

# 08.20 교수님 연구 미팅

■ 태그	
■ 날짜	@2025년 8월 20일
■ 상태	시작 전
■ 실험 제목	

## 지난 피드백

- spurious forgetting 직접 확인 방법
  - 기존 연구들 확인

## Findings

- 학습을 할 때 같은 task라도 class 별로 학습 수렴 속도가 다름
- Task=1부터 엄청나게 빠른 수렴 속도를 보임
  - 바로 높은 성능이 나타남
  - 이게 클래스 디펜던트 한지 확인 필요

## Findings을 위한 실험

### 1. Spurious forgetting 확인

### 2. class 별로 학습 성능 차이

- + Forgetting 심한 클래스

### 3. Forward transfer

1. Task=0 학습 후 Task=1에서 바로 성능이 좋아지는 것
2. Task=0의 학습 성능이 이후의 forgetting에 영향을 주는 정도
3. Task=0의 성능이 수렴 했음에도 계속 학습했을 때 forgetting과 forward transfer 차이

### 1. 같은 Task에 있는 클래스 끼리 Unbalance

- a. 수렴 순간이 달라 어떤 클래스는 학습 중에 다시 성능이 저하 됨

### 2. 이후 Task에서 수렴 속도가 매우 빠름 + 초기 성능이 좋음

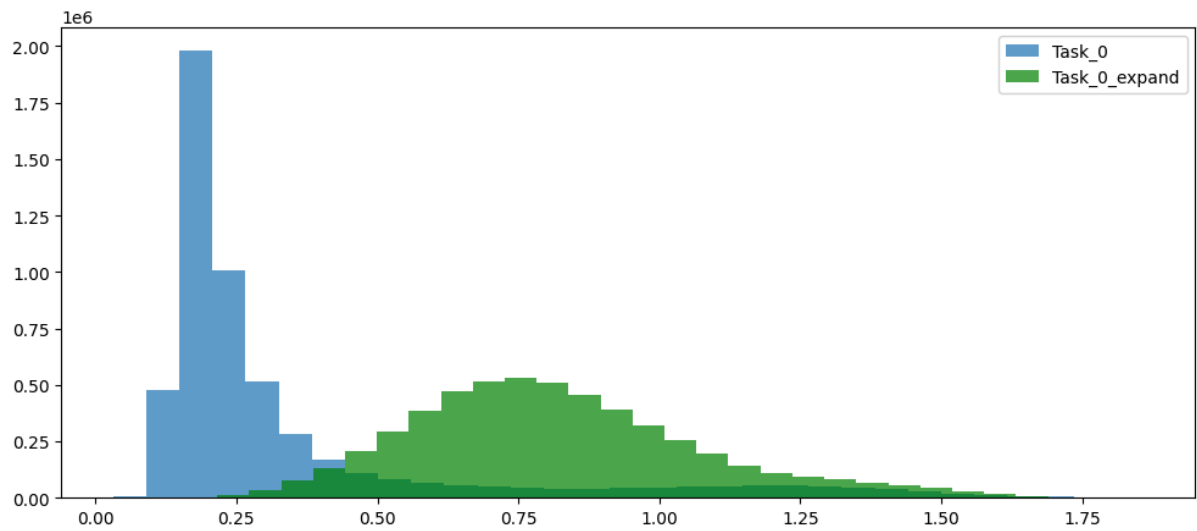
- a. Forward transfer
- b. 이후 Task를 반대로 최초 Task로 학습 시 바로 성능이 좋지 못 함
- c. Forward transfer가 좋다는 것 확인
- d.

