MAML is a Noisy Contrastive Learner

NewInML Workshop@NeurIPS'21

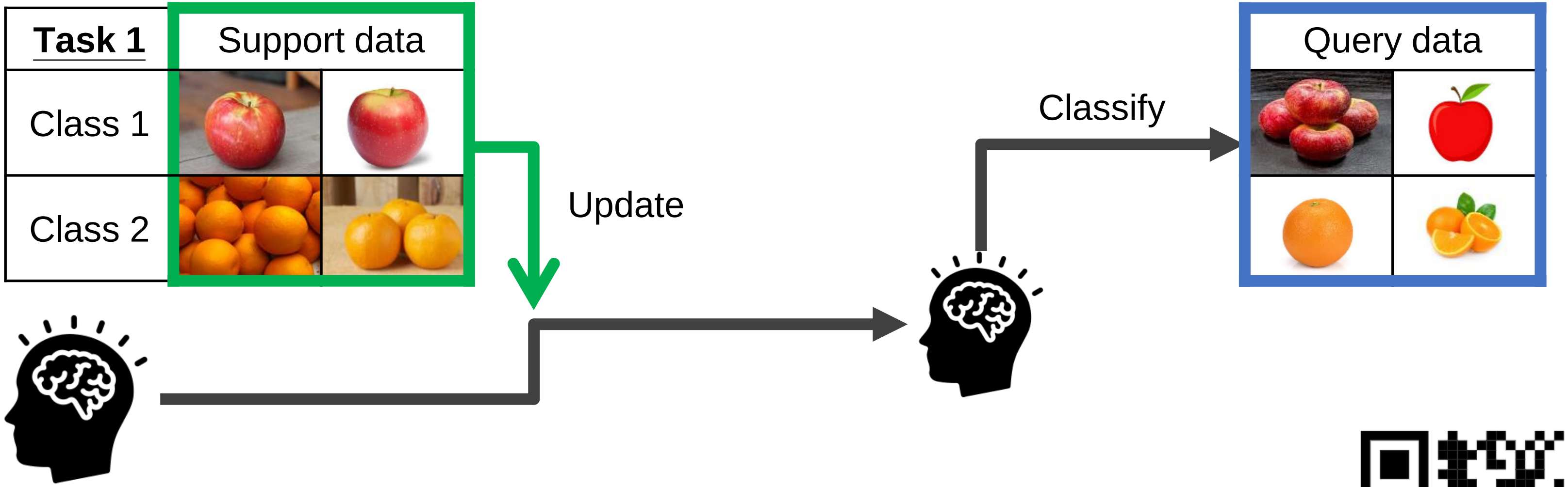
Chia-Hsiang Kao[1], Wei-Chen Chiu[1], Pin-Yu Chen[2]

[1]National Yang Ming Chiao Tung University

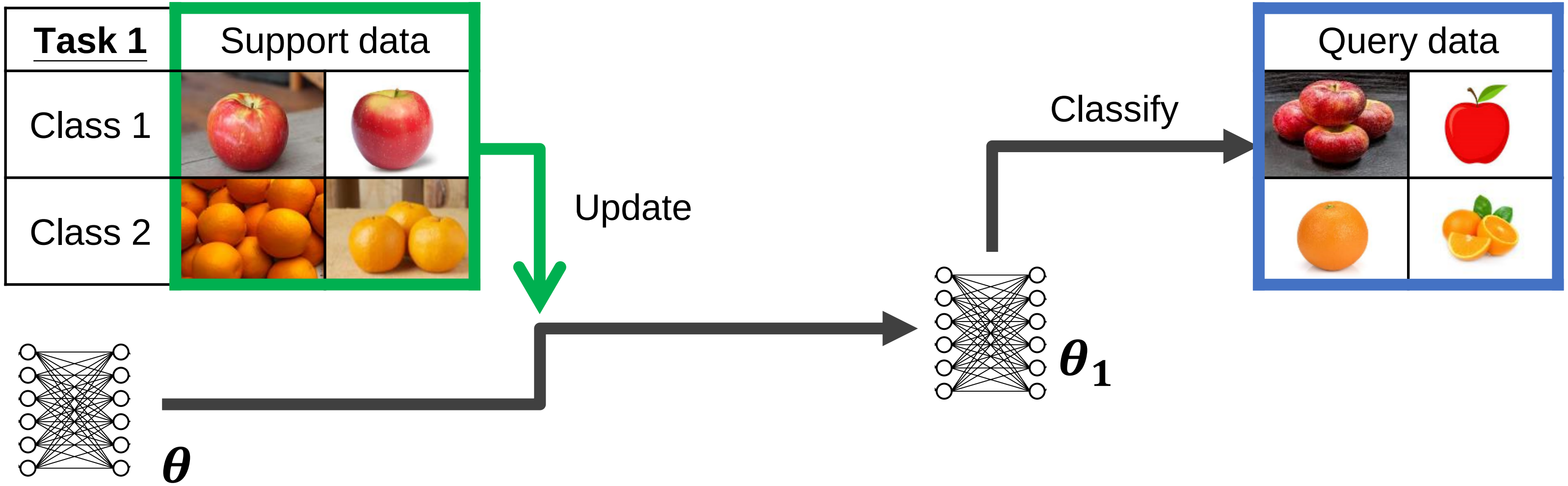
[2]MIT-IBM Watson AI Lab

Introduction

Humans learn to classify even with limited experience.

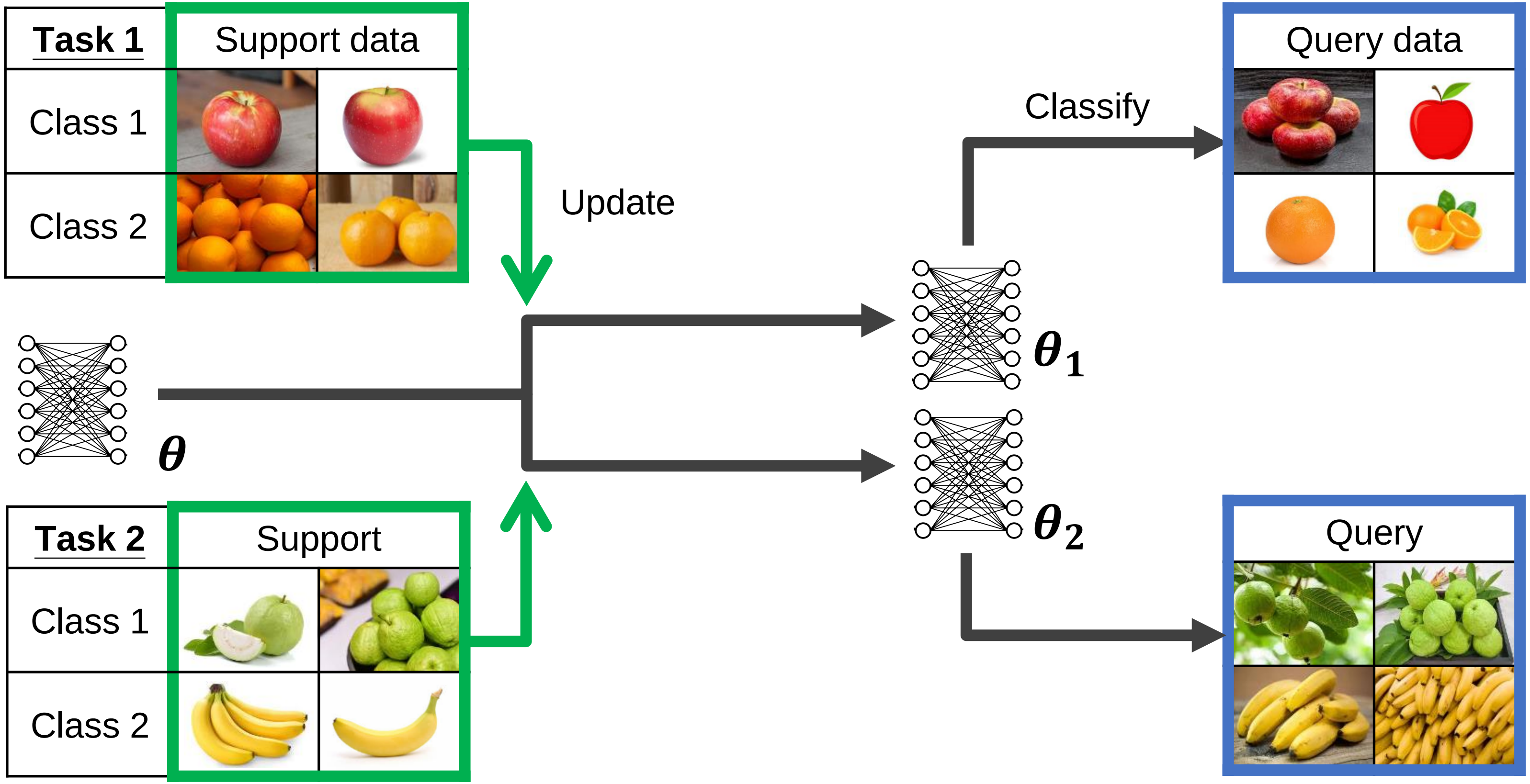


Introduction
Humans learn to classify even with limited experience.



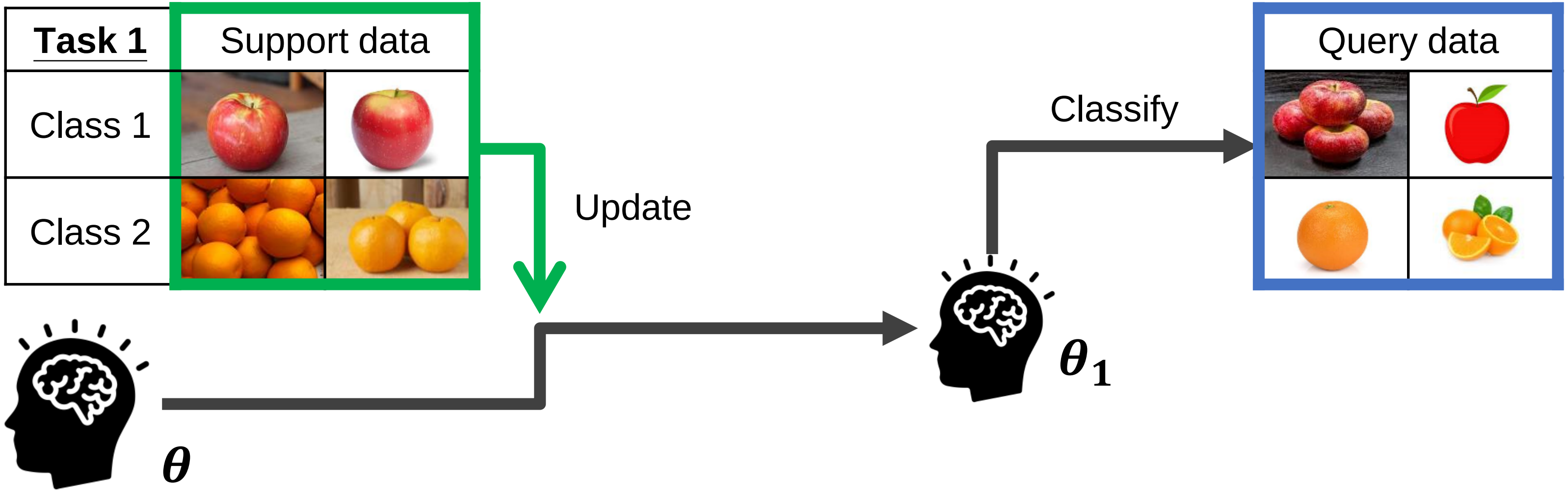
Introduction

Humans learn to classify even with limited experience.



Introduction

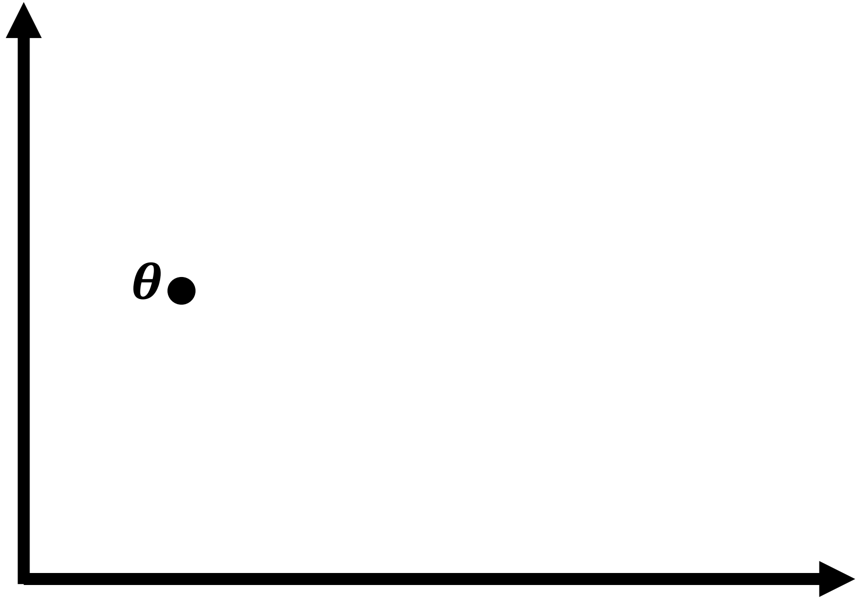
Humans learn to classify even with limited experience.

