CROSS-PLATFORM ANALYSIS OF CLOUD RESOURCE UTILIZATION PATTERNS FOR OPTIMIZED RESOURCE ALLOCATION

CS 8803: Datacenter Networks & Systems (Spring 2025)

Akshat Karwa, Mehul Rastogi

Table of Contents

Introduction

Background & Challenges

- We observed that there is a critical gap existing between the cloud resources allocated versus utilized creating inefficiencies.

- Moreover, predicting the amount of resources required is extremely challenging. Thus, cloud providers end up over provisioning resources.

- On top of this, there are variations in workloads which lead to suboptimal resource allocation.

- Overall, these inefficiencies increase costs for consumers and reduce the efficiency of cloud providers making allocation suboptimal.

Introduction

Related Work & Current Research Gap -

- Analysing existing studies, we found that their focus is on the isolated analysis of individual cloud providers: Google* and Alibaba*.

- There is limited comparative research and analysis available that can clearly show contrast between the major cloud providers.

- There is no clear generalization of patterns recognized across different cloud providers. Therefore, the most optimal techniques have not been identified yet.
  We aim to reduce this lack of cross-platform insights into common patterns and provider-specific approaches.

* Reiss, C., Tumanov, A., Ganger, G. R., Katz, R. H., Kozuch, M. A., Intel Science and Technology Center for Cloud Computing, & Carnegie Mellon University. (2012) Towards understanding heterogeneous clouds at scale: Google trace analysis (Report ISTC-CC-TR-12-101)

* Lu, C., Ye, K., Xu, G., Xu, C.-Z., & Bai, T. (2017). Imbalance in the cloud: An analysis on Alibaba cluster trace.

Introduction

Motivation & Objectives

- We aim to compare trace data across 3 cloud providers - Google Cloud, Microsoft Azure, and Alibaba Cloud - to identify optimal resource management strategies and the inefficiencies that are either common between them or specific to providers.
- The goal is to:
    a. Develop insights which can be generalized and utilized to improve resource allocation strategies.
    b. Analyze diverse approaches of infrastructure management, evaluating their effectiveness.
    c. Help cloud providers reduce costs by eliminating unnecessary allocation of resources that will not be optimally utilized.
    d. Identify current best practices and suggest better infrastructure designs.
    e. Overall, we will try to increase data center efficiency.

Approach

- Preprocessing heterogeneous trace data
- Exploratory data analysis
- Informative plotting
- Predicting utilization using XGBoost Regression ML model
- Output comparison
- Drawing insightful conclusions

Challenges

- Large datasets
- Compute resources
- Heterogeneous data
- Parameter selection for training ML model
- Drawing/Developing insights from different observations and findings

Methodology

Data Sources

- Google Cluster Data (2019)
- Microsoft Azure Public Dataset (2019)
- Alibaba Cluster Trace Data (2018)

Methodology - Google Cluster Data (2019)

- Traces from Google clusters spanning 31 days of data from 2019.

- For each cluster, there are 8 different cells (a through h).

- Based on the Borg cluster management system, the data is further split into shards, where each shard has the following tables:

- **Core Tables:**
    - MachineEvents
    - MachineAttributes
    - CollectionEvents
    - InstanceEvents
    - InstanceUsage

Methodology - Google Cluster Data (2019)

- Data Cleaning
- Inner Join on Instance Usage Data and Instance Events Data
  - Instance Index
  - Collection ID
  - Machine ID
- Calculated CPU Utilization Percentage and Memory Utilization Percentages
- Instance-level & Machine-level Analysis
- Plotting
- Predictions

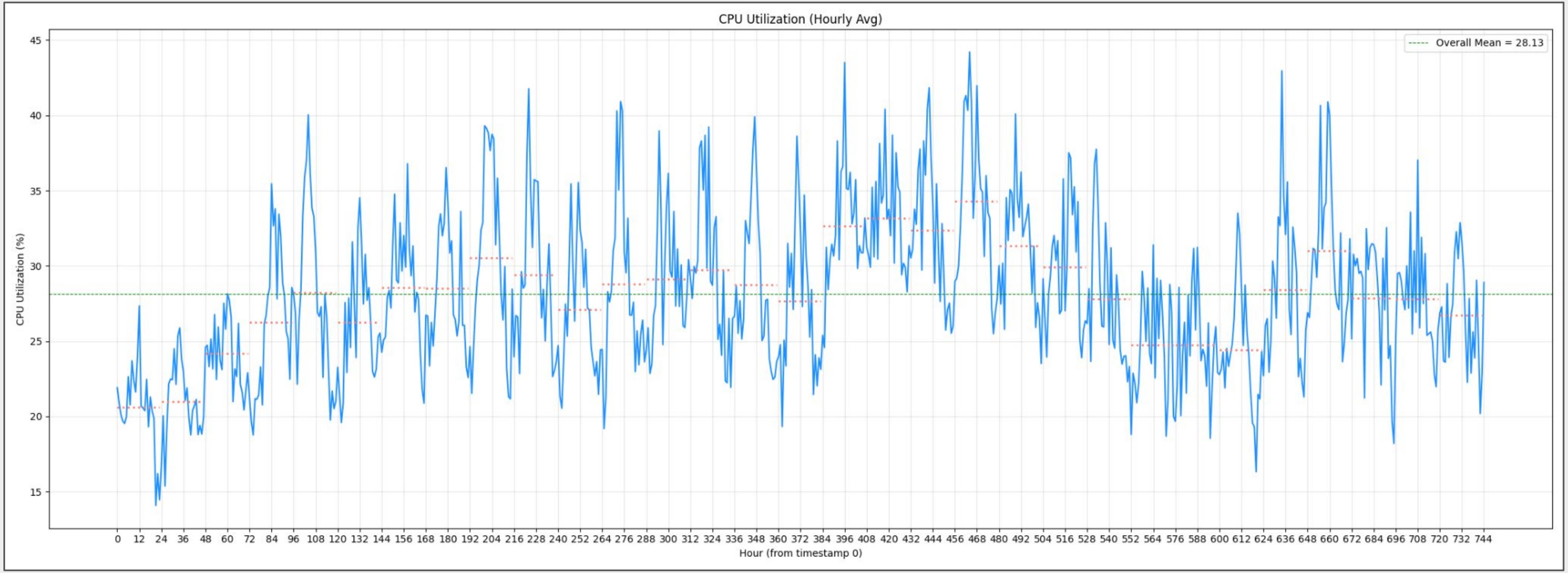
Methodology - Google Cluster Data (2019)

