Wrap-up quiz 5

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

Open Notebook

Module 5 - Wrap-up Quiz

Importing Data

In [5]:
```
import pandas as pd

ames_housing = pd.read_csv(
    "../datasets/ames_housing_no_missing.csv",
    na_filter=False,  # required for pandas>2.0
)
target_name = "SalePrice"
data = ames_housing.drop(columns=target_name)
target = ames_housing[target_name]
```

Selecting Numerical Features

In [6]:
```
numerical_features = [
    "LotFrontage", "LotArea", "MasVnrArea", "BsmtF
    "BsmtUnfSF", "TotalBsmtSF", "1stFlrSF", "2ndFl
    "GrLivArea", "BedroomAbvGr", "KitchenAbvGr", "
    "GarageCars", "GarageArea", "WoodDeckSF", "Ope
    "3SsnPorch", "ScreenPorch", "PoolArea", "MiscV
]

data_numerical = data[numerical_features]
```

Building Models

Linear Model

In [7]:
```
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import make_pipeline
from sklearn.linear_model import LinearRegression

linear_model = make_pipeline(StandardScaler(), Lin
```

Powered by OVHcloud
Powered by OVHcloud

Open the dataset `ames_housing_no_missing.csv` with the following command:

```
ames_housing = pd.read_csv(
    "../datasets/ames_housing_no_missing.csv",
    na_filter=False,  # required for pandas>2.0
)
target_name = "SalePrice"
data = ames_housing.drop(columns=target_name)
target = ames_housing[target_name]
```

`ames_housing` is a pandas dataframe. The column "SalePrice" contains the target variable.

To simplify this exercise, we will only used the numerical features defined below:

```
numerical_features = [
    "LotFrontage", "LotArea", "MasVnrArea", "BsmtFinSF1",
"BsmtFinSF2",
    "BsmtUnfSF", "TotalBsmtSF", "1stFlrSF", "2ndFlrSF", "LowQualFinSF",
    "GrLivArea", "BedroomAbvGr", "KitchenAbvGr", "TotRmsAbvGrd",
"Fireplaces",
    "GarageCars", "GarageArea", "WoodDeckSF", "OpenPorchSF",
"EnclosedPorch",
    "3SsnPorch", "ScreenPorch", "PoolArea", "MiscVal",
]

data_numerical = data[numerical_features]
```

We will compare the generalization performance of a decision tree and a linear regression. For this purpose, we will create two separate predictive models and evaluate them by 10-fold cross-validation.

Thus, use `sklearn.linear_model.LinearRegression` and `sklearn.tree.DecisionTreeRegressor` to create the models. Use the default parameters for the linear regression and set `random_state=0` for the decision.

Be aware that a linear model requires to scale numerical features. Please use `sklearn.preprocessing.StandardScaler` so that your linear regression model behaves the same way as the quiz author intended ;)

fold, count the number of times the linear model has a better test score than the decision tree model. Select the range which this number belongs to:

- ○ a) [0, 3]: the linear model is substantially worse than the decision tree
- ○ b) [4, 6]: both models are almost equivalent
- [✓ c) [7, 10]: the linear model is substantially better than the decision tree]

You have used 1 of 1 submissions

Question 2 (1/1 point)

Instead of using the default parameters for the decision tree regressor, we will optimize the `max_depth` of the tree. Vary the `max_depth` from 1 level up to 15 levels. Use nested cross-validation to evaluate a grid-search (`sklearn.model_selection.GridSearchCV`). Set `cv=10` for both the inner and outer cross-validations, then answer the questions below

What is the optimal tree depth for the current problem?

- ○ a) The optimal depth is ranging from 3 to 5
- [✓ b) The optimal depth is ranging from 5 to 8]
- ○ c) The optimal depth is ranging from 8 to 11
- ○ d) The optimal depth is ranging from 11 to 15

Now, we want to evaluate the generalization performance of the decision tree while taking into account the fact that we tune the depth for this specific dataset. Use the grid-search as an estimator inside a `cross_validate` to automatically tune the `max_depth` parameter on each cross-validation fold.

A tree with tuned depth

- ○ a) is always worse than the linear models on all CV folds
- ● b) is often but not always worse than the linear model [✓]
- ○ c) is often but not always better than the linear model
- ○ d) is always better than the linear models on all CV folds

Note: Try to set the random_state of the decision tree to different values e.g. random_state=1 or random_state=2 and re-run the nested cross-validation to check that your answer is stable enough.

You have used 1 of 1 submissions

Question 4 (1/1 point)

Instead of using only the numerical features you will now use the entire dataset available in the variable `data`.

Create a preprocessor by dealing separately with the numerical and categorical columns. For the sake of simplicity, we will assume the following:

- categorical columns can be selected if they have an `object` data type;
- use an `OrdinalEncoder` to encode the categorical columns;
- numerical columns should correspond to the `numerical_features` as defined above.

Evaluate this model using `cross_validate` as in the previous questions.

A tree model trained with both numerical and categorical features

○ a) is most often worse than the tree model using only the numerical features

[✓ b) is most often better than the tree model using only the numerical features]

Note: Try to set the random_state of the decision tree to different values e.g. random_state=1 or random_state=2 and re-run the (this time single) cross-validation to check that your answer is stable enough.

You have used 1 of 1 submissions

YOUR EXPERIENCE

According to you, the 'Wrap-up Quiz' of this module was:

○ **Too easy, I got bored**

○ **Adapted to my skills**

○ **Difficult but I was able to follow**

○ **Too difficult**

Submit

To follow this lesson, I spent:

○ **less than 30 minutes**

○ **30 min to 1 hour**

○ **1 to 2 hours**

○ **I don't know**

Submit

FORUM (EXTERNAL RESOURCE)

There are no more M5. Wrap-up quiz 5 topics. Ready to start a new conversation?