بسم الله الرحمن الرحیم

گزارش کار تمرین کامپیوتری آمار و احتمال مهندسی

فاز اول

دکتر میرمحسنی

ترم بهار 96-97

مهرسا پوریا

شماره دانشجویی : 95101247

مقدمه

در این تمرین قصد داریم با کاربرد های آمار و احتمال در سیستم های مخابراتی آشنا شویم. برای این منظور در فاز اول متغیر های تصادفی برنولی ۱ ، گاوسی ۲ تولید می کنیم و سپس در فاز دوم به کمک متغیر های تصادفی یمتولید شده مدل ساده شده ی سیستم های مخابراتی (دیجیتال) را شبیه سازی می کنیم.

۱ تولید متغیر های تصادفی

در این قسمت فقط از تابع rand متلب می توانیم استفاده کنیم.

اجازه استفاده از توابع دیگر مانند randn ، rayland ، exprand و ... را نداریم.

در همین ابتدا دو پرسش مطرح می شود که از روی علاقه به آنها جواب می دهیم.

الف - تابع rand متغیر تصادفی با چه ویژگی هایی تولید می کند و چگونه این کار را انجام می دهد؟

ب - سایر توابع گفته شده چه فرقی با randn دارند؟

پاسخ پرسش الف :

تابع rand اعداد تصادفی در بازه ی 0 تا 1 تولید می کند که به صورت یکنواخت پخش شده اند.

اما این که این تابع چگونه این کار را انجام می دهد پاسخی طولانی تر و جالب تر دارد .
به طور کلی برای تولید اعداد تصادفی در کامپیوتر از الگوریتم های تولید اعداد تصادفی شبه تصادفی استفاده می کنند (pseudorandom number generator) ؛ شبه تصادفی زیرا این الگوریتم ها برای شروع به یک مقدار اولیه نیاز دارند و اگر این عدد شروع یکسان باشد دنباله حاصل یکی می شود که خلاف شهود ما از تصادفی بودن است. اگر این مساله را کنار بگذاریم الگوریتم با دنباله ی حاصلی که خواص بیشتری از اعداد تصادفی داشته باشد بهتر است همچنین مسله ی دیگر سرعت اجرای این الگوریتم است. نکته دیگر این است که در این الگوریتم ها پس از مدتی اگر به عددی مشابه عدد اولیه برسیم از ان پس دنباله تکرار می شود بنابراین شاهد تناوبی بودن خواهیم بود. هر چه این دوره ی تناوب بیشتر باشد بهتر است زیرا دنباله های طولانی تر از اعداد تولید می شوند.

اما این الگوریتم ها چه هستند ؟

به نقل از ویکیپدیای فارسی :

یکی از روش‌های اولیهٔ مبتنی بر کامپیوتر، به وسیلهٔ ون نیومن در سال ۱۹۴۶ ارائه شد. این روش به روش ریشهٔ میانی معروف است. شرح آن از این قرار است که اگر یک عدد دلخواه در نظر بگیرید، آن را به توان ۲ برسانید، ۴ رقم میانی آن را به عنوان یک عدد تصادفی در نظر بگیرید و از عدد به دست آمده به عنوان عدد بعدی مورد استفاده در الگوریتم، استفاده کنید. به‌طور مثال با شروع از عدد «۱۱۱۱»، عدد «۱۲۳۴۳۲۱» حاصل می‌شود، که اگر آن را به صورت هشت رقمی بنویسیم، «۰۱۲۳۴۳۲۱» به دست می‌آید. این عدد «۲۳۴۳» را به عنوان یک عدد تصافی به ما می‌دهد. با تکرار این عمل روی عدد «۲۳۴۳» به عدد «۴۸۹۶» به عنوان عدد تصادفی بعدی، خواهیم رسید. می‌توان همین روند را ادامه داد. قابل ذکر است که در روش اصلی از اعداد ۵ رقمی استفاده می‌شد، که روندی کاملاً مشابه دارد.

