

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра Информационной безопасности

ОТЧЕТ
по лабораторной работе № 3
по дисциплине «Распределенные системы обработки данных»
Тема: Использование PySpark для самостоятельной реализации
витрины данных

Студентка гр. 1361

Токарева У.В.

Преподаватель

Троценко В.В.

Санкт-Петербург

2024

Цель работы.

Очистить и загрузить сырые данные. Выделить категориальные переменные. Выделить бинарные признаки. Построить собственную витрину.

Основные теоретические положения.

Категориальные переменные (Categorical Variables) — это переменные, которые принимают одно из нескольких возможных значений, но для каждого наблюдения может быть указано только одно значение. Категориальные переменные удобны для анализа, когда каждая запись может быть отнесена к одной четко определенной категории, что облегчает агрегацию и создание групп по этим переменным.

Бинарные признаки (Binary Features) — это признаки, которые могут быть либо "истинными" (1), либо "ложными" (0) для каждого наблюдения. Они отражают наличие или отсутствие определенных характеристик и позволяют применять несколько признаков к одному объекту. Бинарные признаки удобны для отображения наличия или отсутствия множества характеристик в одном объекте. Они позволяют использовать несколько признаков для одного видео, что делает их гибким инструментом для меток.

Ход работы.

1. В первую очередь мы должны взять данные из предоставленного нами файла. Производим чтение CSV-файла и сохраняем их в новую таблицу:

```
csv_path = "lab3/" + NAME_DATA_CSV
df = spark.read.csv(csv_path, header=True, inferSchema=True, multiLine=True)
df.createOrReplaceTempView("YT_Data")
```

2. Необходимо выделить категориальные переменные. В моем случае категория: 'Трендовые персоны (Илон Маск, 21 Savage, Магнус Карлсен)'. Поэтому определяем структуру для представления категорий видео:

```
schema = StructType([
```

```

        StructField("id", IntegerType(), True),
        StructField("name", StringType(), True),
        StructField("template", StringType(), True)
    ])

```

Определяем категории видео

```

category = [
    (1, "Илон Маск", "%Илон Маск%"),
    (2, "21 Savage", "%21 Savage%"),
    (3, "Магнус Карлсен", "%Магнус Карлсен%")
]

```

3. Создаем статистику по общим показателям видео для каждого месяца и сохраняем результаты во временной таблице:

```

view("df_total_stat", """
    SELECT
        date_format(YT_data.trending_date, 'yyyy-MM')
AS YM,
        SUM(YT_data.view_count) AS total_view,
        SUM(YT_data.comment_count) AS total_comment,
        SUM(YT_data.likes) AS total_likes,
        ROUND      ((SUM(YT_data.comment_count)      /
SUM(YT_data.likes)), 2) AS total_CLR
    FROM YT_Data AS YT_data
    GROUP BY date_format(YT_data.trending_date, 'yyyy-
MM')
""")

```

В данной таблице собрана основная статистика по просмотрам, комментариям, лайкам и коэффициент CLR за каждый месяц. Вывод общей статистики представлен на рисунке 1.

toktalk@LAPTOP-I92UI0JP: ~/leti-spark-course-2024

YM	total_view	total_comment	total_likes	total_CLR
2020-12	3524072622	3.0699365E7	286818208	0.11
2023-08	5468967208	1.5162869E7	305918347	0.05
2024-02	6897478524	1.1091736E7	292969924	0.04
2021-07	4644279727	3.1820584E7	332950747	0.1
2022-04	4681734913	1.6416418E7	258894893	0.06
2023-12	6742291412	1.2821919E7	364242277	0.04
2021-04	3557247089	2.9190677E7	255985391	0.11
2023-11	6084022728	1.2382188E7	314392817	0.04
2022-06	4735167314	2.6830589E7	322408117	0.08
2021-06	6792900896	4.7180891E7	425362051	0.11
2022-02	3944307350	1.776408E7	219736901	0.08
2022-08	5134911223	3.0337922E7	327417897	0.09
2023-07	4340800759	1.3167347E7	280881868	0.05
2021-11	3632109978	1.8646073E7	263235644	0.07
2023-03	4945941220	1.5626489E7	289556726	0.05
2023-10	7298890701	1.3710554E7	350066718	0.04
2021-03	3875417671	3.3547777E7	303693165	0.11
2021-02	3111801392	2.2457705E7	228490913	0.1
2022-07	4917541950	2.0388889E7	312371852	0.07
2023-02	4664164676	1.1944183E7	230182230	0.05

Рисунок 1 – Вывод общей статистики.

4. Далее создаем таблицу со всеми категориями с изначальной таблицы. Рассматриваются совпадения в названиях, описаниях и тегах. Результат обработки сохраняется в таблице:

```
view("df_categories", """
    SELECT
        YT_data.description,YT_data.comment_count,YT_data.likes
        ,YT_data.view_count,YT_data.tags,YT_data.trending_date,
        YT_data.publishedAt,YT_data.title,
            c.id AS cat_id,
            c.name AS cat_name
    FROM YT_Data AS YT_data
    JOIN category AS c
    ON YT_data.description ILIKE c.template
        OR YT_data.title ILIKE c.template
        OR YT_data.tags ILIKE c.template
    """)
```

Вывод таблицы со всеми категориями представлен на рисунке 2.

