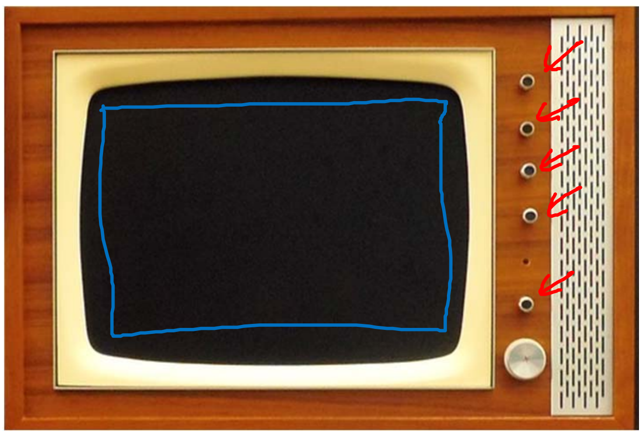
deeplearning.ai

# Introduction to ML strategy

## Orthogonalization

# TV tuning example



Orthogonalization

0.1 x

+ 0.3 x

- 1.7 x

+ 0.8 x

+ . . .

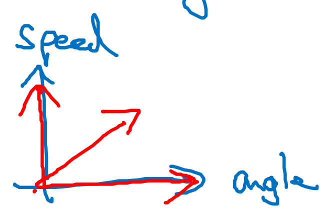
## Car

→ Steering

→ { Accelerating, Braking }

→ 0.3 x angle − 0.8 speed

→ 2 x angle + 0.9 speed.



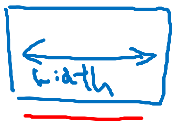

Andrew Ng

# Chain of assumptions in ML

→ Fit training set well on cost function

(≈ human-level performance)

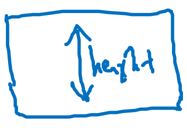
bigger network
Adam
...




early stopping

→ Fit dev set well on cost function

Regularization
Bigger train set




→ Fit test set well on cost function

Bigger dev set

→ Performs well in real world

(Happy cat pic app users.)

Change dev set or cost function

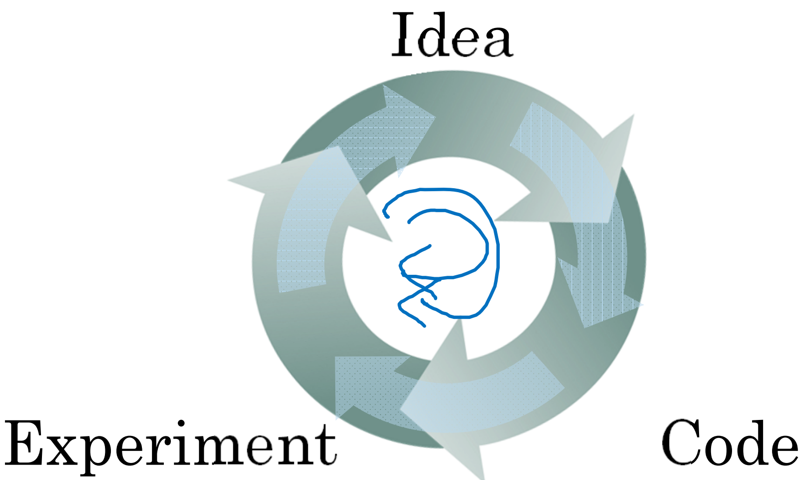
Andrew Ng

deeplearning.ai

# Setting up your goal

# Single number evaluation metric

# Using a single number evaluation metric




→ Of examples recognized as cat, what % actually are cats?

→ what % of actual cats are correctly recognized

| Classifier | Precision | Recall |
|---|---|---|
| A | 95% | 90% |
| B | 98% | 85% |

$F_1$ score = "Average" of P and R.

$\left( \frac{2}{\frac{1}{P} + \frac{1}{R}} \right.$ . "Harmonic mean" $\left. \right)$

Dev set + Single number evaluation metric

real

Speed up iterating

Andrew Ng

# Another example

| Algorithm | US | China | India | Other |
|---|---|---|---|---|
| A | 3% | 7% | 5% | 9% |
| B | 5% | 6% | 5% | 10% |
| C | 2% | 3% | 4% | 5% |
| D | 5% | 8% | 7% | 2% |
| E | 4% | 5% | 2% | 4% |
| F | 7% | 11% | 8% | 12% |

deeplearning.ai

# Setting up your goal

# Satisficing and optimizing metrics

# Another cat classification example

optimizing — Satisficing

| Classifier | Accuracy | Running time |
|---|---|---|
| A | 90% | 80ms |
| B | 92% | 95ms |
| C | 95% | 1,500ms |

cost = accuracy − 0.5 × running Time

maximize accuracy

subject to running Time ≤ 100 ms.

