Dataset popularity, analytics R&D

Valentin Kuznetsov

Aug 24, 2015

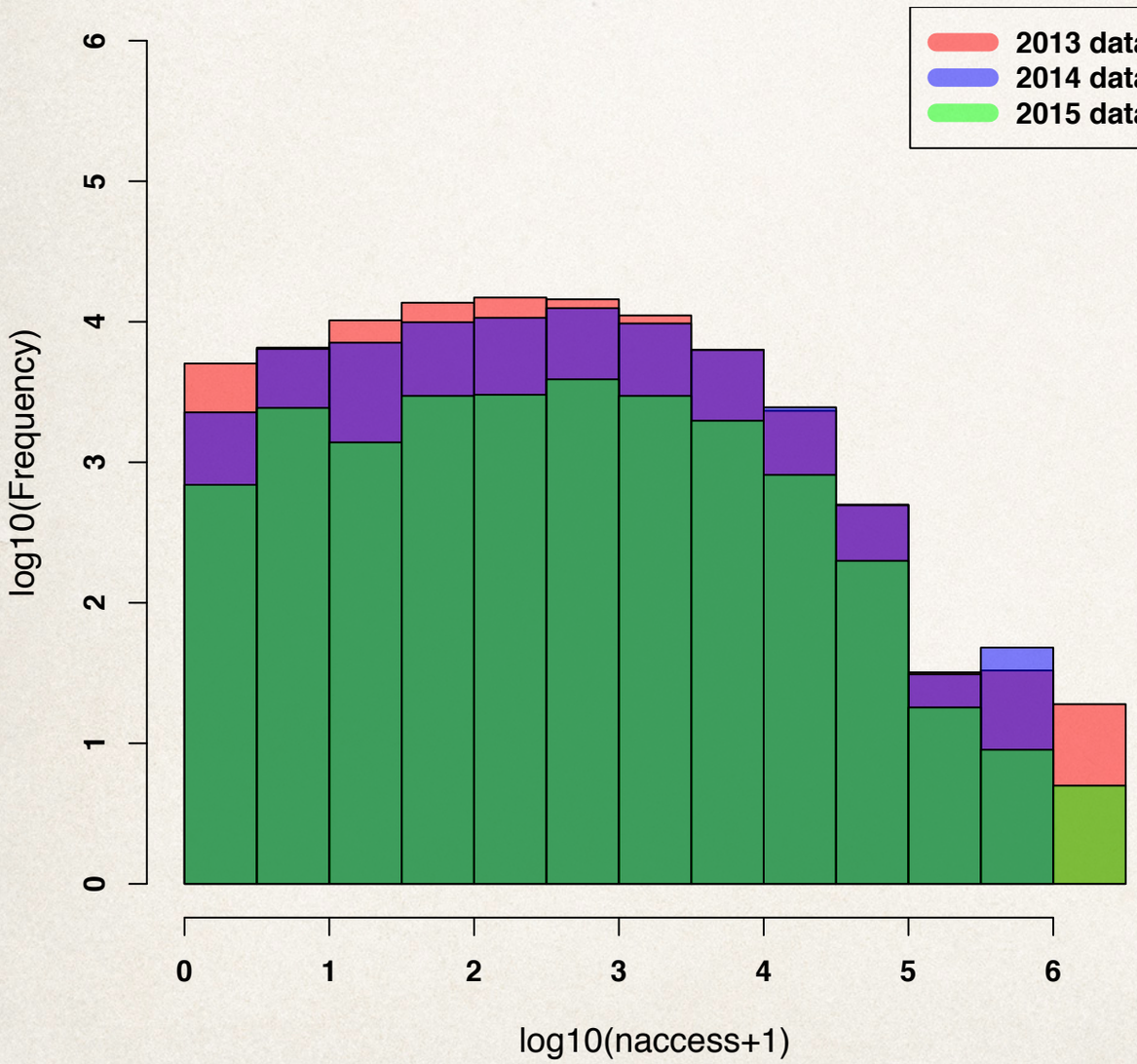
Highlights

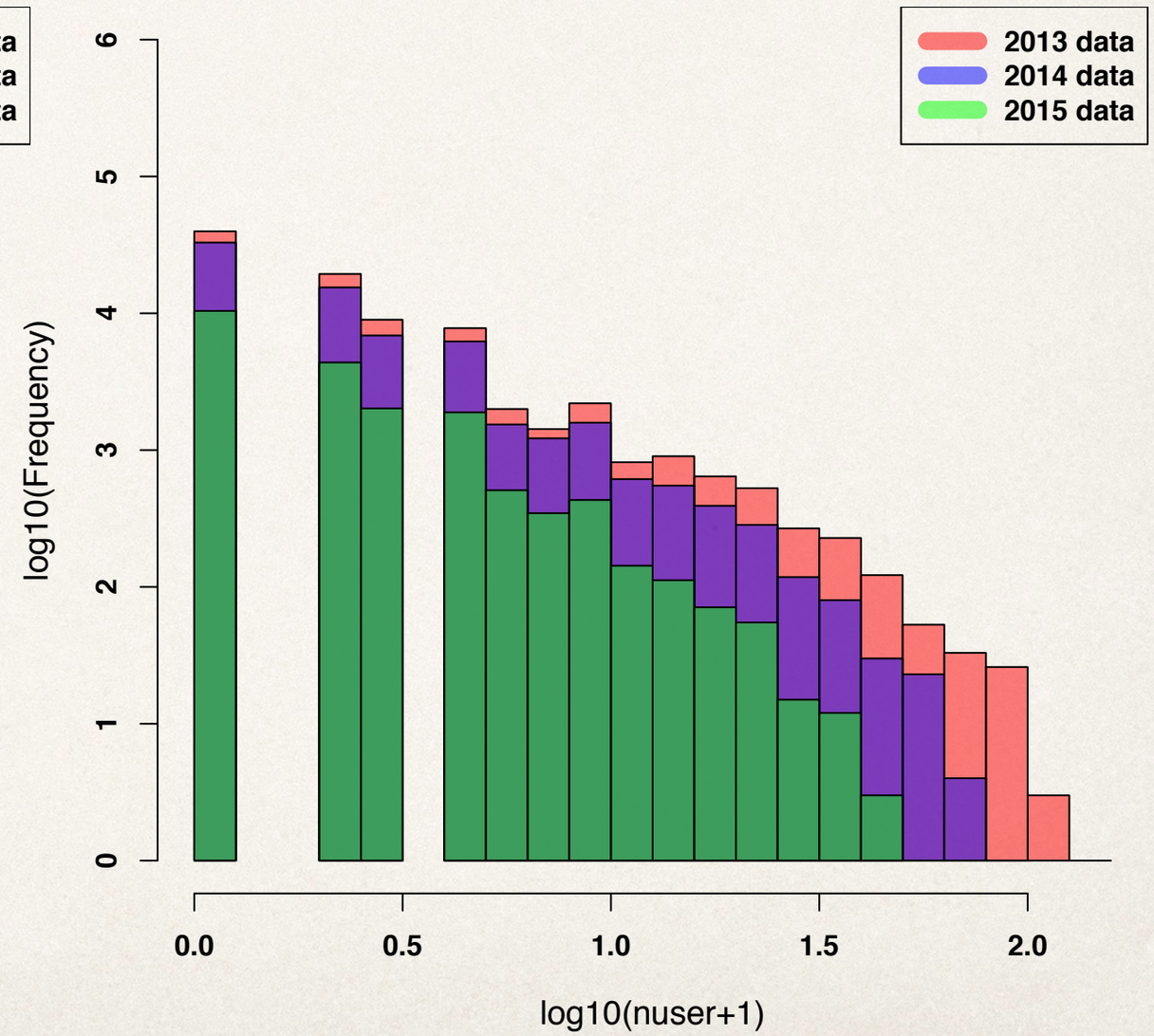
- Definition of popularity metric
  - Bologna student: Luca Giommi (supervisor D.Bonacorsi), late summer
- Seasonality effect
  - Cornell student: Ting Li, spring semester
- Rolling forecast
  - CERN summer OpenLab student: Siddha Ganju
- We concentrated only on AOD, AODSIM, MINIAOD, MINIAODSIM and USER dataset in these studies. The popularity of other datasets are driven by production machinery.

