camila-emr-workspace

March 16, 2024

[1]:
```
from pyspark.sql import SparkSession
from pyspark.sql.functions import col
```

```
VBox()
```

```
Starting Spark application
```

```
<IPython.core.display.HTML object>
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
SparkSession available as 'spark'.
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

[3]:
```
spark = SparkSession.builder.appName("CamilaRedfinDataAnalysis").getOrCreate()
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

[4]:
```
# extrai do bucket emr-camila-raw-data para o início das análises
redfin_data = spark.read.csv("s3://emr-camila-raw-data/city_market_tracker.
 ↪tsv000.gz", header=True, inferSchema=True, sep= "\t")
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

[5]:
```
redfin_data.show(3)
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
+------------+----------+---------------+-----------+--------------+--------+---
------------------+--------------------+-----------------+-------------+-------
---+--------------------+----------------+-----------------+--------------------
```

```
-+--------------------+----------------+--------------------+---------------
-----+------------------+--------------------+--------------------+-------------
-----+--------------------+--------------------+----------+-------------------+-
---------------+-------------+--------------------+------------------+--------
----+------------------+--------------------+---------+--------------------+------
-------------+----------------+--------------------+--------------------+------
----+--------------+--------------+------------------+--------------------+-----
--------------+--------------------+--------------------+--------------------+--
----------------+--------------------+--------------------+--------------------
---+----------------------------+----------------------------+-------------------+
------------------------------+-------------------+
|period_begin|period_end|period_duration|region_type|region_type_id|table_id|is_
seasonally_adjusted|              region|             city|
state|state_code|       property_type|property_type_id|median_sale_price|median_
sale_price_mom|median_sale_price_yoy|median_list_price|median_list_price_mom|med
ian_list_price_yoy|       median_ppsf|     median_ppsf_mom|     median_ppsf_yoy|
median_list_ppsf|median_list_ppsf_mom|median_list_ppsf_yoy|homes_sold|
homes_sold_mom|    homes_sold_yoy|pending_sales|   pending_sales_mom|
pending_sales_yoy|new_listings|  new_listings_mom|   new_listings_yoy|inventory|
inventory_mom|       inventory_yoy|months_of_supply|months_of_supply_mom|months_
of_supply_yoy|median_dom|median_dom_mom|median_dom_yoy|
avg_sale_to_list|avg_sale_to_list_mom|avg_sale_to_list_yoy|    sold_above_list|
sold_above_list_mom| sold_above_list_yoy|        price_drops|
price_drops_mom|     price_drops_yoy|off_market_in_two_weeks|off_market_in_two_w
eeks_mom|off_market_in_two_weeks_yoy|parent_metro_region|parent_metro_region_met
ro_code|       last_updated|
+------------+----------+--------------+-----------+-------------+--------+---
------------------+--------------------+----------------+-------------+-------
---+--------------------+----------------+-----------------+--------------------
-+--------------------+-----------------+--------------------+----------------
-----+------------------+--------------------+--------------------+-------------
-----+--------------------+--------------------+----------+-------------------+-
---------------+-------------+--------------------+------------------+--------
----+------------------+--------------------+---------+--------------------+------
-------------+----------------+--------------------+--------------------+------
----+--------------+--------------+------------------+--------------------+-----
--------------+--------------------+--------------------+--------------------+--
----------------+--------------------+--------------------+--------------------
---+----------------------------+----------------------------+-------------------+
------------------------------+-------------------+
|  2021-02-01|2021-02-28|             30|      place|             6|    5779|
f|       Fair Lawn, NJ|        Fair Lawn|   New Jersey|        NJ|Single Family
Res…|               6|         542500.0|  0.21378230227094752|
0.35964912280701755|         469000.0| -0.14711765775595564|
0.04454342984409809| 278.1372476167421|-0.03125719269388…|-0.00459122296016…
|291.19741100323625|-0.02531486920188…|-0.02420798775354…|
20|-0.4117647058823529|0.4285714285714286|
14|-0.26315789473684215|0.2727272727272727|
```

