index

January 29, 2022

1 Interactions - Lab

1.1 Introduction

In this lab, you'll explore interactions in the Ames Housing dataset.

1.2 Objectives

You will be able to: - Implement interaction terms in Python using the `sklearn` and `statsmodels` packages - Interpret interaction variables in the context of a real-world problem

1.3 Build a baseline model

You'll use a couple of built-in functions, which we imported for you below:

[1]:
```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

If you still want to build a model in the end, you can do that, but this lab will just focus on finding meaningful insights in interactions and how they can improve R^2 values.

[2]:
```
regression = LinearRegression()
```

Create a baseline model which includes all the variables we selected from the Ames housing data set to predict the house prices. Then use 10-fold cross-validation and report the mean R^2 value as the baseline R^2.

[11]:
```
ames = pd.read_csv('ames.csv')

continuous = ['LotArea', '1stFlrSF', 'GrLivArea', 'SalePrice']
categoricals = ['BldgType', 'KitchenQual', 'SaleType', 'MSZoning', 'Street',
                'Neighborhood']



ames_cont = ames[continuous]
ames_dummies = pd.get_dummies(ames[categoricals], prefix = categoricals)
```

```python
data = pd.concat([ames_cont, ames_dummies], axis = 1)

Y =  data["SalePrice"]
X = data.drop("SalePrice", axis = 1)


kfold = KFold(n_splits=10, shuffle=True, random_state=1)

baseline = np.mean(cross_val_score(regression, X, Y, scoring='r2', cv=kfold))

baseline
```

[11]:

```
      LotArea  1stFlrSF  GrLivArea  SalePrice  BldgType_1Fam  BldgType_2fmCon  \
39       6040      1152       1152      82000              0                0
97      10921       960        960      94750              1                0
98      10625       835        835      83000              1                0
110      9525      1216       1855     136900              1                0
117      8536      1125       1125     155000              1                0
...       ...       ...        ...        ...            ...              ...
1384     9060       698       1258     105000              1                0
1423    19690      1575       2201     274970              1                0
1448    11767       796       1346     112000              1                0
1452     3675      1072       1072     145000              0                0
1459     9937      1256       1256     147500              1                0

      BldgType_Duplex  BldgType_Twnhs  BldgType_TwnhsE  KitchenQual_Ex  ...  \
39                  1               0                0               0  ...
97                  0               0                0               0  ...
98                  0               0                0               0  ...
110                 0               0                0               0  ...
117                 0               0                0               0  ...
...               ...             ...              ...             ...  ...
1384                0               0                0               0  ...
1423                0               0                0               0  ...
1448                0               0                0               0  ...
1452                0               0                1               0  ...
1459                0               0                0               0  ...

      Neighborhood_NoRidge  Neighborhood_NridgHt  Neighborhood_OldTown  \
39                       0                     0                     0
97                       0                     0                     0
98                       0                     0                     0
110                      0                     0                     0
117                      0                     0                     0
...                    ...                   ...                   ...
1384                     0                     0                     0
```

```
1423                    0                    0                     0
1448                    0                    0                     0
1452                    0                    0                     0
1459                    0                    0                     0

      Neighborhood_SWISU  Neighborhood_Sawyer  Neighborhood_SawyerW  \
39                     0                    0                     0
97                     0                    0                     0
98                     0                    0                     0
110                    0                    0                     0
117                    0                    0                     0
…                    …                    …                     …
1384                   0                    0                     0
1423                   0                    0                     0
1448                   0                    0                     0
1452                   0                    0                     0
1459                   0                    0                     0

      Neighborhood_Somerst  Neighborhood_StoneBr  Neighborhood_Timber  \
39                       0                     0                    0
97                       0                     0                    0
98                       0                     0                    0
110                      0                     0                    0
117                      0                     0                    0
…                      …                     …                    …
1384                     0                     0                    0
1423                     0                     0                    0
1448                     0                     0                    0
1452                     0                     0                    0
1459                     0                     0                    0

      Neighborhood_Veenker
39                       0
97                       0
98                       0
110                      0
117                      0
…                      …
1384                     0
1423                     0
1448                     0
1452                     0
1459                     0

