Distribution Shifts And How We Can Learn from Them

Sean Kulinski

Probabilistic and Understandable Machine Learning Lab
- Purdue University -

View of the Next Hour:

- Part 1: Background on distribution shifts
    - What is distribution shift?
    - Why are distribution shifts (currently) problematic?
- Part 2: Detecting and understanding distribution shifts
    - So what are we doing about distribution shifts?
    - Looking forward, can we utilize distribution shifts to help us learn better?

About me :)

- Ph.D. Student in Computer Engineering @ Purdue University
- Belong to the Probabilistic and Understandable Machine Learning Lab lead by [Dr. David Inouye]
- Outside of research, I enjoy:
  - 🥾 hiking/backpacking,
  - 🚵 mountain biking
  - 🗣️ spending time with friends and family
  - ⚒️ figuring out how things work

My partner and our dog

PUML Research Lab Group

PURDUE
UNIVERSITY

My research path:

- My main research interest is:

"How can we build generalizable Machine Learning models for deployment to dynamic environments seen in the wild?"

Research Assistantship @ Purdue (Lead By Dr. David Inouye)
Fall '19

Research Intern @ Lawrence Livermore National Lab (Lead by Bhavya Kailkhura)
Summer '19

ML Scientist @ AbbVie (Lead by David Masica)
Fall '21 – Spring '22

Data Scientist Intern @ Microsoft365 Research (Lead By Yi-Cheng Pan)
Summer '22

Research Intern @ Microsoft365 Research (Lead By Ankur Mallick and Kevin Hu)
Summer '22

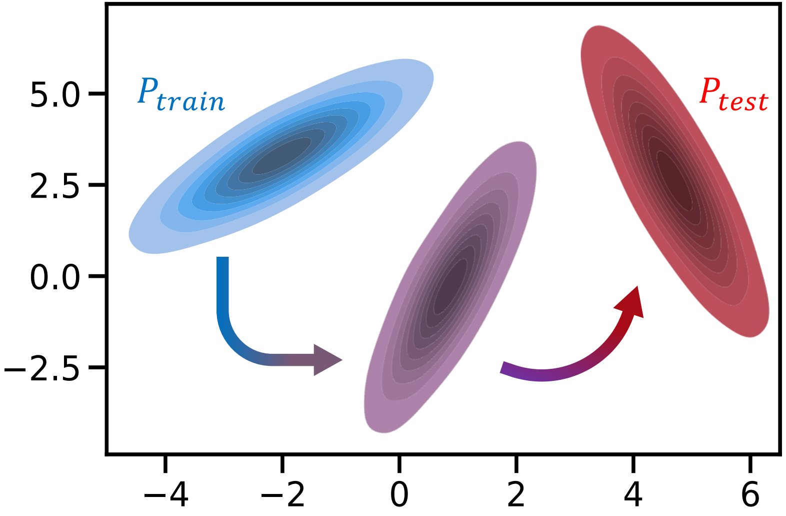
Part 1:
What are Distribution Shifts?

How the real world breaks fundamental ML assumptions.

A distribution shift is when a data distribution changes from what is expected

- In machine learning, a distribution shift is when a **testing distribution** no longer matches the **training distribution**

$$P_{test}(\boldsymbol{x}) \neq P_{train}(\boldsymbol{x})$$



Most ML assumes train/test data distributions match

- Fundamental to most ML is the

  i.i.d. assumption:

  1. Independent: All samples $\boldsymbol{x}$ are unrelated to each other

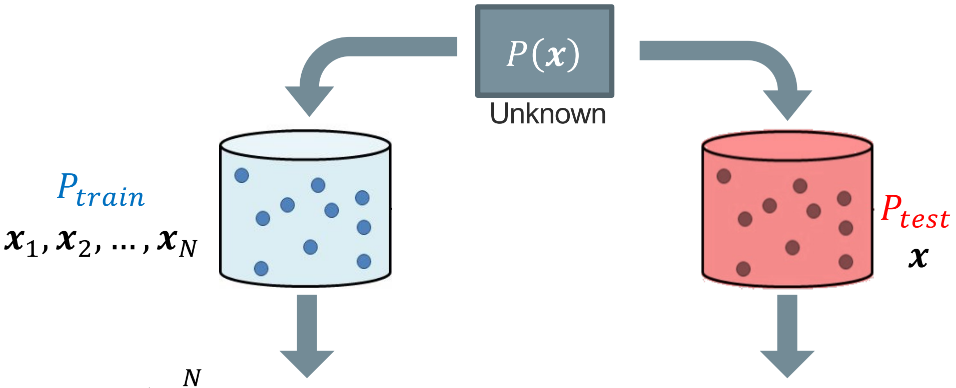
     $$P(\boldsymbol{x}_i \mid \boldsymbol{x}_{i'}) = P(\boldsymbol{x}_i) \;\; \forall i \neq i'$$

