CHIDATA

Announcements

Last exam: Dec 2nd to Dec 4th

New office hours: Tu 2pm-4pm

Projects: Should submit initial results ASAP.

Why Your Data Structures Class Still Matters

Why Machine Learning Doesn't Solve Everything

CHIDATA

CHIDATA

Recap: Model Fitting

Fitting: Observe xi, yi, **infer theta**

Regression Problems -> Continuous Y

Classification Problems -> Discrete Y

CHIDATA

Recap: ~~Model Fitting~~ Machine Learning

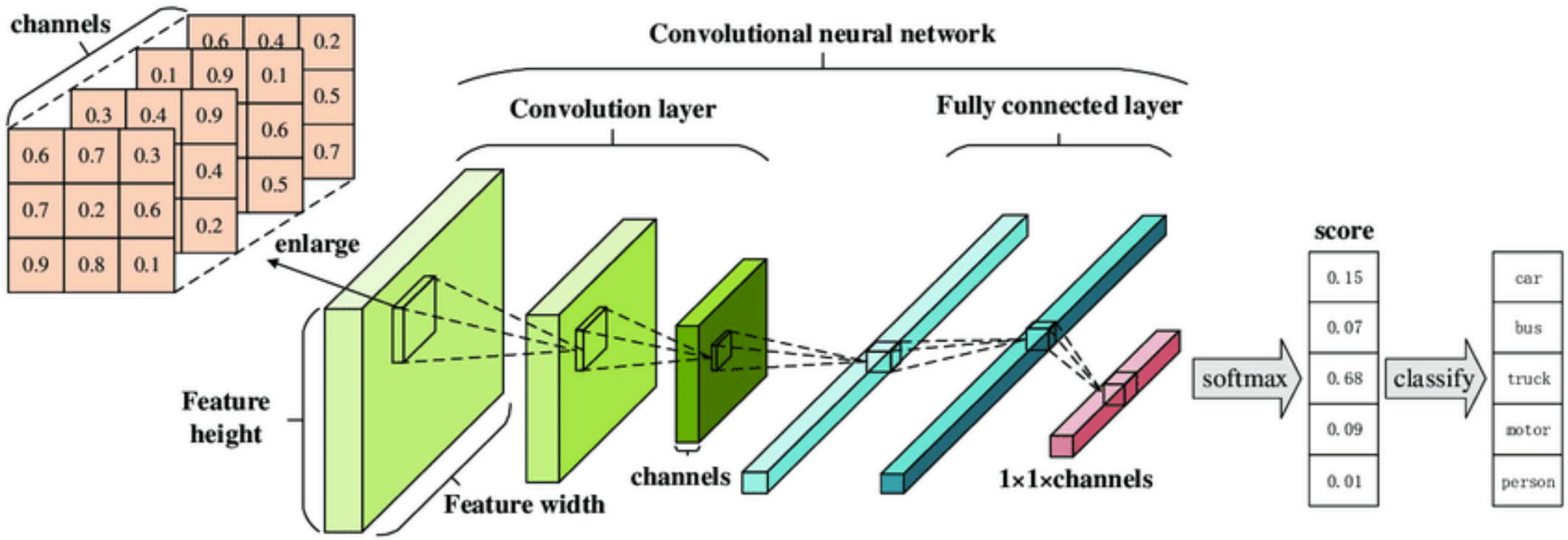
Fitting: Observe xi, yi, **infer theta**

