Estimating transfer times of large datasets for scientific computing

IRIS -HEP Fellow - Oleksii Brovarnyk

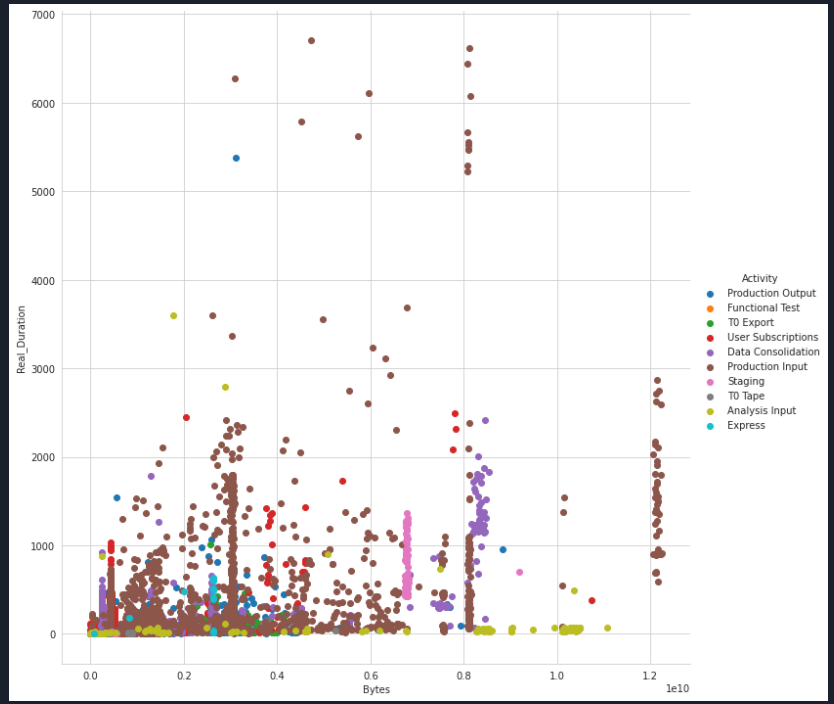
Mentor - Mario Lassnig (CERN)

Project description

This project will continue the existing research of the Rucio team on **the estimation of the duration of file transfers** for large scale sciences. The distributed data management environment for scientific experiments forms a complex ecosystem with dynamic interactions between users and data centers. Rucio's central role as the data management system, and the rich amount of data gathered about the transfers and data rules life cycles will help in creating machine learning for transfer time estimation.

Project description

To start, the research needs data on successful file transfers. This is based on the Rucio events **(event_type: transfer-done)** which are available in the Rucio Elasticsearch instance. The idea is then to generate **time series** from this data, including event metadata like **"started_at"** (when did the transfer start), **"transferred_at"** (when did the transfer finish), **"bytes"** (how big was the file), and many more. There are more than 30 different variables that can be used for this model.



What was used to work

- Google Colaboratory
- Python
- TensorFlow
- Keras

Data analysis

Previously, I removed unnecessary columns, such as _type, _id, _score and others, since they do not represent any informational value.
Left the following columns: **'Account', 'Activity', 'Scope', 'Dst_Rse', 'Src_Rse', 'Bytes', 'Started_At', 'Transferred_At', 'Created_At', 'Submitted_At'** , because when checking the correlation, these values showed good results.
Calculated these values: **'Transfer_Duration', 'Queue_Duration'** , because all files are first queued before being sent, in which they spend a long time.

