**Data Visualization with Matplotlib Review Notes**

Python has a wide variety of useful packages for machine learning and statistical analysis such as \*tensorflow\*, \*numpy\*, \*scikitlearn\*, \*pandas\*, and more. One package that is essential to most data science projects is \*\*matplotlib\*\*.

->

\*TensorFlow\*, \*NumPy\*, \*scikit-learn\*, \*Pandas\*

**Comment: Please capitalize concept, frameworks and classnames accordingly.**

\*TensorFlow\*, \*NumPy\*, \*scikit-learn\*, \*Pandas\*

->

<a rel=”nofollow” target=”\_blank” href=”https://www.tensorflow.org/”>TensorFlow</a>, <a rel=”nofollow target=”\_blank” href=”http://www.numpy.org/”>NumPy</a>, <a rel=”nofollow” target=”\_blank” href=”http://scikit-learn.org/stable/”>scikit-learn</a>, <a rel=”nofollow” target=”\_blank” href=”https://pandas.pydata.org/”>Pandas</a>

**Comment: When writing articles, please assume that the reader is a novice. This means linking to relevant frameworks and concepts when introducing them.**

pip -> `pip`

**Comment: When referencing code snippets, variable names, classnames or any other code/command related text shorter than a line, please format it using the `name` syntax.**

`python3 -m pip install matplotlib`

->

```sh

python3 -m pip install matplotlib

```

**Comment: I can see that you’re aware of the proper formatting rules, as well as how to write code snippets, judging by the rest of the article, so there’s no point in explaining how to format code. Since this is a single line long, it’s considered a code example and, therefore, should be formatted using adequate syntax.**

In a Python file, we want to import the \*pyplot\* function that allows us to interface with a MATLAB-like plotting environment. We also import a \*lines\* function that lets us add lines to plots.

->

In a Python file, we want to import the `pyplot()` function that allows us to interface with a MATLAB-like plotting environment. We also import a `lines()` function that lets us add lines to plots: