



SIGNATE TECHNOPRO

2nd Place Solution

@213tubo



はじめに

今日の発表は、**ある程度分析コンペに参加したことがある人向け**に作成しています

- CV、CNN、LB、疑似ラベル...等の言葉、手法の説明は行いません

自己紹介

- 情報系学部3年
- SIGNATE Expert
- Kaggle Competition Master



[@213tubo](#)

今回の結果

Public 3rd(AUC1.0)



Private 2nd(AUC1.0)

今回使用したコードは発表会が終わり次第 githubで
公開します。
https://github.com/tubo213/signate_technopro

順位	チーム名 / ユーザ名		暫定評価	最終評価 ▼	投稿 件数	投稿日時
1	YK		1.0000000	1.0000000	29	2021-12-17 15:16:02
2	213tubo		1.0000000	1.0000000	12	2021-12-26 05:55:02
3	youtoo		1.0000000	0.9932886	3	2021-12-26 23:04:05
4	kzrn		1.0000000	0.9894334	4	2021-12-13 20:04:05
5	mugencai		0.9862545	0.9659435	35	2022-01-05 21:27:02
6	KouFu		0.9470199	0.9597315	9	2021-12-18 17:44:05
7	Fukumoto		0.9566487	0.9461176	5	2022-01-05 17:08:05
8	arukisumaho		0.9071124	0.9099223	23	2022-01-05 03:00:04
9	rin12		0.7345624	0.7423918	1	2021-12-15 12:51:04
10	jun		0.6603657	0.7012015	6	2021-12-12 03:15:04



発表内容

主に解法に至った**経緯、考えたこと**を発表します

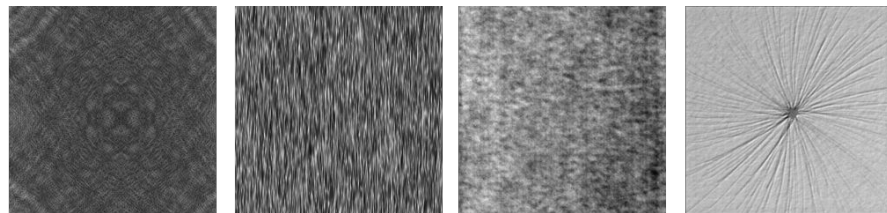
- タスクの理解
- 解法
- 実務への応用

コンペ概要

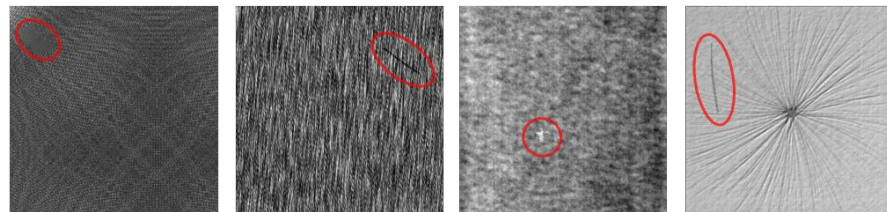
画像から製品が欠陥かを予測する二値分類

- 評価指標: ROC-AUC
- データ
 - train: 4600 (正常=4000, 欠陥=600)
 - test: 2300

正常



欠陥



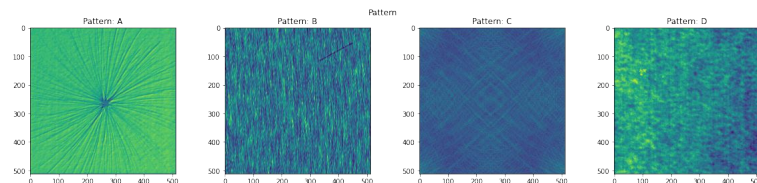
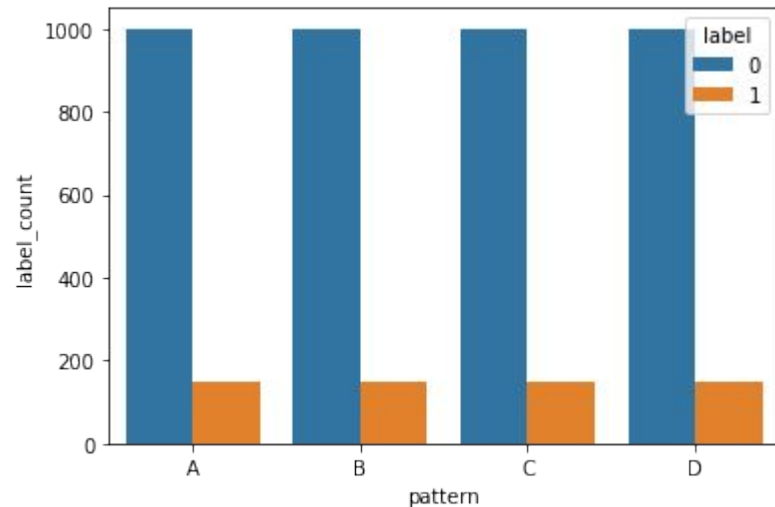
<https://static.signate.jp/competitions/574/SBhxrDdC1rG0t3U1lPLgzs48rF1zloQJndDdt6iY.png>

