Examen Parcial 1

Ramírez David R.

Department: Computación
Course: Pensamiento Computacional 108
Instructor: Benito Granados-Rojas, PhD.
Date: September 15, 2022

1 Introduction

En este trabajo se resuelven dos problemas planteados. El primero consite en programar un código en Python que construya una matriz cuadrada de mxm llena con los m^2 primeros números primos, posteriormente deberá de obtener la suma de todos los elementos que se encuentran en la diagonal principal y por encima de esta, el valor de m deberá de ser proporcionado por el usuario.

Por otro lado, el segundo problema consiste en analizar una serie de datos que contienen los resultados de una encuesta realizada a más de 200 personas sobre su estatura, talla de calzado y sexo; el objetivo es obtener el histograma, media, desviación estándar, la función gaussiana ajustada y una curva normal ajustada tanto para la estatura como para la talla de calzado. Además, se calculará la probabilidad de que dado una persona cualquiera esta se encuentre dentro de la primera desviación estándar para cada variable. Finalmente, se obtendrá la aproximación lineal para la relación entre la estatura y la talla de calzado tanto para la población general como para cada sexo.

2 Procedures and Results

2.1 Números Primos

Para el primer problema, se usó Numpy para facilitar ciertas peraciones. Primero se solicitó al usuario ingresar la dimensión m de la matriz A, para fines prácticos este valor cambió a $m = m^2$. Posteriormente se contruyó un vector de tamaño m lleno de zeros enteros sin signo de 8 bits (para facilitar la presentación de la matriz y reducir la memoria), es importante señalar que debido al tamaño del tipo de dato usado, la matriz solicitada puede ser, a lo mucho, de 7x7.

Una vez declarado el vector, se procede al llenado con números primos, para ello, se recorre una a una las m casillas del vector y se calcula el siguiente número primo para asignar dicho número a la casilla actual; sabemos que el primer número primo es el 2 por lo que la primera iteración asigna 2 a la primera casilla. A partir de la casilla 2, se revisa el último primo asignado p_{i-1} y se crea una variable temporal $k = p_{i-1} + 1$ pues sabemos que el siguiente número primo será al menos una unidad superior al anterior. Ahora, para comprobar que k es un número primo nos valemos del teorema fundamental de la aritmética que establece que todo número natural tiene una representación única como un producto de factores primos (Wikipedia 2022), por lo que un número natural N cualquiera será primo si no es múltiplo alguno de los primos menores a N, es decir, solo necesitamos revisar si el módulo es diferente de cero para los primos que ya hemos encontrado. De esta forma, recorremos uno a uno los primos encontrados hasta entonces y, si resulta que no es divisible entre ninguno de los primos anteriores, entonces k es primo y se asigna $p_i = k$ a la casilla actual, de lo contrario se inicia el cilco nuevamente con $k = k + 1$. El segmento de código descrito se muestra en Fig. 1.

Teminado el ciclo anterior, ya hemos obtenido un vector con los m primeros números primos. Para imprimir el vector en forma de una matriz sin corchetes usamos una variable auxiliar $n = \sqrt{m}$ y otra variable axiliar $s = ""$. Mediante dos ciclos anidados iguales dependientes de $i, j = [0 : n - 1]$ recorremos uno a uno los elementos del vector de primos asignando concatenado el elemento como string a la variable s, cada vez que $j = n - 1$ se incerta un salto de línea. Una vez recorrido todo el vector de primos se imprime la variable s. El segmento de código descrito se muestra en Fig. 2.

Finalmente, para obtener la suma de todos los elementos en y por encima de la diagonal principal de la matriz de primos reformamos el vector para convertirlo en una matriz cuadrada. Ahora, recorremos una a una las filas de esta nueva matriz y en cada fila recorremos las elementos de la fila partiendo del número de fila actual hasta llegar a la última columna, tal como se muestra en (1). En cada iteración sumamos el elemento actual a la variable auxiliar previamente declarada $sumDS$, al final del recorrido se imprime la suma total.El segmento de código descrito se muestra en Fig. 3.

```python
matrix = np.zeros(m, dtype=np.uint8)
for i in range(m):
    if i == 0:
        matrix[0] = 2
        continue
    k = matrix[i - 1]
    isprime = 0
    while (not isprime):
        k += 1
        isprime = 1
        for j in range(i):
            if k % matrix[j] == 0:
                isprime = 0
                break

    matrix[i] = k
```

Figure 1: Construcción de vector de primos

.

$$sumDS = \sum_{i=0}^{n} \sum_{j=i}^{n} a_{ij} \tag{1}$$

2.1.1 Resultados

El código se probó con matrices desde dimensión 3 hasta 7. En Fig. 4 se muestran los resultados para una matriz de 3x3 y en Fig. 5 se muestran los resultados para una matriz 7x7. Si se revisan estos resultados podemos comprobar que el código se comporta como se espera.

```
n = int(np.sqrt(m))
s = ""
for i in range(n):
    for j in range(n):
        s += "{}\t".format(matrix[n * i + j])
    s += "\n"
print("La matriz A de numeros primos consecutivos es:")
print(s)
```

Figure 2: Construcción de matriz de primos para imprimir

.

```
x=np.reshape(matrix, (n,n))
sumDS = 0
for i in range(n):
  for j in range(i,n):
    sumDS += x[i][j]

print("La suma de los elementos en la matriz diagonal superior es: {}".format(sumDS))
```

Figure 3: Suma de elementos en la diagonal superior

.

```
Introduce la dimension de la matriz A: 3
La matriz A de numeros primos consecutivos es:
2	3	5
7	11	13
17	19	23

La suma de los elementos en la matriz diagonal superior es: 57
```

Figure 4: Resultados para una matriz de 3x3

.

```
Introduce la dimension de la matriz A: 7
La matriz A de numeros primos consecutivos es:
2	3	5	7	11	13	17
19	23	29	31	37	41	43
47	53	59	61	67	71	73
79	83	89	97	101	103	107
109	113	127	131	137	139	149
151	157	163	167	173	179	181
191	193	197	199	211	223	227

La suma de los elementos en la matriz diagonal superior es: 2013
```

Figure 5: Resultados para una matriz de 7x7

.

2.2 Experiment/simulation/measurement 2

Para este problema, primero se corrigieron los datos de la encuesta para que no se encontraran problemas durante su manipulación. Posteriormente, extraemos los datos de cada columna en una variable distinta. Para visualizar los datos de talla y estatura se graficaron tal y como fueron extraidos, dichas gráficas se muestran en Fig. 6 y en Fig. 7. Ahora, usando los comandos $mean(A)$ y $std(A)$ obtuvimos las medidas de tendencias central (μ, σ) tanto para la estatura como para la talla.

$$\mu_{estatura} = 167.15cm \quad \mu_{talla} = 25.75cm$$
$$\sigma_{estatura} = 9.46cm \quad \sigma_{talla} = 1.91cm$$

