

- ▶ Welcome
- Introduction: Machine Learning concepts
- Module 1. The Predictive Modeling Pipeline
- Module 2.Selecting the best model
- Module 3.Hyperparameter tuning

Module overview

Manual tuning

Quiz M3

Automated tuning

Quiz M3

Wrap-up quiz

Wrap-up quiz

Main take-away

- Module 4. Linear Models
- Module 5.Decision tree models
- ► Module 6. Ensemble of

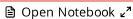


>

In this wrap-up quiz you will need to write some code in order to answer quiz questions:

- an empty notebook is available just below to write your code
- quiz questions are located after the notebook here
- the button Open Notebook at the bottom right of the screen allows you to open the notebook in full page at any time

+ Click here to see a demo video of the notebook user interface





- Module 7.
 Evaluating model
 performance
- Conclusion
- Appendix

INIOUUIE 3 - WIAP-UP WUIZ

3 **-** VVI ALIBULII \31117

Importing Data

```
In [1]: import pandas as pd

penguins = pd.read_csv("../datasets/penguins.csv")

columns = ["Body Mass (g)", "Flipper Length (mm)", "Culm target_name = "Species"

# Remove lines with missing values for the columns of in penguins_non_missing = penguins[columns + [target_name]]

data = penguins_non_missing[columns]
target = penguins_non_missing[target_name]
```

Checking Target

Checking Features



Load the dataset file named penguins.csv with the following command:



7



```
penguins = pu.reau_csv( ../qatasets/penguins.csv )

columns = ["Body Mass (g)", "Flipper Length (mm)", "Culmen
Length (mm)"]
target_name = "Species"

# Remove lines with missing values for the columns of interest
penguins_non_missing = penguins[columns +
[target_name]].dropna()
```

penguins is a pandas dataframe. The column "Species" contains the target variable. We extract the numerical columns that quantify some attributes of such animals and our goal is try to predict their species based on those attributes stored in the dataframe named data.

Inspect the loaded data to select the correct assertions:

data = penguins_non_missing[columns]

target = penguins_non_missing[target_name]

Question 1 (1/1 point)

Inspect the target variable and select the correct assertions from the following proposals.

- a) The problem to be solved is a regression problem
- O b) The problem to be solved is a binary classification problem (exactly 2 possible classes)
- c) The problem to be solved is a multiclass classification problem (more than 2 possible classes)

Hint: target.nunique() is a helpful method to answer to this question.

You have used 1 of 1 submissions

Question 2 (1/1 point)

Inspect the statistics of the target and individual features to select the correct statements.





☑ b) The proportions of the class counts are imbalanced: some classes have more than twice as many rows than others

☐ c) The input features have similar scales (ranges of values)



Select all answers that apply

Hint: data.describe(), and target.value_counts() are methods that are helpful to answer to this question.

You have used 1 of 2 submissions

Question 3 (1/1 point)

Let's now consider the following pipeline:

Evaluate the pipeline using stratified 10-fold cross-validation with the balanced-accuracy scoring metric to choose the correct statement in the list below.

You can use:

- sklearn.model_selection.cross_validate to perform the cross-validation routine;
- provide an integer 10 to the parameter cv of cross_validate to use the cross-validation with 10 folds;
- provide the string "balanced_accuracy" to the parameter scoring of cross_validate .
- a) The average cross-validated test balanced accuracy of the above pipeline is between 0.9 and 1.0



\circ	c) The average cross-validated test balanced accuracy of the above
nin	eline is between 0.5 and 0.8

You have used 1 of 1 submissions

Question 4 (1/1 point)

Repeat the evaluation by setting the parameters in order to select the correct statements in the list below. We recall that you can use <code>model.get_params()</code> to list the parameters of the pipeline and use <code>model.set_params(param_name=param_value)</code> to update them. Remember that one way to compare two models is comparing the cross-validation test scores of both models fold-to-fold, i.e. counting the number of folds where one model has a better test score than the other.

- a) Looking at the individual cross-validation scores, using a model with $n_{neighbors=5}$ is substantially better (at least 7 of the cross-validations scores are strictly better) than a model with $n_{neighbors=51}$
- b) Looking at the individual cross-validation scores, using a model with n_neighbors=5 is substantially better (at least 7 of the cross-validations scores are strictly better) than a model with n_neighbors=101
- c) Looking at the individual cross-validation scores, a 5 nearest neighbors using a StandardScaler is substantially better (at least 7 of the cross-validations scores are strictly better) than a 5 nearest neighbors using the raw features (without scaling).