N metrics : 1 optimizing

N−1 satisficing

Wakewords / Trigger words

Alexa, Ok Google,

Hey Siri, nihaobaidu

你好百度

accuracy.

#false positive

maximize accuracy.

s.t. ≤ 1 false positive every 24 hours.

Andrew Ng

deeplearning.ai

Setting up your goal

# Train/dev/test distributions

# Cat classification dev/test sets

development set, hold out cross validation set

Regions:

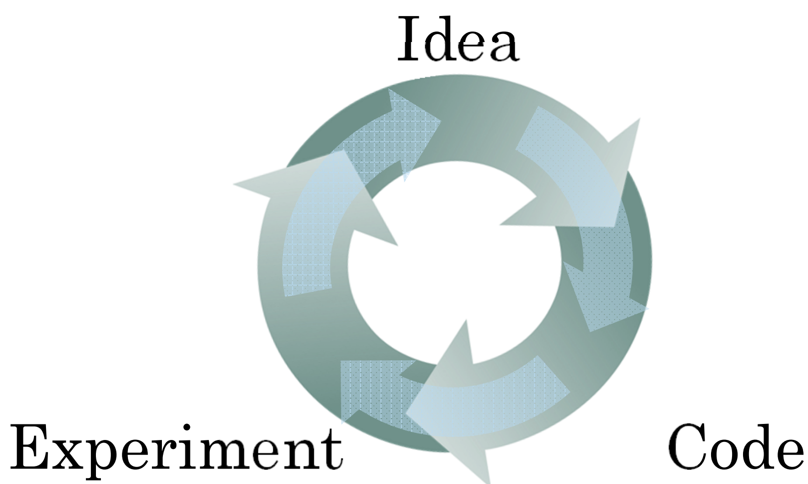
- US
- UK
- Other Europe
- South America
- India
- China
- Other Asia
- Australia

Dev

Test

Randomly shuffle into dev/test

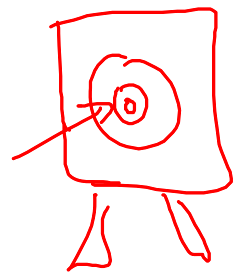
dev set + metric



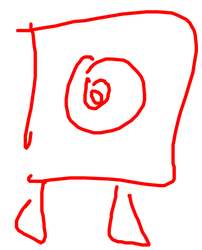
# True story (details changed)

Optimizing on dev set on loan approvals for medium income zip codes

$x \rightarrow y$ (repay loan?)



Tested on low income zip codes



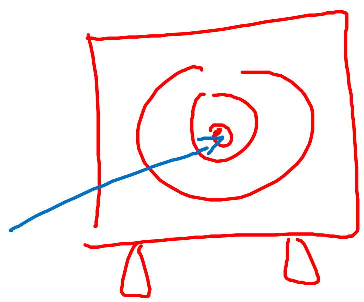
$\sim 3$ month

# Guideline

Same distribution

Choose a dev set and test set to reflect data you expect to get in the future and consider important to do well on.

