Course name: Introduction to Machine Learning

Course code: 822047-B-6

Take-home Assignment
Academic Year 2020-2021

Sharon Ong and Emmanuel Keuleers
Department of Cognitive Science and Artificial Intelligence
Tilburg University

The Individual take-home assignment is worth 100 points and is 40% of your grade. A pass is not required for this component to pass the course as long as the total grade for all components is 5.5 or higher. Because a pass is not required, no resit will be provided for the whole class.

**Late submission policy**: There is a 5% penalty per day for late submissions. Zero credit is earned if submitted after 7 days past the due date. Start early, most assignments will take longer than you expect.

The assignment consist of two parts. You will submit code and a report in a zip file. A template for the report will be provided. For each part of the assignment, you should include the following:

1. A summary of your methods and findings. If your method performed poorly on the data then you may want to include an explanation as to why and suggest how your method may be improved.
2. References to any code, methods, or ideas that you used that are not your own. Remember, these must be publicly available and free to use.
3. Any special instructions that are required to run your code.

# 1 Classification of Facial Expressions

You will develop a program to implement solutions on a real-world dataset. You will use Python libraries for building and evaluation models and learning parameters. We will use a subset of the Karolinska DirectedEmotional Faces (KDEF) dataset, the Yale Faces dataset and the Toronto face dataset. These datasets are a set pictures of human facial expressions of emotion which you will use for training, validation and testing.

## 1.1 Dataset

You will each receive individual npz files with different subsets of the original dataset (**XXXXXXX_face.npz**, where XXXXXXX is your student ID). Your individual dataset comprises of grayscale images that contain faces that have been extracted from a variety of sources. The faces have been rotated, scaled and aligned to make the task easier. These faces have been labeled by experts and research assistants based on their expression. These expressions fall into one of three categories, labelled **0**, **1** and **2**. Note that each labelled face image has an identity, and there are different expressions associated with each person in the labelled set. Therefore to avoid having one person's face in both training, validation and test sets, cross validation should not be used. The data set has been partitioned into training, testing and validation for you.

## 1.2 Exploratory Data Analysis Tasks

1. (5 points) Using a subplot, display three images from the dataset, one from each category. Label these images according to their category.
Marks breakdown:

- 3 points for displaying each image with labels in a subplot.
- 2 points for automatically finding which indexes in your data belong to each label.

2. (5 points) Combine the target labels together in one array. Display your target class in a histogram. Is the dataset balanced?
   Marks breakdown:
   - 1 points for combining the target labels.
   - 3 points for display the histogram of the target classes with axis labelled and title included.
   - 1 points for whether the dataset is balanced.

### 1.3 Classification Tasks

1. (7 points) Train a k-nearest neighbour classifier ($k = 1$) with uniform weights. This classifier will be your baseline classifier. Marks breakdown:
   - 3 points for training and evaluating the accuracy of the KNN (with k = 1).
   - 2 points for displaying two mis-classified images for each class.
   - 1 points for providing other metrics (besdies accuracy) to evaluate this classifier.
2. (3 points) Find the best hyperparameters (number of neighbours) for the knn classifier.
3. (45 points) Create and train classifiers and apply preprocessing methods to beat the baseline classifier. You can compare many machine learning approaches including supervised learning methods (k-nearest neighbours, SVMs, kernel SVMs, etc ) and dimensionality reduction (PCA, etc..). Your solution does not have to be limited to simply applying machine learning models, as well.
   Marks breakdown:
   - 15 points for comparing at least 4 other (low-level) machine learning algorithms, searching for best hyperparameters and beating the baseline classifier. Choose one or two hyperparameters (if there are any) to search for. (2 points per algorithm, 1 point for finding the best hyperparameters (if any) for each algorithm, 1 point for beating the baseline classifier and 2 points for clean well structured code.)
   - 5 points for training and evaluating the models with Random Forests or Gradient Boosting.
   - 5 points for training and evaluating with a heterogeneous ensemble learner such as Voting.
   - 5 points for evaluating all classifiers with additional metrics (beyond accuracy), comparing the computational times of each classifier and describing the results in a table in your report.
   - 5 points for two dimensional reduction or feature selection techniques. You should justify the number of components selected.
   - 5 points for balancing the dataset with an appropriate data balancing technique.
   - 5 points for implementing additional solutions such as streamlining your code with functions or pipelines.

## 2 Regression to estimate the width of a grey kangaroo's nose

You will create code to perform regression on a dataset from Australian Journal of Zoology, Vol. 28, p607-613 with Python libraries.

### 2.1 Dataset

The dataset consists of two continuous variables; the nasal length and nasal width (in mm) for male grey kangaroos.

## 2.2 Exploratory Data Analysis Tasks

1. (3 points) Present a scatterplot displaying the relationship between these two variables. Label the x axis as nasal length (mm) and y axis as nasal width (mm). Add a title to the scatterplot.

## 2.3 Regression Tasks

1. (2 points) Split your dataset into 90% training and 10% testing.
2. (5 points) Linear fitting: Fit a linear regression model to the data. Evaluate how successful the fit is by computing the $R^2$ score.
3. (3 points) Fit the linear regression model with cross validation and evaluate the mean $R^2$ score.
4. (10 points) Using Grid Search and Cross Validation, compare the performance of regression analysis with SVM regression, and Decision Tree regression using the $R^2$ score as a metric. Display your solutions in a plot.
5. (2 points) Select the best performing regression model.

## 2.4 Missing Data Imputation Task

1. (10 points) In your individual file XXXXXXX_nose.csv (where XXXXXXX is your student ID), some variables are labelled with NaN. There variables are considered missing data. Perform a mean and a kNN (k = 3) data imputation. Compare the performance of a linear regression model with the two different data imputations.