

## 1. Анализ данных

Описание набора данных:

Датасет - <https://www.kaggle.com/competitions/playground-series-s4e10/data>

Таргет: loan\_status – Статус кредита (0 – нет дефолта, 1 – дефолт)

person\_age – Возраст

person\_income – Годовой доход

person\_home\_ownership – Владение жильем

person\_emp\_length – Стаж работы (в годах)

loan\_intent – Цель кредита

loan\_grade – Кредитный рейтинг

loan\_amnt – Сумма кредита

loan\_int\_rate – Процентная ставка

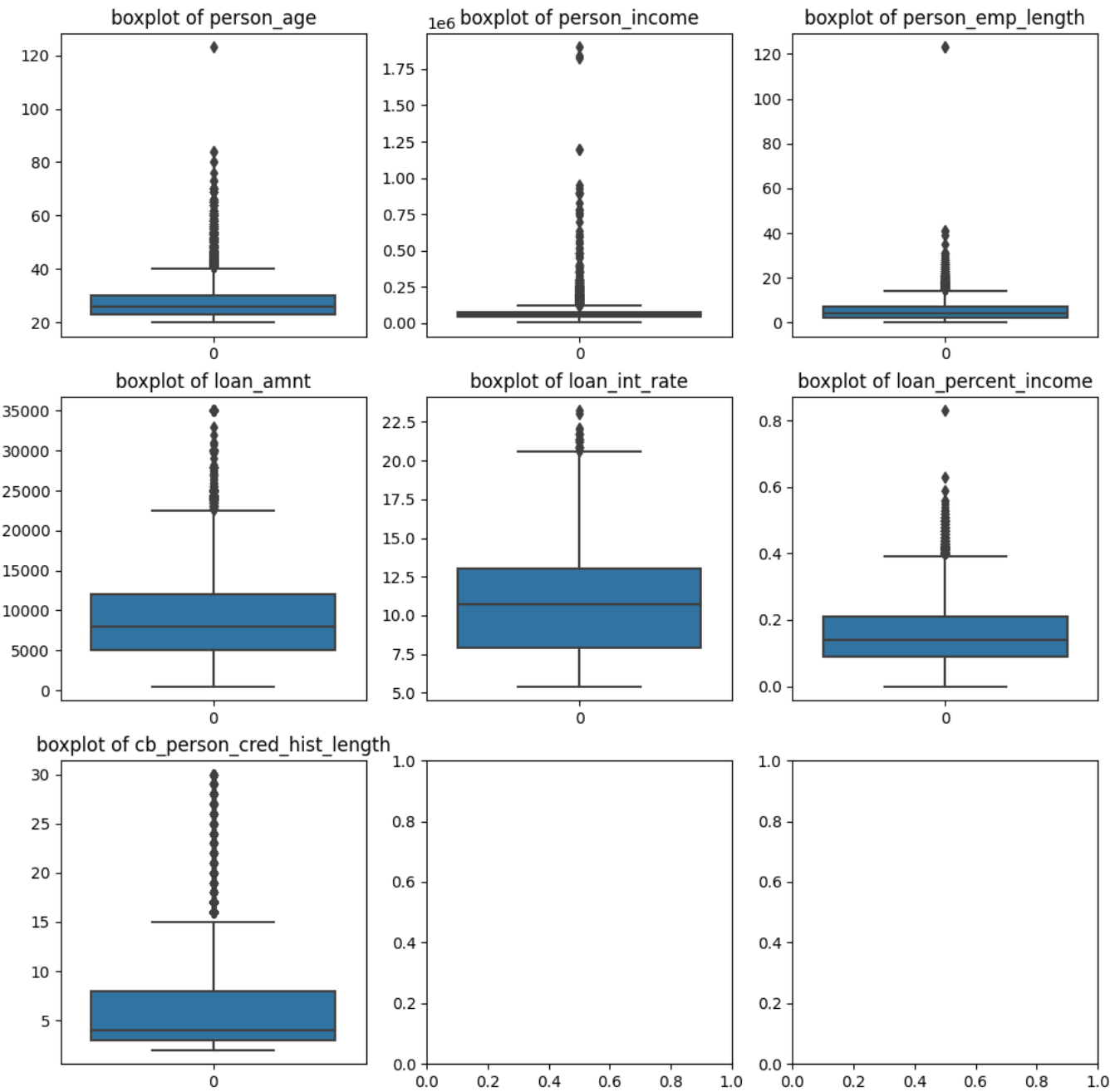
loan\_percent\_income – Доля дохода, выделяемая на кредит

