STRUCTURING ML PROJECTS

#orthogonalization

'You need 'knobs' to tune something acc to your preference.
And you need these 'knobs' to control only their corresponding feature/setting

Chain Assumptions in ML

- Fit training set well on cost fn → better network, Adam
- Fit dev set well on cost fn → Regularization, Bigger train set
- Fit test set well on cost fn → Bigger/better dev set
- Performs well in real world → change dev set or cost fn

* Single no evaluatⁿ metric

For classifiers, we usually use two evaluatⁿ metrices - Precision & recall. ~~To~~ And often there's a trade off b/w the two.
These can be reduced to a single metric - the F1 score : $\frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$

Dev set + having a single evaluatⁿ metric speeds up the model selection process

*Satisficing & optimizing metric

Lets say we have two constraints
accuracy & running time

eg we want the max accuracy for time ≤ 100 ms

This is usually used in wakewords in
Speech recognition.
wakewords: Alexa, ok Google, Hey Siri etc.

* Train / dev / test sets
Should come from the same dist.
-Shuffle your dataset

* When to change dev/test sets & metrics
- Classifier algorithms ~~to~~ to classify cats:
Model A works with 3% ~~accuracy~~ error
You dont want the users to be
displayed with a pornographic
image [in the 3% error range]

You have the metric

$$\frac{1}{m} \sum \mathcal{I}(\hat{y} \neq y)$$

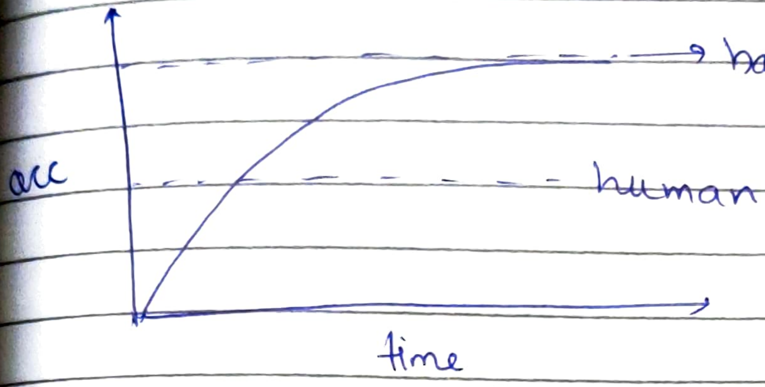
add another variable here

$$\frac{1}{m} \sum w \, \mathcal{I}(\hat{y} \neq y)$$

st

$$w = \begin{cases} 1 & \text{non porn} \\ 100 & \text{porn} \end{cases}$$

Comparing to Human Level performance



bayes error : the least possible error % which cannot be surpassed no matter how good the algo/fn is.

In case the model gives an accuracy rate below the human accuracy rate.
- get labelled data from humans (x, y)
- gain insight from manual error analysis
- better analysis of bias/variance

* Avoidable Bias

Lets say the images were blurry so human acc dropped

	errors	
Human	1%	7.5%
Train Set	8%	8%
Dev set	10%	10%

bias

variance

put your efforts here

put your efforts here

Human level error is often used as a proxy for bayes error in computer vision/speech recognition tasks etc

~~Suppose~~ Consider a medical image classification

	Error
Typical human	3%
Typical doctor	1%
experienced doc	0.7%
team of exp docs	0.5 %

← bayes error ≤ 0.5%

Problems where human level performance is often surpassed

- loan approval
- online advertising
- logistics
- product recommendations

* Improving your Model

Assumptions in supervised Learning

1. The model fits training set well
   ~ avoidable bias
2. The training set generalizes well on the dev set
   ~ variance

Human-level

↕ bias

Training error

↕ variance

Dev error

{ Train~~ing~~ a bigger model
Train longer/ optimize
momentum/ rms/ adam
NN arch/ hyperparameter search }

{ more data
regularizat^n
'L2, dropout
data augmentation }

PAGE NO.:
DATE: / /

Error Analysis

Incorrectly labelled examples

- DL algos are quite robust to random errors ↓ Some dog was incorrectly labelled as cat.

~~but not to~~

systematic error ↓ all of the white colored pups were labelled as cats

add an additional column for incorrectly labelled examples.