UNIVERSITY OF OXFORD

JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

KAUST

Computationally Budgeted Continual Learning
What Does Matter?

TUE-AM-352







Ameya Prabhu* (Oxford) | **Hasan Hammoud* (KAUST)** | Puneet Dokania (Oxford) | Philip Torr (Oxford) | Sernam Lim (Meta) | Bernard Ghanem (KAUST) | Adel Bibi (Oxford)

* equal contribution

The Memory Dilemma

Most prior art restrict memory because:

Cost

Privacy

The Memory Dilemma

Does restricting the memory really address cost concerns?

Ref.	Dataset	Ordering	Memory	Cost	Iters	Cost
[9]	CIFAR10	Cls Inc	1-25MB	0.05¢	250K-375K	20$
[44]	CIFAR100	Cls Inc	10 MB	0.02¢	50K	8$
[19, 25]					125K	15$
[9]	TinyImageNet	Cls Inc	5-20 MB	0.04¢	350K-500K	25$
[19, 25]	ImageNet100	Cls Inc	0.3-1 GB	2¢	100K	50$
[19, 25]	ImageNet1K	Cls Inc	33GB	66¢	1M	500$
[29]	CLEAR	Dist Shift	0.4-1.2GB	2¢	300K	100$
[24]	ResNet50 (bs=256)		22GB			
Ours	CGLM	Dist Shift	**90GB**	2$	2K	10$
	ImageNet2K	ClsInc, DataInc	**400GB**	10$	8K	35$

Cost of Compute >> Cost of Storage

The Privacy Dilemma

Does restricting the memory size really address privacy concerns?

Model Inversion

Original Image

Deep Networks ***memorize*** information

Privacy needs Forgetting

Incompatible with Continual Learning

Fredrikson, Matt et al. "Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures." *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (2015): n. pag.

Key Contributions

Our Key Contribution: A New Setup

Reduce computational costs

→ Storage is (virtually) free

But

→ GPUs are expensive

Key Contributions

Dir.	Reference	Applicability (our setup)	Distillation	MemUpdate	MemRetrieve	FC Correction	Others
	Naive	✓	-	Random	Random	-	-
Distillation	iCARL [44]	✓	**BCE**	**Herding**	Random	-	**NCM**
	LUCIR [25]	✓	**Cosine**	Herding	MargRank	**CosFC**	NCM
	PODNet [19]	✓	**POD**	Herding	Random	**LSC**	Imprint,NCM
	DER [9]	✓	**MSE**	Reservoir	Random	-	-
	CO^2L [12]	×	**IRD**	Random	Random	**Asym.SupCon**	-
	SCR [35]	✓	-	Reservoir	Random	**SupCon**	NCM
Sampling	TinyER [15]	✓	-	**FIFO,KMeans,Reservoir**	-	-	-
	GSS [5]	×	-	**GSS**	Random	-	-
	MIR [3]	×	-	Reservoir	**MIR**	-	-
	GDumb [40]	✓	-	**Balanced**	Random	-	**MemOnly**
	Mnemonics [31]	×	-	**Mnemonics**	-	-	BalFineTune
	OCS [57]	×	-	**OCS**	Random	-	-
	InfoRS [49]	×	MSE	**InfoRS**	Random	-	-
	RMM [30]	×	-	**RMM**	-	-	-
	ASER [48]	×	-	**SV**	**ASV**	-	-
	RM [6]	✓	-	**Uncertainty**	Random	-	AutoDA
	CLIB [27]	×	-	**Max Loss**	Random	-	MemOnly,AdaLR
FC Layer	BiC [53]	×	CrossEnt	Random	Random	**BiC**	-
	WA [60]	×	CrossEnt	Random	Random	**WA**	-
	SS-IL [2]	×	TKD	Random	Balanced	**SS**	-
	CoPE [17]	✓	-	Balanced	Random	**PPPLoss**	-
	ACE [10]	✓	-	Reservoir	Random	**ACE**	-

Three Principal Directions

Reduce effect of distribution shift from past data by:

- **Distillation:** Enforce Output Similarity with Old Models
- **Sampling Old Data:** Create representative coreset of past knowledge
- **Correcting FC Layer:** Posits knowledge in representations far less affected, but classifier gets worse