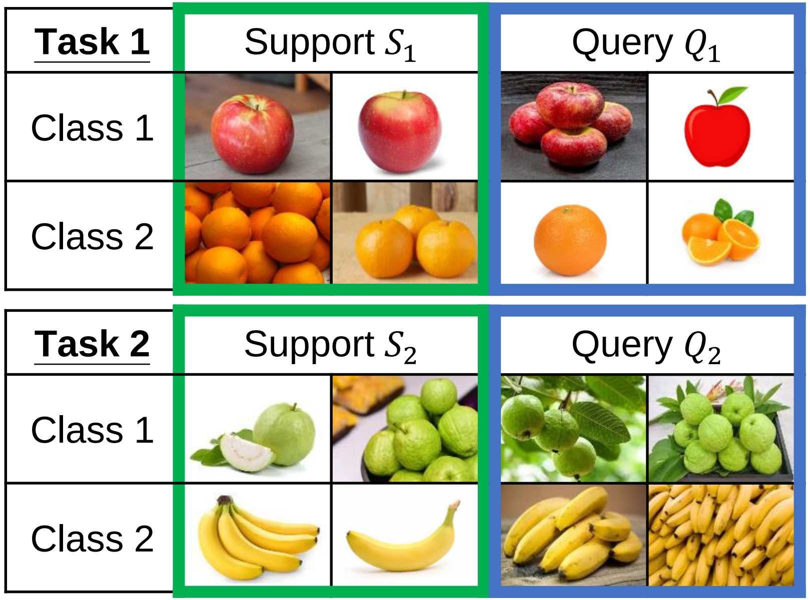


- MAML is a gradient-based meta-learning algorithm that finds a good θ.

Introduction

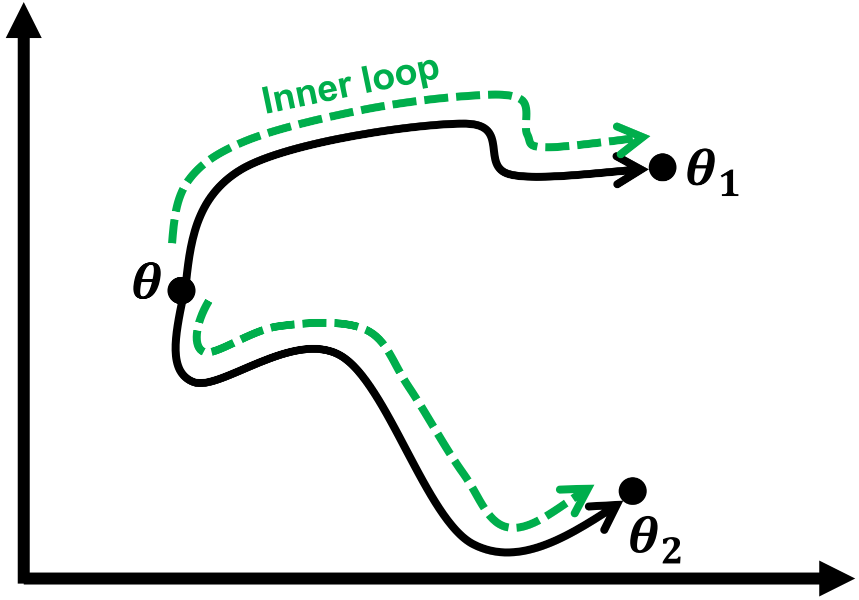
MAML makes model learn to classify after seeing little data.

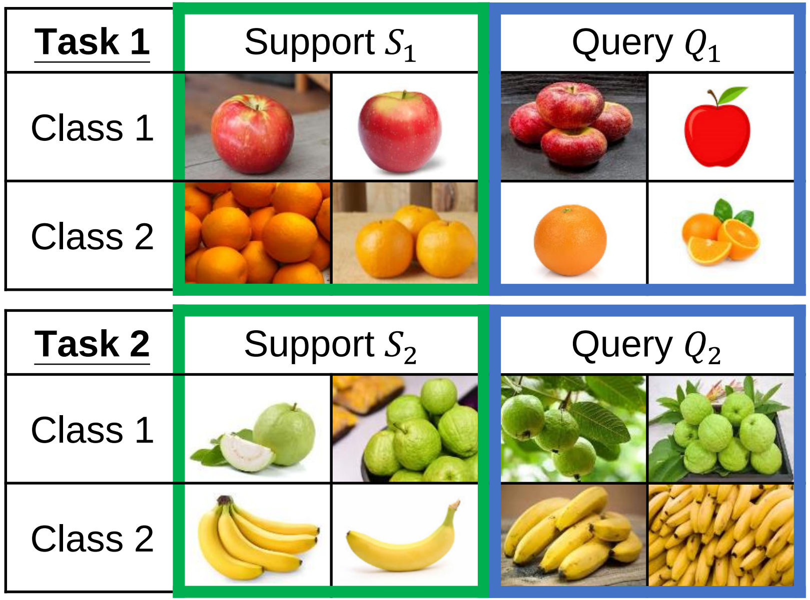




Introduction

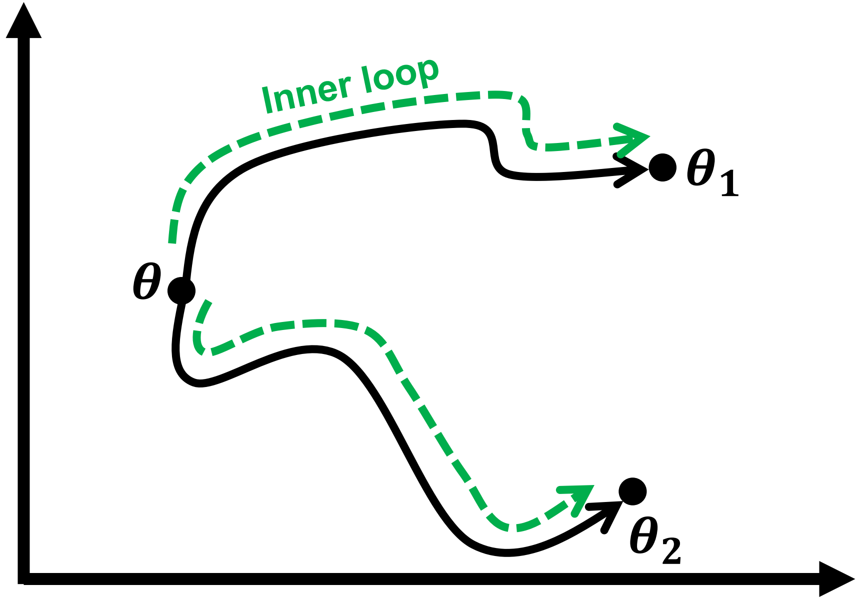
MAML makes model learn to classify after seeing little data.

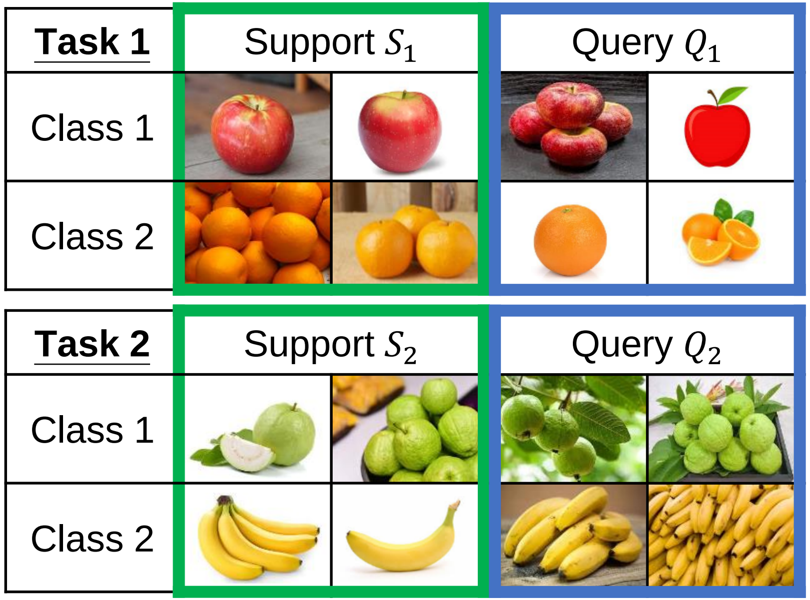




Introduction

MAML makes model learn to classify after seeing little data.

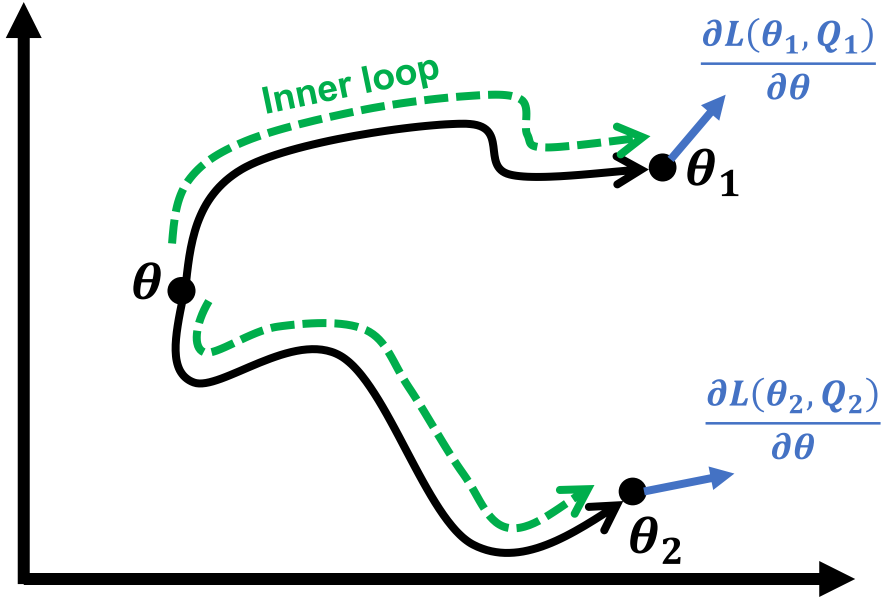


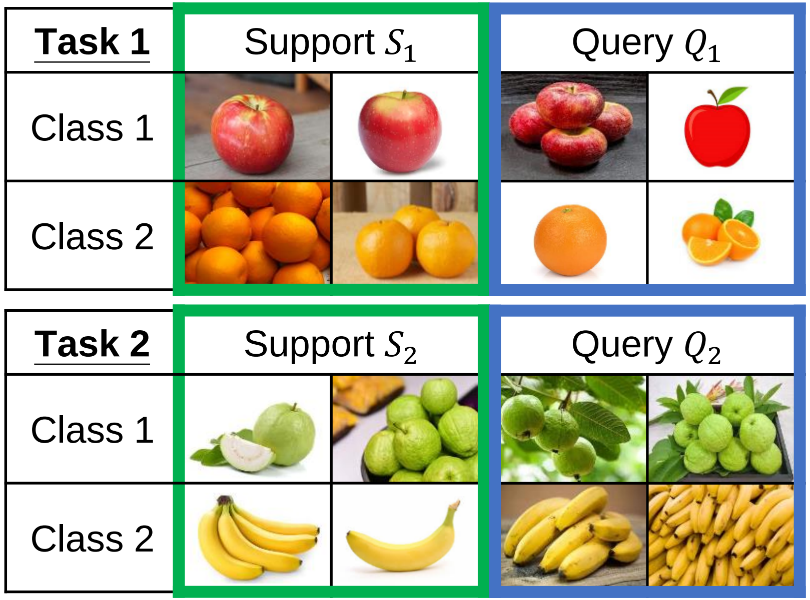


- Goal: Minimize $\mathrm{L}(\theta_1, Q_1)$ and $\mathrm{L}(\theta_2, Q_2)$ by finding best θ.

Introduction

MAML makes model learn to classify after seeing little data.

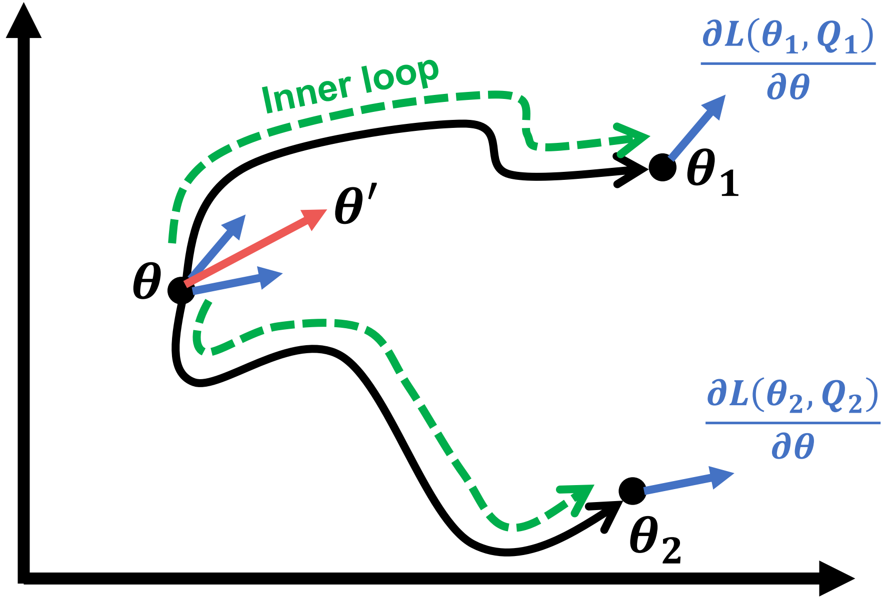


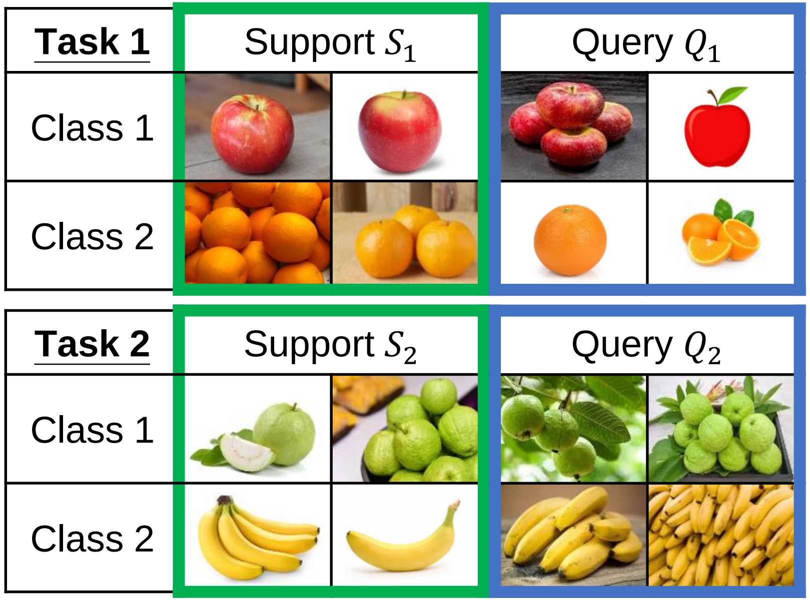


- Goal: Minimize $\mathrm{L}(\theta_1, Q_1)$ and $\mathrm{L}(\theta_2, Q_2)$ by finding best θ.

