

# AIL 862

## Lecture 3

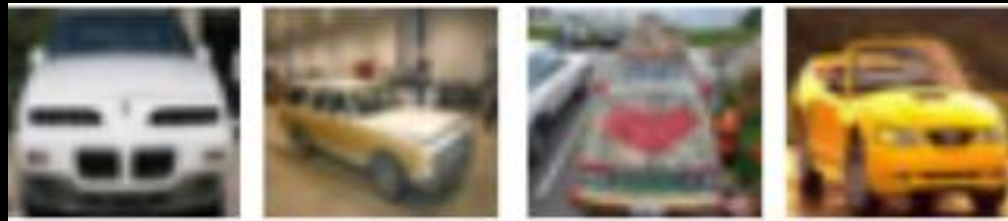
# Usual Recipe

- Understand your task: domain, classes etc.
- Collect a lot of images accordingly.
- (Somehow) annotate those images.
- Use images and the labels (annotations) to train a (CNN) model.
- Deploy: use trained model from now on the test data.

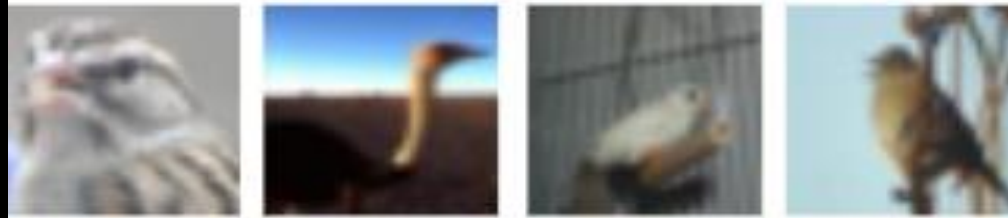
Does Recipe Always Work?

# Label

Automobile



Bird



Examples from CIFAR-10 dataset, source:  
<https://www.cs.toronto.edu/~kriz/cifar.html>

