

INFORME DEL TRABAJO FINAL BIGDATA

APARTADO 1:

Con la librería Dask, he leído el archivo y los tipos de datos que forma cada columna. Su clasificación es la siguiente:

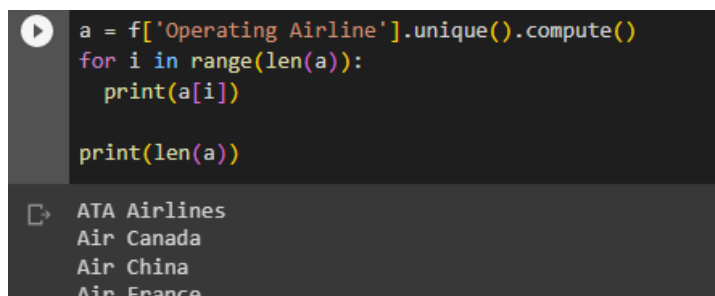
| NOMBRE COLUMNA | TIPO DE DATO |
|-----------------------------|----------------|
| Activity Period | object |
| Operating Airline | object |
| Operating Airline IATA Code | object |
| Published Airline | object |
| Published Airline IATA Code | object |
| GEO Summary | object |
| Geo Region | object |
| Activity Type Code | Object |
| Price Category Code | object |
| Terminal | object |
| Boarding Area | object |
| Passenger Count | Entero (int64) |
| Adjusted Activity Type Code | object |
| Adjusted Passenger Count | Entero (int64) |
| Year | Entero (int64) |
| Month | object |

APARTADO 2

¿Cuántas compañías diferentes aparecen en el fichero?

Tras cargar el csv en un dataframe y analizar los tipos de datos que contenía cada columna, he guardado en una variable los valores únicos de la columna “Operating Airline”, construida por las distintas compañías aéreas. Esta variable es una lista con las 77 compañías que se registran en la columna.

Es decir, en el fichero aparecen 77 compañías aéreas distintas. El código utilizado es:

```
a = f['Operating Airline'].unique().compute()
for i in range(len(a)):
    print(a[i])

print(len(a))

ATA Airlines
Air Canada
Air China
Air France
```

¿Cuántos pasajeros tienen de media los vuelos de cada compañía?

Para responder esta pregunta, voy a agrupar los valores de la columna "Passenger Count" por compañías aéreas. De esta manera, puedo calcular la media de pasajeros por vuelo de cada una. La función groupby devuelve una serie.

El código es el siguiente:

```
gb = f.groupby(by = "Operating Airline") #En esta línea juntamos todos los valores con la misma compañía aérea
gb_medias = gb["Passenger Count"].mean().compute() #Hacemos la media de los valores juntados, solamente en la columna de pasajeros

for i in range(len(gb_medias)):
    print("Compañía: ", gb_medias.index[i], " Media de pasajeros: ", gb_medias[i])
```

En las imágenes inferior se pueden ver las medias de pasajeros y su compañía aérea respectiva:

```
Compañía: ATA Airlines Media de pasajeros: 8744.636363636364
Compañía: Aer Lingus Media de pasajeros: 4407.183673469388
Compañía: Aeromexico Media de pasajeros: 5463.822222222222
Compañía: Air Berlin Media de pasajeros: 2320.75
Compañía: Air Canada Media de pasajeros: 18251.560109289618
Compañía: Air Canada Jazz Media de pasajeros: 294.2142857142857
Compañía: Air China Media de pasajeros: 6618.335907335907
Compañía: Air France Media de pasajeros: 11589.077519379845
Compañía: Air India Limited Media de pasajeros: 2834.5
Compañía: Air New Zealand Media de pasajeros: 7452.339768339768
Compañía: AirTran Airways Media de pasajeros: 10569.238938053097
Compañía: Alaska Airlines Media de pasajeros: 17251.637816245006
Compañía: All Nippon Airways Media de pasajeros: 6385.523255813953
Compañía: Allegiant Air Media de pasajeros: 1516.8125
Compañía: American Airlines Media de pasajeros: 127164.38970588235
Compañía: American Eagle Airlines Media de pasajeros: 4006.5283018867926
Compañía: Ameriflight Media de pasajeros: 5.0
Compañía: Asiana Airlines Media de pasajeros: 5902.961240310077
Compañía: Atlantic Southeast Airlines Media de pasajeros: 2176.909090909091
Compañía: Atlas Air, Inc Media de pasajeros: 34.0
Compañía: BelAir Airlines Media de pasajeros: 415.3636363636364
Compañía: Boeing Company Media de pasajeros: 18.0
Compañía: British Airways Media de pasajeros: 17625.124031007752
Compañía: COPA Airlines, Inc. Media de pasajeros: 3418.0714285714284
Compañía: Cathay Pacific Media de pasajeros: 17121.325581395347
Compañía: China Airlines Media de pasajeros: 9857.51550387597
Compañía: China Eastern Media de pasajeros: 5498.402777777777
Compañía: China Southern Media de pasajeros: 4321.4375
Compañía: Compass Airlines Media de pasajeros: 23358.55681818182
Compañía: Delta Air Lines Media de pasajeros: 68498.49740932643
Compañía: EVA Airways Media de pasajeros: 13116.356589147286
Compañía: Emirates Media de pasajeros: 9070.866666666667
Compañía: Etihad Airways Media de pasajeros: 6476.088235294118
```

