Toward a inductive modeling language for distribution shifts

Hongseok Namkoong  
namkoong@gsb.columbia.edu  

Decision, Risk, and Operations Division, Columbia Business School

Based on joint works with Tiffany Cai, Peng Cui, Jiashuo Liu, Tianyu Wang, Steve Yadlowsky
ImageNet

Large image classification dataset: 1.2 mio training images, 1,000 image classes.

Golden retriever

![Bar chart showing error over years with AlexNet highlighted in 2012.](chart.png)
AI builds on data as infrastructure

Countries are distorted by frequency of usage. Datasets originating in the US account for the most usages (26,910).

Research by: Koch, Denton, Hanna, and Foster (2021)
Visual by: The Mozilla Internet Health Report 2022
Pattern recognition will reflect existing biases
Executive Order on the Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence

“Artificial Intelligence systems deployed irresponsibly have reproduced and intensified existing inequities, caused new types of harmful discrimination, and exacerbated online and physical harms….It is necessary to hold those developing and deploying AI accountable to standards that protect against unlawful discrimination and abuse, including in the justice system and the Federal Government.”
A “process” view of AI systems (not just a model)

- Data collection
- Develop & Train
- Test predictions
- Monitor & Maintain

“Decision-making” (experimentation)

Focus of most research in ML
Trustworthy data-driven decision-making

- Reliability is a first-order problem in AI-driven decisions
  - Standard CS ML benchmarking view breaks down

- I study AI systems with **distribution shifts** as a central concern
  - Build algorithmic + empirical foundation with a **modern ML lens**

- Main application: online platforms where AI-systems influence high-stakes decisions
  - Algorithmic hiring / sourcing, e.g., allocation of limited recruiter bandwidth across candidates at LinkedIn
A “process” view of AI systems

- Data collection
- Develop & Train
- Test predictions
- Monitor & Maintain
- “Decision-making” (experimentation)
ML as stochastic optimization

- Standard approach: Solve average-case risk minimization

\[
\minimize_{f(\cdot)} \mathbb{E}_P[\ell(Y, f(X))]
\]

- Distributionally robust optimization: Solve worst-case problem

\[
\minimize_{f(\cdot)} \max_{Q \in \mathcal{P}} \mathbb{E}_Q[\ell(Y, f(X))]
\]

- Idea: Do well almost all the time, instead of on average!
Recent progress

- DRO can contribute to generalization, robustness, and fairness
- Intellectual foundations: training algorithms and data efficiency
- Practical impact: algorithms useful when real shifts can be modeled succinctly, e.g., fairness across demographic groups

Vignette: auto-complete service

**Motivation:** Autocomplete system for text

**Problem:** Atypical text doesn’t get surfaced

*American Vernacular English (AAVE)*

If u wit me den u pose to RESPECT ME

*Standard American English (SAE)*

If you are with me then you are supposed to respect me.
DRO mitigates disparity amplification

DRO mitigates disparity amplification

Takeaway:
Control minority proportion
Uniform performance over time

Causal inference and experimentation

Data collection ➔ Develop & Train ➔ Test predictions

Monitor & Maintain ➔ “Decision-making” (experimentation)

Gap between predictions (clicks) and long-term metrics (revenue) bridged via experimentation
Distributional robustness is a useful diagnostic

- Causal inference is fundamental to scientific decision-making
- Its reliability depends on the ability to extrapolate a study’s findings
- Assess validity of findings under distribution shifts
  - Example: finding fails to hold over subpopulations comprising 80% of the study population

Ma, Huang, and N.. A practical minimax approach to causal inference with limited overlap. Work in progress
Industry applications

- **Engineering constraints**: Robust algos under infrastructural constraints
- **Compliance**: Disparate treatment, design best practices for “due diligence”
- **Governance**: Standardize & scale requirements at the company level
Today: Diagnostics

Understand why predictive performance degraded

Data collection → Develop & Train → Test predictions

Monitor & Maintain "Decision-making" (experimentation)
Back to ImageNet

ImageNetV2

Big drop

EfficientNet-B7

VGG, ResNet, DenseNet, ResNeXt, Inception, NASNet, etc.

