Week03

Assignment

Assignment 02

- Build linear regression model
  - Among numeric variables, select input variables
    - Describe reasons for variable selection
  - You can apply variable transformation
    - Describe reasons of variable transformation
  - You can discard some rows satisfying specific conditions
    - Specify conditions
- Summarize the process and result using Power Point
  - Some students have to create video clip to explain their results

- Submit both slide and python code

Add constant

- When you use OLS of statsmodels with intercept term (β_0), you should add a constant column

```
import statsmodels.api as sm

X2=sm.add_constant(X)
```

Model without constant

OLS Regression Results

Dep. Variable:	price	R-squared (uncentered):	0.860
Model:	OLS	Adj. R-squared (uncentered):	0.860
Method:	Least Squares	F-statistic:	1.209e+04
Date:	Mon, 14 Sep 2020	Prob (F-statistic):	0.00
Time:	20:14:43	Log-Likelihood:	-2.9879e+05
No. Observations:	21613	AIC:	5.976e+05
Df Residuals:	21602	BIC:	5.977e+05
Df Model:	11		
Covariance Type:			

	coef					0.975]
bedrooms	-4.634e+04				4	-4.2e+04
bathrooms	2205.5631				0	9354.626
sqft_lot	0.0749				9	0.188
floors	2.027e+04				4	2.84e+04
waterfront	7.592e+05				5	7.97e+05
sqft_above	246.2137				0	254.217
sqft_basement	288.2430				5	297.941
yr_built	-10.8950	4.282	-2.544	0.011	-19.289	-2.501
yr_renovated	68.5709	4.169	16.447	0.000	60.399	76.743
sqft_living15	78.6768	3.850	20.435	0.000	71.130	86.223
sqft_lot15	-0.8858	0.088	-10.023	0.000	-1.059	-0.713

It is uncentered R^2,
it is not R^2

Model with constant

OLS Regression Results

Dep. Variable:	price	R-squared:	0.598
Model:	OLS	Adj. R-squared:	0.598
Method:	Least Squares	F-statistic:	2923.
Date:	Mon, 14 Sep 2020	Prob (F-statistic):	0.00
Time:	20:16:16	Log-Likelihood:	-2.9775e+05
No. Observations:	21613	AIC:	5.955e+05
Df Residuals:	21601	BIC:	5.956e+05
Df Model:	11		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	6.418e+06	1.38e+05	46.627	0.000	6.15e+06	6.69e+06
bedrooms	-5.813e+04	2150.015	-27.038	0.000	-6.23e+04	-5.39e+04
bathrooms	6.615e+04	3737.401	17.701	0.000	5.88e+04	7.35e+04
sqft_lot	0.0371	0.055	0.671	0.502	-0.071	0.145
floors	5.498e+04	4020.903	13.673	0.000	4.71e+04	6.29e+04
waterfront	7.247e+05	1.86e+04	39.027	0.000	6.88e+05	7.61e+05
sqft_above	239.6824	3.895	61.538	0.000	232.048	247.317
sqft_basement	243.7353	4.812	50.654	0.000	234.304	253.167
yr_built	-3338.9292	71.492	-46.703	0.000	-3479.059	-3198.799
yr_renovated	11.9013	4.156	2.864	0.004	3.756	20.047
sqft_living15	90.4224	3.679	24.581	0.000	83.212	97.633
sqft_lot15	-0.7360	0.084	-8.731	0.000	-0.901	-0.571

Comparation using MSE

- Using House sales prices in King County data

```
varlist=['bedrooms','bathrooms','sqft_lot','floors','waterfront','sqft_abo
ve','sqft_basement','yr_built','yr_renovated','sqft_living15','sqft_lot15']
X=house[varlist]
y=house['price']
X2=sm.add_constant(X)

reg1=sm.OLS(y,X)
result1=reg1.fit()
y_pred1=result1.predict(X)
reg2=sm.OLS(y,X2)
result2=reg2.fit()
y_pred2=result2.predict(X2)
np.mean((y-y_pred1)**2)
np.mean((y-y_pred2)**2)
```

```
np.mean((y-y_pred1)**2)
59610125773.46967

np.mean((y-y_pred2)**2)
54159131608.177925
```