

# Domain adaptation and fairness

Fiona Guo

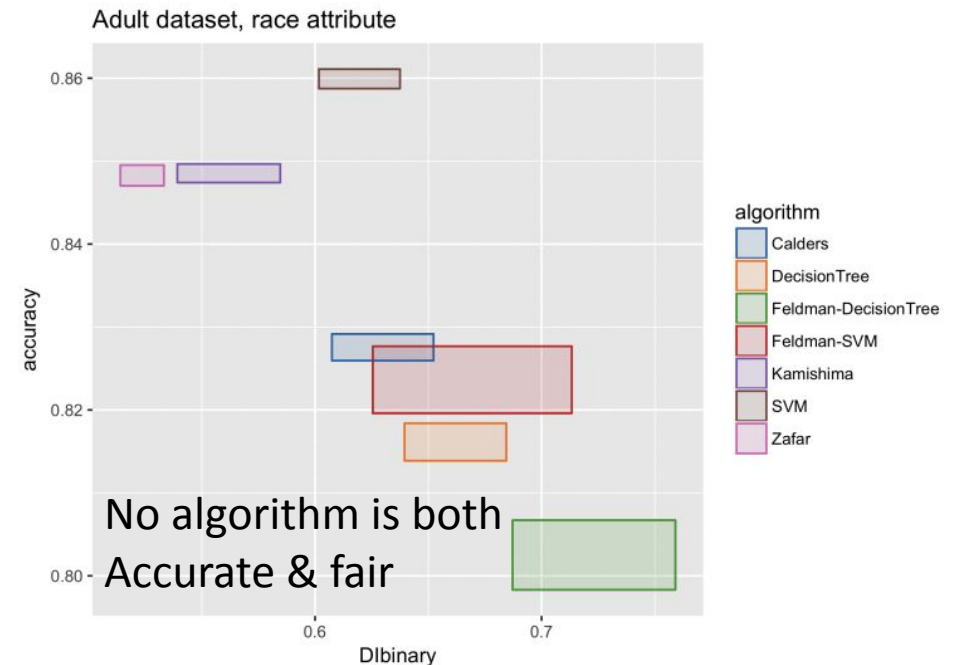
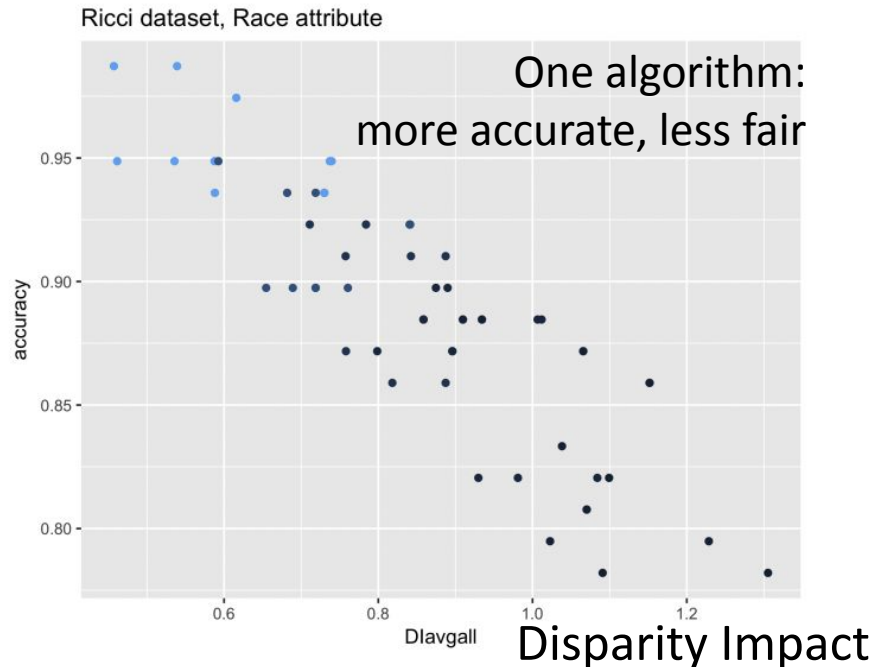
Feb. 26, 2025

# Themes of this class

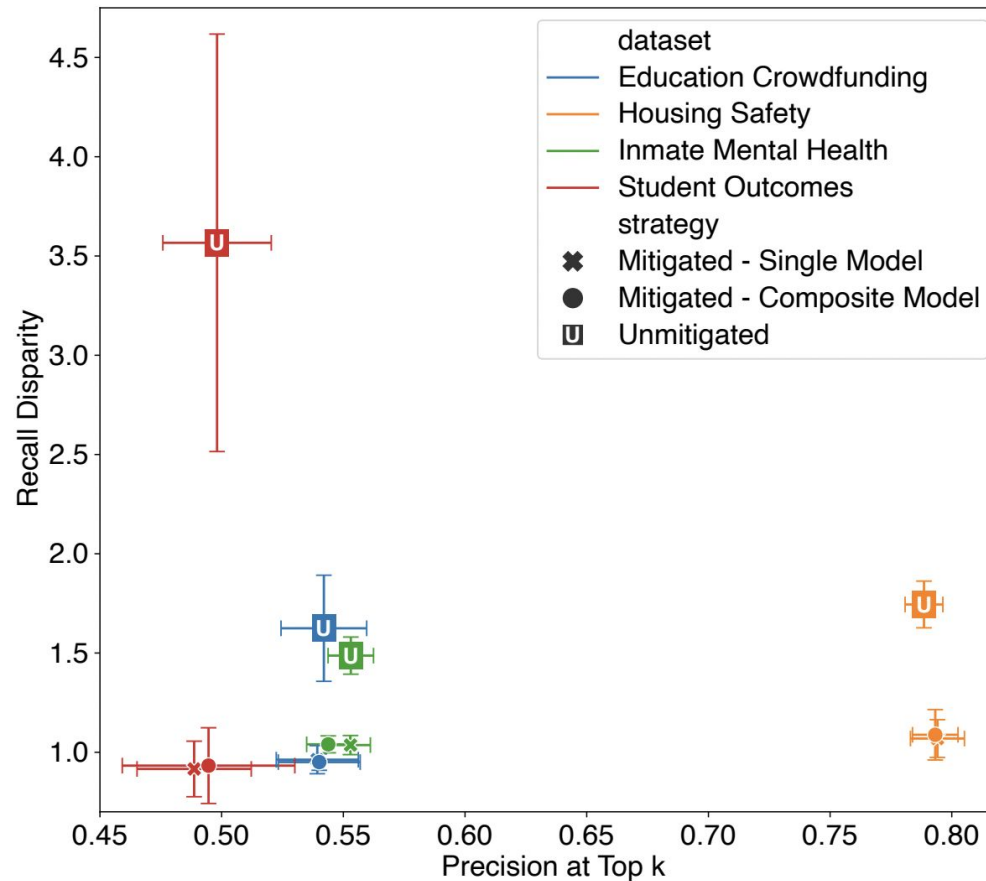
- The cost and reward of fairness
- When are domain adaptation methods needed?
  - When is data “in domain” versus “out domain”
- Domain adaptation methods
  - ADDA
  - Multi-task learning
- Limitations to domain adaptation
  - When is a domain too different?
  - What to do if it is?

# The Fairness Tradeoff

- Fairness reduces accuracy, assuming fairness metrics
- E.g., improving statistical parity reduces accuracy
- Intuition: any constraint on a model outside of accuracy maximization reduces model effectiveness



# Empirical tradeoff

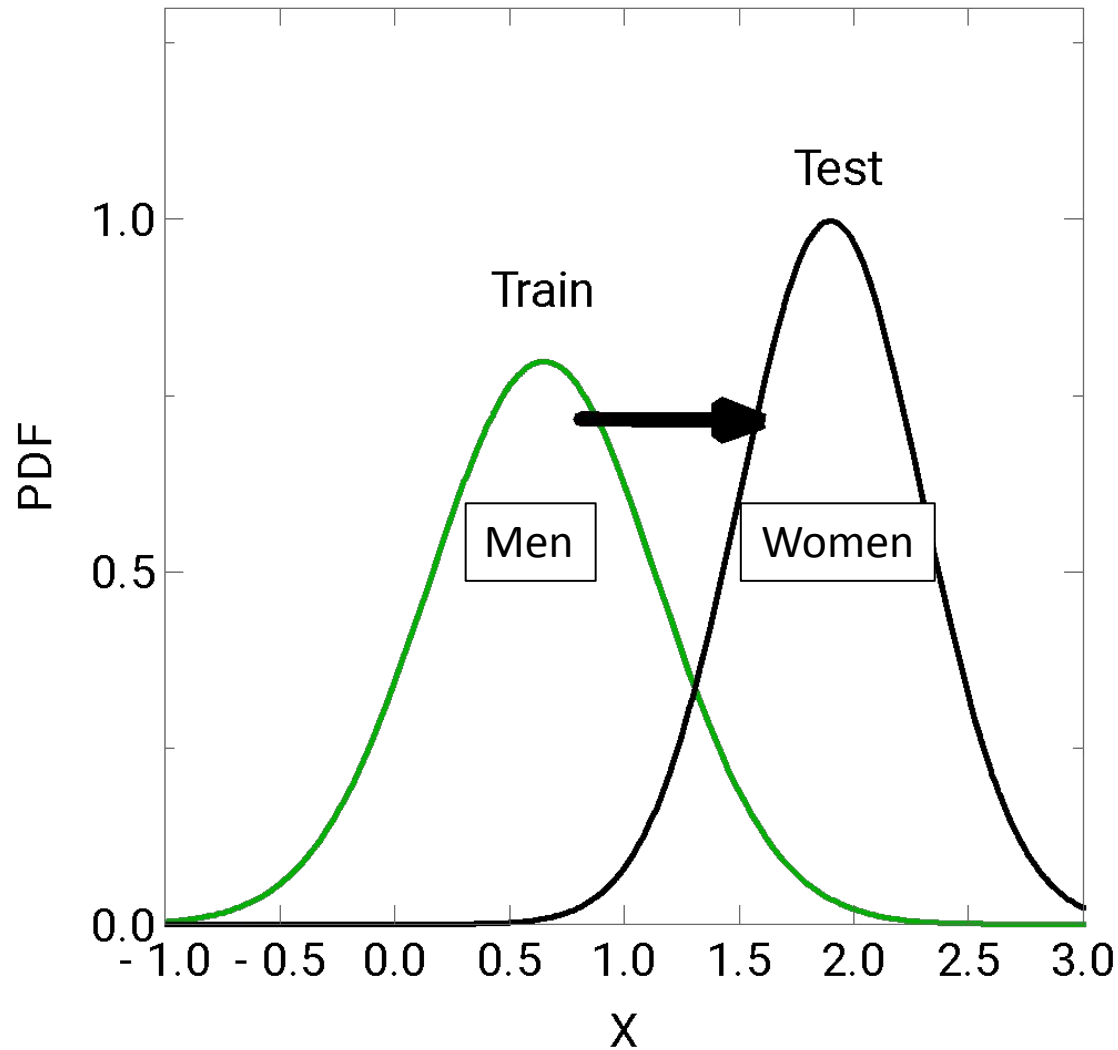


- Sometimes tradeoff is minimal
- Recall disparity: difference in recall between protected groups
- Precision at top k: precision that ground truth in top k predictions
- Plot shows reduction in recall disparity, no appreciable reduction in top k precision

# Fairness bargains?

- Why do some data show a trade-off while others do not?
- When is there a trade-off in data and when can models be both fairer and more accurate?

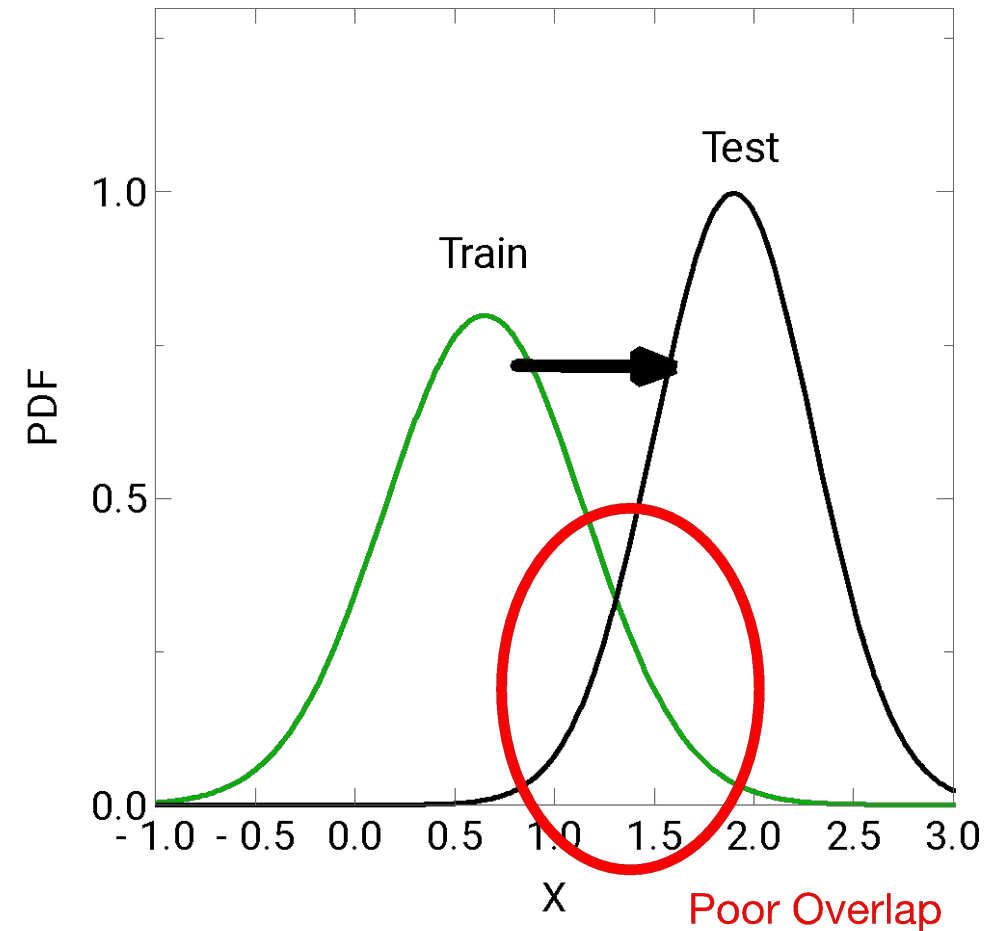
# This is why data shift is important!



