Assignment 7

April 30, 2023

Gabriel Avinaz
Week 7
4/26/23

0.1 Assignment 7.1

0.1.1 Part a

[ ]:
```
import os
import shutil
import pandas as pd
```

[ ]:
```
path = "data"
if not os.path.exists(path):
    os.makedirs(path)
    shutil.copy("../assignment03/results/routes.parquet", path)
```

[ ]:
```
routes_df = pd.read_parquet("data/routes.parquet", engine='pyarrow')
routes_df = pd.json_normalize(routes_df["Flight Info"])
routes_df.sample(10)
```

[ ]:
```
       codeshare   equipment  airline.active  airline.airline_id  \
18187      False  [320, E90]            True                1767
14722      False  [320, 767]            True                1355
45345      False       [DH1]            True                  -1
13988      False       [77W]            True                3029
15003      False       [737]            True                4740
40351      False       [737]            True                5484
59592      False  [757, 763]            True                5265
24070       True       [330]            True                2222
21260       True  [CRJ, ERJ]            True                2009
66593      False       [AT7]            True                  -1

                    airline.alias airline.callsign      airline.country  \
18187         SN Brussels Airlines   CHINA SOUTHERN                China
14722  Pulkovo Aviation Enterprise        SPEEDBIRD       United Kingdom
45345                           \N               \N                   \N
13988                          nan          JETBLUE        United States
15003               Swiss European          SKYMARK                Japan
```

```
40351                        Varig       XIAMEN AIR                 China
59592                          TWA          U S AIR         United States
24070           Emirates Airlines           ETIHAD  United Arab Emirates
21260          CSA Czech Airlines            DELTA         United States
66593                          \N              \N                    \N

      airline.iata airline.icao             airline.name  …  \
18187           CZ           CSN  China Southern Airlines  …
14722           BA           BAW          British Airways  …
45345            -           nan                  Unknown  …
13988           B6           JBU          JetBlue Airways  …
15003           BC           SKY         Skymark Airlines  …
40351           MF           CXA          Xiamen Airlines  …
59592           US           USA               US Airways  …
24070           EY           ETD           Etihad Airways  …
21260           DL           DAL          Delta Air Lines  …
66593            -           nan                  Unknown  …

       src_airport.icao  src_airport.latitude src_airport.longitude  \
18187              ZGGG             23.392401            113.299004
14722              EGLL             51.470600             -0.461941
45345              CYWK             52.921902            -66.864403
13988              OMDB             25.252800             55.364399
15003              RJAA             35.764702            140.386002
40351              ZGSZ             22.639299            113.810997
59592              SPIM            -12.021900            -77.114305
24070              OMAA             24.433001             54.651100
21260              KMEM             35.042400            -89.976700
66593              HTDA             -6.878110             39.202599

                         src_airport.name src_airport.source  \
18187  Guangzhou Baiyun International Airport        OurAirports
14722                London Heathrow Airport        OurAirports
45345                        Wabush Airport        OurAirports
13988           Dubai International Airport        OurAirports
15003          Narita International Airport        OurAirports
40351   Shenzhen Bao'an International Airport        OurAirports
59592      Jorge Chávez International Airport        OurAirports
24070        Abu Dhabi International Airport        OurAirports
21260          Memphis International Airport        OurAirports
66593    Julius Nyerere International Airport        OurAirports

      src_airport.timezone src_airport.type     src_airport.tz_id  \
18187                  8.0          airport         Asia/Shanghai
14722                  0.0          airport         Europe/London
45345                 -4.0          airport       America/Halifax
13988                  4.0          airport            Asia/Dubai
```

```
15003                  9.0          airport            Asia/Tokyo
40351                  8.0          airport          Asia/Shanghai
59592                 -5.0          airport           America/Lima
24070                  4.0          airport             Asia/Dubai
21260                 -6.0          airport        America/Chicago
66593                  3.0          airport   Africa/Dar_es_Salaam

       dst_airport src_airport
18187          NaN         NaN
14722          NaN         NaN
45345          NaN         NaN
13988          NaN         NaN
15003          NaN         NaN
40351          NaN         NaN
59592          NaN         NaN
24070          NaN         NaN
21260          NaN         NaN
66593          NaN         NaN

[10 rows x 40 columns]
```

[ ]:
```
routes_df = routes_df.dropna(subset=['src_airport.iata', 'dst_airport.iata',␣
 ↪'airline.iata'])
routes_df.isna().sum()
```

