Praca domowa 1
Ryzyko empiryczne oraz metoda najmniejszych kwadratów

Damian Jankowski s188597

23 kwietnia 2023

1 Wstęp

Celem tego sprawozdania jest przedstawienie dwóch ważnych koncepcji w dziedzinie sztucznej inteligencji - wyznaczania parametrów modelu statystycznego metodą najmniejszych kwadratów oraz obliczania ryzyka empirycznego. Metoda najmniejszych kwadratów jest jedną z najpopularniejszych technik szacowania parametrów modelu na podstawie danych. a ryzyko empiryczne jest ważnym narzędziem do oceny jakości modelu i prognozowania jego skuteczności na nowych danych.

Poniżej znajdują się rozwiązania przykładowych zadań z zastosowaniem tych pojęć.

2 Zadanie 1

2.1 Dane

Mamy zbiór danych wejściowych x oraz wyników oczekiwanych d.

$$\mathbf{x} = \begin{bmatrix} -1.21 \\ 0.1 \\ 0.4 \\ 0.82 \end{bmatrix} \mathbf{d} = \begin{bmatrix} 2.3 \\ 3.4 \\ 5.1 \\ 6.3 \end{bmatrix}$$

Odwzorowanie jest przedstawione wzorem $y = f(x, w) = w_1 x^2 + w_2 x$.

2.2 Rozwiązanie

2.2.1 Wyznaczenie parametrów modelu

Parametry modelu można wyznaczyć za pomocą metody najmniejszych kwadratów.

Najpierw należy wyznaczyć równanie ryzyka empirycznego, korzystając z tego wzoru:

$$E_{emp}(w) = \frac{1}{n}\sum_{i=1}^{n}(f(x_i, w) - d_i)^2 = \frac{1}{n}\sum_{i=1}^{n}(w_1 x_i^2 + w_2 x_i - d_i)^2 \tag{1}$$

$$\begin{aligned} E_{emp}(w) = \frac{1}{4}\Big(&((-1.21)^2 w_1 - 1.21 w_2 - 2.3)^2 + ((0.1)^2 w_1 + 0.1 w_2 - 3.4)^2 + \\ &((0.4)^2 w_1 + 0.4 w_2 - 5.1)^2 + ((0.82)^2 w_1 + 0.82 w_2 - 6.3)^2\Big) \\ = 0.655353 w_1^2 &- 0.577595 w_1 w_2 - 4.22678 w_1 + 0.576625 w_2^2 - 2.3815 w_2 + 20.6575 \end{aligned}$$

Następnie należy wyznaczyć pochodne cząstkowe i wyznaczyć minimum funkcji $E_{emp}(w)$, czyli punkt, w którym pochodne cząstkowe są zerowe.

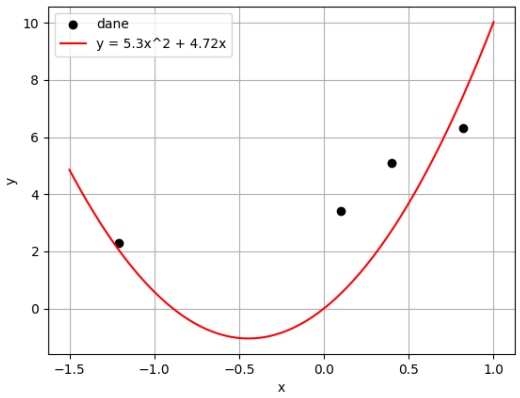
$$\begin{cases} \dfrac{\partial E_{emp}}{\partial w_1} = 0 \\ \dfrac{\partial E_{emp}}{\partial w_2} = 0 \end{cases} \tag{2}$$

Dla wyznaczonej funkcji $E_{emp}(w)$ otrzymujemy:

$$\begin{cases} 1.31071w_1 - 0.577595w_2 - 4.22678 = 0 \\ -0.577595w_1 + 1.15325w_2 - 2.3815 = 0 \end{cases}$$

$$\begin{cases} w_1 = 5.30587682 \\ w_2 = 4.72244169 \end{cases}$$

Wzór na funkcję $f(x, w)$ przyjmuje postać $y = f(x, w) = 5.30587682x^2 + 4.72244169x$.



Rysunek 1: Wykres funkcji $f(x, w)$

2.2.2 Obliczenie ryzyka empirycznego

By obliczyć ryzyko empiryczne, należy podstawić do wzoru na $E_{emp}(w)$ obliczone wcześniej wagi.
Dla tego przykładu otrzymujemy:

$$E_{emp}(w) = 3.800878800281463$$

3 Zadanie 2

3.1 Dane

Mamy zbiór danych wejściowych x_1, x_2 oraz wyników oczekiwanych d.

$$\mathbf{x1} = \begin{bmatrix} 0.2 \\ -0.3 \\ -0.5 \\ -0.1 \\ -1.0 \\ -0.3 \\ 0.1 \end{bmatrix} \mathbf{x2} = \begin{bmatrix} 0.3 \\ 0.4 \\ 3.3 \\ 4.8 \\ 3.2 \\ 7.2 \\ 3.4 \end{bmatrix} \mathbf{d} = \begin{bmatrix} 0.8 \\ 0.2 \\ -0.3 \\ 1.2 \\ 1.6 \\ 0.5 \\ -0.2 \end{bmatrix}$$

Odwzorowanie jest przedstawione wzorem $y = f(x, w) = w_1x_1 + w_2x + w_3x_1x_2 + w_4x_1/x_2 + w_5x_1^2 + w_6x_2^2$.

