

# ВЛИЯНИЕ МЕТРОПОЛИТЕНА НА ЛЮБОВЬ К ШОКОЛАДУ

ИССЛЕДОВАТЕЛЬСКИЙ АНАЛИЗ ФАКТОРОВ ПОТРЕБЛЕНИЯ ШОКОЛАДА В  
СТРАНАХ С РАЗВИТОЙ СИСТЕМОЙ МЕТРОПОЛИТЕНА И БЕЗ НЕЁ



Итоговый проект по курсу обучения «DATA ANALYST»

Каштанов Павел

## Цель исследования:

Определение наличия статистически значимой связи между наличием метрополитена в стране и уровнем потребления шоколада на душу населения.

## Определение задач:

- Провести статистический анализ данных, выявить имеющиеся корреляционные связи.
- Оценить влияние экономических и демографических факторов на потребление шоколада.
- Проверить статистическую значимость выявленных закономерностей.
- Визуализировать полученные данные.

# Проработка этапов анализа данных

## 1. Определение основной задачи.

- Провести статистический анализ данных;
- Определить факторы, влияющие на формирование потребительских привычек потребления шоколада.



## 2. Источник данных.

Базы данных с сайтов Kaggle, World Bank Group, FAOSTAT

### Методы исследования:

Описательная статистика; Корреляционный анализ; Регрессионный анализ;  
Методы статистического сравнения групп



## 3. Сбор данных.

Поиск и выгрузка данных с сайтов Kaggle, World Bank Group, FAOSTAT





# Проработка этапов анализа данных

## 4. Очистка данных.

- Проверка целостности данных;
- Преобразование типов и форматов данных;
- Обработка пропущенных значений;
- Обработка выбросов.

### Инструменты:

SQL (SQLite), Python (Pandas, Missingno, Seaborn, Matplotlib, NumPy), Microsoft Excel (Power Query)

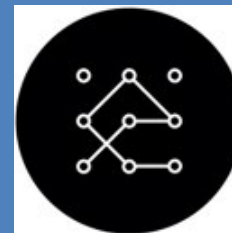


## 5. Анализ данных.

- Описательный анализ;
- Методы статистического сравнения групп (t-test, test Mann–Whitney);
- Корреляционный анализ (тепловая карта);
- Регрессионный анализ;
- Машинное обучение;
- Визуализация.

### Инструменты:

Python (Pandas, SciPy, Statsmodels, NumPy, Matplotlib, Seaborn, Xgboost)



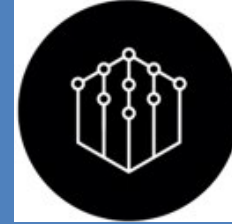
# Проработка этапов анализа данных

## 6. Результат исследования.

- Выводы;
- Интерпретация полученных данных.

### Инструменты:

Python

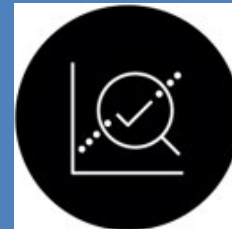


## 7. Итоговый отчёт.

- Создание Dashboard;
- Подготовка презентации.

### Инструменты:

Power BI, Microsoft PowerPoint

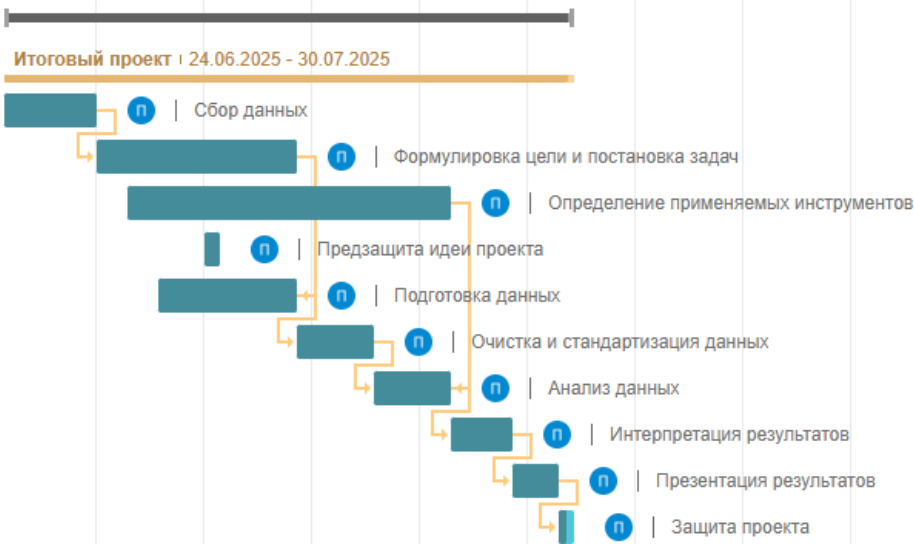


# Стадии выполнения проекта

Задачи	Дата начала	Срок выполнения	Дни	Статус
Сбор данных	24.06.2025	29.06.2025	6	Выполнено
Формулировка цели и постановка задач	30.06.2025	12.07.2025	13	Выполнено
Определение применяемых инструментов	02.07.2025	22.07.2025	21	Выполнено
Предзащита идеи проекта	07.07.2025	07.07.2025	1	Выполнено
Подготовка данных	04.07.2025	12.07.2025	9	Выполнено
Очистка и стандартизация данных	13.07.2025	17.07.2025	5	Выполнено
Анализ данных	18.07.2025	22.07.2025	5	Выполнено
Интерпретация результатов	23.07.2025	26.07.2025	4	Выполнено
Презентация результатов	27.07.2025	29.07.2025	3	Выполнено
Защита проекта	30.07.2025	30.07.2025	1	Выполняется

# Стадии выполнения проекта

Задача		Начало	Длительность	Завершение	Исполнитель	Статус	Июнь 2025		Июль 2025				Август 2025				
							16-22 (25н)	23-29 (26н)	30-6 (27н)	7-13 (28н)	14-20 (29н)	21-27 (30н)	28-3 (31н)	4-10 (32н)	11-17 (33н)	18-24 (34н)	
		24.06.2025	37д	30.07.2025				<div><div></div></div>									
1	<div><div></div>Итоговый проект</div>	24.06.2025	37д	30.07.2025				<div><div></div>Итоговый проект   24.06.2025 - 30.07.2025</div>									
1.1	Сбор данных	24.06.2025	6д	29.06.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>Выполнено</div>		<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Сбор данных</div>							
1.2	Формулировка цели и постановка задач	30.06.2025	13д	12.07.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>Выполнено</div>			<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Формулировка цели и постановка задач</div>						
1.3	Определение применяемых инструментов	02.07.2025	21д	22.07.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>Выполнено</div>			<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Определение применяемых инструментов</div>						
1.4	Предзащита идеи проекта	07.07.2025	1д	07.07.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>Выполнено</div>			<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Предзащита идеи проекта</div>						
1.5	Подготовка данных	04.07.2025	9д	12.07.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>Выполнено</div>			<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Подготовка данных</div>						
1.6	Очистка и стандартизация данных	13.07.2025	5д	17.07.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>Выполнено</div>			<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Очистка и стандартизация данных</div>					
1.7	Анализ данных	18.07.2025	5д	22.07.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>Выполнено</div>				<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Анализ данных</div>				
1.8	Интерпретация результатов	23.07.2025	4д	26.07.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>Выполнено</div>					<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Интерпретация результатов</div>			
1.9	Презентация результатов	27.07.2025	3д	29.07.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>Выполнено</div>						<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Презентация результатов</div>		
1.10	Защита проекта	30.07.2025	1д	30.07.2025	<div><div></div>Павел Каштанов</div>	<div><div></div>В работе</div>							<div><div></div></div>	<div><div></div><div><div></div></div>п</div>	<div><div></div><div><div></div></div>п</div>	<div><div></div>Защита проекта</div>	





