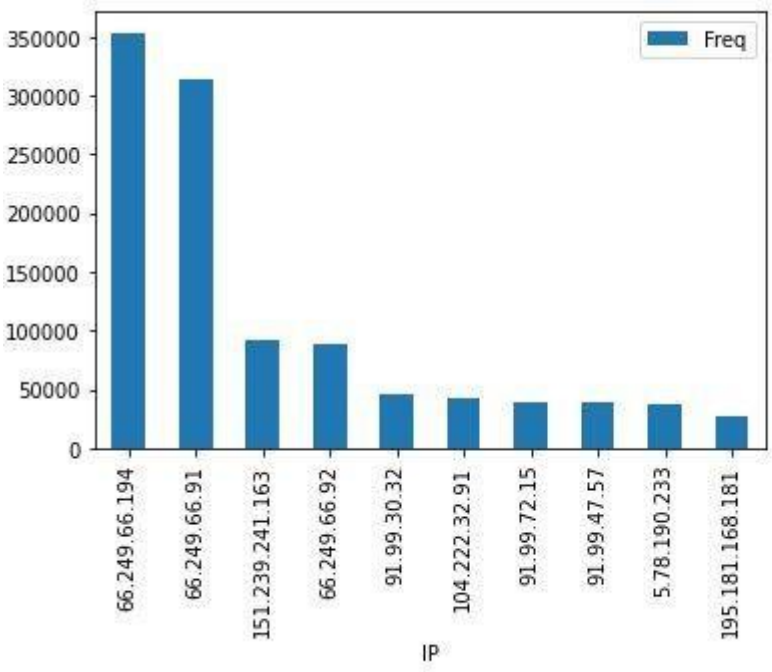
گزارش تمرین کدی دوم هوش.

امیرحسین ادواری ۹۸۲۴۳۰۰۴

سوال اول)

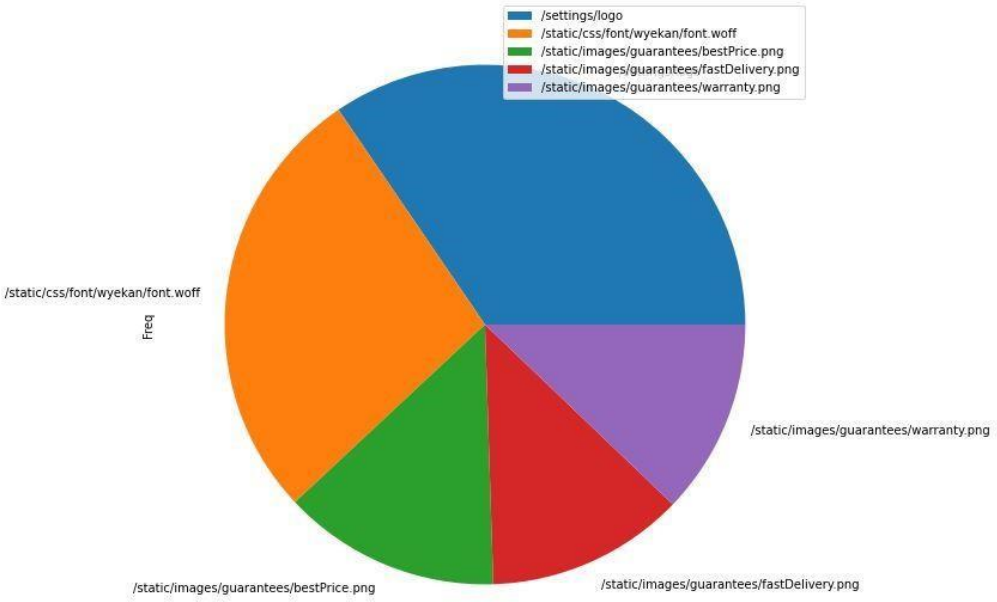
در سوال اول ابتدا شمار وقوع هر کلاینت را شمارش کرده(value counts) و سرت میکنیم (خود تابع ذکر شده اینکار را انجام میدهد)سپس کلیدها و مقادیر آنرا به تفکیک به تابع رسم نمودار ستونی در پانداز میدهیم و نمودار مربوطه رسم میشود :



سوال دوم)

عملی مشابه قسمت قبل را روی request ها انجام میدهیم و اینبار از تابع رسم pie در pandas استفاده میکنیم.

کلیدها و مقادیر خرویج را به تفکیک به تابع رسم pandas میدهیم و نمودار مربوط به آن رسم میشود.