Standard models

y = x

AlexNet

[Taori, Dave, Shankar, Carlini, Recht, Schmidt ’20]
How do we go up the red line?

- Algorithmic interventions do not provide robustness; only larger training data does—AI community focus on scaling internet data

- But cost of data collection remains a binding constraint; need to understand **which** data to collect

- Implicit assumptions in the CS benchmarking view (one-size-fits-all)
  - Building a universally robust model, just like humans!
  - Focus on covariate shift (X-shift), e.g., image recognition

Even tabular benchmarks mainly focus on $X$-shifts

- Look at loss ratio of deployed model vs. best model for target

\[
\frac{\mathbb{E}_Q[\ell(Y, f_P(X))]}{\min_{f \in \mathcal{F}} \mathbb{E}_Q[\ell(Y, f(X))]} - 1, \quad \text{where} \quad f_P \in \arg\min_{f \in \mathcal{F}} \mathbb{E}_P[\ell(Y, f(X))]
\]

**Existing datasets**

Today: Design new datasets from US census data!
Even tabular benchmarks mainly focus on $X$-shifts

- Look at loss ratio of deployed model vs. best model for target

\[
\frac{\mathbb{E}_Q[\ell(Y, f_P(X))]}{\min_{f \in \mathcal{F}} \mathbb{E}_Q[\ell(Y, f(X))]} - 1, \quad \text{where } f_P \in \arg\min_{f \in \mathcal{F}} \mathbb{E}_P[\ell(Y, f(X))]
\]

Accuracy-on-the-line doesn’t hold under strong $Y|X$-shifts

- Train & target performance correlated only when $X$-shifts dominate
Out of 169 train-target pairs, 80% primarily suffer $Y|X$-shifts.

Existing algos do not show consistent robustness gains:
- They make assumptions about data distributions but do not check them.
- We need an understanding of why the distribution changed!

WhyShift:

<table>
<thead>
<tr>
<th>ACS Income (Young→Old)</th>
<th>ACS Pub.Cov (2010→2017)</th>
<th>ACS Pub.Cov (NE→LA)</th>
<th>US Accident (CA→OR)</th>
<th>ACS Income (CA→PR)</th>
<th>ACS Mobility (MS→HI)</th>
<th>Taxi (NYC→BOG)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td>MLP</td>
<td>Balance</td>
<td>Linear-DRO</td>
<td>SVM</td>
<td>Tree</td>
<td>Fairness</td>
</tr>
<tr>
<td>Target Acc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
DRO revisited

- Distributionally robust optimization: Solve worst-case problem

\[
\minimize_{f(\cdot)} \max_{Q \in \mathcal{P}} \mathbb{E}_Q[\ell(Y, f(X))]
\]

- Choice of ambiguity set $\mathcal{P}$ arbitrary; primarily driven by mathematical convenience and details “left to the modeler”

- Little thought given to model class $f(\cdot)$
Empirical analysis of 10,000 DRO models

- Analyze impact of algorithmic design knobs on model robustness

\[ \text{Acc}_{i,j,s,t} = \alpha + \beta_1^T X_{i,j,s} + \beta_2^T D_{i,j,s,t} + \beta_3 Z_{i,j} + \beta_4^T V_{i,j} + \mu_i + \tau_j + \varepsilon_{i,j,s} \]

Model Class
(Tree, Linear, MLP)

Ambiguity Set
(Distance Type, Radius)

Shift Pattern
(Y|X-ratio)

Validation Type
(Average, Worst)

Task/State fixed effect
Target performance: single state

- Model class most important! Trees >>> ambiguity set
- Effect of ambiguity set inconsistent across different outcomes
Target performance: single state

- Effect of ambiguity set inconsistent across different outcomes

Upper: Predict whether a low-income individual, not eligible for Medicare, has coverage from public health insurance.

