Premio Nazionale "Giulia Vita Finzi" 2018

Luca Giommi
University of Bologna and INFN, Italy
luca.giommi3@unibo.it

Workshop di CCR: La Biodola, 3 - 7 giugno 2019

09/2015 - Dissertation of the Bachelor's thesis
Predicting CMS datasets popularity with Machine Learning (1)

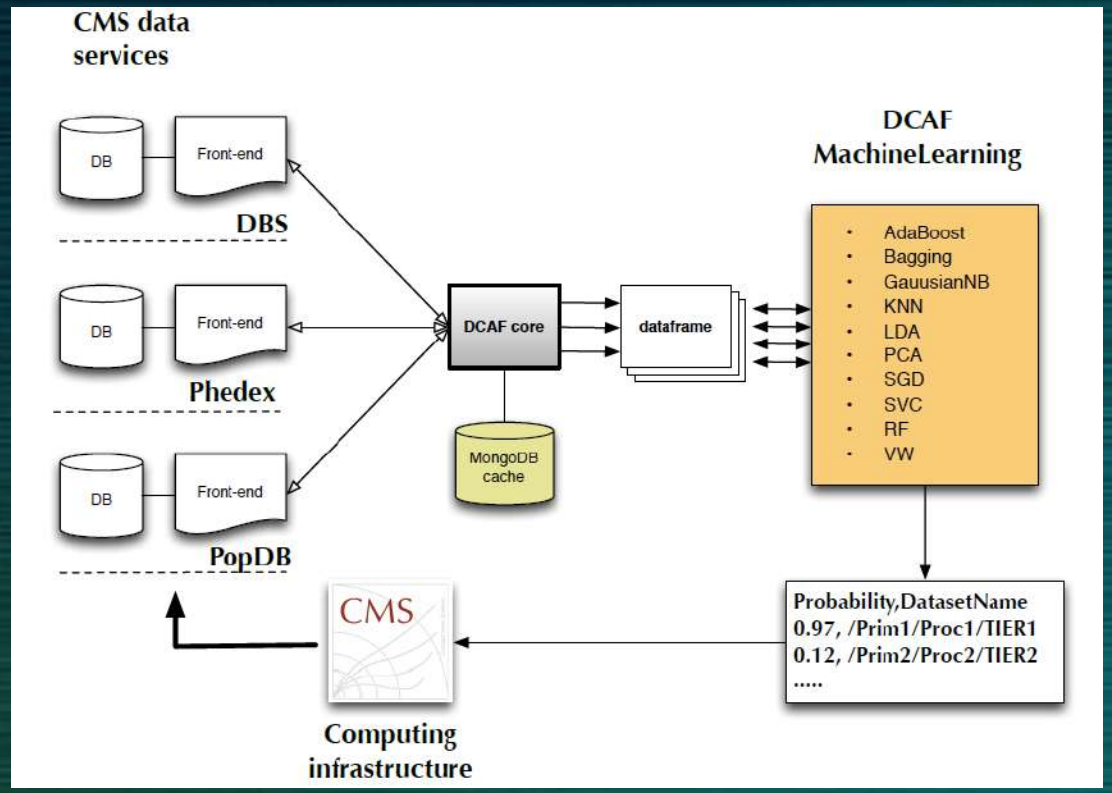
GOAL: prediction of the future popularity of the CMS datasets based on their previous one.

Popularity: observable that quantifies the interest of the CMS physicists for the datasets (in terms of the number of accesses, number of users and CPU hours).

Reasons

- Optimization of the use of distributed resources
- Evolution towards dynamic and adaptive data management models

Use of Machine Learning techniques through the DCAFPilot prototype



http://www.infn.it/thesis/thesis_dettaglio.php?tid=10091

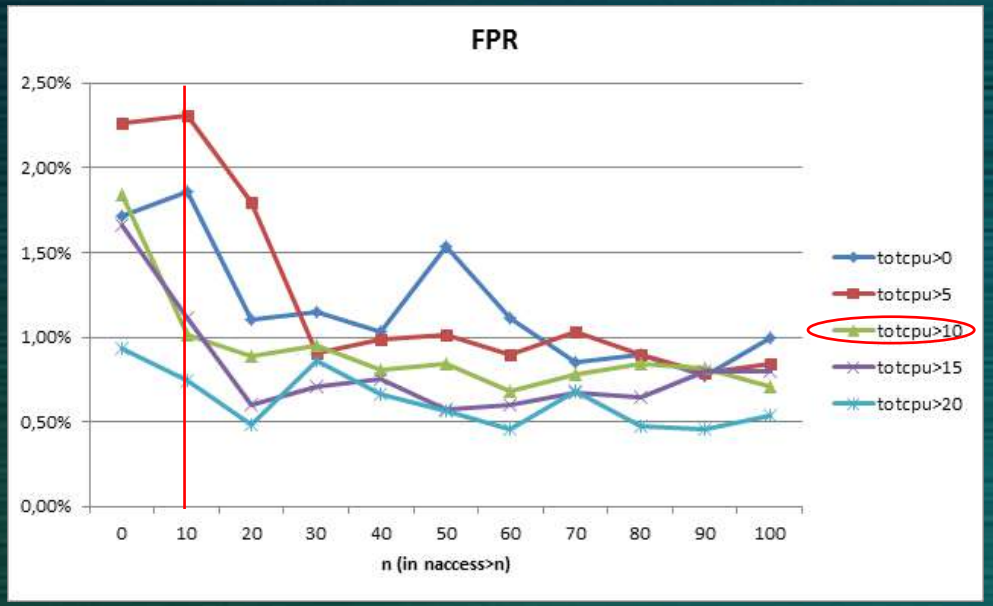
Predicting CMS datasets popularity with Machine Learning (2)

1. **Maximise**: accuracy, precision, recall, F1, % TruePositive
2. **Minimise**: FalsePositiveRate, FalseNegativeRate

Using information about datasets of 2014, the optimal model is characterised by:

- RandomForestClassifier
- Number of accesses > 10 & CPU hours > 10

Classifier	Accuracy	Precision	Recall	F1	FPR
LinearSVC	0,967	0,820	0,781	0,800	1,58%
SVC	0,971	0,767	0,950	0,849	2,66%
RandomForestClassifier	0,981	0,875	0,910	0,892	1,20%
ExtraTreesClassifier	0,976	0,852	0,866	0,859	1,38%
BernoulliNB	0,728	0,237	1,000	0,383	29,75%
SGDClassifier	0,971	0,758	0,965	0,849	2,83%
RidgeClassifier	0,834	0,261	0,530	0,350	13,84%
GradientBoostingClassifier	0,980	0,845	0,931	0,886	1,57%
DecisionTreeClassifier	0,970	0,811	0,839	0,825	1,80%
AdaBoostClassifier	0,978	0,806	0,970	0,880	2,16%
BaggingClassifier	0,984	0,901	0,907	0,904	0,92%



The results of this work have been presented at the International Symposium on Grids and Clouds 2015 (ISGC15), Taipei (Taiwan) and at the 17th International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT 2016), Valparaiso (Chile)

PoS ISGC2015 (2015) 008

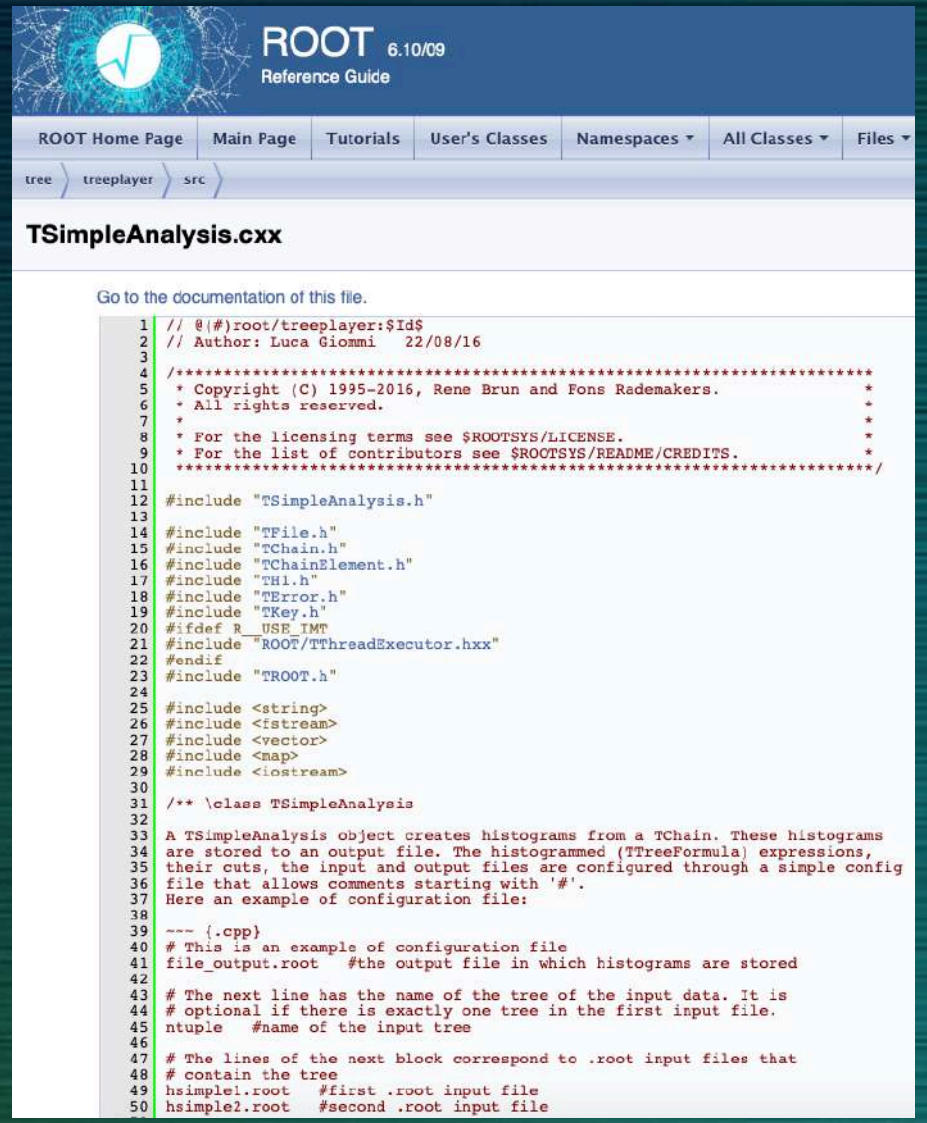
J.Phys.Conf.Ser. 762 (2016) no.1, 012048

Summer Student @ CERN in 2016

TSimpleAnalysis: histogramming many trees in parallel (1)

Final report: http://cds.cern.ch/record/2217995?ln=it
File Reference: https://root.cern.ch/doc/v610/TSimpleAnalysis_8cxx.html
***rootdrawtree* command line tool**: https://github.com/root-project/root/blob/master/main/python/rootdrawtree.py

Supervisors: Axel Naumann, Danilo Piparo



TSimpleAnalysis: histogramming many trees in parallel (2)

```
rootdrawtree --output output.root --input hsimple.root --histo 'x=px if px >0'
```

