Employing Machine Learning Models for CESM Timing Data

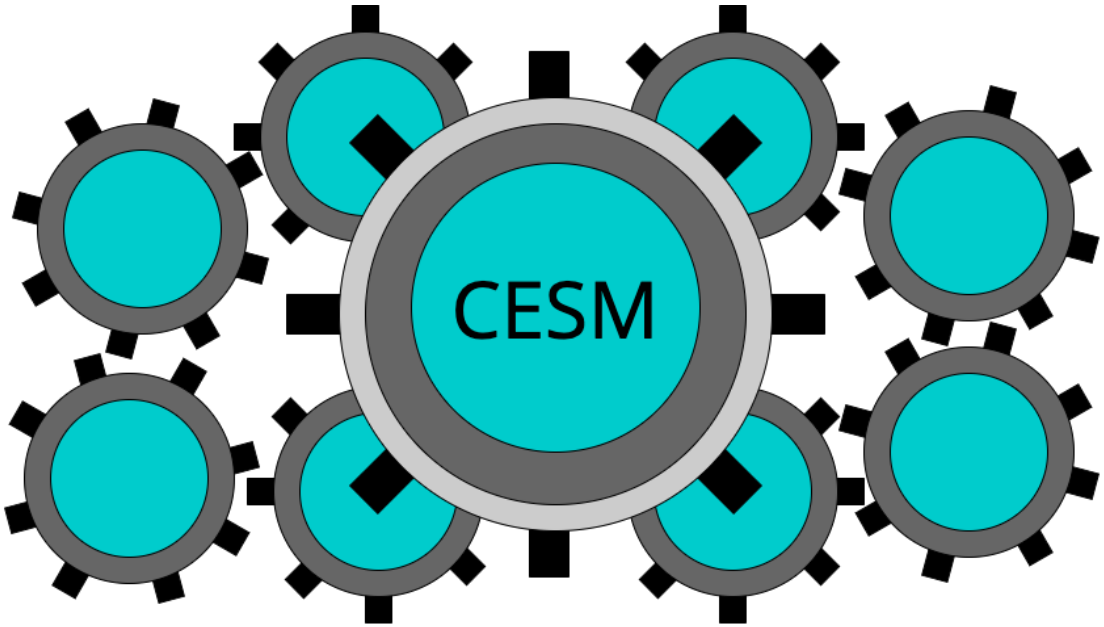
Thomas Johnson III
+Sheri Mickelson, Brian Dobbins, and John Dennis

July 28, 2021

This material is based upon work supported by the National Center for Atmospheric Research, which is a major facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977.

