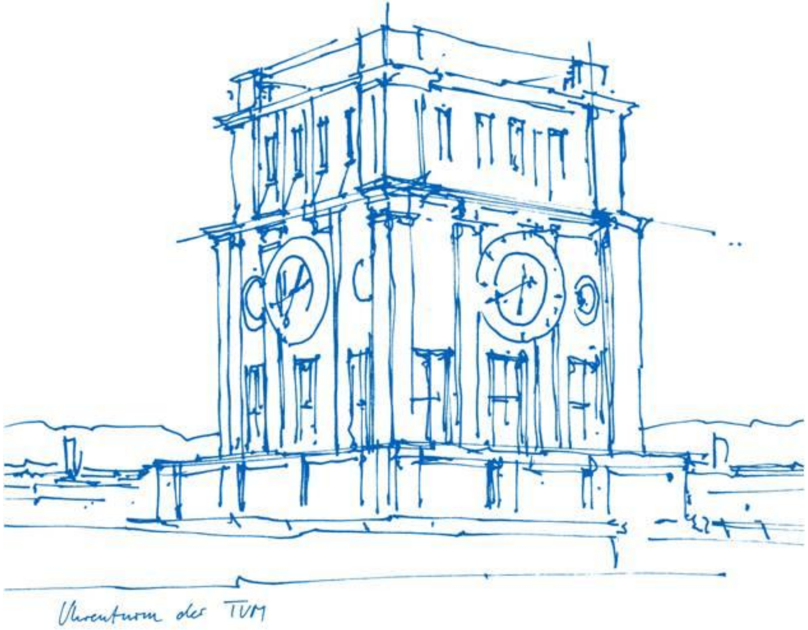
Self-Supervised Training of Interpretable Neural Networks for Medical Applications

Michelle Espranita Liman

April 15, 2024



Agenda

1. Motivation
2. Method
3. Results
4. Conclusion

💪 Motivation

Problem #1

💊 Although Deep Learning is advancing rapidly, its adoption in the medical field has been **slow**.

Most neural networks are **black-box** models. → We don’t understand how they make predictions.

💡 Neural networks need to be **interpretable**!

Problem #2

🏷️ Labelling medical datasets is **laborious** and **expensive**, in terms of time and money.

Supervised learning is possible only on a small, labelled dataset.

💡 We need to leverage large, unlabelled datasets using **self-supervised learning**!

Goal

Build a neural network that:

1. is **interpretable** via the classification head
2. uses **self-supervised learning** to leverage large, unlabelled datasets

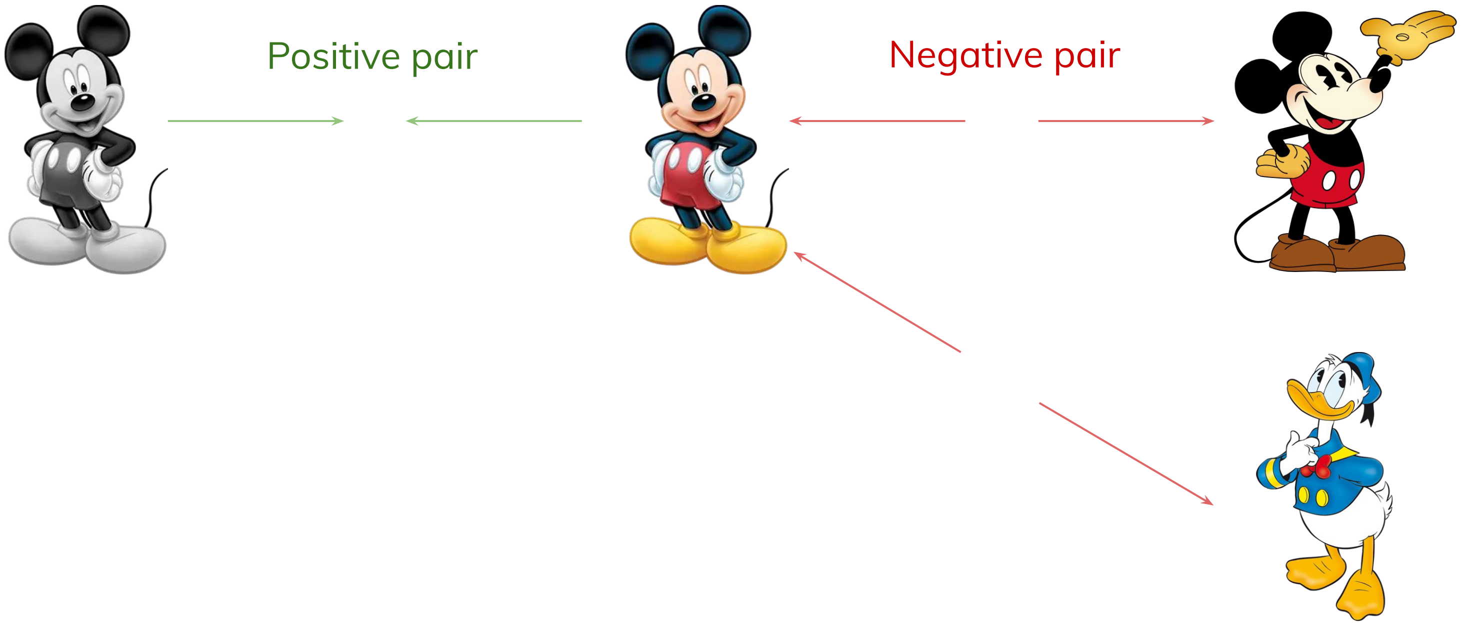
➡️ We propose two methods: **PCL-ProtoPNet** and **PCL-NW.**

➡️ We evaluate our methods on the **Alzheimer's Disease classification** (AD vs. MCI vs. CN) task.

🧠 Method

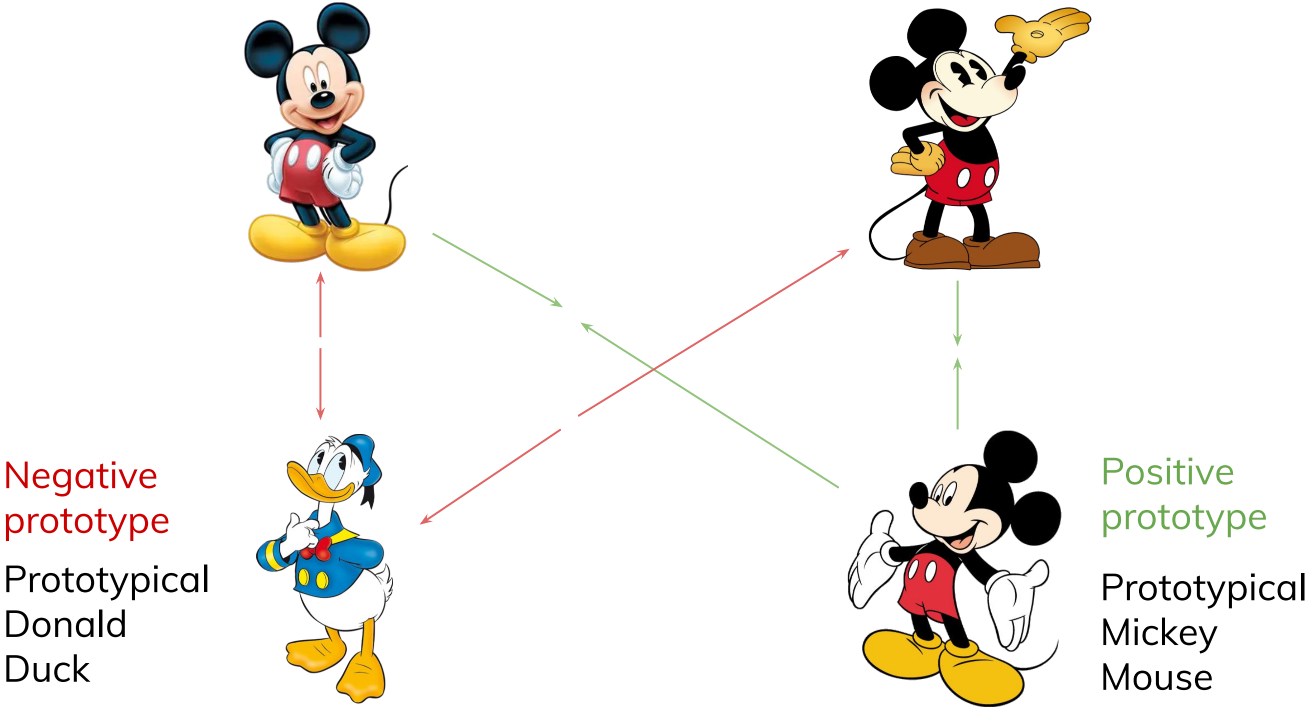
Prototypical Contrastive Learning (PCL)

Instance-wise Contrastive Learning



Prototypical Contrastive Learning (PCL)

Prototypical Contrastive Learning



Prototypical Contrastive Learning (PCL)

