

✓ Actividad 5.1 Matriz HAT - Equipo 10

Dhali Tejeda

Enrique Mora

2.Verificar los resultados utilizando Python para los valores del primer inciso.

```
import numpy as np

# Datos
X1 = np.array([2, 3, 5, 7, 9]) # Variable independiente
y = np.array([5, 8, 7, 10, 12]) # Variable dependiente

# Añadimos una columna de 1s a X1 para el término de intercepto
X = np.column_stack((np.ones(len(X1)), X1))

# Paso 1: Calcular los coeficientes de regresión β
# β = (X^T * X)^-1 * X^T * y
beta = np.linalg.inv(X.T @ X) @ (X.T @ y)
print("Coeficientes de regresión (β):", beta)

# Paso 2: Calcular la Matriz HAT (H)
# H = X * (X^T * X)^-1 * X^T
H = X @ np.linalg.inv(X.T @ X) @ X.T
print("\nMatriz HAT (H):")
print(H)

# Paso 3: Calcular los valores predichos y^
y_pred = H @ y
print("\nValores predichos (y^):", y_pred)
```

Coeficientes de regresión (β): [3.86585366 0.87195122]

Matriz HAT (H):

```
[[ 0.51219512  0.41463415  0.2195122  0.02439024 -0.17073171]
 [ 0.41463415  0.34756098  0.21341463  0.07926829 -0.05487805]
 [ 0.2195122  0.21341463  0.20121951  0.18902439  0.17682927]
 [ 0.02439024  0.07926829  0.18902439  0.29878049  0.40853659]
 [-0.17073171 -0.05487805  0.17682927  0.40853659  0.6402439 ]]
```

Valores predichos (y^): [5.6097561 6.48170732 8.22560976 9.9695122 11.71341463]

Podemos observar que mis resultados fueron parecidos más no exactos y esto es debido a que yo utilicé los valores con 4 decimales en vez de utilizar los valores como fracciones para no perder información. Esto fue un error de mi parte.

3.Utilizando los Datos “Cirugía de Hígado” obtener la matriz HAT, los valores predichos, \hat{y} y los coeficientes de la ecuación de regresión β utilizando el método de matrices.

```
import pandas as pd
from sklearn.preprocessing import MinMaxScaler

# Cargar los datos
df = pd.read_csv('/content/sample_data/Datos Modelos Lineales (1).csv')

# Identificar las variables categóricas (0s y 1s)
categorical_vars = ['genero', 'alcohol_moderado', 'alcohol_severo']

# Separar las variables numéricas
numeric_vars = df.drop(columns=categorical_vars)

# Aplicar MinMax a las variables numéricas
scaler = MinMaxScaler()
df[numeric_vars.columns] = scaler.fit_transform(numeric_vars)

print(df.head())
```

```
factor_coagulacion  indice_pronostico  funcion_de_enzima \
0                0.476744             0.593407           0.604167
```

```

1      0.290698      0.560440      0.447917
2      0.558140      0.538462      0.625000
3      0.453488      0.714286      0.187500
4      0.604651      0.626374      0.958333

funcion_de_higado  edad  genero  alcohol_moderado  alcohol_severo \
0      0.326855  0.500      0      1      0
1      0.169611  0.225      0      0      0
2      0.250883  0.625      0      0      0
3      0.224382  0.450      0      0      0
4      0.628975  0.375      0      0      1

sobrevivencia_dias
0      0.237743
1      0.102683
2      0.244681
3      0.077706
4      1.000000

```

```
import numpy as np
```

```
# Definir las variables predictoras (X) y la variable dependiente (y)
X = df.drop(columns='sobrevivencia_dias').values # Todas las columnas excepto 'sobrevivencia_dias'
y = df['sobrevivencia_dias'].values # Columna de sobrevivencia
```

```
# Agregar columna de unos para el intercepto en X
X = np.column_stack((np.ones(X.shape[0]), X))
```

```
# Paso 1: Calcular los coeficientes de regresión β
# β = (X^T * X)^-1 * X^T * y
beta = np.linalg.inv(X.T @ X) @ (X.T @ y)
```

```
# Paso 2: Calcular la Matriz HAT (H)
# H = X * (X^T * X)^-1 * X^T
H = X @ np.linalg.inv(X.T @ X) @ X.T
```

```
# Paso 3: Calcular los valores predichos y^
y_pred = H @ y
```

```
# Paso 4: Extraer los valores de apalancamiento (diagonal de la matriz HAT)
leverage_values = np.diag(H)
```

```
# Imprimir los resultados
print("Coeficientes de regresión (beta):")
print(beta)
```

```
print("\nMatriz HAT:")
print(H)
```

```
print("\nValores de apalancamiento (diagonal de la matriz HAT):")
print(leverage_values)
```

```
print("\nValores predichos (y^):")
print(y_pred)
```

```

Coeficientes de regresión (beta):
[-0.35007574  0.20964887  0.34145811  0.39507079  0.19862703  0.01185889
  0.00605577 -0.01908772  0.09052129]