タスクの理解

画像には複数のパターンがある

- パターンA (正常=1000、欠陥=150)
- パターンB (正常=1000、欠陥=150)
- パターンC (正常=1000、欠陥=150)
- パターンD (正常=1000、欠陥=150)

欠損の比率(13%)はtestでも同じであることが仮定できそう

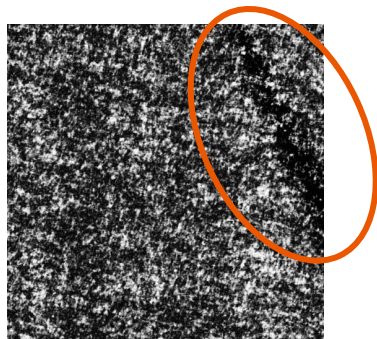


タスクの理解

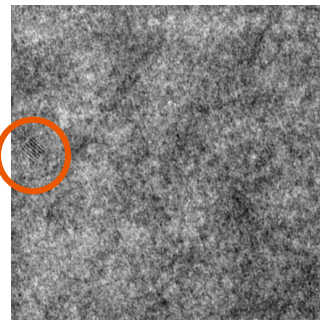
Test setはTrain setに**含まれないパターン、欠陥のみ**で構成されている



ラベルの無い画像の欠損を検知しなければならない



新パターン(便宜上Xとする)



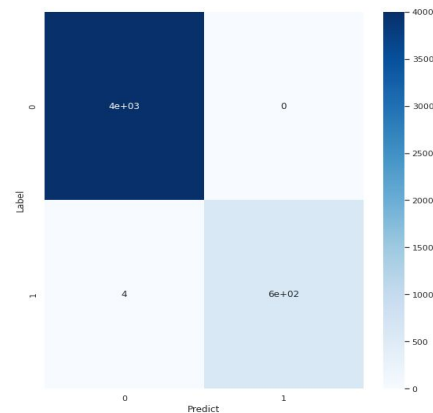
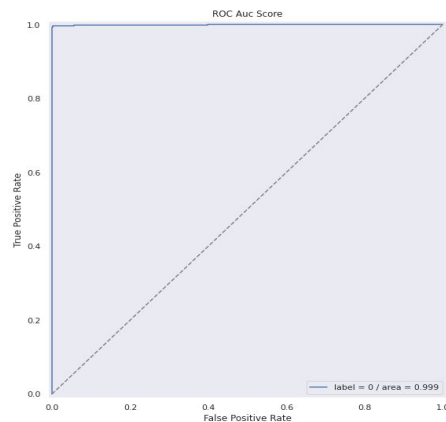
新パターン(便宜上Yとする)

タスクの理解

欠陥の検知自体は**極めて簡単**

パラメータ調整をしていない ResNet18をTrain setで学習させると...

