13th August 2024

Anomaly Detection in Grid Compute Nodes: A Machine Learning Approach Leveraging HEP Benchmark Suite

CERN openlab

AGH

AGH University of Krakow

Author:

Kacper Kamil Kozik

AGH University of Krakow

Supervisors:

Natalia Diana Szczepanek

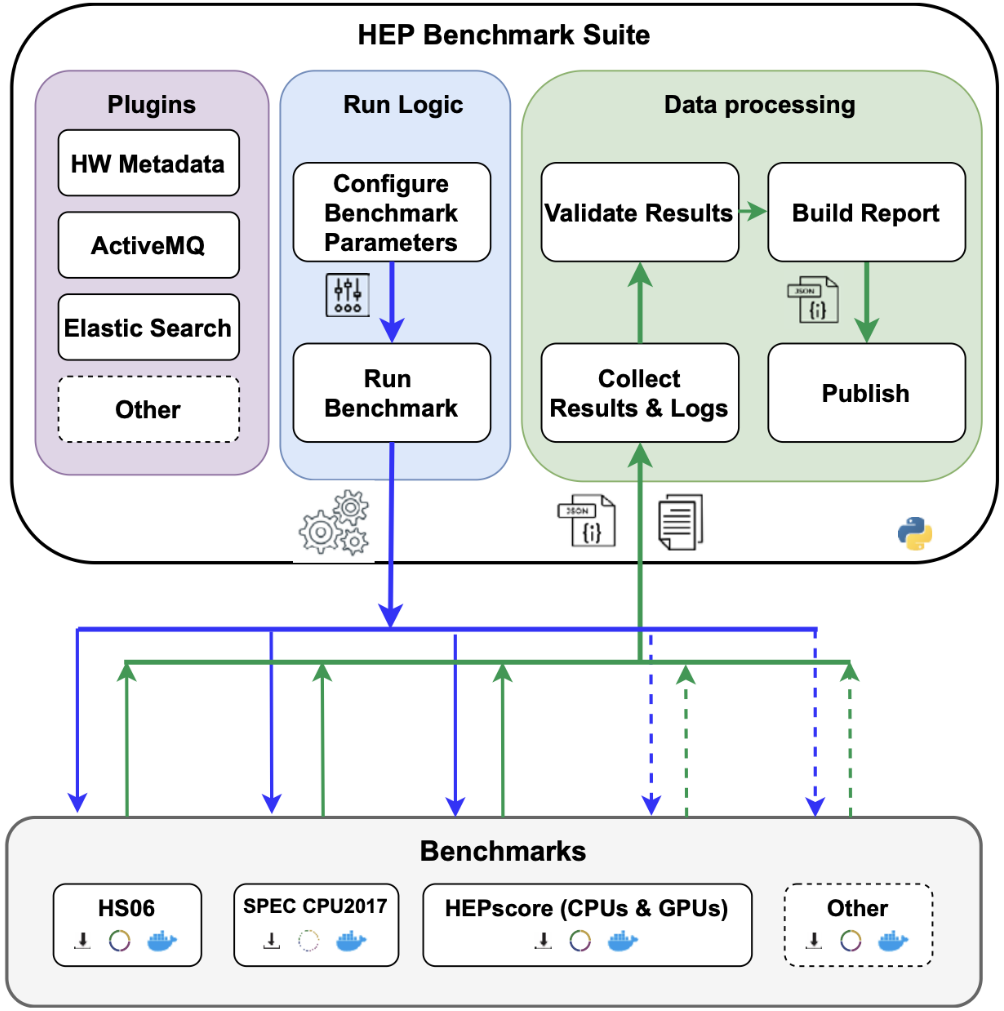
Ewoud Ketele

What is Benchmarking ?

