Learning Math - Whose Idea Was That?!

Jessica Sorrell

EMC

June 29, 2016
Outline

- High School
- College
- Recurse Center
- EMC
- Wrap it Up
High School

College

Recurse Center

EMC

Wrap it Up
“What’s wrong here?”

Let $x = 1$
“What’s wrong here?”

Let $x = 1$

$x = x^2$
“What’s wrong here?”

Let $x = 1$

\[ x = x^2 \]

\[ x - 1 = x^2 - 1 \]
“What’s wrong here?”

Let $x = 1$

$$x = x^2$$

$$x - 1 = x^2 - 1$$

$$\frac{x - 1}{x - 1} = \frac{x^2 - 1}{x - 1}$$
“What’s wrong here?”

Let $x = 1$

$x = x^2$

$x - 1 = x^2 - 1$

\[
\frac{x - 1}{x - 1} = \frac{x^2 - 1}{x - 1}
\]

$1 = x + 1$
“What’s wrong here?”

Let $x = 1$

$$x = x^2$$

$$x - 1 = x^2 - 1$$

$$\frac{x - 1}{x - 1} = \frac{x^2 - 1}{x - 1}$$

$$1 = x + 1$$

$$1 = 2$$
Mr. Johnson’s Classroom
...Is Mr. Johnson a Tool fan?

Vision Crystal by Alex Grey

Jessica Sorrell
Learning Math - Whose Idea Was That?! 7 / 31
"The Mandelbrot set is the set of complex numbers $c$ for which the function

$$f_c(z) = z^2 + c$$

does not diverge when iterated from $z = 0$ ..."

- Wikipedia
"The Mandelbrot set is the set of complex numbers $c$ for which the function

$$f_c(z) = z^2 + c$$

does not diverge when iterated from $z = 0$ ..."

- Wikipedia
What Mr. Johnson Taught Me

Math is not an arbitrary set of rules created by the ur-teacher
What Mr. Johnson Taught Me

Math is not an arbitrary set of rules created by the ur-teacher

Simple rules can yield complex structure
What Mr. Johnson Taught Me

Math is not an arbitrary set of rules created by the ur-teacher

Simple rules can yield complex structure

Calculus!

\[ f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \ldots \]
What Mr. Johnson Taught Me

How much fun it is to ask good questions and use math to get sound answers.
High School

College

Recurse Center

EMC

Wrap it Up
“The Recurse Center is a free, self-directed, educational retreat for people who want to get better at programming, whether they’ve been coding for three decades or three months.”

- Recurse Center landing page
Arabic programming language

Fibonacci sequence
source code

امر عامل فب (ن) ليسوي
د = 0
لو ن > 2 سوي
د = ن
والا
ش = ن - 1
س = ن - 2
م = فب(ش)
ك = فب(س)
د = م + ك

باس

باس

اطبع (فب(0)
اطبع (فب(1)
اطبع (فب(10)

Ahmed Abdalla
Markov Chain Twitterbots

Samantha Goldstein

LeBron James Joyce Twitterbot

LeBron James Joyce
@LBJamesJoyce

this is bigger than basketball the dark autumn evening in you,

3:24 PM - 20 Jul 2015

1 1
Markov Chain Twitterbots

“You have to be able to accept failure to get better.” - LeBron James
Markov Chain Twitterbots

“You have to be able to accept failure to get better.” - LeBron James

Diagram showing transitions between states such as 'You', 'have', 'to', 'be', 'better', 'get', 'failure', 'accept', 'able' with probabilities indicated on the arrows.
Markov Chain Twitterbots

“You have to be able to accept failure to get better.” - LeBron James

$$M_{i,j} = P(\text{word}_j \mid \text{word}_i)$$

$$
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.3 & 0 & 0.3 & 0 & 0.3 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$
Markov Chain Twitterbots

\[
\begin{pmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.3 & 0 & 0.3 & 0 & 0.3 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
\]

\[
\begin{pmatrix}
0 & 0 & 0 & 0.3 & 0 & 0.3 & 0.3 & 0 & 0
\end{pmatrix}
\]
The Social Rules

- No feigning surprise
- No well-actually’s
- No back-seat driving
- No subtle -isms
High School

College

Recurse Center

EMC

Wrap it Up
Soon to be....
Software Engineering

Cluster
Software Engineering

Cluster

Xbrick

Solid-state drive (SSD)
An engineer at EZIO is trying to design a new storage device

- RealCompany SSDs have an annual failure probability of 0.009
Word Problem!

