Patrick Walsh
ECE241 Project III
Part I: Written Portion and Part II: All Necessary Components

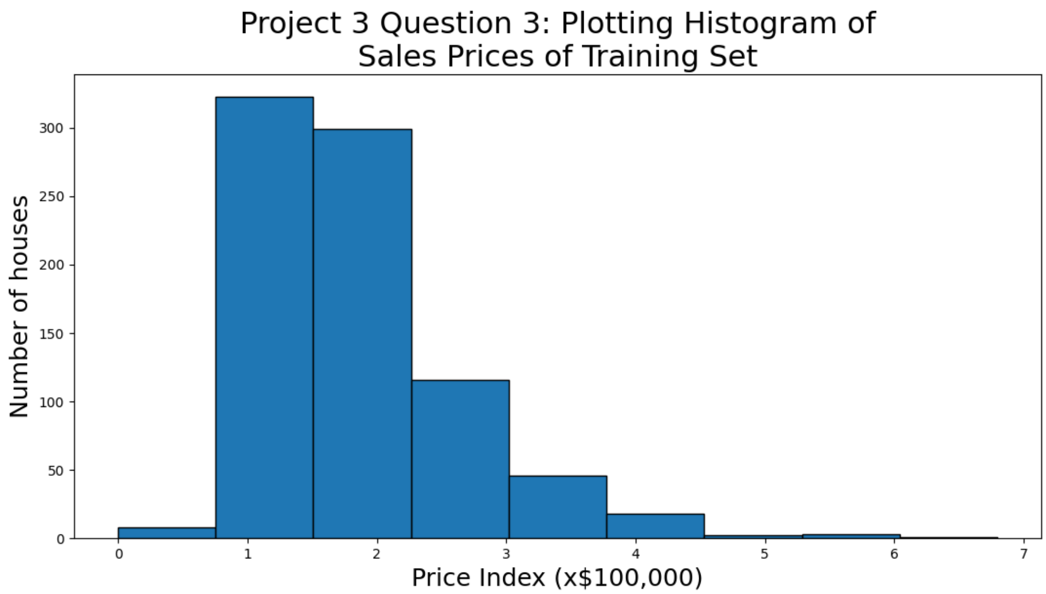
Question 1				Question 2	Question 3
Price	Calculated Price	Percent Error		MSE	MTE
320000	366009.5	14.38%		1163022702	-32983.2
185000	206718	11.74%		(Prefer to be small as possible)	
215000	248825.5	15.73%			
319900	347087	8.50%		MSE better amplifies outliers which is better for ML algorithms. You need the algorithm to work in most cases *including* outlier cases, where MTE doesn't necessarily show as pronounced. Also if one calculated price is w less than its actual and another is w more, those two will cancel in MTE, where they won't in MSE.	
164000	182662	11.38%			
205000	236120	15.18%			
340000	377814	11.12%			
260000	298521.5	14.82%			
227875	257705	13.09%			
438780	483924.5	10.29%			

Question 1				Question 2	Question 3
Price	Calculated Price	Percent Error		MSE	MTE
320000	=SUM(part1!C2:G2)+10*part1!B2	=(B8-part1!G2)/part1!G2		=SUM((A8:A17-B8:B17)^2)/10	=SUM(A8:A17-B8:B17)/10
185000	=SUM(part1!C3:G3)+10*part1!B3	=(B9-part1!G3)/part1!G3		(Prefer to be small as possible)	
215000	=SUM(part1!C4:G4)+10*part1!B4	=(B10-part1!G4)/part1!G4			
319900	=SUM(part1!C5:G5)+10*part1!B5	=(B11-part1!G5)/part1!G5		MSE better amplifies outliers which is better for ML algorithms. You need the algorithm to work in most cases *including* outlier cases, where MTE doesn't necessarily show as pronounced. Also if one calculated price is w less than its actual and another is w more, those two will cancel in MTE, where they won't in MSE.	
164000	=SUM(part1!C6:G6)+10*part1!B6	=(B12-part1!G6)/part1!G6			
205000	=SUM(part1!C7:G7)+10*part1!B7	=(B13-part1!G7)/part1!G7			
340000	=SUM(part1!C8:G8)+10*part1!B8	=(B14-part1!G8)/part1!G8			
260000	=SUM(part1!C9:G9)+10*part1!B9	=(B15-part1!G9)/part1!G9			
227875	=SUM(part1!C10:G10)+10*part1!B10	=(B16-part1!G10)/part1!G10			
438780	=SUM(part1!C11:G11)+10*part1!B11	=(B17-part1!G11)/part1!G11			

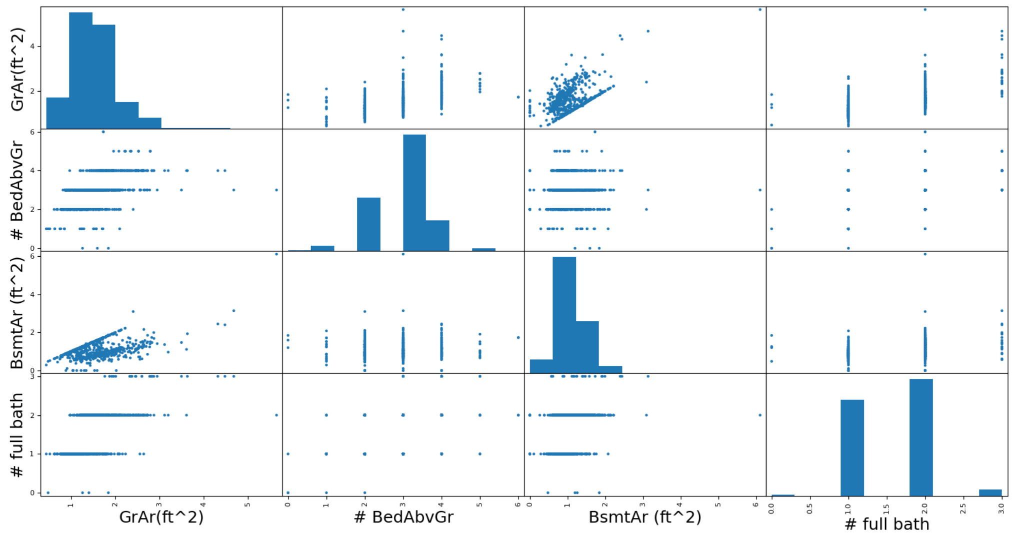
4. In the gradient descent algorithm, alpha is the learning rate of the system chosen by the user. A value of alpha too high can lead to hopping from one extreme to the other, while a value of alpha too small will never converge to a low MSE. A good alpha to choose is one in between the high and low values. Normally, that value is found through trial and error.
5. If the model perfectly fits the house price the model may not work for real world applications as well. The goal of the model is to have it generally work for real-world testing, not work solely with training models. Overfitting like that can cause lower than optimal results during real world application testing.
6. To fix overfitting, dimensionality reduction could be of assistance. It states that, “in some cases, features might be highly correlated, and therefore redundant.” You can improve generalization by taking out features that may not actually have an impact on the result.

Patrick Walsh
ECE241 Project III
Part I: Written Portion and Part II: All Necessary Components

```
-------- question 2 --------
Length of training set: 818
Mean price for house: 1.8618897432762842
Max price of house in training set: 7.55
Min price of house in training set: 0.35311
Standard deviation of training set price: 0.8237942215026949
```



Patrick Walsh
ECE241 Project III
Part I: Written Portion and Part II: All Necessary Components
4.



To accelerate the machine learning process while not sacrificing accuracy, some features can be removed or added together so they no longer have to be individually taken into account. For example, basement area and garage area both have a clear relationship as there's a limiting line limiting basement area from going above some linear line to the garage area. Removing basement area from the equation likely wouldn't have a large impact on accuracy because garage area seems to have some sort of relationship with it.

10.

```
MSE after 495 epochs: nan
MSE after 496 epochs: nan
MSE after 497 epochs: nan
MSE after 498 epochs: nan
MSE after 499 epochs: nan
MSE after 500 epochs: nan
```

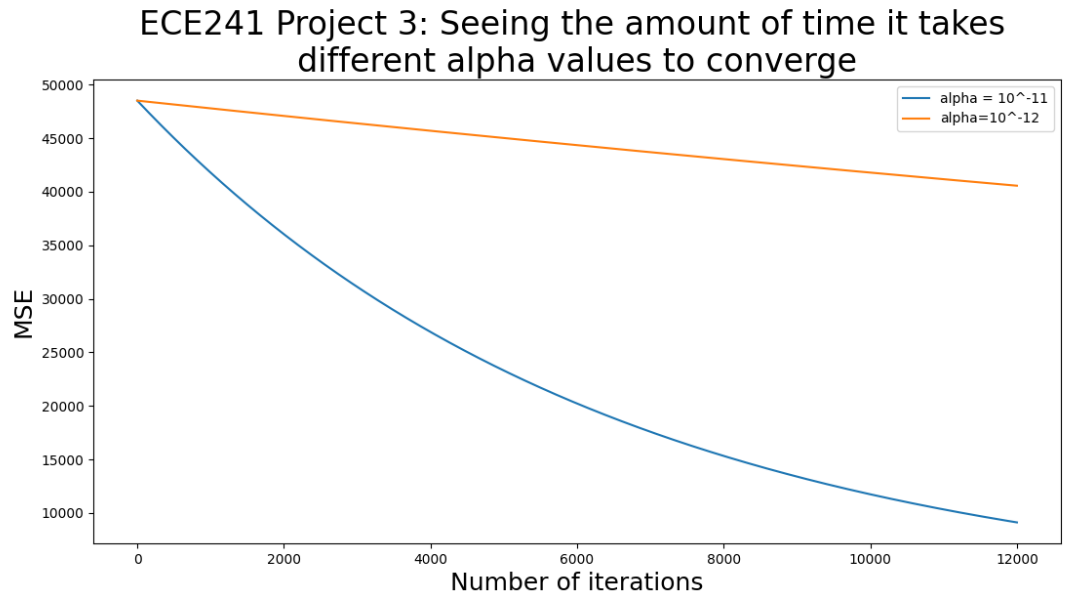
No, the number does not converge to anything meaningful. It diverges to infinity almost instantly.

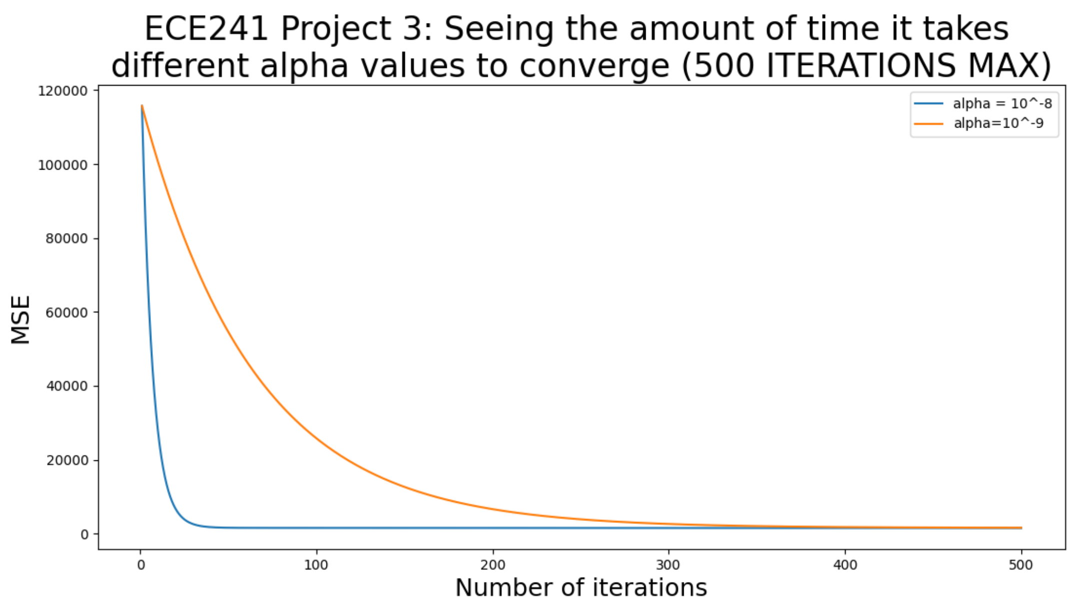
Patrick Walsh

ECE241 Project III

Part I: Written Portion and Part II: All Necessary Components

11.





12. The larger of the two, 10^-11 converges faster than 10^-12. I would assume this is because the alpha values act almost like a bell curve: values larger than a certain value (I believe to be between 10^-6 and 10^-7) diverge to infinity and everything after that point diverges. However, the closer you are to that value, the faster your system converges. Therefore, 10^-11 will converge faster than 10^-12, just as 10^-8 will converge faster than 10^-9.

Patrick Walsh
ECE241 Project III
Part I: Written Portion and Part II: All Necessary Components

13.

```
-------- question 11 --------
MSE (alpha = 10^-8): 1056.1756785434716

-------- question 13 --------
MSE of test data (alpha = 10^-8): 722.8586913367459
```

Surprisingly, I was getting a consistently lower MSE on my testing data compared to the training data. Normally, the opposite would be the case: as the model was optimized on the training data, it would perform better on the training data. However, that wasn't the case today. I ran the MSE multiple times and every time, my final MSE from the training data was higher than my first and only MSE of the test data.