Quantile Functions and the Inversion Method

Mark M. Fredrickson (mfredric@umich.edu)

Computational Methods in Statistics and Data Science (Stats 406)

Quantile Functions

Distribution Functions and Quantile Functions

Recall: the **distribution function** for a random variable is

$$F_X(t) = \Pr(X \leq t) = \begin{cases} \int_{-\infty}^{t} f(x)\, dx & \text{(continuous)} \\ \sum_{x=-\infty}^{t} \Pr(X = t) & \text{(discrete)} \end{cases}$$

Some properties of all F_X:

- $0 \leq F_X(x) \leq 1$ for all $x \in (-\infty, \infty)$
- F_X is **non-decreasing** and **right continuous**: $x_1 \geq x_2 \Rightarrow F_X(x_1) \geq F_X(x_2)$.

This allows defining the **quantile function**:

$$Q_X(u) = F_X^{-1}(x) = \inf\{x : F(x) \geq u\}, \quad u \in [0, 1]$$

(finds smallest x (or limit from the right) where $F(x)$ is at least as large as u)

Quantile-Quantile Plots

Like distribution functions, **quantile functions uniquely define RVs**.

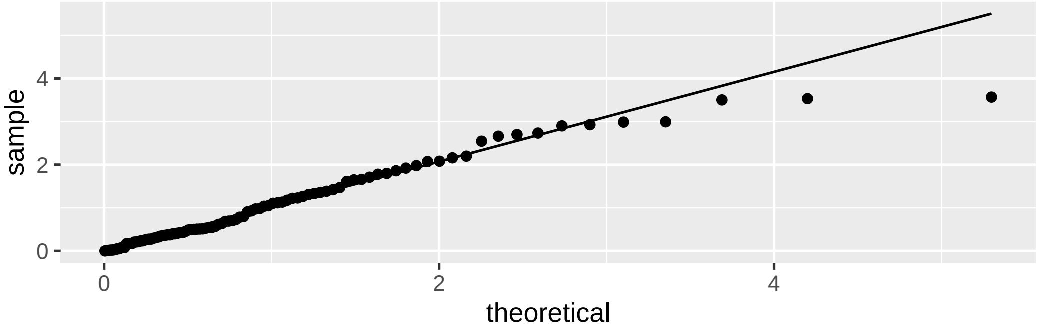
While two two RVs with F_X and F_Y might be on different scales, both Q_X and Q_Y map $(0, 1) \to \mathbb{R}$.

The **quantile-quantile plot** computes all points $(Q_X(u), Q_Y(u))$ for many points $u \in (0, 1)$.

If $F_X = F_Y$, the points fall on the 45 degree line.

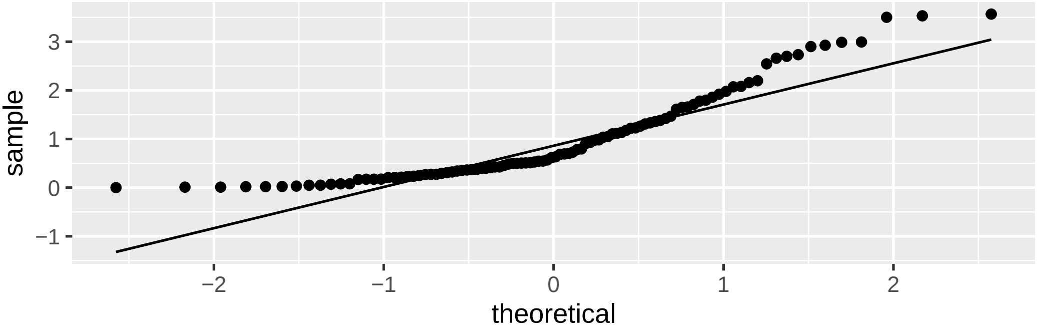
Example QQ Plot: Same distribution

```
> x <- rexp(100)
> ggplot(data.frame(sample = x), aes(sample = sample)) +
+ geom_qq(distribution = qexp) + geom_qq_line(distribution = qexp)
```



Example QQ Plot: Different distribution

```
> ggplot(data.frame(sample = x), aes(sample = sample)) +
+ geom_qq(distribution = qnorm) + geom_qq_line(distribution = qnorm)
```



Example: Uniform on $(0, \theta)$

Recall $X \sim U(0, \theta)$:

$$f(x) = \frac{1}{\theta}, \quad 0 < x < \theta, \theta > 0$$

The distribution function can be found by:

$$F(t) = \int_0^t \frac{1}{\theta}\, dx = \frac{x}{\theta}\Big|_0^t = \frac{t}{\theta}$$

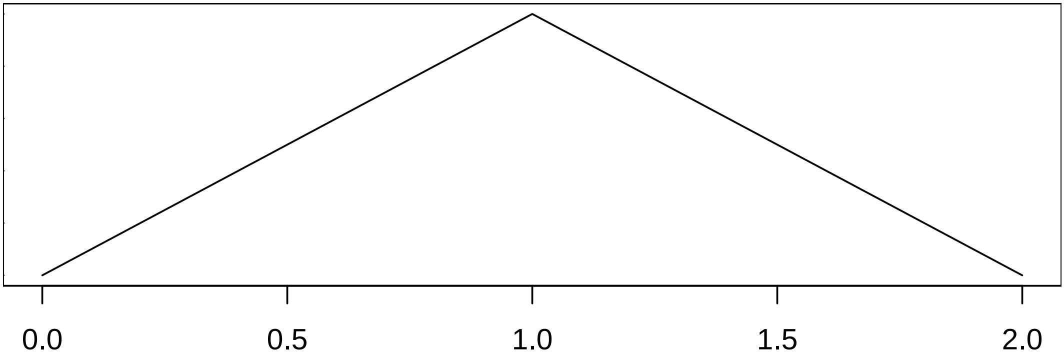
Since X is continuous, we can just take the inverse:

$$u = \frac{t}{\theta} \Rightarrow Q_{(0,\theta)}(u) = \theta u$$

Example: Triangular distribution

Suppose X has density function:

$$f(x) = \begin{cases} x & 0 \leq x \leq 1 \\ 2 - x & 1 < x \leq 2 \end{cases}$$



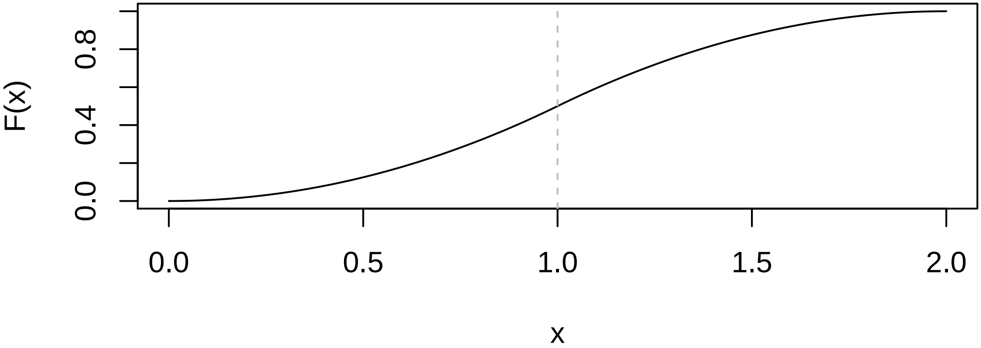
Cumulative Distribution Function

To get the quantile function, we need to derive the **cumulative distribution function (CDF)**. Consider these cases:

- $t < 1$: $F(t) = \int_0^t x\,dx = \frac{t^2}{2}$
- $t = 1$: $F(1) = \int_0^1 x\,dx = \frac{1}{2}$
- $t > 1$:

$$\begin{aligned} F(t) &= \int_0^t f(x)\,dx = \int_0^1 f(x)\,dx + \int_1^t f(x)\,dx \\ &= F(1) + \int_1^t 2 - x\,dx \\ &= \frac{1}{2} + \left[2 - \frac{x^2}{2} \bigg|_{x=1}^{t} \right] = 2t - \frac{t^2}{2} - 1 \end{aligned}$$

CDF



Triangular quantile function

We found a **piece-wise distribution** distribution function:

$$F(t) = \begin{cases} \frac{t^2}{2} & 0 \leq x \leq 1 \\ 2t - \frac{t^2}{2} - 1 & 1 < x \leq 2 \end{cases}$$

This implies the **quantile function will also be piece-wise**.

There was a break at $x = 1$, so there were be a break at $F(1) = 1/2 = u$:

$$u \leq 1/2 : u = F(x) = x^2/2 \Rightarrow x = \sqrt{2u} = x$$

$$u > 1/2 : u = F(x) = 2x - x^2/2 - 1 \Rightarrow x = 2 - \sqrt{2 - 2u}$$

$$Q(u) = \begin{cases} \sqrt{2u} & 0 \leq u \leq 1/2 \\ 2 - \sqrt{2 - 2u} & 1/2 < x \leq 1 \end{cases}$$

Inversion Method

Connection to Monte Carlo Methods

When discussing **psuedorandom number generation** (PRNGs), we considered IID

$$U \sim U(0, 1)$$

Notice: the **domain** of any quantile function is precisely (0,1)!

Question: What is the distribution of the random variable $Q(U)$?

Inversion Method, Continuous RVs

Suppose X is a **continuous random variable** with quantile function $Q_X(u)$.

For a **uniform random variable** $U \sim \text{Uniform}(0,1)$ the variable

$$Y = Q(U)$$

has the same distribution as X.

For example, as we already knew:

$$Q_{(0,\theta)}(U) = \theta U \sim U(0,\theta)$$

Observations

Before proceeding to the proof, let's make two observations:

Since X is continuous,

$$\inf\{x : F(x) \geq u\} = \inf\{x : F(x) = u\}$$

Therefore, for any u

$$F(\inf\{x : F(x) = u\}) = u$$

The cumulative distribution function for $U \sim U(0,1)$ is

$$F_U(u) = P(U \leq u) = \int_0^u 1\,dx = u$$

Proof of Inversion Method

Goal: Show that $Q_X(U)$ has same distribution as X (for any continuous X).

$$\begin{aligned}
\Pr(Q(U) \leq x) &= \Pr(\inf\{x : F(x) \geq U\} \leq x) && \text{(definition)} \\
&= \Pr(\inf\{x : F(x) = U\} \leq x) && \text{(continuity)} \\
&= \Pr(F(\inf\{x : F(x) = U\}) \leq F(x)) && \text{(F is non-dec.)} \\
&= \Pr(U \leq F(x)) && \text{(first obs.)} \\
&= F(x) && \text{(def. F_U)} \\
&= \Pr(X \leq x)
\end{aligned}$$

Interesting Corollary: Probability Integral Transformation

Recall: If for any t, $P(X \leq t) = P(Y \leq t)$, then X and Y have the same distribution.

For continuous X and $U \sim U(0,1)$, consider the following:

$$
\begin{aligned}
P(Q_X(U) \leq t) &= P(X \leq t) \Rightarrow \\
P(F_X(Q_X(U)) \leq F_X(t)) &= P_X(F(X) \leq F_X(t)) \Rightarrow \\
P(U \leq F_X(t)) &= P(F_X(X) \leq F_X(t)) \Rightarrow \\
F_X(t) &= P(F_X(X) \leq F_X(t))
\end{aligned}
$$

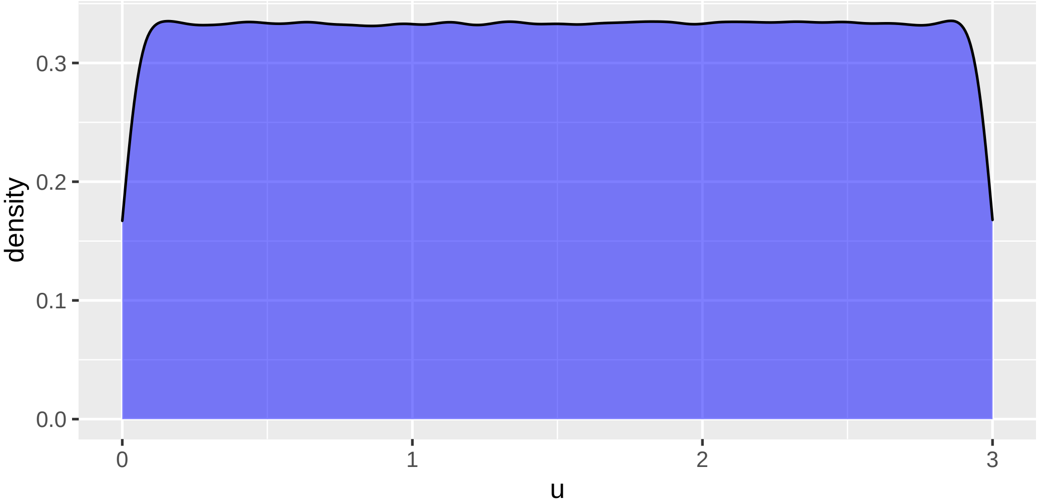
Writing $u = F_X(t)$, $Y = F_X(X)$, we have $P(Y \leq u) = u$, **the uniform CDF**!