description	comment_count	likes	view_count	tags	trending_date	publishedAt	title	cat_id	cat_name
Качай Raid: Sha...	5215	156835	928553	Руслан Усачев usa...	2020-08-14 03:00:00	2020-08-13 20:07:47	БЕЛАРУСЬ ПРОТИВ Л...	1	Илон Маск
Качай Raid: Sha...	5867	179888	1161912	Руслан Усачев usa...	2020-08-15 03:00:00	2020-08-13 20:07:47	БЕЛАРУСЬ ПРОТИВ Л...	1	Илон Маск
Качай Raid: Sha...	6114	189914	1254828	Руслан Усачев usa...	2020-08-16 03:00:00	2020-08-13 20:07:47	БЕЛАРУСЬ ПРОТИВ Л...	1	Илон Маск
По вопросам рекла...	331	6894	84977	джонни депп пират...	2020-08-23 03:00:00	2020-08-23 12:00:06	ГРЕБАНЫЙ СТЫД! ДЖ...	1	Илон Маск
По вопросам рекла...	549	11820	192560	джонни депп пират...	2020-08-24 03:00:00	2020-08-23 12:00:06	ГРЕБАНЫЙ СТЫД! ДЖ...	1	Илон Маск
Скачивай Hearthst...	267	0	34618	ШАРЛАТАН ГЕНИЙ ИЛ...	2020-08-25 03:00:00	2020-08-24 12:00:52	ШАРЛАТАН ИЛИ ГЕНИ...	1	Илон Маск
Не сиди без дела,...	1292	19802	142046	what if илон маск...	2020-08-26 03:00:00	2020-08-25 18:16:40	ЧТО, ЕСЛИ ты ИЛОН...	1	Илон Маск
Не сиди без дела,...	1382	22727	174311	what if илон маск...	2020-08-27 03:00:00	2020-08-25 18:16:40	ЧТО, ЕСЛИ ты ИЛОН...	1	Илон Маск
Подписывайся на С...	6952	100420	829406	игры разума откро...	2020-08-29 03:00:00	2020-08-28 19:18:09	Боб покоряет солн...	1	Илон Маск
Подписывайся на С...	7657	112526	1026092	игры разума откро...	2020-08-30 03:00:00	2020-08-28 19:18:09	Боб покоряет солн...	1	Илон Маск
Как можно описать...	286	3786	56419	Бермудский треуго...	2020-08-31 03:00:00	2020-08-30 15:12:16	ЧТО ОБНАРУЖИЛИ ПО...	1	Илон Маск
Подписывайся на С...	7959	118925	1153255	игры разума откро...	2020-08-31 03:00:00	2020-08-28 19:18:09	Боб покоряет солн...	1	Илон Маск
Купили на аукцион...	4426	21569	271988	аукцион контейнер...	2020-08-31 03:00:00	2020-08-30 15:47:18	Нашли аккумулятор...	1	Илон Маск
#невзоровскиесред...	2207	25827	438913	Nevzorov Новости	2020-09-03 03:00:00	2020-09-02 22:14:37	Невзоров / Невзор...	1	Илон Маск
Покупайте новую к...	1264	28670	251537	кузьма юлик антон...	2020-09-04 03:00:00	2020-09-03 14:00:27	БАЛДЁЖНЫЙ ПОДКАСТ...	1	Илон Маск
Таймкоды: 00:00 В...	2024	9788	230738	Артемиий Лебедев Л...	2020-09-04 03:00:00	2020-09-03 23:50:09	Подробности отрав...	1	Илон Маск
#невзоровскиесред...	2631	29201	540537	Nevzorov Новости	2020-09-04 03:00:00	2020-09-02 22:14:37	Невзоров / Невзор...	1	Илон Маск
Германия заявила,...	3829	36506	660217	варламов чп чб пр...	2020-09-07 03:00:00	2020-09-06 13:52:42	Чб Происходит #25...	1	Илон Маск
Выбери новую проф...	533	5499	60738	Кик Обзор Кик Бре...	2020-09-09 03:00:00	2020-09-08 16:31:59	Starlink уже реал...	1	Илон Маск
Подпишись - htt...	250	4237	55777	Плеяды звёздное с...	2020-09-20 03:00:00	2020-09-19 16:46:27	ЧТО СКРЫВАЮТ ПЛЕЯ...	1	Илон Маск

Рисунок 2 – Вывод таблицы со всеми категориями.

5. Далее создаем статистику для категорий по просмотрам, комментариям и лайкам:

```
view("df_cats_stat", """)
SELECT
    date_format(a.trending_date, 'yyyy-MM') AS YM,
    SUM(a.view_count) AS cats_view,
    SUM(a.comment_count) AS cats_comment,
    SUM(a.likes) AS cats_likes,
    a.cat_id,
    a.cat_name
FROM df_categoryes AS a
GROUP BY a.cat_id, a.cat_name,
date_format(a.trending_date, 'yyyy-MM')
""")
```

Вывод таблицы со статистикой по категориям представлен на рисунке 3.

toktalk@LAPTOP-I92UI0JP: ~/leti-spark-course-2024

YM	cats_view	cats_comment	cats_likes	cat_id	cat_name
2022-07	8786736	41075.0	534076	1	Илон Маск
2022-12	21551069	93777.0	1013365	1	Илон Маск
2022-11	20425282	72519.0	882849	1	Илон Маск
2023-09	2698295	9043.0	136861	1	Илон Маск
2020-09	2238457	12738.0	139728	1	Илон Маск
2021-01	3075220	18998.0	221970	1	Илон Маск
2022-02	4366644	18902.0	374159	1	Илон Маск
2020-11	1433003	10163.0	198175	1	Илон Маск
2021-07	3977970	26615.0	204058	1	Илон Маск
2022-05	19643462	107779.0	991697	1	Илон Маск
2022-06	9830552	53835.0	608053	1	Илон Маск
2020-08	7310965	48297.0	945106	1	Илон Маск
2023-02	4064744	25418.0	233757	1	Илон Маск
2021-03	7017350	55633.0	513564	1	Илон Маск
2024-01	2122229	1537.0	121483	1	Илон Маск
2020-12	2934959	20922.0	338903	1	Илон Маск
2021-06	16436199	77465.0	1806276	1	Илон Маск
2023-08	3923772	32121.0	241393	1	Илон Маск
2021-11	2467697	8890.0	255567	1	Илон Маск
2023-12	13865888	21399.0	818103	1	Илон Маск

Рисунок 3 – Вывод таблицы со статистикой по категориям.