Popularity metric definition

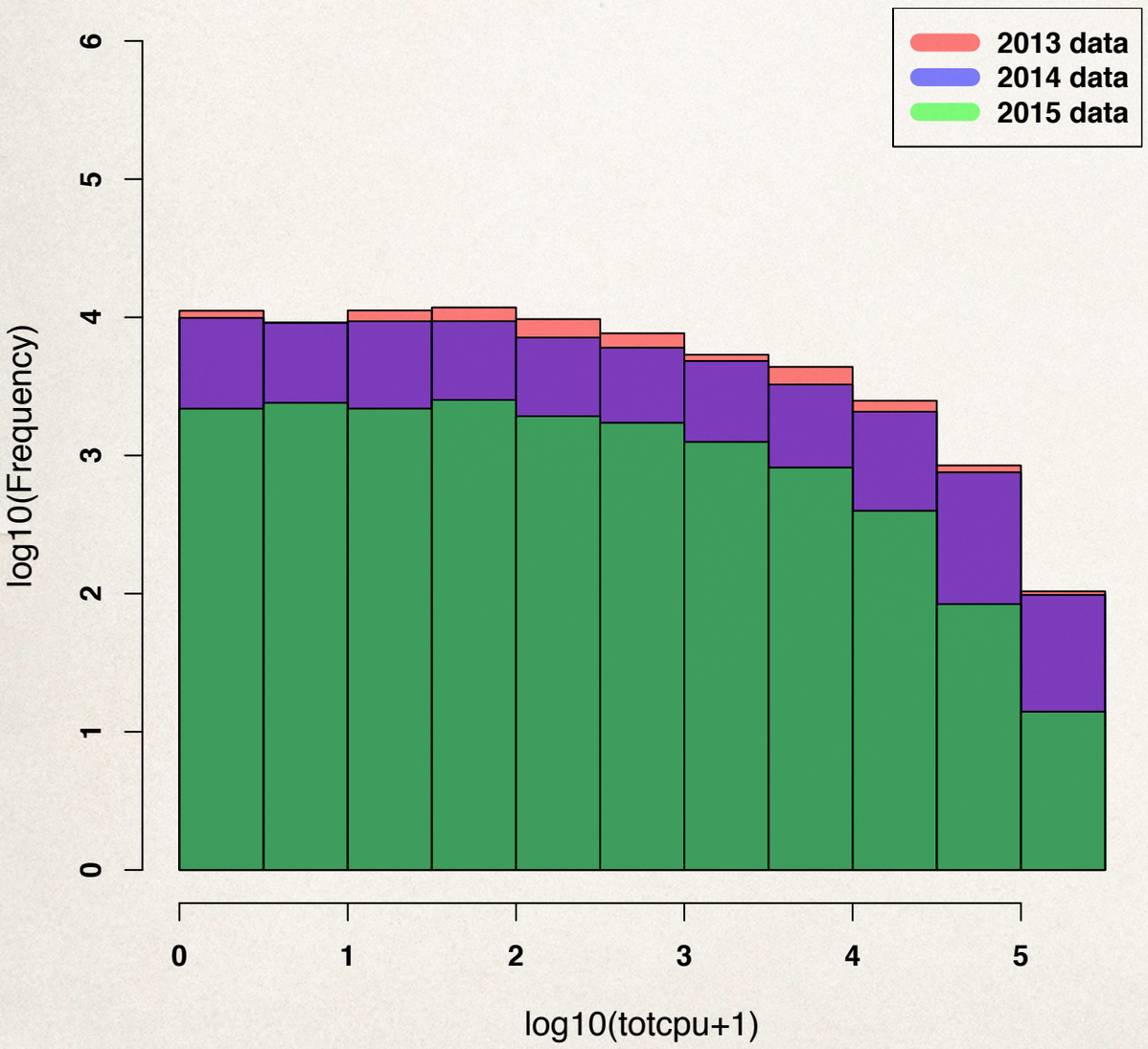
- We need to define what popularity means
  - this decision will influence model precision and cost function
- Use popularity DB information and optimize popularity metrics against False Positive yield
  - FP rate can be translated into data transfer overhead, while FN can be treated as job latency one.
- Perform studies of different cuts based on #accesses, #users/day, totcpu time metrics from popDB

