علی آراسته ۹۶۱۰۱۱۶۵

گزارش سری اول تمرین کامپیوتری آمار و احتمال

سوال ۱:

۱)

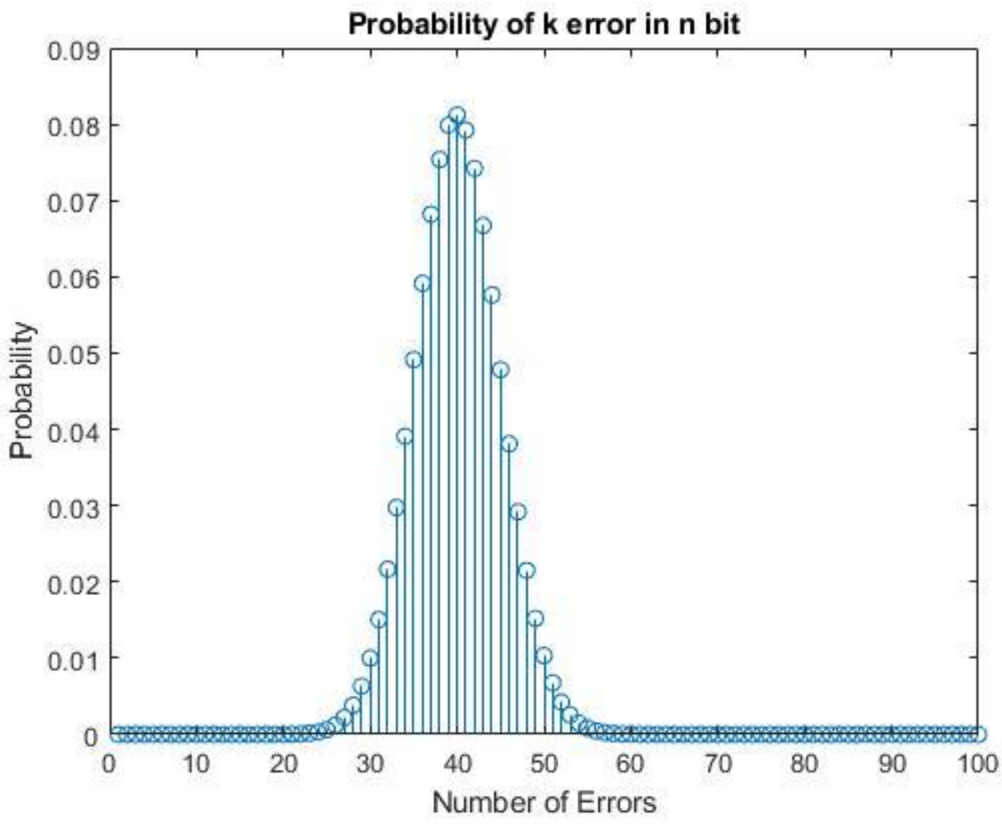
تئوری:

به $\binom{100}{k}$ طریق می‌توان k بیت از ۱۰۰ بیت را انتخاب کرد. هر بیت با احتمال ۰/۴ در ارسال دچار خطا می‌شود. بنابراین برای محاسبه احتمال k بت خطا در ۱۰۰ بیت می‌توان از فرمول زیر استفاده کرد:

$$Probability\ of\ k\ k\ bit\ error\ in\ 100\ bit = \binom{100}{k} * (0.4)^k * \ (1-0.4)^{100-k}$$

شبیه‌سازی:

با استفاده از متلب و دستور stem، نمودار احتمال k بیت خطا در ۱۰۰ بیت را بر حسب k رسم می‌کنیم:



۲)

تابع مورد نظر در انتهای فایل متلب (m فایل) نوشته شده است. sentData ماتریس ورودی و receivdData ماتریس خروجی است.

۳)

برای مثال یک بردار به عنوان ورودی به صورت تصادفی تولید می‌کنیم:

