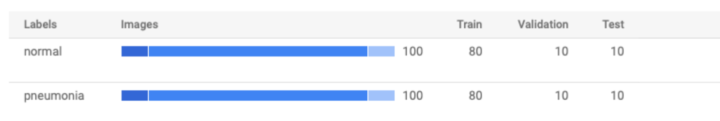
AutoML Modeling Report

Prathamesh Morde

Binary Classifier with Clean/Balanced Data

Train/Test Split
How much data was used for training? How much data was used for testing?

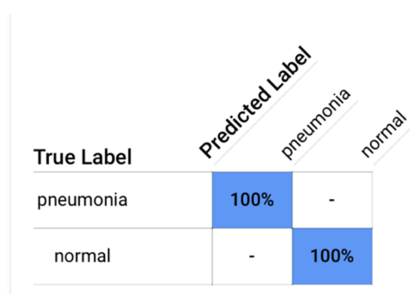
For training I have used 100 images from the normal class and 100 images from the pneumonia class. The ML engine used 20 images for testing (20%) and remaining data is used for training (160) and validation (20).



Labels	Images	Train	Validation	Test
normal	100	80	10	10
pneumonia	100	80	10	10

I also deployed the model and try to test the model on new data i.e. untrained data (10 images) that the model has never seen.

Confusion Matrix
What do each of the cells in the confusion matrix describe? What values did you observe (include a screenshot)? What is the true positive rate for the “pneumonia” class? What is the false positive rate for the “normal” class?



True Label \ Predicted Label	pneumonia	normal
pneumonia	100%	-
normal	-	100%

As per the confusion matrix above, the top left-hand corner contains True Positives, which means these are the patients that have pneumonia that were correctly identified by the algorithm.
The bottom right-hand corner contains True Negatives,

<table>
<tr>
<td></td>
<td>
which means the patients that did not have pneumonia that were correctly identified by the algorithm.

100% rate for True Positive and 100% rate for True Negative.

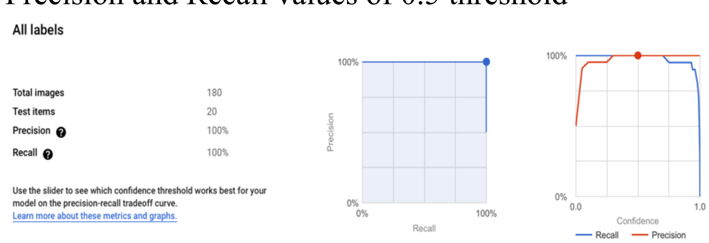
False positive rate for the normal class -> The normal case was positively predicted and its false. The model able to predict that patient is normal therefore the TN of the model is 100%.

The evaluation as shown in the screenshot did not show the false positive rate because of the following reason

1. We have provided the balanced data and due to which it resulted into high precision. The high precision model results into fewer false positives.

When I used the new data and tested, I found out that for normal image the model marked it as a pneumonia which is False Positive.
</td>
</tr>
<tr>
<td>Precision and Recall
What does precision measure? What does recall measure? What precision and recall did the model achieve (report the values for a score threshold of 0.5)?</td>
<td>
Precision -
It quantifies the number of correct positive predictions made by the model. In our example it tells us out of all pneumonia predications how many we got it right.

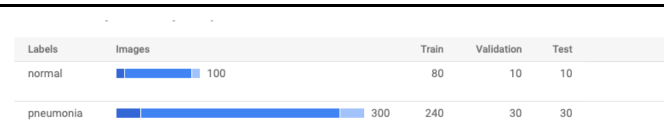
Recall -
It tells us the fraction of relevant or true instances that were retrieved. In our example it tells us out of all pneumonia truth how many we got it right.