  2. Identically Distributed: All samples $\boldsymbol{x}$ come from the same distribution

     $$P_{train}(\boldsymbol{x}) = P_{test}(\boldsymbol{x})$$

- The i.i.d. assumption allows our ML model h to generalize to P_{test}

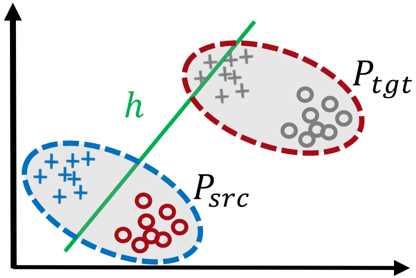
Typical Assumed ML Paradigm



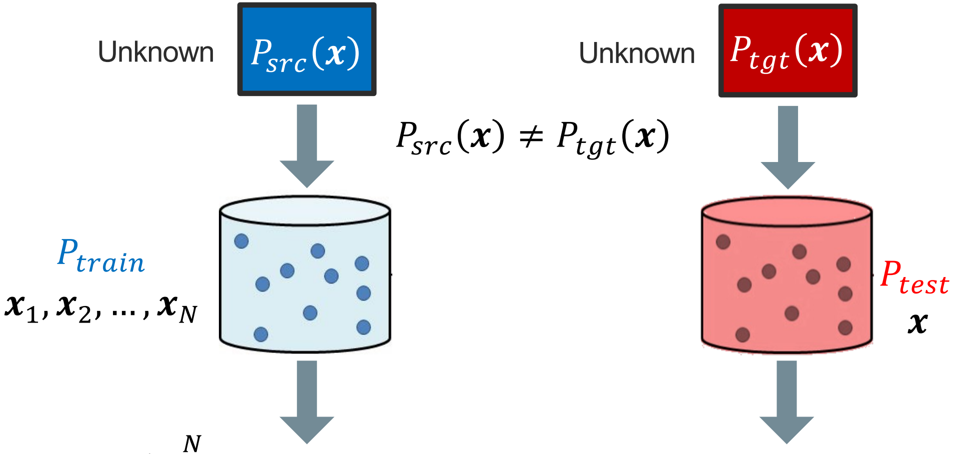
$$\min_h \frac{1}{N}\sum_{n}^{N}\left[\mathcal{L}\left(h(\boldsymbol{x}_n), \underbrace{f(\boldsymbol{x}_n)}_{y_n}\right)\right] \approx \mathbb{E}_{P_{test}}\left[\mathcal{L}\left(h(\boldsymbol{x}), \underbrace{f(\boldsymbol{x})}_{y}\right)\right]$$

Figure Inspired from Figure 4.3, of [2]

Distribution shift violates this core assumption in ML

- Distribution shift usually breaks the identically distributed assumption
- Under distribution shift, the patterns learned by h might not hold under $P_{tgt}(\boldsymbol{x})$
- 

~~Typical Assumed~~ A Shifted ML Paradigm



$$\min_h \frac{1}{N}\sum_n^N [\mathcal{L}(h(\boldsymbol{x}_n), f(\boldsymbol{x}_n))] \; \xcancel{\approx} \; \mathbb{E}_{P_{test}}[\mathcal{L}(h(\boldsymbol{x}), f(\boldsymbol{x}))]$$

PURDUE UNIVERSITY

Distribution shifts are classically broken down to three types

- In a supervised regime, we can write the joint distribution of data and labels as:

$$P(\boldsymbol{x}, y) = P(\boldsymbol{x}|y)P(y) \quad \text{-or-} \quad P(\boldsymbol{x}, y) = P(y|\boldsymbol{x})P(\boldsymbol{x})$$

- Covariate Shift: $P_{test}(y|\boldsymbol{x}) = P_{train}(y|\boldsymbol{x})$, but $P_{test}(\boldsymbol{x}) \neq P_{train}(\boldsymbol{x})$
  - Ex: $P_{test}(x)$ has more people over 60, but the per-person probability of polio has not changed
- Label Shift: $P_{test}(\boldsymbol{x}|y) = P_{train}(\boldsymbol{x}|y)$, but $P_{test}(y) \neq P_{train}(y)$
  - Ex: Everyone in P_{test} has been vaccinated. So, similar people still get polio, but it is less frequent
- Concept Drift: $P_{test}(y) = P_{train}(y)$, but $P_{test}(\boldsymbol{x}|y) \neq P_{train}(\boldsymbol{x}|y)$
  - Ex: Polio has mutated in P_{tgt} to affect younger instead of older people, but the *total* risk is the same

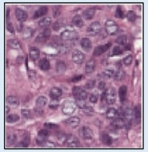
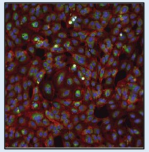
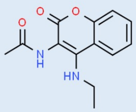
PURDUE
UNIVERSITY

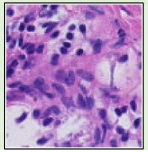
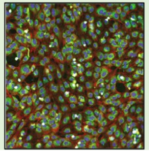
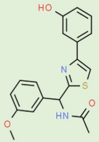
Distribution shifts are ubiquitous

- Any changes in a current data generating environment can cause shifts

- Applying a model to a new domain is almost always a shift

Dataset	iWildCam	Camelyon17	RxRx1	FMoW	PovertyMap	GlobalWheat	OGB-MolPCBA	CivilComments	Amazon	Py150
Input (x)	camera trap photo	tissue slide	cell image	satellite image	satellite image	wheat image	molecular graph	online comment	product review	code
Prediction (y)	animal species	tumor	perturbed gene	land use	asset wealth	wheat head bbox	bioassays	toxicity	sentiment	autocomplete
Domain (d)	camera	hospital	batch	time, region	country, ru/ur	location, time	scaffold	demographic	user	git repo

Source example								What do Black and LGBT people have to do with bicycle licensing?	Overall a solid package that has a good quality of construction for the price.	`import numpy as np`
										…
										`norm=np.___`

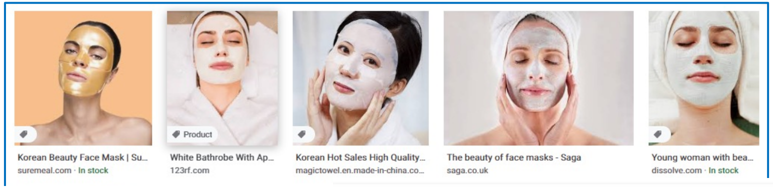
Target example								As a Christian, I will not be patronizing any of those businesses.	I *loved* my French press, it's so perfect and came with all this fun stuff!	`import subprocess as sp`
										`p=sp.Popen()`
										`stdout=p.___`

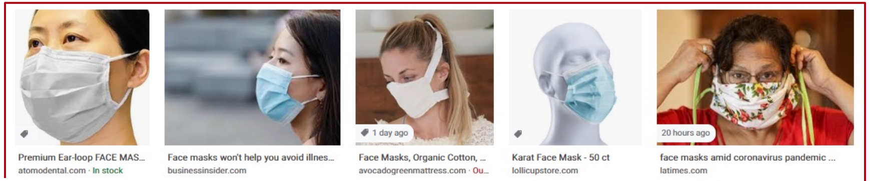
Exemplar Real-World Distribution Shift datasets from Stanford WILDS datasets overview [1]

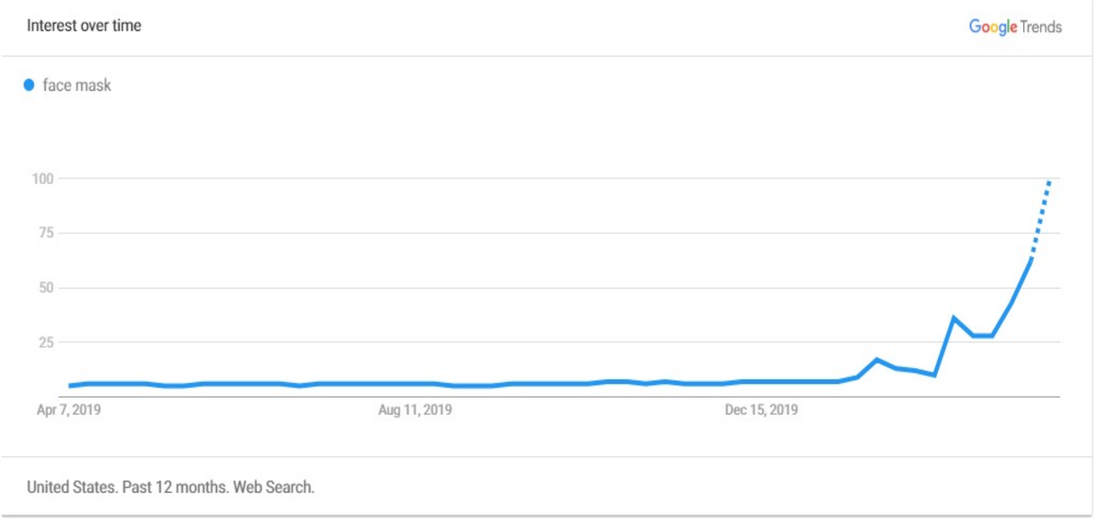
PURDUE UNIVERSITY

Example: Google Search Results

Face mask vs. Face mask?







PURDUE
UNIVERSITY

Part 2:
Living With Distribution Shifts

Detecting the problem and trying to answer *what happened?*

Detecting distribution shifts – common methods

- Distribution Shift detection answer the binary question: “Has a shift occurred?”
- Detecting distribution shift is a well-studied topic [3], most methods involve either:

1. Statistical Hypothesis testing between P_{src} and P_{tgt}:
   $\phi(\hat{P}_{src}, \hat{P}_{tgt}) \geq \epsilon$, ϕ := statistical divergence function (e.g., KL-divergence) and $\hat{P}$:= a density model of of the data (e.g., a normalizing flow)

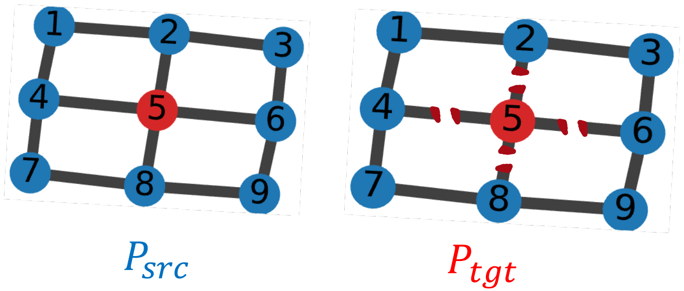
2. Training a domain classifier model f to classify between x_{src} and $\hat{x}_{tgt}$:
   $\mathbb{E}_{\boldsymbol{x} \sim P_{tgt}}[f(\boldsymbol{x})] \geq \epsilon$, $\hat{x}_{tgt}$:= an *estimate* of what samples from P_{tgt} will look like

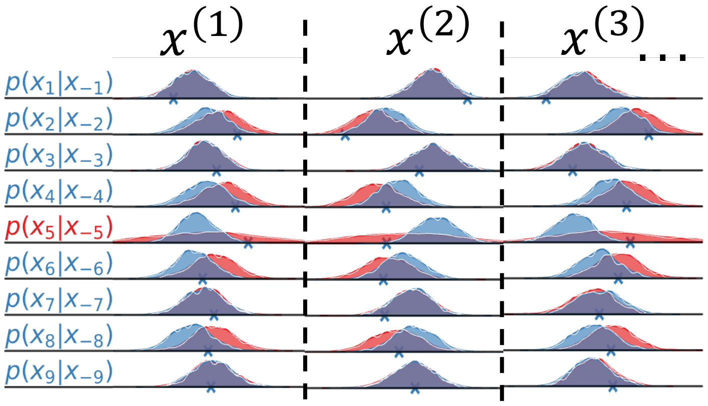
PURDUE UNIVERSITY

We can use feature shift detection to localize the problem to specific features

- To detect feature shift [4], we define a conditional distribution hypothesis test:
  - $H_O: \forall \boldsymbol{x}_{-j} \in \mathcal{X}_{-j},\ \hat{P}_{src}(\boldsymbol{x}_j|\boldsymbol{x}_{-j}) = \hat{P}_{tgt}(\boldsymbol{x}_j|\boldsymbol{x}_{-j})$
  - $H_A: \exists \boldsymbol{x}_{-j} \in \mathcal{X}_{-j},\ \hat{P}_{src}(\boldsymbol{x}_j|\boldsymbol{x}_{-j}) \neq \hat{P}_{tgt}(\boldsymbol{x}_j|\boldsymbol{x}_{-j})$
- Feature shift can happen in two stages:
  - Detection: Do the conditional distributions of $\hat{P}_{tgt}$ differ from the conditional distribution $\hat{P}_{src}$?
  - Localization: Which feature(s) have caused this difference?

Feature Shift Toy Problem





$P(x_j|\,x_{-j})$ conditional distributions for $P \in \hat{P}_{src}, \hat{P}_{tgt}$

PURDUE UNIVERSITY

Feature Shift Detection is fast with Fisher divergence

- Fisher divergence test statistic based on the score function, $\psi \coloneqq \nabla_x \log\big(p(x)\big)$

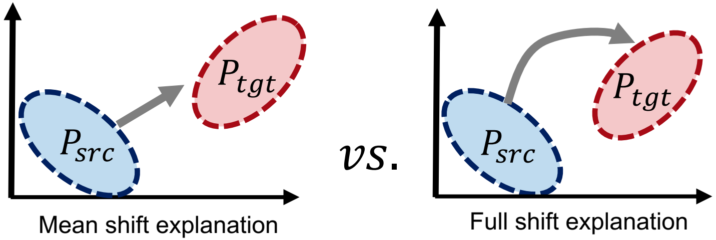
$$\phi_{Fisher}(p,q) \triangleq \mathbb{E}_{p(\boldsymbol{x})+q(\boldsymbol{x})}[\|\psi(\boldsymbol{x};p)-\psi(\boldsymbol{x};q)\|^2] = \mathbb{E}_{p(\boldsymbol{x})+q(\boldsymbol{x})}\left[\left\|\nabla_x \log\frac{p(\boldsymbol{x})}{q(\boldsymbol{x})}\right\|^2\right]$$

- Can compute multiple feature test statistics simultaneously
  - $\phi_{Fisher}\left(p_{x_j|x_{-j}}, q_{x_j|x_{-j}}\right) = \mathbb{E}_{p(\boldsymbol{x})+q(\boldsymbol{x})}\left[\big(\psi(\boldsymbol{x};p)-\psi(\boldsymbol{x};q)\big)^2\right]_j$
  - Only a **single forward and backward pass** is needed to compute all conditional score functions, which is already done when updating a density model, $\hat{P}$
- Feature Shift tells us: “Has a shift occurred?” + “What set of features shifted?”