- **Benchmark Scores:** Compare performance of systems or system components (e.g. smartphones, CPU models)
- **Purpose:** Identify the best value based on your specific needs and tasks
- **In Computing:** Measure system performance using specific predefined tests or workloads
- **Outcome:** Determine the most efficient option for your requirements

CPU Benchmarking on the Worldwide LHC Computing Grid

➢ HEP Benchmark Suite:



WLCG
Worldwide LHC Computing Grid

➢ Purpose:

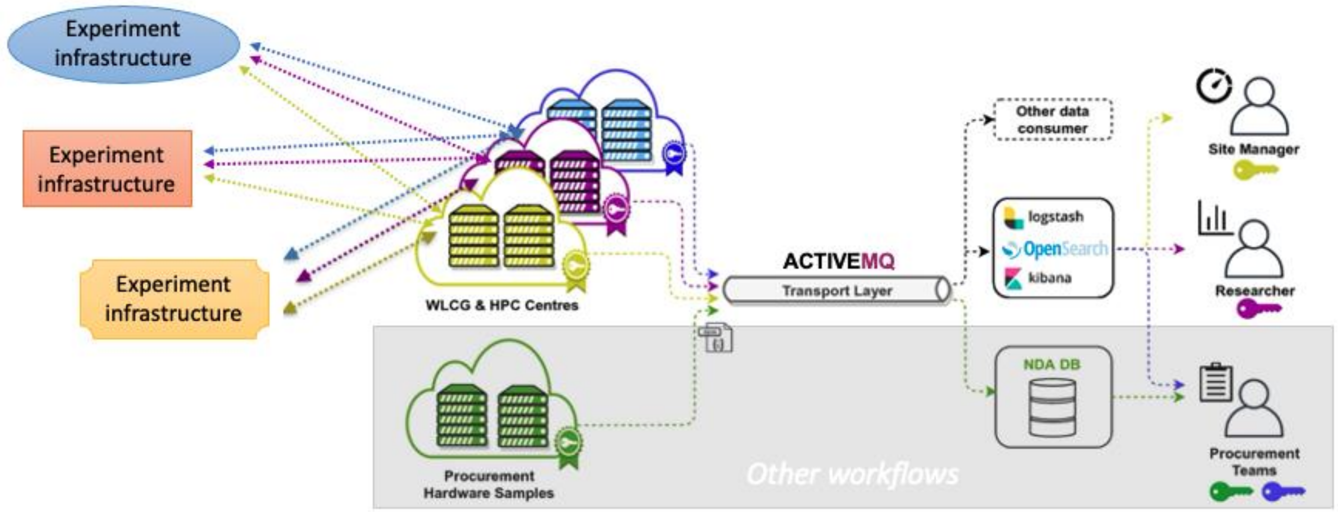
- Compare different CPU models (for accounting and financial planning)

➢ HEPScore23:

- Includes 7 workloads from 5 experiments: **ATLAS, CMS, ALICE, Belle II, LHCb**

CPU Benchmarking on the Worldwide LHC Computing Grid

- The **HEP Benchmark Suite** is submitted as a standard job to the grid
  - Probing the performance of the grid servers in production environment
  - Results, together with metadata such as load, memory usage and power consumption are sent back to us
- The collected data can also be used to detect misconfigured servers
  - Correlation between **Load** of the server and the **HEPScore** (performance of the server) can be used to detect those misconfigurations (anomalies)
  - This process is done manually right now
  - The goal of the project is to automate it



Statistics:

Over 200k jobs finished:

- 139 unique sites
- 227 unique CPU Models
- 28246 unique hosts

Not All Anomalies Are the Same

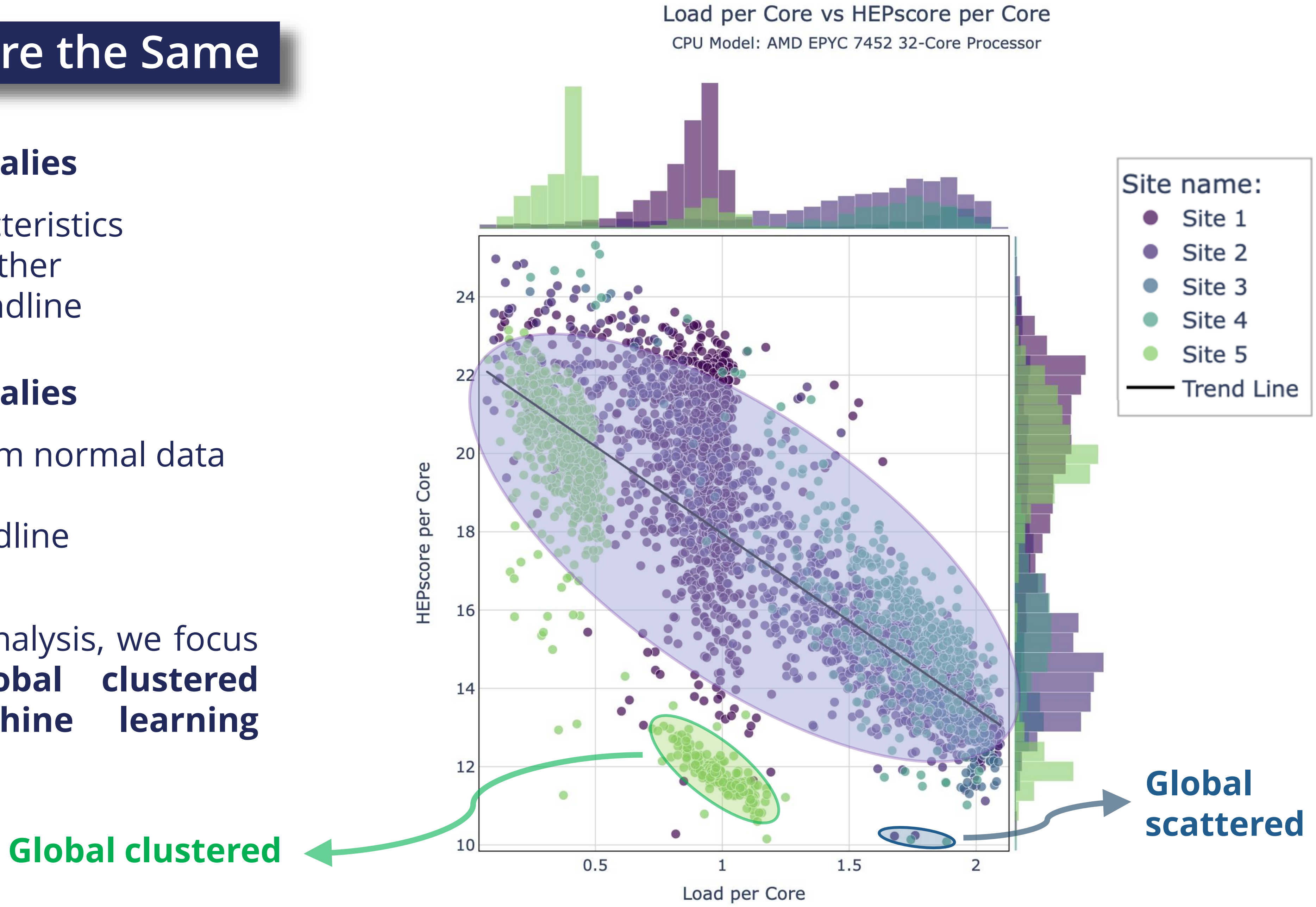
➢ **Global Clustered Anomalies**

- Exhibit similar characteristics
- Grouped closed together
- Located far from trendline

➢ **Global Scattered Anomalies**

- Differ significantly from normal data
- Spread far apart
- Located far from trendline

For the purposes of this analysis, we focus only on identifying **global clustered anomalies using machine learning techniques**

