Permutation Tests

Livio Finos

University of Padova

Contents

1 Introduction

1.1 Introduction

- Well-established nonparametric inference approach: Fisher (1935); Pitman (1937, 1938).
- Generally requires fewer assumptions about the data generating process than parametric counterparts.
- Excellent inferential properties, typically:
  - Exactness (exact control of Type I error)
  - Asymptotic optimality and convergence to parametric counterparts when they exist.
- Fisher's exact test is a prototypical example, but
- The general approach had limited applicability without computational support.

1.2 Renewed Interest in Permutation Testing

- A milestone: Westfall and Young (1993). *Resampling-Based Multiple Testing: Examples and Methods for p-value Adjustment*. Wiley.
- Many active research areas adopt these methods in daily statistical analysis (e.g., genetics and neuroscience: Nichols and Holmes (2002); Pantazis et al. (2009); Winkler et al. (2014)).
- Permutation approach:
  - Ideal for **randomized experimental designs**
  - Handles complex models without formal definition of the data generating process.

1.3 The `flip` Package

Available on CRAN and GitHub (https://github.com/livioivil/flip).

To install the GitHub version in R:

```
library(devtools)
install_github('livioivil/flip')
```

Before starting

```
# Clean memory
rm(list = ls())

# Customize graph output
par.old <- par()
par(cex.main = 1.5, lwd = 2, col = "darkgrey", pch = 20, cex = 3)
palette(c("#FF0000", "#00A08A", "#FFCC00", "#445577", "#45abff"))

# Customize knitr output
knitr::opts_chunk$set(fig.align = "center")  # fig.width=6, fig.height=6
```

1.4 The Age vs Reaction Time Dataset

Subjects' reaction times were tested by having them grab a meter stick after it was released. The number of centimeters the meter stick dropped before being caught directly measures response time.

`Age` values are in years. `Gender` is coded as `F` for female and `M` for male. `Reaction.Time` values are in centimeters.

(Data are fictitious)

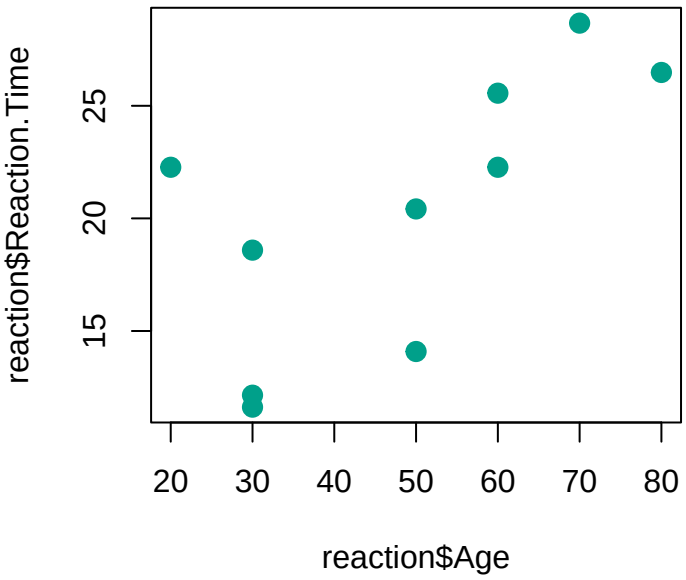
To read the data:

```
data(reaction, package = "flip")
# Alternatively, download from: https://github.com/livioivil/flip/tree/master/data
# or load("reaction.rda")
str(reaction)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    10 obs. of  3 variables:
##  $ Age          : num  70 50 30 60 80 60 30 30 20 50
##  $ Gender       : Factor w/ 2 levels "F","M": 1 1 2 1 2 1 2 2 1 2
##  $ Reaction.Time: num  28.7 20.4 11.6 22.3 26.5 ...
```

Plot the data:

```
plot(x = reaction$Age, y = reaction$Reaction.Time, pch = 20, col = 2, cex = 2)
```



1.5 Measuring Dependence Between Two Variables

Define: - $X = Age$
- $Y = Reaction.Time$

Review common indices for measuring (linear) dependence between two variables.

1.5.1 Covariance and Variance

Covariance between X and Y:
$\sigma_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n}$

- Values between $-\infty$ and ∞
- $\sigma_{xy} \approx 0$: No dependency between X and Y
- $\sigma_{xy} \gg 0$ ($\ll 0$): Strong positive (negative) dependency

Variance of X:
$\sigma_{xx} = \sigma_x^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}$
Standard Deviation of X:
$\sigma_x = \sqrt{\sigma_{xx}}$

1.5.2 Correlation

Covariance alone makes it difficult to assess relationship strength. Note that:
$-\sigma_x\sigma_y \leq \sigma_{xy} \leq \sigma_x\sigma_y$
is equivalent to:
$-1 \leq \frac{\sigma_{xy}}{\sigma_x\sigma_y} \leq 1$

Correlation between X and Y:

$\rho_{xy} = \frac{\sigma_{xy}}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$

- Values between -1 and 1
- $\rho_{xy} \approx 0$: No dependency
- $\rho_{xy} \approx 1$ (-1): Strong positive (negative) dependency

1.5.3 Linear Trend: Least Squares Method

Describe the relationship between `Reaction.Time` and `Age` with a straight line:

$E(Reaction.Time) \approx \beta_0 + \beta_1 Age$
$E(Y) = \beta_0 + \beta_1 X$

Draw a line through the center of the data.

Least-squares estimator: Find parameters minimizing the sum of squared residuals:

Find $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize: $\sum_{i=1}^{n}(y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i))^2$

Estimates:
- Slope: $\hat{\beta}_1 = \frac{\sigma_{xy}}{\sigma_{xx}} = \rho_{xy}\frac{\sigma_y}{\sigma_x} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = 0.2064719$
- Intercept: $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} = 10.3013483$ - Estimated response: $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$ - Residuals: $y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i) = y_i - \hat{y}_i$

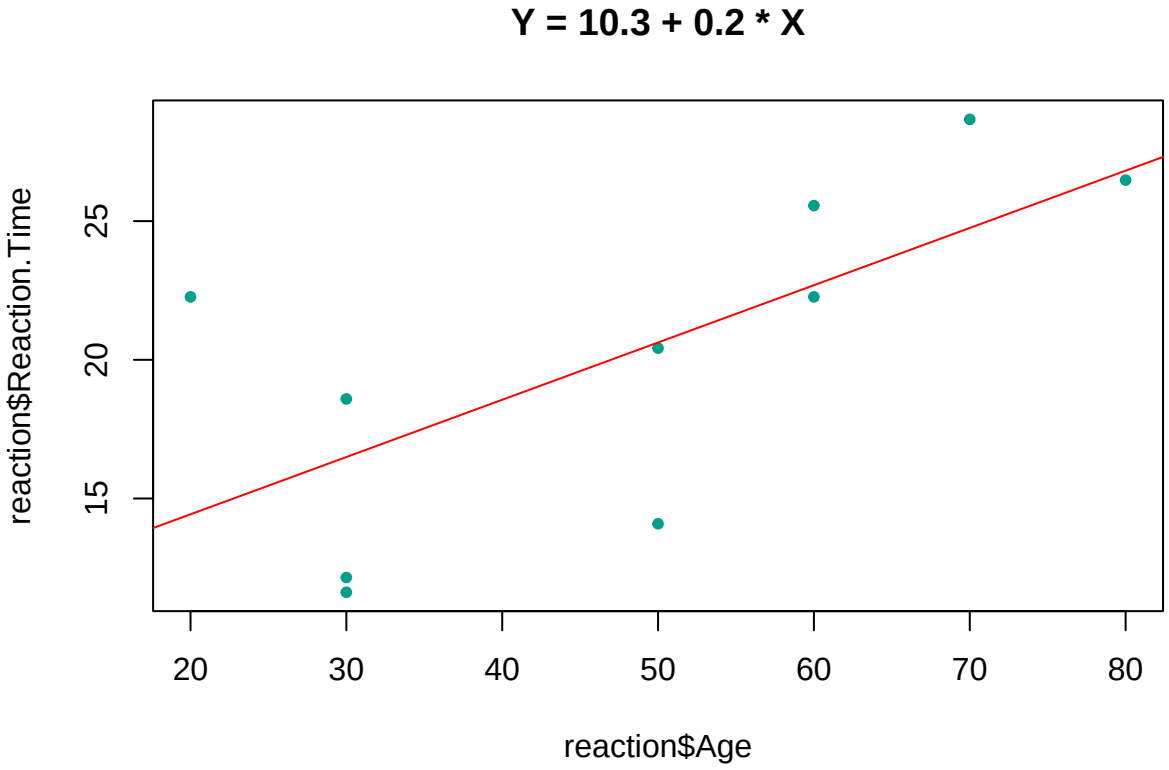
Sum of squared residuals: $\sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2 = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

Visual representation:

```
model <- lm(Reaction.Time ~ Age, data = reaction)
coefficients(model)
```

```
## (Intercept)         Age
##  10.3013483   0.2064719
```

```
plot(reaction$Age, reaction$Reaction.Time, pch = 20, col = 2, cex = 1)
coeff <- round(coefficients(model), 1)
title(paste("Y =", coeff[1], "+", coeff[2], "* X"))
abline(model, col = 1)
```



2 Permutation Approach to Hypothesis Testing

2.0.1 Preliminary Remarks

Note that all measures above make no assumptions about the random process generating the data.

Now assume Y (and possibly X) is generated by a random variable. Further minimal assumptions will be specified later.

Question: Is there a relationship between Y and X?

We estimated $\hat{\beta}_1 = 0.2064719$

But is the **true value** β_1 actually different from 0 (indicating no relationship)? Or is the difference from 0 due to random sampling?

- **Null Hypothesis** $H_0 : \beta_1 = 0$ (the **true** β_1, not its estimate $\hat{\beta}_1$!). No relationship between X and Y.
- **Alternative Hypothesis** $H_1 : \beta_1 > 0$ (positive relationship).

Other possible H_1 specifications: $\beta_1 < 0$ or, more commonly, $\beta_1 \neq 0$.

2.1 Permutation Tests in a Nutshell

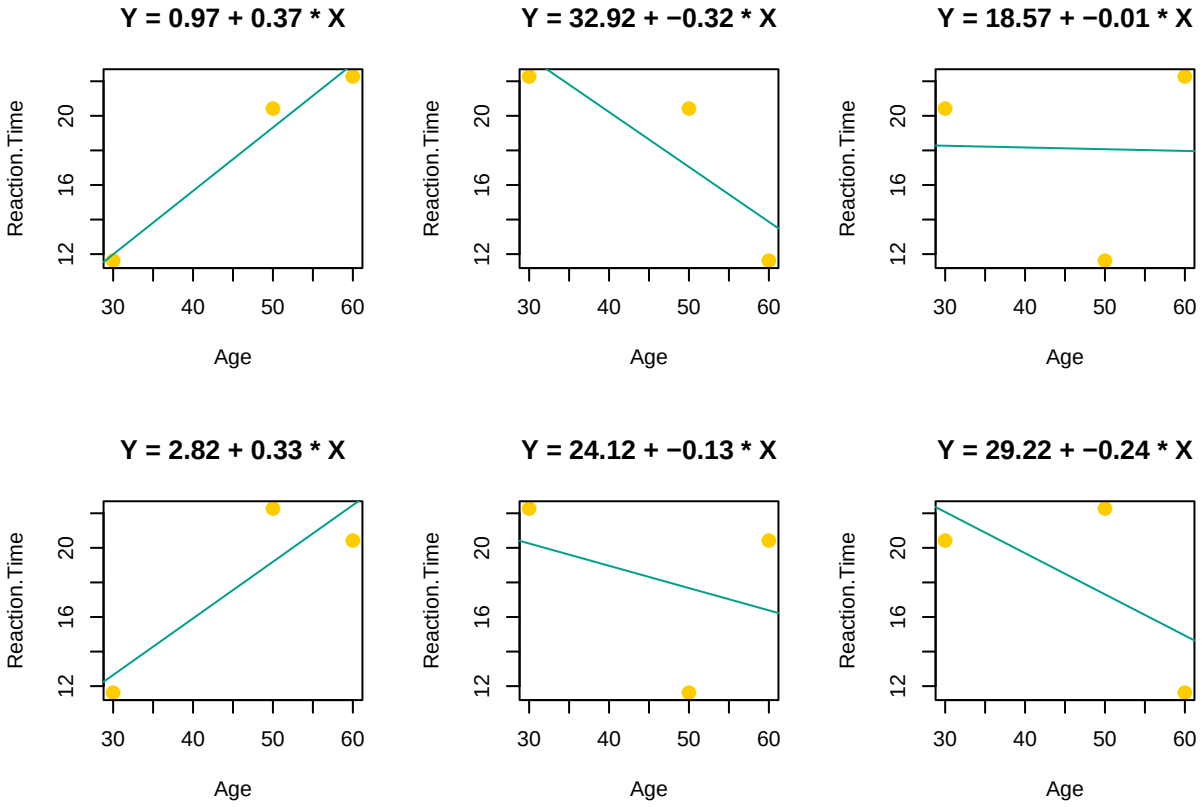
As a toy example, use a data subset:

```
##   Age Gender Reaction.Time
## 2  50      F         20.42
## 3  30      M         11.62
## 4  60      F         22.27
```

- *If H_0 is true*: No linear relationship between X and Y
- Observed trend is due to chance
- Any other pairing of x_i and y_i was equally likely
- Generate hypothetical datasets by permuting Y observations
- How many equally likely datasets exist with observed X and Y? $3 \times 2 \times 1 = 3! = 6$ possible datasets.

Remark: We only assume Y is a random variable. The key assumption is exchangeability: the joint density $f(y_1, \ldots, y_n)$ is invariant to permutations of $y_1, \ldots, y_n$.

