Faculty of
Computer Science

Big Data and Information
Retrieval School

Moscow
2023

Analysis of the Rationality of Usage of Physical Server Resources

Time-Series Data Preprocessing

Group software project by
Konstantin Shashkov, DSBA 201
Supervised by Dmitry Frolov,
Senior System Analyst at "NSPK" JSC

Subject Area: Server Load Analysis

As the number of servers grows, the costs of maintenance and horizontal scaling become notable.

For an IT business to be profitable, it is crucial to use the servers rationally. However, in order not to affect the company's performance, one should take into account the peculiarities of server usage in software development and the prospective growth of demand for computational power.

Our project aims to develop a tool for "NSPK" JSC to analyse how rationally their test and development servers are utilised.

My personal goal is to prepare the previously collected time-series data for further analysis given the specifics of its structure.

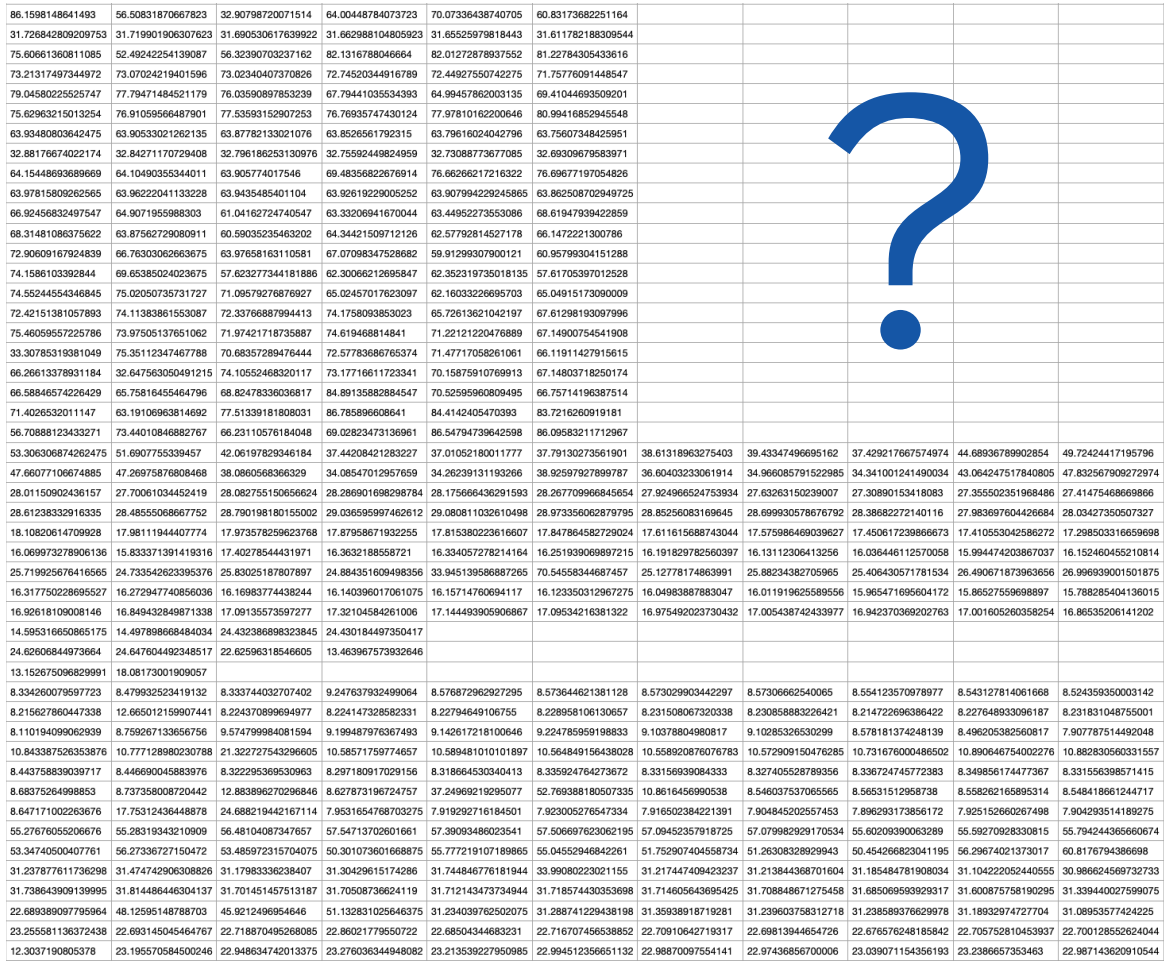
"Облачные Вычисления." *PNGWING*, https://www.pngwing.com/ru/free-png-ziagq. Accessed 15 June 2023.

