Principles of Safe Autonomy
ECE 498 SM
Lecture 1: Overview

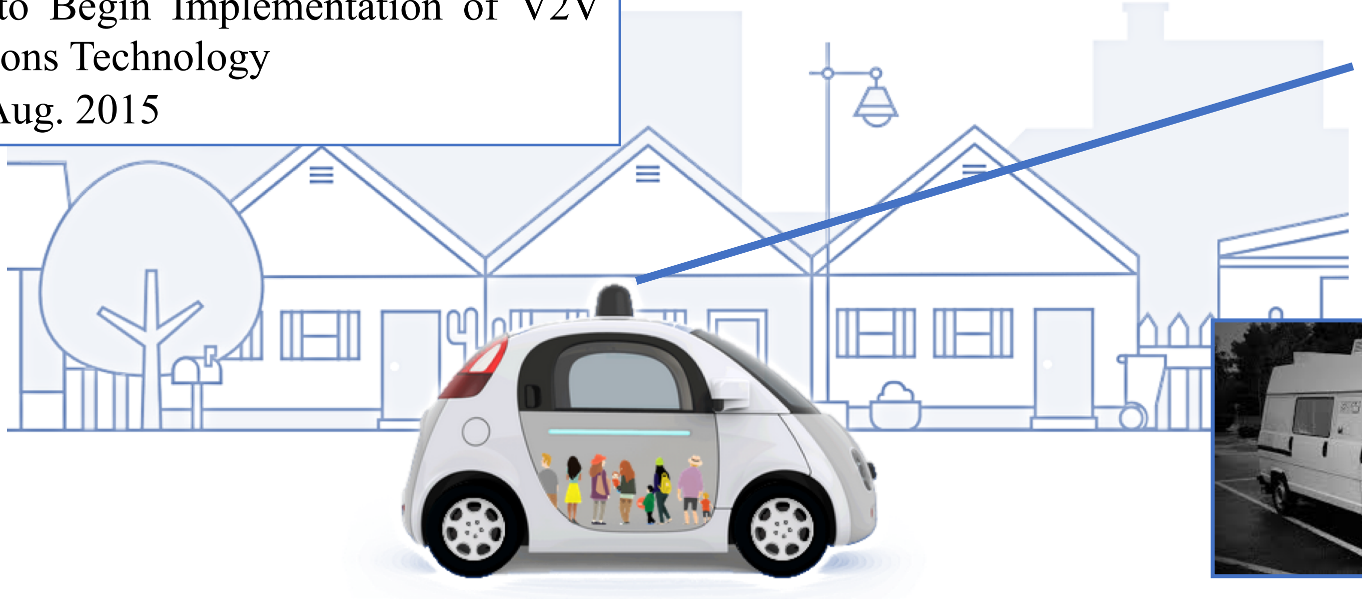
Katie Driggs-Campbell and Sayan Mitra

Chuchu Fan, Ted Liu, and Pulkit Katdare

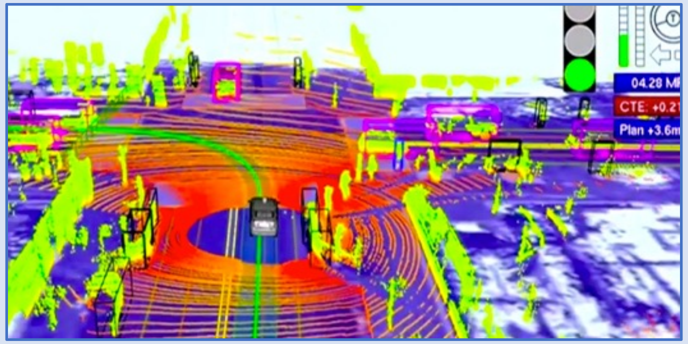
Cars are *communicating* more and more

USDOT Issues Advance Notice of Proposed Rulemaking to Begin Implementation of V2V Communications Technology

— NHTSA, Aug. 2015



Cars are *sensing* more and more



Demonstrations of driverless cars date back to the 80s/90s in the Eureka/ Prometheus Project

There is a greater societal push than ever before…

THE WALL STREET JOURNAL.

Home World U.S. Politics Economy Business Tech Markets Opinion Arts Life Real Estate

BUSINESS | AUTOS & TRANSPORTATION | AUTOS

U.S. Proposes Spending $4 Billion to Encourage Driverless Cars

Obama administration aims to remove hurdles to making autonomous cars more widespread

Autonomous Vehicles in the News

Science

Google promises autonomous cars for a

N

By

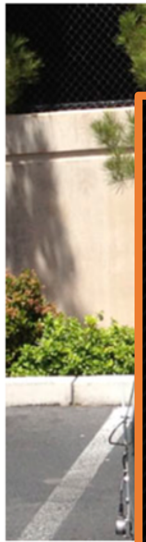
The Atlantic

TECHNOLOGY

Google's Self-Driving Cars: 300,000 Mil a Single Accident Under Computer Control

REBECCA J. ROSEN AUG 9, 2012

The automated cars are slowly building a driving record that's better than that of your average American.



MORE STORIES

The Quiet Way Automation Is Service Work

SIDNEY FUSSELL

WIRED

12.19.17

After Peak Hype, Self-Driving Cars Enter the Trough of Disillusionment

ars TECHNICA BIZ & IT TECH SCIENCE POLICY CARS GAMING & CUL

HYPE CYCLE —

The hype around driverless cars came crashing

earch

Bloomberg Businessw

December 31, 2018, 6:08 AM CST

eep

the

The 10 Worst Self-Driving Stories of 2018

How many ways can you say *Suboptimal*? Here's 10 more.

BY ALEX ROY DECEMBER 30, 2018

TECH | ANTHONY LEVANDOWSKI | AUDI | FULL SELF-DRIVING | HIGHWAY PILOT | NEWSWEEK | OPINION | PRONTO AI | SELF-DRIVING CARS | SUBOPTIMAL | TESLA | THE BORING COMPANY | TROUGH OF DISILLUSIONMENT | UBER | WAYMO | ZOOX

What could possibly go wrong?

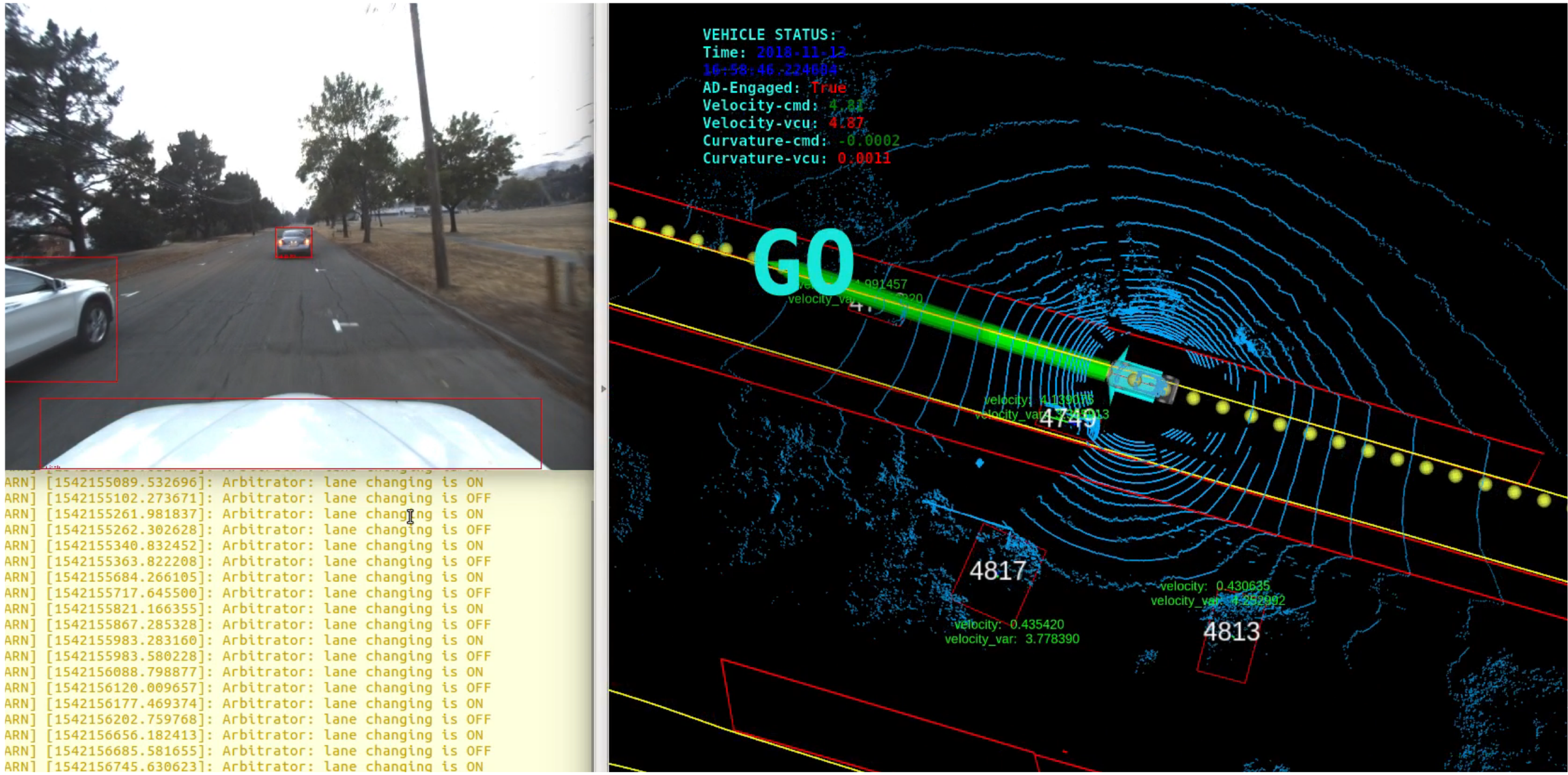
A lot of things. For example:

- sensor failure

What could possibly go wrong?

A lot of things. For example:

- sensor failure
- strict region of operation



What could possibly go wrong?

A lot of things. For example:

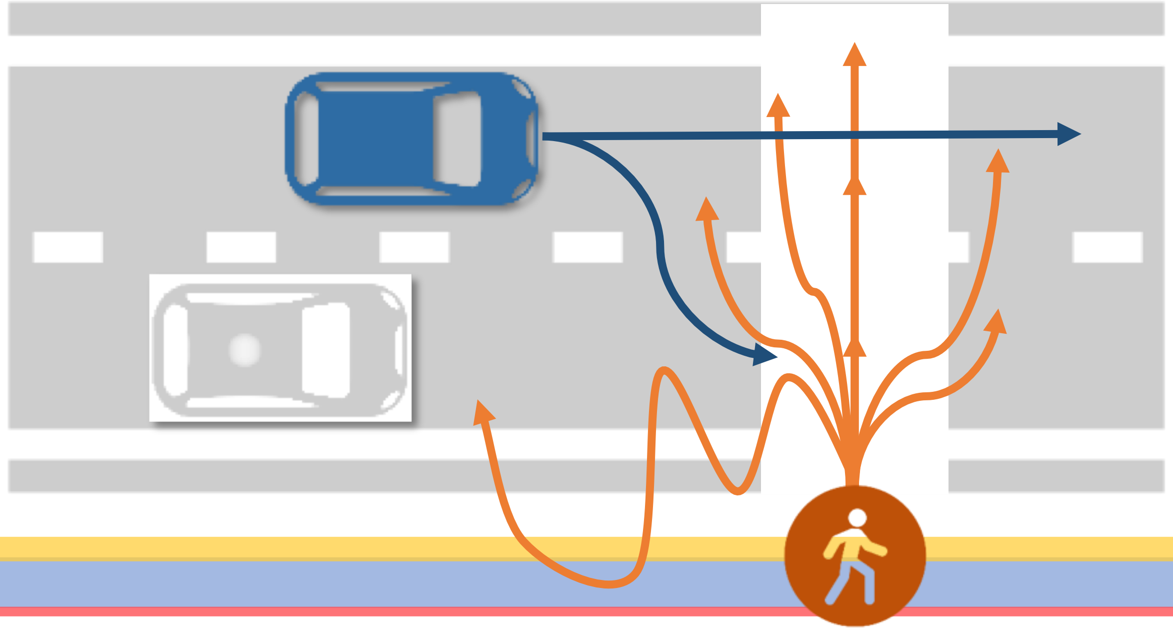
- sensor failure
- strict region of operation
- weird things happen

Low Probability, High Risk Events

Hazardous Event Frequencies	
Disengagement Rate	0.12 per 1000 km
Collision Rate	12.5 per 100 million km
Fatality Rate	0.70 per 100 million km

Low Probability, High Risk Events

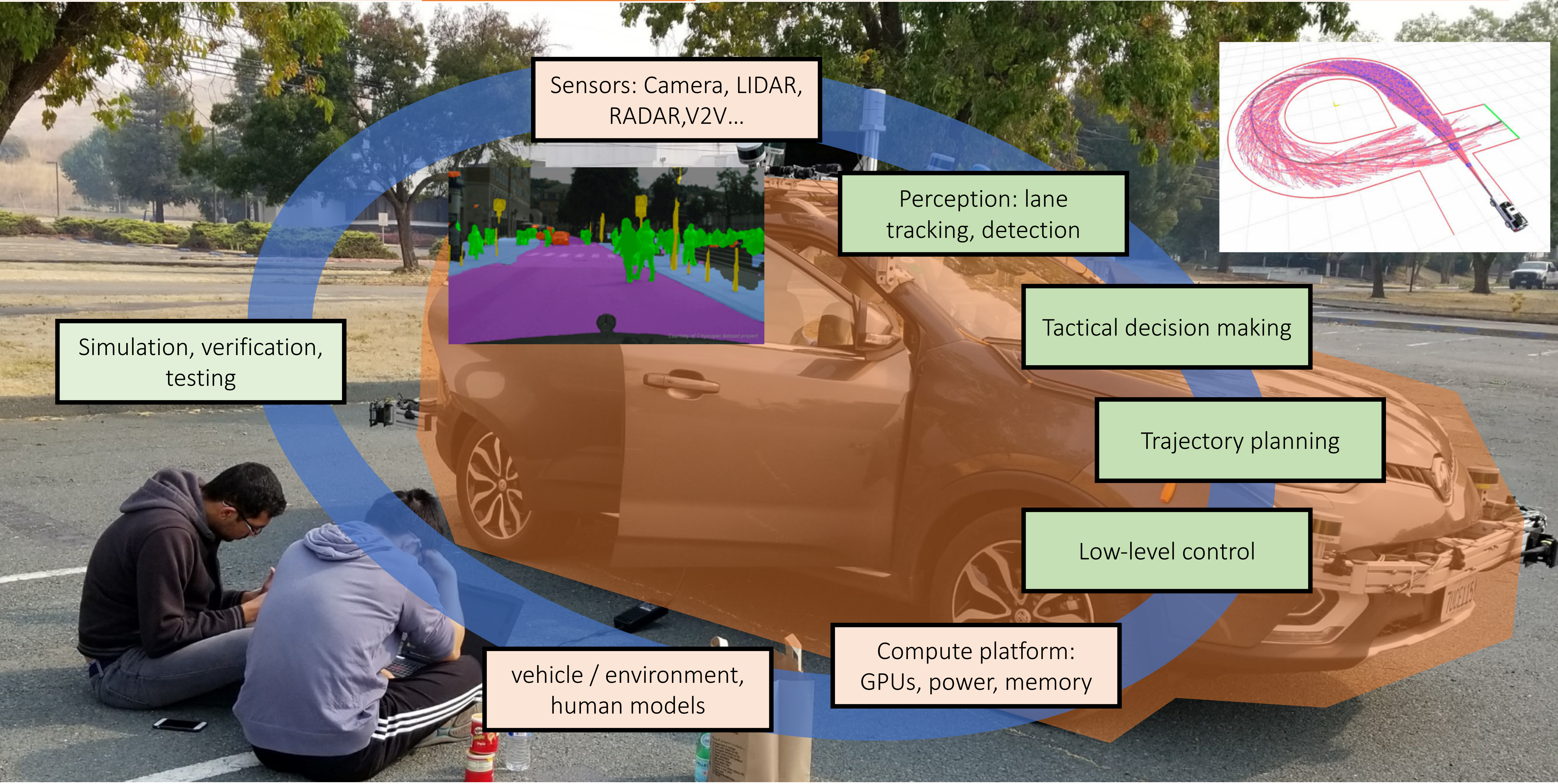
Hazardous Event Frequencies	
Disengagement Rate	0.12 per 1000 km
Collision Rate	12.5 per 100 million km
Fatality Rate	0.70 per 100 million km



- If an agent's motion is discretized, sampling will not give good coverage
- As more agents are added, the number of trajectories to test grows exponentially
- How can we verify such a complex, multi-agent system?


