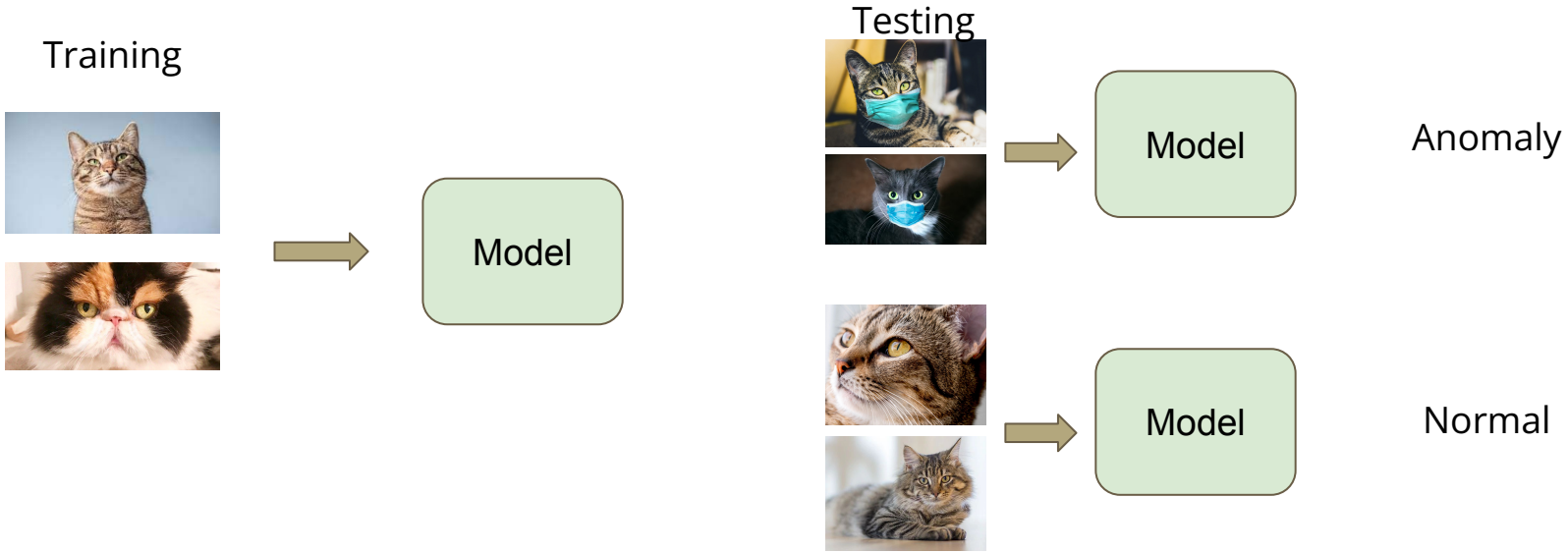
Machine Learning Homework 8
Anomaly Detection

Outline

- Task introduction
- Data
- Methodology
- Evaluation
- Baseline
- Report

Task Introduction

- Unsupervised anomaly detection
  - Training a model to determine whether the given image is similar with the training data.



