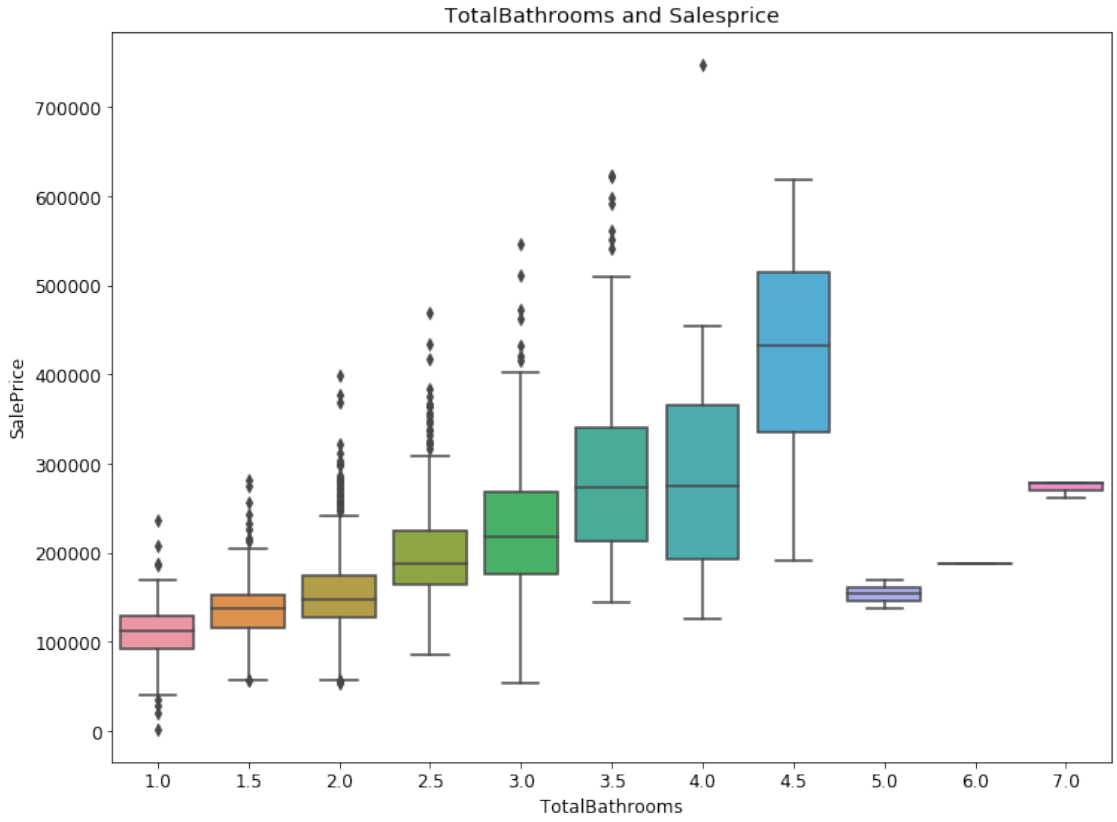
0.1 Question 2b

Create a visualization that clearly and succintly shows that `TotalBathrooms` is associated with `SalePrice`. Your visualization should avoid overplotting.

```
In [15]: sns.boxplot(x='TotalBathrooms', y='SalePrice', data=training_data_with_bathrooms)
         plt.title('TotalBathrooms and Salesprice')
```

```
Out[15]: Text(0.5, 1.0, 'TotalBathrooms and Salesprice')
```



0.2 Question 5d

What changes could you make to your linear model to improve its accuracy and lower the validation error? Suggest at least two things you could try in the cell below, and carefully explain how each change could potentially improve your model's accuracy.

In order to improve the linear model I would add more features to the residuals. This would in turn help the validation error by lowering it.

0.3 Question 6a

Based on the plot above, what can be said about the relationship between the houses' sale prices and their neighborhoods?

house's sale prices and the count of neighborhoods do not have a directly proprtional relationship. StoneBr has the highest sale price average but has 28 neighborhoods. the data is also not evenly distributed among neighborhoods. for example greens only has 6 samples and NAmes has 299.

0.4 Question 8a

Although the fireplace quality variable that we explored in Question 2 has six categories, only five of these categories' indicator variables are included in our model. Is this a mistake, or is it done intentionally? Why?

This was done intentionally. The first variable is removed by the first drop method of data.drop(columns=fireplace_qu_cats[0]). It is a dependent variable, since it is 1 when the rest of the other variables are all equal to 0 and 0 when everything is 0. Dropping the indicator variable allows us to ensure full rank.