Mục lục

Chương 8

Tương quan và hồi quy

8.1 Hồi quy

Xét bài toán

$$E\left\{\left[Y-\varphi(X)\right]^2\right\} \to \min_{\varphi}.$$

Nghiệm của bài toán ký hiệu là $Y^* = \varphi^*(X)$ và giá trị min gọi là sai số bình phương trung bình.

Định nghĩa 8.1. *Hàm hồi quy của Y theo X, là đại lượng ngẫu nhiên $Z = E\left(Y \mid X\right)$ tức là khi $X = x$ thì $Z = E\left(Y \mid X = x\right)$.*

a) Xét (X, Y) rời rạc. Nếu $X = x_i$ thì $W = \left(Y \mid X = x_i\right)$ có luật phân bố

$$P\left(W = y_j\right) = \frac{p_{ij}}{p_{i*}} \ \forall j.$$

Khi đó

$$Z = EW = \sum_j y_j \frac{p_{ij}}{p_{i*}} = \frac{1}{p_{i*}} \sum y_j p_{ij}.$$

b) Xét (X, Y) liên tục. Nếu $X = x$ thì $W = \left(Y \mid X = x\right)$ có hàm mật độ (công thức 3.15)

$$\psi\left(y \mid x\right) = \frac{f(x, y)}{f_1(x)} \ \forall y.$$

Kỳ vọng của W, tức là kỳ vọng có điều kiện của Y khi biến cố $X = x$ đã xảy ra là

$$\begin{aligned} EW = E\left(Y \mid X = x\right) &= \int_{-\infty}^{\infty} y\psi\left(y \mid x\right) dy \\ &= \int_{-\infty}^{\infty} y\frac{f(x, y)}{f_1(x)} dy = \frac{1}{f_1(x)} \int_{-\infty}^{\infty} yf(x, y)\, dy. \end{aligned} \tag{8.1}$$

Định lý 8.1. $\varphi^*(X) = E\left(Y \mid X\right)$.

Ví dụ 8.1. (B)

Giải.

X	$Z = E\left(Y \mid X\right)$
0	$E\left(Y \mid X = 0\right) = \dfrac{(-1)\,0.08 + 0 \cdot 0.16 + 2 \cdot 0.12 + 3 \cdot 0.04}{0.4} = 0.7$
1	$E\left(Y \mid X = 1\right) = \dfrac{(-1)\,0.05 + 0 \cdot 0.1 + 2 \cdot 0.08 + 3 \cdot 0.02}{0.25} = 0.68$
4	$E\left(Y \mid X = 4\right) = \dfrac{(-1)\,0.07 + 0 \cdot 0.14 + 2 \cdot 0.1 + 3 \cdot 0.04}{0.35} = 0.7143$

□

Ví dụ 8.2. (C)

Giải. Khi $X = x$ thì

$$Z = E\left(Y \mid X = x\right) = \frac{1}{f_1(x)} \int_{-\infty}^{\infty} yf(x, y)\, dy = \begin{cases} \dfrac{10x}{9}, & 0 \le x < 1 \\ 0, & x \notin [0, 1). \end{cases}$$

□

8.2 Hồi quy tuyến tính

8.2.1 Hồi quy tuyến tính lý thuyết

Bài toán

$$E\left\{[Y - aX - b]^2\right\} \to \min_{a,b}$$

có nghiệm

$$\begin{aligned} a &= \rho\sqrt{\frac{DY}{DX}} \\ b &= EY - a\,EX. \end{aligned} \tag{8.2}$$

và sai số $\sigma^2_{Y/X} = \left(1-\rho^2\right) DY$.

Hàm $\varphi(X) = aX + b$ gọi là hàm hồi quy tuyến tính lý thuyết của Y theo X. Đường thẳng $y = ax + b$ gọi là đường hồi quy tuyến tính lý thuyết.

Ví dụ 8.3. (B) $EX = 1.65$, $EY = 0.7$, $DX = 3.1275$, $DY = 1.81$, $\rho = 0.006305$.

