Dano

Пока только первую часть написал, остальное допишу. Вопросы лучше в общий чат писать, как сделаете первую часть кидайте ссылку на colab с кодом.

Часть 1: Общий анализ данных

Первую часть нужно выполнить всем, даю небольшие подсказки что-бы начать

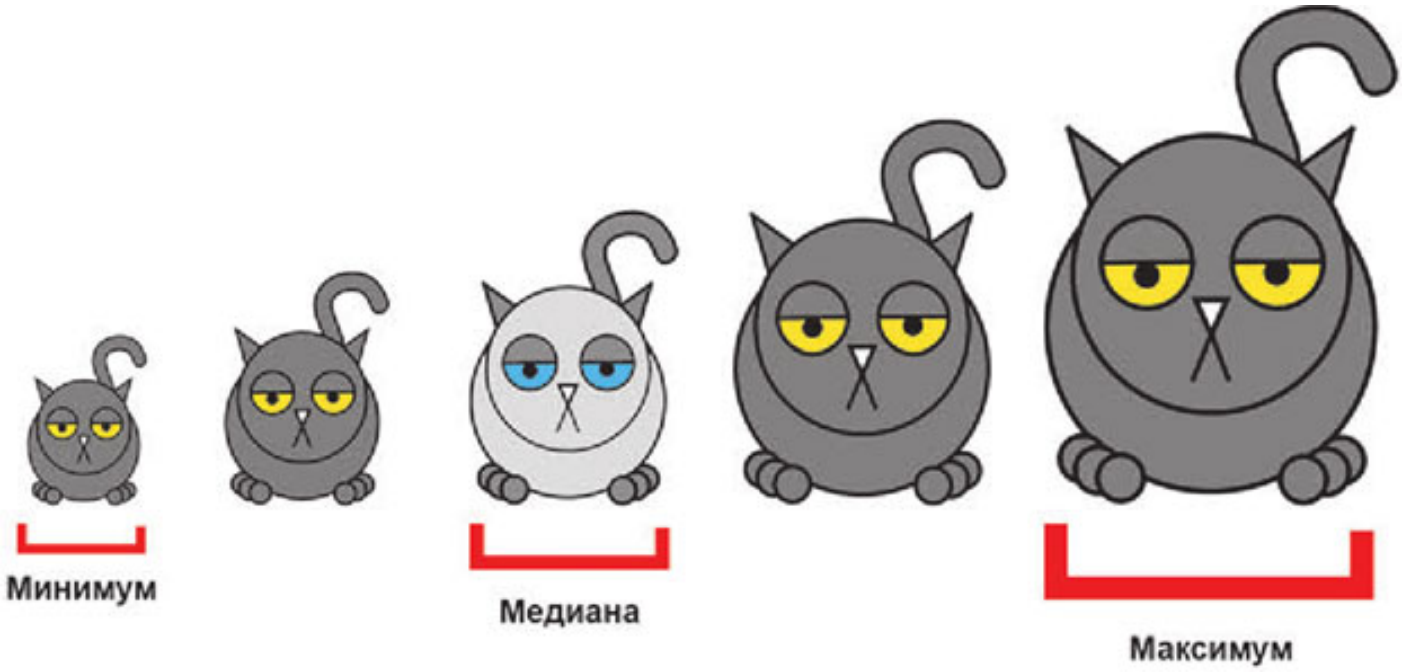
- По каждому пункту нужно написать вывод, можно лаконично
- Графики должны быть красиво визуализированы, должны быть подписаны оси, функции и значения цветов
- Ваш код должен запускаться в colab, проверьте перед отправкой

Анализ и обработка пропусков и выбросов

Если в ходе работы добавляете новые фичи, по ним тоже нужно сделать пункты анализ.

- Id-фичи можно не визуализировать и практически не проводить по ним анализа, но немного проверить все-же стоит, тут решайте сами
- Может быть фича у которой только одно значение, по ней тоже особо ничего делать не нужно, но сказать про это стоит

Посмотреть пропуски по данным




```
df.info()
```

Если есть пропуски их нужно заменить, придумайте как, на среднее например или медиану для числовых, пустая строка для `string`.

