

Lecture 3: Distribution Shift

CIS 7000: Trustworthy Machine Learning

Spring 2024

Agenda

- **Robustness to distribution shift**
 - Basic examples
 - Definitions
 - Unsupervised domain adaptation setting
- **Algorithms for distributional robustness**
 - Importance weighting
 - Application to label shift
 - Application to covariate shift

Robustness to Distribution Shift

- Neural networks generalize well **on distribution**
- **Ideal scenario**
 - Test set and training set are i.i.d. from the same distribution
 - **Equivalently:** Test set is obtained by shuffling entire dataset and then splitting
- **Often fails in practice! “Distribution shift”**

Robustness to Distribution Shift

- **Images/computer vision**

- Added noise, color shifts, lighting changes, different resolution, etc.

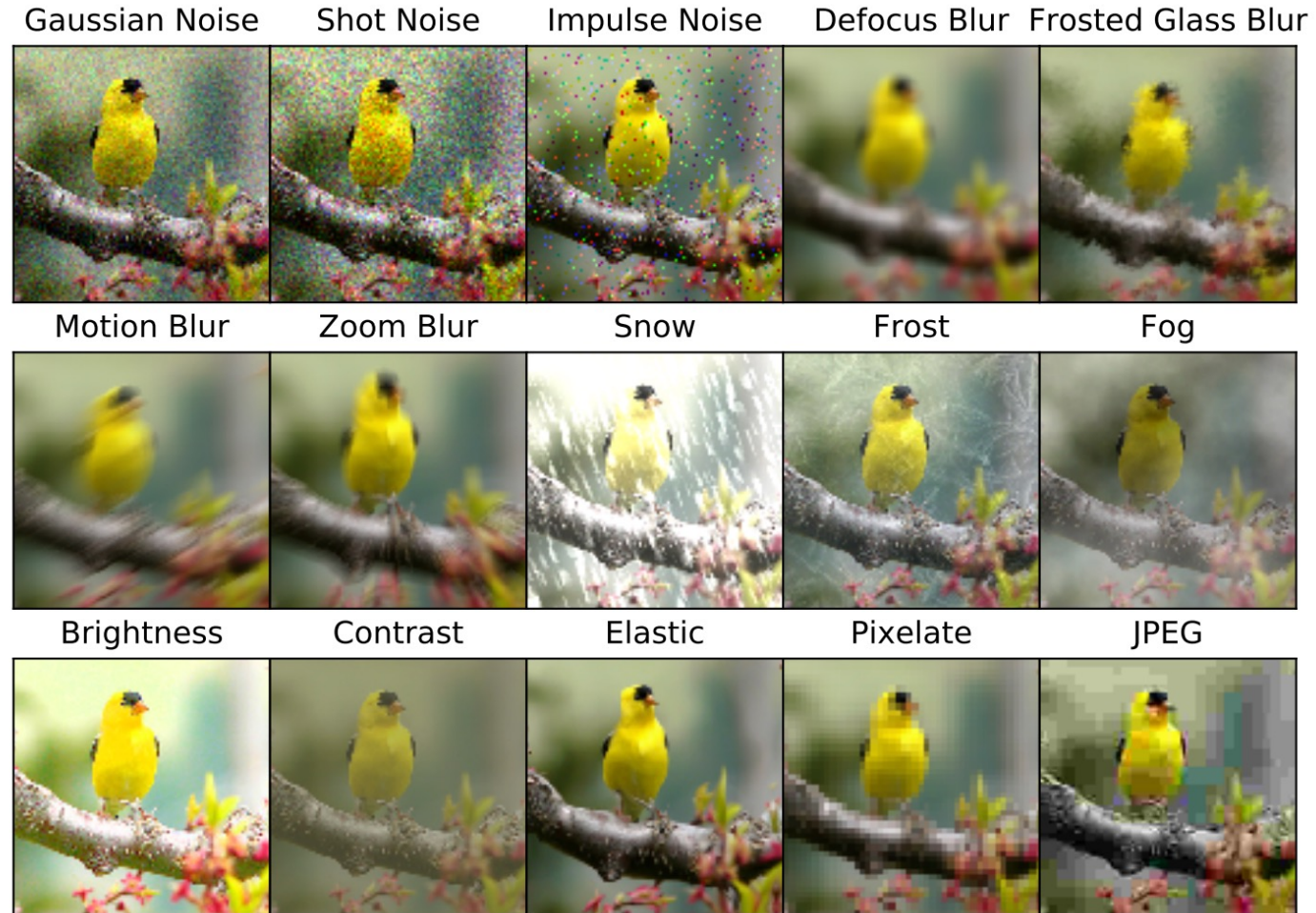
- **Audio/speech-to-text**

- Noisy background, changes in recording device, etc.

- **Natural language processing**

- Substitute synonyms, add unrelated text, etc.

Example: Synthetic Perturbations



Example: Synthetic Perturbations

- **Question:** Why should the model be robust?
- **Answer:** Humans are robust!

Example: Synthetic Perturbations

- **Significantly reduces performance**
 - 20% error rate → 80% error rate
- **Data augmentation can help (but not 100% solution)**