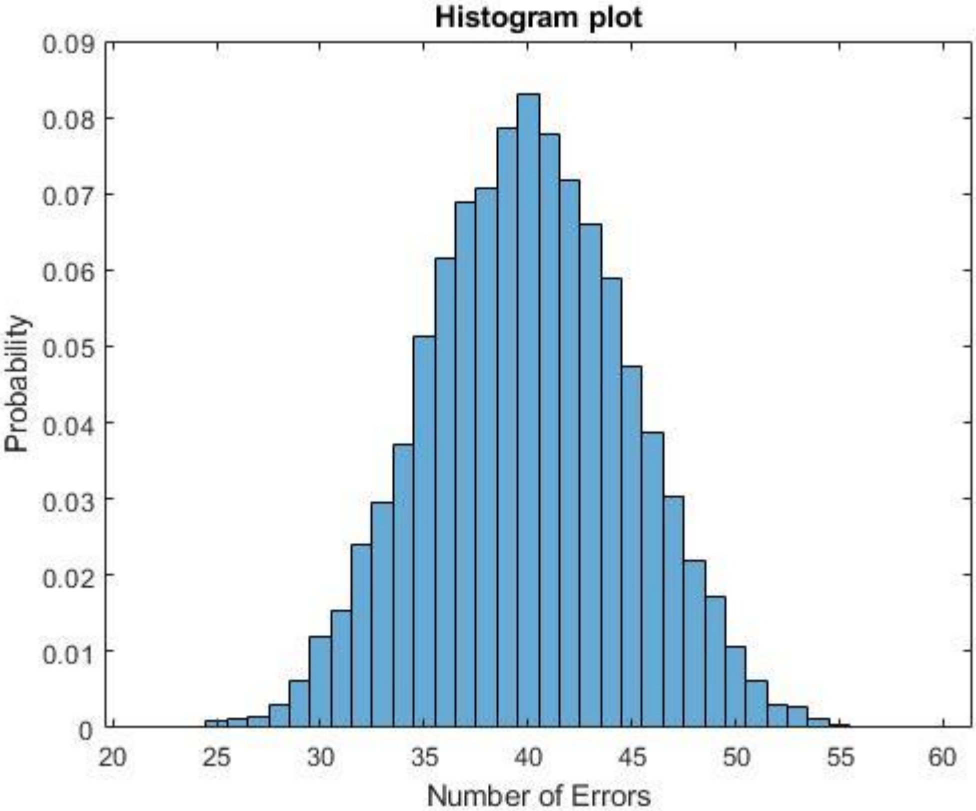
$$sentData = [1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 1]$$

با استفاده از بردار sentData و تابع sendSimulator، بردار خروجی receivdData را تولید می‌کنیم:

$$receivdData = [1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1]$$

$$number\ of\ different\ bits = 8\ bit\ in\ 20\ bit$$

با استفاده از متلب و دستور histogram، نمودار احتمال k بیت خطا در ۱۰۰ بیت را بر حسب k رسم می‌کنیم:



شکل کلی هر دو نمودار یکسان است؛ اما تفاوت‌هایی به علت نادقیق بودن نمودار هیستوگرام وجود دارد. هر چه تعداد آزمایش‌های تکراری بیشتر شود، دقت نمودار هیستوگرام نیز بیشتر می‌شود و به نمودار قسمت ۱ نزدیک‌تر می‌شود.

پارامتر normalization نمودار هیستوگرام تعیین کننده معیار محور y این نمودار است. اگر مقدار این پارامتر count باشد، تعداد تکرار نتیجه مورد نظر و اگر probability باشد، احتمال نتیجه مورد نظر (تعداد تکرار تقسیم بر تعداد کل)، روی محور y نمایش داده می‌شود. در نتیجه اگر مقدار این پارامتر probability باشد برای خواسته مسئله مناسب‌تر است.

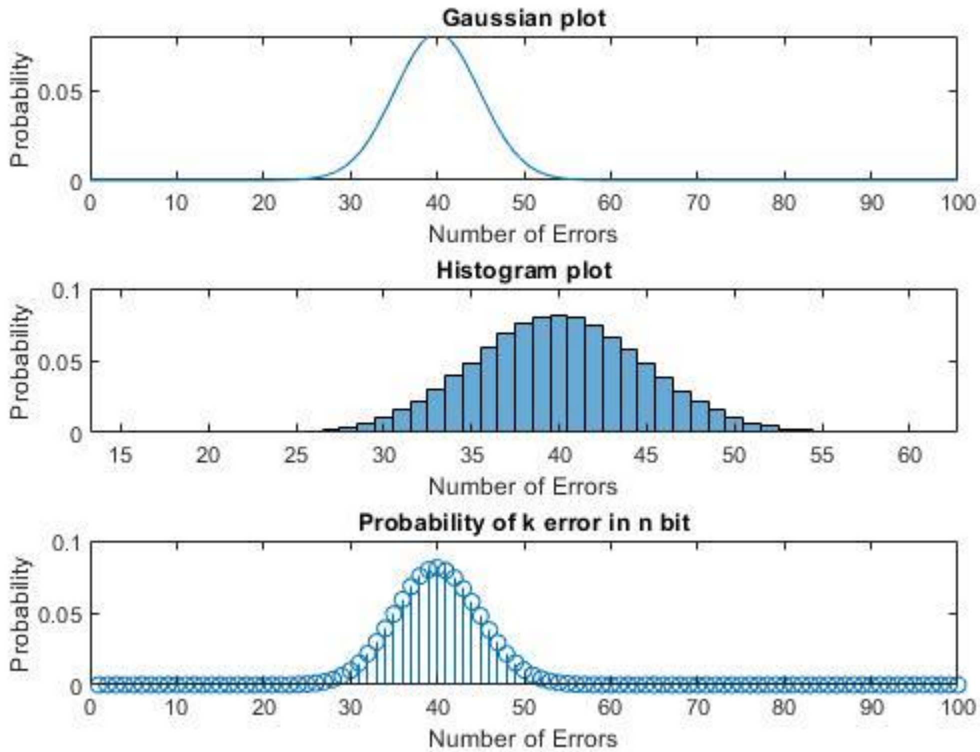
۴)