For continuous X, $F_X(X) \sim U(0,1)$ (probability integral transformation).

Inv. Method for $U(0, \theta)$

Here is an implementation of the quantile function we found for $U(0, \theta)$:

```
> Q_theta <- function(u, theta) {
+     u * theta
+ }
> k <- 10e5
> u_0_3 <- Q_theta(runif(k), 3)
```


density
0.3
0.2
0.1
0.0
0
1
2
3
u

Inv. Method for Triangular RVs

Recall we found that for X with density function:

$$f(x) = \begin{cases} x & 0 \leq x \leq 1 \\ 2 - x & 1 < x \leq 2 \end{cases}$$

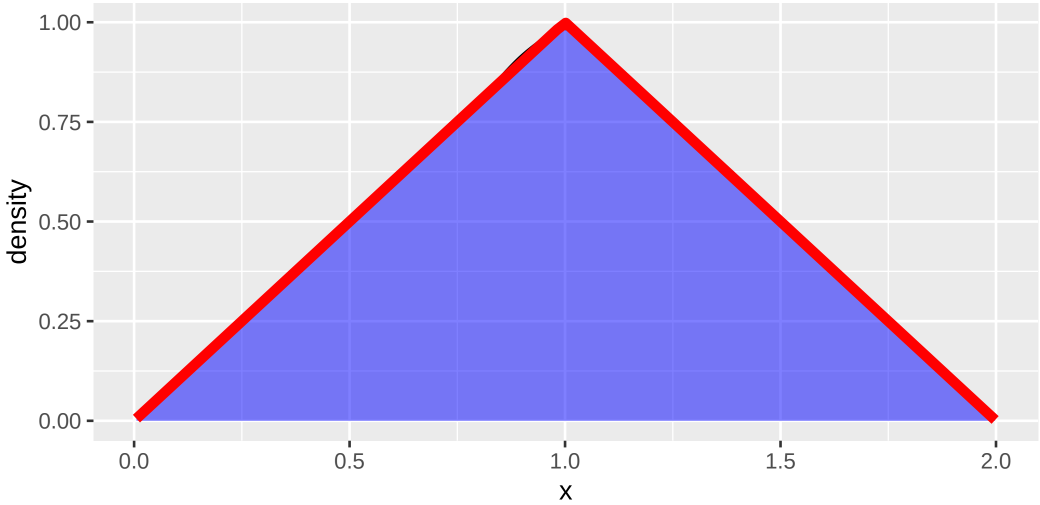
the **quantile function** is given by

$$Q(u) = \begin{cases} \sqrt{2u} & 0 \leq u \leq 1/2 \\ 2 - \sqrt{2 - 2u} & 1/2 < x \leq 1 \end{cases}$$

R Implementation

```
> Q_tri <- function(u) {
+     ifelse(u <= 1/2,
+            sqrt(2 * u),
+            2 - sqrt(2 - 2 * u))
+ }
> Q_tri(c(0.25, 0.5, 0.75, 1))

[1] 0.7071 1.0000 1.2929 2.0000

> triangulars <- Q_tri(runif(100000))
```

Triangular random variables



Estimating the variance

From inspection, we can realize that the triangular PDF is symmetric about 1, so the **mean and median** must be 1.

$$f(x) = \begin{cases} x & 0 \leq x \leq 1 \\ 2 - x & 1 < x \leq 2 \end{cases}$$

We could find the variance analytically, but we'll estimate it. Recall,

$$\mathrm{Var}\left(X\right) = \mathrm{E}\left(X^2\right) - \mathrm{E}\left(X\right)^2$$

```
> x <- Q_tri(runif(10000))
> mean(x^2) - 1^2
```

```
[1] 0.1743
```

Example: $f(x) = \theta x^{\theta-1}$

Suppose we have PDF:

$$f(x) = \theta x^{\theta-1}, \quad 0 \leq x \leq 1, 0 \leq \theta$$

$$F(x) = \int_0^x \theta t^{\theta-1}\,dt = \left. t^{\theta} \right|_0^x = x^{\theta}$$

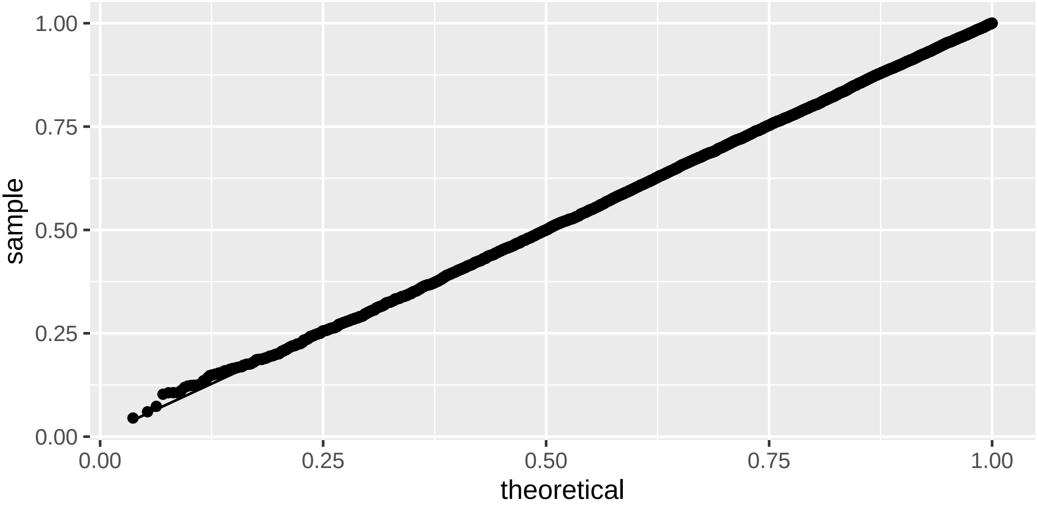
For this function, we can simply solve for x in the expression $u = x^{\theta}$:

$$Q(u) = u^{1/\theta}$$

Implementing in R

```
> rx <- function(n, theta) {
+     runif(n)^(1/theta)
+ }
> xs_theta3<- rx(10000, 3)
```

QQ-plot



Comparing Estimators of θ

Suppose we have n observations and we think they follow F for some θ. What will be a good estimator of θ?

