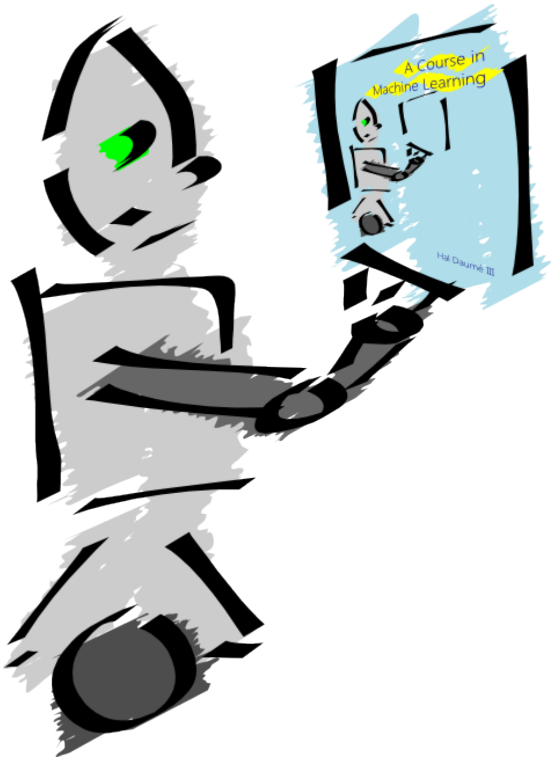
Machine Learning II



HAL DAUMÉ III, UMD
me@hal3.name
http://hal3.name
@haldaume3

Credit: many slides due to Marine Carpuat (UMD) or John Blitzer (Google)
Loosely in parts based on A Course in Machine Learning (ciml.info)
And "Understanding Machine Learning" (SSS & SBD)

What is this course about?

Machine learning studies algorithms for learning to do stuff

- By finding (and exploiting) patterns in data
- Sometimes in ways we'd rather they didn't
- Theory helps us understand this!

Last time….

- What does it mean to learn?
- Inductive bias
- Linear models
- Overfitting & underfitting

Formalizing Induction

- Given
  - a loss function l
  - a sample from some unknown data distribution D

- Our task is to compute a function f that has low expected error over D with respect to l.

$$\mathbb{E}_{(x,y)\sim D}\{l(y,f(x))\} = \sum_{(x,y)} D(x,y)l(y,f(x))$$

Overfitting

- Consider a hypothesis *h* and its:
  - Error rate over training data
  - True error rate over all data

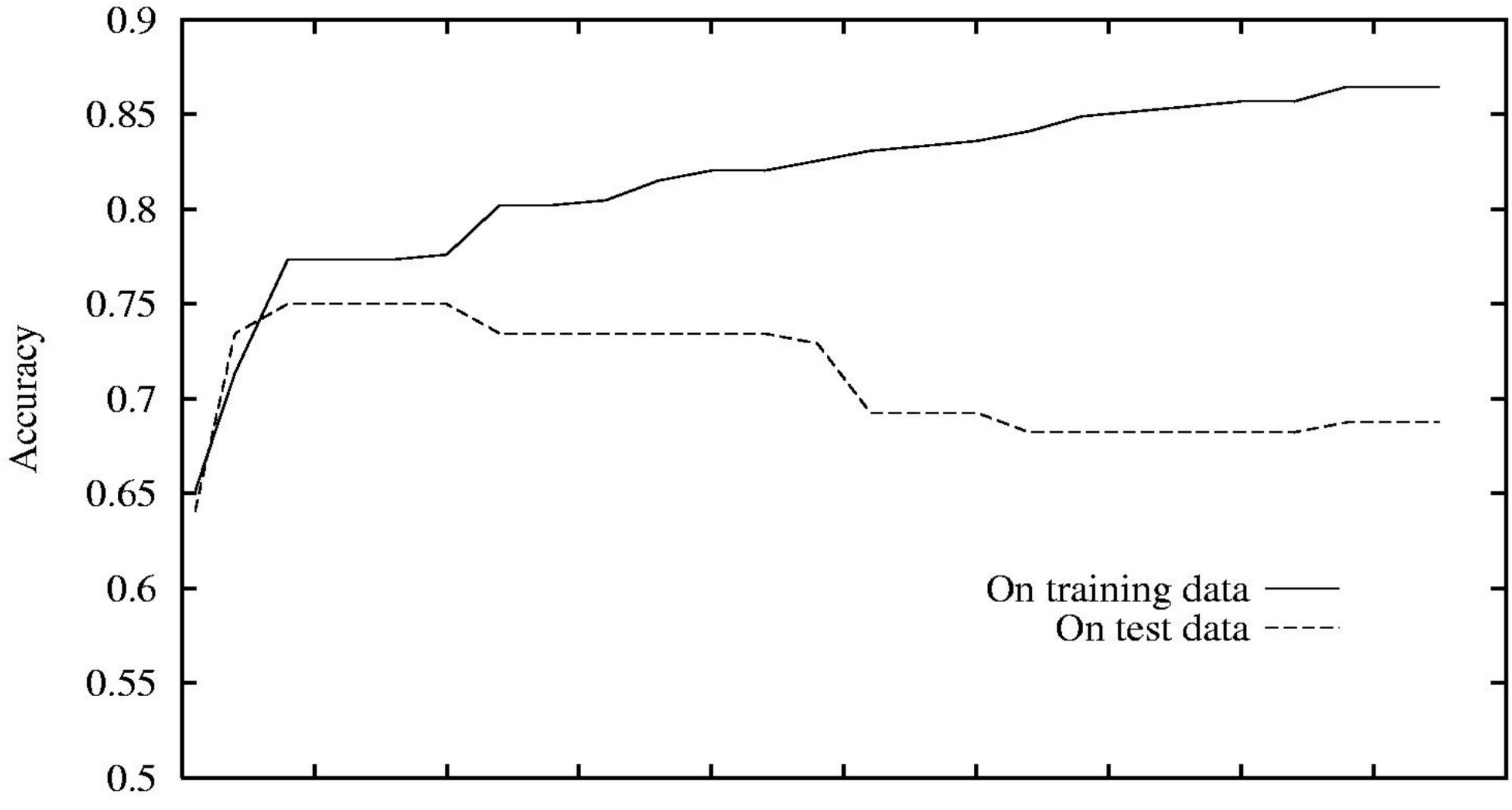
- We say h overfits the training data if

  Training error << Test error

- Amount of overfitting =

  Test error – Training error

Measuring effect of overfitting in linear models



Lots ← Amount of Regularization → Very Little

Formalizing Errors

The learned classifier

$\mathcal{F}$ set of all possible classifiers sing a fixed representation

$$\text{error}(f) = \underbrace{\left[\text{error}(f) - \min_{f^* \in \mathcal{F}} \text{error}(f^*)\right]}_{\textbf{estimation error}} + \underbrace{\left[\min_{f^* \in \mathcal{F}} \text{error}(f)\right]}_{\textbf{approximation error}}$$

How far is the learned classifier f from the optimal classifier f*?

Quality of the model family
aka hypothesis class

The bias/variance trade-off

- Trade-off between
  - approximation error (bias)
  - estimation error (variance)

- Example:
  - Consider the learning algorithm that always returns the "always positive classifier"
    - Low variance as a function of a random draw of the training set
    - Strongly biased toward predicting +1 no matter what the input

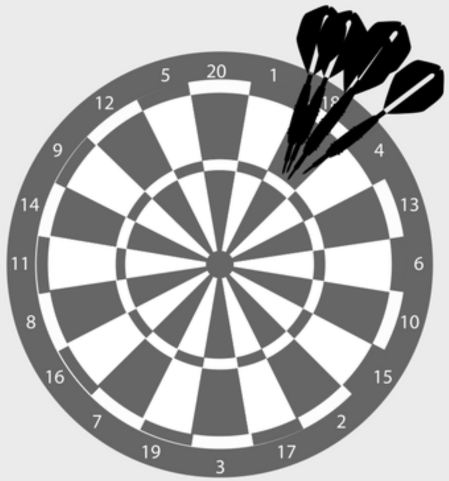
Ok, let's do a thought experiment...

Imagine you've collected 5 different training datasets for the same problem. Now, imagine using one algorithm to train 5 models (one for each training set).

Here's what those 5 models tell you about your chosen algorithm:

High Bias

Low Variance



High bias, low variance algorithms train models that are consistent, but inaccurate *on average*.

High Variance

Low Bias

High variance, low bias algorithms train models that are accurate *on average*, but inconsistent.

Source: elitedatascience.com

Today…

- Quantifying what can and cannot be learned
  - No free lunch
  - VC dimension
- What are our core assumptions / how to break them
- How to unbreak (some of) them
  - Sample selection bias
  - Covariate shift

No free lunch

Thm: Let A be any learning algorithm for binary classification with 0/1 loss over X, and let $m < |X|/2$ be the training set size. Then, there exists D such that:

1. There exists f st $L_D(f) = 0$
2. With prob at least 1/7 over choice of $S \sim D^m$, we have $L_D(A(S)) > 1/8$

No free lunch – why?

Thm: Let *A* be any learning algorithm, let $m < |X|/2$ be the training size. Then, exists *D* st: (1) exists good *f* and (2) *A* doesn't find it.

- Pick set C of size 2m, consider all f : C → {0,1}
- Consider $D_{C,f}$ that puts all mass on { (x, f(x)) : x in C }
- Based on $S \sim D_{c,f}^{m}$, can only distinguish half such *f*s
- Given "test data", might get ½ correct due to memorization, and get ½ of the rest correct by luck
- So expected loss is at least ¼
- Some simple bounds complete the statement

How to block NFL?

Thm: Let *A* be any learning algorithm, let $m < |X|/2$ be the training size. Then, exists *D* st: (1) exists good *f* and (2) *A* doesn't find it.

- Pick set C of size 2m, consider all f : C → {0,1}
- Consider $D_{C,f}$ that puts all mass on { (x, f(x)) : x in C }
- Based on $S \sim D_{c,f}^m$, can only distinguish half such *f*s
- Given "test data", might get ½ correct due to memorization, and get ½ of the rest correct by luck
- So expected loss is at least ¼
- Some simple bounds complete the statement

How do we block NFL?

Def (Shattering): Let H be a set of functions X → {0,1} and let C bet a subset of X. H shatters C if H contains all functions C → {0,1}.

Thm (NFL restated): Let A be a learning algorithm that outputs a function in H. If there exists a set C of size 2m that is shattered by H, then NFL applies.

Goal: make sure that no large sets are shattered by H.

Def (VC-dimension): VCdim(H) = size of largest C that is shattered by H.

What does VC buy us?

Def (VC-dimension): VCdim(H) = size of largest C that is shattered by H.

Thm: Assume H has VCdim d, and we have N iid training examples, then with probability at least δ over choice of training data an any internal randomization, empirical risk minimization (ERM) has:

$$error^{test} \leq error^{train} + \sqrt{\frac{8 \log d + 8 \log \frac{4}{\delta}}{N}}$$

Often called the "fundamental theorem of statistical learning"

Assumptions = vulnerabilities

What does the Fundamental Theorem of Statistical Learning assume?

- Training distribution matches test distribution
- What we care about is zero/one loss
- Number of training examples grows like sqrt(log(d))
- Training set is iid
- We don’t get unlucky

ACM Code of Ethics

“To minimize the possibility of indirectly harming others, computing professionals must minimize malfunctions by following generally accepted standards for system design and testing. Furthermore, it is often necessary to assess the social consequences of systems to project the likelihood of any serious harm to others. If system features are misrepresented to users, coworkers, or supervisors, the individual computing professional is responsible for any resulting injury.”

https://www.acm.org/about-acm/acm-code-of-ethics-and-professional-conduct

Immigration and Customs Enforcement's Homeland Security Investigations "Industry Day"

EXTREME VETTING INITIATIVE –
OVERARCHING VETTING

Extreme Vetting Initiative Objectives (cont.)

Performance Objectives of the Overarching Vetting Contract:

1. Centralizes screening and vetting processes to mitigate case backlog and provide law enforcement and field agents with timely, actionable information;

2. Allows ICE to develop richer case files that provide more value-added information to further investigations or support prosecutions in immigration or federal courts;

3. Allows ICE to perform regular, periodic and/or continuous review and vetting of nonimmigrants for changes in their risk profile after they enter the United States and;

4. Automates at no loss of data quality or veracity any manually-intensive vetting and screening processes that inhibit ICE from properly and thoroughly vetting individuals in a timely fashion.

ICE

https://theintercept.com/2017/08/07/these-are-the-technology-firms-lining-up-to-build-trumps-extreme-vetting-program/

Real AI...

1) Make AI vastly capable
2) Make vastly capable AI beneficial

1) Make AI beneficial
2) Make beneficial AI vastly capable

Slide credit:
Margaret Mitchell
m-mitchell.com

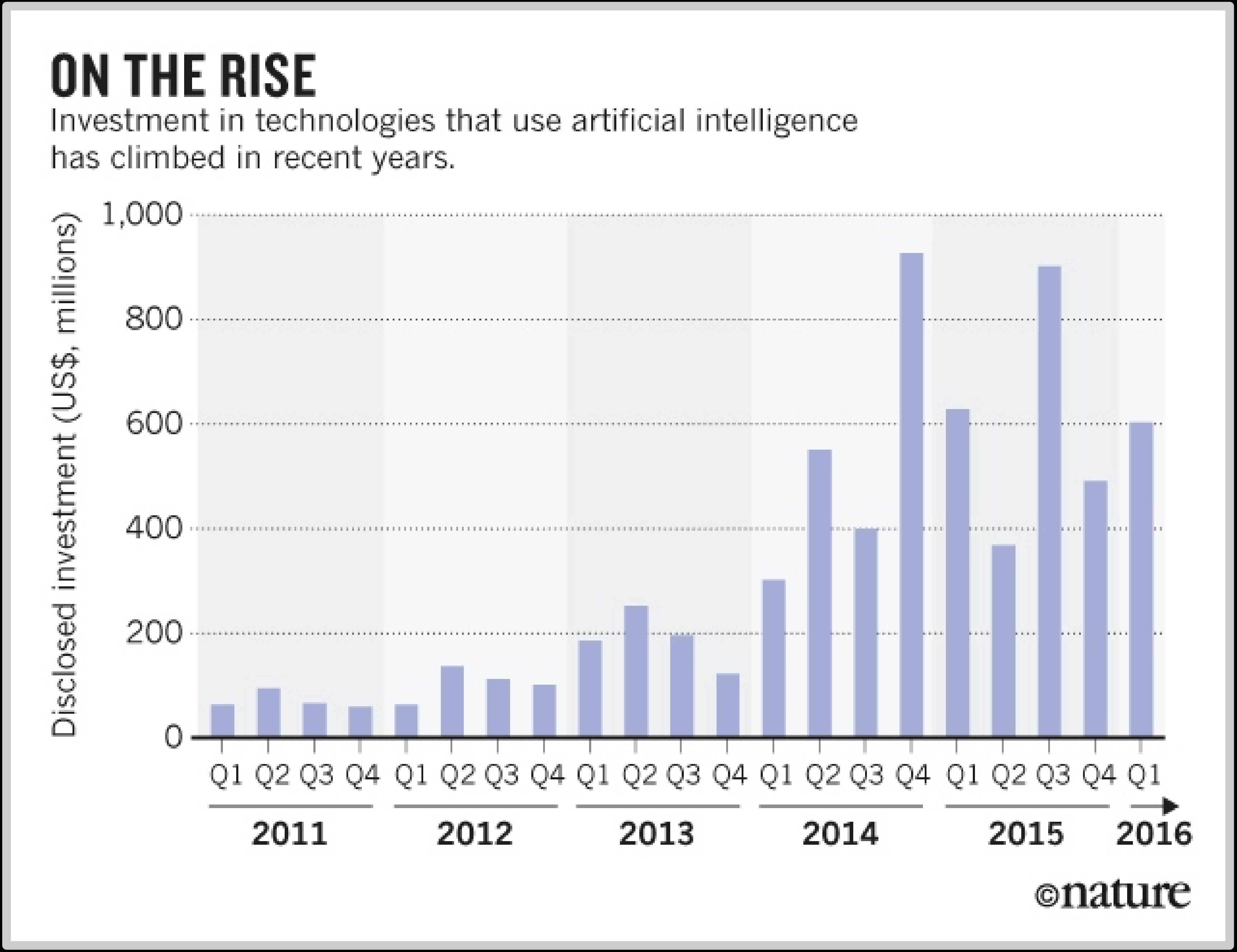
Train/Test Mismatch

- When working with real data, training sample
  - reflects human biases
  - is influenced by practical concerns
    - e.g., what kind of data is easy to obtain

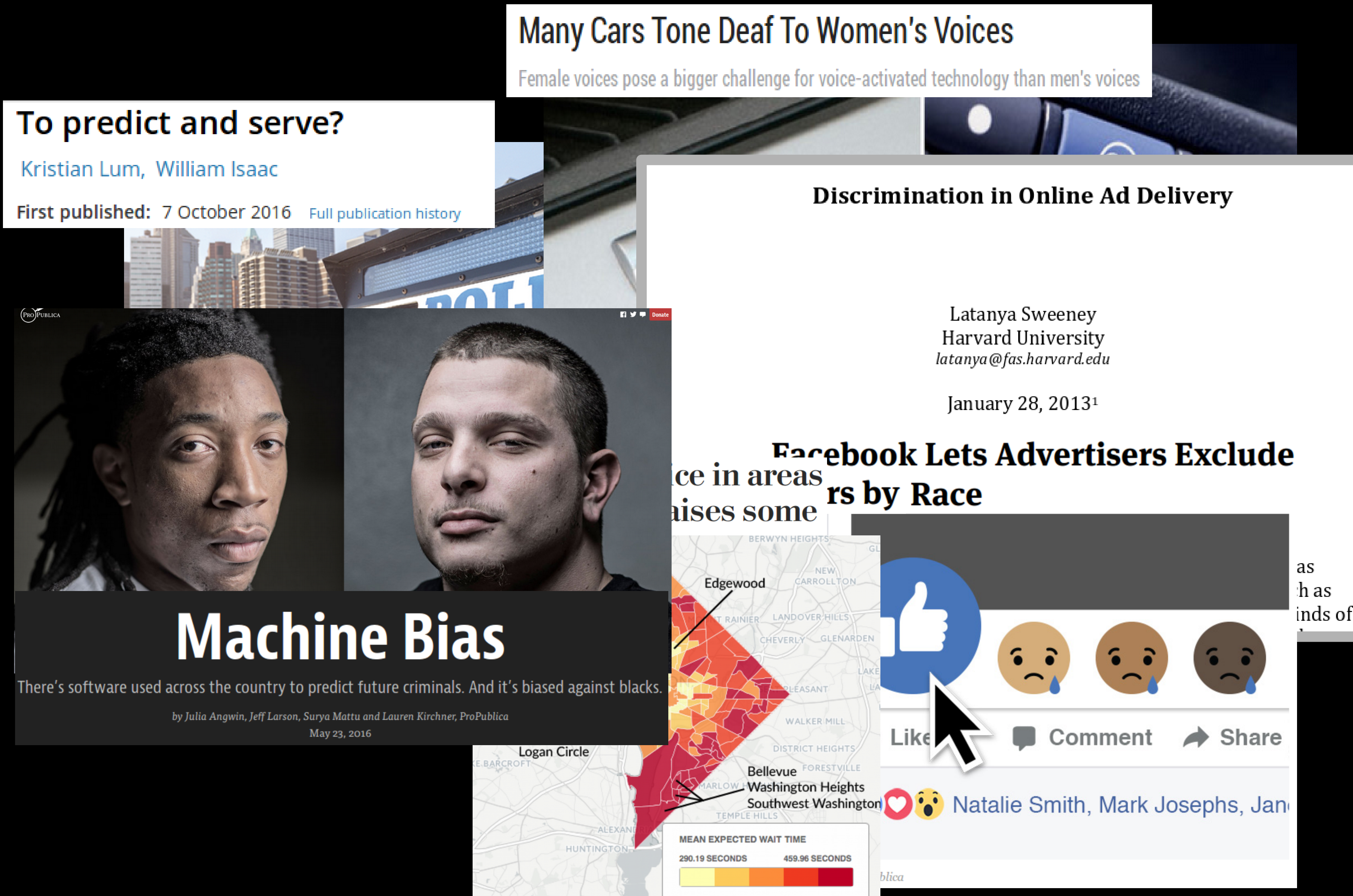
bbc.com/news/technology-40416606

- Train/test distribution mismatch is frequent issue
  - aka covariate shift, sample selection bias, domain adaptation