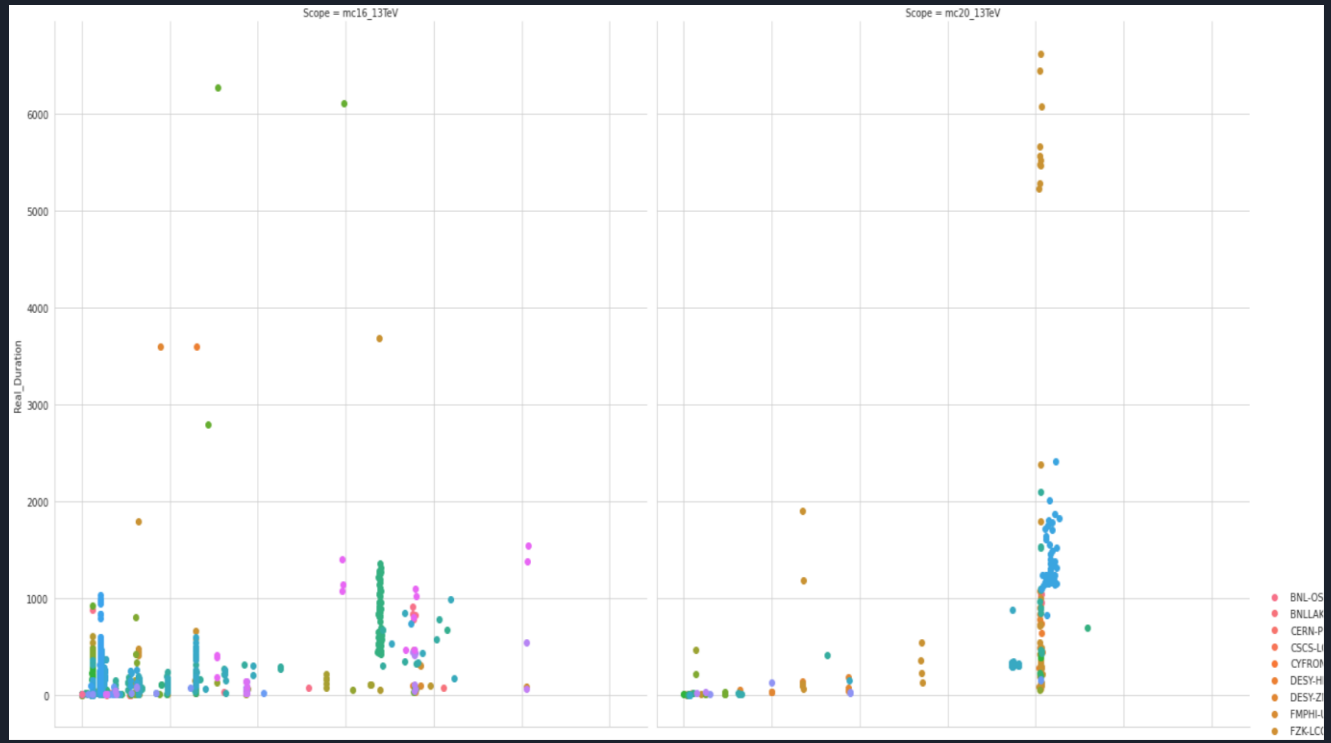
Data correlation

	Bytes	Transfer_Duration	Sec_Created_At	Sec_Submitted_At	Sec_Started_At	Sec_Transferred_At	Queue_Duration	index
Bytes	1.000000	0.362268	0.068922	0.069530	0.071481	0.071541	0.095818	-0.098391
Transfer_Duration	0.362268	1.000000	0.036627	0.037057	0.039451	0.039618	0.115893	-0.050941
Sec_Created_At	0.068922	0.036627	1.000000	0.999701	0.999481	0.999481	0.025225	0.127817
Sec_Submitted_At	0.069530	0.037057	0.999701	1.000000	0.999782	0.999781	0.025291	0.127858
Sec_Started_At	0.071481	0.039451	0.999481	0.999782	1.000000	1.000000	0.046170	0.128766
Sec_Transferred_At	0.071541	0.039618	0.999481	0.999781	1.000000	1.000000	0.046189	0.128757
Queue_Duration	0.095818	0.115893	0.025225	0.025291	0.046170	0.046189	1.000000	0.048011
index	-0.098391	-0.050941	0.127817	0.127858	0.128766	0.128757	0.048011	1.000000

Data analysis

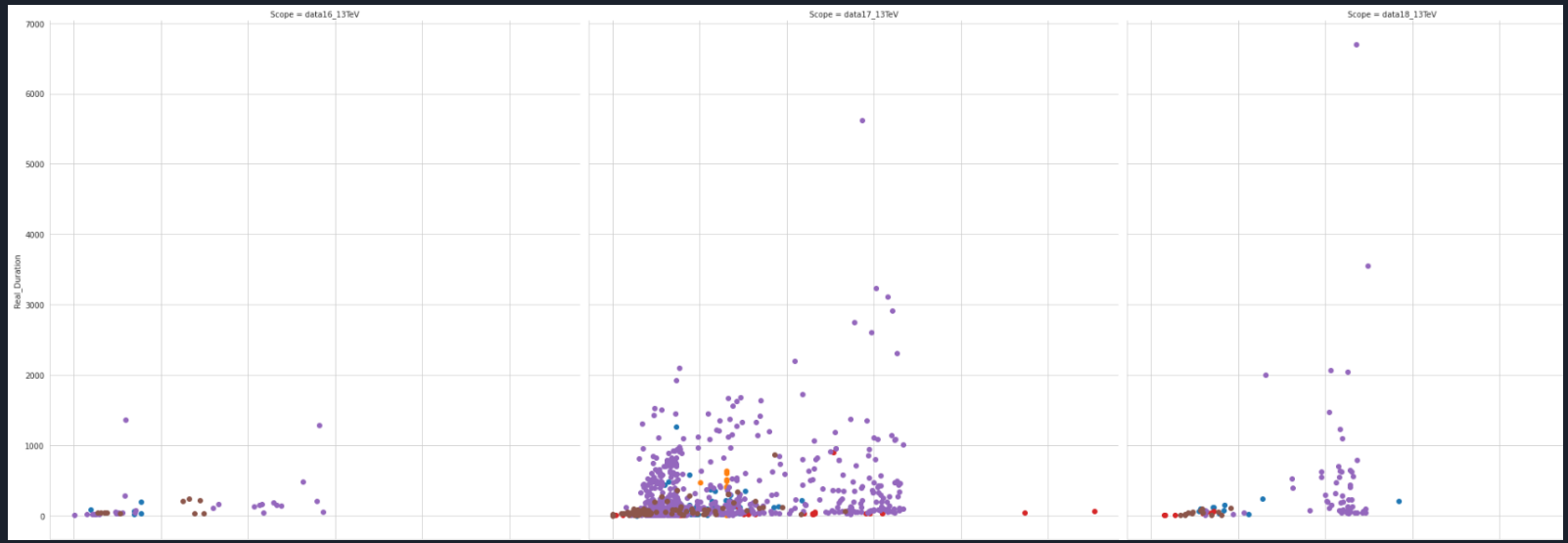
We used a random sample of **10,000** records to analyze the data, and from time to time I took different samples to see if the results were the same for the other samples.

