

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

по курсу: Data Science

Мавлютова Альфия Галиановна
МГТУ им. Н.Э. Баумана
DS11838/4
2022г.

Мавлютова Альфия Галиановна

- **образование:** высшее экономическое
- **опыт работы :** главный бухгалтер с 2004г.
- **навыки в IT до начала обучения на курсе:** олимпиадный школьный уровень, базовые знания Basic, Pascal, Ассемблер
- **цель обучения:** освоение навыков работы с большими данными, их анализа, обработки, построения моделей, прогнозирования для возможности применить имеющиеся знания и опыт в сфере финансов с современными технологиями и методами.

Прогнозирование конечных свойств новых материалов (композиционных материалов)

Исходные данные:

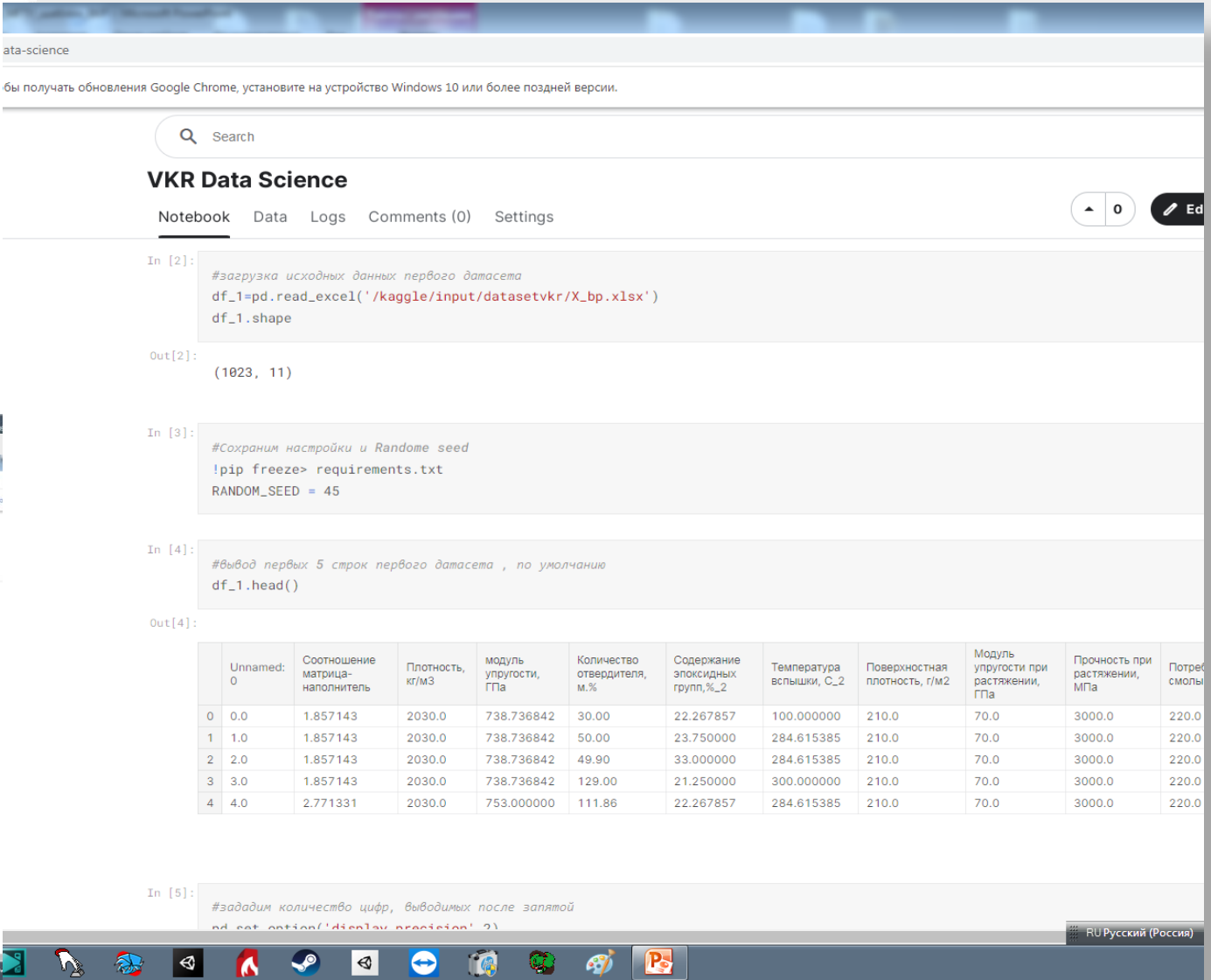
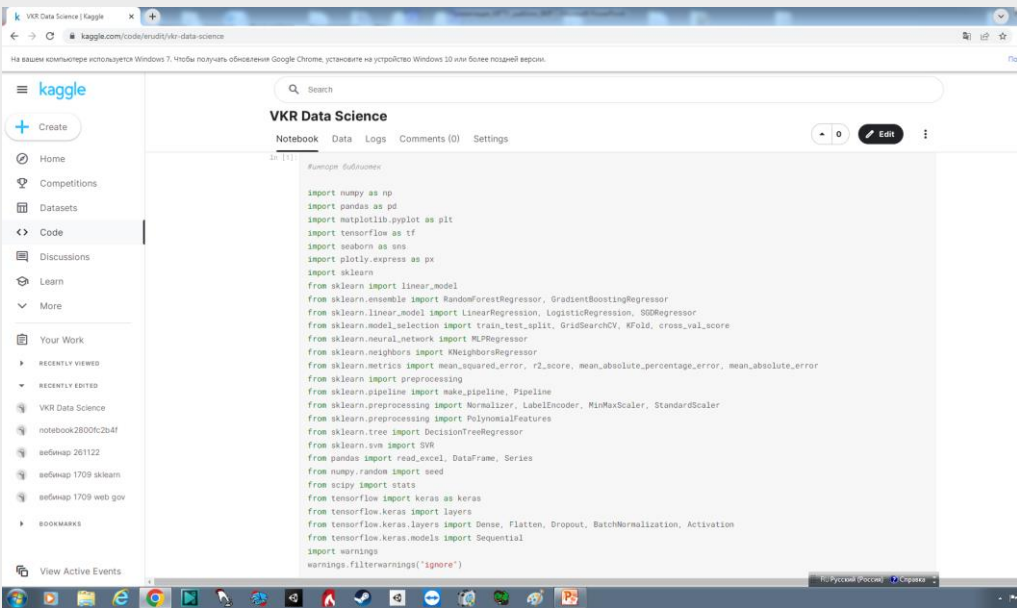
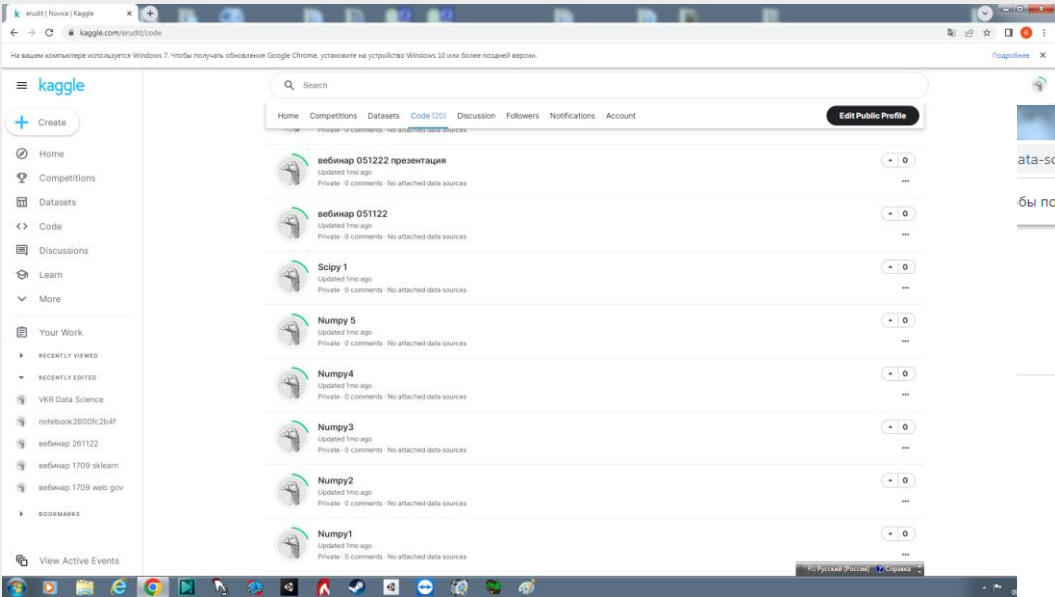
- Данные о параметрах (файл X_br.xlsx)
- Данные нашивок (файл X_nip.xlsx)

Цель :

