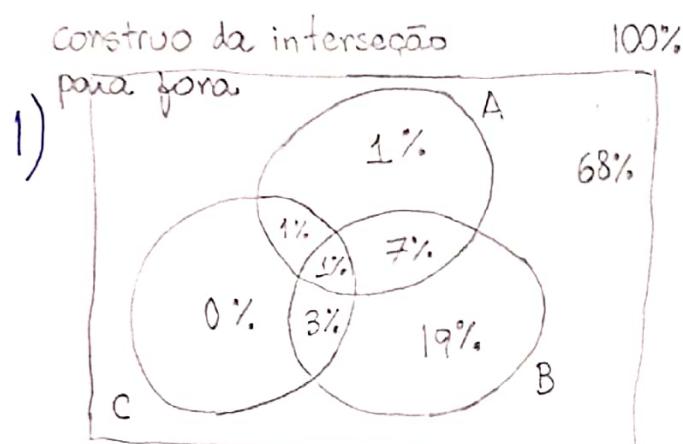


# Lista 0 - Mineração de Dados - 2020.QS

Carlo Domenico Longo de Lemos 21002915

1/8



- 1) As pessoas que lêem apenas um jornal são as pessoas que estão em A, B, C mas não estão nem  $A \cap B$ ,  $B \cap C$ ,  $A \cap C$ , ou  $A \cap B \cap C$ . Portanto

$$\text{apenas } A : |A \setminus (A \cap B \cap C)| \xrightarrow{\text{apenas em } A} 1\% + 19\% + 0\% = 20\% \quad \begin{matrix} \nearrow \text{apenas em } C \\ \searrow \text{apenas em } B \end{matrix}$$

$$20\% \text{ de } 100\,000 \text{ moradores} = \boxed{20\text{K}} \text{ moradores}$$

- 2) Analogamente vamos somar as parcelas individuais adequadadas:

$$|A \cap B \cap C|$$

$$\underbrace{1\%}_{|A \cap C \setminus (A \cap B \cap C)|} + \underbrace{7\%}_{|A \cap B \setminus (A \cap B \cap C)|} + \underbrace{3\%}_{|B \cap C \setminus (A \cap B \cap C)|} + \underbrace{1\%}_{|A \cap B \cap C|} = 12\%$$

$\Rightarrow \boxed{12\text{K}} \text{ moradores} \text{ lêem pelos dois jornais.}$

- 3) Se o morador lê B e algum regional (A e C) então ele está no conjunto  $B \cap (A \cup C) = (B \cap A) \cup (B \cap C)$ . Utilizando explicitamente o princípio da inclusão-exclusão basta calcular  $|B \cap (A \cup C)| = |B \cap A| + |B \cap C| - |\overbrace{(B \cap A) \cap (B \cap C)}^{A \cap B \cap C}| =$
- $$= 8\% + 4\% - 1\% = 11\%$$

Dando o mesmo resultado que nossa construção "individualizada" colocando só quem está em cada segmento

$$\begin{array}{c} \text{"apenas BnA"} \quad \text{"apenas BnC"} \quad \text{"apenas AnBnC"} \\ \swarrow \quad \searrow \quad \swarrow \\ 7\% \quad + \quad 3\% \quad + \quad 1\% = 11\% \end{array}$$

Ou seja, 11K moradores leem pelo menos um dos jornais locais e o jornal mesoregional.

4) Pela nossa construção:

$$100\% - (1\% + 7\% + 1\% + 3\% + 1\% + 19\% + 0\%) = 68\%$$

68K moradores não leem nenhum jornal.

5) Pela nossa construção:

$$7\% + 3\% = 10\%$$

"apenas BnA" "apenas BnC"

10K moradores leem apenas um jornal local e o jornal da mesoregião.

2) Seja  $\Omega$  = conjunto clientes de dado ano, defina uma função mensurável risco  $R: \Omega \rightarrow \{A, M, B\}$  (alto, médio, baixo).

Os eventos  $\{R=B\}$ ,  $\{R=M\}$  e  $\{R=A\}$  são eventos de  $\Omega$ .

Analogamente definir  $A: \Omega \rightarrow \{S, N\}$  mensurável e temos os eventos  $\{A=S\}$  e  $\{A=N\}$ . (sim, não)

Temos  $R$  e  $A$ , "variáveis aleatórias" que nos dão se aquele cliente de um determinado ano tem tal risco e se sofreu

acidente ou não.