the age of automated decision making



things can go really badly



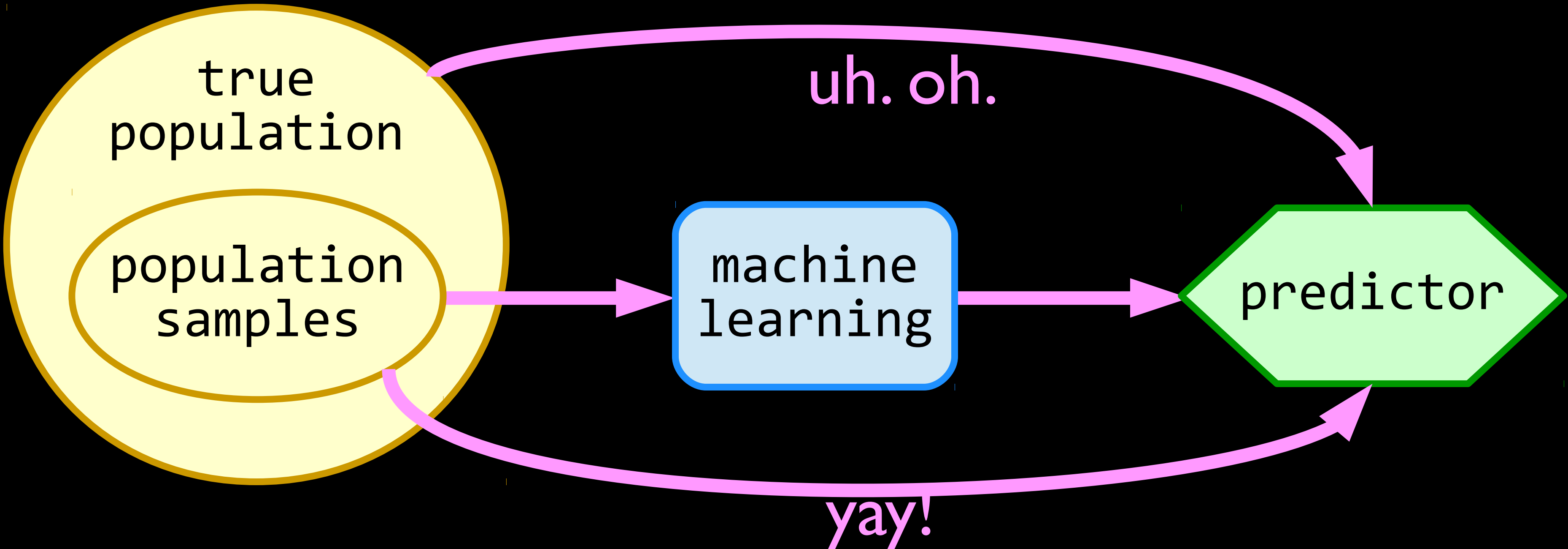
three (out of many) sources of bias

data collection

objective function

feedback loops

sample selection bias

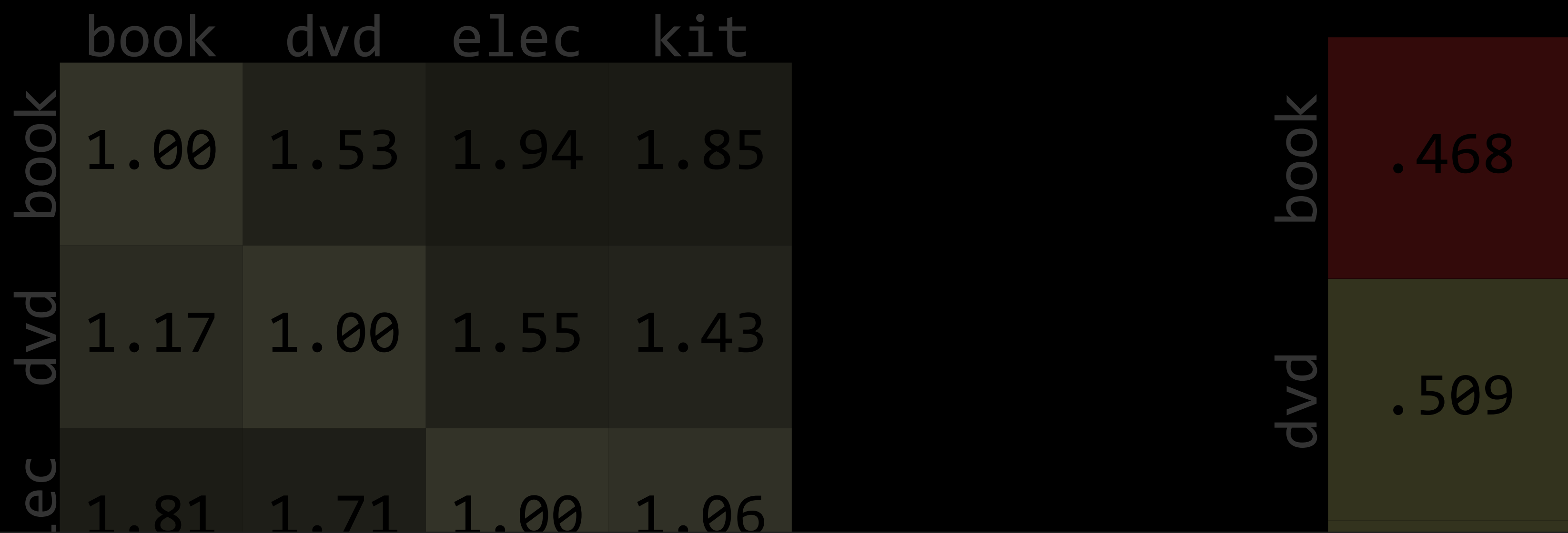


James Heckman,
Nobel prize econ
(2000)
Sample selection bias as specification error.
Econometrica
(1979)

Corinna Cortes,
Domain adaptation and sample bias correction theory and algorithm for regression
TCS, 2013

it's not just that error rate goes up…

train on electronics
rate of positive predictions

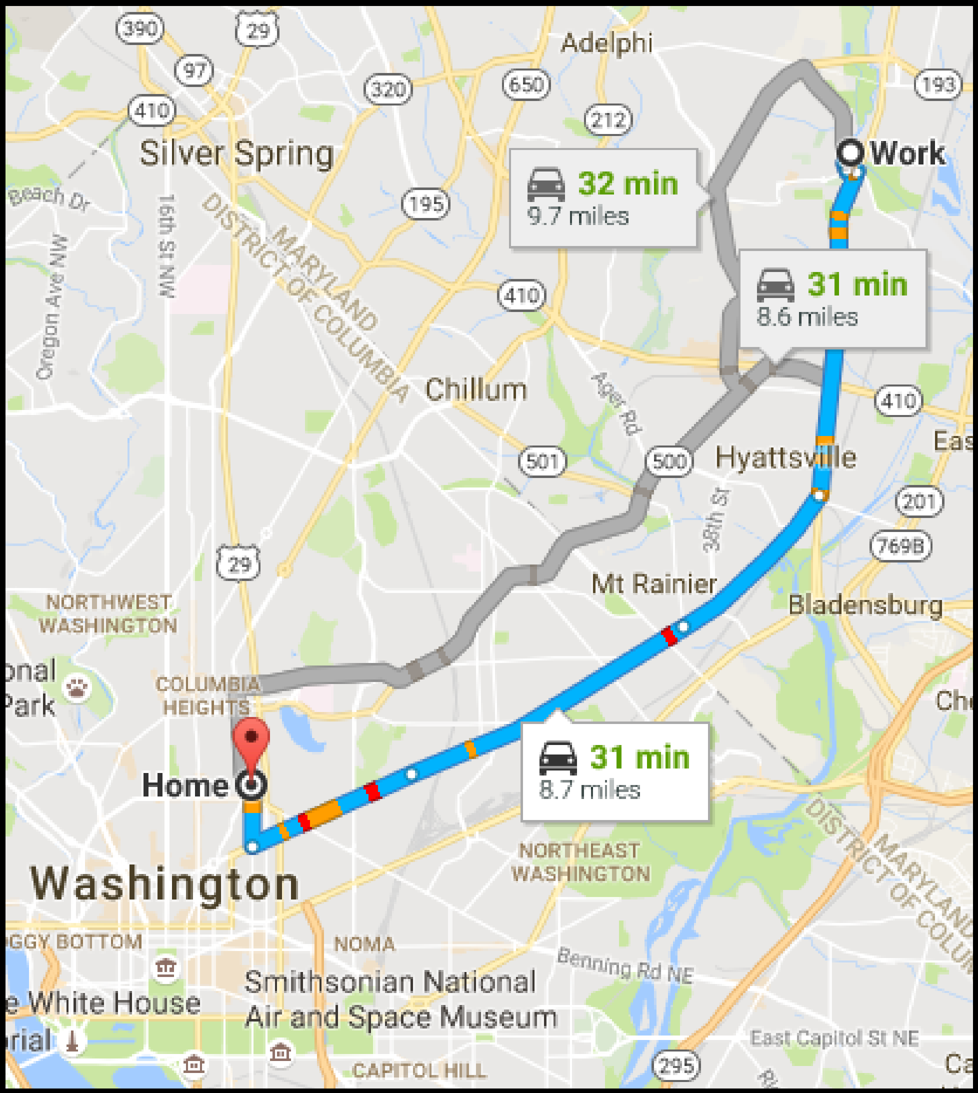


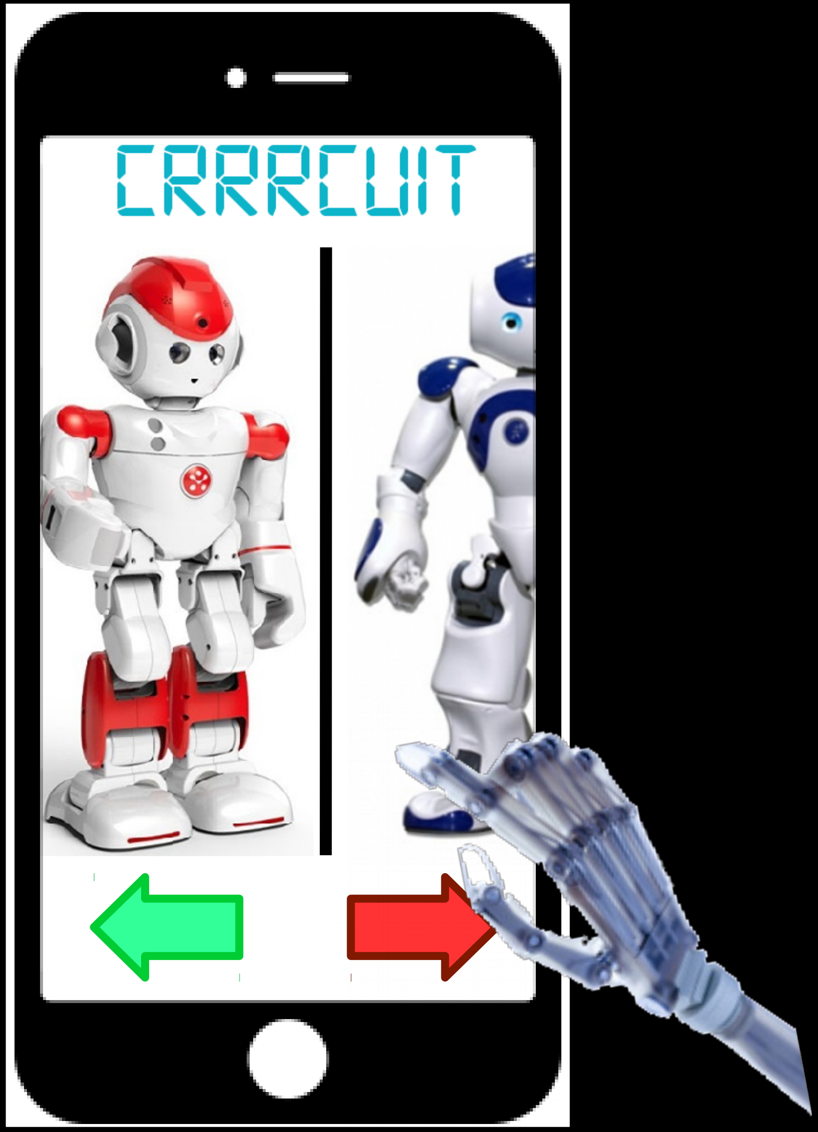
Prediction Fails Differently for Black Defendants

	WHITE	AFRICAN AMERICAN
Labeled Higher Risk, But Didn't Re-Offend	23.5%	44.9%
Labeled Lower Risk, Yet Did Re-Offend	47.7%	28.0%

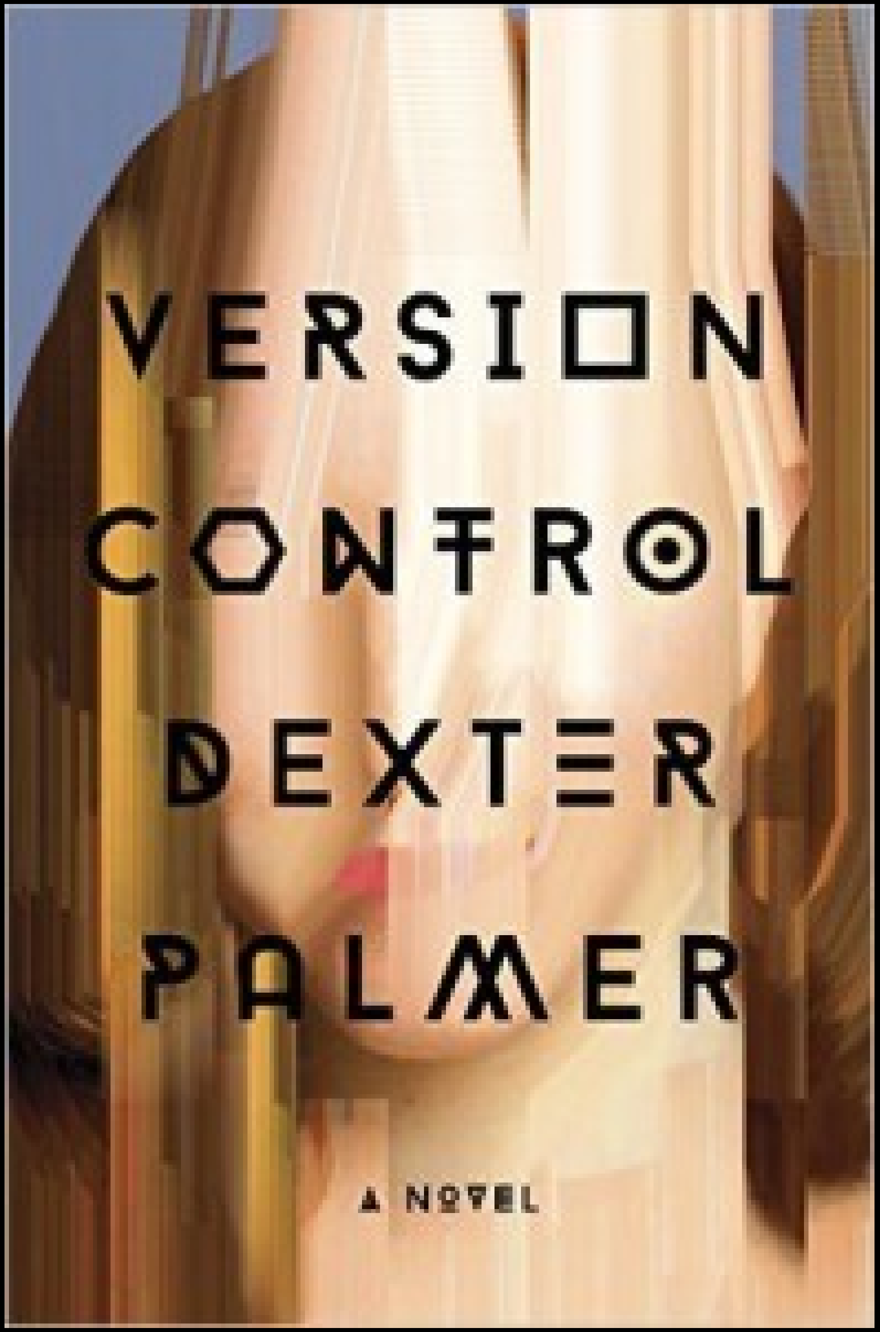
Source: Propublica, "Machine Bias"

what are we optimizing for?





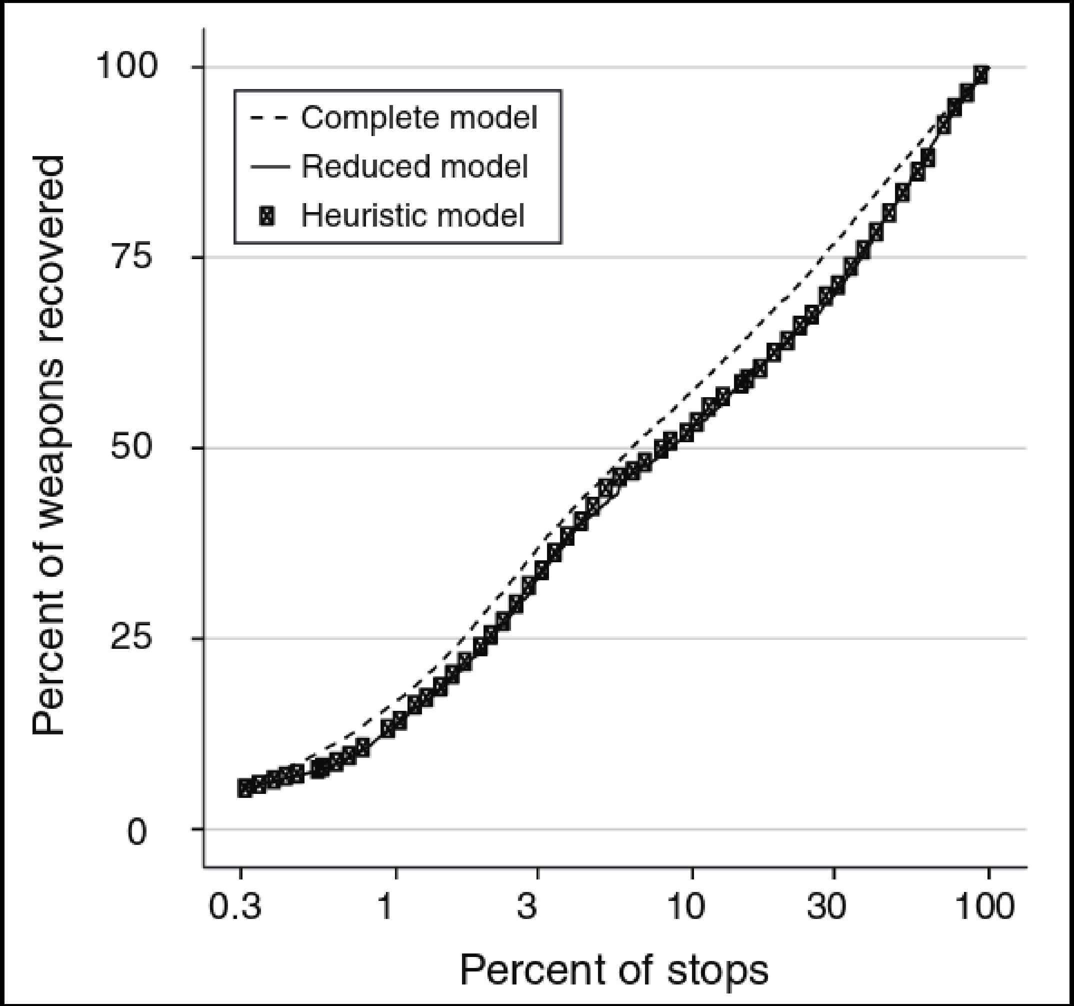
what are we optimizing for?



feedback loops in stop+frisk

Can we reduce the number of (and bias in) stops under a stop and frisk policy?