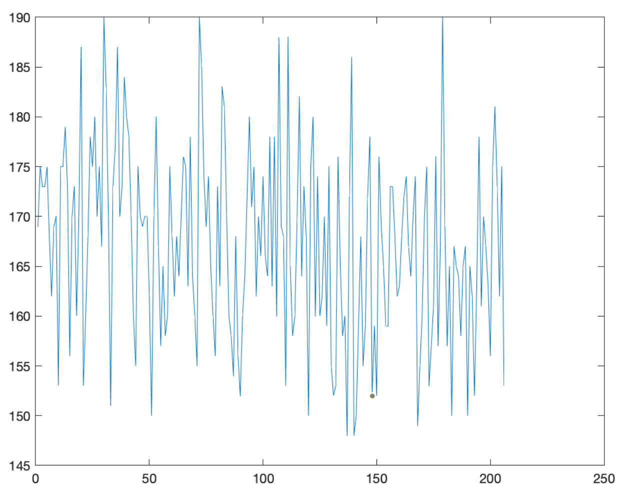

Figure 6: Gráfica de estaturas

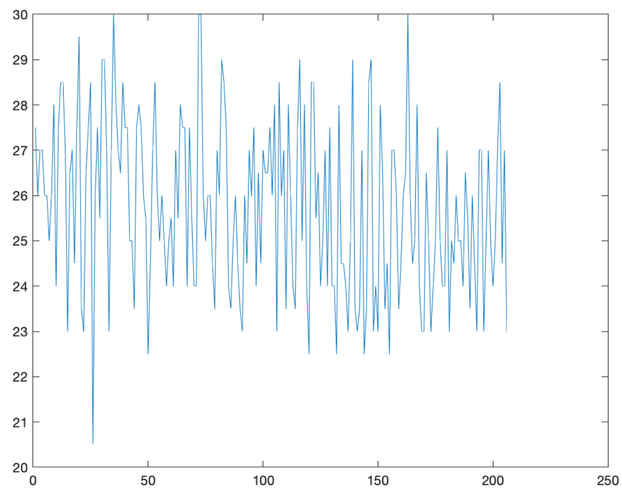

Figure 7: Gráfica de tallas

Ahora, para obtener la función gaussiana que mejor ajuste a cada dataset (estatura y talla), se siguió la ecuación de una distribución normal (2). Por lo tanto, las funciones gaussianas que mejor ajustan a los datos de estatura y talla son (3) y (4) respectivamente.

$$G_{\mu,\sigma}(x) = \frac{1}{\sigma\sqrt{2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2}$$

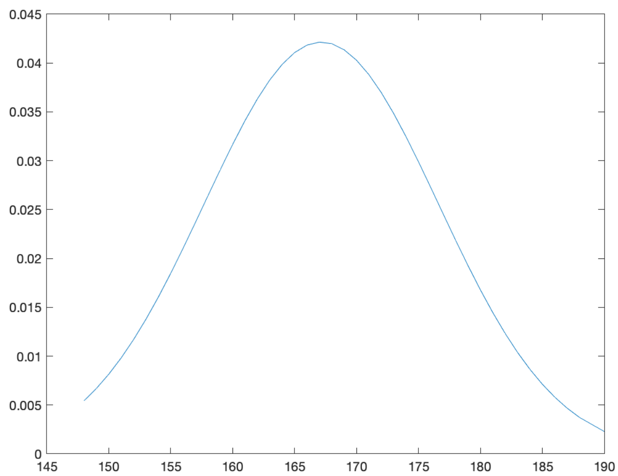

Figure 8: Curva Normal de estaturas

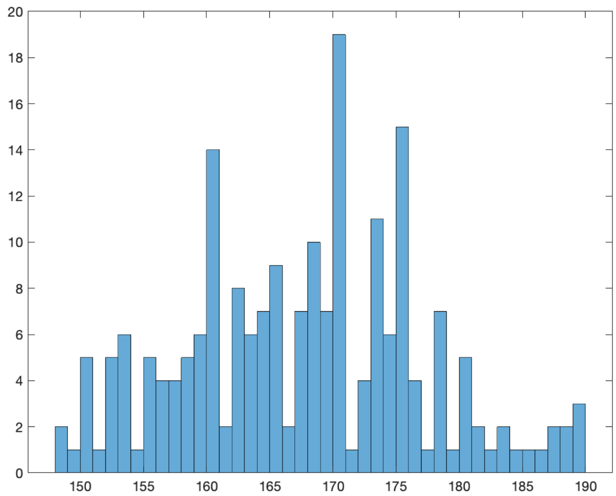

Figure 9: Histograma de estaturas

$$G_{estatura}(x) = \frac{1}{9.46\sqrt{2}} e^{-\frac{(x-167.15)^2}{2(9.46)^2}} \tag{3}$$

$$G_{talla}(x) = \frac{1}{1.91\sqrt{2}} e^{-\frac{(x-25.75)^2}{2(1.91)^2}} \tag{4}$$

Para graficar las curvas, primero se ordenaron los datos con el comando $sort(data)$. Después, se construyó un vector con la evalución de la función gaussiana respectiva para cada dato del dataset. Además, se graficó el histograma de cada dataset. Los gráficos de estaura se muestran en Fig. 8 y en Fig. 9, mientras que los gráficos de talla se muestran en Fig. 10 y en Fig. 11.

Con las funciones gaussianas obtenidas, ahora obtendremos la probabilidad de que una persona cualquiera se encuentre dentro de la primera desviación estándar para cada variable. Sin embargo, como podemos apreciar en (5), esta probabilidad no depende de los datos, solo se tiene que obtener la probabilidad para $Z = 1$ y para $Z = -1$ y restarlas. Esto nos dará $P = 68.26\%$ como se puede apreciar en (6). De esta forma, la probabilidad de que una persona cualquiera esté dentro de la primera desviación estándar es de 68.26% para ambos datasets.

$$P(Z_1) - P(Z_0) = P(\frac{\mu + \sigma - \mu}{\sigma}) - P(\frac{\mu - \sigma - \mu}{\sigma}) = P(1) - P(-1) \tag{5}$$

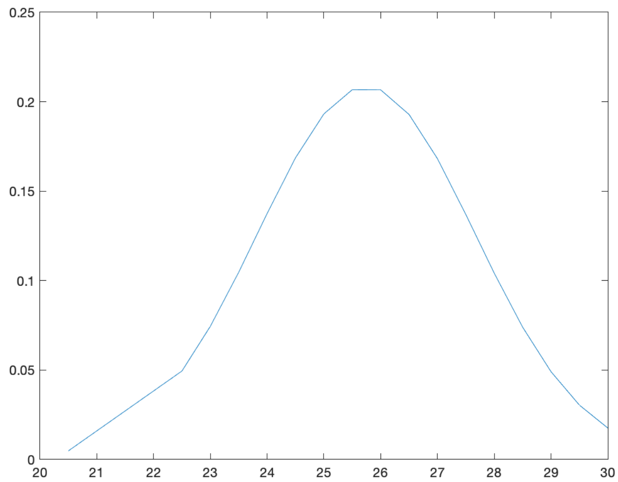

Figure 10: Curva Normal de tallas

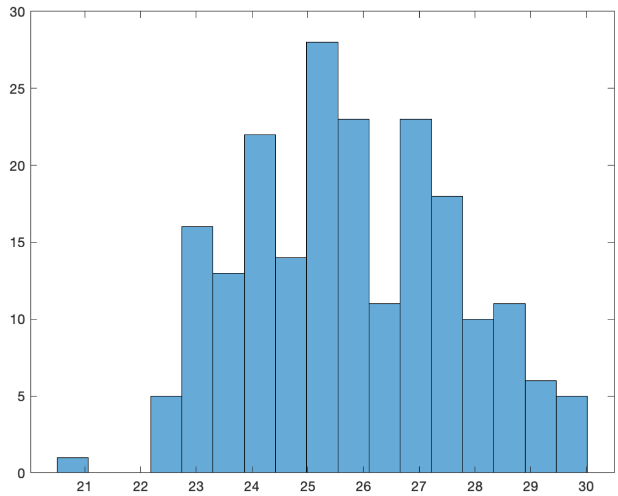

Figure 11: Histograma de tallas

$$P = 0.8413 - 0.1587 = 0.6826 \tag{6}$$

Finalmente, para obtener la relación entre estatura y talla, primero se ordenaron los datos, pero ahora como un solo conjunto mediante el comando $sortrows([estura, talla])$. Después, se obtuvo la aproximación lineal de esta relación mediante el comando $fit(estaturaOrdenada, tallaOrdenada,' poly1')$, la ecuación obtenida es (7). La gráfica de la relación y la aproximación lineal se muestra en Fig. 12.

$$talla(estatura) = 0.1618 * estatura - 1.299 \tag{7}$$

Ahora, para separar los datos por sexo, primero se convirtio la variable sexo a una variable númerica con 1 para mujeres y 0 para hombres, recorriendo uno a uno los elementos de la variable sexo y asignando el correspondiente valor a un nuevo vector sx. Esto hizo posible un reordenamiento similar al anterior pero tomando como primer eje al sexo con el comando $seg = sortrows([sx'estatura, talla])$. Con los datos ordenados entre hombre y mujer, procedemos a separalos contando el número de 1's en el nuevo vector sx con el comando $nm = sum(sx == 1)$, ahora, podemos separar los datos con el índice de 1 a nm y de nm a 206. Con los datos separados, obtendremos una aproximación lineal para cada nuevo dataset

