Week 04: Monte Carlo Inference Continued

Mark M. Fredrickson (mfredric@umich.edu)

Computational Methods in Statistics and Data Science (Stats 406)

Monte Carlo Inference Review

- We have a sample $X_1, X_2, \ldots, X_n$ from a population F with a parameter θ (we often assume IID, but not necessary)
- We will either:
  - Test a **null hypothesis** about θ against an **alternative hypothesis**.
  - **Estimate** the parameter θ
- In both cases, apply a **statistic** to the sample (usually use $T(X_1, \ldots, X_n)$ for testing, $\hat{\theta}(X_1, \ldots, X_n)$ for estimation).
- Our goal is to **understand the operating characteristics** of the statistic.
  - Testing: **size** (Type I error rate) and **power** (probability of rejecting false null)
  - Estimation: **bias**, **variance**, and **mean squared error**.
- Replicate samples from F using PRNG, computing T or $\hat{\theta}$ to get **null/alternative/sampling distribution**.
- Usually we wish to compare statistics or change aspects of the problem to see how the performance changes (ample size, parameter values).

Confidence Intervals

Beyond point estimates

So far, we have looked at the **operating characteristics** of **point estimates**.

Point estimates are useful, but **they do not communicate uncertainty**.

When performing hypothesis tests, we started by either **accepting or rejecting the null hypothesis** and then extended this idea to computing **p-values**.

A similar concept in estimation is to construct a **confidence interval** for θ (target of inference).

Capturing Uncertainty: Confidence Intervals

A $(1-\alpha)\times 100\%$ **confidence interval** (CI) is a pair of **random variables** A and B such that

$$\Pr(A \leq \theta, B \geq \theta) > 1-\alpha \quad \text{and} \quad \Pr(A \leq B) = 1$$

Equivalent notation:

$$\Pr(\theta \in [A, B]) > 1-\alpha$$

Confidence intervals are sometimes called **interval estimators** because A and B are typically functions of the data, $X_1, \ldots, X_n$:

$$A = A(X_1, \ldots, X_n) \quad B = B(X_1, \ldots, X_n)$$

Confidence interval construction

The notation $A(X_1, \ldots, X_n), B(X_1, \ldots, X_n)$ has a natural connection to **test statistics** and highlights one way of construction confidence intervals: **find the set of null hypotheses not rejected at the α-level**.

$$A = \inf\{\theta_0 : T(X_1, \ldots, X_n) \notin \mathcal{R}(\theta_0, \alpha)\} \quad \text{(“infimum”, like minimum)}$$
$$B = \sup\{\theta_0 : T(X_1, \ldots, X_n) \notin \mathcal{R}(\theta_0, \alpha)\} \quad \text{(“supremum”, like maximum)}$$

where $\mathcal{R}(\theta_0, \alpha)$ is the rejection region for T when $H_0 : \theta = \theta_0$ versus $H_1 : \theta \neq \theta_0$.

This is called the “test inversion” method of CI construction.

Connections to hypothesis tests

Many of the concepts from hypothesis tests have direct analogs to CIs (no matter how they are constructed):

- Type I error: The **confidence coefficient/coverage** is $\Pr(\theta \in [A, B])$, which is greater than $1 - \alpha$.
  In other words, the probability of $[A, B]$ **excluding** θ is no more than α.
- Power: Short intervals will exclude more false null hypotheses, so we want $E(B - A)$ (**expected length**) to be small.

As with tests, we would reject any procedure that did not have proper coverage, and then select the procedure with the smallest expected width.

Connections to "estimator $\pm c$"

For $X_i \sim N(\mu, \sigma^2)$, independent with σ^2 known, let's test

$$H_0 : \mu = \mu_0 \quad \text{versus} \quad H_1 : \mu \neq \mu_0$$

at the α level using $\bar{X}$ as the statistic.

