



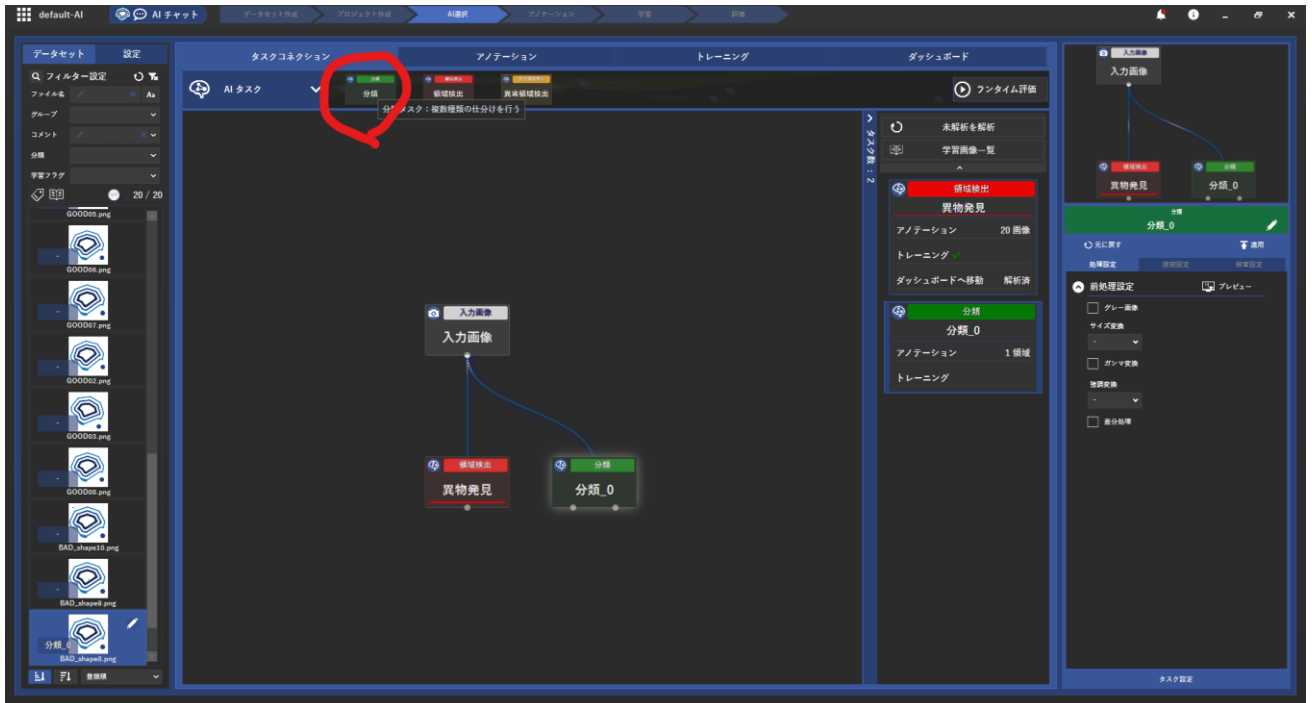
## 1 分類

領域検出についてはチュートリアル 1 でご紹介したので、もう一つの AI タスクである「分類」を試してみたいと思います。

### 1.1 タスクコネクションに追加

「タスクコネクション」タブをクリックし、タスクコネクションの画面にします。

上の AI タスクウィンドウ内の「分類」という緑の箱を選びます。



すると、先ほどの領域検出の隣に新しく分類タスクができています。

### 1.2 分類タスクのアノテーション

セグメンテーションと違って、分類タスクのアノテーションは画像単位で行うので非常に簡単です。先ほどの分類タスクの箱をクリックするか、「アノテーション」タブをクリックし、アノテーション画面に移ります。

