Hypothesis testing framework

Summary of Wald tests

Let $\theta \in \mathbb{R}^p$ be some parameter of interest. We wish to test the hypotheses

$$H_0 : C\theta = \gamma_0 \qquad H_A : C\theta \neq \gamma_0$$

for some $C \in \mathbb{R}^{q \times p}$. Given an estimator $\hat{\theta}_n$ such that

$$V_n^{-\frac{1}{2}}(\hat{\theta}_n - \theta) \xrightarrow{d} N(0, I),$$

the **Wald test** rejects when

$$(C\hat{\theta}_n - \gamma_0)^T V_n^{-1}(C\hat{\theta}_n - \gamma_0) > \chi^2_{q,\alpha}$$

General framework for hypothesis tests

Definition: Let $\theta \in \mathbb{R}^p$ be some parameter of interest. A **hypothesis** is a statement about θ. The goal of a hypothesis test is to compare two competing hypotheses:

$$H_0 : \theta \in \Theta_0 \qquad H_A : \theta \in \Theta_1$$

Outcomes

$$H_0 : \theta \in \Theta_0 \qquad H_A : \theta \in \Theta_1$$

The outcome of the test is a decision to either **reject** H_0 or **fail to reject** H_0.

Constructing a test

$$H_0 : \theta \in \Theta_0 \qquad H_A : \theta \in \Theta_1$$

Power function

Suppose we reject H_0 when $(X_1, \ldots, X_n) \in R$. The **power function** $\beta(\theta)$ is

$$\beta(\theta) = P_\theta((X_1, \ldots, X_n) \in R)$$

Example

$X_1, \ldots, X_n$ iid from a population with mean μ and variance σ^2.

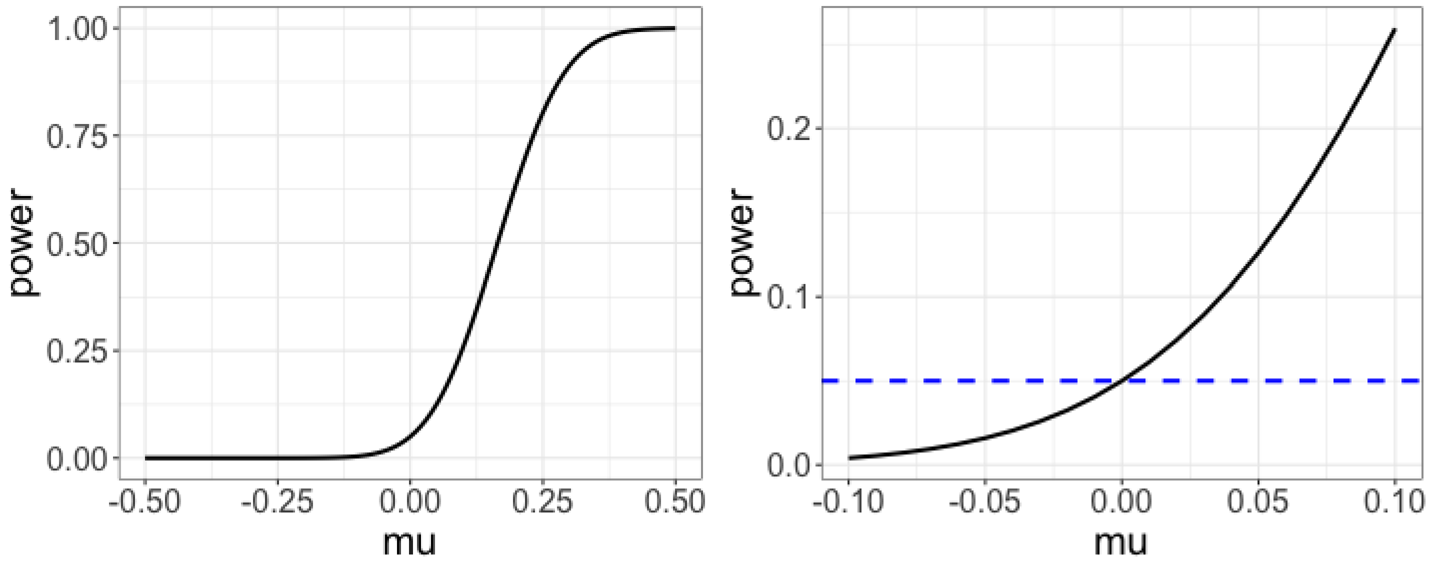
$$H_0 : \mu = \mu_0 \qquad H_A : \mu > \mu_0$$

Class activity

$$\beta(\mu) \approx 1 - \Phi\left(z_\alpha - \frac{(\mu - \mu_0)}{\sigma/\sqrt{n}}\right)$$

- Suppose that $\mu_0 = 0, n = 100$, and $\sigma = 1$. Make a plot of $\beta(\mu)$ vs. μ for $\alpha = 0.05$.
- Now consider testing $H_0 : \mu \leq \mu_0$ vs. $H_A : \mu > \mu_0$. Will this change our rejection region if we want a size α test?

Class activity



Simulation verification

```
n <- 100
sigma <- 1
mu0 <- 0
nreps <- 5000
test_stats <- rep(0, nreps)
for(i in 1:nreps){
  x <- rnorm(n, mu0, sigma)
  test_stats[i] <- (mean(x) - mu0)/(sigma/sqrt(n))
}
mean(test_stats > qnorm(0.05, lower.tail=F))
```

```
## [1] 0.0524
```

Simulation verification

```r
n <- 100
sigma <- 1
mu0 <- 0
nreps <- 5000
test_stats <- rep(0, nreps)
for(i in 1:nreps){
  x <- rnorm(n, 0.1, sigma)
  test_stats[i] <- (mean(x) - mu0)/(sigma/sqrt(n))
}
mean(test_stats > qnorm(0.05, lower.tail=F))
```

```
## [1] 0.257
```