2.1.1 All Potential Datasets



2.1.1.1 In Our Complete Dataset Apply the same principle to the full dataset...

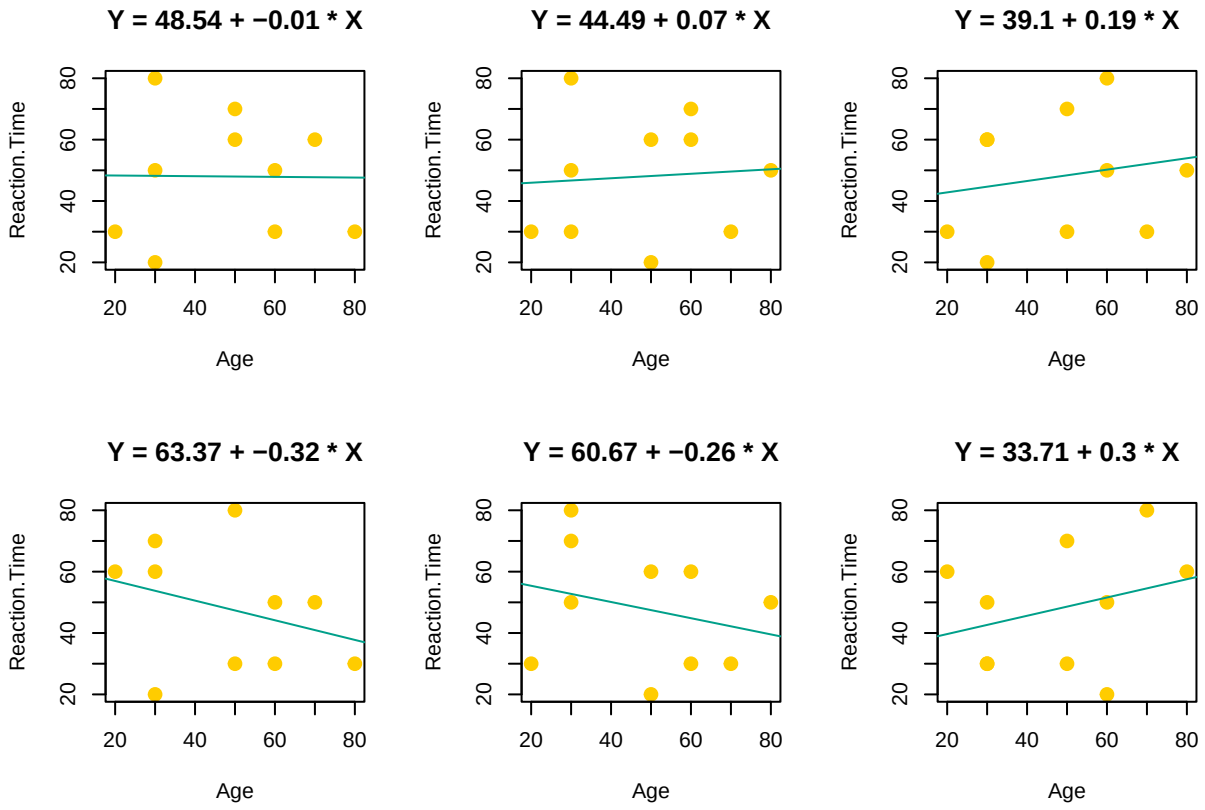
How many permutations of $y_1, \ldots, y_n$ are possible? $n! = 10! = 3,628,800$.

Manageable, but with $n = 20$? $20! = 2.43 \times 10^{18}$ - too large!

Calculate a smaller but sufficiently large number B of random permutations.

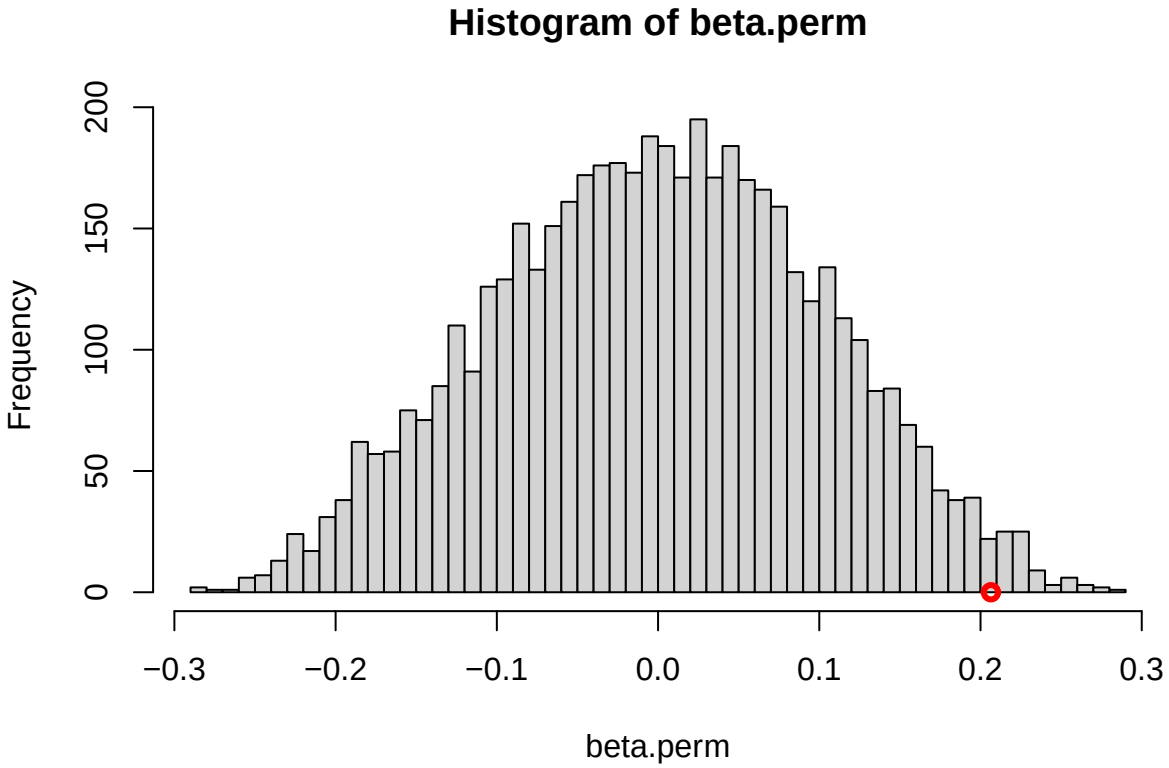
Examples:

`Age` vs Permuted `Reaction.Time`



Repeat 5000 times and examine the $\hat{\beta}_1$ histogram:

```
# beta_1 estimated on observed data
beta1 <- coefficients(lm(Reaction.Time ~ Age, data = reaction))[2]

# Function to permute y and calculate beta_1
my.beta.perm <- function(Y, X) {
  model <- lm(sample(Y) ~ X)
  coefficients(model)[2]
}

# Replicate B-1 times
beta.perm <- replicate(B, my.beta.perm(reaction$Reaction.Time, reaction$Age))
```



2.1.2 How Likely Was $\hat{\beta}_1^{obs}$?

(before the experiment!)

What was the probability of obtaining a value $\geq \hat{\beta}_1^{obs}$ among possible $\hat{\beta}_1^{*b}$ values (from permuted data)?

Remarks: - $\hat{\beta}_1^{*b} < \hat{\beta}_1^{obs}$ (closer to 0): Less evidence against H_1 than $\hat{\beta}_1^{obs}$ - $\hat{\beta}_1^{*b} \geq \hat{\beta}_1^{obs}$: Equal or stronger evidence for H_1 than $\hat{\beta}_1^{obs}$

2.1.3 P-value Calculation

Out of B = 5000 permutations, we obtained 4921 cases where $\hat{\beta}_1^{*b} \leq \hat{\beta}_1^{obs}$.

The p-value (significance) is: $p = \frac{\#(\hat{\beta}_1^{*b} \geq \hat{\beta}_1^{obs})}{B} = 0.0162$

($\hat{\beta}_1^{obs}$ counts as one random permutation)

2.1.4 Interpretation

The probability $P(\hat{\beta}_1^* \geq \hat{\beta}_1 = 0.206 \,|H_0) = p = 0.0162$ is very small. It was unlikely to obtain such a value **if H_0 is true**.

The Neyman-Pearson approach established significance thresholds like $\alpha = .05$ (or $= .01$). When $p \leq \alpha$, we reject H_0 (no relationship). We then conclude H_1 is likely true (positive relationship).

- Type I error: False Positive
  True hypothesis is H_0 (no correlation), but we accept H_1 (positive correlation)
- Type II error: False Negative
  True hypothesis is H_1 (positive correlation), but we fail to reject H_0 (no correlation)

2.2 Summary

p-value: Proportion of experiments providing equal or stronger evidence against H_0 compared to observed data.

To compute it, we need: - **Orbit** $\mathcal{O}$ - **Test statistic** ($T : \mathbb{R}^n \to \mathbb{R}$) quantifying evidence against H_0 - Higher values indicate stronger evidence against H_0 - Computing T for each $\mathcal{O}$ element induces an ordering on $\mathcal{O}$

In our example: $T = \hat{\beta}_1 = \hat{\sigma}_{xy}/\hat{\sigma}_{xx}$ (estimated slope). Higher slope indicates stronger evidence for H_1.

Type I Error Control

We want to limit false discoveries (be conservative). Bound the probability of false discovery:

$P(\text{p-value} \leq \alpha | H_0) \leq \alpha$

This ensures that over many experiment replications, we find false correlations with probability α (e.g., $0.05 = 5\%$).

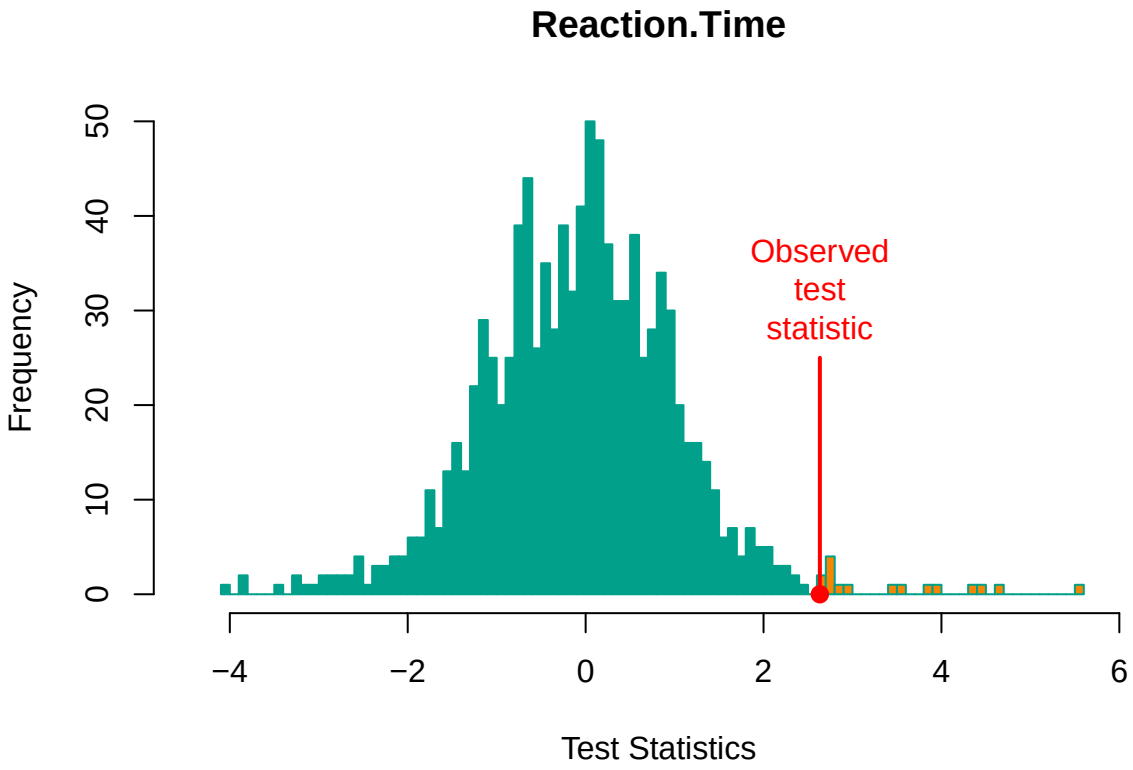
2.2.1 Implementation in `flip`

```
library(flip)
(res <- flip(Reaction.Time ~ Age, data = reaction, tail = 1))
```

```
##
##               Test  Stat tail p-value
## Reaction.Time    t 2.633    >  0.0150
```

```
# Compare with:
# flip(Reaction.Time ~ Age, data = reaction, tail = 1, statTest = "cor")
# flip(Reaction.Time ~ Age, data = reaction, tail = 1, statTest = "coeff")
```

```
plot(res)
```



Type I Error Control

We want to guarantee few false positives. Bound false discovery probability:

$P(\text{p-value} \leq \alpha | H_0) \leq \alpha$

This ensures long-run false correlation discovery rate $\leq \alpha$ (e.g., 5%).

2.2.2 Two-sided Alternatives

$H_1 : \beta_1 > 0$ (positive relationship) requires a priori justification.

More commonly, the two-sided alternative is appropriate: $H_1 : \beta_1 \neq 0$ (relationship exists, direction unspecified)

We consider both very small and very large estimated coefficients as anomalous ('far from 0').

