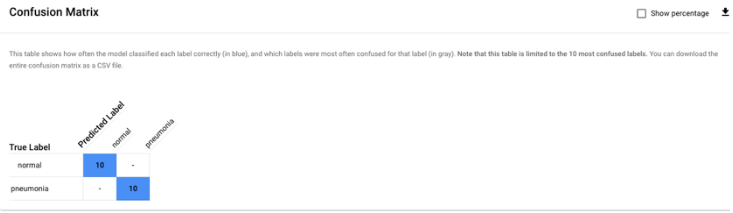
AutoML Modeling Report

Georgi Kardzhaliyski

Binary Classifier with Clean/Balanced 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>Total images: 180
Test items: 20

AutoML Vision automatically uses 80% of your images for training, 10% for validating, and 10% for testing.</td>
</tr>
<tr>
<td><b>Confusion Matrix</b>
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?</td>
<td>
Confusion Matrix
☐ Show percentage
This table shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in gray). <b>Note that this table is limited to the 10 most confused labels.</b> You can download the entire confusion matrix as a CSV file.
<table>
<tr><td>True Label</td><td>Predicted Label
normal</td><td>pneumonia</td></tr>
<tr><td>normal</td><td>10</td><td>-</td></tr>
<tr><td>pneumonia</td><td>-</td><td>10</td></tr>
</table>
A confusions matrix shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in gray). It is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. The number of correct and incorrect predictions are summarized with count values and broken down by each class.

A confusion matrix has two rows and two columns, and shows how many normal samples were predicted as Normal or Pneumonia (the first column), and how many Pneumonia samples were predicted as Normal or Pneumonia (the second column)</td>
</tr>
</table>

There are two labels in the confusion matrix: true / actual label and predicted label, with normal and pneumonia being the two categories.

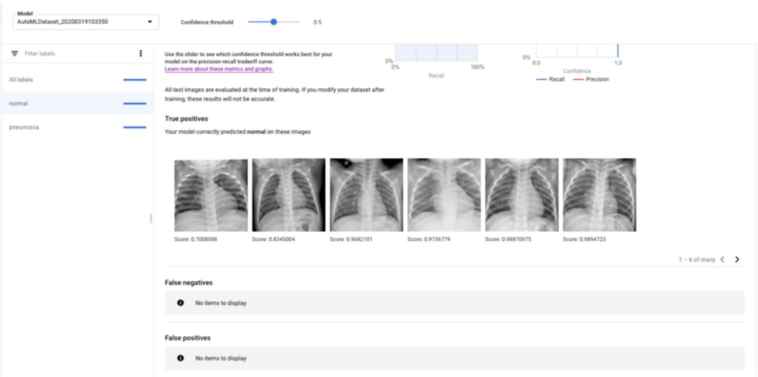
Every time the classifier makes a prediction, one of the cells in the table is incremented by one. By the end of the process, we can see exactly how our classifier performed.

https://towardsdatascience.com/multi-class-metrics-made-simple-part-i-precision-and-recall-9250280bddc2

We use the labels in the confusion matrix to understand the ways in which our classification model is confused when it makes predictions. In this case, when an image was incorrectly classified as normal when it should've been recognized as pneumonia. And the other way around.

The results from the confusion matrix show that there were no errors, all normal and pneumonia images were classified correctly. The errors are at 0%, as it can be seen from the screenshot of the confusion matrix. The classifier was perfect. All normal images were classified as normal, and all pneumonia images were classified as pneumonia.

The table gives us an insight into where we can improve our training to increase the model's accuracy.



Model: AutoMLDataset_20200319103350

Confidence threshold: 0.5

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.

All test images are evaluated at the time of training. If you modify your dataset after training, these results will not be accurate.

True positives

Your model correctly predicted **normal** on these images

Score: 0.7008588 | Score: 0.8345004 | Score: 0.9682101 | Score: 0.9736779 | Score: 0.98870975 | Score: 0.9894723

