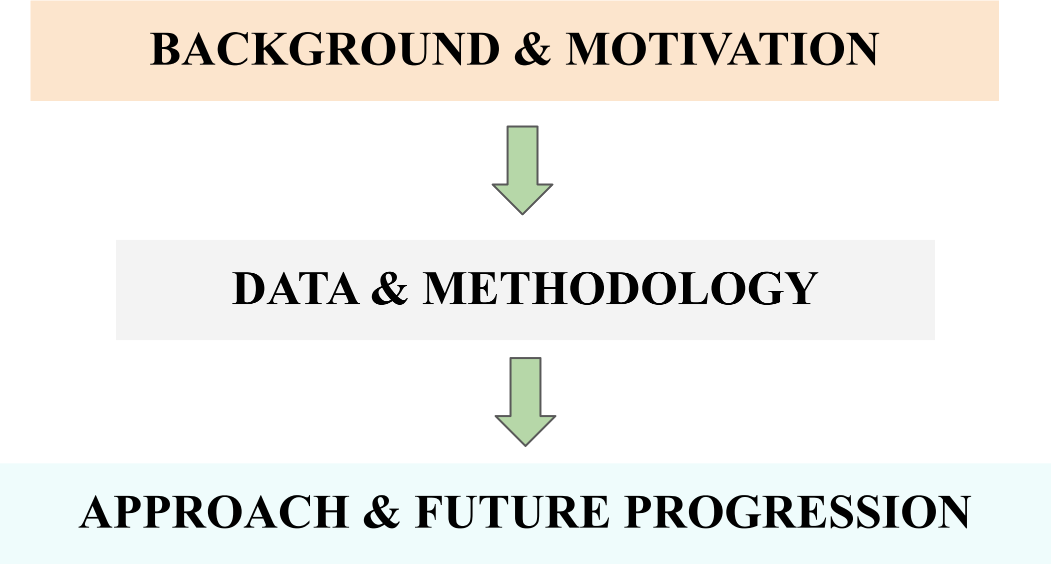
CROSS-PLATFORM ANALYSIS OF CLOUD RESOURCE UTILIZATION PATTERNS FOR OPTIMIZED RESOURCE ALLOCATION

CS 8803: Datacenter Networks & Systems (Spring 2025)
Akshat Karwa, Mehul Rastogi

Table of Contents



BACKGROUND & MOTIVATION

DATA & METHODOLOGY

APPROACH & FUTURE PROGRESSION

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 allocation strategies, and enhance load balancing techniques.
  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.

Methodology

Data Sources

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

We will perform data normalization across the different heterogeneous formats to get all the data in a similar format.

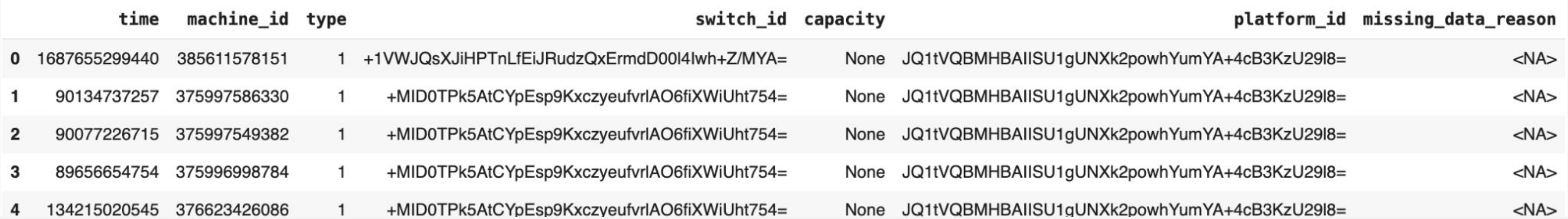
Methodology

Google Cluster Data (2019) -

- Contains the detailed usage traces from Google compute clusters that span several days of workload data from 2019.
- For each cluster, there are 8 different Google cells (a through h) across different time zones. Based on the Borg cluster management system. The data is further split into shards, where each shard has the following tables:
- **Core Tables:**
  - **MachineEvents**: Machine lifecycle and capacity information.
  - **MachineAttributes**: Machine properties and configurations.
  - **CollectionEvents**: Job and allocation set events.
  - **InstanceEvents**: Task and instance events.
  - **InstanceUsage**: Resource utilization measurements.

