כלכלה בעולם הביג דאטא – פרויקט סופי

רקע:

המשימה:

בהינתן Dataset של AIRBNB הכולל בתוכו מספר פיצ'רים אשר יפורט אודותיהם בהמשך השייכים ל-user-ים חדשים המצטרפים לשירות, נרצה לחזות האם ה-user-ים הללו יבחרו לעשות booking ראשון בארה"ב (2), booking מחוץ לארה"ב (1) או כלל לא לעשות (0).

שפת הכתיבה:

לבחירתנו.

שפות נבחרות:

R+Python+SQL

Features:

- **id**: user id
- **date_account_created**: the date of account creation
- **timestamp_first_active**: timestamp of the first activity, note that it can be earlier than date_account_created or date_first_booking because a user can search before signing up
- **date_first_booking**: date of first booking
- **gender**
- **age**
- **signup_method**
- **signup_flow**: the page a user came to signup up from
- **language**: international language preference
- **affiliate_channel**: what kind of paid marketing
- **affiliate_provider**: where the marketing is e.g. google, craigslist, other
- **first_affiliate_tracked**: whats the first marketing the user interacted with before the signing up
- **signup_app**
- **first_device_type**
- **first_browser**
- **country_destination**: this is the target variable you are to predict

הוספת משתנה חיצוני:

רקע:

נתבקשנו להוסיף משתנה חיצוני על מנת להעשיר את ה-Data.

פיתרון:

- השתמשנו בסקריפט שנכתב ב-python בשם "script_for_adding_new_columns.py" עקב נוחות וחבילות פתוחות שהשפה מספקת.
- החלטנו להשתמש בחבילה בשם "Holidays" אשר ניתן בעזרתה לדעת האם תאריך מסויים הוא חג או לא (בהתאם למדינה נבחרת).
- מכיוון שה-users הינם מארה"ב בחרנו חגים שחלים בארה"ב. בנוסף, לא בחרנו חגים במדינות אחרות, מכיוון שהחגים משתנים ממדינה למדינה ותוצאות ה-dataset מופרדות להאם המדינה היא ארה"ב, מחוץ לארה"ב או לא נעשה booking בכלל.
- בהתאם לכך, השתמשנו בחבילה זו (אשר מפורסמת בתור חבילה רשמית כחלק מ-pypi).
- בעקבות זאת, יצרנו 2 פיצ'רים חדשים המתאימים לפיצ'רים קיימים (מופיע בסוגריים):
  - **account_created_distance_US_toHoliday** (date_account_created)
  - **first_booking_distance_US_toHoliday** (date_first_booking)
- המשתנים הללו סופרים מה מספר הימים המינימאלי בין הפיצ'ר הקיים לבין החג הקרוב הקיים בארה"ב. כלומר, למשל עבור כל תאריך dateX בפיצ'ר **date_account_created** קיימת מקבילה בפיצ'ר החדש שיצרנו **account_created_distance_US_toHoliday** ובו ההפרש המינימאלי **בימים** מהתאריך ה-dateX ועד ה-closest holiday.
- הרציונאל העומד מאחוריי כך הוא ההנחה שכאשר מתקרבים לחגים, אנשים רוצים להזמין מקומות לינה ב-airbnb (ייתכן בהתאם לחג). תחת ההנחה כי אנשים נוטים להזמין מראש רצינו למצוא את הפרש הימים המתאים.

רשימת פיצ'רים התחלתית:

```
> colnames(airbnbTrain_extra_df)
 [1] "id"                                "date_account_created"
 [3] "timestamp_first_active"            "date_first_booking"
 [5] "gender"                            "age"
 [7] "signup_method"                     "signup_flow"
 [9] "language"                          "affiliate_channel"
[11] "affiliate_provider"                "first_affiliate_tracked"
[13] "signup_app"                        "first_device_type"
[15] "first_browser"                     "country_destination"
[17] "first_booking_distance_US_toHoliday"  "account_created_distance_US_toHoliday"
```

נתונים כלליים על ה-Dataset:

- **מספר פיצ'רים התחלתי:** 18
- **מספר תצפיות:** 170760

PreProcessing:

רקע:

בחלק זה נעבור על הפיצ'רים ונעבד אותם על מנת שנוכל להשתמש בהם במודל עתידי.

id:

- להלן בדיקה המראה שכל הנתונים הינם ייחודיים ואין כפילויות ב-IDים:

```
> sqldf("select count(*) from (select * from airbnbTrain_extra_df group by id)");
  count(*)
1   170760
> sqldf("select id,id_count from (select *,count(*) as id_count
+         from airbnbTrain_extra_df group by id) where id_count>1");
[1] id       id_count
<0 rows> (or 0-length row.names)
```

Age:

- בבדיקה ראשונית של ה-data על מנת לראות התפלגות, נראה כי ישנם ערכים עפ"י שנת לידה ולא עפ"י גיל:

```
> data.frame(t(sqldf("select age,count(age) from airbnbTrain_extra_df group by age")))
          X1 X2 X3 X4 X5 X6 X7 X8  X9 X10 X11 X12  X13  X14  X15  X16  X17  X18  X19  X20  X21  X22  X23  X24  X25  X26  X27
age       NA  1  2  4  5 15 16 17  18  19  20  21   22   23   24   25   26   27   28   29   30   31   32   33   34   35   36
count(age)  0  2  7  2 36  7 19 54 548 864 422 784 1389 1968 2583 3602 3998 4575 4740 4746 4908 4807 4690 4413 4016 3898 3265
           X28  X29  X30  X31  X32  X33  X34  X35  X36  X37  X38  X39  X40  X41  X42 X43 X44 X45 X46 X47 X48 X49 X50 X51 X52
age         37   38   39   40   41   42   43   44   45   46   47   48   49   50   51  52  53  54  55  56  57  58  59  60  61
count(age) 2971 2670 2375 2230 2023 1790 1623 1726 1744 1509 1307 1176 1072 1096 1049 983 912 813 811 762 747 650 629 592 530
          X53 X54 X55 X56 X57 X58 X59 X60 X61 X62 X63 X64 X65 X66 X67 X68 X69 X70 X71 X72 X73 X74 X75 X76 X77 X78 X79 X80 X81
age        62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
count(age) 489 439 441 421 340 320 296 224 213 146 155 116  99  76  57  60  38  40  35  27  25  21  14  17  22  23   9   7  14
          X82 X83 X84 X85 X86 X87 X88 X89 X90 X91 X92 X93 X94 X95 X96 X97 X98 X99 X100 X101 X102 X103 X104 X105 X106 X107 X108
age        91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108  109  110  111  113  115  132  150 1924 1925
count(age)  9  13  15   9  38  16   8  11  13  22  19  29  24  42 914  14  20   8   24  151    1    1   11    1    1    2    1
          X109 X110 X111 X112 X113 X114 X115 X116 X117 X118 X119 X120 X121 X122 X123 X124 X125
age       1926 1927 1928 1929 1931 1932 1933 1935 1936 1942 1947 1949 1953 1995 2008 2013 2014
count(age)    1    1    1    2    3    2    1    1    1    1    2    2    1    1    1   35  572
>
```

```
> sqldf("select count(*) from airbnbTrain_df where age is NULL") #70398
  count(*)
1    70398
```

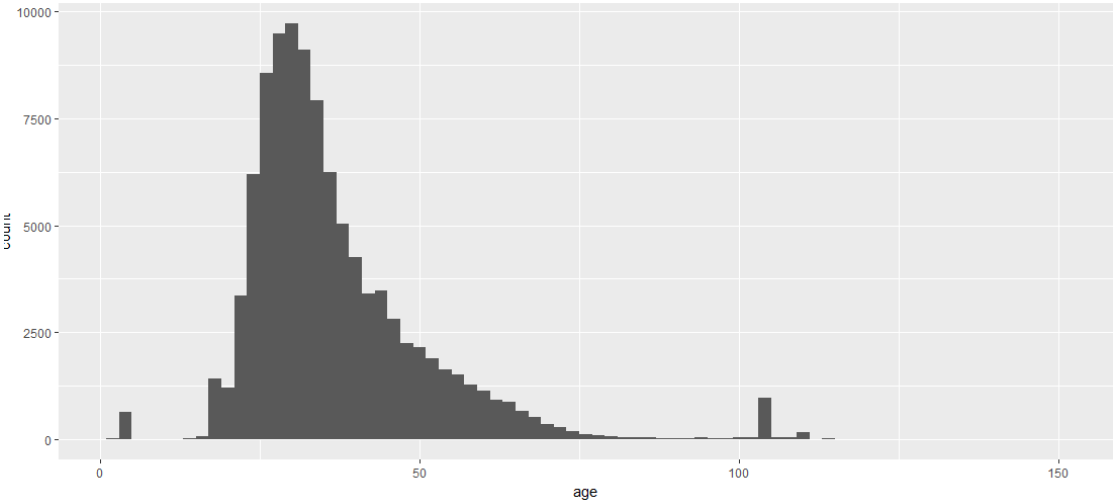
- קיימים 70398 ערכים חסרים (NA)
- קיימים גילאים בשנים (החל מ-1900)
- קיימים גילאים לא הגיוניים כמו 132 ו-150. כמו-כן הסבירות שאנשים בני 100 ומעלה או בני 15 ומטה ישתמשו ב- airbnb ויעשו booking היא נמוכה מאוד.

- לכן, נחליף את כל הנתונים תחת עמודת age שעבורם הגיל הוא מעל 1900 ל-2018-age, אחרת נשאיר כרגיל:

```
airbnbTrain_extra_df = airbnbTrain_extra_df[, age := ifelse(age >= 1900, 2018-age, age)]
```

- כעת ההתפלגות יותר הגיונית:

```
> data.frame(t(sqldf("select age,count(age) from airbnbTrain_extra_df group by age")))
            X1 X2 X3  X4 X5 X6 X7 X8 X9 X10 X11 X12 X13  X14  X15  X16  X17  X18  X19  X20  X21  X22  X23  X24  X25  X26  X27
age         NA  1  2   4  5 10 15 16 17  18  19  20  21   22   23   24   25   26   27   28   29   30   31   32   33   34   35
count(age)   0  2  7 574 71  1  7 19 54 548 864 422 784 1389 1969 2583 3602 3998 4575 4740 4746 4908 4807 4690 4413 4016 3898
            X28  X29  X30  X31  X32  X33  X34  X35  X36  X37  X38  X39  X40  X41  X42  X43 X44 X45 X46 X47 X48 X49 X50 X51 X52
age          36   37   38   39   40   41   42   43   44   45   46   47   48   49   50   51  52  53  54  55  56  57  58  59  60
count(age) 3265 2971 2670 2375 2230 2023 1790 1623 1726 1744 1509 1307 1176 1072 1096 1049 983 912 813 811 762 747 650 629 592
           X53 X54 X55 X56 X57 X58 X59 X60 X61 X62 X63 X64 X65 X66 X67 X68 X69 X70 X71 X72 X73 X74 X75 X76 X77 X78 X79 X80 X81
age         61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89
count(age) 530 489 439 441 422 340 320 296 226 213 148 155 116  99  76  58  60  38  40  35  27  26  22  14  18  24  26   9   9
           X82 X83 X84 X85 X86 X87 X88 X89 X90 X91 X92 X93 X94 X95 X96 X97 X98 X99 X100 X101 X102 X103 X104 X105 X106 X107
age         90  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107  108  109  110  111  113  115  132  150
count(age)  15  10  14  16  11  38  16   8  11  13  22  19  29  24  42 914  14  20    8   24  151    1    1   11    1    1
>
```

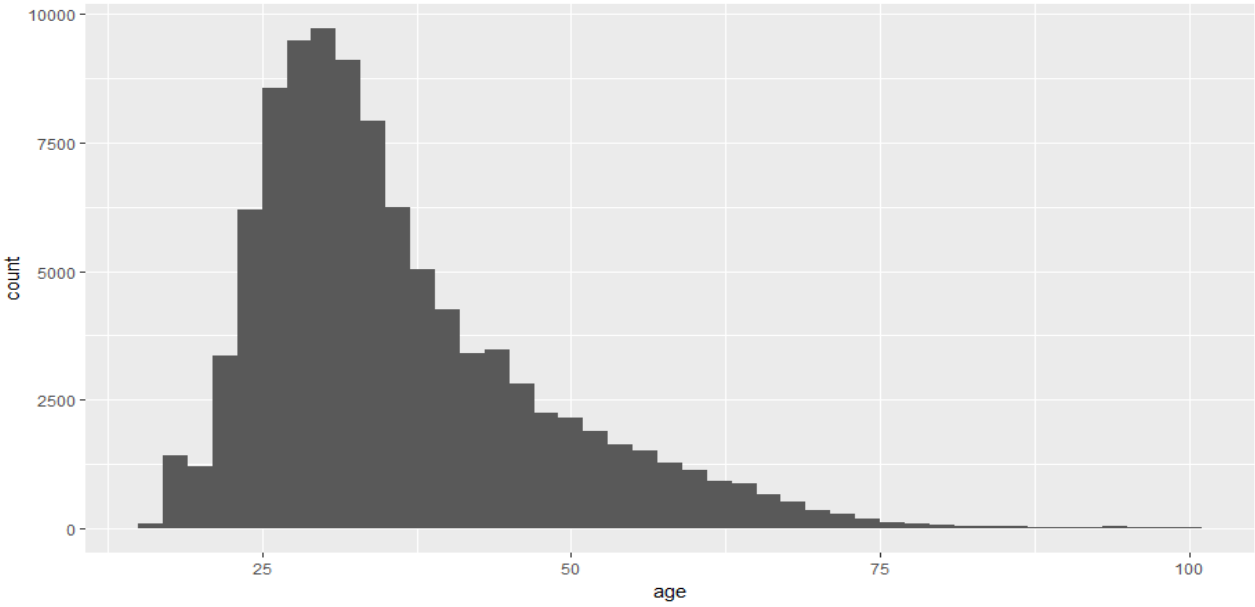


- ניתן לראות שיש כאן התפלגות זנב-עבה יחד עם outliers.
- נבדוק כמה כאלה קיימים שהם מתחת לגיל 15 ומעל גיל 100:

```
sqldf("select count(*) from airbnbTrain_extra_df where age<15 or age>100")
count(*)
    1915
```

- בחרנו לשנות את כל מי שבטווח הגילאים הנ"ל לערכים שהם NA (אשר קיימים כבר כ-70,000 כאלה) מ-2 סיבות:
  1. מכיוון שיש יותר מ-10% מה-data אנשים בגילאים האלה, לחתוך אנשים אלה מה-data עלול להשפיע יותר מדי על ה-data.
  2. ייתכן ואנשים שציינו גיל לא הגיוני, יציינו נתונים נוספים לא הגיוניים. אולם, מבדיקה שנעשתה, שאר הנתונים שלהם הגיוניים לכן נעדיף לא לחתוך את ה-users האלה מה-data.
- אופציה אחרת היא להפוך אותם לערך שאינו בשימוש – למשל את כולם לערך 1-, זאת כי ייתכן וישנה משמעות לערכים חסרים, או משמעות ל-outliers. אך ככל הנראה מדובר

בטעויות או שאנשים רשמו או גיל מזוייף ולכן זה סביר והגיוני להפוך אותם ל-NA במקום לערך משלהם.



- בסה״כ קיימים 72313 ערכי NA בפיצ׳ר age.

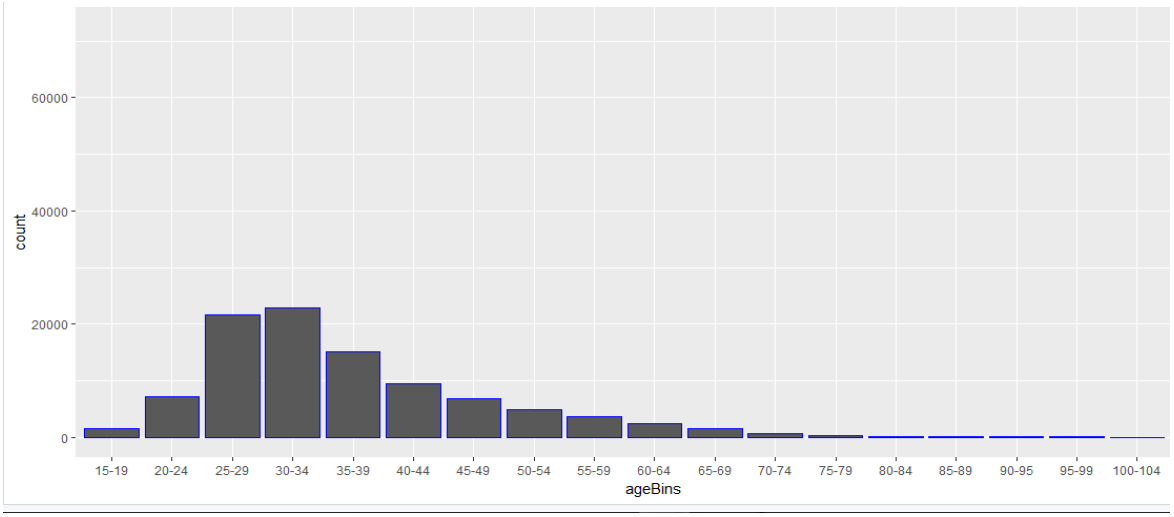
```
sqldf("select count(*) from airbnbTest_df_real where age is NULL") #72313
count(*)
   72313
```

- סיכום של ה-data ב-age:

```
summary(airbnbTrain_extra_df$age)
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
15.00   28.00   34.00   36.57   42.00  100.00   72336
```

ageBins

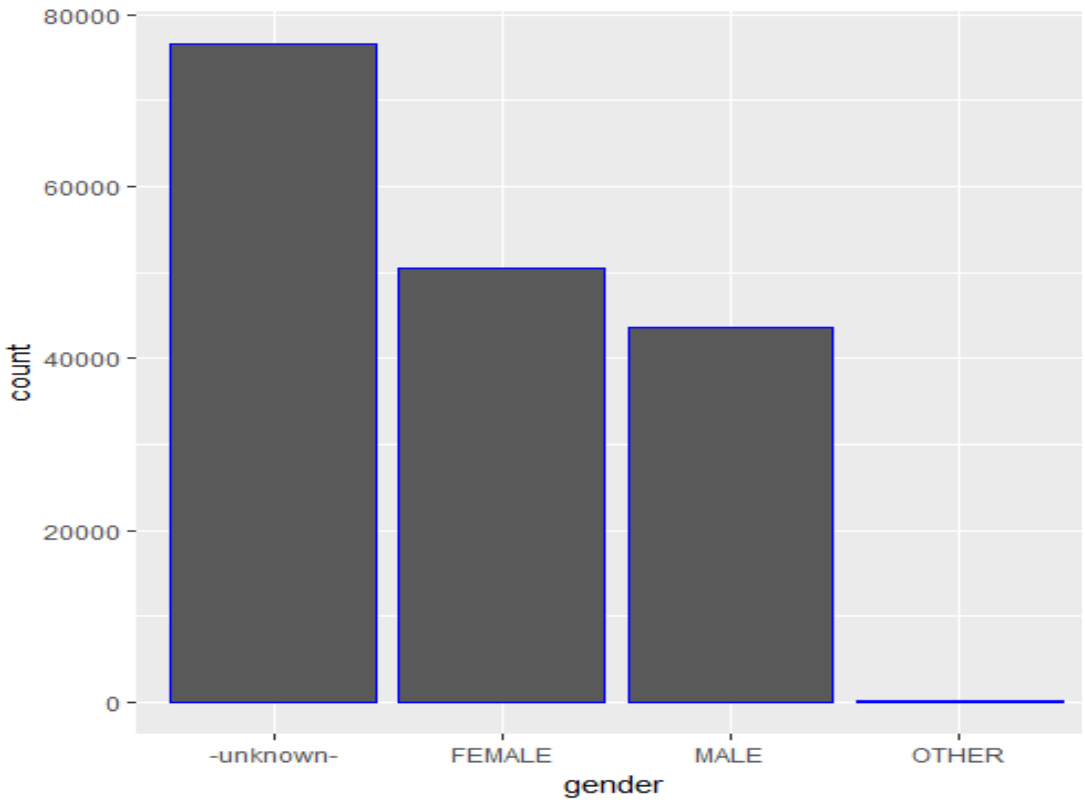
- החלטנו ליצור משתנה נוסף קטגוריאלי עבור age אשר ייתכן וישמש אותנו בהמשך בשם ageBins.
- משתנה זה מאחד כל 5 שנים לידי קטגוריה אחת . לדוגמא, גילאים 15-19 יאוחדו תחת קטגוריה אחת.
- הייתה דילמה האם לחלק זאת ל-bins עפ״י כמות, כלומר שכל bin יכיל את אותה כמות של משתנים, או לחלק לפי גילאים. לבסוף החלטנו לחלק לפי גילאים. אמנם הדבר יוצר bias יותר גדול ול-bins שהם outliers ישנה יותר חשיבות, אך יותר הגיוני לחלק את הגילאים לפי קבוצות גיל ולא לפי כמות שווה של data.
- **יש לזכור** כי גם בפיצ׳ר זה קיימים 72313 ערכי NA.
- להלן גרף ההיסטוגרמה של ageBins:



gender

- ניתן לראות את החלוקה לפי מגדר באופן הבא:

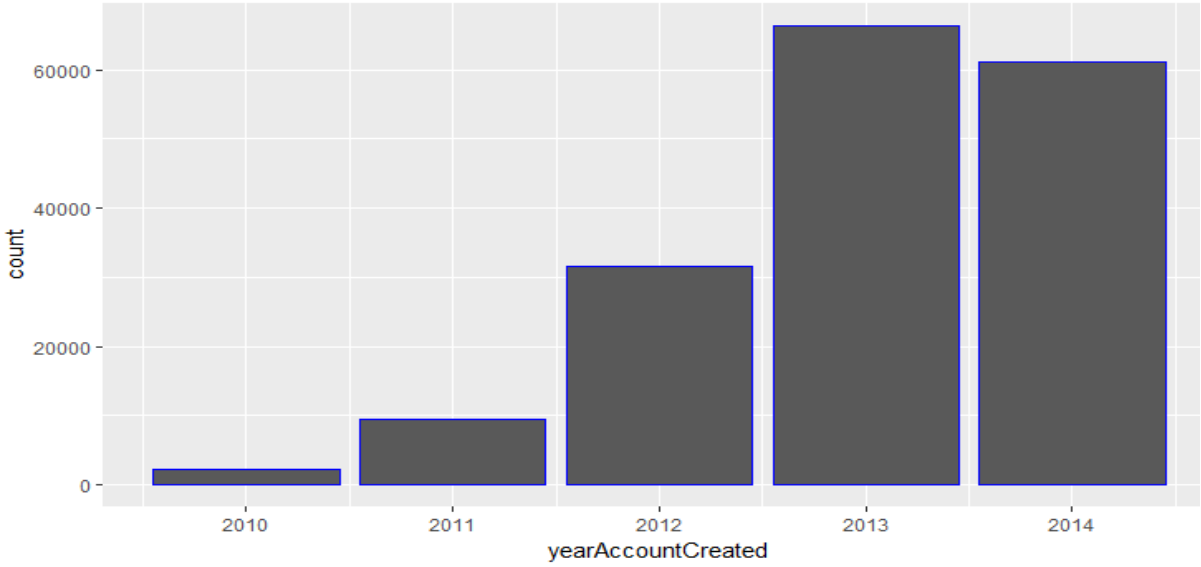
```
sqldf("select gender,count(gender) from airbnbTrain_df group by gender")
   gender count(gender)
-unknown-         76534
   FEMALE         50444
     MALE         43553
    OTHER           229
```



- ניתן לראות שיש כ-75,000 אנשים אשר צויינו כ-"unknown" לכן נהפוך אותם מ-string ל-NA.
- כמו-כן ניתן לראות כי יש כ-229 תצפיות של OTHER. אמנם מדובר במספר תצפיות מועט (כ-0.001% מה-train), אולם ייתכן והוא בעל משמעות עבור אנשים שאינם רואים את עצמם שייכים למגדר. בנוסף, מבדיקה שנערכה על שאר הפיצ'רים של ה-user-ים בעלי Gender של OTHER, לא נראו דברים חריגים. על-כן, הוחלט להשאיר את OTHER כמו שהוא.
- כמו-כן, הוחלט להפוך את GENDER למשתנה קטגוריאלי (ממשתנה מסוג character) בעל 4 קטגוריות.

date_account_created

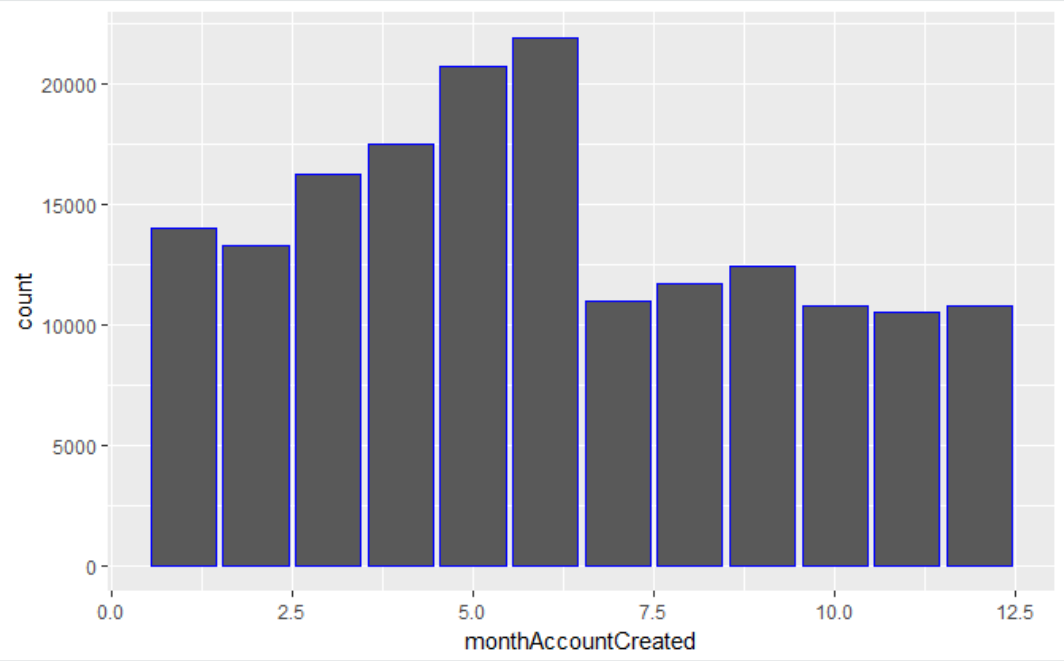
- בפיצ'ר זה לא קיימים כלל ערכי NA.
- יצרנו פיצ'ר חדש זהה בפורמט Datetime בשם **dateAccountCreated**.
- על בסיס פיצ'ר **dateAccountCreated** יצרנו 3 פיצ'רים נוספים בשם:
  - **yearAccountCreated** – השנה בה נוצר החשבון.
  - **monthAccountCreated** – החודש בו נוצר החשבון.
  - **weekdayAccountCreated** – היום בשבוע בו נוצר החשבון.
- **yearAccountCreated**:
  - ניתן לראות שרב המידע הוא בין השנים 2013 ו-2014:



  - ניתן לראות במספרים את כמות הרשומות מהתפלגות השנים:

yearAccountCreated	YearCount
2010	2243
2011	9429
2012	31535
2013	66352
2014	61201

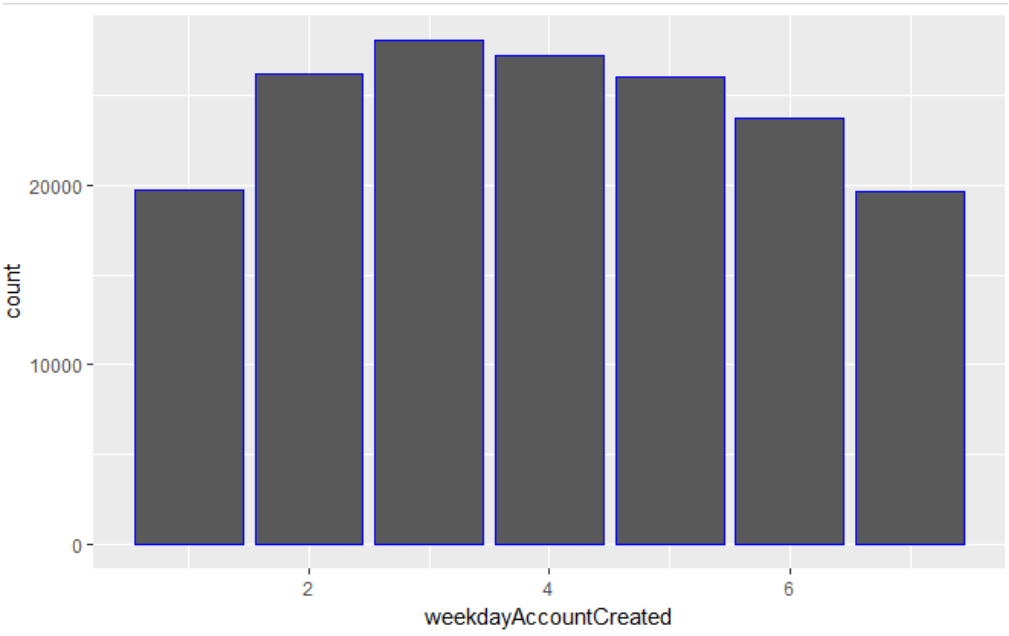
- **monthAccountCreated** :
  - ניתן לראות שאין אינדקציה לחריגות בחודשים :



  - ומבחינה מספרית :

monthAccountCreated	MonthCount
1	13995
2	13304
3	16268
4	17484
5	20721
6	21868
7	11000
8	11669
9	12426
10	10750
11	10534
12	10741

- **weekdayAccountCreated** :
  - ניתן לראות שאין אינדקציה לחריגות מבחינת ימות השבוע :



- ומבחינה מספרית:

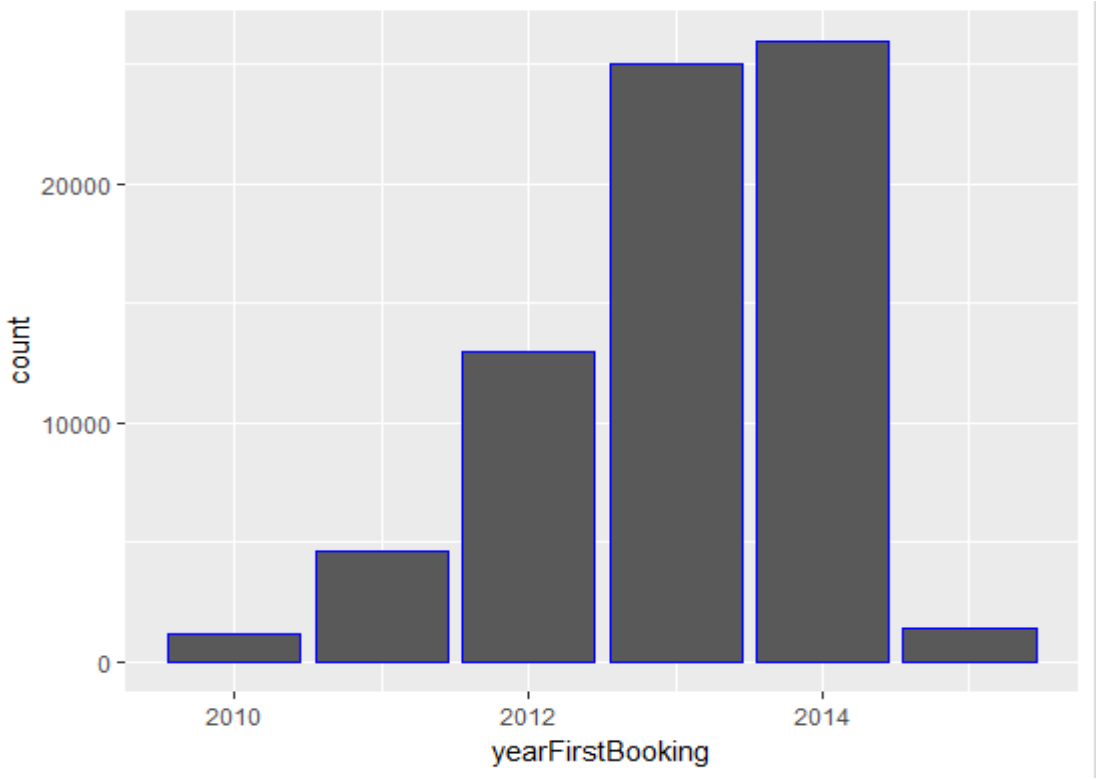
weekdayAccountCreated	weekdayCount
1	19745
2	26252
3	28077
4	27246
5	26065
6	23751
7	19624

date_first_booking

- בפיצ'ר זה כלל לא קיימים ערכי NA אולם מכיוון שהוא מוגדר ב-character, ערכי ה- NA שלו הושמו כמחרוזת ריקה "". מבדיקה של הכמויות עולה כי קיימים 99661 ערכים כאלה. את ערכים אלה נמיר להיות NA.

```
sqldf("select count(*) from airbnbTrain_df where date_first_booking==''") #99661
count(*)
   99661
```

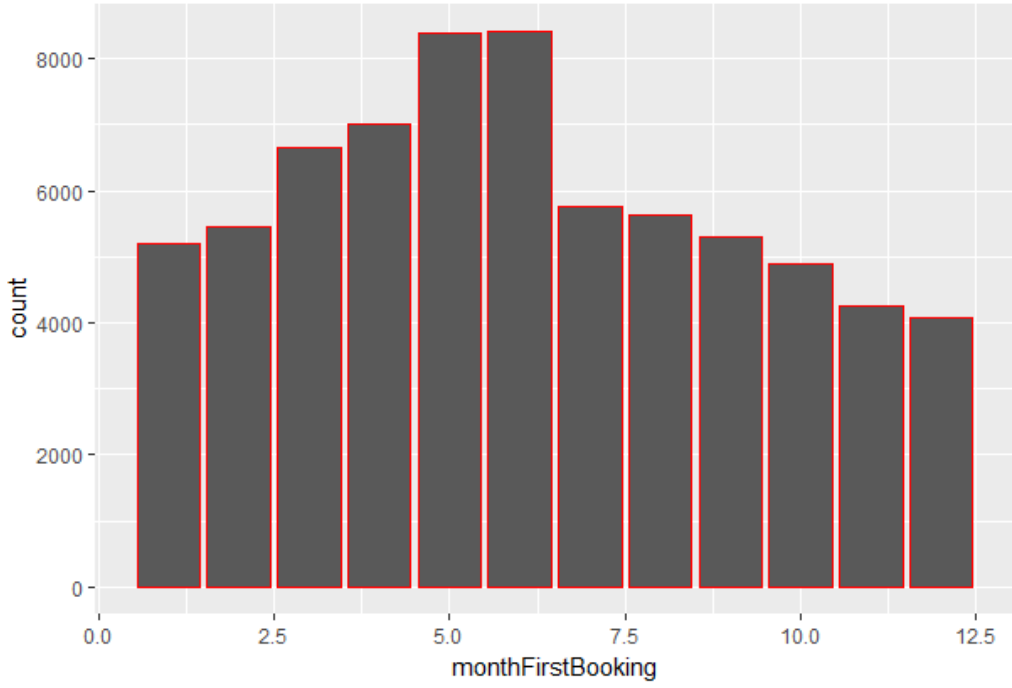
- יצרנו פיצ'ר חדש זהה בפורמט Datetime בשם **dateFirstBooking**.
- על בסיס פיצ'ר **dateAccountCreated** יצרנו 3 פיצ'רים נוספים בשם:
  - **yearFirstBooking**– השנה בה התרחש ה-booking הראשון.
  - **monthFirstBooking**– החודש בו התרחש ה-booking הראשון.
  - **weekdayFirstBooking**– היום בשבוע בו התרחש ה-booking הראשון..
- **yearFirstBooking**:
  - ניתן לראות שרב המידע הוא בין השנים 2013 ו-2014:



- ניתן לראות במספרים את כמות הרשומות מהתפלגות השנים:

yearFirstBooking	YearCount
NA	99661
2010	1177
2011	4610
2012	12943
2013	25008
2014	25930
2015	1431

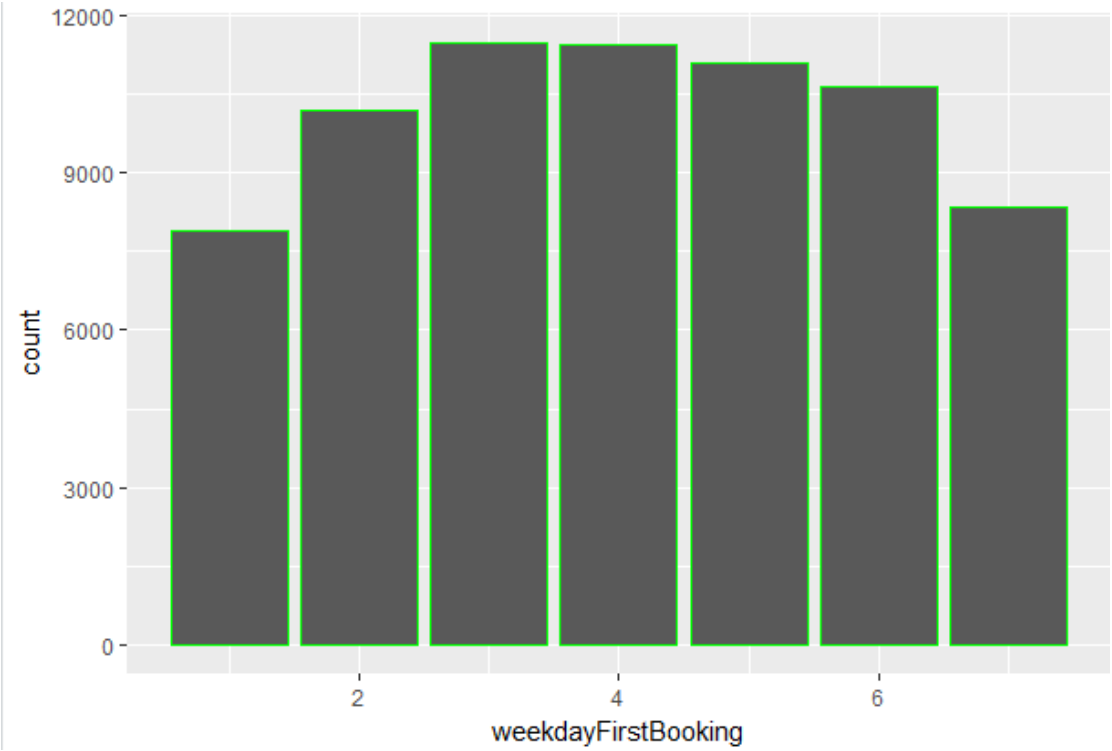
- **monthFirstBooking**:
  - ניתן לראות שאין אינדקציה לחריגות בחודשים NA:



  - ומבחינה מספרית:

1	NA	99661
2	1	5210
3	2	5443
4	3	6666
5	4	7024
6	5	8402
7	6	8408
8	7	5770
9	8	5636
10	9	5298
11	10	4900
12	11	4255
13	12	4087

- **weekdayFirstBooking**:
  - ניתן לראות שאין אינדקציה לחריגות מבחינת ימות השבוע:



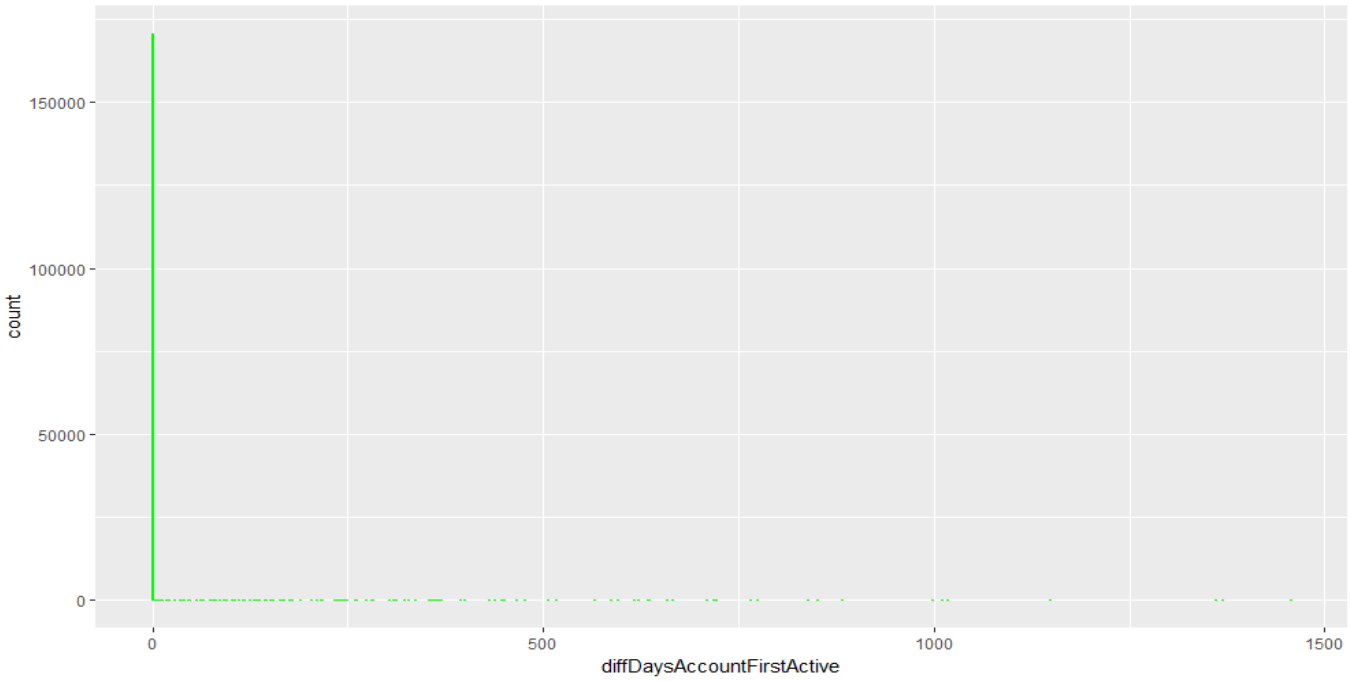
- ומבחינה מספרית:

weekdayFirstBooking	WeekdayCount
NA	99661
1	7887
2	10205
3	11467
4	11455
5	11104
6	10649
7	8332

Timestamp_first_active

- בפיצ'ר זה מדובר על מתי ה-User היה active בפעם הראשונה.
- נשים לב כי אין כלל ערכים חסרים בפיצ'ר זה.
- ניצור 2 משתנים חדשים ממנו:
  - **timestampFirstActive** – נמצא בפורמט רגיל של תאריך
  - **dateTimestampFirstActive** – חילוץ של התאריך ללא מרכיב ה-time
- <u>הוספת פיצ'רים נוספים רלוונטיים:</u>
  - **diffDaysAccountFirstActive** – ההבדל במספר הימים מהרגע שהחשבון נפתח ועד לרגע שהייתה בו פעילות ראשונה.

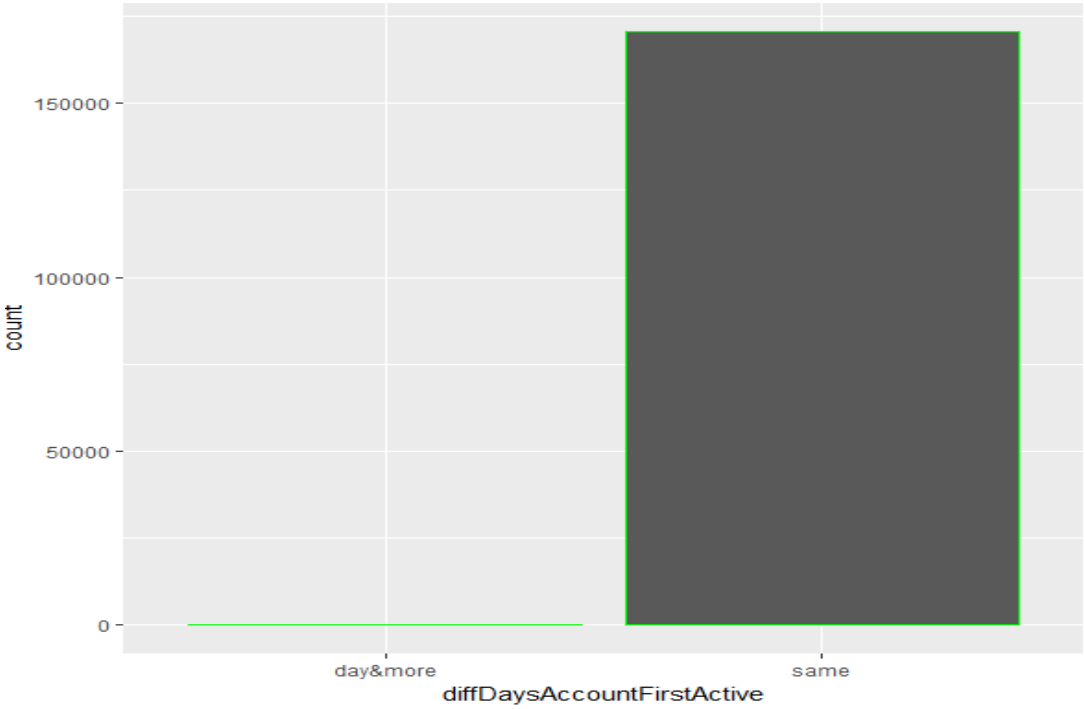
1. ניתן לראות שרב פתיחת החשבונות התרחשה באותו יום של הפעילות הראשונה, אך ישנם בודדים שהדבר אינו כך עבורם, כלומר פתיחת החשבון התרחשה יום ומעלה לאחר הפעילות הראשונה באתר:



2. ניתן לראות שישנו רב מוחץ. ישנם 170614 ערכים עבורם פעילות ראשונה ופתיחת חשבון התרחשן באותו היום ו-146 בימים שאחרי:

```
> sqldf("select count(*) from airbnbTest_df_real where diffDaysAccountFirstActive==0")
  count(*)
1   170614
> sqldf("select count(*) from airbnbTest_df_real where diffDaysAccountFirstActive>0")
  count(*)
1      146
>
```

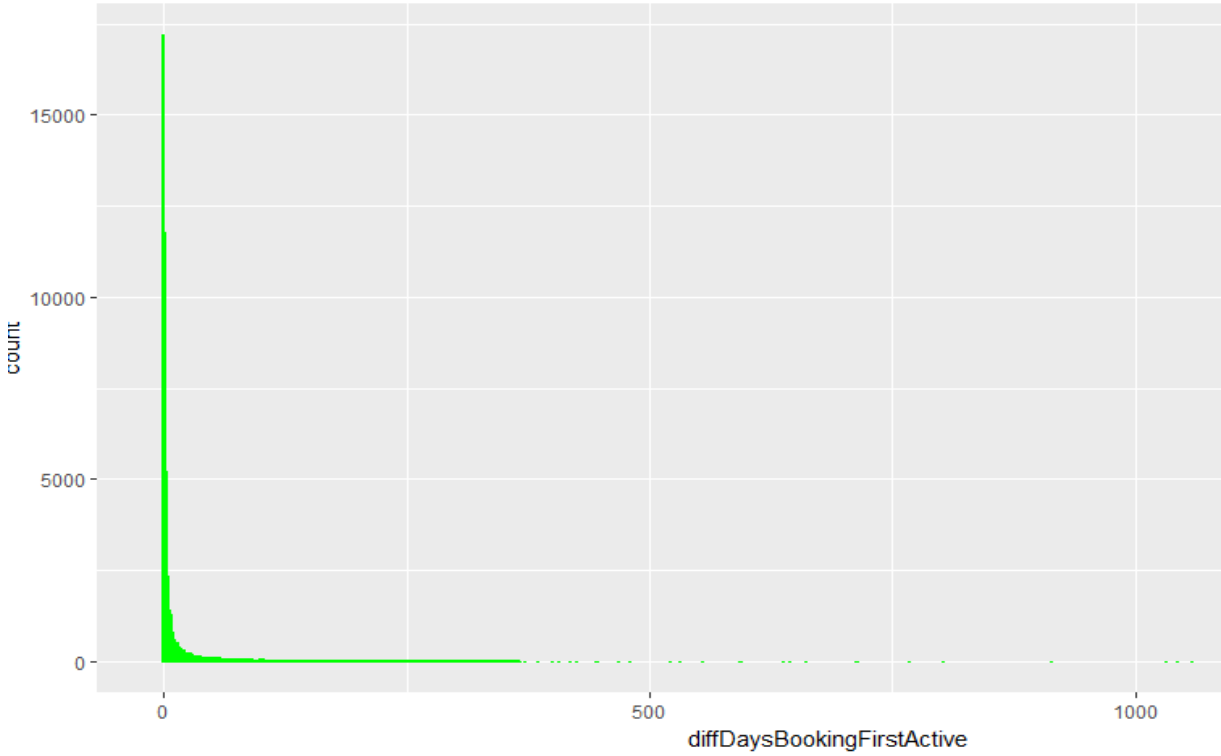
3. לכן, החלטנו להפוך את המשתנה למשתנה קטגוריאלי בינארי אשר יכיל עמודת אנשים שנרשמו באותו יום של הפעילות הראשונה וכאלה של יום אחרי ומעלה.



- **diffDaysBookingFirstActive** – ההבדל במספר הימים מהרגע שהיה booking ראשונה ועד לפעילות הראשונה בחשבון.

1. במקרה זה יש לנו התפלגות זנב, לכן החלטנו להשאיר זאת באותו אופן שהתקבל ולא להפוך למשתנה קטגוריאלי:

```
> sqldf("select diffDaysBookingFirstActive, count(*) from air
bnbTest_df_real group by diffDaysBookingFirstActive")
    diffDaysBookingFirstActive count(*)
1                           NA    99661
2                            0    17198
3                            1    11796
4                            2     5204
5                            3     3206
6                            4     2356
7                            5     1831
8                            6     1429
9                            7     1316
```

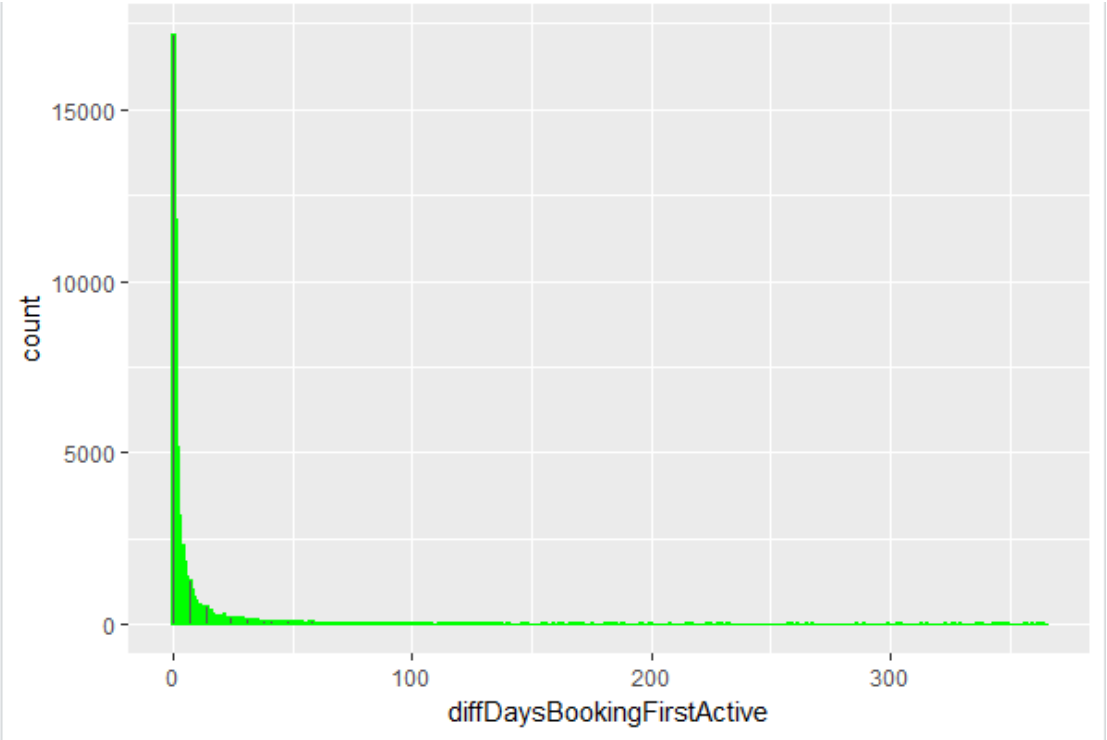


- נשים לב שבהתפלגות הזנב יש הרבה outliers. מכיוון שכך ולאחר בדיקות נראה כי לאחר 365 יום הכמויות מתחילות להיות 1 או 2 ובסה"כ קיימות 32 רשומות כאלה. מכיוון שאלו הן לא הרבה רשומות, נסיר אותן.

```
365                    363      34
366                    364      53
367                    365      39
368                    366       2
369                    367       1
370                    371       1
371                    385       1
```

```
sqldf("select sum(count)
      from (
          select diffDaysBookingFirstActive, count(*) as count
          from airbnbTrain_df_real
          group by diffDaysBookingFirstActive
          having count<3
      )")
sum(count)
        32
```

- ולאחר השינוי:

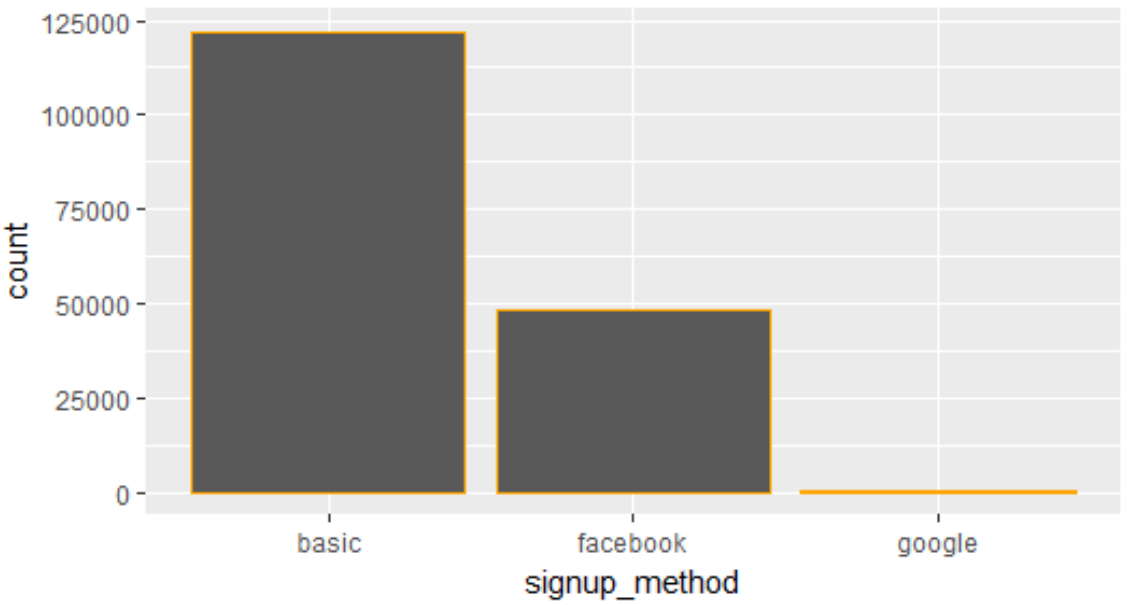


Signup method:

- בפיצ'ר זה מדובר דרך איזו שיטה נרשם ה-user לאתר Airbnb.
- נראה את ההתפלגות של הערכים:

```
> sqldf("select signup_method, count(*) from airbnbTrain_df group by signup_method")
  signup_method count(*)
1         basic   122185
2      facebook    48142
3        google      433
>
```

- נשים לב כי אין כלל ערכים חסרים בפיצ'ר זה.
- כמו-כן נשים לב שהשיטה הכי נפוצה היא דרך האתר של Airbnb.

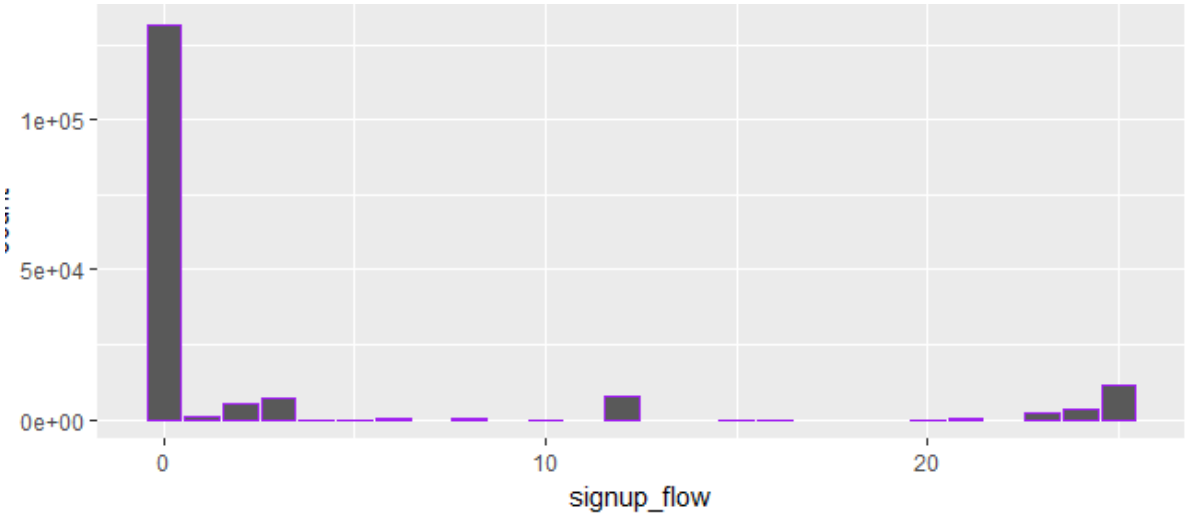


- מכיון שישנם רק 3 ערכים, נהפוך פיצ'ר זה לקטגוריאלי.

