# Cheat Sheet: independent *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 (but it can be any dichotomous variable, such as gender, etc). If each participant is randomly assigned to one condition, use an independent *t* test.

## Power analysis

- Use Test family "*t* tests", "Means: Difference between two independent means (two groups)".
- A power analysis has four variables: Effect size, $\alpha$ (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 ≠ B, without hypothesizing which one is larger), you should choose Two. Otherwise (i.e. you hypothesize a specific direction), you should choose One.
- "Allocation ratio N2/N1" is the ratio of number of participants that receive each condition. Usually we try to balance this, which makes the allocation ratio 1.
- By clicking on "Determine", you can compute the effect size *d* from the expected mean and standard deviation in each group.
- Click on "Calculate" to calculate the missing parameter.

## Plotting a bar chart and a box plot

- Use the `ggplot2` package to plot a bar chart with error bars.
  ```
  ggplot(data, 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(data, aes(varX, varY)) + geom_boxplot()
  ```

## Pre-testing assumptions

- In a *t* test, *Y* should be independent, continuous, and unbounded. The error variance should be normally distributed, which is true if *Y* is normally distributed for each level of *X*.
- If your *N* is small:
  - Test for significant skewness, kurtosis, and Shapiro-Wilk test within each group using `stat.desc` in the `pastecs` package and the `by` function.
    ```
    by(data$varY, data$varX, stat.desc, basic=F, norm=T)
    ```
  - 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 ($Z > \pm1.96$: $p < .05$, $Z > \pm2.58$: $p < .01$, $Z > \pm3.29$: $p < .001$). The significance of the Shapiro-Wilk test is listed under `normtest.p`.
- If your *N* is large:
  - Subset the data into two groups based on `varX`:
    ```
    Adata <- subset(data, data$varX == "A")
    Bdata <- subset(data, data$varX == "B")
    ```
  - Draw the histogram for `varY` in `Adata`, overlaid with a normal curve (using `ggplot2`), and visually inspect whether it follows the normal distribution:
    ```
    ggplot(Adata, aes(varY)) + geom_histogram(aes(y=..density..),
    binwidth=1, color="black", fill="white") + stat_function(fun = dnorm,
    args = list(mean = mean(Adata$varY), sd = sd(Adata$varY)))
    ```
  - Change the `binwidth` setting based on what is suitable for your data.
  - Draw normal a Q-Q plot, and visually inspect whether the data follows the diagonal line:
    ```
    qplot(sample = Adata$varY, stat="qq")
    ```
  - Do the same for `Bdata`.
- If your data has positive skew, and your data only has positive values, you can possibly fix this by transforming your *Y* variable:
  - Log transform:
    ```
    data$varYlog <- log(data$varY + 1)
    ```
  - Or, square root transform:
    ```
    data$varYsqrt <- sqrt(data$varY)
    ```
- In other cases of violations of assumptions (e.g. outliers), you can conduct a robust test (see below).

## Running the test

- Run the *t* test as follows:
  ```
  result <- t.test(varY ~ varX, data = data)
  ```
- Display the result:
  ```
  result
  ```

- The *t*-statistic and its *p*-value tells us whether there is a significant difference between the two groups. 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* using the `cohen.d` function in the `effsize` package:

```
cohen.d(varY ~ varX, data = data)
```

## (optional) Robust versions

- You can use functions in the `WRS2` package to run trimmed and/or bootstrapped versions of the *t* test.
- Robust *t* test using 10% trimmed means (change the percentage if desired):

```
yuen(varY ~ varX, data = data, tr = 0.1)
```

- Robust *t* test using 10% trimmed means and 2000 bootstrap samples:

```
yuenbt(varY ~ varX, data = data, tr = 0.1, nboot = 2000)
```

- Robust *t* test using an M-estimator:

```
pb2gen(varY ~ varX, data = data, est = “mom”)
```

## Reporting

- Use the following format to report on an independent *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."
- If varX presents a trait (e.g. gender) rather than an experimental manipulation, "participants in the [A] condition" should be replaced with "[A] participants".