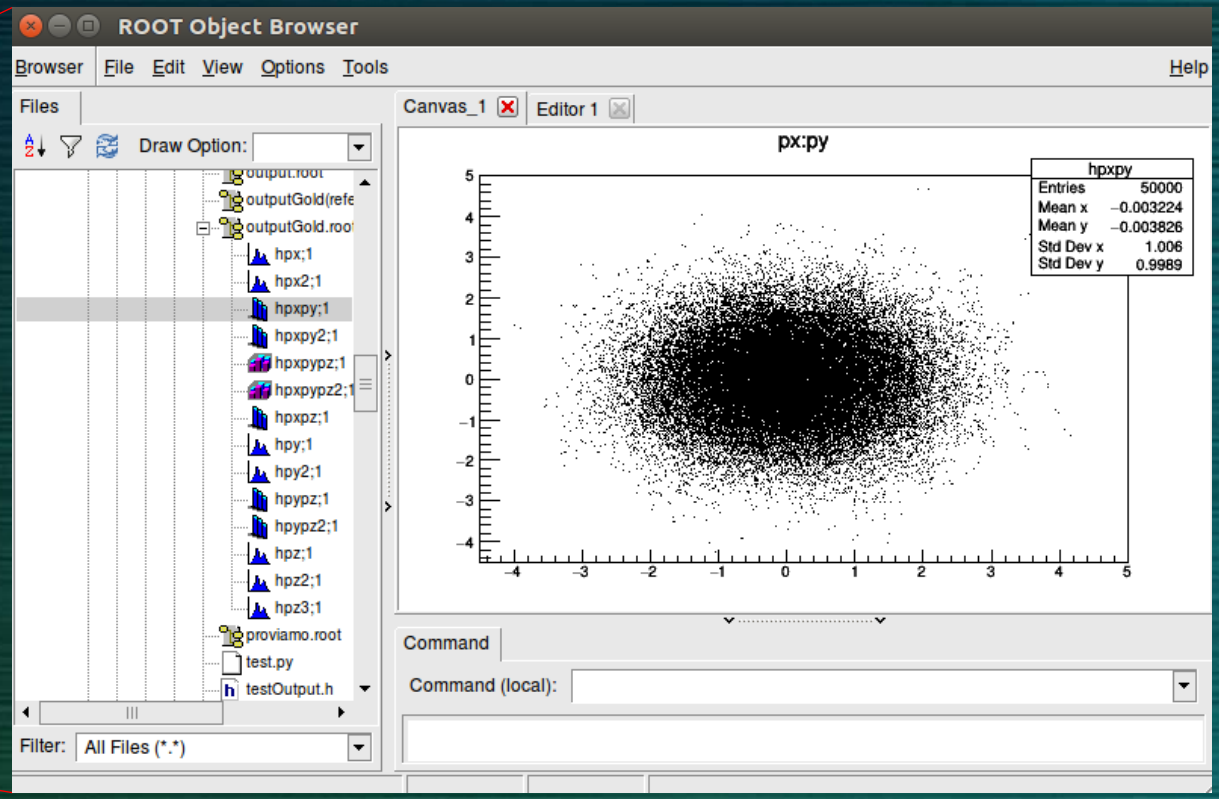
```
rootdrawtree configFile.txt
```

```
#This is a the configuration file with comments
#
     outputGold.root  #I can put here a comment
  ntuple  #name of the tree that is optional
        hsimple.root  #name of the first .root input file
hsimple2.root   #name of the second .root input file

   #we can put here a comment line
#white lines are skipped in the configuration procedure

#the following lines represents the histograms we want to create
hpx=px if px<-3
hpx2=px if px>-1
hpy=py
hpy2=py if py>1.5
hpz=pz if pz>8
hpz2=pz if pz<1
hpz3=pz if pz>6 & pz<9
hpxpy=px:py
hpxpy2=px:py if px>2 & py<-2
hpxpz=px:pz if px>0.5 &pz>2
hpypz=py:pz
hpypz2=py:pz if py>-0.5
hpxpypz=px:py:pz
hpxpypz2=px:py:pz if px<0 & py<0 & pz<3

 #this is the end of the configuration file
```



Prototype of Machine Learning "as a Service" for CMS Physics in Signal vs Background discrimination

Supervisor:
Prof. Daniele Bonacorsi

Cosupervisors:
Dott. Valentin Kuznetsov
Prof. Andrea Castro

Slides presented at the dissertation

Dissertation date:
23/03/2018

http://www.infn.it/thesis/thesis_dettaglio.php?tid=11847

Towards more, better and easier Machine Learning in HEP

Improve nowadays analyses and towards High Luminosity LHC

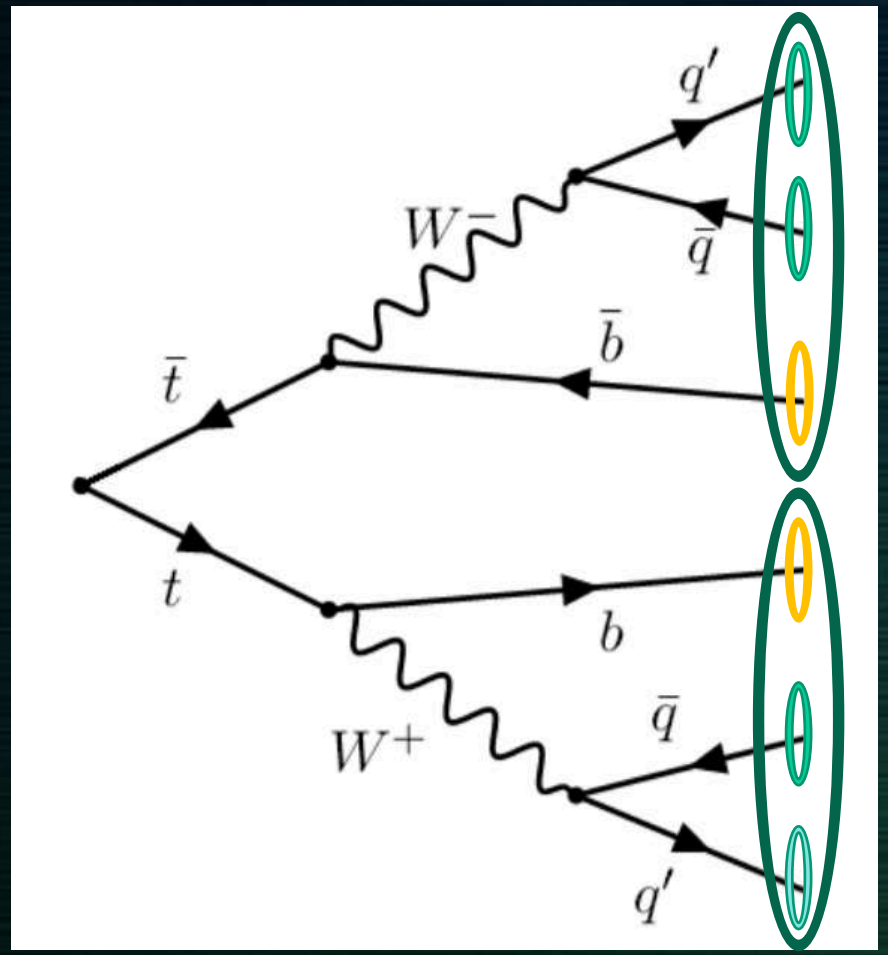
Importance of artificial intelligence

Build a system that uses Machine Learning for physics use cases in a better and easier way for the user

GOAL

A full proof-of-concept demonstration of an end-to-end data service to provide trained ML models to the CMS software framework (CMSSW) and its usage in Signal/Background (S/B) discrimination in $t\bar{t}$ selection

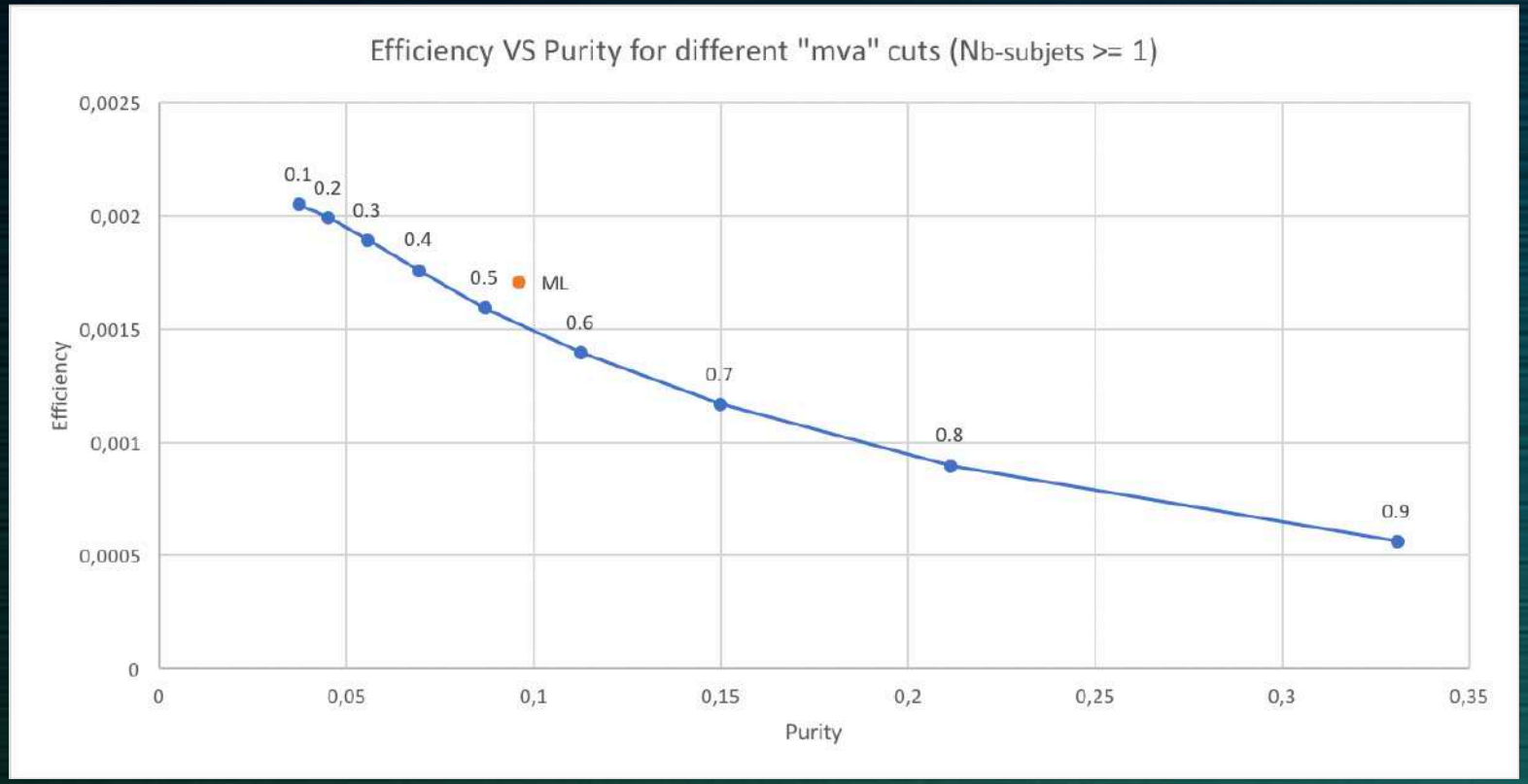
The $t\bar{t}$ selection use case



Motivation → Huge background and difficulties in signal versus background discrimination