What happens if/when police officers start using this system?



Personalized risk assessments in the criminal justice system
Goel, Rao & Shroff; American Economic Review, 2016

three (out of many) sources of bias

data collection

objective function

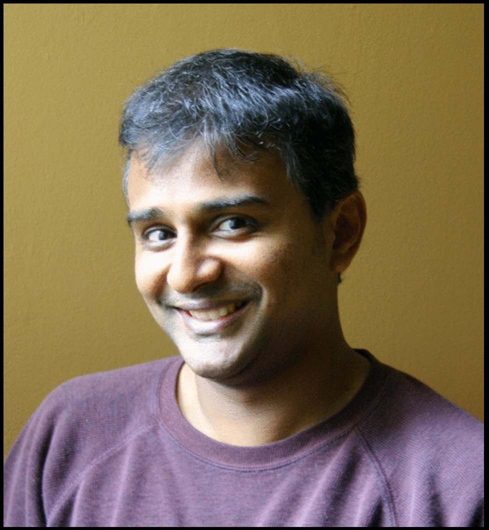
feedback loops

Joanna Bryson
@j2bryson

Kate Crawford
@katecrawford

Nick Diakopoulos
@ndiakopoulos

Sorelle Friedler
@kdphd



Suresh Venkat
@geomblog

Hanna Wallach
@hannawallach

Fairness, Accountability & Transparency in ML
fatml.org

Critical Algorithm Studies: A Reading List
socialmediacollective.org/reading-lists/critical-algorithm-studies

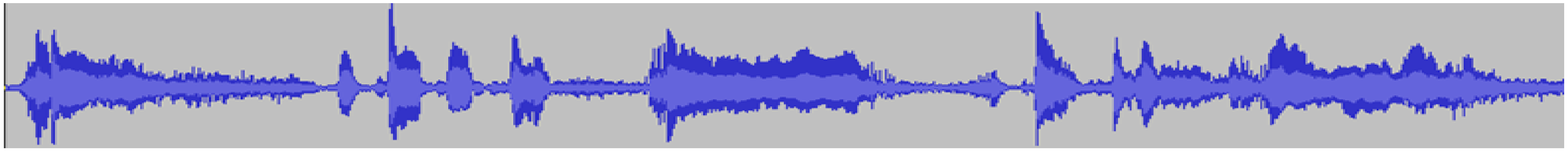
Classical "Single-domain" Learning

Predict: $x \to y$ $(x, y) \sim \Pr[x, y]$

amazon.com

Running with Scissors **Title:** Horrible book, horrible.

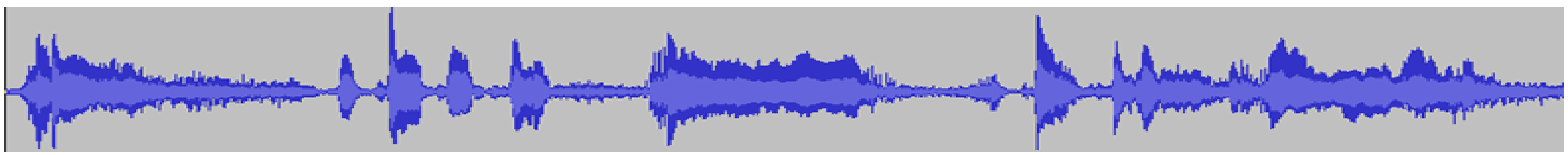
This book was horrible. I read half, suffering from a headache the entire time, and eventually i lit it on fire. 1 less copy in the world. Don't waste your money. I wish i had the time spent reading this book back. It wasted my life



Domain Adaptation

$$(x, y) \sim \Pr_S[x, y]$$

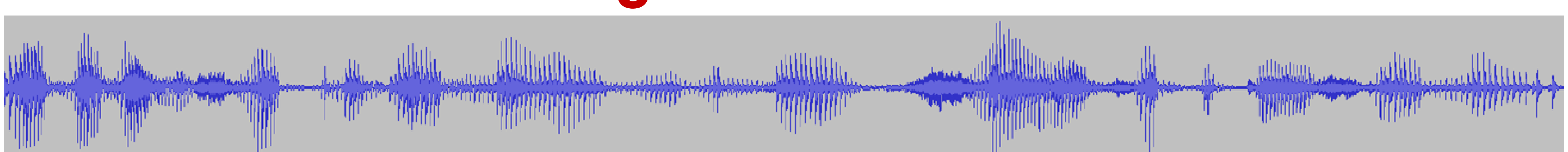
Training **Source**



So the topic of ah the talk today is online learning

$$(x, y) \sim \Pr_T[x, y]$$

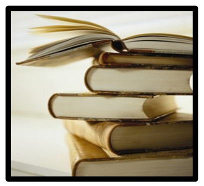
Testing **Target**



Everything is happening online. Even the slides are produced on-line

Domain Adaptation

Natural Language Processing

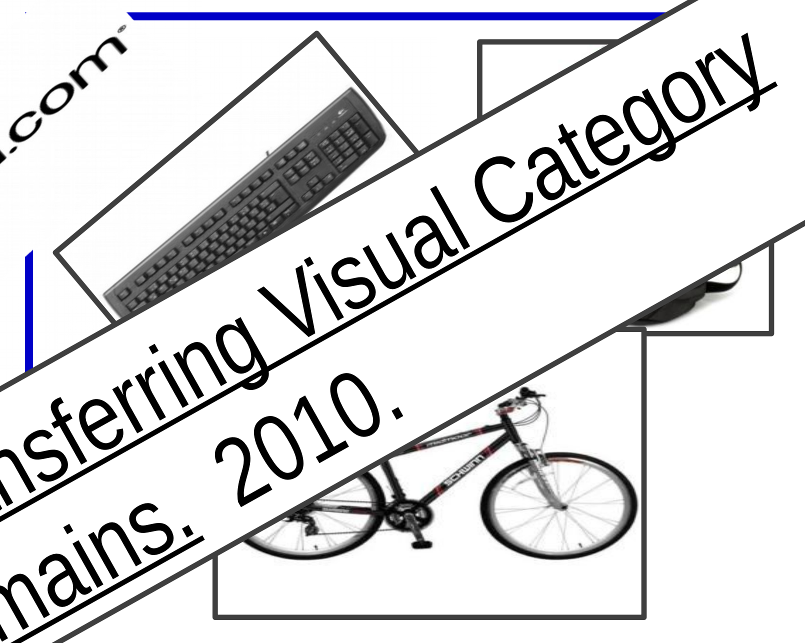


Packed with fascinating info

A breeze to clean up


THE WALL STREET JOURNAL
Cracking the Neanderthal Code
A Mystery Multiplies Over a Math Prize
What's News
As Emission Restrictions Loom, Texas Utility Bets Big on Coal

Visual Object Recognition



K. Saenko et al. Transferring Visual Category Models to New Domains. 2010.

Classical vs Adaptation Error

Classical Test Error:

$$\epsilon_{\text{test}} \leq \hat{\epsilon}_{\text{train}} + \sqrt{\frac{\text{complexity}}{n}}$$

Measured on the same distribution!

Adaptation Target Error:

$$\epsilon_{\text{test}} \leq ??$$

Measured on a **new** distribution!

Common Concepts in Adaptation

Covariate Shift

$$\Pr_S[y|x] = \Pr_T[y|x]$$

 understands  & 

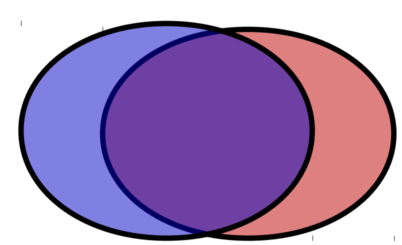
Single Good Hypothesis

$$\exists h^*, \epsilon_S(h^*),\ \epsilon_T(h^*) \text{ small}$$

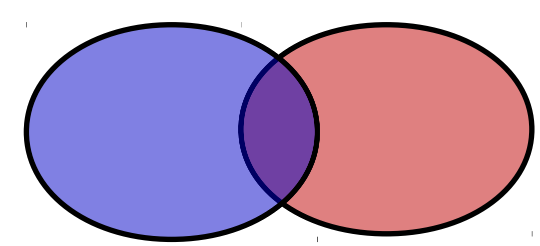
 understands  & 

Domain Discrepancy and Error

Easy



Hard



A bound on the adaptation error

Let h be a binary hypothesis. If $\Pr_S(y|x) = \Pr_T(y|x)$, then

$$\epsilon_T(h) \quad \leq \quad \epsilon_S(h) + \int_{\mathcal{X}} \left|\Pr_T(x) - \Pr_S(x)\right| dx$$

Minimize the total variation

Covariate Shift with Shared Support

Assumption: Target & Source Share Support

$$\forall x, \ \Pr_S[x] \neq 0 \text{ iff } \Pr_T[x] \neq 0$$

Reweight source instances to minimize discrepancy

Source Instance Reweighting

Defining Error

$$\epsilon_T(h) = \mathbb{E}_{\Pr_T[x]}\ \mathbb{E}_{\Pr[y|x]}\ [h(x) \neq y]$$

Using Definition of Expectation

$$= \sum_x \Pr_T[x]\, \mathbb{E}_{\Pr[y|x]}\ [h(x) \neq y]$$

Multiplying by 1

$$= \sum_x \frac{\Pr_S[x]}{\Pr_S[x]} \Pr_T[x] \mathbb{E}_{\Pr[y|x]}\ [h(x) \neq y]$$

Rearranging

$$\epsilon_T(h) = \epsilon_S(h, w) = \mathbb{E}_{\Pr_S[x]}\ \frac{\Pr_t[x]}{\Pr_s[x]}\ \mathbb{E}_{\Pr[y|x]}\ [h(x) \neq y]$$

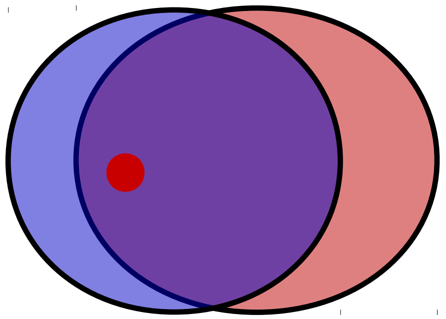
per-instance weights w

Sample Selection Bias

Another Way to View

1) Draw from the target

$\mathrm{Pr}_T[x]$



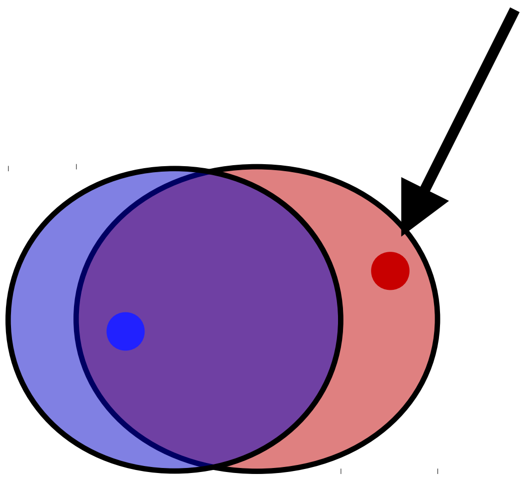
Sample Selection Bias

Redefine the source distribution

1) Draw from the target $\Pr_T[x]$

2) Select into the source with

$$\Pr[\sigma = 1|x]$$



$$\Pr_S[x] = \frac{\Pr_T[x]\Pr[\sigma = 1|x]}{\Pr[\sigma = 1]} = \Pr_T[x|\sigma = 1]$$

Rewriting Source Risk

$$\Pr_S[x] = \frac{\Pr_T[x]\Pr[\sigma = 1|x]}{\Pr[\sigma = 1]}$$

Rearranging

$$\frac{\Pr_T[x]}{\Pr_S[x]} = \frac{\Pr[\sigma = 1]}{\Pr[\sigma = 1|x]}$$

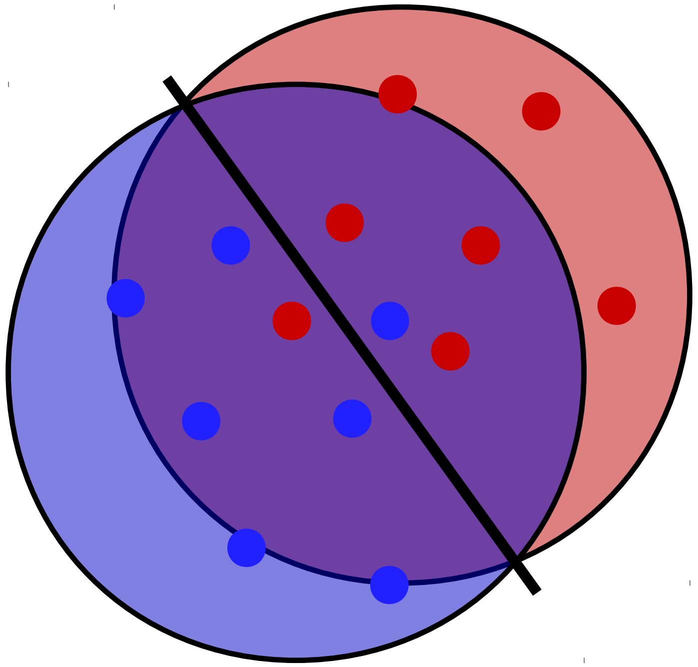
per-instance weights w

$\Pr[\sigma = 1]$ not dependent on x

$$\epsilon_S(h, w) \propto \mathbb{E}_{\Pr_S[x]} \frac{1}{\Pr[\sigma = 1|x]} \mathbb{E}_{\Pr[y|x]} [h(x) \neq y]$$

Logistic Model of Source Selection

$$\Pr[\sigma = 1 | x] = \frac{1}{1+\exp(\theta^\top x + b)}$$



Training Data

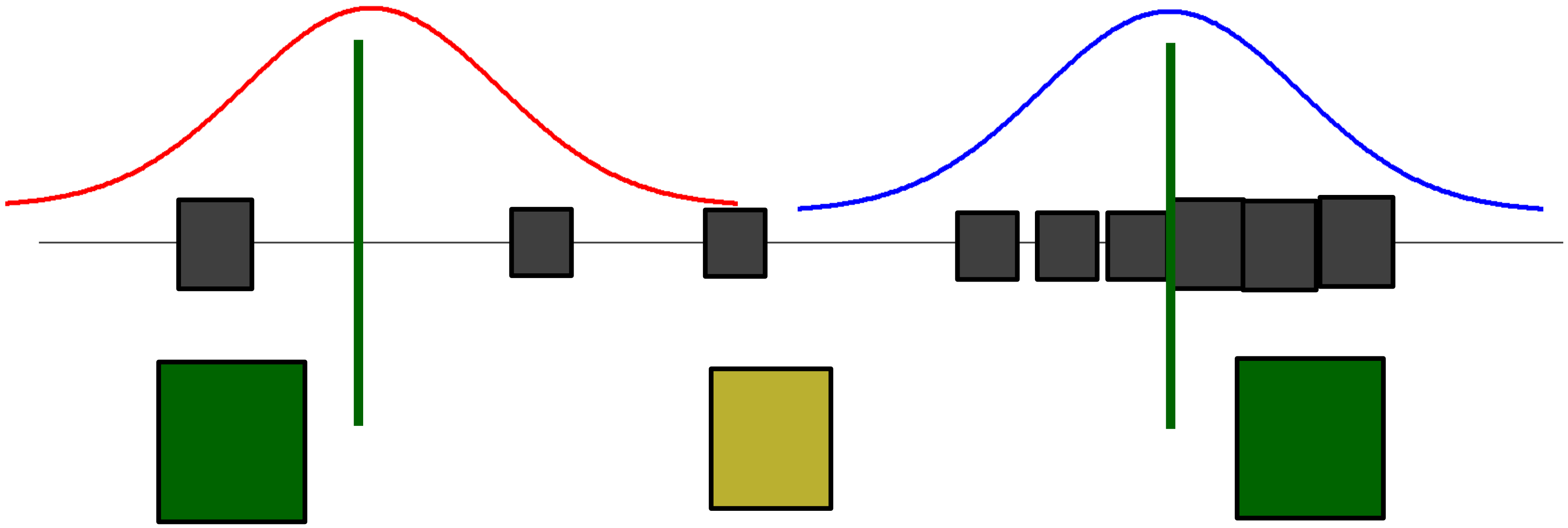
Source instances, $\sigma = 1$

Target unlabeled instances, $\sigma = 0$

Selection Bias Correction Algorithm

Input:

Labeled source data

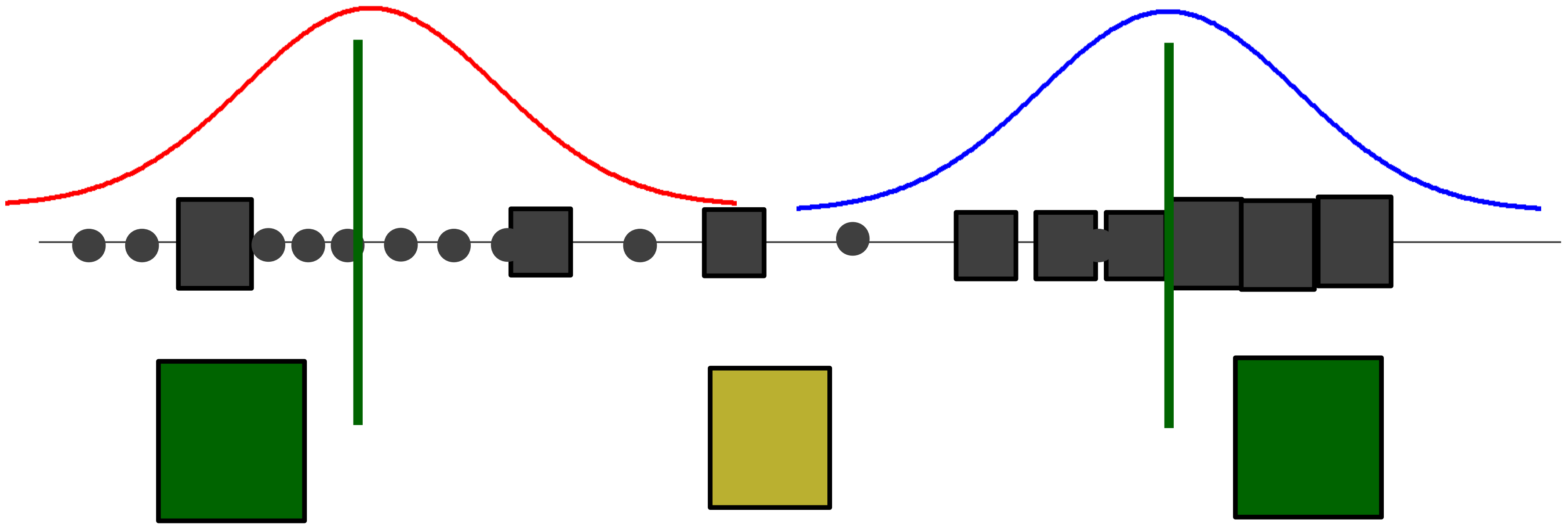


Selection Bias Correction Algorithm

Input:

Labeled source data

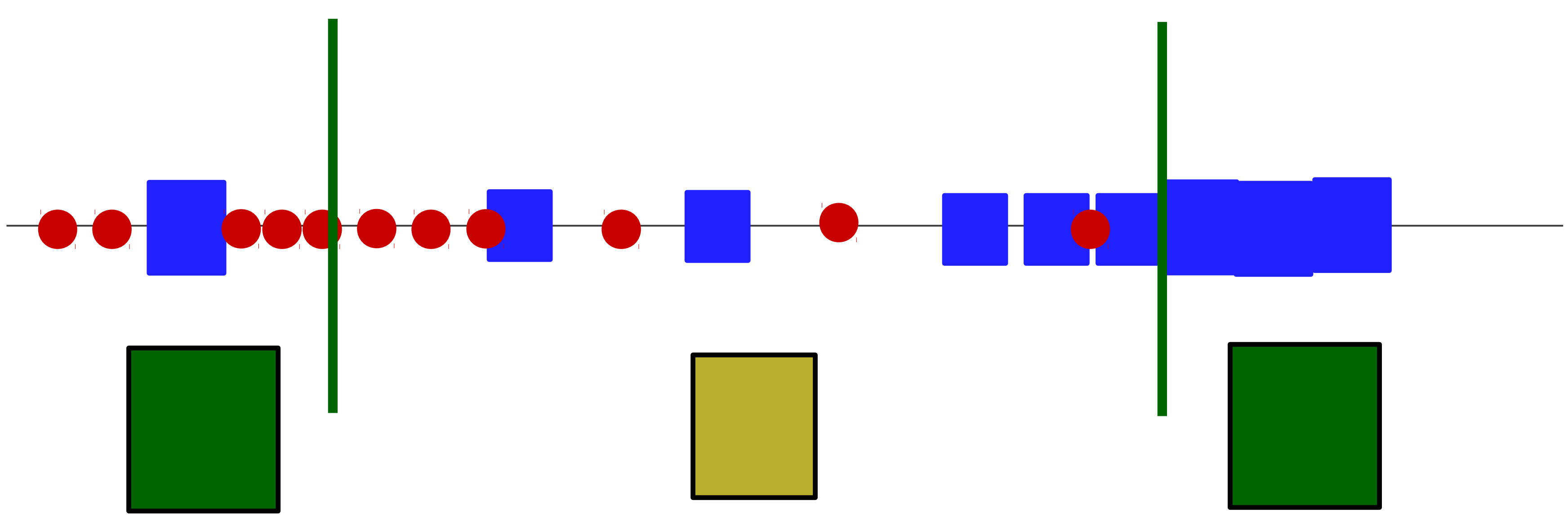
Unlabeled target data



Selection Bias Correction Algorithm

Input: Labeled source and unlabeled target data

1) Label source instances as $\sigma = 1$, target as $\sigma = 0$

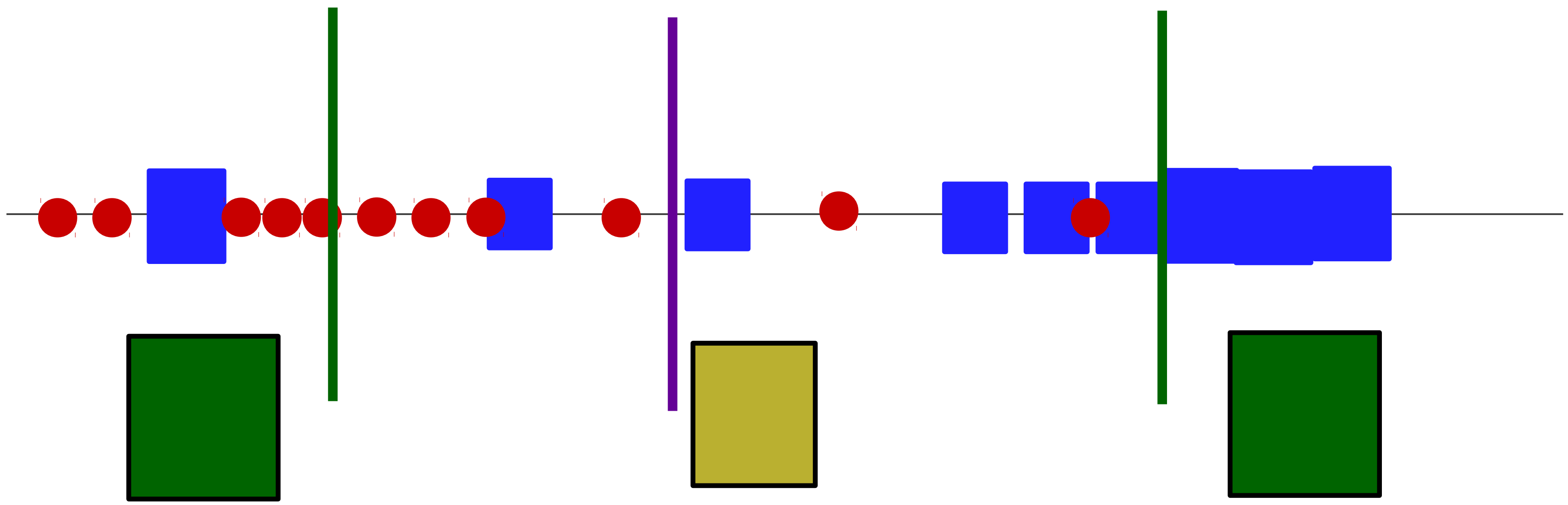


Selection Bias Correction Algorithm

Input: Labeled source and unlabeled target data

1) Label source instances as $\sigma = 1$, target as $\sigma = 0$

2) Train predictor $\Pr[\sigma = 1 | x] = \frac{1}{1+\exp(\theta^\top x + b)}$



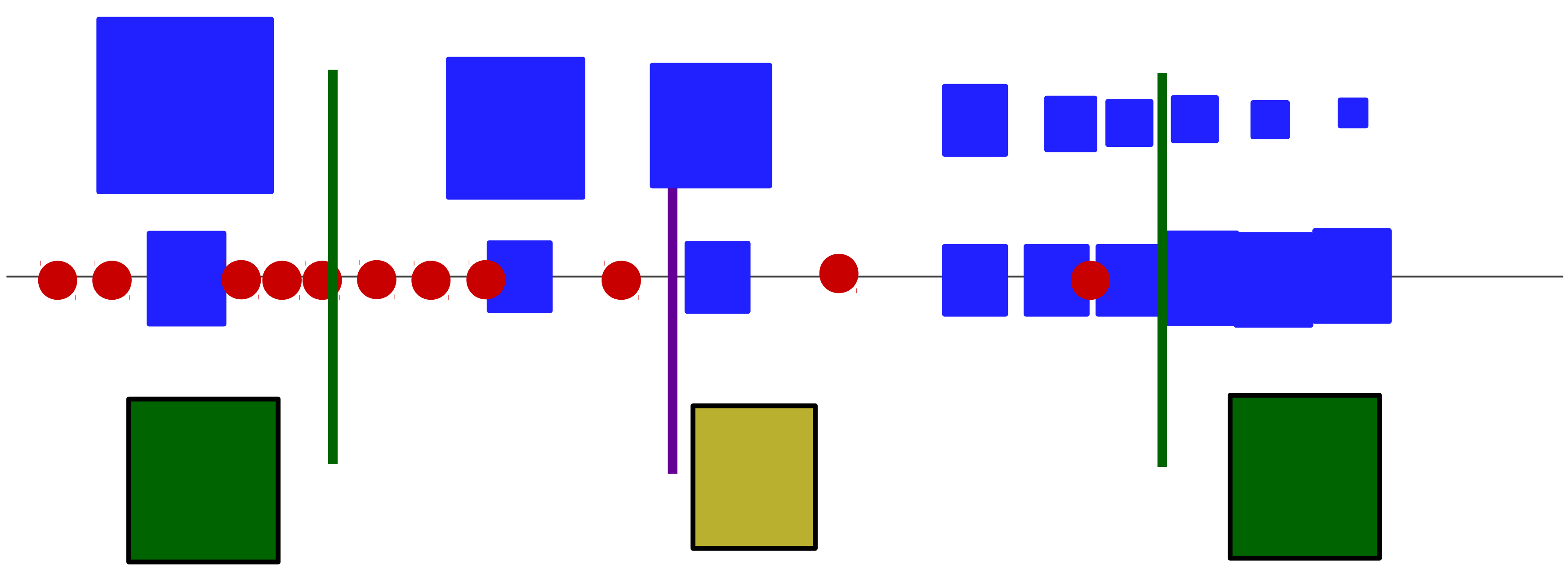
Selection Bias Correction Algorithm

Input: Labeled source and unlabeled target data

1) Label source instances as $\sigma = 1$, target as $\sigma = 0$

2) Train predictor $\Pr[\sigma = 1 | x] = \frac{1}{1+\exp(\theta^\top x + b)}$

3) Reweight source instances



Selection Bias Correction Algorithm

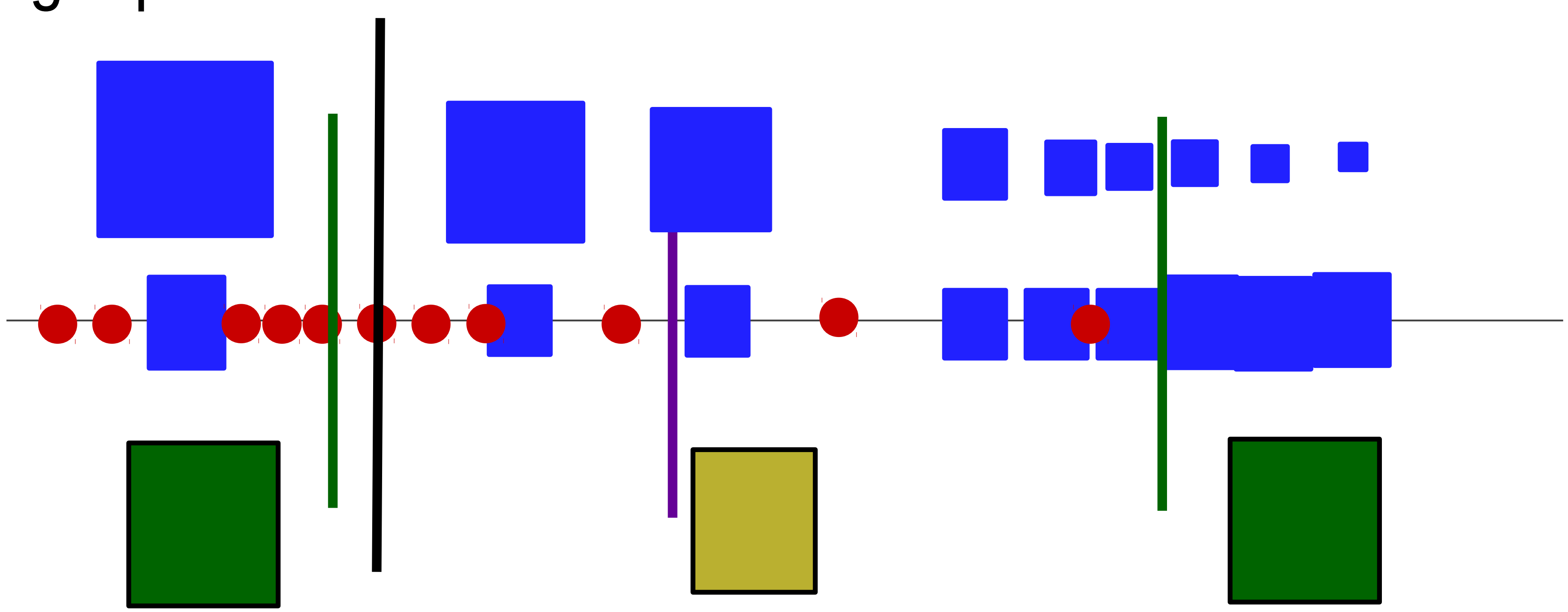
Input: Labeled source and unlabeled target data

1) Label source instances as $\sigma = 1$, target as $\sigma = 0$

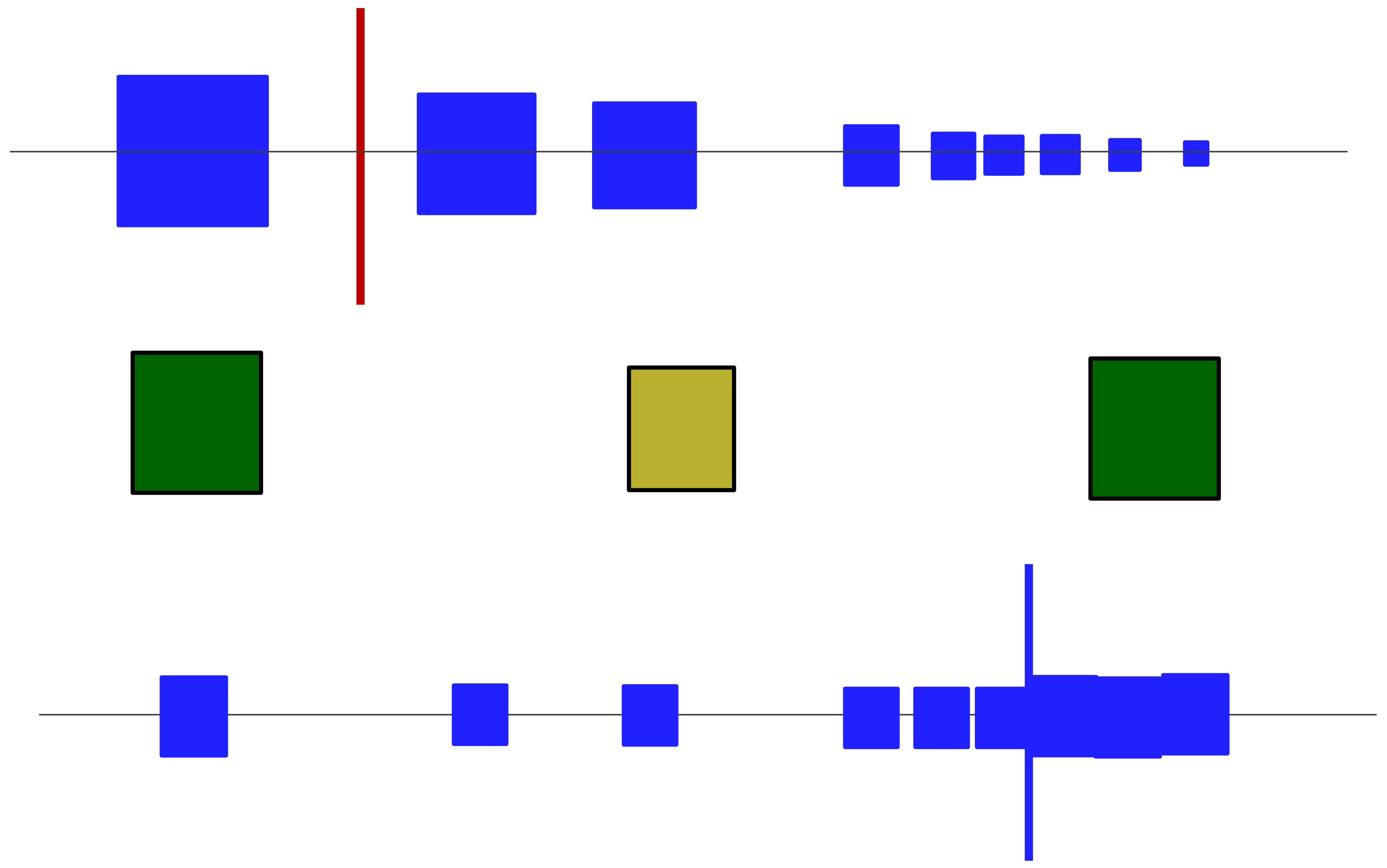
2) Train predictor $\Pr[\sigma = 1 | x] = \frac{1}{1+\exp(\theta^\top x + b)}$

3) Reweight source instances

4) Train target predictor



How Bias gets Corrected



Rates for Re-weighted Learning

$\hat{\epsilon}_s^n(h, w)$: weighted source test error on sample of size n

With probability $1 - \delta$, for every h

$$|\hat{\epsilon}_S^n(h, w) - \epsilon_T(h)| \leq \sqrt{\frac{O\left(\frac{1}{\delta}\right) + O\left(\max_{x \in \mathcal{X}} w(x)^2\right)}{n}}$$

Adapted from Gretton et al.

Sample Selection Bias Summary

Two Key Assumptions

1) Covariate shift: $\Pr_S[y|x] = \Pr_T[y|x]$

2) Shared support: $\forall x,\ \Pr_S[x] \neq 0$ iff $\Pr_T[x] \neq 0$

Advantage

$$\hat{\epsilon}_S^n(h, w) \xrightarrow[\infty]{n} \epsilon_T(h)$$

Optimal target predictor without labeled target data

Sample Selection Bias Summary

Two Key Assumptions

1) Covariate shift: $\Pr_S[y|x] = \Pr_T[y|x]$

2) Shared support: $\forall x,\ \Pr_S[x] \neq 0$ iff $\Pr_T[x] \neq 0$

Advantage

$$\hat{\epsilon}_S^n(h, w) \xrightarrow[\infty]{n} \epsilon_T(h)$$

Disadvantage

Convergence to $\epsilon_T(h)$ depends on $\max_x \frac{\Pr_T(x)}{\Pr_S(x)}$

Sample Selection Bias References

http://adaptationtutorial.blitzer.com/references/

[1] J. Heckman. Sample Selection Bias as a Specification Error. 1979.

[2] A. Gretton et al. Covariate Shift by Kernel Mean Matching. 2008.

[3] C. Cortes et al. Sample Selection Bias Correction Theory. 2008

[4] S. Bickel et al. Discriminative Learning Under Covariate Shift. 2009.

Unshared Support in the Real World

amazon.com

Running with Scissors

Title: Horrible book, horrible.

This book was horrible. I read half, suffering from a headache the entire time, and eventually i lit it on fire. 1 less copy in the world. Don't waste your money. I wish i had the time spent reading this book back. It wasted my life

Avante Deep Fryer; Black

Title: lid does not work well...

I love the way the Tefal deep fryer cooks, however, I am returning my second one due to a defective lid closure. The lid may close initially, but after a few uses it no longer stays closed. I won’t be buying this one again.

Unshared Support in the Real World

Running with Scissors

Title: Horrible book, horrible.

This book was horrible. I read half, suffering from a headache the entire

copy in the world. Don't waste your money. I wish i had the time spent reading this book back. It wasted my life

Avante Deep Fryer; Black

Title: lid **does not work** well...

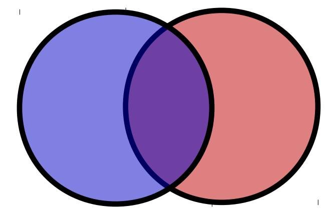
I love the way the Tefal deep fryer cooks, however, I am **returning** my

but after a few uses it no longer stays closed. I won't be buying this one again.