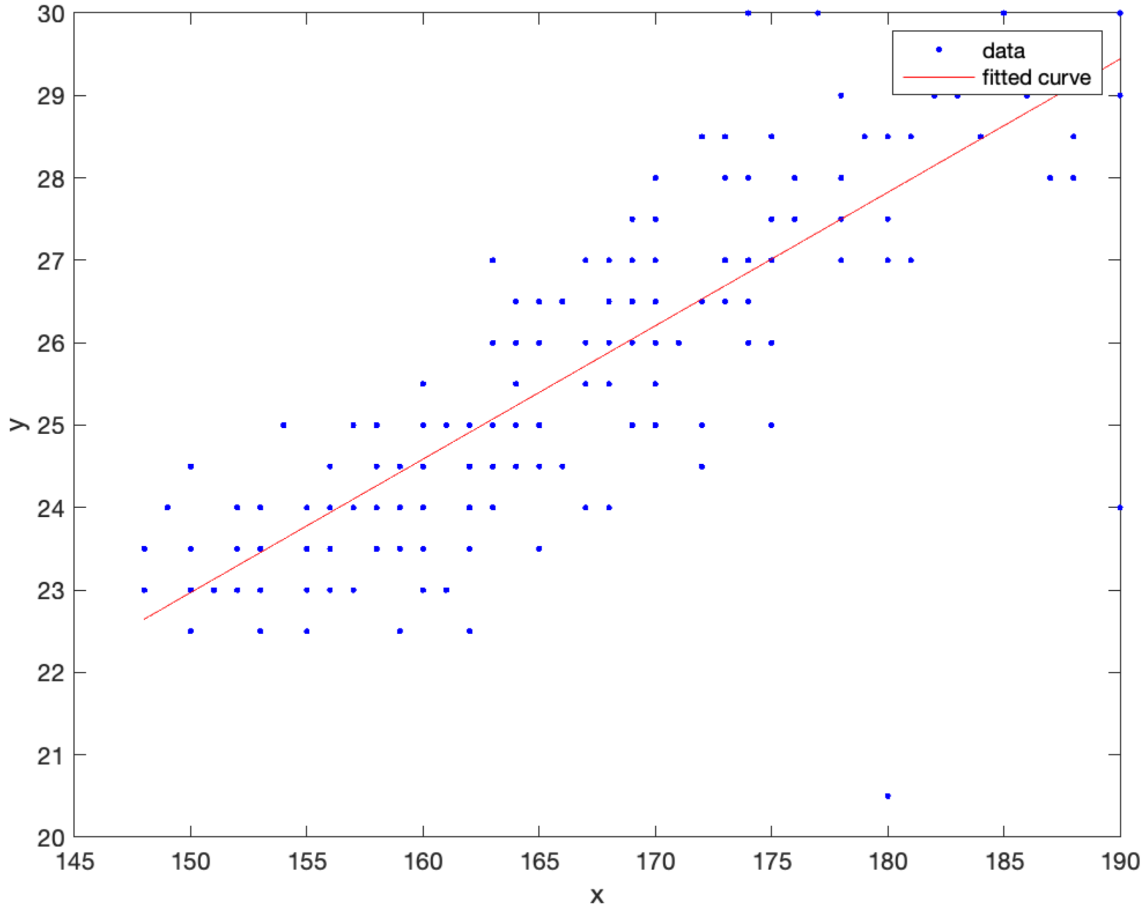

Figure 12: Relación estatura/talla para la población general

.

mediante el comando fit y procederemos a gráficar estas nuevas relaciones con su respectiva aproximación.

La aproximación obtenida para el dataset de hombres es (8) y para el dataset de mujeres es (9). El gráfico para hombres se muestra en Fig. 13, mientras que el gráfico para mujeres se muestra en Fig. 14.

