**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**

**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**

**(РУТ (МИИТ)**

Институт/факультет «Академия “Высшая Инженерная Школа”, АВИШ»

Специальность/Направление подготовки Информатика и вычислительная техника

**КУРСОВАЯ РАБОТА**

**по дисциплине:** Технологии хранения больших данных

|  |  |
| --- | --- |
| **на тему:** | “Анализ применимости методов обработки данных чековых транзакций” |

|  |  |
| --- | --- |
| **Студент группы ШАД 211** | **Леманский К.Ю.** |
| **Научный руководитель** | **Зуев И.А.** |

Оглавление

[Основные термины / сокращения 2](#_Toc123093829)

[Введение 3](#_Toc123093830)

[Анализ данных 4](#_Toc123093831)

[Объект исследования 4](#_Toc123093832)

[Инструментарий исследования 5](#_Toc123093833)

[Процедура исследования 5](#_Toc123093834)

[Выводы по работе 11](#_Toc123093835)

[Заключение 11](#_Toc123093836)

[Список использованной литературы 13](#_Toc123093837)

[Приложения 13](#_Toc123093838)

# Основные термины / сокращения

Классификация – крупнейший вид задачи машинного обучения с учителем, заключающаяся в разделении выборки на классы на основе известных признаков.

Датасет – набор данных, используемый для анализа и машинного обучения.

Профайлинг – Профилирование данных для того или иного датасета.

Data-engineering – совокупность практик и средств обработки данных для решения задач машинного обучения.

K-Means – алгоритм кластеризации основанный на нахождении центроидов.

DBSCAN - алгоритм кластеризации основанный на растении до ближайшей точки.

EDA – разведочный анализ (этап работы с датасетом).

Метод “локтя” (elbow method) – популярный метод подбора количества кластеров для алгоритма K-Means.

# Введение

Сегодня пользователь хочет получать информацию как можно быстрее и наилучшего качества. В следствие происходит бурное развитее поисковиков, а так же рекомендательных систем. Так, всемирно популярная компания Netflix заняла свою позицию благодаря сложной рекомендательной системе выстроенной так, что бы пользователь находил именно те сериалы, которые нравятся именно ему.

Рекомендательные системы — это сложный ансамбль алгоритмов машинного обучения. На разработку таких ансамблей тратится много сил, но на сегодняшний день даже маленький онлайн магазин должен иметь так ую систему. В этой работе мы попробуем создать простую рекомендательную систему основанную на кластеризации клиентов на основании их товаров.

**Основная часть**

Начнем с EDA: Датасет содержит информацию о транзакциях, а конкретно дата, номер транзакции, ID покупателя, ID товара, кол-во товара, сумма покупки.

Есть несколько пропусков в датасете. На основе анализа пропусков можно сделать вывод, что это единичные ошибки, которые можно просто удалить.

Так же присутствуют дубликаты, но это значения вызванные тем что была совершенна покупка в одно и то же время, так что удалять их не стоит.

# Анализ данных

Анализ данных о клиентах начинается с подготовки данных и Разведочного Анализа данных. Далее, на основе полученной информации строятся модель для кластеризации клиентов на группы.

## Объект исследования

Датасет - информация о транзакциях:

Дата - время покупки

Транзакция - уникальный номер покупки

Клиент - уникальный номер клиента

Товар - уникальный номер товара

Количество - количество купленных товаров

Сумма - цена покупки этого товара

## Инструментарий исследования

1.Jupyter Notebook

2. Python 3.10.4

3. Pandas – Средство работы с набором данных.

4. Pandasql– Средство работы с pandas посредством SQL запросов.

5. Matplotlib – Средство визуализации

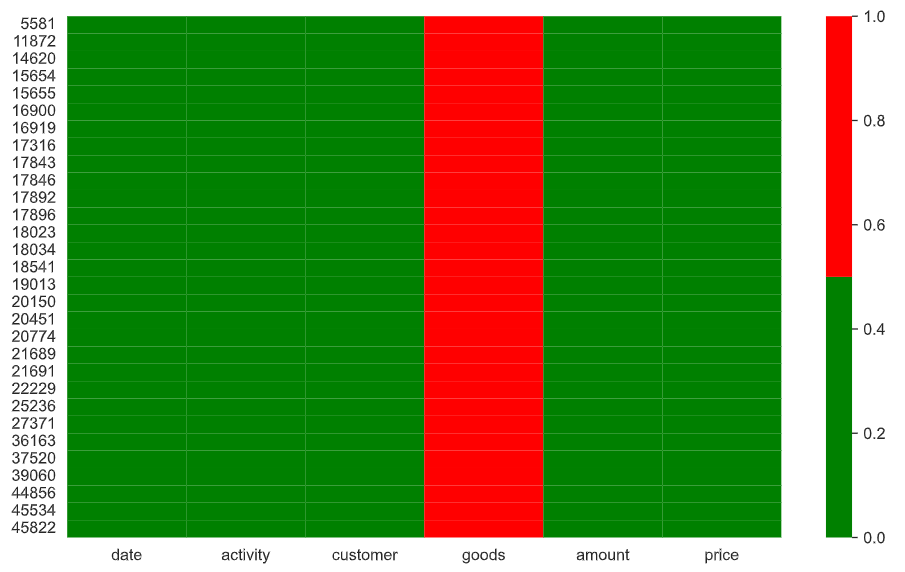
6. Seaborn. – Средство визуализации

7. Sklearn– Средство для построения моделей машинного обучения.

## Процедура исследования

Для полноценного анализа данных перед проверкой гипотез следует провести Data Engineering.

Первым делом следует проанализировать и удалить пропуски.

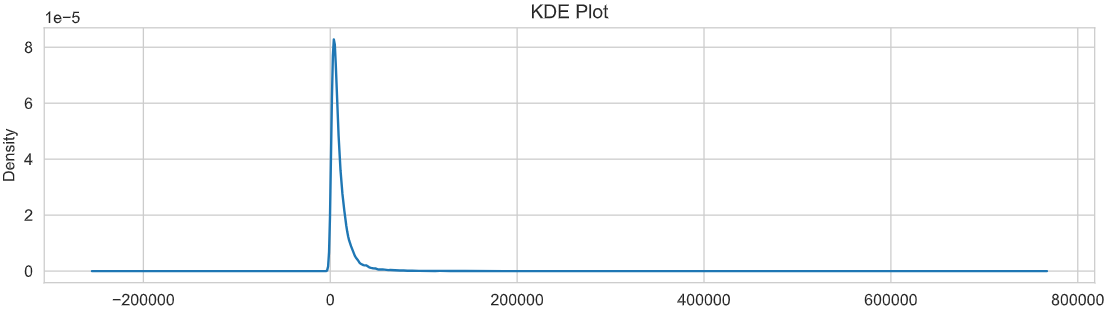


Пропусков не много и вызваны они ошибкой внесения данных, значит их действительно следует удалить.

