Random numbers and random events

Mathcad has a built in function rnd(x) that selects a number randomly on the interval 0 to x. If x = 1, this means that 10% of the numbers will be less than 0.1 and 10% will be between 0.9 and 1.0. We can check this by making a list of 1000 of numbers created by rnd(1) and then plotting the distribution.

The number of random numbers will be: $n := 1000$

$$Z := \left| \begin{array}{l} \text{for } i \in 1 .. n \\ \quad Z_i \leftarrow \text{rnd}(1) \\ Z \end{array} \right.$$

This programming loop will find 1000 values and put them in the vector Z which will have 1001 elements. We will not use the first element Z_0.

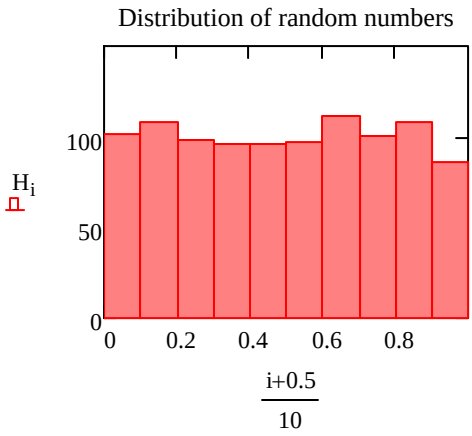
Now let's plot the distribution as a histogram. First we divide the domain (0,1) into 10 intervals. The vector containing the intervals will have 11 values: 0,0.1... 1.0.

$i := 0 .. 10$ $\quad$ $\text{intervals}_i := 0.1 \cdot i$

$H := \text{hist}(\text{intervals}, Z)$

The **hist** function counts the number of entries in each interval and returns a vector with one fewer elements than interval values. This vector, H, contains the data for the histogram plot below.

$i := 0 .. 9$



This is a 2-d plot with "solidbar" selected as the line type using the graph dialog box. The horizontal axis has 0.5 added to i so that the bars are plotted in the middle of the interval, not at the beginning of the interval. This means that the bar for values 0 to 0.1 is centered on 0.05.

Inspection shows that indeed about 10% of the 1000 random numbers fall in each interval.

We expect n/10 = 100 numbers counted in each bin.
What is the standard deviation of the number in the bins (call it Sdev)?

$$Sdev := \sqrt{\frac{1}{9}\cdot\sum_i \left(H_i - \frac{n}{10}\right)^2}$$

Sdev = 7.38617

H_i =

101
108
98
96
96
97
111
100
108
86

i was last defined as going from 0 to 9 so there are 10 bins. The variation from one sample to the next is about ±7. Note that the standard deviation is defined so that the division is by one less than the number of bins.

Try it: Change n from 1000 to 100,000 and watch what happens to the standard deviation. Is it a larger or smaller percentage of the H vaues?

Radioactive decay

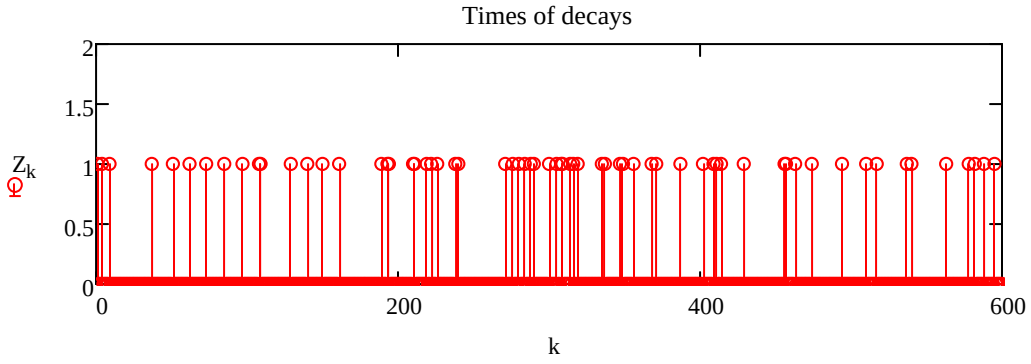
Suppose a radioisotope generates, on average, one count about every 10 seconds. That means there is a 10% probability each second of there being a decay. How can we model that? Let's select a random number using rnd(1). Then there is a 10% chance of the number being below 0.1. We will say that there has been a decay if rnd(1) is less than 0.1 and no decay if it is not. We will keep a record of decays in the following way: We will set the elements of a vector Z initially to zeroes and then change the zero to 1 if there is a decay. For this example, "1" means true and "0" means false (no decay). If there is a decay in the first second, Z_1 is 1, otherwise it is zero. The counter n below counts the seconds.

```
Z := | for i ∈ 1 .. n                    n = 1000
     |   Z_i ← 0
     | for i ∈ 1 .. n
     |   Z_i ← if(rnd(1) < 0.1, 1, 0)
     | Z
```