Regression Problems -> Continuous Y

Classification Problems -> Discrete Y

Convolutional Neural Network

Cascade of filters that can be trained (fit) as one unit



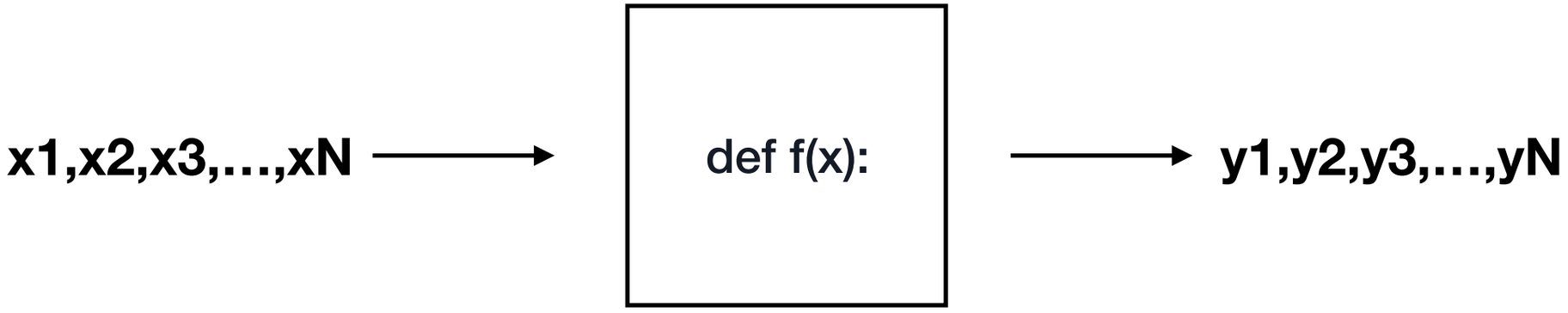
"Layers" that consist of convolutions, classifications, regressions

CHIDATA

What is the implication?

No more programming?

- Just show sufficient input and output examples?



CHIDATA

What is the implication?

No more databases?

- Just show sufficient queries and result examples?

CHIDATA

Project Idea

Train a natural language model that predicts the year in which a battle occurred based on a description of the the parties involved and location.

Collect years and associated words from wikipedia.

Train model that regresses a collection of words to years.

CHIDATA

Remember!

A library keeps yearly weather records from the year 1850 - present.

We want to calculate the average year-to-year temperature fluctuation between 1850 and 1900.

CHIDATA

A "Closed-World" Model

All relevant facts are in the universe and are assumed to be correct.

All students grades from an exam (compute median grade)

All transactions made at a bank (compute account balances)

All users in a web application (compute total users)

All English articles on wikipedia

CHIDATA

A “Retrieval” Problem

Library

1850 1851 ... 1899 1900 1901 ... 2019 2020

CHIDATA

A “Retrieval” Problem

All of the relevant data are contained in the library, you just have to “find it”.

CHIDATA

Remember! #2

A library keeps yearly weather records from the year 1850 - present, **but some of them are missing**

We want to calculate the average year-to-year temperature fluctuation between 1850 and 1900.

CHIDATA

No Choice About Incomplete Data

“Hypothetical” Library

“Actual” Library

CHIDATA

Pitfalls of Machine Learning (1)

Scenario 1. Database problem

Scenario 2. Machine learning

Machine learning is useful when you need to determine a property of **unseen data**.

Unseen data: same population as training data but haven't sampled it

CHIDATA

Pitfalls of Machine Learning (1)

Scenario 1. Database problem

Scenario 2. Machine learning

Machine learning is useful when you need to determine a property of **unseen data**.

Unseen data: cannot enumerate all possible inputs even if we tried.

CHIDATA

Pitfalls of Machine Learning (1)

Scenario 1. Database problem

- Guarantee of accuracy over data that you have seen before.

Scenario 2. Machine learning

- Weak guarantees. Theory doesn't always predict practice.

CHIDATA

Another Case Study

Operating systems often have magic constants for scheduling, resource allocation, and error recovery.

E.g., rate limit a program after it requests 15 concurrent threads.

Idea: replace all hand-tuned constants with ones learned from data.

Computer tries different constant configurations observes performance, builds model to predict best settings.

CHIDATA

Pitfalls of Machine Learning (2)

Hand-tuned

- Predictable behavior, consistent, and stateless.

Machine learning

- Depends on the input data, could vary deployment to deployment.

CHIDATA

Pitfalls of Machine Learning (2)

Hand-tuned

- Easy to debug

Machine learning

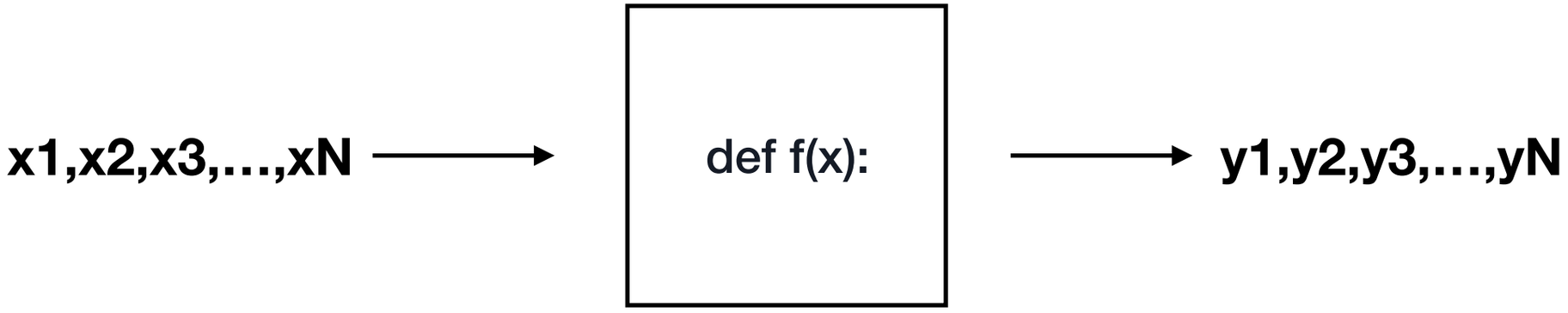
- Harder to debug

CHIDATA

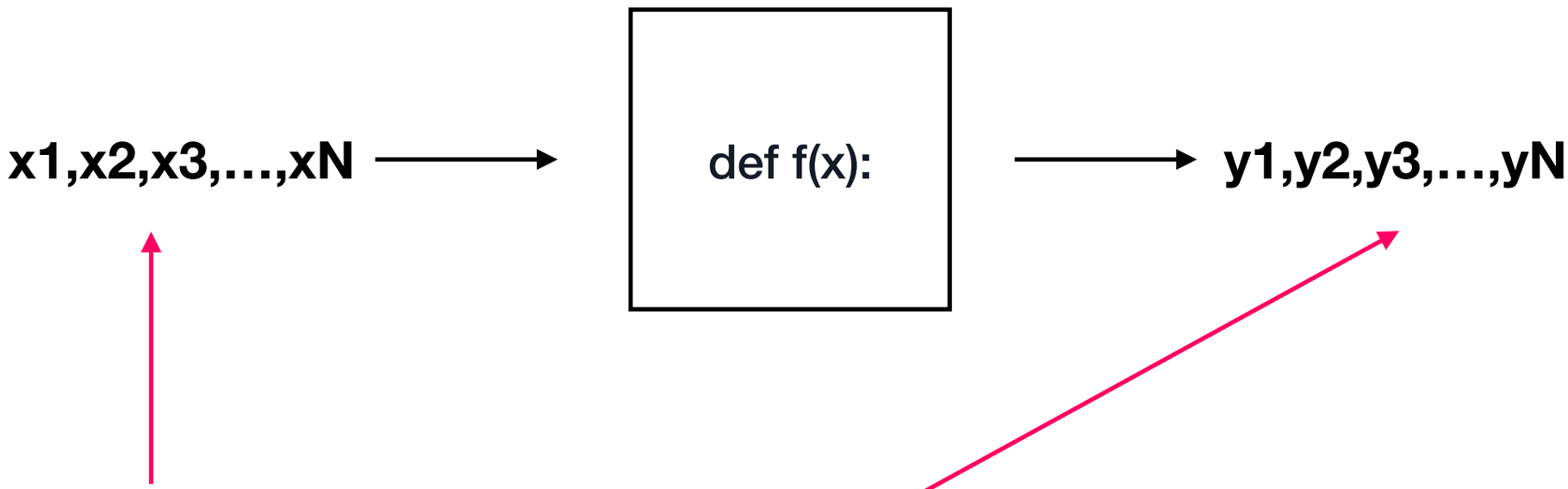
What is the implication?

No more programming?

- Just show sufficient input and output examples?



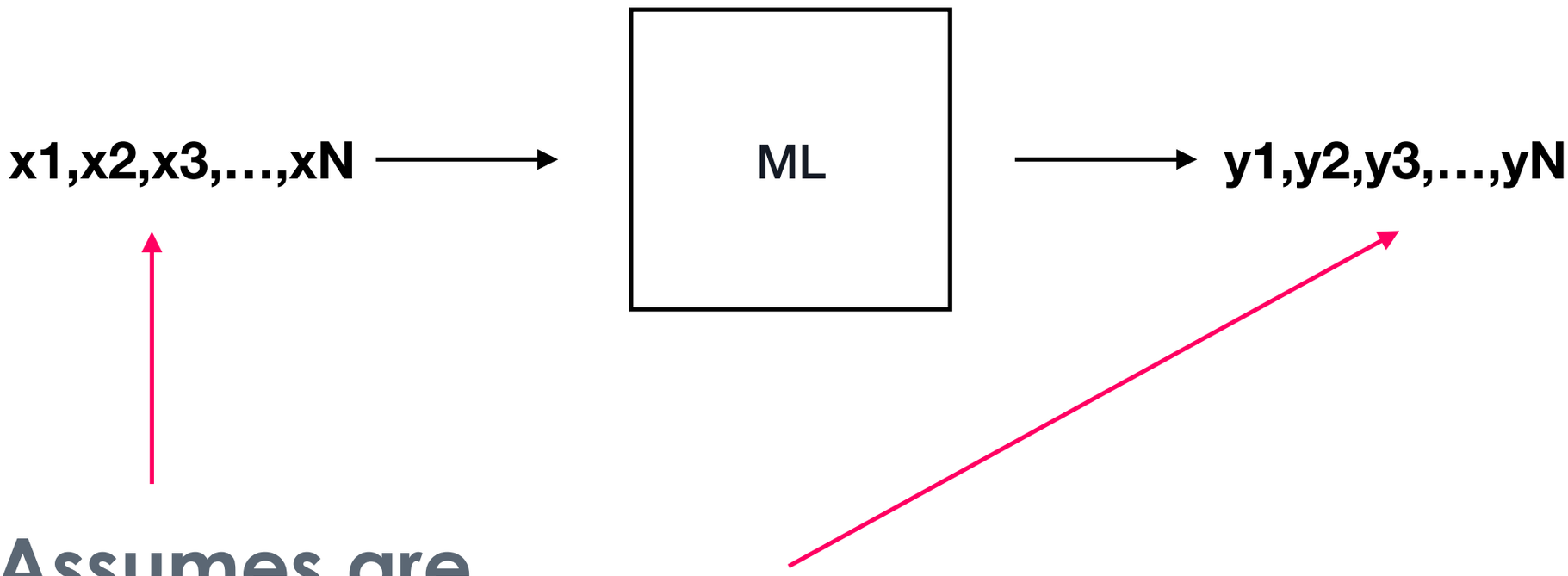
Problem?



Not all inputs/outputs are readily numerical

CHIDATA

Problem?



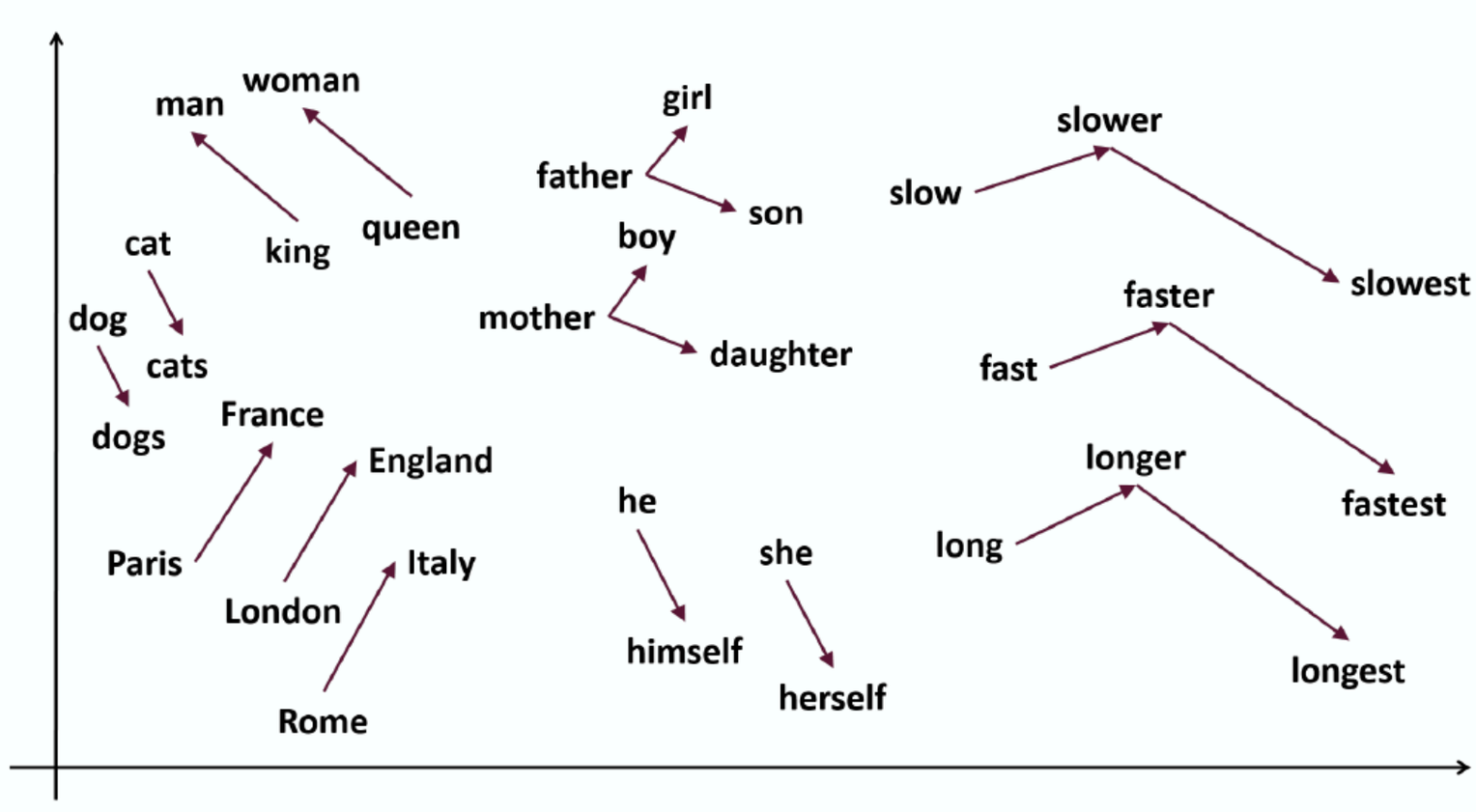
Assumes are fixed-length vectors

Assumes Euclidean Space

Natural Language

Circumvent problem by embedding

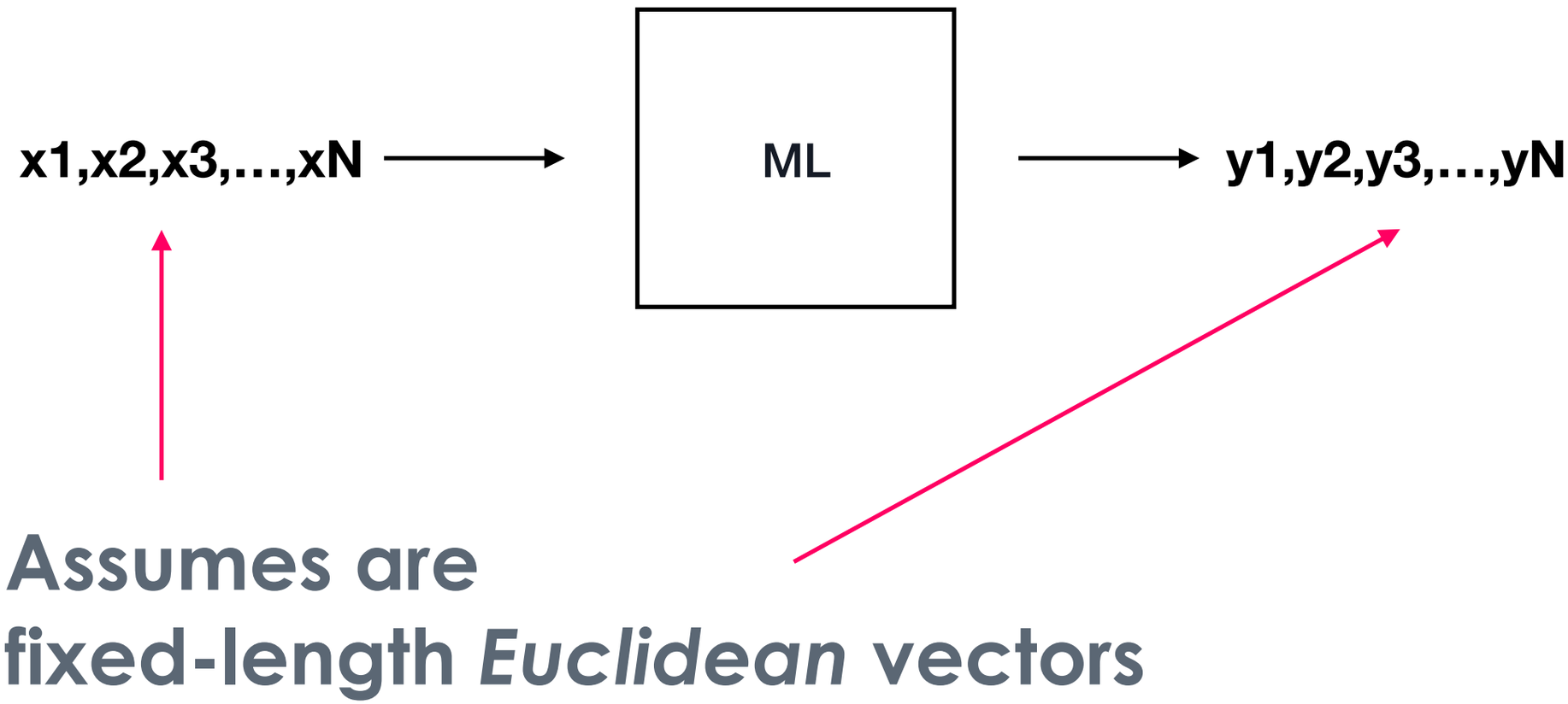
Use large corpora of text (News, Books, etc.) to find associations between words.



CHIDATA

Problem?

Embedding doesn't always work!



CHIDATA

Problem?

Highly structured spaces…

```
10 import math
11
12 def f(x):
13     y=math.sin(math.pi*x)
14     return y
15
16 a=0.0
17 b=1.0
18 n=4
19
20 h=(b-a)/n
21 I=f(a)
22
23 for k in range (1,n):
24     I=I+2*f(a+k*h)
25
26 I=I+f(b)
27 I=h*I/2
28 print(n,I)
```

Interpreting differences in code with ML is significantly harder than natural language.

CHIDATA

Pitfalls of Machine Learning (3)

Compilers

- Use deterministic parsers that understand nested, composed, and linked structures.

Machine learning

- Relies on proximity in space as a proxy for related meaning.

We Are All Familiar With This Issue

Highly structured low-dimensional spaces…

CHIDATA

The Narrative….

Machine Learning Libraries

Spark MLlib

KeystoneML

TensorFlow™

ML

Caffe

torch
Facebook AI Research

WEKA
The University of Waikato

scikits learn
machine learning in Python

CHIDATA

The Narrative….

From Their Websites…

"You can instantiate your DNNClassifier with just a couple lines of code"

"Its goal is to make practical machine learning scalable and easy."

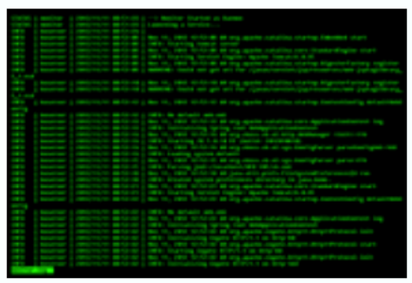
"This high-level language significantly increases the productivity of data scientists"

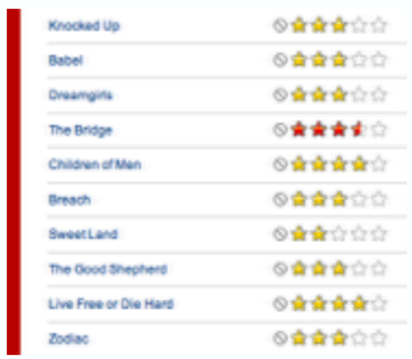
"Accessible to everybody, and reusable in various contexts"

CHIDATA

What we show you in class

Data Sources





"a couple lines of code"

ML Library

Spark
MLlib

KeystoneML

TensorFlow™

ML

$$\theta^* = \arg\min_{\theta} \sum_{i=1}^{N} \phi(x_i, y_i, \theta)$$

Predictive Actions

CHIDATA

Industrial Reality

Machine Learning: The High-Interest Credit Card of Technical Debt

D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov,
Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young
{dsculley,gholt,dgg,edavydov}@google.com
{toddphillips,ebner,vchaudhary,mwyoung}@google.com
Google, Inc

Abstract

Machine learning offers a fantastically powerful toolkit for building complex systems quickly. This paper argues that it is dangerous to think of these quick wins as coming for free. Using the framework of *technical debt*, we note that it is remarkably easy to incur massive ongoing maintenance costs at the system level when applying machine learning. The goal of this paper is highlight several machine learning specific risk factors and design patterns to be avoided or refactored where possible. These include boundary erosion, entanglement, hidden feedback loops, undeclared consumers, data dependencies, changes in the external world, and a variety of system-level anti-patterns.

1 Machine Learning and Complex Systems

Real world software engineers are often faced with the challenge of moving quickly to ship new products or services, which can lead to a dilemma between speed of execution and quality of engineering. The concept of *technical debt* was first introduced by Ward Cunningham in 1992 as a way to help quantify the cost of such decisions. Like incurring fiscal debt, there are often sound strategic reasons to take on technical debt. Not all debt is necessarily bad, but technical debt does tend to compound. Deferring the work to pay it off results in increasing costs, system brittleness

CHIDATA

Industrial Reality

Machine Learning:
The High-Interest Credit Card of Technical Debt

Currently convenient design decisions that you pay for later in terms of maintenance and upkeep.

CHIDATA

Industrial Reality

Machine Learning:
The High-Interest Credit Card of Technical Debt

- Get apparent benefits very quickly, impressive results
- Easy to under-estimate how hard it is to debug, maintain, and deploy such software.
- Often requires “glue code” that is unreliable

CHIDATA

Should I use Machine Learning?

Are current solutions to the problem sufficiently capable?

Is the cost of collecting labeled data is cheaper than programming a general solution?

Is the cost of “maintaining” the model worth the generality?

CHIDATA

Should I use Machine Learning?

Are current solutions to the problem sufficiently capable?

Does machine learning improve on accuracy, generality, or both.

Bad ideas:

- Replace all function calls in a program with learned models. **(Google tried this…)**

- Using machine learning where the decision is simple/obvious from other data sources: cameras for speeding tickets/room occupancy. **(NVIDIA tried this…)**

CHIDATA

Should I use Machine Learning?

Is the cost of collecting labeled data is cheaper than programming a general solution?

How hard is it to collect labeled data?

Bad ideas:

- Using machine learning to determine experimental treatments for rare cancers. **(IBM tried this)**

CHIDATA

Should I use Machine Learning?

Is the cost of maintaining the model worth the generality?

Bad ideas:

- Continuous learning for conversation **(Microsoft tried this…)**

- Machine learning to determine parole cases **(Florida tried this…)**

CHIDATA

Should I use Machine Learning?

Are current solutions to the problem sufficiently capable?