- Fairness-bias tradeoff theorems implicitly assumes training/test data are similar
  - No impact from covariate/data shift
- Model can be unfair if
  - Data is unfair (e.g., historical biases)
  - **Demographics show covariate shift**
    - Even with representative data, model may not work well for demographics with unique patterns
  - **Data is not representative**
    - Features differ, or
    - Covariates differbetween protected groups, more/less of some protected groups in test set

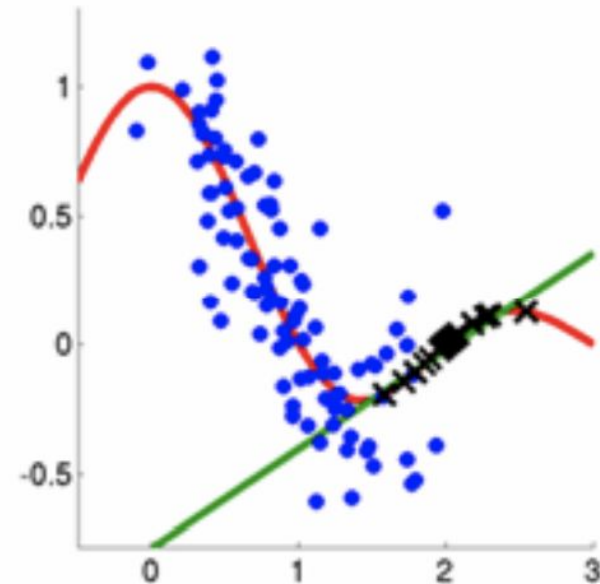
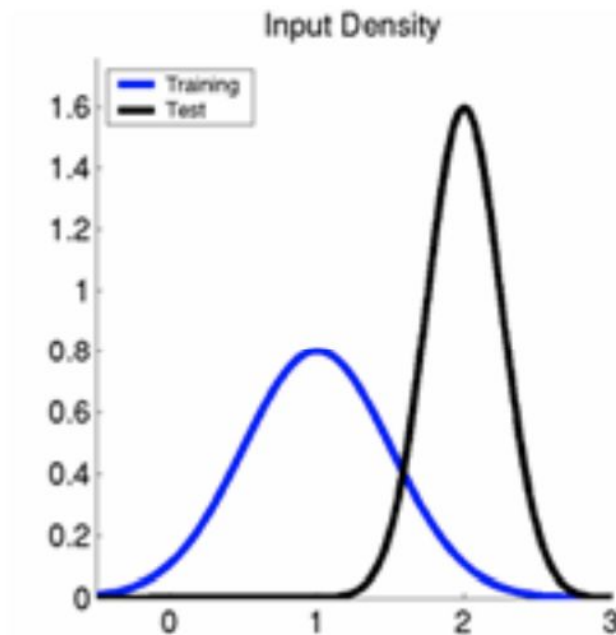
# What is Data Shift?

- Protected groups create different feature distributions
- One issue: train mostly on one distribution but test on another distribution
- Overlap is poor! Invariant solutions might perform poorly
- Solution 1: domain adaptation
  - Make distributions similar
- Solution 2: transfer learning & multi-task learning
  - Use model from one class to help solve other



# Why Data Shift is Critical: Covariate Shift

Training and test input follow different distributions, but functional relation remains unchanged.

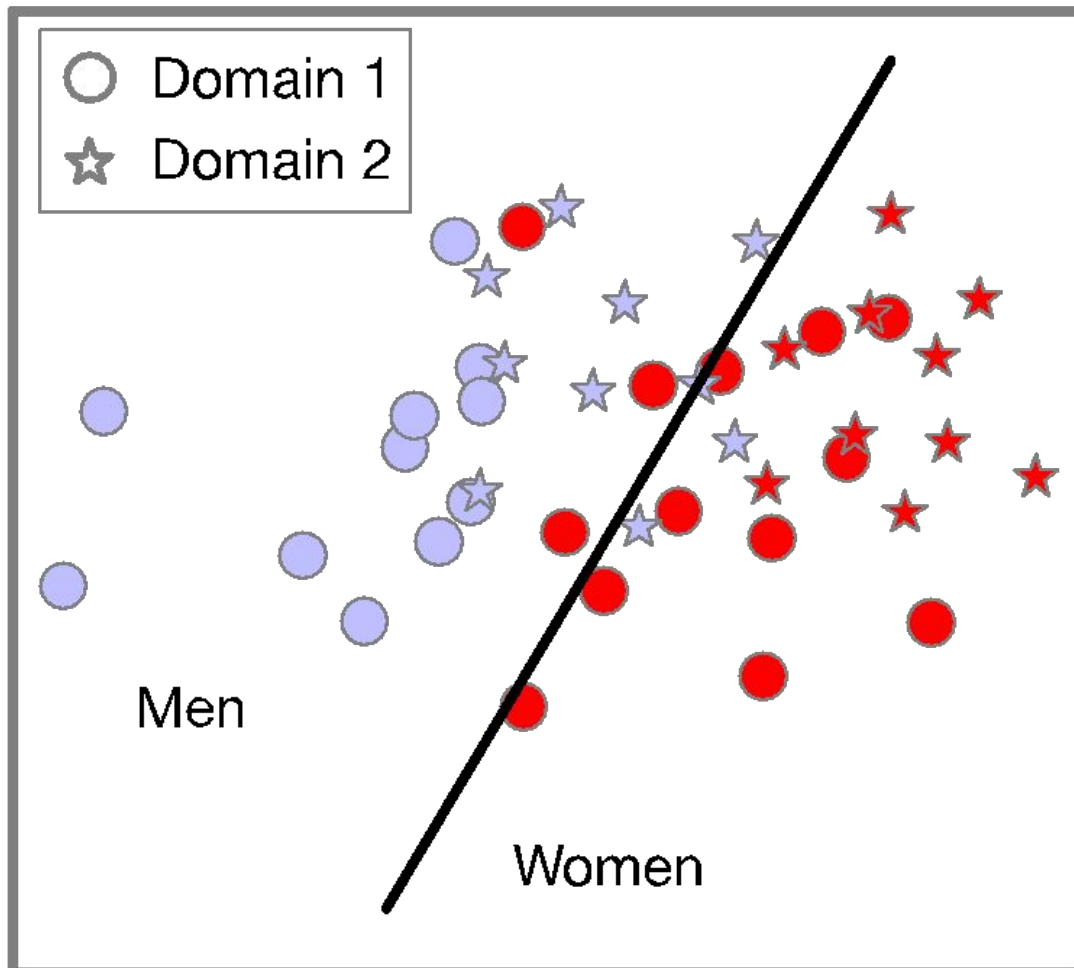


Goal: Estimate test output from  $\{(x_i, y_i)\}_{i=1}^n$

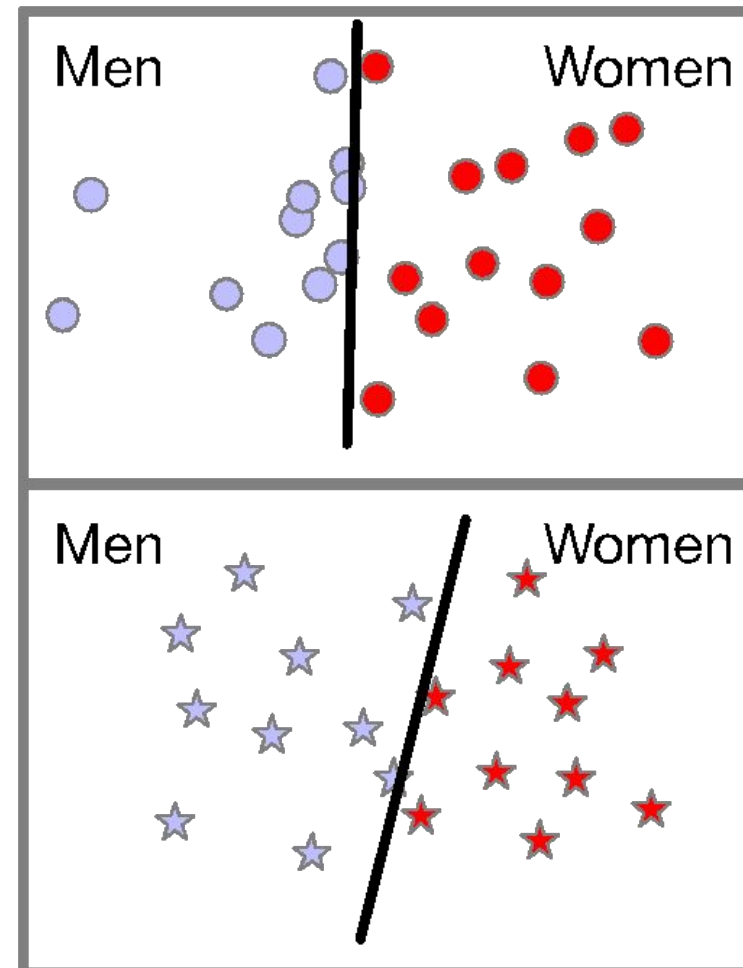


# Why Data Shift is Critical: Changes in Priors

Low Accuracy



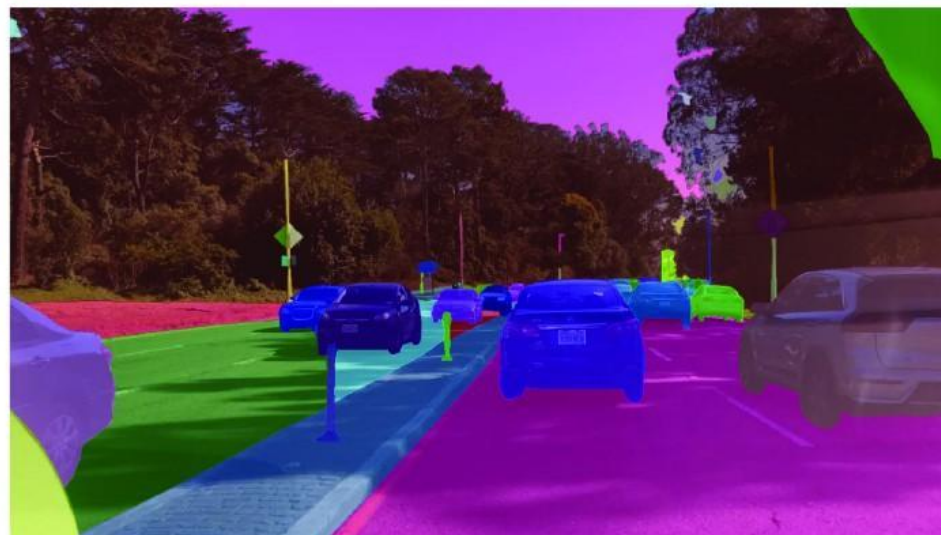
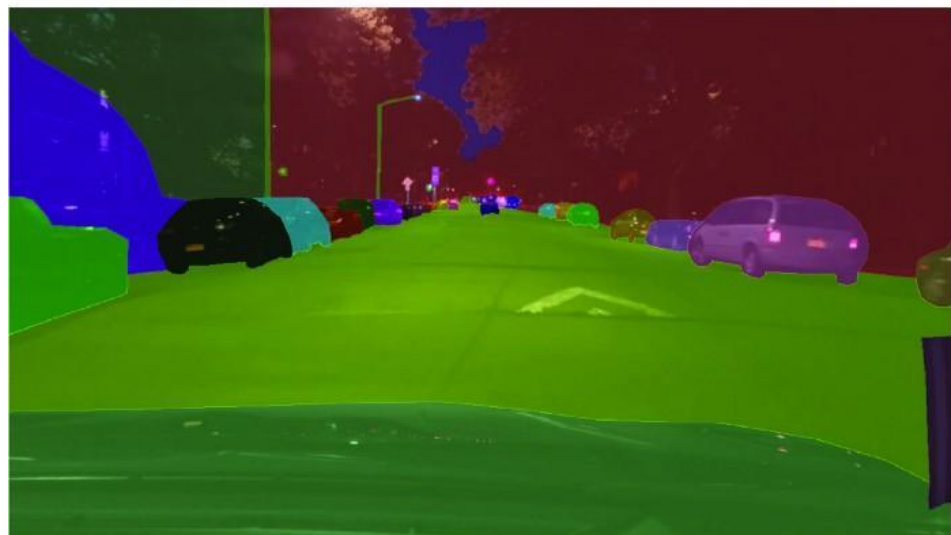
Perfect Accuracy



# Data Shift in Applications

- Applications need not be focused on demographic fairness!
- Many datasets contain systematic biases outside of social data
- Addressing these biases can make a better model
- Motivating Examples: Self-driving cars

# Self-driving cars and biases

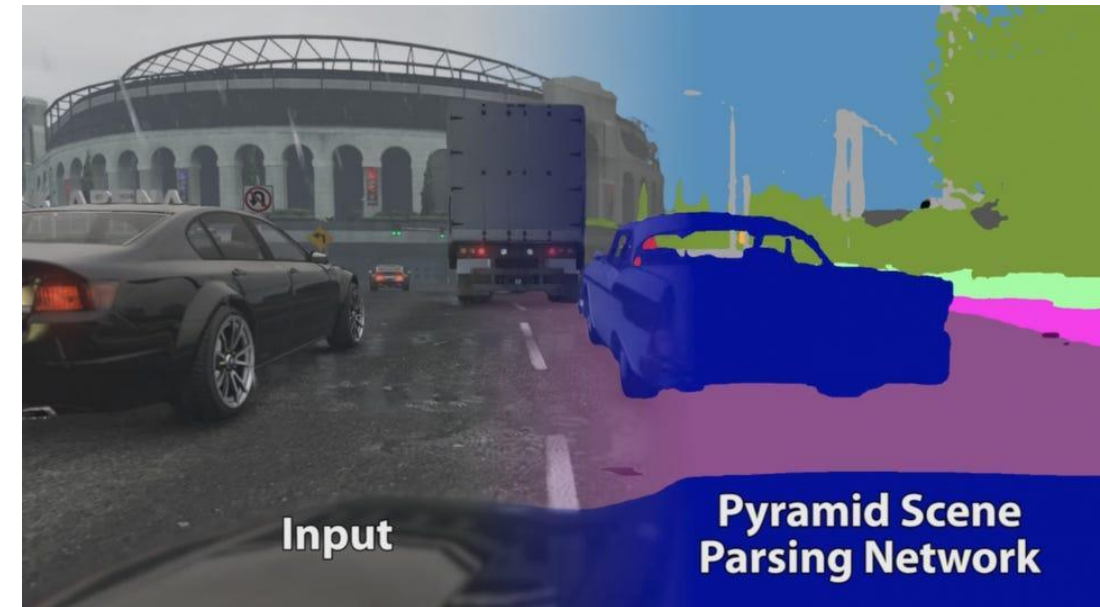


# How self-driving cars work

- AI models need to distinguish the road from cars or children
- Models use image segmentation (e.g., UNet) to label each pixel as a set of categories
- Realistic data is very expensive (\$20/image x 1000s of images) and needs to work across a variety of domains
  - ...which is why fully self-driving cars are so bad in urban environments
  - 1000s of images of daytime traffic work poorly when we are in nighttime or rainy environments

