Class Ratio Estimation using MMD

J. SAKETHA NATH (IIT BOMBAY)
COLLABORATORS: ARUN IYER (YAHOO!), SUNITA SARAWAGI (IIT B)
Motivation

Excellent middle eastern cuisine on historic Murphy avenue in Sunnyvale. We had a reservation for 8, and they were kind enough to seat us outdoors, which was wonderful on this beautiful day in...

Came here for the first time a couple weeks ago on a week night - wait was not that bad. We were seated promptly and had time to look over menu. I ordered the Beriani Dajaj with Chicken (I saw...

SO MAD! I have been driving past this place for months now. It always looked good, and the pictures online looked lovely. Sadly, not the case when you come in. I walked in and no one was at the...

Yahoo! Local Restaurant Reviews
Motivation

Excellent middle eastern cuisine on historic Murphy avenue in Sunnyvale. We had a reservation for 8, and they were kind enough to seat us outdoors, which was wonderful on this beautiful day in...

Came here for the first time a couple weeks ago on a week night - wait was not that bad. We were seated promptly and had time to look over menu. I ordered the Beriani Dajaj with Chicken (I saw...

SO MAD! I have been driving past this place for months now. It always looked good, and the pictures online looked lovely. Sadly, not the case when you come in. I walked in and no one was at the...

Yahoo! Local Restaurant Reviews

Laborious
Too many reviews!
Motivation

Excellent middle eastern cuisine on historic Murphy avenue in Sunnyvale. We had a reservation for 8, and they were kind enough to seat us outdoors, which was wonderful on this beautiful day in...more

Came here for the first time a couple weeks ago on a week night - wait was not that bad. We were seated promptly and had time to look over menu. I ordered the Beriani Dajaj with Chicken (I saw...more

SO MAD! I have been driving past this place for months now. It always looked good, and the pictures online looked lovely. Sadly, not the case when you come in. I walked in and no one was at the...more
Definition: Class Ratio Estimation

- Estimate fraction of instances belonging to each class in unlabelled set
  - Need **not** estimate per-instance labels
- Pose as supervised Learning problem
  - Labelled training instances
A key issue

- Negative
- Positive

Chart 1:
- 10/60
- 50/60
- 0
- 50
- 100

Chart 2:
- 20/100
- 80/100
- 0
- 50
- 100

Chart 3:
- 30/75
- 45/75
- 0
- 50
A key issue

- Training, test distr. may be different
- Class ratios vary
- Class-conditionals are same
Existing methods

- **Multi-class classification** *(Baseline)*
  - Optimized for instance level accuracy
  - Class shift is not well-studied

- **Class ratio estimation** *(train, test class conditionals are same)*
  - F-divergence based [PS12]
  - Maximum mean discrepancy [Zh13]
  - No theoretical analysis
Existing methods

- **Multi-class classification** (Baseline)
  - Optimized for instance level accuracy
  - Class shift is not well-studied

- **Class ratio estimation** (train, test class conditionals are same)
  - F-divergence based [PS12]
  - Maximum mean discrepancy [Zh13]
  - No theoretical analysis
Notation

Given:
- Labelled training set $L = \{ (x_1, y_1), \ldots, (x_l, y_l) \}$, $y_i \in \{1, \ldots, c\}$.
- Unlabelled set $U = \{ z_1, \ldots, z_u \}$
- Universal Kernel $k$, its feature map $\phi$, and its RKHS $H$

Goal: Find fraction of each class in $U$
- i.e., find $\theta_1, \ldots, \theta_c$
Notation

- Given:
  - Labelled training set \( L = \{ (x_1, y_1), \ldots, (x_l, y_l) \} \), \( y_i \in \{1, \ldots, c\} \).
  - Unlabelled set \( U = \{ z_1, \ldots, z_u \} \).
  - Universal Kernel \( k \), its feature map \( \phi \), and its RKHS \( H \).

- Goal: Find fraction of each class in \( U \)
  - i.e., find \( \theta_1, \ldots, \theta_c \).