### class 분류능력 확인

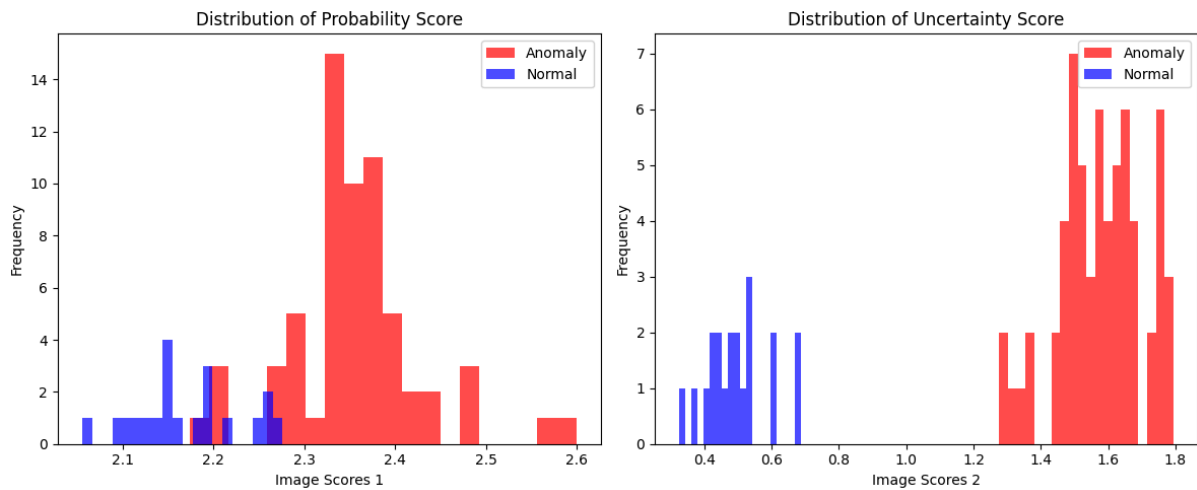
- classification bias? 확인 → 이게 일종의 spurious forgetting으로 나타나는 것 아닐까
- task\_identity를 줄 때나 안 줄 때나 성능 차이 크게 나타나지는 것 같음
  - 주든 안주든 여전히 성능 낮음
- 다만 자기 task stage에서는 task\_identity 무관하게 성능은 잘나옴
  - 학습 직후에는 class 분류 능력은 좋은 것으로 보임