- Разработать модели для прогноза модули упругости при растяжении, прочности при растяжении и соотношении «матрица-наполнитель»

Этапы работы:

1. Изучен теоретический материал
 - конспектирование лекций
 - повторение практического материала на языке Python в Kaggle
2. Импорт библиотек
3. Загрузка исходных данных (1023,11) и (1040,4)
4. Разведочный анализ данных
 - просмотр начальных и конечных строк датасета
 - округление с указанием количества цифр, вводимых после запятой
 - удаление первого столбца с порядковыми номерами (неинформативный)
 - изучение информации о датасете
 - проверка типов данных в столбцах
 - проверка на наличие пропусков
 - проверка уникальных значений (функция `nunique`)
5. Объединение файлов по индексу по типу объединения INNER
 - размер получившегося DataFrame (двумерный массив) (1023,13)



Ваш браузер использует Windows 7. Чтобы получать обновления Google Chrome, установите на устройство Windows 10 или более поздней версии.

+

к

VKR Data Science | Kaggle

← → ↻

kaggle.com/code/erudit/vkr-data-science

На вашем компьютере используется Windows 7. Чтобы получать обновления Google Chrome, установите на устройство Windows 10 или более поздней версии.

☰ kaggle

+

 Create

🏠 Home

🏆 Competitions

📁 Datasets

🔗 Code

💬 Discussions

🎓 Learn

⌵ More

📁 Your Work

➤ RECENTLY VIEWED

▼ RECENTLY EDITED

📁 VKR Data Science

📁 notebook2800fc2b4f

📁 вебинар 261122

📁 вебинар 1709 sklearn

📁 вебинар 1709 web gov

➤ BOOKMARKS

📁 View Active Events

🔍 Search

VKR Data Science

Notebook Data Logs Comments (0) Settings

In [12]:

```
# объединение исходных файлов в единый набор данных
# согласно условию задачи по типу INNER
df=df_1.merge(df_2,left_index=True, right_index=True, how='inner')
df.head().T
```

Out[12]:

	0	1	2	3	4
Соотношение матрица-наполнитель	1.86	1.86	1.86	1.86	2.77
Плотность, кг/м3	2030.00	2030.00	2030.00	2030.00	2030.00
модуль упругости, ГПа	738.74	738.74	738.74	738.74	753.00
Количество отвердителя, м.%	30.00	50.00	49.90	129.00	111.86
Содержание эпоксидных групп, %_2	22.27	23.75	33.00	21.25	22.27
Температура вспышки, C_2	100.00	284.62	284.62	300.00	284.62
Поверхностная плотность, г/м2	210.00	210.00	210.00	210.00	210.00
Модуль упругости при растяжении, ГПа	70.00	70.00	70.00	70.00	70.00
Прочность при растяжении, МПа	3000.00	3000.00	3000.00	3000.00	3000.00
Потребление смолы, г/м2	220.00	220.00	220.00	220.00	220.00
Угол нашивки, град	0.00	0.00	0.00	0.00	0.00
Шаг нашивки	4.00	4.00	4.00	5.00	5.00
Плотность нашивки	57.00	60.00	70.00	47.00	57.00

In [13]:

```
#размер полученной DataFrame
df.shape
```

Out[13]:

(1023, 13)

Ваш браузер использует Windows 7. Чтобы получать обновления Google Chrome, установите на устройство Windows 10 или более поздней версии.

к

VKR Data Science | Kaggle

← → ↻

kaggle.com/code/erudit/vkr-data-science

На вашем компьютере используется Windows 7. Чтобы получать обновления Google Chrome, установите на устройство Windows 10 или более поздней версии.

☰ kaggle

+

 Create

🏠 Home

🏆 Competitions

📁 Datasets

🔗 Code

💬 Discussions

🎓 Learn

⌵ More

📁 Your Work

➤ RECENTLY VIEWED

▼ RECENTLY EDITED

📁 VKR Data Science

📁 notebook2800fc2b4f

📁 вебинар 261122

📁 вебинар 1709 sklearn

📁 вебинар 1709 web gov

➤ BOOKMARKS

📁 View Active Events

🔍 Search

VKR Data Science

Notebook Data Logs Comments (0) Settings

Разведочный анализ полученного датасета

In [14]:

```
#информация о столбцах
print(df.info())
```

<class 'pandas.core.frame.DataFrame'>
Int64Index: 1023 entries, 0 to 1022
Data columns (total 13 columns):
Column Non-Null Count Dtype

