Review Test Submission: SelfCheck: L10-Machine learning (Advanced)

User	Rakshit Yadav
Course	BMAN73701 Programming in Python for Business Analytics 2023-24 1st Semester
Test	SelfCheck: L10-Machine learning (Advanced)
Started	29/11/23 10:13
Submitted	29/11/23 10:17
Status	Completed
Attempt Score	20 out of 50 points
Time Elapsed	4 minutes
Results Displayed	All Answers, Submitted Answers, Correct Answers, Feedback

Question 1

0 out of 10 points

Examples of hyper-parameter optimization are (tick all that apply):

Selected Answers:

✓ Finding the number of trees in a random forest model using cross-validation

✓ Finding the optimal number of hidden layers of a neural network

✗ Finding the optimal values of the weights of a neural network given training data

Answers:

Learning the structure of the trees of a random forest model

Optimizing the parameters of a model

✓ Finding the number of trees in a random forest model using cross-validation

← OK

✓ Finding the optimal number of hidden layers of a neural network

Finding the optimal values of the weights of a neural network given training data

Response Feedback: Incorrect! Remember that ML models have both parameters that are learned during training and hyper-parameters that are set before learning and must be optimized manually or using cross-validation.

Question 2

0 out of 10 points

When doing hyper-parameter optimization, the score that should be optimized is...

Selected Answers: ✗ the score on the test data

Answers:

✓ the cross-validation score

the training score

the score on the test data

the optimization score

Response Feedback: Incorrect! See the lectures slides and the examples there.

Question 3

0 out of 10 points

The ".score()" method of an ML model in scikit-learn returns

Selected Answers:

✗ Always accuracy

✗ It depends on the object, X.score() returns accuracy if X is a classifier, R^2 if it is a regression model, optimization scores if X is GridSearchCV() or RandomSearchCV(), and cross_val_score() returns cross-validation scores.

Answers:

✓ It depends on the model, most classifiers return accuracy and most regression models return R^2.

Always accuracy

Always f1-score

Always R^2

It depends on the object, X.score() returns accuracy if X is a classifier, R^2 if it is a regression model, optimization scores if X is GridSearchCV() or RandomSearchCV(), and cross_val_score() returns cross-validation scores.

Response Feedback:	Not quite correct. You need to revise the lectures more carefully.

Question 4

10 out of 10 points

```
import pandas as pd
# Read data
df = [a].read_csv("data.csv")
# Separate train and test
X_train, y_train, X_test, y_test = [b](df.drop("target", axis=1), df[c])
rf = RandomForestClassifier()
# Train
rf.[d]([e],[f])
# Score on test data
rf.[g]([h],[i])
```

Specified Answer for: a	✓ pd
Specified Answer for: b	✓ train_test_split
Specified Answer for: c	✓ ["target"]
Specified Answer for: d	✓ fit
Specified Answer for: e	✓ X_train
Specified Answer for: f	✓ y_train
Specified Answer for: g	✓ score
Specified Answer for: h	✓ X_test
Specified Answer for: i	✓ y_test

Correct Answers for: a		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	pd	Case Sensitive
Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	train_test_split	Case Sensitive
Correct Answers for: c		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	["target"]	Case Sensitive
✓ *Exact Match*	['target']	Case Sensitive

Correct Answers for: d		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	fit	Case Sensitive
Correct Answers for: e		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	X_train	Case Sensitive
Correct Answers for: f		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	y_train	Case Sensitive
Correct Answers for: g		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	score	Case Sensitive
Correct Answers for: h		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	X_test	Case Sensitive
Correct Answers for: i		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	y_test	Case Sensitive

Response Feedback: All correct!

Question 5

10 out of 10 points

In sklearn, each ML model has its own score() method and we should always use that one to evaluate the results of the model.

Selected Answer: ✓ False. Although it is true that each ML model has its own score method, we can use other metrics (such as precision, recall, f1-score, etc, in the case of classifiers) to evaluate the results.

Answers: ✓ False. Although it is true that each ML model has its own score method, we can use other metrics (such as precision, recall, f1-score, etc, in the case of classifiers) to evaluate the results.

True because the score() method has been chosen carefully for each model and we cannot change it.

False, all score() methods calculate the same metric

True, because using other metrics is difficult.

Response Feedback: Good!

Wednesday, 29 November 2023 11:00:52 o'clock GMT