

Yolov5 confusion matrix with background FP=1 and TN=0

#11194

New issue

🔒 Closed

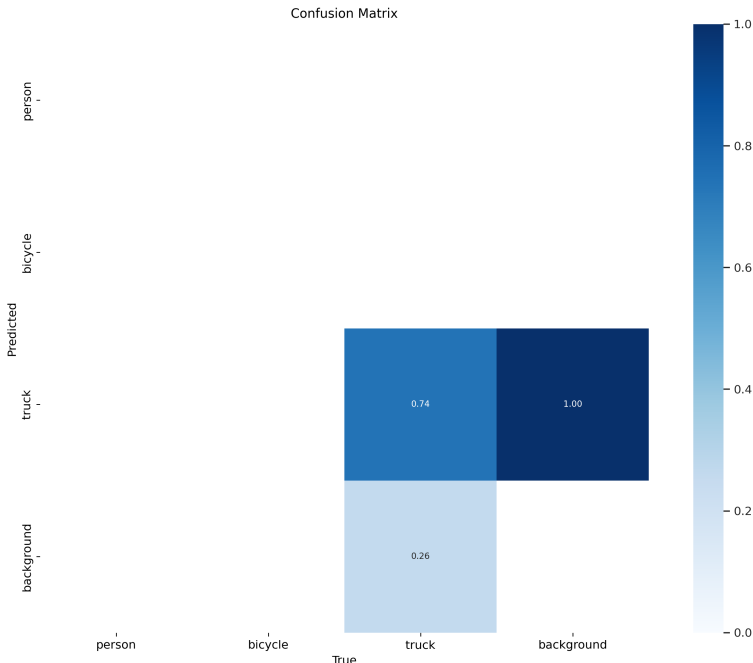
yiluny217 opened this issue on Mar 20, 2023 · 12 comments



yiluny217 commented on Mar 20, 2023 · edited ▾

Hello,

I was training a model to detect trucks in pictures and here is the result confusion matrix of my val data. Following the convention of reading a confusion matrix, I'll call TP for the upper left cell, FP for upper right cell, FN for lower left cell and TN for lower right cell. (please ignore the class 'bicycle' and 'person' because the original dataset only have trucks labeled but 'truck' was assigned a 'class=2' during the manual annotation)



For the 'background', there are 1 and 0. I searched online and found a lot of people are having the same issue, here are some examples:

[yolov5 issue 10365](#)
[yolov5 issue 1665](#)
[stackoverflow](#)

In [yolov5 issue 1665](#), I noticed [@glenn-jocher](#) gave a brief explanation that 'columns are normalized', but I'm still quite confused. My I get a more clear explanation about why it happens and is there a possible way to fix it?

Another thing bothering me is that actually I didn't have any annotation of background in my training data, so I guess that's why TN=0?



Assignees

No one assigned

Labels

Stale

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

5 participants



github-actions (bot) commented on Mar 20, 2023 ·
edited by glenn-jocher ▾

Contributor

👋 Hello [@yiluny217](#), thank you for your interest in YOLOv5 🚀! Please visit our [Tutorials](#) to get started, where you can find quickstart guides for simple tasks like [Custom Data Training](#) all the way to advanced concepts like [Hyperparameter Evolution](#).

If this is a 🐛 Bug Report, please provide a **minimum reproducible example** to help us debug it.

If this is a custom training ❓ Question, please provide as much information as possible, including dataset image examples and training logs, and verify you are following our [Tips for Best Training Results](#).

Requirements





[Python>=3.7.0](#) with all [requirements.txt](#) installed including [PyTorch>=1.7](#). To get started:

```
git clone https://github.com/ultralytics/yolov5 # clone
cd yolov5
pip install -r requirements.txt # install
```



Environments

YOLOv5 may be run in any of the following up-to-date verified environments (with all dependencies including [CUDA/CUDNN](#), [Python](#) and [PyTorch](#) preinstalled):

- **Notebooks** with free GPU:  [Run on Gradient](#)  [Open in Colab](#)
 [Open in Kaggle](#)
- **Google Cloud** Deep Learning VM. See [GCP Quickstart Guide](#)
- **Amazon** Deep Learning AML. See [AWS Quickstart Guide](#)
- **Docker Image**. See [Docker Quickstart Guide](#) 

Status



If this badge is green, all [YOLOv5 GitHub Actions](#) Continuous Integration (CI) tests are currently passing. CI tests verify correct operation of YOLOv5 [training](#), [validation](#), [inference](#), [export](#) and [benchmarks](#) on MacOS, Windows, and Ubuntu every 24 hours and on every commit.

Introducing YOLOv8 🚀

We're excited to announce the launch of our latest state-of-the-art (SOTA) object detection model for 2023 - [YOLOv8](#) 🚀!