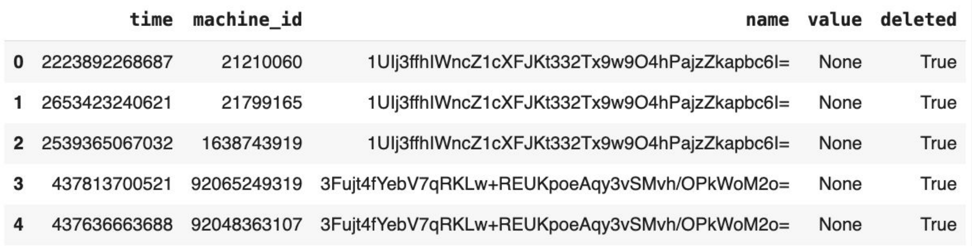
Data and Schema - Machine Events & Attributes (Google 2019)

MachineEvents



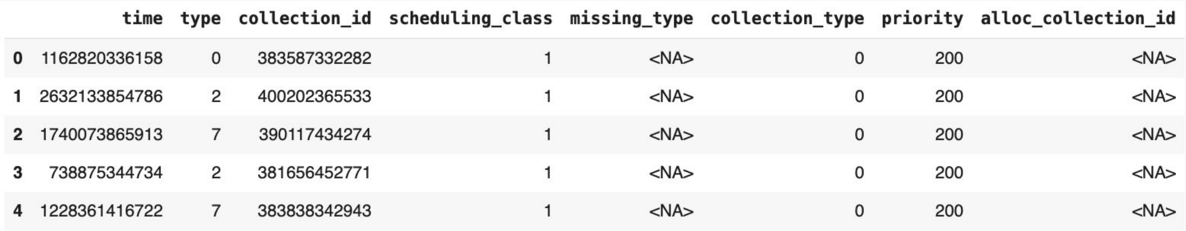
	time	machine_id	type	switch_id	capacity	platform_id	missing_data_reason
0	1687655299440	385611578151	1	+1VWJQsXJiHPTnLfEiJRudzQxErmdD00l4Iwh+Z/MYA=	None	JQ1tVQBMHBAIISU1gUNXk2powhYumYA+4cB3KzU29l8=	<NA>
1	90134737257	375997586330	1	+MID0TPk5AtCYpEsp9KxczyeufvrlAO6fiXWiUht754=	None	JQ1tVQBMHBAIISU1gUNXk2powhYumYA+4cB3KzU29l8=	<NA>
2	90077226715	375997549382	1	+MID0TPk5AtCYpEsp9KxczyeufvrlAO6fiXWiUht754=	None	JQ1tVQBMHBAIISU1gUNXk2powhYumYA+4cB3KzU29l8=	<NA>
3	89656654754	375996998784	1	+MID0TPk5AtCYpEsp9KxczyeufvrlAO6fiXWiUht754=	None	JQ1tVQBMHBAIISU1gUNXk2powhYumYA+4cB3KzU29l8=	<NA>
4	134215020545	376623426086	1	+MID0TPk5AtCYpEsp9KxczyeufvrlAO6fiXWiUht754=	None	JQ1tVQBMHBAIISU1gUNXk2powhYumYA+4cB3KzU29l8=	<NA>

MachineAttributes

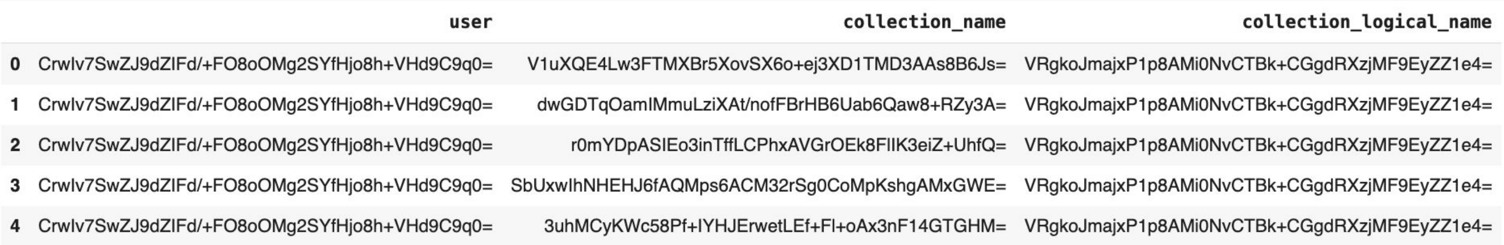


	time	machine_id	name	value	deleted
0	2223892268687	21210060	1UIj3ffhIWncZ1cXFJKt332Tx9w9O4hPajzZkapbc6I=	None	True
1	2653423240621	21799165	1UIj3ffhIWncZ1cXFJKt332Tx9w9O4hPajzZkapbc6I=	None	True
2	2539365067032	1638743919	1UIj3ffhIWncZ1cXFJKt332Tx9w9O4hPajzZkapbc6I=	None	True
3	437813700521	92065249319	3Fujt4fYebV7qRKLw+REUKpoeAqy3vSMvh/OPkWoM2o=	None	True
4	437636663688	92048363107	3Fujt4fYebV7qRKLw+REUKpoeAqy3vSMvh/OPkWoM2o=	None	True