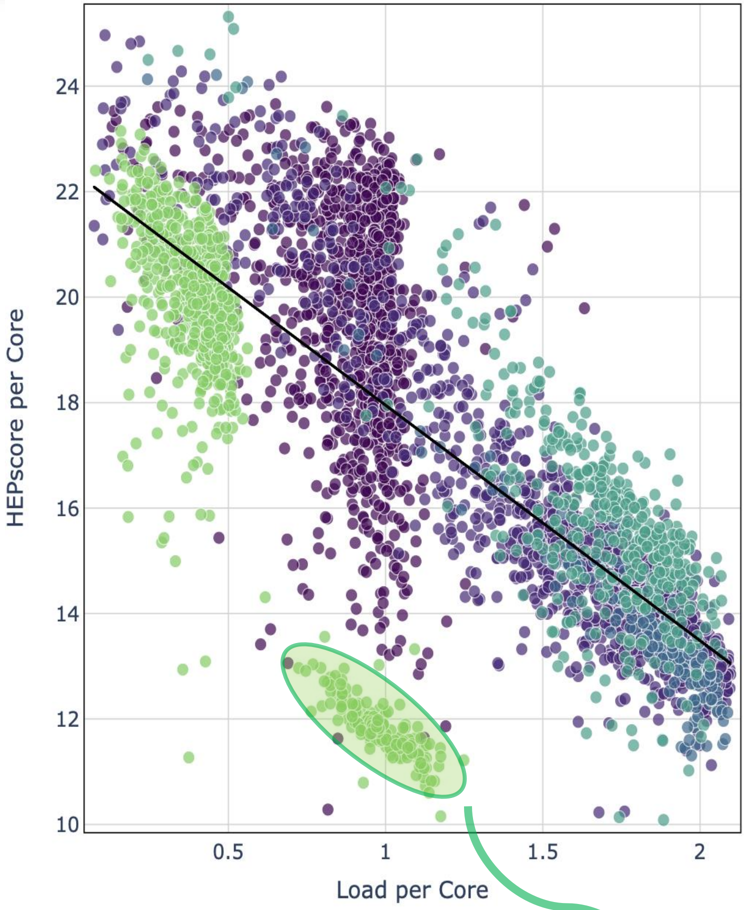


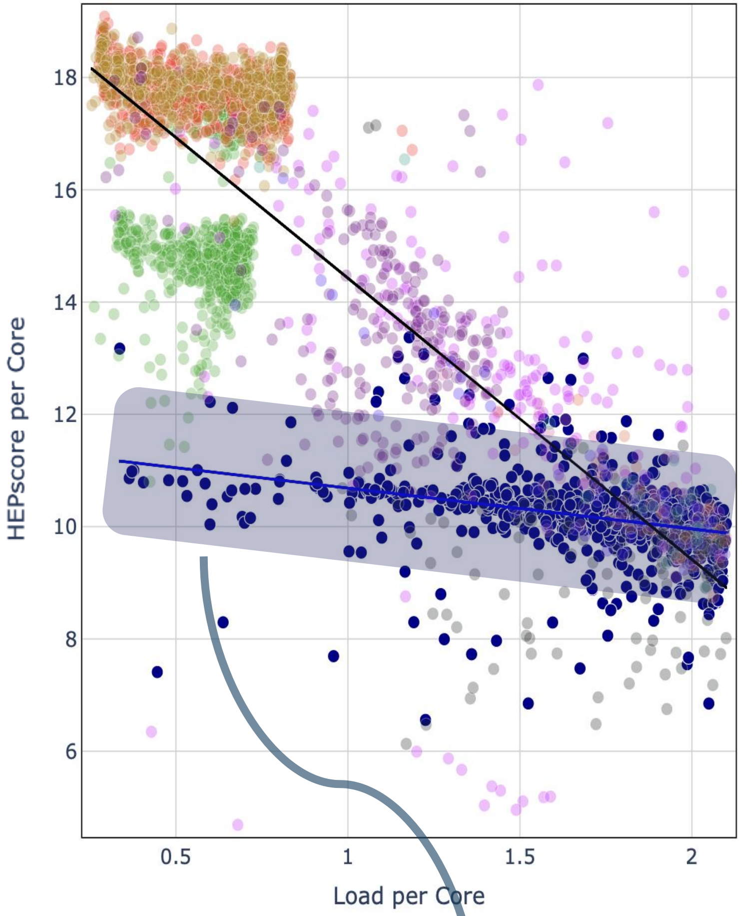
Three Types of Clustered Anomalies on the Grid

- **"Underperformance"** type of anomalies
  - Identified by a cluster of points that fall below the general trendline
  - Typically, far from the all-sites trendline

- **"Overperformance"** type of anomalies
  - Identified by a cluster of points that fall above the general trendline
  - Typically, far from the all-sites trendline

- **"Other"** type of anomalies
  - This type of anomaly can be characterized by a "flat" area of data points (highlighted on the plot) or any other trendline which is "unusual" comparing to the general trend