PURDUE UNIVERSITY

A distribution shift has been detected…now what? We need to know more to respond effectively

- Problem: Once a shift has been detected, an operator needs to figure out what has changed in order to effectively respond
- Current simple approach: See how the means have shifted, $\mu_{src} - \mu_{tgt}$
  - Gives a rough approximation of shift
  - However, this can miss important information:

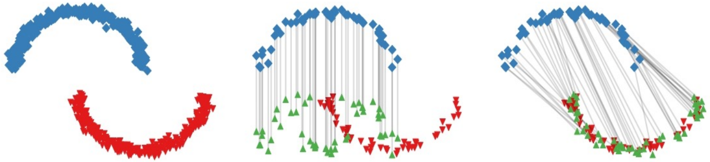


- Our goal: Aid the operator by explaining how P_{src} shifted to P_{tgt}

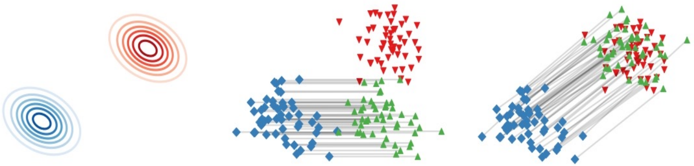
PURDUE UNIVERSITY

Distribution shifts can be explained by hypothesizing how to map P_{src} to P_{tgt}

- Given two distributions P_{src}, P_{tgt}:
  - a transport map $T(\cdot)$, is a function which moves a point from P_{src} to P_{tgt}, such that $T_{\#P_{tgt}} \approx P_{src}$



- If T is interpretable, it can explain how P_{src} shifted to P_{tgt}



$T(x) = \boldsymbol{x} + [2,2]$

PURDUE UNIVERSITY

We can leverage prior Optimal Transport work to find **good** interpretable mappings

- Optimal Transport finds a minimum cost mapping T_c that aligns two distributions [10]
- By relaxing alignment and restricting our possible mappings to be interpretable we get *intrinsically interpretable transport* T_{IIT}:

$$T_{IIT} \coloneqq \operatorname{argmin}_{T \in \Omega_{int}} \mathbb{E}_{P_{train}} \left[c\big(\boldsymbol{x}, T(\boldsymbol{x})\big) \right] + \lambda\, \phi\big(P_{T(X)}, P_{test}\big)$$

where $\Omega_{int} = \{T\colon \ \mathrm{s.t.\,T\ is\ interpretable}\}, \mathrm{c}(\cdot,\cdot)$ is a cost function (e.g., ℓ_2), and ϕ is a divergence

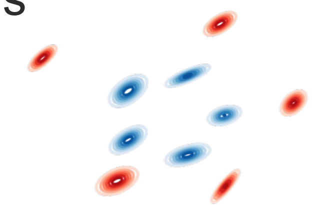
- T_{IIT} gives us a mapping which is faithful $\big(P_{T(X)} \approx P_{test}\big)$, interpretable $(T \in \Omega_{\mathrm{int}})$, and simple (minimizes a transport cost)
- Ω_{int} can be defined based on context, or one can use our pre-defined sparse-feature mappings or cluster-based mappings [5]

PURDUE UNIVERSITY

T_{IIT} can be used to gain actionable insights from explanations of complex shifts

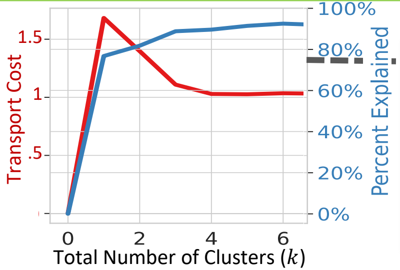
- Using our k-cluster mappings $\Omega^k_{\text{cluster}}$, we can see how heterogenous groups (clusters) moved differently under a distribution shift

$$\Omega^k_{\text{cluster}} = \{T: T(\boldsymbol{x}) = \boldsymbol{x} + [\Delta]_c\}, \text{ where } \Delta \in \mathbb{R}^{d x k}, c = [k]$$



Example of Cluster mapping / Example of Subpopulation shift

- We can use $\Omega^k_{\text{cluster}}$ to compare male and female responses to the 1994 US Census



k - Cluster Explanation (ours), $k = 4$:

$\mu_{C_M^1}$	: [Age: 35.6, Edu: 12.9, Inc: 0.00]	$\mu_{C_M^2}$	: [Age: 29.7, Edu: 8.9, Inc: 0.03]
$\mu_{C_M^1 \to C_F^1}$	: [Age: 32.1, Edu: 12.5, Inc: 0.01]	$\mu_{C_M^2 \to C_F^2}$	: [Age: 26.3, Edu: 9.0, Inc: 0.00]
$\mu_{C_M^3}$	: [Age: 56.3, Edu: 8.4, Inc: 0.13]	$\mu_{C_M^4}$	: [Age: 43.5, Edu: 12.1, Inc: 1.00]
$\mu_{C_M^3 \to C_F^3}$	: [Age: 53.7, Edu: 8.7, Inc: 0.01]	$\mu_{C_M^4 \to C_F^4}$	: [Age: 40.2, Edu: 11.9, Inc: 0.38]

