Week 1

Wednesday, November 15, 2017 8:07 PM

Orthogonalization

Hone in on parameters that are independent to expedite results

Fitting training set: Larger networks, better algorithm

Fitting dev set: Regularization, larger training set

Fitting test set: Bigger dev set

Real world performance: Change dev set or cost function

Setting Up Your Goal

Wednesday, November 15, 2017 8:23 PM

Single Number Evaluation Metric

Precision: $\frac{\text{correct classifications}}{\text{positive classifications}}$, Recall: $\frac{\text{correct classifications}}{\text{positive examples in dataset}}$

F_1 score = "Avg" of precision & recall $\left(\frac{2}{\frac{1}{P}+\frac{1}{R}}\right)$ ← Harmonic mean

Multi-class: Take avg. of error over each class

Optimizing & Satisficing Metrics

Maximize accuracy (optimizing) under constraint $t_{run} \leq 100$ ms (satisficing metric)

N metrics: 1 optimizing, N-1 satisficing

Ex. Wake words / Trigger words

"Alexa", "OK Google", "Nihao, Baidu", etc.

Maximize accuracy, $\leq$ 1 false positive per 24 hrs.

Train / dev / test Set Distributions

Training / dev / test Set Distributions

Regions — Bad — Good

- US
- UK
- Europe
- S. America
- India
- China
- other Asia
- Australia

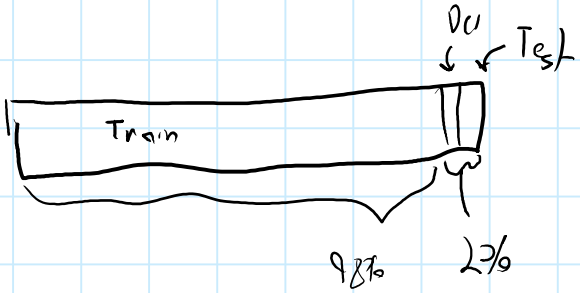
Dev (US, UK, Europe, S. America) / Test (India, China, other Asia, Australia)

Shuffle & divide

Dev & test sets should come from same distribution

Size of Dev & Test sets

10^7 examples



When to Change Dev/Test set metrics

e.x, Cat image app porn delivery

Metric: classification error

Algorithm A: 3% error, errors contain porn

Algorithm B: 5% error, no porn

Algorithm B: 5% error, no porn

$$\text{Error} \quad \frac{1}{\sum_{i=1}^{m_{dev}} w^{(i)}} \sum_{i=1}^{m_{dev}} w^{(i)} \mathbb{1}\{y_{pred}^{(i)} \neq y^{(i)}\}$$

problem: mislabels of porn should be weighted more harshly

$$w^{(i)} \begin{cases} 1 & \text{if } x^{(i)} \neq \text{porn} \\ 10 & \text{if } x^{(i)} = \text{porn} \end{cases}$$ (must label images as porn or not)

ex. User images

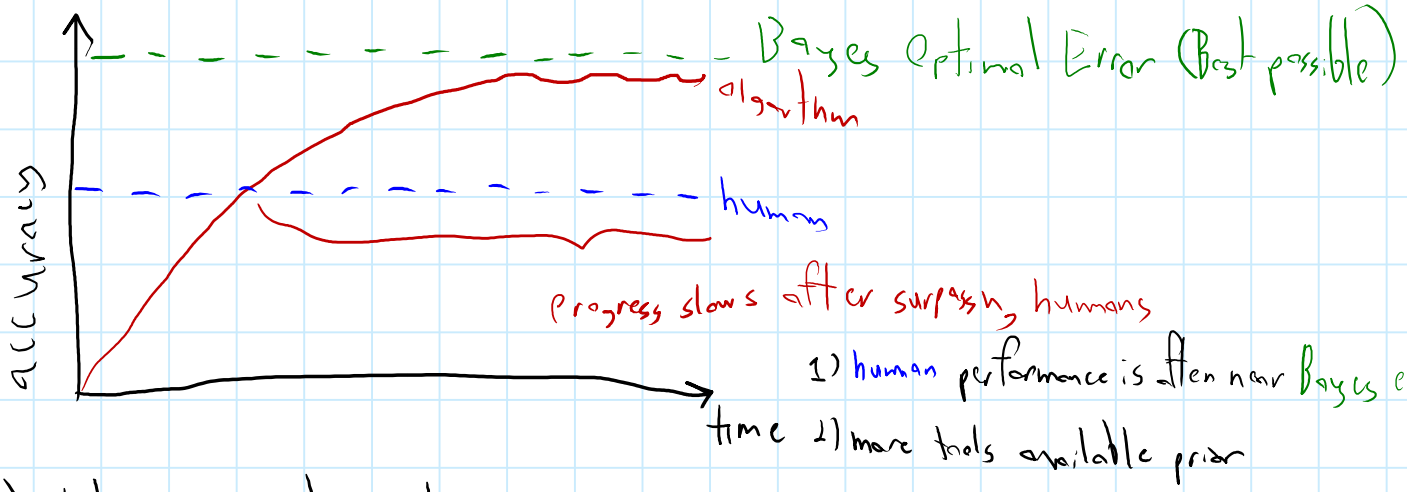
Alg A: 3% error, only ID's high quality images