# Label



Building, also road,  
cars

Example from UC  
Merced

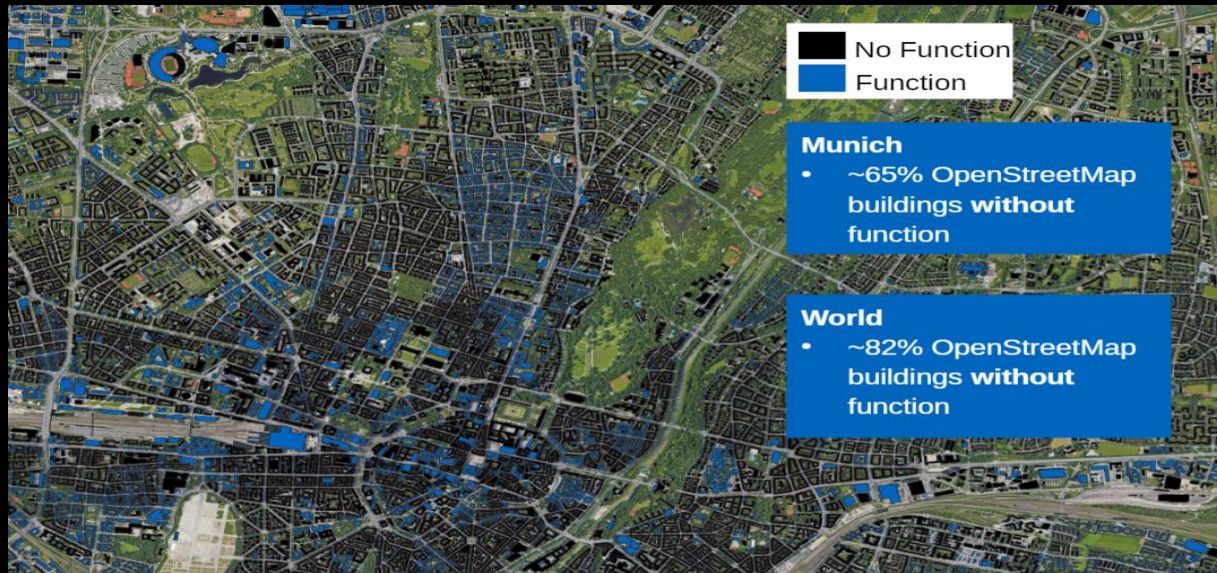
# Annotation: difficulty and ambiguity



Domain knowledge  
is needed

# Weak Supervision

- Example of OSM-based building annotation



# Weak Supervision

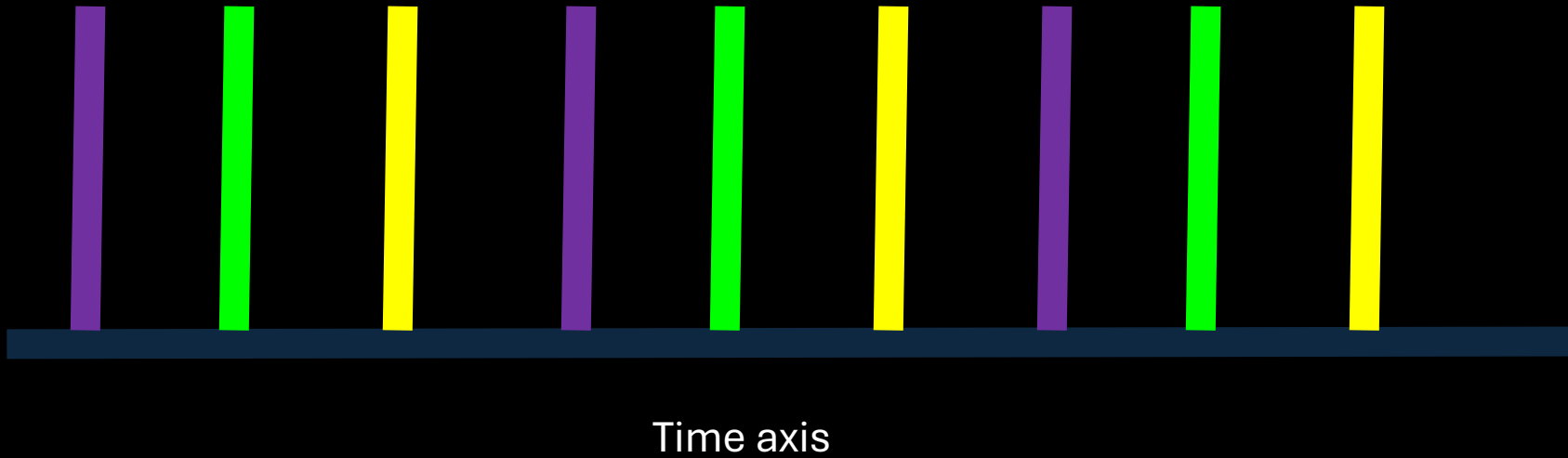
- Often misaligned
- In case of the OSM example, georeferencing issue



# Multi-source/sensor/modality data

- Challenge or complementary information?

# Improving temporal resolution using multi-sensor information



# Multi-Sensor EO

- Differences in spatial and temporal resolution.
- Differences in spectral characteristics.

# Multi-modality

- Building type classification: images + tweets.
- Geographic localization of non-EO data is challenging.
- Can they be processed using similar architectures?