Com o espaço e eventos definidos, o enunciado nos diz que

- $P(A=S | R=B) = 5\%$  (e, consequentemente,  $P(A=N | R=B) = 95\%$ )
- $P(A=S | R=M) = 15\%$  (e  $P(A=N | R=M) = 85\%$ )
- $P(A=S | R=A) = 30\%$  (e  $P(A=N | R=A) = 70\%$ )
- $P(R=B) = 20\%$ ,  $P(R=M) = 50\%$ ,  $P(R=A) = 30\%$

Primeiro, queremos calcular a proporção de pessoas que sofreram acidentes, ou seja  $P(A=S)$ . Observando que

$$\Omega = \{R=B\} \cup \{R=M\} \cup \{R=A\} \text{ então}$$

é união disjunta

$$\begin{aligned} P(A=S) &= P(\{A=S\} \cap \Omega) = P\left[\{A=S\} \cap (\{R=B\} \cup \{R=M\} \cup \{R=A\})\right] \\ &= P\left[\{A=S, R=B\} \cup \{A=S, R=M\} \cup \{A=S, R=A\}\right] = \begin{array}{l} \text{prob da união} \\ \text{de eventos disj.} \\ \text{é a soma das prob.} \end{array} \\ &= P[A=S, R=B] + P[A=S, R=M] + P[A=S, R=A] \end{aligned}$$

Não temos diretamente  $P(A=S, R=i)$ ,  $i \in \{B, M, A\}$ , mas pela definição de prob. condicional temos

$$P(A=S | R=i) = \frac{P(A=S, R=i)}{P(R=i)} \Rightarrow P(A=S, R=i) = P(A=S | R=i) P(R=i)$$

Quantidades que sabemos!

Portanto

$$P(A=S) = P[A=S|R=B] \cdot P[R=B] + P[A=S|R=M] \cdot$$

$$P[R=M] + P[A=S|R=A] \cdot P[R=A] =$$

$$= (0.05) \cdot 0.2 + (0.15) 0.5 + (0.3) 0.3 =$$

$$= \boxed{17,5\%}$$

A proporção de pessoas que sofrem acidente em dado ano foi de 17,5%.

Agora queremos calcular a probabilidade de alguém ser de risco baixo dado que não sofreu acidente. Matematicamente,  $P(R=B|A=N)$ .

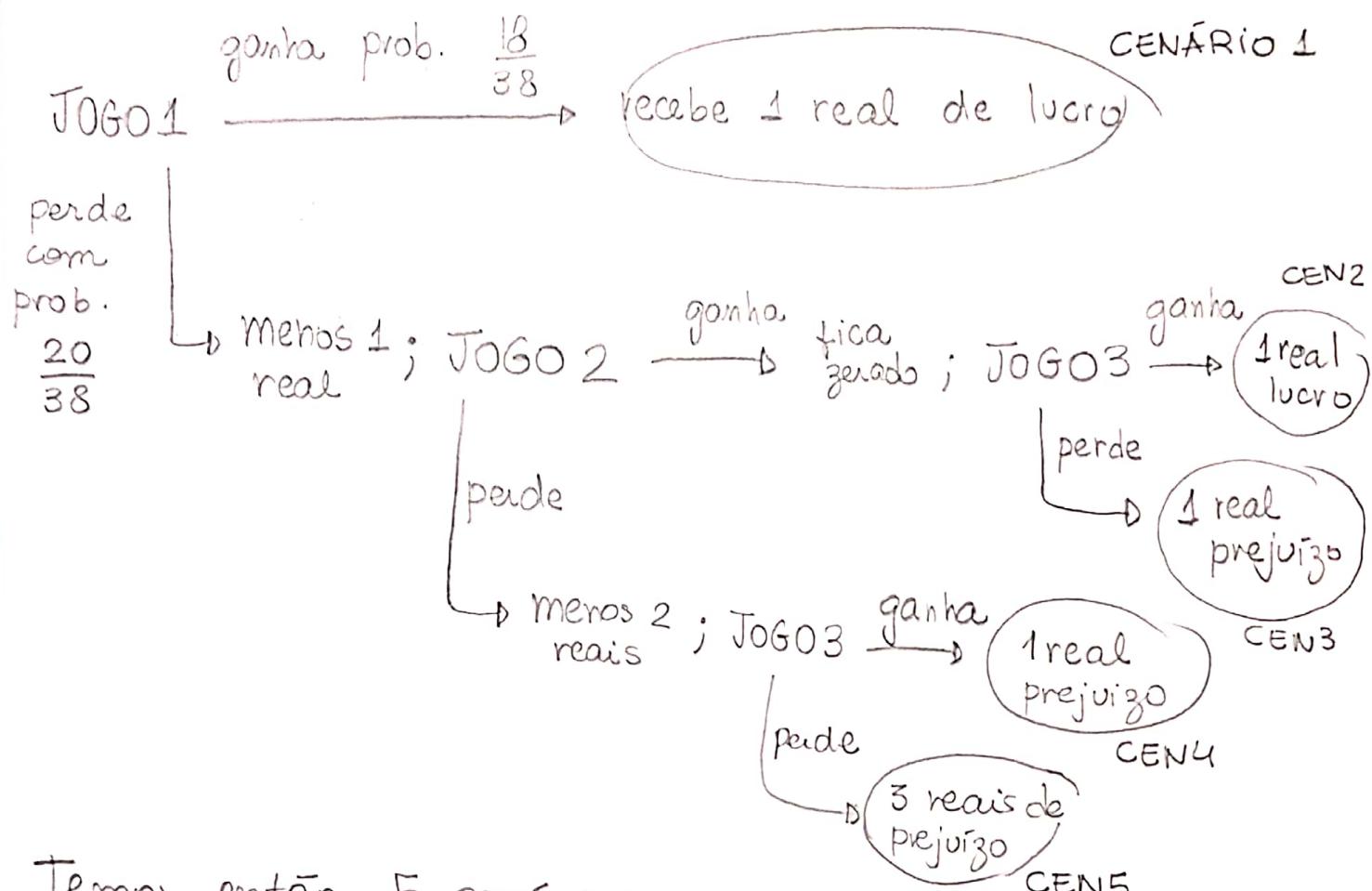
Aqui vamos usar o teorema de Bayes que nos garante que  $P(R=B|A=N) = \frac{P(A=N|R=B) \cdot P(R=B)}{P(A=N)}$ .