Precision and Recall values of 0.5 threshold


</td>
</tr>
<tr>
<td>Score Threshold
When you increase the threshold</td>
<td>As you can see, by increasing the threshold value the Precision increases but Recall decreases and if you</td>
</tr>
</table>

<table>
<tr><td>what happens to precision? What happens to recall? Why?</td><td>decrease the value then Recall increases but Precision decreases. At default threshold value (Zero), Precision is less than 80% and Recall is higher than 80%.
After seeing the above cases we can now say that precision and recall both are in are in a contradictory relationship with each other. In some situations, we might want to maximize either recall or precision. For example, for our use-case we would probably want a recall near 1.0 because we want to find the patients who have pneumonia.</td></tr>
</table>

Binary Classifier with Clean/Unbalanced Data

<table>
<tr><td>Train/Test Split
How much data was used for training? How much data was used for testing?</td><td>
<table><tr><th>Labels</th><th>Images</th><th>Train</th><th>Validation</th><th>Test</th></tr><tr><td>normal</td><td>100</td><td>80</td><td>10</td><td>10</td></tr><tr><td>pneumonia</td><td>300</td><td>240</td><td>30</td><td>30</td></tr></table>
320 data images used for training.
40 data images used for validation.
40 data images used for testing.

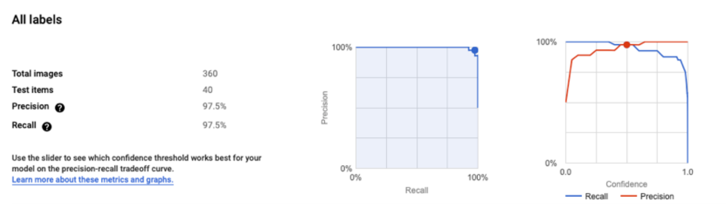
I have also used untrained data (10 images) for further validation.</td></tr>
<tr><td>Confusion Matrix
How has the confusion matrix been affected by the unbalanced data? Include a screenshot of the new confusion matrix.</td><td>

As per the confusion matrix above, the top left-hand corner contains True Positives, which means these are the patients that have pneumonia that were correctly identified by the algorithm.
The bottom right-hand corner contains True Negatives, which means the patients that did not have pneumonia that were correctly identified by the algorithm.</td></tr>
</table>

<table>
<tr>
<td></td>
<td>100% rate for True Positive - Patients that have pneumonia that were correctly identified by the algorithm.

90% rate for True negative - Patients that did not have pneumonia that were correctly identifies by the algorithm.

10% rate for False negative – Patients that did not have pneumonia, but the model still marked it as pneumonia.

Due to unbalanced data the model predicted higher FN (10%) as compared to the balanced dataset.</td>
</tr>
<tr>
<td>Precision and Recall
How have the model’s precision and recall been affected by the unbalanced data (report the values for a score threshold of 0.5)?</td>
<td>Precision and Recall model for 0.5 threshold


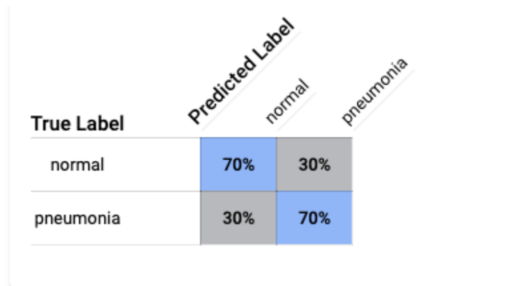
Due to unbalanced nature of the dataset we saw the slight drop-in precision and recall. When you have a low recall + low precision usually the class is poorly handled by the model.</td>
</tr>
<tr>
<td>Unbalanced Classes
From what you have observed, how do unbalanced classed affect a machine learning model?</td>
<td>The unbalanced data usually refers to the classification problem. Most of the machine learning classification models are sensitive to the unbalanced datasets. For our example, assume we had 100 normal vs 300 pneumonia samples and it has been trained and tested on such a dataset could now predict “Pneumonia” for all samples with high accuracy. In conclusion the unbalanced data will lean towards more common class.

When I used the untrained data (10 normal images) to test the model I found out that the model classified 7 images as pneumonia.</td>
</tr>
</table>

Binary Classifier with Dirty/Balanced Data

<table>
<tr>
<td>Confusion Matrix
How has the confusion matrix been affected by the dirty data? Include a screenshot of the new confusion matrix.</td>
<td>For this example, I used two samples

70% pneumonia and 30% normal
70% normal and 30% pneumonia.



