Booking Cancellation

October 7, 2022

[ ]:

Table 1
Variables description.

Variable	Type	Description	Source/Engineering
ADR	Numeric	Average Daily Rate as defined by [5]	BO, BL and TR / Calculated by dividing the sum of all lodging transactions by the total number of staying nights
Adults	Integer	Number of adults	BO and BL
Agent	Categorical	ID of the travel agency that made the booking[a]	BO and BL
ArrivalDateDayOfMonth	Integer	Day of the month of the arrival date	BO and BL
ArrivalDateMonth	Categorical	Month of arrival date with 12 categories: "January" to "December"	BO and BL
ArrivalDateWeekNumber	Integer	Week number of the arrival date	BO and BL
ArrivalDateYear	Integer	Year of arrival date	BO and BL
AssignedRoomType	Categorical	Code for the type of room assigned to the booking. Sometimes the assigned room type differs from the reserved room type due to hotel operation reasons (e.g. overbooking) or by customer request. Code is presented instead of designation for anonymity reasons	BO and BL
Babies	Integer	Number of babies	BO and BL
BookingChanges	Integer	Number of changes/amendments made to the booking from the moment the booking was entered on the PMS until the moment of check-in or cancellation	BO and BL/Calculated by adding the number of unique iterations that change some of the booking attributes, namely: persons, arrival date, nights, reserved room type or meal
Children	Integer	Number of children	BO and BL/Sum of both payable and non-payable children
Company	Categorical	ID of the company/entity that made the booking or responsible for paying the booking. ID is presented instead of designation for anonymity reasons	BO and BL.
Country	Categorical	Country of origin. Categories are represented in the ISO 3155–3:2013 format [6]	BO, BL and NT
CustomerType	Categorical	Type of booking, assuming one of four categories: Contract - when the booking has an allotment or other type of contract associated to it; Group – when the booking is associated to a group; Transient – when the booking is not part of a group or contract, and is not associated to other transient booking; Transient-party – when the booking is transient, but is associated to at least other transient booking	BO and BL
DaysInWaitingList	Integer	Number of days the booking was in the waiting list before it was confirmed to the customer	BO/Calculated by subtracting the date the booking was confirmed to the customer from the date the booking entered on the PMS
DepositType	Categorical	Indication on if the customer made a deposit to guarantee the booking. This variable can assume three categories: No Deposit – no deposit was made; Non Refund – a deposit was made in the value of the total stay cost; Refundable – a deposit was made with a value under the total cost of stay.	BO and TR/Value calculated based on the payments identified for the booking in the transaction (TR) table before the booking's arrival or cancellation date. In case no payments were found the value is "No Deposit". If the payment was equal or exceeded the total cost of stay, the value is set as "Non Refund". Otherwise the value is set as "Refundable"

Table 1 (*continued*)

Variable	Type	Description	Source/Engineering
DistributionChannel	Categorical	Booking distribution channel. The term "TA" means "Travel Agents" and "TO" means "Tour Operators"	BO, BL and DC
IsCanceled	Categorical	Value indicating if the booking was canceled (1) or not (0)	BO
IsRepeatedGuest	Categorical	Value indicating if the booking name was from a repeated guest (1) or not (0)	BO, BL and C/ Variable created by verifying if a profile was associated with the booking customer. If so, and if the customer profile creation date was prior to the creation date for the booking on the PMS database it was assumed the booking was from a repeated guest
LeadTime	Integer	Number of days that elapsed between the entering date of the booking into the PMS and the arrival date	BO and BL/ Subtraction of the entering date from the arrival date
MarketSegment	Categorical	Market segment designation. In categories, the term "TA" means "Travel Agents" and "TO" means "Tour Operators"	BO, BL and MS
Meal	Categorical	Type of meal booked. Categories are presented in standard hospitality meal packages: Undefined/SC – no meal package; BB – Bed & Breakfast; HB – Half board (breakfast and one other meal – usually dinner); FB – Full board (breakfast, lunch and dinner)	BO, BL and ML
PreviousBookingsNotCanceled	Integer	Number of previous bookings not cancelled by the customer prior to the current booking	BO and BL / In case there was no customer profile associated with the booking, the value is set to 0. Otherwise, the value is the number of bookings with the same customer profile created before the current booking and not canceled.
PreviousCancellations	Integer	Number of previous bookings that were cancelled by the customer prior to the current booking	BO and BL/ In case there was no customer profile associated with the booking, the value is set to 0. Otherwise, the value is the number of bookings with the same customer profile created before the current booking and canceled.
RequiredCardParkingSpaces	Integer	Number of car parking spaces required by the customer	BO and BL
ReservationStatus	Categorical	Reservation last status, assuming one of three categories: Canceled – booking was canceled by the customer; Check-Out – customer has checked in but already departed; No-Show – customer did not check-in and did inform the hotel of the reason why	BO

Table 1 (*continued*)

Variable	Type	Description	Source/Engineering
ReservationStatusDate	Date	Date at which the last status was set. This variable can be used in conjunction with the *ReservationStatus* to understand when was the booking canceled or when did the customer checked-out of the hotel	BO
ReservedRoomType	Categorical	Code of room type reserved. Code is presented instead of designation for anonymity reasons	BO and BL
StaysInWeekendNights	Integer	Number of weekend nights (Saturday or Sunday) the guest stayed or booked to stay at the hotel	BO and BL/ Calculated by counting the number of weekend nights from the total number of nights
StaysInWeekNights	Integer	Number of week nights (Monday to Friday) the guest stayed or booked to stay at the hotel	BO and BL/Calculated by counting the number of week nights from the total number of nights
TotalOfSpecialRequests	Integer	Number of special requests made by the customer (e.g. twin bed or high floor)	BO and BL/Sum of all special requests

[a] ID is presented instead of designation for anonymity reasons.

[ ]:

0.1 Guidelines based on this analysis:

https://www.sciencedirect.com/science/article/pii/S2352340918315191

https://www.kaggle.com/datasets/jessemostipak/hotel-booking-demand

1 Questions:

- 1. What's the cancellation rate?
- 2. What's the highest day/month for the cancellation rate?
- 3. Is cancellation rate related to single/married type?
- 4. What's the proportion of the cancellation rates?

[ ]:

2 1. Data Preprocessing

[1]:
```
# Import the required libraries

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from statistics import mode
```

[2]:
```
# Load the raw data frame "df"

df = pd.read_csv("hotel_bookings.csv")
df
```

[2]:

```
                hotel  is_canceled  lead_time  arrival_date_year  \
0        Resort Hotel            0        342               2015
1        Resort Hotel            0        737               2015
2        Resort Hotel            0          7               2015
3        Resort Hotel            0         13               2015
4        Resort Hotel            0         14               2015
…                …           …         …                  …
119385     City Hotel            0         23               2017
119386     City Hotel            0        102               2017
119387     City Hotel            0         34               2017
119388     City Hotel            0        109               2017
119389     City Hotel            0        205               2017

       arrival_date_month  arrival_date_week_number  \
0                    July                        27
1                    July                        27
2                    July                        27
3                    July                        27
4                    July                        27
…                     …                         …
119385             August                        35
119386             August                        35
119387             August                        35
119388             August                        35
119389             August                        35

        arrival_date_day_of_month  stays_in_weekend_nights  \
0                               1                       0
1                               1                       0
2                               1                       0
3                               1                       0
4                               1                       0
…                             …                       …
119385                         30                       2
119386                         31                       2
119387                         31                       2
119388                         31                       2
119389                         29                       2

        stays_in_week_nights  adults  …  deposit_type  agent company  \
0                          0       2  …    No Deposit    NaN     NaN
1                          0       2  …    No Deposit    NaN     NaN
```

```
2                          1       1  …    No Deposit    NaN     NaN
3                          1       1  …    No Deposit  304.0     NaN
4                          2       2  …    No Deposit  240.0     NaN
…                        …     …  …            …    …     …
119385                     5       2  …    No Deposit  394.0     NaN
119386                     5       3  …    No Deposit    9.0     NaN
119387                     5       2  …    No Deposit    9.0     NaN
119388                     5       2  …    No Deposit   89.0     NaN
119389                     7       2  …    No Deposit    9.0     NaN

       days_in_waiting_list customer_type     adr  \
0                         0     Transient    0.00
1                         0     Transient    0.00
2                         0     Transient   75.00
3                         0     Transient   75.00
4                         0     Transient   98.00
…                       …           …     …
119385                    0     Transient   96.14
119386                    0     Transient  225.43
119387                    0     Transient  157.71
119388                    0     Transient  104.40
119389                    0     Transient  151.20

        required_car_parking_spaces  total_of_special_requests  \
0                                 0                          0
1                                 0                          0
2                                 0                          0
3                                 0                          0
4                                 0                          1
…                               …                        …
119385                            0                          0
119386                            0                          2
119387                            0                          4
119388                            0                          0
119389                            0                          2

        reservation_status reservation_status_date
0                Check-Out              2015-07-01
1                Check-Out              2015-07-01
2                Check-Out              2015-07-02
3                Check-Out              2015-07-02
4                Check-Out              2015-07-03
…                      …                       …
119385           Check-Out              2017-09-06
119386           Check-Out              2017-09-07
119387           Check-Out              2017-09-07
119388           Check-Out              2017-09-07
```

```
119389           Check-Out              2017-09-07

[119390 rows x 32 columns]
```

[3]:
```
# Checking the data D-types and overall insights

df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119390 entries, 0 to 119389
Data columns (total 32 columns):
 #   Column                          Non-Null Count   Dtype
---  ------                          --------------   -----
 0   hotel                           119390 non-null  object
 1   is_canceled                     119390 non-null  int64
 2   lead_time                       119390 non-null  int64
 3   arrival_date_year               119390 non-null  int64
 4   arrival_date_month              119390 non-null  object
 5   arrival_date_week_number        119390 non-null  int64
 6   arrival_date_day_of_month       119390 non-null  int64
 7   stays_in_weekend_nights         119390 non-null  int64
 8   stays_in_week_nights            119390 non-null  int64
 9   adults                          119390 non-null  int64
 10  children                        119386 non-null  float64
 11  babies                          119390 non-null  int64
 12  meal                            119390 non-null  object
 13  country                         118902 non-null  object
 14  market_segment                  119390 non-null  object
 15  distribution_channel            119390 non-null  object
 16  is_repeated_guest               119390 non-null  int64
 17  previous_cancellations          119390 non-null  int64
 18  previous_bookings_not_canceled  119390 non-null  int64
 19  reserved_room_type              119390 non-null  object
 20  assigned_room_type              119390 non-null  object
 21  booking_changes                 119390 non-null  int64
 22  deposit_type                    119390 non-null  object
 23  agent                           103050 non-null  float64
 24  company                         6797 non-null    float64
 25  days_in_waiting_list            119390 non-null  int64
 26  customer_type                   119390 non-null  object
 27  adr                             119390 non-null  float64
 28  required_car_parking_spaces     119390 non-null  int64
 29  total_of_special_requests       119390 non-null  int64
 30  reservation_status              119390 non-null  object
 31  reservation_status_date         119390 non-null  object
dtypes: float64(4), int64(16), object(12)
memory usage: 29.1+ MB
```

[4]:
```
# Checking the statistical proporation of each column in df

df.describe()
```

[4]:

	is_canceled	lead_time	arrival_date_year
count	119390.000000	119390.000000	119390.000000
mean	0.370416	104.011416	2016.156554
std	0.482918	106.863097	0.707476
min	0.000000	0.000000	2015.000000
25%	0.000000	18.000000	2016.000000
50%	0.000000	69.000000	2016.000000
75%	1.000000	160.000000	2017.000000
max	1.000000	737.000000	2017.000000

	arrival_date_week_number	arrival_date_day_of_month
count	119390.000000	119390.000000
mean	27.165173	15.798241
std	13.605138	8.780829
min	1.000000	1.000000
25%	16.000000	8.000000
50%	28.000000	16.000000
75%	38.000000	23.000000
max	53.000000	31.000000

	stays_in_weekend_nights	stays_in_week_nights	adults
count	119390.000000	119390.000000	119390.000000
mean	0.927599	2.500302	1.856403
std	0.998613	1.908286	0.579261
min	0.000000	0.000000	0.000000
25%	0.000000	1.000000	2.000000
50%	1.000000	2.000000	2.000000
75%	2.000000	3.000000	2.000000
max	19.000000	50.000000	55.000000

	children	babies	is_repeated_guest
count	119386.000000	119390.000000	119390.000000
mean	0.103890	0.007949	0.031912
std	0.398561	0.097436	0.175767
min	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000
50%	0.000000	0.000000	0.000000
75%	0.000000	0.000000	0.000000
max	10.000000	10.000000	1.000000

	previous_cancellations	previous_bookings_not_canceled
count	119390.000000	119390.000000
mean	0.087118	0.137097

```
std                  0.844336                       1.497437
min                  0.000000                       0.000000
25%                  0.000000                       0.000000
50%                  0.000000                       0.000000
75%                  0.000000                       0.000000
max                 26.000000                      72.000000

       booking_changes          agent      company  days_in_waiting_list  \
count    119390.000000  103050.000000  6797.000000         119390.000000
mean          0.221124      86.693382   189.266735              2.321149
std           0.652306     110.774548   131.655015             17.594721
min           0.000000       1.000000     6.000000              0.000000
25%           0.000000       9.000000    62.000000              0.000000
50%           0.000000      14.000000   179.000000              0.000000
75%           0.000000     229.000000   270.000000              0.000000
max          21.000000     535.000000   543.000000            391.000000

                adr  required_car_parking_spaces  total_of_special_requests
count  119390.000000                119390.000000              119390.000000
mean      101.831122                     0.062518                   0.571363
std        50.535790                     0.245291                   0.792798
min        -6.380000                     0.000000                   0.000000
25%        69.290000                     0.000000                   0.000000
50%        94.575000                     0.000000                   0.000000
75%       126.000000                     0.000000                   1.000000
max      5400.000000                     8.000000                   5.000000
```

3 2. Data Wrangling

[5]:
```
# Missing value counts in the Data Frame

missing_values = df.isnull().sum()/len(df)
missing_values = missing_values[missing_values > 0]
missing_values.sort_values(inplace=True)
missing_values
```

