Кафедра дискретной математики МФТИ

Курс математической статистики

Игашов Илья, 593 групппа

Задание №9

№2. ¶

Пусть $X_i = \beta_1 + i\beta_2 + \epsilon_0 + \dots + \epsilon_i, \;\; i = 0, 1, \dots, n$ — расстояния, которое проехал трамвай за i секунд по показанию датчика. Здесь β_1 — начальное расстояние, β_2 — скорость трамвая, ϵ_0 — ошибка начального показания датчика. Трамвай едет с постоянной скоростью, и через каждую секунду датчик фиксирует расстояние, которое проехал трамвай. Отсчет времени идет от предыдущего замера, причем отсчет происходит с ошибкой. Для $i = 1, \dots, n$ величина ϵ_i есть ошибка приращения расстояния, то есть то есть $\epsilon_i = \epsilon_i^t \beta_2$, где ϵ_i^t — ошибка отсчета времени. Все ошибки ϵ_i независимы и распределены по закону $N(0, \sigma^2)$. Сведите задачу к линейной модели и найдите оценки наименьших квадратов для начального расстояния β_1 и скорости β_2, а также несмещенную оценку для σ^2, из которой выразите оценку дисперсии отсчета времени. Данные взять из файла на диске. Сделайте выводы.

In [1]:

```
import numpy as np
from numpy import linalg
```

In [2]:

```
# Открываем и читаем файл.
f = open('Regression.csv', 'r')
text = f.read()
data = np.array(text.split('\n')[:-1]).astype(float)
N = len(data)
```

Вместо измерений $X = (X_0, \dots, X_n)^T$ рассмотрим измерения $\hat{X} = (X_0, X_1 - X_0, X_2 - X_1, \dots, X_n - X_{n-1})^T$.

Тогда $\hat{X} = (\beta_1 + \epsilon_0, \beta_2 + \epsilon_1, \dots, \beta_2 + \epsilon_n)^T = l + \epsilon$, где

$$\epsilon = (\epsilon_0, \dots, \epsilon_n)^T, \;\; l = Z\theta, \;\; \theta = (\beta_1, \beta_2)^T, \;\; Z = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ \dots & \dots \\ 0 & 1 \end{pmatrix}.$$

Посчитаем оценку по методу наименьших квадратов для $\theta = (\beta_1, \beta_2)^T$:

$$\theta^* = (Z^T Z)^{-1} Z^T \hat{X}.$$

Посчитаем несмещенную оценку для σ^2:

$$\sigma^{2^*} = \frac{1}{m-k} \left\| \hat{X} - Z\theta^* \right\|^2, \;\; \text{где} \;\; m = n + 1 = 1000, \;\; k = 2.$$

In [3]:

```
# Данные в новом виде:
X = np.array([
        data[i] - data[i-1] if i > 0 else data[i]
        for i in range(N)
    ])

# Считаем оценки:
Z = np.array([[1, 0] if i == 0 else [0, 1] for i in range(N)])
theta = linalg.inv(Z.T @ Z) @ Z.T @ X
sigma = (linalg.norm(X - Z @ theta) ** 2) / (N - 2)

# Вывод:
print('Initial distance: %.2f' % theta[0])
print('Velocity: %.2f' % theta[1])
print('Error variance: %.2f' % sigma)
```

```
Initial distance: 63.57
Velocity: 9.97
Error variance: 4.22
```

Поскольку для $i = 1, \ldots, n$ величина ϵ_i есть ошибка приращения расстояния, то есть $\epsilon_i = \epsilon_i^t \beta_2$, где ϵ_i^t — ошибка отсчета времени, и $\epsilon_i \sim N(0, \sigma^2)$, то

$$\epsilon_i^t \sim N(0, \frac{\sigma^2}{\beta_2^2}),$$

и оценка дисперсии отсчета времени выражается как

$$\sigma_t^{2^*} = \frac{\sigma^{2^*}}{\beta_2^{*^2}}.$$

In [4]:

```
# Оценка дисперсии отсчета времени:
sigma_t = sigma / (theta[1] ** 2)
print('Variance of time marking: %.2f' % sigma_t)
```

```
Variance of time marking: 0.04
```

Вывод

Для решения поставленной задачи мы свели ее к регрессионной модели. Мы нашли оценки начального расстояния β_1 и скорости трамвая β_2 по методу наименьших квадратов:

$$\beta_1^* = 63.57,$$
$$\beta_2^* = 9.97.$$

Кроме того, мы оценили (несмещенные оценки) дисперсию ошибки приращения расстояния:

$$\sigma^{2^*} = 4.22$$

и дисперсию отсчета времени:

$$\sigma_t^{2^*} = 0.04$$

в предположении, что все ошибки ϵ_i независимы и распределены по закону $N(0, \sigma^2)$.

In [ ]:

```

```