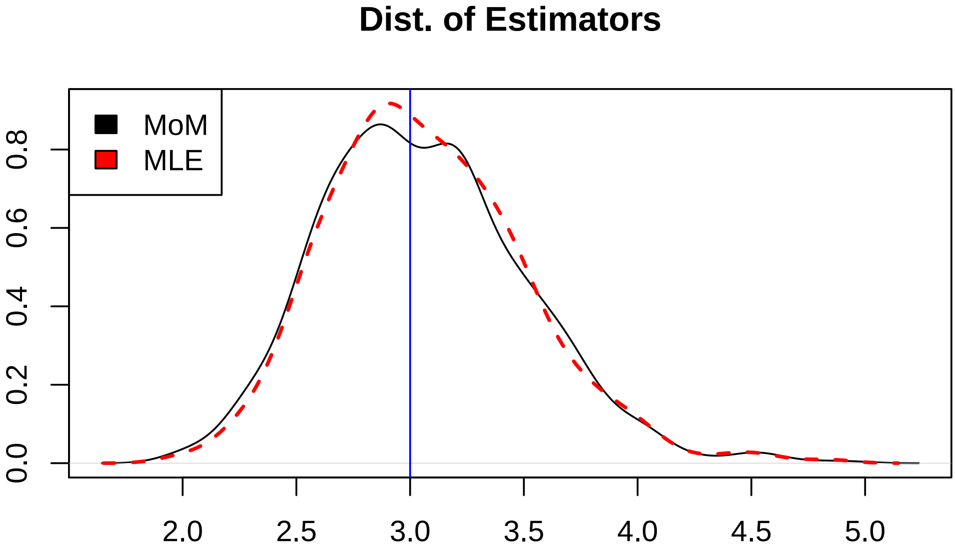
During our statistical review we found **two estimators** of θ:

- **Method of Moments**:
$$\tilde{\theta} = \frac{\bar{X}_n}{1 - \bar{X}_n}$$
- **Maximum Likelihood**:
$$\hat{\theta} = -\frac{n}{\sum_{i=1}^{n} \log(X_i)}$$

We will evaluate these for **mean squared error**.

Setup the Monte Carlo simulation

```
> mom <- function(x) { mean(x) / (1 - mean(x)) }
> mle <- function(x) { - length(x) / sum(log(x)) }

> theta <- 3
> n <- 50
> k <- 1000
> samples <- rerun(k, rx(n, theta = 3))
> moms <- map_dbl(samples, mom)
> mles <- map_dbl(samples, mle)
```


Dist. of Estimators
MoM
MLE
0.0
0.2
0.4
0.6
0.8
2.0
2.5
3.0
3.5
4.0
4.5
5.0

MSE

```
> mean((moms - theta)^2)

[1] 0.2074

> mean((mles - theta)^2)

[1] 0.1995
```

Inversion Method for Discrete Random Variables

Discrete versus Continuous

In our proof of the inversion method for **continuous random variables**, we used the continuity of $f(x)$ to imply the continuity of $F(x)$ and make the equality:

$$Q(u) = \inf\{x : F(x) \geq u\} = \inf\{x : F(x) = u\}$$

(we knew that there must be at least one point where $F(x) = u$)

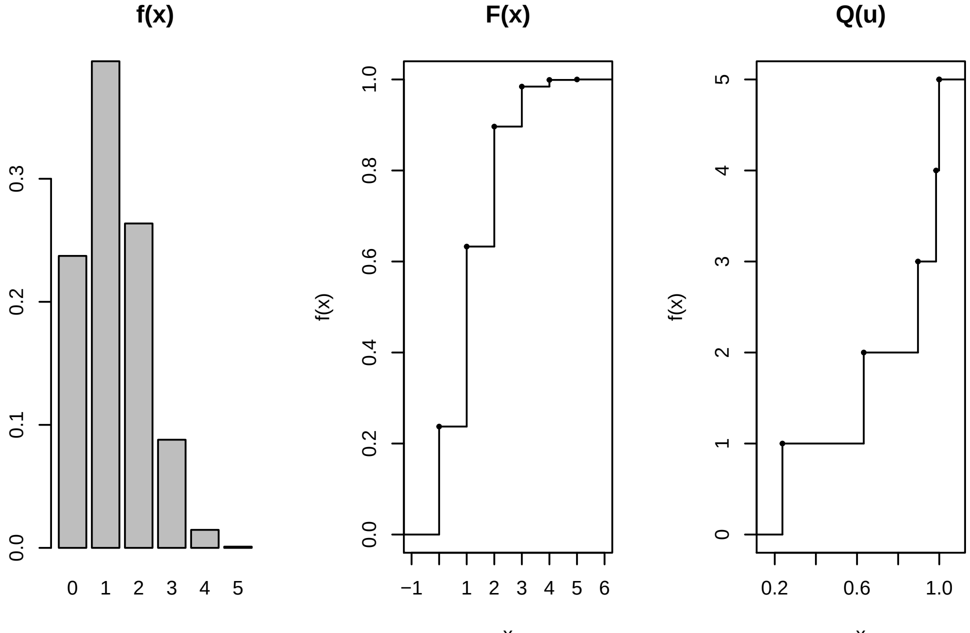
This **does not hold** for **discrete distributions**. E.g., $X \sim \text{Bin}(1, 0.5)$, $F(x) \neq 0.75$ for any x.

Note: $Q(u)$ remains well defined in the discrete case (it is a “step-function”).

