ANOMALY DETECTION @ CELTRA

Project report on October 25, 2014

University of Ljubljana, Faculty of Computer and Information Science
Contact e-mail: lovro.subelj@fri.uni-lj.si

The main goal of the project is the development of different models for anomaly detection in Celtra data and their evaluation in practical scenarios. Particularly, the models should address various deploy, operational, performance and other issues identified by Celtra team. The data comes in the form of aggregated hourly or daily metrics including different counts and rates, and can be filtered according to different dimensions including accounts, campaigns and creatives, and platforms, SDK, formats and devices. The developed models adopt techniques from the fields of data mining, stream mining, statistical analysis, network analysis and link analysis, and are presented in the following.

STREAM MINING

Problem definition. Stream mining aims to address *deploy and operational issues, and business performance issues*, through time. Particularly, the goal is to detect substantial changes and anomalous behavior in different count and rate metrics at particular points in time.

Data representation. The data is represented as a temporal stream for each of the adopted count and rate metrics. These include one session and one video count metric, and five session and four video rate metrics. The streams are further supplemented with different aggregate, lagged, periodic and other attributes, and normalized and preprocessed accordingly.

Methods analysis & selection. Analysis of the resulting temporal streams shows that a large majority of these are relatively consistent through time and reveal clear periodic behavior. The latter can be assessed by mining daily, weekly and monthly periodic attributes. Anomalous behavior in different count and rate metrics is thus detected by stream regression algorithms trained over the specified time period, while the best is selected by a sliding window evaluation. Substantial changes in metrics are finally determined by concept drift detection for the specified sensitivity.

Stream preprocessing. Stream attribute normalization, missing data imputation and outlier instance detection, and mode, mean, aggregate, lagged and periodic attributes construction.

Stream regression. Mean, mode, nearest neighbor, model rules, regression tree, stochastic gradient descent, support vector machines, ensemble-based and meta regression algorithms.

Stream postprocessing. Regression correction, explanation and reliability, and concept drift detection.

Stream evaluation. Sliding window performance evaluation, and time and memory complexity assessment.

Results analysis & discussion. Stream mining effectively detects changes and anomalous behavior in different count and rate metrics at particular points in time. Best performance and also complexity is obtained by instance-based regression models like the nearest neighbor algorithm based on a sliding window. Nevertheless, certain rate metrics with particularly inconsistent behavior can not be predicted very accurately and thus only substantial changes in such metrics can be detected.

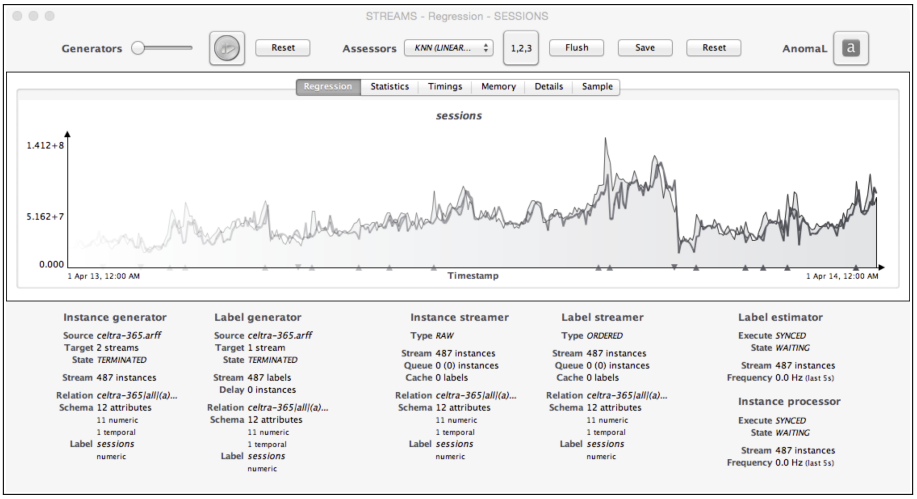


Figure 1: Stream mining for sessions count on April 1, 2014.

Final assessment. Stream mining provides a prominent model for the detection of deploy and operational issues, and business performance issues, while its use in practice is unknown.

PoC. PoC implementation is provided within STREAMS repository as documented Java library that streams data for the specified period from MySQL database or Celtra API and shows the current results, performance and complexity of stream mining within an interactive GUI (see Figure 1).