Compañía: Emirates Media de pasajeros: 9070.866666666667
 Compañía: Etihad Airways Media de pasajeros: 6476.088235294118
 Compañía: Evergreen International Airlines Media de pasajeros: 2.0
 Compañía: ExpressJet Airlines Media de pasajeros: 5631.84375
 Compañía: Frontier Airlines Media de pasajeros: 17787.676923076924
 Compañía: Hawaiian Airlines Media de pasajeros: 8282.186046511628
 Compañía: Horizon Air Media de pasajeros: 5577.583333333333
 Compañía: Icelandair Media de pasajeros: 2799.7
 Compañía: Independence Air Media de pasajeros: 6391.3
 Compañía: Japan Airlines Media de pasajeros: 6470.332046332046
 Compañía: Jet Airways Media de pasajeros: 4280.3125
 Compañía: JetBlue Airways Media de pasajeros: 35261.13963963964
 Compañía: KLM Royal Dutch Airlines Media de pasajeros: 9221.813953488372
 Compañía: Korean Air Lines Media de pasajeros: 5678.461240310077
 Compañía: LAN Peru Media de pasajeros: 2786.011111111111
 Compañía: Lufthansa German Airlines Media de pasajeros: 19301.96511627907
 Compañía: Mesa Airlines Media de pasajeros: 3710.5811965811968
 Compañía: Mesaba Airlines Media de pasajeros: 2864.727272727275
 Compañía: Mexicana Airlines Media de pasajeros: 7993.806451612903
 Compañía: Miami Air International Media de pasajeros: 107.375
 Compañía: Midwest Airlines Media de pasajeros: 3883.0
 Compañía: Northwest Airlines Media de pasajeros: 26109.25
 Compañía: Pacific Aviation Media de pasajeros: 160.0
 Compañía: Philippine Airlines Media de pasajeros: 10248.635658914729
 Compañía: Qantas Airways Media de pasajeros: 4991.2164179104475
 Compañía: Republic Airlines Media de pasajeros: 2452.5
 Compañía: SAS Airlines Media de pasajeros: 5865.847222222223
 Compañía: Servisair Media de pasajeros: 90.05555555555556
 Compañía: Singapore Airlines Media de pasajeros: 14746.647286821706
 Compañía: SkyWest Airlines Media de pasajeros: 37083.83904465213
 Compañía: Southwest Airlines Media de pasajeros: 81188.15857605178
 Compañía: Spirit Airlines Media de pasajeros: 2921.0416666666665
 Compañía: Sun Country Airlines Media de pasajeros: 3992.652

Compañía: Sun Country Airlines Media de pasajeros: 3992.652
 Compañía: Swiss International Media de pasajeros: 6061.640287769784
 Compañía: Swissport USA Media de pasajeros: 258.6
 Compañía: TACA Media de pasajeros: 5066.197674418605
 Compañía: Turkish Airlines Media de pasajeros: 8162.416666666667
 Compañía: US Airways Media de pasajeros: 55317.81578947369
 Compañía: United Airlines Media de pasajeros: 72732.05829596413
 Compañía: United Airlines - Pre 07/01/2013 Media de pasajeros: 48915.46750232126
 Compañía: Virgin America Media de pasajeros: 74405.35359116022
 Compañía: Virgin Atlantic Media de pasajeros: 9847.10465116279
 Compañía: WestJet Airlines Media de pasajeros: 5338.155339805825
 Compañía: World Airways Media de pasajeros: 261.6666666666667
 Compañía: XL Airways France Media de pasajeros: 2223.1612903225805
 Compañía: Xtra Airways Media de pasajeros: 73.0

