ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
по курсу
«Data Science»

Слушатель: Алексеева Анна Александровна

Постановка задачи:

- **Цель решения задачи**: прогнозировать характеристики композиционного материала на основе имеющихся данных.
- **Входные данные**:
- - общее описание свойств композиционного материала
- - два датасета, которые содержат данные о количественных характеристиках различных свойств и составляющих композитного материала. Всего 13 характеристик.
- - постановка задач для решения с помощью методов машинного обучения:
  - решение задачи регрессии для прогнозирования двух из 13 представленных характеристик
  - разработка рекомендательной системы (задача регрессии) для прогнозирования показателя «Соотношение матрица-наполнитель»

1 Этап. Изучение и описание датасета

- **Входные переменные:**
- Соотношение матрица-наполнитель
- Плотность, кг/м3
- Модуль упругости, Гпа
- Количество отвердителя, м
- Содержание эпоксидных групп,%_2
- Температура вспышки, С_2
- Поверхностная плотность, г/м2
- Модуль упругости при растяжении, Гпа
- Прочность при растяжении, Мпа
- Потребление смолы, г/м2
- Угол нашивки, град
- Шаг нашивки
- Плотность

- **Выходные переменные (исключаются в момент решения задачи из входных):**
- **Задача регрессии 1:**
- Модуль упругости при растяжении, Гпа

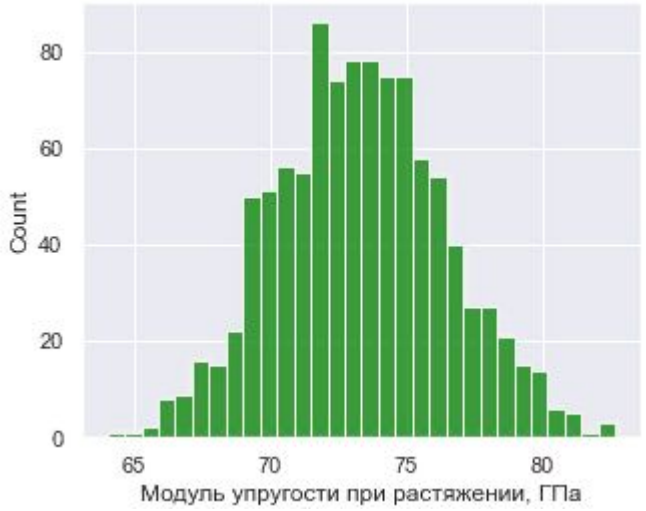
- **Задача регрессии 2:**
- Прочность при растяжении, Мпа

- **Разработка рекомендательной системы:**
- Соотношение матрица-наполнитель

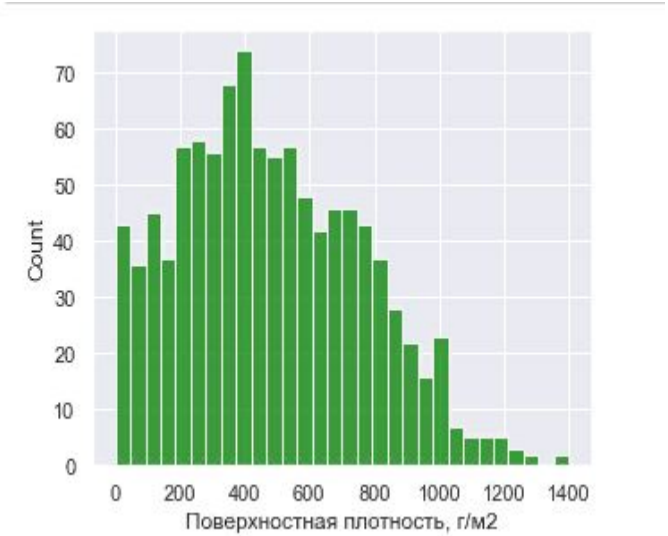
- **Первый шаг в обработке данных:**
- Объединение датасетов по индексу с отсечением последних 17 строк второго датасета

2 Этап. Разведочный анализ данных

- Использованы методы описательной статистики.
- Метод describe(). Выявлена одна дискретная величина, отсутствие пропусков в данных.



