



FH Salzburg

Machine Learning

Reuf Kozlica & Maximilian Schirl

March 20, 2020

Course Outline

	Dates	Lecture	Assignment
ILV01	24.03.2020	Recap Feature Engineering	Not Graded
LB02	31.03.2020	Naive Bayes	Before Next Lecture
LB03	05.05.2020	The Perceptron	Before Next Lecture
ILV04	19.05.2020	Interpreting Learning Progress	Not Graded
LB05	26.05.2020	Multilayer Perceptron & Parameter Tuning	14.06.2020
LB06	26.05.2020		

Assignment	Percentage
LB02	25%
LB03	25%
LB05 & LB06	50%
This course constitutes 33% of your overall grade.	

- [Anaconda](#) for Python 3.7 with [Jupyter Notebook](#)
- [scikit-learn](#) 0.22.1
- [Keras](#) 2.3.1
- [TensorFlow](#) 2.1.0
- ... or use the given [Docker](#) image

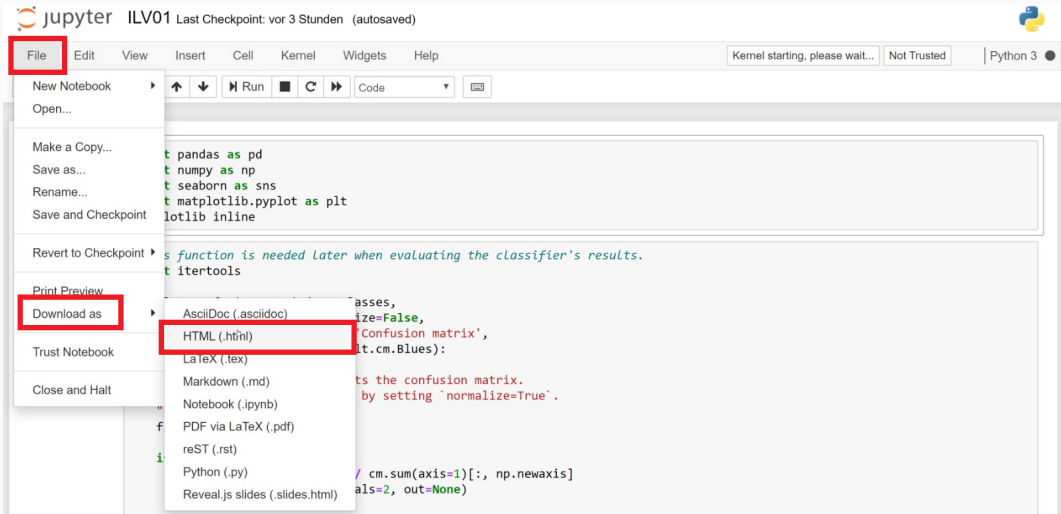
Preparation

- Install [Docker for your operating system](#)
- Follow the usage section of the custom [Docker image](#)
- Within the Docker container, change your working directory to /notebooks
- Clone the given repository
`https://github.com/mschirl/machine-learning.git`

Submission of Assignments

- Export your Jupyter Notebook as HTML file
- Upload the exported file to Moodle before the deadline specified in the Course Outline

Jupyter Notebook HTML Export



The screenshot shows the Jupyter Notebook interface. The top bar displays the Jupyter logo, the name 'ILV01', and the status 'Last Checkpoint: vor 3 Stunden (autosaved)'. The right side of the top bar shows the Python version 'Python 3' and a 'Not Trusted' warning. The 'File' menu is open, and the 'Download as' option is selected, which has opened a submenu. In this submenu, the 'HTML (.html)' option is highlighted. Other options in the 'File' menu include 'New Notebook', 'Open...', 'Make a Copy...', 'Save as...', 'Rename...', 'Save and Checkpoint', 'Revert to Checkpoint', 'Print Preview', 'Trust Notebook', and 'Close and Halt'. The 'Download as' submenu also includes options for 'AsciiDoc (.asciidoc)', 'LaTeX (.tex)', 'Markdown (.md)', 'Notebook (.ipynb)', 'PDF via LaTeX (.pdf)', 'reST (.rst)', 'Python (.py)', and 'Reveal.js slides (.slides.html)'. The background shows a code editor with Python code for importing libraries (pandas, numpy, seaborn, matplotlib) and a function definition.

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix

def plot_confusion_matrix(cm, title='Confusion matrix', cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    By default the confusion matrix is printed by setting 'normalize=True'.
    """
    plt.figure(figsize=(10, 10))
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    text = cm.sum(axis=1)[:, np.newaxis]
    fmt = 'r'
    for i, j in range(cm.shape[0]):
        for k, l in range(cm.shape[1]):
            text[i, j] = cm[i, j]
    plt.xticks(range(cm.shape[1]), labels, format=fmt, align='center')
    plt.yticks(range(cm.shape[0]), labels, format=fmt, align='center')
    plt.tight_layout()
```