Кафедра дискретной математики МФТИ

Курс математической статистики

Игашов Илья, 593 группа

Задание №9

№1.

Загрузите данные из набора Forest Fires о лесных пожарах в Португалии. Задача состоит в том, чтобы с помощью линейной регрессии научиться предсказывать координату area (площадь пожара) в виде линейной комбинации других данных.

Преобразование данных. Чтобы работать с числовыми координатами, нечисловые координаты (month, day) нужно перевести в числовые. Для простоты можно заменить координату month на индикатор летнего сезона, а координату day не использовать вообще. По желанию можете сделать преобразование другим способом. Также желательно добавить координату, тождественно равную единице. Она будет отвечать свободному члену.

Разбейте выборку на две части в соотношении $7:3$. Перед этим желательно ее перемешать (random.shuffle). По первой части постройте регрессионную модель. Примените модель ко второй части выборки и посчитайте по ней среднеквадратичную ошибку.

Сделайте для area преобразование $f(x) = \ln(c + x)$ и постройте для нее регрессионную модель. Посчитайте среднеквадратичную ошибку для преобразованных значений по данному правилу и для исходных, применив в последнем случае к оценкам обратное к f преобразование. При каком с предсказания получаются лучше всего?

При выбраном с сделайте разбиение выборки в соотношении $7:3$ разными способами (перемешивая каждый раз). Сильно ли зависит качество от способа разбиения? Сделайте выводы.

In [12]:

```
import numpy as np
from numpy import linalg
from random import shuffle
from math import log
```

In [2]:

```
# Открываем и читаем файл.
f = open('forestfires.csv', 'r')
text = f.read()
lines = text.split('\n')
```

In [3]:

```
attributes = np.array(lines[0].split(',')) # Названия атрибутов
observations = np.array([line.split(',') for line in lines[1:-1]]) # Векторы измерени
```

In [4]:

```
print(observations)
```

```
[['7' '5' 'mar' ..., '6.7' '0' '0']
 ['7' '4' 'oct' ..., '0.9' '0' '0']
 ['7' '4' 'oct' ..., '1.3' '0' '0']
 ...,
 ['7' '4' 'aug' ..., '6.7' '0' '11.16']
 ['1' '4' 'aug' ..., '4' '0' '0']
 ['6' '3' 'nov' ..., '4.5' '0' '0']]
```

In [5]:

```
# Заменим координату "month" на индикатор летнего сезона, координату "day"  на единицу - свобо
for obs in observations:
    obs[2] = 1 if (obs[2] in ['jun', 'jul', 'aug']) else 0
    obs[3] = 1

# Изменим тип данных.
observations = (observations.astype(float)).astype(int)
```

In [6]:

```
print(observations)
```

```
[[ 7  5  0 ...,  6  0  0]
 [ 7  4  0 ...,  0  0  0]
 [ 7  4  0 ...,  1  0  0]
 ...,
 [ 7  4  1 ...,  6  0 11]
 [ 1  4  1 ...,  4  0  0]
 [ 6  3  0 ...,  4  0  0]]
```

In [7]:

```
# Перемешали выборку и разделили в отношении ~ 7:3 на обучающую и тестовую.
shuffle(observations)
educ_data = observations[:7 * int(len(observations) / 10)]
test_data = observations[7 * int(len(observations) / 10):]
```

Регрессионная модель.

Имееем N векторов $X_1, \dots, X_N$.

Рассмотрим вектор $X_i = (X_i^1, X_i^2, \dots, X_i^K)$, где $K = 13$ — данные, дающие информацию о конкретном пожаре. Будем считать, что $\forall i \in \{1, \dots, N\}$ мы измерили только координату X_i^K (это и есть площадь пожара area), а остальные данные $X_i^1, X_i^2, \dots, X_i^{K-1}$ нам известны. Тогда area можно представить как линейную комбинацию остальных данных:

$$\forall i \in \{1, \dots, N\} \;\; X_i^K = \theta_1 X_i^1 + \theta_2 X_i^2 + \dots + \theta_{K-1} X_i^{K-1}.$$

Оценим параметры $\theta = (\theta_1, \dots, \theta_{K-1})^T$.

$$\theta = (Z^T Z)^{-1} Z^T \hat{X}, \;\; \text{где} \;\; Z = \begin{pmatrix} X_1^1 & X_1^2 & \dots & X_1^{K-1} \\ X_2^1 & X_2^2 & \dots & X_2^{K-1} \\ \dots & \dots & \dots & \dots \\ X_N^1 & X_N^2 & \dots & X_N^{K-1} \end{pmatrix}, \;\; \hat{X} = (X_1^K, \dots, X_N^K)^T.$$

Оценка для area на тестовых данных (известны первые $K - 1 = 12$ координат каждого вектора) равна:

$$\hat{A} = Z\theta.$$

In [8]:

```
# Линейная регрессионная модель на обучающей выборке:
# theta - оценка коэффициентов в линейной комбинации.
Z = educ_data[:, :-1]
X = educ_data[:, -1]
theta = linalg.inv(Z.T @ Z) @ Z.T @ X
```

In [11]:

```
# Считаем площадь пожаров на тестовых данных.
Z = test_data[:, :-1]
X = test_data[:, -1]
A = Z @ theta # Оценка для area

# Среднеквадратичная ошибка:
err = np.sum((X - result) ** 2) / len(X)
print(err)
```

```
425.949525747
```

Вывод

Мы построили линейную регрессионную модель для предсказания площади пожара по известным данным. Среднеквадратичная ошибка на тестовой выборке оказалась равна 425.95.