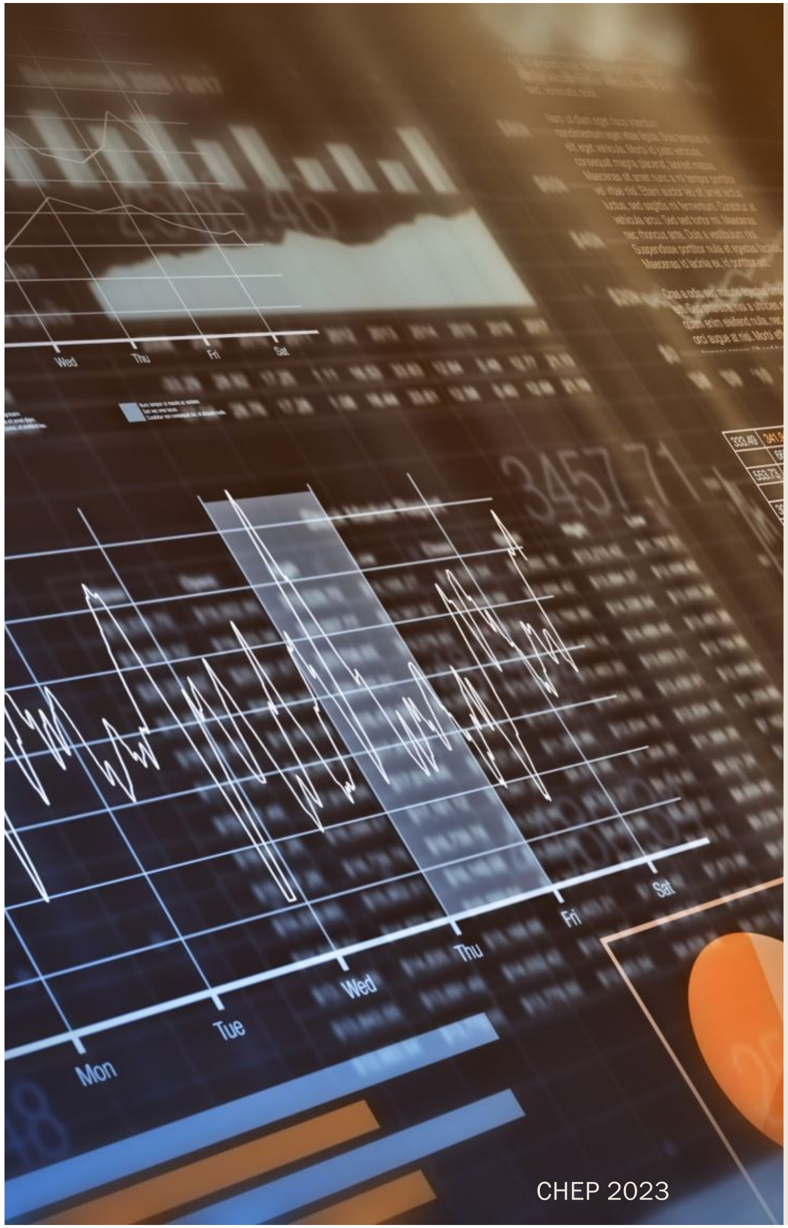
OPERATIONAL ANALYTICS STUDIES FOR ATLAS DISTRIBUTED COMPUTING: DATA POPULARITY FORECAST AND UTILIZATION OF THE WLCG CENTERS

Maria Grigorieva and Alexei Klimentov, on behalf of the ATLAS Computing Activity

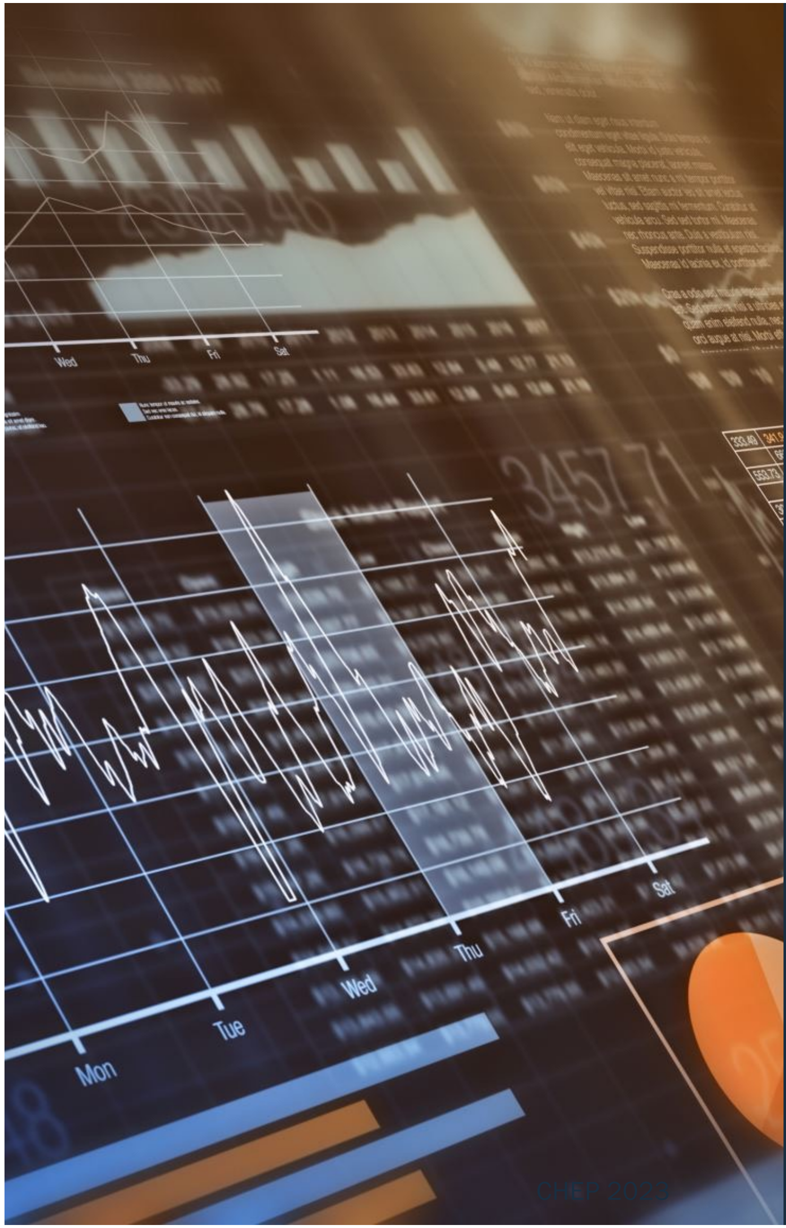
Lomonosov Moscow State University

Brookhaven National Laboratory



Outline

- Operational Analytics
- WLCG Centers and PanDA queues utilization
- ATLAS Data Popularity Studies using Neural Network, decomposition and Naïve Forecast algorithms
- Conclusion



Operational Analytics

- **Operational analytics** (commonly referred to as operational intelligence) -- is the practice of utilizing data in real-time to make instant decisions in business operations.
- This talk is focused on one of several ongoing **ATLAS Operational Analytics Projects:**
  - *Analysis of the current state of workflows in order to anticipate imbalances and take timely measures to stabilize the distributed environment*

WLCG Centers and Payload Brokerage

- The WLCG has more than 170 sites running millions of payloads daily
- ATLAS Workload Management System (WMS) – PanDA – is responsible for job brokering, execution and load balancing in the distributed and heterogeneous computing environment
  - *ATLAS workflows (except Tier-0) are orchestrated by PanDA*
- HL-LHC → much more data will require more resources → more analysis jobs → more challenges to workload and data management systems
  - *Pseudo-interactive Grid analysis is still to be addressed*
- Evaluation of the utilization of WLCG centers by various metrics will help to find an optimal distribution of data and payloads to multiple computing resources.
  - In this research we address only user analysis payloads

Outline

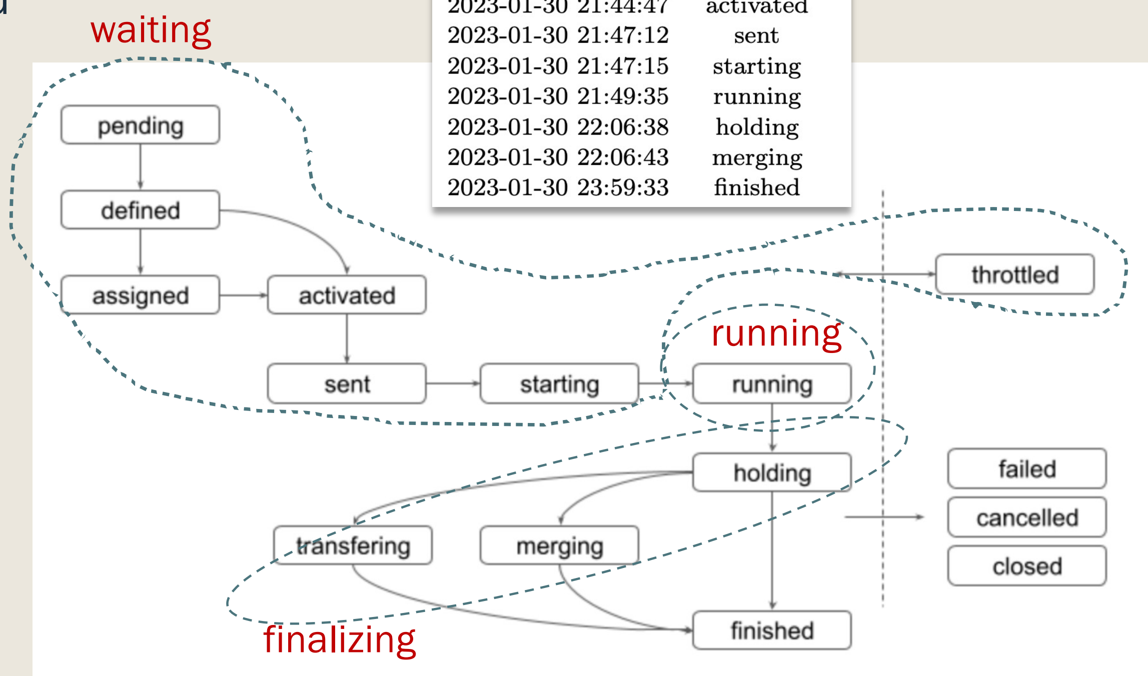
- Operational Analytics
- WLCG Centers and PanDA queues utilization
- ATLAS Data Popularity Studies using Neural Network, decomposition and Naïve Forecast algorithms
- Conclusion

User Analysis Jobs

- For more than a decade, PanDA has maintained a full history of payload execution, computing sites issues, and input/output payload data volume
- Resource unit in PanDA **queue**
- There are three primary job execution phases (states)
  - *Waiting*
  - *Running*
  - *Finalizing*
- Advanced method for calculating jobs within 1-hour intervals was used (described on slides 16-17)

Table 1. Fragment of table JOBS STATUSLOG for one job

DateTime	Status
2023-01-30 21:44:41	pending
2023-01-30 21:44:42	defined
2023-01-30 21:44:47	activated
2023-01-30 21:47:12	sent
2023-01-30 21:47:15	starting
2023-01-30 21:49:35	running
2023-01-30 22:06:38	holding
2023-01-30 22:06:43	merging
2023-01-30 23:59:33	finished



Detailed job states in PanDA database

User Analysis Queues Utilization Metrics I

1. Number of jobs

$$R_1 = \frac{N_{jw}}{N_{jr}}, R_2 = \frac{N_{jr}}{N_{jf}}, U_n = \overline{R_1, R_2}$$

