XÁC SUẤT. XÁC SUẤT CÓ ĐIỀU KIỆN. CÔNG THỨC BAYES

Nguyễn Văn Hạnh

AI Academy Vietnam

Tháng 7 năm 2021

Nội dung

Giới thiệu khoá học: Mô tả khoá học

- Cung cấp khái quát các kiến thức cơ bản về xác suất thống kê
- Các nguyên lý của các hiện tượng không chắc chắn
- Ứng dụng các nguyên lý vào một số bài toán thực tế
- Chương trình nhắc lại khái niệm, tóm tắt các kết quả lý thuyết
- Xét các ví dụ minh hoạ và thực hành lập trình trên Python với các bộ dữ liệu cụ thể.
- Kiến thức góp phần vào việc học tập tiếp các khóa học về học máy, trí tuệ nhân tạo, khoa học dữ liệu.

Giới thiệu khoá học: Mô tả khoá học (tiếp)

- Khóa học có hai phần: xác suất, thống kê
- Xác suất và phân bố xác suất
  - khái niệm xác suất, xác suất có điều kiện
  - phân bố của biến ngẫu nhiên và véc tơ ngẫu nhiên; các số đặc trưng của biến ngẫu nhiên;
  - phân bố của mẫu ngẫu nhiên và định lý giới hạn trung tâm.
- Thống kê mô tả và thống kê suy luận
  - Thống kê mô tả
  - Ước lượng tham số
  - Kiểm định giả thuyết
  - Tương quan – hồi quy.

Công cụ sử dụng, thời gian

- Python, Google Colab, Jupyter Notebook
- Numpy, Pandas, Scikit-learn,
- Khóa học được chia thành 10 buổi, mỗi buổi ứng với một project gồm: 1 giờ 30' Lý thuyết + 30' Thực hành.

Đánh giá, tài liệu tham khảo

- Phương pháp đánh giá: thông qua bài tập về nhà, thi giữa kỳ và thi cuối kỳ.
- Tài liệu tham khảo
    1. Norman Matloff, Probability and Statistics for Data Sciences, Taylor & Francis Group (2020)
    2. Peter Bruce, Andrew Bruce & Peter Gedeck, Practical Statistics for Data Scientists, O'reilly Media (2020)
    3. José Unpingco, Python for Probability, Statistics and Machine Learning, Springer (2019)
    4. Paul R. Cohen, Empirical Methods for Artificial Intelligence, The MIT Press (1995)

Các khái niệm xác suất

- Phép thử ngẫu nhiên là một chuỗi các phương thức thực hiện và quan sát một thí nghiệm nào đó cho chúng ta kết quả mà ta không thể dự đoán trước được. ← experiment
- Sự kiện sơ cấp: là kết quả quan sát được đơn giản nhất không thể tách nhỏ hơn của một phép thử. ← outcome
- Không gian mẫu: Là tập hợp tất cả các sự kiện sơ cấp của một phép thử và xung khắc với nhau, ký hiệu là S. ← sample space
- Tập con bất kỳ của không gian mẫu là sự kiện.
  ↳ là 1 tập hợp

VINBIGDATA VINGROUP vd AI Academy Vietnam

Các khái niệm cơ bản về sự kiện

$A = B \Leftrightarrow \begin{cases} A \subset B \\ B \subset A \end{cases}$

tương đg

- Không gian mẫu S là một tập hợp, sự kiện là tập con của S nên các mỗi quan hệ (tập con, tương đương) và các phép toán (hợp, giao, phần bù, trừ) cũng tương tự như lý thuyết tập hợp.
- Tính xung khắc: $A_1, \cdots, A_n$ được gọi là xung khắc nếu $A_i \cap A_j = \emptyset, \forall i \neq j$.
- Tính đầy đủ: $A_1, \cdots, A_n$ được gọi là đầy đủ nếu $A_i \cup \cdots \cup A_n = S$.
- Không gian các sự kiện: $A_1, \cdots, A_n$ được gọi là một không gian các sự kiện nếu nó vừa xung khắc, vừa đầy đủ.