training



dev

metric

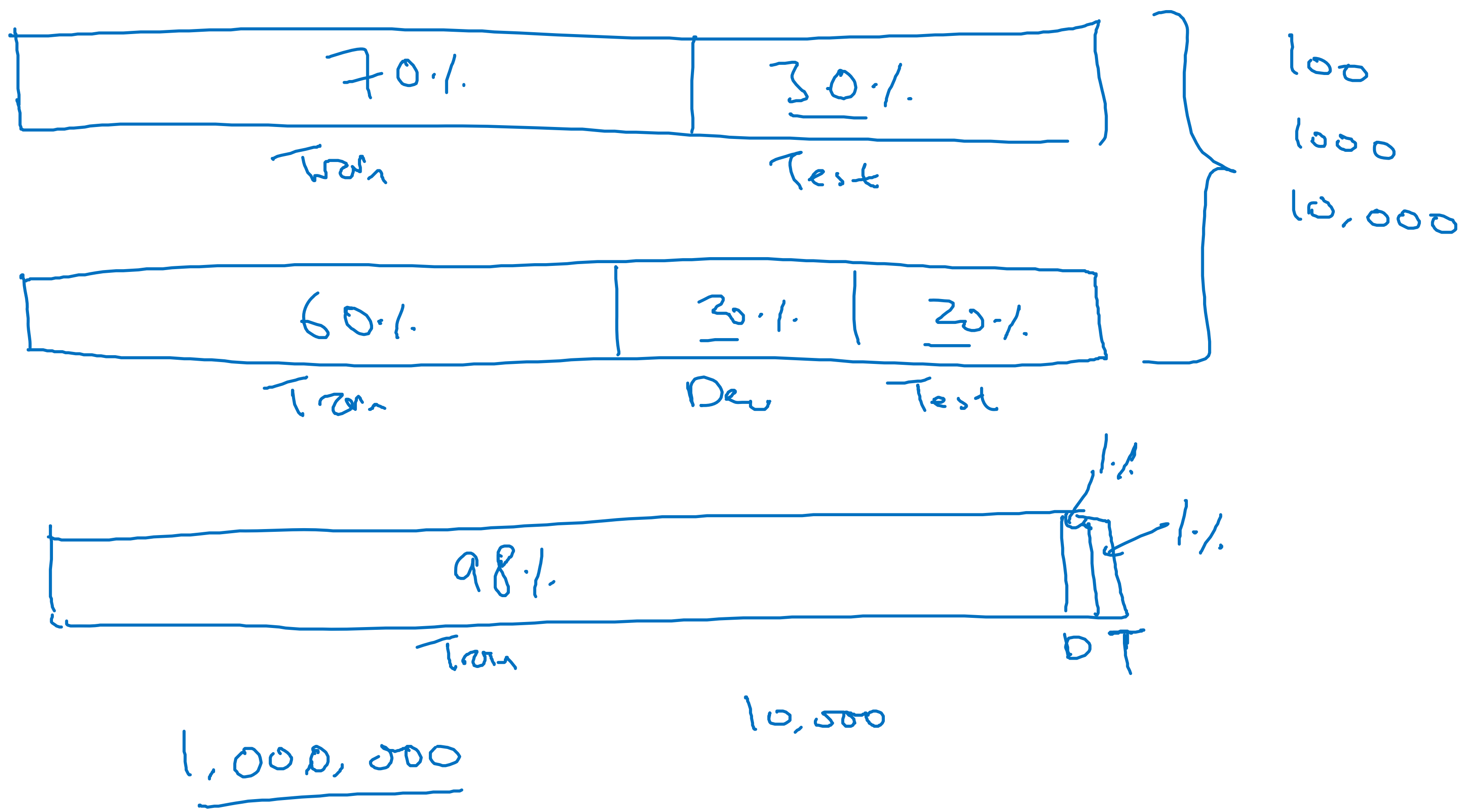
test

deeplearning.ai

# Setting up your goal

## Size of dev and test sets

# Old way of splitting data




Andrew Ng

# Size of dev set

A B

Set your dev set to be big enough to detect differences in algorithm/models you're trying out.

100 : small

1%

1,000

10,000

100,000

A B

97% → 97.1%

0.1%

0.01%

0.001%

Online advertising

Andrew Ng

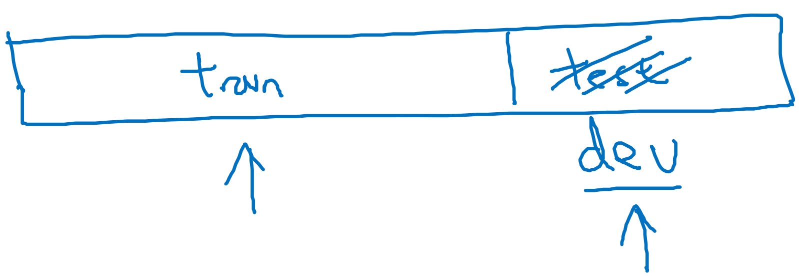
# Size of test set

- Set your test set to be big enough to give high confidence in the overall performance of your system.

10,000

100,000

Train + Dev. No test set.