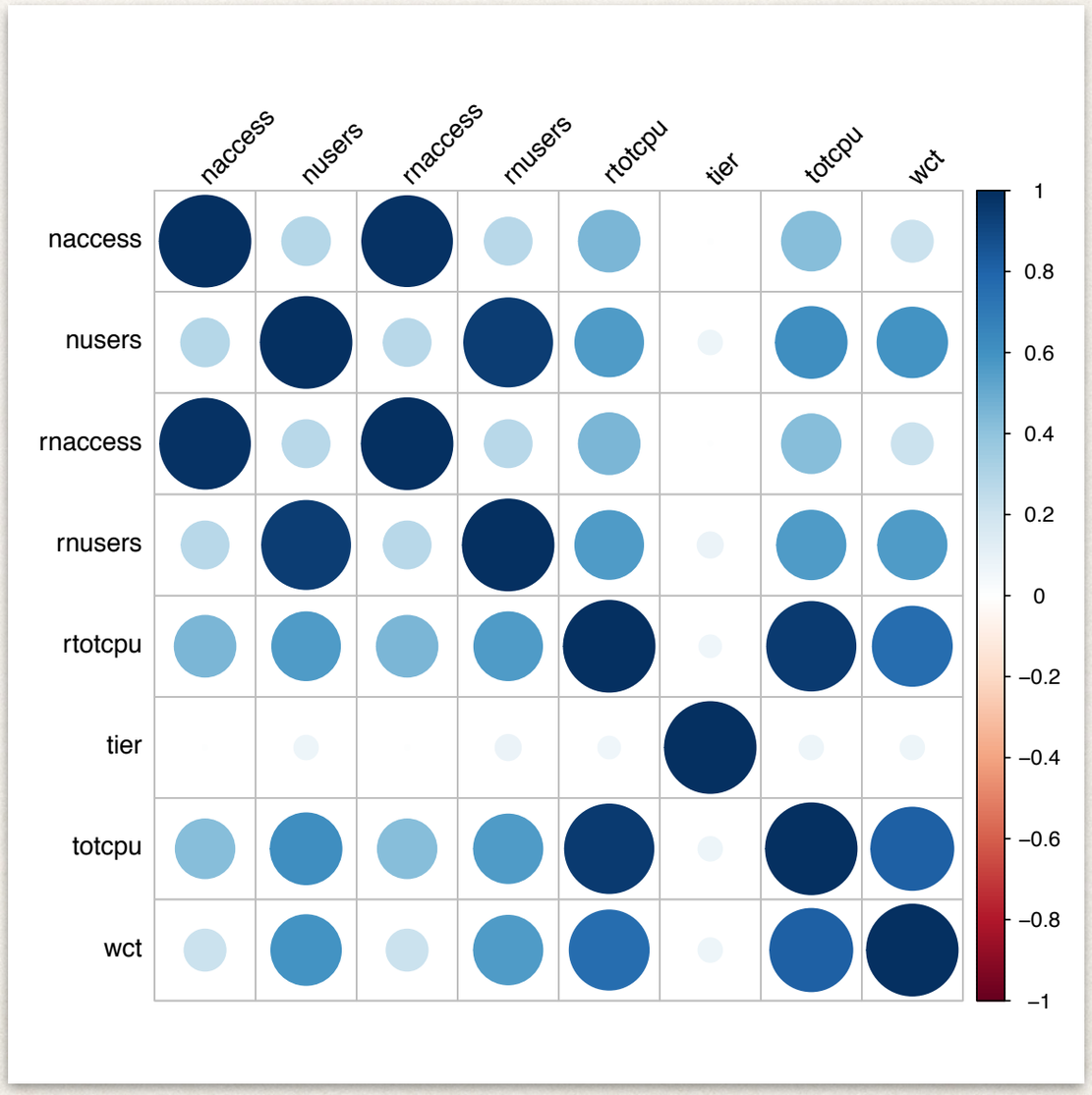
Popularity metrics





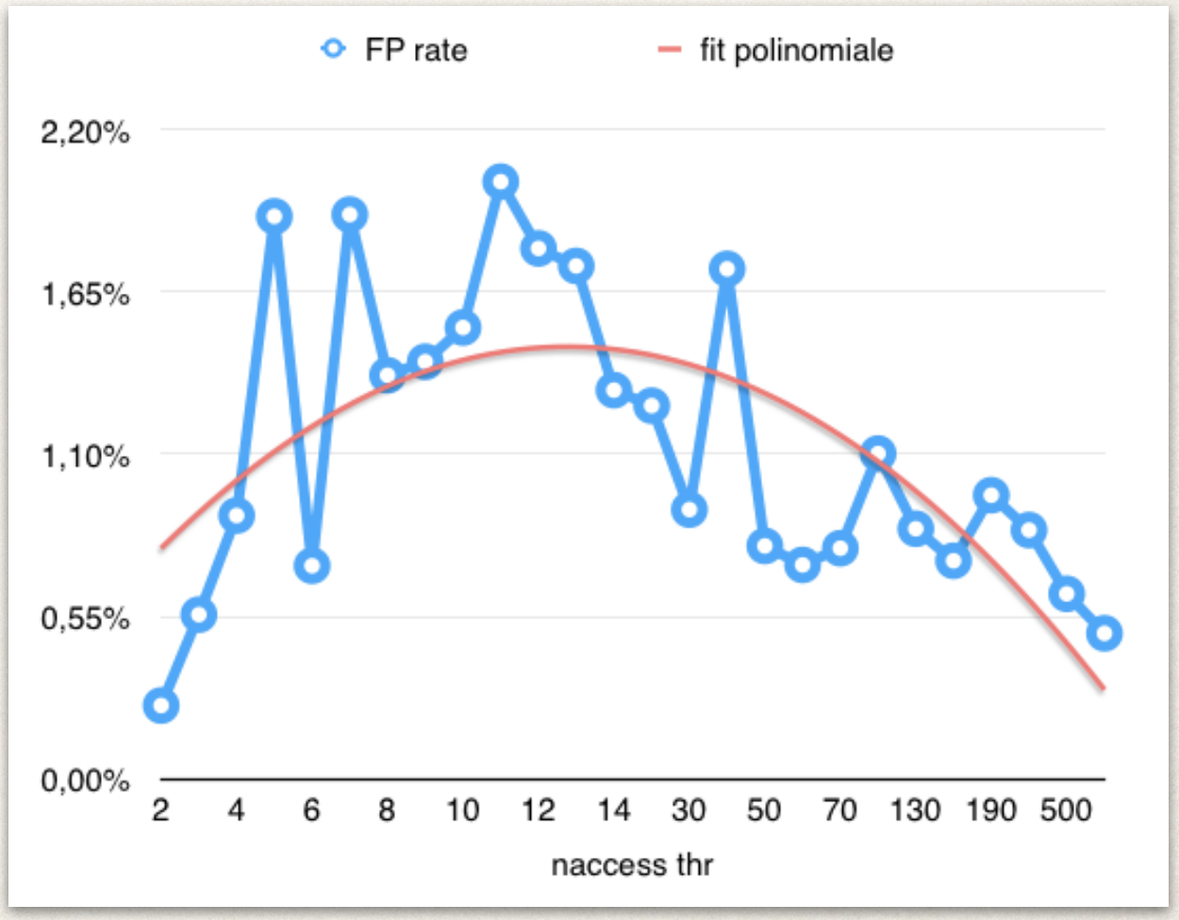
Popularity metrics





Popularity metric

- Define popular datasets as those which passed the cut, e.g. naccess>10
- Train model and look at FP yield
- Study cut effect on yield of FP vs data tier



