Plotting Nathan’s Hot Dog Contest Results

Dr. Tiffani L. Williams

Department of Computer Science and Engineering
Texas A&M University

Fall 2013
The Nathan’s Hot Dog Eating Contest is an annual American competitive-eating competition held on July 4th at Nathan’s Famous restaurant in Coney Island.

Since it is a sporting event, ESPN has televised the competition live since 2004.

Besides free hot dogs and Pepto Bismal, the competitors can also win cash prizes.

- First Place: $10,000
- Second Place: $5,000
- Third Place: $2,500
- Fourth Place: $1,500
- Fifth Place: $1,000

In sum, a disgusting contest that is a fun way to celebrate Independence Day in the United States. 😊
Motivation

- All data that we will use in our plots is available from Wikipedia.
- Also, there are links in the slides for obtaining the data from Nathan Yau, the author who wrote an interesting book called "Visualize This: The FlowingData Guide to Design, Visualization, and Statistics."
- In Yau’s book, he presented a neat visualization example using Nathan’s Hot Dog Eating Contest data. His work provided the inspiration to see what kinds of plots we can make in Python.
- Besides, plotting data from a disgusting contest is fun. 😊
Here's our data. I

(Download the hot dog contest data.)

<table>
<thead>
<tr>
<th></th>
<th>Year</th>
<th>Winner</th>
<th>Hot dogs eaten</th>
<th>Country</th>
<th>Contest duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1980</td>
<td>Paul Siederman &amp; Joe Baldini</td>
<td>9.1</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>1981</td>
<td>Thomas DeBerry</td>
<td>11</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>1982</td>
<td>Steven Abrams</td>
<td>11</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>1983</td>
<td>Luis Llamas</td>
<td>19.5</td>
<td>Mexico</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>1984</td>
<td>Birgit Felden</td>
<td>9.5</td>
<td>Germany</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>1985</td>
<td>Oscar Rodriguez</td>
<td>11.75</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>7</td>
<td>1986</td>
<td>Mark Heller</td>
<td>15.5</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>8</td>
<td>1987</td>
<td>Don Wolfman</td>
<td>12</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>9</td>
<td>1988</td>
<td>Jay Green</td>
<td>14</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>10</td>
<td>1989</td>
<td>Jay Green</td>
<td>13</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>11</td>
<td>1990</td>
<td>Mike DeVito</td>
<td>16</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>1991</td>
<td>Frank Dellarosa</td>
<td>21.5</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>1992</td>
<td>Frank Dellarosa</td>
<td>19</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>14</td>
<td>1993</td>
<td>Mike DeVito</td>
<td>17</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>15</td>
<td>1994</td>
<td>Mike DeVito</td>
<td>20</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>16</td>
<td>1995</td>
<td>Edward Krachie</td>
<td>19.5</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>17</td>
<td>1996</td>
<td>Edward Krachie</td>
<td>22.25</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>18</td>
<td>1997</td>
<td>Hirofumi Nakajima</td>
<td>24.5</td>
<td>Japan</td>
<td>12</td>
</tr>
<tr>
<td>19</td>
<td>1998</td>
<td>Hirofumi Nakajima</td>
<td>19</td>
<td>Japan</td>
<td>12</td>
</tr>
<tr>
<td>20</td>
<td>1999</td>
<td>Steve Keiner</td>
<td>20.25</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>21</td>
<td>2000</td>
<td>Kazutoyo Arai</td>
<td>25.13</td>
<td>Japan</td>
<td>12</td>
</tr>
<tr>
<td>22</td>
<td>2001</td>
<td>Takeru Kobayashi</td>
<td>50</td>
<td>Japan</td>
<td>12</td>
</tr>
<tr>
<td>23</td>
<td>2002</td>
<td>Takeru Kobayashi</td>
<td>50.5</td>
<td>Japan</td>
<td>12</td>
</tr>
</tbody>
</table>
Here's our data:

<table>
<thead>
<tr>
<th>Year</th>
<th>Name</th>
<th>Weight</th>
<th>Country</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>2003</td>
<td>Takeru Kobayashi</td>
<td>44.5</td>
<td>Japan</td>
<td>12</td>
</tr>
<tr>
<td>2004</td>
<td>Takeru Kobayashi</td>
<td>53.5</td>
<td>Japan</td>
<td>12</td>
</tr>
<tr>
<td>2005</td>
<td>Takeru Kobayashi</td>
<td>49</td>
<td>Japan</td>
<td>12</td>
</tr>
<tr>
<td>2006</td>
<td>Takeru Kobayashi</td>
<td>53.75</td>
<td>Japan</td>
<td>12</td>
</tr>
<tr>
<td>2007</td>
<td>Joey Chestnut</td>
<td>66</td>
<td>United States</td>
<td>12</td>
</tr>
<tr>
<td>2008</td>
<td>Joey Chestnut</td>
<td>59</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>2009</td>
<td>Joey Chestnut</td>
<td>68</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>2010</td>
<td>Joey Chestnut</td>
<td>54</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>2011</td>
<td>Joey Chestnut</td>
<td>62</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>2012</td>
<td>Joey Chestnut</td>
<td>68</td>
<td>United States</td>
<td>10</td>
</tr>
<tr>
<td>2013</td>
<td>Joey Chestnut</td>
<td>69</td>
<td>United States</td>
<td>10</td>
</tr>
</tbody>
</table>
A line plot

(Download the Python code to make a line plot.)
Listing 1: plot-nathans-hotdog-results-lineplot.py

```python
# Making a line plot for the results from Nathan’s Hot Dog Contest.

import matplotlib.pyplot as plot

def make_line_plot(years, hotdogs_eaten):
    # create canvas for figure
    fig = plot.figure(figsize=(15,5))

    # setup initial labels and width of bar plot
    x_label_pos = range(len(years))

    # create bar plot
    plot.plot(x_label_pos, hotdogs_eaten, color='blue', marker='o')

    # add formatting to bar plot
    plot.title("Nathan’s Hotdog Contest Results")
    plot.xlabel("Year")
    plot.ylabel("Number of hotdogs eaten")
    plot.grid(True)

    plot.xticks(x_label_pos, years)

    # scale axes
    plot.yticks(range(0,85,10))  # manual scale of y-axis
    plot.autoscale(enable=True, axis='x', tight=True)  # autoscale of x-axis

    # A nice function to auto-rotate the x axis labels.
    # It was made for dates (hence the name), but it works
    # for any long x tick labels
    fig.autofmt_xdate()
```

# save and show plot
plot.savefig("nathans-hotdog-first-place-results-lineplot.pdf")
plot.show()

def main():
    input_file = open("nathans-hotdog-data.csv")

    # ignore CSV header line
    input_file.readline()

    # read rest of CSV file
    lines = input_file.readlines()

    year = []
    hotdogs = []
    for line in lines:
        split_line = line.strip().split(',','

        # add elements of interest from the line to the appropriate lists
        year += [split_line[0]]
        hotdogs += [float(split_line[2])]

    make_line_plot(year, hotdogs)

main()
A bar plot

(Download the Python code to make a bar plot.)
# Making a bar plot for showing the results from Nathan’s Hot Dog Contest.

```python
import matplotlib.pyplot as plot

def make_bar_plot(years, hotdogs_eaten):
    # create canvas for figure
    fig = plot.figure(figsize=(15,5))

    # setup initial labels and width of bar plot
    x_label_pos = range(len(years))
    width = 0.7

    # create bar plot
    plot.bar(x_label_pos, hotdogs_eaten, width, color='blue')

    # add formatting to bar plot
    plot.title("Nathan’s Hotdog Contest Results")
    plot.xlabel("Year")
    plot.ylabel("Number of hotdogs eaten")
    plot.grid(True)

    x_label = [i + (width / 1.2) for i in x_label_pos]
    plot.xticks(x_label, years)

    # scale axes
    plot.autoscale(enable=True, axis='both', tight=True)

    # A nice function to auto-rotate the x axis labels.
    # It was made for dates (hence the name), but it works
    # for any long x tick labels
```

fig.autofmt_xdate()

# save and show plot
plot.savefig("nathans-hotdog-first-place-results-barplot.pdf")
plot.show()

def main():
    input_file = open("nathans-hotdog-data.csv")

    # ignore CSV header line
    input_file.readline()

    # read rest of CSV file
    lines = input_file.readlines()
    year = []
    hotdogs = []
    for line in lines:
        split_line = line.strip().split(',

        # add elements of interest from the line to the appropriate lists
        year += [split_line[0]]
        hotdogs += [float(split_line[2])]

    make_bar_plot(year, hotdogs)

main()
Data that shows the number of hot dogs eaten by first, second, and third place winners.

(Download the data for the first, second, and third place winners.)

2  25, 50, 50.5, 44.5, 53.5, 49, 54, 66, 59, 68, 54, 62, 68, 69
3  24, 31, 26, 30.5, 38, 37, 52, 63, 59, 64.5, 43, 53, 52, 51
4  22, 23.5, 25.5, 29.5, 32, 32, 37, 49, 42, 55, 37, 45, 51, 50
A bar plot with bars side-by-side

(Download the Python code to make a bar plot.)
Listing 3: plot-nathans-hotdog-places-barplot.py

```python
''' Making a bar plot of the first, second, and third place results from Nathan’s Hot Dog Contest. '''

import matplotlib.pyplot as plot

def make_bar_plot(years, first, second, third):
    # create canvas for figure
    fig = plot.figure(figsize=(15,6))

    # setup initial labels and width of bar plot
    x_label_pos_orig = range(len(years))
    width = 0.2

    # plot first place bars
    plot.bar(x_label_pos_orig, first, width, label = 'first place', color='blue')

    # plot second place bars + width from first place bars
    x_label_pos = [i + width for i in x_label_pos_orig]
    plot.bar(x_label_pos, second, width, label = 'second place', color='red')

    # plot third place bars + 2 times width from first place bars
    x_label_pos = [i + (2 * width) for i in x_label_pos_orig]
    plot.bar(x_label_pos, third, width, label = 'third place', color='white')

    # add formatting to bar plot
    plot.legend(loc='upper left')
    plot.title("Nathan’s Hotdog Contest Results")
    plot.xlabel("Year")
    plot.ylabel("Number of hotdogs eaten")
    x_label_pos = [i + (1.6 * width) for i in x_label_pos_orig]
```
plot.xaxis(plt.xticks(x_label_pos, years)
plot.yaxis(plt.xticks(range(0,75, 10))
plot.grid(True)

# save and show plot
plot.savefig("nathans-hotdog-places-barplot.pdf")
plot.show()

def strip_and_split_string(csv_string):
    return csv_string.strip().split(',','

def convert_to_list_of_floats(csv_string):
    values = strip_and_split_string(csv_string)
    new_list = []
    for i in values:
        new_list += [float(i)]
    return new_list

def main():
    input_file = open("nathans-hotdog-places.csv")

    # process lines of CSV file
    years = strip_and_split_string(input_file.readline())
    first = convert_to_list_of_floats(input_file.readline())
    second = convert_to_list_of_floats(input_file.readline())
    third = convert_to_list_of_floats(input_file.readline())

    make_bar_plot(years, first, second, third)

main()
Nathan's Hotdog Contest Results

- **first place**
- **second place**
- **third place**


Number of hotdogs eaten: 0, 10, 20, 30, 40, 50, 60, 70

- **2000**
  - First place: 30
  - Second place: 20
  - Third place: 10

- **2001**
  - First place: 40
  - Second place: 30
  - Third place: 20

- **2002**
  - First place: 50
  - Second place: 40
  - Third place: 30

- **2003**
  - First place: 60
  - Second place: 50
  - Third place: 40

- **2004**
  - First place: 70
  - Second place: 60
  - Third place: 50

- **2005**
  - First place: 80
  - Second place: 70
  - Third place: 60

- **2006**
  - First place: 90
  - Second place: 80
  - Third place: 70

- **2007**
  - First place: 100
  - Second place: 90
  - Third place: 80

- **2008**
  - First place: 110
  - Second place: 100
  - Third place: 90

- **2009**
  - First place: 120
  - Second place: 110
  - Third place: 100

- **2010**
  - First place: 130
  - Second place: 120
  - Third place: 110

- **2011**
  - First place: 140
  - Second place: 130
  - Third place: 120

- **2012**
  - First place: 150
  - Second place: 140
  - Third place: 130

- **2013**
  - First place: 160
  - Second place: 150
  - Third place: 140
A stacked bar plot

(Download the Python code to make a stacked bar plot.)
# Making a bar plot of the first, second, and third place results from
# Nathan's Hot Dog Contest.

```python
import matplotlib.pyplot as plot

def make_bar_plot(years, first, second, third):
    # create canvas for figure
    fig = plot.figure(figsize=(15, 6))

    # setup initial labels and width of bar plot
    x_label_pos = range(len(years))
    width = 0.75

    # create stacked bar plot
    first_and_second = [first[i] + second[i] for i in range(len(first))]
    p1 = plot.bar(x_label_pos, first, width, label='first place',
                   color='blue')
    p2 = plot.bar(x_label_pos, second, width, label='second place',
                   color='red', bottom=first)
    p3 = plot.bar(x_label_pos, third, width, label='third place',
                   color='white', bottom=first_and_second)

    ''' Customizing the plot curves/bars to be displayed in the legend
    (i.e., their order of appearance). If you use this method,
    you must provide the labels for the plot curves/bars.
    '''
    plot.legend([p3[0], p2[0], p1[0]],
                 ['third place', 'second place', 'first place'], loc='upper left')

    # remainder of plot formatting
    plot.title("Nathan's Hotdog Contest Results")
```
plot.xlabel("Year")
plot.ylabel("Number of hotdogs eaten")
x_label = [i + (width / 2) for i in x_label_pos]
plot.xticks(x_label, years)
plot.yticks(range(0,210, 25))
plot.grid(True)

# save and show plot
plot.savefig("nathans-hotdog-places-stacked-barplot.pdf")
plot.show()

def strip_and_split_string(csv_string):
    return csv_string.strip().split(',' , )

def convert_to_list_of_floats(csv_string):
    values = strip_and_split_string(csv_string)
    new_list = []
    for i in values:
        new_list += [float(i)]
    return new_list

def main():
    input_file = open("nathans-hotdog-places.csv")

    # process lines of CSV file
    years = strip_and_split_string(input_file.readline())
    first = convert_to_list_of_floats(input_file.readline())
    second = convert_to_list_of_floats(input_file.readline())
    third = convert_to_list_of_floats(input_file.readline())

    make_bar_plot(years, first, second, third)

    main()}
Nathan's Hotdog Contest Results

Number of hotdogs eaten

Year


thick place  second place  first place