Introduction

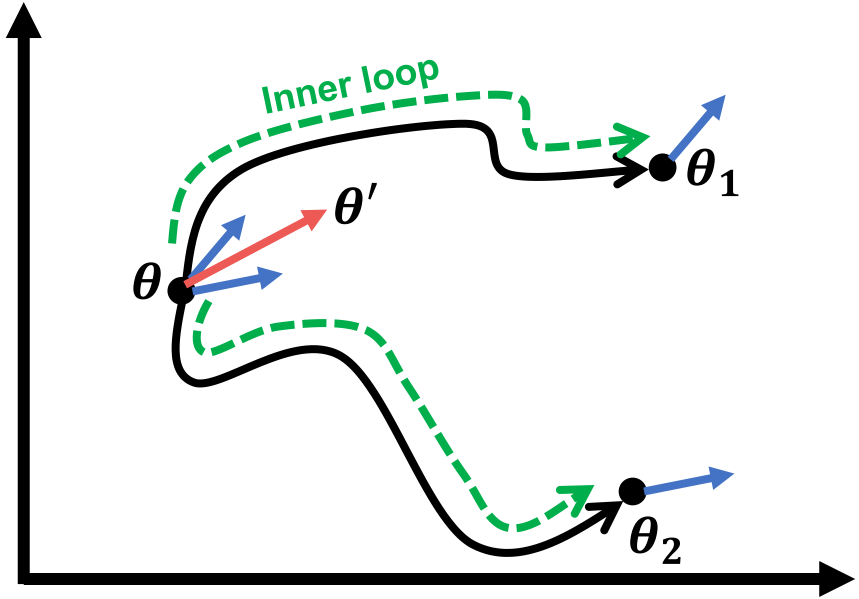
MAML makes model learn to classify after seeing little data.





- Goal: Minimize $\mathrm{L}(\theta_1, Q_1)$ and $\mathrm{L}(\theta_2, Q_2)$ by finding best θ.

Introduction
MAML makes model learn to classify after seeing little data.



Algorithm 1 Second-order MAML

while not done **do**
  Sample tasks $\{T_1, T_2\}$
  for $n = 1, 2$ **do**
    $\{S_n, Q_n\} \leftarrow$ sample from T_n
    $\theta_n = \theta$
    for $i = 1, 2, \ldots, N_{step}$ **do**
      $\theta_n \leftarrow \theta_n - \eta \nabla_{\theta_n} L(\theta_n, S_n)$
    end for
  end for
  Update $\theta \leftarrow \theta - \rho \sum_{n=1}^{N_{batch}} \nabla_\theta L(\theta_n, Q_n)$
end while

- Goal: Minimize $\mathrm{L}(\theta_1, Q_1)$ and $\mathrm{L}(\theta_2, Q_2)$ by finding best θ.
- Method: update θ by $\theta' = \theta - \sum \frac{\partial L(\theta_n, Q_n)}{\partial \theta}$

Introduction

MAML makes model learn to classify after seeing little data.

Why is MAML successful?

- It is widely believed that MAML **encourages** models to learn a general-purpose representations which are applicable to novel tasks.

Introduction

MAML makes model learn to classify after seeing little data.

Why is MAML successful?

- It is widely believed that MAML **encourages** models to learn a general-purpose representations which are applicable to novel tasks.

In this paper, we step further and ask:

- **How** does MAML **encourage** any model to learn general-purpose representations?
- What is the role of the support and query data and how do they interact with each other?

Introduction

MAML makes model learn to classify after seeing little data.

Why is MAML successful?

- It is widely believed that MAML **encourages** models to learn a general-purpose representations which are applicable to novel tasks.

In this paper, we step further and ask:

- **How** does MAML **encourage** any model to learn general-purpose representations?
- What is the role of the support and query data and how do they interact with each other?

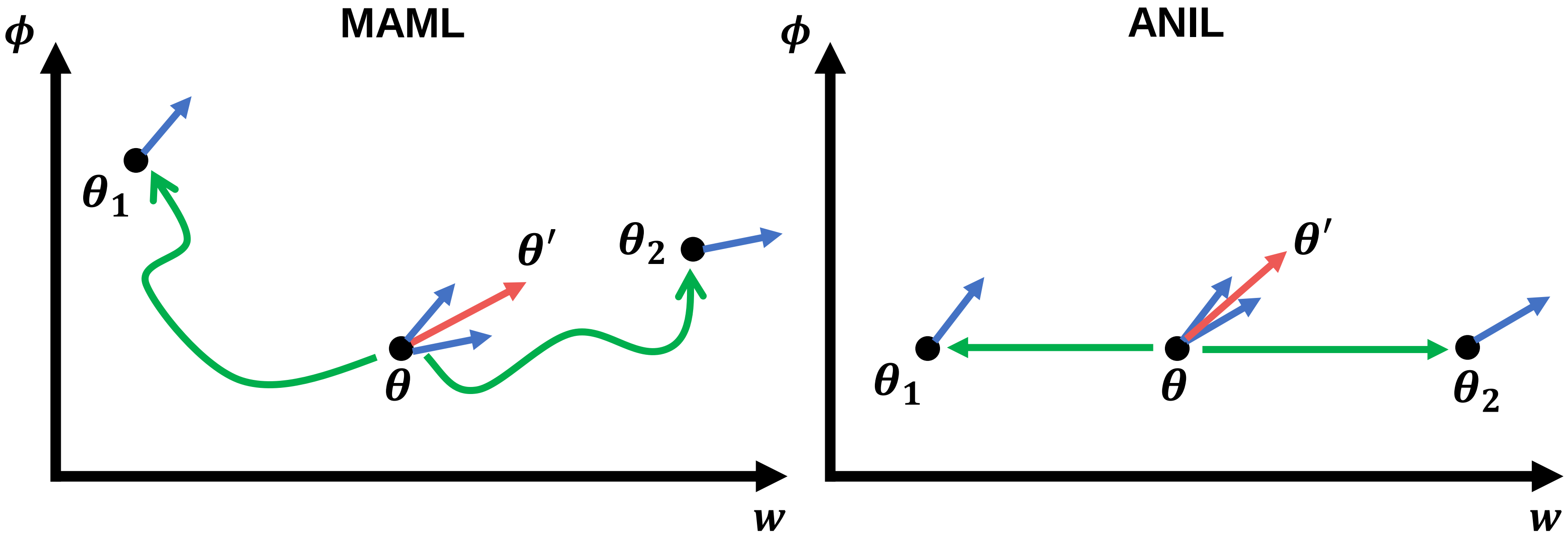
Our contribution:

- We show that MAML is a noisy supervised contrastive learning algorithm.

Assumption
ANIL (Almost no inner loop)

Consider a model $\boldsymbol{\theta} = \{\boldsymbol{\phi}, \boldsymbol{w}\}$, where $\boldsymbol{\phi}$ is an encoder and $\boldsymbol{w}$ is a linear classifier.

ANIL states that **the encoder $\boldsymbol{\phi}$ is not updated during the inner loop**.



Assumption
ANIL (Almost no inner loop)

Consider a model $\boldsymbol{\theta} = \{\boldsymbol{\phi}, \boldsymbol{w}\}$, where $\boldsymbol{\phi}$ is an encoder and $\boldsymbol{w}$ is a linear classifier.

ANIL states that **the encoder $\boldsymbol{\phi}$ is not updated during the inner loop**.

The ANIL Assumption empirically sounds.

	Mini-ImageNet 5way-1shot	**Mini-ImageNet** 5way-5shot	**Omniglot** 20way-1shot	**Omniglot** 20way-5shot
MAML	**46.9±0.2**	**63.1±0.4**	93.7±0.7	96.4±0.1
ANIL	46.7±0.4	61.5±0.5	**96.2±0.5**	**98.0±0.3**

Rapid Learning or Feature Reuse? Towards Understanding the Effectiveness of MAML (ICLR 2020)

Main Derivation
Motivating example

Assumptions:

- ANIL.
- Linear classifier w is zeroed at the beginning.

Loss: Mean square error.

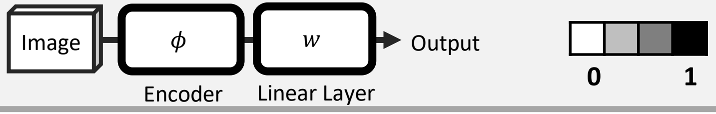
Condition: One inner loop update.

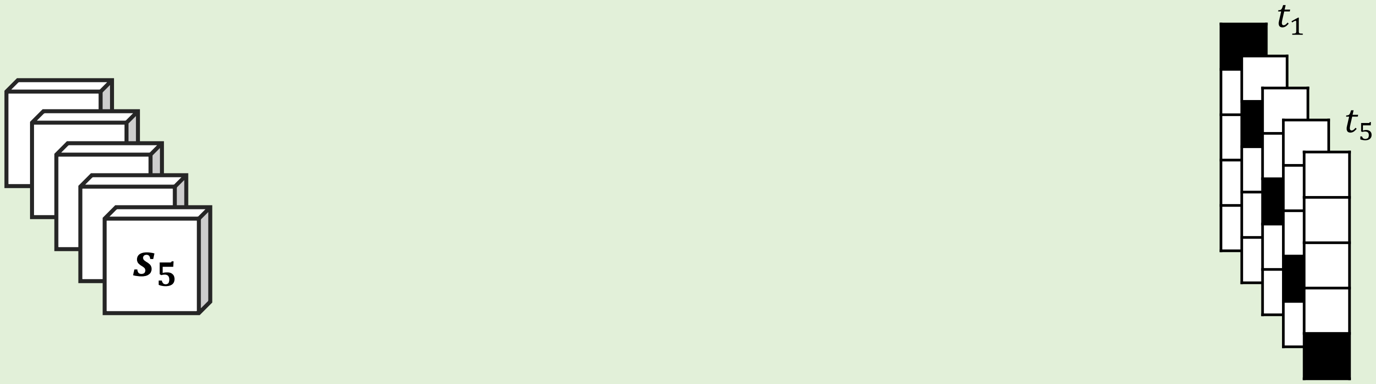
Setting

- 5-way: Each task contains 5 classes of images.
- 1-shot: Only one image per class in the support data.

Setting:
5-way 1-shot using MAML with **one** inner-loop update under **MSE loss**.

Model:



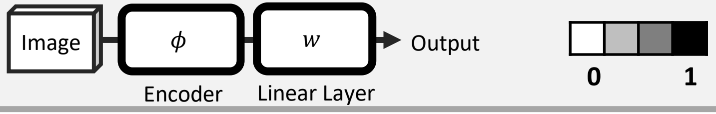


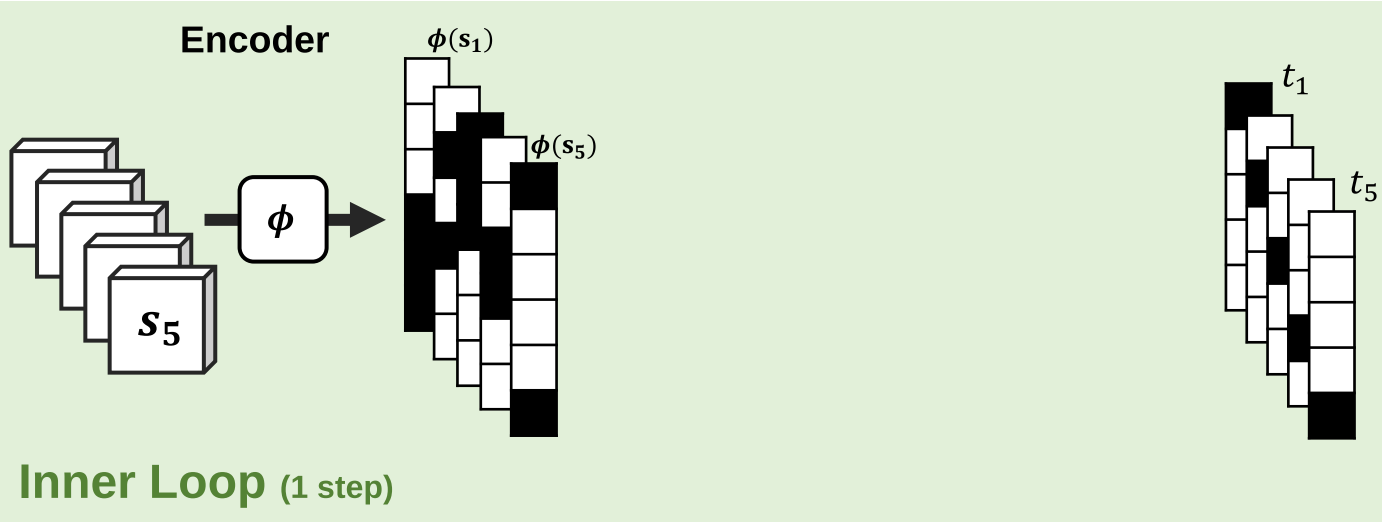
Inner Loop (1 step)

19

Setting:
5-way 1-shot using MAML with **one** inner-loop update under **MSE loss**.