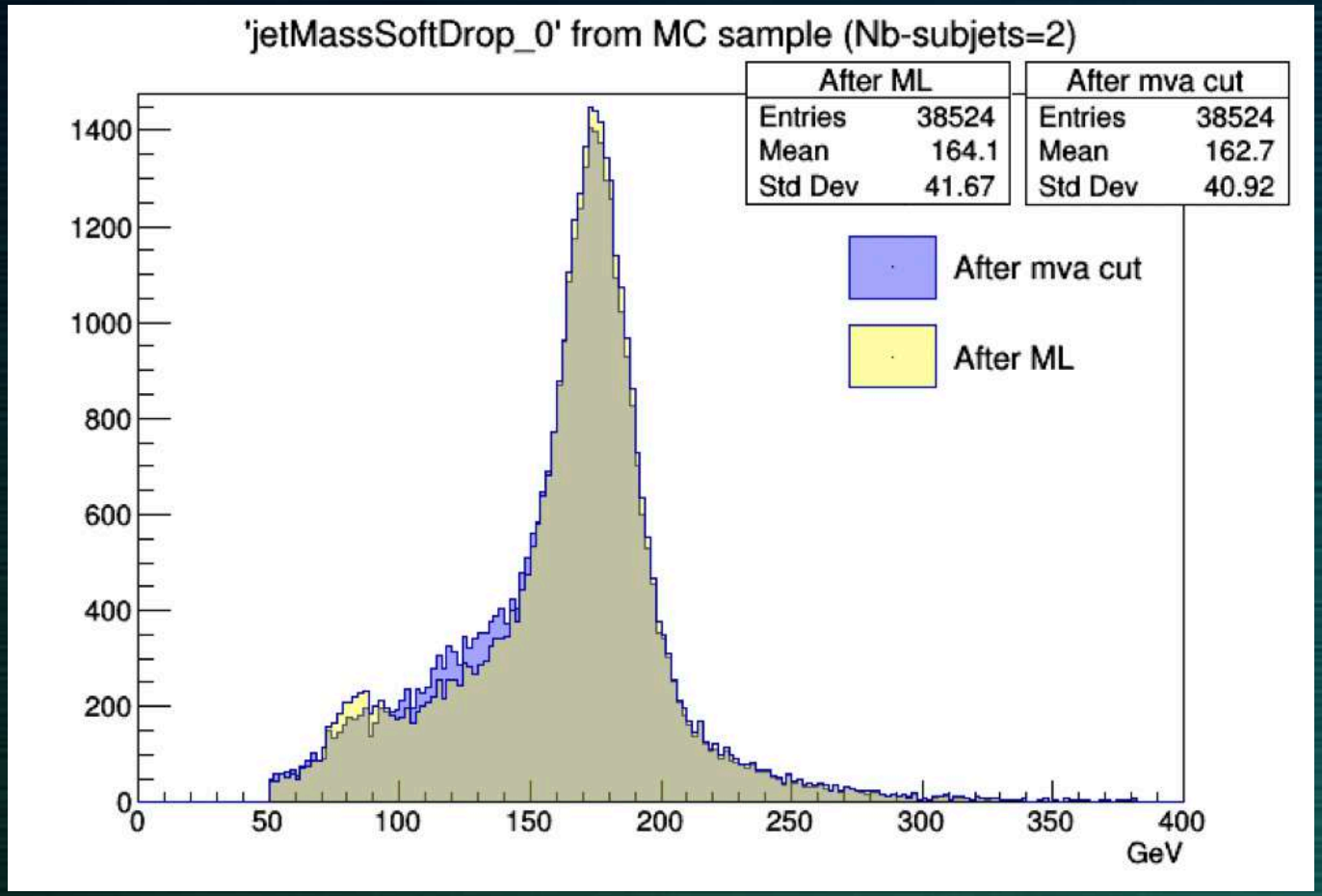
Process: train the algorithm
- Signal → Monte Carlo sample
- Background → signal-depleted data sample

Comparison between current analysis and ML



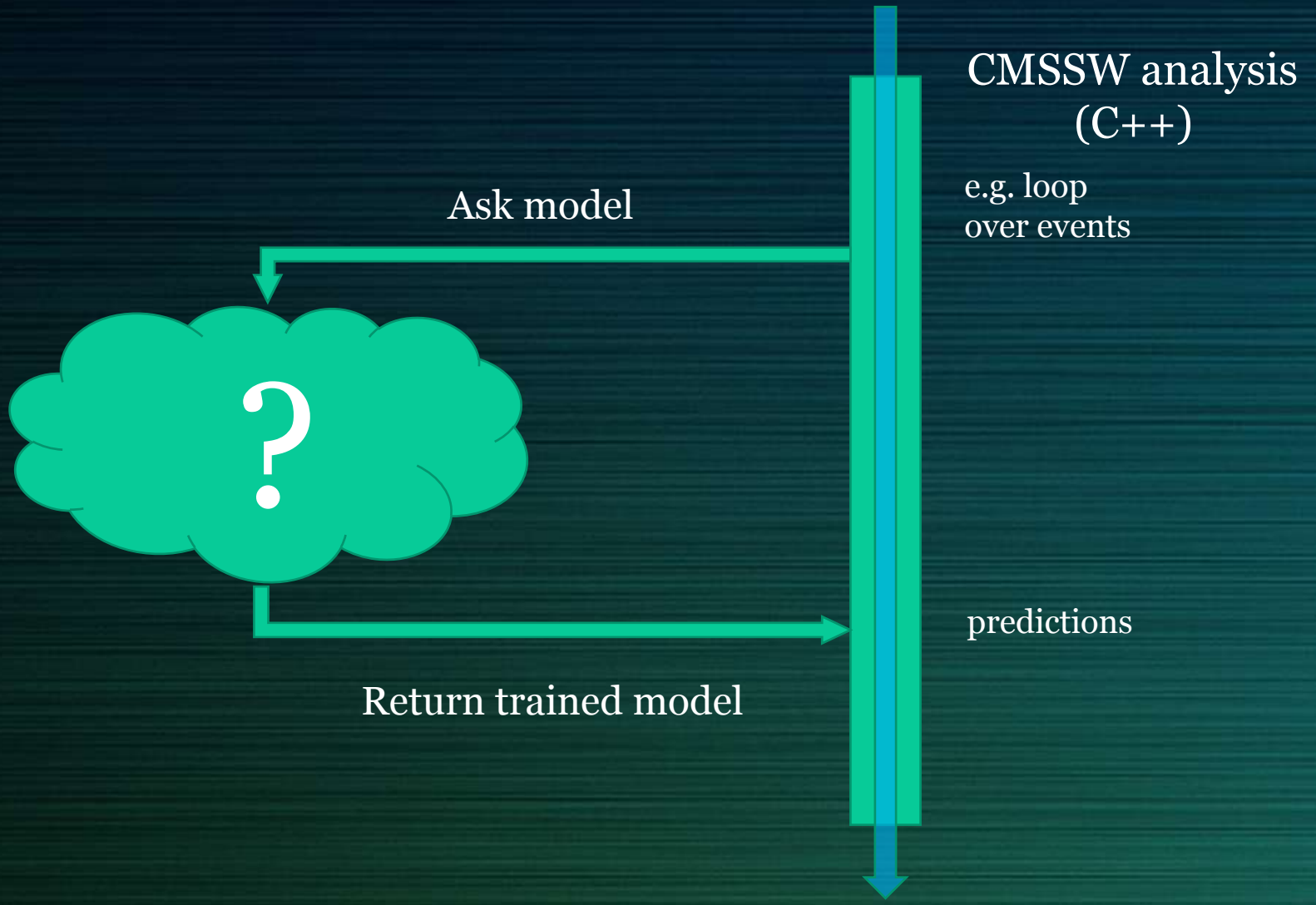
Efficiency and purity for different mva cuts (the values quoted above the dots), compared to the same obtained by ML, in the case of Nb–subjets >= 1

No significant bias between ML and MVA

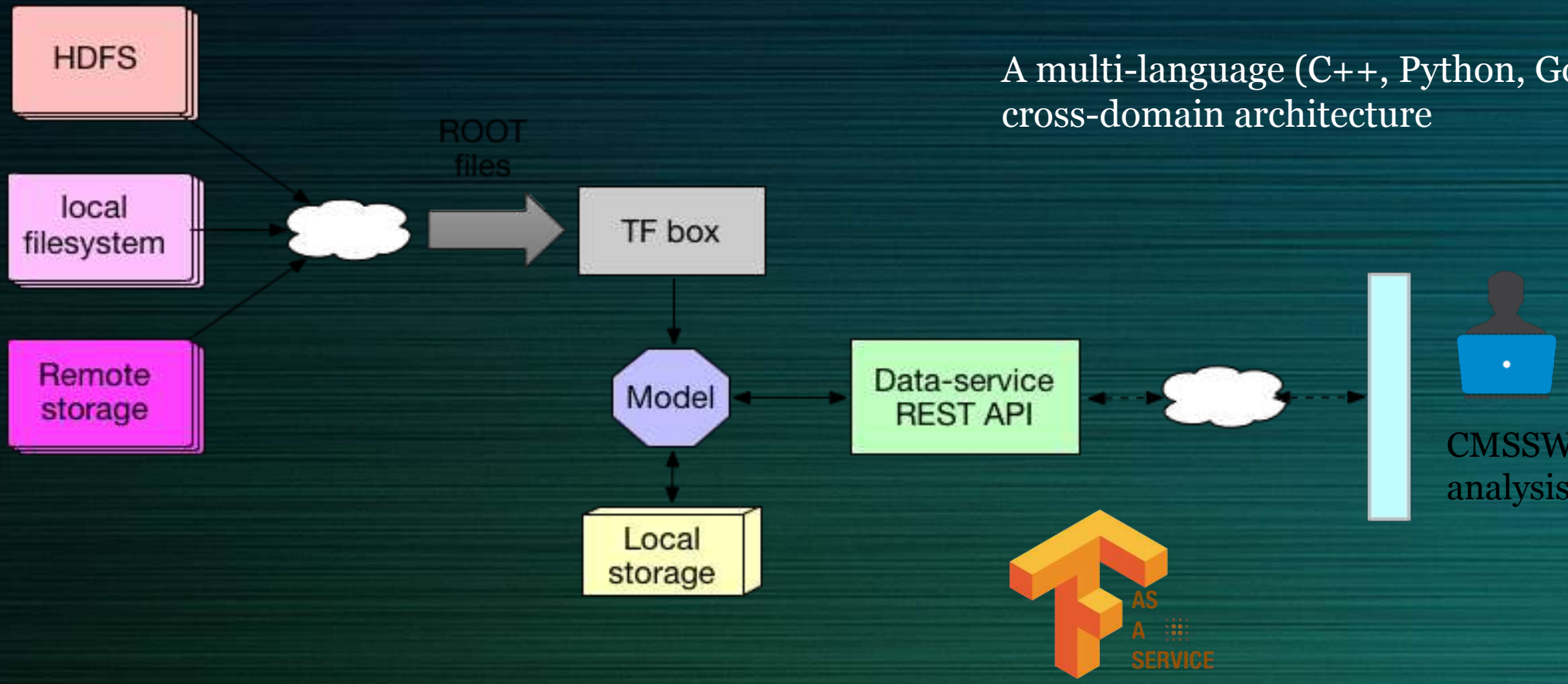


Invariant mass of the leading jet after the "soft-drop declustering" algorithm for the MC data with Nb–subjets = 2

Basic idea of the data service



Architecture of the ML “as a service” (TFaaS)



A multi-language (C++, Python, Go) and cross-domain architecture

General process

- Select use case → $t\bar{t}$ selection
- Create a model → scikitlearn-based
  1. *Data Preparation* → hardest part to read ROOT file using a new tool ("uproot")
  2. *Data Validation*
  3. *Algorithm selection*
  4. *Parameter tuning*
- Interface with the prototype → create a keras-tensorflow model

In this thesis

Demonstrated that the prototype works!

These results have been presented at the International Symposium on Grids and Clouds 2018, ISGC18, Taipei, and presented as poster at the Sixth Annual Conference on Large Hadron Collider Physics (LHCP2018), Bologna

PoS(ISGC 2018 & FCDD)022

PoS LHCP2018 (2018) 093

uproot is a reader and a writer of the ROOT file format using only Python and Numpy.

Unlike PyROOT and root_numpy, uproot does not depend on C++ ROOT. Instead, it uses Numpy to cast blocks of data from the ROOT file as Numpy arrays.

Unlike the standard C++ ROOT implementation, uproot is only an I/O library, primarily intended to stream data into machine learning libraries in Python.

It allows to access remote files through XRootD and HTTP.

