Assignment #1 – Data Preprocessing and Visualisation

Yoke Kai Wen, 2020280598

September 25, 2020

We were given 2 weeks of search logs from a Chinese search engine. Using python libraries: `pandas`, `matplotlib` and `seaborn`, I generated the data visualisations below. Timestamps were formatted to the 'Asia/Shanghai' time zone.

1 Data visualisation

1.1 Line chart showing SRT every 10min

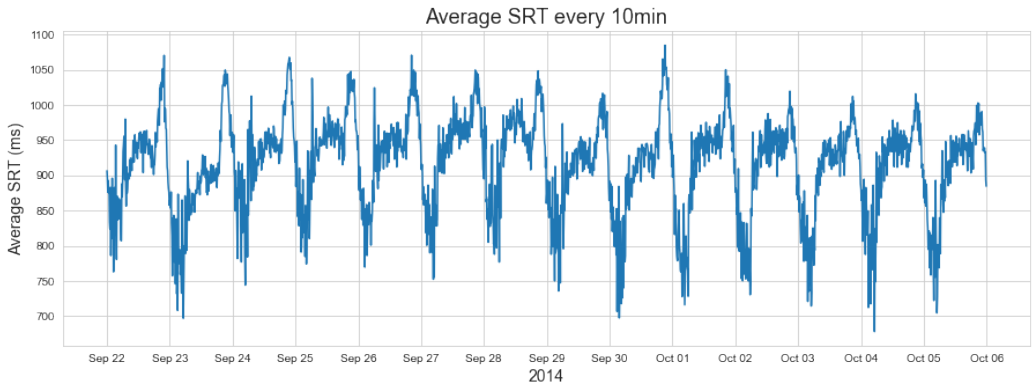

Figure 1: Average SRT every 10min

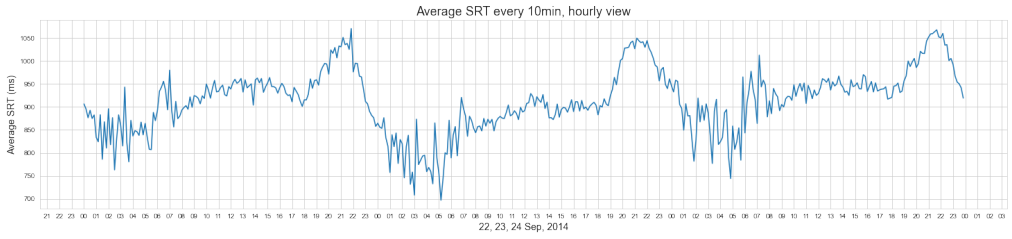

Figure 2: Average SRT every 10min zoomed in to see every hour

From Figure 1, SRT follows a daily seasonal pattern. Zooming in, (see Figure 2), I observe that the SRT peaks at 9-10pm every day, and SRT dips at 1am-6am every day. This makes sense

because after work and before sleep, people would have free time to browse through the internet, thus the traffic on the search engine would increase, leading to slower service as reflected by higher SRT. The converse happens in the wee hours of morning when most people are sleeping, leading to low SRT.