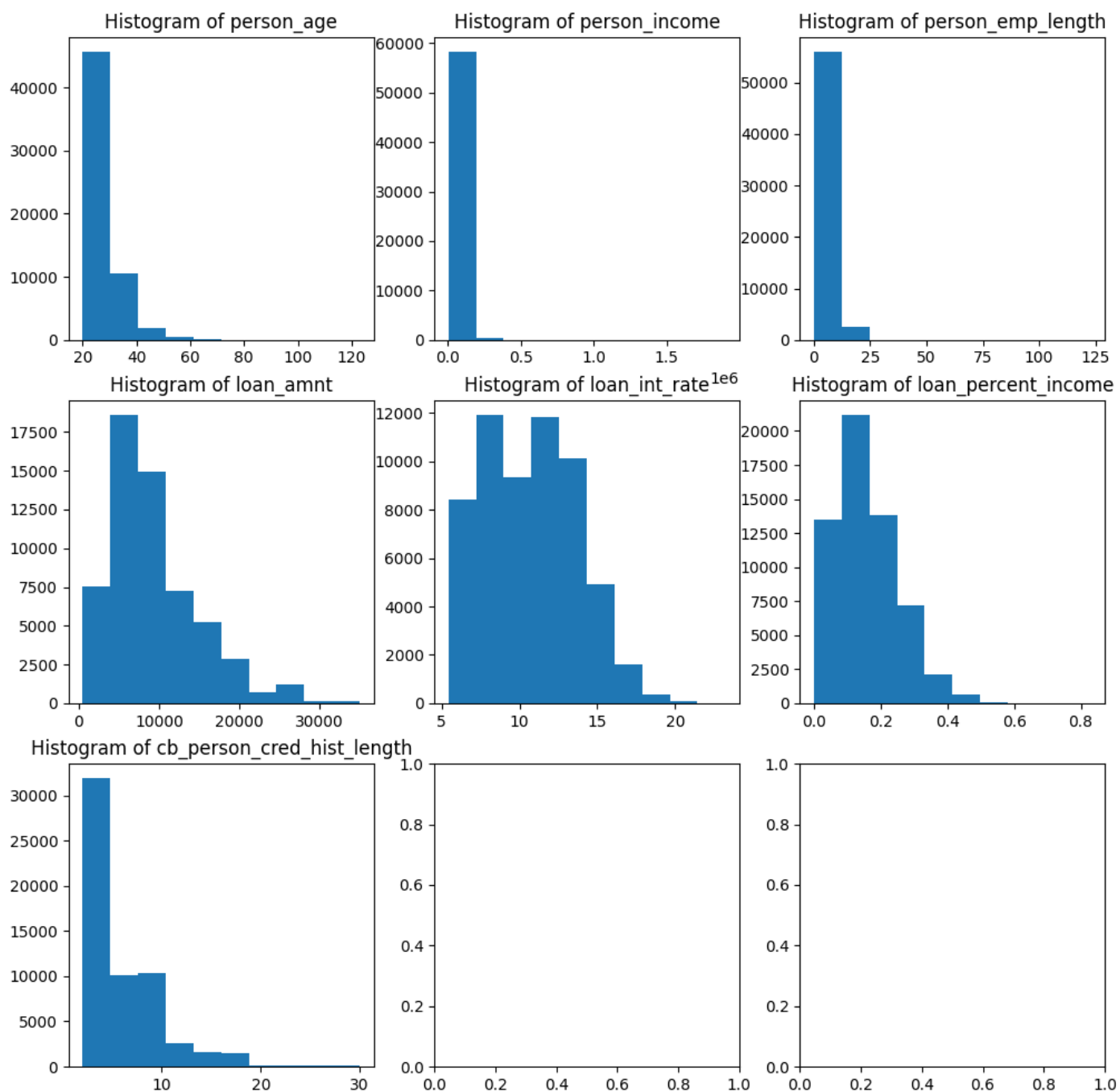
cb\_person\_default\_on\_file – Наличие дефолта в истории

cb\_preson\_cred\_hist\_length – Длина кредитной истории

Размер: 58645 \* 12

Визуализация распределений:





Проверка пропусков и выбросов:

```
df.isna().sum()
```

```
: id                0
  person_age        0
  person_income      0
  person_home_ownership 0
  person_emp_length  0
  loan_intent         0
  loan_grade         0
  loan_amnt          0
  loan_int_rate      0
  loan_percent_income 0
  cb_person_default_on_file 0
  cb_person_cred_hist_length 0
  loan_status        0
  dtype: int64
```

Удаление аномальных значений:

```
: data = data.drop(data[data['person_age'] > 100].index, axis=0)
  df = df.drop(df[df['person_age'] > 100].index, axis=0)
```

```
: data = data.drop(data[data['person_emp_length'] > 80].index, axis=0)
  df = df.drop(df[df['person_emp_length'] > 80].index, axis=0)
```

## 2. Выбор модели

- Описание используемой архитектуры нейронной сети (например, MLP, CNN):

MLP - многослойный перцептрон. Простая модель нейронной сети, подходящая для задач регрессии и классификации. Состоит из входного слоя и в частности полносвязных скрытых слоёв.