[ ]:
```
codeshare                   0
equipment                   0
airline.active              0
airline.airline_id          0
airline.alias               0
airline.callsign            0
airline.country             0
airline.iata                0
airline.icao                0
airline.name                0
dst_airport.airport_id      0
dst_airport.altitude        0
dst_airport.city            0
dst_airport.country         0
dst_airport.dst             0
dst_airport.iata            0
dst_airport.icao            0
dst_airport.latitude        0
dst_airport.longitude       0
dst_airport.name            0
dst_airport.source          0
dst_airport.timezone        0
```

```
dst_airport.type              0
dst_airport.tz_id             0
src_airport.airport_id        0
src_airport.altitude          0
src_airport.city              0
src_airport.country           0
src_airport.dst               0
src_airport.iata              0
src_airport.icao              0
src_airport.latitude          0
src_airport.longitude         0
src_airport.name              0
src_airport.source            0
src_airport.timezone          0
src_airport.type              0
src_airport.tz_id             0
dst_airport               66771
src_airport               66771
dtype: int64
```

[ ]:
```
def key_gen(row):
    return str(row['src_airport.iata']) + str(row['dst_airport.iata']) +␣
 ↪str(row['airline.iata'])
```

[ ]:
```
routes_df["key"] = routes_df.apply(key_gen, axis=1)
routes_df[['src_airport.iata', 'dst_airport.iata', 'airline.iata', 'key']].
 ↪sample(10)
```

[ ]:

	src_airport.iata	dst_airport.iata	airline.iata	key
66766	CTU	NKG	ZH	CTUNKGZH
16546	KWL	CKG	CA	KWLCKGCA
16142	CKG	KMG	CA	CKGKMGCA
36691	ILM	ATL	KL	ILMATLKL
5555	IAH	ORD	AA	IAHORDAA
24864	ATL	CUN	FL	ATLCUNFL
59322	GSO	PHL	US	GSOPHLUS
8710	MAO	TFF	AD	MAOTFFAD
3952	AUH	TRV	9W	AUHTRV9W
56604	IAD	BOG	UA	IADBOGUA

[ ]:
```
partitions = (
    ('A', 'A'), ('B', 'B'), ('C', 'D'), ('E', 'F'),
    ('G', 'H'), ('I', 'J'), ('K', 'L'), ('M', 'M'),
    ('N', 'N'), ('O', 'P'), ('Q', 'R'), ('S', 'T'),
    ('U', 'U'), ('V', 'V'), ('W', 'X'), ('Y', 'Z')
)
```

```
def kv_key_gen(row):
    for tuple in partitions:
        first_char = row["key"][:1]
        if first_char >= tuple[0] and first_char <= tuple[1]:
            if tuple[0] == tuple[1]:
                return tuple[0]
            else:
                return tuple[0] + '-' + tuple[1]
```

[ ]:
```
routes_df["kv_key"] = routes_df.apply(kv_key_gen, axis=1)
routes_df[['src_airport.iata', 'dst_airport.iata', 'airline.iata', 'key',␣
 ↪'kv_key']].sample(10)
```

[ ]:

	src_airport.iata	dst_airport.iata	airline.iata	key	kv_key
13295	FCO	FRA	AZ	FCOFRAAZ	E-F
15622	GUM	TPE	BR	GUMTPEBR	G-H
26978	HHN	TMP	FR	HHNTMPFR	G-H
44848	SPU	CDG	OU	SPUCDGOU	S-T
43325	SRA	GEL	nan	SRAGELnan	S-T
39960	OZH	VKO	M9	OZHVKOM9	O-P
33157	LIN	NAP	IG	LINNAPIG	K-L
23960	CTU	CAN	EU	CTUCANEU	C-D
54917	LGW	ALC	U2	LGWALCU2	K-L
24914	BDL	ATL	FL	BDLATLFL	B

[ ]:
```
routes_df.to_parquet('results/kv/', partition_cols=['kv_key'])
```

0.1.2 Part b

[ ]:
```
import hashlib

def hash_key(key):
    m = hashlib.sha256()
    m.update(str(key).encode('utf-8'))
    return m.hexdigest()
```

[ ]:
```
routes_df["hashed"] = routes_df.apply(lambda x: hash_key(x["key"]), axis=1)
routes_df[['src_airport.iata', 'dst_airport.iata', 'airline.iata', 'key',␣
 ↪'hashed']].sample(10)
```