Model:



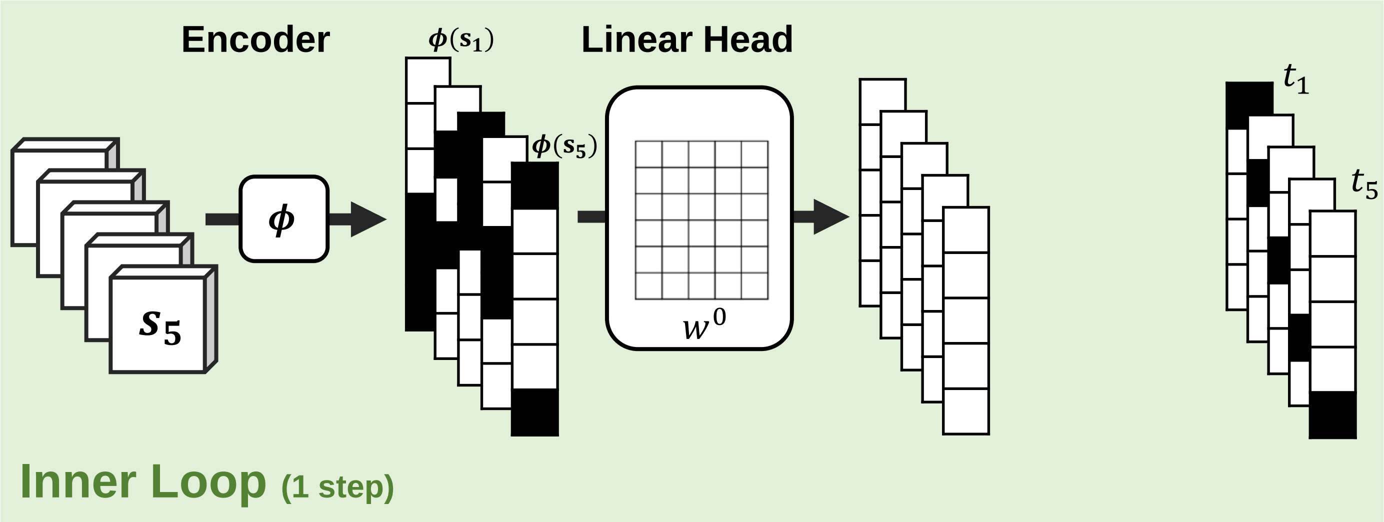


20

Setting:
5-way 1-shot using MAML with **one** inner-loop update under **MSE loss**.

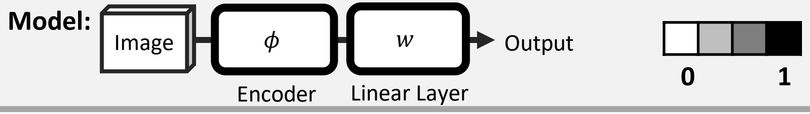
Model: Image → ϕ (Encoder) → w (Linear Layer) → Output

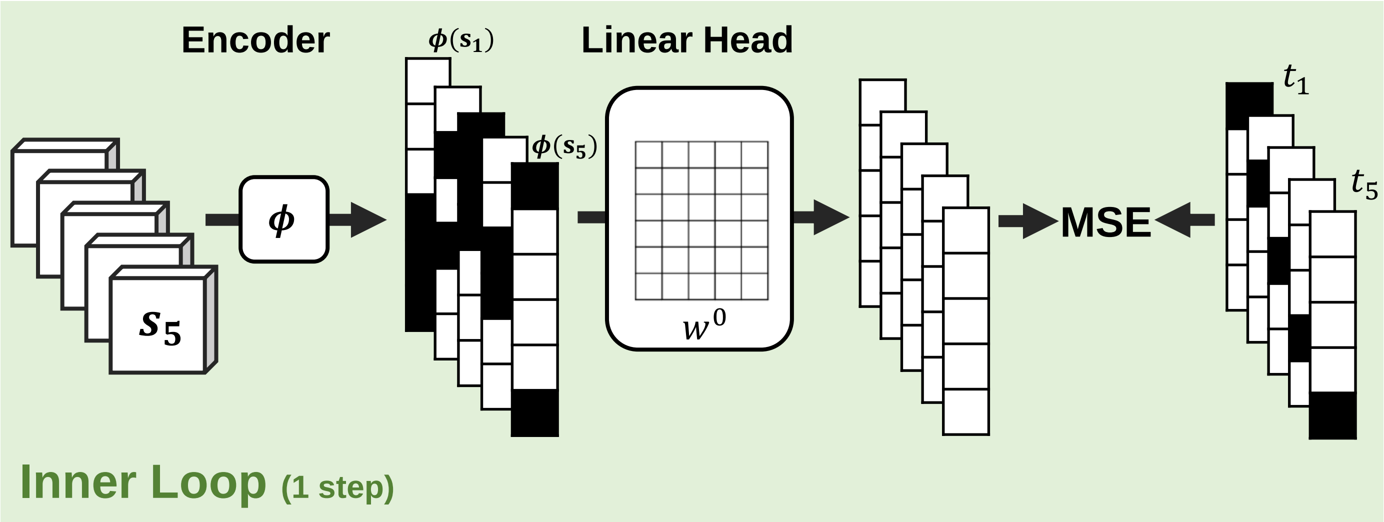
0 1



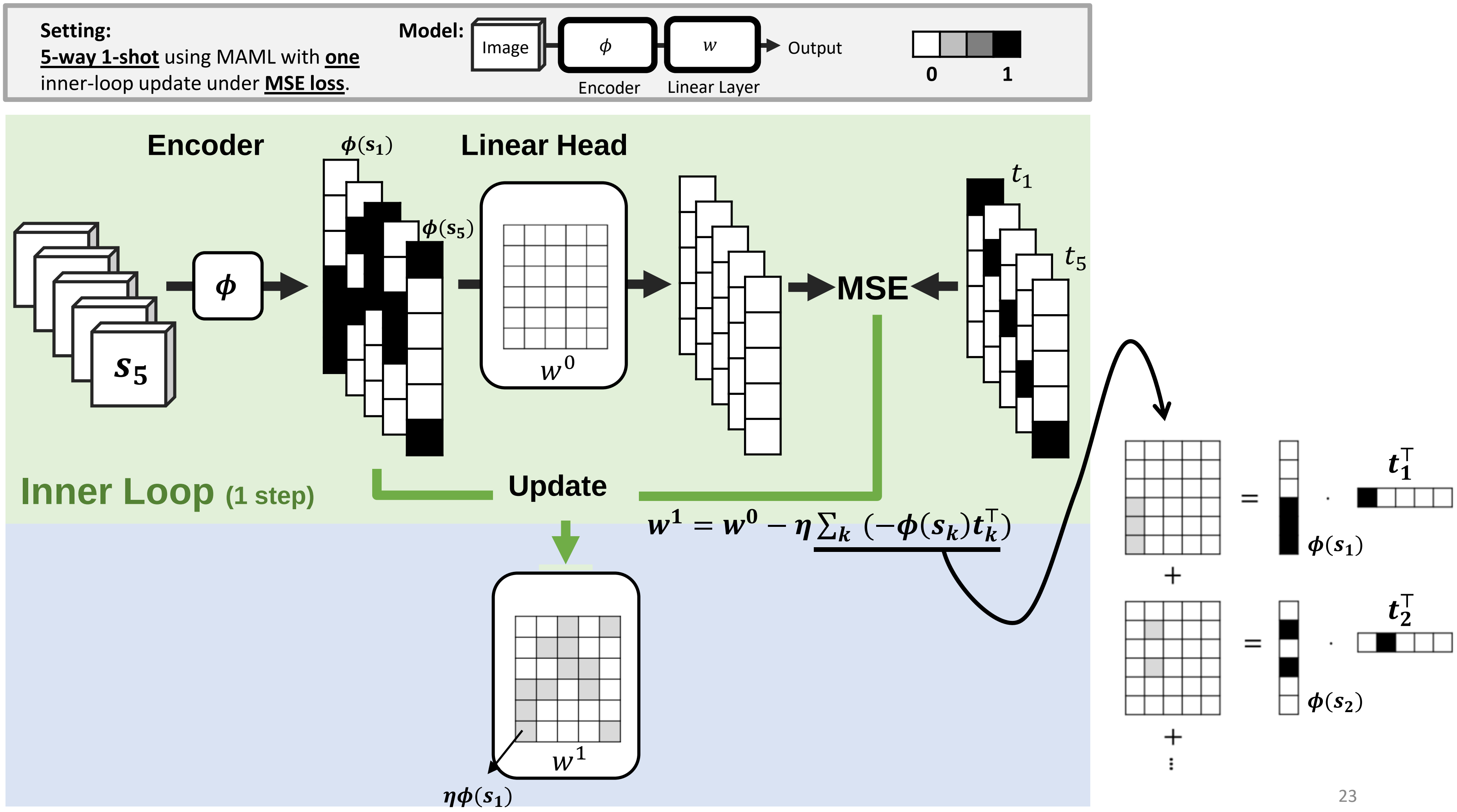
21

Setting:
5-way 1-shot using MAML with **one** inner-loop update under **MSE loss**.