Giải.

$$a = 0.006305\sqrt{\frac{1.81}{3.1275}} = 0.04797, b = 0.7 - 0.04797 \cdot 1.65 = 0.6208$$
$$\sigma^2_{Y/X} = \left(1 - 0.006305^2\right) 1.81 = 1.8099.$$

□

Ví dụ 8.4. (C) $EX = \frac{3}{4}, EY = \frac{5}{6}, DX = \frac{3}{80}, DY = \frac{43}{180}, \rho = \frac{5}{\sqrt{129}}$.

Giải.

$$a = \frac{5}{\sqrt{129}}\sqrt{\frac{43/180}{3/80}} = \frac{10}{9}, b = \frac{5}{6} - \frac{10}{9}\cdot\frac{3}{4} = 0$$
$$\sigma^2_{Y/X} = \left[1 - \left(\frac{5}{\sqrt{129}}\right)^2\right]\frac{3}{80} = \frac{13}{430} = 0.03023.$$

□

8.2.2 Hệ số tương quan mẫu

Nhắc lại công thức (3.21) về hệ số tương quan giữa hai đại lượng ngẫu nhiên X và Y

$$\rho = \rho_{XY} = \frac{E(XY) - EX \cdot EY}{\sqrt{DX}\sqrt{DY}}.$$

Giả sử $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)$ là mẫu ngẫu nhiên từ véctơ ngẫu nhiên (X, Y). Hệ số tương quan mẫu giữa X và Y là đại lượng ước lượng cho hệ số tương quan, xác định bởi

$$r = \frac{\overline{XY} - \overline{X}\cdot\overline{Y}}{S_X S_Y} \tag{8.3}$$

trong đó $\overline{XY} = \frac{1}{n}\sum_{i=1}^{n} X_i Y_i$.

Nếu mẫu cụ thể được rút gọn dưới dạng bảng với $\left(x_i, y_j\right)$ lặp lại n_{ij} lần thì $n = \sum_{i,j} n_{ij}$ và

$$\overline{xy} = \frac{1}{n}\sum_{i,j} n_{ij} x_i y_j,\ \overline{x} = \sum_{i} n_{i*} x_i,\ \overline{y} = \sum_{j} n_{*j} y_j.$$

8.2.3 Hồi quy tuyến tính thực nghiệm

Cho mẫu ngẫu nhiên $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)$ từ véctơ ngẫu nhiên (X, Y). Bài toán

$$\frac{1}{n}\sum_{i=1}^{n} (Y_i - aX_i - b)^2 \to \min_{a,b}$$

có nghiệm

$$\begin{aligned} a &= r\frac{S_Y}{S_X} \\ b &= \overline{Y} - a\overline{X}. \end{aligned} \tag{8.4}$$

và sai số $S^2_{Y/X} = \left(1 - r^2\right) S^2_Y$.

Hàm $\widetilde{\varphi}(X) = aX + b$ gọi là hàm hồi quy tuyến tính thực nghiệm của Y theo X, đường thẳng $y = ax + b$ gọi là đường hồi quy tuyến tính thực nghiệm.

Ví dụ 8.5. Cho mẫu từ véctơ ngẫu nhiên (X, Y)

X	1	1.5	3	4.5	5
Y	1.25	1.4	1.5	1.75	2.25

a) Mô tả dữ liệu bởi các điểm trên mặt phẳng.

b) Tính hệ số tương quan mẫu.

c) Tìm hàm hồi quy tuyến tính thực nghiệm của Y theo X. Vẽ đường hồi quy tuyến tính thực nghiệm cùng với các điểm ở ý (a).

Giải. b) $\overline{xy} = \dfrac{1 \cdot 1.25 + 1.5 \cdot 1.4 + 3 \cdot 1.5 + 4.5 \cdot 1.75 + 5 \cdot 2.25}{5} = 5.395.$

$$\overline{x} = 3, \overline{y} = 1.63, s_X^2 = 2.5, s_Y^2 = 0.1226 \Rightarrow r = \frac{5.395 - 3 \cdot 1.63}{\sqrt{2.5 \cdot 0.1226}} = 0.9122.$$

Cách 1:

```
X = [1, 1.5, 3, 4.5, 5]
Y = [1.25, 1.4, 1.5, 1.75, 2.25]
```

```
from scipy.stats import pearsonr
r, _ = pearsonr(X, Y)  # r là thành phần thứ nhất trong hai thành phần của kết quả
```