# Описание исходных данных

Наименование столбца	Определение
Country	Название страны
Year	Год (1996-2016)
Total_weight_kg	Общий вес импортируемого и ре-импортируемого шоколада в страну (кг)
availability_of_metro	Наличие метро (да/нет)
lines	Количество линий метро в стране
stations	Количество станций метро в стране
annual_ridership_mill	Пассажиропоток (млн человек)
Population_eating_chocolate	Часть населения, которая потребляет шоколад
Cost_kg_USD	Стоимость 1 кг шоколада в стране (USD)
Urban_population	Доля городского населения страны (%)
Lifetime	Средняя продолжительность жизни в стране
GDP_USD	ВВП на душу населения (USD)
Inflation	Процент инфляции
kg_per_person	Потребление шоколада на душу населения
kg_per_GDP	Потребление шоколада на единицу ВВП (шоколадоёмкость экономики)



# Очистка данных (DB Browser for SQLite)

The screenshot shows the DB Browser for SQLite interface. On the left, the 'Schema БД' (Database Schema) pane lists tables (8), indexes (0), and views (16). The 'Views' section is expanded, showing a list of views including CommodityS, CommodityS1, MetroS, CommoditySM, PopulationS, and others. The main editor displays a SQL script to create a view named 'PopulationS'. The script uses a CASE statement to correct various country names. Below the script, a preview of the data is shown in a table format.

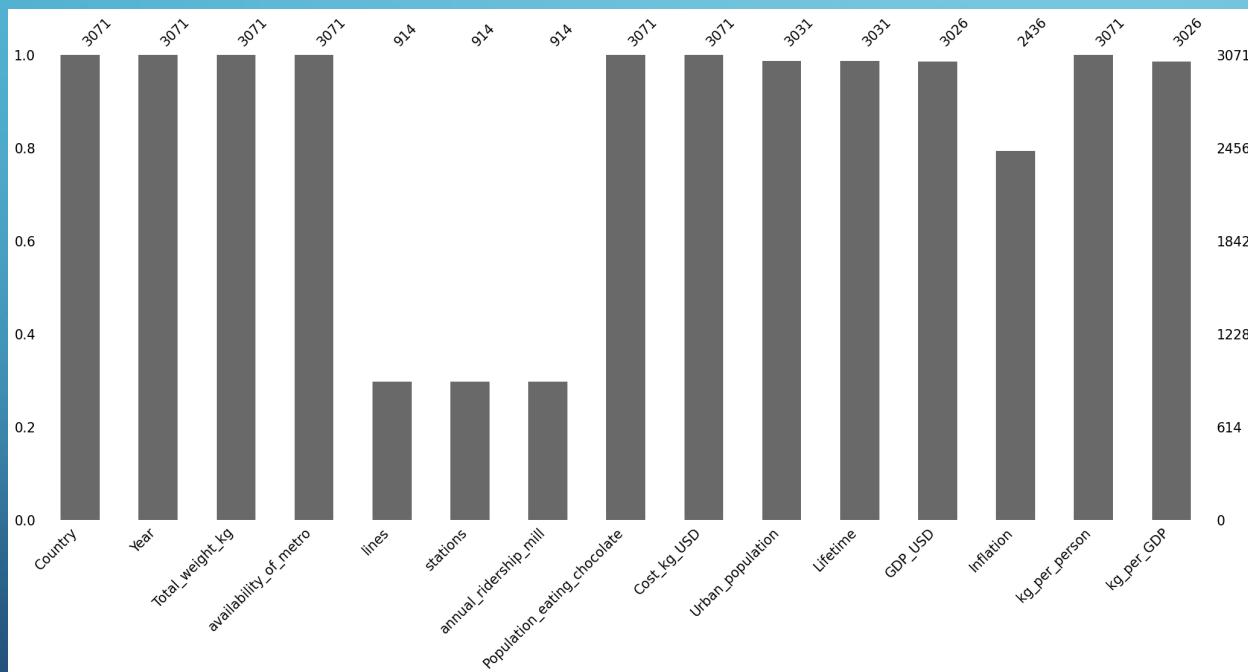
```
1  --CREATE VIEW PopulationS AS
2  WITH corrected_countries AS (SELECT CASE Area WHEN 'Bosnia Herzegovina' THEN 'Bosnia and Herzegovina'
3    WHEN 'Bolivia (Plurinational State of)' THEN 'Bolivia'
4    WHEN 'Central African Rep.' THEN 'Central African Republic'
5    WHEN 'Cook Isds' THEN 'Cook Islands'
6    WHEN 'Côte d'Ivoire' THEN 'Cote d'Ivoire'
7    WHEN 'Czechia' THEN 'Czech Republic'
8    WHEN 'Dominican Rep.' THEN 'Dominican Republic'
9    WHEN 'EU-28' THEN 'European Union (27)'
10   WHEN 'Fmr Fed. Rep. of Germany' THEN 'Germany'
11   WHEN 'Fmr Sudan' THEN 'Sudan'
12   WHEN 'FS Micronesia' THEN 'Micronesia'
13   WHEN 'Faeroe Isds' THEN 'Faroe Islands'
14   WHEN 'Iran (Islamic Republic of)' THEN 'Iran'
15   WHEN 'Lao People's Democratic Republic' THEN 'Laos'
16   WHEN 'Netherlands Antilles (former)' THEN 'Netherlands Antilles'
17   WHEN 'Netherlands (Kingdom of the)' THEN 'Netherlands'
18   WHEN 'Neth. Antilles' THEN 'Netherlands Antilles'
19   WHEN 'Republic of Korea' THEN 'South Korea'
20   WHEN 'Rep. of Moldova' THEN 'Republic of Moldova'
21   WHEN 'Russian Federation' THEN 'Russia'
22   WHEN 'Solomon Isds' THEN 'Solomon Islands'
23   WHEN 'State of Palestine' THEN 'Palestine'
24   WHEN 'Sudan (former)' THEN 'Sudan'
25   WHEN 'Swaziland' THEN 'Eswatini'
26   WHEN 'Syria' THEN 'Syrian Arab Republic'
27   WHEN 'TFYR of Macedonia' THEN 'North Macedonia'
28   WHEN 'Turks and Caicos Isds' THEN 'Turks and Caicos Islands'
29   WHEN 'Türkiye' THEN 'Turkey'
30   WHEN 'United Kingdom' THEN 'United Kingdom of Great Britain and Northern Ireland'
31   WHEN 'United Republic of Tanzania' THEN 'Tanzania'
32   WHEN 'United States of America' THEN 'USA'
33   WHEN 'Venezuela (Bolivarian Republic of)' THEN 'Venezuela'
34   WHEN 'Wallis and Futuna Isds' THEN 'Wallis and Futuna Islands'
35   ELSE Area END AS Country, Year, Value * 1000 AS Population FROM Population
36   WHERE Element = 'Total Population - Both sexes'
37   AND Year BETWEEN 1996 AND 2016)
38  SELECT Country, Year, Population FROM corrected_countries
39  ORDER BY Country, Year
```

	Country	Year	Population
1	Afghanistan	1996	17763266.0
2	Afghanistan	1997	18452091.0
3	Afghanistan	1998	19159996.0
4	Afghanistan	1999	19887785.0
5	Afghanistan	2000	20130327.0

# Очистка данных (здесь и далее: Jupyter Notebook и Python)

## 1 ЭТАП – Проверка и заполнение миссингов:

Inflation, Urban\_population, Lifetime: заполняются средними значениями по году  
GDP\_USD: для некоторых стран (Faroe Islands, Mayotte и др.) заполняется на основе значений других стран  
kg\_per\_person, kg\_per\_GDP: вычисляется после заполнения вышеуказанных столбцов



### Заполнение миссингов в Inflation:

```
inflation_means_by_year =  
df.groupby('Year')['Inflation'].transform('mean')  
df['Inflation'] =  
df['Inflation'].fillna(inflation_means_by_year)
```