Data Augmentation

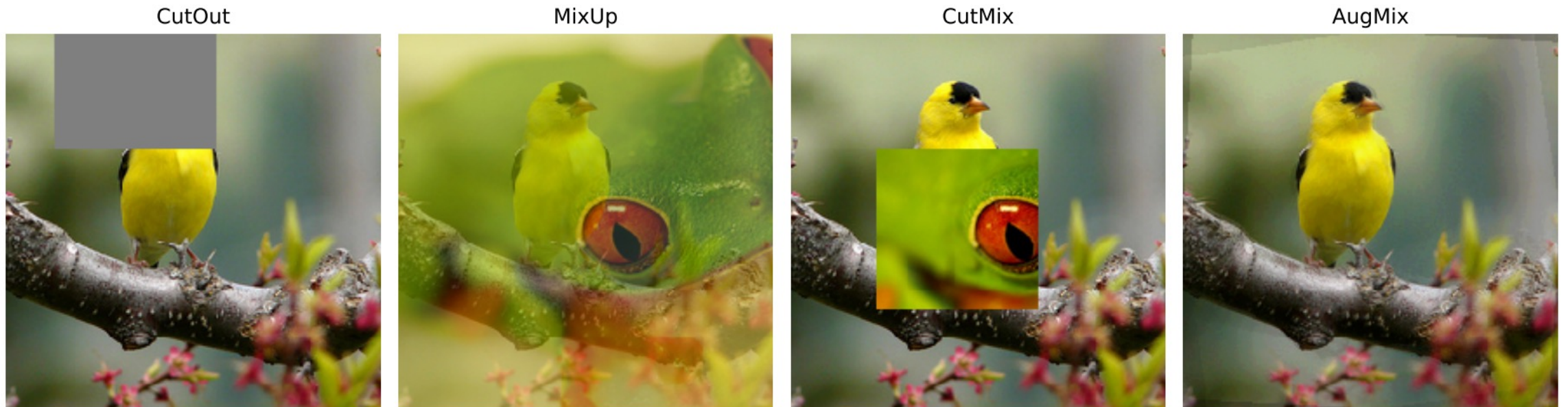
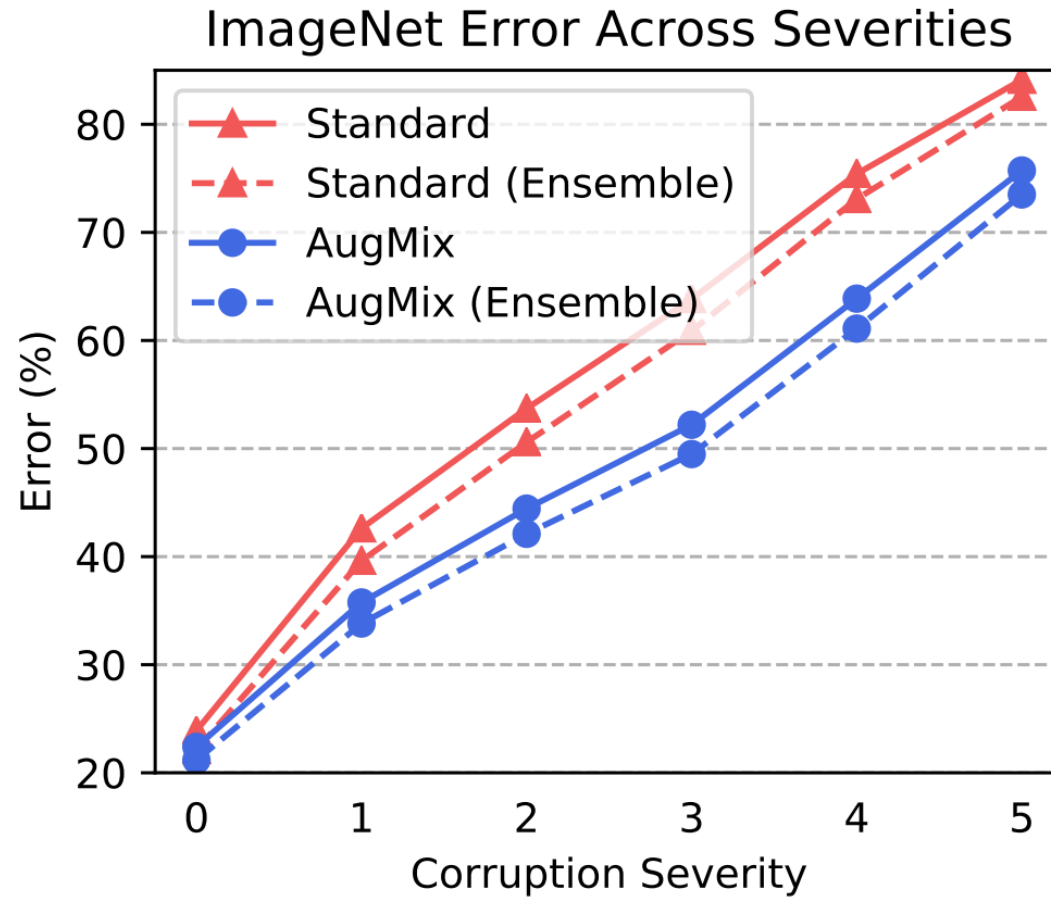


Figure 1: A visual comparison of data augmentation techniques. AUGMIX produces images with variety while preserving much of the image semantics and local statistics.

Data Augmentation



Example: Natural Language Processing

Article: Super Bowl 50








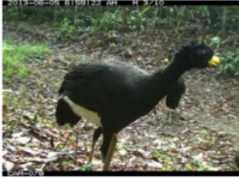


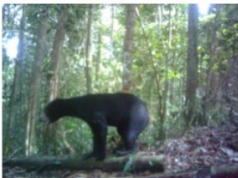
Paragraph: *“Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.”*

Question: *“What is the name of the quarterback who was 38 in Super Bowl XXXIII?”*