Lower: Predict whether annual income > $50K
Target performance: **worst state**

- Even for worst-state performance, DRO is unreliable

Upper: Predict whether a low-income individual, not eligible for Medicare, has coverage from public health insurance.
Lower: Predict whether annual income > $50K
Problems with deductive reasoning

Worst-case distribution does not match real targets

Even train (CA) accuracy is low!

Blue bars: Accuracy of logistic regression models trained on each state.

Red bars: Accuracy on worst-case distribution from a DRO model trained on CA.
Last week’s discussion scientific methods

We lack a modeling language

We lack empirical foundations

Figure from Christopher Ryan, DRO Brown Bag, April 2024
Inductive approach to ambiguity sets: $X$-shifts

- Consider shifts induced by age groups: [20,25), [25,30), …, [75,100)
- Consider DRO methods (DHN’22) tailored to shifts on a subset of covariates
- Variable selection for ambiguity set: top-K with largest subgroup differences
- Performance varies a lot over variables selected

Inductive approach to ambiguity sets: $Y|X$-shifts

- Consider $Y|X$-shifts from NE -> LA (public coverage task)
- Consider DRO methods that consider shifts on a subset of covariates and $Y$
- Variable selection for ambiguity set: $Y|\text{income}$ suffers the largest shift
- Performance varies a lot over variables selected

Consider $Y|X$-shifts from NE -> LA (public coverage task)
Consider DRO methods that consider shifts on a subset of covariates and $Y$
Variable selection for ambiguity set: $Y \mid \text{“income”}$ suffers the largest shift
Performance varies a lot over variables selected

For conditional $Y|X$-DRO: the h-net is a SVM. the alpha-net is a two-layer MLP.
Inductive approach to ambiguity sets

- $Y | X$-shifts from NE -> LA; DRO over shifts on a subset of $(X, Y)$
- Variable selection for ambiguity set: $Y | \text{“income”}$ suffers the largest shift
- Performance varies a lot over variables selected

Takeaways so far

- Underlying model class (neural networks vs. tree ensembles) has first-order impact on robustness, yet frequently overlooked.

- Ambiguity sets should be modeled. Move from deductive to inductive reasoning; do not optimize for math convenience.

- Validation methods for hyperparameter selection matters a lot.

Cai, Liu, Cui, and N., Data Heterogeneity and Distributional Robustness, NeurIPS Tutorial 2023
Rest of the talk: a step toward an inductive modeling language for distribution shifts

- Current ML community: out-of-distribution performance is worse than in-distribution performance,
  - i.e., $P: \text{train} \neq Q: \text{target}$

- How do we attribute performance degradation? Not all shifts matter for model performance

- Different shifts warrant different interventions
  - Our goal today: differentiate $X$- vs. $Y|X$-shifts
Diagnosing Model Performance Under Distribution Shift

- **Density of $X$**: $P_x$ and $Q_x$
- **Expected Loss Given $X$**: $E_{Q}[L|X]$ and $E_{P}[L|X]$

$L$: loss  
$P$: train  
$Q$: target  

$L$ is loss
You can only compare $Y|X$ on shared $X$

$L$: loss
$P$: train
$Q$: target

$E_Q[L|X]$ not well-defined
$E_P[L|X]$ not well-defined

$L$ is loss

Define **Shared Distribution**

\[ \text{density of } X \]

\[ P_x \quad Q_x \]

\[ S_x \]

\[ s_x(x) \propto \frac{p_x(x)q_x(x)}{p_x(x) + q_x(x)} \]

