Disciplina: Estatística Computacional	**(Entrega: 16/10/19)**
Lista de Exercícios 1	
Professor(a): Eduardo Mendes	**Aluno:** Franklin Alves de Oliveira

Exercise 1 (Inversion and Rejection)

$\boxed{\textbf{1.}}$ Let $Y \sim \mathcal{E}xp(\lambda)$ and $a > 0$. We consider the variable after restricting its support to be $[a, +\infty)$. That is, let $X = Y \mid Y \geq a$, i.e., X has the law of Y conditionally on being in $[a, +\infty)$. Calculate $F_X(x)$, the cumulative distribution function of X, and $F_X^{-1}(u)$, the quantile function of X. Describe an algorithm to simulate X from $U \sim \mathcal{U}_{[0,1]}$.

Solução:

Para resolver essa questão, vamos tomar como base o resultado do seguinte teorema:

> **Teorema Fundamental da Simulação:** Seja F uma CDF contínua e estritamente crescente. Então, existe $F^{-1} : [0, 1] \to \mathbb{R}$ e valem os seguintes resultados:
>
> 1. $U \sim \mathcal{U}_{(0,1)}$ e $X = F^{-1}(U) \;\Rightarrow\; X$ é uma variável aleatória com CDF F.
> 2. Se X é uma variável aleatória contínua com CDF F, então $F(X) \sim \mathcal{U}_{[0,1]}$.

Assim, se Y é uma v.a. tal que $Y \sim \mathcal{E}xp(\lambda)$, então Y tem PDF $f_Y(y) = \lambda e^{-\lambda y}$, com $supp(Y) = [0, +\infty)$. Definindo $X = Y \mid Y \geq a$, onde $a > 0$, temos:

$$F_X(x) = P(X \leq x) = P(Y \leq x \mid Y \geq a) \tag{1}$$

Aplicando a Regra de Bayes, podemos reescrever 1 da seguinte forma:

$$F_X(x) = P(Y \leq x \mid Y \geq a) = \frac{P(a \leq Y \leq x)}{P(Y \geq a)} = \frac{\int_a^x \lambda e^{-\lambda y} dy}{\int_0^{+\infty} \lambda e^{-\lambda y} dy} =$$
$$\frac{-e^{-\lambda x} + e^{-\lambda a}}{-e^{-\lambda a}} = \frac{e^{-\lambda a} - e^{-\lambda x}}{e^{-\lambda a}} = 1 - \frac{e^{-\lambda x}}{e^{-\lambda a}} = 1 - e^{-\lambda(x-a)} \tag{2}$$

Pelo **Teorema Fundamental da Simulação**, sabemos que $F_x(x) = U \sim \mathcal{U}_{[0,1]}$. Então, podemos "simular"X fazendo $X = F_x^{-1}(U)$. Ainda, podemos determinar F_X^{-1} resolvendo a seguinte equação para x:

$$F_x(x) = U = 1 - e^{-\lambda(x-a)} \tag{3}$$

Assim,

$$U - 1 = -e^{-\lambda(x-a)} \Rightarrow 1 - U = e^{-\lambda(x-a)} \Rightarrow$$
$$\Rightarrow \log(1-U) = -\lambda(x-a) \Rightarrow -\frac{\log(1-U)}{\lambda} = x - a \Rightarrow x = a - \frac{\log(1-U)}{\lambda} \tag{4}$$

Portanto, definimos $F_X^{-1}(U) = a - \frac{\log(1-U)}{\lambda}$. Logo, para simular X, tome os seguintes passos:

1. Simule $U \sim \mathcal{U}_{[0,1]}$.
2. A partir do valor gerado de U, determine $X = a - \frac{\log(1-U)}{\lambda}$.

$\boxed{\textbf{2.}}$ Let a and b be given, with $a < b$. Show that we can simulate $X = Y \mid a \leq Y \leq b$ from $U \sim \mathcal{U}_{[0,1]}$ using

$$X = F_Y^{-1}\left(F_Y(a)(1-U) + F_Y(b)U\right),$$

i.e., show that if X is given by the formula above, then $\Pr(X \leq x) = \Pr(Y \leq x \mid a \leq Y \leq b)$. Apply the formula to simulate an exponential random variable conditioned to be greater than a, as in the previous question.

Solução:

Sejam a e b dados, com $a < b$. Se X é uma v.a. tal que

$$X = F_Y^{-1}\left(F_Y(a)(1-U) + F_Y(b)U\right), \text{ então}$$

$$F_X(x) = P(X \leq x) = P\left(F_Y^{-1}\left(F_Y(a)(1-U) + F_Y(b)U\right) \leq x\right) \tag{5}$$

Aplicando F_Y em ambos os lados da desigualdade em 5, temos:

$$F_X(x) = P(X \leq x) = P\left(\left(F_Y(a)(1-U) + F_Y(b)U\right) \leq F_Y(x)\right) =$$
$$P\left(F_Y(a) - F_Y(a)U + F_Y(b)U \leq F_Y(x)\right) = P\left((F_Y(b) - F_Y(a))U \leq F_Y(x) - F_Y(a)\right) =$$
$$P\left(U \leq \frac{F_Y(x) - F_Y(a)}{F_Y(b) - F_Y(a)}\right) = \frac{F_Y(x) - F_Y(a)}{F_Y(b) - F_Y(a)}. \tag{6}$$

Agora, vamos aplicar essa fórmula no caso em que $Y \sim \mathcal{E}xp(\lambda)$. Assim, Y tem PDF $f_Y(y) = \lambda e^{-\lambda y}$, e sua CDF é:

$$F_Y(y) = \int_0^{+\infty} \lambda e^{-\lambda y} dy = 1 - e^{-\lambda y}$$

Se condicionarmos Y a $Y > a > 0$, fizermos $b \to +\infty$ de forma que $F_Y(b) = 1$ e, ainda, tomando $U \sim \mathcal{U}_{[0,1]}$, temos:

$$F_Y(a)(1-U) + F_Y(b)U = (1 - e^{-\lambda a})(1-U) + U$$

e, portanto,

$$X = F_Y^{-1}\left(F_Y(a)(1-U) + F_Y(b)U\right) = F_Y^{-1}\left((1 - e^{-\lambda a})(1-U) + U\right) \tag{7}$$

Sabemos que, se $A = F_Y(y) = 1 - e^{-\lambda y}$, então $F_Y^{-1} = -\frac{\log(1-A)}{\lambda}$. Assim, podemos reescrever 7 como:

$$X = -\lambda^{-1}\log\left(1 - [F_Y(a)(1-U) + F_Y(b)U]\right) = -\lambda^{-1}\log\left(1 - \left[(1 - e^{-\lambda a})(1-U) + U\right]\right) =$$

$$= -\lambda^{-1}\log\left(1 - \left(1 - e^{-\lambda a} - U + Ue^{-\lambda a} + U\right)\right) = -\lambda^{-1}\log e^{-\lambda a}(1-U) =$$

$$= -\lambda^{-1}(-\lambda a) + -\lambda^{-1}\log(1-U) = a - \lambda^{-1}\log(1-U) \tag{8}$$

como já havíamos encontrado na questão 1.

3. Here is a simple algorithm to simulate $X = Y \mid Y > a$ for $Y \sim \mathcal{E}xp(\lambda)$:

(a) Let $Y \sim \mathcal{E}xp(\lambda)$. Simulate $Y = y$.

(b) If $Y > a$ then stop and return $X = y$, and otherwise, start again at step (a).

Show that this is just a rejection algorithm, by writing the proposal and target densities π and q, as well as the bound $M = \max_x \pi(x)/q(x)$. Calculate the expected number of trials to the first acceptance. Why is inversion to be preferred for $a \gg 1/\lambda$?

Solução:

Defina a variável aleatória $Y \sim \mathcal{E}xp(\lambda)$. Como, no primeiro passo do algoritmo, estamos amostrando de uma exponencial com parâmetro λ, a nossa distribuição proposta é a PDF de Y, i.e.,

$$q(y) = \lambda e^{-\lambda y}, \text{ onde } y \geq 0.$$

Em particular, estamos interessados nos casos em que $y \geq a > 0$. Logo, nossa distribuição alvo (*Target*) é:

$$\pi(y) = \lambda e^{-\lambda(y-a)} \mathbb{1}_{y \geq a}$$

Note que $\pi(y) = \lambda e^{-\lambda(y-a)} \mathbb{1}_{y \geq a} \leq q(y) = \lambda e^{-\lambda y}$, para todo $y \geq 0$. Não obstante, definindo $M = \max_x \pi(x)/q(x)$, onde $X = Y \mid Y \geq a$.

