**Reports on evaluation of Yolov5 custom training model performance:**

The provided report gives information on how well the YOLOv5 model performed on both the training and test data for each epoch of training. The report includes several metrics for evaluating the performance of the model, including box\_loss, obj\_loss, cls\_loss, precision, recall, mAP\_0.5, and mAP\_0.5:0.95.

For each epoch of training, the report provides the values of these metrics for both the training and validation data sets. The box\_loss metric represents the localization loss, which measures how well the model is able to accurately predict the bounding boxes around the objects in the image. The obj\_loss metric represents the confidence loss, which measures how well the model is able to correctly identify objects in the image. The cls\_loss metric represents the classification loss, which measures how well the model is able to accurately classify the objects in the image.

The precision metric measures the percentage of predicted objects that are correctly identified, while the recall metric measures the percentage of actual objects that are correctly identified. The mAP\_0.5 metric represents the mean average precision for an IoU (intersection over union) threshold of 0.5, which is a commonly used metric for evaluating object detection models. The mAP\_0.5:0.95 metric represents the mean average precision for IoU thresholds ranging from 0.5 to 0.95.

In addition to these metrics, the report also provides information on the learning rate for each epoch of training. This information can be useful for monitoring the learning rate schedule and making adjustments to improve the performance of the model.

**Evaluation of the performance metrics:**

During the first few epochs, the train and validation losses for box, object, and class detections decrease, while the precision, recall, and mAP increase. This suggests that the model is learning and improving over time. The learning rate also decreases over time, which is a common technique for improving model performance.

Around epoch 50, the model appears to reach a local minimum as the losses and metrics stabilize. This is a good indication that the model has converged to a reasonable solution. However, the metrics show some variation after epoch 50, indicating that the model performance may fluctuate.

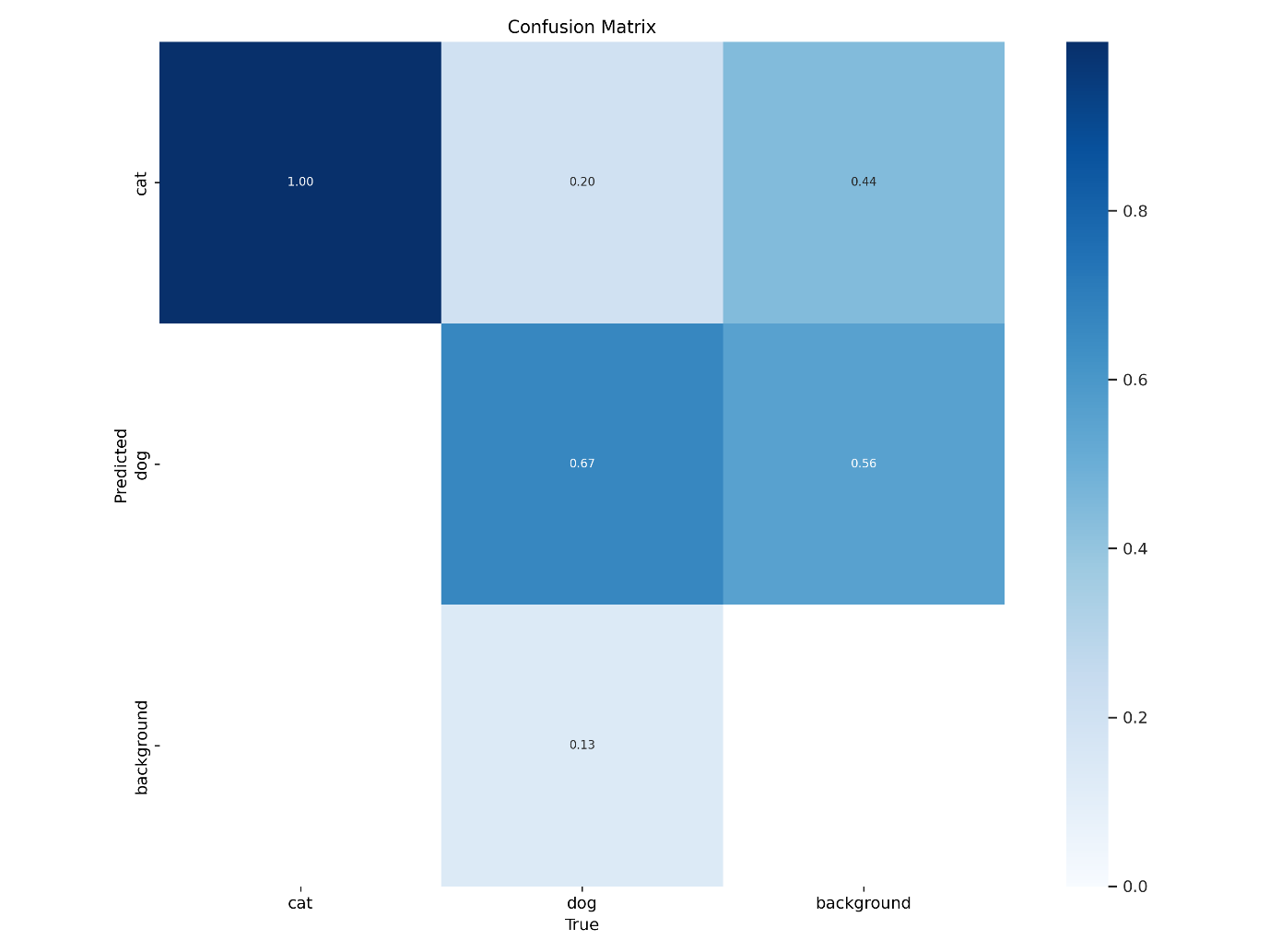
By epoch 100, the box, object, and class detection losses have decreased substantially, and the precision, recall, and mAP have all improved especially the mAP 0.5:0.95. The learning rate is also quite small by this point, which suggests that the model is nearing convergence.