```
25|0.7857142857142858|-0.3902439024390244|       69|
0.0|-0.34905660377358494|             3.5|                1.5|
-4.1|        91|             8|            -2|
1.0|-0.01214315981126557|0.020401919432001314|               0.0|-0.20588235294
117646|-0.14285714285714285|0.08695652173913043|-0.07246376811594202|
0.03978671041837571|    0.21428571428571427|        0.10902255639097744|
-0.05844155844155843|       New York, NY|
35614|2024-03-10 14:36:40|
|  2019-04-01|2019-04-30|             30|      place|             6|    6504|
f|          Elyria, OH|           Elyria|         Ohio|        OH|Multi-Family
(2-4…|               4|          30000.0|  -0.7014925373134329|
-0.6557165399512266|         110000.0| -0.04264577893820709|
-0.06542056074766356|18.610421836228287| -0.6026777101693752|
-0.5572373301855638|
81.52173913043478|0.004286982260352179|0.005327375837772763|         3|
0.5|               0.5|            2|                null|               1.0|
4|               3.0|0.33333333333333326|        7|                0.0|
-0.5|             2.3| -1.2000000000000002|               -4.7|       130|
-149|            33|
0.860358532597008|-0.02600566975054…|0.007997142263547308|               0.0|
0.0|                 0.0|               null|               null|
null|                    1.0|                        null|
1.0|      Cleveland, OH|                         17460|2024-03-10 14:36:40|
|  2020-08-01|2020-08-31|             30|      place|             6|   24459|
f|Northwest Harwich…|Northwest Harwich|Massachusetts|        MA|Single Family
Res…|               6|         625000.0|   0.3789299503585217|
-0.14617486338797814|         649000.0|  0.26019417475728157|
0.42668718399648276|340.65934065934067| 0.12348060103162162|
0.31284153005464477| 393.0817610062893|  0.2546073635603787|
0.18042570872759556|        11|0.22222222222222232|2.6666666666666665|
8| -0.4285714285714286|               1.0|
14|0.5555555555555556|0.07692307692307687|       14|0.07692307692307687|
-0.7407407407407407|             1.3|-0.09999999999999987|               -16.7|
26|           -51|            12|0.9771452130091354|0.022599268740304268|-0.0270
9829717127…|0.18181818181818182| 0.07070707070707072| -0.4848484848484848|
0.2857142857142857|-0.09890109890109894|0.007936507936507908|
0.5|         0.4285714285714286|                        0.5|Barnstable Town, MA|
12700|2024-03-10 14:36:40|
+------------+----------+--------------+-----------+-------------+--------+---
------------------+--------------------+-----------------+-------------+-------
---+--------------------+----------------+-----------------+--------------------
-+--------------------+-----------------+---------------------+----------------
-----+------------------+--------------------+--------------------+-------------
-----+--------------------+--------------------+----------+--------------------+-
----------------+-------------+--------------------+------------------+--------
----+------------------+------------------+---------+------------------+------
-------------+----------------+--------------------+--------------------+------
----+--------------+--------------+------------------+--------------------+-----
```

```
--------------+------------------+--------------------+--------------------+--
----------------+--------------------+--------------------+--------------------
---+--------------------------+--------------------------+-------------------+
------------------------------+-------------------+
only showing top 3 rows
```

[6]:
```
#visualiza o schema
redfin_data.printSchema()
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
↪layout=Layout(height='25px', width='50%'),…
```

