8. Курсовой проект по модулю Потоковая обработка данных
```
Есть некий магазин, требуется в режиме реального времени, на основании скомпилированной нейросети определять тип покупателя на основании его персональных данных users_known и его корзины sales_known(метка known — известные). Те данные которые находятся в процесинге тип которых предсказали помечаются users_unknown и их корзины помечаются sales_unknown
```
Запускаю hive

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
Last login: Sat Feb  5 21:29:48 2022 from 109.252.19.10
[student898_2@bigdataanalytics-worker-3 ~]$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/hdp/3.1.4.0-315/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/hdp/3.1.4.0-315/hadoop/lib/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Connecting to jdbc:hive2://bigdataanalytics-worker-3.mcs.local:2181,bigdataanalytics-worker-2.mcs.local:2181,bigdataanalytics-worker-0.mcs.local:2181,
bigdataanalytics-worker-1.mcs.local:2181/default;password=student898_2;serviceDiscoveryMode=zooKeeper;user=student898_2;zooKeeperNamespace=hiveserver2
22/02/07 14:50:49 [main]: INFO jdbc.HiveConnection: Connected to bigdataanalytics-head-0.mcs.local:10000
Connected to: Apache Hive (version 3.1.0.3.1.4.0-315)
Driver: Hive JDBC (version 3.1.0.3.1.4.0-315)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 3.1.0.3.1.4.0-315 by Apache Hive
0: jdbc:hive2://bigdataanalytics-worker-3.mcs>
```

Подключаюсь к базе sint_sales, смотрю таблицы
use sint_sales;
show tables;

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
0: jdbc:hive2://bigdataanalytics-worker-3.mcs> use sint_sales;
INFO  : Compiling command(queryId=hive_20220207145134_69f7ef1a-b2e4-4162-b400-b14a3d41bb72): use sint_sales
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:null, properties:null)
INFO  : Completed compiling command(queryId=hive_20220207145134_69f7ef1a-b2e4-4162-b400-b14a3d41bb72); Time taken: 0.016 seconds
INFO  : Executing command(queryId=hive_20220207145134_69f7ef1a-b2e4-4162-b400-b14a3d41bb72): use sint_sales
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20220207145134_69f7ef1a-b2e4-4162-b400-b14a3d41bb72); Time taken: 0.01 seconds
INFO  : OK
No rows affected (0,128 seconds)
0: jdbc:hive2://bigdataanalytics-worker-3.mcs> show tables;
INFO  : Compiling command(queryId=hive_20220207145231_c047035a-2da5-48e4-b374-d233e431bde9): show tables
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name:tab_name, type:string, comment:from deserializer)], properties:null)
INFO  : Completed compiling command(queryId=hive_20220207145231_c047035a-2da5-48e4-b374-d233e431bde9); Time taken: 0.011 seconds
INFO  : Executing command(queryId=hive_20220207145231_c047035a-2da5-48e4-b374-d233e431bde9): show tables
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20220207145231_c047035a-2da5-48e4-b374-d233e431bde9); Time taken: 0.008 seconds
INFO  : OK
+----------------+
|    tab_name    |
+----------------+
| sales          |
| sales_known    |
| sales_unknown  |
| users          |
| users_known    |
| users_unknown  |
+----------------+
6 rows selected (0,07 seconds)
0: jdbc:hive2://bigdataanalytics-worker-3.mcs>
```

В другом окне терминала запускаю spark packages с kafka и cassandra connector
/opt/spark-2.4.8/bin/pyspark --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.5,com.datastax.spark:spark-cassandra-connector_2.11:2.4.2

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~
        confs: [default]
        0 artifacts copied, 14 already retrieved (0kB/11ms)
22/02/07 14:57:04 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/02/07 14:57:05 WARN util.Utils: Service 'SparkUI' could not bind on port 4040. Attempting port 4041.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 2.4.8
      /_/

Using Python version 2.7.5 (default, Nov 16 2020 22:23:17)
SparkSession available as 'spark'.
>>>
```

```
# Импортирую необходимые пакеты
from pyspark.ml import Pipeline, PipelineModel
from pyspark.sql import SparkSession, DataFrame
from pyspark.sql.types import StructType, StringType, IntegerType, TimestampType
from pyspark.sql import functions as F
from pyspark.ml.classification import LogisticRegression
from pyspark.ml.feature import OneHotEncoderEstimator, VectorAssembler, CountVectorizer, StringIndexer,
IndexToString

# Создаю пустой DataFrame
my_df = spark.createDataFrame( range( 1 , 200000 ), IntegerType())

# Создаю датасет
items_df = my_df.select(F.col("value").alias("order_id"), \
                        F.round( (F.rand()*49999)+1 ).alias("user_id").cast("integer"), \
                        F.round( (F.rand()*9)+1).alias("items_count").cast("integer")). \
   withColumn("price", (F.col("items_count")* F.round( (F.rand()*999)+1)).cast("integer") ). \
   withColumn("order_date", F.from_unixtime(F.unix_timestamp(F.current_date()) +
(F.lit(F.col("order_id")*10))))
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~
>>> from pyspark.ml import Pipeline, PipelineModel
>>> from pyspark.sql import SparkSession, DataFrame
>>> from pyspark.sql.types import StructType, StringType, IntegerType, TimestampType
>>> from pyspark.sql import functions as F
>>> from pyspark.ml.classification import LogisticRegression
>>> from pyspark.ml.feature import OneHotEncoderEstimator, VectorAssembler, CountVectorizer, StringIndexer, IndexToString
>>> my_df = spark.createDataFrame( range( 1 , 200000 ), IntegerType())
>>> items_df = my_df.select(F.col("value").alias("order_id"), \
...                         F.round( (F.rand()*49999)+1 ).alias("user_id").cast("integer"), \
...                         F.round( (F.rand()*9)+1).alias("items_count").cast("integer")). \
...     withColumn("price", (F.col("items_count")* F.round( (F.rand()*999)+1)).cast("integer") ). \
...     withColumn("order_date", F.from_unixtime(F.unix_timestamp(F.current_date()) + (F.lit(F.col("order_id")*10))))
>>>
```