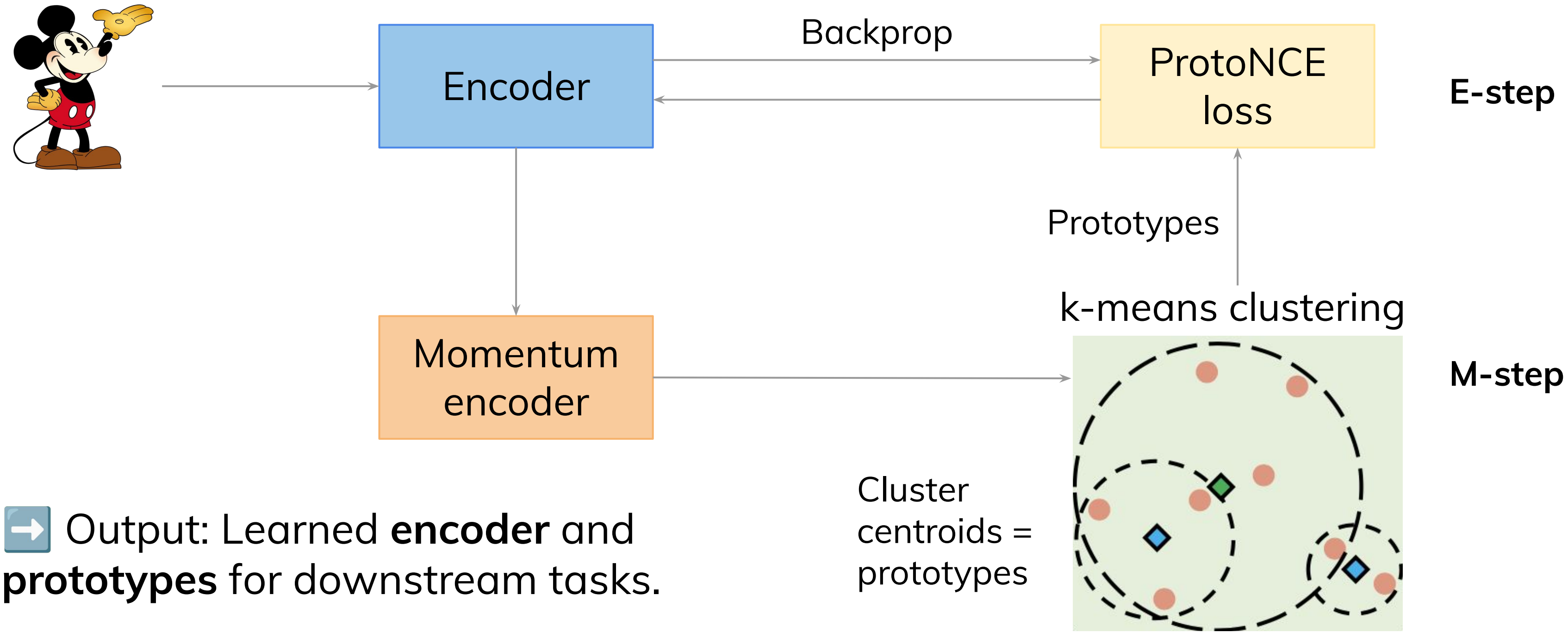
Positive prototype

$$\mathcal{L}_{\text{ProtoNCE}} = \sum_{i=1}^{n} \underbrace{\boxed{-\left(\log \frac{\exp(v_i \cdot v_i' / \tau)}{\sum_{j=0}^{r} \exp(v_i \cdot v_j' / \tau)}\right.}}_{\text{InfoNCE loss}} \left. + \frac{1}{M} \sum_{m=1}^{M} \log \frac{\exp(v_i \cdot c_s^m / \phi_s^m)}{\sum_{j=0}^{r} \exp(v_i \cdot c_j^m / \phi_j^m)} \right)$$

Negative prototype

PCL learns prototypes **without** labels!
(self-supervised)

Prototypical Contrastive Learning (PCL)



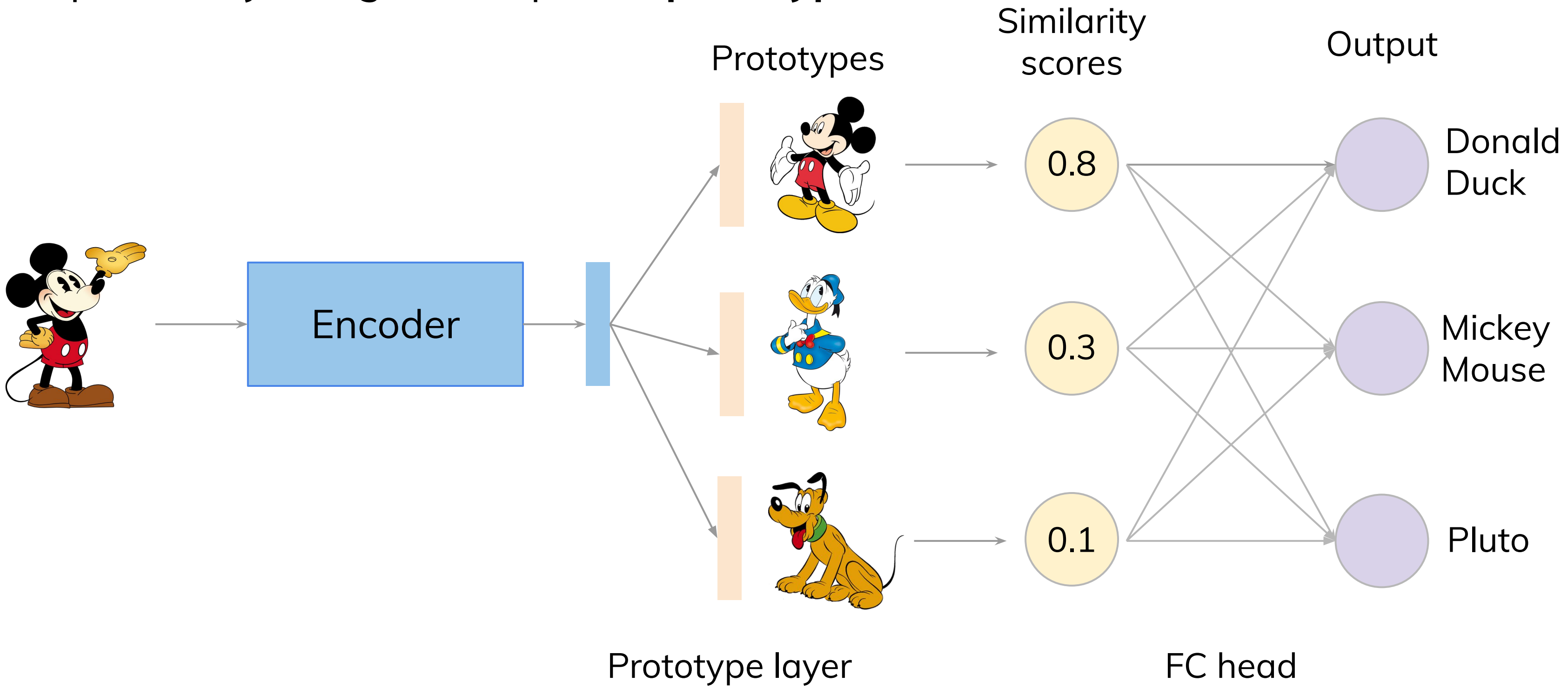
➡️ Output: Learned **encoder** and **prototypes** for downstream tasks.

Prototypical Contrastive Learning (PCL)

(-) Does **not** provide **interpretability** because the prototypes cannot be visualized!

ProtoPNet

Interpretability using class-specific **prototypes**



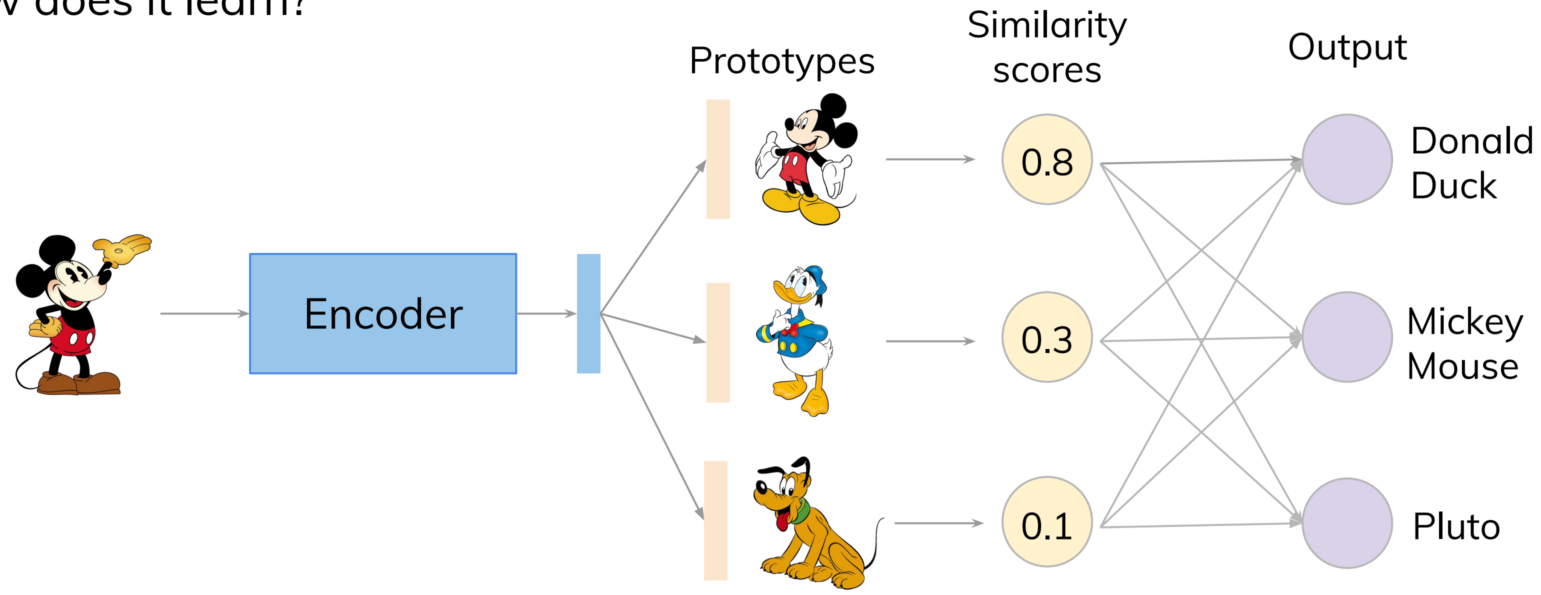
ProtoPNet

How is it interpretable?