Посмотреть типы данных

Тоже используя команду `df.info()` Смотрим какого типа каждая из фичей и их реальные значения, должны совпадать типы, может быть тип `object` у числовых данных из-за пропусков. Что-бы привести к нужному типу используем

```
df['column'].astype(type)
```

Посмотреть выбросы в данных

Получить список значений и их количество по каждой фиче можно например так:

```
df['column'].value_counts()
```

Глазами находим выбросы если они есть их нужно заменить как пропуски в данных. Данные нужно визуализировать, самый простой способ:

```
df.plot.bar(x='x_label', y='y_label')
```

Но лучше попробовать использовать либу для визуализации.

Определение типа данных

Нужно понять по каждой числовой фиче к какому типу данных она относится:

- **Категориальная** - мало значений без порядка (регион)
- **Бинарная** - категориальная с двумя значениями (пол)
- **Порядковая** - категориальная упорядоченная (уровень образования)
- **Количественная** - много значений, числовая (возраст)
- **Временная** - дата или время Категориальных у Вас вроде нет, но проверить стоит.

Анализ корреляция и зависимостей фичей

Корреляция Пирсона

$$\mathbf{r}_{XY} = \frac{\mathbf{cov}_{XY}}{\sigma_X \sigma_Y} = \frac{\sum(X - \bar{X})(Y - \bar{Y})}{\sqrt{\sum(X - \bar{X})^2 \sum(Y - \bar{Y})^2}}.$$

Если совсем незнакомы, немного почитать про корреляцию немного.

Что-бы получить матрицу корреляции используем:

```
df.corr()
```

Нужно красиво визуализировать эти значения, самый простой способ с помощью библиотеки `seaborn` используя функцию `heatmap`, посмотрите как она работает.

Корреляция Спирмена (опционально)

Аналогично сделать матрицу корреляции, вроде функция `spearmanr` в `scipy.stats`, нужно чуть внимательнее посмотреть и разобраться.

Добавление фичей

Стоит придумать какие фичи можно добавить.

Сумма всей корзины

Это просто пример, вроде у Вас `Good_cnt = 1` всегда, но если не так, то добавляем:

```
df['total_price'] = df['Good_cnt'] * df['Good_price']
```

Цена товара к доходу клиента

Вроде уже интереснее:

```
df['price_per_income'] = df['Good_price'] /
df['Monthly_income_amt']
```

Другие фичи

Тут нужно проявить немного креативности, мб что-то интересное можно сделать из признаков. Как минимум нужно проверить что `Education_level` это порядковая фича над `['SCH', 'GRD', 'UGR', 'PGR', 'ACD']`.

Визуализация пар фичей

Каждая фича должна быть визуализирована хоть в одной паре. Посмотрите примеры с seaborn, можно выбрать какая визуализация вам больше нравиться.

Визуализация двух числовых фичей

Например можно использовать relplot, придумайте какие фичи интереснее визуализировать.

Визуализация категориальной и числовой фичи

Если не знаете с чего начать, можно выбрать bar, тут из библиотеки `matplotlib`, если хотите `seaborn` нужно самим выбрать.

Остальные пары

Тут нужно самим посмотреть как визуализировать, например если есть временная фича удобно строить графики lineplot, или из `matplotlib` что-то взять

Часть 2: Углубленный анализ и визуализация

Тут каждому будет выбрано личное задание, после выполнения первого пункта, чем раньше сдадите работу тем больше будет выбор что поделать.

Часть 3: Анализ гипотез

Тут каждый выберет одну из гипотез и будет по ней работать, работаем в парах.

Часть 4: Итоговый анализ

Тут будут смешиваться результаты разных гипотез и будем получать какие-то выводы, работаем вместе.

Полезные ссылки

1. Google Colab - можно тут запускать свой код, если не хотите локально
2. Seaborn- документация библиотеки seaborn
3. Matplotlib- документация библиотеки matplotlib
4. Pandas- документация библиотеки pandas
5. PEP 8- стараемся придерживаться такого стиля кода

Успехов!