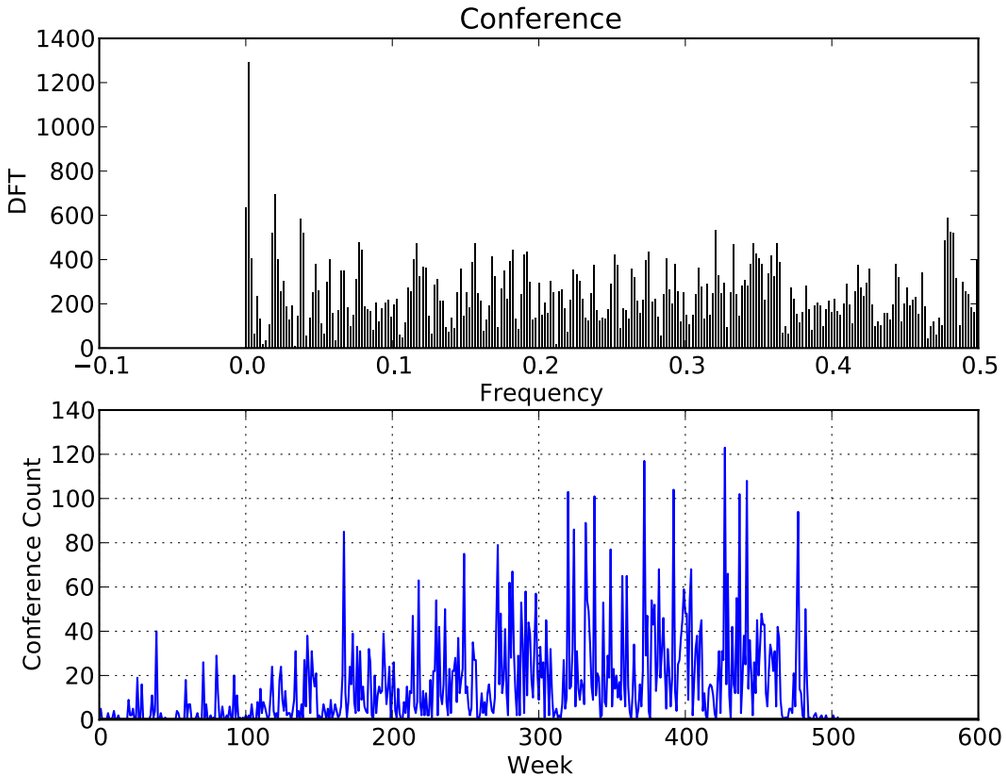
Popularity metrics

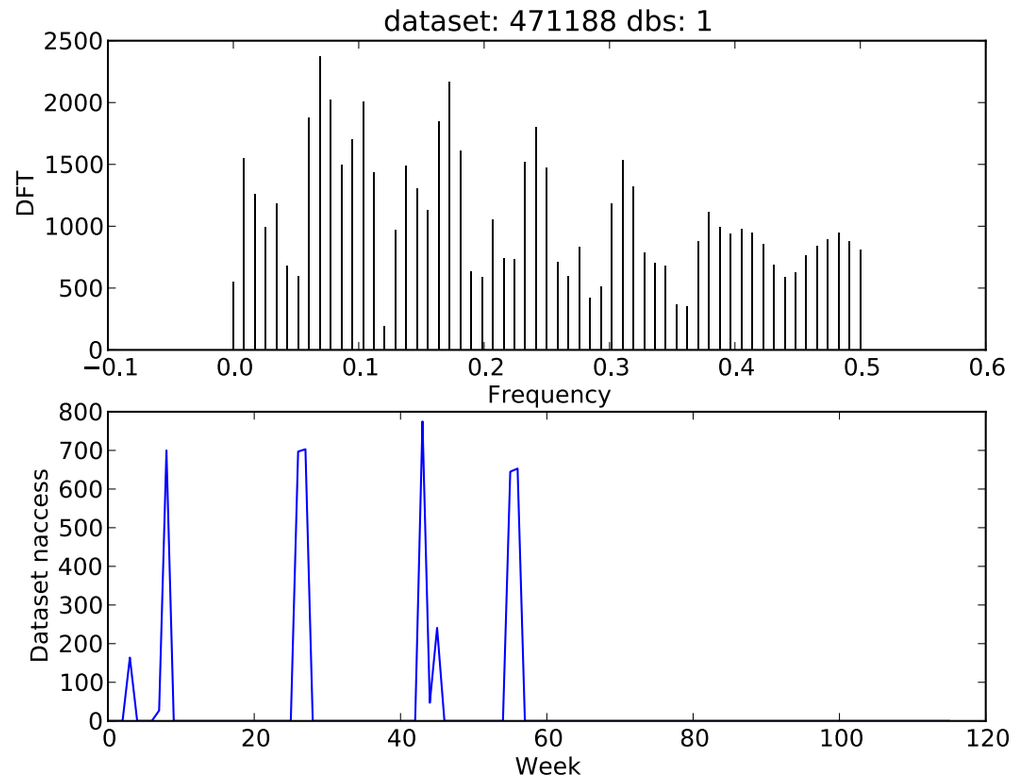
	2013	2013, nacces>10	2013, log(nuser)>2	2014	2014, naccess>10	2014, log(nuser)>2	2015	2015, naccess>10	2015, log(nuser)>2
AOD	7919 / 9%	7454 / 10%	2197 / 37.7%	4924 / 7.25%	4687 / 8%	1285 / 35%	999 / 5%	954 / 5.5%	218 / 26%
AODSIM	31925 / 37.5%	27351 / 37%	2924 / 50%	21090 / 31%	18825 / 32%	1547 / 42%	4563 / 22%	4184 / 24%	159 / 19%
MINIAOD	0	0	0	7 / 0.01%	6 / 0.01%	0	18 / 0.08%	17 / 0.09%	3 / 0.35%
MINIAOD SIM	0	0	0	1083 / 1.5%	792 / 1.3%	28 / 0.8%	2483 / 12%	1767 / 10%	129 / 15%
USER	38308 / 45%	33490 / 45.5%	480 / 8%	34127 / 50%	28777 / 49%	380 / 10%	8947 / 44%	7179 / 42%	69 / 8%
ALL	85115	73523	5819	67892	59222	3683	20381	17254	843