Cách 2:

```
import numpy as np
np.corrcoef([X, Y])  # ma trận hệ số tương quan mẫu (r_ij)_n, trong đó r_ij = r_{X_i X_j}
r = _[0, 1]
```

Cách 3:

```
import pandas as pd
df = pd.DataFrame({'X': X, 'Y': Y})

r = df['X'].corr(df['Y'])  # hoặc gọi từ ma trận hệ số tương quan bằng lệnh df.corr()['X']['Y'], df.corr().at['X', 'Y'] hoặc df.corr().iat[0, 1]
```

c) $a = 0.9122\sqrt{\dfrac{0.1226}{2.5}} = 0.2020$, $b = 1.63 - 0.202 \cdot 3 = 1.0240$.

Hàm hồi quy tuyến tính thực nghiệm của Y theo X là $\widetilde{\varphi}(X) = 0.2020X + 1.0240$.

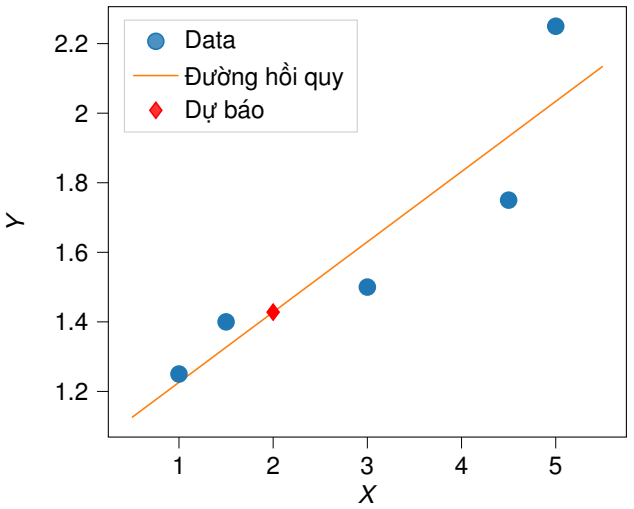
Khi $X = 2$, ta dự báo $Y = 0.2020 \cdot 2 + 1.0240 = 1.428$.

Sai số $s^2_{Y/X} = \left(1 - 0.9122^2\right) 0.1226 = 0.02059$.

```
from scipy.stats import linregress
a, b, r, _, _ = linregress(X, Y)  # kết quả có 5 thành phần

import matplotlib.pyplot as plt

plt.plot(X, Y, 'o', label='Data')
plt.plot([0.5, 5.5], [a*0.5 + b, a*5.5 + b], label='Đường hồi quy')
plt.plot([2], [a*2 + b], 'rD', label='Dự báo')
plt.legend()
plt.xlabel('X')
plt.ylabel('Y')
```



□

8.3 Dữ liệu lớn và học máy

Ví dụ 8.6. Giải bài toán Ví dụ 8.5(c) bằng

d) Mô hình hồi quy trong học máy, với thư viện `scikit-learn`.

e) Mạng thần kinh (học sâu), với thư viện `TensorFlow`

Giải. d) Mô hình dự báo

```
X = [[1], [1.5], [3], [4.5], [5]]
y = [1.25, 1.4, 1.5, 1.75, 2.25]
from sklearn.linear_model import LinearRegression

model = LinearRegression()
model.fit(X, y)

model.coef_        # a
model.intercept_   # b

model.predict([[2]])   # dự báo
```

và đánh giá mô hình

```
y_pred = model.predict(X)

from sklearn.metrics import mean_squared_error, r2_score
```

```
mean_squared_error(y, y_pred)  # s^2_{Y/X}
r2_score(y, y_pred)            # 0 ≤ R^2 ≤ 1, càng gần 1 nghĩa là mô hình càng hiệu quả
```

e)

```
import tensorflow as tf

X = tf.constant([[1], [1.5], [3], [4.5], [5]])
y = tf.constant([1.25, 1.4, 1.5, 1.75, 2.25])

# Bước 1: xây dựng mô hình
model = tf.keras.Sequential([
  tf.keras.layers.InputLayer(input_shape=(1,)),
  tf.keras.layers.Dense(100, activation='relu'),
  tf.keras.layers.Dense(100, activation='relu'),
  tf.keras.layers.Dense(100, activation='relu'),
  tf.keras.layers.Dense(1, activation=None),
])

# Bước 2: chọn phương pháp tối ưu mô hình
model.compile(
  loss=tf.keras.losses.mse,
  optimizer=tf.keras.optimizers.SGD()
)

# Bước 3: tối ưu mô hình với số liệu
model.fit(X, y, epochs=100)  # epochs là số bước lặp

model.predict([2])  # khi X = 2, dự báo Y = 1.3260
```

Sai số của mô hình tại các bước lặp 1, 10, 20,...

