# Problem Set 6

JIAO Hao

## PROBLEM 1

We can find from fig.1 that the 'random numbers (vectors)' producted by C standard library are lying on some planes and they are very much not randomly distributed in 3D space.

But I got more planes than professor: at least 35 horizontal plane (shown in "6-1-7.png"), and there also are planes with other derection (three direction should be the same).

I didn't successfully run the library of python ("JIAO Hao-6-1-1"). But I think the ramdom number generated by python is good. I show 10000 and 100000 ramdom points generated by "1e8*np.random.rand(n)" in the same volume as above, and we can see that they are really random ("JIAO Hao-6-1-1", and fig.2).

## PROBLEM 2

**Which of Lorentzians, Gaussians, and power laws could you use for the bounding distribution?**

If the distribution is bounded in $[0, a]$, we can use all three distribution to get the exponential deviates. But if we consider the distribution at $[0, \infty]$, we can only use Lorentzians and power laws:

$$\lim_{x\to\infty} \frac{\exp[x/\tau]}{\tau} / \frac{1}{\pi(1+x^2)} = 0$$

$$\lim_{x\to\infty} \frac{\exp[x/\tau]}{\tau} / \frac{\exp[-x^2/2\sigma^2]}{\sqrt{2\pi\sigma}} = \infty$$

$$\lim_{x\to\infty} \frac{\exp[x/\tau]}{\tau} / (\alpha+1)x^\alpha = 0$$

I use both Lorentzians and power laws:

Form Lorentzian:

The accept rate from Lorentz to Exponential deviates is 6365045 / 9366561 = 0.6795498369145303

Transformation method use 0.22466683387756348 s.

The rejection method use 1.4030277729034424 s.

Form power laws:

The accept rate from PowerLaw to Exponential deviates is 7034369 / 9500645 = 0.7404096248202096

Transformation method use 0.25522756576538086 s.

The rejection method use 1.9982857704162598 s.

**The histogram of my deviates matches up with the expected exponential curve**

Shown in fig.3. We can see that both the two distribution matched the expected exponential curve very well.

**How efficient can you make this generator, in terms of the fraction of uniform deviates that give rise to an exponential deviate?**

The accept rate from Lorentz to Exponential deviates is 6365045 / 9366561 = 0.6795498369145303

The accept rate from PowerLaw to Exponential deviates is 7034369 / 9500645 = 0.7404096248202096

The accept rate from Uniform to Exponential deviates is 1001223 / 10000000 = 0.1001223