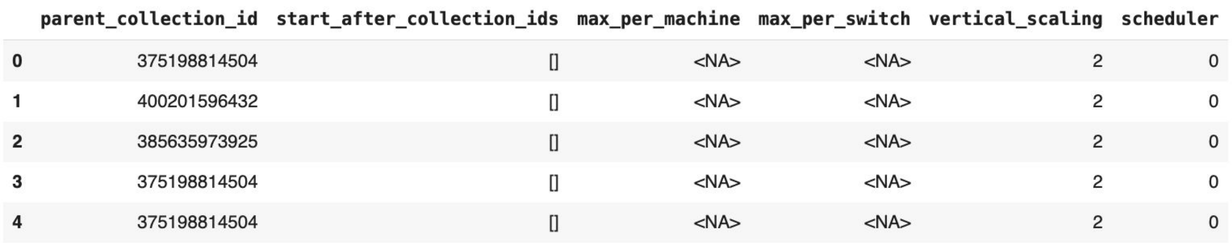
Data and Schema - Collection Events (Google 2019)



	time	type	collection_id	scheduling_class	missing_type	collection_type	priority	alloc_collection_id
0	1162820336158	0	383587332282	1	<NA>	0	200	<NA>
1	2632133854786	2	400202365533	1	<NA>	0	200	<NA>
2	1740073865913	7	390117434274	1	<NA>	0	200	<NA>
3	738875344734	2	381656452771	1	<NA>	0	200	<NA>
4	1228361416722	7	383838342943	1	<NA>	0	200	<NA>

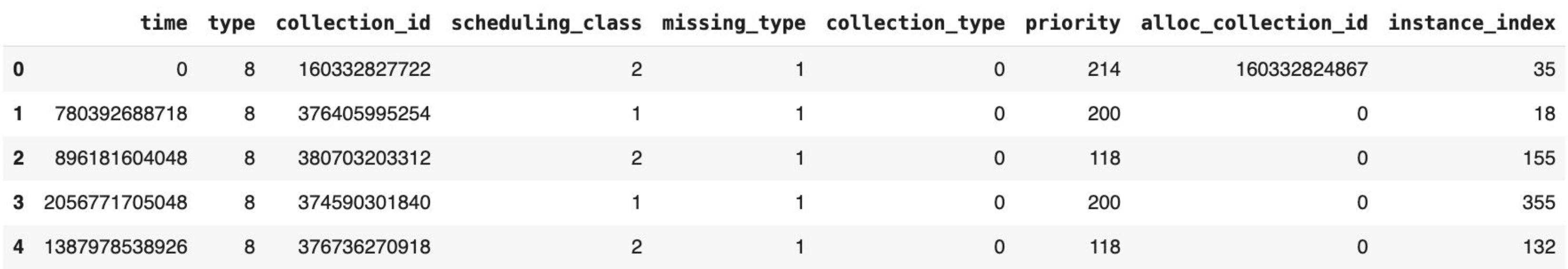


	user	collection_name	collection_logical_name
0	Crwlv7SwZJ9dZIFd/+FO8oOMg2SYfHjo8h+VHd9C9q0=	V1uXQE4Lw3FTMXBr5XovSX6o+ej3XD1TMD3AAs8B6Js=	VRgkoJmajxP1p8AMi0NvCTBk+CGgdRXzjMF9EyZZ1e4=
1	Crwlv7SwZJ9dZIFd/+FO8oOMg2SYfHjo8h+VHd9C9q0=	dwGDTqOamIMmuLziXAt/nofFBrHB6Uab6Qaw8+RZy3A=	VRgkoJmajxP1p8AMi0NvCTBk+CGgdRXzjMF9EyZZ1e4=
2	Crwlv7SwZJ9dZIFd/+FO8oOMg2SYfHjo8h+VHd9C9q0=	r0mYDpASIEo3inTffLCPhxAVGrOEk8FlIK3eiZ+UhfQ=	VRgkoJmajxP1p8AMi0NvCTBk+CGgdRXzjMF9EyZZ1e4=
3	Crwlv7SwZJ9dZIFd/+FO8oOMg2SYfHjo8h+VHd9C9q0=	SbUxwIhNHEHJ6fAQMps6ACM32rSg0CoMpKshgAMxGWE=	VRgkoJmajxP1p8AMi0NvCTBk+CGgdRXzjMF9EyZZ1e4=
4	Crwlv7SwZJ9dZIFd/+FO8oOMg2SYfHjo8h+VHd9C9q0=	3uhMCyKWc58Pf+IYHJErwetLEf+Fl+oAx3nF14GTGHM=	VRgkoJmajxP1p8AMi0NvCTBk+CGgdRXzjMF9EyZZ1e4=

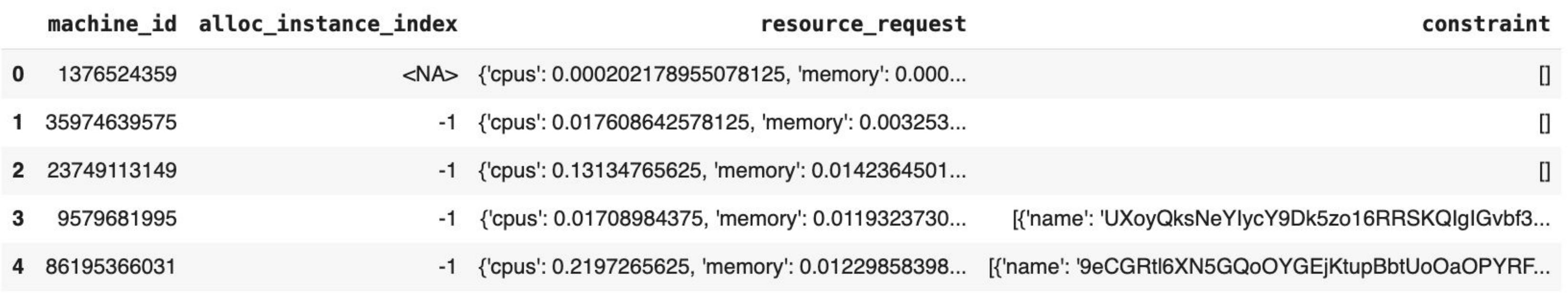


	parent_collection_id	start_after_collection_ids	max_per_machine	max_per_switch	vertical_scaling	scheduler
0	375198814504	[]	<NA>	<NA>	2	0
1	400201596432	[]	<NA>	<NA>	2	0
2	385635973925	[]	<NA>	<NA>	2	0
3	375198814504	[]	<NA>	<NA>	2	0
4	375198814504	[]	<NA>	<NA>	2	0

Data and Schema - Instance Events Data (Google 2019)