Example: Binomial

For $X \sim \mathrm{Bin}(n, p)$,

$$\begin{aligned}
p(x) &= \Pr(X = x) = \binom{n}{x} p^x (1-p)^{n-x} \\
F(x) &= \Pr(X \leq x) = \sum_{i=0}^{x} p(i) \\
Q(u) &= \min\{x : F(x) \geq u\}
\end{aligned}$$


f(x)
0.0
0.1
0.2
0.3
0
1
2
3
4
5
F(x)
f(x)
0.0
0.2
0.4
0.6
0.8
1.0
−1
1
2
3
4
5
6
Q(u)
f(x)
0
1
2
3
4
5
0.2
0.6
1.0

Alternate Characterization

Since $Q(u)$ is a **step function**, we have for **any discrete RV**:

$$\begin{aligned} Q(u) &= \min\{x : F(x) \geq u\} \\ &= \min\left\{x : \sum_{i=0}^{x} p(i) \geq u\right\} \end{aligned}$$

Therefore we it must be that if $Q(u) = x$, then

$$\sum_{i=0}^{x} p(i) \geq u \quad \text{and} \quad \sum_{i=0}^{x-1} p(i) < u$$

In other words:

$$Q(u) = x \text{ such that } F(x-1) < u \leq F(x)$$

Inversion Method, Discrete Case

Since $Q(u)$ is a step function for the discrete case, we can skip computing it expressly and just use $p(x)$.

1. Initialize $v = 0$
2. Generate $U \sim U(0, 1)$
3. Walk up the values $(x_1 < x_2 < \ldots)$:
    - $v = v + p(x_i)$
    - If $v \geq u$, return $X = x_i$

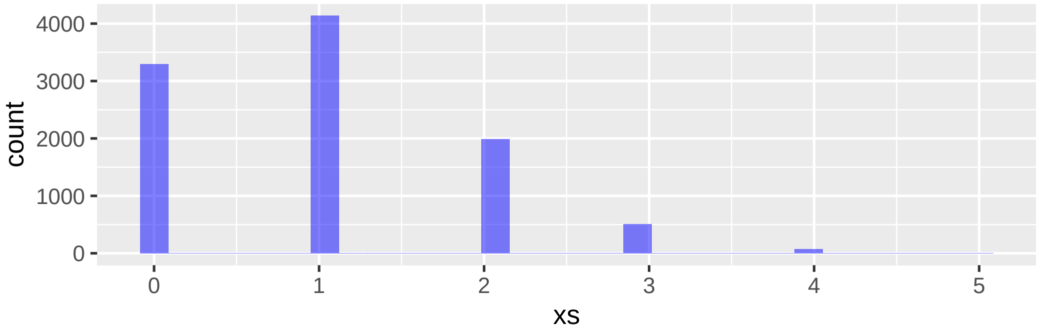
Sometimes we can explicitly find X based on $F(x)$ directly.

Binomial Q in R

```
function(t, n, p) {
  so_far <- 0
  for (i in 0:n) {
      so_far <- so_far + dbinom(i, n, p)
      if (so_far >= t) {
          return(i)
      }
  }
  return(n) # this shouldn't happen, but be safe!
}
<bytecode: 0x7fdb8180cf98>
```

Inversion method with Binomial

```
> xs <- map_dbl(runif(10000), ~ Q_bin(.x, 5, 0.2)) # non-vectorized Q_bin
```



Example Revisited: Benford's Law

Recall the definition of Benford's Law for **leading digits**:

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

Previously, we relied on R's method for sampling from a finite set. We'll reimplement using the inversion method.

Benford CDF

Notice that we can write the cumulative distribution function for D as:

$$\begin{aligned}
\Pr(D \leq d) &= \sum_{i=1}^{d} \Pr(D = i) \\
&= \sum_{i=1}^{d} \log_{10}\left(\frac{i+1}{i}\right) \\
&= \log_{10}\left(\frac{\prod_{i=2}^{d+1} i}{\prod_{i=1}^{d} i}\right) \\
&= \log_{10}\left(\frac{2 \times 3 \times \cdots \times d \times (d+1)}{1 \times 2 \times \cdots \times d}\right) \\
&= \log_{10}(d+1)
\end{aligned}$$

Quantile Function of D

Since D is discrete, the quantile function is a step function with steps defined at $F(d)$, for $d = 1, \ldots, 9$.

$$\begin{aligned} Q(u) &= d \text{ s. t. } F(d-1) < u \leq F(d) \\ &= d \text{ s. t. } \log_{10}(d) < u \leq \log_{10}(d+1) \\ &= d \text{ s. t. } d < 10^u \leq d+1 \\ &= d \text{ s. t. } d-1 < 10^u - 1 \leq d \\ &= \lceil 10^u - 1 \rceil \end{aligned}$$

Sampling from *D*

```
> rbenford <- function(n) { ceiling(10^runif(n) - 1) }

> k <- 10000
> rbind(log10((2:10) / (1:9)), # analytical P(D = d)
+       table(rbenford(k)) / k) # empirical P(D = d)

          1      2      3       4       5       6       7
[1,] 0.3010 0.1761 0.1249 0.09691 0.07918 0.06695 0.05799
[2,] 0.3093 0.1782 0.1150 0.09700 0.07420 0.07160 0.06230
           8       9
[1,] 0.05115 0.04576
[2,] 0.04660 0.04580
```

Saving results for speed

It can be expensive to compute the $p(x)$, so if we are going to **many RVs**, it can help to **pre-compute the CDF**.

The **Poisson distribution** with mean λ, has PMF:

$$P(X = k) = \frac{e^{-\lambda}\lambda^k}{k!} = \frac{\lambda}{k}\frac{e^{-\lambda}\lambda^{k-1}}{(k-1)!} = \frac{\lambda}{k}P(X = k-1), \quad P(X=0) = e^{-\lambda}$$

Strategy:

- Generate all the $U_i \sim U(0,1)$ we need
- Compute the CDF from $F(0)$ to $F(k) > \max u$
- Use the table to connect U_i to X_i

Making the CDF

```
> makeCDFTable <- function(u, lambda) {
+     maxu <- max(u)
+     # base case: x= 0
+     fx <- exp(-lambda)
+     cdf <- c(fx)
+     x <- 0
+     # build table until F(k) >= maxu
+     while (last(cdf) < maxu) {
+         x <- x + 1
+         fx <- fx * lambda / x
+         cdf <- c(cdf, last(cdf) + fx)
+     }
+     return(cdf)
+ }
```

Checking our results

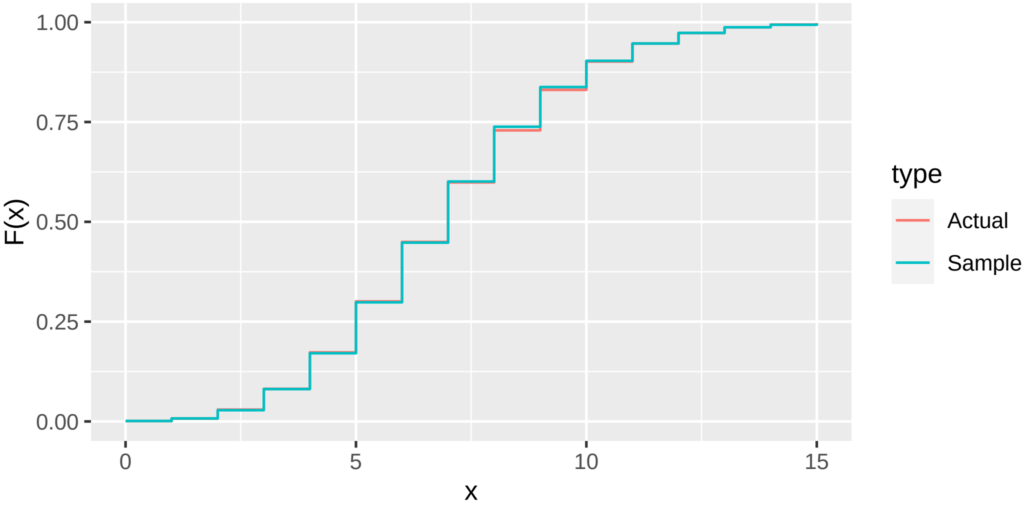
```
> makeCDFTable(c(0, 0.95), 3)

[1] 0.04979 0.19915 0.42319 0.64723 0.81526 0.91608 0.96649

> ppois(0:6, lambda = 3)

[1] 0.04979 0.19915 0.42319 0.64723 0.81526 0.91608 0.96649
```

Using the table

```
> rpoisson <- function(n, lambda) {
+     u <- runif(n)
+     tbl <- makeCDFTable(u, lambda)
+     map_dbl(u, function(u_i) {
+         min(which(tbl >= u_i)) - 1 ## table is defined on 0, 1, 2, ...
+     })
+ }
```

Checking our work

```
> df <- data.frame(
+     type = c(rep("Sample", 16), rep("Actual", 16)),
+     x = c(0:15, 0:15),
+     y = c(ecdf(rpoisson(10000, 7))(0:15),
+           ppois(0:15, lambda = 7)))
> plt <- ggplot(df, aes(x = x, y = y, color = type)) +
+     geom_step() + labs(y = "F(x)")
```


1.00
0.75
0.50
0.25
0.00
F(x)
0
5
10
15
x
type
Actual
Sample

Summary

- **Quantile functions** "invert" the CDF to tell give us $Q(u) = \inf\{x : F(x) \geq u\}$ (smallest x that has CDF value of at least u).
- For RV X with quantile function Q_X, $Q_X(U) \sim X$. (**inversion method**)
- Useful trick: look for changes in CDF (e.g. $x = 1$), there will be regions in the quantile at same places (e.g., at $u = F(1)$).
- Discrete case: Can always fall back to using CDF directly, often opportunities to take short cuts.

Other Examples

Product failures

Suppose we want to model **failure times** of products.

Each of these products has a **usable lifespan**, say one year, after which it will be considered to have failed, but it might fail at **any time between 0 and 1** lifespans.

In other words, we can ask, “at what proportion of the usable life did the product fail?”

We can use our $f(x)$ to model failure time probabilities. $\theta < 1$ indicates most products tend to **early**, whereas $\theta > 1$ indicates more products **late failures**.

Hypothesis test

Here are some observed data:

```
> fail_times[1:5] # just the first 5 of 20

[1] 0.06501 0.05342 0.16014 0.04545 0.13355
```

we will test the hypothesis:

$$H_0 : \theta = 1/2 \quad \text{vs} \quad H_1 : \theta = 1 \text{ (uniform failure prob.)}$$

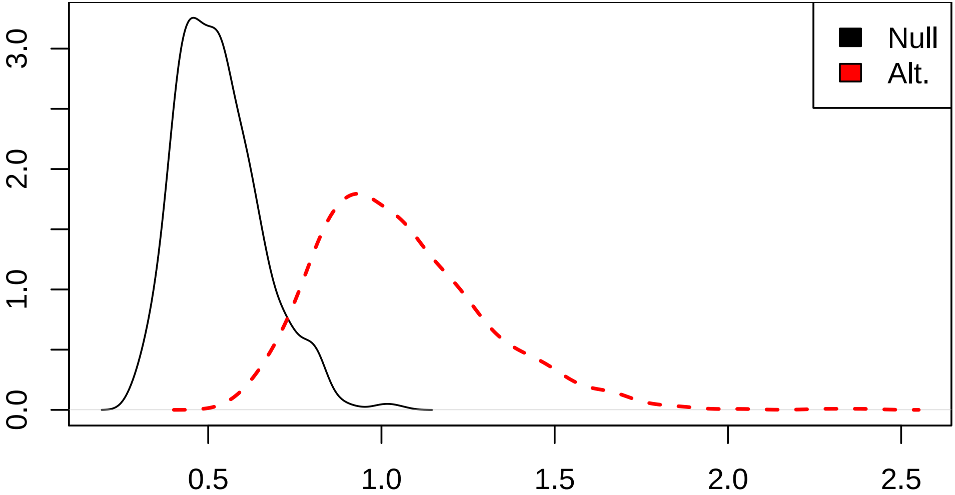
```
> n <- length(fail_times) # 20
> k <- 1000
> null_samples <- rerun(k, rx(n, theta = 1/2))
> alt_samples  <- rerun(k, rx(n, theta = 1))
```