We accept the null if

$$-\Phi(1-\alpha/2) \leq \frac{\bar{X} - \mu_0}{\sigma/\sqrt{n}} \leq \Phi(1-\alpha/2)$$

Solving for μ_0

Solving for μ_0 gives us

$$\bar{X} - \Phi(1-\alpha/2)(\sigma/\sqrt{n}) \leq \mu_0 \leq \bar{X} + \Phi(1-\alpha/2)(\sigma/\sqrt{n})$$

or

$$\mu_0 \in \bar{X} \pm \Phi(1-\alpha/2)(\sigma/\sqrt{n})$$

The "plus-minus" type intervals show up for **shift parameters** that change the center of the sampling distribution but not the variance or other properties.

Example: Coverage and Expected Length $X_i \sim N(\mu, \sigma^2)$

Suppose we believe we know σ^2 and

$$X_i \overset{\text{iid}}{\sim} N(\mu, \sigma^2)$$

we want to use $\bar{X}$ to estimate μ.

We will use two facts (without proof):

$$\frac{\bar{X} - \mu}{\sigma/\sqrt{n}} \sim N(0, 1)$$

$$\frac{\bar{X} - \mu}{\hat{s}/\sqrt{n}} \sim t(n-1), \quad \hat{s} = \sqrt{\frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n-1}}$$

This leads to two kinds of $(1-\alpha) \times 100\%$ confidence intervals:

$$\bar{X} \pm z_{\alpha/2}(\sigma/\sqrt{n}), \quad \bar{X} \pm t_{\alpha/2}(n-1)(\hat{s}/\sqrt{n})$$

Simulation to assess coverage and average width

```
> alpha <- 0.05 ; k <- 10000 ; n <- 20
> mu <- 2 # unknown truth
> sigma2 <- 10 ; sigma <- sqrt(sigma2) # known variance
> samples_norm <- rerun(k, rnorm(n, mean = mu, sd = sigma))

> const_norm <- qnorm(1 - alpha/2) * (sigma / sqrt(n))
> cis_norm <- map(samples_norm,
+                 ~ mean(.x) + c(-1, 1) * const_norm)

> const_t <- qt(1 - alpha/2, df = n - 1) / sqrt(n)
> cis_t <- map(samples_norm,
+ ~ mean(.x) + c(-1, 1) * const_t * sd(.x))
```

Estimating coverage

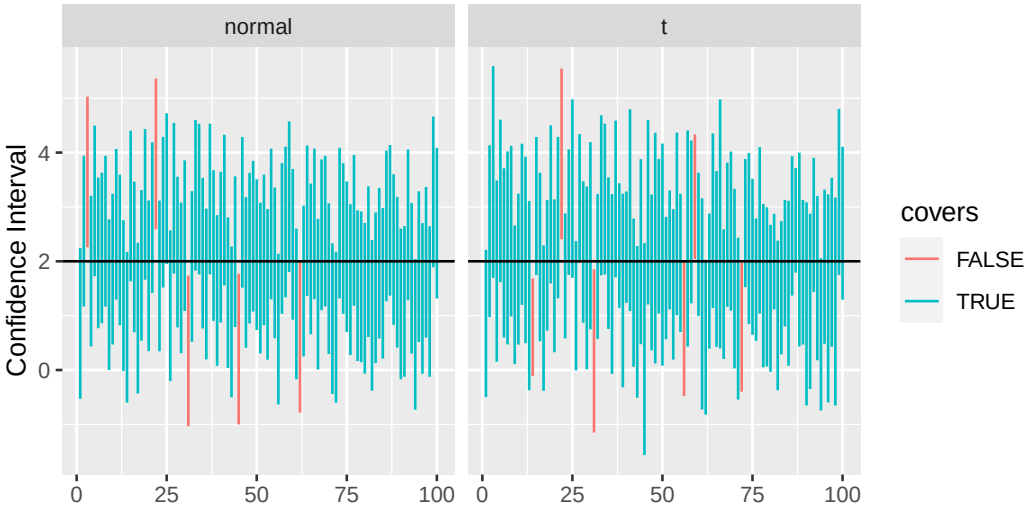
Coverage is $P(A \leq \theta \leq B)$:

```
> map_dbl(cis_norm, ~ .x[1] <= mu && mu <= .x[2]) %>% mean
[1] 0.9472
> map_dbl(cis_t, ~ .x[1] <= mu && mu <= .x[2]) %>% mean
[1] 0.9464
```