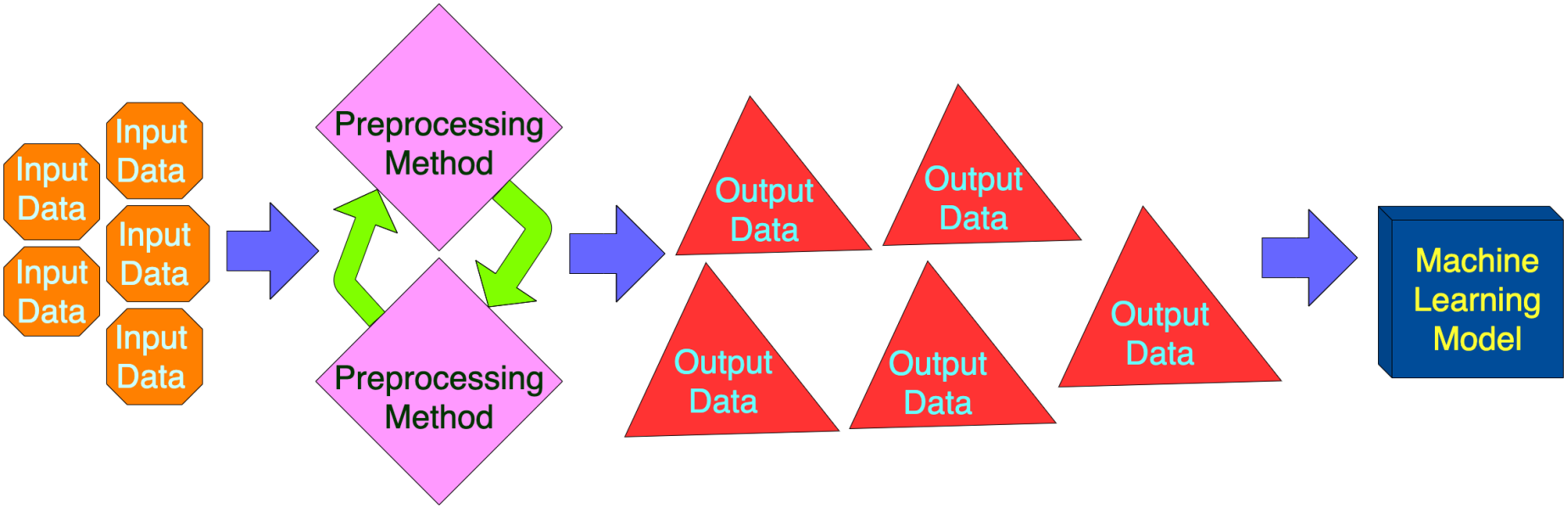
Motivation for Workflow

- The CESM community is perpetually developing.
- Community members want more tools to be able to enhance CESM capabilities and usability.
  - Processing CESM timing data for machine learning.
- High level goal is to predict performance.



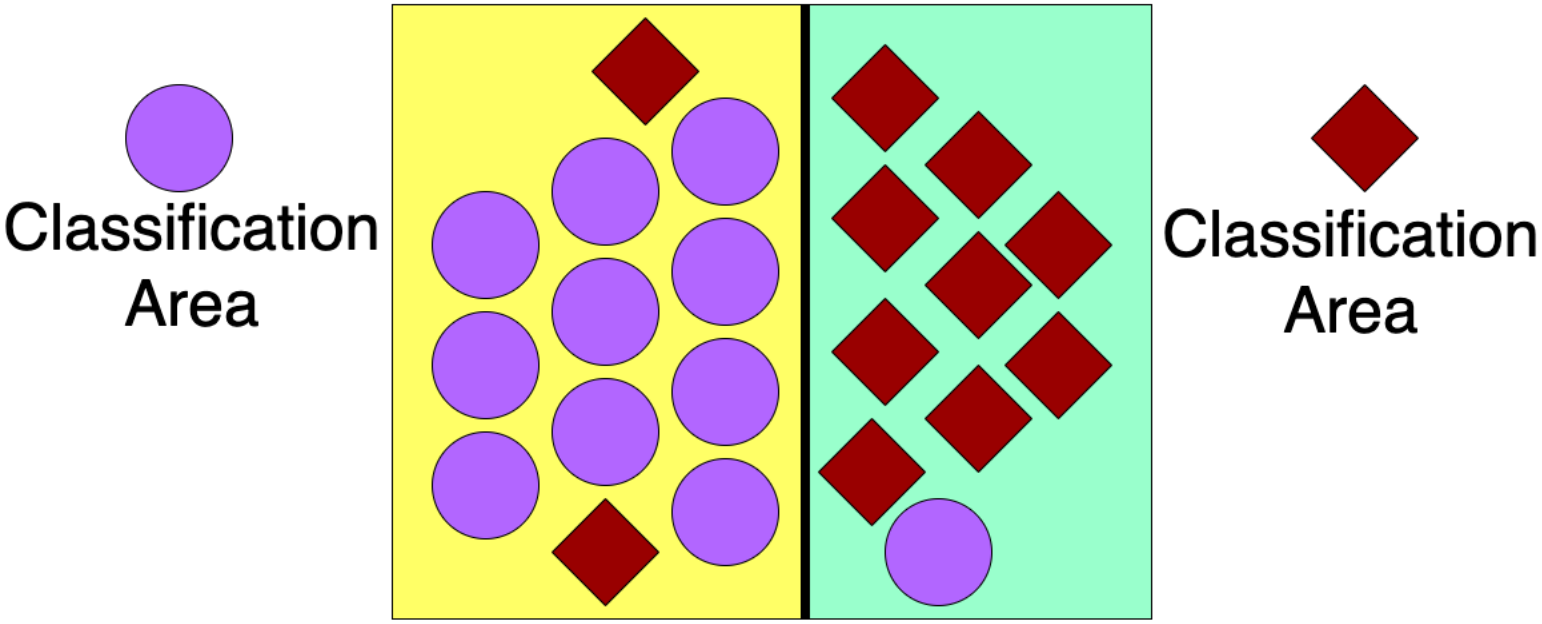
What is Machine Learning

- Machine learning centers on the usage of a subset of algorithms.
- These algorithms seek to become more efficient or effective at a given task.
- The algorithms are trained by providing data to learn from.



