Manufacturing Data Science 製造數據科學

Assignment 3

Due Date: Dec. 10, 2021

Please solve the following questions and justify your answer. Show all your analysis result including equation/calculation or Python code in your report. **Upload your "zip" file including MSWord/PDF report and Python code with 檔名: MDS_Assignment3_ID_Name.zip" to NTU COOL** by due. The late submission is not allowed.

1. **(25%)Nov. 12 資料科學應用案例演講**

(a) **(5%)** 該演講對您來說**印象深刻**的主題為何？為什麼？請摘要此深刻的主題。其對您來說帶來的啟發為何？

(b) **(5%)** 該演講是否有翻轉/顛覆您對過去製造業的**認知**？如果有，什麼認知有了改變？如果無，什麼樣的認知跟您過去的既定印象一樣，是否有任何建議或可改善之處？

(c) **(5%)** 該演講讓您瞭解到製造業應用數據科學方法的**困難與挑戰**在於何處？為什麼？如何建議或解決？

(d) **(5%)** 演講內容中，是否有任何**疑點**？或想問講者的問題為何？

(e) **(5%)** 在演講內容中，是否有任何想給予**建議**的地方？例如問題切入點、問題本質、方法調整、驗證的省思等。

2. **(40%)** Decision Tree Algorithms

Data Source: https://www.kaggle.com/uciml/faulty-steel-plates

Dataset provided by Semeion, Research Center of Sciences of Communication, Via Sersale 117, 00128, Rome, Italy. www.semeion.it

This dataset comes from research by Semeion, Research Center of Sciences of Communication. The original aim of the research was to correctly classify the type of surface defects in stainless steel plates, with six types of possible defects (plus "other"). The Input vector was made up of 27 indicators that approximately describe the geometric shape of the defect and its outline.

There are 1941 plates with 34 variables. The first 27 columns (i.e. independent variables) describe some kind of steel plate faults seen in images, i.e., X1-X27, as
{X_Minimum, X_Maximum, Y_Minimum, Y_Maximum, Pixels_Areas, X_Perimeter, Y_Perimeter
SumofLuminosity, MinimumofLuminosity, MaximumofLuminosity, LengthofConveyer, TypeOfSteel_A300, TypeOfSteel_A400, SteelPlateThickness, Edges_Index,Empty_Index, Square_Index, OutsideXIndex, EdgesXIndex, EdgesYIndex, OutsideGlobalIndex, LogOfAreas, LogXIndex, LogYIndex, Orientation_Index, Luminosity_Index, SigmoidOfAreas}

The last seven columns (i.e. dependent variables) are one hot encoded classes, i.e. if the plate fault is classified as "Stains" there will be a 1 in that column and 0's in the other columns. {Pastry, Z_Scratch, K_Scatch, Stains, Dirtiness, Bumps, Other_Faults}

These data can be found in http://archive.ics.uci.edu/ml/datasets/steel+plates+faults, and are attached in the file **MDS_Assignment3_Steelplates.xlsx**.

(a) **(5%)** Construct a data science framework and show the data summary
(b) **(5%)** What is the problem about the dataset? Any identical column? Any redundant column? Any missing value? How to handle these issues?
(c) **(5%)** After data preprocessing, based on the **prepared dataset**, use the classification and regression tree (CART) to analyze the prepared dataset. Show the classification results by 10-fold cross validation with several metrics (eg. accuracy, area under ROC curve (AUC), and F1-score), and also list the hyperparameters you adjust.
(d) **(5%)** Suggest a method to address the data imbalance issue. Build a new balanced dataset.
(e) **(5%)** Based on the **balanced dataset**, use the classification and regression tree (CART) to analyze the balanced dataset. Show the classification results by 10-fold cross validation with several metrics (eg. accuracy, area under ROC curve (AUC), and F1-score), and also list the hyperparameters you adjust.
(f) **(5%)** Give a comparison between (c) and (e). Any suggestion or insight?
(g) **(5%)** Use "Random Forest" to solve both prepared dataset and balanced dataset, respectively. Give a comparison and provide your insight.
(h) **(5%)** Use "Gradient Boosting Decision Tree (GBDT)" to solve both prepared dataset and balanced dataset, respectively. Give a comparison and provide your insight.

3. (20%) Deep Learning

Use Python to build up Convolutional Neural Network (CNN) for "**casting product image data for quality inspection**".
Data Source:
https://www.kaggle.com/ravirajsinh45/real-life-industrial-dataset-of-casting-product
Dataset provided by Pilot Technocast, Shapar, Rajkot https://pilottechnocast.com/

This dataset is of casting manufacturing product. Casting is a manufacturing process in which a liquid material is usually poured into a mould, which contains a hollow cavity of the desired shape, and then allowed to solidify. Casting defect is an undesired irregularity in a metal casting process. There are many types of defect in casting like blow holes, pinholes, burr, shrinkage defects, mould material defects, pouring metal defects, metallurgical defects, etc. Defects are an unwanted thing in casting industry. For removing this defective product all industry have their quality inspection department. But the main problem is this inspection process is carried out manually. It is a very time-consuming process and due to human accuracy, this is not 100% accurate. This can because of the rejection of the whole order. So it creates a big loss in the

company.