Andrew Ng

deeplearning.ai

Setting up your goal

# When to change dev/test sets and metrics

# Cat dataset examples

Metric + Dev : Prefer A
You/users : Prefer B.

→ Metric: classification error

Algorithm A: 3% error → pornographic

✓ Algorithm B: 5% error

$$\text{Error}: \frac{1}{\sum_i w^{(i)}} \cancel{\frac{1}{m_{dev}}} \sum_{i=1}^{m_{dev}} w^{(i)} \, \mathcal{I}\{ \underbrace{y_{pred}^{(i)}}_{\text{predicted value (0/1)}} \neq y^{(i)} \}$$

$$\rightarrow w^{(i)} = \begin{cases} 1 & \text{if } x^{(i)} \text{ is non-porn} \\ 10 & \text{if } x^{(i)} \text{ is porn} \end{cases}$$

# Orthogonalization for cat pictures: anti-porn

1. So far we've only discussed how to define a metric to evaluate classifiers. ← Place target

2. Worry separately about how to do well on this metric.

Aim/shoot at target

$$J = \frac{1}{\cancel{m}} \sum_{i=1}^{m} w^{(i)} \mathcal{L}(\hat{y}^{(i)}, y^{(i)})$$

$\sum w^{(i)}$

Andrew Ng

# Another example

Algorithm A: 3% error

Algorithm B: 5% error

Dev/test

User images

If doing well on your metric + dev/test set does not correspond to doing well on your application, change your metric and/or dev/test set.