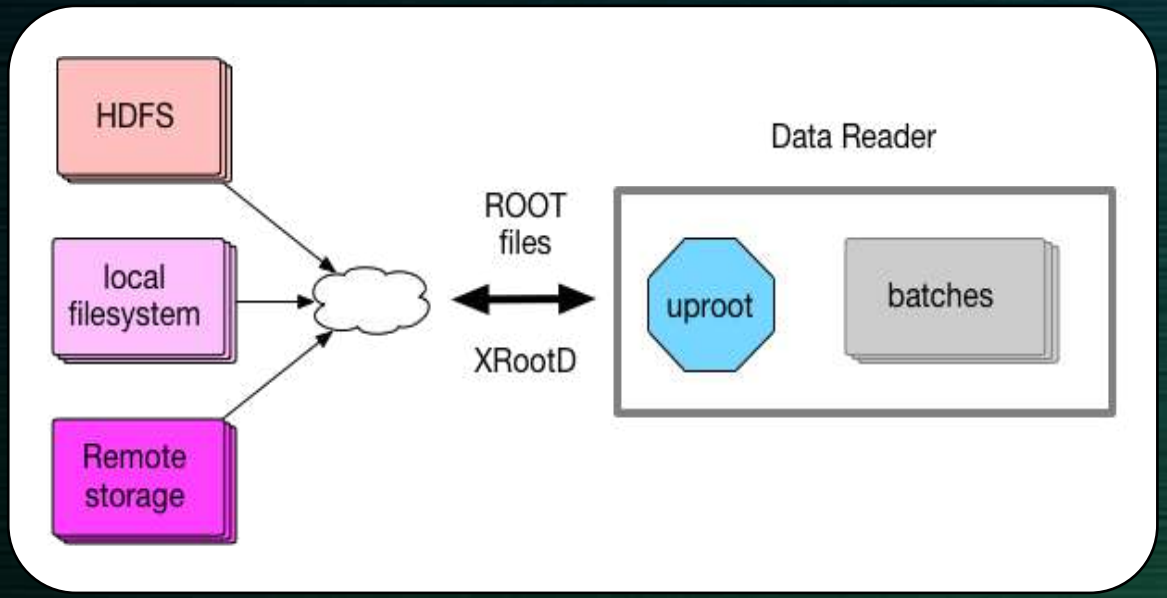
https://uproot.readthedocs.io

Next steps (from TFaaS to MLaaS4HEP)

Exploit the potential of the "Data Streaming Layer"

Use distributed NanoAOD datasets (accessible via xrootd) without intermediate step of converting data into CSV format, pre-processing, etc.

In this way we can easily read (through uproot) and directly use O(10 TB) of datasets in the training of the ML models



Why NanoAOD?

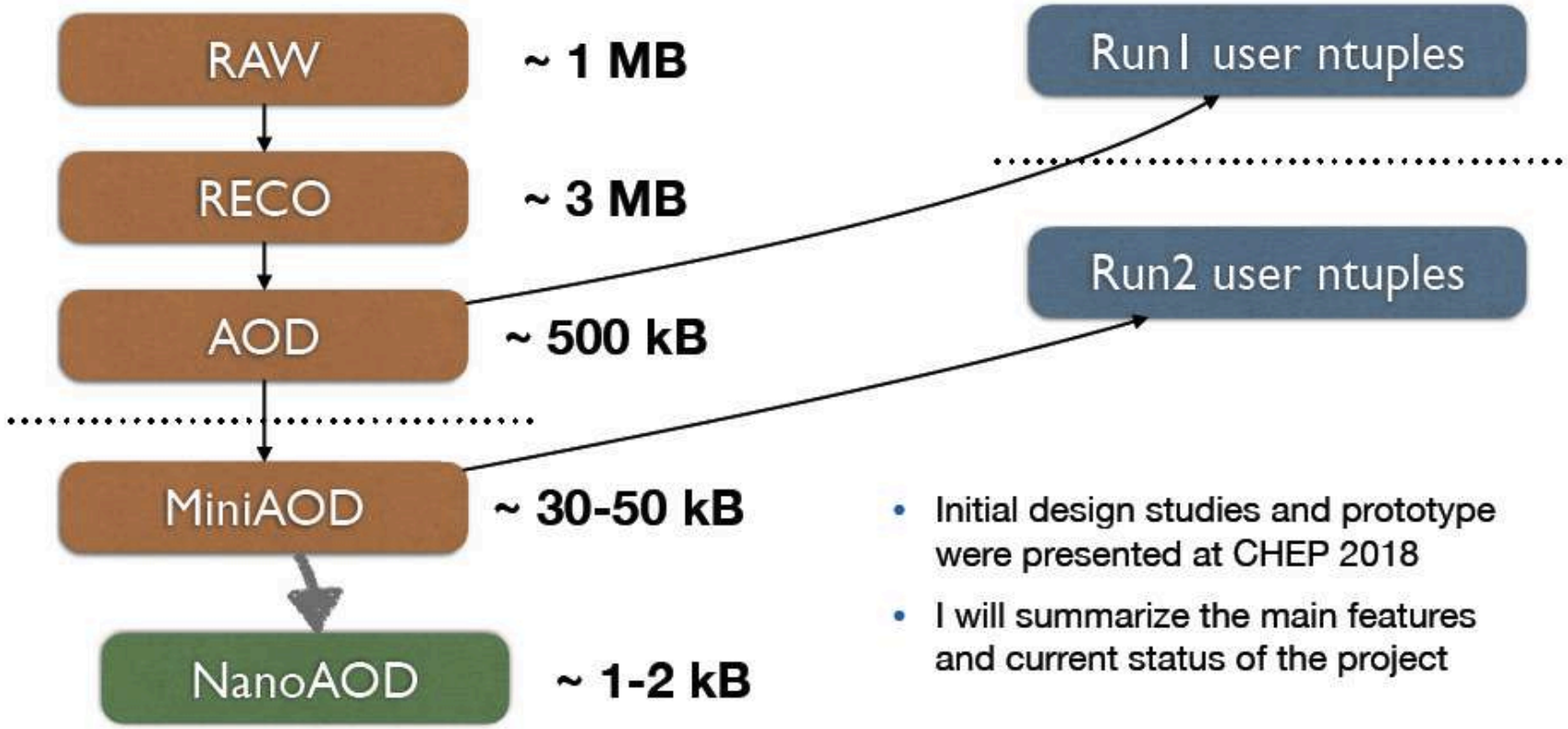
- It is a flat ntuple format with only standard data types (e.g. int, float, vectors), so it is simple to export to modern machine learning frameworks
- Many analysis are switching to the new 1kb/event format

Goals

- We don't aim to reproduce an entire analysis with all details, but its feasibility using NanoAOD ROOT files and the MLaaS4HEP framework.
- Performance benchmarks (CPU vs GPU vs TPU, and various versions) for the training phase

CMS event data formats

- Main event data formats is use for standard pp collision runs:



- Initial design studies and prototype were presented at CHEP 2018
- I will summarize the main features and current status of the project

Towards Predictive Maintenance with Machine Learning at the INFN-CNAF computing center

Luca Giommi
University of Bologna and INFN, Italy
luca.giommi3@unibo.it

Co-authors:
D. Bonacorsi, T. Diotalevi, S. R. Tisbeni, L. Rinaldi
University of Bologna, Italy
L. Morganti, A. Falabella, E. Ronchieri, B. Martelli, A. Ceccanti
INFN-CNAF, Italy

International Symposium on Grids and Clouds, Taipei, 04.04.2019

Goal of the work

In order to increase efficiency and to remain competitive in the long run, CNAF is launching various activities aiming at implementing a global predictive maintenance solution for the site. Because of efficient storage systems are one of the key ingredients of Tier-1 operations, at CNAF an exploratory work started by investigating logs from the StoRm service.

Information about the status and the progress of the requests managed by the service is stored in log files, in a usually complex format

handle and parse the log files to extract relevant information and design it to work automatically

Define a problematic period with anomalies in the system and a normal one

Compare the two behaviors and build ML models for anomaly prediction

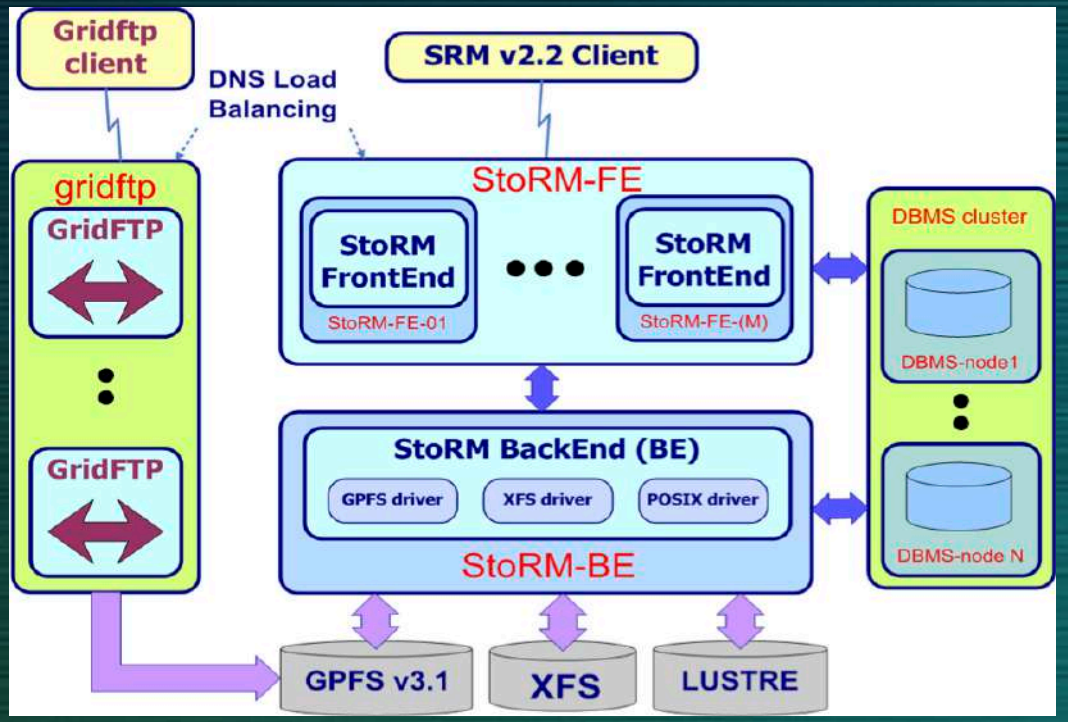
Storage Resource Managers and StoRM

Storage Resource Managers (SRMs) are middleware services whose function is to provide dynamic space allocation and file management of shared geographically distributed storage resources.

