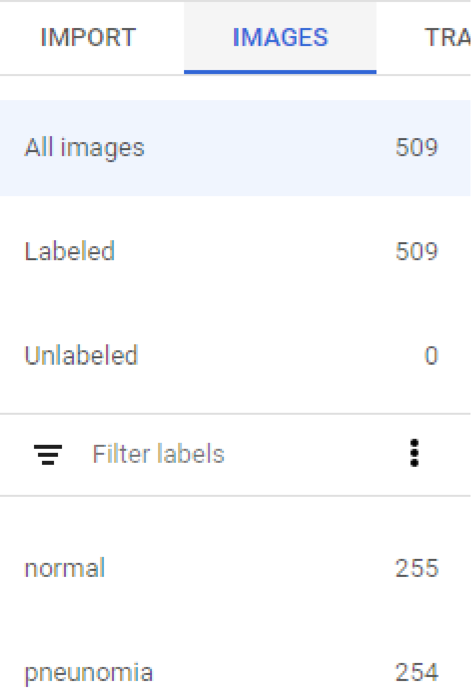
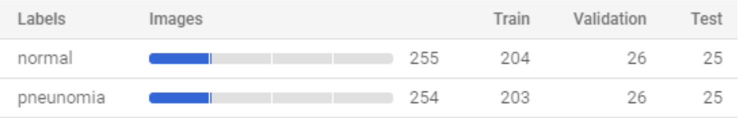
AutoML Modeling Report

Tibor Zahorecz

Binary Classifier with Clean/Balanced Data

Train/Test Split How much data was used for training? How much data was used for testing?	I used 255pcs images for normal and 254pcs images for pneunomia:  All images 509 Labeled 509 Unlabeled 0 normal 255 pneunomia 254 Instructor comments: The student correctly reports the number of images used for training and testing. Good job! But you need to use by default 177 images & 23 images for testing or justify why you have use significantly larger values than provided in the rubrics to pass the test. Student answer: I did by instruction: 'Then copy between 100-300 images of each type of each classification from the original train/ folder (normal, pneumonia) into the appropriate new folder (make sure to copy the same

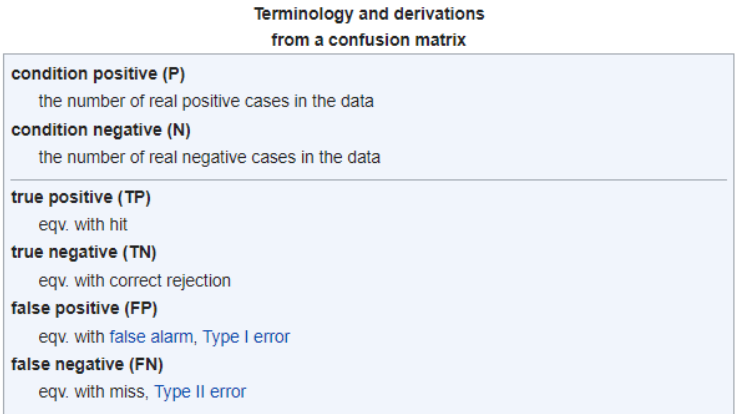
number of each type of image to have a balanced dataset)' + let Google to make train/test/validation set-up. See below
** comments: I can not make new training as I used the 300$ credit for the tasks



Labels	Images		Train	Validation	Test
normal		255	204	26	25
pneunomia		254	203	26	25

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?

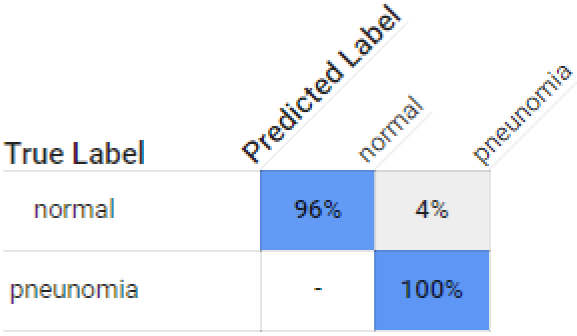
Confusion matrix (wikipedia link)
In the field of machine learning and specifically the problem of statistical classification, a confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning one (in unsupervised learning it is usually called a matching matrix). Each row of the matrix represents the instances in a predicted class while each column represents the instances in an actual class. The name stems from the fact that it makes it easy to see if the system is confusing two classes (i.e. commonly mislabeling one as another).



