Improving Data Center Visibility with Machine Learning

Analiese Gonzalez [1] Ben Matthews [2] Jenett Tillotson [2]

[1]Cypress College [2]NSF National Center for Atmospheric Research

Abstract

NSF NCAR's supercomputers, Casper and Derecho, are powerful machines performing extensive calculations for researchers, often involving large datasets. These systems house high-performance nodes essential for complex computational tasks. Despite their reliability, node failures occasionally occur. This project leverages AI and machine learning techniques to attempt to predict when a node may fail, aiming to reduce troubleshooting time and prevent future issues. By using advanced methods such as neural networks, we seek to anticipate node failures. Neural networks are particularly useful because they can improve decision making processes. Casper and Derecho provide real-time data center metrics stored in a timescale database, encompassing 20 tables of time-series data. This data includes metrics about Cheyenne, Casper, Gust, and Derecho. The student analyzed various variables, including CPU usage and memory usage, using a K-Means clustering model developed in JupyterHub with Python to explore patterns in the data. After examining the clusters, the student utilized neural networks with sliding windows to predict node unavailability. The models revealed differences and similarities between the two machines, such as how the time a node changes state affects node failure equally for both, but CPU and memory usage impacts Derecho more than Casper.

Background and Goals

What **insight** can machine learning provide about our supercomputers?

Is there a way to effectively **predict node failures** using machine learning?

- **Derecho** is NSF National Center for Atmospheric Research's (NCAR) newest supercomputer home to 2570 nodes with 3 cooling distribution units (CDUs) to prevent overheating. The CDUs use 650-800 gallons of chilled water per minute [1].
- **Casper** is an older supercomputer owned by NSF NCAR and home to 121 nodes. Casper is mainly used for data processing [1].
- **Nodes** are individual computers with each having its own "brain", random access memory (RAM), and storage space. A group of nodes is called a cluster.
- **Portable Batch System Professional (PBS Pro)** is a workload management and job scheduling system designed to optimize the use of the supercomputers. It helps to efficiently schedule and manage jobs to maximize efficiency and minimize downtime.

Data and Variables

The data consists of 20 tables of time series data where it is collected into a timescale database. The database involves information about Casper, Derecho and other machines. Some tables include as many as 142 columns and millions of rows. To access the data, the command line and DBeaver, which is a postgresql management software, were used to organize and query the data with SQL.

Figure 1. Left: DBeaver logo Right: SQL logo

The data used in this project includes:

- **Outlet Primary Pressure**: The measure of the pressure of the fluid when it exits the CDU and enters the primary cooling loop
- **Inlet Temperature**: Temperature of the liquid as it enters the system
- **Outlet Temperature**: Temperature of the liquid as it exits the system
- **Actuator Percentage**: Percentage the actuator is opened to let water flow
- **Filter Secondary Pressure**: Measure of the pressure of the cooling fluid
- **Last State Change Time**: The time a node changed state last
- **Resources Available Memory**: The total amount of memory available for use on a node
- **Resources Available nCPUs**: Number of CPUs available on a node
- **Last Used Time**: The amount of time passed since a node was last used
- **Resources Available Virtual Memory**: Amount of virtual memory available on a node
- **Resources Assigned Memory**: The amount of memory assigned to a job
- **Resources Assigned nCPUs**: Amount of CPUs assigned for a job
- **Resources Available nGPUs**: Amount of graphics processing units (GPUs) available

Modeling - K-Means Graphs

K-Means clustering is an unsupervised machine learning algorithm that groups data into clusters based on similarities. Each point is assigned to the nearest cluster using the Euclidean distance formula. An elbow graph was used to determine the optimal number of clusters. Using Python in Jupyterhub, the following K-Means graphs were generated.

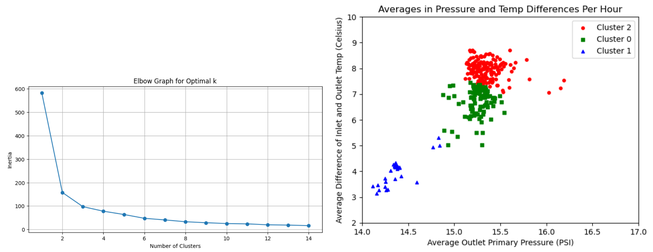

Figure 2. Left: First elbow graph to find optimal number of clusters. Right: Clusters of hourly averages in pressure and temp differences for Derecho CDUs.
Rack x1100 was found to be reporting 0 consistently for its data which was caused by an undiscovered broken sensor.