- Merged Instance Usage & Instance Events data tables

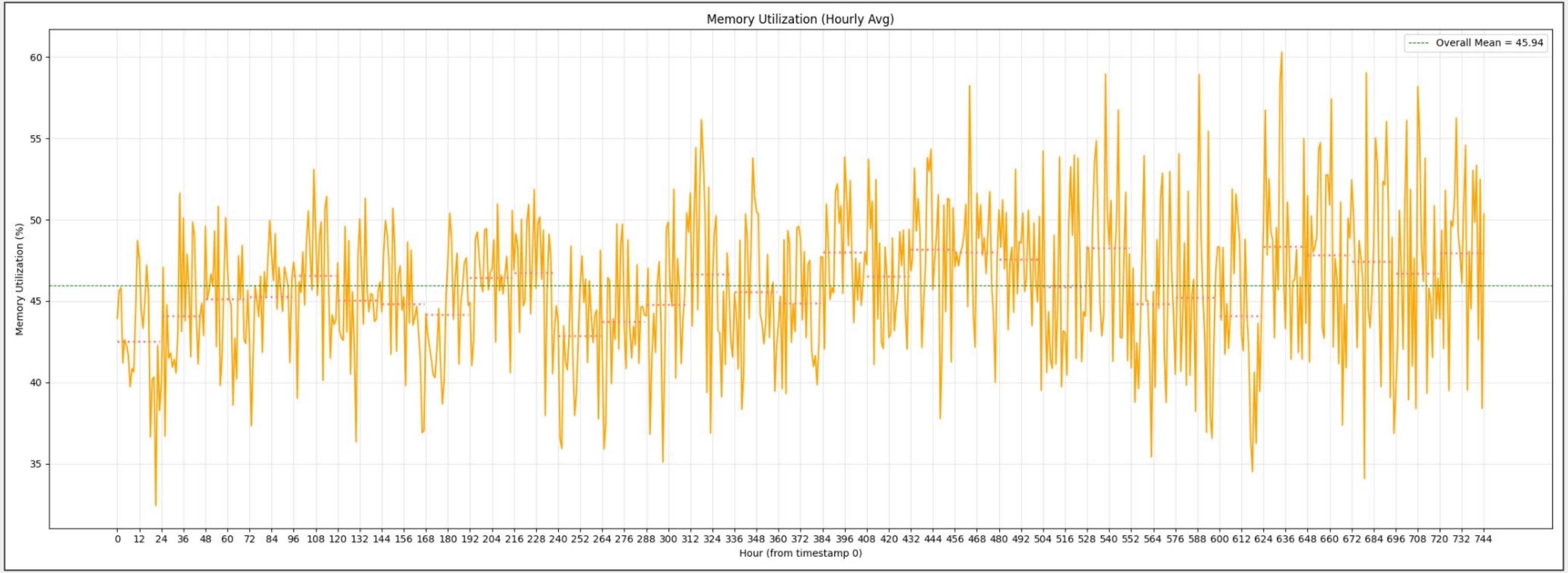
- Got time-series utilization data

start_time	end_time	collection_id	machine_id	type	scheduling_class	priority	cpus_util_perc	mem_util_perc
3000000000	3300000000	291839435167	92043472820	3	3	200	20.083333	88.477801
529000000	600000000	374675861423	1638822237	10	1	105	11.661808	22.525473
2808000000	2812000000	374909856633	35974924787	3	0	0	2.300861	17.358398
3000000000	3300000000	374675978279	2448218583	10	1	105	29.524887	81.513828
3000000000	3300000000	374675978279	2448218583	10	1	105	33.634021	81.513828

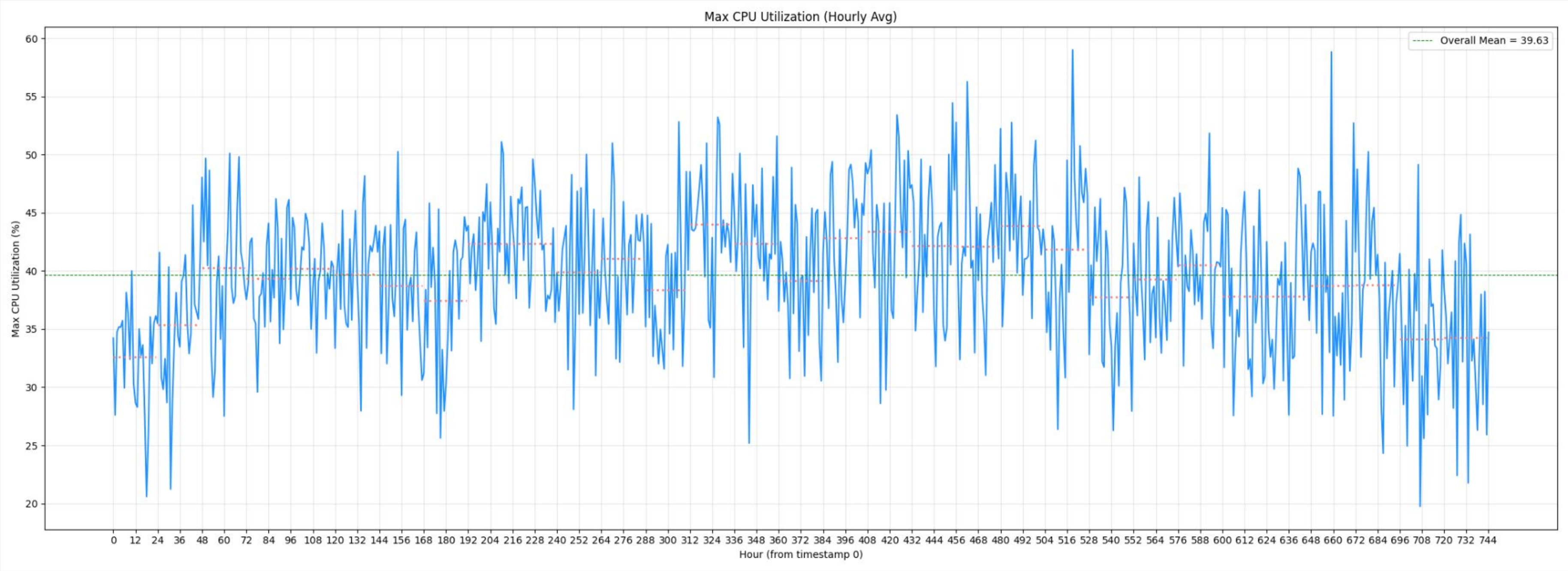
Instance Level Time Series Analysis - Google Cluster Data (2019)



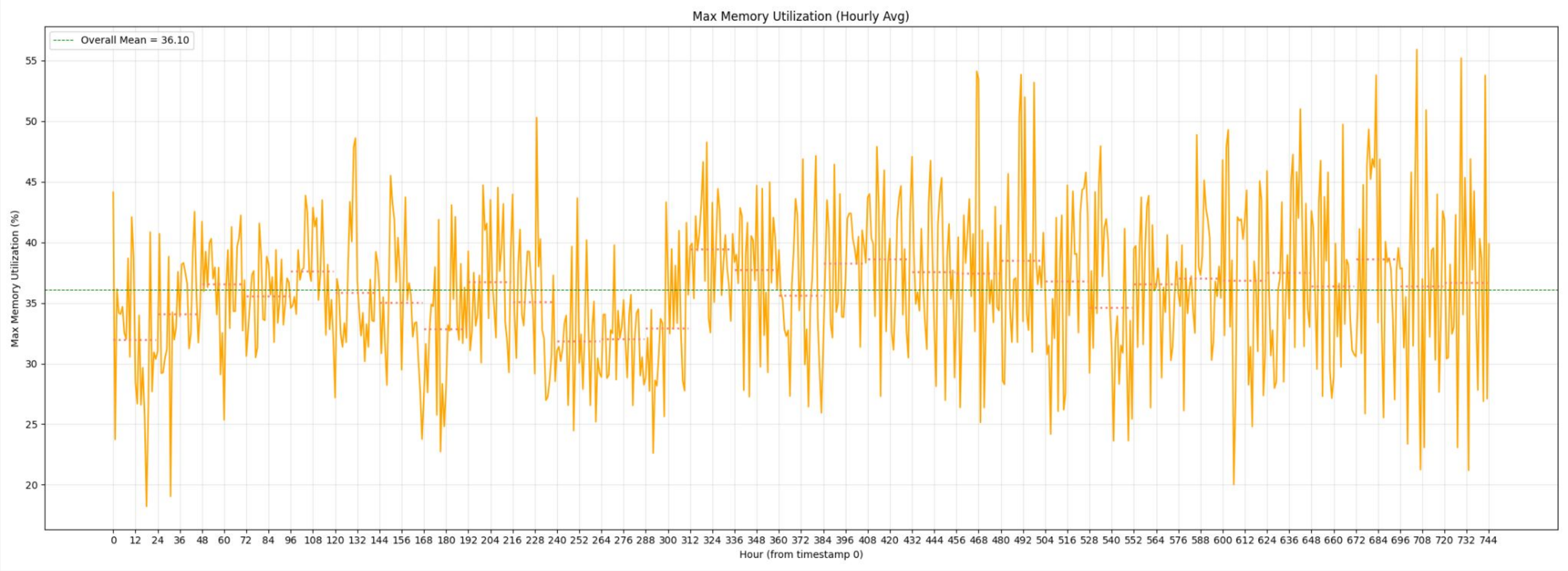
Instance Level Time Series Analysis - Google Cluster Data (2019)