Seasonality effect

- Use CINCO database and define conference counters data frame (#conferences in N-th week from current date)
- Merge with datasets meta-data and study effect of seasonality
- Extract datasets with more than 10 records in a future time
- Use Discrete Fourier Transform (DFT) on time series by FFT algorithm and search for significant spikes that represents the frequency of seasonality
- Use random datasets to study seasonality effect

Seasonality effect

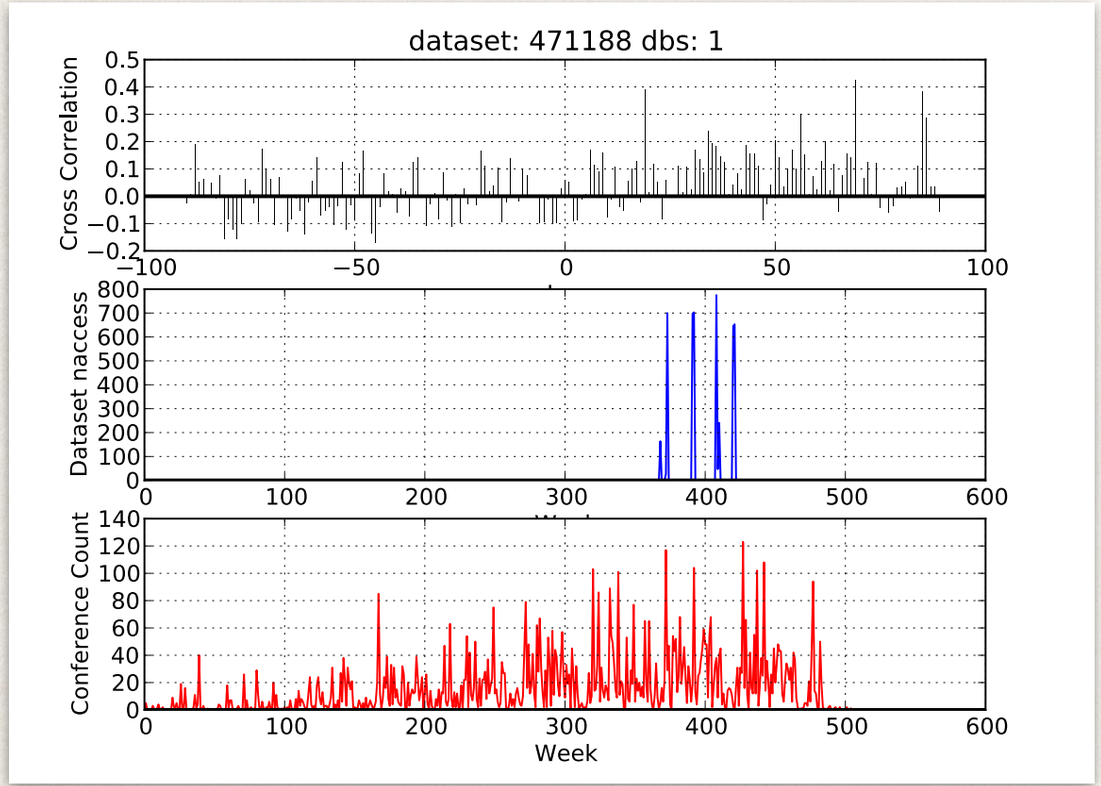


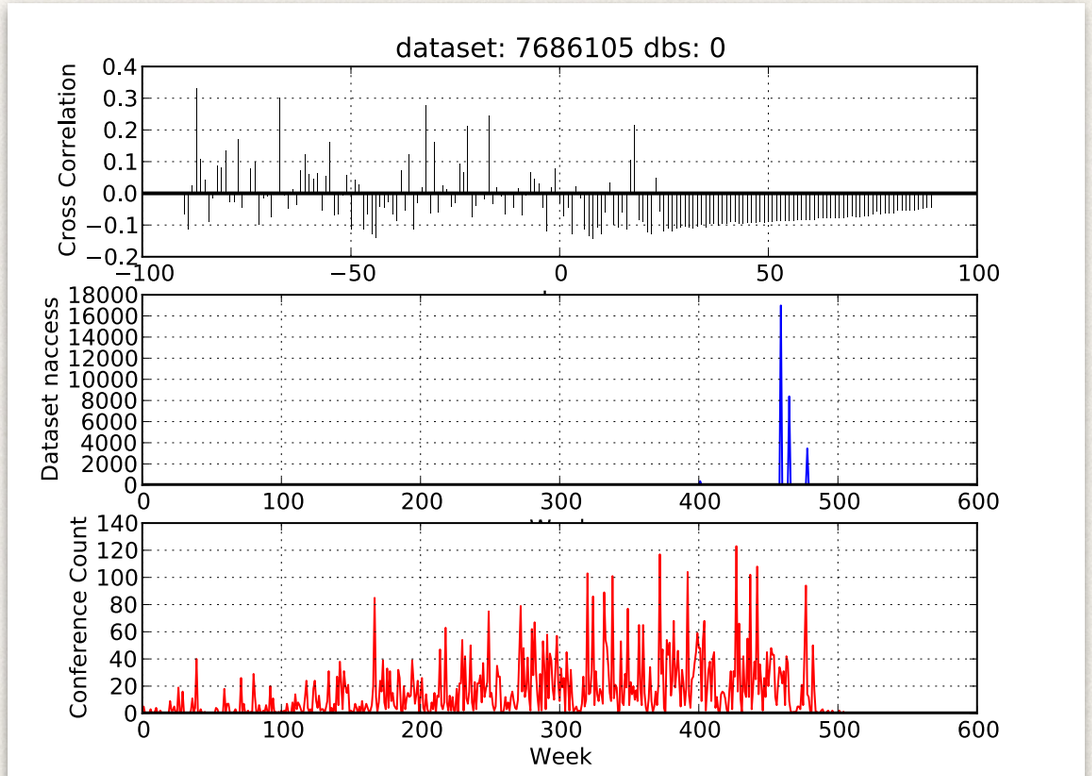


DFT of conference count
cover 2006-2015 years
periodicity every 50.5 week

DFT of access count/dataset
cover 2013-2015 years
periodicity 10-20 weeks

Dataset seasonality effect





Cross-correlation between conference count series and access count series of a dataset. We found that for some datasets cross correlation is peaking at a positive lag which means that future conference schedules can affect the current dataset access, while for others it peaks at a negative lag means that past conferences can still have residual influence on the current dataset access.