0 Соотношение матрица-наполнитель 1023 non-null float64
1 Плотность, кг/м3 1023 non-null float64
2 модуль упругости, ГПа 1023 non-null float64
3 Количество отвердителя, м.% 1023 non-null float64
4 Содержание эпоксидных групп, %_2 1023 non-null float64
5 Температура вспышки, C_2 1023 non-null float64

In [19]:

```
df.columns
```

Out[19]:

Index(['Соотношение матрица-наполнитель', 'Плотность, кг/м3',
'модуль упругости, ГПа', 'Количество отвердителя, м.%',
'Содержание эпоксидных групп, %_2', 'Температура вспышки, C_2',
'Поверхностная плотность, г/м2', 'Модуль упругости при растяжении, ГПа',
'Прочность при растяжении, МПа', 'Потребление смолы, г/м2',
'Угол нашивки', 'Шаг нашивки', 'Плотность нашивки'],
dtype='object')

In [20]:

```
bin_cols=['Угол нашивки']
```

In [21]:

```
num_cols= ["Соотношение матрица-наполнитель", "Плотность, кг/м3",  
"модуль упругости, ГПа", "Количество отвердителя, м.%",  
"Поверхностная плотность, г/м2", "Модуль упругости при растяжении, ГПа",  
"Прочность при растяжении, МПа", "Потребление смолы, г/м2",  
"Шаг нашивки", "Плотность нашивки"]
```


In [22]:

```
#проверим на наличие пропущенных данных
df.isnull().sum()
```

Out[22]:

Соотношение матрица-наполнитель 0
Плотность, кг/м3 0
модуль упругости, ГПа 0
Количество отвердителя, м.% 0
Содержание эпоксидных групп, %_2 0
Температура вспышки, C_2 0

7

 **ОБРАЗОВАТЕЛЬНЫЙ
ЦЕНТР** МГТУ им. Н. Э. Баумана

Этапы работы

6. Анализ столбцов

- числовые
- категориальные
- бинарные

7. Описательная статистика

- изучение данных (максимальное, минимальное, квартили, медиана и т.д.)
- вычисление коэффициента корреляции Пирсона (статистической зависимости не наблюдается)

8. Визуализация данных

- гистограмма распределения каждой переменной (параметры в большинстве близки к нормальному распределению, иск. «Угол нашивки», т.к. значения 0 и 1)
- диаграмма «ящик с усами» (выбросы в каждом столбце, кроме «Угол нашивки»)
- тепловая карта (корреляция входных переменных очень слабая)

Kaggle interface showing a notebook titled "VKR Data Science". The notebook contains three code cells:

```
[ ]:
#проверим на наличие пропущенных данных
df.isnull().sum()

[ ]:
#проверка на наличие дубликатов
df.duplicated().sum()

[ ]:
#находим среднее, медианное значение для каждой колонки
#согласно условиям задачи
mean_and_50=df.describe()
mean_and_50.loc[['mean', '50%']]
```

The notebook also includes text annotations: "поскольку пропущенных данных нет, то очистка не требуется" and "Дубликатов также нет".

Kaggle interface showing a notebook titled "VKR Data Science". The notebook contains a code cell:

```
In [28]:
#гистограмма распределения
df.hist(figsize=(20,20), color="b")
plt.show()
```

The notebook displays eight histograms showing the distribution of various features: "Соотношение матрица-наполнитель", "Плотность, кг/м3", "модуль упругости, ГПа", "Количество отвердителя, м. %", "Содержание эпоксидных групп, %", "Температура вспенивания, C, 2", "Поверхностная плотность, г/м2", and "Модуль упругости при растяжении, ГПа".

Kaggle interface showing a notebook titled "VKR Data Science". The notebook contains a code cell:

```
In [29]:
#визуализация корреляционной матрицы с помощью тепловой карты
mask=np.triu(df.corr())
plt.figure(figsize=(20,20))
plt.suptitle("Ящик с усами", y=8.9, fontsize=30)
plt.boxplot(pd.DataFrame(scaler.transform(df)), labels=df.columns,
             patch_artist=True, meanline=True, vert=False, boxprops=dict(facecolor='b', color='y'),
             medianprops=dict(color='r'), whiskerprops=dict(color='b'),
             capprops=dict(color='black'), flierprops=dict(color='g', markeredgecolor='r'))
plt.show()
```

The notebook displays a box plot titled "Ящик с усами" showing the distribution of various features. The features are: "Плотность матрицы", "Модуль упругости", "Количество отвердителя", "Содержание эпоксидных групп", "Температура вспенивания", "Поверхностная плотность", "Модуль упругости при растяжении", and "Плотность при растяжении".

