Applying Machine Learning Prof. Daniel Björkegren

PS 2: Visualization

Please submit a single PDF or Word document of your write-up and scripts containing the code you used through Courseworks by 11 am the day before class. Do not submit console output or data files. Include the names of the people you worked with on the first page. AI code assistance *is* allowed for this assignment (you may use tools like Github Copilot or ChatGPT to write code); however, all outputs you include must result from executing your code, and results must be written up without the use of AI.

We recommend using Python and ggplot/plotnine with pandas as it provides a useful framework for making flexible graphics. You are welcome to use another language and whatever packages you would like for these visualization tasks (such as R and ggplot2), as long as it can make more than just simple standard plot types.

# Graphing Napoleon’s March, Continued (optional)[[1]](#footnote-1)

Revisit the graphs we produced in PS 1. The grammar of graphics package (ggplot) makes it easy to layer different dimensions onto the same plot. For example, the following layers the location of the armies, direction of the march, the number of survivors, and city names.

Reproduce Minard’s plot with the following code. I recommend adding each ggplot command one at a time so see the effect on the plot. Include your plot in your writeup. Feel free to alter the plot to make it more legible or informative.

import pandas as pd import plotnine as p9

troops = pd.read\_table("tables/minard-troops.txt", index\_col=0)

cities = pd.read\_table("tables/minard-cities.txt",

header='infer', sep=' ', index\_col=False, engine='python')

plot\_minard = (p9.ggplot(troops)

+ p9.aes('long', 'lat')

+ p9.geom\_path(p9.aes(size='survivors', colour='direction', group='group'),

linejoin="round", lineend="round")

+ p9.geom\_text(p9.aes(label='city'), size=8, data=(cities))

+ p9.scale\_size(range=[1, 12],

breaks=True, labels=(['100k', '200k', '300k']))

+ p9.scale\_colour\_manual(values=['grey', 'red']))

p9.ggsave(plot\_minard, filename="minard.pdf", width=12, height=3)

(If using R, also see <http://motioninsocial.com/tufte/>for examples of implementing various Tufte plots.)

# Top Pop[[2]](#footnote-2)

How has popular music changed over time? [Billboard](http://www.billboard.com/) maintains a list of top singles, the hot 100. We’ve compiled a dataset of these top songs. This part of the assignment allows you use this data to visualize trends in popular music.

You can download billboard.csv from the assignment page on the course’s website, and open it in Python using the below commands:

import pandas as pd

billboard = pd.read\_csv("tables/billboard.csv", encoding='unicode\_escape')

billboard["week"] = pd.to\_datetime(billboard["week"], format="%m/%d/%Y") billboard["entry\_date"] = pd.to\_datetime(billboard["entry\_date"], format="%m/%d/%Y")

*Part A*

First, let’s understand the data:

* What years does the data cover?

# 1940 -2016

* What is the difference between week\_peak\_position and overall\_peak\_position?

week\_peak\_position: This is probably the highest position that the song reached on the chart during the specific week being reported. This means if you're looking at the data for a particular week, the week\_peak\_position tells you what the highest chart ranking was for each song in that week.

overall\_peak\_position: This refers to the highest position the song has ever achieved on the chart during its entire time since being listed. This is an aggregate measure that doesn't change week to week unless the song reaches a new peak position. It gives you an idea of the song's highest level of popularity or success over time.

* What song has been in the charts for the longest number of weeks?

Week\_total\_weeks, and overall\_total\_weeks seems are the same.

Radioactive, Imagine Dragons

* What is the top song in the most recent chart?
* Hello – Adele Q: multiple songs listed as “1” , but most recent ..

*Part B*

Next, let’s explore patterns in the data using visualizations. I recommend reading the following blog post: [https://web.archive.org/web/20160208102955/http://www.modestinsights.com/analyz ing-the-billboard-hot-100/](https://web.archive.org/web/20160208102955/http:/www.modestinsights.com/analyzing-the-billboard-hot-100/) which analyzes the chart trajectory of different songs, the Beatle’s popularity in 1964, and patterns in artists’ careers.

(i) Replicate the plot of the Beatles’ popularity in 1964 with the following code:

import plotnine as p9

# Beatles Dominance in 1964

beatles = billboard[(billboard.artist=='BEATLES') & (billboard.week >=

'1964-02-01') & (billboard.week <= '1964-06-01')].reset\_index()

plot\_beatles = (p9.ggplot(beatles, p9.aes('week', 'this\_week\_position', group = 'song', colour = 'song'))

+ p9.scale\_y\_reverse() + p9.geom\_point() + p9.geom\_line()

+ p9.scale\_x\_datetime(breaks='1 month', date\_labels = "%b")

+ p9.ggtitle("Beatles Dominance in 1964") + p9.ylab("Position"))

p9.ggsave(plot\_beatles, filename='beatles.pdf')

This plot shows time on the x-axis and position on the y-axis. But we could represent the information in a different way.

Instead of putting rank on the y-axis, we could display the track name and use another aesthetic to display the rank.

* List possible aesthetics (other than position) that we could use to display rank, and then create plots using those aesthetics.
* What do you notice? What aesthetic makes it easiest to see the pattern over time? What makes it hardest? Does combining aesthetics make it better? What makes the original plot so good?
* The track names are displayed on the y-axis and utilize the size of the circles to indicate the songs' ranks. The size corresponds to the songs' current weekly position, with larger circles representing higher positions and smaller circles for lower ones.
* Additionally, the color gradient is also employed to represent the rank, with darker colors signifying higher positions and lighter colors indicating lower positions, transitioning from dark purple to light yellow.
* You’ve probably used size as one of the aesthetics - but what is size really mapped to? Create a small experiment to determine whether size is mapped to radius or area? Which do you think it should be mapped to?

(ii) Explore

Use the data to explore questions that you find interesting. We encourage you to choose your own investigations; some ideas:

* How has the popularity of female artists changed over time?
* How have pop musician career trajectories changed over time? Do artists peak sooner or later, and is their popularity more or less stable?
* How did pop music evolve in the 1990s?

Use plotnine.ggplot to produce at least 2 informative plots for your investigation. For each visualization you produce, produce two alternates that present the same information using different aesthetics. Describe which visualization you find most informative and why.

Summarize what you’ve learned in a half page writeup that refers to your visualization.

# Poisonous Mushrooms

A foodie friend wants to cook a dish with fresh collected mushrooms. However, he knows that some wild mushrooms are delicious and others can be deadly.

*‘There is no test to determine edible versus poisonous mushrooms. Ignore any advice such as “a poisonous mushroom will tarnish a silver spoon,” “if it bruises blue, it’s poisonous,” etc. These are old wives’ tales and folk myths, and completely untrue.’[[3]](#footnote-3)*

You’ve managed to collect data on 8,124 mushrooms, their features, and whether they are edible. It would be difficult to remember all of these individual mushrooms, so your goal in this assignment is to determine if there are rules of thumb that can help your friend.

The data is in the file “agaricus-lepiota.data.txt”

mushroom\_data = pd.read\_table("tables/agaricus-lepiota.data.txt", sep = ',', header=None)

First, let’s graph the data to try to identify patterns using the skills we’ve learned so far. The data records 22 different features for each mushroom, so we have a high dimensional space.

The feature values can be seen in the ‘agaricus-lepiota.names’ file. You are welcome to create new, more legible columns using apply() in Pandas if you prefer.

Explore different graphs of the data trying to identify which dimensions appear to be important for predicting whether a mushroom is edible. Include the dimension

‘edible’.

You can represent many dimensions, using the x-axis, y-axis, colors, shapes, sizes, and small multiples (facets).

*Hint: you can use geom\_jitter() at the end of your ggplot command, or set the opacity between zero and 1 (alpha) to prevent overplotting.*

Explore the data to come up with a rule of thumb to help your friend identify edible mushrooms. Write your rule of thumb and include graphs that demonstrate its performance.

1. Source: Hadley Wickham [↑](#footnote-ref-1)
2. Adapted from Hadley Wickham; licensed under the [CC Attribution-Noncommercial-Share Alike 3.0](http://creativecommons.org/licenses/by-nc-sa/3.0/us/)

   [License.](http://creativecommons.org/licenses/by-nc-sa/3.0/us/) Borrows from <http://www.modestinsights.com/analyzing-the-billboard-hot-100/> [↑](#footnote-ref-2)
3. https://mdc.mo.gov/discover-nature/activities/mushroom-hunting [↑](#footnote-ref-3)