Now let's plot Z k := 1 .. 600

The 2-d plot below has "stem" selected as the trace type which puts circles on stalks. The vertical scale is set to 2. The stems are randomly spaced. The clustering at certain places and gaps at other places are random!

We plot only the first 600 seconds containing about 60 randomly spaced decays.



The points indicate the times when the "clicks" were heard on the geiger counter.

We can find the number of decays by simply summing the "1" values in the vector Z:

$k := 1 .. 1000$ $\sum_k Z_k = 106$

Indeed there are about 100 decays in 1000 time intervals if the probability is 10% for each interval.
Try it: If you select Calculate Worksheet form the Math menu, the calculation will be done again with a new set of random numbers. Does this change by much the number of decays?

Selecting random time intervals

Suppose you want to find the time interval between one decay and the next.
How would that be done?
Suppose the decay time is $\tau = 1$. If we select a random number on the domain (0,1), the function -τ ln[rnd(1)] converts the random number to a time interval. Since the log of rnd(1) will be less than one we put a minus sign in front of the logarithm to create a positive number from the negative logarithm. If rnd(1) is 0.001 then the logarithm is -6.9 which becomes +6.9. Thus long time intervals (t >> τ) are very unlikely because they require a very small value for rnd(1).

Try it: Show that the probability of 6.9 decay times passing without a decay is 0.001.

On the other hand, if rnd(1) is 0.9, the time interval generated is +0.1. Thus intervals between 0 and 0.1 are generated with 10% probability. So this function seems to be what we need.
Let's check that.

$\tau := 10$ This is the decay time corresponding to 0.1 probability per second.

$n = 1000$ Number of decays.

$Z :=$ | $Z_n \leftarrow 0$ | Initialize the last Z to initialize all Z values and to reserve memory space for all of Z.
| for $i \in 0 .. n$
| $Z_i \leftarrow -\tau \ln(\mathrm{rnd}(1))$
| Z

Z now contains the time intervals between decays. We have to sum the intervals to get the time elapsed since t = 0.

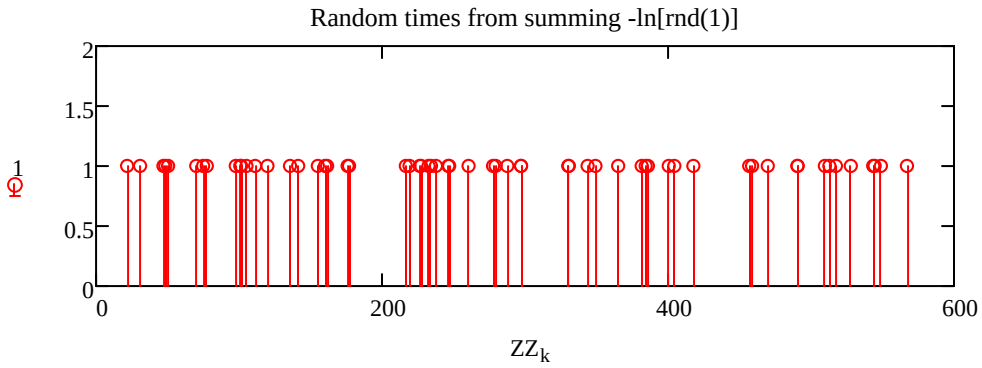
$k := 1 .. 60$

$ZZ_k := \sum_{m=0}^{k} Z_m$ ZZ contains the times of the decays.

Intervals between decays:

$Z =$

	0
0	5.415
1	16.35
2	9.168
3	16.253
4	1.353
5	1.922
6	19.648
7	4.798
8	2.639
9	20.203
10	3.17
11	0.609
12	3.701
13	6.08
14	8.635
15	...