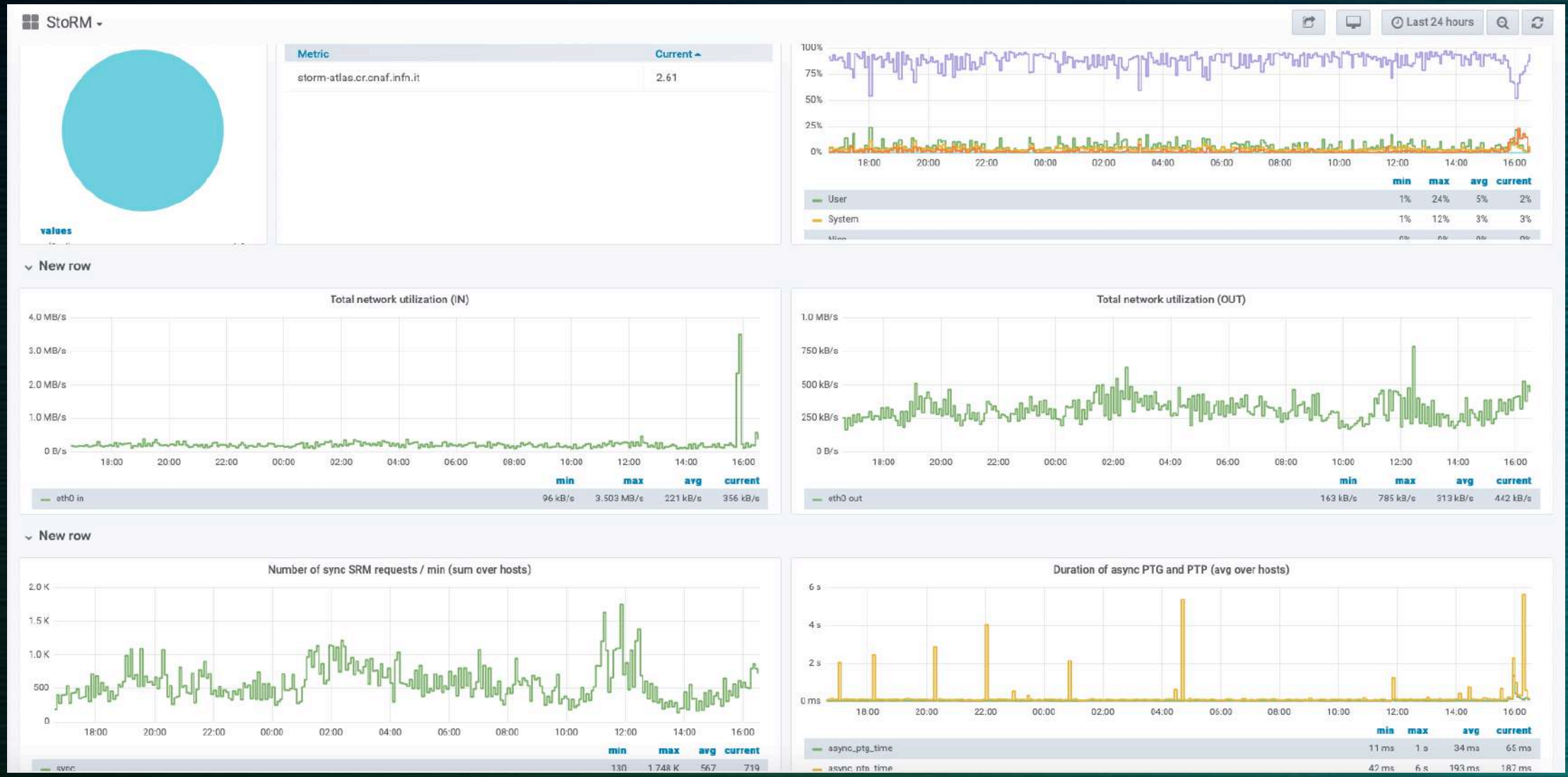
StoRM is the SRM solution adopted by the INFN-CNAF Tier-1. StoRM has a multilayer architecture made by two stateless components, called *Frontend* and *Backend*, and one database.

Frontend:
- exposes the SRM web service interface
- manages user authentication
- stores SRM requests data into the database and retrieve the status of ongoing requests
- interacts with the Backend

Backend:
- processes the SRM requests managing files and space
- enforces authorization permissions
- can interact with other Grid services



Current monitoring via Graphana



More insight on events via plain log files

```
[61998] Fri Nov 30 03:24:32 2018 :: Configuration read from /etc/gridftp.conf.
[61998] Fri Nov 30 03:24:32 2018 :: Server started in inetd mode.
[61998] Fri Nov 30 03:24:32 2018 :: New connection from: fts804.cern.ch:41956
[61998] Fri Nov 30 03:24:32 2018 :: DN /DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=ddmadmin/CN=
531497/CN=Robot: ATLAS Data Management successfully authorized.
[61998] Fri Nov 30 03:24:32 2018 :: User atlasprd045 successfully authorized.
[61998] Fri Nov 30 03:24:32 2018 :: Starting to transfer "/storage/gpfs_tsm_atlas/atlas/atlasda
tatape/data18_hi/RAW/other/data18_hi.00367134.physics_MinBias.daq.RAW/data18_hi.00367134.physic
s_MinBias.daq.RAW._lb0100._SFO-1._0001.data".
[61998] Fri Nov 30 03:24:36 2018 :: Finished transferring "/storage/gpfs_tsm_atlas/atlas/atlasd
atatape/data18_hi/RAW/other/data18_hi.00367134.physics_MinBias.daq.RAW/data18_hi.00367134.physi
cs_MinBias.daq.RAW._lb0100._SFO-1._0001.data".
[61998] Fri Nov 30 03:24:36 2018 :: Closed connection from fts804.cern.ch:41956
```

```
48:11.701 Thread 41 -  INFO [41eec2e2-80a7-4c1b-82b4-d42ff57f0b7e]: Result for request
itus' is 'SRM_REQUEST_INPROGRESS'
48:11.717 Thread 13 -  INFO [1b3a9db9-1325-467f-9b8b-68347dbb6ad3]: process_request :
Connection from 2001:1470:ff80:12:8e23:c32e:495d:3846
12/01 03:48:11.849 Thread 13 -  INFO [1b3a9db9-1325-467f-9b8b-68347dbb6ad3]: Request 'BOL statu
om Client IP='2001:1470:ff80:12:8e23:c32e:495d:3846' Client DN='/DC=ch/DC=cern/OU=Organic
s/OU=Users/CN=atlact1/CN=555105/CN=Robot: ATLAS aCT 1' # Requested token '17b26868-7752-4e3
jb-c05d1f72c1b9'
l 03:48:11.852 Thread 13 -  INFO [1b3a9db9-1325-467f-9b8b-68347dbb6ad3]: Result for request
status' is 'SRM_REQUEST_INPROGRESS'
```

```
00:00:00.976 - INFO [xmlrpc-36531] - srmReleaseFiles: user</DC=ch/DC=cern/OU=Organic Units/OU=U
sers/CN=atlpilo1/CN=614260/CN=Robot: ATLAS Pilot1> Request for [token: 79c544f6-a414-4da2-bbf7-
841ca62ed46a] for [SURL: [srm://storm-fe.cr.cnaf.infn.it/atlas/atlasdatadisk/rucio/mc16_13TeV/7
d/47/AOD.11188997._000493.pool.root.1]] succesfully done with  [status: SRM_SUCCESS: Files rele
ased]
00:00:00.984 - INFO [xmlrpc-36532] - srmLs: user </DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=d
dmadmin/CN=531497/CN=Robot: ATLAS Data Management> Request for [SURL: [srm://storm-fe.cr.cnaf.i
nfn.it/atlas/atlasdatatape/data18_hi/RAW/other/data18_hi.00367321.physics_UPC.daq.RAW/data18_hi
.00367321.physics_UPC.daq.RAW._lb0374._SFO-4._0001.data]] failed with: [status: SRM_FAILURE: Al
l requests failed]
00:00:01.003 - ERROR [xmlrpc-36541] - srmRm: File does not exist
```

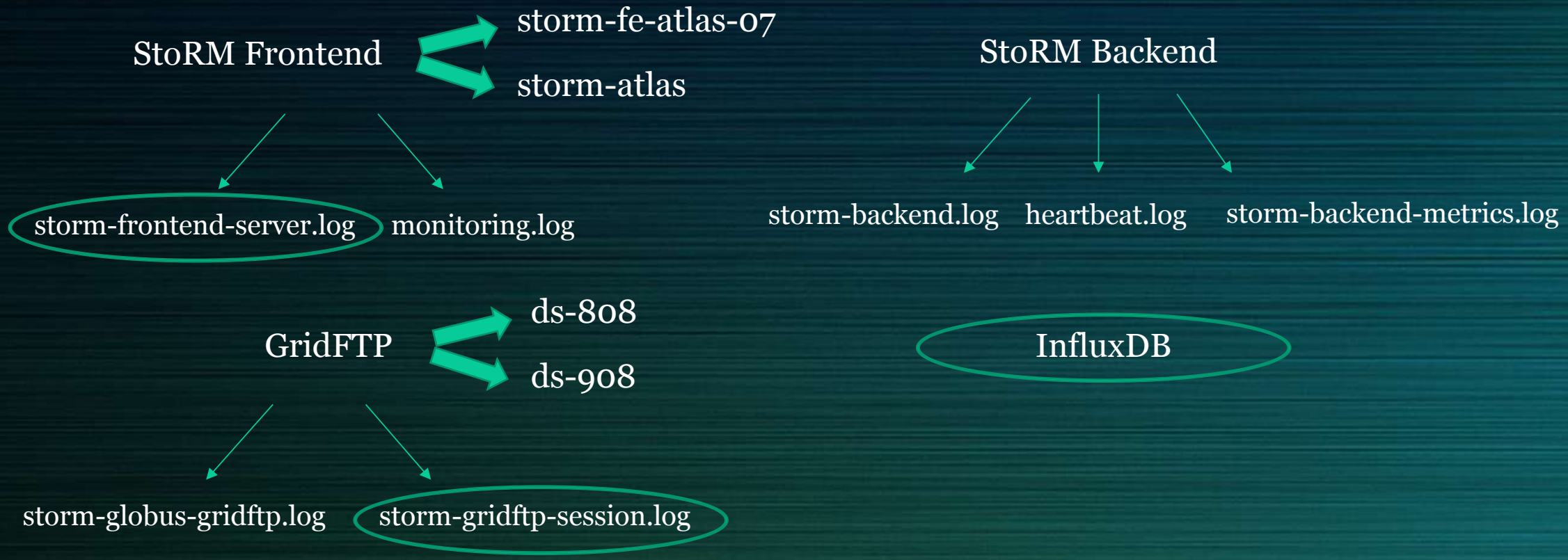
```
03:48:42 : [#  1105 lifetime=18:25:00] S [OK:604700,F:74281,E:0,m:0.000,M:612.382,Avg:0.1
[OK:61070,F:0,E:0,m:0.006,M:0.497,Avg:0.013] Last:(S [OK:510,F:58,E:0,m:0.000,M:4.230] A
```

```
00:00:42.005 - synch [(count=1679600, m1_rate=705.251823604344, m5_rate=609.0933503831365, m15_
rate=579.1089899935228) (max=1528.8399909999998, min=0.084372999999999999, mean=78.6504215405824
, p95=339.85041, p99=660.244572)] duration_units=milliseconds, rate_units=events/minute
```

```
[2018-12-06 00:00:51,872]:  [#..8078 lifetime=134:37.01] Heap Free:9274504090 SYNCH [1951] Msynch
 [PTG:505092 PTP:494798] Last:( [#PTG=14 OK=14 M.Dur.=17] [#PTP=13 OK=13 M.Dur.=200] )
```

```
> select * from "one_month"."iostat.avg-cpu.pct_user" where host='ds-908.cr.cnaf.infn.it'
name: iostat.avg-cpu.pct_user
time                 domain          duration          host                    metric                  tag1          tag2  value
----                 ------          --------          ----                    ------                  ----          ----  -----
2019-01-30T00:00:00Z cr.cnaf.infn.it 1.9276666666666664 ds-908.cr.cnaf.infn.it metrics-iostat-extended gridftp-xrootd atlas 1.97
```