How do we create a safe and effective autonomous vehicle?

Sensors → Perception → Tactical decision making → Trajectory planning → Low level controller → Simulation and validation



Principles of Safe Autonomy

2019 Edition

About the course

Everything starts here: https://publish.illinois.edu/safe-autonomy/

- team
- schedule
- resources, papers, MPs, code, gitlab links
- lectures: https://gitlab.engr.illinois.edu/GolfCar/lectures

Piazza for Q&A, discussions

Compass for grades

Sayan Mitra (mitras)

Katie Driggs-Campbell (krdc)

Chuchu Fan (cfan10)

Ted Liu (tliu51)

Pulkit Katdare (katdare2)

Outlook of this class

Principles

- panoply of technologies behind the Self-driving project; we will zoom in on a few fundamental elegant ideas that we believe are also important

Practice

- hack, learn the language of autonomy today; use some of the latest and most interesting tools.

Fun

- “To do things right, first you need love, then technique.” – Antoni Gaudí

Course

Midterm and final exam (15 + 15%)

Analysis, concept questions, individual

March 4 midterm

May 1 in class final

Assignments, MPs 45%

5-6 sets, mostly coding, using tools, in groups

Project 20%

Start today! more in the next slides ...

Participation 5%

Notes, piazza, contribute code, make cool videos for class page, more...

Next 2 weeks: MPs

Next lecture very important!

MP0 will be released on **Wednesday Jan 16th**

- Get started asap (install FastX)
- Tutorial on using VMs, Righthook, ROS by Pulkit and Ted
- MP0 not graded, but the tools will be used for MP* and Projects
- MP1 will be released next week

10 minute project pitch/discussion with us **Wednesday Jan 23rd**

- Form your team now

Next 2 weeks: Project

Project ideas released today here

You can also design your own project

Important dates:

Jan 23	Form your team; discussion
Mar 13	Intermediate progress report
Apr 29	Poster and demos to course staff
May 11	Final report

Expectation: Explore new ideas, build end-to-end system, argue about safety/correctness
Outcomes: Technical papers, jumpstart grad research, incubate startup ideas

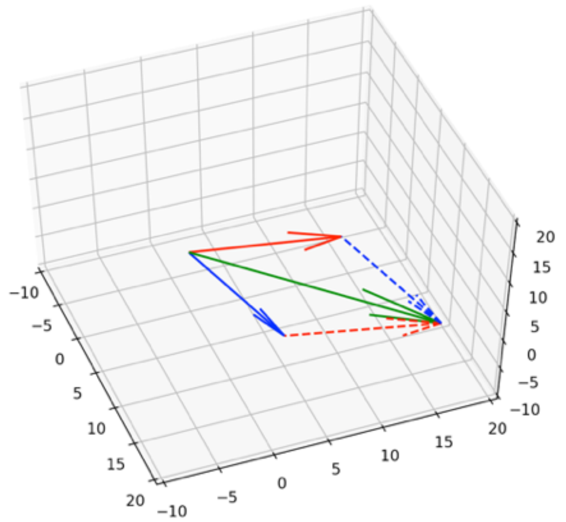
Participation

- Latex template for notes, reports and poster
- Make cool videos for the class
- Contribute answers on piazza, help troubleshoot
- Contribute code

A brief intro to some linear algebra concepts

- Linear spaces, linear transformation, linearly independent
- Norms
- Eigenvalues, eigenvectors
- Linear regression

Vectors



- a *vector* is an ordered list of numbers
  - $v = [1,0,0.5,2.8]; \overline{u} = (0,1,0,0); \boldsymbol{x} = (x + iy, 0, a + ib, -i)$
- the numbers are *scalars* that come from a *field*
- $v_4 = 2.8$; v_i is the i^{th} member or component of the vector
- unit vectors: $[0,0,1]; [0,1,0]; [1,0,0]$
- operations on vectors
  - addition $v + u$; commutative, associative, 0 identity, additive inverse
  - scalar multiplication av; associative, left and right distributive
- inner product: $v^T u = \sum v_i \times u_i$

Linear combinations

- A linear combination of a finite set of vectors $v_1, v_2, \dots, v_k$ is a vector $\sum_{i=1}^{k} \lambda_i v_i$ where $\lambda_i \in F$

- A set of vectors $v_1, v_2, \dots, v_k$ is *linearly independent* if the only solution of $\sum_{i=1}^{k} \lambda_i v_i$ =0 is the trivial solution $\lambda_1 = \lambda_2 = \cdots \lambda_k = \mathbf{0}$

- The set of vectors $v_1, v_2, \dots, v_k$ is *linearly dependent* if one of the vectors can be written as a linear combination of the others, that is, $v_1 = \sum_{i=2}^{k} \lambda_i v_i$

