**Style Guide for Python Programming**

Please keep to our style guide for programming in Python in the module machine learning   
(mandatory for seminars

Changelog:

* SH 7.1.2021, first creation
* NL 14.04.2021, visual revamp

Follow the general instructions below

* If possible, comment in English language.
* Please keep to the style the guidelines of PEP 8 standard
  + PEP8 online Guide, https://www.python.org/dev/peps/pep-0008/#indentation
  + use a Pep 8 checker e.g., <https://pypi.org/project/pep8/>
  + or online, http://pep8online.com/

Please write a **header** as in the following example:

# ---------------------------------------------------------------------------

# Number recognition

# Authors: Fabian Wittenbrink, Nils Langer

# Copyright: Free to use

# Credits: E. Selbstreich, S. Hafner, Python-exemplarisch

# Version: 1.0.0

# Maintainer: Prof. Hafner

# E-Mail: hafner.sigrid@fh-swf.de

# Status: Development

# Creation Date: 23-11-2020

# Last Edit: 16-04-2021

# Python-Version: 3.7.9

# Description: Programm for number recognition based on the pixels of a picture

# ---------------------------------------------------------------------------

If you use other version of libraries than in class, please write a batch file to include it, like we gave it to you in the beginning of the module.

(for seminar work you have to use our libraries and version of the batch file, no others are allowed)

Please do numpy **docstring** for functions, modules and classes

In programming, a **docstring** is a string literal specified in source code that is used, like a comment, to document a specific segment of code.

* Docstring online guide, <https://python-sprints.github.io/pandas/guide/pandas_docstring.html>
* Docstring conventions are defined in PEP-257, <https://www.python.org/dev/peps/pep-0257/>

Example for a docstring:

def complex(real=0.0, imag=0.0):

"""Form a complex number.

Keyword arguments:

real -- the real part (default 0.0)

imag -- the imaginary part (default 0.0)

"""

**Please write a Readme for bigger projects**

In the Readme file you give a brief summary of the project and its purpose. Include any special requirements for installation or operating the project, bug reporting. https://realpython.com/documenting-python-code/

Please write an additional file readme.txt and put in:

Filename:

Authors:

Copyright:

Credits:

Version:

Maintainer:

E-Mail:

Status:

Creation Date:

Last Edit:

Python Version:

Also, you should include the following things:

* Example how the program is used.
* Development setup.
* Release History
* Needed libraries
* Possible problems and their solutions