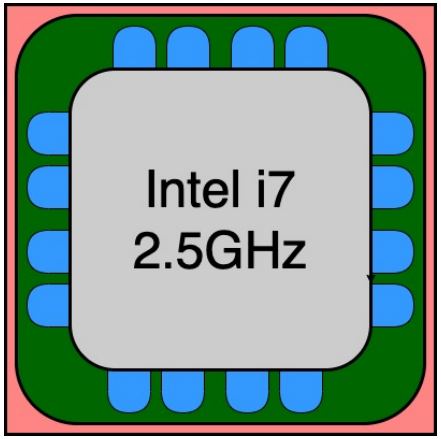
Processor Comparison Dataset Motivation

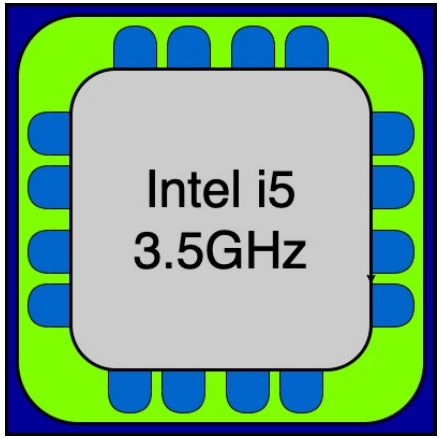
- Simpler scenario to test the workflow.
  - Measure how effectively the machine learning methods can classifying different systems.
- Performing a classification of the hardware to later extend to more complex scenarios.



Processor Comparison Dataset Processing

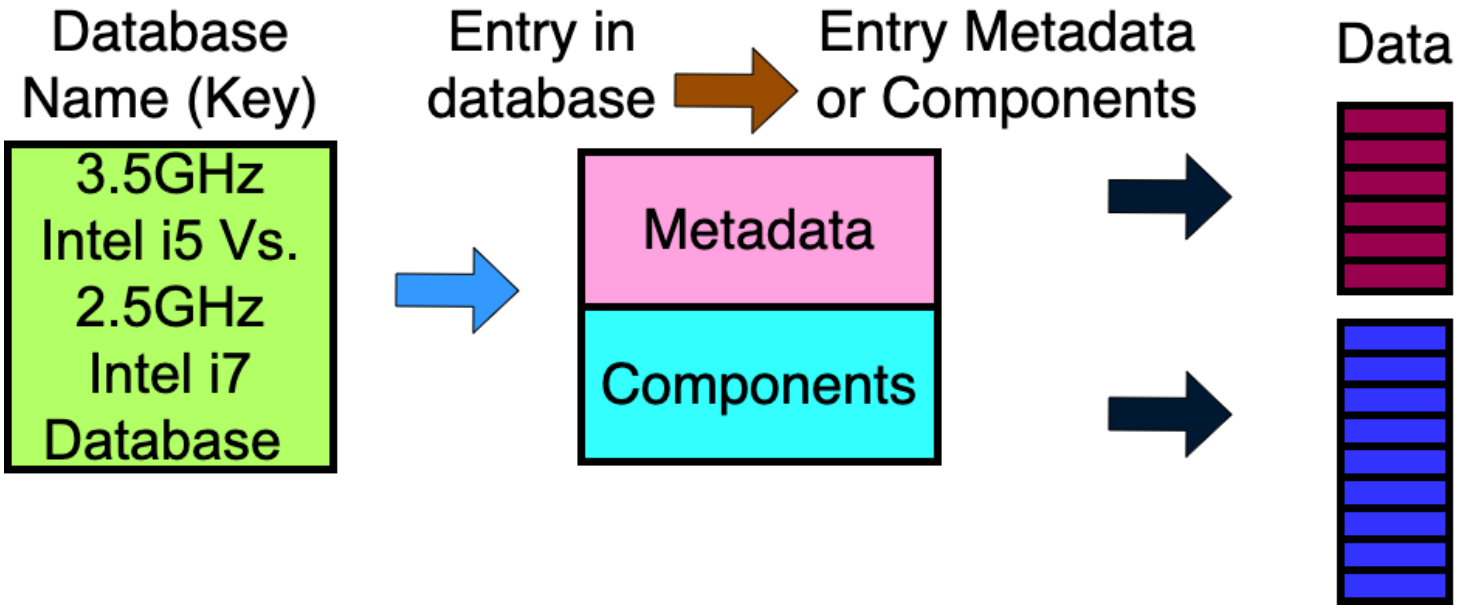
- Goal: For classification, could the models distinguish between the 3.5GHz Intel i5 vs. 2.5GHz Intel i7 runs.
- Data preprocessed with One Hot Encoding and Standard Scaling [1] [2].
- Models evaluated: SVM, Decision Trees, Random Forests, Multi-layer Perceptron, KNN.
- Principal Component Analysis, Select K Best methods.





Processor Comparison Data

- Using different hardware, but same run parameters.
- Same containerized CESM compset used: Aqua Planet.
- 4 cores used for each model run.
- Running at 5, 10, 15 model days for 6 runs each on both 3.5GHz Intel i5 Vs. 2.5GHz.

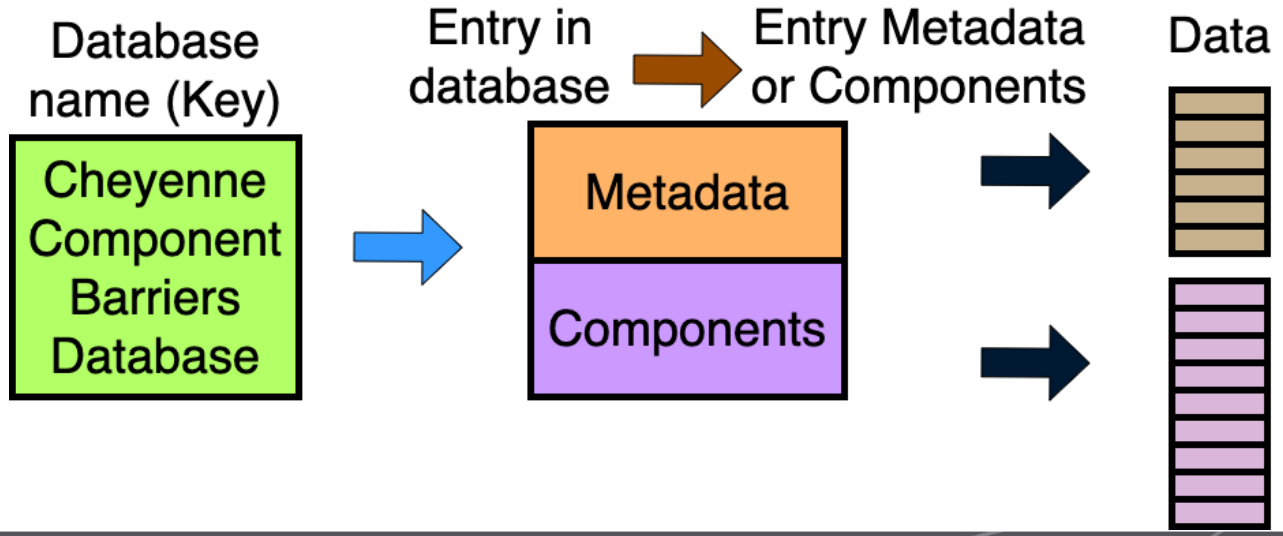


Barrier Comparison Dataset Motivation

- The print statements in the component barriers have been commented out.
- The standing idea is that component barriers on will influence performance.
  - Exploring the performance of the machine learning methods to see if a significant difference can be found.
- Provides a more complex scenario for the machine learning methods.
- Provide some insight as to whether the component barriers are significantly affecting CESM runs.

