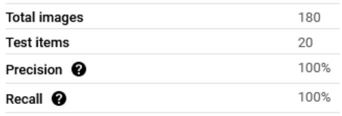
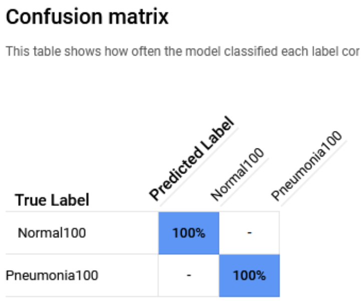
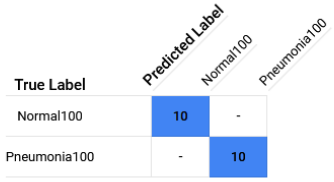
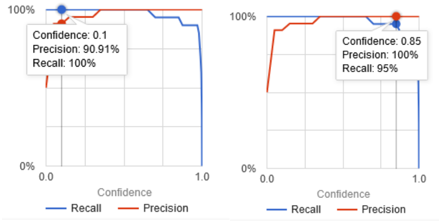
# AutoML Modeling Report

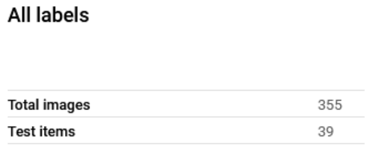
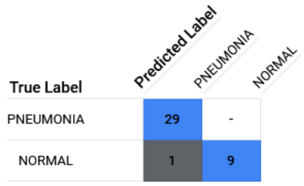
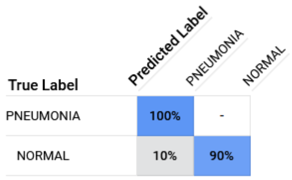
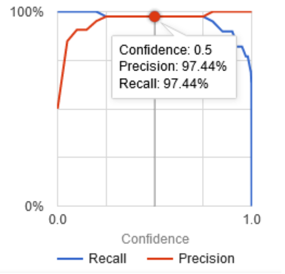
*Olivier POLACK*

## Binary Classifier with Clean/Balanced Data

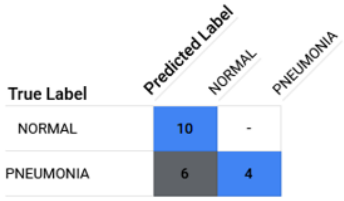
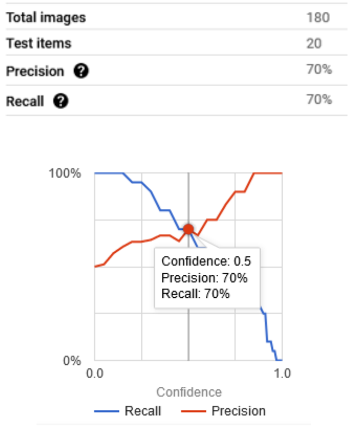
| | |
|---|---|
| **Train/Test Split**<br>How much data was used for training? How much data was used for testing? | For this first model, 180 images were used training the model and the remaining 20 others were used for testing the model.<br><br>Total images 180<br>Test items 20<br>Precision ❓ 100%<br>Recall ❓ 100% |
| **Confusion Matrix**<br>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? | Figures in the confusion matrix describe the results of the tests performed on the model after it has been trained. The number of correctly predicted or classified items for each possible class is shown as well as the number of incorrect predictions for each class.<br><br>Confusion matrix<br>This table shows how often the model classified each label cor<br>Predicted Label: Normal100, Pneumonia100<br>True Label Normal100: 100%, -<br>True Label Pneumonia100: -, 100%<br><br>Predicted Label: Normal100, Pneumonia100<br>True Label Normal100: 10, -<br>True Label Pneumonia100: -, 10<br>In the first example, we have 100% of actual "normal" class images predicted as such, so 0% (or here "–" is displayed) of "normal" class images. For the "pneumonia" class, we also have 100% of actual "pneumonia" class detected as such, and no "pneumonia" class label classified as "normal".<br>Assuming "Pneumonia" is being considered as the positive class, the true positive rate is 100%.<br>The false positive rate for the "normal" class is 0% as no "normal" cases are predicted as "pneumonia" cases. |

| | |
|---|---|
| **Precision and Recall**<br>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)? | Precision is the ratio of true positives to the total of the true positives and false positives. It can be expressed in %. If the number of False Positives is 0, the model is as precise or accurate as it could be, thus Precision ratio equals 1 or 100%. The less accurate the model is, the more false positives are predicted, the more the Precision ratio decreases.<br>On the opposite, instead of focusing on the number of False Positives the model predicted, Recall looks at the number of False Negatives**.** Recall is the ratio of True Positives to the actual Positives:  True Positives plus False Negatives. Recall can be expressed in % too. So a model with a perfect Recall is a model that does not predict any False Negatives at all, regardless of the number of False Positives predictions. A model with a perfect Recall does not let any actually positive cases miss-classified as Negative. |
| **Score Threshold**<br>When you increase the threshold what happens to precision? What happens to recall? Why? |  <br>There will be a trade-off between Precision and Recall in almost all models unless the model is perfect. Higher precision leads to lower recall as the threshold increases and a higher recall will lead to lower precision as threshold decreases. |