P-value: $p = \frac{\#(|\hat{\beta}_1^{*b}| \geq |\hat{\beta}_1^{obs}|)}{B} = 0.0326$

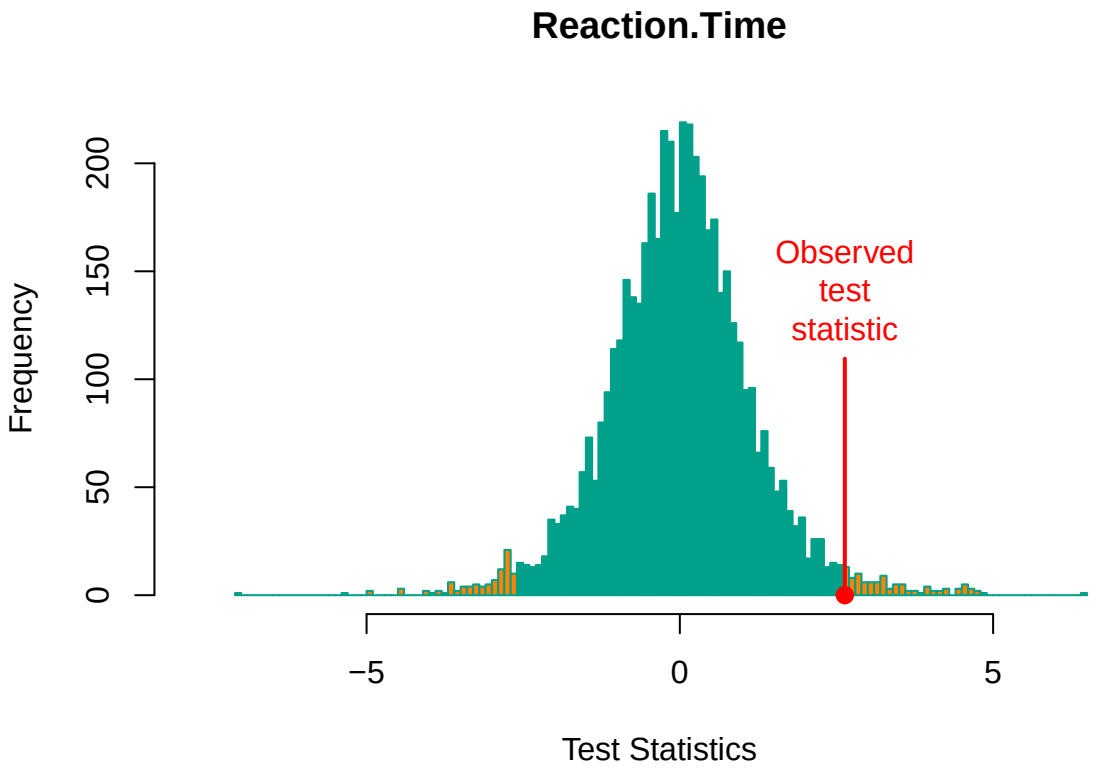
(Note: observed test statistic included among permutations)

In `flip`:

```
library(flip)
(res <- flip(Reaction.Time ~ Age, data = reaction, tail = 0, perms = 5000))
```

```
##
##               Test  Stat tail p-value
## Reaction.Time    t 2.633   ><  0.0374
```

```r
plot(res)
```



2.3 A More Formal Approach

(See also Pesarin, 2001; Hemerik & Goeman, 2018)

Let Y be data in sample space $\mathcal{Y}$. Let Π be a (usually finite) set of transformations $\pi : \mathcal{Y} \to \mathcal{Y}$ forming a **group** under composition: - Contains identity - Every element has an inverse - Closed: if $\pi_1, \pi_2 \in \Pi$, then $\pi_1 \circ \pi_2 \in \Pi$

(e.g., Π = all possible permutations)

Null Hypothesis
$H_0 : Y \sim P \in \Omega_0$

Randomization Hypothesis Under H_0, the distribution of Y is invariant under Π: for every $\pi \in \Pi$, πY and Y have the same distribution when $Y \sim P \in \Omega_0$.

(See also Lehmann & Romano, 2006. *Testing Statistical Hypotheses*. Springer.)

Test statistic $T(Y) : \mathbb{R}^n \to \mathbb{R}$

$T^{(k)}(Y)$ is the $\lceil (1-\alpha)|\Pi| \rceil$-th sorted value of $T(\pi Y)$

Define the test:

$$\phi(Y) = \begin{cases} 1 & \text{if } T(Y) > T^{(k)}(Y) \\ 0 & \text{otherwise} \end{cases}$$

Theorem: Under H_0, $E_P(\phi(Y)) = \alpha$, i.e., $P(T(Y) > T^{(k)}) \leq \alpha$.

Proof

For each $Y \in \mathcal{Y}$, define orbit $O_Y = \{\pi Y : \pi \in \Pi\} \subseteq \mathcal{Y}$.

Let $A = \{Y \in \mathcal{Y} : T(Y) > T^{(k)}(Y)\}$ be the rejection set. Under H_0, by group structure, $\Pi\pi = \Pi$ for all $\pi \in \Pi$, so $T^{(k)}(\pi Y) = T^{(k)}(Y)$ for all $\pi \in \Pi$. Thus:

$$\#\{\pi \in \Pi : \pi Y \in A\} = \#\{\pi \in \Pi : T(\pi Y) > T^{(k)}(\pi Y)\} = \#\{\pi \in \Pi : T(\pi Y) > T^{(k)}(Y)\} \leq \alpha |O_Y|$$

Endow orbits with inherited σ-algebra. As in Lehmann (2005, Theorem 15.2.2):

$$P(Y \in A | O_Y) = \frac{|A|}{|O_Y|}$$

Bounded by α. Hence:

$$P(Y \in A) = \mathbb{E}\{P(Y \in A | O_Y)\} \leq \alpha$$

Alternative Proof

By construction, $\sum_{\pi \in \Pi} \phi(\pi Y) = |\Pi|\alpha$. Thus:

$$|\Pi|\alpha = E_P\left(\sum_{\pi \in \Pi} \phi(\pi Y)\right) = \sum_{\pi \in \Pi} E_P(\phi(\pi Y))$$

By null hypothesis: $E_P(\phi(Y)) = E_P(\phi(\pi Y))$, so:

$$|\Pi|\alpha = \sum_{\pi \in \Pi} E_P(\phi(Y)) = |\Pi| E_P(\phi(Y))$$

giving $E_P(\phi(Y)) = \alpha$.

2.3.1 Further Notes on Permutation Testing

Orbit: $\mathcal{O} = \{\pi Y : \pi \in \Pi\} \subseteq \mathcal{Y}$

i.e. $\mathcal{O} = \{\text{all permutations of observed data } \mathbf{y}\} = \{\mathbf{y}^* : \pi^* \circ \mathbf{y}\}$

2.3.1.1 Exchangeability **Exchangeability Assumption**: Under H_0, observations are exchangeable: e.g., $f(y_1, y_2) = f(y_2, y_1)$.

Therefore the $\mathcal{O}$ is a set of samples in $\mathcal{Y}$ sharing the same likelihood under the null hypothesis: $\mathcal{O} = \{\pi \mathbf{y} : f_{H_0}(\pi \mathbf{y}) = f_{H_0}(\mathbf{y})\}$

($|\mathcal{O}|$ = number of elements)

With exchangeability:

Proof Intuition (alternative Type I error control proof):

$$f(\mathbf{y}|\mathcal{O}) = \frac{f(\mathbf{y} \cap \mathcal{O})}{f(\mathcal{O})} = \frac{f(\mathbf{y})}{f(\mathcal{O})} = \frac{f(\mathbf{y})}{f(\cup_{y \in \mathcal{O}} y)} = \frac{1}{|\mathcal{O}|} \; \forall \; \mathbf{y} \in \mathcal{O}$$

Each permutation is equally likely in $\mathcal{O}$ (due to group structure).

$$\begin{aligned} E(\phi(Y)|\mathbf{y} \in \mathcal{O}, H_0) &= P(T(\mathbf{y}) > T^{(k)}|\mathbf{y} \in \mathcal{O}, H_0) \\ &= \int_{T^{(k)}}^{+\infty} f(T(\mathbf{y})) dT(\mathbf{y}) \\ &= \sum_{\mathbf{y} \in \mathcal{O}} I(T(\mathbf{y}) > T^{(k)})/|\mathcal{O}| \leq \alpha \quad \forall \mathcal{O} \end{aligned}$$

Then: $E(\phi(\mathbf{y})) = \int_P E(\phi(\mathbf{y})|\mathbf{y} \in \mathcal{O}, H_0) d\mathbf{y}$

2.3.1.2 Independence vs exchangeability

Always true if observations:
- Are **identically distributed** - Have **same dependence** (e.g., same correlation)

Parametric t-tests and linear models assume independence (stricter than 'same dependence') and normality of errors—more stringent than permutation approach.

When normality fails, parametric approaches only provide asymptotic Type I error control, while permutation provides exact control.

2.3.2 Properties (see Pesarin, 2001)

The theorem proves permutation tests have **exact Type I error control**: $P(\text{p-value} \leq \alpha | H_0) = \alpha$, assuming $\alpha \in \{1/|\mathcal{O}|, 2/|\mathcal{O}|, \ldots, 1\}$ (since $\mathcal{O}$ is finite and $T(\pi\mathbf{y})$ distribution is a step function). For other α values, tests are slightly conservative (or require randomized tests, not discussed here).

Additional properties: - **Unbiased**: $P(\text{p-value} \leq \alpha | H_1) > \alpha$ - **Consistent**: $P(\text{p-value} \leq \alpha | H_1) \to 1$ as $n \to \infty$ - Converges to parametric counterpart when it exists

2.3.3 Estimated p-values

In practice, the p-value is often *estimated* using random permutations when it is computationally infeasible to compute the exact permutation p-value based on the entire permutation group.

Random permutations are typically drawn uniformly from the orbit $\mathcal{O}$ without replacement. In this Section, p denotes the exact p-value computed from the entire orbit $\mathcal{O}$, and $\hat{p}$ denotes its estimate computed from B randomly sampled permutations.

If we force the first element to be the observed test statistic $T(Y)$, then $T(Y)$ becomes over-represented in the sample of B elements from $\mathcal{O}$. Consequently, $E(\hat{p}) > p$, although $\lim_{B\to\infty} \hat{p} = p$.

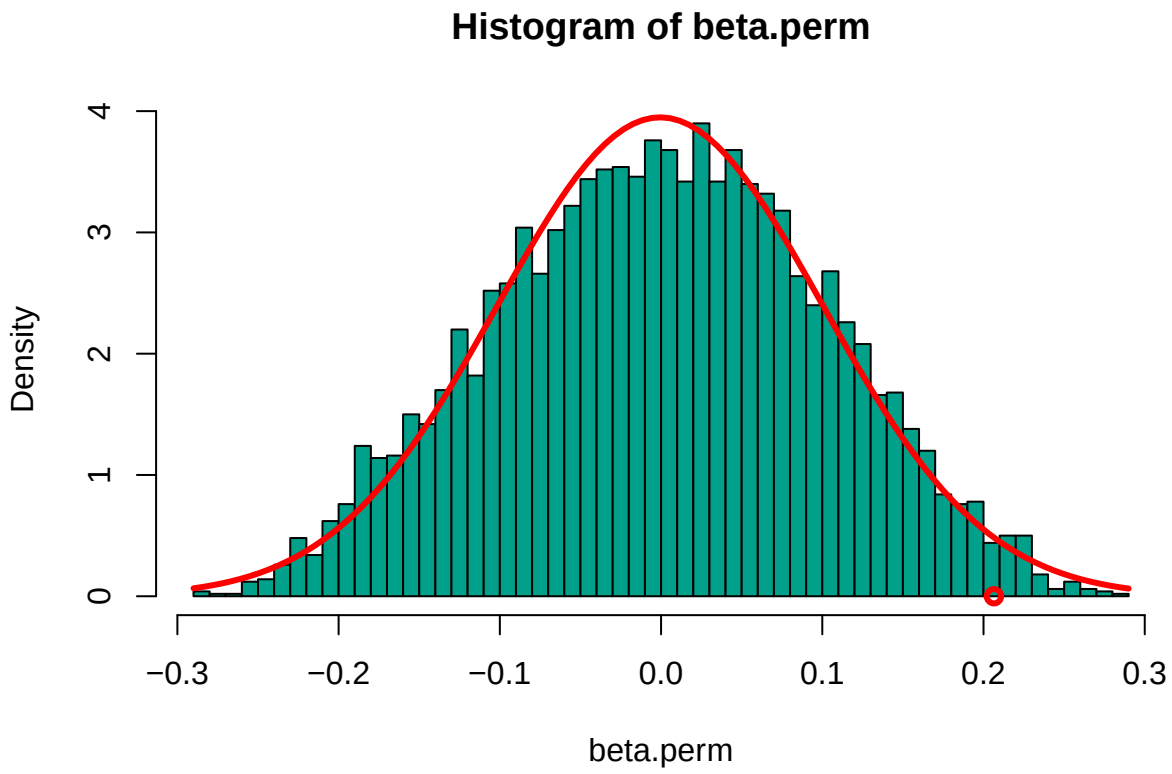
An unbiased estimator of p, denoted $\hat{p}_0$, can be obtained by removing the constraint that $T(Y)$ must be included in the sample (i.e., by sampling permutations without including the observed statistic by default). This ensures $E(\hat{p}_0) = p$. However, as Phipson and Smyth (2010) thoroughly explain, using this unbiased estimator can be problematic because $\hat{p}_0$ is almost never stochastically larger than a uniform distribution on $[0, 1]$ under H_0. This is evident from the fact that $\hat{p}_0$ typically has a positive probability of being exactly zero.

In any case, when computationally feasible, computing exact p-values is always preferable to estimating them.

2.4 Comparison with Parametric Linear Models

The histogram of test statistics from permuted data is well approximated by a **Gaussian** curve.

```
hist(beta.perm, 50, probability = TRUE, col = 2)
curve(dnorm(x, mean(beta.perm), sd(beta.perm)), add = TRUE, col = 1, lwd = 3)
points(beta1, 0, lwd = 3, col = 1)
```



2.4.1 Simple Linear Parametric Model

Assume observed values distribute around true values $\beta_0 + \beta_1 X$ according to a Gaussian distribution:

$Y = \text{linear part} + \text{normal error}$

$Y = \beta_0 + \beta_1 X + \varepsilon$

Linear model assumptions: - $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$ (true linear relationship plus error) - $\varepsilon_i \sim N(0, \sigma^2)$ for all $i = 1, \ldots, n$ (normal errors with zero mean and constant variance/homoscedasticity)

2.4.2 Hypothesis Testing

If assumptions hold:

$\hat{\beta}_1 \sim N(\beta_1, \sigma^2 / \sum (x_i - \bar{x})^2)$

Test statistic:

$$t = \frac{\hat{\beta}_1}{\text{std.dev}(\hat{\beta}_1)} = \frac{\hat{\beta}_1}{\sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum (x_i - \bar{x})^2} / (n-2)}}$$

If $H_0 : \beta_1 = 0$ is true, $t \sim t(n-2)$.