[5]:
```
children    0.000034
country     0.004087
agent       0.136862
company     0.943069
dtype: float64
```

[6]:
```
# Drop the agent & company as both have the highest missing values > 90% of the␣
 ↪total values in the column
# Imputation technique or KNN can't help in predicting those values
```

```
df.drop(["agent", "company"], axis = 1, inplace = True)
```

[7]:
```
# Fille the null values with mode; the most repeative values in each column

df["country"] = df["country"].fillna(df["country"].mode()[0])
df["children"] = df["children"].fillna(df["children"].mode()[0])
```

[8]:
```
df
```

[8]:

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month	stays_in_weekend_nights
0	Resort Hotel	0	342	2015	July	27	1	0
1	Resort Hotel	0	737	2015	July	27	1	0
2	Resort Hotel	0	7	2015	July	27	1	0
3	Resort Hotel	0	13	2015	July	27	1	0
4	Resort Hotel	0	14	2015	July	27	1	0
…	…	…	…	…	…	…	…	…
119385	City Hotel	0	23	2017	August	35	30	2
119386	City Hotel	0	102	2017	August	35	31	2
119387	City Hotel	0	34	2017	August	35	31	2
119388	City Hotel	0	109	2017	August	35	31	2
119389	City Hotel	0	205	2017	August	35	29	2

```
        stays_in_week_nights  adults  …  assigned_room_type  \
0                          0       2  …                   C
1                          0       2  …                   C
2                          1       1  …                   C
3                          1       1  …                   A
4                          2       2  …                   A
…                        …     …  …                 …
119385                     5       2  …                   A
119386                     5       3  …                   E
119387                     5       2  …                   D
119388                     5       2  …                   A
119389                     7       2  …                   A

        booking_changes deposit_type days_in_waiting_list customer_type  \
0                     3   No Deposit                    0     Transient
1                     4   No Deposit                    0     Transient
2                     0   No Deposit                    0     Transient
3                     0   No Deposit                    0     Transient
4                     0   No Deposit                    0     Transient
…                   …           …                    …           …
119385                0   No Deposit                    0     Transient
119386                0   No Deposit                    0     Transient
119387                0   No Deposit                    0     Transient
119388                0   No Deposit                    0     Transient
119389                0   No Deposit                    0     Transient

           adr  required_car_parking_spaces  total_of_special_requests  \
0         0.00                            0                          0
1         0.00                            0                          0
2        75.00                            0                          0
3        75.00                            0                          0
4        98.00                            0                          1
…         …                             …                        …
119385   96.14                            0                          0
119386  225.43                            0                          2
119387  157.71                            0                          4
119388  104.40                            0                          0
119389  151.20                            0                          2

        reservation_status reservation_status_date
0                Check-Out              2015-07-01
1                Check-Out              2015-07-01
2                Check-Out              2015-07-02
3                Check-Out              2015-07-02
4                Check-Out              2015-07-03
…                      …                       …
```

```
119385          Check-Out              2017-09-06
119386          Check-Out              2017-09-07
119387          Check-Out              2017-09-07
119388          Check-Out              2017-09-07
119389          Check-Out              2017-09-07

[119390 rows x 30 columns]
```