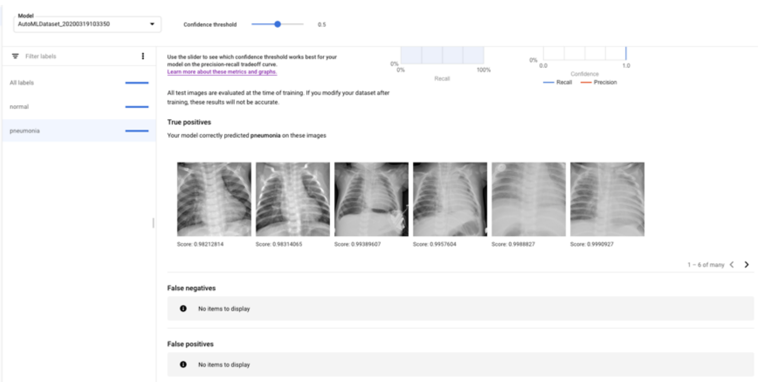
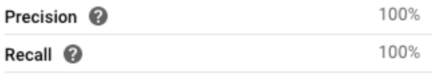
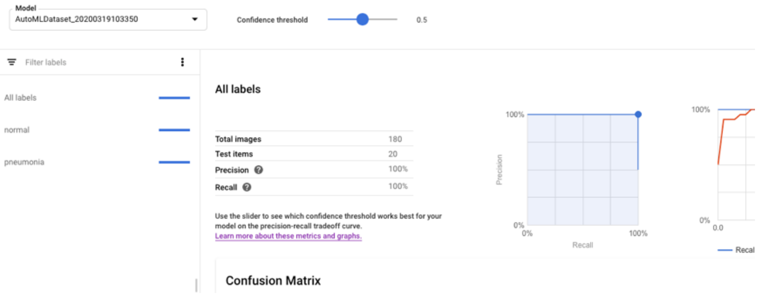
1 – 6 of many

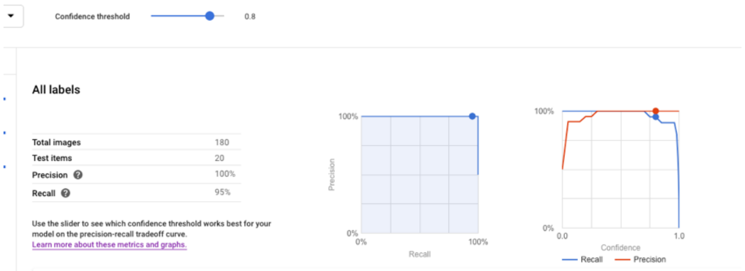
False negatives

No items to display

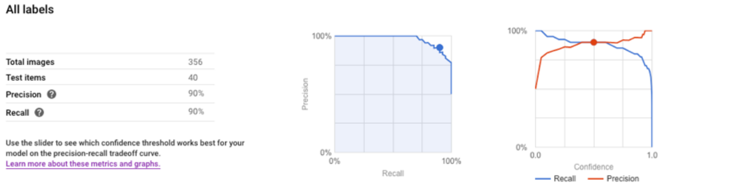
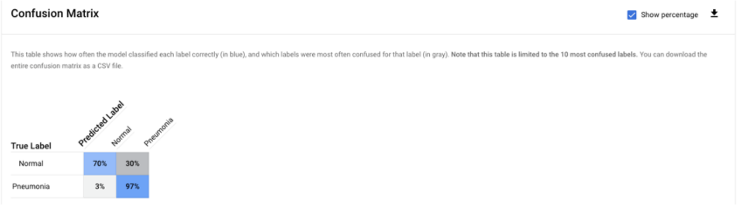
False positives

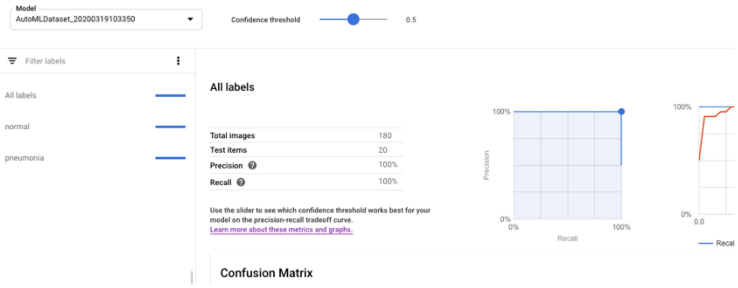
No items to display

	 Model AutoMLDataset_20200319103350 Confidence threshold 0.5 Filter labels All labels normal pneumonia 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. All test images are evaluated at the time of training. If you modify your dataset after training, these results will not be accurate. True positives Your model correctly predicted **pneumonia** on these images Score: 0.98212814 Score: 0.98314065 Score: 0.99389607 Score: 0.9957604 Score: 0.9988827 Score: 0.9990927 1 – 6 of many False negatives No items to display False positives No items to display 0% 0% 100% Recall 0% 0.0 1.0 Confidence Recall Precision A high precision produces fewer false positives. A high recall produces fewer false negatives,
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)?	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.  Precision ? 100% Recall ? 100%  Model AutoMLDataset_20200319103350 Confidence threshold 0.5 Filter labels All labels normal pneumonia All labels Total images 180 Test items 20 Precision ? 100% Recall ? 100% 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. Confusion Matrix 100% Precision 0% 0% 100% Recall 100% 0% 0.0 Recal