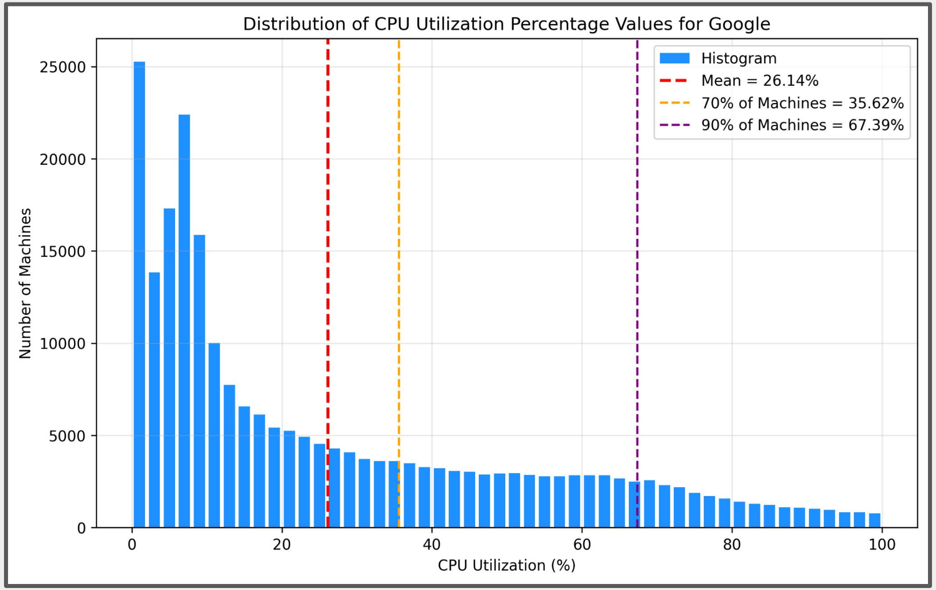
Instance Level Time Series Analysis - Google Cluster Data (2019)

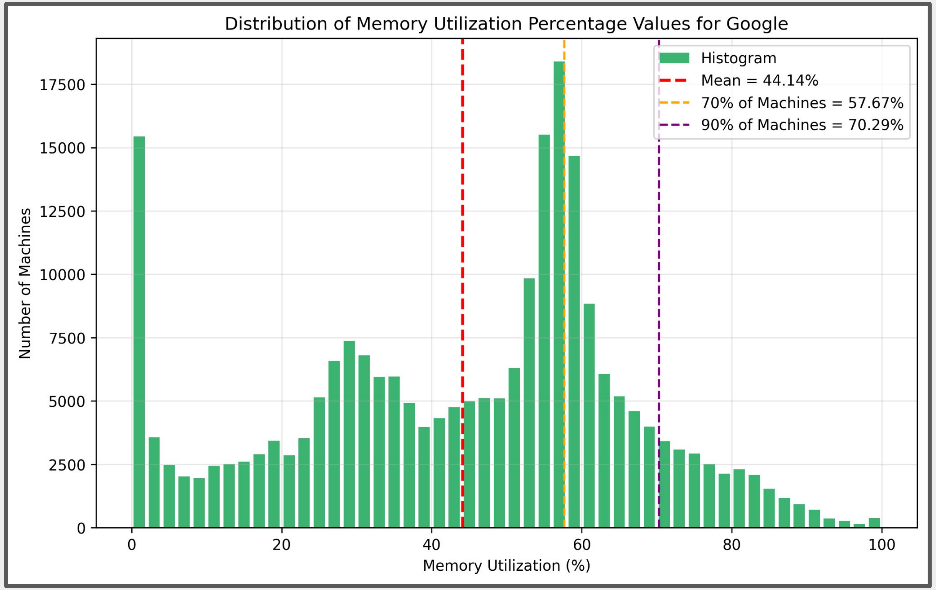


Instance Level Time Series Analysis - Google Cluster Data (2019)



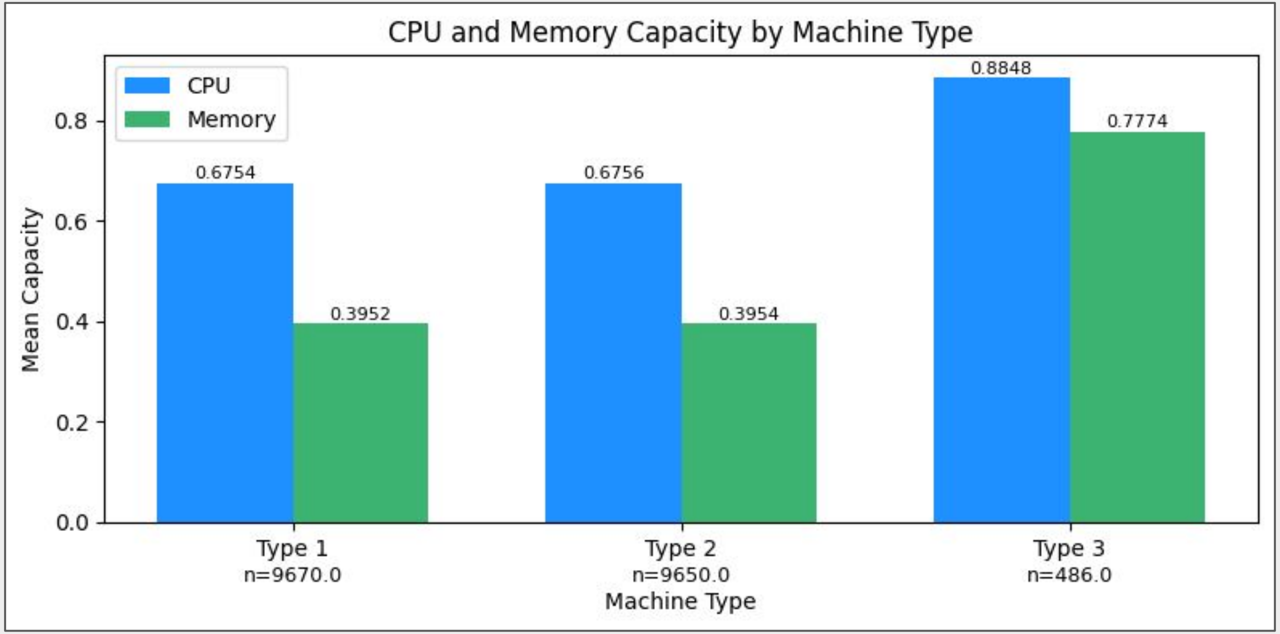
Machine-Level Analysis - Google Cluster Data (2019)





Georgia Tech

Task-Type Analysis - Google Cluster Data (2019)



Predictive Analysis - Google Cluster Data (2019)

- XGBoost Regression Machine Learning Model
  - 1500 Decision Trees
  - 95% Training Set
  - 5% Testing Set

- Training Data Variables:
  - Start Hour (0-23)
  - Total Time
  - Task Type
  - Task Priority
  - Scheduling Class

Predictive Analysis - Google Cluster Data (2019)

- CPU Utilization Evaluation Metrics
  - RMSE - 9.2035
  - Prediction Accuracy - 81.8068%
- Memory Utilization Evaluation Metrics
  - RMSE - 15.8080
  - Prediction Accuracy - 67.2593%

Methodology - Azure Cluster Data (2019)

- Traces from Microsoft Azure's clusters for the year 2019.