Kaggle interface showing a notebook titled "VKR Data Science". The notebook contains a code cell:

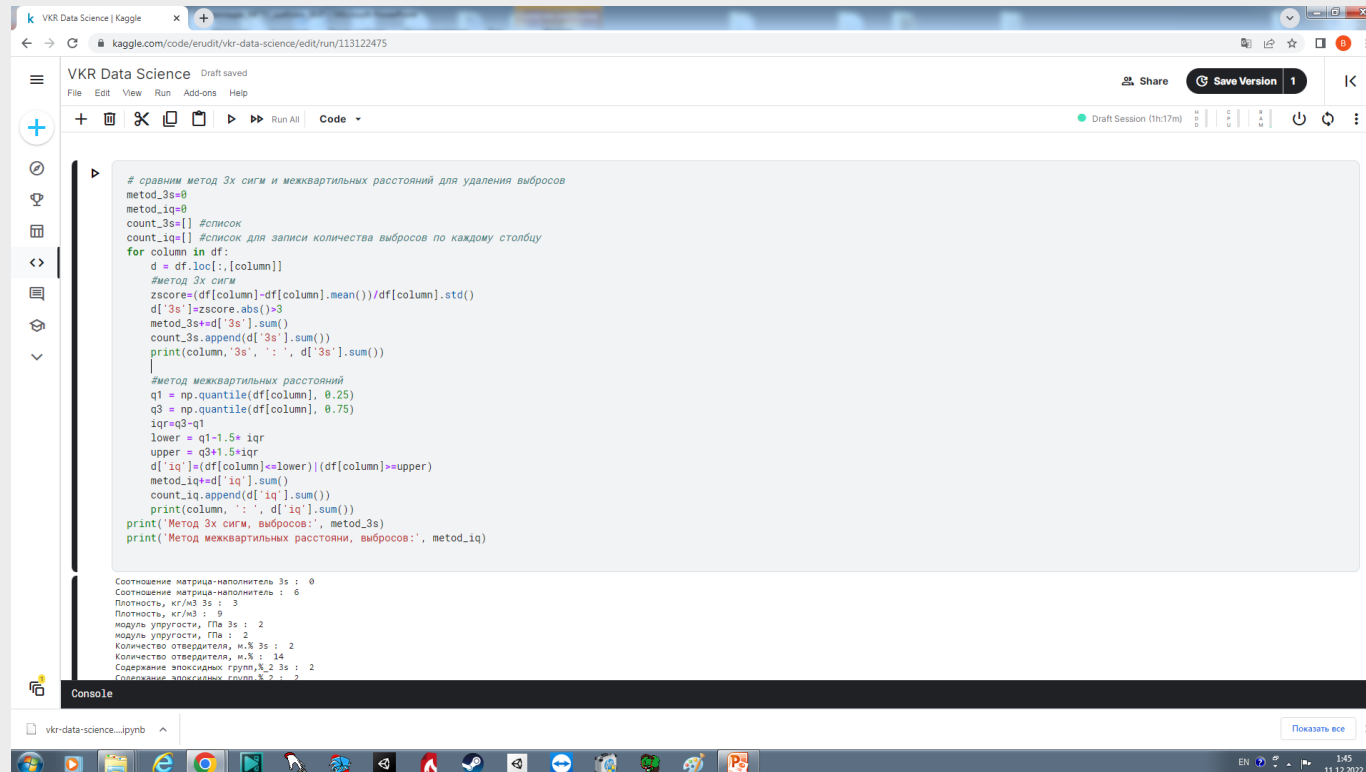
```
#визуализация корреляционной матрицы с помощью тепловой карты
mask=np.triu(df.corr())
plt.figure(figsize=(20,20))
plt.suptitle("Ящик с усами", y=8.9, fontsize=30)
plt.boxplot(pd.DataFrame(scaler.transform(df)), labels=df.columns,
             patch_artist=True, meanline=True, vert=False, boxprops=dict(facecolor='b', color='y'),
             medianprops=dict(color='r'), whiskerprops=dict(color='b'),
             capprops=dict(color='black'), flierprops=dict(color='g', markeredgecolor='r'))
plt.show()
```

The notebook displays a heatmap titled "Ящик с усами" showing the correlation matrix of various features. The features are: "Плотность матрицы", "Модуль упругости", "Количество отвердителя", "Содержание эпоксидных групп", "Температура вспенивания", "Поверхностная плотность", "Модуль упругости при растяжении", and "Плотность при растяжении".