# Multi-sensor Medical Image Analysis

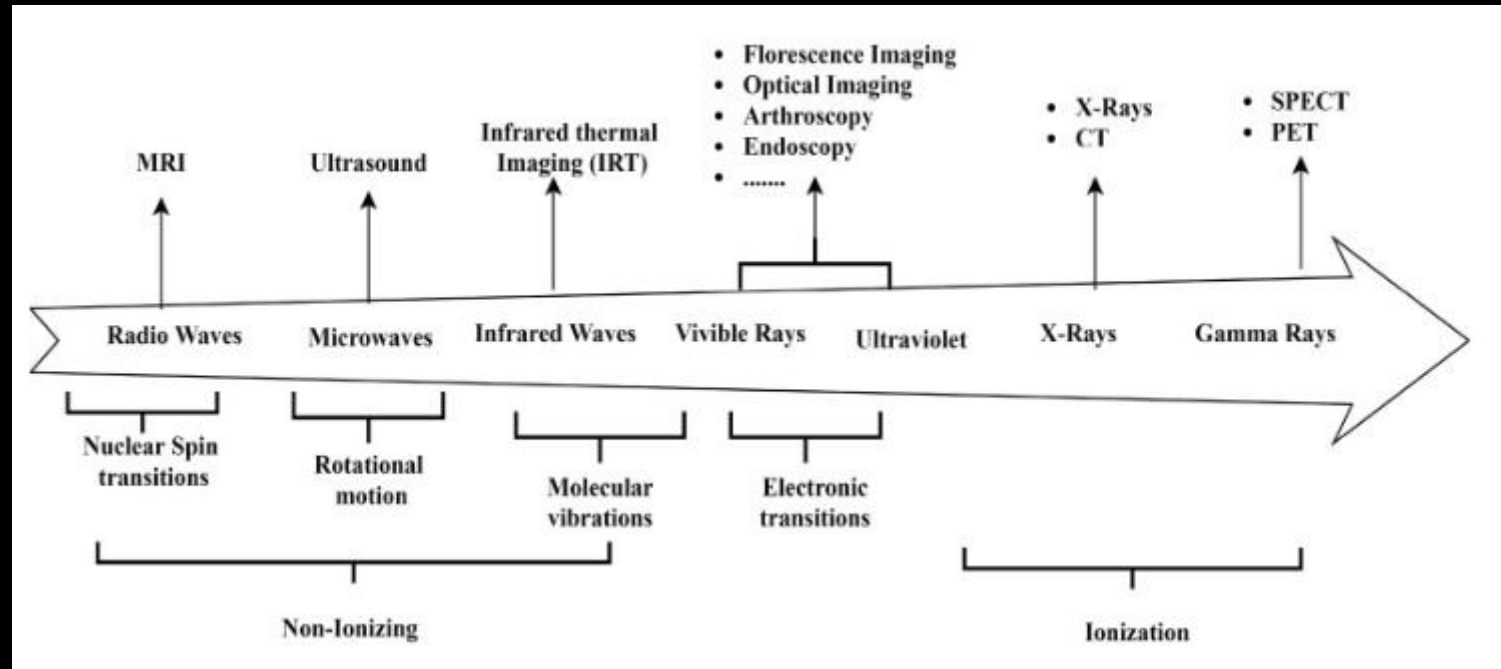


Figure from A review on multimodal medical image fusion: Compendious analysis of medical modalities, multimodal databases, fusion techniques and quality metrics

# Multi-modality in medical image processing

- Images + medical text data + maybe patient history (text or tabular format)

# Domain Differences

## DomainNet dataset

**Clipart:** collection of clipart images

**Real:** photos and real world images

**Sketch:** sketches of specific objects

**Infograph:** infographic images with specific object

**Painting** artistic depictions of objects in the form of paintings

**Quickdraw:** drawings of the worldwide players of game “Quick Draw”

# Domain Differences

Concept	In-domain	Target domain
<b>Sensor</b>	Sentinel-2	Worldview-2, Sentinel-1, ...
<b>Season</b>	Summer	Winter, Autumn, Fall
<b>Geography</b>	Munich	HongKong, New York, ...



# Domain Differences

Think of autonomous driving

**Sunny**

**Rainy**

**Foggy**

# OOD

Concept	In-domain	Target domain
<b>Sensor</b>	Sentinel-2	Worldview-2, Sentinel-1, ...
<b>Season</b>	Summer	Winter, Autumn, Fall
<b>Geography</b>	Munich	HongKong, New York, ...

# OOD

Two options

- Identify and leave out OOD domains
- Generalize to OOD domains?

# Big Noisy Data

- Visual data continuously collected
- Generally challenging to annotate
- No straightforward labeling mechanism (however, many implicit mechanisms available)

# Supervised Learning: 4 components

- Training data
- Learner
- Learning algorithm
- Performance

# Supervised Learning: 4 components

- Training data : input features, target/feedback
- Learner : parameters  $\theta$
- Learning algorithm : changes the parameters and improves performance
- Performance : cost function
- Predict - Score - Learn - ...