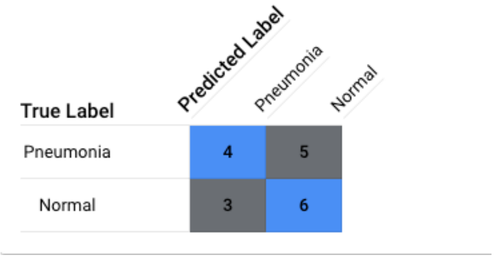
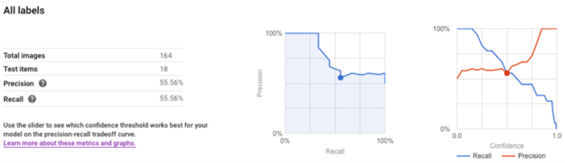
Score Threshold When you increase the threshold what happens to precision? What happens to recall? Why?	When I increased the confidence threshold, the prevision % stayed the same (100%), but the recall % went down from 100% to 95%. The score threshold refers to the level of confidence the model must have to assign a category to a test item. 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.  Confidence threshold 0.8 All labels Total images 180 Test items 20 Precision 100% Recall 95% 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. 100% 0% Precision 0% 100% Recall 100% 0% 0.0 1.0 Confidence Recall Precision

Binary Classifier with Clean/Unbalanced Data

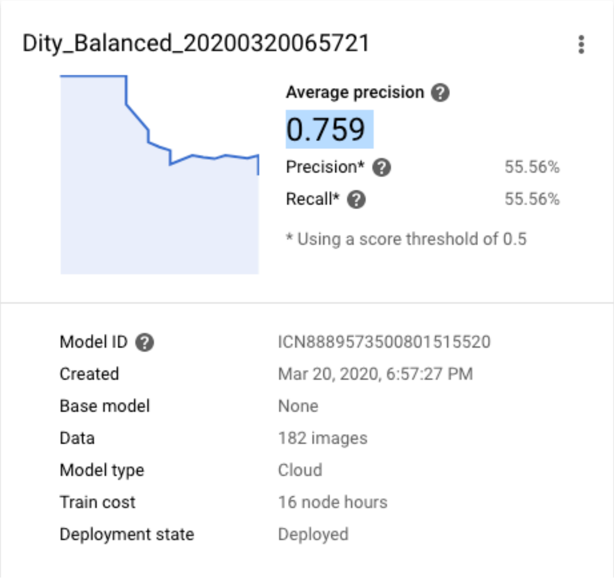
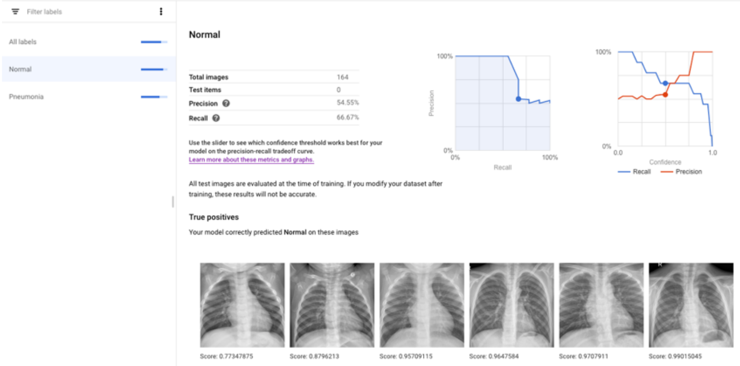
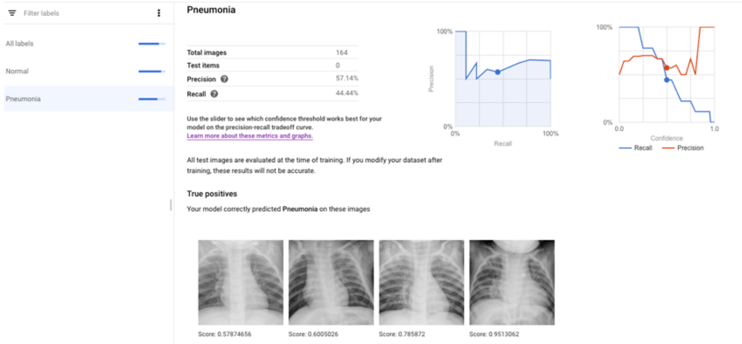
Train/Test Split How much data was used for training? How much data was used for testing?	 All labels Total images 356 Test items 40 Precision 90% Recall 90% Use the slider to see which confidence threshold works best for your model on the precision-recall tradeoff curve. Learn more about these metrics and graphs. 100% 0% Precision 0% 100% Recall 100% 0% 0.0 1.0 Confidence Recall Precision
Confusion Matrix How has the confusion matrix been affected by the unbalanced data? Include a screenshot of the new confusion matrix.	 Confusion Matrix ☑ Show percentage This table shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in gray). **Note that this table is limited to the 10 most confused labels.** You can download the entire confusion matrix as a CSV file. True Label | Predicted Label: Normal | Pneumonia Normal | 70% | 30% Pneumonia | 3% | 97% There are more misclassified images when there is unbalanced data. Unbalanced data causes the model to skew towards the incorrect classification, because there