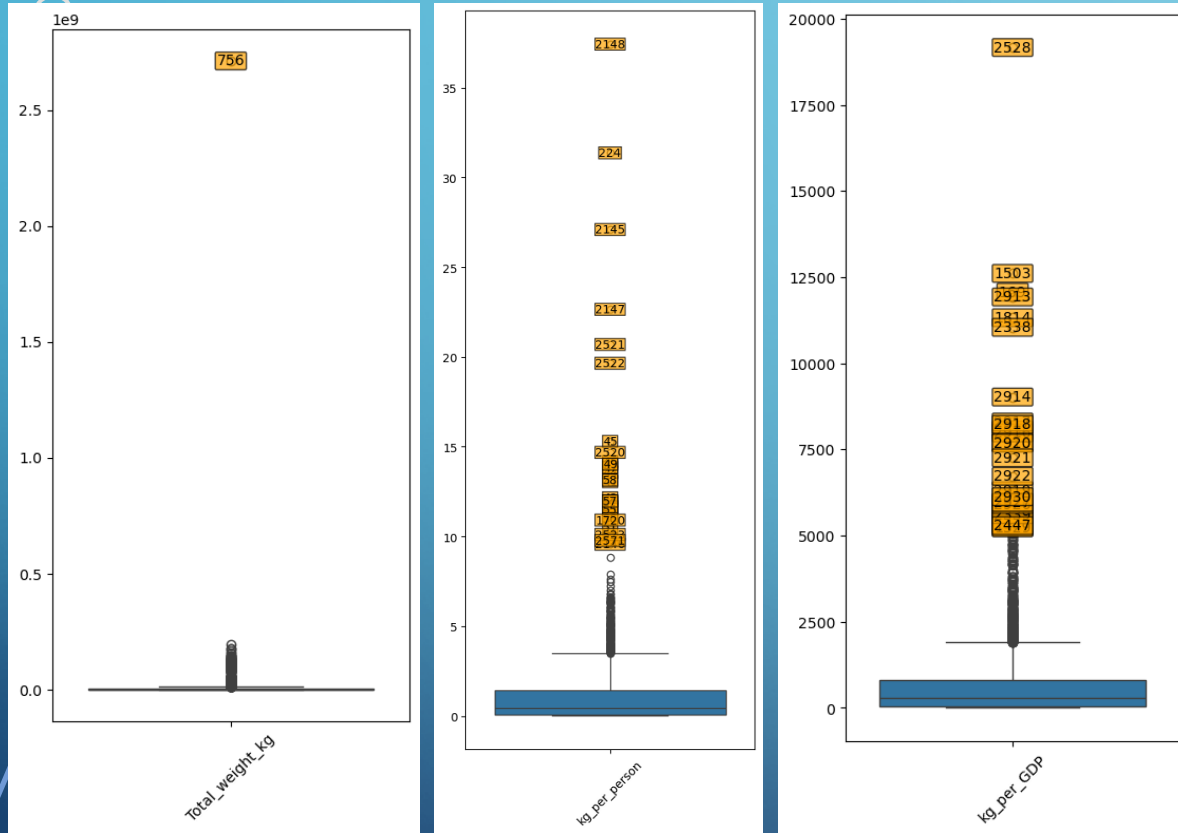
### Заполнение миссингов в kg\_per\_person:

```
mask_kg_per_person = df['kg_per_person'].isna()  
df.loc[mask_kg_per_person, 'kg_per_person'] =  
(df.loc[mask_kg_per_person, 'Total_weight_kg'] /  
df.loc[mask_kg_per_person,  
'Population_eating_chocolate']).astype('float32')
```

# Очистка данных

## 2 ЭТАП – Проверка и удаление выбросов:

Наиболее серьёзные выбросы замечены в столбцах Total\_weight\_kg, kg\_per\_person, kg\_per\_GDP



Найдено выбросов в 'Total\_weight\_kg': 1  
Найдено выбросов в 'kg\_per\_person': 29  
Найдено выбросов в 'kg\_per\_GDP': 33  
**Итого оставшихся строк: 3008**

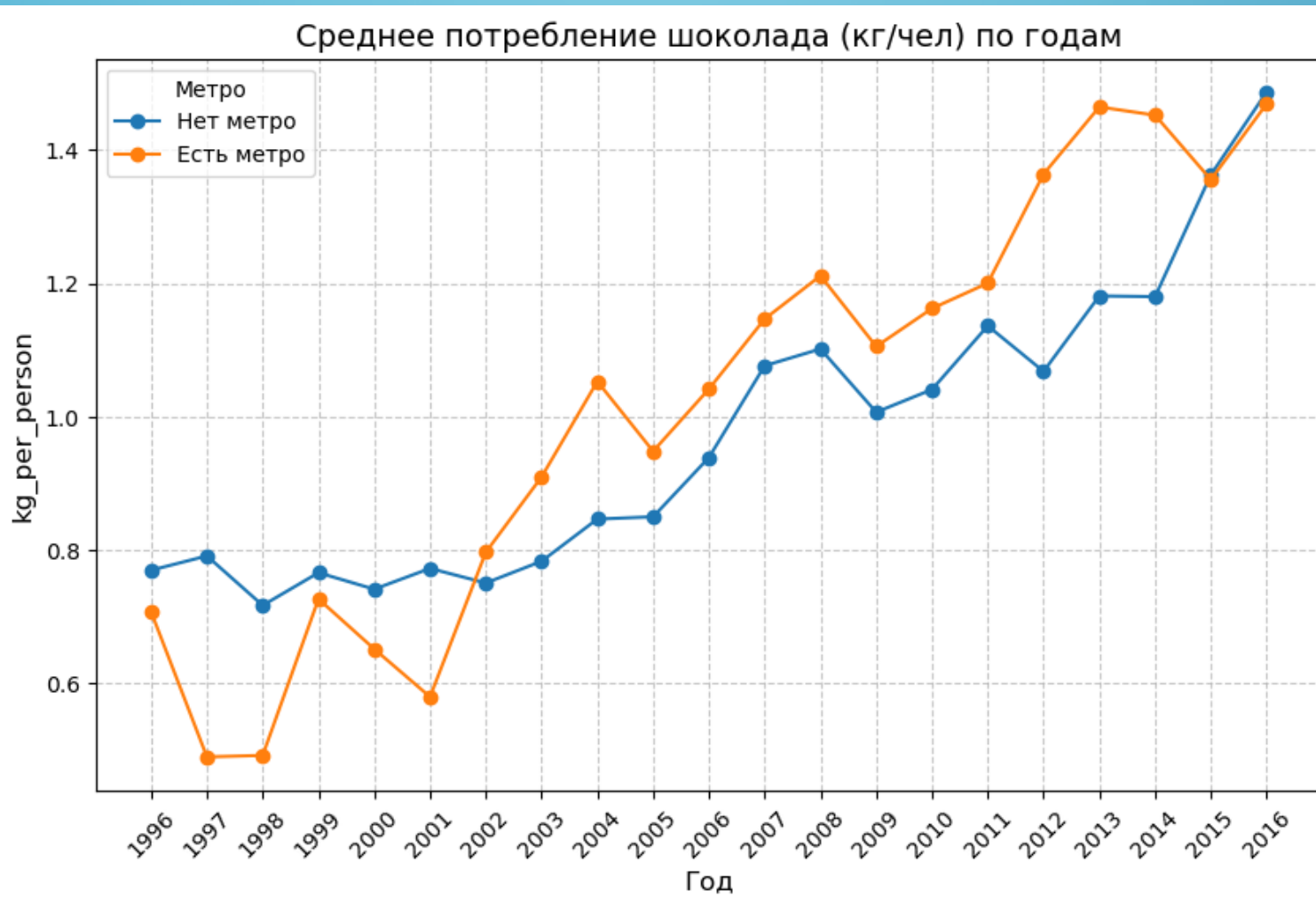
Функция для поиска выбросов методом z-score  
(из библиотеки NumPy):

```
def find_outliers_zscore(data, column, threshold=4):  
    z_scores = np.abs(stats.zscore(data[column]))  
    return data.loc[z_scores > threshold].index.tolist()  
* threshold=4 принят, чтобы не удалять «почти нормальные»  
значения.
```

# Описательный анализ

## Основные наблюдения:

1. Потребление шоколада изменяется с течением времени без резких скачков.
2. Потребление шоколада стабильно выше, чем в городах без метро.
3. Спад 2009 года – скорее всего связан с мировым финансовым кризисом.