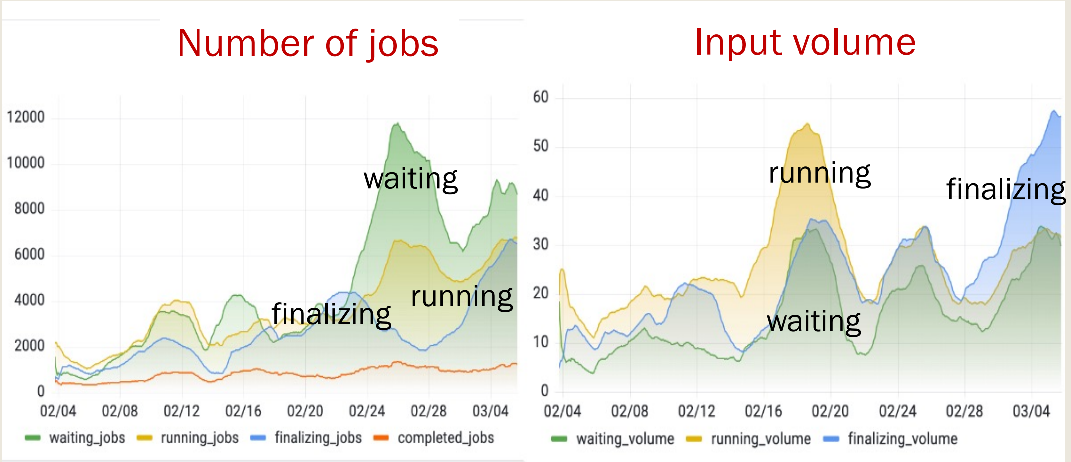
2. Input volume

$$R_1 = \frac{Vol_{jw}}{Vol_{jr}}, R_2 = \frac{Vol_{jr}}{Vol_{jf}}, U_n = \overline{R_1, R_2}$$

3. *Time to complete*

$$U_t = \frac{\overline{T_{jw}} + \overline{T_{jf}}}{\overline{T_{jr}}}$$

Proportion mismatch between the number of jobs and their input volume



"Analysis" Jobs without input: The number of jobs without input ~20%. Total CPU consumption of the non-input jobs ~ 7% (1 month statistics)

jw – waiting jobs, jr – running jobs, jf – finalizing jobs

User Analysis Queues Utilization Metrics II

1. *Number of jobs*

$$R_1 = \frac{N_{jw}}{N_{jr}}, R_2 = \frac{N_{jr}}{N_{jf}}, U_n = \overline{R_1, R_2}$$

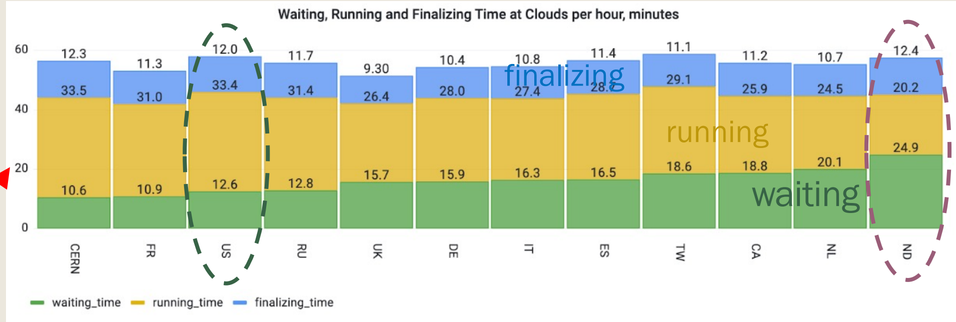
2. *Input volume*

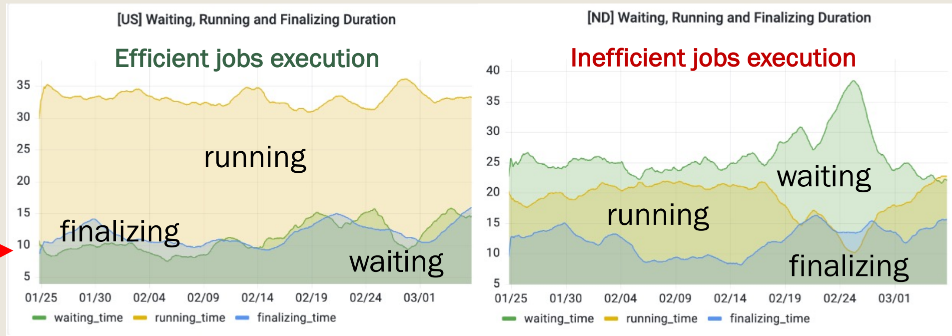
$$R_1 = \frac{Vol_{jw}}{Vol_{jr}}, R_2 = \frac{Vol_{jr}}{Vol_{jf}}, U_n = \overline{R_1, R_2}$$

3. Time to complete

$$U_t = \frac{\overline{T_{jw}} + \overline{T_{jf}}}{\overline{T_{jr}}}$$

The shares of time jobs spend in waiting, running and finalizing states at CERN and national facilities are about 24%, 55% and 21%