```
root
 |-- period_begin: date (nullable = true)
 |-- period_end: date (nullable = true)
 |-- period_duration: integer (nullable = true)
 |-- region_type: string (nullable = true)
 |-- region_type_id: integer (nullable = true)
 |-- table_id: integer (nullable = true)
 |-- is_seasonally_adjusted: string (nullable = true)
 |-- region: string (nullable = true)
 |-- city: string (nullable = true)
 |-- state: string (nullable = true)
 |-- state_code: string (nullable = true)
 |-- property_type: string (nullable = true)
 |-- property_type_id: integer (nullable = true)
 |-- median_sale_price: double (nullable = true)
 |-- median_sale_price_mom: double (nullable = true)
 |-- median_sale_price_yoy: double (nullable = true)
 |-- median_list_price: double (nullable = true)
 |-- median_list_price_mom: double (nullable = true)
 |-- median_list_price_yoy: double (nullable = true)
 |-- median_ppsf: double (nullable = true)
 |-- median_ppsf_mom: double (nullable = true)
 |-- median_ppsf_yoy: double (nullable = true)
 |-- median_list_ppsf: double (nullable = true)
 |-- median_list_ppsf_mom: double (nullable = true)
 |-- median_list_ppsf_yoy: double (nullable = true)
 |-- homes_sold: integer (nullable = true)
 |-- homes_sold_mom: double (nullable = true)
 |-- homes_sold_yoy: double (nullable = true)
 |-- pending_sales: integer (nullable = true)
 |-- pending_sales_mom: double (nullable = true)
 |-- pending_sales_yoy: double (nullable = true)
 |-- new_listings: integer (nullable = true)
 |-- new_listings_mom: double (nullable = true)
 |-- new_listings_yoy: double (nullable = true)
```

```
 |-- inventory: integer (nullable = true)
 |-- inventory_mom: double (nullable = true)
 |-- inventory_yoy: double (nullable = true)
 |-- months_of_supply: double (nullable = true)
 |-- months_of_supply_mom: double (nullable = true)
 |-- months_of_supply_yoy: double (nullable = true)
 |-- median_dom: integer (nullable = true)
 |-- median_dom_mom: integer (nullable = true)
 |-- median_dom_yoy: integer (nullable = true)
 |-- avg_sale_to_list: double (nullable = true)
 |-- avg_sale_to_list_mom: double (nullable = true)
 |-- avg_sale_to_list_yoy: double (nullable = true)
 |-- sold_above_list: double (nullable = true)
 |-- sold_above_list_mom: double (nullable = true)
 |-- sold_above_list_yoy: double (nullable = true)
 |-- price_drops: double (nullable = true)
 |-- price_drops_mom: double (nullable = true)
 |-- price_drops_yoy: double (nullable = true)
 |-- off_market_in_two_weeks: double (nullable = true)
 |-- off_market_in_two_weeks_mom: double (nullable = true)
 |-- off_market_in_two_weeks_yoy: double (nullable = true)
 |-- parent_metro_region: string (nullable = true)
 |-- parent_metro_region_metro_code: integer (nullable = true)
 |-- last_updated: timestamp (nullable = true)
```

[7]:
```
#print os nomes das colunas
redfin_data.columns
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
↪layout=Layout(height='25px', width='50%'),…
```

