Slides for Pre-reading

Slides with yellow borders will not be covered in class, but is still testable content - you should review this before class.

Moore’s Law

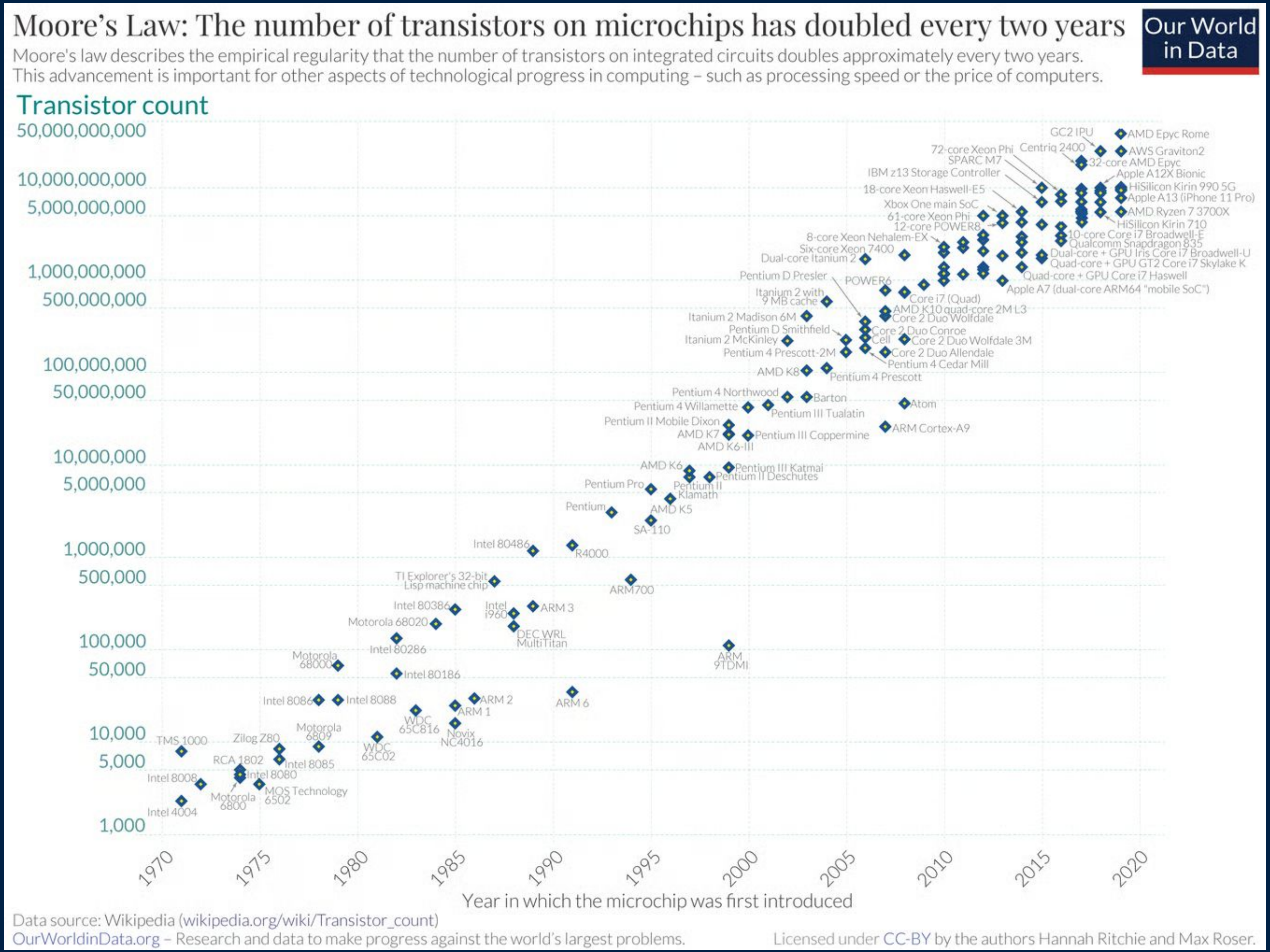
Moore's Law

Gordon Moore one of the co-founders of **Intel** corporation established the term "Moore's Law" in 1965, This law explains how the number of transistors on integrated circuits is increasing exponentially, which boosts computing capability and lowers prices. Moore's law has had a significant and far-reaching impact on technology, from the introduction of personal computers and smartphones to the advancement of artificial intelligence and Internet of Things(IoT) devices. it has accelerated development in sectors including telecommunications, healthcare, transportation, enabling organizations and people to accomplish things that were previously unthinkable.

Moore's Law Definition

The exponential increase in the number of transistors on integrated circuits over time is referred to as Moore's law. According to this, a chip transistor count tends to double every two years or so, resulting in higher processing power and better performance.

Source


Moore's Law: The number of transistors on microchips has doubled every two years
Our World in Data
Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years.
This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.
Transistor count
50,000,000,000
10,000,000,000
5,000,000,000
1,000,000,000
500,000,000
100,000,000
50,000,000
10,000,000
5,000,000
1,000,000
500,000
100,000
50,000
10,000
5,000
1,000
1970
1975
1980
1985
1990
1995
2000
2005
2010
2015
2020
Year in which the microchip was first introduced
GC2 IPU
AMD Epyc Rome
72-core Xeon Phi
Centriq 2400
AWS Graviton2
SPARC M7
32-core AMD Epyc
IBM z13 Storage Controller
Apple A12X Bionic
HiSilicon Kirin 990 5G
18-core Xeon Haswell-E5
Apple A13 (iPhone 11 Pro)
Xbox One main SoC
AMD Ryzen 7 3700X
61-core Xeon Phi
HiSilicon Kirin 710
12-core POWER8
10-core Core i7 Broadwell-E
8-core Xeon Nehalem-EX
Qualcomm Snapdragon 835
Six-core Xeon 7400
Dual-core + GPU Iris Core i7 Broadwell-U
Dual-core Itanium 2
Quad-core + GPU GT2 Core i7 Skylake K
Pentium D Presler
POWER6
Quad-core + GPU Core i7 Haswell
Itanium 2 with 9 MB cache
Apple A7 (dual-core ARM64 "mobile SoC")
Core i7 (Quad)
AMD K10 quad-core 2M L3
Itanium 2 Madison 6M
Core 2 Duo Wolfdale
Pentium D Smithfield
Core 2 Duo Conroe
Itanium 2 McKinley
Cell
Core 2 Duo Wolfdale 3M
Pentium 4 Prescott-2M
Core 2 Duo Allendale
Pentium 4 Cedar Mill
AMD K8
Pentium 4 Prescott
Pentium 4 Northwood
Barton
Pentium 4 Willamette
Pentium III Tualatin
Atom
Pentium II Mobile Dixon
AMD K7
Pentium III Coppermine
ARM Cortex-A9
AMD K6-III
AMD K6
Pentium III Katmai
Pentium II Deschutes
Pentium Pro
Pentium II Klamath
Pentium
AMD K5
SA-110
Intel 80486
R4000
TI Explorer's 32-bit Lisp machine chip
ARM700
Intel 80386
Intel i960
ARM 3
Motorola 68020
DEC WRL MultiTitan
Intel 80286
ARM 9TDMI
Motorola 68000
Intel 80186
Intel 8086
Intel 8088
ARM 2
ARM 1
ARM 6
WDC 65C816
Novix NC4016
Motorola 6809
WDC 65C02
TMS 1000
Zilog Z80
RCA 1802
Intel 8085
Intel 8080
Intel 8008
Motorola 6800
MOS Technology 6502
Intel 4004
Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)
OurWorldinData.org – Research and data to make progress against the world's largest problems.
Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.

Classification

Classification

The idea behind classification is that we want to use **patterns** and/or **correlations** to make decisions

Classification happens all the time in real life

- The doctor uses your symptoms and other measurements like weight/blood pressure/etc. to help make a diagnosis
- Google uses classification to determine what an image is

Classification is a **general class** of algorithms.

