Deep Learning for Social Sciences - Assignment: Regression Task with Multilayer Perceptrons

Giordano De Marzo

8 May 2024

To be handed in by 28.5.2024 23:59 at `https://github.com/orgs/DLSS-24/`.

Overview

In this assignment, you will use a Multilayer Perceptron (MLP) to tackle a regression task. Your objective is to apply a deep learning model to predict the house price in California from a dataset containing various input features. Follow the steps outlined below to complete this assignment.
Complete tasks for receiving points, the maximum number of points you can get is 30. You can get bonus points completing the optional tasks. These points will not increase your mark over 30, but can increase it up to 30 if you made some errors in the mandatory tasks (see below if this is not clear).

Tasks

1. Data Import and Preparation (5 points)

- Download the California Housing dataset from sklearn (see code snippet below if you need help).
- Visualize the distribution of each feature and identify any missing values or outliers.

- Apply appropriate data cleaning and preprocessing techniques, including:
  - Extract the features (x) and the target (y) from the dataset
  - Standardization of numeric features
  - Handling missing values
  - Encoding of categorical variables, if applicable

2. Train-Validation-Test Split (2 points)

- Divide the cleaned dataset into training, validation, and testing sets.
- Ensure that the test set is not used during model training.

3. MLP Model Building and Training (10 points)

- Design and build an MLP architecture suitable for regression, including:
  - Appropriate activation functions and loss
  - An output layer that aligns with regression requirements
- Train at least 3 different variants of the model (varying layers, neuron counts, dropout, regularization, etc.).
- Track the learning curves for each variant.

4. Model Evaluation and Selection (2 points)

- Evaluate the performance of each model on the validation set.
- Select the best-performing model based on validation set results.

5. Final Testing and Comparison (3 points)

- Test the best model on the test set and compare its results to a simple machine learning model, such as a linear regressor.

6. Report Writing (8 points)

- Write a report (max 3 pages) commenting on the results of the different models, noting any signs of overfitting or underfitting and explaining potential causes.

- The report must report the learning curves of the 3 models and descriptions of the model architectures. Use a table to report the main hyper-parameters of each model.
- Ensure your report discusses your approaches, insights, challenges, and conclusions.

Bonus Task 1: Standard Machine Learning (2 bonus points)

- Train a Random Forest Regressor to predict the housing price.
- Tune its hyper-parameters to get a good competitor for the MLP.

Bonus Task 2: Model Ensemble (2 bonus points)

- Create an ensemble of at least three different MLP architectures.
- Use a voting strategy to make predictions and compare performance against individual models.

Submission Guidelines

- Submit your code and a report documenting your findings.
- The code must be a colab (ipynb file) notebook, well formatted and commented.
- The report must be in PDF format.
- Upload everything on Github.

Code Snippet: Data Download

To help you get started, use the code below to download the California Housing dataset:

Listing 1: Downloading the California Housing Dataset

```
from sklearn.datasets import fetch_openml
import pandas as pd
```

```
# Fetch the dataset from OpenML
california = fetch_openml(data_id=537, as_frame=True)

# Extract features and target
data = california.frame

# Display the first few rows of the dataset
print(data.head())
```

The dataset is returned as a DataFrame (by setting 'as_frame=True'), making it easier for feature engineering and visualization.

Grading

Example 1:

- Task 1 - 5/5 points
- Task 2 - 2/2 points
- Task 3 - 7/10 points
- Task 4 - 2/2 points
- Task 5 - 3/3 points
- Task 6 - 6/8 points
- Bonus Task 1 - 2/2 points
- Bonus Task 2 - 1/2 points

Points 25/30
Bonus Points 3/4
Total Points 28/30
Final Mark 28/30

Example 2:

- Task 1 - 5/5 points
- Task 2 - 2/2 points

- Task 3 - 9/10 points
- Task 4 - 2/2 points
- Task 5 - 3/3 points
- Task 6 - 7/8 points
- Bonus Task 1 - 2/2 points
- Bonus Task 2 - 1/2 points

Points 28/30
Bonus Points 4/4
Total Points 32/30
Final Mark 30/30