Relevance of my section for the project

The data of server load is collected as a series of records parametrised on the metric of interest, the considered server, and the moment in time.

Although Prometheus is a highly-automate tool collecting metrics from computers, it is configured and maintained by people. In the company, there are many teams of developers, each administrating their own servers. As a result, not only is the resultant data likely to be incomplete and nonuniform, it also may happen to be inadequate at times. The longest period, for which we have complete data on every server, is 1 week only.

Meanwhile, the quality of data is crucial for the analysis to be informative. Large blocks of missing data present an even greater difficulty for reconstruction and further processing.



Key Terms & Definitions

Time-Series Data
Data, which is parametrised on time.

Missing Value
A data point, whose value is unknown.

Inadequate value
A data point, whose value is unrealistic, i.e. falls outside of its defined range.

Prometheus
A popular open-source time-series database and monitoring tool. It is designed to collect, store, and query metrics from various sources, such as cloud-based infrastructure, containers, and microservices.

Imputation
The process of estimating the missing values.

Planning the Project

Requirements for the Product:

- The application is expected to remove inadequate values from the data.
- The application is expected to impute the missing values.
- The application is expected to be horizontally scalable.
- The application is expected to be universally employable given similar infrastructure (availability of similar data collection, storage, processing and visualisation tools).

Tasks to Accomplish:

- Compare and contrast the existing solutions for missing data imputation.
- Propose and implement a data-specific approach to impute the missing values.
- Compare the resultant product to the existing analogues.
- Choose and implement a technique to deal with inadequate data.

"Клипарт С Буфером Обмена Черно-Белый." *FlyClipart*, https://flyclipart.com/ru/clipboard-clipart-black-and-white-clipboard-clipart-black-and-white-644986. Accessed 15 June 2023.

Existing Approaches to Estimation of Missing Values

Standard Statistical Estimations

- Sample Mean
- Sample Mode
- Sample Median
- Simulations from *Normal(sample mean, sample variance)*

sklearn.impute.IterativeImputer

- Iteratively regresses every feature with missing values on the rest of the dataset.
- May improve its performance by re-imputing the missing values employing the estimations from previous iterations.
- Allows to choose a desired estimator for the regression.
- Uses *BayesianRidge* regression by default.

sklearn.impute.KNNImputer

- Based on the k-Nearest Neighbours clustering algorithm with *k* as a parameter.
- Estimates every missing values as an average or distance-weighted average of *k* nearest neighbouring features for the given observation.
- Allows to specify a desired distance metric.
- Uses *nan_euclidean_distances* as a default distance metric.

Proposed Solution

Based on Autoregression

Approach Description

1. Consider the data for every metric (RAM, CPU, DISK) separately.
2. Using the OLS estimator, regress a value from server *X* on day *t* on 7 more recent observations from the same server to train the AR model:

$$X_t = \sum_{i=1}^{7} a_i X_{t+i} + \epsilon_t$$

3. Use the obtained model to iteratively estimate every missing value as a function of 7 more recent observations from the same server:

```
for server in server_list:
  for day_offset in range(7, 30):
     if metric[server, day_offset] is null:
        metric[server, day_offset] = AR(server, day_offset)
```

Implementation Details

Out of the available complete data, a train and a test sets were selected.

The training set was restructured for convenience. For every server *X*, the following series of observations was constructed:

7	6	5	4	3	2	1	Target
X[0]	X[-1]	X[-2]	X[-3]	X[-4]	X[-5]	X[-6]	X[-7]
X[-1]	X[-2]	X[-3]	X[-4]	X[-5]	X[-6]	X[-7]	X[-8]
…	…	…	…	…	…	…	…
X[-22]	X[-23]	X[-24]	X[-25]	X[-26]	X[-27]	X[-28]	X[-29]

In the testing set, some observations were synthetically omitted following the patterns present in the original missing data.

Dealing with **Inadequate Data**

- By definition, all 3 metrics are measured in % of used resource, so the values should lie in between 0 and 100.
- High-magnitude negative values were detected in the dataset.
- The numbers did not seem interpretable on their own. Neighbouring observations appeared to be more informative about the presumable true values of these cells.
- I decided to apply the proposed imputing solution to estimate the needed observations from the context.