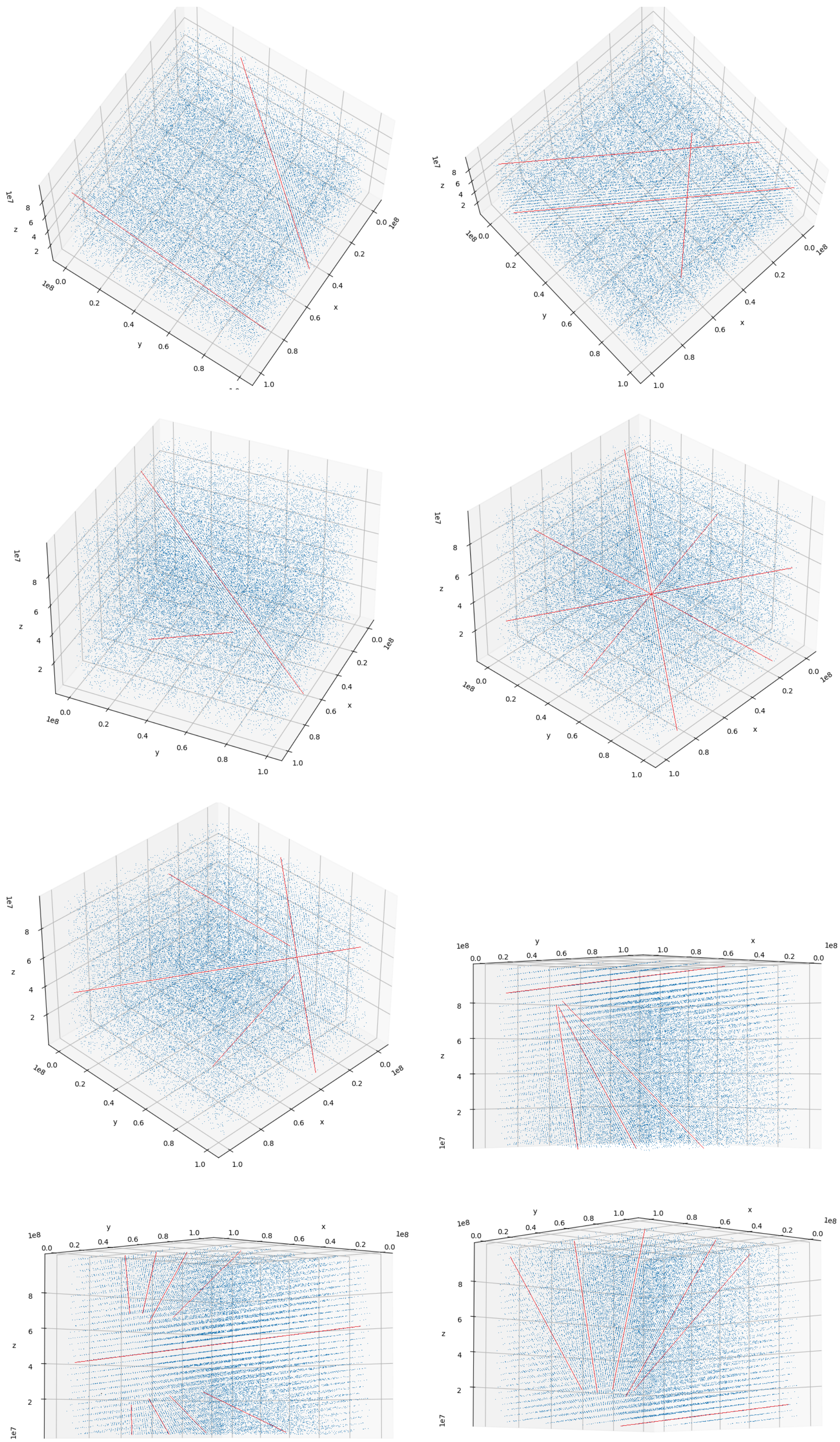


FIG. 1. Show that these 'random points' by the random number generator in the C standard library are in some planes. I denotes these planes by red lines. If these figures are not clear enough, you can see them in the files named '6-1-1.png' to '6-1-8.png'.