[100 rows x 54 columns]
```

1.4 See how interactions improve your baseline

Next, create all possible combinations of interactions, loop over them and add them to the baseline model one by one to see how they affect the R^2. We'll look at the 3 interactions which have the biggest effect on our R^2, so print out the top 3 combinations.

You will create a `for` loop to loop through all the combinations of 2 predictors. You can use `combinations` from itertools to create a list of all the pairwise combinations. To find more info on how this is done, have a look here.

Since there are so many different neighbourhoods we will exclude

[4]:
```
from itertools import combinations
combo = combinations(X.columns, 2)
```

[8]:
```
# code to find top interactions by R^2 value here
# dic = {}
# XX = X.copy()
# for i, (a,b) in enumerate(combo):
#     XX[a + "*" + b] = X[a]*X[b]
#     results = np.mean(cross_val_score(regression, XX,Y, scoring='r2',␣
 ↪cv=kfold))
#     dic[np.round(results,3)] = (a,b)
#     if i%50 == 0:
#         print(i)
```

[9]:
```
# lis = sorted(dic.keys(), reverse = True)
# lis
```

It looks like the top interactions involve the Neighborhood_Edwards feature so lets add the interaction between LotArea and Edwards to our model.

We can interpret this feature as the relationship between LotArea and SalePrice when the house is in Edwards or not.

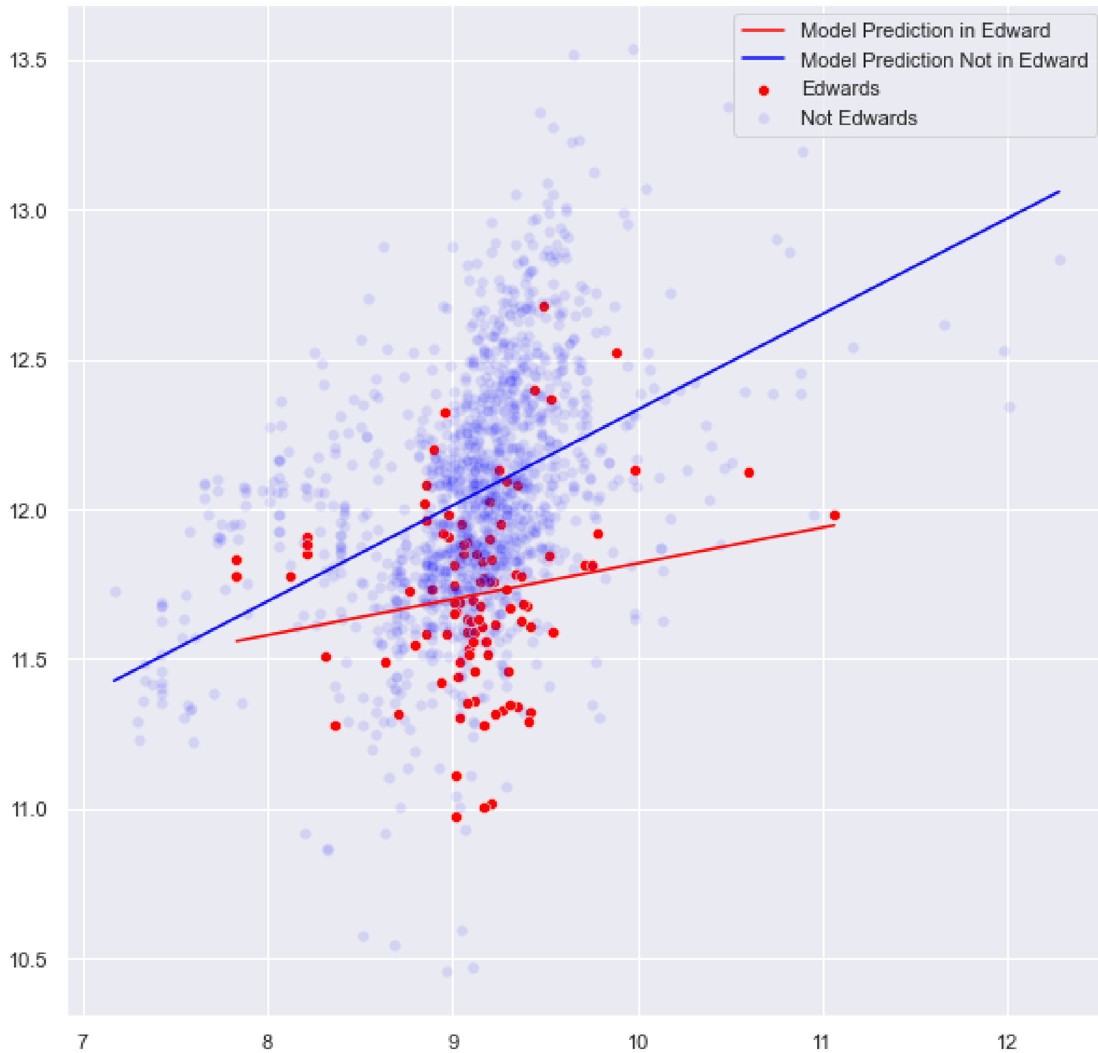
1.5 Visualize the Interaction

Separate all houses that are located in Edwards and those that are not. Run a linear regression on each population against `SalePrice`. Visualize the regression line and data points with price on the y axis and LotArea on the x axis.

[ ]:
```

```

[61]:
```
# Visualization code here
import seaborn as sns
sns.set(rc={"figure.figsize":(10, 10)})


to_pick = ["LotArea", "SalePrice"]
```

```
Edwards = data.loc[data["Neighborhood_Edwards"] == 1][to_pick]
Not_Edwards = data.loc[data["Neighborhood_Edwards"] == 0][to_pick]

model_edward = LinearRegression()
model_not_edward = LinearRegression()