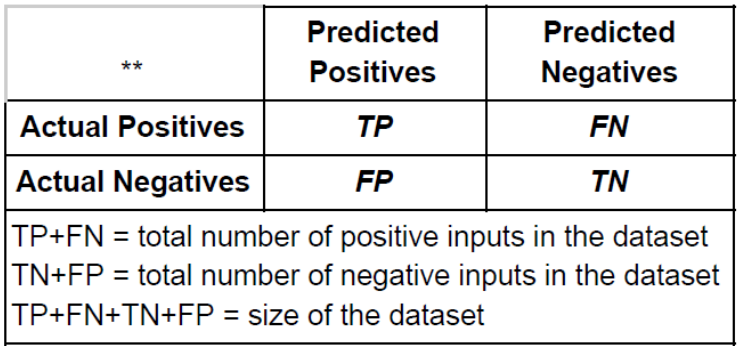
Terminology and derivations from a confusion matrix

condition positive (P)
the number of real positive cases in the data
condition negative (N)
the number of real negative cases in the data

true positive (TP)
eqv. with hit
true negative (TN)
eqv. with correct rejection
false positive (FP)
eqv. with false alarm, Type I error
false negative (FN)
eqv. with miss, Type II error



True Label	Predicted Label normal	pneunomia
normal	96%	4%
pneunomia	-	100%

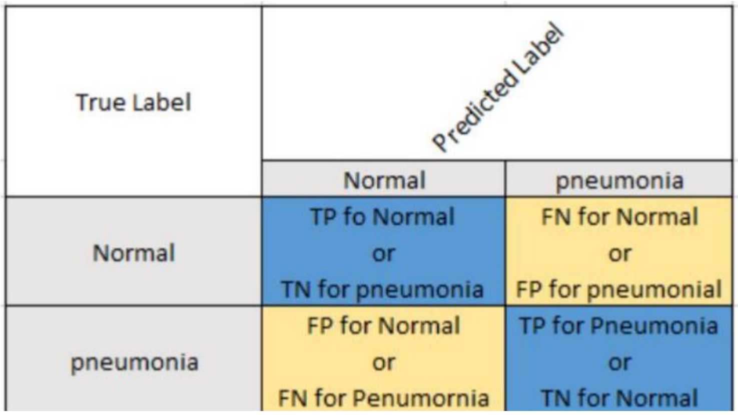


**	Predicted Positives	Predicted Negatives
Actual Positives	***TP***	***FN***
Actual Negatives	***FP***	***TN***

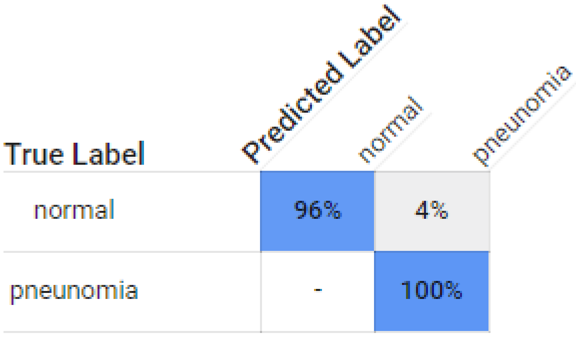
TP+FN = total number of positive inputs in the dataset
TN+FP = total number of negative inputs in the dataset
TP+FN+TN+FP = size of the dataset

TP = 96%
FP= 0%

Instructor comments: Great job explaining the confusion matrix and also on the True positive rate for the pneumonia class. However, for the False Positive Rate for the normal class, you answered the question "How many normal were classified as pneumonia?" Instead, you should answer the question "How many pneumonia were classified as normal?": 0

