dashboard

December 23, 2020

[4]:

```python
import seaborn as sns
from pyspark import SparkContext
from pyspark.sql import SparkSession
from pyspark.streaming import StreamingContext
from pyspark.ml.feature import PCA, RFormula
import pandas as pd
import json
import matplotlib.pyplot as plt
import numpy as np
from datetime import datetime
```

0.0.1 Connect to Spark

[5]:

```python
ss = SparkSession.Builder() \
     .appName("DashBoard") \
     .master("spark://post-batch-processing-spark-master:7077") \
     .getOrCreate()
```

0.0.2 Read data from parquet file "trips.parquet" in hdfs

[6]:

```python
df = ss.read.parquet("hdfs://namenode:9000/trips/trips.parquet")
```

[7]:

```python
print(f"Number of records: {df.count()}")
df = df.sort('ArrivalTime')
```

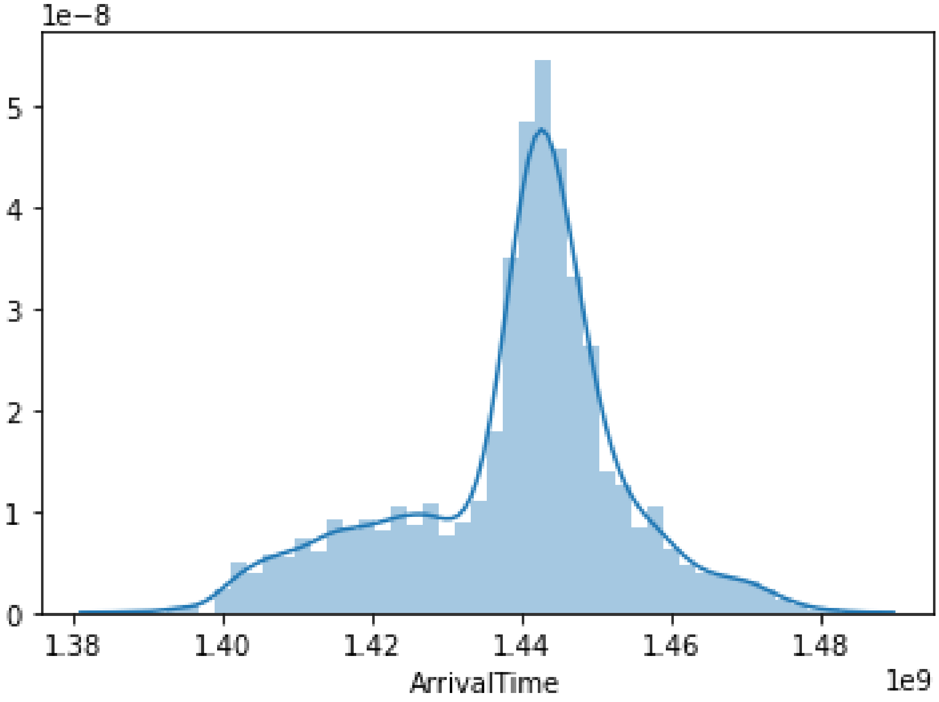
```
Number of records: 2313
```

1 Data Mining

1.0.1 Distribution of trips over time

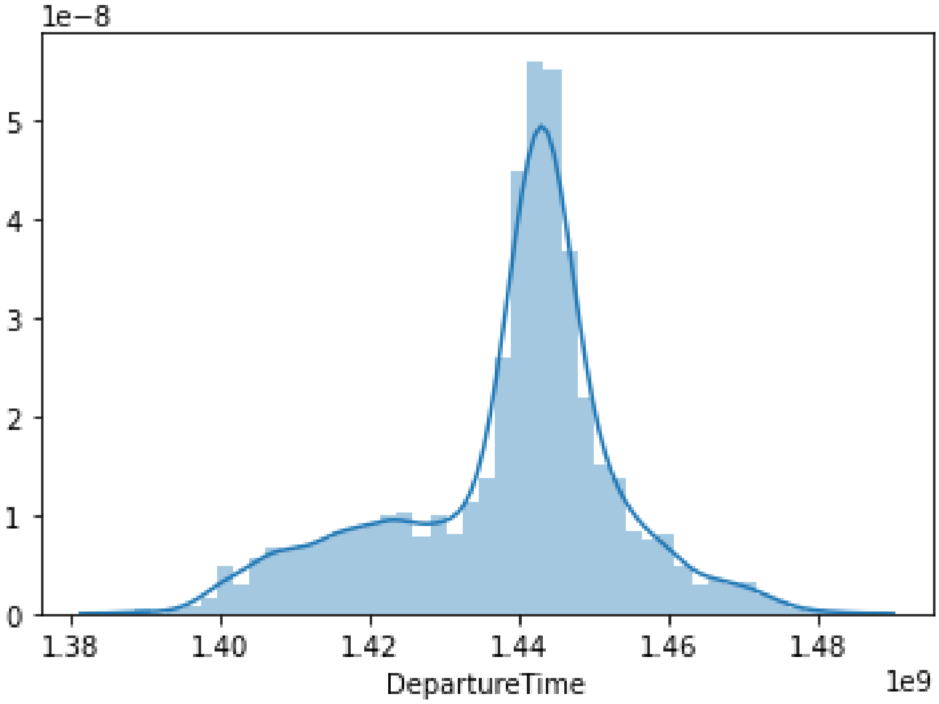
[8]:
```
arrivalTime = df.select('ArrivalTime').toPandas()['ArrivalTime'].astype('int64')
sns.distplot(arrivalTime)
```

[8]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7d0fbe1a50>
```



[9]:
```
arrivalTime = df.select('DepartureTime').toPandas()['DepartureTime'].
 ↪astype('int64')
sns.distplot(arrivalTime)
```

[9]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7d407645d0>
```



1.0.2 Top 10 most visited destinations in the last year

```
[10]: arrivalTime = df.select('DepartureTime').toPandas()['DepartureTime'].
       ↪astype('int32')
      thirty_days = 86400 * 30 * 12
      lastRecordTime = arrivalTime.iloc[-1]
      pivot = int(lastRecordTime - thirty_days)
      s = df.filter(df.DepartureTime > pivot).select('Destination')
```

```
[11]: s.head(10)
```

```
[11]: [Row(Destination='HAV'),
       Row(Destination='BCN'),
       Row(Destination='SJO'),
       Row(Destination='PEK'),
       Row(Destination='JNB'),
       Row(Destination='BMA'),
       Row(Destination='BGA'),
       Row(Destination='VIE'),
       Row(Destination='HRE'),
       Row(Destination='MSP')]
```

2 Anomaly detection

2.0.1 PCA

[12]:

```
df = ss.read.parquet("hdfs://namenode:9000/trips/processed_trips.parquet")
```

[13]:

```
print(f"Number of records: {df.count()}")
df = df.sort('ArrivalTime')
```

```
Number of records: 2609
```

[14]:

```
df = df.select(
    'ArrivalTime',
    'BusinessLeisure',
    'CabinCategory',
    'CreationDate',
    'CurrencyCode',
    'DepartureTime',
    'Destination',
    'OfficeIdCountry',
    'Origin',
    'TotalAmount',
    'nPAX'
)
```

[15]:

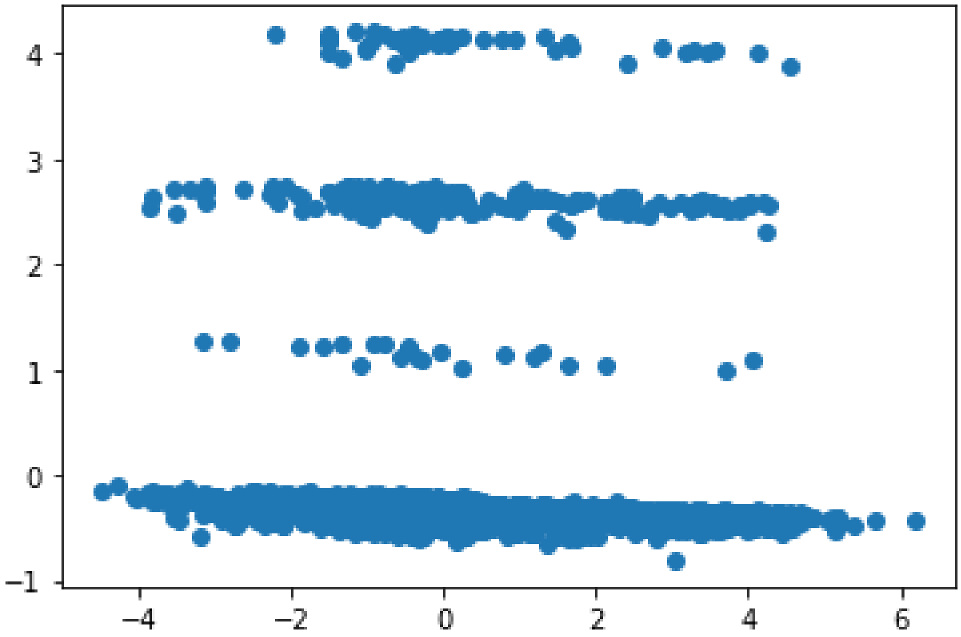
```
pca = PCA().setInputCol("features").setK(2)
data = RFormula(formula=" ~ {0}".format(" + ".join(df.columns))).fit(df).
 ↪transform(df)
s = pca.fit(data).transform(data)
r = s.select(s.columns[-1]).toPandas()[s.columns[-1]]

X = []
Y = []
for i in range(len(r)):
    X.append(r[i][0])
    Y.append(r[i][1])
```

[16]:

```
plt.scatter(X, Y)
```

[16]:

```
<matplotlib.collections.PathCollection at 0x7f7d406ecc10>
```


4
3
2
1
0
−1
−4
−2
0
2
4
6

```
[ ]:
```