```
# В третьем окне терминала смотрю свою директорию в hdfs и удаляю ранее созданные папки
parquet_data, my_LR_model8
hdfs dfs -ls
hdfs dfs -rm -f -r parquet_data
hdfs dfs -rm -f -r my_LR_model8
hdfs dfs -ls
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls
Found 6 items
drwx------   - student898_2 student898_2          0 2022-02-05 06:00 .Trash
drwxr-xr-x   - student898_2 student898_2          0 2022-01-30 10:56 .sparkStaging
drwxr-xr-x   - student898_2 student898_2          0 2022-01-29 21:38 for_stream
drwxr-xr-x   - student898_2 student898_2          0 2022-01-29 21:49 input_csv_for_stream
drwxr-xr-x   - student898_2 student898_2          0 2022-02-05 22:04 my_LR_model8
drwxr-xr-x   - student898_2 student898_2          0 2022-02-05 19:06 parquet_data
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -rm -f -r parquet_data
22/02/07 15:00:37 INFO fs.TrashPolicyDefault: Moved: 'hdfs://bigdataanalytics-head-0.mcs.local:8020/user/student898_2/parquet_data' to trash at: hdfs:
//bigdataanalytics-head-0.mcs.local:8020/user/student898_2/.Trash/Current/user/student898_2/parquet_data
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -rm -f -r my_LR_model8
22/02/07 15:00:58 INFO fs.TrashPolicyDefault: Moved: 'hdfs://bigdataanalytics-head-0.mcs.local:8020/user/student898_2/my_LR_model8' to trash at: hdfs:
//bigdataanalytics-head-0.mcs.local:8020/user/student898_2/.Trash/Current/user/student898_2/my_LR_model8
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls
Found 4 items
drwx------   - student898_2 student898_2          0 2022-02-07 15:00 .Trash
drwxr-xr-x   - student898_2 student898_2          0 2022-01-30 10:56 .sparkStaging
drwxr-xr-x   - student898_2 student898_2          0 2022-01-29 21:38 for_stream
drwxr-xr-x   - student898_2 student898_2          0 2022-01-29 21:49 input_csv_for_stream
[student898_2@bigdataanalytics-worker-3 ~]$
```

# Во втором окне терминала записываю и сохраняю данные в виде parquet
items_df.write.format("parquet").option("path", "parquet_data/sales").saveAsTable("sint_sales.sales", mode="overwrite")

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~
>>> items_df.write.format("parquet").option("path", "parquet_data/sales").saveAsTable("sint_sales.sales", mode="overwrite")
22/02/07 15:03:58 WARN shortcircuit.DomainSocketFactory: The short-circuit local reads feature cannot be used because libhadoop cannot be loaded.
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/readwriter.py", line 781, in saveAsTable
    self._jwrite.saveAsTable(name)
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u'org.apache.hadoop.hive.ql.metadata.HiveException: MetaException(message:java.security.AccessControlException: P
ermission denied: user=student898_2, access=WRITE, inode="/user/teacher_danilov/parquet_data/sales":teacher_danilov:teacher_danilov:drwxr-xr-x\n\tat o
rg.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.check(FSPermissionChecker.java:399)\n\tat org.apache.hadoop.hdfs.server.namenode.FSPermissio
nChecker.checkPermission(FSPermissionChecker.java:261)\n\tat org.apache.hadoop.hdfs.server.namenode.FSPermissionChecker.checkPermission(FSPermissionCh
ecker.java:193)\n\tat org.apache.hadoop.hdfs.server.namenode.FSDirectory.checkPermission(FSDirectory.java:1857)\n\tat org.apache.hadoop.hdfs.server.na
menode.FSDirectory.checkPermission(FSDirectory.java:1841)\n\tat org.apache.hadoop.hdfs.server.namenode.FSDirectory.checkPathAccess(FSDirectory.java:17
91)\n\tat org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkAccess(FSNamesystem.java:7804)\n\tat org.apache.hadoop.hdfs.server.namenode.NameNod
eRpcServer.checkAccess(NameNodeRpcServer.java:2217)\n\tat org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolServerSideTranslatorPB.checkAccess(C
lientNamenodeProtocolServerSideTranslatorPB.java:1659)\n\tat org.apache.hadoop.hdfs.protocol.proto.ClientNamenodeProtocolProtos$ClientNamenodeProtocol
$2.callBlockingMethod(ClientNamenodeProtocolProtos.java)\n\tat org.apache.hadoop.ipc.ProtobufRpcEngine$Server$ProtoBufRpcInvoker.call(ProtobufRpcEngin
e.java:524)\n\tat org.apache.hadoop.ipc.RPC$Server.call(RPC.java:1025)\n\tat org.apache.hadoop.ipc.Server$RpcCall.run(Server.java:876)\n\tat org.apach
e.hadoop.ipc.Server$RpcCall.run(Server.java:822)\n\tat java.security.AccessController.doPrivileged(Native Method)\n\tat javax.security.auth.Subject.do
As(Subject.java:422)\n\tat org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1730)\n\tat org.apache.hadoop.ipc.Server$Han
dler.run(Server.java:2682)\n);'
>>>
```

# Смотрю что записалось
items_df.show()

```
>>> items_df.show()
22/02/07 15:05:40 WARN scheduler.TaskSetManager: Stage 0 contains a task of very large size (412 KB). The maximum recommended task size is 100 KB.
+--------+-------+-----------+-----+-------------------+
|order_id|user_id|items_count|price|         order_date|
+--------+-------+-----------+-----+-------------------+
|       1|   4681|         10| 4760|2022-02-07 00:00:10|
|       2|   8176|          4| 2424|2022-02-07 00:00:20|
|       3|  41998|          5| 1860|2022-02-07 00:00:30|
|       4|   5838|          9|  621|2022-02-07 00:00:40|
|       5|  42264|          5| 2105|2022-02-07 00:00:50|
|       6|  35668|          9| 4365|2022-02-07 00:01:00|
|       7|  25763|          9| 1071|2022-02-07 00:01:10|
|       8|  27944|          5| 3955|2022-02-07 00:01:20|
|       9|  37806|          1|  429|2022-02-07 00:01:30|
|      10|  25500|          8| 6864|2022-02-07 00:01:40|
|      11|   5005|          8| 3800|2022-02-07 00:01:50|
|      12|  42143|          5| 4285|2022-02-07 00:02:00|
|      13|  33600|          2| 1762|2022-02-07 00:02:10|
|      14|   2437|          6| 5526|2022-02-07 00:02:20|
|      15|  17743|          6| 5976|2022-02-07 00:02:30|
|      16|  41504|          3|  903|2022-02-07 00:02:40|
|      17|  34912|          8|  712|2022-02-07 00:02:50|
|      18|  25558|          9| 4023|2022-02-07 00:03:00|
|      19|   9037|          7| 6475|2022-02-07 00:03:10|
|      20|  26382|          1|   82|2022-02-07 00:03:20|
+--------+-------+-----------+-----+-------------------+
only showing top 20 rows

>>>
```

В третьем окне терминала проверяю и создаю директорию parquet_data
hdfs dfs -ls parquet_data
hdfs dfs -mkdir parquet_data
hdfs dfs -ls

```
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls parquet_data
ls: `parquet_data': No such file or directory
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -mkdir parquet_data
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls
Found 5 items
drwx------   - student898_2 student898_2          0 2022-02-07 15:00 .Trash
drwxr-xr-x   - student898_2 student898_2          0 2022-01-30 10:56 .sparkStaging
drwxr-xr-x   - student898_2 student898_2          0 2022-01-29 21:38 for_stream
drwxr-xr-x   - student898_2 student898_2          0 2022-01-29 21:49 input_csv_for_stream
drwxr-xr-x   - student898_2 student898_2          0 2022-02-07 15:10 parquet_data
[student898_2@bigdataanalytics-worker-3 ~]$
```

В первом окне терминала делаю выборку 1 строки из sales
select * from sales limit 1;

```
0: jdbc:hive2://bigdataanalytics-worker-3.mcs> select * from sales limit 1;
INFO  : Compiling command(queryId=hive_20220207151321_3ddda11e-f540-468a-b168-9070b17ae1ea): select * from sales limit 1
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name:sales.order_id, type:int, comment:null), FieldSchema(name:sales.user_id, type:int
, comment:null), FieldSchema(name:sales.items_count, type:int, comment:null), FieldSchema(name:sales.price, type:int, comment:null), FieldSchema(name:
sales.order_date, type:string, comment:null)], properties:null)
INFO  : Completed compiling command(queryId=hive_20220207151321_3ddda11e-f540-468a-b168-9070b17ae1ea); Time taken: 0.12 seconds
INFO  : Executing command(queryId=hive_20220207151321_3ddda11e-f540-468a-b168-9070b17ae1ea): select * from sales limit 1
INFO  : Completed executing command(queryId=hive_20220207151321_3ddda11e-f540-468a-b168-9070b17ae1ea); Time taken: 0.001 seconds
INFO  : OK
+-----------------+----------------+--------------------+--------------+----------------------+
| sales.order_id  | sales.user_id  | sales.items_count  | sales.price  |   sales.order_date   |
+-----------------+----------------+--------------------+--------------+----------------------+
| 1               | 48936          | 4                  | 260          | 2022-02-03 00:00:10  |
+-----------------+----------------+--------------------+--------------+----------------------+
1 row selected (0,329 seconds)
0: jdbc:hive2://bigdataanalytics-worker-3.mcs>
```

# Во втором окне терминала создаю items_df sint_sales
items_df=spark.table("sint_sales.sales")
Создаю таблицу users
spark.sql("""create table sint_sales.users
(user_id int,
gender string,
age string,
segment string)
stored as parquet location 'parquet_data/users'""")

