Lab 6: Analyzing Precedent in the Supreme Court

Objectives:
Use and sort data in dictionaries and tuples
Gain experience with plotting
Lab Overview

• We will analyzing Supreme Court data and impact of cases
• Based on Fowler and Jeon's interesting analysis of the 30,288 majority US Supreme Court decisions on dockets through 2002
• We will plot impact of cases in chronological order
Lab Overview

• We will analyzing Supreme Court data and impact of cases
• Based on Fowler and Jeon's interesting analysis of the 30,288 majority US Supreme Court decisions on dockets through 2002
• We will plot impact of cases in chronological order
Background: h-Index
**h-Index**

- “The **h-index** is an author-level metric that attempts to measure both the **productivity** and **citation impact** of the **publications** of a **scientist** or **scholar**.” —Wikipedia
- **Definition**: The maximum $h$ where the top $h$ papers have been cited at least $h$ times.

h-index from a plot of decreasing citations for numbered papers, source: https://en.wikipedia.org/wiki/H-index
**h-Index: Examples**

- Citation counts = \( (0, 2, 15, 9, 7, 48, 4, 82, 14, 6) \)
- Sorted citation counts = \( (82, 48, 15, 14, 9, 7, 6, 3, 2, 0) \)
- First 6 papers have at least 6 cites, seventh paper has less than 7 cites: **h-index is 6**

h-index from a plot of decreasing citations for numbered papers, source: https://en.wikipedia.org/wiki/H-index
**h-Index: Algorithmic Idea**

- Sort citation count sequence in descending order (use `reverse=True` in `sorted`)
- Start with estimate  \( h = 0 \)
- Iterate through the sorted citation counts

<table>
<thead>
<tr>
<th>20</th>
<th>11</th>
<th>7</th>
<th>3</th>
<th>3</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
</tbody>
</table>

\[ h \]
**h-Index: Algorithmic Idea**

- Sort citation count sequence in descending order (use `reverse=True` in `sorted`)
- Start with estimate $h = 0$
- Iterate through the sorted citation counts

![Sorted Citation Counts](image)

$h$-Index: Algorithmic Idea

<table>
<thead>
<tr>
<th>20</th>
<th>11</th>
<th>7</th>
<th>3</th>
<th>3</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
</tbody>
</table>

`sortedCites`
**h-Index: Algorithmic Idea**

- Sort citation count sequence in descending order (use `reverse=True` in `sorted`)
- Start with estimate \( h = 0 \)
- Iterate through the sorted citation counts

\[
\begin{array}{cccccccc}
20 & 11 & 7 & 3 & 3 & 1 & 1 & 0 & 0 \\
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
\end{array}
\text{sortedCites}
\]
**h-Index: Algorithmic Idea**

- Sort citation count sequence in descending order (use `reverse=True` in `sorted`)
- Start with estimate \( h = 0 \)
- Iterate through the sorted citation counts

<table>
<thead>
<tr>
<th>20</th>
<th>11</th>
<th>7</th>
<th>3</th>
<th>3</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
</tbody>
</table>

\[ \rightarrow h \]
**h-Index: Algorithmic Idea**

- Sort citation count sequence in descending order (use `reverse=True` in `sorted`)
- Start with estimate  $h = 0$
- Iterate through the sorted citation counts

<table>
<thead>
<tr>
<th>20</th>
<th>11</th>
<th>7</th>
<th>3</th>
<th>3</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
</table>

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

- If the `sortedCites[h]` has more than $h$ citations
  - We know the $h$-Index is at least $h + 1$, so we can increment our estimate and keep iterating
- Else, we have found our $h$-Index which is $h$
Lab tasks

• Read data using csv.reader()
  • “Fancy” file reader for more robust interpretation of CSV data
  • Correctly handles commas and punctuation that may exist in data aside from column separators
• Build dictionary
  • Map years (int) to tuples of citation counts (ints)
• Calculate the h-index
  • For a given year, calculate h-index of tuples of citation counts
• Plot results
  • Use matplotlib as described in lab handout and in class
  • (See note about installing matplotlib)
Acknowledgments

These slides have been adapted from:

- [http://cs111.wellesley.edu/spring19](http://cs111.wellesley.edu/spring19) and
- [https://www.python-course.eu/python3_object_oriented_programming.php](https://www.python-course.eu/python3_object_oriented_programming.php)