از آن جایی که در مسئله n (۱۰۰)، np (۴۰) و npq (۲۴) هر سه بسیار بزرگتر از ۱ هستند، تقریب نرمال مناسب‌تر است و می‌توانیم از تابع نرمال با پارامترهای مناسب تقریب خوبی از جواب بدست آوریم:

$$f(k) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(k-\eta)^2}{2\sigma^2}} = \frac{1}{\sqrt{2\pi(npq)}}\, e^{-\frac{(k-(np))^2}{2(npq)}} = \frac{1}{\sqrt{48\pi}}\, e^{-\frac{(k-40)^2}{48}}$$

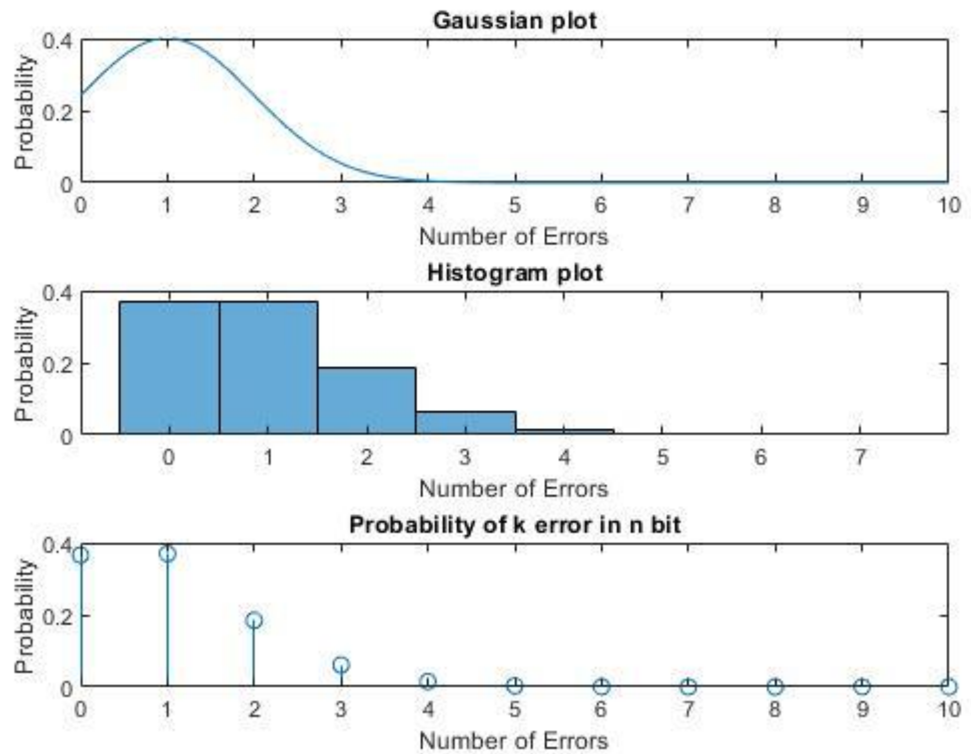
۵)

با استفاده از متلب، تعریف تابع تحلیلی و دستور fplot، نمودار تقریب نرمال احتمال k بیت خطا در ۱۰۰ بیت را بر حسب k رسم می‌کنیم:



۶)

مقدار احتمال p را از ۰/۴ به ۰/۰۱ تغییر می‌دهیم و مراحل بخش قبل را بر اساس پارامترهای جدید تکرار می‌کنیم:



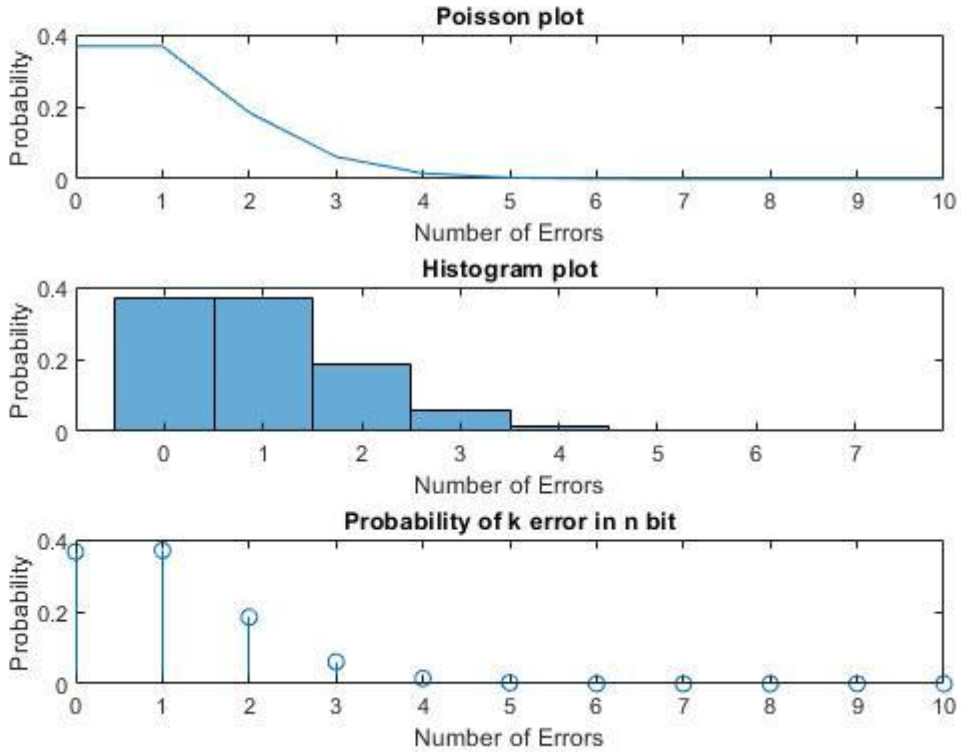
از نمودارهای بدست آمده مشخص است که دقت تقریب در این بخش از بخش قبل کمتر است. علت نادقیق بودن تقریب در این بخش، قابل مقایسه بودن np و npq با ۱ است. در نتیجه در این بخش تقریب پواسون از تقریب نرمال دقیق‌تر است.

۷)

می‌توانیم از تقریب پواسون با پارامترهای مناسب استفاده کنیم:

$$f(k) = e^{-np} * \frac{(np)^k}{k!} = e^{-1} * \frac{(1)^k}{k!} = \frac{e^{-1}}{k!}$$

با استفاده از متلب، تعریف تابع تحلیلی و دستور plot، نمودار تقریب پواسون احتمال k بیت خطا در ۱۰۰ بیت را بر حسب k رسم می‌کنیم:



سوال ۲: تقریب عدد پی

۱)

احتمال پیشامد مطلوب (نقطه در داخل دایره‌ای به مرکز مبدا و شعاع ۱ قرار بگیرد)، برابر با حاصل تقسیم مساحت پیشامد مطلوب (مساحت دایره به مرکز مبدا و شعاع ۱) بر مساحت فضای نمونه (مربع به طول و عرض ۲) است:

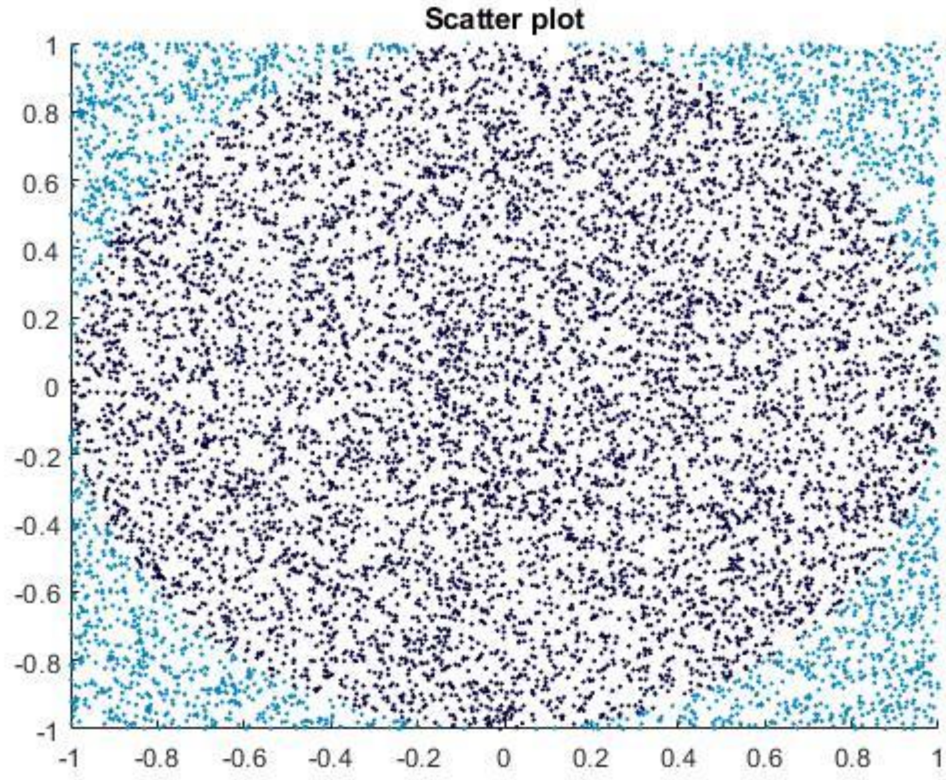
$$P\left(\text{پیشامد مطلوب}\right) = \frac{S_{\text{پیشامد مطلوب}}}{S_{\text{فضای نمونه}}} = \frac{\pi}{4}$$

۲)

با انتخاب تعداد بسیار زیادی نقطه تصادفی، محاسبه تعداد نقاطی که داخل دایره به مرکز مبدا و شعاع ۱ و تقسیم این تعداد بر تعداد کل نقاط، احتمال پیشامد مطلوب را بدست می‌آوریم:

$$P\left(\text{پیشامد مطلوب}\right) = 0.7867$$

با استفاده از متلب و دستور scatter، نقاط تصادفی بدست آمده را نمایش می‌دهیم:



۳)

بر اساس قوانین احتمال، جواب‌های بدست آمده از دو روش باید یکسان باشند. در نتیجه داریم:

$$\frac{\pi}{4} = 0.7867$$

$$\pi = 3.1468$$

عدد بدست آمده تقریب خوبی از عدد پی است.

سوال ۲: تقریب عدد نپر

۱)

اگر پیشامد A_i را به صورت رو به رو تعریف کنیم: «نامه iام در پاکت درست قرار بگیرد»، احتمال A_i به صورت رو به رو محاسبه می‌شود:

$$P(A_i) = \frac{(n-1)!}{n!} = \frac{1}{n}$$

پیشامد مطلوب (حداقل یک نامه در پاکت درست قرار بگیرد) را می‌توانیم به صورت اجتماع پیشامدهای A_i برای i از ۱ تا n تعریف کنیم:

$$P\left(\text{پیشامد مطلوب}\right) = \mathrm{P}\left(\bigcup_{i=1}^{n} A_i\right) = \sum_{i=1}^{n} P(A_i) - \sum_{1\le i<j\le n} P\left(A_i \bigcap A_j\right) + \cdots + (-1)^{n+1} P\left(\bigcap_{i=1}^{n} A_i\right)$$

$$P\left(\text{پیشامد مطلوب}\right) = \binom{n}{1} * \frac{1}{n} - \binom{n}{2} * \frac{1}{n*(n-1)} + \cdots + (-1)^{n+1} * \binom{n}{n} * \frac{1}{n!}$$

$$P\left(\text{پیشامد مطلوب}\right) = 1 - \frac{1}{2!} + \cdots + (-1)^{n+1} * \frac{1}{n!}$$

احتمال پیشامد مطلوب را با میل دادن n به سمت بی‌نهایت بدست می‌آوریم:

$$\lim_{n\to\infty} P\left(\text{پیشامد مطلوب}\right) = \lim_{n\to\infty}\left(1 - \frac{1}{2!} + \cdots + (-1)^{n+1} * \frac{1}{n!}\right) = \frac{e-1}{e}$$

۲)

با استفاده از متلب و دستور randperm، احتمال پیشامد مطلوب را از روش آزمایش‌های تکراری بدست می‌آوریم:

$$P\left(\text{پیشامد مطلوب}\right) = 0.6321$$

۳)

بر اساس قوانین احتمال، جواب‌های بدست آمده از دو روش باید یکسان باشند. در نتیجه داریم:

$$\frac{e-1}{e} = 0.6321$$

$$e = 2.7182$$