```
['period_begin', 'period_end', 'period_duration', 'region_type',
'region_type_id', 'table_id', 'is_seasonally_adjusted', 'region', 'city',
'state', 'state_code', 'property_type', 'property_type_id', 'median_sale_price',
'median_sale_price_mom', 'median_sale_price_yoy', 'median_list_price',
'median_list_price_mom', 'median_list_price_yoy', 'median_ppsf',
'median_ppsf_mom', 'median_ppsf_yoy', 'median_list_ppsf',
'median_list_ppsf_mom', 'median_list_ppsf_yoy', 'homes_sold', 'homes_sold_mom',
'homes_sold_yoy', 'pending_sales', 'pending_sales_mom', 'pending_sales_yoy',
'new_listings', 'new_listings_mom', 'new_listings_yoy', 'inventory',
'inventory_mom', 'inventory_yoy', 'months_of_supply', 'months_of_supply_mom',
'months_of_supply_yoy', 'median_dom', 'median_dom_mom', 'median_dom_yoy',
'avg_sale_to_list', 'avg_sale_to_list_mom', 'avg_sale_to_list_yoy',
'sold_above_list', 'sold_above_list_mom', 'sold_above_list_yoy', 'price_drops',
'price_drops_mom', 'price_drops_yoy', 'off_market_in_two_weeks',
'off_market_in_two_weeks_mom', 'off_market_in_two_weeks_yoy',
'parent_metro_region', 'parent_metro_region_metro_code', 'last_updated']
```

[8]:
```
df_redfin = redfin_data.select(['period_end','period_duration', 'city',␣
 ↪'state', 'property_type',
    'median_sale_price', 'median_ppsf', 'homes_sold', 'inventory',␣
 ↪'months_of_supply', 'median_dom', 'sold_above_list', 'last_updated'])
df_redfin.show(3)
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
+----------+--------------+-----------------+-------------+--------------------
+-----------------+------------------+----------+---------+----------------+----
------+-------------------+-------------------+
|period_end|period_duration|             city|        state|
property_type|median_sale_price|
median_ppsf|homes_sold|inventory|months_of_supply|median_dom|
sold_above_list|       last_updated|
+----------+--------------+-----------------+-------------+--------------------
+-----------------+------------------+----------+---------+----------------+----
------+-------------------+-------------------+
|2021-02-28|             30|        Fair Lawn|   New Jersey|Single Family
Res…|         542500.0| 278.1372476167421|         20|       69|
3.5|         91|                0.0|2024-03-10 14:36:40|
|2019-04-30|             30|           Elyria|         Ohio|Multi-Family
(2-4…|          30000.0|18.610421836228287|          3|        7|
2.3|        130|                0.0|2024-03-10 14:36:40|
|2020-08-31|             30|Northwest Harwich|Massachusetts|Single Family
Res…|         625000.0|340.65934065934067|         11|       14|
1.3|         26|0.18181818181818182|2024-03-10 14:36:40|
+----------+--------------+-----------------+-------------+--------------------
+-----------------+------------------+----------+---------+----------------+----
------+-------------------+-------------------+
only showing top 3 rows
```

[9]:
```
#verifica o número total de linhas
print(f"Total number of rows: {df_redfin.count()}")
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
Total number of rows: 5245047
```

[10]:
```
from pyspark.sql.functions import isnull
#Conta valores nulos em cada coluna
#usamos uma list comprehension
```

```
null_counts = [df_redfin.where(isnull(col_name)).count() for col_name in␣
 ↪df_redfin.columns]
null_counts
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
[0, 0, 0, 0, 0, 6066, 69125, 5646, 413699, 334643, 69227, 36370, 0]
```

[11]:
```
#Exibe os resultados
for i, col_name in enumerate(df_redfin.columns):
    print(f"{col_name}: {null_counts[i]} null values")
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
period_end: 0 null values
period_duration: 0 null values
city: 0 null values
state: 0 null values
property_type: 0 null values
median_sale_price: 6066 null values
median_ppsf: 69125 null values
homes_sold: 5646 null values
inventory: 413699 null values
months_of_supply: 334643 null values
median_dom: 69227 null values
sold_above_list: 36370 null values
last_updated: 0 null values
```

[12]:
```
# Verifica se há valores faltantes em todo o DataFrame
remaining_count = df_redfin.na.drop().count()

print(f"Number of missing rows: {df_redfin.count() - remaining_count}")
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
Number of missing rows: 501546
```

[13]:
```
print(f"Total number of remaining rows: {remaining_count}")
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
Total number of remaining rows: 4743501
```

[14]:
```
#remove NA e conta o número total de linhas restantes
df_redfin = df_redfin.na.drop()
print(f"Total number of rows: {df_redfin.count()}")
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
Total number of rows: 4743501
```

[15]:
```
# Conta os valores nulos em cada coluna para confirmar se removemos todos os␣
 ↪na(tem q ficar tudo zero)
null_counts = [df_redfin.where(isnull(col_name)).count() for col_name in␣
 ↪df_redfin.columns]
null_counts
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

```
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

[16]:
```
from pyspark.sql.functions import year, month

#Extrai o ano de period_end e salva em uma nova coluna "period_end_yr"
df_redfin = df_redfin.withColumn("period_end_yr", year(col("period_end")))

#Extrai mês de period_end e salva em uma nova coluna "period_end_month"
df_redfin = df_redfin.withColumn("period_end_month", month(col("period_end")))
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