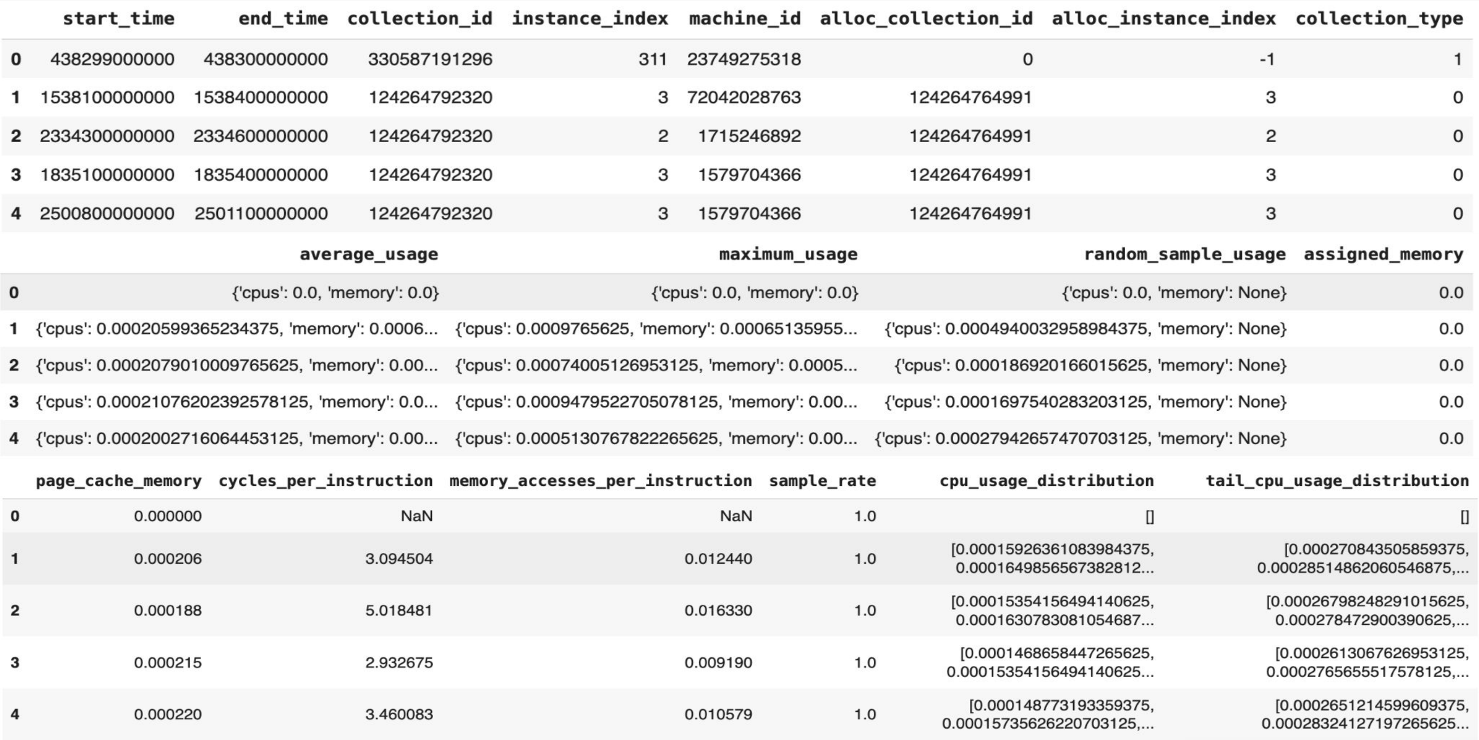


	time	type	collection_id	scheduling_class	missing_type	collection_type	priority	alloc_collection_id	instance_index
0	0	8	160332827722	2	1	0	214	160332824867	35
1	780392688718	8	376405995254	1	1	0	200	0	18
2	896181604048	8	380703203312	2	1	0	118	0	155
3	2056771705048	8	374590301840	1	1	0	200	0	355
4	1387978538926	8	376736270918	2	1	0	118	0	132



	machine_id	alloc_instance_index	resource_request	constraint
0	1376524359	<NA>	{'cpus': 0.000202178955078125, 'memory': 0.000...	[]
1	35974639575	-1	{'cpus': 0.017608642578125, 'memory': 0.003253...	[]
2	23749113149	-1	{'cpus': 0.13134765625, 'memory': 0.0142364501...	[]
3	9579681995	-1	{'cpus': 0.01708984375, 'memory': 0.0119323730...	[{'name': 'UXoyQksNeYlycY9Dk5zo16RRSKQlgIGvbf3...
4	86195366031	-1	{'cpus': 0.2197265625, 'memory': 0.01229858398...	[{'name': '9eCGRtl6XN5GQoOYGEjKtupBbtUoOaOPYRF...

Data and Schema - Instance Data Usage (Google 2019)



	start_time	end_time	collection_id	instance_index	machine_id	alloc_collection_id	alloc_instance_index	collection_type
0	438299000000	438300000000	330587191296	311	23749275318	0	-1	1
1	1538100000000	1538400000000	124264792320	3	72042028763	124264764991	3	0
2	2334300000000	2334600000000	124264792320	2	1715246892	124264764991	2	0
3	1835100000000	1835400000000	124264792320	3	1579704366	124264764991	3	0
4	2500800000000	2501100000000	124264792320	3	1579704366	124264764991	3	0

	average_usage	maximum_usage	random_sample_usage	assigned_memory
0	{'cpus': 0.0, 'memory': 0.0}	{'cpus': 0.0, 'memory': 0.0}	{'cpus': 0.0, 'memory': None}	0.0
1	{'cpus': 0.00020599365234375, 'memory': 0.0006...	{'cpus': 0.0009765625, 'memory': 0.00065135955...	{'cpus': 0.0004940032958984375, 'memory': None}	0.0
2	{'cpus': 0.0002079010009765625, 'memory': 0.00...	{'cpus': 0.00074005126953125, 'memory': 0.0005...	{'cpus': 0.00018692016601 5625, 'memory': None}	0.0
3	{'cpus': 0.0002107620239257812 5, 'memory': 0.0...	{'cpus': 0.0009479522705078125, 'memory': 0.00...	{'cpus': 0.0001697540283203125, 'memory': None}	0.0
4	{'cpus': 0.0002002716064453125, 'memory': 0.00...	{'cpus': 0.0005130767822265625, 'memory': 0.00...	{'cpus': 0.0002794265747070312 5, 'memory': None}	0.0

	page_cache_memory	cycles_per_instruction	memory_accesses_per_instruction	sample_rate	cpu_usage_distribution	tail_cpu_usage_distribution
0	0.000000	NaN	NaN	1.0	[]	[]
1	0.000206	3.094504	0.012440	1.0	[0.00015926361083984375, 0.0001649856567382812...	[0.000270843505859375, 0.0002851486206054687 5,...
2	0.000188	5.018481	0.016330	1.0	[0.00015354156494140625, 0.0001630783081054687...	[0.00026798248291015625, 0.000278472900390625,...
3	0.000215	2.932675	0.009190	1.0	[0.0001468658447265625, 0.00015354156494140625...	[0.0002613067626953125, 0.0002765655517578125,...
4	0.000220	3.460083	0.010579	1.0	[0.000148773193359375, 0.0001573562622070312 5,...	[0.000265121459960937 5, 0.0002832412719726562 5...

