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/3kjzgMTfqKS3y3qrvCBklH0HdNMT9KKjp ifEMqvK8DIIP9cyOuCiIbb2VeaTOi6RwatECgIxOW1ddCCgX6k9Bg-Xrgo_iKRHNbi8AiLf32wl

IIEMqvK8DiIP9cvQuGIJbb2vea1Qj6PwatECqJxQWTddCCqX6K9Bg.Yrge_JKPHNfIJ8AILI3ZwL
g.T6ClDmjMwsqHuTlfSO7Ex46QcAYhrANe1PLznEdWvFH403FPUF1oK_PQ3z6a-
L_7b57gNuF7Ngh-nDZvEmkKcdJub8WXUZ4l-kGeVxm-5-
y1oIcVKGM_4P9oOx81R65095umdJvmlILrFHT_moDMH0395p6brtPrkkHTky4Jt6iF75xhozS
bUhQMI-t-VYysnjGH8kR1TArb667GHFrwoQMA4nT0btrlKEx3Vhv-
Q3MZNmMJq5fBO8CtOKUf6irEjSMWsJVw-8gt-
NTSO2Bf7iH70WLj5PeftncCTDtteBZRJ5eGo_3KfJdC3Yw0tLQ5Og4ThQ6V9HWkfz909q62q
9u7Q74HFEdXV0TyaIbjZ8ayLxLqAKEaN2DWxNSIFMbcULUUbz7QnRGYk5K8CZTRRuSV85
YHfRfjTmv1YBqH8ZYlXlpOTb-h33V9Y8JyHswJ)

- Download the companion IPython Notebook: week-2-multiple-regression-assignment-1-
  blank.ipynb
  (https://eventing.coursera.org/api/redirectStrict/vGuYZKz_-2b_MCGNPXHQz_PriDVzgg63
  O71_Dexew3QaAwoQefcBEvtOKPR1UTfDy-dB-SQmZ9jliiHXx07CWw.lFG4Rf8L3nhjuZ7HR-
  S_Zw.8JN85Q-
  EyIOHxD6mJ4cYoaiFonsA6XKpH2RyAKPIgm9xzfmeNCTNAK805zlVoPrI96ZmIOcLHb_eDq
  r4zlppKL4m60-_aaO34WZIws0xjttWHONAKMV-
  TDLKKBbvPQM2IUpQW3E1sRMGrKfzTsO6ZWB5MzP_HRSFH0tQ73btPFN5TXHBgUv6R3Vf
  L1Rwwo1GnZx4dwW2bPOvcwxYhVpqIO2bGunRalyQ-
  VahJ_l0HzaLQ9Xwg8sap7dt65y2enlyYX-
  HkpqW43JypRSqLPbnY2GTTL3QOBEB1RCy07d_lk7ZerihI1kn_CNbKSh383HNAOugtJlSdtU
  uhxZv2AFGWjK7ZpmKohrxs33yo-
  iflZzXYoPj1ORlpFowLVPnYu8Rdbf8eqJxac46C9ANwlDJtd9c179TGYLDbZsiXbOMGxuF_Ox9
  dL7mprCGdNlUUmY1eD2J2gOE52b67YyZ9xhCtJlNexud2juJAy0rbbSZtiD1-t-
  oyAdvOyt93EgmKP2_)