اکثر این روش ها بدین شکل است که یک عدد اولیه را برای شروع برمی گزینیم که امری غیر تصادفی است سپس با انجام عملیات ریاضی روی آن عدد ، عدد بعدی را تولید می کنیم و این روند با مبنا قرار دادن این عدد تولید تکرار می شود.

اما هنوز به پرسش به وجود آمده پاسخ نداده ایم. اکنون برای پرسش به آن می توانیم سوالی بهتر بپرسیم.

تابع rand متلب برای تولید اعداد تصادفی از چه الگوریتم هایی استفاده می کند؟

با مراجعه به راهنمای متلب با جدول زیر رو به رو می شویم. (rand documentation : Creating and Controlling a Random Number Stream)

Choosing a Random Number Generator

MATLAB software offers six generator algorithms. The following table summarizes the key properties of the available generator algorithms and the keywords used to create them. To return a list of all the available generator algorithms, use the `RandStream.list` method.

Generator algorithms

Keyword	Generator	Multiple Stream and Substream Support	Approximate Period In Full Precision
mt19937ar	Mersenne twister (default)	No	$2^{19937} - 1$
dsfmt19937	SIMD-oriented fast Mersenne twister	No	$2^{19937} - 1$
mcg16807	Multiplicative congruential generator	No	$2^{31} - 2$
mlfg6331_64	Multiplicative lagged Fibonacci generator	Yes	2^{124}
mrg32k3a	Combined multiple recursive generator	Yes	2^{191}
shr3cong	Shift-register generator summed with linear congruential generator	No	2^{64}
swb2712	Modified subtract with borrow generator	No	2^{1492}

همانطورکه در بالا مشاهده می کنیم روش پیش فرض (default) مورد استفاده الگوریتم مرسن (Mersenne twister) است.

به نقل از ویکی پدیای فارسی :

در سال ۱۹۹۷ اختراع الگوریتم مرسن، به وسیلهٔ ماکوتو ماکسوموتو و تاکوجو نیشیمورا، بسیاری از ضعف‌های الگوریتم‌های قبلی را برطرف نمود. ثابت شده‌است برای خروجی‌های ۳۲ بیتی تا ۶۲۳ بعد، به‌طور یکنواخت، توزیع شده‌است. از طرفی این الگوریتم، از الگوریتم‌های مشابه خود از نظر دقت، بسیار سریع‌تر است. به همین خاطر، روز به روز استفاده از این الگوریتم برای تولید اعداد تصادفی مورد نیاز در شبیه سازی‌ها و مدل سازی‌های مولد رو به افزایش است. همچنین نوع خاصی از این الگوریتم، که برای سیستم‌هایی با توانایی اجرای یک دستور برای تعداد زیادی داده را دارند، وجود دارد که بسیار سریع می‌باشد. پردازش گرهای گرافیکی از این نوع سیستم‌ها محسوب می‌شوند.

الگوریتم مرسن در واقع یک رابطه ی بازگشتی ماتریسی روی یک فضای دودویی است. (یک عملیات ریاضی که دنباله ی حاصل آن تخمین خوبی از اعداد تصادفی هستند.)

لینک زیر حاوی اطلاعات جالبی از این الگوریتم است :

https://en.wikipedia.org/wiki/Mersenne_Twister

پاسخ به پرسش ب :

دیدیم که تابع rand متغیر تصادفی یکنواخت تولید می کند همانگونه که از اسم سایر توابع پیداست آنها متغیر تصادفی با توزیع های دیگر تولید می کنند. برای مثال randn مولد توزیع نرمال و exprand مولد توزیع نمایی و ... است. به این دلیل که ما می خواهیم در این تمرین خودمان توزیع های مختلف را از متغیر با توزیع یکنواخت بسازیم اجازه استفاده از این توابع را نداریم.

به بحث اصلی این تمرین باز میگردیم.