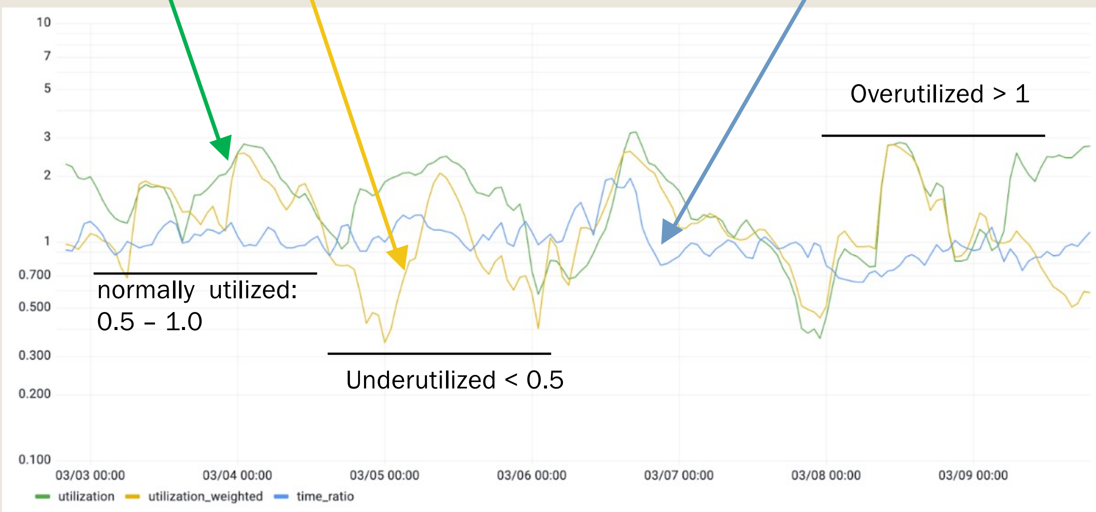
The proportion of waiting, running and finalizing jobs duration at the US and ND clouds over time

jw – waiting jobs, jr – running jobs, jf – finalizing jobs

Comparison between Metrics

Alternative metrics for utilization by the number of jobs (green) and weighted utilization (orange)

Utilization, measured by job status durations , TTC (blue)

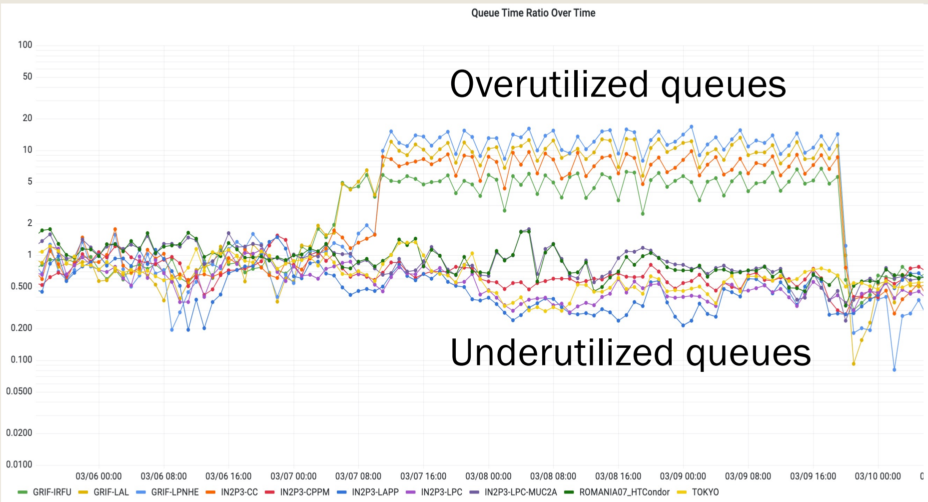


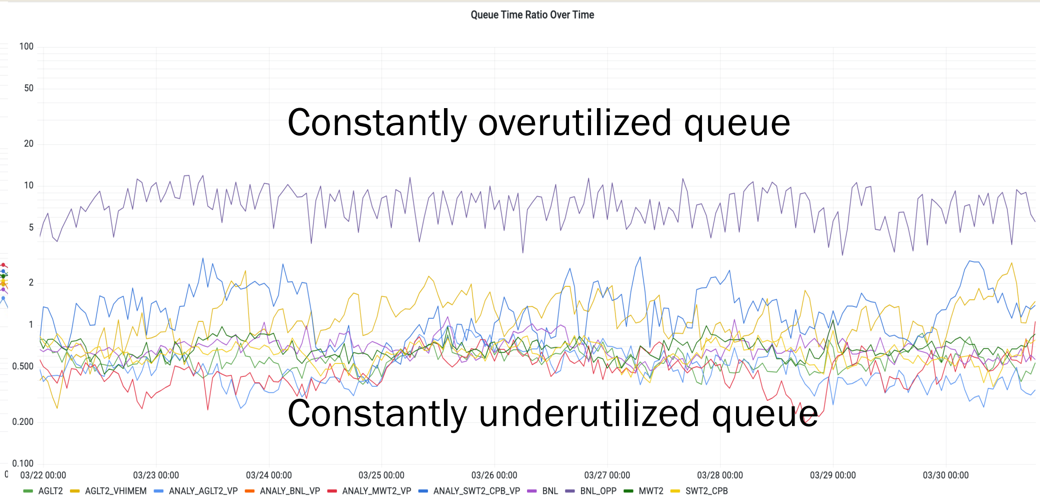
- *Utilization* – utilization by the number of jobs
- *Utilization_weighted* – utilization by the input volume
- *Time_ratio* – utilization by time to complete (TTC)

Time_ratio gives a general result that can be considered averaged. It was chosen as the metric to estimate and to assess queue load.