L: loss  
P: train  
Q: target  
S: shared

Decompose change in performance

$E_p[E_p[L|X]]$  
Performance on the training distribution

$E_Q[E_Q[L|X]]$  
Performance on the target distribution

Decompose into $X$-shift vs. $Y|X$-shift

L: loss  
P: train  
Q: target  
S: shared

Decompose change in performance

\[ E_p[E_p[L|X]] \xrightarrow{X \text{ shift } (P \to S)} E_S[E_p[L|X]] \]

Diagnosis:

**S** has more X’s that are harder to predict than **P**

Potential interventions:

Use domain adaptation, e.g. importance weighting

Decompose change in performance

Diagnosis:
\( Y|X \) moves farther from predicted model

Potential interventions:
- Re-collect data
- or modify covariates
Decompose change in performance

Diagnosis:
Q has “new” X’s that are harder to predict than S

Potential interventions:
Collect + label more data on “new” examples

\[ \mathbb{E}_S[\mathbb{E}_Q[\mathbb{L}|X]] \xrightarrow{X \text{ shift } (S \rightarrow Q)} \mathbb{E}_Q[\mathbb{E}_Q[\mathbb{L}|X]] \]

Decompose change in performance

\[ E_p[E_p[L|X]] \xrightarrow{X \text{ shift (} P \rightarrow S \text{)}} E_s[E_p[L|X]] \]
\[ \xdownarrow{Y \mid X \text{ shift}} \]
\[ E_s[E_Q[L|X]] \xrightarrow{X \text{ shift (} S \rightarrow Q \text{)}} E_Q[E_Q[L|X]] \]

Legend:
- L: loss
- P: train
- Q: target
- S: shared

Estimation

\[ \mathbb{E}_p[\mathbb{E}_p[L|X]] \xrightarrow{X \text{ shift } (P \rightarrow S)} \mathbb{E}_s[\mathbb{E}_p[L|X]] \]

Legend:
- **L**: loss
- **P**: train
- **Q**: target
- **S**: shared

\[ \mathbb{E}_s[\mathbb{E}_q[L|X]] \xrightarrow{X \text{ shift } (S \rightarrow Q)} \mathbb{E}_q[\mathbb{E}_q[L|X]] \]
Estimation

\[ E_p[E_p[L|X]] \xrightarrow{X \text{ shift } (P \to S)} E_S[E_p[L|X]] \]

\[ E_S[E_Q[L|X]] \xrightarrow{X \text{ shift } (S \to Q)} E_Q[E_Q[L|X]] \]

Legend:
- **L**: loss
- **P**: train
- **Q**: target
- **S**: shared
How do you take expectations over $S$???

Legend:

- **L**: loss
- **P**: train
- **Q**: target
- **S**: shared

$E_p[E_p[L|X]]$ → $X$ shift $(P \rightarrow S)$ → $E_S[E_p[L|X]]$ → $Y | X$ shift → $E_S[E_Q[L|X]]$ → Importance weighting!
Importance weights look like classifier probabilities

Reweight samples from $P$ and $Q$ into $S$ using importance weighting. The importance weights are

$$
\frac{dS_X}{dP_X}(x) \propto \frac{q(x)}{p(x) + q(x)} \quad \text{and} \quad \frac{dS_X}{dQ_X}(x) \propto \frac{p(x)}{p(x) + q(x)}
$$

Importance weights look like classifier probabilities of $X$ being from $P$ vs $Q$
Method

1. Train domain classifier to classify X as coming from $P$ vs $Q$
2. Reweight losses from $P$ and $Q$ into $S$ using class probabilities

**Shared $S$ inputs are those that can’t be confidently classified as $P$ vs $Q$**
Confidence intervals

[Theorem: asymptotics] For a nonparametric classifier / reweighting that is asymptotically accurate, our estimator for $\theta_Q = \mathbb{E}_S[\mathbb{E}_Q[L|X]]$ is asymptotically normal

$$\sqrt{n}(\hat{\theta}_Q - \theta_Q) \overset{d}{\sim} N(0, \operatorname{Var}(\psi_Q(W)))$$

and we can estimate $\operatorname{Var}(\psi_Q(W))$ using plug-ins to calculate confidence intervals.

