

سوال ۱

۱. به سوالات زیر پاسخ دهید:

- الف) مدل‌های صف با جمعیت نامحدود (infinite population) و جمعیت محدود (finite population) را توصیف کنید و تفاوت‌های آن‌ها را بیان نمایید.
- ب) انواع نظم‌دهی در صف‌ها یا نظام صف‌بندی (Queue discipline) چیست؟ سه مورد از این انواع را نام برده و هر کدام را به طور خلاصه شرح دهید.
- ج) جریان‌های شماره‌گذاری تصادفی (random-number streams) را توضیح دهید.

جواب سوال ۱

الف) مدل‌های صف با جمعیت نامحدود و جمعیت محدود: جمعیت نامحدود به معنای آن است که تعداد مشتریان بالقوه برای ورود به صف بی‌نهایت است، بدین معنی که همیشه مشتری جدیدی برای ورود به صف وجود دارد و محدودیتی برای تعداد کل مشتریان وجود ندارد. این مدل معمولاً در سیستم‌هایی با ترافیک بالا مانند وب‌سایت‌ها یا سیستم‌های تلفنی مورد استفاده قرار می‌گیرد. در مقابل، جمعیت محدود به وضعیتی اشاره دارد که تعداد مشتریانی که ممکن است وارد صف شوند محدود است. این مدل در موقعیت‌هایی که تعداد کاربران سیستم محدود است، مثلاً در یک کسب‌وکار کوچک یا سیستمی با تعداد محدود کاربر، کاربرد دارد.

ب) انواع نظم‌دهی در صف‌ها:

- **FIFO (First In, First Out):** این رویه بر اساس ترتیب ورود مشتریان به صف عمل می‌کند. اولین مشتری که وارد صف می‌شود، اولین کسی است که خدمات دریافت می‌کند. این روش در اکثر موقعیت‌های رایج مانند صف‌های بانک یا فروشگاه‌ها به کار می‌رود.
- **LIFO (Last In, First Out):** در این روش، آخرین مشتری که وارد صف می‌شود، اولین کسی است که خدمات دریافت می‌کند. این روش عمدتاً در موقعیت‌های خاص مانند برخی فرآیندهای صنعتی یا در مدیریت داده‌ها به کار می‌رود.
- **اولویت‌بندی:** در این روش، مشتریان بر اساس اولویت‌های مختلف خدمات دریافت می‌کنند. اولویت‌ها می‌توانند بر اساس فاکتورهای مختلفی مانند اورژانسی بودن، وضعیت ویژه یا اهمیت مشتری تعیین شوند. این روش در موقعیت‌هایی مانند بیمارستان‌ها یا مراکز خدماتی که نیاز به سرویس‌دهی فوری دارند، کاربرد دارد.

ج) **جریان‌های شماره‌گذاری تصادفی:** جریان‌های شماره‌گذاری تصادفی ابزارهایی هستند که برای تولید دنباله‌هایی از اعداد تصادفی در شبیه‌سازی‌ها استفاده می‌شوند. این اعداد به منظور مدل‌سازی رفتارهای تصادفی در سیستم‌های واقعی استفاده می‌شوند، مانند تعیین زمان‌های ورود تصادفی مشتریان به صف یا زمان‌های متفاوت سرویس‌دهی در یک مرکز خدماتی. استفاده از این جریان‌ها امکان پذیرش یک مدل شبیه‌سازی را که به خوبی پویایی‌های واقعی را بازتاب می‌دهد، فراهم می‌کند.