- **Core Tables:**
  - VMTable
  - Deployments
  - Subscriptions
  - VM_CPU_Readings

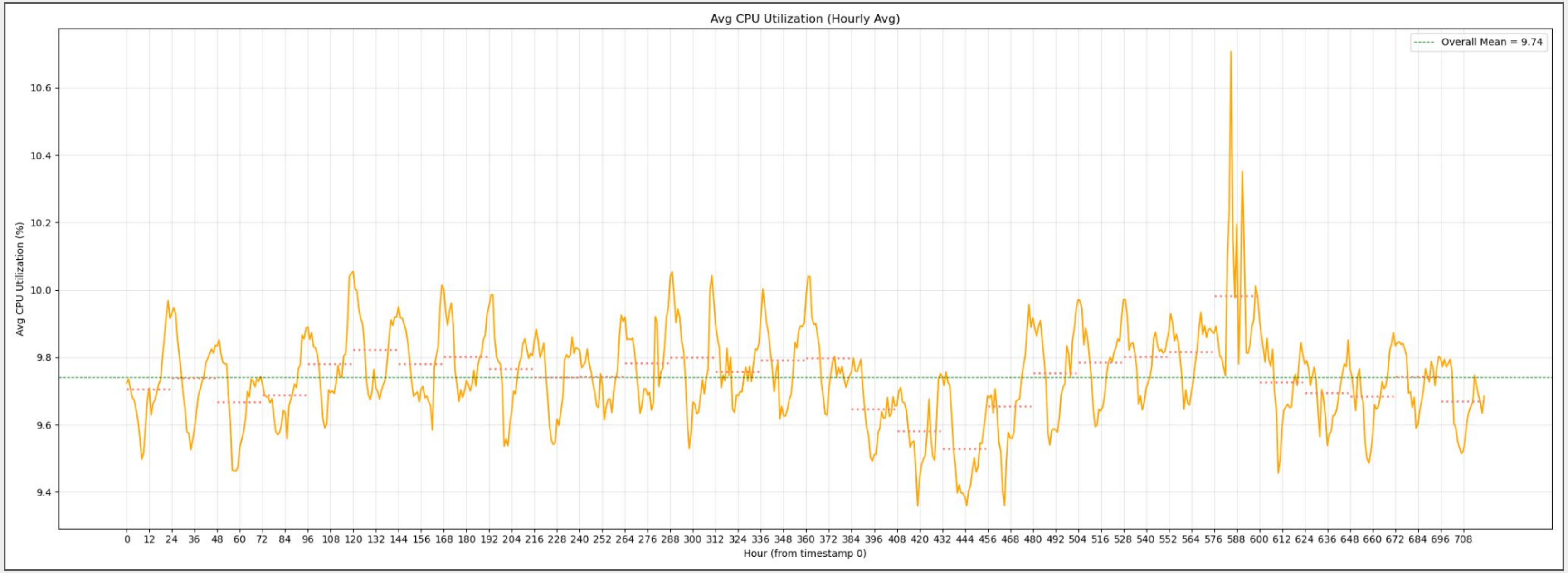
- The CPU readings are further divided into 195 shards.

Methodology - Azure Cluster Data (2019)

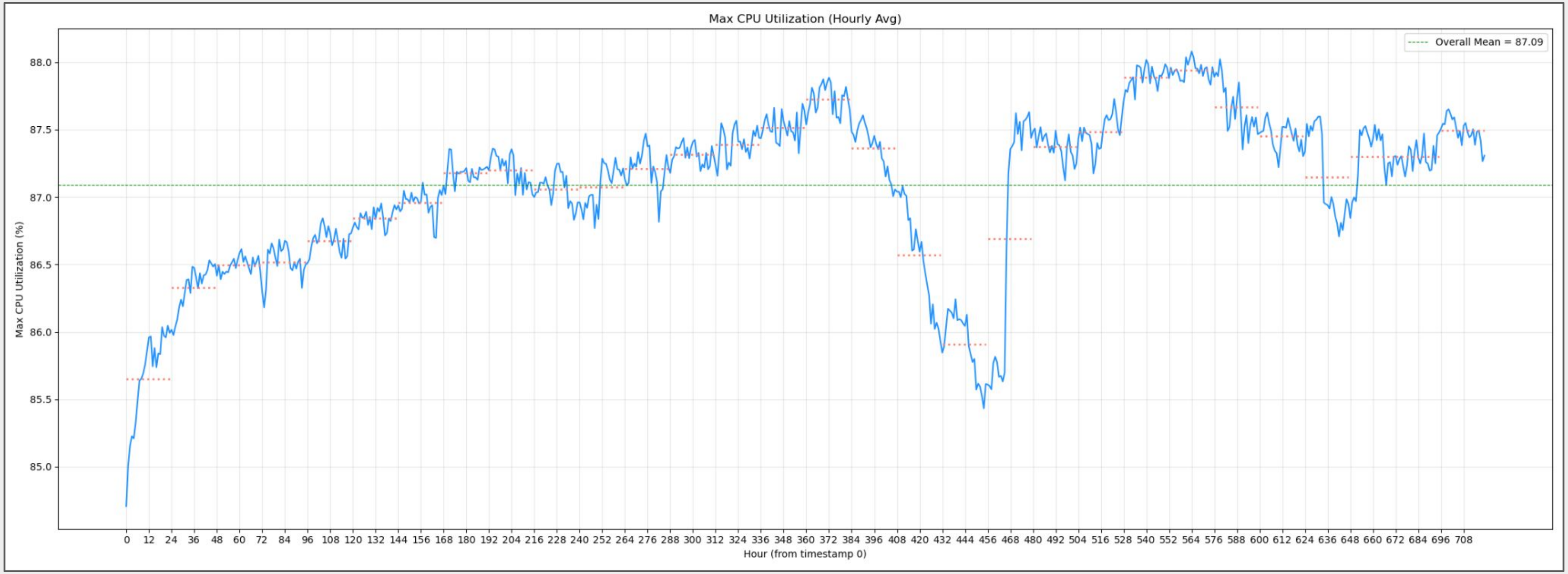
	vm_id	subscription_id	deployment_id
0	rKggHO/04j31UFy65mDTwtjdMQL/G03xWfl3xGeiilB4/W...	ub4ty8ygwOECrIz7eaZ/9hDwnCsERvZ3nJJ03sDSpD85et...	+ZralDUNaWYDZMBiBtZm7xSjr+j3zcHGjup1+wyKxHFmyJ...
1	YrR8gPtBmfNaOdnNEW5If1SdTqQgGQHEnLHGPjySt53bKW...	9LrdYRcUfGbmL2fFfLR/JUg2OTkjGRe3iluwlhDRPnPDPa...	GEylElfPSFupze8T+T1niQMepeqG88VpLNuxUMylDbz8VF...
2	xzQ++JF1UAkh70CDhmzkiOo+DQn+E2TLErCFKEmSswv1pl...	0XnZZ8sMN5HY+Yg+0dykYB5oenlgsrCpzpgFSvn/MX42Ze...	7aCQS6fPUw9rwCPiqvghk/WCEbMV3KgNJjA+sssdfY5Ybl...
3	vZEivnhabRmImDr+JqKqZnplM3WxtypwoxjfjnklR/idyR...	HUGaZ+piPP4eHjycCBki2yq0raJywdzrVuriR6nQceH3hA...	/s/D5VtTQDxyS6wq7N/VQAMczx61Ny1Ut3a3iFmDSOCXxp...
4	MqvcZ6Au5oul6if56MJHmoSqHtX8oRv0dPkaxCld3aUcr1...	p14cXGYqCKCcF7b7OdV6bdr/0gCim+u1LeqKoyEkyNNMWf...	ZFCk80slQzr43FUSqy2DOrcvBhuQkyfVz7gus8SORhyBxC...

vm_creation_timestamp	vm_deletion_timestamp	max_cpu	avg_cpu	p95_max_cpu	vm_category	vm_core_count	vm_memory
424500	425400	37.879261	3.325358	37.879261	Unknown	4	32
1133100	1133700	0.304368	0.220553	0.304368	Unknown	4	32
0	2591400	98.573424	30.340054	98.212503	Interactive	2	4
228300	229800	82.581449	13.876299	82.581449	Unknown	2	4
1395600	1397700	0.097875	0.035215	0.097875	Unknown	4	32