Key Contributions

Dir.	Reference	Applicability (our setup)	Components				
			Distillation	MemUpdate	MemRetrieve	FC Correction	Others
	Naive	✓	-	Random	Random	-	-
Distillation	iCARL [44]	✓	**BCE**	**Herding**	Random	-	**NCM**
	LUCIR [25]	✓	**Cosine**	Herding	MargRank	**CosFC**	NCM
	PODNet [19]	✓	**POD**	Herding	Random	**LSC**	Imprint,NCM
	DER [9]	✓	**MSE**	Reservoir	Random	-	-
	CO^2L [12]	×	**IRD**	Random	Random	**Asym.SupCon**	-
	SCR [35]	✓	-	Reservoir	Random	**SupCon**	NCM
Sampling	TinyER [15]	✓	-	**FIFO,KMeans,Reservoir**	-	-	-
	GSS [5]	×	-	**GSS**	Random	-	-
	MIR [3]	×	-	Reservoir	**MIR**	-	-
	GDumb [40]	✓	-	**Balanced**	Random	-	**MemOnly**
	Mnemonics [31]	×	-	**Mnemonics**	-	-	BalFineTune
	OCS [57]	×	-	**OCS**	Random	-	-
	InfoRS [49]	×	MSE	**InfoRS**	Random	-	-
	RMM [30]	×	-	**RMM**	-	-	-
	ASER [48]	×	-	**SV**	**ASV**	-	-
	RM [6]	✓	-	**Uncertainty**	Random	-	AutoDA
	CLIB [27]	×	-	**Max Loss**	Random	-	MemOnly,AdaLR
FC Layer	BiC [53]	×	CrossEnt	Random	Random	**BiC**	-
	WA [60]	×	CrossEnt	Random	Random	**WA**	-
	SS-IL [2]	×	TKD	Random	Balanced	**SS**	-
	CoPE [17]	✓	-	Balanced	Random	**PPPLoss**	-
	ACE [10]	✓	-	Reservoir	Random	**ACE**	-

Three Principal Directions

Reduce effect of distribution shift from past data by:

- **Distillation:** Enforce Output Similarity with Old Models
- **Sampling Old Data:** Create representative coreset of past knowledge
- **Correcting FC Layer:** Posits knowledge in representations far less affected, but classifier gets worse

Conclusions

Testing across streams with new classes, new data & across time says.

I. **All three** algorithmic directions **fail**
  I. when computational costs **are equalized**

II. **Baseline: Sample class-balanced subset and train using all budget.**
  I. Best across major past directions

III. Conclusion **consistent** across:
  I. Varying computational budgets
  II. Varying stream sizes and timesteps

JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

Computationally Budgeted Continual Learning
What Does Matter?

TUE-AM-352

Ameya Prabhu*
(Oxford)

Hasan Hammoud*
(KAUST)

Puneet Dokania
(Oxford)

Philip Torr
(Oxford)

Sernam Lim
(Meta)

Bernard Ghanem
(KAUST)

Adel Bibi
(Oxford)

* equal contribution

Continual Learning: A Recap

Learn a function $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially

over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

Continual Learning: A Recap

Learn a function $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially

over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

1. $\mathcal{S}$ reveals a set of image-label pairs $\{(x_i^t, y_i^t)\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$

Continual Learning: A Recap

Learn a function $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially

over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

$\mathcal{D}_{j \leq t}$ denotes a varying distribution

1. $\mathcal{S}$ reveals a set of image-label pairs $\{(x_i^t, y_i^t)\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$

Continual Learning: A Recap

Learn a function $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially

over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

1. $\mathcal{S}$ reveals a set of image-label pairs $\{(x_i^t, y_i^t)\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$
2. Memory is updated to $\mathcal{T}_t = \cup_{r=1}^{t} \{(x_i^r, y_i^r)\}_{i=1}^{n_r}$

Continual Learning: A Recap

Learn a function $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially

over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

1. $\mathcal{S}$ reveals a set of image-label pairs $\{(x_i^t, y_i^t)\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$
2. Memory is updated to $\mathcal{T}_t = \cup_{r=1}^{t} \{(x_i^r, y_i^r)\}_{i=1}^{n_r}$
3. Continual learner updates θ_t to θ_{t+1}

Continual Learning: A Recap

Learn a function $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially

over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

Prior art limits access to past memory

1. $\mathcal{S}$ reveals a set of image-label pairs $\{(x_i^t, y_i^t)\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$

2. Memory is updated to $\mathcal{T}_t = \cup_{r=1}^{t} \{(x_i^r, y_i^r)\}_{i=1}^{n_r}$

3. Continual learner updates θ_t to θ_{t+1}

The Memory Dilemma

Most prior art restrict memory because:

Cost

Privacy

The Memory Dilemma: Cost

Does restricting the memory really address cost concerns?

Ref.	Dataset	Ordering	Memory	Cost	Iters	Cost
[9]	CIFAR10	Cls Inc	1-25MB	0.05¢	250K-375K	20$
[44]	CIFAR100	Cls Inc	10 MB	0.02¢	50K	8$
[19, 25]					125K	15$
[9]	TinyImageNet	Cls Inc	5-20 MB	0.04¢	350K-500K	25$
[19, 25]	ImageNet100	Cls Inc	0.3-1 GB	2¢	100K	50$
[19, 25]	ImageNet1K	Cls Inc	33GB	66¢	1M	500$
[29]	CLEAR	Dist Shift	0.4-1.2GB	2¢	300K	100$
[24]	ResNet50 (bs=256)		22GB			
Ours	CGLM	Dist Shift	**90GB**	2$	2K	10$
	ImageNet2K	ClsInc, DataInc	**400GB**	10$	8K	35$

Cost of Compute >> Cost of Storage

The Memory Dilemma: Privacy

Does restricting the memory size really address privacy concerns?