Prototypes of Mickey Mouse	Similarity scores	Class connections	Points contributed
	0.9	1.5	1.35
	0.6	1.2	0.72
	0.8	1.3	1.04 +
			3.11

ProtoPNet

How does it learn?



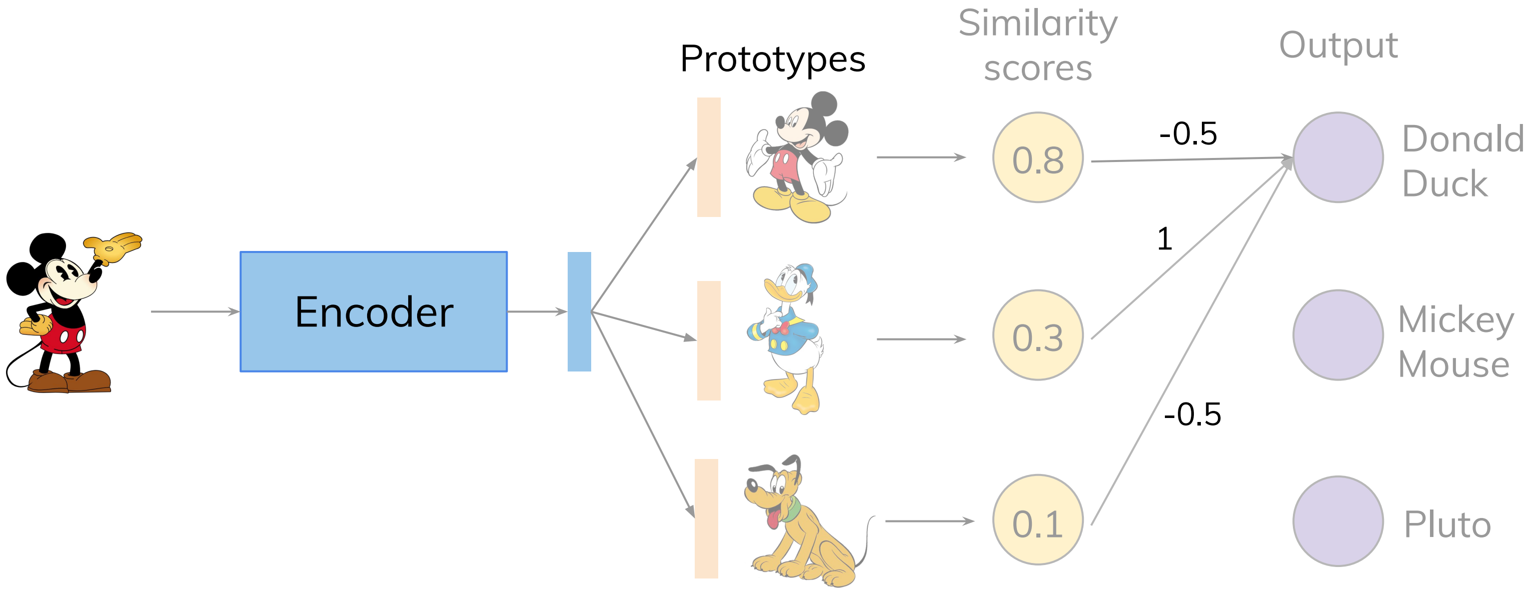
1) Train encoder & learn prototypes

2) Projection of prototypes

3) Train FC head

ProtoPNet

The FC head is **fixed.**

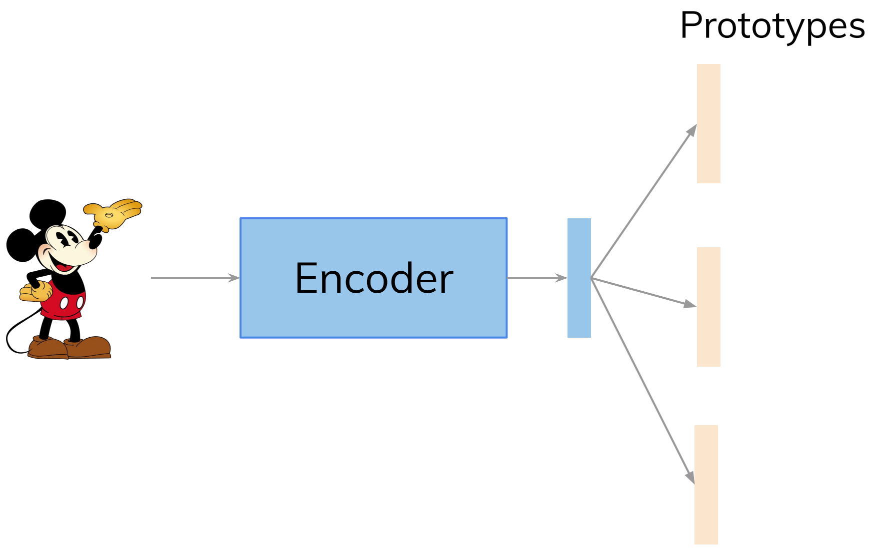


1) Train encoder & learn prototypes

2) Projection of prototypes

3) Train FC head

ProtoPNet

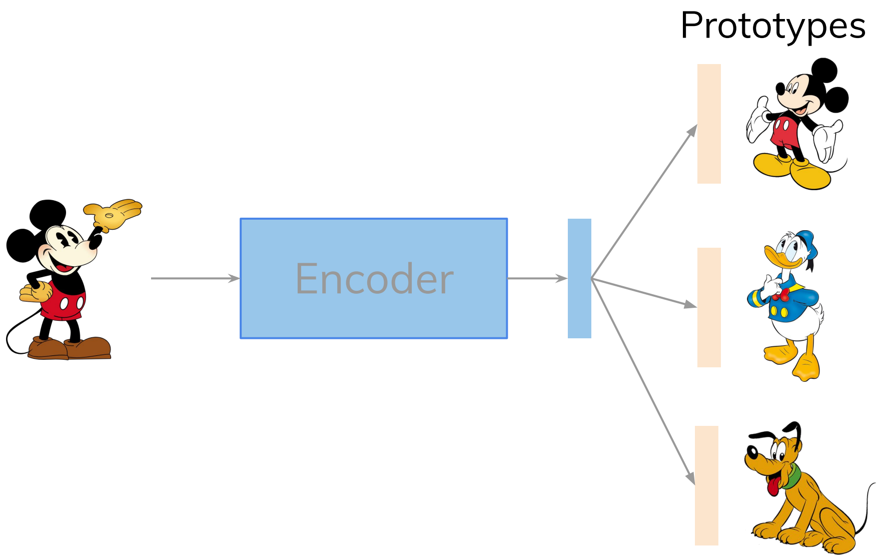


1) Train encoder & learn prototypes

Unlike PCL, we need **labels** to train the encoder and learn the prototypes.

➡️ Output: Learned **encoder** and **prototypes** (do not represent any image)