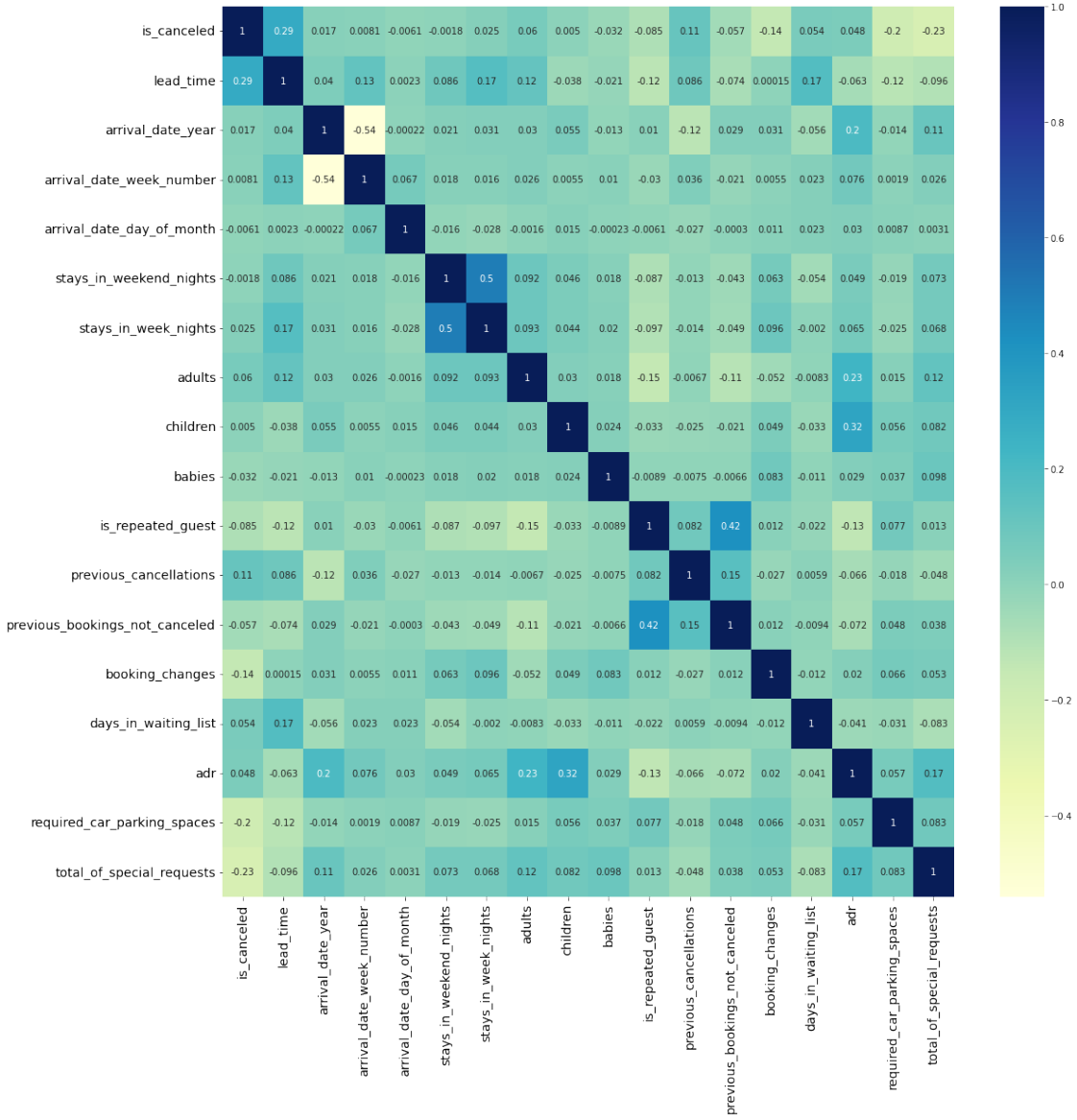
4 3. Explatory Data Analysis

[10]:
```
df.hist(figsize = (18,18))
```

[10]:
```
array([[<AxesSubplot:title={'center':'is_canceled'}>,
        <AxesSubplot:title={'center':'lead_time'}>,
        <AxesSubplot:title={'center':'arrival_date_year'}>,
        <AxesSubplot:title={'center':'arrival_date_week_number'}>],
       [<AxesSubplot:title={'center':'arrival_date_day_of_month'}>,
        <AxesSubplot:title={'center':'stays_in_weekend_nights'}>,
        <AxesSubplot:title={'center':'stays_in_week_nights'}>,
        <AxesSubplot:title={'center':'adults'}>],
       [<AxesSubplot:title={'center':'children'}>,
        <AxesSubplot:title={'center':'babies'}>,
        <AxesSubplot:title={'center':'is_repeated_guest'}>,
        <AxesSubplot:title={'center':'previous_cancellations'}>],
       [<AxesSubplot:title={'center':'previous_bookings_not_canceled'}>,
        <AxesSubplot:title={'center':'booking_changes'}>,
        <AxesSubplot:title={'center':'days_in_waiting_list'}>,
        <AxesSubplot:title={'center':'adr'}>],
       [<AxesSubplot:title={'center':'required_car_parking_spaces'}>,
        <AxesSubplot:title={'center':'total_of_special_requests'}>,
        <AxesSubplot:>, <AxesSubplot:>]], dtype=object)
```

[201]:
```
plt.figure(figsize = (18,18))

sns.heatmap(data = df.corr(), cmap="YlGnBu", annot=True)
plt.xticks(fontsize = 14)
plt.yticks(fontsize = 14)

plt.show()
```

```
/tmp/ipykernel_27581/4044032107.py:3: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  sns.heatmap(data = df.corr(), cmap="YlGnBu", annot=True)
```



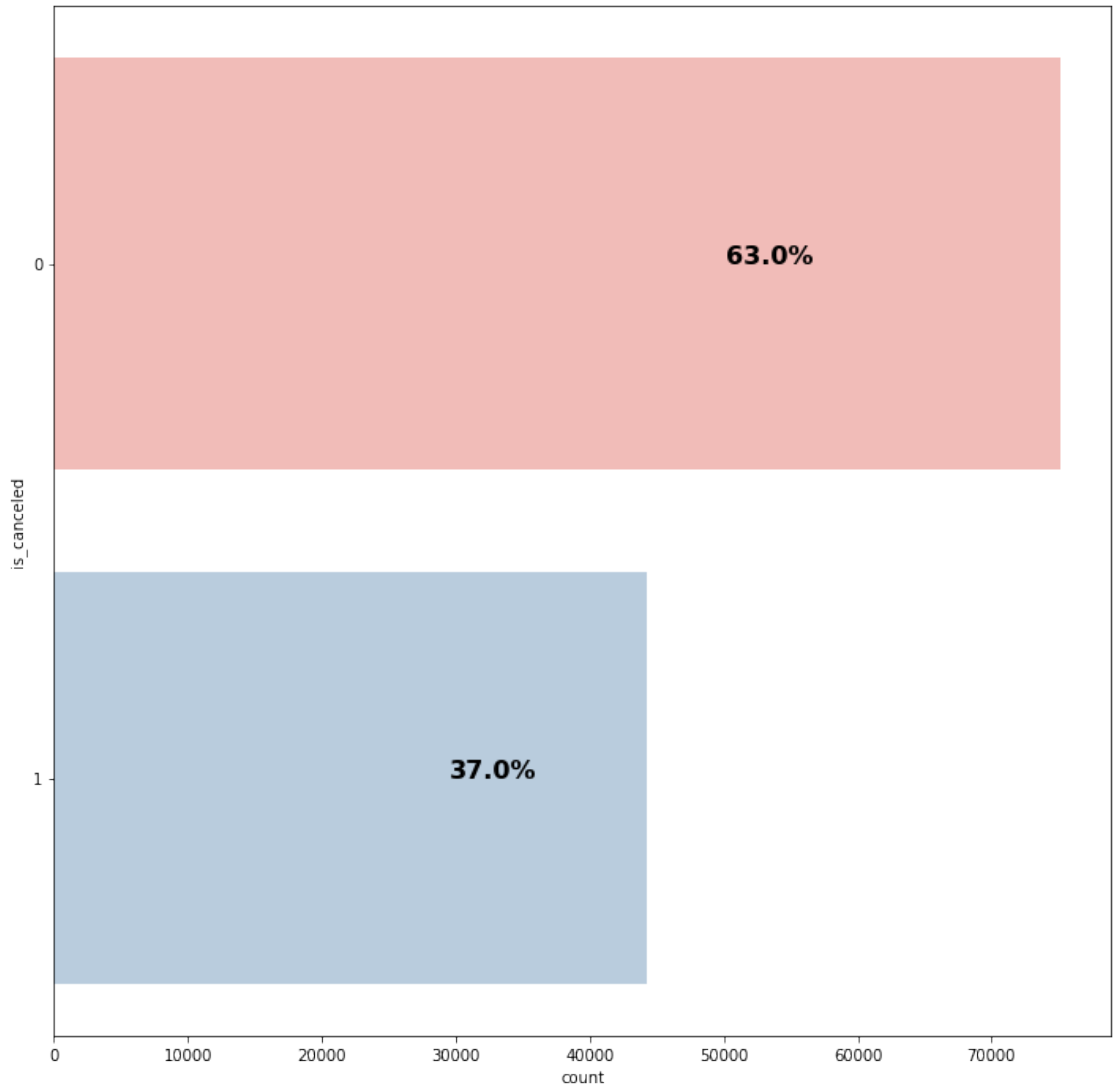
4.1 1. What's the cancellation rate?

[209]:
```
# Calculating the ratios

print("The ratios of non-cancelled reservations is:",␣
 ↪(round((df["is_canceled"][df["is_canceled"] == 0]).count()/df["is_canceled"].
 ↪count(), 2))*100, "%")
print("The ratios of cancelled reservations is:",␣
 ↪(round((df["is_canceled"][df["is_canceled"] == 1]).count()/df["is_canceled"].
 ↪count(), 2))*100, "%")
```

```
The ratios of non-cancelled reservations is: 63.0 %
```

```
The ratios of cancelled reservations is: 37.0 %
```