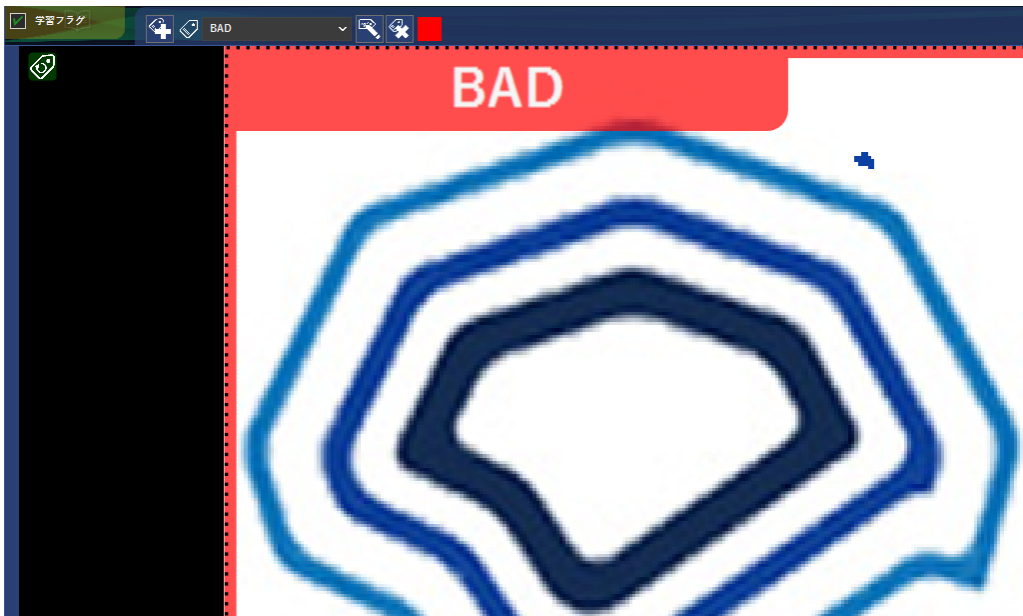
分類タスクのアノテーションでは、「クラス」という分類に画像を紐づけることで行うため、分かりやすいクラス名をつけます。

左上の「クラス名編集」アイコンをクリックし、欠陥のある画像には BAD、欠陥のない画像には GOOD という名前をつけることにしましょう。



クラス名のドロップダウンを切り替えて、GOOD の名称も変えましょう。  
分類タスクでは最低2つの分類が必要ですが、もっと数を増やすことも可能です。

画像をクラスに紐づけるには、クラス名のドロップダウンを指定し、「不明」と書いてある画像の枠をクリックします。





領域検出 (セグメンテーション) のときと同じように、GOOD01.png ~ GOOD07.png にアノテーションしてみましょう。

クラス名を指定後、学習フラグも忘れずにチェックします。

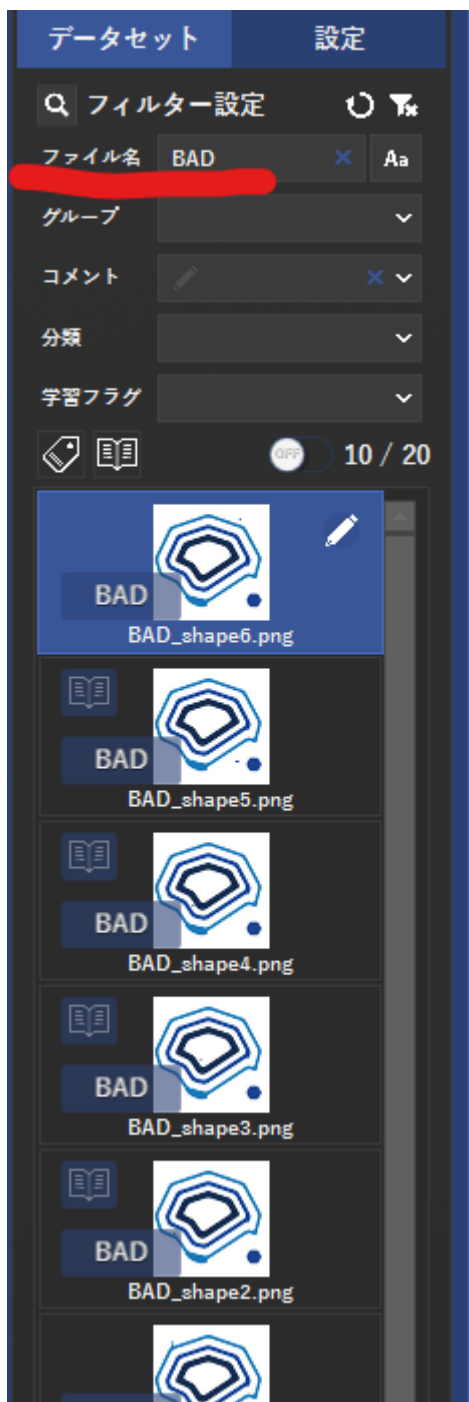
今回は、アノテーションを行わない GOOD08.png ~ GOOD10.png にも最初から正解のクラスを設定しておきます。