۱.۱ تولید متغیر تصادفی برنولی

Bernoulli

Parameters	$0 < p < 1, p \in \mathbb{R}$
Support	$k \in \{0, 1\}$
pmf	$\begin{cases} q = (1-p) & \text{for } k = 0 \\ p & \text{for } k = 1 \end{cases}$

در این قسمت تابعی می نویسیم که با دریافت n و p برداری به طول n تولید کند که هر یک از درایه های آن مستقل از هم توزیع برنولی صفر و یک با پارامتر p داشته باشند. یعنی :

$$\mathbb{P}\{X = 1\} = 1 - \mathbb{P}\{X = 0\} = p$$

فرمت تابع :

$$Output = Bernoulli(p, n)$$

روش استفاده شده : هر درایه خروجی تابع rand یک عدد بین 0 تا 1 است که توزیع یکنواخت دارد. احتمال اینکه این درایه کوچکتر از عدد دلخواه p باشد ، برابر خود p است. همچنین احتمال اینکه بزرگتر از p باشد 1-p است. اثبات این امر :

$$f_X(x_i) = \begin{cases} 1 & 0 \le x_i \le 1 \\ 0 & o.w. \end{cases}$$

$$P(x_i \le p) = \int_0^p f_X(x_i) dx_i = \int_0^p dx_i = p$$

$$P(x_i > p) = 1 - P(x_i \le p) = 1 - p$$

بنابراین برای تولید متغیر برنولی کافی است به وسیله ی rand بردار تصادفی به طول n تولید کنیم سپس تک تک داریه ها را با عدد p مقایسه کنیم اگر آن درایه بزرگتر از p بود مقدار آن درایه را 0 (احتمال این امر طبق اثبات بالا 1-p است.) و اگر کوچتر از p بود آن درایه را 1 بگیریم. (احتمال این حالت p است.) با این روش برداری به طول n ساخته ایم که هر درایه آن مستقل از دیگری با احتمال p یک است و با احتمال 1-p صفر است به عبارت دیگری تک تک درایه ها مستقل از هم توزیع برنولی دارند. بدین شرح خواسته سوال محقق می شود.

کد : در متلب برای پیاده سازی روش گفته شده از rand(1,n) < p برای تولید دنباله برنولی استفاده می کنیم. این عبارت یک عبارت شرطی است که که مقایسه را روی تک تک درایه ها انجام می دهد اگر مقدار آن درایه کمتر از p باشد شرط برای آن درایه درست است و مقدار درایه متناظر خروجی 1 می شود و اگر آن درایه از p بزرگتر باشد شرط برای آن اشتباه است و مقدار متناظر در خروجی 0 می شود. خروجی های تولید شده از نوع logical هستند برای تبدیل آنها به اعداد اعشاری صفر و یک می توان از تابع double استفاده کرد.

همچنین در تابع نوشته شده دو شرط اینکه n مثبت باشد و اینکه p بین صفر و یک را باشد چک می کنیم. شرط صحیح بودن n در تابع rand چک می شود.

خروجی تابع برای n و p ای که به صورت دلخواه انتخاب شده اند به صورت زیر است :

```
p =

    0.3000


n =

    10


ourbernoulliseries =

     1     1     1     0     0     0     0     1     0     0
```

۲.۱ تولید متغیر تصادفی گاوسی

۱.۲.۱ یک قضیه مفید

فرض کنید متغیر تصادفی X از توزیع دلخواهی پیروی می‌کند و تابع توزیع تجمعی آن $F_X(x) = \mathbb{P}\{X \leq x\}$ است. اگر متغیر تصادفی U را این گونه تعریف کنیم که $U = F_X(x)$ ، متغیر U توزیع یکنواخت بین صفر و یک دارد.

قضیه گفته شده را ثابت می کنیم.

$$F_U(u) = P(U \le u) = P(F_X(X) \le u) = P\left(X \le F^{-1}{}_X(u)\right) = F\left(F^{-1}{}_X(u)\right) = u$$

در اثبات قضیه بالا از این فرض استفاده کردیم که به علت اکیدا صعودی بودن تابع احتمال پیوسته یک به یک است و بنابراین وارون پذیر است.

حال تابع CDF به دست آمده به شکل CDF یک تابع یکنواخت در بازه ی [0,1] است. بنابرین U توزیع یکنواخت بین صفر و یک دارد.