# Статистическое сравнение групп

## T-test

Нулевая гипотеза ( $H_0$ ): «В странах с метро уровень потребления шоколада (kg\_per\_person) среди населения такой же как и без метро»

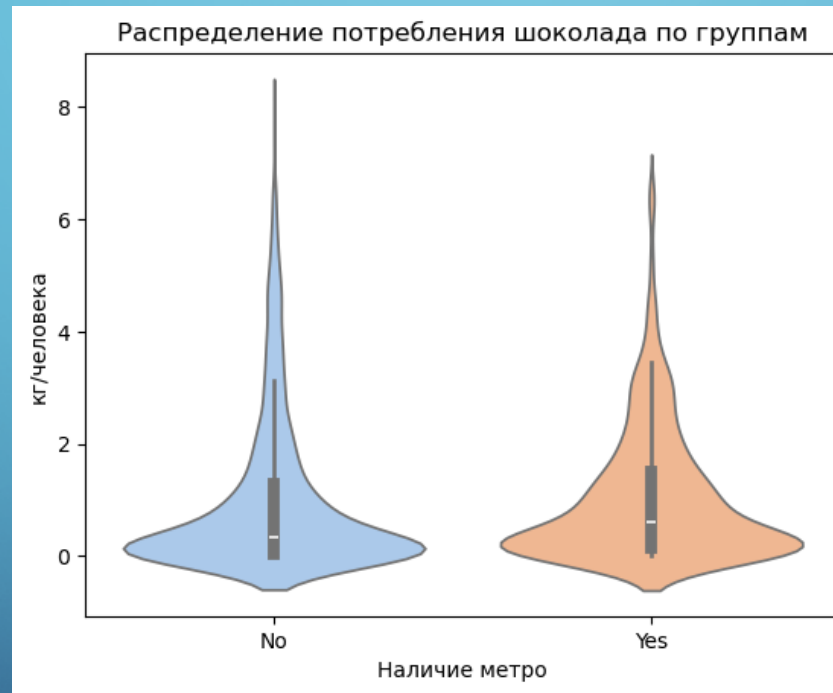
### Проведение T-теста:

```
with_metro =  
dfW2[dfW2['availability_of_metro'] == "Yes"]['kg_per_person']  
without_metro =  
dfW2[dfW2['availability_of_metro'] == "No"]['kg_per_person']  
t_statistic, p_value =  
stats.ttest_ind(with_metro,  
without_metro)
```

### Результаты T-теста:

**T-статистика: 1.336**

**P-значение: 0.1816 ( $>0.05$ )**



## Test Mann–Whitney (U-test)

### Результаты U-теста:

Медиана потребления (Страны с метро) :  
0.61 кг/чел

Медиана потребления (Страны без метро) : 0.34  
кг/чел

**Статистика: 1083137.50**

**P-значение: 0.0000**

### Результаты теста Cohen's d

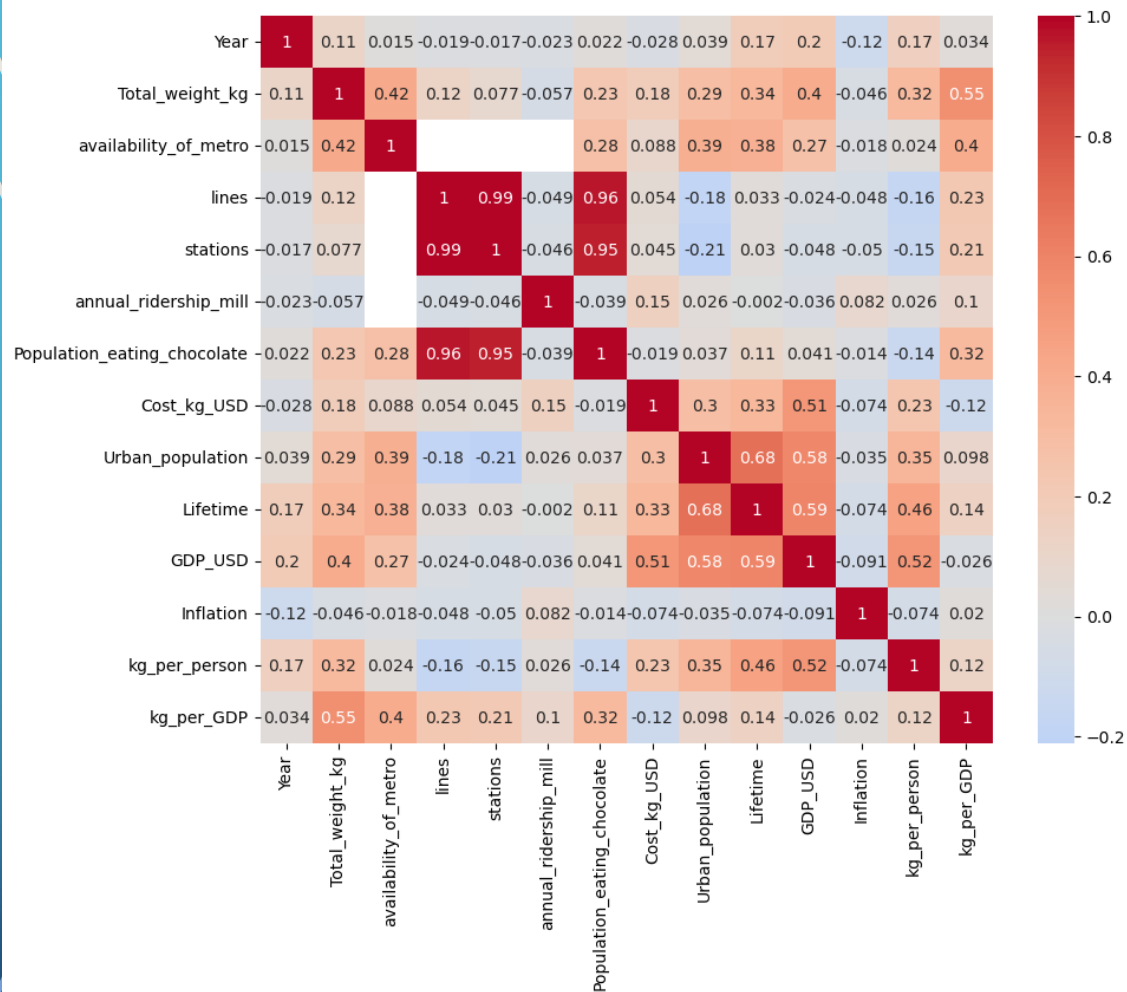
(Стандартизированная разница  
средних):

**Cohen's d: 0.055 (Очень слабый эффект)**

### Вывод:

Согласно U-тесту: медианное потребление шоколада значительно выше в странах с метро (нулевая гипотеза отвергается). Согласно T-тесту и Cohen's d-тесту: нет статистически значимых различий в среднем потреблении шоколада (нулевая гипотеза не отвергается).

# Корреляционный анализ



## Интересные корреляции:

1. Inflation и annual\_ridership\_mill практически не коррелируют ни с одним из показателей.
2. Lines и stations практически не связаны с количеством потребления шоколада.
3. GDP\_USD и kg\_per\_GDP (-0.026) – богатые страны тратят на шоколад меньшую долю экономики.
4. Urban\_population и kg\_per\_person (0.35) – городские жители едят больше шоколада, чем сельские.
5. kg\_per\_person и Lifetime (0.459) – шоколад продлевает жизнь!

Корреляция между availability\_of\_metro и kg\_per\_person: 0.024, т.е. практически отсутствует. Корреляция между availability\_of\_metro и kg\_per\_GDP: 0.4 (средняя). На основании того что, нет статистически значимых различий в среднем потреблении шоколада для дальнейшего исследования примем новую гипотезу: Наличие в стране метро увеличивает шоколадоемкость экономики (kg\_per\_GDP).

