

Entregue todos os métodos em um arquivo chamado `lab9.py`. Importe a biblioteca `numpy` como `np`, `matplotlib.pyplot` como `plt` e `scipy.stats` como `st`.

1. (3 pontos) Crie a função `alturas` com parâmetro de entrada `n`. Crie um `np.ndarray` com `n` números aleatórios representando alturas corporais de uma população de `n` pessoas adultas. Use a distribuição normal com média 1.7 e desvio padrão 0.08 para modelar esses dados, ou seja, use a distribuição normal de `scipy` com a chamada `st.norm.rvs(loc = 1.7, scale = 0.08, size = n)`. Crie uma figura com o histograma desses dados divididos em 20 classes uniformes (`bins`), título do gráfico deve ser `Alturas`. Salve a figura no arquivo `alturas.png`. O valor de retorno é a `np.ndarray` com as alturas da população.

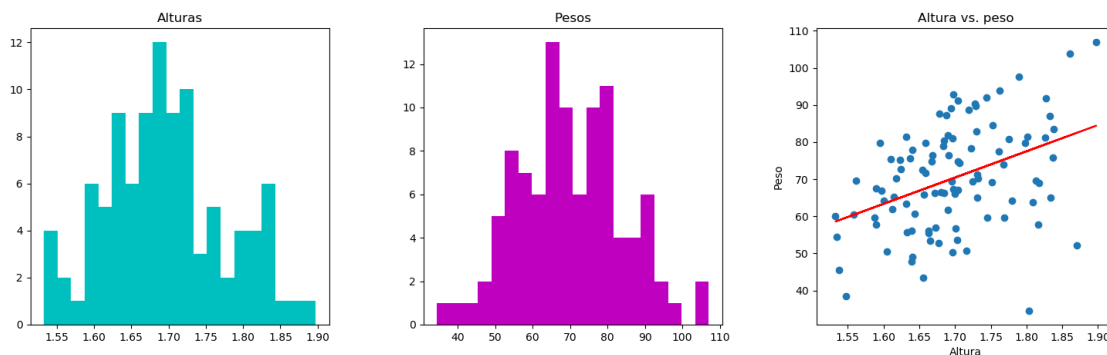


Figure 1: À esquerda é um histograma criado no exercício 1 para $n = 100$. No centro é um histograma criado no exercício 2 para alturas geradas no exercício 1 (com $n = 100$). À direita é um gráfico produzido no exercício 3 para alturas e pesos gerados nos exercícios anteriores (com $n = 100$). Observação: Como se trata de variáveis aleatórias, os gráficos mudam com cada chamada.

2. (3 pontos) Implemente o método `pesos` cujo parâmetro de entrada é um `np.ndarray` com as alturas de um conjunto de pessoas. Crie um `np.ndarray` com os índices da massa corporal (IMC) dessas pessoas, modele esses dados como variáveis aleatórias que seguem a distribuição normal com média 24.5 e desvio padrão 4.3. Portanto chame `st.norm.rvs(loc = 24.5, scale = 4.3, size = m)`, onde `m` é o comprimento do `np.ndarray` de entrada. Calcule os pesos das pessoas seguindo a equação $peso = imc \cdot altura^2$. Crie uma figura com o histograma dos pesos divididos em 20 classes uniformes (`bins`), título do gráfico deve ser `Pesos`. Salve a figura no arquivo `pesos.png`. O valor de retorno é o `np.ndarray` com os pesos.
3. (4 pontos) Crie o método `regressaoLinear`, cujos parâmetros de entrada são dois `np.ndarrays` que representam alturas e pesos de uma população. Assumimos que há uma relação linear entre a altura e o peso, portanto calcularemos os parâmetros a e b para poder desenhar o gráfico da função linear

$$y = ax + b,$$

onde x é altura e y é peso. O valor de retorno da chamada `st.linregress(x,y)` é uma tupla, cujo primeiro elemento é a variável `a` e o segundo é `b`. Calcule e desenhe o gráfico dessa função numa figura.

Desenhe na mesma figura também o gráfico de dispersão dos dados, onde o eixo `x` representa a altura e o eixo `y` representa o peso de novo. Lembrando que um gráfico de dispersão deve ser desenhado usando a função `scatter` do módulo `matplotlib.pyplot`. Inclua na figura os rótulos dos eixos e o título do gráfico que é `Altura vs. peso`. Salve a figura no arquivo `regressao.png`.

O valor de retorno do método é a tupla `(a,b)`.