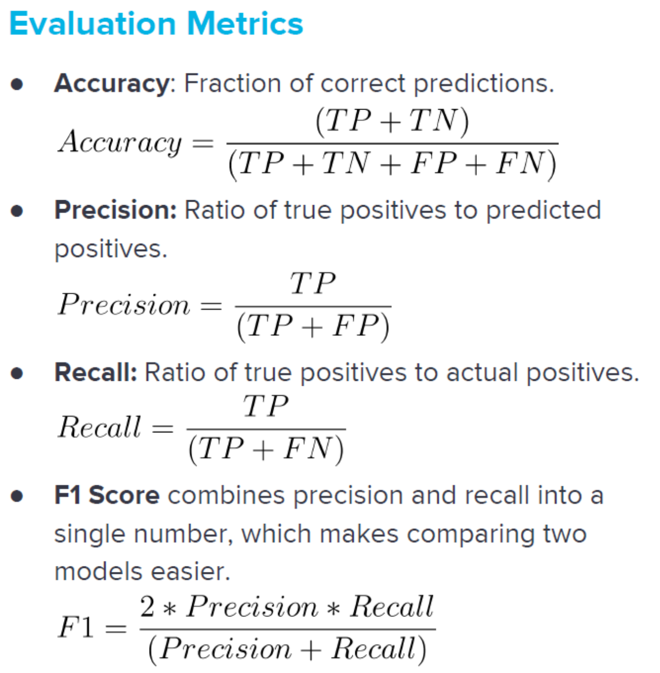


True Label	Predicted Label	
	Normal	pneumonia
Normal	TP fo Normal or TN for pneumonia	FN for Normal or FP for pneumonial
pneumonia	FP for Normal or FN for Penumornia	TP for Pneumonia or TN for Normal



True Label	Predicted Label normal	pneunomia
normal	96%	4%
pneunomia	-	100%

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)?



Evaluation Metrics

- **Accuracy**: Fraction of correct predictions.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

- **Precision:** Ratio of true positives to predicted positives.

$$Precision = \frac{TP}{(TP + FP)}$$

- **Recall:** Ratio of true positives to actual positives.

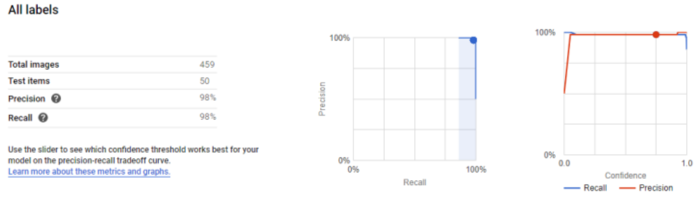
$$Recall = \frac{TP}{(TP + FN)}$$

- **F1 Score** combines precision and recall into a single number, which makes comparing two models easier.

$$F1 = \frac{2 * Precision * Recall}{(Precision + Recall)}$$

What are precision and recall?
Precision and recall help us understand how well our model is capturing information, and how much it's leaving out. Precision tells us, from all the test examples that were assigned a label, how many actually were supposed to be categorized with that label. Recall tells us, from all the test examples that should have had the label assigned, how many were actually assigned the label.

A **high precision model** produces fewer false positives.
A **high recall model** produces fewer false negatives.



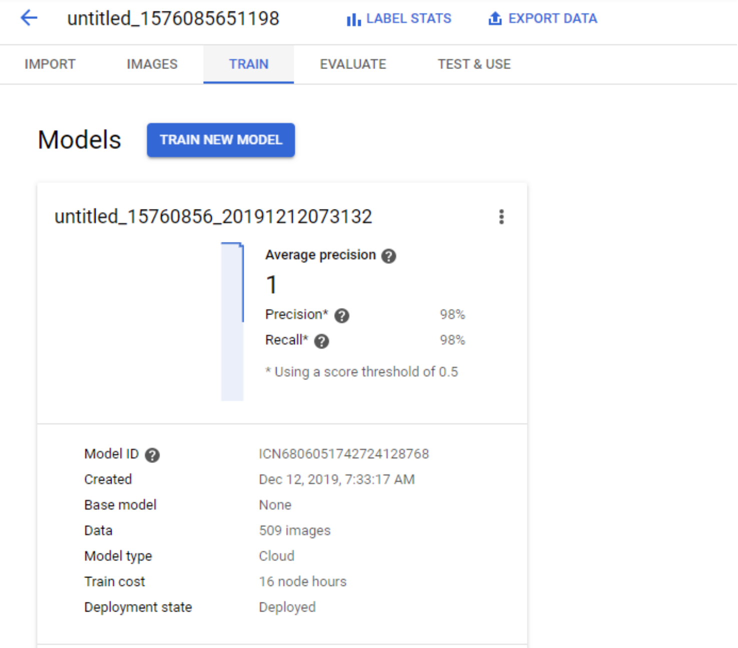
All labels

Total images	459
Test items	50
Precision ❓	98%
Recall ❓	98%

Use the slider to see which confidence threshold works best for your model on the precision-recall tradeoff curve.
Learn more about these metrics and graphs.