Methodology

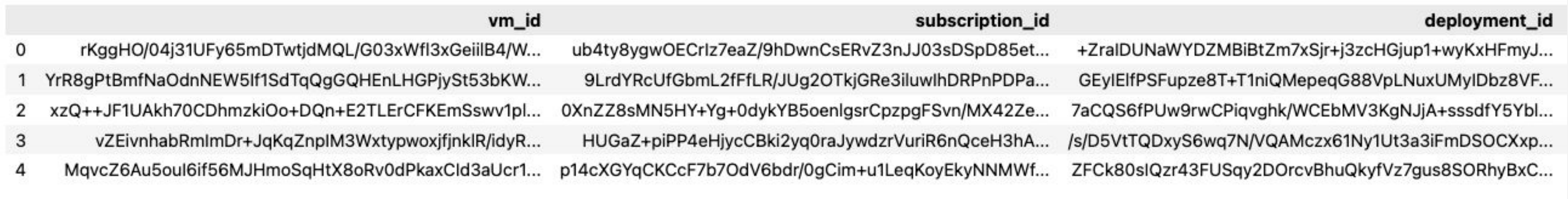
Azure Public Dataset (2019)

- Azure's public dataset contains robust trace data from Microsoft Azure for the year 2019.

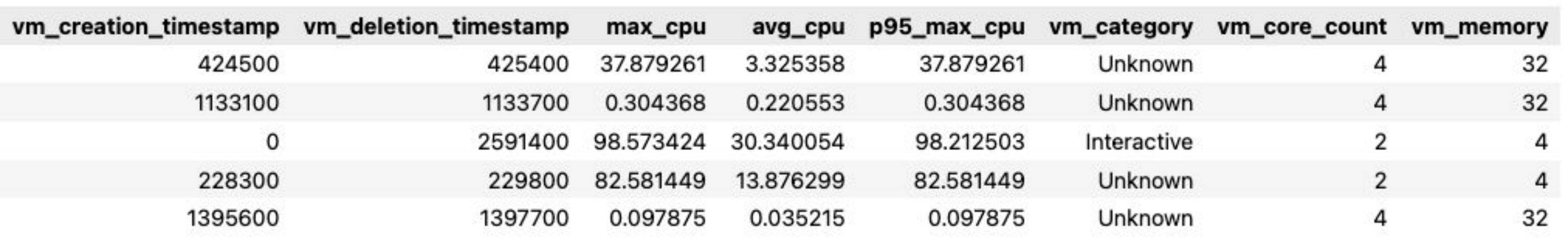
- **Core Tables:**
  - **VMTable** - Robust data for each Virtual Machine (VM)
  - **Deployments** - Deployment size for each Deployment ID
  - **Subscriptions** - VM data for each subscription ID
  - **VM_CPU_Readings** - CPU Readings for each VM ID

- The CPU readings are further divided into 195 shards - [1, 195].

Data and Schema - VMTable (Azure 2019)



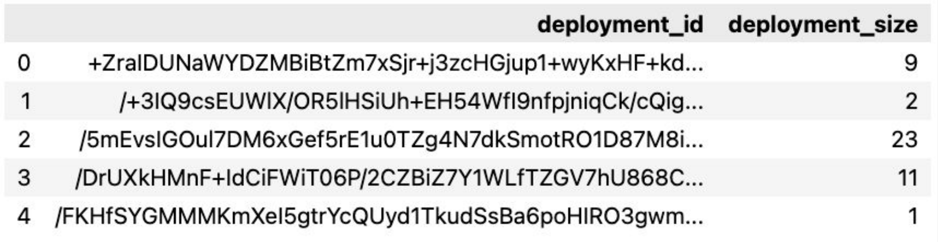
	vm_id	subscription_id	deployment_id
0	rKggHO/04j31UFy65mDTwtjdMQL/G03xWfl3xGeiilB4/W...	ub4ty8ygwOECrIz7eaZ/9hDwnCsERvZ3nJJ03sDSpD85et...	+ZraIDUNaWYDZMBiBtZm7xSjr+j3zcHGjup1+wyKxHFmyJ...
1	YrR8gPtBmfNaOdnNEW5If1SdTqQgGQHEnLHGPjySt53bKW...	9LrdYRcUfGbmL2fFfLR/JUg2OTkjGRe3iluwIhDRPnPDPa...	GEylElfPSFupze8T+T1niQMepeqG88VpLNuxUMyIDbz8VF...
2	xzQ++JF1UAkh70CDhmzkiOo+DQn+E2TLErCFKEmSswv1pl...	0XnZZ8sMN5HY+Yg+0dykYB5oenlgsrCpzpgFSvn/MX42Ze...	7aCQS6fPUw9rwCPiqvghk/WCEbMV3KgNJjA+sssdfY5Ybl...
3	vZEivnhabRmlmDr+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

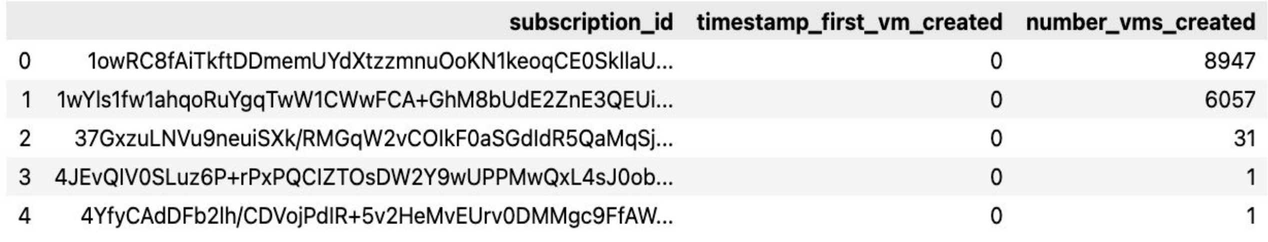
Data and Schema - (Azure 2019)