1.2 Stacked area chart showing breakdown of SRT every 10min

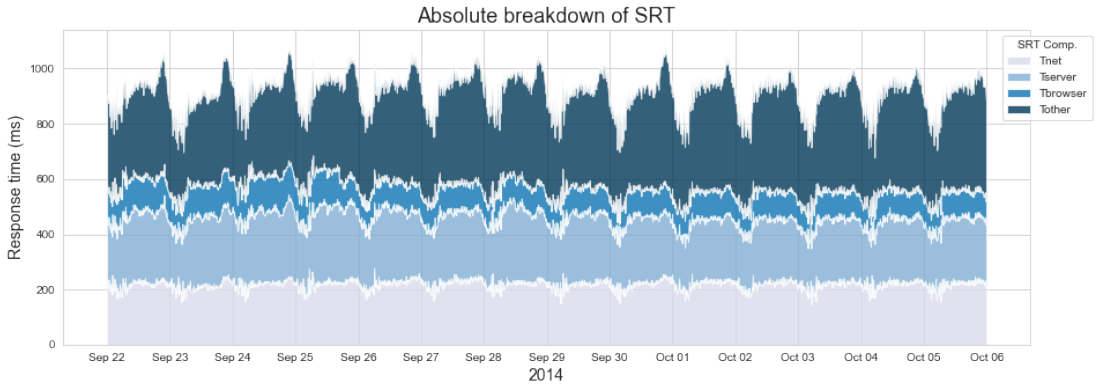

Figure 3: Absolute breakdown of SRT every 10min

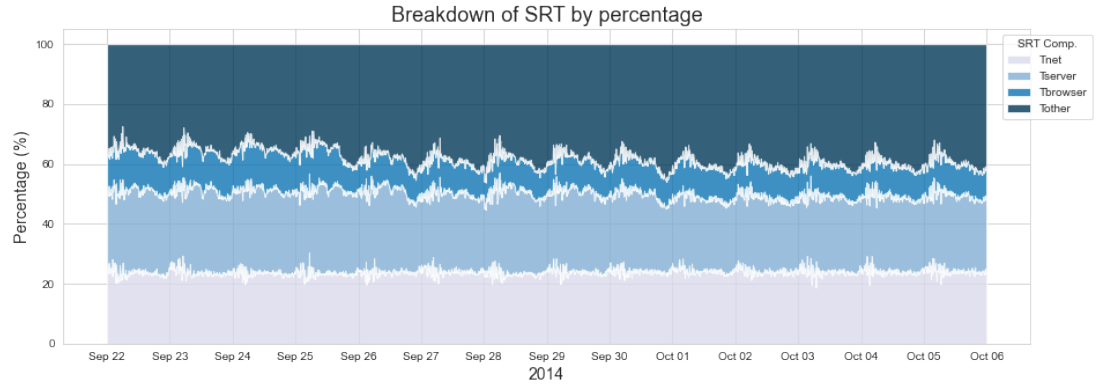

Figure 4: Percentage breakdown of SRT every 10min

From Figure 3, I observe that all four components of SRT follow the same daily seasonal pattern, with T_other showing the largest fluctuation and T_net showing the smallest fluctuation. However, this fluctuation is quite difficult to tell since each additional quantity is added onto the quantity before, so the amount of fluctuation naturally increases with each added quantity. From Figure 4, we can more easily tell the proportions occupied by each component, with $T\_browser$ the smallest, T_other the largest and T_server and T_net roughly the same. I found it quite surprising that T_net was relatively stable throughout the day despite the variation in traffic, perhaps this suggests that the network's capacity is greater than peak demand.