Select all answers that apply

You have used 2 of 2 submissions

Question 5 (1/1 point)

We will now study the impact of different preprocessors defined in the list ূ্ৰ below:



```
all_preprocessors = [
   None,
   StandardScaler(),
   MinMaxScaler(),
   QuantileTransformer(n_quantiles=100),
   PowerTransformer(method="box-cox"),
]
```

The Box-Cox method is common preprocessing strategy for positive values. The other preprocessors work both for any kind of numerical features. If you are curious to read the details about those method, please feel free to read them up in the preprocessing chapter of the scikit-learn user guide but this is not required to answer the guiz questions.

Use sklearn.model_selection.GridSearchCV to study the impact of the choice of the preprocessor and the number of neighbors on the stratified 10-fold cross-validated balanced_accuracy metric. We want to study the n_neighbors in the range [5, 51, 101] and preprocessor in the range all_preprocessors.

Which of the following statements hold:

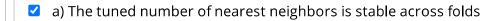
- a) Looking at the individual cross-validation scores, the best ranked model using a StandardScaler is substantially better (at least 7 of the cross-validations scores are strictly better) than using any other preprocessor
- b) Using any of the preprocessors has always a better ranking than using no preprocessor, irrespective of the value of n_neighbors
- c) Looking at the individual cross-validation scores, the model with n_neighbors=5 and StandardScaler is substantially better (at least 7 of the cross-validations scores are strictly better) than the model with n_neighbors=51 and StandardScaler





the cross-validations scores are strictly better) than the model with
n_neighbors=101 and StandardScaler
✓
Select all answers that apply Hint: pass
<pre>{"preprocessor": all_preprocessors, "classifiern_neighbors": [5, 51, 101]</pre>
for the param_grid argument to the GridSearchCV class.
You have used 1 of 2 submissions
Question 6 (1/1 point)
Evaluate the generalization performance of the best models found in each fold using nested cross-validation. Set return_estimator=True and cv=10 for the outer loop. The scoring metric must be the balanced-accuracy.
The mean generalization performance is :
O a) better than 0.97
o b) between 0.92 and 0.97
O c) below 0.92
You have used 1 of 1 submissions
Question 7 (1/1 point)
Explore the set of best parameters that the different grid search models found in each fold of the outer cross-validation. Remember that you can access them with the best_params_ attribute of the estimator. Select all the

statements that are true.

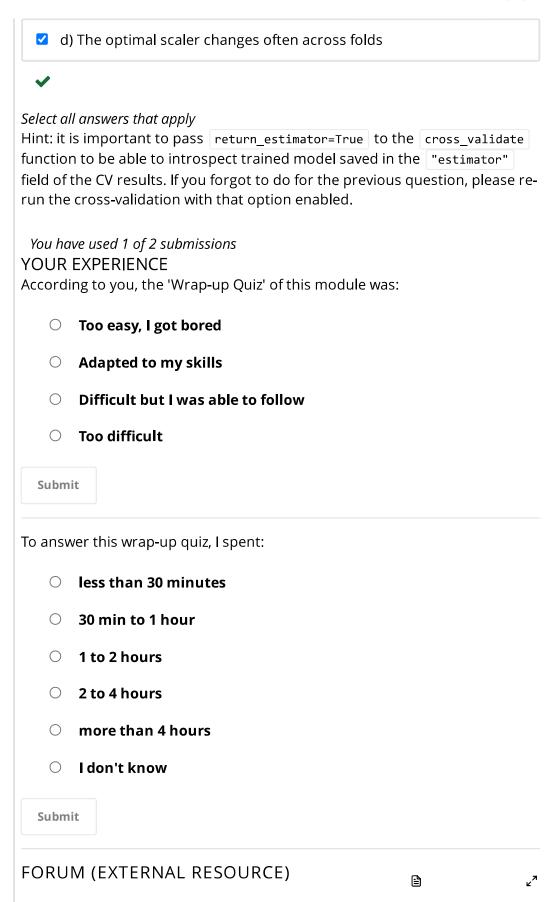


 $\hfill \Box$ b) The tuned number of nearest neighbors changes often across folds















Home > M3. Hyperparameter tuning > M3. Wrap-up quiz 3

Ļ

There are no more M3. Wrap-up quiz 3 topics. Ready to start a new conversation?

About...

Help and Contact

Terms of use

Usage policy