قضیه گفته شده در واقع همان قضیه (Probability integral transform) است.

حال اگر ما توزیع یکنواخت U بین صفر و یک را داشته باشیم چگونه می توانیم به توزیع دلخواه برسیم؟

$$U \sim Unif(0,1) \quad , \quad Y = g(U) \quad , \quad F_Y(y) \text{ or } f_Y(y)$$

در واقع ما به دنبال تابع g مناسب هستیم.

اگر فرض کنیم g یک به یک است :

$$Y = g(U) \;\rightarrow\; g^{-1}(Y) = U \;\rightarrow\; g^{-1}(Y) = F_Y(Y) \;\rightarrow g = F^{-1}{}_Y$$

از قضیه گفته شده در بالا استفاده کردیم.

بنابرین تابع g مناسب همان وارون تابع CDF است.

بنابراین ادعا می کنیم که اگر توزیع یکنواخت بین صفر و یک U را داشته باشیم ، اگر به خواهیم به توزیعی با CDF $F_Y(y)$ برسیم کافی است روی U تابع وارون CDF گفته شده را اعمال کنیم.ی

اثبات :

$$F_Y(y) = P(F_X{}^{-1}(U) \le y) = P(U < F_X(y)) = \int_0^{F_X(y)} f_U(u)du = \int_0^{F_X(y)} du = F_X(y)$$

بدین ترتیب به توزیعی با همان CDF ای که می خواستیم می رسیم.

این ادعا ی اثبات شده در واقع صورتی از قضیه inverse transform sampling است.

۲.۲.۱ تولید متغیر تصادفی نمای

الف- با استفاده از rand یک بردار اعداد تصادفی به طول n می سازیم که هر درایه آن توزیع یکنواخت بین صفر و یک دارد. (U)

$$X \sim exp(\lambda) \Longleftrightarrow f_X(x) = \frac{1}{\lambda}\exp(-\frac{x}{\lambda})$$

اولین کار ما این است که وارون CDF را محاسبه کنیم.

$$F_X(x) = 1 - e^{\frac{-x}{\lambda}} \qquad , F_X^{-1}(x) = -\lambda \ln(1-x)$$

حال کافی است X=g(U) را محاسبه کنیم که g همان ضابطه ی وارون CDF است که در بالا به دست آوردیم.
طبق قضیه بخش قبل حال هر درایه X مستقل از بقیه توزیع نمایی دارد.

فرمت تابع نوشته شده :

$$Output = Exponential(\lambda, n)$$

ب) به ازای n های گفته شده در متن تمرین تابع pdf را محاسبه می کنیم.