Seasonality effect summary

- Data shows some seasonality effect
- Conference counters can be used for prediction without other meta-data attributes, but they're less significant with respect to CMS meta-data attributes.
- Studies has been done with random datasets but further analysis for specific data-tier is interested to pursued

Rolling forecast approach

- Start with certain time interval, e.g. whole 2013 year
- Merge data from this interval and train the model
- Look-up data from the following week from the end of used time interval, e.g. use 2013 data for training and predict 20140101-20140108 week
- Make prediction for this week and find out efficiency of the algorithm
- Constrain ourselves only to AOD, AODSIM, MINIAOD, MINIAODSIM and USER data tiers
- Merge predicted week with previous data and repeat entire procedure

Rolling forecast approach

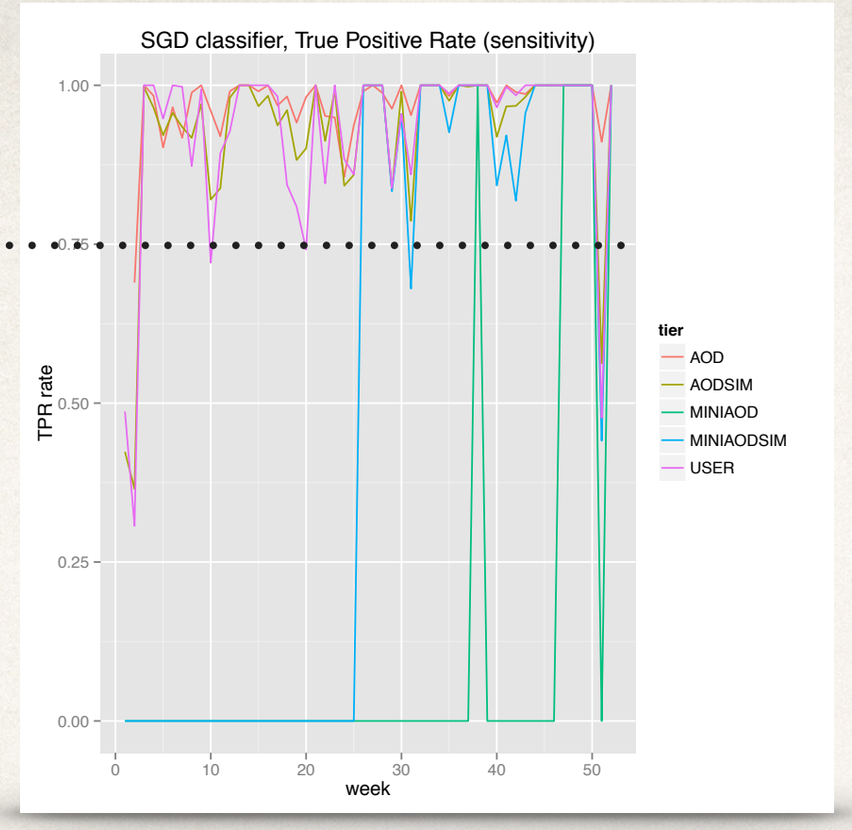
- Run multiple algorithms to check their performance over different period of time

- Compare algorithms results for consistency of predictions

- Measure algorithms efficiencies, benchmark their running time (CPU/RAM usage)

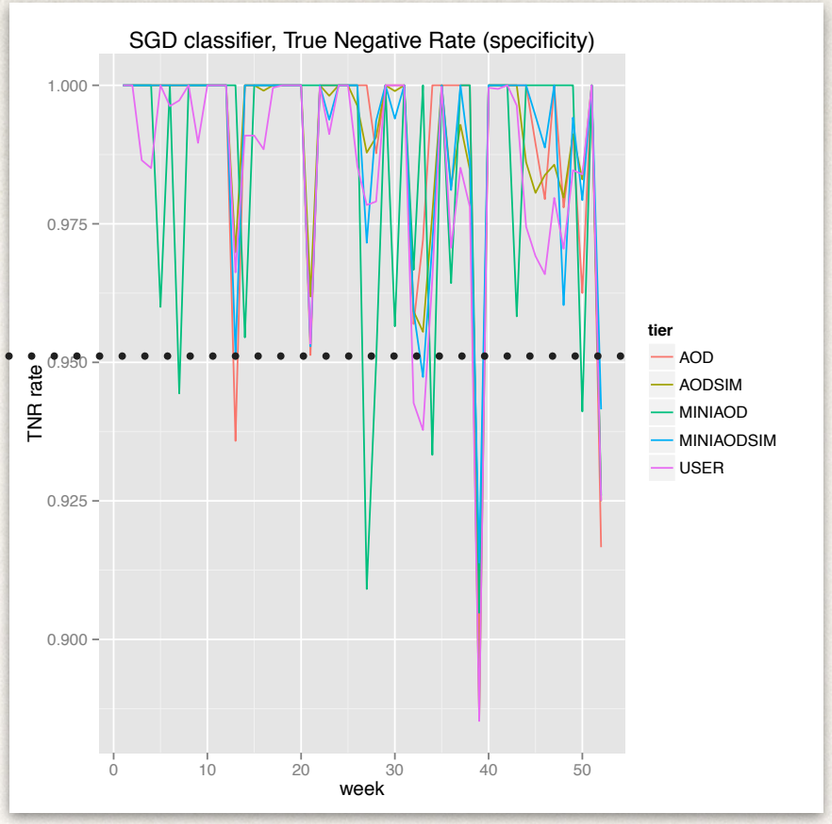
- Test ensemble model

75%



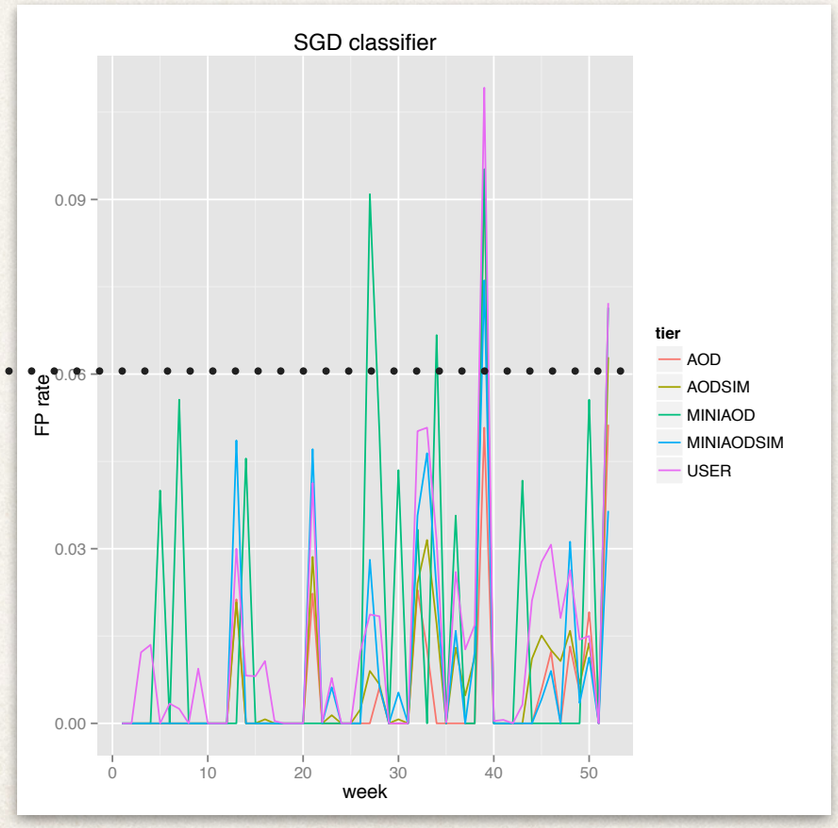
TPR=TP/(TP+FN)