The diagram shows that **'Scope'** affects the transmission time, and the names containing substrings 'mc{some number}' show the worst result.



Data analysis

A scatter plot of **Bytes** vs. **Real_Duration** for other **‘Scope’** . As you can see in the picture, there is a correlation. Colors correspond to different **"Activity".**
And so I checked all the columns and saw how much they affect the file transfer time.
As a result, I came to the conclusion that only **7 columns** with data should be left.

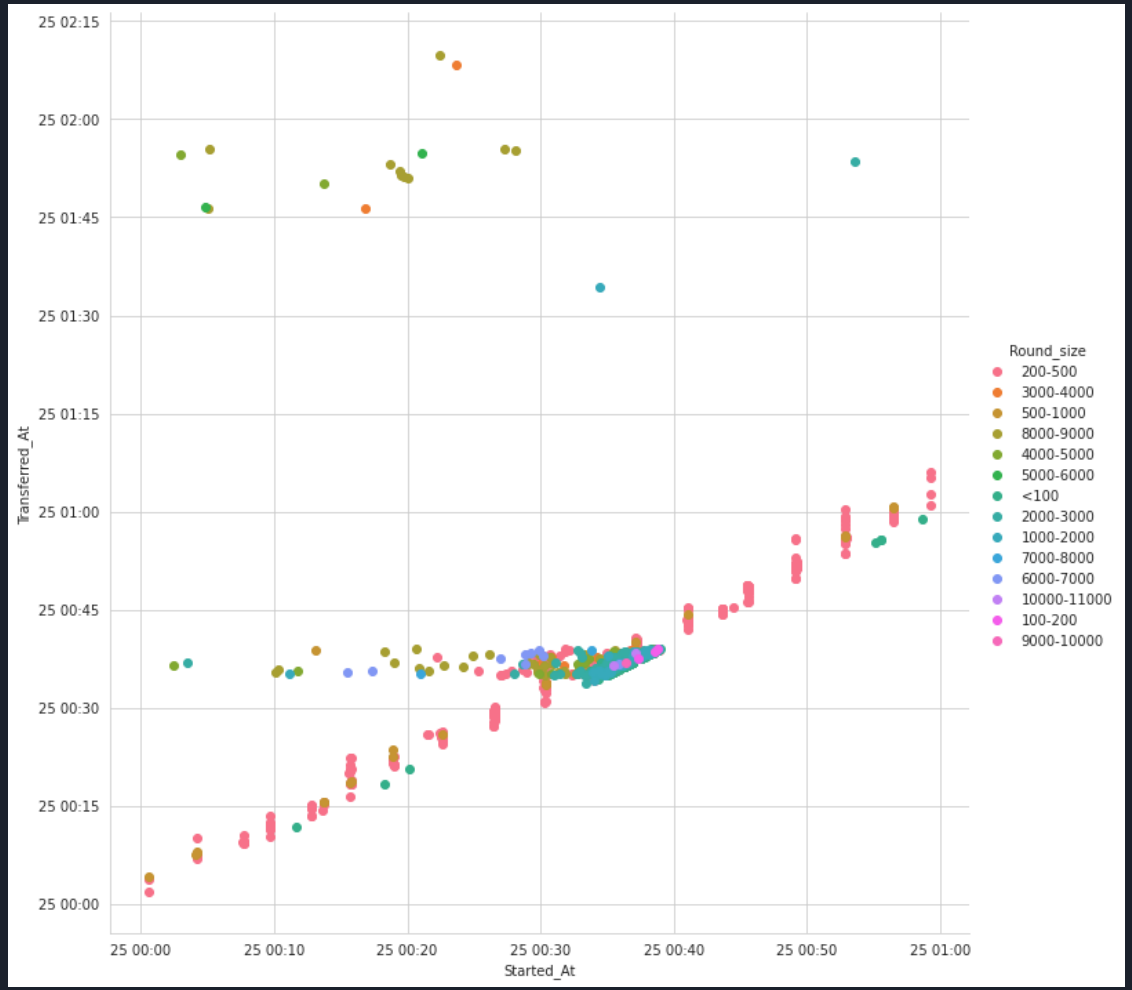


Data analysis

Also considered the correlation between **Started_At** and **Tranfarred_At**, divided file sizes into categories (<100Mb, 100-200Mb, ...) and displayed on the dot plot. This is done to see the relationship between file size and transfer time.