Нормальное распределение



Распределение со смещением вправо

2 Этап. Разведочный анализ данных

- Поиск выбросов и правило трех сигм

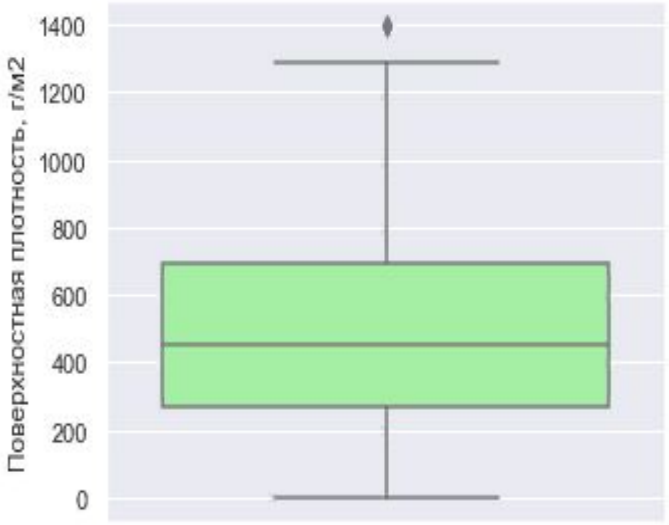


Диаграмма «Ящик с усами» с наличием выбросов в стороне больших значений

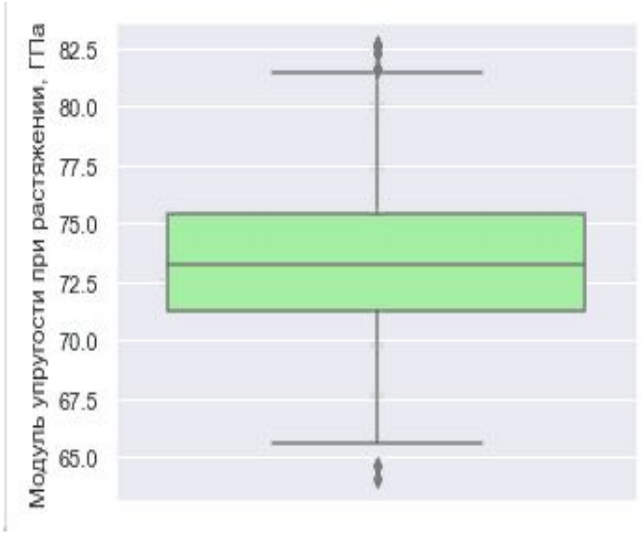
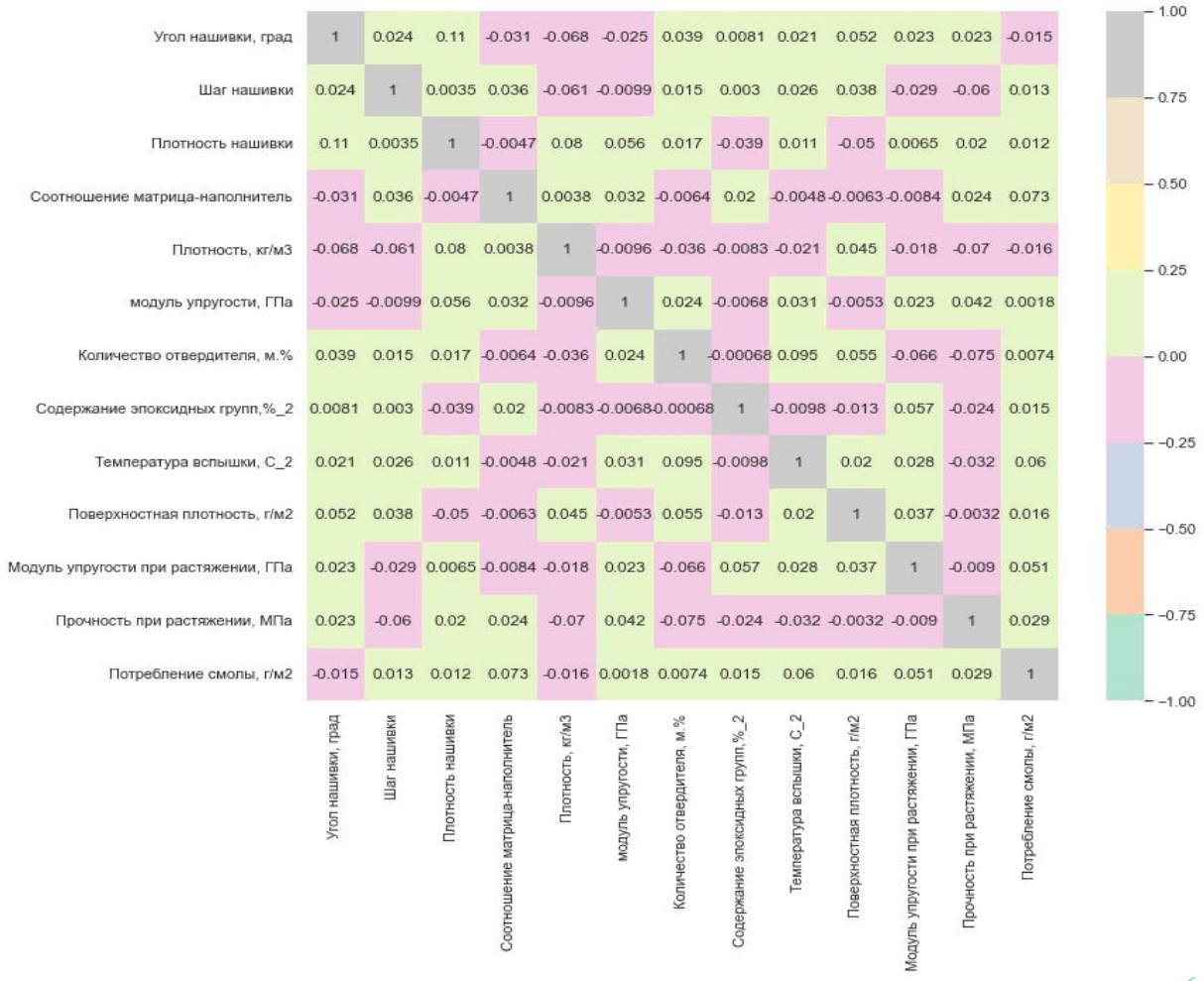