X1 = np.reshape(np.log(np.array(Edwards["LotArea"])),(-1,1))
X2 = np.reshape(np.log(np.array(Not_Edwards["LotArea"])),(-1,1))

Y1 = np.log(Edwards["SalePrice"])
Y2 = Y = np.log(Not_Edwards["SalePrice"])

model_edward.fit(X1, Y1)
model_not_edward.fit(X2, Y2)

prediction_edward = model_edward.predict(X1)
prediction_not_edward = model_not_edward.predict(X2)

sns.scatterplot(x = np.log(np.array(Edwards["LotArea"])),
                y = np.array(Y1), color = "red", label = "Edwards")

sns.scatterplot(x = np.log(np.array(Not_Edwards["LotArea"])),
                y = np.array(Y2), color = "blue", label = "Not Edwards",
               alpha = 0.1);

sns.lineplot(x = np.log(np.array(Edwards["LotArea"])),
             y = prediction_edward, color = "red",
             label = "Model Prediction in Edward")

sns.lineplot(x = np.log(np.array(Not_Edwards["LotArea"])),
             y = prediction_not_edward, color = "blue",
             label = "Model Prediction Not in Edward");
```



1.6 Build a final model with interactions

Use 10-fold cross-validation to build a model using the above interaction.

[63]:
```
# code here

Y =  data["SalePrice"]
X = data.drop("SalePrice", axis = 1)

X["interaction"] = X["Neighborhood_Edwards"] * X["LotArea"]



kfold = KFold(n_splits=10, shuffle=True, random_state=1)
```

```
baseline_final = np.mean(cross_val_score(regression, X, Y, scoring='r2',␣
 ↪cv=kfold))

baseline_final
```

[63]: 0.8093314939294032

Our R^2 has increased considerably! Let's have a look in `statsmodels` to see if this interactions are significant.

[65]:
```
# code here
import statsmodels.api as sm

X_added = sm.add_constant(X)
model = sm.OLS(Y,X_added)
results = model.fit()

print(results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              SalePrice   R-squared:                       0.835
Model:                            OLS   Adj. R-squared:                  0.829
Method:                 Least Squares   F-statistic:                     148.6
Date:                Sat, 29 Jan 2022   Prob (F-statistic):               0.00
Time:                        02:21:35   Log-Likelihood:                -17229.
No. Observations:                1460   AIC:                         3.456e+04
Df Residuals:                    1411   BIC:                         3.482e+04
Df Model:                          48
Covariance Type:            nonrobust
================================================================================
========
                           coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
--------
const                 2.082e+04   4252.352      4.896      0.000    1.25e+04
2.92e+04
LotArea                  0.6108      0.103      5.916      0.000       0.408
0.813
1stFlrSF                35.0664      3.288     10.664      0.000      28.616
41.517
GrLivArea               58.1426      2.405     24.171      0.000      53.424
62.861
BldgType_1Fam         2.602e+04   2638.692      9.861      0.000    2.08e+04
3.12e+04
BldgType_2fmCon       9217.1096   5696.381      1.618      0.106   -1957.176
```

