Optimización

Pamela Lincoqueo

Abril, 2021

1. Ajuste de Curvas

1.1. Ajuste Lineal: coeficiente de Pearson

Un método que se puede utilizar es el coeficiente de Pearson (EL UNIVERSO DE LAS MATEMÁTICAS, 2020), cuya fórmula es:

$$r_{xy} = \frac{n \cdot (\Sigma xy) - [(\Sigma x) \cdot (\Sigma y)]}{\sqrt{\left[n \cdot (\Sigma x^2) - (\Sigma x)^2\right]\left[n \cdot (\Sigma y^2) - (\Sigma y)^2\right]}}$$

Se utilizan los datos experimentales que salen en uno de los documentos presentados en clases. Se deben obtener las sumatorias respectivas de la fórmula presentada.

	x	y	$x \cdot y$	x^2	y^2
	0.1	0.2	0.02	0.01	0.04
	0.2	0.3	0.06	0.04	0.09
	0.3	0.45	0.135	0.09	0.2025
	0.4	0.55	0.22	0.16	0.3025
	0.5	0.6	0.3	0.25	0.36
	0.6	0.7	0.42	0.36	0.49
	0.7	0.75	0.525	0.49	0.5625
	0.8	0.8	0.64	0.64	0.64
	0.9	0.8	0.72	0.81	0.64
	1.0	0.8	0.8	1.0	0.64
Sumatorias	5.5	5.95	3.84	3.85	3.9675

Cuadro 1: Datos y sumatorias.

Entonces se procede a reemplazar en la fórmula.

$$r_{xy} = \frac{(10 \cdot 3{,}84) - (5{,}5 \cdot 5{,}95)}{\sqrt{\left[(10 \cdot 3{,}85) - (5{,}5)^2\right]\left[(10 \cdot 3{,}9675) - (5{,}95)^2\right]}}$$

$$r_{xy} = \frac{38{,}4 - 32{,}725}{\sqrt{(38{,}5 - 30{,}25)\,(39{,}675 - 35{,}4025)}} = \frac{5{,}675}{\sqrt{8{,}25 \cdot 4{,}2725}} = \frac{5{,}675}{7{,}745} = 0{,}732$$

Para obtener la función, se utiliza:

$$y = m \cdot x + b$$

Así que solo se despeja la pendiente m y la variable b.

$$m = \frac{n \cdot (\Sigma xy) - [(\Sigma x) \cdot (\Sigma y)]}{n \cdot (\Sigma x^2) - (\Sigma x)^2} = \frac{(10 \cdot 3{,}84) - (5{,}5 \cdot 5{,}95)}{(10 \cdot 3{,}85) - (5{,}5)^2} = \frac{5{,}675}{8{,}25} = 0{,}68\widehat{78}$$

$$b = \frac{\Sigma y - m \cdot \Sigma x}{n} = \frac{5{,}95 - 0{,}6878 \cdot 5{,}5}{10} = \frac{5{,}95 - 3{,}7829}{10} = \frac{2{,}1672}{10} = 0{,}2167$$

Reemplazando queda:

$$y = 0{,}6878x + 0{,}21671 \Rightarrow f(x) = 0{,}6878x + 0{,}21671$$

Obteniendo así, el siguiente gráfico:

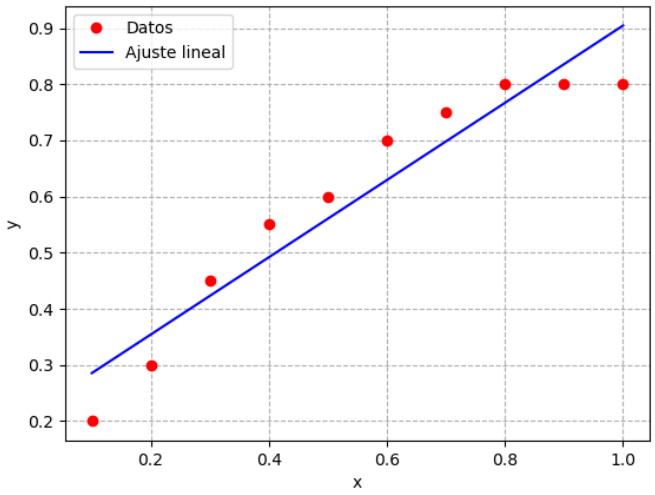

Figura 1: Gráfico de ajuste lineal.