جواب سوال ۲

برای متغیر تصادفی X (تاس چهار وجهی):

- امید ریاضی X : $E[X] = \frac{1+2+3+4}{4} = 2.5$

- واریانس X : $Var[X] = E[X^2] - (E[X])^2 = 7.5 - (2.5)^2 = 1.25$

برای متغیر تصادفی Y (تاس شش وجهی):

- امید ریاضی Y : $E[Y] = \frac{1+2+3+4+5+6}{6} = 3.5$

- واریانس Y : $Var[Y] = E[Y^2] - (E[Y])^2 = \frac{91}{6} - (3.5)^2 = 2.9167$

برای متغیر تصادفی Z (میانگین X و Y):

- امید ریاضی Z : $E[Z] = \frac{E[X]+E[Y]}{2} = 3$

- واریانس Z : $Var[Z] = \frac{Var[X]+Var[Y]}{4} = 1.0417$

(ب) امید ریاضی سود شما پس از ۶۰ دست پرتاب دو تاس:

فرض کنید در هر پرتاب دو تاس، اگر عدد نمایش داده شده روی تاس چهار وجهی (X) بیشتر از عدد نمایش داده شده روی تاس شش وجهی (Y) باشد، شما $2 \times X$ دلار برنده می‌شوید. در غیر این صورت، شما ۱ دلار می‌بازید. محاسبه امید ریاضی سود شما برای یک پرتاب و سپس برای ۶۰ پرتاب به شرح زیر است:

- محاسبه امید ریاضی سود برای $X > Y$:

— احتمال $X > Y$ برابر است با تعداد حالاتی که X بزرگتر از Y است تقسیم بر تعداد کل حالات.

— در هر حالت که $X > Y$ ، سود $2 \times X$ دلار است.

- محاسبه امید ریاضی زیان برای $X \leq Y$:

— احتمال $X \leq Y$ برابر است با تعداد حالاتی که X کمتر یا مساوی Y است تقسیم بر تعداد کل حالات.

— در هر حالت که $X \leq Y$ ، زیان ۱ دلار است.

- امید ریاضی سود برای یک پرتاب دو تاس: تقریباً ۰/۹۱۷ دلار.

- امید ریاضی کلی سود برای ۶۰ پرتاب: تقریباً ۵۵ دلار.

جواب سوال ۳

الف) نشان دهید:

$$\sum_{j=1}^{\infty} (a_1 + a_2 + \dots + a_j) P(N = j) = \sum_{i=1}^{\infty} a_i P(N \geq i)$$

برای این بخش، ابتدا مجموعه داخل پرانتز را به صورت تجمعی نویسیم و سپس با تغییر ترتیب جمع‌زنی، این را به صورت زیر بیان می‌کنیم:

$$= \sum_{i=1}^{\infty} a_i \sum_{j=i}^{\infty} P(N = j)$$

که در آن $\sum_{j=i}^{\infty} P(N = j)$ معادل احتمال این است که N حداقل برابر با i باشد. در نتیجه، به معادله نهایی می‌رسیم:

$$= \sum_{i=1}^{\infty} a_i P(N \geq i)$$

ب) ثابت کنید:

$$E[N] = \sum_{j=1}^{\infty} P(N \geq j)$$

برای این بخش، از تعریف امید ریاضی برای متغیر تصادفی گسسته استفاده می‌کنیم. امید ریاضی $E[N]$ برابر است با مجموع وزن‌دار تمام مقادیر ممکن N ، که به صورت مجموعه احتمالات N بزرگتر یا مساوی با هر عدد طبیعی j است.

ج) ثابت کنید:

$$E[N(N+1)] = \sum_{j=1}^{\infty} j \cdot P(N \geq j)$$

برای این بخش، ما توجه می‌کنیم که $N(N+1)$ می‌تواند به صورت $N^2 + N$ بیان شود. این بیان به ما امکان می‌دهد تا امید ریاضی $E[N(N+1)]$ را به صورت مجموعه‌ای از احتمالات بیان کنیم که در آن N حداقل برابر با هر عدد طبیعی j است.

