Probability and Statistics

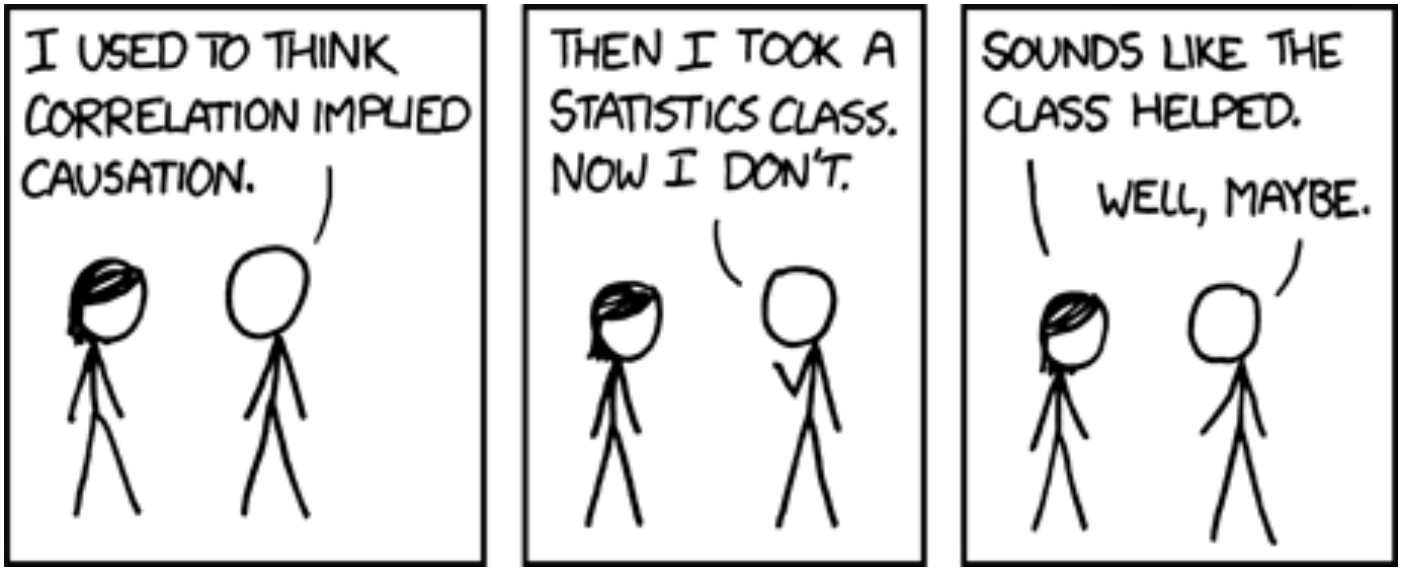
slides adapted from courses.d2l.ai/berkeley-stat-157

Outline

- **Basic Probability**
  Random variables, conditional probabilities, Bayes rule
- **Naive Bayes**
  - Multiple tests
  - Examples - OCR
- **Sampling**
  - Distributions (categorial, normal, uniform)
  - Central limit theorem

slides adapted from courses.d2l.ai/berkeley-stat-157

Basic Probability



slides adapted from courses.d2l.ai/berkeley-stat-157

Probability

Space of events X

- server working; slow response; server broken
- income of the user (e.g. $95,000)
- query text for search (e.g. “statistics tutorial”)

Probability axioms (Kolmogorov)

$$\Pr(X) \in [0, 1], \Pr(\mathcal{X}) = 1$$
$$\Pr(\cup_i X_i) = \sum_i \Pr(X_i) \text{ if } X_i \cap X_j = \emptyset$$

Example queries

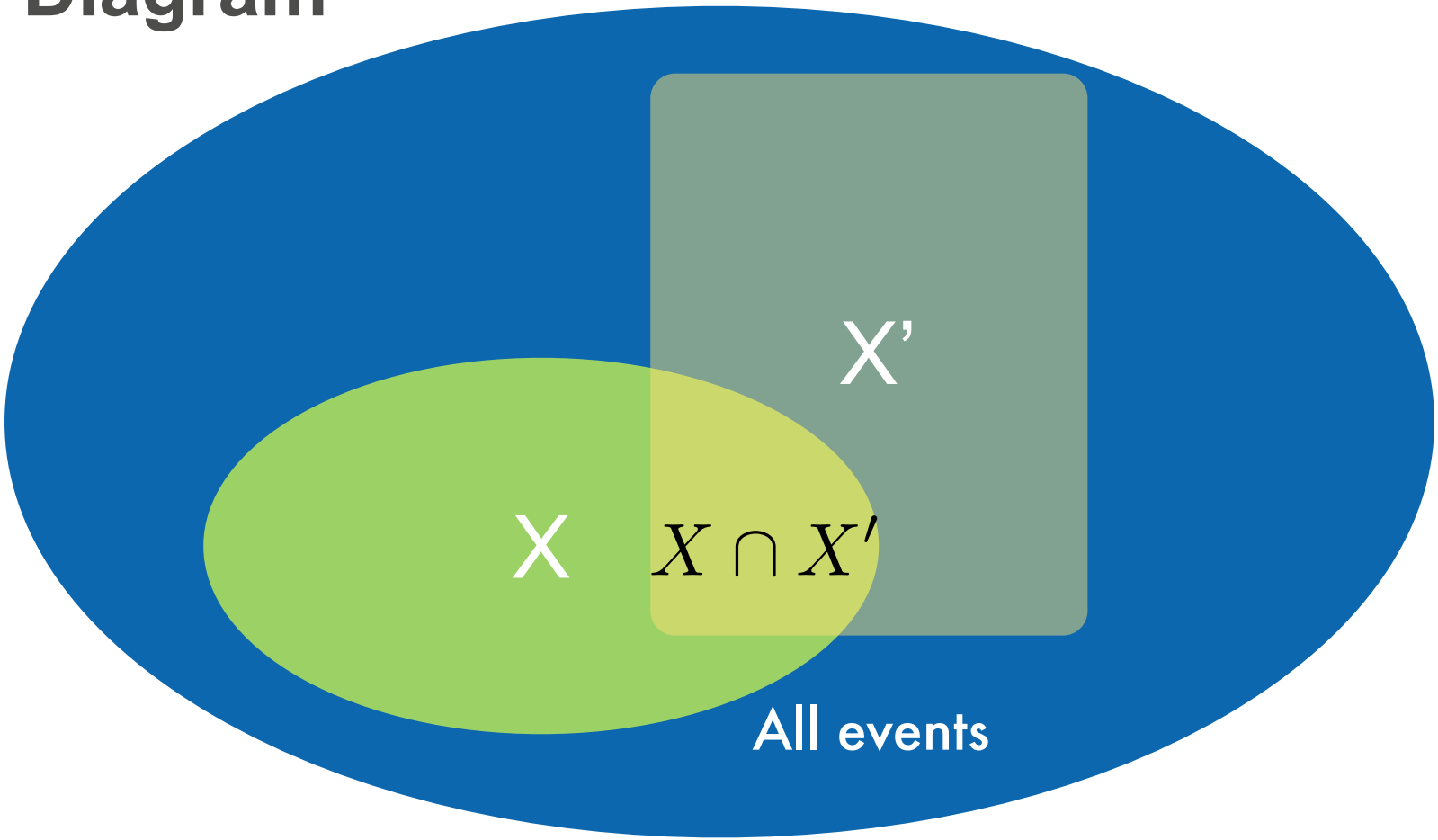
- P(server working) = 0.999
- P(90,000 < income < 100,000) = 0.1