VD: tung xúc xắc: $S = \{0, 1, 2, 3, 4, 5, 6\}$

Kg gian mẫu

$D = \{1, 3, 5\}$ ← sự kiện

VINBIGDATA VINGROUP AI Academy Vietnam

Định nghĩa xác suất

- Xác suất của một pháp thử là một ánh xạ $P(.)$ từ không gian mẫu vào tập số thực thoả mãn:

$A \in S; \; P(A) \longrightarrow ? \in \mathbb{R}[0;1]$

3 tiên đề

1. Với mọi sự kiện A thì $P(A) \geq 0$
2. $P(S) = 1$
3. Cho $A_1, A_2, \cdots$ là xung khắc thì

$$P(A_1 \cup A_2 \cdots) = P(A_1) + P(A_2) \cdots$$

- Từ các tiên đề ta có các tính chất:
  - $P(\emptyset) = 0$
  - A, B xung khắc thì $P(A \cup B) = P(A) + P(B)$
  - A, B bất kỳ $P(A \cup B) = P(A) + P(B) - P(A \cap B)$

CM:

1) $P(\emptyset) = 0$ dựa vào 3 tiên đề

2) g/s $A_1, A_2, \ldots A_n$ là hữu hạn cái sự kiện xung khắc với nhau từng đôi

$$A_i \cap A_j = \emptyset; \forall i \neq j$$

CM:

$$P(A_1 \cup A_2 \cup \ldots \cup A_n) = P(A_1) + P(A_2) + \ldots P(A_n)$$

Sol:

$$A_{n+i} = \emptyset \quad (i \geq 1)$$

$$A_1, A_2, \ldots A_n, A_{n+1}, \ldots$$

$$A_i \cap A_j = \emptyset \quad \forall i \neq j$$

Theo tiên đề 3:

$$P(A_1 \cup A_2 \cup \ldots \cup A_n \cup A_{n+1} \cup \ldots)$$

$$= P(A_1) + P(A_2) + \ldots P(A_n) + \underbrace{P(A_{n+1})}_{0} + \underbrace{\ldots}_{0}$$

$$\Rightarrow P(A_1 \cup A_2 \cup \ldots A_n) = P(A_1) + P(A_2) + \ldots P(A_n)$$

Định nghĩa xác suất cổ điển

- Nếu một phép thử có không gian mẫu là hữu hạn và đồng khả năng $S = \{w_1, \cdots, w_n\}$
- Đồng khả năng nên $P(\{w_1\}) = \cdots = P(\{w_n\})$
- Do $1 = P(S) = P(\{w_1\}) + \cdots + P(\{w_n\}) = nP(\{w_1\})$ nên $P(\{w_i\}) = \frac{1}{n}, \forall i = \overline{1, n}$.
- A là một sự kiện thì $P(A) = \frac{\#A}{\#S}$
- Ví dụ tung một đồng xu 10000 lần, xác suất để tung được mặt sấp bằng bao nhiêu?

Ví dụ 1: Gieo một đồng xu cân đối đồng chất 10000 lần

```
In [38]: import numpy as np
         so_lan_tung=10000
         tung_dong_xu = np.random.randint(2, size=so_la
         so_lan_0 = (tung_dong_xu == 0).sum()
         so_lan_1 = (tung_dong_xu == 1).sum()
         P_0=so_lan_0/so_lan_tung
         P_1=so_lan_1/so_lan_tung
         print(P_0)
         print(P_1)

0.496
0.504
```

Ví dụ 2: Gieo một đồng xu bất cân đối 10000 lần

```
In [41]: def tung_xu():
          if np.random.random()<0.6:
            return 0
          else:
            return 1
         ket_qua=np.zeros(so_lan_tung)
         for i in range(so_lan_tung):
             ket_qua[i]=(tung_xu())

         P_3=(ket_qua==0).sum()/so_lan_tung
         P_4=(ket_qua==1).sum()/so_lan_tung
         print(P_3)
         print(P_4)

0.5986
0.4014
```

Xác suất có điều kiện

Xác suất có điều kiện

Một phép thử nếu biết sự kiện $B, P(B) \neq 0$ đã xảy ra thì xác suất sự kiện A xảy ra là xác suất có điều kiện ký hiệu $P(A|B)$ được xác định bởi công thức

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

- Công thức nhân

$$P(A \cap B) = P(B).P(A|B) = P(A).P(B|A)$$

- Hai sự kiện được gọi là độc lập nếu và chỉ nếu

$$P(A \cap B) = P(A).P(B)$$

Công thức Bayes

- Cho hai sự kiện A, B và $P(A), P(B)$ là hai xác suất được quan sát độc lập với nhau.
- $P(A)$ được gọi là xác suất tiên nghiệm (Prior)
- $P(B)$ gọi là xác suất bằng chứng (Evidence)
- $P(B) = P(B|A) \times P(A) + P(B|\overline{A}) * P(\overline{A})$
- $P(A|B)$ được gọi là xác suất hậu nghiệm (Posterior)
- $P(B|A)$ được gọi là xác suất có thể đúng (Likelihood)
- Công thức Bayes $P(A|B) = \frac{P(A).P(B|A)}{P(B)}$
- Posterior = Likelihood * Prior / Evidence