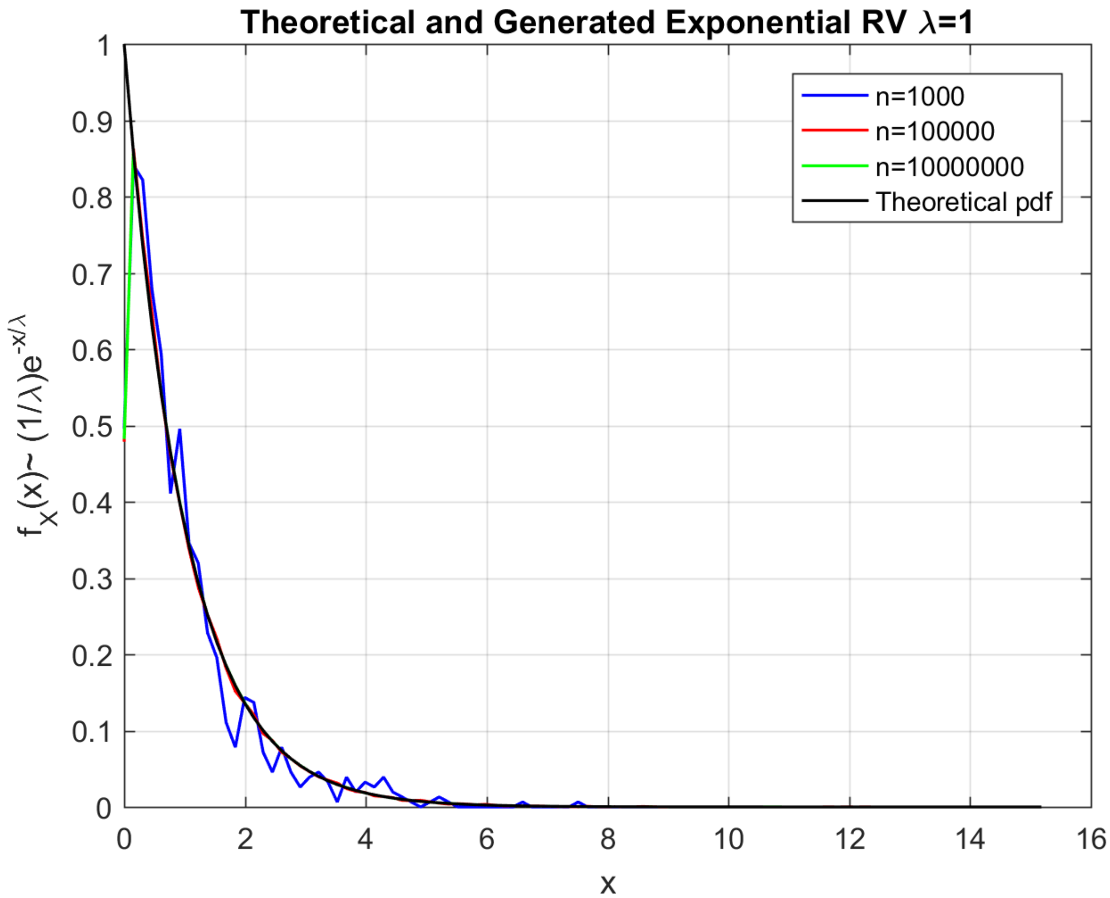
یک توضیح درباره ی محاسبه ی pdf تجربی :

با توجه به بیشترین و کمترین اعدادی که در خروجی تابعمان داریم باید محور اعداد را به تعداد مناسب بازه تقسیم کنیم. (در متن تمرین تعداد این بازه ها صد است.) سپس تعداد اعداد در هر بازه را محاسبه می کنیم. (همان خروجی تابع hist) و برای تبدیل تعداد اعداد در ان بازه به چگالی احتمال آنها باید تعداد را بر ضریبی بگونه تقسیم کنیم مساحت زیر نمودار یک شود. می دانیم که این مساحت از رابطه $\sum dt_i * C(t_i)$ محاسبه می شود (C نشان دهنده تعداد هر بازه است.) که با توجه به اینکه dt_i برای همه ی بازه ها برابر است از سیگما بیرون می اید و حاصل سیگمای باقی مانده همان تعداد کل اعداد یا n است. بنابرین ضرب مناسب برای آنکه خروجی تابع hist را بر آن تقسیم کنیم و به pdf برسیم همانطورکه در کد داده شده آمده است dt*n است. (البته در این اینجا یک تقریب از بینهایت برای ماکسیمم عدد خروجی زده ایم که هر چه ماکس عدد خروجی بزرگتر باشد تقریب مان بهتر است ، با بزرگتر کردن n شانس اینکه این عدد بزرگتر شود وجود دارد و تقریب دیگر تقریب مساحت زیر نمودار به علت افراز به m قسمت گسسته است که هر چه m بیشتر شود تقریب دقیق تر است.)کد داده شده :

```
m=100;
t=linspace(min(x),max(x),m);
dt=t(2)-t(1);
f_x=hist(x,t)/(N*dt);
figure
plot(t,f_x);
```

برای pdf نظری نیز از رابطه چگالی احتمال داده شده استفاده کردیم.