Диаграмма Ящик с усами с наличием выбросов с двух сторон.

2 Этап. Разведочный анализ данных

Тепловая карта коэффициентов корреляции



3. Этап. Предобработка данных

- 1. Расчет количества выбросов и удаление выбросов
- 2. Нормализация и стандартизация данных
- 3. Выявление внутренних невидимых факторов, которые будут влиять на модель с помощью метода главных компонент и факторного анализа
- Пример факторного анализа на 4 фактора:

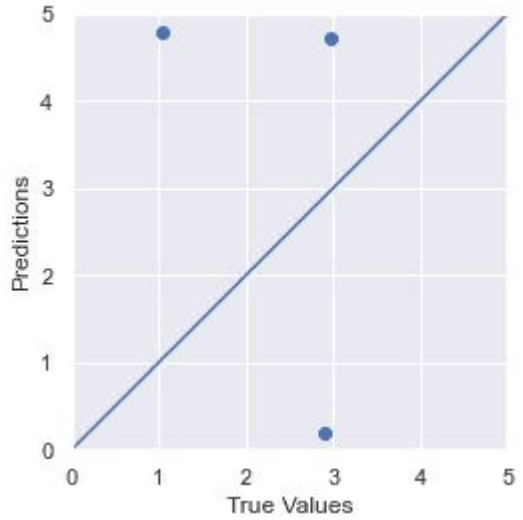
	Угол нашивки, град	Шаг нашивки	Плотность нашивки	Соотношение матрица-наполнитель	Плотность, кг/м3	модуль упругости, ГПа	Количество отвердителя, м.%	Содержание эпоксидных групп,%_2	Температура вспышки, С_2	Поверхностная плотность, г/м2	Модуль упругости при растяжении, ГПа	Прочность при растяжении, МПа	Потребление смолы, г/м2
0	-0.120244	-0.048363	0.220935	0.029813	0.581996	0.031755	-0.146221	-0.028341	-0.059310	0.040166	-0.019645	-0.082034	-0.01154
1	-0.469949	-0.038434	-0.393359	-0.019148	0.016406	-0.071638	-0.108426	-0.002496	-0.024403	-0.022026	-0.047283	-0.026570	-0.00123
2	0.057936	0.009598	0.055925	-0.007254	-0.097085	-0.041777	-0.484338	0.000065	-0.120405	-0.122759	0.082003	0.160390	0.01010
3	0.256765	0.001198	-0.268709	-0.151428	0.120485	-0.184513	-0.058783	0.063913	-0.040257	0.165013	0.034009	-0.096676	-0.06613