セグメンテーションのとき同様、画像を切り替えながらクラスを適用してみましょう。

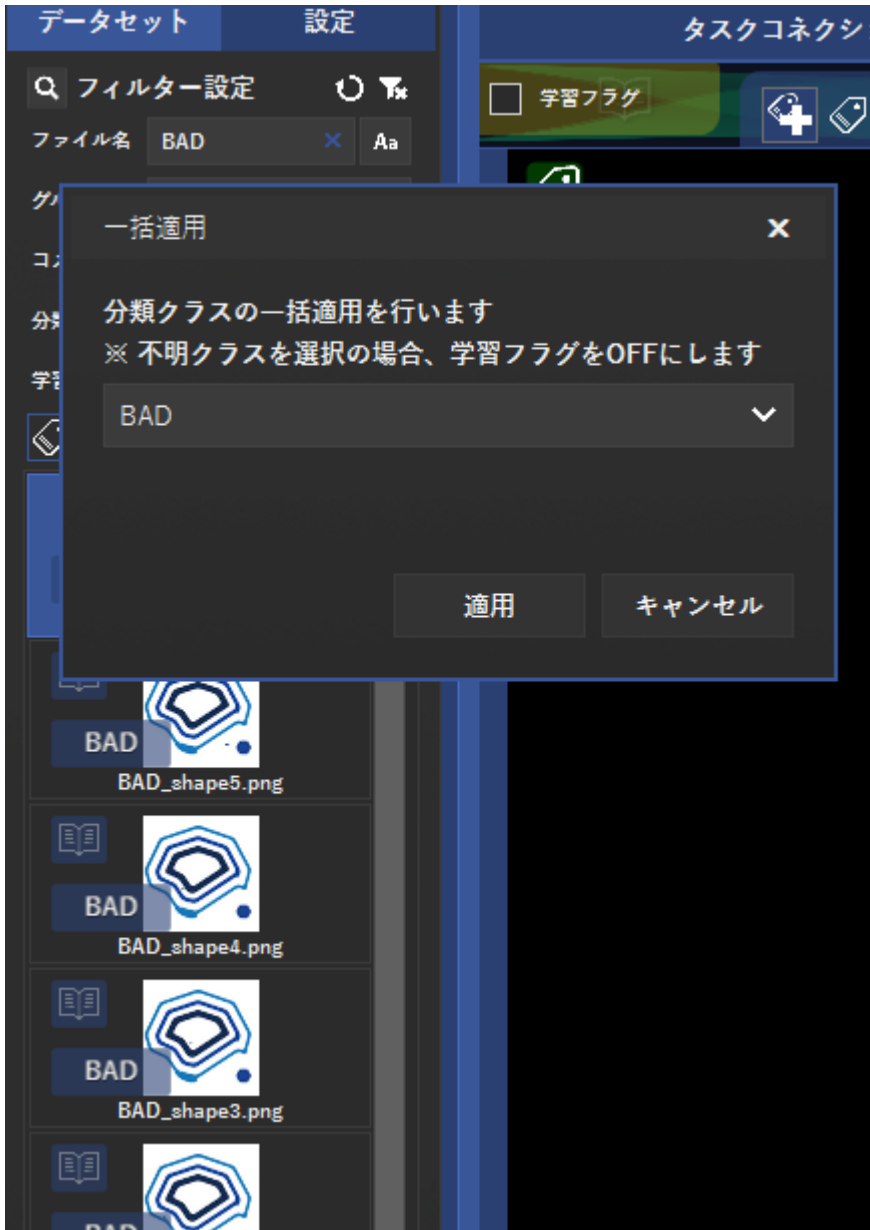
同様に、BAD\_shape1.png ~ BAD\_shape7.png をアノテーションします。