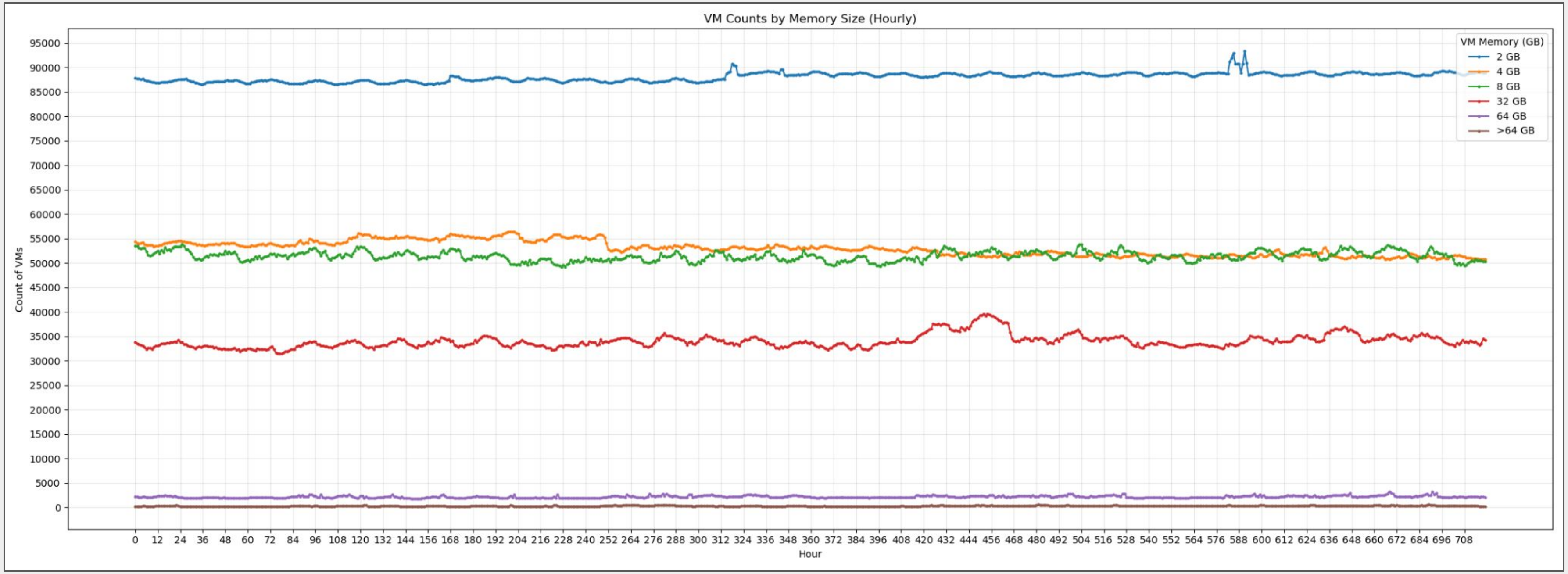
Azure Analysis - Avg CPU Utilization vs Allocation



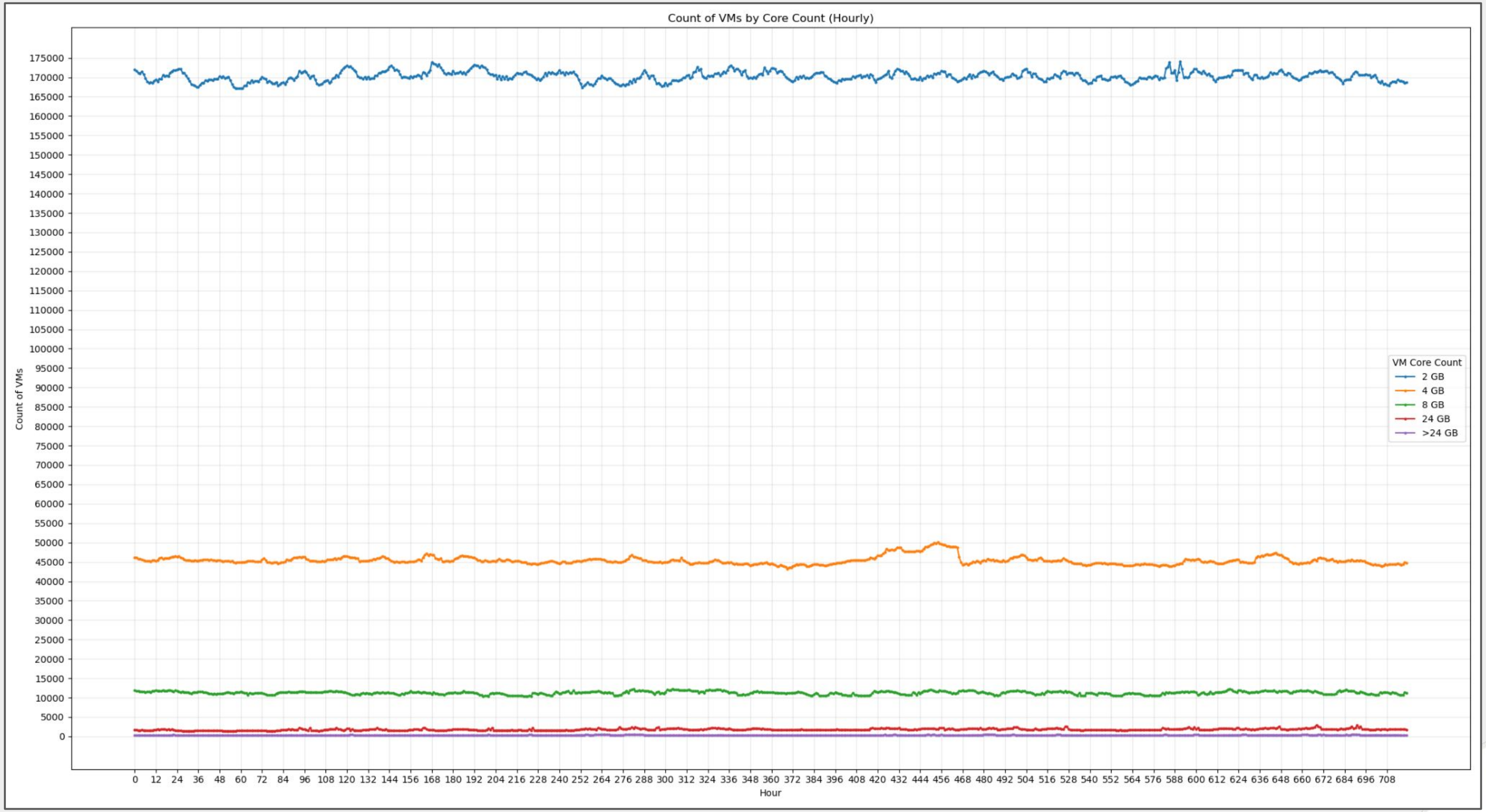
Azure Analysis - Max CPU Utilization vs Allocation