Bước	Sai số	Bước	Sai số
1	2.9204	60	0.0522
10	0.1415	70	0.0445
20	0.1121	80	0.0387
30	0.0909	90	0.0341
40	0.0745	100	0.0306
50	0.0619		

□

Khi xây dựng mô hình dự báo với tập dữ liệu lớn về (X, y), gồm $\left(X^1, y_1\right)$, $\left(X^2, y_2\right)$,..., $\left(X^N, y_N\right)$.

1) Chia dữ liệu thành hai phần $X = X_{train} \cup X_{test}$, $y = y_{train} \cup y_{test}$, tương ứng theo các cặp. Thông thường, tỷ lệ dữ liệu học 75% $\leq \frac{|X_{train}|}{|X|} = \frac{|y_{train}|}{|y|} \leq$ 80%.

2) Tối ưu mô hình với số liệu X_{train} và y_{train}.

3) Dự báo các giá trị của y khi X nhận giá trị trong X_{test}, lưu vào dãy y_{pred}.

4) Đánh giá độ hiệu quả của mô hình bằng cách so sánh độ lệch giữa giá trị thực y_{test} và giá trị dự báo y_{pred}.

Ví dụ 8.7. Tiếp tục với số liệu về giới tính, chiều cao, cân nặng trong Ví dụ 5.2.

a) Mô tả dữ liệu về chiều cao – cân nặng bằng biểu đồ điểm.

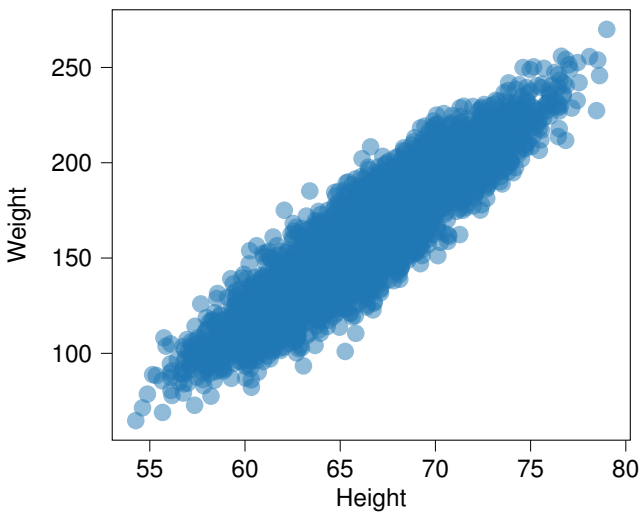
b) Dự đoán chiều cao theo cân nặng, áp dụng mô hình hồi quy tuyến tính của học máy. Mô tả mô hình bằng hình vẽ.

c) Thực hiện yêu cầu của ý (b) theo mô hình học sâu.

Giải. a)

```
import pandas as pd
df = pd.read_csv('https://raw.githubusercontent.com/
    Dataweekends/zero_to_deep_learning_video/master/data/
    weight-height.csv')

df.plot(kind='scatter', x='Height', y='Weight', alpha=0.5)
```



```
X, y = df[['Height']], df['Weight']
# Chia dữ liệu thành hai phần: dữ liệu học và dữ liệu thử
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y)
```

b)