Round_size	Created_At	Checks
100-200	23	
1000-2000	673	
10000-11000	27	
11000-12000	1	
12000-13000	46	
200-500	4534	
2000-3000	491	
3000-4000	550	
4000-5000	127	
500-1000	621	
5000-6000	88	
6000-7000	158	
7000-8000	84	
8000-9000	156	
9000-10000	12	
<100	2399	

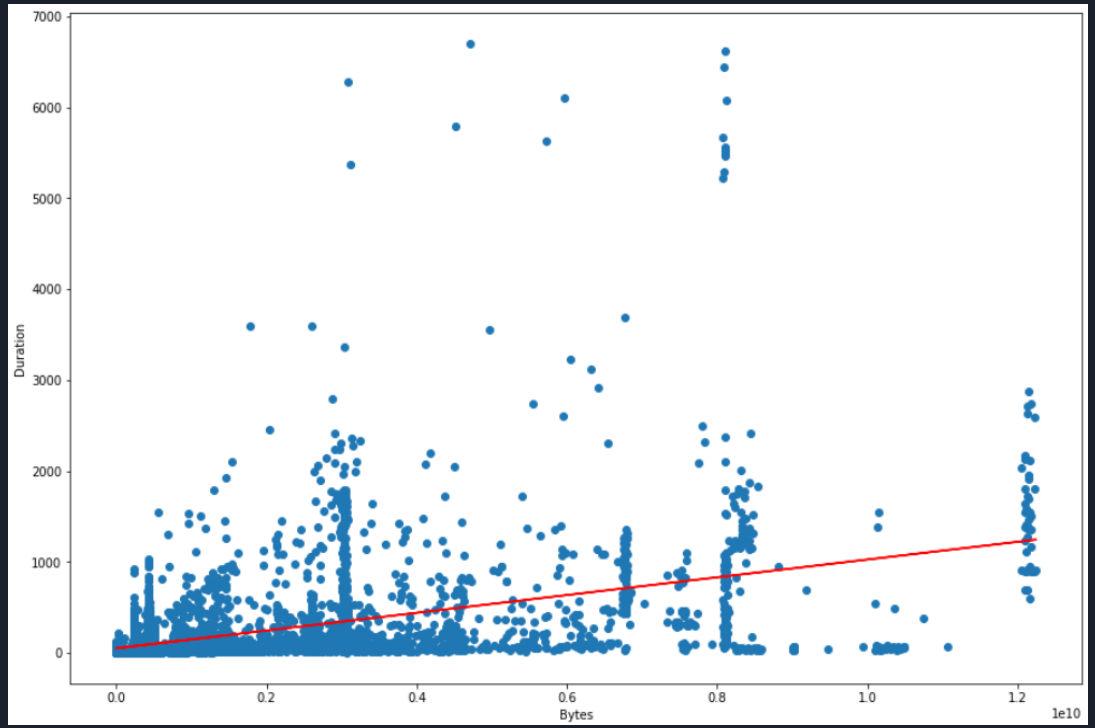
16 rows × 33 columns



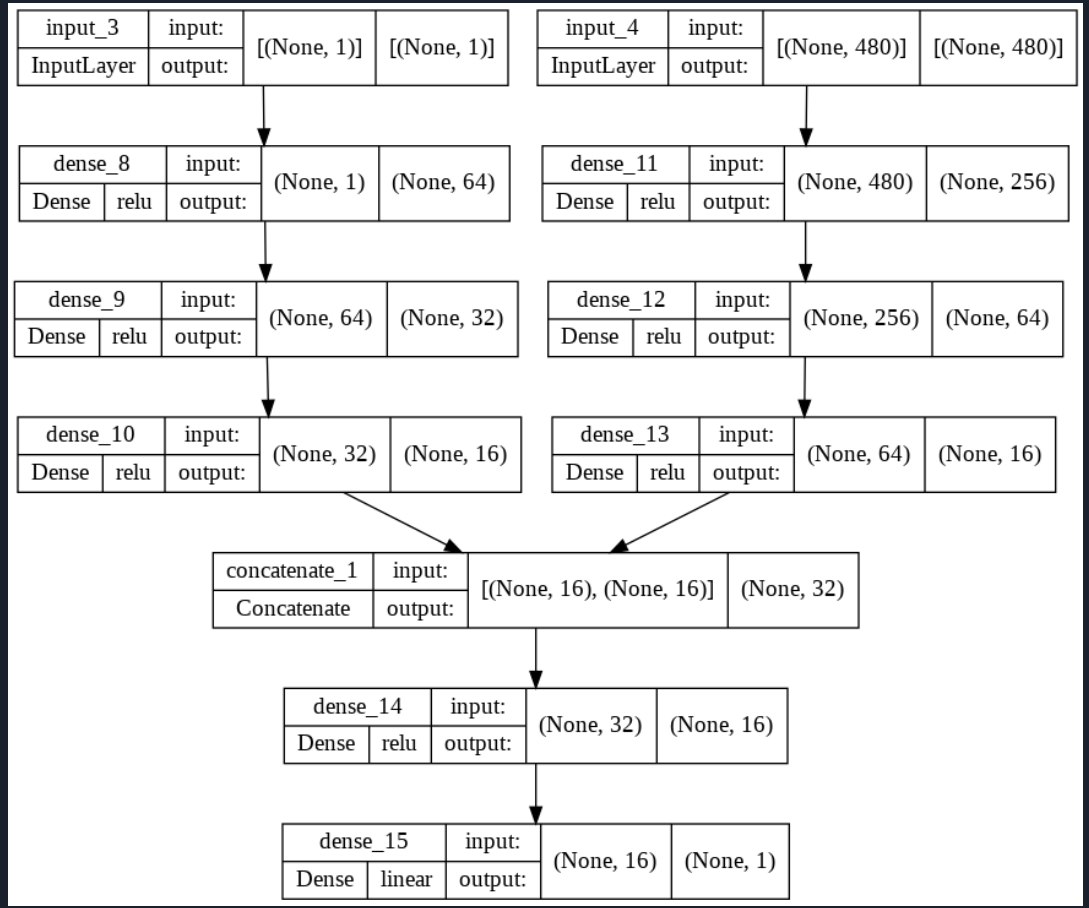
This slide shows the enlarged area that was written about in slide 8. Here you can see that the files are transferred in small groups.

Linear regression

