\[ \{ x \geq \{(\frac{1}{n})_{1\leq i \leq n}\}\}\} d = \{ x \geq x\} d = (x)^n d \]

Let \( X \) denote the distribution function of \( x \). Then \( \{ n = x \} \) is defined to be the value of \( x \) such that \( \int_0^x f(t) dt = \frac{n}{X} \) for any continuous \( f \).

**Proposition**

Let \( X \) be a random variable. For any continuous \( f \), the random variable \( X \) defined by

\[ \{ n = x \} \int f(t) dt = \frac{n}{X} \]

is the distribution function \( F \) of the random variable \( X \) defined by \( n = x \).

**Algorithm**

The algorithm begins with a random variable \( X \), known to be a random variable. Finally, in section \( S \), we consider a problem of generating a random variable. In section \( S \), we consider a problem of generating a random variable. In section \( S \), we consider a problem of generating a random variable. In section \( S \), we consider a problem of generating a random variable.

Each of the algorithms generates a discrete random variable by finding

**Random Variables**

**Continuous Generators**

4. **Generating Discrete Random Variables**
\[
\begin{align*}
\{r, \varphi \in \mathbb{R} \mid \varphi^2 < r^2 \} &= \text{Max} \\
\{\varphi \in (0, \ldots, 1 \} \mid \varphi^2 < r^2 \} &= \text{Max} \\
\{\varphi \in (0, \ldots, 1 \} \mid \varphi^2 \leq r^2 \} &= \text{Max} \\
\{1 \geq \varphi \mid \varphi^2 \leq r^2 \} &= \text{Max} = N
\end{align*}
\]

Random numbers \\

\[
\begin{align*}
\{r, \varphi \in \mathbb{R} \mid \varphi^2 \leq r^2 \} &= \text{Max} \\
\{\varphi \in (0, \ldots, 1 \} \mid \varphi^2 \leq r^2 \} &= \text{Max} \\
\{\varphi \in (0, \ldots, 1 \} \mid \varphi^2 < r^2 \} &= \text{Max} = (1/N)
\end{align*}
\]

The above also provides us with another algorithm for generating

\[
\Omega \left\{ x \mid x \leq \frac{1}{\lambda} \right\} = X
\]

\[
\begin{align*}
\{r, \varphi \in \mathbb{R} \mid \varphi^2 \leq r^2 \} &= \text{Max} \\
\{\varphi \in (0, \ldots, 1 \} \mid \varphi^2 \leq r^2 \} &= \text{Max} \\
\{\varphi \in (0, \ldots, 1 \} \mid \varphi^2 < r^2 \} &= \text{Max} = (1/N)
\end{align*}
\]

Example 5b \\

If \( X \) is an exponential random variable, its distribution function is given by

\[
F_X(x) = 1 - e^{-\lambda x}
\]

The inverse transform method yields a powerful approach to generating exponential random variables, as illustrated in the next example.

\[
\begin{align*}
\lambda \cdot \ln(1 - u) &= x \\
F_X^{-1}(u) &= x = (\lambda)^{-1} \ln(1 - u)
\end{align*}
\]

Since is uniform on \([0, 1]\),

\[
\begin{align*}
\Omega &= \{(\lambda)^{-1} \ln(1 - u) \mid u \in (0, 1)\} \\
(x) &= (x)_{\lambda^{-1}}
\end{align*}
\]

\[
\begin{align*}
\{(\lambda)^{-1} \ln(1 - u) \mid u \in (0, 1)\} &= \{x \mid x > 0\} \\
\Rightarrow (\Omega, x) &= (\lambda)^{-1} \ln(1 - u)
\end{align*}
\]

Now since \( X \) is a distribution function if follows that \( P(X) \) is a monononic

\[
\begin{align*}
(n - 1) \log n &= x \\
\Rightarrow (n)^{1/n} &= \frac{x}{n - 1}
\end{align*}
\]

Therefore, from equation (5.1) we see that

\[
\begin{align*}
\Omega &= \{(x) \mid x > 0, x \in (0, 1)\} \\
(x) &= x^N
\end{align*}
\]

\section{The Inverse Transform Algorithm}

Suppose we wanted to generate a random variable \( X \) having

\[
\begin{align*}
(\lambda)^{-1} \ln(1 - u) &= x \\
F_X^{-1}(u) &= x = (\lambda)^{-1} \ln(1 - u)
\end{align*}
\]

Then the cumulative distribution function of \( X \), obtained by

\[
\begin{align*}
\Omega &= \{(\lambda)^{-1} \ln(1 - u) \mid u \in (0, 1)\} \\
(x) &= (x)_{\lambda^{-1}}
\end{align*}
\]

\[
\begin{align*}
\{(\lambda)^{-1} \ln(1 - u) \mid u \in (0, 1)\} &= \{x \mid x > 0\} \\
\Rightarrow (\Omega, x) &= (\lambda)^{-1} \ln(1 - u)
\end{align*}
\]

Now since \( X \) is a distribution function it follows that \( P(X) \) is a monononic

\[
\begin{align*}
(n - 1) \log n &= x \\
\Rightarrow (n)^{1/n} &= \frac{x}{n - 1}
\end{align*}
\]

Therefore, from equation (5.1) we see that

\[
\begin{align*}
\Omega &= \{(x) \mid x > 0, x \in (0, 1)\} \\
(x) &= x^N
\end{align*}
\]
function: \[ f(x) = \begin{cases} 0 & \text{if } x < 0 \\ \lambda e^{-\lambda x} & \text{if } x \geq 0 \end{cases} \]

Figure 5.1: The reflection method for simulating a random variable X having density

Step 1: Generate a random number D.
Step 2: Compute N = LN(D).
Step 3: Compute X = -\ln(D) / \lambda.

The Reflection Method

A random variable having density \( f(x) \) for \( x \geq 0 \) is defined by

\[ P(Y \leq x) = \begin{cases} 0 & \text{if } x < 0 \\ \frac{\lambda e^{-\lambda x}}{\lambda} & \text{if } x \geq 0 \end{cases} \]

Step 1: Let E = \( \mathbb{E}(Y) \).
Step 2: Compute X = E - Y.
Step 3: Compute the density of X.

Suppose we have a method for generating a random variable having density

\[ f(x) = \begin{cases} \lambda e^{-\lambda x} & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \]

Then we can use this method to generate random numbers from this distribution by reflecting around the point \( x = 0 \).

For all \( c \geq 0 \),

\[ \frac{\lambda e^{-\lambda x}}{\lambda} = \begin{cases} 0 & \text{if } x < 0 \\ \lambda e^{-\lambda x} & \text{if } x \geq 0 \end{cases} \]

Specifically, let \( Y \) be a constant such that

\[ \frac{\lambda e^{-\lambda Y}}{\lambda} = c \]

Specifically, let \( Y \) be a constant such that

\[ \frac{\lambda e^{-\lambda Y}}{\lambda} = c \]

Then we can use this method to generate random numbers from this distribution by reflecting around the point \( x = 0 \).

Suppose we have a method for generating a random variable having density

\[ f(x) = \begin{cases} \lambda e^{-\lambda x} & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \]

Then we can use this method to generate random numbers from this distribution by reflecting around the point \( x = 0 \).

For all \( c \geq 0 \),

\[ \frac{\lambda e^{-\lambda x}}{\lambda} = \begin{cases} 0 & \text{if } x < 0 \\ \lambda e^{-\lambda x} & \text{if } x \geq 0 \end{cases} \]

Example 5.2: Generating Continuous Random Variables

Suppose we want to simulate the value of a gamma \( \Gamma(n, \theta) \) distribution with shape parameter \( n \) and scale parameter \( \theta \). We can use the following algorithm:

1. Generate \( U \) uniformly distributed on \( (0, 1) \).
2. Compute \( V = \Gamma(n, \theta) \).
3. Compute \( X = U - \ln(1 - U) / \theta \).

The results of Example 5.2 show the relationship between the gamma and

\[ X \sim \Gamma(n, \theta) \]

Example 5.2: Generating Continuous Random Variables

Suppose we want to simulate the value of a gamma \( \Gamma(n, \theta) \) distribution with shape parameter \( n \) and scale parameter \( \theta \). We can use the following algorithm:

1. Generate \( U \) uniformly distributed on \( (0, 1) \).
2. Compute \( V = \Gamma(n, \theta) \).
3. Compute \( X = U - \ln(1 - U) / \theta \).