## Anomaly Score 변동 확인

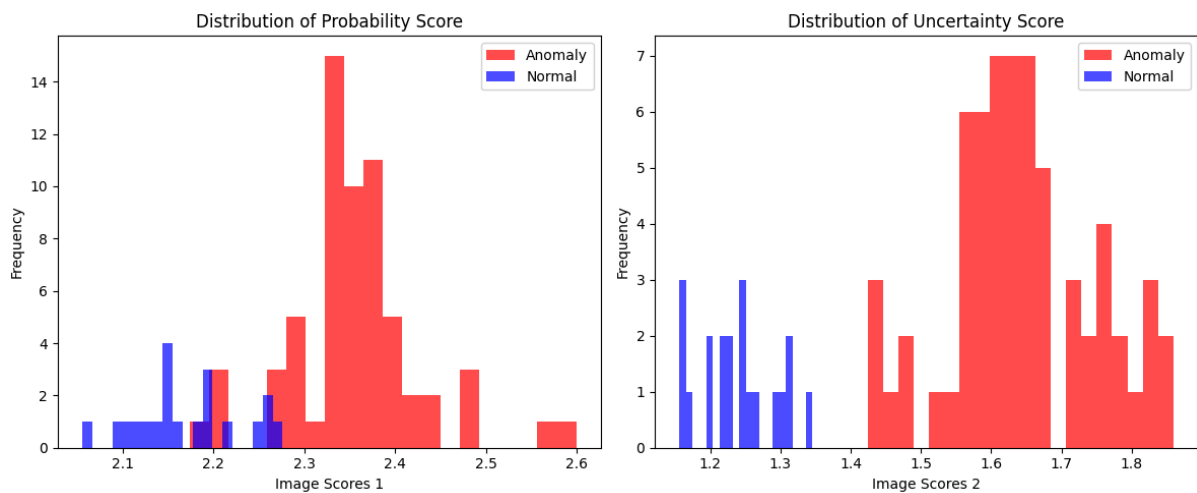
### Task0 학습 직후 및 expand 후 비교

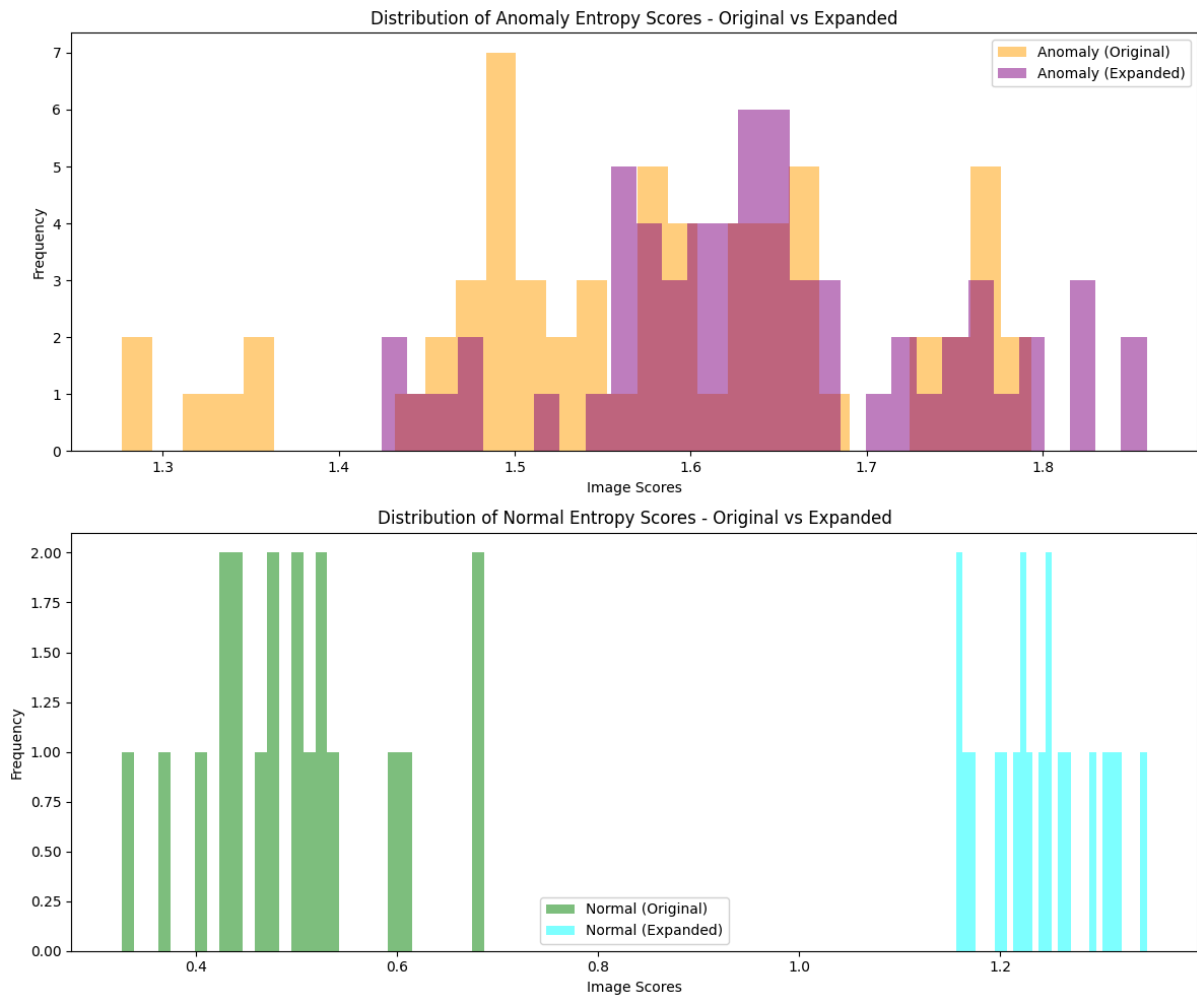


- 학습 직후



- expand 후





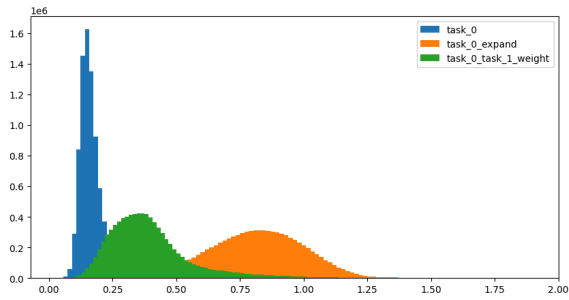
uncertainty score 정상 / 이상 비교 : 이상치에 대한 score는 변동이 적고, 정상에 대한 score의 변동이 큼

- 정상과 이상의 anomaly score를 나눠서 봤을 때 probability score는 expand를 하던 안하던 변화 없음
  - 예측한 target의 클래스 중심점 만을 가지고 확률을 예측하기 때문