To start the estimation, the inputs for a first model was the **number and size of files**, and the outputs was the **duration in seconds**. From this, a first **linear regression** model was learned, which should answer the question: For a given dataset (number of files, and gigabytes), **how long will it take for this dataset to finish transferring**.



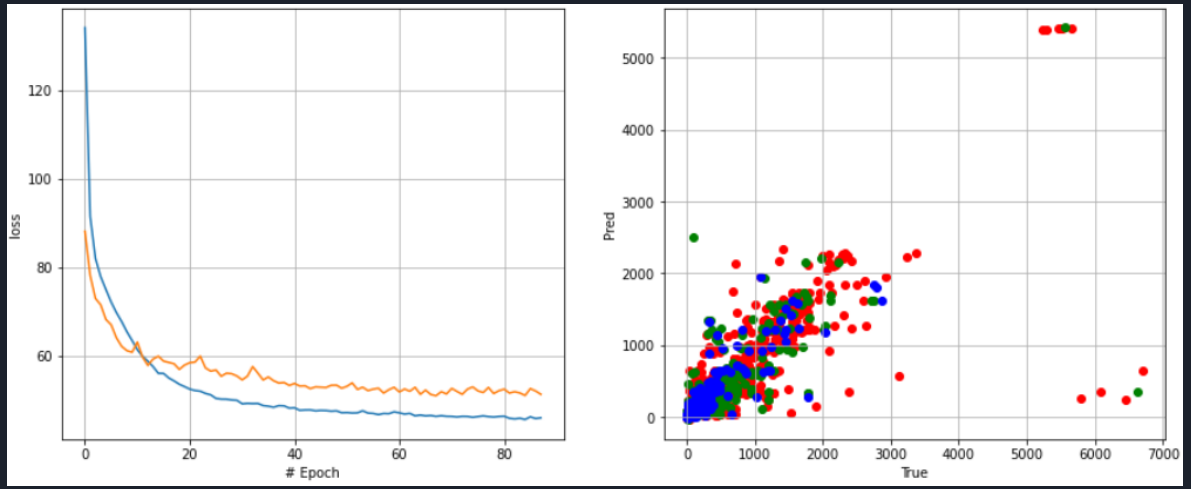
Model



Since we have both numeric data (**'Bites'**) and categorical data (**'Account', 'Activity', 'Scope', 'Src_Rse', 'Dst_Rse'**), we create two branches for these different data and then merge them.

