МИНОБРНАУКИ РОССИИ

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ

ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)

Кафедра МО ЭВМ

ОТЧЕТ

по практическому заданию №1

по дисциплине «Машинное обучение»

Студент гр. 6304	________________	Иванов Д.В.
Преподаватель	________________	Жангиров Т. Р.

Санкт-Петербург

2020

Задание 1

Исходные данные

```
X = [69, 74, 68, 70, 72, 67, 66, 70, 76, 68, 72, 79, 74, 67, 66, 71, 74, 75
, 75, 76]
Y = [153, 175, 155, 135, 172, 150, 115, 137, 200, 130, 140, 265, 185, 112,
140,  150, 165, 185, 210, 220]
```

A. Среднее, медиана, мода величины X

X	66	67	68	69	70	71	72	74	75	76	79
p_i	0.1	0.1	0.1	0.05	0.1	0.05	0.1	0.15	0.1	0.1	0.05

a. Среднее

$$\mu = \sum_x x f(x) = 71.45$$

b. Медиана

```
sorted_X = sorted(X)
len_X = len(sorted_X)
median = sorted_X[(len_X+1)//2] if len_X % 2 == 1 else (sorted_X[len_X//2-
1] + sorted_X[len_X//2])/2
```

$$m = 71.5$$

c. Мода

$$mode(X) = arg\ \max_x f(x) = 74$$

B. Дисперсия величины Y

Y	112	115	130	135	137	140	150	153	155	165	172	175	185	200	210	220	265
p_i	0.05	0.05	0.05	0.05	0.05	0.1	0.1	0.05	0.05	0.05	0.05	0.05	0.1	0.05	0.05	0.05	0.05

$$\mu\ = 164.7$$

$$\sigma^2 = var(Y) = \sum_x (x - \mu)^2 f(x) = 1369.21$$

- При использовании pandas:

```
df = pd.DataFrame({'Y': Y})
df.Y.var()
➢  1441.2
```

C. График плотности нормального распределения для X.

```
df = pd.DataFrame({'X': X})
df.X. density()
```

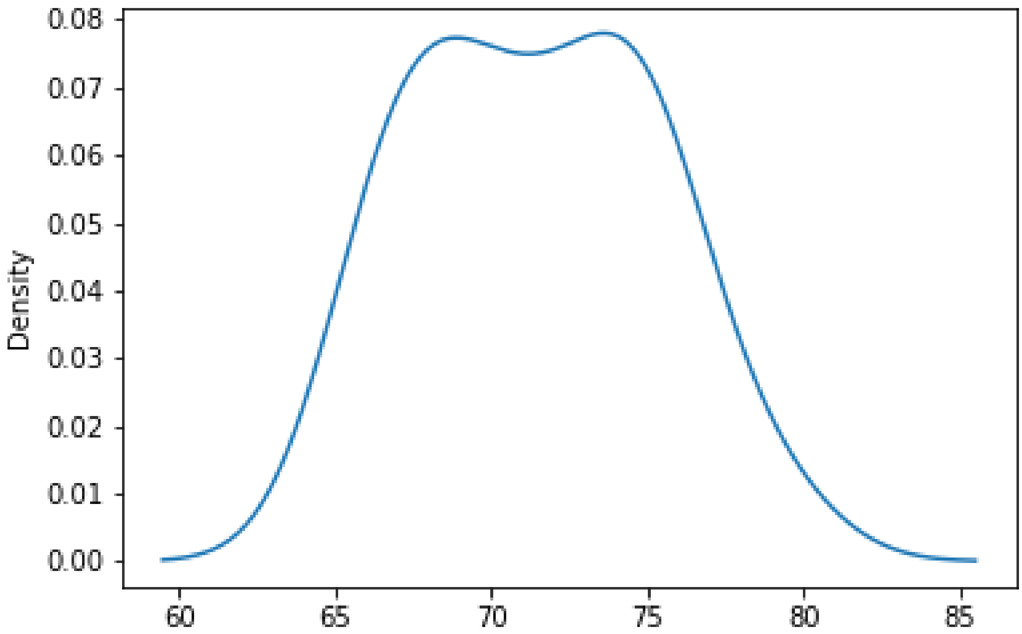

Рисунок 1 — Плотность распределения X.