Original Prediction: John Elway

Prediction under adversary: Jeff Dean

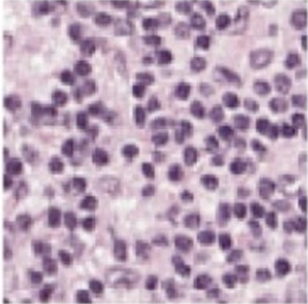
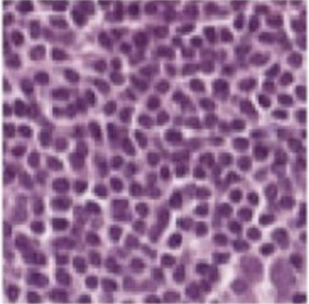
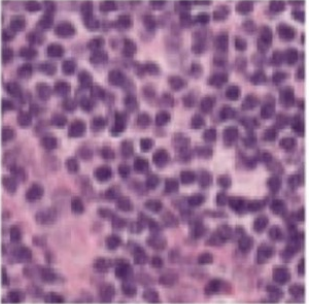
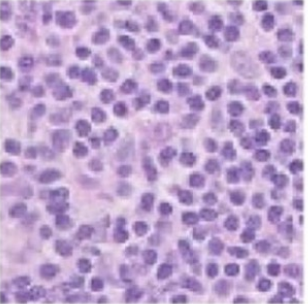
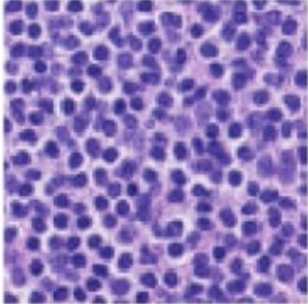
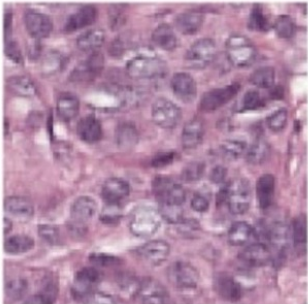
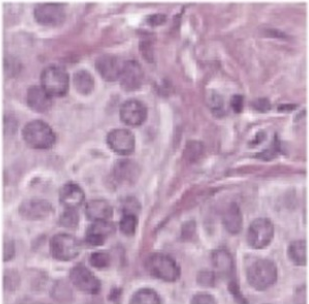
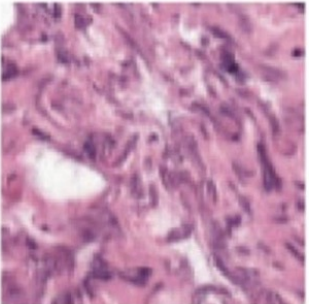
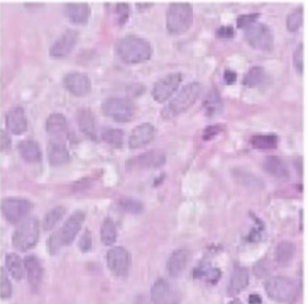
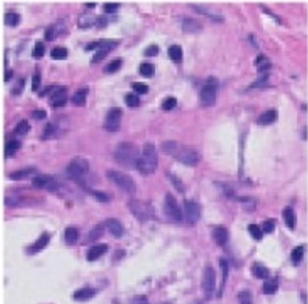
Example: Real Perturbations

Train			Test (OOD)
$d = \text{Location 1}$	$d = \text{Location 2}$	$d = \text{Location 245}$	$d = \text{Location 246}$
			
Vulturine Guineafowl	African Bush Elephant	...	Wild Horse
			
Cow	Cow	Southern Pig-Tailed Macaque	Great Curassow
Test (ID)			
$d = \text{Location 1}$	$d = \text{Location 2}$	$d = \text{Location 245}$	
			
Giraffe	Impala	Sun Bear	

Example: Real Perturbations

	Train			Test	
Satellite Image (x)					
Year / Region (d)	2002 / Americas	2009 / Africa	2012 / Europe	2016 / Americas	2017 / Africa
Building / Land Type (y)	shopping mall	multi-unit residential	road bridge	recreational facility	educational institution

Example: Real Perturbations

	Train			Val (OOD)	Test (OOD)
	d = Hospital 1	d = Hospital 2	d = Hospital 3	d = Hospital 4	d = Hospital 5
y = Normal					
y = Tumor					

Agenda

- **Robustness to distribution shift**
 - Basic examples
 - Definitions
 - Unsupervised domain adaptation setting
- **Algorithms for distributional robustness**
 - Importance weighting
 - Application to label shift
 - Application to covariate shift

Traditional Supervised Learning

- **Problem setup**

- Consider a parametric model family $\{f_\theta: \mathcal{X} \rightarrow \mathcal{Y} \mid \theta \in \Theta\}$
- Consider a loss function $\ell(\theta; x, y)$
- Consider a data distribution P over $\mathcal{X} \times \mathcal{Y}$

- **Supervised learning problem**

- Given training dataset $Z \subseteq \mathcal{X} \times \mathcal{Y}$ consisting of i.i.d. samples $(x, y) \sim P$
- Expected/empirical loss $\mathbb{E}_P[\ell(\theta; x, y)] \approx |Z|^{-1} \sum_{(x,y) \in Z} \ell(\theta; x, y)$
- Goal is to compute $\hat{\theta} = \min_{\theta} |Z|^{-1} \sum_{(x,y) \in Z} \ell(\theta; x, y)$

Distribution Shift

- **Distribution shift:** Training and test distributions differ
 - Training set consists of samples $(x_1, y_1), \dots, (x_n, y_n) \sim P$
 - Test set consists of samples $(x'_1, y'_1), \dots, (x'_m, y'_m) \sim Q$
- **Supervised learning under distribution shift**
 - Given training dataset $Z \subseteq \mathcal{X} \times \mathcal{Y}$ consisting of i.i.d. samples $(x, y) \sim P$
 - Goal is to minimize loss $\mathbb{E}_Q[\ell(\theta; x, y)] \neq |Z|^{-1} \sum_{(x,y) \in Z} \ell(\theta; x, y)$
 - Computing $\hat{\theta} = \min_{\theta} |Z|^{-1} \sum_{(x,y) \in Z} \ell(\theta; x, y)$ may not work

Aside: Adversarial Robustness

- In adversarial robustness, the goal is to be robust to all perturbations of the form $x' = x + \epsilon$, where ϵ is small but **arbitrary**
- **Question:** Can we treat $x + \epsilon$ as a distribution shift?
- **Answer:** Yes! But with a major caveat...
 - The shifted distribution Q can depend on the model f
 - If f is fixed, this works fine!

Distribution Shift

- Intuitively, when can we hope to perform well on Q ?
- **Impossible in general (what if we swap the labels?)**
- Can we leverage additional information about the shift?
 - Make additional assumptions about shift
 - Leverage additional data
 - Both

Distribution Shift

- Intuitively, when can we hope to perform well on Q ?
- **Impossible in general (what if we swap the labels?)**
- Can we leverage additional information about the shift?
 - **Make additional assumptions about shift**
 - Leverage additional data
 - Both

Distributionally Robust Optimization

- **Idea:** Robust to an arbitrary small shift
- **Example:** Small in KL divergence:

$$\{ Q \mid D_{\text{KL}}(P \parallel Q) \leq \epsilon \}$$

- Very similar to adversarial robustness (covered later)
- We can do much better with a little extra information