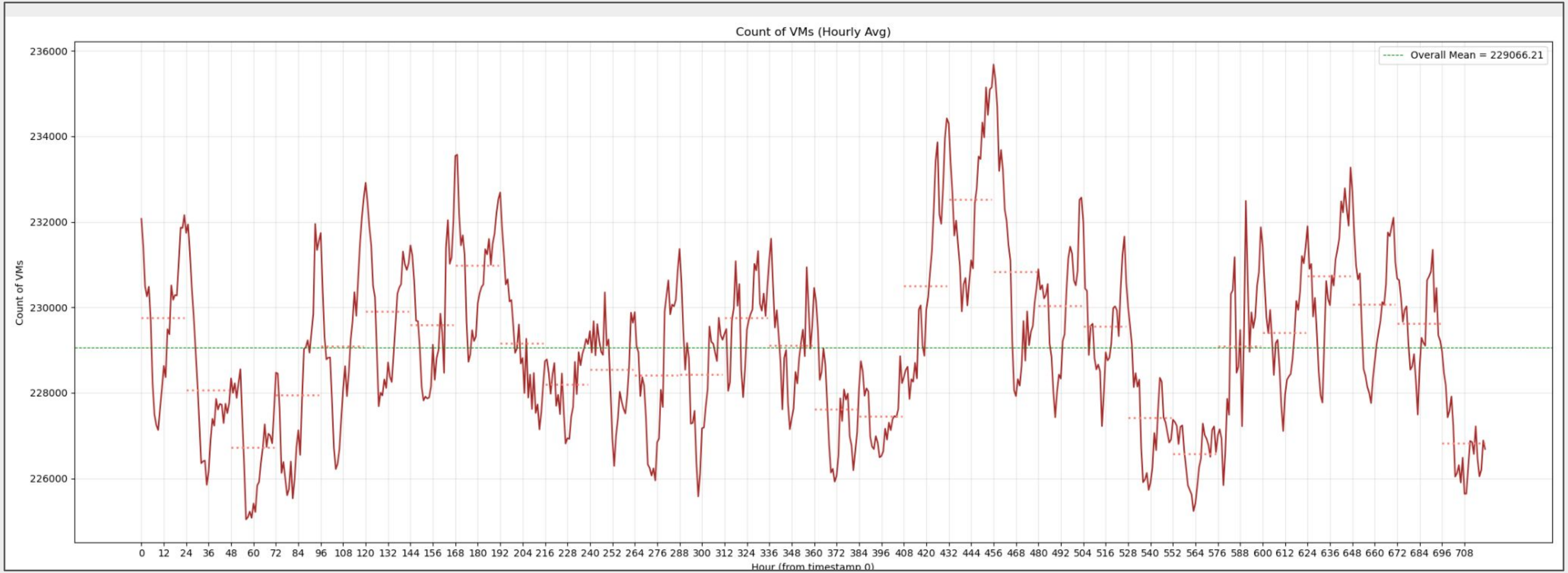
Azure Analysis - Number of VMs by Memory



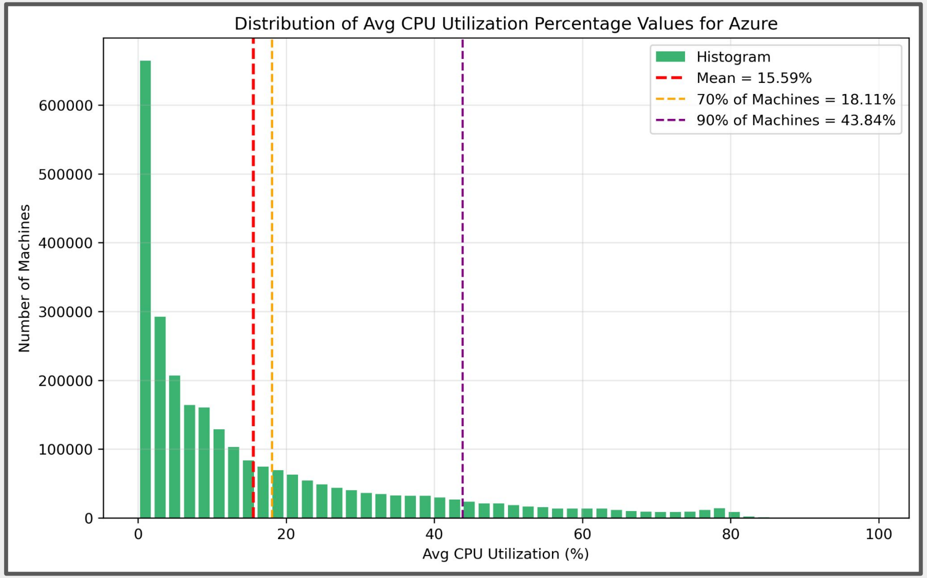
Azure Analysis - Number of VMs by Core Count

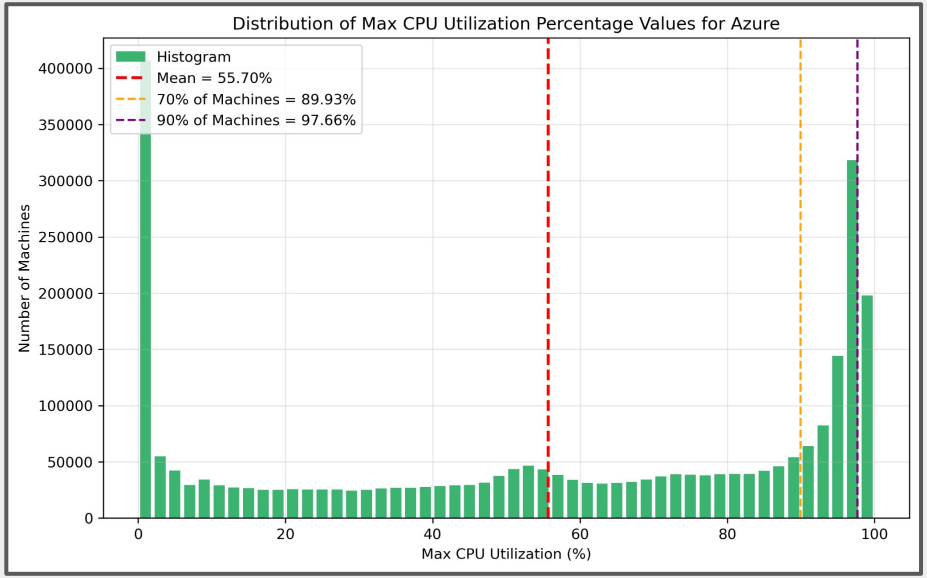


Azure Analysis - Number of VMs



Machine-Level Analysis - Azure Cluster Data (2019)





Predictive Analysis - Azure Cluster Data (2019)

- XGBoost Regression Machine Learning Model
  - 1500 Decision Trees
  - 95% Training Set
  - 5% Testing Set

- Training Data Variables
  - Start Hour (0 - 23)
  - Total Time
  - VM Core Count
  - VM Memory
  - VM Category

- CPU Utilization Evaluation Metrics
  - RMSE - 14.8718
  - Prediction Accuracy - 65.3044%

Methodology - Alibaba Cluster Data (2018)

- Traces from Alibaba clusters for 4000 machines.
- Traces spanning 8 days of data from 2018 (247 million rows).
- **Alibaba's 2018 Trace Data** with tables:
    - MachineMeta
    - MachineUsage
    - ContainerMeta
    - ContainerUsage
    - BatchTask
    - BatchInstance

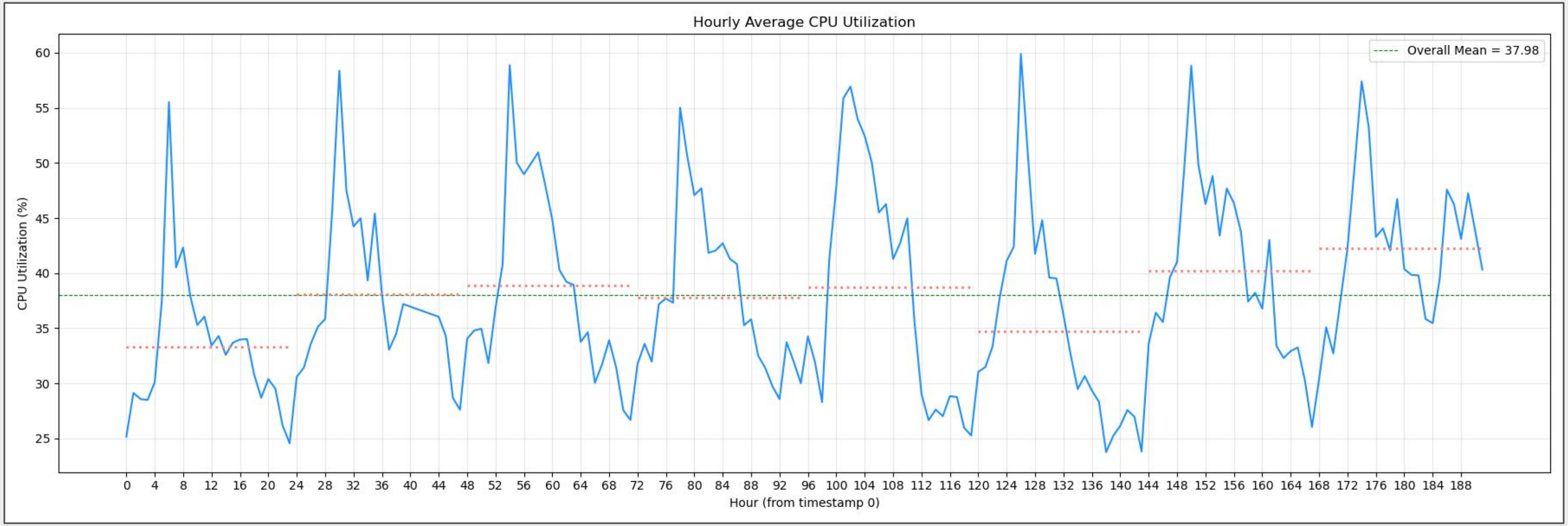
Methodology - Alibaba Cluster Data (2018)