	is more noise in the data.
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)?	 Model: AutoMLDataset_20200319103350 Confidence threshold 0.5 All labels Total images 180 Test items 20 Precision 100% Recall 100% 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. Confusion Matrix Filter labels All labels normal pneumonia 100% 0% Precision Recall 0.0 Recal The precision and recalled are lower than when the data were clean and balanced.
Unbalanced Classes From what you have observed, how do unbalanced classed affect a machine learning model?	Models will only learn about data that they are trained with. When we use unbalanced data, this causes the model to skew towards a particular outcome.

Binary Classifier with Dirty/Balanced Data

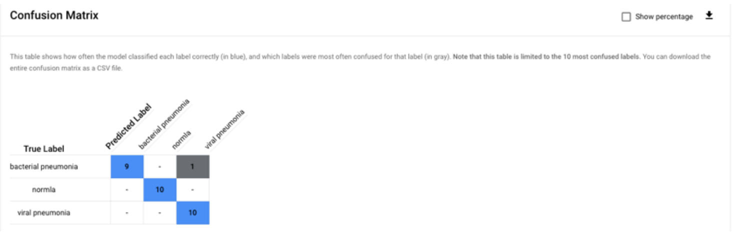
Confusion Matrix How has the confusion matrix been affected by the dirty data? Include a screenshot of the new confusion matrix.	 True Label / Predicted Label: Pneumonia, Normal Pneumonia: 4, 5 Normal: 3, 6 More pneumonia images were misclassified as normal. More normal images were misclassified as pneumonia images.
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?	 All labels Total images 164 Test items 18 Precision 55.56% Recall 55.56% 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. 100% 0% Precision Recall 0.0 1.0 Confidence Recall Precision Precision and recall are both at 55.56%. The average precision is 0.759. It measures how well the model performs across all score thresholds.

	The precision % for the normal is 55.55, while the precision % for the pneumonia is 57.14%. Of the binary classifiers (clean/balanced, dirty/balanced and clean/unbalanced), the precision and recall values at 0.5 score threshold for each of them were as follows: Clean/balanced: 1. Precision: 100% 2. Recall: 100% Dirty/balanced: 1. Precision: 55.56% 2. Recall: 55.56% Clean/unbalanced: 1. Precision: 90% 2. Recall: 90% The clean/balanced classifier had the highest score.

	 Dity_Balanced_20200320065721 Average precision 0.759 Precision* 55.56% Recall* 55.56% * Using a score threshold of 0.5 Model ID ICN8889573500801515520 Created Mar 20, 2020, 6:57:27 PM Base model None Data 182 images Model type Cloud Train cost 16 node hours Deployment state Deployed  Filter labels All labels Normal Pneumonia 100% 0% Precision Recall 0.0 1.0 Confidence Recall Precision Normal Total images 164 Test items 0 Precision 54.55% Recall 66.67% 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. All test images are evaluated at the time of training. If you modify your dataset after training, these results will not be accurate. True positives Your model correctly predicted **Normal** on these images Score: 0.77347875 Score: 0.8796213 Score: 0.95709115 Score: 0.9647584 Score: 0.9707911 Score: 0.99015045  Filter labels All labels Normal Pneumonia 100% 0% Precision Recall 0.0 1.0 Confidence Recall Precision Pneumonia Total images 164 Test items 0 Precision 57.14% Recall 44.44% 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. All test images are evaluated at the time of training. If you modify your dataset after training, these results will not be accurate. True positives Your model correctly predicted **Pneumonia** on these images Score: 0.57874656 Score: 0.6005026 Score: 0.785872 Score: 0.9513062
Dirty Data From what you have observed, how does dirty data affect a machine learning model?	Mislabeled, dirty data, significantly impacts the model performance, the precision and recall metrics, confusion metrics.