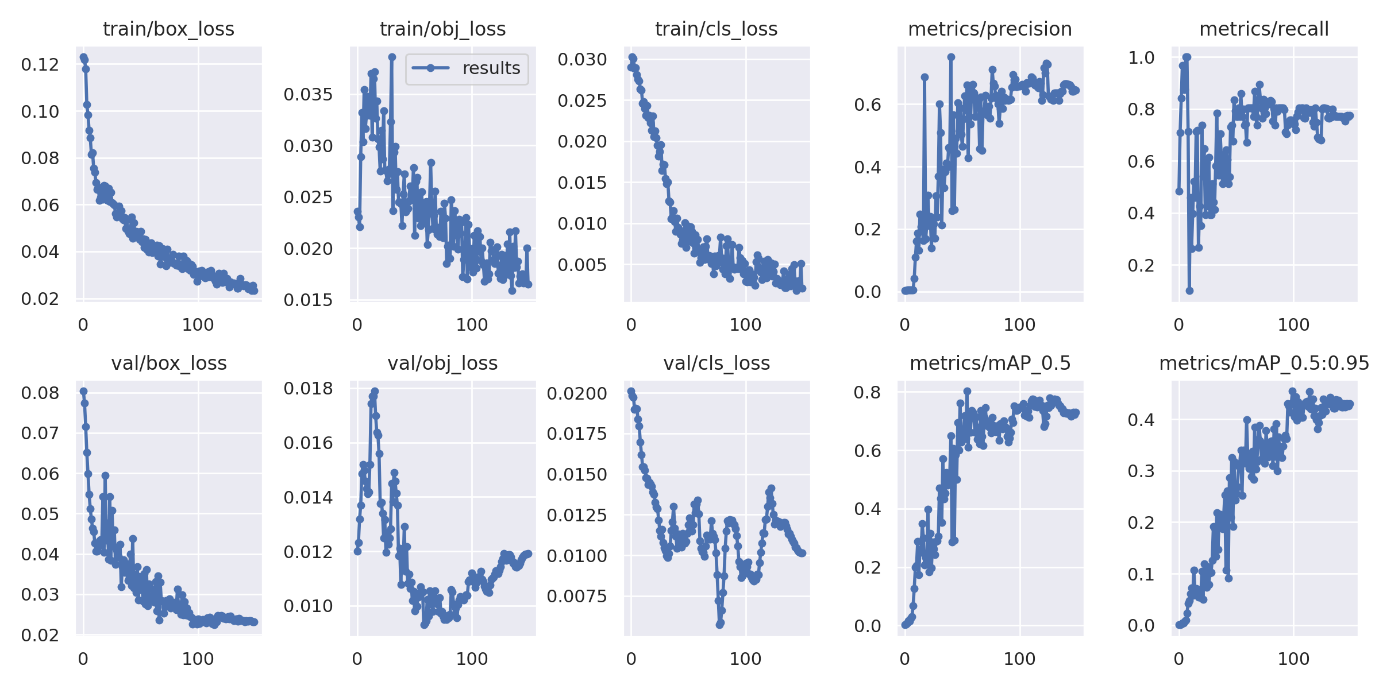
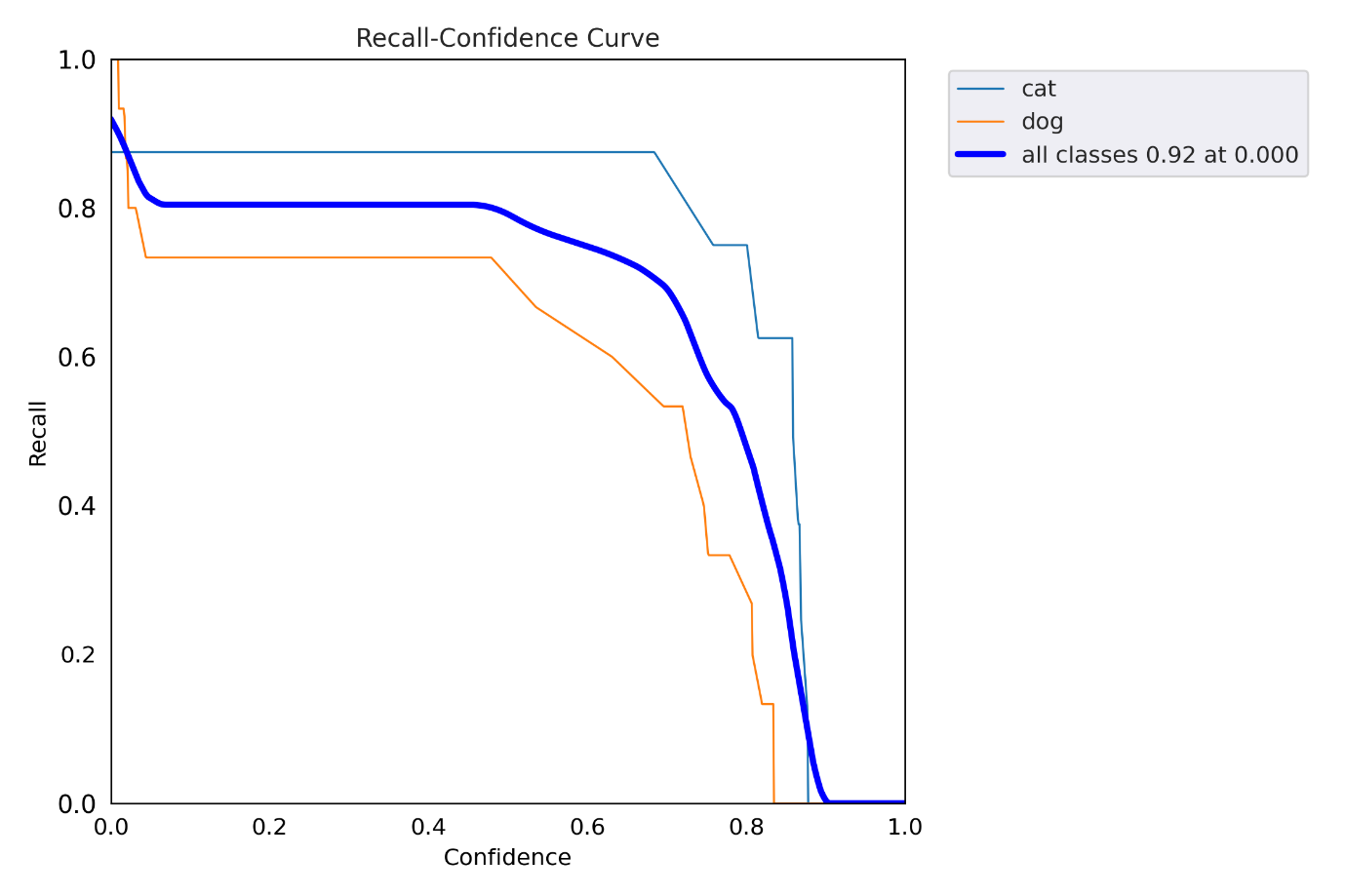
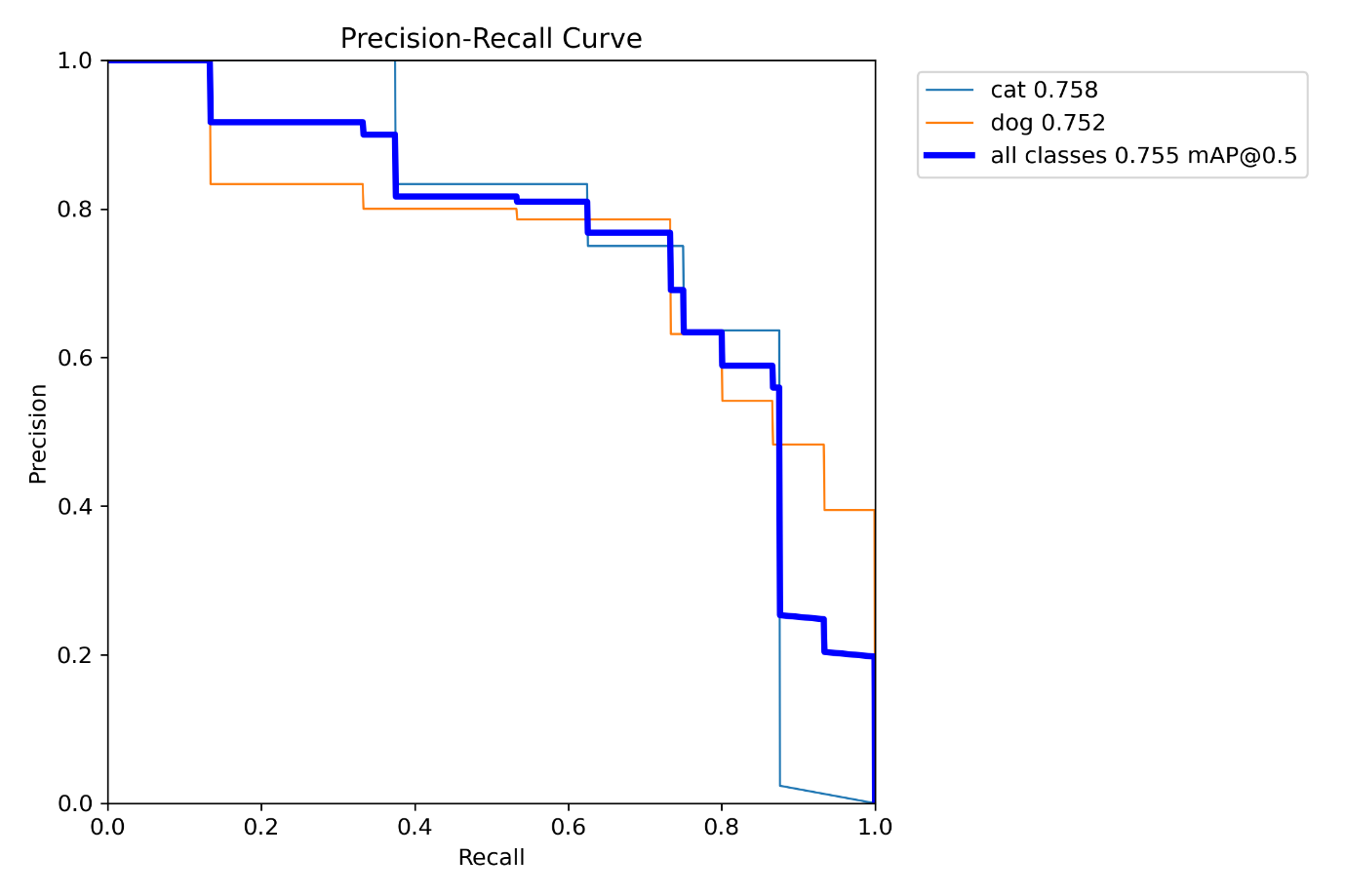
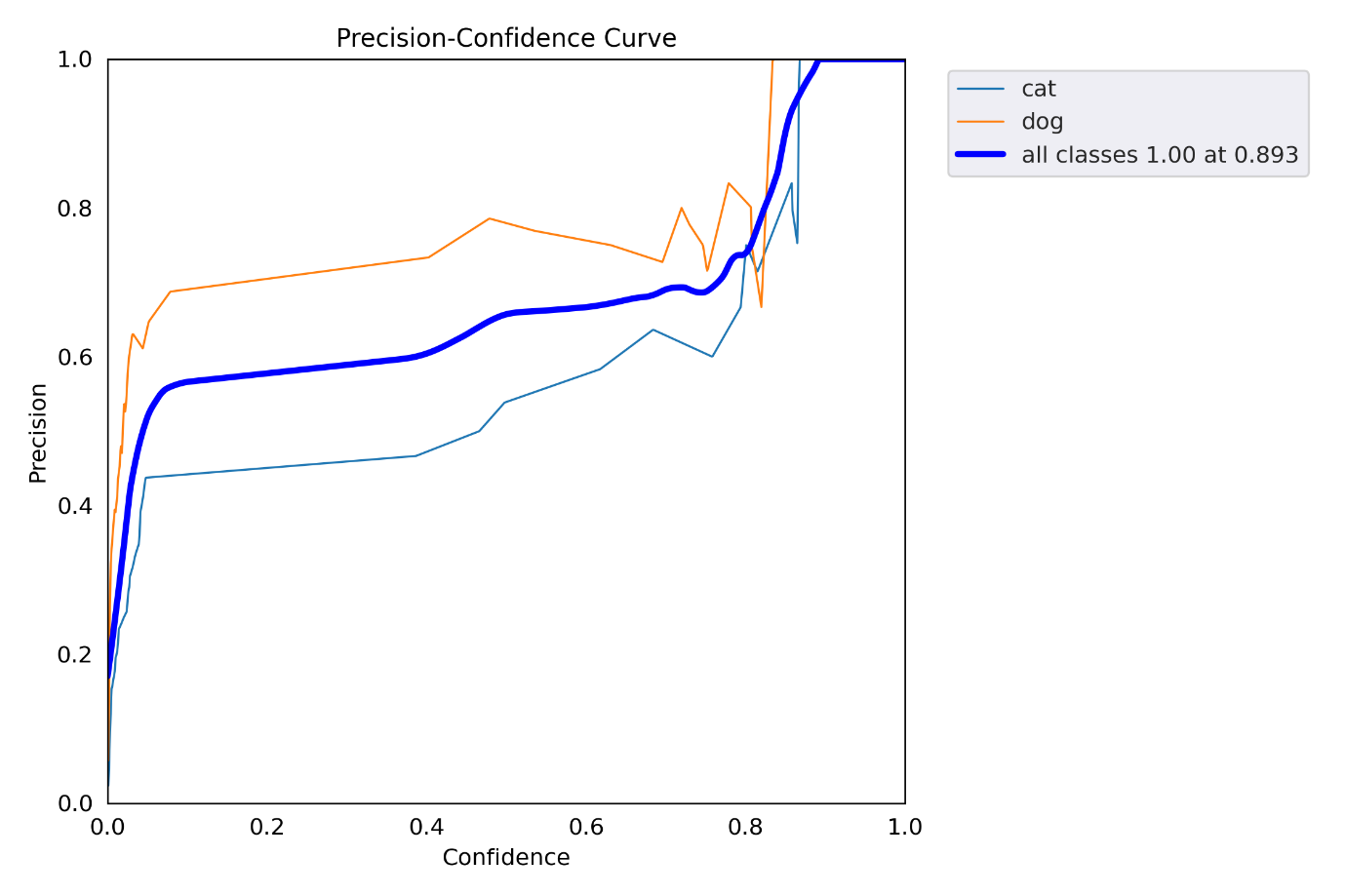