Этапы работы

9. Предобработка данных

- проверка на наличие неинформативных признаков, визуализация
- проверка на выбросы при помощи IQR
- удаление выбросов методом сигм и межквартильных расстояний



```
# сравним метод 3х сигм и межквартильных расстояний для удаления выбросов
metod_3s=0
metod_iq=0
count_3s=[] #список
count_iq=[] #список для записи количества выбросов по каждому столбцу
for column in df:
    d = df.loc[:, [column]]
    #метод 3х сигм
    zscore=(df[column]-df[column].mean())/df[column].std()
    d['3s']=zscore.abs()>3
    metod_3s+=d['3s'].sum()
    count_3s.append(d['3s'].sum())
    print(column, '3s', ': ', d['3s'].sum())
    #метод межквартильных расстояний
    q1 = np.quantile(df[column], 0.25)
    q3 = np.quantile(df[column], 0.75)
    iqr=q3-q1
    lower = q1-1.5*iqr
    upper = q3+1.5*iqr
    d['iq']=(df[column]<=lower)|(df[column]>=upper)
    metod_iq+=d['iq'].sum()
    count_iq.append(d['iq'].sum())
    print(column, ': ', d['iq'].sum())
print('Метод 3х сигм, выбросов:', metod_3s)
print('Метод межквартильных расстояний, выбросов:', metod_iq)
```

Соотношение матрица-наполнитель 3s : 0
Соотношение матрица-наполнитель : 6
Плотность, кг/м3 3s : 3
Плотность, кг/м3 : 9
модуль упругости, ГПа 3s : 2
модуль упругости, ГПа : 2
Количество отвердителя, м.л 3s : 2
Количество отвердителя, м.л : 14
Содержание эпоксидной группы, % 3s : 2
Содержание эпоксидной группы, % : 2

10. Machine Learning

- нормализация числовых признаков (StandardScaler())
- нормализация данных (Normalizer())
- тренировочная и тестовая выборка
- применение алгоритмов машинного обучения в scikit-learn
 - метод опорных векторов
 - метод линейной регрессии

11. Нейронная сеть по рекомендации соотношения матрица-наполнитель

12. Репозиторий на github.com

<https://github.com/tinysteelybird/VKR>



The screenshot shows a Windows taskbar at the bottom with icons for Windows, Edge, Chrome, and several other applications. Above the taskbar is a console window with a dark background. The console displays the command 'vkr-data-science...ipynb' and a button labeled 'Показать все' (Show all).

VKR Data Science | Kaggle

Кросс-энтропия (Cross-Entropy)

kaggle.com/code/erudit/vkr-data-science/edit/run/113122475

Share

Save Version 1

File Edit View Run Add-ons Help

+ Code

+ Markdown

Draft Session (3h:44m)

DD

CC

PP

MM

[80]:

#нейронная сеть по рекомендации соотношения матрица-наполнитель

smn=df['Соотношение матрица-наполнитель']

smn_1 = df.loc[:, df.columns!='Соотношение матрица-наполнитель']

X_train, X_test, Y_train, Y_test=train_test_split(smn_1, smn, test_size=0.3, random_state=15)

[81]:

#нормализация данных

X_train_norm=tf.keras.layers.Normalization(axis=-1)

X_train_norm.adapt(np.array(X_train))

2022-12-11 01:12:44.029462: I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with default inter op setting: 2. Tune using inter_op_parallelism_threads for best performance.

2022-12-11 01:12:44.087532: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:185] None of the MLIR Optimization Passes are enabled (registered 2)

[82]:

#построение модели

model= keras.Sequential([X_train_norm, layers.Flatten (input_shape=(28,28,1)),

layers.Dense(128, activation='relu'),

layers.Dense(128, activation='relu'),

layers.Dense(128, activation='relu'),

layers.Dense(10, activation='softmax')

])

model.summary()

Model: "sequential"

Layer (type)	Output Shape	Param #
normalization (Normalization)	(None, 12)	25
flatten (Flatten)	(None, 12)	0
dense (Dense)	(None, 128)	1664
dense_1 (Dense)	(None, 128)	16512

Console

vkr-data-science....ipynb

Показать все





edu.bmstu.ru

+7 495 182-83-85

edu@bmstu.ru

Москва, Госпитальный переулок ,
д. 4-6, с.3