1.2. Ajuste Polinomial

Para obtener el ajuste de curva más cercano a cada punto, se puede usar la fórmula para ajuste polinomial, del cual se rige de la siguiente forma:

$$p(x) = a_0 + a_1x + a_2x^2 + ... + a_{n-1}x^{n-1} = y_1$$

Ya que no encontré una forma de hacer sistema de ecuaciones en python, se utilizó el sitio web "matrix-calc.org"para desarrollar a mano.

$$\begin{aligned}
1a_0 + 0{,}1a_1 + 0{,}1^2a_2 + 0{,}1^3a_3 + 0{,}1^4a_4 + 0{,}1^5a_5 + 0{,}1^6a_6 + 0{,}1^7a_7 + 0{,}1^8a_8 + 0{,}1^9a_9 &= 0{,}2\\
1a_0 + 0{,}2a_1 + 0{,}2^2a_2 + 0{,}2^3a_3 + 0{,}2^4a_4 + 0{,}2^5a_5 + 0{,}2^6a_6 + 0{,}2^7a_7 + 0{,}2^8a_8 + 0{,}2^9a_9 &= 0{,}3\\
1a_0 + 0{,}3a_1 + 0{,}3^2a_2 + 0{,}3^3a_3 + 0{,}3^4a_4 + 0{,}3^5a_5 + 0{,}3^6a_6 + 0{,}3^7a_7 + 0{,}3^8a_8 + 0{,}3^9a_9 &= 0{,}45\\
1a_0 + 0{,}4a_1 + 0{,}4^2a_2 + 0{,}4^3a_3 + 0{,}4^4a_4 + 0{,}4^5a_5 + 0{,}4^6a_6 + 0{,}4^7a_7 + 0{,}4^8a_8 + 0{,}4^9a_9 &= 0{,}55\\
1a_0 + 0{,}5a_1 + 0{,}5^2a_2 + 0{,}5^3a_3 + 0{,}5^4a_4 + 0{,}5^5a_5 + 0{,}5^6a_6 + 0{,}5^7a_7 + 0{,}5^8a_8 + 0{,}5^9a_9 &= 0{,}6\\
1a_0 + 0{,}6a_1 + 0{,}6^2a_2 + 0{,}6^3a_3 + 0{,}6^4a_4 + 0{,}6^5a_5 + 0{,}6^6a_6 + 0{,}6^7a_7 + 0{,}6^8a_8 + 0{,}6^9a_9 &= 0{,}7\\
1a_0 + 0{,}7a_1 + 0{,}7^2a_2 + 0{,}7^3a_3 + 0{,}7^4a_4 + 0{,}7^5a_5 + 0{,}7^6a_6 + 0{,}7^7a_7 + 0{,}7^8a_8 + 0{,}7^9a_9 &= 0{,}75\\
1a_0 + 0{,}8a_1 + 0{,}8^2a_2 + 0{,}8^3a_3 + 0{,}8^4a_4 + 0{,}8^5a_5 + 0{,}8^6a_6 + 0{,}8^7a_7 + 0{,}8^8a_8 + 0{,}8^9a_9 &= 0{,}8\\
1a_0 + 0{,}9a_1 + 0{,}9^2a_2 + 0{,}9^3a_3 + 0{,}9^4a_4 + 0{,}9^5a_5 + 0{,}9^6a_6 + 0{,}9^7a_7 + 0{,}9^8a_8 + 0{,}9^9a_9 &= 0{,}8\\
1a_0 + 1{,}0a_1 + 1{,}0^2a_2 + 1{,}0^3a_3 + 1{,}0^4a_4 + 1{,}0^5a_5 + 1{,}0^6a_6 + 1{,}0^7a_7 + 1{,}0^8a_8 + 1{,}0^9a_9 &= 0{,}8
\end{aligned}$$