# How to make better data?

- Create simulations across a wide variety of situations
  - Ex: Grand Theft Auto 5
  - Advantages:
    - Data is pre-labeled (we know what a stop sign is)
    - Adaptable (create similar simulations for different countries, environments)
  - Disadvantage: lack of realism
- Domain Adaptation Approaches



Stephen Richter

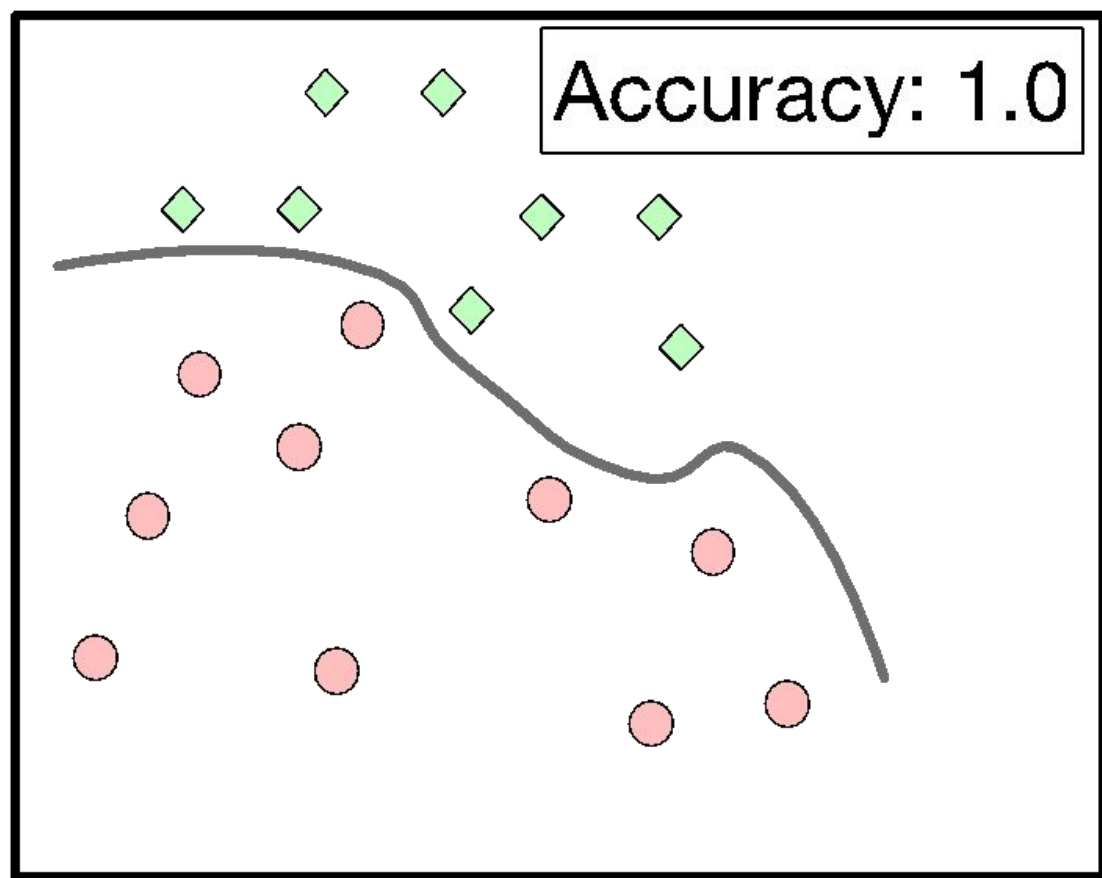
# Caveats to domain adaptation fairness

- It is hard to correct for historical biases
  - E.g., over-policing or over-arresting African Americans
  - Red-lining: using features correlated with protected groups to harm these groups
- In these cases, domain shift will not necessarily change fairness
- **Historical biases therefore reinforce a fairness-bias trade-off**
  - If domain adaptation is not a significant factor, theoretical assumptions better match data If domain adaptation is not a major factor, theoretical approaches (e.g., fairness-aware learning) may work better for handling bias.
- Anonymity and fairness tradeoff (e.g., a fairer Facebook algorithm might run into issues if it requires detailed demographic information user's do not want to share)

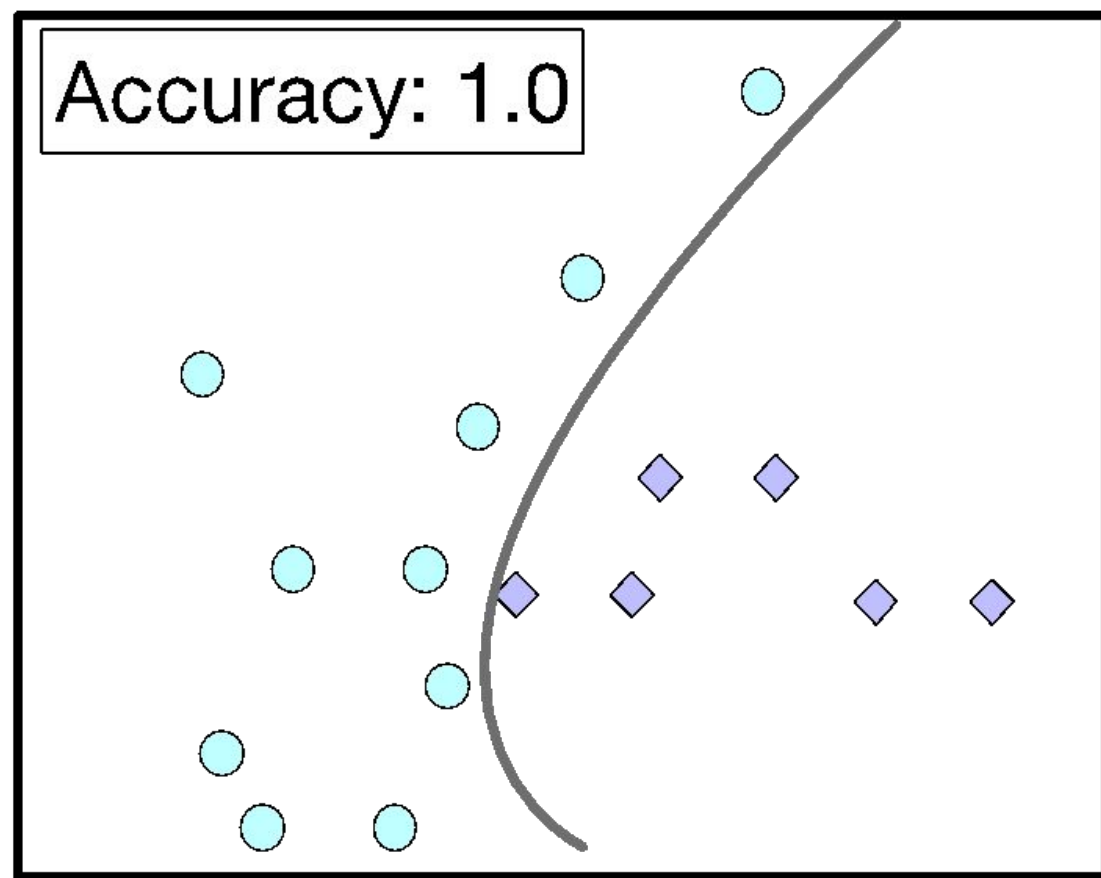
# Domain Adaptation Methods

# Case Study

Group 1

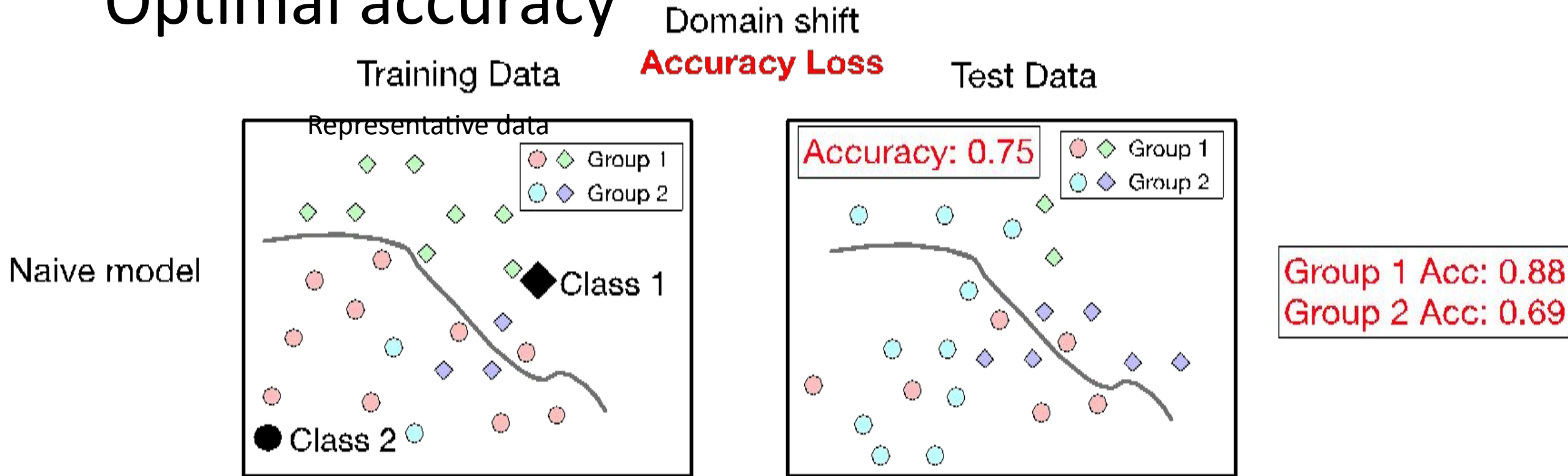


Group 2



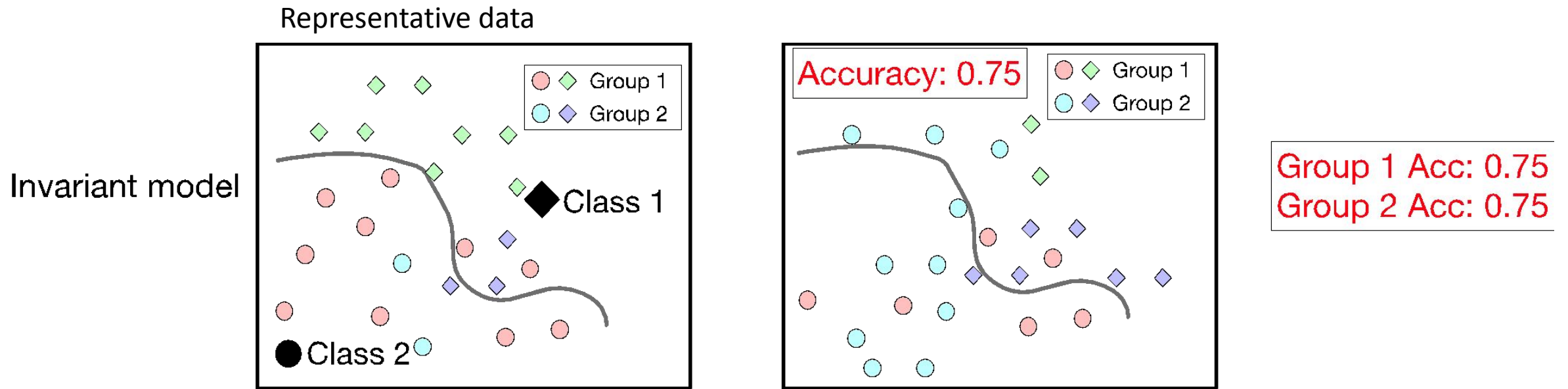


# Optimal accuracy



- Good model may show disparities for different groups
- Domain shift (fewer of Group 2 in training) demonstrates accuracy loss

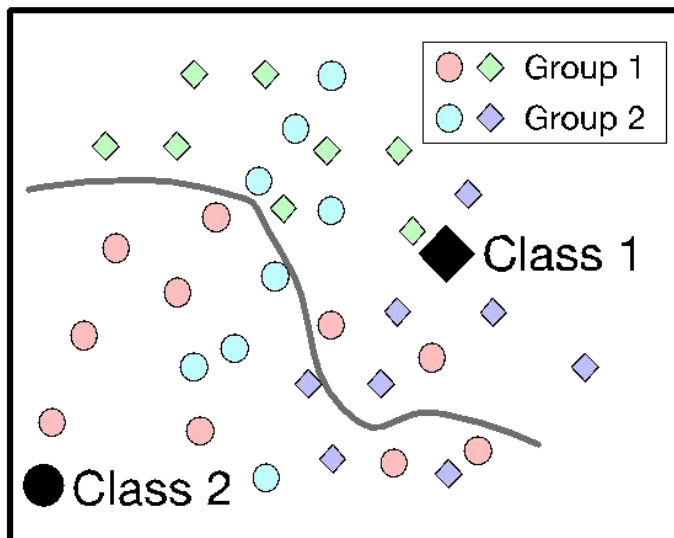
# Optimal fairness



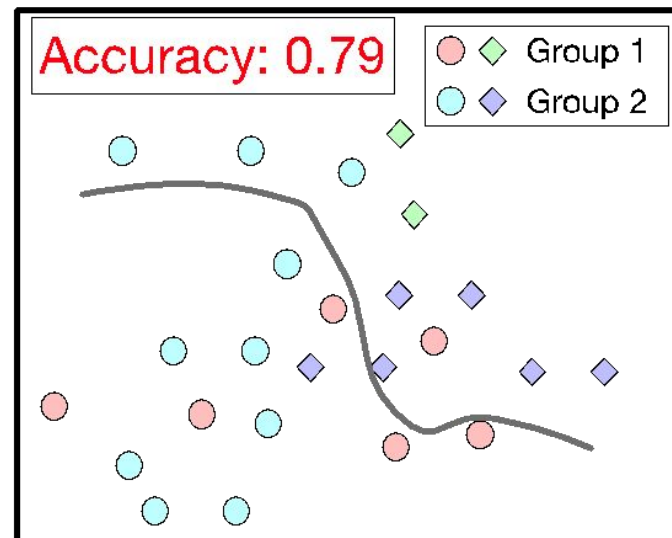
- Fair model: accuracy similar across groups (one metric of fairness)
- Fairer model does not need to be less accurate
- But may ignore important features