- 반면 uncertainty score의 경우 expand 시 정상에 대한 score가 크게 증가 (anomaly에 대한 score는 크게 변화 없음)
  - 0.0869 → 0.4335

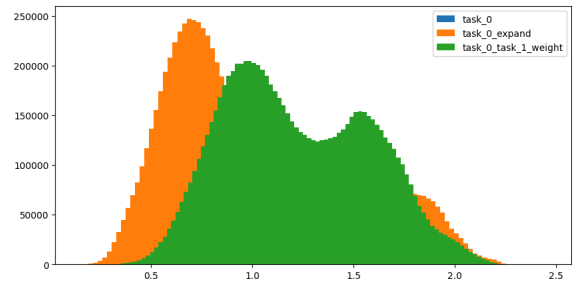
## Task1 학습 후 Task0

uncertainty score

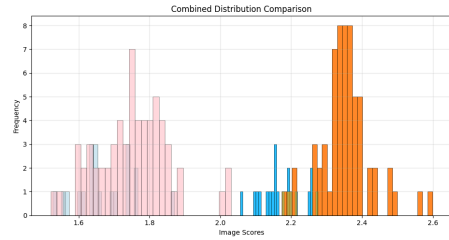
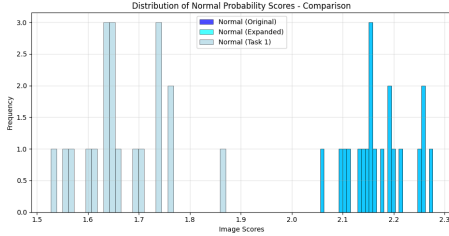
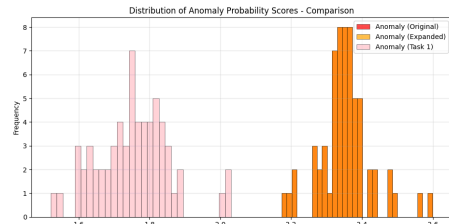
probability score



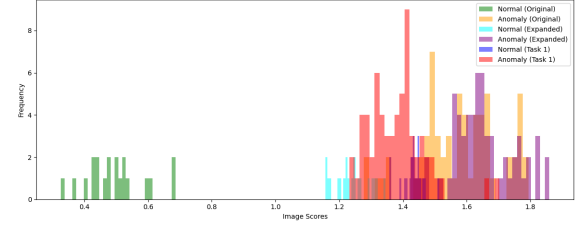
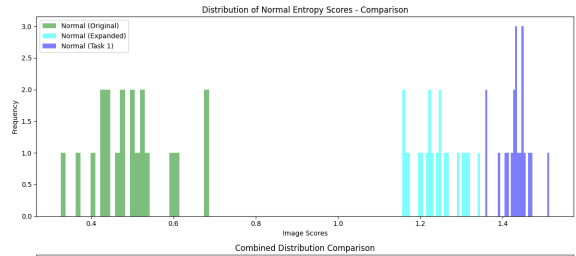
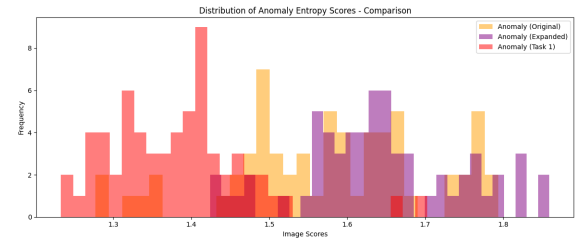
픽셀 수준 probability score



