COMP2019 Group Assignment

Due date: Monday, 8 November 2021, 11:59 PM
Weighting: 30% of the total course marks.

Instructions:

This assessment item must be conducted in groups. Students must work in the groups published on the course website. If there are any issues with your group, please let the teaching staff know early.

You will be training multiple machine learning models on a given dataset and recommend which model to use.

Submission instructions are at the end of this document.

All submitted materials must be your own work. If you adopt code that is not your own work, then you must clearly indicate the specific source of the material you used. Penalties apply for academic misconduct.

The word count for this assignment will not be verified. Please do not write excessively long answers (5k+ words). Excessively short answers, such as submitting only source code, charts or figures generated by your program without explanation, are unlikely to satisfy the marking criteria.

You must use only the Python libraries that have been used throughout the practical in this course: numpy, pandas, scikit-learn, Tensorflow/Keras, and the python 3.x standard libraries.

Assignment tasks

This assignment consists of five tasks. You will examine and prepare the given dataset (see Appendix), train multiple prediction models based on the data, and recommend which model to use based on your results.

Throughout this process, document your findings and assumptions.

You shall adopt the recommended practices discussed in the course. You will need to make decisions about data splitting, training and evaluation procedure, metrics for assessment, recommendation of a model, etc. Justify your decisions in the notebook.

For all tasks, train and assess the model using the correct datasets (train/dev/test or cross-validation).

Write your code and analysis in a single Google Colab/Jupyter notebook. Use headings to delineate your answers to each task in the notebook. Text contributions should be written in separate cells in Markdown format (not as comments interspersed with the source code). The notebook shall execute without errors when run from top to bottom.

Task 1. Review and Prepare the Dataset

Task 1a) Understand the Data

Review the data set to assess distributions of the features and the target and understand the relationships among the features and the target.

Document your findings.

Task 1b) Select a Metric

Choose a metric that you will use to assess the models.

Justify your selection of a metric.

Task 1c) Prepare the data sets

Partition the data for training and evaluation.

Use either a three-way split (train/dev/test) or cross-validation.

Justify the approach to partitioning the data.

Task 2. Train a Logistic Regression Model

Train a Logistic Regression Model (*LogisticRegression* in scikit-learn) that predicts the target.

You may need to either normalise the data or increase the model's hyperparameter *max_iter* to 2000 to achieve convergence of the optimiser.

Assess the model's quality of fit (bias/variance).

Task 3. Train a Decision Tree

Train a Decision Tree (*DecisionTreeClassifier* in scikit-learn) that predicts the target.

Optimise the model's quality of fit by tuning its *min_samples_leaf* hyperparameter. Justify the selection of the candidate values for the hyperparameter and describe how you identified its optimal value.

Assess the model's quality of fit (bias/variance).

Task 4. Train a Feed-Forward Neural Net

Train a Feed-Forward neural network that predicts the target. Use Tensorflow/Keras.

You will define your own network architecture. Use only *Dense* layers, only *ReLU* activation functions in the hidden layers, and a single output unit with *sigmoid* activation function in the output layer. Use *binary_crossentropy* as the loss function.

Aim to develop the simplest (fewest layers/units) that predicts no worse than the decision tree classifier created in Task 3.

Remember that normalisation of the input data can have a significant impact on the performance of a neural net model.

Describe the process and decisions that you have employed to arrive at the final neural net architecture. Justify the selection of the number of hidden layers and units per layer and explain the (iterative) process that you have followed while optimising the neural net. Explain how you have chosen the values for any hyperparameters that you may have set (such as batch size, number of epochs, learning rate).

Assess the model's quality of fit (bias/variance).

Task 5. Recommendation

Compare the results of the models trained in Task 2-4 and identify the best model. Which of the models exhibits best results?

Estimate the expected performance of the chosen model on unseen data.

Discuss the results.

Submission Instructions

Write all code and your analysis/recommendation in a single Google Colab (Jupyter) notebook.