# Upsampling

Training Data



Test Data



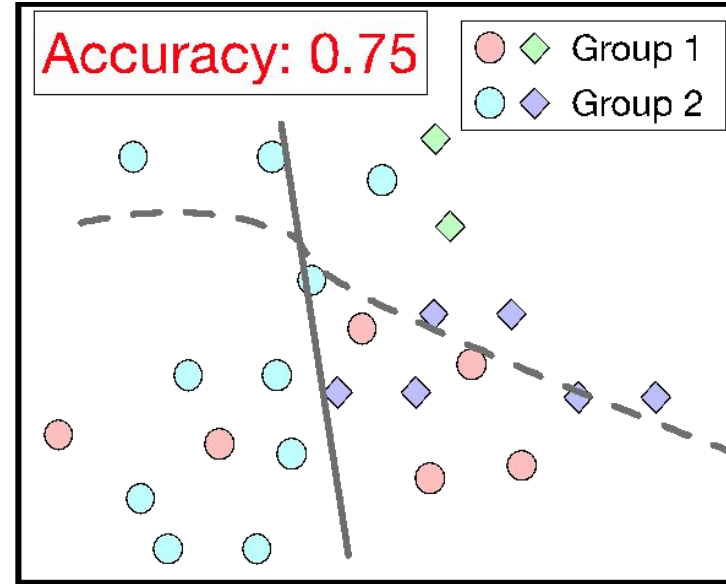
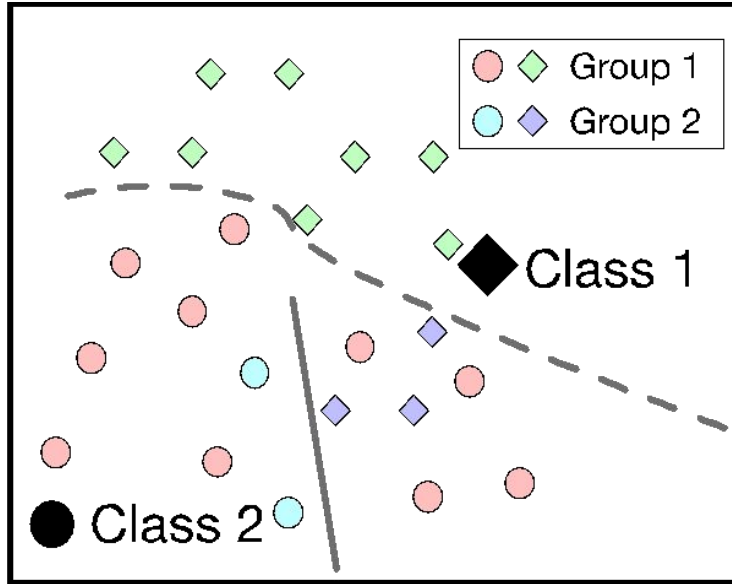
Group 1 Acc: 0.88  
Group 2 Acc: 0.75

- Overall accuracy improves
- Accuracy increases for each group
- Gap in accuracy reduced
- Not a panacea: we still see a gap

万能药

# Fine-tuning to different demographics

Fine-tuned  
Model



Group 1 Acc: 1.0  
Group 2 Acc: 0.88

- First: predict demographic, then train model on these demographics
  - eg Choi et al., 2020; Burghardt et al., 2023
- Accuracy increases! Gap is reduced (still not as good as fair model)

# Overall conclusions

- Domain adaptation method examples

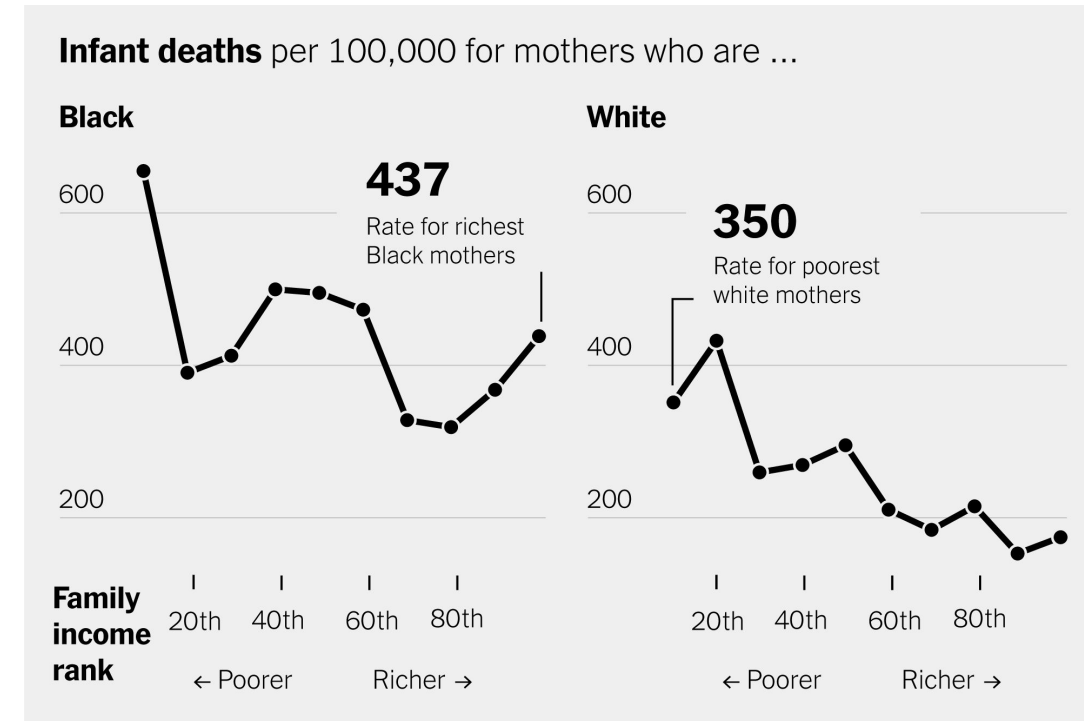
- Upsampling
- Fine-tune model

Improve data representation to make testing/training distributions similar

- Domain adaptation can create more accurate and fairer models
- No guarantee they are fairer! Not optimized to improve fairness
- Fairer models show fairness/accuracy tradeoff (no need for this tradeoff if we address domain adaptation)

# Universal versus demographic-specific characteristics

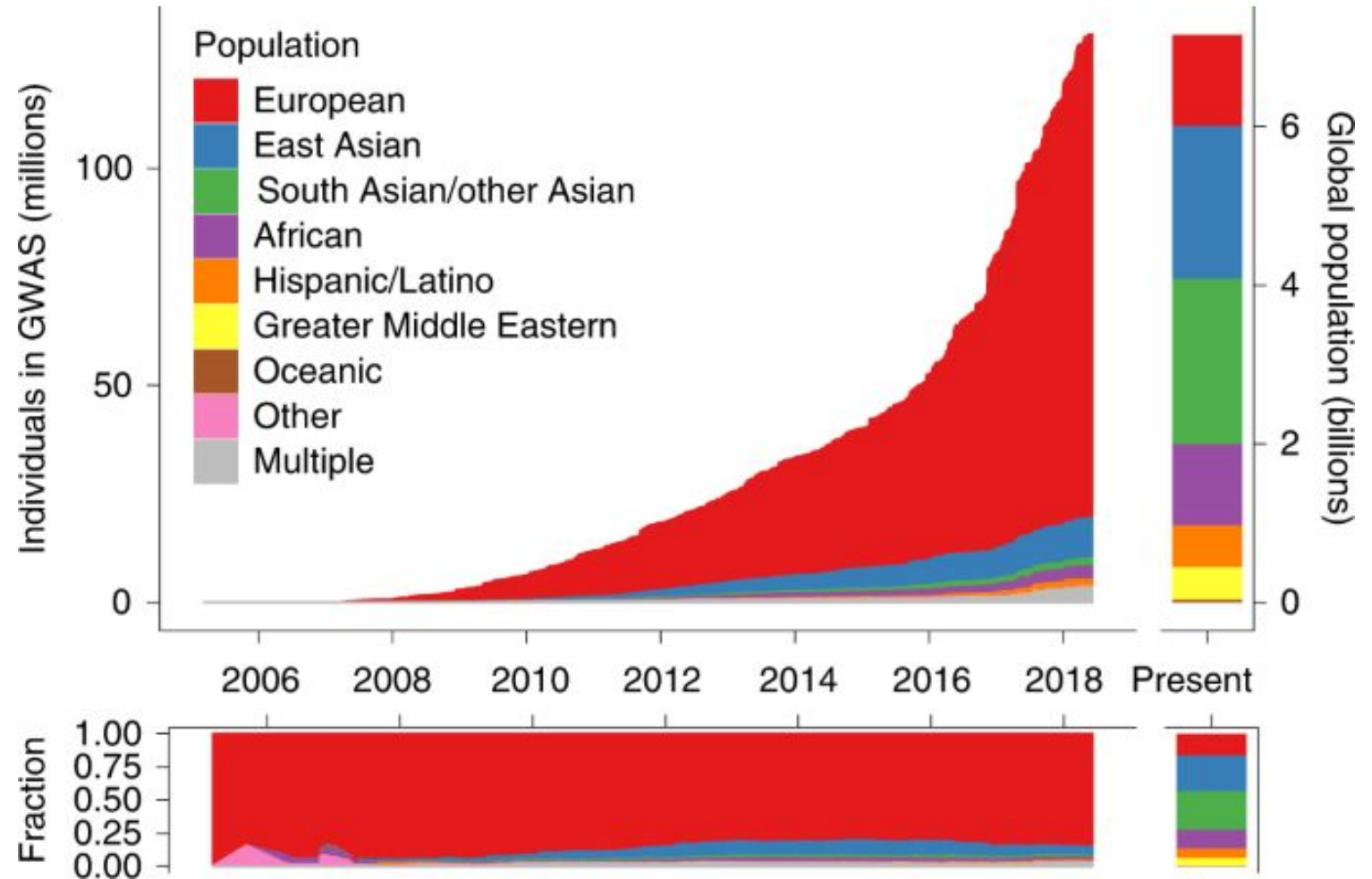
- Issue in medical domain
  - Demographics may be important predictor
  - E.g., greater likelihood of infant mortality due to systematic racism
  - Invariant models could counter-intuitively perform more poorly on some demographics
- Another motivation for domain adaptation



# Domain adaptation applications

- Polygenic risk scores
- ADDA
- Multi-task learning

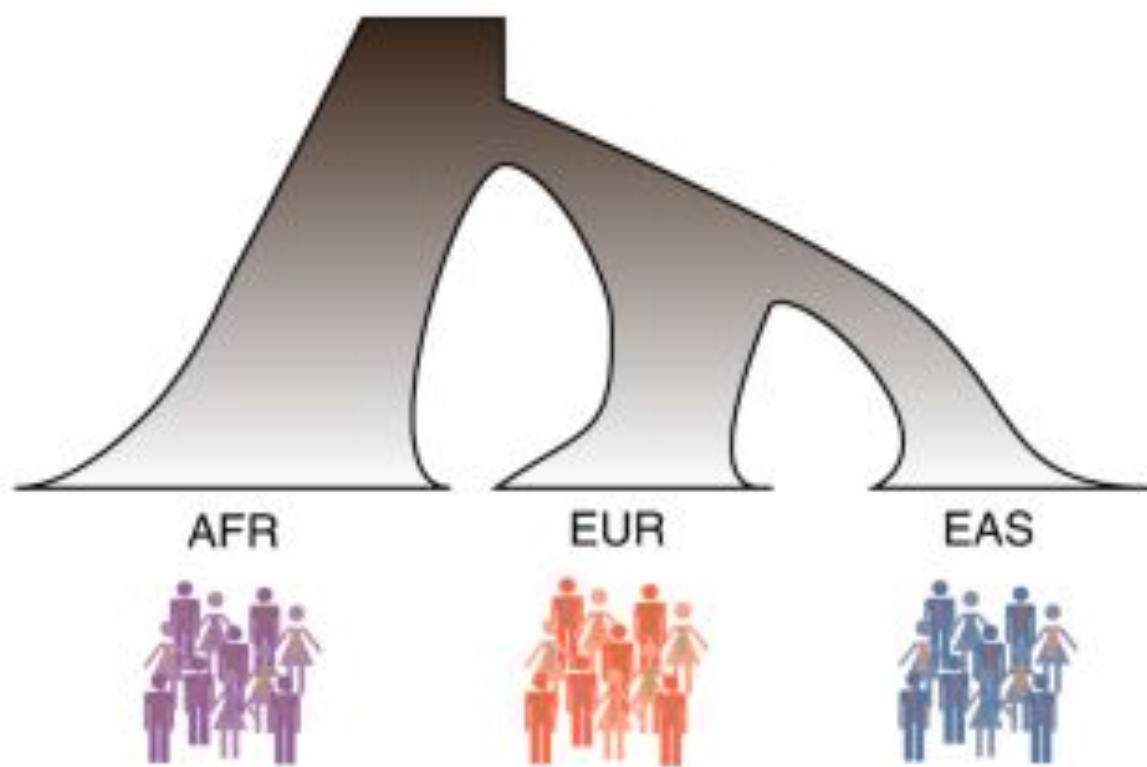
numerical estimates of an individual's genetic predisposition to developing a complex disease or trait