Norms

A norm of a vector measures its size

- Any function $f: \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ such that
  - Homogeneity: $f(ax) = af(x)$
  - Triangle inequality: $f(x+y) \leq f(x) + f(y)$
  - Definiteness: $f(x) = 0 \Leftrightarrow x = \mathbf{0}$
- Euclidean $|x|_2 = \sqrt{x_1^2 + x_2^2 + \ \ldots + x_n^2}$
  - $|x-y|_2$ is the standard notion of Euclidean distance between x and y
- $|x|_1 = |x_1| + \cdots + |x_n|$
- $|x|_\infty = \max(x_1, x_2, \ldots, x_n)$
- $\mathrm{rms}(x) = \frac{|x|_2}{\sqrt{n}}$

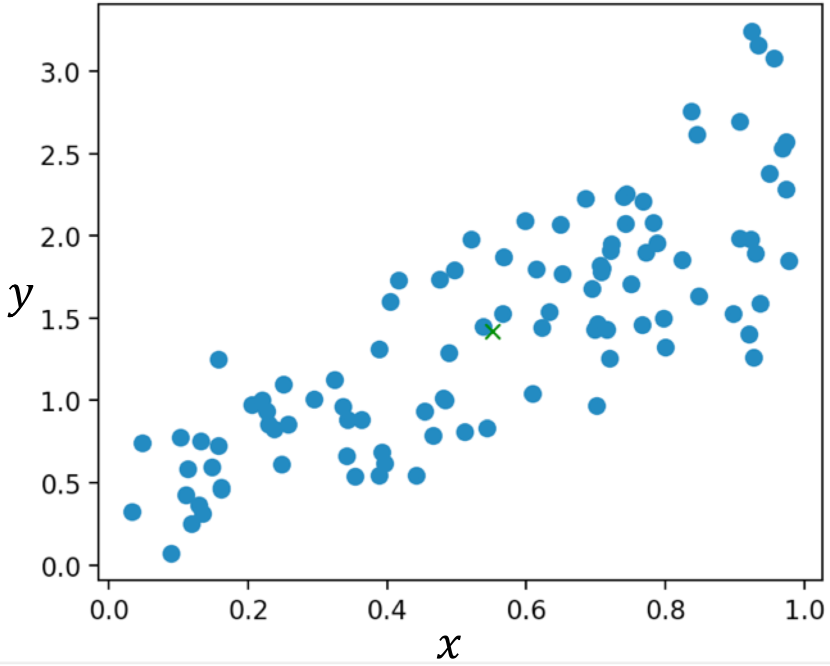
Means and deviations

- sum $\mathbf{1}^T x = x_1 + x_2 + \ \ldots + x_n$
- mean, average: $\bar{x} = \frac{1}{\mathrm{n}} \mathbf{1}^T x$
- demeaned vectors: $\tilde{x} = x - \bar{x}$
  - $\bar{\tilde{x}} = ?$
- standard deviation: $\mathbf{rms}(\tilde{x}) = \frac{1}{\sqrt{n}} |x - \bar{x}|_2$
- correlation: $\rho_{x,y} = \frac{\tilde{x}^T \tilde{y}}{|\tilde{x}|_2 |\tilde{y}|_2}$

Regression

We would like to *fit* a model to a bunch of data points

- Affine model: $y = x^T \theta + b$
  - $x \in \mathbb{R}^n$ "feature" vector
  - $\theta \in \mathbb{R}^n$ vector of coefficients
  - $b \in \mathbb{R}$ an offset
  - $y \in \mathbb{R}$ is the prediction, dependent variable



Fitting models to data

Suppose we have m data points $(y^1, x^1), (y^2, x^2), \ldots, (y^m, x^m)$

We would like to find the coefficients $\theta \in \mathbb{R}^n$ of linear combination s.t.

$$y^1 = x_1^1\theta_1 + x_2^1\theta_2 + \cdots + x_k^1\theta_k$$

$$y^m = x_1^m\theta_1 + x_2^m\theta_2 + \cdots + x_k^m\theta_k$$

Many more equations than unknowns, overdetermined system, so we will not find a solution that works

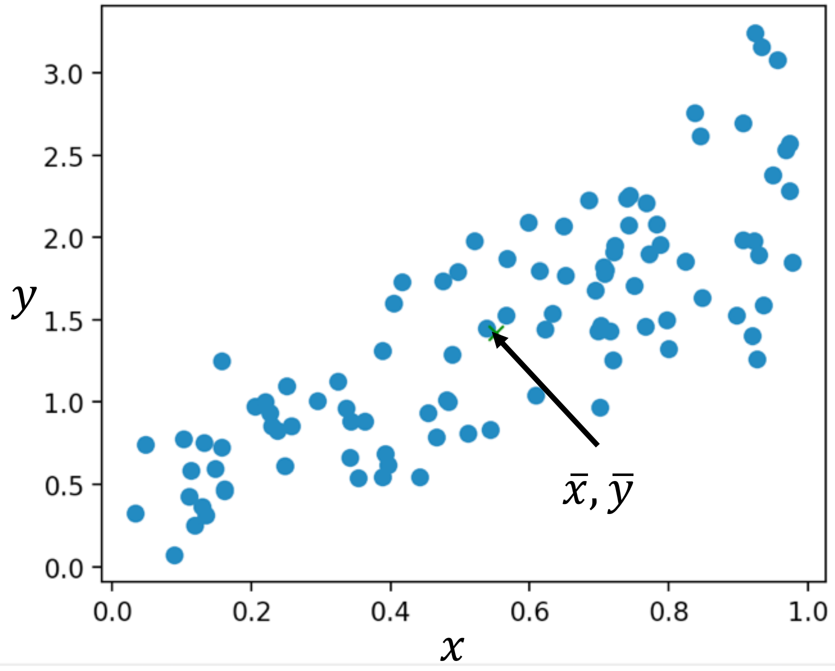
Given θ, $\hat{y}^1 = (x^1)^T\theta$ is the prediction for y