بنابراین، ما می‌توانیم امید ریاضی $E[N(N+1)]$ را به صورت زیر بنویسیم:

$$E[N(N+1)] = E[N^2] + E[N]$$

و از آنجا که $E[N^2]$ را می‌توان به صورت مجموعه‌ای از احتمالات که N حداقل برابر با هر عدد طبیعی j است بیان کرد، می‌توانیم آن را به صورت زیر بیان کنیم:

$$E[N^2] = \sum_{j=1}^{\infty} j^2 \cdot P(N \geq j)$$

و از آنجا که $E[N]$ نیز به صورت مجموعه‌ای از احتمالات بیان می‌شود:

$$E[N] = \sum_{j=1}^{\infty} j \cdot P(N \geq j)$$

بنابراین، امید ریاضی $E[N(N+1)]$ را می‌توان به صورت زیر محاسبه کرد:

$$E[N(N+1)] = \sum_{j=1}^{\infty} j^2 \cdot P(N \geq j) + \sum_{j=1}^{\infty} j \cdot P(N \geq j)$$

که در نهایت به صورت $\sum_{j=1}^{\infty} j \cdot P(N \geq j)$ بیان می‌شود.

جواب سوال ۴

محاسبه امید ریاضی تعداد دفعاتی که باید تاس انداخته شود تا توالی عدد مضرب ۳ نباشد - عدد مضرب ۳ باشد - عدد مضرب ۳ نباشد مشاهده شود:

برای حل این مسئله، ما از نظریه زنجیره‌های مارکوف استفاده می‌کنیم. سه حالت ممکن برای هر پرتاب تاس وجود دارد:

الف) حالت A : هیچ عددی تاکنون پرتاب نشده یا آخرین عدد پرتاب شده مضرب ۳ نبوده است.

ب) حالت B : آخرین عدد پرتاب شده مضرب ۳ بوده است.

ج) حالت C : توالی کامل شده است (یعنی عدد مضرب ۳ نباشد - عدد مضرب ۳ باشد - عدد مضرب ۳ نباشد).

ماتریس انتقال P به صورت زیر است:

$$P = \begin{bmatrix} \frac{4}{6} & \frac{2}{6} & 0 \\ \frac{1}{6} & 0 & \frac{5}{6} \\ 0 & 0 & 1 \end{bmatrix}$$

این ماتریس نشان‌دهنده احتمال جابه‌جایی بین حالت‌ها است.

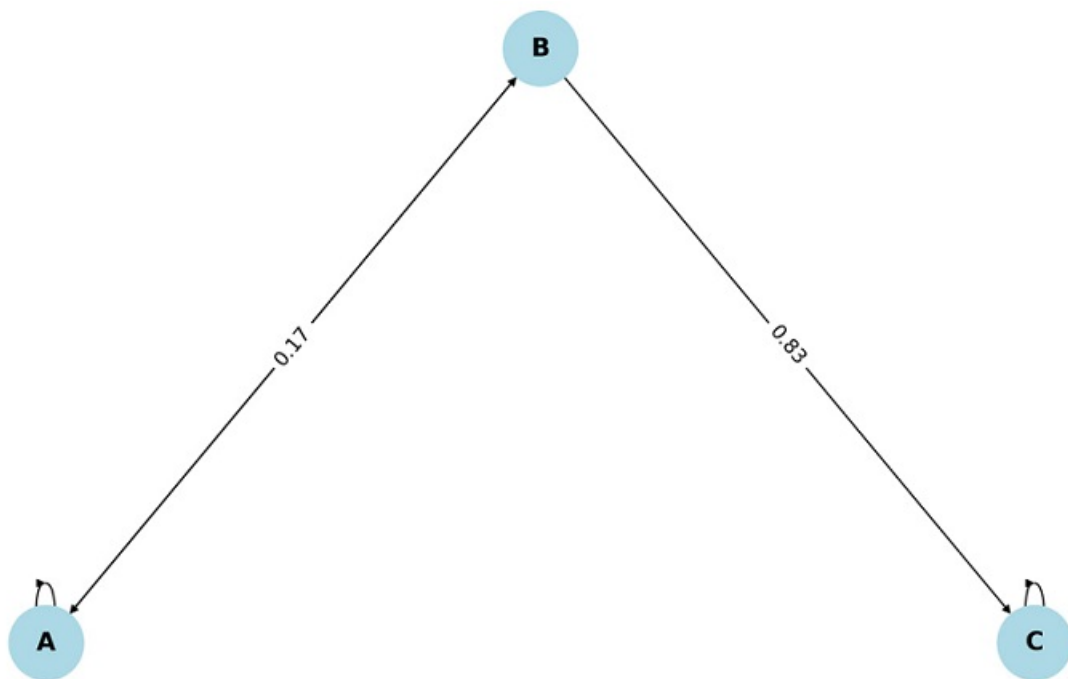
برای محاسبه امید ریاضی تعداد دفعات پرتاب تاس تا رسیدن به حالت C بعد از حالت A ، ما از خواص زنجیره‌های مارکوف استفاده می‌کنیم. این محاسبه شامل محاسبه زمان اولین ورود به حالت C از حالت A است.