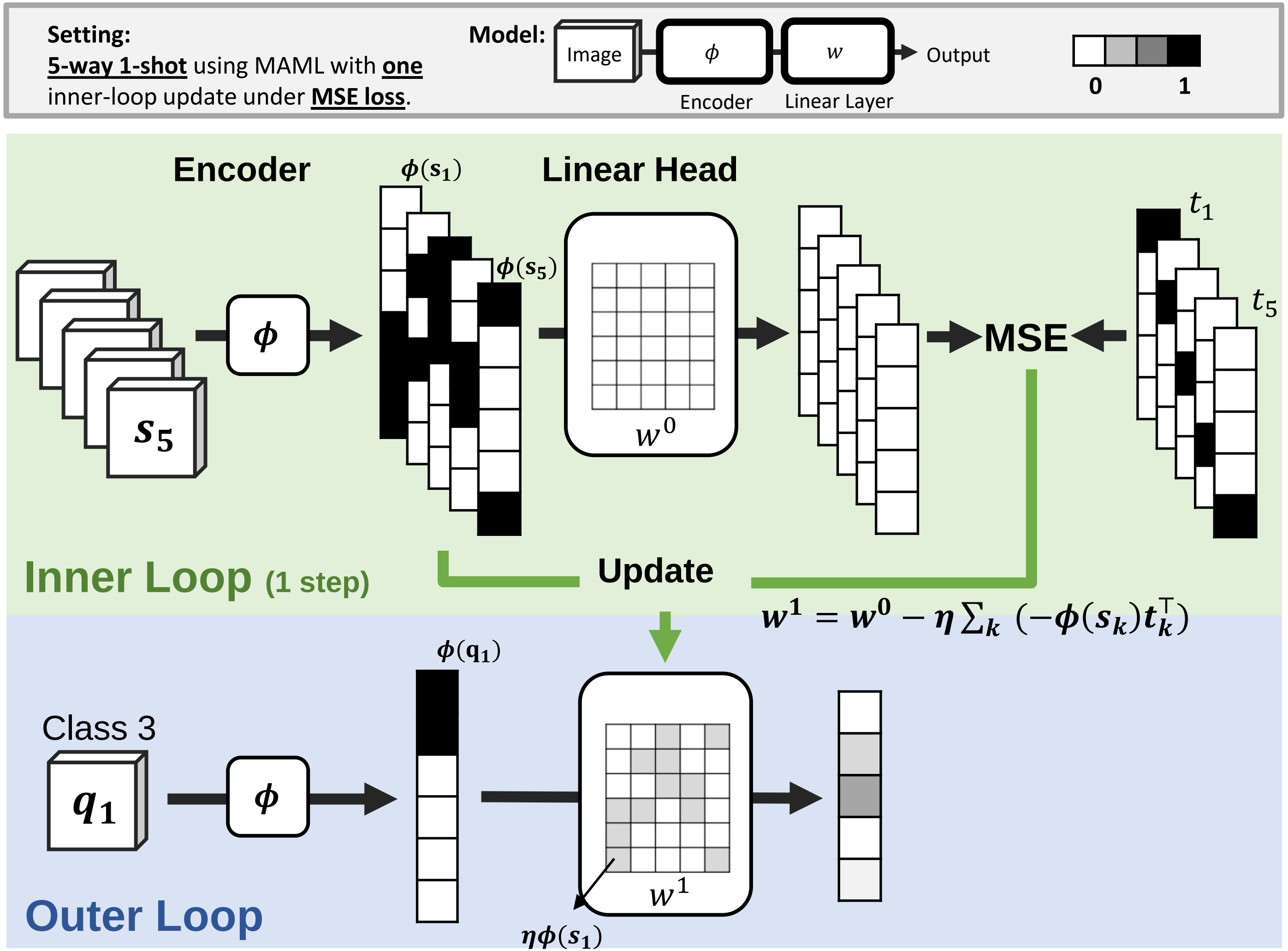


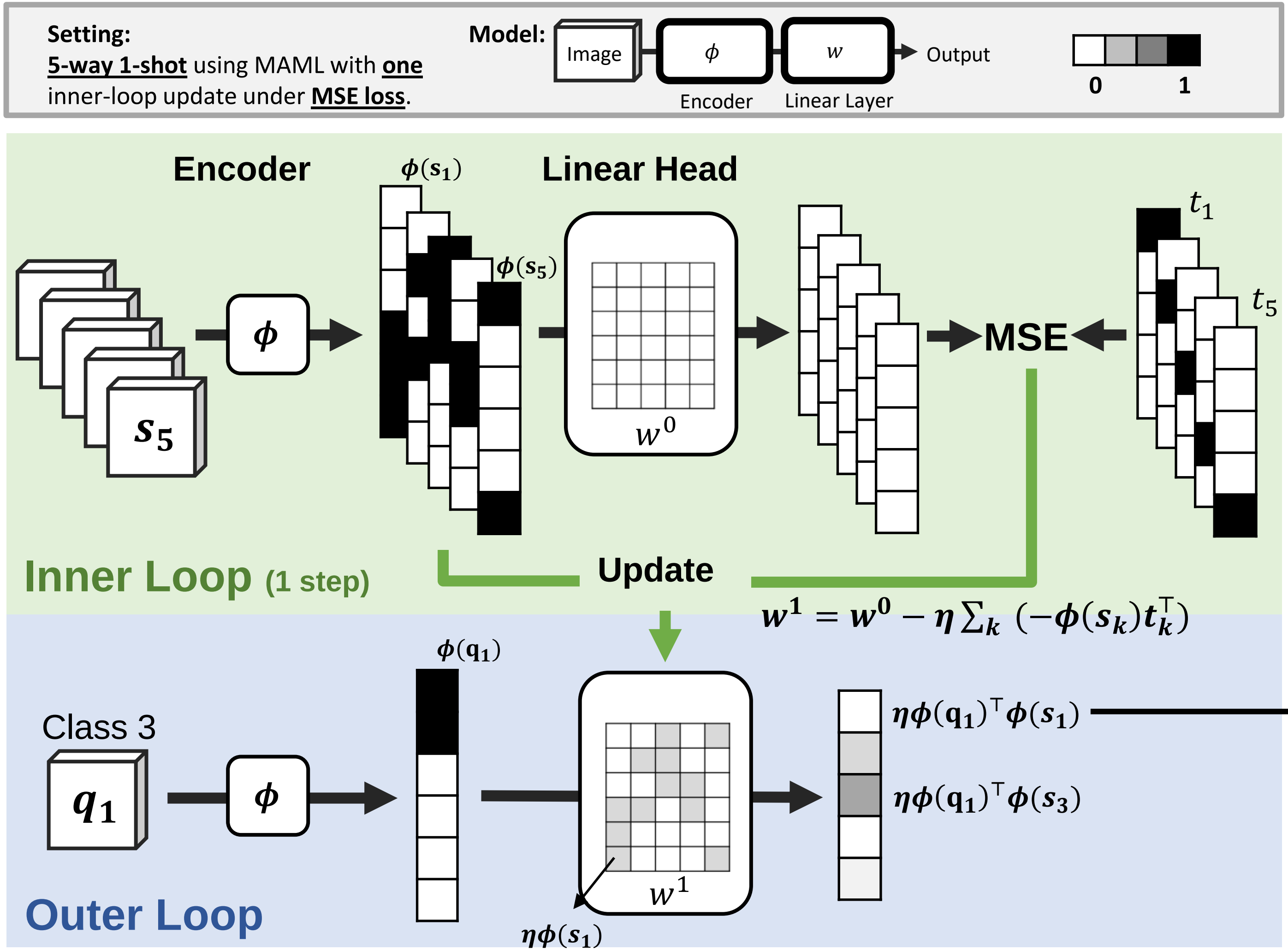


22

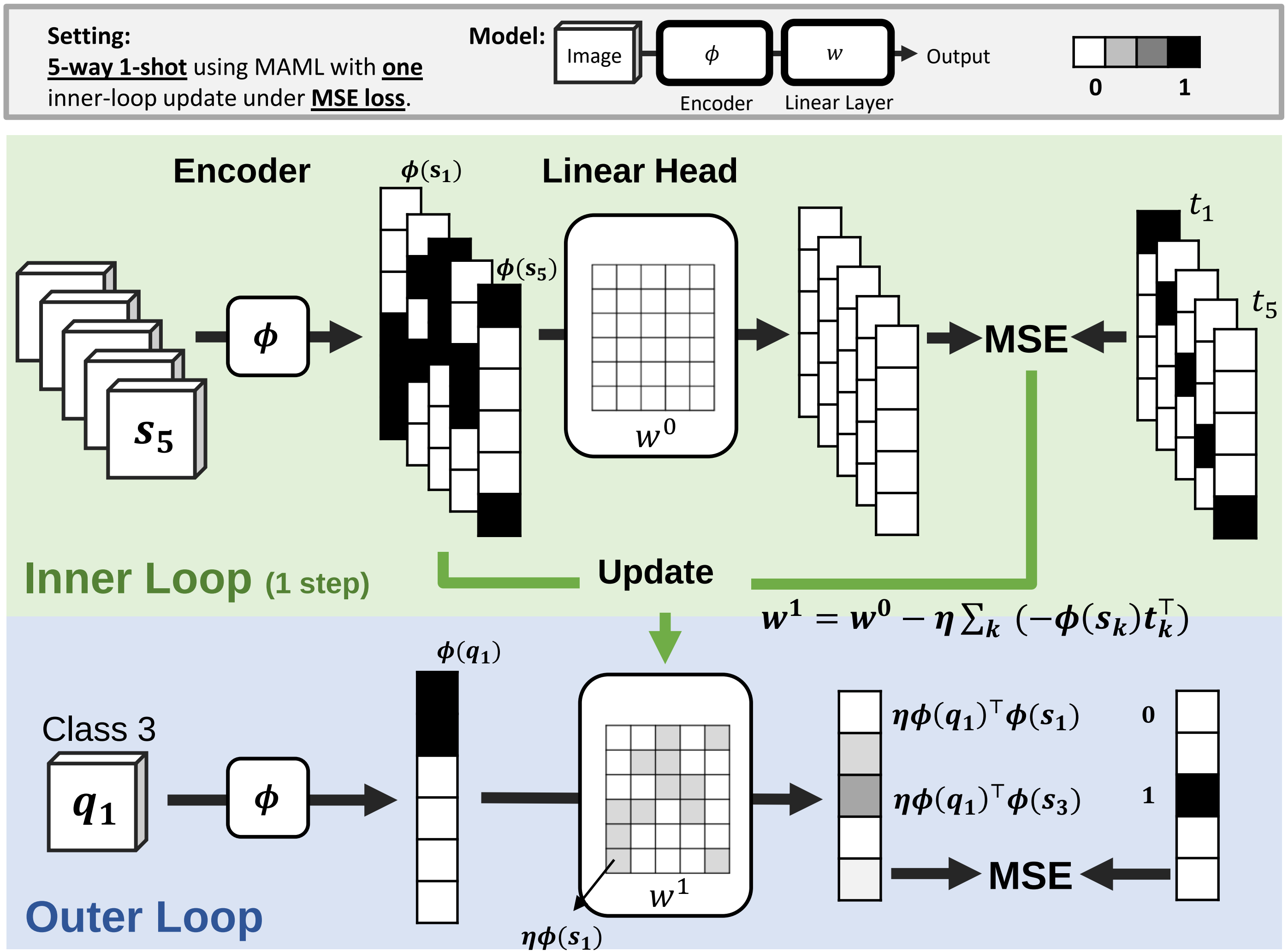


Setting:
5-way 1-shot using MAML with one inner-loop update under MSE loss.
Model:
Image
ϕ
w
Output
Encoder
Linear Layer
0
1
Encoder
$\phi(s_1)$
Linear Head
$\phi(s_5)$
s_5
ϕ
w^0
t_1
t_5
MSE
Inner Loop (1 step)
Update
$w^1 = w^0 - \eta \sum_k (-\phi(s_k) t_k^\top)$
w^1
$\eta\phi(s_1)$
$=$
$\phi(s_1)$
$t_1^\top$
$+$
$=$
$\phi(s_2)$
$t_2^\top$
$+$
⋮


Setting:
5-way 1-shot using MAML with one inner-loop update under MSE loss.
Model:
Image
ϕ
w
Output
Encoder
Linear Layer
0
1
Encoder
$\phi(s_1)$
Linear Head
$\phi(s_5)$
s_5
ϕ
w^0
t_1
t_5
MSE
Inner Loop (1 step)
Update
$w^1 = w^0 - \eta \sum_k (-\phi(s_k) t_k^\top)$
$\phi(q_1)$
Class 3
q_1
ϕ
w^1
$\eta\phi(s_1)$
Outer Loop

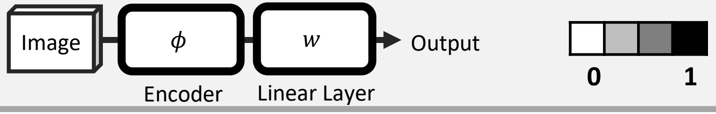


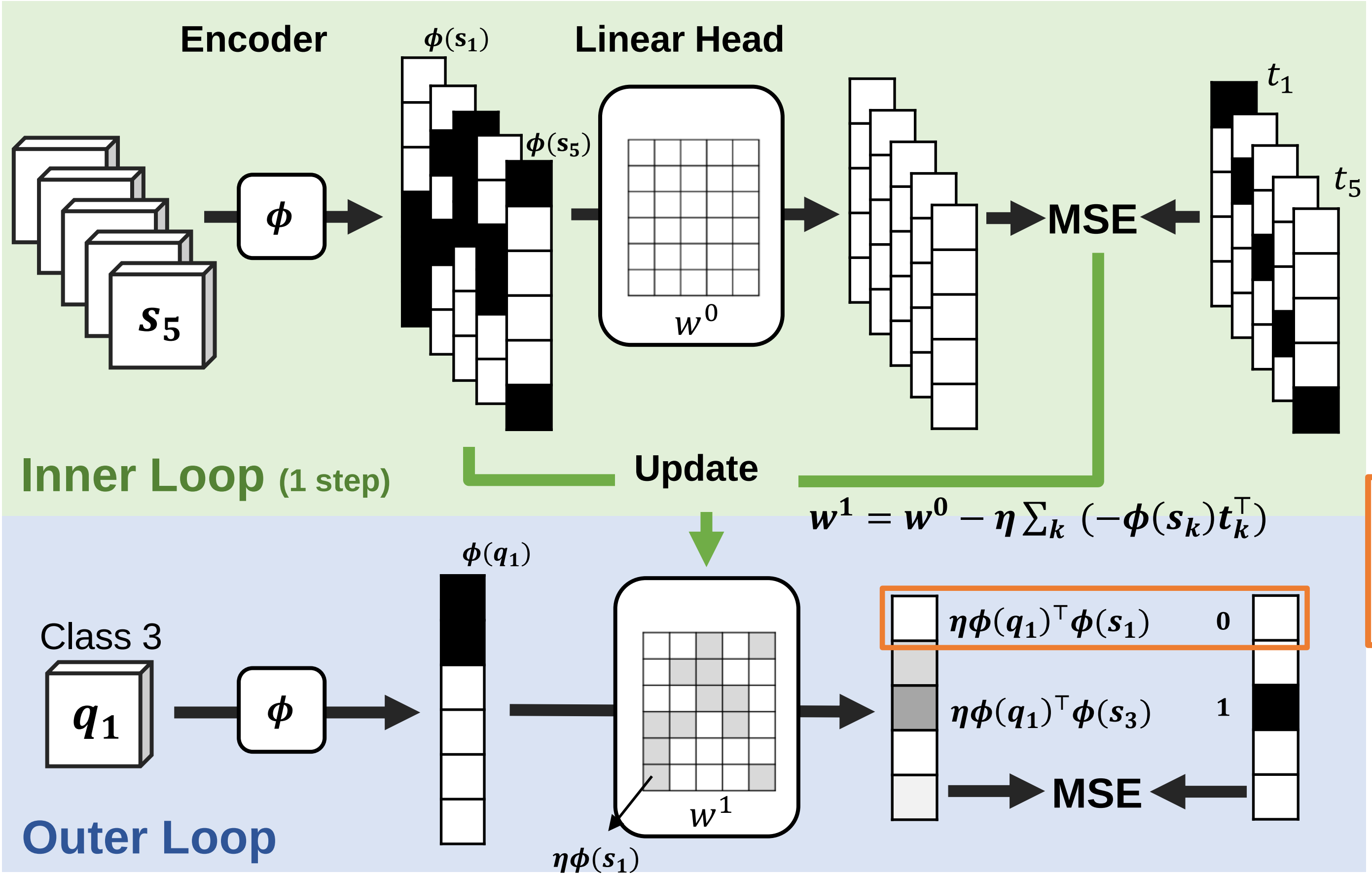
The inner product between support feature s_1 and query feature q_1.


Setting:
5-way 1-shot using MAML with one inner-loop update under MSE loss.
Model:
Image
ϕ
w
Output
Encoder
Linear Layer
0
1
Encoder
$\phi(s_1)$
Linear Head
$\phi(s_5)$
t_1
t_5
s_5
ϕ
w^0
MSE
Inner Loop (1 step)
Update
$w^1 = w^0 - \eta \sum_k (-\phi(s_k) t_k^\top)$
$\phi(q_1)$
Class 3
q_1
ϕ
$\eta\phi(q_1)^\top\phi(s_1)$
0
$\eta\phi(q_1)^\top\phi(s_3)$
1
w^1
MSE
Outer Loop
$\eta\phi(s_1)$

Setting:
5-way 1-shot using MAML with **one** inner-loop update under **MSE loss**.

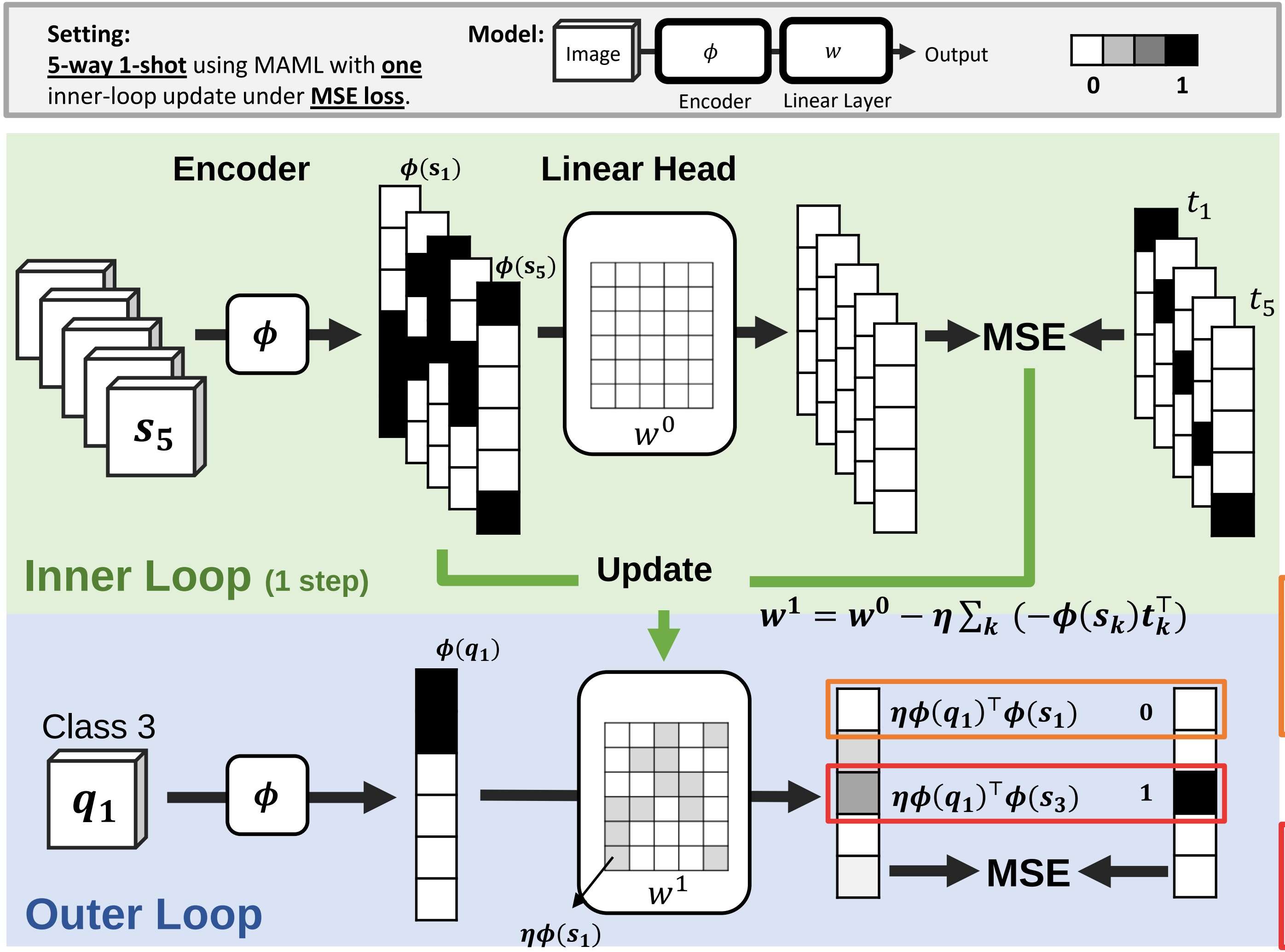
Model:





Negative sample

- q_1 and s_1 have **different labels**
- Their **inner product** of their features should **be zero**.

