

AI・データ科学に関する演習

ランダムフォレストを使った
各都道府県の住宅戸数の予測

演習の概要

- ◆ この演習では、e-Statから得た都道府県別の人口、面積と地価から住宅戸数を予測するモデルを作成し、このモデルを使って実際に住宅戸数を予測します。このような問題は回帰問題と呼ばれます。
- ◆ この演習では、回帰問題の分析方法として機械学習法の一つであるランダムフォレストを使います。
- ◆ ランダムフォレストによる予測結果をExcelの（重）回帰分析による結果と比較します

身近な回帰問題

No.	性別	身長	靴のサイズ(cm)
1	男	168	25
2	女	155	22.5
3	男	181	27.5
...
100	女	162	24

「説明変数」と呼ばれます

データから
予測モデル
(回帰式など)作成

性別、身長

予測モデル

足のサイズ

本演習の回帰問題

No.	都道府県	人口(人)	面積(km ²)	地価(円/m ²)	住宅数(戸)
1	北海道	5722908	83424	18000	2416700
2	青森県	1307942	9645	16700	501500
3	岩手県	1280046	15275	24500	483600
...
47	沖縄県	1433455	2281	45700	577000

「目的変数」と呼ばれます

ランダムフォレスト

- ・ データ学習
- ・ 人口、面積、地価から予測

住宅戸数

演習の流れ

1. 各自のPCにデータとソースコードをダウンロード
2. デスクトップにフォルダ作成、データとソースコード(プログラム)を移動(コピー)
3. Spyder (Python 3.8)の起動
4. ランダムフォレストプログラムの実行
5. ランダムフォレストによる予測結果の整理
6. Excelを用いた重回帰分析による予測結果と比較