- Key assumption: \( P_{X/Y}^L = P_{X/Y}^U \)
  - \( P_{Y}^U \) need not be \( P_{Y}^L \).
  - \( P_{Y}^U \) may be de-generate!
MMD based method

- Idea:
  - \( P^U_X (x) = \sum_{i=1}^c P^U_Y (i) P^U_{X/Y}(x/i) \)
MMD based method

- Idea:
  \[ P_X^U(x) = \sum_{i=1}^{c} \theta_i P_{X/y}^U(x/i) \]
MMD based method

- Idea:
  - $P_X^U(x) = \sum_{i=1}^c \theta_i P_{X/Y}^L(x/i)$
MMD based method

- Idea:
  - $P_X^U(x) = \sum_{i=1}^{c} \theta_i P_{X/Y}^L(x/i)$
  - Find $\theta$ minimizes dist. between above
    - Use MMD as distance
MMD based method

- Idea:
  - $P_X^U(x) = \sum_{i=1}^{c} \theta_i P_{X/Y}^L(x/i)$
  - Find $\theta$ minimizes dist. between above
    - Use MMD as distance

- Maximum Mean Discrepancy (MMD) [FM53]
  - $MMD(P_1, P_2) \equiv \|E_{P_1}[\phi(X)] - E_{P_2}[\phi(X)]\|_H$, where $k$ is universal
MMD based method

- **Idea:**
  - $P_X^U(x) = \sum_{i=1}^{c} \theta_i P_{X/Y}^L(x/i)$
  - Find $\theta$ minimizes dist. between above
    - Use MMD as distance

- **Maximum Mean Discrepancy (MMD)** [FM53]
  - $MMD(P_1, P_2) \equiv \left\| E_P[\phi(X)] - E_P[\phi(X)] \right\|_H$, where $k$ is universal

\[
\min_{\theta \in \Delta_c} \left\| E_{P_X^U}[\phi(X)] - \sum_{i=1}^{c} \theta_i E_{P_X^L}[\phi(X)/i] \right\|_H^2
\]
MMD based method

- Idea:
  - $P_X^U(x) = \sum_{i=1}^c \theta_i P_{X/Y}^L(x/i)$
  - Find $\theta$ minimizes dist. between above
    - Use MMD as distance

- Maximum Mean Discrepancy (MMD) [FM53]
  - $\text{MMD}(P_1, P_2) \equiv \|E_{P_1}[\phi(X)] - E_{P_2}[\phi(X)]\|_H$, where $k$ is universal

$$ \approx \min_{\theta \in \Delta c} \left\| \frac{1}{u} \sum_{j=1}^u \phi(z_j) - \sum_{i=1}^c \theta_i \left( \frac{1}{l_i} \sum_{j=1}^{l_i} \phi(x_j) \right) \right\|_2^2 $$
MMD based method

- Idea:
  - $P_X^U(x) = \sum_{i=1}^c \theta_i P_{X/Y}^L(x/i)$
  - Find $\theta$ minimizes dist. between above
    - Use MMD as distance
  - Maximum Mean Discrepancy (MMD) [FM53]
    - $MMD(P_1, P_2) \equiv \|E_{P_1}[\phi(X)] - E_{P_2}[\phi(X)]\|_H$, where $k$ is universal

$$\approx \min_{\theta \in \Delta c} \left\| \frac{1}{u} \sum_{j=1}^u \phi(z_j) - \sum_{i=1}^c \theta_i \left( \frac{1}{l_i} \sum_{j=1}^{l_i} \phi(x_j) \right) \right\|_2^2$$
MMD based method

- Idea:
  - \( P_X^U(x) = \sum_{i=1}^{c} \theta_i P_{X/Y}^L(x/i) \)
  - Find \( \theta \) minimizes dist. between above
    - Use MMD as distance

- Maximum Mean Discrepancy (MMD) [FM53]
  - \( MMD(P_1, P_2) \equiv \| E_{P_1}[\phi(X)] - E_{P_2}[\phi(X)] \|_H \), where \( k \) is universal

\[ \approx \min_{\theta \in \Delta_c} \left\| \frac{1}{u} \sum_{j=1}^{u} \phi(z_j) - \sum_{i=1}^{c} \theta_i \left( \frac{1}{l_i} \sum_{j=1}^{l_i} \phi(x_j) \right) \right\|^2 \]
MMD based method

- Idea:
  - $P_X^U(x) = \sum_{i=1}^{c} \theta_i P_{X/Y}^L(x/i)$
  - Find $\theta$ minimizes dist. between above
    - Use MMD as distance
  - Maximum Mean Discrepancy (MMD) [FM53]
    - $MMD(P_1, P_2) \equiv \left\| E_{P_1}[\phi(X)] - E_{P_2}[\phi(X)] \right\|_H$, where $k$ is universal

\[ \approx \min_{\theta \in \Delta_c} \left\| \frac{1}{u} \sum_{j=1}^{u} \phi(z_j) - \sum_{i=1}^{c} \theta_i \left( \frac{1}{l_i} \sum_{j=1}^{l_i} \phi(x_j) \right) \right\|_2^2 \]
Key Contributions

