**Tech choices -**

The language I have chosen write code in is R. I have chosen R because matrices are very easy to handle and base functions like assign, aggregate and quantile make the given assignment more convenient to finish. Also, ggplot is a quick and easy option for plots.

**Description of approach to the exploration -**

First, I removed the only row containing NA values.

Second, all the rows were deleted where ‘connected’ is equal to ‘false’ (49201 such rows).

This was based on the assumption that the sessions in which the user didn’t connect with relevant backend simply are not pertinent to the analysis.

To check the performance of each session (row), I decided to add a new column named ‘p2p\_cdn’, whose value is equal to (p2p / p2p+cdn), for each row. The variable will have a value between 0 and 1, with 1 being the best case scenario. If it is 0 for any case, that would mean no data was downloaded through p2p.

Now, I decided to split the data into two parts - one where ‘p2p\_cdn’ is equal to 0, and where it is not. This action was based on the assumption that the reasons behind ‘p2p\_cdn’ being equal to 0 and being greater than, but very close to 0, could be different. For the first scenario, it is possible that the p2p backend simply fails to work for whatever reason, or because no peers were available at the time of the video being streamed. A value close to 0 could be attributed to poor performance of a particular stream, isp or browser (or, again, to very few relevant peers available at time of streaming).

Another reason to separate the data is that analysis of the data as a whole created skewed results which were hard to diagnose. More on this in the next section.

To analyse the first set of data, I have found the percentage of rows for each entity (i.e isp, browser, stream\_id) where p2p = 0 (despite connected = ‘true’) corresponding to total rows for each. The higher the percentage, the poorer the performance.

To analyse the second set of data (where ‘p2p\_cdn’ > 0), I have found the quartile for each entity (i.e isp, browser, stream\_id). A quartile is three points that divide sorted data set into four equal groups (by count of numbers), each representing a fourth of the distributed sampled population - Q1, Median, Q3. The higher these three values are, the better the performance.

**Data-driven recommendations to improve the service**

Following is the analyses of the two datasets -

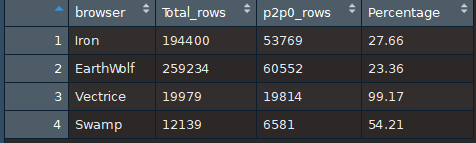
1) Where ‘p2p\_cdn’ is 0.

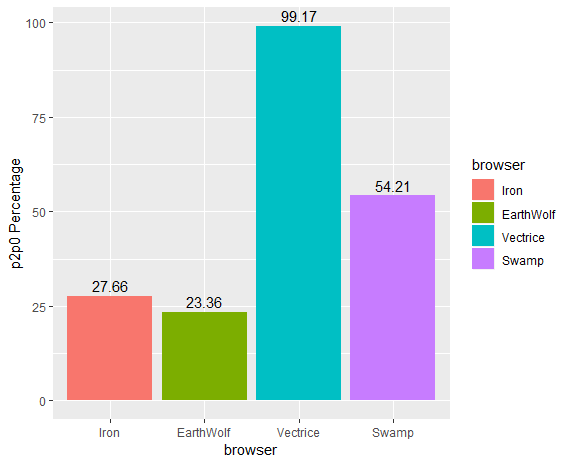
For each browser, isp, and stream, the percentage of rows where ‘p2p\_cdn’ is 0 out of the total number of rows for that particular entity is calculated. Higher the percentage, worse the performance.

Each data point has the following dimensions :

* Total\_rows - the number of rows corresponding to each entity (i.e a browser, an isp or a stream id)
* p2p0\_rows - the number of rows of each entity where data downloaded through p2p is 0.
* Percentage - (p2p0\_rows/Total\_rows)\*100

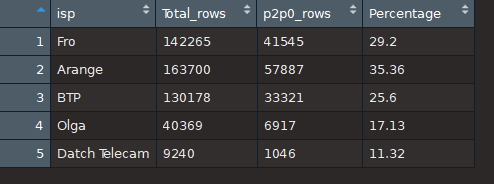
**a) browser**

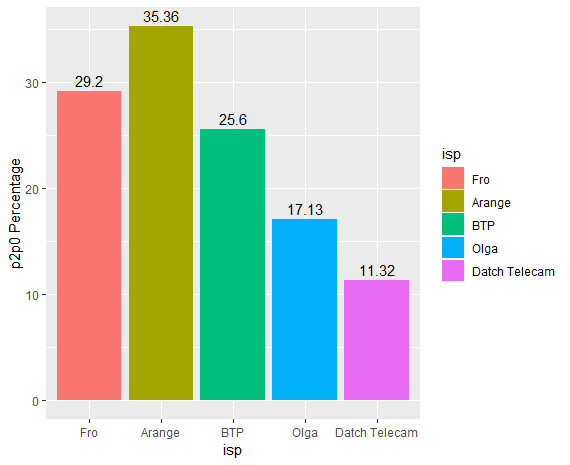




* As you can see in the table above, the browser **‘Vectrice’** has an astonishing 99.17 percentage of rows where **‘p2p\_cdn’** = 0. Such a high percentage cannot be attributed to any factor other than that **‘Vectrice’** simply fails to use the p2p technology in almost all scenarios.