instance_name	task_name	task_type	start_time_instance	end_time_instance	machine_id	cpu_avg	cpu_max	mem_avg	mem_max
ins_74901673	task_LTg0MTUwNTA5Mjg4MDkwNjIzMA==	10	673795	673797	m_2637	0.13	0.16	0.02	0.02
ins_815802872	M1	1	158478	158520	m_3430	0.03	0.19	0.13	0.18
ins_564677701	M1	1	372602	372616	m_1910	0.87	1.16	0.04	0.05
ins_257566161	M1	1	372602	372615	m_2485	0.91	1.23	0.05	0.05
ins_688679908	M1	1	372602	372615	m_993	0.93	1.41	0.05	0.05

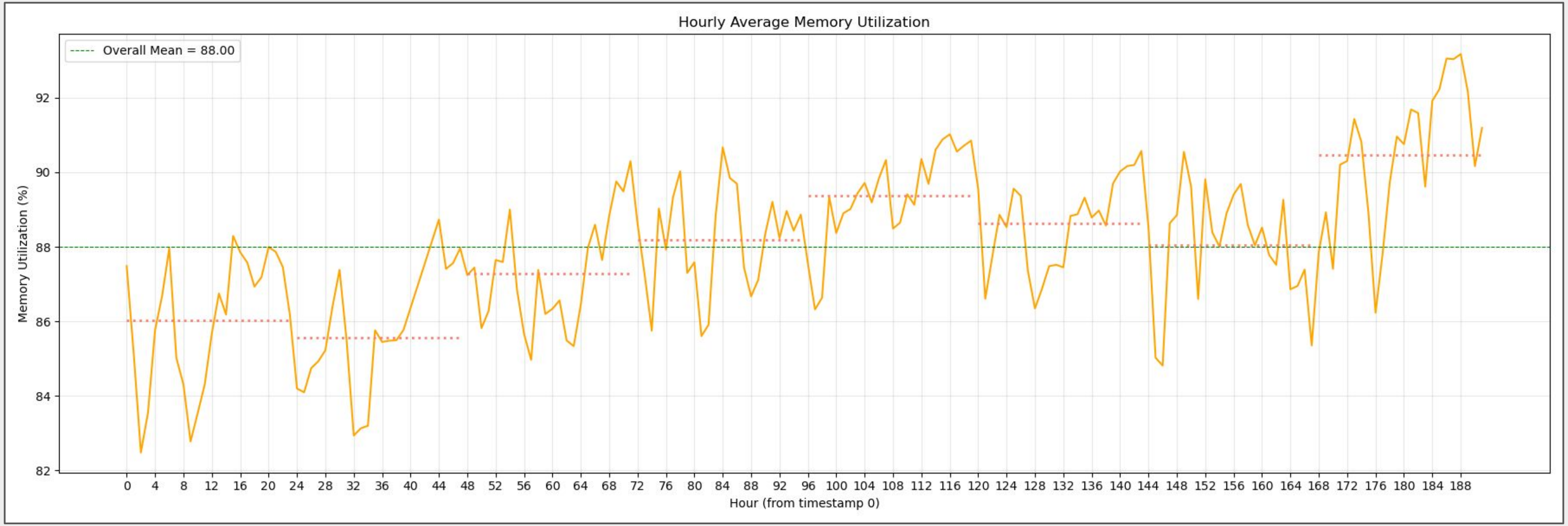
machine_id	time_stamp	cpu_util_percent	mem_util_percent	mem_gps	mkpi	net_in	net_out	disk_io_percent
m_425	0	47	89	NaN	NaN	34.90	28.60	3
m_626	0	20	90	NaN	NaN	37.23	32.58	5
m_3089	0	7	88	NaN	NaN	29.93	20.88	1
m_111	0	18	92	NaN	NaN	39.17	32.09	3
m_796	0	24	75	NaN	NaN	41.86	37.79	5

Georgia Tech

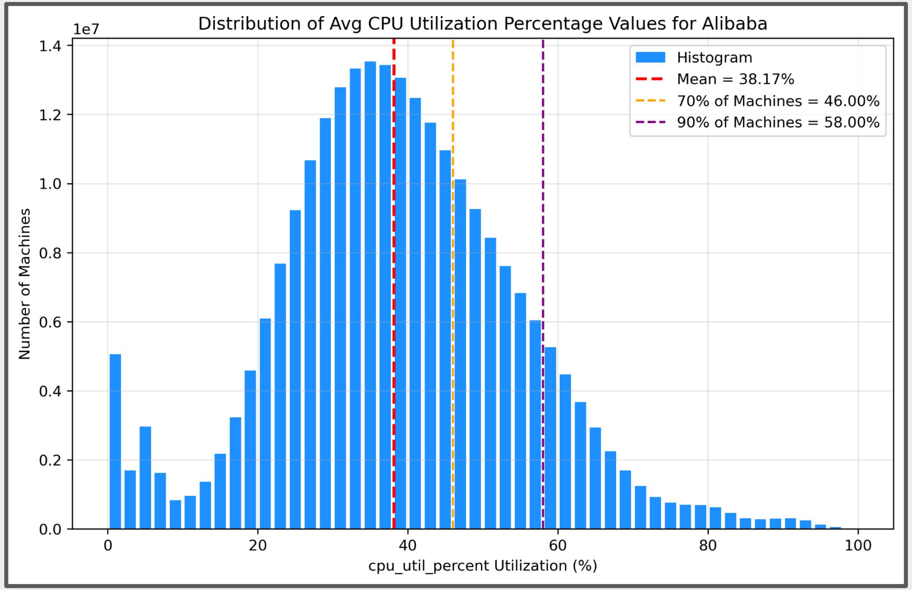
Alibaba Analysis - CPU Utilization vs Allocation