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
>>> items_df=spark.table("sint_sales.sales")
>>> spark.sql("""create table sint_sales.users
...     (user_id int,
...     gender string,
...     age string,
...     segment string)
...     stored as parquet location 'parquet_data/users'""")
Traceback (most recent call last):
  File "<stdin>", line 6, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/session.py", line 767, in sql
    return DataFrame(self._jsparkSession.sql(sqlQuery), self._wrapped)
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 71, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u"Table or view 'users' already exists in database 'sint_sales';"
>>>
```

В третьем окне терминала смотрю директорию parquet_data
hdfs dfs -ls parquet_data

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls parquet_data
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls parquet_data/
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls parquet_data
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls parquet_data
[student898_2@bigdataanalytics-worker-3 ~]$
```

В первом окне терминала смотрю таблицы
show tables;

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
0: jdbc:hive2://bigdataanalytics-worker-3.mcs> show tables;
INFO  : Compiling command(queryId=hive_20220207152646_90b4f585-d274-4b74-b9ab-fc394c270057): show tables
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name:tab_name, type:string, comment:from deserializer)], properties:null)
INFO  : Completed compiling command(queryId=hive_20220207152646_90b4f585-d274-4b74-b9ab-fc394c270057); Time taken: 0.015 seconds
INFO  : Executing command(queryId=hive_20220207152646_90b4f585-d274-4b74-b9ab-fc394c270057): show tables
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20220207152646_90b4f585-d274-4b74-b9ab-fc394c270057); Time taken: 0.01 seconds
INFO  : OK
+----------------+
|    tab_name    |
+----------------+
| sales          |
| sales_known    |
| sales_unknown  |
| users          |
| users_known    |
| users_unknown  |
+----------------+
6 rows selected (0,047 seconds)
0: jdbc:hive2://bigdataanalytics-worker-3.mcs>
```

Во втором окне терминала провожу нормализация по полу, группирую и поместил это все в таблицу users

```
spark.sql("""insert into sint_sales.users
        select user_id, case when pmod( user_id, 2 )=0 then 'M' else 'F' end,
        case when pmod(user_id, 3 )=0 then 'young' when pmod(user_id, 3 )=1 then 'midage' else 'old' end ,
        case when s>23 then 'happy' when s>15 then 'neutral' else 'shy' end
        from (
        select sum(items_count) s, user_id from sint_sales.sales group by user_id ) t""")
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
        at org.apache.hadoop.ipc.Client.call(Client.java:1475)
        at org.apache.hadoop.ipc.Client.call(Client.java:1412)
        at org.apache.hadoop.ipc.ProtobufRpcEngine$Invoker.invoke(ProtobufRpcEngine.java:229)
        at com.sun.proxy.$Proxy27.mkdirs(Unknown Source)
        at org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.mkdirs(ClientNamenodeProtocolTranslatorPB.java:558)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at org.apache.hadoop.io.retry.RetryInvocationHandler.invokeMethod(RetryInvocationHandler.java:191)
        at org.apache.hadoop.io.retry.RetryInvocationHandler.invoke(RetryInvocationHandler.java:102)
        at com.sun.proxy.$Proxy28.mkdirs(Unknown Source)
        at org.apache.hadoop.hdfs.DFSClient.primitiveMkdir(DFSClient.java:3000)
        ... 35 more

>>>
```

В первом окне терминала делаю выборку одной записи из таблицы users

```
select * from users limit 1;
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
0: jdbc:hive2://bigdataanalytics-worker-3.mcs> select * from users limit 1;
INFO  : Compiling command(queryId=hive_20220207153021_934cbe58-a753-498e-b11a-4d9586323130): select * from users limit 1
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name:users.user_id, type:int, comment:null), FieldSchema(name:users.gender, type:strin
g, comment:null), FieldSchema(name:users.age, type:string, comment:null), FieldSchema(name:users.segment, type:string, comment:null)], properties:null
)
INFO  : Completed compiling command(queryId=hive_20220207153021_934cbe58-a753-498e-b11a-4d9586323130); Time taken: 0.118 seconds
INFO  : Executing command(queryId=hive_20220207153021_934cbe58-a753-498e-b11a-4d9586323130): select * from users limit 1
INFO  : Completed executing command(queryId=hive_20220207153021_934cbe58-a753-498e-b11a-4d9586323130); Time taken: 0.001 seconds
INFO  : OK
+----------------+---------------+------------+----------------+
| users.user_id  | users.gender  | users.age  | users.segment  |
+----------------+---------------+------------+----------------+
| 28577          | F             | old        | neutral        |
+----------------+---------------+------------+----------------+
1 row selected (0,196 seconds)
0: jdbc:hive2://bigdataanalytics-worker-3.mcs>
```

Во втором окне терминала делаю выборку и создаю таблицы для известных/неизвестных пользователей и покупок

```
spark.sql("""create table sint_sales.users_known stored as parquet location 'parquet_data/users_known' as
        select * from sint_sales.users where user_id < 30000
        """)
spark.sql("""create table sint_sales.users_unknown stored as parquet location 'parquet_data/users_unknown'
as
        select user_id, gender, age from sint_sales.users where user_id >= 30000
        """)
spark.sql("""create table sint_sales.sales_known stored as parquet location 'parquet_data/sales_known' as
        select * from sint_sales.sales where user_id < 30000
        """)
spark.sql("""create table sint_sales.sales_unknown stored as parquet location 'parquet_data/sales_unknown'
as
        select * from sint_sales.sales where user_id >= 30000
        """)
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~
>>> spark.sql("""create table sint_sales.users_known stored as parquet location 'parquet_data/users_known' as
...     select * from sint_sales.users where user_id < 30000
...     """)
Traceback (most recent call last):
  File "<stdin>", line 3, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/session.py", line 767, in sql
    return DataFrame(self._jsparkSession.sql(sqlQuery), self._wrapped)
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u'`sint_sales`.`users_known` already exists.;'
>>> spark.sql("""create table sint_sales.users_unknown stored as parquet location 'parquet_data/users_unknown' as
...     select user_id, gender, age from sint_sales.users where user_id >= 30000
...     """)
Traceback (most recent call last):
  File "<stdin>", line 3, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/session.py", line 767, in sql
    return DataFrame(self._jsparkSession.sql(sqlQuery), self._wrapped)
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u'`sint_sales`.`users_unknown` already exists.;'
>>> spark.sql("""create table sint_sales.sales_known stored as parquet location 'parquet_data/sales_known' as
...     select * from sint_sales.sales where user_id < 30000
...     """)
Traceback (most recent call last):
  File "<stdin>", line 3, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/session.py", line 767, in sql
    return DataFrame(self._jsparkSession.sql(sqlQuery), self._wrapped)
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u'`sint_sales`.`sales_known` already exists.;'
>>> spark.sql("""create table sint_sales.sales_unknown stored as parquet location 'parquet_data/sales_unknown' as
...     select * from sint_sales.sales where user_id >= 30000
...     """)
Traceback (most recent call last):
  File "<stdin>", line 3, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/session.py", line 767, in sql
    return DataFrame(self._jsparkSession.sql(sqlQuery), self._wrapped)
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u'`sint_sales`.`sales_unknown` already exists.;'
>>>
```

В первом окне терминала смотрю таблицы
show tables;

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20220207153809_50cece7f-4454-4f4d-9fe1-de9f37a18f04); Time taken: 0.008 seconds
INFO  : OK
+----------------+
|    tab_name    |
+----------------+
| sales          |
| sales_known    |
| sales_unknown  |
| users          |
| users_known    |
| users_unknown  |
+----------------+
6 rows selected (0,053 seconds)
0: jdbc:hive2://bigdataanalytics-worker-3.mcs>
```

Делаю выборку по 1 записи из таблиц ales_known и sales_unknown
select * from sales_known limit 1;
select * from sales_unknown limit 1;