Deployments



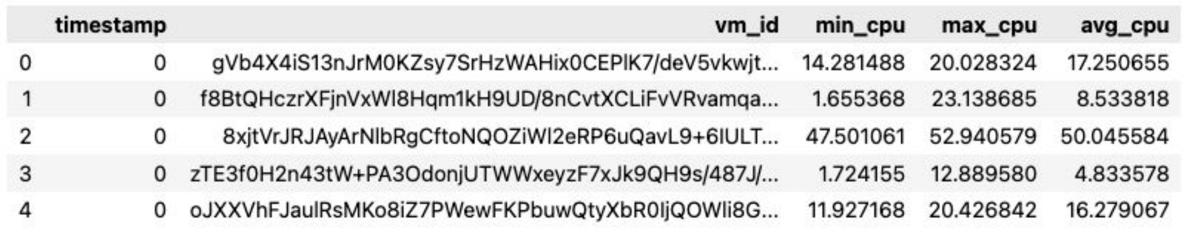
	deployment_id	deployment_size
0	+ZralDUNaWYDZMBiBtZm7xSjr+j3zcHGjup1+wyKxHF+kd...	9
1	/+3IQ9csEUWlX/OR5lHSiUh+EH54WfI9nfpjniqCk/cQig...	2
2	/5mEvsIGOul7DM6xGef5rE1u0TZg4N7dkSmotRO1D87M8i...	23
3	/DrUXkHMnF+IdCiFWiT06P/2CZBiZ7Y1WLfTZGV7hU868C...	11
4	/FKHfSYGMMMKmXel5gtrYcQUyd1TkudSsBa6poHIRO3gwm...	1

Subscriptions



	subscription_id	timestamp_first_vm_created	number_vms_created
0	1owRC8fAiTkftDDmemUYdXtzzmnuOoKN1keoqCE0SkllaU...	0	8947
1	1wYls1fw1ahqoRuYgqTwW1CWwFCA+GhM8bUdE2ZnE3QEUi...	0	6057
2	37GxzuLNVu9neuiSXk/RMGqW2vCOIkF0aSGdldR5QaMqSj...	0	31
3	4JEvQIV0SLuz6P+rPxPQCIZTOsDW2Y9wUPPMwQxL4sJ0ob...	0	1
4	4YfyCAdDFb2lh/CDVojPdIR+5v2HeMvEUrv0DMMgc9FfAW...	0	1

VM CPU Readings



	timestamp	vm_id	min_cpu	max_cpu	avg_cpu
0	0	gVb4X4iS13nJrM0KZsy7SrHzWAHix0CEPlK7/deV5vkwjt...	14.281488	20.028324	17.250655
1	0	f8BtQHczrXFjnVxWl8Hqm1kH9UD/8nCvtXCLiFvVRvamqa...	1.655368	23.138685	8.533818
2	0	8xjtVrJRJAyArNlbRgCftoNQOZiWl2eRP6uQavL9+6IULT...	47.501061	52.940579	50.045584
3	0	zTE3f0H2n43tW+PA3OdonjUTWWxeyzF7xJk9QH9s/487J/...	1.724155	12.889580	4.833578
4	0	oJXXVhFJaulRsMKo8iZ7PWewFKPbuwQtyXbR0IjQOWli8G...	11.927168	20.426842	16.279067

Methodology

Alibaba's 2018 Trace Data with tables: MachineMeta, MachineUsage, ContainerMeta, ContainerUsage, BatchTask, and BatchInstance represent hierarchical operational data from infrastructure to workload execution.

Machine Data: machine_id, timestamp, status, cpu_num, mem_size, CPU util%, Memory util%, Network I/O, Disk I/O%

Container Data: container_id, machine_id, app_du, status, cpu_request/limit, memory_size, CPU/Memory util%, network traffic, disk I/O

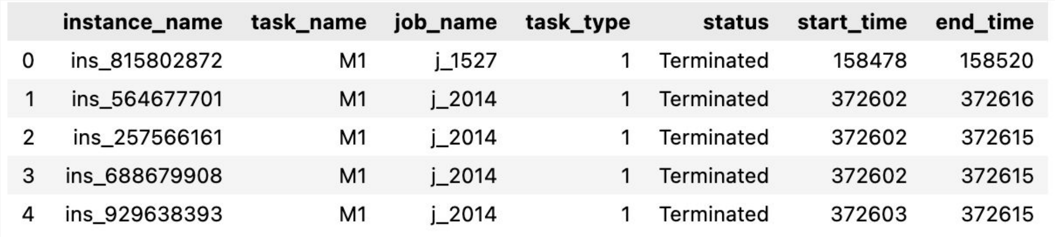
Task Data: task_name, job_name, task_type, status, instance_num, start/end time, number of cpu needed, normalized memory size