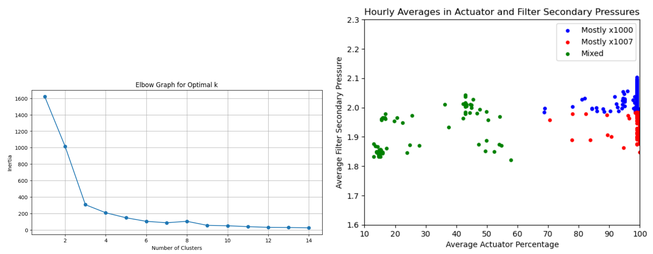

Figure 3. Left: Second elbow graph to find optimal number of clusters. Right: Clusters of hourly averages in actuator and filter pressures from the Derecho CDUs excluding rack x1100.
One cluster was mainly rack x1000 while another was x1007. This graph revealed one of the CDUs was **misconfigured** which was then fixed by an employee at NCAR Wyoming Supercomputing Center (NWSC).

Correlation Charts

Casper

Feature	Importance
last_state_change_time	20.50%
resources_available.mem	19.18%
resources_available.ncpus	18.73%
last_used_time	16.28%
resources_available.vmem	8.72%
resources_assigned.mem	7.60%
resources_assigned.ncpus	5.43%
resources_available.ngpus	3.56%

Derecho

Feature	Importance
last_state_change_time	43.23%
last_used_time	30.23%
resources_assigned.ncpus	9.62%
resources_assigned_mem	9.11%
resources_avail_mem	2.76%
resources_avaiable_vmem	2.43%
resources_available.ncpus	1.65%
resources_available.ngpus	0.97%

Figure 4. Left: Casper PBS data correlation chart to node failure. Right: Derecho PBS data correlation chart to node failure.

For both supercomputers, the last state change time is the most important factor in node failures. This suggests nodes may return to "available" too quickly after an issue is found. In Casper, assigned CPUs and memory have less impact on node failures compared to Derecho.

Modeling - Neural Networks

- Training set was 70% of original data
- Testing set was 15% of original data
- Validation set was 15% of original data
- Sliding windows were integrated to capture temporal dependencies
- Model has a total of 7 layers (5 hidden)
- Model was trained over 50 times

Neural Network Results

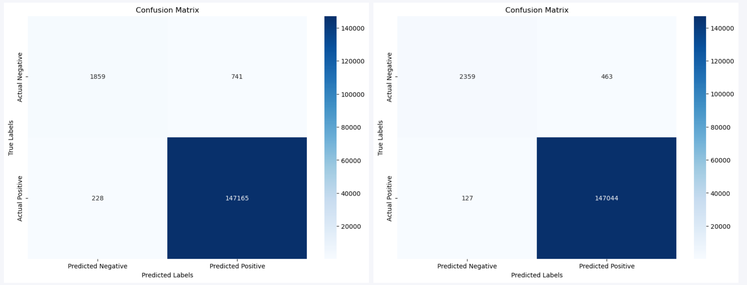

Figure 5. Left: Confusion matrix of Casper PBS data. Right: Confusion Matrix of Derecho PBS data.

- The confusion matrices show how often the models predicted 1 (node available) and 0 (node unavailable) for the output. The matrix contains four quadrants: true positives, true negatives, false positives, false negatives.
- For Casper, the accuracy of my model is 99.3% and Derecho was 99.6% which is calculated by the formula: $\frac{TP+TN}{TP+TN+FP+FN}$

Parameter	Value
Window size	50.0
Batch size	16.0
Epochs	50.0
Mean Absolute Error	0.006460301480735768
Mean Squared Error	0.006460301480735768
Root Mean Squared Error	0.08037600065153633
R-squared	0.6207334316575936

Parameter	Value
Window size	50.0
Batch size	16.0
Epochs	50.0
Mean Absolute Error	0.003933516897455215
Mean Squared Error	0.003933516897455215
Root Mean Squared Error	0.06271775583879907
R-squared	0.7869194775815369

Figure 6. Top: Neural network metrics of Casper PBS data. Bottom: Neural network metrics of Derecho PBS data.

- Derecho had a higher R-squared value of 80% while Casper had a value of 62%.
- Even while using the same variables, the machines generated different results
- PBS data explains the variability in node failure for Derecho than Casper

Future Work

- Use more correlation charts and heat maps
- Group nodes by specific type
- Create my own database to be able to index the data which would help to easily combine data between different tables
- Look into different variables such as CPU data

Acknowledgments

I would like to express thanks to my mentors, Ben Matthews and Jenett Tillotson for their support this summer. I would also like to thank Virginia Do, Jerry Cyccone, Jessica Wang, Eva Sosoo, Ben Fellman, and the rest of the SiParCS team. I also would like to express a special gratitude for the funding by the National Science Foundation under Grant No. ICER-2019758.

References

[1] CISL HPC Allocations Panel. Ncar hpc documentation, 2023.

https://www.cisl.ucar.edu/outreach/internships/2024projects