Classification under Data Contamination with Applications

Donghui Yan

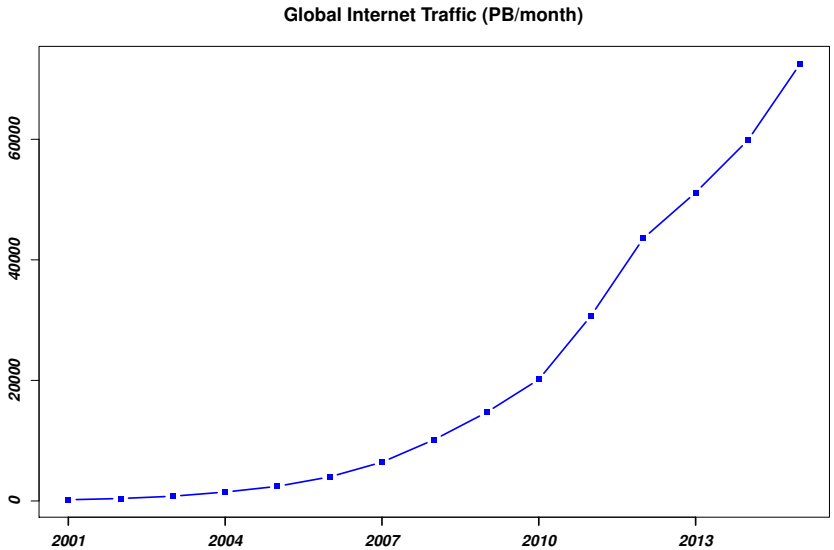
September 12, 2016

Outline

- Introduction
- Classification under data contamination
- Further applications
  - Item recommendations in online shopping
  - A new perspective in understanding Co-training
- Summary

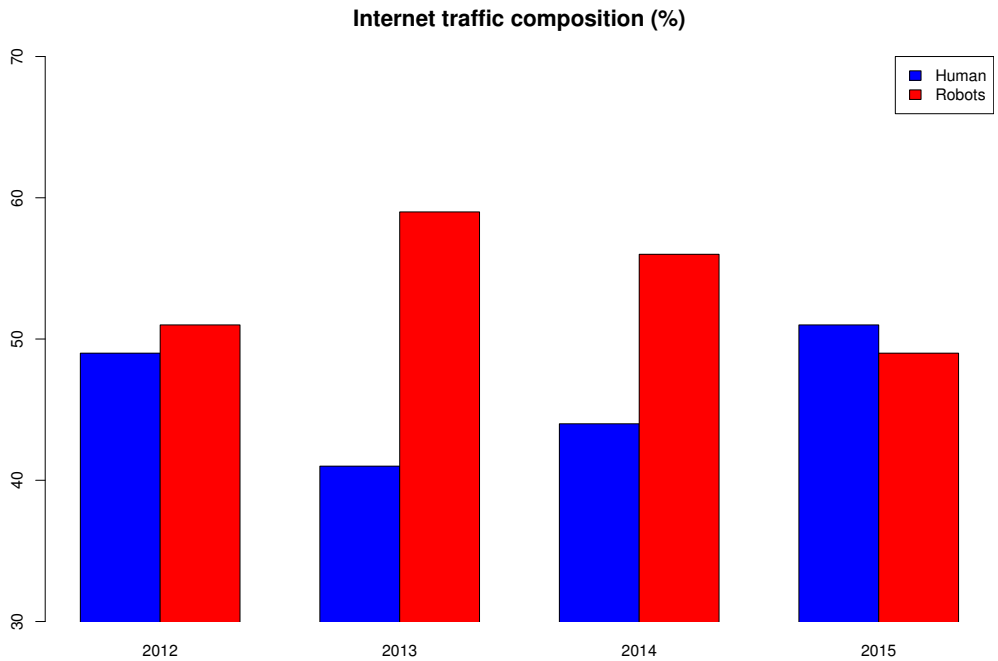
Exponential growth of the Internet

- Total # websites reached 1 billion
  - *NetCraft Web Server Survey*, as of Sep 2014
- Storage capacity at 10^{24} bytes
  - *Supercomputing Frontiers and Innovations*, 2014
- Amount of traffic *72521 PB/month* by 2015
  - Doubles almost every year since 2001.



A surprising fact about the Internet

- About half of the Internet traffic are from web robots (Bots)
  - A web robot is a program or scripts that can automatically download the content of a web site
    - Often in large amount and short time
  - Desirable to get rid of those unwanted Bots traffic.



Detection of web robots traffic

- A humble start for web related security initiative
- Focus on the detection of downloading web robots
  - Useful for large banks or commercial web sites
- Rely on server side web logs.

Example web log entries

```
"ip","utc","request","status","bytes","referrer","agent",
"time","tt"

"222.36.0.12",1138977224,"GET /cm/cs/cbook/ HTTP/1.1", 200, 8046
"http://www.google.com/search?hl=en\&lr=\&q=the+c+
programming+language\&btnG=Search","Mozilla/5.0 (Windows; U; Win
NT 5.1; en-US; rv:1.8.0.1) Gecko/20060111 Firefox/1.5.0.1",
"03/Feb/2006:14:33:44"


"222.36.0.12",1138977225,"GET /cm/cs/cbook/covers/main.gif
HTTP/1.1",200,14312,"http://cm.bell-labs.com/cm/cs/cbook/",
"Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.0.1)
Gecko/20060111 Firefox/1.5.0.1","03/Feb/2006:14:33:45"
```

Detection of web robots traffic

- Observation: an individual log entry does not reveal much
  - Have to rely on contextual information
  - Combining log entries as a session
- Any rules to characterize web robots traffic?
  - Session duration? # content requests? etc
  - None satisfactory due to too many exceptions
- Have to consider many features together
- Formulate the problem as a classification problem
  - Estimate how those features together tell 'robot' or 'non-robot'
  - Model fitting with a set of training examples.