ProtoPNet



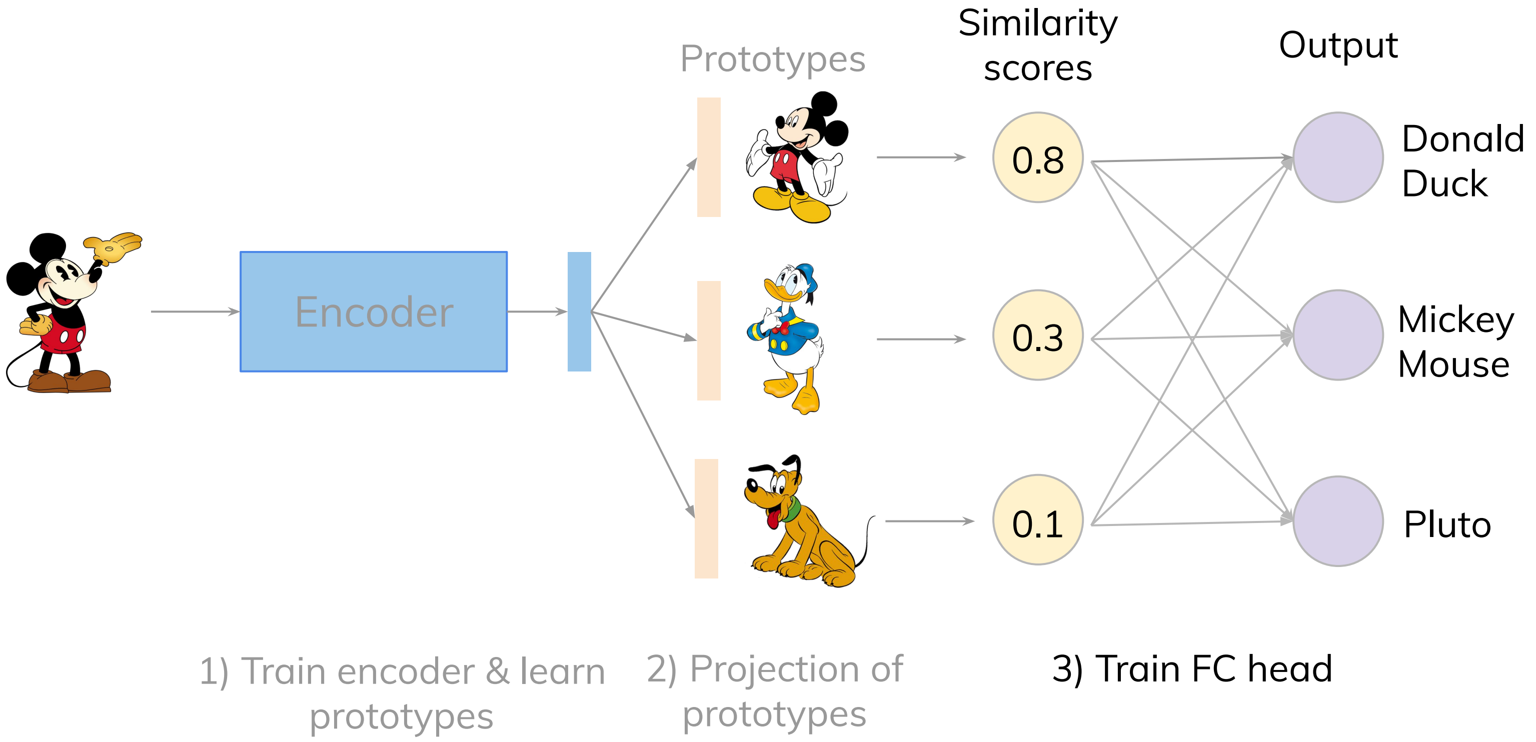
1) Train encoder & learn prototypes

2) Projection of prototypes

Calculate the cosine **similarity** between each prototype and all train images

➡️ Output: **Prototypes** that correspond to images

ProtoPNet

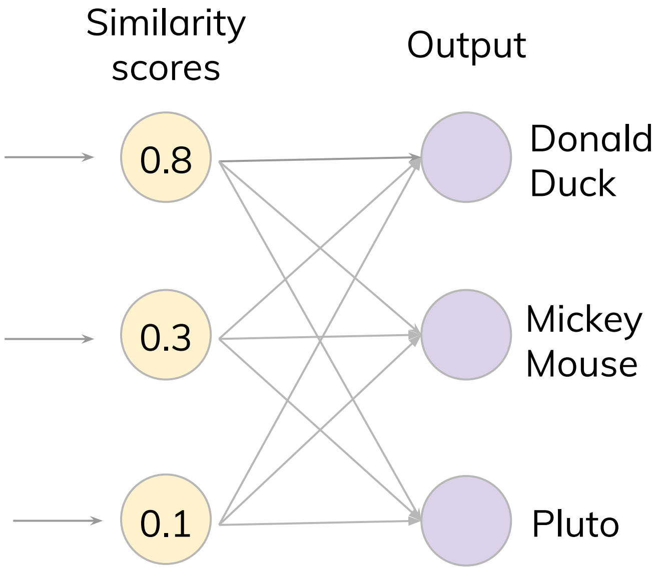


Prototypes
Similarity scores
Output
Encoder
0.8
0.3
0.1
Donald Duck
Mickey Mouse
Pluto
1) Train encoder & learn prototypes
2) Projection of prototypes
3) Train FC head

ProtoPNet

- The encoder and prototypes are **fixed.**
- Cross Entropy loss

➡️ Output: Trained ProtoPNet



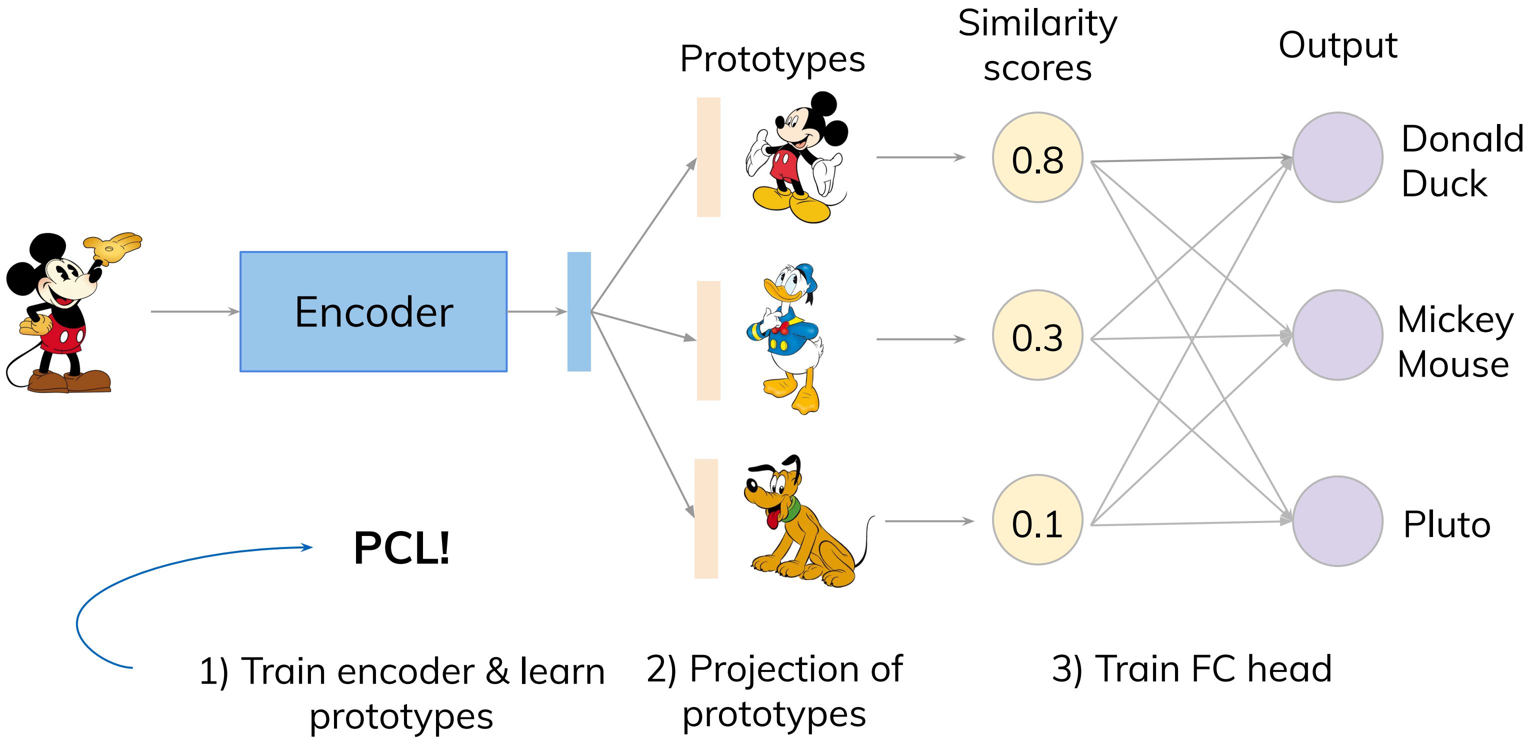
3) Train FC head

ProtoPNet

(-) Supervised learning requires **labels**!

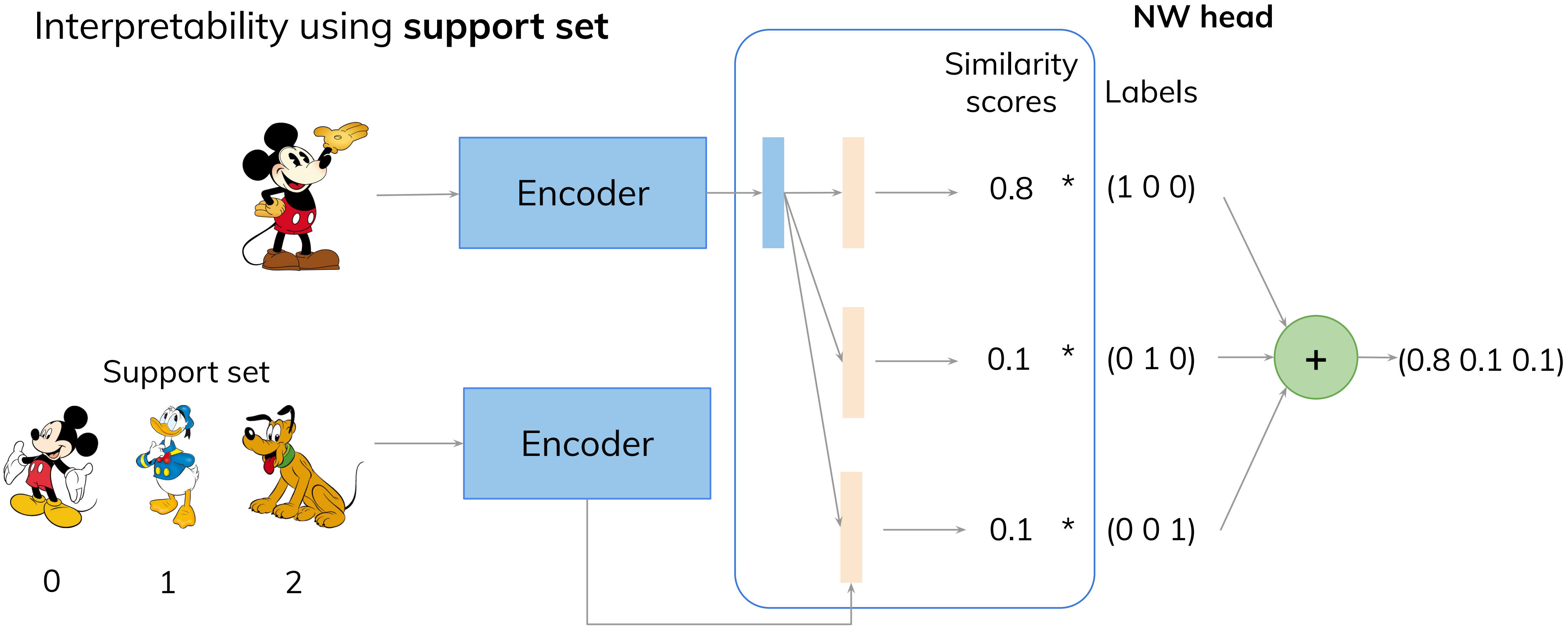
✨ PCL-ProtoPNet ✨

Combines PCL's **self-supervised learning** and ProtoPNet's **interpretability.**



