DS 4400

Machine Learning and Data Mining I
Spring 2021

Alina Oprea
Associate Professor
Khoury College of Computer Science
Northeastern University

April 1 2021

Announcements

- Final exam will start at 11:45am on Tuesday, April 6
  - It will be up for 6 hours
  - You can pick up a time frame of 2 hours
- Ethics of AI: Thu, April 8, by Kevin Mills
  - Over Zoom
  - Please fill in survey before class

- Tuesday, April 20
- Thu, Apr. 22.

PROJECT PRESENTATION

REPORT >1 week.

Outline

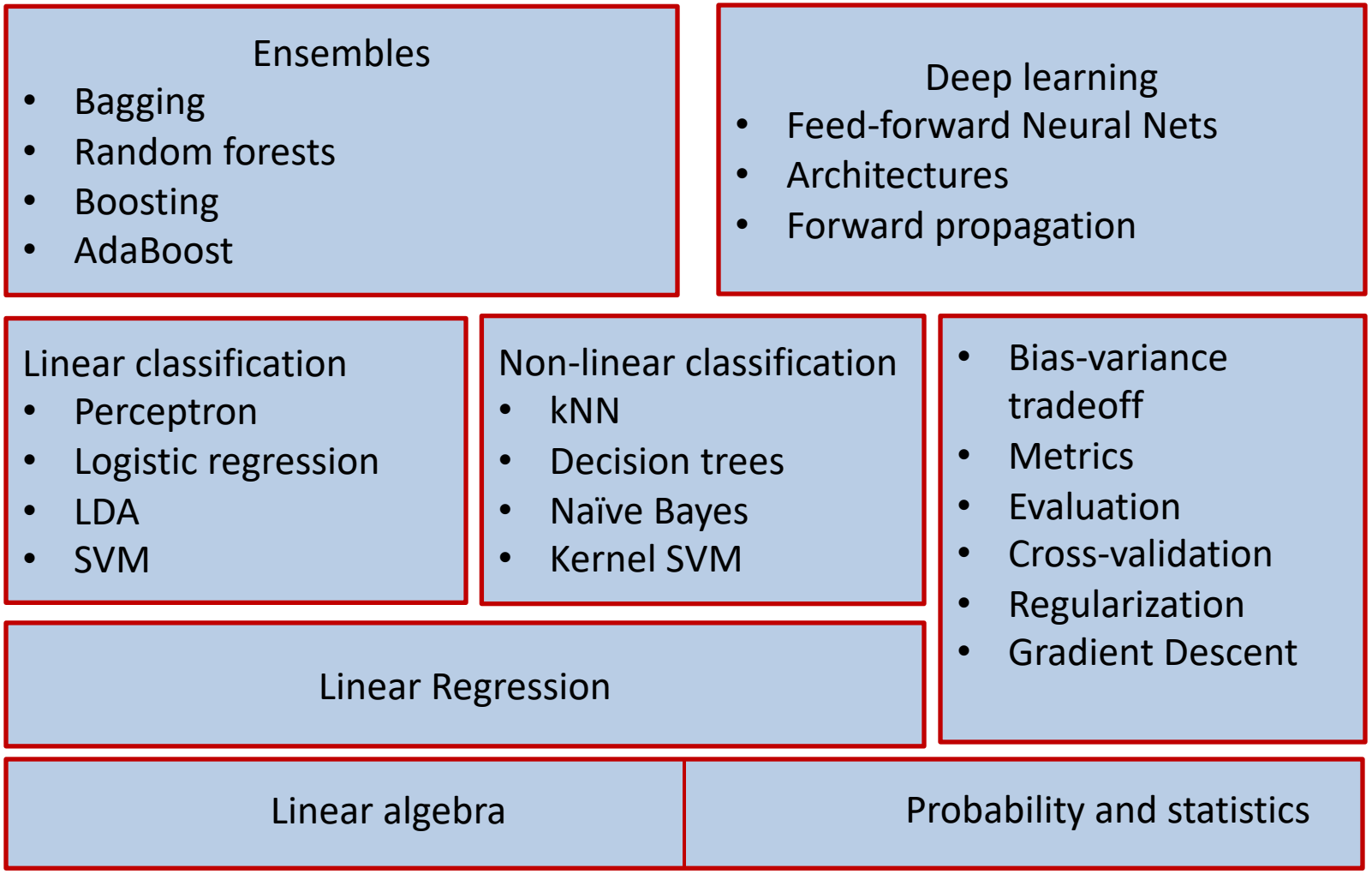
- Final exam review
- Transfer learning
  - Using pre-trained models for new tasks
- Training Neural Networks
  - Backpropagation
  - Parameter Initialization
  - Stochastic Gradient Descent

Exam Review

DS-4400 Course objectives

- Become familiar with machine learning tasks
  - Supervised learning vs unsupervised learning
  - Classification vs Regression
- Study most well-known algorithms and understand their details
  - Regression (linear regression)
  - Classification (Naïve Bayes, decision trees, ensembles, neural networks)
- Learn to apply ML algorithms to real datasets
  - Using existing packages in R and Python
- Learn about security challenges of ML
  - Introduction to adversarial ML

What we covered



Bias-Variance Tradeoff

- Why learning is hard
- What overfitting means
- How to avoid it

  REGULARIZATION (LASSO; RIDGE)

  CROSS-VALIDATION (HYPER-PARAM SELECTION)