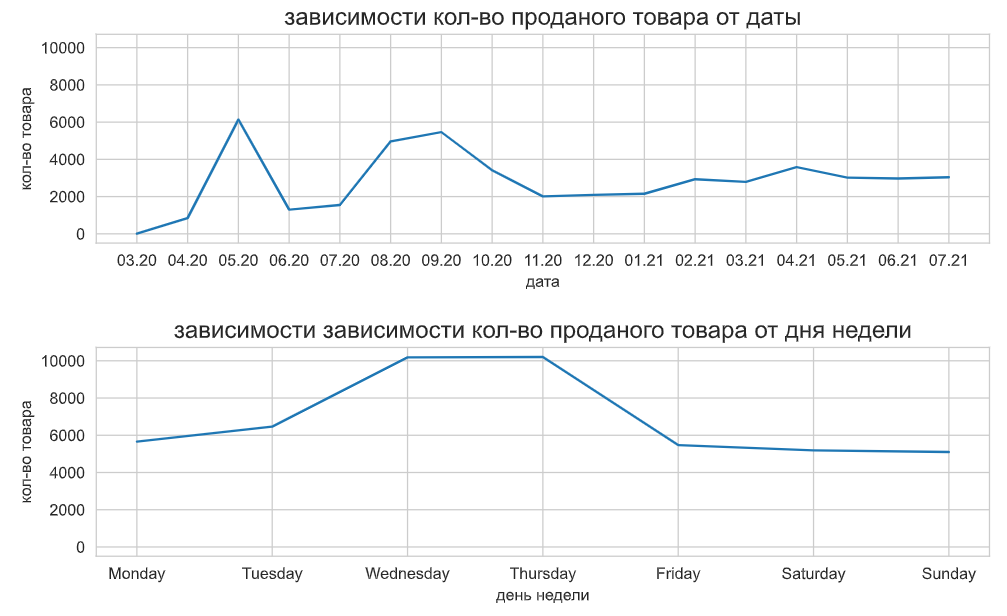
Дубликатов намного больше и их природа уже не столь очевидна, ведь покупатель мог купить один и тот же товар в один момент. Что бы подробнее разобраться в этом посмотрим на показатель amount. Оказывается в поле amount содержит только 1. Отсюда можно сделать вывод что произошло недопонимание между архитектором БД и человеком который заполнял БД. И при заполнении вместо увеличения показателя amount, запись дублировалась amount раз. Исправим ошибку, удалим дубликаты, увеличивая показатель amount.

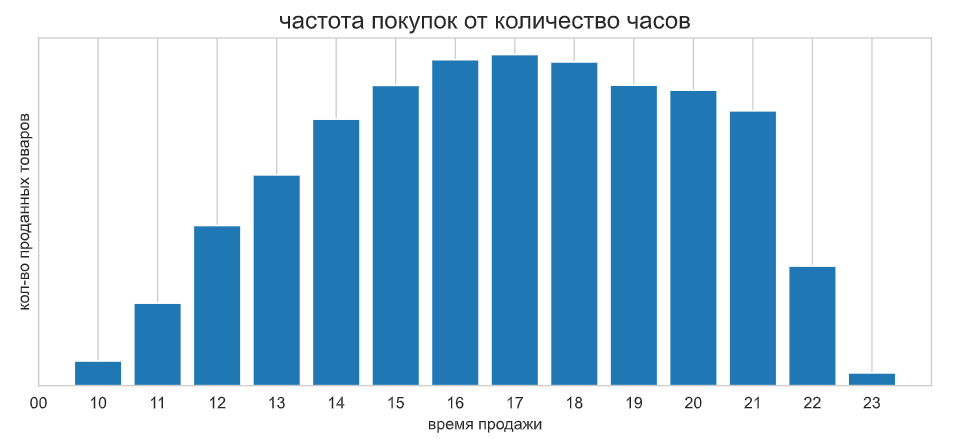
Так же следует провести преобразование типов. В нашем случае строковую дату преобразуем в datetime (тип хранения времени в python), а так же проверим действительно ли price это тип с плавающей точкой. Как оказалось тип был определен верно (price действительно float).

Далее нужно сделать профилирование данных. Можно заметить что сумма транзакции распределена логарифмически, но мы это не будем исправлять так, как при кластеризации использовать цену не будем.



