



## PROJECT

## Titanic Survival Exploration

A part of the Machine Learning Engineer Nanodegree Program

## PROJECT REVIEW

## NOTES

SHARE YOUR ACCOMPLISHMENT!  

## Meets Specifications

Keep up the momentum!

## Answers to Each Question

The `predictions_0` function has been run and the accuracy of the predictions is reported.

The `predictions_1` function has been correctly implemented. The expected accuracy of the predictions is reported.

Good first step towards building an accurate decision tree. Note how much information and performance we gain at this early step. This is what we want for the initial split of a decision tree.

The `predictions_2` function has been correctly implemented. The expected accuracy of the predictions is reported.

The `predictions_3` function has been correctly implemented and obtains a prediction accuracy of at least 80%. The approach to the task has been documented, including features that were explored and intermediate steps taken to complete the function.

Excellent work building and documenting this final version of the predictor.

The next step in development is to consider the underlying mechanisms that drive survival of one group over another. What was going on in this scenario (Titanic) that may or may not apply to another disaster scenario. Considering these elements can help adjust algorithms to generalize well to new data.

Notice how our early splits yield a lot of information and performance gain but this diminishes as we descend the tree. This is because very few members of the population are ending up in each leaf, at this point, and this is reflected in the performance gains. This is how overfitting occurs.

Given this, be careful about "improving" a predictor. We must not only concern ourselves with metrics on the training set but also how well the predictor will generalize to new data.

A valid scenario where supervised learning can be applied is reported. A clear outcome variable and at least two potential predictor variables are identified as part of the description.

This is a great scenario! I'm guessing a highly predictive feature would be `vo2max`.

[↓ DOWNLOAD PROJECT](#)[RETURN TO PATH](#)

[Student FAQ](#)