Nadaraya-Watson (NW) Head

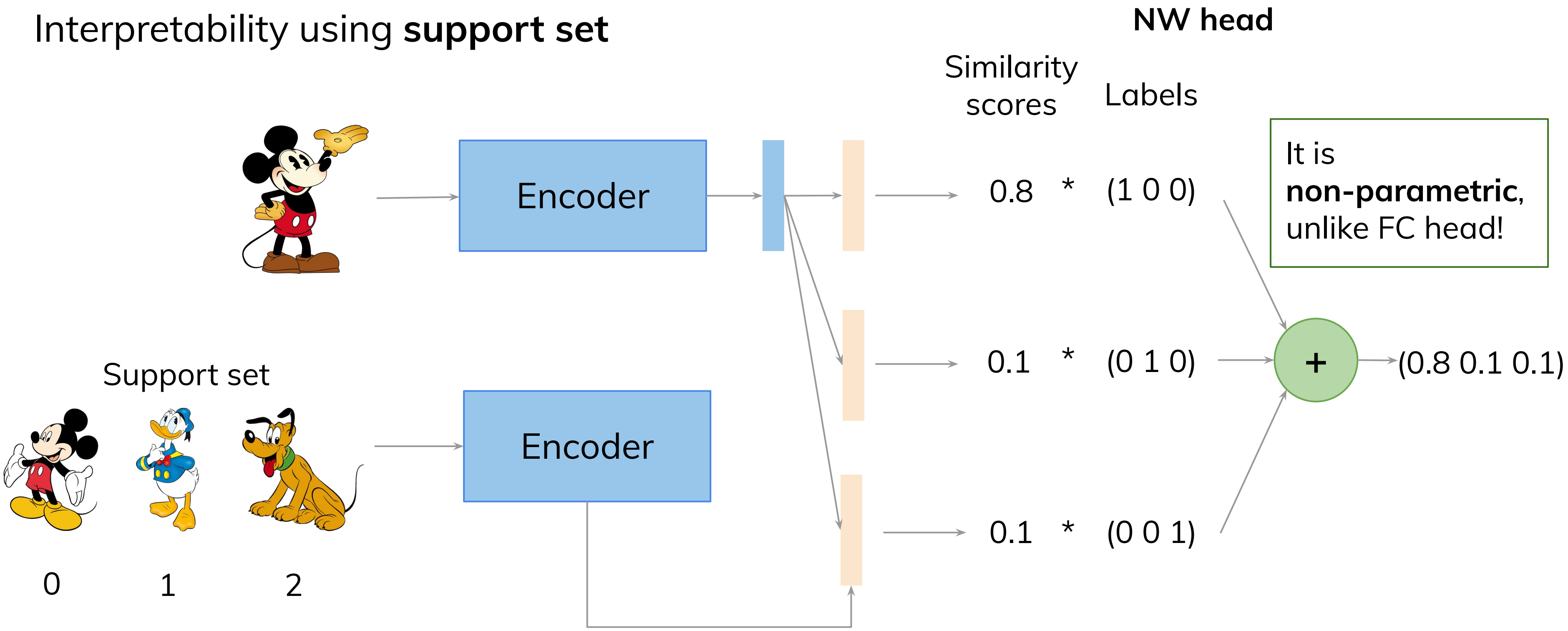
Interpretability using **support set**



Instead of prototypes, support samples

Nadaraya-Watson (NW) Head

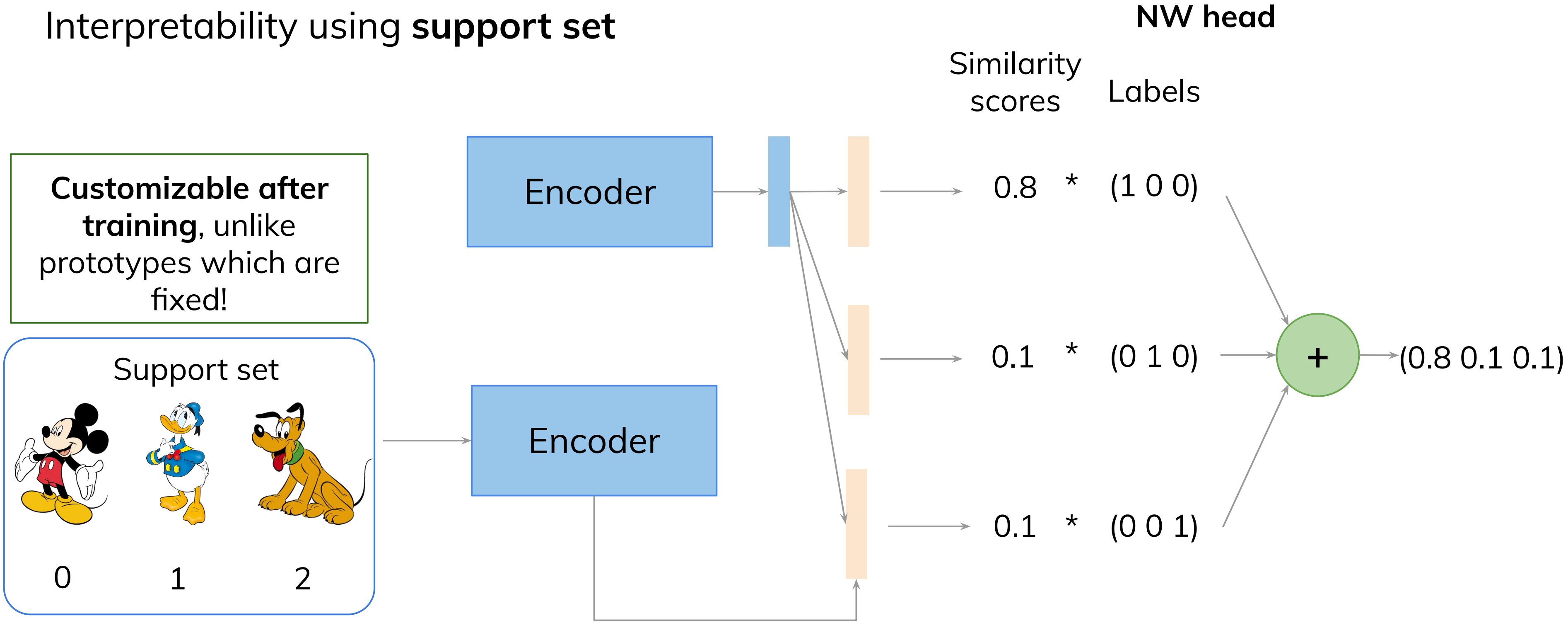
Interpretability using **support set**



Nadaraya-Watson (NW) Head

Interpretability using **support set**

Customizable after training, unlike prototypes which are fixed!



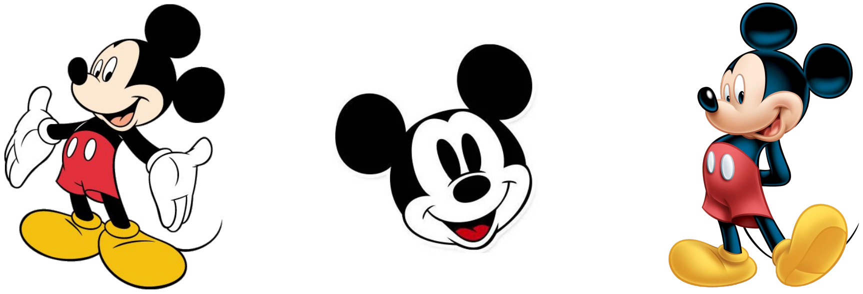
Nadaraya-Watson (NW) Head

How is it interpretable?

Query

Prediction:
Mickey Mouse

Most similar support samples to query

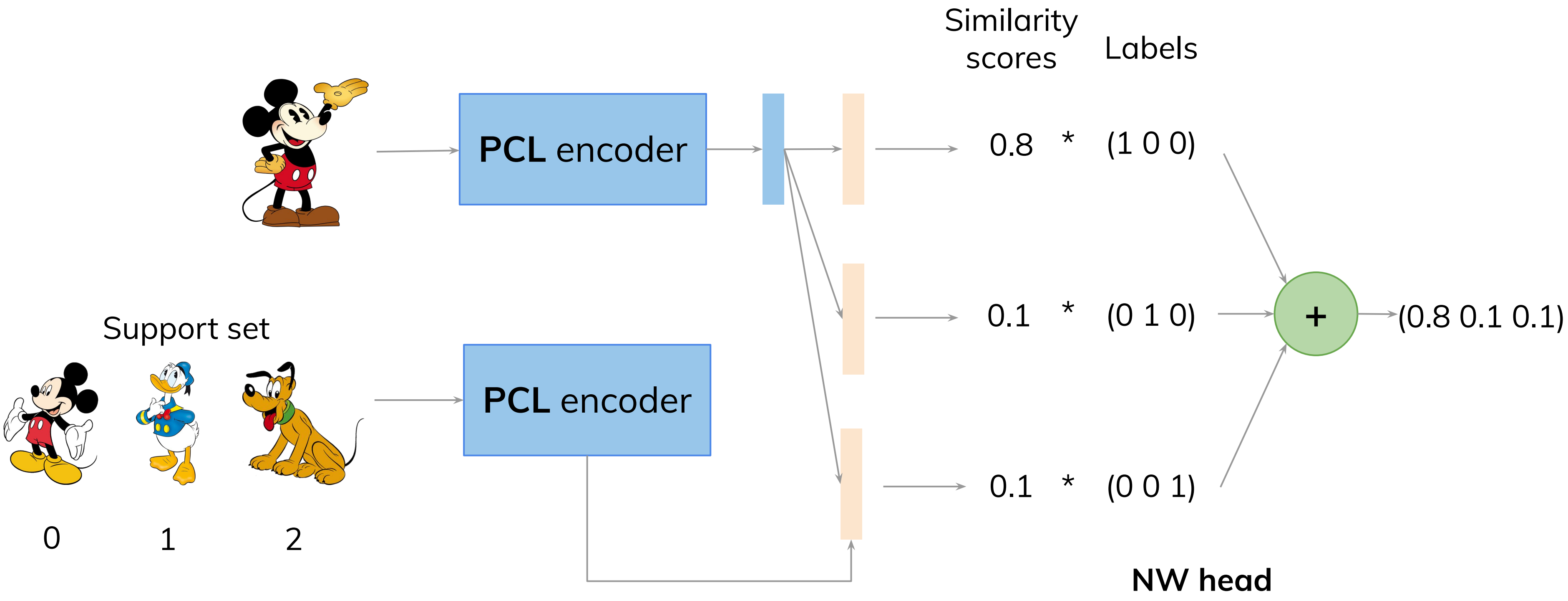


Nadaraya-Watson (NW) Head

(-) Like ProtoPNet, supervised learning requires **labels**!