6. Далее делаем сборку финальной таблицы для категорий:

```
view("df_final_cats_stat", """
WITH sub AS (
SELECT t.YM,
      t.total_comment,
      t.total_likes,
      t.total_view,
      t.total_CLR,

      c.cat_name,
      c.cats_view,
      c.cats_comment,
      c.cats_likes,
      ROUND ((c.cats_comment / c.cats_likes), 2) AS
cats_CLR
FROM df_total_stat AS t
JOIN df_cats_stat AS c
```

```

ON t.YM = c.YM

ORDER BY t.YM DESC, c.cat_id ASC

)

SELECT *,

        ROUND((total_CLR      -      cats_CLR),      2)      AS

diff_CLR_abs,

        CASE

                WHEN cats_CLR != 0

                        THEN ROUND((total_CLR / cats_CLR), 2)

                        ELSE NULL

                END AS diff_CLR_relative

FROM sub

""")

```

Вывод финальной таблицы для категорий представлен на рисунке 4.

YM	total_comment	total_likes	total_view	total_CLR	cat_name	cats_view	cats_comment	cats_likes	cats_CLR	diff_CLR_abs	diff_CLR_relative
2024-03	9218794.0	351500982	8676586316	0.03	Илон Маск	13791788	56576.0	579172	0.1	-0.07	0.3
2024-02	1.1091736E7	292969924	6897478524	0.04	Илон Маск	10456901	38302.0	514000	0.07	-0.03	0.57
2024-01	9150624.0	271780911	6060824937	0.03	Илон Маск	2122229	1537.0	121483	0.01	0.02	3.0
2023-12	1.2821919E7	364242277	6742291412	0.04	Илон Маск	13865888	21399.0	818103	0.03	0.01	1.33
2023-11	1.2382188E7	314392817	6084022728	0.04	Илон Маск	1033966	3609.0	50384	0.07	-0.03	0.57
2023-10	1.3710554E7	350066718	7298890701	0.04	Илон Маск	8631473	50838.0	364512	0.14	-0.1	0.29
2023-09	1.1577666E7	301046210	5963195144	0.04	Илон Маск	2698295	9043.0	136861	0.07	-0.03	0.57
2023-08	1.5162869E7	305918347	5468967208	0.05	Илон Маск	3923772	32121.0	241393	0.13	-0.08	0.38
2023-08	1.5162869E7	305918347	5468967208	0.05	Магнус Карлсен	451348	182.0	16339	0.01	0.04	5.0
2023-07	1.3167347E7	280881868	4340800759	0.05	Илон Маск	8544303	54794.0	696039	0.08	-0.03	0.63
2023-06	1.4849175E7	272594274	5153586714	0.05	Илон Маск	10699341	129006.0	619005	0.21	-0.16	0.24
2023-05	1.2243495E7	273610800	5281841221	0.04	Илон Маск	5414009	31273.0	348292	0.09	-0.05	0.44
2023-05	1.2243495E7	273610800	5281841221	0.04	Магнус Карлсен	2217571	875.0	98808	0.01	0.03	4.0
2023-04	1.5712181E7	282561679	4998614296	0.06	Илон Маск	24457324	179058.0	869315	0.21	-0.15	0.29
2023-04	1.5712181E7	282561679	4998614296	0.06	Магнус Карлсен	23442292	160937.0	731488	0.22	-0.16	0.27
2023-03	1.5626489E7	289556726	4945941220	0.05	Илон Маск	6445182	23416.0	566580	0.04	0.01	1.25
2023-02	1.1944183E7	230182230	4664164676	0.05	Илон Маск	4064744	25418.0	233757	0.11	-0.06	0.45
2023-01	1.2962921E7	228015517	4394086204	0.06	Илон Маск	4131615	11976.0	152807	0.08	-0.02	0.75
2022-12	1.5069554E7	286671504	5078994752	0.05	Илон Маск	21551069	93777.0	1013365	0.09	-0.04	0.56
2022-11	1.3361672E7	250505617	4596294902	0.05	Илон Маск	20425282	72519.0	882849	0.08	-0.03	0.63

Рисунок 4 – Вывод финальной таблицы для категорий.

7. Создаем таблицу с бинарными признаками, отдельно рассматривая для каждого признака и той ситуации, когда они встречаются вместе:

```

view("df_bins_challenge", "")

SELECT

YT_data.description, YT_data.comment_count, YT_data.likes

```

```
,YT_data.view_count,YT_data.tags,YT_data.trending_date,
YT_data.publishedAt,YT_data.title
    FROM YT_Data AS YT_data
    WHERE (YT_data.description ILIKE '%#challenge%'
        OR YT_data.tags ILIKE '%challenge%')
        AND (YT_data.description NOT ILIKE '%#asmr%'
        OR YT_data.tags NOT ILIKE '%asmr%')

    """)
```

```
view("df_bins_asmr", """)
    SELECT
YT_data.description,YT_data.comment_count,YT_data.likes
,YT_data.view_count,YT_data.tags,YT_data.trending_date,
YT_data.publishedAt,YT_data.title
    FROM YT_Data AS YT_data
    WHERE (YT_data.description ILIKE '%#asmr%'
        OR YT_data.tags ILIKE '%asmr%')
        AND (YT_data.description NOT ILIKE '%#challenge%'
        OR YT_data.tags NOT ILIKE '%challenge%')

    """)
```

```
view("df_bins_chandas", """)
    SELECT
YT_data.description,YT_data.comment_count,YT_data.likes
,YT_data.view_count,YT_data.tags,YT_data.trending_date,
YT_data.publishedAt,YT_data.title
    FROM YT_Data AS YT_data
    WHERE (YT_data.description ILIKE '%#challenge%'
        OR YT_data.tags ILIKE '%challenge%')
```



```

AND (YT_data.description ILIKE '%#asmr%'
OR YT_data.tags ILIKE '%asmr%')
""")

```

Вывод таблицы с бинарным признаком «challenge» с изначальной таблицей представлен на рисунке 5.