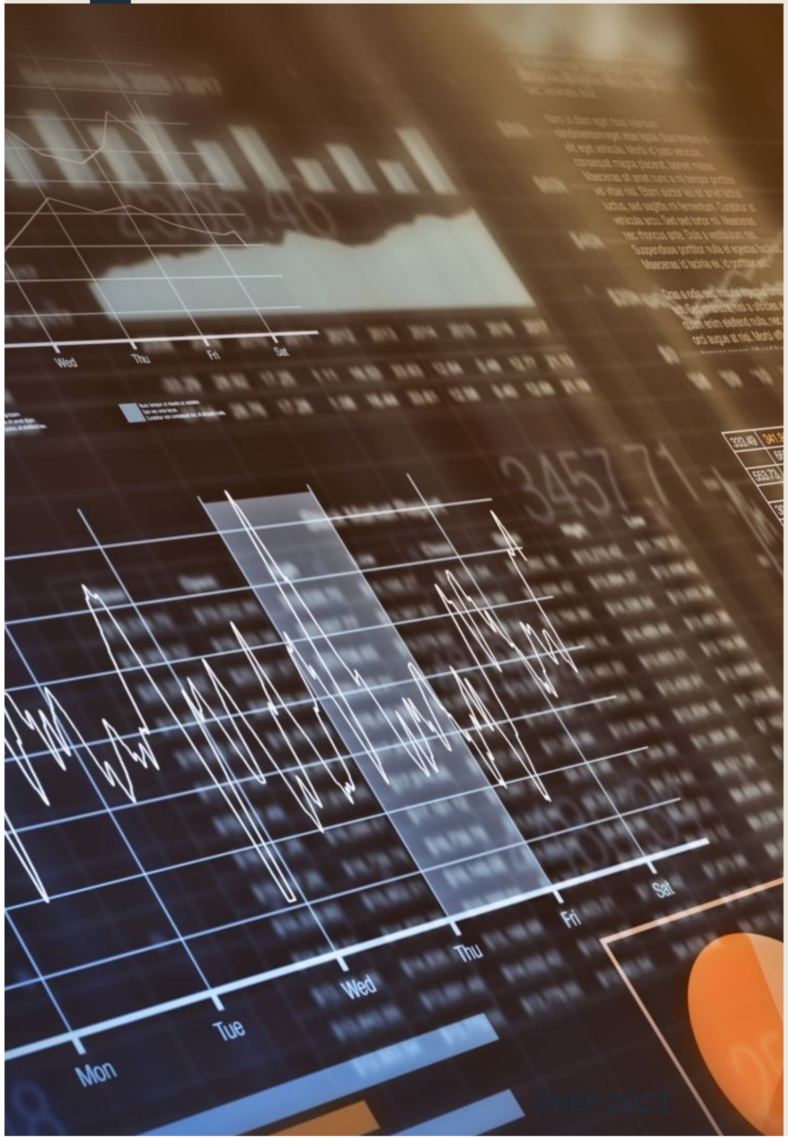
Correlations between the utilization and waiting/running/finalizing duration is shown on Slide 18

PanDA Queues Utilization





Too many jobs were assigned to these queues, that made them overutilized for the next 3 days, while other queues were underutilized.

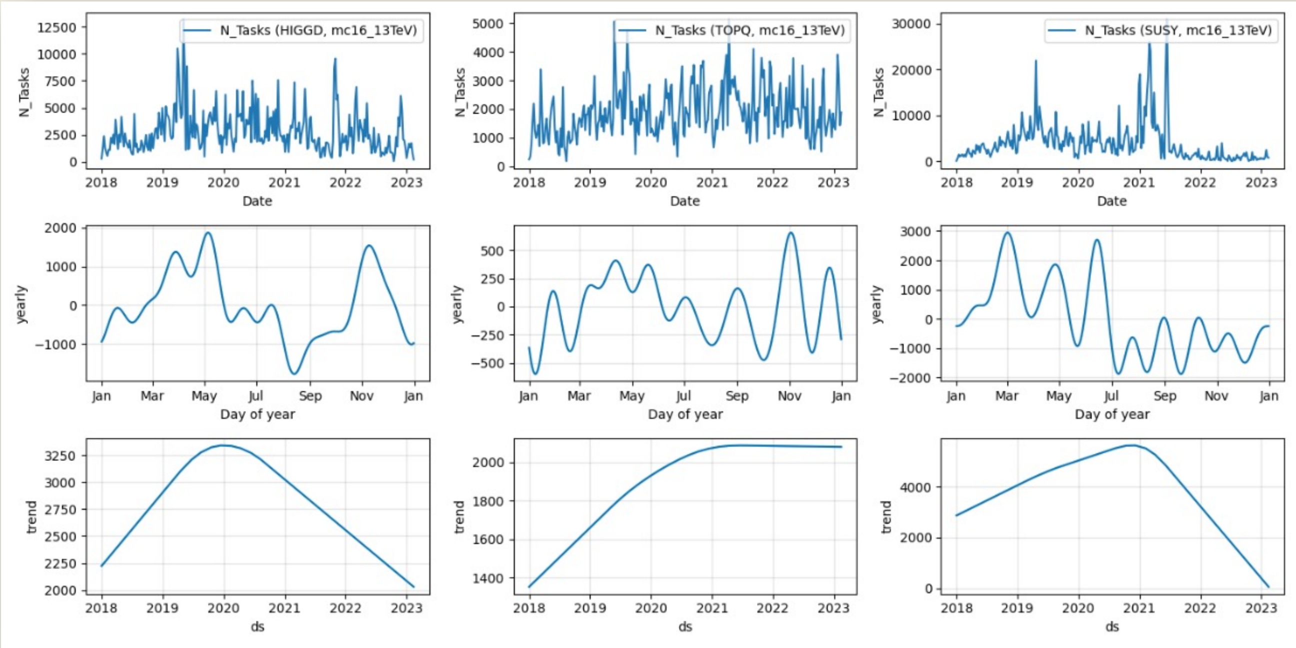


Outline

- Operational Analytics
- WLCG Centers and PanDA queues utilization
- ATLAS Data Popularity Studies using Neural Network, decomposition and Naïve Forecast algorithms
- Conclusion