$$talla_h(estatura_h) = 0.1177 * estatura_h - 6.77 \tag{8}$$

$$talla_m(estatura_m) = 0.1107 * estatura_m - 6.523 \tag{9}$$

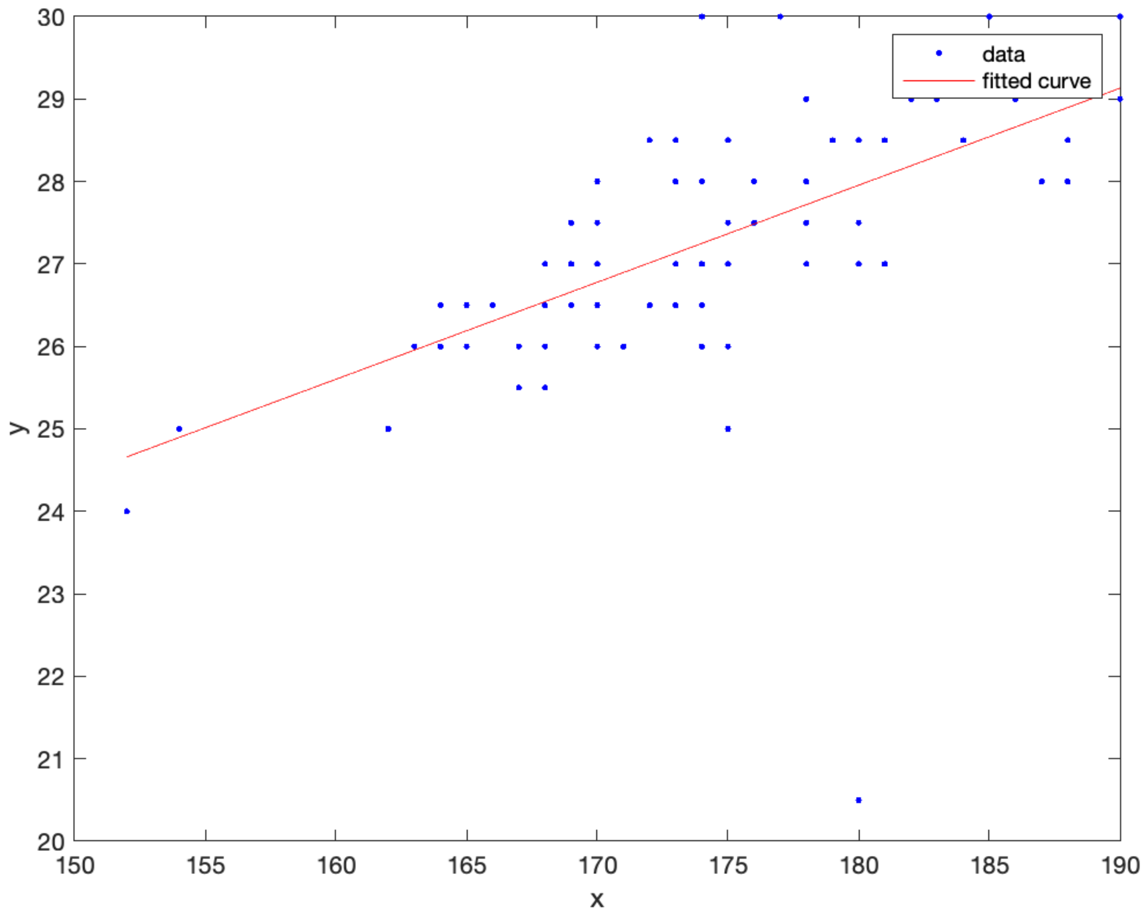

Figure 13: Relación estatura/talla para la población de hombres

.

2.2.1 Análisis

Dados los datos obtenidos, es acertado decir que la mayoría de personas tendrá una talla de calzado cercana a 25.5cm, mientras que las personas tienen una estatura promedio de 168cm, cabe destacar que esto es para la población general, si se separarán los datos entre hombres y mujeres posiblemente estas tendencias serían diferentes, también es importante mencionar que el número de mujeres y hombres entrevistados es exatamente el mismo (103), por lo que el peso que tienen sobre la estadística es el mismo.

Por otro lado, encontramos que existe una correlación lineal entre la estatura y la talla de calzado tanto para la población general como para las poblaciónes separadas. Resulta interesante mecionar que cuando los datos se aproximan a