For `reaction` data with $H_1 : \beta_1 \neq 0$ (two-sided):

```
model <- lm(Reaction.Time ~ Age, data = reaction)
summary(model)
```

```
##
## Call:
```

```
## lm(formula = Reaction.Time ~ Age, data = reaction)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.535 -3.364 -0.272  2.676  7.839
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.30135    4.04407   2.547   0.0343 *
## Age          0.20647    0.07841   2.633   0.0300 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.678 on 8 degrees of freedom
## Multiple R-squared:  0.4643, Adjusted R-squared:  0.3973
## F-statistic: 6.934 on 1 and 8 DF,  p-value: 0.03003
```

Similar result, but with many more assumptions!

2.4.3 Permutation Test Assumptions

What model do permutation tests assume?

Under H_0: $f(y) = f(y|x)\ \forall x$

Under H_1: No assumptions. For power, we hope: $H_1 : E(y|x) = \beta_0 + \beta_1 x$ with $\beta_1 \neq 0$ for some x
i.e., $H_1 : E(yx) \neq E(x)E(y)$

No other assumptions about $f(y|x)$ distribution (normality, finite moments, etc.).

2.5 Permutationally Equivalent Tests

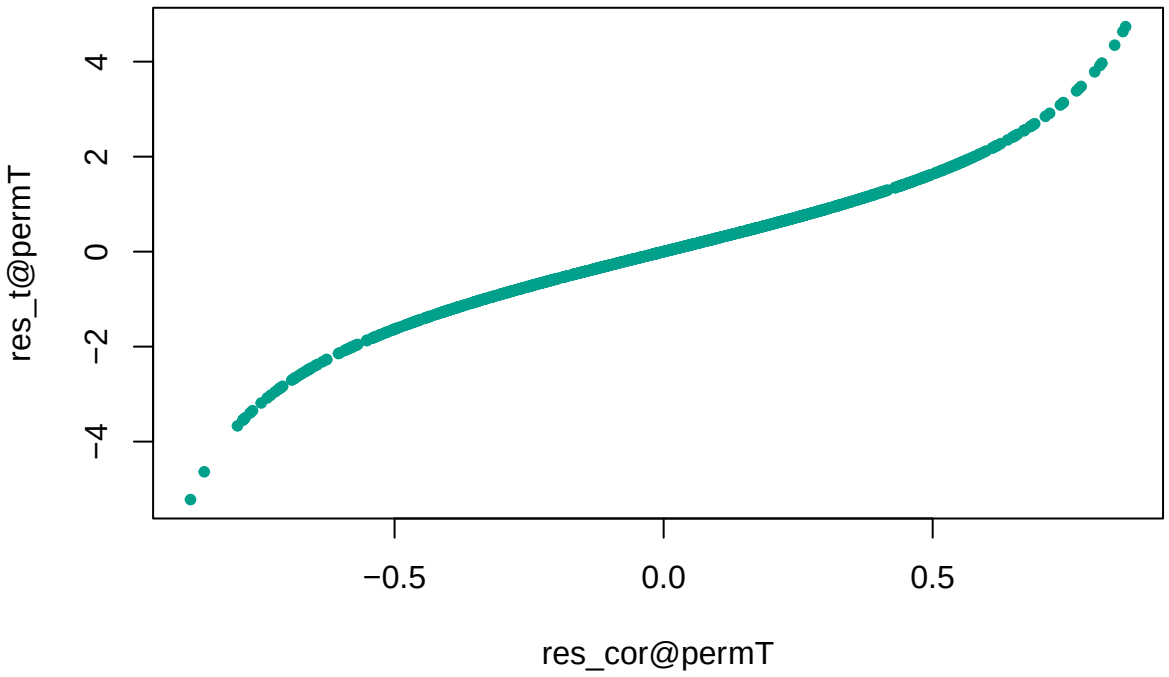
```
set.seed(1)
(res_cor <- flip(Reaction.Time ~ Age, data = reaction, statTest = "cor"))
```

```
##
##               Test   Stat tail p-value
## Reaction.Time  cor 0.6814   ><  0.0410
```

```
set.seed(1)
(res_t <- flip(Reaction.Time ~ Age, data = reaction, statTest = "t"))
```

```
##
##               Test  Stat tail p-value
## Reaction.Time    t 2.633   ><  0.0410
```

```
plot(res_cor@permT, res_t@permT, pch = 20, col = 2)
```



2.5.1 Conclusion

Permutation tests: - Different from bootstrap methods (permutation: without replacement; bootstrap: with replacement). Permutation tests have optimal properties and (usually) exact Type I error control. - General approach applicable in many contexts with minimal assumptions. - Dedicated R packages:
* `coin` http://cran.r-project.org/web/packages/coin/index.html
* `permuco` https://cran.r-project.org/web/packages/permuco/index.html
* `flip` http://cran.r-project.org/web/packages/flip/index.html (development: https://github.com/livioivil/flip)
* `flipscores` http://cran.r-project.org/web/packages/flipscores/index.html (development: https://github.com/livioivil/flipscores)
* `multcomp` https://cran.r-project.org/web/packages/multcomp/index.html
* `GFD` https://cran.r-project.org/web/packages/GFD/index.html

3 Special Cases

3.1 Rank Correlation

- n observations of y, interest in $F(y|x)$
  - Don't need y_1 and y_2 to be continuous or have finite moments
- Hypotheses:
  - $H_0 : F(y|x) = F(y|x') \ \forall x, x'$
  - $H_1 : \exists x < x' : F(y|x) < F(y|x')$ or directional alternatives
  - Test statistic: rank correlation

```
  (res <- flip(Reaction.Time ~ Age, data = reaction, perms = 5000, statTest = "rank"))
```

```
##
##                   Test  Stat tail p-value
## Reaction.Time Wilcoxon 2.189   ><  0.0204
```

```
  # Alternative using rank transformation:
  (res <- flip(rank(reaction$Reaction.Time) ~ rank(reaction$Age), perms = 5000, statTest = "cor"))
```

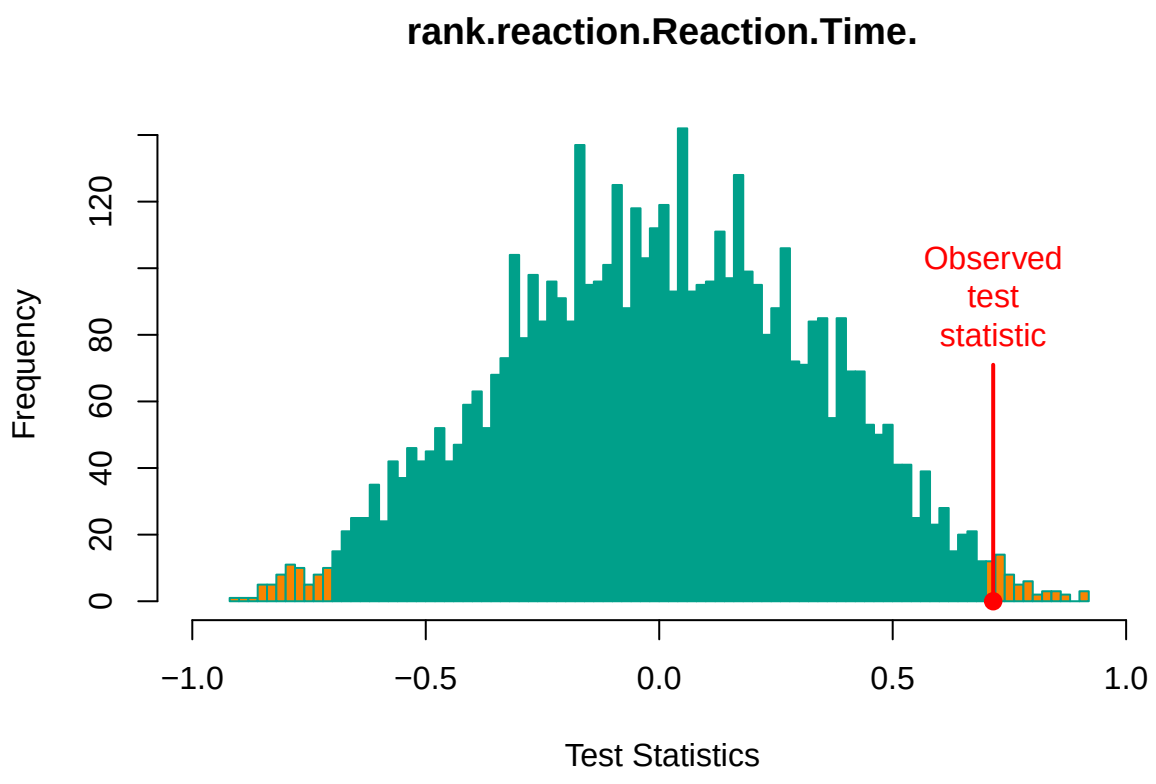
```
##
##                               Test   Stat tail p-value
## rank.reaction.Reaction.Time.  cor 0.7153   ><  0.0222
```

```
  (cor.test(reaction$Reaction.Time, reaction$Age, method = "spearman"))
```

```
## Warning in cor.test.default(reaction$Reaction.Time, reaction$Age, method =
## "spearman"): Impossibile calcolare p-value esatti in presenza di ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  reaction$Reaction.Time and reaction$Age
## S = 46.983, p-value = 0.02005
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.715256
```

```
  plot(res)
```



3.2 Two Independent Samples Problem

- Two samples: n_1 observations from y_1, n_2 from y_2
  - No continuity or finite moment requirements
- Hypotheses:
  - $H_0 : F(y_1) = F(y_2)$
  - $H_1 : F(y_1) \neq F(y_2)$ (or directional)
- Test statistics:
  - Standardized mean difference (t-statistic)
- Estimated slope coefficient (group labels as dummy predictor)
- Other permutationally equivalent statistics

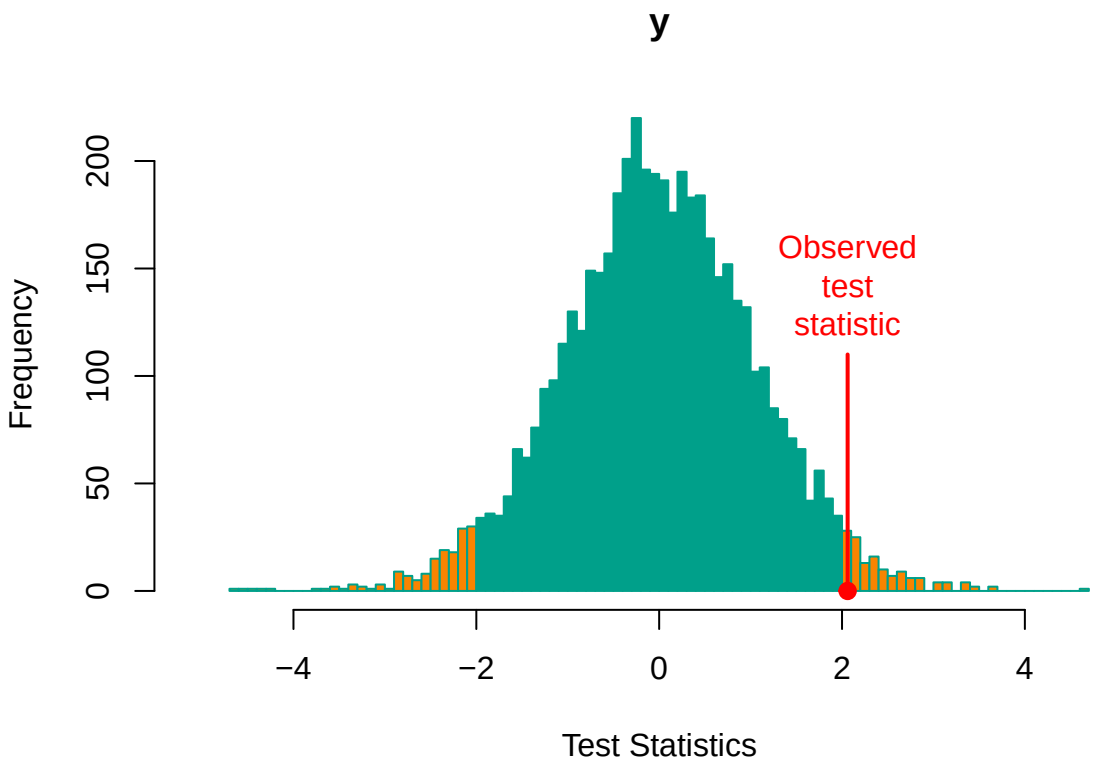
```
  data("seeds")
  seeds <- na.omit(seeds)
  (res <- flip(y ~ grp, data = seeds, perms = 5000))
```

```
##
##   Test  Stat tail p-value
## y    t 2.061   ><  0.0526
```

```
  summary(lm(y ~ grp, data = seeds))
```

```
##
## Call:
## lm(formula = y ~ grp, data = seeds)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.331 -2.931 -1.651  4.663  7.863
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   10.147      1.242   8.168    9e-09 ***
## grp            3.345      1.623   2.061    0.049 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.303 on 27 degrees of freedom
## Multiple R-squared:  0.136,  Adjusted R-squared:  0.104
## F-statistic: 4.249 on 1 and 27 DF,  p-value: 0.04903
```

```
  plot(res)
```



3.2.1 Rank Test

Can use rank-based statistics? Yes—equivalent to rank tests but with exact distribution (no tie limitations).

```
  (res <- flip(y ~ grp, data = seeds, statTest = "rank", perms = 5000))
```

```
##
##       Test  Stat tail p-value
## y Wilcoxon 2.148   ><  0.0292
```

```
  wilcox.test(y ~ grp, data = seeds)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): non è
## possibile calcolare p-value esatto in presenza di ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  y by grp
## W = 53.5, p-value = 0.03353
## alternative hypothesis: true location shift is not equal to 0
```

3.3 Chi-square and Other Categorical Methods

```r
data("seeds")
seeds$Germinated <- !is.na(seeds$x)
seeds$Germinated <- factor(seeds$Germinated)
seeds$grp <- factor(seeds$grp)
table(seeds$grp, seeds$Germinated)
```

```
##
##     FALSE TRUE
##   0     8   12
##   1     3   17
```

