Question 2a Recall the optimal value of θ should minimize our loss function. One way we've approached solving for θ is by taking the derivative of our loss function with respect to θ, like we did in HW5.

In the space below, use LaTeX to write/compute the following values: * $R(\mathbf{x}, \mathbf{y}, \theta_1, \theta_2)$: our loss function, the empirical risk/mean squared error * $\frac{\partial R}{\partial \theta_1}$: the partial derivative of R with respect to θ_1 * $\frac{\partial R}{\partial \theta_2}$: the partial derivative of R with respect to θ_2

Recall that $R(\mathbf{x}, \mathbf{y}, \theta_1, \theta_2) = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{y}_i - \hat{\mathbf{y}}_i)^2$

$R(x, y, \theta_1, \theta_2) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \theta_1 x - \sin\left(\theta_2 x\right)\right)^2$

$\frac{\partial R}{\partial \theta_1} = -\frac{2}{n}\sum_{i=1}^{n} x\left(y_i - \theta_1 x - \sin\left(\theta_2 x\right)\right)$

$\frac{\partial R}{\partial \theta_2} = -\frac{2}{n}\sum_{i=1}^{n}\left(y_i - \theta_1 x - \sin\left(\theta_2 x\right)\right)\left(x\cos\left(\theta_2 x\right)\right)$

In 1-2 sentences, describe what you notice about the path that theta takes with a static learning rate vs. a decaying learning rate. In your answer, refer to either pair of plots above (the 3d plot or the contour plot).

Looking at the contour plot, the static learning rate seems to jump about between sides of the loss surface, creating a zigzag pattern that would be prone to overshooting. The decaying learning rate converges to a minimum much quicker (although the minimum has a higher MSE than that of the static rate).

0.0.1 Question 4b

Is this model reasonable? Why or why not?

It is not reasonable. If a team scores zero points, it is not reasonable to expect that said team would have a 50% chance of winning.

0.0.2 Question 4c

Try playing around with other theta values. You should observe that the models are all pretty bad, no matter what θ you pick. Explain why below.

We've failed to add an intercept term to t in order to account for translation in the dataset. Suppose the probability "cusp" where losses and wins have equal probability is not at zero. Then the lack of an intercept term would grossly misrepresent the data since the model *assumes* that the cusp is at zero.

0.0.3 Question 5b

Using the plot above, try adjusting θ_2 (only). Describe how changing θ_2 affects the prediction curve. Provide your description in the cell below.

Changing θ_2 adjusts the point where the logistic curve is 0.5. That is, θ_2 is the location where `class-0` and `class-1` (ie. losses and wins) have equal probability.

0.0.4 Question 7c

Look at the coefficients in `theta_19_hat` and identify which of the parameters have the biggest effect on the prediction. For this, you might find useful_numeric_fields.columns useful. Which attributes have the biggest positive effect on a team's success? The biggest negative effects? Do the results surprise you?

It seems that the parameter `FG_PCT` has a very large negative effect on the prediction by an order of magnitude more than the next most negative prediction. Conversely, the most positive parameter is `FG3_PCT`, which is still an order of magnitude less than that of `FG_PCT`. The difference in magnitudes is not that suprising considering that the parameters may be normalised to different maximimums and minimums. As someone not *that* into basketball, the other parameters are somewhat meaningless to me, and as such I am not in a position to pass judgement on the results. However, it would be prudent to show these coefficients to someone more well-versed in matters of basketball as a form of sanity check to make sure that the model is not overfitting.

```
In [143]: list(zip(theta_19_hat, useful_numeric_fields.columns))
```

```
Out[143]: [(2.124692759922619, 'FGM'),
           (-0.451172059747732, 'FGA'),
           (-21.90165000882123, 'FG_PCT'),
           (0.923445802180814, 'FG3M'),
           (-0.003558515378822562, 'FG3A'),
           (2.724898268191739, 'FG3_PCT'),
           (0.8830998065270763, 'FTM'),
           (-0.07165227422286279, 'FTA'),
           (2.1079984185261127, 'FT_PCT'),
           (0.30519224478702595, 'OREB'),
           (0.32588261293925896, 'DREB'),
           (0.048336404992882574, 'REB'),
           (0.0195557693456402, 'AST'),
           (0.38725654011094396, 'STL'),
           (0.06964299113447128, 'BLK'),
           (-0.3111864224106059, 'TOV'),
           (-0.05578939650320456, 'PF'),
           (-0.7549816715563751, 'PTS'),
           (5.193195623381068, 'BIAS')]
```

To double-check your work, the cell below will rerun all of the autograder tests.

```
In [ ]: grader.check_all()
```

0.1 Submission

Make sure you have run all cells in your notebook in order before running the cell below, so that all images/graphs appear in the output. The cell below will generate a zipfile for you to submit. **Please save before exporting!**

In [ ]:
```
# Save your notebook first, then run this cell to export your submission.
grader.export()
```