** **The score threshold** tool allows you to explore how your chosen score threshold affects your precision and recall. As you drag the slider on the score threshold bar, you can see where that threshold places you on the precision-recall tradeoff curve, as well as how that threshold affects your precision and recall individually (for multiclass models, on these graphs, precision and recall means the only label used to calculate precision and recall metrics is the top-scored label in the set of labels we return). This can help you find a good balance between false positives and false negatives.

Summary: I was really surprised how well this model worked out:



untitled_15760856_20191212073132

Average precision
1

Precision*	98%
Recall*	98%

* Using a score threshold of 0.5

Model ID	ICN6806051742724128768
Created	Dec 12, 2019, 7:33:17 AM
Base model	None
Data	509 images
Model type	Cloud
Train cost	16 node hours
Deployment state	Deployed

Score Threshold
When you increase the threshold what happens to precision? What happens to recall? Why?

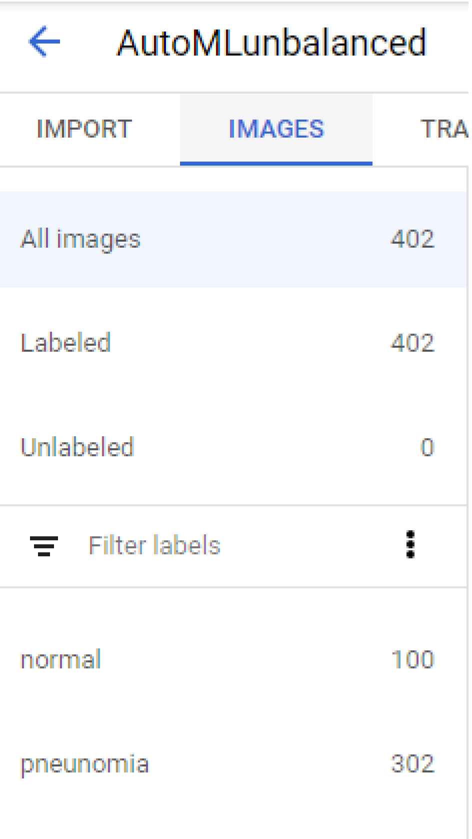
If your score threshold is low, your model will classify more images, but runs the risk of misclassifying a few images in the process. **If your score threshold is high**, your model will classify fewer images, but it will have a lower risk of misclassifying images.

Instructor comments: The student correctly explains the effect of increasing the score threshold on precision and recall, and describes why.

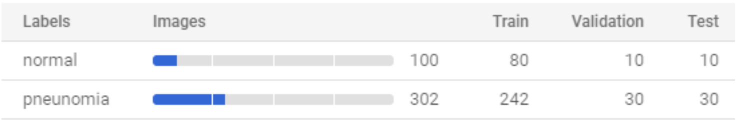
Nicely done! Requires to mention effects on precision and recall value. Does it increases or decrease for both the cases?

