

به نام خدا



دانشگاه تهران

دانشکده علوم و فناوری های میان رشته ای

Machine Learning

تمرین اول

سوال اول

$$a) f(x_k; \theta) = \theta \exp(-\theta x_k) \quad x_k \geq 0 \quad \theta > 0$$

$$L(\theta) = \prod_{k=1}^N f(x_k; \theta) = \prod_{k=1}^N \theta \exp(-\theta x_k) = \theta^N \exp\left(-\theta \sum_{k=1}^N x_k\right)$$

$$\Rightarrow \ln L(\theta) = N \ln \theta - \theta \sum_{k=1}^N x_k$$

$$\Rightarrow \frac{d}{d\theta} \ln L(\theta) = N \frac{1}{\theta} - \sum_{k=1}^N x_k = 0 \rightarrow \theta = \frac{N}{\sum_{k=1}^N x_k}$$

Beta Density

$$\Rightarrow L(\theta) = \prod_{k=1}^N f(x_k; \theta) = \prod_{k=1}^N \sqrt{\theta} x_k^{\theta-1} = \theta^{N/2} \prod_{k=1}^N x_k^{\theta-1}$$

$$\Rightarrow \ln L(\theta) = \frac{N}{2} \ln \theta + (\theta - 1) \sum_{k=1}^N \ln x_k$$

$$\Rightarrow \frac{d}{d\theta} \ln L(\theta) = \frac{N}{2} \frac{1}{\theta} + \sum_{k=1}^N \ln x_k = 0 \rightarrow \frac{N}{2\theta} = - \sum_{k=1}^N \ln x_k$$

$$\rightarrow \frac{N^2}{\theta^2} = \frac{\left(\sum_{k=1}^N \ln x_k\right)^2}{\theta} \rightarrow \theta = \frac{N^2}{\left(\sum_{k=1}^N \ln x_k\right)^2}$$

سوال دوم

الف)

1.

a. واریانس: واریانس گسترش ی ا پراکندگی مجموعه ای از مقادیر را حول میان گین آنها اندازه می گیرد.

b. کوواریانس: کوواریانس میزان تغییر دو متغیر را با هم را نشان می دهد. کوواریانس مثبت به این معنی است که با افزایش یک متغیر، متغیر دیگری ز تمایل به افزایش دارد و بالعکس.

c. میانگین: مقداری است که به عنوان نماینده های از یک دسته از اعداد استفاده م ی شود و نشان دهنده مقدار متوسط یا متعادل در یک سری از اعداد است.

d. Correlation: همبستگی ی ک معیار نرمال شده از کوواریانس است که قدرت و جهت رابطه خطی بین دو متغیر را نشان می دهد.

2. با محاسبه مقداری B^0 و B^1 ، می توانیم مدل رگرسیون خطی را با به حداقل رساندن مجموع مربعات باقیمانده (RSS) استخراج کنیم.

$$RSS = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 = \sum_{i=1}^n (Y_i - (B_0 + B_1 X_i))^2$$

$$\left. \begin{array}{l} \frac{\partial RSS}{\partial B_0} = -2 \sum (Y_i - (B_0 + B_1 X_i)) = 0 \rightarrow Y_i = B_0 + B_1 X_i \\ \frac{\partial RSS}{\partial B_1} = -2 \sum (Y_i - (B_0 + B_1 X_i)) X_i = 0 \rightarrow \sum Y_i X_i = B_0 \sum X_i + B_1 \sum X_i^2 \end{array} \right\} \text{مشتق گیری}$$

$$B_0 = \bar{Y} - B_1 \bar{X}, \quad B_1 = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sum (X_i - \bar{X})^2}$$

3.

$$Cov(X, Y) = \sum (X_i - \bar{X})(Y_i - \bar{Y}), \quad Var(X) = \sum (X_i - \bar{X})^2$$

$$B_1 = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sum (X_i - \bar{X})^2}, \quad B_0 = \bar{Y} - B_1 \bar{X}$$

$$\text{با باز کردن عبارت: } \sum Y_i X_i = B_0 \sum X_i + B_1 \sum X_i^2$$

4.

فرمول کلی:

$$\rho = \frac{\text{Cov}(x, y)}{\sigma_x \sigma_y} \rightarrow \rho = \frac{\beta_1 \text{Var}(x)}{\sigma_x \sigma_y} \Rightarrow \frac{\rho \sigma_x \sigma_y}{\sigma_x^2} \rightarrow \beta_1 \rightarrow \beta_1 = \rho \frac{\sigma_y}{\sigma_x}$$

$$\beta_1 = \frac{\text{Cov}(x, y)}{\text{Var}(x)}$$

Corr: مستقیماً β_1 تأثیر ندارد (در σ_x و σ_y هست) زیرا اینها کمر دارند. β_1 باید 2 باشد.

