Data Analysis

Homework 5.

—

Ali Izadi

810199102

بخش اول

تمیز کردن داده

۱- همان طور که در زیر مشاهده میشود تنها ستون ChannelID دارای مقدار null است که چون در تحلیل های سوالات به غیر از قسمت آخر به این ستون نیاز نداریم سطرهای بدون channelID را ابتدا در داده نگه میداریم:

```
transaction.isna().sum()
```
[3] ✓ 0.3s

```
UserID          0
Date            0
Time            0
Paid Amount     0
ChannelID     204
dtype: int64
```

۲- ابتدا تاریخ شمسی داده شده را parse کرده و با استفاده از کتابخانه jdatetime آن را به فرمت قابل تبدیل به datetime میکنیم و در ستون Date ذخیره میکنیم.

سپس زمان دقیق تراکنش را parse کرده و نیز به فرمت timedelta تبدیل کرده و با ستون date جمع میکنیم تا ستون نهایی DateTime اطلاعات مربوط به تاریخ و زمان دقیق هر تراکنش را در خود داشته باشد و ستون های Date و Time را حذف میکنیم. پیاده سازی مراحل بالا نیز مطابق با زیر انجام شده است.

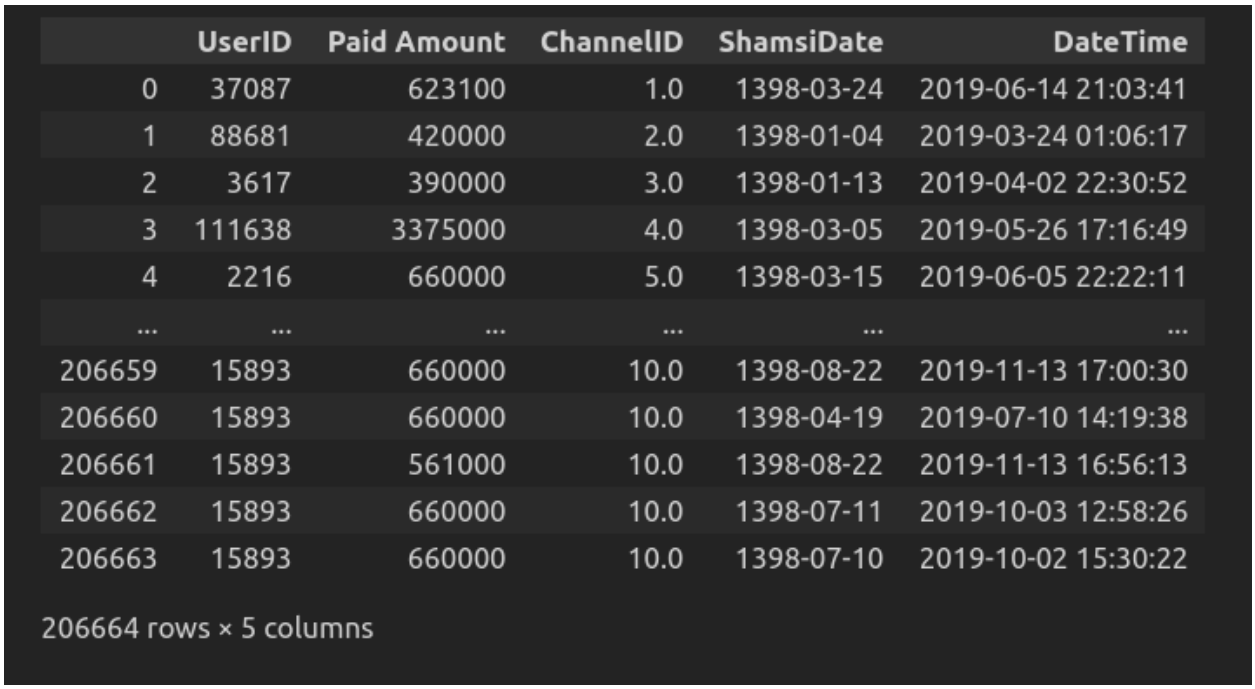
```
import jdatetime
import datetime

transaction['ShamsiDate'] = transaction['Date'].apply(lambda x: jdatetime.date(int(str(x)[:4]), int(str(x)[4:6]), int(str(x)[6:])))
# convert to jalali
transaction['Date'] = transaction['ShamsiDate'].apply(lambda x: jdatetime.date.togregorian(x))
# convert date to datetime
transaction['Date'] = transaction['Date'].apply(lambda x: datetime.datetime.strptime(str(x), '%Y-%m-%d'))

transaction['Time'] = transaction['Time'].apply(lambda x: str(x).zfill(6))
transaction['DateTime'] = transaction['Date'] + transaction['Time']\
.apply(lambda x: datetime.timedelta(hours=int(str(x)[:2]), minutes=int(str(x)[2:4]), seconds=int(str(x)[4:])))

del transaction['Date']
del transaction['Time']
```

نتیجه داده ی پیش پردازش شده در زیر مشاهده میشود:



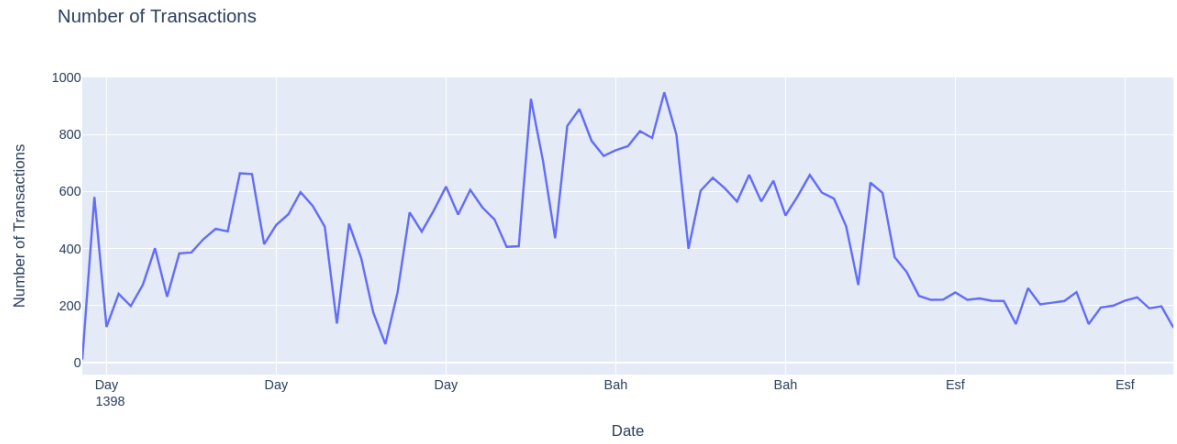
	UserID	Paid Amount	ChannelID	ShamsiDate	DateTime
0	37087	623100	1.0	1398-03-24	2019-06-14 21:03:41
1	88681	420000	2.0	1398-01-04	2019-03-24 01:06:17
2	3617	390000	3.0	1398-01-13	2019-04-02 22:30:52
3	111638	3375000	4.0	1398-03-05	2019-05-26 17:16:49
4	2216	660000	5.0	1398-03-15	2019-06-05 22:22:11
...	...	...	...	...	...
206659	15893	660000	10.0	1398-08-22	2019-11-13 17:00:30
206660	15893	660000	10.0	1398-04-19	2019-07-10 14:19:38
206661	15893	561000	10.0	1398-08-22	2019-11-13 16:56:13
206662	15893	660000	10.0	1398-07-11	2019-10-03 12:58:26
206663	15893	660000	10.0	1398-07-10	2019-10-02 15:30:22

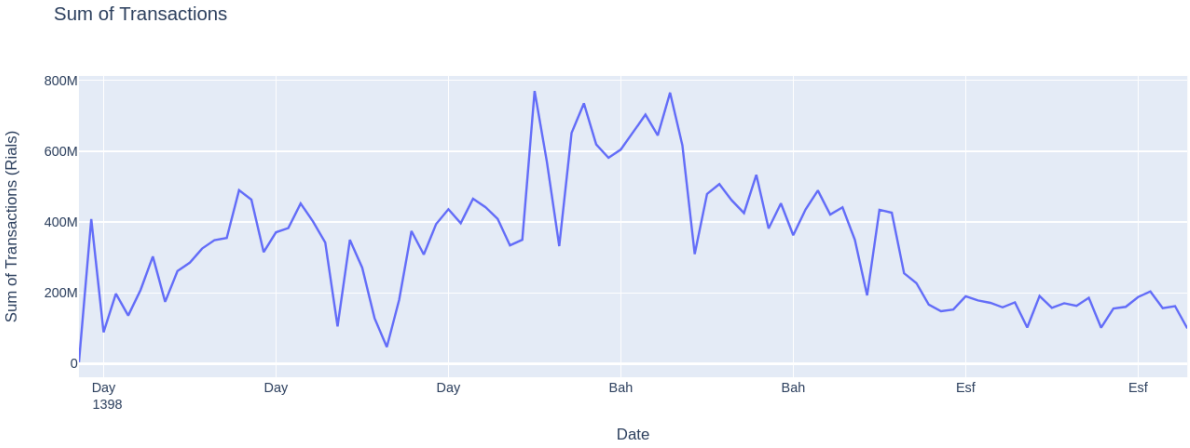
206664 rows × 5 columns

متریک ها

زمان ها بر اساس تاریخ تبدیل شده شمسی به GMT پردازش شده تا بتوان از کتابخانه های datetime و pandas برای پردازش استفاده کرد و در نهایت برای مشاهده از کتابخانه plotly برای نمایش تاریخ شمسی استفاده شده است.

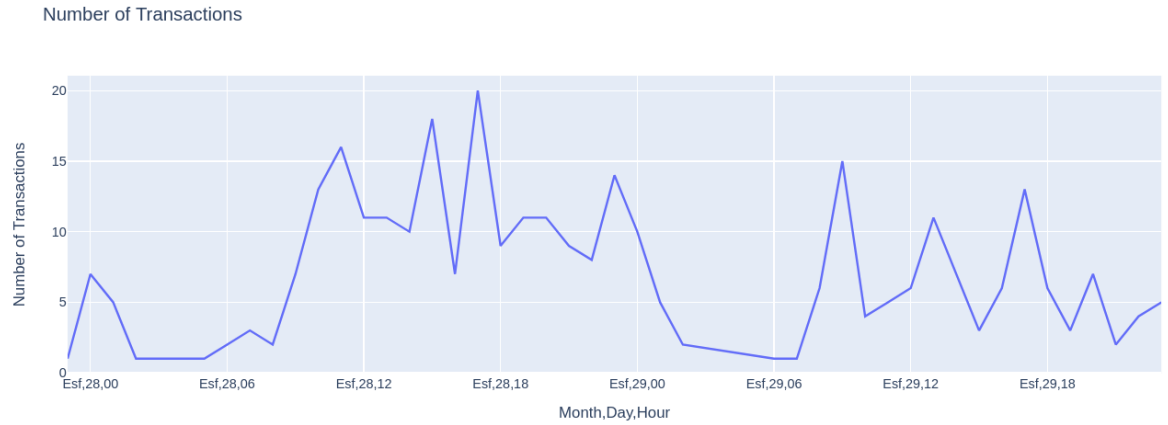
۱- تعداد و ارزش تراکنش های روزانه برای ۳ ماه آخر