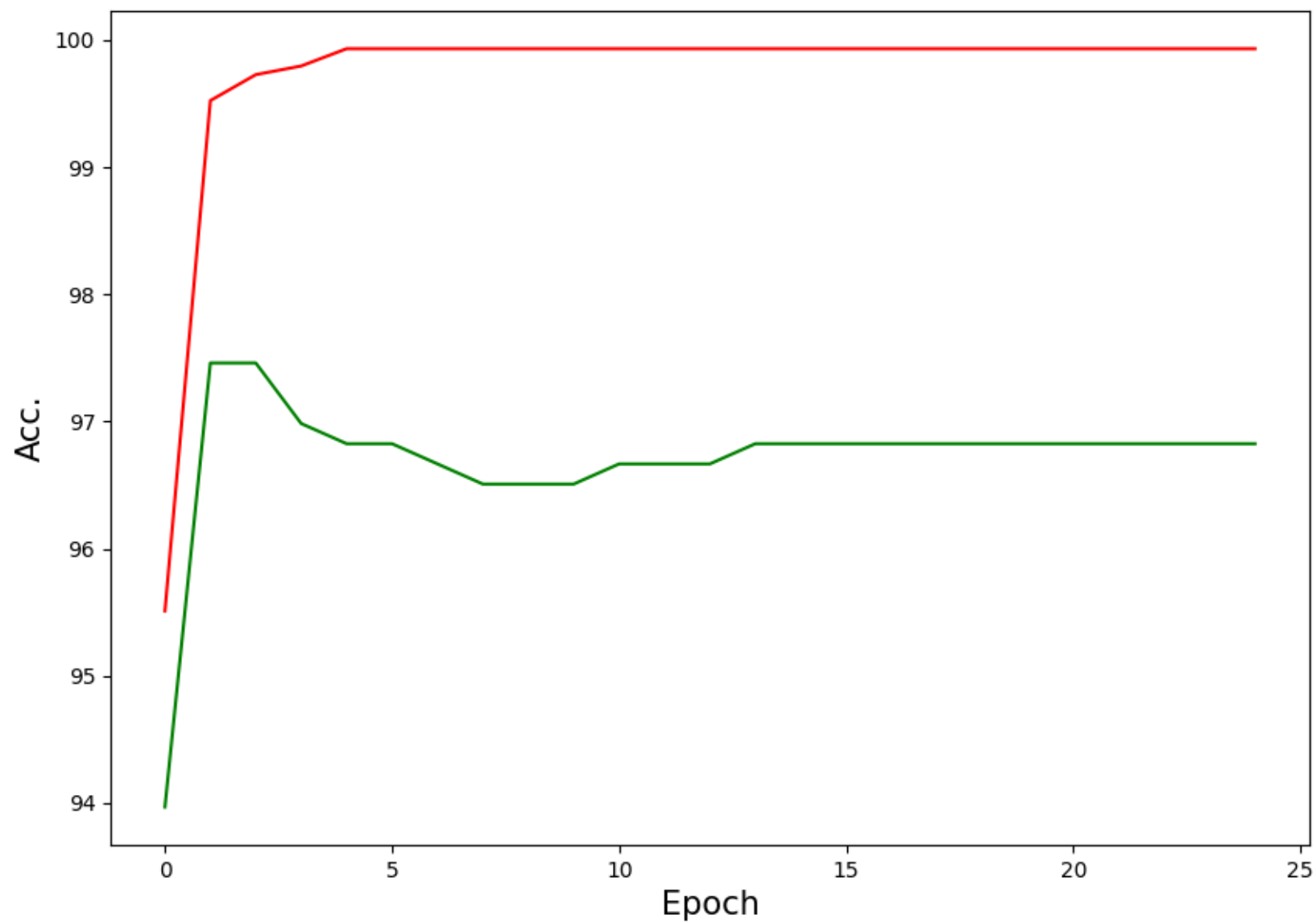
# Deep Learning

- Analogy to our multi-year education

# Supervised Learning: various issues

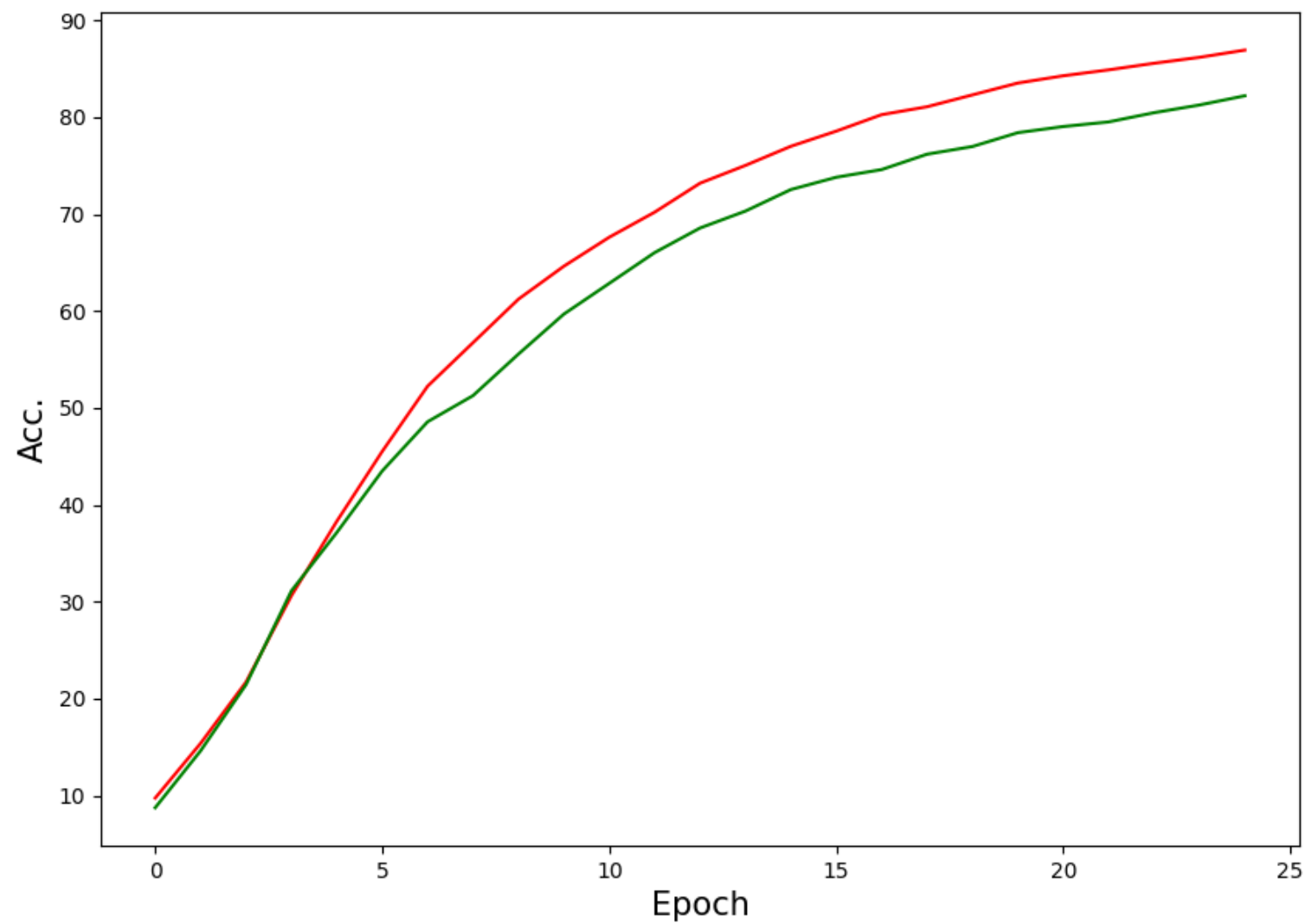
- Dependence on training data
- Overfitting and complexity (complex models overfit)
- Choice of (hyper)parameters





