Storytelling Data Visualization: Takeaways

Syntax

- Creating a grid chart of four plots:

```
fig, (ax1, ax2, ax3, ax4) = plt.subplots(nrows=4, ncols=1)
```

- Modifying all Axes objects using a for loop:

```
fig, (ax1, ax2, ax3, ax4) = plt.subplots(nrows=4, ncols=1)
axes = [ax1, ax2, ax3, ax4]
for ax in axes:
    ax.plot(x_coordinates, y_coordinates)
    ax.set_yticklabels([])
plt.show()
```

- Adding a horizontal line:

```
ax.axhline(y)
```

- Adding a horizontal line with various parameters:

```
ax1.axhline(y=1600, xmin=0.5, xmax=0.8,
            linewidth=6, color='#af0b1e', alpha=0.1)
```

Concepts

- In a broad sense, a story is a sequence of events: something happens, then something else happens, and so on. A graph that only shows numerical facts isn't a story.
- Another story element is change: something or someone changes throughout the story. A static graph that doesn't show any element of change isn't a story.
- To create a data story, we need to wrap numerical facts into events that show change.
- Matplotlib can be very powerful if you're a little imaginative. You'll often find yourself wanting to do X and search for a specific function to do X. This approach won't always work because the function you want may not exist.
- However, you can often combine what Matplotlib already has to get what you want. To do that, identify the basic parts of what you want to create. Then, try to create the basic parts using Matplotlib.

Resources

- The Visual Display of Quantitative Information — Edward Tufte
- Matplotlib Gallery
- Is Your Data Story Actually a Story — Joshua Smith