National Chung Cheng University

Introduction to Deep Learning

Mid-Term Programming Exam (Take-Home)

2023/4/13 ~ 2022/4/30 (deadline: 23:59:59)   50% of total grade

You are provided a time-series forecasting problem centered around restaurant visitors. The data comes from two separate sites:

- Hot Pepper Gourmet (hpg): similar to Yelp, here users can search restaurants and also make a reservation online
- AirREGI / Restaurant Board (air): similar to Square, a reservation control and cash register system

You must use the reservations, visits, and other information from these sites to forecast future restaurant visitor totals on a given date. The dataset covers the dates from 2016 until April 2017. The dataset should be divided into training data (full year of 2016) and testing data (whatever is given for 2017).

There are days in the test set where the restaurant were closed and had no visitors. These are ignored in scoring. The training set omits days where the restaurants were closed.

File Descriptions

This is a relational dataset from two systems. Each file is prefaced with the source (either air_ or hpg_) to indicate its origin. Each restaurant has a unique `air_store_id` and `hpg_store_id`. Note that not all restaurants are covered by both systems, and that you have been provided data beyond the restaurants for which you must forecast. Latitudes and Longitudes are not exact to discourage de-identification of restaurants.

air_reserve.csv

This file contains reservations made in the air system. Note that the `reserve_datetime` indicates the time when the reservation was created, whereas the `visit_datetime` is the time in the future where the visit will occur.

- `air_store_id` - the restaurant's id in the air system
- `visit_datetime` - the time of the reservation
- `reserve_datetime` - the time the reservation was made

- `reserve_visitors` - the number of visitors for that reservation

hpg_reserve.csv

This file contains reservations made in the hpg system.

- `hpg_store_id` - the restaurant's id in the hpg system
- `visit_datetime` - the time of the reservation
- `reserve_datetime` - the time the reservation was made
- `reserve_visitors` - the number of visitors for that reservation

air_store_info.csv

This file contains information about select air restaurants. Column names and contents are self-explanatory.

- `air_store_id`
- `air_genre_name`
- `air_area_name`
- `latitude`
- `longitude`

Note: latitude and longitude are the latitude and longitude of the *area* to which the store belongs

hpg_store_info.csv

This file contains information about select hpg restaurants. Column names and contents are self-explanatory.

- `hpg_store_id`
- `hpg_genre_name`
- `hpg_area_name`
- `latitude`
- `longitude`

Note: latitude and longitude are the latitude and longitude of the *area* to which the store belongs

store_id_relation.csv

This file allows you to join select restaurants that have both the air and hpg system.

- `hpg_store_id`

- `air_store_id`

air_visit_data.csv

This file contains historical visit data for the air restaurants.

- `air_store_id`
- `visit_date` - the date
- `visitors` - the number of visitors to the restaurant on the date

Submission:

You need to submit the code of your model and fill in the following information:

Model Information:

- Number of layers: 4(input layer + 2 x hidden layer + output layer)
- Number of units in each layer: 174, 200, 150, 1
- Activation functions used: Relu
- Loss function: Root Mean Squared Log Error (RMSLE)
- Cost function: Root Mean Squared Log Error (RMSLE)

Training Epochs: 50

Training Accuracy: 0.79629 (RMSLE loss)

Testing Accuracy: 0.80233 (RMSLE loss)

Optimization techniques employed: Adam

Difference in accuracies after each optimization technique that you applied:

(1) Optimization technique name: SGD

Before optimization: Training/Testing Accuracies =0.79629 (RMSLE loss)/ 0.80233 (RMSLE loss)

After optimization: Training/Testing Accuracies = 0.79720(RMSLE loss)/0.80292(RMSLE loss)

Any other changes: learning_rate = 0.01

(2) Optimization technique name: Adagrad

Before optimization: Training/Testing Accuracies = 0.79629 (RMSLE loss)/ 0.80233 (RMSLE loss)

After optimization: Training/Testing Accuracies = 0.78539(RMSLE loss)/0.78870(RMSLE loss)

Any other changes: learning_rate = 0.01

(3) Optimization technique name: Adam

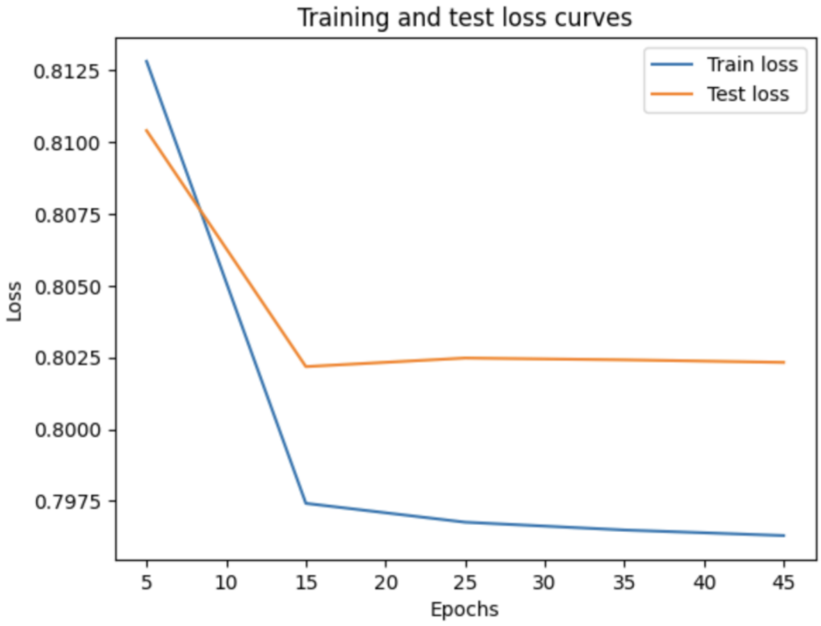
Before optimization: Training/Testing Accuracies = 0.79629 (RMSLE loss)/ 0.80233 (RMSLE loss)