1.3 CDF chart of SRT

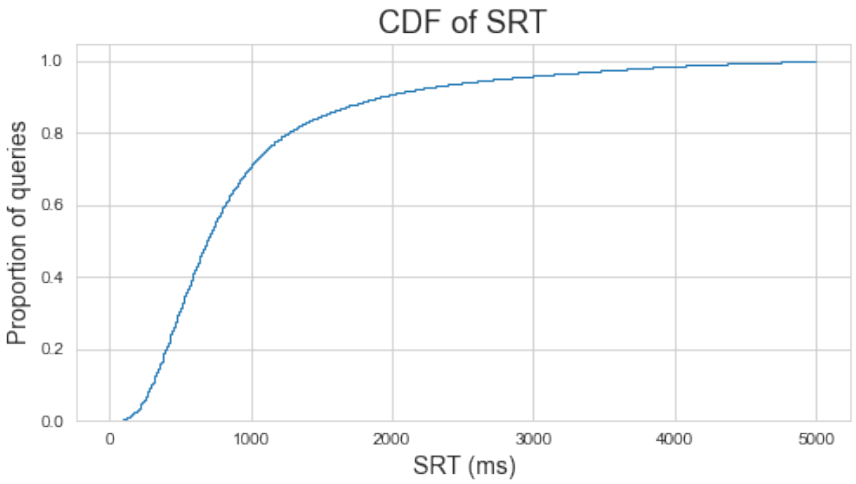

Figure 5: CDF chart of SRT

From the CDF chart of SRT in Figure 5, I observe that most queries (70%) are served under 1s, while there are occasions when the SRT of a query would take as long as 5s. This narrows our focus on the queries that take especially long to be served, and it seems like 1s is a good cut-off point to determine if the SRT of a query is too long.

1.4 CDF chart of #Images

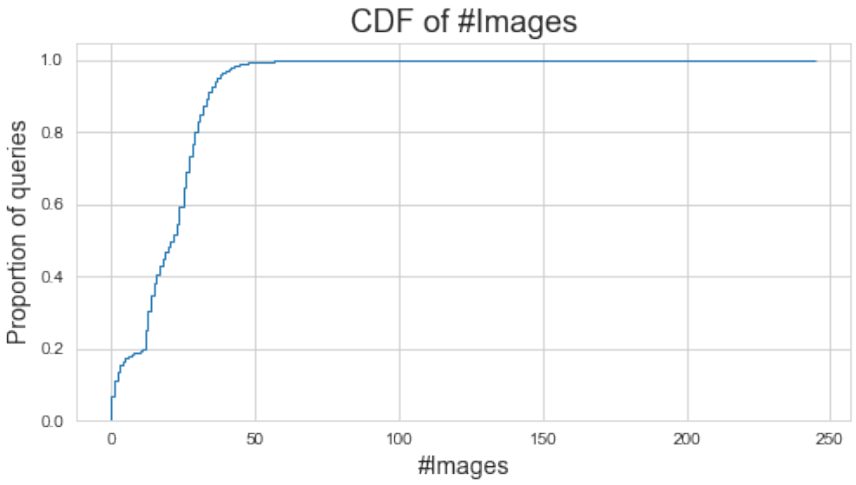

Figure 6: CDF chart of #Images

From the CDF chart of number of images embedded in result page (Figure 6), I observe that most result pages have under 50 images, with each page having an average of roughly 25 images.

The outliers are also very obvious, with pages having 250 images being very rare, and probably can be ignored during analysis.

1.5 Line chart showing number of queries (PV) every minute

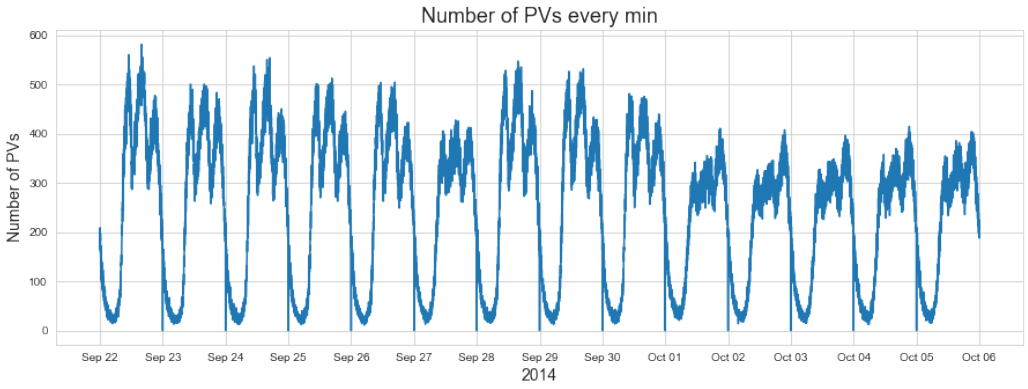

Figure 7: Number of PVs per minute

From Figure 7, I observe:

1. A daily seasonal pattern for the number of page views per minute;
2. The number of page views was significantly lower on 27 Sep 2014, and for 1-6 Oct 2014. I am not sure why, maybe because the October period was a long public holiday in China and people tended to travel or hang out with friends during this holiday, and so they would use the internet and search engine less often, but I am not sure what is special about 27 Sep 2014;
3. At 23:59 GMT+8 every day there are zero queries (I double checked with the unix timestamp) as seen by the vertical lines at the end of each day on the line chart, and I don't understand why, maybe it was intentional when this dataset was selected for the assignment;

Also, I guessed that the number of page views should be closely correlated to SRT, so I plotted Figure 8 to compare PV and SRT.

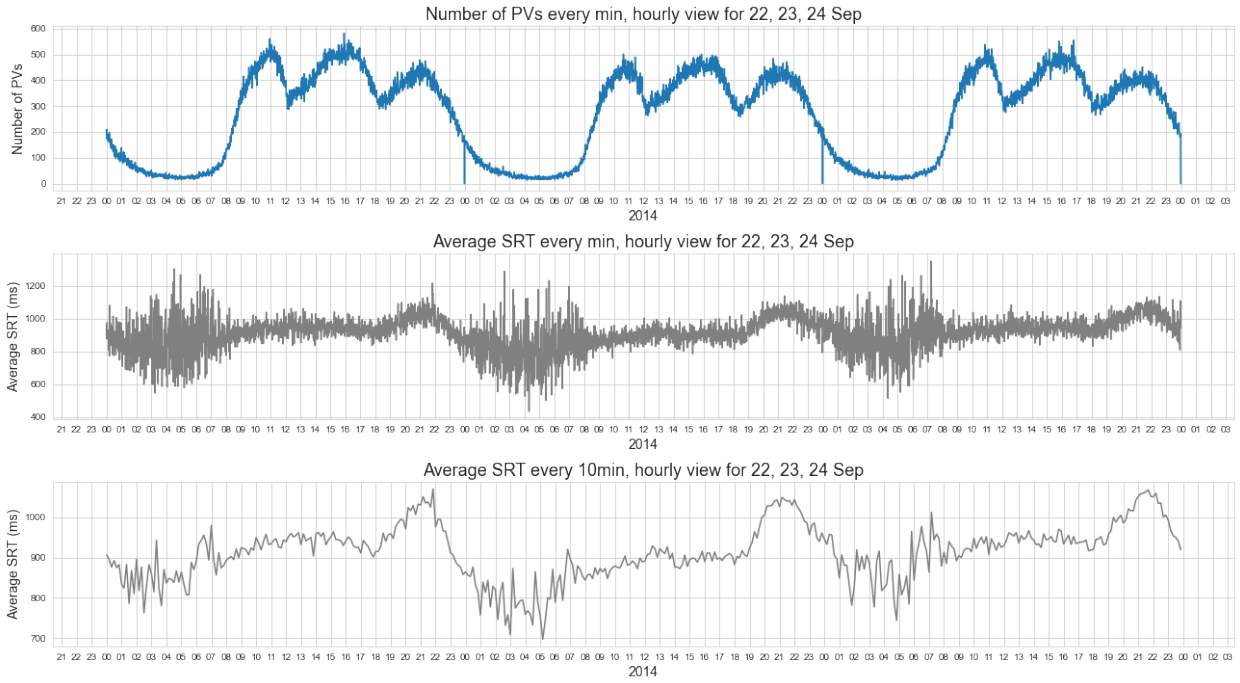

Figure 8: Comparison between number of PVs per minute, mean SRT per min and mean SRT per 10min, zoomed in for hourly view

Figure 8 shows that PV and SRT are not that closely correlated which is quite surprising. In fact, at non-peak periods (1am to 6am), the per-minute-SRT seems to fluctuate a lot, while the 10minute-SRT shows the troughs more clearly. Also, the PV line chart shows three peaks during the busy period, but SRT only shows one prominent peak from 9pm to 10pm. Perhaps at 9pm to 10pm, which is usually the time when people rest, watch dramas and game on internet, there is a lot more burden on the browser, network and servers, thus slowing down the search engine's search response time? Since page views only reflect the activity on search engine and do not reflect other activities hogging communication resources, therefore we do not see an explicitly proportional relationship with SRT.

