پاییز ۱۴۰۲ **تمرین ۴**

ایمان محمدی (۹۹۱۰۲۲۰۷) **زمان آپلود: ۲ بهمن**

سوال ۱

برای هر کدام از موارد زیر، توزیع احتمالاتی مناسب آن را نامیده و علت انتخاب خود را شرح دهید.

الف) سرعت ماشین ها در اتوبان

ب) درآمد یک سوپرمارکت در یک ماه

ج) زمان طول کشیده برای نوشتن یک تمرین

د) نقطه برخورد در بازی دارت

جواب سوال ۱

الف) **سرعت ماشین‌ها در اتوبان:** توزیع نرمال (گاوسی) می‌تواند مناسب باشد. بسیاری از ماشین‌ها با سرعت‌هایی نزدیک به میانگین سرعت مجاز حرکت می‌کنند، با این حال، برخی از رانندگان ممکن است کمی سریع‌تر یا کمی آهسته‌تر حرکت کنند. توزیع نرمال به خوبی این واقعیت را که اکثر داده‌ها حول میانگین تجمع می‌یابند و داده‌های دور از میانگین نادر هستند، منعکس می‌کند.

ب) **درآمد یک سوپرمارکت در یک ماه:** درآمد یک سوپرمارکت در یک ماه: توزیع لگاریتمی نرمال ممکن است گزینه مناسبی باشد. درآمدها اغلب توزیعی نامتقارن دارند که در آن مقادیر بزرگ‌تر (درآمدهای بالاتر) کمتر اتفاق می‌افتند. توزیع لگاریتمی نرمال این ویژگی را نشان می‌دهد، جایی که داده‌ها پس از لگاریتم گرفتن، توزیع نرمال‌تری را نشان می‌دهند.

ج) **زمان طول کشیده برای نوشتن یک تمرین:** توزیع اگزپوننشیال می‌تواند مناسب باشد، زیرا این توزیع برای مدل‌سازی زمان انتظار یا زمان تا رخداد یک رویداد منفرد استفاده می‌شود. در این مورد، زمان لازم برای اتمام تمرین می‌تواند به عنوان یک "زمان انتظار" برای تکمیل کار در نظر گرفته شود.

د) **نقطه برخورد در بازی دارت:** توزیع دایره‌ای یا توزیع یکنواخت بر روی دایره می‌تواند مناسب باشد. اگر فرض کنیم که هدف‌گیری‌ها تصادفی و یکنواخت بر روی سطح هدف صورت می‌گیرد، پس توزیع یکنواخت یک گزینه مناسب است. این توزیع نشان می‌دهد که هر نقطه بر روی هدف به طور یکسان احتمال دارد.

سوال ۲

سوال: روشی را برای تولید واریته تصادفی برای متغیر X با تابع چگالی احتمال (pdf) زیر پیدا کنید:

$$f(x) = \begin{cases} x^{a-1} & \text{if} x \leq 1 \\ \lambda^{-\lambda x} & \text{otherwise} \end{cases}$$

جواب سوال ۲

برای تولید یک متغیر تصادفی X بر اساس تابع چگالی احتمال داده شده، ابتدا تابع توزیع تجمعی (CDF) را برای هر بخش از $f(x)$ محاسبه می‌کنیم.

برای $x \leq ۱$:

$$f(x) = x^{a-۱}$$
$$F(x) = \int_{۰}^{x} t^{a-۱} dt = \frac{x^a}{a}$$

برای $x > ۱$:

$$f(x) = \lambda^{-\lambda x}$$
$$F(x) = \int_{۱}^{x} \lambda^{-\lambda t} dt = \frac{-\lambda^{-\lambda x-۱}}{ln(\lambda)} + C$$

با این حال، محاسبه C و حل معادله تبدیل معکوس برای $x > ۱$ به دلیل مشکلات موجود در تابع چگالی احتمال، ناقص است.

برای محاسبه دقیق تر این مقادیر و تعیین C و همچنین حل معادله تبدیل معکوس برای $x > ۱$، این محاسبات را انجام می‌دهیم.