discrete

continuous

slides adapted from courses.d2l.ai/be

Venn Diagram



$$\Pr(X \cup X') = \Pr(X) + \Pr(X') - \Pr(X \cap X')$$

aws

(In)dependence

Independence

- Login behavior of two users (approximately)

$$\Pr(x, y) = \Pr(x) \cdot \Pr(y)$$

Aside: Pr(x,y)>P(x).P(y) - positive dependence
Pr(x,y)<P(x).P(y) - negative dependence

slides adapted from courses.d2l.ai/berkeley-stat-157

Uncertainty and Conditioning

- **Uncertainty**
  - Coin flip (head, tail, edge)
  - Lottery
- **Conditioning**
  - More information makes things more certain (if the information is related). p(y|x) rather than p(y)
  - We can build classifiers, regressors etc.
    (the point of this course)

slides adapted from courses.d2l.ai/berkeley-stat-157

Bayes Rule

- Joint Probability

$$\Pr(X, Y) = \Pr(X|Y)\Pr(Y) = \Pr(Y|X)\Pr(X)$$

- Bayes Rule

$$\Pr(X|Y) = \frac{\Pr(Y|X) \cdot \Pr(X)}{\Pr(Y)}$$

slides adapted from courses.d2l.ai/berkeley-stat-157

AIDS test (Bayes rule)

- Data
  - Approximately 0.1% are infected
  - Test detects all infections
  - Test reports positive for 1% healthy people
- Probability of having AIDS if test is positive

$$\begin{aligned}\Pr(a=1|t) &= \frac{\Pr(t|a=1)\cdot\Pr(a=1)}{\Pr(t)} \\ &= \frac{\Pr(t|a=1)\cdot\Pr(a=1)}{\Pr(t|a=1)\cdot\Pr(a=1)+\Pr(t|a=0)\cdot\Pr(a=0)} \\ &= \frac{1\cdot 0.001}{1\cdot 0.001+0.01\cdot 0.999} = 0.091\end{aligned}$$

slides adapted from courses.d2l.ai/berkeley-stat-157


SPAMMER
the other pink meat
UserFriendly.Org
IMPAIRING PRODUCTIVITY SINCE 1997

Naive Bayes

Naive Bayes Spam Filter

- **Key assumption**
  Words occur independently of each other
  given the label of the document

$$p(w_1, \ldots, w_n|\text{spam}) = \prod_{i=1}^{n} p(w_i|\text{spam})$$

- Spam classification via Bayes Rule

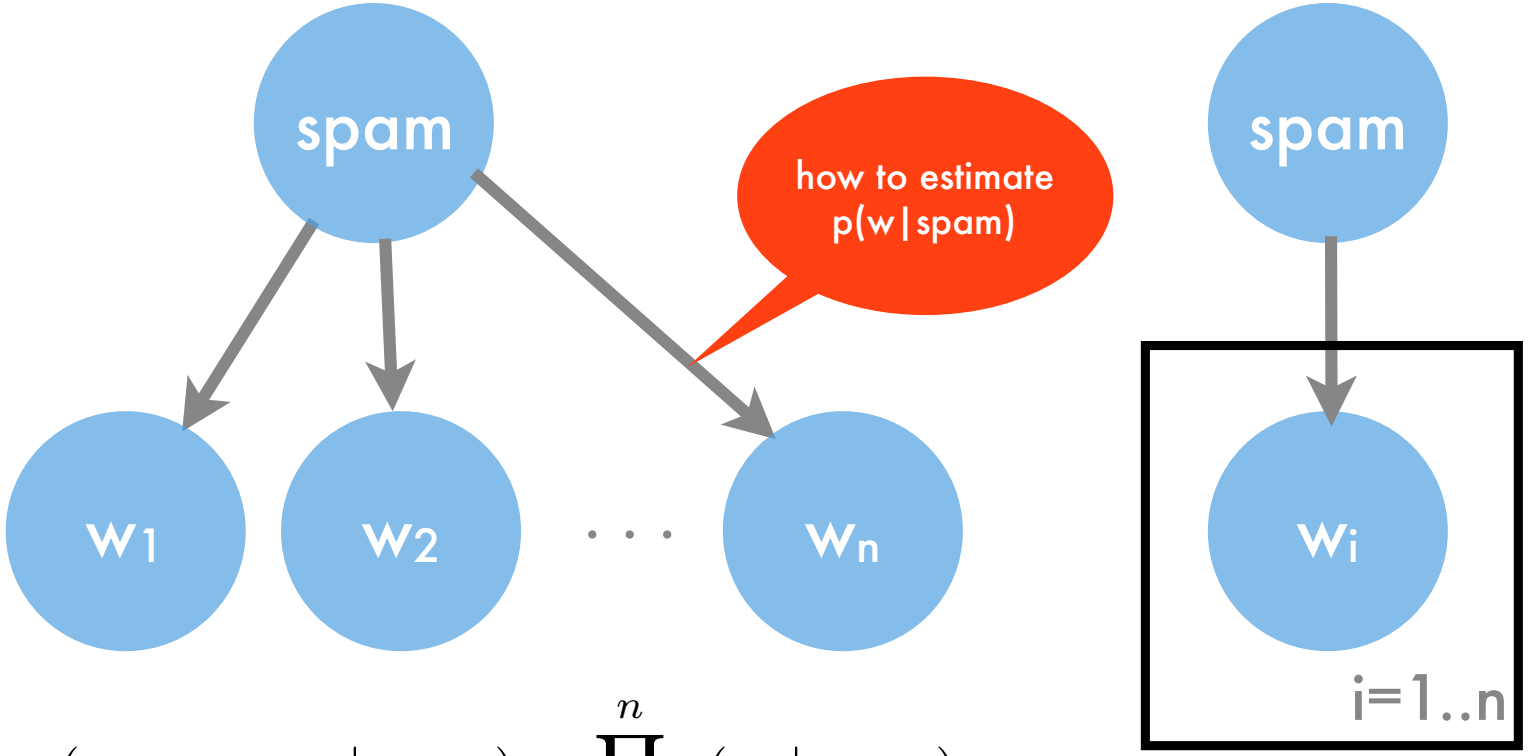
$$p(\text{spam}|w_1, \ldots, w_n) \propto p(\text{spam}) \prod_{i=1}^{n} p(w_i|\text{spam})$$