Classifiers

Classifiers are **algorithms** that **perform classification**

They are specific

- e.g. - we don't give loans to anyone an income of less than $50,000 per year

The algorithm you come up with is no different than the other algorithms you've come up with so far

You still need to state the steps you need to take to come up with the solution

Classifiers

Classifier

- A **classifier** is an **algorithm** that maps the input data to a specific category
  - Classifiers are derived from patterns or correlations from data.

Classifier: Training vs Test Data

- The data that classifiers learn the patterns has the "answer"
  - This data is called **training data**.

- Some of the training data is held back to check and see if the classifier works.
  - This is called **test data.**

Classifiers + Data

Classifiers then apply these patterns to new data where we don't know the answer

- **Input:** Digital image
- **Output:** Cat/not a cat
- **Training data:**
  Labeled images of cats and images that are not cats

CPSC 100

Computational Thinking

Algorithm, Classifiers and Trees!

Instructor: Firas Moosvi
Department of Computer Science
University of British Columbia

Course Admin

Agenda

- Course Admin
- Clicker Questions
- Classification
- Real world examples

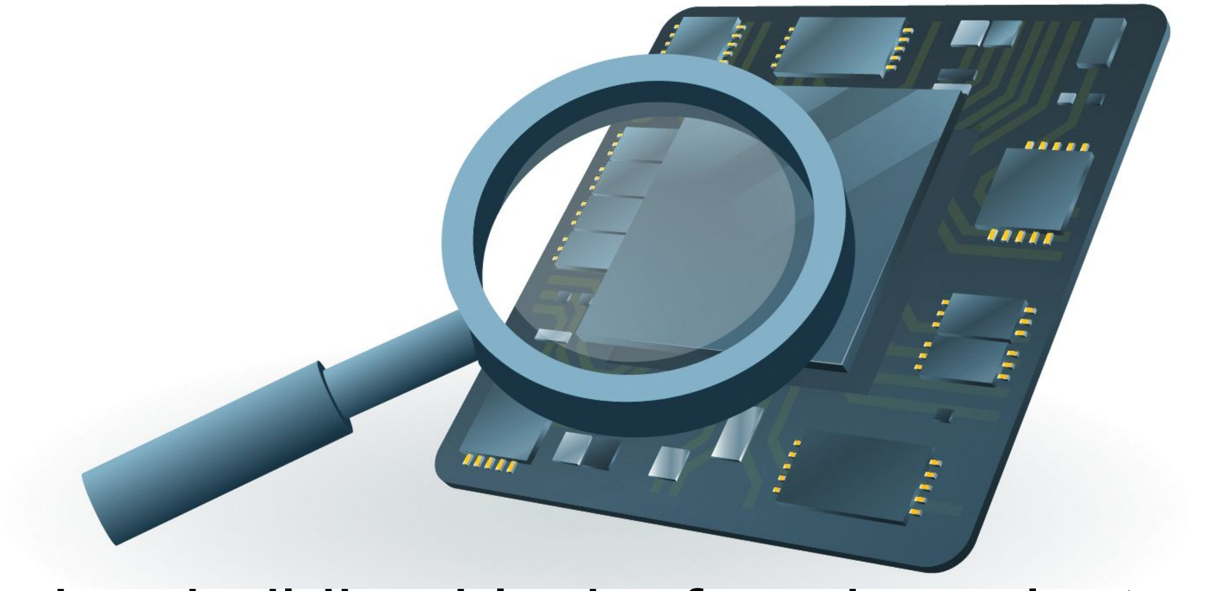
Learning Goals

Learning Goals

After this lecture, you should be able to:

- Describe the **classification** steps.
- Explain the concept of a **rooted tree** and **decision tree**.
- Describe what the **general decisions** are in building a decision tree.

Transistors



A transistor is a building block of modern electronics. Tens of billions of transistors are jammed into a “chip” so that data can be stored and processed.

Moore’s Law

Recall

Gordon Moore one of the co-founders of **Intel** corporation established the term "Moore's Law" in 1965, This law explains how the number of transistors on integrated circuits is increasing exponentially, which boosts computing capability and lowers prices. Moore's law has had a significant and far-reaching impact on technology, from the introduction of personal computers and smartphones to the advancement of artificial intelligence and Internet of Things(IoT) devices. it has accelerated development in sectors including telecommunications, healthcare, transportation, enabling organizations and people to accomplish things that were previously unthinkable.

Moore's Law Definition

The exponential increase in the number of transistors on integrated circuits over time is referred to as Moore's law. According to this, a chip transistor count tends to double every two years or so, resulting in higher processing power and better performance.

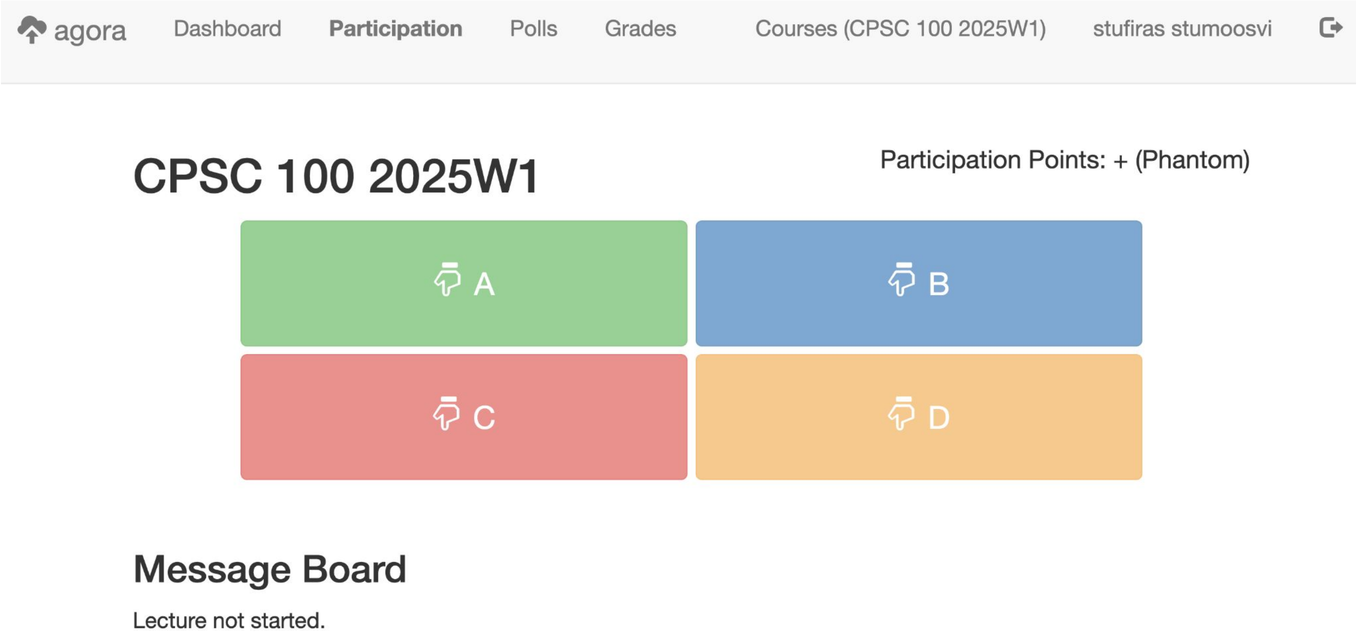
Source

Is Moore's Law still valid?

Agora: Discussions and Polling

Page: agora.students.cs.ubc.ca (login with your CWL)

Enroll code: **psychohistory**



Participation Question

Is Moore's Law still valid in 2025?

Yes; explain	No; explain
-	-

The Transistor Cliff

Sarah Constantin

Moore's law may be coming to an end. What happens to AI progress if it does?

How Hardware Affects AI Progress

The biggest AI models are trained on expensive, state-of-the-art microchips, or semiconductors. Only a few organizations, such as Google and OpenAI, have the budgets to train them. For years, improvements in AI performance have been driven by progress in this underlying hardware.

For most of the history of semiconductor manufacturing, steadily and predictably accelerating improvements in performance and reductions in price have been the norm. This pattern has been codified as "Moore's Law," Intel CEO Gordon Moore's observation that the number of transistors that could be placed on a chip for the same price doubled approximately every two years. That may be coming to an end. Depending on the specific semiconductor performance metric, Moore's Law has either stalled out already, or is on course to soon hit fundamental physical limits.

So, what could happen "after Moore's Law?" And how would that affect AI performance?

Let's zoom in and look at the details.

Source

Clicker Questions (using Agora)

Clicker Question

Q: What is an algorithm?

A. A series of steps written to solve a specific problem with specific inputs

B. A series of steps written to solve a specific problem with non-specific inputs

C. A series of steps written to solve a non-specific problem with specific inputs

D. A series of steps written to solve a non-specific problem with non-specific inputs

Classification Task: Loans

Recall

Algorithms

An ***algorithm*** describes a sequence of steps that is:

1. Unambiguous
   - No “assumptions” are required to execute the algorithm
   - The algorithm uses precise instructions
2. Executable
   - The algorithm can be carried out in practice
3. Terminating
   - The algorithm will eventually come to an end, or halt

Classification

Recall

The idea behind classification is that we want to use **patterns** and/or **correlations** to make decisions

Classification happens all the time in real life

- The doctor uses your symptoms and other measurements like weight/blood pressure/etc. to help make a diagnosis
- Google uses classification to determine what an image is

Classification is a **general class** of algorithms.

Classification Task - Cancer Treatment

Input: Genome sequence from cancerous biopsy tissue

- Age, gender, race, weight, family history, overall health, etc.

Output: Which cancer treatment is likely to work best

Training data:

Labeled genome sequences and which treatments were successful from both cancerous tissue

Classification Task - Loan Applications

Input: Individual's loan application

- Address, age, gender, credit rating, etc.

Output: Accept or reject loan application.

Training data:

List of loan apps, decisions made, and for those who were approved, whether they repaid the loan or not

Building a Classifier: Loans

Class Activity (on PrairieLearn)

Building a Classifier: Loans

Here is your past (training) data on some loans

Applicant	Income	Gave loan?
#1	$50,005	Yes
#2	$25,004	No
#3	$75,005	Yes
#4	$95,005	Yes
#5	$45,007	No

***Task**: Create an algorithm to decide what your classifier does: i.e., when will you give a loan, and when will you not give a loan?*

Participation Question

Q: Would your classifier give a loan if an applicant's income was $78,000?

Yes; explain

No; explain

-

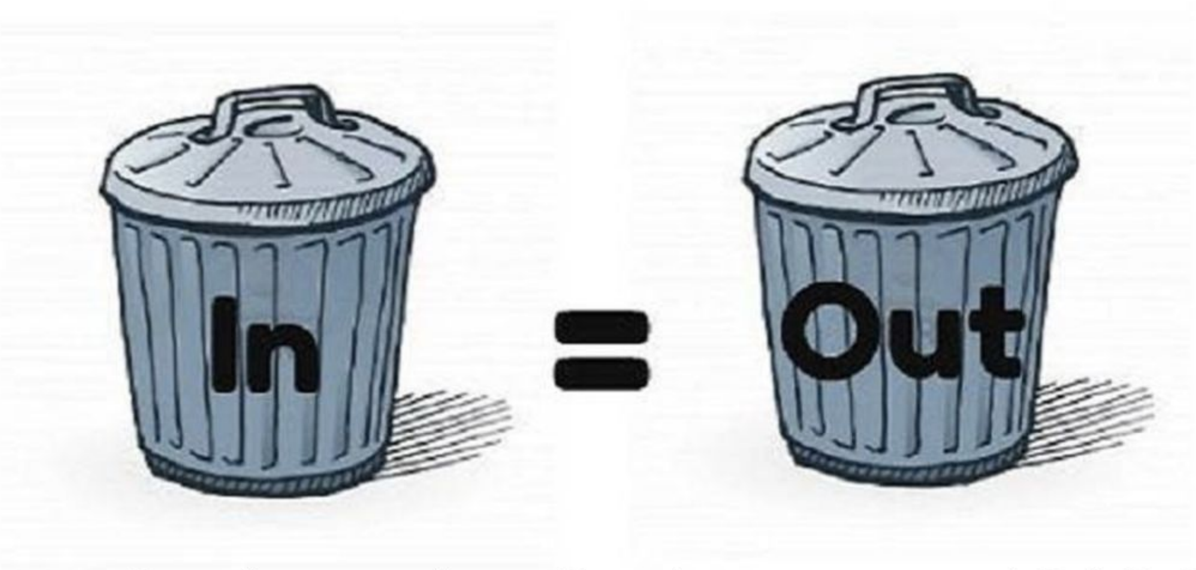
-

Applicant	Income	Gave loan?
#1	$50,005	Yes
#2	$25,004	No
#3	$75,005	Yes
#4	$95,005	Yes
#5	$45,007	No

36

Real World Examples

Training data influences the classifier



Garbage in, Garbage out (GIGO)

Reuters

World Business Markets Sustainability More

Insight - Amazon scraps secret AI recruiting tool that showed bias against women

By **Jeffrey Dastin**

October 10, 2018 5:50 PM PDT · Updated 6 years ago

Aa

SAN FRANCISCO (Reuters) - Amazon.com Inc's machine-learning specialists uncovered a big problem: their new recruiting engine did not like women.

The team had been building computer programs since 2014 to review job applicants' resumes with the aim of mechanizing the search for top talent, five people familiar with the effort told Reuters.

Source

How We Analyzed the COMPAS Recidivism Algorithm

by Jeff Larson, Surya Mattu, Lauren Kirchner and Julia Angwin

May 23, 2016

← Read the story

Across the nation, judges, probation and parole officers are increasingly using algorithms to assess a criminal defendant's likelihood of becoming a recidivist – a term used to describe criminals who re-offend. There are dozens of these risk assessment algorithms in use. Many states have built their own assessments, and several academics have written tools. There are also two leading nationwide tools offered by commercial vendors.

We set out to assess one of the commercial tools made by Northpointe, Inc. to discover the underlying accuracy of their recidivism algorithm and to test whether the algorithm was biased against certain groups.

Our analysis of Northpointe's tool, called COMPAS (which stands for Correctional Offender Management Profiling for Alternative Sanctions), found that black defendants were far more likely than white defendants to be incorrectly judged to be at a higher risk of recidivism, while white defendants were more likely than black defendants to be incorrectly flagged as low risk.

Source

IEEE.ORG IEEE XPLORE DIGITAL LIBRARY IEEE STANDARDS MORE SITES

IEEE Spectrum FOR THE TECHNOLOGY INSIDER

Type to search

ARTICLE | AI

In 2016, Microsoft's Racist Chatbot Revealed the Dangers of Online Conversation › The bot learned language from people on Twitter—but it also learned values

BY OSCAR SCHWARTZ | PUBLISHED 25 NOV 2019 | UPDATED 04 JAN 2024 | 5 MIN READ

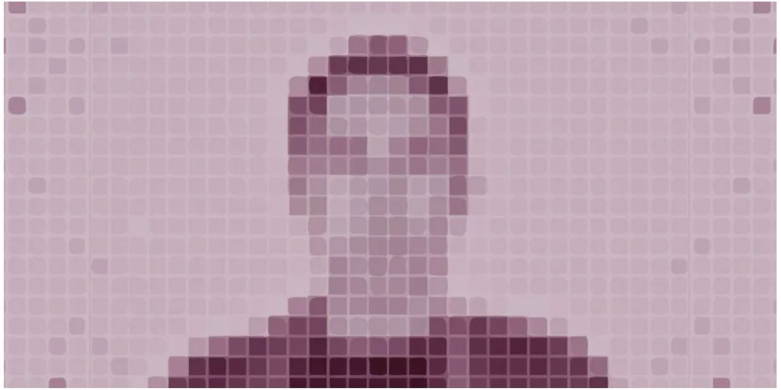
Source

Harvard Griffin GSAS Science Policy Group

Connecting policy & science in the graduate student community

Racial Discrimination in Face Recognition Technology

October 24, 2020



By Alex Najibi

Source

BBC Sign in | Home | News | Sport | Business | Innovation | Culture

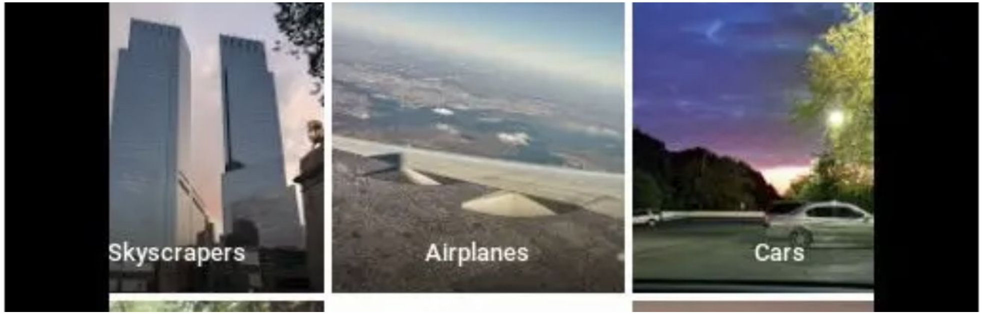
NEWS

Home | Israel-Gaza war | War in Ukraine | Climate | Video | World | US & Canada | UK | Business | Tech

Tech

Google apologises for Photos app's racist blunder

1 July 2015



Source

Steps to do Classification

Step 1: Start with the data you have

Applicant	Annual Income	Loan Approved?
#1	26 000	No
#2	60 000	Yes
#3	50 000	Yes
#4	47 000	No
#5	12 000	No
#6	108 000	Yes

Step 2: Split data into training and test sets

We chose a 50/50 split for our demo but you could do other splits like 60/40, 70/30. For large datasets, 80/20 split is used

Training Data

Applicant	Annual Income	Loan Approved?
#1	26 000	No
#2	60 000	Yes
#3	50 000	Yes

Test Data

Applicant	Annual Income	Loan Approved?
#4	47 000	No
#5	12 000	No
#6	108 000	Yes

Step 3: Build classifier
(i.e., Find pattern in training set)

Given your training data, can you find a pattern that can tell you when to approve a loan?

Earlier, we decided an annual income of ~$50,000 seemed Like a good cut off point. **That was a classifier!**

Applicant	Annual Income	Loan Approved?
#1	26 000	No
#2	60 000	Yes
#3	50 000	Yes

} **Training Data**

Step 4: Use classifier on test data

Applicant	Annual Income	Loan Approved?
#4	47 000	?
#5	12 000	?
#6	108 000	?

} **Test Data**

After you come up with a classifier that seems to do okay with your training data, you use it on your test data to see what kinds of decisions it makes.

Step 5: Calculate Accuracy

Applicant	Annual Income	Loan Approved?	Classifier said to…
#4	47 000	No	No
#5	12 000	No	No
#6	108 000	Yes	Yes

} **Test Data**

If the results of your classifier match up with the decisions you've made in your test data, it's looking good.

You can start trying to use it on data that you haven't made any decisions on yet.

Seems Straightforward

- What happens when we have more than one attribute?
- In the example before, we only had to consider annual income
- But what would happen if we had multiple attributes, like 5 or 10 or 100?
- **How do we decide which attribute to use?**

Seems Straightforward

- What happens when we have more than one attribute?
- In the example before, we only had to consider annual income
- But what would happen if we had multiple attributes, like 5 or 10 or 100?
- **How do we decide which attribute to use?**

Decision Trees!

Seems Straightforward

- What happens when we have more than one attribute?
- In the example before, we only had to consider annual income
- But what would happen if we had multiple attributes, like 5 or 10 or 100?
- **How do we decide which attribute to use?**

Decision Trees!

(next class!)

Wrap up