Selecting a test statistic

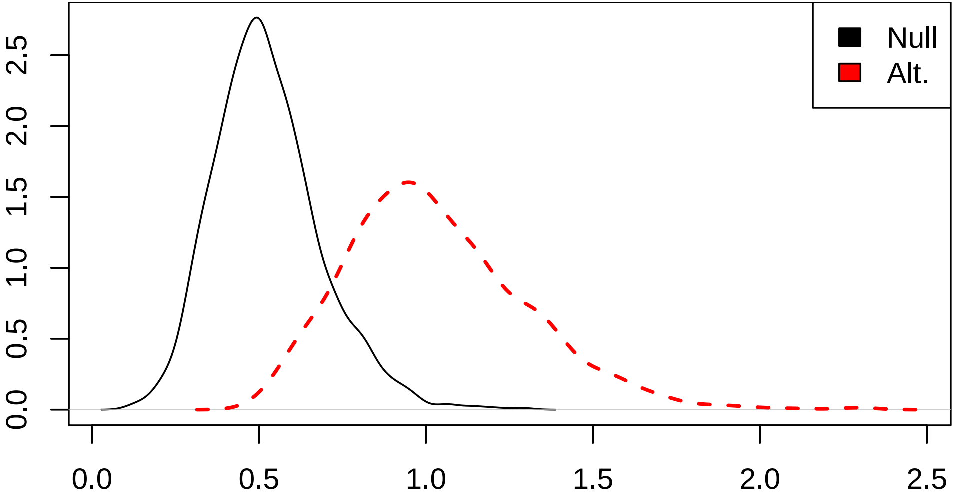
We have two ready made test statistics: the **MoM estimator** and the **MLE estimator** for θ.

```
> null_mle <- map_dbl(null_samples, mle)
> null_mom <- map_dbl(null_samples, mom)
> alt_mle <- map_dbl(alt_samples, mle)
> alt_mom <- map_dbl(alt_samples, mom)
```

MLE distributions



MoM distributions



Picking rejection region, computing power

For both estimators, **right tailed tests** would be reasonable choices. We'll fix $\alpha = 0.10$

```
> cutoff_mle <- quantile(null_mle, 0.9)
> cutoff_mom <- quantile(null_mom, 0.9)
```

Power:

```
> mean(alt_mle > cutoff_mle)

[1] 0.964

> mean(alt_mom > cutoff_mom)

[1] 0.893
```

Performing the test

```
> (observed_mle <- mle(fail_times))

[1] 0.4235

> ## accept if true
> observed_mle <= cutoff_mle

 90%
TRUE
```

One-sided confidence bound for θ

Let's use the **test inversion** method to create a confidence interval for θ.

```
> thetas <- seq(0.001, 1, length.out = 1000)
> test_theta <- function(theta) {
+     samples <- rerun(k, rx(n, theta))
+     null_mles <- map_dbl(samples, mle)
+     cutoff <- quantile(null_mles, c(0.025, 0.975))
+     cutoff[1] <= observed_mle & observed_mle <= cutoff[2]
+ }
```

```
> accepted <- map_lgl(thetas, test_theta)
> min(thetas[accepted])

[1] 0.247

> max(thetas[accepted])

[1] 0.634
```

Another quantile method example

Suppose we have X with the density function:

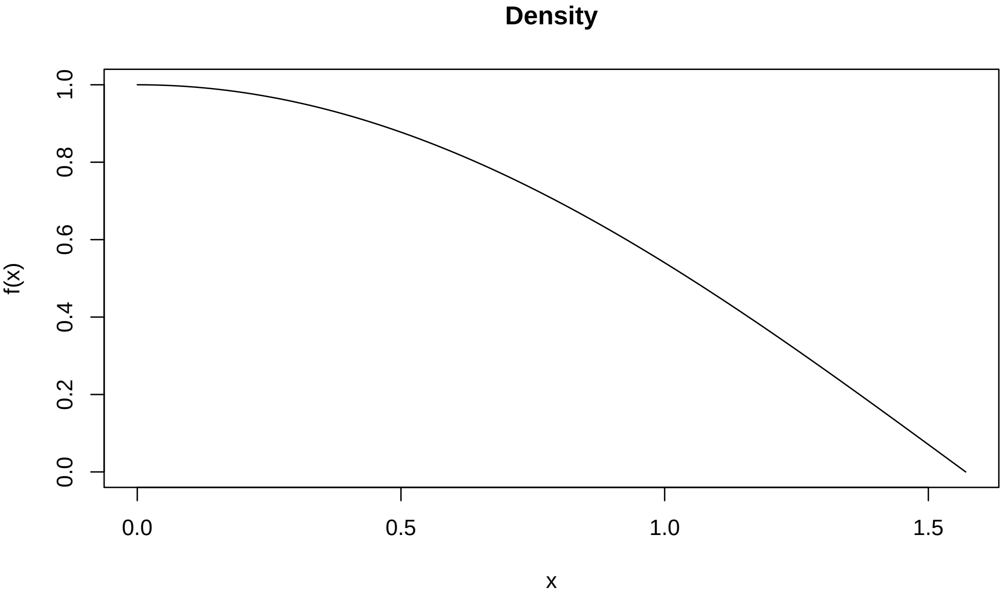
$$f(x) = \cos(x), 0 \leq x \leq \frac{\pi}{2}$$