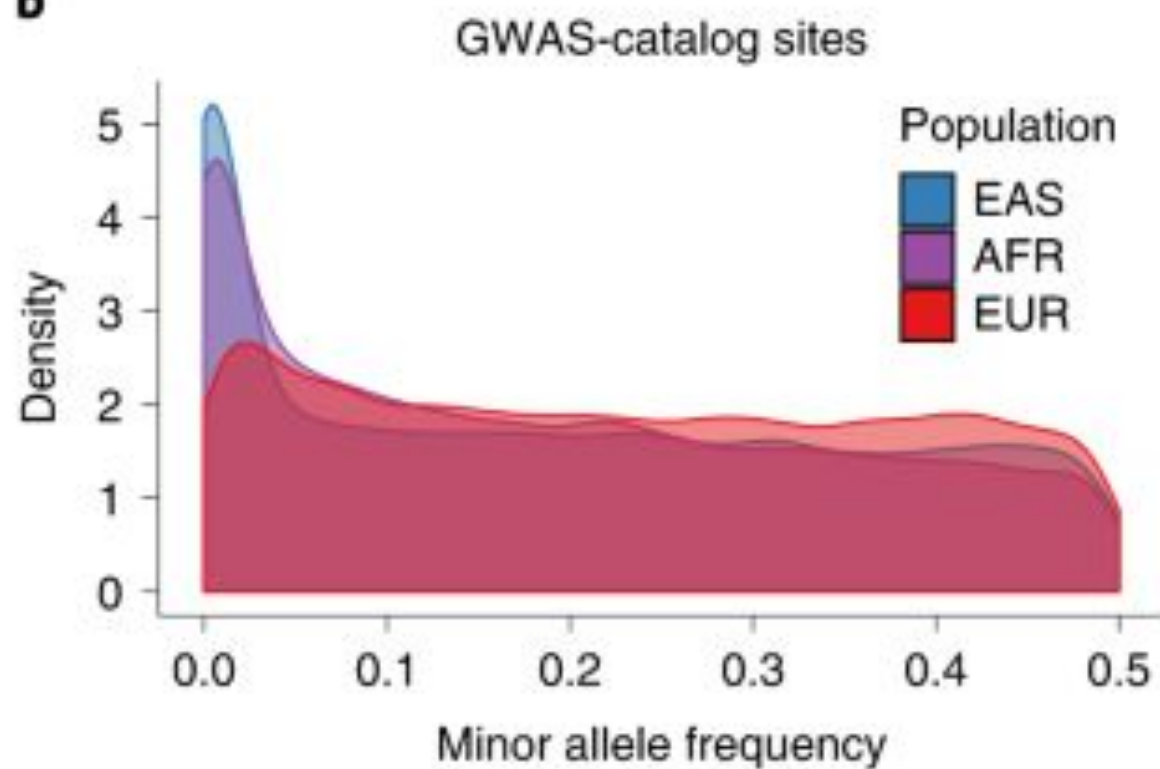


# Feature shift

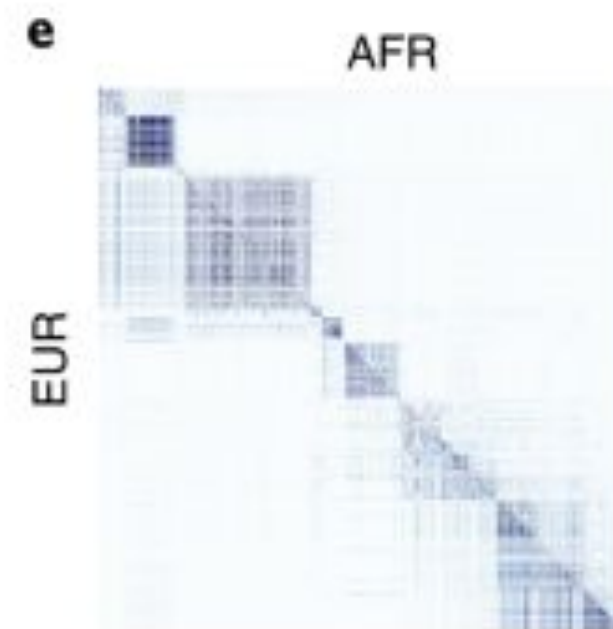
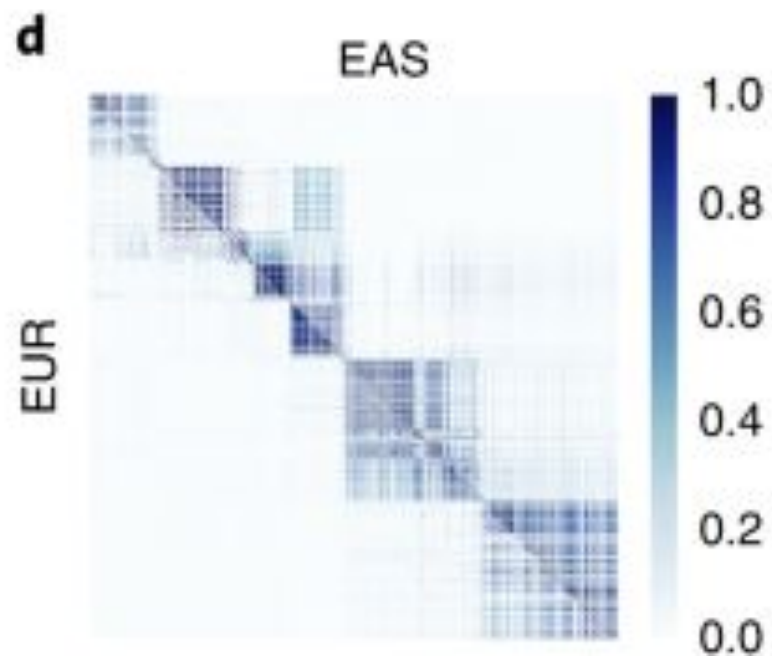
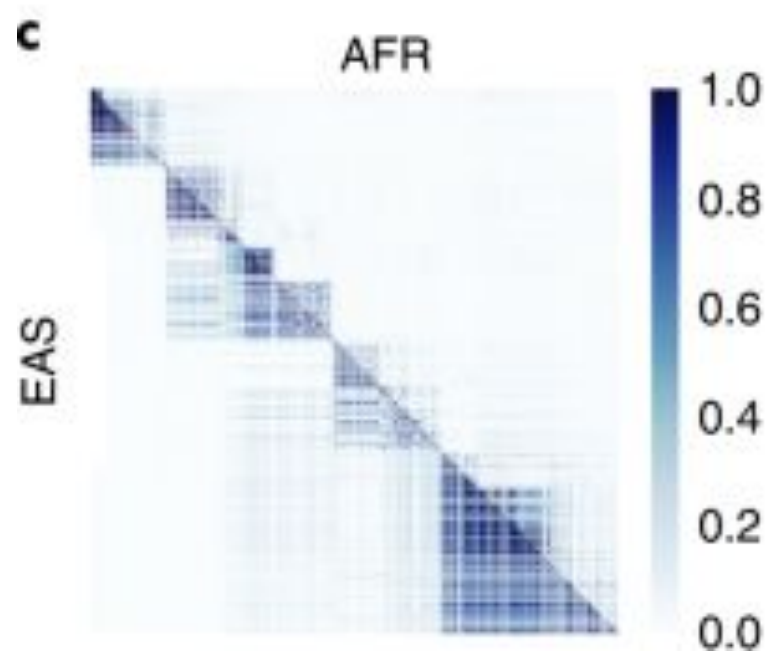
**a**



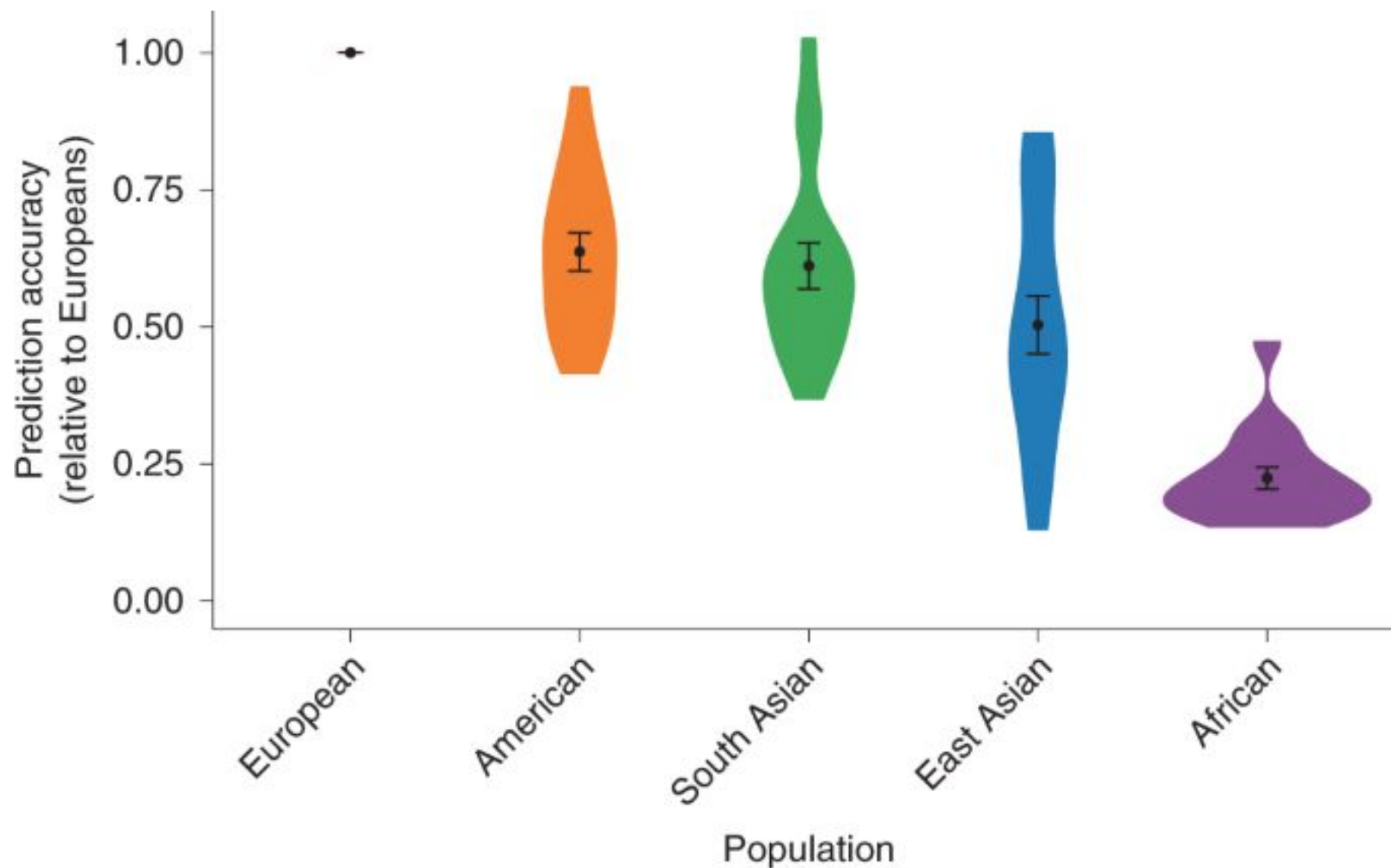
**b**



# Feature shift



Less data on Africans + feature shift  
= Less accurate predictions



# Group heterogeneity affects models

异质性

- Ex: accuracy differs when detecting gender in whites and blacks
  - Models trained on images of Caucasians may work poorly on African-Americans
  - Accuracy may improve if distributions similar
- Ex: detecting lesions on skin 病变
  - Dermatologists and AI, trained on/work with many white patients
    - Might not spot lesions in black patients
    - Rashes may look purple, or not readily apparent on non-white skin
- Use Demographic features/Polygenic risk scores to adjust for predictions

皮肤科医生

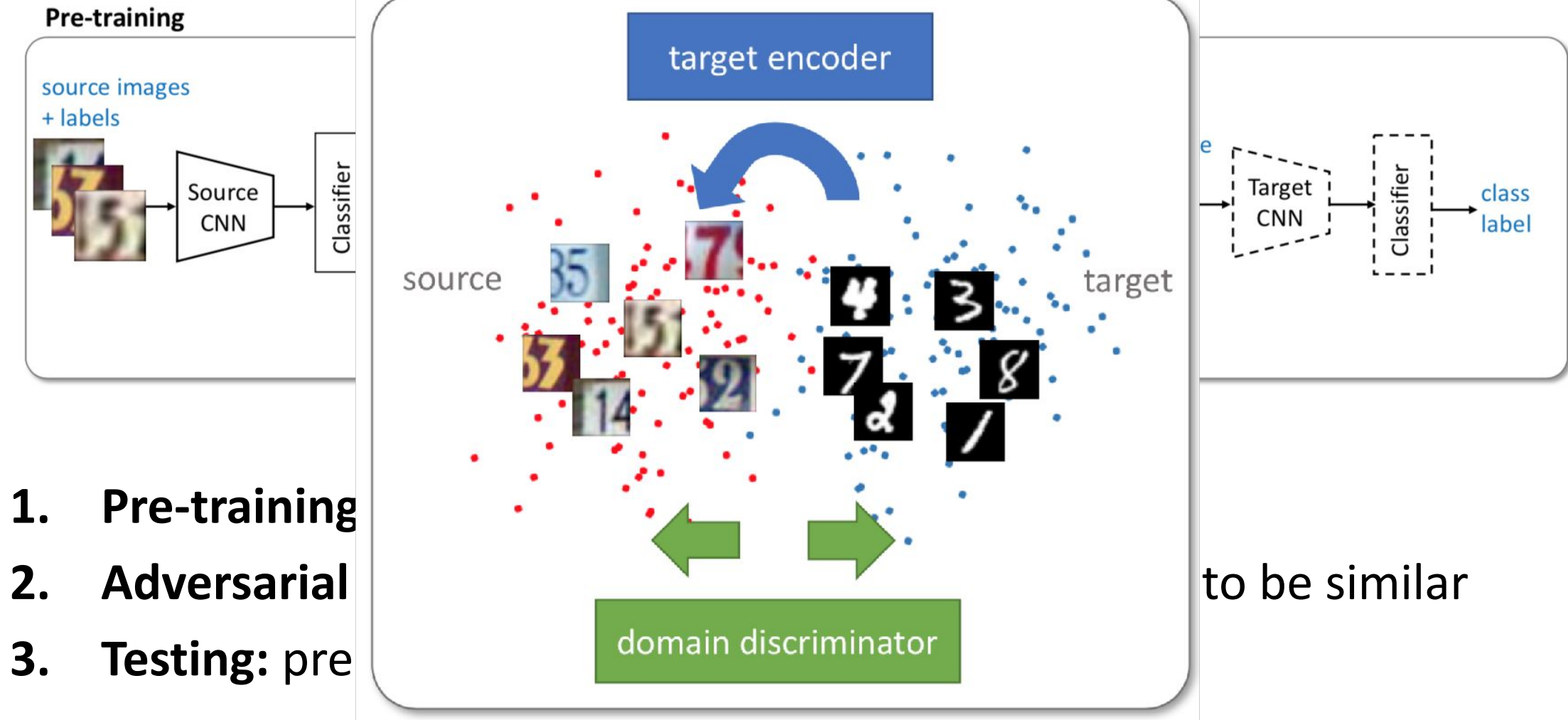
**Table I.** Images of different skin phototypes in dermatology textbooks

Textbook	Dark	Light	Indeterminate	Total	Dark skin images
Bologna	254	1011	61	1326	19%
Freedberg	240	1339	67	1646	15%
Rook	178	1255	79	1522	12%
Fitzpatrick 5th	97	721	39	857	11%
Fitzpatrick 4th	73	602	26	701	10%
Sauer's	57	550	8	615	9%
Habif	36	944	32	1012	4%

*Dark*, Skin phototype V-VI; *light*, skin phototype I-IV.