Signup flow:

- בפיצ'ר זה מדובר בכמה עמודים ה-user עבר עד שנרשם לאתר Airbnb.
- בפיצ'ר זה אין ערכי NULL.
- ניתן לראות את ההתפלגות:

```
> data.frame(t(sqldf("select signup_flow, count(*) from airbnbTrain_df group by signup_flow")))
                X1  X2   X3   X4 X5 X6  X7  X8 X9  X10 X11 X12 X13 X14  X15  X16   X17
signup_flow      0   1    2    3  4  5   6   8 10   12  15  16  20  21   23   24    25
count(*)    131843 845 5552 7009  1 31 243 189  1 7497   5   7  10 149 2225 3465 11688
>
```

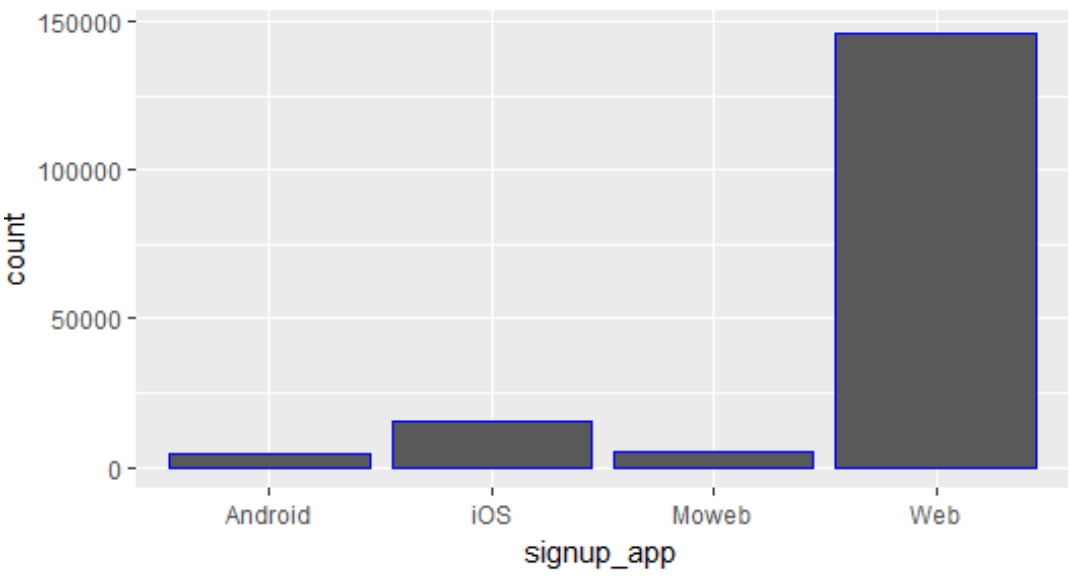


- נשים לב שהרב המוחלט עבר 0 דפים וכן שישנם מספרים בטווח שאין בהם כלל user'ים.

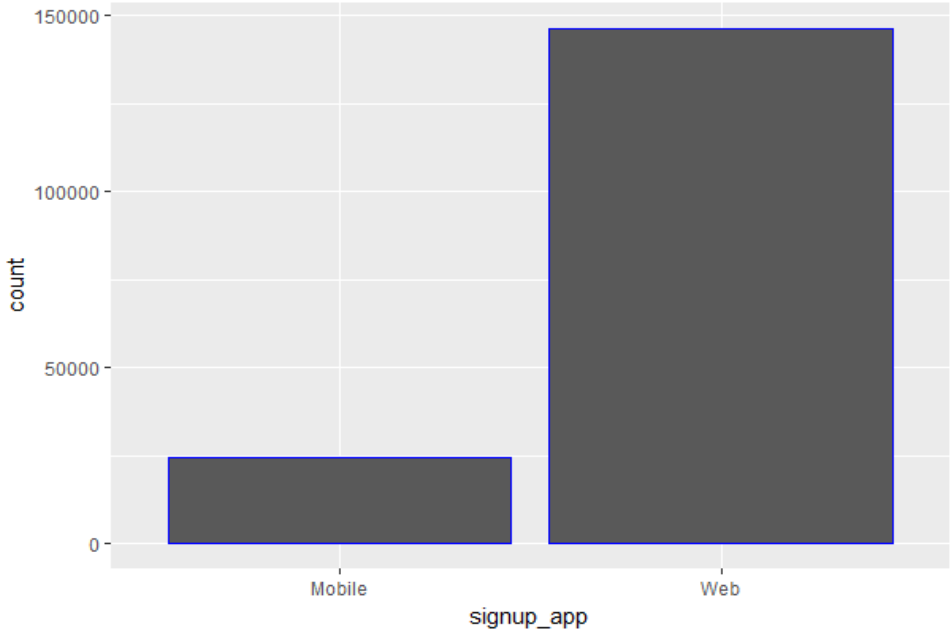
נתוני signup app :

- בפיצ'ר זה אין ערכי NA.
- ניתן לשים לב שישנם 4 קטגוריות : 3 השייכים לסוג ה- mobile ואחד ל-web :

```
sqldf("select signup_app, count(*) from airbnbTest_df_real group by signup_app")
signup_app count(*)
   Android     4334
     Moweb     5049
       Web   146218
       iOS    15159
```



- איחדנו את iphone, android, moweb תחת mobile כי נראה לנו שההבדל המהותי הוא בסוג האינטראקציה של המשתמש עם airbnb – mobile או web.
- נראה שרוב המשתמשים נכנסים airbnb דרך web :

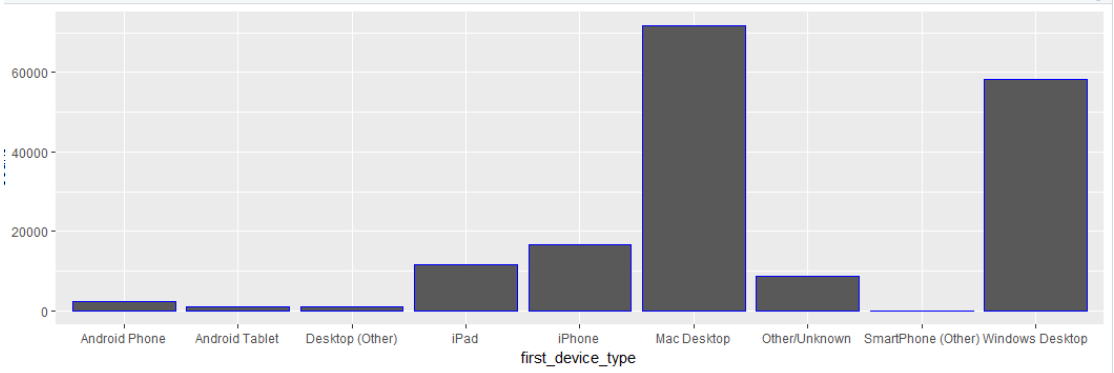


- בנוסף הפכנו את המשתנה לקטגוריאלי.

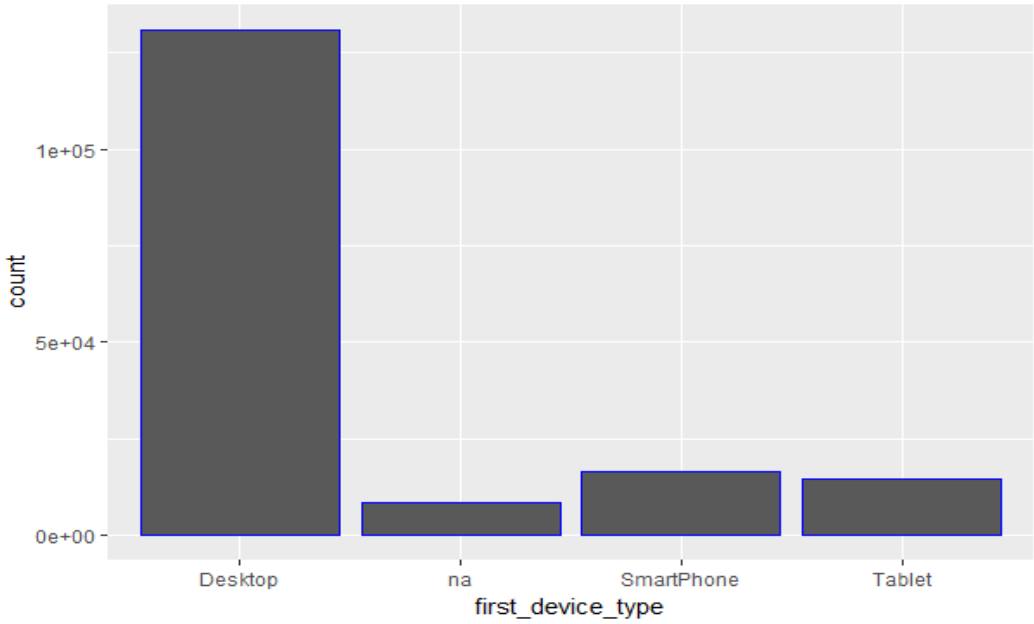
first device type :

- להלן התפלגות הנתונים בפיצ'ר זה :

```
> sqldf("select first_device_type, count(*) from airbnbTest_df_real group by first_device_type")
   first_device_type count(*)
1      Android Phone     2216
2    Android Tablet      1036
3    Desktop (Other)      985
4        Mac Desktop    71635
5      Other/Unknown     8567
6 SmartPhone (Other)       62
7    Windows Desktop    58226
8               iPad    11495
9             iPhone    16538
```



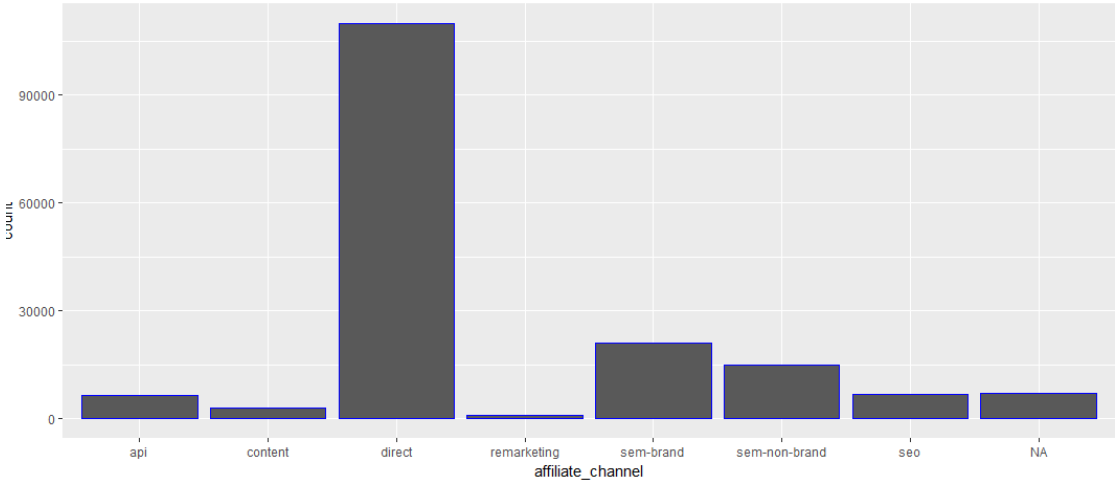
- ניתן לשים לב שיש כ-8567 ערכים שהם לא ידועים, אולם רשומים בתור String. לכן, נהפוך אותם ל-NA.
- איחדנו את iphone, android, smartphone תחת SmartPhone, ואת ipad, androidtablet תחת Tablet כי נראה לנו שההבדל המהותי הוא בסוג המכשיר. נראה שרוב המשתמשים נכנסים לAIRBNB דרך הDesktop, כלומר לא מהכשירים שזמינים לנו בכל סביבה.
- יתכן שזה מעיד על כך שכשאנשים מזמינים דירה הם עושים זאת אל מול מחשב נוח.
- הפכנו משתנה זה למשתנה קטגוריאלי.



affiliate channel :

- נראה שסוג השיווק בתשלום שממנו מגיעים הכי הרבה משתמשים הוא ישיר.
- ישנם 7191 ערכים שמוגדרים כ-Other. הפכנו אותם ל-NA.

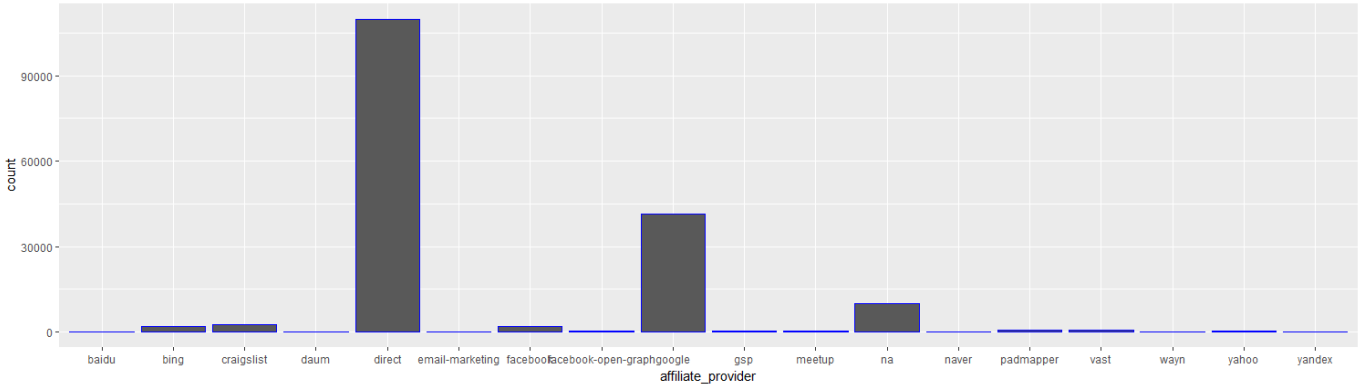
affiliate_channel	count(*)
api	6570
content	3169
direct	110012
other	7191
remarketing	861
sem-brand	20954
sem-non-brand	15108
seo	6895



- גם משתנה זה הפכנו לקטגוריאלי.

affiliate provider :

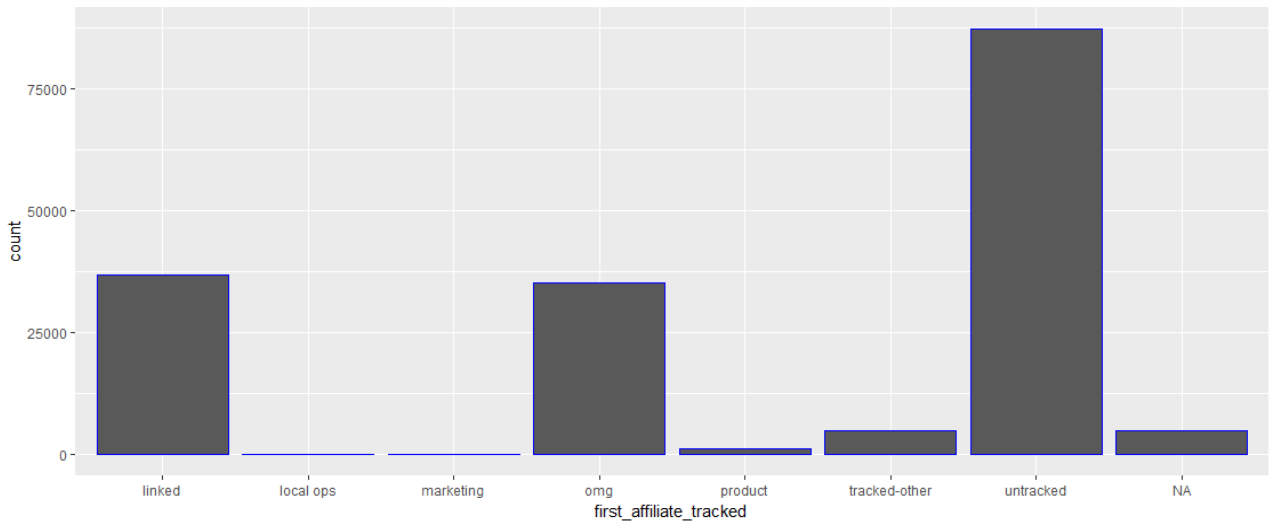
- נראה שמיקום השיווק של רוב המשתמשים הוא ישיר. יתכן שזה מעיד על כך שמי שנכנס עשה זאת במכוון ולא במקרה, כלומר יש בכוונתו להשכיר דירה או לפחות לבחון את האופציה.
- בוצע אותו טיפול עבור ערכים שהם "other" (הפיכה ל-NA) בדומה ל-affiliate channel.
- גם משתנה זה הפכנו לקטגוריאלי.



first affiliate tracked:

- ל-4894 מהתצפיות אי אפשר לדעת מה היא האינטראקציה השיווקית שגרמה להם להגיע לairbnb (תצפיות אשר הוגדרו כ- string ריק). לכן הפכנו אותן ל-NA.
- נשים לב כי קיימים בפיצ'ר זה גם untracked, גם NA וגם tracked-other. החלטנו להשאירן ב-3 קטגוריות נפרדותלמקרה שהן בעלות משמעות שונה.
- להלן התפלגות הנתונים:

```
  first_affiliate_tracked count(*)
1                    <NA>     4894
2                  linked    36933
3               local ops       27
4               marketing      105
5                     omg    35310
6                 product     1273
7           tracked-other     4984
8               untracked    87234
>
```



- אמנם יש משתנים שמופיעים מעט פעמים כמו local ops או marketing, אך בחרנו לא להשמיטם כי זוהי קטגוריה שבה יש מספר תצפיות מועטות ולא חריגה כלשהי מה-data.
- גם משתנה זה הפכנו לקטגוריאלי.

first browser:

- ישנם כ-21,778 משתנים שהם -unknown-. הפכנו אותם ל-NA.
- נראה שהאפליקציות המובילות הן chrome, safari, firefox ו-IE:

```
> sqldf("select first_browser, count(*) as count
+             from airbnbTest_df_real group by first_browser order by count desc");
         first_browser count
1               Chrome 51075
2               Safari 36206
3              Firefox 26815
4                 <NA> 21778
5                   IE 16912
6        Mobile Safari 15410
7        Chrome Mobile  1022
8      Android Browser   678
9         AOL Explorer   198
10               Opera   156
```

- על פלטפורמת desktop נראה שאין הבדל משמעותי בין PC ל Mac, אולם בsmartphone נראה שהרבה יותר משתמשי iphone נכנסים לairbnb מאשר משתמשי android.