Alg B: 5% error, ID's user uploaded low quality images

Comparing to Human Performance

Sunday, November 19, 2017 11:23 AM

Why human level performance?



While worse than humans:

- Get human labeled data
- Manual error analysis
- Better analysis of bias & variance

Avoidable Bias

ex. Cat classification

Human	1	7.5
Training error	8	8
Dev error	10	10
Goal	focus on bias	focus on variance

"Avoidable bias"

Variance

human-level error is a proxy for Bayes error

Understanding Human-level Performance

ex. Medical image classification

	error (%)
a) typical human	3
b) typical doctor	1 ← useful result, still deployable
c) experienced doctor	0.7
d) team of c	0.5 ← standard for human level (Bayes proxy)

Human	1 / 0.5	0.5
Training	5	0.7
Dev	6	0.8
	reduce bias	reduce bias

Bayes error is NOT always zero! (noisy data, etc.)

Surpassing human performance

hard to quantify avoidable bias when better than human

ML already does better with structured data such as ads & logistics where there are massive amounts of data

Improving model performance

2 assumptions

1) Fit training set well (Avoidable bias)

2) Training set generalizes well to dev/test (variance)

Human level

↕ Avoidable bias

Training error

↕ variance

Dev error

Train bigger model, train longer, better algorithms
NN architecture, hyperparameter tuning

More data, regularization (L2, dropout, data augmentation)
NN architecture, hyperparameter tuning

Week 2

Wednesday, November 22, 2017 4:30 PM

Error Analysis

Wednesday, November 22, 2017 4:30 PM

Carrying Out Error Analysis

Look at dev examples to eval ideas
(cat classifier; learn dogs)
Get ~100 mislabeled dev set examples
Count how many are dogs
5/100, potential improvement is very small
50/100, half error, worthwhile effort

Evaluate multiple ideas in II

Image	Idea 1	Idea 2	Idea 3	Comments
1				
2				
3				
⋮				
%	8	43	61	prioritize

Cleaning up Mislabeled data (human labeled)

DL algorithms are fairly robust to random error if they're a small amount in training set

Add column to error analysis for mislabeled examples

Is it worthwhile to correct?

			A	B
Overall dev error	10	2	2.1	1.9
Incorrect labels	0.6	0.6	0.6	0.1
Other	9.4	1.4	1.5	1.8
	✗	✓	higher ceiling, A can be better	

Build Fast, then Iterate

Use Bias/Variance + error to prioritize next steps

Exceptions: large amounts of academic lit on topic

Mismatched Training and Dev/test Set

Thursday, November 23, 2017 1:34 PM

Training & testing on different distributions

Cat app example

Data from web (200k) — professionally shot

Data from mobile app (10k) — more common

combine & shuffle

✗ Option 1: train 205k; dev/test 2.5k ea. — dev set still optimizes for web

✓ Option 2: train all 200k web + 5k user; dev/test rest of mobile app — targeting app images