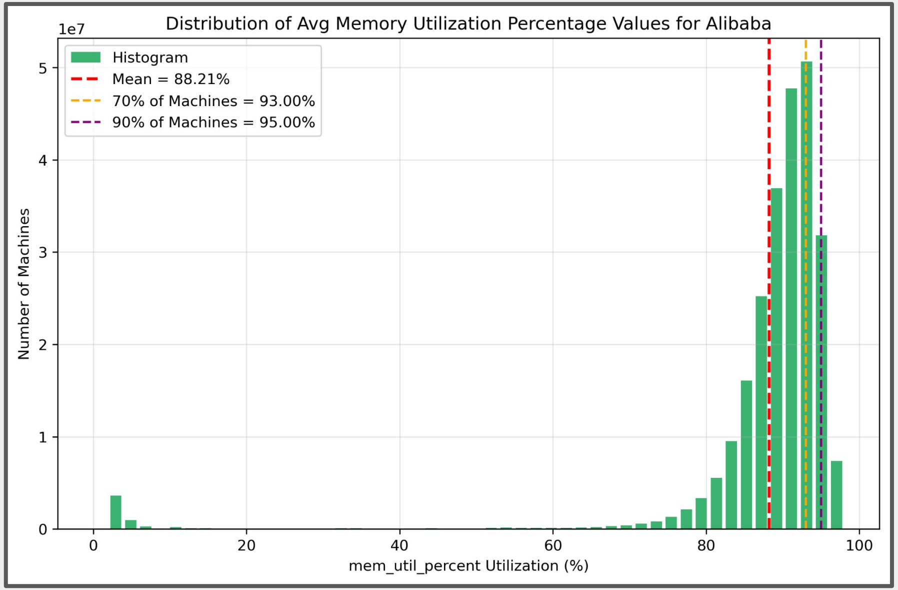


Alibaba Analysis - Memory Utilization vs Allocation

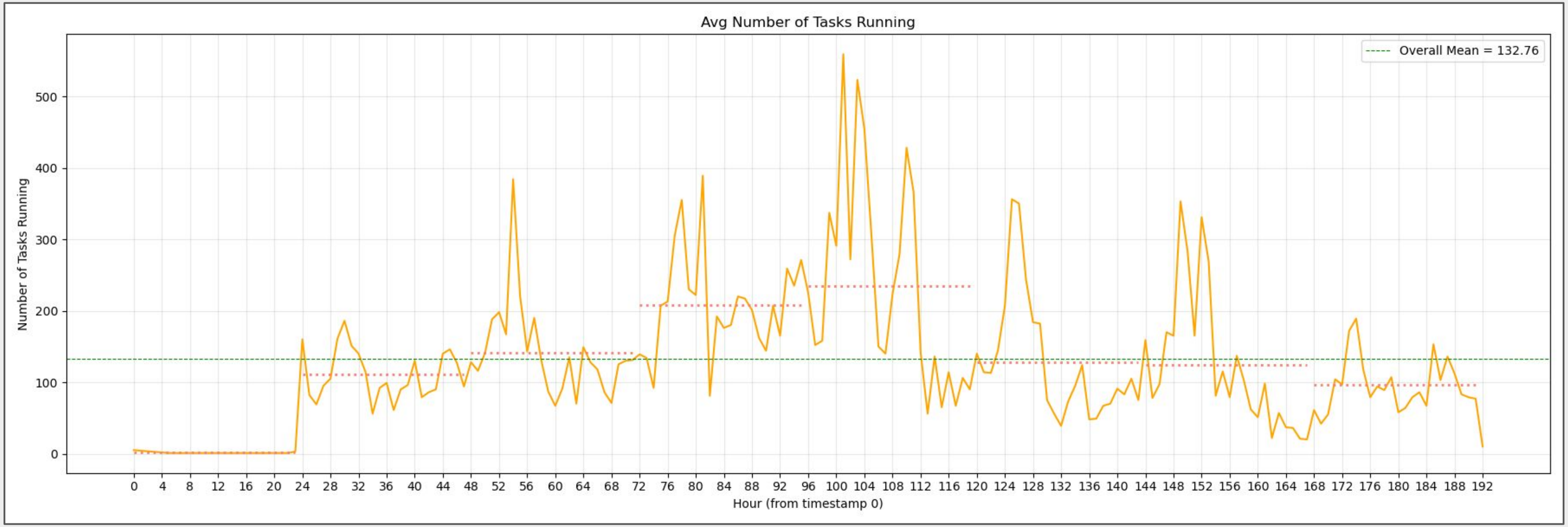


Machine-Level Analysis - Alibaba Cluster Data (2018)

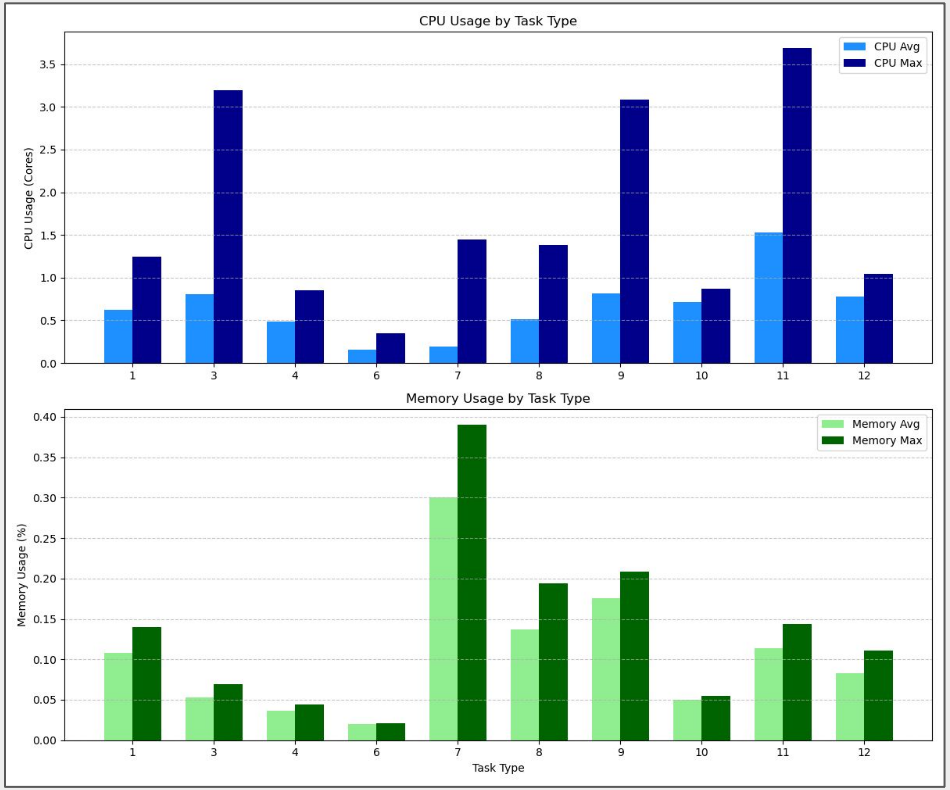




Alibaba Analysis - Tasks Running Per Hour



Alibaba Analysis - Usage by Task Type



Predictive Analysis - Alibaba Cluster Data (2018)

- XGBoost Regression Machine Learning Model
  - 1500 Decision Trees
  - 95% Training Set
  - 5% Testing Set

- Training Data Variables
  - Start Hour (0 - 23)
  - Total Time
  - Task Type

- CPU Utilization Evaluation Metrics
  - RMSE - 23.3734
  - Prediction Accuracy - 52.5361%

- Memory Utilization Evaluation Metrics
  - RMSE - 18.4607
  - Prediction Accuracy - 87.7181%

Conclusions

- **Google**
  - Average CPU utilization ~26% => Dynamic scheduling
  - Average Memory utilization ~44% Moderately utilized

- **Azure**
  - Average Mean CPU utilization ~16% => Moderately idle on-demand workloads
  - Average Maximum CPU ~ 56% Occasional usage spikes

- **Alibaba**
  - Average CPU utilization ~38% => Consistent, batch-heavy usage
  - Average Memory utilization ~88% High sustained memory pressure

Conclusions

- **Google**
  - CPU Utilization Prediction ~82%
  - Memory Utilization Prediction ~67%

- **Azure**
  - CPU Utilization Prediction ~65%

- **Alibaba**
  - CPU Utilization Prediction ~53%
  - Memory Utilization Prediction ~88%

Conclusions

Incorrect assumptions for datacenters developed due to isolated cloud provider analysis -

- **CPU utilization is low for majority consumers.**
  Azure (~16%), not Google (~26%), Alibaba (~38%) - wide variations exists in reality
- **Datacenters have significant memory that remains underutilized.**
  Google (~44%), not Alibaba (~88%), diverse.
- **Cloud workloads have random rare spikes.**
  Google => random, Azure => flat, idle, Alibaba => periodic.
- **Clouds use centralized scheduling.**
  Google => Borg, Azure => user-managed (no global coordination), Alibaba => dynamic
- **Cloud workloads behave similarly.**
  Each provider has distinct workload types and resource patterns.

THANK YOU!

QUESTIONS?