The results of Example 5.2 show the relationship between the gamma and
example 55

Let us use the rejection method to generate a random variable 

\[ X \sim \text{Gamma}(\alpha, \beta) \] 

and then accept \( X \) with probability \( \frac{\beta X^{\alpha-1}}{\alpha \Gamma(\alpha)} \).

As in the discrete case, we should be careful that the way in which one accepts

\( X \) if \( \frac{\beta X^{\alpha-1}}{\alpha \Gamma(\alpha)} \geq U \) and \( X \) if \( \frac{\beta X^{\alpha-1}}{\alpha \Gamma(\alpha)} < U \),

\( U \) being a uniform random variable.

(ii) The number of iterations of the algorithm that are needed is a geometric

\[ (1 - \frac{\beta}{\alpha \Gamma(\alpha)}) \] 

The reader should note that this rejection method is exactly the same as in the

discussion before.

\[ (x - 1)^x = \frac{x! \delta x}{952} \]

Setting this equal to zero the maximand value is attained when \( x = \frac{5}{4} \).

\[ \frac{d \hat{V}}{dx} = \frac{\delta x}{x^2} \]

There is where \( \delta x = \frac{\delta}{x} \). Hence

\[ \frac{d \hat{V}}{dx} = \frac{\delta}{x} \] 

The theorem

we can prove the following result

\[ \frac{d \hat{V}}{dx} = \frac{\delta}{x} \] 

Therefore, the probability distribution of the \( \delta X \) is needed.

\[ \text{Gamma}(1, \beta) \]

\[ \frac{d}{dx} \left( \frac{x^{\alpha-1} \beta}{\alpha \Gamma(\alpha)} \right) = \frac{(\alpha - 1) \beta x^{\alpha-2}}{\alpha \Gamma(\alpha)} \]
We start by generating from the exponential density function by using the reflection method with its probability density function

\[ f(x) = \frac{e^{-x}}{\beta} \quad (x > 0) \]

To obtain the value of a random variable having the exponential density function, we differ.

\[ s(\gamma - \delta)^2 \left( 1 - e^{-x} \right) = s(\gamma - \delta)^2 \left( e^{-x} - 1 + x \right) = \left( \frac{y}{x} \right) \frac{\gamma}{\delta} \]

We see that when \( y \leq \gamma \), we have

\[ \left( \frac{y}{x} \right) \frac{\gamma}{\delta} \]

The average number of iterations that will be needed is

\[ \sqrt{\frac{\pi}{2\delta}} \approx 1.257 \]

The expected number of iterations that will be needed is

\[ \sqrt{\frac{\pi}{2\delta}} \approx 1.257 \]
where the preceding integral was evaluated by integration by parts, because
\[ \int x^{- \alpha} \ln x \, dx = (-1)^{\alpha} \frac{1}{\alpha^2} \ln x - x^{- \alpha} + C. \]

Valuable having density function

Example 5.9 Suppose we want to generate a geometric (\(1\)) random variable.

The rejection method is particularly useful when we need to simulate a random variable conditioned on the value of another random variable. This is illustrated by the next example.

Example 6.3

The rejection method is particularly useful when we need to simulate a random variable conditioned on the value of another random variable. This is illustrated by the next example.

\[ \text{Step 1: Generate an exponential random variable } Z \sim \text{Exp}(\lambda) \text{ with density } f(x) = \lambda e^{-\lambda x}. \]

\[ \text{Step 2: Generate a standard normal random variable } Y \sim \mathcal{N}(0,1). \]

\[ \text{Step 3: If } Z < Y, \text{ accept } (Z, Z - Y), \text{ otherwise reject.} \]

\[ \text{Step 4: Accept only those pairs } (Z, Z - Y) \text{ such that } Z < 1. \]

\[ \text{Step 5: Calculate } \frac{Z}{Z - Y}. \]

\[ \text{Remark: If } X \text{ denotes the number of trials until the first success, then } X \sim \text{Geom}(\theta) \text{ with probability mass function } p_X(k) = \theta(1 - \theta)^k. \]

\[ \text{Step 1:} \text{ Generate a Poisson random variable } N \sim \text{Poi}(\lambda). \]

\[ \text{Step 2:} \text{ Generate a standard normal random variable } Y \sim \mathcal{N}(0,1). \]

\[ \text{Step 3:} \text{ If } Y < \frac{1}{2}\sqrt{2\pi} \ln(1 - \lambda), \text{ accept } (Y, N), \text{ otherwise reject.} \]

\[ \text{Step 4:} \text{ Accept only those pairs } (Y, N) \text{ such that } Y < \frac{1}{2}\sqrt{2\pi} \ln(1 - \lambda). \]

\[ \text{Step 5: Calculate } \frac{Y}{\frac{1}{2}\sqrt{2\pi} \ln(1 - \lambda)}. \]
and being uniformly distributed over (0, \infty) and \( z \) are independent with \( X \) having exponential with mean \( \mu \).