Error increase: 13% → 26%

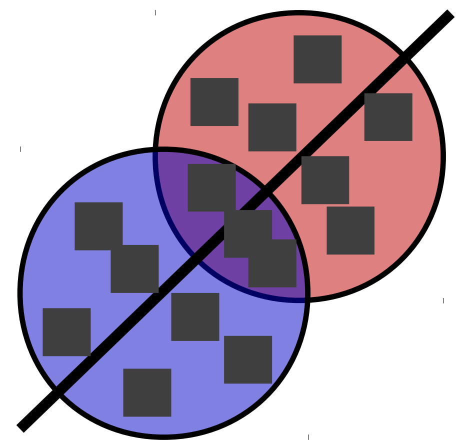
Coupled Subspaces

No Shared Support



Single Good Linear Hypothesis

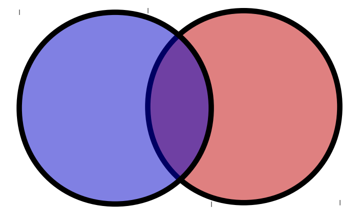
$$\exists \theta^*, \quad \epsilon_S(\theta^*) + \epsilon_T(\theta^*) \quad \text{small}$$



Stronger than $\Pr_S[y|x] = \Pr_T[y|x]$

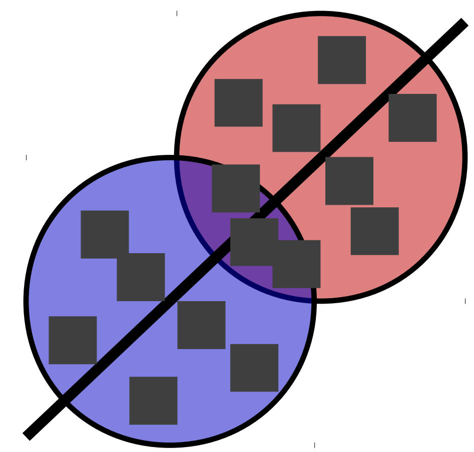
Coupled Subspaces

No Shared Support



Single Good Linear Hypothesis

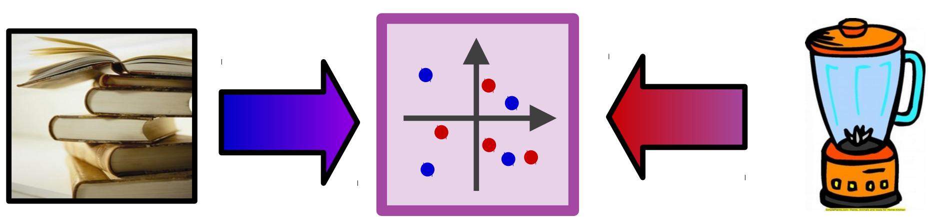
$$\exists \theta^*, \quad \epsilon_S(\theta^*) + \epsilon_T(\theta^*) \quad \text{small}$$



Coupled Representation Learning

Px couples domains

Bound target error $\epsilon_{P,T}(\theta)$



Single Good Linear Hypothesis?

$$\exists \theta^*, \quad \epsilon_S(\theta^*) + \epsilon_T(\theta^*) \quad \text{small}$$

Adaptation Squared Error

Source \ Target	Books	Kitchen
Books	**1.35**	
Kitchen		**1.19**
Both		

Single Good Linear Hypothesis?

$$\exists \theta^*, \quad \epsilon_S(\theta^*) + \epsilon_T(\theta^*) \quad \text{small}$$

Adaptation Squared Error

Source \ Target	Books	Kitchen
Books	**1.35**	
Kitchen		**1.19**
Both	**1.38**	**1.23**

Single Good Linear Hypothesis?

$$\exists \theta^*, \quad \epsilon_S(\theta^*) + \epsilon_T(\theta^*) \quad \text{small}$$

Adaptation Squared Error

Source \ Target	Books	Kitchen
Books	**1.35**	1.68
Kitchen	1.80	**1.19**
Both	**1.38**	**1.23**

A bound on the adaptation error

Let h be a binary hypothesis. If $\Pr_S(Y|x) = \Pr_T(Y|x)$, then

$$\epsilon_T(h) \quad \leq \quad \epsilon_S(h) + \int_{\mathcal{X}} |\Pr_T(x) - \Pr_S(x)| \, dx$$

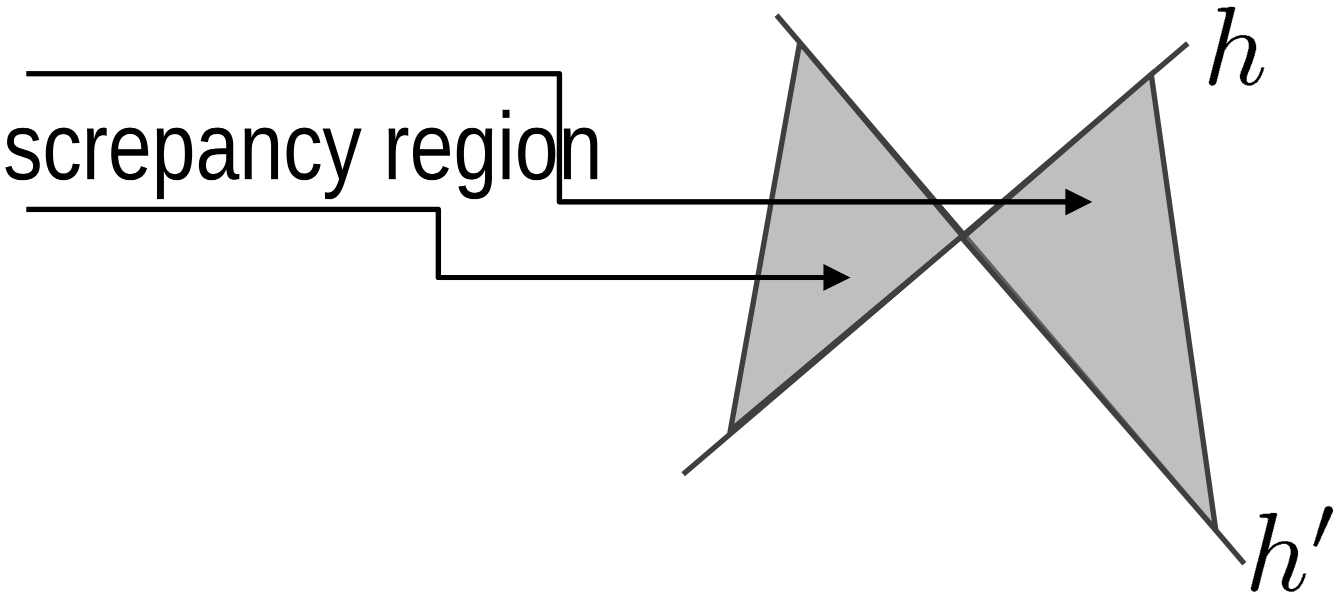
What if a single good hypothesis exists?

A better discrepancy than total variation?

A generalized discrepancy distance

Measure how hypotheses make mistakes

Linear, binary discrepancy region

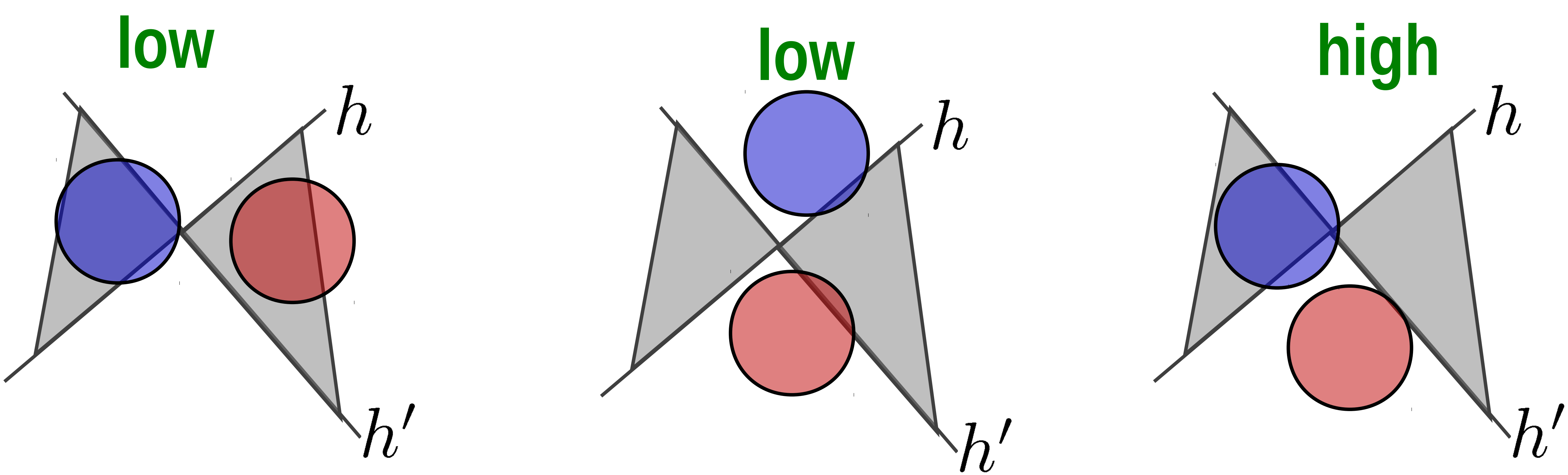


A generalized discrepancy distance

Measure how hypotheses make mistakes

$$\mathrm{disc}_H(Q,P) =$$

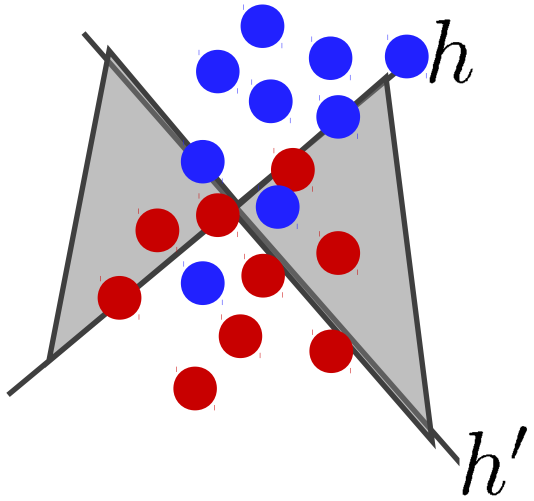
$$\max_{h,h' \in H} \left| E_Q[h(x) \neq h'(x)] - E_P[h(x) \neq h'(x)] \right|$$



Discrepancy vs. Total Variation

Discrepancy

Computable from finite samples.



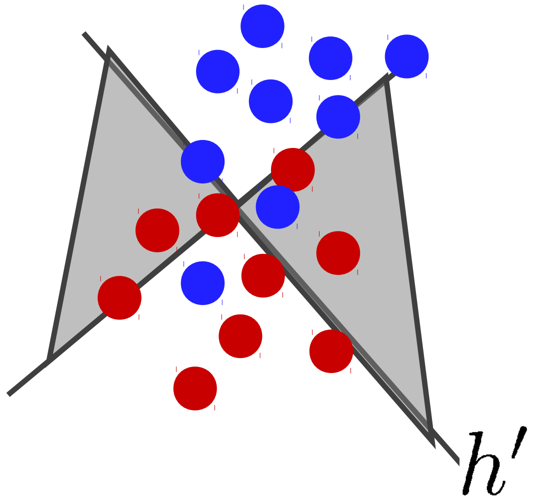
Total Variation

Not computable in general

Discrepancy vs. Total Variation

Discrepancy

Computable from finite samples.



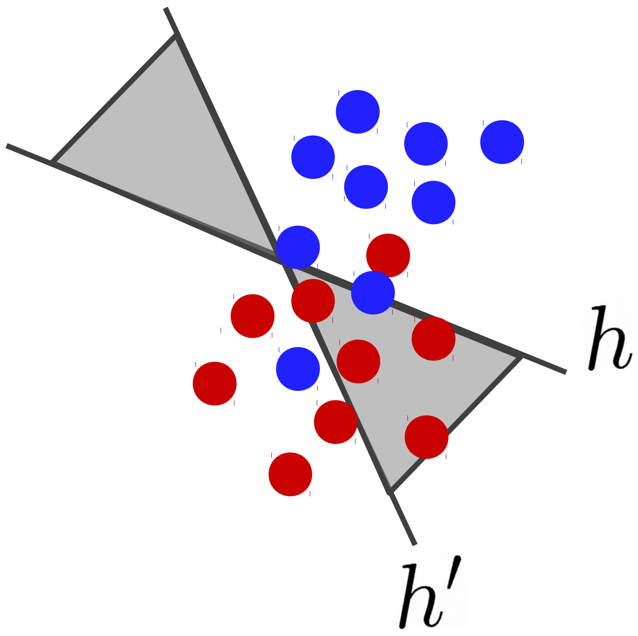
Total Variation

Not computable in general

Discrepancy vs. Total Variation

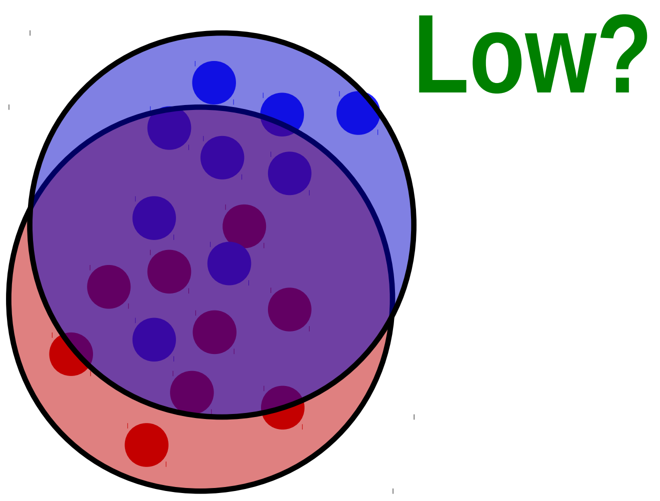
Discrepancy

Computable from finite samples.



Total Variation

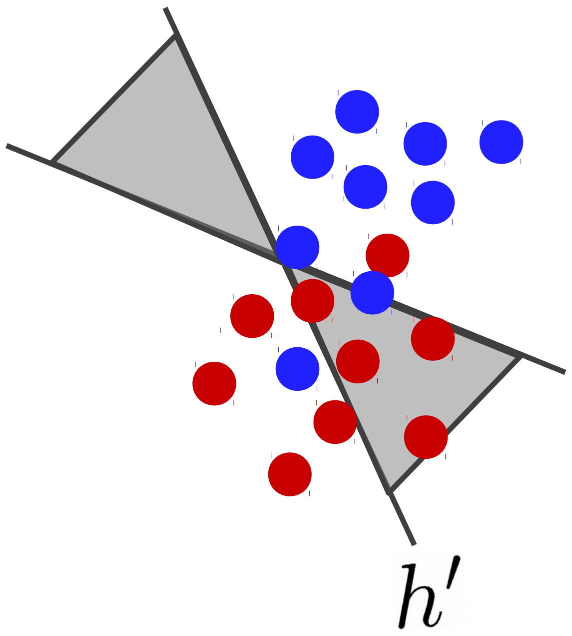
Not computable in general



Discrepancy vs. Total Variation

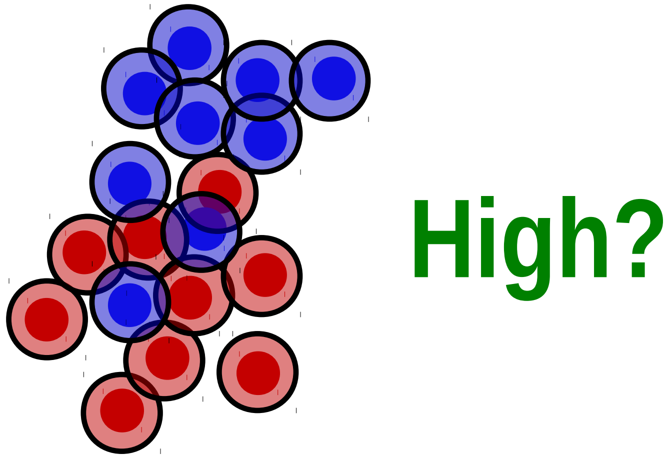
Discrepancy

Computable from finite samples.



Total Variation

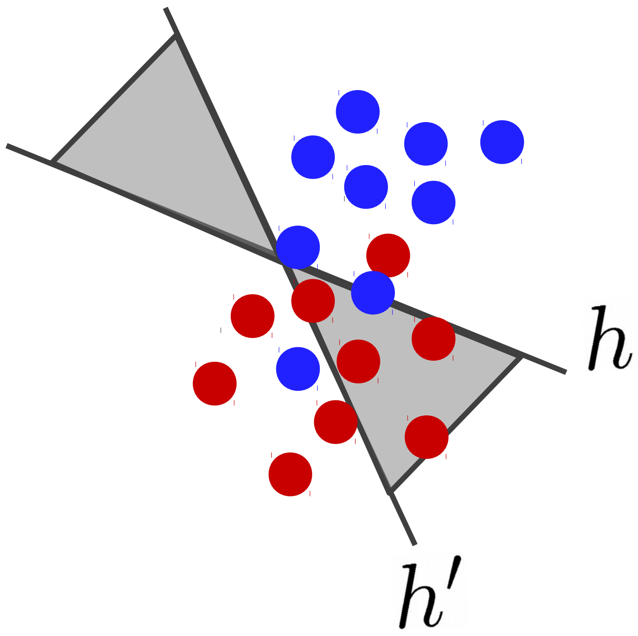
Not computable in general



Discrepancy vs. Total Variation

Discrepancy

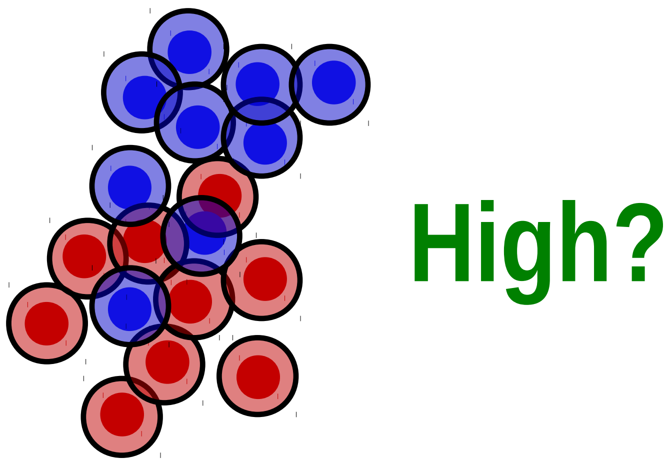
Computable from finite samples.