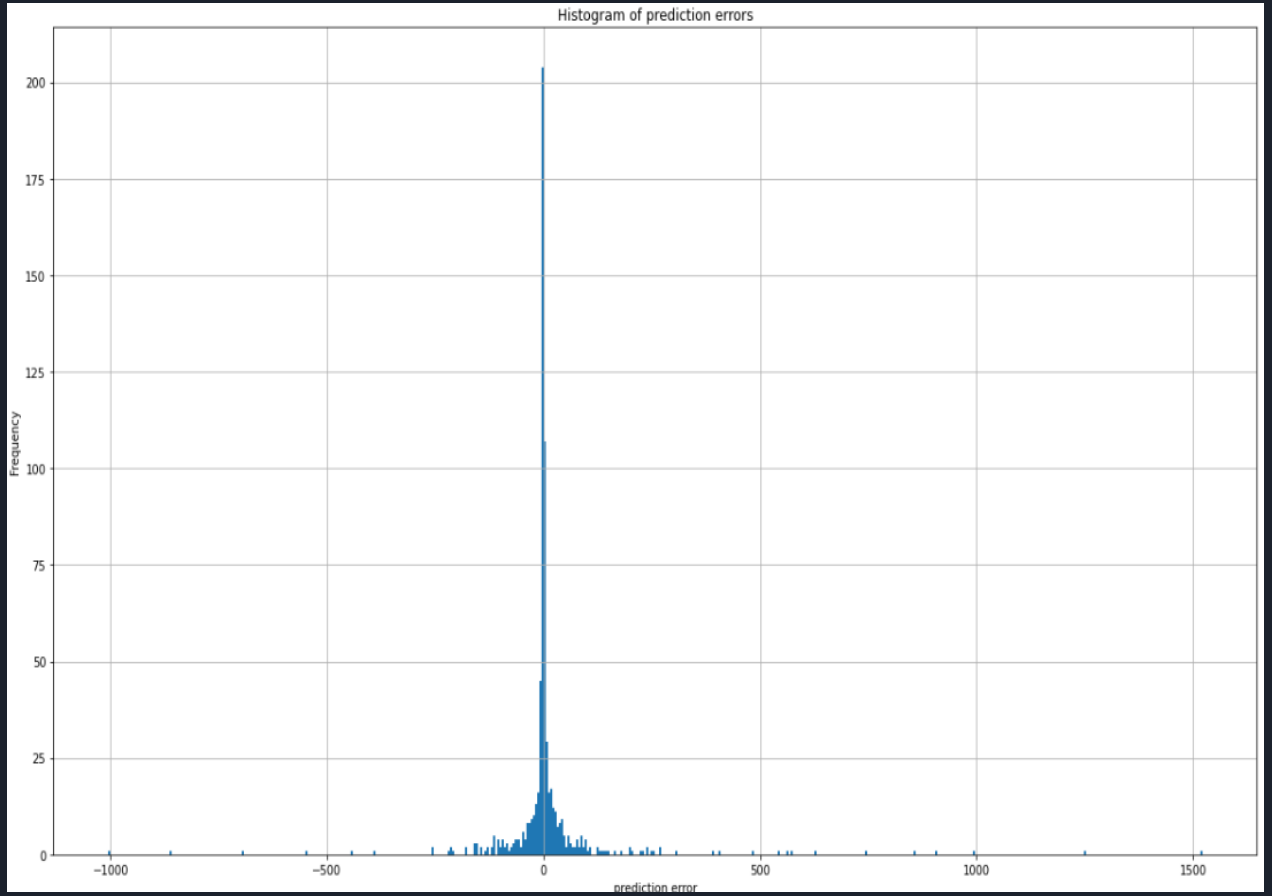
Predicted values

The **RMSE metric** is quite large and you should strive to decrease this indicator, although **R2_score** is quite high, also on the scatter plot you can see how much the training data differ from the validation and test data (the closer they are to the diagonal the better, blue dots - test, green - val, red - train).




```
R² val: 0.6556456197406808
R² train: 0.7526150251051504
R² test: 0.7991679321950043
Max_error : 1521.350313964844
MAE : 47.89690371285969
RMSE: 143.52553572080794
```