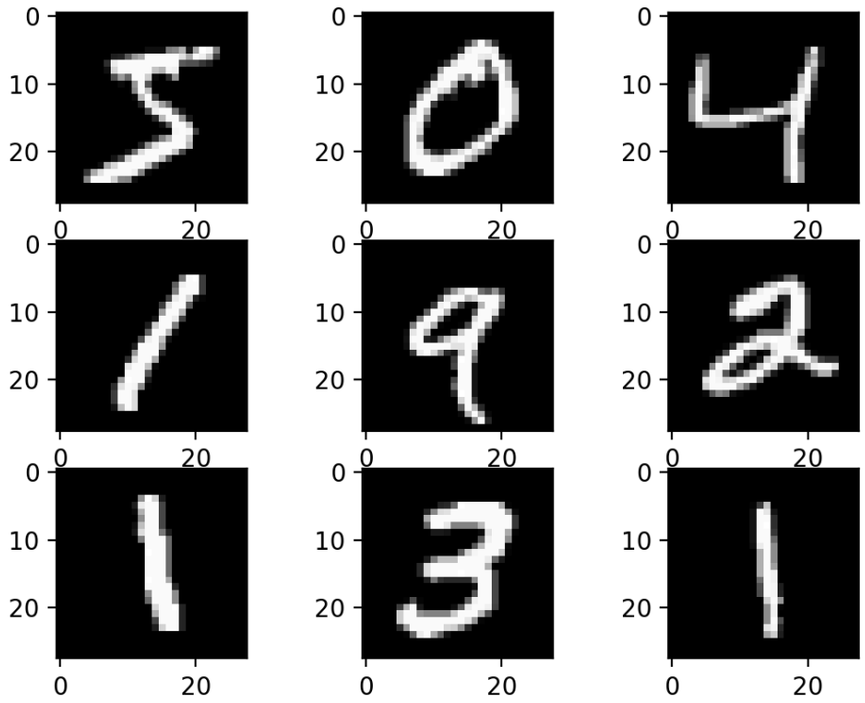
Is the cost of collecting labeled data is cheaper than programming a general solution?

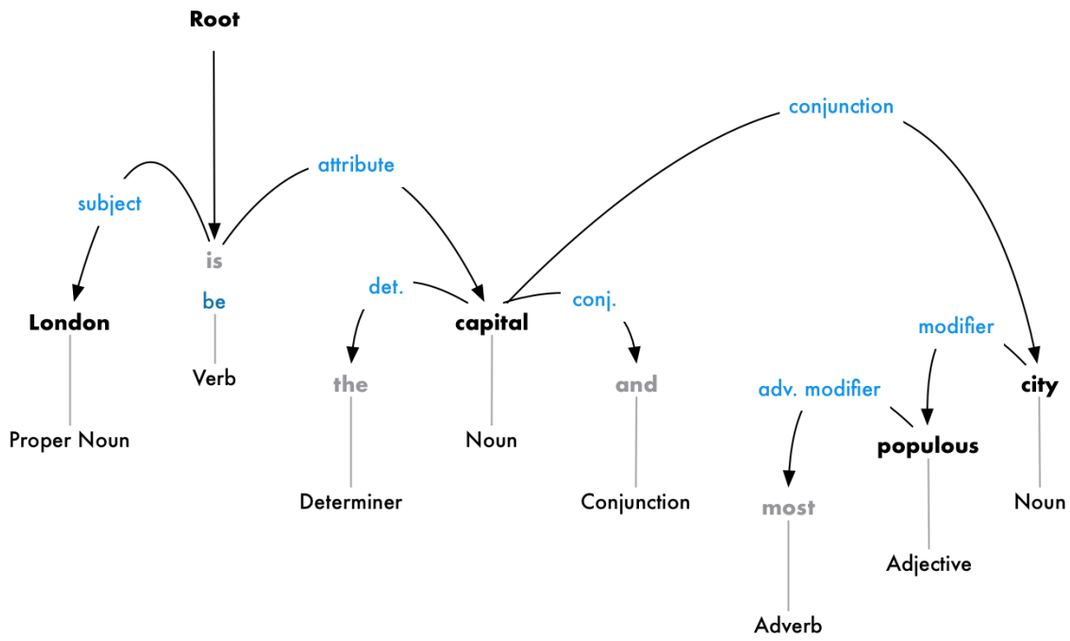
Is the cost of “maintaining” the model worth the generality?

CHIDATA

Sweet Spot

Computer Vision and Natural Language





Hard to enumerate all possible inputs, no general solutions.

Anyone can label data

Labels don’t really change

CHIDATA

Should I use Machine Learning?

Are current solutions to the problem sufficiently capable?

Is the cost of collecting labeled data is cheaper than programming a general solution?

Is the cost of “maintaining” the model worth the generality?