Todos estes valores foram dados no enunciado, a menos de  $P(A=N)$  que calculamos indiretamente observando que  $P(A=N) = 1 - P(A=S)$ , logo

$$P(R=B|A=N) = \frac{(0.95)(0.2)}{0.825} \approx \boxed{23\%}$$

Logo, a probabilidade do segurado ser de risco baixo é  $\approx 23\%$ .

3) Vamos analisar os rumos que o jogo pode seguir:



Temos então 5 cenários possíveis.

Esse pode ser considerado meu espaço amostral.

Cenários diferentes podem resultar em mesmos ganhos e prejuízos (definido posteriormente pela VA  $X$ ), mas calcular as probabilidades individuais dos diferentes cenários nos ajudará depois com  $X$ .

$$\Omega = \{ \text{cen}1, \text{cen}2, \text{cen}3, \text{cen}4, \text{cen}5 \} \quad (\mathcal{A} = \xrightarrow{\text{sigma-álgebra}} P(\Omega)).$$

$$\cdot P[\{\text{cen}1\}] = P(\text{ganhando na primeira}) = \frac{18}{38}$$

jogos independentes

$$\cdot P[\{\text{cen}2\}] = P(\text{perde } 1^{\text{a}}, \text{ ganha } 2^{\text{a}} \text{ e } 3^{\text{a}}) = \frac{20}{38} \cdot \frac{18}{38} \cdot \frac{18}{38} = \frac{810}{6859}$$

•  $P[\{cen\ 3\}] = P(\text{perde a } 1^{\text{a}}, \text{ ganha a } 2^{\text{a}} \text{ e perde a } 3^{\text{a}}) =$

$$= \frac{20}{38} \cdot \frac{18}{38} \cdot \frac{20}{38} = \frac{900}{6859}$$

•  $P[\{cen\ 4\}] = P(\text{perde a } 1^{\text{a}}, \text{ a } 2^{\text{a}} \text{ e ganha a } 3^{\text{a}}) =$

$$= \frac{20}{38} \cdot \frac{20}{38} \cdot \frac{18}{38} = \frac{900}{6859}$$

•  $P[\{cen\ 5\}] = P(\text{perde a } 1^{\text{a}}, 2^{\text{a}} \text{ e } 3^{\text{a}}) = \frac{20}{38} \cdot \frac{20}{38} \cdot \frac{20}{38} = \frac{1000}{6859}$

Agora que temos  $\Omega$  e  $P$  bem definidos (dado  $A \subset \Omega$ ,  $P(A) = \sum_{w \in A} P[\{w\}]$ ), podemos definir  $X$  e calcular sua distribuição. Cada um dos cenários da um determinado lucro ou prejuízo, esse será o valor de  $X$ .

•  $X(cen\ 1) = +1$ , •  $X(cen\ 2) = +1$ , •  $X(cen\ 3) = -1$

•  $X(cen\ 4) = -1$ , •  $X(cen\ 5) = -3$ .

