Statistical & Causal Analysis of Chimbuko Provenance Database

Margaret Ajuwon
Industrial & Systems Engineering
Morgan State University
Baltimore MD, USA
maaju1@morgan.edu

Serges Love Teutu Talla
Department of Mathematics
Morgan State University
Baltimore MD, USA
seteu1@morgan.edu

Isabelle Kemjou-Brown
Department of Mathematics
Morgan State University
Baltimore MD, USA
elisabeth.brown@morgan.edu

Christopher Kelly
Computational Science Initiative
Brookhaven National Laboratory
Upton NY, USA
ckelly@bnl.gov

Kerstin Kleese Van Dam
Computational Science Initiative
Brookhaven National Laboratory
Upton NY, USA
kleese@bnl.gov

***Abstract*— Performance data are collected to establish how well Exascale applications are doing with executing their code or workflow as efficiently as possible. Chimbuko, a tool specifically focused on the analysis of performance data in real time, looks through these data and collects performance anomalies that are being detected. These anomalies are saved into the Chimbuko Provenance Database, together with as much contextual information as needed. The goal of our work is to perform statistical analysis on the Chimbuko Provenance Database by presenting simple visualizations and determining if the information collected for each anomaly is sufficient to conduct a causal analysis. Statistical methods such as Theil's U correlation analysis, logistic regression, and K-Prototype clustering were used to identify association between variables. Furthermore, feature selection was conducted with Decision Tree and Random Forest. We identified association between call_stack and several variables, which reveals that call_stack is a very important feature of the dataset.**

Keywords—Chimbuko, Feature Selection, Regression, Correlation, Clustering, Causal Analysis

I. INTRODUCTION

CODAR (Center for Online Data Analysis and Reduction) is a project focused on addressing needs for data reduction, analysis, and management in the exascale era. Chimbuko was developed within CODAR, and it is specifically focused on the analysis of performance data in real time. Performance data are collected to establish how well exascale applications are doing with executing their code or workflow as efficiently as possible. Chimbuko looks through the performance data and collects performance anomalies that are being detected. These anomalies are saved into the Chimbuko provenance database, together with as much contextual information as needed.

The data in the provenance database can be used in two distinct ways - visualization of the application performance in real time, so that the user can investigate anomalies as they are happening; and an archive of performance anomalies that can be studied later, including performance information for many different application/workflow executions. While the CODAR project has made much progress on enhancing its Chimbuko tools real time analysis capabilities, it has not been able to harness the information in its provenance databases. In this work, we performed an offline analysis of trace data obtained from a 512-rank run of the XGC/WDMapp fusion reactor simulation on the Crusher supercomputer.

II. CHIMBUKO PROVENANCE DATATBASE SCHEMA

As discussed earlier, Chimbuko collects performance anomalies that are being detected and saves it together with as many contextual information as needed. The data was unstructured with a total of 26 variables. The explanation and details of each is further explained in [1]. Of the 26 variables, we discovered that not all variables were relevant for our analysis, hence we dropped some of the variables such as: __id, algo-param, is_anomaly, gpu_location, gpu_parent, pid, host_name, runtime-exclusive and io_step. The variables which are used for the analysis are explained below.

Variables	Explanation
call_stack	*Function execution call stack (starting with anomalous function)*
entry	*Timestamp of function execution entry*
exit	*Timestamp of function execution exit,*
func	*Function name*
fid	*Global function index (can be used as a key instead of function name)*

is_gpu_event	*True or false depending on whether function executed on a GPU*
rid	*Process rank*
tid	*Thread index*
io_step_tstart	*Time of start of IO step*
io_step_tend	*Time of end of IO step,*
outlier_score	*The anomaly score of the execution reflecting how unlikely the event is (algorithm dependent, larger is more anomalous)*
outlier_severity	*The severity of the anomaly, reflecting how important the anomaly is*

III. Goals & Objectives

This project was designed to develop a set of initial data analytics for the provenance databases to answer a set of core questions:

- Is the contextual information collected for each anomaly sufficient to conduct causal analysis to identify the underlying reason for the anomaly?
- Have the applications/workflows improved the utilization of the systems over time? Where have they specifically improved?
- Can we discover interesting trends of performance variability (differences in performance statistics), when the same workflow is executed several times on the same system, with the same set of parameters? Can we find indicators for what may cause this variability?

To answer the above questions, we will develop an initial set of analytical algorithms to query the performance data using the existing Python interface. Also, simple visualization of data analysis results will be provided, using Jupyter notebook.

IV. METHODOLOGY

The data for this analysis was obtained offline from a 512-rank run of the XGC/WDMapp fusion reactor simulation on the Crusher supercomputer. This data was accessed in the Chimbuko Provenance Database using Jupyter Python interface and structured into numerical and categorical variables. Statistical analysis such as the Theil's U correlation, Machine Learning Techniques; Regression for prediction, K-Prototypes for clustering, Random Forest Model and Decision Tree for feature selection were all performed on the data set.

A. Theil's U Correlation

Based on the selected variables we performed correlation using the Theil's U correlation. The Theil's U, also known as the uncertainty coefficient, is based on the conditional entropy between x and y, in other words, given the value of x, how many possible states does y have and how often do they occur. This correlation analysis was performed to the entire dataset to identify association between variables.

B. K-Prototype Clustering

The k-prototype algorithm was introduced for clustering mixed-type (categorical and numerical) data. The algorithm combines the ideas of k-means algorithm and k-modes algorithm by dividing the dataset into k ($k \in N+$) different subclusters to minimize the value of the cost function. The cluster analysis was performed to check for clusters among variables and identification of trends**.**

C. Feature Selection

Feature selection was performed on selected variables (entry, is_gpu_event, outlier_severity, rid, runtime_total, and tid) using Random Forest and Decision Tree. The aim is to select only the important features for prediction

D. Regression Analysis

Regression Analysis was performed to examine the relationship between variables and outlier_score. The analysis was executed using Binary Logistic Regression. In this case, the outlier_score was chosen as the outcome while the predictors were entry, is_gpu_event, outlier_severity, rid, runtime_total, and tid.. In order to use binary logistic regression, outlier_score was binned into 2 groups of 0 and 1.

V. Results and Discussion.

- The correlation coefficient of call_stack is above 0.93 with all features except for rid (rank index), and outlier_score with the values of 0.49 and 0.61 respectively. This reveals that call_stack is a very important feature of the dataset and could highlight interesting trends.
- Regression analysis performed before feature selection indicates that there is strong evidence of association between the outcome (outlier_score) and predictors: is_gpu_event, tid, runtime_total and outlier_severity.
- Feature selection was performed using Random Forest and Decision Tree and, showed that outlier_severity and entry are the only important features. The regression analysis results indicate that there is strong evidence of association between the outlier_score and the outlier_severity.

Acknowledgment

The authors thank DOE, ECP, Sustainable Horizons Institute, and Brookhaven National Lab for their support

References

[1] Provenance Database Schema - Performance Analysis 3.0.0 documentation. (n.d.). Retrieved August 12, 2022, from https://chimbuko-performance-analysis.readthedocs.io/en/ckelly_develop/io_schema/schema.html