

Intelligente Systeme Praktikum (SoSe 2017)

In den **Praktika 3 und 4** sollen die Inhalte der Vorlesung zum Thema **Machine Learning** gefestigt und vor allem der Umgang mit derzeit besonders verbreiteten Frameworks aus diesem Bereich vermittelt werden. Dadurch soll die Möglichkeit eines interessanten Querschnitts durch Anwendungsbereiche des **Supervised Machine Learning** geboten werden.

Thematisch sind die beiden Termine folgendermaßen eingeordnet:

- Termin 3: lineare und logistische Regression mit Gradientenverfahren
- Termin 4: Neuronale Netze zur Klassifikation

Anleitung zur Installation der Software

Die Aufgaben sollen mit **Python** (3.5.x unter Windows) und einigen aktuellen Paketen zur Lösung von Machine-Learning-Problemen bearbeitet werden. Dazu ist neben Python die Installation der folgenden Pakete notwendig:

- Numpy
- Matplotlib
- Scipy
- Scikit-Learn
- Pandas
- TensorFlow
- Keras (ab TensorFlow 1.1 als Modul von Tensorflow verfügbar)

Die Installation kann auf Windows-, Linux- oder Mac-Systemen erfolgen. Zu empfehlen ist allerdings ein Linux System wie z.B. Ubuntu 16, da hier die Installation besonders unkompliziert ist.

Als Editor zur Bearbeitung der Aufgaben kann ein herkömmlicher Texteditor, oder die IDE der Wahl verwendet werden. Zu empfehlen ist hier der flexible Editor **Atom** (<https://atom.io/>), welcher durch Pakete z.B. zur Autovervollständigung von Python-Code erweitert werden kann.

Hinweis: In dieser Anleitung wird lediglich eine Variante von TensorFlow installiert, die ausschließlich auf der CPU läuft. Sobald eine Nvidia-GPU zur Verfügung steht, kann eine Version von TensorFlow mit GPU-Unterstützung installiert werden. Eine herkömmliche Gaming-GPU (z.B. GTX 1080/1070, GTX 980Ti) kann hierbei das Training von neuronalen Netzen deutlich beschleunigen.

Installation unter Linux

Zunächst muss **Python** installiert werden.

```
$ sudo apt install python3 python3-pip
```

Hierdurch wird auch der Paketmanager **pip** installiert, sodass über diesen nun die restlichen Pakete heruntergeladen werden können.

```
$ pip3 install numpy matplotlib pandas scipy scikit-learn tensorflow keras
```

Bekannte Probleme

Matplotlib

Der Python-Paketmanager installiert ebenfalls Abhängigkeiten zu anderen Paketen. Ausgenommen hiervon sind allerdings System-Bibliotheken. Sobald z.B. bei der Installation des Paketes **matplotlib** der folgende Fehler auftritt, so kann dies auf fehlende Bibliotheken hinweisen.

```
Command python setup.py egg_info failed with error code 1 in /tmp/pip-build-9fk3l21g/matplotlib
```

Die Fehlermeldung spezifiziert weiter, welche Abhängigkeiten fehlen:

* The following required packages can not be built: freetype, png

In diesem Fall können die entsprechenden Bibliotheken einfach installiert werden.

```
$ sudo apt install libpng-dev libfreetype6-dev
```

Eine vollständige Liste der Abhängigkeiten des Paketes **matplotlib** ist hier zu finden: <http://matplotlib.org/users/installing.html#build-requirements>

Ein weiterer möglicher Fehler bei der Verwendung von matplotlib ergibt die folgende Fehlermeldung:

```
ImportError: No module named _tkinter, please install the python-tk package
```

Dies kann behoben werden, indem das folgende Paket installiert wird:

```
$ sudo apt install python-tk
```

Scipy

Auch das Paket **Scipy** benötigt System-Bibliotheken. Tritt ein vergleichbarer Fehler zu diesem auf **NotFoundError: no lapack/blas resources found**, so müssen die folgenden Bibliotheken installiert werden:

```
$ sudo apt install libblas-dev liblapack-dev libatlas-base-dev gfortran
```

TensorFlow

Möglicherweise wird bei der Installation mittels **pip3** das Paket **tensorflow** nicht gefunden:

```
Could not find any downloads that satisfy the requirement tensorflow
```

In diesem Fall kann TensorFlow folgendermaßen installiert werden:

```
# Für Python 3.4.x
```

```
$ pip3 install --upgrade
```

```
https://storage.googleapis.com/tensorflow/linux/cpu/tensorflow-1.0.0-cp34-cp34m-linux\_x86\_64.whl
```

```
# Für Python 3.5.x
```

```
$ pip3 install --upgrade
```

```
https://storage.googleapis.com/tensorflow/linux/cpu/tensorflow-1.0.0-cp35-cp35m-linux\_x86\_64.whl
```

Weitere Informationen sind hier zu finden: <https://www.tensorflow.org/install/>

Installation unter Windows (mittels Anaconda)

Die Installation unter Windows kann am einfachsten über wissenschaftliche Python-Distributionen wie z.B. **Anaconda** oder **WinPython** erfolgen. Im Folgenden wird die Installation mittels **Anaconda** beschrieben.

Anhand des folgenden Links kann Anaconda heruntergeladen und installiert werden:

- Windows 64-Bit: https://repo.continuum.io/archive/Anaconda2-4.2.0-Windows-x86_64.exe
- Windows 32-Bit: <https://repo.continuum.io/archive/Anaconda3-4.3.0.1-Windows-x86.exe>

Anaconda basiert in der aktuellen Version auf Python 3.6, allerdings wird für die verwendeten Pakete unter Windows **Python 3.5** benötigt. Aus diesem Grund muss zunächst eine virtuelle Umgebung für die korrekte Python-Version erstellt werden. Dies kann mittels **cmd** erfolgen:

```
# Erstellen einer Python 3.5 Umgebung unter dem Namen "isp"
```

```
$ conda create -n isp python=3.5
```

```
# Aktivieren der Umgebung
```

```
$ activate isp
```

Nun können in der virtuellen Umgebung die benötigten Pakete installiert werden:

```
# Installation von Paketen, die bereits in Anaconda enthalten sind  
$ conda install numpy matplotlib pandas scipy scikit-learn
```

```
# installation der übrigen Pakete mittels pip  
$ pip install tensorflow keras
```

Zu beachten: Für die Bearbeitung der Aufgaben muss nun immer die virtuelle Umgebung verwendet werden, da nur dort die benötigten Pakete installiert wurden.

Testen der Installation

Nach der erfolgreichen Installation können die geladenen Pakete anhand eines einfachen Imports getestet werden.

```
$ python  
...  
>>> import numpy, matplotlib, pandas, scipy, sklearn, tensorflow, keras
```