Adamson and Smith, 2018  
Gorbatenko-Roth et al., 2019

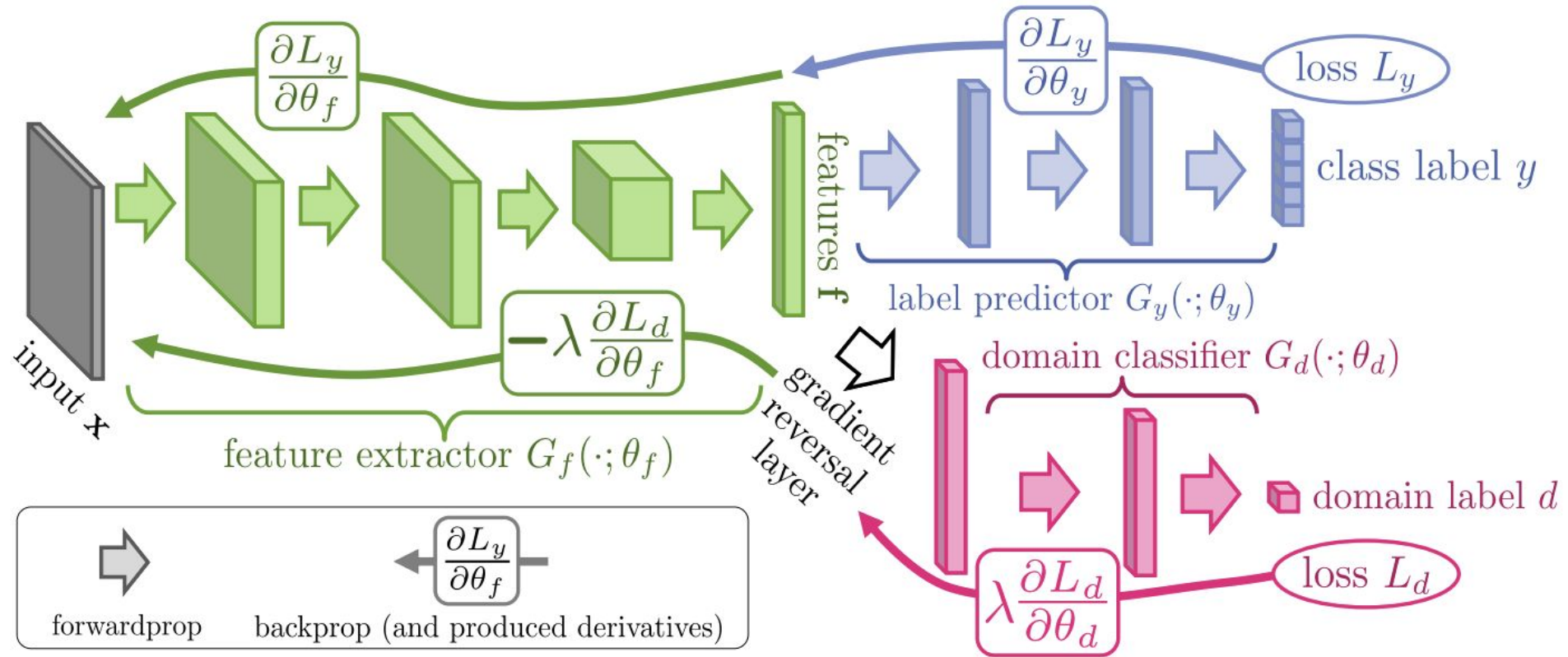
# Adversarial Discriminative Domain Adaptation



1. Pre-training
2. Adversarial
3. Testing: pre

Convert new domain to old domain and train model



















# ADDA Pipeline





## Digits adaptation



Method	MNIST → USPS	USPS → MNIST	SVHN → MNIST
	   →   	   →   	   →   
Source only	$0.752 \pm 0.016$	$0.571 \pm 0.017$	$0.601 \pm 0.011$
Gradient reversal	$0.771 \pm 0.018$	$0.730 \pm 0.020$	$0.739$ [16]
Domain confusion	$0.791 \pm 0.005$	$0.665 \pm 0.033$	$0.681 \pm 0.003$
CoGAN	$0.912 \pm 0.008$	$0.891 \pm 0.008$	did not converge
ADDA (Ours)	$0.894 \pm 0.002$	$0.901 \pm 0.008$	$0.760 \pm 0.018$

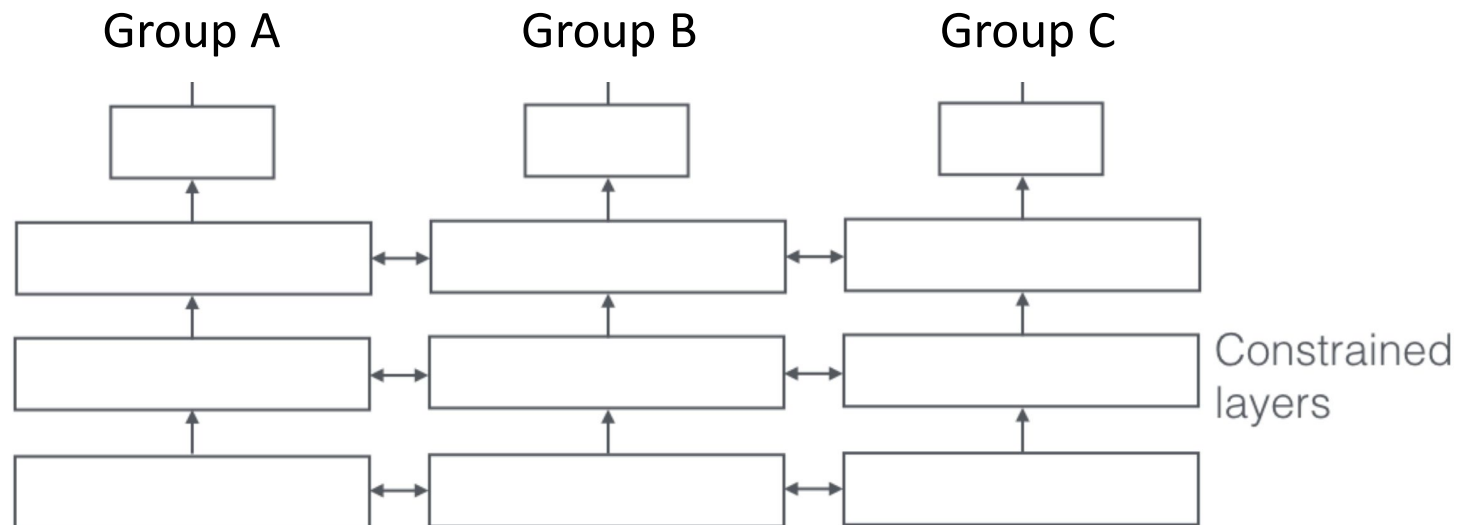
# Applications to AI Fairness

- If source data is dominated by a particular group
  - Adversarial approach makes feature distributions similar
  - Might improve fairness, but not its explicit design
  - Multi-task learning (next slide) or model invariance may be applied in tandem
- Advantages of this method
  - Works for unknown/unlabeled protected groups
  - Reduces overfitting (model all protected groups at once)
  - Data agnostic



# Multi-Task Learning

- Apply multi-task learning to find both invariant and domain-specific features (Oneto et al., 2018)  
<https://dl.acm.org/doi/10.1145/3306618.3314255>
  - Skin example mentioned previously demonstrates difficulties of domain-invariant models
  - Method assumes we know protected class labels
  - jointly learns a shared model between the groups as well as a specific model per group

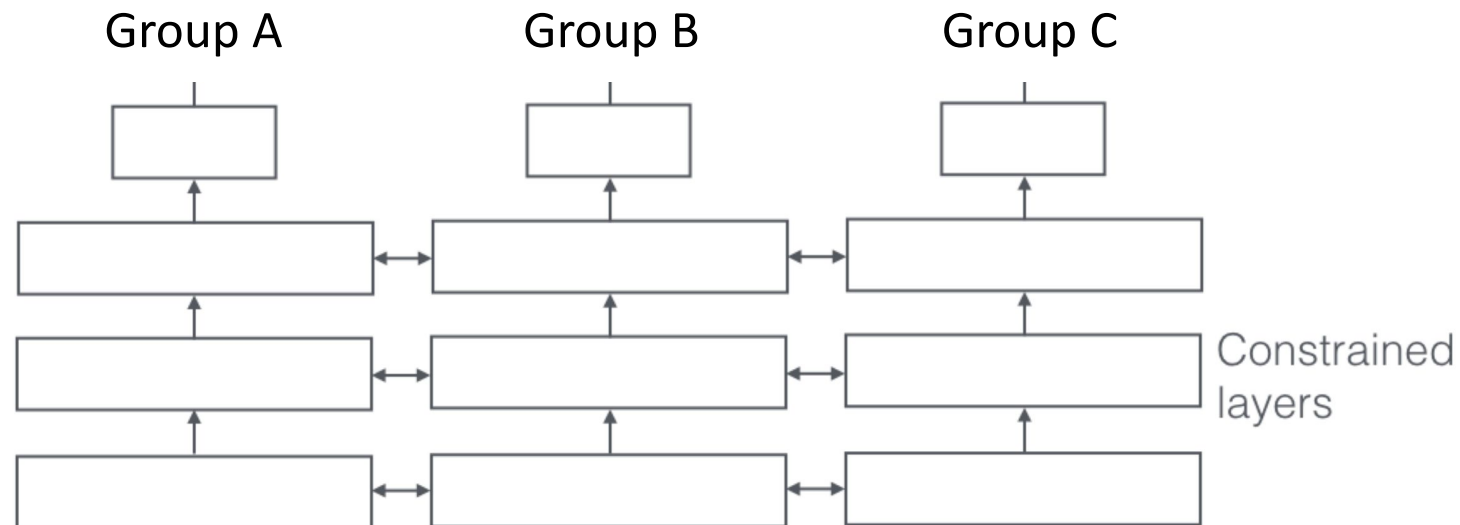


# Multi-Task Learning

- Advantage:

<https://dl.acm.org/doi/10.1145/3306618.3314255>

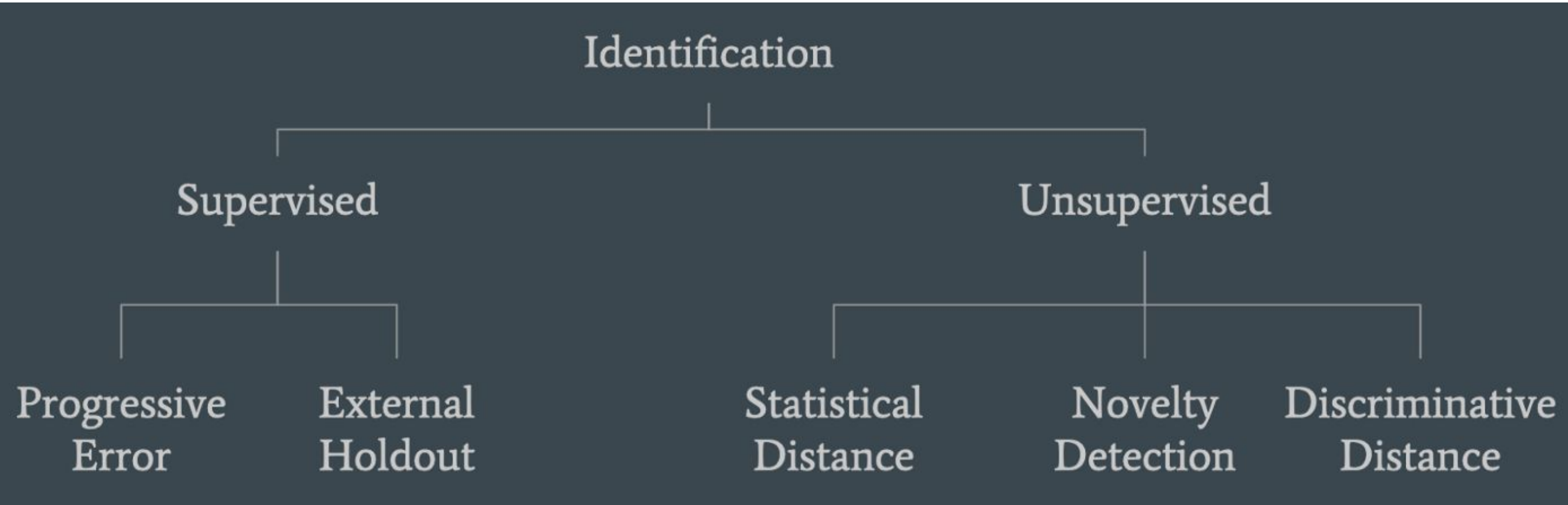
- May work better for small protected groups
- Aims to improve accuracy in all groups rather than overall accuracy