Công thức Bayes hai sự kiện A, B quan sát độc lập

$$\underbrace{P(A|B)}_{\text{Posterior}} = \frac{\overbrace{P(B|A)}^{\text{Likelihood}}.\overbrace{P(A)}^{\text{Prior}}}{\underbrace{P(B)}_{\text{Evidence}}}$$

Ví dụ công thức Bayes

- Một bệnh viện phải làm xét nghiệm với một số lượng lớn bệnh nhân và thấy rằng có 0.1% bị mắc bệnh còn 99.9% là khoẻ. Để biết rằng việc xét nghiệm là đúng người ta tiến hành người khoẻ xét nghiệm âm tính 99%. Nếu xét nghiệm trên một người bị bệnh thì xác suất dương tính là 98%. Chọn ngẫu nhiên một người và thấy rằng người này dương tính, tìm xác suất người này là khoẻ?
- +(-) là sự kiện người này dương (âm tính). $A(\overline{A})$ là người này khoẻ (mắc bệnh).
- ta có $P(A) = 0.999; P(\overline{A}) = 0.001; P(+|A) = 0.01; P(-|A) = 0.99; P(+|\overline{A}) = 0.98; P(-|\overline{A}) = 0.02$
- Xác suất cần tìm $P(\overline{A}|+) = \frac{P(+|\overline{A}).P(\overline{A})}{P(+)} = \frac{0.98 \times 0.001}{0.98 \times 0.001 + 0.01 \times 0.999} = 0.0893$

Hàm tính công thức Bayes

```
def bayes_theorem(p_a, p_b_given_a, p_b_given_not_a):
    # Tinh P(not A)
    p_not_a = 1 - p_a
    # Tinh P(B) bang cong thuc xac suat toan phan
    p_b = p_b_given_a * p_a + p_b_given_not_a * p_not_a
    # Tinh P(A|B) bang cong thuc Bayes
    p_a_given_b = (p_b_given_a * p_a) / p_b
    return p_a_given_b

# P(A)
p_a = 0.999
# P(B|A)
p_b_given_a = 0.01
# P(B|not A)
p_b_given_not_a = 0.98
# calculate P(A|B)
result = bayes_theorem(p_a, p_b_given_a, p_b_given_not_a)
# summarize
print('P(A|B) = %.4f%%' % (result * 100))
print('P(not A|B) = %.4f%%' % ((1-result) * 100))
```

```
P(A|B) = 91.0665%
P(not A|B) = 8.9335%
```

Công thức Bayes tổng quát

- Cho không gian các sự kiện $A_1, \cdots, A_n$,
- B là một sự kiện nào đó thì ta có công thức

Xác suất toàn phần

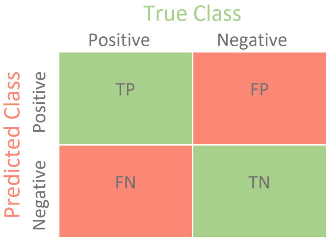
$$P(B) = \sum_{i=1}^{n} P(A_i).P(B|A_i)$$

Công thức Bayes

$$P(A_i|B) = \frac{P(A_i \cap B)}{P(B)} = \frac{P(A_i \cap B)}{\sum_{j=1}^{n} P(A_j).P(B|A_j)}$$

Ứng dụng công thức Bayes vào phân lớp nhị phân

- Xét ví dụ về xét nghiệm mắc bệnh theo các thuật ngữ phổ biến trong phân lớp nhị phân, chúng ta có khái niệm đặc hiệu và độ nhạy (specificity and sensitivity).
- Xét ma trận



  - True Positive Rate (TPR) = TP / (TP + FN)
  - False Positive Rate (FPR) = FP / (FP + TN)
  - True Negative Rate (TNR) = TN / (TN + FP)
  - False Negative Rate (FNR) = FN / (FN + TP)

Ứng dụng công thức Bayes vào phân lớp nhị phân

- Tương ứng với lý thuyết Bayes:
  - P(B|A): True Positive Rate (TPR).
  - P(not B|not A): True Negative Rate (TNR).
  - P(B|not A): False Positive Rate (FPR).
  - P(not B|A): False Negative Rate (FNR).
- Từ đây chúng ta có các xác suất
  - $P(A)$: Xác suất lớp Positive
  - $P(notA)$: Xác suất lớp Negative
  - $P(B)$: Xác suất dự báo Positive
  - $P(notB)$: Xác suất dự báo Negative