Note que o valor máximo da razão $\pi(x)/q(x)$ acontece quando $\pi(x)$ é máximo e $q(x)$ é mínimo. Porém, tanto $\pi(x)$ quanto $q(x)$ são decrescentes em x, sendo que $q(x)$ decresce mais rápido com um incremento em x do que $\pi(x)$. Logo, o máximo da razão entre as duas funções acontece quando o valor de x é mínimo. Como o supporte de X é $[a, +\infty)$, o máximo de $\pi(x)/q(x)$ acontece quando $X = a$. Assim,

$$M = \max_x \pi(x)/q(x) = \pi(a)/q(a) = \lambda e^{-\lambda(a-a)}/\lambda e^{-\lambda a} = e^{\lambda a}.$$

Denotando por α a probabilidade de aceitação do algoritmo, temos

$$\alpha = \frac{\pi(y)}{Mq(y)} = \begin{cases} 1 & \text{se } y \geq a \\ 0 & \text{caso contrário} \end{cases} \tag{9}$$

Isso quer dizer que, segundo esse algoritmo, sempre aceitamos se $y \geq a$.

Por definição, o número de tentativas até o primeiro sucesso (n) segue uma distribuição geométrica(p), onde p é a probabilidade de sucesso, i.e., $p = P(Y \geq a)$. Logo,

$$p = P(Y \geq a) = 1 - P(Y < a) = 1 - \int_0^a \lambda e^{-\lambda y} dy = 1 - \left[-e^{-\lambda y}\right]_0^a = 1 - (1 - e^{-\lambda a}) = e^{-\lambda a}.$$

Assim, $\mathbb{E}[n] = 1/p = e^{\lambda a}$.

Exercise 2 (Rejection)

Consider the following "squeeze" rejection algorithm for sampling from a distribution with density $\pi(x) = \tilde{\pi}(x)/Z_\pi$ on a state space $\mathbb{X}$ such that

$$h(x) \le \tilde{\pi}(x) \le M\tilde{q}(x)$$

where h is a non-negative function, $M > 0$ and $q(x) = \tilde{q}(x)/Z_q$ is the density of a distribution that we can easily sample from. The algorithm proceeds as follows:

- **(a)** Draw independently $X \sim q$, $U \sim \mathcal{U}_{[0,1]}$.
- **(b)** Accept X if $U \le h(X)/(M\tilde{q}(X))$.
- **(c)** If X was not accepted in step (b), draw an independent $V \sim \mathcal{U}_{[0,1]}$ and accept X if .

$$V \le \frac{\tilde{\pi}(X) - h(X)}{M\tilde{q}(X) - h(X)}$$

$\boxed{\textbf{1.}}$ Show that the probability of accepting a proposed $X = x$ in either step (b) or (c) is

$$\frac{\tilde{\pi}(x)}{M\tilde{q}(x)}.$$

Solução:

Ora,

$$P(\text{aceitar } x \text{ no passo (b)}) = \int_0^1 \mathbb{1}_{\left(U \le \frac{h(x)}{M\tilde{q}(x)}\right)} dU = \frac{h(x)}{M\tilde{q}(x)} \tag{10}$$

Analogamente,

$$\begin{aligned} P(\text{aceitar } x \text{ no passo (c)}) &= \int_0^1 \int_0^1 \mathbb{1}_{\left(U > \frac{h(x)}{M\tilde{q}(x)}\right)} \mathbb{1}_{\left(V \le \frac{\tilde{\pi}(X) - h(X)}{M\tilde{q}(X) - h(X)}\right)} dU dV = \\ &= \left(1 - \frac{h(x)}{M\tilde{q}(x)}\right)\left(\frac{\tilde{\pi}(X) - h(X)}{M\tilde{q}(X) - h(X)}\right) = \frac{\tilde{\pi}(x) - h(x)}{M\tilde{q}(x)} \end{aligned} \tag{11}$$

□

$\boxed{\textbf{2.}}$ Deduce from the previous question that the distribution of the samples accepted by the above algorithm is π.

Solução:

Considerando $A \subset \mathbb{X}$ um conjunto mensurável, pelo método de *Rejection Sampling*, temos:

$$P(X \in A \mid X \text{ aceito}) = \frac{P(X \in A, X \text{ aceito})}{P(X \text{ aceito})}$$

Note que,

$$P(X \in A, X \text{ aceito}) = \int_{\mathbb{X}} \int_0^1 \mathbb{1}_{(x \in A)} \mathbb{1}_{\left(u \le \frac{\pi(x)}{M \cdot q(x)}\right)} q(x) du dx =$$

$$= \int_{\mathbb{X}} \mathbb{1}_{(x\in A)} \frac{\pi(x)}{M\cdot q(x)} q(x)dx = \int_X \mathbb{1}_{(x\in A)} \frac{\pi(x)}{M} dx = \frac{\pi(A)}{M}. \tag{12}$$

Ainda,

$$P(X \text{ aceito}) = P(X \in \mathbb{X}, X \text{ aceito}) = \frac{\pi(\mathbb{X})}{M} = \frac{1}{M}. \tag{13}$$

Combinando os resultados em 12 e 13, temos:

$$P(X \in A \mid X \text{ aceito}) = \frac{\pi(A)/M}{1/M} = \pi(A). \tag{14}$$

Com isso, concluímos que a distribuição dos valores aceitos é, exatamente, π.

3. Show that the probability that step (c) has to be carried out is

$$1 - \frac{\int_{\mathbb{X}} h(x)dx}{MZ_q}$$

Solução:

Ora,

$$P(\text{passo (c) ser executado}) = P(X \text{ ser rejeitado no passo (b)}) = \int_0^1 \int_{\mathbb{X}} \mathbb{1}_{\left(U > \frac{h(x)}{M\tilde{q}(x)}\right)} q(x)dxdU$$

$$= \int_{\mathbb{X}} \left(1 - \frac{h(x)}{M\tilde{q}(x)}\right) q(x)dx = \int_{\mathbb{X}} q(x) - q(x)\frac{h(x)}{M\tilde{q}(x)}dx = \int_{\mathbb{X}} q(x)dx - \int_{\mathbb{X}} q(x)\frac{h(x)}{M\tilde{q}(x)}dx =$$

$$= 1 - \int_{\mathbb{X}} \frac{h(x)}{MZ_q} dx, \tag{15}$$

Dado que $q(x) = \tilde{q}(x)/Z_q$.

4. Let $\tilde{\pi}(x) = \exp\left(-x^2/2\right)$ and $\tilde{q}(x) = \exp\left(-|x|\right)$. Using the fact that

$$\tilde{\pi}(x) \geq 1 - \frac{x^2}{2}$$

for any $x \in \mathbb{R}$, how could you use the squeeze rejection sampling algorithm to sample from $\pi(x)$. What is the probability of not having to evaluate $\tilde{\pi}(x)$? Why could it be beneficial to use this algorithm instead of the standard rejection sampling procedure?

Solução:

Para aplicar o método *Squeeze Rejection Sampling*, precisamos que a seguinte relação seja válida:

$$h(x) \leq \tilde{\pi}(x) \leq M\tilde{q}(x) \tag{16}$$

Onde $h(x)$ é uma função não-negativa, i.e., $x \leq \pm\sqrt{2}$. O enunciado nos sugere que $\tilde{\pi}(x) \geq 1 - x^2/2$, $\forall\ x \in \mathbb{R}$. Então, convenientemente, podemos definir $h(x) = 1 - x^2/2$. Para definir o valor de M, tome:

$$M = \sup_{x\in\mathbb{X}} \widetilde{\pi}(x)/\widetilde{q}(x) = \sup_{x\in\mathbb{X}} \exp\left(-x^2/2\right)\exp\left(-|x|\right)^{-1} = \sup_{x\in\mathbb{X}} \exp\left(-x^2/2 + |x|\right) =$$

Para $x \geq 0$, temos:

$$M = \sup_{x\in\mathbb{X}} \exp\left(-x^2/2 + x\right) = \exp\left(\max_{x\in\mathbb{X}}\left(-x^2/2 + x\right)\right)$$

Aplicando a condição de primeira ordem para maximizar a função $-x^2/2 + x$, encontramos:

$$\frac{d\left(-x^2/2 + x\right)}{dx} = -x + 1 = 0 \iff x = 1$$

Logo,

$$M = \exp\left(-(1^2)/2 + 1\right) = \sqrt{e} \tag{17}$$

Fazendo as devidas substituições em 16, obtemos:

$$0 \leq 1 - x^2/2 \leq \tilde{\pi}(x) \leq \sqrt{e}\ \tilde{q}(x) \tag{18}$$

Assim, podemos aplicar o método *Squeeze Rejection Sampling* da seguinte forma:

(a) Simulamos $U \sim \mathcal{U}_{[0,1]}$ e $X \sim \tilde{q}$

(b) Verificamos se $u \leq \frac{h(x)}{M\tilde{q}(x)}$, i.e., $u \leq \frac{1-x^2/2}{\sqrt{e}\tilde{q}(x)}$. Se sim, aceitamos X. Caso contrário, seguimos para o passo (c).

(c) Geramos $v \sim \mathcal{U}_{[0,1]}$ e aceitamos X se $v \leq \frac{\tilde{\pi}(x)-h(x)}{M\tilde{q}(x)-h(x)} = \frac{\tilde{\pi}(x)-(1-x^2/2)}{\sqrt{e}\ \tilde{q}(x)-(1-x^2/2)}$.