The classification problem

Given a training set, $(X_1, Y_1), ..., (X_n, Y_n)$, a classifier learns a decision rule, i.e., a function $f : \mathcal{X} \mapsto \mathcal{Y} = \{1, ..., J\}$, so that a prediction can be made to a future example X. Here

- $X_i \in \mathcal{X}$ are called features and $Y_i \in \mathcal{Y}$ class labels
- (X, Y) from an unknown distribution $\mathbb{P}$, and
- The goal is to learn a function f that predicts well.

- $X_i =$ Features vector, $Y_i \in \{1, 2, ..., J\}$
- Allowing Y to be continuous $\Longrightarrow$ regression problem.

Preprocessed and labeled web log

```
"ip","utc","duration","nreq","avg.iArr","sd.iArr",
"max.iArr","min.iArr","err.rate","img.rate","unique.rate",
"p.popular","p.reload","p.unpopular","isRobot"

"31.156.55.89",1140783017,145,95,1.53,7.12,63,0,0,0.96,
0.99,0,0,0.75,TRUE

"79.129.215.12",1140176495,755,346,2.18,1.64,23,0,0.05,
0.01,0.96,0.06,0.02,0.90,TRUE

"155.58.226.12",1139888213,46,19,2.42,5.46,20,0,0,0.74,
1,0.4,0,0.4,TRUE

"171.185.55.89",1139844867,27,7,3.86,3.67,10,0,0.14,0.14,
1,0.17,0,0.67,FALSE

"144.71.198.12",1139859646,8,5,1.6,2.83,6,0,0.2,0.6,1, 0.5,0,0,F
```

Preprocessed and labeled web log (continued)

```
"4.193.53.89",1139185142,189,6,31.5,81.17,183,1,0,0,0.5,
0.33,0,0.167,FALSE

"5.27.104.12",1139842300,60,34,1.76,7.02,34,0,0,0.94,1,
0.5,0,0,FALSE

"130.40.105.12",1139255454,53,7,7.57,8.04,19,0,0,0.86,1, 0,0,0,F

"120.84.26.12",1140458167,52,8,6.5,4.50,17,4,0,0,1, 0,0,0.875,FA

"100.12.31.12",1138974956,3,32,0.09,0.30,1,0,0,0.97,1, 1,0,0,FAL

"67.225.39.12",1138901232,1239,5,247.8,618.83,1238,0,
0.2,0.6,1,0,0,0.5,FALSE
```

Detection of web robots traffic

- Fit Random Forests model with a total of 12 features
- Classification accuracy as performance metric
  - To keep model fitting simple
  - Sensitive to misclassify non-robot access
    - Computer program identifies potential web robots
    - Verification by human involvement
  - Achieved an accuracy at about 96%
- Potential issues
  - Noise in the label (Robot Vs Non)
    - Many random factors play (e.g., order, mood of labeler etc)
    - labels by different labelers may not always agree
  - Drift in data distribution in training/test set
  - ♠ All examples of *data contamination*.

Classification under data contamination

Data Contamination

Label noise as a form of data contamination

- Common in difficult labeling tasks
- Important for any applications that use label
  - Experiments on label noise (Dietrich 1998)
  - Robustness under label noise (RF and Adaboost, Breiman 2001)
- Existing work mostly empirical and often classifier-dependant
  - We seek to understand its nature via data contamination.

How data contamination affect web robots detection?

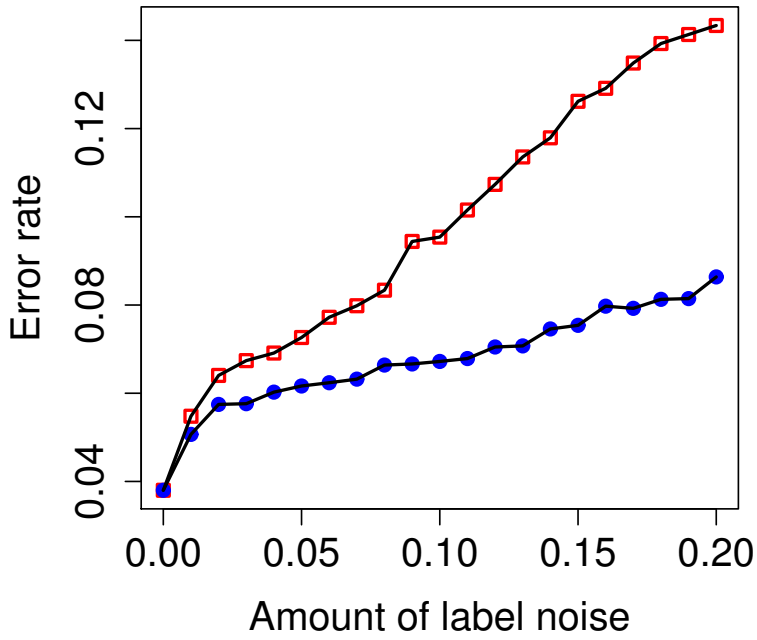


Figure: *Red: random flip of labels; blue: feature contamination on a random selected set of training examples.*

Data contamination

- Term coined by Tukey (1960)
- "No data is clean", a ubiquitous phenomenon in virtually all data
- Estimated at 0.1% to 10% for typical data (Hampel, 1971)
- Examples
  - Label noise (e.g., robot Vs non-robot)
  - Drift in distribution (data may change over time)
  - Accidental human errors
    - E.g., misplaced decimal points, or
    - Data misplaced across different rows/columns in a table.