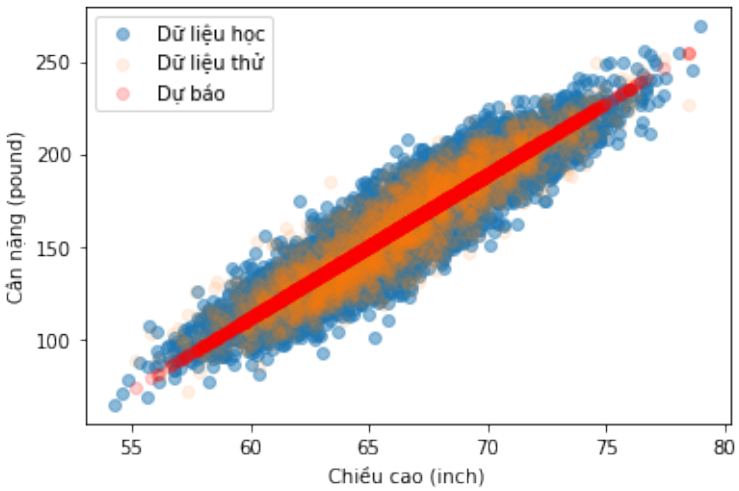
```
from sklearn.linear_model import LinearRegression
model = LinearRegression()
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

from sklearn.metrics import mean_squared_error, r2_score
mean_squared_error(y_test, y_pred)                    # 149.70
r2_score(y_test, y_pred)                              # 0.8515

mean_squared_error(y_train, model.predict(X_train)) # 149.17
r2_score(y_train, model.predict(X_train))             # 0.8563
```

```
import matplotlib.pyplot as plt
plt.plot(X_train, y_train, 'o', alpha=0.5, label='Dữ liệu
    học')
plt.plot(X_test, y_test, 'o', alpha=0.1, label='Dữ liệu thử')
plt.plot(X_test, y_pred, 'or', alpha=0.2, label='Dự báo')
plt.legend()
plt.xlabel('Chiều cao (inch)')
plt.ylabel('Cân nặng (pound)')
```



Hình 8.1: Mô hình hồi quy tuyến tính của cân nặng theo chiều cao

c)

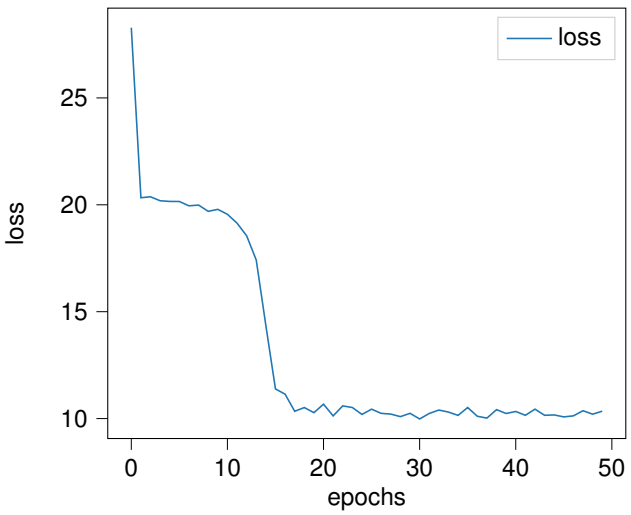
```
import tensorflow as tf

model = tf.keras.Sequential([
  tf.keras.layers.InputLayer(input_shape=(1,)),
  tf.keras.layers.Dense(100, activation='relu'),
  tf.keras.layers.Dense(100, activation='relu'),
  tf.keras.layers.Dense(100, activation='relu'),
  tf.keras.layers.Dense(1, activation=None),
])

model.compile(
  loss=tf.keras.losses.mae,
  optimizer=tf.keras.optimizers.Adam(),
)

history = model.fit(X_train, y_train, epochs=50)

pd.DataFrame(history.history).plot(xlabel='epochs', ylabel='loss')  # biểu đồ sai số sau mỗi bước lặp
```



Sai số của mô hình học sâu cũng xấp xỉ các sai số ở ý (b)

```
mean_squared_error(y_test, y_pred)  # 149.81
r2_score(y_test, y_pred)            # 0.8545
```

và đồng thời, đồ thị mô tả giống hệt Hình 8.1.

□

Một ưu điểm của thuật toán học sâu với thư viện `TensorFlow`, chẳng hạn, bài toán phân loại được cấu hình khá tương tự bài toán hồi quy.