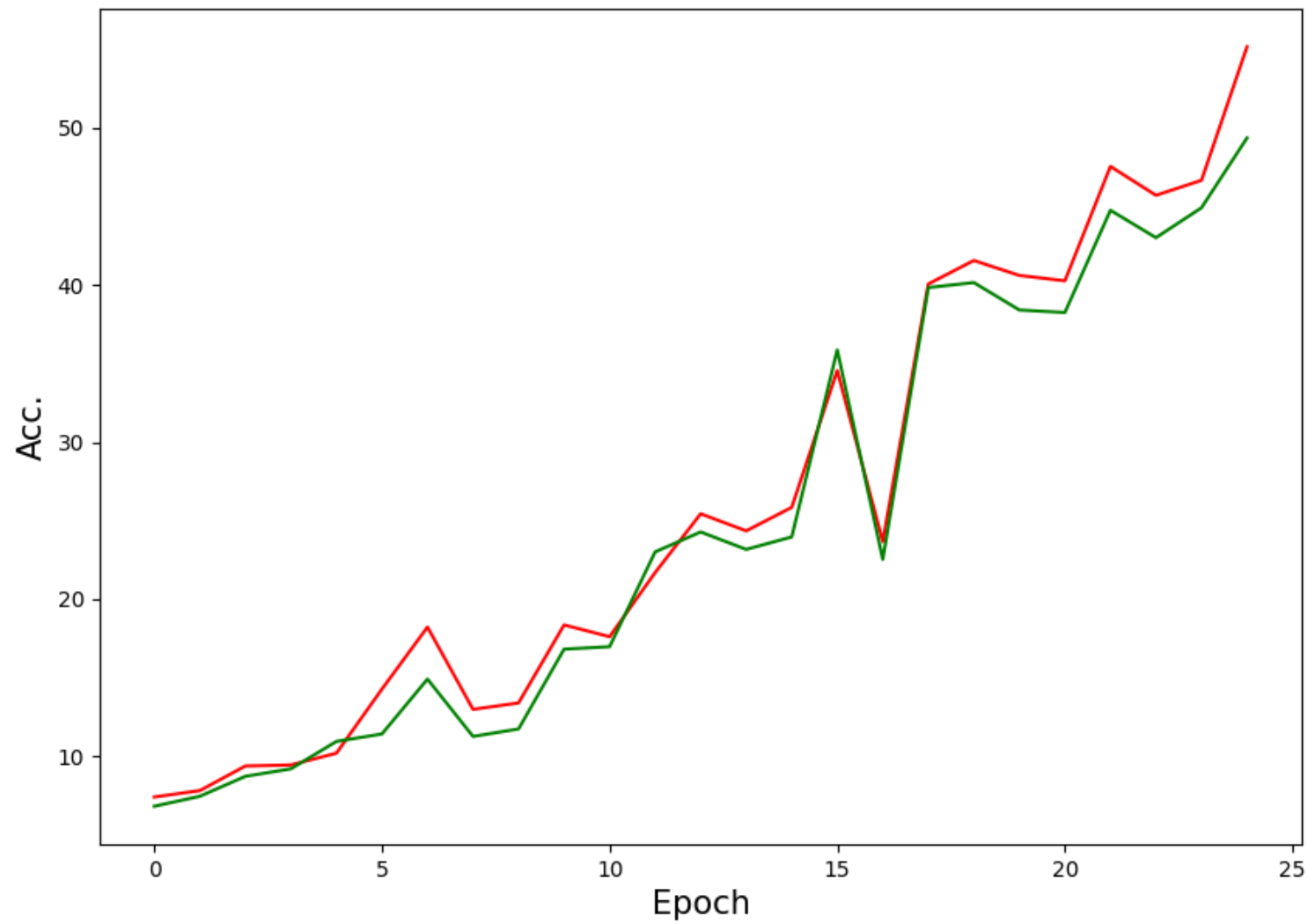
Starting from  
pretrained model

LR - good



Starting from  
pretrained model

LR - comparatively low