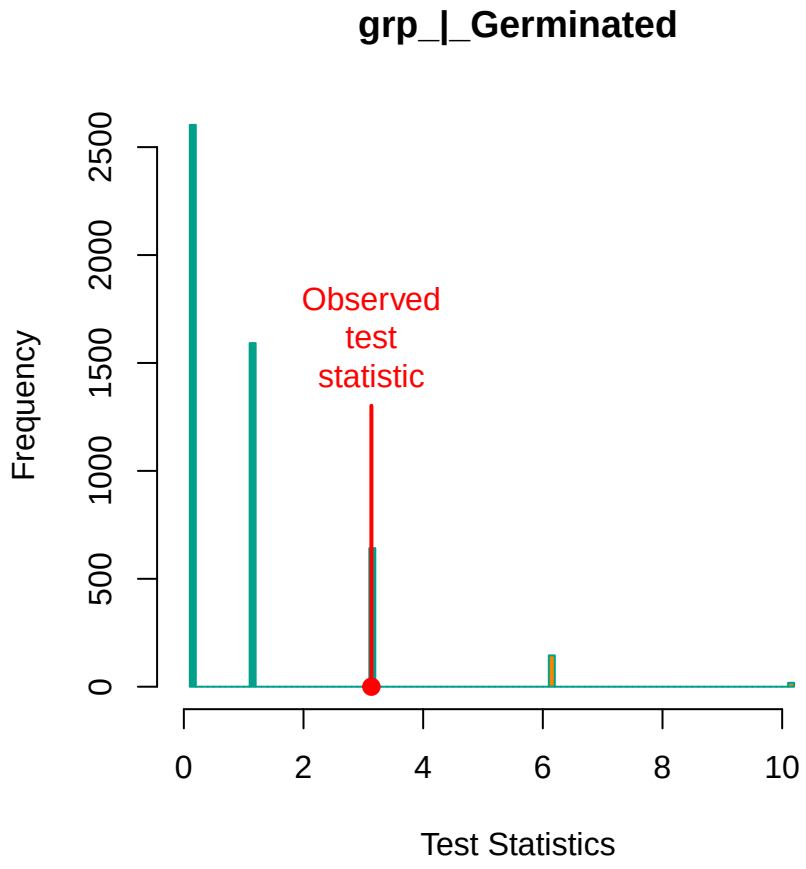
```r
chisq.test(seeds$grp, seeds$Germinated)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  seeds$grp and seeds$Germinated
## X-squared = 2.0063, df = 1, p-value = 0.1567
```

```r
(res <- flip(Germinated ~ grp, data = seeds, statTest = "Chisq", perms = 5000))
```

```
##
##                          Test  Stat tail p-value
## grp_|_Germinated Chi Squared 3.135    >  0.1610
```

```r
plot(res)
```



. . . and Fisher's exact test:

```
fisher.test(seeds$grp, seeds$Germinated)$p.value
```

```
## [1] 0.1551874
```

```
flip(Germinated ~ grp, data = seeds, perms = 5000)
```

```
##
##                 Test   Stat tail p-value
## GerminatedFALSE    t -1.798   ><  0.1596
## GerminatedTRUE     t  1.798   ><  0.1596
```

3.4 ANOVA (C-sample)

Example: 3 `Age` groups: young $[18-35)$, middle $[35-60)$, old $[60-100)$

- C samples: n_i observations from y_i $(i = 1, \ldots, C)$
  - No continuity or finite moment requirements
- Hypotheses:
  - $H_0 : F(y_i) = F(y_j)\ \forall (i, j)$

  - $H_1 : \exists (i,j) : F(y_i) \neq F(y_j)$
- Test statistics:
  - F-statistic
  - R^2
  - Other permutationally equivalent statistics
  - Rank-based alternatives

```
reaction$AgeCateg <- cut(reaction$Age, c(18, 35, 65, 100), right = FALSE)
(res <- flip(Reaction.Time ~ AgeCateg, data = reaction, perms = 5000, statTest = "ANOVA"))
```

```
##
##               Test Stat tail p-value
## Reaction.Time    F 4.02    >  0.0780
```

```
summary(lm(Reaction.Time ~ AgeCateg, data = reaction))
```

```
##
## Call:
## lm(formula = Reaction.Time ~ AgeCateg, data = reaction)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.495 -3.279  0.465  2.246  6.112
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        16.157      2.331   6.932 0.000225 ***
## AgeCateg[35,65)     4.428      3.296   1.343 0.221144
## AgeCateg[65,100)   11.418      4.037   2.828 0.025478 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.662 on 7 degrees of freedom
## Multiple R-squared:  0.5346, Adjusted R-squared:  0.4016
## F-statistic:  4.02 on 2 and 7 DF,  p-value: 0.06878
```

3.4.1 Stochastic Ordering

- Same assumptions as ANOVA
- Hypotheses:
  - Same $H_0 : F(y_i) = F(y_j)\ \forall (i,j)$
  - But $H_1 : \exists (i,j) : F(y_i) < F(y_j)$ (or $>$)

(More details on NPC later)

```
(res <- flip(Reaction.Time ~ AgeCateg, data = reaction, perms = 5000, tail = 1))
```

```
##
##                                    Test   Stat tail p-value
## Reaction.Time_|_AgeCateg.[35,65).     t 0.1423    >  0.4336
## Reaction.Time_|_AgeCateg.[65,100).    t 2.2444    >  0.0220
```

```r
npc(res)
```

```
##
##    comb.funct nVar  Stat p-value
## V1     Fisher    2 4.652  0.0202
```

3.5 Stratified Permutations (Discrete Nuisances)

Test $X =$ `Age` with $Z =$ `Gender` as nuisance in `reaction` data.

Under H_0: $f(y|x,z) = f(y|x',z) = f(y|z)\ \forall(x,x')$

Thus, under H_0, $f(y_i) = f(y_j)$ only if $z_i = z_j$ (same gender).

Can we permute as before? NO. Permute only within strata defined by Z.

Remark: - No linear nuisance effect assumption - Allow heteroscedastic errors across strata

Test statistic remains unchanged.

```r
(res <- flip(Reaction.Time ~ Age, Strata = ~Gender, data = reaction, perms = 5000))
```

```
##
##               Test  Stat tail p-value
## Reaction.Time    t 2.633   ><  0.0718
```

Alternative model (more on NPC later):

```r
(res <- flip(Reaction.Time ~ Age*Gender, Strata = ~Gender, data = reaction, perms = 5000))
```

```
##
##                                Test    Stat tail p-value
## Reaction.Time_|_Age               t  2.4826   ><  0.0750
## Reaction.Time_|_Age:Gender.M.     t -0.6518   ><  0.3394
```

```r
npc(res)
```

```
##
##    comb.funct nVar  Stat p-value
## V1     Fisher    2 3.671  0.1406
```

4 Paired samples

Understanding the Model: $Y = \gamma_0 + \gamma_1 Z + \beta X + \varepsilon$

Consider the model: - Z represents Subject ID ($i = 1, \ldots, n$) - X indicates pre/post-treatment status (typically coded as 0 for pre-treatment, 1 for post-treatment) - Each subject in Z has one pre-treatment and one post-treatment observation

The null hypothesis to test is $H_0 : \beta = 0$.

When permutations are constrained within levels of Z (within each subject), this becomes equivalent to flipping the sign of the difference between post- and pre-treatment measurements.

Let D be the vector of these differences with n observations. The null hypothesis can be rewritten as $H_0 : E(D) = 0$.

4.1 Advantages of this approach

One major advantage is that we don't need to estimate the Fisher Information (i.e., the residual variance). Let's demonstrate this with a paired t-test example:

```
n <- 20
y <- rnorm(n)

# Generate sign flips for permutation testing
FLIPS <- flipscores:::.make_flips(n, 1000)
head(FLIPS)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14]
## [1,]    1    1    1    1    1    1    1    1    1     1     1     1     1     1
## [2,]   -1    1   -1   -1    1    1   -1    1   -1     1     1     1    -1     1
## [3,]   -1   -1    1    1    1   -1   -1    1   -1     1    -1    -1     1     1
## [4,]    1   -1   -1    1    1    1    1   -1   -1     1     1    -1    -1     1
## [5,]    1   -1    1   -1   -1    1   -1   -1    1     1    -1     1    -1    -1
## [6,]    1    1    1   -1    1   -1   -1   -1   -1     1     1    -1     1    -1
##      [,15] [,16] [,17] [,18] [,19] [,20]
## [1,]     1     1     1     1     1     1
## [2,]    -1    -1     1    -1    -1    -1
## [3,]    -1     1     1     1    -1     1
## [4,]    -1     1     1     1     1     1
## [5,]     1     1    -1     1     1     1
## [6,]    -1     1     1    -1    -1    -1
```

```
# Calculate test statistics for each permutation
Tstat <- FLIPS %*% y
flipscores:::.t2p(Tstat)
```

```
## [1] 0.124
```

```
# Monte Carlo simulation to check Type I error rate
MCMC <- 10000
Y <- matrix(rnorm(n * MCMC), n, MCMC)
Tstat <- FLIPS %*% Y
p.values <- apply(Tstat, 2, flipscores:::.t2p)
mean(p.values < 0.05)
```

```
## [1] 0.047
```

```
# Lightweight t-test function
t.test.light <- function(Y, tail = 1) {
  sd <- apply(Y, 2, sd, na.rm = TRUE)
  n <- nrow(Y)
  ts <- colMeans(Y, na.rm = TRUE) / sd * sqrt(n)
  pt(-ts, df = n - 1)
}

# Test with heteroscedastic data
Y <- replicate(MCMC, rnorm(n, sd = exp(1:n)))
```

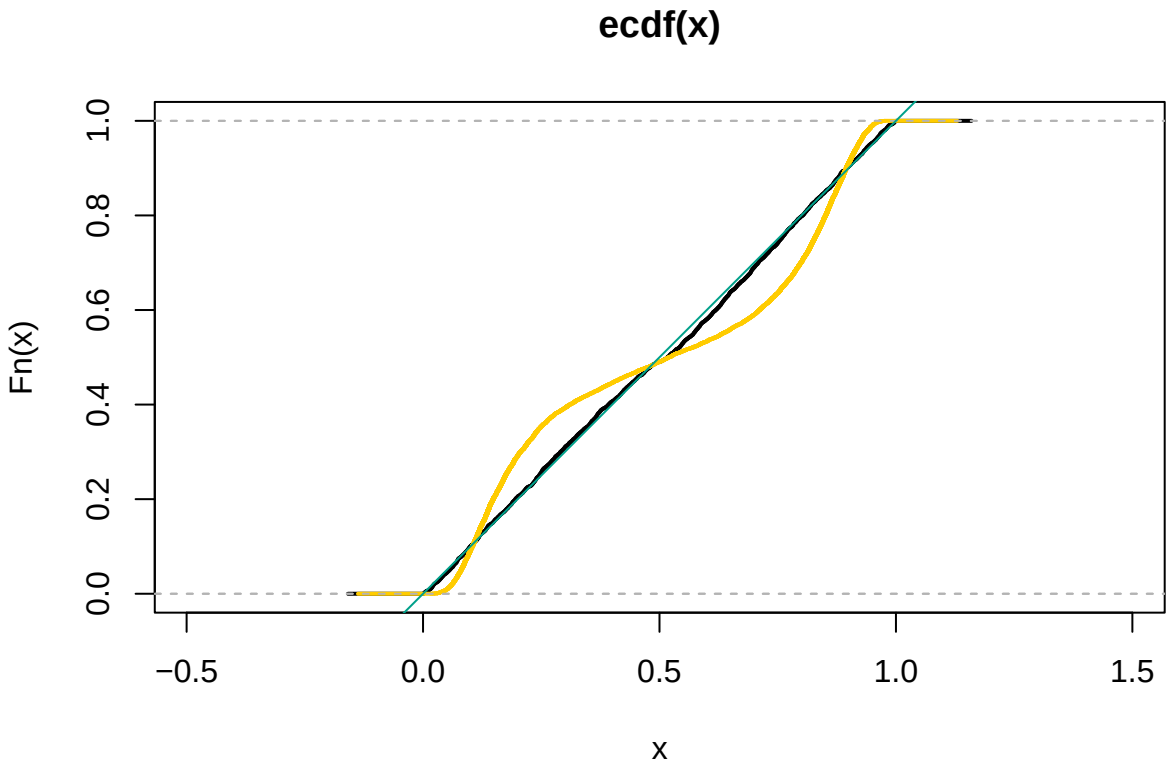
```r
Tstat <- FLIPS %*% Y
p.values <- apply(Tstat, 2, flipscores:::.t2p)
p.values_param <- t.test.light(Y)

# Compare Type I error rates
mean(p.values < 0.05)
```

```
## [1] 0.0438
```

```r
mean(p.values_param < 0.05)
```

```
## [1] 0.0095
```

```r
# Visualize p-value distributions
plot.ecdf(p.values,lwd=2,asp=1)
plot.ecdf(p.values_param,lwd=2, add = TRUE, col = 3)
abline(0, 1, col = 2)
```



In this example, we compare a permutation-based approach (sign-flipping test) with a parametric t-test:

1. **Permutation Test**: We generate sign flips that correspond to within-subject permutations. The test statistic is calculated for each sign-flipped version of the data, creating a null distribution against which we compare our observed statistic.
2. **Parametric t-test**: We use a standard t-test that assumes normality and constant variance.

3. **Heteroscedasticity Simulation**: The second part of the code generates data with increasing variance across observations (`sd = exp(1:n)`). This violates the constant variance assumption of the parametric t-test.

The empirical cumulative distribution function (ECDF) plot shows how the p-values from both methods compare. Under the null hypothesis with correct assumptions, both should produce uniformly distributed p-values (follow the diagonal line). When assumptions are violated, the parametric test may produce inflated Type I error rates, while the permutation test maintains correct error control due to its distribution-free nature.