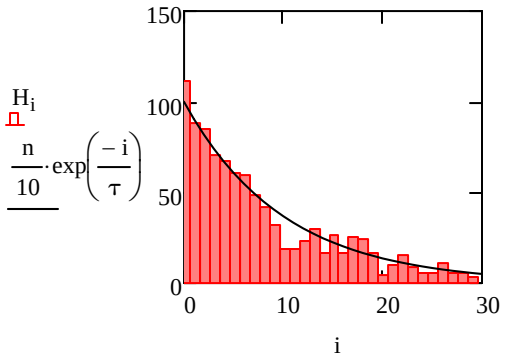


$i := 0 .. 30$

$intervals_i := i$

$H := hist(intervals, Z)$

As in our previous graph, some of the intervals are quite long and others are short. Let's plot a histogram of the lengths of the intervals. The widths of the bars in the plot will be 1 sec.

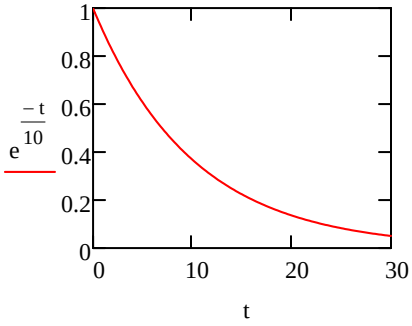


The intervals have a distribution which is exponential as exp(-t/τ). Thus our function -τ ln[rnd(1)] gives the correct "mapping" of random numbers to decay time intervals.

Simulating the decay of a population

$t := 0, 0.2 .. 30$

If the probability is 0.1 per second, the probability that an atom has not decayed is given by exp(-t/10), shown at right.



We will follow 1000 atoms for 30 seconds.
The decay time τ = 10 seconds.
Each 1 second, the probability of decay is 0.1.

$imax := 999$ $jmax := 30$

j counts seconds, so the time interval is $\Delta t := 1$

$$P := \left| \begin{array}{l} \text{for } i \in 0 .. imax \\ \quad \text{for } j \in 0 .. jmax \\ \quad\quad P_{i,j} \leftarrow 1 \\ \text{for } i \in 0 .. imax \\ \quad \text{for } j \in 1 .. jmax \\ \quad\quad P_{i,j} \leftarrow \text{if}\left(\text{rnd}(1) < \frac{\Delta t}{\tau}, 0, P_{i,j-1}\right) \\ P \end{array} \right.$$

Initialize P values to 1 using nested loops. Each row of P represents a nucleus. The row will have value 1 if the nucleus has not decayed and value 0 if it has decayed.

The "if" statement sets $P_{i,j}$ = 0 if there is a decay, otherwise $P_{i,j}$ stays at the previous value of 0 or 1. The decay probability is 0.1.

j is the time step number and i is the particle number.

In the matrix P, each particle is one row. If the particle decays on the 10th time step, the value changes from 1 to 0 and stays zero. Time increases to the right.

P =

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
3	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0
5	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
8	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	...

The first column of P is all ones because none of the particles have decayed at t = 0.

We can sum the columns to see how many particles are still left after each time interval.

There are imax+1 particles to start: $\sum_{i=0}^{imax} P_{i,0} = 1000$

$i := 0 .. jmax \qquad j := 0 .. jmax$

After j time intervals, the number remaining is $F_j := \sum_{i=0}^{imax} P_{i,j}$

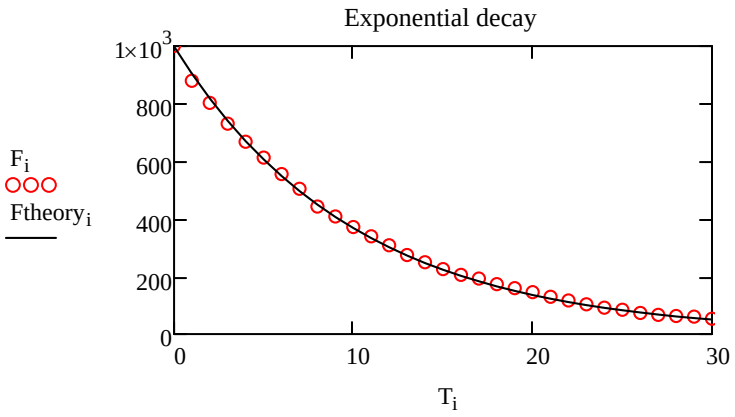
We expect the answer to be exponential

$T_i := i \qquad Ftheory_i := imax \cdot e^{\frac{-T_i}{\tau}}$

Number remaining:

$F =$

	0
0	1000
1	879
2	802
3	730
4	667
5	612
6	555
7	504
8	442
9	408
10	371
11	339
12	308
13	274
14	249
15	...