Estimated effort. The overall effort is estimated to $20x$, where x is a certain time period.

ANOMALY MINING

Problem definition. Anomaly mining aims to address *deploy and operational, general ecosystem and business performance issues* through time. Particularly, the goal is to detect substantial changes and anomalous behavior in different count and rate metrics of platforms, SDK, formats, devices and combinations of these at particular points in time.

Data representation. The data is represented as temporal streams for platforms, SDK, formats, devices and combinations of these for each of the adopted count and rate metrics. These include one session and one video count metric, and five session and four video rate metrics. The streams are further supplemented with different periodic and other attributes.

Methods analysis & selection. Analysis of the resulting temporal streams shows that a large majority of these are relatively consistent through time and reveal clear periodic behavior. Substantial changes and anomalous behavior in different count and rate metrics of platforms, SDK, formats, devices and combinations of these are thus detected by stream mining methods presented above. Due to an exponential number of combinations of different dimensions, the selection of relevant combinations can not be done by hand and has to be automatized.

Stream mining. See stream mining methods above.

Concept detection. Parametric tests, standardized residuals, supervised and unsupervised Page-Hinkley tests.

Dimension selection. Reliability-weighted entropy of dimension anomaly distributions.

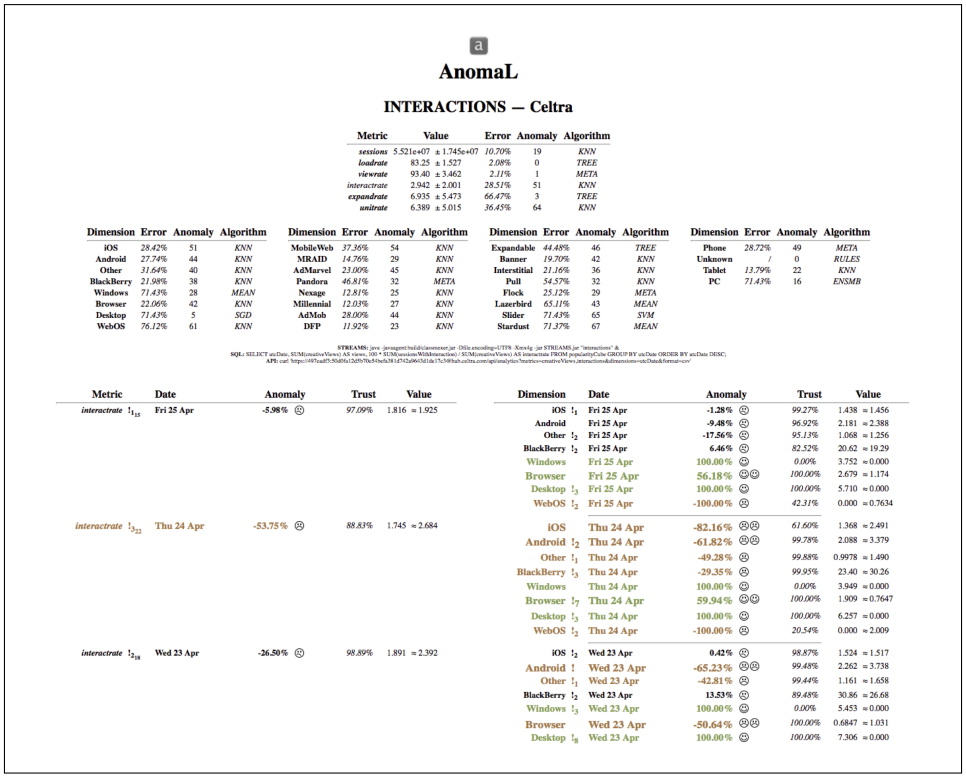


Figure 2: Anomaly mining for creative interaction rate on April 25, 2014.

Results analysis & discussion. Anomaly mining effectively detects changes and anomalous behavior in different count and rate metrics of platforms, SDK, formats, devices and combinations of these through time. Best performance is obtained by stream mining methods presented above and selection of relevant combinations of dimensions based on their weighted entropy. Nevertheless, certain rate metrics with particularly inconsistent behavior can not be predicted very accurately and thus only substantial changes in such metrics can be detected.

Final assessment. Anomaly mining provides a prominent model for the detection of deploy and operational, general ecosystem and business performance issues, while it is already used in practice.