Distribution Shift

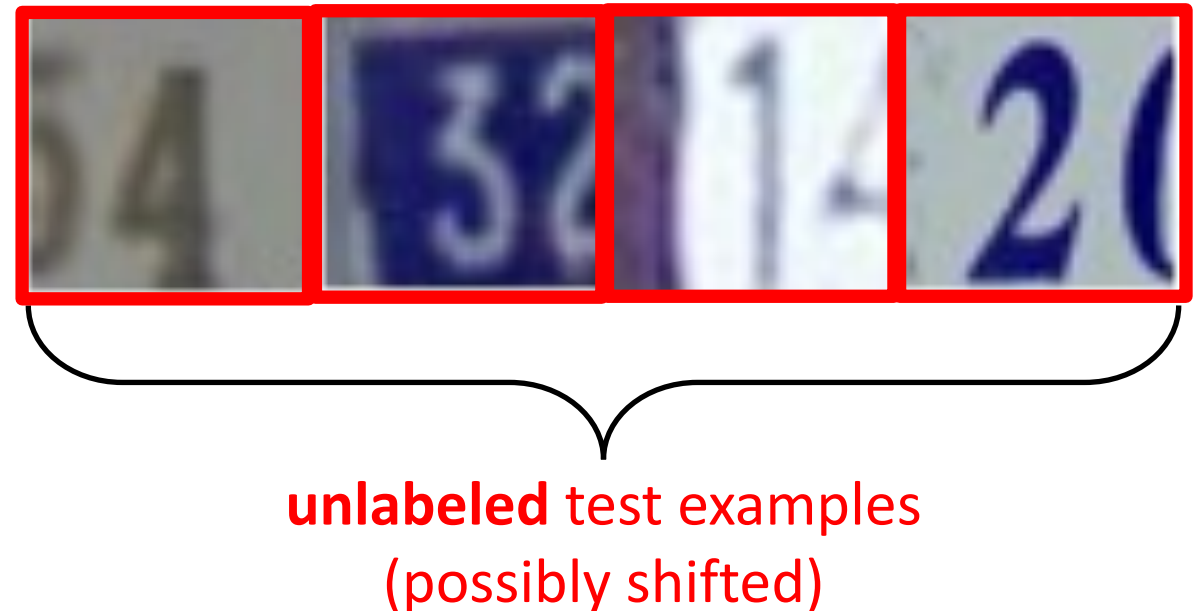
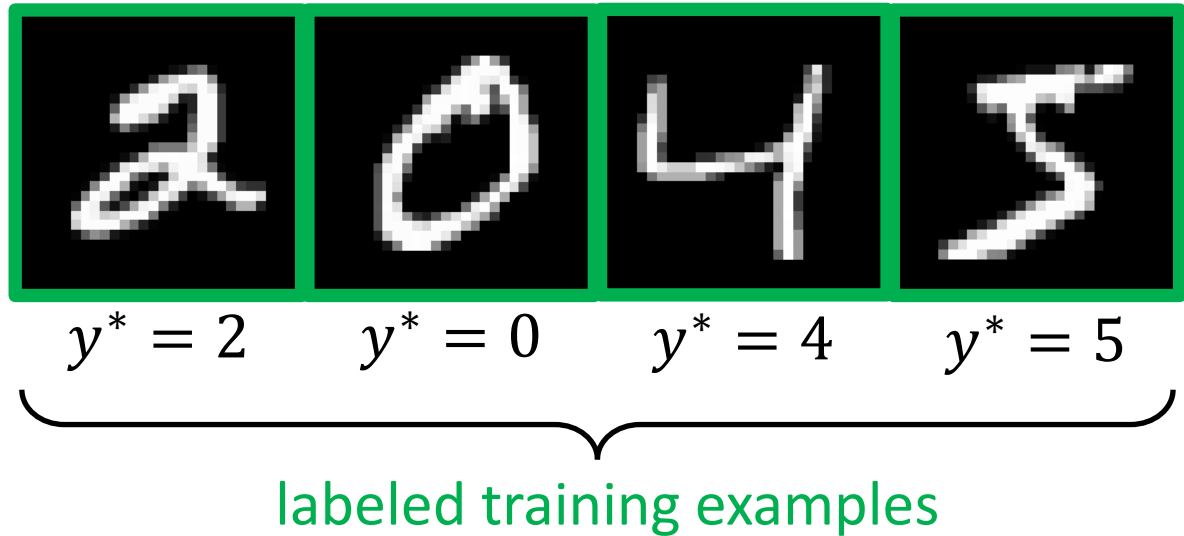
- Intuitively, when can we hope to perform well on Q ?
- **Impossible in general (what if we swap the labels?)**
- Can we leverage additional information about the shift?
 - Make additional assumptions about shift
 - Leverage additional data
 - Both

Distribution Shift

- Intuitively, when can we hope to perform well on Q ?
- **Impossible in general (what if we swap the labels?)**
- Can we leverage additional information about the shift?
 - Make additional assumptions about shift
 - Leverage additional data
 - **Both**

Unsupervised Domain Adaptation

- **Idea:** Use **some** information about the distribution shift
- Consider **unsupervised domain adaptation** setting



Unsupervised Domain Adaptation

- Data is easy to collect but labeling costs money
 - **Example:** Data from a different hospital
- Collect data during run time
 - **Example:** Self-driving car

Covariate Shift Assumption

- Let p and q be the density functions for P and Q , respectively
- **Covariate Shift Assumption:** $p(y | x) = q(y | x)$
 - But may have $p(x) \neq q(x)$
 - **Intuition:** The label computation does not change, but the inputs can change
- **Examples**
 - $y = \beta^\top x + \epsilon$, but $P(x) = N(\mu, \sigma^2)$ while $Q(x) = N(\mu', \sigma'^2)$
 - Daytime vs. nighttime, driving in new city, changes in color/lighting