95%



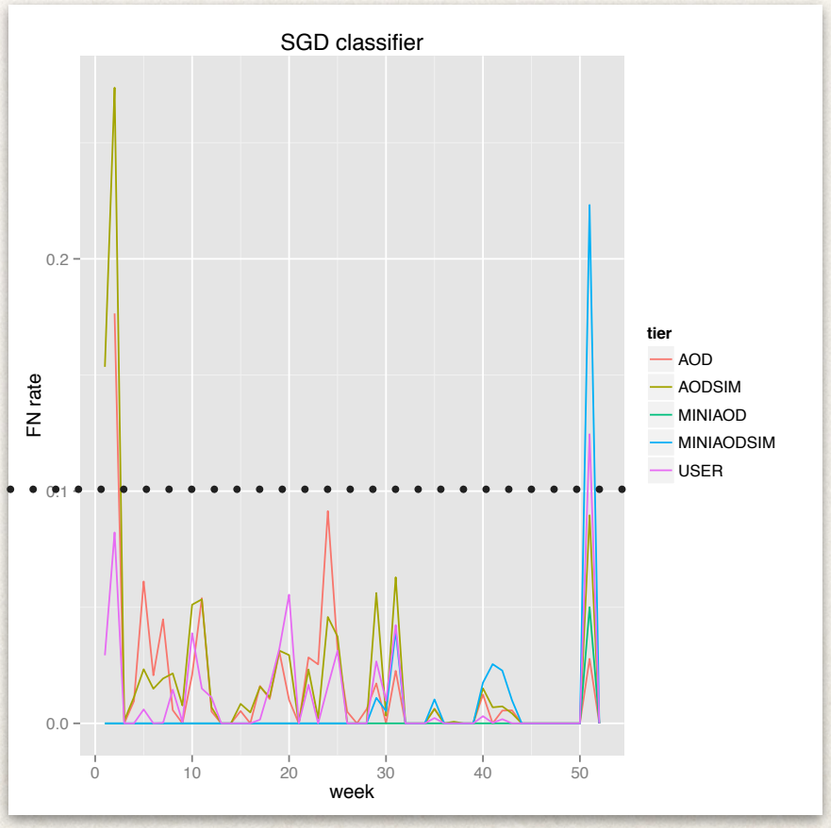
TNR=TN/(TN+FP)