分類タスクのアノテーションの場合は、非常に単純なので、左側のデータプレビュー画面で一度にやってしまうのがいいでしょう。

それには、まずはファイル名のフィルタに「BAD」と入力すると BAD が含まれるファイルのみが表示されるので適用します。



そこで、「一括分類クラス適用」というアイコンをクリックしてみます。



すると、一括でクラス適用するポップアップが出てきました。  
BAD を選び、適用をクリックすると出来ました！

そして、BAD\_shape1.png ～ BAD\_shape7.png をクリックし、右クリックして「学習フラグ適用」  
を選べばアノテーションは完了です。



### 1.3 トレーニング

分類タスクのトレーニングは、セグメンテーションのときとほぼ変わりません。

「トレーニング」タブをクリックし、学習させる準備をしましょう。

今回も「全てを OFF」にして、1000 回の学習を試してみることにします。



画面上部の▶をクリックすると学習が進みます。

画像単位（領域検出ではピクセル単位）でしか学習を行わないので、非常に早いです。2分程度で終わりました。（GPUは3070Tiを利用）

### 1.4 結果



惜しい！

1つ推論が間違っていますね。

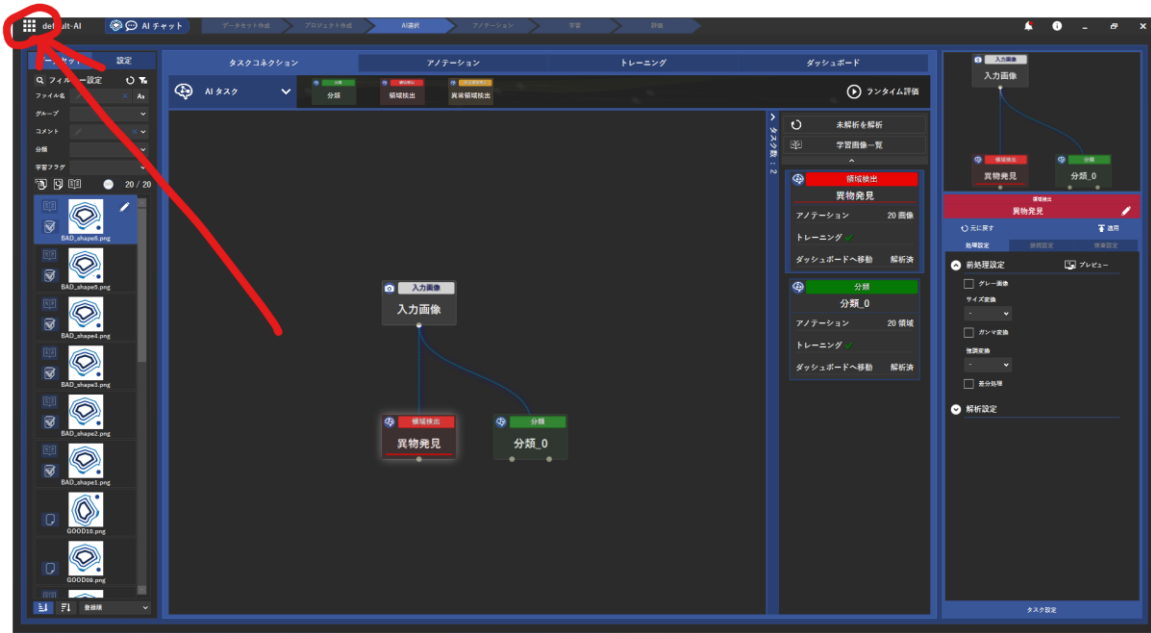
分類タスクは、おおざっぱに画像を見るため、細かい欠陥を見つけることが苦手なことが多いです。ですが、今回は演習ということでもうちょっと頑張ってみることにします。