Starting from  
pretrained model

LR – comparatively  
high

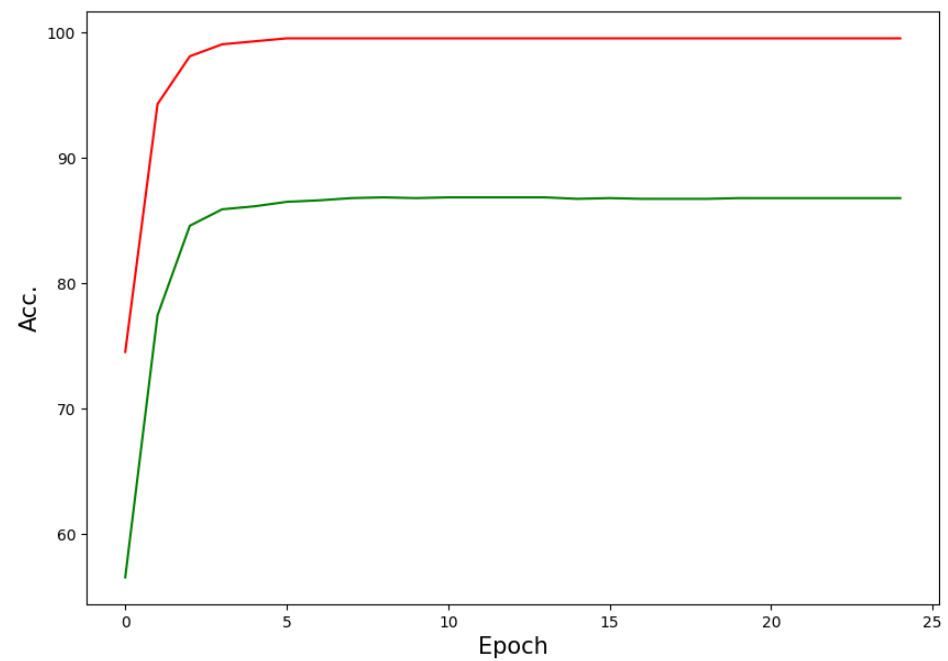
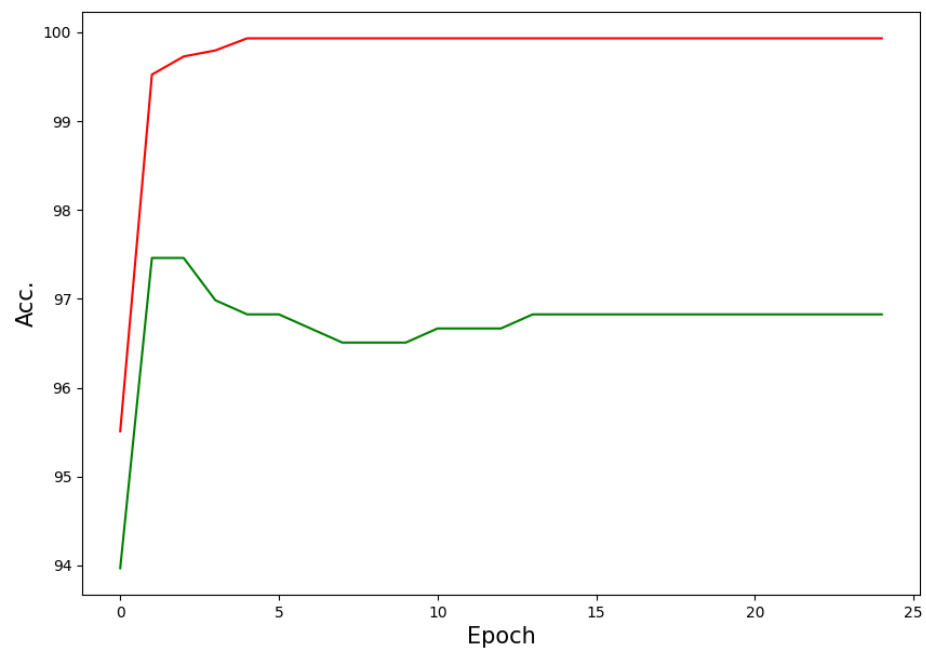
# Data Split

