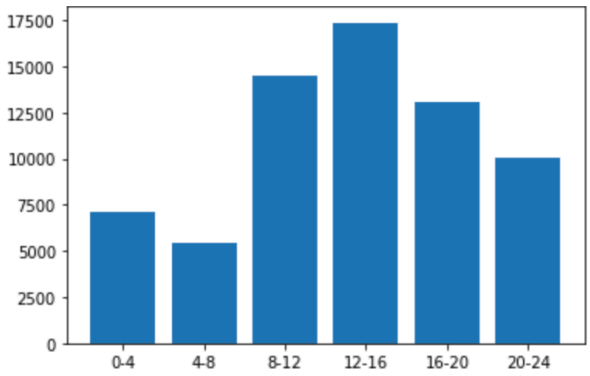
Scalable Machine Learning Assignment 1 Report

Question 1. Log Mining

The whole Log file have 1891715 lines but valid request (with time and date)number is 1891714, this means there is 1 missing line in the log file. The number of existing days in July which have requests is 28, so we divide total requests in 6 slot-hour by 28 and get the results:

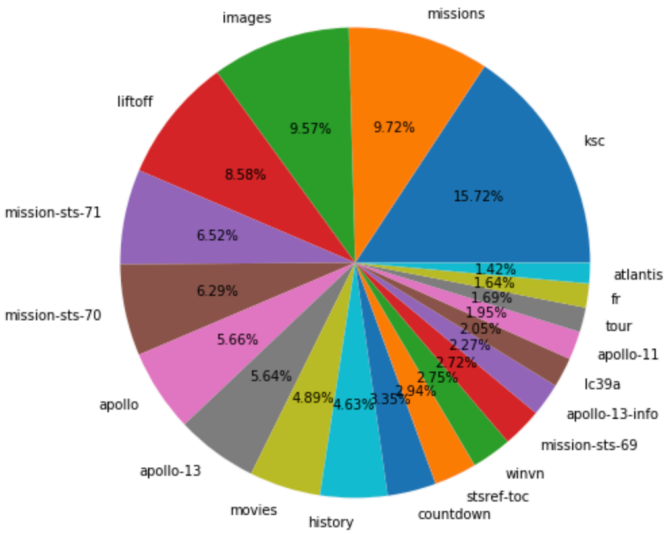
[7078.964285714285,
5479.392857142857,
14462.357142857143,
17377.785714285714,
13096.57142857143,
10066.142857142857] and here is the bar graph:



Observations:

1. From the bar graph we can see that most requests are happened on 12-16 time-slot and the least requests are happened on 4-8 time-slot.
2. The number of requests is rising from 4- 16 and it goes down from 16 to the second day of 4.

The table on the right is Top-20 popular html files and their counts of request. Here is the bie chart of these data:



file	count
ksc.html	40317
missions.html	24921
images.html	24536
liftoff.html	22012
mission-sts-71.html	16736
mission-sts-70.html	16136
apollo.html	14527
apollo-13.html	14457
movies.html	12538
history.html	11873
countdown.html	8586
stsref-toc.html	7538
winvn.html	7043
mission-sts-69.html	6987
apollo-13-info.html	5833
lc39a.html	5263
apollo-11.html	5014
tour.html	4322
fr.html	4219
atlantis.html	3640

only showing top 20 rows

Observations:
1.The most popular file is ksc.html, and it makes up 15.72% of the whole counts.
2. There are many counts of html files which are less than 5%, and these counts of files account for more than 1/4 of the whole count.

Question 2 Movie Recommendation

We use 3-fold validation to get three models for each of the three ALS with different parameter settings. For each model we evaluate the Rmse and Mae values and calculate the mean and std for these three models.Here is the result table below.

Q2A TABLE

	RMSE	MAE	RMSE_mean	MAE_mean	RMSE_std	MAE_std
FIRST_ALS. 1 slot	0.80406	0.61933	0.80440	0.61969	4.7462E-04	5.001E-04
FIRST_ALS. 2 slot	0.80419	0.61947				
FIRST_ALS. 3 slot	0.80492	0.62026				
SECOND_ALS. 1 slot	0.81948	0.62965	0.82012	0.63021	0.001232	0.001112
SECOND_ALS. 2 slot	0.81934	0.62949				
SECOND_ALS. 3 slot	0.82154	0.63149				
THIRD_ALS. 1 slot	0.88410	0.69734	0.88417	0.69738	1.159E-04	4.6066E-04
THIRD_ALS. 2 slot	0.88410	0.69738				
THIRD_ALS. 3 slot	0.88430	0.69743				

Observation:
1.For each ALS, the RMSE and MAE of three model are quite similar, but for different setting ALS, the RMSE and MAE are different.
2. Both std values of RMSE and MAE are very small.

For the first ALS, we need to use k-means to cluster the movie factors for three model.
Here is the first model result:

```
+----------+-----+
|prediction|count|
+----------+-----+
|        19| 4180|
|         3| 3682|
|        18| 3525|
+----------+-----+
```

This is the top-3 largest cluster. For each cluster, we need to get top3 tags.

After sum up the relevance scores, we get top-3 tags for cluster 19 :

```
+-----+------------------+
|tagId|   sum(relevance)|
+-----+------------------+
|  742|47.11100000000001|
|  646|          32.21825|
|  867|30.56274999999999|
+-----+------------------+
```

Here are the tag name with corresponding tagId below:

```
.-----.--------.
|tagId|     tag|
+-----+--------+
|  742|original|
+-----+--------+
```

```
+-----+------+
|tagId|   tag|
+-----+------+
|  646|mentor|
+-----+------+
```

```
+-----+-------+
|tagId|    tag|
+-----+-------+
|  867|runaway|
+-----+-------+
```

And the number of movie for each tagId:

```
number of movie when tagId = 742 66
number of movie when tagId = 646 66
number of movie when tagId = 867 66
```

This above is the whole process of the question 2C. I will give the result directly.

Split 1:
Cluster 19: tagid 742(num of movie: 66) tagid 646(num of movie: 66) tagid 867(num of movie: 66)
Cluster 3: tagid 742(num of movie: 857) tagid 646(num of movie: 857) tagid 323(num of movie: 857)
Cluster 18: tagid 742(num of movie: 19) tagid 807(num of movie: 19) tagid 646(num of movie: 19)

Split 2:
Cluster 1: tagid 742(num of movie: 927) tagid 646(num of movie: 927) tagid 195(num of movie: 927)
Cluster 4: tagid 742(num of movie: 92) tagid 646(num of movie: 92) tagid 867(num of movie: 92)
Cluster 5: tagid 742(num of movie: 1727) tagid 270(num of movie: 1727) tagid 1008(num of movie: 1727)

Split 3:
Cluster 17: tagid 742(num of movie: 1168) tagid 646(num of movie: 1168) tagid 195(num of movie: 1168)
Cluster 8: tagid 742(num of movie: 1513) tagid 972(num of movie: 1513) tagid 1104(num of movie: 1513)
Cluster 5: tagid 742(num of movie: 123) tagid 646(num of movie: 123) tagid 270(num of movie: 123)

Observations:
1.For each split and each cluster, the largest tagid are always 742. The tagid 646 also appears frequently
2.In each cluster, the different tagids have the same number of movies

tagId	tag
742	original

tagId	tag
646	mentor

tagId	tag
867	runaway