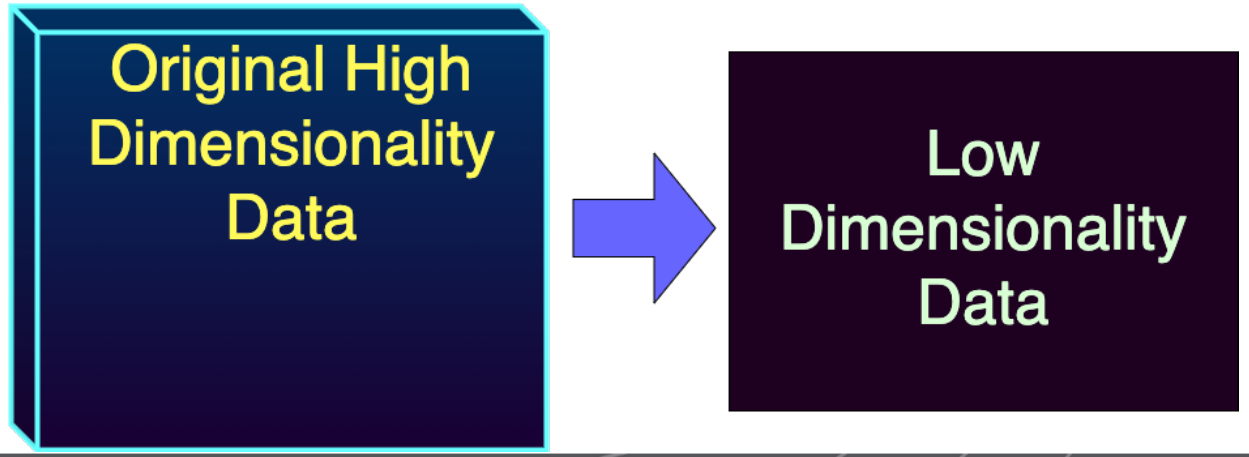
Barrier Comparison Dataset Processing

- Goal: Classifying whether the input data was from a barriers on data point or barriers off data point.
- Similar overall process, with Recursive Feature Elimination (RFE) added for alternative feature selection.
- All ran on Cheyenne.
- All other parameters similar to 3.5GHz Intel i5 Vs. 2.5GHz Intel i7 experiment.



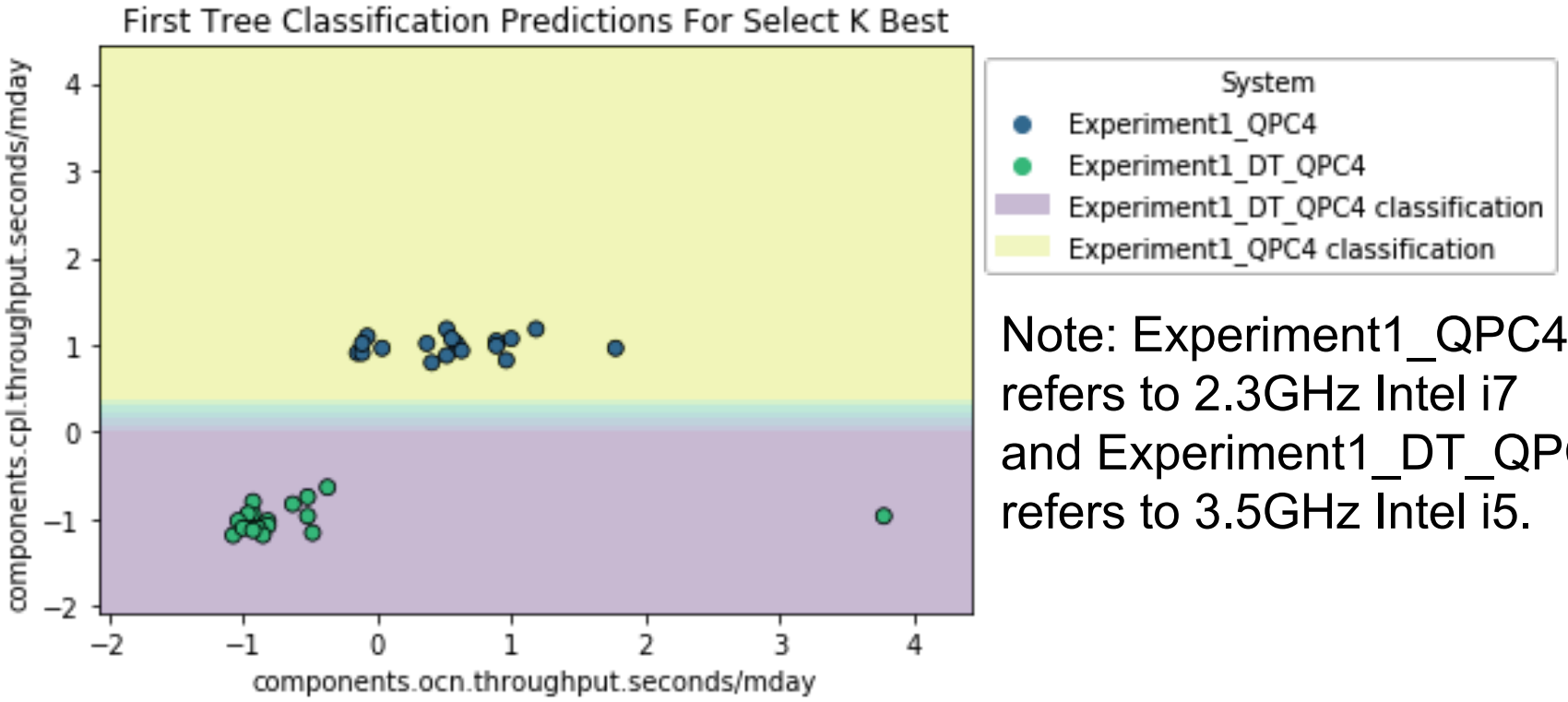
Dimensionality Reduction and Feature Selection Methods

- PCA
  - Using linear algebra operations to combine features into new features
- Select K Best
  - Using a metric to select k amount of features from the total features
  - Mutual Information Classifier
- RFE
  - Builds a model and uses said model's metrics to select k features
  - Decision tree