4 Этап. Решение задачи регрессии

- **Разделение выборки на обучающую и тестовую:**
- X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1)

- **Линейная регрессия:**

- model_LN_1 = LinearRegression()
- model_LN_1.fit(X_train, y_train)
- y_pred = model_LN_1.predict(X_test)



4 Этап. Решение задачи регрессии

- Случайный лес:

```
random_forest_tuning = RandomForestRegressor(random_state = 42)
param_grid = {
    'n_estimators': [20, 40, 60],
    'max_features': ['auto', 'sqrt', 'log2'],
    'max_depth' : [3,4,5,6]
}
GSCV = GridSearchCV(estimator=random_forest_tuning, param_grid=param_grid,
cv=10, verbose=0)
GSCV.fit(X_train, y_train)
GSCV.best_params_
```

5 Этап. Оценка качества моделей для задачи регрессии

Средняя абсолютная ошибка:

$$MAE = \frac{\sum_{i=1}^{N} |y_i - \hat{y}_i|}{N}$$

Коэффициент детерминации:

$$R^2 = 1 - \frac{D[y|x]}{D[y]} = 1 - \frac{\sigma^2}{\sigma_y^2},$$

	наименование модели	mean_absolute_error	r2_score
0	Линейная регрессия_МУ	0.148908	-37.247571
1	Случайный лес_МУ	0.149932	-47.374302
2	Линейная регрессия_ПР	0.150917	-32.786002
3	Случайный лес_ПР	0.149897	-107.726809

Этап 6. Решение задачи по разработке рекомендательной модели с использованием нейронных сетей

- Многослойный персептрон:

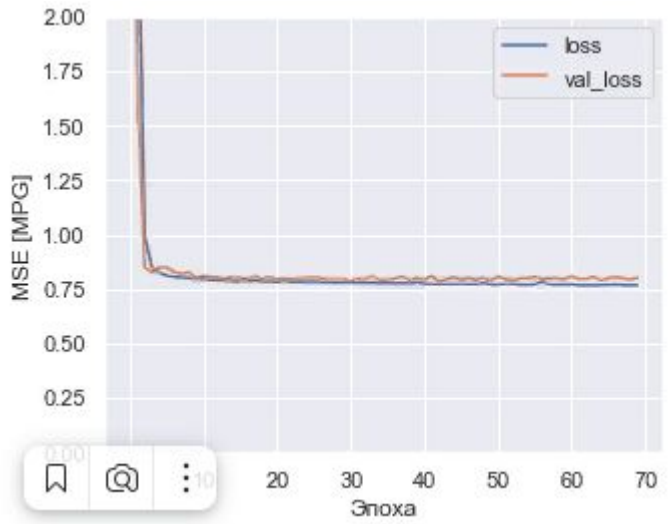
```
def build_and_compile_model(norm):
    model = keras.Sequential([
      norm,
      layers.Dense(256, activation='relu'),
      layers.Dense(256, activation='relu'),
      layers.Dense(64, activation='linear'),
      layers.Dense(1)
    ])

    model.compile(loss='mean_squared_error',
                  optimizer=tf.keras.optimizers.Adam(0.0001))
    return model
```