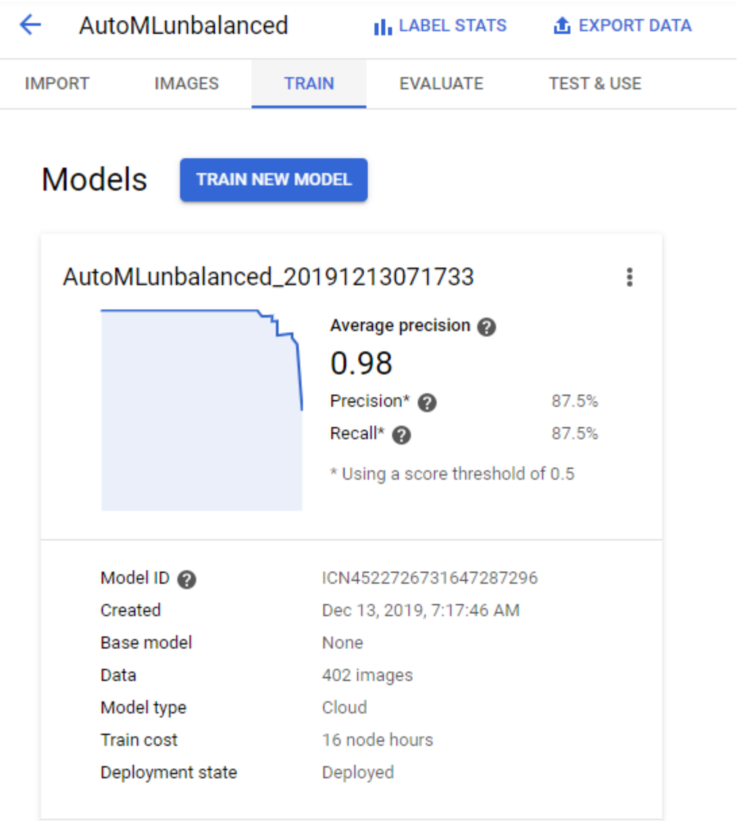
Student answer:
When we increase the threshold (from 0.5 to 0.8)
the precision goes up because the high threshold produces fewer FP.
the recall decreases.
Reference:
https://cloud.google.com/vision/automl/docs/beginners-guide

Binary Classifier with Clean/Unbalanced Data

<table>
<tr>
<td><b>Train/Test Split</b>
How much data was used for training? How much data was used for testing?</td>
<td>


All images 402
Labeled 402
Unlabeled 0
normal 100
pneumonia 302


<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>302</td><td>242</td><td>30</td><td>30</td></tr>
</table>
</td>
</tr>
<tr>
<td><b>Confusion Matrix</b>
How has the confusion matrix been affected by the unbalanced data? Include a screenshot of the new confusion matrix.</td>
<td>The model is good but far from the first case, higher false negative and positive</td>
</tr>
</table>

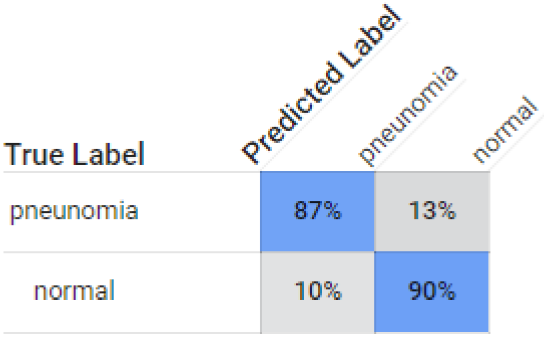


AutoMLunbalanced_20191213071733

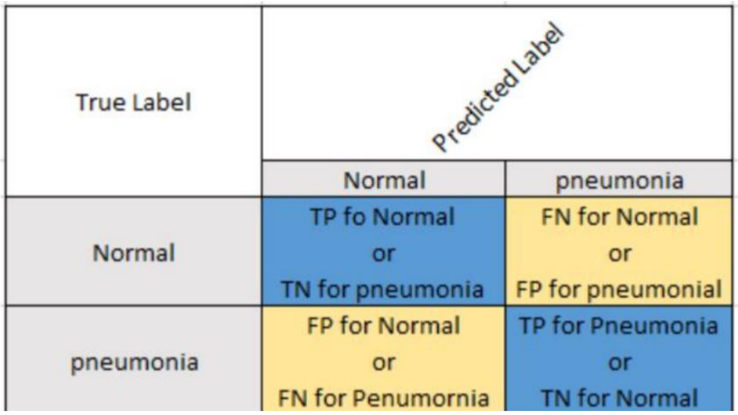
Average precision	0.98
Precision*	87.5%
Recall*	87.5%

* Using a score threshold of 0.5

Model ID	ICN4522726731647287296
Created	Dec 13, 2019, 7:17:46 AM
Base model	None
Data	402 images
Model type	Cloud
Train cost	16 node hours
Deployment state	Deployed