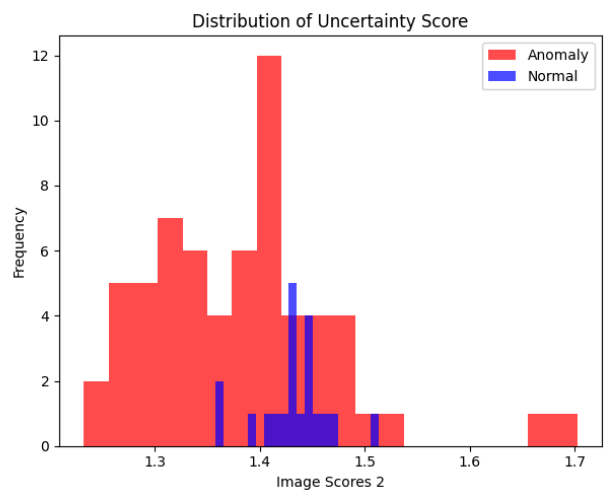
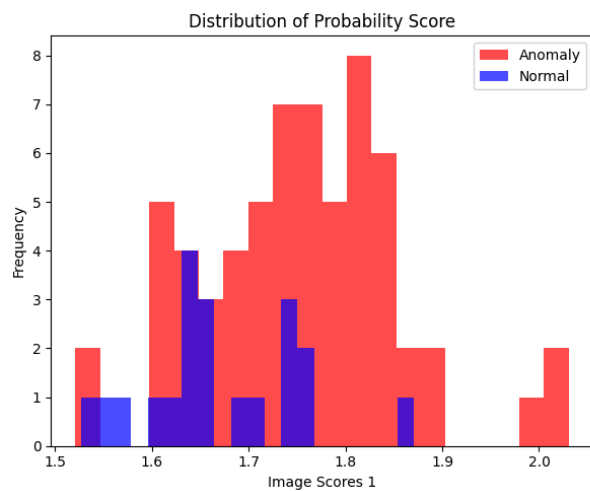
픽셀 수준 uncertainty score



probability score 비교



uncertainty score 비교 : 이상에 대한 score는  
변동이 없지만 정상에 대한 score가 계속 증가



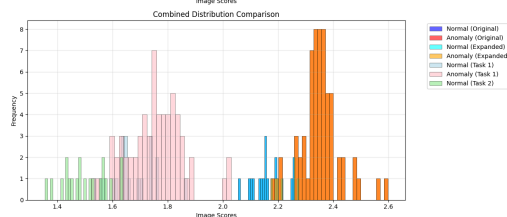
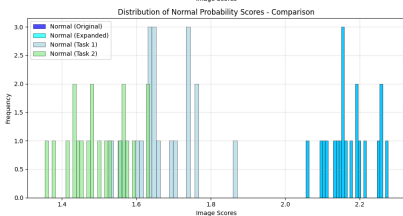
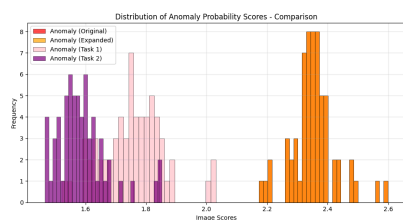
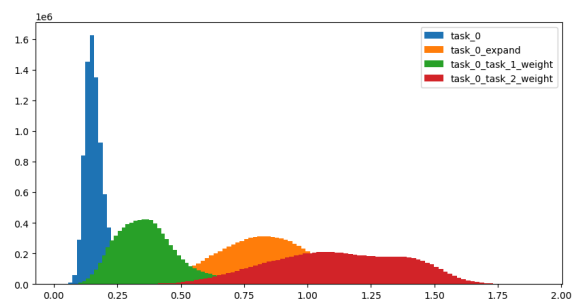
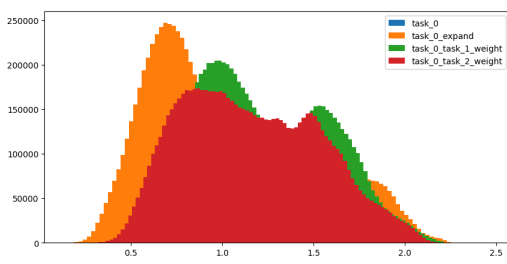
정상 / 이상에 따라 이미지 레벨 이상 스코어

