Homework 0: Assignments Tips and Self-Assessment

CSCI 2565 · Fall 2023 · NYU Computer Science

1 Assignments Tips

The homeworks will start with the instructions below. The rest of this section is meant to help you choose a solution to typeset your work. For the programming part of the assignments, you should start getting confortable with the Python packages NumPy and matplotlib if you are not already.

Due: N/A

Instructions: Your answers to the questions below, including plots and mathematical work, should be submitted as a single PDF file. It's preferred that you write your answers using software that typesets mathematics (e.g.LaTeX, LyX, or MathJax via iPython), though if you need to you may scan handwritten work. You may find the minted package convenient for including source code in your LaTeX document. If you are using LyX, then the listings package tends to work better.

To use Minted package, you may need to install pygmentize package for python by running `python3 -m pip install Pygments`, and add the executable to PATH. You can then build LaTeX with "`latexmk -pdf -shell-escape hw0_latex.tex`"

1.1 LaTeX

You should check the source file "hw0_latex.tex" in the .zip for an example of LaTeXtypesetting.

Minted Package The minted package is convenient for including source code in your LaTeX document.

Including Python Code from File Here we're extracting lines 4 through 13 from the file code.py.

```python
def dotProduct(d1, d2):
    """
    @param dict d1: a feature vector represented by a mapping from a
    ↪  feature (string) to a weight (float).
    @param dict d2: same as d1
    @return float: the dot product between d1 and d2
    """
    if len(d1) < len(d2):
```

```
        return dotProduct(d2, d1)
    else:
        return sum(d1.get(f, 0) * v for f, v in d2.items())
```

Python Code Inline   Here we're extracting lines 4 through 13 from the file code.py.

```python
def dotProduct(d1, d2):
    """
    @param dict d1: a feature vector represented by a mapping from a
    ↪  feature (string) to a weight (float).
    @param dict d2: same as d1
    @return float: the dot product between d1 and d2
    """
    if len(d1) < len(d2):
        return dotProduct(d2, d1)
    else:
        return sum(d1.get(f, 0) * v for f, v in d2.items())
```

1.2   Jupyter notebooks

Check "hw0_jupyter.ipynb" for a solution relying on Jupyter Notebooks.

2   Self-Assessment

In the .zip file, you will also find a "math-self-assement.pdf". The questionnaire is meant to give you a preview of the mathematical objects and notations we will use in the class. Take the time to have a quick look to be aware of were you stand!