```

```

Matriz HAT:
[[ 0.03714196  0.01278513  0.03153001 ... -0.0107616  0.03002929
   0.00244981]
 [ 0.01278513  0.06106075  0.04827415 ...  0.02342793  0.0143413
   0.03831749]
 [ 0.03153001  0.04827415  0.09511866 ...  0.02121118  0.02055136
   0.03079561]
 ...
 [-0.0107616  0.02342793  0.02121118 ...  0.10349403  0.00417997
   0.03037262]
 [ 0.03002929  0.0143413  0.02055136 ...  0.00417997  0.06246334
  -0.00802438]
 [ 0.00244981  0.03831749  0.03079561 ...  0.03037262 -0.00802438
   0.05864247]]

```

```

Valores de apalancamiento (diagonal de la matriz HAT):
[0.03714196 0.06106075 0.09511866 0.07234516 0.11942645 0.08065189
 0.07642418 0.07843142 0.05461582 0.07009281 0.07159644 0.04962392
 0.13226894 0.05700003 0.07881398 0.07780303 0.11185145 0.10082189
 0.05958639 0.08650172 0.06043351 0.11175274 0.12660129 0.03285494]

```

```
0.06822654 0.05336978 0.06117105 0.21015924 0.0637799 0.04627201
0.07765548 0.1382091 0.07017845 0.07616134 0.06096139 0.05026388
0.09800896 0.2127648 0.03475134 0.08648634 0.05249874 0.14410901
0.18602117 0.03727933 0.12612543 0.05678346 0.06061771 0.12127331
0.05897746 0.13267214 0.05612883 0.12188549 0.05172967 0.11945507
0.08982727 0.06521368 0.15044357 0.08925879 0.09766654 0.1493418
0.09388399 0.10980073 0.05614709 0.0760336 0.04568778 0.05055969
0.11489641 0.05145828 0.10340551 0.06096153 0.05972407 0.06740464
0.09018131 0.10956971 0.11351692 0.09159203 0.05941746 0.05413813
0.08807509 0.12946688 0.03965555 0.09675421 0.08221149 0.05637406
0.09542024 0.10265085 0.10503717 0.08818435 0.0813272 0.03353834
0.04151066 0.08399356 0.05901144 0.08170849 0.10319491 0.05440942
0.06707476 0.09243314 0.05273941 0.05580399 0.07672453 0.08439488
0.12997958 0.03064927 0.07615033 0.10349403 0.06246334 0.05864247]
```

Valores predichos (y^):

```
[ 0.24294923 0.11555177 0.25496292 0.11287677 0.58907842 0.06336037
 0.17596408 0.20331651 0.29324347 0.26858645 0.37416833 0.02174922
 0.56111684 0.40207758 0.35011873 0.25508064 0.10966687 -0.0567424
 0.31313444 0.35061736 0.17323383 0.08558054 0.14085167 0.294532
 0.31467104 0.22588256 0.19045271 0.66553503 0.10105708 0.19197023
 0.03525361 -0.00789271 0.19133763 0.44523565 0.1531579 0.24137895
 0.20725947 0.17691605 0.17374178 0.19734863 0.11214694 0.17969739
 0.54683579 0.20555463 0.23485034 0.18949916 0.35478549 0.53320106
 0.24560713 0.40905676 0.13029994 0.21676264 0.23557025 0.35941462
 0.06937137 0.29177436 0.2101063 0.08632834 0.20749977 0.13349748
 -0.01215068 0.08733299 0.10280747 0.33352588 0.19407906 0.27349271
 0.01236256 0.09964471 0.48512468 0.2090677 0.11277513 0.13613201
 0.37571055 0.00758058 0.51299009 0.19614777 0.38182996 0.29435553
 0.20152207 0.07036944 0.20394476 0.2298884 0.12691732 0.30611101
 0.01600996 0.39922901 0.2074814 0.15994243 0.33352408 0.2504029
 0.01441111 0.26122992 0.02845716 0.177711 0.01902236 0.3044333
 0.1639206 0.31809252 0.19708786 0.04444076 0.41884099 0.23919779
 0.4219001 0.17831409 0.17915583 0.18657166 0.09477793 0.12891766]
```

Conclusiones

Haz doble clic (o ingresa) para editar