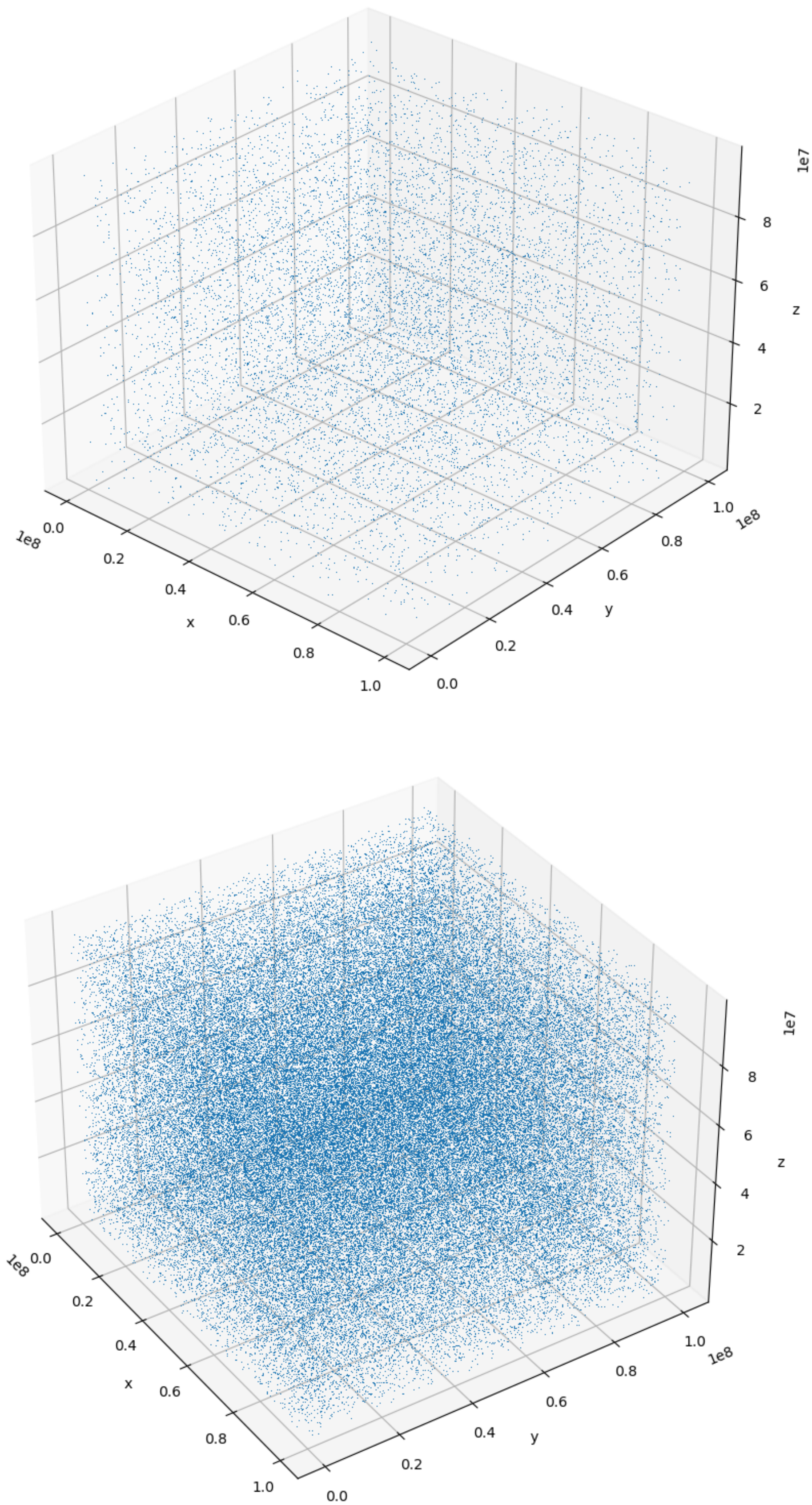


FIG. 2. Random points generated by python random number generator.

## PROBLEM 3

Since the random number with exponential distribution is alway positive, we shuold keep $v/u > 0 \quad \Rightarrow \quad v \in [0, 1]$ rather than $v \in [-1, 1]$.

The result is shown in fig.4. From this, we can see that this generator also produces the correct answer.

And we can get that the efficiency of this genertor is acceptable:

The accept rate of ratio-of-uniforms generator is 0.5002858

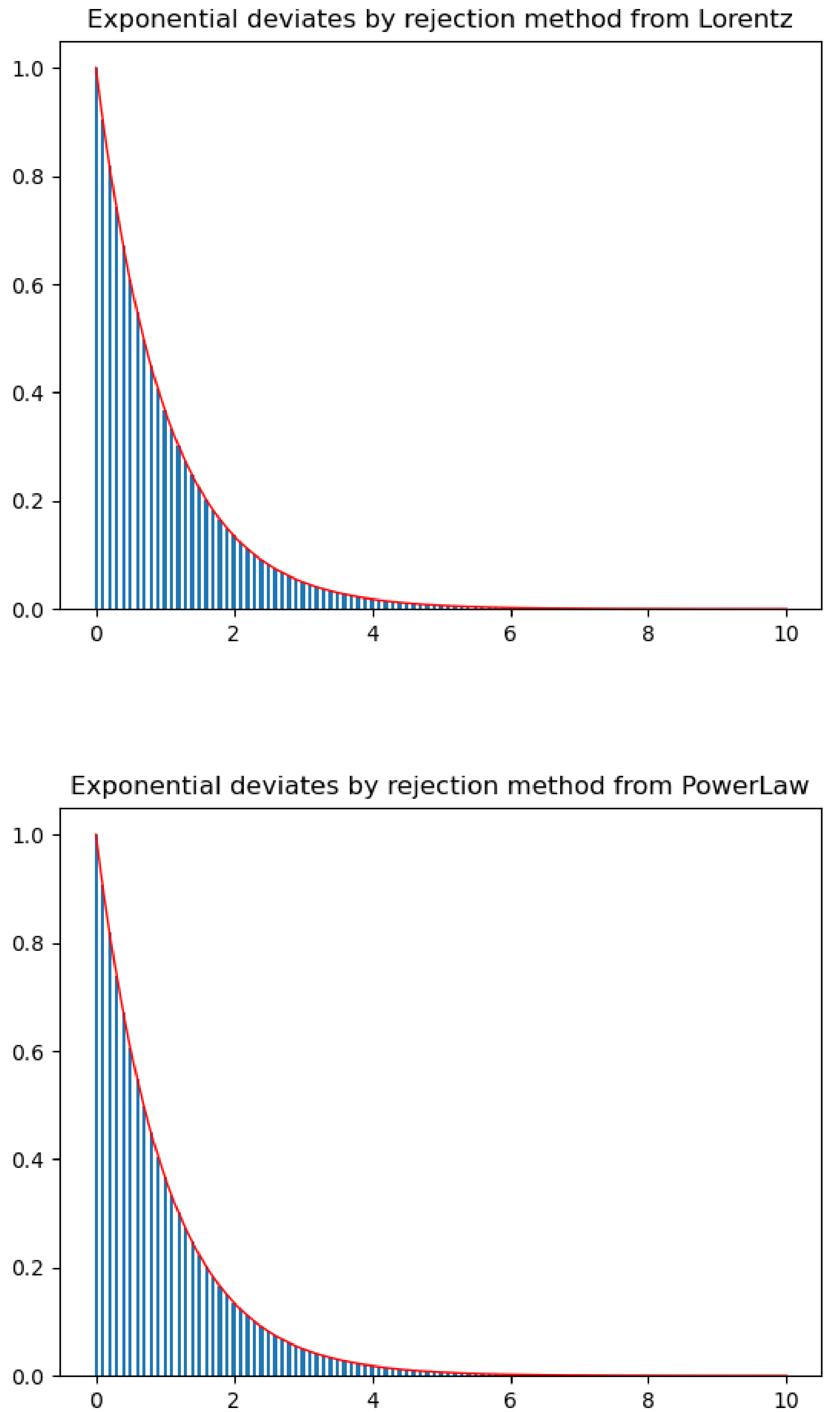


FIG. 3. Generate exponential deviates from another distribution by rejection method

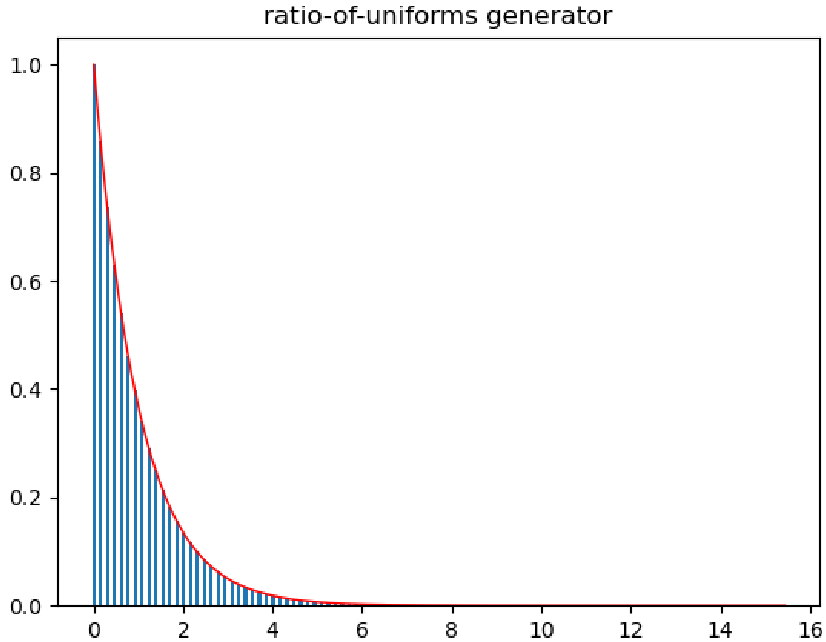


FIG. 4. Generate exponential deviates by ratio-of-uniforms generator.