Startified
KFold
0.99



解法

分析結果

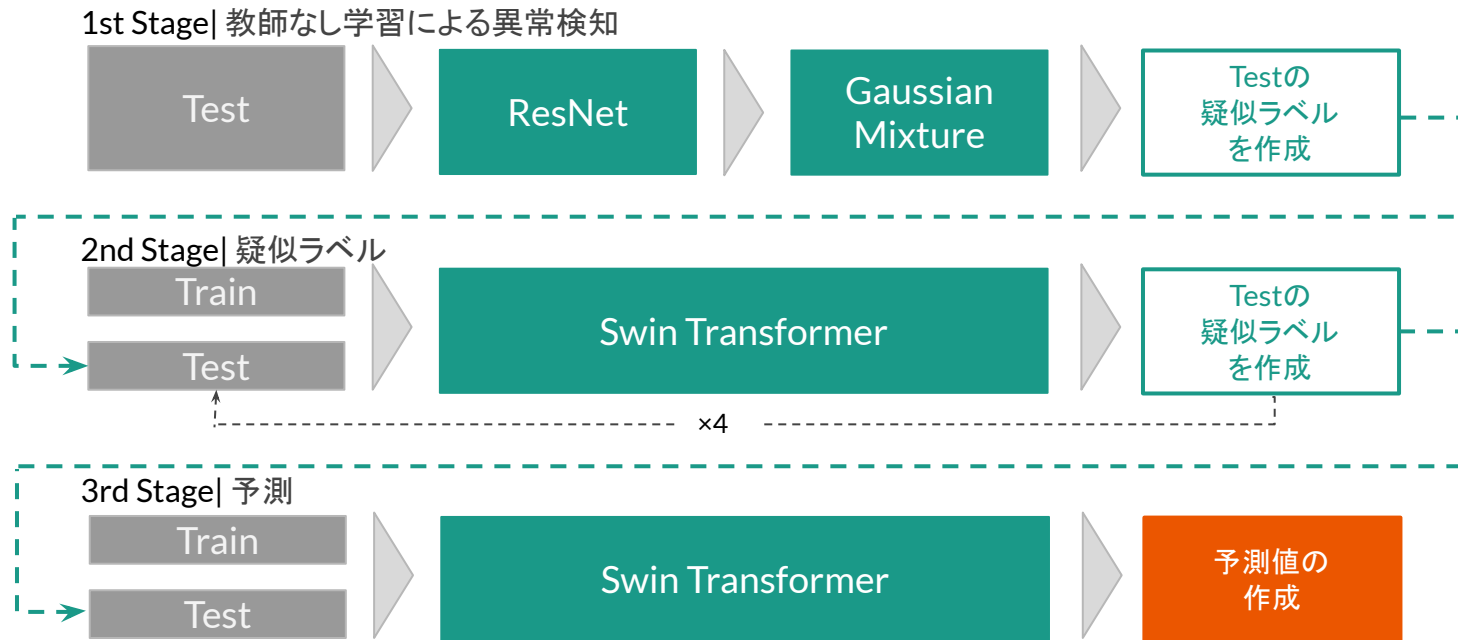
- 実質的に
ラベルが与えられていない
 - 手元での適切な交差検証は難しい
- ラベルさえあれば
欠陥を検知するのは極めて簡単



アプローチ

- 学習方法
 1. 教師なし学習による異常検知
 2. 1の結果を用いて教師有り学習を行い疑似ラベルを繰り返す
- 検証方法
 - Publicのスコアを見て検証
- 問題点
 - 1、2の疑似ラベルには一定数の誤りが含まれる

解法全体像



1st stage| 教師なし学習による異常検知

各画像に対して異常スコアを算出し、下位10%を欠陥とする

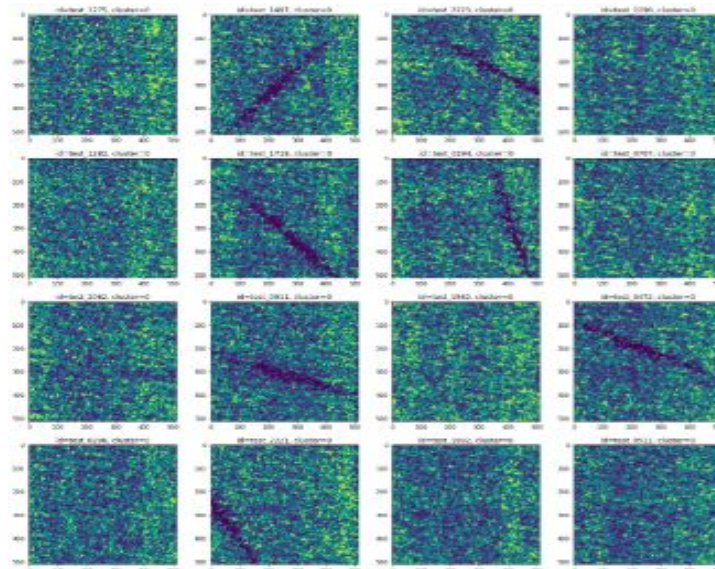
- 試した異常検知モデル
 - Auto Encoder
 - **Gaussian Mixture(採用)**



