HW2

Part I

In this homework, you will build an Jupyter notebook for predicting house prices, using data from King County, USA, the region where the city of Seattle is located.

1. Add Extra Features

Add 4 new variables in the data set.

- 'bedrooms_squared' = 'bedrooms'*'bedrooms'
- 'bed_bath_rooms' = 'bedrooms'*'bathrooms'
- 'log_sqft_living' = log('sqft_living')
- 'lat_plus_long' = 'lat' + 'long'

Before we continue let's explain these new variables:

- Squaring bedrooms will increase the separation between not many bedrooms (e.g. 1) and lots of bedrooms (e.g. 4) since 1^2 = 1 but 4^2 = 16. Consequently this variable will mostly affect houses with many bedrooms.
- Bedrooms times bathrooms is what's called an "interaction" variable. It is large when both of them are large.
- Taking the log of square feet has the effect of bringing large values closer together and spreading out small values.
- Adding latitude to longitude is non-sensical but we will do it anyway (you'll see why)

What are the mean (arithmetic average) values of your 4 new variables? (round to 2 digits)

2. Train-Test Split

Split the data into training set (80%) and test set (20%) with random state = 0.

3. Estimate the regression coefficients/weights for predicting 'price' for the following three models.

- Model 1: 'sqft_living', 'bedrooms', 'bathrooms', 'lat', and 'long'
- Model 2: 'sqft_living', 'bedrooms', 'bathrooms', 'lat','long', and 'bed_bath_rooms'
- Model 3: 'sqft_living', 'bedrooms', 'bathrooms', 'lat','long', 'bed_bath_rooms', 'bedrooms_squared', 'log_sqft_living', and 'lat_plus_long'

You'll note that the three models here are "nested" in that all of the features of the Model 1 are in Model 2 and all of the features of Model 2 are in Model 3.

What is the sign (positive or negative) for the coefficient/weight for ‘bathrooms’ in Model 1?
What is the sign (positive or negative) for the coefficient/weight for ‘bathrooms’ in Model 2?

Is the sign for the coefficient the same in both models? Think about why this might be the case.

4. Now using your three estimated models compute the RSS (Residual Sum of Squares) on the Training data. Which model (1, 2 or 3) had the lowest RSS on TRAINING data?

5. Now using your three estimated models compute the RSS on the Testing data. Which model (1, 2, or 3) had the lowest RSS on TESTING data?

6. Train-Validate-Test

Split the original data into train-validate set (90%) and test set (10%) using random state = 0.

Use GridSearchCV and 5-fold cross validation to compare the performance of the following polynomial regression models on the train-validation set.

Build the following polynomial regression models to predict house prices using just ‘sqft_living’

Polynomial degree in [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]

Which model is the best? Report best model’s performance on the test set.

Part II

1. Load the breast cancer dataset. Import the functions `load_breast_cancer` from `sklearn.datasets`.

2. Split the data into train-validate set (90%) and test set (10%) using random_state = 0.

3. Split the train-validate set into training set (80%) and validation set (20%) using random_state = 0. Build KNN models with n_neighbors = {1, 3, 5, 7, 9} using the training set. Compare their performance on the validation set and pick the best model. Report the best model’s performance on the test set.

4. Use GridSearchCV and 5-fold cross validation to build and compare KNN models with n_neighbors = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19} using the train-validate set. Report the best model’s performance on the test set.