画像を追加しましょう。

[こちら](#)から画像サンプルをダウンロードして、解凍後データセットに加えてください。

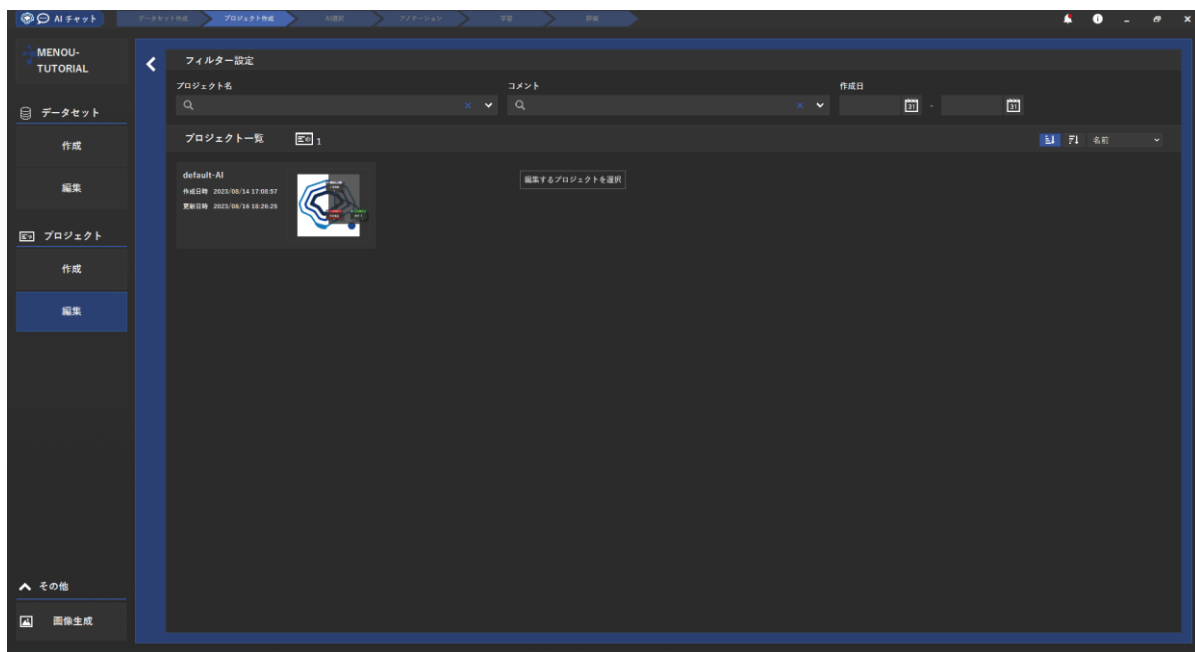
最初の 20 枚同様の画像（良品画像 10 枚、不具合画像 10 枚）をさらに 20 枚加えることにします。