ATLAS use case



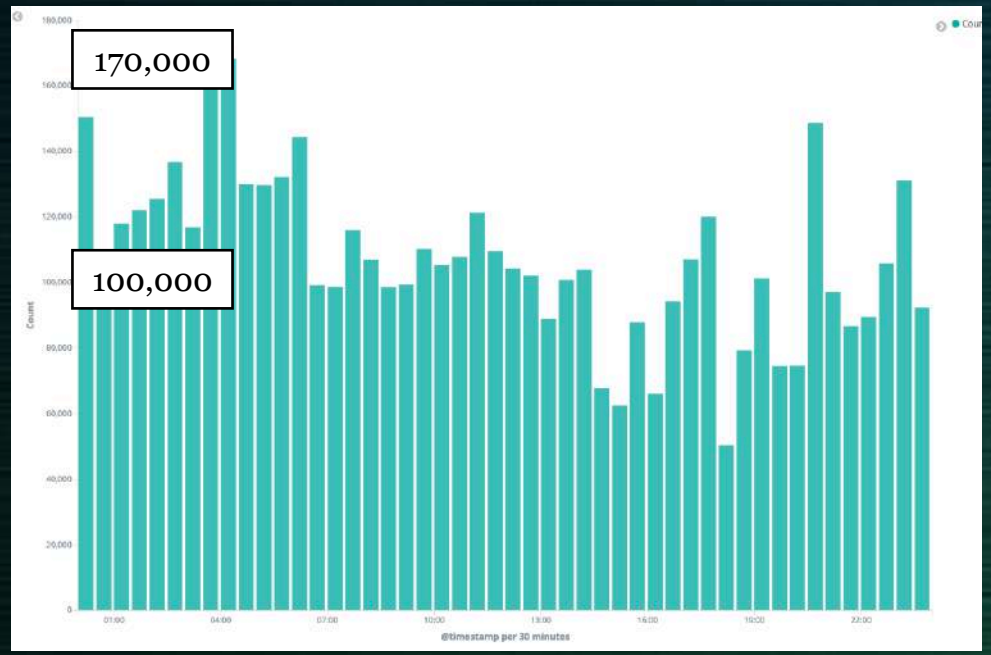
Followed path: take the sources individually, parse log files producing csv files, investigate the behavior of the features contained inside each log file, create a predictive ML model for each source

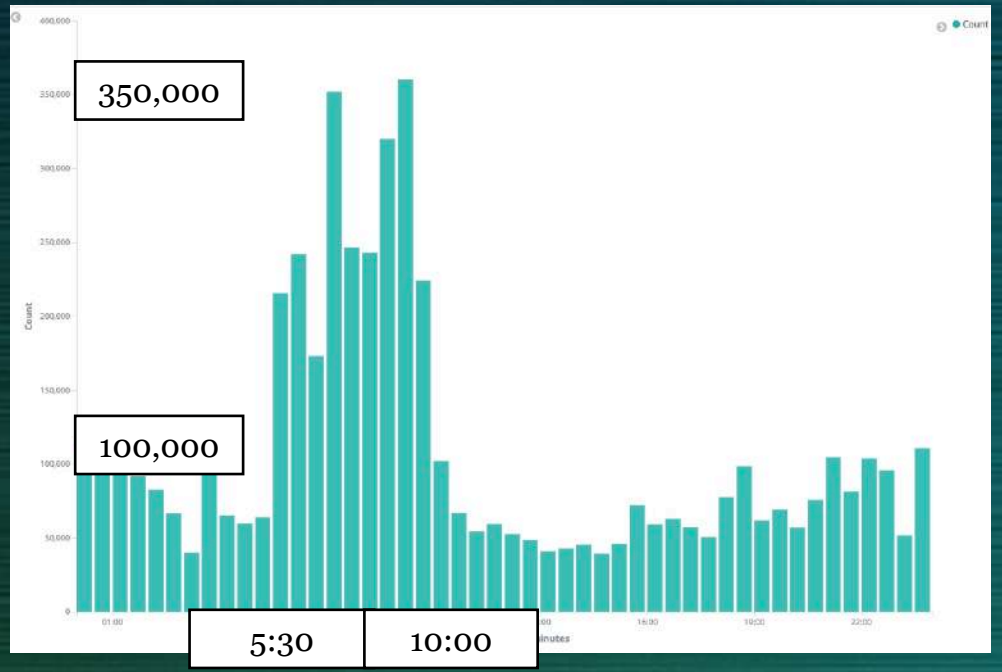
Choice of the critical period

NORMAL

CRITICAL

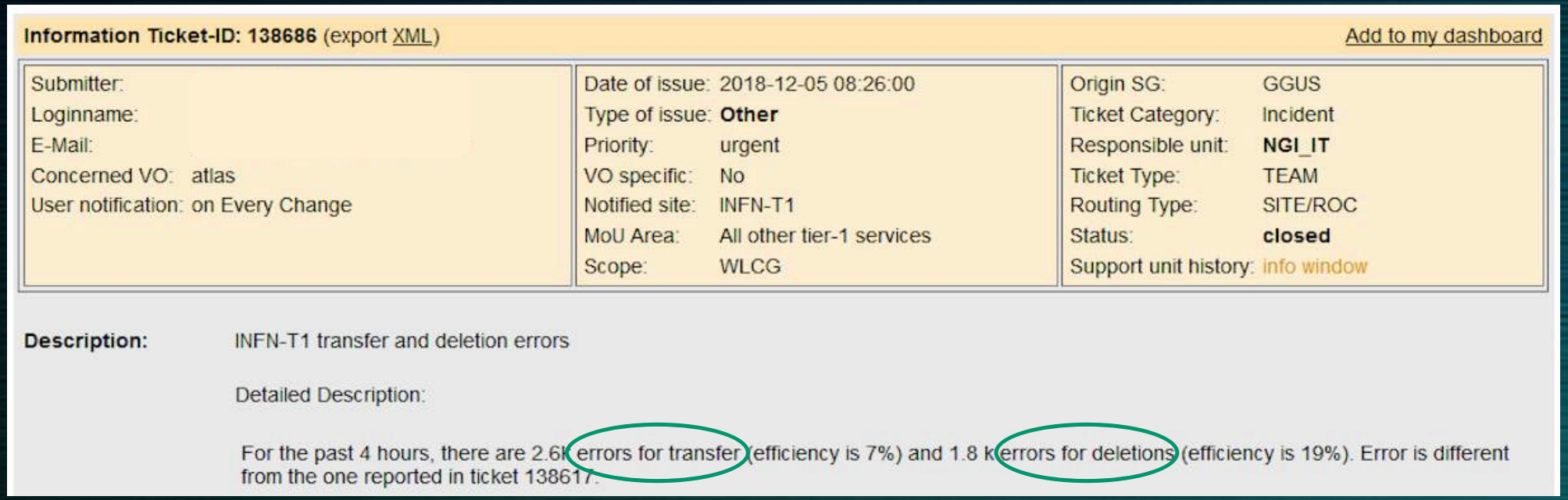
1 2 3 4 5 6 7 8 Dec 2018





Log entries count in storm-frontend-server.log file per 30 minutes in one day for the set of the two Frontend services.

What is critical about this period



Information Ticket-ID: 138686 (export XML) Add to my dashboard

Submitter:		Date of issue:	2018-12-05 08:26:00	Origin SG:	GGUS
Loginname:		Type of issue:	**Other**	Ticket Category:	Incident
E-Mail:		Priority:	urgent	Responsible unit:	**NGI_IT**
Concerned VO:	atlas	VO specific:	No	Ticket Type:	TEAM
User notification:	on Every Change	Notified site:	INFN-T1	Routing Type:	SITE/ROC
		MoU Area:	All other tier-1 services	Status:	**closed**
		Scope:	WLCG	Support unit history:	info window

Description: INFN-T1 transfer and deletion errors

Detailed Description:

For the past 4 hours, there are 2.6k errors for transfer (efficiency is 7%) and 1.8 k errors for deletions (efficiency is 19%). Error is different from the one reported in ticket 138617.

Two problems found: wrong configuration of the **file system** and wrong configuration of the **queues coming from the farm**

Situation back to normal the 13th December after the issues have been fixed and the addition of one more GridFTP server

Steps followed for each source

Parse log files, converting them in the csv form (Fontend case)

Log file

```
12/01 00:00:00.010 Thread 14 -  INFO [4c99ea76-eb8d-413e-8cd9-89253facb4e6]: process_request : Connection from 2001:948:61:1::10
12/01 00:00:00.032 Thread 53 -  INFO [153a16cc-522d-47b1-8f5f-6e022204cf64]: Result for request 'Put done' is 'SRM_SUCCESS'
```

Csv file

```
timestamp,datetime,thread,type,token,Request,DN,requested_token,num,surl,result,ip
1543622400.01,2018-12-01 00:00:00.010000,14,INFO,4c99ea76-eb8d-413e-8cd9-89253facb4e6,Connection,,,,,,2001:948:61:1::10
1543622400.032,2018-12-01 00:00:00.032000,53,INFO,153a16cc-522d-47b1-8f5f-6e022204cf64,Put done,,,,,SRM_SUCCESS,
```

Table

timestamp	datetime	thread	type	token	Request	DN	requested_token	num	surl	result	ip
1.543622e+09	2018-12-01 00:00:00.010000	14	INFO	4c99ea76-eb8d-413e-8cd9-89253facb4e6	Connection	NaN	NaN	NaN	NaN	NaN	2001:948:61:1::10
1.543622e+09	2018-12-01 00:00:00.032000	53	INFO	153a16cc-522d-47b1-8f5f-6e022204cf64	Put done	NaN	NaN	NaN	NaN	SRM_SUCCESS	NaN

Extract new features from the messages (Frontend case)

timestamp	datetime	thread	type	token	Request
1.543622e+09	2018-12-01 00:00:00.010000	14	INFO	4c99ea76-eb8d-413e-8cd9-89253facb4e6	Connection
1.543622e+09	2018-12-01 00:00:00.032000	53	INFO	153a16cc-522d-47b1-8f5f-6e022204cf64	Put done

```
list(data['Request'].unique())
```

```
['Connection',
 'Put done',
 'BOL status',
 'Ls',
 'PTP status',
 'Release files',
 'Rm',
 'Ping',
 'Get space tokens',
 'PTP',
 'PTG',
 'PTG status',
 'Get space metadata',
 'Mv',
 'Mkdir',
 nan,
 'BOL',
 'Abort request',
 'Check permission',
 'Abort files']
```

One hot encoding and summary of the log content in one row at each 15 minutes (Frontend case)

timestamp	datetime	thread	type	token	Request	DN	requested_token	num	surl	result	ip
1.543622e+09	2018-12-01 00:00:00.010000	14	INFO	4c99ea76-eb8d-413e-8cd9-89253facb4e6	Connection	NaN	NaN	NaN	NaN	NaN	2001:948:61:1::10
1.543622e+09	2018-12-01 00:00:00.032000	53	INFO	153a16cc-522d-47b1-8f5f-6e022204cf64	Put done	NaN	NaN	NaN	NaN	SRM_SUCCESS	NaN