Estimating expected width

```
> map_dbl(cis_norm, ~ .x[2] - .x[1]) %>% mean
[1] 2.772
> map_dbl(cis_t, ~ .x[2] - .x[1]) %>% mean
[1] 2.917
```

Visual interpretation



Binomial proportion CI

We've already seen that if we are estimating

$$\theta = P(X \leq t)$$

using

$$\hat{\theta} = \frac{1}{n} \sum_{i=1} I(X_i \leq t)$$

the variables $Y_i = I(X_i \leq t)$ are **Bernoulli with** $P(Y_i = 1) = \theta$.

Using the central limit theorem, if we have a large sample, then

$$\bar{Y} \approx N\left(\theta, \frac{\theta(1-\theta)}{n}\right)$$

Binomial proportion continued

Again,

$$\bar{Y} \approx N\left(\theta, \frac{\theta(1-\theta)}{n}\right)$$

so by our previous analysis the interval:

$$\bar{Y} \pm z_{\alpha/2}\sqrt{\frac{\theta(1-\theta)}{n}}$$

would be a $100 \times (1-\alpha)$ **confidence interval for** θ.

Of course, we don't know θ, but we can stick in an **estimate**:

$$\bar{Y} \pm z_{\alpha/2}\sqrt{\frac{\bar{Y}(1-\bar{Y})}{n}}$$

This is the standard confidence interval for a proportion.

95% CI for $P(X > 1)$, $X \sim Cauchy(0)$

```
> n <- 10000
> x <- rcauchy(n)
> y <- x > 1
> ybar <- mean(y)

> (norm_ci <- ybar + c(-1,1) * qnorm(0.975) * sqrt(ybar * (1 - ybar) / n))

[1] 0.2359 0.2527
```

Test inversion method

Let's use the duality of hypothesis tests to create another interval.

If we were testing $H_0 : P(X > 1) = \theta_0$, the null hypothesis tells us:

$$\sum_{i=1}^{n} I(X_i > 1) = \sum_{i=1}^{n} Y_i \sim \text{Binomial}(n, \theta_0)$$

Let a_0 be the $\alpha/2$ quantile for $\sum_{i=1}^{n} Y_i$ and b_0 be the smallest value such that $P(\sum Y_i \geq b_0) \leq \alpha/2$.

We would accept θ_0 if either

$$a_0 < \sum_{i=1}^{n} Y_i < b_0$$

Finding a_0s and b_0s

We will search over a range of possible θ_0 values:

```
> theta_0 <- seq(0, 1, length.out = 1000)
```

For each, compute the bounds:

```
> a0 <- qbinom(0.025, size = n, prob = theta_0)
> b0 <- qbinom(0.025, size = n, prob = theta_0,
+               lower.tail = FALSE) - 1
```

Reject or Accept θ_0

Now see for which θ_0 we would accept:

```
> w <- sum(y)
> range(theta_0[a0 < w & w < b0])
```

```
[1] 0.2362 0.2523
```

Which shows that the Normal approximation is quite good at $n = 10000$:

```
[1] 0.2359 0.2527
```

Binomial proportions intervals

R provides three methods for computing binomial confidence intervals:

- `t.test`: Using a standard normal approximation
- `binom.test`: Pearson-Clapper interval (similar to exact interval earlier)
- `prop.test`: Wilson "score" interval

Which has proper coverage? Smallest size?