description	comment_count	likes	view_count	tags	trending_date	publishedAt	title
Самый сложный чел...	1057	10272	72790	Как похудеть как ...	2020-08-13 03:00:00	2020-08-12 14:49:38	КАК Я ПОХУДЕЛА ЗА...
ЭТО по-настоящему...	13184	281865	2228152	КТО ВЫЖИВЕТ НА ОТ...	2020-08-14 03:00:00	2020-08-13 16:00:11	КТО ВЫЖИВЕТ НА ОТ...
Соня и разносчица...	0	46496	585215	Непета Непета Стр...	2020-08-14 03:00:00	2020-08-14 10:00:11	СИРЕНОГОЛОВЫЕ ПРО...
В кадре — команда...	1322	29505	256457	Реакция NAVI CSGO...	2020-08-14 03:00:00	2020-08-13 16:05:16	Реакция NAVI CSGO...
Соня и разносчица...	0	56312	1005383	Непета Непета Стр...	2020-08-15 03:00:00	2020-08-14 10:00:11	СИРЕНОГОЛОВЫЕ ПРО...
ЭТО по-настоящему...	14508	317794	2746499	КТО ВЫЖИВЕТ НА ОТ...	2020-08-15 03:00:00	2020-08-13 16:00:11	КТО ВЫЖИВЕТ НА ОТ...
Вы готовы к новом...	97	3105	187964	123go челлендж 12...	2020-08-15 03:00:00	2020-08-14 17:00:00	ЧЕЛЛЕНДЖ «МАЛЕНЬК...
Подписывайся на м...	6471	101163	1112021	Купить ПОНТОРЕЗКУ...	2020-08-16 03:00:00	2020-08-15 19:29:14	Купить ПОНТОРЕЗКУ...
Соня и разносчица...	0	59450	1140056	Непета Непета Стр...	2020-08-16 03:00:00	2020-08-14 10:00:11	СИРЕНОГОЛОВЫЕ ПРО...
ЭТО по-настоящему...	15187	337087	3026862	КТО ВЫЖИВЕТ НА ОТ...	2020-08-16 03:00:00	2020-08-13 16:00:11	КТО ВЫЖИВЕТ НА ОТ...
Жеребьевка турнир...	69	5742	33197	FIFA FIFA 17 FIFA...	2020-08-17 03:00:00	2020-08-17 11:07:51	МЫСЛИТЬ КАК ПОДЛИ...
Подписывайся на м...	7237	115705	1377461	Купить ПОНТОРЕЗКУ...	2020-08-17 03:00:00	2020-08-15 19:29:14	Купить ПОНТОРЕЗКУ...
Соня и разносчица...	0	62133	1223877	Непета Непета Стр...	2020-08-17 03:00:00	2020-08-14 10:00:11	СИРЕНОГОЛОВЫЕ ПРО...
Dream Team Family...	4372	64492	450557	дрим тим хаус дри...	2020-08-18 03:00:00	2020-08-17 15:00:06	РЕАКЦИЯ НА УЖАСНЫ...
Внезапно захотело...	134	3484	202515	123go челлендж 12...	2020-08-19 03:00:00	2020-08-19 06:00:05	ЧЕЛЛЕНДЖ С СИНЕЙ ...
ROMAROY https://w...	2460	11053	89143	ФУТБОЛ FOOTBALL Г...	2020-08-20 03:00:00	2020-08-20 10:59:17	ЛУЧШИЙ МАТЧ КУБКА...
Внезапно захотело...	200	4592	338084	123go челлендж 12...	2020-08-20 03:00:00	2020-08-19 06:00:05	ЧЕЛЛЕНДЖ С СИНЕЙ ...
Слава:Инстаграм: ...	135	1864	23327	КТО ВЫЖИВЕТ НА ОТ...	2020-08-20 03:00:00	2020-08-19 19:00:28	КАК РАЗРЕЗАТЬ АРБ...
Обязательно подп...	4630	50996	278395	ДЕНЬГИ ВЗЛОМАЕШЬ ...	2020-08-21 03:00:00	2020-08-21 11:30:11	ДЕНЬГИ или ВЗЛОМА...
ROMAROY https://w...	4337	13051	129866	ФУТБОЛ FOOTBALL Г...	2020-08-21 03:00:00	2020-08-20 10:59:17	ЛУЧШИЙ МАТЧ КУБКА...

Рисунок 5 – Вывод таблицы с бинарным признаком «challenge» с изначальной таблицей.

Вывод таблицы с бинарным признаком «asmr» с изначальной таблицей представлен на рисунке 6.