✨ PCL-NW ✨

Combines PCL's **self-supervised learning** and NW Head's **interpretability**.



Experimental Setup

Datasets

UK BioBank (**UKBB**) ➡️ Large, **unlabelled** dataset

- 3D brain MRI images
- # samples = **39,541**
- Collected from predominantly healthy individuals for tracking health outcomes

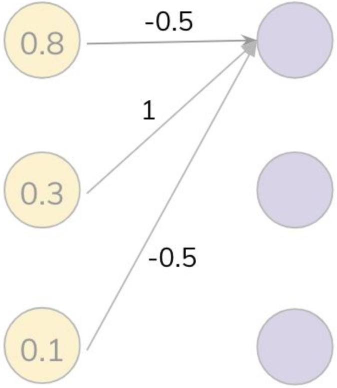
Alzheimer's Disease Neuroimaging Initiative (**ADNI**) ➡️ Small, **labelled** dataset

- 3D brain MRI images
- Labels: **AD** (Alzheimer's Disease), **MCI** (Mildly Cognitively Impaired), **CN** (Cognitively Normal)
- # samples = **1,245** → 256 AD, 610 MCI, 379 CN
- Collected for understanding the development of AD

Evaluation Strategy

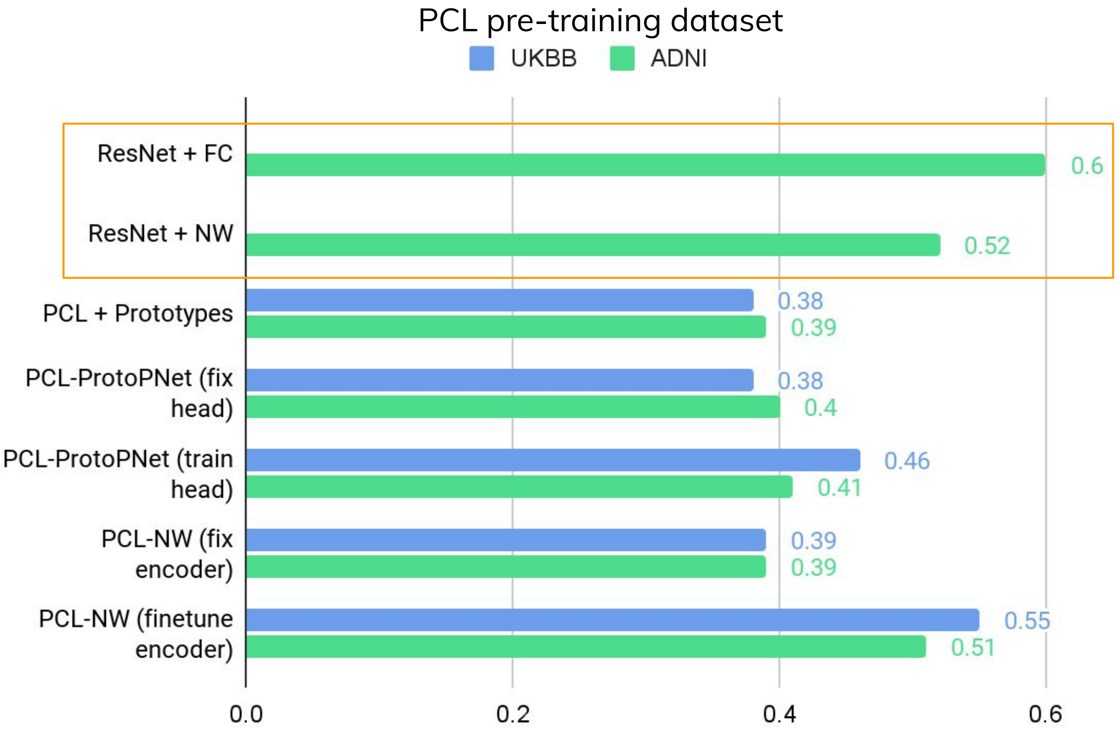
We compare the **balanced accuracy** of:

1. Baselines → ResNet + FC, ResNet + NW
2. PCL + Prototypes → Train encoder on UKBB vs. ADNI using PCL and project their prototypes
    - Predicted label = The label of the most similar prototype with query
3. PCL-ProtoPNet → Use PCL-trained encoder on UKBB vs. ADNI
    - Fix head
    - Train head
4. PCL-NW → Use PCL-trained encoder on UKBB vs. ADNI
    - Fix encoder
    - Finetune encoder



Results

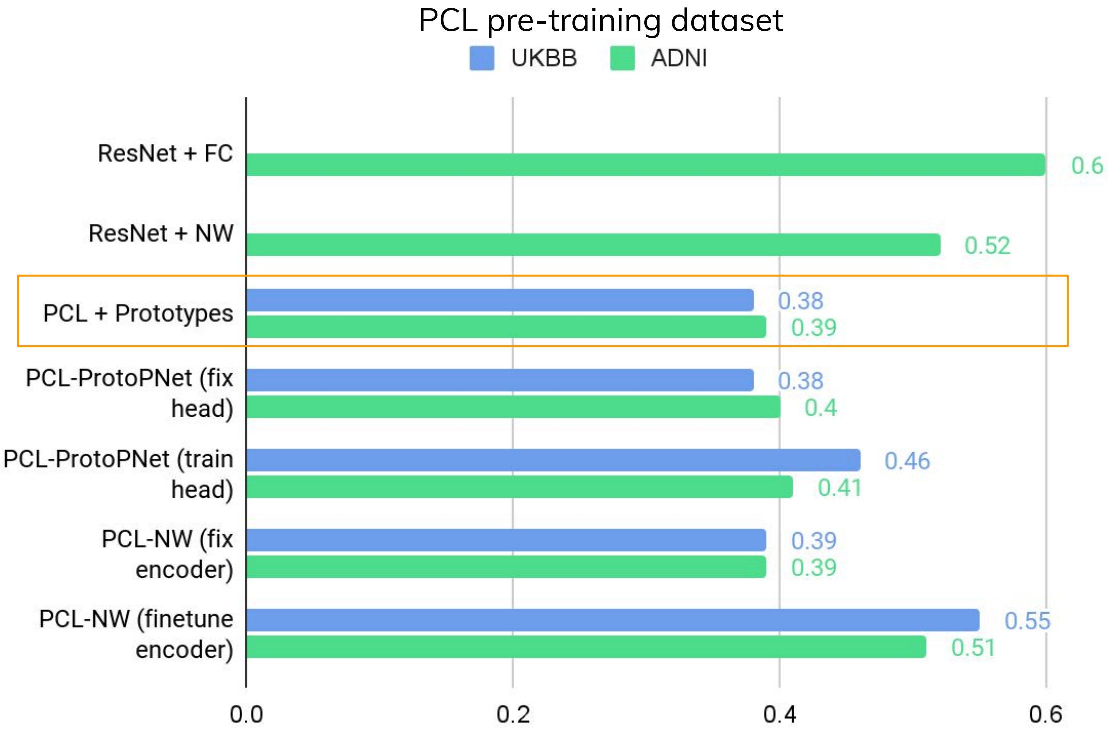
bAcc averaged over 5 folds of the ADNI test set



Baselines:

The **interpretability** provided by the NW head comes at the cost of **performance.**

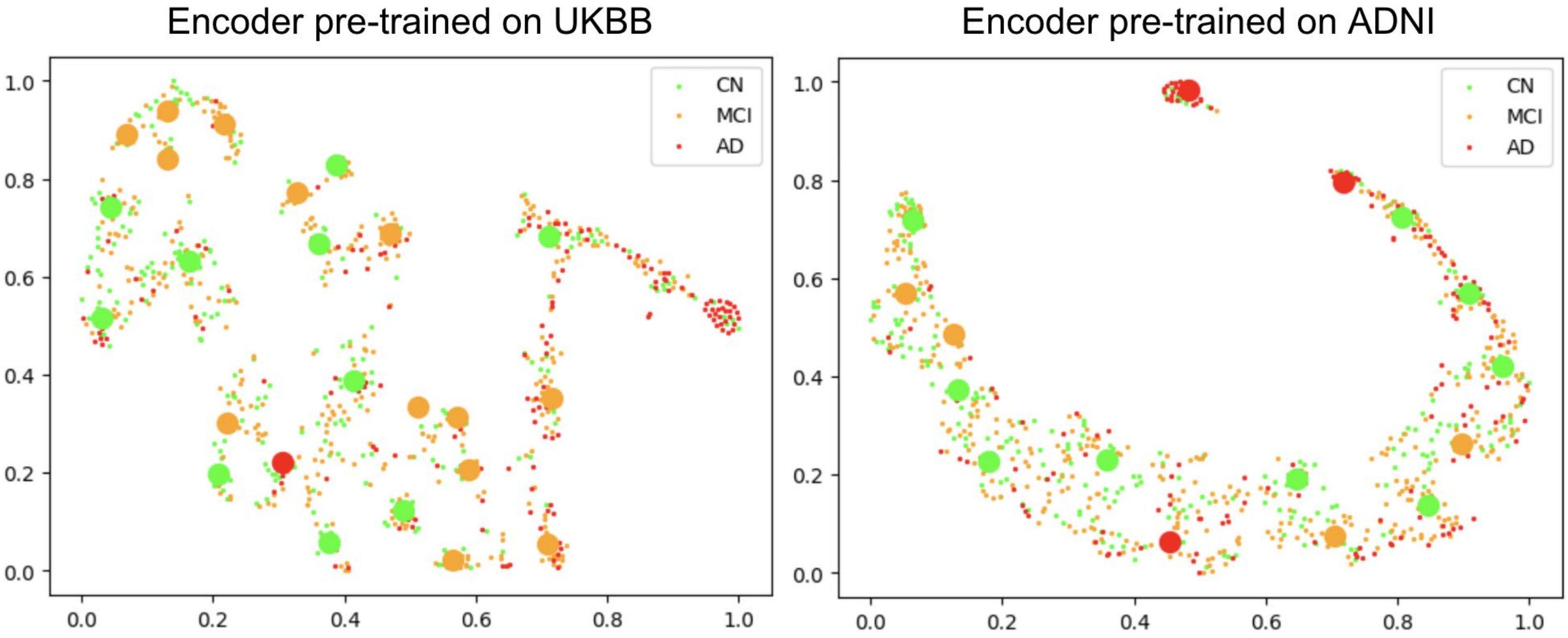
bAcc averaged over 5 folds of the ADNI test set