Prediction error or residual $r_1 = \hat{y}^1 - y^1 = (x^1)^T\theta - y^1$

$$r = \hat{y} - y = \begin{bmatrix} x_1^1 & \cdots & x_k^1 \\ \vdots & \vdots & \vdots \\ x_1^m & \cdots & x_k^m \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_k \end{bmatrix} - y$$

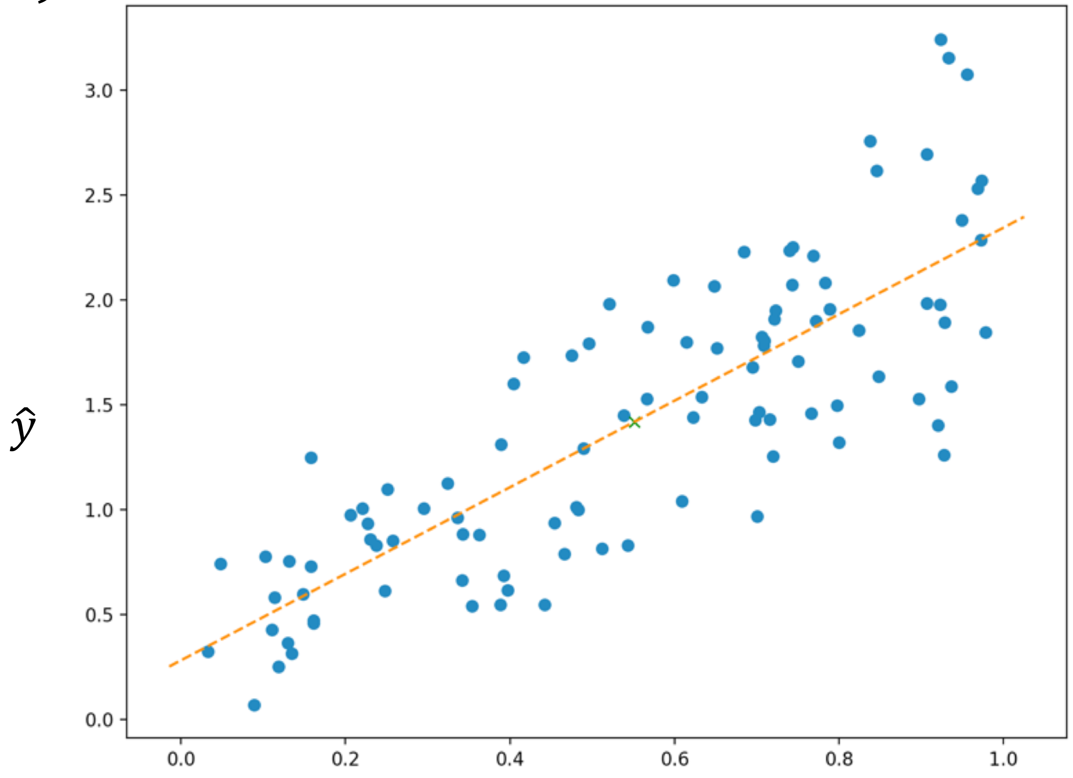
Reduce the RMS error on r: choose θ such that minimize prediction $\left(\frac{r_1^2 + r_2^2 + \cdots r_m^2}{n}\right)^{\frac{1}{2}}$ error on the data set

This is called the *Least Squares* problem: choose θ to make r as small as possible if not 0



Simple regression: straight line fit

- m data points $\boldsymbol{x} = (x^1, \ldots, x^m); \boldsymbol{y} = (y^1, \ldots, y^m)$
- $y^i, x^i \in \mathbb{R}$
- $\hat{y}^i = b + \theta x^i$
- can work out θ_0, θ_1 explicitly
  - $\theta = \rho_{\boldsymbol{x},\boldsymbol{y}} \dfrac{rms(\tilde{\boldsymbol{x}})}{rms(\tilde{\boldsymbol{y}})}$
  - $b = \overline{\boldsymbol{y}} - \theta_1 \, \overline{\boldsymbol{x}}$



- Code

Clustering

Given N vectors $x_1, \dots, x_N \in \mathbb{R}^n$, the goal is to partition them into k groups so that the vectors in the same group are close to one another