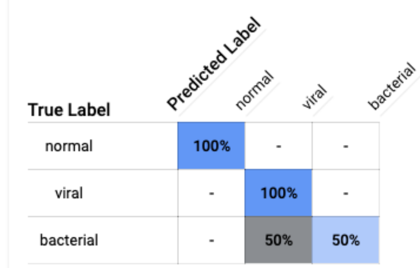
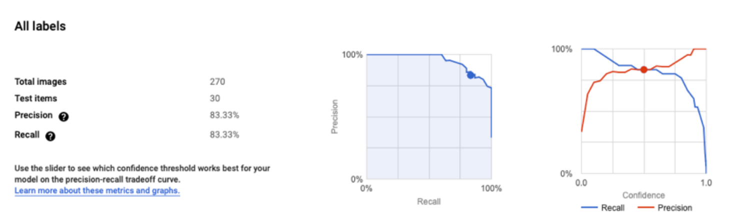
As you can see from the confusion matrix above with 30% data mislabelled shows than the diagonal boxes (TP and TN) are higher than FP and FN boxes. This represents that desired categories identified correctly to the extent. The normal labels are predicted 70% with 30% of ambiguity and pneumonia labels are predicted with 70% and 30% of misclassification.</td>
</tr>
<tr>
<td>Precision and Recall
How have the model's precision and recall been affected by the dirty data (report the values for a score threshold of 0.5)? Of the binary classifiers, which has the highest precision? Which has the highest recall?</td>
<td>Model's precision and recall for threshold of 0.5


Even with 30% of mislabelled data the mislabelled images share similar visual patterns as correctly labelled images and the model might be biased towards one class of label/image over the other.

Of the binary classifiers, the binary classifier with Clean/Balanced data has the highest precision – 100 % and highest recall – 100%</td>
</tr>
<tr>
<td>Dirty Data
From what you have observed,</td>
<td>The dirty data impacts the machine learnings results but also has implications on the performance. In our example,</td>
</tr>
</table>

<table>
<tr><td>how does dirty data affect a machine learning model?</td><td>the dirty data did not reflect the complete classification of the labels as the dirty data causes some biasness in the model.</td></tr>
</table>

3-Class Model

<table>
<tr><td>Confusion Matrix
Summarize the 3-class confusion matrix. Which classes is the model most likely to confuse? Which class(es) is the model most likely to get right? Why might you do to try to remedy the model's "confusion"? Include a screenshot of the new confusion matrix.</td><td>For a balanced dataset of 100 in each class (normal, viral pneumonia, bacterial pneumonia) the confusion matrix shows that the model able to classify normal and viral pneumonia labels 100%. The bacterial pneumonia was predicated 50% and 50% confused for the viral pneumonia.

<table>
<tr><th>True Label</th><th>Predicted Label: normal</th><th>viral</th><th>bacterial</th></tr>
<tr><td>normal</td><td>100%</td><td>-</td><td>-</td></tr>
<tr><td>viral</td><td>-</td><td>100%</td><td>-</td></tr>
<tr><td>bacterial</td><td>-</td><td>50%</td><td>50%</td></tr>
</table>
High confusion usually leads to incorrect predictions. The way to remedy this confusion is to provide more additional training data.</td></tr>
<tr><td>Precision and Recall
What are the model's precision and recall? How are these values calculated (report the values for a score threshold of 0.5)?</td><td>

The model's precision and recall at 0.5 threshold is 83.33%

The formula for precision is True Positive / (True Positive + False Positive). In this example since we have three classes the calculation is done for each class so precision for viral, bacterial and normal. The final value is basically an average of all the precision.

Precision = (P normal + P bacterial + P viral) / 3 (No. of classes)</td></tr>
</table>

	The formula for recall is True Positive/ (True Positive + False Negative) Recall = (R normal + R bacterial + R viral) / 3 (No. of classes)
F1 Score What is this model's F1 score?	F1 score for overall performance of the measure model is F1 = 2 * (precision* recall)/ (precision + recall) = 2 * (0.833 * 0.833) / (0.833 + 0.833) = 2 (0.69/1.66) = 0.836 or 83.6%