- 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/z8U8xiLx9liJOcWyb_8eZ1kQif8Lzwp1Efd
  GY5Uep0pcVQQXef9VJCisfhsl7WE7GUKW7xXbM9TUmYJWxgeCbQ.ylEIVN-
  Mmbmeg3lJaAXrAg.u6PzCJxIGNMJX5N7EqDaqi6H2ZhtsD1H1_fJlCMQ789rWFA-
  dr_A8RakWT5oy-ClkzxAUXtvWV1NLy9rRPboFU0DSMy1tT7CpSrox-
  VUO1WJRaFXHji_AkSdewhShM-
  Lk_zQ5qGT3yJKTFSrSA4wPmm1RiVLecG9XqiX0yF87Scf9mh4AlqbxW4mlmXph_vCis2I5W
  GTZ6ziPTzeT704esiKPsNTx-U6l5timcnWtJJhDBEjfScJdLFT_siAlZefCrnugeYXNOzS-
  ky67kZZgRWreD_xfJYlywKdBXZZchw1Cfy94Jj7pu1E7K-YrDAZb9-
  dOVBmpUIGzrvv7iBVweic_PNX_6wx2u85ma9qE3-
  kdn0btD1PUU4SdXSENaA0FeTz_YbjcVzl-
  8fUgm3YvaEMbTiFd5tJwEV9L7SZL5rqptt1tsNDrxcuJn9SG4VM)

- Download the King County House Sales training data csv file: kc_house_train_data.csv
  (https://eventing.coursera.org/api/redirectStrict/khH1qvlq9sLVdWLA2XK_zyM_zinuSn7B_
  t3AIfxwy_xpO-EZ2gEBYuWMQUF-96jIEJC5o_Ox8xCl5zJ8zZhnpQ.vYt-
  ulToBuQvX4_nHxDmMw_bi1aigV5CVzZOzpuniP00mbBb2sLu3uikgVAf_IYWrVqrbLLkyMCI

uTreBuOvX4_nOxDmMw.nrrajqV5CVzZOzpufjRooTmRn2sLu3ujkqVAr_rXWrVgmLLRVMCL
mTsRPlmoCXwdUzoojyrIVGX3nfMSLvmKA3h2SV0Q2vez9vz73IAsTBv188xZXyNuELaUiCsT
GweyhJ1KzEXUZDxTlM1jYwLKh9kxVWmTR5wnjqNQwGlEhyv9ef_n_WB7nvO3KzAZCLC8fa
A-
EtFfxrPukAL5cSd5qARzNuSaFWuSg429gpMgoPZ97XO1dSwfvKAyUMllx5adTKuyA2nt9Y9
PJmWrC0ICdY9cE35jRFp2R4qFV5B8xXYKZeP7DiyJgi5jyZNUE3X9aPASs95SMb52wKF877fs
ecSRMkEo6AEF-PJ0jzbAKkQXWEam6ciC6kW50Da-1-mzUOV-
xluladvlF8BSr4vwWw3F9zdKu7KExE9DtZO2B5slZ7PnBky29aJp-PXmSPk)

- Download the King County House Sales testing data csv file: kc_house_test_data.csv (https://eventing.coursera.org/api/redirectStrict/856L9l_ow9XaiSmvDDquCcPjVmjrIO79E
  gvpeiCiCLMghf3N5k737qZTKkre5xvJnUOgdiLM_dTGFjijSL0ZdA.jgxSaWjao7gIoebFYu4qDg
  .G7wV79oTzCWZVR1ig1SR7LHXhC0hclDRvjsIghHe90bJMLKLio_0ec-
  AY6NlSXB0b7fCYKed_JN8xPXgxbSV-B3vKvqq0ftjzQ-
  _Mdc7DLJXrz76aCFC4jgmWUgRcWyanwN9zPSJ0B2VSDJ-
  yivDn36ijz8FUQeL9ZsB7IdVF9zrxFDdHYU0hDGRrueJFieJMcZmiHSfrbqeq9XtDF6QLD2eM
  DepFFEl8QWVy30u2LlDKunsO_ICNA5jICjzrFi0KSq5t42F2QAcBFmhYS1fbVkcuSOwukg1Co
  jWLEgtuROltyhcbf1f_W_HgqMKL0YzXKFLF-
  MA5CkwZmRO9tyofnPI0BgErNobGBO2UyytYsUjboNsh32Sj6c3yEPoT_FTqSs4VwlBTH9n5
  Tjc-IsMGOTfSInIrE8RgrnKI335jneLlbuTx0cfwgFPr3Fp8ohy)

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

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

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.