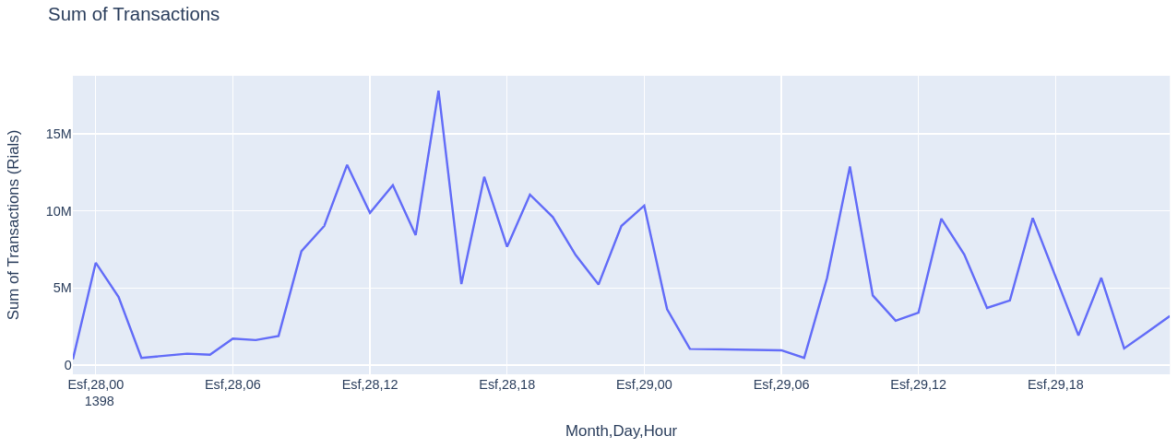




- همان طور که مشاهده میشود شباهت زیادی بین تغییرات تعداد و ارزش تراکنش وجود دارد که منطقی است زیرا احتمالا اکثر تراکنش ها توزیع مقادیر کم تراکنش را دارند و جمع مقدار آن ها در روز را تعداد آن ها مشخص میکند.
- در ماه بهمن تراکنش افزایش یافته است ولی در ماه اسفند کاهش پیدا کرده و از ماه دی نیز کمتر شده است.
- کاهش های تراکنش در هر ماه نیز مشاهده میشود که مربوط به روزهای تعطیل است.

۲- تعداد و ارزش تراکنش ساعتی برای ۴۸ ساعت اخیر

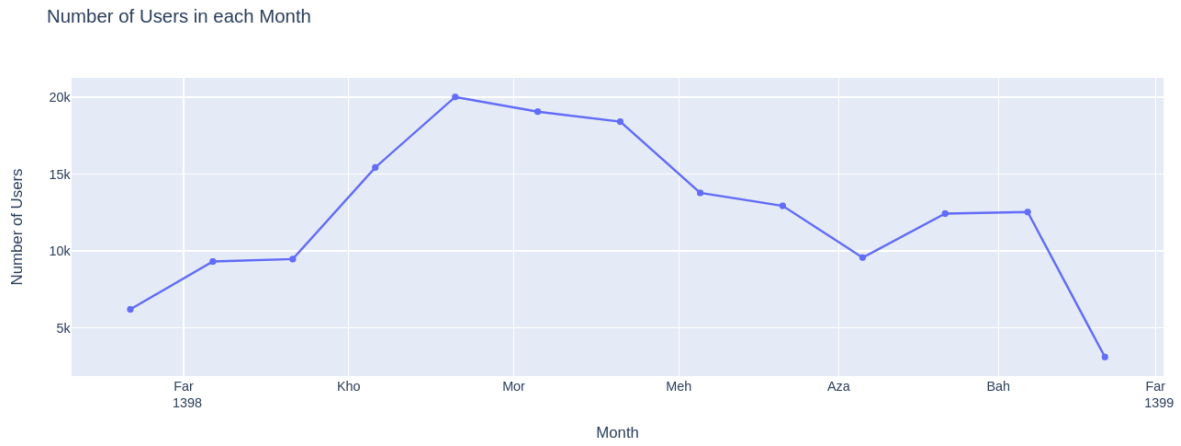




- نیمه شب تراکنش به کمترین میزان خود رسیده است.
- روز گذشته ساعت ۱۷ در مقایسه با بقیه ساعات پر تراکنش و امروز افزایش چشم گیری در تراکنش داشته است.
- به طور کلی روز قبل تراکنش بیشتری نسبت به امروز داشته است.

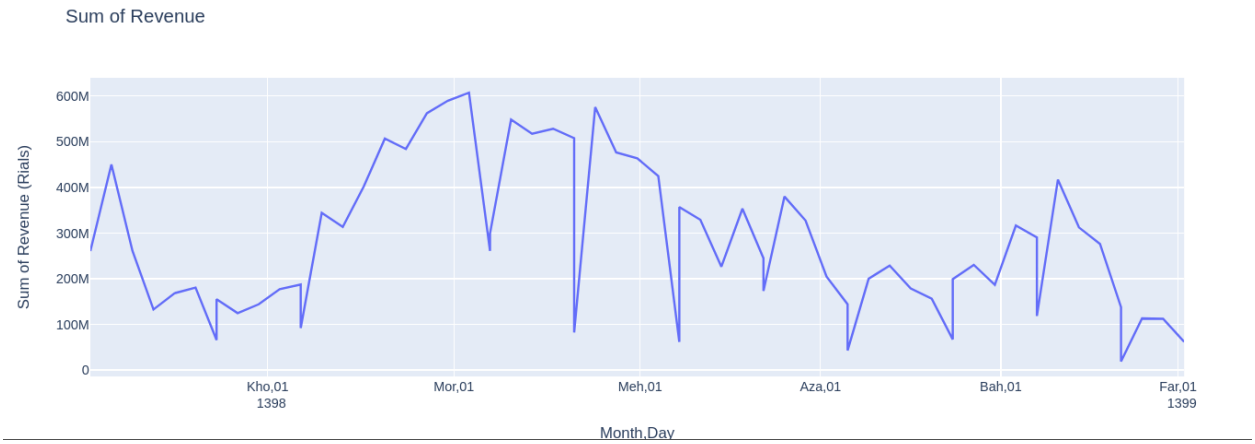
۳- تعداد مشتریان ماهانه

ماه های هر سال به صورت جداگانه برای هر سال محاسبه و رسم شده است.