1.6 Histogram chart showing PVs of each province

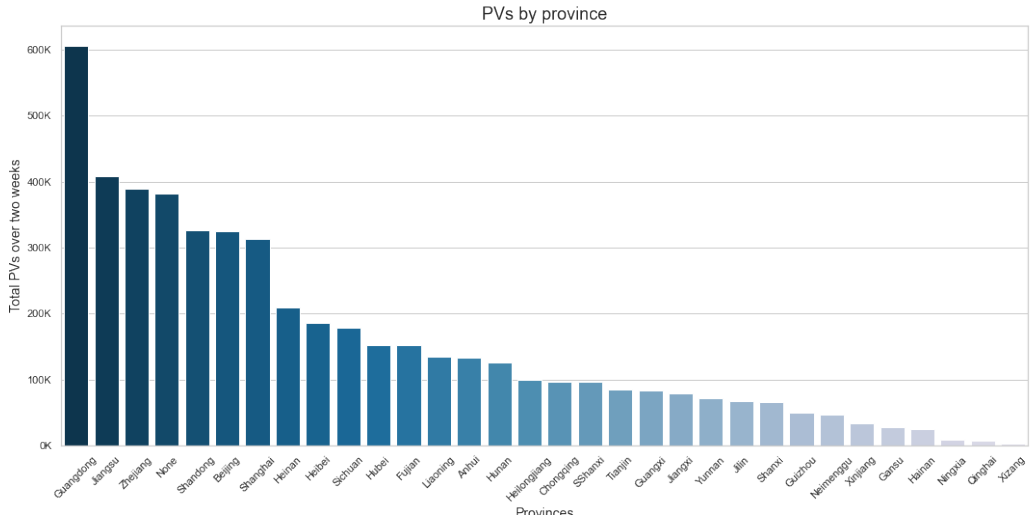

Figure 9: PV per province

The histogram chart in Figure 9 shows the distribution of page views across different provinces, with Guangdong leading ahead by far, followed by Jiangsu, Zhejiang, None, Shandong, Beijing, Shanghai and the rest trailing far behind. The top provinces are wealthier with more people having access to the internet and are also younger. Guangdong has Shenzhen where many of the big Chinese tech companies reside in, so it is unsurprising that it had the busiest search engine usage. I am not sure what 'None' is, data entries lacking important details should be removed from the dataset.