- How different models improve generalization

  - DT (PRUNING, LIMIT DEPTH)
  - ENSEMBLES (BOOTSTRAP SAMPLES; VARY FEATURES)
  - NN: DROPOUT; WEIGHT DECAY (L2/ RIDGE)
  - NAIVE BAYES: LAPLACE SMOOTHING

ML Models

- Categorization
  - Is it a linear or non-linear?
  - Is it generative or discriminative?
  - Is it an ensemble?
- For each ML model
  - Understand how training is done
  - Take a small example and train a model
  - Once you have a model know how to evaluate a point and generate a prediction
    - Example: predict output by Naïve Bayes, decision tree, SVM, or neural network

Naïve Bayes Classifier

TRAIN

- For each class label k
  1. Estimate prior $\pi_k = P[Y = k]$ from the data
  2. For each value v of attribute X_j
     - Estimate $\mathrm{P}[X_j = v | Y = k]$

NB MODEL

TEST

- Classify a new point via:

$$h(\mathbf{x}) = \arg\max_{y_k} \log P(Y = k) + \sum_{i=1}^{d} \log P(X_j = x_j \mid Y = k)$$

π_k

$P(Y = h(X = x))$

ASSUMPTION: COND INDEP OF FEATURES ON CLASS LABELS

Learning Decision Trees

- Start from empty decision tree
- Split on **next best attribute (feature)**
  - Use, for example, information gain to select attribute:

$$\arg\max_i IG(X_i) = \arg\max_i H(Y) - H(Y \mid X_i)$$

- Recurse

Information Gain reduces uncertainty on Y

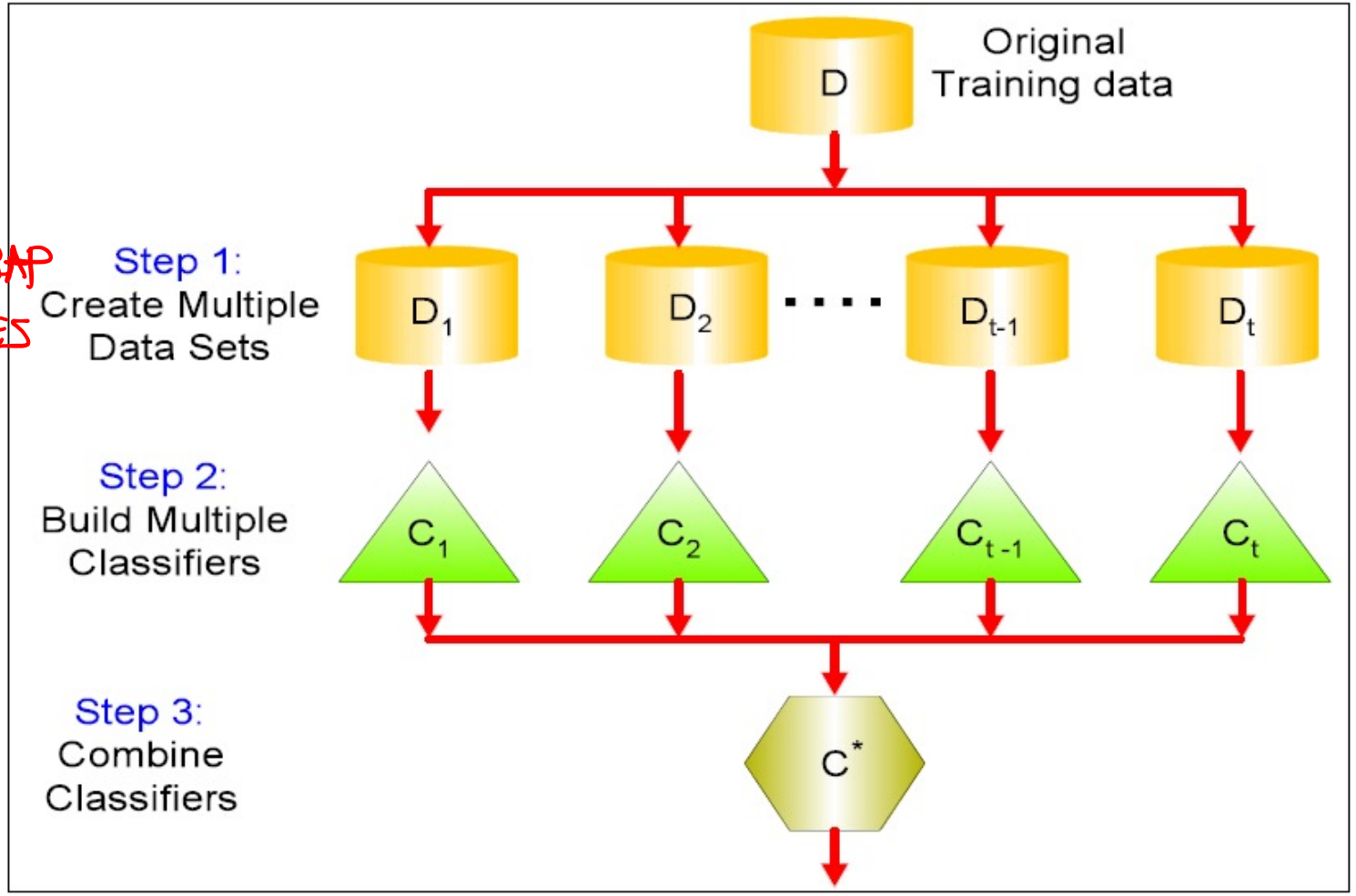
Can use Gini index

Bagging

RF .