- **Parameter estimation**
  Compute spam probability and word distributions
  for spam and ham

slides adapted from courses.d2l.ai/berkeley-stat-157

A Graphical Model

A Graphical Model Captures the ASSUMED factorization.



$$p(w_1, \ldots, w_n|\text{spam}) = \prod_{i=1}^{n} p(w_i|\text{spam})$$

The assumed factorization - Joint probability (GIVEN THE PARENT) follows independence

Naive Bayes Spam Filter

- Data
  - Emails (headers, body, metadata)
  - Labels (spam/ham)
    assume that users actually label all mails
  - Images and labels
- Need to estimate $p(y)$, $p(x_i|y)$
  - Compute distribution of x_i for every y
  - Compute distribution of y

slides adapted from courses.d2l.ai/berkeley-stat-157

this is a gross simplification

- date
- time
- recipient path
- IP number
- sender
- encoding
- many more features

```
Delivered-To: alex.smola@gmail.com
Received: by 10.216.47.73 with SMTP id s51cs361171web;
        Tue, 3 Jan 2012 14:17:53 -0800 (PST)
Received: by 10.213.17.145 with SMTP id s17mr2519891eba.147.1325629071725;
        Tue, 03 Jan 2012 14:17:51 -0800 (PST)
Return-Path: <alex+caf_=alex.smola=gmail.com@smola.org>
Received: from mail-ey0-f175.google.com (mail-ey0-f175.google.com [209.85.215.175])
        by mx.google.com with ESMTPS id n4si29264232eef.57.2012.01.03.14.17.51
        (version=TLSv1/SSLv3 cipher=OTHER);
        Tue, 03 Jan 2012 14:17:51 -0800 (PST)
Received-SPF: neutral (google.com: 209.85.215.175 is neither permitted nor denied by best guess
record for domain of alex+caf_=alex.smola=gmail.com@smola.org) client-ip=209.85.215.175;
Authentication-Results: mx.google.com; spf=neutral (google.com: 209.85.215.175 is neither
permitted nor denied by best guess record for domain of
alex+caf_=alex.smola=gmail.com@smola.org) smtp.mail=alex+caf_=alex.smola=gmail.com@smola.org;
dkim=pass (test mode) header.i=@googlemail.com
Received: by eaal1 with SMTP id l1so15092746eaa.6
        for <alex.smola@gmail.com>; Tue, 03 Jan 2012 14:17:51 -0800 (PST)
Received: by 10.205.135.18 with SMTP id ie18mr5325064bkc.72.1325629071362;
        Tue, 03 Jan 2012 14:17:51 -0800 (PST)
X-Forwarded-To: alex.smola@gmail.com
X-Forwarded-For: alex@smola.org alex.smola@gmail.com
Delivered-To: alex@smola.org
Received: by 10.204.65.198 with SMTP id k6cs206093bki;
        Tue, 3 Jan 2012 14:17:50 -0800 (PST)
Received: by 10.52.88.179 with SMTP id bh19mr10729402vdb.38.1325629068795;
        Tue, 03 Jan 2012 14:17:48 -0800 (PST)
Return-Path: <althoff.tim@googlemail.com>
Received: from mail-vx0-f179.google.com (mail-vx0-f179.google.com [209.85.220.179])
        by mx.google.com with ESMTPS id dt4si11767074vdb.93.2012.01.03.14.17.48
        (version=TLSv1/SSLv3 cipher=OTHER);
        Tue, 03 Jan 2012 14:17:48 -0800 (PST)
Received-SPF: pass (google.com: domain of althoff.tim@googlemail.com designates 209.85.220.179
as permitted sender) client-ip=209.85.220.179;
Received: by vcbf13 with SMTP id f13so11295098vcb.10
        for <alex@smola.org>; Tue, 03 Jan 2012 14:17:48 -0800 (PST)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;
        d=googlemail.com; s=gamma;
        h=mime-version:sender:date:x-google-sender-auth:message-id:subject
         :from:to:content-type;
        bh=WCbdZ5sXac25dpH02XcRyDOdts993hKwsAVXpGrFh0w=;
        b=WK2B2+ExWnf/gvTkw6uUvKuP4XeoKnlJq3USYTm0RARK8dSFjyOQsIHeAP9Yssxp6O
         7ngGoTzYqd+ZsyJfvQcLAWp1PCJhG8AMcnqWkx0NMeoFvIp2HQooZwxSOCx5ZRgY+7qX
         uIbbdna4lUDXj6UFe16SpLDCkptd8OZ3gr7+o=
MIME-Version: 1.0
Received: by 10.220.108.81 with SMTP id e17mr24104004vcp.67.1325629067787;
 Tue, 03 Jan 2012 14:17:47 -0800 (PST)
Sender: althoff.tim@googlemail.com
Received: by 10.220.17.129 with HTTP; Tue, 3 Jan 2012 14:17:47 -0800 (PST)
Date: Tue, 3 Jan 2012 14:17:47 -0800
X-Google-Sender-Auth: 6bwi6D17HjZIkxOEol38NZzyeHs
Message-ID: <CAFJJHDGPBW+SdZg0MdAABiAKydDk9tpeMoDijYGjoGO-WC7osg@mail.gmail.com>
Subject: CS 281B. Advanced Topics in Learning and Decision Making
From: Tim Althoff <althoff@eecs.berkeley.edu>
To: alex@smola.org
Content-Type: multipart/alternative; boundary=f46d043c7af4b07e8d04b5a7113a

--f46d043c7af4b07e8d04b5a7113a
Content-Type: text/plain; charset=ISO-8859-1
```