description	comment_count	likes	view_count	tags	trending_date	publishedAt	title
Хочешь знать, что...	1125	9057	103667	асмп асмп для сна...	2020-08-12 03:00:00	2020-08-11 19:30:03	АСМР КАК ИСПЫТАТЬ...
Спасибо за просмо...	999	3817	56286	АСМР АСМР Расслаб...	2020-08-14 03:00:00	2020-08-13 19:56:02	АСМР LICKING MARV...
Спасибо за просмо...	1236	5060	86494	АСМР АСМР Расслаб...	2020-08-15 03:00:00	2020-08-13 19:56:02	АСМР LICKING MARV...
В этом видео я от...	1157	14291	145195	Вопрос Ответ АСМР...	2020-08-15 03:00:00	2020-08-14 18:17:03	АСМР 100+ ВОПРОСО...
Thanks for watchi...	44	2587	317726	KLuna Tik KLuna T...	2020-08-18 03:00:00	2020-08-17 16:00:12	KLUNA TIK Destroy...
Внезапно захотело...	134	3484	202515	123go челлендж 12...	2020-08-19 03:00:00	2020-08-19 06:00:05	ЧЕЛЛЕНДЖ С СИНЕЙ ...
Внезапно захотело...	200	4592	338084	123go челлендж 12...	2020-08-20 03:00:00	2020-08-19 06:00:05	ЧЕЛЛЕНДЖ С СИНЕЙ ...
Во время просмотр...	517	5964	57995	асмп асмп мурашки...	2020-08-22 03:00:00	2020-08-21 18:00:09	Мурашки по коже о...
Во время просмотр...	600	7190	75182	асмп асмп мурашки...	2020-08-23 03:00:00	2020-08-21 18:00:09	Мурашки по коже о...
♠ КУПИТЬ МОЙ МЕРЧ...	605	2616	43455	асмп асмп relax w...	2020-08-25 03:00:00	2020-08-24 19:00:00	АСМР НЕДОВОЛЬНЫЙ ...
♠ КУПИТЬ МОЙ МЕРЧ...	788	3546	72846	асмп асмп relax w...	2020-08-26 03:00:00	2020-08-24 19:00:00	АСМР НЕДОВОЛЬНЫЙ ...
Сегодня тебя ждет...	1277	15751	192603	АСМР для сна АСМР...	2020-09-01 03:00:00	2020-08-31 18:02:24	АСМР ТЕСТ [Audio-...
Сегодня тебя ждет...	1650	21411	304251	АСМР для сна АСМР...	2020-09-02 03:00:00	2020-08-31 18:02:24	АСМР ТЕСТ [Audio-...
[ASMR] Во время п...	282	3598	29569	асмп асмп асмп дл...	2020-09-05 03:00:00	2020-09-04 19:31:58	АСМР [ASMR] 100% ...
Hello friends! Ap...	15789	149720	1049185	I tried acrylic p...	2020-09-06 03:00:00	2020-09-06 04:53:34	I Tried Acrylic P...
[ASMR] Во время п...	347	4874	48392	асмп асмп асмп дл...	2020-09-06 03:00:00	2020-09-04 19:31:58	АСМР [ASMR] 100% ...
Hello friends! Ap...	21212	231120	2377984	I tried acrylic p...	2020-09-07 03:00:00	2020-09-06 04:53:34	I Tried Acrylic P...
Как приготовить с...	1435	31218	353950	rfr как приготови...	2020-09-08 03:00:00	2020-09-08 12:00:47	ГОВЯЖЬЕ СЕРДЦЕ в ...
Как приготовить с...	2359	51291	901422	rfr как приготови...	2020-09-09 03:00:00	2020-09-08 12:00:47	ГОВЯЖЬЕ СЕРДЦЕ в ...
Как приготовить с...	2674	59065	1178107	rfr как приготови...	2020-09-10 03:00:00	2020-09-08 12:00:47	ГОВЯЖЬЕ СЕРДЦЕ в ...

Рисунок 6 – Вывод таблицы с бинарным признаком «asmr» с изначальной таблицей.

Вывод таблицы с бинарными признаками с изначальной таблицей представлен на рисунке 7.

[illegible]

Рисунок 7 – Вывод таблицы с бинарными признаками с изначальной таблицей.

8. Создаем статистику для бинарных признаков:

```
view("df_bins_stat_challenge", ""
    SELECT SUM(a.likes) AS bins_likes,
           SUM(a.comment_count) AS bins_comment,
           SUM(a.view_count) AS bins_view,
           date_format(a.trending_date, 'yyyy-MM') AS YM
    FROM df_bins_challenge AS a
    GROUP BY date_format(a.trending_date, 'yyyy-MM')
    """)
```

```
view("df_bins_stat_asmr", ""
    SELECT SUM(a.likes) AS bins_likes,
           SUM(a.comment_count) AS bins_comment,
           SUM(a.view_count) AS bins_view,
           date_format(a.trending_date, 'yyyy-MM')AS YM
FROM df_bins_asmr AS a
GROUP BY date_format(a.trending_date, 'yyyy-MM')
""")
```


```
view("df bins stat chandas", "")
```

```

SELECT SUM(a.likes) AS bins_likes,
       SUM(a.comment_count) AS bins_comment,
       SUM(a.view_count) AS bins_view,
       date_format(a.trending_date, 'yyyy-MM') AS YM
FROM df_bins_chandas AS a
GROUP BY date_format(a.trending_date, 'yyyy-MM')
""")

```

Вывод таблицы статистики для бинарного признака «challenge» представлен на рисунке 8.

 toktalk@LAPTOP-I92UI0JP: ~/leti-spark-course-2024

bins_likes	bins_comment	bins_view	YM
10462419	714363.0	104271164	2020-12
2846149	46318.0	57730238	2023-08
7519272	116776.0	329973126	2024-02
21622837	451671.0	246460367	2021-07
6834511	170810.0	150237564	2022-04
10945070	90271.0	399940626	2023-12
4111545	246566.0	79805061	2021-04
5807445	145414.0	184340548	2023-11
3031466	92283.0	93863097	2022-06
15150929	309776.0	479480365	2021-06
7333679	596288.0	99873263	2022-02
7516061	117750.0	148262084	2022-08
7178076	156710.0	131998799	2023-07
10021541	313714.0	186404722	2021-11
7806492	153599.0	243654860	2023-03
6627626	68053.0	230588443	2023-10
14480102	724373.0	128532182	2021-03
7659101	467662.0	66012398	2021-02
10136750	190375.0	308453523	2022-07
5059505	278507.0	122884379	2023-02

Рисунок 8 – Вывод статистики для «challenge».

Вывод таблицы статистики для бинарного признака «asmr» представлен на рисунке 9.

toktalk@LAPTOP-I92UI0JP: ~/leti-spark-course-2024

bins_likes	bins_comment	bins_view	YM
806898	195182.0	25479054	2020-12
972978	9812.0	21934161	2023-08
372812	3710.0	16948984	2024-02
139422	9782.0	2793246	2021-07
19753	1223.0	197798	2022-04
981541	8305.0	40940254	2023-12
185301	11669.0	3683672	2021-04
3050175	8195.0	97983460	2023-11
145550	7799.0	4030143	2022-06
102886	6443.0	5274694	2021-06
790544	13393.0	16592853	2022-02
109548	4806.0	3001010	2022-08
1376859	13366.0	24240074	2023-07
1143205	30478.0	24852924	2021-11
4612315	28594.0	79240618	2023-03
3746006	22475.0	150644595	2023-10
171147	14773.0	3023245	2021-03
66368	6814.0	834353	2021-02
114734	5675.0	2710350	2022-07
7980218	191878.0	111873187	2023-02

Рисунок 9 – Вывод статистики для «asmr».