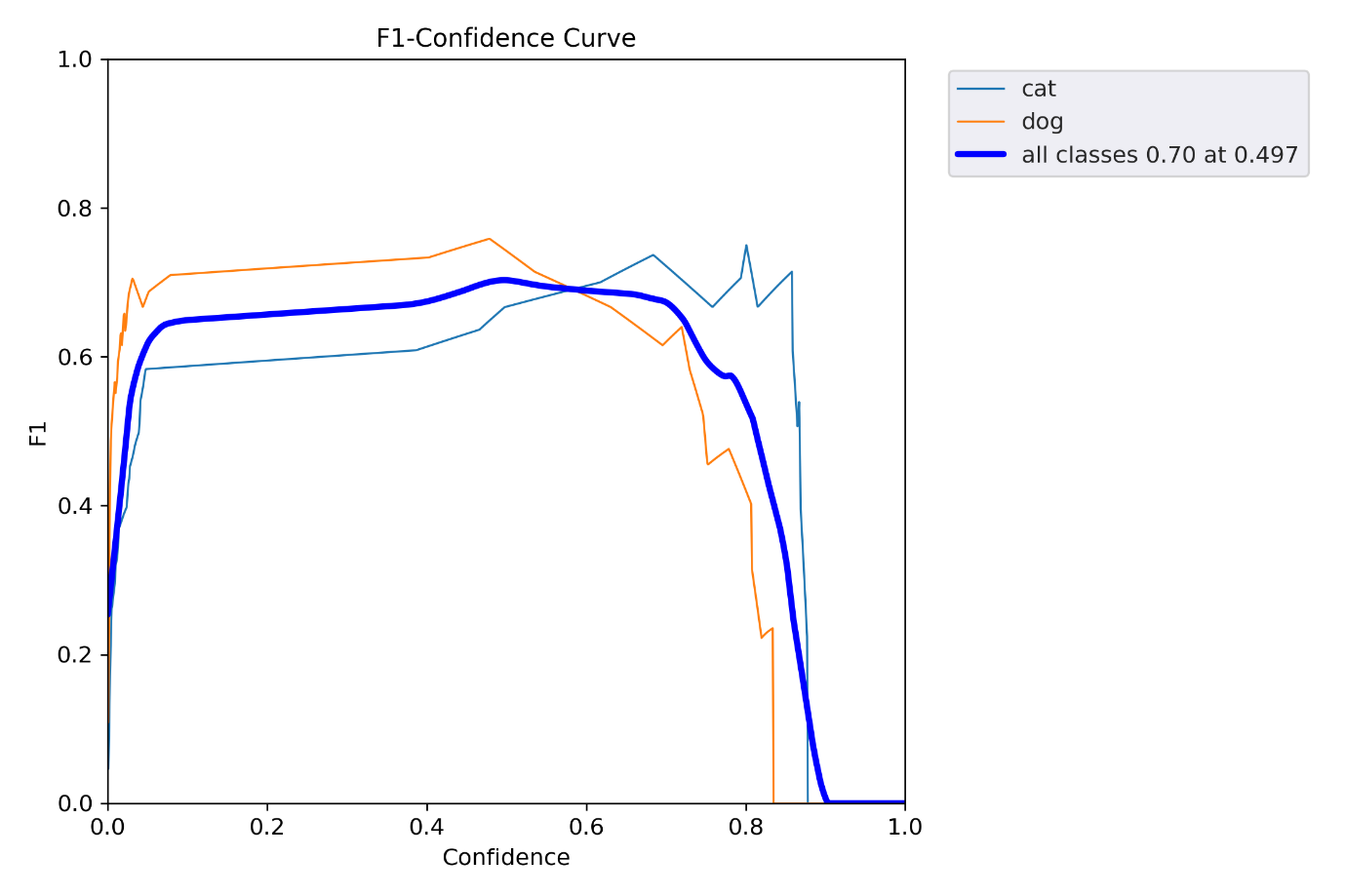
After the 135th epoch, the model achieved a box loss of 0.026803, object loss of 0.020198, and class loss of 0.0029202. The precision, recall, and mAP 0.5 were 0.64087, 0.77383, and 0.7413, respectively. The model's mAP 0.5:0.95 was 0.42147, and the validation box loss, object loss, and class loss were 0.023962, 0.011554, and 0.011603, respectively. The learning rate in this epoch was 0.00109 for all three levels (x/lr0, x/lr1, x/lr2).