- Model pre-trained on **UKBB < ADNI**
- Worse compared to the baselines
- The prototype closest to the query doesn't represent the class the query belongs to

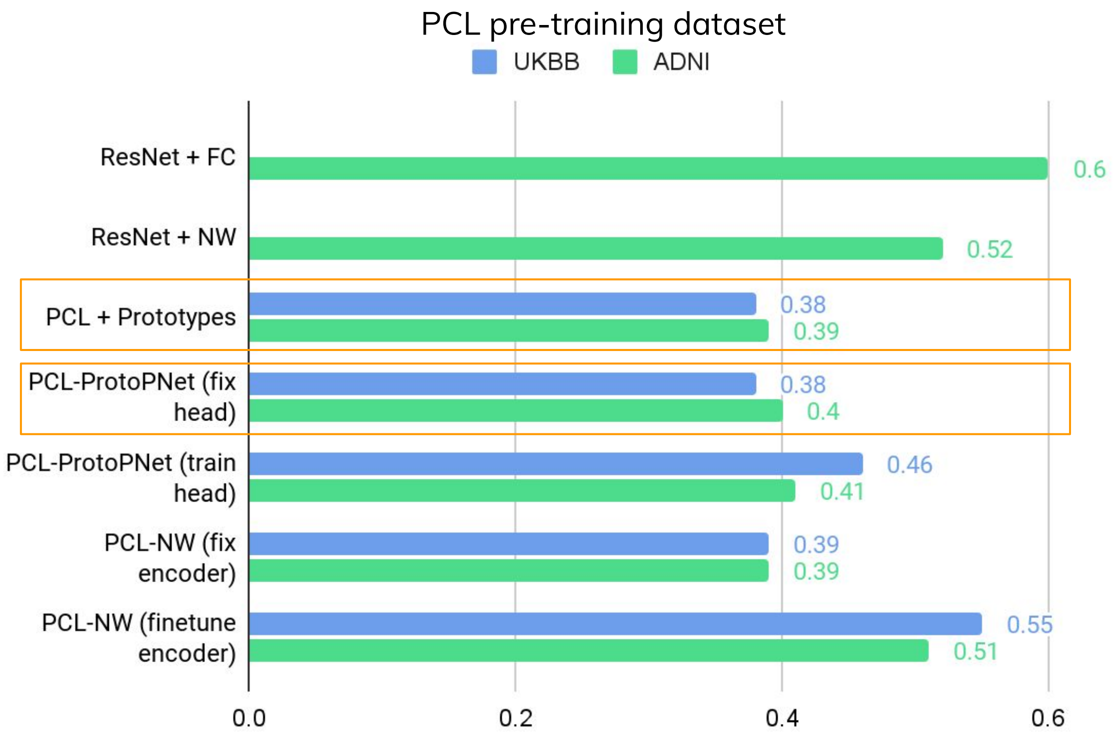
t-SNE plots of ADNI features by *PCL + Prototypes*

Bigger dots = Prototypes



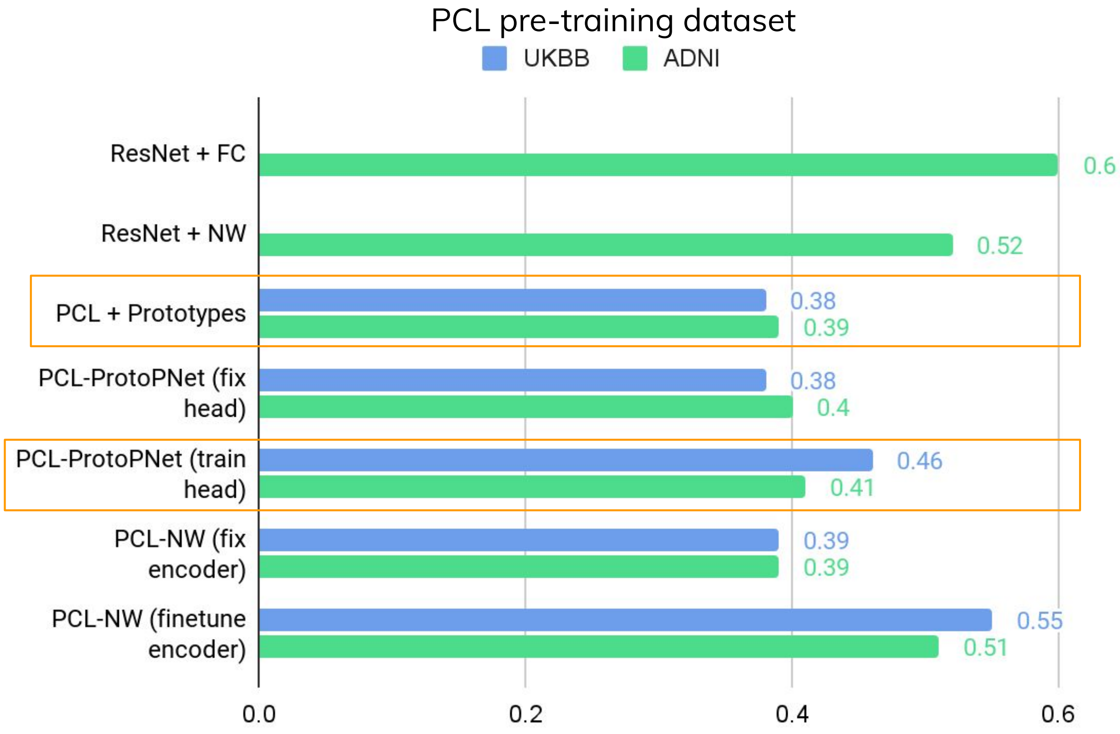
PCL failed to learn disease-specific features from UKBB / ADNI.

bAcc averaged over 5 folds of the ADNI test set



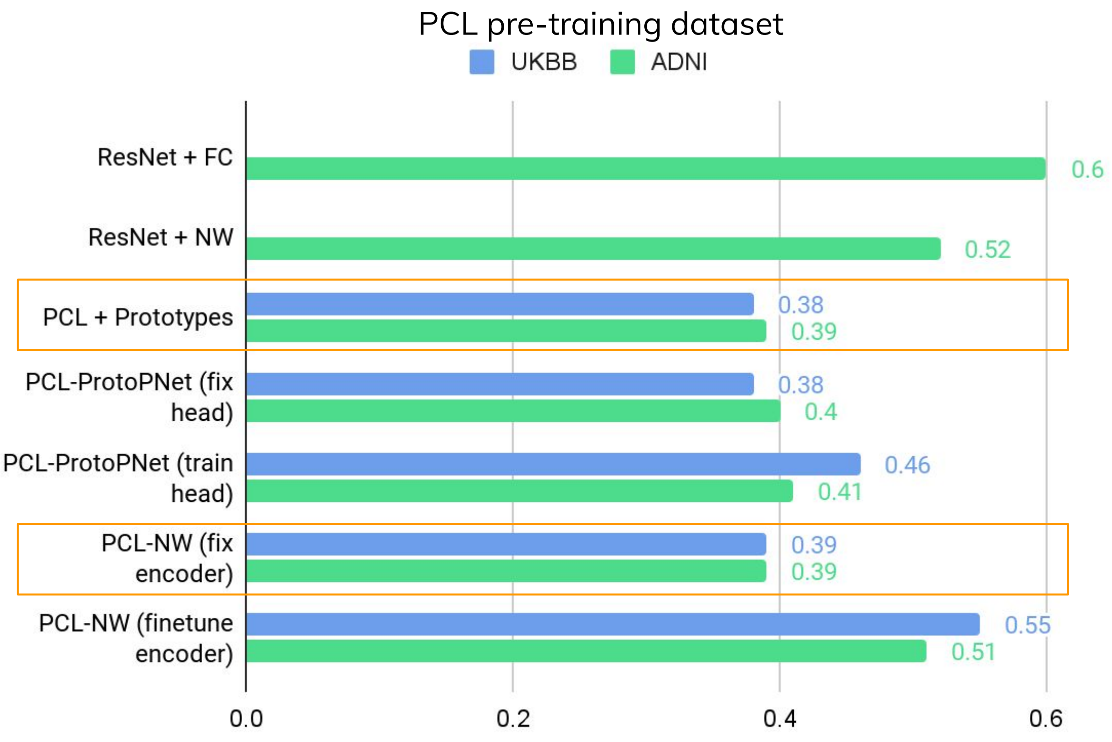
- Adding a fixed FC head does **not** significantly affect performance.
- Even though similarity scores to other prototypes are also weighed by the head, the **prototype closest to the query** has the **most influence** on the prediction.