Вывод таблицы статистики для бинарных признаков представлен на рисунке 10.

toktalk@LAPTOP-I92UI0JP: ~/leti-spark-course-2024

bins_likes	bins_comment	bins_view	YM
109887	1791.0	1413359	2024-02
360038	2405.0	21008692	2023-12
1026287	1570.0	40799257	2023-11
117149	6387.0	1445523	2022-06
24804	663.0	3677081	2021-06
40748	32.0	2716941	2023-07
780131	401.0	13885801	2021-11
320574	2774.0	23666501	2023-03
156619	143.0	7209875	2023-10
375651	320.0	7203295	2021-10
72129	17989.0	661959	2020-11
84853	485.0	3046824	2023-09
80524	99.0	7407704	2023-04
104972	95.0	6544252	2023-05
47124	193.0	3776090	2023-01
479495	1470.0	29808958	2024-03
105407	281.0	7103180	2023-06
8076	334.0	540599	2020-08
936580	2361.0	45754954	2024-04
500451	2286.0	7088648	2021-09

Рисунок 10 – Вывод статистики для бинарных признаков.

9. Делаем сборку статистики для бинарных признаков:

```
view("df_total_bins_stat", """
    SELECT chandas.YM,
           challenge.bins_likes AS challenge_likes,
           challenge.bins_comment AS challenge_comment,
           challenge.bins_view AS challenge_view,

           ROUND      ((challenge.bins_comment      /
challenge.bins_likes ), 2) AS challenge_CLR,

           asmr.bins_likes AS asmr_likes,
           asmr.bins_comment AS asmr_comment,
           asmr.bins_view AS asmr_view,

           ROUND      ((asmr.bins_comment / asmr.bins_likes),
2) AS asmr_CLR,

           chandas.bins_likes AS chandas_likes,
           chandas.bins_comment AS chandas_comment,
           chandas.bins_view AS chandas_view,

           ROUND      ((chandas.bins_comment      /
chandas.bins_likes), 2) AS chandas_CLR

    FROM df_bins_stat_chandas AS chandas
    JOIN df_bins_stat_asmr AS asmr ON chandas.YM =
asmr.YM
    JOIN df_bins_stat_challenge AS challenge ON
chandas.YM = challenge.YM
""")
```

Вывод сборки статистики для бинарных признаков представлен на рисунке 11.

YM	challenge_likes	challenge_comment	challenge_view	challenge_CLR	asmr_likes	asmr_comment	asmr_view	asmr_CLR	chandas_likes	chandas_comment	chandas_view	chandas_CLR
2024-02	7519272	116776.0	329973126	0.02	372812	3710.0	16948984	0.01	109887	1791.0	1413359	0.02
2023-12	10945070	98271.0	399940626	0.01	981541	8305.0	40940254	0.01	360038	2405.0	21008692	0.01
2023-11	5807445	145414.0	184340548	0.03	3050175	8195.0	97983460	0.0	1026287	1570.0	40799257	0.0
2022-06	3031466	92283.0	93863097	0.03	145550	7799.0	4030143	0.05	117149	6387.0	1445523	0.05
2021-06	15150929	309776.0	479480365	0.02	102886	6443.0	5274694	0.06	24804	663.0	3677081	0.03
2023-07	7178076	156710.0	131998799	0.02	1376859	13366.0	24240074	0.01	40748	32.0	2716941	0.0
2021-11	10021541	313714.0	186404722	0.03	1143205	30478.0	24852924	0.03	780131	401.0	13885801	0.0
2023-03	7806492	153599.0	243654860	0.02	4612315	28594.0	79240618	0.01	320574	2774.0	23666501	0.01
2023-10	6627626	68053.0	230588443	0.01	3746006	22475.0	150644595	0.01	156619	143.0	7209875	0.0
2021-10	14801569	362706.0	203672229	0.02	516629	6440.0	27130424	0.01	375651	320.0	7203295	0.0
2020-11	2050806	254963.0	21128326	0.12	742397	60237.0	35456742	0.08	72129	17989.0	661959	0.25
2023-09	2206797	23226.0	93495890	0.01	989793	12566.0	38835217	0.01	84853	485.0	3046824	0.01
2023-04	11246955	235054.0	275995270	0.02	1399815	7504.0	25215741	0.01	80524	99.0	7407704	0.0
2023-05	6004279	43659.0	179958384	0.01	512853	5188.0	14187719	0.01	104972	95.0	6544252	0.0
2023-01	8342470	78480.0	240475832	0.01	5660338	43529.0	75418502	0.01	47124	193.0	3776090	0.0
2024-03	7006495	216475.0	277495149	0.03	5417774	21397.0	221344128	0.0	479495	1470.0	29808958	0.0
2023-06	6654362	155217.0	166299081	0.02	2201740	7114.0	58434370	0.0	105407	281.0	7103180	0.0
2020-08	1850965	93238.0	19876769	0.05	62204	7405.0	1499445	0.12	8076	334.0	540599	0.04
2024-04	7007256	10636.0	259298410	0.0	1150379	10375.0	48465618	0.01	936580	2361.0	45754954	0.0
2021-09	7293034	287000.0	97063473	0.04	1024458	10696.0	15580880	0.01	500451	2286.0	7088648	0.0

Рисунок 11 – Вывод сборки статистики для бинарных признаков.

