CSCE 110 — Programming I
Plotting Nathan’s Hot Dog Contest Results

Dr. Tiffani L. Williams

Department of Computer Science and Engineering
Texas A&M University

Fall 2011
Nathan’s Hot Dog Eating Contest

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

<table>
<thead>
<tr>
<th>Year</th>
<th>Name</th>
<th>Score</th>
<th>Country</th>
<th>Place</th>
</tr>
</thead>
<tbody>
<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>
</tbody>
</table>
A line plot

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

```python
import matplotlib.pyplot as plot
import numpy as np

def make_line_plot(years, hotdogs_eaten):
    # create canvas for figure
    fig = plot.figure(figsize=(10, 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)

    # x_label_pos is a numpy array. Suppose x_label_pos is array([0, 1, 2, 3]).
    # Suppose we want to add the value 0.5 to each value in the x_label_pos
    # array. We can simply write x_label_pos + 0.5 which will result in
    # array([0.5, 1.5, 2.5, 3.5]). Thus, numpy arrays give us a lot of power
    # to manipulate arrays with using for or while loops.
    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()

# close interactive plot viewer
#plot.close()

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.)
Listing 2: plot-nathans-hotdog-results-barplot.py

# Making a bar plot for showing the results from Nathan’s Hot Dog Contest.
import matplotlib.pyplot as plot
import numpy as np

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 = np.arange(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_pos is a numpy array. Suppose x_label_pos is array([0, 1, 2, 3]).
    # Suppose we want to add the value 0.5 to each value in the x_label_pos
    # array. We can simply write x_label_pos + 0.5 which will result in
    # array([0.5, 1.5, 2.5, 3.5]). Thus, numpy arrays give us a lot of power
    # to manipulate arrays with using for or while loops.
    plot.xticks(x_label_pos + (width/1.2), 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()

# close interactive plot viewer
#plot.close()

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.)

\[
\begin{array}{|l|}
\hline
2 & 25, 50, 50.5, 44.5, 53.5, 49, 54, 66, 59, 68, 54, 62 \\
3 & 24, 31, 26, 30.5, 38, 37, 52, 63, 59, 64.5, 43, 53 \\
4 & 22, 23.5, 25.5, 29.5, 32, 32, 37, 49, 42, 55, 37, 45 \\
\hline
\end{array}
\]
A stacked bar plot

(Download the Python code to make a stacked bar plot.)
Listing 3: plot-nathans-hotdog-places-stacked-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
import numpy as np

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 = np.arange(len(years))
    width = 0.75

    # create stacked bar plot
    second_half = np.array(first) + np.array(second)
    p1 = plot.bar(x_label_pos, first, width, color='blue')
    p2 = plot.bar(x_label_pos, second, width, color='red', bottom=first)
    p3 = plot.bar(x_label_pos, third, width, color='white', bottom=second_half)

    # add formatting to bar plot
    plot.legend((p3[0], p2[0], p1[0]),
                 ('third place', 'second place', 'first place'), loc='upper left')
    plot.title("Nathan's Hotdog Contest Results")
    plot.xlabel("Year")
    plot.ylabel("Number of hotdogs eaten")
    plot.xticks(x_label_pos + width/2, 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

- First place
- Second place
- Third place


Number of hotdogs eaten:
A bar plot with bars side-by-side

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

import matplotlib.pyplot as plot
import numpy as np

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 = np.arange(len(years))
    width = 0.2

    # plot first place bars
    p1 = plot.bar(x_label_pos, first, width, color='blue')

    # plot second place bars + width from first place bars
    p2 = plot.bar(x_label_pos + width, second, width, color='red')

    # plot third place bars + 2 times width from first place bars
    p3 = plot.bar((x_label_pos + (2 * width)), third, width, color='white')

    # add formatting to bar plot
    plot.legend((p3[0], p2[0], p1[0]),
                 ('third place', 'second place', 'first place'), loc='upper left')
    plot.title("Nathan’s Hotdog Contest Results")
    plot.xlabel("Year")
    plot.ylabel("Number of hotdogs eaten")
    plot.xticks(x_label_pos + (1.6 * width), years)
plot.yticks(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

Year:
- 2000
- 2001
- 2002
- 2003
- 2004
- 2005
- 2006
- 2007
- 2008
- 2009
- 2010
- 2011

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

- third place
- second place
- first place