Now, if (z, z') \sim \text{Exp}(\theta) \times \text{Exp}(\mu) \),

\[
\mathbb{P}(X = j) = \frac{\lambda^j e^{-\theta \lambda}}{j!}, j = 0, 1, 2, \ldots
\]

and the probability density is

\[
\frac{\lambda^j e^{-\theta \lambda}}{j!}
\]

However, as this is equal to the product of an exponential density having mean \( \frac{\lambda}{\mu} \)

\[
\frac{e^{-\frac{\theta}{\mu} z}}{\frac{\theta}{\mu}} = (\frac{\theta}{\mu})^z
\]

Therefore, this is the product of an exponential density having mean \( \frac{\lambda}{\mu} \).

Let \( X \) and \( Y \) be independent standard normal random variables and let \( \theta \) and \( \mu \) be constants.

\[
x + \theta Y = \theta X + Y
\]

Where the polar coordinates of the vector \((X, Y)\) are defined (see Figure 5.3).

The polar method for generating normal random variables

Figure 5.2: Polar Coordinates

5.3 The Polar Method for Generating Normal Random Variables

Illustrated in Section 5.8, the derivation of the polar method based on an exponential function and the polar method used for the exponential function is illustrated in Section 5.8. However, the exponential distribution is the product of an exponential density having mean \( \frac{\lambda}{\mu} \).

To determine the joint density of \( X \) and \( Y \), we make the change

\[
\begin{align*}
\frac{X}{\lambda} &= \theta, \\
\frac{Y}{\mu} &= \mu
\end{align*}
\]

Then \( X + Y = \theta X + \theta Y = \theta X + Y \).  

Similarly, the exponential function is

\[
\frac{X}{\lambda} = \theta, \quad \frac{Y}{\mu} = \mu
\]

Since \( X \) and \( Y \) are independent, their joint density is the product of their individual densities.

The diagram shows the joint density of \( X \) and \( Y \), where \( X \) and \( Y \) are independent.

\[
\begin{align*}
\frac{X}{\lambda} &= \theta, \\
\frac{Y}{\mu} &= \mu
\end{align*}
\]

Because this is a decreasing function of \( x \) when \( x \leq \theta \), it follows that

\[
\begin{align*}
\frac{X}{\lambda} &= \theta, \\
\frac{Y}{\mu} &= \mu
\end{align*}
\]

This is the polar method based on an exponential function and mean \( \frac{\lambda}{\mu} \) is conditioned to be at least 5, 5.

\[
\begin{align*}
\frac{X}{\lambda} &= \theta, \\
\frac{Y}{\mu} &= \mu
\end{align*}
\]
5.2.2 Generating Normally Distributed Random Numbers

The Polar Method for Generating Normal Random Variables

\[ \frac{X}{Y} = \frac{\frac{1}{2}(\bar{X} + \|A\|)}{\|A\|} = \frac{1}{\|A\|} \cos \theta \]

\[ \frac{Y}{X} = \frac{\frac{1}{2}(\bar{Y} + \|A\|)}{\|A\|} = \frac{1}{\|A\|} \sin \theta \]

The polar method is used to generate normally distributed random variables. This method involves generating two random numbers, \(X\) and \(Y\), and using them to calculate the values of \(\bar{X}\) and \(\bar{Y}\). These values are then used to generate a normally distributed random variable. The method is as follows:

\[ X = \bar{X} + \|A\| \cos \theta \]

\[ Y = \bar{Y} + \|A\| \sin \theta \]

where \(\|A\|\) is the magnitude of the random variable and \(\theta\) is a uniformly distributed random variable over \([0, 2\pi)\).
5.5 Generating a Nonhomogeneous Poisson Process

\[ \lambda(t) = \begin{cases} \lambda_0 & \text{if } 0 < t < T_1 \\ \lambda_1 & \text{if } T_1 < t < T_2 \\ \vdots & \text{etc.} \end{cases} \]

where \( \lambda(t) \) is the instantaneous rate of occurrence at time \( t \), and \( T_1, T_2, \ldots \) are the points in time at which \( \lambda(t) \) changes.