10. Делаем сборку общей статистики для бинарных признаков:

```
view("df_final_bins_stat", "")
WITH sub AS (
    SELECT t.YM,
           t.total_comment,
           t.total_likes,
           t.total_view,
           ROUND ((t.total_comment / t.total_likes), 2)
    AS total_CLR,
           b.challenge_likes,
           b.challenge_comment,
           b.challenge_view,
           b.challenge_CLR,
           b.asmr_likes,
           b.asmr_comment,
           b.asmr_view,
           b.asmr_CLR,
           b.chandas_likes,
           b.chandas_comment,
           b.chandas_view,
```


В результате были получены 2 таблицы: для категориальных переменных и для бинарных признаков.

Исходный код программы.

```
from pyspark.sql import DataFrame, SparkSession
from pyspark.sql.functions import *
from pyspark.sql.types import *
from pyspark.shell import spark

# Определяем переменную для названия таблицы CSV-файла
NAME_DATACSV = "RU_youtube_trending_data.csv" #
Название      таблицы      CSV-файла

def view(name: str, target: DataFrame | str) ->
DataFrame:
    """Создает или заменяет временную представительскую
таблицу."""
    df = target if isinstance(target, DataFrame) else
spark.sql(target)
    df.createOrReplaceTempView(name)
    return df

#      Создаем      экземпляр      SparkSession
spark =
SparkSession.builder.appName("MyApp").getOrCreate()

# Указываем путь к CSV-файлу и читаем его в Dataframe
csv_path = "lab3/" + NAME_DATACSV
df = spark.read.csv(csv_path, header=True,
inferSchema=True, multiLine=True)
```



```

df.createOrReplaceTempView("YT_Data")

# Бинарный признак (надо выбрать 2 признака из группы):
'Популярные теги (#challenge, #asmr)';
# Категория: 'Трендовые персоны (Илон Маск, 21 Savage,
Магнус Карлсен)';
# Коэффициент 'CLR - Процент комментариев от общего
числа лайков'

# Определяем схему для категорий видео
schema = StructType([
    StructField("id", IntegerType(), True),
    StructField("name", StringType(), True),
    StructField("template", StringType(), True)
])

# Определяем категории видео
category = [
    (1, "Илон Маск", "%Илон Маск%"),
    (2, "21 Savage", "%21 Savage%"),
    (3, "Магнус Карлсен", "%Магнус Карлсен%")
]

# Создаем DataFrame для категорий видео
categorical_df = spark.createDataFrame(category,
schema)
categorical_df.createOrReplaceTempView("category")

# -----создание общей статистики -----
-----

```

```

view("df_total_stat",
    SELECT
        date_format(YT_data.trending_date, 'yyyy-MM')
AS
        SUM(YT_data.view_count) AS total_view,
        SUM(YT_data.comment_count) AS total_comment,
        SUM(YT_data.likes) AS total_likes,
        ROUND ((SUM(YT_data.comment_count) /
SUM(YT_data.likes)), 2) AS total_CLR
        FROM YT_Data AS YT_data
        GROUP BY date_format(YT_data.trending_date, 'yyyy-
MM')
    """)

# ----- создание таблицы со всеми категориями с
изначальной таблицой -----
view("df_categoryes",
    SELECT
YT_data.description, YT_data.comment_count, YT_data.likes
, YT_data.view_count, YT_data.tags, YT_data.trending_date,
YT_data.publishedAt, YT_data.title,
        c.id AS cat_id,
        c.name AS cat_name
        FROM YT_Data AS YT_data
        JOIN category AS c
        ON YT_data.description ILIKE c.template
        OR YT_data.title ILIKE c.template
        OR YT_data.tags ILIKE c.template
    """)

```

```

# ----- создание статистики для категорий ---
-----
view("df_cats_stat",
    SELECT
        date_format(a.trending_date, 'yyyy-MM') AS YM,
        SUM(a.view_count) AS cats_view,
        SUM(a.comment_count) AS cats_comment,
        SUM(a.likes) AS cats_likes,
        a.cat_id,
        a.cat_name
    FROM df_categoryes AS a
    GROUP BY a.cat_id, a.cat_name,
date_format(a.trending_date, 'yyyy-MM')
)

```

```

# ----- сборка таблицы для категорий -----
view("df_final_cats_stat",
    WITH sub AS (
    SELECT
        t.YM,
        t.total_comment,
        t.total_likes,
        t.total_view,
        t.total_CLR,
        c.cat_name,
        c.cats_view,
        c.cats_comment,
        c.cats_likes,
        ROUND ((c.cats_comment / c.cats_likes), 2) AS

```

```

cats_CLR
FROM          df_total_stat          AS          t
JOIN          df_cats_stat          AS          c
ON            t.YM                    =          c.YM
ORDER        BY          t.YM        DESC,        c.cat_id        ASC
)
SELECT                                                *,
              ROUND((total_CLR        -        cats_CLR),        2)        AS
diff_CLR_abs,
              CASE
                  WHEN          cats_CLR          !=          0
                  THEN  ROUND((total_CLR  /  cats_CLR),  2)
                  ELSE          NULL
                  END          AS          diff_CLR_relative
FROM          sub
""")

```

```

# Показываем содержимое конечного DataFrame с
категориями

```

```

spark.sql("SELECT * FROM df_total_stat").show()
spark.sql("SELECT * FROM df_categoryes").show()
spark.sql("SELECT * FROM df_cats_stat").show()
spark.sql("SELECT * FROM df_final_cats_stat").show()

```

```

# Бинарный признак (надо выбрать 2 признака из группы):
'Популярные теги (#challenge, #asmr)';
# ----- создание таблицы с бинарными признаками с
изначальной таблицы -----
view("df_bins_challenge",

```

```

SELECT
YT_data.description,YT_data.comment_count,YT_data.likes
,YT_data.view_count,YT_data.tags,YT_data.trending_date,
YT_data.publishedAt,YT_data.title
FROM          YT_Data          AS          YT_data
WHERE  (YT_data.description  ILIKE  '%#challenge%'
OR      YT_data.tags        ILIKE  '%challenge%')
AND    (YT_data.description  NOT  ILIKE  '%#asmr%'
OR      YT_data.tags        NOT  ILIKE  '%asmr%')

""")

