|  |  |
| --- | --- |
| AutoML Modeling Report |  |

*<Erkan Hatipoglu >*

Binary Classifier with Clean/Balanced Data

|  |  |
| --- | --- |
| **Train/Test Split**  How much data was used for training? How much data was used for testing? | ***“Training Dataset:*** *The sample of data used to fit the model.*  ***Validation Dataset:*** *The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters. The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration.*  ***Test Dataset:*** *The sample of data used to provide an unbiased evaluation of a final model fit on the training dataset.”* **\***  There were 200 images in total (100 normal & 100 pneumonia). From this dataset, 160 images (80 normal & 80 pneumonia) were used for training, 20 images were used for testing (10 normal & 10 pneumonia) and 20 images were used for validation (10 normal & 10 pneumonia) as shown in the image below:    **\* https://machinelearningmastery.com/difference-test-validation-datasets/** |
| **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 is a grid which shows all the predicted labels relative to all true labels. In our case we have 4 cells in confusion matrix. For ***normal*** class those cells can be explained as below:   1. True Positives (Row 1 Column 1, TP in short)   Positive labels (actual) that are predicted as positives   1. False Negatives (Row 1 Column 2, FN in short)   Positive labels (actual) that are predicted as negatives   1. False Positives (Row 2 Column 1, FP in short)   Negative labels (actual) that are predicted as positives   1. True Negatives (Row 2 Column 2, TN in short)   Negative labels (actual) that are predicted as negatives  In Google Vision Confusion matrix can be displayed using percentages or item counts. I have included both images below.  As can be seen from the image below, all the normal images (10 in total) were predicted as normal. On the other hand, 2 of the pneumonia images (10 in total) were predicted as normal and 8 of the pneumonia images were predicted as pneumonia.    As a result, for ***normal*** class:   * TP = 10 (100%), * FN = 0 (0%), * FP = 2 (20%), * TN = 8 (80%).   For ***pneumonia*** class:   * TP = 8 (80%), * FN = 2 (20%), * FP = 0 (0%), * TN = 10 (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)? | Precision and recall measure the model such that we can understand how the model performs for an individual class, as well as how it performs across classes.  ***Model precision*** measures the percentage of correct predictions against total number of predictions.  ***Model recall*** measures the percentage of correctly identified instances against total possible instances.  ***Normal class***  ***Precision*** = TP/(TP+FP) = 10/(10+2) = 10/12 ~= 0,83  ***Recall*** = TP/(TP+FN) = 10/(10+0) = 10/10 = 1  ***Pneumonia class***  ***Precision*** = TP/(TP+FP) = 8/(8+0) = 8/8 = 1  ***Recall*** = TP/(TP+FN) = 8/(8+2) = 8/10 = 0,8 |
| **Score Threshold**  When you increase the threshold what happens to precision? What happens to recall? Why? | ***Normal class***  When we increase the threshold (0,60 for example) first the number of TP starts to decrease, and the number of FN starts to increase since there are some relatively low-level confidence normal images (P=0,59), but the number of FP stays the same. As a result, both precision and recall decrease.  For Confidence threshold = 0,6  ***Precision*** = TP/(TP+FP) = 9/(9+2) = 9/11 ~= 0,81  ***Recall*** = TP/(TP+FN) = 9/(9+1) = 9/10 = 0,9  If we increase the threshold further (0,75 for example) this time both TP and FP decrease and FN increases. The decrease of FP is due to some relatively mid-level confidence pneumonia images (P=0,71). That is some pneumonia images have been given a high probability of being normal. As a result, precision increases but recall decreases.  For Confidence threshold = 0,75  ***Precision*** = TP/(TP+FP) = 8/(8+1) = 8/9 ~= 0,89  ***Recall*** = TP/(TP+FN) = 8/(8+2) = 8/10 = 0,8  If we increase the threshold further (0,99 for example) both TP and FP continue to decrease, and TP becomes 1 while FP becomes 0. FN continues to increase and becomes 9. As a result, precision increases to 100% but recall decreases to 10%.  For Confidence threshold = 0,99  ***Precision*** = TP/(TP+FP) = 1/(1+0) = 1/1 = 1  ***Recall*** = TP/(TP+FN) = 1/(1+9) = 1/10 = 0,1  If we increase the threshold to 1, both TP and FP continue to decrease. For this case TP and FP becomes 0. FN continues to increase and becomes 10. As a result, precision increases to 100% but recall decreases to 0%.  For Confidence threshold = 1  ***Precision*** = TP/(TP+FP) = 0/(0+0) = 0/0 = Unknown (approaches 100%).  ***Recall*** = TP/(TP+FN) = 0/(0+10) = 0/10 = 0  Consequently, for the normal class, we can say that precision tends to increase (to 100%), and recall tends to decrease (to 0%) for higher threshold values.  ***Pneumonia class***  Like normal class precision tends to increase (to 100%), and recall tends to decrease (to 0%) for higher threshold values. |

Binary Classifier with Clean/Unbalanced Data

|  |  |
| --- | --- |
| **Train/Test Split**  How much data was used for training? How much data was used for testing? | There were 400 images in total (100 normal & 300 pneumonia). From this dataset, 320 images (80 normal & 240 pneumonia) were used for training, 40 images were used for testing (10 normal & 30 pneumonia) and 40 images were used for validation (10 normal & 30 pneumonia) as shown in the image below: |
| **Confusion Matrix**  How has the confusion matrix been affected by the unbalanced data? Include a screenshot of the new confusion matrix. | With unbalanced data that have a majority of pneumonia images, prediction of normal images has been affected negatively. As can be seen in the images below while 100% (10 out of 10) of predictions of normal images were correct for the balanced dataset, for the unbalanced dataset only 60% (6 out of 10) of normal images are correct. On the other hand, all the pneumonia images were predicted as pneumonia.  This is an expected result since the model tends to predict in favor of unbalanced label (pneumonia).    For ***normal*** class:   * TP = 6 (60%), * FN = 4 (40%), * FP = 0 (0%), * TN = 30 (100%).   For ***pneumonia*** class:   * TP = 30 (100%), * FN = 0 (0%), * FP = 4 (40%), * TN = 6 (60%).   We can conclude from the above results that for the normal class with unbalanced data TP and FP decrease and TN and FN increase. |
| **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)? |  |
| **Unbalanced Classes**  From what you have observed, how do unbalanced classed affect a machine learning model? |  |

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. |  |
| **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? |  |
| **Dirty Data**  From what you have observed, how does dirty data affect a machine learning model? |  |

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. |  |
| **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)? |  |
| **F1 Score**  What is this model’s F1 score? |  |