Datasets Popularity Forecast

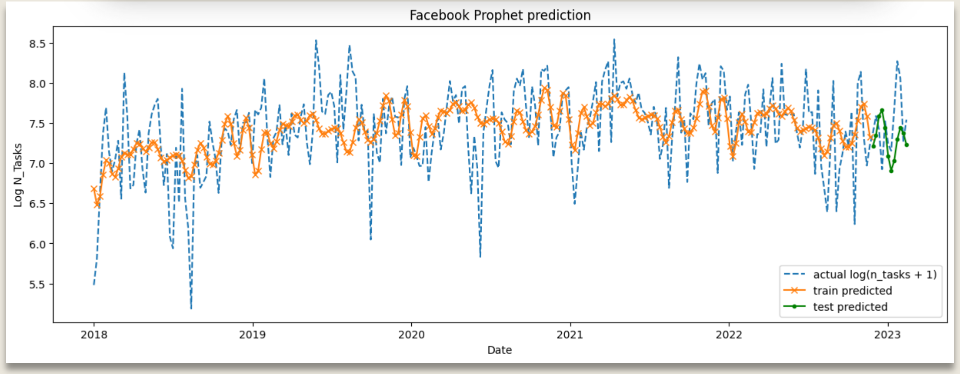
- Datasets[1] were grouped by physics groups pattern : HIGGD, SUSY, TOPQ
- **Group representation:** weekly aggregated number of user analysis tasks
- The seasonality and trends differ for the groups
- Evaluated algorithms for the prediction of the data popularity in future:
  - [LSTM]() (Long short-term memory neural network)
  - [Facebook Prophet]() (decomposition model)
  - Naïve Forecast (**we use the previous period to forecast for the next period)**



[1]Only DAOD (Derived Analysis Object Data), mc16_13TeV datasets were analyzed
ATLAS dataset - group of files taken/produced under the same conditions, dataset - unit of data processing and replication

Forecast Models Comparison

Model	Type of Prediction	# weeks	HIGGD	TOPQ	SUSY
LSTM	One-step	1	7.21%	4.90%	11.68%
	Multi-step	4	7.70%	4.97%	13.81%
		12	7.85%	5.05%	15.03%
	Multi-output	4	8.64%	6.20%	13.05%
		12	8.83%	6.67%	12.78%
Prophet	Multi-output	4	8.16%	7.23%	9.36%
		12	7.67%	6.83%	12.15%
Naive	Multi-output	4	9.85%	6.77%	13.31%
		12	11.21%	8.03%	13.69%



Forecasts performed well only on logarithmically-transformed data, and significantly worse when the data was transformed to a real-world scale.

LSTM and Prophet were unable to significantly surpass naive predictions on the logarithmic data.

The model's limitations: prediction of the increase or decrease in the number of requests for a given group of datasets, but not a specific number of those requests.

The detailed results for each model are shown in the slides 19-22

Conclusion

- PanDA queues utilization metric was proposed based on estimating jobs status duration at the resources
  - *Advanced jobs calculation methods were used allowing to get more precise number of jobs at PanDA queues within 1-hour intervals*
  - *Next task of this research is the evaluation of the developed ranking method*
- ATLAS Data popularity forecast models demonstrated:
  - *85-95% accuracy for different dataset groups [logarithmically transformed]*
  - *LSTM and Facebook Prophet can't beat the Naïve Forecast*
  - *Such models can be utilized for the prediction of overall trends, but not exact values*
  - ***Possible improvement:***
    - to filter out both test workloads and user-initiated benchmarking workloads
    - to change grouping method (i.e. group by process description)

BACKUP SLIDES

Advanced Jobs Calculation within 1-hour Intervals

- Queue utilization is measured within an hour
- ATLAS timeout rules for jobs
  - *21 days for running jobs or 7 days for throttled jobs*
- Timestamps of jobs that have not changed state before start of time interval under study are not updated in database for this interval and will not be captured within this interval
- Solution: Retrospective search for jobs in previous states

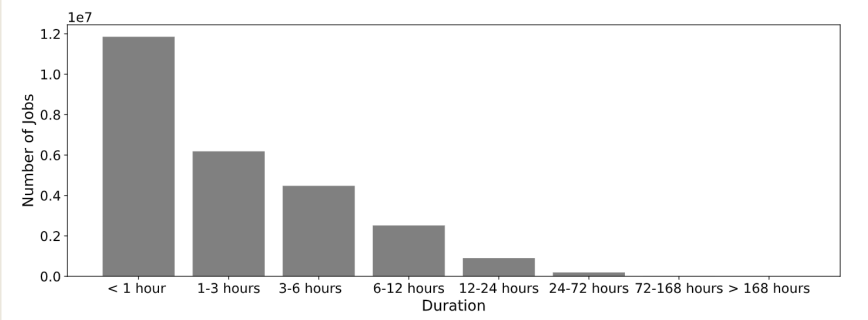


Table 2. Fragment of table JOBS STATUSLOG for a job with two statuses that had been recorded within the fixed interval between 22:00 and 23:00

	DateTime	Status
Statuses outside [22:00 – 23:00]	2023-01-30 21:49:35 $\rightarrow$2023-01-30 22:00:00	running
Statuses within [22:00 – 23:00]	2023-01-30 22:06:38	holding
	2023-01-30 22:06:43	merging

