

COMP-5413 - Topics in Natural Language Processing
Assignment 1
Due by 5:00 pm Thursday, February 13

January 28, 2020

1 Task: Nonlinear regression w/t DL

- Load the modified version of the California Housing dataset (*housing.csv*) available at <https://github.com/ageron/handson-ml/tree/master/datasets/housing> into a Colab notebook.
- Perform the following visualization using the Python and Pandas libraries:
 - Print the first ten records of the dataset.
 - Plot each feature of the dataset on separate sub-plots, like shown in Fig. 1.

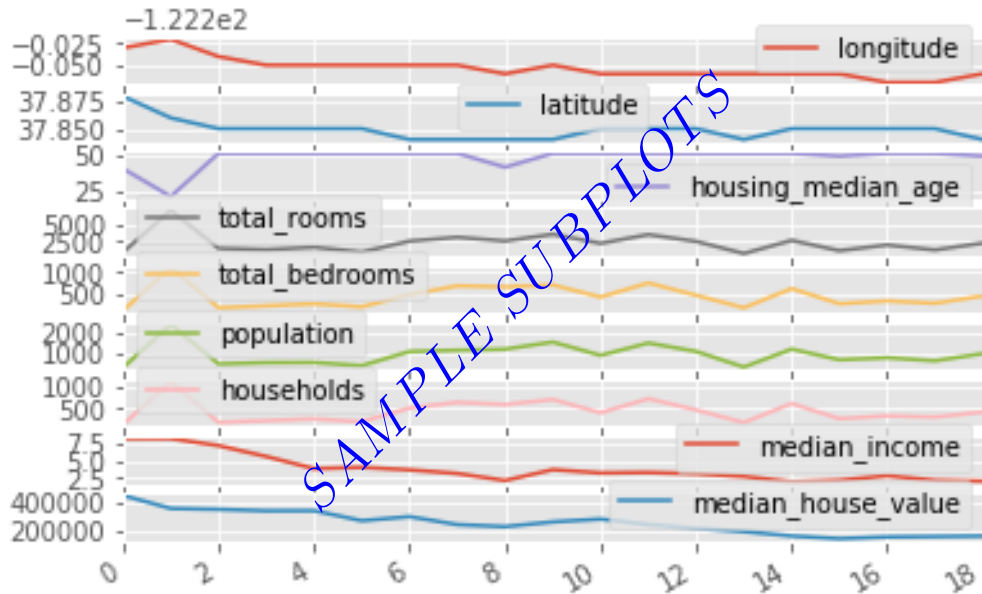


Figure 1: Example sub-plots of first eighteen samples from the dataset in a single figure.

- Implement a one dimensional convolution (Conv1D)-based neural network for predicting median house value using *longitude*, *latitude*, *housing median age*, *total number of rooms*, *total number of bedrooms*, *population*, *number of households*, and *median income*. The solution must be a nonlinear regression model. It must consider the following:

- Modular coding.
- Over/under fitting issue.
- Vanishing/exploding gradient issue.
- Number of trainable parameters.
- Kernel size
- Inference time.
- Train/test split ratio of 70:30 using *sklearn.model_selection* library with random state is set to 2003.

2 Deliverable

- A 3 to 5-page limit formal Scientific/Engineering report in IEEE format that clearly elaborates all the steps you carried out in completing the task.

A formal report by default covers: topic title, abstract, introduction, background/literature review, proposed model, experimental analysis/comparisons with other methods/approaches/applications if applicable, conclusion and references.

The write-up should take advantage of figures, plots, charts, flow charts, diagrams, tables, graphs, or such tools to clearly communicate the findings to any reader. The complexity of the writing must trade off between storytelling and presenting scientific/engineering concepts.

- All source code/script files in a separate directory.
- A trained model named in the format of *student id_1dconv_reg*.
- A Github link that contains all the details of your model.

3 Submission

All soft-copies must be submitted to the assignment folder on the D2L before the deadline, while a hard-copy of the report must be handed in to the instructor at the beginning of the class.

4 Evaluation

- It will be evaluated based on the merit of the solution and quality of the report.
- The top-3 models according to the test results in terms of MSE and R^2 score will be award with the following:
 - The best model - A gift card worth of \$50.00, a certificate, and 1 bonus mark.
 - The 2nd and 3rd models - A certificate and 1 bonus mark.

5 Hints

- i. Way of including source code in the report:

Let's consider there is a function written that acts differently based on an input string. Then, one of the excellent ways for adding the code of the function as follows.

Firstly, explain the function or the script in the main context of the report (again you can use flow charts or pseudo code), then include the code/script in actual code format (not in regular font) in Appendix as an example given below.

Example 1: The function *myCallService(url: String, urlFreeVer: String)* accepts two arguments: the URL of build variant from *BuildConfig* object and a preset URL for free version. It launches a second activity if the URL belongs to the Premium build. Otherwise, it produces a warning message to the user. The actual code snippet is given in Appendix 6.1.

6 Appendix

6.1 myCallService

```
1 fun callService(url: String, urlFreeVer: String) {
2     // calling code here
3     val url = "http://www.myver.com"
4     if (url.equals(urlFreeVer)) {
5         Toast.makeText(this, "Free ver", Toast.LENGTH_LONG).show()
6     }
7     else{
8         // Setting up an intent to open new activity
9         val myIntent = Intent(this, nextActivity::class.java)
10
11         // starting the 2nd activity
12         startActivity(myIntent)
13     }
14 }
```

Listing 1: myCallService Custom Function.