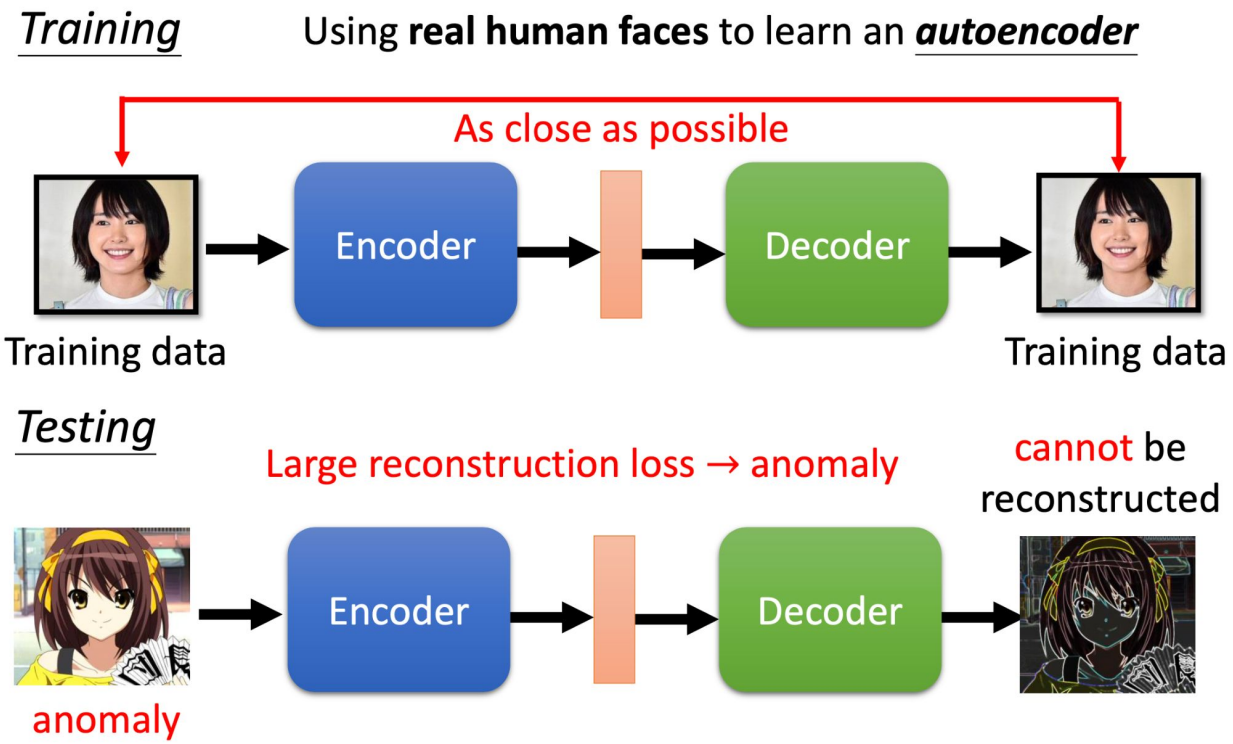
Data

- Training data
  - 100000 human faces
- Testing data
  - About 10000 from the same distribution with training data (label 0)
  - About 10000 from another distribution (anomalies, label 1)
- Format
  - data/

    |----- trainingset.npy

    |----- testingset.npy
  - Shape: (#images, 64, 64, 3) for each .npy file

Methodology



Methodology

- Train an autoencoder with small reconstruction error.
- During inference, we can use reconstruction error as anomaly score.
  - Anomaly score can be seen as the degree of abnormality of an image.
  - An image from unseen distribution should have higher reconstruction error.
- Anomaly scores are used as our predicted values.

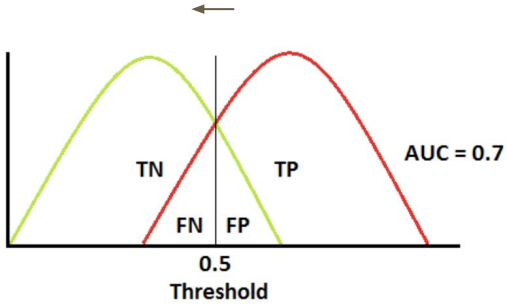
Evaluation - ROC AUC score

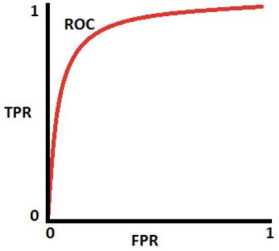
Why using ROC AUC score?