※各自のPCのOSはWindows10であり、Anaconda3(64-bit)がインストールされているものとする

1. 各自のPCにデータとソースコードをダウンロード

分析に用いるデータ(csv形式)とソースコードをPCにダウンロード

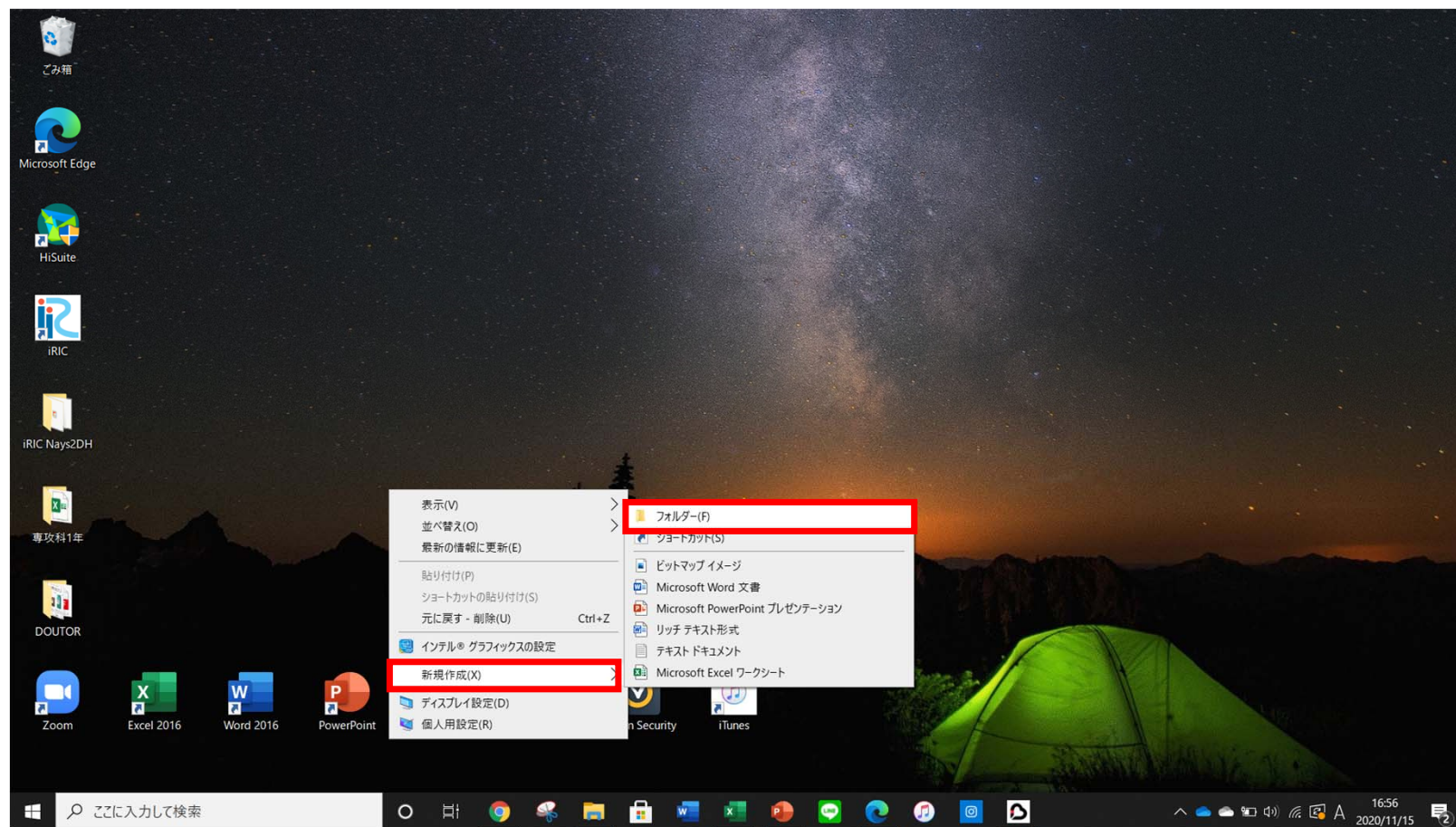


① 「Estat_B.csv」：入力データ

② 「RF_for_Esta.py」：ランダムフォレストのプログラム

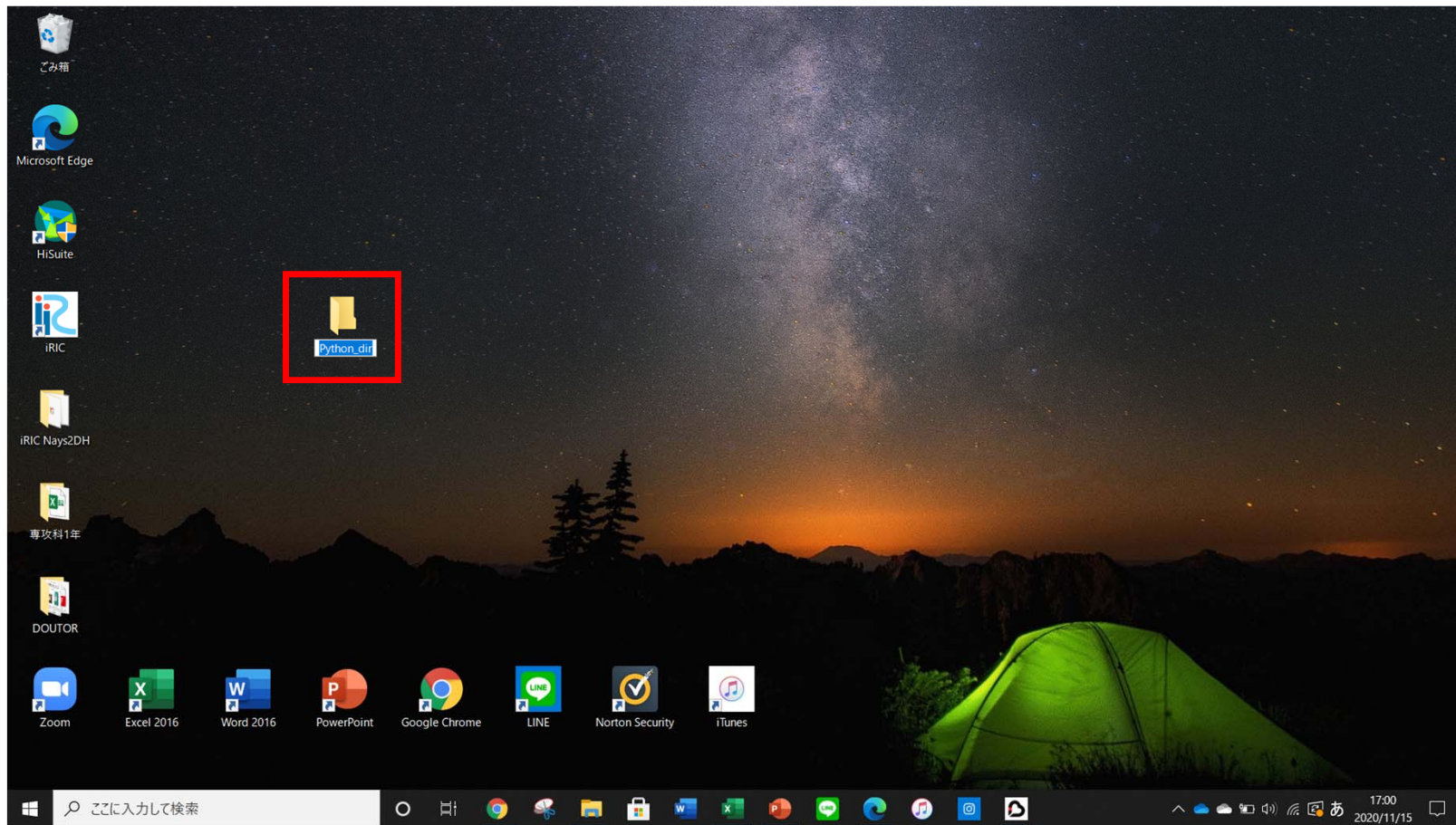
2. デスクトップへのフォルダ作成

各自のPCのデスクトップで右クリックし、「新規作成」→「フォルダー」をクリック



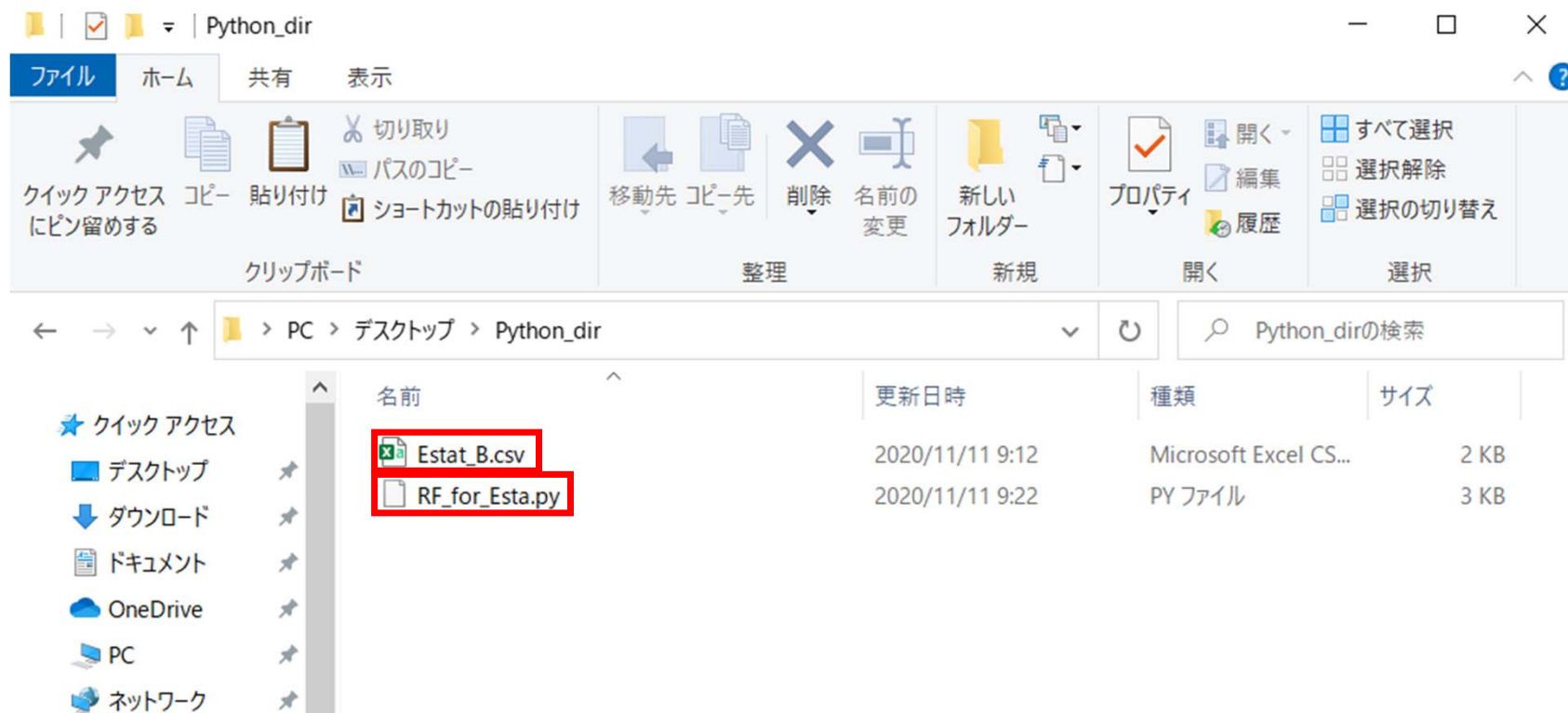
2. デスクトップへのフォルダ作成

デスクトップ上に「新しいフォルダー」が作成されるので「Python_dir」と名前を付ける



2. デスクトップへのフォルダ作成

「Python_dir」を開き、入力データとソースコード(プログラム)をコピー(移動)



① 「Estat_B.csv」：入力データ

② 「RF_for_Esta.py」：ランダムフォレストのプログラム

入力データの内容

✓「Estat_B.csv」をダブルクリック

✓Excelで開くことができる

	A	B	C	D	E	F	G	H
1	5722908	83424.31	18000	2416700				
2	1307942	9645.59	16700	501500				
3	1280046	15275.01	24500	483600				
4	2333952	7282.20	34000	953600				
5	1022910	11637.54	14200	383800				
6	1123440	9323.15	19200	393200				
7	1914561	13783.74	22500	731100				
8	2916834	6097.06	32800	1126600				
9	1974333	6408.09	33200	761400				
10	1972943	6362.28	30700	786600				
11	7266615	3797.75	105400	3023300				
12	6222705	5157.65	71500	2635200				
13	13515190	2190.93	323800	6805500				
14	9126281	2415.83	173200	4000000				
15	2304149	12584.1	26500	844300				
16	1066150	4247.61	30500	390900				
17	1154105	4186.09	41600	455000				
18	786555	4190.49	31500	279300				
19	835005	4465.27	25900	329200				
20	2090320	12561.56	25500	806600				

都道府県番号(1~47)→次ページを参照

人口 (人)

面積 (km²)

地価 (円/m²)

住宅数 (戸)

