Assignment: CNN and RNN

Problem 1 (20 points)

Choose a small ($< 3,000$) image dataset for classification. Include the link where you have downloaded the pictures from.

1. Train a model from scratch using what little data you have without any regularization, to set a baseline for what can be achieved.
   (a) Plot the loss and accuracy curves for training and validation sets.
   (b) Explain if you observe any overfitting and why.
   (c) Report the model accuracy on the test set.
2. Use data augmentation to generate more training data from your existing training samples. Also add a Dropout layer to your model, right before the densely connected classifier.
   (a) Plot the loss and accuracy curves for training and validation sets.
   (b) Explain if you observe any overfitting and why.
   (c) Report the model accuracy on the test set.
3. Use one of the pretrained image-classification models prepackaged in Keras trained on ImageNet dataset. The list includes Xception, Inception V3, ResNet50, VGG16, VGG19, and MobileNet. You can import these from the keras.applications module.

   Run the convolutional base (or only its first few layers) over your dataset, and then build a dense fully connected classifier on top of it.
   (a) Plot the loss and accuracy curves for training and validation sets.
   (b) Explain if you observe any overfitting and why.
   (c) Report the model accuracy on the test set.
4. Which one of the 3 models above would you choose as your final model? Why?

Problem 2 (80 points)

For this problem you use the data in "admData.csv" on Canvas. This file contains the accumulative number of admitted students to a certain program with 5 annual start dates.

The data has seasonal behavior: the accumulative number of admissions is *monotonically increasing* during the interval between 2 start dates and then it resets once a new term starts as shown in figure 1.

The objective of the problem is to predict the accumulative number of admissions 7, 14, 21, 28, 35, 42, 49, 56, 63, and 70 days from the current date.

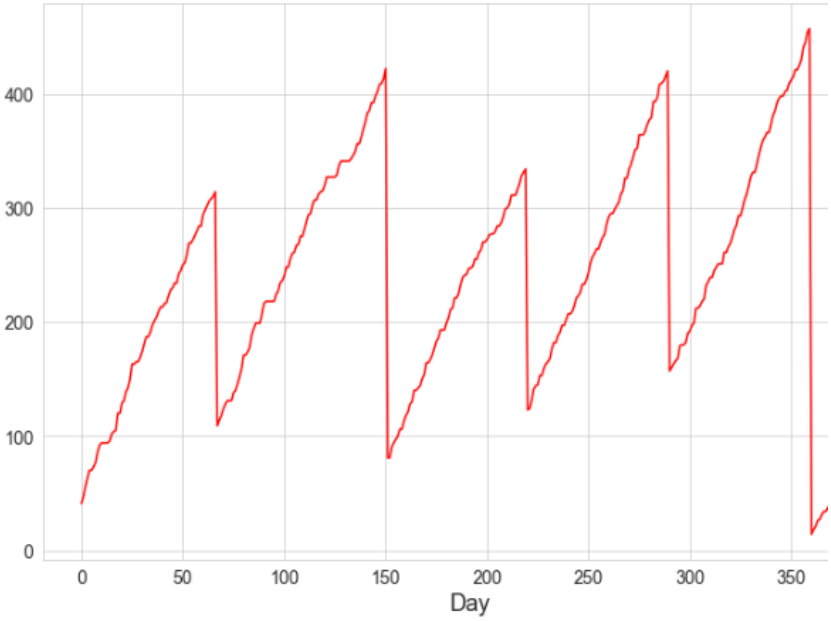

Figure 1: Number of Daily Accumulative Admissions

i.e. on any given day, you need to forecast what the accumulative number of admissions will be in 1 week, 2 weeks, ... 10 weeks from that day.

You need to use 70% of the data for training, 15% for validation, and 15% (the most recent) for test.

1. Create a recurrent neural network model. Explore both gru and lstm layers.
   (a) Report the MAE of the test set on your best model.
   (b) Plot the loss curves for training and validation sets.
2. Create a 1d convolutional network model. Explore if stacking lstm or gru layers helps with the performance.
   (a) Report the MAE of the test set on your best model.
   (b) Plot the loss curves for training and validation sets.
3. Use any machine learning model (Linear Regression, KNN, Random Forests, etc.) or an ensemble of models to achieve the lowest MAE.
   (a) Report the MAE of the test set on your best model.
   (b) Plot the loss curves for training and validation sets.

Make sure to upload the model (.hdf5 file for NN) on Canvas with any instruction needed for me to run your model/s on my test set.