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~

0: jdbc:hive2://bigdataanalytics-worker-3.mcs> select * from sales_known limit 1;
INFO  : Compiling command(queryId=hive_20220207153906_d377285c-1c16-44f1-8829-098b66ccf6f5): select * from sales_known limit 1
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name:sales_known.order_id, type:int, comment:null), FieldSchema(name:sales_known.user_
id, type:int, comment:null), FieldSchema(name:sales_known.items_count, type:int, comment:null), FieldSchema(name:sales_known.price, type:int, comment:
null), FieldSchema(name:sales_known.order_date, type:string, comment:null)], properties:null)
INFO  : Completed compiling command(queryId=hive_20220207153906_d377285c-1c16-44f1-8829-098b66ccf6f5); Time taken: 0.113 seconds
INFO  : Executing command(queryId=hive_20220207153906_d377285c-1c16-44f1-8829-098b66ccf6f5): select * from sales_known limit 1
INFO  : Completed executing command(queryId=hive_20220207153906_d377285c-1c16-44f1-8829-098b66ccf6f5); Time taken: 0.0 seconds
INFO  : OK
+-----------------------+---------------------+-------------------------+-------------------+-----------------------+
| sales_known.order_id  | sales_known.user_id | sales_known.items_count | sales_known.price | sales_known.order_date |
+-----------------------+---------------------+-------------------------+-------------------+-----------------------+
| 2                     | 19669               | 9                       | 8739              | 2022-02-03 00:00:20   |
+-----------------------+---------------------+-------------------------+-------------------+-----------------------+
1 row selected (0,17 seconds)
0: jdbc:hive2://bigdataanalytics-worker-3.mcs> select * from sales_unknown limit 1;
INFO  : Compiling command(queryId=hive_20220207153954_5bffc4f4-9785-4712-a260-d83f4d179a07): select * from sales_unknown limit 1
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name:sales_unknown.order_id, type:int, comment:null), FieldSchema(name:sales_unknown.u
ser_id, type:int, comment:null), FieldSchema(name:sales_unknown.items_count, type:int, comment:null), FieldSchema(name:sales_unknown.price, type:int,
comment:null), FieldSchema(name:sales_unknown.order_date, type:string, comment:null)], properties:null)
INFO  : Completed compiling command(queryId=hive_20220207153954_5bffc4f4-9785-4712-a260-d83f4d179a07); Time taken: 0.104 seconds
INFO  : Executing command(queryId=hive_20220207153954_5bffc4f4-9785-4712-a260-d83f4d179a07): select * from sales_unknown limit 1
INFO  : Completed executing command(queryId=hive_20220207153954_5bffc4f4-9785-4712-a260-d83f4d179a07); Time taken: 0.0 seconds
INFO  : OK
+-------------------------+-----------------------+---------------------------+---------------------+--------------------------+
| sales_unknown.order_id  | sales_unknown.user_id | sales_unknown.items_count | sales_unknown.price | sales_unknown.order_date |
+-------------------------+-----------------------+---------------------------+---------------------+--------------------------+
| 1                       | 48936                 | 4                         | 260                 | 2022-02-03 00:00:10      |
+-------------------------+-----------------------+---------------------------+---------------------+--------------------------+
1 row selected (0,194 seconds)
0: jdbc:hive2://bigdataanalytics-worker-3.mcs>
```

```
# Во втором окне терминала беру объект sales загружаю в df
items_df=spark.table("sint_sales.sales")

# Делаю  join всех покупок и покупателей кое-то кол-во статистических данных, по ключу соединил,
сгруппировал и агрегировал, заполнил  gender,  age,  segment
df = spark.sql("""
        select count(*) as c, sum(items_count) as s1, max(items_count) as ma1, min(items_count) as mi1,
        sum(price) as s2, max(price) as ma2, min(price) as mi2 ,u.gender, u.age, u.user_id, u.segment
        from sint_sales.sales_known s join sint_sales.users_known u
        where s.user_id = u.user_id
        group by u.user_id, u.gender, u.age, u.segment""")
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~

>>> items_df=spark.table("sint_sales.sales")
>>> df = spark.sql("""
...     select count(*) as c, sum(items_count) as s1, max(items_count) as ma1, min(items_count) as mi1,
...     sum(price) as s2, max(price) as ma2, min(price) as mi2 ,u.gender, u.age, u.user_id, u.segment
...     from sint_sales.sales_known s join sint_sales.users_known u
...     where s.user_id = u.user_id
...     group by u.user_id, u.gender, u.age, u.segment""")
>>>
```

```
# Делаю просмотр sales_unknown
sales_unknown = spark.table("sint_sales.sales_unknown")
sales_unknown.show()
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
>>> sales_unknown = spark.table("sint_sales.sales_unknown")
>>> sales_unknown.show()
+--------+-------+-----------+-----+-------------------+
|order_id|user_id|items_count|price|         order_date|
+--------+-------+-----------+-----+-------------------+
|       1|  48936|          4|  260|2022-02-03 00:00:10|
|       4|  40342|          9| 3483|2022-02-03 00:00:40|
|       5|  31735|          6| 4920|2022-02-03 00:00:50|
|      12|  38683|         10| 7580|2022-02-03 00:02:00|
|      15|  42648|          2|  494|2022-02-03 00:02:30|
|      17|  40660|          7| 4025|2022-02-03 00:02:50|
|      23|  33895|          5| 1080|2022-02-03 00:03:50|
|      30|  42285|          6|  288|2022-02-03 00:05:00|
|      35|  32977|          6| 1554|2022-02-03 00:05:50|
|      39|  48692|          6|   36|2022-02-03 00:06:30|
|      40|  47728|          5| 2530|2022-02-03 00:06:40|
|      42|  46755|          2|  818|2022-02-03 00:07:00|
|      44|  33632|          2| 1596|2022-02-03 00:07:20|
|      46|  40369|          2|  198|2022-02-03 00:07:40|
|      50|  36538|          7| 4067|2022-02-03 00:08:20|
|      54|  35436|         10| 4820|2022-02-03 00:09:00|
|      58|  41768|          6|  204|2022-02-03 00:09:40|
|      59|  49023|          7| 1526|2022-02-03 00:09:50|
|      61|  38910|          5| 4665|2022-02-03 00:10:10|
|      63|  31783|          5| 1620|2022-02-03 00:10:30|
+--------+-------+-----------+-----+-------------------+
only showing top 20 rows

>>>
```

# Делаю чтение в parquet sint_sales и parquet_data в бд sales_unknown
sales_unknown = spark.read.parquet("/apps/spark/warehouse/sint_sales.db/sales_unknown")
sales_unknown = spark.read.parquet("parquet_data/sales_unknown")

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
>>> sales_unknown = spark.read.parquet("/apps/spark/warehouse/sint_sales.db/sales_unknown")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/readwriter.py", line 316, in parquet
    return self._df(self._jreader.parquet(_to_seq(self._spark._sc, paths)))
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u'Path does not exist: hdfs://bigdataanalytics-head-0.mcs.local:8020/apps/spark/warehouse/sint_sales.db/sales_unk
nown;'
>>> sales_unknown = spark.read.parquet("parquet_data/sales_unknown")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/readwriter.py", line 316, in parquet
    return self._df(self._jreader.parquet(_to_seq(self._spark._sc, paths)))
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u'Path does not exist: hdfs://bigdataanalytics-head-0.mcs.local:8020/user/student898_2/parquet_data/sales_unknown
;'
>>>
```

# Смотрю записи в sales_unknown и запускаю kafka_brokers
sales_unknown.show()
kafka_brokers = 'bigdataanalytics-worker-3:6667'

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
>>> sales_unknown.show()
+--------+-------+-----------+-----+-------------------+
|order_id|user_id|items_count|price|         order_date|
+--------+-------+-----------+-----+-------------------+
|       1|  48936|          4|  260|2022-02-03 00:00:10|
|       4|  40342|          9| 3483|2022-02-03 00:00:40|
|       5|  31735|          6| 4920|2022-02-03 00:00:50|
|      12|  38683|         10| 7580|2022-02-03 00:02:00|
|      15|  42648|          2|  494|2022-02-03 00:02:30|
|      17|  40660|          7| 4025|2022-02-03 00:02:50|
|      23|  33895|          5| 1080|2022-02-03 00:03:50|
|      30|  42285|          6|  288|2022-02-03 00:05:00|
|      35|  32977|          6| 1554|2022-02-03 00:05:50|
|      39|  48692|          6|   36|2022-02-03 00:06:30|
|      40|  47728|          5| 2530|2022-02-03 00:06:40|
|      42|  46755|          2|  818|2022-02-03 00:07:00|
|      44|  33632|          2| 1596|2022-02-03 00:07:20|
|      46|  40369|          2|  198|2022-02-03 00:07:40|
|      50|  36538|          7| 4067|2022-02-03 00:08:20|
|      54|  35436|         10| 4820|2022-02-03 00:09:00|
|      58|  41768|          6|  204|2022-02-03 00:09:40|
|      59|  49023|          7| 1526|2022-02-03 00:09:50|
|      61|  38910|          5| 4665|2022-02-03 00:10:10|
|      63|  31783|          5| 1620|2022-02-03 00:10:30|
+--------+-------+-----------+-----+-------------------+
only showing top 20 rows