## 1.5 画像の追加



一番左上のメニューをクリックします。

すると、プロジェクト一覧が表示されますので、データセットの編集を選びます。



現在のデータセットに追加する画像を選び、ドラッグアンドドロップして追加します。

「プロジェクト 編集」を選び、現在のプロジェクトをクリックして再度 AI 設計画面に戻ります。

すると、アノテーションされていない画像が増えていることが分かります。

## 1.6 追加学習

分類タスクのアノテーション作業を追加した画像で行います。

今回は説明を端折りますが、追加の画像をすべてアノテーションしたことにしましょう。



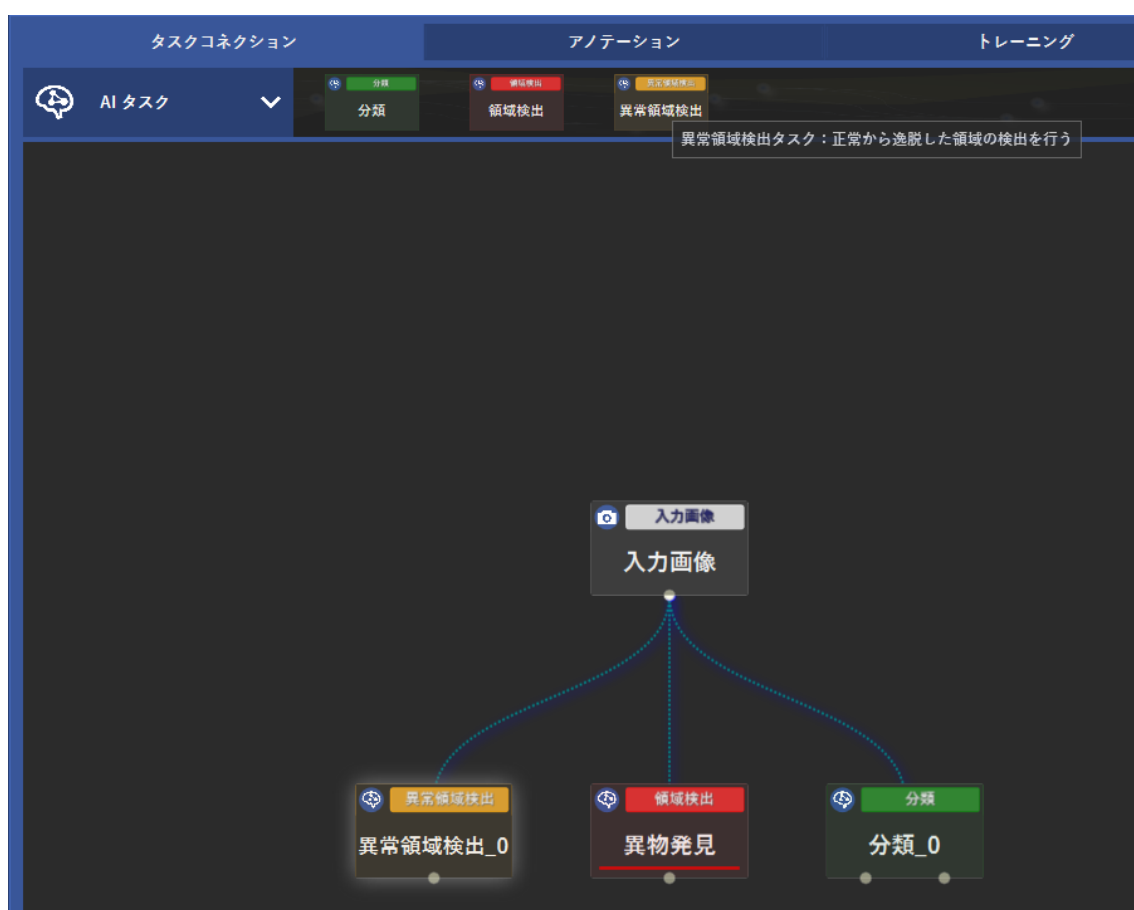




## 2 異常領域検出（教師なし学習）

MENOU-TE では3つの AI タスクを利用できますが、最後に紹介するのが「異常領域検出（教師なし学習）」です。教師なし学習と聞くと、アノテーションがいらぬような印象を持ちますが、別称：良品学習とも呼ばれているように、良いものだけは教えてあげる必要があります。良品を学習させれば、異常品を見つける、というメリットがあります。その簡単さの代わりに、検出力はどうしても落ちます。実際に見ていきましょう。