6%



FP

10%

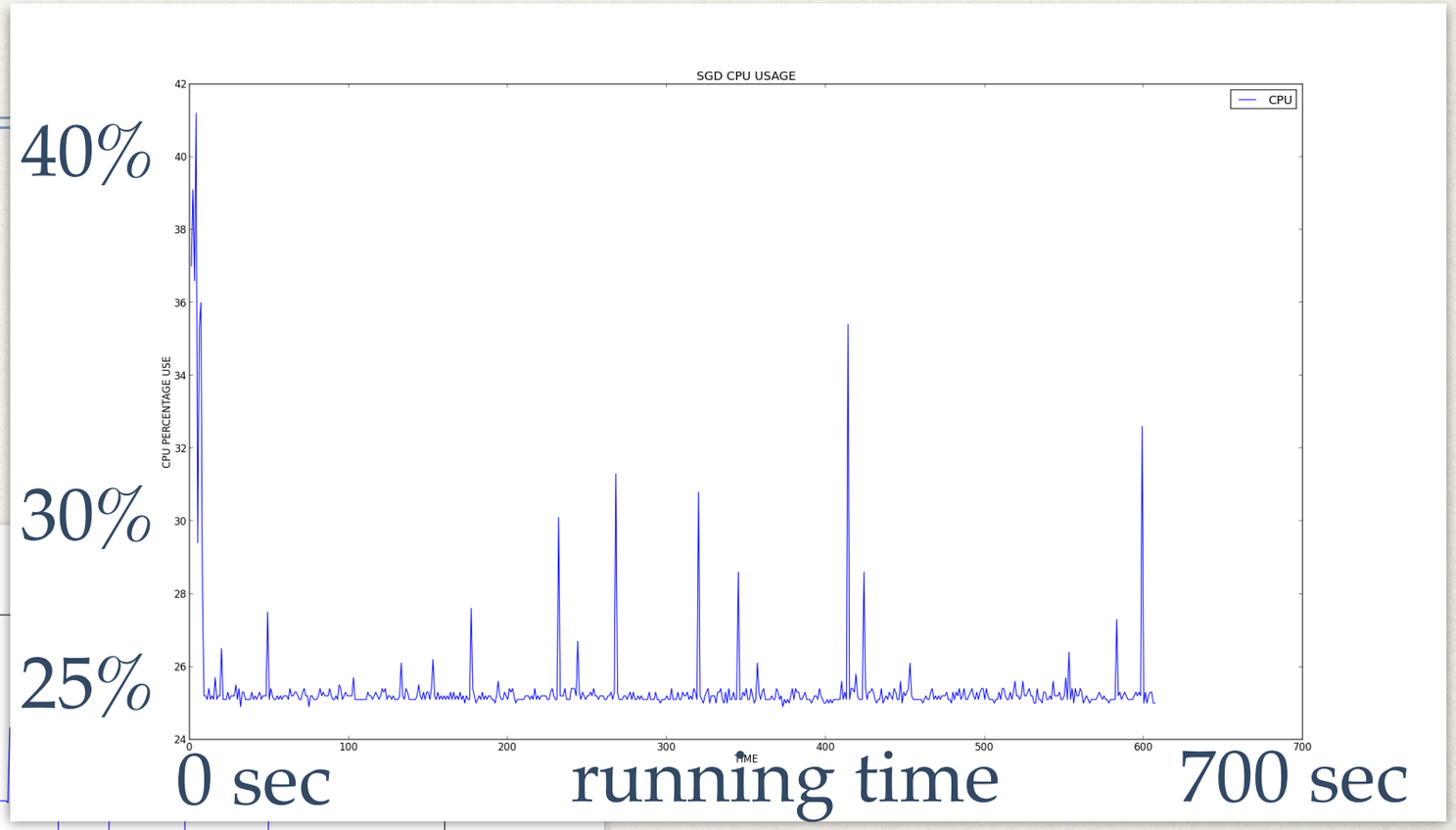


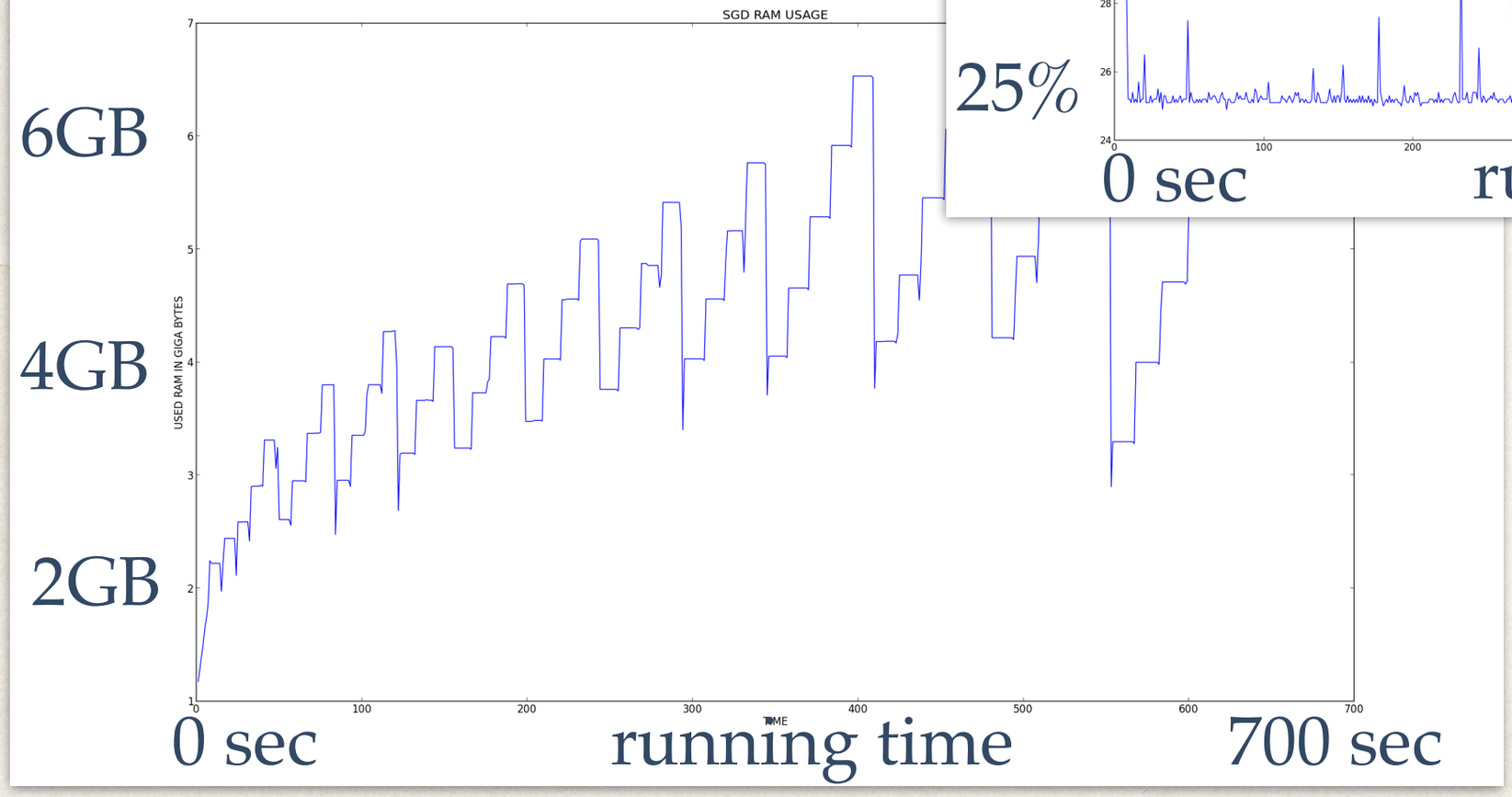
FN

Popularity statistics

Data Tier	TPR= TP/(TP+FN)	TNR= TN/(TN+FP)	FPR= FP/(FP+TN)	PPV= TP/(TP+FP)	NPV= TN/(TN+FN)	FP	FN
AOD	0.97+-0.05	0.99+-0.02	0.01+-0.02	0.99+-0.02	0.97+-0.06	0.005+-0.011	0.015+-0.029
AODSIM	0.93+-0.13	0.99+-0.02	0.01+-0.02	0.97+-0.06	0.97+-0.05	0.008+-0.016	0.021+-0.045
MINIAOD	0.11+-0.32	0.99+-0.02	0.01+-0.03	0.09+-0.28	0.99+-0.01	0.014+-0.026	0.001+-0.007
MINIAODSIM	0.49+-0.48	0.99+-0.02	0.01+-0.02	0.47+-0.47	0.99+-0.04	0.009+-0.016	0.007+-0.031
USER	0.93+-0.15	0.98+-0.02	0.02+-0.02	0.90+-0.15	0.99+-0.03	0.014+-0.021	0.011+-0.023

CPU & RAM usage (SGD classifier)





FP/FN rates

- FP/FN rates can be easily translated into data-transfer overhead or latency of CRAB jobs, respectively

- In 2014 we recorded ~565 ± 300 datasets every week in DBS. Using 1% FP rate and ~2TB as average size of dataset this translates into ~10TB of additional data transfer/site usage.

- FN rate can be interpreted as a “normal” latency of CRAB jobs waiting for “hot” datasets if such datasets reside on a single site.

Prediction summary

- We understand machinery and be able to run DCAFPilot as a service
  - cronjobs are used to generate dataframes, run model and check predictions; we have a web service with API to access this information
- On average we have ~60K AOD+USER datasets per year and that represent main interest for dataset popularity prediction
- From performed studies we see that we can have FP/FN errors on the level of few percent through the year, but errors are tier and week dependent
  - FP rate of 1% => 10TB of additional data-transfer
- We clearly observed when MINOAOD/MINIAODSIM data gain popularity
- There is a seasonality effect for some datasets, this observation can be further studied via FFT and conference count studies

Paper

- Paper draft can be found here:
  - https://www.dropbox.com/s/ldjxme0upoonbfb/paper.pdf?dl=0