Lembrando que não teremos que avaliar $\tilde{\pi}(x)$ se X for aceito no passo (b). Então,

$$P(\text{não avaliar } \tilde{\pi}(x)) = P(\text{passo (c) não ser executado}) = 1 - \frac{\int_X h(x)dx}{MZ_q}, \tag{19}$$

Como já havíamos mostrado no item 3. Por definição, $Z_q = \int_X \tilde{q}(x)dx$. Logo,

$$Z_q = \int_X \exp(-|x|)dx = 2\int_0^{+\infty} exp(-x)dx = 2\left[-e^{-x}\right]_0^{+\infty} = 2\left[0 - (-1)\right] = 2 \tag{20}$$

Agora, fazendo as devidas substituições em 19, temos:

$$P(\text{não avaliar } \tilde{\pi}(x)) = 1 - \frac{\int_{-\sqrt{2}}^{\sqrt{2}} 1 - x^2/2\ dx}{\sqrt{e}\ 2} = 1 - \frac{\left[x - x^3/6\right]_{-\sqrt{2}}^{\sqrt{2}}}{\sqrt{e}\ 2} = 1 - \frac{2\sqrt{2} - 2\sqrt{2}^3/6}{\sqrt{e}\ 2} =$$

$$= 1 - \frac{2\sqrt{2} - 2\sqrt{2}/3}{\sqrt{e}\ 2} = 1 - \frac{6\sqrt{2}/3 - 2\sqrt{2}/3}{\sqrt{e}\ 2} = 1 - \frac{4\sqrt{2}/3}{\sqrt{e}\ 2} = \frac{\sqrt{e} - 2\sqrt{2}/3}{\sqrt{e}} \tag{21}$$

□

Vale destacar que, como $h(x)$ é computacionalmente menos custosa de avaliar do que $\tilde{\pi}(x)$, em média, esse método terá menos custo computacional do que o *Rejection Sampling* tradicional.

Exercise 3 (Transformation)

Consider the following algorithm known as Marsaglia's polar method.

- **Step a:** Generate independent U_1, U_2 according to $\mathcal{U}_{[-1,1]}$ until $Y = U_1^2 + U_2^2 \leq 1$.
- **Steb b:** Define

$$Z = \sqrt{-2\log(Y)}$$

and return

$$X_1 = Z\frac{U_1}{\sqrt{Y}},\ X_2 = Z\frac{U_2}{\sqrt{Y}}.$$

$\boxed{\textbf{1.}}$ Define $\vartheta = \arctan(U_2/U_1)$. Show that the joint distribution of Y and ϑ has density

$$f_{Y,\vartheta}(y,\theta) = \mathbb{1}_{[0,1]}(y)\frac{\mathbb{1}_{[-\frac{\pi}{2},\frac{\pi}{2}]}(\theta)}{2\pi}$$

Solução:

Considere $U_1,\ U_2 \sim \mathcal{U}_{[-1,1]}$ independentes. Agora, defina $Y = U_1^2 + U_2^2 \leq 1$ e $\vartheta = \arctan(U_2/U_1)$. Podemos definir a transformação $T : [-1,1] \times [-1,1] \to [0,1] \times \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ como:

$$T(U_1, U_2) = \left(U_1^2 + U_2^2,\ \arctan(U_2/U_1)\right)$$

É fácil verificar que a função arctan estritamente crescente e, portanto, inversível. Como, para quaisquer pares distintos $(\overline{U}_1, \overline{U}_2)$ e $(\widetilde{U}_1, \widetilde{U}_2)$, $T(\overline{U}_1, \overline{U}_2) \neq T(\widetilde{U}_1, \widetilde{U}_2)$ (é garantido que ao menos as segundas coordenadas de $(U_1^2 + U_2^2,\ \arctan(U_2/U_1))$ serão distintas), concluímos que a transformação T é inversível. Agora, vamos determinar a transformação inversa T^{-1}.

Faremos isso resolvendo o seguinte sistema para U_1 e U_2:

$$\begin{cases} Y = U_1^2 + U_2^2 \\ \vartheta = \arctan(U_1/U_2) \end{cases} \tag{22}$$

Da segunda equação, encontramos:

$$\tan(\vartheta) = \frac{U_2}{U_1} \Rightarrow U_2 = U_1\tan(\vartheta) \tag{23}$$

Substituindo o resultado de 23 na primeira equação de 22, encontramos:

$$Y = U_1^2 + U_1^2\tan^2(\vartheta) = U_1^2\left(1 + \tan^2(\vartheta)\right) = U_1^2\left(1 + \frac{\sin^2(\vartheta)}{\cos^2(\vartheta)}\right) = U_1^2\left(\frac{\cos^2(\vartheta) + \sin^2(\vartheta)}{\cos^2(\vartheta)}\right) \Rightarrow$$

$$\Rightarrow Y = \frac{U_1^2}{\cos^2(\vartheta)} \Rightarrow \cos(\vartheta)\sqrt{Y} = U_1 \tag{24}$$

Finalmente, substituindo 24 em 23, obtemos:

$$U_2 = \tan(\vartheta)\cos(\vartheta)\sqrt{Y} = \sin(\vartheta)\sqrt{Y} \tag{25}$$

Dos resultados 24 e 25, definimos T^{-1} como:

$$T^{-1}(y,\vartheta) = \left(\cos(\vartheta)\sqrt{Y}, \sin(\vartheta)\sqrt{Y}\right)$$

A partir deste ponto, estamos prontos para encontrar a PDF conjunta de Y e ϑ pelo método da transformação.

$$f_{Y,\vartheta}(y,\theta) = f_{U_1,U_2}\left(T^-1(y,\theta)\right)|det\left(\mathcal{J}_{T^{-1}}\right)| \tag{26}$$

Onde $\mathcal{J}_{T^{-1}}$ denota a matriz Jacobiana de T^{-1}. Assim,

$$|det\left(\mathcal{J}_{T^{-1}}\right)| = \left|\det\left(\frac{\partial\left(U_1,U_2\right)}{\partial(Y,\vartheta)}\right)\right| = \left|\det\begin{pmatrix} \frac{\partial U_1}{\partial Y} & \frac{\partial U_1}{\partial \vartheta} \\ \frac{\partial U_2}{\partial Y} & \frac{\partial U_2}{\partial \vartheta}\end{pmatrix}\right| = \left|\det\begin{pmatrix} \frac{\cos\vartheta}{2\sqrt{Y}} & -\sqrt{Y}\sin\vartheta \\ \frac{\sin\vartheta}{2\sqrt{Y}} & \sqrt{Y}\cos\vartheta\end{pmatrix}\right| \Rightarrow$$

$$\Rightarrow |det\left(\mathcal{J}_{T^{-1}}\right)| = \left|\frac{\cos\vartheta}{2\sqrt{Y}}\sqrt{Y}\cos(\vartheta) + \frac{\sin(\vartheta)}{2\sqrt{Y}}\sqrt{Y}\sin(\vartheta)\right| = \frac{1}{2}. \tag{27}$$

Portanto,

$$f_{Y,\vartheta}(y,\theta) = f_{U_1,U_2}\left(T^{-1}(y,\theta)\right)\cdot\frac{1}{2} = f_{U_1,U_2}\left(\cos(\vartheta)\sqrt{Y},\sin(\vartheta)\sqrt{Y}\right)\cdot\frac{1}{2} = \mathbb{1}_{y\in[0,1]}\frac{\mathbb{1}_{\theta\in[0,2\pi]}}{2\pi} \tag{28}$$

Uma vez que $\mathbb{1}_{y\in[0,1]}$ é a PDF da $\mathcal{U}_{[0,1]}$ e $\frac{\mathbb{1}_{\theta\in[0,2\pi]}}{\pi}$ é a PDF da $\mathcal{U}_{[-\frac{\pi}{2},\frac{\pi}{2}]}$, dado que a função $\arctan(\vartheta)$ tem como imagem o conjunto $\left[-\frac{\pi}{2},\frac{\pi}{2}\right]$.

2. Show that X_1 and X_2 are independent standard normal random variables.

Solução:

Do item 1, usamos o resultado de que $Y \sim \mathcal{U}_{[0,1]}$. Isso é facilmente verificável:
Dado que $Y = U_1^2 + U_2^2 \leq 1$,

$$P(Y \leq t) = P(U_1^2 + U_2^2 \leq t) = \frac{\pi\sqrt{t}^2}{\pi\sqrt{1}^2} = t \tag{29}$$

Como $P(Y \leq t)$ é a CDF de Y, deduzimos daí que $Y \sim \mathcal{U}_{[0,1]}$. Seguindo em frente, defina $Z = \sqrt{-2\log Y}$. Assim, $Y = e^{\frac{-Z^2}{2}}$. Pelo **Teorema fundamental da simulação** (enunciado na questão 1), temos que $Y \sim \mathcal{U}_{[0,1]}$ e $Z^2 = F^{-1}(Y) \Rightarrow Z^2 \sim F$, onde $F = \exp(1/2)$, i.e., concluímos que $Z^2 \sim \exp(1/2)$.
Como Z^2 e ϑ são independentes, com $\vartheta \sim \mathcal{U}_{[0,2\pi]}$, temos que

$$f_{Z^2,\vartheta}(z^2,\theta) = f_{Z^2}(z^2)\cdot f_\vartheta(\theta) = \frac{1}{2}\exp\left(-\frac{1}{2}z^2\right)\cdot\frac{1}{2\pi} \tag{30}$$