# Статистическое сравнение групп-2

## T-test

## Test Mann–Whitney (U-test)

Нулевая гипотеза ( $H_0$ ): «В странах с метро шоколадоёмкость экономики (kg\_per\_GDP) такая же как и без метро»

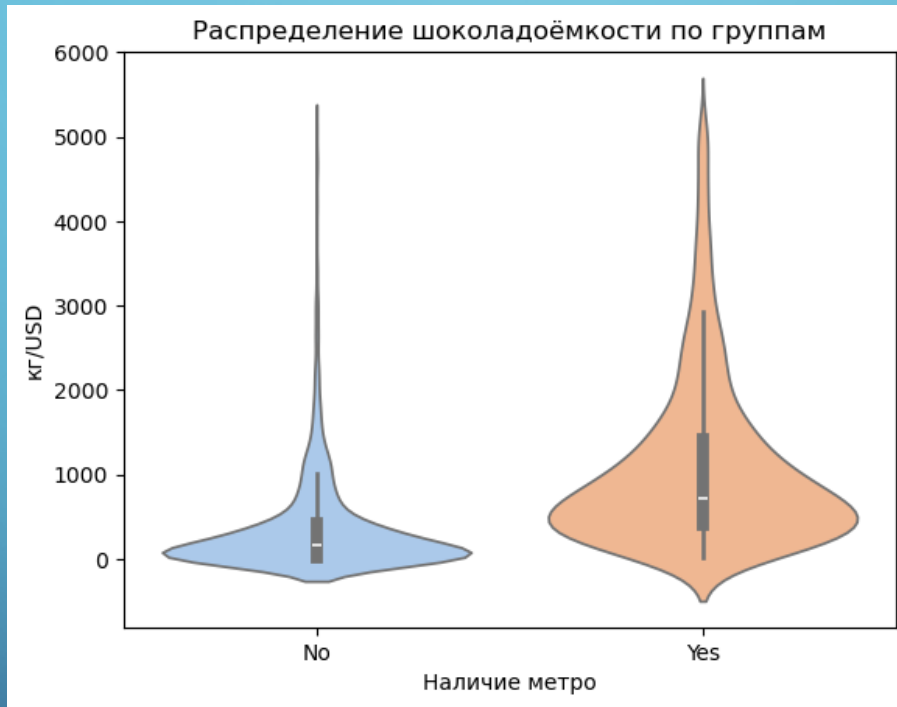
Проведение Т-теста:

```
with_metro =  
dfW2[dfW2['availability_of_metro'] == "Yes"]['kg_per_GDP']  
without_metro =  
dfW2[dfW2['availability_of_metro'] == "No"]['kg_per_GDP']  
t_statistic, p_value =  
stats.ttest_ind(with_metro,  
without_metro)
```

Результаты Т-теста:

**Т-статистика: 23.8247**

**Р-значение: 0.000 (<0.05)**



Результаты U-теста:

Медиана шоколадоёмкости (Страны с метро)  
: 724.68 кг/USD

Медиана шоколадоёмкости (Страны без метро)  
: 163.30 кг/USD

**Статистика: 1519378.00**

**Р-значение: 0.0000**

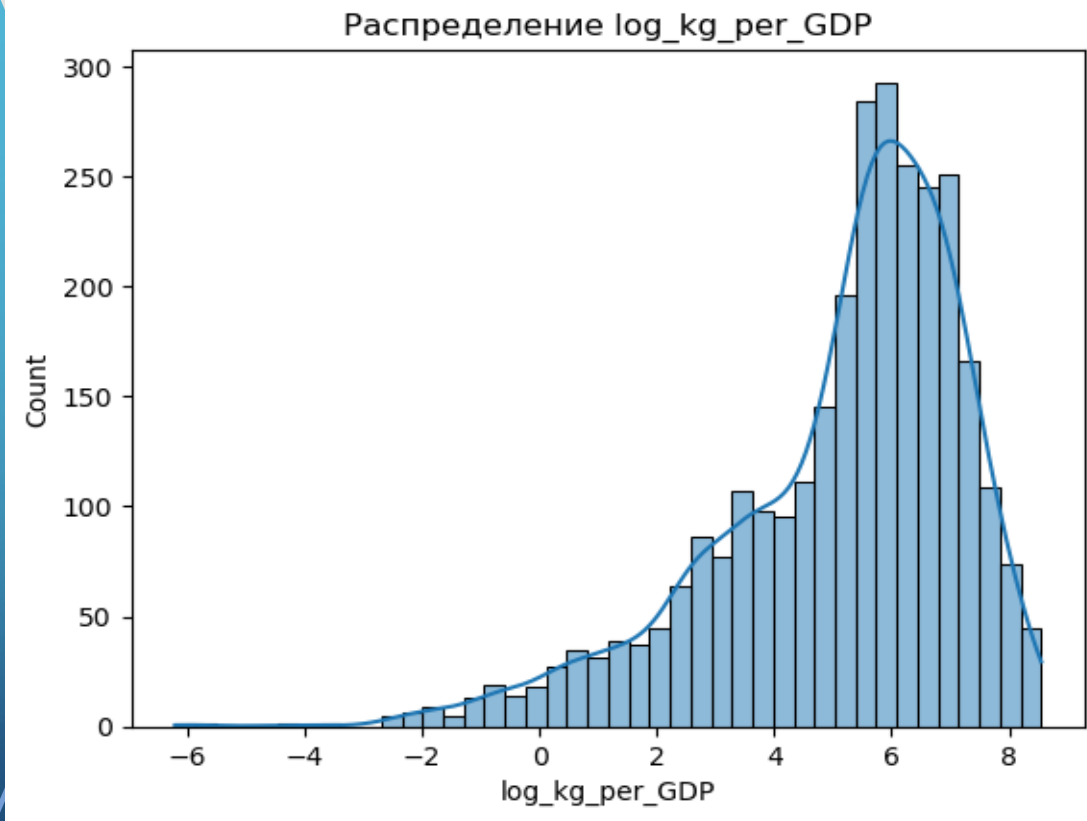
Результаты теста Cohen's d  
(Стандартизированная разница средних):

**Cohen's d: 0.855 (Очень сильный эффект)**

**Вывод:** Согласно всем 3 тестам: шоколадоёмкость значимо выше в странах с метро. Таким образом, нулевая гипотеза отвергается.

# Регрессионный анализ (библиотека Statsmodels)

Регрессия позволяет количественно оценить связь между предикторами и зависимой переменной, контролируя влияние других переменных. Принимаем зависимую переменную - `kg_per_GDP` и предикторы - `availability_of_metro`, `Urban_population`, `Cost_kg_USD`, `GDP_USD`, `kg_per_person`.



Для показателей `kg_per_GDP`, `Cost_kg_USD`, `GDP_USD`, `kg_per_person` применено логарифмирование для линеаризации зависимости.

## Ключевые показатели:

$R^2 = 0.464$  — объяснено 46,4% вариации `log_kg_per_GDP` (умеренно высокий показатель)

**p-value = 0** (для всех переменных) — все значимые.

**Cond. No. = 675** — умеренная мультиколлинеарность.

**Durbin-Watson = 0.185** — сильная положительная автокорреляция.

Результаты теста Shapiro-Wilk (Нормальности распределения зависимой переменной):

**Shapiro-Wilk Test Statistic: 0.92** и **P-значение: 0.0000**  
(распределение ненормальное)

Коэффициент при `availability_of_metro` = 2.396 — наличие метро сильно увеличивает шоколадоёмкость.

**Вывод:** Гипотеза подтверждается с высокой статистической значимостью.

## Коэффициент асимметрии (skewness) = -1.158

Отрицательное значение указывает на левую асимметрию (длинный хвост влево).