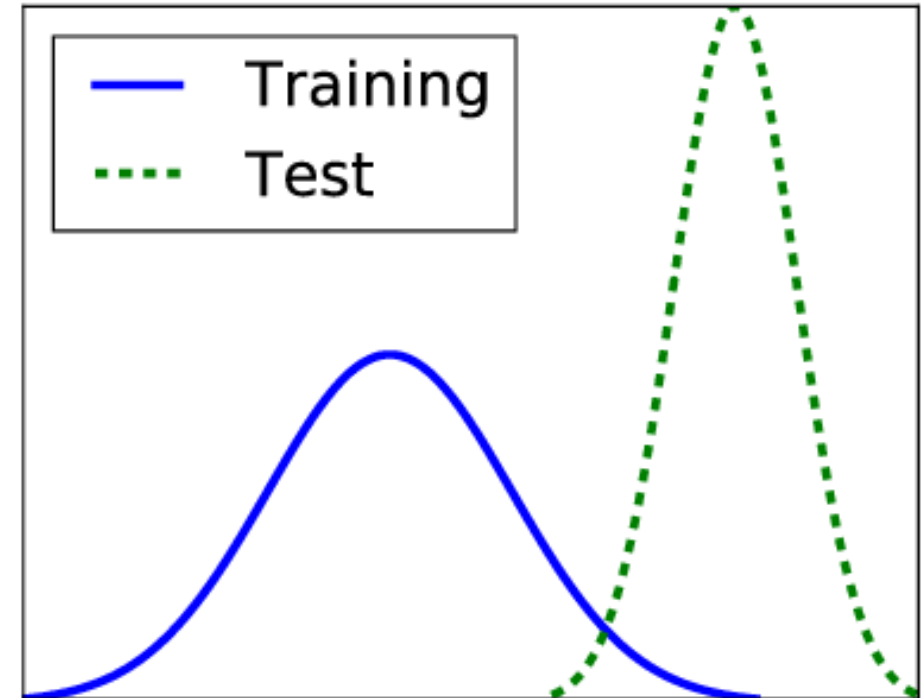
Covariate Shift Assumption

- **Covariate distributions**

- $P(x) = N(\mu, \sigma^2)$
- $Q(x) = N(\mu', \sigma'^2)$

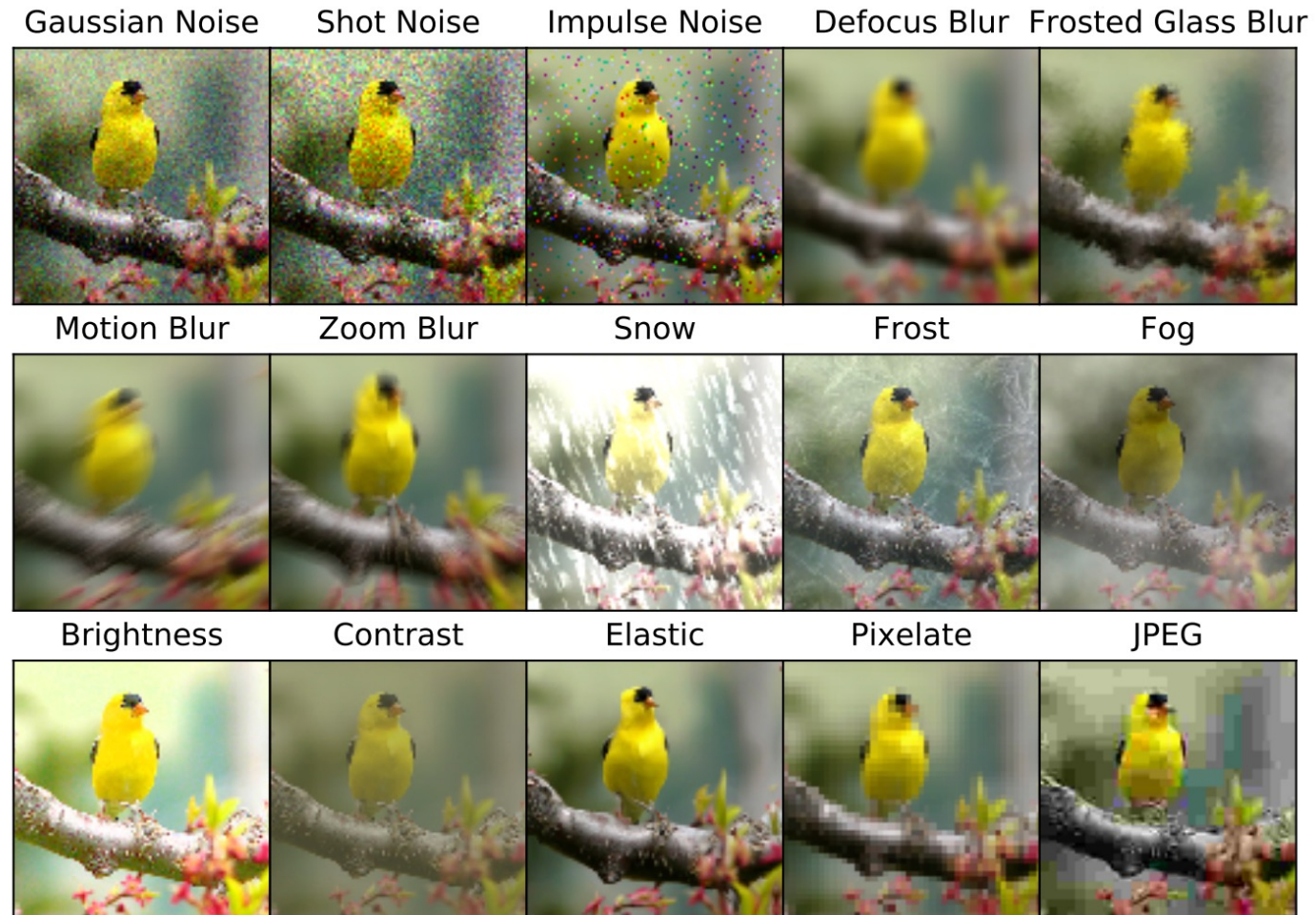
- **Label distribution**

- $P(y | x) = Q(y | x) = N(\beta^\top x, \sigma''^2)$
- i.e., $y = \beta^\top x + \epsilon$, where $\epsilon \sim N(0, \sigma''^2)$











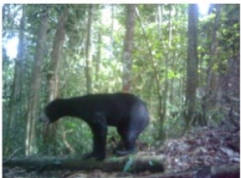


Image; Glauner et al., 2018

Covariate Shift Assumption



Covariate Shift Assumption

Train			Test (OOD)
$d = \text{Location 1}$	$d = \text{Location 2}$	$d = \text{Location 245}$	$d = \text{Location 246}$
			
Vulturine Guineafowl	African Bush Elephant	...	Wild Horse
			
Cow	Cow	Southern Pig-Tailed Macaque	Great Curassow
Test (ID)			
$d = \text{Location 1}$	$d = \text{Location 2}$	$d = \text{Location 245}$	
			