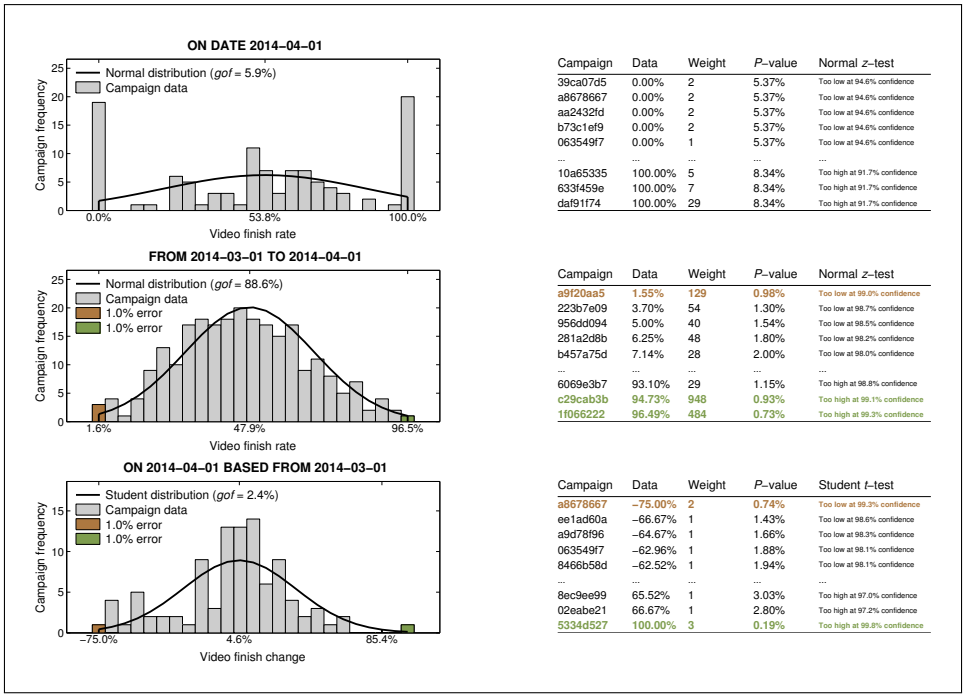
PoC. PoC implementation is provided within STREAMS repository as Java code that collects data for the specified period from MySQL database and outputs the results of anomaly detection over all metrics and dimensions as a series of HTML documents including visual reports (see Figure 2).

Estimated effort. The overall effort is estimated to $15x$, where x is a certain time period.

STATISTICAL ANALYSIS

Problem definition. Statistical analysis aims to address *account, campaign and creative performance issues* both in time and comparison. Particularly, the goal is to detect statistically significant changes in different rate metrics of particular accounts, campaigns, creatives and points in time.

Data representation. The data is represented as a set of account, campaign and creative distributions for each of the adopted rate metrics. These include four session and four video rate metrics. Each set of distributions includes distribution over the specified time period, distribution for the current point in time and distribution of the relative changes based on the specified time period.



Campaign	Data	Weight	*P*–value	Normal *z*–test
39ca07d5	0.00%	2	5.37%	Too low at 94.6% confidence
a8678667	0.00%	2	5.37%	Too low at 94.6% confidence
aa2432fd	0.00%	2	5.37%	Too low at 94.6% confidence
b73c1ef9	0.00%	2	5.37%	Too low at 94.6% confidence
063549f7	0.00%	1	5.37%	Too low at 94.6% confidence
...	...	...	...	...
10a65335	100.00%	5	8.34%	Too high at 91.7% confidence
633f459e	100.00%	7	8.34%	Too high at 91.7% confidence
daf91f74	100.00%	29	8.34%	Too high at 91.7% confidence

Campaign	Data	Weight	*P*–value	Normal *z*–test
a9f20aa5	1.55%	129	0.98%	Too low at 99.0% confidence
223b7e09	3.70%	54	1.30%	Too low at 98.7% confidence
956dd094	5.00%	40	1.54%	Too low at 98.5% confidence
281a2d8b	6.25%	48	1.80%	Too low at 98.2% confidence
b457a75d	7.14%	28	2.00%	Too low at 98.0% confidence
...	...	...	...	...
6069e3b7	93.10%	29	1.15%	Too high at 98.8% confidence
c29cab3b	94.73%	948	0.93%	Too high at 99.1% confidence
1f066222	96.49%	484	0.73%	Too high at 99.3% confidence