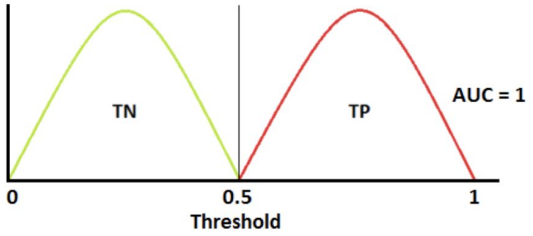
- If accuracy is applied, then a threshold is needed to determine the given image is an anomaly or not.
  - We only want a model that tells us how anomalous an image is.
  - e.g. MSE is a kind of anomaly score
- More about ROC curve
  - https://en.wikipedia.org/wiki/Receiver_operating_characteristic

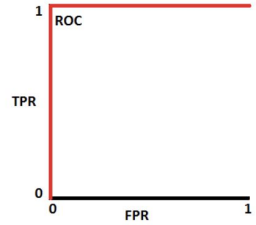
Evaluation - ROC AUC score

- TPR = TP / (TP + FN)
- FPR = FP / (FP + TN)









https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

Evaluation - ROC AUC score Example

ID	Anomaly score	Label
0	11383	0
1	256676	1
2	862365	1
3	152435	0
4	848171	0

Sort by score

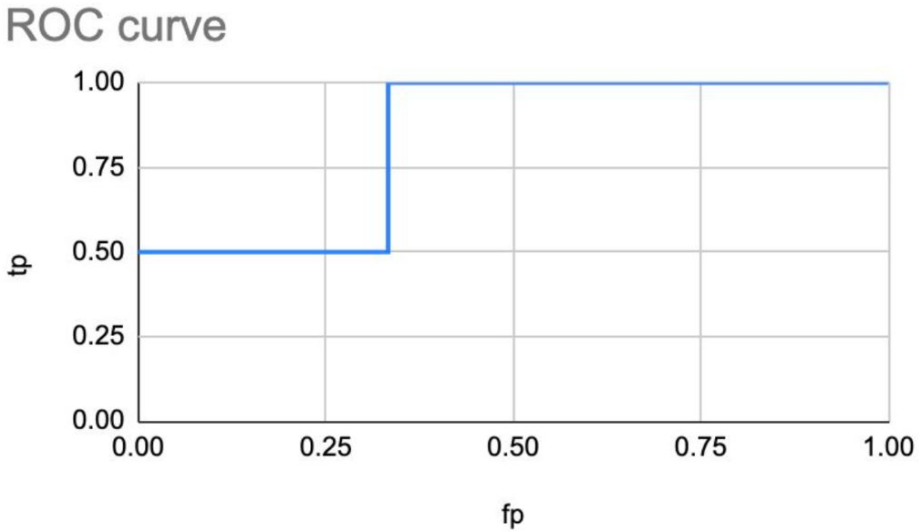
ID	Anomaly score	Label
2	862365	1
4	848171	0
1	256676	1
3	152435	0
0	11383	0

Evaluation - ROC AUC score Example

ID	Anomaly score	Label	fp before normalization	tp before normalization
2	862365	1	0	1
4	848171	0	1	1
1	256676	1	1	2
3	152435	0	2	2
0	11383	0	3	2

Evaluation - ROC AUC score Example

ID	Anomaly score	Label	fp	tp
0	11383	0	0	0.5
3	152435	0	0.333333	0.5
1	256676	1	0.333333	1
4	848171	0	0.666667	1
2	862365	1	1	1