1.7 Pie chart showing PVs of each UA

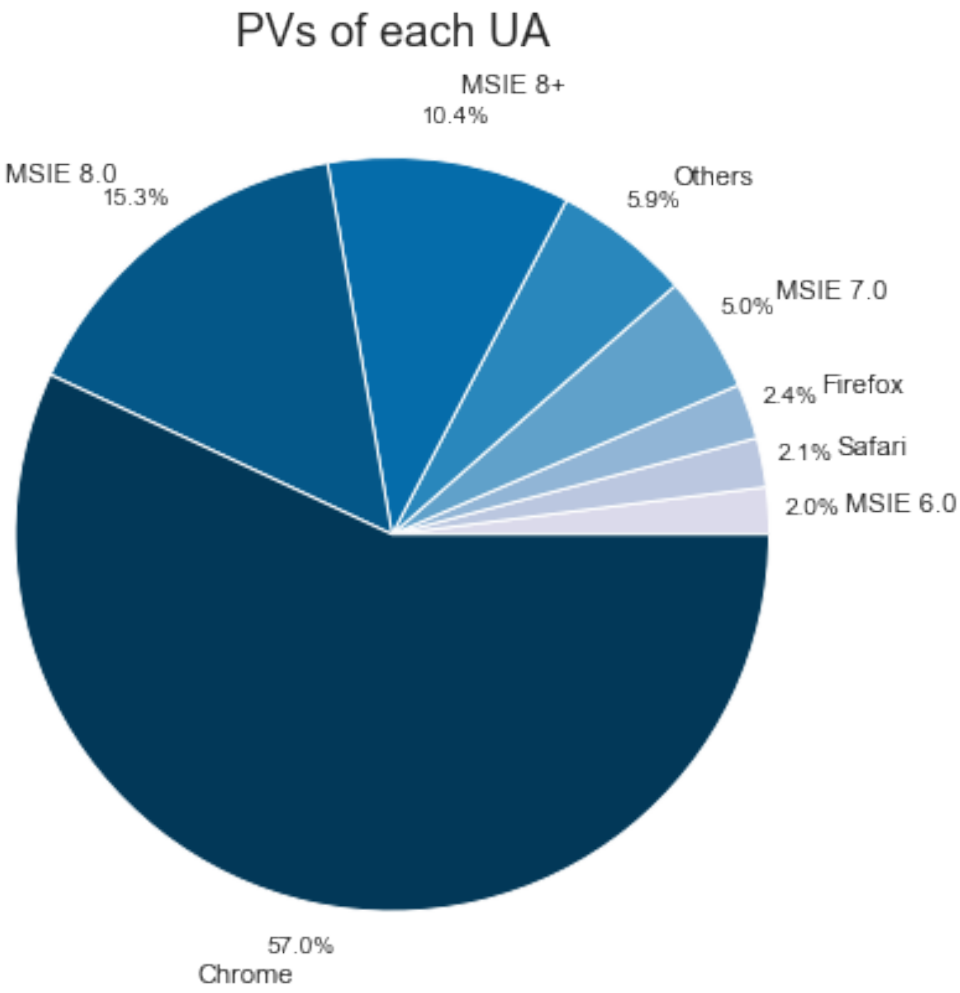

Figure 10: PV per UA

The pie chart in Figure 10 shows the percentage of page views served by each browser type. Chrome dominates with 57% of all page views, followed by different versions of Internet Explorer and the rest occupying a small fraction. I always hear that Google is banned in China so I was surprised to see that not only was Google Chrome usable, it was in fact the most popular choice.

2 Discussion

I found data visualisation very useful in helping me to notice interesting insights about the raw dataset.

2.1 Differences among different charts

In this assignment, we were asked to try out various chart types: line chart, area chart, CDF chart, histogram chart and pie chart.

1. Line chart is a simple and effective way to see the variation of a variable (or more) over time;
2. Area chart is good at seeing how one component is broken down to its individual components - however it is difficult to see the absolute value of each component since they are all stacked on top of each other;
3. Percentage stacked chart does not appear to be very useful to me, except to compare the consistency of the proportion dominated by each individual component;
4. CDF chart is good for understanding the range and variation of values of a variable, and also helps us notice outlying values, and find quantiles, and thus is significantly more powerful than histograms which simply shows the distribution of each quantity range;
5. Histogram/ bar chart is good for observing distribution over a number of discrete categories;
6. Pie chart is good for observing the proportion occupied by each category compared to the whole, especially when there are not too many catgories.

2.2 My Experience of data preprocessing and visualisation

I spent quite a long time figuring out what functions I can use for aggregating data and formatting the timestamp in `pandas` and for tuning colours, font sizes etc in `matplotlib`. Fortunately, there are plenty of tutorials and help online. These python libraries are incredibly useful and offer very flexible customisation, if we spend the effort looking through documentation and tuning everything from font to position to colour. Unfortunately, the default settings are usually quite ugly, so it takes a lot of work. Because of this, I tried turning to Power BI to see if they had better design options offered by default, but it turned out to be harder to use than I thought. Another thing was data loading in Power BI was excruciatingly slow. So I went back to python.