[Theorem: semiparametric efficiency] Our estimator gives the tightest possible confidence interval, achieving the lowest possible (asymptotic) variance.
Employment prediction case study

[X shift]  **P**: only age ≤25,  **Q**: general population

Performance attributed to X shift (S → Q), meaning “new examples” such as older people.
Employment prediction case study

[X shift] \( P \): age \( \leq 25 \) overrepresented, \( Q \): evenly sampled population

Substantial portion attributed to X shift (\( P \rightarrow S \)), suggesting domain adaptation may be effective
Better data can be effective

\[ \text{[Y|X shift]} \quad \text{P: California (CA), } \quad \text{Q: Puerto Rico (PR)} \]

No language features

<table>
<thead>
<tr>
<th>Accuracy</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA: 81.7</td>
<td>PR: 71.4</td>
</tr>
</tbody>
</table>

With language features

<table>
<thead>
<tr>
<th>Accuracy</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA: 81.8</td>
<td>PR: 79.7</td>
</tr>
</tbody>
</table>

CA model does not use language.

\[ \text{Y|X shift because of missing covariate: language affects outcome} \]

\[ \rightarrow \text{better performance in PR} \]
A methodological bottleneck: uncertainty

Only observe outcomes on items we recommend. How do we collect outcomes across a huge space?

Data collection → Develop & Train → Test predictions

Monitor & Maintain → “Decision-making” (experimentation)

Distribution Shift Decomposition (DISDE)

- Diagnostic for understanding why performance dropped in terms of $X$ vs $Y|X$ shift
- Can help articulate modeling assumptions + data collection

We need a modeling language for a data-centric view of AI

- Develop modeling tools in an application-specific manner!
- Top of mind: resolving methodological bottlenecks in uncertainty quantification

What’s next?

- Industrial applications
  - Governance: Scale minimal requirements at the company level
  - Compliance: Design best practices for “due diligence” in responsible AI
  - Engineering constraints: Design algorithms under infrastructural constraints
- Methodological bottlenecks: uncertainty quantification, objective and actions defined on different timescales
- Top of mind: Measurement and mitigation in shifting AI paradigms
Employment prediction case study

[Y|X shift]  P: West Virginia,  Q: Maryland

WV model does not use education.

Y|X shift because of missing covariate: education affects employment.
Better data can be more effective than better algorithms!

\[ Y|X \text{ shift} \quad \mathbf{P}: \text{California (CA)}, \quad \mathbf{Q}: \text{Puerto Rico (PR)} \]