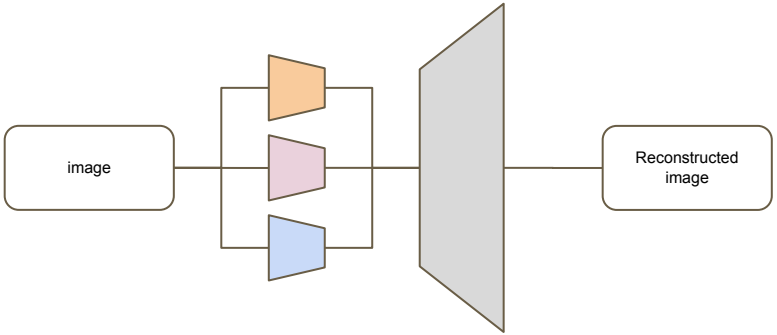


Area Under Curve: 0.5*⅓ + ⅔ = 0.8333

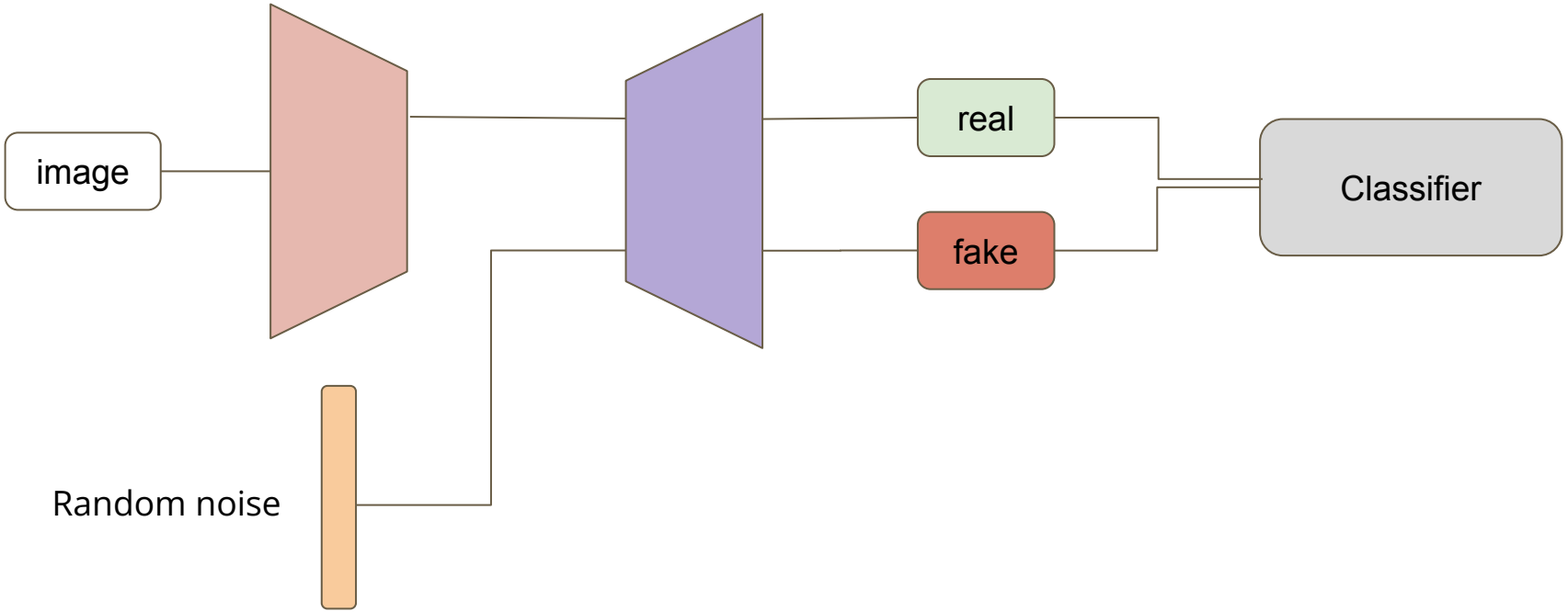
Baseline

- Simple
  - Sample code
- Medium
  - Adjust model structure
- Strong
  - Multi-encoder autoencoder
- Boss
  - Add random noise and an extra classifier
  - Papers of anomaly detection

Multi-encoder autoencoder



Add random noise and extra classifier



Report

1. Make a brief introduction about variational autoencoder (VAE). List one advantage comparing with vanilla autoencoder and one problem of VAE.

2. Train a fully connected autoencoder and adjust at least two different element of the latent representation. Show your model architecture, plot out the original image, the reconstructed images for each adjustment and describe the differences.

Report - Q2 (added at 4/26)

For instance, let z be the output of encoder. Then we can adjust the first dimension of z as follows:

- z[0] = 2 * z[0]

Note: you should use the same autoencoder and only adjust the latent representation (output of encoder).

Grading

- Simple Baseline (Public /Private)
- Medium Baseline (Public /Private)
- Strong Baseline (Public /Private)
- Boss Baseline (Public /Private)
- Code Submission
- Report