※都道府県番号と都道府県名の対応を示したファイル「都道府県名データ.xlsx」はHPにアップロード済み

入力データの内容

都道府県番号について

	A	B	C	D	E	F	G	H
1			人口(人)	面積(km ²)	地価(円/m ²)	住宅数(戸)		
2	1	北海道	5722908	83424.31	18,000	2416700		
3	2	青森県	1307942	9645.59	16,700	501500		
4	3	岩手県	1280046	15275.01	24,500	483600		
5	4	宮城県	2333952	7282.22	34,000	953600		
6	5	秋田県	1022940	11637.54	14,200	383800		
7	6	山形県	1123440	9323.15	19,200	393200		
8	7	福島県	1914561	13783.74	22,500	731100		
9	8	茨城県	2916834	6097.06	32,800	1126600		
10	9	栃木県	1974333	6408.09	33,200	761400		
11	10	群馬県	1972943	6362.28	30,700	786600		
12	11	埼玉県	7266615	3797.75	105,400	3023300		
13	12	千葉県	6222705	5157.65	71,500	2635200		
14	13	東京都	13515190	2190.93	323,800	6805500		
15	14	神奈川県	9126281	2415.83	173,700	4000000		
16	15	新潟県	2304149	12584.1	26,500	844300		
17	16	富山県	1066150	4247.61	30,500	390900		
18	17	石川県	1154105	4186.09	41,600	455000		
19	18	福井県	786555	4190.49	31,500	279300		
20	19	山梨県	835005	4465.27	25,900	329200		
21	20	長野県	2099329	13561.56	25,500	806600		
22	21	岐阜県	2031853	10621.29	33,800	750300		
23	22	静岡県	3700496	7777.42	66,700	1425100		
24	23	愛知県	7483027	5172.48	97,900	3069200		

	A	B	C	D	E	F	G	H
25	24	三重県	1816049	5774.4	31,200	720000		
26	25	滋賀県	1412912.546	4017.38	46,400	543000		
27	26	京都府	2610499.54	4612.19	102,400	1158900		
28	27	大阪府	8839468.572	1905.14	146,900	3949600		
29	28	兵庫県	5534552.448	8400.96	100,700	2308700		
30	29	奈良県	1364171.424	3690.94	52,600	529000		
31	30	和歌山県	963364.291	4724.69	34,700	383900		
32	31	鳥取県	573402.675	3507.05	20,100	215600		
33	32	島根県	694302.84	6708.24	22,100	264700		
34	33	岡山県	1921626.45	7114.5	29,200	771100		
35	34	広島県	2844007.53	8479.45	51,900	1208800		
36	35	山口県	1404606.54	6112.3	25,600	591000		
37	36	徳島県	755934.295	4146.65	30,400	305300		
38	37	香川県	976269.744	1876.72	33,100	397600		
39	38	愛媛県	1385538.451	5676.11	37,900	581400		
40	39	高知県	728152.825	7103.93	31,500	315400		
41	40	福岡県	5101585.84	4986.4	44,600	2239000		
42	41	佐賀県	832760.016	2440.68	20,200	300300		
43	42	長崎県	1377225.597	4132.09	23,600	555200		
44	43	熊本県	1786394.285	7409.35	27,800	698100		
45	44	大分県	1166056.569	6340.71	24,800	481800		
46	45	宮崎県	1103828.737	7735.31	24,600	460200		
47	46	鹿児島県	1648137.036	9186.94	27,900	709000		
48	47	沖縄県	1433455.808	2281.12	45,700	577000		

○プログラムの内容

RF_for_Esta - メモ帳

ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)

```
from sklearn.ensemble import RandomForestRegressor
import pandas as pd
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import numpy as np
from pandas import DataFrame
```

```
df = pd.read_csv('Estat_B.csv', header=None)
a_df = df.values
```

```
#目的変数のセット
y = a_df[:, 3]
#説明変数のセット
X = a_df[:, 0:3]
```

~~#学習用データと検証用データに分離~~

```
(X_train, X_test, y_train, y_test) = train_test_split(X, y, test_size = 0.3, random_state = 666)
```

```
#モデルの構築, パラメータはデフォルト
forest = RandomForestRegressor()
forest.fit(X_train, y_train)
```

#予測値の計算

```
y_train_pred = forest.predict(X_train)
y_test_pred = forest.predict(X_test)
```

✓「RF_for_Esta.py」をダブルクリック

✓メモ帳やテキストエディタで開くことができる

ランダムフォレストに必要な関数の呼び出しなど

入力データの読み込み

データの3割をテスト用に
(残りの7割を学習させる)

Train: 学習させるためのデータ

Test: 実際に分析させる(検証)データ

○プログラムの内容

```
#平均二乗誤差
from sklearn.metrics import mean_squared_error
print('MSE train : %.3f, test : %.3f' % (mean_squared_error(y_train, y_train_pred), mean_squared_error(y_test, y_test_pred)))

# R^2の計算
from sklearn.metrics import r2_score
print('MSE train : %.3f, test : %.3f' % (r2_score(y_train, y_train_pred), r2_score(y_test, y_test_pred)))

#print(y_train_pred, y_train_pred - y_train)
#print()
#print(y_test_pred, y_test_pred - y_test)
#print()
```

```
#残差のプロット
plt.figure(figsize = (10, 7))
plt.scatter(y_train_pred, y_train_pred - y_train, c = 'blue', marker = 'o', s = 35, alpha = 0.5, label = 'Training data')
plt.scatter(y_test_pred, y_test_pred - y_test, c = 'red', marker = 's', s = 35, alpha = 0.7, label = 'Test data')
plt.xlabel('Predicted values')
plt.ylabel('Residuals')
plt.legend(loc = 'upper left')
plt.hlines(y = 0, xmin = 0, xmax = 6000000, lw = 2, color = 'green')
plt.xlim([0, 6000000])
plt.show()
```

```
train_predicted = []
test_predicted = []

train_predicted.append(y_train_pred)
test_predicted.append(y_test_pred)

train_predicted_np = np.array(train_predicted)
test_predicted_np = np.array(test_predicted)
```

```
df = pd.DataFrame(train_predicted_np)
df = df.T
df.to_csv('y_train_pred.csv')

df = pd.DataFrame(test_predicted_np)
df = df.T
df.to_csv('y_test_pred.csv')
```