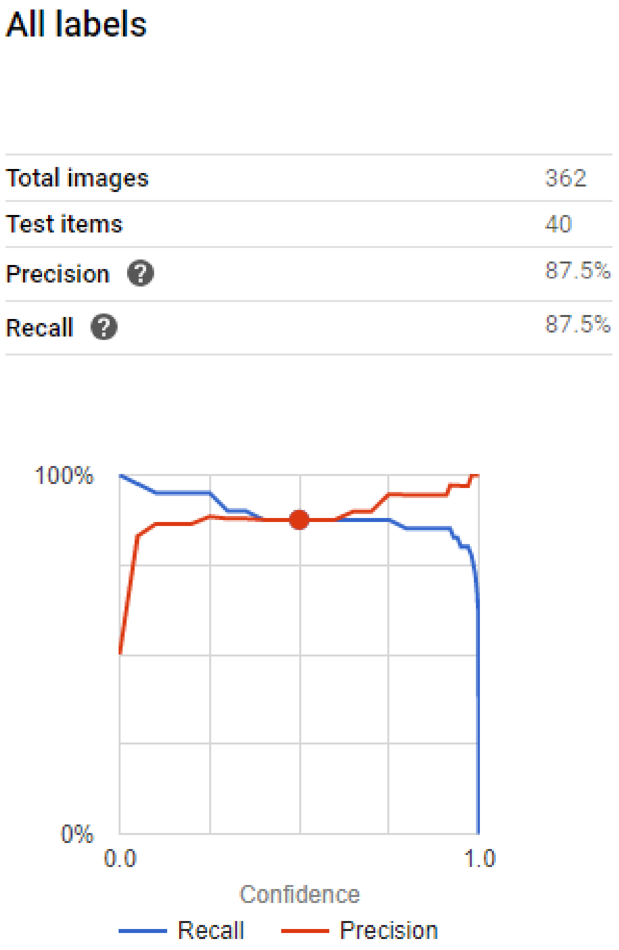


	Predicted Label	
True Label	pneumonia	normal
pneumonia	87%	13%
normal	10%	90%

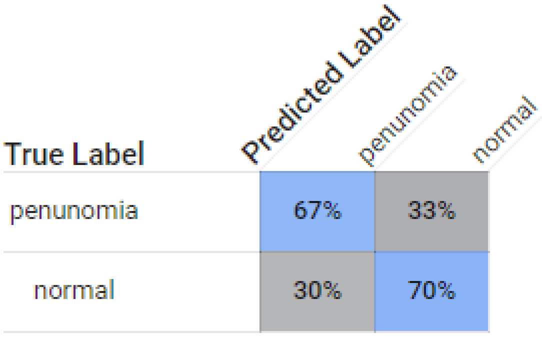
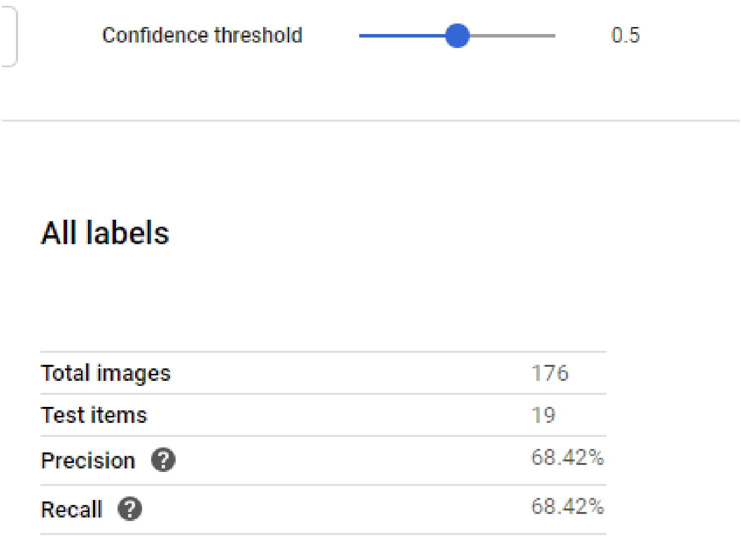


True Label	Predicted Label	
	Normal	pneumonia
Normal	TP fo Normal or TN for pneumonia	FN for Normal or FP for pneumonial
pneumonia	FP for Normal or FN for Penumornia	TP for Pneumonia or TN for Normal

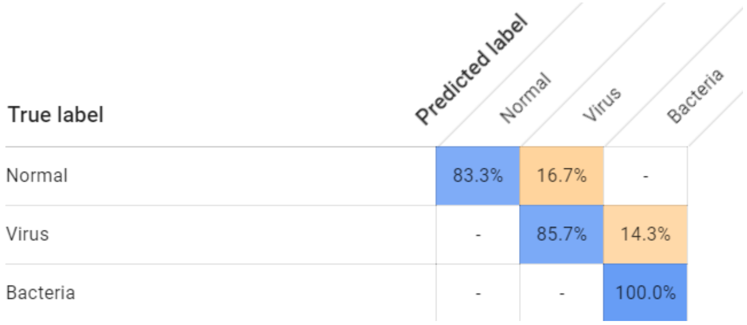
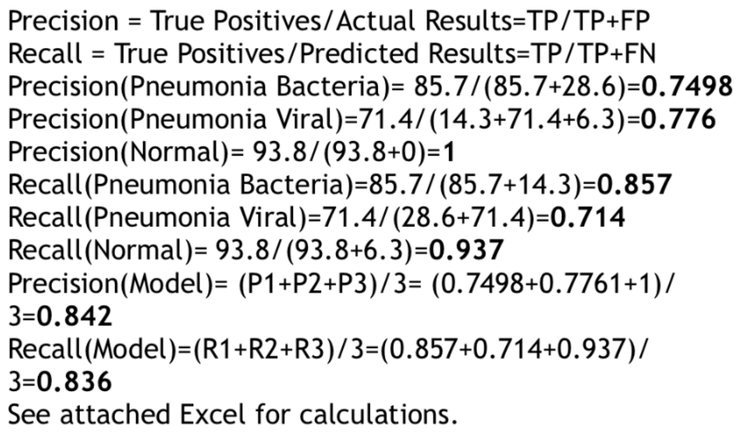
Comparing to Balanced dataset, TP went down to 87%.

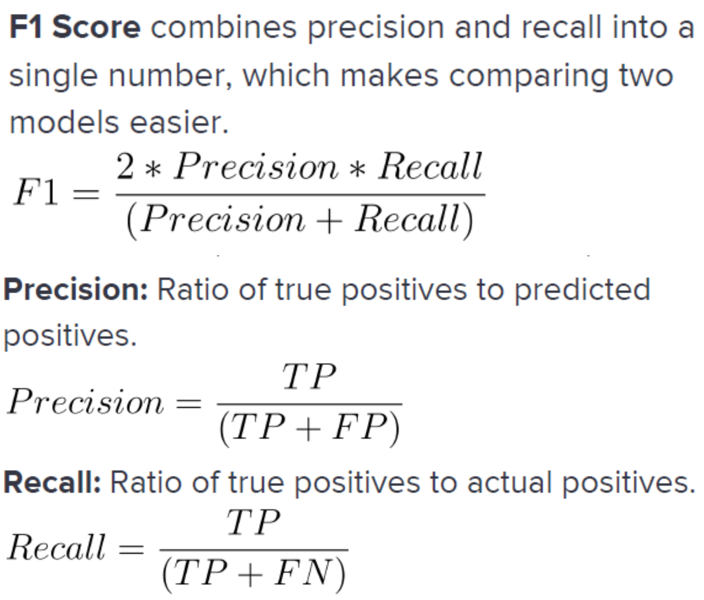
	Instructor request: What did you observed for “normal” false negatives and “pneumonia” false negatives? Does it increased or decreased? **Student answer**: The Normal and Pneumonia FN also went up (10% and 13%)
Precision and Recall How have the model’s precision and recall been affected by the unbalanced data (report the values for a score threshold of 0.5)?	 All labels Total images 362 Test items 40 Precision 87.5% Recall 87.5% 100% 0% 0.0 1.0 Confidence Recall Precision **Instructor request**: Required to explain points on How have the model’s precision and recall been affected by the unbalanced data? **Student answer**: Both Precision and Recall is 87.5% Compering with balanced dataset it is lower one.
Unbalanced Classes From what you have observed, how do unbalanced classed affect a machine learning model?	In general the model works fine and healthy but unbalanced data bring bias to the model so need more training.