PARALLEL

BOOTSTRAP SAMPLES



Majority Votes

Boosting

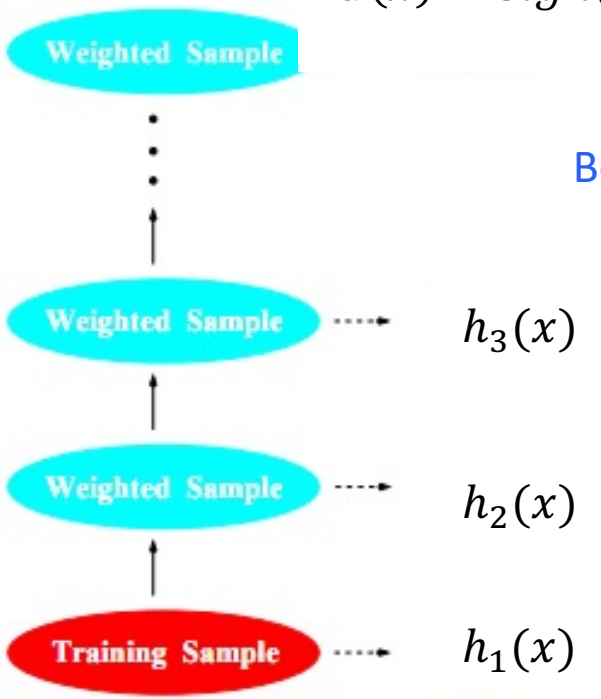
AdaBoost

- Both Bagging & Boosting reduce var.
- Boosting reduces bias

$$G(x) = sign\left(\sum_{i=1}^{T} \beta_t h_t(x)\right)$$

Better classifiers will get higher weights

- Mis-classified examples get higher weights
- Correct examples get lower weights

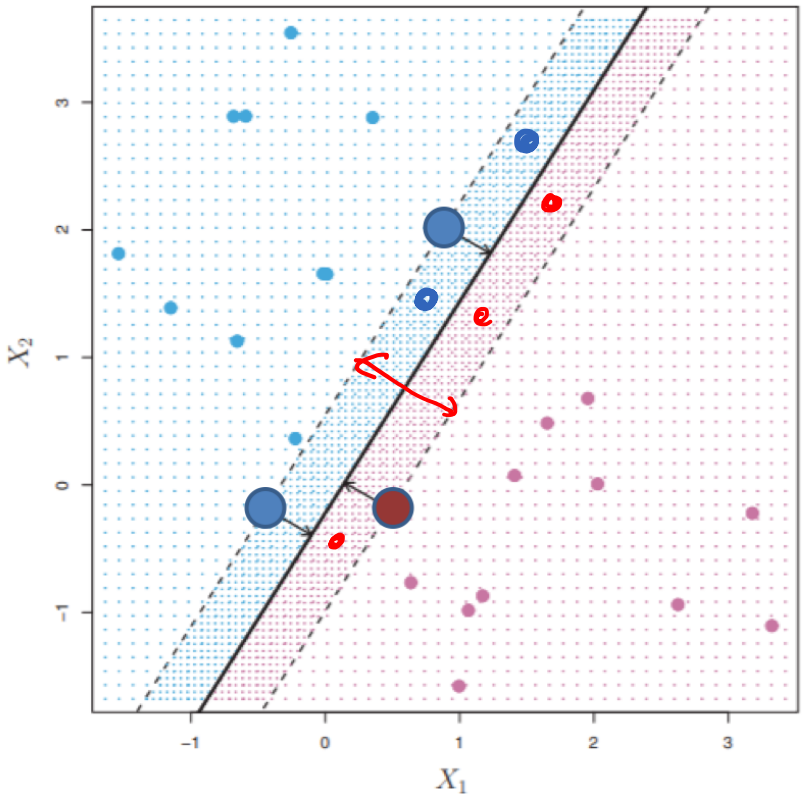


Uniform weights

FIGURE 10.1. *Schematic of AdaBoost. Classifiers are trained on weighted versions of the dataset, and then combined to produce a final prediction.*

SEQUENTIAL.

Support Vector Machines



- Support vectors = points “closest” to hyperplane
- Linear SVM: maximum margin
- Kernel SVM: radial basis, polynomial