Campaign	Data	Weight	*P*–value	Student *t*–test
a8678667	–75.00%	2	0.74%	Too low at 99.3% confidence
ee1ad60a	–66.67%	1	1.43%	Too low at 98.6% confidence
a9d78f96	–64.67%	1	1.66%	Too low at 98.3% confidence
063549f7	–62.96%	1	1.88%	Too low at 98.1% confidence
8466b58d	–62.52%	1	1.94%	Too low at 98.1% confidence
...	...	...	...	...
8ec9ee99	65.52%	1	3.03%	Too high at 97.0% confidence
02eabe21	66.67%	1	2.80%	Too high at 97.2% confidence
5334d527	100.00%	3	0.19%	Too high at 99.8% confidence

Figure 3: Campaign distributions for video finish rate on April 1, 2014.

Methods analysis & selection. Analysis of the resulting distributions shows that a large majority of these have a bell-shaped structure and exponentially decreasing tails. This is expected under the assumption of independence of accounts, campaigns and creatives, while it also excludes distributions with polynomial tails like power-laws as a valid fit. Other distributions are fitted using well-known techniques, while the best is determined by χ^2 goodness-of-fit test. Statistically significant changes in metrics of particular accounts, campaigns and creatives are finally determined by a standard statistical hypothesis testing for the specified probability of error.

Probability distributions. Uniform, normal or Gaussian, log-normal, Poisson, Student, extreme value, exponential, inverted exponential, power-law and other heavy-tailed distributions.

Hypothesis testing. Statistical hypothesis tests corresponding to distributions and χ^2 goodness-of-fit test.

Results analysis & discussion. Statistical analysis effectively detects changes in rate metrics of particular accounts, campaigns, creatives and points in time. Nevertheless, metrics with particularly low or high values cannot be properly fitted by any of the distributions and thus allow only for the detection of either significantly large or significantly small changes in rate metrics.

Final assessment. Statistical analysis provides a prominent model for the detection of account, campaign and creative performance issues, while it is already regularly used in practice.

PoC. PoC implementation is provided within STATS repository as Matlab script that collects data for the specified period from MySQL database and outputs the results of statistical analysis over all metrics and dimensions as a series of vector figures and reports (see Figure 3). These are then merged into a single document using a LaTeX script.

Estimated effort. The overall effort is estimated to $10x$, where x is a certain time period.

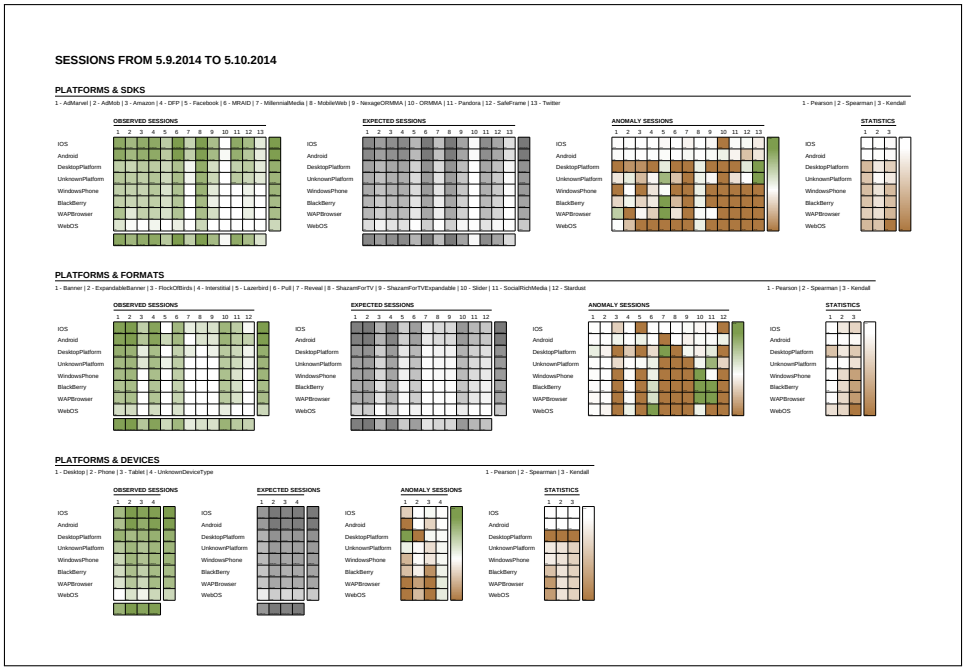


Figure 4: Platform distributions for sessions count on October 5, 2014.

