ГУАП

КАФЕДРА № 41

ОТЧЕТ ЗАЩИЩЕН С ОЦЕ	НКОЙ		
ПРЕПОДАВАТЕЛЬ			
ст. преподава	атель		В.В. Боженко
должность, уч. степо	ень, звание	подпись, дата	инициалы, фамилия
		.	
	ОТЧЕТ О ЛА	АБОРАТОРНОЙ РАБ	OTE №1
	ПРЕДВАРИТ	ЕЛЬНЫЙ АНАЛИЗ Д	[АННЫХ
	, ,	,	1
	по курсу: ВВЕЛ	ДЕНИЕ В АНАЛИЗ Д	АННЫХ
	по куреу. ВВД		
РАБОТУ ВЫПОЛН	ИЛ		
СТУДЕНТ ГР. №	4316		Э.А.Чылдырлы
C10 ADIII 11.51		подпись, дата	инициалы, фамилия

Цель лабораторной работы: осуществить предварительную обработку данных сsv-файла, выявить и устранить проблемы в этих данных.

Описание предметной области:

- movies.csv Набор данных
- title Название фильма
- release year Год выпуска
- score Рейтинг
- number of votes Количество оценок
- duration Продолжительность
- main genre Основной жанр
- main production Страна производства

Индивидуальное задание:

- 1. Группировка жанр и количество фильмов каждого main_production.
- 2. Группировка жанр и количество фильмов каждого main_production. Создать датафрейм. Переименовать столбец с количеством в "count". Отсортировать по убыванию столбца "count"
- 3. Сводная таблица (pivot_table) средний рейтинг (score) фильмов по main_production. Отсортировать по убыванию рейтинга. Округлить до одного знака.
- 4. Сводная таблица (pivot_table) медианный рейтинг (score) фильмов по жанрам столбцы и main_production- строки. Отсортировать по возрастанию main production

Ход работы

В первую очередь была импортирована необходимая для работы библиотека pandas После чего осуществлено чтение файла с данными с помощью метода read csv() (рис. 1).

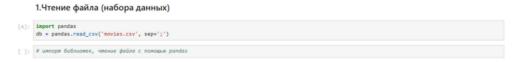


Рисунок 1 — Чтение данных

Далее с помощью метода head(20) ,были выведены первые 20 строк таблицы. Это дало представление о том, какая информация содержится в базе даныых (ее столбцы и содержащиеся в них значения) (рис. 2).

[6]:	db.head(20)								Œ	1	, 1	, ±	7	
6]:		title	release_year	score	number_of_votes	duration	main_genre	main_production						
	0 David Attenborough	: A Life on Our Planet	2020.0	9.0	31180.0	83	documentary	GB						
	1	Inception	2010.0	8.8	2268288.0	148	scifi	GB						
	2	Forrest Gump	1994.0	8.8	1994599.0	142	drama	US						
	3	Anbe Sivam	2003.0	8.7	20595.0	160	comedy	IN						
	4	Bo Burnham: Inside	2021.0	8.7	44074.0	87	comedy	US						
	5	Saving Private Ryan	1998.0	8.6	1346020.0	169	drama	US						
	6	Django Unchained	2012.0	8.4	1472668.0	165	western	US						
	7	Dangal	2016.0	8.4	180247.0	161	action	IN						
	8 Bo B	urnham: Make Happy	2016.0	8.4	14356.0	60	comedy	US						
	9	Louis C.K.: Hilarious	2010.0	8.4	11973.0	84	comedy	US						
	10 Dave Chap	pelle: Sticks & Stones	2019.0	8.4	25687.0	65	comedy	US						
	11	3 Idiots	2009.0	8.4	385782.0	170	comedy	IN						
	12	Black Friday	2004.0	8.4	20611.0	143	crime	IN						
	13	Super Deluxe	2019.0	8.4	13680.0	176	thriller	IN						
	14 Winter on Fire: Ukrain	-	2015.0	8.3	17710.0		documentary	UA						
		oon a Time in America	1984.0	8.3	342335.0	229	drama	US						
	16	Taxi Driver	1976.0	8.3	795222.0	113	crime	US						
	17	Like Stars on Earth	2007.0	8.3	188234.0	165	drama	IN						
	18	Bo Burnham: What.	2013.0	8.3	11488.0	60	comedy	US						
	19	Full Metal Jacket	1987.0	8.3	723306.0	116	drama	GB						