Aplicando uma transformação de variáveis,

$$f_{(X_1,X_2)}(x_1,x_2) = f_{(Z^2,\vartheta)}(z^2,\theta)\left|\det\left(\frac{\partial\left(Z^2,\vartheta\right)}{\partial(X_1,X_2)}\right)\right| \tag{31}$$

Onde,

$$\left|\det\left(\frac{\partial\left(Z^2,\vartheta\right)}{\partial(X_1,X_2)}\right)\right|^{-1} = \left|\det\begin{pmatrix} \frac{\partial X_1}{\partial Z^2} & \frac{\partial X_1}{\partial \vartheta} \\ \frac{\partial X_2}{\partial Z^2} & \frac{\partial X_2}{\partial \vartheta}\end{pmatrix}\right|^{-1} = \left|\det\begin{pmatrix} \frac{\cos\vartheta}{2Z} & -Z\sin\vartheta \\ \frac{\sin\vartheta}{2Z} & Z\cos\vartheta\end{pmatrix}\right|^{-1} =$$

$$= \left|\frac{\cos\vartheta Z\cos\vartheta}{2Z} + \frac{\sin\vartheta Z\sin\vartheta}{2Z}\right|^{-1} = \left|\frac{\cos^2\vartheta + \sin^2\vartheta}{2}\right|^{-1} = 2 \tag{32}$$

Assim, substituindo o resultado 32 em 31, temos:

$$f_{(X_1,X_2)}(x_1,x_2) = \frac{1}{2\pi}\exp\left(-\frac{1}{2}z^2\right) = \frac{1}{2\pi}\exp\left(-\frac{1}{2}(X_1^2+X_2^2)\right) =$$
$$= \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{X_1^2}{2}\right) + \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{X_2^2}{2}\right) = f_{X_1}(x_1)\cdot f_{X_2}(x_2) \tag{33}$$

Portanto, concluímos que X_1 e X_2 são independentes, e seguem uma distribuição normal padrão.

$\boxed{\textbf{3.}}$ What are the potential benefits of this approach over the Box-Muller algorithm?
Solução:

Diferentemente da abordagem de Box-Muller, o método Polar de Marsaglia tem a vantagem de substituir o cálculo de funções trigonométricas (seno e cosseno) por pontos, i.e., coordenadas U_1 e U_2 normalizadas pelo raio da circunferência ($\sqrt{U_1^2+U_2^2}$).

No caso anterior, quando tomamos um ponto aleatório (U_1,U_2), vimos que esse ponto, pelo método de Marsaglia, será projetado no ponto $(X_1,X_2) = \left(Z\frac{U_1}{\sqrt{Y}}, Z\frac{U_2}{\sqrt{Y}}\right)$, cujo cálculo das coordenadas é computacionalmente menos custoso do que o proposto pelo método de Box-Muller, cujas coordenadas seriam $(\overline{X}_1,\overline{X}_2) = (Z\cos(\vartheta), Z\sin(\vartheta))$.

Em outras palavras, o algoritmo polar de Marsaglia tem um menor custo computacional.

Exercise 4 (Transformation)

$\boxed{\textbf{1.}}$ Let $\pi(x) = \tilde{\pi}(x)/Z_\pi$ be any probability function on $\mathbb{R}$. Prove that if (U, V) is uniformly distributed on $G = \left\{(u, v); 0 \leq u \leq \sqrt{\tilde{\pi}(v/u)}\right\}$, then V/U is distributed according to π, i.e., admits π as a probability density function.

Solução:

Tome a transformação $T(u, v) = (u, v/u) = (x_1, x_2)$. É fácil notar que T é inversível. Assim, podemos aplicar a fórmula da transformação de variáveis (que já foi apresentada no exercício anterior), obtendo:

$$f_{X_1,X_2}(x_1, x^2) = f_{U,V}(x_1, x_2) \left|\det \mathcal{J}_T\right| \tag{34}$$

Onde $\mathcal{J}_T$ é o Jacobiano da inversa da função T. Então,

$$\left|\det \mathcal{J}_T\right|^{-1} = \left|\det \frac{\partial(x_1, x_2)}{\partial(u, v)}\right|^{-1} = \left|\det \begin{pmatrix} 1 & 0 \\ -v/u^2 & 1/u \end{pmatrix}\right|^{-1} = u = x_1 \tag{35}$$

Portanto, fazendo a devida substituição em 34,

$$f_{X_1,X_2}(x_1, x_2) = \mathbb{1}_{\{(x_1,x_2):0\leq x_1 \leq \sqrt{\tilde{\pi}(x_2)}\}} \cdot x_1 \tag{36}$$

Não obstante,

$$\begin{aligned} f_{X_2}(x_2) &= \int_0^{\sqrt{\tilde{\pi}(y)}} f_{X_1,X_2}(x_1, x_2) dx_1 = \int_0^{\sqrt{\tilde{\pi}(y)}} \mathbb{1}_{\{(x_1,x_2):0\leq x_1 \leq \sqrt{\tilde{\pi}(x_2)}\}} \cdot x_1 \; dx_1 = \\ &= \int_0^{\sqrt{\tilde{\pi}(y)}} x_1 \; dx_1 = \frac{\widetilde{\pi}(x_2)}{2} = \pi(x_2) \end{aligned}$$

Com isso, mostramos que

$$f_{V/U}(v/u) = \pi(v/u)$$

□

$\boxed{\textbf{2.}}$ In order to use the result of (1) in practice, we need to be able to sample uniformly from G. Show that if $\sup_x \sqrt{\tilde{\pi}(x)} < \infty$ and $\sup_x |x|\sqrt{\tilde{\pi}(x)} < \infty$, then $G \subseteq R$, where

$$R = \left[0, \sup_x \sqrt{\tilde{\pi}(x)}\right] \times \left[\inf_x x\sqrt{\tilde{\pi}(x)}, \sup_x x\sqrt{\tilde{\pi}(x)}\right].$$

Suggest a way to sample uniformly from G.

Solução:

Considere a dupla $(u, v) \in G$, e $v/u \in \pi$, $u \neq 0$. Assim, é fácil ver que

$$0 \leq u \leq \sqrt{\tilde{\pi}(v/u)} \Rightarrow 0 \leq u \leq \sup_x \sqrt{\tilde{\pi}(v/u)} \tag{37}$$

Logo, $v/u \in \mathbb{R}$ e $u \in \left(0, \sqrt{\tilde{\pi}(v/u)}\right)$.
Revisitando a desigualdade em 37, podemos usar o seguinte truque:

$$
\begin{aligned}
0 \leq u = \frac{v}{v/u} \leq \sqrt{\tilde{\pi}(v/u)} \Rightarrow 0 \leq u = \frac{v}{v/u} \leq \sqrt{\tilde{\pi}(v/u)} \Rightarrow \\
-\frac{v}{u}\sqrt{\tilde{\pi}(v/u)} \leq v \leq \frac{v}{u}\sqrt{\tilde{\pi}(v/u)} \Rightarrow \inf_{v/u} \frac{v}{u}\sqrt{\tilde{\pi}(v/u)} \leq v \leq \sup_{v/u} \frac{v}{u}\sqrt{\tilde{\pi}(v/u)}
\end{aligned} \tag{38}
$$

Portanto, $v \in \left[-\inf_{v/u} \frac{v}{u}\sqrt{\tilde{\pi}(v/u)}, \sup_{v/u} \frac{v}{u}\sqrt{\tilde{\pi}(v/u)}\right]$. Com isso, concluímos que $(u, v) \in G \Rightarrow (u, v) \in R$. Logo, $G \subseteq R$.

Como $G \subseteq R$, podemos amostrar uniformemente de G usando um algoritmo de *Rejection Sampling* de uma Uniforme.

$\boxed{\textbf{3.}}$ Let $\tilde{\pi}(x) = \exp(-x^2/2)$. Using results from (1) and (2), propose a method to sample from $\pi(x)$.

Solução:

Aplicando os resultados do item (2) para o caso em que $\tilde{\pi}(x) = \exp\left(-x^2/2\right)$, temos que:

$$
\sup_{v/u} \sqrt{\tilde{\pi}(v/u)} = 1, \ \inf_{v/u} \frac{v}{u}\sqrt{\tilde{\pi}(v/u)} = -\sqrt{2/e}, \ \sup_{v/u} \frac{v}{u}\sqrt{\tilde{\pi}(v/u)} = \sqrt{2/e}.
$$

Assim,

$$
R = [0, 1] \times [-\sqrt{2}/e, \sqrt{2}/e]
$$

Ainda, note que:

$$
u \leq \sqrt{\tilde{\pi}(v/u)} \Leftrightarrow u^2 \leq \tilde{\pi}(v/u) \Leftrightarrow u^2 \leq \exp(-x^2/2)
$$

Onde $x = v/u$. Portanto,

$$
u^2 \leq \exp(-x^2/2) \Leftrightarrow 2\log u \leq \frac{-x^2}{2} \Leftrightarrow 4\log u \leq -\frac{-v^2}{u^2} \Leftrightarrow v^2 \leq -4u^2 \log u.
$$

Um possível algoritmo para amostrar de $\pi(x)$ é:

1. Gerar $U \sim \mathcal{U}_{[0,1]}$ e $V \sim \mathcal{U}_{[-\sqrt{2}/e, \sqrt{2}/e]}$ independentes até que $V^2 \leq -4U^2 log\ U$
2. Armazenar $\frac{V}{U}$.