An engineer at EZIO is trying to design a new storage device

- RealCompany SSDs have an annual failure probability of 0.009
- The engineer wants each EZbrick to contain 25 drives

What is the probability of a drive failure in any of those 100 EZbricks over the span of their lifetime?
Word Problem!

An engineer at EZIO is trying to design a new storage device

- RealCompany SSDs have an annual failure probability of 0.009
- The engineer wants each EZbrick to contain 25 drives
- The lifetime of an EZbrick is about 5 years
Word Problem!

An engineer at EZIO is trying to design a new storage device

- RealCompany SSDs have an annual failure probability of 0.009
- The engineer wants each EZbrick to contain 25 drives
- The lifetime of an EZbrick is about 5 years
- The engineer expects to sell 100 EZbricks as soon as they become available
An engineer at EZIO is trying to design a new storage device

- RealCompany SSDs have an annual failure probability of 0.009
- The engineer wants each EZbrick to contain 25 drives
- The lifetime of an EZbrick is about 5 years
- The engineer expects to sell 100 EZbricks as soon as they become available

What is the probability of a drive failure in any of those 100 EZbricks over the span of their lifetime?
Word Problem!

<table>
<thead>
<tr>
<th>Annual failure probability</th>
<th>0.009</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drives per EZbrick</td>
<td>25</td>
</tr>
<tr>
<td>EZbrick lifetime</td>
<td>5 years</td>
</tr>
<tr>
<td>EZbricks</td>
<td>100</td>
</tr>
</tbody>
</table>
Word Problem!

<table>
<thead>
<tr>
<th>Annual failure probability</th>
<th>0.009</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drives per EZbrick</td>
<td>25</td>
</tr>
<tr>
<td>EZbrick lifetime</td>
<td>5 years</td>
</tr>
<tr>
<td>EZbricks</td>
<td>100</td>
</tr>
</tbody>
</table>

\[
P(\text{failure in single EZbrick after 1 year}) = 1 - (1 - 0.009)^{25} \approx 0.20
\]
Word Problem!

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Annual failure probability</td>
<td>0.009</td>
</tr>
<tr>
<td>Drives per EZbrick</td>
<td>25</td>
</tr>
<tr>
<td>EZbrick lifetime</td>
<td>5 years</td>
</tr>
<tr>
<td>EZbricks</td>
<td>100</td>
</tr>
</tbody>
</table>

\[ P(\text{failure in single EZbrick after 1 year}) = 1 - (1 - 0.009)^{25} \approx 0.20 \]

\[ P(\text{failure in single EZbrick after lifetime}) = 1 - (1 - 0.20)^5 \approx 0.67 \]
**Word Problem!**

<table>
<thead>
<tr>
<th>Annual failure probability</th>
<th>0.009</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drives per EZbrick</td>
<td>25</td>
</tr>
<tr>
<td>EZbrick lifetime</td>
<td>5 years</td>
</tr>
<tr>
<td>EZbricks</td>
<td>100</td>
</tr>
</tbody>
</table>

\[
P(\text{failure in single EZbrick after 1 year}) = 1 - (1 - 0.009)^{25} \approx 0.20
\]

\[
P(\text{failure in single EZbrick after lifetime}) = 1 - (1 - 0.20)^5 \approx 0.67
\]

\[
P(\text{failure in 100 EZbricks after lifetime}) = 1 - (1 - 0.67)^{100} \approx .99
\]
Word Problem!

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Annual failure probability</td>
<td>0.009</td>
</tr>
<tr>
<td>Drives per EZbrick</td>
<td>25</td>
</tr>
<tr>
<td>EZbrick lifetime</td>
<td>5 years</td>
</tr>
<tr>
<td>EZbricks</td>
<td>100</td>
</tr>
</tbody>
</table>

\[
P(\text{failure in single EZbrick after 1 year}) = 1 - (1 - 0.009)^{25} \approx 0.20
\]

\[
P(\text{failure in single EZbrick after lifetime}) = 1 - (1 - 0.20)^5 \approx 0.67
\]

\[
P(\text{failure in 100 EZbricks after lifetime}) = 1 - (1 - 0.67)^{100} \approx 0.99
\]

This is going to happen
RAID

Redundant Array of Inexpensive/Independent Disks

RAID 4

Disk 0
A1
B1
C1
D1

Disk 1
A2
B2
C2
D2

Disk 2
A3
B3
C3
D3

Disk 3
A_p
B_p
C_p
D_p
RAID

Let’s say each “chunk” of a disk stores 1 bit, either a 0 or a 1
Let’s say each “chunk” of a disk stores 1 bit, either a 0 or a 1

\[ A_p \equiv A_1 + A_2 + A_3 \pmod{2} \]
Let’s say each “chunk” of a disk stores 1 bit, either a 0 or a 1

\[ A_p \equiv A_1 + A_2 + A_3 \pmod{2} \]

\[ A_1 = 1, \quad A_2 = 1, \quad A_3 = 0, \quad A_p = \]
Let’s say each “chunk” of a disk stores 1 bit, either a 0 or a 1

\[ A_p \equiv A_1 + A_2 + A_3 \pmod{2} \]

\[ A_1 = 1, \quad A_2 = 1, \quad A_3 = 0, \quad A_p = 0 \]
Let’s say each “chunk” of a disk stores 1 bit, either a 0 or a 1

\[ A_p \equiv A_1 + A_2 + A_3 \pmod{2} \]

\[ A_1 = 1, \quad A_2 = 1, \quad A_3 = 0, \quad A_p = 0 \]

\[ A_1 = 0, \quad A_2 = 1, \quad A_3 = 0, \quad A_p = \]
RAID

Let’s say each “chunk” of a disk stores 1 bit, either a 0 or a 1

\[ A_p \equiv A_1 + A_2 + A_3 \pmod{2} \]

\[ A_1 = 1, \quad A_2 = 1, \quad A_3 = 0, \quad A_p = 0 \]

\[ A_1 = 0, \quad A_2 = 1, \quad A_3 = 0, \quad A_p = 1 \]
Let’s say each “chunk” of a disk stores 1 bit, either a 0 or a 1.

\[ A_p \equiv A_1 + A_2 + A_3 \pmod{2} \]

- **Case 1:**
  - \( A_1 = 1 \), \( A_2 = 1 \), \( A_3 = 0 \)
  - \( A_p = 0 \)

- **Case 2:**
  - \( A_1 = 0 \), \( A_2 = 1 \), \( A_3 = 0 \)
  - \( A_p = 1 \)

- **Case 3:**
  - \( A_1 = 1 \), \( A_2 = 1 \), \( A_3 = 1 \)
  - \( A_p = \)
Let’s say each “chunk” of a disk stores 1 bit, either a 0 or a 1

\[ A_p \equiv A_1 + A_2 + A_3 \pmod{2} \]

\[ A_1 = 1, \quad A_2 = 1, \quad A_3 = 0, \quad A_p = 0 \]

\[ A_1 = 0, \quad A_2 = 1, \quad A_3 = 0, \quad A_p = 1 \]

\[ A_1 = 1, \quad A_2 = 1, \quad A_3 = 1, \quad A_p = 1 \]
So when a drive goes down, the data that was stored in it can be recovered!

\[ A_1 = 1, \quad A_2 = ???, \quad A_3 = 1, \quad A_p = 0 \]
So when a drive goes down, the data that was stored in it can be recovered!

\[ A_1 = 1, \quad A_2 = ???, \quad A_3 = 1, \quad A_p = 0 \]

\[ A_p \equiv A_1 + A_2 + A_3 \pmod{2} \]
So when a drive goes down, the data that was stored in it can be recovered!

\[ A_1 = 1, \quad A_2 = \text{???,} \quad A_3 = 1, \quad A_p = 0 \]

\[ A_p \equiv A_1 + A_2 + A_3 \quad (\text{mod} \ 2) \]

\[ 0 \equiv 1 + x + 1 \quad (\text{mod} \ 2) \]
So when a drive goes down, the data that was stored in it can be recovered!

\[ A_1 = 1, \quad A_2 = ???, \quad A_3 = 1, \quad A_p = 0 \]

\[ A_p \equiv A_1 + A_2 + A_3 \pmod{2} \]
\[ 0 \equiv 1 + x + 1 \pmod{2} \]
\[ 0 \equiv 2 + x \pmod{2} \]
So when a drive goes down, the data that was stored in it can be recovered!

\[ A_1 = 1, \quad A_2 = ???, \quad A_3 = 1, \quad A_p = 0 \]

\[
A_p \equiv A_1 + A_2 + A_3 \pmod{2}
\]

\[
0 \equiv 1 + x + 1 \pmod{2}
\]

\[
0 \equiv 2 + x \pmod{2}
\]

\[
0 \equiv x \pmod{2}
\]
High School

College

Recurse Center

EMC

Wrap it Up
Math helps us answer questions…

...about beautiful things

...about silly things

...about business-critical things
You give us the confidence to answer questions of our own
You give us the confidence to answer questions of our own

Thank you!
You give us the confidence to answer questions of our own

Thank you!

Email: optimistindustries@gmail.com
Web: optimistindustries.com
Twitter: @optimistsinc