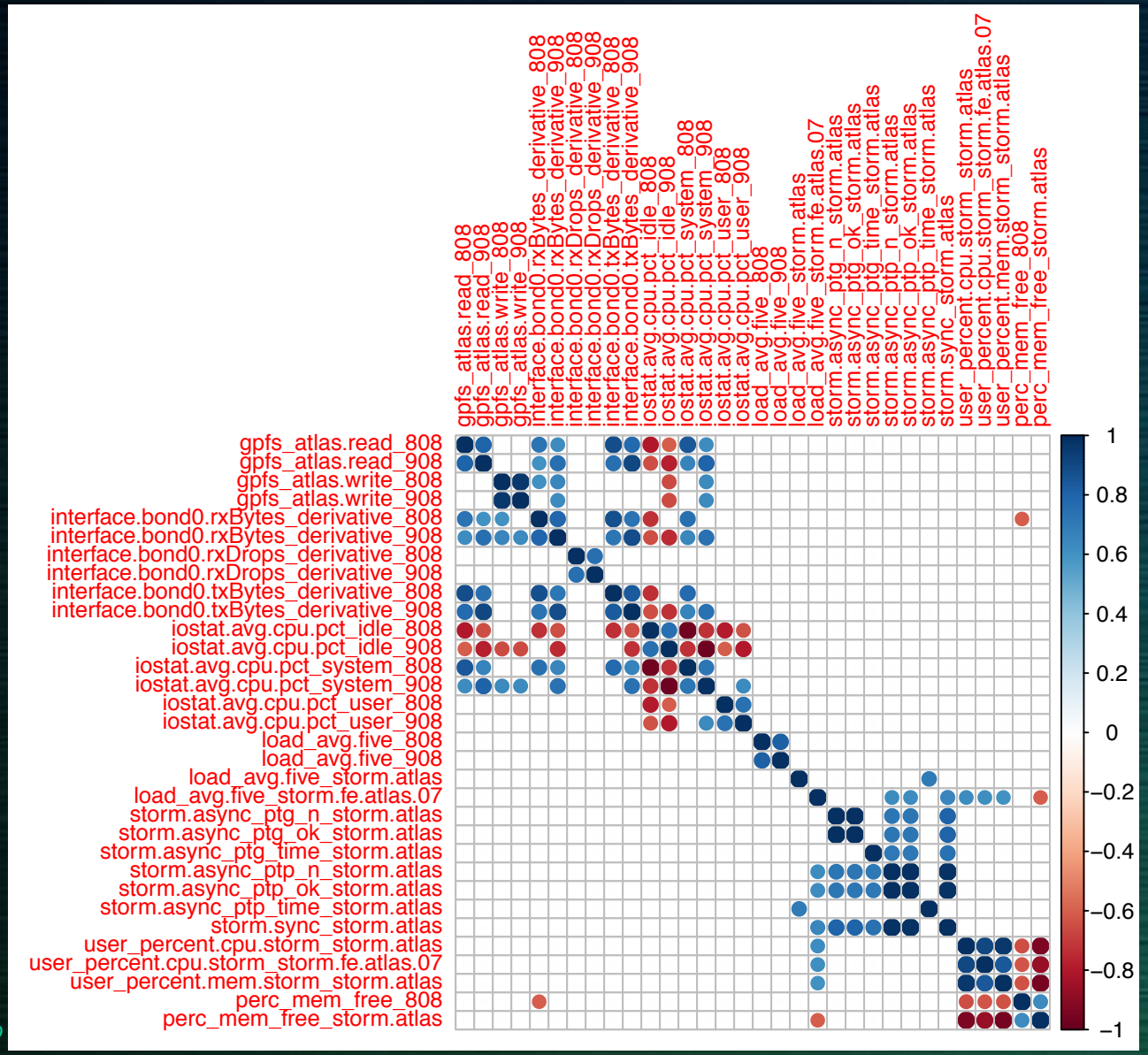
One hot encoding

timestamp	datetime	DN	requested_token	num	ip	DN_Atlas_Data_Management_YES	DN_Atlas_Data_Management_NOT	INFO	WARN	...
1.543622e+09	2018-12-01 00:00:00.010000	0	0	0	1	0	0	1	0	...
1.543622e+09	2018-12-01 00:00:00.032000	0	0	0	0	0	0	1	0	...

Final csv

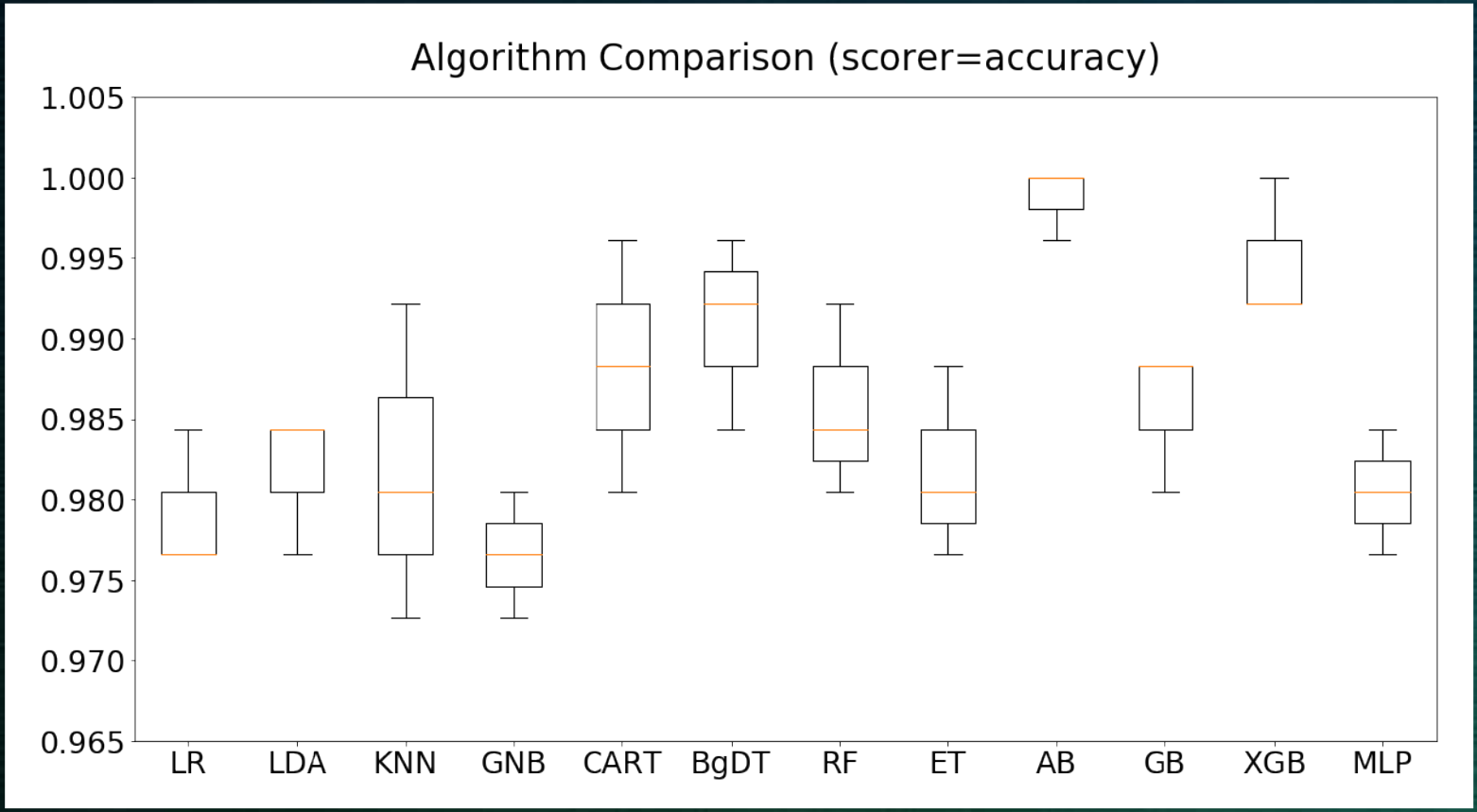
datetime	DN	requested_token	num	ip	DN_Atlas_Data_Management_YES	DN_Atlas_Data_Management_NOT	INFO	WARN	ERROR	...
2018-12-1 0:15:0	11613	10710	28816	24371	8727	2886	41800	0	0	...
2018-12-1 0:30:0	14580	13115	26994	32114	6972	7608	56970	0	0	...

Correlation matrix (InfluxDB case)



Correlation matrix of the more interesting InfluxDB metrics considering only “bad” days, with the absolute value of the correlation coefficients greater than 0.6

Build a ML model: comparison between different algorithms (InfluxDB case)

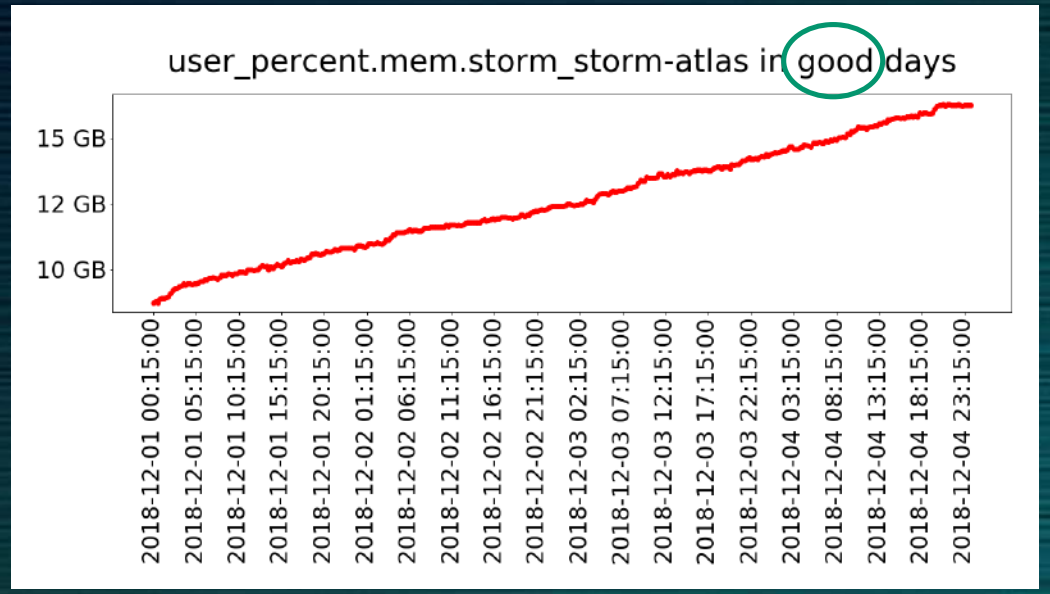


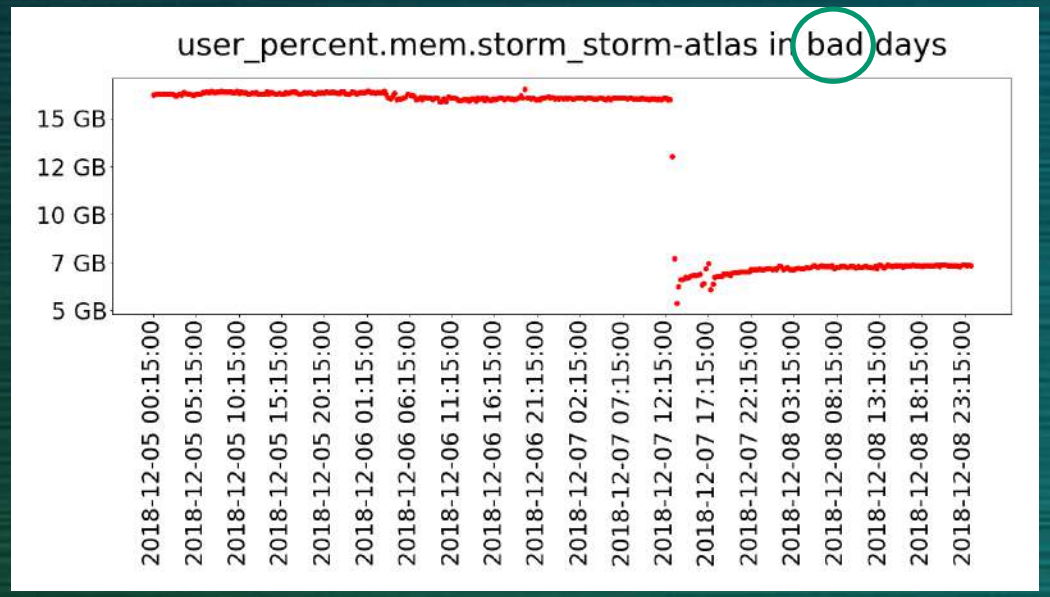
Legend of ML algorithms

LR: LogisticRegression
LDA: LinearDiscriminantAnalysis
KNN: KNeighborsClassifier
GNB: GaussianNB
CART: DecisionTreeClassifier
BgDT: BaggingClassifier
RF: RandomForestClassifier
ET: ExtraTreesClassifier
AB: AdaBoostClassifier
GB: GradientBoostingClassifier
XGB: XGBoostClassifier
MLP: MultiLayerPerceptronClassifier

Feature selection (InfluxDB case)

	Metric	Scoring
1	user_percent.mem.storm_storm-atlas	75
2	user_percent.cpu.storm_storm-atlas	37
3	perc_mem_free_storm-fe-atlas-07	36
4	perc_mem_free_808	31
5	interface.bond0.txBytes_derivative_808	25
6	perc_mem_free_storm-atlas	23
7	perc_mem_free_908	22
8	gpfs_atlas.write_808	20
9	storm.async_ptp_n_storm-atlas	18
10	interface.bond0.txBytes_derivative_908	17





Techniques used for the feature selection procedure:

- SelectKBest with the chi-squared statistical test
- Recursive Feature Elimination
- Principal Component Analysis (PCA)
- Feature Importance from ensembles of decision tree methods

Thank you for the attention!

 luca.giommi3@unibo.it

Frontend Logging

The Frontend stores information about the service status and about the SRM requests received and managed by the process.

```
12/01 03:48:11.701 Thread 41 -  INFO [41eec2e2-80a7-4c1b-82b4-d42ff57f0b7e]: Result for request
 'PTP status' is 'SRM_REQUEST_INPROGRESS'
12/01 03:48:11.717 Thread 13 -  INFO [1b3a9db9-1325-467f-9b8b-68347dbb6ad3]: process_request :
Connection from 2001:1470:ff80:12:8e23:c32e:495d:3846
12/01 03:48:11.849 Thread 13 -  INFO [1b3a9db9-1325-467f-9b8b-68347dbb6ad3]: Request 'BOL statu
s' from Client IP='2001:1470:ff80:12:8e23:c32e:495d:3846' Client DN='/DC=ch/DC=cern,
                                CN=Robot: ATLAS aCT 1' # Requested token '17b26868-7752-4e3
0-bcdb-c05d1f72c1b9'
12/01 03:48:11.852 Thread 13 -  INFO [1b3a9db9-1325-467f-9b8b-68347dbb6ad3]: Result for request
 'BOL status' is 'SRM_REQUEST_INPROGRESS'
```

Example of the storm-frontend-server.log file content.