برای $x \leq ۱$:

$$F(۱) = \frac{۱}{a}$$

برای $x > ۱$: مقدار C را محاسبه کردیم، اما به نظر می‌رسد که نتیجه به دلیل ماهیت معادله به صورت "lr"zoo (یعنی undefined) بدست آمده است. معادله تبدیل معکوس برای x در قسمت $x > ۱$ نیز حل نشده است. مشکل اصلی اینجا این است که فرمول دوم برای $f(x)$ برای $x > ۱$، یعنی $\lambda^{-\lambda x}$، یک فرمول غیرمعمول برای یک تابع چگالی احتمال است و انتگرال آن به سادگی قابل حل نیست. این موضوع باعث می‌شود که محاسبه C و حل معادله تبدیل معکوس برای این قسمت دشوار یا حتی غیرممکن باشد.

برای حل این مشکل، توصیه می‌کنم که مجدداً فرمول $f(x)$ برای $x > ۱$ را بررسی کنید تا اطمینان حاصل شود که این فرمول مناسب برای یک تابع چگالی احتمال است. اگر امکان تغییر فرمول وجود دارد، استفاده از یک فرمول متداول‌تر مانند $\lambda e^{-\lambda x}$ برای $x > ۱$ می‌تواند مسئله را قابل حل‌تر کند.

برای تولید متغیر تصادفی X از روش تبدیل معکوس استفاده می‌کنیم:

برای $x \leq ۱$:

$$F(x) = \frac{x^a}{a}$$

برای تولید X، یک عدد تصادفی یکنواخت U در بازه $[۰, ۱]$ تولید می‌کنیم و حل می‌کنیم:

$$X = (aU)^{\frac{۱}{a}}$$

برای $x > ۱$:

$$F(x) = \frac{-\lambda^{-\lambda x-۱}}{\ln(\lambda)} + C$$

برای تولید X، یک عدد تصادفی یکنواخت U در بازه $[۰, ۱]$ تولید می‌کنیم و حل می‌کنیم:

$$X = \frac{-۱ - \ln(-U\ln(\lambda) + C\ln(\lambda))}{\lambda}$$

سوال ۳

جواب سوال ۳

سوال ۴

داده‌های جدول زیر، نمونه‌های جمع‌آوری شده از مدت زمان سرویس در یک سیستم صف هستند. با کمک این داده‌ها، یک جدول برای تولید زمان‌های سرویس دهی ایجاد کنید (مشابه اسلاید ۱۶ از فصل ۷) و برای ۵ عدد تصادفی R_i زمان سرویس متناظر را تعیین کنید. برای تولید R_i می‌توانید از روشی دلخواه استفاده کنید.

Frequency	(seconds) Interval
۱۰	۳۰-۱۵
۲۰	۴۵-۳۰
۲۵	۶۰-۴۵
۳۵	۹۰-۶۰
۳۰	۱۲۰-۹۰
۲۰	۱۸۰-۱۲۰
۱۰	۳۰۰-۱۸۰

جواب سوال ۴

جدول تولید زمان‌های سرویس دهی بر اساس داده‌های جدول موجود:

Distribution Cumulative	Frequency Relative	Frequency	(seconds) Interval
۰/۰۶۷	۰/۰۶۷	۱۰	۱۵ − ۳۰
۰/۲۰۰	۰/۱۳۳	۲۰	۳۰ − ۴۵
۰/۳۶۷	۰/۱۶۷	۲۵	۴۵ − ۶۰
۰/۶۰۰	۰/۲۳۳	۳۵	۶۰ − ۹۰
۰/۸۰۰	۰/۲۰۰	۳۰	۹۰ − ۱۲۰
۰/۹۳۳	۰/۱۳۳	۲۰	۱۲۰ − ۱۸۰
۱/۰۰۰	۰/۰۶۷	۱۰	۱۸۰ − ۳۰۰

برای پنج عدد تصادفی R_i تولید شده، بازه‌های زمانی متناظر به شرح زیر هستند:

- $R_۱ = ۰/۳۳۳$: بازه زمانی ۴۵-۶۰
- $R_۲ = ۰/۶۸۶$: بازه زمانی ۹۰-۱۲۰
- $R_۳ = ۰/۲۴۴$: بازه زمانی ۴۵-۶۰
- $R_۴ = ۰/۴۳۰$: بازه زمانی ۶۰-۹۰
- $R_۵ = ۰/۴۸۶$: بازه زمانی ۶۰-۹۰

در این کد، ابتدا داده‌های مربوط به زمان سرویس در یک سیستم صف و فراوانی هر بازه زمانی را در یک دیکشنری به نام data ذخیره می‌کنیم. سپس، این دیکشنری را به یک DataFrame پانداس تبدیل می‌کنیم تا بتوانیم روی داده‌ها به راحتی عملیات‌های مختلفی انجام دهیم.

توضیح خطوط کد:

ایجاد DataFrame از داده‌ها:

df = pd.DataFrame(data): این خط کد، داده‌های موجود در دیکشنری data را به یک DataFrame پانداس تبدیل می‌کند.

محاسبه فراوانی کل:

total_frequency = df['Frequency'].sum(): با استفاده از این خط، فراوانی کلی تمام بازه‌های زمانی محاسبه می‌شود.

محاسبه فراوانی نسبی و توزیع تجمعی:

df['Relative Frequency'] = df['Frequency'] / total_frequency: این خط، فراوانی نسبی هر بازه زمانی را با تقسیم فراوانی هر بازه بر فراوانی کل محاسبه می‌کند.

df['Cumulative Distribution'] = df['Relative Frequency'].cumsum(): در این خط، توزیع تجمعی برای هر بازه زمانی را با جمع تراکمی فراوانی‌های نسبی محاسبه می‌کنیم.

تولید ۵ عدد تصادفی و تعیین بازه‌های زمانی متناظر:

random_numbers = np.random.uniform(0, 1, 5): این خط، ۵ عدد تصادفی در بازه [۰, ۱] تولید می‌کند.

تابع determine_service_time: این تابع برای هر عدد تصادفی R ، بازه زمانی متناظر با آن را با توجه به توزیع تجمعی مشخص می‌کند.

service_times = [determine_service_time(R, df) for R in random_numbers]: با استفاده از این خط، برای هر یک از اعداد تصادفی تولید شده، بازه زمانی متناظر را مشخص می‌کنیم.

در نهایت، بازه‌های زمانی متناظر با هر عدد تصادفی به همراه خود اعداد تصادفی چاپ می‌شوند. این اطلاعات می‌توانند برای تحلیل‌های بعدی یا برای تعیین زمان‌های سرویس در سیستم صف مورد استفاده قرار گیرند.

سوال ۵

جواب سوال ۵

گام‌های الگوریتم:

۱. تولید یک عدد تصادفی Y از توزیع R :

برای این کار می‌توان از روش تبدیل معکوس استفاده کرد.

۲. تولید یک عدد تصادفی U از یک توزیع یونیفرم بین ۰ و ۱

۳. قبول یا رد کردن Y :

اگر $U \leq \frac{s(Y)}{c.r(Y)}$ باشد، Y به عنوان یک متغیر تصادفی از توزیع S قبول می‌شود، در غیر این صورت Y رد می‌شود و به گام اول برمی‌گردد.

اجرای الگوریتم:

ما ابتدا باید توزیع R را انتخاب کنیم. یک انتخاب مناسب برای این توزیع می‌تواند یک توزیع نرمال استاندارد باشد، چرا که مشابه با توزیع S است و همچنین می‌توانیم به راحتی از آن نمونه‌گیری کنیم.

اکنون، یک عدد تصادفی از توزیع $|Z|$ با استفاده از این روش تولید می‌کنیم. برای این کار، ابتدا یک عدد تصادفی از توزیع نرمال استاندارد تولید می‌کنیم و سپس از قانون پذیرش و رد استفاده می‌کنیم. در نهایت، صحت این روش را با بررسی شرط $U \leq \frac{s(Y)}{c.r(Y)}$ تایید می‌کنیم.

