Statistics: Variance

Hoàng-Nguyên Vũ

1. Lý thuyết về Variance (Phương sai)

1.1 Định nghĩa

Variance (phương sai) là đại lượng thể hiện mức độ phân tán của một biến ngẫu nhiên quanh giá trị kỳ vọng của nó.

1.2 Công thức

Biến ngẫu nhiên rời rạc:

$$\text{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])^2] = \sum_i (x_i - \mu)^2 \cdot P(X = x_i)$$

Phương sai mẫu:

$$s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2$$

Phương sai tổng thể:

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2$$

Trong đó:

- $\mu = \mathbb{E}[X]$: kỳ vọng của biến ngẫu nhiên
- $\bar{x}$: trung bình mẫu
- n: số lượng phần tử trong mẫu

1.3 Ứng dụng trong AI

- Variance giúp đánh giá sự ổn định của mô hình.
- Trong học máy, tồn tại khái niệm **bias-variance tradeoff** để cân bằng độ phức tạp mô hình.
- Dùng để chuẩn hóa dữ liệu (standardization) khi huấn luyện.

1.4 Tính toán bằng NumPy

```python
import numpy as np

data = np.array([1, 2, 3, 4, 5])

population_var = np.var(data)
sample_var = np.var(data, ddof=1)

print("Population Variance:", population_var)
print("Sample Variance:", sample_var)
```

2. Bài tập thực hành Variance với NumPy

Bài 1: Tính phương sai rời rạc

Cho biến ngẫu nhiên $X = [1, 3, 5]$ với xác suất tương ứng $P = [0.2, 0.5, 0.3]$.

Yêu cầu: Tính kỳ vọng $\mathbb{E}[X]$ và phương sai $\text{Var}(X)$ bằng công thức rời rạc.

Bài 2: Phân tích phương sai mẫu dữ liệu thực

Sinh ra dữ liệu gồm 1000 số thực từ phân phối chuẩn $\mathcal{N}(0, 2^2)$.

Yêu cầu: Tính phương sai mẫu và phương sai tổng thể. So sánh với giá trị lý thuyết.

Bài 3: So sánh độ ổn định giữa hai mô hình AI

```python
model_a_scores = np.array([0.8, 0.7, 0.9, 0.75, 0.85])
model_b_scores = np.array([0.6, 0.4, 0.9, 0.3, 0.8])
```

Yêu cầu: Tính phương sai và xác định mô hình nào ổn định hơn.

Bài 4: Phân tích phân tán ảnh đầu vào

Giả sử bạn có trung bình pixel của 5 ảnh:

```python
pixel_means = np.array([122, 120, 119, 123, 121])
```

Yêu cầu: Tính phương sai để đánh giá mức độ phân tán dữ liệu.

Bài 5: Variance trong Reinforcement Learning

Một tác nhân nhận phần thưởng trong 10 lần thử nghiệm:

```python
rewards = np.array([10, 9, 8, 10, 7, 6, 9, 10, 5, 8])
```

Yêu cầu: Tính phương sai phần thưởng để đánh giá độ ổn định của chiến lược.