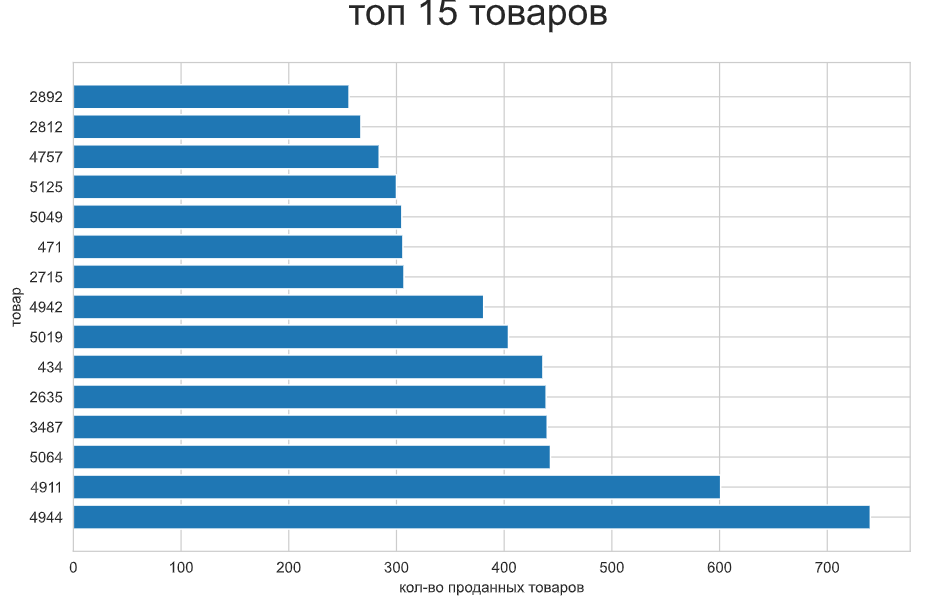
Для более глубокого понимания данных необходимо визуализировать зависимости цены от времени.



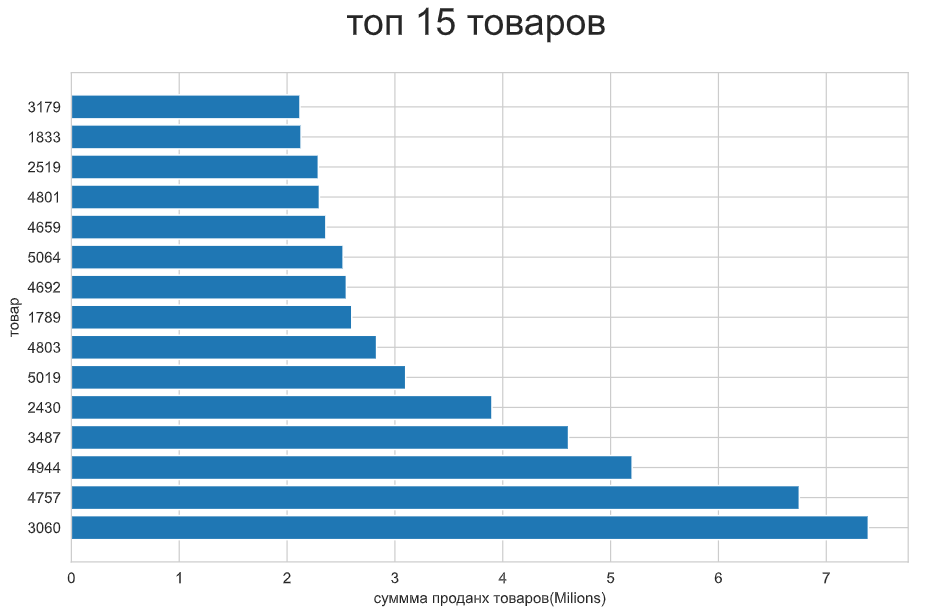


Из представленных графиков видно, что магазин получат наибольшее количество заказов в будние дни (четверг-пятница), а так же что наиболее активные месяца это май, август и сентябрь. Распределение по времени дня не более чем нормальное (любые продуктовые магазины или даже уличные палатки имеют схожее распределение). Из этих данных можно даже предположить линейку товаров, которую реализует магазин. Пик в сентябре свойственен товарам для школы, а частые покупки в четверг-пятницу со спадом в выходные, свойственны товарам для работы/учебы. Наиболее вероятны **канцелярские товары**, но это лишь предположение.