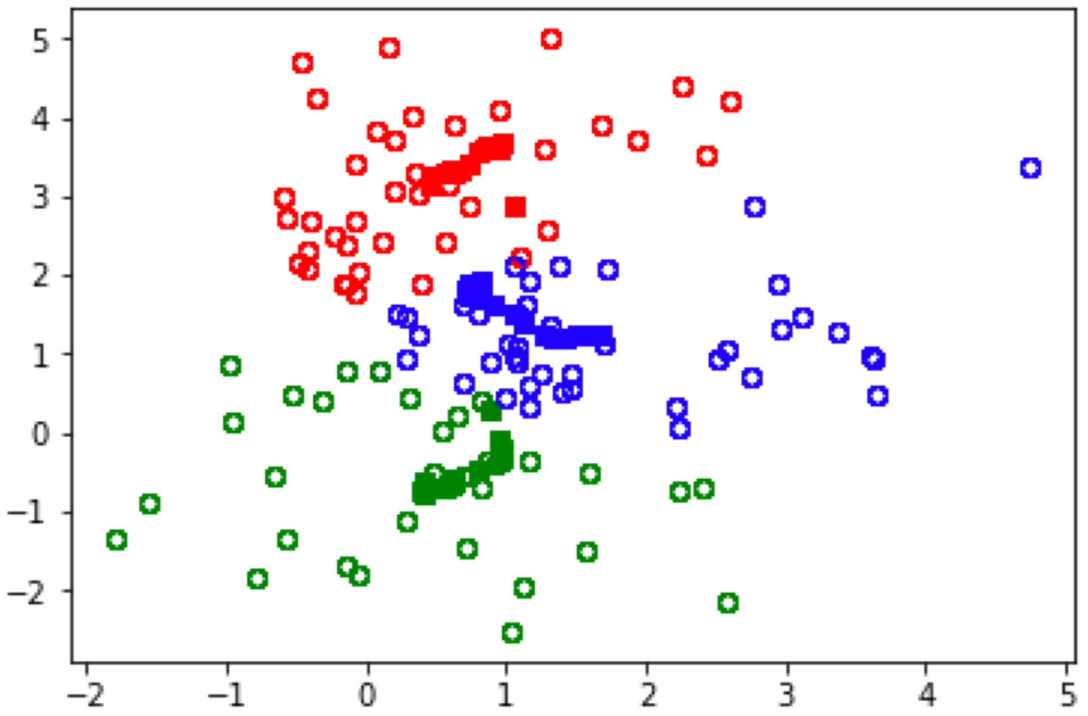
Examples: image compression (vectors are pixel values); patient clustering (patient attributes, tests);

$c_i \in \{1, \dots, k\}$ is the group x_i belongs to

$G_{c_i} \subseteq \{x_1, \dots, x_n\}$ group

z_{c_i} group representative

Clustering objective minimize $\mathrm{J}_{clust} = \frac{1}{N}\sum_{i=1}^{N}\left|x_i - z_{c_i}\right|^2$ by choosing the groups $\{c_i\}$ and the representatives



from Boyd & Vandenberghe

Algorithm: Step 1

- Suppose the representatives $z_1, \dots, z_k$ are given, how do we assign the vectors $x_1, \dots, x_N$ to the k groups?
- Recall $\mathrm{J}_{clust} = \frac{1}{n}\sum_{i=1}^{N}\left|x_i - z_{c_i}\right|^2$
- $\min\limits_j \frac{1}{N}\sum_{i=1}^{N}\left|x_i - z_j\right|^2 = \frac{1}{N}\sum_{i=1}^{N}\min\limits_j\left|x_i - z_j\right|^2$
- That is, assign x_i to the nearest representative z_j

Algorithm: Step 2

- Given the partition $G_1, \ldots, G_k$, how to choose the representatives $z_1, \ldots, z_k$?
- $\mathrm{J}_{clust} = J_1 + \cdots + J_k = \sum_{j=1}^{k} \frac{1}{|G_j|} \sum_{i \in G_j} \left| x_i - z_j \right|^2$
- Choose z_j to minimize J_j, that is $z_j = \frac{1}{|G_j|} \sum_{i \in G_j} x_i$ the mean (centroid)

Algorithm: Combined

- alternate between updating the partition, then the representatives
- a famous algorithm called *k-means clustering*
- objective J_{clust} decreases in each step

given $x_1, \ldots, x_N \in \mathbb{R}^n$ and $z_1, \ldots, z_k \in \mathbb{R}^n$
repeat
    update partition: assign i to G_j, $j \;=\; \operatorname{argmin}_j \left| x_i - \; z_j \right|_2^2$
    update centroids: $z_j \;= \frac{1}{|P_j|} \sum_{i \in P_j} x_i$
until $z_1, \ldots, z_k$ stop changing

Linear transformations

- A linear function or linear transformation $f: \mathbb{R}^n \to \mathbb{R}^m$ has the property that $f(\alpha v_1 + \beta v_2) = \alpha f(v_1) + \beta f(v_2)$

- Any linear function can be written as $y = Ax$, where $A \in \mathbb{R}^{m \times n}$

Examples

- Projection on x: $\Pi_x = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$
- CCW rotation by θ: $\mathrm{R}_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$

Eigenvalues and eigenvectors

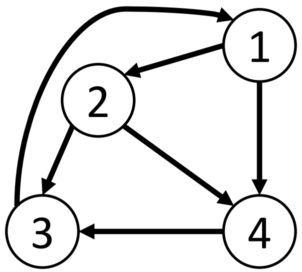
- The set of eigenvectors of a matrix A is a special set of input vectors for which the action A is described as a simple scaling
- If $v \in \mathbb{R}^n$ is an eigenvector, then $Av = \lambda v$ for some scalar λ
- To find eigenvectors and eigenvalues, solve for the roots of the characteristic polynomial $\mathrm{p}(\lambda) = \det(A - \lambda I)$
- This gives n roots which are the n eigenvalues
- For each eigenvalue λ, solve $(A - \lambda I)v = 0$ to find an eigenvector v.

`python function eig(A)`

Eigenvectors as limits of repeated improvement

Application: ranking web pages

- Each webpage (node) i has a PageRank $r_i \in \mathbb{R}$
- r_i is refined repeatedly according to the update rule: Each page divides its current PageRank equally among the outgoing links
- N_{ij}: Portion of i's PageRank that j should get in one step; $N_{ij} = 1/outdeg_i$
- $r_i(k+1) = N_{1i}r_1(k) + N_{2i}r_2(k) + \cdots + N_{ni}r_n(k)$
- $r(k+1) = N^T r(k)$
- Say, $r(0) = \left[\frac{1}{n}, \frac{1}{n}, \ldots \frac{1}{n}\right]$, what is $r(k) = (N^T)^k\, r(0)$?
- If the update rule converges in the limit, then we expect $r(*) = N^T\, r(*)$; that is $r(*)$ to be an eigenvector of N with corresponding eigenvalue of 1.



$$N = \begin{bmatrix} 0 & .5 & 0 & .5 \\ 0 & 0 & .5 & .5 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$N_s = \begin{bmatrix} .05 & .45 & .05 & .45 \\ .05 & .05 & .45 & .45 \\ .85 & .05 & .05 & .05 \\ .05 & .05 & .85 & .05 \end{bmatrix}$$

Perron's theorem. Any matrix N with all positive entries has a real eigenvalue $\lambda_{max} > 0$ such that $\lambda_{max} > |\lambda|$ for all other eigenvalues λ.

There is an eigenvector v corresponding to λ_{max} with positive real coordinates that is unique up to scaling. If $\lambda_{max} = 1$, then for any starting vector x the sequence $P^k x$ converges to a vector in the direction of v as k goes to infinity.

The max eigenvector of the scaled matrix N_s gives the stable PageRank!

The PageRank Citation Ranking: Bringing Order to the Web

January 29, 1998

Abstract

The importance of a Web page is an inherently subjective matter, which depends on the readers interests, knowledge and attitudes. But there is still much that can be said objectively about the relative importance of Web pages. This paper describes PageRank, a method for rating Web pages objectively and mechanically, effectively measuring the human interest and attention devoted to them.

We compare PageRank to an idealized random Web surfer. We show how to efficiently compute PageRank for large numbers of pages. And, we show how to apply PageRank to search and to user navigation.

1 Introduction and Motivation

The World Wide Web creates many new challenges for information retrieval. It is very large and heterogeneous. Current estimates are that there are over 150 million web pages with a doubling life of less than one year. More importantly, the web pages are extremely diverse, ranging from "What is Joe having for lunch today?" to journals about information retrieval. In addition to these major challenges, search engines on the Web must also contend with inexperienced users and pages engineered to manipulate search engine ranking functions.

However, unlike "flat" document collections, the World Wide Web is hypertext and provides considerable auxiliary information on top of the text of the web pages, such as link structure and link text. In this paper, we take advantage of the link structure of the Web to produce a global "importance" ranking of every web page. This ranking, called PageRank, helps search engines and users quickly make sense of the vast heterogeneity of the World Wide Web.

1.1 Diversity of Web Pages

Although there is already a large literature on academic citation analysis, there are a number of significant differences between web pages and academic publications. Unlike academic papers which are scrupulously reviewed, web pages proliferate free of quality control or publishing costs. With a simple program, huge numbers of pages can be created easily, artificially inflating citation counts. Because the Web environment contains competing profit seeking ventures, attention getting strategies evolve in response to search engine algorithms. For this reason, any evaluation strategy which counts replicable features of web pages is prone to manipulation. Further, academic papers are well defined units of work, roughly similar in quality and number of citations, as well as in their purpose – to extend the body of knowledge. Web pages vary on a much wider scale than academic papers in quality, usage, citations, and length. A random archived message posting

Sergey Brin received his B.S. degree in mathematics and
he is a Ph.D. candidate in computer science at Stanford U
Foundation Graduate Fellowship. His research interests ir
of large text collections and scientific data.

Lawrence Page was born in East Lansing, Michigan, and
1995. He is currently a Ph.D. candidate in Computer Scie
web, human computer interaction, search engines, scalabi

8 Appendix A: Advertising and Mix

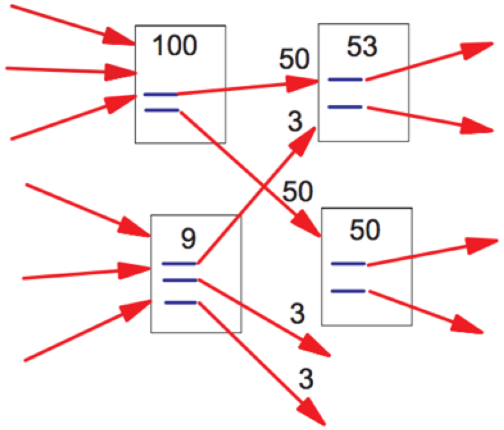
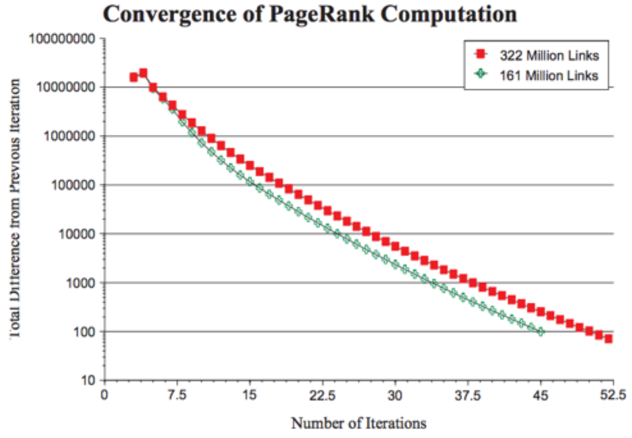


Figure 2: Simplified PageRank Calculation



Exercise: Try coding PageRank; It is essentially just matrix multiplication or computing eigenvectors.

Summary

- Vector operations and norms
- Regression
- Clustering
- Eigenvalues and eigenvectors

- Homework
  - Form team; decide on project; we discuss in 2 weeks
  - Start MP0 before next lecture