# Some issues of domain adaptation

- What subspace is domain adaptation needed most?
  - E.g., gray vs brown hair = domain, will not affect gender detection
  - Skin color *\*is\** important
- Could hand-label several features in data subset, audit each feature ([Multiaccuracy](#)-like approach)
  - Kicks can down the road: have we labeled all features? Is model fair or is auditing not separating features?
- What to do when domains are very different?
  - First we should define domain distance

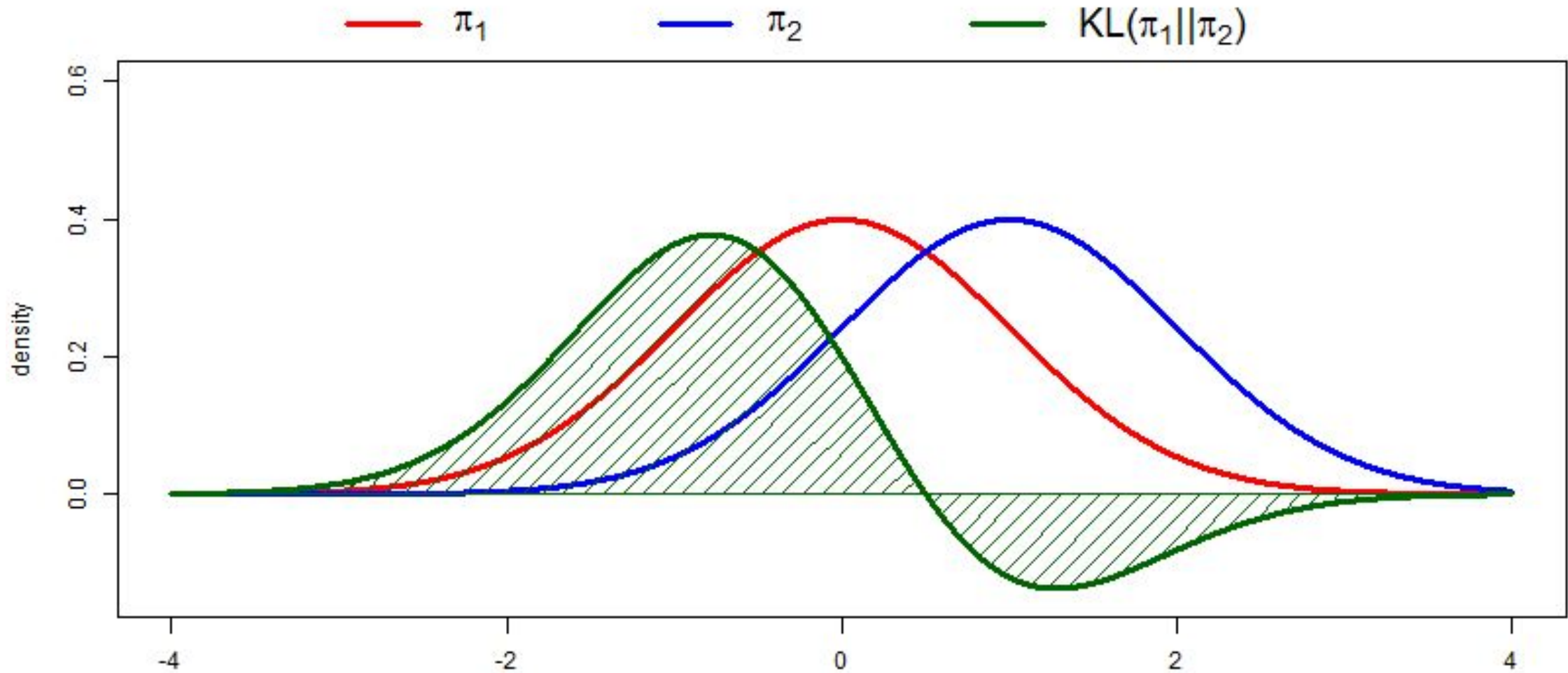
# When does domain adaptation appear?



Progressive error refers to the gradual increase in prediction error when a model trained on a source domain is applied to a target domain. This happens because the model was optimized for the source distribution, but as the data gradually shifts (e.g., over time or in different environments), the performance degrades.

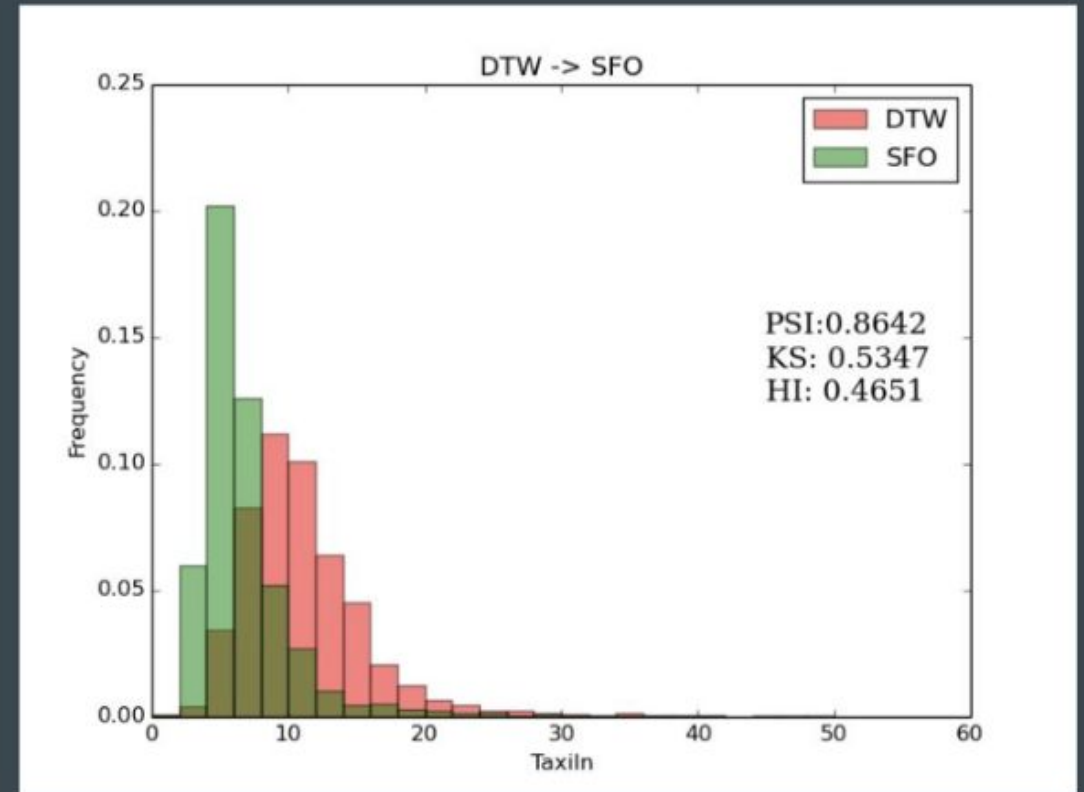
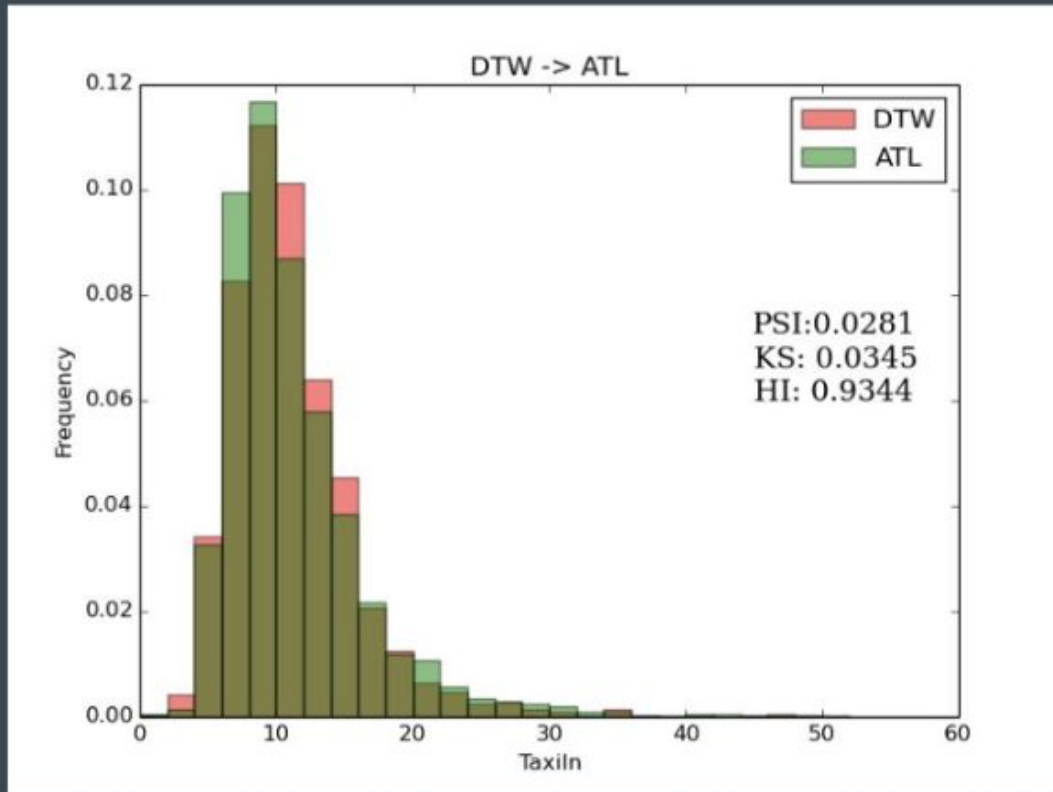
Discriminative distance measures how well a classifier can distinguish between the source and target domain samples. If a classifier can easily separate the two distributions, it means the domains are very different, indicating a large domain shift.

# Unsupervised: statistical distance



Statistical distance: how different are the distributions?

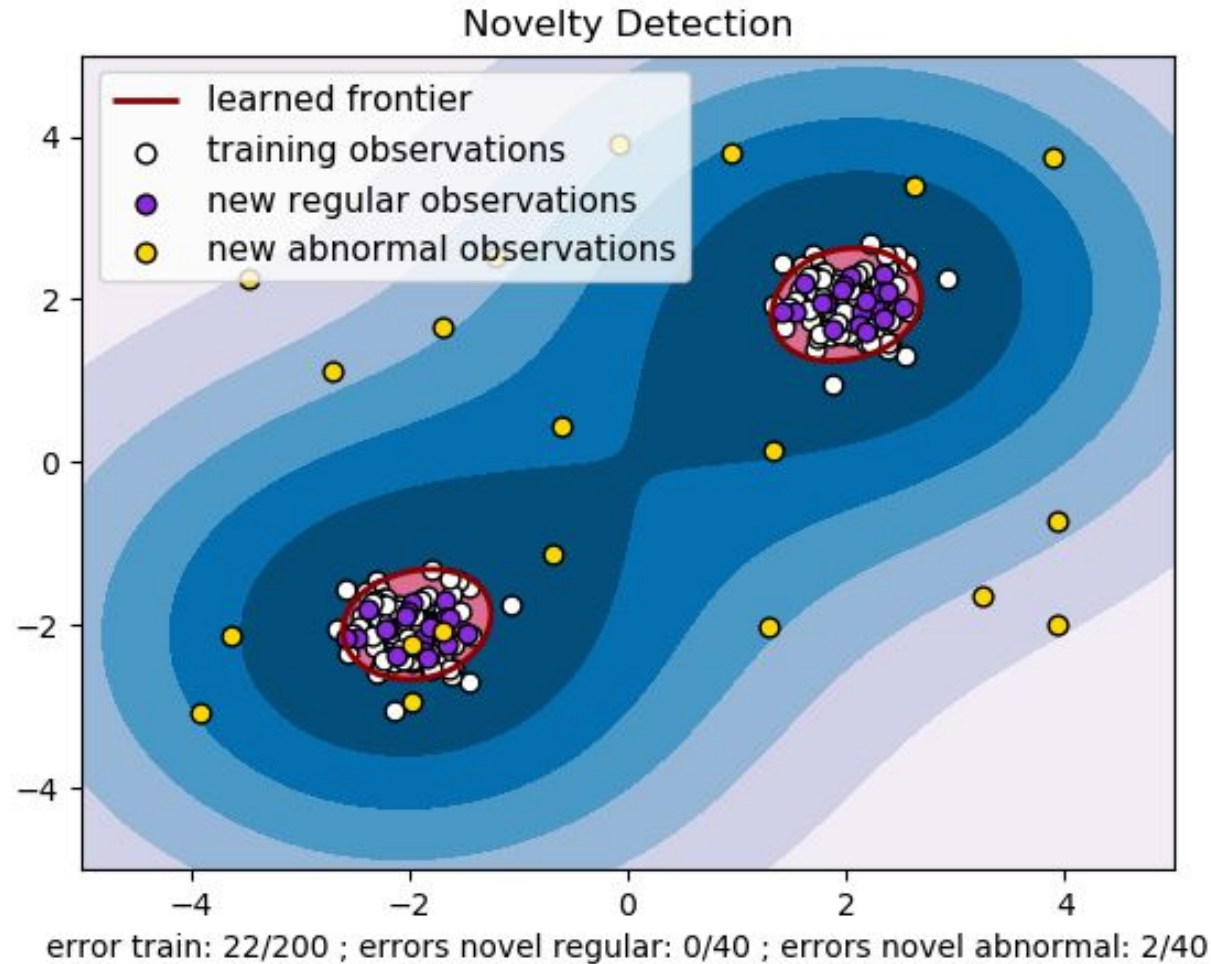
# Alternative metrics (there are many)