>>> kafka_brokers = 'bigdataanalytics-worker-3:6667'
>>>
```

В третьем окне терминала смотрю лист с topics
/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --zookeeper bigdataanalytics-worker-3:2181 —list

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
sales_unknown
shadrin_data
shadrin_iris
shadrin_iris_559
shadrin_iris_pikalev
st_covid_desc
st_covid_variants
st_lesson5
student559-12-ks
student559_12
student559_17
student559_17_iris
student559_17_sink
student559_8_bank
student559_8_lesson2
student559_8_lesson3
student559_8_lesson4_iris
student559_8_lesson5
test-lesson2
test_iris_sink
test_lesson2
test_lesson2_1
test_lesson_2_sapr
test_lesson_3_sapr
tolstykov_les3
tolstykov_les4
tolstykov_les4_sink
tolstykov_les5
tolstykov_les8
us_navy_20
[student898_2@bigdataanalytics-worker-3 ~]$
```

Удалю старые topics и создаю новый sales_unknown
/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les8 --zookeeper bigdataanalytics-worker-3:2181
/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les5 --zookeeper bigdataanalytics-worker-3:2181

/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les4 --zookeeper bigdataanalytics-worker-3:2181
/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les3 --zookeeper bigdataanalytics-worker-3:2181
/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les4_sink --zookeeper bigdataanalytics-worker-3:2181
/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic sales_unknown --zookeeper bigdataanalytics-worker-3:2181
/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --create --topic sales_unknown --zookeeper bigdataanalytics-worker-3:2181 --partitions 1 --replication-factor 1

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les8 --zookeeper bigdataanaly
tics-worker-3:2181
Topic tolstykov_les8 is marked for deletion.
Note: This will have no impact if delete.topic.enable is not set to true.
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les5 --zookeeper bigdataanaly
tics-worker-3:2181
Topic tolstykov_les5 is marked for deletion.
Note: This will have no impact if delete.topic.enable is not set to true.
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les4 --zookeeper bigdataanaly
tics-worker-3:2181
Topic tolstykov_les4 is marked for deletion.
Note: This will have no impact if delete.topic.enable is not set to true.
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les3 --zookeeper bigdataanaly
tics-worker-3:2181
Topic tolstykov_les3 is marked for deletion.
Note: This will have no impact if delete.topic.enable is not set to true.
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic tolstykov_les4_sink --zookeeper bigdata
analytics-worker-3:2181
Topic tolstykov_les4_sink is marked for deletion.
Note: This will have no impact if delete.topic.enable is not set to true.
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-topics.sh --delete --topic sales_unknown --zookeeper bigdataanalyt
ics-worker-3:2181
Topic sales_unknown is marked for deletion.
Note: This will have no impact if delete.topic.enable is not set to true.
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-topics.sh --create --topic sales_unknown --zookeeper bigdataanalyt
ics-worker-3:2181 --partitions 1 --replication-factor 1
WARNING: Due to limitations in metric names, topics with a period ('.') or underscore ('_') could collide. To avoid issues it is best to use either, b
ut not both.
Created topic "sales_unknown".
[student898_2@bigdataanalytics-worker-3 ~]$
```

Во втором окне терминала запускаю запись bootstrap.servers

```
sales_unknown.selectExpr("cast (null as string) as key", "cast (to_json(struct(*)) as string) as value"). \
    write.format("kafka"). \
    option("kafka.bootstrap.servers", kafka_brokers). \
    option("topic", "sales_unknown"). \
    save()
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~ | student898_2@bigdataanalytics-worker-3:~

```
>>> sales_unknown.selectExpr("cast (null as string) as key", "cast (to_json(struct(*)) as string) as value"). \
...     write.format("kafka"). \
...     option("kafka.bootstrap.servers", kafka_brokers). \
...     option("topic", "sales_unknown"). \
...     save()
>>>
```

Запускаю запись в cassandra

```
users_unknown = spark.table("sint_sales.users_unknown")
users_unknown.write \
    .format("org.apache.spark.sql.cassandra") \
    .options(table="users_unknown", keyspace="keyspace1") \
    .mode("append")\
    .save()
```

```
>>> users_unknown = spark.table("sint_sales.users_unknown")
>>> users_unknown.write \
...     .format("org.apache.spark.sql.cassandra") \
...     .options(table="users_unknown", keyspace="keyspace1") \
...     .mode("append")\
...     .save()
>>>
```

В третьем окне терминала запускаю cassandra и смотрю базы
cqlsh
select keyspace_name, table_name from system_schema.tables where keyspace_name = 'keyspace1';

```
[student898_2@bigdataanalytics-worker-3 ~]$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.0.0 | Cassandra 4.0.1 | CQL spec 3.4.5 | Native protocol v5]
Use HELP for help.
cqlsh> select keyspace_name, table_name from system_schema.tables where keyspace_name = 'keyspace1';

 keyspace_name | table_name
---------------+---------------
     keyspace1 |    prog_langs
     keyspace1 | users_unknown

(2 rows)
cqlsh>
```

Смотрю базы и таблицы
select keyspace_name, table_name from system_schema.tables;

```
               system |          transferred_ranges_v2
               system |        view_builds_in_progress
       student559_12_2 |                  universities
         student559_12 |                       example
         keyspace_test |                 shadrin_animals
         system_traces |                        events
         system_traces |                      sessions
            lesson7_20 |                  aircraft_copy
            lesson7_20 |                  aircraft_rows
            lesson7_20 |                       animals

(71 rows)
cqlsh>
```

Переключаюсь на keyspace1 и создаю таблицу users_unknown
use keyspace1;
CREATE TABLE users_unknown(user_id int, gender text, age text, c int, s1 int, ma1 int, mi1 int, s2 int, ma2 int, mi2 int, segment text, primary key (user_id));

```
cqlsh> use keyspace1;
cqlsh:keyspace1> CREATE TABLE users_unknown( user_id int, gender text, age text, c int, s1 int,  ma1 int, mi1 int, s2 int, ma2 int, mi2 int, segment t
ext, primary key (user_id));
AlreadyExists: Table 'keyspace1.users_unknown' already exists
cqlsh:keyspace1>
```

Делаю выборку из users_unknown
select * from users_unknown;

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~   student898_2@bigdataanalytics-worker-3:~

cqlsh:keyspace1> select * from users_unknown;

 user_id | age    | c    | gender | ma1  | ma2  | mi1  | mi2  | s1   | s2   | segment
---------+--------+------+--------+------+------+------+------+------+------+---------
   35262 |  young | null |      M | null | null | null | null | null | null |    null
   39433 | midage | null |      F | null | null | null | null | null | null |    null
   37032 |  young | null |      M | null | null | null | null | null | null |    null
   48451 | midage | null |      F | null | null | null | null | null | null |    null
   40239 |  young | null |      F | null | null | null | null | null | null |    null
   47076 |  young | null |      M | null | null | null | null | null | null |    null
   41114 |    old | null |      M | null | null | null | null | null | null |    null
```

Во втором окне терминала запускаю чтения parquet с созданием временных представлений известных покупок и известных пользователей

spark.read.parquet("parquet_data/sales_known").createOrReplaceTempView("sales_known")
spark.read.parquet("parquet_data/users_known").createOrReplaceTempView("users_known")

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-wor...   student898_2@bigdataanalytics-wor...   student898_2@bigdataanalytics-wor...   student898_2@bigdataanalytics-wor...

>>> spark.read.parquet("parquet_data/sales_known").createOrReplaceTempView("sales_known")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/readwriter.py", line 316, in parquet
    return self._df(self._jreader.parquet(_to_seq(self._spark._sc, paths)))
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u'Path does not exist: hdfs://bigdataanalytics-head-0.mcs.local:8020/user/student898_2/parquet_data/sales_known;'
>>> spark.read.parquet("parquet_data/users_known").createOrReplaceTempView("users_known")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/opt/spark-2.4.8/python/pyspark/sql/readwriter.py", line 316, in parquet
    return self._df(self._jreader.parquet(_to_seq(self._spark._sc, paths)))
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u'Path does not exist: hdfs://bigdataanalytics-head-0.mcs.local:8020/user/student898_2/parquet_data/users_known;'
>>>
```