در نمودار زنجیره مارکوف که برای مسئله داده شده رسم شده است، گره‌ها حالت‌های مختلف زنجیره را نشان می‌دهند و یال‌ها احتمال انتقال بین حالت‌ها را نمایش می‌دهند.

• حالت A : این حالت نشان‌دهنده شروع توالی یا پرتاب عددی غیر مضرب ۳ است.

• حالت B : این حالت نشان‌دهنده پرتاب عدد مضرب ۳ است.

• حالت C : این حالت نشان‌دهنده اتمام توالی مورد نظر ما (عدد مضرب ۳ نباشد - عدد مضرب ۳ باشد - عدد مضرب ۳ نباشد) است.

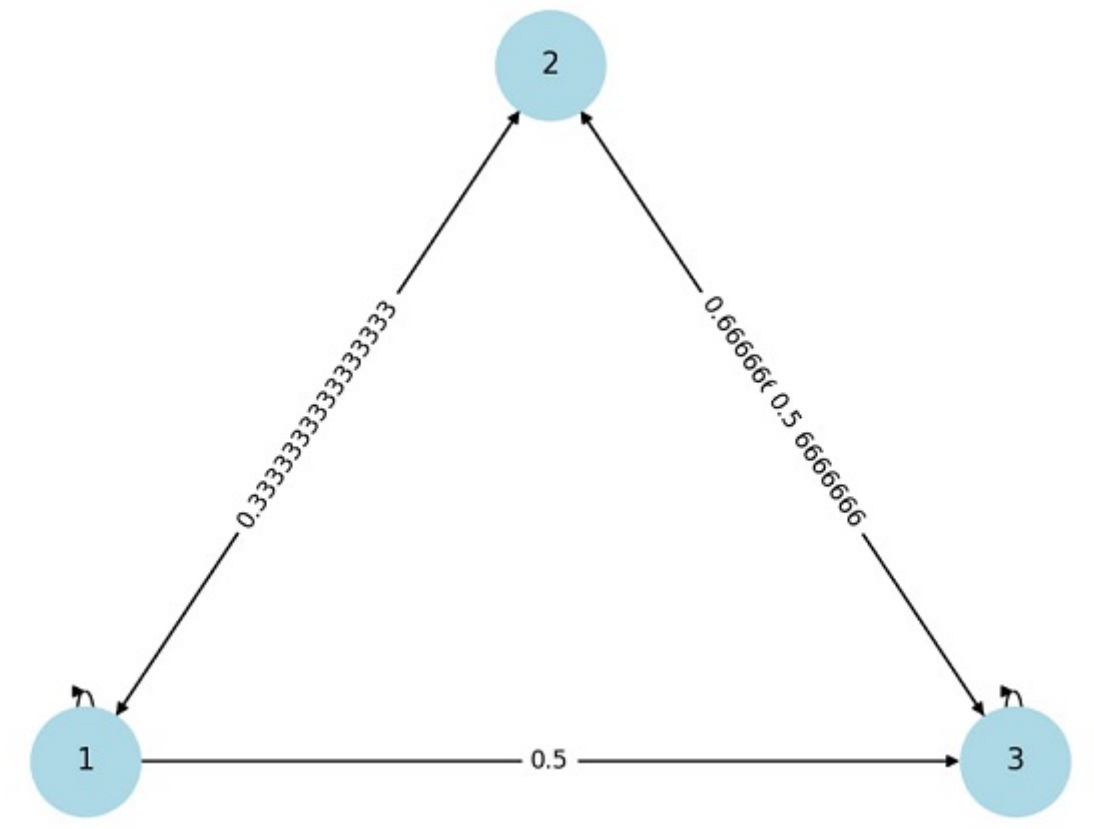


جواب سوال ۵

برای یک زنجیره مارکوف با استیت‌های $\{۱, ۲, ۳\}$ و ماتریس انتقال زیر:

$$P = \begin{pmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \\ \frac{1}{3} & 0 & \frac{2}{3} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

نمودار زنجیره مارکوف مربوط به این ماتریس به صورت زیر است:



محاسبه احتمال $P(X_1 = 3, X_2 = 2, X_3 = 1)$ بر اساس قانون زنجیره‌ای احتمال انجام می‌شود:

$$P(X_1 = 3, X_2 = 2, X_3 = 1) = P(X_1 = 3) \times P_{32} \times P_{21}$$

از آنجا که $P(X_1 = 3) = \frac{1}{3}$ و $P(X_1 = 1) = P(X_1 = 2) = \frac{1}{3}$ محاسبه صحیح به صورت زیر خواهد بود:

$$P(X_1 = 3, X_2 = 2, X_3 = 1) = \frac{1}{3} \times \frac{1}{3} \times \frac{1}{3} = \frac{1}{27}$$

جواب سوال ۶

توضیح کد پایتون برای زنجیره مارکوف

این کد پایتون برای مدل‌سازی و تحلیل یک زنجیره مارکوف گسسته استفاده می‌شود که در آن تعدادی حالت و احتمالات انتقال بین آنها وجود دارد.