Model Inversion

Original Image

Deep Networks ***memorize*** information

Privacy needs Forgetting

Incompatible with Continual Learning

Continual Learning: A Recap

Learn a function $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially

over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

Prior art limits access to past memory

1. $\mathcal{S}$ reveals a set of image-label pairs $\{(x_i^t, y_i^t)\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$

2. Memory is updated to $\mathcal{T}_t = \cup_{r=1}^{t} \{(x_i^r, y_i^r)\}_{i=1}^{n_r}$

3. Continual learner updates θ_t to θ_{t+1}

Our Proposal: Budgeted Continual Learning

Learn a function $f_\theta : \mathcal{X} \to \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially

over steps $t \in \{1, 2, \ldots, \infty\}$ where at every step:

Restrict compute

1. $\mathcal{S}$ reveals a set of image-label pairs $\{(x_i^t, y_i^t)\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$
2. Memory is updated to $\mathcal{T}_t = \cup_{r=1}^{t} \{(x_i^r, y_i^r)\}_{i=1}^{n_r}$
3. Continual learner updates θ_t to θ_{t+1} spending a computational budget $\mathcal{C}_t$

Our Proposal: Budgeted Continual Learning

Learn a function $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ from a stream $\mathcal{S}$ revealing data sequentially

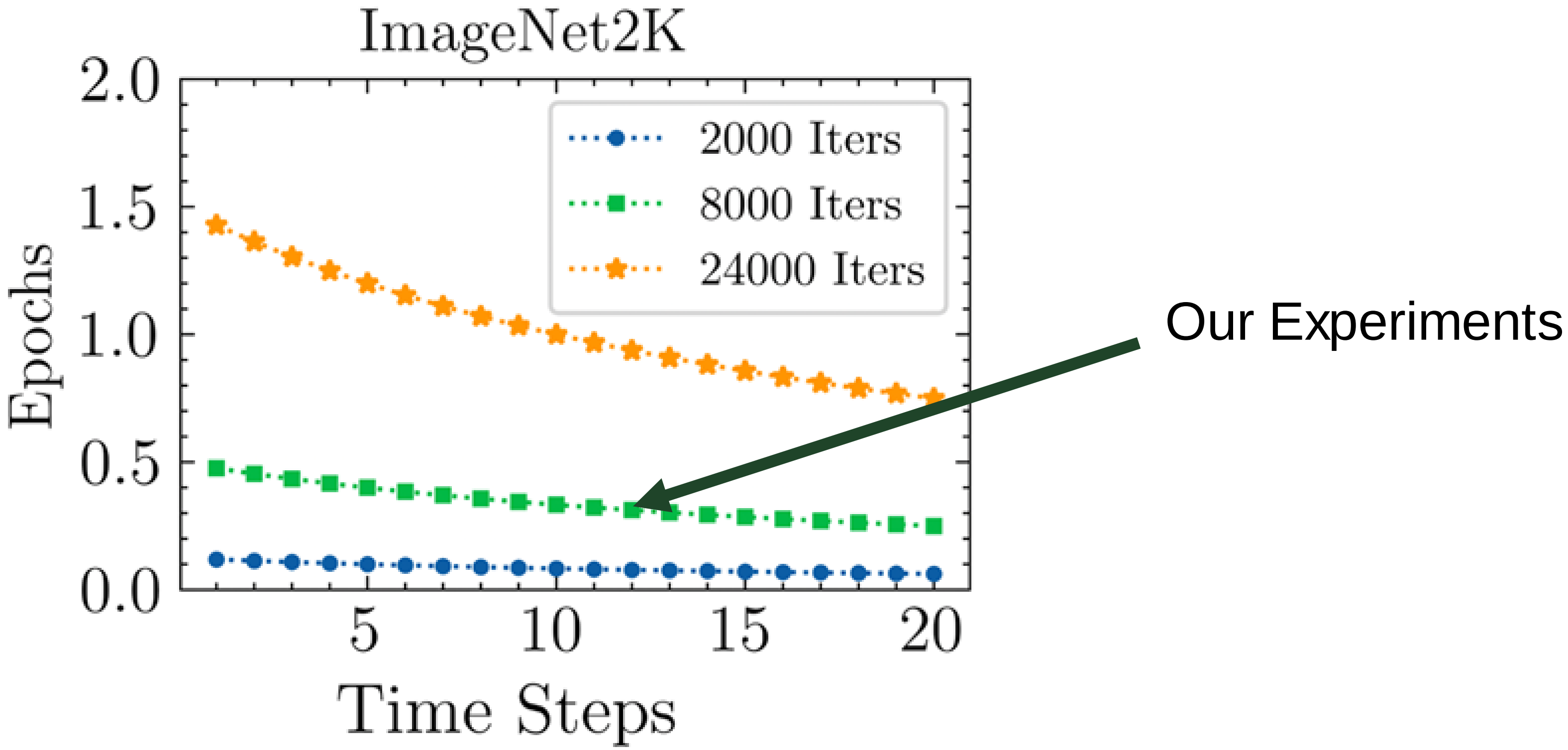
over steps $t \in \{1, 2, \dots, \infty\}$ where at every step:

Limited computation implicitly imposes memory restrictions

Restrict compute

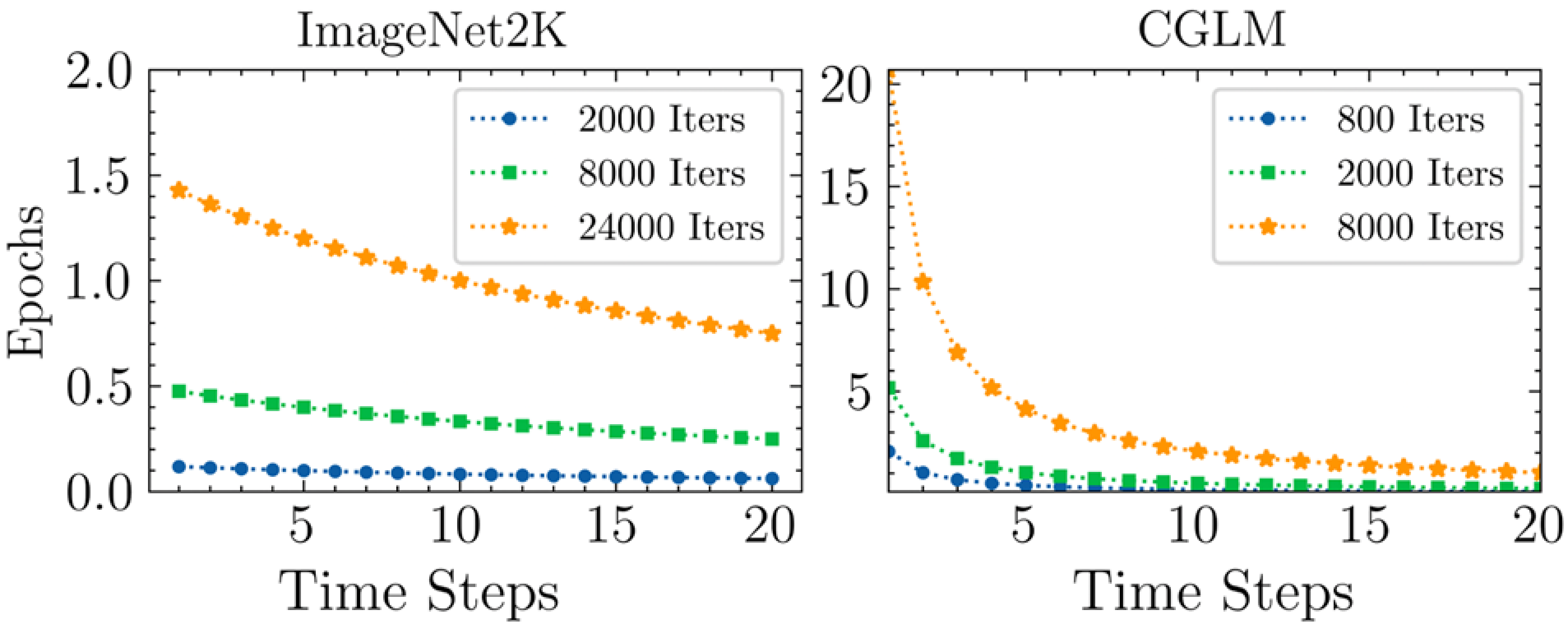
1. $\mathcal{S}$ reveals a set of image-label pairs $\{(x_i^t, y_i^t)\}_{i=1}^{n_t} \sim \mathcal{D}_{j \leq t}$
2. Memory is updated to $\mathcal{T}_t = \cup_{r=1}^{t} \{(x_i^r, y_i^r)\}_{i=1}^{n_r}$
3. Continual learner updates θ_t to θ_{t+1} spending a computational budget $\mathcal{C}_t$

Limited Computation: Practicals



ImageNet2K: ImageNet1K + 1.2M samples (1K classes) from ImageNet21K
Start: Pretrained model & ImageNet1K (in memory) -> Learn new 1K classes

Limited Computation



CGLM: Transfer learning from ImageNet1K pretrained model

Constructing the Streams

We consider three types of streaming settings:

- Class Incremental: Data ordered class wise
- Data Incremental: Data ordered by a random shuffling
- Time Incremental: Data ordered by the upload time to a server (natural)