```
2.04e+04
BldgType_Duplex      -6841.6893   4591.768     -1.490      0.136   -1.58e+04
2165.737
BldgType_Twnhs       -1.026e+04   5569.935     -1.842      0.066   -2.12e+04
668.832
BldgType_TwnhsE       2679.5402   3722.000      0.720      0.472   -4621.708
9980.788
KitchenQual_Ex        5.641e+04   3780.490     14.921      0.000     4.9e+04
6.38e+04
KitchenQual_Fa       -2.433e+04   4554.502     -5.342      0.000   -3.33e+04
-1.54e+04
KitchenQual_Gd        2308.8324   2294.526      1.006      0.314   -2192.217
6809.881
KitchenQual_TA       -1.357e+04   2152.934     -6.302      0.000   -1.78e+04
-9345.160
SaleType_COD         -1.794e+04   6477.445     -2.769      0.006   -3.06e+04
-5228.796
SaleType_CWD          3318.0182   1.54e+04      0.216      0.829   -2.69e+04
3.35e+04
SaleType_Con          4.498e+04   2.17e+04      2.075      0.038    2451.689
8.75e+04
SaleType_ConLD       -1510.0213   1.11e+04     -0.136      0.892   -2.33e+04
2.02e+04
SaleType_ConLI        -889.2966   1.39e+04     -0.064      0.949   -2.82e+04
2.64e+04
SaleType_ConLw       -7014.0283   1.39e+04     -0.503      0.615   -3.44e+04
2.03e+04
SaleType_New          1.434e+04   5483.489      2.614      0.009    3579.608
2.51e+04
SaleType_Oth         -9223.3553   1.76e+04     -0.525      0.600   -4.37e+04
2.52e+04
SaleType_WD          -5248.4845   4613.283     -1.138      0.255   -1.43e+04
3801.147
MSZoning_C (all)     -1.969e+04   1.05e+04     -1.875      0.061   -4.03e+04
912.451
MSZoning_FV            1.83e+04   7715.456      2.371      0.018    3161.253
3.34e+04
MSZoning_RH          -1635.0973   7891.083     -0.207      0.836   -1.71e+04
1.38e+04
MSZoning_RL           9561.0427   3920.921      2.438      0.015    1869.582
1.73e+04
MSZoning_RM           1.428e+04   4419.309      3.232      0.001    5615.470
2.3e+04
Street_Grvl           1.203e+04   8875.663      1.356      0.175   -5377.530
2.94e+04
Street_Pave           8785.4294   6111.477      1.438      0.151   -3203.130
2.08e+04
Neighborhood_Blmngtn   1.02e+04   8807.813      1.158      0.247   -7077.770
```

```
2.75e+04
Neighborhood_Blueste  1.821e+04   2.29e+04      0.794      0.427   -2.68e+04
6.32e+04
Neighborhood_BrDale  -1003.6293   9697.145     -0.103      0.918       -2e+04
1.8e+04
Neighborhood_BrkSide -3.116e+04   5053.420     -6.166      0.000   -4.11e+04
-2.12e+04
Neighborhood_ClearCr -1.337e+04   6616.187     -2.020      0.044   -2.63e+04
-386.442
Neighborhood_CollgCr  2241.7603   3289.701      0.681      0.496   -4211.471
8694.992
Neighborhood_Crawfor  2102.0934   4845.481      0.434      0.664   -7403.028
1.16e+04
Neighborhood_Edwards  3.228e+04   6269.634      5.149      0.000        2e+04
4.46e+04
Neighborhood_Gilbert -1491.7134   4183.056     -0.357      0.721   -9697.392
6713.965
Neighborhood_IDOTRR  -4.174e+04   7493.343     -5.571      0.000   -5.64e+04
-2.7e+04
Neighborhood_MeadowV -1.728e+04   8868.480     -1.949      0.052   -3.47e+04
113.443
Neighborhood_Mitchel -9966.6701   4957.856     -2.010      0.045   -1.97e+04
-241.108
Neighborhood_NAmes   -2.442e+04   3017.409     -8.092      0.000   -3.03e+04
-1.85e+04
Neighborhood_NPkVill  1.652e+04   1.13e+04      1.461      0.144   -5655.498
3.87e+04
Neighborhood_NWAmes  -1.636e+04   4279.043     -3.824      0.000   -2.48e+04
-7969.153
Neighborhood_NoRidge  6.052e+04   5779.198     10.472      0.000    4.92e+04
7.19e+04
Neighborhood_NridgHt  5.279e+04   4722.046     11.179      0.000    4.35e+04
6.21e+04
Neighborhood_OldTown  -4.98e+04   4947.695    -10.064      0.000   -5.95e+04
-4.01e+04
Neighborhood_SWISU   -4.812e+04   6950.579     -6.924      0.000   -6.18e+04
-3.45e+04
Neighborhood_Sawyer  -2.517e+04   4291.324     -5.864      0.000   -3.36e+04
-1.67e+04
Neighborhood_SawyerW -5213.2848   4655.479     -1.120      0.263   -1.43e+04
3919.121
Neighborhood_Somerst  1.426e+04   7448.993      1.914      0.056    -355.522
2.89e+04
Neighborhood_StoneBr  6.415e+04   7064.721      9.081      0.000    5.03e+04
7.8e+04
Neighborhood_Timber   6900.0704   5705.660      1.209      0.227   -4292.418
1.81e+04
Neighborhood_Veenker  2.572e+04   9988.488      2.575      0.010    6130.720
```

```
4.53e+04
interaction              -7.1552      0.513    -13.959      0.000      -8.161
-6.150
==============================================================================
Omnibus:                      381.039   Durbin-Watson:                   1.945
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             3465.080
Skew:                           0.947   Prob(JB):                         0.00
Kurtosis:                      10.306   Cond. No.                     5.53e+17
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 1.02e-24. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

What is your conclusion here?

```
[ ]: # formulate your conclusion
```

1.7 Summary

You should now understand how to include interaction effects in your model! As you can see, interactions can have a strong impact on linear regression models, and they should always be considered when you are constructing your models.