- Обоснование выбора активационных функций, оптимизатора и loss-функции.

Loss-функция - кросс-энтропия с логитами. Оптимизирована для вычислений на GPU, численно стабильна (благодаря сигмоидам в формуле)

Активационная функция скрытых слоёв: Relu - простая и самая распространенная функция активации для простых моделей нейронных сетей, при этом довольно эффективная.

Активационная функция на выходе нейросети: сигмоида - для получения вероятностей из выхода нейросети

Оптимизатор: Adam - адаптивный моментум. Позволяет избежать колебаний при стохастическом градиентном спуске благодаря накоплению градиентов с прошлых шагов, а также эффективно справляется с преодолением локальных минимумов и седловых точек.

- Гиперпараметры (число слоев, размер батча, эпох).

Число слоёв = 4

Размер батча = 32

Количество эпох обучения = 5

Следующие гиперпараметры подобраны на валидационной выборке с применением Optuna:

Количество нейронов в скрытом слое: 199

Вероятность для Дропаута: 0.3264149947599042

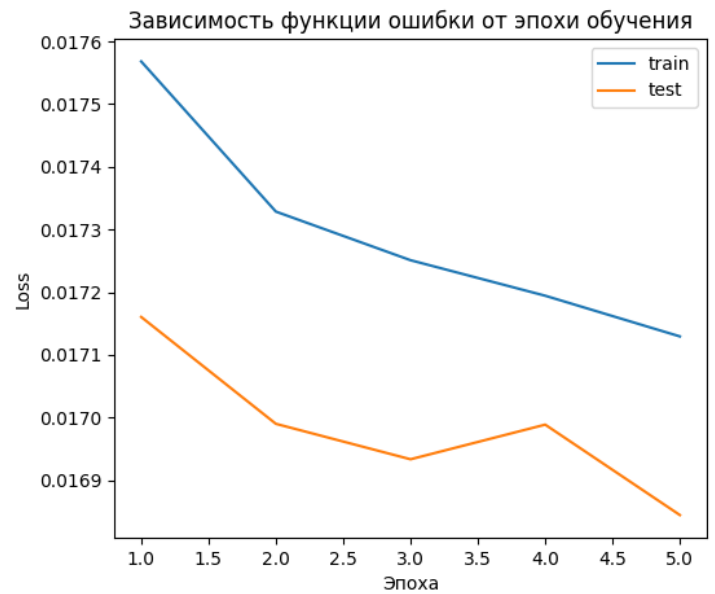
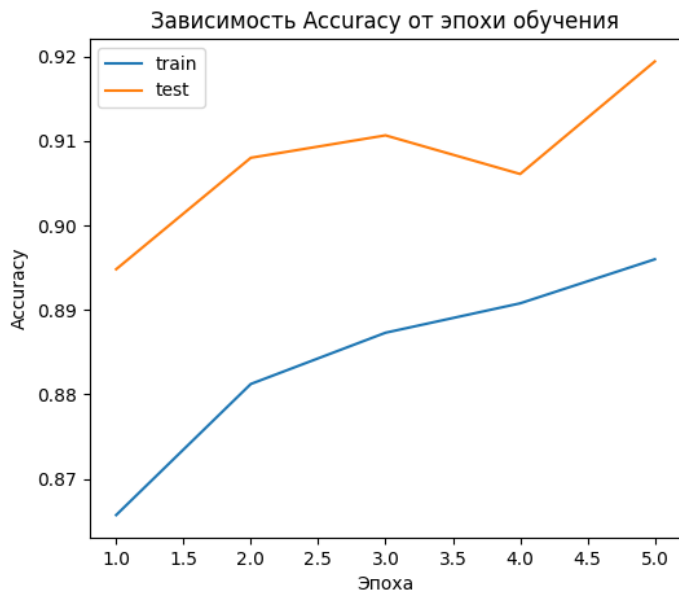
Скорость обучения для Adam: 0.0021396619124599905

### 3. Результаты:


- Таблица с метриками вычисленными на тестовой выборке

	Метрика	Значение
0	Loss	0.0168
1	Accuracy	0.9194
2	AUC-ROC	0.9192
3	F1-score	0.9153

- График обучения (loss и accuracy на train/val) на отдельном рисунке.



- Результат после отправки submission:


**submission (11).csv**

Complete (after deadline) · 1h ago

**0.88831**

**0.89279**

☐