GridFTP Logging

StoRM involves the GridFTP middleware component to perform file transfer operations.

```
[61998] Fri Nov 30 03:24:32 2018 :: Configuration read from /etc/gridftp.conf.
[61998] Fri Nov 30 03:24:32 2018 :: Server started in inetd mode.
[61998] Fri Nov 30 03:24:32 2018 :: New connection from: fts804.cern.ch:41956
[61998] Fri Nov 30 03:24:32 2018 :: DN /DC=ch/DC=cern
       CN=Robot: ATLAS Data Management successfully authorized.
[61998] Fri Nov 30 03:24:32 2018 :: User atlasprd045 successfully authorized.
[61998] Fri Nov 30 03:24:32 2018 :: Starting to transfer "/storage/gpfs_tsm_atlas/atlas/atlasda
tatape/data18_hi/RAW/other/data18_hi.00367134.physics_MinBias.daq.RAW/data18_hi.00367134.physic
s_MinBias.daq.RAW._lb0100._SFO-1._0001.data".
[61998] Fri Nov 30 03:24:36 2018 :: Finished transferring "/storage/gpfs_tsm_atlas/atlas/atlasd
atatape/data18_hi/RAW/other/data18_hi.00367134.physics_MinBias.daq.RAW/data18_hi.00367134.physi
cs_MinBias.daq.RAW._lb0100._SFO-1._0001.data".
[61998] Fri Nov 30 03:24:36 2018 :: Closed connection from fts804.cern.ch:41956
```

Example of the storm-gridftp-session.log file content.

InfluxDB Logging

The monitoring infrastructure at CNAF is based on InfluxDB as time series database to store data gathered from sensors.

```
[> select * from "one_month"."iostat.avg-cpu.pct_user" where host='ds-908.cr.cnaf.infn.it'
name: iostat.avg-cpu.pct_user
time                 domain          duration           host                   metric                  tag1           tag2  value
----                 ------          --------           ----                   ------                  ----           ----  -----
2019-01-30T00:00:00Z cr.cnaf.infn.it 1.9276666666666664 ds-908.cr.cnaf.infn.it metrics-iostat-extended gridftp-xrootd atlas 1.97
```

Example of a query to InfluxDB.

Summary of actions so far

- Handle and parse the log files to extract relevant information and design it to work automatically → Already an improvement respect to the current situation
- Create a ML model for each source individually taken → At each 15 minutes we have a prediction, in terms of probability, about its belonging to a good day or a bad day
- Do the correlation matrix → Check if there are unexpected relations between features
- Create a procedure for feature selection → Define which are the most relevant features for discrimination between good and bad days

What is missing

Use all the log sources. Currently missing:

- monitoring.log
- storm-backend.log
- heartbeat.log
- storm-backend-metrics.log

- Define other periods with anomalies in order to test the ML model produced
- create a model for each specific case

Problems in the critical period

Wrong configuration of the file system

The quota disk of GPFS is almost 30 PB, and the doubt quota disk was of the order of 200-300 TB during the problematic days, whereas in a normal day it is of the order of 1 TB. In this situation, the sum of the assigned memory plus the doubt quota was almost, or overcoming, the limit quota.

In this case, StoRM tells to GridFTP that there is free space even if it is not possible to write on the file system, hence the transfers fail.

Wrong configuration of the queues coming from the farm

"storm" (POSIX) access was not set as primary and the "rucio copytool" was selected, this causing an abnormal increase of access through StoRM-GridFTP and overload of the system.

InfluxDB metrics

Metric	Description
gpfs_atlas.*	* (read, write) reading and writing speed from the file system for the two GRIDFTP machines measured in bytes per second
interface.bond0.*xBytes	bytes * (r,t) received and transferred on the net interface bond0
interface.bond0.*xDrops	packet lost in * (r,t) reading and writing on the net interface bond0 measured in bytes
interface.bond0.*xErrors	* (r,t) reading and writing errors on the net interface bond0 measured in bytes
iostat.avg-cpu.pct_*	percentage of time where the cpu is * (idle, iowait, nice, user, system)
load_avg.five_*	average over 5 minutes of the CPU load average for the two GRIDFTP machines and the two Frontend services
storm.async_*_*_storm-atlas	average number of * (ptg, ptp), the average of those that fails, of those that are successfully ended, average in duration * (n, fail, ok, time) in the machine storm-atlas
storm.sync_storm-atlas	average number of synchronous operations for the storm user in the machine storm-atlas
user_percent.*.*	* (cpu, mem) CPU time, memory used by the storm process in the machine * (storm-atlas, storm-fe-atlas-07)
perc_mem_free_*	percentage of free memory of the machines where the two GRIDFTP and the two Frontend services are located

SelectKBest with the chi-squared statistical test

The chi-square test measures dependence between stochastic variables, so this function "weeds out" the features that are the most likely to be independent of class and therefore irrelevant for classification.

Recursive Feature Elimination

It recursively removes attributes and it builds a model on those attributes that remain. It uses the model accuracy to identify which attributes (and combination of attributes) contribute the most to predict the target attribute.

Techniques used for the feature selection procedure

Principal Component Analysis (PCA)

uses linear algebra to transform the dataset into a compressed form. The PCA procedure produces eigenvectors-eigenvalues pairs where an eigenvalue tells us how much variance there is in the data in the direction defined by the eigenvector.

Feature Importance from ensembles of decision tree methods

The importance of a feature is the increase in the prediction error of the model after we permuted the features values. Generally, importance provides a score that indicates how useful or valuable each feature was in the construction of the boosted decision trees within the model.

storm-frontend-server.log

	Metric	Scoring
1	BOL status	58
2	Abort request	55
3	num_surl	55
4	Rm	30
5	SRM_INTERNAL_ERROR	25
6	rpcResponseHandler_ReleaseFiles	18
7	PTG	16
8	DN_Atlas_Data_Management_NOT	16
9	rpcResponseHandler_Rm	15
10	Mv	12

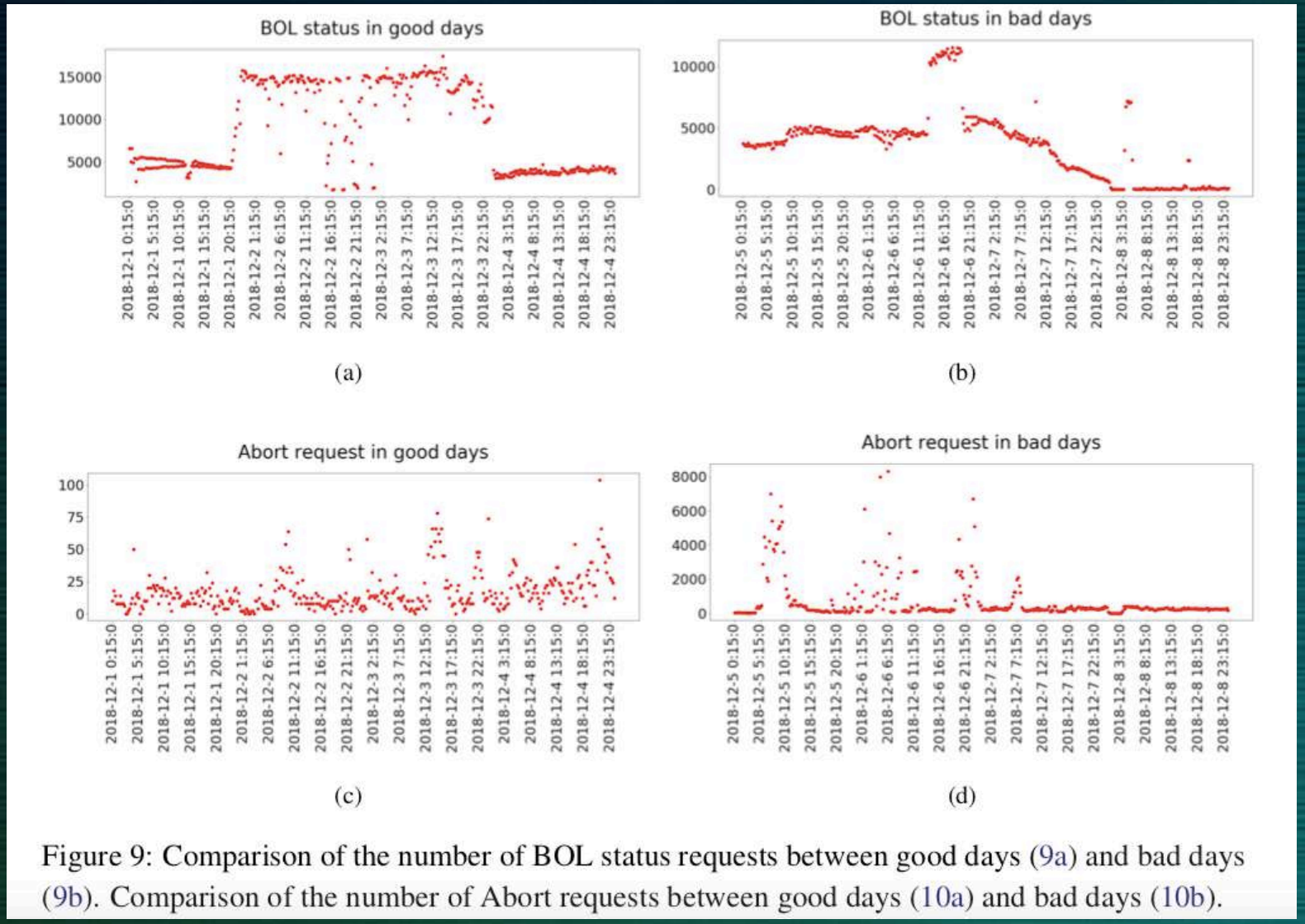


Figure 9: Comparison of the number of BOL status requests between good days (9a) and bad days (9b). Comparison of the number of Abort requests between good days (10a) and bad days (10b).

storm-gridftp-session.log

	Metric	Scoring
1	abort	62
2	disk_area_atlasdatatape	45
3	duration_mean	39
4	DN_ADM	37
5	globus_xio: System error in send	30
6	user_atlasprd	26
7	other_ip	25
8	disk_area_atlasmctape	25
9	Forcefully_terminating_process	19
10	duration_p95	16