	Mastery 20 points	Comments / Activities to Review
Data Preprocessing	The submission does all of the following: ✓ Data is correctly split into a training and test set. ✓ Categorical features are converted to numeric with get_dummies ✓ Missing categorical features in the testing data are filled in programmatically ✓ Data is scaled with `StandardScaler`.	19.2.4 on preprocessing data and get_dummies() and StandardScaler(). Make sure you avoid bias (think about what data you are applying the preprocessing to, train vs test). Make sure to loop over the dataset columns to fill in missing features (columns/variables) present in train data but not test data
Reflection and Reporting	The submission does all of the following: ✓ Makes a prediction on which model will perform better on the *unscaled* data. The prediction is made with adequate justification. (Note: no points are removed for predictions that prove to be incorrect). ✓ Makes a prediction on which model will perform better on the *scaled* data. The prediction is made with adequate justification. (Note: no points are removed for predictions that prove to be incorrect). ✓ Makes a comparison between predicted behavior of the models on unscaled data and the actual results. ✓ Makes a comparison between predicted behavior of the models on scaled data and the actual results.	For insights on the importance of scaling, please read: https://scikit-learn.org/stable/auto_examples/preprocessing/plot_scaling_importance.html Optional: https://towardsdatascience.com/scale-standardize-or-normalize-with-scikit-learn-6ccc7d176a02 Do you think random forest classifiers require scaling? Google it! And cite your sources in the report! Make sure you make note of what you expect before you run the model (you must make an *educated* guess), and whether the output concurs and not, and **why**
Model Creation	The submission does all of the following:	Logistic regression activities: 19.1.5-6, 19.2.1 Random forest activities: 19.3.1-5 (no need to do feature selection, just define model, train and test)

	✓ Creates, trains, and scores a `LogisticRegression` model on unscaled data ✓ Creates, trains, and scores a `LogisticRegression` model on scaled data ✓ Creates, trains, and scores a `RandomForestClassifier` model on unscaled data ✓ Creates, trains, and scores a `RandomForestRegression` model on scaled data	