Eliminaremos los registros duplicados por el campo “GEO Región”, manteniendo únicamente aquel con mayor número de pasajeros.

En este punto, he agrupado los registros por regiones, y he creado una serie con solamente los pasajeros máximos de la región.

```
gb_region = f.groupby(by = "GEO Region")
gb_region_max = gb_region["Passenger Count"].max().compute()
```

Resultado:

| GEO Region | |
|---------------------|--------|
| Asia | 86398 |
| Australia / Oceania | 12973 |
| Canada | 39798 |
| Central America | 8970 |
| Europe | 48136 |
| Mexico | 29206 |
| Middle East | 14769 |
| South America | 3685 |
| US | 659837 |

Esta serie representa el número de pasajeros máximos que ha habido en los viajes en cada región. Ahora, habría que eliminar del csv original todos los registros que no cumplan este requisito.

Para esto, tras hacer una copia de seguridad de la tabla original, vamos a hacer dos bucles: en el primero, vamos a añadir los registros que queremos borrar en la variable p. Esta variable será un

dataframe, pero lo que necesitamos es el índice de cada registro, es decir, su número de fila. Por lo tanto, añadiré en una lista llamada "tabla_borrar" los valores del índice.

```
tabla_borrar = []
for i in range(len(gb_region_max)):
    #Comparo la serie gb_region_max con el dataframe original, añadiendo en p sólo los que no cumplan la condición:
    p = (f[(f["GEO Region"]== str(gb_region_max.index[i])) & (f["Passenger Count"] < gb_region_max.values[i])])
    tabla_borrar.append(p.index.values) #Añado en tabla_borrar los índices de esos registros.
```

Tabla_borrar es una lista creada por el tipo de dato que devuelve index.values, arrays. En el segundo bucle, transformo cada array en una lista y la añado en la lista "lista_borrar". Así consigo una lista de listas. Todo esto lo hago porque la función drop, que borra registros, necesita como argumento una lista con los índices. Es decir:

```
lista_borrar = []
for i in range(len(tabla_borrar)):
    lista_borrar.append(tabla_borrar[i].tolist()) #Añado en la lista los arrays con índices transformados a listas
    save_f = save_f.drop(lista_borrar[i]) #Cojo cada lista y se la paso a drop para que elimine las filas correspondientes
```

El resultado de la eliminación me devuelve un dataframe con 9 registros: los correspondientes al mayor número de pasajeros por región.

Volcaremos los resultados de los dos puntos anteriores a un CSV

Resumamos: El último punto nos da como resultado un dataframe, y el penúltimo una serie. Transformaremos estos resultados a csv con la función "to_csv". La imagen de la izquierda muestra el csv con los pasajeros máximos por región, mientras que el de la derecha muestra las medias de pasajeros por compañía aérea.

```
f_medias = gb_medias.to_csv(sep = ",", index = True)
print(f_medias)

f_maxpass = save_f.to_csv(sep = ",", index = True)
print(f_maxpass)
```

| Operating Airline | Passenger Count |
|--------------------|--------------------|
| ATA Airlines | 8744.636363636364 |
| Aer Lingus | 4407.183673469388 |
| Aeromexico | 5463.822222222222 |
| Air Berlin | 2320.75 |
| Air Canada | 18251.560109289618 |
| Air Canada Jazz | 294.2142857142857 |
| Air China | 6618.335907335907 |
| Air France | 11589.077519379845 |
| Air India Limited | 2834.5 |
| Air New Zealand | 7452.339768339768 |
| AirTran Airways | 10569.238938053097 |
| Alaska Airlines | 17251.637816245006 |
| All Nippon Airways | 6385.523255813953 |
| Allegiant Air | 1516.8125 |
| American Airlines | 127164.38970588235 |

APARTADO 3

Antes de comenzar el cálculo de medias, y desviaciones, miro qué valores únicos tiene cada columna. Entre las que no son numéricas (tipo de dato objeto), hay algunas que puedo transformar a numéricas y hay otras que no. Estas tienen muchos valores distintos, por lo que el cambio nos haría perder información más que obtenerla.