MC Setup

```
> theta <- 0.25
> n <- 20
> k <- 1000
> xs <- rbinom(k, size = n, prob =  theta)
```

```
> tints <- map(xs, function(x) {
+     if (x == 0 || x == n) {
+         return(c(0, 1)) # can't estimate
+     } else {
+        return(t.test(c(rep(1, x), rep(0, n - x)))$conf.int)
+     }
+ })

> bints <- map(xs, ~ binom.test(.x, n)$conf.int)
> pints <- map(xs, ~ prop.test(.x, n)$conf.int)
```

Confidence coefficient

Recall, the **confidence coefficient** is defined as

$$P(A \leq \theta, \theta \leq B)$$

```
> cover <- function(x) { x[1] <= theta && theta <= x[2] }
> (tcover <- map_dbl(tints, cover) %>% mean)

[1] 0.896

> (bcover <- map_dbl(bints, cover) %>% mean)

[1] 0.963

> (pcover <- map_dbl(pints, cover) %>% mean)

[1] 0.982
```

Expected Width

```
> (twidth <- map_dbl(tints, diff) %>% mean)
[1] 0.4044
> (bwidth <- map_dbl(bints, diff) %>% mean)
[1] 0.393
> (pwidth <- map_dbl(pints, diff) %>% mean)
[1] 0.3882
```

Binomial confidence interval routines

- The t-test method tends to **undercover** and has the **largest intervals**. Has computational issues when $X = 0$.
- The Pearson-Clopper method is **slightly conservative** (i.e., overcovers), which is reflected by having slightly higher average width.
- The score test inversion method has the **smallest intervals**, but also has **good coverage**.
- Important: these conclusions are for $n = 20$ and $\theta = 0.25$; other values might have other conclusions (e.g., when $\theta \approx 1$)

Confidence Intervals Summary

- A confidence interval is a **pair of statistics** A and B with

$$P(A \leq \theta, B \geq \theta) \geq 1 - \alpha$$

- There is a **duality** between hypothesis tests and confidence intervals. One way to create intervals is to **invert a set of hypothesis tests**.
- Confidence interval interpretation: set of hypotheses **not rejected** at the α level.
- As with Type I error and power, we can **investigate the operating characteristics** of confidence intervals.
- **Confidence coefficient** (actual probability of including θ, analogous to Type I error)
- **Expected width** (ability to exclude incorrect θ, analogous to power)

Extended Example: Benford's Law

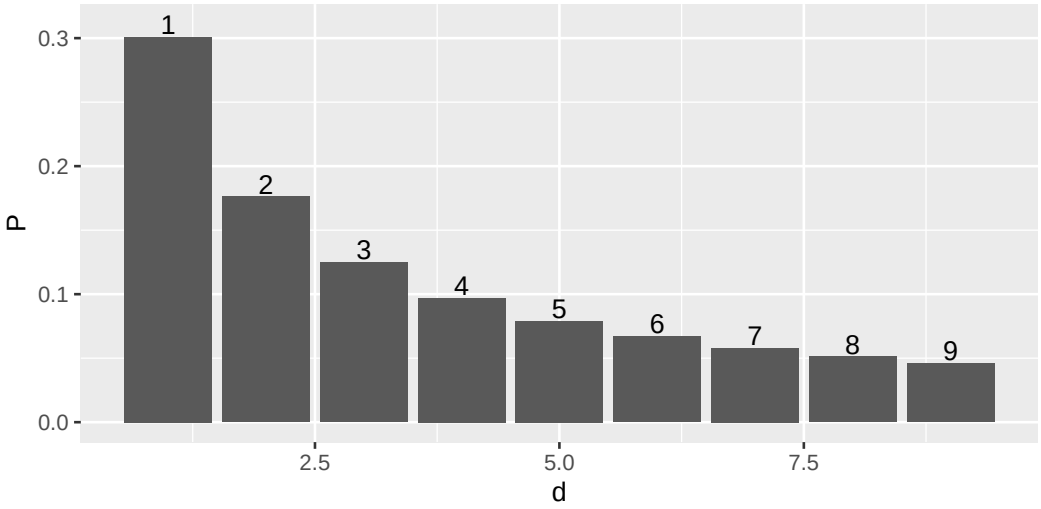
Example: Benford's Law

Benford's Law holds that the distribution of **leading digits** in a collection of numbers spanning several orders of magnitudes will follow the following distribution:

$$\Pr(D = d) = \log_{10}\left(\frac{d+1}{d}\right), \quad d = 1, \ldots, 9$$

```
> dbenford <- function(x) {
+     ifelse(x >= 1 & x <= 9, log((x + 1)/ x, base = 10), 0)
+ }
```