سوال سوم)

در این سوال 2 دیتافریم جدید میسازیم در یکی از آنها ردیف هاین را که is_image درست برمیگرداند را True کرده (از طریق apply)و در دیگری نیز همینکار را برای ردیف هایی که is_real_image آنها درست برمیگرداند انجام میدهیم.

سپس تعداد ردیف هاین در real_images که ستون request آنها True شده را تقسیم بر تعداد ردیف های images که ستون request آنها True شده است میکنیم و جواب بدست می آید.

نکته 1 : بسته به اینکه چه رجکسی استفاده شود (با توجه به بحث های گروه) نتیجه متفاوت خواهد بود.

نکته 2: اجرای این عملیات زمان نسبتا زیادی را میطلبد .

سوال چهارم)

قسمت اول:

i) **مشابه سوالهای اول و دوم تعداد رخداد هر status را از طریق value_counts() به دست می آوریم و کلیدها و مقادیر را جداگانه به تابع رسم(pandas.DataFrame.plot.bar) میدهیم تا رسم را انجام دهد.**

ii) **برای اینکه status کد های منحصربفرد را بدست بیاوریم کافیست تابع drop_duplicates(subset=["status"]) را روی دیتافریم اصلی فراخوانی کنیم تا ردیفهای با status کد تکراری حذف شوند و انحصار در هر ردیف باقیمانده برقرار شود.**

قسمت دوم :

همه status های4xx و5xx موجود را در یک لیست ریخته و مقادیر 4xx و 5xx آنهارا در دو لیست دیگر(np.array) نگه میداریم .

ابتدا همه ردیف هاین که status آنها در لیست اول وجود دارد را تحت یک دیتافریم بدست می آوریم و سپس از طریق ستون datetime آنها ستون hour را که از apply کردن hour_function (تابعی که datetime میگرید و hour برمیگرداند)بدست می آید را به آن اضافه میکنیم .

سپس از طریق دولیستی که داشتیم table5xx و table4xx را مشابها بدست می آوریم و برای هرکدام روی hour گروپبای زده و شمار رکورد های موجود در هر گروپ را ست میکنیم .

درنهایت با InnerJoin کردن این د و دیتافریمی که بدست می آید به پاسخ مریسیم.

[5]:

hour	5xx	4xx
0	17	6021
1	10	4215
2	8	2394
3	26	1388
4	3	1871
5	10	1683
6	3	2036
7	6	2923
8	76	4985
9	220	7574
10	124	9117
11	255	10102
12	429	10395
13	528	10095
14	597	10584
15	581	10513
16	621	9823
17	202	9351
18	4816	8979
19	6550	8302
20	36	7758
21	6	7019
22	22	7342
23	21	7786

سوال پنجم:

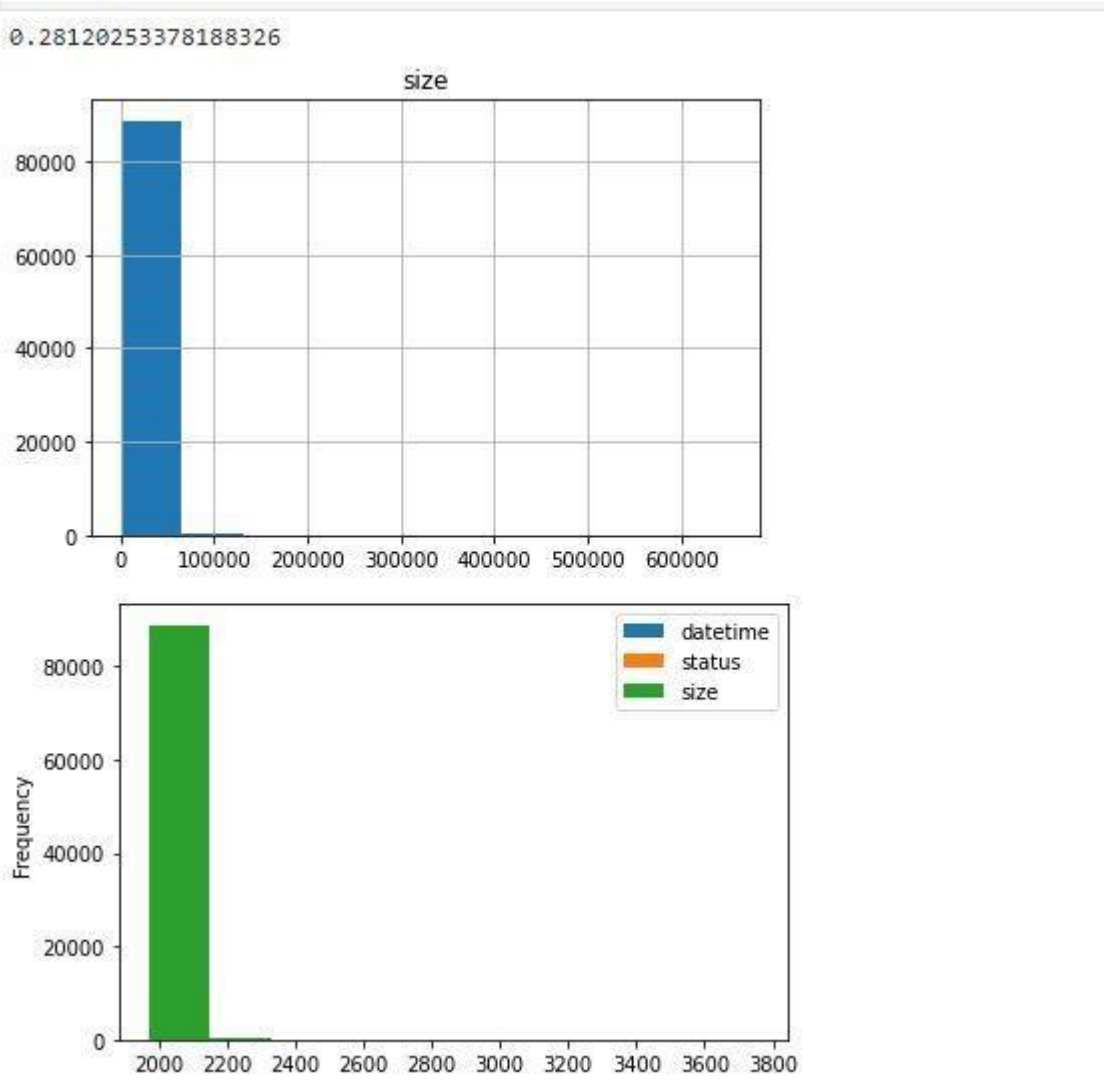
در این سوال بر اساس user_agent و client گروپ بای میکنیم و سپس یک دیتافریم جدید میسازیم مطابق کد به ازای هر ستون یکی از توابع کتابخانه user_agents را روی المنت های ستون user_agent اپلای میکنیم.

	browser-family	os-family	is-bot	is-pc
0	IE	Windows	False	True
1	Chrome	Windows	False	True
2	Chrome	Windows	False	True
3	Firefox	Windows	False	True
4	Chrome	Windows	False	True
...	...	...	...	...
295082	Chrome Mobile	Android	False	False
295083	Android	Android	False	False
295084	Firefox	Windows	False	True
295085	Google	Android	False	False
295086	Chrome Mobile	Android	False	False

295087 rows × 4 columns

سوال ششم :

یک دیتافریم جدید میسازیم و در آن ردیف هاین که با شروط داده شده همخوانی دارند را مریزیم (از دیتافریم اصلی) سپس این دیتافریم را از طریق ۲ تابع مربوط به هیستوگرام که در پانداز وجود دارد رسم میکنیم و لیبل های مناسب را به آنها میدهیم. (این توابع خودشان n_bins را بعنوان ورودی میگیرند)



سوال هفتم:

۱. نرخ تعداد کلیک:

یک خصیصه قابل محاسبه در هر نشست با درخواست های http وجود دارد میتوان از طریق تعداد درخواست ارسالی تعداد کلیک را که یکی از این خصیصه ها میباشد بدست آورد اگر تعداد کلیک ها به طرز غیر طبیعی بالا باشد خبر از وجود یک اسکریپت در حال اجرا میدهد .

۲. نسبت صفحات html به عکس :

مشابه حالت قبلی اگر این نسبت به طور غیر طبیعی زیاد شود نشان از وجود یک کرالر است زیرا آنها معمولا از عکسها صرفنظر میکنند.

۳.درصد دریافت pdf/ps :

کرالر ها برخلاف انسانها تمایل زیادی به این درخواست ها دارند. رشد غیرطبیعی این درخواست ها میتواند حاکی از وجود یک کرالر باشد.

۴. درصد رسیپانس های از نوع ۴۰۰ یا 4xx :

در مورد کرالر ها نرخ این ریسپانس ها بسیار بیشتر از انسانهاست زیرا بسیار محتمل است که به صفحات حذف شده درخواست بزنند.

۵. درصد درخواست های HEAD :

یک انسان هنگام بازدید از وبسایت از درخواست های GET استفاده میکند اما در مورد رباتها اینطور نیست آنها برای کاهش میزان اطلاعات دریافتی از متد HEAD استفاده میکنند لذا اگر نرخ این درخواست ها در یک نشست بالا رود میتوانیم وجود یک کرالر را تشخیص دهیم.

۶.نرخ درخواستهای بدون پر شدن Referrer :

بیشتر درخواست هایی که کرالر ها میزنند این فیلد را ندارند لذا از این طریق میتوان در یک نشست به حضور آنها پی برد.

۷.درخواست به Robots.txt:

این اندپوینت برای اینکه به باتها نشان دهند کدام قسمت های سایت را میتوانند ملاقات کنند ایجاد میشود و فرمت خاص خود را دارد. درخواست زدن به این اندپوینت در یک نشست غالبا توسط کرالر ها صورت میگرید لذا از این طریق میتوان به حضور آنها پی برد.

۸. انحراف معیار عمق صفحات درخواستی در یک نشست :

اگر انحراف معیار عمق صفحات درخواستی در یک نشست زیاد شود نشان از ربات است .

مثلا عمق اندپوینت روبرو ۲ است : src/app

۹. درصد درخواست های متوالی با یک الگوی خاص :

مثلا اگر متوالیا به اندپوینت هایی با الگوی :

Src/app/*

درخواست زده شود میتواند حاکی از وجود کرالر باشد.

۱۰.از طرف دیگر اگر میزان منابع درخواستی در یک نشست بسیار زیاد باشد نیز میتوانیم به وجود یک کرالر پی ببریم.

منبع : مقاله قرارداده شده.

سوال ۸ام :

کست کردن در سوال ۸ام با کل دیتاست منجر به کرش کردن میشود.

کد سوال هشتم زده شده که صرفا یک گروپبای و شمارش و پس از آن سرت کردن است ..

با یک سمپل ۱ درصدی گرفتن از دیتاست میتوان از مشکل کرش کردن جلوگیری کرد.

برای حل سوال ۹ توابعی مینویسیم که هرکدام یک سریز گرفته و مقدار مدنظر را برمیگرداند (اعم از میانگین و تعداد و ....)

این توابع را در تابع تجمعی agg که روی گروپبای زده میشود کال میکنیم. و خود پانداز عملیات های مختلف را روی گروهها انجام میدهد و نتیجه را در ستون با نام مرتبط قرار میدهد. (توضیحات تکمیلی هنگام ارائه)

		status	method	referer			request	size	datetime
		percentage4xx	percentageHEAD	percentageUNASSIGNED	countRobotsTxt	stdLength	requests_count	mean	avgReqTime
client	user_agent								
1.234.99.77	Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko	0.0	0.0	0.0	0	0.000000	2	894.000000	1.00
10.1.52.71	Mozilla/5.0 (Windows NT 6.1; rv:59.0) Gecko/20100101 Firefox/59.0	0.0	0.0	0.0	0	0.816497	3	32631.333333	589.00
10.1.68.25	Mozilla/5.0 (Windows NT 6.1; rv:64.0) Gecko/20100101 Firefox/64.0	0.0	0.0	0.0	0	0.471405	6	3040.666667	32.00
10.10.56.92	Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36	0.0	0.0	0.0	0	0.000000	2	5982.000000	399.00
10.103.61.149	Mozilla/5.0 (Windows NT 6.1; rv:62.0) Gecko/20100101 Firefox/62.0	0.0	0.0	0.0	0	0.000000	1	5807.000000	0.00
...	...	...	...	...	...	...	...	...	...
97.107.137.22	Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)	0.0	0.0	0.0	0	0.000000	2	2151.500000	11.00

سوال نهم قسمت دوم)

تعداد ریکوئست بالا در مدت زمان کوتاه،

سایز غیرطبیعی،

تعداد درخواست بالا برای Robots.txt ،

تعداد بالای درخواست های HEAD ،

و به عبارتی همه پارامتر هایی که در سوال 9 خواسته شده اند، میتوانند در تعیین کرالر ها تاثیر زیادی داشته باشد.