Key Contributions

Dir.	Reference	Applicability (our setup)	Distillation	MemUpdate	MemRetrieve	FC Correction	Others
			Components				
	Naive	✓	-	Random	Random	-	-
Distillation	iCARL [44]	✓	**BCE**	**Herding**	Random	-	**NCM**
	LUCIR [25]	✓	**Cosine**	Herding	MargRank	**CosFC**	NCM
	PODNet [19]	✓	**POD**	Herding	Random	**LSC**	Imprint,NCM
	DER [9]	✓	**MSE**	Reservoir	Random	-	-
	CO^{2}L [12]	×	**IRD**	Random	Random	**Asym.SupCon**	-
	SCR [35]	✓	-	Reservoir	Random	**SupCon**	NCM
Sampling	TinyER [15]	✓	-	**FIFO,KMeans,Reservoir**	-	-	-
	GSS [5]	×	-	**GSS**	Random	-	-
	MIR [3]	×	-	Reservoir	**MIR**	-	-
	GDumb [40]	✓	-	**Balanced**	Random	-	**MemOnly**
	Mnemonics [31]	×	-	**Mnemonics**	-	-	BalFineTune
	OCS [57]	×	-	**OCS**	Random	-	-
	InfoRS [49]	×	MSE	**InfoRS**	Random	-	-
	RMM [30]	×	-	**RMM**	-	-	-
	ASER [48]	×	-	**SV**	**ASV**	-	-
	RM [6]	✓	-	**Uncertainty**	Random	-	AutoDA
	CLIB [27]	×	-	**Max Loss**	Random	-	MemOnly,AdaLR
FC Layer	BiC [53]	×	CrossEnt	Random	Random	**BiC**	-
	WA [60]	×	CrossEnt	Random	Random	**WA**	-
	SS-IL [2]	×	TKD	Random	Balanced	**SS**	-
	CoPE [17]	✓	-	Balanced	Random	**PPPLoss**	-
	ACE [10]	✓	-	Reservoir	Random	**ACE**	-

Three Principal Directions

Reduce effect of distribution shift from past data by:

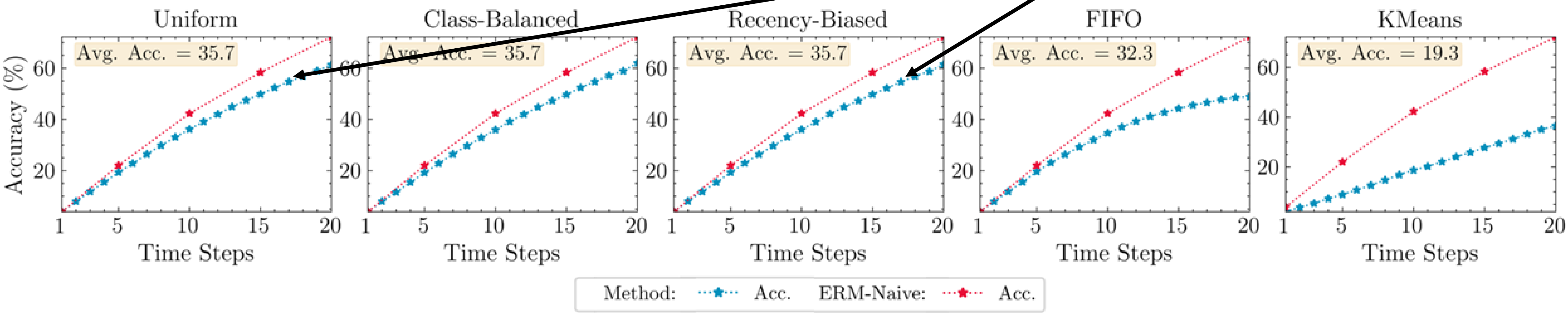
- **Distillation:** Enforce Output Similarity with Old Models
- **Sampling Old Data:** Create representative coreset of past knowledge
- **Correcting FC Layer:** Posits knowledge in representations far less affected, but classifier gets worse

Do Sampling Strategies Matter?

CGLM (Budget: 2000 Iterations): Method and ERM-Naive

ERM-Naive:Trains a model from scratch given the data and cumulative compute budget until current timestep (empirical upper bound)

Most cheap sampling strategies perform similarly

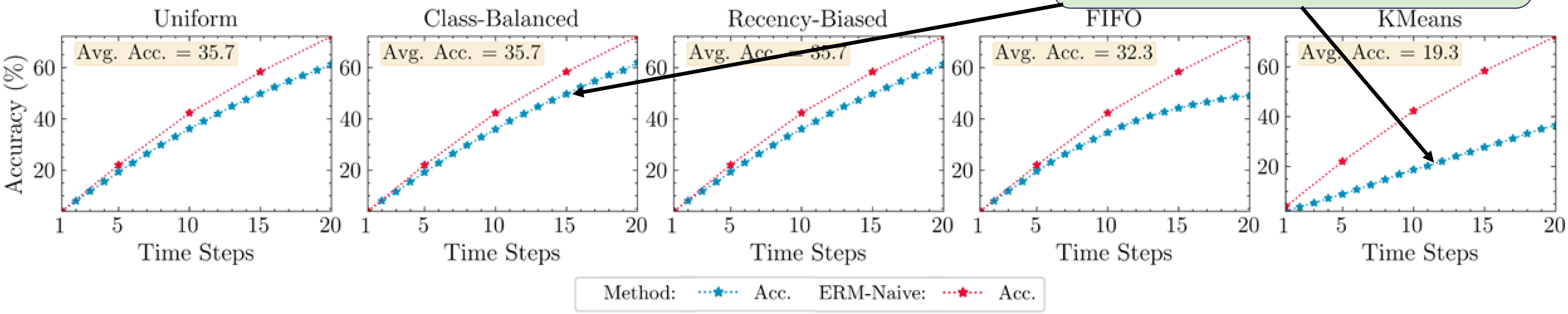


Do Sampling Strategies Matter?