```

```

view("df_bins_asmr",          """)
SELECT
YT_data.description,YT_data.comment_count,YT_data.likes
,YT_data.view_count,YT_data.tags,YT_data.trending_date,
YT_data.publishedAt,YT_data.title
FROM          YT_Data          AS          YT_data
WHERE  (YT_data.description  ILIKE  '%#asmr%'
OR      YT_data.tags        ILIKE  '%asmr%')
AND    (YT_data.description  NOT  ILIKE
'%#challenge%'
OR      YT_data.tags        NOT  ILIKE  '%challenge%')

""")

```

```

view("df_bins_chandas",          """)
SELECT
YT_data.description,YT_data.comment_count,YT_data.likes
,YT_data.view_count,YT_data.tags,YT_data.trending_date,
YT_data.publishedAt,YT_data.title

```

```

FROM          YT_Data          AS          YT_data
WHERE  (YT_data.description  ILIKE  '%#challenge%'
        OR  YT_data.tags      ILIKE  '%challenge%')
AND    (YT_data.description  ILIKE  '%#asmr%'
        OR  YT_data.tags      ILIKE  '%asmr%')
""")

```

```

#  -----  создание  статистики  для  бинарных
признаков  -----

```

```

view("df_bins_stat_challenge",          """)
    SELECT          SUM(a.likes)          AS          bins_likes,
        SUM(a.comment_count)          AS          bins_comment,
        SUM(a.view_count)          AS          bins_view,
        date_format(a.trending_date,  'yyyy-MM')AS  YM
FROM          df_bins_challenge          AS          a
GROUP  BY  date_format(a.trending_date,  'yyyy-MM')
""")

```

```

view("df_bins_stat_asmr",          """)
    SELECT          SUM(a.likes)          AS          bins_likes,
        SUM(a.comment_count)          AS          bins_comment,
        SUM(a.view_count)          AS          bins_view,
        date_format(a.trending_date,  'yyyy-MM')AS  YM
FROM          df_bins_asmr          AS          a
GROUP  BY  date_format(a.trending_date,  'yyyy-MM')
""")

```

```

view("df_bins_stat_chandas",          """)
    SELECT          SUM(a.likes)          AS          bins_likes,

```

```

SUM(a.comment_count)      AS      bins_comment,
SUM(a.view_count)         AS      bins_view,
date_format(a.trending_date, 'yyyy-MM') AS  YM
FROM      df_bins_chandas      AS      a
GROUP BY  date_format(a.trending_date, 'yyyy-MM')
""")

```

```

# ----- Сборка статистики для бинарных
признаков -----

```

```

view("df_total_bins_stat",      """)

```

```

SELECT      chandas.YM,
challenge.bins_likes      AS      challenge_likes,
challenge.bins_comment    AS      challenge_comment,
challenge.bins_view       AS      challenge_view,

```

```

ROUND      ((challenge.bins_comment      /
challenge.bins_likes      ), 2)      AS      challenge_CLR,

```

```

asmr.bins_likes      AS      asmr_likes,
asmr.bins_comment    AS      asmr_comment,
asmr.bins_view       AS      asmr_view,

```

```

ROUND      ((asmr.bins_comment / asmr.bins_likes),
2)      AS      asmr_CLR,

```

```

chandas.bins_likes      AS      chandas_likes,
chandas.bins_comment    AS      chandas_comment,
chandas.bins_view       AS      chandas_view,

```

```

ROUND      ((chandas.bins_comment      /

```

```

chandas.bins_likes),          2)          AS          chandas_CLR

FROM          df_bins_stat_chandas          AS          chandas
JOIN  df_bins_stat_asmr  AS  asmr  ON  chandas.YM =
asmr.YM
JOIN  df_bins_stat_challenge  AS  challenge  ON
chandas.YM          =          challenge.YM
""")

# ----- Сборка общей статистики для бинарных
признаков -----
view("df_final_bins_stat",          """)
WITH          sub          AS          (
SELECT          t.YM,
                t.total_comment,
                t.total_likes,
                t.total_view,
                ROUND ((t.total_comment / t.total_likes), 2)
AS          total_CLR,
                b.challenge_likes,
                b.challenge_comment,
                b.challenge_view,
                b.challenge_CLR,
                b.asmr_likes,
                b.asmr_comment,
                b.asmr_view,
                b.asmr_CLR,
                b.chandas_likes,
                b.chandas_comment,
                b.chandas_view,

```



```

        b.chandas_CLR

FROM          df_total_stat          AS          t
JOIN          df_total_bins_stat      AS          b
ON            t.YM                    =          b.YM
ORDER        BY                      t.YM        DESC)
SELECT
        *,
        ROUND((total_CLR - challenge_CLR), 2) AS
diff_CLR_abs_ch,
        ROUND((total_CLR - asmr_CLR), 2) AS
diff_CLR_abs_as,
        ROUND((total_CLR - chandas_CLR), 2) AS
diff_CLR_abs_chas,
        ROUND((total_CLR / challenge_CLR), 2) AS
diff_CLR_relative_ch,
        ROUND((total_CLR / asmr_CLR), 2) AS
diff_CLR_relative_as,
        ROUND((total_CLR / chandas_CLR), 2) AS
diff_CLR_relative_chas
FROM          sub
""")

```

Показываем содержимое конечного DataFrame с бинарными признаками

```

spark.sql("SELECT * FROM df_bins_challenge").show()
spark.sql("SELECT * FROM df_bins_asmr").show()
spark.sql("SELECT * FROM df_bins_chandas").show()
spark.sql("SELECT * FROM df_bins_stat_challenge").show()
spark.sql("SELECT * FROM df_bins_stat_asmr").show()

```

```
spark.sql("SELECT * FROM df_bins_stat_chandas").show()
spark.sql("SELECT * FROM df_total_bins_stat").show()
spark.sql("SELECT * FROM df_final_bins_stat").show()

# Завершаем работу с SparkSession
spark.stop()
```