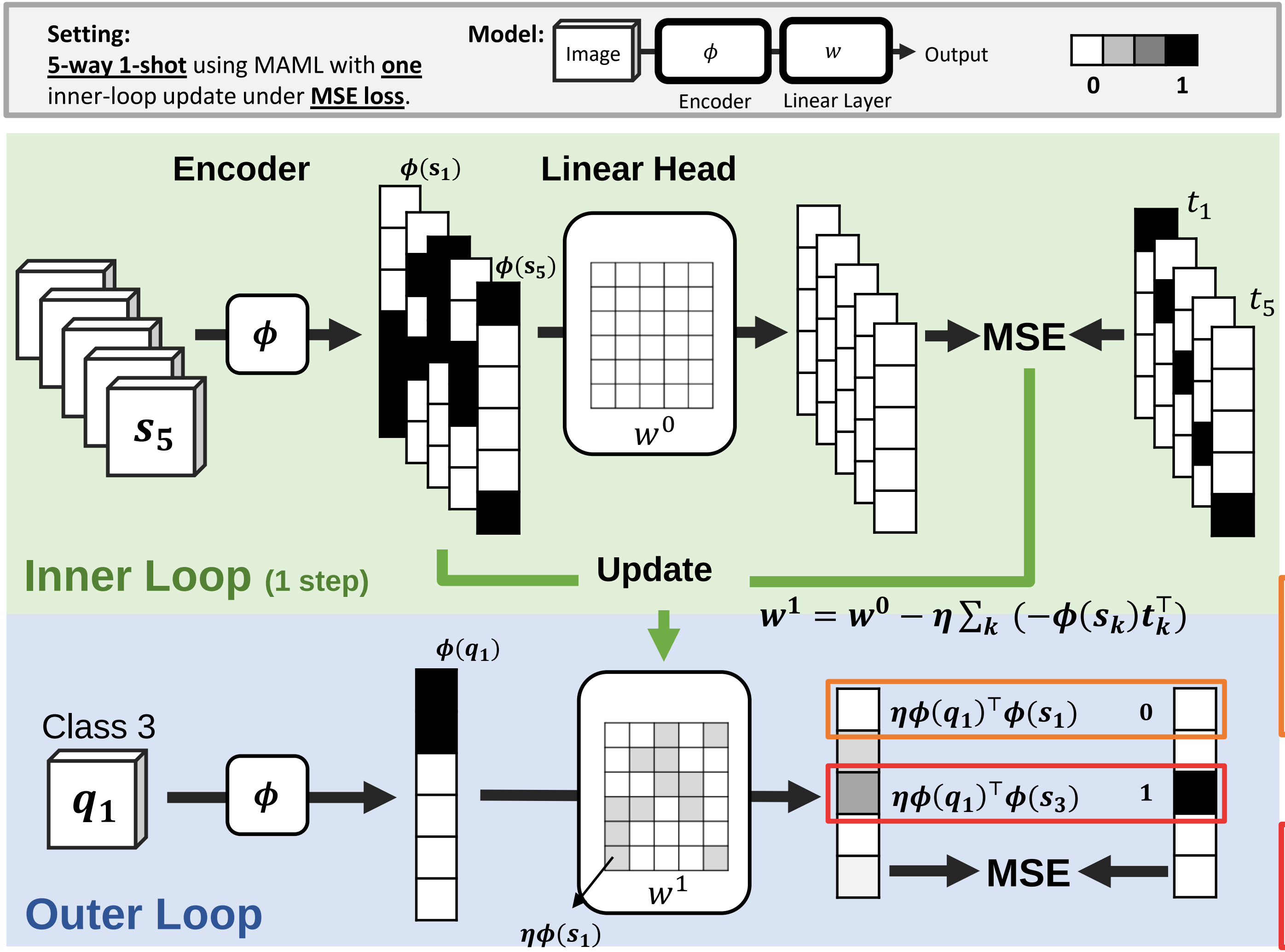


Negative sample

- q_1 and s_1 have **different labels**
- Their **inner product** of their features should **be zero**.

Positive sample

- q_1 and s_3 have **same labels**,
- Their **inner product** of their features should **be one**.



Supervised contrastive learning

Negative sample

- q_1 and s_1 have **different labels**
- Their **inner product** of their features should **be zero**.

Positive sample

- q_1 and s_3 have **same labels**,
- Their **inner product** of their features should **be one**.

Main Derivation
Main result

Consider support data $S = \{(s, t)\}$ and one query data (q, u).

Under **ANIL assumption**, the **loss for the encoder** is :

- First-order MAML:

$$L = \sum_{i=1}^{N_{class}} \underbrace{(\mathrm{q}_i - 1_{i=u})\mathbf{w_i}}_{\text{stop gradient}}^\top \phi(q) + \eta \mathop{\mathbf{E}}_{(s,t)\sim S} \underbrace{\underbrace{[-\sum_{i=1}^{N_{class}} \mathrm{q}_i \mathrm{s}_i + \mathrm{s}_u + \mathrm{q}_t - 1_{t=u}]}_{\text{Contrastive coefficient}} \phi(s)}_{\text{stop gradient}}^\top \phi(q)$$

Contrastive coefficient Inner product

Main Derivation
Main result

Consider support data $S = \{(s,t)\}$ and one query data (q,u).

Under **ANIL assumption**, the **loss for the encoder** is :

- First-order MAML:

$$L = \sum_{i=1}^{N_{class}} \underbrace{(\mathrm{q}_i - 1_{i=u})\mathbf{w_i}^\top}_{\text{stop gradient}} \phi(q) + \eta \mathop{\mathbf{E}}_{(s,t)\sim S} \underbrace{\underbrace{[-\sum_{i=1}^{N_{class}} \mathrm{q}_i \mathrm{s}_i + \mathrm{s}_u + \mathrm{q}_t - 1_{t=u}]}_{\text{Contrastive coefficient (CC)}} \phi(s)^\top}_{\text{stop gradient}} \underbrace{\phi(s)^\top \phi(q)}_{\text{Inner product}}$$

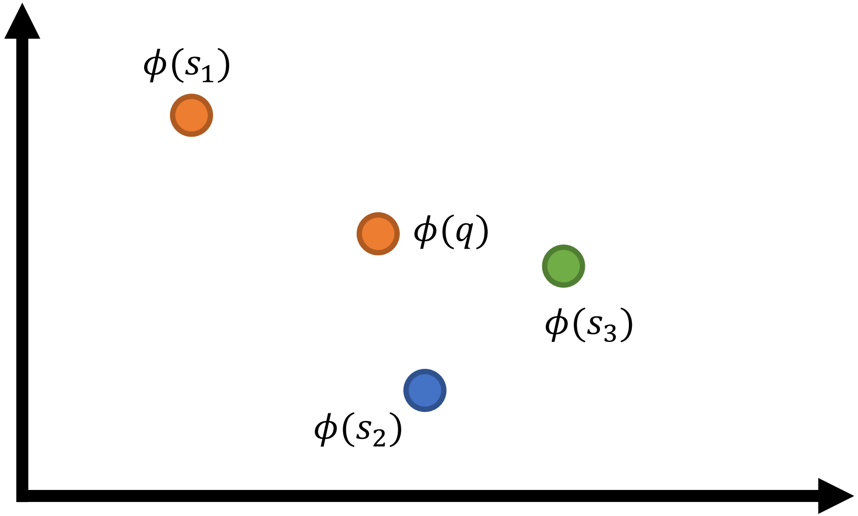
- Expected effect of the second term:
  - If q and s have sample label, then the coefficients is negative, meaning that we are to update ϕ s.t. $\phi(q)$ is closer to $\phi(s)$.
  - If q and s have different labels, then the coefficients should be positive, meaning that we are to update ϕ s.t. $\phi(q)$ is further to $\phi(s)$.

Main Derivation

Main result. Feature space illustration.

Gradient from positive sample
Gradient from negative sample

First-order MAML



update ϕ such that $\phi(q)$ is closer/further to $\phi(s)$

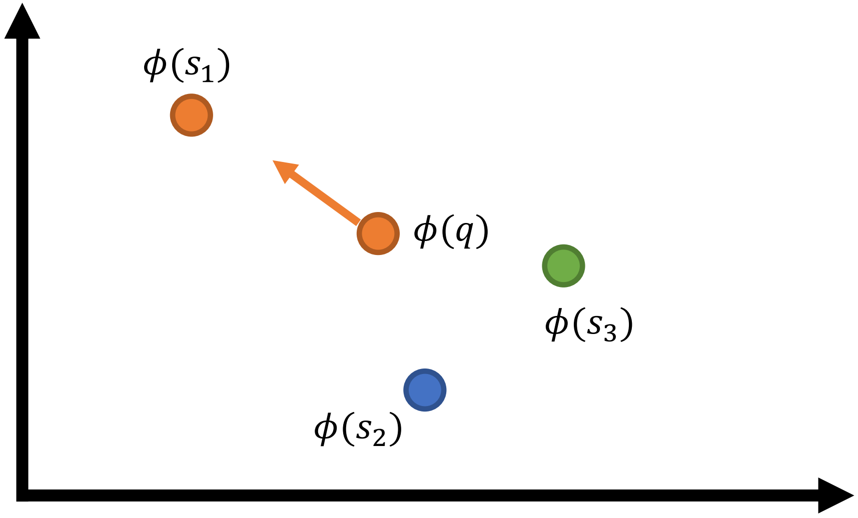
Main Derivation

Main result. Feature space illustration.

Gradient from positive sample

Gradient from negative sample

First-order MAML

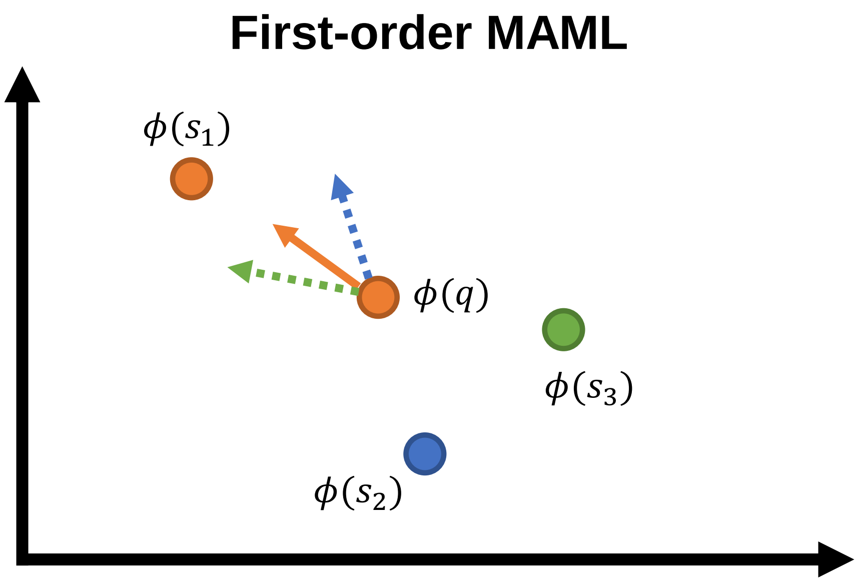


update ϕ such that $\phi(q)$ is closer/further to $\phi(s)$

Main Derivation

Main result. Feature space illustration.

Gradient from positive sample
Gradient from negative sample



update ϕ such that $\phi(q)$ is closer/further to $\phi(s)$

Main Derivation
Main result

Consider support data $S = \{(s, t)\}$ and one query data (q, u).