Рисунок 2 — Вывод первых 20 строк таблицы

С помощью метода info() была получена структурная информация о датасете: объем данных, индексация, объем памяти, типы данных и их полнота (рис. 3). Помимо этого, с помощью метода describe() были получены статистические закономерности числовых атрибутов (рис. 4). Теперь мы понимаем, что датасет содержит информацию о 389 фильмах с 2010-х годов (медианный год выпуска - 2014), где большинство картин имеют высокие рейтинги (среднее 7.5/10) при стандартной продолжительности около 2 часов. Количество оценок варьируется от 10 тысяч до 2.3 миллионов, что указывает на смесь нишевых и популярных фильмов, при этом основные жанры представлены драмами и комедиями из разных стран, преимущественно США и Индии.

```
2.2 Оценка данных с помощью метода info.
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 389 entries, 0 to 388 Data columns (total 7 columns):
 # Column
                          Non-Null Count Dtype
     title
                          384 non-null
                                              object
                          387 non-null
386 non-null
      release_year
      score
                                              float64
     number_of_votes 388 non-null duration 389 non-null
                                              float64
                                              int64
     main_genre
                           389 non-null
                                              object
      main_production
                                              object
dtypes: float64(3), int64(1), object(3) memory usage: 21.4+ KB
```

Рисунок 3 — Оценка данных с помощью метода info()

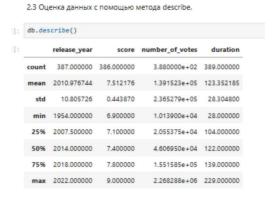


Рисунок 4 — Оценка данных с помощью метода describe()

Для выявления несоответствий между названием столбцов и их содержимым был использован метод columns() (рис. 5). Проблем с несоответствием или неверным написанием невыявлено.

Рисунок 5 — Вывод названий столбцов с помощью метода columns()

Далее была проделана работа с возможными пропусками. Данные были проверены с помощью метода db[db.isnull().any(axis=1)], который возвращает датафрейм со значениям True/False в зависимости от наличия или отсутствия NaN в строке, и проверяет, есть ли в ней хотя бы одно такое, и удалены с помощью метода dropna() (рис. 6).

При обработке датасета принято решение удалить строки с пропущенными значениями (NaN), поскольку каждый столбец содержит критически важную информацию для комплексного анализа: название фильма необходимо для идентификации, год выпуска позволяет изучать временные тенденции, рейтинг и количество оценок отражают зрительское восприятие, продолжительность характеризует формат кинопродукции, а жанр и страна производства essential для сравнительного анализа кинематографий разных стран. Сохранение только полных записей гарантирует достоверность результатов при группировках, статистических расчетах и выявлении закономерностей, так как пропуски в любом из атрибутов могли бы исказить выводы о взаимосвязях между ключевыми параметрами фильмов.

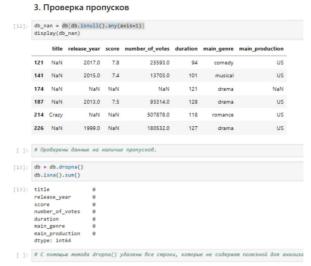


Рисунок 6 — Проверка пропусков

Далее датафрейм проверяется на дубликаты. В первую очередь были найдены и удалены повторяющиеся строки. Для этого были использованы метод duplicated() и drop_duplicates() соответственно (рис. 7). После чего данные были проверены на наличие неявных дубликатов (например, одинаковых названий с разным регистром или некорректным написанием одного и того же названия) с помощью метода unique(). Такие данные просто были заменены с помощью метода replace() (рис. 8).

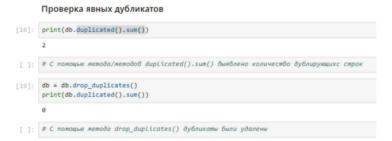


Рисунок 7 — Проверка явных дубликатов

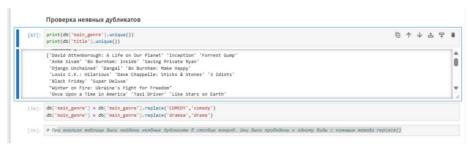


Рисунок 8 — Проверка неявных дубликатов

Далее были исправлены несоответствия типов данных. В процессе анализа были выявлены несоответствия данных и их типов в столбцах с годом выпуска и количеством

оценок. В обоих случаях тип данных float был заменен на datetime и int соответственно (рис. 9).