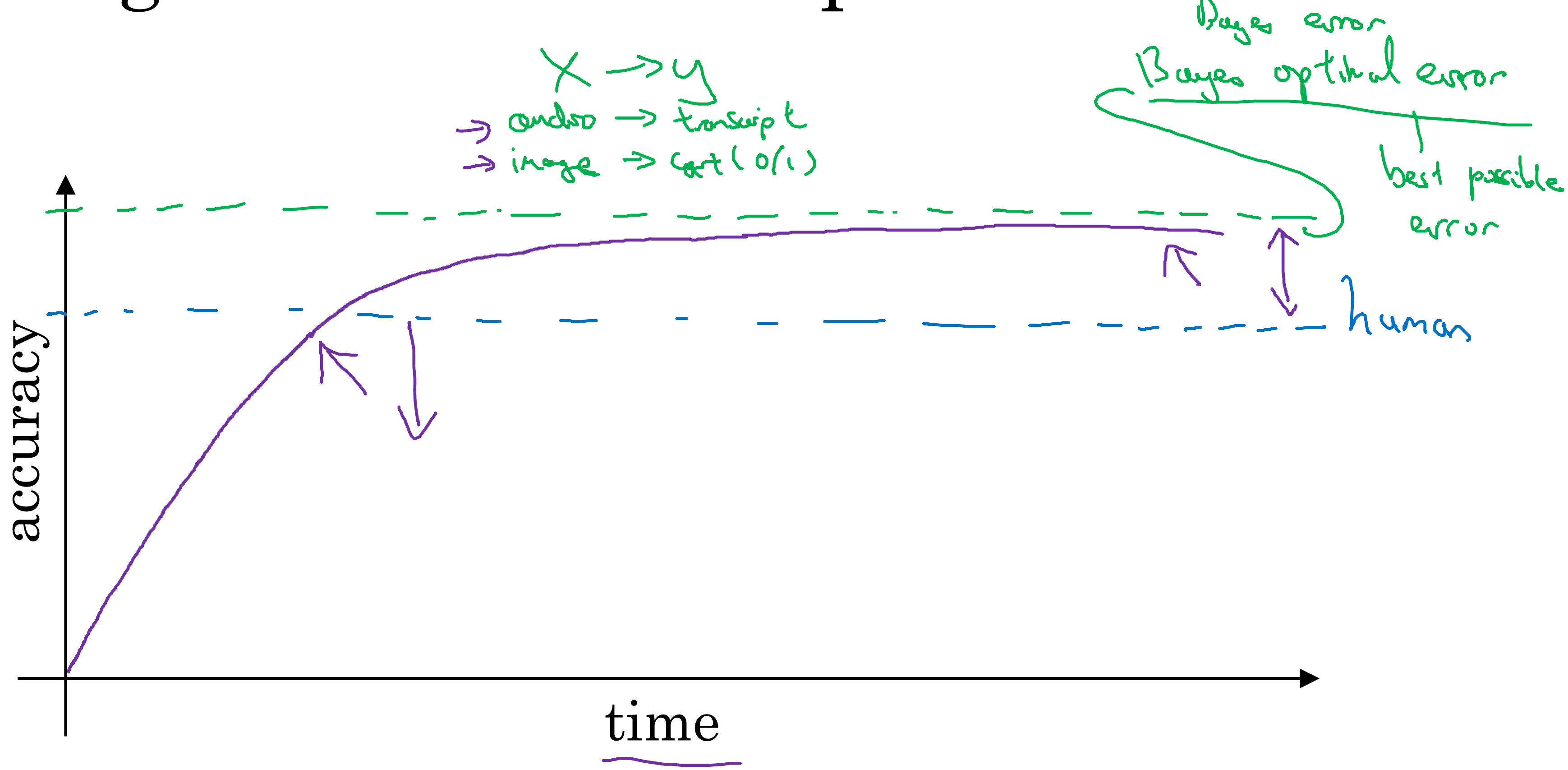
Andrew Ng

deeplearning.ai

# Comparing to human-level performance

## Why human-level performance?

# Comparing to human-level performance




Andrew Ng

# Why compare to human-level performance

Humans are quite good at a lot of tasks. So long as ML is worse than humans, you can:

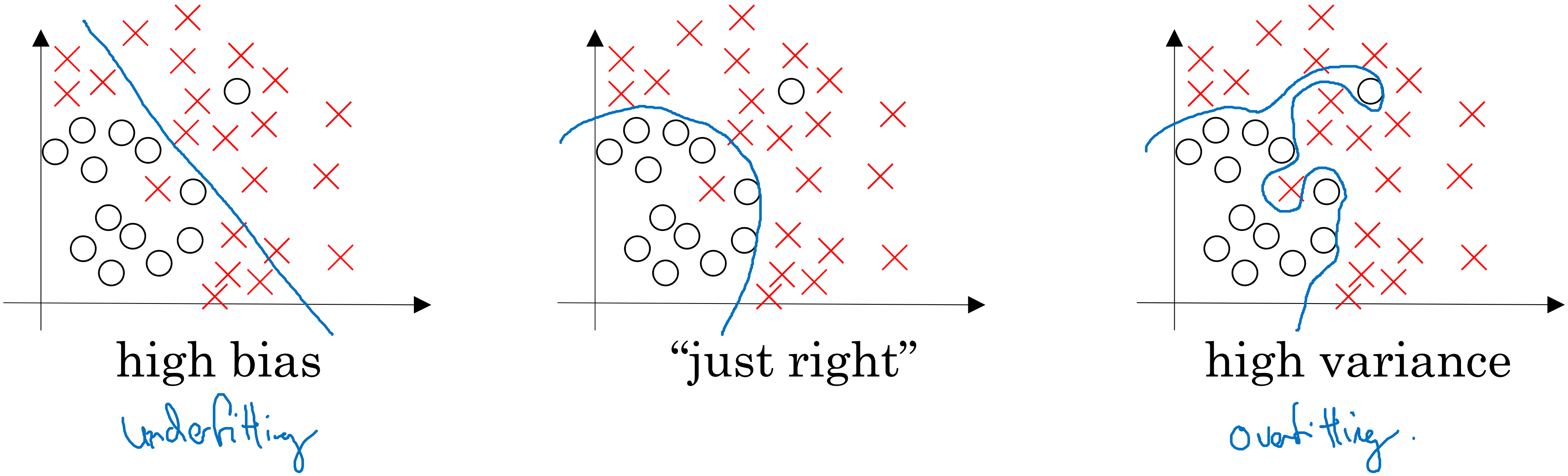
- Get labeled data from humans. $(x, y)$
- Gain insight from manual error analysis:
  Why did a person get this right?
- Better analysis of bias/variance.

deeplearning.ai

# Comparing to human-level performance

## Avoidable bias

# Bias and Variance




Andrew Ng

# Bias and Variance

Cat classification

$\boxed{\text{Human-level} \approx 0\%}$ ....

Training set error: ___ ___

Dev set error:

high variance    high bias    high bias, high variance    low bias, low variance

# Cat classification example

| | | |
|---|---|---|
| Humans (≈ Bayes) | 1% | 7.5% |
| Training error | 8% | 8% |
| Dev error | 10% | 10% |

7%

2%

0.5%

Avoidable bias

2%

Variance

2%

Variance

Focus on bias

Focus on variance

Human-level error as a proxy for Bayes error.

Andrew Ng

deeplearning.ai

# Comparing to human-level performance

## Understanding human-level performance

# Human-level error as a proxy for Bayes error

Medical image classification example:

Suppose:

(a) Typical human .................... 3 % error

(b) Typical doctor ..................... 1 % error

(c) Experienced doctor ............... 0.7 % error

(d) Team of experienced doctors .. 0.5 % error

Bayes error $\leq$ 0.5%

What is "human-level" error?

Andrew Ng

# Error analysis example

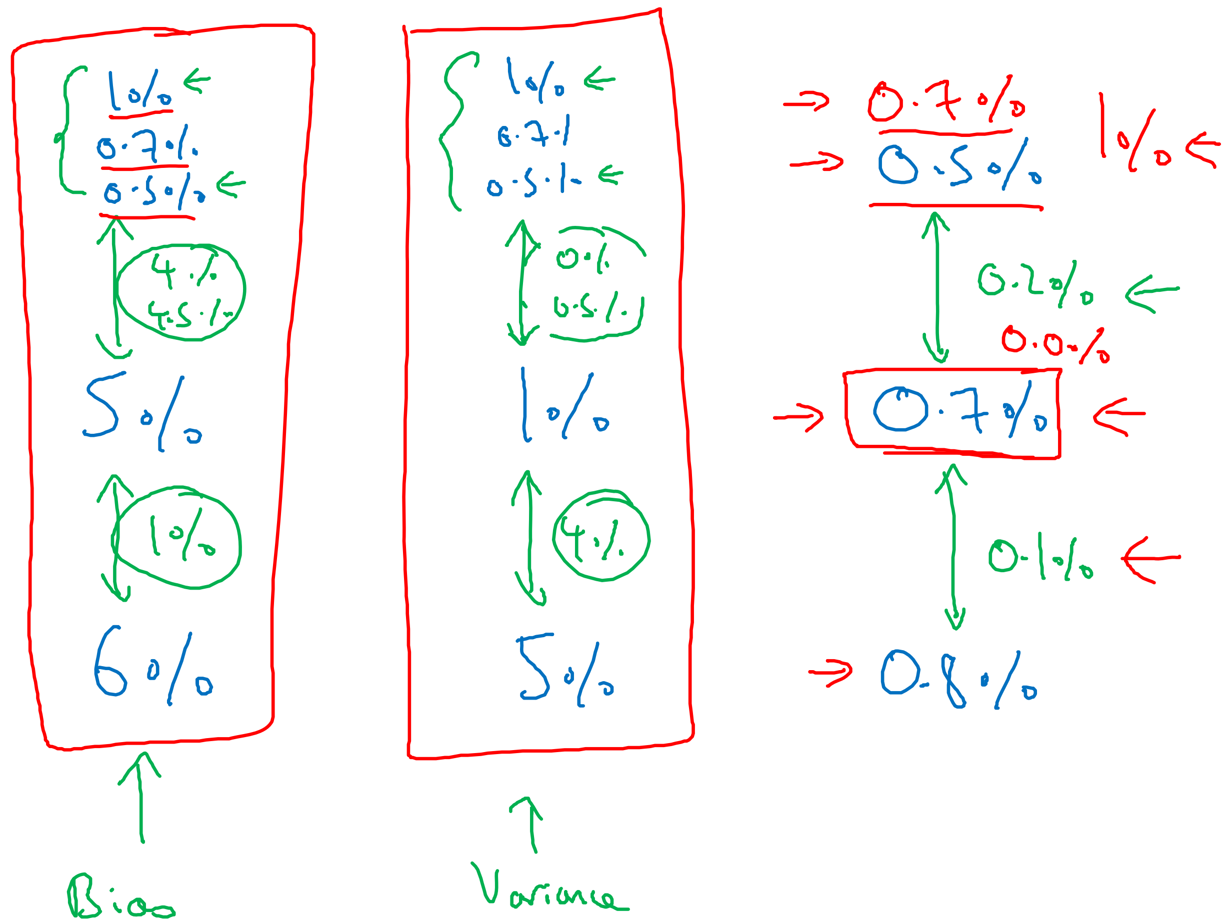
Human (proxy for Bayes error)