CGLM (Budget: 2000 Iterations): Method and ERM-Naive

ERM-Naive:Trains a model from scratch given the data and cumulative compute budget until current timestep (empirical upper bound)

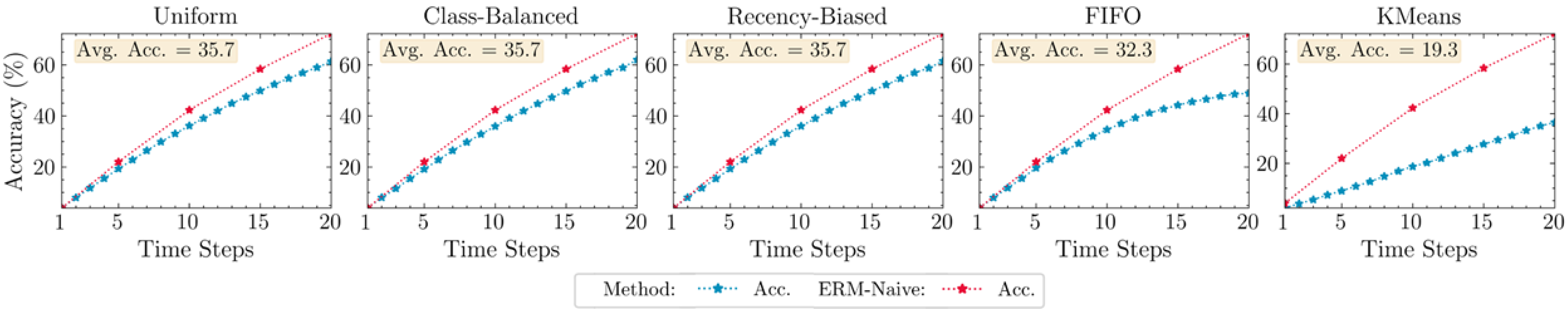
Expensive sampling strategies waste compute, lowering performance



Do Sampling Strategies Matter?

CGLM (Budget: 2000 Iterations): Method and ERM-Naive

ERM-Naive:Trains a model from scratch given the data and cumulative compute budget until current timestep (empirical upper bound)

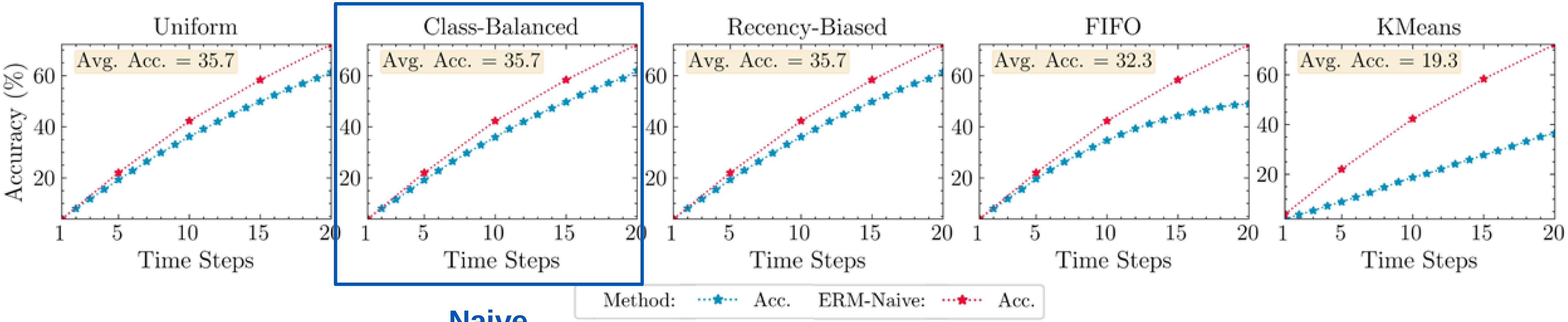


Conclusion: Do not spend computational budget on expensive sampling. Train your model with the budget!

Do Sampling Strategies Matter?

CGLM (Budget: 2000 Iterations): Method and ERM-Naive

ERM-Naive:Trains a model from scratch given the data and cumulative compute budget until current timestep (empirical upper bound)

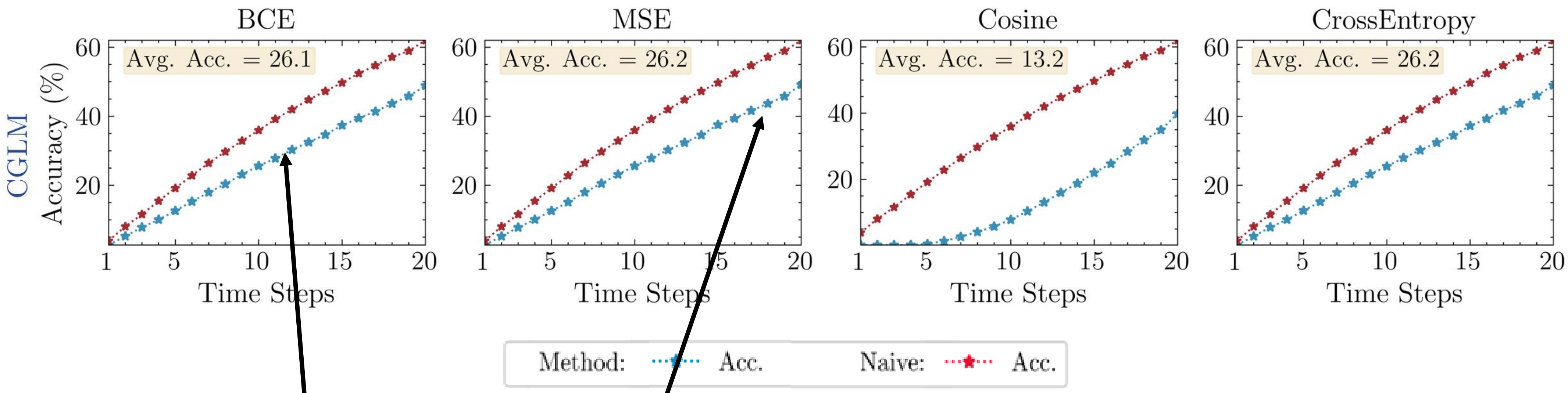


Naive

(Not to be confused with ERM-Naive)

Does Distillation Matter?

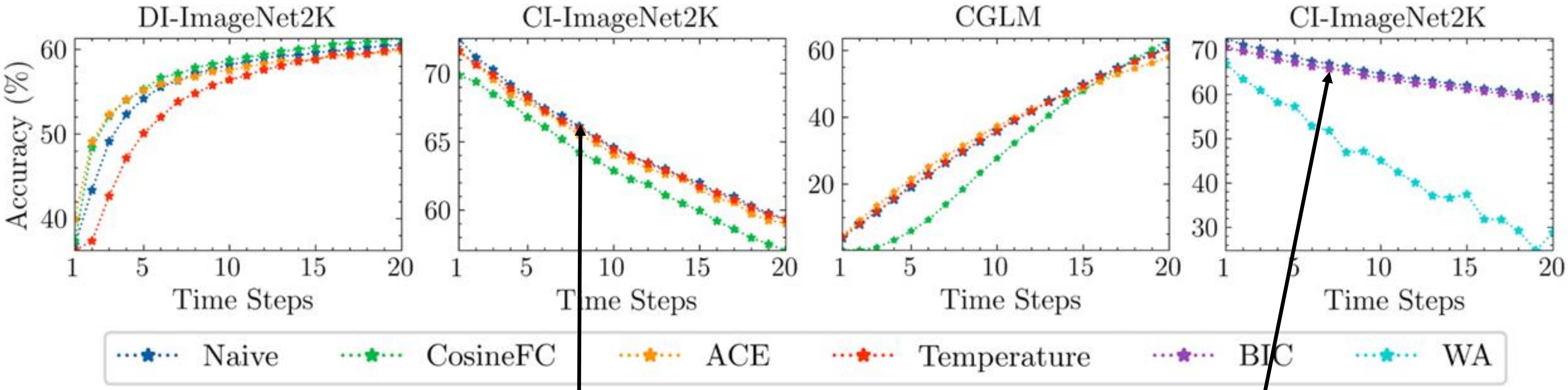
CGLM (Budget: 2000 Iterations): Naive with Distillation Methods