[60]:	db[<pre>'release_year'] = pandas.to_datetime 'number_of_votes'] = db['number_of_v nead(20)</pre>], format='XY')				
[60]:		title	release_year	score	number_of_votes	duration	main_genre	main_production	
	0	David Attenborough: A Life on Our Planet	2020-01-01	9.0	31180	83	documentary	GB	
	1	Inception	2010-01-01	8.8	2268288	148	scifi	GB	
	2	Forrest Gump	1994-01-01	8.8	1994599	142	drama	US	
	3	Anbe Sivam	2003-01-01	8.7	20595	160	comedy	IN	
	4	Bo Burnham: Inside	2021-01-01	8.7	44074	87	comedy	US	
	5	Saving Private Ryan	1998-01-01	8.6	1346020	169	drama	US	
	6	Django Unchained	2012-01-01	8.4	1472668	165	western	US	
	7	Dangal	2016-01-01	8.4	180247	161	action	IN	
	8	Bo Burnham: Make Happy	2016-01-01	8.4	14356	60	comedy	US	
	9	Louis C.K.: Hilarious	2010-01-01	8.4	11973	84	comedy	US	
	10	Dave Chappelle: Sticks & Stones	2019-01-01	8.4	25687	65	comedy	US	
	11	3 Idiots	2009-01-01	8.4	385782	170	comedy	IN	
	12	Black Friday	2004-01-01	8.4	20611	143	crime	IN	
	13	Super Deluxe	2019-01-01	8.4	13680	176	thriller	IN	
	14	Winter on Fire: Ukraine's Fight for Freedom	2015-01-01	8.3	17710	98	documentary	UA	
	15	Once Upon a Time in America	1984-01-01	8.3	342335	229	drama	US	
	16	Taxi Driver	1976-01-01	8.3	795222	113	crime	US	
	17	Like Stars on Earth	2007-01-01	8.3	188234	165	drama	IN	
	18	Bo Burnham: What.	2013-01-01	8.3	11488	60	comedy	US	
	19	Full Metal Jacket	1987-01-01	8.3	723306	116	drama	GB	

Рисунок 9 — Проверка типов данных

Индивидуальное задание

Задание 1. Группировка — жанр и количество фильмов каждого main_production.

После предвратительной подготовки данных было начато выполнение индивидуального задания. Была сделана группировка по жанру и количеству фильмов каждой страны производителя (рис. 10).

Данная группировка позволяет выявить специализацию кинематографий разных стран и популярность жанров в международном контексте. Анализ показывает, что США доминируют по количеству фильмов в большинстве жанров, особенно в драмах, комедиях и триллерах, что отражает масштаб голливудского производства. Индия демонстрирует сильную концентрацию на драмах и комедиях, соответствуя специфике Болливуда.

Рисунок 10 — Результат выполнения первого задания

Задание 2. Группировка — жанр и количество фильмов каждого main_production. Создать датафрейм. Переименовать столбец с количеством в "count". Отсортировать по убыванию столбца "count".

Была выполнена группировка данных по жанрам и странам производства с подсчетом количества фильмов в каждой категории. После создания исходного датафрейма столбец с количеством фильмов был переименован в "count", а затем данные были отсортированы по убыванию данного показателя (рис. 11).

```
Группировка - жанр и количество фильмов каждого main_production. Создать датафрейм. Пере
      Отсортировать по убыванию столбца "count"
73]: grouped = db.groupby(['main_genre', 'main_production']).size().reset_index(name='movie_count')
     grouped = grouped.rename(columns={'movie count': 'count'})
     grouped_sorted = grouped.sort_values('count', ascending=False)
      current_genre = None
      for index, row in grouped_sorted.iterrows():
         if row['main_genre'] != current_genre:
             print(f"[row['main_genre']:<20) {row['main_production']:<10} {row['count']:<10}")
current_genre = row['main_genre']</pre>
           print(f"{'':<20} {row['main_production']:<10} {row['count']:<10}")</pre>
     drama
     thriller
     drama
     romance
crime
      fantasy
     crime
      western
     comedy
```

Рисунок 11 — Результат выполнения второго задания

Задание 3. Сводная таблица (pivot_table) — средний рейтинг (score) фильмов по main_production. Отсортировать по убыванию рейтинга. Округлить до одного знака.