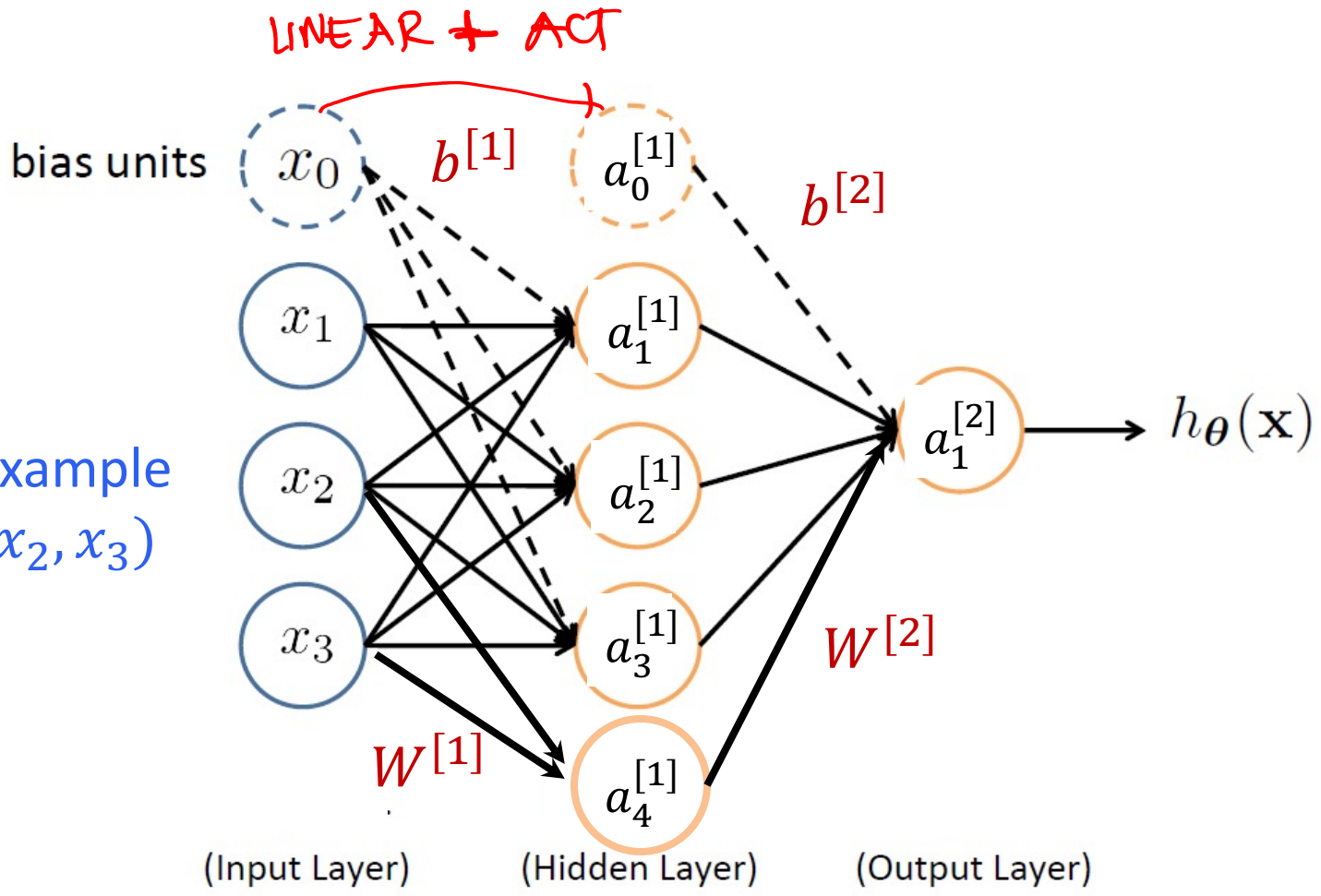
Online Perceptron

Let $\theta \leftarrow$[0,0,…,0]
Repeat:
    Receive training example (x_i, y_i)
    If $y_i \theta^T x_i \leq 0$ // prediction is incorrect
        $\theta \leftarrow \theta + y_i \, x_i$
Until stopping condition

Online learning – the learning mode where the model update is performed each time a single observation is received

Batch learning – the learning mode where the model update is performed after observing the entire training set

