Regression Week 2: Multiple Linear Regression Assignment 1

Predicting House Prices (Multiple Variables)

In this notebook you will use data on house sales in King County to predict prices using multiple regression. The first assignment will be about exploring multiple regression in particular exploring the impact of adding features to a regression and measuring error. In the second assignment you will implement a gradient descent algorithm. In this assignment you will:

- Use SFrames to do some feature engineering
- Use built-in GraphLab Create (or otherwise) functions to compute the regression weights (coefficients)
- Given the regression weights, predictors and outcome write a function to compute the Residual Sum of Squares
- Look at coefficients and interpret their meanings
- Evaluate multiple models via RSS

If you are doing the assignment with IPython Notebook

An IPython Notebook has been provided below to you for this assignment. This notebook contains the instructions, quiz questions and partially-completed code for you to use as well as some cells to test your code.

What you need to download

If you are using GraphLab Create:

- Download the King County House Sales data In SFrame format: kc_house_data.gl.zip (https://eventing.coursera.org/api/redirectStrict /M2ETkKvi40MXKHkmO8T06aSEmVptrkeCoJiqegBKDBc8QFJ7n6ffv4bod3Yz4GsLhj8LBHzwAp2KVAAm84SR9Q.Yx NrhdBa5Wz1NI_S3he_kw.noxf4Lmf9ldbM_W8HqE1OLNQjN3D7A9zIswdYylr2fKnJNSNp2QlL_pMDe8exQbUT68Nq DLix3ieti2ekKsugrSguAR-nWiSg8XQI8WyxfHQKV8tqiVA7cjCk8YeqGVaM5rbtix0t-TSWClaFYOB_glhUJ1m47z4J8V6qg3P5wzECE9Mlb0QkIUqx2ybjFnY0RRQO6_PTiwbsAqQ1eKLvlJJ84Dqc0FHEaa19i MEoea4D1MoWFPvJ1qn6pK5bkRj1mJ6UgTicydxXWsT-ssj6gw9hm4YFjoXTgwCVGnCkwOA4c-VKULxg5lwa2WYOc-wiegQ5N1gMYY9LvA_HaYfpz8U49oMkRQFe0feabu6gEb7WuNZr213kpo6AtSA9dL5c18VME0Jq3mOZFKVDzuGdEo lREP-G9iVGT9Eh8R8D1v-xJh1LSzeRvZt8E7Lvb4q)

- Download the companion IPython Notebook: week-2-multiple-regression-assignment-1-blank.ipynb (https://eventing.coursera.org/api/redirectStrict/cfsr5ZTYltw31yc3cbFl0p78YUTIF8CVdnfA-wvEZHDjmpZ2fWvTQZoNu5ES1MrZPF_LFbF9V4Jimk9d-qsiCQ.AgB918iqmn4ajPBruCScSA.jv-Y6qkITABUjnPoy83UNe5aQ9cGehOLPMSJaS25IxstO60fFwbPHyufqznK2Iiw10VudJN-lhu-7vn0X7v4m0Y153jY4E6yfSGvO4wD8c6KbkjkJWY4W4Rk0npn9BRkfyZsiJqxs-5CTqgiorBzJfc5AphCQykNxD2Tik3OZw74pYk4WAfHebvdg7gmreg_IHmlyaD_0uL9h5J0q6N60Pn3qgZwFHCnaqrXU -JhWOJhZQLjv_WoxCqSzXoY2yutvw1-OIz4uvOIkNnFCp_clRe3wXeK3cF5Cqam9SPeavw3rjxTvG-8d4IOoz4k_rkmO7mhWUMwliX6oSr4Q2jdVKdnRbcVNxdrnGXibCWOmbE7LstoLsTRJgEHZwS7HpcsMpY2s5Z_S3R dnvYuRkzIcMmyo7HxQL6BgSmtku4f2ut9OsuPZHYZAwdosVxZKKs1aZpSB8j95ecv4O5FYUAvn3ItWJvHtMD-Q4imbpJy1VIpcB8FYbPoCq2NiAA2f5RrOGDGc1UBLaHNYFjblfJBoA)

- Save both of these files in the same directory (where you are calling IPython notebook from) and unzip the data file.

If you are not using GraphLab Create:

- Download the King County House Sales data csv file: kc_house_data.csv (https://eventing.coursera.org /api/redirectStrict/LX05qEhYgulYZd_H9aLSmBLBfyRewVkJI_PwajjqcSqXR78APeqy8W10-w5TRj2Yzgi3AFRLcBtemZ4zUA0MNg.D-q2RQRva2mHBsemBAoPUg.BwQcX2VgwPbzdF4E14WkNR6_v4BacP2uMkgIsHUVgVPAPKE_kutW4WLvCqIlt8DmK qUhFQB0IWeqWsrPubzo1fgIpzFtz_4vt-6kdvGLvEfbPMKX4e-k9VEfqxr7A6wHzGBps-L9I0NVe9m2DpN2g8iU-- OyMUDaKVP3YF4xnfi4G_TLuBn56YGOBpsEk5GPaCBWCWmy2FOOayF11p8kAPlSel59-23lHViX-go1e0pYlBl4XF4vmIG0UPfyzRUBPXRQvGzD4xgoEhqAF1vECSS0zN4rF0WUBvmIPxeRSDVlwgNV9qf0H9oI0s0CgSK yJhrTtxjj6JwGIX6HlQTo02hgW28Li1nqEK06n0gie_KFiLmDsJbwDch9nQQO_GQwSYVRIko8jY4mSeMqERdWeZwKLH GiczT7XkStfENbpUB6qHX7KIeLPo9Ik__zcl18)