Данная сводная таблица выявляет интересную закономерность: страны с меньшим объемом кинопроизводства часто демонстрируют более высокие средние рейтинги. Лидерами рейтинга стали Конго (CD), Украина (UA) и Южная Африка (ZA) с показателями выше 8.0, что может объясняться тщательным отбором фильмов для международного проката или пісhе-специализацией. Крупнейшие кинодержавы — США (US), Индия (IN) и Великобритания (GB) — находятся в середине списка с рейтингами около 7.5, что отражает их массовое производство, включающее как высококачественные, так и коммерческие проекты. Наблюдается обратная корреляция между объемом производства и средним рейтингом (рис. 12).

```
pivot_db = db.pivot_table(
                              # Столбец для агрегации
   values='score', # Столбец для агрегации index='main_production', # Группировка по странам
    aggfunc='mean'
                              # Среднее значение
).reset_index()
pivot_db = pivot_db.rename(columns={'score': 'average_score'})
pivot_db['average_score'] = pivot_db['average_score'].round(1)
pivot_db = pivot_db.sort_values('average_score', ascending=False)
for index, row in pivot_db.iterrows():
print(f"{row['main_production']:<10} {row['average_score']:<15}")</pre>
           8.2
            8.1
           8.1
NZ
           7.8
```

Рисунок 12 — Результат выполнения первого задания

Задание 4. Сводная таблица (pivot_table) — медианный рейтинг (score) фильмов по жанрам — столбцы и main_production — строки. Отсортировать по возрастанию main production.

Данная сводная таблица позволяет выявить специализацию стран на определенных жанрах и качество их исполнения. Анализ показывает, что разные страны достигают максимальных рейтингов в специфичных для них жанрах: например, Турция (TR) демонстрирует высокие показатели в драмах и комедиях, Южная Корея (KR) - в драмах, а Япония (JP) - в романсах и фантастике (рис. 13).

```
63]: pivot_db = db.pivot_table(
     values='score',
     index='main production'.
     aggfunc='median',
     fill_value=0
   pivot_db = pivot_db.round(1)
   pivot_db = pivot_db.sort_index(ascending=True)
   display(pivot db)
    main genre action animation comedy crime documentary drama fantasy horror musical romance scifi sports thriller war western
   main_production
                  0.0 7.5 0.0
                                 00 00 00 00
                  0.0 8.1 0.0
                                                   0.0 0.0 0.0
                                                               0.0
            0.0
         AU
                                 0.0 7.3 0.0
                                              0.0
                                                                   7.3 0.0
                                                                           0.0
         BE 0.0 0.0 0.0 0.0 7.9 0.0 0.0 7.0 0.0 0.0 0.0 0.0
                                                                   0.0 0.0
                                                                           0.0
      0.0 7.2
                   0.0
                       7.1 0.0
                                           0.0 0.0
                                                  0.0
                                                        0.0 8.0 0.0
      0.0
                                  0.0 7.2
                                           7.2
                                                   0.0
    DE 7.1 0.0 0.0 0.0 0.0 7.2 0.0 7.5 0.0 0.0 0.0 7.4 7.8 0.0 0.0
```

Рисунок 13 — Результат выполнения четвертого задания

Вывод

Были осовоены навыки работы в JupyterNotebook и получены знания в сфере предварительного анализа данных.

Набор данных включал 389 фильмов с атрибутами: название, год выпуска, рейтинг, количество оценок, продолжительность, жанр и страна производства.

Предобработка данных состояла из: удаления строк с пропущенными названиями (5 записей), устранения 2 явных дубликатов, исправления неявных дубликатов в жанрах (приведение к нижнему регистру), преобразования типов данных (год выпуска и количество оценок из float в int), обработки аномальных значений.

Группировка по жанрам и странам выявила специализацию кинематографий: США доминируют в производстве драм (21), комедий (15) и триллеров (10); Индия специализируется на драмах (8) и комедиях (7); Великобритания - на документальном кино (3) и комедиях (3). Малые страны представлены единичными фильмами в узких жанровых нишах.

Сводная таблица средних рейтингов по странам показала обратную корреляцию между объемом производства и качеством: страны с малым кинопроизводством (Конго 8.2, Украина - 8.1, ЮАР - 8.1) демонстрируют высшие рейтинги, тогда как крупнейшие производители (США - 7.5, Индия - 7.6, Великобритания - 7.6) находятся в середине рейтинга.

Анализ медианных рейтингов по жанрам и странам подтвердил жанровую специализацию: документальное кино имеет стабильно высокие оценки across стран,

тогда как коммерческие жанры варьируются в зависимости от национальных особенностей кинопроизводства.

Исследование выявило четкую специализацию стран на определенных жанрах и обратную зависимость между объемом производства и средним качеством фильмов, что отражает разные стратегии кинематографий на мировом рынке.