Naive NaiveBayes Classifier

- Two classes (spam/ham)
- Binary features (e.g. presence of $$$, viagra)
- Simplistic Algorithm
  - Count occurrences of feature for spam/ham
  - Count number of spam/ham mails

feature probability

spam probability

$$p(x_i = \text{TRUE}|y) = \frac{n(i,y)}{n(y)} \text{ and } p(y) = \frac{n(y)}{n}$$

$$p(y|x) \propto \frac{n(y)}{n} \prod_{i:x_i=\text{TRUE}} \frac{n(i,y)}{n(y)} \prod_{i:x_i=\text{FALSE}} \frac{n(y)-n(i,y)}{n(y)}$$

slides adapted from courses.d2l.ai/berkeley-stat-157

Introduction to (Deep) Learning

slides adapted in part from courses.d2l.ai/berkeley-stat-157

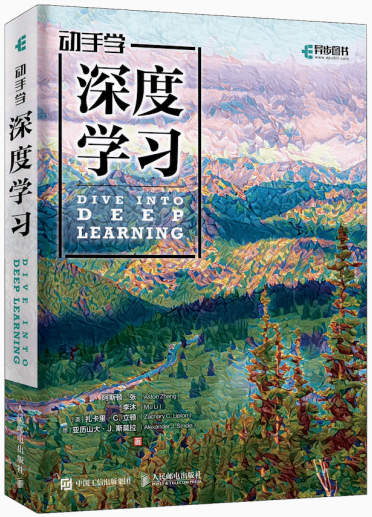
Goals

- **Introduction** to Deep Learning
  (basic MLP, optimization, convolutions, sequences)
- **Theory**
  - Capacity control (weight decay, dropout, batch norm)
  - Optimization, models, overfitting, objective functions
- **Practice**
  - Write code in Python / R (+tensorflow…)
  - Solve “realistic” problems

adapted in part from courses.d2l.ai/berkeley-stat-157

Further Reading/Resources…

- **Dive into Deep Learning**
  - Jupyter Notebooks
  - Github repository at **d2l-ai/d2l-en**



《动手学深度学习》

面向中文读者的能运行、可讨论的深度学习教科书

http://zh.gluon.ai

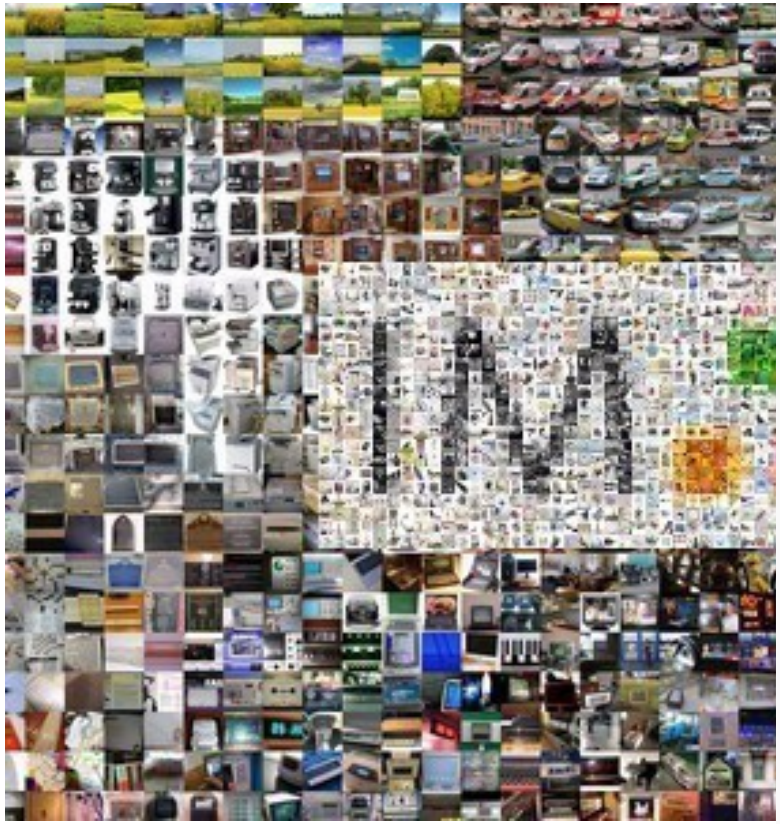
adapted in part from courses.d2l.ai/berkeley-stat-157

