Requires Changes

**3 SPECIFICATIONS REQUIRE CHANGES**

Very impressive submission here!  
It seems that you have acquired great knowledge of supervised learning algorithms and are now ready to apply it on your own problems.  
However, a few very minor changes are required to meet our rubric.  
Keep doing this great job! Keep experimenting and be curious.  
All the best for your future.  
Happy Learning!!

**Exploring the Data**

**Student's implementation correctly calculates the following:**

* **Number of records**
* **Number of individuals with income >$50,000**
* **Number of individuals with income <=$50,000**
* **Percentage of individuals with income > $50,000**

Nice job finding out the correct numbers!

I would encourage you to do more exploratory data analysis (EDA) on this dataset using pandas and other libraries like seaborn. You may refer to the following resources to know more about how EDA is done using these libraries. [link1](https://medium.com/open-machine-learning-course/open-machine-learning-course-topic-1-exploratory-data-analysis-with-pandas-de57880f1a68) [link2](https://towardsdatascience.com/visualize-world-trends-using-seaborn-in-python-2e563e7d35da)

**Preparing the Data**

**Student correctly implements one-hot encoding for the feature and income data.**

Nice job encoding the features using get\_dummies! Nice job converting the target labels to correct numerical values!!

Please note that there are different encoding strategies suitable for different tasks. I would encourage you to check out the following resources to know more about encoding strategies. [link1](https://www.kdnuggets.com/2015/12/beyond-one-hot-exploration-categorical-variables.html) [link2](http://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelBinarizer.html)

**Evaluating Model Performance**

**Student correctly calculates the benchmark score of the naive predictor for both accuracy and F1 scores.**

The accuracy and fscore both are incorrect.. Please check the formulas for FP.

Also, please note that, this dataset is not balanced, i.e., the number of samples of positive class (income > 50k) is very less compared to the number of samples of negative class (income <= 50k). In such cases, accuracy is not the correct metric to evaluate the performance of any model. F1 or F-beta scores are more useful. You may check out the following resource to know more about metrics. [link1](https://medium.com/greyatom/performance-metrics-for-classification-problems-in-machine-learning-part-i-b085d432082b)

**The pros and cons or application for each model is provided with reasonable justification why each model was chosen to be explored.**

**Please list all the references you use while listing out your pros and cons.**

Very impressive discussion on the real-world applications, strengths and weaknesses of the models you chose! It seems that you have a very good understanding of the models.

You also have a good understanding on model selection, i.e., which model to choose for a particular problem at hand. This is a 'must-have' skill for any machine learning practitioner. I would further encourage you to take a look at the following resources which will help you enhancing this skill even further. [link1](https://docs.microsoft.com/en-us/azure/machine-learning/studio/algorithm-choice) [link2](http://scikit-learn.org/stable/tutorial/machine_learning_map/index.html) [link3](https://blogs.sas.com/content/subconsciousmusings/2017/04/12/machine-learning-algorithm-use/)

**Student successfully implements a pipeline in code that will train and predict on the supervised learning algorithm given.**

Great job implementing the training and prediction pipeline!

Please note that, it is always good to write a function like train\_predict which can be called from other modules with different samples of train and test data. It makes the code cleaner, organised and easy to debug.

**Student correctly implements three supervised learning models and produces a performance visualization.**

Great job training the models with different sample sizes and getting the predictions on test set!

However, you missed one rubric requirement - you need to use random\_state while initializing decision tree model. You already did it for SVM and Logistic Regression, you need to do the same for decision tree as well. Random seeds help in debugging and reproducing the same result at a later time. Please check out the following resource to know more about this. [link1](https://machinelearningmastery.com/randomness-in-machine-learning/). This is just a rubric requirement which I am bound to follow.

**Improving Results**

**Justification is provided for which model appears to be the best to use given computational cost, model performance, and the characteristics of the data.**

Very very impressive justification on choosing Logistic Regression.

**Student is able to clearly and concisely describe how the optimal model works in layman's terms to someone who is not familiar with machine learning nor has a technical background.**

Very good discussion on Logistic Regression. I really liked the way you described how the model gets trained and makes prediction. It seems that you have a very good understanding of the model. Great job!

**The final model chosen is correctly tuned using grid search with at least one parameter using at least three settings. If the model does not need any parameter tuning it is explicitly stated with reasonable justification.**

Great job! Very nice implementation of GridSearch and the choice of hyperparameters and their values were great.

**Student reports the accuracy and F1 score of the optimized, unoptimized, models correctly in the table provided. Student compares the final model results to previous results obtained.**

Good work comparing your tuned model, the untuned model and the Naive one! Please rewrite this section once you make changes to the naive classifier scores section.

**Feature Importance**

**Student ranks five features which they believe to be the most relevant for predicting an individual's’ income. Discussion is provided for why these features were chosen.**

Nice intuition, these are some great features to check out!

**Student correctly implements a supervised learning model that makes use of the feature\_importances\_ attribute. Additionally, student discusses the differences or similarities between the features they considered relevant and the reported relevant features.**

Great job implementing Random Forest classifier to get feature importance!

Please note that, different algorithms use different strategies to get feature importance. So, the result could be different if you choose a different model. I would encourage you to try AdaBoost or Gradient Boosting to get feature importance and check if your intuition match with their results.

**Student analyzes the final model's performance when only the top 5 features are used and compares this performance to the optimized model from Question 5.**

Nice analysis of the final model's performance with only the top 5 features.