[こちら](#)の画像をデータセットに追加して用います。



タスクコネクションのタブをクリックし、タスクコネクションを表示します。そこで、AI タスク内の「異常領域検出」をクリックすると、このように、もう一つ黄色い箱ができました。この黄色い箱が「異常領域検出」のタスクを表しています。

### 1.7 アノテーション

異常領域検出のアノテーションでは、良品のみを学習させます。



ファイル名で分けられていますので、左側のデータセットウィンドウを使うと便利です。

GOOD で始まるファイル名だけを表示させ

→一括検出状態適用 「検出なし」

→一括学習フラグ適用 「学習に使用する」

とします。

また、bad で始まるファイル名だけを表示させ (ファイル名フィルタは大文字小文字の区別を切り替えることができます)、検出状態のみ「検出あり」とします。

このように、アノテーションは非常に簡単です。

## 1.8 トレーニング

アノテーションが出来たので学習へと進めます。



これまでの AI タスク同様の設定画面になっています。

今回は「全てを OFF」にし、「データ拡張回数」を 5 回にしましょう。

異常領域検出では最低 30 枚の良品画像がないと、トレーニングが行えません。

この設定は、MENOU-TE 内で良品画像にシンプルな加工を加え教師画像を増量することをを行う回数を指示するパラメータです。今回は良品が少ないため 5 回の拡張を指定します。