El cual desarrollado con dicho sitio, nos arroja el siguiente resultado:

$$
\begin{aligned}
a_0 &= \frac{-38}{5} \\
a_1 &= \frac{90473}{420} \\
a_2 &= \frac{-801991}{336} \\
a_3 &= \frac{7141315}{504} \\
a_4 &= \frac{-7204225}{144} \\
a_5 &= \frac{7931375}{72} \\
a_6 &= \frac{-2748125}{18} \\
a_7 &= \frac{8153125}{63} \\
a_8 &= \frac{-3859375}{63} \\
a_9 &= \frac{781250}{63}
\end{aligned}
$$

Sacando los cálculos, se puede obtener la función polinomial.

$$f(x) = -7{,}6 + 215{,}412x - 2386{,}878x^2 + 14169{,}276x^3 - 50029{,}34x^4 + 110157{,}986x^5 - 152673{,}611x^6 + \ldots$$
$$129414{,}683x^7 - 61259{,}921x^8 + 12400{,}794x^9$$

Entonces, si utilizamos esta nueva función, se obtiene la siguiente curva en el gráfico.

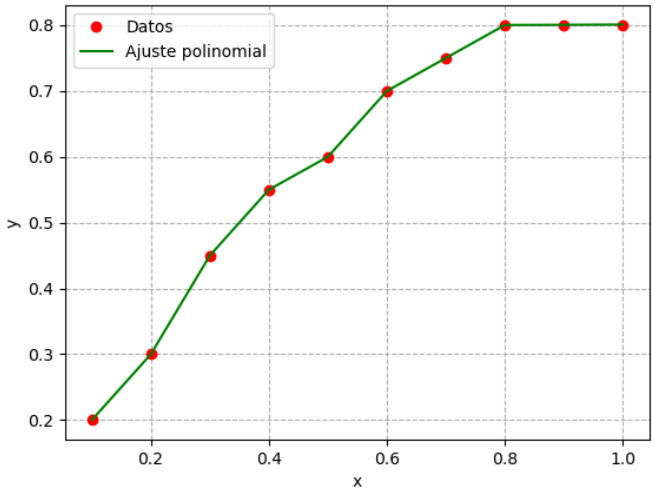

Figura 2: Gráfico de ajuste polinomial.

1.3. Modelo Higuchi

De los modelos a elegir, se seleccionó el modelo Higuchi, cuya fórmula es:

$$u(t) = at^{\frac{1}{2}}$$

Del cual es necesario despejar la variable a.

$$
\begin{aligned}
u(0{,}3) &= a \cdot (0{,}3)^{\frac{1}{2}} \\
0{,}45 &= a \cdot 0{,}54772255750516611345696978280 08 \\
\frac{0{,}45}{0{,}54772255750516611345696978280 08} &= a \\
0{,}82158 &\approx a
\end{aligned}
$$

Por ende, la fórmula para despejar a generalizado, sería:

$$a = \frac{u(t)}{t^{\frac{1}{2}}}$$

Entonces despejando el a en cada iteración, se obtiene el siguiente gráfico.

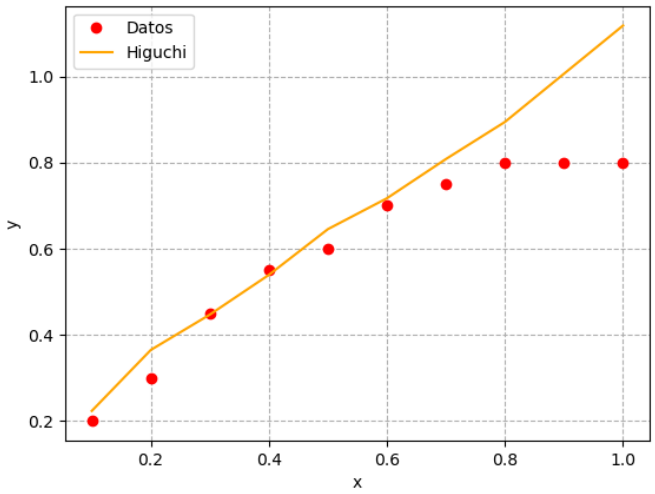

Figura 3: Gráfico con modelo Higuchi.

