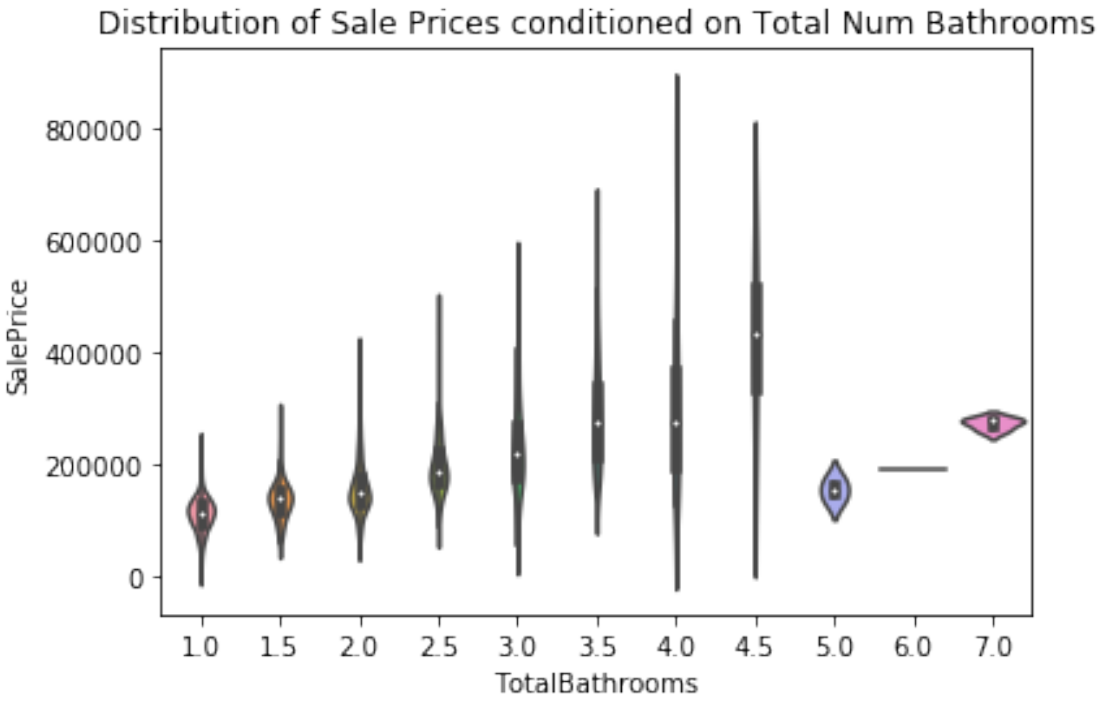
0.1 Question 2b

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

```
In [16]: (sns.violinplot(x="TotalBathrooms", y="SalePrice", data=training_data)
             .set(title="Distribution of Sale Prices conditioned on Total Num Bathrooms"))
```

```
Out[16]: [Text(0.5, 1.0, 'Distribution of Sale Prices conditioned on Total Num Bathrooms')]
```



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.

To decrease the validation error, we can try to improve the generalisability of the model by decreasing model variance ie. removing features so the model doesn't overfit. We could also adjust the train/test split so more data is allocated to the training process such that the data the model is trained on is more robust against outliers.

0.3 Question 6a

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

There is obviously some form of relationship between the neighbourhoods and the sale prices distribution within each neighbourhood. We can actually do an ANOVA (assuming equal variance across neighbourhood strata) to test whether the distributions are in fact different:

```
In [35]: from scipy.stats import f_oneway
         stratified = (
             [training_data[training_data["Neighborhood"] == s]["SalePrice"]
              .astype(float).to_numpy() for s in training_data["Neighborhood"].unique()]
         )
         stat, pvalue = f_oneway(*stratified)
         print(stat)
         print(pvalue)
```

```
99.11526011760812
0.0
```

With such a small pvalue, clearly the distributions are different because the F-statistic rejects the H_0 that the neighbourhoods share a population mean.

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 is intentional. If you augment the design matrix with the sixth category, the design matrix becomes overdetermined and a linearly *dependent* column results; this is because since we have six total categories, the sixth category is determined by the other five.