Key Insights: - The permutation test doesn't require estimating residual variance - It remains valid even when parametric assumptions (like homoscedasticity) are violated - The method is particularly powerful for paired data where within-subject comparisons are natural - Sign-flipping is equivalent to permuting within subjects when testing for treatment effects in paired designs

4.2 Repeated measures and mixed models

More properly this approach is known as Random coefficient Analysis / Group-level analysis

- Basso & Finos (2012). Exact Multivariate Permutation Tests for Fixed Effects in Mixed-Models. *Communications in Statistics - Theory and Methods*, 41: 2991 - 3001.
- Finos & Basso (2014) Permutation tests for between-unit fixed effects in multivariate generalized linear mixed models. *Stat Comput* 24, 941–952.

A more complete approach:

- Andreella, Goeman, Hemerik, Finos (2025) Robust Inference for Generalized Linear Mixed Models: A "Two-Stage Summary Statistics" Approach Based on Score Sign Flipping. Psychometrika, 1-23.

5 Multivariate Testing

5.1 Seeds Data

```
# install.packages("flip")
library(flip)
```

Remove `NA`s:

```
data(seeds, package = "flip")
seeds <- na.omit(seeds)
seeds
```

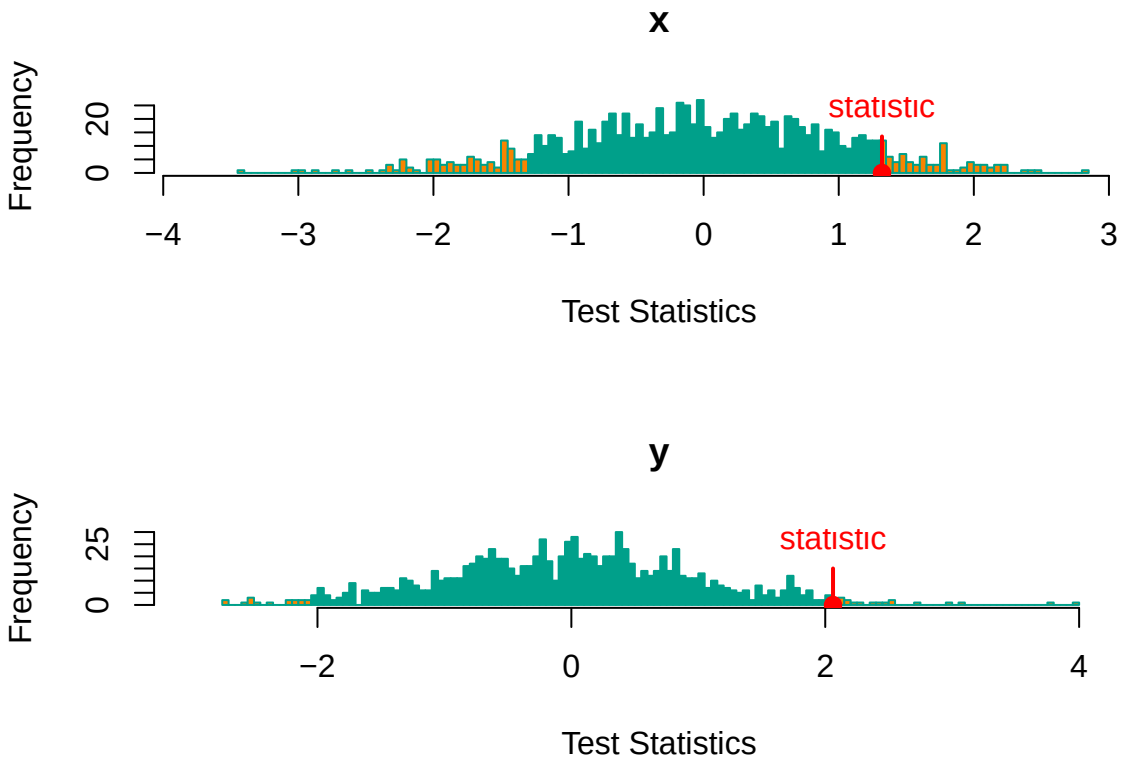
```
##    grp    x     y
## 9    0 6.03 12.54
## 10   0 4.20 14.81
## 11   0 4.49 16.71
## 12   0 2.00  7.53
## 13   0 2.84  7.02
```

```
## 14   0 3.88  8.09
## 15   0 2.04  5.76
## 16   0 5.48 18.01
## 17   0 2.31  8.81
## 18   0 1.90  8.17
## 19   0 1.75  6.62
## 20   0 3.02  7.69
## 24   1 3.31 18.49
## 25   1 6.56 19.20
## 26   1 3.16  9.85
## 27   1 4.07 15.83
## 28   1 2.09  6.16
## 29   1 6.72 17.58
## 30   1 3.93 19.29
## 31   1 2.56 10.77
## 32   1 8.30 18.31
## 33   1 4.21 10.56
## 34   1 1.86  9.48
## 35   1 3.09 12.54
## 36   1 5.09 18.35
## 37   1 4.08 11.84
## 38   1 3.63 11.44
## 39   1 2.61  7.66
## 40   1 5.21 12.00
```

5.2 Marginal vs Joint Distribution

Use permutation methods to test for group differences (`grp`) on both `x` and `y`:

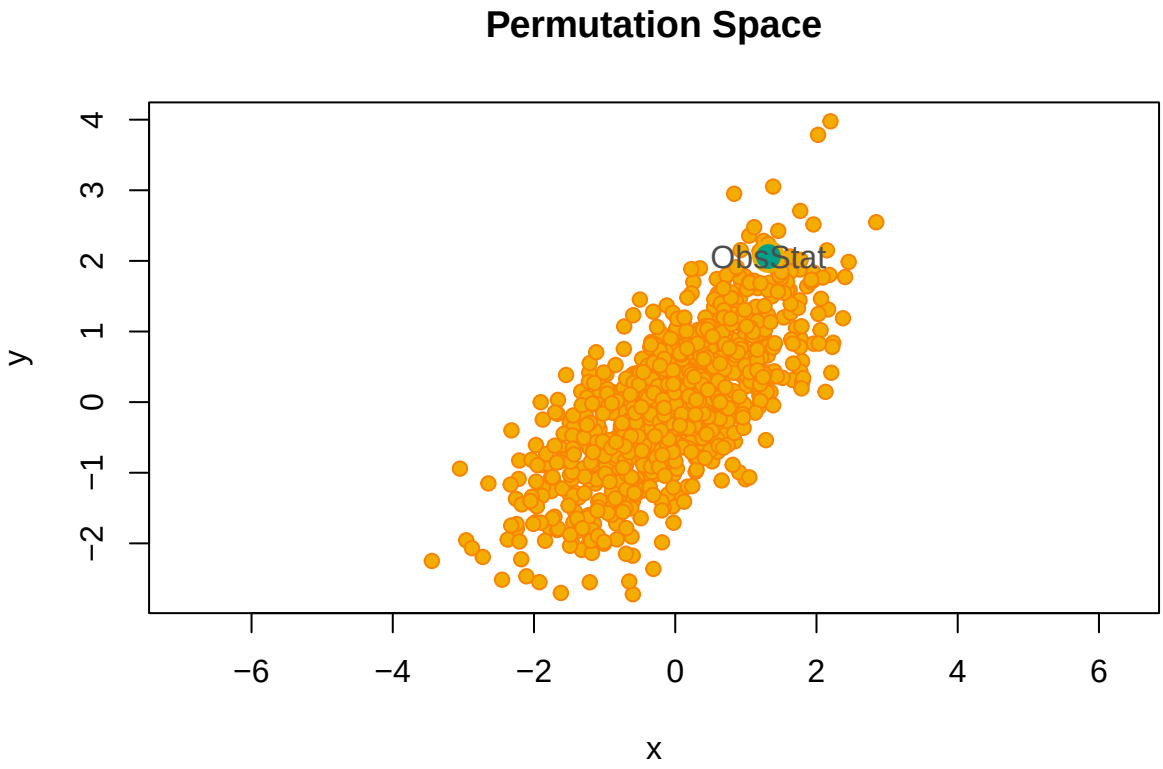
```
library(flip)
res <- flip(. ~ grp, data = seeds, flipReturn = list(permP = TRUE, permT = TRUE))
hist(res)
```




```
# flipReturn = list(permP = TRUE, permT = TRUE) not strictly needed; useful later
```

We can test the two variables separately but lack an overall p-value (is there ANY difference?).

```
plot(res)
```



Next we'll see: - How to combine p-values (e.g., Fisher's combining function) for global hypothesis testing - How to use closed testing procedures to adjust p-values: which variables differ?

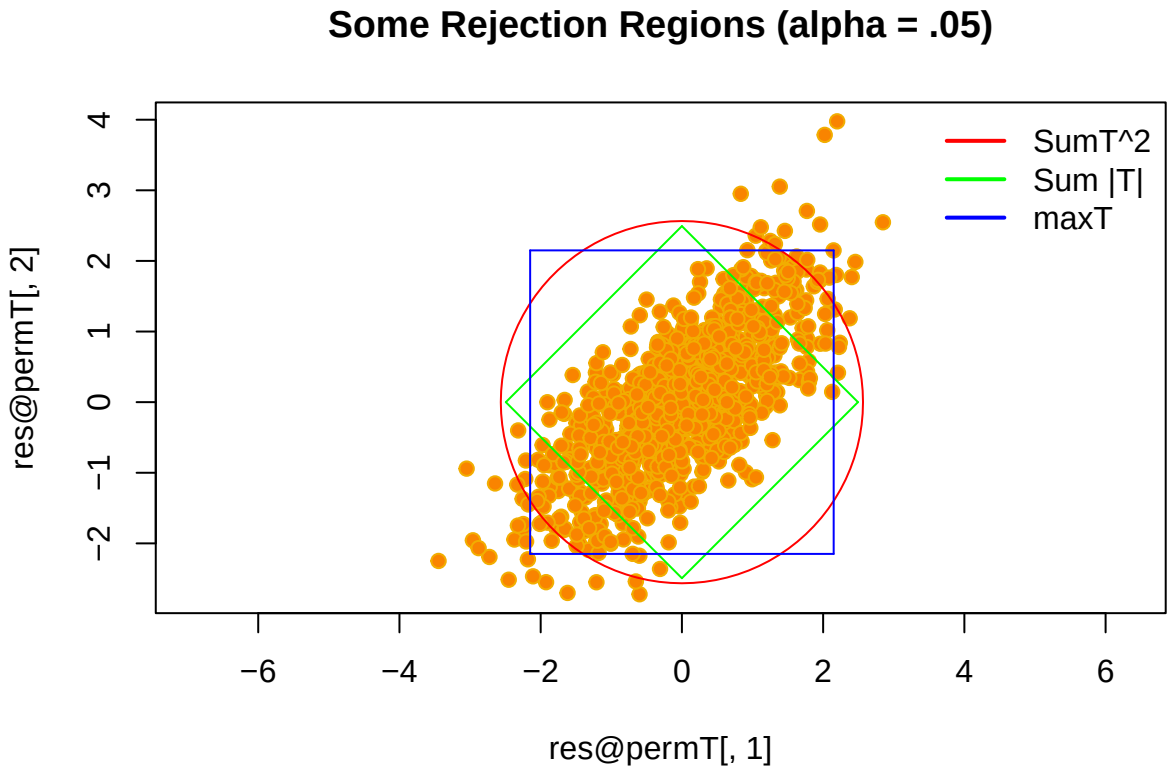
5.3 Rejection Regions (and Overall Testing)

In univariate settings, defining 'far from null' (usually from test statistic = 0) is straightforward. In multivariate settings, there are multiple (no uniformly best) approaches.

```
# install.packages("plotrix")
library("plotrix")
```

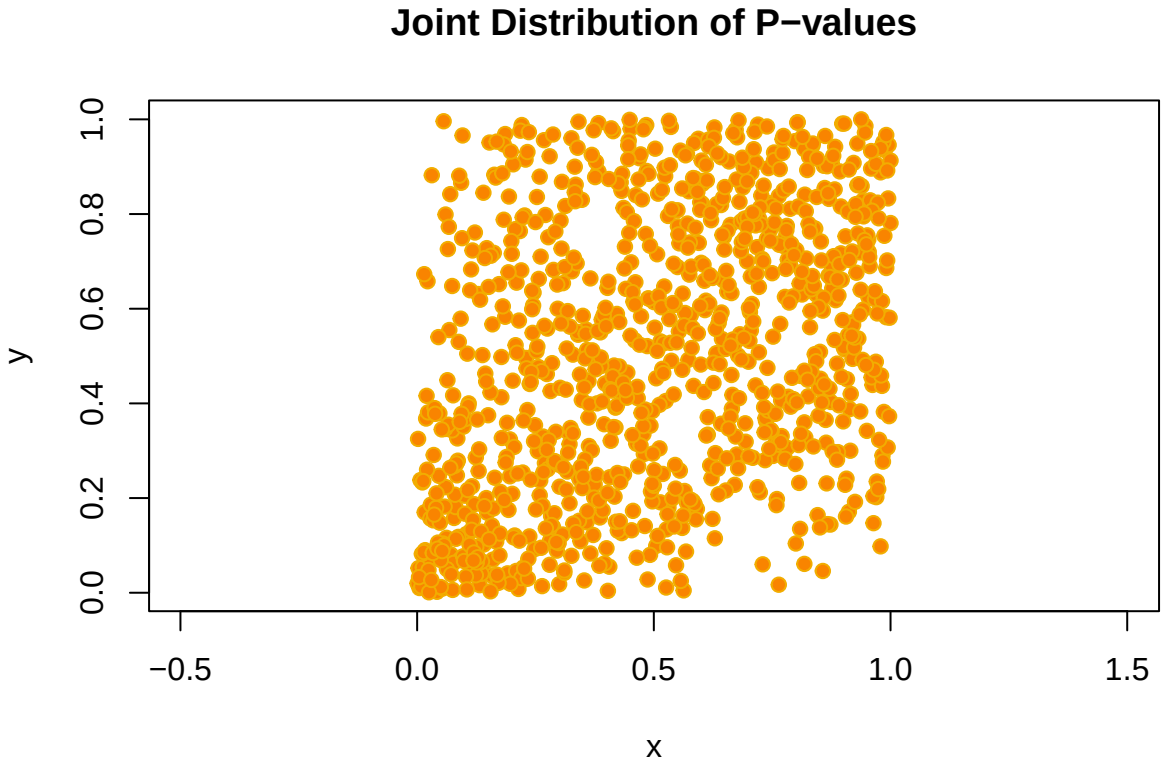
```
## Warning: il pacchetto 'plotrix' è stato creato con R versione 4.5.2
```

```
res.sumt2 <- npc(res, "sumT2", flipReturn = list(permP = TRUE, permT = TRUE))
limsumt2 <- res.sumt2@permT[which.min(abs(res.sumt2@permP - 0.05))]
res.sumt <- npc(res, "sumT", flipReturn = list(permP = TRUE, permT = TRUE))
limsumt <- res.sumt@permT[which.min(abs(res.sumt@permP - 0.05))]
res.maxt <- npc(res, "maxT", flipReturn = list(permP = TRUE, permT = TRUE))
limmaxt <- res.maxt@permT[which.min(abs(res.maxt@permP - 0.05))]

plot(res@permT[,1], res@permT[,2], col = "#F2AD00", bg = "#F98400", pch = 21,
     main = "Some Rejection Regions (alpha = .05)", asp = 1)
draw.circle(0, 0, limsumt2^.5, border = "red")
segments(c(limsumt, -limsumt, limsumt, -limsumt), c(0, 0, 0, 0),
         c(0, 0, 0, 0), c(limsumt, -limsumt, -limsumt, limsumt), col = "green")
segments(c(limmaxt, -limmaxt, -limmaxt, limmaxt), c(limmaxt, limmaxt, -limmaxt, -limmaxt),
         c(-limmaxt, -limmaxt, limmaxt, limmaxt), c(limmaxt, -limmaxt, -limmaxt, limmaxt), col = "blue")
legend("topright", legend = c("SumT^2", "Sum |T|", "maxT"),
       col = c("red", "green", "blue"), bty = "n", lwd = 2)
```