ضرب مثبت یا منفی خواهد شد. اگر $\rho = 0$ باشد 2 خط 0 خواهد شد.

یکدست‌تر بود. در این حالت که اگر ρ کوچک‌تر و σ_y خیلی بزرگتر از σ_x باشد β_1 هم بزرگ‌تر.

Scanned with CamScanner

5. β_0 به طور مستقیم با $-X$ و $-Y$ مرتب است زیرا به گونه ای محاسبه می شود که اطمینان حاصل شود که خط رگرسیون از مرکز توزیع داده عبور می کند. این مدل رگرسیون را قابل اعتماد می کند و تضمین می کند که روند مشاهده شده در مجموعه داده را به درستی نشان می دهد.

فرمول نهایی β_1 هم در قسمت های قبلی داده شده است.

6. در رگرسیون خطی، ضریب شیب (β_1) نشاندهنده میزان تغییرات متغیر وابسته Y به ازای هر واحد تغییر در متغیر مستقل X است. این ضریب به رابطه آماری میان دو متغیر بستگی دارد و با اندازه گیری همبستگی و نحوه تغییرات آنها نسبت به هم به دست می آید. اگر دو متغیر با هم همبستگی قوی داشته باشند، ضریب شیب بزرگتر و نشاندهنده تأثیر بیشتر X بر Y خواهد بود. همچنین ضریب عرض از مبدا (β_0) مقدار پیشبینی شده برای Y زمانی که X برابر صفر است را نشان می دهد و به میانگینهای هر دو متغیر وابسته است. این ضرایب به ما کمک می کنند که رابطه خطی میان دو متغیر را بهتر بفهمیم و بتوانیم پی شبنیهای دقیقتری انجام دهیم.

ب)

```
# data
x = [16, 27, 11, 20, 30, 25, 5, 24, 21, 10]
y = [46, 80, 36, 52, 98, 75, 10, 70, 64, 30]

n = len(x)

# Mean
mean_x = sum(x) / n
mean_y = sum(y) / n

# Variance
variance_x = sum((xi - mean_x) ** 2 for xi in x) / (n - 1)
variance_y = sum((yi - mean_y) ** 2 for yi in y) / (n - 1)

# Covariance
covariance_xy = sum((x[i] - mean_x) * (y[i] - mean_y) for i in range(n)) / (n - 1)

# Correlation
correlation_xy = covariance_xy / (variance_x ** 0.5 * variance_y ** 0.5)

# Print results
print(f"Mean of x: {mean_x}")
print(f"Mean of y: {mean_y}")
print(f"Variance of x: {variance_x}")
print(f"Variance of y: {variance_y}")
print(f"Covariance between x and y: {covariance_xy}")
print(f"Correlation between x and y: {correlation_xy}")
```

```
⇒ Mean of x: 18.9
   Mean of y: 56.1
   Variance of x: 66.76666666666665
   Variance of y: 696.5444444444444
   Covariance between x and y: 213.1222222222222
   Correlation between x and y: 0.9882674062434095
```

سوال سوم

```

from google.colab import drive
drive.mount('/content/drive')

↳ Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_re

data = "/content/drive/MyDrive/Colab Notebooks/HW/ML/Q3_Dataset/*"

import cv2 as cv
import glob

image_paths = glob.glob(data)
features = []
labels = []

for image_path in image_paths:
    img = cv.imread(image_path)
    img = cv.cvtColor(img, cv.COLOR_BGR2RGB)

    if img is not None:
        features.append(img)

        label = image_path.split("/")[-1].removesuffix(".jpg")[0]
        if label == 'm':
            label = 0
        else:
            label = 1
        labels.append(label)

    else:
        print(f"Could not load image: {image_path}")

def classify_image(image):
    mean_r = image[:, :, 0].mean()
    mean_b = image[:, :, 2].mean()

    if mean_r > mean_b:
        return 0 # ManU
    else:
        return 1 # Chelsea

from sklearn.metrics import confusion_matrix, accuracy_score, precision_score, recall_score

y_pred = [classify_image(image) for image in features]

cm = confusion_matrix(labels, y_pred)
print("Confusion Matrix:")
print(cm)

↳ Confusion Matrix:
[[54  1]
 [21 46]]

accuracy = accuracy_score(labels, y_pred)
precision = precision_score(labels, y_pred)
recall = recall_score(labels, y_pred)

print(f"Accuracy: {accuracy:.2f}")
print(f"Precision: {precision:.2f}")
print(f"Recall: {recall:.2f}")

↳ Accuracy: 0.82
Precision: 0.98
Recall: 0.69

```

سوال چهارم

فایل پیوست