Insight 1: Income is largest predictor between M and F

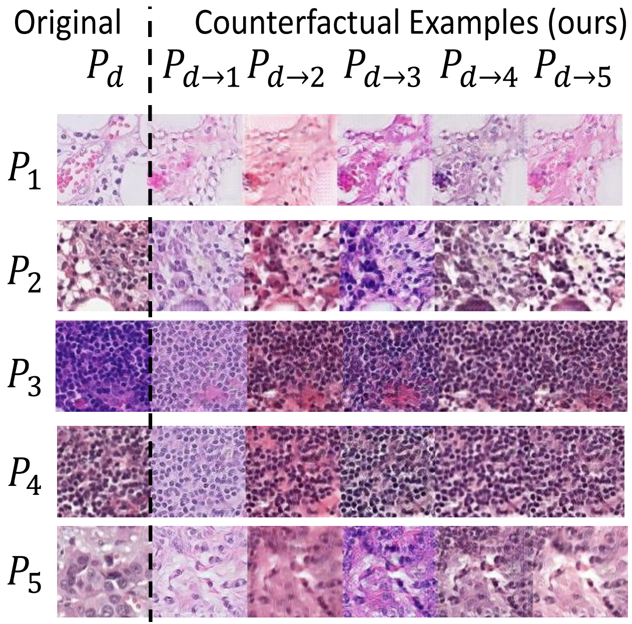
Insight 2: The income difference is largest in M_{C^4}, middle-aged adults with a bachelor's degree

Transport Maps can also explain distribution shifts in high-dimensional regimes (images)

- When raw features are not semantically meaningful, but samples are (e.g., images), we can use post-hoc methods to understand T such as:

Distributional-Counterfactuals $\coloneqq \{\boldsymbol{x}, T(\boldsymbol{x}) : x \sim P_{src}, T(\boldsymbol{x}) \sim P_{tgt}\}$

- We can use distributional-counterfactuals to explain how H&E staining of tissue samples change across multiple hospitals [6]



Using StarGAN [7] to show the difference between tissue samples across 5 hospitals

PURDUE UNIVERSITY

Take-Aways on Distribution Shifts

- Distribution Shifts are ubiquitous, complex, and problematic for ML
- To combat distribution shifts we need to:

1. Detect a shift has happened
   - Perform statistical hypothesis testing between $\hat{P}_{src}$ and $\hat{P}_{tgt}$ e.g., check for feature-shift
2. Understand what the distribution shift has changed
   - Solve for a distribution shift explanation T and see if the changes are problematic
3. React to the fix the shift
   - Possibly retrain models, fix the change in our environment, update training set, etc.

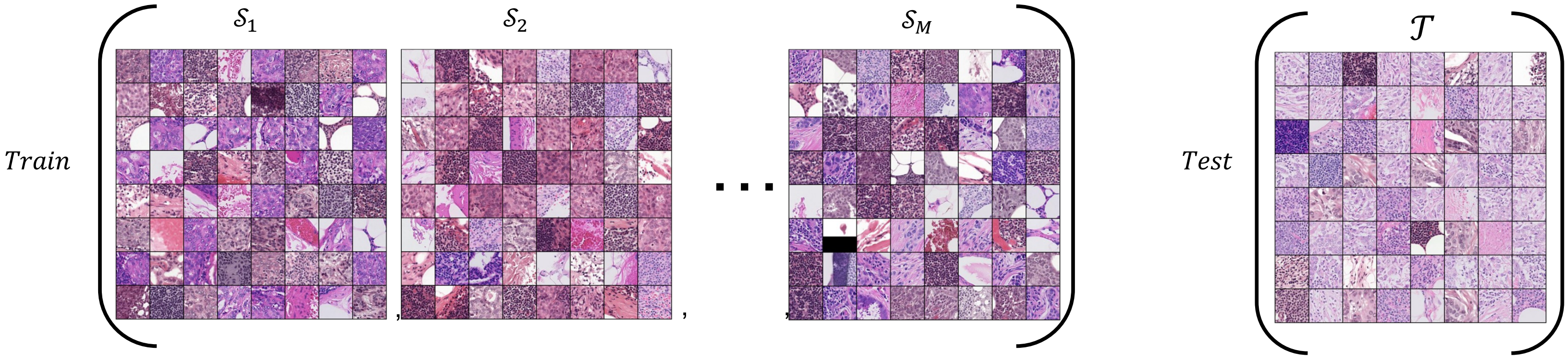
PURDUE UNIVERSITY

Part 3:
How to Avoid Problems with Distribution Shifts

Turning the problem into the solution – methods for domain generalization.

We can use sets of shifted distributions to build robust models

Given: M training domains $\mathcal{S} = \{\mathcal{S}_i |\ i = 1, \dots, M\}$ where $\mathcal{S}_i = \{(x_j^i, y_j^i, i)\}_{j=1}^{n_i}$



Goal:

- Find a model which can achieve a minimum error on an **unseen** test domain,

  $\mathcal{T} = \{x_j, y_j\}_{j=1}^{n_t}$

- $\min_h \mathbb{E}_{(x,y)\in\mathcal{S}_{test}} [\ell(h(x), y)]$ for some loss function $\ell(\cdot)$ and $P_{XY}^{test} \neq P_{XI}^{i}$

PURDUE UNIVERSITY

Images edited from [7].

