Issue #1:

This is the first issue that we decided to work on. As described above, the issue here is that the "fontproperties" kwarg can overwrite previous kwargs which were used to update a Text object, since the kwargs are processed sequentially.

The reason that we decided to select this issue, is that it seems like we can issue a simple fix without changing too much of the code. As we are still new to this codebase, it seems like this was a good introductory bug for us to fix, and we could do it without worrying too much about dependencies in the code. As mentioned above, we do not envision this bug taking much time to fix, and we estimate the majority of time will be spent testing rather than coding.

Potential risks with this bug, is that the fix we are proposing will be used all over the code, anytime that a text object is updated, so we will need to make sure that the fix behaves the way we want it to in all situations so that the text is not affected in unexpected ways in other areas of the code.

Software Development Process:
Issue #1 Analysis:
The issue occurs in the method set_fontproperties inside text.py. Based on its implementation, any kwargs passed before the fontproperties kwarg will be overridden (not just the size kwarg). It is because this method creates a copy of all the defaults and then directly replaces all of them back to their defaults afterwards. To fix the issue, we will get all the font properties beforehand (using getters from font_manager.py) to get all properties, and then set all of them back after setting the font properties.

Reproduction of error:
data = np.random.randn(100)
plt.hist(data)

[Note: works]
plt.xlabel("Days", fontproperties="Comic Sans MS", weight="bold", size=16)

[Note: weight, size will be set back to default]
plt.xlabel("Days", weight="bold", size=16, fontproperties="Comic Sans MS")

plt.ylabel("Food Consumption", fontproperties="Comic Sans MS", weight="bold", size=16)
plt.show()

Issue #1 Design:
Refer to Issue16389FixPlan.pdf

Issue #1 Commentary:
The changes made to solve this issue did not change the design much, but rather serves as a workaround for **FontProperties()** overriding the properties that are already set. For solving this issue, the update method in the Text class is forced to update fontproperties first. No files were removed, and one test file is added (i.e. test_fix16389.py). Thus, the change did not have a big impact on the code base.