1.4. Código Python

Se presenta el código utilizado para generar los gráficos.

```
import matplotlib.pyplot as plt
import numpy as np

def plot_config():
  plt.grid(linestyle="--")
  plt.xlabel("x")
  plt.ylabel("y")

def higuchi(u,t):
  y=[]; a=[]
  for i in range(len(t)):
    a.append(u[i]/t[i])
    y.append(a[i]*(t[i]**(1/2)))
  return y

def pearson(x,y):
  n=len(x); sumX=0; sumY=0; sumXY=0; sumX2=0; sumY2=0
  for i in range(n):     sumX = sumX + x[i]
  for i in range(len(y)): sumY = sumY + y[i]
  for i in range(n):     sumXY = sumXY + x[i]*y[i]
  for i in range(n):     sumX2 = sumX2 + x[i]**2
  for i in range(n):     sumY2 = sumY2 + y[i]**2
  m = ((n*sumXY)-(sumX*sumY))/((n*sumX2)-((sumX)**2))
  b = (sumY-(m*sumX))/n
  return m,b

def lineal(m,x,b):
  y=[]
  for i in range(len(x)):
    a=(m*x[i]+b)
    y.append(a)
  return y

def polinomial(x):
  y=[]
  for i in range(len(x)): y.append(-7.6+(215.412*x[i])+(-2386.878*x[i]**2)+(14169.276*x[i
    ]**3)+(-50029.34*x[i]**4)+(110157.986*x[i]**5)+(-152673.611*x[i]**6)+(129414.683*x[i
    ]**7)+(-61259.921*x[i]**8)+(12400.794*x[i]**9))
  return y

```

```
# -------- Datos experimentales
x = [0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0]
y = [0.2,0.3,0.45,0.55,0.6,0.7,0.75,0.8,0.8,0.8]

m,b = pearson(x,y)
ajuste_lineal = lineal(m,x,b)#y=m*x+b
ajuste_poli = polinomial(x)#p(x)=a_0+a_1x+a_2x^2+...+a_n-1x^n-1
higuchi=higuchi(x,y)

# -------- Plotting
plt.plot(x,y, "ro",label="Datos")
plt.plot(x,ajuste_lineal,"b",label="Ajuste lineal")
plt.plot(x,ajuste_poli,"g",label="Ajuste polinomial")
plt.plot(x,higuchi,"orange",label="Higuchi")
plot_config()
plt.legend()
plt.show()
```

2. Optimización

Para esta sección se utilizará la función polinomial que se obtuvo con anterioridad.

$$f(x) = -7{,}6 + 215{,}412x - 2386{,}878x^2 + 14169{,}276x^3 - 50029{,}34x^4 + 110157{,}986x^5 - 152673{,}611x^6 + \dots$$

$$129414{,}683x^7 - 61259{,}921x^8 + 12400{,}794x^9$$

Se procede a derivar la función:

$$f'(x) = 215{,}412 - 4773{,}574x + 42507{,}828x^2 - 200117{,}36x^3 + 550789{,}93x^4 - 916041{,}666x^5 + 905902{,}781x^6 -$$

$$490079{,}368x^7 + 111607{,}146x^8$$

Se iguala a 0 para obtener los puntos críticos.

$$x_1 \approx 0{,}13768; x_2 \approx 0{,}19557; x_3 \approx 0{,}96748; x_4 \approx 0{,}86335$$

3. Bibliografía

EL UNIVERSO DE LAS MATEMÁTICAS. (2020). *Ajuste de curvas.* Descargado de `https://youtu.be/8UNqLjCfAbA`

Ivan F. Carmona Nogales. (2020). *Ajuste polinomial (algebra matricial).* Descargado de `https://youtu.be/r-WbXVomzFU`