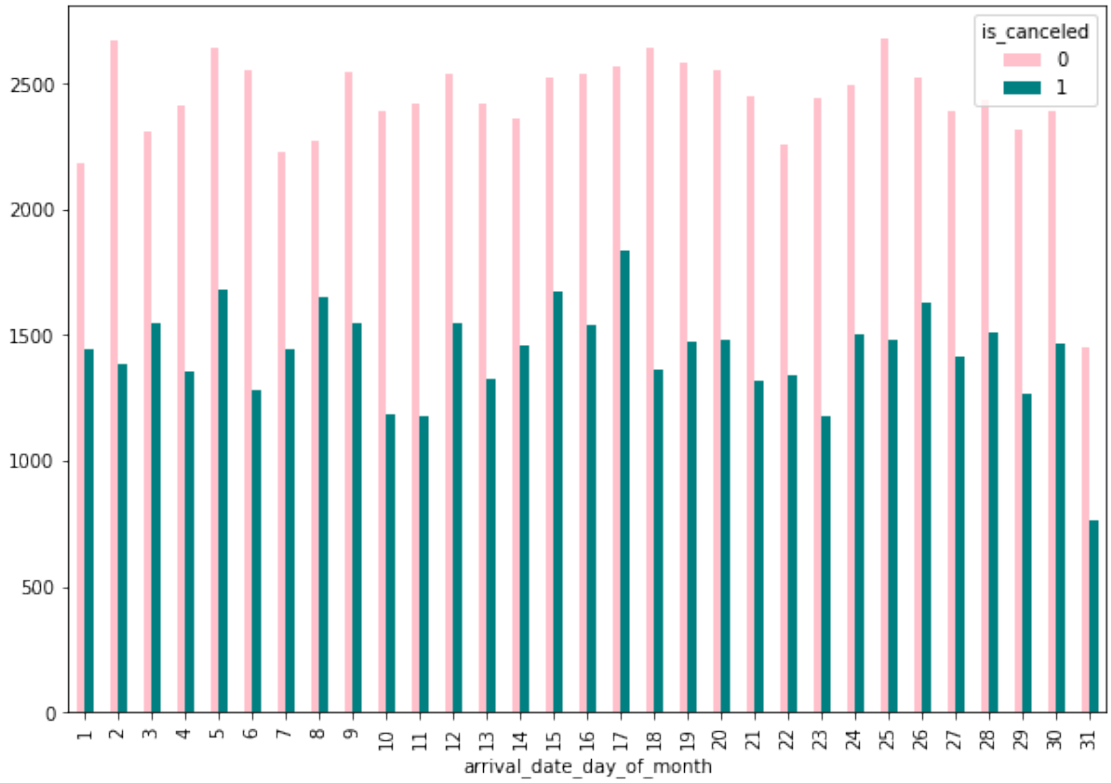
[347]:
```
######### Essential plot for Tableau

plt.figure(figsize=(12,12))
ax = sns.countplot(y="is_canceled", data=df, palette="Pastel1")
total = len(df['is_canceled'])

for v in ax.patches:
        percentage = '{:.1f}%'.format(100 * v.get_width()/total)
        x = v.get_x() + v.get_width() / 1.5
        y = v.get_y() + v.get_height()/2
        ax.annotate(percentage, (x, y), fontsize = 16, weight='bold')


plt.show()


# https://matplotlib.org/stable/gallery/color/named_colors.html
# http://rstudio-pubs-static.s3.amazonaws.com/
 ↪5312_98fc1aba2d5740dd849a5ab797cc2c8d.html
```



4.2 2. What's the highest day/month for the cancellation rate?

[204]:
```
df.groupby("arrival_date_day_of_month")["is_canceled"].value_counts().unstack().
↪plot.bar(figsize=(10,7),

↪          color =('pink', 'teal'))
```

[204]: <AxesSubplot:xlabel='arrival_date_day_of_month'>



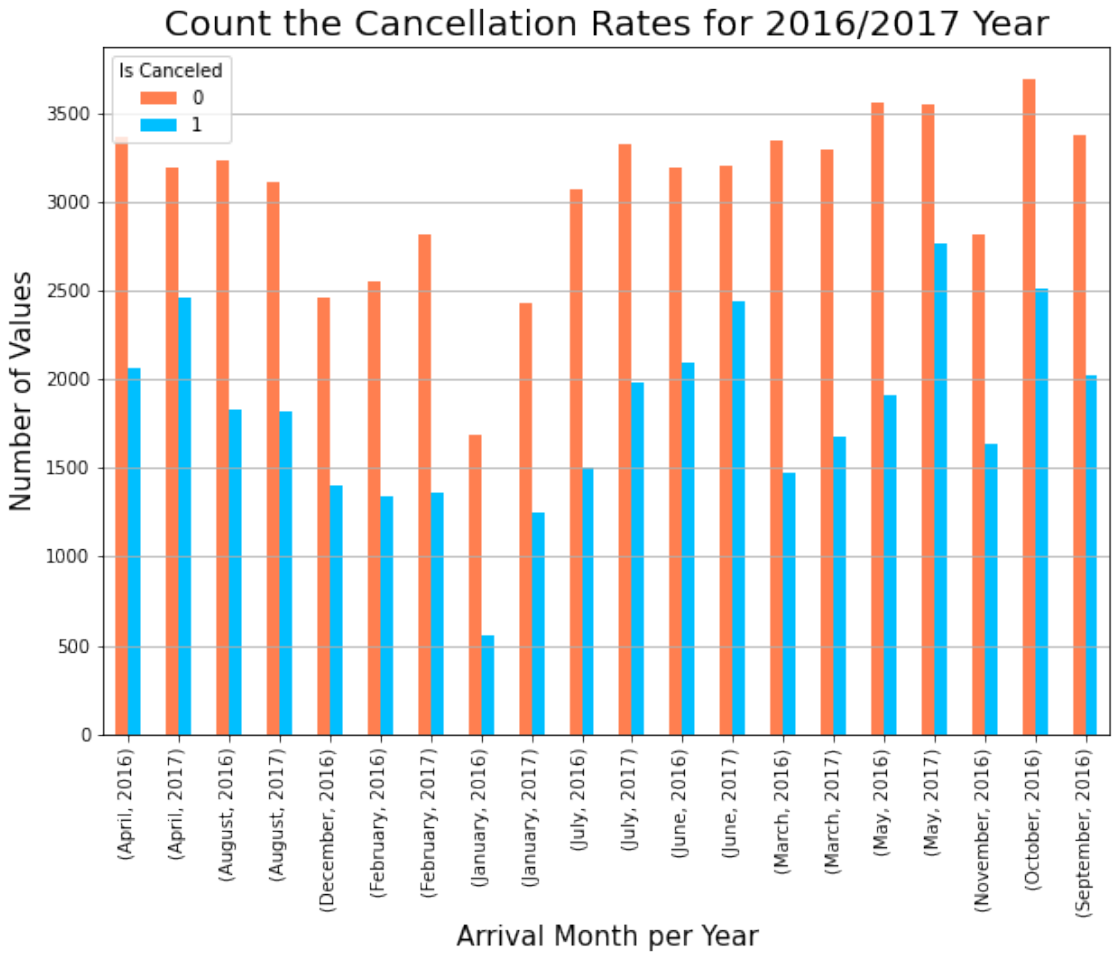
[134]:
```
# Highlight the unique values

df["arrival_date_year"].value_counts()
```

[134]:
```
2016    56707
2017    40687
2015    21996
Name: arrival_date_year, dtype: int64
```

[149]:
```
# Select the observations for years 2016 & 2017

data = df[(df["arrival_date_year"] == 2016) | (df["arrival_date_year"] == 2017)]

# https://stackoverflow.com/questions/67332003/
 ↪pandas-select-rows-from-a-dataframe-based-on-column-values
```