- همان طور که مشاهده میشود در فصل تابستان تعداد مشتریان نسبت به زمستان بیشتر بوده است.
- بیشترین میزان مشتریان در ماه تیر بوده است.
- ماه های آذر، اردیبهشت و فروردین جز ماه های کم مشتری بوده اند.

۴- در آمد هفتگی بر اساس ۱۰ درصد کارمزد از هر تراکنش



- میزان سود چون بر اساس میزات تراکنش هاست رابطه مشابه فصلی با تعداد مشتریان و تراکنش ها دارد که در زمستان کمتر از تابستان است.
- نوساناتی در هفته های پشت سر هم وجود دارد و علت میتواند تعداد تعطیلی در روزهای آن هفته باشد که میزان تراکنش را پایین آورده باشد.

۵- retention rate ماهیانه

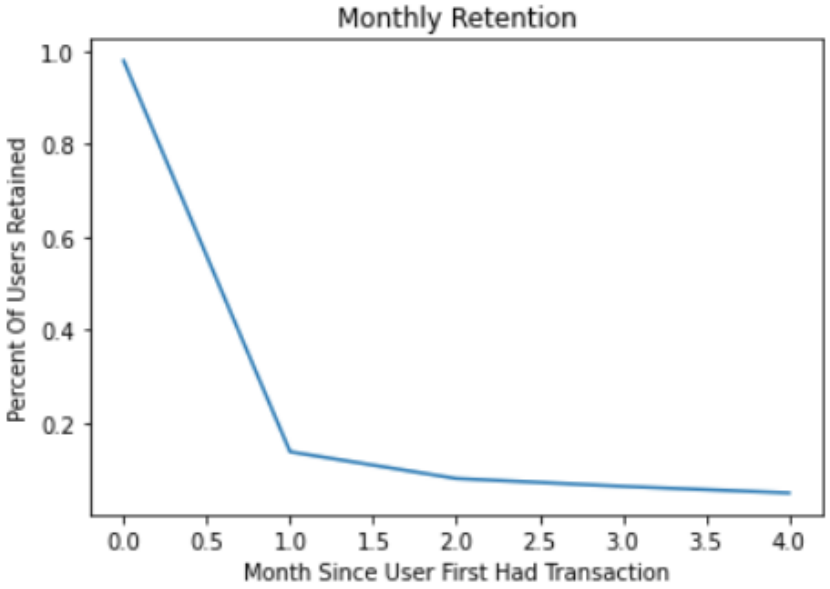
برای محاسبه نرخ ماندگاری ابتدا اولین تراکنش هر کاربر استخراج شده است.

سپس یک ستون به داده های اصلی اضافه میشود که مشخص کننده تاریخ اولین تراکنش هر کاربر است.

سپس با استفاده از این ستون جدید تعداد ماه های بین اولین تراکنش و هر تراکنش محاسبه میشود.

سپس اگر روی این ستون ماه های بین تراکنش ها groupby زده شود و در هر کدام تعداد کاربران متمایز شمرده شود در نتیجه مشخص میشود پس از هر تعداد ماه فاصله پس از اولین تراکنش چند کاربر متمایز داریم که اگر بر کل کاربران تقسیم شود retention rate به دست می آید.

- همان طور که در زیر مشاهده میشود تا تنها بعد از چهار ماه کاربر فعال داشته ایم. که مقدار آن نیز بسیار کم و ۵ درصد است. و اکثر کاربران پس از ماه اول ریزش داشته اند.

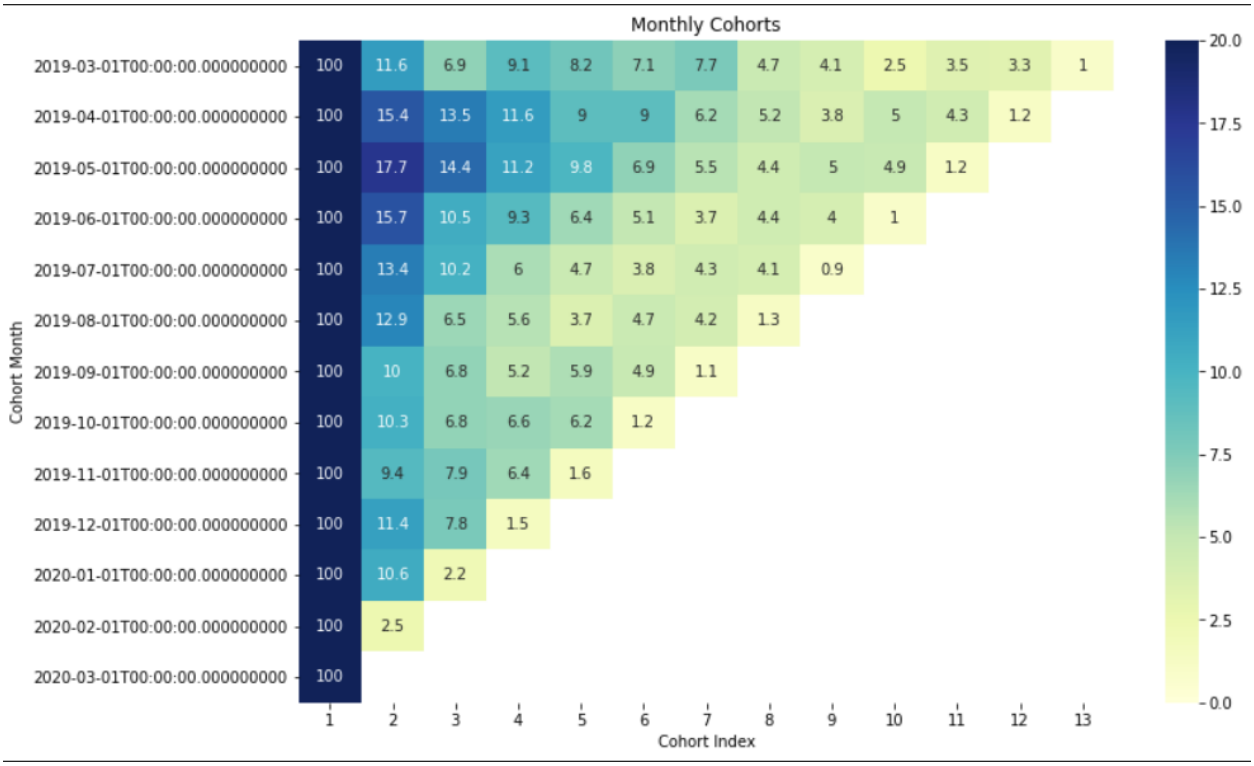


۶- جدول کوهورت ماهانه:

برای محاسبه جدول کوهورت مطابق با retention عمل میکنیم ولی اینبار به جای تنها groupby بر روی فاصله بین اولین تراکنش و تراکنش های بعدی بر روی ماه های مختلف نیز groupby زده میشود.

و سپس بر روی نتیجه pivot_table بر روی نتیجه زده میشود تا تعداد UserID برای ماه های مختلف و فواصل ماه بین اولین تراکنش تا تراکنش های بعدی شمرده شود و در نهایت بر کل کاربران هر ماه تقسیم میشود تا retention rate به دست آید.

نتیجه در زیر مشاهده میشود.

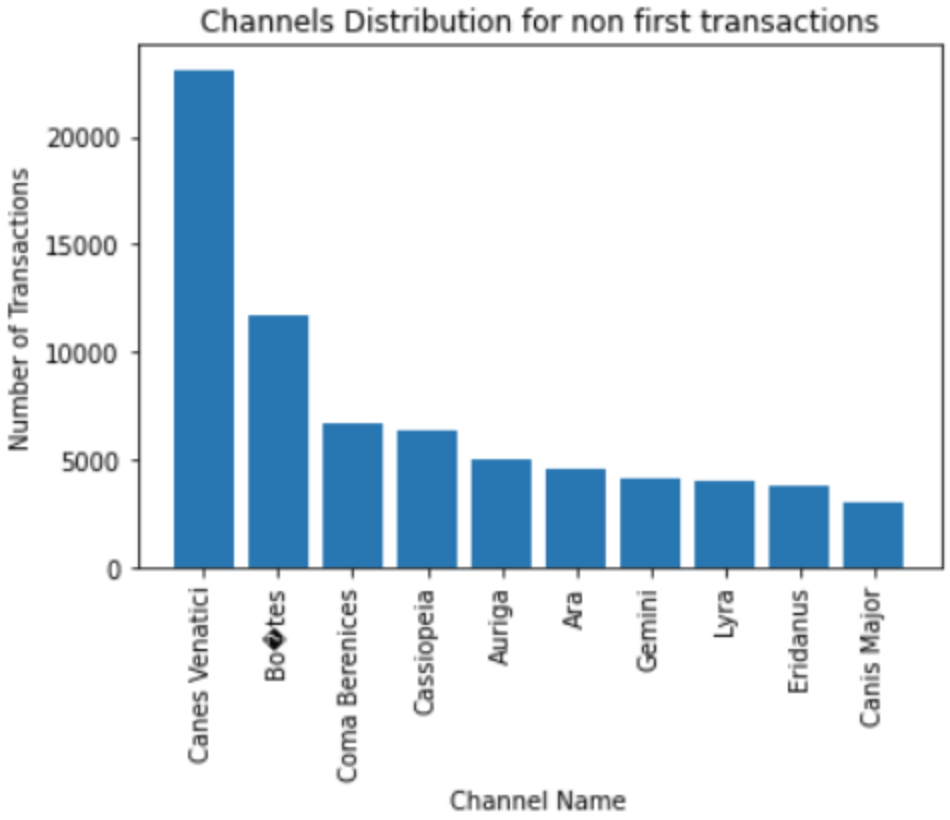


- همان طور که مشاهده میشود در ماه های مختلف پس از یک ماه ریزش بسیار شدید تراکنش ها را داشته باشیم و در واقع به طور کلی میتوان نتیجه گرفت که اکثریت کاربران یک بار در این سیستم تراکنش انجام میدهد.

- البته ماه دوم تا پنجم به نسبت سرعت ریزش کاربران کمتری داشته اند نسبت به ماه های بعدی خود داشته اند. در واقع میتوان نتیجه گرفت بعد از این ماه اول کاربران سریع ریزش داشته اند تغییری در سیستم ایجاد شده که باعث ماندگاری بیشتر کاربران شده است ولی پس از چهار پنج ماه این تغییرات دیگر نتوانسته اند باعث برگشت کاربران شوند و با سرعت بیشتری کاربران ریزش میکنند.