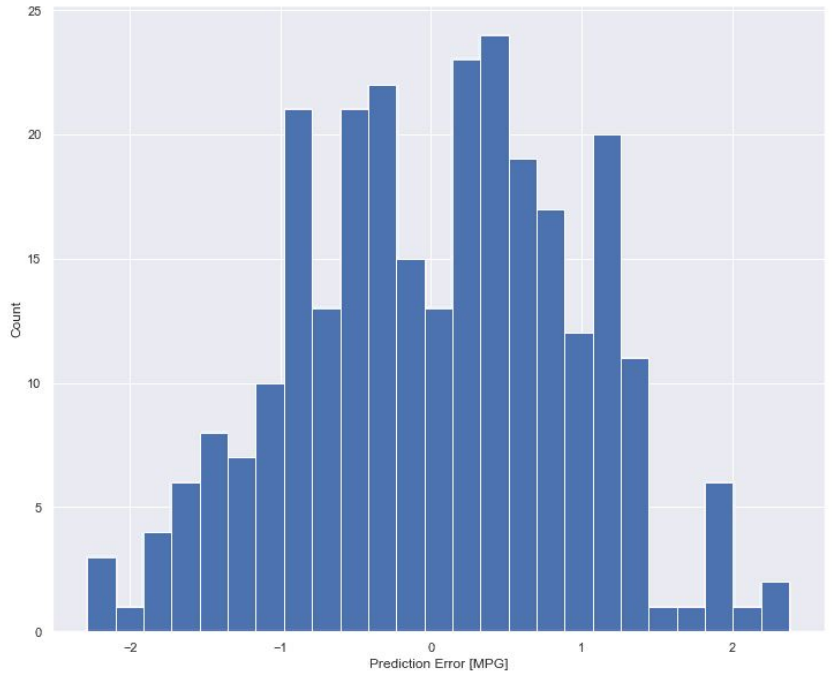
- Гиперпараметры модели:
- - количество скрытых слоев
- - количество нейронов на слое
- - активационная функция
- - количество нейронов на выходном слое
- - оптимизатор
- - метрика оценки качества
- Так же задается количество эпох

Этап 7. Оценка качества модели

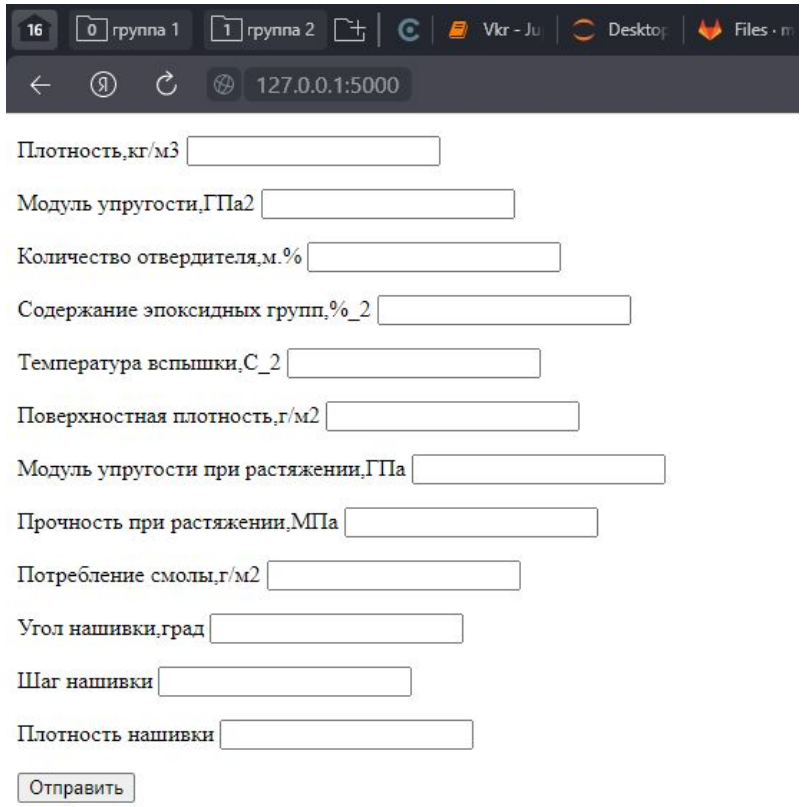
Изменение MSE за время обучения модели



Распределение ошибки (test predictions – y test)