Designed to be fast, accurate, and easy to use, YOLOv8 is an ideal choice for a wide range of object detection, image segmentation and image classification tasks. With YOLOv8, you'll be able to quickly and accurately detect objects in real-time, streamline your workflows, and achieve new levels of accuracy in your projects.

Check out our [YOLOv8 Docs](#) for details and get started with:

```
pip install ultralytics
```





github-actions (bot) commented on Apr 19, 2023 •

edited by glenn-jocher ▾

Contributor

👋 Hello, this issue has been automatically marked as stale because it has not had recent activity. Please note it will be closed if no further activity occurs.

Access additional [YOLOv5](#) 🚀 resources:

- Wiki – <https://docs.ultralytics.com/yolov5>
- Tutorials – <https://docs.ultralytics.com/yolov5>
- Docs – <https://docs.ultralytics.com>

Access additional [Ultralytics](#) ⚡ resources:

- Ultralytics HUB – <https://ultralytics.com/hub>
- Vision API – <https://ultralytics.com/yolov5>
- About Us – <https://ultralytics.com/about>
- Join Our Team – <https://ultralytics.com/work>
- Contact Us – <https://ultralytics.com/contact>

Feel free to inform us of any other **issues** you discover or **feature requests** that come to mind in the future. Pull Requests (PRs) are also always welcomed!

Thank you for your contributions to YOLOv5 🚀 and Vision AI ⭐!



🔖 github-actions (bot) added `Stale` and removed `Stale` labels on Apr 19, 2023



github-actions (bot) commented on May 23, 2023

Contributor

👋 Hello there! We wanted to give you a friendly reminder that this issue has not had any recent activity and may be closed soon, but don't worry - you can always reopen it if needed. If you still have any questions or concerns, please feel free to let us know how we can help.

For additional resources and information, please see the links below:

- Docs: <https://docs.ultralytics.com>
- HUB: <https://hub.ultralytics.com>
- Community: <https://community.ultralytics.com>

Feel free to inform us of any other **issues** you discover or **feature requests** that come to mind in the future. Pull Requests (PRs) are also always welcomed!

Thank you for your contributions to YOLO 🚀 and Vision AI ⭐



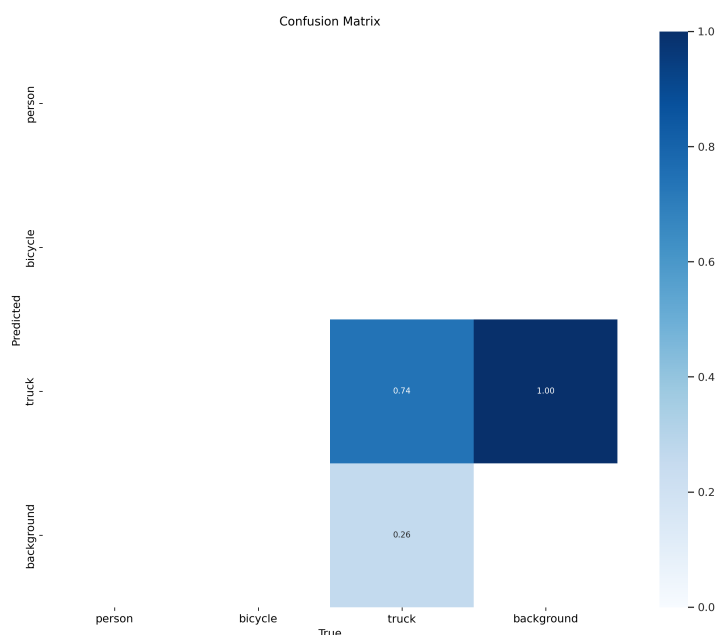
🔖 github-actions (bot) added the `Stale` label on May 23, 2023



jbezovsek commented on May 27, 2023

Hello,

I was training a model to detect trucks in pictures and here is the result confusion matrix of my val data. Following the convention of reading a confusion matrix, I'll call TP for the upper left cell, FP for upper right cell, FN for lower left cell and TN for lower right cell. (please ignore the class 'bicycle' and 'person' because the original dataset only have trucks labeled but 'truck' was assigned a 'class=2' during the manual annotation)



For the 'background', there are 1 and 0. I searched online and found a lot of people are having the same issue, here are some examples: [yolov5 issue 10365](#) [yolov5 issue 1665 stackoverflow](#) In [yolov5 issue 1665](#), I noticed [@glenn-jocher](#) gave a brief explanation that 'columns are normalized', but I'm still quite confused. My I get a more clear explanation about why it happens and is there a possible way to fix it?

Another thing bothering me is that actually I didn't have any annotation of background in my training data, so I guess that's why $TN=0$?

I have the same issue of interpreting this kind of results, however my interpretation would be that the columns do not depend on each other as you could assume in 2x2 simple confusion matrix. The 2x2 confusion matrix that I have in mind should ideally look like $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. Because the columns are normalized, the sum of the columns has to be 1, but the sum of the rows can be over 1. In your case I would say that if the actual object was truck, the model predicted truck in 74% and in 26% predicted background. I am still a little confused about the background, because [@glenn-jocher](#) here: [#1665 \(comment\)](#) said, that the background is not predicted, so this could be the reason for the results in background column?



glenn-jocher commented on May 27, 2023

Member


[@jbezovsek](#) thank you for sharing your concerns about interpreting the confusion matrix for YOLOv5. It can be confusing to understand the results of the matrix, especially when dealing with single-class detection and background.

You are correct that the columns of this matrix do not necessarily depend on each other, as in a simple 2x2 confusion matrix. In the case of YOLOv5, the columns represent the predicted classes, and they are normalized. As a result, the sum of each of the columns would be equal to 1.

Regarding the background class, according to the YOLOv5 developers, it is not predicted by the model. Therefore, it is possible that the TN value being 0 is a result of not having any true negative samples for the background class in the validation set.

Once again, thank you for your question, and please let us know if you have any further concerns about YOLOv5 or vision AI in general.



 **github-actions** (bot) removed the `Stale` label on May 28, 2023



ilhamalvindo commented on May 31, 2023 • edited ▼

Hello [@glenn-jocher](#), is that mean for single class that has $FP=1$ and $TN=0$ on background class is because we don't have any samples of background on validation set. Not because our model is wrong, right?

Because i have the same issue above, my $FP=1$ and $TN=0$ for single class label



glenn-jocher commented on May 31, 2023

Member

Hello [@ilhamalvindo](#), thank you for reaching out with your question regarding YOLOv5's confusion matrix and understanding the results.

You are correct. In a single-class detection with background, if the TN value is 0, it likely means that there were no true negative samples for the background class in the validation set. Therefore, it does not necessarily imply that your model is wrong. Similarly, if the FP value is 1, it could mean that there was only one false positive detected as a background.

Please let me know if you have any further questions or concerns.



ilhamalvindo commented on May 31, 2023

Thank you for the explanation! [@glenn-jocher](#)



glenn-jocher commented on May 31, 2023

Member

[@ilhamalvindo](#) you're welcome, happy to be of help! If you have any further questions or issues with YOLOv5, please don't hesitate to ask.



github-actions (bot) commented on Jul 1, 2023

Contributor

👋 Hello there! We wanted to give you a friendly reminder that this issue has not had any recent activity and may be closed soon, but don't worry - you can always reopen it if needed. If you still have any questions or concerns, please feel free to let us know how we can help.

For additional resources and information, please see the links below:

- Docs: <https://docs.ultralytics.com>
- HUB: <https://hub.ultralytics.com>

- **Community:** <https://community.ultralytics.com>

Feel free to inform us of any other **issues** you discover or **feature requests** that come to mind in the future. Pull Requests (PRs) are also always welcomed!

Thank you for your contributions to YOLO 🚀 and Vision AI ⭐



🔖 **github-actions** (bot) added the **Stale** label on Jul 1, 2023

🕒 **github-actions** (bot) closed this as **not planned** on Jul 12, 2023



RyanTNN commented on Nov 10, 2023



Hello [@glenn-jocher](#) . I have some questions. I quit not understanding the background FP and the background FN. This confusion matrix shows background FP 0.77 and background FN 0.22.

1. What is exactly the meaning of background FP and FN?
2. Does it affect on the prediction? Why does it affect on the prediction or why does it not affect on the prediction?
3. As you know, changing conf 0.25 or 0.9 that only changes the object accuracy but does not change the background FP or FN. why?
4. how can I reduce background FP and FN?

Thank you!



glenn-jocher commented on Nov 10, 2023

Member

[@RyanTNN](#) hi there! It seems like your link to the confusion matrix image is not accessible. However, I will still address your questions based on the information provided.

1. The "background FP" represents the false positive rate for the background class, i.e., the rate at which the model incorrectly predicts the presence of the background class when it's not actually there. On the other hand, "background FN" stands for the false negative rate for the background class, i.e., the rate at which the model fails to detect the background class when it is present in the image.
2. The background FP and FN can affect the overall performance of the model, especially if the background class is being misclassified frequently, which might lead to incorrect predictions for other classes as well. However, in some scenarios, particularly in single-class detection tasks, the impact might be minimal depending on the specific use case.
3. Changing the confidence threshold (conf) primarily affects the object accuracy as it determines the minimum confidence score required for an object to be considered as detected. It might not directly impact the background FP or FN if the background class is not being considered in the confidence threshold settings.
4. To reduce background FP and FN, you may try various techniques such as refining the training data to include more diverse backgrounds, enhancing the model architecture, adjusting training hyperparameters, and possibly introducing data augmentation to expose the model to a wider variety of background scenarios.

Feel free to provide additional details or share the confusion matrix image for a more precise analysis or assistance.