Percebemos que  $X$  assume 3 valores possíveis:  $1, -1, -3$ .

Para calcular a probabilidade deles vamos olhar a pré-imagem de  $X$  e calcular em  $\Omega, P$ . Isso é o que chamamos de probabilidade induzida:

$$P(X=a) = P[\{w \in \Omega : X(w)=a\}]$$

$$\bullet P(X=1) = P[\{cen\ 1, cen\ 2\}] = \frac{18}{38} + \frac{810}{6859} = \frac{4059}{6859}$$

- $P(X = -1) = P[\text{cen } 3, \text{cen } 4] = \frac{900}{6859} + \frac{900}{6859} = \frac{1800}{6859}$
- $P(X = -3) = P(\text{cen } 5) = \frac{1000}{6859}$

Todos os outros valores tem probabilidade 0.

Neste caso  $P(X > 0) = P(X = 1) = \boxed{\frac{4059}{6859}} \approx 0,59$

Em um primeiro momento a estratégia parece interessante: você vai sair ganhando aproximadamente 60% das vezes. O problema é que quando você perde, existe a chance de perder até 3 reais (o que demoraria 3 estratégias vencedoras) e nas vitórias você só ganha 1 real. Ainda não fica claro se é bom ou ruim.

Chamaria de estratégia vencedora aquela que quando você repete várias vezes, em média, tem um resultado positivo. Para medir se essa estratégia é vencedora calcularemos a esperança de  $X$

$$E[X] = \int_{\mathbb{R}} X dP = \sum_{x \in \{-3, -1, 1\}} x P[X=x] = (-3)P[X=-3] + (-1)P[X=-1]$$

$$+ (1)P[X=1] = \frac{-3000}{6859} - \frac{1800}{6859} + \frac{4059}{6859} = \boxed{\frac{-39}{361}}$$

Vendo agora que a esperança é negativa fico convencido que a estratégia é ruim (como esperado). Em média perdemos cerca de 10 centavos cada vez que jogamos. Isso acontece justamente pois uma rodada ruim (cenário

5) mas da um prejuízo difícil de recuperar pois os ganhos das rodadas boas são pequenos.

Pelas leis dos grandes números podemos concluir que seguir essa estratégia nos levaria a falência quase certamente.

4) Escreveremos  $\vec{x} \in \mathbb{R}^2$  como  $\vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ . Temos

$$\Delta = (x_1 \ x_2) \begin{pmatrix} 3 & 0 \\ 0 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} =$$

$$= (x_1 \ x_2) \begin{pmatrix} 3x_1 \\ 5x_2 \end{pmatrix} = 3x_1^2 + 5x_2^2$$

Fixando agora  $\Delta = 1$ , temos

$$1 = \frac{x_1^2}{1/3} + \frac{x_2^2}{1/5}$$

Temos uma elipse centrada na origem sem rotações. cruzando o eixo  $x_1$  em  $\sqrt{1/3}$  e cruzando o eixo  $x_2$  em  $\sqrt{1/5}$ .

In [1]:

```
import numpy as np
from scipy.stats import norm
import matplotlib.pyplot as plt
```

**Problema:**

Gerar dados a partir de uma distribuição  $f(x) > 0$  não trivial e que não sei se satisfaz as condições para ser uma função densidade de probabilidade, isto é, não sei se vale

$$\int_{-\infty}^{\infty} f(x)dx = 1.$$

**Exemplo:**

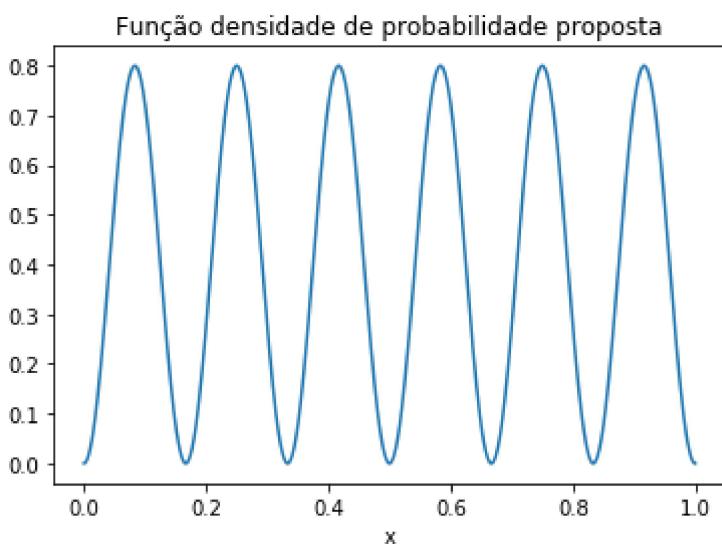
Suponha que temos

$$f(x) = 0.8 \sin^2(6\pi x) 1_{(0,1)}(x).$$

In [2]:

```
def f_(x):
    return np.select([x <= 0, (x > 0) & (x < 1), x >= 1], [0, 0.8*(np.sin(6*np.pi*x))**2, 0])

plt.plot(np.linspace(0,1,1000),f_(np.linspace(0,1,1000)))
plt.title('Função densidade de probabilidade proposta')
plt.xlabel('x')
plt.show()
```



Neste caso, poderíamos fazer

$$\int_0^1 0.8 \sin^2(6\pi x)dx = 0.4$$

e redefinir  $f(x)$  de maneira normalizada como

$$f(x) = 2 \sin^2(6\pi x) 1_{(0,1)}(x).$$

Mas mesmo com  $f(x)$  normalizada não é óbvio como criar amostras para  $f$ .

## Rejection Sampling

A ideia será criar pontos uniformemente em uma região e aceitar os pontos que caem na região desejada.

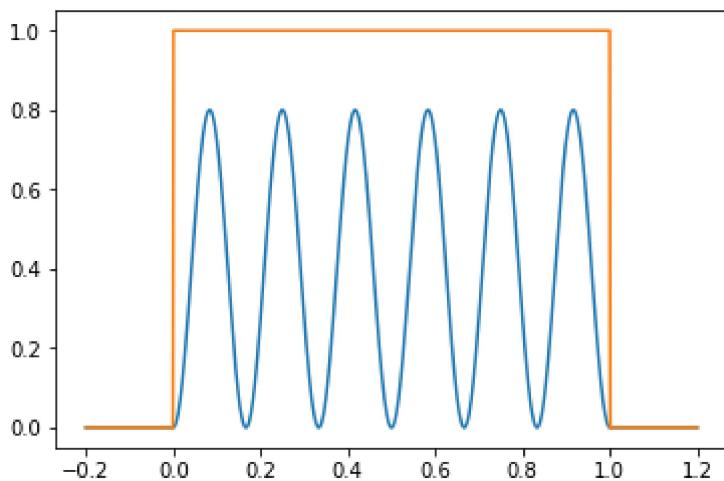
In [3]:

```
def g_(x):
    return np.select([x <= 0, (x > 0) & (x < 1), x >= 1], [0, 1, 0])
```

In [4]:

```
plt.plot(np.linspace(-0.2, 1.2, 1000), f_(np.linspace(-0.2, 1.2, 1000)))
plt.plot(np.linspace(-0.2, 1.2, 1000), g_(np.linspace(-0.2, 1.2, 1000)))

plt.show()
```

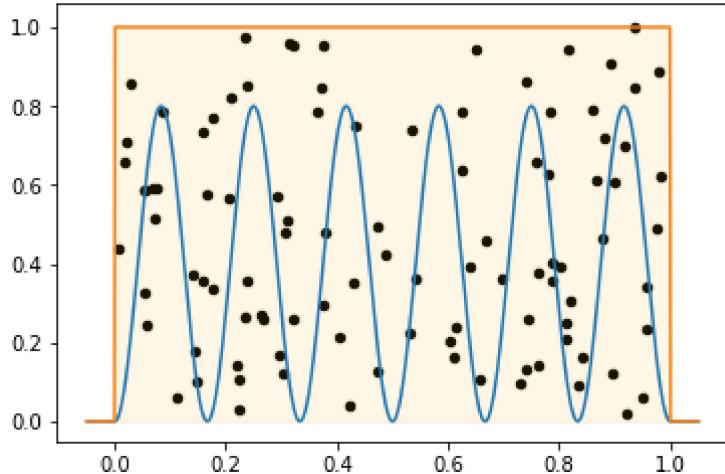


In [5]:

```
v1 = np.random.uniform(0, 1, 100)
v2 = np.random.uniform(0, 1, 100)
```

In [6]:

```
plt.scatter(v1,v2,s=20,c='k')
plt.plot(np.linspace(-0.05,1.05,1000),f_(np.linspace(-0.05,1.05,1000)))
plt.plot(np.linspace(-0.05,1.05,1000),g_(np.linspace(-0.05,1.05,1000)))
plt.fill_between(np.linspace(-0.05,1.05,1000), g_(np.linspace(-0.05,1.05,1000)),alpha=0.1,color='orange')
plt.show()
```