List all group members and their contributions to this assignment at the top of the notebook.

Submit a single notebook for the entire group on learnonline. You must upload the ipynb file containing the notebook. Submissions in HTML format or links to Google Colab etc are not accepted.

It is sufficient if one group member submits the notebook on behalf of the entire group.

The submitted notebook shall contain all code, the output of each cell, and markdown cells showing discussion/analysis.

Present the tasks in the order given in this document.
Use the headings in this document as headings in the markdown sections in the notebook.

Do *not* include the instructions given in this document in the notebook.

Remove all obsolete and erroneous code from the notebook prior to submission.

Marking Criteria

Task	Marks
Task 1: Review and Prepare the Dataset Distributions of data set analysed Correlations in data analysed Metric selected and justified	15
Task 2: Train LR Model Correct data split and training process applied Quality of fit analysed Correct evaluation process followed Analysis of results documented	15
Task 3: Train DT Model Correct data split and training process applied Correct optimisation process applied and justified Quality of fit analysed Correct evaluation process followed Analysis of results documented	20
Task 4: Train NN Model Appropriate network architecture defined Development process and decisions explained Quality of fit analysed Correct evaluation process followed Analysis of results documented	35
Task 5: Recommendation Clear recommendation given Recommendations justified using outcomes of tasks 2-4 Correct evaluation process followed Analysis of evaluation results documented	15
Notebook Team members details included Contributions by each team member listed Uploaded in the correct format (.ipynb) Notebook runs without errors Notebook does not contain irrelevant code Tasks addressed in the given order Appropriate headings given Uses Markdown cells for discussion Free of grammar and spelling errors	Deductions apply if criteria are not met

Marks may be adjusted to reflect (non-)contributions by team members.

Appendix – Data Set

The data set represents results from a customer satisfaction survey of airline passengers. It comprises 25 features and aims to predict the level of satisfaction ('neutral or dissatisfied' vs 'satisfied').

The data set has been split into a training set and a test set. Each is available as a separate file in CSV format (train.csv, test.csv).

Data dictionary

- Gender: Gender of the passengers (0 - Male, 1 - Female)
- Customer Type: The customer type (0 - Disloyal customer, 1 - Loyal customer)
- Age: The actual age of the passengers in years
- Type of Travel: Purpose of the flight of the passengers (0 - Personal Travel, 1 - Business Travel)
- Class: Travel class in the plane of the passengers (0 – Eco, 1 – Eco+, 2 - Business)
- Flight distance: The flight distance of this journey in miles
- Inflight wifi service: Satisfaction level of the inflight wifi service (0-5)
- Departure/Arrival time convenient: Satisfaction level of Departure/Arrival time convenient (0-5)
- Ease of Online booking: Satisfaction level of online booking (0-5)
- Gate location: Satisfaction level of Gate location (0-5)
- Food and drink: Satisfaction level of Food and drink (0-5)
- Online boarding: Satisfaction level of online boarding (0-5)
- Seat comfort: Satisfaction level of Seat comfort (0-5)
- Inflight entertainment: Satisfaction level of inflight entertainment (0-5)
- On-board service: Satisfaction level of On-board service (0-5)
- Leg room service: Satisfaction level of Leg room service (0-5)
- Baggage handling: Satisfaction level of baggage handling (0-5)
- Check-in service: Satisfaction level of Check-in service (0-5)
- Inflight service: Satisfaction level of inflight service (0-5)
- Cleanliness: Satisfaction level of Cleanliness (0-5)
- Departure Delay in Minutes: Minutes delayed on departure (0-5)
- Arrival Delay in Minutes: Minutes delayed on arrival (0-5)
- Order: Aggregate score of satisfaction with ordering/arrival (0-5)
- Comfort: Aggregate score of comfort (0-5)
- Service: Aggregate score of service (0-5)
- Satisfaction: Airline satisfaction level (0 – Neutral or dissatisfied; 1 - Satisfied). This is the target to predict.