# Binary Classifier with Clean/Unbalanced Data

| | |
|---|---|
| **Train/Test Split**<br>How much data was used for training? How much data was used for testing? | 400 images were imported, 100 normal images and 300 pneumonia images. 394 images were actually used, among which 355 were used for training the model and 39 were used for testing the model.<br>The reason why some images were not used is because these images are already considered as ground truth by Google. I have tried two times and I always had this warning for few images, please let me know in the review what I should do, I found in the FAQ an answer that seems to imply this is acceptable.<br><br>All labels<br>Total images 355<br>Test items 39 |
| **Confusion Matrix**<br>How has the confusion matrix been affected by the unbalanced data? Include a screenshot of the new confusion matrix. | The new confusion matrix shows among tested labels all Pneumonia labels are correctly predicted as such (there are no false positives). However, one Normal image is classified as Pneumonia, ie we have 1 false negative. This means that the much larger amount of Pneumonia labels among the dataset caused the model to be biased towards Pneumonia.<br><br>True Label · Predicted Label: PNEUMONIA · NORMAL<br>PNEUMONIA · 29 · -<br>NORMAL · 1 · 9<br><br>True Label · Predicted Label: PNEUMONIA · NORMAL<br>PNEUMONIA · 100% · -<br>NORMAL · 10% · 90% |
| **Precision and Recall**<br>How have the model's precision and recall been affected by the unbalanced data (report the values for a score threshold of 0.5)? | For a score threshold of 0.5, the model's precision and recall have dropped from 100% down to 97.44%.<br> <br>Confidence: 0.5<br>Precision: 97.44%<br>Recall: 97.44% |
| **Unbalanced Classes**<br>From what you have observed, how do unbalanced classed affect a machine learning model? | Unbalanced classes decrease precision and recall rates as they tend to create a biased model predicting false positives in case more labels of the positive class have been thrown into the training dataset. |

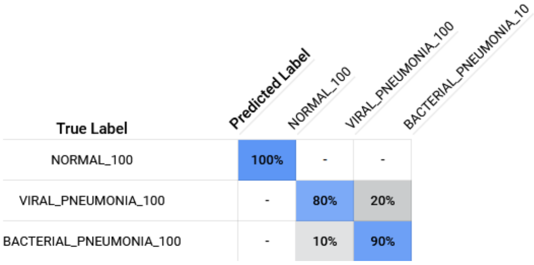
## Binary Classifier with Dirty/Balanced Data

<table>
<tr>
<td><strong>Confusion Matrix</strong><br>How has the confusion matrix been affected by the dirty data? Include a screenshot of the new confusion matrix.</td>
<td>With the dirty data, the confusion matrix shows results are less precise and with poorer recall.<br>

<table>
<tr><th>True Label</th><th>Predicted Label NORMAL</th><th>PNEUMONIA</th></tr>
<tr><td>NORMAL</td><td>10</td><td>-</td></tr>
<tr><td>PNEUMONIA</td><td>6</td><td>4</td></tr>
</table>
</td>
</tr>
<tr>
<td><strong>Precision and Recall</strong><br>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>The precision and recall are both down at 70% with the dirty data.<br>

<table>
<tr><td>Total images</td><td>180</td></tr>
<tr><td>Test items</td><td>20</td></tr>
<tr><td>Precision ?</td><td>70%</td></tr>
<tr><td>Recall ?</td><td>70%</td></tr>
</table>

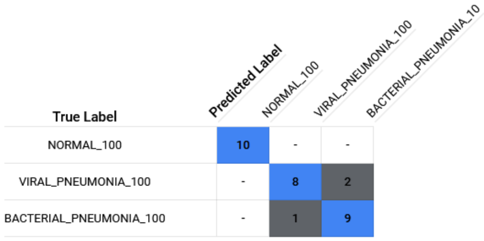
Confidence: 0.5<br>Precision: 70%<br>Recall: 70%<br>
Results benchmark of the binary classifier models
<table>
<tr><th>Model</th><th>Precision</th><th>Recall</th></tr>
<tr><td>1: Clean/ Balanced Data</td><td>100 %</td><td>100 %</td></tr>
<tr><td>2. Clean/Unbalanced Data</td><td>97,4 %</td><td>97,4 %</td></tr>
<tr><td>3. Dirty/Balanced Data</td><td>70 %</td><td>70 %</td></tr>
</table>
The model with balanced data has the highest precision and recall of 100% each at a threshold of 0.5 among the binary classifiers.</td>
</tr>
<tr>
<td><strong>Dirty Data</strong><br>From what you have observed, how does dirty data affect a machine learning model?</td>
<td>Dirty data obviously deteriorates the quality of the trained model, precision, recall and accuracy are affected.</td>
</tr>
</table>

# 3-Class Model

<table>
<tr>
<td><b>Confusion Matrix</b><br>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>The model is most likely the two Pneumonia classes labels and is most likely to get right the images of the Normal class. This is harder for the model to properly distinguish between different Pneumonia types as between Normal and Pneumonia, probably because differences between the two Pneumonia types are much subtle as the difference between Normal and Pneumonia.<br>New confusion matrix in % and items count

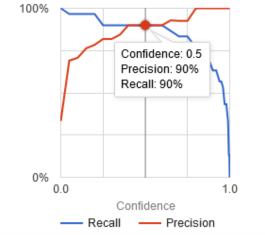


<table>
<tr><th>True Label</th><th>Predicted Label</th><th>NORMAL_100</th><th>VIRAL_PNEUMONIA_100</th><th>BACTERIAL_PNEUMONIA_10</th></tr>
<tr><td>NORMAL_100</td><td></td><td>100%</td><td>-</td><td>-</td></tr>
<tr><td>VIRAL_PNEUMONIA_100</td><td></td><td>-</td><td>80%</td><td>20%</td></tr>
<tr><td>BACTERIAL_PNEUMONIA_100</td><td></td><td>-</td><td>10%</td><td>90%</td></tr>
</table>



<table>
<tr><th>True Label</th><th>Predicted Label</th><th>NORMAL_100</th><th>VIRAL_PNEUMONIA_100</th><th>BACTERIAL_PNEUMONIA_10</th></tr>
<tr><td>NORMAL_100</td><td></td><td>10</td><td>-</td><td>-</td></tr>
<tr><td>VIRAL_PNEUMONIA_100</td><td></td><td>-</td><td>8</td><td>2</td></tr>
<tr><td>BACTERIAL_PNEUMONIA_100</td><td></td><td>-</td><td>1</td><td>9</td></tr>
</table>

In order to remedy the model’s confusion, I would try to increase the number of labels of each of the three classes.</td>
</tr>
<tr>
<td><b>Precision and Recall</b><br>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><b><u>Calculation of Precision and Recall for each classifier:</u></b><br><b>‘Normal’ Class</b><br>True Positives = 10  / False Positives = 0<br>True Negatives = 8+9  = 17 / False Negatives = 2+1 = 3<br>True Positives + False Positives (TP+FP) = 10<br>True Positives + False Negatives (TP+FN) = 13<br>  Precision = TP/(TP+FP) = 10/10 = 100%<br>  Recall = TP/(TP+FN) = 10/13 = 76,9%<br><br><b>‘Viral Pneumonia’ Class</b><br>True Positives = 8  / False Positives = 1<br>True Negatives = 10+9  = 19 / False Negatives =  2<br>True Positives + False Positives (TP+FP) = 9<br>True Positives + False Negatives (TP+FN) = 10<br>  Precision = TP/(TP+FP) = 8/9 = 88,8%<br>  Recall = TP/(TP+FN) = 8/10 = 80%</td>
</tr>
</table>

| | |
|---|---|
| | **'Bacteria Pneumonia' Class**<br>True Positives = 9 / False Positives = 2<br>True Negatives = 18 / False Negatives = 1<br>True Positives + False Positives (TP+FP) = 11<br>True Positives + False Negatives (TP+FN) = 10<br>Precision = TP/(TP+FP) = 9/11 = 81,8%<br>Recall = TP/(TP+FN) = 9/10 = 90%<br><br>**Precision and Recall calculated using micro-averaged method:**<br>TP = 10 + 8 + 9 = 27<br>FP = 0 + 1 + 2 = 3<br>Precision = 27 / (27+3) = **90% Precision**<br><br>FN = 3 + 2 + 1<br>Recall = 27 / (27 + 6) = **81,8% Recall**<br><br>Model's precision and recall for a score threshold of 0.5<br><br> |
| **F1 Score**<br>What is this model's F1 score? | Formula of the F1 Score<br>F1 Score = 2 *( Precision * Recall)/ (Precision + Recall)<br><br>The F1 score of this model can be calculated for each class:<br>'Normal' Class: 2*( 100% * 76,9%) / ( 100% + 76,9%)<br>F1 Score = 86,9%<br><br>'Viral Pneumonia' Class:<br>2*( 88,8% * 80%) / ( 88,8% + 80%)<br>F1 Score = 84,1%<br><br>'Bacterial Pneumonia' Class:<br>2*( 81,8% * 90%) / ( 91,8% + 90%)<br>F1 Score = 85,7%<br><br>**Then the F1 score can be calculated as the average score of the three classes:**<br><br>F1 Score = (86,9% + 84,1% + 85,2%) / 3<br>**F1 Score = 85,4%** |