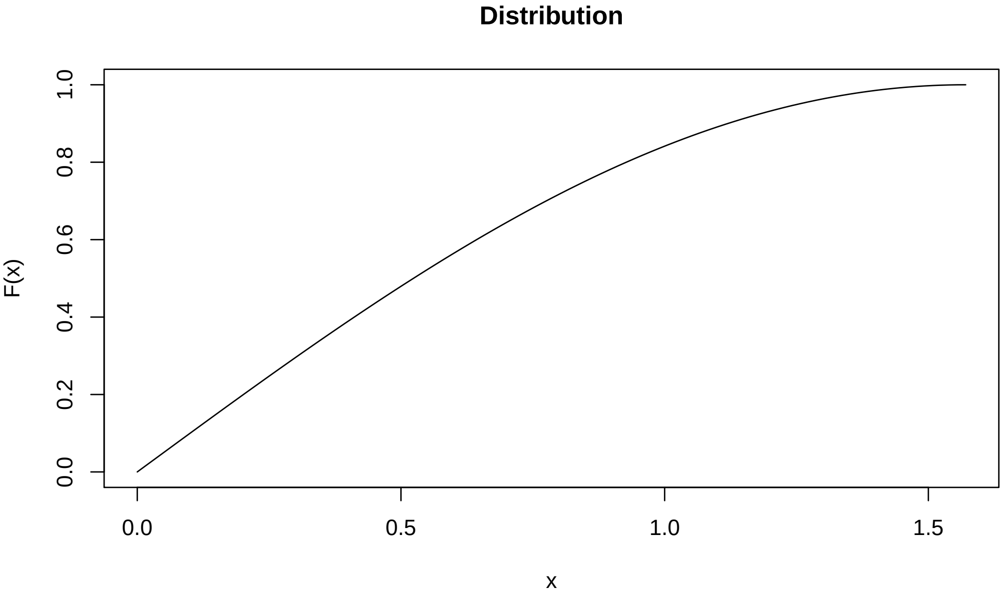
Then

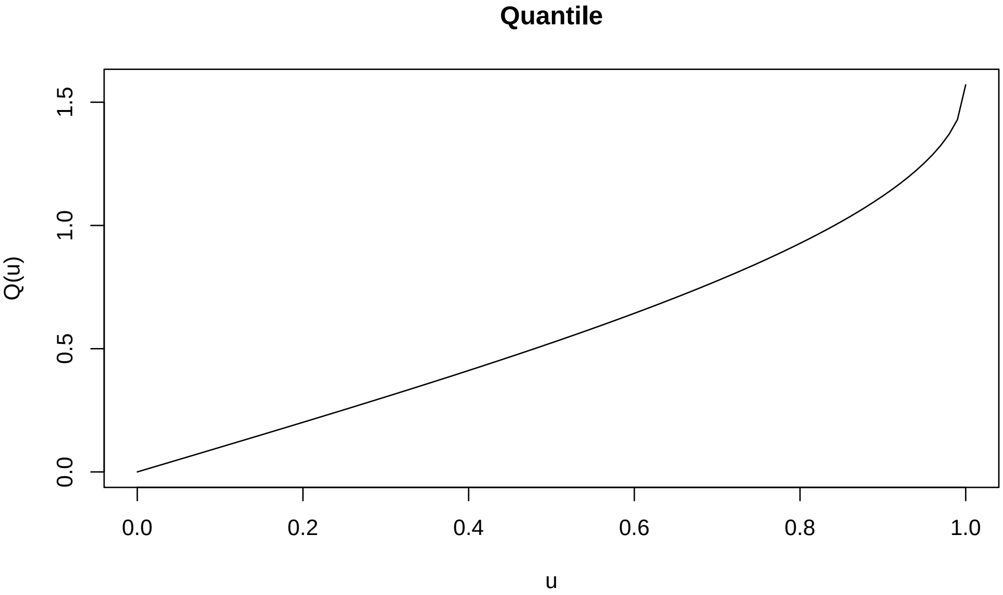
$$F_X(t) = \int_0^t \cos(x)\, dx = \sin(x)|_0^t = \sin(t)$$

And

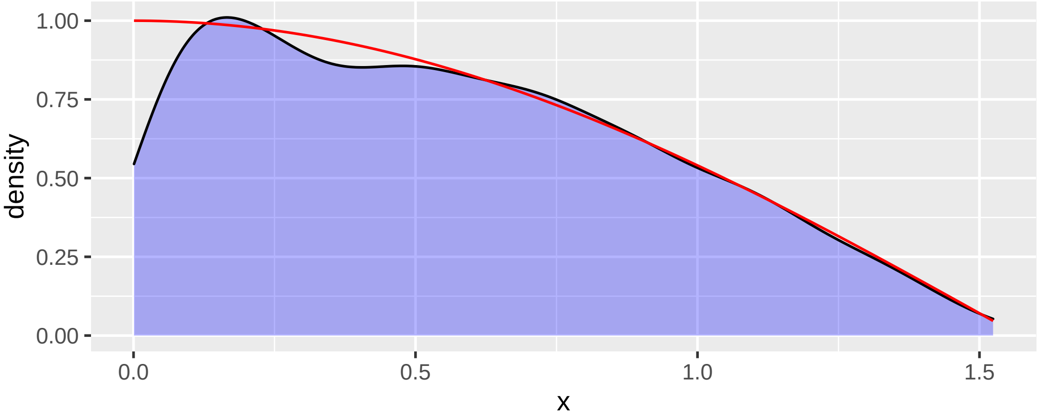
$$Q_X(p) = \sin^{-1}(t)$$


Density
f(x)
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.5
1.0
1.5
x


Distribution
F(x)
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.5
1.0
1.5
x


Quantile
Q(u)
0.0
0.5
1.0
1.5
0.0
0.2
0.4
0.6
0.8
1.0
u

Simulating from $f(x) = \cos(x)$



Example: Geometric distribution

The **geometric distribution** measure the number of Bernoulli random variables required for the first success (a special case of the negative binomial).

$$f(x) = \theta(1-\theta)^{x-1}, x = 1, 2, \ldots$$

To get the CDF for x, observe that if $X \leq x$, it must be the case that we observed **at least one** success in the first x trials. In other words, the **complement of observing zero successes**:

$$F(x) = 1 - (1-\theta)^x$$

Therefore the quantile function finds x such that

$$1 - (1-\theta)^{x-1} < u \leq 1 - (1-\theta)^x$$

Writing this in closed form

From the previous slide, we are finding x such that

$$1-(1-\theta)^{x-1} < u \leq 1-(1-\theta)^{x}$$

or equivalently

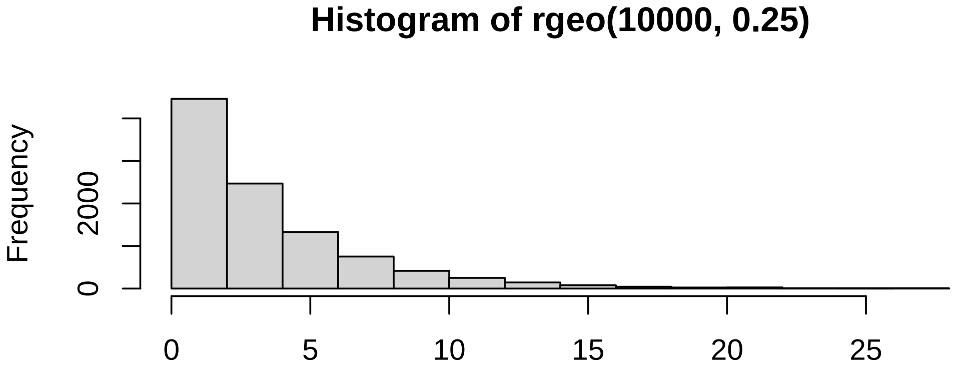
$$(1-\theta)^{x} \leq 1-u < (1-\theta)^{x-1}$$

Taking the log of each term and dividing by $\log(1-\theta)$ (which is negative), yields

$$x \geq \frac{\log(1-u)}{\log(1-\theta)} > x-1 \Rightarrow x = \left\lceil \frac{\log(1-u)}{\log(1-\theta)} \right\rceil$$

Implementing

```
> rgeo <- function(n, theta) {
+     ceiling(log(1 - runif(n)) / log(1 - theta))
+ }
```


Histogram of rgeo(10000, 0.25)
Frequency