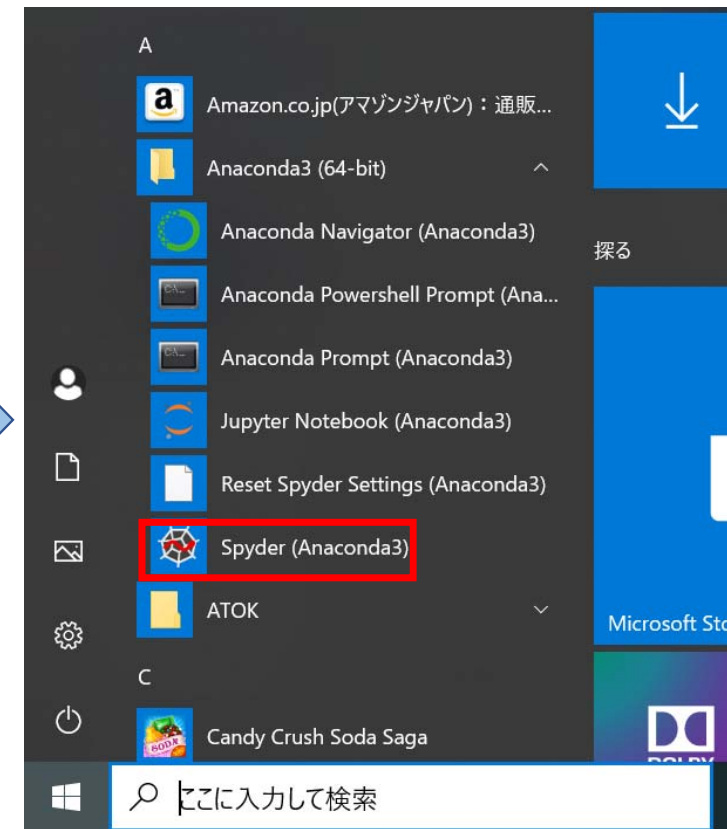
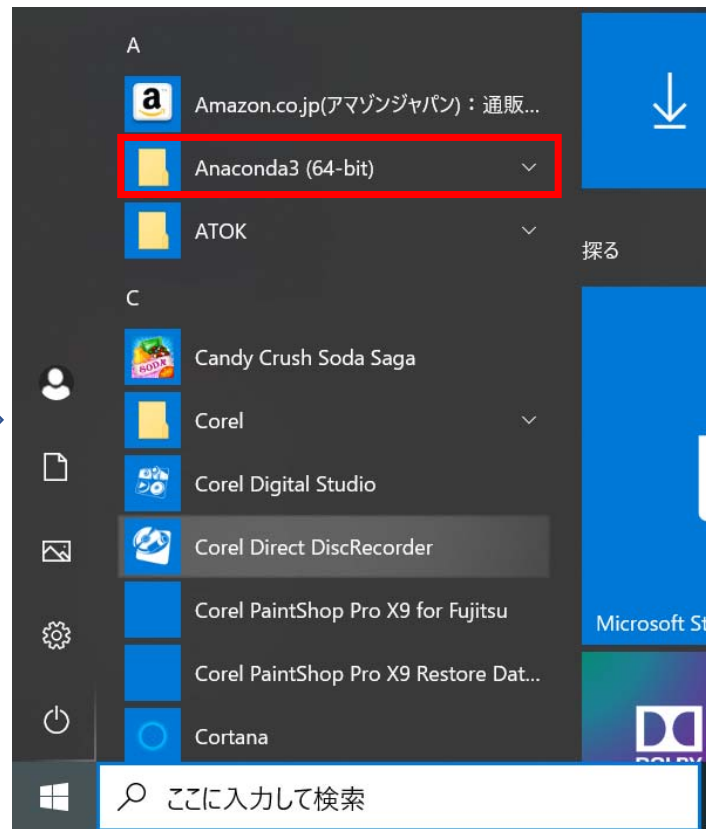
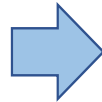
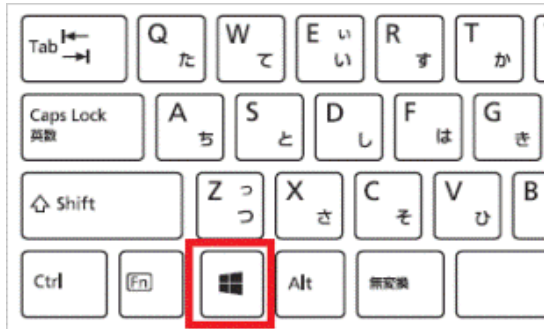
残差と住宅戸数の予測値の関係を表すグラフの設定

