Lecture 10

Groups
Announcements

- Project 1 out this Friday
- Go to section this week to find a partner!
Defining Functions
User-defined functions give names to blocks of code

```python
def spread(values):
    return max(values) - min(values)
```

(Demo)
Apply
The `apply` method creates an array by calling a function on every element in input column(s)

- First argument: Function to apply
- Other arguments: The input column(s)

```
table_name.apply(function_name, 'column_label')
```

(Demo)
Apply

The *apply* method creates an array by calling a function on every element in one or more input columns

- First argument: Function to apply
- Other arguments: The input column(s)

```python
table_name.apply(one_arg_function, 'column_label')
```

```python
table_name.apply(two_arg_function,
                  'column_label_for_first_arg',
                  'column_label_for_second_arg')
```

*apply* called with only a function applies it to each row

(Demo)
Example: Prediction
Grouping Rows
The **group** method aggregates all rows with the same value for a column into a single row in the result.

- **First argument:** Which column to group by.
- **Second argument:** (Optional) How to combine values:
  - `len` — number of grouped values (default)
  - `sum` — total of all grouped values
  - `list` — list of all grouped values

(Demo)
The **groups** method can also aggregate all rows that share the combination of values in multiple columns

- **First argument:** A list of which columns to group by
- **Second argument:** (Optional) How to combine values

(Demo)
**Challenge Question**

Which NBA teams spent the most on their starters in 2016?

- Each team has one *starter* per position
- Assume the starter for a team & position is the player with the highest salary on that team in that position

<table>
<thead>
<tr>
<th>team</th>
<th>player</th>
<th>position</th>
<th>salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlanta Hawks</td>
<td>Brandon Goodwin</td>
<td>PG</td>
<td>0.522738</td>
</tr>
<tr>
<td>Atlanta Hawks</td>
<td>Bruno Fernando</td>
<td>C</td>
<td>1.4</td>
</tr>
<tr>
<td>Atlanta Hawks</td>
<td>Cam Reddish</td>
<td>SG</td>
<td>4.24572</td>
</tr>
<tr>
<td>Atlanta Hawks</td>
<td>Chandler Parsons</td>
<td>SF</td>
<td>25.1025</td>
</tr>
<tr>
<td>Atlanta Hawks</td>
<td>Damian Jones</td>
<td>C</td>
<td>2.30506</td>
</tr>
<tr>
<td>Atlanta Hawks</td>
<td>DeAndre’ Bembry</td>
<td>SG</td>
<td>2.60398</td>
</tr>
</tbody>
</table>
Pivot Tables
Pivot

- Cross-classifies according to two categorical variables
- Produces a grid of counts or aggregated values
- Two required arguments:
  - First: variable that forms column labels of grid
  - Second: variable that forms row labels of grid
- Two optional arguments (include both or neither)
  - `values='column_label_to_aggregate'`
  - `collect=function_with_which_to_aggregate`

(Demo)
The topic giving me the most trouble...

Tables, table manipulation

Arrays

Histograms

Functions/Apply

Other

I'm good, thanks