Underperformance anomaly cluster

"Other" type of anomaly

Anomaly Detection Machine Learning Models

Types of Anomaly Detection Models:

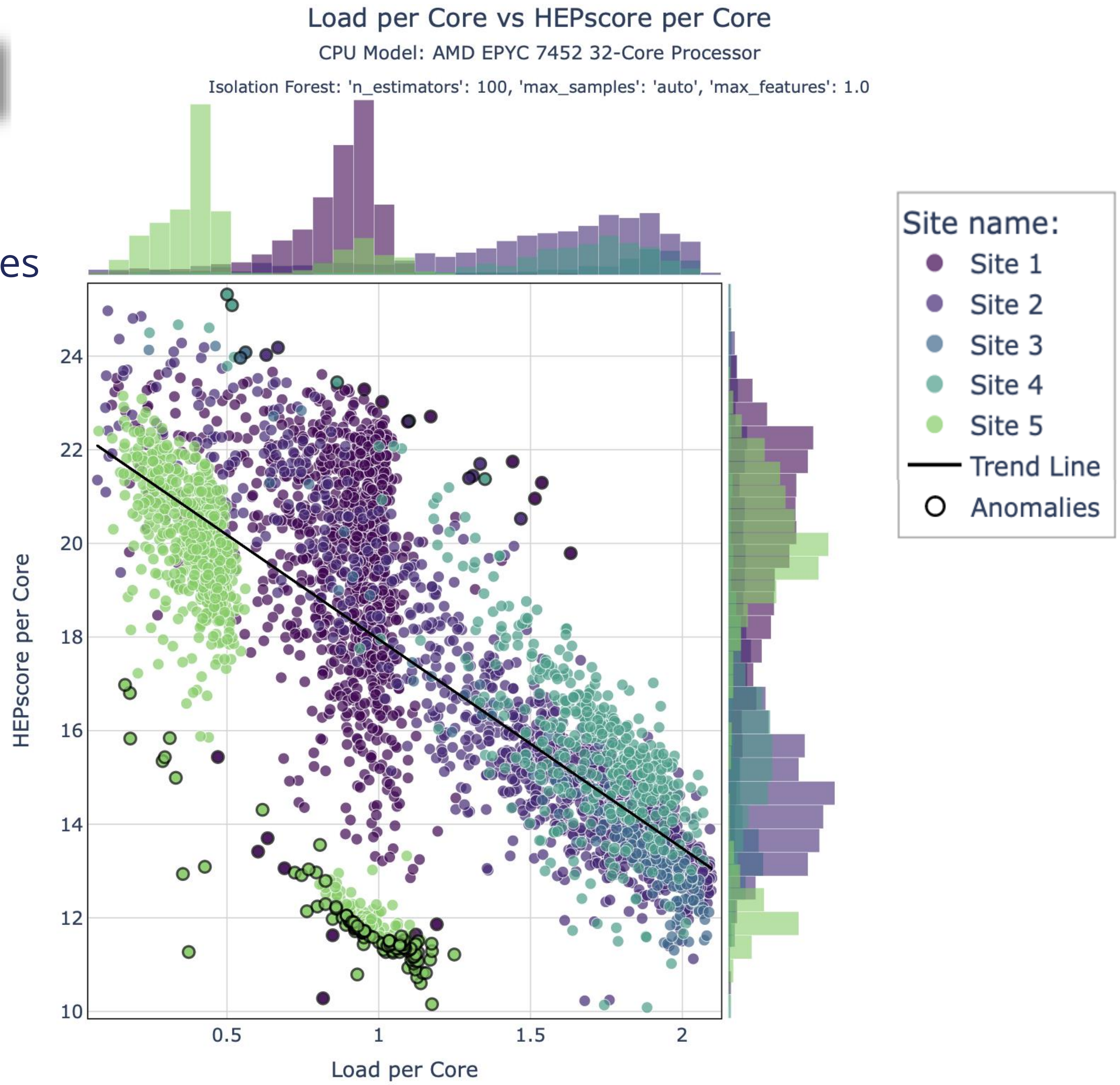
- Z-Score
- K-Nearest Neighbors (k-NN)
- Local Outlier Factor (LOF)
- DBSCAN
- One-Class SVM
- **Isolation Forest**
- **Isolation Forest with Split-selection Criterion (SCiForest)**
- Autoencoders
- Principal Component Analysis (PCA)
- T-Distributed Stochastic Neighbor Embedding (t-SNE)