MULTIVARIATE ANALYSIS

Problem definition. Multivariate statistical analysis aims to address *general ecosystem issues and campaign performance issues* over the specified period in time. Particularly, the goal is to detect substantial inconsistencies in different count metrics of particular campaigns, platforms, SDK, formats, devices and combinations of these.

Data representation. The data is represented as a set of campaign, platform, SDK, format and device joint and marginal distributions for each of the adopted count metrics. These include one session and one video count metric. Each set of distributions includes distributions based on count frequencies observed from the data, expected distributions under the assumption of independence and anomaly distributions based on the difference between of the latter.

Methods analysis & selection. Analysis of the resulting distributions shows that these extremely nonuniform and are better considered on a logarithmic scale. Under the assumption of independence, the joint distributions can be estimated from the marginal distributions. Substantial inconsistencies in count frequencies of particular campaigns, platforms, SDK, formats and devices can thus be determined by comparison between the observed and expected distributions using statistical goodness-of-fit tests or by considering measures of correlations between the distributions.

Probability distributions. Joint and marginal distributions.

Hypothesis testing. Statistical hypothesis tests including χ^2 goodness-of-fit test and alternatives, and Pearson product moment, Spearman rank and Kendall τ rank correlation coefficients.

Results analysis & discussion. Multivariate statistical analysis seems less appropriate for exploring inconsistencies in count frequencies of campaigns and more appropriate for other dimensions or combinations of these, since the resulting distributions are smaller and reveal clearer trends. Comparison of the observed and expected distributions provides a clear overview of the inconsistencies

in the count frequencies, which is best detected by different correlation coefficients, while the statistical goodness-of-fit tests are not discriminatory here. Nevertheless, due to an unsupervised nature of the adopted methods, substantial human endeavor is required to reveal any interesting conclusions.

Final assessment. Multivariate statistical analysis provides an alternative model for the detection of general ecosystem issues and campaign performance issues to some extend, while its use in practice might be questionable or should be supplemented with other methods.

PoC. PoC implementation is provided within CLUSTERS repository as Java code that collects data for the specified period from MySQL database and outputs the results of multivariate statistical analysis over all metrics and dimensions as a series of documents including vector figures (see Figure 4).

Estimated effort. The overall effort is estimated to $10x$, where x is a certain time period.

NETWORK & LINK ANALYSES

Problem definition. Network and link analyses aim to address *campaign performance issues and general ecosystem issues* over the specified period in time. Particularly, the goal is to detect substantial inconsistencies in different count metrics of particular campaigns, platforms, SDK, formats, devices and combinations of these.

Data representation. The data is represented as a set of campaign, platform, SDK, format and device networks for each of the adopted count metrics. These include one session and one video count metric. Each set of networks includes correlation networks based on Pearson correlation of different count profiles, bipartite affiliation networks based on nonzero counts in the profiles and collaboration networks defined as the projections of the latter.

Methods analysis & selection. Analysis of the resulting networks shows that these are particularly dense and have to be reduced to a common density of real networks. Affiliation and collaboration networks reveal clear group structure that can be detected by community detection algorithms. Substantial inconsistencies in count profiles of particular campaigns, platforms, SDK, formats and devices can be determined by exploring the resulting group structure or by referring to link analysis methods for detection of influential nodes and prediction of missing links.

Graph representation. Unipartite correlation and collaboration, and bipartite affiliation networks.

Community detection. Greedy optimization of modularity, hierarchical optimization of modularity or Louvain method and structural compression algorithm or Infomap.

Network analysis. Network and group sizes, components, mixing, clustering, conductance, modularity.

Link analysis. Node influence by PageRank algorithm and link prediction by resource allocation index.

Results analysis & discussion. Network and link analyses seem more appropriate for exploring inconsistencies in count profiles of campaigns and less appropriate for other dimensions or combinations of these, since the resulting networks are rather small and reveal no clear structure. Community detection algorithms provide a rough overview of the inconsistencies in the count profiles, which is best detected by the Louvain method, while the exploration can be somewhat guided by link analysis methods. Nevertheless, due to an unsupervised nature of the adopted methods, substantial human endeavor is required to reveal any interesting conclusions.