Выборка sint_sales по 10 строк из sales_known и users_known

spark.sql("select * from sint_sales.sales_known").show(10, False)
spark.sql("select * from sint_sales.users_known").show(10, False)

```
>>> spark.sql("select * from sint_sales.sales_known").show(10, False)
+--------+-------+-----------+-----+-------------------+
|order_id|user_id|items_count|price|order_date         |
+--------+-------+-----------+-----+-------------------+
|100353  |233    |5          |1860 |2022-02-14 14:45:30|
|100355  |29361  |2          |1372 |2022-02-14 14:45:50|
|100357  |26046  |3          |1380 |2022-02-14 14:46:10|
|100358  |19783  |4          |2284 |2022-02-14 14:46:20|
|100359  |25546  |5          |590  |2022-02-14 14:46:30|
|100360  |25016  |2          |504  |2022-02-14 14:46:40|
|100361  |5539   |4          |316  |2022-02-14 14:46:50|
|100363  |11636  |6          |4890 |2022-02-14 14:47:10|
|100366  |26542  |2          |1920 |2022-02-14 14:47:40|
|100367  |14151  |10         |440  |2022-02-14 14:47:50|
+--------+-------+-----------+-----+-------------------+
only showing top 10 rows

>>> spark.sql("select * from sint_sales.users_known").show(10, False)
+-------+------+------+-------+
|user_id|gender|age   |segment|
+-------+------+------+-------+
|26893  |F     |midage|shy    |
|18334  |M     |midage|neutral|
|2655   |F     |young |shy    |
|22443  |F     |young |neutral|
|20195  |F     |old   |neutral|
|21298  |M     |midage|happy  |
|27228  |M     |young |shy    |
|883    |F     |midage|happy  |
|23362  |M     |midage|neutral|
|6154   |M     |midage|neutral|
+-------+------+------+-------+
only showing top 10 rows

>>>
```

Считаю сегмент зависимым от количества покупок клиента, суммы всех купленых товаров клиента, максимального числа купленных товаров клиента, минимального числа купленных товаров клиента, суммы потраченных рублей клиента, максимально потраченных рублей клиента, минимально потраченных рублей клиента

```
users_known = spark.sql("""
    select count(*) as c, sum(items_count) as s1, max(items_count) as ma1, min(items_count) as mi1,
    sum(price) as s2, max(price) as ma2, min(price) as mi2 ,u.gender, u.age, u.user_id, u.segment
    from sint_sales.sales_known s join sint_sales.users_known u
    where s.user_id = u.user_id
    group by u.user_id, u.gender, u.age, u.segment""")
```

```
>>> users_known = spark.sql("""
...     select count(*) as c, sum(items_count) as s1, max(items_count) as ma1, min(items_count) as mi1,
...     sum(price) as s2, max(price) as ma2, min(price) as mi2 ,u.gender, u.age, u.user_id, u.segment
...     from sint_sales.sales_known s join sint_sales.users_known u
...     where s.user_id = u.user_id
...     group by u.user_id, u.gender, u.age, u.segment""")
>>>
```

Подготовка модели машиннго обучения.

```
df = users_known
df.show()
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor...

```
>>> df = users_known
>>> df.show()
+---+---+---+---+-----+----+----+------+------+-------+-------+
|  c| s1|ma1|mi1|   s2| ma2| mi2|gender|   age|user_id|segment|
+---+---+---+---+-----+----+----+------+------+-------+-------+
|  5| 34| 10|  2|16304|7060| 676|     M|midage|    148|  happy|
|  1|  7|  7|  7| 4935|4935|4935|     F|midage|    463|    shy|
|  5| 31| 10|  1|23446|8560| 918|     F| young|    471|  happy|
|  4| 15|  7|  2| 6461|2842| 171|     M|midage|    496|    shy|
|  2| 11|  6|  5| 8019|5664|2355|     F|   old|    833|    shy|
|  3| 11|  7|  1| 4792|2898| 694|     M|   old|   1088|    shy|
|  9| 59| 10|  3|27227|7083|1197|     M|   old|   1238|  happy|
|  5| 35|  9|  4|18199|5154|2320|     M|midage|   1342|  happy|
|  5| 28|  8|  3|13614|5874| 408|     M|   old|   1580|  happy|
|  2|  9|  5|  4| 2238|1450| 788|     F|midage|   1591|    shy|
|  4| 17|  8|  2| 8648|5368| 116|     F|midage|   1645|neutral|
|  5| 41| 10|  6|27161|8442| 420|     F|   old|   1829|  happy|
|  2| 12| 10|  2| 5220|3920|1300|     F| young|   1959|    shy|
|  3| 18|  9|  2| 9009|7524| 329|     M|midage|   2122|neutral|
|  4| 26|  9|  3| 7269|5184| 264|     M|   old|   2366|  happy|
|  3| 20|  9|  5|15316|6165|4446|     F|midage|   2659|neutral|
|  6| 21|  7|  1| 5689|1323| 204|     M|midage|   2866|neutral|
|  1|  9|  9|  9| 4131|4131|4131|     F|midage|   3175|    shy|
|  8| 43|  9|  2|14454|4585|  30|     F|   old|   3749|  happy|
|  2|  9|  7|  2| 4930|3080|1850|     M|   old|   3794|    shy|
+---+---+---+---+-----+----+----+------+------+-------+-------+
only showing top 20 rows

>>>
```

```
# Беру колонки которые будет обрабатывать  stringIndexer, потом  encoder
categoricalColumns = ['gender', 'age']
stages = []
for categoricalCol in categoricalColumns:
        stringIndexer = StringIndexer(inputCol = categoricalCol, outputCol = categoricalCol + 'Index')
        encoder = OneHotEncoderEstimator(inputCols=[stringIndexer.getOutputCol()],
outputCols=[categoricalCol + "classVec"])
        stages += [stringIndexer, encoder]
stages
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor...

```
>>> categoricalColumns = ['gender', 'age']
>>> stages = []
>>> for categoricalCol in categoricalColumns:
...     stringIndexer = StringIndexer(inputCol = categoricalCol, outputCol = categoricalCol + 'Index')
...     encoder = OneHotEncoderEstimator(inputCols=[stringIndexer.getOutputCol()], outputCols=[categoricalCol + "classVec"])
...     stages += [stringIndexer, encoder]
...
>>> stages
[StringIndexer_f311599de7bc, OneHotEncoderEstimator_ce0dbf596089, StringIndexer_c4cc4b820dae, OneHotEncoderEstimator_5d16b7b63811]
>>>
```

```
# Определяю label, является  segment, т. е. та колонка которую буду предсказывать
label_stringIdx = StringIndexer(inputCol = 'segment', outputCol = 'label')
stages += [label_stringIdx]

# Колонки с численными значениями кол-во записей, покупок, общее, макс, мин, общий счет, макс
чек, мин чек, преобразование в вектор
numericCols = ['c' ,'s1', 'ma1', 'mi1','s2', 'ma2', 'mi2']
assemblerInputs = [c + "classVec" for c in categoricalColumns] + numericCols
assembler = VectorAssembler(inputCols=assemblerInputs, outputCol="features")
stages += [assembler]

# Добавляю LR в поле stages, переименование колонки label в  prediction, то что предсказываю
lr = LogisticRegression(featuresCol = 'features', labelCol = 'label', maxIter=10)
```

```
stages += [lr]
label_stringIdx_fit = label_stringIdx.fit(df)
indexToStringEstimator =
IndexToString().setInputCol("prediction").setOutputCol("category").setLabels(  label_stringIdx_fit.labels)
stages +=[indexToStringEstimator]

# Собираю  Pipeline, соединение, запись, сохранение
pipeline = Pipeline().setStages(stages)
pipelineModel = pipeline.fit(df)

# Сохраняю модель на HDFS
pipelineModel.write().overwrite().save("my_LR_model8")
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