In the 149th epoch, the box loss was 0.023273, the object loss was 0.016514, and the class loss was 0.0020681. The precision, recall, and mAP 0.5 were 0.64535, 0.77355, and 0.73016, respectively. The mAP 0.5:0.95 was 0.232, and the validation box loss, object loss, and class loss were 0.023141, 0.011917, and 0.010162, respectively. The learning rate in this epoch was 0.000232 for all three levels.

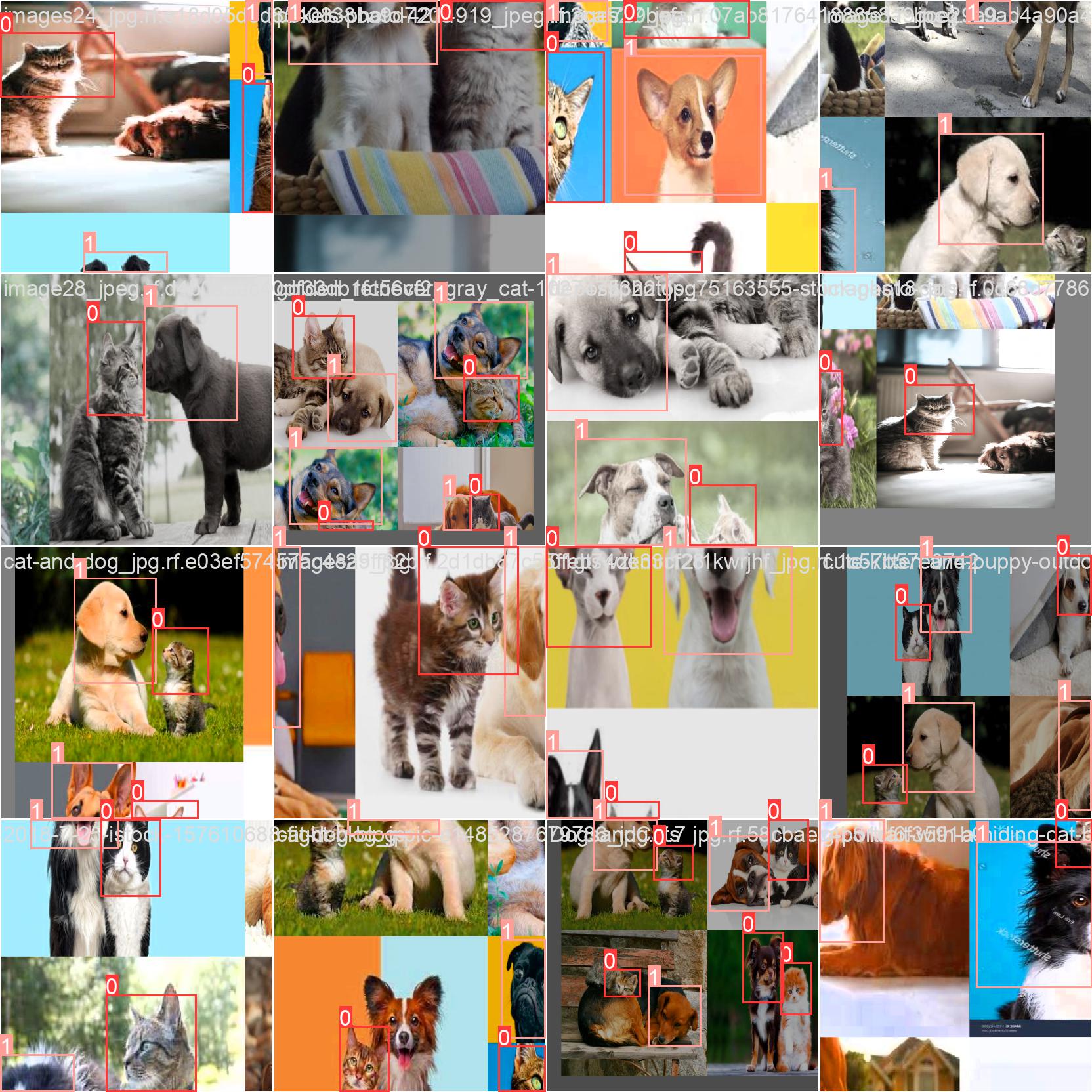
Overall, the model's box loss and object loss decreased over the epochs, indicating improvement in the model's ability to detect objects accurately. However, the class loss varied throughout the epochs, indicating that the model may have struggled to classify certain objects. The precision, recall, and mAP also fluctuated, indicating that the model's performance varied depending on the epoch. The mAP 0.5:0.95 was consistently low throughout the epochs, indicating that the model may have had difficulty detecting objects with high levels of precision.