خروجی ها به شکل زیر است :



۳.۲.۱ تولید متغیر تصادفی

الف) فرض کنید که متغیر تصادفی X از توزیع نمایی با پارامتر $\lambda = 1$ پیروی می‌کند و هم‌چنین متغیر تصادفی Y از رابطه‌ی $Y = \sigma\sqrt{2X}$ بدست می‌آید. ثابت کنید که متغیر تصادفی Y از توزیع رایلی با پارامتر σ پیروی می‌کند.

$$Y \sim Rayleigh(\sigma) \Longleftrightarrow f_Y(y) = \frac{y}{\sigma^2}\exp(-\frac{y^2}{2\sigma^2})$$

اثبات :

$$Y = \sigma\sqrt{2X} \qquad f_X(x) = \exp(-x)$$

با استفاده از قضیه اساسی :

$$f_Y(y) = \frac{f_X(x_i)}{\dot{g}(x_i)} = \frac{\exp(-(\frac{y^2}{2\sigma^2})^1)}{\frac{\sigma}{\sqrt{\frac{y^2}{\sigma^2}}}} = \frac{y}{\sigma^2}\exp(-\frac{y^2}{2\sigma^2}) \qquad x_i = \frac{y^2}{2\sigma^2}$$

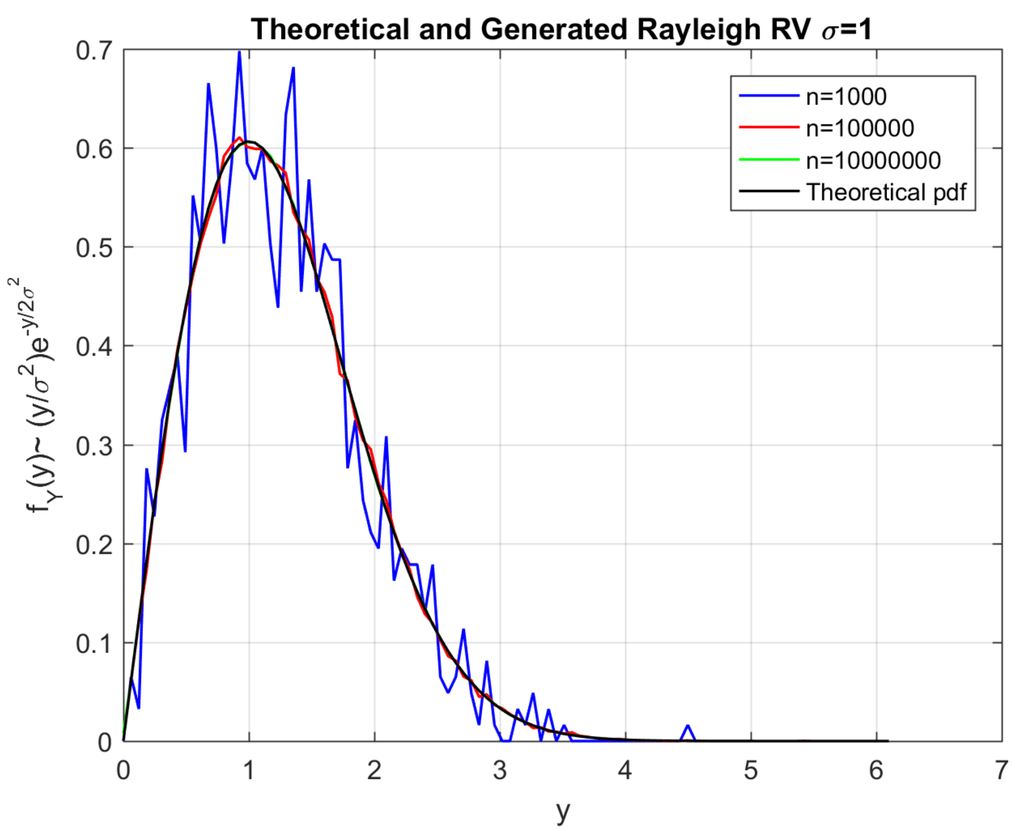
Y توزیع رایلی دارد. گزاره اثبات شد.

ب) در بخش قبل از متغیر تصادفی یکنواخت به توزیع نمایی رسیدم و در این قبل اثبات کردیم با رابطه گفته شده می توان از نمایی به رایلی رسید. از همین شهود استفاده می کنیم و متغییر تصادفی رایلی می سازیم.