- Download the King County House Sales training data csv file: kc_house_train_data.csv (https://eventing.coursera.org/api/redirectStrict /-1n7iEDoTIGMd34aTHcmYQQaa2O5v82Peg2M1aoGayR9XsmAM-7tt26r8j9kseTQDK_3_LOw_R6Fn1Vl1CPfKQ.SG2OvQYsx9exHrZxNSVD1g.j9motxN-tlx1_P6xhEIO7z7CZ_OPwkietoWZ87QVsm94eiGm_W3ljw2ecRvRr8wXqVntrZO8NNW4XPFM82FO6VYHbr18ngnQ6 sMrfQI_8KRYMvwJEQs-1EdLtmSwH9vtjNlriniCzblTH7DIysLXNc1q7J2vb_oYZlH3dc5ATAdvFfgcFMZ5meXzTS8AK1TOeA1YEbENbX-DyOAnxRR6N_9CCh-Ld977nBthwzbNu2Ghd5ROy_Za7dQD3FWWEi6qHxEANX7GyBZbWqbk3QVjZ1uSvRv0vVzQrP1jP-Y5Zy0LJObH4PKYS9MobMOaDwsBXQrLf3ojK-uDNigxVDSYZZk3pqC_-STYmaTnFdoH1UxYPFBexSRLLu-czA7tzQWukxuF7Ya8XAXSbXm8OM6gFeybc_8IZrWzo_gfbpfkD9lej8b1ExmYdkkDzxFEHXKySIZE1pRcl6CF62--9id1P w)

- Download the King County House Sales testing data csv file: kc_house_test_data.csv (https://eventing.coursera.org/api/redirectStrict/bloGA0zJXuWH05PJxllty1nzJfyz4gMIroTs39Pb1c-9bNkp0vxNJr77t6ffK4sy3tfMYIXoTBCtQeGxzjL4sA.uJYPPHK-u0q_M9GPa8cKxw.8AZZ7UF5dXgi3iwDDFzpnpjZF4Nb1T58V0uBKW7C9761Ni01BYi4DAAOBo3w1hf6SXfy-JpxKEw8v895fflT6tpX6gYa2ElbQPL7vztzEMl4D9_wUX7QZaK__uhXUINJPXRbKu-E5gzin2p_--8ovUnf3-RBUCpxPfEDtQb1saI9cO05ITEIuI4VE0VVuCzmonx6LwA8AeZVE7q_gexlRCE5LAOyly8Jqmpuj1uK4A4AFCNmYi6zZe sCrjQoEmGLrNyzPO15uviVnWLIg6TQ3iSNFY3MC8mzcxQxxpfRDGi0C6bsnJWyI_okdXhn0S33otadK0A_KLFq9q8TiJj - Gfnrl0QBUkopQQqLmGHl1U_fAsgDAeq0ykookvzWmcddXlcZ9XMvp7QbaM0akH_sLfJHbIqNWtgkIh_vYTjWEyu6Td 2KdAcqX5df3KES5tJ-)

Useful resources

You may need to install the software tools or use the free Amazon EC2 machine. Instructions for both options are provided in the reading for Module 1.

If instead you are using other tools to do your homework

You are welcome, however, to write your own code and use any other libraries, like Pandas or R, to help you in the process. If you would like to take this path, follow the instructions below.

1. If you are using SFrame, import graphlab and load in the house data, otherwise you can also download the csv. (Note that we will be using the training and testing csv files provided) e.g in python with SFrames:

```
sales = graphlab.SFrame('kc_house_data.gl/')
```

2. Split into training and test data Use this command to set the same seed for everyone: e.g. in python with

SFrames:

```
train_data,test_data = sales.random_split(.8,seed=0)
```

For those students not using SFrames please download the training and testing data csv files.

From now on we will train the models using train_data. It will be important that we use the same split here to ensure the results are the same.

3. Although we often think of multiple regression as including multiple different features (e.g. # of bedrooms, square feet, and # of bathrooms) but we can also consider transformations of existing variables e.g. the log of the square feet or even "interaction" variables such as the product of bedrooms and bathrooms. Add 4 new variables in both your train_data and test_data.

- '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)

For those students not using SFrames you should first download and import the training and testing data sets provided and then add the four new variables each to both data sets (training and testing)

4. Quiz Question: what are the mean (arithmetic average) values of your 4 new variables on TEST data? (round to 2 digits)

5. Use graphlab.linear_regression.create (or any other regression library/function) to estimate the regression coefficients/weights for predicting 'price' for the following three models:(In all 3 models include an intercept -- most software does this by default).

- 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', '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.

If you use graphlab.linear_regression.create() to estimate these models please ensure that you set validation_set = None. This way you will get the same answer every time you run the code.

Learn all three models on the TRAINING data set. Save your model results for quiz questions later.

6. Quiz Question: What is the sign (positive or negative) for the coefficient/weight for 'bathrooms' in Model 1?

7. Quiz Question: What is the sign (positive or negative) for the coefficient/weight for 'bathrooms' in Model 2?

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

9. Now using your three estimated models compute the RSS (Residual Sum of Squares) on the Training data.

10. Quiz Question: Which model (1, 2 or 3) had the lowest RSS on TRAINING data?

11. Now using your three estimated models compute the RSS on the Testing data

12. Quiz Question: Which model (1, 2, or 3) had the lowest RSS on TESTING data?

13. Did you get the same answer for 9 and 11? Think about why this might be the case.