Giraffe	Impala	Sun Bear	

Covariate Shift Assumption

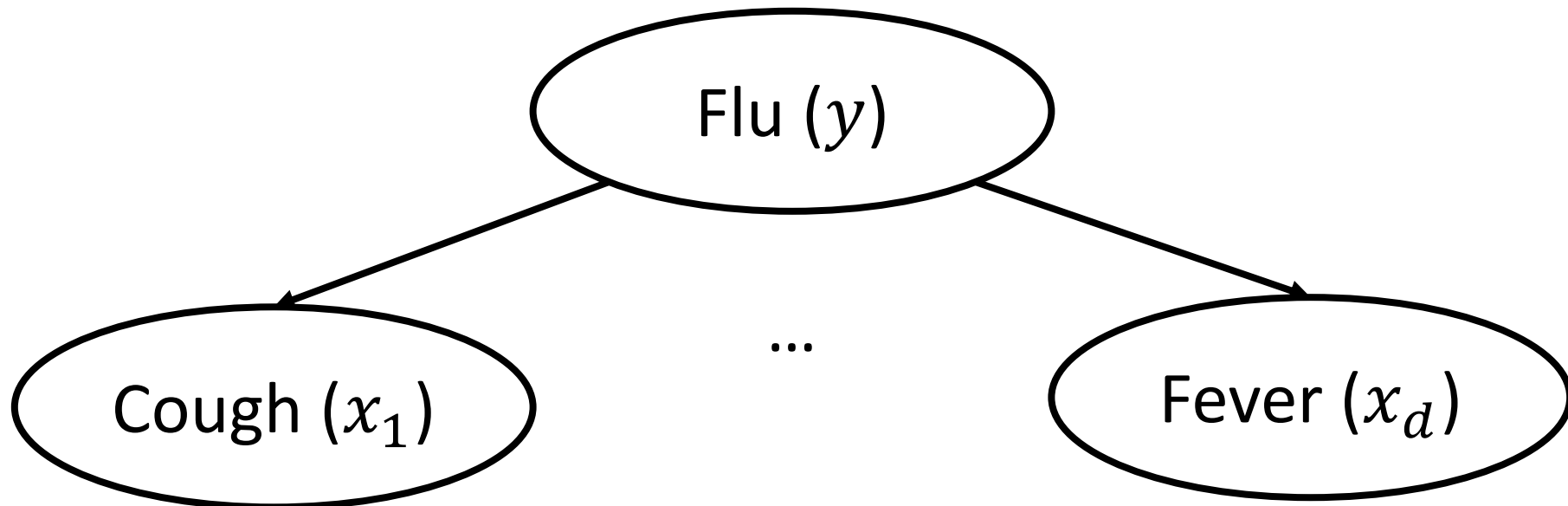
- **Computer vision**
 - Daytime vs. nighttime
 - Color shifts, lighting shifts, etc.
 - Driving in a new city
- **Natural language processing**
 - Change in vocabulary frequency over time
 - Regional vocabulary
 - News writing vs. conversational writing
- Covariate shift is pervasive

Label Shift Assumption

- Let p and q be the density functions for P and Q , respectively
- **Label Shift Assumption:** $p(x | y) = q(x | y)$
 - But may have $p(y) \neq q(y)$
 - **Intuition:** The rates of labels changes, but the kinds of

Label Shift Assumption

- **Example:** Increase in flu cases due to an outbreak
 - x are the symptoms, y is indicator for flu
 - $P(x | y)$ is rate of symptoms conditioned on having disease (stays the same)
 - $P(y)$ is rate of flu (can change if there is an outbreak)



Label Shift Assumption

- **Example:** Changes in label distribution
 - x is an image, y is the label
 - $P(x | y)$ is the distribution of images of a given label
 - $P(y)$ is rate of that label
- Often, the training labels are balanced, which is a source of label shift

airplane



automobile



bird



cat



deer



dog



frog



horse



ship



truck



Agenda

- **Robustness to distribution shift**
 - Basic examples
 - Definitions
 - Unsupervised domain adaptation setting
- **Algorithms for distributional robustness**
 - Importance weighting
 - Application to label shift
 - Application to covariate shift

Importance Weighting

- Given distributions P and Q , the **importance weight (function)** is

$$w(x, y) = \frac{q(x, y)}{p(x, y)}$$

- **Key property (by definition):**

$$\mathbb{E}_Q[\ell(\theta; x, y)] = \mathbb{E}_P[\ell(\theta; x, y) \cdot w(x, y)]$$

Importance Weighting

- Note that

$$\mathbb{E}_Q[\ell(\theta; x, y)]$$

Importance Weighting

- Note that

$$\begin{aligned}\mathbb{E}_Q[\ell(\theta; x, y)] &= \int_{x \times y} \ell(\theta; x, y) \cdot q(x, y) \cdot dx \cdot dy \\ &= \int_{x \times y} \ell(\theta; x, y) \cdot \frac{q(x, y)}{p(x, y)} \cdot p(x, y) \cdot dx \cdot dy \\ &= \int_{x \times y} \ell(\theta; x, y) \cdot w(x, y) \cdot p(x, y) \cdot dx \cdot dy \\ &= \mathbb{E}_P[\ell(\theta; x, y) \cdot w(x, y)]\end{aligned}$$

- We have assumed the support of Q is contained in the support of P !

Importance Weighting

- Given distributions P and Q , the **importance weight (function)** is

$$w(x, y) = \frac{q(x, y)}{p(x, y)}$$

- **Key property (by definition):**

$$\mathbb{E}_Q[\ell(\theta; x, y)] = \mathbb{E}_P[\ell(\theta; x, y) \cdot w(x, y)]$$

- **Key question:** How to compute importance weights?