```
>>> label_stringIdx = StringIndexer(inputCol = 'segment', outputCol = 'label')
>>> stages += [label_stringIdx]
>>> numericCols = ['c' ,'s1', 'ma1', 'mi1','s2', 'ma2', 'mi2']
>>> assemblerInputs = [c + "classVec" for c in categoricalColumns] + numericCols
>>> assembler = VectorAssembler(inputCols=assemblerInputs, outputCol="features")
>>> stages += [assembler]
>>> lr = LogisticRegression(featuresCol = 'features', labelCol = 'label', maxIter=10)
>>> stages += [lr]
>>> label_stringIdx_fit = label_stringIdx.fit(df)
>>> indexToStringEstimator = IndexToString().setInputCol("prediction").setOutputCol("category").setLabels(  label_stringIdx_fit.labels)
>>> stages +=[indexToStringEstimator]
>>> pipeline = Pipeline().setStages(stages)
>>> pipelineModel = pipeline.fit(df)
22/02/07 17:05:54 WARN netlib.BLAS: Failed to load implementation from: com.github.fommil.netlib.NativeSystemBLAS
22/02/07 17:05:54 WARN netlib.BLAS: Failed to load implementation from: com.github.fommil.netlib.NativeRefBLAS
>>> pipelineModel.write().overwrite().save("my_LR_model8")
>>>
```

```
# В четвертом окне терминала hdfs dfs -ls
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

```
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls
Found 6 items
drwx------   - student898_2 student898_2          0 2022-02-07 15:00 .Trash
drwxr-xr-x   - student898_2 student898_2          0 2022-01-30 10:56 .sparkStaging
drwxr-xr-x   - student898_2 student898_2          0 2022-01-29 21:38 for_stream
drwxr-xr-x   - student898_2 student898_2          0 2022-01-29 21:49 input_csv_for_stream
drwxr-xr-x   - student898_2 student898_2          0 2022-02-07 17:06 my_LR_model8
drwxr-xr-x   - student898_2 student898_2          0 2022-02-07 15:10 parquet_data
[student898_2@bigdataanalytics-worker-3 ~]$
```

```
# Во втором окне терминала разделил датасет на train и test 70/30
train, test = df.randomSplit([0.7, 0.3], seed = 2018)
print("Training Dataset Count: " + str(train.count()))
print("Test Dataset Count: " + str(test.count()))
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

```
>>> train, test = df.randomSplit([0.7, 0.3], seed = 2018)
>>> print("Training Dataset Count: " + str(train.count()))
Training Dataset Count: 20664
>>> print("Test Dataset Count: " + str(test.count()))
Test Dataset Count: 8810
>>>
```

```
# трансформация  test, просмотр
pipelineModel.transform(test).show(100)
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-wor...  student898_2@bigdataanalytics-wor...  student898_2@bigdataanalytics-wor...  student898_2@bigdataanalytics-wor...
Test Dataset Count: 8810
>>> pipelineModel.transform(test).show(100)
+---+---+---+---+-----+----+----+------+------+-------+-------+-----------+--------------+--------+-------------+-----+--------------------+----------
----------+--------------------+----------+--------+
|  c| s1|ma1|mi1|   s2| ma2| mi2|gender|   age|user_id|segment|genderIndex|genderclassVec|ageIndex|  ageclassVec|label|            features|       raw
Prediction|         probability|prediction|category|
+---+---+---+---+-----+----+----+------+------+-------+-------+-----------+--------------+--------+-------------+-----+--------------------+----------
----------+--------------------+----------+--------+
|  1|  5|  5|  5|  315| 315| 315|     F| young|  24171|    shy|        0.0| (1,[0],[1.0])|     2.0|    (2,[],[])|  1.0|[1.0,0.0,0.0,1.0,...|[-3.575145
8342887...|[0.00127799518498...|       1.0|     shy|
|  1|  5|  5|  5| 1055|1055|1055|     M|midage|   9376|    shy|        1.0|     (1,[],[])|     1.0|(2,[1],[1.0])|  1.0|[0.0,0.0,1.0,1.0,...|[-3.886816
0537901...|[7.38882364438537...|       1.0|     shy|
|  1|  7|  7|  7| 4935|4935|4935|     F|midage|    463|    shy|        0.0| (1,[0],[1.0])|     1.0|(2,[1],[1.0])|  1.0|[1.0,0.0,1.0,1.0,...|[-7.354593
3390288...|[1.36048951709207...|       1.0|     shy|
|  1|  9|  9|  9| 4131|4131|4131|     F|midage|   3175|    shy|        0.0| (1,[0],[1.0])|     1.0|(2,[1],[1.0])|  1.0|[1.0,0.0,1.0,1.0,...|[-8.050101
7821170...|[4.68611483334799...|       1.0|     shy|
|  2|  6|  4|  2|  598| 440| 158|     M|   old|  29834|    shy|        1.0|     (1,[],[])|     0.0|(2,[0],[1.0])|  1.0|[0.0,1.0,0.0,2.0,...|[-1.800708
```

```
pipelineModel.transform(test).select("segment", "label", "probability", "prediction", "category").show(1)
pipelineModel.transform(test).select("segment", "label", "probability", "prediction", "category").show(10)
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-wor...  student898_2@bigdataanalytics-wor...  student898_2@bigdataanalytics-wor...  student898_2@bigdataanalytics-wor...
>>> pipelineModel.transform(test).select("segment", "label", "probability", "prediction", "category").show(1)
+-------+-----+--------------------+----------+--------+
|segment|label|         probability|prediction|category|
+-------+-----+--------------------+----------+--------+
|    shy|  1.0|[0.00127799518498...|       1.0|     shy|
+-------+-----+--------------------+----------+--------+
only showing top 1 row

>>> pipelineModel.transform(test).select("segment", "label", "probability", "prediction", "category").show(10)
+-------+-----+--------------------+----------+--------+
|segment|label|         probability|prediction|category|
+-------+-----+--------------------+----------+--------+
|    shy|  1.0|[0.00127799518498...|       1.0|     shy|
|    shy|  1.0|[7.38882364438537...|       1.0|     shy|
|    shy|  1.0|[1.36048951709207...|       1.0|     shy|
|    shy|  1.0|[4.68611483334799...|       1.0|     shy|
|    shy|  1.0|[0.02636789545321...|       1.0|     shy|
|    shy|  1.0|[0.04181979137745...|       1.0|     shy|
|    shy|  1.0|[0.03127384502272...|       1.0|     shy|
|    shy|  1.0|[0.03256454050483...|       2.0| neutral|
|    shy|  1.0|[0.00146472458139...|       1.0|     shy|
|    shy|  1.0|[9.56566232482099...|       1.0|     shy|
+-------+-----+--------------------+----------+--------+
only showing top 10 rows

>>>
```

stages

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-wor...  student898_2@bigdataanalytics-wor...  student898_2@bigdataanalytics-wor...  student898_2@bigdataanalytics-wor...

>>> stages
[StringIndexer_f311599de7bc, OneHotEncoderEstimator_ce0dbf596089, StringIndexer_c4cc4b820dae, OneHotEncoderEstimator_5d16b7b63811, StringIndexer_2b4dc
7f3854f, VectorAssembler_d0b3b62dae72, LogisticRegression_63f5c641f4af, IndexToString_578d3318e1da]
>>>
```

```
for categoricalCol in categoricalColumns:
    stringIndexer = StringIndexer(inputCol = categoricalCol, outputCol = categoricalCol + 'Index')
    encoder = OneHotEncoderEstimator(inputCols=[stringIndexer.getOutputCol()],
outputCols=[categoricalCol + "classVec"])
stages += [stringIndexer, encoder]
pipeline = Pipeline(stages = stages)
pipelineModel = pipeline.fit(df)
df = pipelineModel.transform(df)
```

cols = df.columns
selectedCols = ['label', 'features'] + cols
df = df.select(selectedCols)
df.printSchema()