Ví dụ 8.8. Nối tiếp ví dụ Ví dụ 8.7

d) Mô tả số liệu chiều cao – cân nặng theo giới tính trên cùng biểu đồ điểm.

e) Dự đoán những người có chiều cao 65, 66, 67, 68, 69, và 70 inch là nam hay nữ.

f) Cùng có chiều cao 70 inch, dự đoán người có cân nặng 160, 170, và 180 pound là nam hay nữ.

Giải. d)

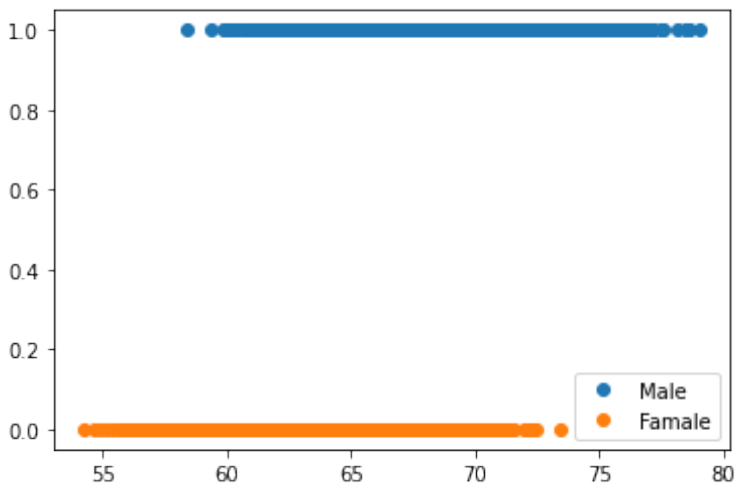
```
X = df[['Height']]

y = [0]*len(X)
for i in range(len(X)):
  if df['Gender'][i] == 'Male':
    y[i] = 1

y = tf.constant(y)

X_Male = df[ df['Gender'] == 'Male' ]['Height']
plt.plot(X_Male, [1]*len(X_Male), 'o', label='Male')

X_Female = df[ df['Gender'] == 'Female' ]['Height']
plt.plot(X_Female, [0]*len(X_Female), 'o', label='Famale')
plt.legend();
```



e)

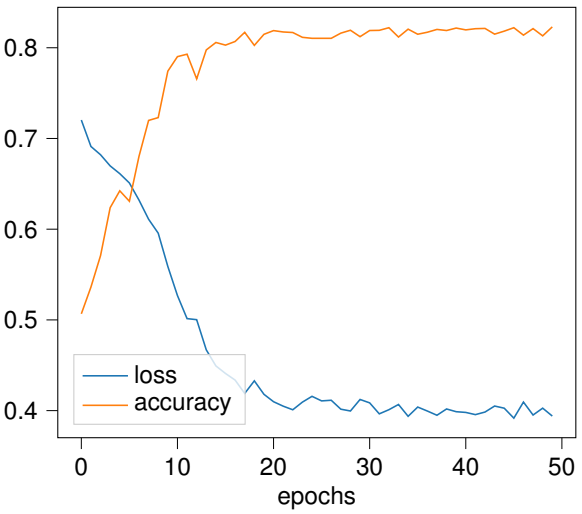
```
model = tf.keras.Sequential([
  tf.keras.layers.Dense(100, activation='relu'),
  tf.keras.layers.Dense(100, activation='relu'),
  tf.keras.layers.Dense(100, activation='relu'),
  tf.keras.layers.Dense(1, activation='sigmoid'),
])

model.compile(
  loss=tf.keras.losses.BinaryCrossentropy(),
  optimizer=tf.keras.optimizers.Adam(),
  metrics=['accuracy']
)

history = model.fit(X, y, epochs=50, verbose=0)

pd.DataFrame(history.history).plot()

model.predict([62, 64, 66, 68, 70])
```



Chiều cao	Xác suất dự đoán	Kết luận
62	0.08283	Nữ
64	0.2277	Nữ
66	0.5038	Nam
68	0.7776	Nam
70	0.9233	Nam