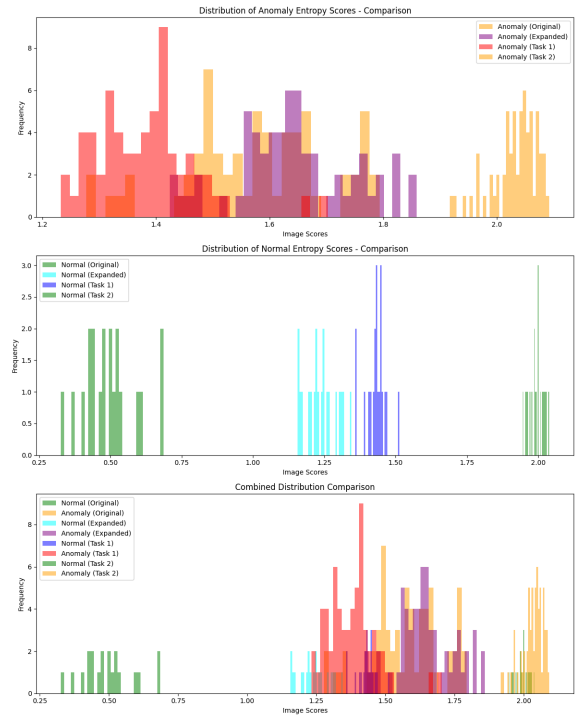
- 최종 Anomaly score는 probability score와 uncertainty score의 곱으로 결정되기 때문에 expand 후 uncertainty score가 변함에 따라 성능이 변하게 됨
- 정상에 대한 이상 스코어가 증가했으므로 FPR이 증가
- task\_0 학습 후 expand 했을 때는 task\_0의 mu들과 task\_1의 mu들 간의 정렬이 안 되었기 때문에 uncertainty가 크게 증가함. 그래서 오히려 task\_1을 학습하고 난 다음에는 이것들이 어느정도 정렬 되었기 때문에 오히려 uncertainty들이 감소한 것을 확인
- 그러나 task\_1을 학습한 다음 log-likelihood에 의한 anomaly score가 증가한 것을 확인

## Task 2

task\_2 학습 후 task\_0의 anomaly score

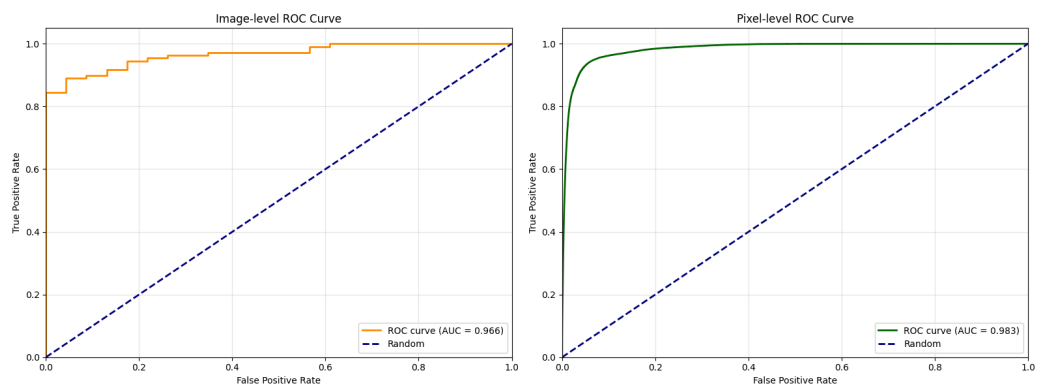
- log-likelihood score
- uncertainty score