Isolation Forest:

- Effective at detecting global scattered anomalies
- Uses random splits with hyperplanes in the feature space to isolate anomalies
- Generalizes well
- Preliminary results shown on the next slide

Isolation Forest Preliminary Results

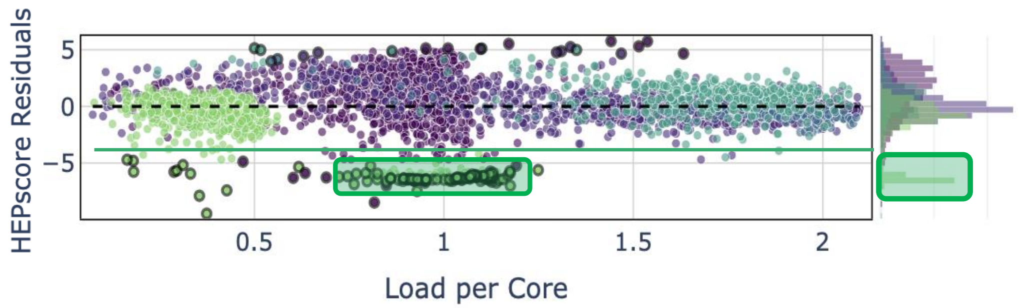
- Effectively detects global scattered anomalies
- Sometimes fails to detect global clustered anomalies
- A lot of False Positives and False Negatives

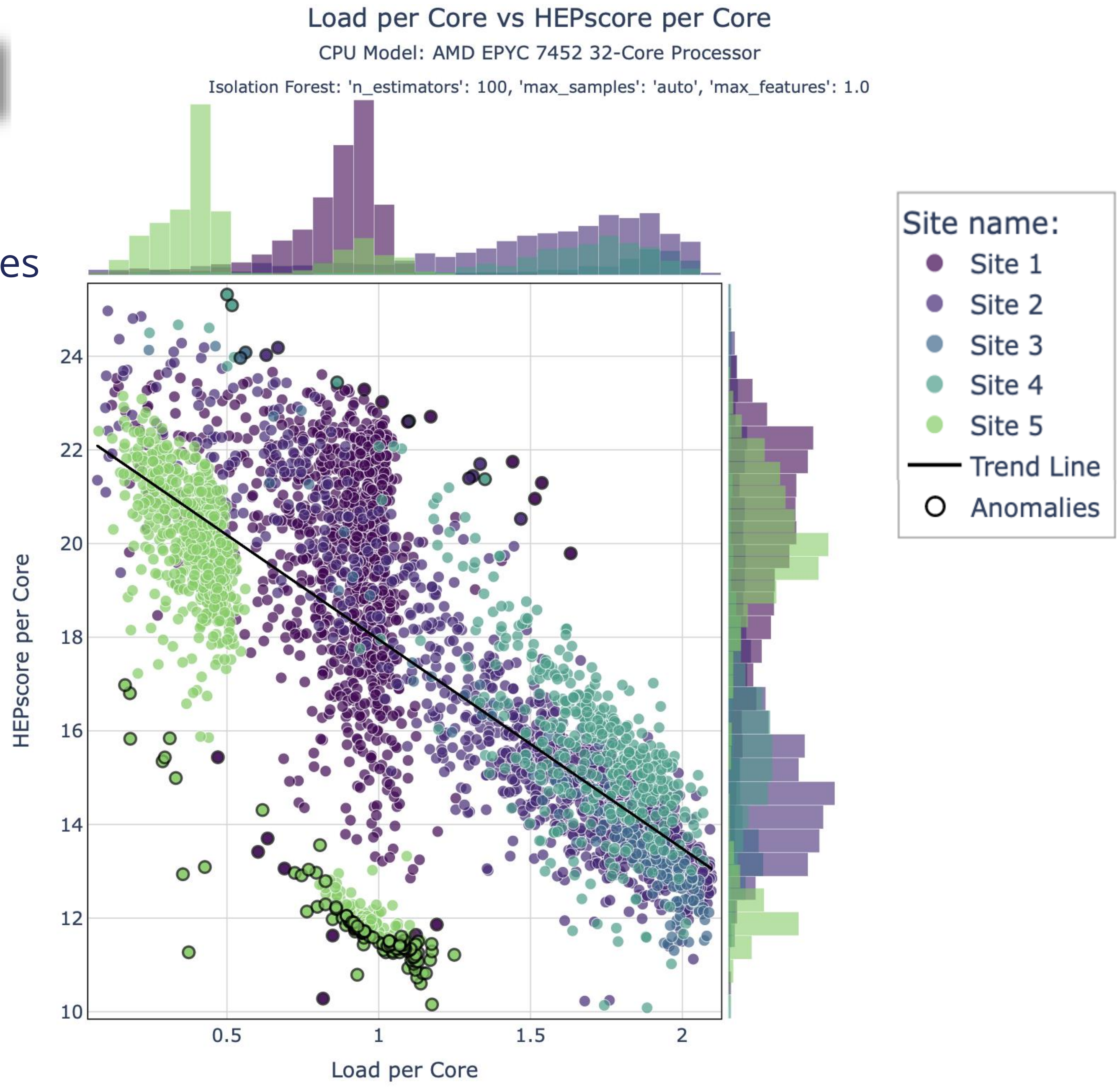


Isolation Forest Preliminary Results

- Effectively detects global scattered anomalies
- Sometimes fails to detect global clustered anomalies
- A lot of False Positives and False Negatives

- **SCiForest** should help with this based on the literature review
  - Well-suited for detecting both global scattered and clustered anomalies
  - Utilizes improved split selection with hyperplanes to better separate local distribution peaks





Next Steps

- Implementation of **Isolation Forest with Split-selection Criterion (SCiForest)**
- Results validation and different models comparison
- Add labels or anomaly score from the unsupervised learning algorithm to the dataset as ”ground truth”
- Apply a semi-supervised learning algorithm, such as XGBoost Outlier Detection (XGBOD), on the entire dataset with all features and “ground truth”
- Generate JSON report file with results and save plots with anomalies

Example JSON:

September

```
{
    'cpu1': {'site1':'overperformance'},
    'cpu2': {'site1':'other','site2':'underperformance'}
}
```

October

```
{
    'cpu1': {'site1':'overperformance', 'site2':'underperformance'},
    'cpu2': {'site1':'other','site2':'underperformance'}
}
```

Thank you for your attention

Questions?

kacperkozik999@gmail.com
kacper.kamil.kozik@cern.ch

linkedin.com/in/kacper-kozik

@Kacper0199

CERN openlab

References

Giordano D. *et. al.* HEPScore: A new CPU benchmark for the WLCG.
arXiv:2306.08118v2 [hep-ex] (2023).
https://doi.org/10.48550/arXiv.2306.08118

Giordano D., Alef M., Atzori L. *et al.* HEPiX Benchmarking Solution for WLCG Computing Resources.
Comput Softw Big Sci 5, 28 (2021).
https://doi.org/10.1007/s41781-021-00074-y

Valassi A. *et. al.* Using HEP experiment workflows for the benchmarking and accounting of WLCG computing resources.
EPJ Web Conf 245:07035 (2020).
https://doi.org/10.1051/epjconf/202024507035

Han S. *et. al.* ADBench: Anomaly Detection Benchmark.
arXiv:2206.09426v2 [cs.LG] (2022).
https://doi.org/10.48550/arXiv.2206.09426

Liu F.T. *et. al.* On Detecting Clustered Anomalies Using SCiForest.
In: Balcázar J.L., Bonchi F., Gionis A., Sebag M. (eds) Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2010. Lecture Notes in Computer Science, vol 6322.
Springer, Berlin, Heidelberg (2010).
https://doi.org/10.1007/978-3-642-15883-4_18