Speech Recognition ex. Car

Training (500k)
Purchased Data
Smart speaker
Voice Keyboard
⋮

Dev/test (20k)
Speech activated rearview mirror

train 500k + 500k + 10k; dev/test 10k ea.; 5k ea — both acceptable

Bias & Variance w/ mismatched data distributions

human error	~0
Training	1
Dev	10

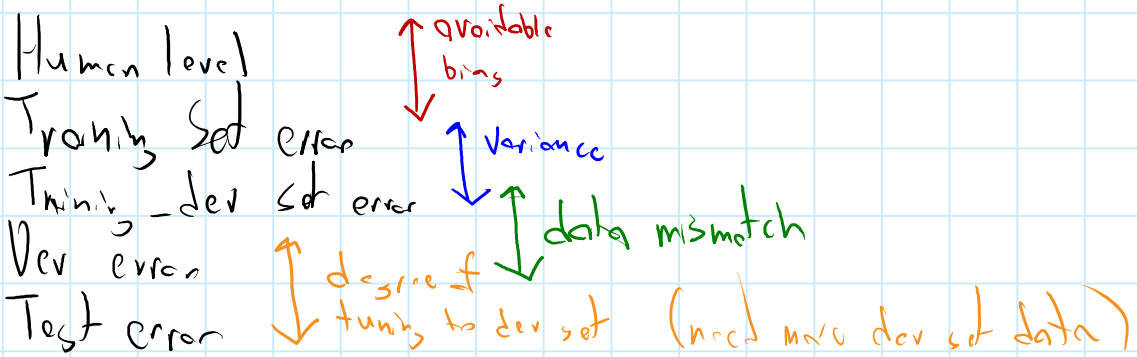
Are they the same distribution? If not, may be nothing wrong, other than poor generalization due to data mismatch

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

Train	1	1	10	10
Train-dev	9 (variance)	1.5 (data mismatch)	11	11
Dev	10	10	12	20

(unavoidable bias)

General Principles



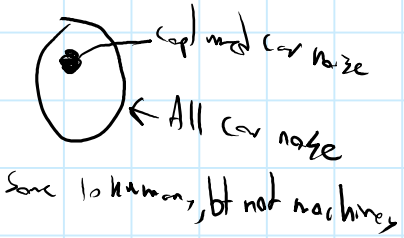
Addressing Data Mismatch

Manual error analysis to understand differences between training & dev/test sets

Make training data more similar (e.g. add car noise), or collect more data similar to dev/test sets

Data synthesis:

noise	Car noise
looking	1 hr



Some to human, but not machines

Car recognition

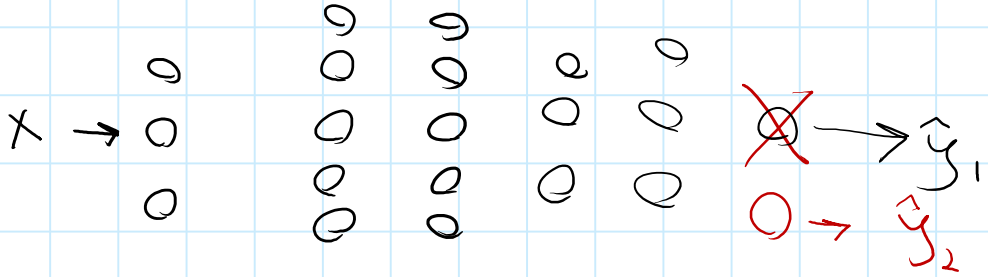
Computer graphics? Game w/ LL cars looks good, but not realistic representation

Learning for Multiple Tasks

Saturday, November 25, 2017 12:00 PM

Transfer Learning

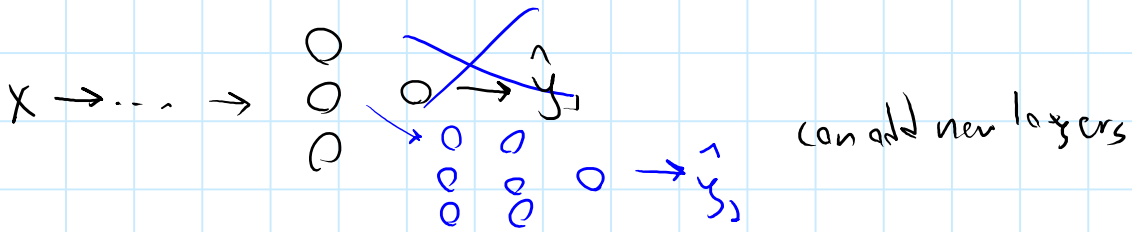
image recognition → radiology change final weights