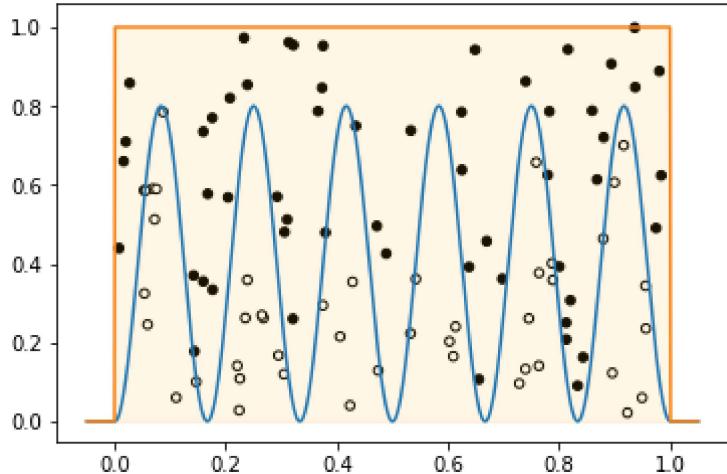


In [7]:

```
accept = v2<f_(v1)
```

In [8]:

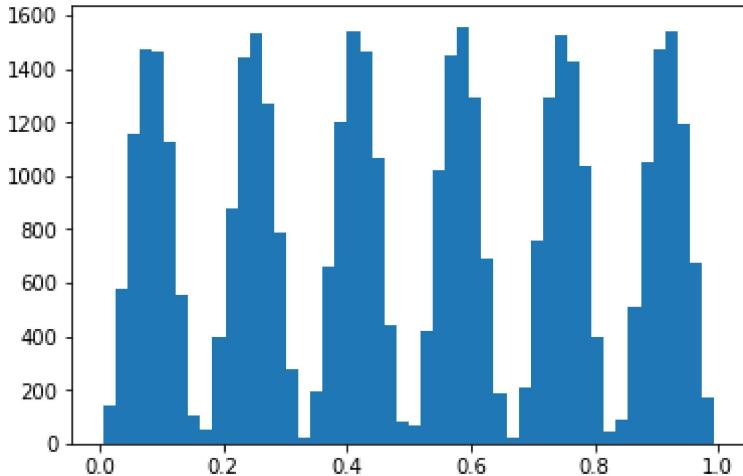
```
plt.scatter(v1,v2,s=20,c=accept,cmap='gray', edgecolors='k')
plt.plot(np.linspace(-0.05,1.05,1000),f_(np.linspace(-0.05,1.05,1000)))
plt.plot(np.linspace(-0.05,1.05,1000),g_(np.linspace(-0.05,1.05,1000)))
plt.fill_between(np.linspace(-0.05,1.05,1000), g_(np.linspace(-0.05,1.05,1000)),alpha=0.1,color='orange')
plt.show()
```



In [9]:

```
v1 = np.random.uniform(0,1,100000)
v2 = np.random.uniform(0,1,100000)
accept = v2<f_(v1)

plt.hist(v1[accept],bins=50)
plt.show()
```



## Problemas:

- Estamos jogando muitos caras fora porque a nossa região é "maior do que precisava ser".

In [10]:

```
sum(~accept)/len(accept)
```

Out[10]:

0.59995

- Imagina que tenhamos um  $f(x)$  que tenha suporte na reta inteira (como a gaussiana). Se eu limitar a fazer o sample só no quadrado eu não vou estar sendo realista com a distribuição de verdade de  $f(x)$ .

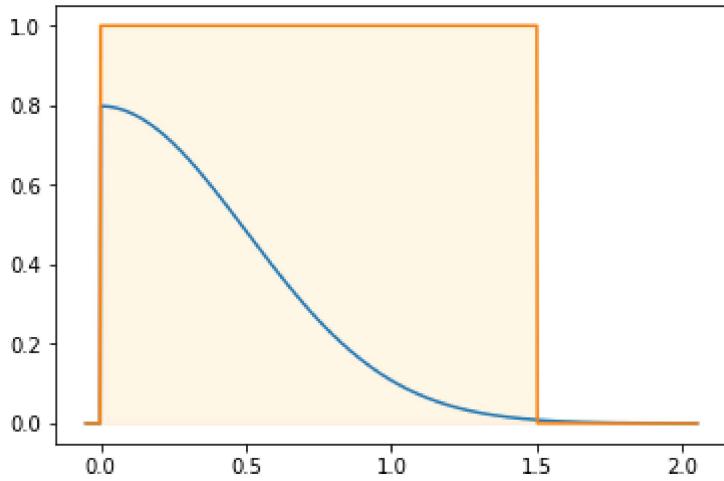
In [11]:

```
def f2_(x):
    return np.select([x <= 0, x>0], [0, norm(0, 0.5).pdf(x)])

def g2_(x):
    return np.select([x <= 0,(x > 0) & (x< 1.5), x>=1.5], [0, 1, 0])
```

