Health_and_eating_dataset

Notebook link: ML_healthy_eating.ipynb

Brief description of each dataset and tasks

- ***Description***: This dataset is about what makes a dish healthy. The data includes information such as: Fat, Sugar, Calories, Cooking method,...
- ***Tasks***: Our task is to build a model to predict if a dish is healthy or not based on the provided features.

Summary of model architectures and training strategies

a. Model architecture:

- The model architectures I used were 2 **Relu** layers, 1 **Dropout** layer, and 1 **sigmoid** layer.

➔ The reason why I used this model architecture is that:
  - RELU: Because it is fast and safe
  - Dropout: As mentioned in class, Dropout might make the learning process more efficient by creating more difficulties for the model
  - Softmax: Because our output is binary

b. Training Strategies:

- My approach was to clean all the data, followed by splitting the train and the test set. Then I did the preprocessing process before actually training the model, and finally ended with validating and testing the model. Along the way, I did add EarlyStopping to make sure the learning process was 'safe'. Specifically in this dataset, I used class_weight to help my model focus more on the minority, which is very significant in this dataset, where it heavily shifted to unhealthy.

Comparative analysis of performance and feature importance

a. Analysis of performance:

- The model stopped at epoch 26, with loss: 0.0639 - precision: 0.6624 - recall: 1.0000 - val_loss: 0.1989 - val_precision: 0.5152 - val_recall: 0.6071
- Test loss : 0.1801
- Test precision: 0.6047
- Test recall : 0.7027

Confusion Matrix:

[[346 17]
 [11 26]]

Classification Report:

```
              precision    recall  f1-score   support

         0.0       0.97      0.95      0.96       363
         1.0       0.60      0.70      0.65        37

    accuracy                           0.93       400
   macro avg       0.79      0.83      0.81       400
weighted avg       0.94      0.93      0.93       400
```

➔ Despite using class weight to focus more on the minority, my model still performs very bad with healthy food.

b. Feature Importance:

Feature Importance Table:

	Feature	Importance
3	minmaxscaler__fat_g	0.295380
5	minmaxscaler__sugar_g	0.271978
0	minmaxscaler__calories	0.211739
34	onehotencoder__cooking_method_Raw	0.154545
44	onehotencoder__meal_Wrap	0.138199
15	onehotencoder__cuisine_Italian	0.136895
42	onehotencoder__meal_Soup	0.136801
31	onehotencoder__cooking_method_Boiled	0.135203
24	onehotencoder__diet_type_Balanced	0.135168
43	onehotencoder__meal_Stew	0.133315
37	onehotencoder__meal_Curry	0.132092
41	onehotencoder__meal_Sandwich	0.129632
33	onehotencoder__cooking_method_Grilled	0.129111

27	onehotencoder__diet_type_Paleo	0.128861
7	minmaxscaler__cholesterol_mg	0.128311
25	onehotencoder__diet_type_Keto	0.127910
10	minmaxscaler__cook_time_min	0.127676
18	onehotencoder__cuisine_Mexican	0.126329
26	onehotencoder__diet_type_Low-Carb	0.126240
6	minmaxscaler__sodium_mg	0.124831
17	onehotencoder__cuisine_Mediterranean	0.124563
28	onehotencoder__diet_type_Vegan	0.122959
29	onehotencoder__diet_type_Vegetarian	0.122251
2	minmaxscaler__carbs_g	0.121748
20	onehotencoder__meal_type_Breakfast	0.121134
32	onehotencoder__cooking_method_Fried	0.119973
40	onehotencoder__meal_Salad	0.119619
9	minmaxscaler__prep_time_min	0.118990
39	onehotencoder__meal_Rice	0.118852
1	minmaxscaler__protein_g	0.118366
22	onehotencoder__meal_type_Lunch	0.118272
23	onehotencoder__meal_type_Snack	0.118205
14	onehotencoder__cuisine_Indian	0.117720
36	onehotencoder__cooking_method_Steamed	0.116033
13	onehotencoder__cuisine_Chinese	0.115148
19	onehotencoder__cuisine_Thai	0.114427
35	onehotencoder__cooking_method_Roasted	0.112300

11	minmaxscaler__rating	0.112236
30	onehotencoder__cooking_method_Baked	0.111803
8	minmaxscaler__serving_size_g	0.110121
12	onehotencoder__cuisine_American	0.109715
4	minmaxscaler__fiber_g	0.109225
16	onehotencoder__cuisine_Japanese	0.107582
21	onehotencoder__meal_type_Dinner	0.107279
38	onehotencoder__meal_Pasta	0.100040

- The features are somewhat similar in terms of importance in this model.

Insights into what you discovered in your experiments

- Different usage of metrics in different cases. For example, my first approach was to use accuracy, but when I thought deeply about it, accuracy was not a good choice to evaluate in this situation, especially in cases where the label is so shifted to one specific character.
- I can not fully rely on the model and let it learn by itself. In my first attempt, I did not use class weight to make the model focus on the minority value (1). This led to a very bad result, which was the reason why I approached it in this way. But in the end, it is still very vulnerable with a minority label, so there must be a better way to do this.