Ao fim do processo, teremos uma amostra $X_i = \frac{V_i}{U_i} \sim \exp(-x^2/2)$.

Exercise 5 (Rejection and Importance Sampling)

Consider two probability densities π, q on $\mathbb{X}$ such that $\pi(x) > 0 \Rightarrow q(x) > 0$ and assume that you can easily draw samples from q. Whenever $\pi(x)/q(x) \leq M < \infty$ for any $x \in \mathbb{X}$, it is possible to use rejection sampling to sample from π. When M is unknown or when this condition is not satisfied, we can use importance sampling techniques to approximate expectations with respect to π. However it might be the case that most samples from q have very small importance weights.

Rejection control is a method combining rejection and importance weighting. It relies on an arbitrary threshold value $c > 0$. We introduce the notation $w(x) = \pi(x)/q(x)$ and

$$Z_c = \int_{\mathbb{X}} \min\{1, w(x)/c\}\, q(x)dx$$

Rejection control proceeds as follows.

- **Step a.** Generate independent $X \sim q$, $U \sim \mathcal{U}_{[0,1]}$ until $U \leq \min\{1, w(x)/c\}$
- **Step b.** Return X.

$\boxed{\textbf{1.}}$ Give the expression of the probability density $q*(x)$ of the accepted samples.

Solução:

Como já foi visto nos exercícios anteriores,

$$P(X \in A \mid X \text{ aceito}) = \frac{P(X \in A, X \text{ aceito})}{P(X \text{ aceito})} \tag{39}$$

Assim,

$$P(X \in A, X \text{ aceito}) = \int_A \int_0^1 \mathbb{1}_{\{u \in (0, \min\{1, w(x)/c\})\}} q(x)\ dudx = \int_A \min\{1, w(x)/c\} q(x)dx \tag{40}$$

e

$$P(X \text{ aceito}) = \int_X \int_0^1 \mathbb{1}_{\{u \in (0, \min\{1, w(x)/c\})\}}\ dx = \int_{\mathbb{X}} \min\{1, w(x)/c\} q(x)\ dx \tag{41}$$

Substituindo 40 e 41 em 39:

$$P(X \in A \mid X \text{ aceito}) = \frac{\int_A \min\{1, w(x)/c\} q(x)dx}{\int_{\mathbb{X}} \min\{1, w(x)/c\} q(x)dx} = \frac{\int_A \min\{1, w(x)/c\} q(x)dx}{Z_c} \tag{42}$$

Portanto,

$$q^*(x) = \frac{\min\{1, w(x)/c\}\ q(x)}{Z_c} \tag{43}$$

$\boxed{\textbf{2.}}$ Prove that

$$\mathbb{E}_{q^*}\left([w^*(X)]^2\right) = Z_c \mathbb{E}_q\left(\max\{w(X), c\} w(X)\right)$$

where $w^*(x) = \pi(x)/q^*(x)$.

Solução:

Do exercício anterior,

$$q^*(x) = \frac{\min\{1, w(x)/c\}\ q(x)}{Z_c}.$$

Com isso, inferimos que

$$\frac{\pi(x)}{q^*(x)} = \frac{\pi(x)\ Z_c}{\min\{1, w(x)/c\}\ q(x)} = \frac{w(x)\ Z_c}{\min\{1, w(x)/c\}}, \tag{44}$$

Uma vez que $\frac{1}{min\{1,w(x)/c\}} = max\{1, c/w(x)\}$, podemos reescrever 44 como:

$$\frac{\pi(x)}{q^*(x)} = \frac{w(x)\ Z_c}{\min\{1, w(x)/c\}} = w(x)\ Z_c\ max\{1, c/w(x)\} = Z_c\ max\{w(x), c\} \tag{45}$$

Com isso, podemos facilmente calcular a esperança:

$$E_{q^*(x)}\left([w^*(X)]^2\right) = \int_{\mathbb{X}} \frac{\pi^2(x)}{q^{*2}(x)} q(x)\ dx = \int_{\mathbb{X}} \frac{\pi(x)}{q^*(x)} \pi(x)\ dx = Z_c \int_{\mathbb{X}} \max\{w(x), c\}\ \pi(x) dx =$$

$$= Z_c \int_{\mathbb{X}} \max\{w(x), c\}\ w(x)\ q(x)\ dx = Z_c\ \mathbb{E}_{q(x)}\big(\max\{w(X), c\}\ w(X)\big) \tag{46}$$

$\square$

3. Establish that

$$\mathbb{E}_q(\min\{w(X), c\})\mathbb{E}_q\left(\max\{w(X), c\}w(X)\right) \leq \mathbb{E}_q\left(\min\{w(X), c\} \max\{w(X), c\}w(X)\right)$$

(*Hint. Show first that for any* $c > 0,\ w_1 > 0,\ w_2 > 0$

$$h\left(w_1, w_2\right) = \left[\min\left\{w_1, c\right\} - \min\left\{w_2, c\right\}\right]\left[w_1 \max\left\{w_1, c\right\} - w_2 \max\left\{w_2, c\right\}\right] \geq 0$$

This is related to the Harris inequality).

Solução:

(Não consegui fazer)

4. Deduce from the results established in (2) and (3) that

$$\mathbb{V}_{q^*}\left(w^*(X)\right) \leq \mathbb{V}_q(w(X))$$

Solução:

(Não consegui fazer)

Exercise 6 (Rejection and Importance Sampling)

We want to use Monte Carlo methods to approximate the integral

$$I = \int_{\mathbb{X}} \phi(x)\pi(x)dx$$

where $\phi : \mathbb{X} \to \mathbb{R}$ and π is a probability density on $\mathbb{X}$. Assume we have access to a proposal probability density q such that $w(x) = \pi(x)/q(x) \leq M < \infty$ for any $x \in \mathbb{X}$.

$\boxed{\textbf{1.}}$ Consider the extended probability density

$$\overline{\pi}_{X,U}(x,u) = \begin{cases} Mq(x) & \text{for } x \in \mathbb{X}, u \in \left[0, \frac{w(x)}{M}\right] \\ 0 & \text{otherwise} \end{cases}$$

Verify that $\overline{\pi}_X(x) = \pi(x)$.

Solução:

Ora,

$$\overline{\pi}_X(x) = \int_0^1 \overline{\pi}_{X,U}(x,u)\ du = \int_0^{w(x)/M} Mq(x)\ du = M[q(x)u]_0^{w(x)/M} = w(x)q(x) = \pi(x) \tag{47}$$

$\boxed{\textbf{2.}}$ Using the identity

$$I = \int_0^1 \int_{\mathbb{X}} \phi(x)\overline{\pi}_{X,U}(x,u)dxdu,$$

give the expression of the normalised importance sampling estimate $\widehat{I}_n$ of I when using n independent samples (X_i, U_i) such that $(X_i, U_i) \sim \overline{q}_{X,U}$, where $\overline{q}_{X,U}(x,u) = q(x) \times \mathbb{I}_{[0,1]}(u)$ (that is under $\overline{q}_{X,U}$ we have $X \sim q$, $U \sim \mathcal{U}_{[0,1]}$ and X and U are independent). Express this estimate as a function the importance weight function

$$\overline{w}(x,u) = \frac{\overline{\pi}_{X,U}(x,u)}{\overline{q}_{X,U}(x,u)}$$

Show that this estimate is equivalent to the estimate one would obtain by sampling from π using rejection sampling using n proposals from q.

Solução:

Primeiramente, vamos calcular $\overline{w}_{X,U}(x,u)$:

$$\overline{w}_{X,U}(x,u) = \frac{\overline{\pi}_{X,U}(x,u)}{\overline{q}_{X,U}(x,u)} = \begin{cases} \frac{Mq(x)}{q(x)} = M, & \text{se } x \in \left[0, \frac{w(x)}{M}\right] \\ 0, & \text{caso contrário} \end{cases} \tag{48}$$

Agora, podemos fazer uma aproximação de Monte Carlo $\widehat{I}$ para I, considerando n amostras independentes (X_i, U_i), por meio da seguinte relação:

$$\widehat{I}_n = \frac{\sum_{i=1}^n \phi(X_i)\,\bar{w}(X_i, U_i)}{\sum_{i=1}^n \bar{w}(X_i, U_i)} = \frac{M\sum_{i=1}^d \phi(X_i)}{d \cdot M} = \frac{\sum_{i=1}^d \phi(X_i)}{d} \tag{49}$$

Onde d representa o número de pesos $\overline{w}$ diferentes de 0.

Equivalentemente, poderíamos obter uma estimativa para I por meio de um algoritmo de *Rejection Sampling*, tomando a $\overline{q}_{X,U}(x,u)$ como distribuição proposta, e $\overline{\pi}_{X,U}(x,u)$ como distribuição *target*.