Overview of Results for 3.5GHz Intel i5 Vs. 2.5GHz Intel i7 Dataset

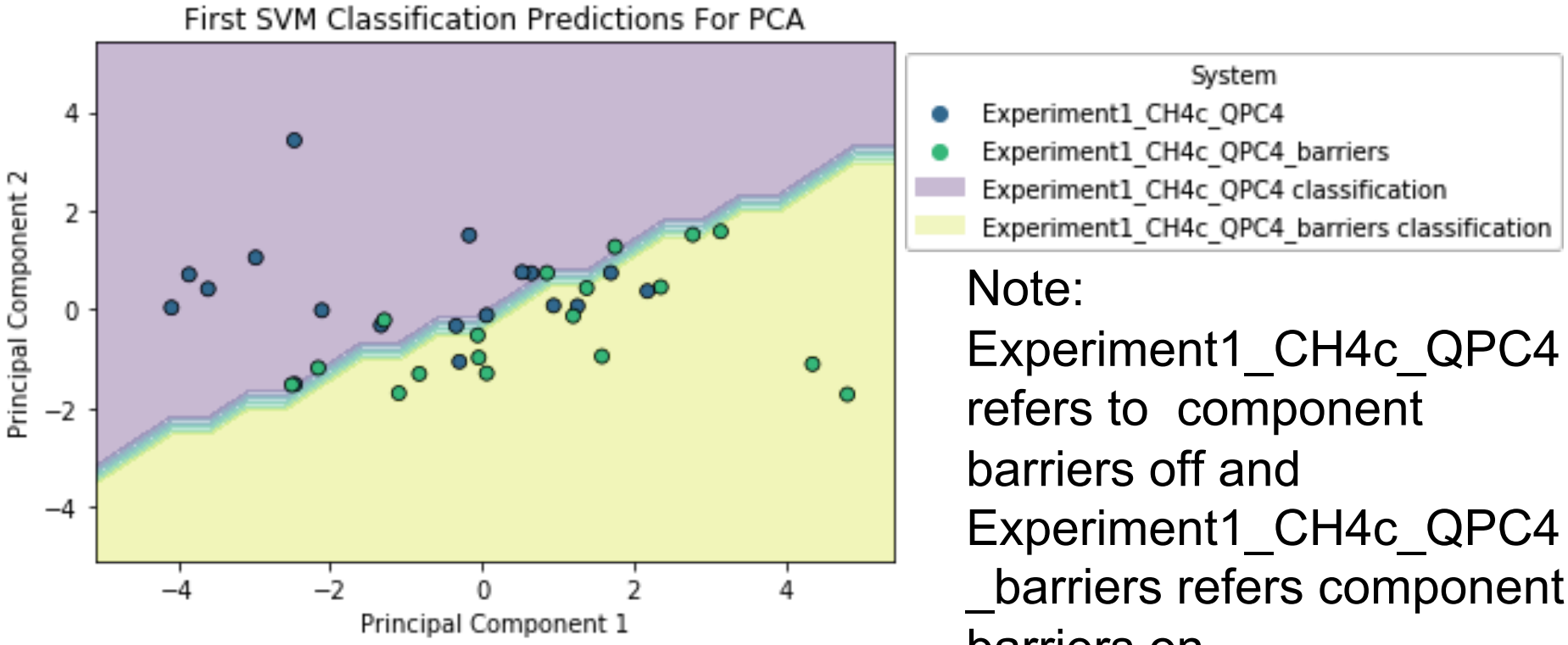
- 3.5GHz Intel i5 Vs. 2.5GHz Intel i7 accuracy results:
  - Highest mean accuracy is 99.2% from Decision Trees with Select K Best (example shown in plot below).
  - Lowest mean accuracy is 83.4% from Decision Trees with PCA.



Note: Experiment1_QPC4 refers to 2.3GHz Intel i7 and Experiment1_DT_QPC4 refers to 3.5GHz Intel i5.