bAcc averaged over 5 folds of the ADNI test set



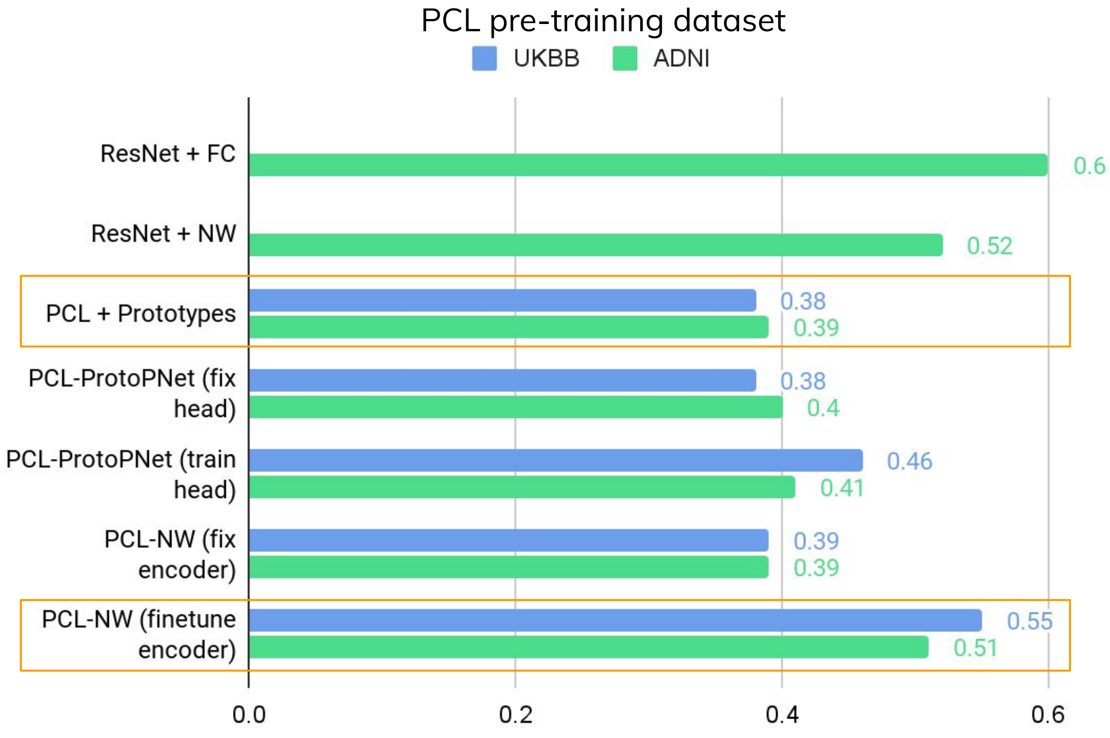
- **UKBB** encoder → Performance **increase**

bAcc averaged over 5 folds of the ADNI test set



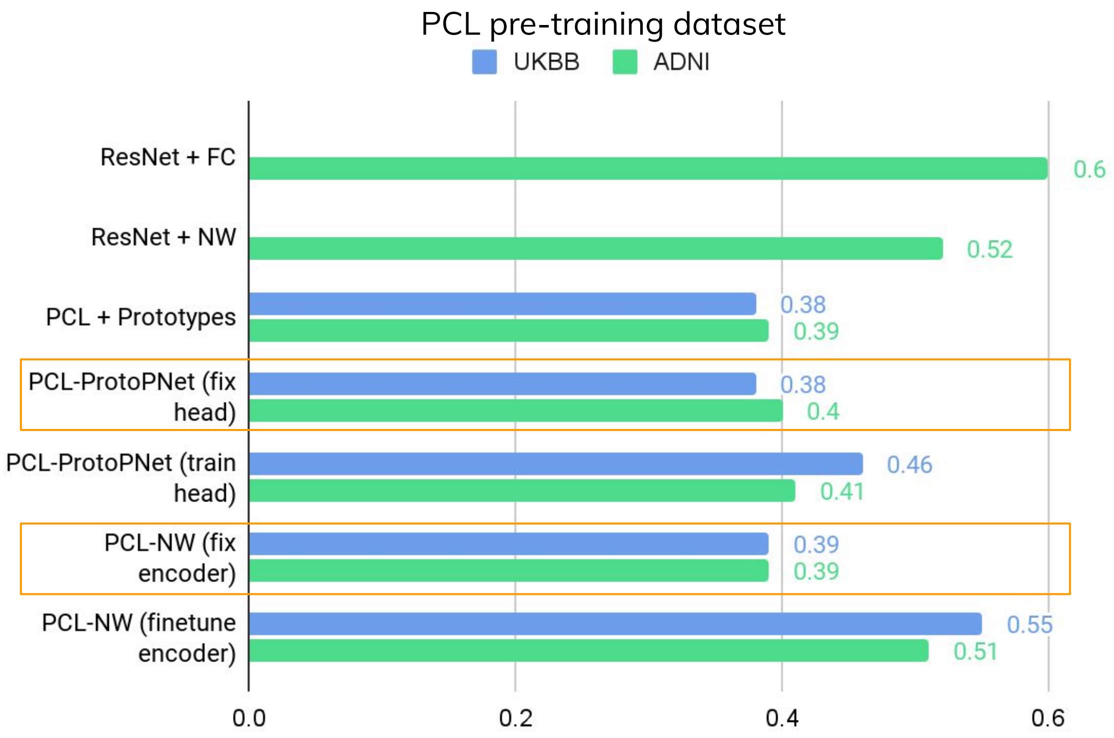
- **Not much difference** in performance.

bAcc averaged over 5 folds of the ADNI test set



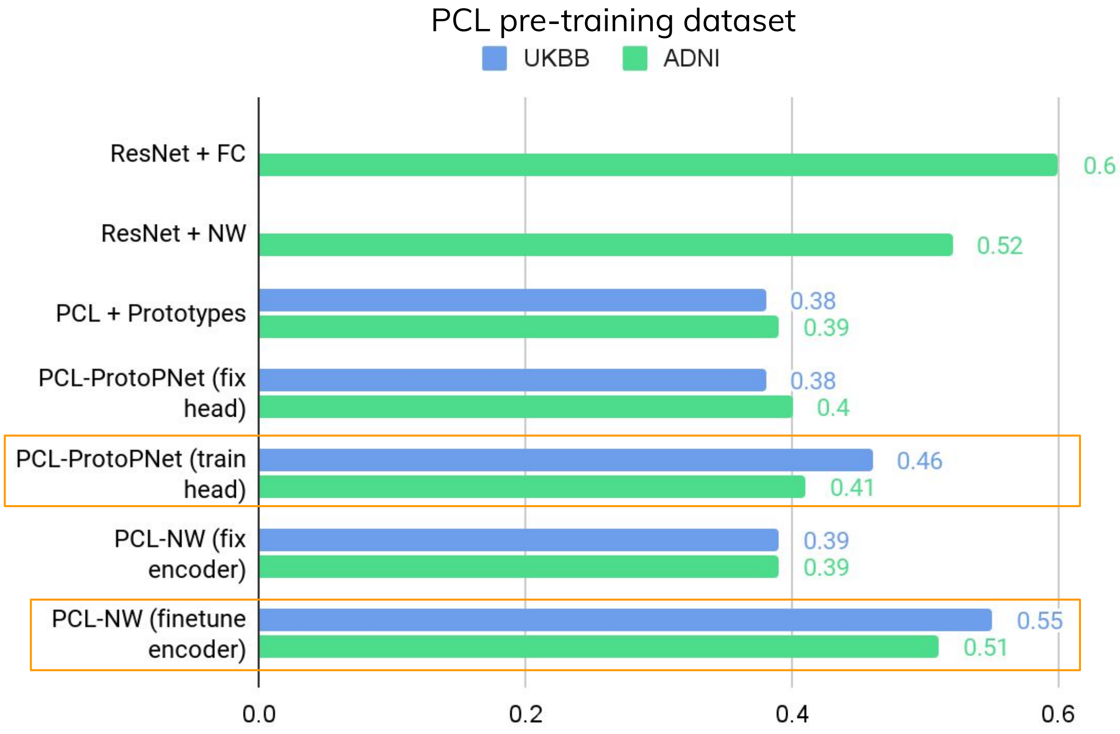
- Finetuning the encoder after PCL **significantly improves** performance, especially for the UKBB encoder.

bAcc averaged over 5 folds of the ADNI test set



- No further training after PCL
- Differ only in their **heads** and the usage of **prototypes**
- **Not much difference** in performance

bAcc averaged over 5 folds of the ADNI test set



- Further training after PCL
- **Finetuning the encoder > Training the head**
- The head's ability to improve depends on the features produced by the encoder.
- **UKBB > ADNI**

💡 Conclusion

Conclusion

- **Finetuning the encoder** after PCL and applying the **NW head** delivers the **best performance** among all our proposed methods.
- In cases where **further training** is done after PCL, pre-training on **UKBB > ADNI.**
- **Not much improvement** from the baseline ***ResNet + NW*** → Initializing the encoder with PCL weights might not help a lot
- Since both methods provide interpretability: **Directly training** *ResNet + NW* on the ADNI dataset **> PCL pre-training** the encoder first

Future Directions

- **y-aware PCL** → Guide PCL pre-training using **metadata** related to the 3D brain MRI images
- For AD classification, **age** can be useful → AD is correlated with older ages
- A combination of metadata can also be useful.