そして、実行を押すと。。。今回は非常に早いですね。

すぐに結果が得られました。

## 1.9 結果の確認

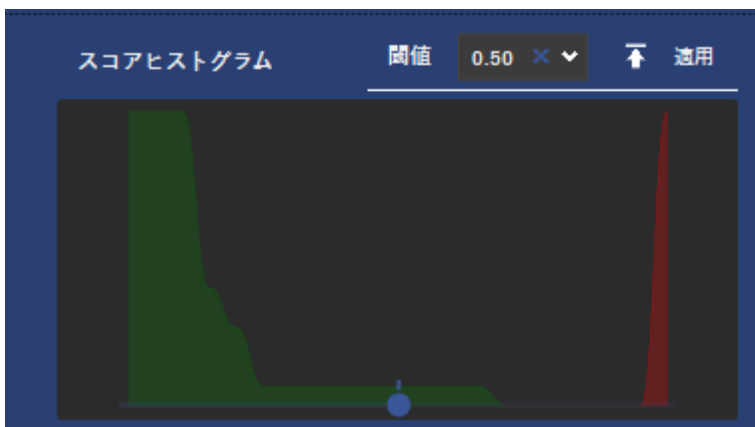


良品学習の結果画面も、見慣れてきた結果画面に近いです。

このように、コンフュージョン・マトリクスを見るとちょっと惜しく、1つの画像で良品画像に不具合を検出しています。

### 1.10 修正

このスコアヒストグラムを見ると簡単に閾値の調整で精度を高められそうですね。

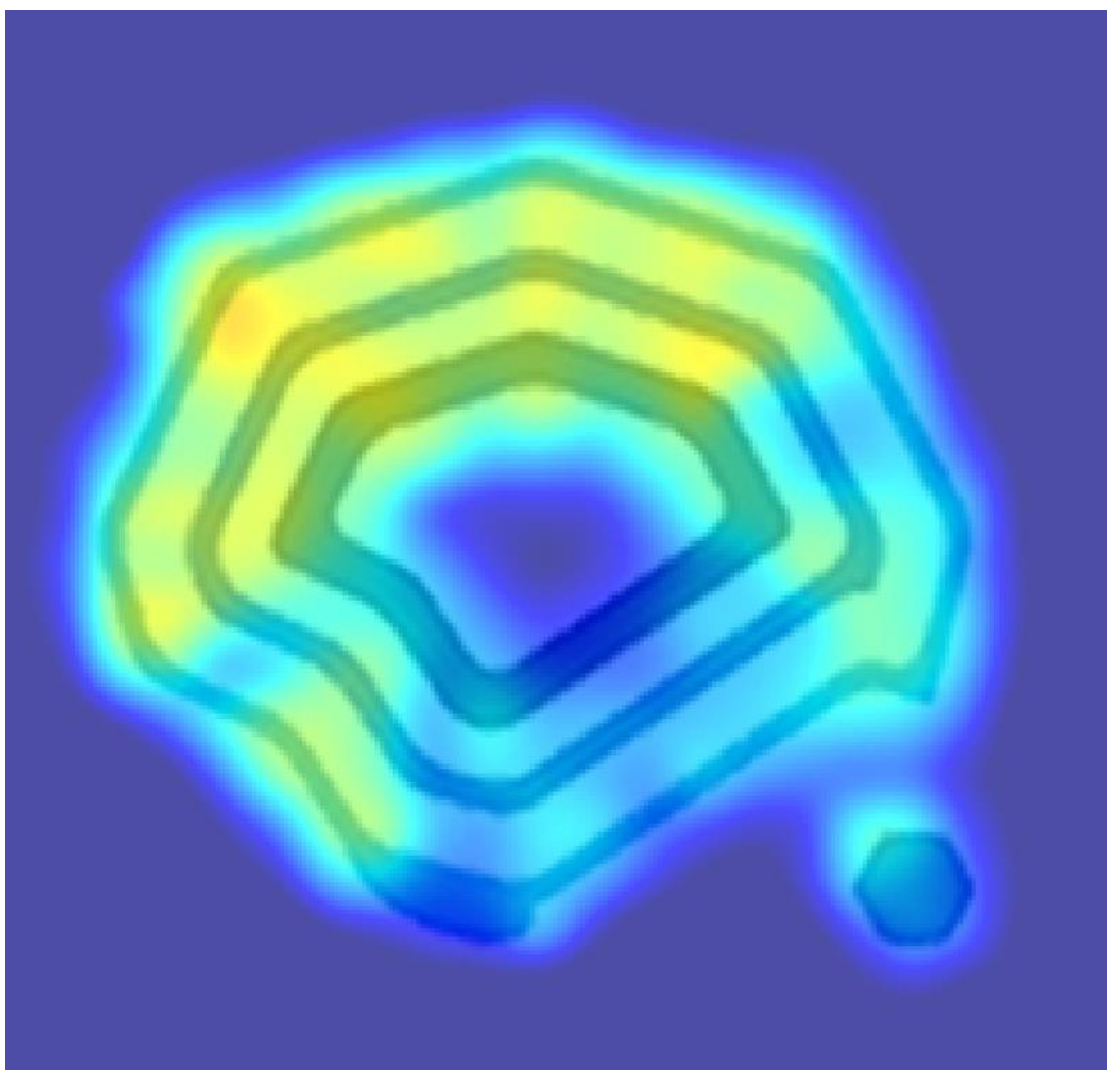


一応、念のためにどのように間違えているのかを確認します。



		実行結果			正解率
		検出なし	検出あり	合計	
正解	検出なし	29	1	30	96.6%
	検出あり	0	15	15	100.0%
	不明	0	10	10	-
	合計	29	26	55	97.7%

このコンフュージョン・マトリクスの数字をクリックすると簡単にこの画像を見つけることができます。



なんとなく全体的に AI が怪しんでいるようですね。



AI の特性上、厳密に異常を見つけることは苦手なので、教師なし学習は「怪しい」画像を見つける程度のものだと考えておいた方がいいでしょう。特に今回は画像枚数も少なかったので、今回の演習はここで終了とします。

教師なし学習を効果的に使うには、他のタスクと組み合わせるのが有効なので、次は複数のタスクを組み合わせる方法をご紹介します。