ECE449 Lab 2

September 20th, 2023

Lab 2

- Download the wine dataset from https://archive.ics.uci.edu/dataset/109/wine

- A classification problem with 3 classes.

- Design a pipeline that preprocess the data using Minmax scaling, and replaces NAN values with the mean value of each class ➔ from lab 1

Lab 2

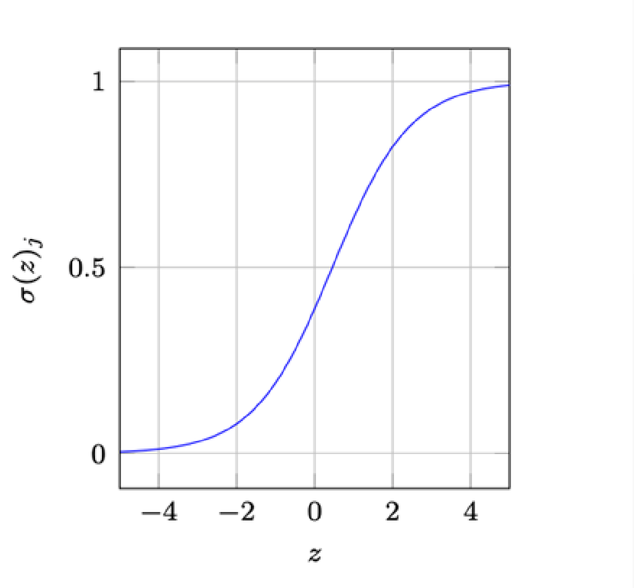
- Divide the data to 10% test and 90% train using the stratified sampling

- One-hot encode each class
- Class 1: 1 0 0
- Class 2: 0 1 0
- Class 3: 0 0 1
- You can use➔ from sklearn.preprocessing import OneHotEncoder

Lab 2

- Use a Softmax activation in your last layer to train the network to recognize this encoding.

- Converts a vector of K real numbers into a probability distribution of K possible outcomes.



from https://www.nomidl.com/deep-learning/what-is-the-difference-between-sigmoid-and-softmax-activation-function/

Lab 2

- Perform a parameter exploration on the “training” dataset using the tenfold cross-validation design (again, using stratified sampling).

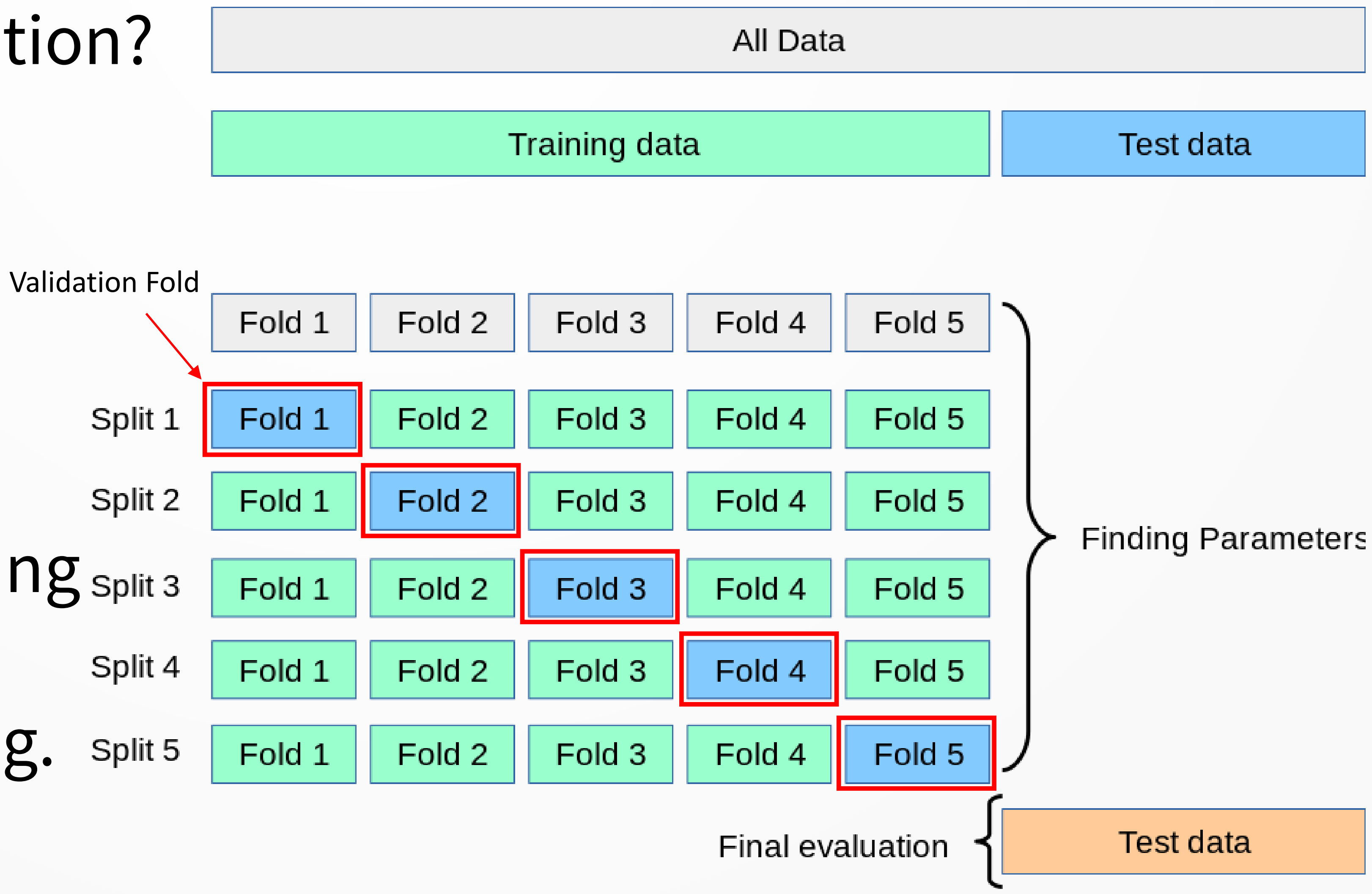
- You can use this for stratified cross-validation ➔