We decided to make the inspection process automatic and for this, we need to make deep learning classification model for this problem. These all photos are top view of submersible pump impeller (google search for better understanding). For capturing these images requires stable lighting, for this we made a special arrangement.
We focus on the **dataset with Augmentation**: the dataset contains total 7348 image data. These all are the size of (300*300) pixels grey-scaled images. In all images, augmentation already applied. Making classification model we already split data for training and testing into two folders. Both train and test folder contains deffront and okfront subfolders.
train:- deffront have 3758 and okfront have 2875 images
test:- deffront have:- deffront have 453 and ok_front have 262 images

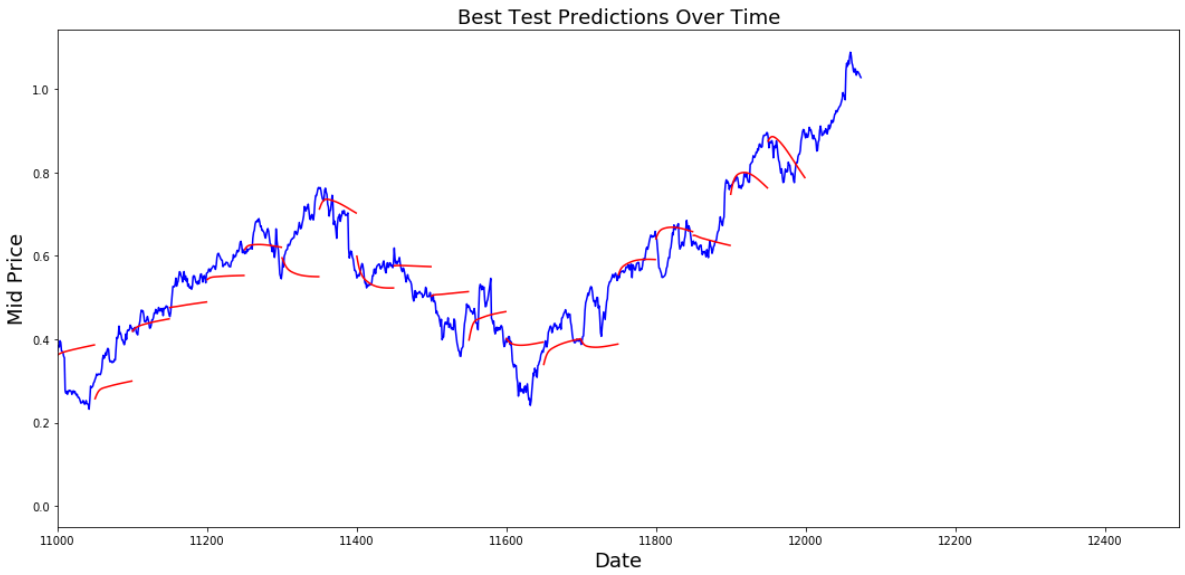
(The data set also includes the images size of 512x512 grayscale without Augmentation. This contains 519 okfront and 781 deffront impeller images. **We don't focus on this data set**.)

Any question, you can google it (keyword: casting product image data for quality inspection) or refer to the following linkage. https://www.youtube.com/watch?v=4sDfwS48p0A
If you would like to use Tensorflow, Keras, numpy, and pillow, you may refer to
https://www.kaggle.com/ginsaputra/visual-inspection-of-casting-products-using-cnn

(a) **(20%)** For CNN, try to investigate the effects of changing "**PARAMETERS**" such as learning rates, momentum, # of hidden/convolutional layers, dropout rate, etc. Show the numerical results and "**DIAGRAM**" from different perspectives (e.g., accuracy, F1-score, convergence time, error of training data, error of testing data, etc.). Please show all your work in detail, in particular, you "MAY" need to design your **experiments with different parameters** systematically.

4. **(15%) Time-Series Prediction**

Use Python to build up long short-term memory (LSTM), which is one type of recurrent neural network (RNN). Collect the dataset related to **weekly** raw material price **OR** consumption (i.e. demand). Build a price/demand forecast. Don't use STOCK PRICE for prediction. You may read the tutorial: https://www.datacamp.com/community/tutorials/lstm-python-stock-market. Note that, you only have price/demand data as response variable Y and it should be a **time-rolling prediction**, that is, for example, use the past 8 weeks dataset for 8-week ahead prediction. Thus, the prediction should be like the following diagram.



Dataset could be found as follows.

eg. Brent oil price: https://www.investing.com/commodities/brent-oil-historical-data

Commodity prices: https://fred.stlouisfed.org/categories/32217

Commodity prices: https://sdw.ecb.europa.eu/browse.do?node=9691219

The summary table of raw materials, https://just2.entrust.com.tw/z/ze/zeq/zeq.djhtm

Pick one raw material and collect its dataset. The collection period should be as long as possible (eg. from 2000 to 2020) to guarantee the sufficient samples for LSTM training.

(a) **(10%)** Prepare and transform the data to appropriate format (eg. use Data Generator in https://www.datacamp.com/community/tutorials/lstm-python-stock-market). Build LSTM model and show the prediction results via Time-series Nested Cross Validation.

(b) **(5%)** Visualize the time-rolling prediction as above diagram.

Note

1. Show all your work in detail. **Innovative** idea is encouraged.
2. If your answer refers to any external source, please "must" give an academic citation. Any "plagiarism" is not allowed.