95.32700823369376	91.70710874717702	87.90947325173649
27.831258209146448	26.741558308528035	25.66300596097578
99.99994981561264	-604663245697.3097	-1400215500337.04
46.0910485147207	47.09559371922345	44.27578293065546
57.55613469207649	57.43818233650124	56.74330556953684
42.23475661028133	42.15057606959907	42.072366066015775
18.746586939697774	18.77027707179655	18.72999160052643
23.88964272481483	23.83838848323141	23.811468107930633
31.774083927149288	31.698838610479196	31.492459393042317
18.888066398527105	18.85885783182737	19.04598840945041
93.52228747573642	90.3747043259586	87.37909028250237

Testing & Comparison of Performance

The testing set was given for reconstruction to the proposed AR imputer and 2 ML-based algorithms from *sklearn.impute* with default parameters. The comparative results of these 3 models are presented in the table below.

	Proposed AR model	**IterativeImputer**	**KNNImputer**
Root Mean Squared Error	6,60117	9,99687	5,71086
Mean Absolute Error	3,60792	6,85515	2,70549
Mean Absolute Percentage Error	0,46682	1,37328	0,48849

Conclusion

& Prospects for Future Work

The proposed solution has outperformed *sklearn*'s *IterativeImputer* and was very close to *KNNImputer* in the context of the project's structure of data. It was later successfully used to impute the missing values in the original dataset.

Nevertheless, a downward trend was observed in the model's estimates over long periods of time. Presumably, accuracy drops when the imputer gets its own estimates as input arguments. One could experiment with the model's hyperparameters to improve its performance.

Alternatively, given how well *KNNImputer* has performed, one may try to achieve even better results by adapting it to the data, e.g. developing a custom time-based distance metric for the features.

	Proposed AR model	**IterativeImputer**	**KNNImputer**
Root Mean Squared Error	6,60117	9,99687	5,71086
Mean Absolute Error	3,60792	6,85515	2,70549
Mean Absolute Percentage Error	0,46682	1,37328	0,48849

References

1. "6.4. Imputation of Missing Values." Scikit Learn, 2023, scikit-learn.org/stable/modules/impute.html#impute. Accessed 9 June 2023.

2. Buuren, Stef Van, and Karin Groothuis-Oudshoorn. "Mice: Multivariate Imputation by Chained Equations inr." Journal of Statistical Software, vol. 45, no. 3, 2011, https://doi.org/10.18637/jss.v045.i03.

3. "Sklearn.Impute.IterativeImputer." Scikit, scikit-learn.org/stable/modules/generated/sklearn.impute.IterativeImputer.html#sklearn.impute.IterativeImputer. Accessed 9 June 2023.

4. "Sklearn.Linear_model.Bayesianridge." Scikit, scikit-learn.org/stable/modules/generated/sklearn.linear_model.BayesianRidge.html#sklearn.linear_model.BayesianRidge. Accessed 9 June 2023.

5. "Sklearn.Impute.KNNImputer." Scikit, scikit-learn.org/stable/modules/generated/sklearn.impute.KNNImputer.html#sklearn.impute.KNNImputer. Accessed 9 June 2023.

6. "Sklearn.Metrics.Pairwise.Nan_euclidean_distances." Scikit, scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.nan_euclidean_distances.html. Accessed 9 June 2023.

7. "The Elements of Statistical Learning: Data Mining, Inference, and Prediction" - Trevor Hastie, Robert Tibshirani, Jerome Friedman, p. 206 https://hastie.su.domains/ElemStatLearn/printings/ESLII_print12.pdf

8. "The Elements of Statistical Learning: Data Mining, Inference, and Prediction" - Trevor Hastie, Robert Tibshirani, Jerome Friedman, p. 11 – 18 https://hastie.su.domains/ElemStatLearn/printings/ESLII_print12.pdf

9. "Statsmodels.Regression.Linear_model.Ols." Statsmodels.Regression.Linear_model.OLS - Statsmodels 0.15.0 (+14), www.statsmodels.org/dev/generated/statsmodels.regression.linear_model.OLS.html. Accessed 9 June 2023.

10. "SKLEARN.METRICS.MEAN_SQUARED_ERROR." Scikit, scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_squared_error.html#sklearn.metrics.mean_squared_error. Accessed 9 June 2023.

11. "SKLEARN.METRICS.MEAN_ABSOLUTE_ERROR." Scikit, scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_absolute_error.html#sklearn.metrics.mean_absolute_error. Accessed 9 June 2023.

12. "SKLEARN.METRICS.MEAN_ABSOLUTE_PERCENTAGE_ERROR." Scikit, scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_absolute_percentage_error.html#sklearn.metrics.mean_absolute_percentage_error. Accessed 9 June 2023.

Faculty of
Computer Science

Big Data and Information
Retrieval School

Moscow
2023

Thank you for your attention

"IBM Virtual Server." *Mavink*, 2019, https://1.cms.s81c.com/sites/default/files/2019-12-30/LS003.jpg. Accessed 15 June 2023.