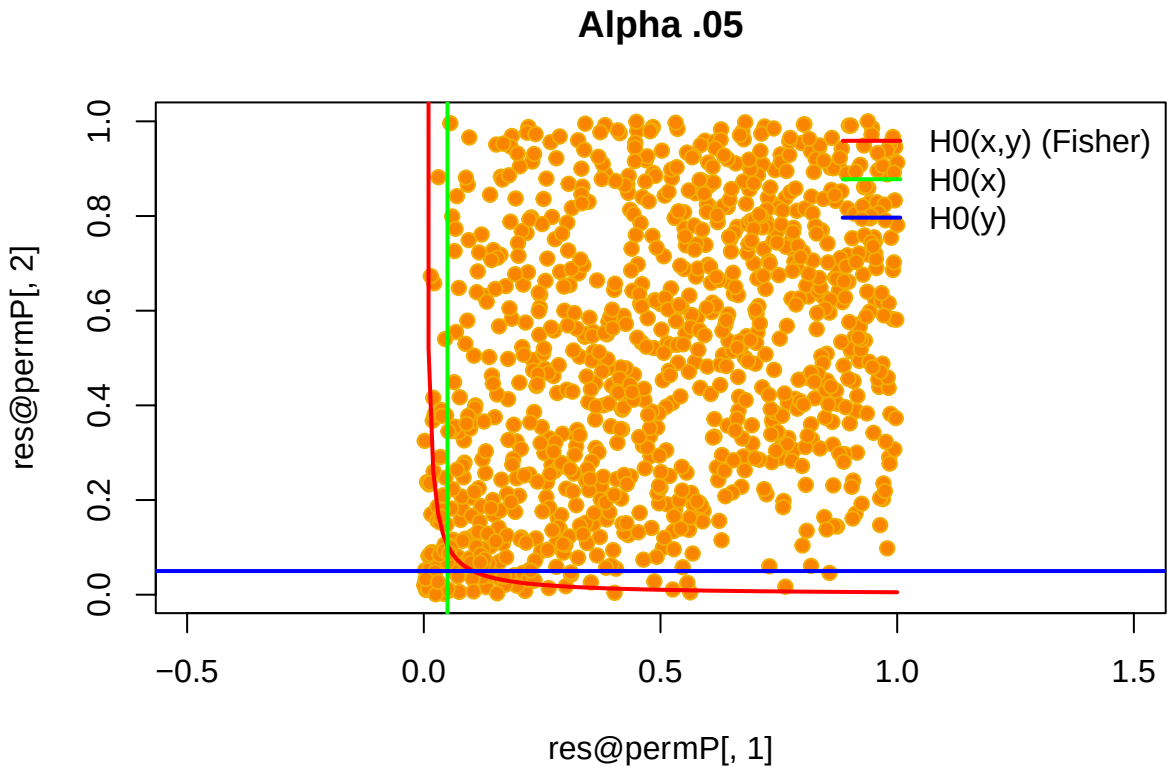
REMARK: We can derive the p-value distribution by computing p-values for each test statistic (observed and permuted data). This yields the multivariate p-value distribution:

```
plot(res@permP, col = "#F2AD00", bg = "#F98400", pch = 21,
     main = "Joint Distribution of P-values", asp = 1)
```



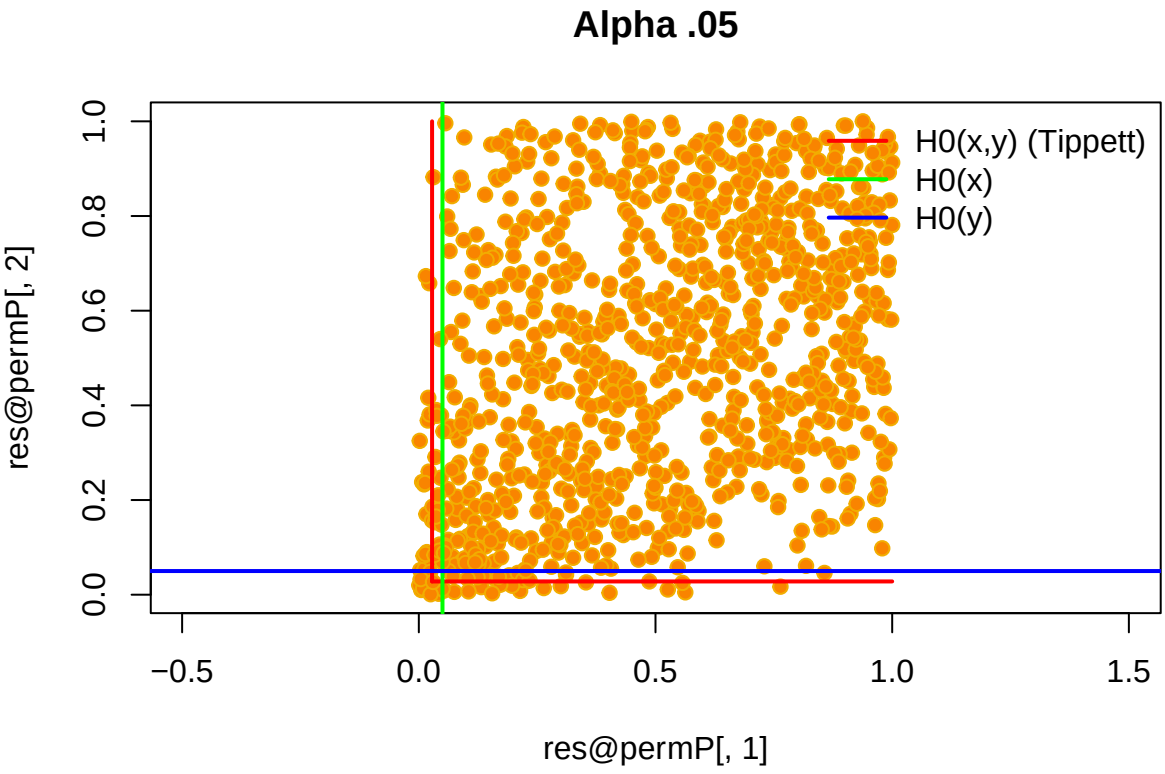
5.3.1 Fisher Combining Function

Examine rejection regions for univariate tests and Fisher combination. Intersection of each univariate test with Fisher region defines closed testing rejection region (adjusted for multiple testing).



5.3.2 Tippett (min-p) Combining Function

Examine rejection regions for univariate tests and Tippett combination. Intersection defines closed testing rejection region. This coincides with Westfall & Young shortcut.



6 FWER Control via Permutation Tests

6.1 Permutation Bonferroni

Bonferroni is conservative:

- **Bonferroni bound**: Reject if p-value $\leq \alpha/m$
- **By Boole's inequality**: Guaranteed FWER $\leq \alpha$, but often FWER $< \alpha$
- **Can we improve?**: Reject if p-value $\leq \tilde{\alpha} > \alpha/m$ while maintaining FWER control
- **Yes**: Via permutations

6.2 Improved Bonferroni

- **Reduced** α: Reject H_i if $p_i \leq \tilde{\alpha}$
- **FWER control?**:

$$\begin{aligned}\text{FWER} &= P(p_i \leq \tilde{\alpha} \text{ for at least one true } H_i) \\ &= P\left(\bigcup_{i \in T} \{p_i \leq \tilde{\alpha}\}\right) \\ &= P\left(\min_{i \in T} p_i \leq \tilde{\alpha}\right) \leq \alpha\end{aligned}$$

- **How to determine $\tilde{\alpha}$?**: Use permutations to find minimum p-value distribution

6.3 Multiple Testing via Permutations

Single-step min-P method:

1. Calculate smallest p-value m for real data
2. Randomly permute data
3. Calculate new p-values for all tests on permuted data
4. Calculate smallest p-value m^π for permuted data
5. Repeat permutations many times (e.g., k = 1000): $m_1^\pi, \ldots, m_k^\pi$
6. Calculate $\tilde{\alpha}$ as α-quantile of $m_1^\pi, \ldots, m_k^\pi$

Multiple testing result: Reject all hypotheses with (non-permuted) p-values $\leq \tilde{\alpha}$

6.4 P-value Correlation Structure

Permutation:

- Destroys covariate-response correlation
- Preserves covariate correlations

Consequence:

- P-values of correlated tests remain correlated in permutations
- Minimum p-value distribution correctly accounts for correlations

When is improvement over Bonferroni large?:

- Negatively correlated p-values: typically no gain
- Independent p-values: minimal gain
- Positively correlated p-values: potentially large gain

6.5 Improved Holm: Westfall & Young

Westfall PH, Young SS (1993) Resampling-Based Multiple Testing: Examples and Methods for p-Value Adjustment. Wiley **Sequential permutation multiple testing**:

- **Single-step**: Permutation equivalent of Bonferroni

- **Holm equivalent**: Westfall & Young method
  min-P algorithm:

1. Start with all hypotheses 2. Repeat:

- Perform single-step min-P to calculate $\tilde{\alpha}$
- Reject hypotheses with p-value $\leq \tilde{\alpha}$
- Remove rejected hypotheses

3. Until no new rejections

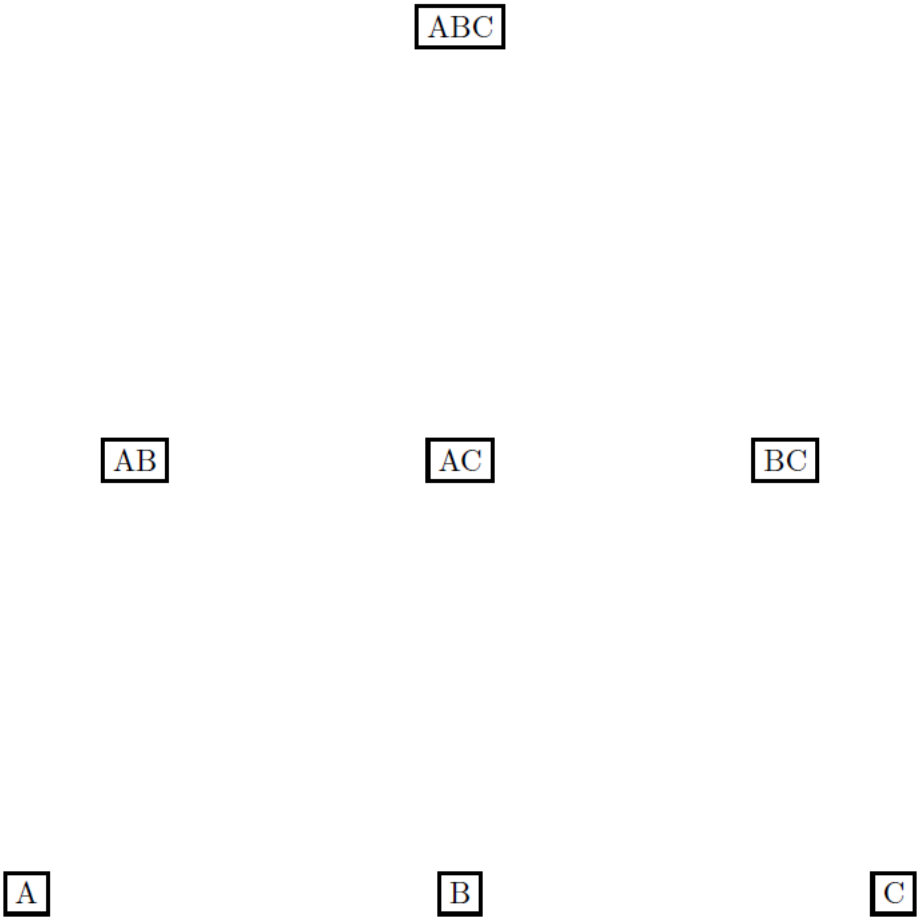
6.6 General Framework: Closed Testing

Marcus R, Peritz E, Gabriel KR (1976). On closed testing procedures with special reference to ordered analysis of variance. Biometrika 63: 655-660.

Test each node with any multivariate permutation test.

Westfall & Young is a special case of closed testing (each node uses min-p/Tippett or max-T combining function).