تابع `calculate_state_probability`

این تابع برای محاسبه احتمال یک حالت خاص (s) در روز معین (N) در زنجیره مارکوف استفاده می‌شود. این تابع بردار احتمال اولیه (p_0) و ماتریس انتقال را به عنوان ورودی می‌گیرد و با استفاده از حلقه‌ای برای ضرب ماتریس انتقال در بردار احتمال فعلی در هر مرحله، احتمال حالت s در روز N را محاسبه می‌کند.

تعریف بردار احتمال اولیه و ماتریس انتقال

این بخش از کد بردار احتمال اولیه و ماتریس انتقال را تعریف می‌کند که برای مدل‌سازی زنجیره مارکوف استفاده می‌شوند.

محاسبه احتمال انتخاب جوجه کباب در روز ۱۷۷۷ام

با استفاده از تابع `calculate_state_probability`، احتمال انتخاب جوجه کباب در روز ۱۷۷۷ام محاسبه می‌شود.

محاسبه احتمالات انتخاب جوجه کباب در روزهای مختلف و نمایش نمودار

این بخش از کد احتمال انتخاب جوجه کباب در روزهای مختلف را محاسبه کرده و سپس این احتمالات را در یک نمودار نشان می‌دهد.

تابع `simulate_markov_chain`

این تابع برای شبیه‌سازی روند زنجیره مارکوف استفاده می‌شود. با تعداد دفعات مشخص شده، این تابع یک زنجیره مارکوف را شبیه‌سازی کرده و احتمالات حالت‌های مختلف در پایان دوره را محاسبه می‌کند.

مقایسه نتایج محاسباتی و شبیه‌سازی

در این بخش، نتایج به دست آمده از محاسبات تئوری و شبیه‌سازی مقایسه می‌شوند. این مقایسه شامل بررسی خطای بین نتایج تئوری و شبیه‌سازی است.