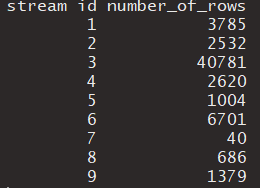
**b) isp**





A certain percentage of rows can be ignored for every isp - based on the assumption that some sessions are going to have p2p = 0 in every sample set because of geographical factors, the video not being streamed at the same time, etc.

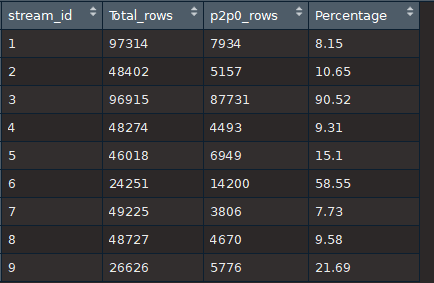
* Each isp seems to have a reasonably high percentage of p2p0 rows other than **‘Arange’** isp. Let us analyse further the distribution of ‘Arange’ isp for each **‘stream id**’ where p2p = 0.

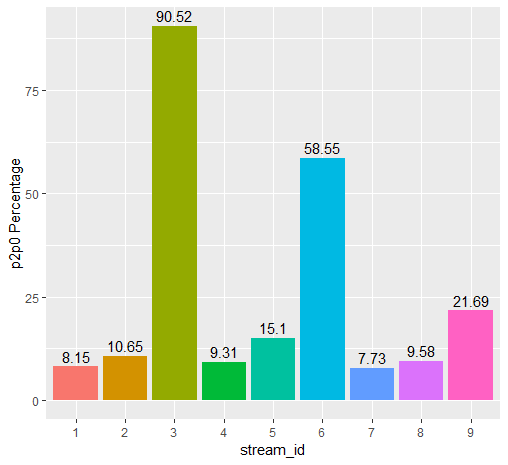


Interestingly, 40781 rows out of 57887 rows of ‘Arange’ for p2p = 0 (over 70%) belong to ‘stream 3’.

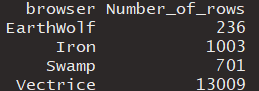
Let us analyse ‘stream id’ data now.

**c) stream id**





* ‘**Stream\_id 3**’ has a huge 90.52% of p2p = 0 rows; it performs poorly in general. Therefore, the high percentage of **‘Arange’** isp can be attributed to this and can be ignored.
* The only other stream having a notably high percentage is ‘**stream\_id 6**’ (58.55%). Here is the distribution of **‘stream\_id 6’** where p2p = 0, factored by **browsers** –



As you can see, 13009 rows out of 14200 of **‘stream\_id 6**’ (over 91%) belong to **‘Vectrice’** browser - which we have already established performs poorly regardless of other variables. Therefore, this high percentage for ‘stream\_id 6’ is ignored.

**RESULTS OF PART 1.**

* The browser **‘Vectrice’** performs poorly in almost each situation regardless of other factors.
* ‘**Stream\_id 3**’ performs poorly regardless of other variables.

2) Where ‘p2p\_cdn’ is greater than 0.

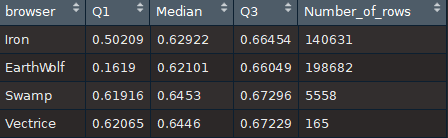
The column ‘p2p\_cdn’ formulated as (p2p/p2p+cdn). The quartile values for each entity of ‘p2p\_cdn’ are compared and analysed.

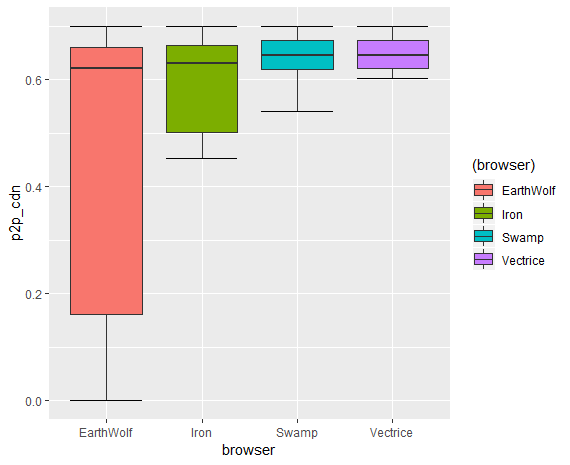
Each data point has the following dimensions :

* Total\_rows - the number of rows corresponding to each entity (i.e a browser, an isp or a stream id)
* Q1 - The first quartile is equal to the 25th percentile of the data.
* Median - The second quartile or median of a data set is equal to the 50th percentile of the data.
* Q3 - The third quartile is equal to the 75th percentile of the data.

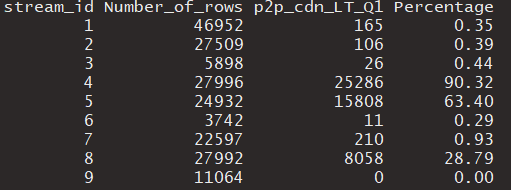
The higher the values of Q1, Median and Q3, better the performance.