# Регрессионный анализ (библиотека Scikit-Learn)

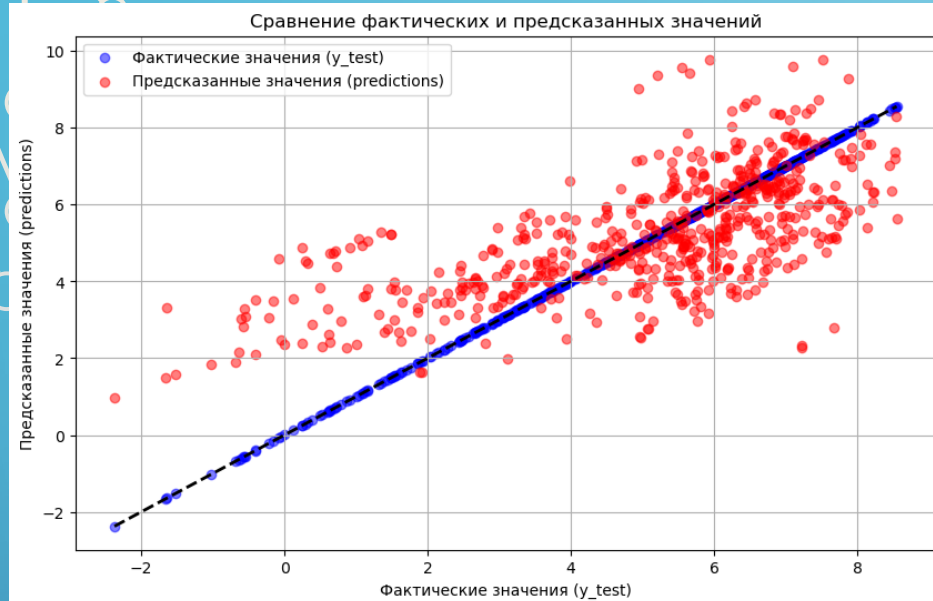
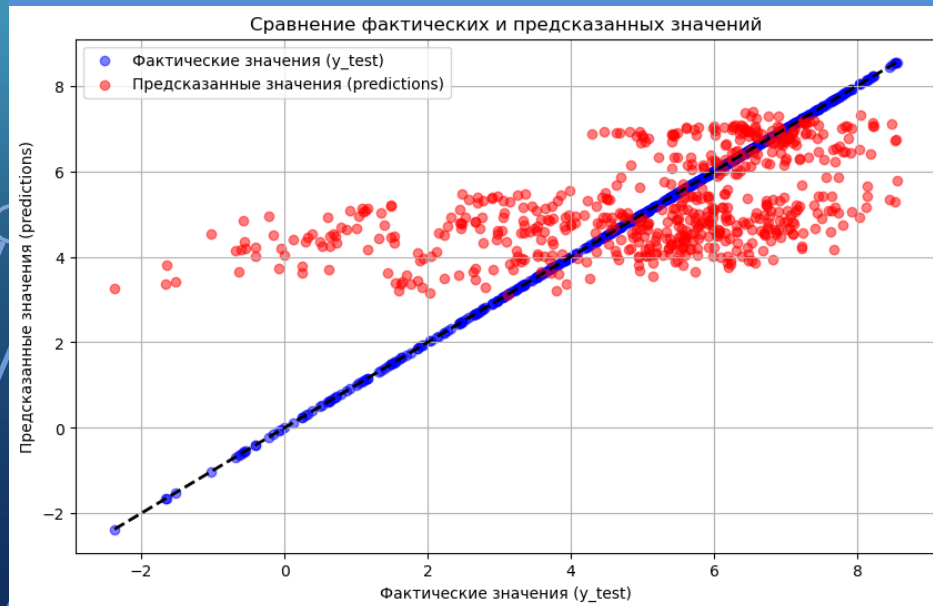


График без показателей  $\log\_Cost\_kg\_USD$ ,  $\log\_GDP\_USD$  (снижение ошибок модели)



Модель:

```
X = dfK1[['availability_of_metro', 'Urban_population', 'log_Cost_kg_USD',  
'log_GDP_USD', 'log_kg_per_person']]  
y = dfK1['log_kg_per_GDP']  
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,  
random_state=170)  
lm = LinearRegression()  
lm.fit(X_train, y_train)  
predictions = lm.predict(X_test)
```

Результаты теста:

MAE: 1.24; MSE: 2.56; RMSE: 1.6 — неплохие, но не идеальные предсказания (11–15% от общего разброса данных).

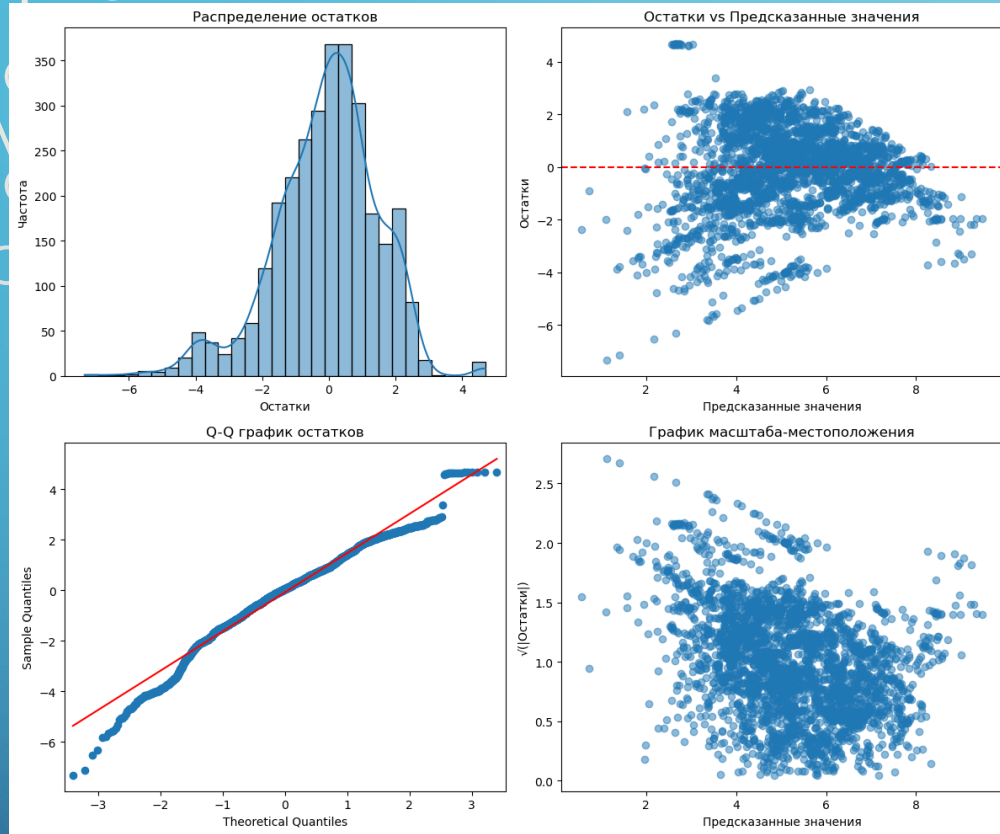
$\log\_Cost\_kg\_USD$ ,  $\log\_GDP\_USD$  — показали высокую мультиколлинеарность.

После масштабирования (все признаки приводятся к единому масштабу без изменения распределения) коэффициент при  $availability\_of\_metro = +1.09$  — т.е. наличие метро увеличивает шоколадоёмкость.