۷- توزیع تراکنش های غیر اول

توزیع ۱۰ کانال با بیشترین تراکنش غیر اول در زیر آورده شده است



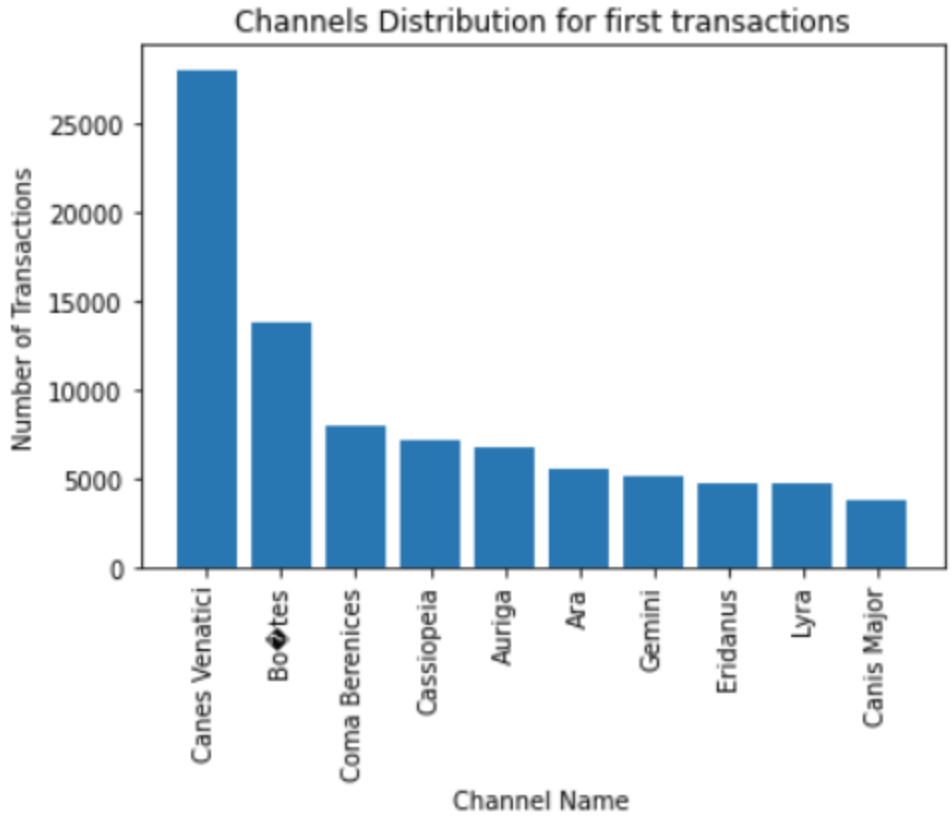
- همان طور که مشاهده میشود Canes Venatici بیشترین کانال تراکنش های غیر اولی را به خود اختصاص داده

تحلیل نتایج

تحلیل نتایج در قسمت هر سوال در بالا به صورت bullet list آورده شده است.

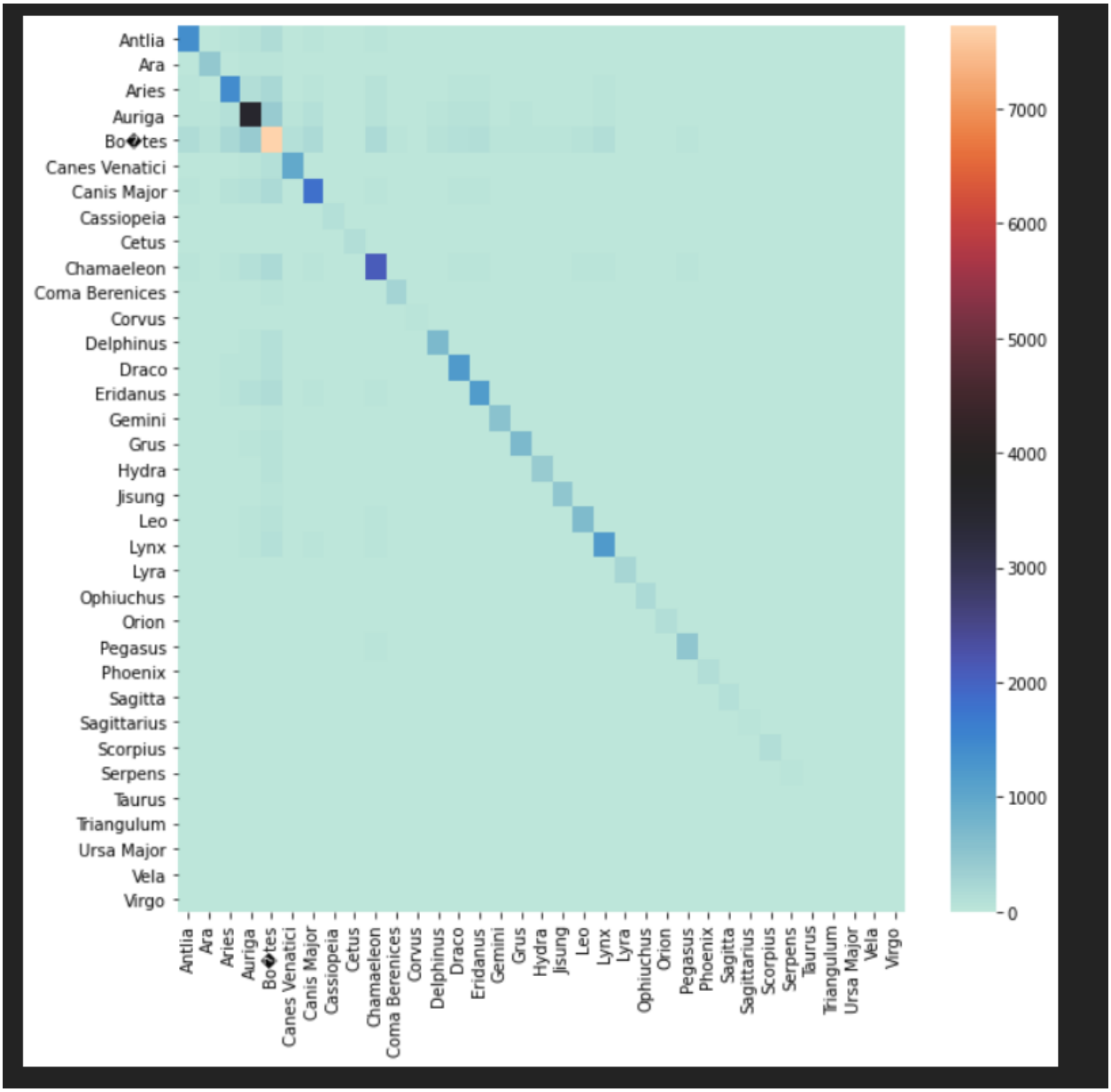
۸- توزیع تراکنش های اول

توزیع ۱۰ کانال با بیشترین تراکنش اول در زیر آورده شده است.