[ ]:

	src_airport.iata	dst_airport.iata	airline.iata	key	\
3054	WLK	OTZ	8E	WLKOTZ8E	
42837	FUK	FUJ	NH	FUKFUJNH	
18104	CAN	AVA	CZ	CANAVACZ	
57993	JED	HBE	UJ	JEDHBEUJ	
6558	PHX	MSP	AA	PHXMSPAA	
37910	LUX	FAO	LG	LUXFAOLG	

```
46589              AUU              CNS            Q6  AUUCNSQ6
15211              LBA              GLA            BE  LBAGLABE
3181               WUH              LJG            8L  WUHLJG8L
42386              TSN              WUH            MU  TSNWUHMU

                                                    hashed
3054   04154d9573f0af18b1c17858d6d0337e7e9926e20be767…
42837  4a84f0ec368ac45d25f43b443b82b5de7eae34dbc719e2…
18104  4ee8d3847d5766ed4b9c931eec9cb31e1a48e8ca22f37a…
57993  00dd8987322b2b1db7e8806e199457bdc7461c8d65a6f9…
6558   6370b2725731c9b6b3425e80aab7650563e46d590f740f…
37910  3721338429aa3b26ed787c62ad5d3e14ba635640a49996…
46589  0c9ec1de06445f3613f75582e3434189da6a9be463b6c0…
15211  bacfa6ecd470a65363de0f75c99700b3fde776dedfdf26…
3181   149d47f0899102a4d9957d454f389ff6f7acfb62af74b0…
42386  c8943f873f0588e4bb4292acb5fc47a4ba994dbd6117ef…
```

[ ]:
```
routes_df["hash_key"] = routes_df.apply(lambda x: x["hashed"][0].upper(),␣
↪axis=1)
```

[ ]:
```
routes_df.to_parquet('results/hash/', partition_cols=['hash_key'])
```

0.1.3 Part c

[ ]:
```
import pygeohash
```

[ ]:
```
data_centers = {'Location': ['West', 'Central', 'East'],
        'City': ['The Dalles, Oregon', 'Papillion, NE', 'Loudoun County,␣
↪Virginia'],
        'Latitude': [45.5945645, 41.1544433, 39.08344],
        'Longitude': [-121.1786823, -96.0422378, -77.6497145]}

data_center_df = pd.DataFrame(data_centers)
data_center_df
```

[ ]:
```
  Location                      City   Latitude   Longitude
0     West        The Dalles, Oregon  45.594564 -121.178682
1  Central             Papillion, NE  41.154443  -96.042238
2     East  Loudoun County, Virginia  39.083440  -77.649715
```

[ ]:
```
# Assignment 7.1.c
routes_df['src_airport_geohash'] = routes_df.apply(
    lambda row: pygeohash.encode(row['src_airport.latitude'], row['src_airport.
↪longitude']), axis=1
)
def determine_location(src_airport_geohash):
    locations = dict(
```

```
        central=pygeohash.encode(41.1544433, -96.0422378),
        east=pygeohash.encode(39.083440, -77.649715),
        west=pygeohash.encode(45.594564, -121.178682)
        ## TODO: add west and east
    )

    distances = [[pygeohash.geohash_haversine_distance(locations[center],␣
 ↪src_airport_geohash), center] for center in locations]

    distances.sort()
    return distances[0][1]
routes_df['location'] = routes_df['src_airport_geohash'].
 ↪apply(determine_location)
routes_df.to_parquet('results/geo', partition_cols=['location'])
```

[ ]:
```
routes_df[['src_airport.iata', 'src_airport.latitude', 'src_airport.longitude',␣
 ↪'key', 'location']].sample(10)
```

[ ]:

	src_airport.iata	src_airport.latitude	src_airport.longitude	key	location
39363	LBA	53.865898	-1.660570	LBABUDLS	east
56581	HNL	21.320620	-157.924228	HNLOGGUA	west
21478	MTY	25.778500	-100.107002	MTYLASDL	central
60990	BME	-17.944700	122.232002	BMEPERVA	west
59405	JAN	32.311199	-90.075897	JANDFWUS	central
63790	FLL	26.072599	-80.152702	FLLBNAWN	east
56336	EWR	40.692501	-74.168701	EWRINDUA	east
579	TIV	42.404701	18.723301	TIVDME3R	east
26276	BUD	47.429760	19.261093	BUDBRSFR	east
27462	NYO	58.788601	16.912201	NYOWMIFR	east

[ ]:
```
routes_df.to_parquet('results/geo', partition_cols=['location'])
```

0.1.4 Part d

[ ]:
```
def balance_partitions(keys, num_partitions):
    partition_size = (len(keys) + num_partitions - 1) // num_partitions
    partitions = [keys[i:i + partition_size] for i in range(0, len(keys),␣
 ↪partition_size)]
    partitions = [sorted(partition) for partition in partitions]
    return partitions
```

[ ]:
```
balanced_example = balance_partitions(list(routes_df.key), 30)
print(len(balanced_example))

for partition in balanced_example:
    print(len(partition))
```

```
30
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2226
2217
```