Feed-Forward Neural Network



Training example

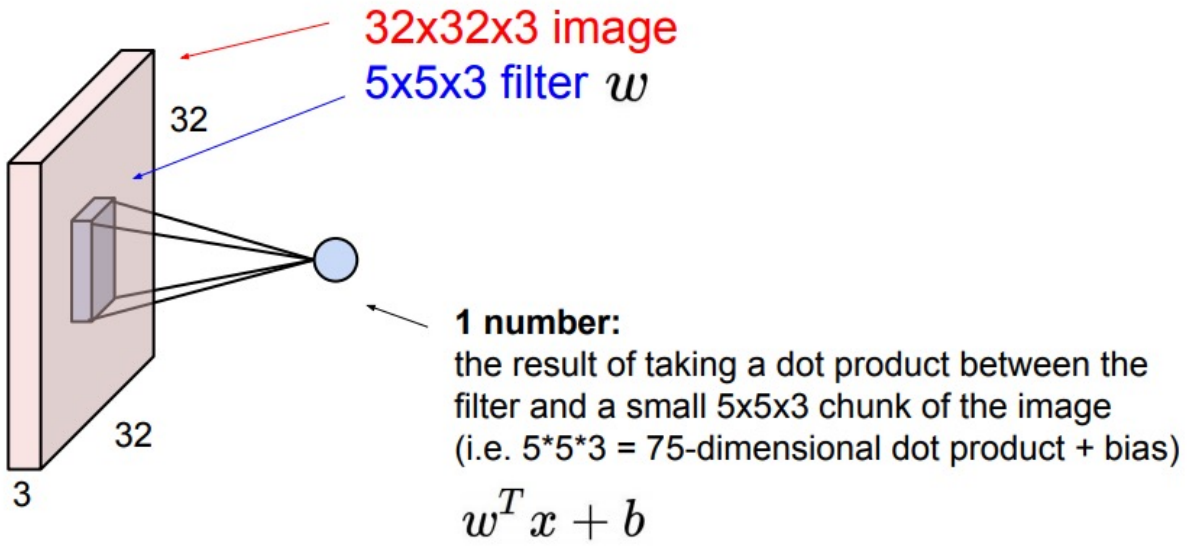
$x = (x_1, x_2, x_3)$

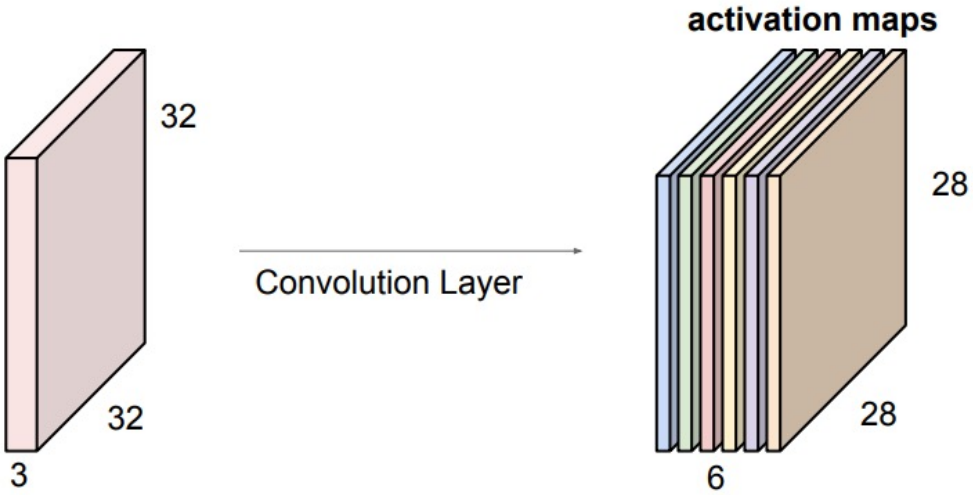
No cycles

4 12 1 4

$\theta = (b^{[1]}, W^{[1]}, b^{[2]}, W^{[2]})$

Convolution Layer





When to use each model

- Assumptions:
  - Naïve Bayes assumes conditional independence between features given class
- Linear models work well for linearly separable data
  - SVM results in linear model of max margin
  - Logistic regression estimates a probability
- Decision trees work well for categorical data
- Ensembles are powerful models
  - Need a lot of training data available
  - Reduce variance of single models

Comparing classifiers

H | M | L

Algorithm	Interpretable	Model size	Predictive accuracy	Training time	Testing time
Logistic regression	M	L	L	L	L
kNN	M	H	L	L	H
LDA	M	L	L	L	L
Decision trees	H	M	L	M	L
Ensembles	L	H	H	H	M-H
Naïve Bayes	M	M	L	M	L
Neural Networks	L	H	H	H	L
SVM	M	L	L	M	L

Type I: Conceptual

- Example 1: Describe difference between generative and discriminative models
- Example 2: Given some dataset with certain properties, what is the best model
- Example 3: Provide advantages and disadvantages, and compare the following:
  - Linear classifiers compared to non-linear ones
  - Naïve Bayes versus LDA
  - FFNNs versus CNNs
- Important: write short answers

Type II: Computational

- Example 1: Given a small dataset, train a particular ML model
  - E.g., decision tree, Naïve Bayes, etc.
  - Evaluate model on some small training and testing data
- Example 2: Given a particular model, describe the training process and count the number of parameters
- Example 3: Compute different metrics: true positives, false positives, precision, recall, accuracy, error