from sklearn.model_selection import StratifiedKFold

Lab 2

- What is cross validation?

- Find the best parameters for your dataset while reducing the bias towards a specific data splitting.



from https://scikit-learn.org/stable/modules/cross_validation.html

Lab 2

- Parameter exploration must include number of layers, neurons per layer, learning rate.
- Must at least try the following parameters ➔
  - ❑ Number of layers = [1, 2, 3]
  - ❑ Number of neurons = [32, 64, 128]
  - ❑ Learning rate = [0.001, 0.01, 0.1]
- F1 score should be used for performance evaluation.

Lab 2

- Create your own pipeline by passing the output of one library function to the next, and adding in the control flow needed to organize the tenfold cross-validation experiments.

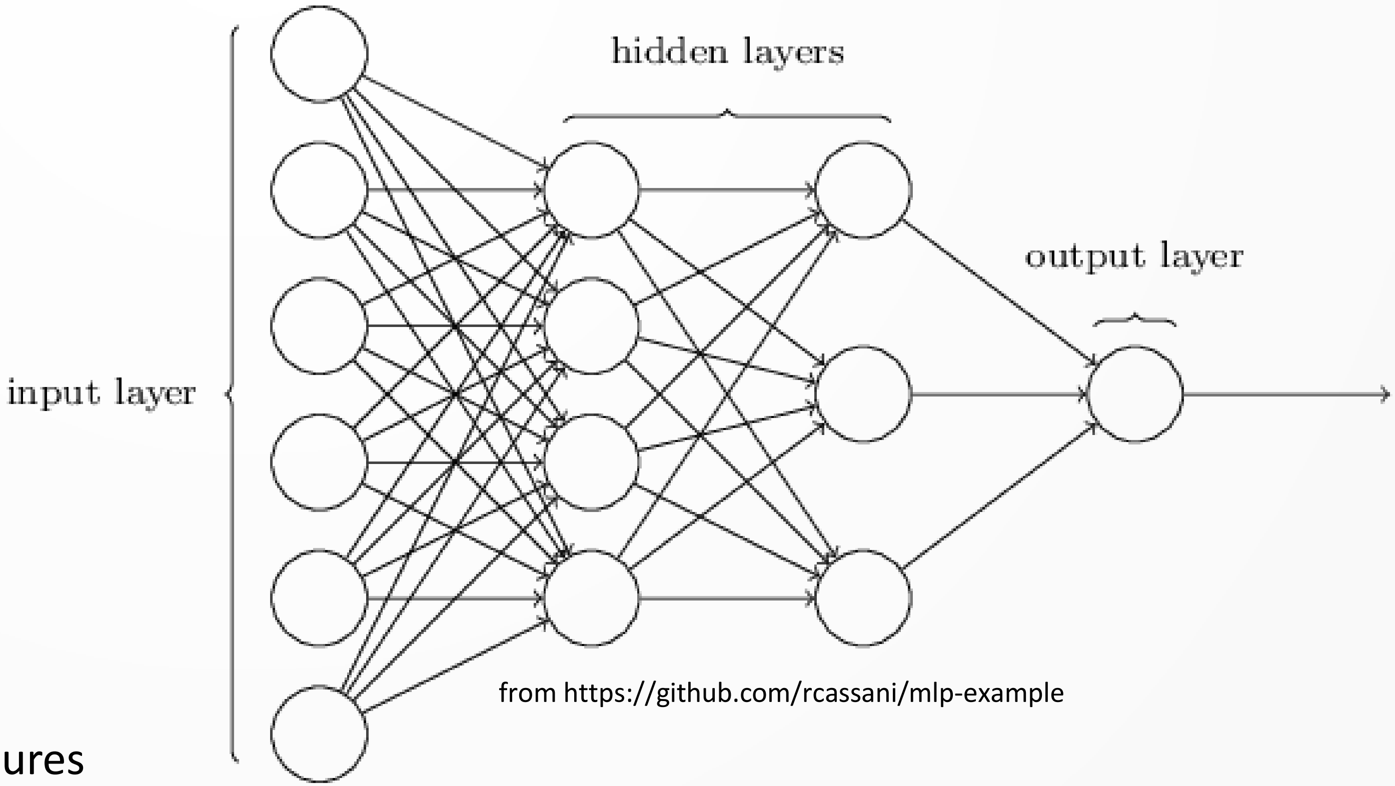
Lab 2

- Employ the multi-layer perceptron (MLP) network, using the backpropagation with momentum learning algorithm, to solve the Wine Dataset classification problem.

- Create your MLP with Tensorflow and Keras.

Lab 2

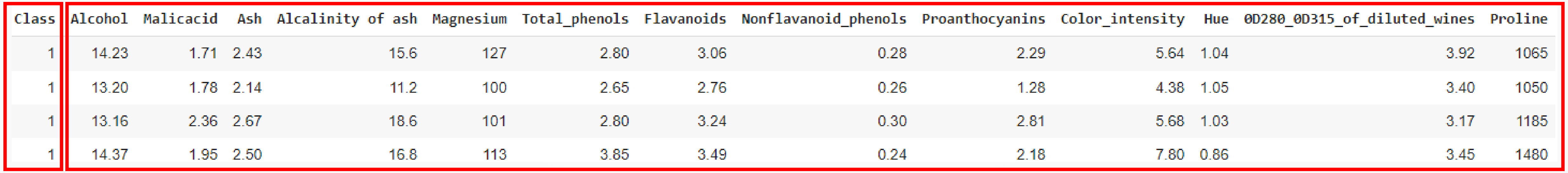
- An MLP structure



from https://github.com/rcassani/mlp-example

Output

Input Features



Class	Alcohol	Malicacid	Ash	Alcalinity of ash	Magnesium	Total_phenols	Flavanoids	Nonflavanoid_phenols	Proanthocyanins	Color_intensity	Hue	0D280_0D315_of_diluted_wines	Proline
1	14.23	1.71	2.43	15.6	127	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065
1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050
1	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185
1	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480

Lab 2

- from tensorflow import keras

- You can create a neural network by ➔
- model = keras.Sequential()
- Add more layers to this neural network by ➔
- model.add(keras.layers.Dense(neurons, activation='relu’))

Lab 2

- Compile this neural network by ➔
- model.compile(optimizer=keras.optimizers.Adam(learning_rate=learning_rate),loss='categorical_crossentropy’, metrics=['accuracy’])

- The model is trained using the .fit function but it should be inside a pipeline.

Lab 2

- After you find the best parameters, train your network using the best parameters on the whole training dataset.

- Report your F1 score.
- If needed, you can use the f1-score function from ➔ from sklearn.metrics import f1_score