A data contamination model

Let the original data distribution be $\mathbb{P}$. The distribution of the data upon contamination is modeled as

$$\tilde{\mathbb{P}} = (1 - \epsilon)\mathbb{P} + \epsilon\mathbb{G} \tag{1}$$

where

- $\mathbb{G}$ is any proper distribution
- ϵ can be thought as the proportion of data with contamination
- $\tilde{}$ to indicate a quantity associated with $\tilde{\mathbb{P}}$.

- Label noise, or attribute noise, or their combination
- Focus on classification and independent data contamination (correlated contaminations left as future work).

A data contamination bound

Theorem. Assume the classification algorithm is universally consistent (i.e., consistent for all probability distributions, e.g., kNN, SVM, AdaBoost with early stopping). An asymptotic data contamination bound is given by

$$R(\tilde{f}_n) - R^* \leq \frac{\epsilon}{1-\epsilon} + O\left(c(n)\right) \tag{2}$$

where $c(n)$ is related to the complexity of the function class used by the classification algorithm s.t. $c(n) \to 0$ as $n \to \infty$.

- Asymptotically, the loss in accuracy is bounded by $\epsilon/(1-\epsilon)$.

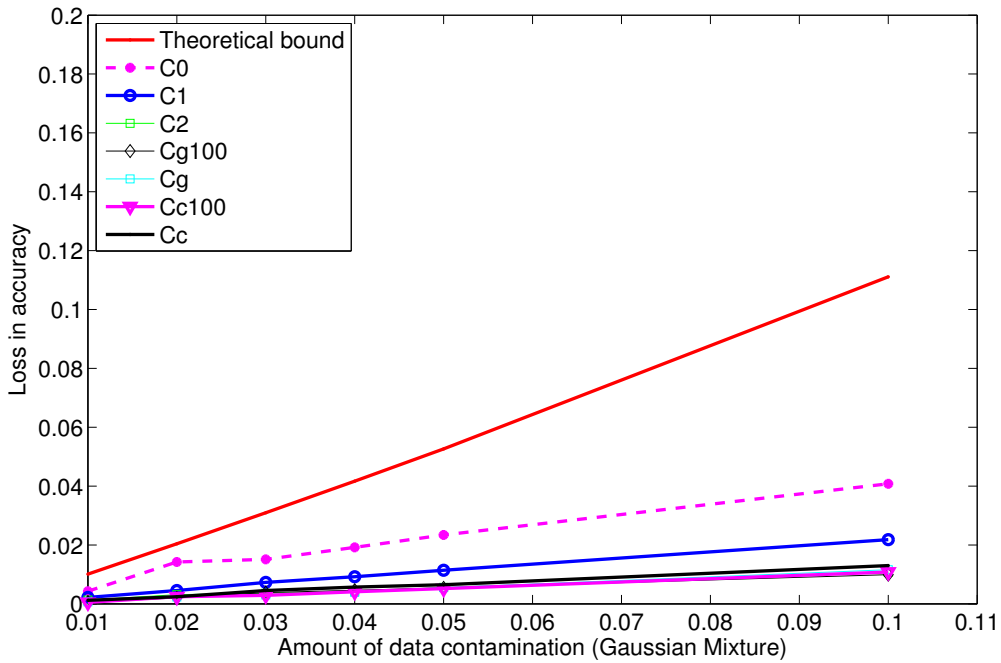
Related work on data contamination

- Statistics (mostly regression, estimation problems)
  - "Sampling from contaminated distribution" (Tukey 1960)
  - Influence function (Hampel 1971, 1974)
  - Various work on measurement error models (Hall 2009)
- Machine learning
  - Empirical work on label noise (Dietrich 1998, Breiman 2001)
  - Attribute noise (Quinlan 1986, Sloan 1988, Zhu et al 2004)
  - Domain adaption (Ben-David et al 2008, Mansour et al 2009).

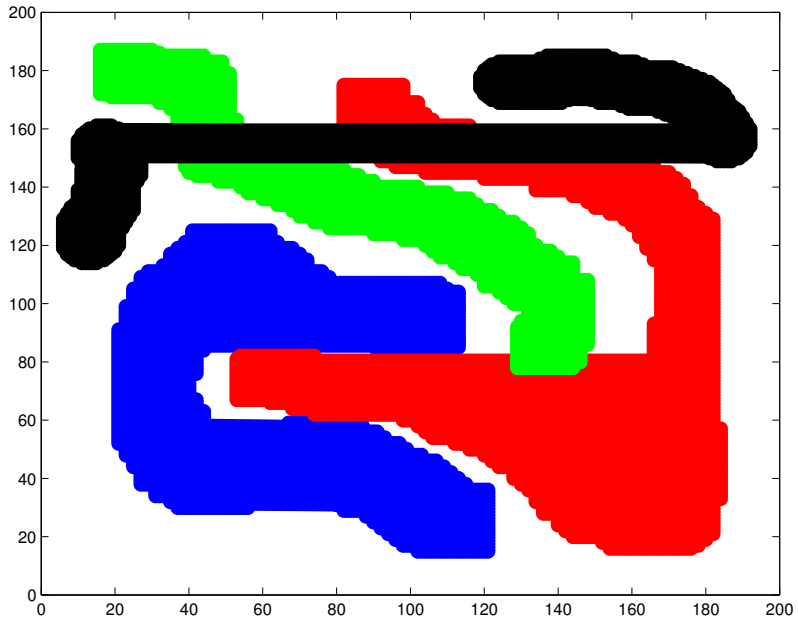
Effect with Gaussian mixtures

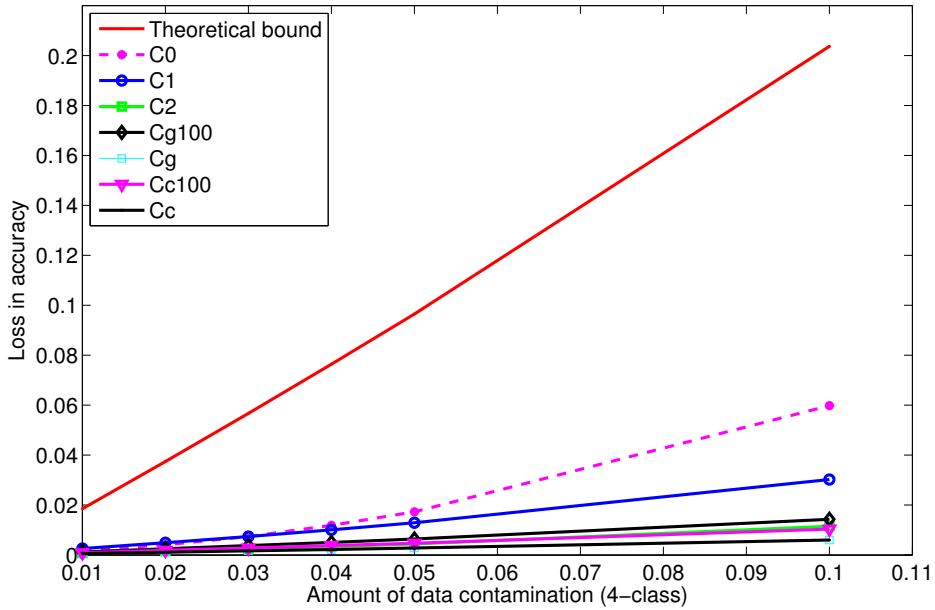
$$\frac{1}{2}\mathcal{N}(\mu, \Sigma_{10\times 10}) + \frac{1}{2}\mathcal{N}(-\mu, \Sigma_{10\times 10})$$

with $\Sigma_{10\times 10} = A^T A$ for $A \in \mathcal{U}[0,1]$, $\mu = (0.5, ..., 0.5)^T$, $n = 1000$.



Effect with Four-class data (Ho and Kleinberg 1996)





Further applications of data contamination

Further Applications

Further applications of data contamination

- Item recommendation in online retailing
- A new perspective to understand co-training.

Item recommendation in online retailing

- To recommend new items when a user
  - Visits a vendor site
  - Views an item
  - Clicks an item
  - Adds an item to the cart
  - Checkouts from a transaction
- The idea is to expose as many potential items to the user as possible.

Item recommendation examples

Anchor item

Recommended items



Item recommendation examples (continued)

