AI701: Foundations of Artificial Intelligence
Fall 2022

Lab-01

Machine Learning Fundamentals and Supervised Learning

August 29, 2022

Instructions:

- Open the provided notebook on google colab or set up the environment locally to use jupyter notebook.

- Familiarize with the python libraries.

- Complete the exercises enlisted in the tasks.

The objective of this introductory lab is to familiarize with some of the important libraries in Python that are essential in practicing Machine Learning methods.

1 Programming environment and libraries

The objective is to set up an environment and familiarize with some python libraries commonly used in machine learning. These include Numpy (for linear algebra tools), Matplotlib (for visualization), and Scikit-learn (for machine learning algorithms).

Setting up the Environment

To run the exercise, you may choose from one of the following options:

1. Google Colab environment: Run the Google Colab file

   (a) Download the notebook (google colab file) provided with this instruction manual.
   (b) Upload the notebook to your google drive to run the notebook on google colaboratory.

2. Non-Google Colab environment (local): Create environment with *conda*

   (a) Open a terminal in the directory containing the week-1 lab.
   (b) Build a new conda environment using the provided .yml file. A .yml file is a text file that contains a list of dependencies. Use the following command

   ```
   conda env create −n ai701 −f ai701.yml
   ```

   (c) Activate the conda enviroment:

   ```
   conda activate ai701
   ```

   (d) Install jupyter notebook in your conda environment:

   ```
   conda install jupyter
   ```

   (e) Open jupyter notebook using the command:

   ```
   jupyter notebook
   ```

2 Getting Started with NumPy

NumPy is the fundamental package for scientific computing in Python that provides many numerical programming tools, such as matrix operations and vector processing, etc. NumPy gives you an enormous range of fast and efficient ways of creating *arrays* and manipulating numerical data inside them. An array is a central data structure of the NumPy library. An array is a grid of values that can be indexed in various ways. The elements are all of the same type, referred to as the array dtype.

A simple difference between Python list and *NumPy arrays*, is that while a Python list can contain different data types within a single list, all of the elements in a NumPy array should be homogeneous. They are faster and more compact than Python lists. An array consumes less memory and is convenient to use. You may find this tutorial useful to get started with NumPy.

The jupyter notebook covers the following tasks:

1. Create NumPy array

2. Indexing NumPy arrays

3. Linear algebra using NumPy operations

4. Common operations and functions

3 Intro to scikit-learn

Scikit-learn, also known as sklearn, is an open source machine learning library that supports supervised and unsupervised learning. Sklearn includes many subpackages designed for machine learning tasks, such as,

- Classification

- Regression

- Dimensionality reduction

- Model selection

- Clustering

- Data preprocessing

4 Visualizing data

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. We will have a brief introduction to the matplotlib.pyplot module, which provides a plotting system similar to that of MATLAB. With a few lines of code, we can generate plots, histograms, bar graphs, and scatter plots, using Matplotlib.

To use pyplot library, you first need to import the package from Matplotlib:

```
import matplotlib.pyplot as plt
```

5 Tasks

In this lab session, we will familiarize with the discussed Python libraries, NumPy (for linear algebra tools), Scikit-learn (for machine learning algorithms) and Matplotlib (for visualization). For each task, please refer to the Jupyter-Notebook that has been provided with this lab instruction manual.
Task 1: Getting started with Numpy: Explore the NumPy library by following the given exercises.
Task 2: Get started with Scikit-learn: Download dataset from Scikit-learn and make a train-test split of dataset.
Task 3: Train the data using Scikit-learn estimators on the train split created.
Task 4: Evaluate your model on the test set.
Task 5: Getting started with data visualization using Matplotlib: Explore the Matplotlib libraries by following the given exercises.