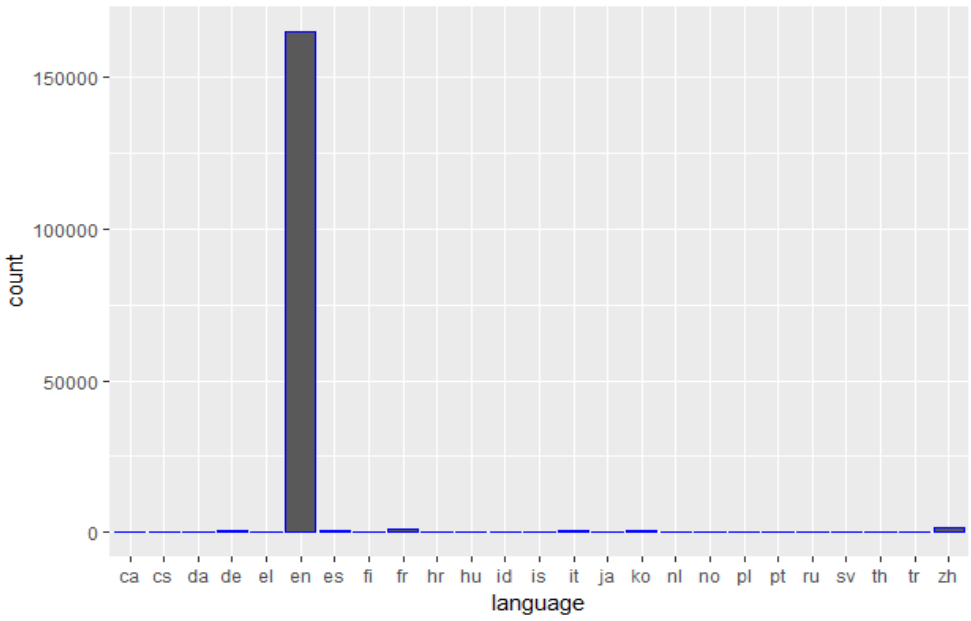
```
> table(airbnbTrain_df$first_browser)

     Android Browser         AOL Explorer           Apple Mail                Arora        Avant Browser
                 678                  198                   29                    1                    4
 BlackBerry Browser               Camino               Chrome        Chrome Mobile             Chromium
                  47                    8                51075                 1022                   56
           CometBird        Comodo Dragon              Conkeror              CoolNovo       Crazy Browser
                   9                    2                    1                    6                    2
                Epic              Firefox                Flock        Google Earth            Googlebot
                   1                26815                    2                    1                    1
           IceDragon            IceWeasel                   IE           IE Mobile                 Iron
                   1                   12                16912                   27                   12
      Kindle Browser              Maxthon       Mobile Firefox       Mobile Safari               Mozilla
                   1                   39                   21                15410                    2
                   na           NetNewsWire             OmniWeb               Opera           Opera Mini
                21778                    1                    1                 156                    4
           Pale Moon Palm Pre web browser             RockMelt               Safari             SeaMonkey
                  10                    1                   20                36206                    9
                Silk            SiteKiosk          SlimBrowser       Sogou Explorer             Stainless
                 101                   21                    2                   30                    1
           TenFourFox     TheWorld Browser           wOSBrowser       Yandex.Browser
                   7                    2                    6                    9
```

- גם משתצנה זה הפכנו לקטגוריאלי.

language:

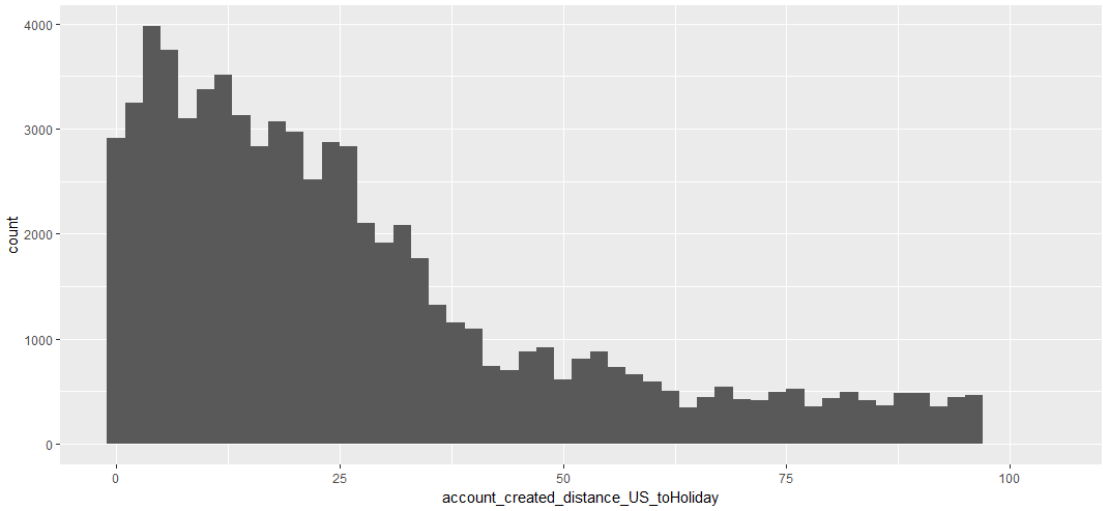
- ניתן לראות שהרוב המוחלט של המשתמשים הינו דובר אנגלית:



- נשים לב גם שאין ערכים לא ידועים בפייצ'ר זה.
- אמנם יש שפות עם מעט תצפיות, אך המשמעות הינה חשובה ולכן לא נשמיט תצפיות אלה.
- בנוסף ייתכן כי השפה משפיעה על ה-destination הסופי ולכן החלטנו לא לאחד ביחד את כל השפות ש**אינן** אנגלית לידי קטגוריה אחת.
- גם פיצ'ר זה נהפוך לקטגוריאלי.

account_created_distance_US_toHoliday:

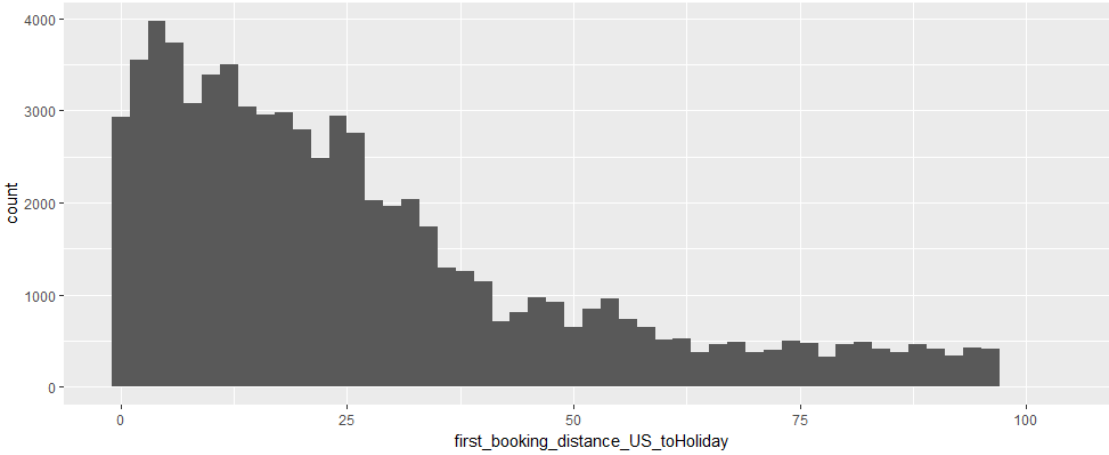
- למשתנה זה אין ערכי NULL.



- ניתן לראות כי ישנה התפלגות מעניינת ביחסת לכמות מבחינת מרחק הימים לחג הכי קרוב כאשר ניתן לראות מגמה שככל שהחג יותר קרוב, יותר חשבונות נפתחו.

first_booking_distance_US_toHoliday:

- למשתנה זה יש 99661 ערכיNULL, כמובן בהתאם לערכי ה-date_first_booking.



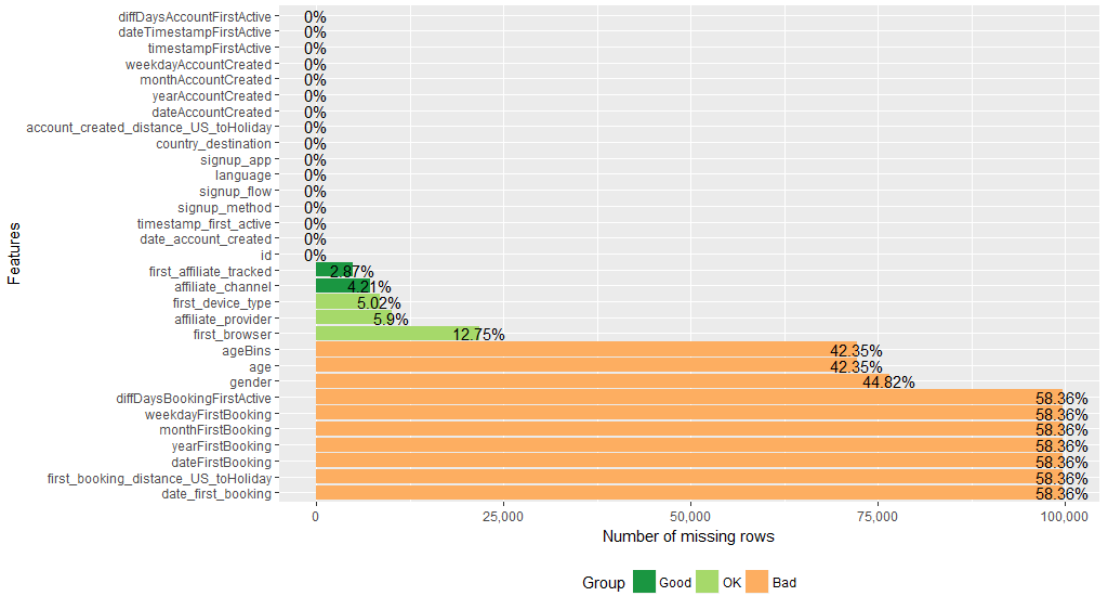
- גם כאן נית לראות קשר ברור לככל שהמרחק לחג הכי קרוב קצר יחסית, יש מגמה של יותר הזמנות (בדומה ליצירת חשבון).

נדגיש:

כל המשתנים שהיו NA נבדקו על מנת לראות האם יש משהו מחשיד בתצפיות על מנת שיהיה ניתן להסירן במידת הצורך. אולם, לא התגלה דבר חריג בנוגע לתצפיות שבהן היו ערכי NA.

טיפול בערכי NA (missing data):

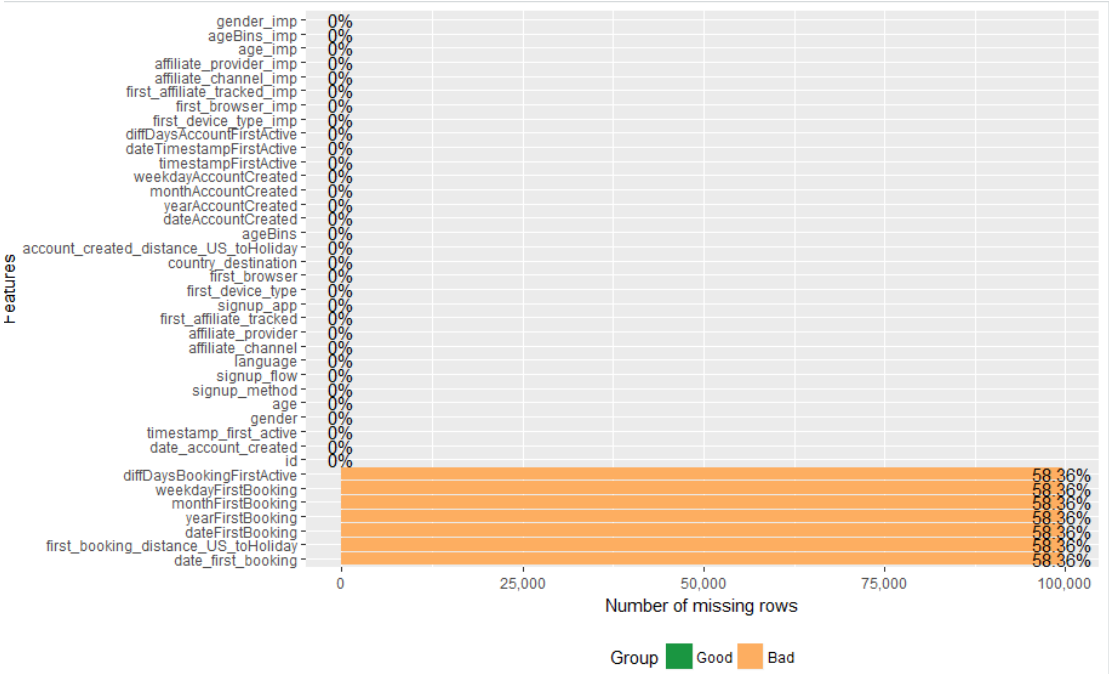
- קיימות מספר שיטות לטיפול בערכי NA:
  - במשתנים נומריים – לקחת את החציון/ממוצע – עלול ליצור bias גבוה עבור המשתנים אם יש הרבה data (כמו במקרה שלנו).
  - ניתן להשמיט – ב-data set שקילבלנו קיימות יותר מדי תצפיות מכדי להשמיט.
  - להתייחס אליו כאלה משתנה קטגורי (במקרים בהם ניתן).
  - לקבלו כחלק מה-data.
  - KNN imputation – שימוש באלגוריתם KNN על מנת להשלים את הערכים החסרים.
  - Sampling – לזהות את התפלגות הפיצ'ר בו יש ערכים חסרים ומהם, ליצור פונקציה שממפה את הערכים להתפלגות, לבחור ערך רנדומאלי מההתפלגות ולהשתמש בפונקציה ההופכית כדי לחשב את ערך המשתנה החסר.
  - שימוש ב- EM לצורך הערכה.
- ב-data שלנו יש משתנים רבים אשר מכילים ערכי NA. להלן פירוטם ויחסיהם ב-data:



- נשים לב שישנם משתנים שקשה או לא רצוי להשלימם בצורה זו – למשל כמו כל המשתנים המבוססים על הפיצ'ר date_first_booking (כל המשתנים בעלי 58.36% מדיע חסר). מכיוון שמעולם לא בוצע booking עפ"י נתוני ה-train, נכון לאותו session, אז המידע הזה חסר ונרצה להשאירו כך.
- לעומת זאת משתנים אחרים כמו פיצ'רי ה- affiliate או משתני ה-first או משתנה ה-age נרצה לטפלם בדרך כלשהי.
- נסיונות לבצע knn imputation כשלו עקב מגבלות טכניות של האמצעים אשר עמדו לרשותנו ומשך זמן החישוב הארוך הדרוש לכך.
- נסיונות לבצע EM גם כשלו בהיעדר חבילות מספיק טובות אשר ישמשו לדבר.
- על-כן, הוחלט להשתמש בשיטת imputation בשם **hotdeck**. שיטה זו אינה גזלה זמן.
- אולם, כיוון שאין אנו בטוחים בתוצאותיה, נבצע את המודלים על 2 סוגי dataset.
  1. ה-dataset עליו בוצע hoteck ובו לא קיימים ערכי NA.
  2. Dataset בו ערכי ה-NA נחשבים ל"label" כחלק ממשתנה קטגוריאלי

לכל דבר. בכל המשתנים אין עם הדבר בעיה מלבד בפיצ'ר age. זאת ניתן לפתור ב-2 דרכים:

  1. נשלים את הערכים באמצעות ממוצע.
  2. נשתמש במשתנה ה-ageBins במקום במשתנה age אשר מחלק את ה-age לקטגוריות לפי גילאים ובו NA נחשב לעוד label (הדרך שנבחרה לבסוף).
- כך נראה המידע לאחר ה-hotdeck:

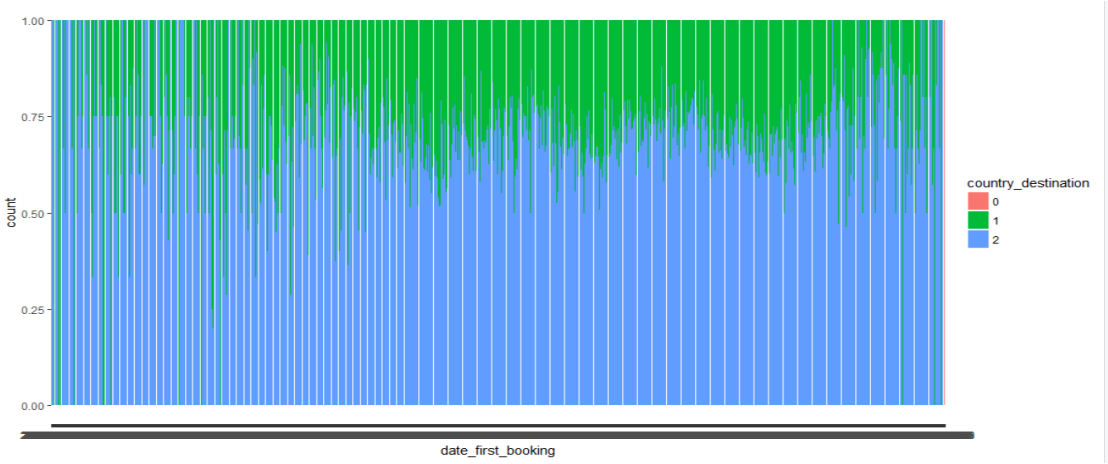


- יש לשים לב כי בכוונה השארנו את ה-booking עם ערכי NA. זאתף כיוון שבעתיד אנו מתכוונים להשתמש באופן ישיר בעובדה זו על מנת לתת "0" (כלומר אי הזמנה) למי שלא עשה booking ולהשתמש בכל מי שכן יש לו date booking בשביל להכריע אם הוא יזמין לארה"ב (2) או אל מחוץ לארה"ב (1) .

קורלציה וקשרים בין המשתנים:

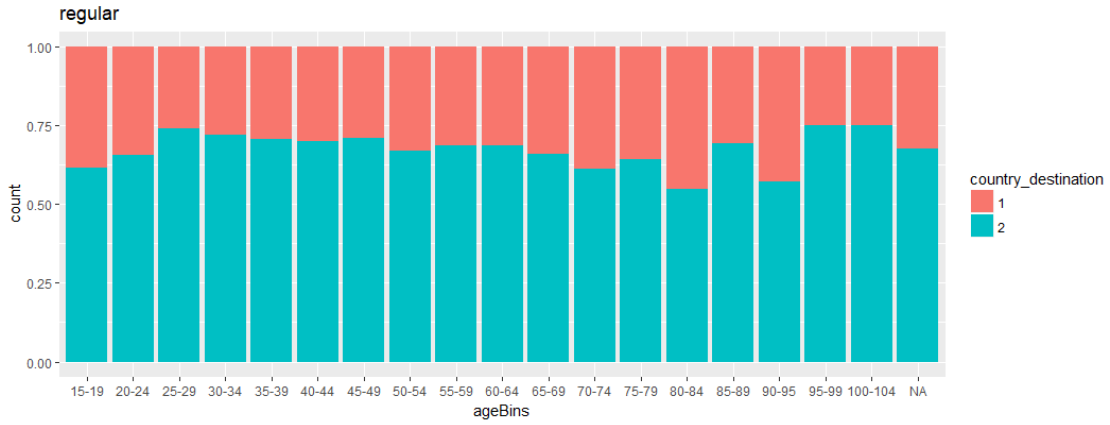
- ראשית ננסה באמצעות bar plots שונים לראות קשר בין משתנים מסויימים מובהקים לבין התוצאה המיועדת.
- לאחר מכן נבדוק קורלציות ולפיהן נחליט אילו משתנים נרצה יותר להכניס למודל בהתאם למודל.
- כמובן שיש שלל דרכים לבחור בין משתנים שונים ולעשות feature selection ויכולנו גם להשתמש בMutual Information על מנת להחליט זאת.

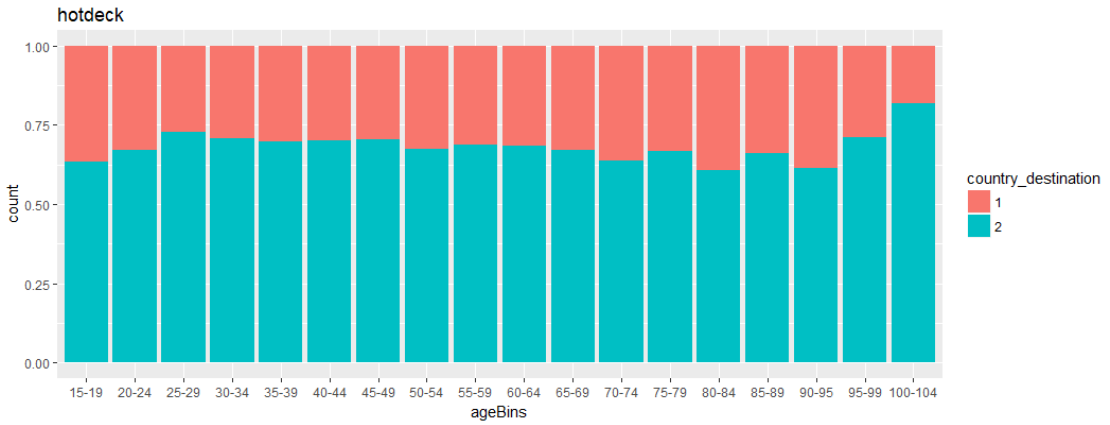
- <u>Date first booking</u>:



- ניתן לראות שרק ערכים שהם NA (מצד ימין קיצון) לאורך השנים (כלומר לא בוצע בהם כלל booking), החזירו ב-100% 0 שזה הגיוני. לכן יש פה קשר ישיר בין משתנה זה לבין התוצאה 0 בתחזית. לכן כשנעשה תחזית נשתמש רק בנתונים שה-date first booking בהן הוא לא NA ונרצה לחלקן לקלאסיפקציה להאם הבוקינג בוצע בארה"ב או מחוצה לו.
- מכיוון שאנו רואים שקשר זה משפיע, נמשיך כעת לבצע את הבדיקות שלה הקשר על subset עם הנתונים עבורם לא מופיע NA ב-date first bookin .

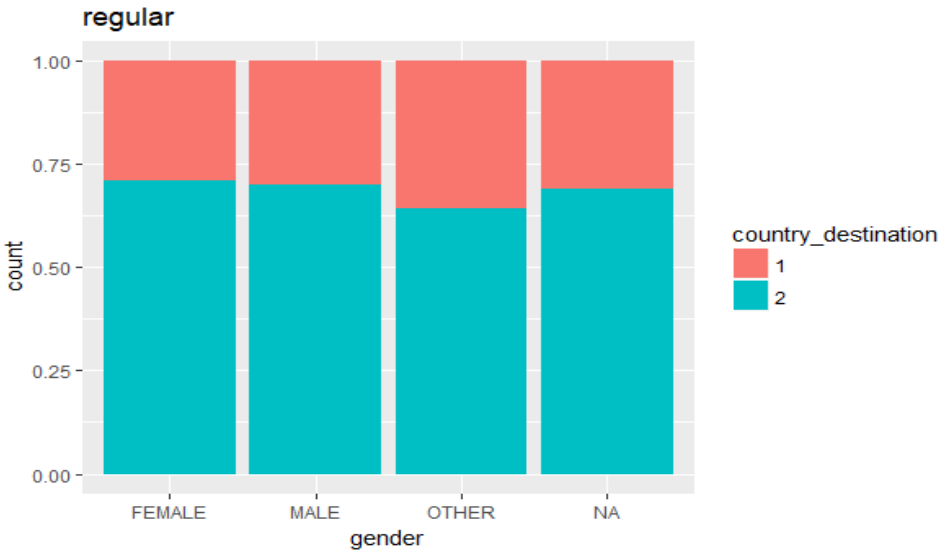
- <u>ageBins</u>:

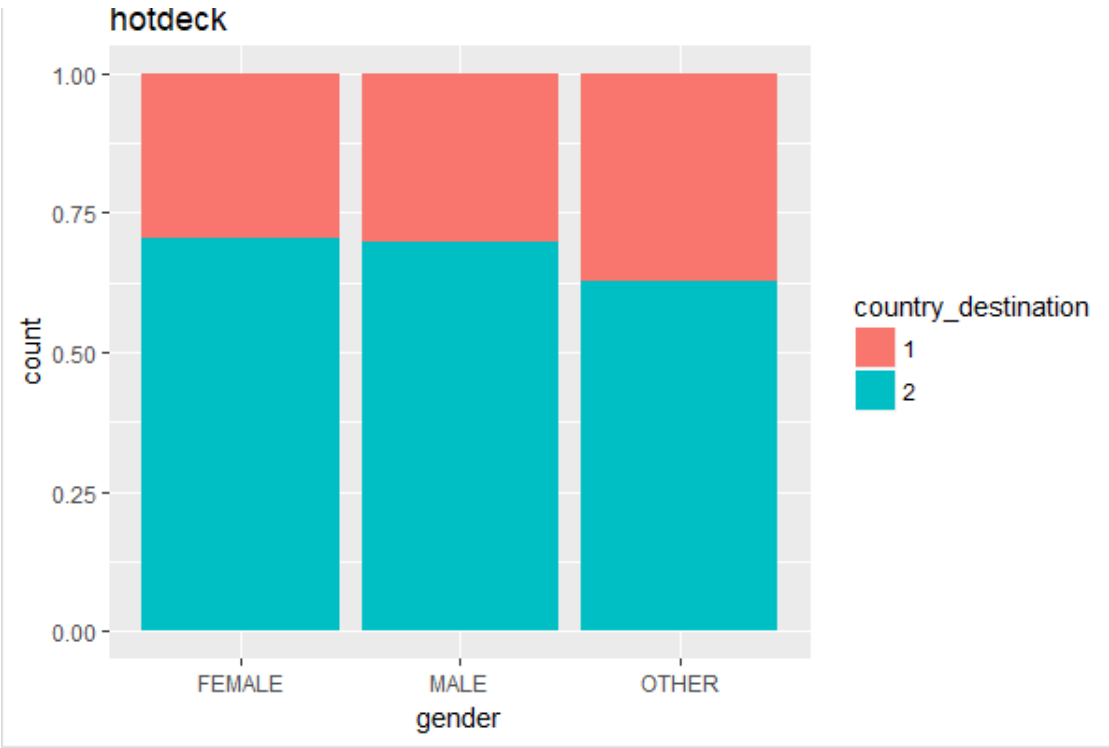




- ניתן לראות שעבור ageBins הן ללא ה-Imputation והן עם ה-Imputatation, ניתן לראות כי הגיל לא משפיע בצורה ניכרת ודי דומה בשניהם.

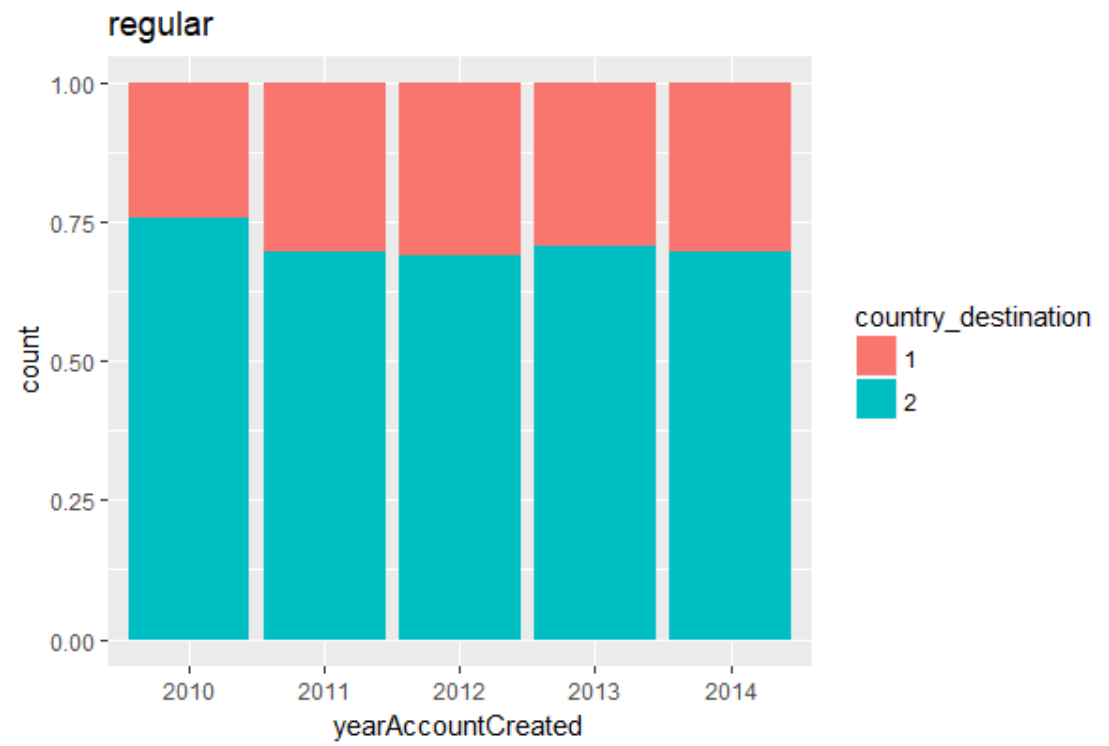
- <u>gender</u>:

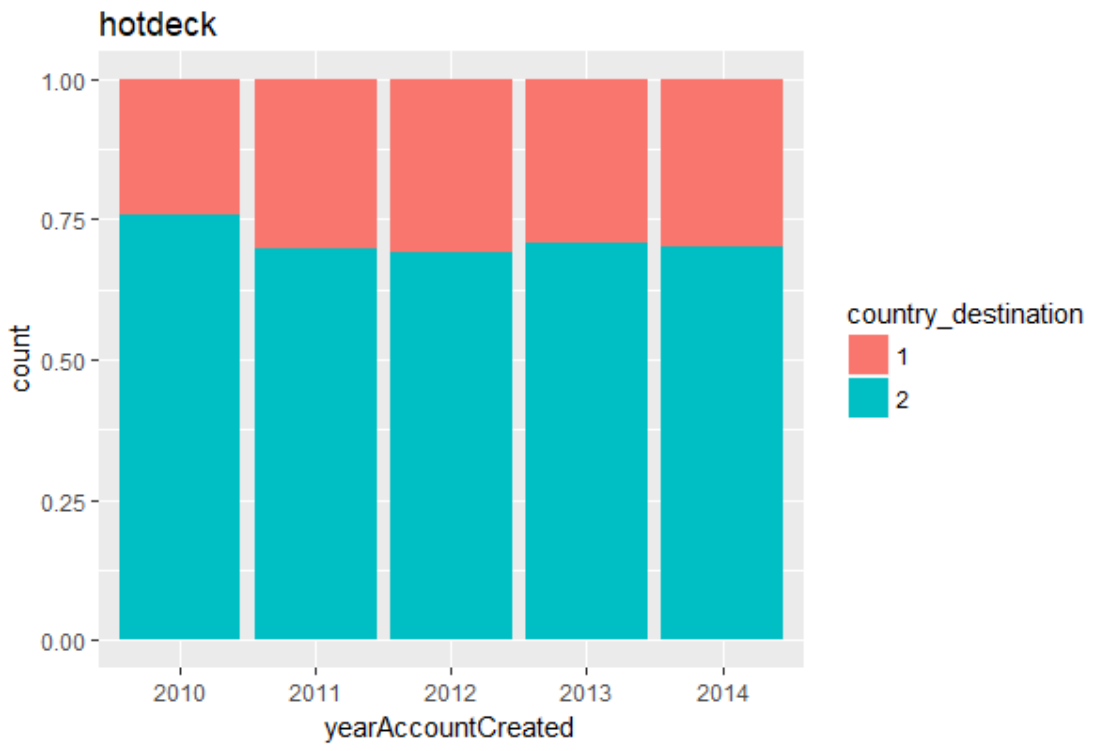




- באופן דומה לעיל ב-gender.

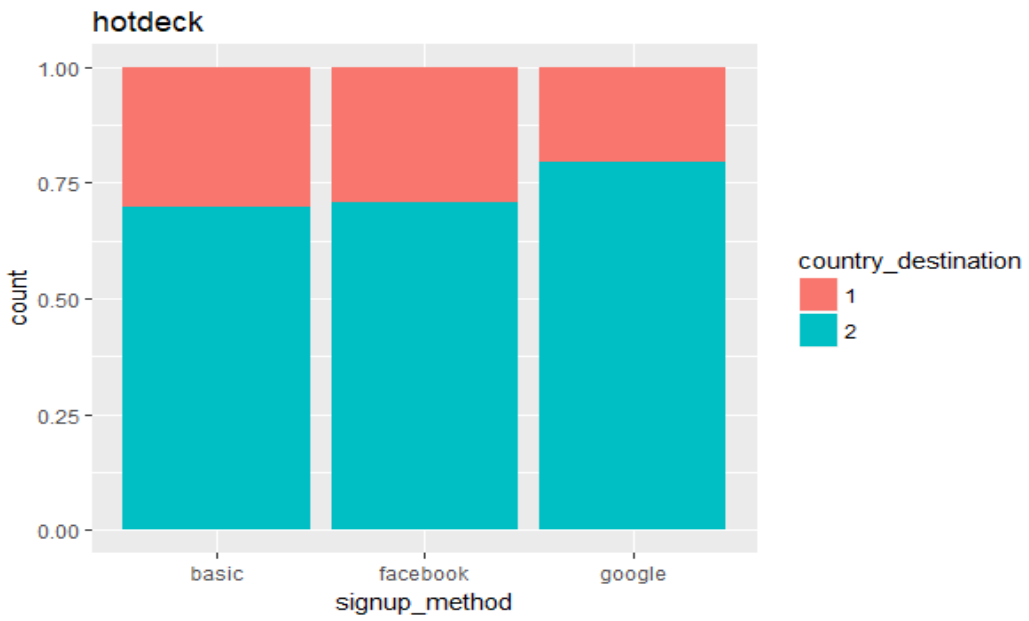
- yearAccountCreated:





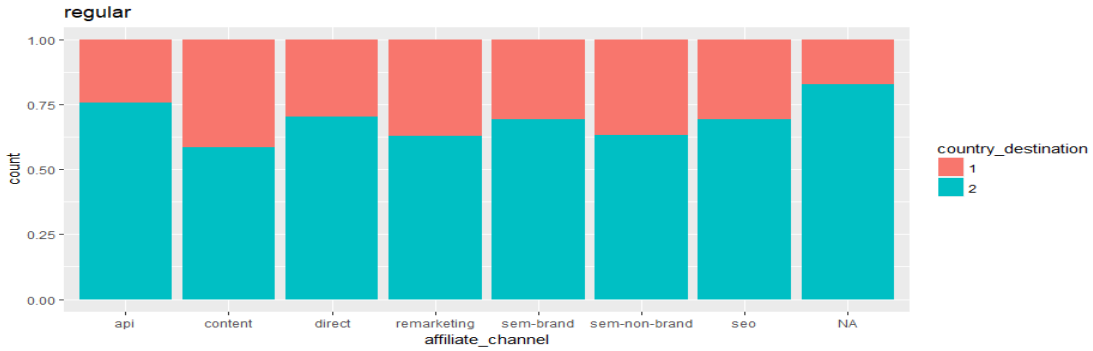
- באופן דומה לעיל ב-gender.
- תוצאות דומות נראו גם ב-weekdayAccountCreate, monthAccountCreate , yearFirstBooking, monthFirstBooking, weekdayFirstBooking, diffDaysAccountFirstActive, diffDaysBookingFirstActive.

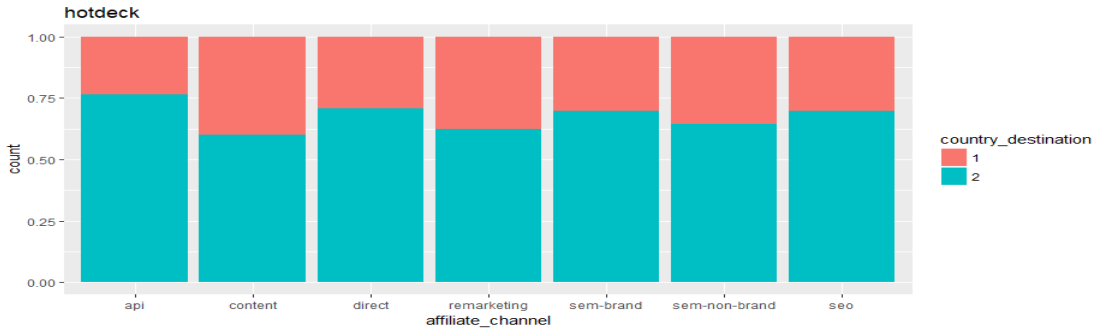
- signup_method:



- ניתן לראות שב-signup method ישנה העדפה קלה למי שמשתמש ב-google להזמין יותר בארה"ב ביחס ל-facebook או ב-basic.

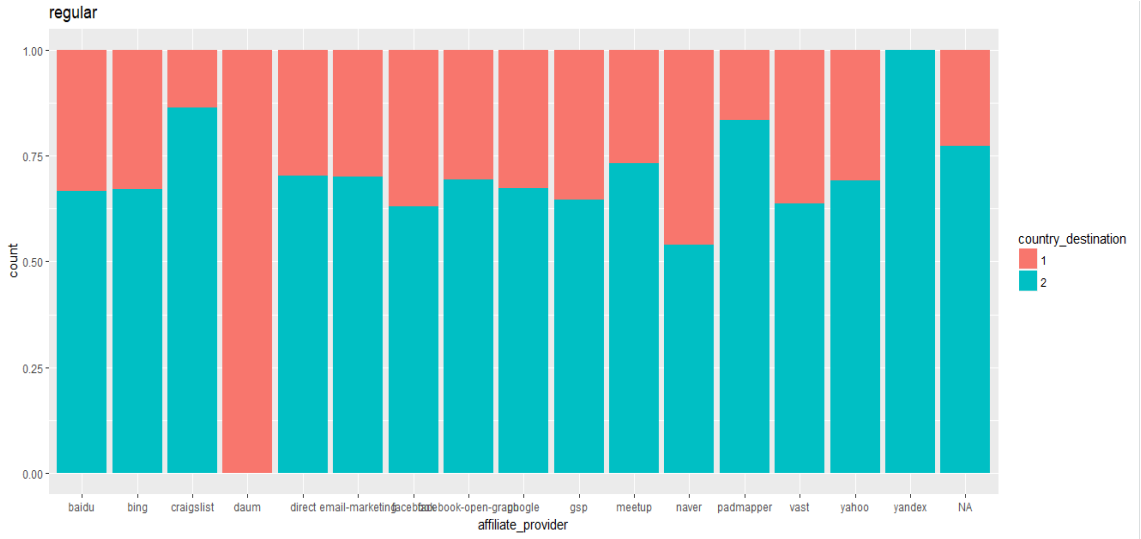
- ב-signup_app זה די דומה גם עם העדפה קלה ב- mobile על פני web להזמין יותר בארה"ב.
- ב-first_device_type זה גם פחות או יותר דומה.
- <u>Affiliate channel</u>:

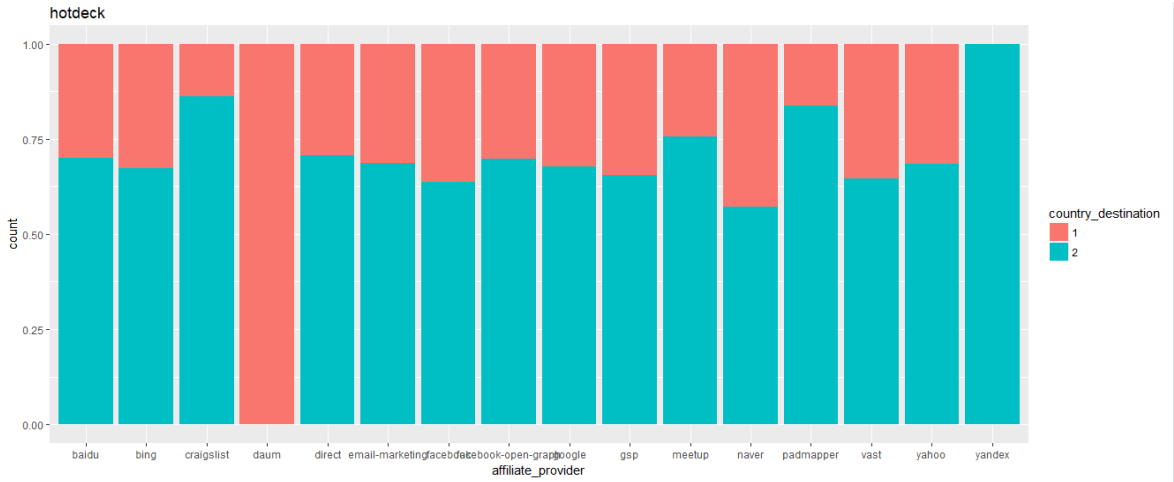




- ניתן לראות שיש העדפה לאנשים שמשתמשים ב-api להזמין יותר בארה"ב.