Importance Weights for Label Shift

- In the label shift setting, we have

$$w(x, y)$$

Importance Weights for Label Shift

- In the label shift setting, we have

$$\begin{aligned}w(x, y) &= \frac{q(x, y)}{p(x, y)} \\ &= \frac{q(x|y)q(y)}{p(x|y)p(y)} \\ &= \frac{q(y)}{p(y)} \\ &:= w(y)\end{aligned}$$

Importance Weights for Label Shift

- If we know $w(y)$, then we have

$$\mathbb{E}_Q[\ell(\theta; x, y)]$$

Importance Weights for Label Shift

- If we know $w(y)$, then we have

$$\mathbb{E}_Q[\ell(\theta; x, y)] = \mathbb{E}_P[\ell(\theta; x, y) \cdot w(x, y)]$$

Importance Weights for Label Shift

- If we know $w(y)$, then we have

$$\mathbb{E}_Q[\ell(\theta; x, y)] = \mathbb{E}_P[\ell(\theta; x, y) \cdot w(x, y)] = \mathbb{E}_P[\ell(\theta; x, y) \cdot w(y)]$$

Label Shift Assumption

- **Training:** $p(y) = \frac{1}{10}$
- **Test:** $q(\text{automobile}) = \frac{1}{2}$
and $q(y) = \frac{1}{18}$ otherwise
- Then, the loss might be

$$\begin{aligned} & \frac{10}{18} \cdot \ell(x_1, \text{dog}) \\ & + \frac{10}{2} \cdot \ell(x_2, \text{automobile}) \\ & + \frac{10}{18} \cdot \ell(x_3, \text{frog}) \end{aligned}$$

airplane



automobile



bird



cat



deer



dog



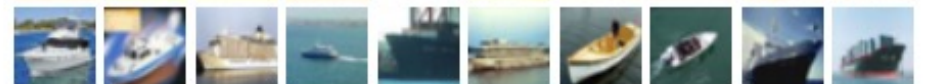
frog



horse



ship



truck



Importance Weights for Label Shift

- If we know $w(y)$, then we have

$$\mathbb{E}_Q[\ell(\theta; x, y)] = \mathbb{E}_P[\ell(\theta; x, y) \cdot w(x, y)] = \mathbb{E}_P[\ell(\theta; x, y) \cdot w(y)]$$

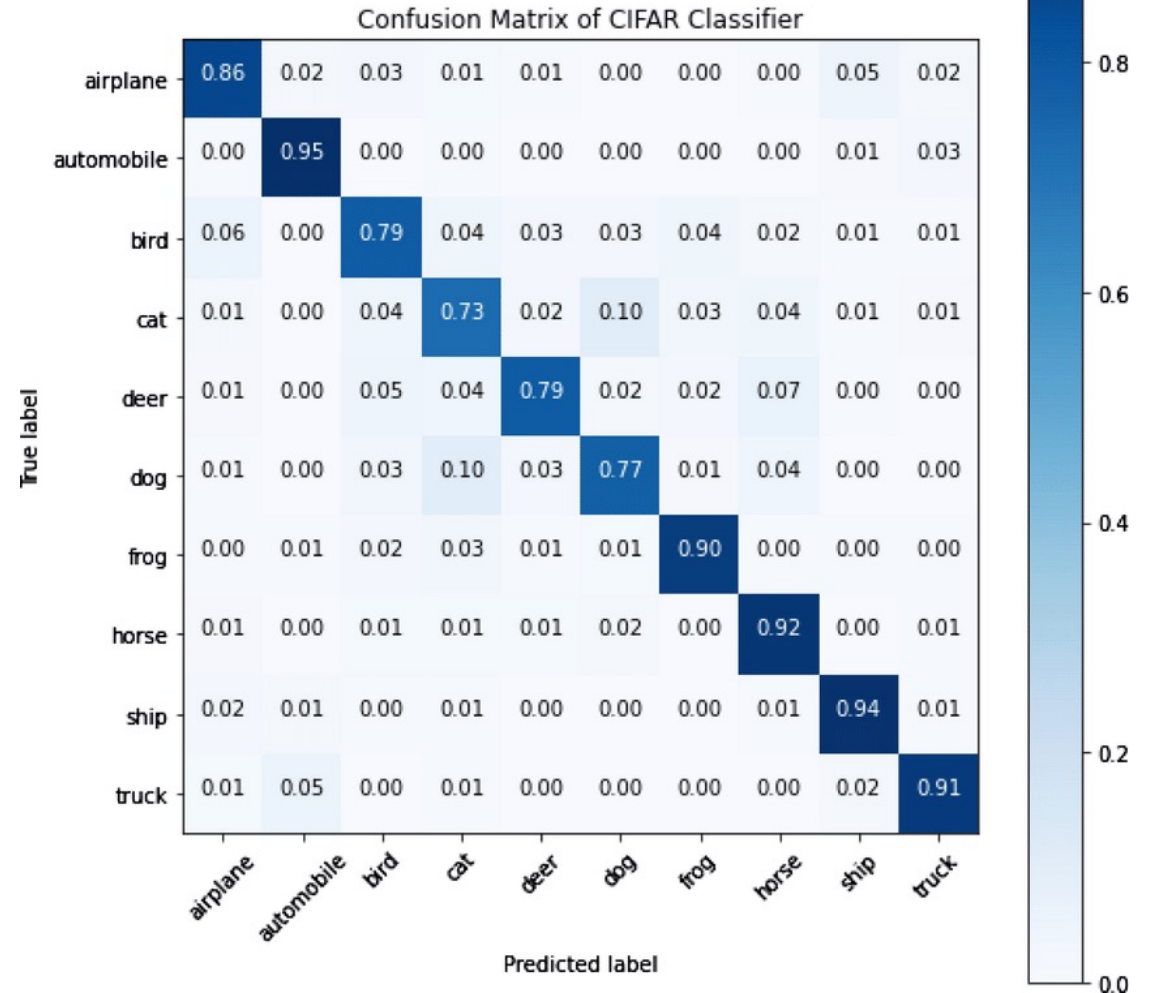
- How do we compute $w(y)$?