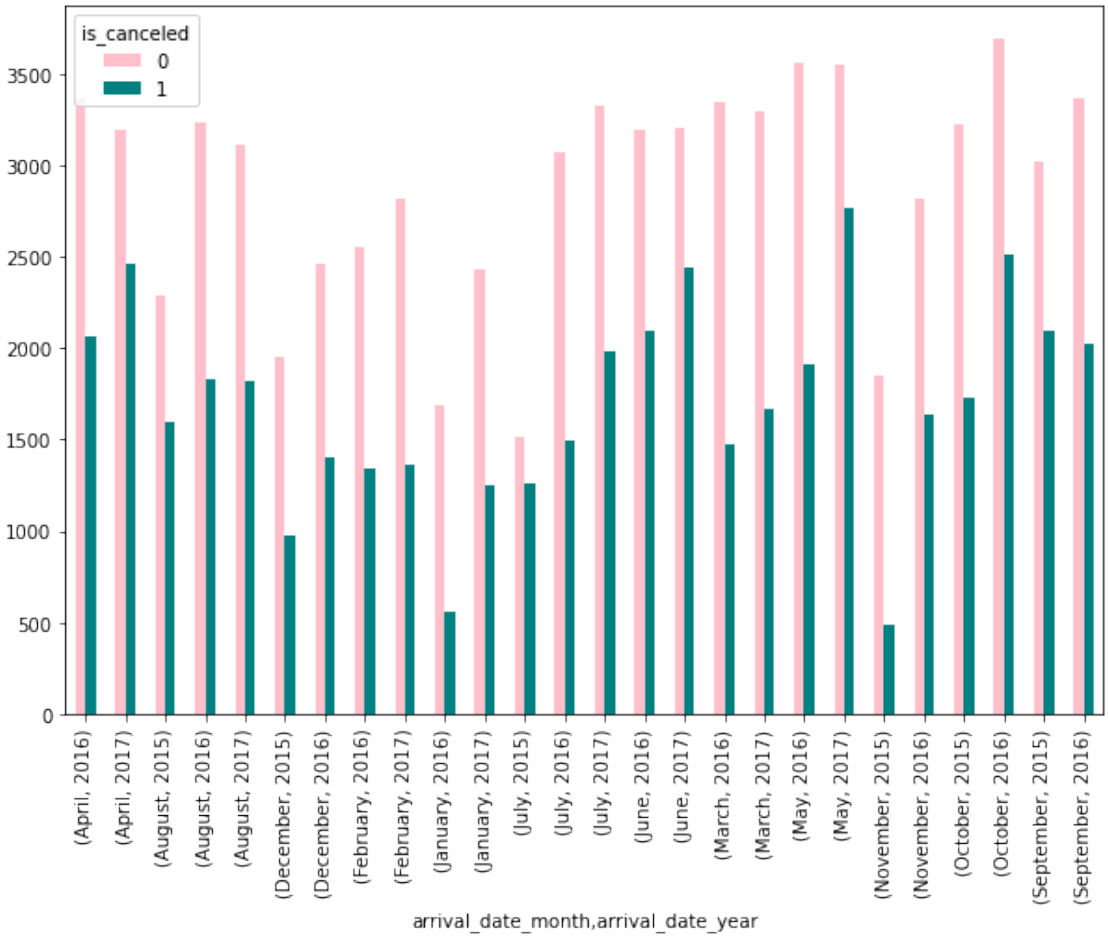
[342]:
```
# Plot the highest month for cancellation rates.

data.groupby(["arrival_date_month", "arrival_date_year"])["is_canceled"].
 ↪value_counts().unstack().plot.bar(
    figsize=(10,7), color =('coral', 'deepskyblue'))
```

```
plt.legend(title = "Is Canceled", loc ='upper left')
plt.xlabel("Arrival Month per Year", fontsize=15)
plt.ylabel("Number of Values", fontsize=15)
plt.title("Count the Cancellation Rates for 2016/2017 Year", fontsize=20)
plt.grid(axis="y")
plt.show()
```



[122]:
```
# Plot all the years/months data "for illustration"

df.groupby(["arrival_date_month", "arrival_date_year"])["is_canceled"].
 ↪value_counts().unstack().plot.bar(figsize=(10,7),

 ↪          color =('pink', 'teal'))
```

[122]: <AxesSubplot:xlabel='arrival_date_month,arrival_date_year'>

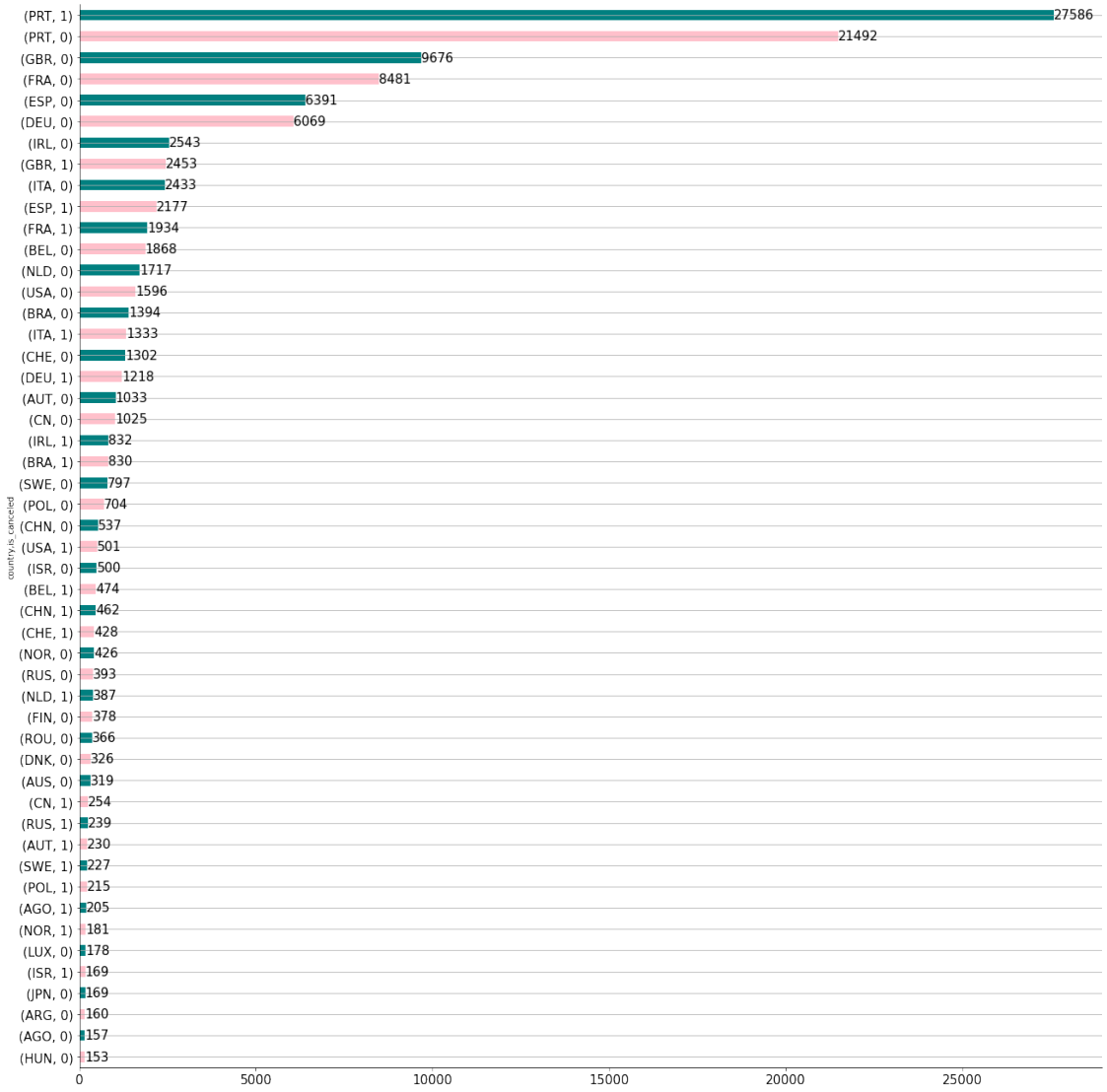


[285]:
```
df.groupby("country")["is_canceled"].count().nlargest(35).sort_values(ascending
 ↪= False)
```

