Homework 1: Linear Regression and Neural Network Regression.

Deadline: 9-16-2024

***GPUs are not necessary for speeding up this neural network training.**

Description

The homework will use the same dataset to train a linear regression model and a deep fully connected neural network. You will compare the model's performance using this dataset. Please follow the six-machine learning training component step by step.

Data preview:

The goal of this project/challenge is to predict the results of Cancer Mortality Rates.
Therefore, the label is "TARGET_deathRate".

These data were aggregated from a number of sources including the American Community Survey (https://www.census.gov), https://www.clinicaltrials.gov, and https://www.cancer.gov.

In the past, the best model achieved **R-squared of 0.9624**.

Step 1: Data

Before starting to train a model, please get familiar with the dataset. When you look at the dataset, please answer the following questions:

1) How many data samples are included in the dataset?
2) Which problem will this dataset try to address?
3) What is the minimum value and the maximum value in the dataset?
4) How many features in each data samples?
5) Does the dataset have any missing information? E.g., missing features.
6) What is the label of this dataset?
7) How many percent of data will you use for training, validation and testing?
8) What kind of data pre-processing will you use for your training dataset?

Hint: You should use the same test dataset to compare the model's performance.

Step2: Model

Here I selected linear regression and deep fully connected neural network as model. However, you will experience different hyperparameters. Please try out the following hyperparameters and report the model performance in the testing dataset.

Model	Test R-squared
Linear regression	0.774

DNN-16	0.788
DNN-30-8	0.827
DNN-30-16-8	0.672
DNN-30-16-8-4	0.843
Any other?	

*DNN-30-8: The deep fully connected neural network contains two hidden layers; the first hidden layer uses 30 nodes, and the second layer uses 8 nodes.

Step 3: Objective

Mean Squared Error (MSE) is the loss function you will use to train your models.

$$L = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Step 4: Optimization

We have not covered the optimization topic yet. Therefore, you will use the default setting of Stochastic Gradient Descent (SGD) to train your model

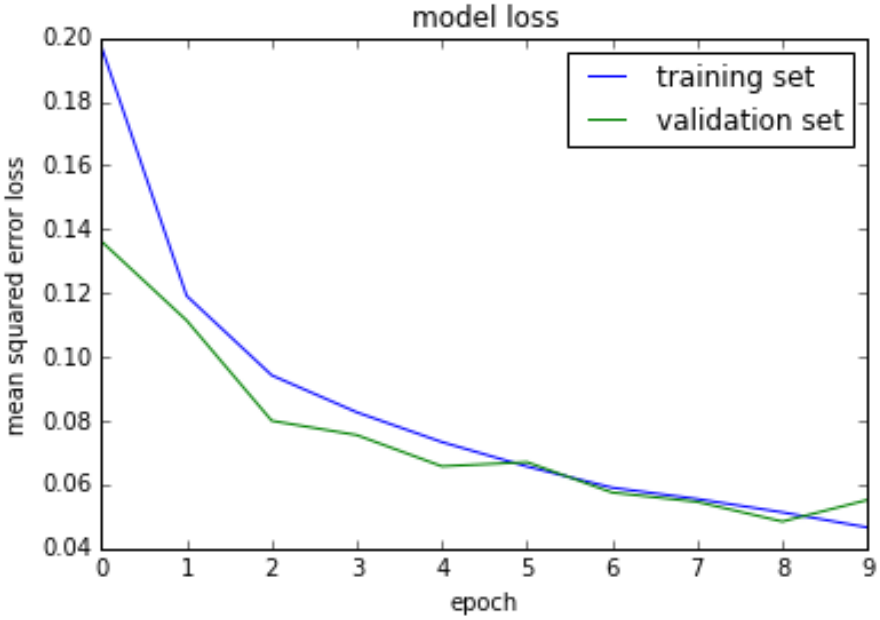
Step 5: Model selection

Model	LR: 0.1 (R^2)	LR: 0.01 (R^2)	LR: 0.001 (R^2)	LR: 0.0001 (R^2)
Linear regression				
DNN-16				
DNN-30-8				
DNN-30-16-8				
DNN-30-16-8-4				
Any other?				

Step 6: Model performance

In this step you should report your model performance, which you did in the previous steps. Report the MSE of linear regression and all the DNN model architecture you tried. Please add the model performance plot in this step.

E.g.



What should you submit?

You should submit a zip file containing:

1. Your homework report from step 1-6. You will answer all the questions in each step and fill the tables in step 2, step 5 and performance plot in step 6. Miss any part will lose some points. Please double check you have addressed all the questions.

2. Your code of linear regression and DNN models. Each model should include a README file explaining how to run the model. Your code should be well commented. In your code, you should have a function called *test_model.* The *test_model* will load the trained model and load test dataset to predict.

3. Your highest performed DNN model weights and Linear regression model.

4. A folder contains screenshot of iteration of model's training and testing with timestamp.