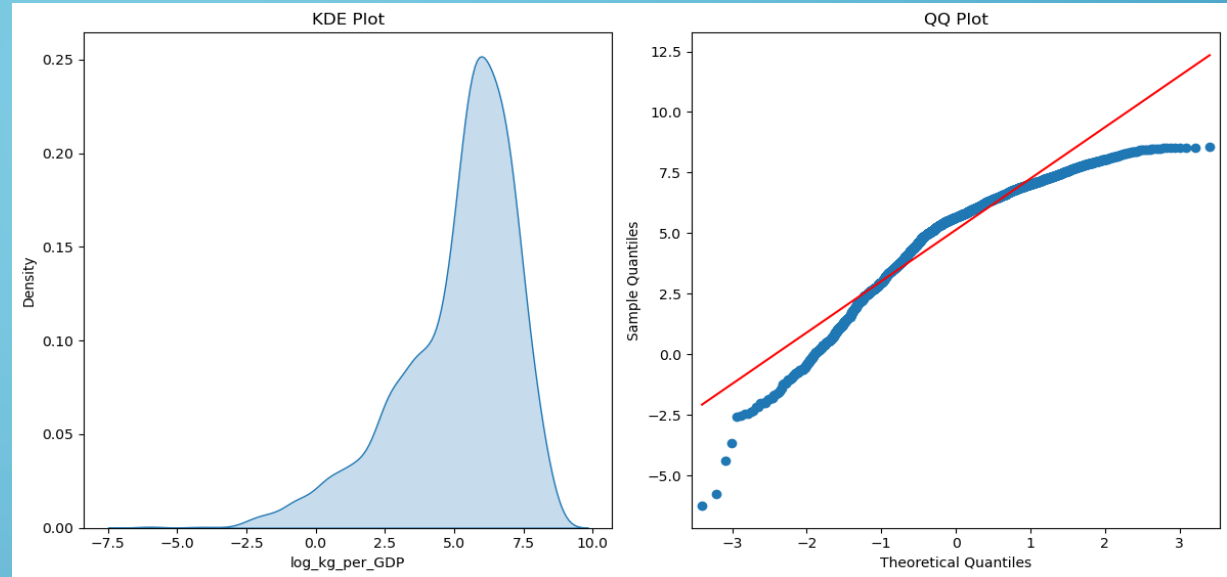
$R^2 = 0.422$  - модель объясняет 42.2% дисперсии целевой переменной (средний показатель).

Вывод: Гипотеза подтверждается.

# Дисперсионный анализ (Робастная линейная регрессия)



Малый эффект выбросов: Всего 3.76% данных (113 строк), т.е. модель в целом устойчива.  
Медианное абсолютное отклонение (MAD) = 0.918 — относительно небольшой разброс ошибок вокруг медианы.



Модель учитывает выбросы и гетероскедастичность, которые искажают оценки OLS. Робастная регрессия уменьшает влияние аномалий за счёт присвоения им меньших весов:

```
model = smf.rlm('log_kg_per_GDP ~ availability_of_metro +  
Urban_population + log_Cost_kg_USD + log_GDP_USD + log_kg_per_person',  
data=dfK1, M=sm.robust.norms.HuberT()).fit()
```

Результаты теста:

Коэффициент при `availability_of_metro` = +2.215 — наличие метро сильно увеличивает шокадоёмкость.  
Сходимость достигнута за 25 итераций, что говорит о стабильности оценок.

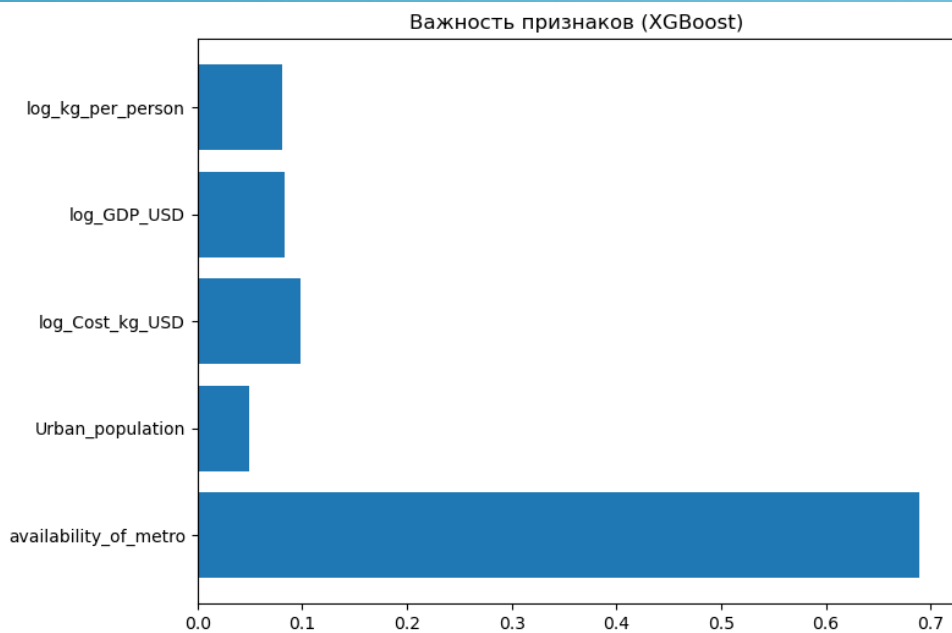
**Вывод:** Гипотеза подтверждается.

# Машинное обучение

Для обучения применим 6 разных моделей:

1. Linear Regression,
2. Polynomial Regression,
3. Decision Tree,
4. Random Forest,
5. Gradient Boosting,
6. XGBoost.

Модель	MAE (Средняя ошибка предсказания)	MSE (Квадратичная ошибка)	R2 (Дисперсия)
XGBoost	0.296	0.232	0.947
Random Forest	0.341	0.375	0.915
Decision Tree	0.397	0.626	0.858
Gradient Boosting	0.804	1.182	0.732
Polynomial Regression	1.084	2.119	0.521
Linear Regression	1.240	2.556	0.421