1st stage| 教師なし学習による異常検知

AutoEncoder

- パラメータ
 - Encoder
 - 2層
 - Decoder
 - 2層
- 異常スコア
 - 平均二乗誤差

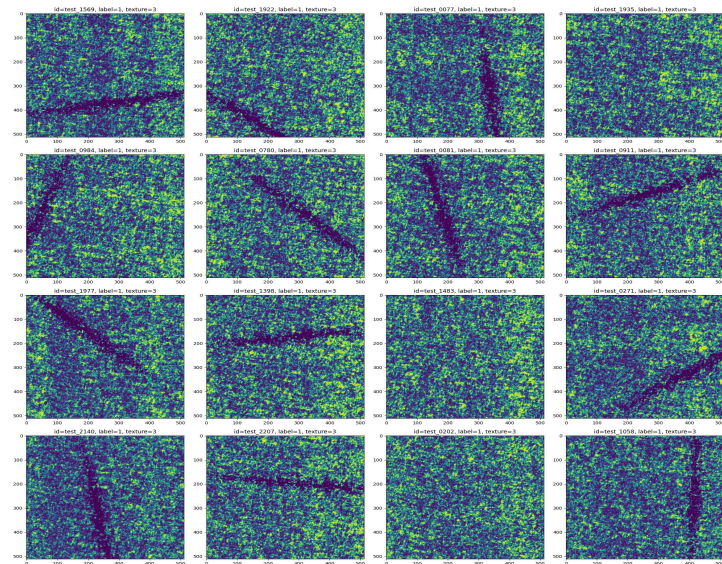


異常判定された 16サンプル

1st stage| 教師なし学習による異常検知

Gaussian Mixture

- 特徴量
 - ResNetの中間層出力を次元削減したもの
- パラメータ
 - `n_components: 1`
- 異常スコア
 - 負の対数尤度



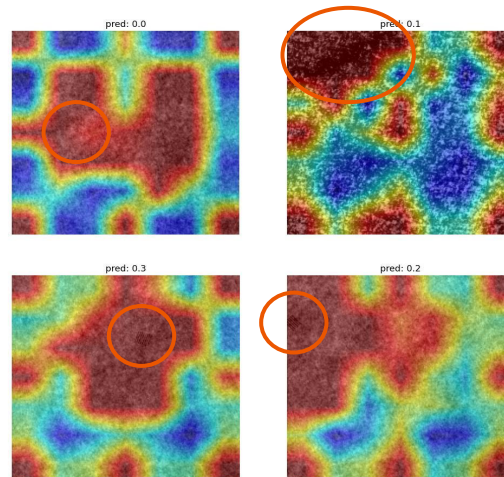
異常判定された 16サンプル

2nd stage| 疑似ラベル

予測確率の上位[5, 10, 12, 13]%を欠陥として順に疑似ラベリングを行う

- Model
 - Swin Transformer
- Optimizer
 - AdamW
- Sheduler
 - CosineAnnealingWarmRestarts
- Augmentation
 - RandomHorizontalFlip
 - RandomVerticalFlip
 - Random90Rotate
 - Mixup
- Loss
 - SCE

Attentionが
効きそう！



欠陥部分以外を
よく見ている

Grad CAMによるモデル解釈

問題点への対処

疑似ラベルには間違いがある



Noisy labelであることを考慮した損失関数を採用

Public: 0.997 -> 1.000

Private: 0.995 -> 1.000



問題点への対処

Symmetric Cross Entropy Loss(SCE)

正解ラベルが真の分布ではなく、**一定の予測値が真の分布**であることを考慮した損失

$$CE = - \sum_{k=1}^K q(k|\mathbf{x}) \log p(k|\mathbf{x})$$

$$RCE = - \sum_{k=1}^K p(k|\mathbf{x}) \log q(k|\mathbf{x})$$

$$SCE = \alpha CE + \beta RCE$$



実務への応用

- 実行速度
 - RTX3070を用いた場合、30秒で**2300枚の推論が可能**
 - 最適化を行えば更に高速化可能
- 実行環境
 - Dockerを使用してモデルを作成したため、**現在のコードをそのまま実務に適応可能**
- 汎用性
 - 手動で設定したパラメータが無いため**どんなパターン、欠陥にも対応可能**



実務への応用

Test setのみで学習を行った結果..

Public Private共に**AUC 1.0を達成**

 **アノテーションコスト0※**でモデル開発可能

※評価の為のアノテーションは必要



振り返り

- **良かったこと**
 - ラベルが無い場合の取り組みを経験できた
 - 分析結果を上手くモデルに組み込めた
- **上手くいかなかったこと**
 - 画像処理系のアプローチ
 - どうしても閾値を決定しなければならず、実務への応用を考えて採用を見送った
 - 深層学習より精度が低かった



ありがとうございました！