- Theoretical Analysis
  - Derive learning bounds
  - Simple proof
  - Works with de-generate $P_Y^U$
Key Contributions

- Theoretical Analysis
  - Derive learning bounds
  - Simple proof
  - Works with de-generate $P^U_Y$

- Hints at right kernel
  - SDP formulation for kernel learning (convex!)
  - Improved generalization
Theorem

\[ \hat{\theta} \equiv \arg \min_{\theta \in \Delta_c} \left\| \frac{1}{u} \sum_{j=1}^{u} \phi(z_j) - \sum_{i=1}^{c} \theta_i \left( \frac{1}{l_i} \sum_{j=1}^{l_i} \phi(x_j) \right) \right\|_2^2 \]
Theorem

\[ \hat{\theta}_{1:c-1} \equiv \text{argmin}_{\theta \in \Lambda_c} (h(\theta) \equiv \|A\theta - a\|_2^2), \]

\[ A^i = \frac{1}{l_i} \sum_{j=1}^{l_i} \phi(x_j) - \frac{1}{l_c} \sum_{j=1}^{l_c} \phi(x_j) \]
Theorem

\[ \hat{\theta}_{1:c-1} \equiv \arg\min_{\theta \in \Lambda_c} h(\theta) \equiv \| A\theta - a \|_2^2, \]

\[ A^i = \frac{1}{l_i} \sum_{j=1}^{l_i} \phi(x_j) - \frac{1}{l_c} \sum_{j=1}^{l_c} \phi(x_j) \]

If \( A \) has full column rank, then with probability at least \( 1 - \delta \), we have:

\[ \| \hat{\theta} - \theta^* \|_2^2 \leq \frac{R^2 \left( \frac{c^2 + 1}{u} + \sum_{i=1}^{c} \frac{2}{l_i} \right) \left( 1 + \sqrt{\log \frac{2}{\delta}} \right)^2}{\text{mineig}(A^T A)} \]
Theorem

\[ \hat{\theta}_{1:c-1} \equiv \arg\min_{\theta \in \Lambda_c} h(\theta) \equiv \|A\theta - a\|_2^2, \]

If \( A \) has full column rank, then with probability at least \( 1 - \delta \), we have:

\[ \|\hat{\theta} - \theta^*\|_2^2 \leq \frac{R^2 \left( \frac{c^2 + 1}{u} + \sum_{i=1}^{c^2} \frac{2}{l_i} \right) \left( 1 + \sqrt{\log \frac{2}{\delta}} \right)^2}{\text{mineig}(A^TA)}. \]
Theorem

\[ \hat{\theta}_{1:c-1} \equiv \text{argmin}_{\theta \in \Lambda_c} (h(\theta) \equiv \|A\theta - a\|_2^2), \]

If \( A \) has full column rank, then with probability at least \( 1 - \delta \), we have:

\[ \| \hat{\theta} - \theta^* \|_2^2 \leq \frac{R^2 \left( \frac{c^2 + 1}{u} + \sum_{i=1}^{c} \frac{2}{l_i} \right) \left( 1 + \sqrt{\log \frac{2}{\delta}} \right)^2}{\text{mineig}(A^T A)} \]
Theorem

\[ \hat{\theta}_{1:c-1} \equiv \arg\min_{\theta \in \Lambda_c} h(\theta) \equiv \|A\theta - a\|_2^2, \]

If \( A \) has full column rank, then with probability at least \( 1 - \delta \), we have:

\[ \|\hat{\theta} - \theta^*\|_2^2 \leq \frac{R^2 \left( \frac{c^2 + 1}{u} + \sum_{i=1}^{c} \frac{2}{l_i} \right) \left( 1 + \sqrt{\log \frac{2}{\delta}} \right)^2}{\text{mineig}(A^T A)} \]
Theorem

\[ \hat{\theta}_{1:c-1} \equiv \arg\min_{\theta \in \Lambda_c} h(\theta) \equiv \|A\theta - a\|_2^2, \]