Taxonomy of DG

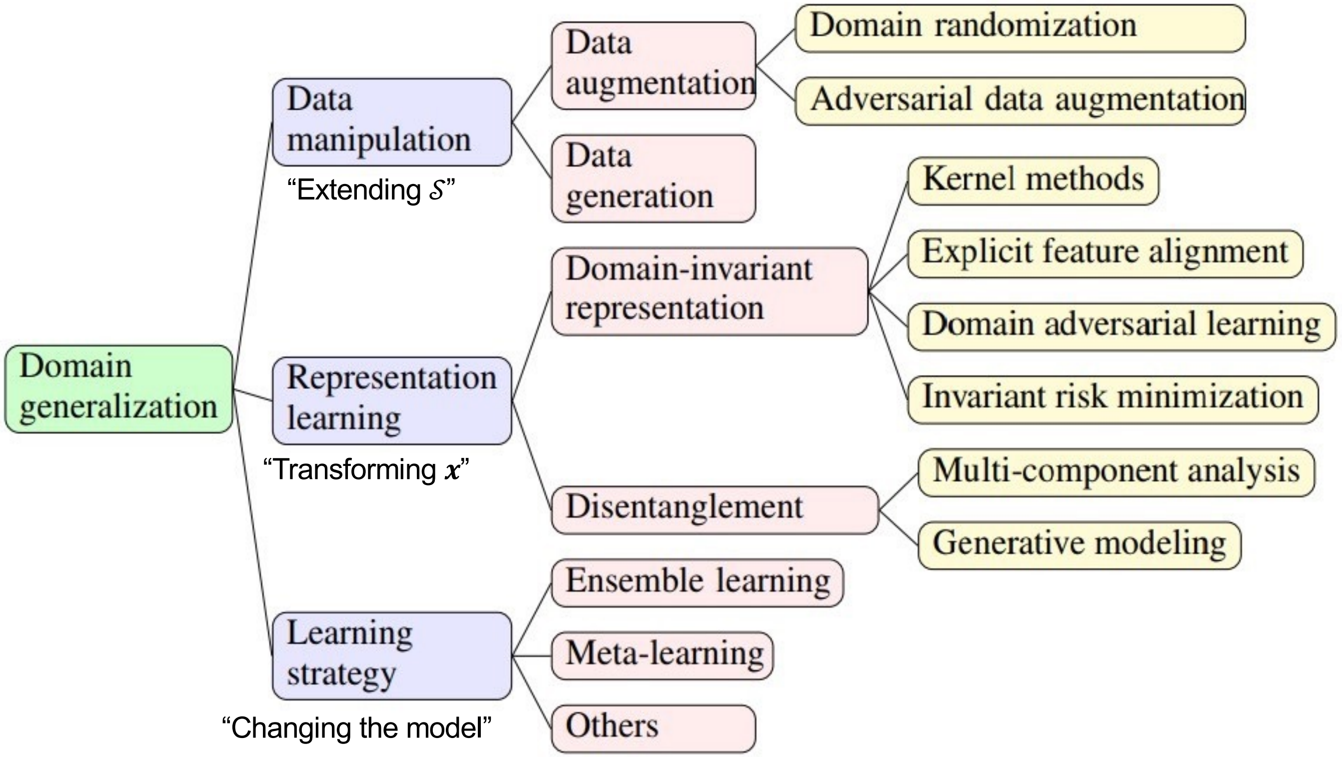
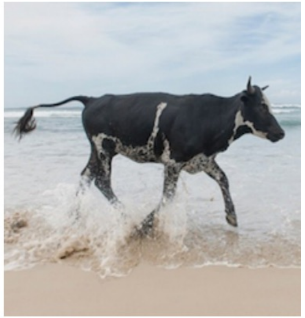


Image from [8].

An optimal representation which is invariant across domains in $\mathcal{S}$ should generalize to unseen domains

- Domain adversarial learning
  - Adversarial optimization where d discriminates the original domain of $g(\boldsymbol{x})$, and g finds a representation which aids the classifier $f\big(g(\boldsymbol{x})\big)$ while fooling the discriminator
  - $\underset{f,g}{\operatorname{argmin}}\ \underset{d}{\operatorname{argmax}} \sum_{j=1}^{M} \sum_{(\boldsymbol{x},y)\in\mathcal{S}_j} \mathcal{L}_{f,g}\big(f\big(g(\boldsymbol{x})\big), y\big) + \mathcal{L}_d\big(d\big(g(\boldsymbol{x}), j\big)\big)$
- Explicit feature alignment
  - Alignment of the domain distributions using a shared feature extractor g
  - $\underset{f,g}{\operatorname{argmin}} \sum_{i\neq j}^{M} \operatorname{dist}\Big(g_{\#}(\mathcal{S}_i), g_{\#}\big(\mathcal{S}_j\big)\Big) + \sum_{(\boldsymbol{x},y)\in\mathcal{S}_i} \mathcal{L}\big(f\big(g(\boldsymbol{x})\big), y\big),$
    where $\operatorname{dist}(\cdot,\cdot)$ is some notion of a distance or statistical divergence metric
  - Common representation functions: kernel methods, batch-instance normalization, neural networks
- Invariant risk minimization
  - Find a data representation such that the optimal classifier $f^*\big(g(\boldsymbol{x})\big)$ is the same across all environments