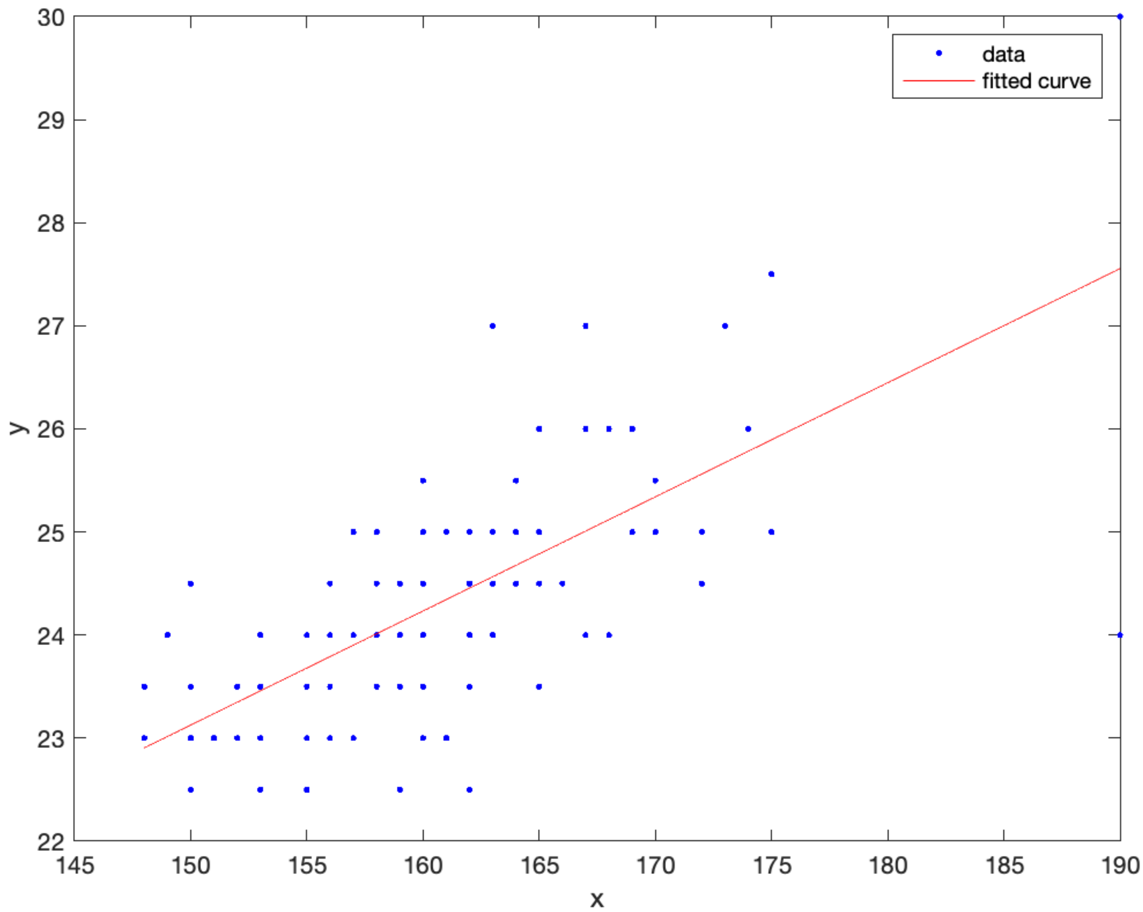

Figure 14: Relación estatura/talla para la población de mujeres

.

la parte superior derecha de la gráfica la mayoría son hombres, mientras que cuando los datos se aproximan a la parte superior izquierda son mujeres. Más aún, dada las ordenadas al origen de cada aproximación lineal separadas ($talla_h$ y $talla_m$), podemos afirmar que los hombres tienden a tener tallas más grandes que las mujeres, mientras que el peso de la estatura en la talla de calzado es muy similar entre hombres y mujeres. Esto quiere decir, que mientras más alta es una persona cualquiera más grande será su talla de calzado, solo que los hombres parten de una talla de calzado de por sí superior. Finalmente, es importante resaltar que los punto de esta relación tanto general como separada se encuentran muy dispersos, por lo que, aunque la correlación existe, no hay forma de afirmar que una persona alta siempre tendrá pies grandes.

3 Conclusions

A través de este trabajo se recopilan algunos métodos para abordar diversos problemas, tanto problemas meramente matemáticos como problemas estadísticos. Es interensante destacar el uso de dos herramientas distintas, Python y MatLab, niguna tiene una superioridad sobre la otra, su utilidad radica principalmente en la experiencia del usuario en la herramienta y el contexto del problema a resolver. Por ello, es importante adquirir habilidades en varias herramientas pero centrarse en las herramientas que se adapten mejor a la mayoría de problemas que encaramos, también es imporante un análisis del problema antes de optar por usar una herramienta u otra, una mala elección podría prolongar la resolución del problema inecesariamente.

4 Repository

https://github.com/A01338802/ComputacionAplicada_E1.git

References

[1] Wikipedia. *Prime Number*. 2022. URL: https://en.wikipedia.org/wiki/Prime_number.