Avoidable bias

Training error

Variance

Dev error



# Summary of bias/variance with human-level performance

0%

"Bias"

Human-level error

(proxy for Bayes error)

"Avoidable bias"

Training error

"Variance"

Dev error

deeplearning.ai

# Programming Frameworks

## TensorFlow

# Motivating problem

$J(w) = \boxed{w^2 - 10w + 25}$

(cost)

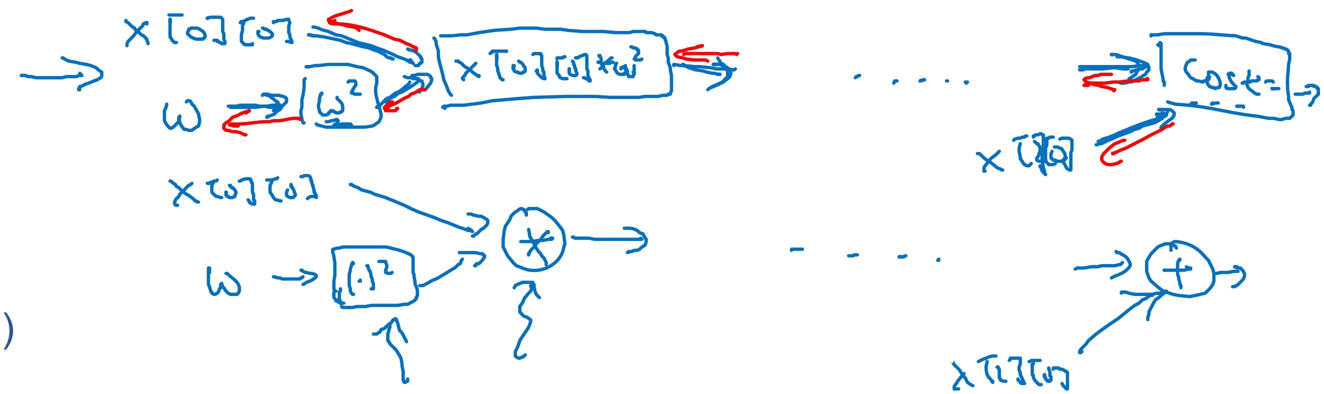
$(w-5)^2$

$w = 5$

$J(w, b)$

Andrew Ng

# Code example




```
import numpy as np
import tensorflow as tf

coefficients = np.array([[1], [-20], [25]])

w = tf.Variable([0],dtype=tf.float32)
x = tf.placeholder(tf.float32, [3,1])
cost = x[0][0]*w**2 + x[1][0]*w + x[2][0]    # (w-5)**2
train = tf.train.GradientDescentOptimizer(0.01).minimize(cost)
init = tf.global_variables_initializer()
session = tf.Session()
session.run(init)
print(session.run(w))

for i in range(1000):
    session.run(train, feed_dict={x:coefficients})
print(session.run(w))
```

```
with tf.Session() as session:
    session.run(init)
    print(session.run(w))
```

Andrew Ng

deeplearning.ai

Comparing to human-level performance

# Surpassing human-level performance

# Surpassing human-level performance

| | | |
|---|---|---|
| Team of humans | 0.5% | 0.5% |
| One human | ~~1%~~ | 1% |
| Training error | 0.6% | 0.3% |
| Dev error | 0.8% | 0.4% |

0.1

0.2

0.1%

0.2%

0.3%

What is avoidable bias?

Andrew Ng

# Problems where ML significantly surpasses human-level performance

- Online advertising
- Product recommendations
- Logistics (predicting transit time)
- Loan approvals

Structured data
Not natural perception
Lots of data

- Speech recognition
- Some image recognition
- Medical
  - ECG, Skin cancer, ...

Andrew Ng

deeplearning.ai

# Comparing to human-level performance

# Improving your model performance

# The two fundamental assumptions of supervised learning

1. You can fit the training set pretty well.

   ~ Avoidable bias

2. The training set performance generalizes pretty well to the dev/test set.

   ~ Variance

# Reducing (avoidable) bias and variance

Human-level

Avoidable bias

Training error

Variance

Dev error

Train bigger model

Train longer/better optimization algorithms

- momentum, RMSprop, Adam

NN architecture/hyperparameters search RNN CNN

More data

Regularization

- $L_2$, dropout, data augmentation

NN architecture/hyperparameters search

Andrew Ng