تابعی به این فرمت تعریف می کنیم.

$$Output = Rayleigh(\sigma, n)$$

ج) نمودار ها برای n های گفته شده و مقدار عملی به شکل زیر است :



۴.۲.۱ تولید متغیر تصادفی گاوسی

الف) قضیه‌ی مهمی در احتمال وجود دارد که می‌گوید اگر متغیر R از توزیع رایلی با پارامتر σ و متغیر ϕ از توزیع یکنواخت در بازه‌ی $[0, 2\pi]$ پیروی کند، آن‌گاه متغیرهای تصادفی $X = R\cos(\phi)$ و $Y = R\sin(\phi)$ هر کدام از توزیع گاوسی با میانگین صفر و واریانس σ^2 پیروی می‌کنند.

هم‌چنین قضیه‌ی مهم دیگری نیز در احتمال وجود دارد که نشان می‌دهد اگر دو متغیر تصادفی گاوسی ناهمبسته [۴] باشند، این دو متغیر تصادفی الزاما مستقل [5] می‌باشند.

ثابت کنید که متغیرهای تصادفی X و Y تعریف شده در همین بخش، دو متغیر تصادفی مستقل می‌باشند.

اثبات : طبق قضیه های گفته شده می دانیم دو متغیر X و Y گاوسی با میانگین صفر هستند. حال کافی است نشان دهیم این دو متغیر ناهمبسته هستند تا به استقلال آنها برسیم بدین منظور کافیست نشان دهیم کوواریانس آنها صفر است.

$$Cov(X,Y) = E[(X-\eta_x)(Y-\eta_y)] = E[XY] = \iint_{-\infty}^{+\infty} xyf_{xy}(x,y)dxdy$$

به وضوح معلوم است R و ϕ مستقل هستند بنابراین توزیع مشترک X و Y را بدین شکل به دست می آوریم.

$$f_{r\phi}(r,\phi) = \frac{r}{\sigma^2}\exp\left(-\frac{r^2}{2\sigma^2}\right) \times \frac{1}{2\pi}u(r) \qquad 0 < \phi < 2\pi$$

$$J = \begin{pmatrix} cos\phi & -rsin\phi \\ sin\phi & rcos\phi \end{pmatrix} \qquad \det J = r$$

$$ri = \sqrt{y^2+x^2} \qquad f\ xy(x,y) = \frac{\frac{r_i}{\sigma^2}\exp\left(-\frac{r_i{}^2}{2\sigma^2}\right) \times \frac{1}{2\pi}u(r_i)}{r_i} = \frac{1}{2\pi\sigma^2}\exp(-\frac{y^2+x^2}{2\sigma^2})$$

بنابراین :

$$E[XY] = \iint_{-\infty}^{+\infty} xyf_{xy}(x,y)dxdy = \int_{-\infty}^{+\infty}\int_0^{2\pi} \frac{1}{2\pi\sigma^2} sin2\phi\ \frac{1}{2}r^2\exp\left(-\frac{r^2}{2\sigma^2}\right) rdrd\phi = 0$$

بنابراین این دو متغیر ناهمبسته هستند (نشان دادیم کوواریانس صفر است.) پس طبق قضیه گفته شده مستقل نیز هستند.

ب) ما توزیع رایلی و یکنواخت را می توانیم با توجه به بخش های قبل تولید کنیم همچنین در قسمت الف نشان دادیم که که می توانیم به متغیر گاوسی برسیم. Y=RCOSX که R توزیع رایلی و X نرمال بین صفر و دو پی را دارد توزیع گاوسی دارد. برای بردن متغیر یکنواخت بین صفر و یک خروجی rand به بین صفر و دو پی کافی است یک دو پی در آن ضرب کنیم.

فرمت تابع :

$$Output = Gaussian(\sigma, n)$$

ج) به ازای مقادیر گفته شده pdf تجربی را رسم می کنیم.

برای توزیع نظری :

$$X \sim \mathcal{N}(\mu, \sigma^2) \Longleftrightarrow f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{x^2}{2\sigma^2})$$