Classify Images

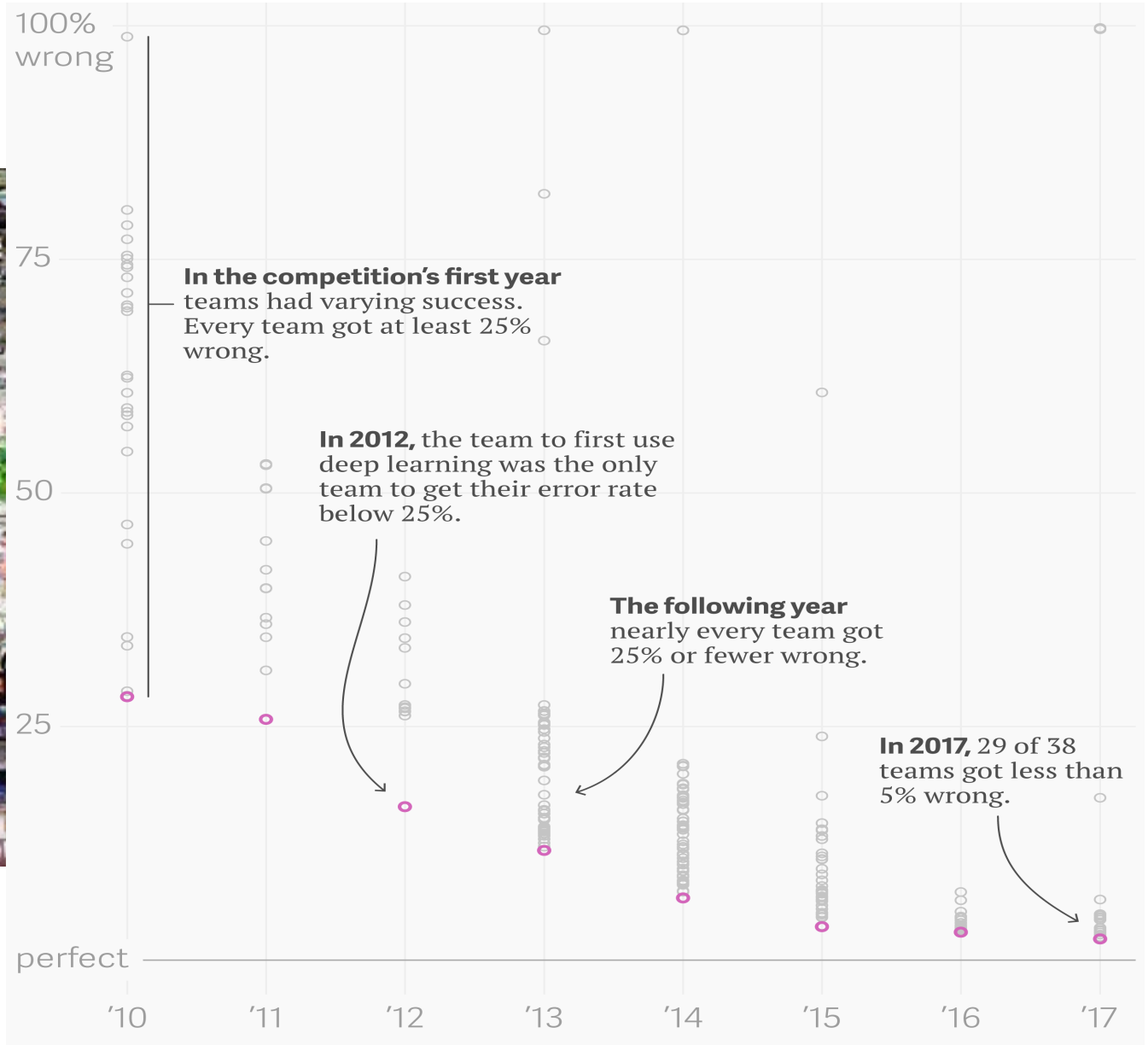
http://www.image-net.org/

slides adapted from courses.d2l.ai/berkeley-stat-157

Classify Images



Yanofsky, Quartz
https://qz.com/1034972/the-data-that-changed-the-direction-of-ai-research-and-possibly-the-world/



Detect and Segment Objects

https://github.com/matterport/Mask_RCNN

slides adapted from courses.d2l.ai/berkeley-stat-157

Style transfer

https://github.com/zhanghang1989/MXNet-Gluon-Style-Transfer/

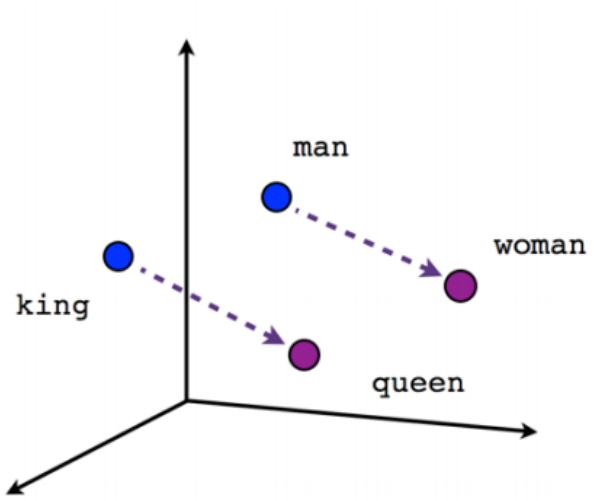
slides adapted from courses.d2l.ai/berkeley-stat-157

Synthesize Faces

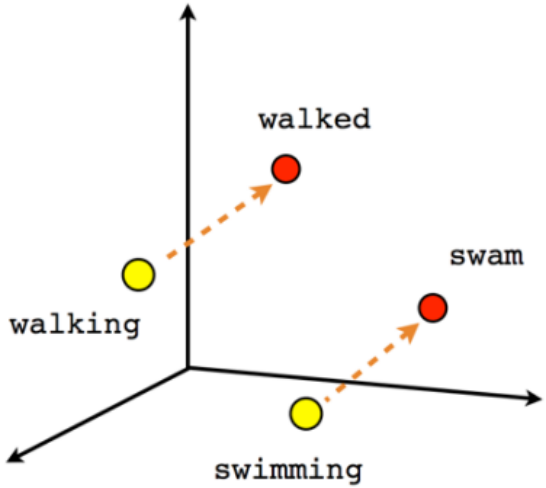
Karras et al, ICLR 2018

slides adapted from courses.d2l.ai/berkeley-stat-157

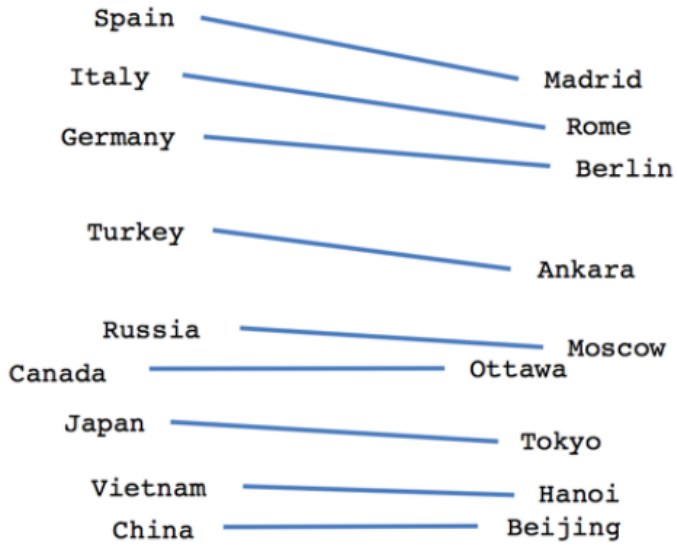
Analogies



Male-Female



Verb tense

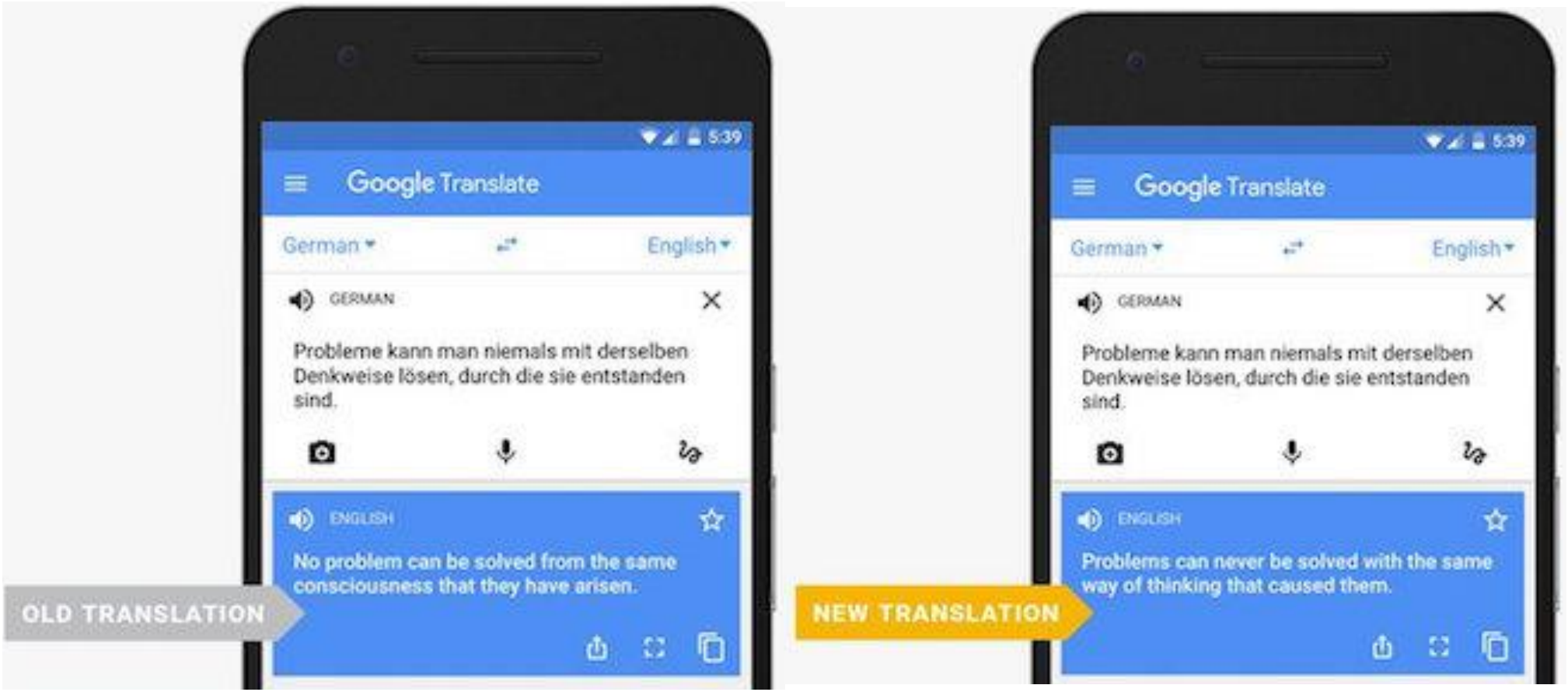


Country-Capital

https://www.tensorflow.org/tutorials/word2vec

slides adapted from courses.d2l.ai/berkeley-stat-157

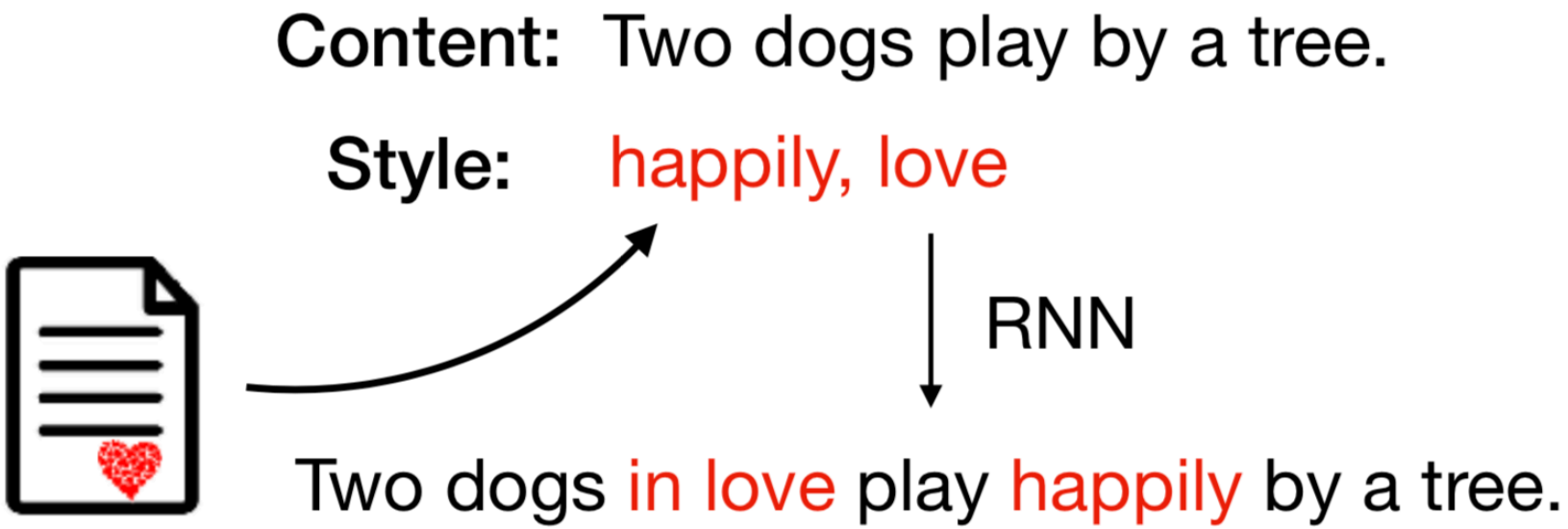
Machine Translation



https://www.pcmag.com/news/349610/google-expands-neural-networks-for-language-translation

slides adapted from courses.d2l.ai/berkeley-stat-157

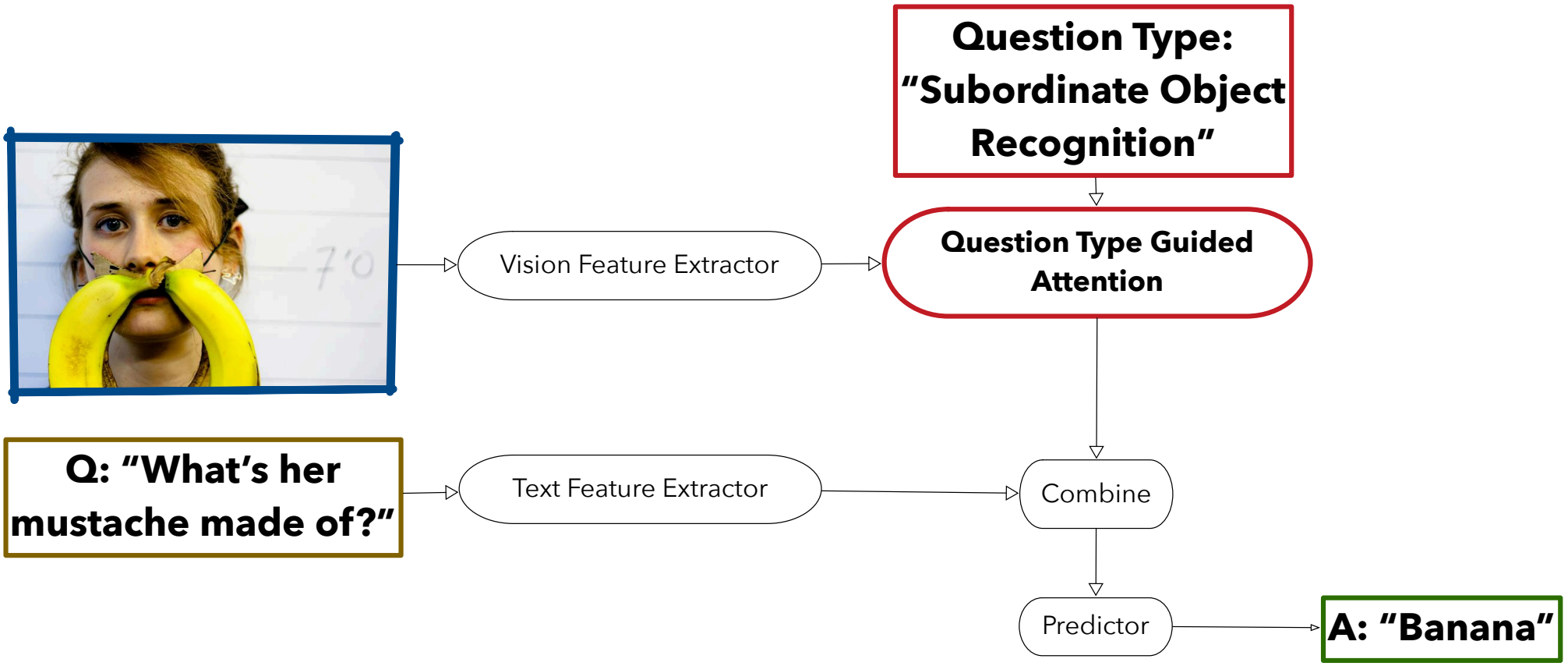
Text synthesis



Li et al, NACCL, 2018

slides adapted from courses.d2l.ai/berkeley-stat-157

Question answering



Shi et al, 2018, Arxiv

slides adapted from courses.d2l.ai/berkeley-stat-157

Image captioning

Human captions from the training set

Automatically captioned

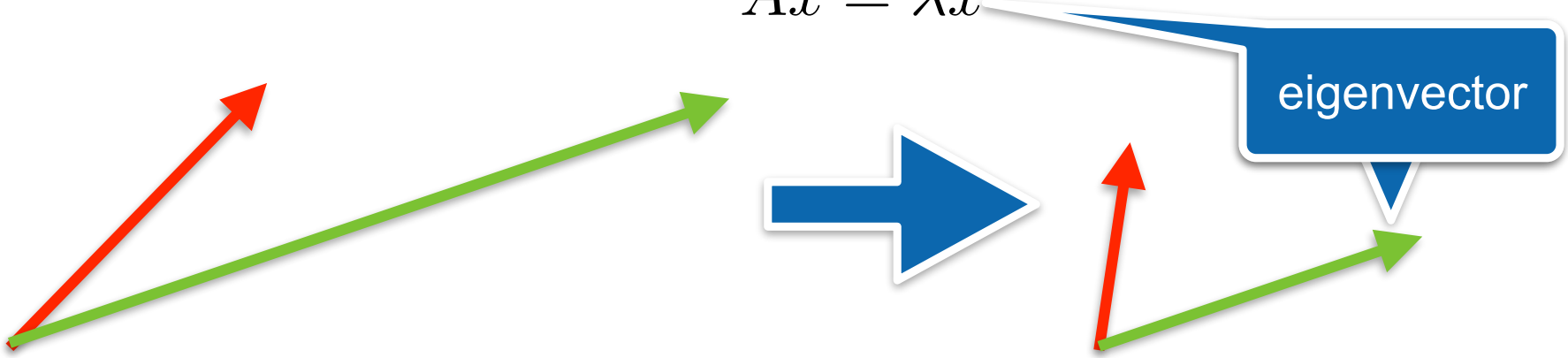
Shallue et al, 2016
https://ai.googleblog.com/2016/09/show-and-tell-image-captioning-open.html

slides adapted from courses.d2l.ai/berkeley-stat-157

Matrices

- **Eigenvectors and eigenvalue**
  - Vectors that aren't changed by the matrix

$$Ax = \lambda x$$



  - For symmetric matrices we can always find this

adapted in part from courses.d2l.ai/berkeley-stat-157

Special Matrices

- **Symmetric, antisymmetric** $A_{ij} = A_{ji}$ and $A_{ij} = -A_{ji}$



- **Positive definite**

$$\|x\|^2 = x^\top x \geq 0 \text{ generalizes to } x^\top A x \geq 0$$

(all positive eigenvalues)

adapted in part from courses.d2l.ai/berkeley-stat-157

Special Matrices

- **Orthogonal Matrices**
  - All rows of the matrix are orthogonal to each other
  - All rows of the matrix have unit length

$$U \text{ with } \sum_j U_{ij} U_{kj} = \delta_{ik}$$

  - Rewrite in matrix form

$$UU^\top = \mathbf{1}$$

Show that $U^\top U = \mathbf{1}$

- **Permutation Matrices**

$$P \text{ where } P_{ij} = 1 \text{ if and only if } j = \pi(i)$$

Show that P is orthogonal

adapted in part from courses.d2l.ai/berkeley-stat-157


GPUs love matrices and vectors
(they have many processors)


ndarray

N-dimensional Array Examples

- N-dimensional array, short for ndarray, is the main data structure for machine learning and neural networks

0-d (scalar)

```
1.0
```

A class label

1-d (vector)

```
[1.0, 2.7, 3.4]
```

A feature vector

2-d (matrix)

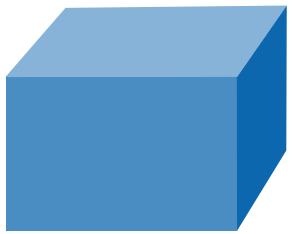
```
[[1.0, 2.7, 3.4]
 [5.0, 0.2, 4.6]
 [4.3, 8.5, 0.2]]
```

A example-by-feature matrix

adapted from courses.d2l.ai/berkeley-stat-157

ND Array Examples, cont

3-d



```
[[[0.1, 2.7, 3.4]
  [5.0, 0.2, 4.6]
  [4.3, 8.5, 0.2]]
 [[3.2, 5.7, 3.4]
  [5.4, 6.2, 3.2]
  [4.1, 3.5, 6.2]]]
```

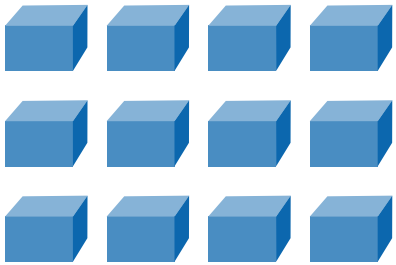
A RGB image (width x height x channels)

4-d

```
[[[[. . .
    . . .
    . . .]]]
```

A batch of RGB images (batch-size x width x height x channels)

5-d



```
[[[[. . .
    . . .
    . . .]]]
```

A batch of videos (batch-size x time x width x height x channels)

adapted from courses.d2l.ai/berkeley-stat-157

Pointers

- d2l.ai/chapter_crashcourse/ndarray.html
- beta.mxnet.io/guide/crash-course/1-ndarray.html
- Gilbert Strang's Linear Algebra course github.com/juanklopper/MIT_OCW_Linear_Algebra_18_06
- Berkeley HPC course (GPU sections) sites.google.com/lbl.gov/cs267-spr2018/

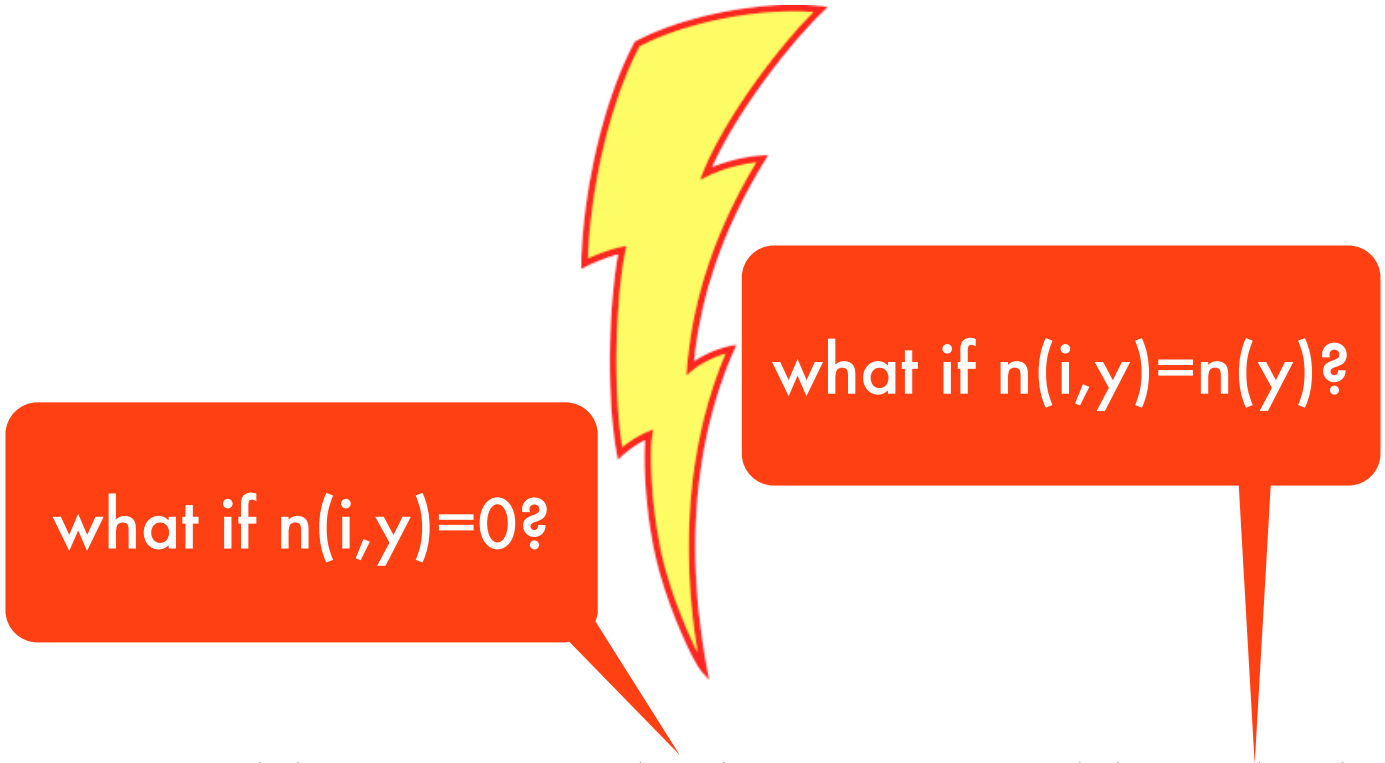
adapted from courses.d2l.ai/berkeley-stat-157

Summary

- **Deep Learning**
  Quick overview
- **Linear Algebra**
  Basic notation
- **NDArray**

adapted in part from courses.d2l.ai/berkeley-stat-157

Naive NaiveBayes Classifier



$$p(y|x) \propto \frac{n(y)}{n} \prod_{i:x_i=\text{TRUE}} \frac{n(i,y)}{n(y)} \prod_{i:x_i=\text{FALSE}} \frac{n(y)-n(i,y)}{n(y)}$$

slides adapted from courses.d2l.ai/berkeley-stat-157