Binary Classifier with Dirty/Balanced Data

<table>
<tr>
<td><b>Confusion Matrix</b>
How has the confusion matrix been affected by the dirty data? Include a screenshot of the new confusion matrix.</td>
<td>

<table>
<tr><th>True Label</th><th>Predicted Label
penunomia</th><th>normal</th></tr>
<tr><td>penunomia</td><td>67%</td><td>33%</td></tr>
<tr><td>normal</td><td>30%</td><td>70%</td></tr>
</table>
The dataset structure has a heavy impact to the outcome
</td>
</tr>
<tr>
<td><b>Precision and Recall</b>
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>

Confidence threshold 0.5
All labels
<table>
<tr><td>Total images</td><td>176</td></tr>
<tr><td>Test items</td><td>19</td></tr>
<tr><td>Precision ?</td><td>68.42%</td></tr>
<tr><td>Recall ?</td><td>68.42%</td></tr>
</table>
Both parameters are lower vs the previous experiments. Out of all these binary classifiers Clean/Balanced data has the highest Precision and Recall.
</td>
</tr>
<tr>
<td><b>Dirty Data</b>
From what you have observed, how does dirty data affect a machine learning model?</td>
<td>“Data is the new oil” means ML algorithms need a good set of date. Data should be clean and balanced.</td>
</tr>
</table>

3-Class Model

<table>
<tr>
<td><b>Confusion Matrix</b>
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>

<table>
<tr><th>True label</th><th>Predicted label Normal</th><th>Virus</th><th>Bacteria</th></tr>
<tr><td>Normal</td><td>83.3%</td><td>16.7%</td><td>-</td></tr>
<tr><td>Virus</td><td>-</td><td>85.7%</td><td>14.3%</td></tr>
<tr><td>Bacteria</td><td>-</td><td>-</td><td>100.0%</td></tr>
</table>
The normal and virus classes can be mixed by the model.
<b>Instructor request</b>: You have created perfect Dataset. Requires to provide the remedy to increase the accuracy of the model such as increase in data would help to train the model better.
<b>Student answer</b>: Thank you! The issue is I used the 300$ credit so I can not run new experiments. Hope you will accept the current experiment,
</td>
</tr>
<tr>
<td><b>Precision and Recall</b>
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>
To calculate precision and recall, calculate their values for each class after take their averages.
Precision for normal, virus and bacteria classes are 83.3%, 85,7%, 100%.

Precision = True Positives/Actual Results=TP/TP+FP
Recall = True Positives/Predicted Results=TP/TP+FN
Precision(Pneumonia Bacteria)= 85.7/(85.7+28.6)=<b>0.7498</b>
Precision(Pneumonia Viral)=71.4/(14.3+71.4+6.3)=<b>0.776</b>
Precision(Normal)= 93.8/(93.8+0)=<b>1</b>
Recall(Pneumonia Bacteria)=85.7/(85.7+14.3)=<b>0.857</b>
Recall(Pneumonia Viral)=71.4/(28.6+71.4)=<b>0.714</b>
Recall(Normal)= 93.8/(93.8+6.3)=<b>0.937</b>
Precision(Model)= (P1+P2+P3)/3= (0.7498+0.7761+1)/3=<b>0.842</b>
Recall(Model)=(R1+R2+R3)/3=(0.857+0.714+0.937)/3=<b>0.836</b>
See attached Excel for calculations.
</td>
</tr>
</table>

F1 Score What is this model’s F1 score?	 **F1 Score** combines precision and recall into a single number, which makes comparing two models easier. $F1 = \frac{2 * Precision * Recall}{(Precision + Recall)}$ **Precision:** Ratio of true positives to predicted positives. $Precision = \frac{TP}{(TP + FP)}$ **Recall:** Ratio of true positives to actual positives. $Recall = \frac{TP}{(TP + FN)}$ F1 Score = 0.89