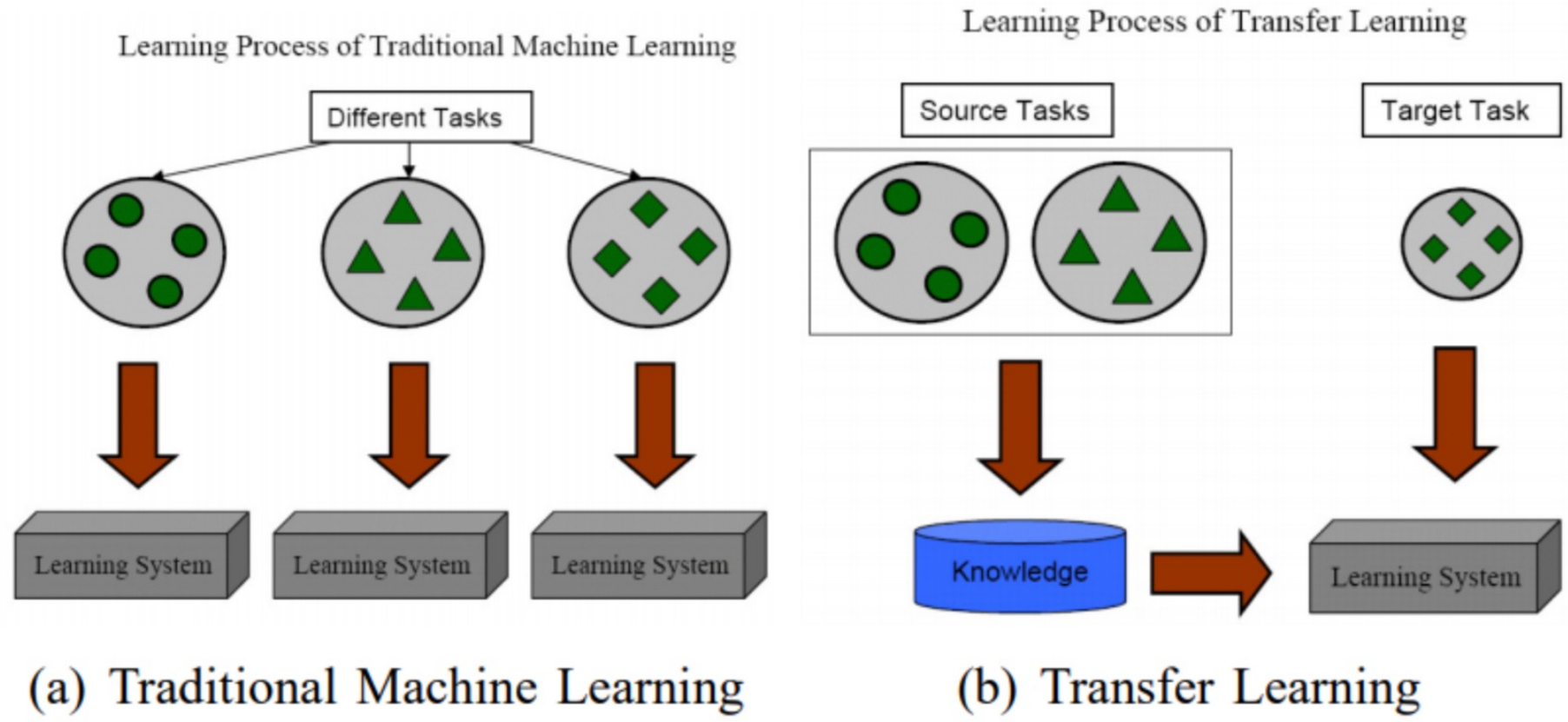
Type III: Constructive

- Example 1: Given a function, construct a FFNN to compute it
- Example 2: Construct a small NN architecture for a particular problem
- Example 3: Given dataset, which features to select for particular prediction problem

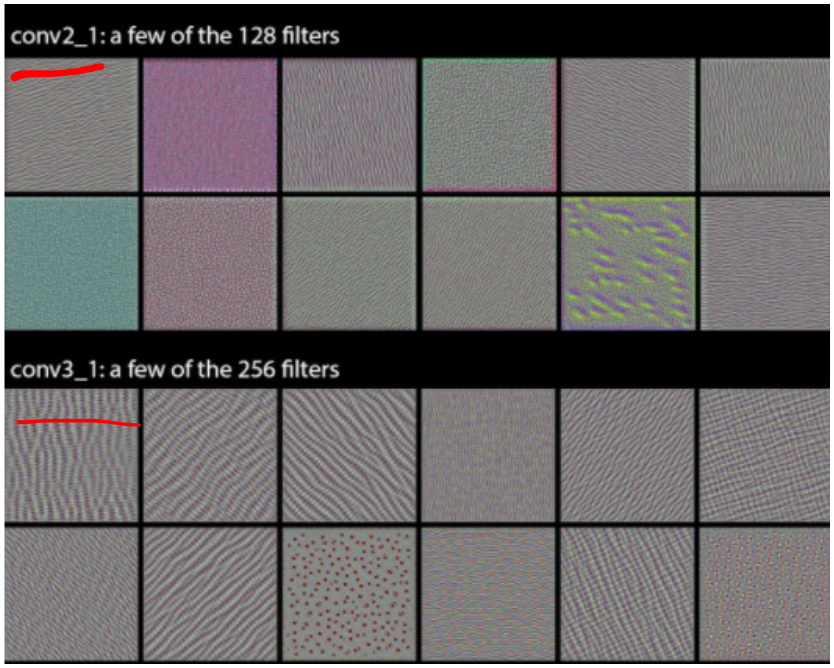
Transfer Learning

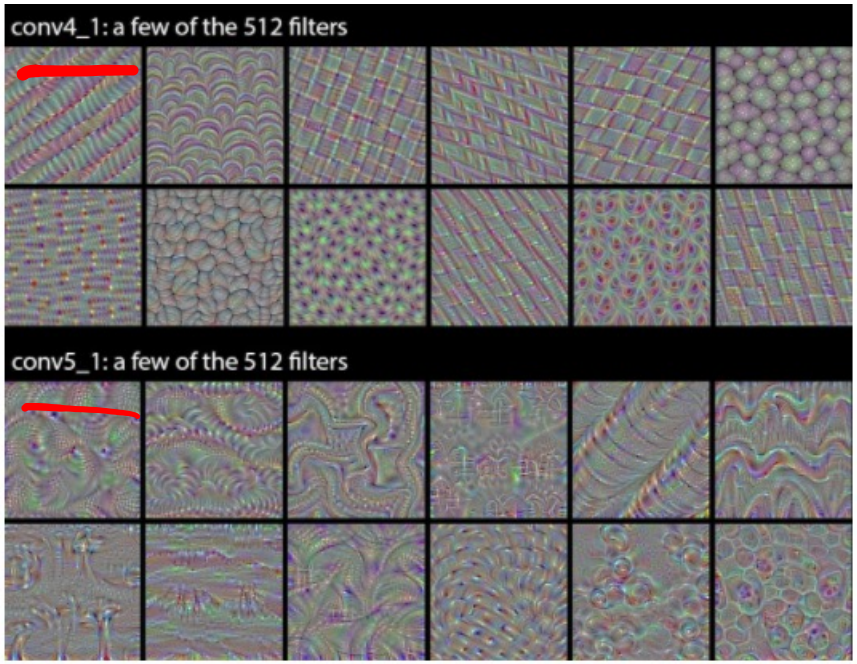
- Improvement of learning in a **new** task through the *transfer of knowledge* from a **related** task that has already been learned.
- Motivation: Reuse representations learned by expensive training procedures that cannot be easily replicated
  - Image classification on ImageNet is very expensive (VGG-16: 138 million, ResNet 50: 23 million parameters)
  - Generative language models very large (BERT: 110 million, GPT-2: 1.5 billion, GPT-3: 175 billion parameters)
- Two major strategies
  - Pretrained Neural Network as fixed feature extractor
  - Fine-tuning the Neural Network