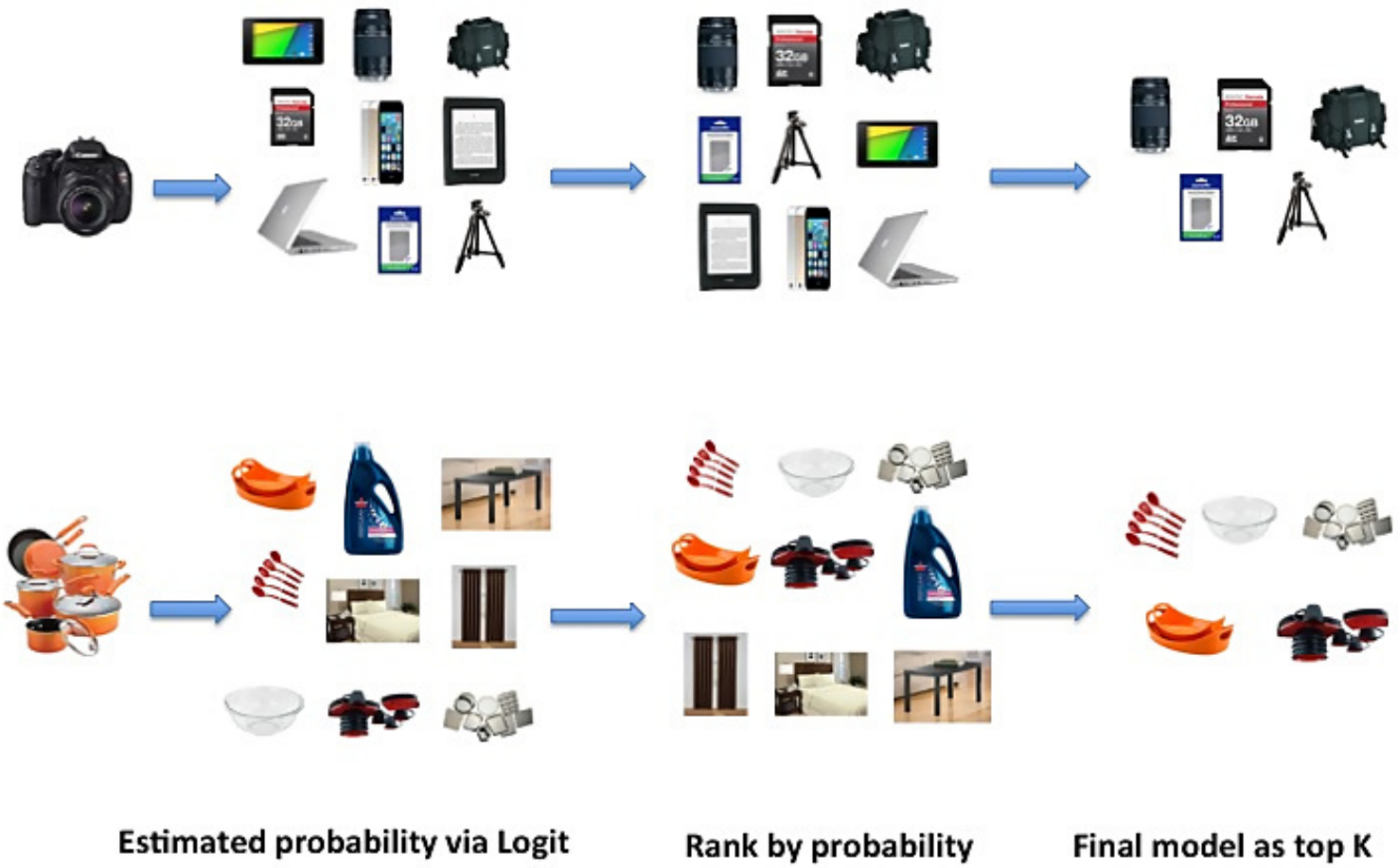
Anchor Item

Recommended items





An item recommendation model



How often should the model be refitted?

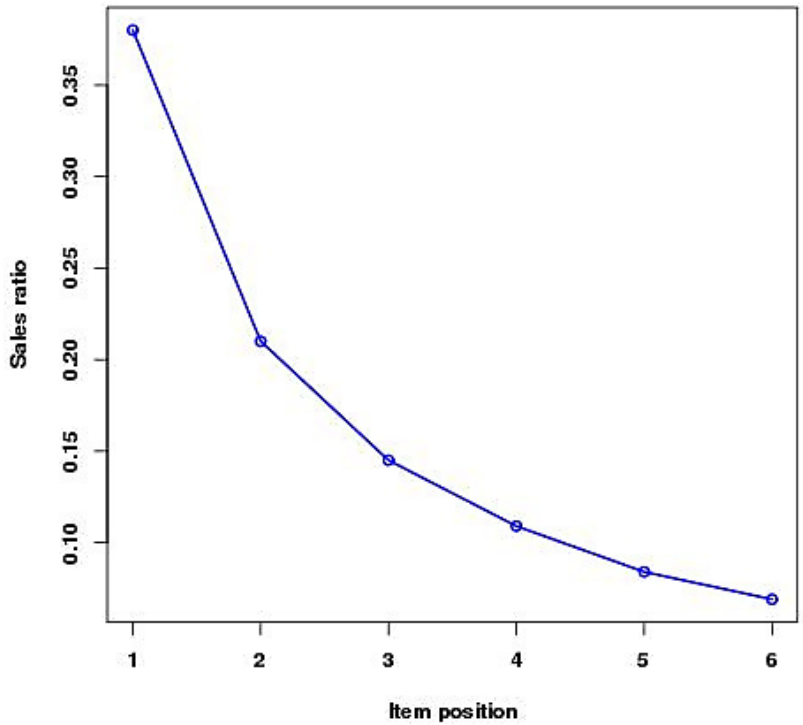
- Models in e-commerce and internet etc often data-driven and fitted from historical data
- However, such historical data may gradually become too "old"
  - Due to drift in data distribution over time
- How often should the model be refitted?
  - An important Q in many data driven model fitting practices
  - Should not be too frequent due to big data issue
    - E.g., 8 million clicks at walmart.com per day.

How often should the model be refitted?

- Our approach
  - How sensitive is the model to drift in data distribution?
  - Captured by data contamination
- Rely heavily on simulation as initial start
  - Theoretical development as future work
  - Parallel to previous work on data contamination/perturbation for classification/clustering.

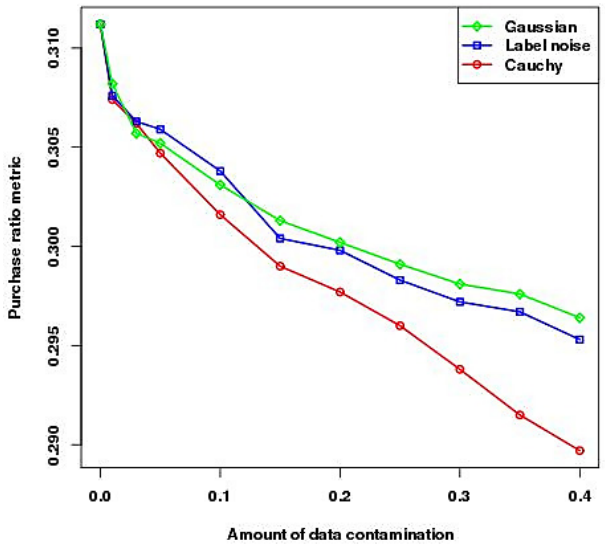
A measure of model performance

$\rho = \rho_1 - \rho_5$ where ρ_i is the percentage of purchases at position i during a "future" period of time.