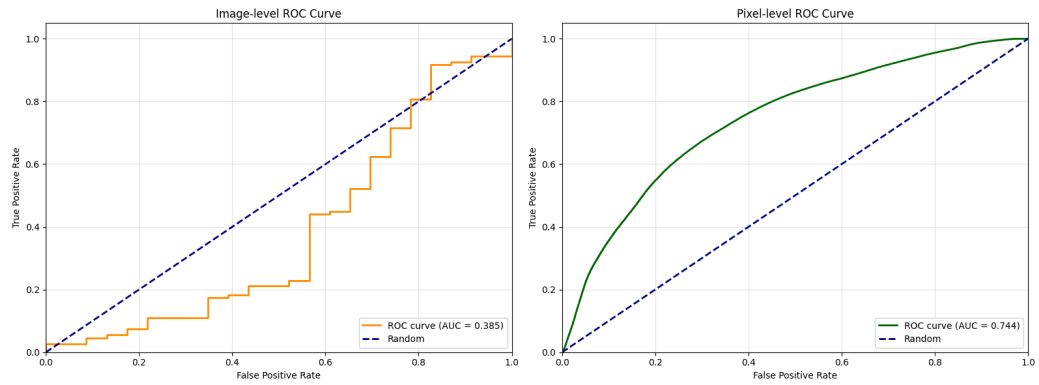


## ROC-AUC curve 확인

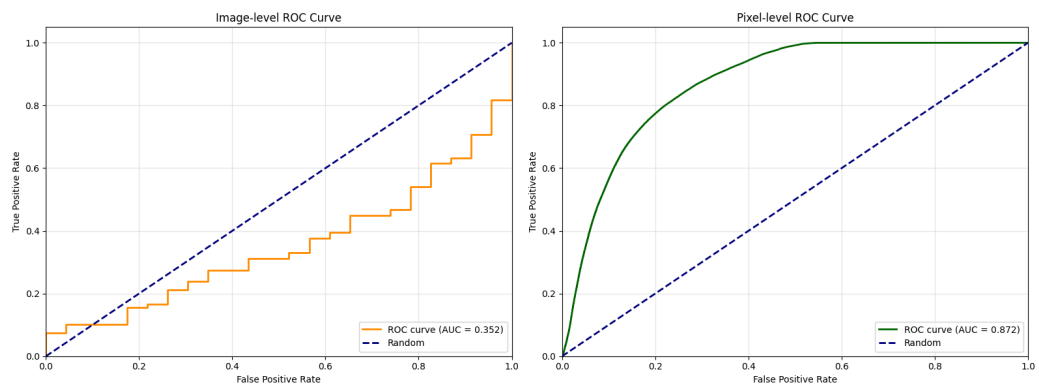
- 보면 전반적으로 log-likelihood에 의한 anomaly score는 크게 변동되지 않는 것으로 확인
- 그러나 uncertainty가 continual learning이 진행됨에 따라 크게 증가하는 것을 확인
  - ROC\_AUC curve 변화
    - task\_0 직후



- Task\_1 직후 Task\_0



## ■ Task\_1 직후 Task\_0 (log-likelihood 만)



## ■ 그렇다고 log-likelihood만 쓴다고 해서 성능이 좋아지지는 않음

- 이는 task\_1 학습 후 task\_0의  $\mu$ 가 변하였기 때문
- NF의 Task\_0에 대한 knowledge는 보존되고 있으나, 이 knowledge들이 task\_0의  $\mu$ 들과 정렬이 틀어졌기 때문