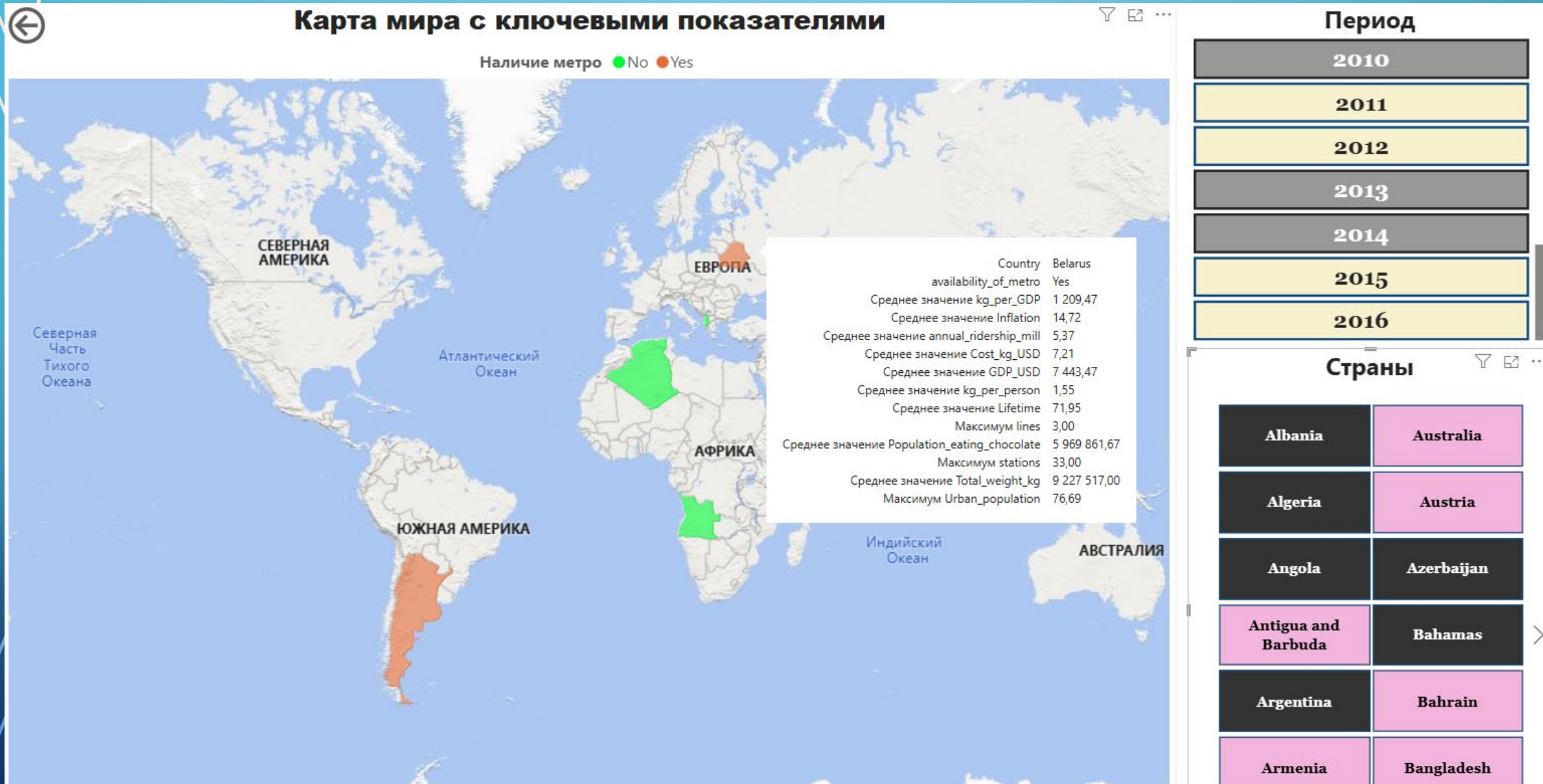


**Вывод:** Модель XGBoost (eXtreme Gradient Boosting) лучше всех улавливает закономерности в данных. Linear и Polynomial Regression показали слабые результаты, что говорит о нелинейности данных.

Анализ важности признаков (XGBoost):  
Самый важный признак влияющий на шоколадоёмкость – наличие метро.  
Остальные признаки – практически одинаково менее важные.



# Создание дашборда (Power BI)

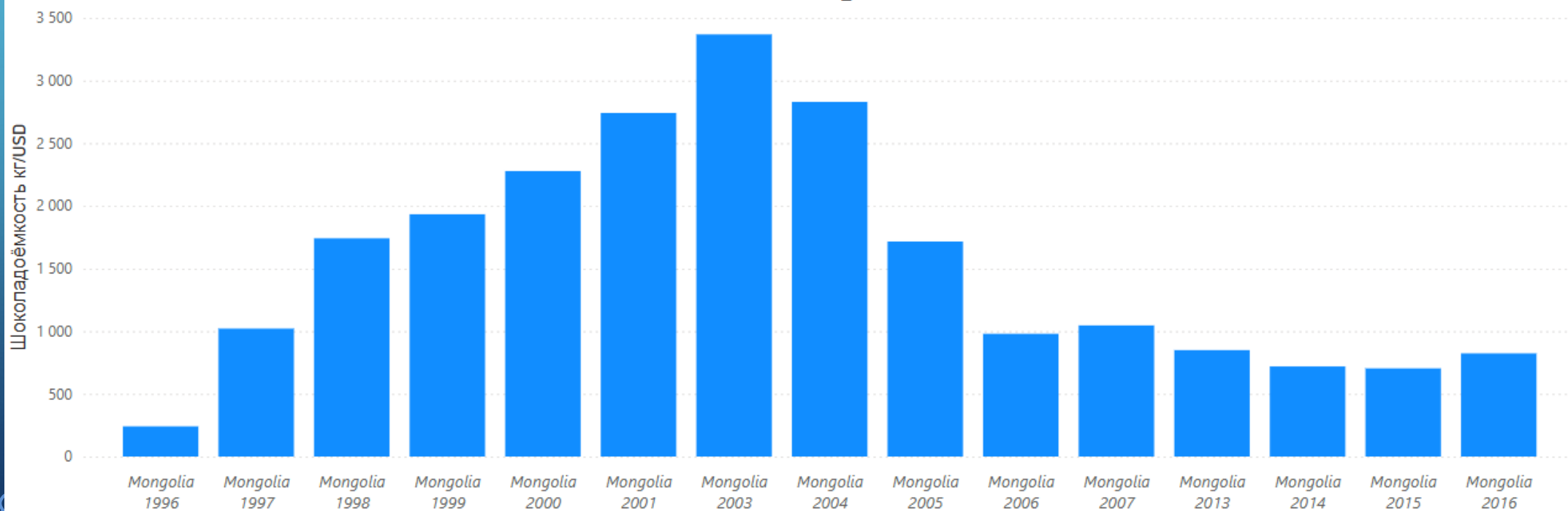




# Создание дашборда (Power BI)-2

Дата					
2000					
Страна	No	9,23	537	11,60	57,13
Mongolia	Наличие метро	Стоимость 1 кг шоколада (USD)	ВВП (USD)	Инфляция, %	Доля городского населения, %
1,22 млн	2 277,57		63,21	1 031 176	
Вес импортируемого шоколада, кг	Потребление на душу населения (кг/человек)		Средняя продолжительность жизни	Население, потребляющее шоколад	

Шоколадоёмкость страны по годам



## Выводы:

В проекте были исследованы 2 гипотезы:

1. «Наличие в стране метро увеличивает уровень потребления шоколада среди населения» – отвергнута (нет статистически значимых различий в странах метро и без).
2. «Наличие в стране метро увеличивает шоколадоёмкость экономики» - подтвердилась (согласно всем проведённым этапам статистического анализа данных).

Однако в исследовании обнаружилось ограничения:

1. Исследование носит корреляционный характер.
2. Наличие метро может быть показателем развитой инфраструктуры, а не прямой причиной роста потребления шоколада.
3. Заполнение пропусков средними значениями может исказить распределения данных.
4. Логарифмирование шоколадоёмкости не полностью устранило асимметрию.
5. Страны с экстремальными значениями (т.е. страны попавшие в выбросы) требуют отдельного анализа.
6. Существуют другие значимые переменные (например, GDP\_USD).
7. Модель данных не является линейной, т.е. имеются сложные взаимосвязи.
8. Высокая мультиколлинеарность у  $\log\_GDP\_USD$  (68) и  $\log\_Cost\_kg\_USD$  (47) затрудняет разделение их влияния на гипотезу.

# Выводы:

В процессе исследования использовались следующие виды аналитики:

## 1. Описательная аналитика (Descriptive Analytics):

Использованные инструменты:

Визуализация данных: Графики распределения (histplot, kdeplot, violinplot); Heatmap корреляции.

Описательная статистика: Средние, медианы, стандартные отклонения для переменных; Группировка данных по наличию метро.

## 2. Диагностическая аналитика (Diagnostic Analytics):

Использованные инструменты:

Анализ пропущенных значений (msno.bar, заполнение значением средними и медианами).

Обнаружение и удаление выбросов: Метод z-score, Визуализация через boxplot.

Статистические тесты: t-test; Test Mann–Whitney; Расчёт эффекта через Cohen's d.

## 3. Предиктивная аналитика (Predictive Analytics):

Использованные инструменты:

Линейная регрессия (с использованием библиотек Statsmodels, Scikit-Learn).

Машинное обучение (6 моделей).

## 4. Предписывающая аналитика (Prescriptive Analytics):

Использованные инструменты:

Робастная регрессия (smf.rlm с HuberT).

Дисперсионный анализ (ANOVA): Проверка остатков на гетероскедастичность; Q-Q графики для проверки нормальности.

Оптимизация моделей: Подбор признаков через VIF (мультиколлинеарность); Масштабирование данных (StandardScaler, MinMaxScaler).



## Рекомендации:

1. Учёт климатических факторов: холодный климат может влиять на потребление шоколада.
2. Локализация производства: В странах с высоким ВВП (где шоколадоёмкость ниже) сделать акцент на премиальный шоколад.
3. Сбор дополнительных данных о культуре питания: наличие кафе/магазинов в метро как медиатор потребления.
4. Анализ кластеров: выделить группы стран с похожими паттернами (например, Европа - Африка).
5. Учёт религиозных факторов.
6. Проверка эффекта времени после постройки метро.



# СПАСИБО ЗА ВНИМАНИЕ!



Ссылка на проект:

[https://github.com/Paskored/Data-Analyst-Project\\_Kashtanov](https://github.com/Paskored/Data-Analyst-Project_Kashtanov)