Related to hypothesis class

Total Variation

Not computable in general



Unrelated to hypothesis class

Bickel covariate shift algorithm heuristically minimizes both measures

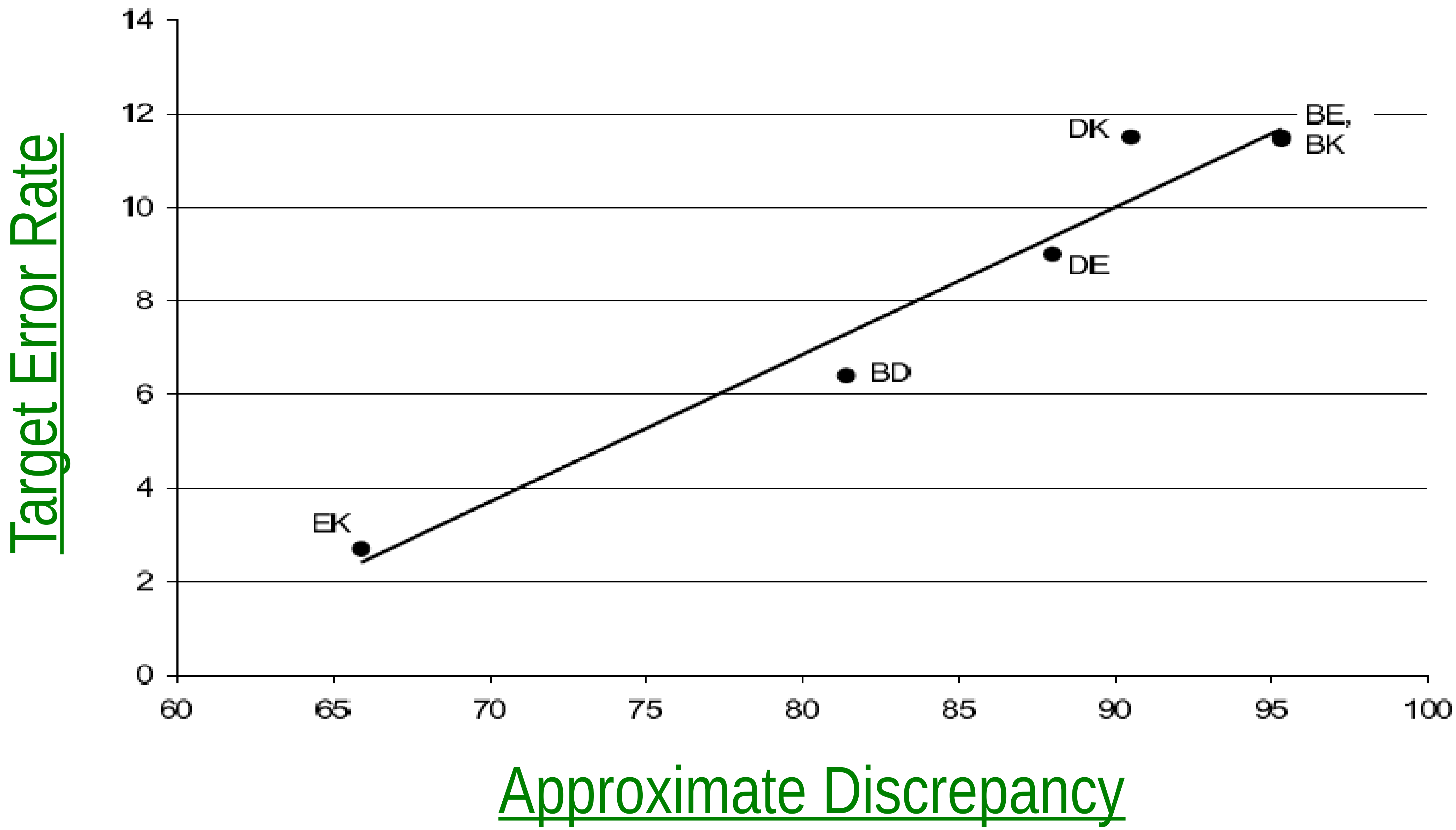
Is Discrepancy Intuitively Correct?

4 domains: **B**ooks, **D**VDs, **E**lectronics, **K**itchen

B&D, E&K Shared Vocabulary

B&D: *fascinating, boring*

E&K: *super easy, bad quality*



An adaptation bound

S, T: Source and target $\quad \mathcal{H}$: Hypothesis class $\quad n$: Sample size

$\hat{S}$: Labeled S sample $\quad \hat{T}$: Unlabeled T sample

$\mathcal{R}_{\hat{S}}(\mathcal{H})$, $\mathcal{R}_{\hat{T}}(\mathcal{H})$: Rademacher complexities

With probability $1-\delta$, for h the ERM of $\hat{S}$:

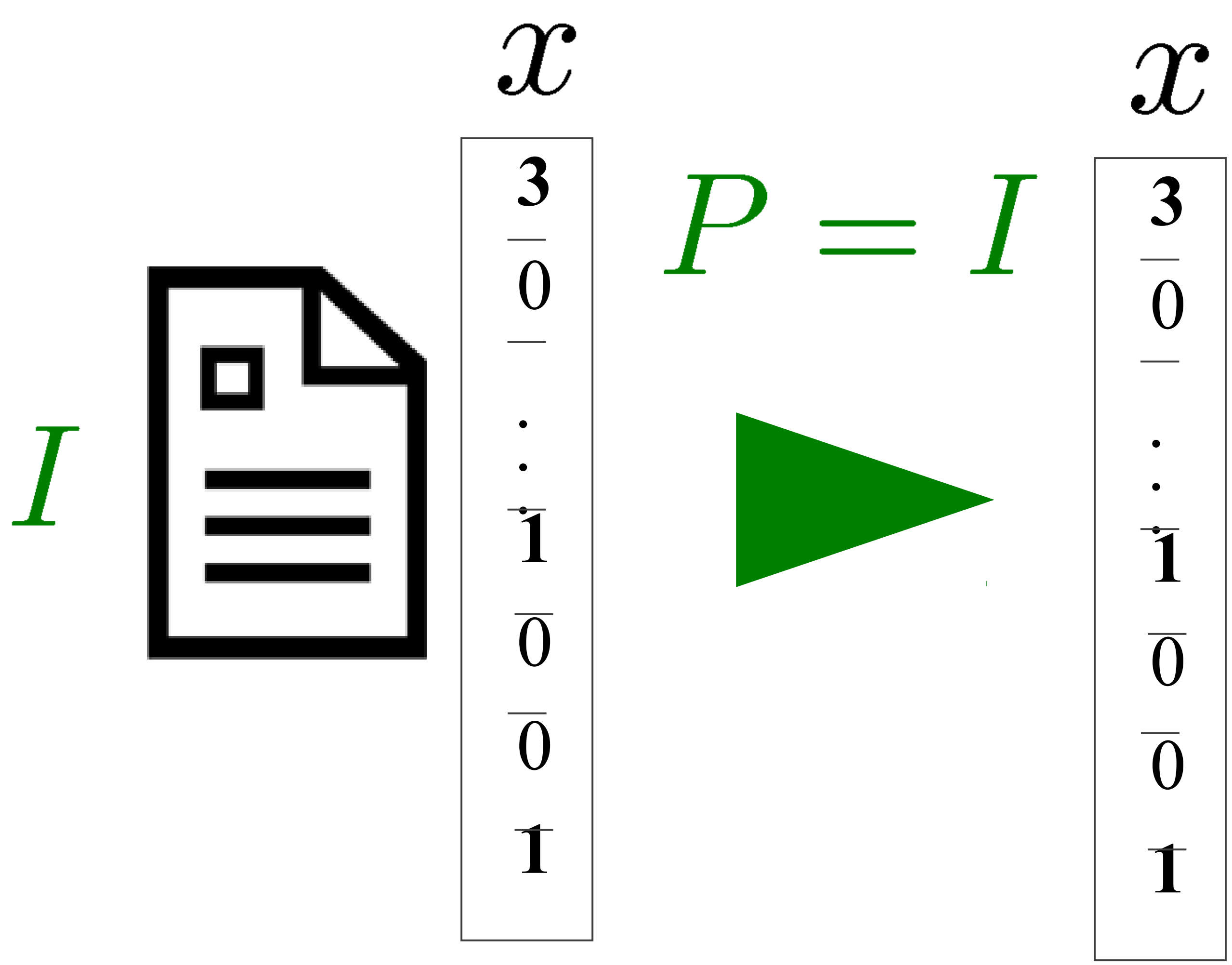
$$\epsilon_T(h) - \epsilon_T(h^*) \quad \leq \quad \epsilon_{\hat{S}}(h, h^*) + O\left(\mathcal{R}_{\hat{S}}(\mathcal{H}) + \mathcal{R}_{\hat{T}}(\mathcal{H})\right)$$

$$+ O\left(\sqrt{\frac{\log \frac{1}{\delta}}{n}}\right) + \mathrm{disc}_{\mathcal{H}}(\hat{S}, \hat{T})$$

Representations and the Bound

Linear Hypothesis Class: $h(x) = \mathrm{sgn}\left(\theta^\top x\right)$

Hypothesis classes from projections P: $\theta^\top P x$

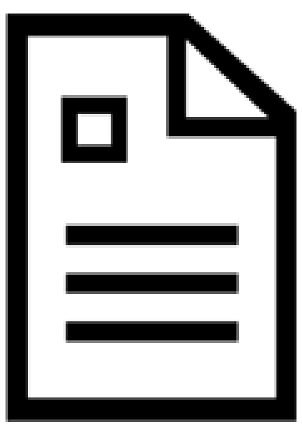


Representations and the Bound

Linear Hypothesis Class: $h(x) = \operatorname{sgn}\left(\theta^\top x\right)$

Hypothesis classes from projections : P $\theta^\top P x$

P

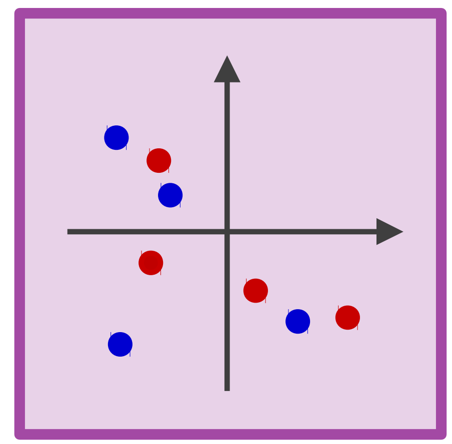


x

$$\begin{bmatrix} 3 \\ 0 \\ \vdots \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

P

$\mathbb{R}^d$

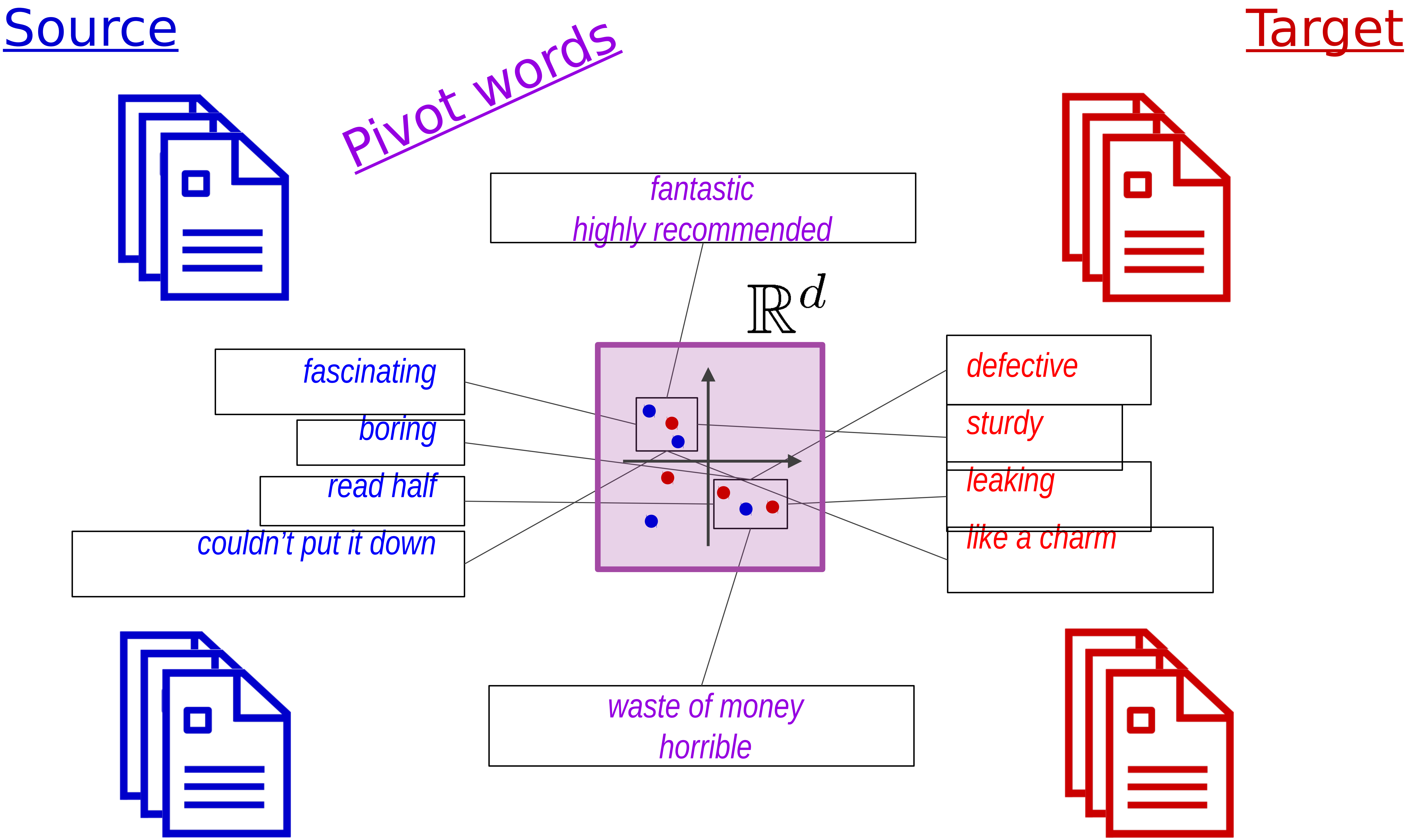


Goals for P

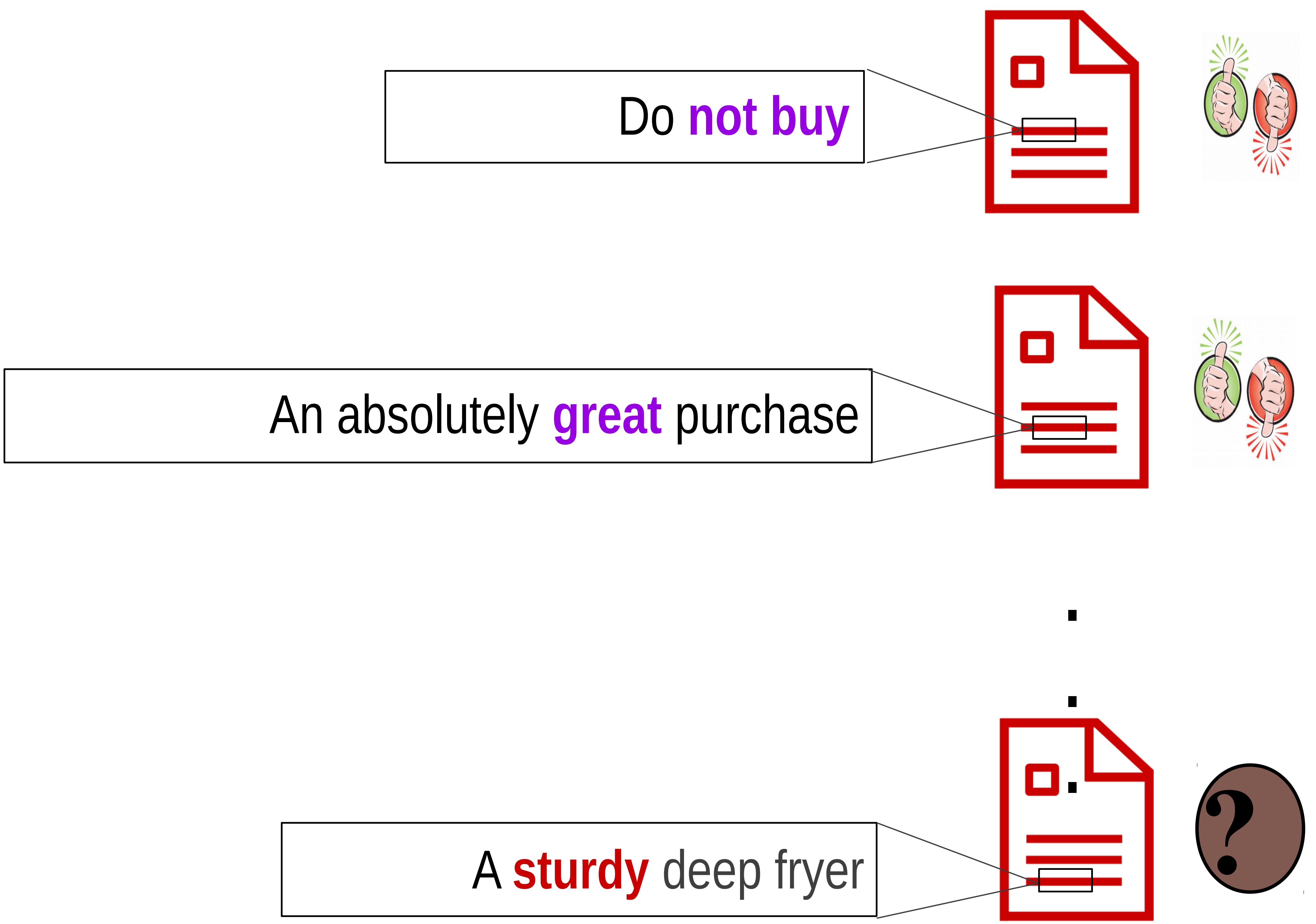
1) Minimize divergence

2) $\epsilon_{P,T}(\theta^*) - \epsilon_{I,T}(\theta^*)$ small

Learning Representations: Pivots

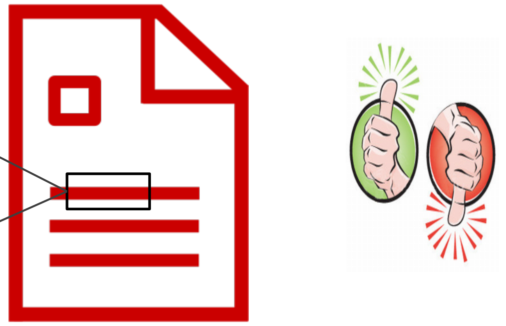


Predicting pivot word presence

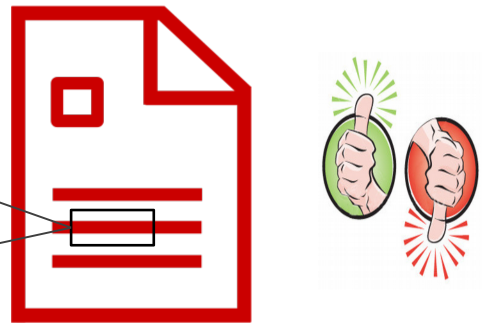


Predicting pivot word presence

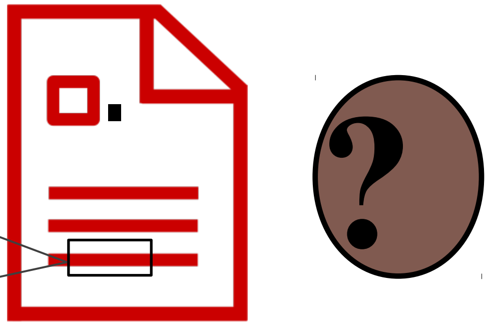
Do **not buy** the Shark portable steamer. The trigger mechanism is **defective**.



An absolutely **great** purchase

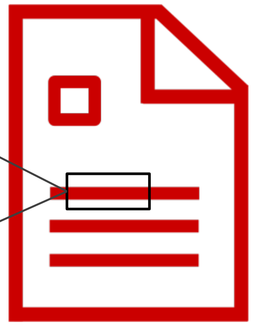


A **sturdy** deep fryer

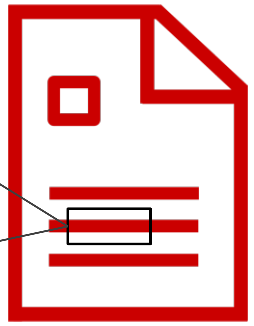


Predicting pivot word presence

Do **not buy** the Shark portable steamer. The trigger mechanism is **defective**.



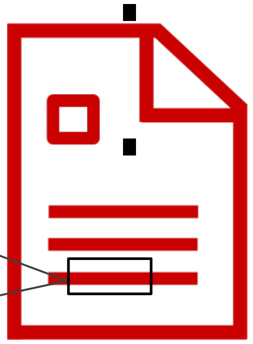
An absolutely **great** purchase. . . . This blender is incredibly **sturdy**.