Еще полезная информация, которая может помочь понять данные это самые покупаемые товары:



Важно составлять топы товаров не только по количеству продаж в штуках, но и по сумме продаж (лучше если есть возможность по морже):

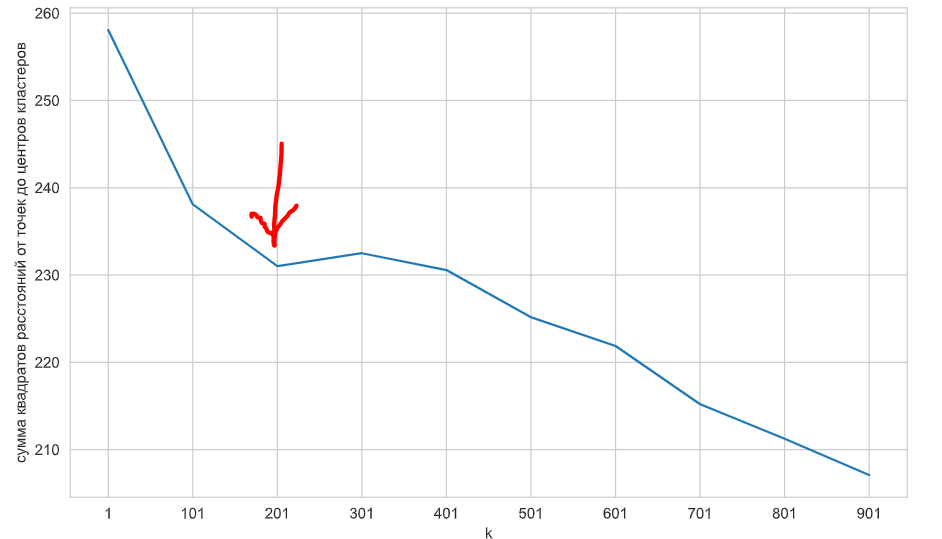


Это может помочь выстроить неперсональные рекомендации, которые могут быть продемонстрированы незарегистрированным пользователям.

Для составления ансамблей можно проанализировать отток клиентов, и на основе анализа составлять рекомендации так, чтобы удержать “хороших клиентов”. Впрочем, это тема отдельной статьи (можете почитать Семена Ануфриева).

Подготовим датасет перед тем как передавать его в модель. Создадим датсает, где каждая колонка – товар (n), каждая строка- пользователь (m), а сам датасет содержит кол-во купленного товара n пользователем m.

Искать количество кластеров будем методом “локтя”:



Построив модель K-Means будем смотреть к какому кластеру принадлежит пользователь и в зависимости от центроида будем находить самый популярный товар в группе и рекомендовать именно его. Так же можно рекомендовать товар в зависимости от того какого товара нет у пользователя, но есть в центроиде. Или более оптимальный алгоритм это из вектора покупателя вычесть вектор центроида и найти минимальное значение, очевидно это будет как раз товар по которому самые большие просадки и скорее всего пользователь захочет купить именно его.

# Выводы по работе

Мы выяснили как построить простой рекомендательный алгоритм. Конечно, его можно улучшить. Но, хоть и с не большой точностью, модель выполняет свои задачи.

Пример работы:

Изображение выглядит как текст

Автоматически созданное описание

# Заключение

Эта модель очень подвержена переобучению (можно попробовать увеличить число кластеров так что каждый пользователь будет центром своего кластера), а также она опирается на то, что выявляет группы людей так, что каждый пользователь принадлежит одной группе, когда на самом деле каждый пользователь принадлежит нескольким группам и подгруппам. Конечно, рекомендательные модели на сегодняшний день ушли далеко вперёд, но для того, чтобы создавать сложные ансамбли моделей нужно понимать, как работают самые простые модели. Эта модель хоть и слабая, но демонстрирует основы машинного обучения и рекомендательных систем.

# Список использованной литературы

1. Чечнев А.А Обзор рекомендательных систем и возможностей учета контекста при формировании индивидуальных рекомендаций
2. Михайловский Н. Анатомия рекомендательных сервисов.
3. Никита Арзамазов Как работают рекомендательные системы

# Приложения

<https://github.com/s21dlemanskiy/DataScinceKursWork>

