deeplearning.ai

Error Analysis

Carrying out error analysis

Look at dev examples to evaluate ideas

Should you try to make your cat classifier do better on dogs?

Error analysis:
- Get ~100 mislabeled dev set examples.
- Count up how many are dogs.

Evaluate multiple ideas in parallel

Ideas for cat detection:

- Fix pictures of dogs being recognized as cats
- Fix great cats (lions, panthers, etc..) being misrecognized
- Improve performance on blurry images

Image		
1		
2		
3		
⋮		
% of total		

deeplearning.ai

Error Analysis

Cleaning up Incorrectly labeled data

Incorrectly labeled examples

x

y 1 0 1 1 0 1 1

DL algorithms are quite robust to random errors in the training set.

Error analysis

Image	Dog	Great Cat	Blurry	Incorrectly labeled	Comments
…					
98				✓	Labeler missed cat in background
99		✓			
100				✓	Drawing of a cat; Not a real cat.
% of total	8%	43%	61%	6%	

Overall dev set error

Errors due incorrect labels

Errors due to other causes

Goal of dev set is to help you select between two classifiers A & B.

Correcting incorrect dev/test set examples

- Apply same process to your dev and test sets to make sure they continue to come from the same distribution

- Consider examining examples your algorithm got right as well as ones it got wrong.

- Train and dev/test data may now come from slightly different distributions.

deeplearning.ai

Error Analysis

Build your first system quickly, then iterate

Speech recognition example

- Noisy background
  - Café noise
  - Car noise
- Accent
- Far fro
- Young
- Stutter
- …

Guideline:

Build your first system quickly, then iterate

- Set up dev/test set and metric
- Build initial system quickly
- Use Bias/Variance analysis & Error analysis to prioritize next steps.

deeplearning.ai

Mismatched training and dev/test data

Training and testing on different distributions

Cat app example

Data from webpages

Data from mobile app

Andrew Ng

Speech recognition example

Training	Dev/test
Purchased data	Speech activated rearview mirror
Smart speaker control	
Voice keyboard	
…	

deeplearning.ai

Mismatched training and dev/test data

Bias and Variance with mismatched data distributions

Cat classifier example

Assume humans get $\approx 0\%$ error.

Training error
Dev error

Training-dev set: Same distribution as training set, but not used for training

Bias/variance on mismatched training and dev/test sets

More general formulation

deeplearning.ai

Mismatched training and dev/test data

Addressing data mismatch

Addressing data mismatch

- Carry out manual error analysis to try to understand difference between training and dev/test sets

- Make training data more similar; or collect more data similar to dev/test sets

Artificial data synthesis

"The quick brown fox jumps over the lazy dog." + Car noise = Synthesized in-car audio

Artificial data synthesis

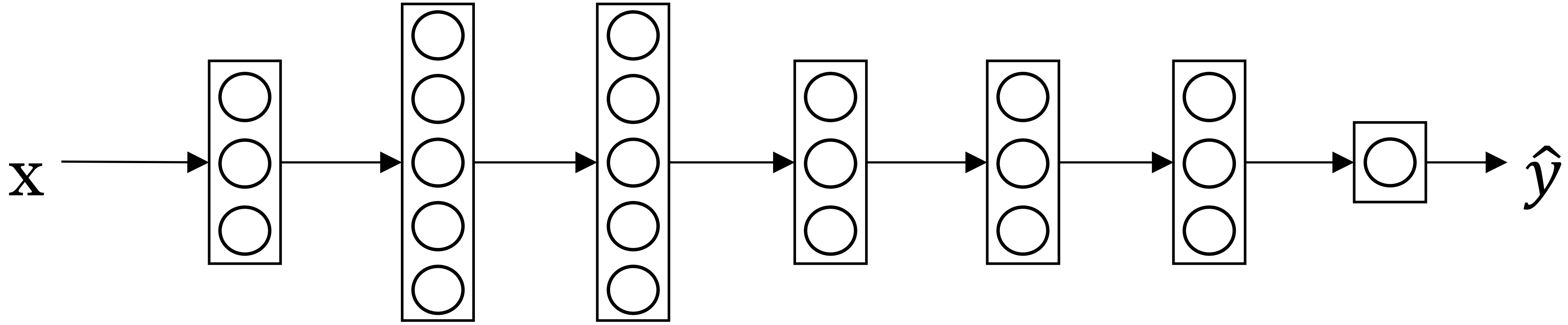
Car recognition:

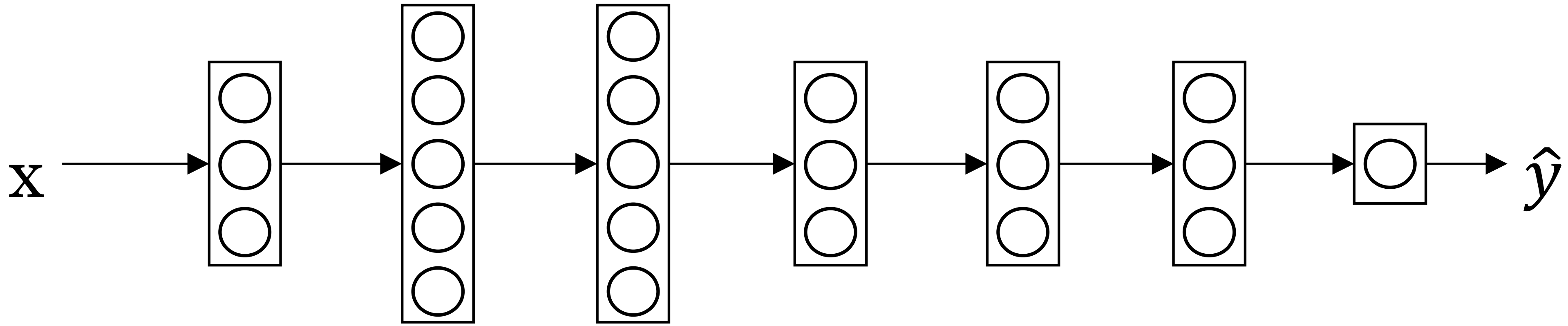
deeplearning.ai

Learning from multiple tasks

Transfer learning

Transfer learning





When transfer learning makes sense

- Task A and B have the same input x.
- You have a lot more data for Task A than Task B.
- Low level features from A could be helpful for learning B.

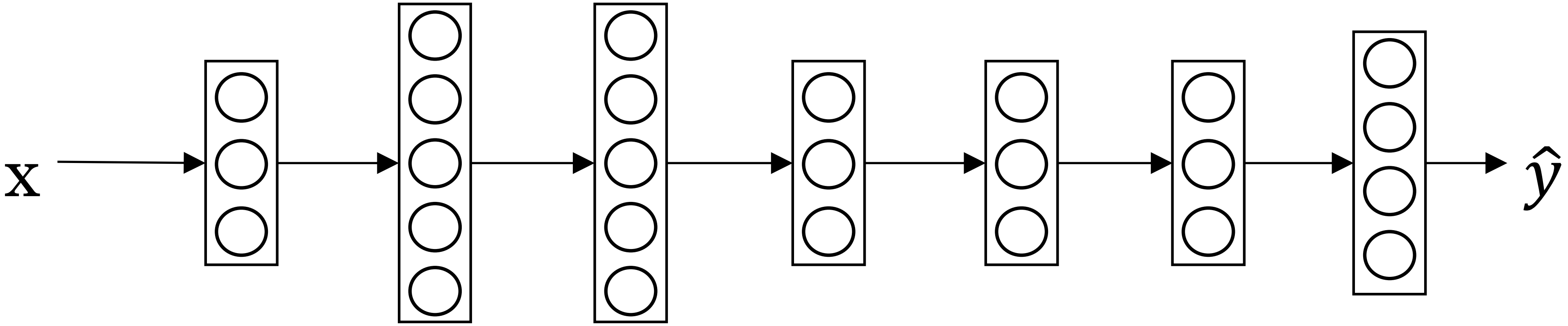
deeplearning.ai

Learning from multiple tasks

Multi-task learning

Simplified autonomous driving example

Neural network architecture



When multi-task learning makes sense

- Training on a set of tasks that could benefit from having shared lower-level features.
- Usually: Amount of data you have for each task is quite similar.

- Can train a big enough neural network to do well on all the tasks.

deeplearning.ai

End-to-end deep learning

What is end-to-end deep learning

What is end-to-end learning?

Speech recognition example

Face recognition

[Image courtesy of Baidu]

More examples

Machine translation

Estimating child's age:

deeplearning.ai

End-to-end deep learning

Whether to use end-to-end learning

Pros and cons of end-to-end deep learning

Pros:

- Let the data speak
- Less hand-designing of components needed

Cons:

- May need large amount of data
- Excludes potentially useful hand-designed components

Applying end-to-end deep learning

Key question: Do you have sufficient data to learn a function of the complexity needed to map x to y?