3-Class Model

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.



Confusion Matrix

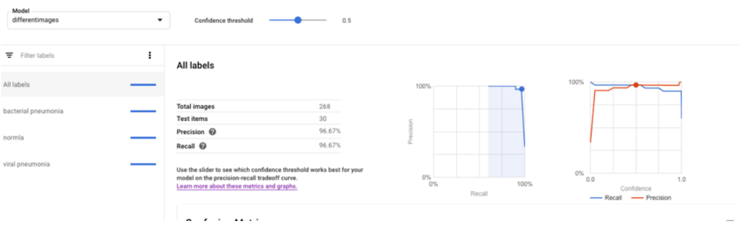
□ Show percentage

This table shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in gray). **Note that this table is limited to the 10 most confused labels.** You can download the entire confusion matrix as a CSV file.

True Label \ Predicted Label	bacterial pneumonia	normla	viral pneumonia
bacterial pneumonia	9	-	1
normla	-	10	-
viral pneumonia	-	-	10

The model is most likely to confuse the viral pneumonia. It classified the normal and viral pneumonia at 100%. In an ideal model, all the values on the diagonal will be high, and all the other values will be low. This shows that the desired categories are being identified correctly.
To remedy the confusion would be to collect more data. A larger dataset might expose a different and perhaps more balanced perspective on the classes. I can add more images of the under-represented class and delete images of the over-represented class. This could give a boost in the accuracy measures. I might also look at using a new platform with different algorithms to spot-check the results I'm getting.

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)?



Model: differentimages

Confidence threshold: 0.5

All labels

Total images	268
Test items	30
Precision	96.67%
Recall	96.67%

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 value of the precision in the multiclass classification is 96.67%; the value of the recall is 96.67% as well.

All labels:
Precision = 96.67%
Recall = 96.67%

Bacterial pneumonia:
Precision = 100%
Recall = 90%

Normal:
Precision = 100%
Recall = 100%

Viral pneumonia:
Precision: 90.91%
Recall: 100%

Precision is calculated using this formula for a binary case problem, where:

TP = True Positive
FP = False Positive
FN = False Negative
TN = True Negative
P=Precision
R=Recall

P = TP / (TP + FP)

Recall is calculated using this formula:
R= TP /(TP+FN)

Similar to a binary case, we can calculate the precision and recall for each of the classes (normal, bacterial pneumonia, viral pneumonia) using the formulas above.

For a multiclass classification with 3 possible output labels, we would need to sum up the values by row and column, so we'd have for the sum of each column:

Total_Bacterial_Pneumonia=9
Total_Normal = 10
Total_Viral_Pneumonia = 11

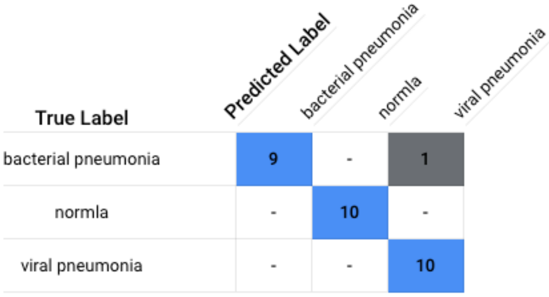
For the sum of rows for the predicted label:
Total_Predicted_Bacterial_Pneumonia = 10
Total_Predicted_Normal = 10
Total_Predicted_Pneumonia = 10

In an imbalanced classification problem with more than two classes, precision is calculated as the sum of true positives across all classes divided by the sum of true positives and false positives across all classes.

- Precision = Sum c in C TruePositives_c / Sum c in C (TruePositives_c + FalsePositives_c)

In an imbalanced classification problem with more than two classes, recall is calculated as the sum of true positives across all classes divided by the sum of true positives and false negatives across all classes.

- Recall = Sum c in C TruePositives_c / Sum c in C (TruePositives_c + FalseNegatives_c)



	Predicted Label		
True Label	bacterial pneumonia	normla	viral pneumonia
bacterial pneumonia	9	-	1
normla	-	10	-
viral pneumonia	-	-	10

F1 Score
What is this model’s F1 score?

F 1 = 2 · p · r /p + r
F1 = 2*(0.97*0.97) / 0.97 + 0.97
F1 = 2*0.94/ 1.94
F1=1.88/1.94
F1=0.96