$\Pr(D = d)$ **under Benford's Law**



Using random Ds

Tam Cho and Gaines (2007) investigated **political contributions between political committees** as reported by the FEC. Here are the digit frequencies for 8,396 contributions in 2004 (Table 1):

```
> pol_digits <- c(23.3, 21.1, 8.5, 11.7, 9.5, 4.2, 3.7, 4.0, 14.1) / 100
```

A typical way to analyze these data would be to use a χ^2 test comparing the **expected** to the **observed counts**. Alternatively, Tam Cho and Gaines suggest the statistic:

```
> distance <- function(v)  { sqrt(sum((v - dbenford(1:9))^2)) }
```

Hypothesis Test

We will test the null hypothesis that Benford's Law holds for political contributions versus the alternative that it does not hold (goodness-of-fit test).

If the null hypothesis is true, then the observed distance statistic should be close to zero:

```
> (observed_dist <- distance(pol_digits))
```

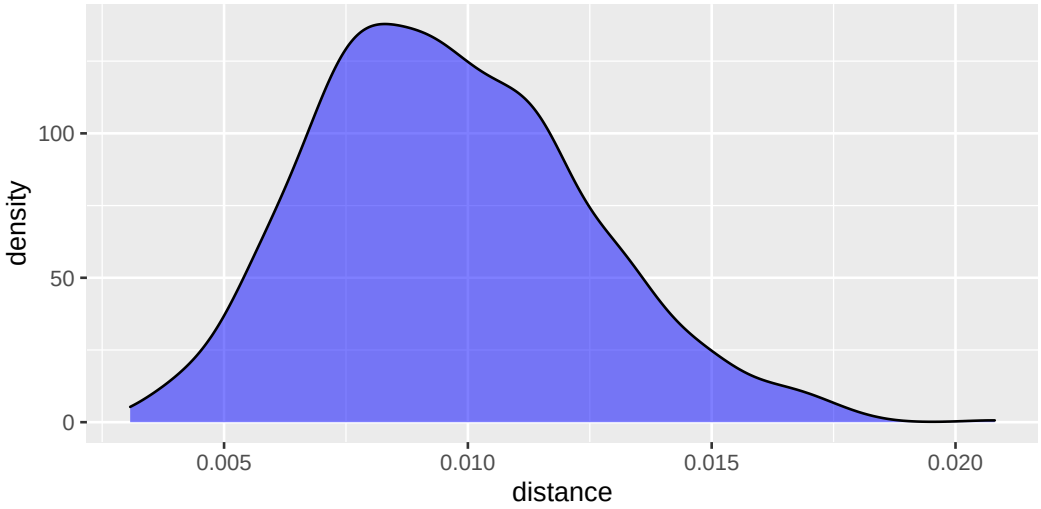
```
[1] 0.1355
```

What is the probability of observing a distance of *at least* 0.1355 if the null hypothesis (Benford's Law) holds?

Distribution of the Test Statistic

```
> rbenford <- function(n) {
+   sample(1:9, size = n, prob = dbenford(1:9), replace = TRUE)
+ }
> n <- 8396
> compute_test_statistic <- function(ds) {
+     probs <- hist(ds, breaks = 0:9, plot = FALSE)$density
+     distance(probs)
+ }
> null_distances <- replicate(1000,
+                             compute_test_statistic(rbenford(n)))
```

Null Distribution



Understanding power of distance test statistic

In order to **pick a rejection region** and **compute the power** of a test statistic, we need to carefully define an alternative hypothesis.

One natural choice would be that **digits are uniformly distributed**: $P(D = d) = 1/9$.

Can we add a parameter θ that controls **how close** to either Bedford or uniform a distribution on digits is?

Parameterizing Alternative

Notice that as $\theta \to \infty$,

$$\frac{d+1+\theta}{d+\theta} \to 1$$

which suggests a model like:

$$P(D=d) = \log_{10}\left(a(\theta)\frac{d+1+\theta}{d+\theta}\right)$$

We need to find the normalizing constant $a(\theta)$.

