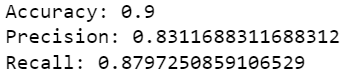
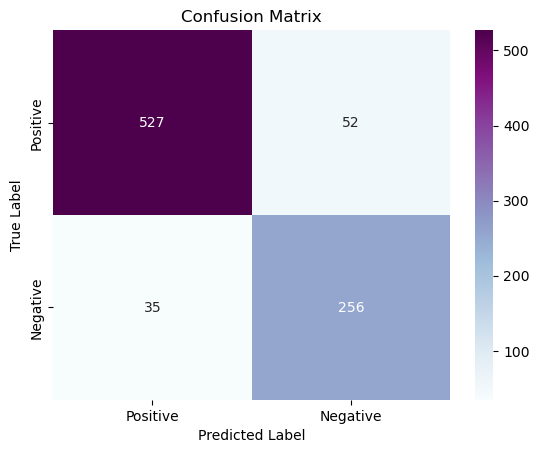
### **Checkpoint 2 Report - CPSC 6300 Hubig**

### **Model Choice & Justification**

The response variable we seek to predict with our model is whether or not a task that is accepted by a crowd sourced worker will be completed. This response variable can only have one of two values: 1 if the task is completed, 0 if the task is not completed. As a result, the model chosen to predict task completion is a k-Nearest Neighbors classification model. This model was chosen over a linear regression classification model because, in our EDA, we found that much of our data was tightly clustered around lower values, and was not easily linear separable. The initial model has a k-value of k = 3 to avoid any erroneous classifications that fall outside the tight cluster groups in the plot of key predictors.

### **Model Analysis - Summary**

For the binary classification model, a confusion matrix is an appropriate visualization to demonstrate how the model performs on test data. Figure 1 demonstrates how the model performed on a test subset of the data using a matrix, where correctly classified predictions are in the top left and bottom right squares, and misclassified predictions are in the other squares. A confusion matrix may be supplemented with classification metrics to understand the rate of model error: accuracy, the percentage of all correct classifications, precision, the percentage of correct positive classifications, and recall, the percentage of positive classifications identified by the model. Figure 2 shows the accuracy, precision, and recall of the model, where the closer the value is to one, the less error the model had in predicting task completion. This means of error rate evaluation serves a binary classifier the best because it puts the rate of successful distinction between two choices into perspective, with respect to different measures of success. For example, if our goal of error analysis is to simply evaluate the rate of any successful classifications, accuracy serves best, but if we are most concerned with correctly identifying positive cases, we should look at recall.



***Figure 1 Figure 2***

Based on the classification metrics obtained with testing data, 90% of the time our kNN model appears to correctly identify whether or not a task will be completed. Relatively speaking, this is a pretty good chance of identification, given the confusing, clustered nature of the data shown in the EDA. The lowest classification metric is precision, which means that the model has the highest rate of error when it makes a positive classification prediction. Nonetheless, this is still greater than 80%, which is still significant, given the density of the plots of the dataset's features. Overall, for an initial model, it does a fairly adequate job of fitting the data and correctly classifying unseen data.

### **Model Analysis - 3 Cases of Interest**

With cases of interest, we were keen to see how our models acted with both completed and non-completed data. We picked 3 cases of interest based on our 3 feature variables to see how our model reacts when we have 3 different data entries with a higher monetary value, a larger hit count, and a larger event count. They were the following entries.

*For figures 3-5 column entries (left to right) are: platform.name, user.id, task.id, task.monetary\_reward\_in\_dollars, task.assignable\_hits\_count, task.requester\_name, task.estimate\_duration\_in\_seconds, total.labor.event\_count, total.labor.duration\_in\_seconds, working.labor.event\_count, working.labor.duration\_in\_seconds, invisible.labor.event\_count, invisible.labor.duration\_in\_seconds, adjusted.monetary\_reward\_in\_dollars, completed\_task*

Case one (Figure 3) had the input feature of a higher monetary reward value ($0.50) compared to our other cases we examined. This case also examined the other two input features of 0 assignable HIT’s along with a total labor event count occurrence of 53. With these features, the person did not complete a task along these parameters.



***Figure 3 - Case one in dataset***

Case two (Figure 4) had the input feature of an extremely high amount of assignable HIT’s (3704) compared to other cases we examined. This case also examined the other two input features of $0.15 monetary reward along with a total labor event count occurrence of 99. With these features, the person did not complete a task along these parameters.



***Figure 4 - Case two in dataset***

Case three (Figure 5) had the input feature of an extremely high amount of total labor events with it having 581 total labor events. This was exceptionally high compared to other cases we examined. This case also examined the other two input features of $0.04 monetary reward along with a assignable hit count of 2. With these features, the person did complete a task along these parameters.



***Figure 5 - Case three in dataset***

Our trained model was able to successfully predict the outputs of these unseen data points when entered and scaled into our model manually. You can see our program's output in Figure 6.



***Figure 6 - Model prediction on three cases when inputted***