$\boxed{\textbf{3.}}$ Show that

$$\mathbb{V}_q(w(X)) \leq \mathbb{V}_{\bar{q}_{X,U}}(\bar{w}(X,U)).$$

Solução:

(Nao consegui fazer)

$\boxed{\textbf{4.}}$ Show similarly that one can reinterpret the rejection control control procedure introduced in Exercise 5 as a standard importance sampling procedure on the extended space $\mathbb{X} \times [0,1]$. Give the expressions of the extended "target"probability density $\tilde{\pi}_{X,U}$ on $\mathbb{X} \times [0,1]$, the associated importance density $\tilde{q}_{X,U}(x,u)$ and show that

$$\mathbb{V}_q(w(X)) \leq \mathbb{V}_{\tilde{q}_{X,U}}(\widetilde{w}(X,U))$$

where $\tilde{w}(x,u) = \tilde{\pi}_{X,U}(x,u)/\tilde{q}_{X,U}(x,u)$.

Solução:

(Não consegui fazer)

Simulation question (Rejection)

1. Reproduce the figures on the estimation of the number π in the slides of Lecture 2.

Solução:

Considere um quadrado $Q \subset \mathbb{R}^2$, cujos lados têm tamanho 1. Por simplificação, suponha $Q = [0,1] \times [0,1]$. Agora, tome um círculo C, inscrito no quadrado, com raio $\frac{1}{2}$. Suponha que estamos interessados em calcular a área do círculo A_C.

$$A_C = \pi \left(\frac{1}{2}\right)^2 = \frac{\pi}{4} = \int\int_C dxdy = \int\int_Q \mathbb{1}_{\{(x,y)\in C\}}\, dxdy \Rightarrow \pi = 4\int_Q \mathbb{1}_{\{(x,y)\in C\}}\, dxdy \tag{50}$$

Sabemos que a área do quadrado $A_Q = 1$. Então,

$$\frac{A_C}{A_Q} \iff \frac{\pi}{4} \Rightarrow A_C = 4A_Q \tag{51}$$

Para aproximar o valor dessa integral, vamos utilizar Monte Carlo. Se gerarmos um número grande o suficiente de pontos, temos:

$$\frac{A_C}{A_Q} = \frac{\eta_C}{\eta} = \frac{\pi}{4} \Rightarrow \pi = 4\frac{\eta_C}{\eta} \tag{52}$$

Onde η_C é o número de pontos dentro do círculo, e η é o total de pontos dentro do quadrado. Por meio da relação em 52, vamos estimar π utilizando o seguinte algoritmo:

1. Geramos $X, Y \sim \mathcal{U}_{[0,1]}$
2. Se $X^2 + Y^2 \leq 1$, adicionamos 1 a η_C e η. Caso contrário, adicionamos 1 apenas a η.
3. Executamos novamente a partir do passo (1).
4. Após atingirmos o tamanho da amostra desejado, estimamos π pela relação 52.

Abaixo, é apresentado o código para execução em Python e, a seguir, o resultado da estimativa de Monte Carlo para π.

```python
# imports
import numpy as np

n = 100       # tamanho da amostra
l = 1         # lado do quadrado
r = 1/2       # raio do circulo
Nc = 0        # Numero de pontos dentro do circulo
inside = []   # armazena um booleano (True se o ponto esta dentro do circulo)

# gerando n numeros aleatorios X e Y
x = np.random.ranf(n)
y = np.random.ranf(n)

# Conta o numero de pontos dentro do circulo.
for x_i,y_i in zip(x,y):

    # checa se o ponto (X,Y) esta dentro do circulo
    if x_i**2 + y_i**2 <= 1.0:
        Nc += 1
        inside.append(True)
```

```
    else:
        inside.append(False)

pi = (float(Nc) / n) * 4       # Nc / Np = pi / 4
print('Estimativa de Monte Carlo para pi:', pi)
```

Após realizar esse experimento 1000 vezes (com um tamanho de amostra fixo em 10 mil), a estimativa média para π foi de 3.1406.

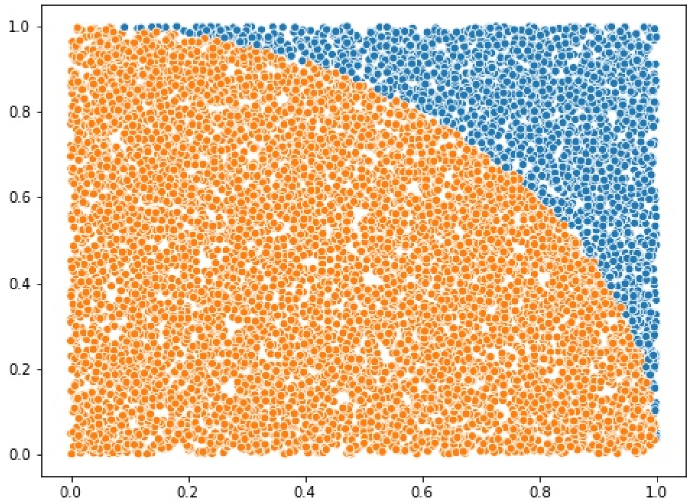

Figura 1: $X, Y \sim \mathcal{U}_{[0,1]}$. De laranja se $X^2 + Y^2 \leq 1$.

2. Implement the Box-Muller algorithm from Lecture 2.

Solução:

Vamos implementar o algoritmo de Box-Muller da Aula 2 com o seguinte código:

```
# imports
import numpy as np

def BoxMuller(n):
    # geramos variaveis em pares
    n_pares = int(np.ceil(n/2))

    R = np.sqrt(-2 * np.log(np.random.ranf(n_pares)))      # simulando R
    theta = np.pi * 2 * np.random.ranf(n_pares)             # simulando theta

    X_1 = R * np.cos(theta)   # amostra x_1
    X_2 = R * np.sin(theta)   # amostra x_2

    return X_1, X_2

n = 10000
x_1, x_2 = BoxMuller(n*2)
```

Agora, vamos plotar X_1 e X_2 e comparar suas densidades com as de duas normais padrão "genuínas".

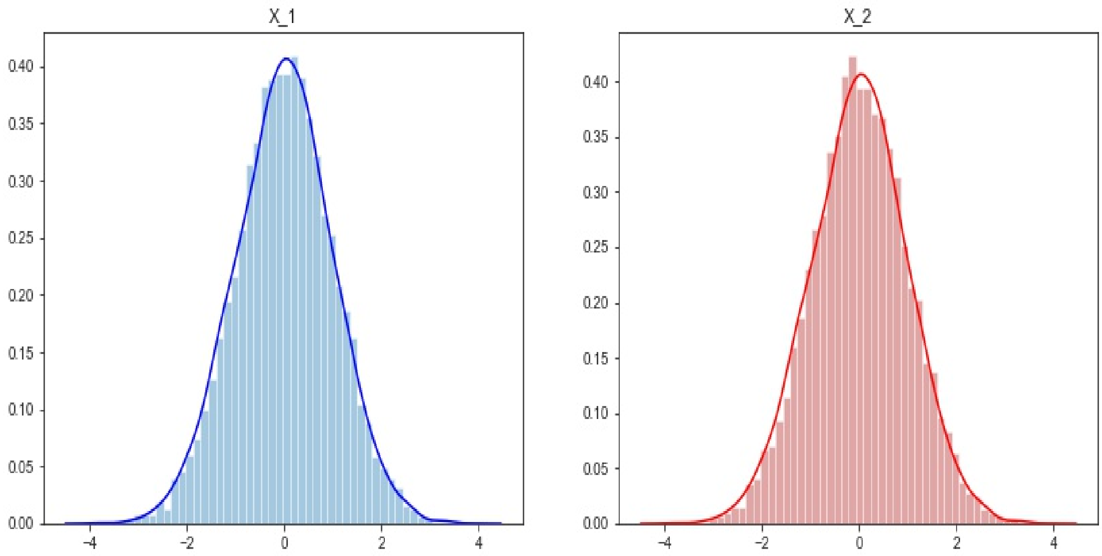

Figura 2: X_1 e X_2 geradas a partir do algoritmo de Box-Muller

Aparentemente, X_1 e X_2 têm densidades bem próximas à de uma normal padrão. Com o próximo gráfico (3), também vemos que X_1 e X_2 mostram, a princípio, ser independentes (conforme já esperávamos como resultado do algoritmo de Box-Muller).

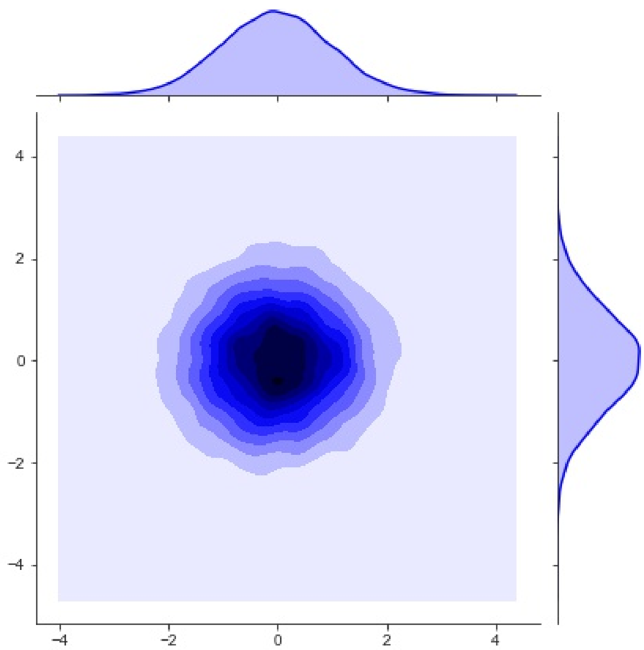

Figura 3: Distribuição conjunta de X_1 e X_2.