Importance Weights for Label Shift

- Given a classifier $f: \mathcal{X} \rightarrow \mathcal{Y}$, where $\mathcal{Y} = \{1, \dots, K\}$, consider the confusion matrix $C \in \mathbb{R}^{K \times K}$ defined by

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j]$$

- Also, define $p, q \in \mathbb{R}^K$ by
 - $p_i = \mathbb{P}_P[f(x) = i]$
 - $q_i = \mathbb{P}_Q[f(x) = i]$



Importance Weights for Label Shift

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \quad p_i = \mathbb{P}_P[f(x) = i] \quad q_i = \mathbb{P}_Q[f(x) = i]$$

- Since $f(x)$ only depends on x , we have

$$\mathbb{P}_P[f(x) = i \mid y = j]$$

Importance Weights for Label Shift

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \quad p_i = \mathbb{P}_P[f(x) = i] \quad q_i = \mathbb{P}_Q[f(x) = i]$$

- Since $f(x)$ only depends on x , we have

$$\begin{aligned} \mathbb{P}_P[f(x) = i \mid y = j] &= \int_X \mathbf{1}(f(x) = i) \cdot p(x \mid y = j) \cdot dx \\ &= \int_X \mathbf{1}(f(x) = i) \cdot q(x \mid y = j) \cdot dx \\ &= \mathbb{P}_Q[f(x) = i \mid y = j] \end{aligned}$$

Importance Weights for Label Shift

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \quad p_i = \mathbb{P}_P[f(x) = i] \quad q_i = \mathbb{P}_Q[f(x) = i]$$

$$\mathbb{P}_P[f(x) = i \mid y = j] = \mathbb{P}_Q[f(x) = i \mid y = j]$$

- Now, we have

$$q_i$$

Importance Weights for Label Shift

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \quad p_i = \mathbb{P}_P[f(x) = i] \quad q_i = \mathbb{P}_Q[f(x) = i]$$

$$\mathbb{P}_P[f(x) = i \mid y = j] = \mathbb{P}_Q[f(x) = i \mid y = j]$$

- Now, we have

$$\begin{aligned} q_i &= \sum_{j=1}^k \mathbb{P}_Q[f(x) = i \mid y = j] \cdot \mathbb{P}_Q[y = j] \\ &= \sum_{j=1}^k \mathbb{P}_P[f(x) = i \mid y = j] \cdot \mathbb{P}_Q[y = j] \\ &= \sum_{j=1}^k \frac{\mathbb{P}_P[f(x)=i, y=j]}{\mathbb{P}_P[y=j]} \cdot \mathbb{P}_Q[y = j] \\ &= \sum_{j=1}^k \mathbb{P}_P[f(x) = i, y = j] \cdot \frac{\mathbb{P}_Q[y=j]}{\mathbb{P}_P[y=j]} \end{aligned}$$

Importance Weights for Label Shift

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \quad p_i = \mathbb{P}_P[f(x) = i] \quad q_i = \mathbb{P}_Q[f(x) = i]$$

$$\mathbb{P}_P[f(x) = i \mid y = j] = \mathbb{P}_Q[f(x) = i \mid y = j]$$

- Now, we have

$$q_i = \sum_{j=1}^k \mathbb{P}_P[f(x) = i, y = j] \cdot \frac{\mathbb{P}_Q[y=j]}{\mathbb{P}_P[y=j]}$$

Importance Weights for Label Shift

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \quad p_i = \mathbb{P}_P[f(x) = i] \quad q_i = \mathbb{P}_Q[f(x) = i]$$

$$\mathbb{P}_P[f(x) = i \mid y = j] = \mathbb{P}_Q[f(x) = i \mid y = j]$$

- Now, we have

$$q_i = \sum_{j=1}^k \mathbb{P}_P[f(x) = i, y = j] \cdot \frac{\mathbb{P}_Q[y=j]}{\mathbb{P}_P[y=j]}$$

Importance Weights for Label Shift

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \quad p_i = \mathbb{P}_P[f(x) = i] \quad q_i = \mathbb{P}_Q[f(x) = i]$$

$$\mathbb{P}_P[f(x) = i \mid y = j] = \mathbb{P}_Q[f(x) = i \mid y = j]$$

- Now, we have

$$q_i = \sum_{j=1}^k \mathbb{P}_P[f(x) = i, y = j] \cdot w(j)$$

$$\begin{bmatrix} q_1 \\ \vdots \\ q_K \end{bmatrix} = \begin{bmatrix} \mathbb{P}_P[f(x) = 1, y = 1] & \cdots & \mathbb{P}_P[f(x) = 1, y = K] \\ \vdots & \ddots & \vdots \\ \mathbb{P}_P[f(x) = K, y = 1] & \cdots & \mathbb{P}_P[f(x) = K, y = K] \end{bmatrix} \begin{bmatrix} w(1) \\ \vdots \\ w(K) \end{bmatrix}$$

$$q = Cw$$

Importance Weights for Label Shift

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \quad p_i = \mathbb{P}_P[f(x) = i] \quad q_i = \mathbb{P}_Q[f(x) = i]$$

$$\mathbb{P}_P[f(x) = i \mid y = j] = \mathbb{P}_Q[f(x) = i \mid y = j]$$

- Now, we have

$$q_i = \sum_{j=1}^k \mathbb{P}_P[f(x) = i, y = j] \cdot w(j)$$

$$\begin{bmatrix} q_1 \\ \vdots \\ q_K \end{bmatrix} = \begin{bmatrix} \mathbb{P}_P[f(x) = 1, y = 1] & \cdots & \mathbb{P}_P[f(x) = 1, y = K] \\ \vdots & \ddots & \vdots \\ \mathbb{P}_P[f(x) = K, y = 1] & \cdots & \mathbb{P}_P[f(x) = K, y = K] \end{bmatrix} \begin{bmatrix} w(1) \\ \vdots \\ w(K) \end{bmatrix}$$

$$q = Cw \Rightarrow w = C^{-1}q$$

Importance Weights for Label Shift

$$C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \quad p_i = \mathbb{P}_P[f(x) = i] \quad q_i = \mathbb{P}_Q[f(x) = i]$$

$$\mathbb{P}_P[f(x) = i \mid y = j] = \mathbb{P}_Q[f(x) = i \mid y = j]$$

- Now, we have

$$w = C^{-1}q$$

Supervised Learning with Label Shift

- **Input:** Training dataset Z , unlabeled test dataset X
- **Step 1:** Train f on Z
- **Step 2:** Estimate using the dataset:
 - $C_{ij} = \mathbb{P}_P[f(x) = i, y = j] \approx |Z|^{-1} \sum_{(x,y) \in Z} \mathbf{1}(f(x) = i \wedge y = j)$
 - $q_i = \mathbb{P}_Q[f(x) = i] \approx |X|^{-1} \sum_{x \in X} \mathbf{1}(f(x) = i)$
- **Step 3:** Compute $w = C^{-1}q$
- **Step 4:** Compute $\hat{\theta} = \arg \min_{\theta} \sum_{(x,y) \in Z} \ell(\theta; x, y) \cdot w(y)$