D. Вероятность того, что возраст > 80.

```
scaler = StandardScaler().fit(df.X.to_numpy().reshape((-1,1)))
t.sf(x=scaler.transform([[80]])[0][0], df=df.X.size-1)
➢  0.0165564
```

Стандартизация значений величины выборки, а затем вычисление вероятности $P(X > 80)$.

E. Двумерное математическое ожидание и ковариационная матрица

$$\boldsymbol{\mu} = E[\boldsymbol{X}] = E\left[\begin{pmatrix} X_1 \\ X_2 \end{pmatrix}\right] = E\begin{pmatrix} E[X_1] \\ E[X_2] \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} = \begin{pmatrix} 71.45 \\ 164.7 \end{pmatrix}$$

```
df.cov()
```

$$\boldsymbol{\Sigma} = E[(\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})^T] = \begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{21} & \sigma_2^2 \end{pmatrix} = \begin{pmatrix} 14.58 & 128.9 \\ 128.9 & 1441.2 \end{pmatrix}$$

F. Корреляция

$$\rho_{12} = \frac{\sigma_{12}}{\sigma_1 \sigma_2} = \frac{\sigma_{12}}{\sqrt{\sigma_1^2 \sigma_2^2}} = \frac{128.9}{\sqrt{14.58 * 1441.2}} = 0.8892257$$

- При использовании pandas:

```
df.corr().loc['X','Y']
➢  0.8891701351748048
```

G. Диаграмма рассеяния, отображающая зависимость между возрастом и весом.

```
df.plot.scatter(x='X', y='Y')
```

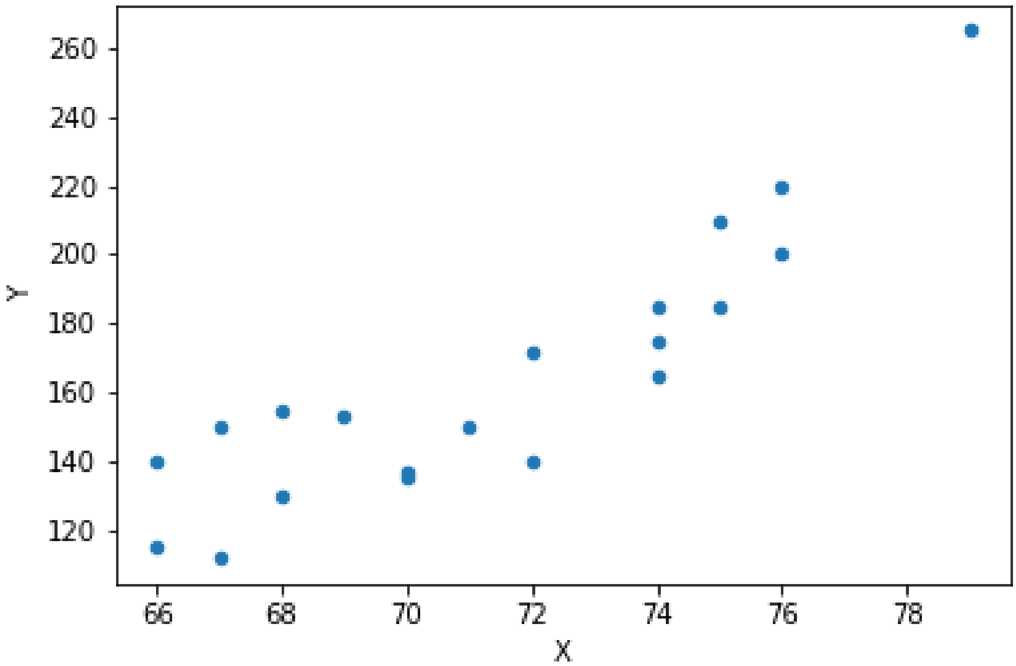

Рисунок 2 — Диаграмма рассеяния.