Include language features when training on CA → better performance in PR

- **No language features**
  - CA: 81.7
  - PR: 71.4
  - Difference: 10.3

- **With language features**
  - CA: 81.8
  - PR: 79.7
  - Difference: 2.1

**collecting better features**

**collecting better target data**
Appendix: Variables in Linear Analysis

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Class</td>
<td>Tree</td>
<td>A dummy variable that takes value one if the base learner of the model configuration is tree-structure</td>
</tr>
<tr>
<td>$X_{i,j,s}$</td>
<td>MLP</td>
<td>A dummy variable that takes value one if the base learner of the model configuration is MLP</td>
</tr>
<tr>
<td>Ambiguity Set</td>
<td>Wasserstein</td>
<td>A dummy variable that takes value one if the model configuration belongs to DRO and uses Wasserstein-type metric</td>
</tr>
<tr>
<td>$D_{i,j,s,t}$</td>
<td>Chi-squared</td>
<td>A dummy variable that takes value one if the model configuration belongs to DRO and uses $\chi^2$-divergence metric</td>
</tr>
<tr>
<td></td>
<td>Kullback-Leibler</td>
<td>A dummy variable that takes value one if the model configuration belongs to DRO and uses KL-divergence metric</td>
</tr>
<tr>
<td></td>
<td>Total Variation</td>
<td>A dummy variable that takes value one if the model configuration belongs to DRO and uses TV-distance metric</td>
</tr>
<tr>
<td></td>
<td>OT-Discrepancy</td>
<td>A dummy variable that takes value one if the model configuration belongs to DRO and uses the optimal transport-discrepancy with conditional moment constraints</td>
</tr>
<tr>
<td></td>
<td>Radius</td>
<td>The rescaled ambiguity size if the model configuration belongs to DRO and equal to zero if the model configuration does not belong to DRO</td>
</tr>
<tr>
<td>Shift Pattern $Z_{i,j}$</td>
<td>Y</td>
<td>X-ratio</td>
</tr>
<tr>
<td>Validation Type $V_{i,j}$</td>
<td>Worst</td>
<td>A dummy variable that takes value one if the best configuration is obtained through the largest accuracy from the worst target domain</td>
</tr>
<tr>
<td></td>
<td>Average</td>
<td>A dummy variable that takes value one if the best configuration is obtained through the largest accuracy from the average-case target domain</td>
</tr>
</tbody>
</table>
Appendix: Configurations

- **Algorithms evaluated in our empirical study:**

  1. **Basic learners:** Logistic Regression (LR), SVM, fully-connected neural networks (MLP) with standard ERM optimization;
  2. **Tree-based learners:** Random Forest (RF) [8], GBM [26], LightGBM [19], XGBoost [9];
  3. **Imbalanced learning algorithms:** SUBY, RWY, SUBG, RWG [17], which reweight or subsample data to balance the samples of different classes (Y) or different demographic groups (G);
  4. **Fairness-enhancing algorithms:** In-processing methods [4] with demographic parity, equal opportunity, and error parity as constraints, and post-processing methods [15] with exponential and threshold controls;
  5. **Linear-DRO algorithms:** Distributionally robust optimization (DRO) methods based on linear SVM using different uncertainty sets, including CVaR-DRO [28], $\chi^2$-DRO [12], TV-DRO [18], KL-DRO [16], Wasserstein-DRO [6], Augmented Wasserstein-DRO [30], Satisficing Wasserstein-DRO [22], Sinkhorn-DRO [31], Holistic-DRO [5], Unified-DRO (with $L_2$-norm) [7], and Unified-DRO (with $L_{inf}$-norm) [7];
  6. **NN-DRO algorithms:** DRO methods based on MLP using different uncertainty sets, including CVaR-DRO (NN) and $\chi^2$-DRO (NN) with fast implementation [21], CVaR-DORO (NN) and $\chi^2$-DORO (NN) that are designed for outlier robustness [34].
Worst-case Distribution Analysis

- Misalignment between worst-case distributions and target distributions
  - when we use the worst-case distribution of KL-DRO to train tree-based methods, their target accuracies **even drop a lot**

(a) ACS Income, LightGBM  
(b) ACS Income, XGB
Worst-case Distribution Analysis

- Recall that KL-DRO improves the worst target performance on ACS Pub.Cov

<table>
<thead>
<tr>
<th>Model Class</th>
<th>Variable Name</th>
<th>Dependent variable: Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Setup 1: one-to-one</td>
</tr>
<tr>
<td></td>
<td></td>
<td>All</td>
</tr>
<tr>
<td>Tree</td>
<td>.0095***</td>
<td>.0047***</td>
</tr>
<tr>
<td></td>
<td>(.0027)</td>
<td>(.0012)</td>
</tr>
<tr>
<td>MLP</td>
<td>.0036*</td>
<td>-.0142***</td>
</tr>
<tr>
<td></td>
<td>(.0019)</td>
<td>(.0009)</td>
</tr>
<tr>
<td>Wasserstein</td>
<td>-.0048*</td>
<td>-.0077***</td>
</tr>
<tr>
<td></td>
<td>(.0027)</td>
<td>(.0012)</td>
</tr>
<tr>
<td>Chi-squared</td>
<td>.0011</td>
<td>.0022*</td>
</tr>
<tr>
<td></td>
<td>(.0025)</td>
<td>(.0011)</td>
</tr>
<tr>
<td>Kullback-Leibler</td>
<td>.0024</td>
<td>.0008</td>
</tr>
<tr>
<td></td>
<td>(.0025)</td>
<td>(.0012)</td>
</tr>
</tbody>
</table>
Worst-case Distribution Analysis

- But still conservative!
  - We train LightGBM and XGBoost models on the worst-case distribution of KL-DRO
  - The worst-case performance over 50 target states improves
  - But the overall target performances drop a lot!
Algorithmic Intervention: design better ambiguity sets?

Case study on covariate shifts:

- for Marginal-DRO and Wasserstein DRO
- only perturb the covariates whose distributions shift a lot among age groups
  - pick the Top-shifted covariates
- measure the worst sub-group accuracy (age groups: [20,25), [25,30), …, [75,100) )

Task: income prediction
Source: Age < 25
Target: Age ≥ 25

Only perturb top-\(i\) features at time \(i\)
Non-Algorithmic Intervention: collect better features/data?

- Region Analysis on $Y|X$-shift

1. Construct shared distribution from training and target
2. Model $Y$ separately on each of training and target: $f_p, f_q$
3. Model difference in $Y$ between train and target $|f_p(x) - f_q(x)|$ on shared distribution using interpretable tree-based model

Find Covariate Regions with Strong $Y|X$-Shifts!

Non-Algorithmic Intervention: collect better features/data?

**Task:** Income Prediction

**Shift:** CA -> PR

---

**Tabular Data**

<table>
<thead>
<tr>
<th>Y</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shift region consists of occupations that require language</td>
<td></td>
</tr>
</tbody>
</table>

Official languages are *different* in CA and PR!

---

(c) Region with $Y|X$-shifts (XGBoost)
WhyShift

- Initial conjecture: Y|X-shifts are more prominent than X-shifts in practice
- Out of 169 source-target pairs with significant performance degradation, 80% of them are primarily attributed to Y|X-shifts.
AI pipeline

Data collection

AI development cycle

Model training

Validation & Monitoring
Decompose change in performance

Legend:
- $L$: loss
- $P$: train
- $Q$: target
- $S$: shared

$E_p[E_p[L|X]] \xrightarrow{X \text{ shift } (P \rightarrow S)} E_s[E_p[L|X]] \xrightarrow{Y \mid X \text{ shift}} E_Q[E_p[L|X]]$

$E_p[E_Q[L|X]] \xrightarrow{} E_s[E_Q[L|X]] \xrightarrow{X \text{ shift } (S \rightarrow Q)} E_Q[E_Q[L|X]]$
Estimation

Legend:
- L: loss
- P: train
- Q: target
- S: shared

\[ \mathbb{E}_p[\mathbb{E}_p[L|X]] \xrightarrow{X \text{ shift } (P \rightarrow S)} \mathbb{E}_s[\mathbb{E}_p[L|X]] \xrightarrow{Y \mid X \text{ shift}} \mathbb{E}_q[\mathbb{E}_p[L|X]] \]

\[ \mathbb{E}_p[\mathbb{E}_q[L|X]] \xrightarrow{X \text{ shift } (S \rightarrow Q)} \mathbb{E}_s[\mathbb{E}_q[L|X]] \xrightarrow{Y \mid X \text{ shift}} \mathbb{E}_q[\mathbb{E}_q[L|X]] \]
Estimation

Legend:
- L: loss
- P: train
- Q: target
- S: shared
How do you take expectations over $S$???

Legend:
- L: loss
- P: train
- Q: target
- S: shared

Importance weighting!
More description of datasets and shifts. Outcomes etc.

In spirit, describe the dro that actually works

Expand on why can’t we just do regular ml benchmarking on distribution shifts