Por otra parte, hay columnas muy parecidas que contienen información redundante, por lo que voy a eliminar: Published Airline, Published Airline IATA Code, Activity Type Code y Passenger Count.

Las que no transformo son:

Airline IATA Code, Published Airline, Operating Airline IATA Code, Operating Airline, Area de aterrizaje, Terminal.

Para las que transformo, por otra parte, creo una función que me haga dos tareas y así ahorrar trabajo: el remplazo de los datos a números y la media.

```
def remplazar_y_media(dicc, columna):
    f[columna].replace(dicc, inplace = True)
    media = f[columna].mean()
    return media
```

El atributo "dicc" es un diccionario que contiene los valores únicos de la columna y los valores a los que cambio.

```
cambiar_act2 = {'Deplaned': 0, 'Enplaned': 1, 'Thru / Transit * 2': 2}
cambiar_precio = {"Low Fare": 0, "Other": 1}
cambiar_geosum = {'Domestic': 0, 'International': 1}
```

Además, voy a cambiar la columna "Month" con una columna con las estaciones, para ver si hay más pasajeros en una estación que en otra, y como la información de los meses ya la tenemos en la columna "Activity Period" no nos preocupamos por perder datos. Cada estación se representa con un número para poder añadirla a la matriz de correlación.

En esta tabla se puede ver el resultado: la columna de la izquierda representa cada estación (0: invierno, 1: primavera, 2: verano, 3: otoño), y la de la derecha el número de pasajeros totales que ha

```
Season
0 100010007
1 105790280
2 123422073
3 110961720
Name: Adjusted Passenger Count
```

habido. Aunque hay más pasajeros en verano, veremos si estas dos variables son dependientes en la matriz de correlación.

Para la desviación estándar, como todo el trabajo de los datos ya está hecho, simplemente aplico la función. Los resultados son los siguientes:

| COLUMNA | MEDIA | DESVIACIÓN ESTÁNDAR |
|-----------------------------|--------------------|---------------------|
| Adjusted Passenger Count | 29331.917105350836 | 58284.18221866232 |
| Year | 2010.385220230559 | 3.1375890431679667 |
| Season | 1.51469314319984 | 1.1259042055238488 |
| Adjusted Activity Type Code | 0.5901246085160259 | 0.6037477066469839 |
| Price Category Code | 0.8720597054707803 | 0.33403444537349963 |
| GEO Summary | 0.6137136003198508 | 0.4869137658738482 |

De esta tabla podemos sacar cierta información relevante. Por ejemplo, aunque la media de pasajeros es más o menos 29000, vemos por la desviación que los datos están muy dispersos. Es decir, que habrá unos viajes con muy pocos pasajeros y otros con muchos (valores extremos).

En cuanto los años, la desviación es mucho más pequeña, es decir, los datos son cercanos a la media. Podemos concluir, que la mayor parte de viajes se hicieron en 2010.

De la misma lógica, los aviones que pasan por San Francisco no suelen ser de tránsito. De hecho, hay una ligera predominación de embarques que desembarques. Sumándole que los viajes suelen ser internacionales, podemos concluir que hay ligeramente más viajes de Japón a San Francisco que viceversa.

Con la matriz de correlación, voy a ver si las variables son dependientes entre ellas.

| | Activity Period | GEO Summary | Price Category Code | Adjusted Activity Type Code | Adjusted Passenger Count | Year | Season |
|-----------------------------|-----------------|-------------|---------------------|-----------------------------|--------------------------|-----------|-----------|
| Activity Period | 1.000000 | 0.066100 | -0.005754 | -0.052450 | 0.059336 | 0.999940 | -0.090009 |
| GEO Summary | 0.066100 | 1.000000 | 0.411498 | -0.026760 | -0.396856 | 0.066046 | -0.002957 |
| Price Category Code | -0.005754 | 0.411498 | 1.000000 | 0.001004 | -0.064661 | -0.005683 | -0.005633 |
| Adjusted Activity Type Code | -0.052450 | -0.026760 | 0.001004 | 1.000000 | -0.067804 | -0.052364 | -0.000698 |
| Adjusted Passenger Count | 0.059336 | -0.396856 | -0.064661 | -0.067804 | 1.000000 | 0.059096 | 0.019067 |
| Year | 0.999940 | 0.066046 | -0.005683 | -0.052364 | 0.059096 | 1.000000 | -0.096348 |
| Season | -0.090009 | -0.002957 | -0.005633 | -0.000698 | 0.019067 | -0.096348 | 1.000000 |