If we want to generate the set of arrival times of a nonhomogeneous Poisson process, we can use the following algorithm:

1. Choose a random number \( U \)
2. If \( U < \lambda_0 \), set \( T_1 = 1 \)
3. If \( \lambda_0 \leq U < \lambda_0 + \lambda_1 \), set \( T_1 = 1, T_2 \) is the first arrival time
4. Repeat the above steps for each change in \( \lambda(t) \)

The above algorithm will generate the arrival times of a nonhomogeneous Poisson process.
Theorem 1. Consider the case where the random variables for a Poisson distribution with mean $\lambda$ are independent. Let $Y_1, Y_2, \ldots, Y_n$ be the $n$ independent random variables. Then, the sum $Y = Y_1 + Y_2 + \cdots + Y_n$ is also a Poisson random variable with mean $\lambda$.

Proof: Let $Y_i$ be the number of events occurring in the $i$th interval. Then, $Y_i$ is a Poisson random variable with mean $\lambda_i$. Since the intervals are independent, the sum $Y = Y_1 + Y_2 + \cdots + Y_n$ is a Poisson random variable with mean $\lambda = \lambda_1 + \lambda_2 + \cdots + \lambda_n$.

Corollary: If $Y_1, Y_2, \ldots, Y_n$ are independent Poisson random variables with means $\lambda_1, \lambda_2, \ldots, \lambda_n$, respectively, then $Y_1 + Y_2 + \cdots + Y_n$ is a Poisson random variable with mean $\lambda_1 + \lambda_2 + \cdots + \lambda_n$.

Theorem 2. Let $Y_1, Y_2, \ldots, Y_n$ be independent Poisson random variables with means $\lambda_1, \lambda_2, \ldots, \lambda_n$, respectively. Then, the negative binomial distribution with parameters $r$ and $p$, denoted by $\text{NB}(r, p)$, is also a Poisson distribution with mean $\lambda_1 + \lambda_2 + \cdots + \lambda_n$.

Proof: The negative binomial distribution is a discrete probability distribution that gives the number of successes in a sequence of independent and identically distributed Bernoulli trials before a fixed number of failures occurs. If $Y_1, Y_2, \ldots, Y_n$ are independent Poisson random variables with means $\lambda_1, \lambda_2, \ldots, \lambda_n$, respectively, then the sum $Y = Y_1 + Y_2 + \cdots + Y_n$ is a Poisson random variable with mean $\lambda_1 + \lambda_2 + \cdots + \lambda_n$. Therefore, the negative binomial distribution with parameters $r$ and $p$, denoted by $\text{NB}(r, p)$, is also a Poisson distribution with mean $\lambda_1 + \lambda_2 + \cdots + \lambda_n$.
Exercise 1

\[ 0 < n \leq x \leq \frac{v + s}{v - s} \]
\[ 0 < x \leq 1 \]

Give a method to generate a random variable having density function

\[ f(x) = (1 - x)^s \]

and in general

\[ \frac{n - 1}{n + 1)^s} = \frac{n - 1}{n - 1} (v + 1)^s + 1 = s \]

We can therefore generate the successive terms of the distribution

\[ \frac{n - 1}{n + 1)^s} = \frac{n - 1}{n - 1} (v + 1)^s + 1 = s \]

and then recursively setting

\[ n \geq f \]

\[ \frac{n - 1}{n + 1)^s} = \frac{n - 1}{n - 1} (v + 1)^s + 1 = s \]

Thus, we have

\[ (v + s)^n = n \]

or equivalently

\[ \frac{v + s + x}{x} = (x)^n \]

The inverse distribution function of \( y \), which is the expected number of trials until the next event, is given by

\[ (u + s)^{v + s + x} \]

Hence, from Equation (5.8),

\[ (\frac{p + s}{v + s + x}) \]

\[ = \frac{u + s}{u + s + x} = (x)^n \]

Example 5

Suppose that \( Y \) is a gamma variate with \( \alpha = 1 \) and \( \beta = (n + 1)/n \) for some positive integer \( n \). Then

\[ f(x) = \alpha x^{\alpha - 1} e^{-\beta x} \]

and so the inverse transformation method can be applied to generate the distribution of \( Y \) in the following example. The distribution of \( Y \) is well-known and hence we do not need to simulate from these distributions or use any special technique to generate the values of \( Y \) from the distribution. Of course, one needs to generate a value of \( Y \) from the distribution for which the simulated value of \( Y \) is well-known and then recursively set up the successive terms of the distribution. We now simulate the event that \( Y \), which is generated from the distribution, is

\[ (\frac{p + s}{v + s + x}) \]

\[ = \frac{u + s}{u + s + x} = (x)^n \]

\[ \int_{x}^{\infty} f(y) dy = 1 \]

\[ \int_{x}^{\infty} \frac{u + s}{u + s + x} \]

\[ \int_{x}^{\infty} \frac{u + s}{u + s + x} \]

Therefore, for some positive integer \( n \), the expected number of trials until the next event is given by

\[ X = (x)^n \]

To invert this, suppose that \( (v + s)^n = n \)

\[ \frac{v + s + x}{x} = (x)^n \]

Hence, from Equation (5.8),

\[ (\frac{p + s}{v + s + x}) \]

\[ = \frac{u + s}{u + s + x} = (x)^n \]

\[ \int_{x}^{\infty} f(y) dy = 1 \]

\[ \int_{x}^{\infty} \frac{u + s}{u + s + x} \]

\[ \int_{x}^{\infty} \frac{u + s}{u + s + x} \]

Therefore, for some positive integer \( n \), the expected number of trials until the next event is given by

\[ X = (x)^n \]
Given the information, it follows that if \( X \) is the conditional distribution of \( Y \mid Y \), then \( X \) is the conditional distribution of \( Y \mid Y \).

\[
q \leq x \leq p \\
\frac{f(y) - f(x)}{f(y) - f(p)} = f(x)
\]

Thus, the result of Exercise 7 follows from the following random number generator:

where \( \alpha \), \( \beta \), \( \gamma \), and \( \delta \) are independent uniform random variables whose sum is 1.

\[
(x_1' | y) = \begin{cases}
1 & \text{if } x_1' < y \\
0 & \text{if } x_1' \geq y
\end{cases}
\]

Using the result of Exercise 7, give algorithms for generating random variables.

From the result of Exercise 7, we can define a random variable having distribution function:

\[
(x) \quad \text{if } x \in [0, 1]
\]

where \( \eta \) is a constant.

\[
(x) = \begin{cases}
1 & \text{if } x \in [0, 1] \\
0 & \text{if } x \notin [0, 1]
\end{cases}
\]

Discuss the efficiency of the above procedure in generating \( F \).

1. Use the rejection method and the results of Exercise 12 to generate two random variables:

\[
(x) \quad \text{if } x \in [0, 1]
\]

where \( \eta \) is a constant.

2. Give a method for generating a random variable having density function:

\[
(x) = \begin{cases}
1 & \text{if } x \in [0, 1] \\
0 & \text{if } x \notin [0, 1]
\end{cases}
\]

4. Give a method for generating a random variable having density function:

\[
(x) = \begin{cases}
1 & \text{if } x \in [0, 1] \\
0 & \text{if } x \notin [0, 1]
\end{cases}
\]

5. Give a method for generating a random variable having density function:

\[
(x) = \begin{cases}
1 & \text{if } x \in [0, 1] \\
0 & \text{if } x \notin [0, 1]
\end{cases}
\]

6. Give a method for generating a random variable having density function:

\[
(x) = \begin{cases}
1 & \text{if } x \in [0, 1] \\
0 & \text{if } x \notin [0, 1]
\end{cases}
\]

7. Generate 1000 random numbers and use them to estimate the probability:

\[
\text{probability that } x > 0.5 \text{ given } x > 0
\]

8. Use the inverse transformation method to generate a random variable having distribution function:

\[
(x) = \begin{cases}
1 & \text{if } x \in [0, 1] \\
0 & \text{if } x \notin [0, 1]
\end{cases}
\]
References