- با مقایسه توزیع کانال های تراکنش های اول و غیر اول نتیجه شباهت کامل به هم دارند که از آن میتوان نتیجه گرفت که افراد کانال های یکسانی را برای تراکنش اول و دوم خود انتخاب کرده اند.

۹- ماتریس جابه جایی بین کانال ها برای تراکنش اول و دوم



- همان طور که مشاهده میشود اکثر افراد یک کانال را برای تراکنش های اول و دوم خود انتخاب کرده اند و کم اتفاق افتاده است که کانال تراکنش خود را عوض کنند.

بخش دوم

سوال ۱

با استفاده از regex هر خط parse شده و در نتیجه در یک dataframe گذاشته شد که در زیر مشاهده میشود.

	timestamp	session_id	event
0	Dec 10 06:55:46	24200	reverse mapping checking getaddrinfo for usern...
1	Dec 10 06:55:46	24200	invalid user username from 173.234.31.186
2	Dec 10 06:55:46	24200	input_userauth_request: invalid user username ...
3	Dec 10 06:55:46	24200	pam_unix(sshd:auth): check pass; user unknown
4	Dec 10 06:55:46	24200	pam_unix(sshd:auth): authentication failure; l...
...	...	...	...
655142	Jan 17 17:07:14	30222	Received disconnect from 185.165.29.69: 11: By...
655143	Jan 17 17:13:12	30238	Accepted password for username from 137.189.20...
655144	Jan 17 17:13:12	30238	pam_unix(sshd:session): session opened for use...
655145	Jan 17 17:22:01	30291	Accepted password for username from 183.11.69....
655146	Jan 17 17:22:01	30291	pam_unix(sshd:session): session opened for use...

655147 rows × 3 columns

سوال ۲

با استفاده از regex ابتدا ip ها با کاراکتر @@ و سپس اعداد با کاراکتر ## جایگزین شده اند که نتیجه برای تعدادی از eventها در زیر مشاهده میشود.

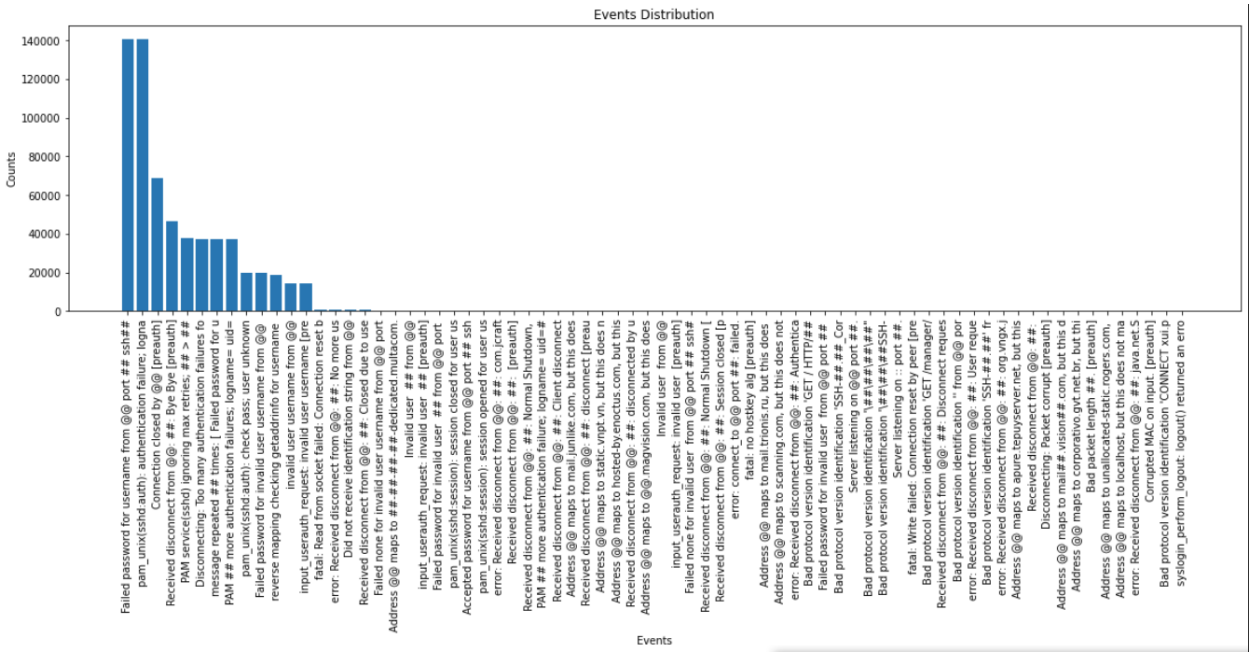
	timestamp	session_id	event
0	Dec 10 06:55:46	24200	reverse mapping checking getaddrinfo for usern...
1	Dec 10 06:55:46	24200	invalid user username from @@
2	Dec 10 06:55:46	24200	input_userauth_request: invalid user username ...
3	Dec 10 06:55:46	24200	pam_unix(sshd:auth): check pass; user unknown
4	Dec 10 06:55:46	24200	pam_unix(sshd:auth): authentication failure; l...
...	...	...	...
655142	Jan 17 17:07:14	30222	Received disconnect from @@: ##: Bye Bye [prea...
655143	Jan 17 17:13:12	30238	Accepted password for username from @@ port ##...
655144	Jan 17 17:13:12	30238	pam_unix(sshd:session): session opened for use...
655145	Jan 17 17:22:01	30291	Accepted password for username from @@ port ##...
655146	Jan 17 17:22:01	30291	pam_unix(sshd:session): session opened for use...

655147 rows × 3 columns

(a

۸۰ نوع event مختلف در داده ها وجود دارد و توزیع آن مطابق با زیر است.

برای این که نمودار را بتوان بهتر نشان داد ۵۰ کاراکتر اول هر نمودار در محور x آورده شده است.