Predict presence of pivot words

$$p_{w(\text{great})}(\text{great} \mid x) \propto \exp\{\langle x, w(\text{great})\rangle\}$$

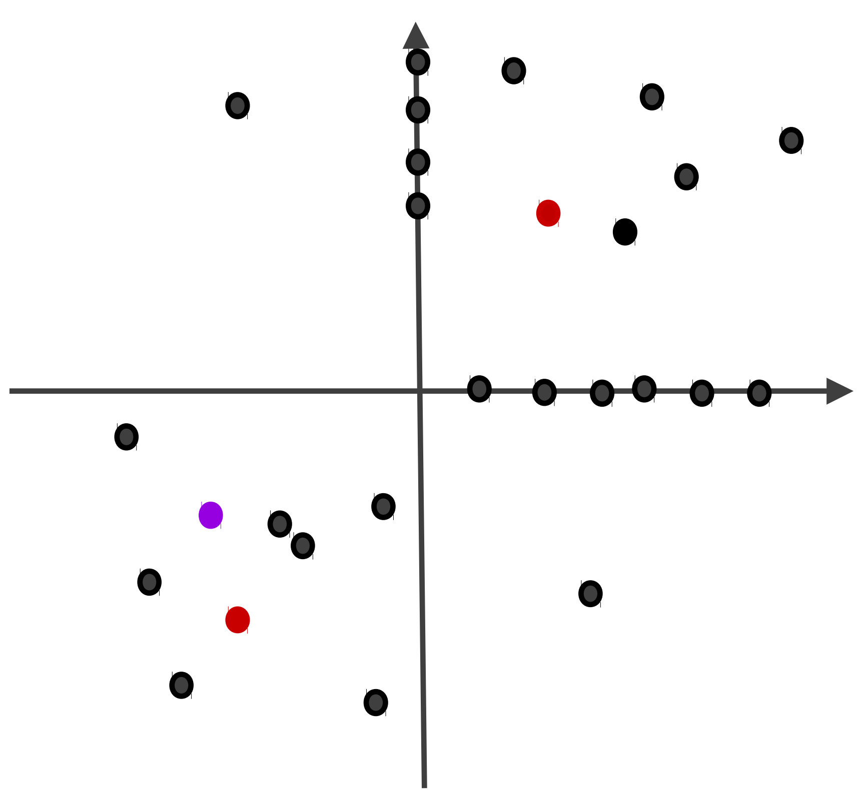
A **sturdy** deep fryer



Finding a shared sentiment subspace

$$W = \begin{bmatrix} | & & | & & | \\ w_1 & \dots & w(\textit{highly recommend}) & \dots & w_N \\ | & & | & & | \end{bmatrix}$$

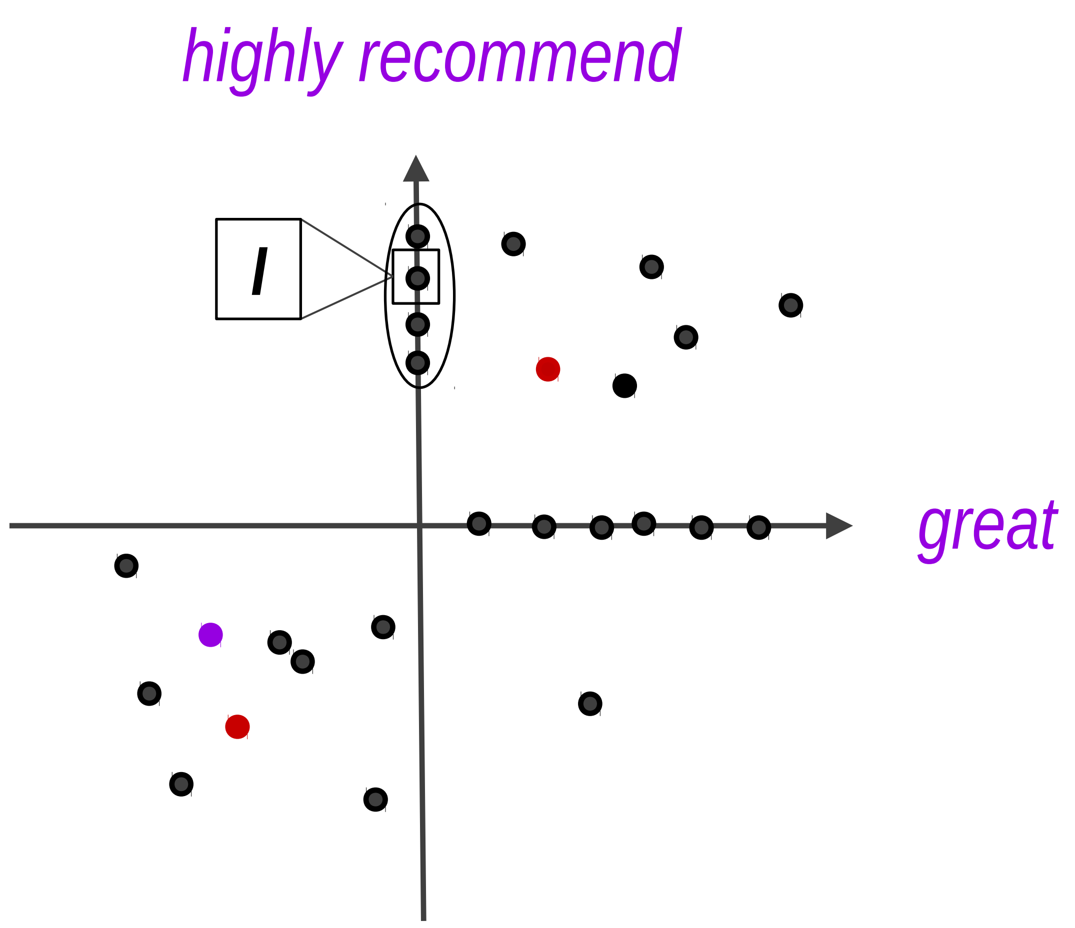
- $p_W(\textit{pivots}|x)$ generates N new features
- $p_{w(\text{highly recommend})}(\text{highly recommend}|x)$: "Did *highly recommend* appear?"
- Sometimes predictors capture non-sentiment information



Finding a shared sentiment subspace

$$W = \begin{bmatrix} | & & | & & | \\ w_1 & \dots & w(\text{highly recommend}) & \dots & w_N \\ | & & | & & | \end{bmatrix}$$

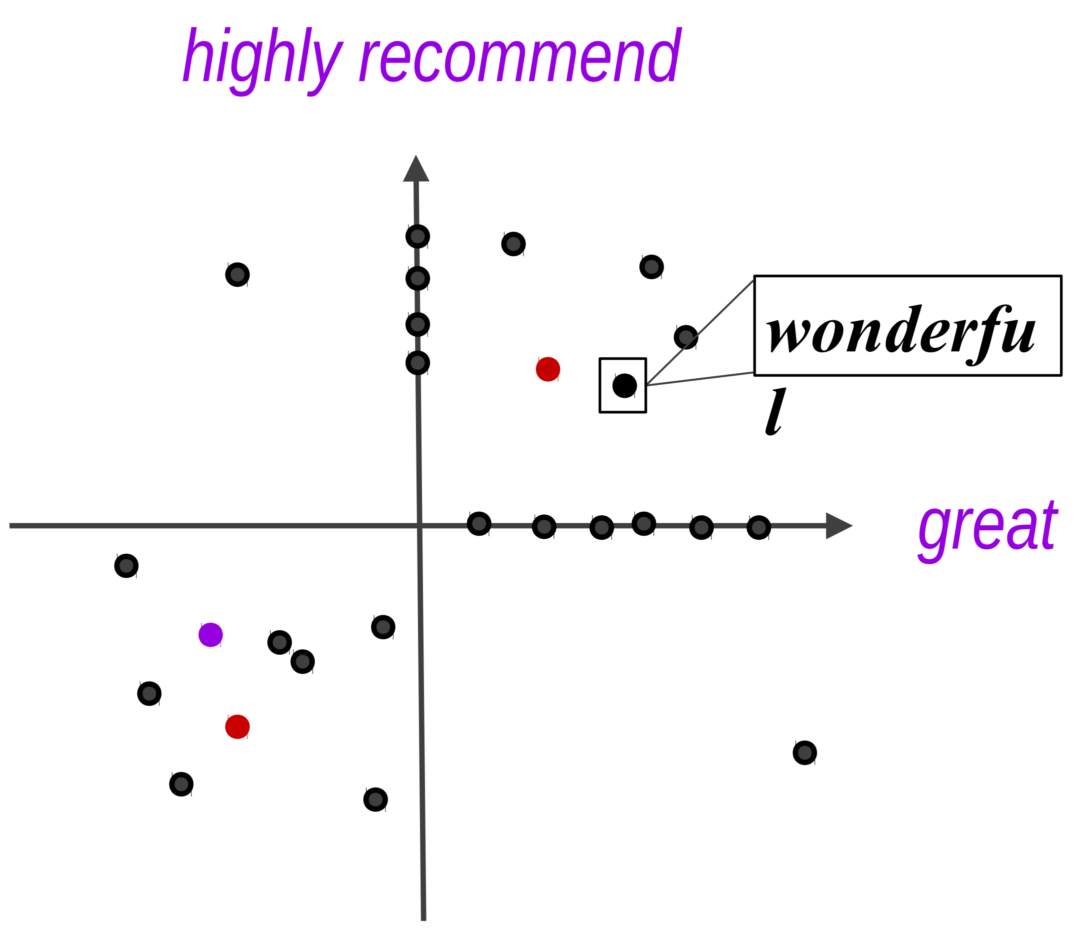
- $p_W(\text{pivots} \mid x)$ generates N new features
- $p_{w(\text{highly recommend})}(\text{highly recommend} \mid x)$: "Did *highly recommend* appear?"
- Sometimes predictors capture non-sentiment information



Finding a shared sentiment subspace

$$W = \begin{bmatrix} | & & | & & | \\ w_1 & \dots & w(\textit{highly recommend}) & \dots & w_N \\ | & & | & & | \end{bmatrix}$$

- $p_W(\textit{pivots} \mid x)$ generates N new features
- $p_{w(\text{highly recommend})}(\text{highly recommend} \mid x)$: "Did *highly recommend* appear?"
- Sometimes predictors capture non-sentiment information

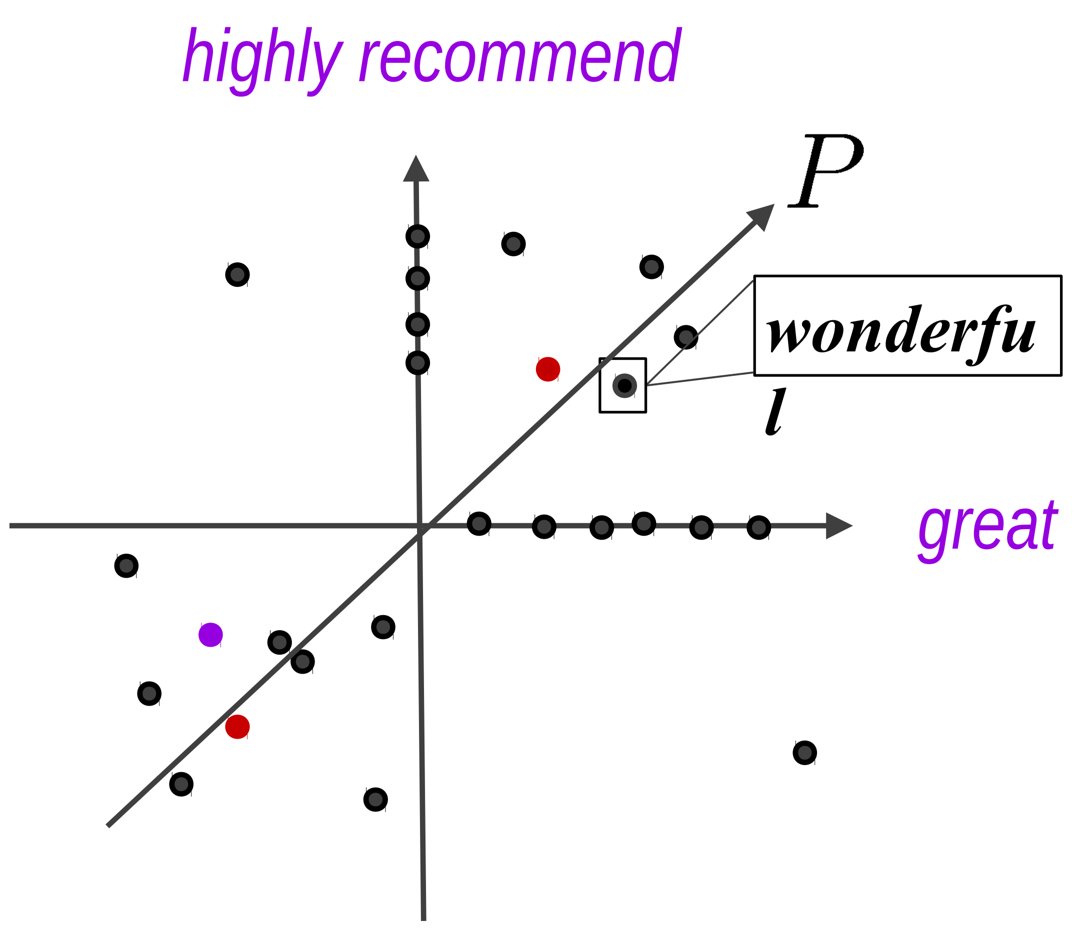


Finding a shared sentiment subspace

$$W = \begin{bmatrix} | & & | & & | \\ w_1 & \dots & w(\textit{highly recommend}) & \dots & w_N \\ | & & | & & | \end{bmatrix}$$

- Let P be a basis for the subspace of best fit to W

- $p_W(\textit{pivots} \mid x)$ generates N new features
- $p_{w(\text{highly recommend})}(\text{highly recommend} \mid x)$: "Did *highly recommend* appear?"
- Sometimes predictors capture non-sentiment information

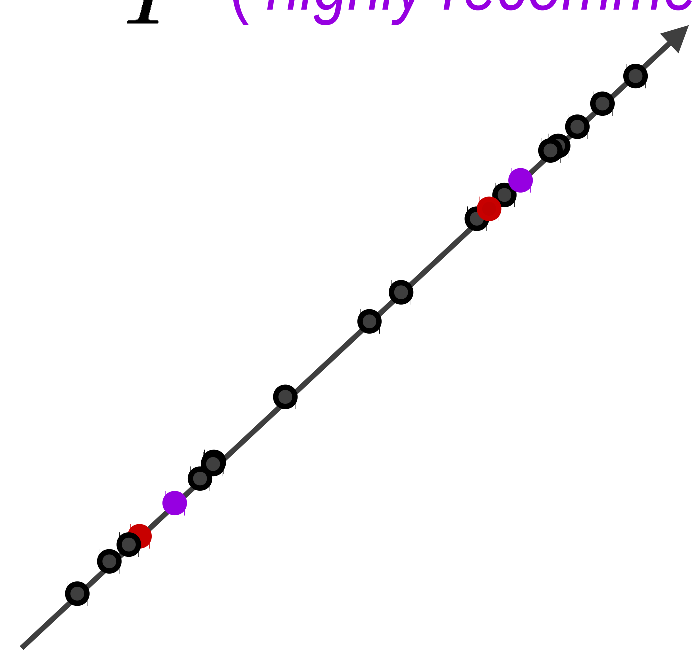


Finding a shared sentiment subspace

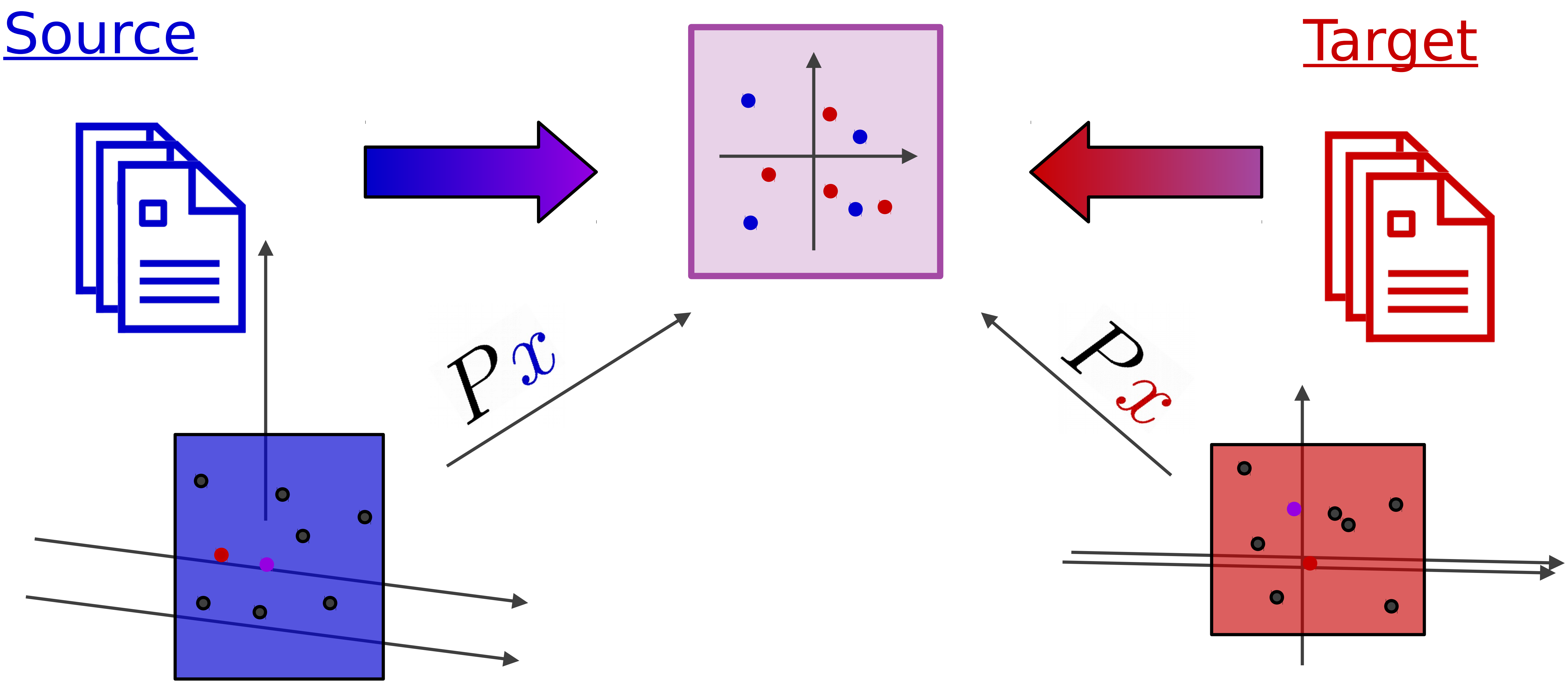
$$W = \begin{bmatrix} | & & | & & | \\ w_1 & \dots & w(\textit{highly recommend}) & \dots & w_N \\ | & & | & & | \end{bmatrix}$$

- Let be a basis for the subspace of best fit to W
- P captures sentiment variance in

- $p_W(\textit{pivots}|x)$ generates N new features
- $p_{w(\textit{highly recommend})}(\textit{highly recommend}|x)$: "Did *highly recommend* appear?"
- Sometimes predictors capture non-sentiment information