A simple vista, podemos decir que las variables no se relacionan, ni positivamente ni negativamente. Hay algunas excepciones, la más clara “Year” con “Activity Period”, con una dependencia prácticamente completa (Tiene sentido ya que las dos columnas dicen casi la misma información).

También podemos ver una ligera relación negativa entre el número de pasajeros y GEO Summary. Podemos concluir que, aunque hay más vuelos internacionales, hay pocos pasajeros en estos vuelos. Pasa al contrario con la relación entre el precio y GEO Summary, habiendo una relación positiva. Los viajeros compran tarifas más caras en los vuelos internacionales.

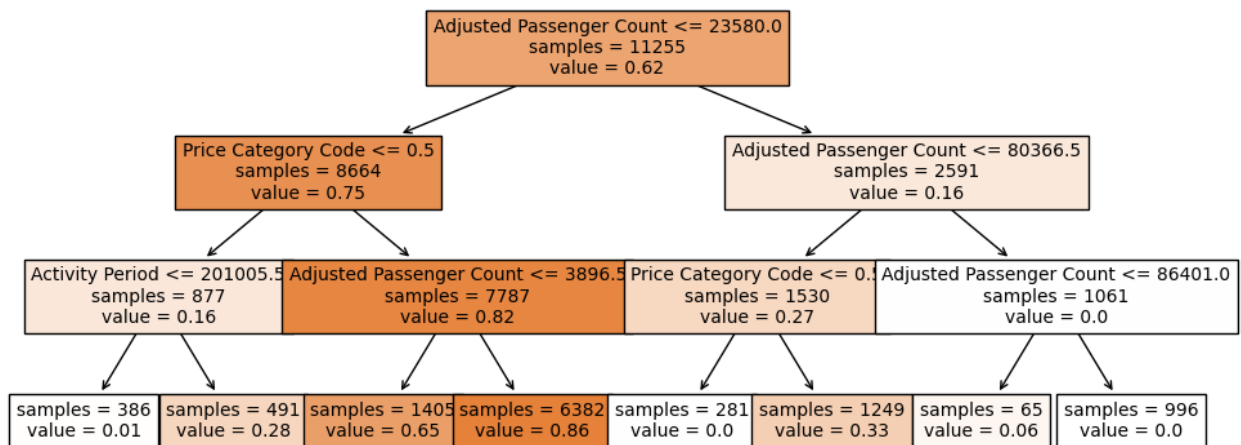
En cuanto a las estaciones, la hipótesis inicial que tenía de que había más pasajeros en verano ha sido totalmente descartada, ya que se ve que hay correlación nula con todas las demás variables.

Según mi hipótesis, las variables independientes son "Adjusted Passenger Count" y "Price Category Code", mientras que la dependiente es "GEO Summary". Como hay dos variables independientes, voy a hacer un árbol de regresión, o regresión factorial, para ver que peso tiene cada una de ellas.

ÁRBOL DE REGRESIÓN

Voy a utilizar la librería sklearn. Primero, borro las columnas que no quiero meter en el árbol, dejando solamente las tres mencionadas anteriormente. Divido el dataset en train y test (algunos datos los usará para entrenar y el resto para comprobar el resultado del entrenamiento). Finalmente, creo el modelo de regresión y lo muestro en una gráfica. Para representarlo he usado la librería matplotlib

El resultado del modelo es el siguiente:



DEFENSA DEL PROYECTO:

Si tenemos un conjunto de datos muy grande que no cabe en una sola máquina, para distribuir los datos en distintas máquinas primero agruparía los datos haciendo un mapeado, para trasladar a las distintas máquinas información relevante con la que poder trabajar.