- همان طور که مشاهده میشود ۱۳ event به نسبت بقیه دارای تعداد تکرار زیاد هستند و بقیه eventها eventهایی با رخداد بسیار کم هستند.

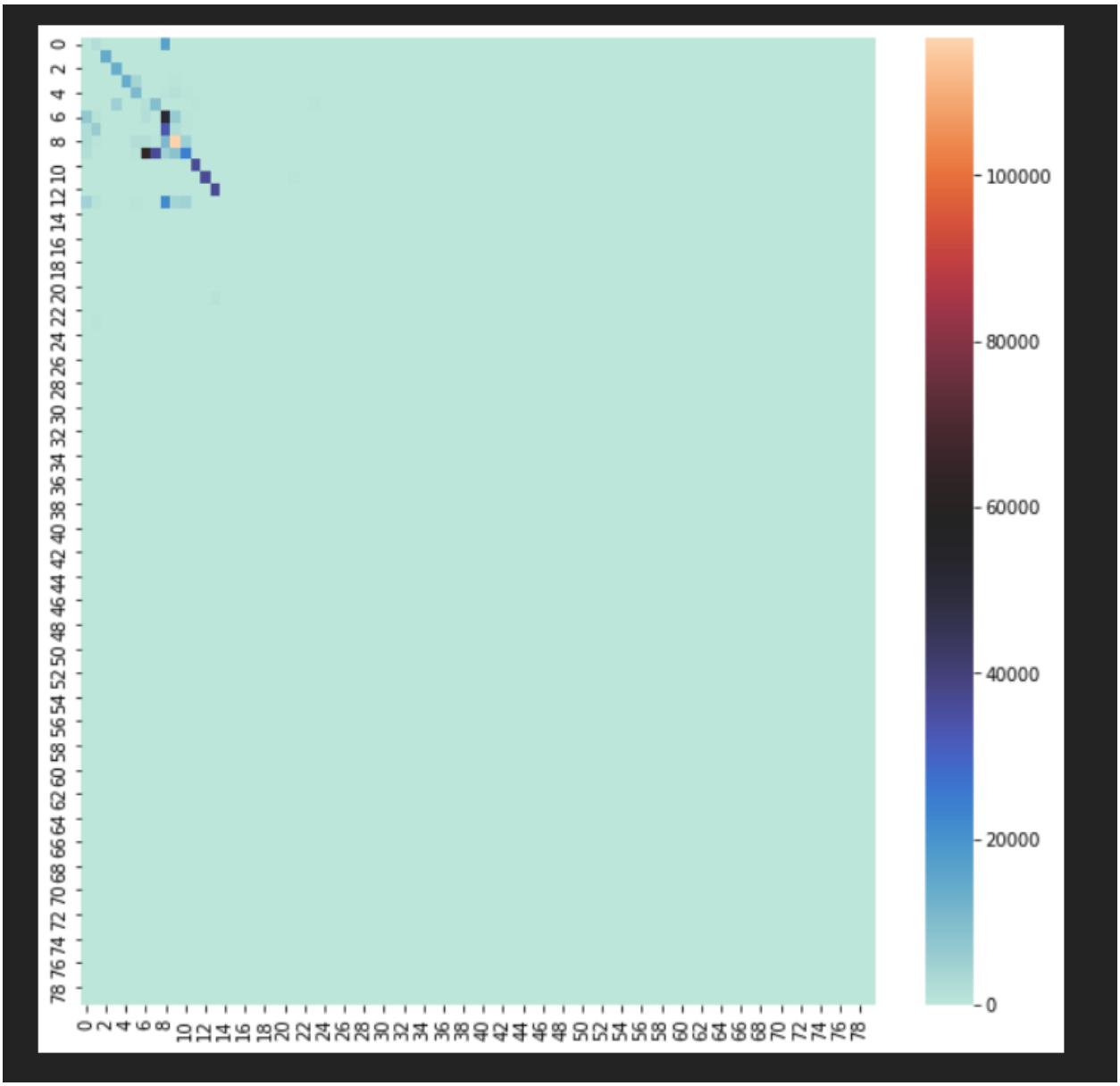
(b

ماتریس جابه جایی بین eventها بر اساس ترتیب آن ها محاسبه شده است.

یعنی دو event متوالی در این ماتریس یک مقدار به سطر event اول و ستون event دوم اضافه کرده اند.

همان طور که مشاهده میشود تنها برای ۱۵ event اول ماتریس جابه جایی مقدار بالایی گرفته است که به این معناست که اولا این eventها خودشان زیاد پشت سر هم قرار گرفته اند و هم چنین در تعدادی از آن ها نیز هر جفت event زیاد پشت سر هم قرار گرفته اند.

اما در بقیه event ها رابطه ای بین پشت سر هم قرار گرفتن eventها وجود ندارد.



این پانزده event ابتدایی نیز در زیر آورده شده اند.

```
0 reverse mapping checking getaddrinfo for username [@@] failed - POSSIBLE BREAK-IN ATTEMPT!
1 invalid user username from @@
2 input_userauth_request: invalid user username [preauth]
3 pam_unix(sshd:auth): check pass; user unknown
4 pam_unix(sshd:auth): authentication failure; logname= uid=## euid=## tty=ssh ruser= rhost=@@
5 Failed password for invalid user username from @@ port ## ssh##
6 Connection closed by @@ [preauth]
7 Received disconnect from @@: ##: Bye Bye [preauth]
8 pam_unix(sshd:auth): authentication failure; logname= uid=## euid=## tty=ssh ruser= rhost=@@  user=username
9 Failed password for username from @@ port ## ssh##
10 message repeated ## times: [ Failed password for username from @@ port ## ssh##]
11 Disconnecting: Too many authentication failures for username [preauth]
12 PAM ## more authentication failures; logname= uid=## euid=## tty=ssh ruser= rhost=@@  user=username
13 PAM service(sshd) ignoring max retries; ## > ##
14 Did not receive identification string from @@
```