RETROSPECTIVE SEARCH EVALUATION

FOR RUNNING JOBS

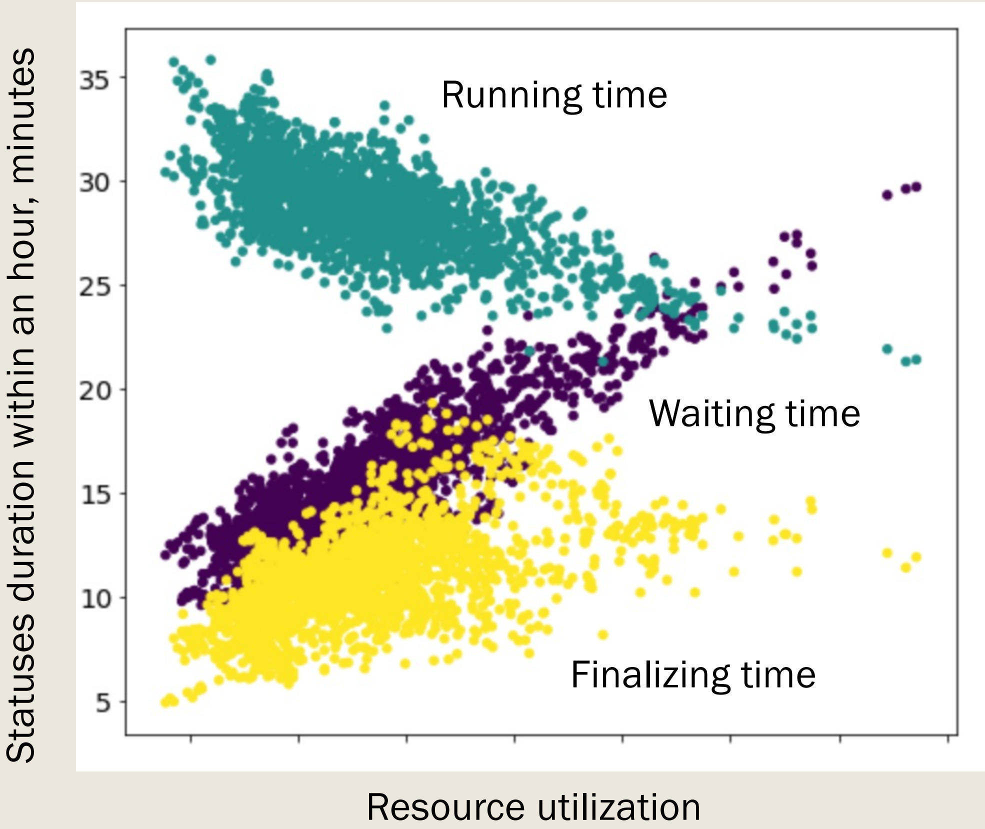
Time Interval	Direct Search	+3 hours look-back window	+6 hours look-back window	+12 hours look-back window	+24 hours look-back window	+48 hours look-back window
1 hour	**21K**	55K/+61%	62K/+66%	75K/+72%	85K/+75%	86K/+75%
3 hours	**67K**	105K/+36%	112K/+40%	130K/+48%	139K/+51%	140K/+52%
6 hours	**128K**	170K/+12%	184K/+18%	201K/+24%	210K/+31%	212K/+35%
12 hours	**273K**	328K/+8%	340K/+11%	354K/+15%	360K/+21%	362K/+25%
24 hours	**568K**	582K/+4%	588K/+5%	600K/+8%	612K/+12%	613K/+14%

- *Time interval* – initial time interval for calculating number of jobs
- *Direct search* – direct search in the PanDA DB within the specified time interval
- *+N hours look-back window* – retrospective jobs search within N hours

Increasing the time interval for searching jobs from one hour to two days allows for detecting 75% more jobs running during an hour compared to searching directly at the specified hour.

Even 3 hours look-back window allows for detecting 60% more jobs.

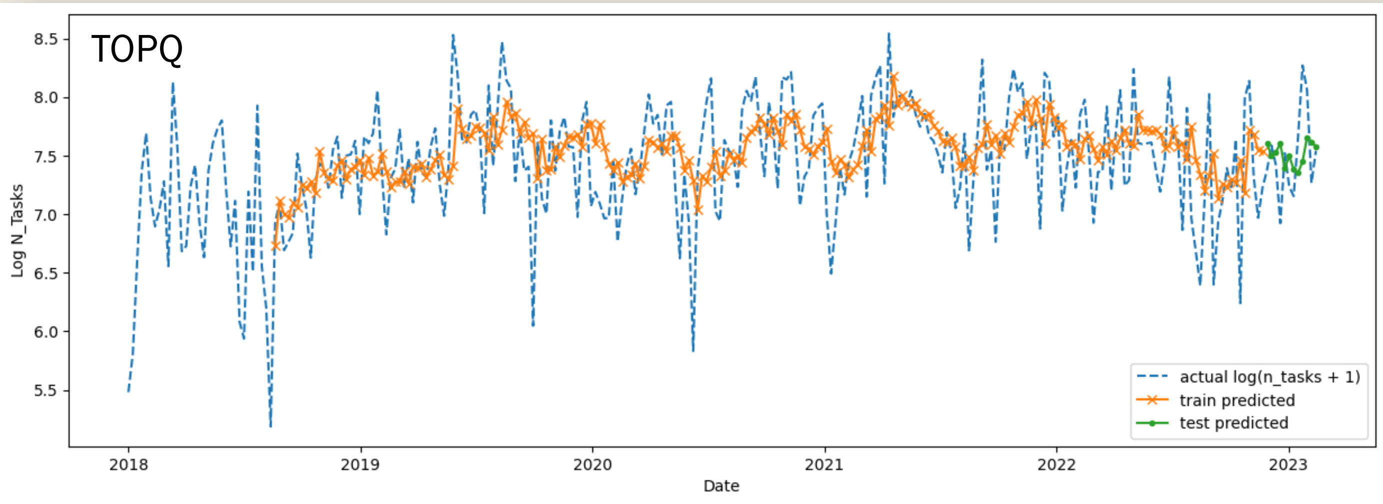
Correlations



- Correlations between resource utilization and job status duration at the resources:
  - *Utilization increases with the growth of waiting and finalizing time and decrease of running time*