Influence of data contamination to modeling

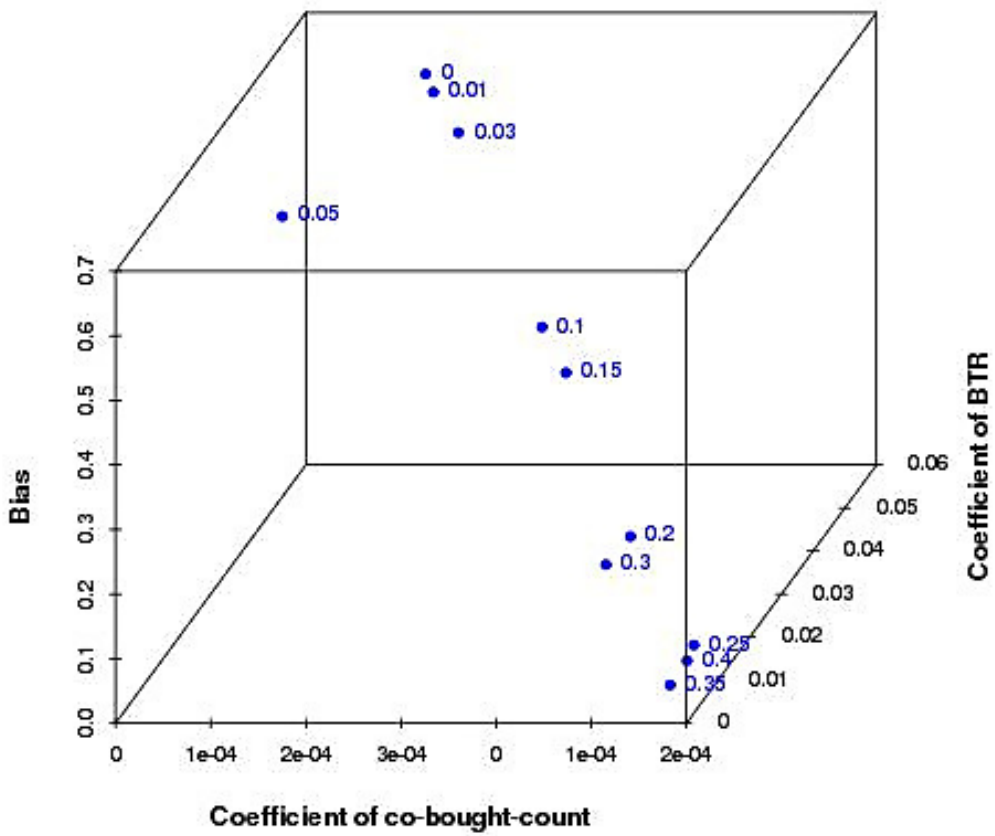
- Observation
  - Influence of data contamination surprisingly small
    - Model much less sensitive than in classification setting
  - Model "degrades" only 2% even at data contamination level 0.4
  - Model fairly robust against drift in data distribution over time
    - Model can be refitted less frequently.



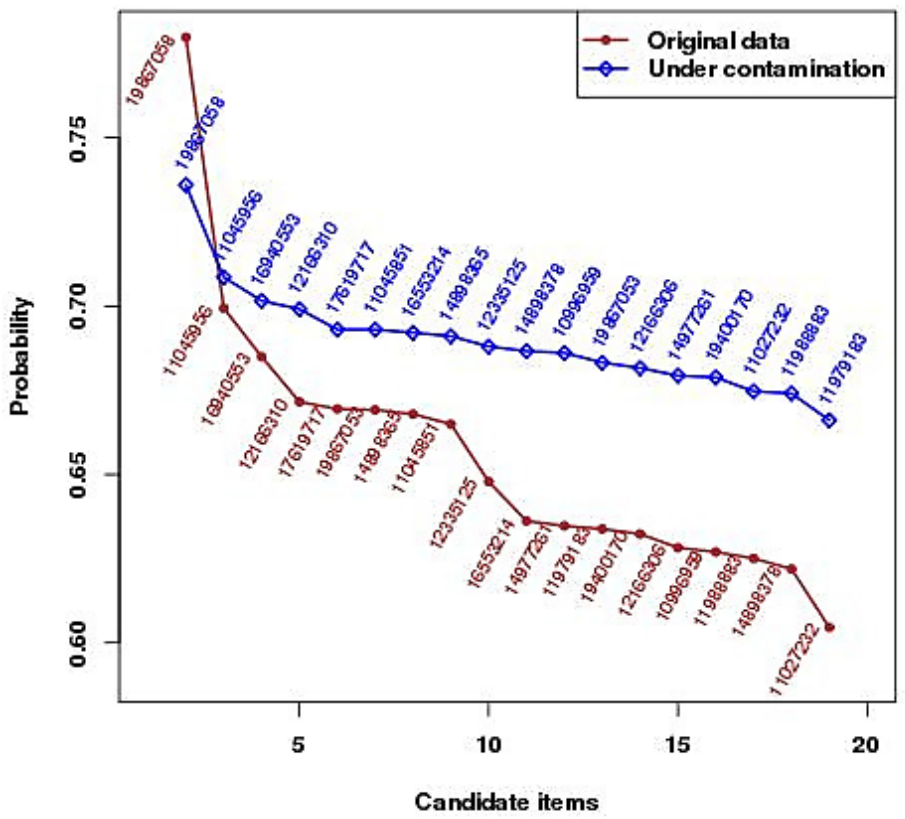
Influence of data contamination to modeling

- To gain understanding
  - Decompose the entire model into two stages
    - Fit Logit model for estimating probability of buying a candidate item conditional on buying the anchor item
    - Rank the probabilities and truncate to top K
  - Probe model performance in each stage.