Overview of Results Cheyenne Component Barriers on vs Component Barriers Off

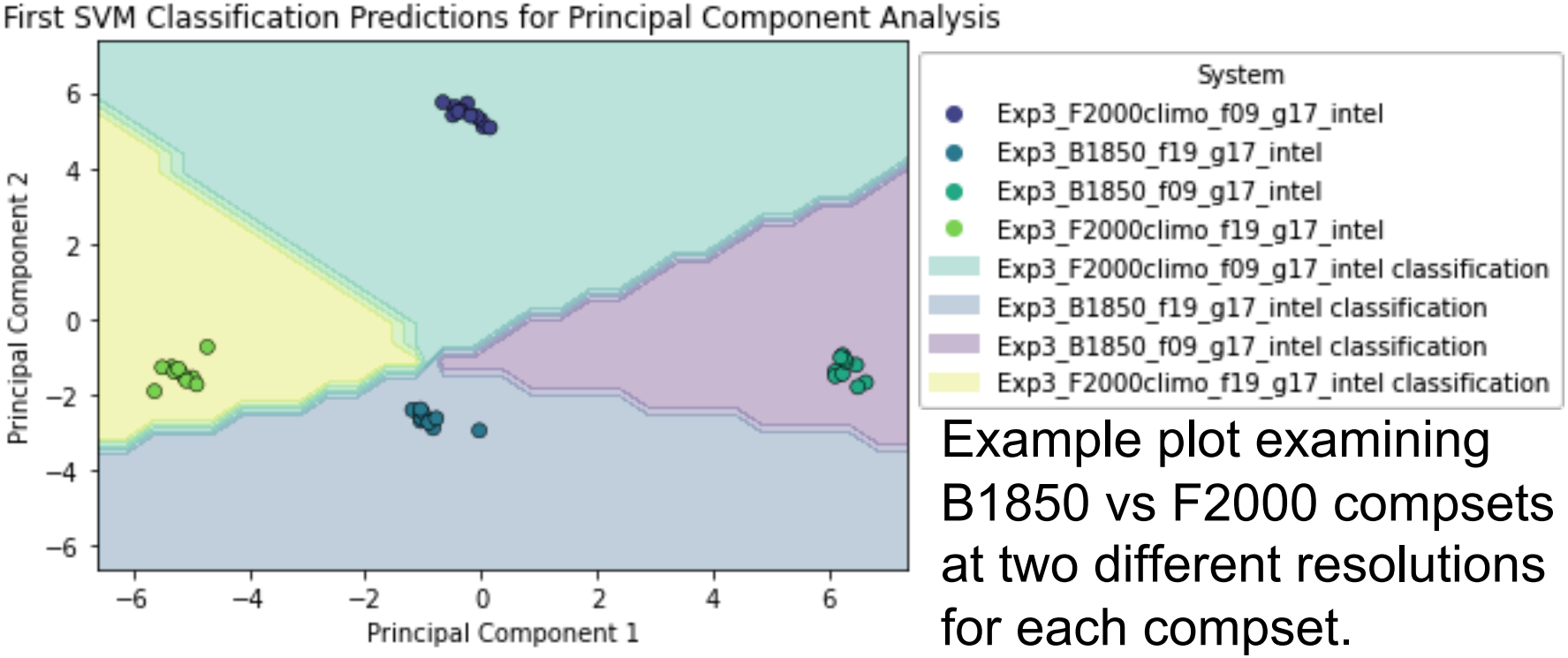
- Cheyenne component barriers on vs component barriers off
  - Highest mean accuracy is 66.1% from SVMs with PCA (example plot shown below).
  - Lowest mean accuracy is 30.1% from random forests with Select K Best.



Note: Experiment1_CH4c_QPC4 refers to component barriers off and Experiment1_CH4c_QPC4 _barriers refers component barriers on.

Future Work

- Increasing data available.
- Adding further machine learning models.
- Implementing widgets for an interactive notebook.
- Exploring more complex scenarios:
  - Exploring compsets and resolutions



Example plot examining B1850 vs F2000 compsets at two different resolutions for each compset.

References

1. Scikit-learn Developers. *sklearn.preprocessing.OneHotEncoder.* 2020. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OneHotEncoder.html.
2. Scikit-learn Developers. *Sklearn.preprocessing.StandardScaler.* 2020. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html.
3. Brownlee, Jason. *LOOCV for Evaluating Machine learning Algorithms.* July 27, 2020. https://machinelearningmastery.com/loocv-for-evaluating-machine-learning-algorithms/.
4. Schneider, Jeff. *Cross Validation*. 1997. https://www.cs.cmu.edu/~schneide/tut5/node42.html.

Acknowledgments

- I would like to thank my excellent mentors Sheri Mickelson, Brian Dobbins, and John Dennis for their guidance and expertise.
- Thanks AJ Lauer, Virginia Do, Max Galbraith, Jerry Cyccone and the SIParCS program as a whole.
- Thanks to TDD, NCAR, and CISL.
- Special Thanks to the NSF.