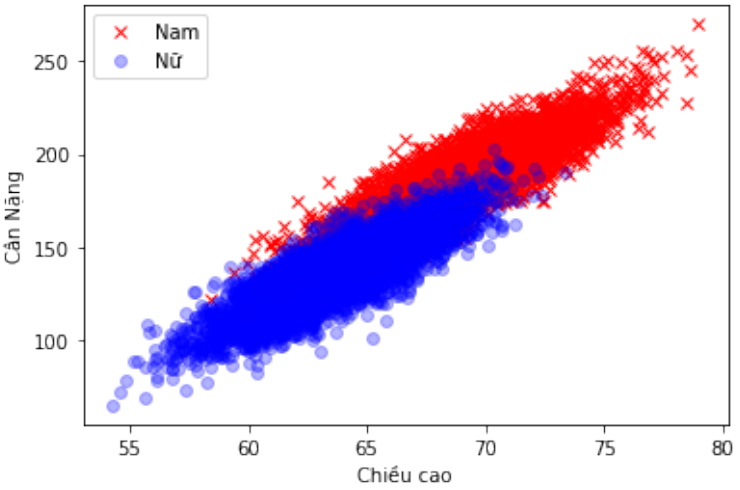
f)

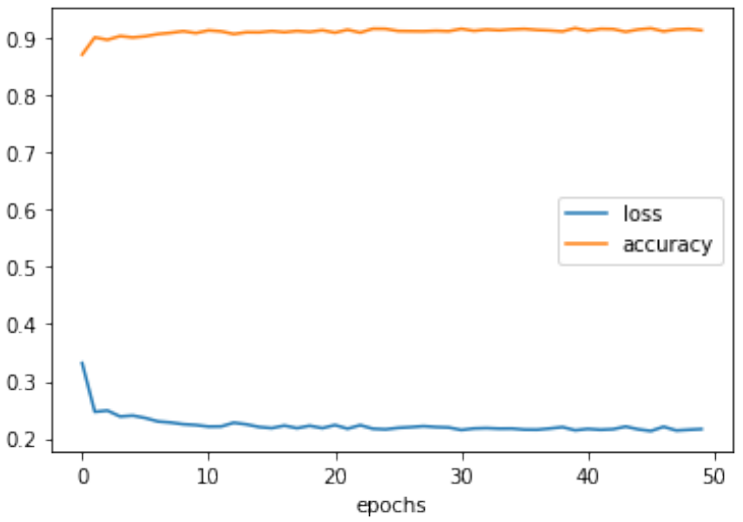
```
X = np.array( pd.DataFrame({'Height': df['Height'], 'Weight'
    : df['Weight']}) )
plt.plot(X[y==1, 0], X[y==1, 1], 'xr', label='Nam')
```

```
plt.plot(X[y==0, 0], X[y==0, 1], 'ob', alpha=0.3, label='Nữ')
plt.legend()
plt.xlabel('Chiều cao')
plt.ylabel('Cân Nặng')

# → Giữ nguyên phần cài đặt, đánh giá mô hình như ý (e)

model.predict([[70, 160], [70, 170], [70, 180]])
```