3.2 Rozwiązanie

Dla podanych danych wagi wynoszą: $\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{bmatrix} = \begin{bmatrix} -35.50911108 \\ -1.60709009 \\ 9.83501361 \\ 11.68327077 \\ 1.78661579 \\ 0.44397207 \end{bmatrix}$. Natomiast $E(w)$ wynosi 0.0306398.

4 Zadanie 3

4.1 Dane

Dane wejściowe x_1, x_2 oraz d są takie same jak w zadaniu 2. Natomiast odzworowanie jest przedstawione wzorem $y = f(x, w) = w_1x_1 + w_2x + w_3x_1x_2 + w_4x_1/x_2$.

4.2 Rozwiązanie

Dla podanych danych wagi wynoszą: $\mathbf{w} = \begin{bmatrix} -2.78280549 \\ 0.09515916 \\ 0.45212008 \\ 1.33611354 \end{bmatrix}$. Natomiast $E(w)$ wynosi 2.2706643.

5 Wnioski

Metoda najmniejszych kwadratów jest prostą i szybką metodą estymacji parametrów modelu, która dobrze sprawdza się w przypadku prostych modeli.

Metoda ta ma jednak kilka wad, takich jak wrażliwość na odstające wartości oraz brak możliwości uwzględnienia zmiennych kategorycznych czy nieliniowych zależności.

W przypadku bardziej skomplikowanych modeli, gdzie zależność między zmiennymi jest nieliniowa, lepsze rezultaty daje stosowanie metod gradientowych, które pozwalają na znalezienie minimum funkcji kosztu w bardziej ogólnym przypadku.

Metoda gradientowa jest jednak bardziej złożona i wymaga większej liczby obliczeń niż metoda najmniejszych kwadratów.

W praktyce stosuje się różne metody w zależności od charakteru danych oraz celu modelowania.

6 Kod programu

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

# Zadanie 1
x = np.array([-1.21, 0.1, 0.4, 0.82])
d = np.array([2.3, 3.4, 5.1, 6.3])
A = np.zeros((4, 2))
for i in range(4):
    A[i, :] = [x[i] ** 2, x[i]]
w = np.linalg.inv(A.T.dot(A)).dot(A.T).dot(d)
print(w)

emp = 0
for i in range(4):
    emp += (w[0] * x[i] ** 2 + w[1] * x[i] - d[i]) ** 2
emp /= 4
print(emp)

plt.scatter(x, d, color='black', label='dane')

x = np.linspace(-1.5, 1, 100)
y = w[0] * x ** 2 + w[1] * x
plt.plot(x, y, '-r', label='y = 5.3x^2 + 4.72x')
plt.grid()
plt.xlabel('x')
plt.ylabel('y')
plt.legend()
plt.savefig('plot.png')
plt.show()

# Zadanie 2
```

```
x1 = np.array([0.2, -0.3, -0.5, -0.1, -1.0, -0.3, 0.1])
x2 = np.array([0.3, 0.4, 3.3, 4.8, 3.2, 7.2, 3.4])
d = np.array([0.8, 0.2, -0.3, 1.2, 1.6, 0.5, -0.2])
A = np.zeros((7, 6))
for i in range(7):
    A[i, :] = [x1[i], x2[i], x1[i] * x2[i],
               x1[i] / x2[i], x1[i] ** 2, x2[i] ** 2]
w = np.linalg.inv(A.T.dot(A)).dot(A.T).dot(d)
print(w)

emp = 0
for i in range(7):
    emp += (w[0] * x1[i] + w[1] * x2[i] + w[2] * x1[i] * x2[i] +
            w[3] * x1[i] / x2[i] + w[4] * x1[i] ** 2 +
            w[5] * x2[i] ** 2 - d[i]) ** 2
print(emp)

# Zadanie 3
x1 = np.array([0.2, -0.3, -0.5, -0.1, -1.0, -0.3, 0.1])
x2 = np.array([0.3, 0.4, 3.3, 4.8, 3.2, 7.2, 3.4])
d = np.array([0.8, 0.2, -0.3, 1.2, 1.6, 0.5, -0.2])
A = np.zeros((7, 4))
for i in range(7):
    A[i, :] = [x1[i], x2[i], x1[i] * x2[i], x1[i] / x2[i]]
w = np.linalg.inv(A.T.dot(A)).dot(A.T).dot(d)
print(w)

emp = 0
for i in range(7):
    emp += (w[0] * x1[i] + w[1] * x2[i] + w[2] * x1[i] * x2[i] +
            w[3] * x1[i] / x2[i] - d[i]) ** 2
print(emp)
```