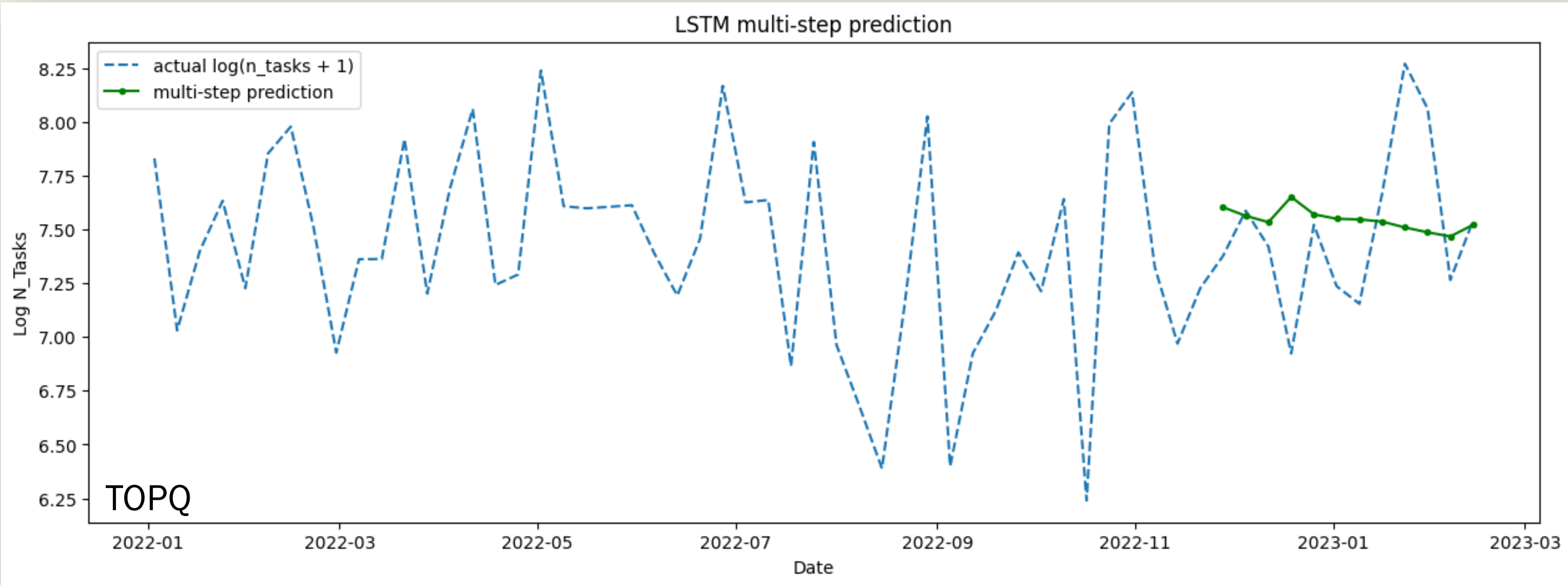
LSTM 1-step Forecast [log]

- Prediction one new point (a value for the future week in our case) based on several past points.
- The model was evaluated on the data, transformed by log, while the accuracy on the original data becomes substantially smaller, that can be caused by rapidly oscillating time series with peaks, which can not be properly predicted by the model.
- Prediction for 12 weeks ahead.
- **Error = 4.9% (cross validation)**



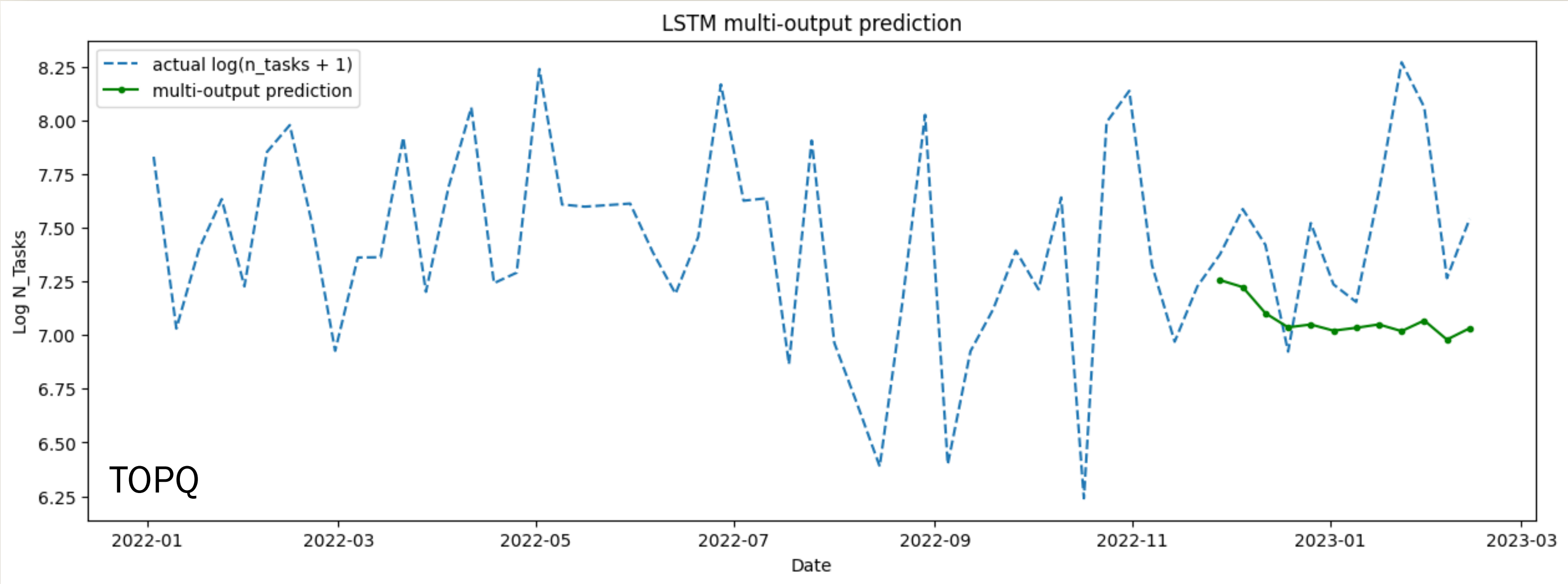
LSTM Multi-step Forecast [log]

- Sequential applying one-step model with addition of a new predicted point to the array of historical values at each step.
- Prediction for 12 weeks ahead.
- **Error = 5.05% (cross validation)**



LSTM Multi-output Forecast [log]

- Using a model that outputs several future points simultaneously based on the several points in the past.
- Prediction for 12 weeks ahead.
- **Error = 6.67% (cross-validation)**



Facebook Prophet Forecast [log]

- Facebook Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects.
- Prediction for 12 weeks ahead.
- **Error = 6.83% (cross-validation)**