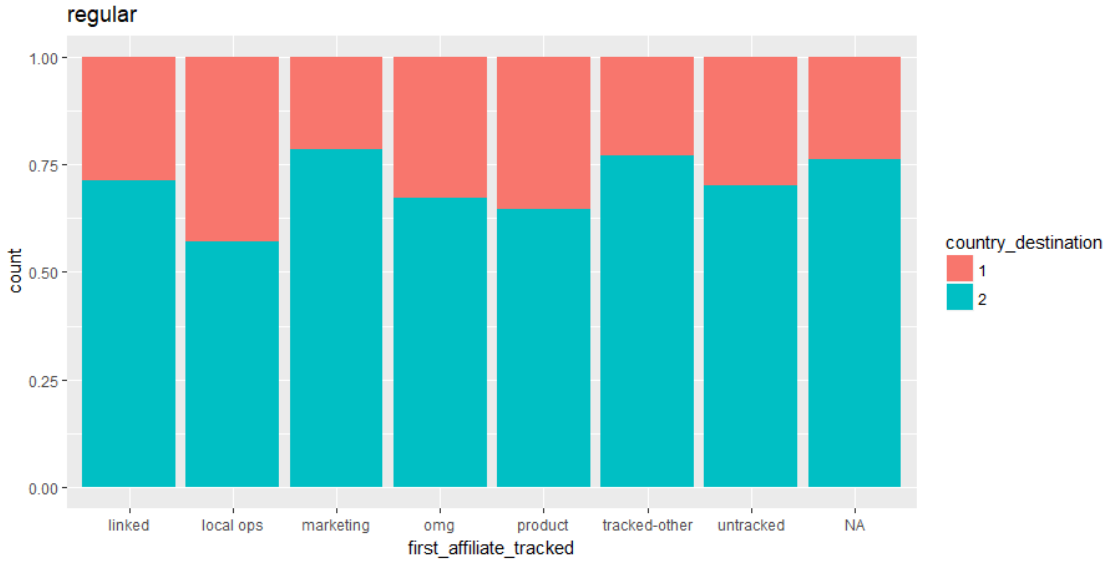
- <u>Affiliate_provider</u>:

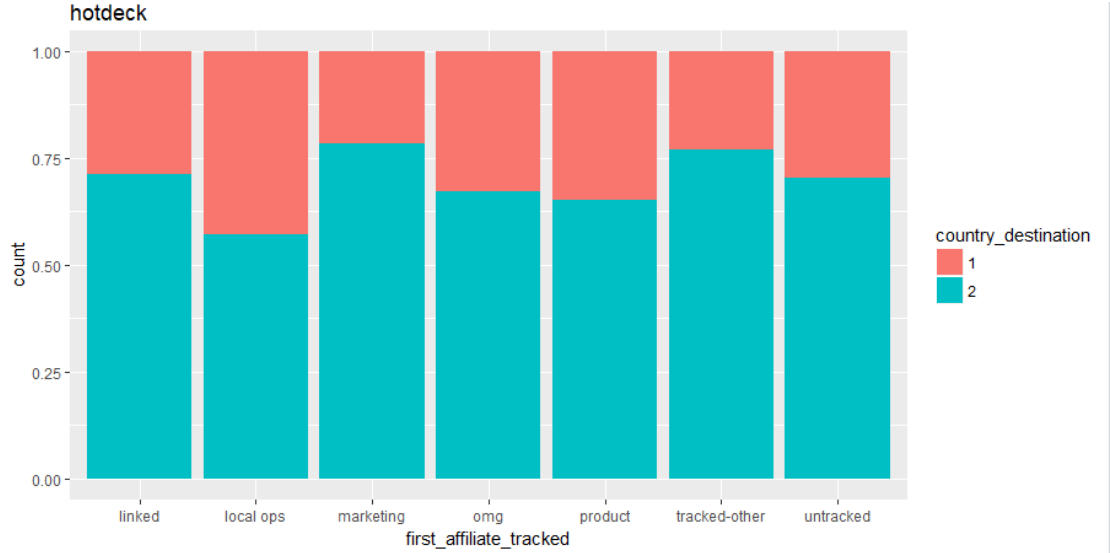




- ניתן לראות שבאופן די גורף ב-daum ישנה העדפה להזמין מחוץ לארה"ב ולמי שב-yandex ישנה העדפה להזמין בתוך משמעית ארה"ב. בנוסף גם למי שמשתמש ב-padmapper ישנה נטייה להזמין יחסית בתותך ארה"ב. מצד שני, אלו עמודות אשר יש בהן מספר בודד של נתונים לכן נוצרת עבורם הטייה (ב-daum יש נתון אחד וב-yandex 4). לעומת זאת ב- paddmapper יש כבר 200 ושם המצב קצת יותר טוב. באופן כללי ניתן לראות שב-labelHים בהם יש הרבה נתונים, המצב יחסית מאוזן.

- <u>First_affiliate_tracked</u>:





- ניתן לראות כי ב-marekting וב-tracked_other יש נטייה עבור ארה"ב כאשר ב-marketing יש יחסית פחות נתונים (כ-105 תצפיות). בסה"כ גם כאן המצב מאוזן יחסית.

- <u>First_browser</u>:

- גם כאן המצב יחסית מאוזן עבור lable-ים עם נתונים דומים (רמות דומות) ורמות לא מאוזנות עבור lable-ים עם מעט תצפיות. כנ"ל לגביי language, account_created_distance_US_toHoliday ו-first_booking_distance_US_toHoliday.

קורלציות:

- על מנת לראות קשר יותר ברור בין המשתנים האחד לשני ובין המשתנים למשתנה התלוי, נרצה לראות קורלציות בין המשתנים השונים.
- מכיוון שקיימים לנו משתנים בעלי ערכים שהם NA, נשנה אותם למשתנה "NA" במקום הערך NA ובכך יהפכו לקטגוריאלים באופן סופי ונוכל לבצע עליהם חישובים.
- שינינו ערכים אלה רק ב-dataset הרגיל כי ב-hotdeck אין כלל ערכי NA בזכות ה-imputation.
  - <u>המשתנים ששונו הם:</u>
    1. Affiliate_channel
    2. first_device_type
    3. first_affiliate_tracked
    4. affiliate_channel
    5. first_browser
    6. ageBins

<u>ב-dataset הרגיל:</u>

- מבחינת קורלציה למשתנה החזוי – לא הצלחנו למצוא (שימוש בחבילת kruskal goodman).
- אולם, לעומת זאת הצלחנו למצוא קורלציה בין פיצ'רים יחסית גבוהה:
  - בין signup_app ל-signup_flow – 0.92
  - בין first_browser ל-first_device_type – 0.71
  - בין year_account_created ל-year_first_booking – 0.77
- הקשר השלישי פחות, אולם בשניים הראשונים נתחשב במודל ובמיוחד בקשר ההדוק הראשון.

<u>ב-dataset עם ה-imputation של hotdeck:</u>

- מבחינת קורלציה למשתנה החזוי – גם כאן לא הצלחנו למצוא (שימוש בחבילת kruskal goodman).
- נמצאו קשורים דומים לעיל, אולם הפעם גם נמצא קשר בין affiliate_provider ל-affiliate_channel.

- ניתן כאמור להשתמש גם ב-Mutual Information כפי שציינו, כי שיטת הקורלציה הנ"ל לאו דווקא הכי טובה והכי מתאימה. אולם מפאת חוסר הזמן לא ביצענו זאת.
- 2 מטריצות הקורלציה מצורפות כקבצי png.

מודלי תחזיות:

- ראשית, לאור כך שבדקנו וראינו כי עבור כל תצפית שבה לא היה תאריך ב- date first booking אז בתוצאות קיבלנו 0, אזי עבור כל המודלים שנריץ מעתה והלאה, מראש נכתוב במשתנה החזוי 0 עבור תצפיות מסוג זה ואת שאר המודלים נריץ על שאר ה-data.
- מכיוון שיש להכריע בין 2 תוצאות אפשריות, אנחנו נמצאים במצב של מודלי קלאסיפיקציה אשר נרצה להשתמש בהם בשביל לחזות האם ה-user יעשה booking בארה"ב או מחוץ לארה"ב.

מודל 1: Random Forest

- המודל הראשון שבחרנו להתבסס עליו הינו **Random Forest**.
- מודל זה מבוסס על עצי החלטה כאשר במהלכו מגדלים עצים ובוחרים משתנים באופן רנדומאלי אשר מוגרלים כאשר לבסוף לוקחים את הממוצע של מספר העצים שמגדלים.

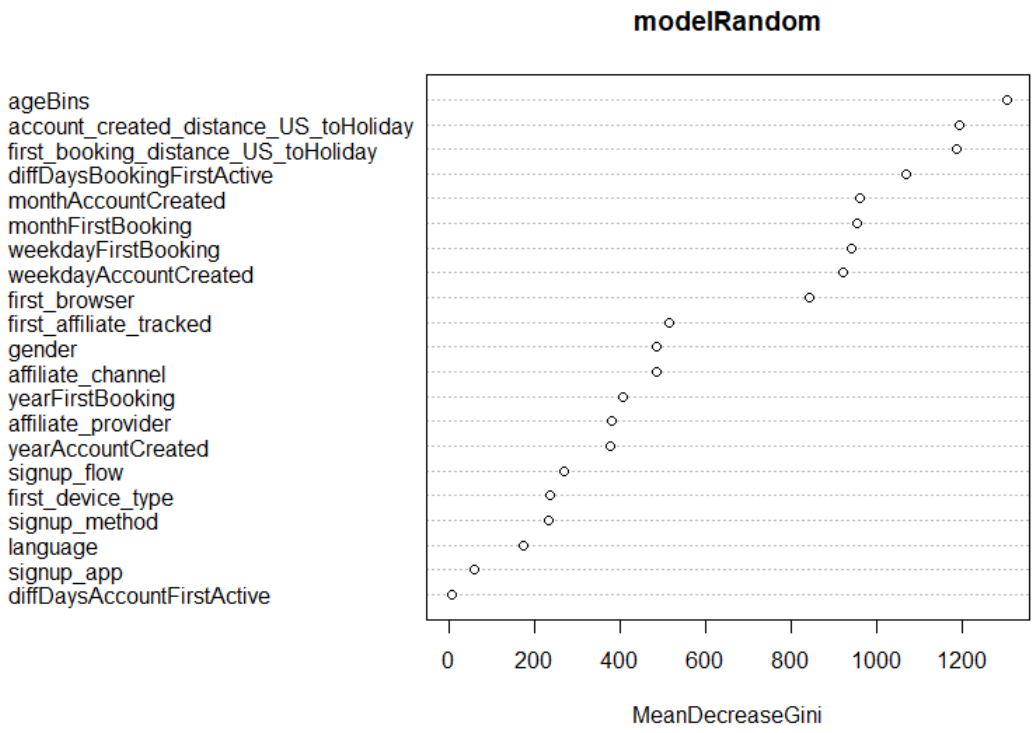
- בכל שלב מחושבת פונקציית Loss אשר האנטרופיה שלה ממוזערת מבין כל המשתנים האפשרים שנבחרים (או לחילופין ממוקסם ה-information gain).
- הפרמטרים העיקריים שצריך להזין הם הפיצ'רים של ה-Dataset שברצוננו להתבסס עליהם, מספר המשתנים הרנדומאליים שנבדקים בכל פעם וכן מספר העצים שנגדל.
- נרצה לא לקחת מעט מדי עצים בשביל שלא נקבל under fitting אולם לא נרצה יותר מדי עצים בשביל over fitting.
- על מנת לחשב את התוצאה הטובה ביותר נרצה להתייחס למדד ה-Accuracy אשר מודד את ה-(TP+TN/(TP+TN+FP+FN, כלומר מה מידת הדיוק שלנו בסה"כ עבור 2 התחזיות שלנו – גם בתוך ארה"ב וגם מחוצה לה.
- נריץ את המודל בלולאה על מנת לבחור את הפרמטרים הכי טובים עבורנו.
- כמו-כן נזכיר כי נמצא מודל טוב פעם עבור ה-hotdeck ופעם עבור הרגיל.
- ה-pool של המשתנים שהוזן כולל בתוכו את כלל המשתנים הפקטוריאלים (ageBins במקום age) ומשתנים רציפים כמו ההפרש בין הזמנים שהוצא באמצעות timestamp וכן המשתנים החדשים שנוספו החיצוניים.
- **מהרצת המודל בעל ה- Dataset הרגיל בלולאה כדי לראות מה הפרמטרים הכי טובים התקבל כי:**
  - החלוקה של המשתנים הרנדומאליים היא 2
  - מספר העצים שנבחר הוא 45

```
            actual
predictions     1     2
          1    72   117
          2  6324 14808
> #--accuracy matrix
> sum(diag(predTable_validation))/sum(predTable_validation)
[1] 0.6979034754
>
```

  - התקבל accuracy של 0.697.
  - להלן טבלת החשיבות:

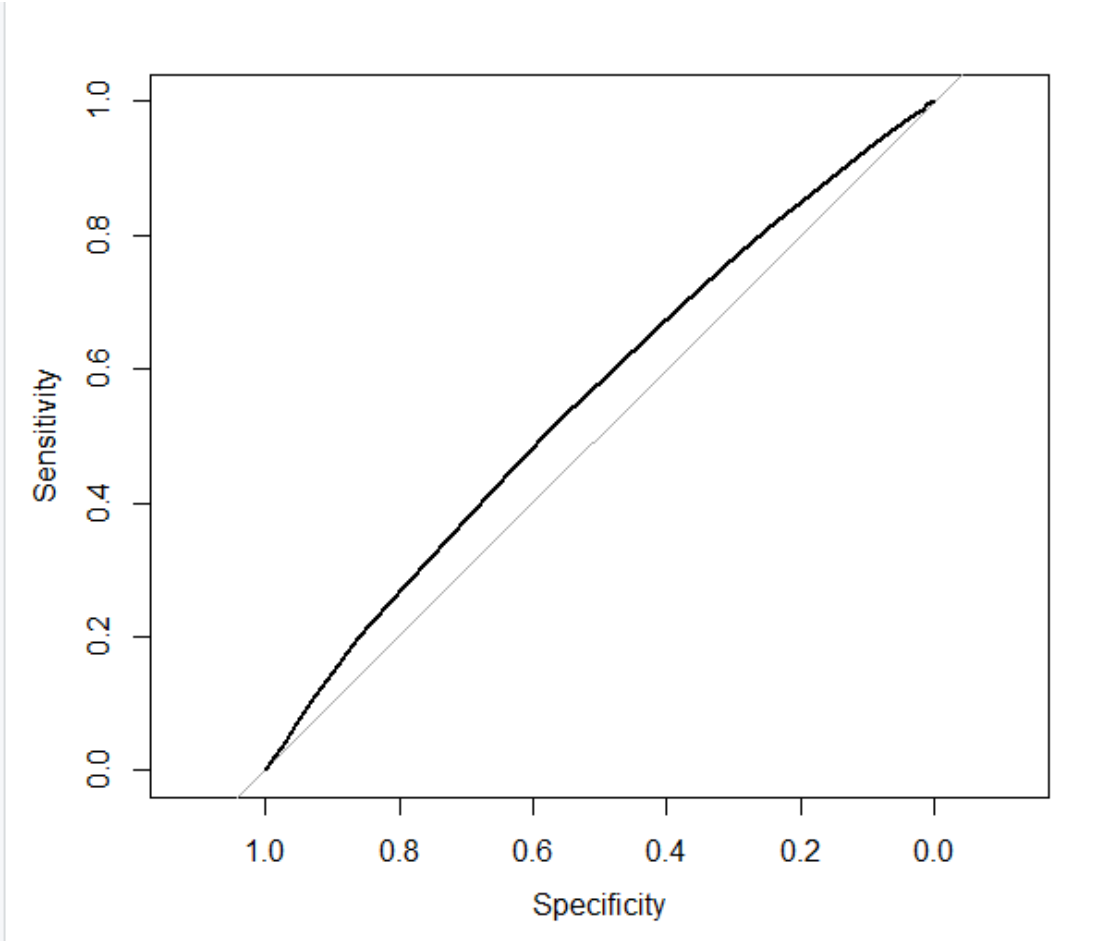
```
gender                                    485.472792350
ageBins                                  1305.800546997
signup_method                             231.465658508
signup_flow                               269.344383450
language                                  174.122819305
affiliate_channel                         484.339452534
affiliate_provider                        379.398397367
first_affiliate_tracked                   514.054662036
signup_app                                 59.068414458
first_device_type                         235.927859571
first_browser                             843.313898942
first_booking_distance_US_toHoliday      1187.028428712
account_created_distance_US_toHoliday    1193.018183727
yearAccountCreated                        376.867398236
monthAccountCreated                       961.662022363
weekdayAccountCreated                     920.975714377
yearFirstBooking                          406.861354052
monthFirstBooking                         952.613306076
weekdayFirstBooking                       940.122105221
diffDaysAccountFirstActive                  6.213277887
diffDaysBookingFirstActive               1069.112527790
>
```

- להלן המחשה לפי סדר חשיבות:



- נשים לב כי הגורמים המשפיע ביותר הם **המרחקים מהחגים** של תאריך ה-Booking וה-ageBins.

- מבחינת AUC, להלן התוצאות:



- מודל זה נבחר ביחס לשאר המודלים אחרי הרצות רבות והשוואות מדד ה- accuracy ומדד ה-AUC בין העצים השונים.

