

This notebook is an exercise in the [Introduction to Machine Learning](#) course. You can reference the tutorial at [this link](#).

Recap

Here's the code you've written so far.

```
In [ ]: # Code you have previously used to load data
import pandas as pd
from sklearn.metrics import mean_absolute_error
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeRegressor

# Path of the file to read
iowa_file_path = '../input/home-data-for-ml-course/train.csv'

home_data = pd.read_csv(iowa_file_path)
# Create target object and call it y
y = home_data.SalePrice
# Create X
features = ['LotArea', 'YearBuilt', '1stFlrSF', '2ndFlrSF', 'FullBath',
            'BedroomAbvGr', 'TotRmsAbvGrd']
X = home_data[features]

# Split into validation and training data
train_X, val_X, train_y, val_y = train_test_split(X, y, random_state=1)

# Specify Model
iowa_model = DecisionTreeRegressor(random_state=1)
# Fit Model
iowa_model.fit(train_X, train_y)
```

```

# Make validation predictions and calculate mean absolute error
val_predictions = iowa_model.predict(val_X)
val_mae = mean_absolute_error(val_predictions, val_y)
print("Validation MAE when not specifying max_leaf_nodes: {:.0f}".format(
    val_mae))

# Using best value for max_leaf_nodes
iowa_model = DecisionTreeRegressor(max_leaf_nodes=100, random_state=1)
iowa_model.fit(train_X, train_y)
val_predictions = iowa_model.predict(val_X)
val_mae = mean_absolute_error(val_predictions, val_y)
print("Validation MAE for best value of max_leaf_nodes: {:.0f}".format(
    val_mae))

# Set up code checking
from learntools.core import binder
binder.bind(globals())
from learntools.machine_learning.ex6 import *
print("\nSetup complete")

```

Exercises

Data science isn't always this easy. But replacing the decision tree with a Random Forest is going to be an easy win.

Step 1: Use a Random Forest

```

In [ ]: from sklearn.ensemble import RandomForestRegressor

# Define the model. Set random_state to 1
rf_model = RandomForestRegressor(random_state=1)

# fit your model

```

```
rf_model.fit(train_X, train_y)

# Calculate the mean absolute error of your Random Forest model on the
# validation data
rf_predict = rf_model.predict (val_X)
rf_val_mae = mean_absolute_error(val_y, rf_predict)

print("Validation MAE for Random Forest Model: {}".format(rf_val_mae))

step_1.check()
```

```
In [ ]: # The lines below will show you a hint or the solution.
        # step_1.hint()
        # step_1.solution()
```

So far, you have followed specific instructions at each step of your project. This helped learn key ideas and build your first model, but now you know enough to try things on your own.

Machine Learning competitions are a great way to try your own ideas and learn more as you independently navigate a machine learning project.

Keep Going

You are ready for [Machine Learning Competitions](#).

Have questions or comments? Visit the [Learn Discussion forum](#) to chat with other Learners.