Statistics: Random Variable

Hoàng-Nguyên Vũ

1 Biến Ngẫu Nhiên (Random Variable)

Một **biến ngẫu nhiên** là một hàm số ánh xạ mỗi kết quả của một thí nghiệm ngẫu nhiên thành một giá trị số thực:

$$X : \Omega \to \mathbb{R}$$

Trong AI, biến ngẫu nhiên được sử dụng để mô hình hóa các giá trị không chắc chắn như:

- Kết quả dự đoán (labels)
- Dữ liệu đầu vào hoặc nhiễu (noise)
- Biến tiềm ẩn trong mô hình sinh (latent variable)

2 Biến Ngẫu Nhiên Rời Rạc (Discrete Random Variable)

Định nghĩa

Là biến chỉ nhận các giá trị rời rạc (đếm được). Xác suất của từng giá trị được biểu diễn qua hàm phân phối xác suất (PMF):

$$P(X = x_i) = p(x_i), \quad \sum_i p(x_i) = 1$$

Ứng dụng trong AI

- Classification: nhãn lớp (0, 1, 2, ...)
- NLP: từ, token, nhãn sentiment
- Reinforcement Learning: action space

Mã Python ví dụ

```python
import numpy as np
import matplotlib.pyplot as plt

labels = np.random.choice(['cat', 'dog', 'rabbit'], size=1000, p=[0.4, 0.4, 0.2])
unique, counts = np.unique(labels, return_counts=True)
```

```
plt.bar(unique, counts / len(labels))
plt.title("Discrete Random Variable - Label Distribution")
plt.ylabel("Proportion")
plt.grid(True)
plt.show()
```

3 Biến Ngẫu Nhiên Liên Tục (Continuous Random Variable)

Định nghĩa

Là biến nhận giá trị từ một khoảng liên tục. Dùng hàm mật độ xác suất (PDF):

$$\int_{-\infty}^{\infty} f(x)\,dx = 1$$

$$P(a \leq X \leq b) = \int_{a}^{b} f(x)\,dx$$

Ứng dụng trong AI

- Regression (hồi quy): dự đoán giá, thời gian, nhiệt độ
- Noise injection trong GANs
- Latent variable trong VAE, Diffusion

Mã Python ví dụ

```
from scipy.stats import norm
import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(-4, 4, 1000)
pdf = norm.pdf(x, loc=0, scale=1)

plt.plot(x, pdf, label="Normal PDF")
plt.fill_between(x, pdf, where=(x > -1) & (x < 1), alpha=0.4, label="
    Region [-1,1]")
plt.title("Continuous Random Variable - Normal Distribution")
plt.legend()
plt.grid(True)
plt.show()
```

4 Mô Hình Naive Bayes và Biến Ngẫu Nhiên

Naive Bayes giả định mỗi feature là một biến ngẫu nhiên độc lập:

$$P(y|X) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

Mã Python ví dụ

```python
from sklearn.naive_bayes import GaussianNB
from sklearn.datasets import make_classification
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

X, y = make_classification(n_samples=1000, n_features=10, n_classes=3,
    random_state=42)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)

model = GaussianNB()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

print("Accuracy:", accuracy_score(y_test, y_pred))
```

5 Biến Ngẫu Nhiên Tiềm Ẩn trong VAE

Trong Variational Autoencoder, latent variable $z \sim \mathcal{N}(\mu, \sigma^2)$. Ta dùng:

$$z = \mu + \sigma \cdot \epsilon, \quad \epsilon \sim \mathcal{N}(0, 1)$$

Mã Python ví dụ

```python
mu = np.zeros(10)
sigma = np.ones(10)
eps = np.random.normal(size=10)
z = mu + sigma * eps
print("Sampled latent vector z:", z)
```

Bài tập:

Bài 1: Sinh Nhãn Phân Loại Ngẫu Nhiên

Mô tả: Sinh 1000 nhãn thuộc 3 lớp `cat`, `dog`, `rabbit` với xác suất tương ứng là 0.4, 0.4, 0.2. In 10 nhãn đầu và tần suất xuất hiện.

Bài 2: Sinh Noise Gaussian

Mô tả: Sinh 100 vector ngẫu nhiên (shape: `(100,10)`) từ phân phối chuẩn $\mathcal{N}(0, 1)$. Tính giá trị trung bình và độ lệch chuẩn của toàn mảng.

Bài 3: Ước Lượng Xác Suất Từ Phân Phối Chuẩn

Mô tả: Mô phỏng 10.000 điểm từ phân phối chuẩn $\mathcal{N}(0, 1)$ và ước lượng xác suất $P(-1 < X < 1)$.

Bài 4: Mô Phỏng Phân Phối PMF

Mô tả: Sinh 1000 nhãn từ các lớp `positive`, `neutral`, `negative` với xác suất tương ứng [0.3, 0.5, 0.2]. Tính PMF thực tế từ dữ liệu sinh ra.

Bài 5: Sampling Biến Tiềm Ẩn trong VAE

Mô tả: Áp dụng công thức *reparameterization trick* trong VAE:

$$z = \mu + \sigma \cdot \epsilon, \quad \epsilon \sim \mathcal{N}(0, 1)$$