訓練データとテストデータを用いた予測値を変数
train_predicted_npとtest_predicted_npにコピー

変数の値をcsvファイルへ書き出し

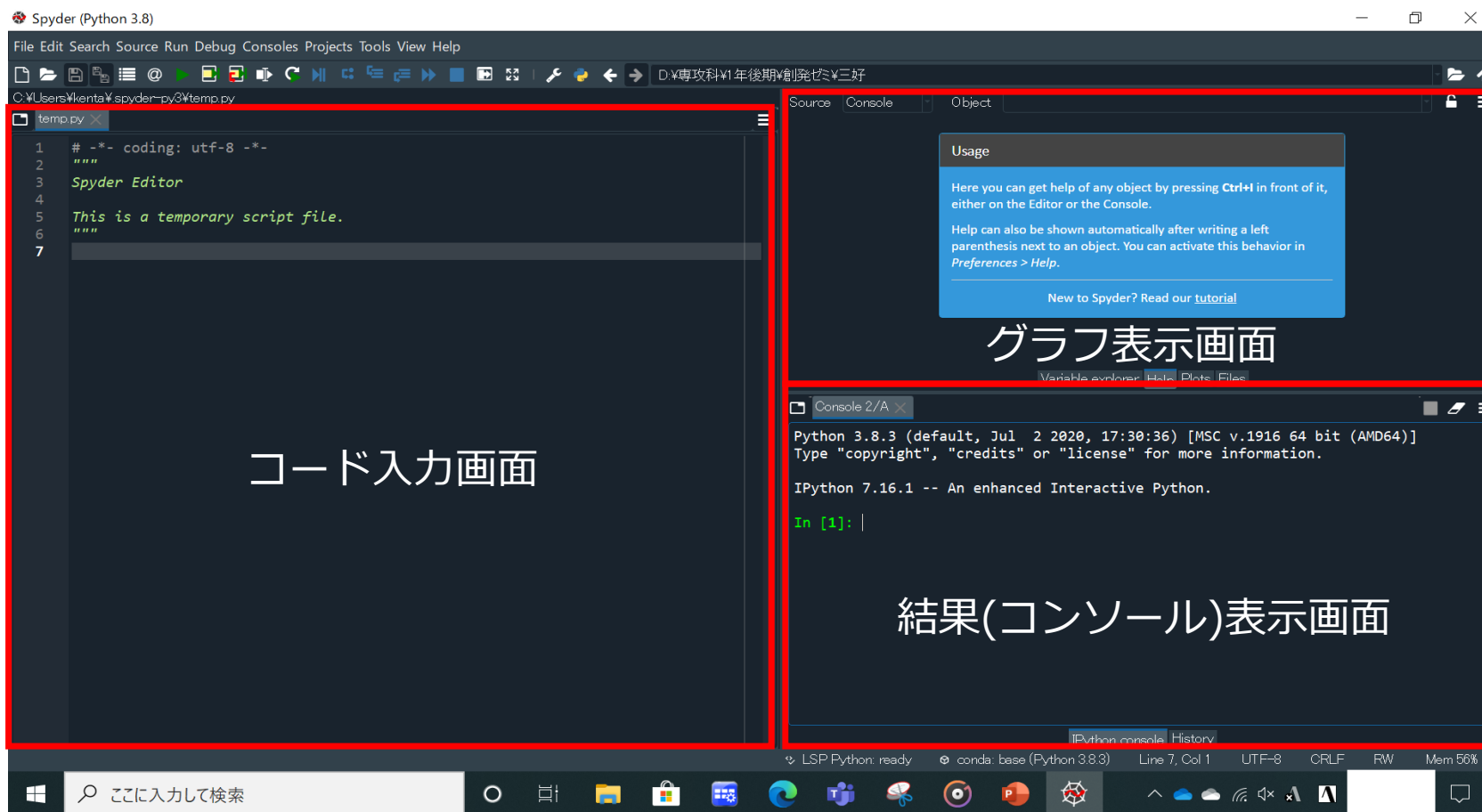
3. Spyder (Python3.8)の起動

Windowsキーを押し、Anaconda3(64-bit)をクリック、Spyderを起動



4. ランダムフォレストプログラムの実行

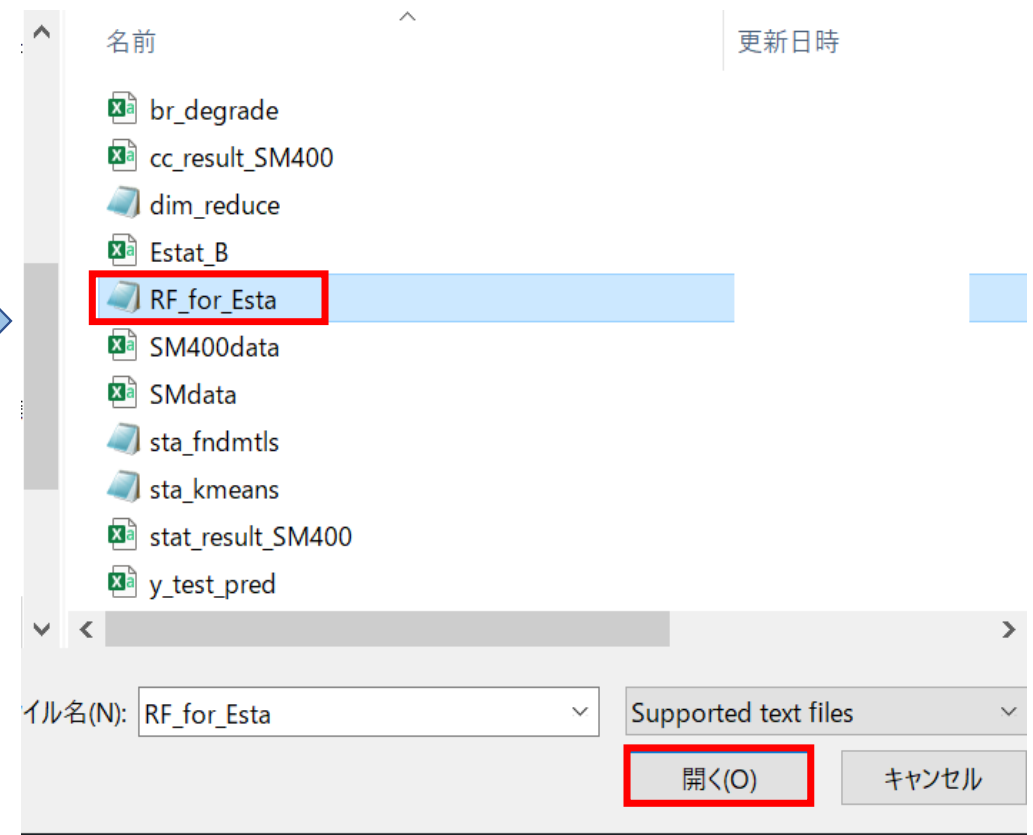
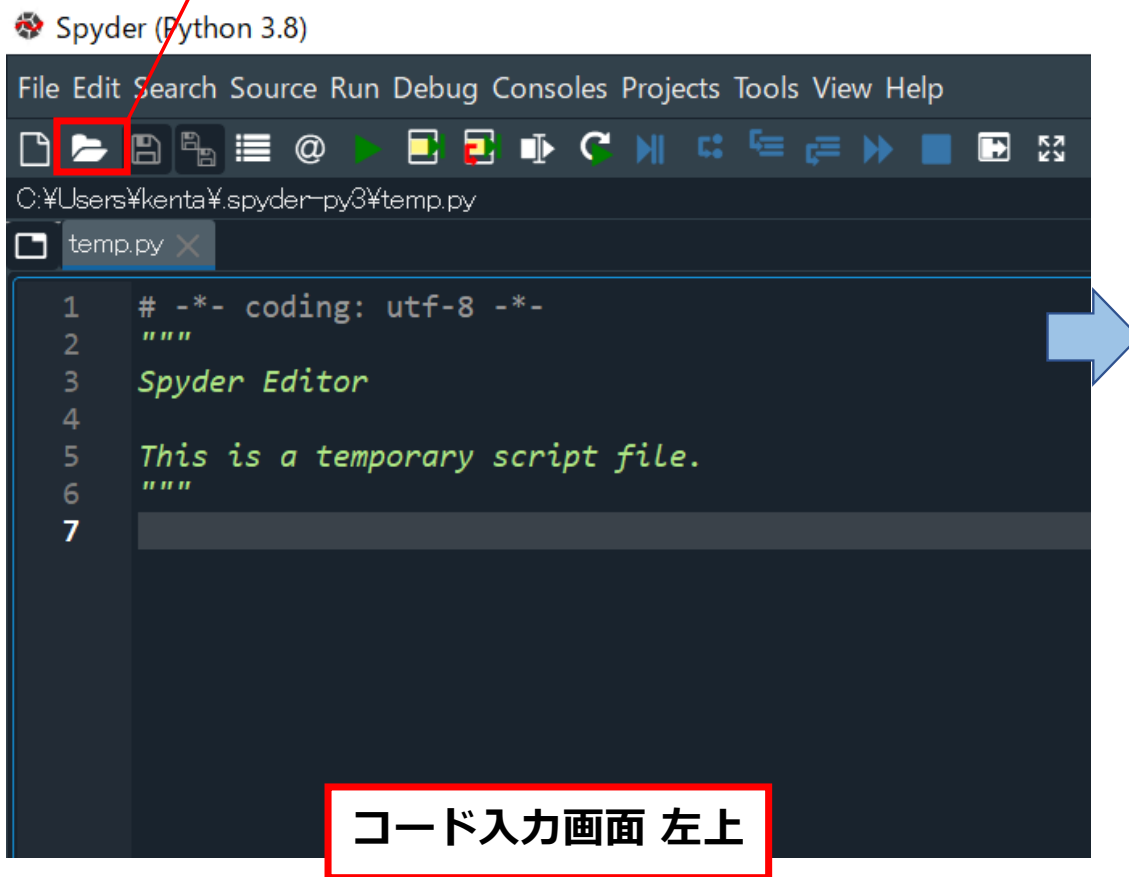
Spyderを起動すると以下のような画面が表示される