Effect of data contamination to Logit model fitting



Influence of data contamination to Logit-based ranking



Ranking under data contamination

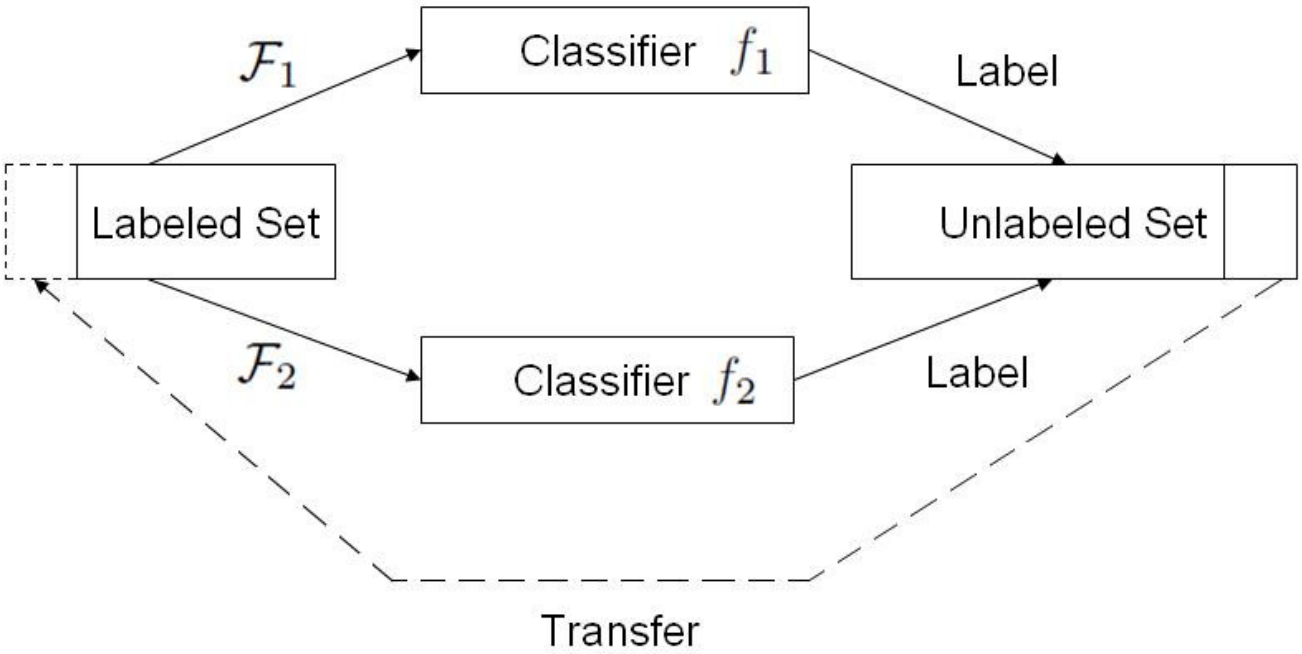
- Observation
  - Top items appear stable but lower-ranked items vary significantly
- "Modeling the variability of rankings"
  - Hall and Miller, Ann Stat 38(5), 2010

> For better or for worse, rankings of institutions, such as universities, schools and hospitals, play an important role today in conveying information about relative performance. They inform policy decisions and budgets, and are often reported in the media. While overall rankings can vary markedly over relatively short time periods, it is not unusual to find that the ranks of a small number of "highly performing" institutions remain fixed, even when the data on which the rankings are based are extensively revised, and even when a large number of new institutions are added to the competition. In the present paper,

Co-training

- Proposed in the seminal paper by Blum and Mitchell (1998)
  - ICML/COLT 10 years best paper (2008)
- Excellent performance with ridiculously small labeled set
  - 12 for web page classification (Blum and Mitchell, 1998)
  - 6 for newsgroup classification (Nigam and Ghani, 2000)
- Can often improve performance on small training sample.

Co-training (Blum and Mitchell, 1998)



Progress of co-training ($|\mathcal{L}| = 30$)