Đánh giá độ nhạy và đặc hiệu, độ chính xác phân lớp

- $sensitivity = \frac{TP}{TP+FN}$
- $Specificity = \frac{TN}{TN+FP}$

Trong phân lớp đánh giá mô hình các tiêu chí độ chính xác được đưa ra

- $Precision = \frac{TP}{TP+FP}$
- $Recall = \frac{TP}{TP+FN}$
- $Accuracy = \frac{TP+FN}{TP+TN+FP+FN}$

Giới thiệu về Google Colab hay Jupyter Notebook

- https://colab.research.google.com/
- Cài Python: https://www.python.org/downloads/
- Cài jupyter: https://jupyter.org/install

Ma trận confusion

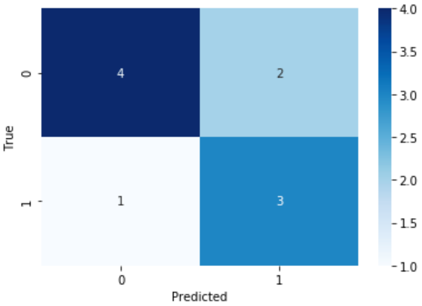
```
import numpy as np
from sklearn.metrics import precision_score, \
    recall_score, confusion_matrix, accuracy_score
y_true=np.array([1,0,1,0,1,1,0,0,0,0])
y_pred=np.array([1,1,1,0,0,1,0,0,0,1])
print ('Accuracy:', accuracy_score(y_true, y_pred))
print ('Recall:', recall_score(y_true, y_pred))
print ('Precision:', precision_score(y_true, y_pred))
print ('\n confussion matrix:\n',confusion_matrix(y_true, y_pred))
```

```
Accuracy: 0.7
Recall: 0.75
Precision: 0.6

 confussion matrix:
 [[4 2]
 [1 3]]
```

Vẽ Ma trận confusion

```
import matplotlib.pyplot as plt
import seaborn as sns

sns.heatmap(confusion_matrix(y_true, y_pred), annot=True, fmt='',cmap='Blues')
plt.xlabel('Predicted')
plt.ylabel('True')
```

```
Text(33,0.5,'True')
```



Bài tập thực hành 1

- Mô phỏng tung một con xúc sắc cân đối đồng chất 5000 lần. Dựa vào các giá trị mô phỏng tìm các xác suất ở câu dưới.
- Tìm xác suất để số chấm xuất hiện là 4.
- Tìm xác suất số chấm xuất hiện lớn hơn hoặc bằng 4.
- Giả sử biết rằng số chấm xuất hiện lớn hơn hoặc bằng 4. Tìm xác suất để mặt 6 chấm xuất hiện.

Bài tập thực hành 2

Iris Versicolor   Iris Setosa   Iris Virginica

- Bài toán: Hãy xây một mô hình phân loại loài hoa diên vỹ (Iris)
- Giải quyết bài toán gồm 4 bước: 1. Thu thập dữ liệu; 2. Xây dựng mô hình; 3. Huấn luyện mô hình; 4. Đánh giá mô hình và triển khai
- Mô tả hoa diên vỹ là một họ hoa có nhiêu loài như setona, versicolor và virginica
- Dưới đây là chương trình thực hiện bước 1, 2, 3. Bạn hãy tìm cách đánh giá mô hình dựa vào Accuracy, Precision, Recall?

Bài tập thực hành 2: Phân lớp hoa diên vỹ

```
from sklearn import datasets# Gọi thư viện sklearn
iris=datasets.load_iris()#Gán dữ liệu iris và biến iris
x=iris.data# x là dữ liệu về các thuộc tính của hoa
y=iris.target# y là dữ liệu loại
```

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test=train_test_split(x, y, test_size=0.3)
```

```
from sklearn import tree
classifier=tree.DecisionTreeClassifier()
classifier.fit(x_train,y_train)
y_pred=classifier.predict(x_test)
```

```
y_pred
```

```
array([1, 2, 1, 0, 0, 1, 0, 1, 1, 1, 1, 0, 2, 1, 2, 2, 2, 1, 2, 2, 2, 1,
       2, 2, 0, 2, 1, 1, 2, 1, 2, 2, 0, 1, 1, 0, 1, 0, 0, 1, 2, 2, 1, 2,
       0])
```