4. ランダムフォレストプログラムの実行

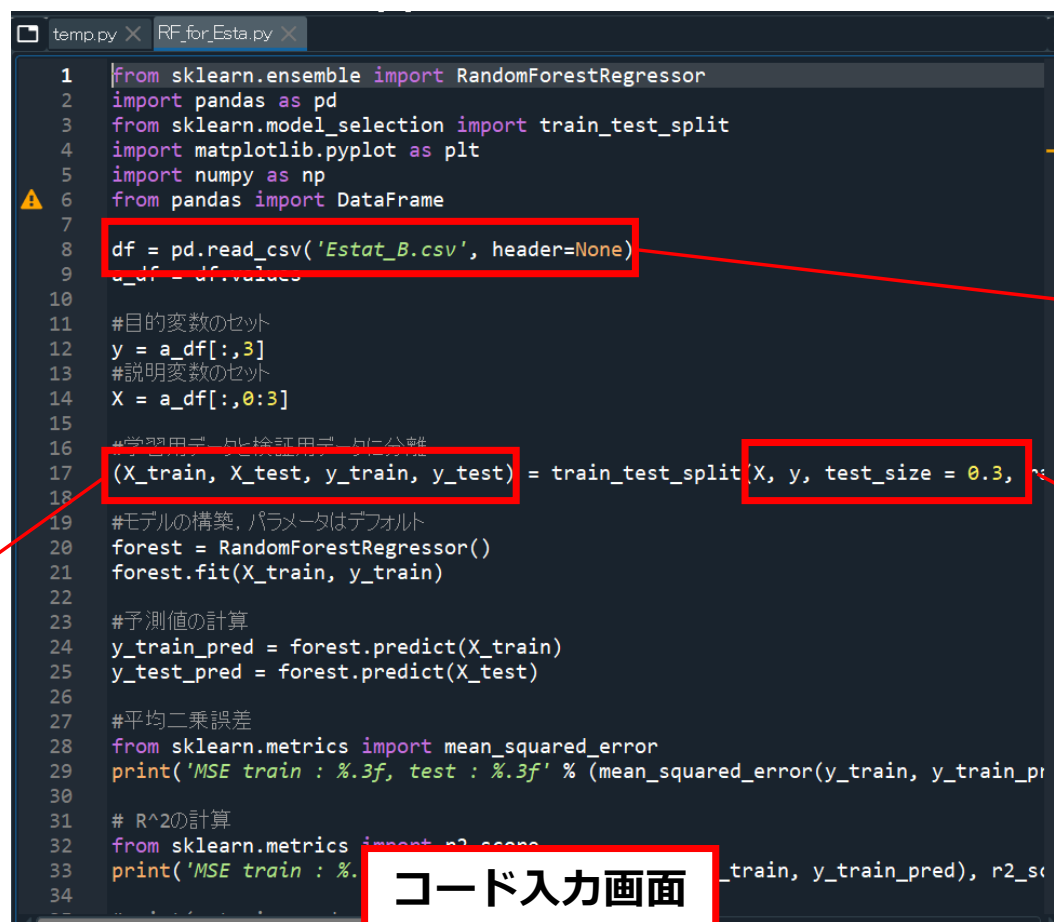
画面左上(Open file)をクリックして、ファイルの一覧から「RF_for_Esta.py」を選択

Open fileボタン



4. ランダムフォレストプログラムの実行

Spyder上に読み込んだコードが表示される



```
1 from sklearn.ensemble import RandomForestRegressor
2 import pandas as pd
3 from sklearn.model_selection import train_test_split
4 import matplotlib.pyplot as plt
5 import numpy as np
6 from pandas import DataFrame
7
8 df = pd.read_csv('Estat_B.csv', header=None)
9 a_df = df.values
10
11 #目的変数のセット
12 y = a_df[:,3]
13 #説明変数のセット
14 X = a_df[:,0:3]
15
16 #学習用データと検証用データに分離
17 (X_train, X_test, y_train, y_test) = train_test_split(X, y, test_size = 0.3,
18
19 #モデルの構築. パラメータはデフォルト
20 forest = RandomForestRegressor()
21 forest.fit(X_train, y_train)
22
23 #予測値の計算
24 y_train_pred = forest.predict(X_train)
25 y_test_pred = forest.predict(X_test)
26
27 #平均二乗誤差
28 from sklearn.metrics import mean_squared_error
29 print('MSE train : %.3f, test : %.3f' % (mean_squared_error(y_train, y_train_pred),
30
31 # R^2の計算
32 from sklearn.metrics import r2_score
33 print('MSE train : %.3f, test : %.3f, R^2 train : %.3f, R^2 test : %.3f' % (
34
```

コード入力画面

入力データの読み込み
(csvファイルより)

データの3割をテスト用に
(残りの7割を学習させる)

Train: 学習させるためのデータ

Test: 実際に分析させるデータ

4. ランダムフォレストプログラムの実行

画面左上(Run file)をクリック、画面右下にコンソールが表示される

Run fileボタン

Spyder (Python 3.8)

File Edit Search Source Run Debug Consoles Projects Tools View Help

temp.py

```
1 # -*- coding: utf-8 -*-
2 """
3 Spyder Editor
4 This is a temporary script file.
5 """
6
7
```

平均二乗誤差

R^2 値

Console 1/A

IPython 7.16.1 -- An enhanced Interactive Python.

In [1]:

MSE train : 53116892499.781, test : 8811853724.133
MSE train : 0.974, test : 0.987

Figures now render in the Plots pane by default. To make them also appear inline in the Console, uncheck "Mute Inline Plotting" under the Plots pane options menu.

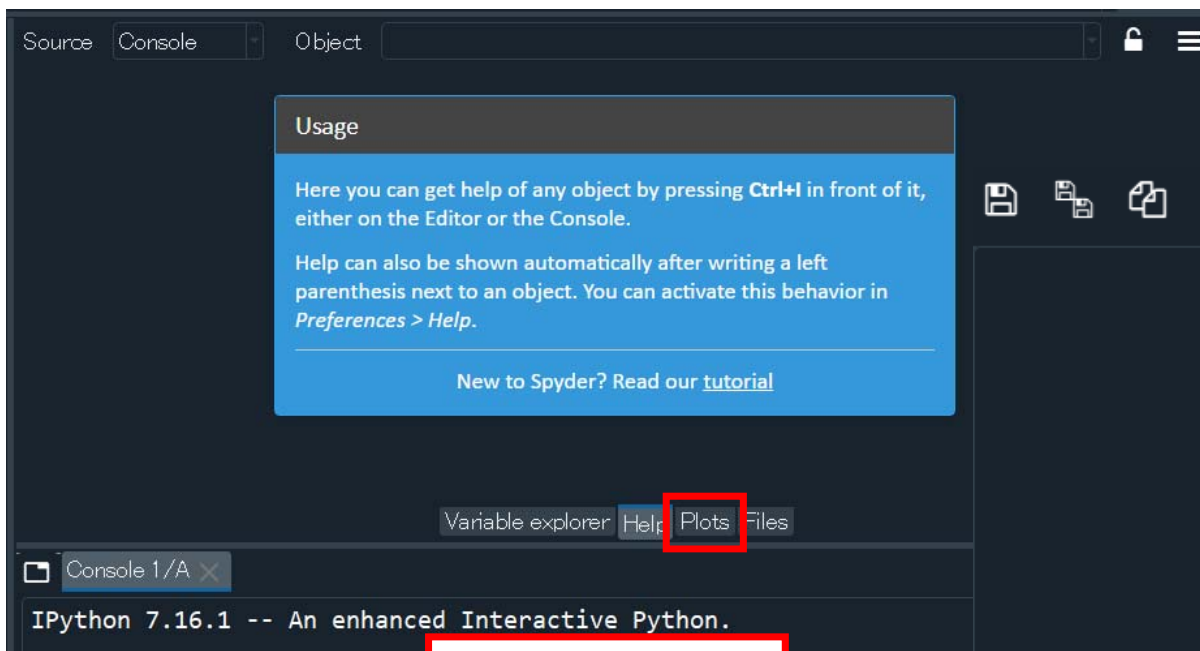
In [2]:

コード入力画面 左上

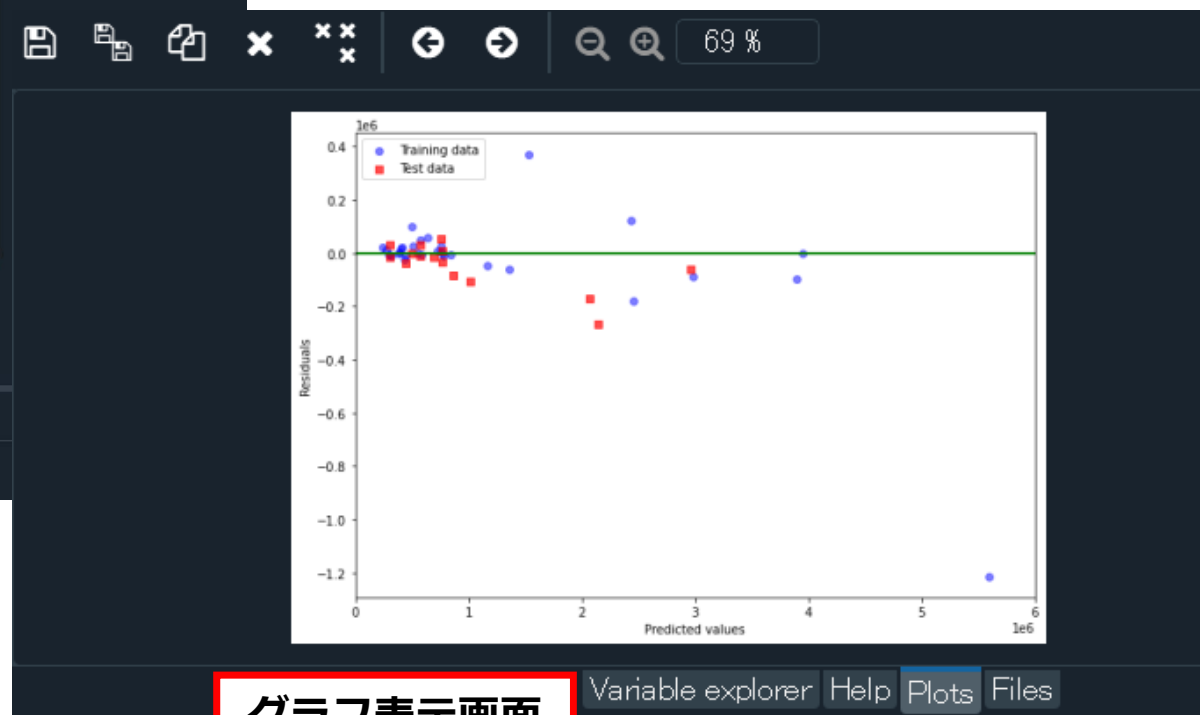
結果(コンソール)表示画面

4. ランダムフォレストプログラムの実行

画面右上(Plots)をクリック、プログラム結果の表示



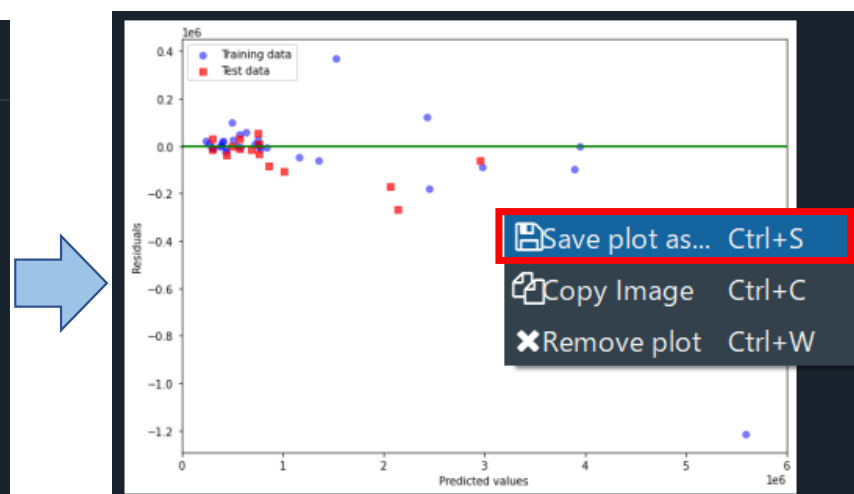
グラフ表示画面



グラフ表示画面

4. ランダムフォレストプログラムの実行

プロットを保存したい場合はグラフ上で右クリック, (Save plot as..)を選択



好きな形式で保存

5. 計算結果の整理

人口、面積、地価と建物の数には強い相関があることがわかる

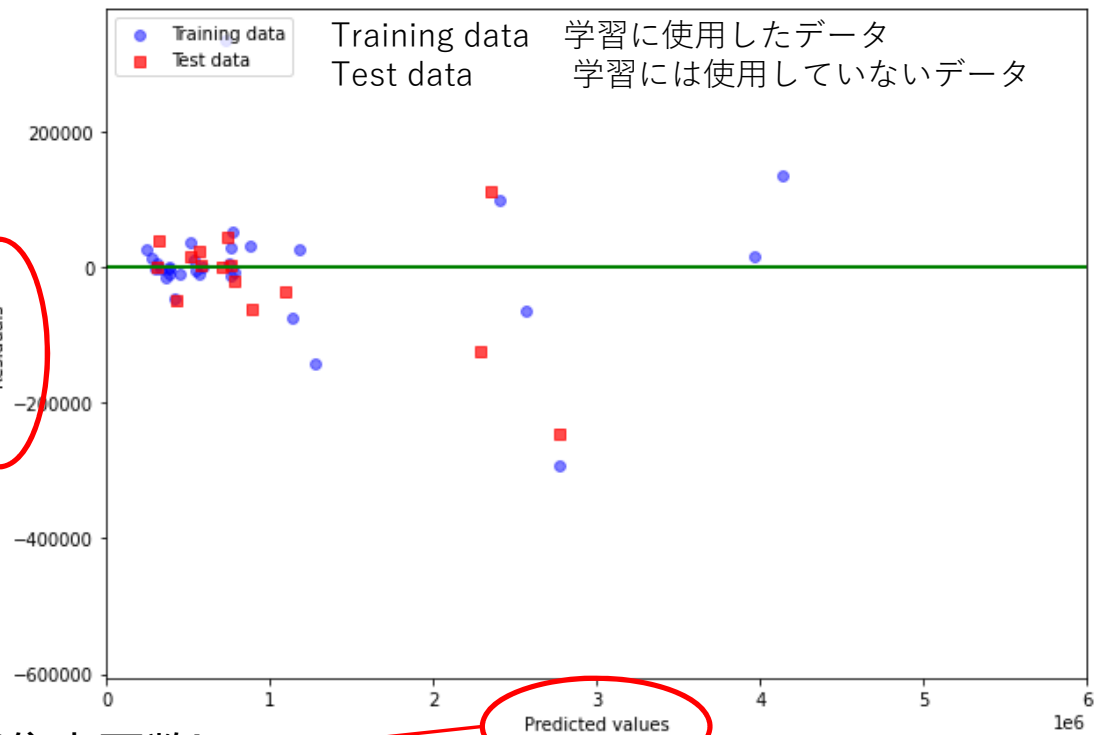
```
MSE train : 53116892499.781, test : 8811853724.133  
MSE train : 0.974, test : 0.987
```

結果(コンソール)表示画面

R^2 値 : 1に近いほど強い正の相関

E-statの住宅戸数とその予測値の差

分析で得た予測値(住宅戸数)