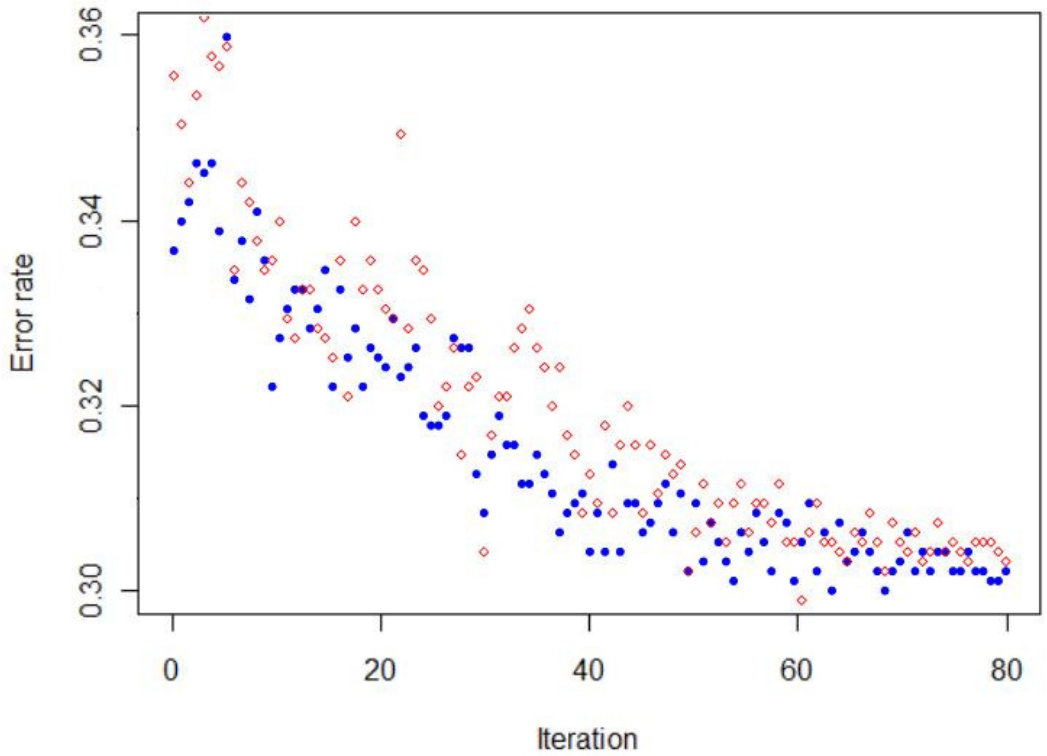


Figure: Co-training on scoring of cancer tissue images.

Work to understand co-training

- Blum and Mitchell (1998)
  - Unlabeled examples help prune away "incompatible" concepts so fewer labeled examples required to learn
- PAC learning bound (Dasgupta, Littman and McAllester 2001)
- Maximizing agreement on unlabeled data (Abney 2002)
- Balcan, Blum and Yang 2004
  - "Expansion" assumption about the hypothesis class.

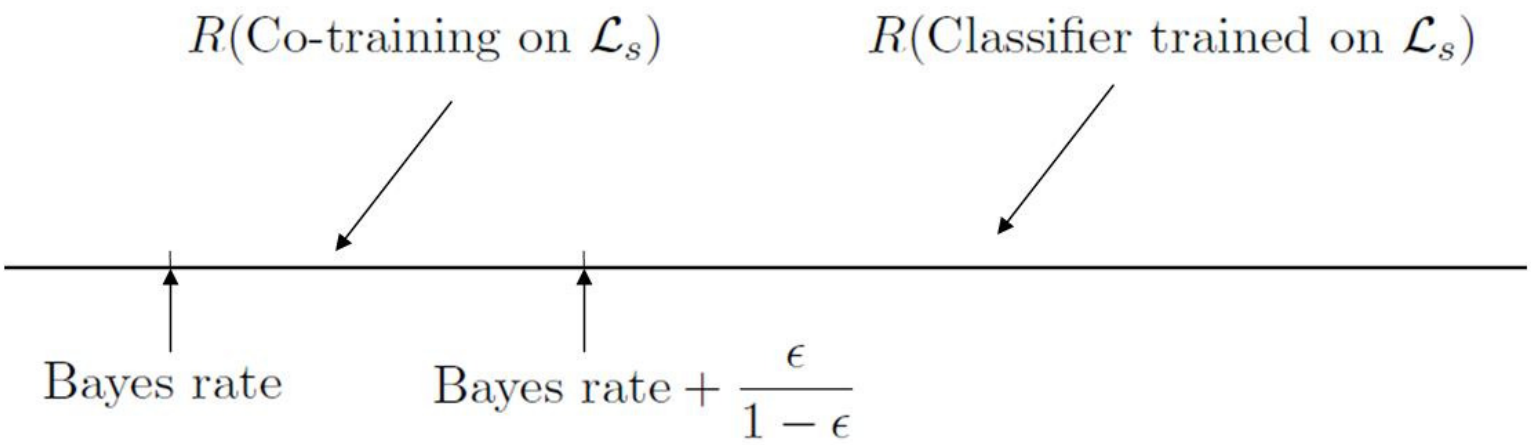
Understanding co-training with data contamination

- Data contamination to understand co-training
  - Progress of co-training $\longleftrightarrow$ adding label noise ϵ
    - Blum and Mitchell (1998)
  - Small labeled set $\mathcal{L}_s$ from $\mathbb{G}$ $\xrightarrow{\text{co-training}}$ large labeled set $\mathcal{L}_l$ from $\tilde{\mathbb{G}}$ with label noise. Thus

$$\begin{aligned} R(\text{Co-training on } \mathcal{L}_s) &= R(\text{Classifier trained on } \mathcal{L}_l) \\ &\leq \text{Bayes rate}_{\mathbb{G}} + \frac{\epsilon}{1-\epsilon} + O\left(c(|\mathcal{L}_l|)\right) \end{aligned}$$

  - Large unlabeled set $\rightarrow$ large $|\mathcal{L}_l|$ and ϵ depends on "confidence".

Understand co-training with data contamination



Summary

- Data contamination model to understand label noise and distribution drift
  - Applicable to a wide range of applications
- Statistical quantification on influence of data contamination to classification.

Acknowledgements

- Collaborators
  - Pei Wang (Mount Sinai School of Medicine)
  - Timothy Randolph (FHCRC)
  - Aiyou Chen (Google)
  - Peng Gong (Berkeley)
- Pathologists
  - Beatrice Knudsen (Cedar-Sinai Medical Center)
  - Michael Linden (U Minnesota)

The end

Thank you!