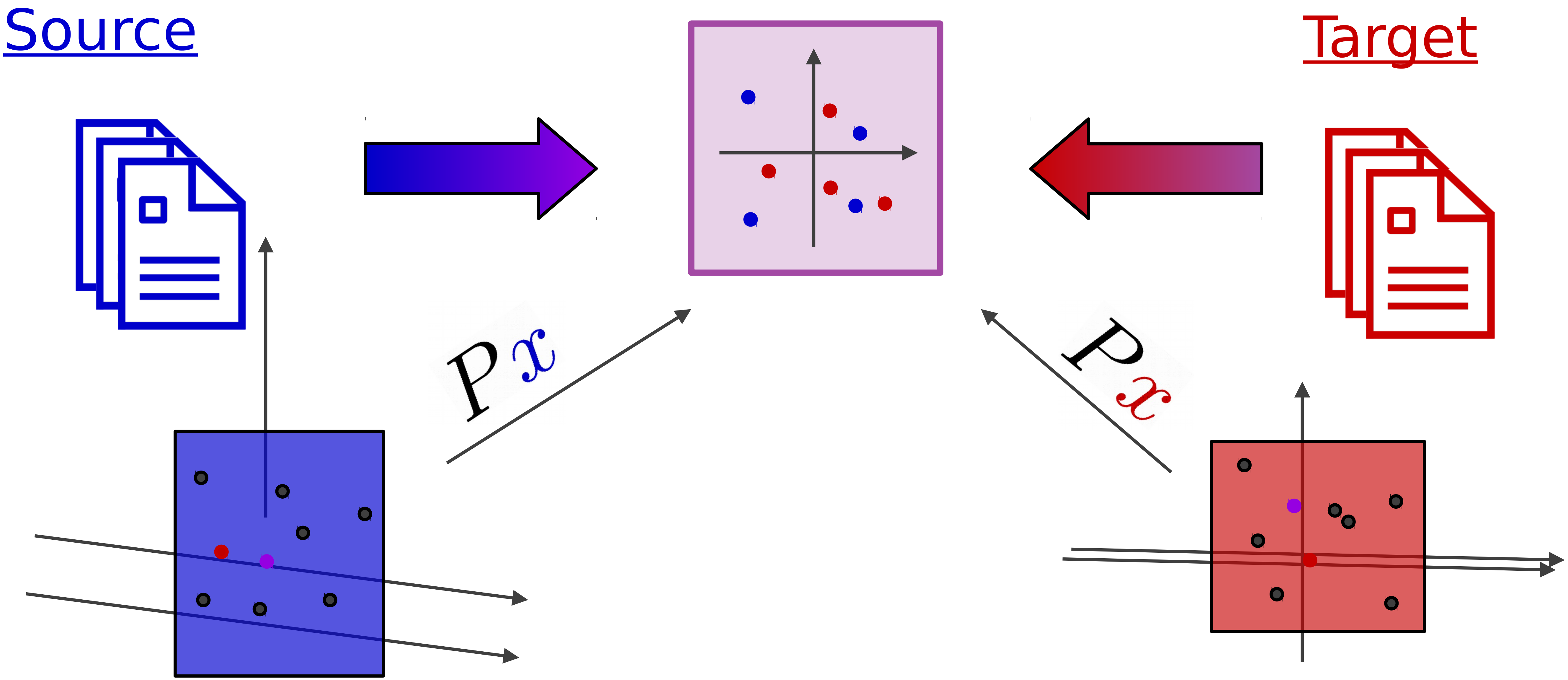


P projects onto shared subspace



$p_{\tilde{\theta}}($  $|x) \propto \exp\left\{ \langle \phi($  $, Px), \tilde{\theta} \rangle \right\}$

P projects onto shared subspace



$$h(x) = \text{sgn}\left(\theta^\top P x\right)$$

Correlating Pieces of the Bound

$$\epsilon_T(h) - \epsilon_T(h^*) \quad \leq \quad \epsilon_{\hat{S}}(h, h^*) + O\left(\mathcal{R}_{\hat{S}}(\mathcal{H}) + \mathcal{R}_{\hat{T}}(\mathcal{H})\right)$$

$$+O\left(\sqrt{\frac{\log \frac{1}{\delta}}{n}}\right) + \mathrm{disc}_{\mathcal{H}}(\hat{S}, \hat{T})$$

Component Projection	Discrepancy	Source Huber Loss	Target Error
Idenitity	1.796	**0.003**	0.253

Correlating Pieces of the Bound

$$\epsilon_T(h) - \epsilon_T(h^*) \quad \leq \quad \epsilon_{\hat{S}}(h, h^*) + O\left(\mathcal{R}_{\hat{S}}(\mathcal{H}) + \mathcal{R}_{\hat{T}}(\mathcal{H})\right)$$

$$+O\left(\sqrt{\frac{\log \frac{1}{\delta}}{n}}\right) + \mathrm{disc}_{\mathcal{H}}(\hat{S}, \hat{T})$$

Component Projection	Discrepancy	Source Huber Loss	Target Error
Idenitity	1.796	**0.003**	0.253
Random	0.223	0.254	0.561

Correlating Pieces of the Bound

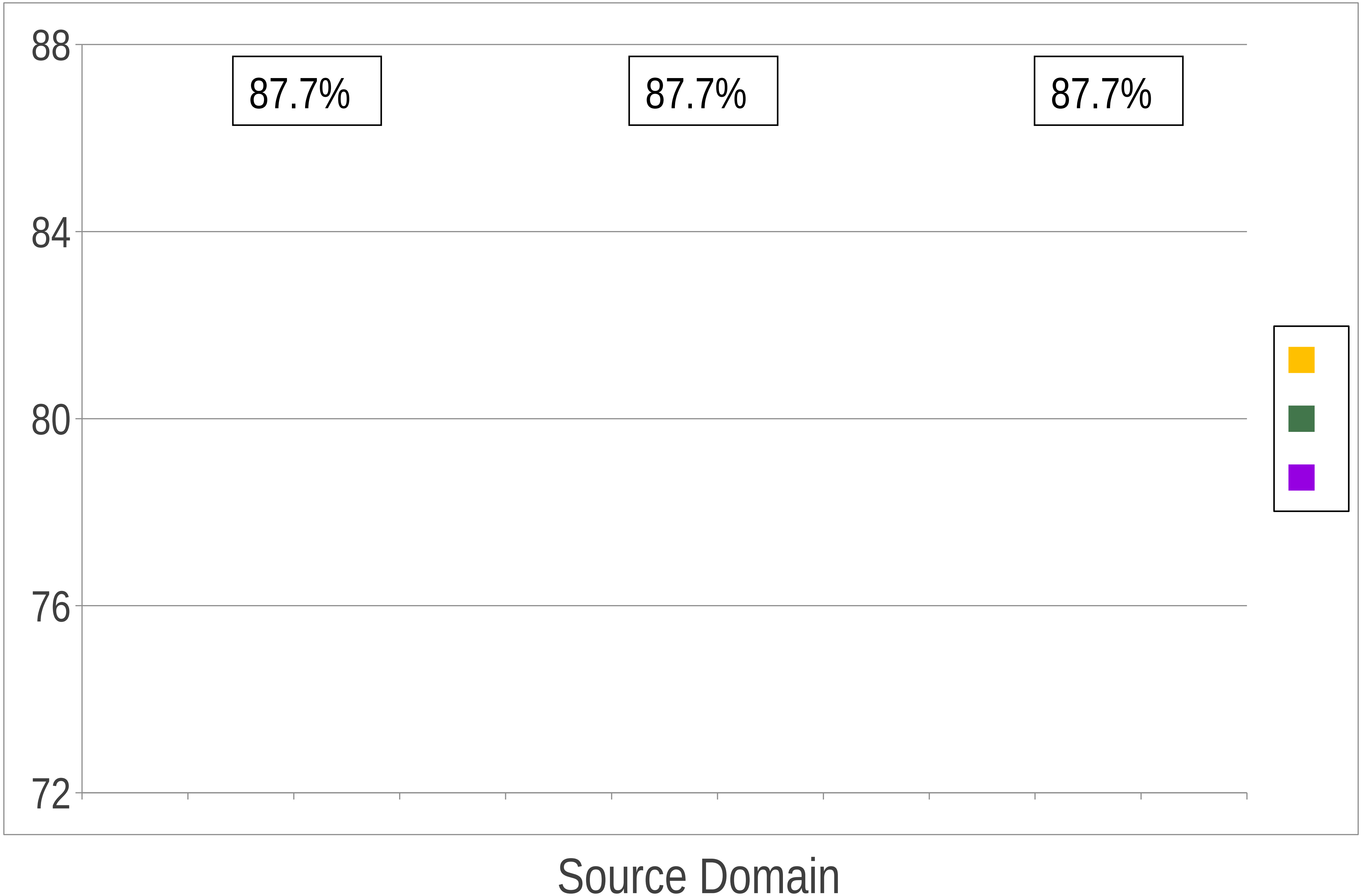
$$\epsilon_T(h) - \epsilon_T(h^*) \quad \leq \quad \epsilon_{\hat{S}}(h, h^*) + O\left(\mathcal{R}_{\hat{S}}(\mathcal{H}) + \mathcal{R}_{\hat{T}}(\mathcal{H})\right)$$

$$+O\left(\sqrt{\frac{\log \frac{1}{\delta}}{n}}\right) + \mathrm{disc}_{\mathcal{H}}(\hat{S}, \hat{T})$$

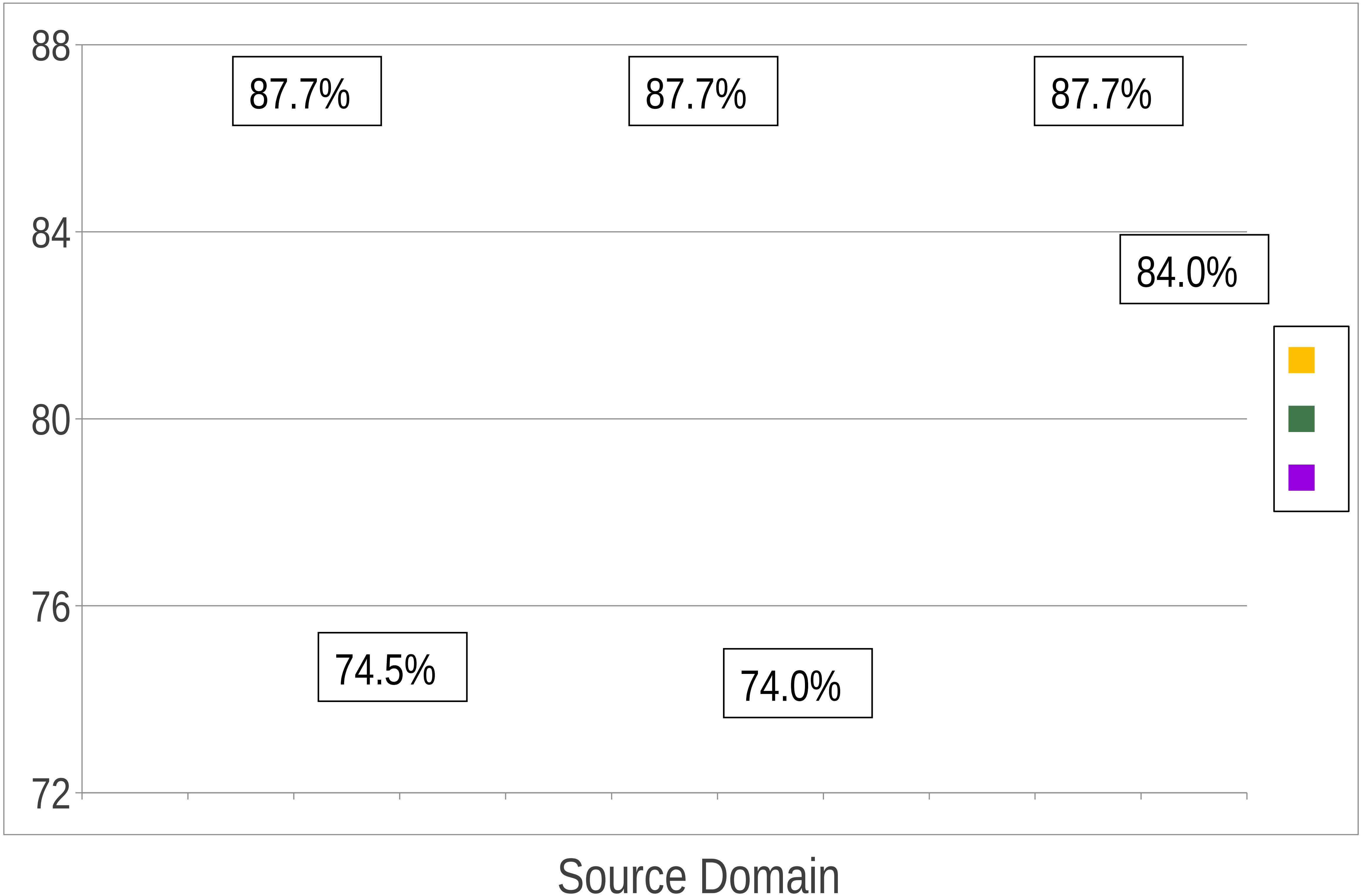
Component Projection	Discrepancy	Source Huber Loss	Target Error
Idenitity	1.796	**0.003**	0.253
Random	0.223	0.254	0.561
Coupled Projection	**0.211**	0.07	**0.216**

Target Accuracy: Kitchen Appliances

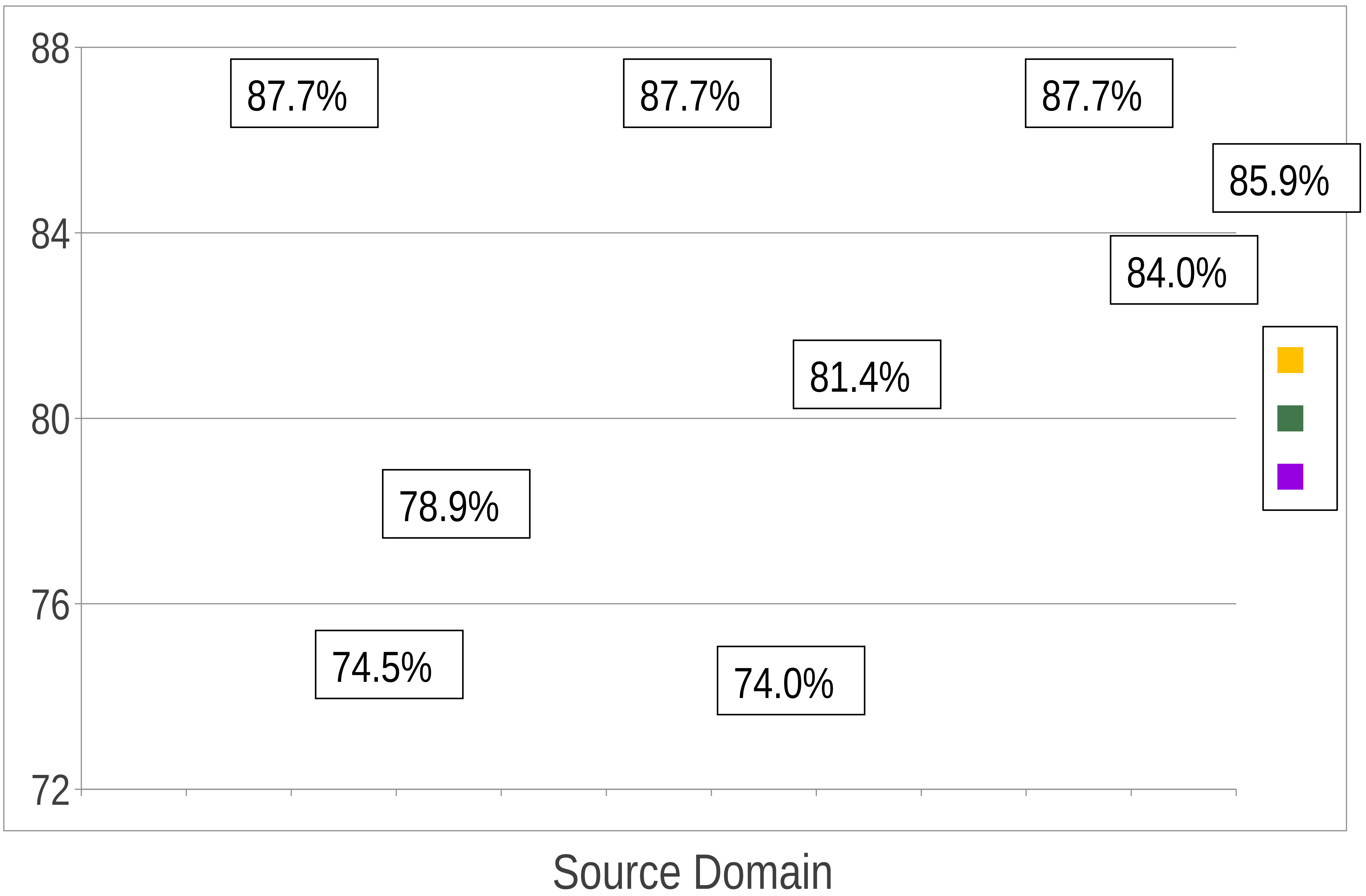
Target Accuracy: Kitchen Appliances



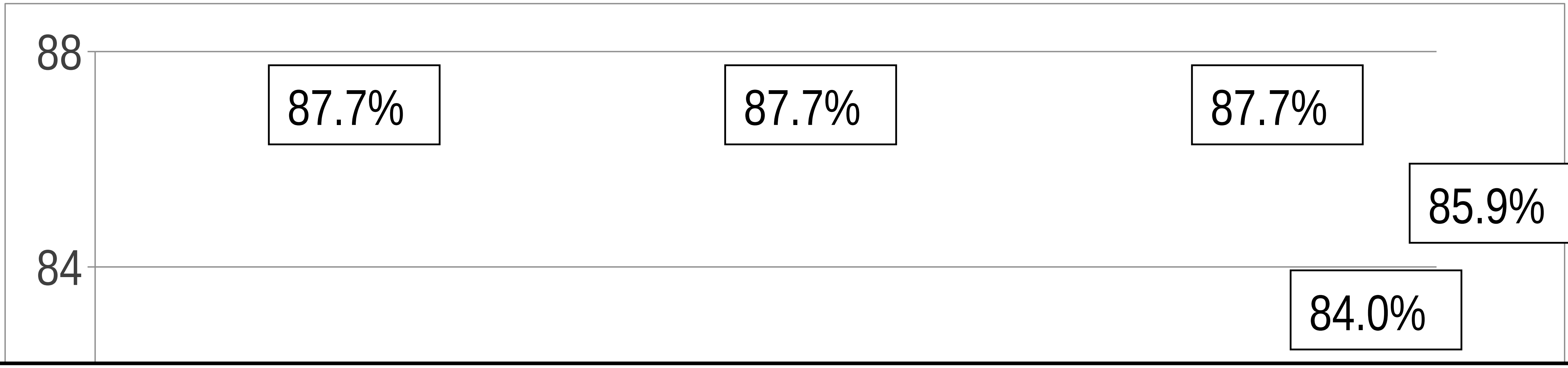
Target Accuracy: Kitchen Appliances



Target Accuracy: Kitchen Appliances



Adaptation Error Reduction



<u>36% reduction in error due to adaptation</u>

Representation References

http://adaptationtutorial.blitzer.com/references/

[1] Blitzer et al. Domain Adaptation with Structural Correspondence Learning. 2006.

[2] S. Ben-David et al. Analysis of Representations for Domain Adaptation. 2007.

[3] J. Blitzer et al. Domain Adaptation for Sentiment Classification. 2008.

[4] Y. Mansour et al. Domain Adaptation: Learning Bounds and Algorithms. 2009.

Today’s summary

- Quantifying what can and cannot be learned
  - No free lunch
  - VC dimension
- What are our core assumptions / how to break them
- How to unbreak (some of) them
  - Sample selection bias
  - Covariate shift

Your homework

- Find an example in the news of a machine learning system that potentially suffers from sample selection bias, or some other related bias

- Bonus points if it's not US-centric! :)

- How would you break the presented sample-selection-bias correction approach?

- Still time to fill out `go.umd.edu/mlvote`