If your N is large:
 - o Draw the histogram for the difference in Y , overlaid with a normal curve (using `ggplot2`), and visually inspect whether it follows the normal distribution:

```
ggplot(data, aes(varY.B-varY.A)) + geom_histogram(aes(y=.density.),
binwidth=1, color="black", fill="white") + stat_function(fun = dnorm,
args = list(mean = mean(data$varY.B-data$varY.A), sd = sd(data$varY.B-
data$varY.A)))
```
 - o Change the `binwidth` setting based on what is suitable for your data.
 - o Draw normal a Q-Q plot, and visually inspect whether the data follows the diagonal line:

```
qplot(sample = data$varY.B-data$varY.A, stat="qq")
```
 - o Do the same for `Bdata`.
- In case your data violates normality and/or other assumptions (e.g. outliers), you can conduct a robust test (see below).

Running the test

- Run the t test as follows:

```
result <- t.test(data$varY.A, data$varY.B, paired = T)
```
- Display the result:

```
result
```

- The t -statistic and its p -value tells us whether there is a significant difference between the two conditions. Divide the p -value by 2 if you were conducting a one-sided test.
- The test also produces a confidence interval.
- You can get the effect size r using the formula $r = \sqrt{t^2 / (t^2 + df)}$:


```
t <- result$statistic[[1]]
df <- result $parameter[[1]]
r <- sqrt(t^2/(t^2+df))
round(r, 3)
```
- You can get the effect size d_z using the formula $d_z = M_{\text{difference}}/sd_{\text{difference}}$:


```
mean(data$varY.A - data$varY.B) / sd(data$varY.A - data$varY.B)
```

(optional) Robust versions

- You can use functions in the “source” version of the [WRS2](https://github.com/nicebread/WRS2) package to run trimmed and/or bootstrapped versions of the dependent t test.
- Load WRS: (see installation instructions at <https://github.com/nicebread/WRS>)
- Load the “source” version of WRS2:


```
install.packages("WRS2", type="source")
```
- Robust t test using 10% trimmed means (change the percentage if desired):


```
yuend(data$varY.A, data$varY.B, tr = 0.1)
```
- Robust t test using 10% trimmed means and 2000 bootstrap samples:


```
ydbt(data$varY.A, data$varY.B, tr = 0.1, nboot = 2000)
```
- Robust t test using an M-estimator:


```
bootdpci(data$varY.A, data$varY.B, est = mom, nboot = 2000)
```

Reporting

- Use the following format to report on a dependent t test (replace the full names (not just the variable names) of A, B and varY, and replace the xx'es with the actual numbers):

“On average, participants in the [A] condition experienced a [higher/lower] level of [varY] ($M = xx$, $SE = xx$) than participants in the [B] condition ($M = xx$, $SE = xx$), $t(xx) = x.xx$, $p = .xxx$, $r = .xxx$.”