※学習に使用する訓練データと検証に使用するテストデータの分割が毎回異なるため、ランダムフォレストによる分析結果も、実行するごとに変わります。

6. Excelを用いた重回帰分析による予測

◆ Excelで重回帰分析を行うために、準備をします

以下の順でクリック

①「ファイル」メニュー

②その他

③オプション

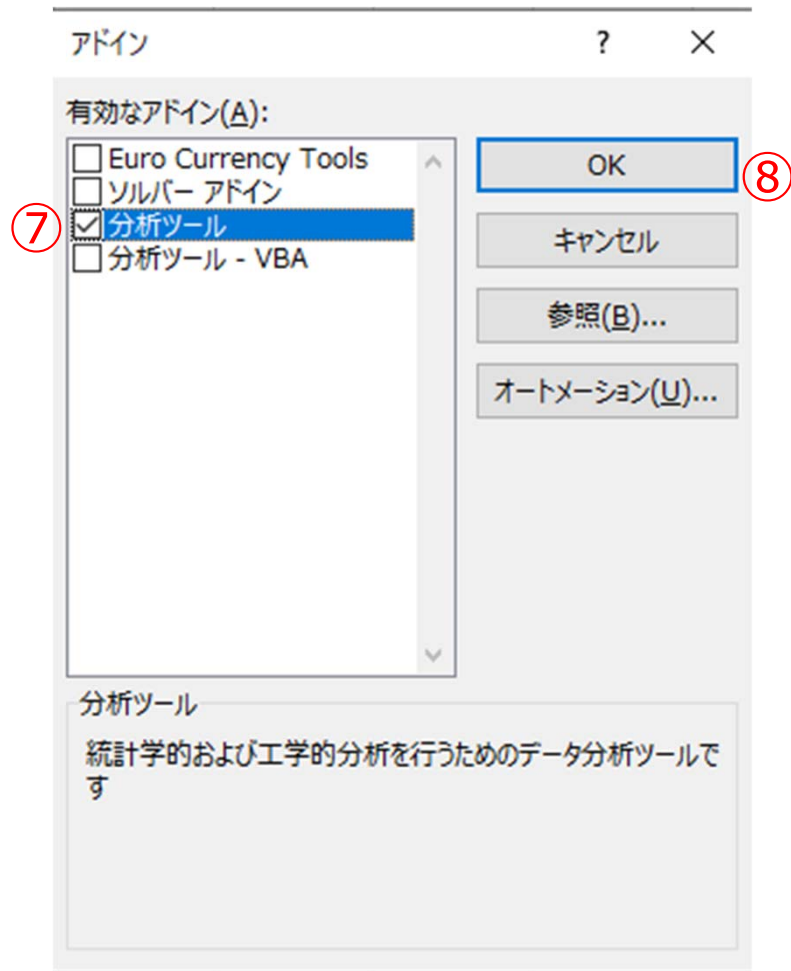
④アドイン

⑤Excelアドイン

⑥設定



6. Excelを用いた重回帰分析による予測



⑦分析ツールにチェックを入れる

⑧OK

（「アドイン」の画面で、既に「分析ツール」にチェックが入っている場合は、「キャンセル」ボタンをクリックし、「ホーム」画面に戻る）

6. Excelを用いた重回帰分析による予測

◆準備ができれば、実際に重回帰分析を行います

以下の順でクリック

①データ

②データ分析

③回帰分析

④OK

Excelの「データ」タブと「データ分析」ツール、および「データ分析」タスクパネのスクリーンショット。以下の手順で操作が行われます。

- ① データ
- ② データ分析
- ③ 回帰分析
- ④ OK

2015	人口(人)	面積(km2)	地価(円/m2)	住宅数(戸)
北海道	5733000	83424.31	18,000	2416700
青森県	1191000	10,768.21	18,000	301500
岩手県	1216000	15,074.35	18,000	283600
宮城県	1409000	16,216.29	18,000	253600
秋田県	1238000	10,768.21	18,000	283800
山形県	1268000	9,534.35	18,000	293200
福島県	2018000	37,757.25	18,000	731100
茨城県	2869000	5,207.25	18,000	126600
栃木県	2821000	9,876.25	18,000	761400
群馬県	1972943	6362.28	30,700	786600
埼玉県	7266615	3797.75	105,400	3023300
千葉県	6222705	5157.65	71,500	2635200
東京都	13515190	2190.93	323,800	6805500
神奈川県	9126281	2415.83	173,700	4000000
新潟県	2304149	12584.1	26,500	844300
富山県	1066150	4247.61	30,500	390900
石川県	1154105	4186.09	41,600	455000

6. Excelを用いた重回帰分析による予測

- 「回帰分析」ウィンドウ内で以下の順で操作
- ⑤「入力Y範囲」内でクリック
 - ⑥目的変数（住宅数）のG列の数値を全てドラッグして選択
 - ⑦「入力X範囲」内でクリック
 - ⑧説明変数（人口、面積、地価）のC～E列の数値を全てドラッグして選択
 - ⑨「OK」ボタンをクリック

回帰分析

入力元

入力 Y 範囲(Y): ⑤

入力 X 範囲(X): ⑦

☐ ラベル(L) ☐ 定数に 0 を使用(Z)

☐ 有意水準(Q) 95 %

出力オプション

☐ 一覧の出力先(S):

☒ 新規ワークシート(P):

☐ 新規ブック(W)

残差

⑨ OK

キャンセル

ヘルプ(H)

人口

面積

地価

住宅数

原因となっ
ている変数

原因を受けて発生
した結果の変数

⑧(説明変数)

⑥(目的変数)

	A	B	C	D	E	F	G	H
			人口(人)	面積(km2)	地価(円/m2)		住宅数(戸)	
1	2015							
2		北海道	5722908	83424.31	18,000		2416700	
3		青森県	1307942	9645.59	16,700		501500	
4		岩手県	1280046	15275.01	24,500		483600	
5		宮城県	2333952	7282.22	34,000		953600	
6		秋田県	1022940	11637.54	14,200		383800	
7		山形県	1123440	9323.15	19,200		393200	
8		福島県	1914561	13783.74	22,500		731100	
9		茨城県	2916834	6097.06	32,800		1126600	
10		栃木県	1974333	6408.09	33,200		761400	
11		群馬県	1972943	6362.28	30,700		786600	
12		埼玉県	7266615	3797.75	105,400		3023300	
13		千葉県	6222705	5157.65	71,500		2635200	
14		東京都	13515190	2190.93	323,800		6805500	
15		神奈川県	9126281	2415.83	173,700		4000000	
16		新潟県	2304149	12584.1	26,500		844300	
17		富山県	1066150	4247.61	30,500		390900	
18		石川県	1154105	4186.09	41,600		455000	

回帰分析

\$G\$2:\$G\$48

6. Excelを用いた重回帰分析による予測

◆ 新たにワークシートが作成され、以下のような重回帰分析結果が表示されます

回帰統計	
重相関 R	0.997919
重決定 R2	0.995843
補正 R2	0.995553
標準誤差	85493.92
観測数	47

1に近いほど強い正の相関

分散分析表

	自由度	変動	分散	観測された分散比	有意 F
回帰	3	7.53E+13	2.51E+13	3433.342	3.39E-51
残差	43	3.14E+11	7.31E+09		
合計	46	7.56E+13			

5%未満なので有意性がある.
(有意水準5%の場合)

	係数	標準誤差	t	P-値	下限 95%	上限 95%	下限 95.0%	上限 95.0%
切片	-162346	20593.27	-7.88345	6.94E-10	-203876	-120816	-203876	-120816
人口	0.385045	0.011885	32.39788	7.65E-32	0.361076	0.409013	0.361076	0.409013
面積	3.422059	1.317157	2.598065	0.012788	0.765759	6.078359	0.765759	6.078359
地価	4.659682	0.623157	7.47754	2.64E-09	3.402966	5.916398	3.402966	5.916398

6. Excelを用いた重回帰分析による予測

◆ ランダムフォレストによる結果とExcelによる重回帰分析結果を R^2 値で比較します

```
MSE train : 53116892499.781, test : 8811853724.133  
MSE train : 0.974, test : 0.987
```

結果(コンソール)表示画面

回帰統計	
重相関 R	0.997919
重決定 R2	0.995843
補正 R2	0.995553
標準誤差	85493.92
観測数	47

1に近いほど強い正の相関

この分析では、ランダムフォレストに比べて、Excelでの分析の方がより精度が高い（より相関がある）結果を示しています