- Training
- Validation
- Test

# Supervised Learning

Dataset	Size
UC Merced	2100
AID	10000
Optimal-31	1860
RSSCN7	2800
Cifar-10	60000
Cifar-100	60000

# Training Data Amount



# Using Models Trained in Supervised Fashion For Some Other Task

- *Just use as feature extractor*

Just use the features extracted from particular layer(s) without any tuning

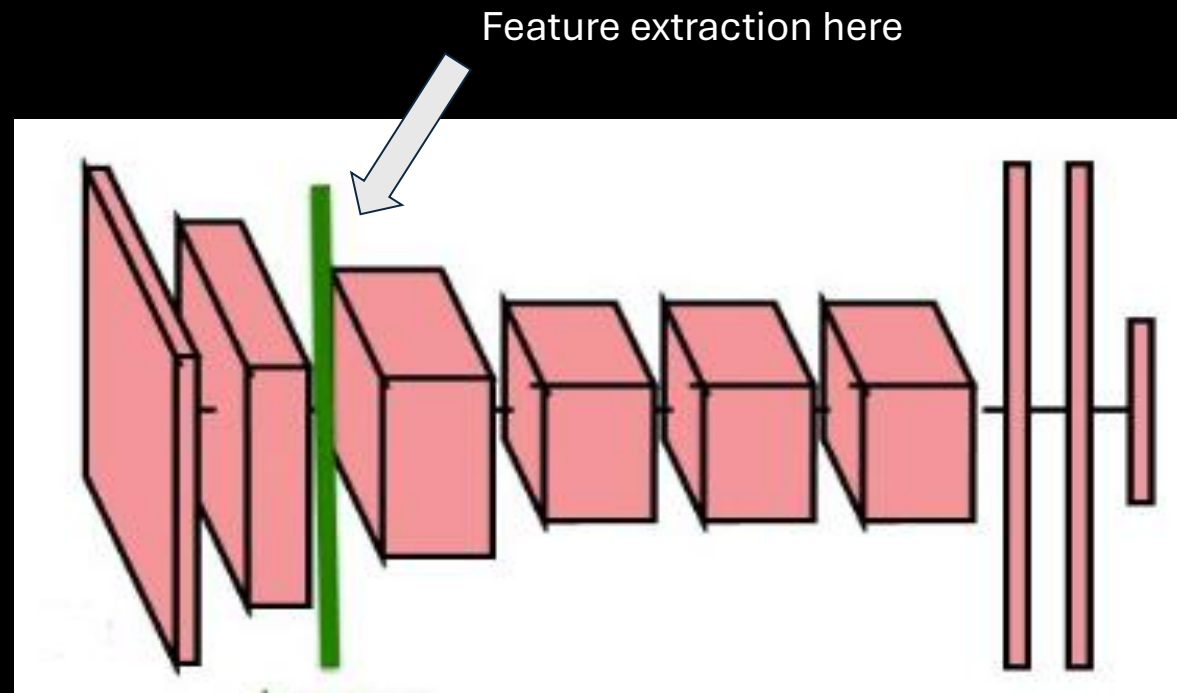
- *Fine tuning on target data*

Further train particular layers of the network

- *Unsupervised domain adaptation*

Adapt the network with unlabeled target data

# Just as Feature Extractor





# Some Aspects

- Which layer?
- Do we have a validation dataset?
- Hypercolumn?
- Any unsupervised way to rank features from different layers?

