Chương 4: Thống kê mô tả

Contents

Thống kê mô tả

Thống kê mô tả là bước đầu tiên trong phân tích dữ liệu, tập trung vào việc tóm tắt và mô tả các đặc điểm cơ bản của dữ liệu. Việc sử dụng thống kê mô tả giúp hiểu rõ hơn về cấu trúc và xu hướng của dữ liệu trước khi tiến hành các phân tích sâu hơn.

4.1 Thống kê mô tả là gì?

Thống kê mô tả bao gồm:

- **Các chỉ số trung tâm** (Central Tendency): Trung bình, trung vị, mode.
- **Các chỉ số phân tán** (Dispersion): Độ lệch chuẩn, phương sai, khoảng giá trị.
- **Các chỉ số hình dạng phân phối**: Độ lệch (Skewness) và độ nhọn (Kurtosis).

4.2 Các chỉ số trung tâm

4.2.1 Mean (Trung bình): Là giá trị trung bình cộng của tất cả các giá trị trong tập dữ liệu.

- Cách tính:

```
import numpy as np

data = [1, 2, 3, 4, 5]
mean = np.mean(data)
print("Mean:", mean)
```

Giải thích:

- np.mean(data): Tính giá trị trung bình của tập dữ liệu. Trong ví dụ này, giá trị trung bình là (1 + 2 + 3 + 4 + 5) / 5 = 3.

4.2.2 Median (Trung vị):

Là giá trị nằm ở giữa tập dữ liệu khi sắp xếp theo thứ tự tăng dần.

- Cách tính:

```
median = np.median(data)
print("Median:", median)
```

Giải thích:

- np.median(data): Trả về giá trị ở giữa danh sách sau khi sắp xếp. Trong ví dụ trên, giá trị trung vị là 3.

4.2.3 Mode (Mode - Giá trị xuất hiện nhiều nhất):

- Cách tính:

```
from scipy import stats

mode = stats.mode(data)
print("Mode:", mode)
```

Giải thích:

- stats.mode(data): Tính giá trị xuất hiện nhiều nhất. Nếu tất cả các giá trị có số lần xuất hiện bằng nhau, mode sẽ trả về giá trị đầu tiên.

4.3 Các chỉ số phân tán

4.3.1 Range (Khoảng giá trị):

Hiệu số giữa giá trị lớn nhất và nhỏ nhất.

- Cách tính:

```
range_value = np.ptp(data)
print("Range:", range_value)
```

Giải thích:

- np.ptp(data): Trả về giá trị chênh lệch giữa giá trị lớn nhất và nhỏ nhất trong dữ liệu. Ví dụ: (5 - 1) = 4.

4.3.2 Variance (Phương sai):

Đo lường mức độ phân tán của dữ liệu.

- Cách tính:

```
variance = np.var(data)
print("Variance:", variance)
```

Giải thích:

- np.var(data): Tính phương sai của tập dữ liệu, đo lường sự biến thiên của dữ liệu so với giá trị trung bình.

4.3.3 Standard Deviation (Độ lệch chuẩn):

Là căn bậc hai của phương sai, đo lường độ phân tán quanh giá trị trung bình.

- Cách tính:

```
std_dev = np.std(data)
print("Standard Deviation:", std_dev)
```

Giải thích:

- np.std(data): Tính độ lệch chuẩn của tập dữ liệu, cho biết mức độ phân tán

4.4 Hình dạng phân phối

4.4.1 Skewness (Độ lệch):

Cho biết sự không đối xứng của dữ liệu. Skewness > 0: lệch phải, Skewness < 0: lệch trái.

- Cách tính:

```
skewness = stats.skew(data)
print("Skewness:", skewness)
```

Giải thích:

- stats.skew(data): Tính độ lệch phân phối của dữ liệu. Nếu giá trị skewness > 0, dữ liệu lệch phải, nếu skewness < 0, dữ liệu lệch trái.

4.4.2 Kurtosis (Độ nhọn):

Đo lường "độ nhọn" của phân phối.

- Cách tính:

```
kurtosis = stats.kurtosis(data)
print("Kurtosis:", kurtosis)
```

Giải thích:

- stats.kurtosis(data): Tính độ nhọn của phân phối. Giá trị kurtosis cao cho thấy phân phối có đỉnh nhọn hơn bình thường.

4.5 Tóm tắt dữ liệu bằng Pandas

Pandas cung cấp một phương thức describe() để nhanh chóng tính toán các chỉ số thống kê mô tả cho tất cả các cột số trong DataFrame.

```
import pandas as pd

# Tạo DataFrame
data = {'Age': [25, 30, 35, 40, 45], 'Salary': [50000, 60000, 75000, 80000, 90000]}
df = pd.DataFrame(data)

# Tóm tắt dữ liệu
print(df.describe())
```

Giải thích:

- df.describe(): Tính toán các chỉ số thống kê mô tả như mean, std, min, max và các percentiles (25%, 50%, 75%) cho tất cả các cột số trong DataFrame.

4.6 Trực quan hóa thống kê mô tả

Trực quan hóa các chỉ số thống kê mô tả giúp nhận diện xu hướng và sự phân bố dữ liệu.

4.6.1 Histogram (Biểu đồ tần suất):

Biểu đồ tần suất cho biết phân phối của một biến.

```
import matplotlib.pyplot as plt

plt.hist(data, bins=5, color='blue', edgecolor='black')
plt.title("Histogram")
plt.xlabel("Values")
plt.ylabel("Frequency")
plt.show()
```

Giải thích:

- plt.hist(data): Vẽ biểu đồ tần suất cho dữ liệu, chia thành 5 bins (ngăn). Trục x là giá trị, và trục y là tần suất xuất hiện của giá trị đó.

4.6.2 Boxplot (Biểu đồ hộp):

Boxplot giúp bạn nhận diện các giá trị ngoại lai (outliers) và phân bố dữ liệu.

```
plt.boxplot(data)
plt.title("Boxplot")
plt.show()
```

Giải thích:

- plt.boxplot(data): Vẽ biểu đồ hộp, giúp trực quan hóa các chỉ số thống kê như trung vị, phần tư (quartiles) và ngoại lai (outliers).

4.6.3 Violin Plot: Violin Plot kết hợp cả Histogram và Boxplot, giúp bạn thấy rõ phân phối và các đặc điểm thống kê của dữ liệu.

```
import seaborn as sns

sns.violinplot(data=data)
plt.title("Violin Plot")
plt.show()
```

Giải thích:

- sns.violinplot(data): Vẽ Violin Plot, cho bạn thấy cả phân phối và các phần tử cơ bản của dữ liệu.

4.7 Thực hành

- **Bài tập 1:** Tạo một tập dữ liệu gồm 10 số ngẫu nhiên. Tính toán và hiển thị: Mean, Median, Mode, Variance, Standard Deviation, Skewness, và Kurtosis.
- **Bài tập 2:** Sử dụng dữ liệu về chiều cao và cân nặng của 50 người, tạo biểu đồ Histogram và Boxplot để phân tích sự phân bố.

4.8 Thực tiễn ứng dụng của thống kê mô tả

Thống kê mô tả có nhiều ứng dụng trong thực tế, bao gồm:

- **Phân tích thị trường**: Sử dụng để hiểu rõ hơn về hành vi của người tiêu dùng.
- **Y tế**: Giúp phân tích dữ liệu bệnh nhân và xu hướng sức khỏe.
- **Giáo dục**: Đánh giá kết quả học tập và hiệu suất của học sinh.

4.9 Những sai lầm thường gặp

Khi sử dụng thống kê mô tả, có một số sai lầm phổ biến mà người dùng cần lưu ý:

- **Chỉ dựa vào trung bình**: Trung bình có thể bị ảnh hưởng bởi các giá trị ngoại lai.
- **Không xem xét độ phân tán**: Chỉ nhìn vào các chỉ số trung tâm mà không xem xét độ phân tán có thể dẫn đến hiểu lầm về dữ liệu.

4.10 So sánh với thống kê suy diễn

Thống kê mô tả chỉ tóm tắt và mô tả dữ liệu hiện có, trong khi thống kê suy diễn sử dụng mẫu dữ liệu để đưa ra kết luận về toàn bộ quần thể.

4.11 Công cụ và thư viện

Một số công cụ và thư viện phổ biến để thực hiện thống kê mô tả trong Python bao gồm:

- **NumPy**: Thư viện cơ bản cho các phép toán số học.
- **Pandas**: Cung cấp các phương thức mạnh mẽ để phân tích và tóm tắt dữ liệu.

- **SciPy**: Hỗ trợ các phép toán thống kê nâng cao.

4.12 Nghiên cứu điển hình

- **Nghiên cứu về sức khỏe**: Phân tích dữ liệu từ một nghiên cứu y tế để xác định các yếu tố ảnh hưởng đến sức khỏe cộng đồng.
- **Phân tích dữ liệu bán hàng**: Sử dụng thống kê mô tả để hiểu rõ hơn về doanh thu và hành vi mua sắm của khách hàng.

4.13 Tổng kết chương 4

Nội dung: Trong chương 4 này đã giúp bạn nắm vững các chỉ số thống kê mô tả và cách áp dụng chúng vào dữ liệu thực tế. Trong chương 5 chúng ta sẽ bắt đầu với trực quan hóa dữ liệu bằng Matplotlib.