If \( A \) has full column rank, then with probability at least \( 1 - \delta \), we have:

\[ \|\hat{\theta} - \theta^*\|_2^2 \leq R^2 \left( \frac{c^2 + 1}{u} + \sum_{i=1}^c \frac{2}{l_i} \right) \left( 1 + \sqrt{\log \frac{2}{\delta}} \right)^2 \cdot \text{mineig}(A^T A) \]
Theorem

\[ \hat{\theta}_{1:c-1} \equiv \arg \min_{\theta \in \Lambda_c} h(\theta) \equiv \| A \theta - a \|_2^2, \]

\[ A^i = \frac{1}{l_i} \sum_{j=1}^{l_i} \phi(x_j) - \frac{1}{l_c} \sum_{j=1}^{l_c} \phi(x_j) \]

If \( A \) has full column rank, then with probability at least \( 1 - \delta \), we have:

\[ \| \hat{\theta} - \theta^* \|_2^2 \leq \frac{R^2 \left( \frac{c^2 + 1}{u} + \sum_{i=1}^{c} \frac{2}{l_i} \right) \left( 1 + \sqrt{\log \frac{2}{\delta}} \right)^2}{\text{mineig}(A^T A)} \]
Proof sketch

- TST: \( \{h(\theta^*) - h(\hat{\theta})\} \xrightarrow{p} 0 \), as \( l, u \to \infty \)
  - \( h(\theta^*) \) satisfies bounded difference property
  - Follows from Mc Diarmid’s inequality and upper bounding \( E[h(\theta^*)] \)
Proof sketch

- **TST:** \( \{ h(\theta^*) - h(\hat{\theta}) \} \overset{p}{\rightarrow} 0, \) as \( l, u \to \infty \)
  - \( h(\theta^*) \) satisfies bounded difference property
  - Follows from Mc Diarmid’s inequality and upper bounding \( \mathbb{E}[h(\theta^*)] \)

- **TST:** \( \| \hat{\theta} - \theta^* \|_2^2 \leq \frac{h(\theta^*) - h(\hat{\theta})}{\text{mineig}(A^T A)} \)
  - Optimality conditions at \( \hat{\theta} \)
  - Elementary properties of quadratic
Kernel Learning

- Pre-processing step (otherwise also possible)
- Given: Universal $k_1, \ldots, k_n$
- Goal: optimize $w \geq 0$ for “best” $k = \sum_{i=1}^{n} w_i k_i$
Kernel Learning

- Pre-processing step (otherwise also possible)
- Given: Universal $k_1, \ldots, k_n$
- Goal: optimize $w \geq 0$ for “best” $k = \sum_{i=1}^{n} w_i k_i$

- Two objectives:
  - $w$ that minimizes terms in bound
  - $w$ that minimizes an empirical term
Kernel Learning – bound terms

\[ \text{mineig}(A^T A) = \text{mineig} \left( \sum_{i=1}^{n} w_i A_i^T A_i \right) \]

- Maximization of above term is convex
- In fact, expressible as LMI
Kernel Learning – bound terms

\[ \text{mineig}(A^T A) = \text{mineig} \left( \sum_{i=1}^{n} w_i A_i^T A_i \right) \]

- Maximization of above term is convex
- Infact, expressible as LMI

\[ R^2 = \sum_{i=1}^{n} w_i^2 R_i^2 = \|w\|_2^2 \] (normalized kernels)
- Minimization of above term is convex
Kernel Learning – empirical term

- Empirical term: $w$ is indeed good for several unlabelled sets
- Unlabelled sets generated from $L$
Kernel Learning – empirical term

- Empirical term: $w$ is indeed good for several unlabelled sets
- Unlabelled sets generated from $L$

\[ U_1 \quad U_2 \quad U_m \]

$L$ is divided into Positive and Negative conditional sampling with $\theta_i^*$ known.
Kernel Learning – empirical term

- Won’t work:
  - $\|\hat{\theta}_i^w - \theta_i^\star\| \leq \epsilon \ \forall \ i$
  - $|h_i^w(\hat{\theta}_i^w) - h_i^w(\theta_i^\star)| \leq \epsilon \ \forall \ i$
  - Both non-convex in $w$
  - Both do not avoid *extraneous* solutions
Kernel Learning – empirical term

- Won’t work:
  - $\|\hat{\theta}_i^w - \theta_i^*\| \leq \epsilon \ \forall \ i$
  - $|h_i^w(\hat{\theta}_i^w) - h_i^w(\theta_i^*)| \leq \epsilon \ \forall \ i$
  - Both non-convex in $w$
  - Both do not avoid *extraneous* solutions

