

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра информационных технологий

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

Дисциплина: Методы машинного обучения

Москва 2022

Вариант № 22

В соответствии с индивидуальным заданием, указанным в записной книжке команды или полученным на занятии, скачайте данные для анализа, сделайте необходимые расчеты и постройте следующие визуализации:

1. При помощи модуля `tensorflow_datasets` загрузите указанный в индивидуальном задании набор данных и оставьте в наборе данных три указанных в индивидуальном задании признака (столбца).
2. В соответствии с индивидуальным заданием вычислите необходимые показатели признаков и визуализируйте соответствующие признаки набора данных.

3,4 В соответствии с индивидуальным заданием выполните необходимые расчеты и визуализируйте диаграмму рассеяния для пары признаков.

1

```
Ввод [2]: import tensorflow_datasets as tfds
```

```
Ввод [3]: ds = tfds.load("diamonds", split='train')
ds
```

```
Out[3]: <PrefetchDataset element_spec={'features': {'carat': TensorSpec(shape=(), dtype=tf.float32, name=None), 'clarity': TensorSpec(shape=(), dtype=tf.int64, name=None), 'color': TensorSpec(shape=(), dtype=tf.int64, name=None), 'cut': TensorSpec(shape=(), dtype=tf.int64, name=None), 'depth': TensorSpec(shape=(), dtype=tf.float32, name=None), 'table': TensorSpec(shape=(), dtype=tf.float32, name=None), 'x': TensorSpec(shape=(), dtype=tf.float32, name=None), 'y': TensorSpec(shape=(), dtype=tf.float32, name=None), 'z': TensorSpec(shape=(), dtype=tf.float32, name=None)}, 'price': TensorSpec(shape=(), dtype=tf.float32, name=None)}>
```

Ввод [4]:

df = tfds.as_dataframe(ds)
df.head()

Out[4]:

	features/carat	features/clarity	features/color	features/cut	features/depth	features/table	features/x	features/y
0	1.26	2	4	2	60.599998	60.0	6.97	7.00
1	0.80	3	4	4	62.099998	54.0	5.96	5.99
2	0.56	4	2	4	61.700001	54.0	5.28	5.32
3	1.51	3	6	1	64.000000	58.0	7.24	7.27
4	0.33	6	5	4	62.200001	54.0	4.43	4.45

Ввод [7]:

data = df
data.head()

Out[7]:

	features/carat	features/clarity	features/color	features/cut	features/depth	features/table	features/x	features/y
0	1.26	2	4	2	60.599998	60.0	6.97	7.00
1	0.80	3	4	4	62.099998	54.0	5.96	5.99
2	0.56	4	2	4	61.700001	54.0	5.28	5.32
3	1.51	3	6	1	64.000000	58.0	7.24	7.27
4	0.33	6	5	4	62.200001	54.0	4.43	4.45

Ввод [8]:

data = data[['features/carat', 'features/x', 'features/y']]
data.head()

Out[8]:

	features/carat	features/x	features/y
0	1.26	6.97	7.00
1	0.80	5.96	5.99
2	0.56	5.28	5.32
3	1.51	7.24	7.27
4	0.33	4.43	4.45

Ввод [9]: data.describe()

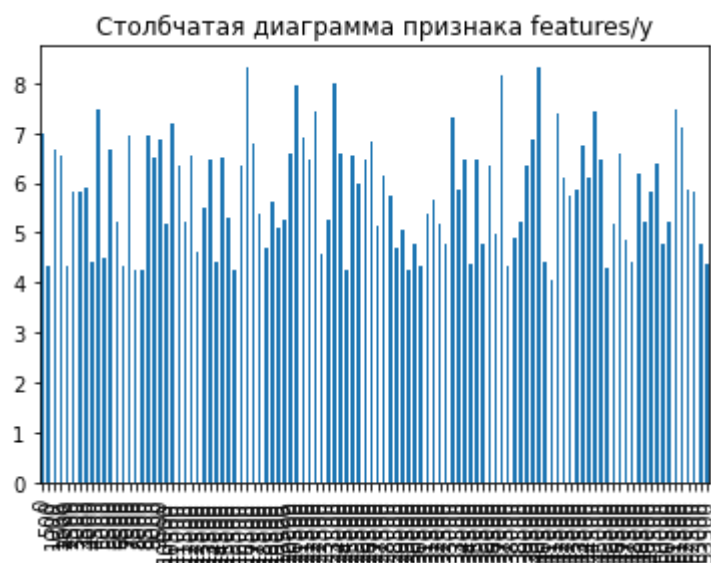
Out[9]:

	features/carat	features/x	features/y
count	53940.000000	53940.000000	53940.000000
mean	0.797950	5.731170	5.734540
std	0.474018	1.121749	1.142133
min	0.200000	0.000000	0.000000
25%	0.400000	4.710000	4.720000
50%	0.700000	5.700000	5.710000
75%	1.040000	6.540000	6.540000
max	5.010000	10.740000	58.900002

Ввод [10]: print (data.max()-data.min())

```
features/carat    4.810000
features/x        10.740000
features/y        58.900002
dtype: float32
```

Ввод [11]: df[:500]['features/y'].plot.bar(title='Столбчатая диаграмма признака features/y'); #



3

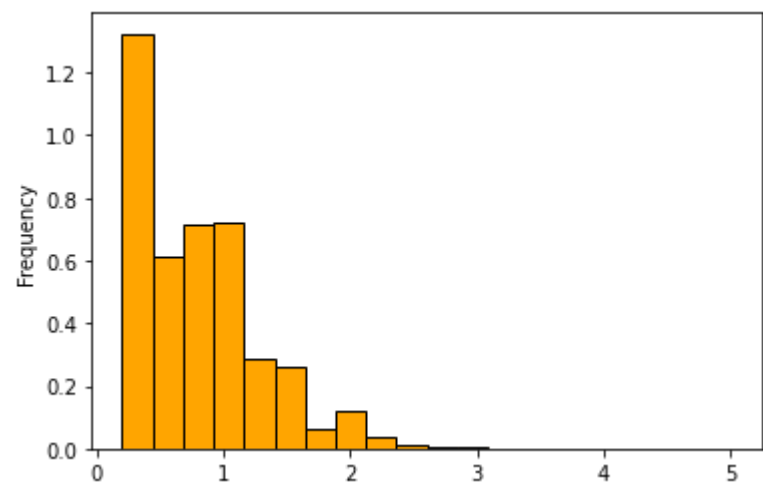
Ввод [13]: data.mean()

```
Out[13]: features/carat    0.79795
features/x    5.73117
features/y    5.73454
dtype: float32
```

Ввод [14]: data.mean().min()

```
Out[14]: 0.7979496717453003
```

```
Ввод [16]: df['features/carat'].plot.hist(color='orange', edgecolor='black', bins=20, density=True)
```



4

```
Ввод [17]: data.cov()
```

Out[17]:

	features/carat	features/x	features/y
features/carat	0.224687	0.518484	0.515248
features/x	0.518484	1.258347	1.248789
features/y	0.515248	1.248789	1.304472

```
Ввод [18]: df.plot.scatter('features/carat','features/y',title='Диаграмма рассеяния признаков carat и y')
```