After optimization: Training/Testing Accuracies = 0.78731 (RMSLE loss)/ 0.79052 (RMSLE loss)

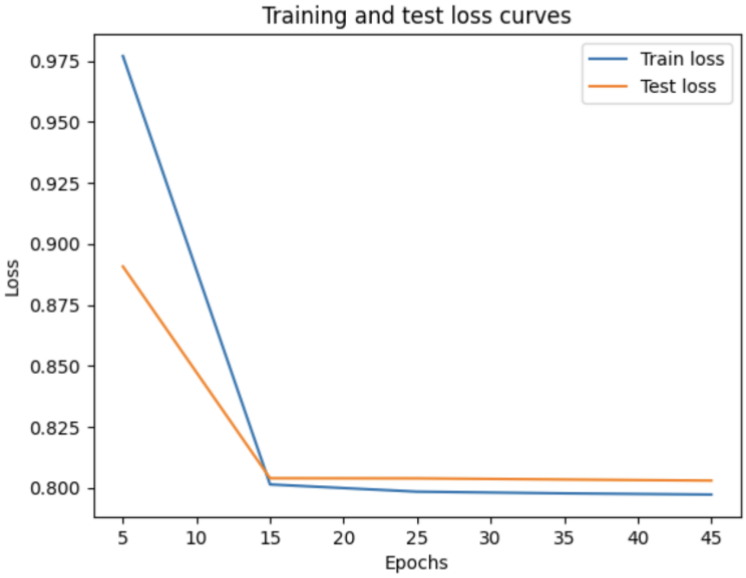
Any other changes: learning_rate = 0.01

Anything special about your model:

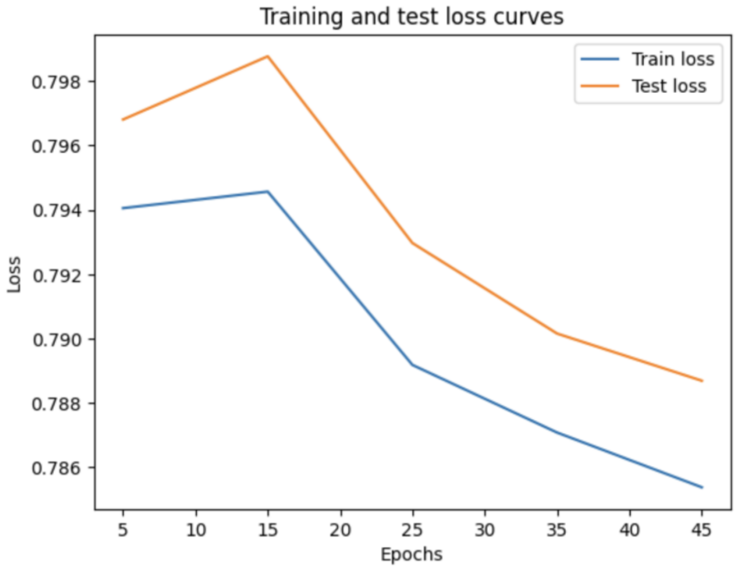
No optimizer:



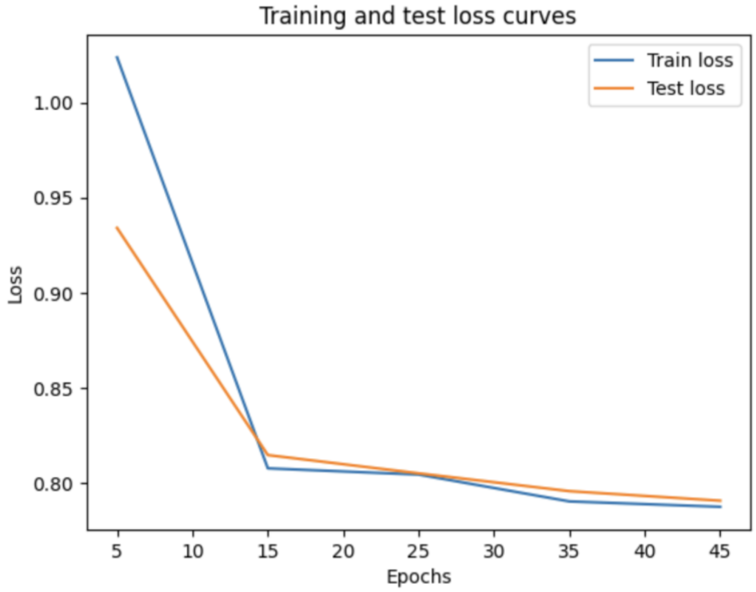
With optimizer(SGD):



With optimizer(Adagrad):



With optimizer(Adam):



Comments on the course:

在這個課程中，老師非常認真地講解了深度學習的基本原理和應用。老師的講課風格非常清晰易懂，讓學生可以很容易地理解和掌握所學內容。

此外，助教都非常耐心地回答學生的問題，讓學生可以更好理解和應用課程內容。他們也很用心準備作業，讓學生可以更深入地學習和應用

所學的知識。

最令人興奮的是，期末有 ART 整合至 kserve 的專案實作，葉信河執行長都會指導和提供寶貴的建議，讓我們在實踐中學習更多的知識和技能。