

## Artificial Intelligence with Python – LAB TASK 3

Kartik Srinivas

16BIS0085

Question:

### Lab Task 3

Download the dataset from the following link:

<https://www.kaggle.com/c/house-prices-advanced-regression-techniques/overview>

The [Ames Housing dataset](#) was compiled by Dean De Cock for use in data science education. It's an incredible alternative for data scientists looking for a modernized and expanded version of the often cited Boston Housing dataset.

Ask a home buyer to describe their dream house, and they probably won't begin with the height of the basement ceiling or the proximity to an east-west railroad. But this playground competition's dataset proves that much more influences price negotiations than the number of bedrooms or a white-picket fence.

With 79 explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, you have to predict the final price of each home.

#### Goal

Predict the sales price for each house. For each Id in the test set, you have to predict the value of the SalePrice variable.

Answer:

```
from typing import List
import numpy as np
import pandas as pd
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_log_error
import matplotlib.pyplot as plt

#READ CSV FILE
train_file = r'C:\Users\karti\Desktop\Fall Semester 2019-20\Artificial
Intelligence with Python\Lab\task3\train.csv'
data = pd.read_csv(train_file)
summary = data.describe()

#SELECT THE FEATURES, DEFINE MODEL
y = data.SalePrice
data_features = [x for x in data.columns if str(data[x][0]).isdigit()[:-1]]
X = data[data_features]
describe = X.describe()
head = X.head()
```

```
data_model = DecisionTreeRegressor(random_state=1)
data_model.fit(X, y)
X = np.array(X)
```

```
test_file = r'C:\Users\karti\Desktop\Fall Semester 2019-20\Artificial Intelligence  
with Python\Lab\task3\test.csv'  
test_data = pd.read_csv(test_file)  
X_test = test_data[data_features]  
X_test = pd.DataFrame(X_test).fillna(X_test.mean())  
print(X_test)  
result = data_model.predict(X_test)  
print(result)
```

```
submission_file = r'C:\Users\karti\Desktop\Fall Semester 2019-20\Artificial  
Intelligence with Python\Lab\task3\submission.csv'  
submission_data = pd.read_csv(submission_file)
```

```
print('RMSE is: ', (mean_absolute_error(submission_data['SalePrice'], result)) **
0.5)
print('RMSLE is: ', (mean_squared_log_error(submission_data['SalePrice'], result))
** 0.5)
```

The screenshot displays a Jupyter Notebook environment. The top menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The file path is C:\Users\karti\Desktop\Fall Semester 2019-20\Artificial Intelligence with Python\Lab\task3\labtask.py. The notebook contains a Python script for a linear regression model. The script imports List, numpy, pandas, DecisionTreeRegressor, mean\_absolute\_error, mean\_squared\_log\_error, and matplotlib.pyplot. It reads data from a CSV file, summarizes it, and defines features and the model. The bottom panel shows the variable inspector with data dimensions and model details.

```

1 from typing import List
2 import numpy as np
3 import pandas as pd
4 from sklearn.tree import DecisionTreeRegressor
5 from sklearn.metrics import mean_absolute_error
6 from sklearn.metrics import mean_squared_log_error
7 import matplotlib.pyplot as plt
8
9
10 #READ CSV FILE
11 train_file = r'C:\Users\karti\Desktop\Fall Semester 2019-20\Artificial Intelligence with Python\Lab\task3\train.csv'
12 data = pd.read_csv(train_file)
13 summary = data.describe()
14
15 #SELECT THE FEATURES, DEFINE MODEL
16 y = data.SalePrice
17 data_features = [x for x in data.columns if str(data[x][0]).isdigit()][:-1]

```

The variable inspector shows the following variables:

- labtask**: 1459 rows x 34 columns. Data range: [127500, 155000, 223500, ..., 150750, 98000, 271900]. RMSE is: 242.2496680596211. RMSE is: 0.41317764193750883.
- In[3]**: In[3]:
- Special Variables**:
  - DecisionTreeRegressor**: (ABCMeta) <class 'sklearn.tree.tree.DecisionTreeRegressor'
  - List**: (\_GenericAlias) typing.List
  - X**: (ndarray) [[ 1 60 8450... 0 2 2008]]
  - X\_test**: (DataFrame) Id MSSubClass Lc... View as DataFrame
  - data**: (DataFrame) Id MSSubClass MSZ... View as DataFrame
  - data\_features**: (list) ['Id', 'MSSubClass', 'LotArea', 'OverallQ... View as DataFrame
  - data\_model**: (DecisionTreeRegressor) DecisionTreeRegressor... View as DataFrame

The bottom status bar shows the Python Console, Terminal, Find, and TODO. The system clock indicates 34:14 CRLF UTF-8 8 spaces\* on 01-09-2019.