6.6.1 Closure Set



Adjusted $\tilde{p}_A = \max(p_A, p_{AB}, p_{AC}, p_{ABC})$

In our data:

```
(res <- flip.adjust(res, method = "Fisher"))
```

```
##
##   Test  Stat tail p-value Adjust:Fisher
## x    t 1.320   ><  0.1710        0.1710
## y    t 2.061   ><  0.0350        0.0620
```

```
(res <- flip.adjust(res, method = "maxT"))
```

```
##
##   Test  Stat tail p-value Adjust:Fisher Adjust:maxT
## x    t 1.320   ><  0.1710        0.1710      0.1710
## y    t 2.061   ><  0.0350        0.0620      0.0600
```

Conclusion

Accounting for dependencies: Adjusted p-values become lower (more rejections).

When?:

- Negative correlation: generally no gain
- Independent p-values: little or no gain
- Positive correlation: substantial gain (note: two-sided tests with negatively correlated test statistics yield positively correlated p-values)

Real data: Often correlated variables → permutations advantageous

How?: `R: library(flip); flip(); flip.adjust()`

7 Case Study: Pharmacokinetic Study of Carbidopa

Description:
http://webserv.jcu.edu/math//faculty/TShort/Bradstreet/part2/part2-table6.html

12 healthy male subjects in three-period crossover design receiving three graded doses (25, 50, 100 mg) of Carbidopa q8h. Seven-day washout between periods. Pharmacokinetic variables AUC, Cmax, Tmax calculated from plasma concentrations at 0, 0.5, 1, 1.5, 2, 3, 4, 5, 6, 7, 8 hours postdosing after second dose on day 6.

Dataset:
http://webserv.jcu.edu/math//faculty/TShort/Bradstreet/part2/Bradp2t6.txt

Analyze without accounting for study periods (randomized within subjects).

Research questions: 1. Is there a dose response for AUC, Cmax, or Tmax? Overall? 2. Can dose proportionality be established? (Fit linear model for each endpoint, discuss results)

7.1 Solution

Address both questions with single analysis: linear model (accounting for individual variability) on log-transformed endpoints.

```
# Read and prepare data
dati <- read.table("http://webserv.jcu.edu/math//faculty/TShort/Bradstreet/part2/Bradp2t6.txt",
                   skip = 1, header = TRUE)
dati <- cbind(dati[,1], matrix(as.matrix(dati[,-1]), nrow(dati)*3, 4))
colnames(dati) <- c("Sub", "Dose", "AUC", "Cmax", "Tmax")
dati <- as.data.frame(dati)
str(dati)
```

```
## 'data.frame':    36 obs. of  5 variables:
##  $ Sub : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ Dose: num  100 25 50 50 50 25 100 25 50 25 ...
##  $ AUC : num  604 140 386 175 605 ...
##  $ Cmax: num  137 44.4 86.6 46.4 194 44.9 318 29 119 58.4 ...
##  $ Tmax: num  1.5 1 1.5 1.5 0.5 1 1 1 2 2 ...
```

```
# Log-transform responses (linear relationship indicates proportionality)
dati[,3:5] <- log(dati[,3:5])
```

```
# Descriptives and plots
summary(dati[,-1])
```

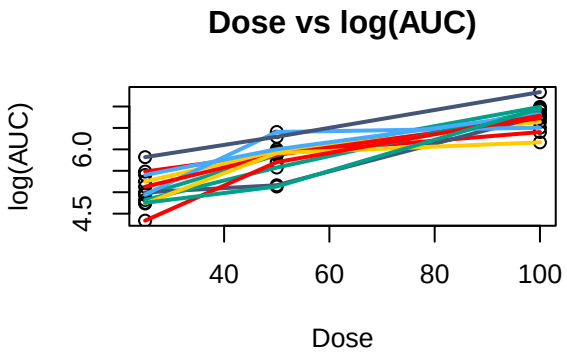
```
##       Dose             AUC             Cmax            Tmax        
##  Min.   : 25.00   Min.   :4.337   Min.   :3.219   Min.   :-0.6931  
##  1st Qu.: 25.00   1st Qu.:5.156   1st Qu.:3.966   1st Qu.: 0.0000  
##  Median : 50.00   Median :5.886   Median :4.485   Median : 0.2027  
##  Mean   : 58.33   Mean   :5.873   Mean   :4.547   Mean   : 0.2474  
##  3rd Qu.:100.00   3rd Qu.:6.539   3rd Qu.:5.280   3rd Qu.: 0.6931  
##  Max.   :100.00   Max.   :7.335   Max.   :5.989   Max.   : 1.0986
```

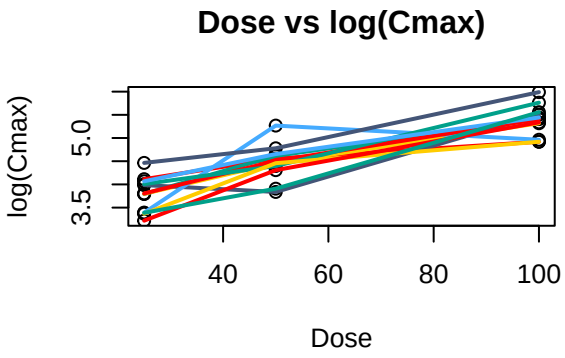
```
by(dati[,3:5], dati$Dose, summary)
```

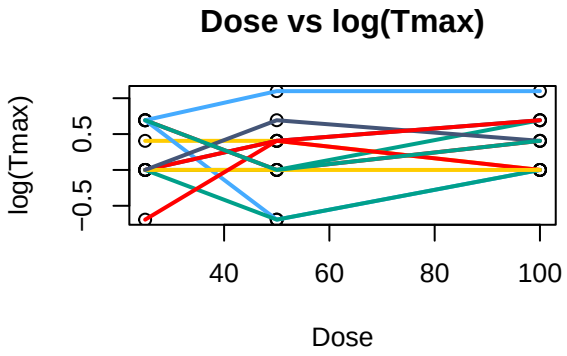
```
## dati$Dose: 25
##       AUC             Cmax            Tmax        
##  Min.   :4.337   Min.   :3.219   Min.   :-0.6931  
##  1st Qu.:4.803   1st Qu.:3.390   1st Qu.: 0.0000  
##  Median :4.972   Median :3.801   Median : 0.0000  
##  Mean   :5.051   Mean   :3.783   Mean   : 0.2071  
##  3rd Qu.:5.289   3rd Qu.:4.022   3rd Qu.: 0.6931  
##  Max.   :5.818   Max.   :4.464   Max.   : 0.6931  
## ------------------------------------------------------------
## dati$Dose: 50
##       AUC             Cmax            Tmax        
##  Min.   :5.133   Min.   :3.837   Min.   :-0.6931  
##  1st Qu.:5.670   1st Qu.:4.374   1st Qu.: 0.0000  
##  Median :5.886   Median :4.484   Median : 0.2027  
##  Mean   :5.815   Mean   :4.479   Mean   : 0.1689  
##  3rd Qu.:5.967   3rd Qu.:4.625   3rd Qu.: 0.4055
```

```
##  Max.   :6.405   Max.   :5.268   Max.   : 1.0986
## ------------------------------------------------------------
## dati$Dose: 100
##       AUC             Cmax            Tmax
##  Min.   :6.164   Min.   :4.920   Min.   :0.0000
##  1st Qu.:6.607   1st Qu.:5.229   1st Qu.:0.0000
##  Median :6.782   Median :5.412   Median :0.4055
##  Mean   :6.751   Mean   :5.378   Mean   :0.3662
##  3rd Qu.:6.922   3rd Qu.:5.515   3rd Qu.:0.6931
##  Max.   :7.335   Max.   :5.989   Max.   :1.0986
```

```r
par(mfrow = c(2,2))
plot(dati$Dose, dati$AUC, ylab = "log(AUC)", xlab = "Dose", main = "Dose vs log(AUC)")
temp=sapply(unique(dati$Sub), function(s) {
d <- subset(dati, Sub == s)
d <- d[order(d$Dose),]
lines(d$Dose, d$AUC, col = s, lwd = 2)
})

plot(dati$Dose, dati$Cmax, ylab = "log(Cmax)", xlab = "Dose", main = "Dose vs log(Cmax)")
temp=sapply(unique(dati$Sub), function(s) {
d <- subset(dati, Sub == s)
d <- d[order(d$Dose),]
lines(d$Dose, d$Cmax, col = s, lwd = 2)
})

plot(dati$Dose, dati$Tmax, ylab = "log(Tmax)", xlab = "Dose", main = "Dose vs log(Tmax)")
temp=sapply(unique(dati$Sub), function(s) {
d <- subset(dati, Sub == s)
d <- d[order(d$Dose),]
lines(d$Dose, d$Tmax, col = s, lwd = 2)
})
```





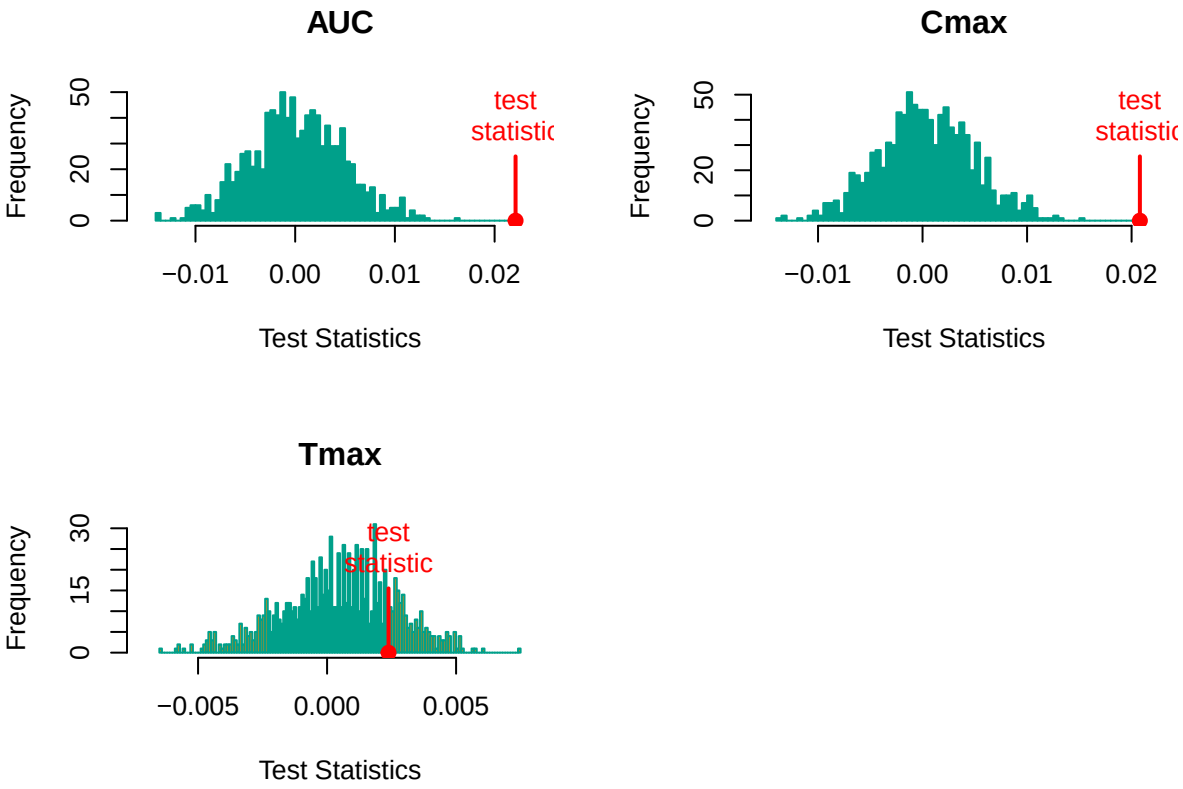


Simple solution:

```
library(flip)
res <- flip(. ~ Dose, data = dati, Strata = ~Sub, statTest = "coeff")
summary(res)
```

```
##  Call:
##  flip(Y = . ~ Dose, data = dati, statTest = "coeff", Strata = ~Sub)
## 999 permutations.
##
##       Test   Stat tail p-value sig.
## AUC  coeff 0.0221   ><  0.0010  ***
## Cmax coeff 0.0208   ><  0.0010  ***
## Tmax coeff 0.0024   ><  0.2950
```

```
# statTest = "coeff": estimated linear model coefficient
hist(res)
```



Multivariate: - Overall

```
res <- flip.adjust(res)
npc(res, "Fisher")
```

```
##
##    comb.funct nVar  Stat p-value
## V1     Fisher    3 15.04  0.0010
```

Dose effect exists overall.

- By endpoint (closed testing with max-t). Try different methods (e.g., `method = "Fisher"`) and compare `method = "minP"` with `method = "Holm"`.

```
res <- flip.adjust(res, method = "holm")
res <- flip.adjust(res, method = "Fisher")
summary(res)
```

```
##  Call:
##  flip(Y = . ~ Dose, data = dati, statTest = "coeff", Strata = ~Sub)
## 999 permutations.
##
##       Test   Stat tail p-value Adjust:maxT Adjust:holm Adjust:Fisher sig.
## AUC  coeff 0.0221   ><  0.0010      0.0010      0.0030        0.0010  ***
## Cmax coeff 0.0208   ><  0.0010      0.0010      0.0030        0.0010  ***
## Tmax coeff 0.0024   ><  0.2950      0.2950      0.2950        0.2950
```

`AUC` and `Cmax` show significant effects after multiplicity correction; `Tmax` does not.

8 Minimal Bibliography

Grounding Theory:
- Pesarin (2001) *Multivariate Permutation Tests: With Applications in Biostatistics*. Wiley, New York

Alternative permutation testing approach:
- Hemerik J, Goeman J. Exact testing with random permutations. Test. 2018;27(4):811-825. doi:10.1007/s11749-017-0571-1

Flexible GLM approach via sign-flip score test:
- Hemerik, Goeman and Finos (2020) Robust testing in generalized linear models by sign flipping score contributions. *Journal of the Royal Statistical Society Series B* 82(3). DOI: 10.1111/rssb.12369
Implemented in R package `flipscores`:
https://cran.r-project.org/web/packages/flipscores/index.html
Development version: https://github.com/livioivil/flipscores

Permutation regression review:
- Winkler AM, Ridgway GR, Webster MA, Smith SM, Nichols TE (2014) Permutation inference for the general linear model. *NeuroImage* 92:381-397. doi:10.1016/j.neuroimage.2014.01.060