Transfer Learning



Visualizing Filters in VGG 16

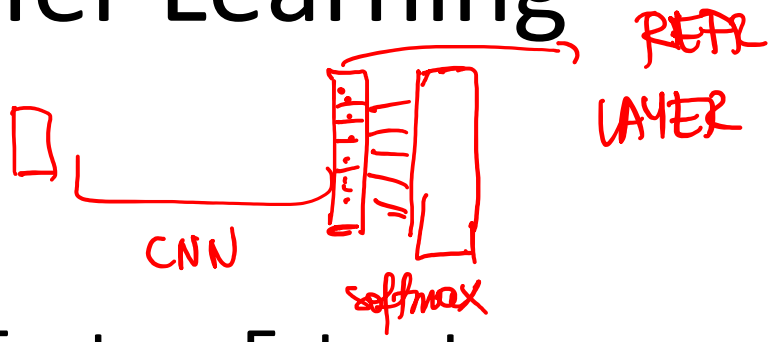




- First layers: general learners
  - Low level notion of edges

- Last layers: specific learners
  - High-level features: eyes, objects

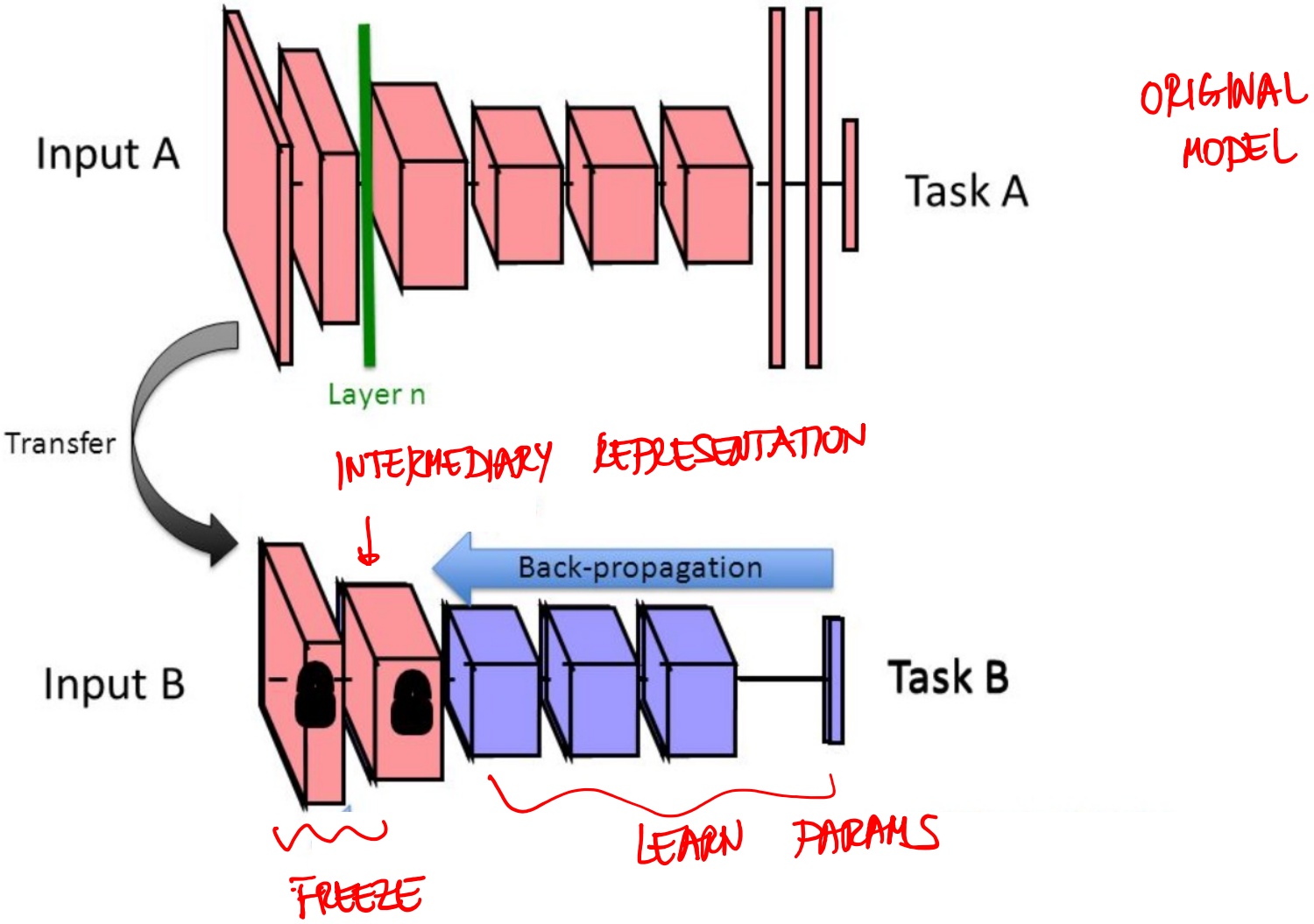
Methods for Transfer Learning



- Use a pre-trained model
  - https://modelzoo.co/

1. Use Convolutional Nets as Feature Extractor
   - Take a ConvNet pretrained on ImageNet
   - Remove the last fully-connected layer
   - Train the last layer on new dataset (usually a linear classifier such as logistic regression or softmax)
2. Fine-tuning
   - Decide to freeze first n layers
   - Train the remaining layers and stop backpropagation at layer n
   - In the limit fine-tuning can be applied to all layers