Задание 2

```
df = pd.DataFrame({'X1': [17,11,11], 'X2': [17,9,8], 'X3': [12,13,19]})
cov = df.cov()
```

$$\mathbf{\Sigma} = E[(\mathbf{X}-\boldsymbol{\mu})(\mathbf{X}-\boldsymbol{\mu})^T] = \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_2^2 & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_3^2 \end{pmatrix}$$

$$= \begin{pmatrix} 12.0 & 17.0 & -8.0 \\ 17.0 & 24.33 & -12.83 \\ -8.0 & -12.83 & 14.33 \end{pmatrix}$$

```
np.linalg.det(cov)
```

$$\det(\mathbf{\Sigma}) = 0.0$$

Задание 3

A. Какое из распределений сгенерировало значение с большей вероятностью

```
distributions = ((4, 1), (8, 2))
values = (5, 6, 7)

find_std = lambda value, mean, std: (value - mean) / std

for val in values:
    stds = [abs(find_std(val, *info)) for info in distributions]
    print(val, stds.index(min(stds)))

➢  5 0
➢  6 1
➢  7 1
```

Вычисление минимальной удаленности от мат. ожидания в СКО.

Для `5` - первое, для `6` и `7` - второе.

B. Значения, которое может быть сгенерировано обоими распределениями с равной вероятностью

```
res = minimize_scalar(lambda x: abs(sum([(find_std(x, *info)) for info in d
istributions])))
res.x
```

Минимизация функции абсолютной суммы значений удаленности от мат. ожидания в СКО. График минимизируемой функции (рис. 4)

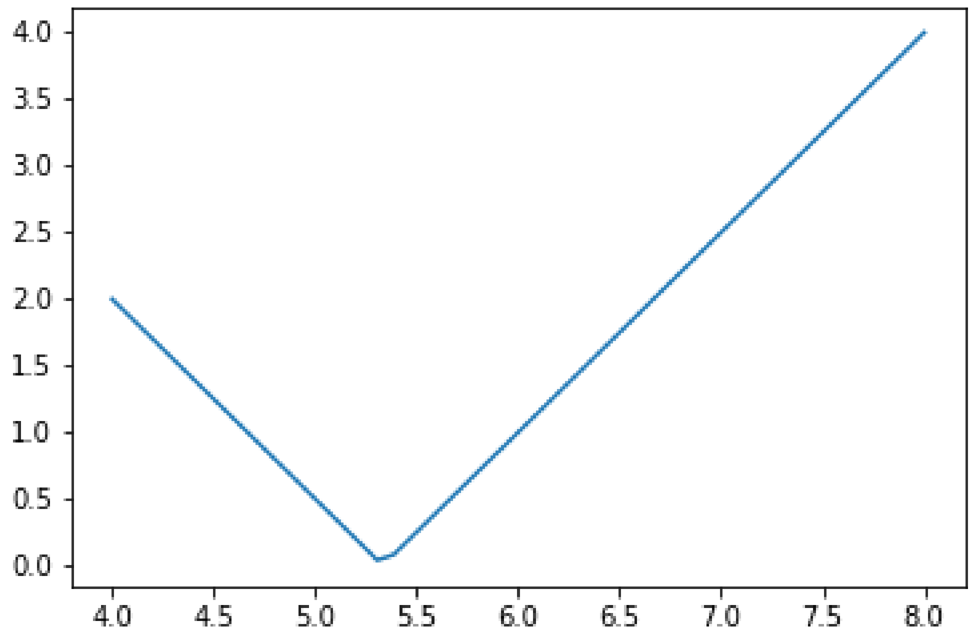

Рисунок 3 — Функция абсолютной суммы значений удаленности для случайной величины.