اجرای الگوریتم به صورت عملی:

ابتدا یک نمونه از توزیع نرمال استاندارد تولید می‌کنیم و سپس مراحل را دنبال می‌کنیم.

عدد تصادفی تولید شده از توزیع قدر مطلق نرمال استاندارد $|Z|$ برابر است با ۰/۵۴۵۷ این نتیجه با استفاده از الگوریتم پذیرش–رد و با فرض توزیع نرمال استاندارد به عنوان توزیع R به دست آمده است.

برای اثبات اینکه الگوریتم ارائه شده به درستی کار می‌کند، باید نشان دهیم که احتمال پذیرش یک نمونه Y برابر با تابع توزیع تجمعی $S(y)$ است. به عبارت دیگر، ما باید نشان دهیم که:

$$P(U \leq \frac{s(Y)}{c \cdot r(Y)}|Y \leq y) = S(y)$$

با توجه به اینکه U یک متغیر تصادفی یکنواخت بین ۰ و ۱ است، احتمال $U \leq \frac{s(Y)}{c \cdot r(Y)}$ معادل با $\frac{s(Y)}{c \cdot r(Y)}$ است. بنابراین، ما می‌توانیم بنویسیم:

$$P(U \leq \frac{s(Y)}{c \cdot r(Y)}|Y \leq y) = P(\frac{s(Y)}{c \cdot r(Y)}|Y \leq y)$$

این احتمال باید برابر با $S(y)$ باشد، که تابع توزیع تجمعی توزیع S است. برای نشان دادن این امر، ما باید از انتگرال‌گیری استفاده کنیم. به طور خاص، این بدین معناست که میانگین شرطی $\frac{s(Y)}{c \cdot r(Y)}$ برای Yهای کوچکتر یا مساوی y باید برابر با $S(y)$ باشد.

برای این بخش کد زده شده و کد و نتیجه‌ی آن پیوست شده‌اند. از طریق محاسبات انتگرال، ما به نتایج زیر رسیدیم:

۱. تابع توزیع تجمعی $S(y)$ برای توزیع قدر مطلق نرمال استاندارد برابر است با $\text{Heaviside}(y) \cdot \text{erf}\left(\frac{\sqrt{۲} \cdot y}{۲}\right)$. این تابع نشان دهنده احتمال این است که یک متغیر تصادفی نرمال استاندارد کمتر یا مساوی y باشد، به شرطی که مقدار مطلق آن گرفته شود.

۲. انتگرال برای محاسبه $E\left[\frac{s(Y)}{c\cdot r(Y)}|Y \leq y\right]$ برابر است با $۲y \cdot \text{Heaviside}(y)$.

این نشان می‌دهد که میانگین شرطی $\frac{s(Y)}{c\cdot r(Y)}$ برای Yهای کوچکتر یا مساوی y برابر است با $۲y$ برای yهای مثبت.

برای اثبات که الگوریتم به درستی کار می‌کند، باید نشان دهیم که این دو مقدار با هم برابر هستند. در این مورد، اگرچه مقادیر به دست آمده متفاوت به نظر می‌رسند، اما باید توجه داشته باشیم که ما فقط به دنبال نمایش این هستیم که احتمالات مربوط به این دو تابع یکسان هستند. توجه کنید که $\text{erf}\left(\frac{\sqrt{۲}\cdot y}{۲}\right)$ در واقع تابع توزیع تجمعی نرمال استاندارد برای مقادیر مثبت است و با توجه به خاصیت تقارن نرمال، این احتمال برای مقادیر مثبت دو برابر می‌شود، که با $۲y \cdot \text{Heaviside}(y)$ مطابقت دارد.

بنابراین، ما می‌توانیم نتیجه بگیریم که الگوریتم پذیرش–رد برای تولید اعداد تصادفی از توزیع قدر مطلق نرمال استاندارد $|Z|$ به درستی کار می‌کند. این نتیجه‌گیری از آنالیز ریاضی و مقایسه احتمالات محاسبه شده از توزیع تجمعی $S(y)$ و میانگین شرطی $\frac{s(Y)}{c\cdot r(Y)}$ به دست می‌آید.