Transfer Learning in NN: Freeze Layers



How to do Transfer Learning

TARGET

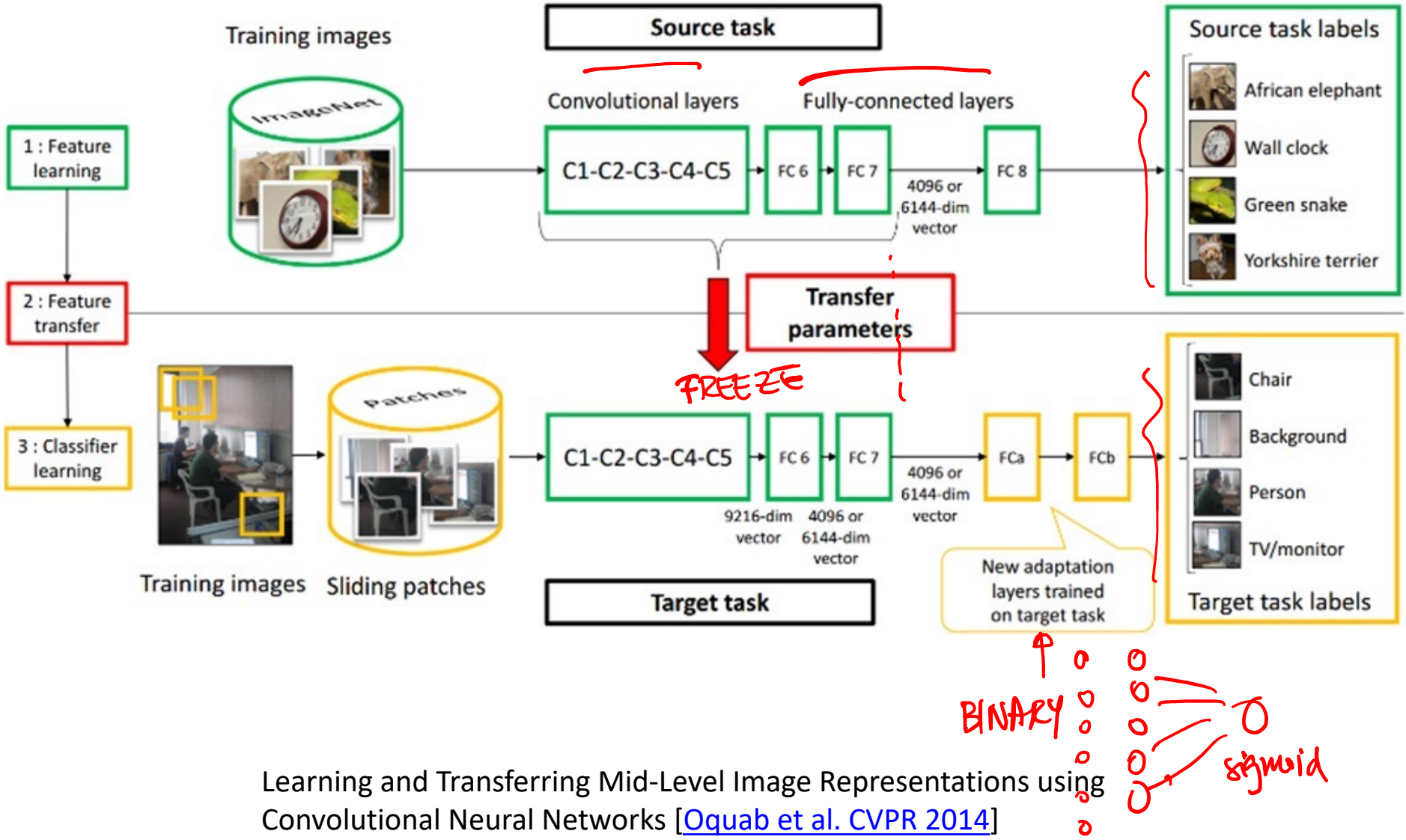
Dataset size	Dataset similarity	Recommendation
Large	Very different	Train model B from scratch, initialize weights from model A
Large	Similar	OK to fine-tune (less likely to overfit)
Small	Very different	Train classifier using the earlier layers (later layers won't help much)
Small	Similar	Don't fine-tune (overfitting). Train a linear classifier

Learning Rates

- Training linear classifier: typical learning rate
- Fine-tuning: use smaller learning rate to avoid distorting the existing weights

Transfer Learning Applications

- Image classification (most common): learn new image classes
- Text sentiment classification
- Text translation to new languages
- Speaker adaptation in speech recognition
- Question answering



Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks [Oquab et al. CVPR 2014]

Acknowledgements

- Slides made using resources from:
  - Andrew Ng
  - Eric Eaton
  - David Sontag
  - Andrew Moore
  - Yann LeCun
- Thanks!