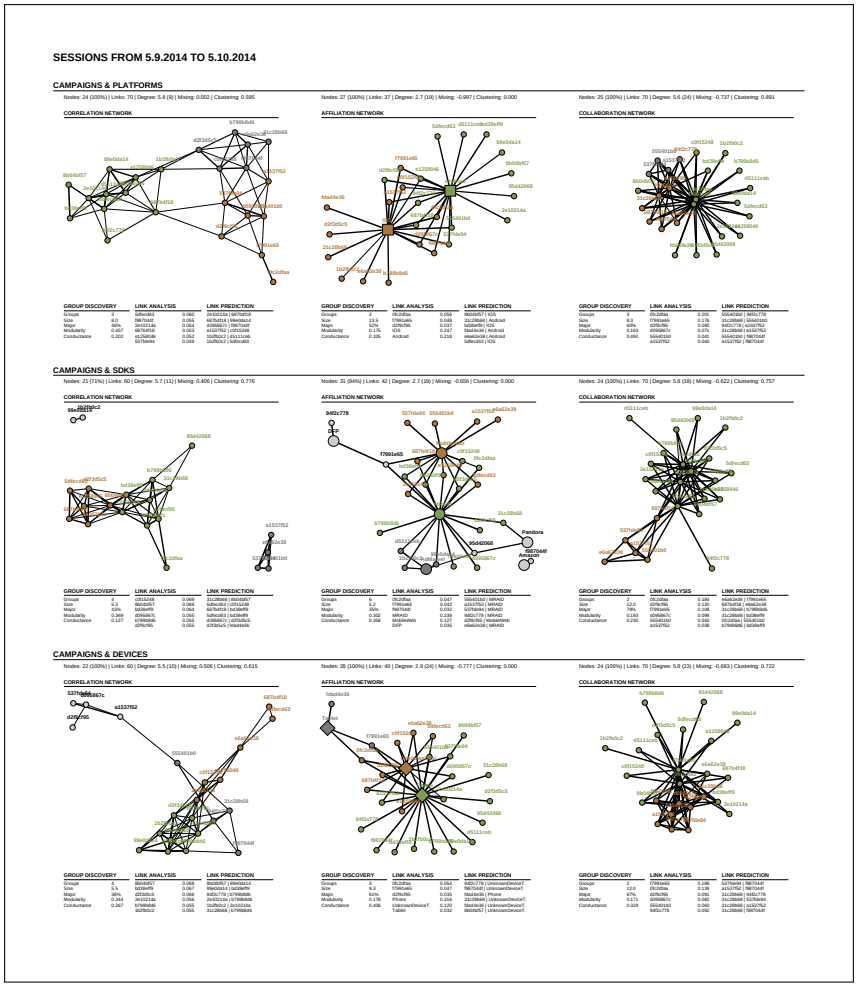


Figure 5: Campaign networks for sessions count on October 5, 2014.

Final assessment. Network and link analyses provide an alternative model for the detection of campaign performance issues and general ecosystem issues to some extend, while its use in practice is questionable and should be supplemented with other methods.

PoC. PoC implementation is provided within CLUSTERS repository as Java and C++ code that collects data for the specified period from MySQL database and outputs the results of network and link analyses over all metrics and dimensions as a series of documents including vector figures and reports (see Figure 5).

Estimated effort. The overall effort is estimated to $15x$, where x is a certain time period.

DATA CLUSTERING

Problem definition. Data clustering aims to address *account, campaign and creative performance issues, and general ecosystem issues*, over specified period in time. Particularly, the goal is to detect substantial inconsistencies in different count and rate metrics of particular accounts, campaigns, creatives, platforms, SDK, formats, devices and combinations of these.

Data representation. The data is represented as a set of account, campaign, creative, platform, SDK, format and device profile tables with all adopted metrics. These include one session and one

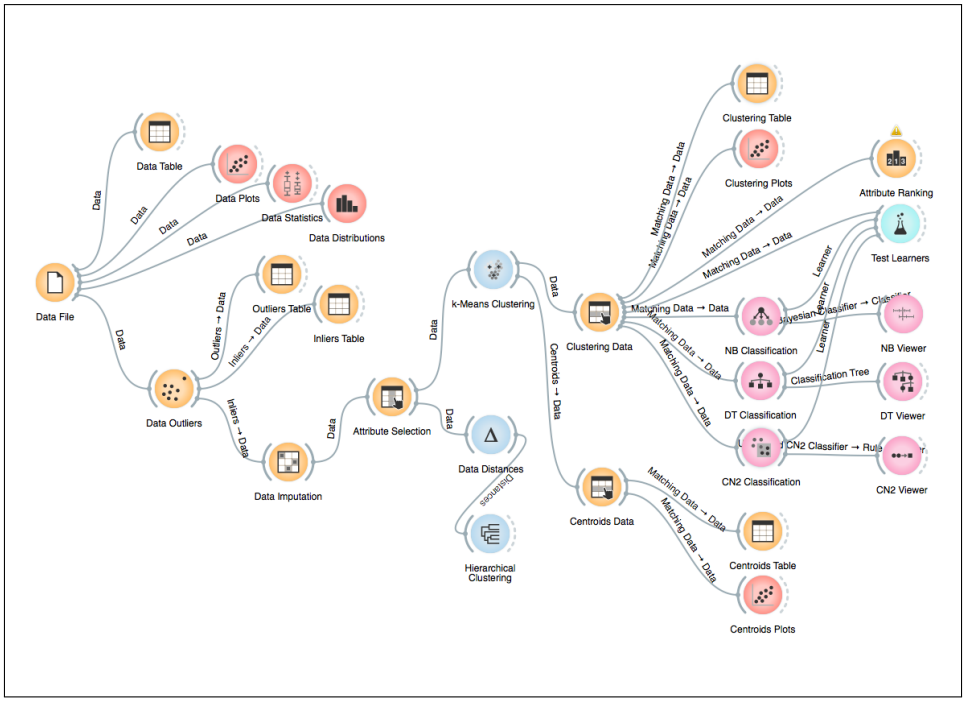


Figure 6: Orange data mining workflow for data clustering.

video count metric, and four session and four video rate metrics.

Methods analysis & selection. Analysis of the resulting profile tables shows that data consists of some clear clusters, while some parts reveal no clear cluster structure. Nevertheless, in the case of the former, clusters can be detected by standard clustering algorithms. Substantial inconsistencies in count or rate profiles of particular accounts, campaigns, creatives, platforms, SDK, formats and devices can be determined by exploring the resulting cluster structure or by referring to data mining methods for detection of relevant attributes that provide explanatory visualizations.

Data preprocessing. Missing data imputation, outlier detection and attribute selection.

Data clustering. k-means partitional and agglomerative hierarchical clustering algorithms.

Data mining. Statistics and distributions, scatter plots, attribute ranking and learning.

Results analysis & discussion. Data clustering seem less appropriate for exploring inconsistencies in count and rate profiles of accounts, campaigns and creatives, and more appropriate for other dimensions, since the resulting profile tables are much smaller and less prone to poor robustness of the algorithms. Session metrics should not be mixed with video metrics, which include many missing values that have to be imputed properly. Data clustering provide a rough overview of the inconsistencies in the count and rate profiles, which is best detected by the k-means algorithm. Nevertheless, due to an unsupervised nature of the adopted methods, substantial human endeavor is required to reveal any interesting conclusions.

Final assessment. Data clustering provides an alternative model for the detection of account, campaign and creative performance issues, and general ecosystem issues, while its use in practice is questionable and should be supplemented with other methods.

PoC. PoC implementation is provided within CLUSTERS repository as Java code that collects data for the specified period from MySQL database and outputs it as a series of TAB files. These are then analyzed within Orange using the provided data mining workflow (see Figure 6).

Estimated effort. The overall effort is estimated to $5x$, where x is a certain time period.

— REINFORCEMENT LEARNING

Problem definition. Reinforcement learning aims to address *dynamical optimization issues*.

Literature analysis & discussion. Reinforcement learning can efficiently solve the exploration-exploitation trade-off or multi-armed bandit problem in dynamical ad optimization. Particularly, probability matching algorithms like Thomson sampling provide an effective method that can also be supplemented with an arbitrary context through logistic regression learning. The logistic model can be efficiently trained using stochastic gradient descend optimization.

Estimated effort. The overall effort is estimated to x, where x is a certain time period.