Finding $a(\theta)$

To get a,

$$\sum_{d=1}^{9} \log_{10}\left(a(\theta)\frac{d+\theta+1}{d+\theta}\right) = 1 \Rightarrow a(\theta)^9 \prod_{d=1}^{9} \frac{d+\theta+1}{d+\theta} = 10$$

Investigating farther, we see

$$a(\theta)^9 \frac{(10+\theta)(9+\theta)\cdots(2+\theta)}{(9+\theta)(8+\theta)\cdots(1+\theta)} = a(\theta)^9 \frac{10+\theta}{1+\theta} = 10$$

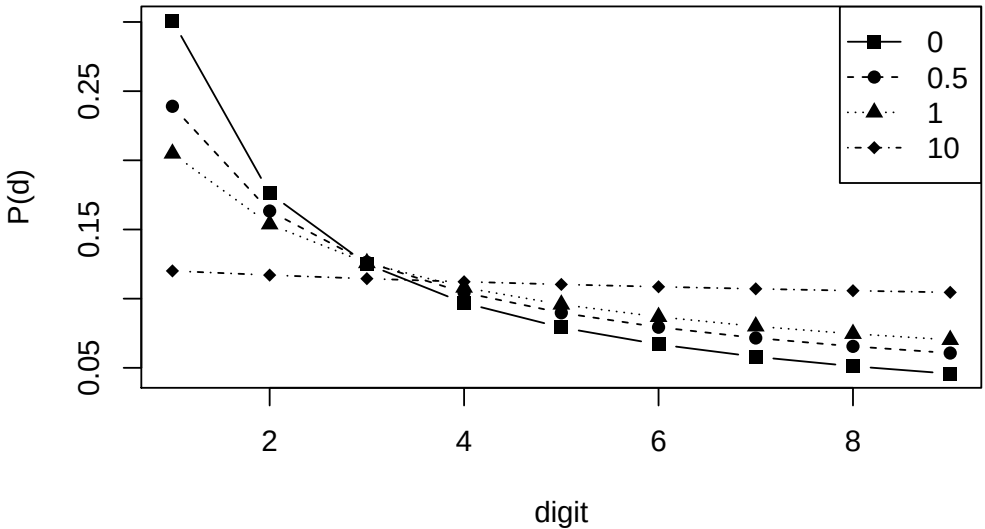
so

$$a(\theta) = \left[\frac{10(1+\theta)}{10+\theta}\right]^{1/9}$$

Putting it together

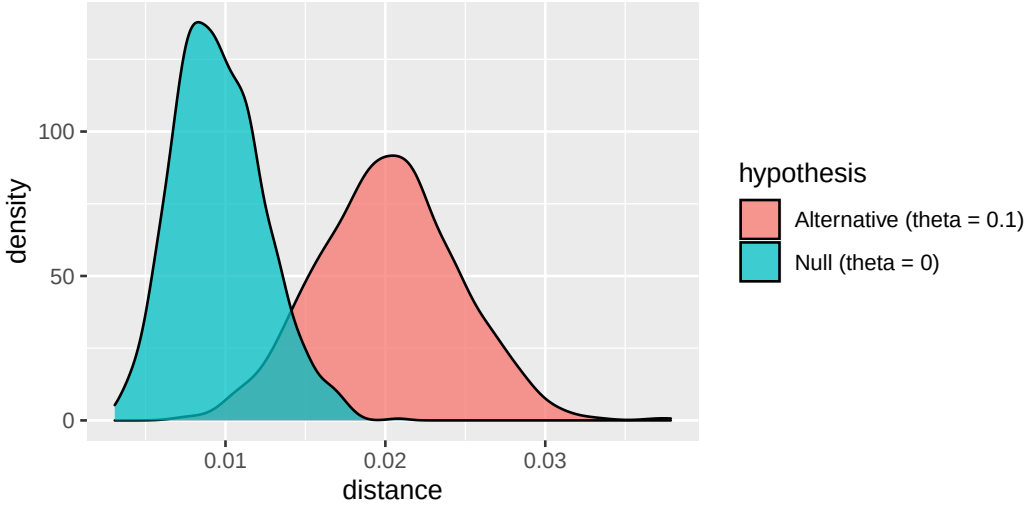
$$P(D = d) = \log_{10}\left(\left[\frac{10(1+\theta)}{10+\theta}\right]^{\frac{1}{9}} \frac{d+\theta+1}{d+\theta}\right), \theta \geq 0$$

```
> alt_dist <- function(theta) {
+   a <- (10 * (1 + theta) / (10 + theta))^(1/9)
+   log10(a * (1:9 + theta + 1) / (1:9 + theta))
+ }
```