# Using Models Trained in Supervised Fashion For Some Other Task

- *Just use as feature extractor*

Just use the features extracted from particular layer(s) without any tuning

- *Fine tuning on target data*

Further train particular layers of the network

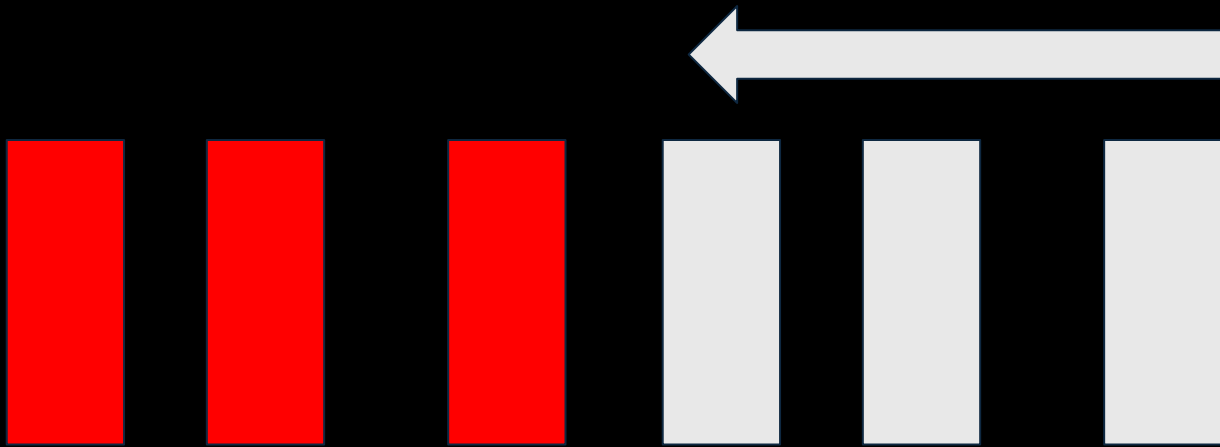
- *Unsupervised domain adaptation*

Adapt the network with unlabeled target data

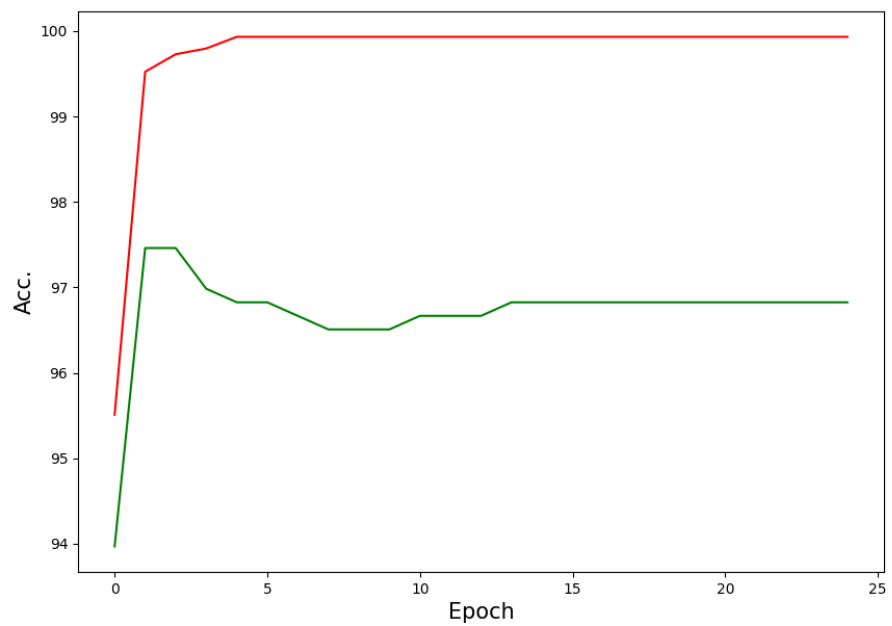
# Fine Tuning

Further train particular layers of the network

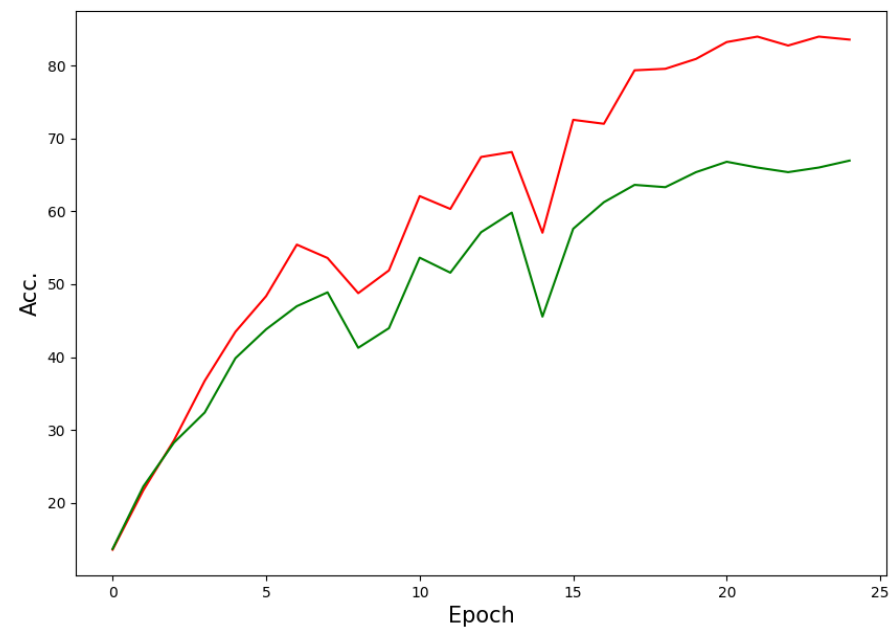
Red layers are frozen.



Train from ImageNet trained model



Train from scratch



# Fine tuning

- Pre-training + finetuning does not always work.
- Generally: initial layers are somewhat domain specific.