A univariate time series has only one feature. This feature also serves as a label. Examples of univariate time series problem include:

* Predict the daily minimum temperature based solely on the past minimum temperature readings.
* Predict the closing price of a stock solely based on the last few days of closing prices.

Assignment 4:

Data: <https://raw.githubusercontent.com/jbrownlee/Datasets/master/daily-min-temperatures.csv>

Your task is to leverage a dataset representing the minimum daily temperatures over a period of 10 years. Your goal is to predict the minimum temperature for the upcoming 10 years using two distinct neural network architectures: a Multilayer Neural Network (MLP) and a Multilayer Recurrent Neural Network (RNN). Following the model training, it is crucial to compare the predictive performance of these two models using suitable metrics, such as Mean Absolute Error (MAE) or Root Mean Squared Error (RMSE).

In addition to presenting the comparison results, provide a comprehensive paragraph discussing your intuition behind selecting a particular model for this time series data. Consider factors such as temporal dependencies and the nature of the data. Substantiate your intuition with the observed results and articulate why one model might be more effective than the other for forecasting minimum temperatures over time. This assignment aims to enhance your understanding of choosing appropriate neural network architectures for time series prediction tasks and interpreting their performance in a real-world context.

Grading Criteria:

1. Well Commented code
2. Data Visualization
   1. Visualization should effectively convey insights about the temperature data, patterns, and model predictions. Consider aspects such as appropriate choice of plots, clarity of labels, and insightful interpretation of visualized information.
3. Your understanding on how to implement RNN
   1. Assess the depth of the student's understanding in implementing RNN. This includes the correct application of RNN architecture to time series data, handling temporal dependencies, and appropriately configuring the network for the given task.

Overall, the assignment should showcase a balanced combination of well-commented code, insightful data visualization, and a thorough understanding of the implementation of RNNs for time series forecasting. Each criterion contributes to a comprehensive evaluation of the student's proficiency in both coding practices and conceptual understanding.

You can upload the notebook onto the Blackboard. Feel free to use any platform, not necessarily Kaggle.

Total Points: 100