Questions

Project GitHub Repository QR Code:
https://github.com/NCAR/SIParCS-2021-Johnson

Thomas Johnson III's:

LinkedIn:
https://www.linkedin.com/in/thomas-j-3804a7a6/

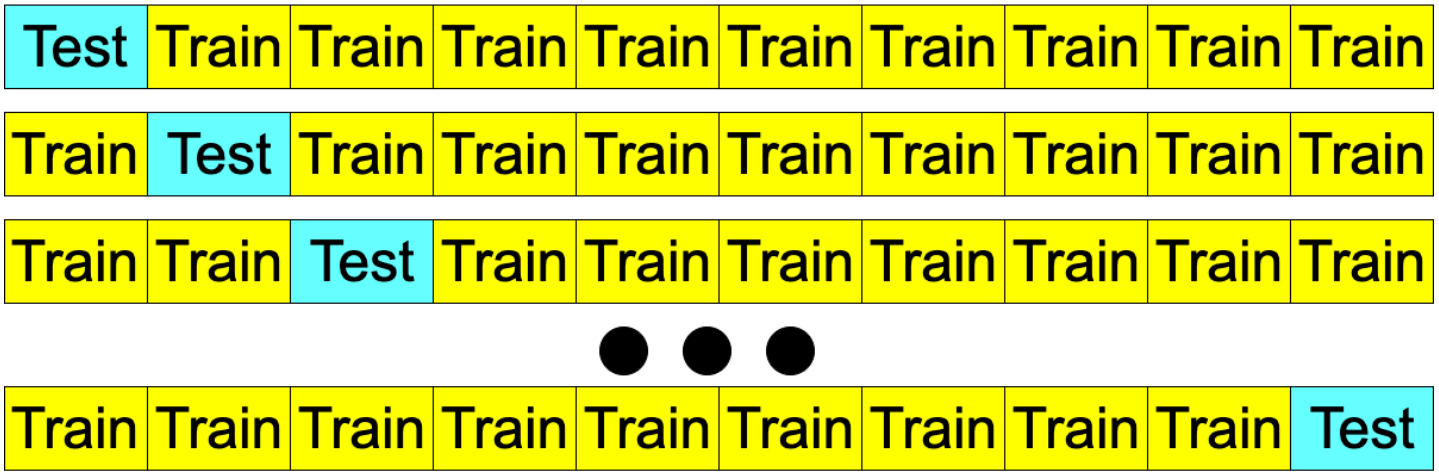
GitHub:
https://github.com/Herok4Build

ORCID:
https://orcid.org/0000-0002-7767-7509

Cross Validation Methods

- Utilizing Leave One Out Cross Validation, abbreviated LOOCV.
  - The preferred cross validation strategy for tiny datasets [3] [4].
- 10-Fold Stratified Cross Validation, abbreviated SCV [4].

Diagram of 10-Fold Cross Validation



Number of Features for Data

- Cheyenne component barriers on vs. component barriers off using 86 features.
- 3.5GHz Intel i5 Vs. 2.5GHz Intel i7 dataset using 86 features.

3.5GHz Intel i5 Vs. 2.5GHz Intel i7 Runs of Aqua World Results

Model Group	Mean of LOOCV for PCA	Mean of 10-Fold SCV for PCA	Mean of LOOCV for Select K Best	Mean of 10-Fold SCV for Select K Best
SVMs	89.4%	91.7%	97.6%	97.8%
Decision Trees	83.4%	83.5%	99.2%	98.8%
Random Forests	85.8%	86.6%	98%	97.9%
Multi-layer Perceptron Neural Network	87.1%	89.2%	94.4%	94.5%
KNN	86.7%	87.7%	95.1%	95%

Component Barriers On Versus Component Barriers Off Results

Model Group	Mean of LOOCV for PCA	Mean of 10-Fold SCV for PCA	Mean of LOOCV for Select K Best	Mean of 10-Fold SCV for Select K Best
SVMs	66.1%	66.6%	36.2%	54.4%
Decision Trees	54.6%	57.6%	32.2%	50.4%
Random Forests	62.6%	62.7%	30.1%	47.2%
Multi-layer Perceptron Neural Network	65.1%	66.6%	36.7%	52.6%
KNN	57.1%	62.6%	49.3%	54.5%

Component Barriers On Versus Component Barriers Off Results

Model Group	Mean of LOOCV for Recursive Feature Elimination	Mean of 10-Fold SCV for Recursive Feature Elimination
SVMs	65.6%	65%
Decision Trees	63.6%	62%
Random Forests	57.8%	58.6%
Multi-layer Perceptron Neural Network	64%	62%
KNN	59.3%	61.5%