Instance Data: instance_name, task_name, machine_id, seq_no, cpu_avg, cpu_max, mem_avg, mem_max

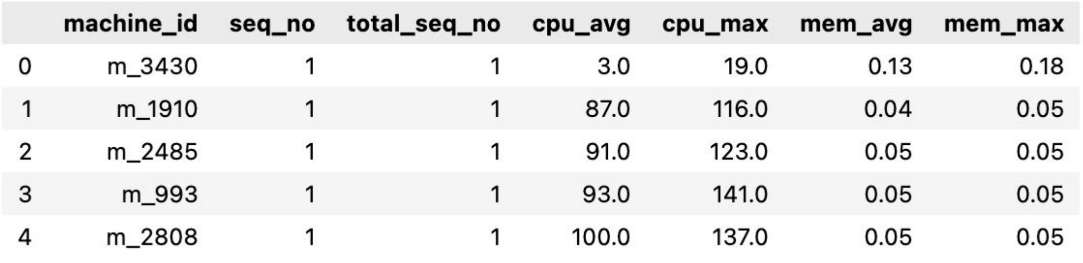
Data and Schema - (Alibaba 2018)

The main data table that we will be using for our analysis is:

Batch Instance Data Table



	instance_name	task_name	job_name	task_type	status	start_time	end_time
0	ins_815802872	M1	j_1527	1	Terminated	158478	158520
1	ins_564677701	M1	j_2014	1	Terminated	372602	372616
2	ins_257566161	M1	j_2014	1	Terminated	372602	372615
3	ins_688679908	M1	j_2014	1	Terminated	372602	372615
4	ins_929638393	M1	j_2014	1	Terminated	372603	372615



	machine_id	seq_no	total_seq_no	cpu_avg	cpu_max	mem_avg	mem_max
0	m_3430	1	1	3.0	19.0	0.13	0.18
1	m_1910	1	1	87.0	116.0	0.04	0.05
2	m_2485	1	1	91.0	123.0	0.05	0.05
3	m_993	1	1	93.0	141.0	0.05	0.05
4	m_2808	1	1	100.0	137.0	0.05	0.05

Future Progression - Approach

- We will start by understanding trends in data of individual cloud providers and then compare their resource allocation techniques. Then, we will understand resource utilization based on demands and contrast all three cloud providers, especially exploring the metrics related to CPU and Memory.

- After simulating and understanding the traces from each of the three datasets, we will simulate trends in the time series to gain actionable insights into decisions taken by cloud providers.

- Then, we will examine variable correlations and identify essential data components through multivariate analysis which will help us in revealing the key relationships between resource utilization, workload behavior, and performance.

Future Progression - Predictive Modelling

- We will be utilizing machine learning and reinforcement learning approaches to make predictions and comparing them with the truth data from the time-series. This would enable us to understand if trends in demand can be accurately forecasted using analysis of historical data to improve allocation of resources. Furthermore, analysing how cloud providers made these allocations in comparison to model predictions of allocation would allow us to evaluate their efficiency.

- The Input Variables and Parameters for these models would be:
    - Task Priority (Google 2019)
    - Allocated Resources
        - CPU & Memory
    - Task Type and Machine ID (Alibaba 2018)

- Using these input parameters, the goal of the model would be to predict future resource demands, workload surges, and job completion times. Furthermore, it is challenging but we will also try to predict task resource requirements and potential failures.

Future Progression - Timeline

Our plan is to complete the project in 4 sprints, each approximately 2 weeks long.

- 24th February - 10th March
  - In-depth data exploration and analysis on all 3 datasets in isolation.
    - Find strategies for optimizing resource utilization in isolation.
  - Start to normalize the three datasets to bring them to a standardized design schema.

- 11th March - 21st March **(Progress Report)**
  - Complete the normalization and Perform deep exploration and analysis.
  - Start to build a data pipeline so that we can work with the *Entire Dataset* from the 3 sources.

- 22nd March - 8th April
  - Perform analyses and experiments to find trends and generalizable insights from the normalized data in the data pipeline.
  - Run automated simulations on the normalized trace data from different cloud providers to recognize patterns and generalizable trends.

- 10th April - 21st April **(Final Report and Presentation)**
  - Find and observe generalizable strategies that can be applied on the trace data in order to improvise resource utilization and reduce costs.
  - Run ML/RL models on small chunks of the dataset to perform predictive modelling.

THANK YOU!

QUESTIONS?