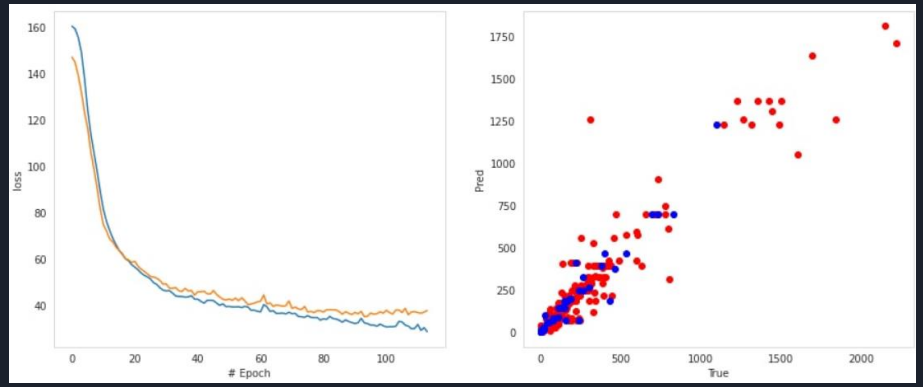
Predicted values

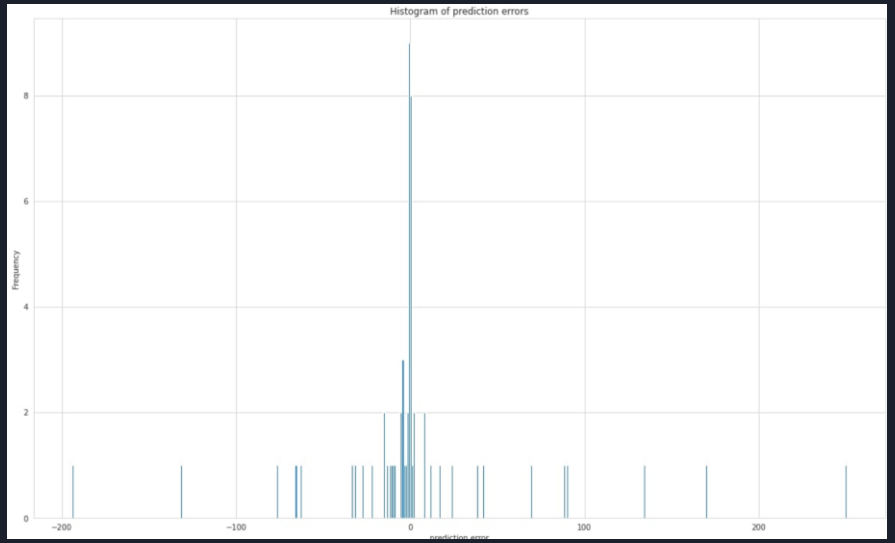


Here is the error histogram for the model and the values from the previous slides. You can see that most of the errors are within zero, but there are a small number of outliers.

Predicted values

Predictions using a model that was trained on a grouped dataframe (first found the most popular file size by category, and then took the average in this category). Grouped by 10 records.



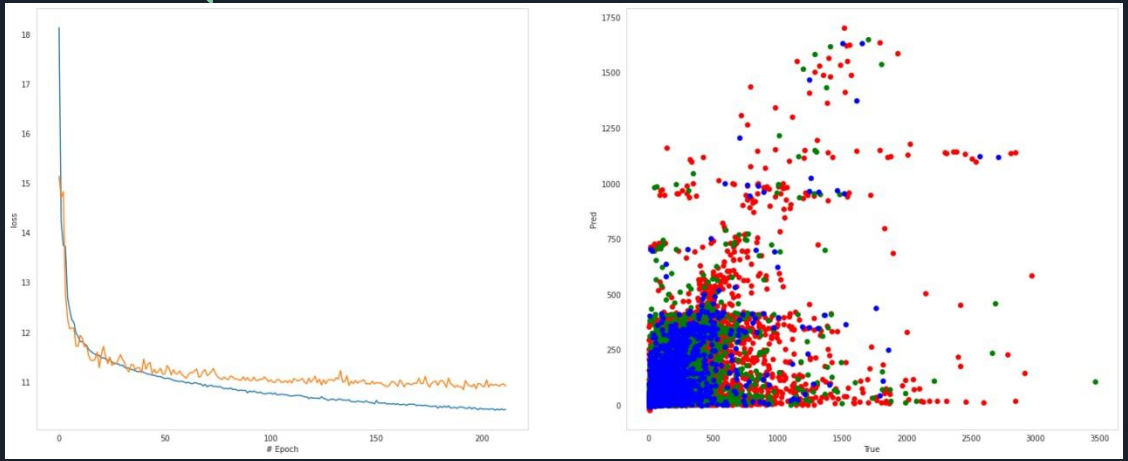


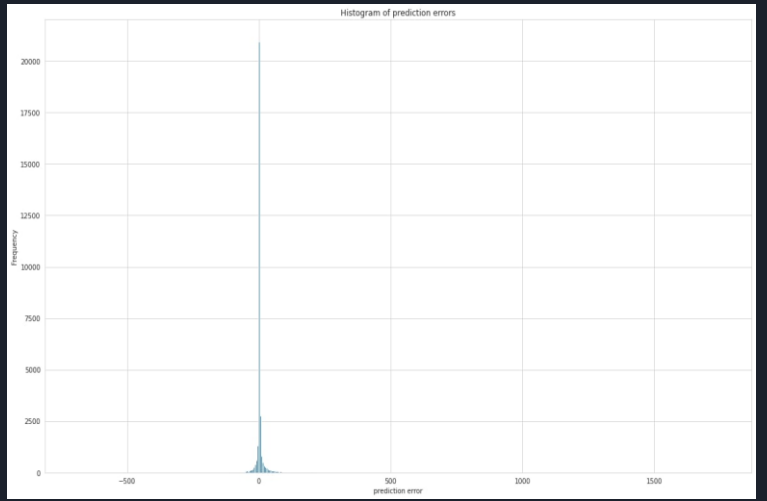

```
R² val: 0.8401741134757792
R² train: 0.9151332154164538
R² test: 0.931934727335838
Max_error : 250.3335096086775
MAE : 29.715935982339452
RMSE: 59.90611658150669
```

The result turned out to be quite good, the loss plot for **training** and **validation** data almos coincides, and the red and blue dots on the plot are located near the diagonal line, which indicates that the prediction errors are insignificant.