(A) **Cow: 0.99**, Pasture: 0.99, Grass: 0.99, No Person: 0.98, Mammal: 0.98

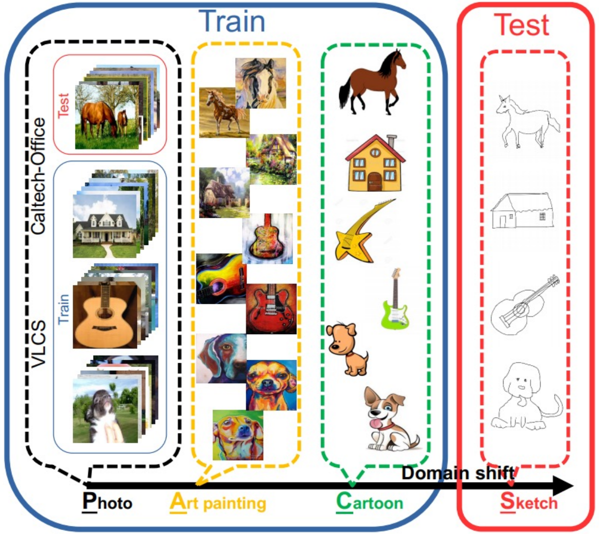


(B) No Person: 0.99, Water: 0.98, Beach: 0.97, Outdoors: 0.97, Seashore: 0.97

From: Recognition in Terra Incognita [10]

Feature-disentanglement learns both domain specific and domain-invariant representations

- Goal: learn function(s) that decompose samples into meaningful domain invariant $g_i(\boldsymbol{x})$ and domain specific features $g_s(\boldsymbol{x})$
  - $\text{argmin}_{g_s, g_i, f}\, \mathbb{E}_{\boldsymbol{x},y}\, \mathcal{L}\big(f\big(g_s(\boldsymbol{x})\big), y\big) + \lambda \mathcal{L}_{recon}([g_s(\boldsymbol{x}), g_i(\boldsymbol{x})], \boldsymbol{x}) + \lambda \mathcal{L}_{reg}\big(g_s(\boldsymbol{x}), g_i(\boldsymbol{x})\big)$
- Multi-component analysis
  - During training, learn a universal model $\theta^{(0)}$ and domain-specific models $\{\theta^{(j)}\}_{j=1}^{M}$, and for inference use functional combination of the two
  - UndoBias: SVM where $\boldsymbol{w}(\boldsymbol{x}) = \boldsymbol{w}^{(0)}(\boldsymbol{x}) + \mathbf{w}^{(\hat{j})}(\boldsymbol{x})$ where $\hat{j} \in \{1, \dots, M\}$ and is found via $\hat{j} = d(\boldsymbol{x})$, where d finds the domain which $\boldsymbol{x}$ is most likely to have come from
- Generative modeling
  - Use VAEs to find a latent space with disentangled representations of domain information, category information, and other information



From: Deeper, Broader and Artier Domain Generalization

PURDUE UNIVERSITY

References

[1] Koh, Pang Wei, et al. "Wilds: A benchmark of in-the-wild distribution shifts." *International Conference on Machine Learning*. PMLR, 2021.

[2] Chan, Stanley. “Chapter 4, Learning Theory.” *ECE 595 / Stat 598: Machine Learning*, https://engineering.purdue.edu/ChanGroup/ECE595/files/chapter4.pdf.

[3] Rabanser, Stephan, Stephan Günnemann, and Zachary Lipton. "Failing loudly: An empirical study of methods for detecting dataset shift." *Advances in Neural Information Processing Systems* 32 (2019).

[4] Kulinski, Sean, Saurabh Bagchi, and David I. Inouye. "Feature shift detection: Localizing which features have shifted via conditional distribution tests." *Advances in Neural Information Processing Systems* 33 (2020): 19523-19533.

[5] Kulinski, Sean, and David I. Inouye. "Towards Explaining Distribution Shifts." *arXiv preprint arXiv:2210.10275* (2022).

[6] Kulinski, Sean, and David I. Inouye. "Towards Explaining Image-Based Distribution Shifts." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

[7] Choi, Yunjey, et al. "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

[8] Wang, Jindong, et al. “Generalizing to Unseen Domains: A Survey on Domain Generalization.” Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, International Joint Conferences on Artificial Intelligence Organization, 2021, pp. 4627–35.

[9] Beery, Sara, Grant Van Horn, and Pietro Perona. "Recognition in terra incognita." *Proceedings of the European conference on computer vision (ECCV)*. 2018.

[10] Peyré, Gabriel, and Marco Cuturi. "Computational optimal transport: With applications to data science." *Foundations and Trends® in Machine Learning* 11.5-6 (2019): 355-607.

Thanks for listening :)

I'm happy to answer any questions you have now.

If you would prefer to chat after, just email me at: skulinski@purdue.edu , or

you may find answers/more ways to reach me on my website: seankulinski.com

PURDUE UNIVERSITY