Selecting randomly from a population

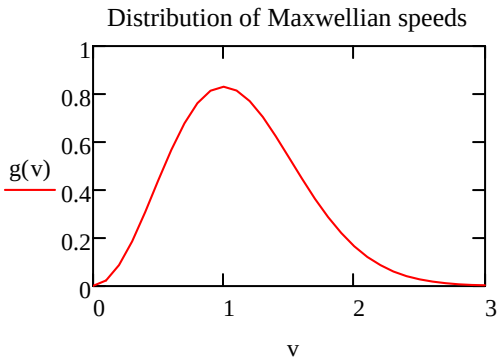
A group of gas atoms with a Maxwellian velocity distribution in three dimensions has a speed distribution given by

First define a thermal velocity: $v_t := 1$

The Maxwellian: $g(v) := \frac{4 \cdot \pi \cdot v^2}{\left(\sqrt{\pi} \cdot v_t\right)^3} \cdot e^{\frac{-v^2}{v_t^2}}$

A set of values for plotting: $v := 0, 0.1 .. 3$

How would you select speeds randomly from this distribution? Place a point X,Y randomly on the graph and accept it if it the "Y" value under the curve and reject it if it is above the curve. At each value of v, the height of the distribution is proportional to the probability. So we choose a guess for v randomly on the interval 0,3. Then the probability of accepting the guess is proportional to height of g(v). This is called the "rejection method" because we will throw away the guesses at points above the curve.



$n := 1000$ Select 1000 speeds from the Maxwellian.

```
Z :=  | Z_n ← 0
      | for i ∈ 0 .. n
      |   | y ← rnd(1)
      |   | v ← rnd(3)
      |   | while y > g(v)
      |   |   | v ← rnd(3)
      |   |   | y ← rnd(1)
      |   | Z_i ← v
      | Z
```

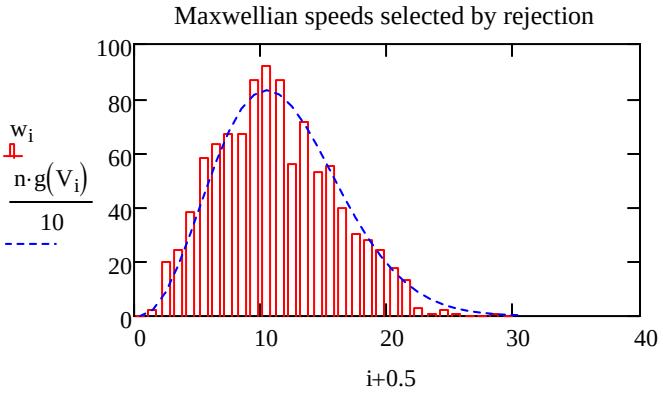
Initialize Z and reserve memory space.
y is a random y-axis value on the graph above.
v is a random guess for v. The "while" loop continues to make guesses for y and v. The probability that v is kept is determined by the height of the curve. If the random number y is below the height of the curve, the guess for v is kept, otherwise it is rejected and a new value is tried. Let's make a histogram of our random guesses:

Prepare a histogram:

$i := 0 .. 30$

$intervals_i := 0.1 \cdot i$ $V_i := \frac{i}{10}$

$w := hist(intervals, Z)$



In the chart dialog box, under traces, click on solid bar to make a histogram plot. Use i + 0.5 on the x axis to centers the bars.

Try it: Is the randomly chosen sample more like the theory curve if n is increased to 10,000? to 100,000?

Note: Our guess for y was on the interval 0,1 because the maximum value of g(v) is a little less than one. If the maximum of g(v) were higher than 1, we would have to increase the maximum allowed value of the random guess.