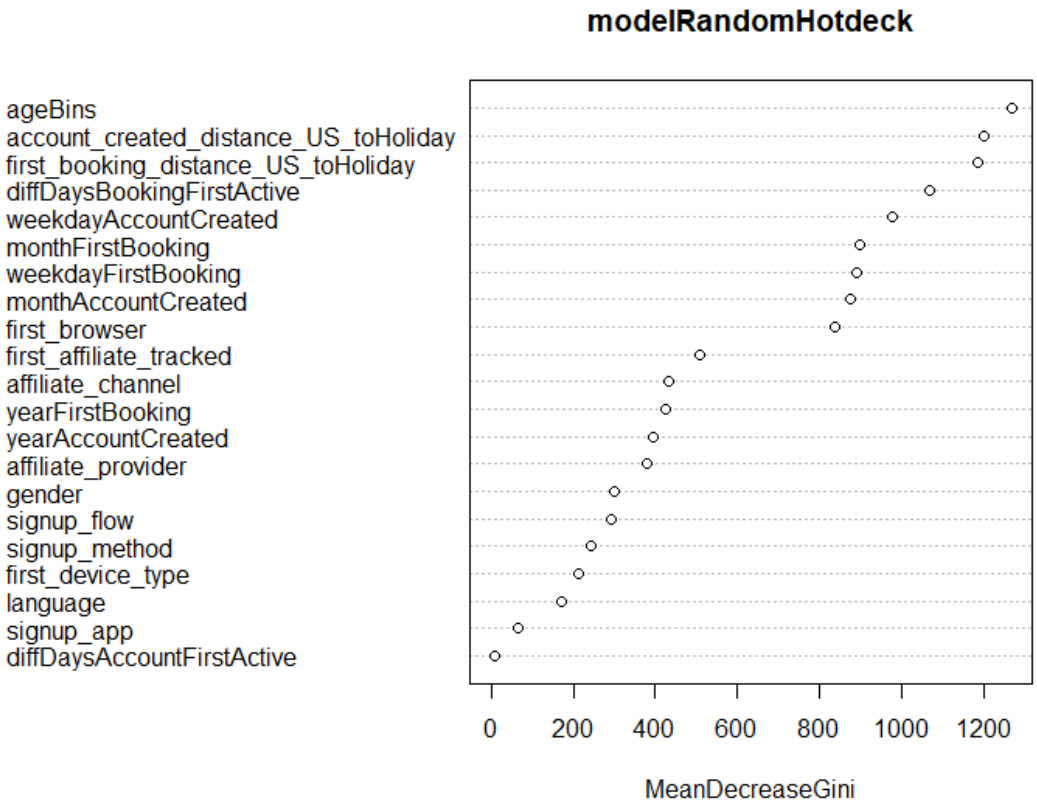
- **מהרצת המודל בעל ה- Dataset ה-hotdeck בלולאה כדי לראות מה הפרמטרים הכי טובים התקבל כי:**
  - מספר המשתנים הרנדומאליים הוא 2
  - מספר העצים הוא 45

```
           actual
predictions     1     2
          1    52    92
          2  6344 14833
> #--accuracy matrix
> sum(diag(predTableHotdeck_validation))/sum(predTableHotdeck_validation);
[1] 0.698137986
>
```

  - התקבל accuracy של 0.698.
  - להלן טבלת החשיבות:

```
                                        MeanDecreaseGini
gender                                       301.27573883
ageBins                                     1267.95840440
signup_method                                243.50859586
signup_flow                                  291.56290478
language                                     170.09677601
affiliate_channel                            430.81493768
affiliate_provider                           378.75821817
first_affiliate_tracked                      506.61069748
signup_app                                    63.44324733
first_device_type                            212.66747156
first_browser                                836.20956747
first_booking_distance_US_toHoliday         1186.17670644
account_created_distance_US_toHoliday       1201.86597180
yearAccountCreated                           394.25805157
monthAccountCreated                          876.18629786
weekdayAccountCreated                        976.70739861
yearFirstBooking                             425.29000338
monthFirstBooking                            899.01472918
weekdayFirstBooking                          890.76588372
diffDaysAccountFirstActive                     6.58743703
diffDaysBookingFirstActive                  1067.11325401
>
```

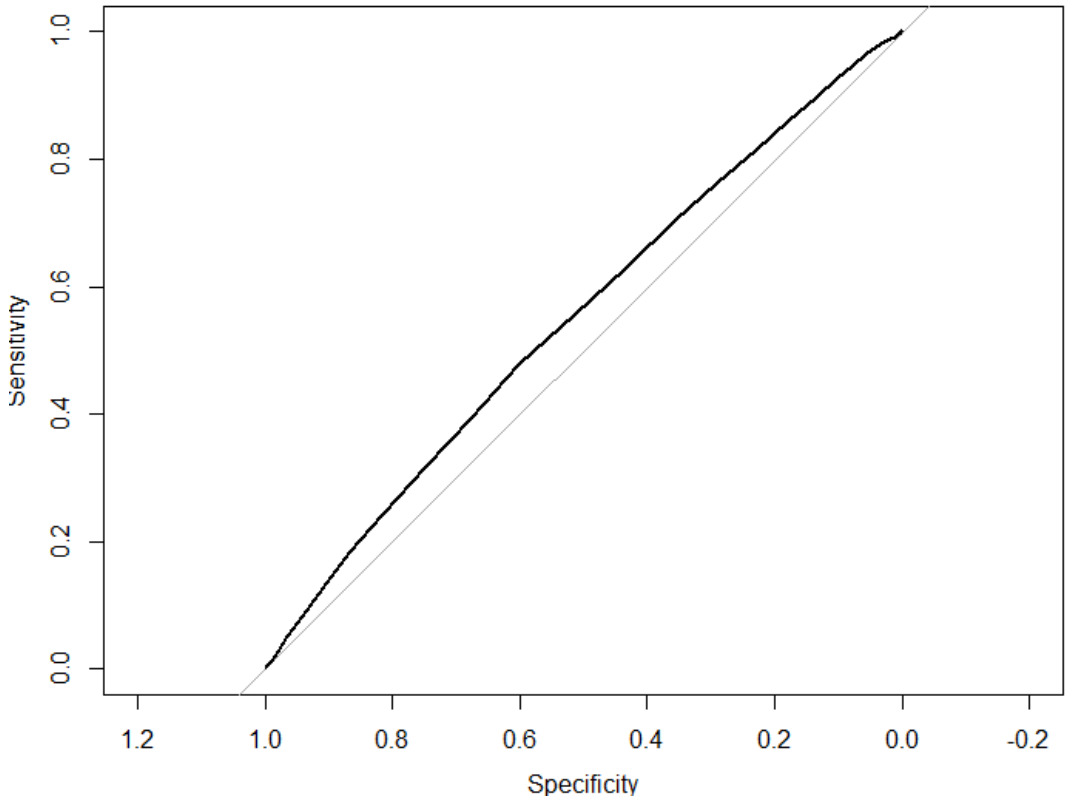
- להלן המחשה לפי סדר חשיבות:



- נשים לב כי הגורמים המשפיע ביותר הם **המרחקים מהחגים** של תאריך ה-Booking וכן ה-ageBins, בדומה למקודם.

-

- מבחינת AUC, להלן התוצאות:



- נשים לב בסה"כ שמכיוון שזהו מודל רנדומאלי התוצאות יוצאות שונות בכל פעם, אך לב הן יוצאות בסביבות accuracy של 0.69.

- יש לציין כי נסיונות להוריד משתנים שביניהם הייתה קורלציה גבוהה הורידו יחסית הרבה את התוצאות – לכן השתמשנו בכל המשתנים האפשריים במודל.

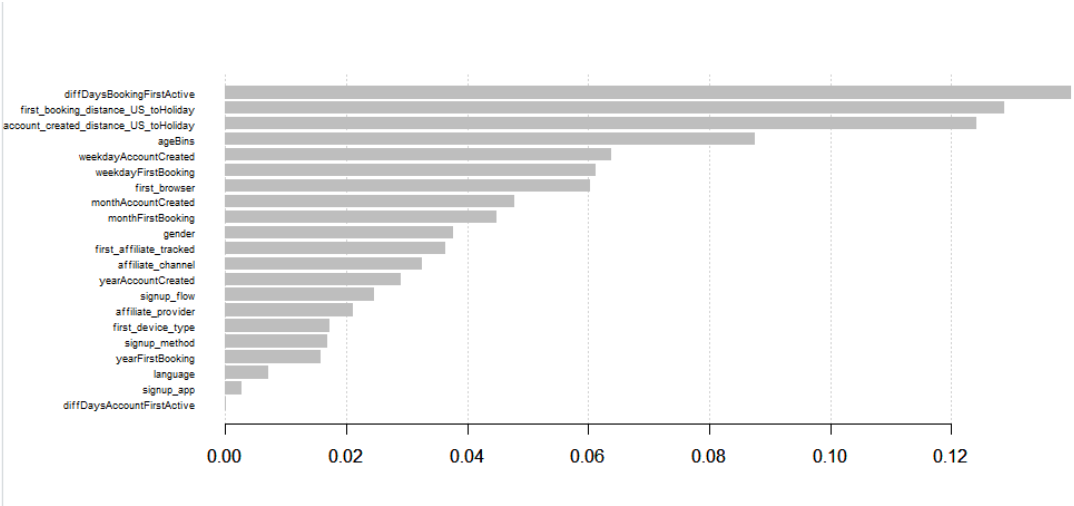
מודל 2: XGBoost

- המודל השני שבחרנו להתבסס עליו הינו XGBoost - eXtreme Gradient Boosting.
- זהו מודל נוסף המבוסס עצי החלטה שעוזר לקלאסיפיקציה ורגרסיה באמצעות שימוש בגרדיאנטים.
- לקריאה נוספת אודותיו: https://xgboost.readthedocs.io/en/latest/model.html
- במודל שלנו אנו נעשה שימוש באלגוריתם זה עפ"י שיטת רגרסיה לוגיסטית על מנת שיתמוך בקלאסיפיקציה בינארית.
- הפרמטרים אותם מזינים הם:
  - Data – המידע שאנחנו רוצים לפעול עליו
  - Label – המשתנה אותו נרצה לחזות
  - Max.depth – עומק העץ

- Nthread – מספר ה-thread-ים במחשב שנשתמש בהם.
- Nround – מספר הפעמים שנעבור על ה-data.

- ראשית, לצורך שימוש בחבילה נרצה להפוך את ה-train שלנו למטריצה.
- את הבדיקות עשינו על הרבה פרמטרים שונים ולבסוף הגענו למספר של 55 במספר החזרות ועומק עץ של 10.
- גם במקרה זה השתמשנו במדד ה-Accuracy, במדד err של שגיאה וב-AUC.
- השגיאה שהתקבלה הינה: 0.3739036631.
- להלן מטריצת החשיבות:

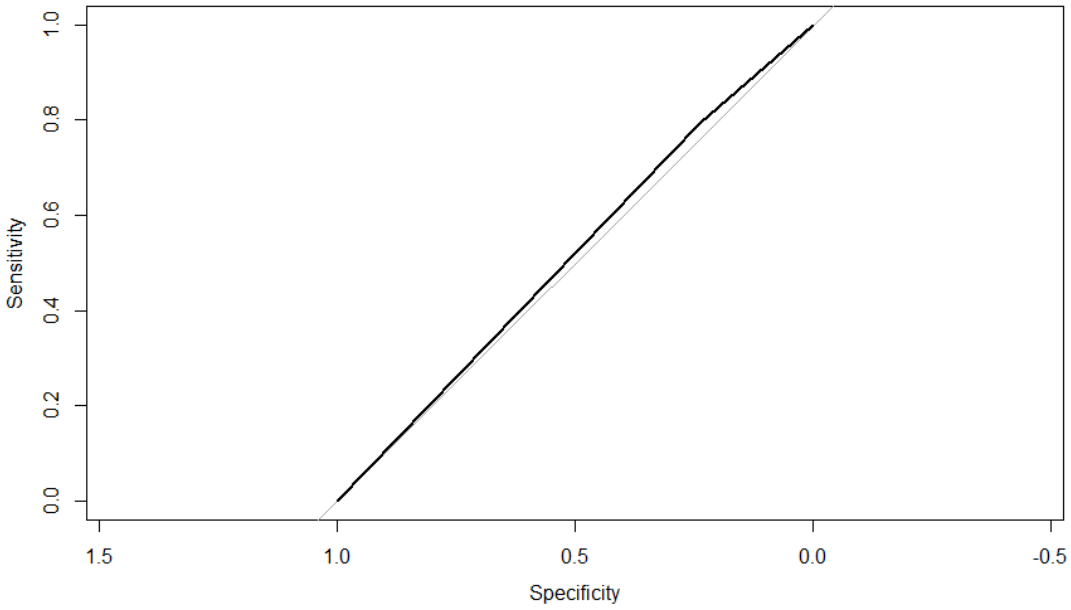
	Feature	Gain	Cover	Frequency
1:	diffDaysBookingFirstActive	0.1398400748207	0.159003564556	0.1302225797801
2:	first_booking_distance_US_toHoliday	0.1287087236642	0.130034357511	0.1312416197372
3:	account_created_distance_US_toHoliday	0.1240877662204	0.126550831543	0.1236792705819
4:	ageBins	0.0874795058383	0.069674784002	0.0893537141325
5:	weekdayAccountCreated	0.0637965432048	0.044440682223	0.0665057656208
6:	weekdayFirstBooking	0.0612799421977	0.043078157432	0.0657012603915
7:	first_browser	0.0603764036771	0.049099752508	0.0628050415661
8:	monthAccountCreated	0.0478214386370	0.049925631401	0.0470903727541
9:	monthFirstBooking	0.0447903940442	0.050625331489	0.0425315097881
10:	gender	0.0377613562390	0.026515259919	0.0408688656476
11:	first_affiliate_tracked	0.0363446110582	0.035073014144	0.0387235183695
12:	affiliate_channel	0.0326019980238	0.043529679955	0.0292303566640
13:	yearAccountCreated	0.0291300117664	0.024546278562	0.0299275945294
14:	signup_flow	0.0246204477989	0.021628741769	0.0236524537409
15:	affiliate_provider	0.0211640741473	0.027082773066	0.0206489675516
16:	first_device_type	0.0173592882480	0.013370020910	0.0163582729954
17:	signup_method	0.0168388688009	0.016140191890	0.0178063824082
18:	yearFirstBooking	0.0157897330062	0.020607511305	0.0156610351301
19:	language	0.0072219540328	0.039197339198	0.0061142397426
20:	signup_app	0.0027952062560	0.006933694331	0.0017162778225
21:	diffDaysAccountFirstActive	0.0001916583181	0.002942402287	0.0001609010459
	Feature	Gain	Cover	Frequency



- ניתן לראות שהמדדים המובילים, בדומה ל random forest הם המרחקים מהחגים ו-ageBins, אולם כאן גם הפרש מספר הימים מהבוקינג הראשון גם היה משמעותי. כמו-כן ניתן לשים לב ששוב **בדומה** ל-random forest, הפיצ'רים של של מספר הימים מאז הפעם הראשונה שה- user היה פעיל, ה-signup_app וה-language הם הכי פחות משמעותיים.
- מבחינת ה-accuracy, הושג כזה של 0.63. פחות טוב משל ה-random forest:

```
> sum(diag(regularPredictionTable))/sum(regularPred
ctionTable);
[1] 0.6384315933
```

- כמו-כן, גם ה-AUC שהתקבל פחות טוב וקיבלנו בסביבות ה-0.5:



- תוצאות דומות התקבלו גם עבור ה-**hotdeck**.

מודל 3: רגרסיה לוגיסטית ומולטינומית

- הרציונאל לבצע אותן הוא עקב קלאסיפיקציה כאשר במולטינומית השתמשנו על כל ה-data (בלי להוריד את האפסים הברורים) או רגרסיה לוגיסטית לאחר הורדת האפסים.
- בגלל שהמשתנה התלוי שלנו הוא קטגוריאלי, רגרסיה לינארית לא אפשרית. לכן בחרנו ברגרסיה לוגיסטית אשר מתאימה למשתנה תלוי בסולם קטגוריאלי.
- במקרה שלנו למשנה התלוי הקטגוריאלי קיימות 3 רמות, ולכן נשתמש בספריית nnet כדי ליצור רגרסיה לוגיסטית multinom.
- בהתאם לקורלציות שראינו מעלה, בחרנו להשתמש במשתנים מסבירים בעלי קשר למשתנה התלוי:
- country_destination=gender+first_device_type+signup_flow+first_booking_distance_US_toHoliday+yearFirstBooking+monthFirstBooking+diffDaysBookingFirstActive

```
> summary(multi_model)
Call:
multinom(formula = country_destination ~ gender + first_device_type +
    signup_flow + first_booking_distance_US_toHoliday + yearFirstBooking +
    monthFirstBooking + diffDaysBookingFirstActive, data = data)

Coefficients:
  (Intercept) genderFEMALE genderMALE genderOTHER first_device_typeDesktop first_device_typeSmartPhone first_device_typeTablet
1  -100.66152     1.464468   9.198120   -7.542778                -7.999709                   -6.447330               -9.035634
2   -96.70176     1.562841   9.245489   -7.758832                -8.218808                   -6.533697               -9.383242
  signup_flow first_booking_distance_US_toHoliday yearFirstBooking2010 yearFirstBooking2011 yearFirstBooking2012
1 -0.10624690                          -0.7152628             150.8307             186.4579             155.1044
2 -0.09877686                          -0.7158963             149.0954             184.4319             153.0694
  yearFirstBooking2013 yearFirstBooking2014 yearFirstBooking2015 monthFirstBooking1 monthFirstBooking2 monthFirstBooking3
1             125.7420             121.0476             132.1001           115.2018           104.0566           75.86018
2             123.7749             119.1180             130.4508           114.3709           103.0690           74.85729
  monthFirstBooking4 monthFirstBooking5 monthFirstBooking6 monthFirstBooking7 monthFirstBooking8 monthFirstBooking9
1           53.27831           27.98161           32.85181           49.78929           35.02274           129.9173
2           52.19353           26.79713           31.72969           48.83299           34.08012           129.0178
  monthFirstBooking10 monthFirstBooking11 monthFirstBooking12 diffDaysBookingFirstActive
1            39.21366            81.14589            126.9634                 0.01423956
2            38.41745            80.30354            126.2710                 0.01415441

Std. Errors:
  (Intercept) genderFEMALE genderMALE genderOTHER first_device_typeDesktop first_device_typeSmartPhone first_device_typeTablet
1  0.02175368    0.0100797  0.0103709  0.08874690               0.02615983                  0.03111097              0.02937626
2  0.02175367    0.0100797  0.0103709  0.08874714               0.02615984                  0.03111097              0.02937626
   signup_flow first_booking_distance_US_toHoliday yearFirstBooking2010 yearFirstBooking2011 yearFirstBooking2012
1 0.0008485953                         0.0002737735           0.03014834           0.01599249           0.01226484
2 0.0008485799                         0.0002738505           0.03014834           0.01599249           0.01226484
  yearFirstBooking2013 yearFirstBooking2014 yearFirstBooking2015 monthFirstBooking1 monthFirstBooking2 monthFirstBooking3
1           0.01094744           0.01094925            0.0315219         0.01564489         0.01519092          0.0184022
2           0.01094744           0.01094925            0.0315219         0.01564489         0.01519092          0.0184022
  monthFirstBooking4 monthFirstBooking5 monthFirstBooking6 monthFirstBooking7 monthFirstBooking8 monthFirstBooking9
1         0.01346622         0.01184874          0.0118803         0.01462388         0.01447459         0.01488768
2         0.01346622         0.01184874          0.0118803         0.01462388         0.01447459         0.01488768
  monthFirstBooking10 monthFirstBooking11 monthFirstBooking12 diffDaysBookingFirstActive
1          0.01599413          0.01722991            0.017969               5.404183e-05
2          0.01599413          0.01722991            0.017969               5.380200e-05
```

- עקב תוצאות פחות טובות בבירור אל מול שאר המודלים הוחלט שלא להשתמש במודלים אלו ועל כן לא נרחיב בנושא.

מודל סופי:

- ה- **random forest** הניב את התוצאות הטובות – קרוב ל-0.7 אחוז ב-validation (בלי לכלול את התוצאות שהן booking=0) אשר תוצאות ודאיות ובאופן יחסית יעלו מאוד את האחוז דיוק של כל ה-data.
- מכיוון שהנתונים של booking היוו כ-**58% מה-data** – אזי בסה"כ הגענו בתוצאה הסופית **לאחוז דיוק באמצעות Random Forest של 87.35.**
- יש לשים לב כי אחוז זה עלול להשתנות בהתאם לתוצאות ה-random forest, אך הדבר נע באזור האחוזים הללו.
- מצורפים **2 קבצים עם תוצאות** (עבור כל dataset):
  - airbnb_test_final.csv
  - airbnb_test_hotdeck_final.csv