**Visualization of all the metrics performance discussed above:**

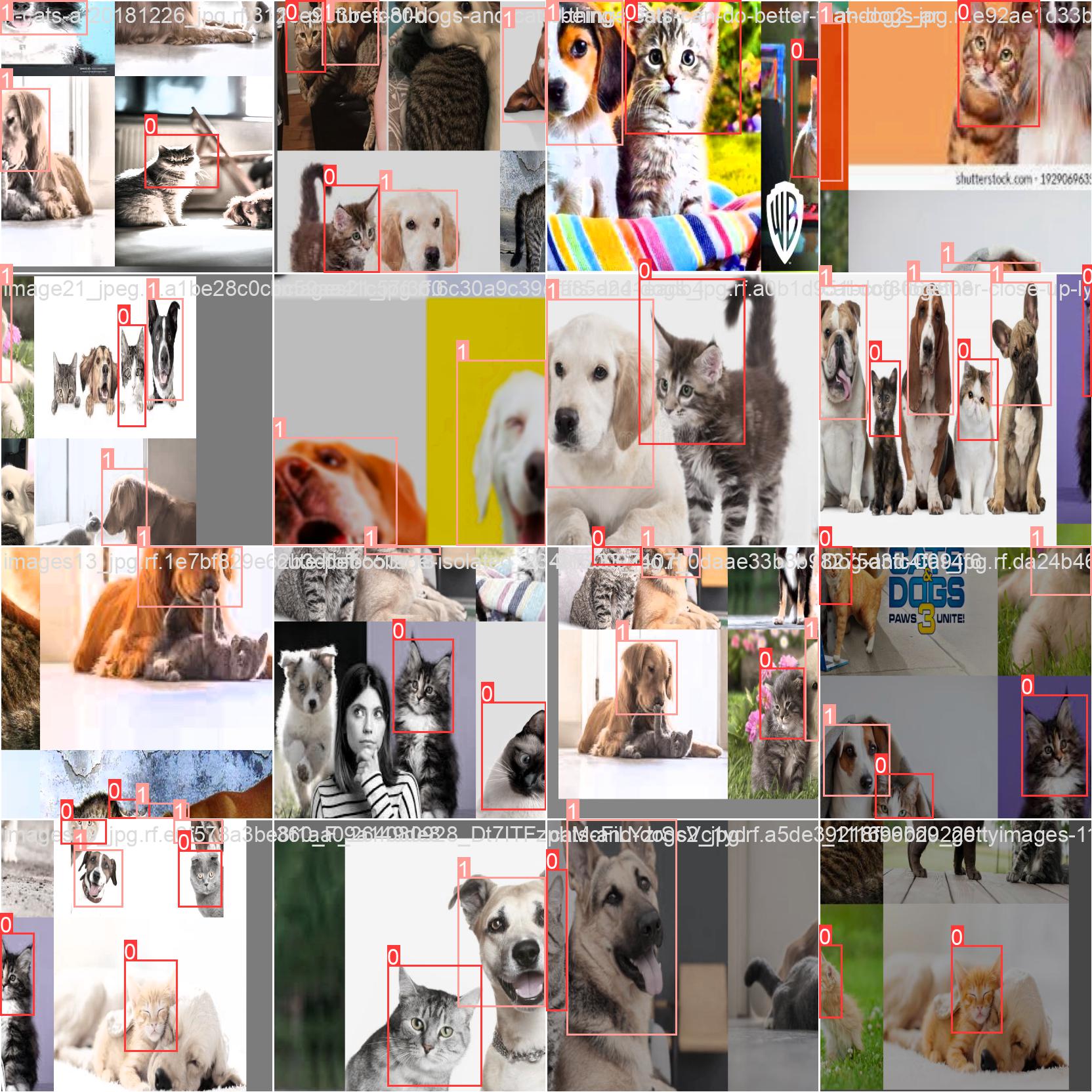




**Train batch 0:**



**Train batch 1:**



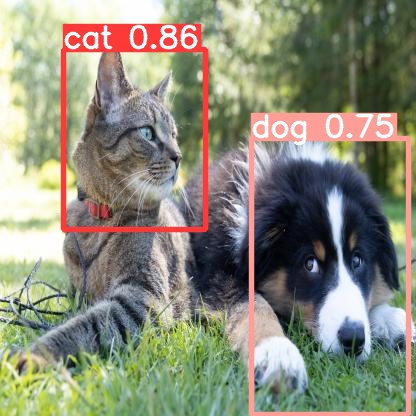
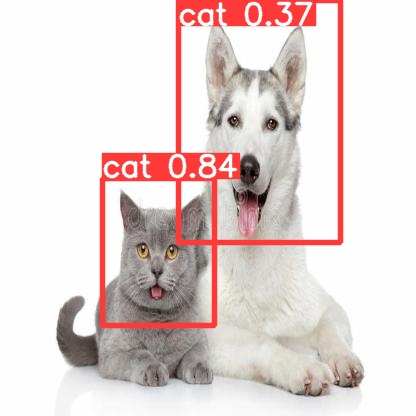
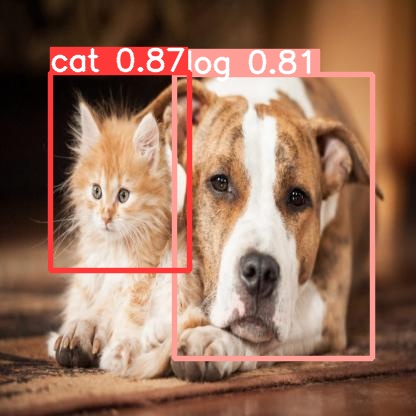
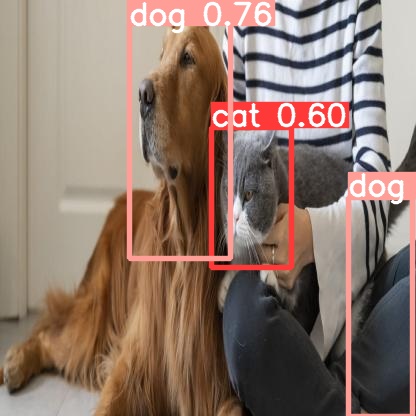
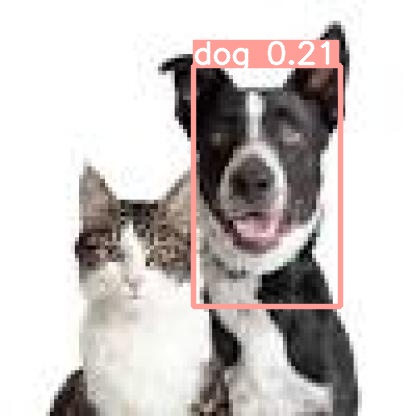
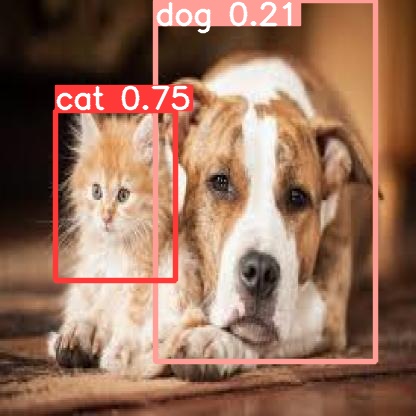
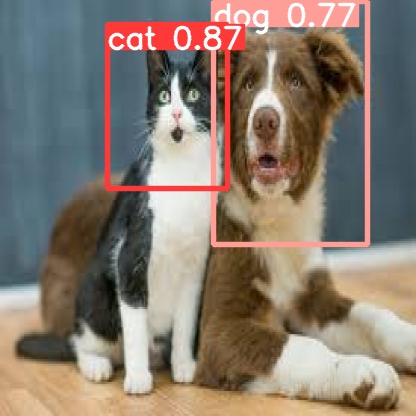
**Train batch 2:**



**Validation batch :**



**Model testing with test image:**

****

From the above images, we saw Yolov5 model accurately detects the test image classes such as cat and dog which is good in sound. Moreover, our images are insufficient, if we provide more images for training purposes, our model can perform very well as we expected.