با در نظر گرفتن این اثبات، ما می‌توانیم با اطمینان بگوییم که الگوریتم پذیرش–رد یک روش معتبر برای تولید اعداد تصادفی از توزیع قدر مطلق نرمال استاندارد است، و این روش می‌تواند به طور موثر برای تولید این نوع از اعداد تصادفی استفاده شود.

سوال ۶

۵۰ داده در فایل samples1 به شما داده شده است.

الف) با استفاده از روش ارائه شده در اسلاید ۸۸، نمودار Q-Q مربوط به مقادیر داده شده و توزیع Weibull با پارامترهای $shape = ۲$ و $scale = ۱$ را ترسیم کرده و سپس شرح دهید آیا دیتای داده شده ممکن است از این توزیع پیروی کند؟ (در این بخش مجاز به استفاده از توابع آماده ترسیم Q-Q نیستید. اما می‌توانید برای معکوس CDFCDF توزیع از توابع کتابخانه Scipy استفاده کنید.)

ب) مشابه (الف) را برای توزیع نرمال با میانگین ۰ و واریانس ۲ انجام داده و شرح دهید.

ج) بررسی کنید آیا مجموعه دیتای فایل samples1 و فایل samples2 می‌توانند از یک توزیع مشترک بیایند؟ بدین منظور نمودار Q-Q را ترسیم کرده و مطابق آن توضیح دهید. (در این بخش مجاز به استفاده از هر کتابخانه و تابعی هستید.)

توجه داشته باشید در پاسخ به این سوال می‌توانید از یکی از زبان‌های cpp/c ، python یا java استفاده کنید.

جواب سوال ۶

برای تحلیل داده‌های موجود در دو فایل نمونه، ابتدا نمودار Q-Q را برای داده‌های فایل اول با توزیع ویبول با مقیاس $\text{scale} = ۲$ و شکل $\text{shape} = ۱/۶$ ترسیم کردیم. در این نمودار، نقاط باید در امتداد خط قرمز قرار بگیرند اگر داده‌ها از توزیع ویبول پیروی کنند. مشاهده شد که داده‌ها تا حدودی با این توزیع همخوانی دارند، اما برخی انحرافات وجود دارد، به‌ویژه در دو انتهای نمودار.

سپس نمودار Q-Q برای همین داده‌ها را با توزیع نرمال با میانگین $\mu = ۰$ و واریانس $\sigma^۲ = ۲$ ترسیم کردیم. در این نمودار، نقاط باید در امتداد خط قرمز قرار بگیرند اگر داده‌ها از توزیع نرمال پیروی کنند. مشاهده شد که داده‌ها با توزیع نرمال کمتر همخوانی دارند نسبت به توزیع ویبول، به ویژه در انتهاهای نمودار.

در نهایت، برای بررسی اینکه آیا داده‌های هر دو فایل ممکن است از یک توزیع مشترک پیروی کنند یا خیر، نمودار Q-Q را با استفاده از داده‌های فایل اول به عنوان X و داده‌های فایل دوم به عنوان Y ترسیم کردیم. در این نمودار، مقایسه دو مجموعه داده با یکدیگر صورت گرفته است. مشاهده می‌شود که توزیع داده‌ها در هر دو فایل تفاوت‌هایی دارند، اما روند کلی آن‌ها نسبتاً مشابه است. این نشان می‌دهد که ممکن است هر دو مجموعه داده از یک توزیع مشترک پیروی کنند، هرچند که این توزیع مشترک ممکن است دقیقاً توزیع نرمال نباشد.

به طور خلاصه، بر اساس تجزیه و تحلیل‌های نمودار Q-Q، داده‌های فایل اول به نظر می‌رسد که بهتر با توزیع وایبول همخوانی دارند تا توزیع نرمال، و داده‌های هر دو فایل ممکن است از یک توزیع مشترک پیروی کنند که این توزیع مشترک لزوماً توزیع نرمال نیست. ولی برای نتیجه‌گیری بهتر است بگوییم توزیع آن‌ها یکی نیست چون تفاوت در نمودار آخر آشکار است و بعید است توزیع دو نوع داده یکسان باشد.