3. Consider the genetic linkage model as in the slides of Lecture 3. Sample some simulated data with a fixed value of θ of your choice. Implement rejection sampling and reproduce the histograms of the posterior of θ and the waiting time before acceptance. Experiment with different proposal distributions.

Solução:

Recorrendo ao modelo *Genetic Linkage* apresentado na Nota de Aula 2, temos a seguinte esrutura:

- Observamos $(Y_1, Y_2, Y_3, Y_4) \sim \mathcal{M}\left(n; \frac{1}{2} + \frac{\theta}{4}; \frac{1}{4}(1-\theta); \frac{\theta}{4}\right)$, onde $\mathcal{M}$ é uma distribuição multinomial e $\theta \in (0, 1)$.

- A verossimilhança é dada por

$$p\left(y_1, \ldots, y_4 \mid \theta\right) \propto (2+\theta)^{y_1}(1-\theta)^{y_2+y_3}\theta^{y_4}$$

- Seguindo a abordagem Bayesiana com *priori* $p(\theta) = \mathbb{I}_{\{\theta \in [0,1]\}}$, temos como *posteriori*:

$$p\left(\theta | y_1, \ldots, y_4\right) \propto (2+\theta)^{y_1}(1-\theta)^{y_2+y_3}\theta^{y_4}$$

Com isso em mente, vamos aplicar *Rejection Sampling* tomando $q(\theta) = \tilde{q}(\theta) = p(\theta)$ como distribuição proposta para amostrar de $p\left(\theta \mid y_1, \ldots, y_4\right)$. Tomaremos como distribuição alvo

$$\widetilde{\pi}(\theta) = (2+\theta)^{y_1}(1-\theta)^{y_2+y_3}\theta^{y_4}$$

e $\widetilde{M} = \sup_{\theta \in [0,1]} \tilde{\pi}(\theta)$, tal que $\sup_x \tilde{\pi}(x)/\tilde{q}(x) \leq \widetilde{M}$.

Abaixo, segue o código da implementação do *Rejection Sampling*.

```
theta_verdadeiro = 0.7

# definindo funcao que retorna as probabilidades da multinomial
def pMultinomial(theta):
    return [1/2 + theta/4, (1/4) * (1 - theta), (1/4) * (1 - theta), theta/4]

def logVerossimilhanca(theta):
    # np.random.seed(17)
    # amostrando de uma multinomial
    sample = np.random.multinomial(100, pMultinomial(theta_verdadeiro), size=1)
    sample = sample[0]

    return sample[0]*np.log(2+theta) + (sample[1]+sample[2])*np.log(1 - theta) +
    sample[3]*np.log(theta)

# avalia a log verossimilhan a em um grid
grid = np.linspace(0,1,10000)
ver = list(map(logVerossimilhanca, grid[1:]))

max_ver = max(ver)  # m x. verossimilhan a

# encontra o estimador de m xima verossimilhan a
max_theta = 0
for i in range(len(ver)):
    if ver[i] == max_ver:
        max_theta = grid[i]

print('Maximum Likelihood Estimation:', max_theta)
```

```
# Aplicando rejection sampling

N = 10000                           # numero de pontos que queremos amostrar
n_aceito = 0                        # numero de pontos que foram aceitos
amostras = []                       # pontos amostrados
total_amostras = 0                  # total de pontos que foram amostrados
n_amostras_ac = np.repeat(0,N)      # numero de pontos amostrados ate a aceitacao

while n_aceito < N:
    total_amostras += 1
    n_amostras_ac[n_aceito] = n_amostras_ac[n_aceito] + 1
    x = np.random.ranf(1)[0]
    log_alfa = logVerossimilhanca(x) - max_ver

    if np.log(np.random.ranf(1)[0]) < log_alfa:
        n_aceito += 1
        amostras.append(x)

print('Numero de amostras aceitas:', n_aceito)
print('Total de amostras retiradas:', total_amostras)
```

Abaixo, temos os gráficos da distribuição da amostra que compomos e a distribuição do número de tentativas até a aceitação de cada ponto sorteado, respectivamente.

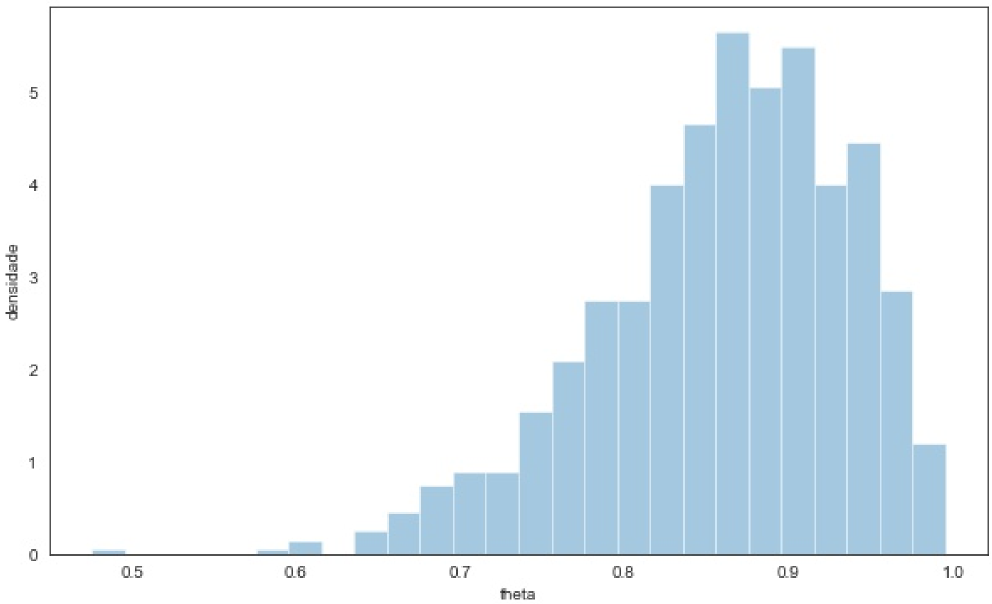

Figura 4: Distribuição *target* | Tempo de aceitação

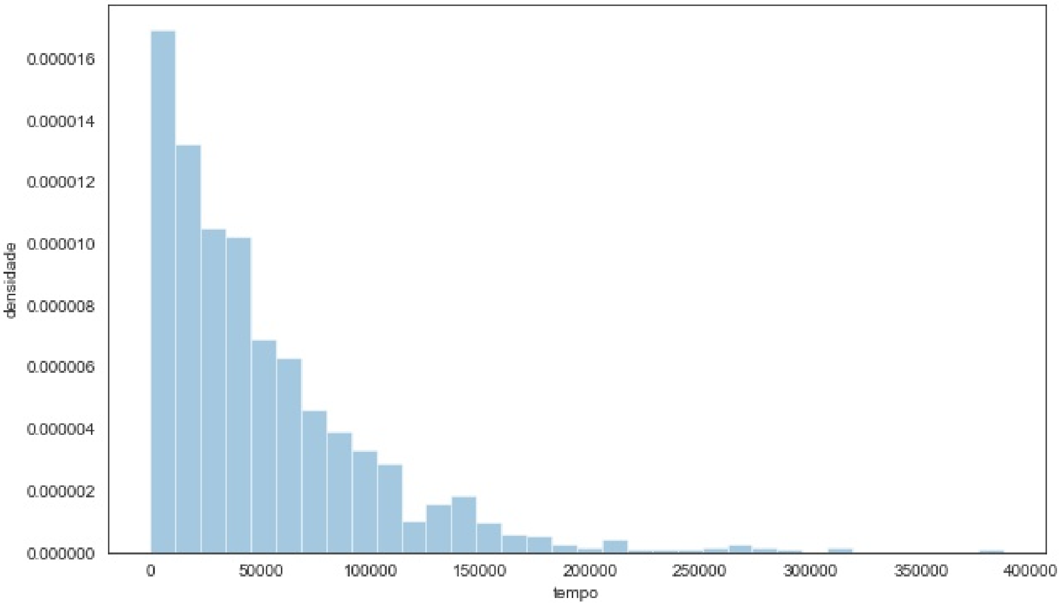

Figura 5: Tempo de aceitação

4. Implement a sampler to draw from a mixture of Gaussians

$$\pi(x) = \omega_1 \phi\left(x; \mu_1, \sigma_1^2\right) + \omega_2 \phi\left(x; \mu_2, \sigma_2^2\right),$$

where ϕ is the Gaussian pdf. You are allowed to use R's Gaussian generator (but feel free to reimplement Box-Muller from Lecture 3 or Marsaglia's method from Question 1of this sheet, just for fun).