Under **ANIL assumption**, the **loss for the encoder** is :

- First-order MAML:

$$L = \sum_{i=1}^{N_{class}} \underbrace{(\mathrm{q}_i - 1_{i=u})\mathbf{w_i}}_{\text{stop gradient}}{}^{\top} \phi(q) + \eta \mathop{\mathbf{E}}_{(s,t)\sim S} \underbrace{[-\sum_{i=1}^{N_{class}} \mathrm{q}_i \mathrm{s}_i + \mathrm{s}_u + \mathrm{q}_t - 1_{t=u}]\phi(s)^{\top}}_{\text{stop gradient}} \phi(q)$$

Contrastive coefficient (CC)

Inner product

- Second-order MAML:

$$L = \sum_{i=1}^{N_{class}} \underbrace{(\mathrm{q}_i - 1_{i=u})\mathbf{w_i}}_{\text{stop gradient}}{}^{\top} \phi(q) + \eta \mathop{\mathbf{E}}_{(s,t)\sim S} \underbrace{[-\sum_{i=1}^{N_{class}} \mathrm{q}_i \mathrm{s}_i + \mathrm{s}_u + \mathrm{q}_t - 1_{t=u}]}_{\text{stop gradient}} \phi(s)^{\top}\phi(q)$$

Contrastive coefficient

Inner product

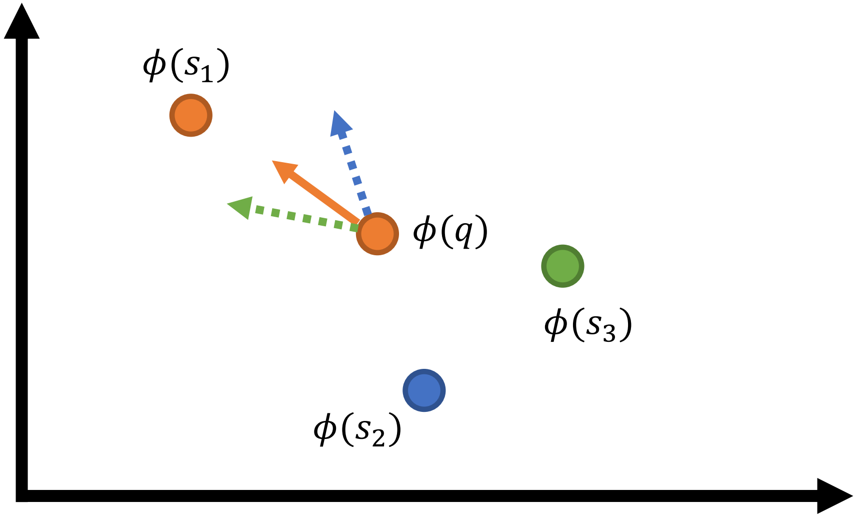
Main Derivation

Main result. Feature space illustration.

Gradient from positive sample

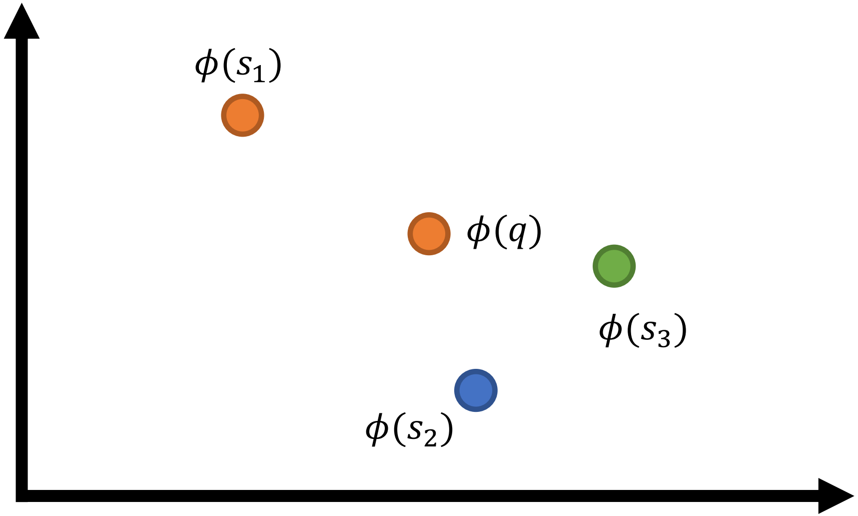
Gradient from negative sample

First-order MAML



update ϕ such that $\phi(q)$ is closer/further to $\phi(s)$

Second-order MAML



update ϕ such that $\phi(q)$ and $\phi(s)$ move closer/further to each other.

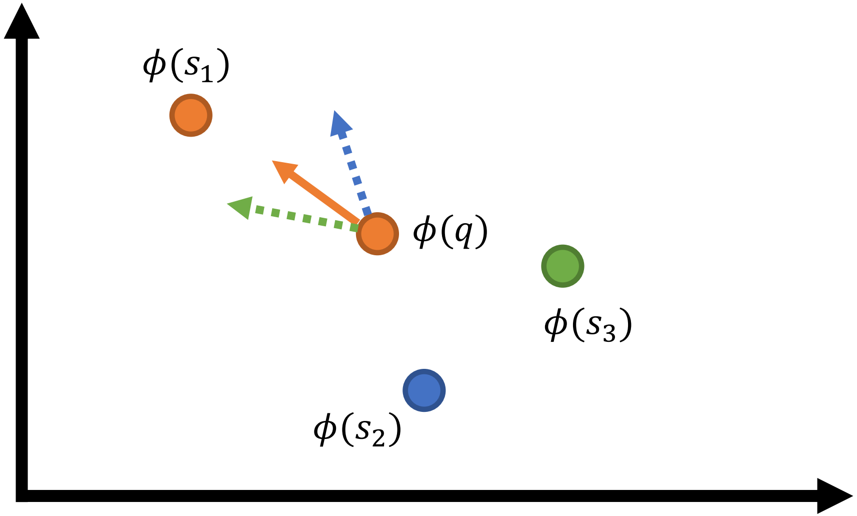
Main Derivation

Main result. Feature space illustration.

Gradient from positive sample

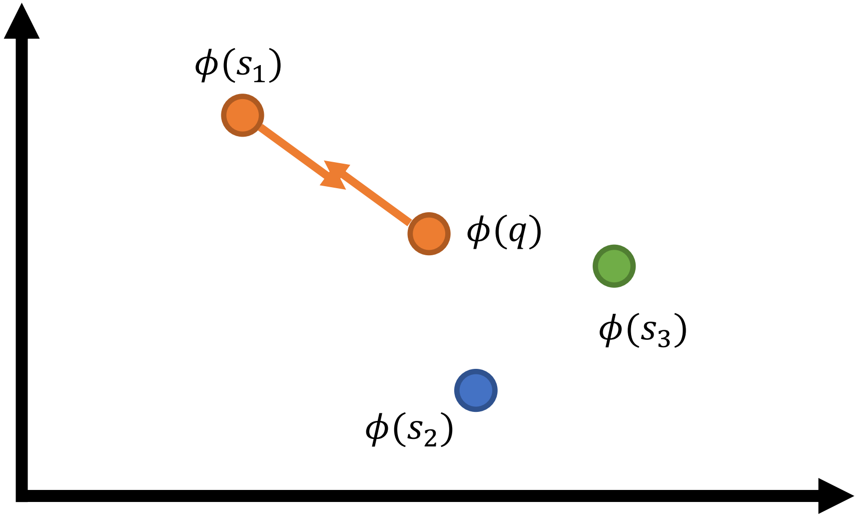
Gradient from negative sample

First-order MAML



update ϕ such that $\phi(q)$ is closer/further to $\phi(s)$

Second-order MAML



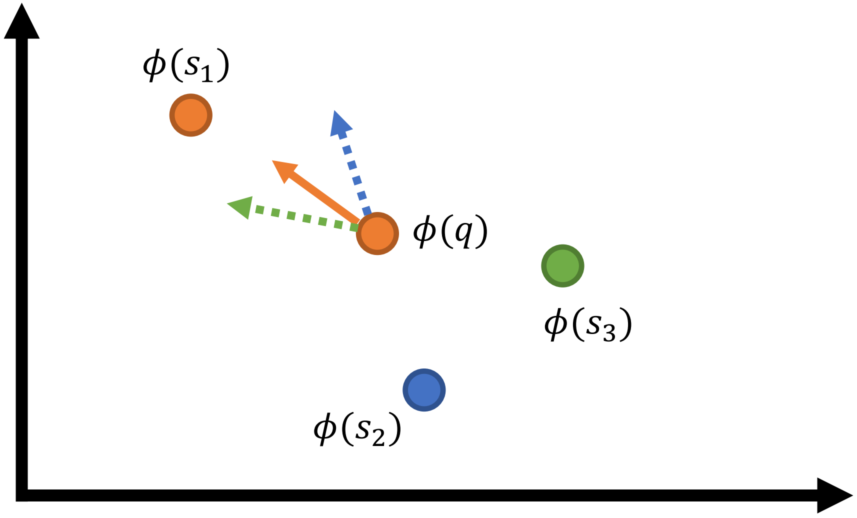
update ϕ such that $\phi(q)$ <u>and</u> $\phi(s)$ move closer/further to <u>each other</u>.

Main Derivation

Main result. Feature space illustration.

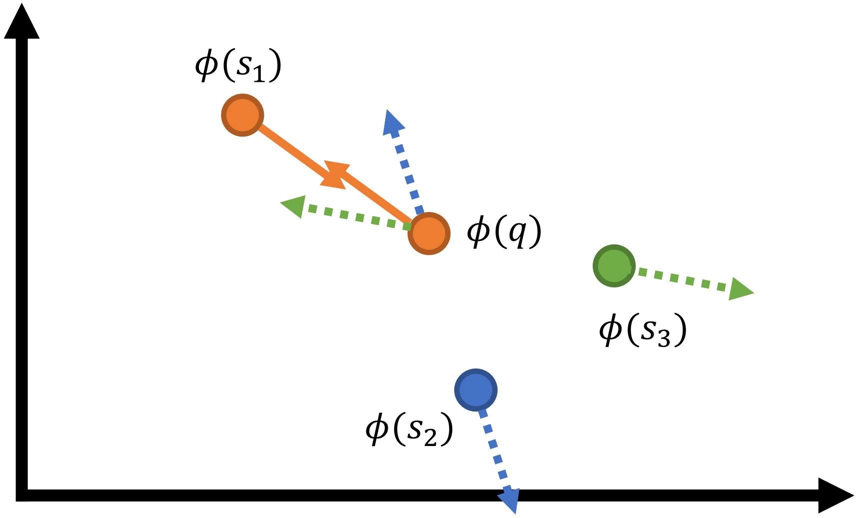
Gradient from positive sample
Gradient from negative sample

First-order MAML



update ϕ such that $\phi(q)$ is closer/further to $\phi(s)$

Second-order MAML



update ϕ such that $\phi(q)$ and $\phi(s)$ move closer/further to each other.

Main Derivation

Main result

Consider support data $S = \{(s,t)\}$ and one query data (q,u).

Under **ANIL assumption**, the **loss for the encoder** is :

- First-order MAML:

$$L = \sum_{i=1}^{N_{class}} \underbrace{(\mathrm{q}_i - 1_{i=u})\mathbf{w_i}^\top}_{\text{stop gradient}} \phi(q) + \eta \mathop{\mathbf{E}}_{(s,t)\sim S} \underbrace{[- \sum_{i=1}^{N_{class}} \mathrm{q}_i \mathrm{s}_i + \mathrm{s}_u + \mathrm{q}_t - 1_{t=u}]}_{\text{stop gradient}} \phi(s)^\top \phi(q)$$

- Random initialization
- Cross task interference

- The coefficient is not always positive when s and q come from different classes

Theorem 1 *With the assumption of (a) no inner loop update of the encoder, FOMAML is a noisy SCL algorithm. With assumptions of (a) no inner loop update of the encoder and (b) a single inner-loop update, SOMAML is a noisy SCL algorithm.*

Main Derivation

Main result. Introducing the zeroing trick.

Consider support data $S = \{(s, t)\}$ and one query data (q, u).

Under **ANIL assumption** and **the zeroing trick**, the **loss for the encoder** is :

- First-order MAML:

$$L = \eta \mathop{\mathbf{E}}_{(s,t)\sim S} \underbrace{(\mathrm{q}_t - {}_{\mathrm{t=u}})\phi(s)^\top}_{\text{stop gradient}} \phi(q)$$

- Second-order MAML:

$$L = \eta \mathop{\mathbf{E}}_{(s,t)\sim S} \underbrace{(\mathrm{q}_t - 1_{t=u})}_{\text{stop gradient}} \phi(s)^\top \phi(q)$$

Corollary 1 *With mild assumptions of (a) no inner loop update of the encoder, (b) a single inner-loop update and (c) training with the zeroing trick (i.e., the linear layer is zeroed at the end of each outer loop), both FOMAML and SOMAML are SCL algorithms.*

Main Derivation

Main result. Introducing the zeroing trick.

Algorithm 1 Second-order MAML

while not done **do**
  Sample tasks $\{T_1, T_2\}$

  for $n = 1, 2$ **do**
    $\{S_n, Q_n\} \leftarrow$ sample from T_n
    $\theta_n = \theta$
    for $i = 1, 2, \ldots, N_{step}$ **do**
      $\theta_n \leftarrow \theta_n - \eta \nabla_{\theta_n} L(\theta_n, S_n)$
    end for
  end for
  Update $\theta \leftarrow \theta - \rho \sum_{n=1}^{N_{batch}} \nabla_\theta L(\theta_n, Q_n)$

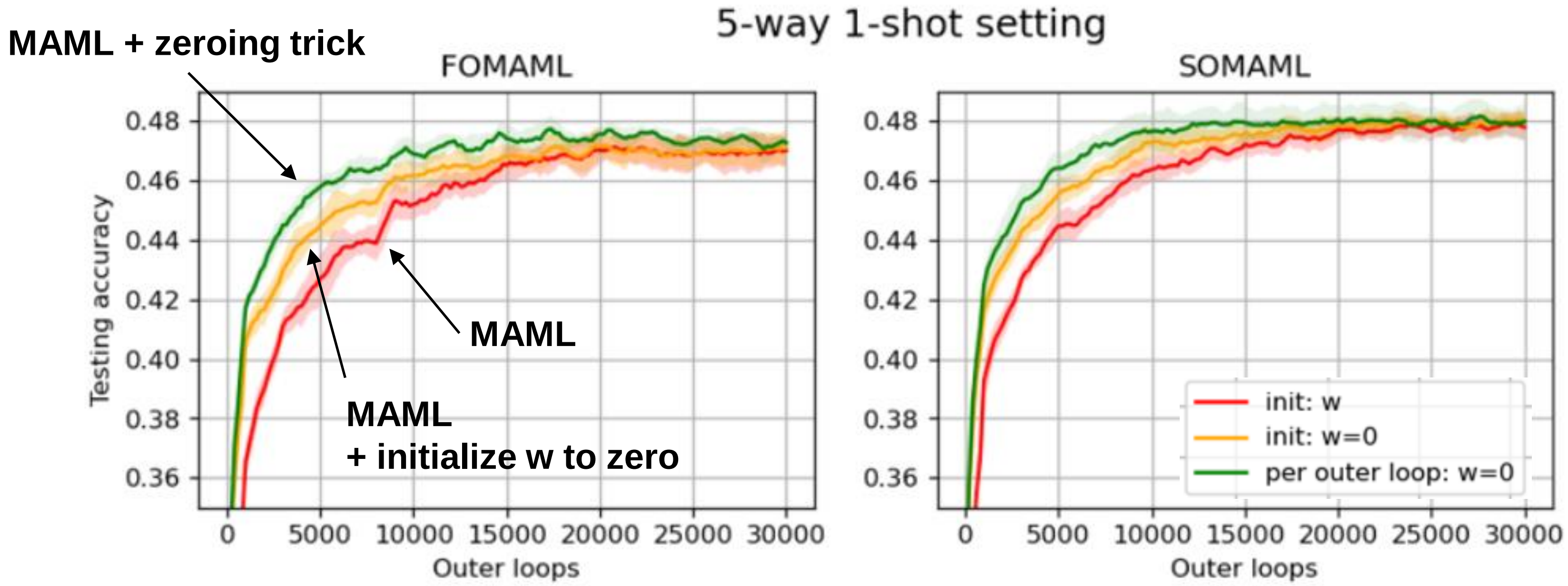
end while

Algorithm 2 Second-order MAML with zeroing trick

while not done **do**
  Sample tasks $\{T_1, T_2\}$
  $w = 0$
  for $n = 1, 2$ **do**
    $\{S_n, Q_n\} \leftarrow$ sample from T_n
    $\theta_n = \theta$
    for $i = 1, 2, \ldots, N_{step}$ **do**
      $\theta_n \leftarrow \theta_n - \eta \nabla_{\theta_n} L(\theta_n, S_n)$
    end for
  end for
  Update $\theta \leftarrow \theta - \rho \sum_{n=1}^{N_{batch}} \nabla_\theta L(\theta_n, Q_n)$
  $w = 0$
end while

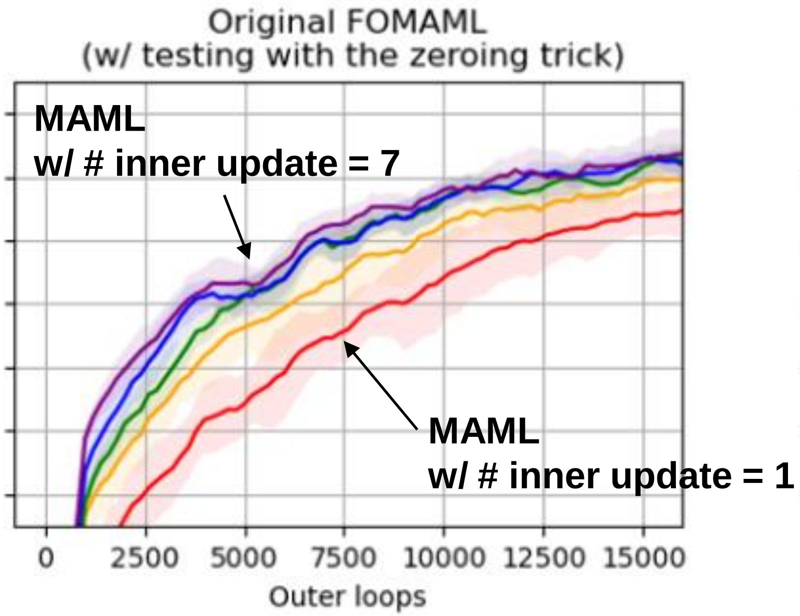
Results
Using the zeroing trick improves performance.

Setting: MiniImageNet.



Results
With zeroing trick, # of inner loop steps no longer matters

Setting: MiniImageNet 5-way 1-shot.

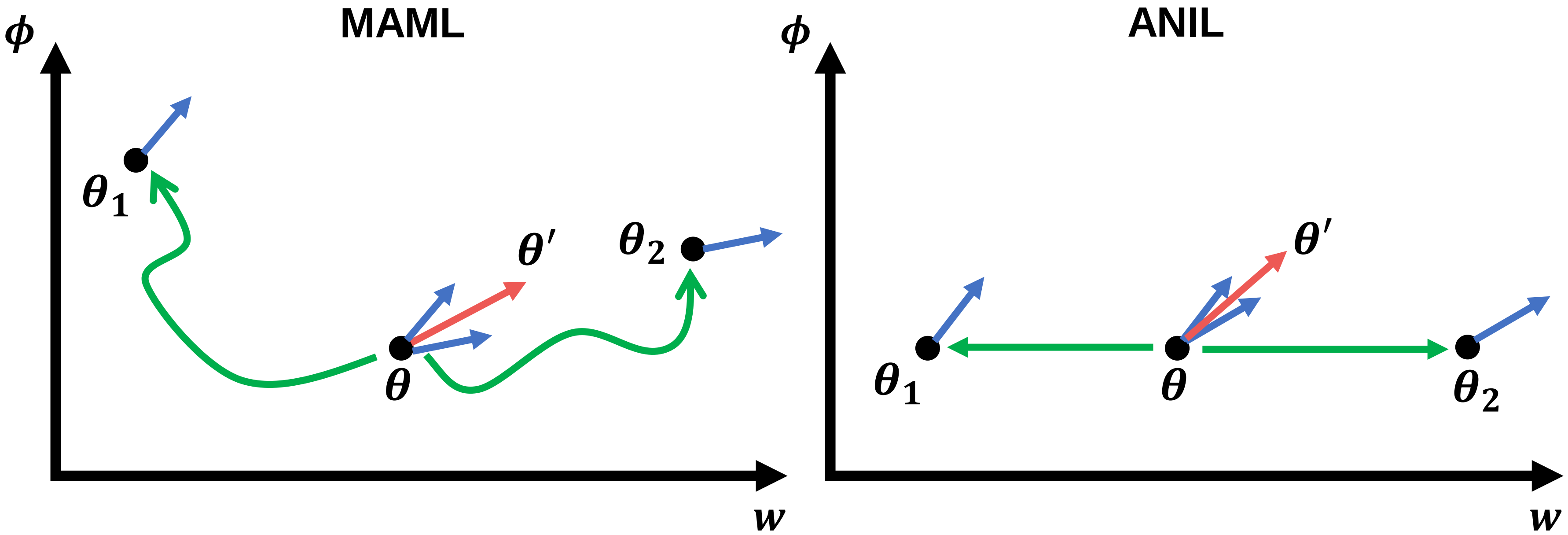


Wrap up

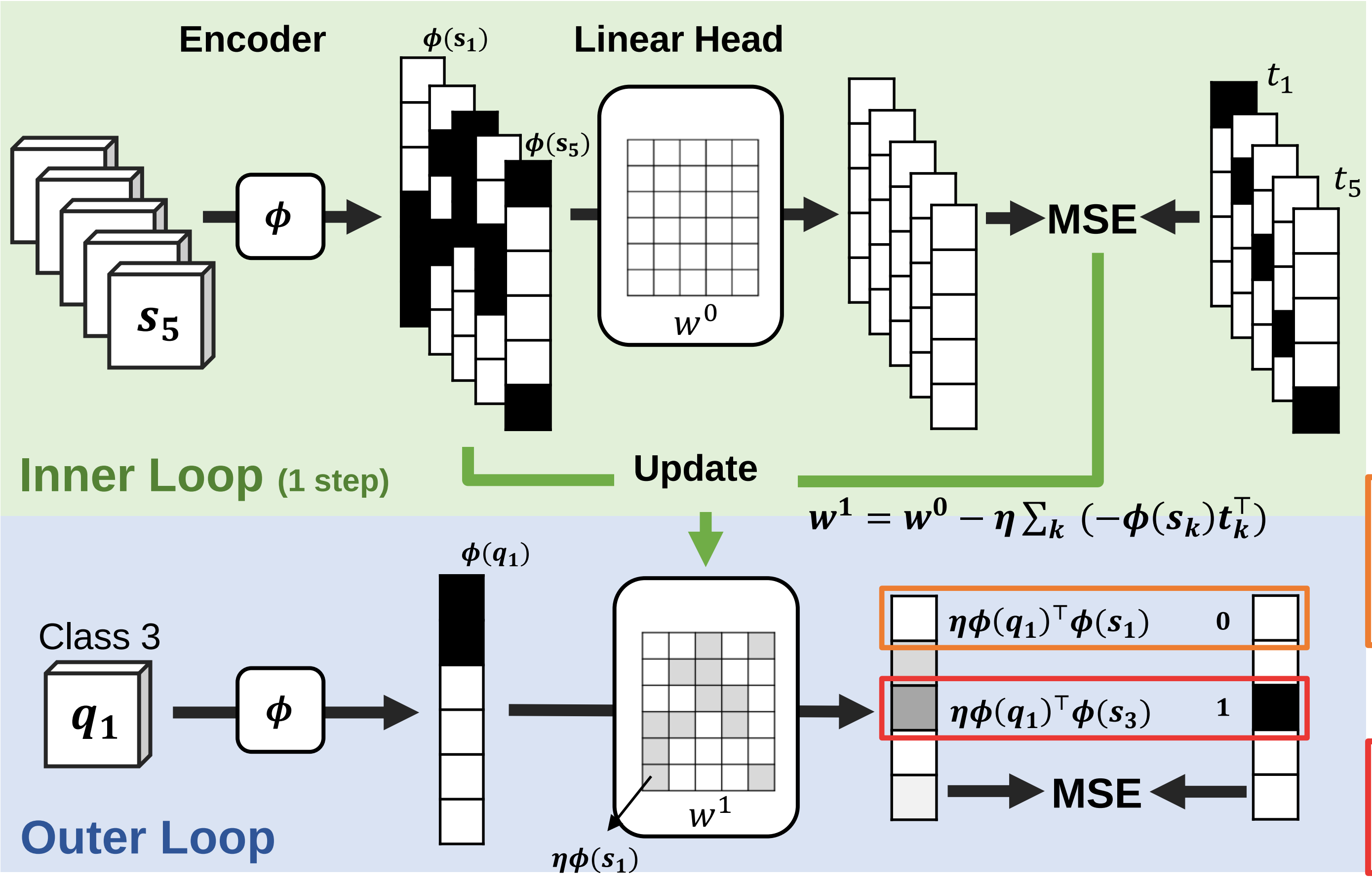
Wrap up
We use the ANIL assumption for derivation.

Consider a model $\boldsymbol{\theta} = \{\boldsymbol{\phi}, \boldsymbol{w}\}$, where $\boldsymbol{\phi}$ is an encoder and $\boldsymbol{w}$ is a linear classifier.

ANIL states that **<u>the encoder $\boldsymbol{\phi}$ is not updated during the inner loop</u>**.



Wrap up



Supervised contrastive learning

Negative sample

- q_1 and s_1 have <u>different labels</u>
- Their <u>inner product</u> of their features should <u>be zero</u>.

Positive sample

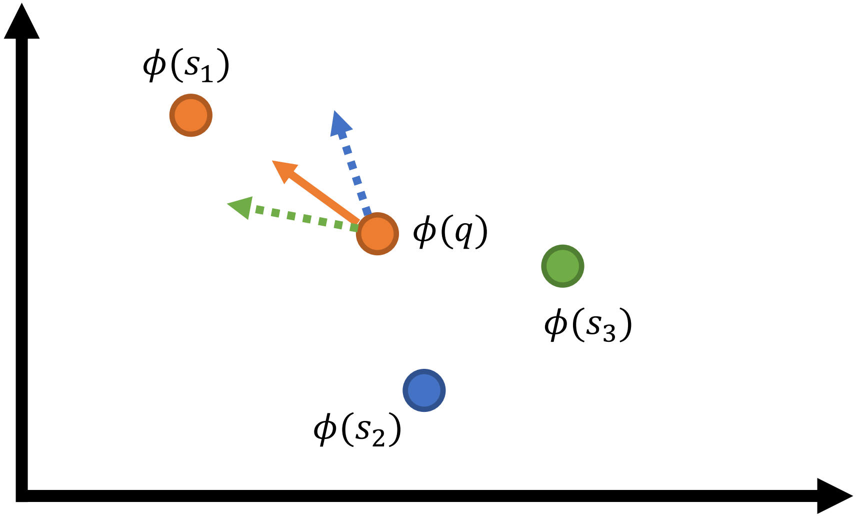
- q_1 and s_3 have <u>same labels</u>,
- Their <u>inner product</u> of their features should <u>be one</u>.

Wrap up

We show how FOMAML different from SOMAML.

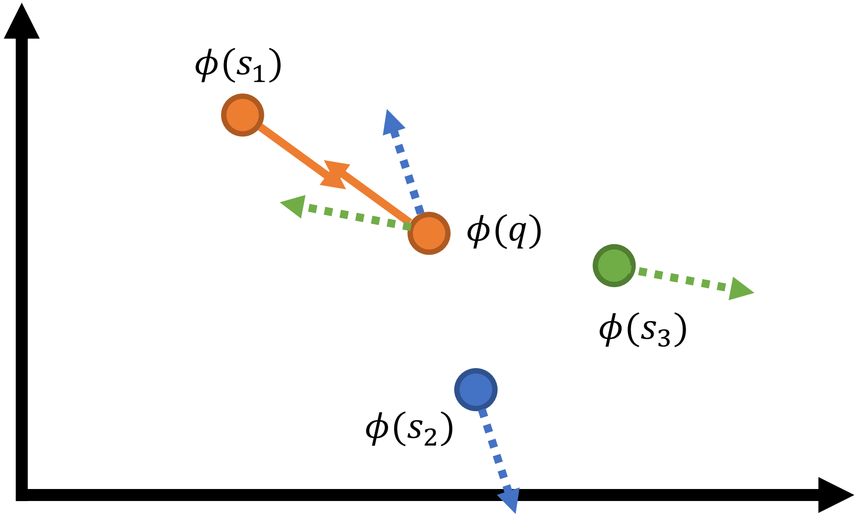
Gradient from positive sample
Gradient from negative sample

First-order MAML



update ϕ such that $\phi(q)$ is closer/further to $\phi(s)$

Second-order MAML



update ϕ such that $\phi(q)$ <u>and</u> $\phi(s)$ move closer/further to <u>each other</u>.

Wrap up

We show that the zeroing trick improves MAML.

Setting: MiniImageNet.