[285]:
```
country
PRT    49078
GBR    12129
FRA    10415
ESP     8568
DEU     7287
ITA     3766
IRL     3375
BEL     2342
BRA     2224
NLD     2104
USA     2097
CHE     1730
CN      1279
AUT     1263
```

```
SWE     1024
CHN      999
POL      919
ISR      669
RUS      632
NOR      607
ROU      500
FIN      447
DNK      435
AUS      426
AGO      362
LUX      287
MAR      259
TUR      248
HUN      230
ARG      214
JPN      197
CZE      171
IND      152
KOR      133
GRC      128
Name: is_canceled, dtype: int64
```

[339]:
```
#

l = df.groupby("country")["is_canceled"].value_counts().nlargest(50).
 ↪sort_values(ascending = True).plot.barh(figsize=(18,18), color =('pink',␣
 ↪'teal'))

l.bar_label(l.containers[0], fontsize = 15, label_type='edge')
plt.tight_layout()

l.spines['top'].set_visible(False)
l.spines['right'].set_visible(False)

plt.xticks("The Total Number", fontsize = 15)
plt.yticks("The Cancellation Rate per Country", fontsize = 15)

l.grid(axis="y")
```



4.3 3. Is cancellation rate related to single/married type?

[371]:
```
dd = df.groupby("is_canceled")[["adults", "children", "babies"]].value_counts().
↪to_frame(name = "Total_Counts").reset_index()
```

[378]:
```
data = dd[(dd["is_canceled"] == 1) & ((dd["children"] != 0) | (dd["babies"] !=␣
↪0))]
data

# Since it's impossible to see 0 adults and 2 children go in a trip or booking␣
↪a hotel, this considered as
# system error and I will replace it with the mode "the most repeated value in␣
↪the column".
```

[378]:

	is_canceled	adults	children	babies	Total_Counts
33	1	2	2.0	0	1392
34	1	2	1.0	0	1269
35	1	3	1.0	0	211
36	1	2	0.0	1	127
37	1	0	2.0	0	80
38	1	1	1.0	0	65
39	1	1	2.0	0	47
41	1	2	1.0	1	21
43	1	2	3.0	0	12
44	1	2	2.0	1	10
45	1	3	2.0	0	9
47	1	1	0.0	1	3
48	1	0	3.0	0	3
50	1	1	3.0	0	2
51	1	2	0.0	2	2
52	1	1	2.0	1	2
59	1	0	2.0	1	1
60	1	4	1.0	0	1
61	1	3	0.0	1	1
62	1	2	10.0	0	1

[380]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 20 entries, 33 to 62
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   is_canceled   20 non-null     int64
 1   adults        20 non-null     int64
 2   children      20 non-null     float64
 3   babies        20 non-null     int64
 4   Total_Counts  20 non-null     int64
dtypes: float64(1), int64(4)
memory usage: 960.0 bytes
```

[401]:
```
from statistics import mode

