Sage-Enabled Textbooks (A Potpourri)

Sage Days: Opening Workshop for a Year of Coding Sprints

Rob Beezer

August 22, 2017

Department of Mathematics and Computer Science
University of Puget Sound
Tacoma, Washington, USA
beezer@pugetsound.edu

Computational Approaches to Undergraduate Mathematics

Sage Exercise 10.5.6: Construct some dihedral groups of order $2n$ (i.e. symmetries of an n-gon, D_n in the text, `DihedralGroup(n)` in Sage). Maybe all of them for $3 \leq n \leq 100$.

```
n = 15
G = DihedralGroup(n)
N = G.normal_subgroups()
[H.order() for H in N]
```

Can you describe all of the normal subgroups of a dihedral group in a way that would let us predict all of the normal subgroups of D_{470448} without using Sage? $470448 = 2^4 \cdot 3^5 \cdot 11^2$

DEMO: JUDSON'S ABSTRACT ALGEBRA

A Sage-Enabled Book

Produced from PreTeXt Source

Computation in Undergraduate Mathematics

- Recognized in *The Mathematical Sciences in 2025* from the National Research Council
- Student email about Sage experience at REU

How Did We Get Here?

Sage in Textbooks

- 2003: Open-source linear algebra textbook, LaTeX
- 2008: Austrian contributor: "Seen Sage?"
- 2009: Sage development – group theory, graph theory
- 2010: Major contributions to linear algebra
- 2010-13: Convert LaTeX to Sage Notebooks (SageNB)
- 2011: Sage Cell Server, Jason Grout, et. al.
- 2013: Founded PreTeXt project (nee MathBook XML)

PreTeXt Authoring System

- An XML "vocabulary"
- Delivers on promise of content-only, presentation-never
- LaTeX output, HTML output
- Experimental: EPUB, Jupyter, CoCalc
- Integration of Sage code is primary motivation

  LaTeX: print input and output

  HTML: Sage Cells

  Jupyter, CoCalc: code cell

  Doctest: compare input and expected output

PreTeXt XML Technical Example

```
<theorem xml:id="power-rule">
    <title>Power Rule</title>
    <idx>power rule</idx>

    <statement>
        <p>The derivative of <m>f(x)=x^n</m>
        is <m>f'(x)=nx^{n-1}</m>.</p>
    </statement>

    <proof>
        <p>Apply induction to the product
        <me>f(x)=x^n=x\cdot x^{n-1}</me>
        using <xref ref="product-rule"/>.</p>
    </proof>
</theorem>
```

Theorem 4.4 (Power Rule)**.** *The derivative of* $f(x) = x^n$ *is* $f'(x) = nx^{n-1}$.

Proof. Apply induction to the product

$$f(x) = x^n = x \cdot x^{n-1}$$

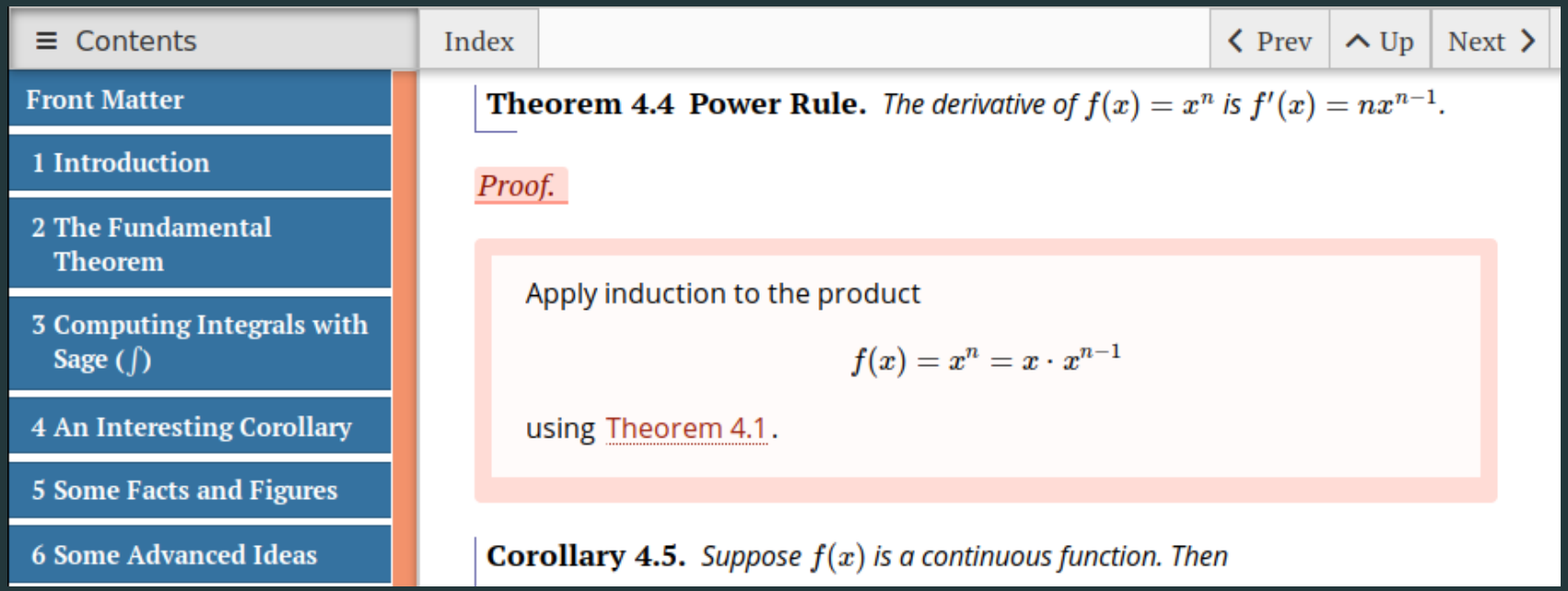
using Theorem 4.1. □

Theorem 4.4 (Power Rule). *The derivative of* $f(x) = x^n$ *is* $f'(x) = nx^{n-1}$.