Retrain on new (x, y) pairs

Little data: only retrain last layer

Another ex: speech recognition → wake word



This factor is useful when target area (like radiology) has fewer examples, but being a subset of images many transfer learning useful

abstract	fine	
lots	lots or 1	Use xfer learning
lots	little or 0	May not hurt, but likely not very helpful

Summarized rules for when to use

Task A & B have same input

$m_A >> m_B$

Low level features of A are helpful for B

Multi-Task Learning

e.g. self driving car images w/ multiple components present, (car + stop sign)

$x \to \cdots \to \begin{matrix} 0 \\ 0 \\ 0 \\ 0 \end{matrix} \to \hat{y}$ (person, car, sign, light) $\quad \hat{y} \in \mathbb{R}^{4,1}$

Loss: $\frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{4} \mathcal{L}(\hat{y}_j^{(i)}, y_j^{(i)})$

↑ logistic loss

$-y_j^{(i)} \log(\hat{y}_j^{(i)}) - (1 - y_j^{(i)}) \log(\hat{y}_j^{(i)})$

Unlike softmax, multiple labels are ok

Training 1 NN to ID multiple aspects can perform better than separate NNs provided underlying structures are the same (e.g. images)

Generalizes ok to missing data in y, (e.g. road images only marked w/ presence of a car) & just skips NaN values

When to use?

Share low-level features
Amount of data for each task is similar
Can train large enough NN to do well on all tasks

End-to-End Deep Learning

Saturday, November 25, 2017 6:24 PM

What is it?

Simplifies process

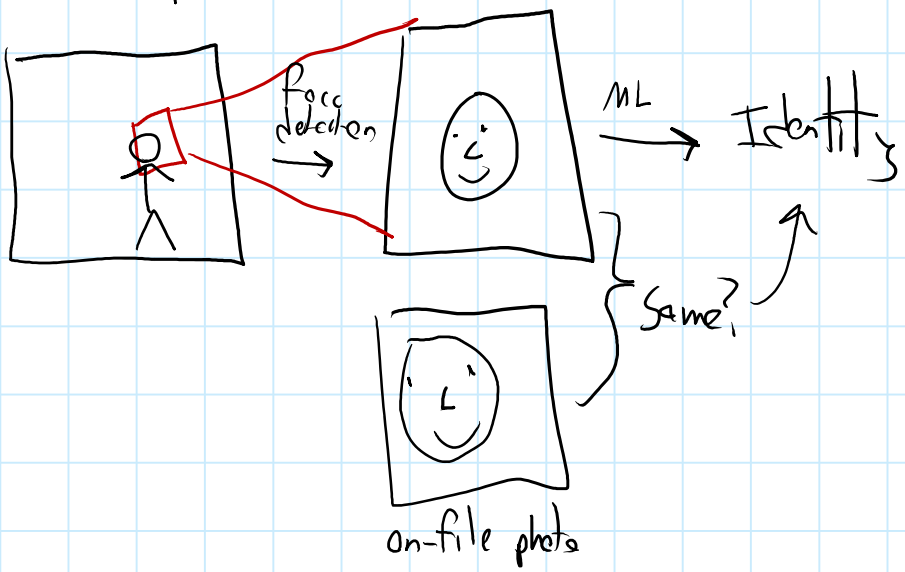
X

audio → (MFCC) features → (ML) phonemes → words → transcript — y

audio → (End-to-End ML) → transcript

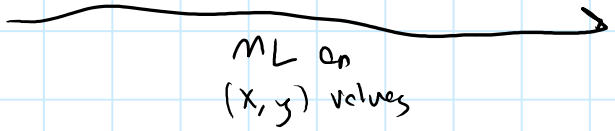
Need large amounts of data to be useful

Facial rec, end-to-end not well suited



Machine translation

English → text analysis → ..... → French



Child's age based on hand X-ray

Image → bones → bone lengths → table of avg lengths for ages → age

not enough pictures to train on

When to use end-to-end?

Pros:

Lets data speak, no human preconceptions getting in the way of optimization
Less hand-designing of components needed

Cons:

Large amounts of data needed
Excludes potentially useful hand designed components

Key question: Is there enough data for how complex your problem is?

end-to-end is not at all suited to self-driving cars