In [12]:

```
plt.plot(np.linspace(-0.05,2.05,1000),f2_(np.linspace(-0.05,2.05,1000)))
plt.plot(np.linspace(-0.05,2.05,1000),g2_(np.linspace(-0.05,2.05,1000)))
plt.fill_between(np.linspace(-0.05,2.05,1000), g2_(np.linspace(-0.05,2.05,1000)),alpha=0.1,color='orange')
plt.show()
```



A ideia agora é fazer a região ser o mais próxima possível da função real. Pra gente minimizar rejeições.

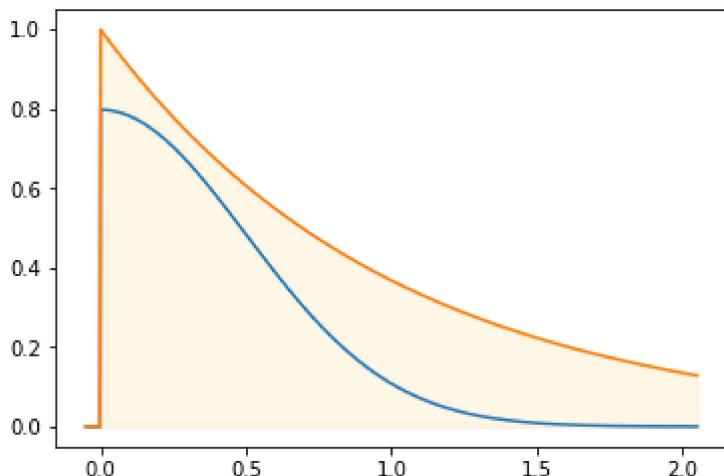
Precisamos usar distribuições que a gente conheça (saiba tirar amostras dela) e que tenham o formato adequado do que a gente tá tentando modelar.

In [13]:

```
def g3_(x):
    return np.select([x <= 0, x>0], [0, np.exp(-x)])
```

In [14]:

```
plt.plot(np.linspace(-0.05,2.05,1000),f2_(np.linspace(-0.05,2.05,1000)))
plt.plot(np.linspace(-0.05,2.05,1000),g3_(np.linspace(-0.05,2.05,1000)))
plt.fill_between(np.linspace(-0.05,2.05,1000), g3_(np.linspace(-0.05,2.05,1000)),alpha=0.1,color='orange')
plt.show()
```



Já está bem melhor, mas multiplicar por uma constante pode nos ajudar a evitar algumas dores de cabeça.

Se pegarmos

$$M = \max \frac{f(x)}{g(x)},$$

então  $Mg(x)$  sempre é maior que  $f(x)$  e além disso, elas ficam mais coladas.

Para estimar  $M$  podemos fazer um early burning para explorar os valores possíveis de  $x$  e daí escolher  $M$  que maximiza nesses pontos explorados.

In [15]:

```
b = np.random.exponential(size=10)
M = np.max(f2_(b)/g3_(b))
M
```

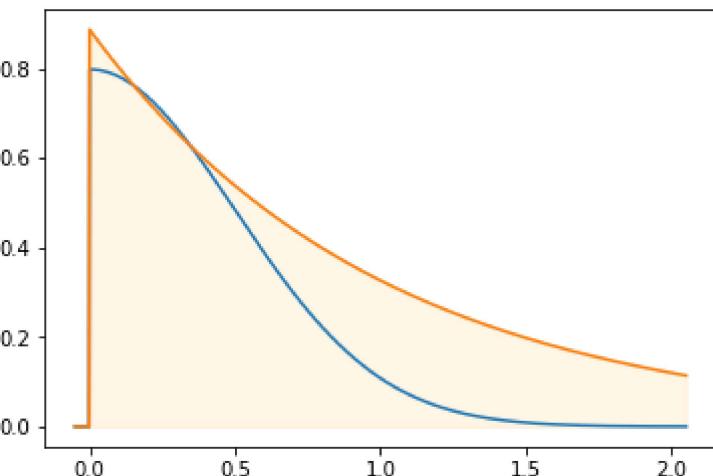
Out[15]:

0.8870502107180794

In [16]:

```
def g4_(x):
    return np.select([x <= 0, x>0], [0, M*np.exp(-x)])

plt.plot(np.linspace(-0.05,2.05,1000),f2_(np.linspace(-0.05,2.05,1000)))
plt.plot(np.linspace(-0.05,2.05,1000),g4_(np.linspace(-0.05,2.05,1000)))
plt.fill_between(np.linspace(-0.05,2.05,1000), g4_(np.linspace(-0.05,2.05,1000)),alpha=0.1,color='orange')
plt.show()
```



## Caso do exercício

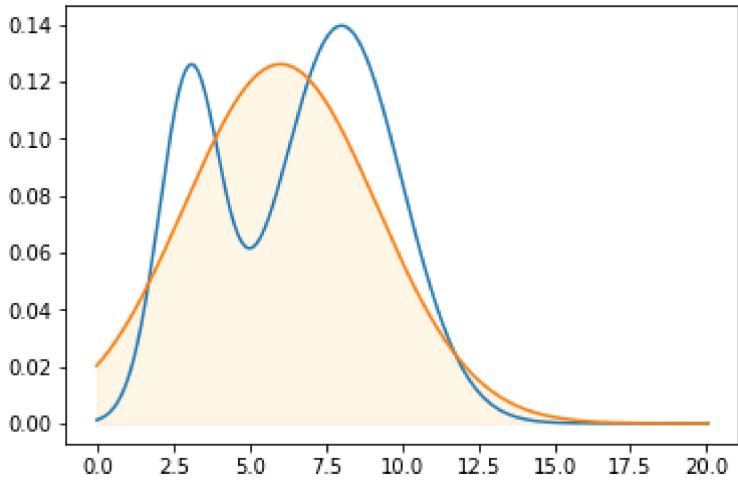
In [17]:

```
def g(x):
    return norm(6,np.sqrt(10)).pdf(x)

def f(x):
    return 0.3*norm(3,1).pdf(x) + 0.7*norm(8,np.sqrt(4)).pdf(x)
```

In [18]:

```
plt.plot(np.linspace(-0.05,20.05,1000),f(np.linspace(-0.05,20.05,1000)))
plt.plot(np.linspace(-0.05,20.05,1000),g(np.linspace(-0.05,20.05,1000)))
plt.fill_between(np.linspace(-0.05,20.05,1000), g(np.linspace(-0.05,20.05,1000)),alpha=0.1,color='orange')
plt.show()
```

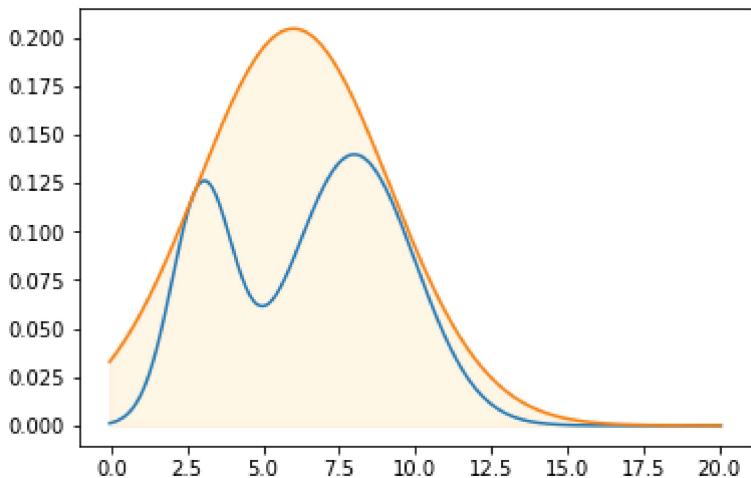


In [19]:

```
b = np.random.normal(6,np.sqrt(10), size=1000)
M = np.max(f(b)/g(b))
```

In [20]:

```
plt.plot(np.linspace(-0.05,20.05,1000),f(np.linspace(-0.05,20.05,1000)))
plt.plot(np.linspace(-0.05,20.05,1000),M*g(np.linspace(-0.05,20.05,1000)))
plt.fill_between(np.linspace(-0.05,20.05,1000), M*g(np.linspace(-0.05,20.05,1000)),alpha=0.1,color='orange')
plt.show()
```



In [22]:

```
sample = []
cont_reject = 0
while len(sample)<10000:
    u = np.random.uniform(0,1)
    z = np.random.normal(6,np.sqrt(10))
    if u < f(z)/(M*g(z)):
        sample.append(z)
    else:
        cont_reject +=1
M = np.max([M, f(z)/g(z)])
```

In [24]:

```
cont_reject
```

Out[24]:

6323

In [27]:

```
len(sample)
```

Out[27]:

10000

In [30]:

```
plt.hist(sample,bins=100)
plt.show()
```