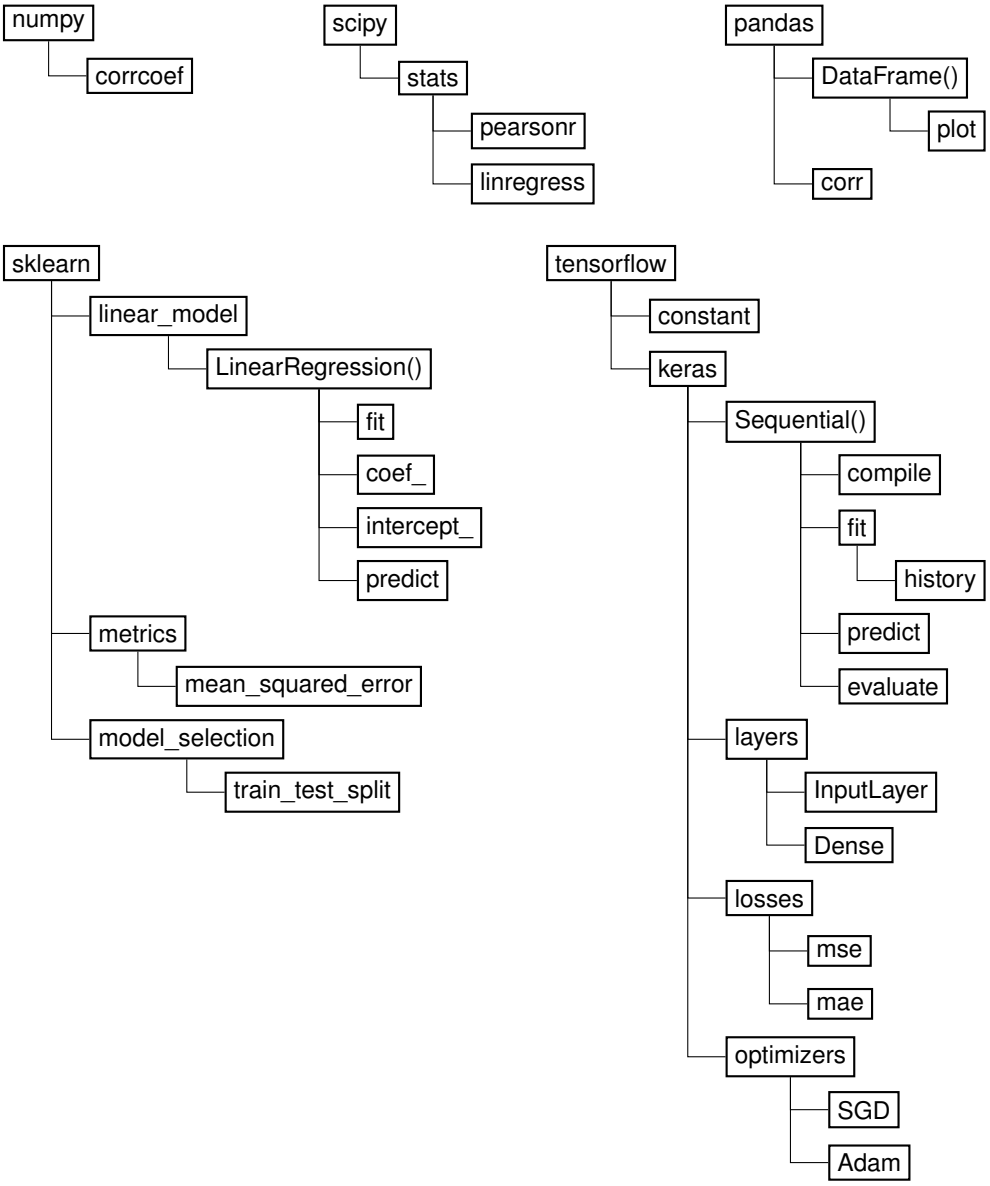




Chiều cao	Cân nặng	Xác suất dự báo	Kết luận
70	160	0.08519	Nữ
70	170	0.4667	Nữ
70	180	0.8859	Nam

□

Tóm tắt về Python



Bài tập Chương 8

Với mẫu từ (X, Y)

a) Tính hệ số tương quan mẫu giữa X và Y.

b) Tìm hồi quy bình phương trung bình tuyến tính thực nghiệm của Y theo X. Ước lượng sai số.

c) Viết phương trình đường thẳng hồi quy bình phương trung bình thực nghiệm của Y

đối với X và vẽ nó.

d) Dự báo giá trị của Y khi X nhận giá trị cho trước.

8.1.

x_i	5	3	4	6	5	5.5	6	7
y_i	2.5	1.5	3	7	6.5	5.5	5	11

Dự báo Y khi $X = 3.5$.

8.2.

Y \ X	7	8	9
200	41	7	
300	1	51	1
400		8	40

Dự báo Y khi $X = 8.5$.

Chương 8

Tương quan và hồi quy

8.1 a) $r = 0.86$

b) $\widetilde{\varphi} = 2.0855X - 5.5684$, $s_Y^2(1 - r^2) = 2.0994$.

c) $y = 2.0855x - 5.5684$

d) Khi $X = 3.5$, ta dự báo $Y = 1.7308$.

8.2 a) $r = 0.9075$

b) $\widetilde{\varphi} = 97.5982X - 480.1350$, $s_Y^2(1 - r^2) = 1129.6972$.

c) $y = 97.5982x - 480.135$.

d) Khi $X = 8.5$, ta dự báo $Y = 349.4498$.