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor...

```
>>> pipeline = Pipeline(stages = stages)
>>> pipelineModel = pipeline.fit(df)
>>> df = pipelineModel.transform(df)
>>> cols = df.columns
>>> selectedCols = ['label', 'features'] + cols
>>> df = df.select(selectedCols)
>>> df.printSchema()
root
 |-- label: double (nullable = false)
 |-- features: vector (nullable = true)
 |-- c: long (nullable = false)
 |-- s1: long (nullable = true)
 |-- ma1: integer (nullable = true)
 |-- mi1: integer (nullable = true)
 |-- s2: long (nullable = true)
 |-- ma2: integer (nullable = true)
 |-- mi2: integer (nullable = true)
 |-- gender: string (nullable = true)
 |-- age: string (nullable = true)
 |-- user_id: integer (nullable = true)
 |-- segment: string (nullable = true)
 |-- genderIndex: double (nullable = false)
 |-- genderclassVec: vector (nullable = true)
 |-- ageIndex: double (nullable = false)
 |-- ageclassVec: vector (nullable = true)
 |-- label: double (nullable = false)
 |-- features: vector (nullable = true)
 |-- rawPrediction: vector (nullable = true)
 |-- probability: vector (nullable = true)
 |-- prediction: double (nullable = false)
 |-- category: string (nullable = true)

>>>
```

train, test = df.randomSplit([0.7, 0.3], seed = 2018)
print("Training Dataset Count: " + str(train.count()))
print("Test Dataset Count: " + str(test.count()))

Делаю обучение модели
from pyspark.ml.classification import LogisticRegression
lr = LogisticRegression(featuresCol = 'features', labelCol = 'label', maxIter=10)
lrModel = lr.fit(train)

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor...

```
>>> train, test = df.randomSplit([0.7, 0.3], seed = 2018)
>>> print("Training Dataset Count: " + str(train.count()))
Training Dataset Count: 20664
>>> print("Test Dataset Count: " + str(test.count()))
Test Dataset Count: 8810
>>> from pyspark.ml.classification import LogisticRegression
>>> lr = LogisticRegression(featuresCol = 'features', labelCol = 'label', maxIter=10)
>>> lrModel = lr.fit(train)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/opt/spark-2.4.8/python/pyspark/ml/base.py", line 132, in fit
    return self._fit(dataset)
  File "/opt/spark-2.4.8/python/pyspark/ml/wrapper.py", line 295, in _fit
    java_model = self._fit_java(dataset)
  File "/opt/spark-2.4.8/python/pyspark/ml/wrapper.py", line 292, in _fit_java
    return self._java_obj.fit(dataset._jdf)
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 79, in deco
    raise IllegalArgumentException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.IllegalArgumentException: u'requirement failed: Column prediction already exists.'
>>>
```

```
lr = LogisticRegression(featuresCol = 'features', labelCol = 'label', maxIter=10)
stages += [lr]
label_stringIdx_fit = label_stringIdx.fit(users_known)
indexToStringEstimator =
IndexToString().setInputCol("prediction").setOutputCol("category").setLabels(  label_stringIdx_fit.labels)
stages +=[indexToStringEstimator]
```

Объединяю в новый pipeline, устанавливаю этапы, получаею pipelineModel, получаю скомпилированную модель

```
pipeline = Pipeline().setStages(stages)
pipelineModel = pipeline.fit(users_known)
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor...

```
>>> lr = LogisticRegression(featuresCol = 'features', labelCol = 'label', maxIter=10)
>>> stages += [lr]
>>> label_stringIdx_fit = label_stringIdx.fit(users_known)
>>> indexToStringEstimator = IndexToString().setInputCol("prediction").setOutputCol("category").setLabels(  label_stringIdx_fit.labels)
>>> stages +=[indexToStringEstimator]
>>> pipeline = Pipeline().setStages(stages)
>>> pipelineModel = pipeline.fit(users_known)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/opt/spark-2.4.8/python/pyspark/ml/base.py", line 132, in fit
    return self._fit(dataset)
  File "/opt/spark-2.4.8/python/pyspark/ml/pipeline.py", line 109, in _fit
    model = stage.fit(dataset)
  File "/opt/spark-2.4.8/python/pyspark/ml/base.py", line 132, in fit
    return self._fit(dataset)
  File "/opt/spark-2.4.8/python/pyspark/ml/wrapper.py", line 295, in _fit
    java_model = self._fit_java(dataset)
  File "/opt/spark-2.4.8/python/pyspark/ml/wrapper.py", line 292, in _fit_java
    return self._java_obj.fit(dataset._jdf)
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 79, in deco
    raise IllegalArgumentException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.IllegalArgumentException: u'requirement failed: Column prediction already exists.'
>>>
```

Сохраняю модель на HDFS

```
pipelineModel.write().overwrite().save("my_LR_model8")
pipelineModel.transform(test).select("segment", "category").show(100)
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor...

```
>>> pipelineModel.write().overwrite().save("my_LR_model8")
>>> pipelineModel.transform(test).select("segment", "category").show(100)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/opt/spark-2.4.8/python/pyspark/ml/base.py", line 173, in transform
    return self._transform(dataset)
  File "/opt/spark-2.4.8/python/pyspark/ml/pipeline.py", line 262, in _transform
    dataset = t.transform(dataset)
  File "/opt/spark-2.4.8/python/pyspark/ml/base.py", line 173, in transform
    return self._transform(dataset)
  File "/opt/spark-2.4.8/python/pyspark/ml/wrapper.py", line 312, in _transform
    return DataFrame(self._java_obj.transform(dataset._jdf), dataset.sql_ctx)
  File "/opt/spark-2.4.8/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/opt/spark-2.4.8/python/pyspark/sql/utils.py", line 79, in deco
    raise IllegalArgumentException(s.split(': ', 1)[1], stackTrace)
pyspark.sql.utils.IllegalArgumentException: u'requirement failed: Output column genderIndex already exists.'
>>>
```

Выполняю метод transform на вход подаю users_known (надо подавать test)

```
pipelineModel.transform(users_known).select("segment", "category").show(100)
```

student898_2@bigdataanalytics-worker-3:~ - Терминал

Файл Правка Вид Терминал Вкладки Справка

student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor... student898_2@bigdataanalytics-wor...

```
|segment|category|
+-------+--------+
|  happy|   happy|
|    shy|     shy|
|  happy|   happy|
|    shy| neutral|
|    shy|     shy|
|    shy| neutral|
|  happy|   happy|
|  happy|   happy|
|  happy|   happy|
|    shy|     shy|
|neutral| neutral|
|  happy|   happy|
|    shy|     shy|
|neutral| neutral|
|  happy| neutral|
|neutral|     shy|
|neutral|   happy|
|    shy|     shy|
|  happy|   happy|
|    shy|     shy|
|    shy| neutral|
|  happy|   happy|
|neutral|   happy|
|    shy|     shy|
|    shy|     shy|
|    shy|     shy|
|    shy|   happy|
|neutral| neutral|
|  happy| neutral|
|    shy|     shy|
|neutral|     shy|
|  happy|   happy|
|neutral|     shy|
|  happy|   happy|
|    shy|     shy|
|    shy|     shy|
|neutral|   happy|
|  happy|   happy|
|neutral| neutral|
|    shy|     shy|
|neutral| neutral|
|  happy|   happy|
|  happy|   happy|
|  happy|   happy|
|  happy|   happy|
|  happy|   happy|
|  happy| neutral|
|  happy|   happy|
|    shy| neutral|
|    shy| neutral|
|    shy|     shy|
|  happy|   happy|
|  happy|   happy|
|neutral|   happy|
|neutral|   happy|
|    shy|     shy|
```