P(d)
0.05
0.15
0.25
2
4
6
8
digit
0
0.5
1
10

Alternative distribution $\theta = 0.1$

```
> alt_0.1 <- replicate(1000, {
+     a_sample <- sample(1:9, size = n, replace = TRUE,
+                        prob = alt_dist(0.1))
+     compute_test_statistic(a_sample)
+ })
>
```


density
100
50
0
0.01
0.02
0.03
distance
hypothesis
Alternative (theta = 0.1)
Null (theta = 0)

p-value for the hypothesis test

```
> (p_value <- mean(null_distances >= observed_dist)) # P(T > t)
```

```
[1] 0
```

The observed test statistic was larger than any sample we generated (so the p-value was zero) and is 45 standard deviations from the mean of the null distribution.

With *extremely high confidence*, we can reject the null hypothesis that these data were a sample from a population that follows Benford's Law.

Power at $\alpha = 0.001$ and $\theta = 0.1$

First, we need to find the 99% quantile under the null:

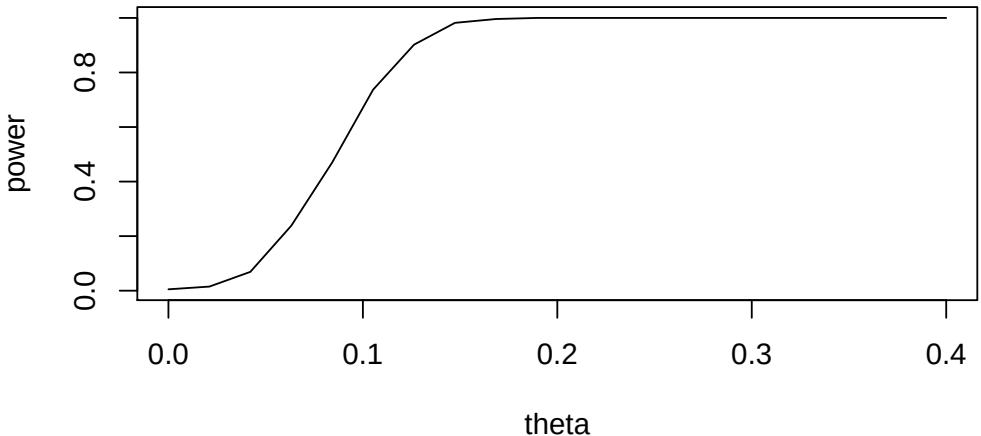
```
> (rejection_cutoff <- quantile(null_distances, 0.999))

  99.9%
0.01798

> mean(alt_0.1 >= rejection_cutoff)

[1] 0.694
```

Power Curves

We saw power at one particular point $\theta = 0.1$. What about **other values of θ**?

We can compute power for many values of θ (holding our α level fixed) to see how it changes.

```
> thetas <- seq(0, 0.4, length.out = 20)
> power_curve <- map_dbl(thetas, function(theta) {
+     alt <- replicate(1000, {
+         a_sample <- sample(1:9, size = n,
+                            replace = TRUE, prob = alt_dist(theta))
+         compute_test_statistic(a_sample)
+     })
+     mean(alt >= rejection_cutoff)
+ })
```


power
0.0
0.4
0.8
0.0
0.1
0.2
0.3
0.4
theta

More on Benford's Law

If you are interested in learning more about Benford's Law,

- A Simple Explanation of Benford's Law by R. M. Fewster.
- Breaking the (Benford) Law: Statistical Fraud Detection in Campaign Finance by Wendy Tam Cho and Brian Gaines