- Our idea:
  - $h_i^w(\theta) - h_i^w(\theta_i^*) \geq 1 \ \forall \ \|\theta - \theta_i^*\| > \epsilon$
Kernel Learning – empirical term

Won’t work:
- $\|\hat{\theta}_i^w - \theta_i^*\| \leq \epsilon \ \forall \ i$
- $|h_i^w(\hat{\theta}_i^w) - h_i^w(\theta_i^*)| \leq \epsilon \ \forall \ i$
- Both non-convex in $w$
- Both do not avoid extraneous solutions

Our idea:
- $h_i^w(\theta) - h_i^w(\theta_i^*) \geq \rho(\theta, \theta_i^*) - \xi_i \ \forall \ |\theta - \theta_i^*| > \epsilon, \xi_i \geq 0$
Kernel Learning – empirical term

- Won’t work:
  - $\|\hat{\theta}_i^w - \theta_i^*\| \leq \epsilon \ \forall \ i$
  - $|h_i^w(\hat{\theta}_i^w) - h_i^w(\theta_i^*)| \leq \epsilon \ \forall \ i$
  - Both non-convex in $w$
  - Both do not avoid extraneous solutions

- Our idea:
  - $w^T u_i \geq \rho(\theta, \theta_i^*) - \xi_i \ \forall \ \|\theta - \theta_i^*\| > \epsilon, \xi_i \geq 0$
  - Convex and avoids extraneous solutions
SDP formulation for Kernel Learning

\[
\begin{align*}
\min_{w \in \mathbb{R}^n, \xi \in \mathbb{R}^m} & \quad \|w\|_2 + B \max\text{eig} \left( - \sum_{i=1}^{n} w_i A_i^T A_i \right) + C \sum_{i=1}^{m} \xi_i \\
\text{s.t.} & \quad w^T u_i \geq \rho(\theta, \theta^*_i) - \xi_i \quad \forall \|\theta - \theta^*_i\| > \epsilon, \xi_i \geq 0
\end{align*}
\]
SDP formulation for Kernel Learning

\[
\begin{align*}
\min_{w \in \mathbb{R}^n, \xi \in \mathbb{R}^m} & \quad \|w\|_1 + B \max_{\text{eig}} \left( - \sum_{i=1}^{n} w_i A_i^T A_i \right) + C \sum_{i=1}^{m} \xi_i \\
\text{s.t.} & \quad w^T u_i \geq \rho(\theta, \theta_i^*) - \xi_i \quad \forall \|\theta - \theta_i^*\| > \epsilon, \xi_i \geq 0
\end{align*}
\]
SDP formulation for Kernel Learning

\[
\begin{align*}
\min_{w \in \mathbb{R}^n, \xi \in \mathbb{R}^m} & \quad \|w\|_1 + B \max_{i=1}^n \left( - \sum_{i=1}^n w_i A_i^T A_i \right) + C \sum_{i=1}^m \xi_i \\
\text{s.t.} & \quad w^T u_i \geq \rho(\theta, \theta_i^*) - \xi_i \quad \forall \|\theta - \theta_i^*\| > \epsilon, \xi_i \geq 0
\end{align*}
\]

Solved using cutting planes algorithm [Ar14]
Results: Binary Class Dataset (UCI)

Varying Negative Class Proportions in U (proportion in D is set to [0.5, 0.5])
Results: Binary Class Dataset (UCI)

Direct methods are flat and instance based is not!
Results: Binary Class Dataset (UCI)

Varying Negative Class Proportions in U
(proportion in D is set to [0.5, 0.5])

Estimation Error

Ger
dan

MMD-MKL
MMD
PE-DR
SMO-MKL

Instance based is good only when $p_U \approx p_L$. 
Results: Binary Class Dataset (UCI)

Varying Negative Class Proportions in U
(proportion in D is set to [0.5, 0.5])

Flatness and acc.: MMD-MKL > MMD > PE-DR
Other binary classification results

Varying Negative Class Proportions in $U$
(proportion in $D$ is set to $[0.5, 0.5]$)

Same trend!
Variation with data size

Multi-class datasets

Baselines need not improve with data size!
Summary

- MMD based estimator for class ratio estimation
- Learning bounds for it
- Bounds provide insight for kernel learning
- SDP formulation for kernel learning
- MMD+MKL improves state-of-the-art
  - Upto 60% overall
  - Upto 40% because of kernel learning
Thanks for listening. Questions?
References


Effect of bound
Kernels