Solução:

Tome $w_1 = 1 - w_2 \in [0, 1]$. Assim, vamos utilizar o seguinte código para amostrar de uma mistura de gaussianas:

```
# funcao para amostrar de uma mistura de 2 gaussianas
def mistura_2gaussianas(w1, mu1, sigma1, mu2, sigma2):
    w2 = 1-w1
    r = np.random.randint(0,2)
    random_number = w1*r + w2*(1-r)

    if random_number == w1:
        a = np.random.normal(mu1, sigma1)
    elif random_number == w2:
        a = np.random.normal(mu2, sigma2)

    return a

N = 10000          # tamanho da amostra
amostras = []      # pontos sorteados

# fazendo a amostragem
for i in range(N+1):
    amostras.append(mistura_2gaussianas(0.41, 4.5, 1, 1.2, 1))
```

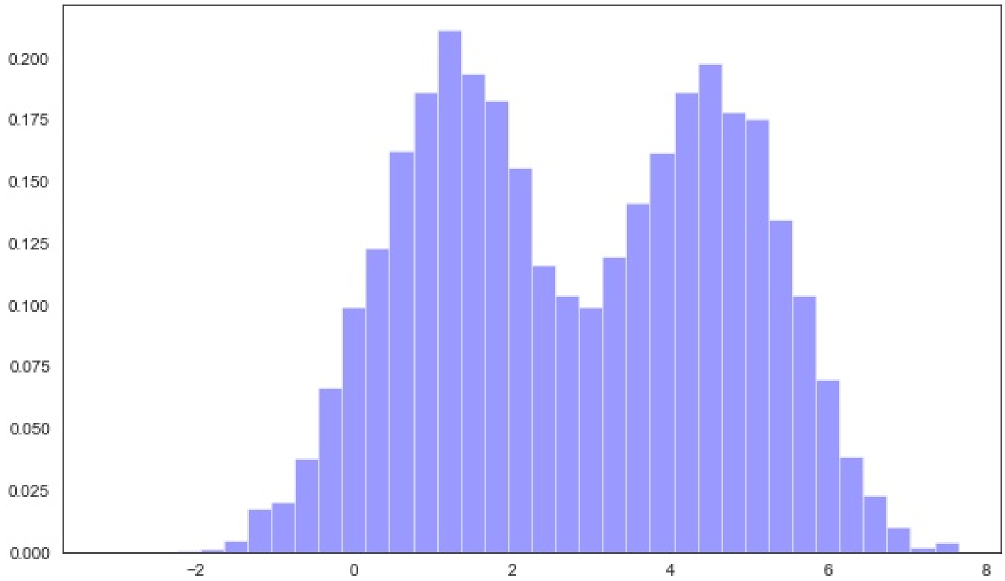

Figura 6: Histograma de $X \sim \pi(x)$, onde $w_1 = 0.41,\ w_2 = 0.59,\ \mu_1 = 4.5, \mu_2 = 1.2,$ e $\sigma_1\sigma_2 = 1$

5. Let

$$h(x) = [\cos(50x) + \sin(20x)].$$

We consider estimating $\int_0^1 h(x)dx$ through Monte Carlo methods.

- First of all, what is the exact answer, to accuracy within 10^{-4}?
- Can you implement rejection sampling with a uniform proposal?
- Find a way to assess how good you are doing.
- Implement an importance sampling solution with a smart proposal (*hint: plot h and find a matching q*).

Solução:

Primeiramente, vamos visualizar a função $h(x)$:

- Integrando $h(x)$, encontramos:

$$\int_0^1 h(x)dx = \left[\frac{x}{2} + \frac{\sin(100x)}{200} + 2\left(\frac{\cos(30x)}{60} - \frac{\cos(70x)}{140}\right) + \frac{x}{2} - \frac{\sin(40x)}{80}\right]_0^1 =$$

$$= H(1) - H(0) = 0.9652009360501459$$

A seguir, é apresentado o código para implementação em Python.

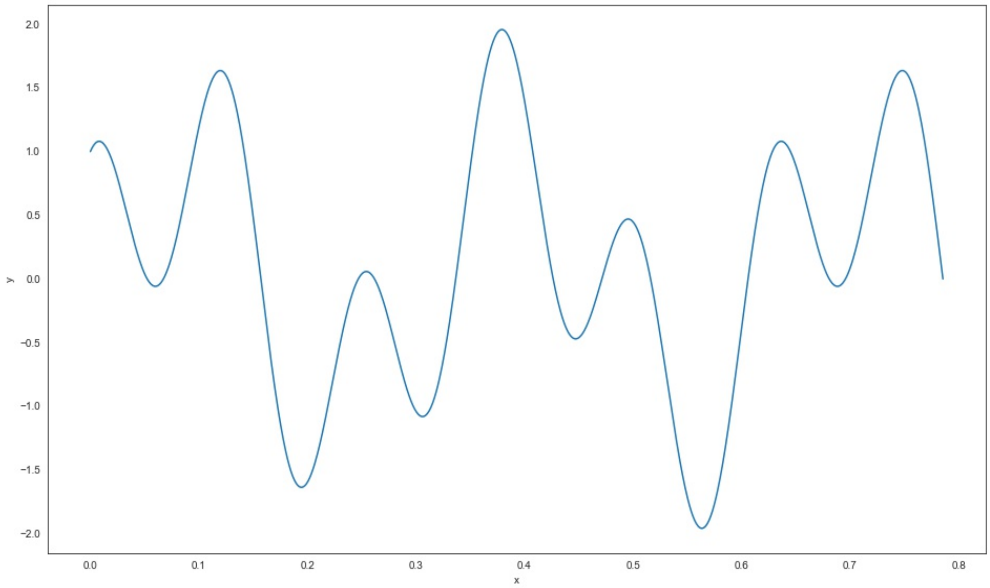

Figura 7: $h(x) = \cos(50x) + \sin(20x)$

```
# define h(x)
def h(x):
    return np.cos(50*x) + np.sin(20*x)

grid = np.linspace(0,np.pi/4,1000)   # grid para o plot
y = list(map(h, grid))               # avalia a funcao no grid

sns.lineplot(y=y, x=grid)            # faz o plot

# define a primitiva H(x)
def H(x):
    return x/2 + np.sin(100*x)/200 + 2* (np.cos(30*x)/60 -np.cos(70*x)/140) + x
   /2 -np.sin(40*x)/80

# calcula o verdadeiro valor da integral
integral = H(1) - H(0)
print('Valor verdadeiro da integral:', integral)
```

- Agora, vamos aplicar *Rejection Sampling* tomando uma distribuição uniforme em $[0, 1]$ como distribuição proposta. (Código abaixo)

```
N = 1000           # numero de pontos
Nrep = 1000        # numero de repeticoes
erros = []         # lista de erros

for i in range(Nrep):
    u = np.random.ranf()
    w = h(u)
    erros.append(np.abs(np.mean(w) - integral)/integral)
```

- (Não consegui implementar a tempo)

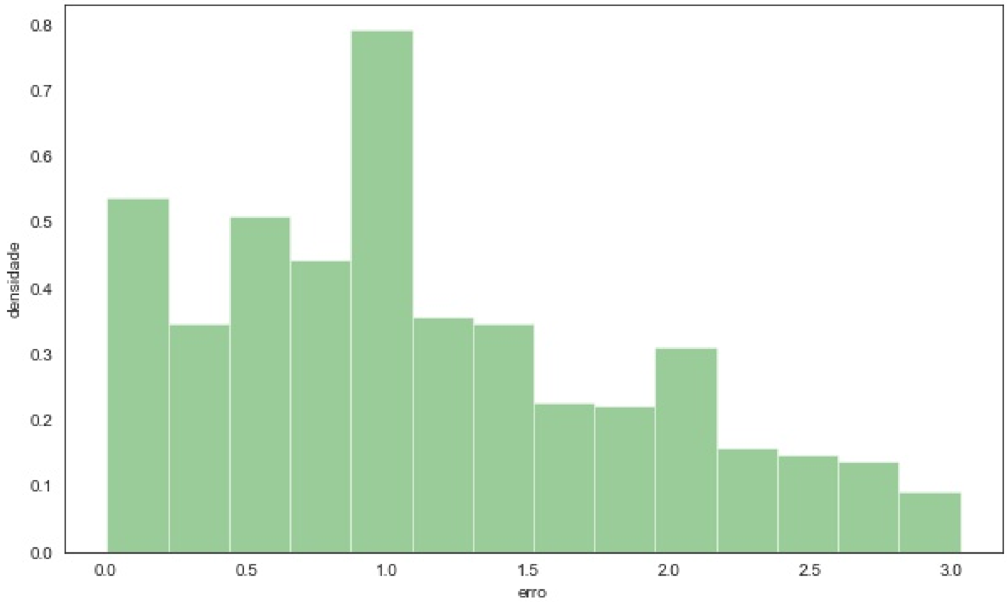

Figura 8: Erros de aproximação do valor verdadeiro da integral.

- (Não consegui implementar a tempo)