

Lab Five: Wide and Deep Networks

Due Apr 15 by 11:59pm **Points** 10 **Submitting** a file upload

Lab Assignment Five: Wide and Deep Network Architectures

In this lab, you will select a prediction task to perform on your dataset, evaluate two different deep learning architectures and tune hyper-parameters for each architecture. If any part of the assignment is not clear, ask the instructor to clarify.

This report is worth 10% of the final grade. Please upload a report (**one per team**) with all code used, visualizations, and text in a rendered Jupyter notebook. Any visualizations that cannot be embedded in the notebook, please provide screenshots of the output. The results should be reproducible using your report. Please carefully describe every assumption and every step in your report.

Dataset Selection

Select a dataset identically to lab one. That is, the dataset must be table data. In terms of generalization performance, it is helpful to have a large dataset for building a wide and deep network. It is also helpful to have many different categorical features to create the embeddings and cross-product embeddings. It is fine to perform binary classification or multi-class classification.

Grading Rubric

- **Preparation (40 points total)**
 - **[10 points]** Define and prepare your class variables. Use proper variable representations (int, float, one-hot, etc.). Use pre-processing methods (as needed) for dimensionality reduction, scaling, etc. Remove variables that are not needed/useful for the analysis. Describe the final dataset that is used for classification/regression (include a description of any newly formed variables you created).
 - **[10 points]** Identify groups of features in your data that should be combined into cross-product features. Provide justification for why these features should be crossed (or why some features should not be crossed).
 - **[10 points]** Choose and explain what metric(s) you will use to evaluate your algorithm's performance. You should give a **detailed argument for why this (these) metric(s) are appropriate** on your data. That is, why is the metric appropriate for the task (e.g., in terms of the business case for the task). Please note: rarely is accuracy the best evaluation metric to use. Think deeply about an appropriate measure of performance.
 - **[10 points]** Choose the method you will use for dividing your data into training and testing (i.e., are you using Stratified 10-fold cross validation? Shuffle splits? Why?). **Explain why your chosen method is appropriate or use more than one method as appropriate.** Convince me that your cross validation method is a realistic mirroring of how an algorithm would be used in practice.
- **Modeling (50 points total)**

- **[20 points]** Create several combined wide and deep networks to classify your data using Keras. Visualize the performance of the network on the training data and validation data in the same plot versus the training iterations. Try to use the "history" return parameter that is part of Keras "fit" function.
- **[20 points]** Investigate generalization performance by altering the number of layers in the deep branch of the network. Try at least two different number of layers. Use the method of cross validation and evaluation metric that you argued for at the beginning of the lab.
- **[10 points]** Compare the performance of your best wide and deep network to a standard multi-layer perceptron (MLP) using the receiver operating characteristic and area under the curve for classification. Use Bland-Altman plots and residual variance calculation for regression. Use proper statistical method to compare the performance of different models.
- **Exceptional Work (10 points total)**
 - You have free reign to provide additional analyses.
 - One idea (**required for 7000 level students**): Capture the embedding weights from the deep network and perform t-SNE clustering on the output of these embedding layers. That is, pass the observations into the network, save the embedded weights (called embeddings), and then perform clustering of these output embeddings. Visualize and explain the results.