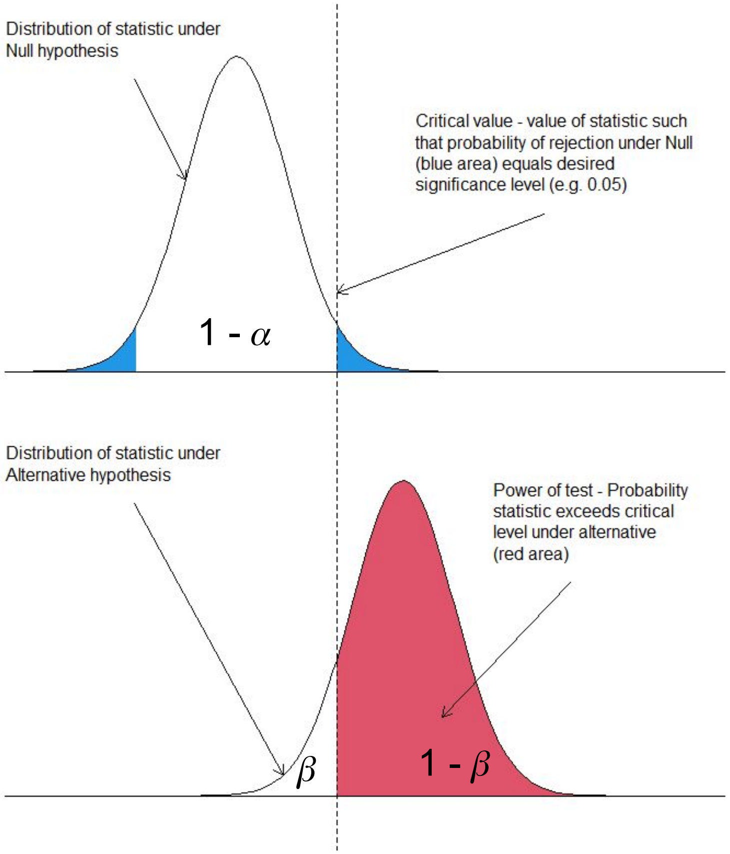
Power analysis

Today's agenda:

- Discuss power analyses
- Example power analyses

What is power?

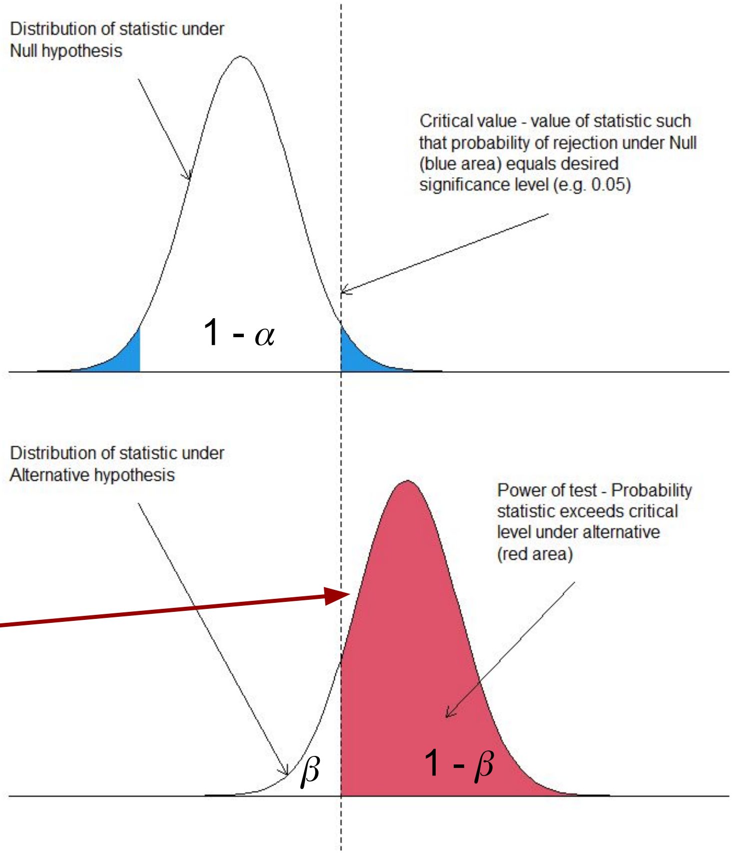
	Probability to reject H_0	Probability to NOT reject H_0
If H_0 is true	α	$1 - \alpha$
If H_1 is true	$1 - \beta$	β



What is power?

	Probability to reject H_0	Probability to NOT reject H_0
If H_0 is true	α	$1 - \alpha$
If H_1 is true	$\boxed{1 - \beta}$	β

power



Power in a broader sense

Rather than focus on p-values, better to focus on:

- How does data quantity impact my answers?
- How does data quality impact my answers?
- How do I trade off experimental costs vs effectiveness?

How do we estimate power?

- In some cases, you can use fancy math
- Most cases are easiest using **simulations!**

Things to think about with power analyses

- Number and distribution of sites
- Number of total observations
- Distribution of observations across sites
- Stratified vs even sampling
- Amount of variation
- Control vs quantification of variation
- Effect sizes

Things to think about with power analyses

Say I wanted to measure leaf area for species across an elevation gradient:

- What kind of factors might I need to consider?

Costs of research

More data is always better

- BUT data collection has costs in terms of time and money
- Power analyses help us allocate available time and money better

Data quality metrics

- Bias: difference between the estimate and the true value
- Variance: variability of estimates around mean estimate
- Confidence interval width: how precise is the estimate?
- Mean Squared Error: (bias^2 + variance) total variation around the true value
- Coverage: probability the CI includes the true value
- Power: probability of correctly rejecting the null hypothesis

Simple example: Linear Regression

We’ll simulate a linear function with normal error:

Slope = 1

Intercept = 2

SD = 8

Samples: x = 1,2,3,... 20

How do we simulate this?

Simple example: Linear Regression

```
x <- 1:20
a <- 2
b <- 1
sd <- 8
N = 20
set.seed(1)
y_det <- a + b*x
y <- rnorm(n = length(y_det),
           mean = y_det,
           sd = sd)
```

Simple example: Linear Regression

```
x <- 1:20
a <- 2
b <- 1
sd <- 8
N = 20
set.seed(1)
y_det <- a + b*x
y <- rnorm(n = length(y_det),
           mean = y_det,
           sd = sd)
```

Once you've done this, make a plot of y ~ x

Compare your plot with others, are they the same?

Simple example: Linear Regression

Now we need to:

- Check whether the relationship we simulated is supported
- Compare how close our estimates are to the true values

```r
# fit a linear model
  m <- lm(y ~ x)
# get the model coefficients
  coef(summary(m))
```

Is b significant?
How close is the estimate of b to the true value?

Simple example: Linear Regression

Remember: Power = probability of rejecting a false null hypothesis

- One random draw doesn’t tell us that
- We need to do this lots of times and record the results

For loops

Basic structure:

```
for(<index> in <some vector>){
    Do some function
}
```

For loops

```
for(i in 1:100){
   print(i)
 }
```

For loops

```
x_unif<- runif(n = 50,min = 0,max = 100)

for(x in length(x_unif)){

  x_unif[x] <- rnorm(n = 1,mean = x_unif[x],sd = 2)

}
```

Simple example: Linear Regression

```r
nsim <- 400
pval <- numeric(nsim)

for(i in 1:nsim){

  y <- rnorm(n = length(y_det),
             mean = y_det,
             sd = sd)

  m <- lm(y ~ x)

  pval[i] <- coef(summary(m))["x","Pr(>|t|)"]

}

sum(pval < 0.05)/nsim
```

Simple example: Linear Regression

```
nsim <- 400
pval <- numeric(nsim)

for(i in 1:nsim){

  y <- rnorm(n = length(y_det),
             mean = y_det,
             sd = sd)

  m <- lm(y ~ x)

  pval[i] <- coef(summary(m))["x","Pr(>|t|)"]

}

sum(pval < 0.05)/nsim
```

Do these components make sense?

Nested for() loops

```
for(i in 1:10){
  for(j in 1:20){

    print(paste("i = ",i," j = ",j))

  }}
```

Comparing different parameters

```
bvec <- seq(-2, 2, by = 0.1)
power.b <- numeric(length(bvec))

      for(j in 1:length(bvec)){
            for(i in 1:nsim){

              b <- bvec[j]
              y_det <- a + b*x
              y <- rnorm(n = length(y_det),
                         mean = y_det,
                         sd = sd)

              m <- lm(y ~ x)

              #get p-value

              pval[i] <- coef(summary(m))["x","Pr(>|t|)"]

              }#end i lloop
        power.b[j] <- sum(pval< 0.05)/nsim

      }#end j loop
```

Comparing different parameters

```
bvec <- seq(-2, 2, by = 0.1)
power.b <- numeric(length(bvec))

      for(j in 1:length(bvec)){
            for(i in 1:nsim){

              b <- bvec[j]
              y_det <- a + b*x
              y <- rnorm(n = length(y_det),
                         mean = y_det,
                         sd = sd)

              m <- lm(y ~ x)

              #get p-value

              pval[i] <- coef(summary(m))["x","Pr(>|t|)"]

              }#end i lloop
        power.b[j] <- sum(pval< 0.05)/nsim

      }#end j loop
```

Check the slope impacts power.
What is the relationship?

Remainder of Class:

- `Challenge: does sample size impact power?`
  - `Can you use fewer samples when the slope is higher?`
  - `Update your code to check this`
  - `Hints:`
    - `Add another for() loop that loops over different sample sizes`
    - `Use col or pch for plotting different sample sizes`

Before next class:

- Turn in Assignment 2 (due Friday)
- Read 6.1 - 6.2 (6.2.2 is optional)