Distillation has overhead computational costs, better to simply train a model

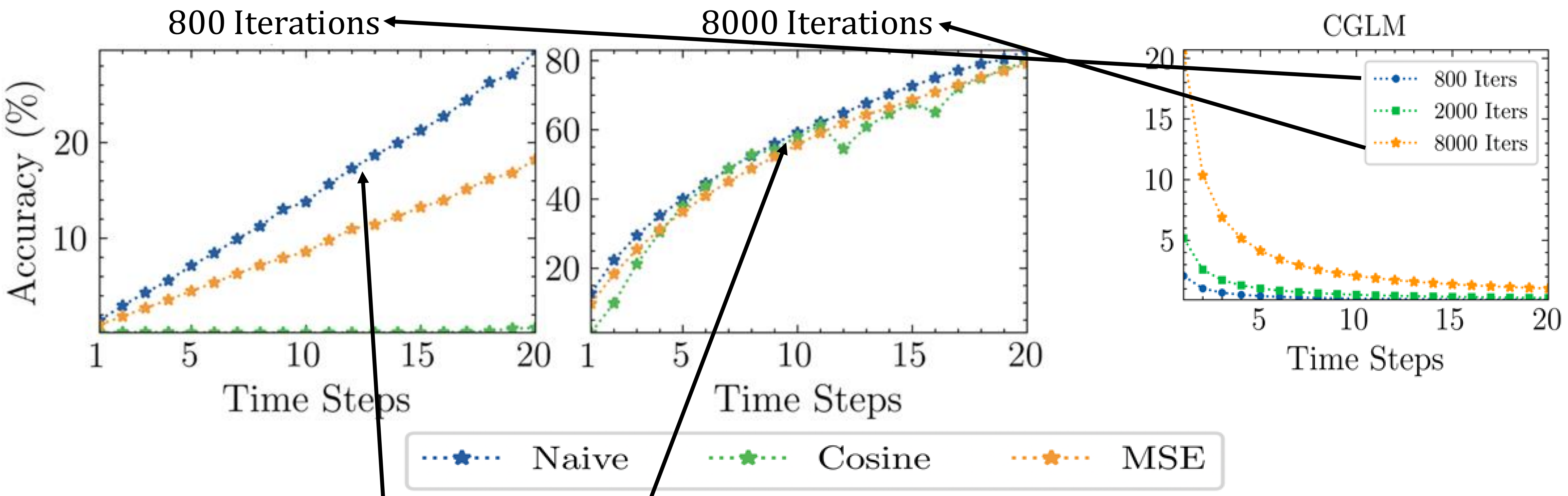
Does FC Layer Correction Matter?

CGLM (Budget: 2000 Iterations): Naive with FC Correction Methods



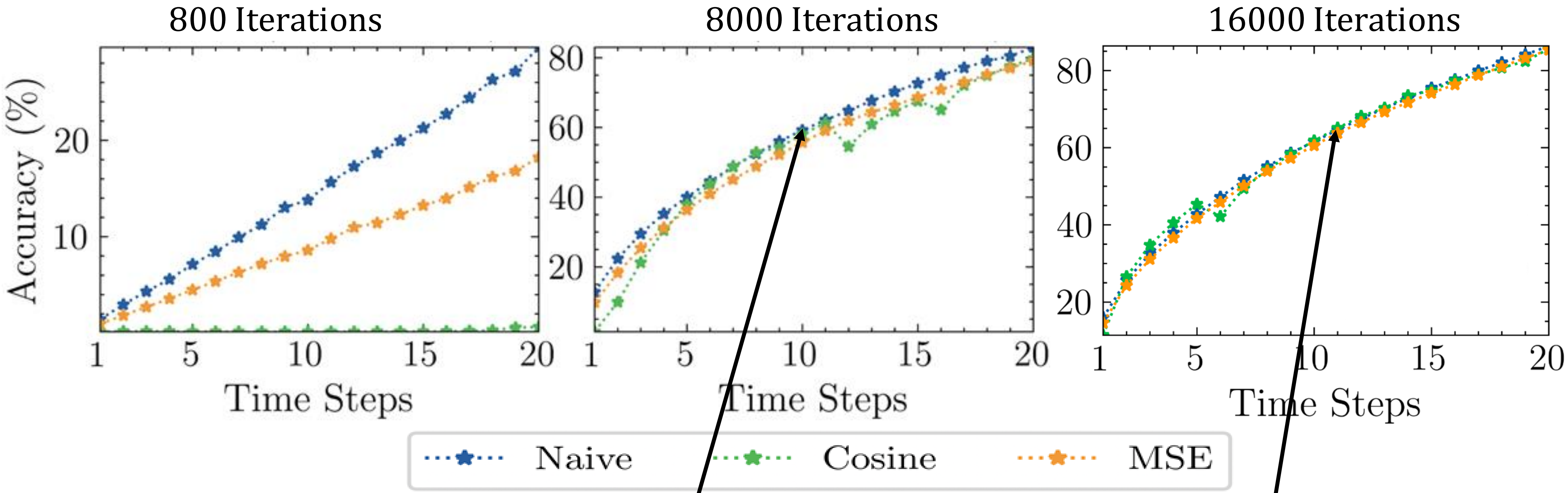
FC Corrections perform equal to or worse than no FC corrections (Naive) baseline

Sensitivity Analysis: Compute Budget



The gap compared to uniform reduces as the computation is abundant

Sensitivity Analysis: Compute Budget



However, methods simply converge instead of outperforming uniform with more compute

Conclusions

- Existing CL algorithms (*sampling, distillation, and FC corrections*) fail in a compute budgeted setup
- Naive baseline of experience replay outperforms all considered CL methods
- Above conclusions persistent across:

  (a) computational budgets (b) varying number of time steps

Thank You!

Questions?