Proof. Apply induction to the product

$$f(x) = x^n = x \cdot x^{n-1}$$

using Theorem 4.1. □



Doctesting Sage Examples

One of roughly 1,000 tests performed every 6 months:

```
~~~~~~~~~~~~~~~~~~~~~~ ::
    sage: G = DihedralGroup(8)
    sage: N = G.normal_subgroups()
    sage: [H.order() for H in N]
    [1, 2, 4, 8, 8, 8, 16]
```

DEMO: KARL-DIETER CRISMAN'S

NUMBER THEORY IN CONTEXT

DEMO: RESOURCE LINKING

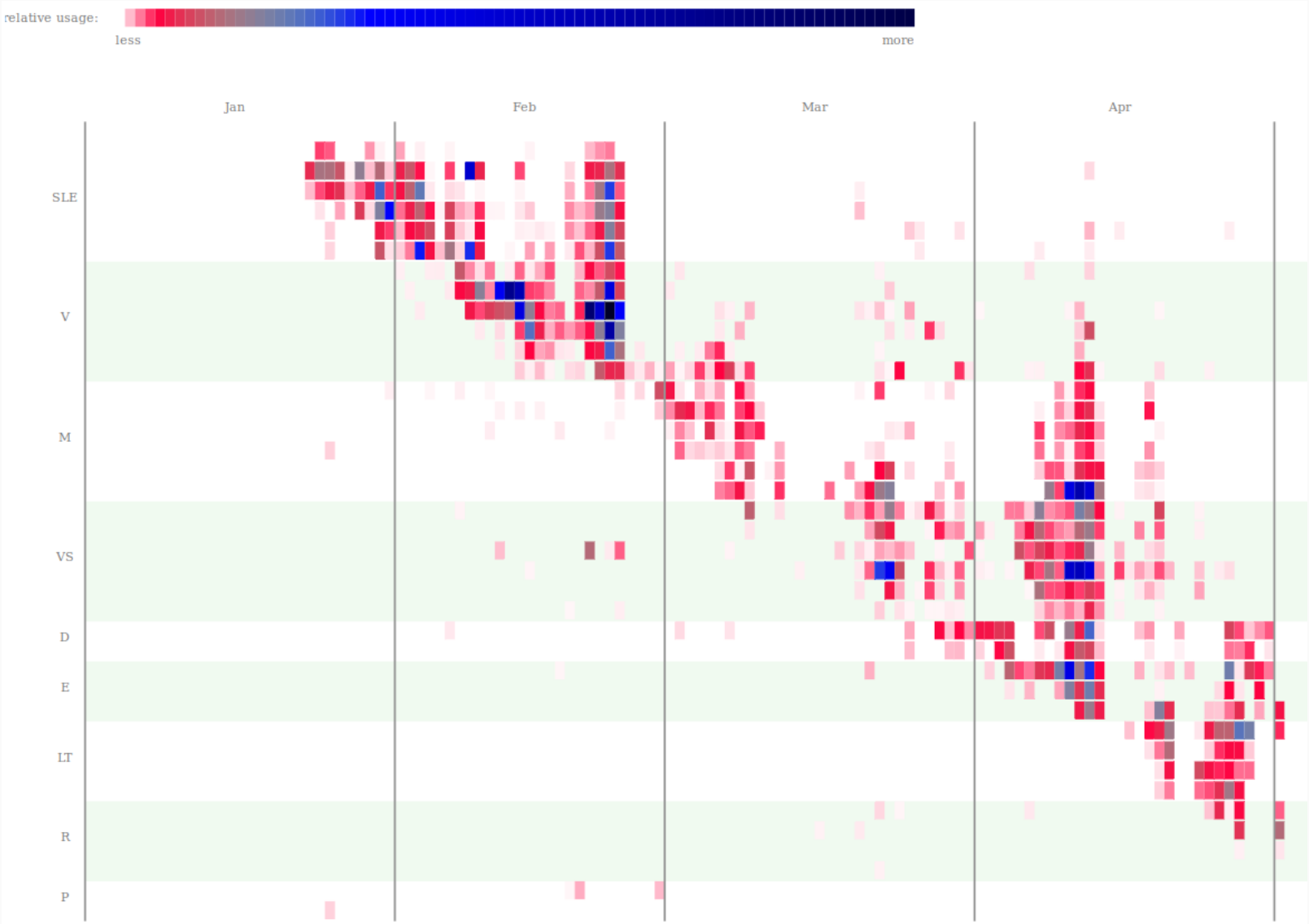
DEMO: WeBWorK PROBLEMS

DEMO: JUPYTER NOTEBOOK CONVERSION

Analytics

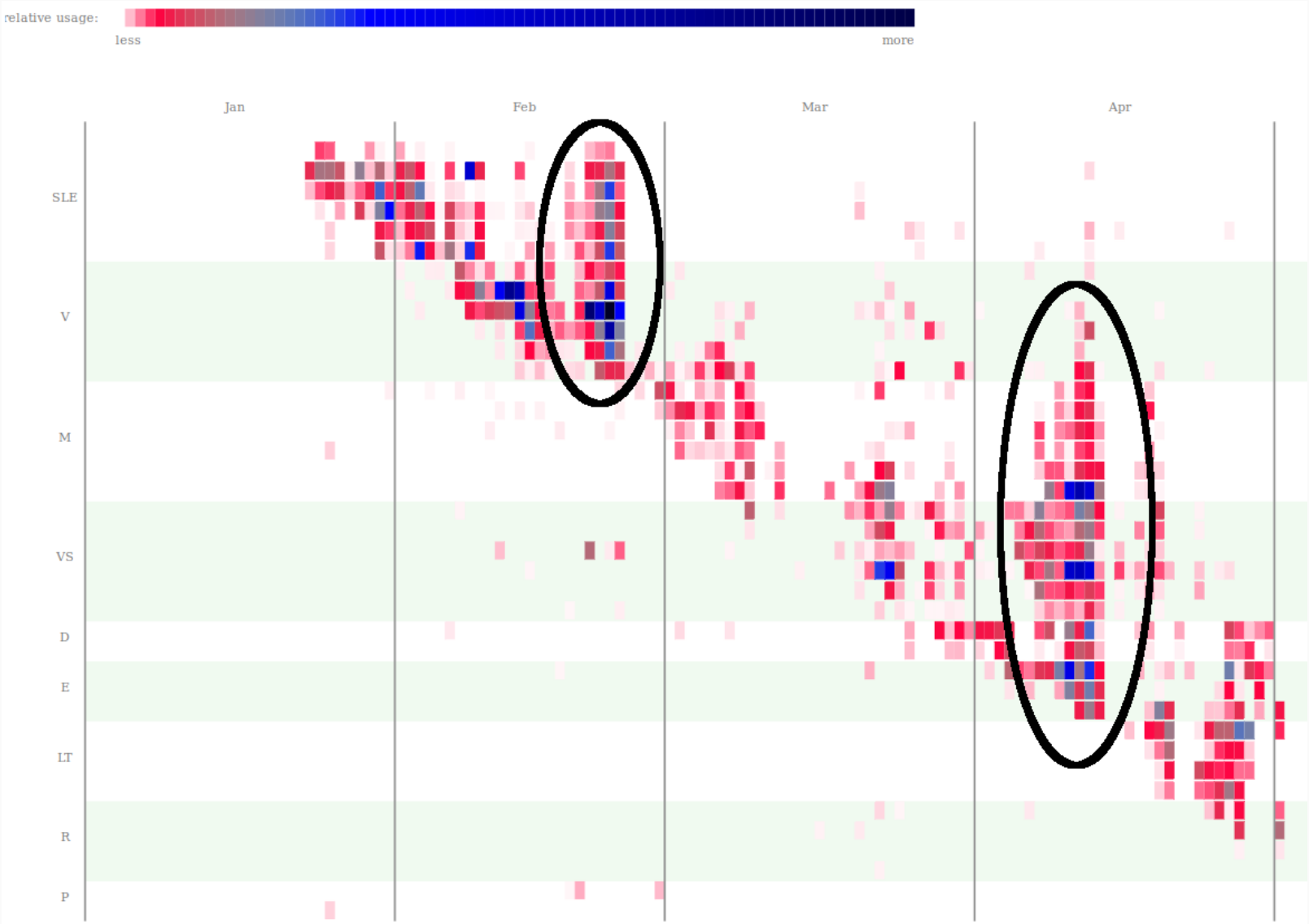
All Students for Entire Semester

Rows are sections, columns are days



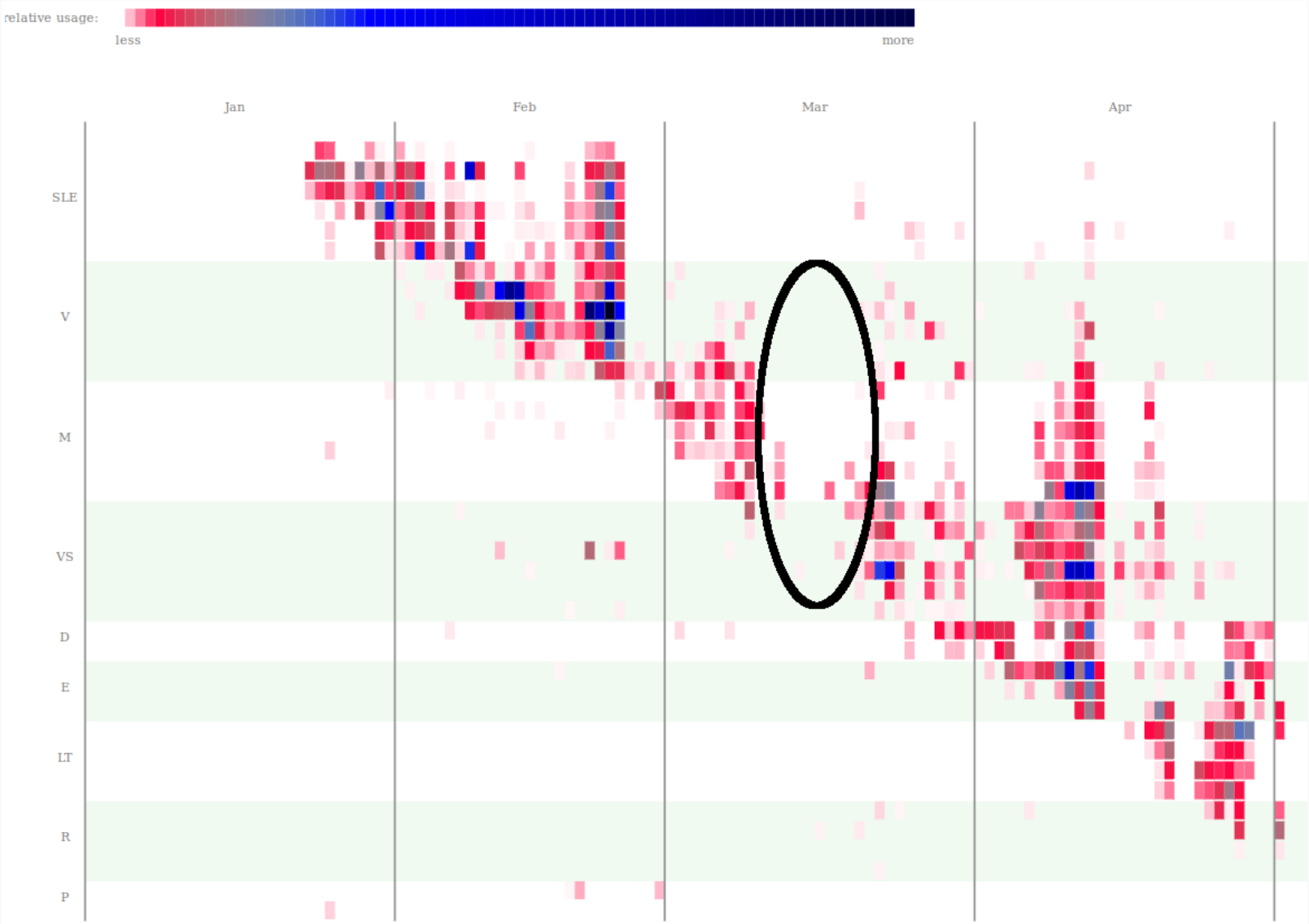
Exams

Rows are sections, columns are days



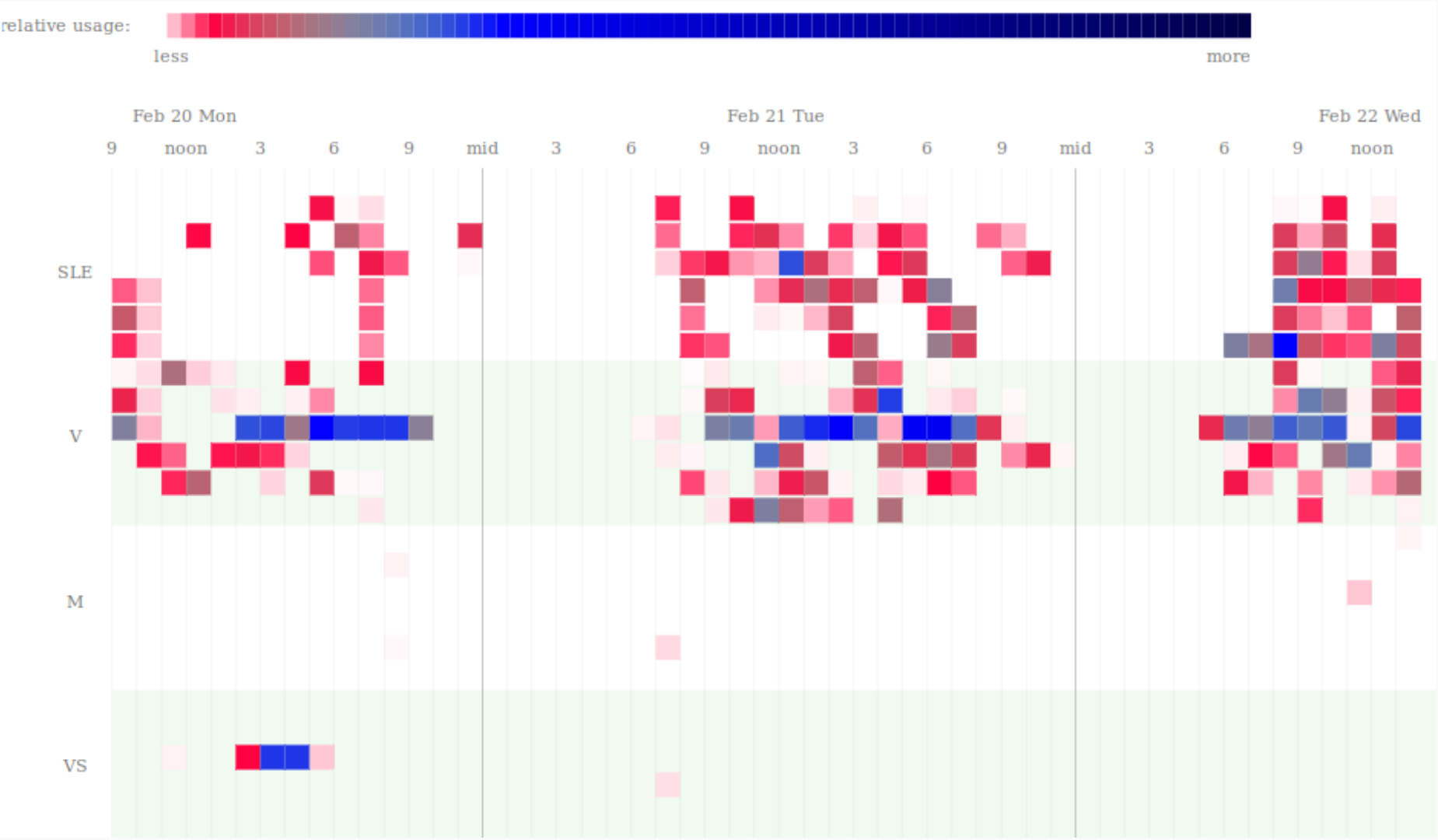
Spring Break!

Rows are sections, columns are days



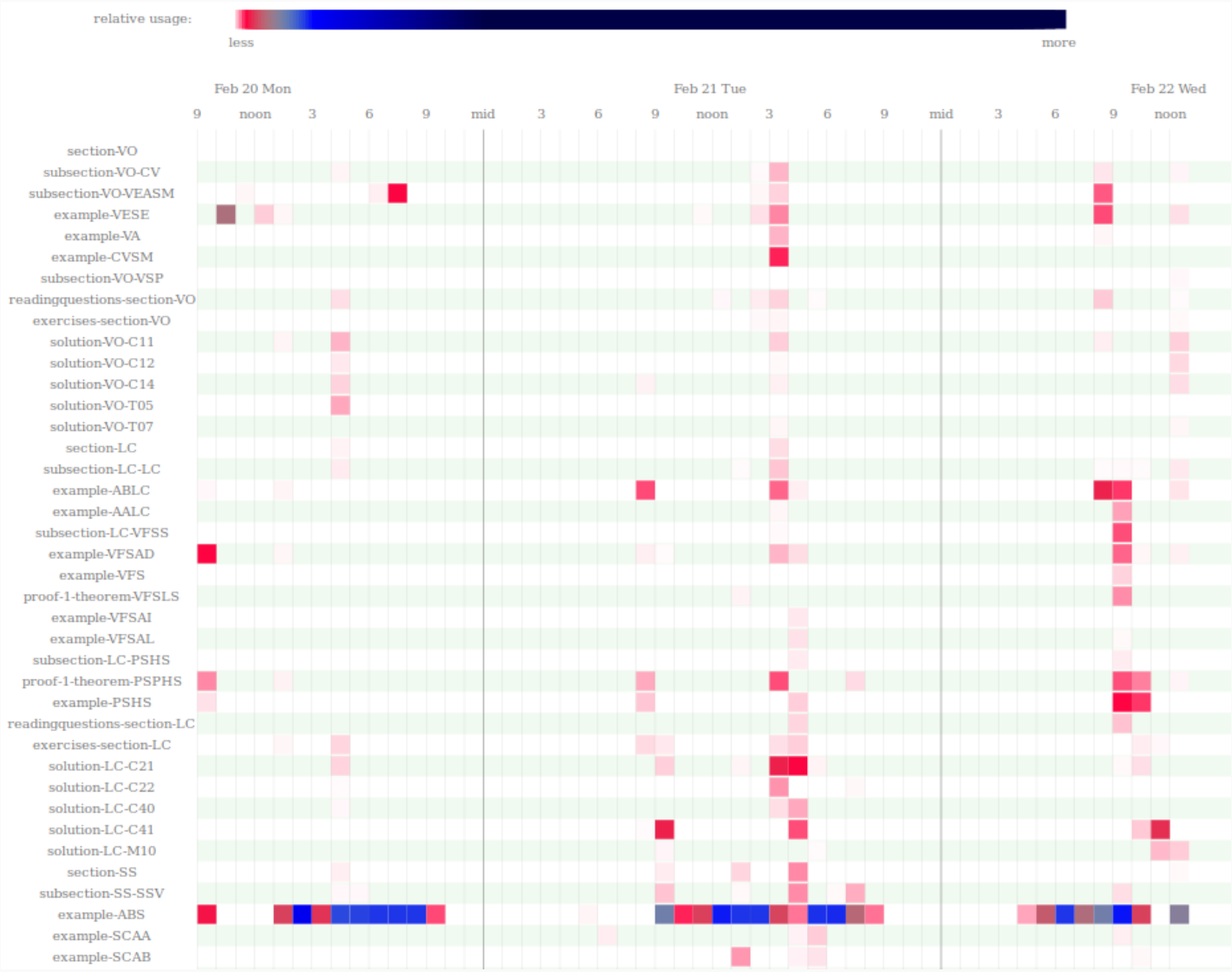
All Students on Tuesday, February 21, 2017

Rows are sections, columns are hours



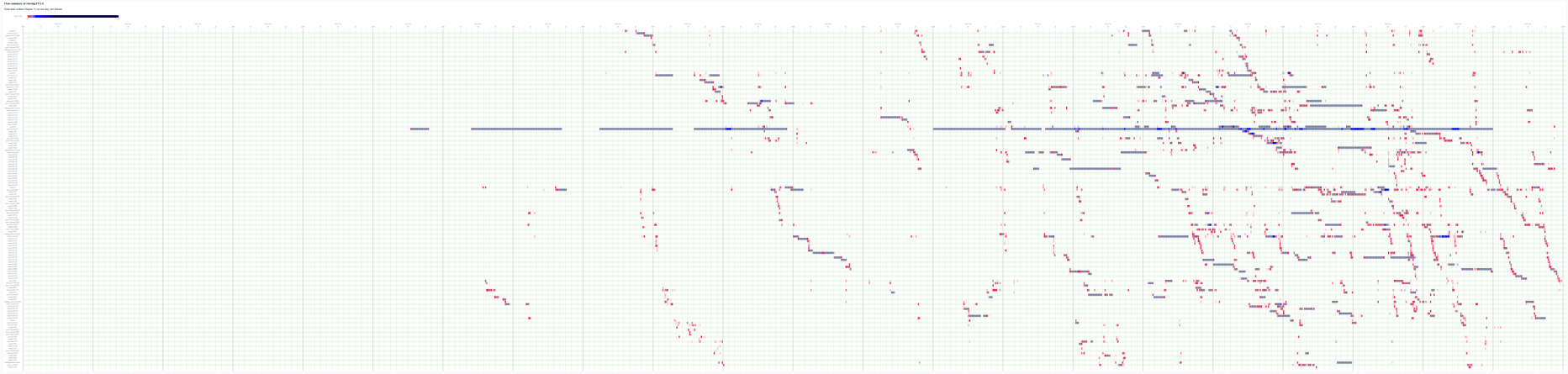
Detail: All Students on Tuesday, February 21, 2017

Rows are components (first third shown), columns are hours



All Students on Tuesday, February 21, 2017, by the Minute

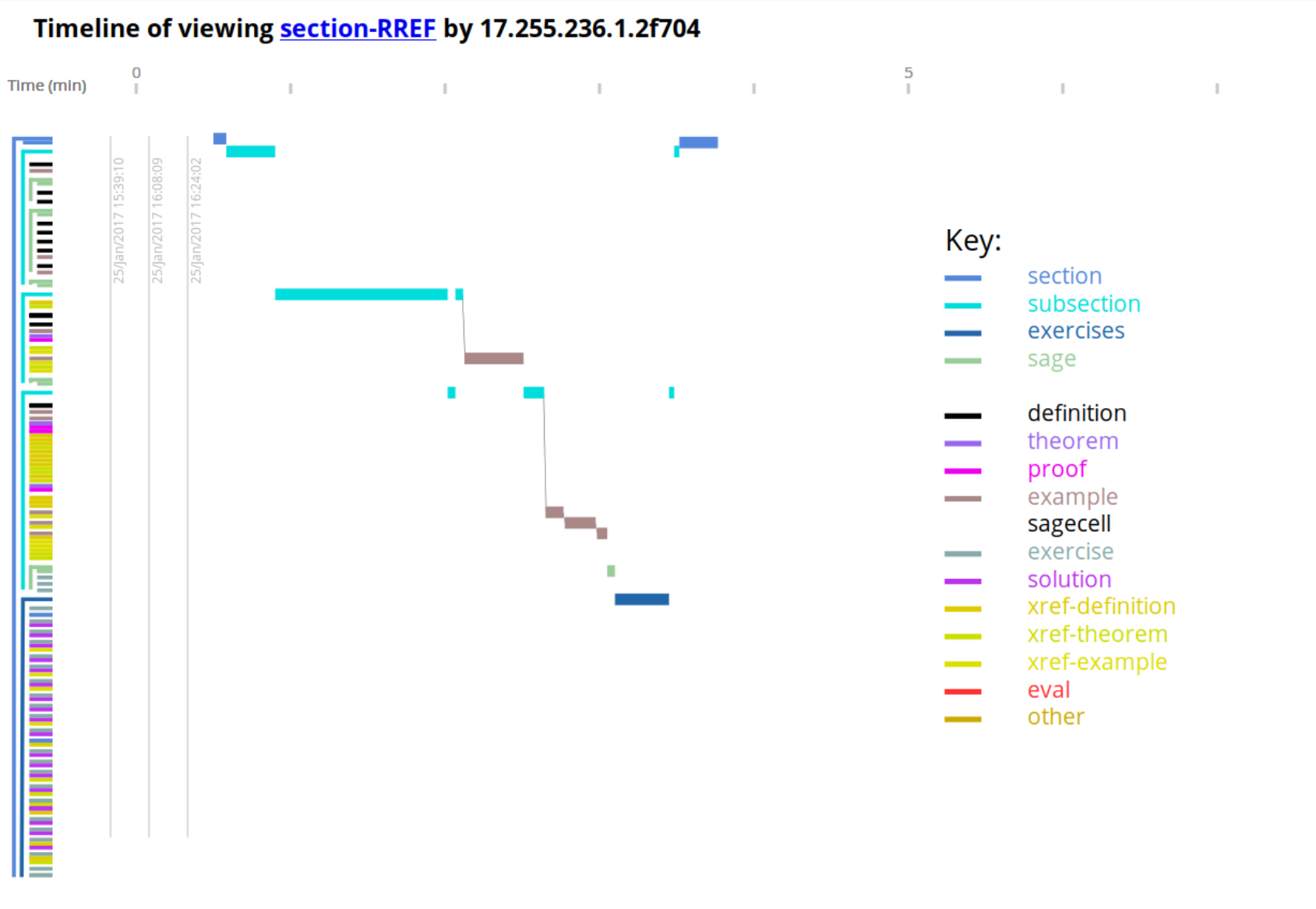
Rows are components, columns are minutes



- "Tracks" move down through material, to the right in time
- Denser in the evening — no surprise there

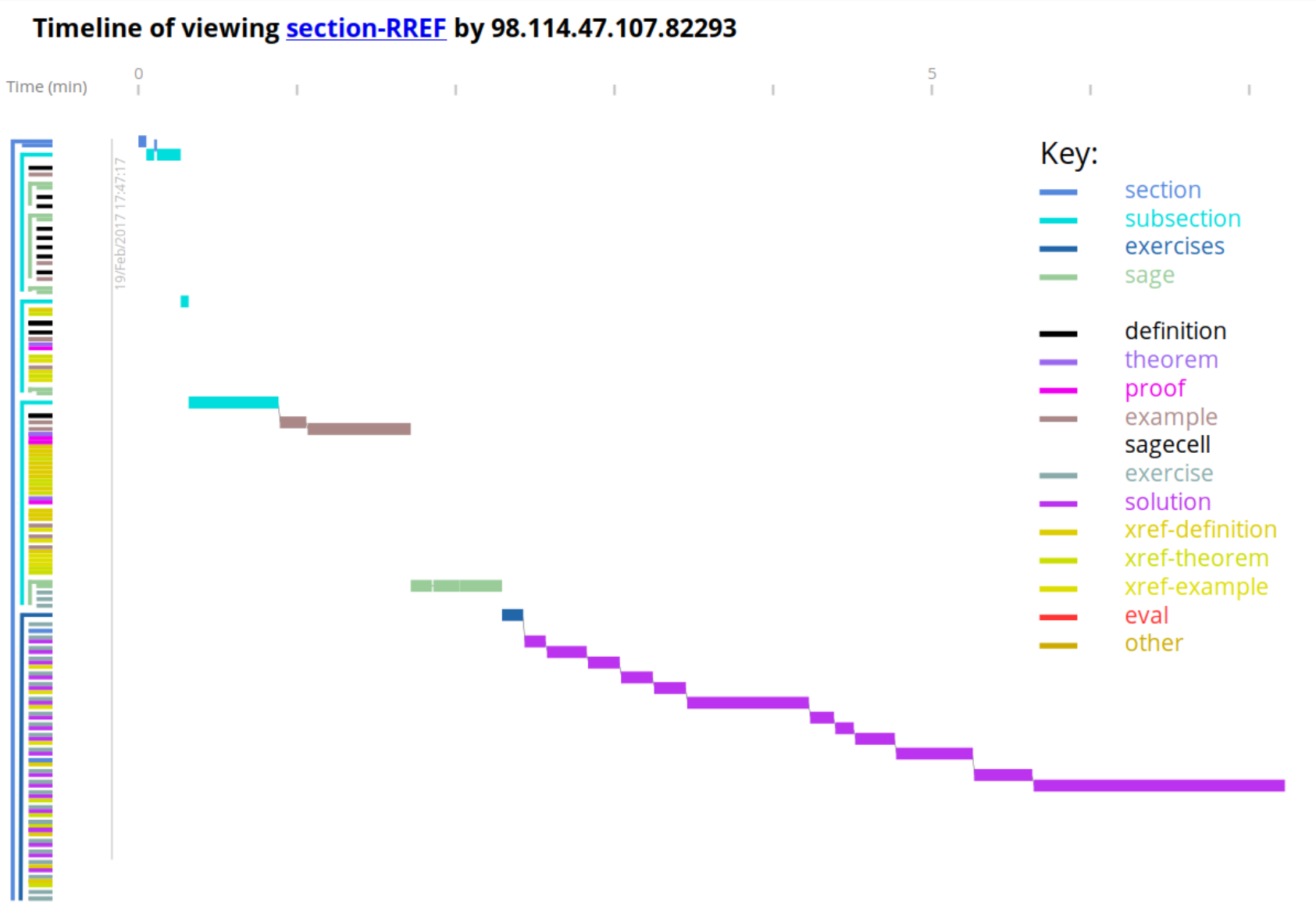
I Like Examples

One student, four minutes, four examples



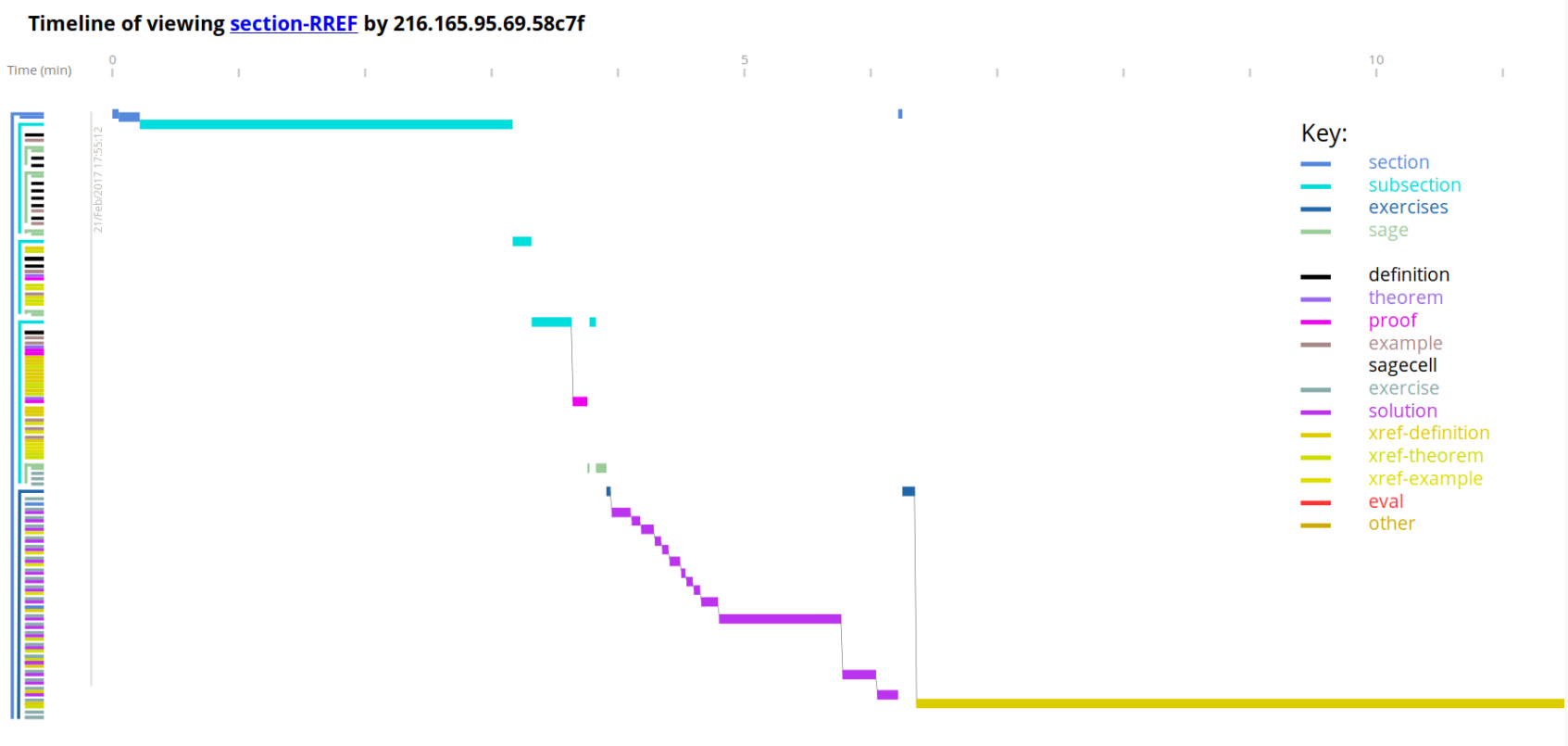
I Like Solutions to Exercises

One student, seven minutes, two examples, twelve solutions



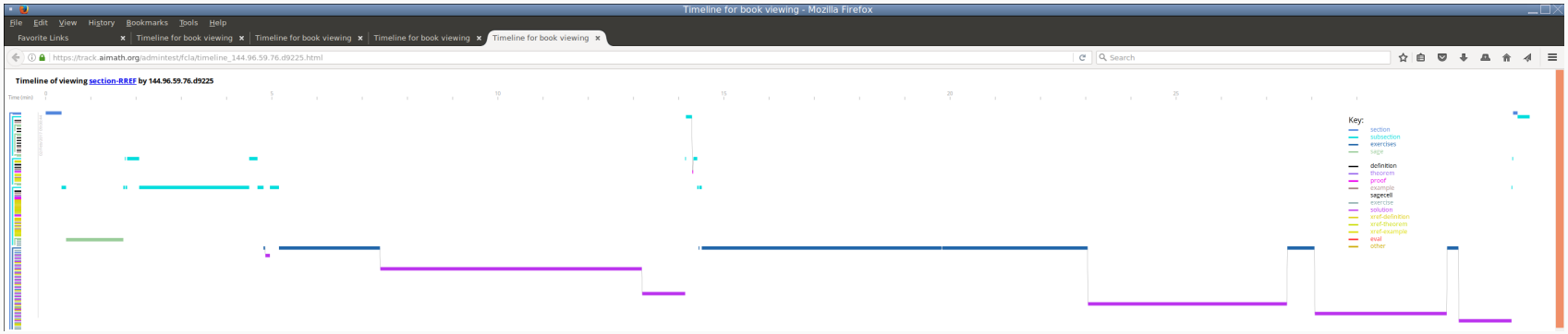
I Like Solutions Too

One student, seven minutes plus, first subsection, thirteen solutions, and cross-referenced an earlier definition as a knowl



I Like To Study

One student, half an hour, Reading Questions, third subsection, then alternately exercises and solutions



- Data display in web browser
- Tooltips on bars identify content
- Bars are links to actual component of the book
- We can match classroom sessions to student activity

Where Are We Going?

Sage's Success

- Open Source – inspect and fix code

Sage's Success

- Open Source – inspect and fix code
- Python – mainstream language

Sage's Success

- Open Source – inspect and fix code
- Python – mainstream language
- Development-Ready – tools in the distribution

Sage's Success

- Open Source – inspect and fix code
- Python – mainstream language
- Development-Ready – tools in the distribution
- Doctesting – rigorous quality control

Sage's Success

- Open Source – inspect and fix code
- Python – mainstream language
- Development-Ready – tools in the distribution
- Doctesting – rigorous quality control

NB: this list does not include "free" (as in "no-cost")

Improvements: Documentation

- Critical for education: both students and faculty
- Much exists, some is very good
  (e.g. Greg Bard's "Sage for Undergraduates")
- Some is out-of-date

  My linear algebra "quickref" (December 2011)
  My group theory "thematic tutorial" (March 2010)
- Less technical ("Adds `self` to `other`".)
- Better error messages ("`left` is not compatible with `right`".)

Improvements: Interacts

- Extremely useful for education
- Smoother, more responsive (more Javascript?)
- Multiple implementations?

Improvements: Doctesting

- Unit tests are very useful

  TEX: `TRIP.TEX`

  PreTeXt: the sample article (aka "the kitchen sink")

  Sage: doctesting (dual-purpose)

Improvements: Doctesting

- Unit tests are very useful

  TEX: `TRIP.TEX`

  PreTeXt: the sample article (aka "the kitchen sink")

  Sage: doctesting (dual-purpose)

- Doctest graphical output? (Hard?)
- Doctest interacts? (Similarly hard?)
- Doctest Sage Code from textbooks?

  Who is responsible for changes?

  When do they happen?

Lessons Learned

- Crypto: DO NOT invent your own encryption function
- Markup: DO NOT invent your own markup syntax

Lessons Learned

- Crypto: DO NOT invent your own encryption function
- Markup: DO NOT invent your own markup syntax
- Testing examples is critical

Sustainability

- Sage Cell Server: no login required
- But *somebody* needs to pay for it!

Credits

PreTeXt Developers: David Farmer, Alex Jordan, RAB

`mathbook.pugetsound.edu`

Undergraduate Teaching in Mathematics with
Open Software and Textbooks

`utmost.aimath.org`

Partial support for this work was provided by the National Science Foundation's Improving Undergraduate STEM Education (IUSE) program under Award No. 1626455. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.