- Population Stability Index (PSI)
- Kolmogorov-Smirnov statistic

- Kullback-Leibler divergence
- Histogram intersection

# Unsupervised: novelty detection



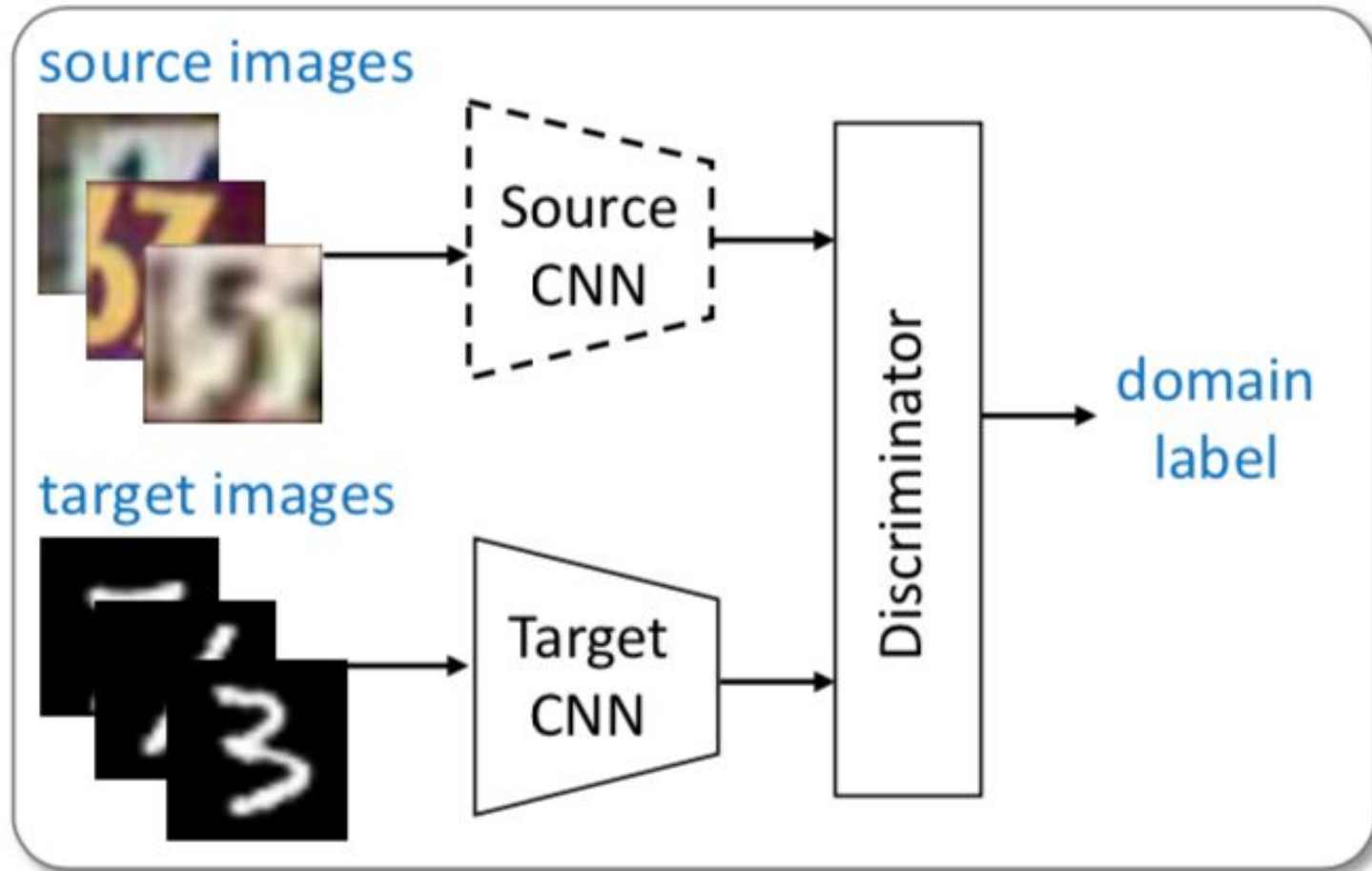
# Novelty detection

- Examples: one-class support vector machine
- Find outlier (what lies outside of training data)
- Better method when data is high-dimensional and sparse
- But outliers are not new domains (could be noise)



# Unsupervised: Discriminative Distance

- Train classifier to distinguish source, target domain
- Target = test data
- Discrimination error = how distinguishable these are (proxy of domain distance)



# Discriminative Distance: (dis)advantages

- Advantages:
  - Works well in high-dimensional, sparse data
  - If NN discriminator cannot distinguish, neither should any sophisticated model
- Disadvantages:
  - Easy to distinguish ! = new domain! (e.g., photos with/without a stop sign)



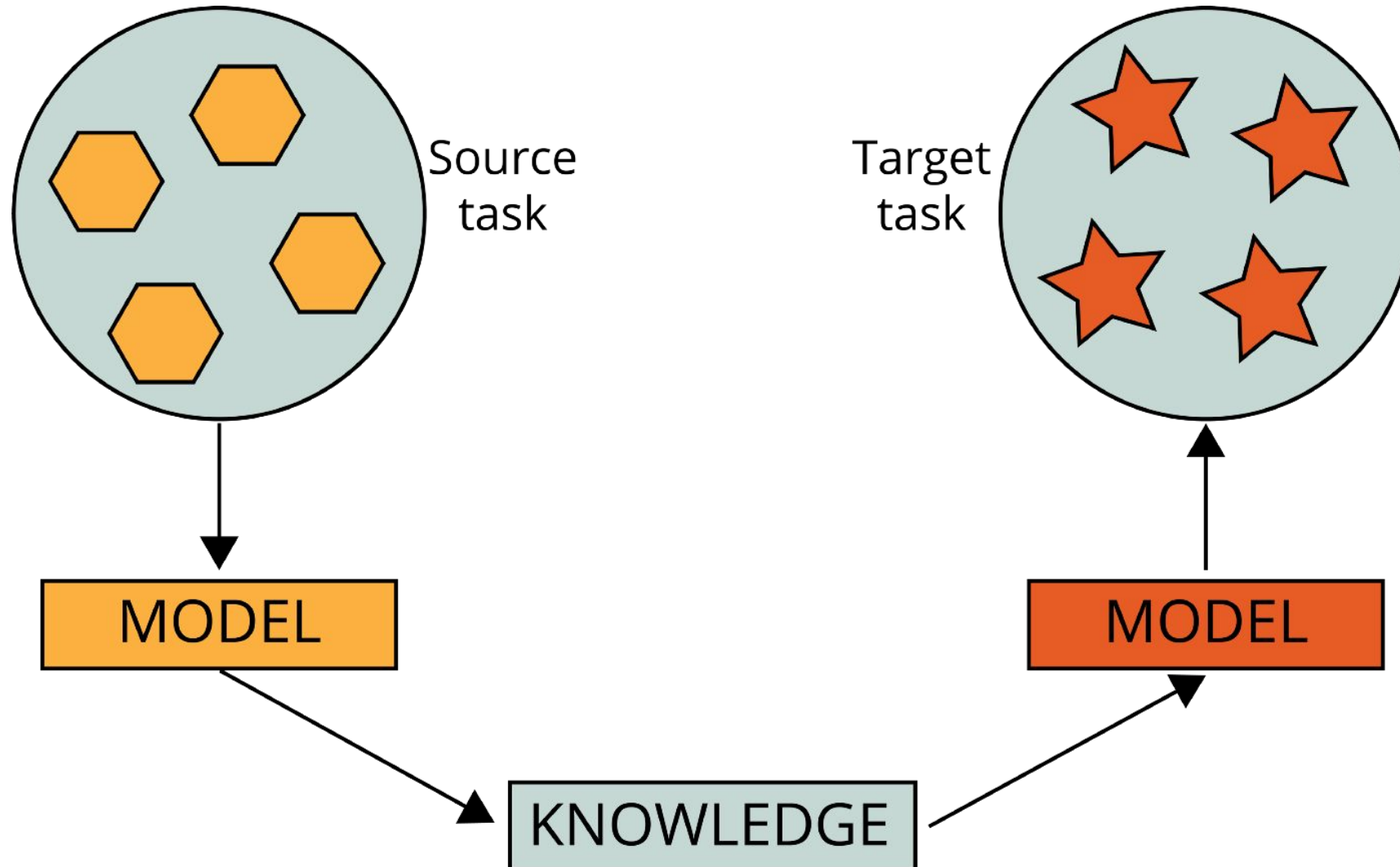
# Supervised: external holdout

- Validation set: by definition same distribution as training data
- Test/target set: unknown distribution/covariate shift
- Error in validation set < target/test set = domain shift
- Degree of error = how different are datasets
- What if domain is very different? → limitation of Domain Adaptation

# Assume domains are too different (breaks down) what do we do?

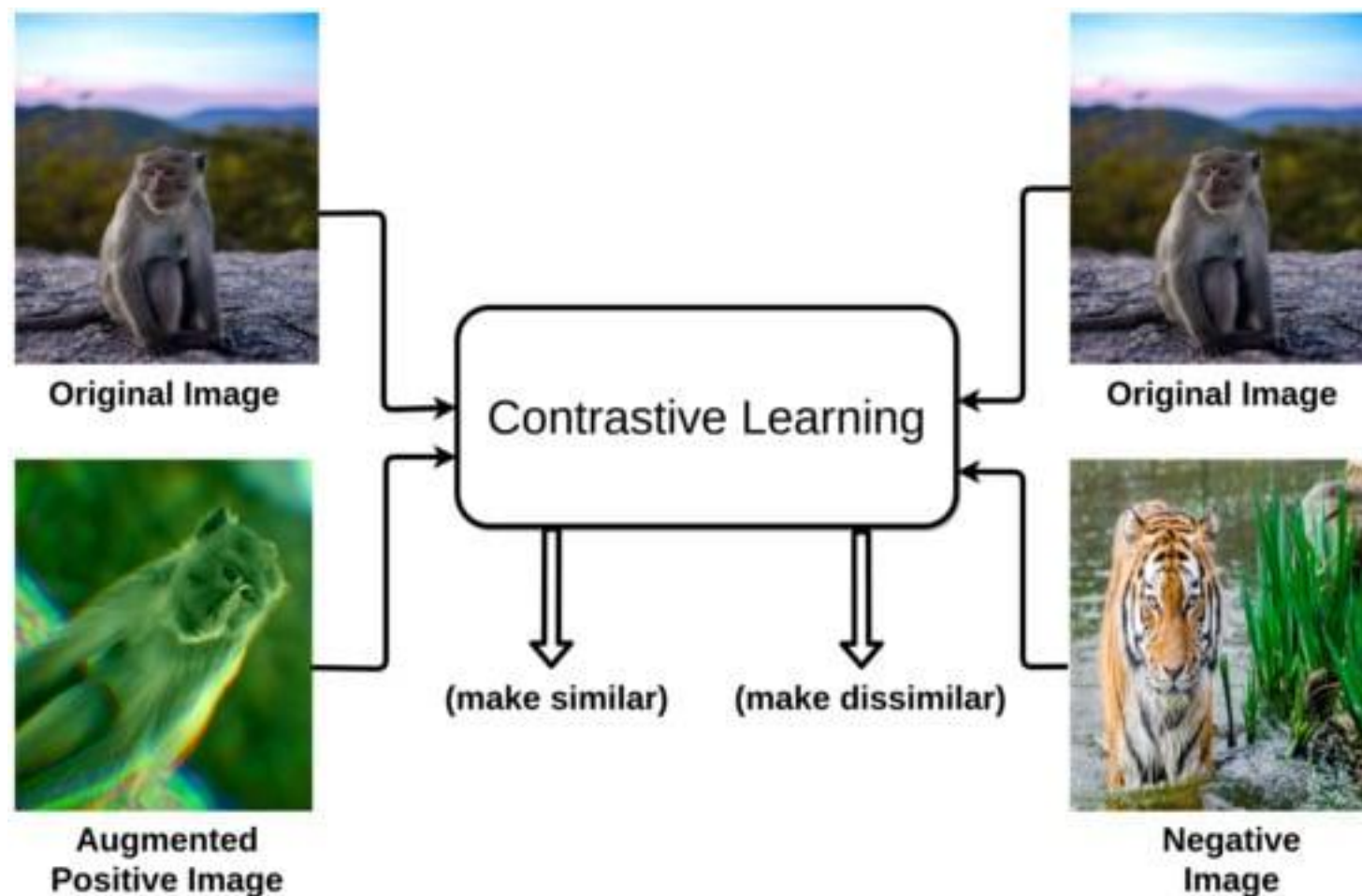
- Examples:
  - Predicting if Ramen is spoiled when model is trained on detecting age of human faces
  - Applying model trained on English data to French (languages are too different)
- Start from beginning (what basic information can source domain give us) and move forwards

# Different domains can still help!



# Example: pre-training

- Self-supervised learning helps distinguish images
- Distinguishing = learn colors, shapes, etc.
- These universal features applicable to wide range of data





# Pre-text tasks



(a)

(b)

# Examples of pre-text tasks in NLP

- Predicting words on either side of current word
- Sentence permutation



# Why this is helpful

- With a large enough source domain, we can learn basic features (e.g., object detection)
- With semi-similar domain, more features may be relevant
- Few-shot learning can tailor universal feature to specific task

Few-shot learning is a machine learning approach where a model learns to make accurate predictions with only a small number of labeled examples per class.

How It Works:

Meta-Learning (Learning to Learn):

The model is trained on multiple small tasks to generalize well to new tasks with limited data.

Embedding-Based Approaches:

The model learns a feature space where similar samples are closer together, making classification easier even with few examples.

Data Augmentation:

Techniques like synthetic data generation or adversarial learning help the model generalize from few samples.

# Summary of what we discussed

- Costs and rewards of fairness
- When domain adaptation can help or hurt
- Where domain adaptation is needed
  - Computer vision (e.g., Gender Shades)
  - Other applications (e.g., polygenic risk scores)
- Domain adaptation Methods
  - ADDA
  - Multi-task learning
- Measuring domain distance
- What to do when domains are too different

# Future work

- We give an outline, but do not fully flesh out:
  - Distinguishing domains
  - Applying domain adaptation to model fairness
  - Finding domains we can easily adapt to
- Domains are helpful, but no concrete work testing how it stacks up to fairness models