Predicted values

Let's test the model on a large dataset (338 thousand records).



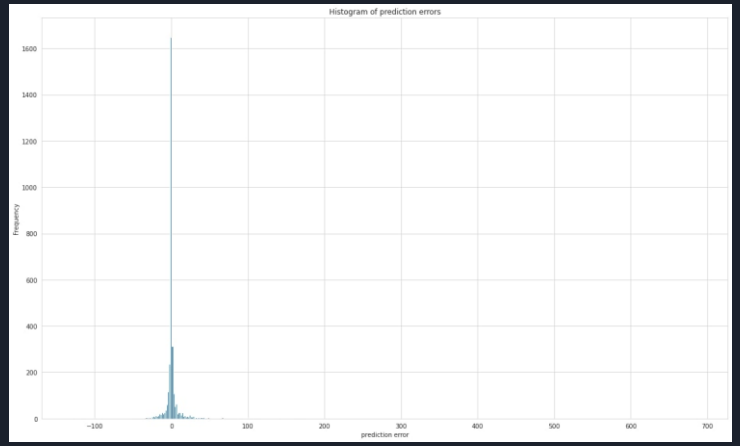


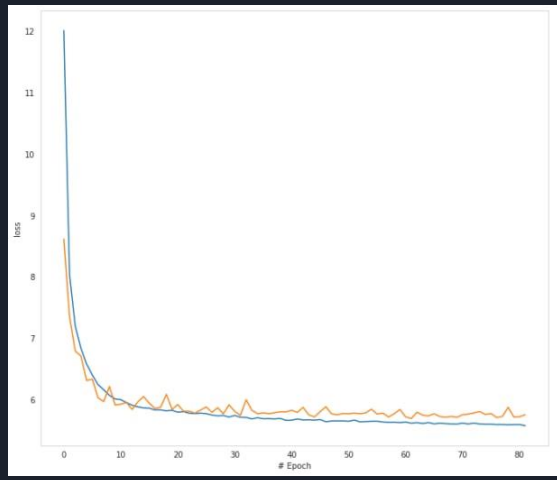

```
R² val: 0.4470257633303709
R² train: 0.4909577898784593
R² test: 0.5449594016672721
test Max_error : 1747.4946250915527
test MAE : 10.674808157065222
test RMSE: 49.03234928289595
train RMSE: 51.162276742608306
test NRMSE(max-min): 0.01807311068296939
train NRMSE(max-min): 0.017226355805592022
```

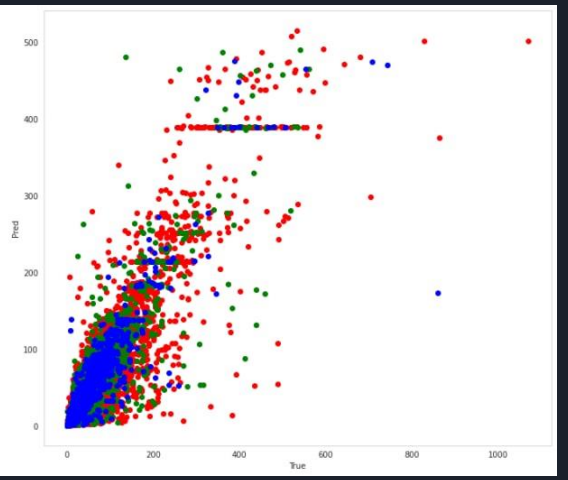
You can see that the normalized RMSE metric is about 2 percent which means that the average prediction error is about 2 percent.

Predicted values

And I will also check the results for the grouped files, made groups of 10 files.







The result is quite good, which means that the model works well with a large amount of data.

```
R² val: 0.8586866228754142
R² train: 0.8703917427272838
R² test: 0.844011185378865
Max_error : 686.1390686035156
MAE : 5.642756743649383
test RMSE: 20.324820611716643
train RMSE: 17.763082370967297
test NRMSE(max-min): 0.023636260741617215
train NRMSE(max-min): 0.01659790914872668
```

Summary

- I really enjoyed participating in this project, I gained a lot of knowledge, gained experience with the real challenge.
- This is a very important event in my life.
- I realized that most of the time is spent on understanding and analyzing data.
- Since this is my first big challenge in this direction, I was worried that it would be difficult, but I realized that I want to continue to develop my skills in this direction.

Thanks for listening!