**a) browser**





As you can see from the table above, only the browser ‘**EarthWolf**’ has a questionable performance in the lowest 25% of the data (as Q1 value is low). Let’s see the distribution of **‘EarthWolf’** browser factored by **‘stream\_id’** –



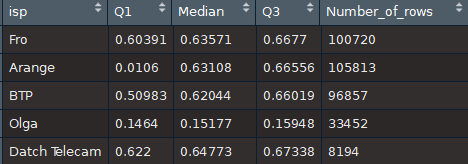
* p2p\_cdn\_LT\_Q1 - Number of rows for which ‘p2p\_cdn’ is less than Q1 (0.1619).
* Percentage - (p2p\_cdn\_LT\_Q1 / Number\_of\_rows)\*100

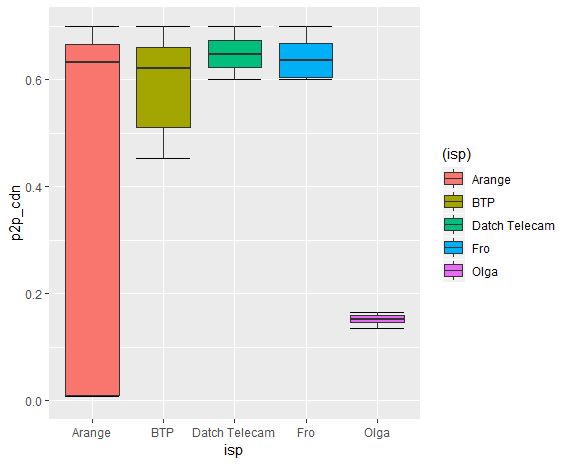
As you can see, **‘stream\_id 4**’ and **‘stream\_id 5**’ have percentages of 90.32% and 63.40%, respectively.

Therefore, the browser **‘EarthWolf’** performs badly for **‘stream\_id 4**’ and **‘stream\_id 5**’.

Please note - Categorising on the basis of ‘**isp**’ yields no meaningful result.

**b) isp**





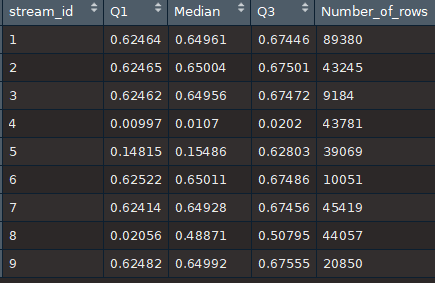
* ‘**Arange**’ has a poor Q1 value. Let’s see the distribution of **‘EarthWolf’** browser factored by **‘stream\_id’.** Following are the values for ‘**Arange**’ isp for ‘p2p\_cdn’ less than equal to Q1.

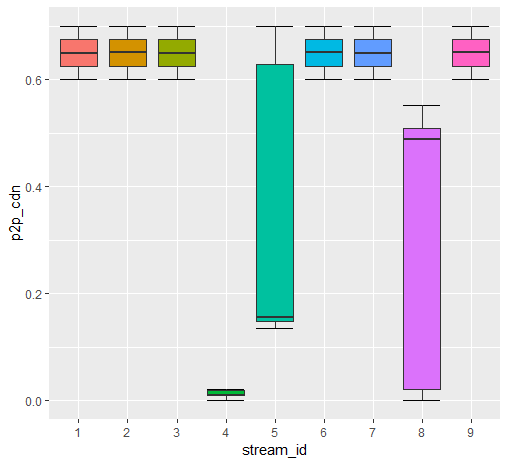


Values exist only for **‘stream\_id 4**’ and ‘**stream\_id 8**’. Therefore, ‘Arange’ isp works badly only for these two streams.

* ‘**Olga**’ isp has poor values all the way up to Q3. Therefore, it performs badly no matter what.

**c) stream\_id**





* ‘**Stream\_id 4**’ performs poorly in every scenario regardless of other factors (low value of Q3).
* ‘**Stream\_id 5**’ has low values of Q1 and Median. Below, we can see that ALL the values less than equal to Median (0.15486) belong to the ‘**Olga’** isp.



Since we have already established that ‘**Olga**’ isp does not work with p2p downloading, low values of Q1 and Median can be excused.

* ‘**Stream\_id 8**’, as we have already established, has low values only for the ‘**Arange**’ isp.

**RESULTS OF PART 2.**

* **‘Stream\_id 4’** performs poorly no matter what the other variables are.
* **‘Olga’** isp works poorly regardless of other factors.
* **‘EarthWolf’** browser has acceptable performance other than when it is streaming ‘**Stream\_id 5**’.
* **‘Arange’** isp works badly only with ‘**Stream\_id 8’**.

**Description of what I would have done if I had more time -**

If provided with the timestamps and locations of every streaming session, I believe a more thorough analysis can be done. As for p2p to occur, the same stream has to be viewed simultaneously by multiple viewers within a certain spatial distance.