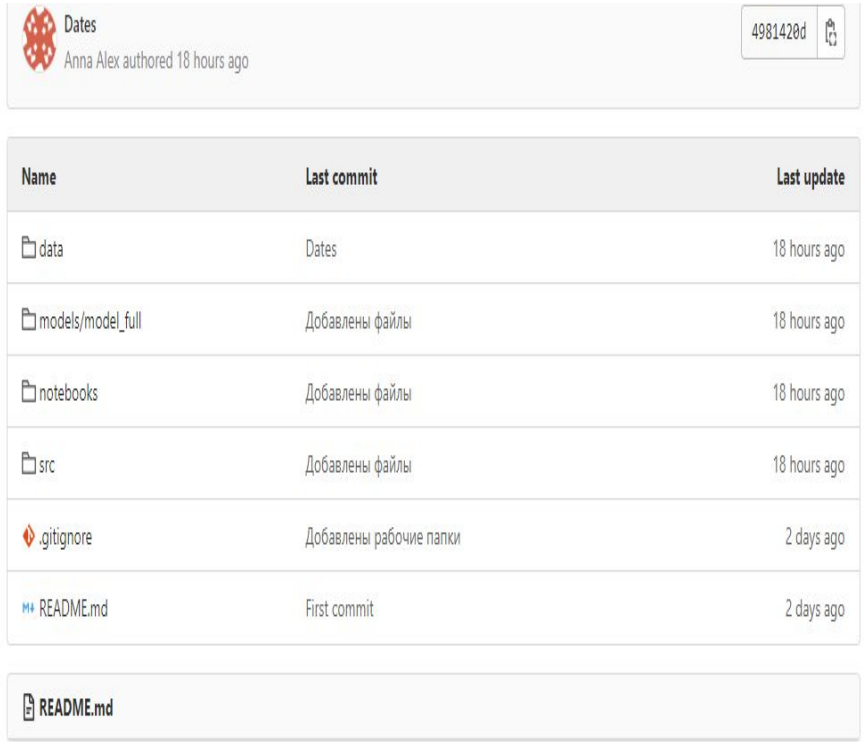
Этап 8. Разработка приложения для рекомендательной системы. Интерпретатор Flask



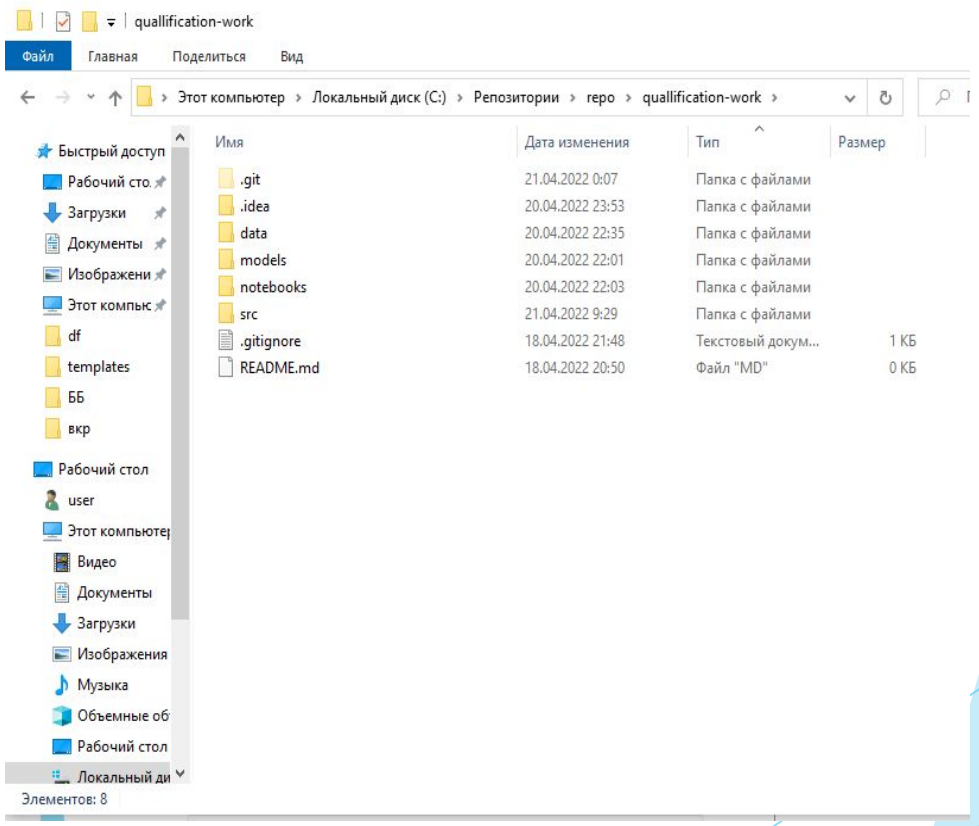
- http://127.0.0.1:5000/start1

Этап 9. Создание репозитория. Выгрузка через Git

- Репозиторий на GitLab



- Репозиторий на рабочем компьютере



Спасибо за внимание!