[17]:
```
# Elimine as colunas period_end e last_updated
df_redfin = df_redfin.drop("period_end", "last_updated")
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
 ↪layout=Layout(height='25px', width='50%'),…
```

[18]:
```
df_redfin.show(3)
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
  ↪layout=Layout(height='25px', width='50%'),…
```

```
+--------------+----------------+------------+--------------------+----------
-------+-----------------+----------+---------+---------------+----------+----
--------------+-------------+----------------+
|period_duration|            city|       state|
property_type|median_sale_price|
median_ppsf|homes_sold|inventory|months_of_supply|median_dom|
sold_above_list|period_end_yr|period_end_month|
+--------------+----------------+------------+--------------------+----------
-------+-----------------+----------+---------+---------------+----------+----
--------------+-------------+----------------+
|             30|        Fair Lawn|   New Jersey|Single Family Res…|
542500.0| 278.1372476167421|        20|       69|             3.5|        91|
0.0|         2021|               2|
|             30|           Elyria|         Ohio|Multi-Family (2-4…|
30000.0|18.610421836228287|         3|        7|             2.3|       130|
0.0|         2019|               4|
|             30|Northwest Harwich|Massachusetts|Single Family Res…|
625000.0|340.65934065934067|        11|       14|             1.3|
26|0.18181818181818182|         2020|               8|
+--------------+----------------+------------+--------------------+----------
-------+-----------------+----------+---------+---------------+----------+----
--------------+-------------+----------------+
only showing top 3 rows
```

[19]:
```
from pyspark.sql.functions import when

#altera o número do mês para o respectivo nome do mês.

df_redfin = df_redfin.withColumn("period_end_month",
                  when(col("period_end_month") == 1, "January")
                  .when(col("period_end_month") == 2, "February")
                  .when(col("period_end_month") == 3, "March")
                  .when(col("period_end_month") == 4, "April")
                  .when(col("period_end_month") == 5, "May")
                  .when(col("period_end_month") == 6, "June")
                  .when(col("period_end_month") == 7, "July")
                  .when(col("period_end_month") == 8, "August")
                  .when(col("period_end_month") == 9, "September")
                  .when(col("period_end_month") == 10, "October")
                  .when(col("period_end_month") == 11, "November")
                  .when(col("period_end_month") == 12, "December")
                  .otherwise("Unknown")
                )
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
↪layout=Layout(height='25px', width='50%'),…
```

[20]:
```
df_redfin.show(3)
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
↪layout=Layout(height='25px', width='50%'),…
```

```
+--------------+-----------------+------------+--------------------+----------
-------+------------------+----------+---------+----------------+----------+----
--------------+------------+----------------+
|period_duration|             city|        state|
property_type|median_sale_price|
median_ppsf|homes_sold|inventory|months_of_supply|median_dom|
sold_above_list|period_end_yr|period_end_month|
+--------------+-----------------+------------+--------------------+----------
-------+------------------+----------+---------+----------------+----------+----
--------------+------------+----------------+
|             30|        Fair Lawn|   New Jersey|Single Family Res…|
542500.0| 278.1372476167421|        20|       69|             3.5|        91|
0.0|         2021|        February|
|             30|           Elyria|         Ohio|Multi-Family (2-4…|
30000.0|18.610421836228287|         3|        7|             2.3|       130|
0.0|         2019|           April|
|             30|Northwest Harwich|Massachusetts|Single Family Res…|
625000.0|340.65934065934067|        11|       14|             1.3|
26|0.18181818181818182|         2020|          August|
+--------------+-----------------+------------+--------------------+----------
-------+------------------+----------+---------+----------------+----------+----
--------------+------------+----------------+
only showing top 3 rows
```

[22]:
```
#salva dataframe final transformado no outro bucket s3 como um arquivo parquet.
s3_bucket = "s3://emr-camila-transformed-data/redfin_data.parquet"
df_redfin.write.mode("overwrite").parquet(s3_bucket)
```

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:',␣
↪layout=Layout(height='25px', width='50%'),…
```