data["adults"] = round(data["adults"].replace(0, mode(data["adults"])),0)
data
```

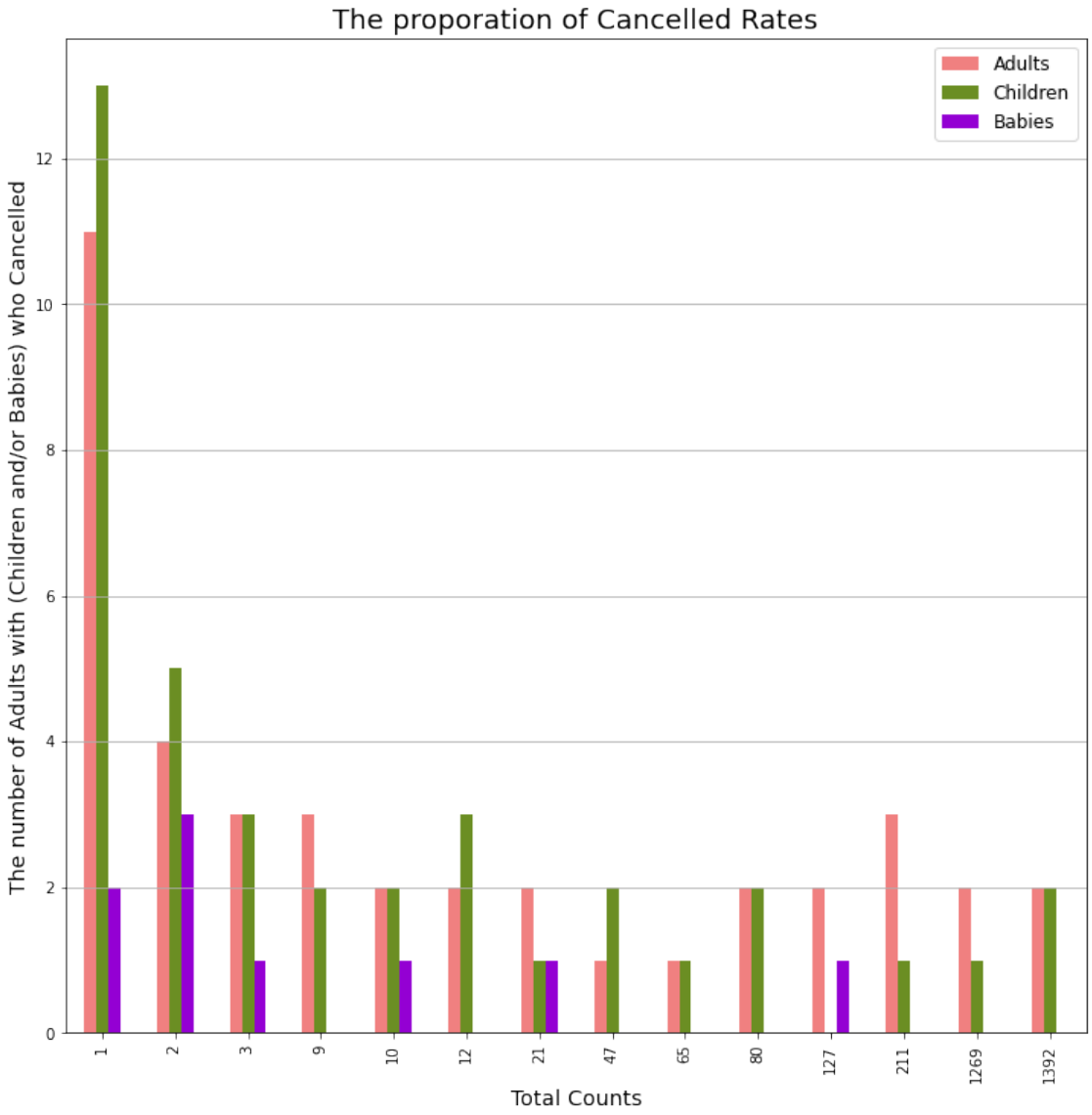
```
/tmp/ipykernel_27581/13432613.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
data["adults"] = round(data["adults"].replace(0, mode(data["adults"])),0)
```

[401]:

	is_canceled	adults	children	babies	Total_Counts
33	1	2.0	2.0	0	1392
34	1	2.0	1.0	0	1269
35	1	3.0	1.0	0	211
36	1	2.0	0.0	1	127
37	1	2.0	2.0	0	80
38	1	1.0	1.0	0	65
39	1	1.0	2.0	0	47
41	1	2.0	1.0	1	21
43	1	2.0	3.0	0	12
44	1	2.0	2.0	1	10
45	1	3.0	2.0	0	9
47	1	1.0	0.0	1	3
48	1	2.0	3.0	0	3
50	1	1.0	3.0	0	2
51	1	2.0	0.0	2	2
52	1	1.0	2.0	1	2
59	1	2.0	2.0	1	1
60	1	4.0	1.0	0	1
61	1	3.0	0.0	1	1
62	1	2.0	10.0	0	1

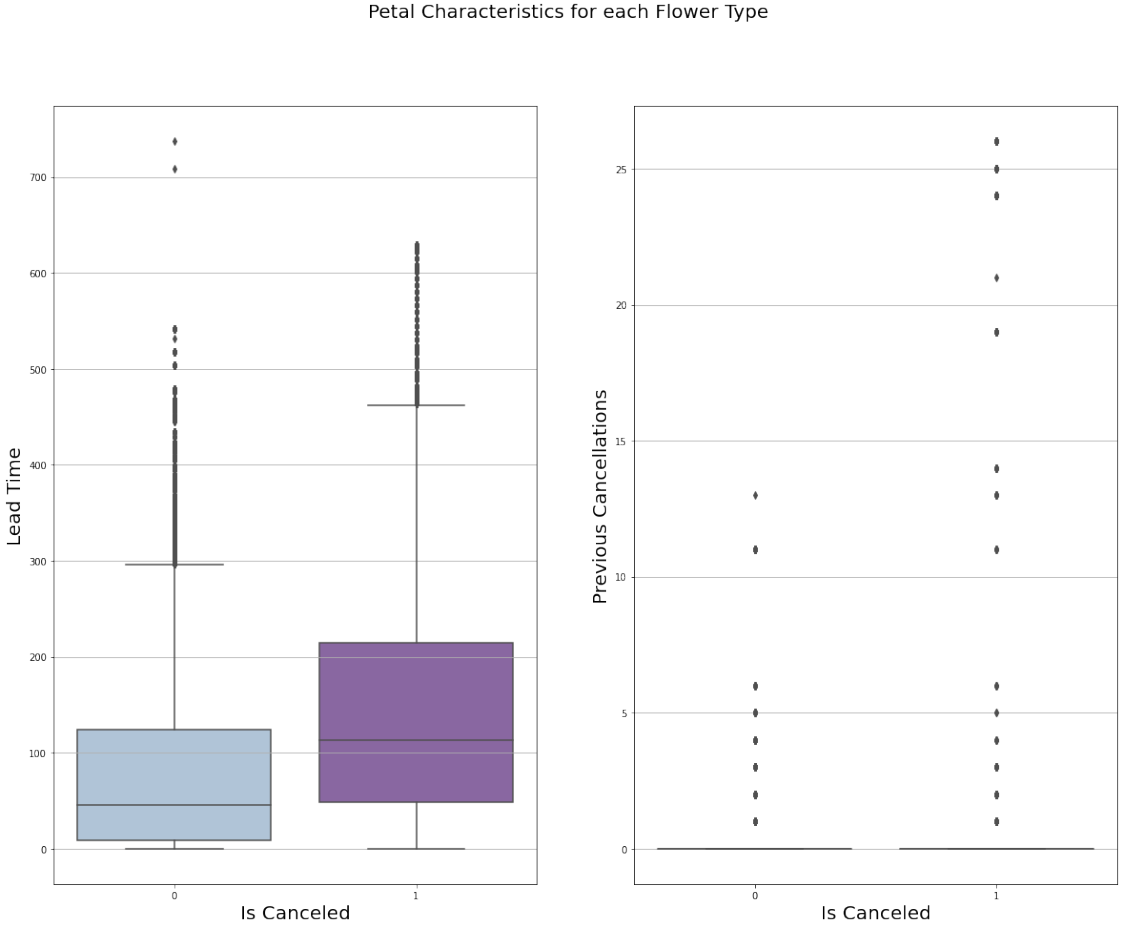
[427]:
```
data.groupby("Total_Counts")[["adults", "children","babies"]].sum().plot.
 ↪bar(figsize = (12,12), color =('lightcoral','olivedrab', 'darkviolet'))

plt.xlabel("Total Counts", fontsize = 14)
plt.ylabel("The number of Adults with (Children and/or Babies) who Cancelled",␣
 ↪fontsize = 14)
plt.title("The Proporation of Cancelled Rates", fontsize = 18)
plt.legend(["Adults", "Children", "Babies"], fontsize = 12)
plt.grid(axis = "y")
plt.show()
```



4.4 4. What's the proportion of the cancellation rates?

[352]:
```
#

fig, axes = plt.subplots(1, 2, figsize=(20,15))

sns.boxplot(x='is_canceled', y="lead_time", data=df, palette="BuPu",␣
 ↪orient='v', ax=axes[0])
sns.boxplot(x='is_canceled', y="previous_cancellations", data=df,␣
 ↪palette="BuPu", orient='v', ax=axes[1])
```

```
fig.suptitle('The Main Characteristics for the Cancellation Rates', fontsize =␣
↪20)

axes[0].set_ylabel("Lead Time",fontsize = 20)
axes[0].set_xlabel("Is Canceled",fontsize = 20)

axes[1].set_ylabel("Previous Cancellations",fontsize = 20)
axes[1].set_xlabel("Is Canceled",fontsize = 20)

axes[0].grid(axis="y")
axes[1].grid(axis="y")

plt.show()

# https://dev.to/thalesbruno/subplotting-with-matplotlib-and-seaborn-5ei8

#https://stackabuse.com/seaborn-box-plot-tutorial-and-examples/
```



5 Interpretation:

```
-- The reason I have picked the "Lead time" & "Previous cancellations" variables are both of th
-- I wanted to discover the statistical interactions around this relationships.
-- As we can see from the box plot that the positive cancellation average is around 130 days.
Since lead time: represents the gap in days between the entering date of the booking into the s
-- From the previous point we can conclude that the longer it recording the reservation and che
-- Unexpected, there is no solid relationship between the possability of cancellation and the n
```

[ ]: