Logistic Regression with an Auxiliary Data Source

Xuejun Liao

Department of Electrical and Computer Engineering
Duke University
Durham, NC 27708-0291, USA
Outline

- What is a classification problem
  - When training data and test data are mismatched ...
  - Migratory-Logit
  - Fast learning of Migratory-Logit
  - Active selection of labelled data
  - A toy example
  - An example on UC Irvine Data
What is a classification problem?

- A data generator $\Pr(x)$
- The true class labeler $\Pr(y|x)$
- What is known: training examples of $(x, y) \sim \Pr(x)\Pr(y|x)$
- The goal: predict $y$ for $x \sim \Pr(x)$
- Point classifier: build a classifier from training instances.
- Ensemble classifiers

$$\Pr(y_{t+1}|x_{t+1}, w), \quad w \sim \Pr(w|x_{1:t}, y_{1:t})$$

predictive distribution

$$\Pr(y_{t+1}|x_{t+1}, x_{1:t}, y_{1:t}) = \int_w \Pr(y_{t+1}|x_{t+1}, w)\Pr(w|x_{1:t}, y_{1:t})$$
Outline

- What is a classification problem
- When training data and test data are mismatched ...
  - Migratory-Logit
  - Fast learning of Migratory-Logit
  - Active selection of labelled data
- A toy example
- An example on UC Irvine Data
When $Pr(x)Pr(y|x)$ varies ...

- Collect data at one UXO site, and make predictions about data at another UXO site
- Will the data be the same?
- Many factors affect data: weather, soil conditions, types of UXO, etc
Selection bias

- True distribution $\Pr(x, y)$
- Biased distribution $\Pr(x, y|s = 1)$, where $(x, y)$ are drawn if and only if $s = 1$
- Let test examples $(x, y) \sim \Pr(x, y)$
- Let training examples $(x, y) \sim \Pr(x, y|s = 1)$
- A simple equation:

$$\Pr(x, y) = \frac{\Pr(s = 1)}{\Pr(s = 1|x, y)} \Pr(x, y|s = 1) \tag{1}$$

Is $s$ independent of $(x, y)$?

- Estimate weighting coefficient $\frac{\Pr(s = 1)}{\Pr(s = 1|x, y)}$?
Reformulating the goal

- Denote by $D^p$ a collect of examples 
  $(x, y \text{ missing}) \sim \Pr(x, y)$

- Denote by $D^a$ a collect of examples 
  $(x, y) \sim \Pr(x, y|s = 1)$

- Goal: predict the labels of $D^p$ given $D^a$ are observed
Outline

- What is a classification problem
- When training data and test data are mismatched ...
- Migratory-Logit
  - Fast learning of Migratory-Logit
  - Active selection of labelled data
  - A toy example
  - An example on UC Irvine Data
Migratory Classification

- Let us relax a little, and assume access to labels of $\mathcal{D}_l^p$, a small subset of $\mathcal{D}^p$
- Goal: predict the labels of $\mathcal{D}_u^p = \mathcal{D}^p \setminus \mathcal{D}_l^p$ given $\mathcal{D}^a \cup \mathcal{D}_l^p$ are observed
- Key idea: we introduce an auxiliary variable $\mu_i$ for each $(x^a_i, y^a_i) \in \mathcal{D}^a$
- The $\mu$ plays a similar role as the weighting factors $\frac{\Pr(s=1)}{\Pr(s=1|x,y)}$ in (1).
- The auxiliary variables can be estimated along with the classifier
Some notations

- $\mathcal{D}^p = \mathcal{D}^p_l \cup \mathcal{D}^p_u$, with $\mathcal{D}^p_l$ nonempty and fixed

- $\mathcal{D}^p_l$ always indexed prior to $\mathcal{D}^p_u$: $\mathcal{D}^p_l = \{(x^p_i, y^p_i)\}_{i=1}^{N^p_l}$ and $\mathcal{D}^p_u = \{(x^p_i, y^p_i) : y^p_i \text{ missing}\}_{i=N^p_l+1}^{N^p}$.

- Let $N^a$, $N^p$, and $N^p_l$ denote the size of $\mathcal{D}^a$, $\mathcal{D}^p$, and $\mathcal{D}^p_l$, respectively.

- $y^a, y^p \in \{-1, 1\}$

- $x$ always includes a 1 as its first element to accommodate a bias (intercept) term, thus $x^p, x^a \in \mathbb{R}^{d+1}$ where $d$ is the number of features.
Migratory-Logit

For \((x^p_i, y^p_i) \in \mathcal{D}_l^p,\)

\[
\Pr(y^p_i | x^p_i; w) = \sigma(y^p_i w^T x^p_i)
\]

(2)

where \(w \in \mathbb{R}^{d+1}\) is a column vector of classifier parameters and \(\sigma(\mu) = \frac{1}{1 + \exp(-\mu)}\) is the sigmoid function.

For \((x^a_i, y^a_i) \in \mathcal{D}^a,\) we define

\[
\Pr(y^a_i | x^a_i; w, \mu_i) = \sigma(y^a_i w^T x^a_i + y^a_i \mu_i)
\]

(3)

where \(\mu_i\) is an auxiliary variable.
Assuming $\mathcal{D}_l^p$ and $\mathcal{D}^a$ both contain i.i.d. instances, we have the log-likelihood function

$$
\ell(w, \mu; \mathcal{D}_l^p \cup \mathcal{D}^a) = \sum_{i=1}^{N_l^p} \ln \sigma(y_i^p w^T x_i^p) + \sum_{i=1}^{N^a} \ln \sigma(y_i^a w^T x_i^a + y_i^a \mu_i) \tag{4}
$$

where $\mu = [\mu_1, \cdots, \mu_{N^a}]^T$ is a column vector of all auxiliary variables.
The role of $\mu_i$

- reflect the mismatch of $(x_i^a, y_i^a)$ with $D^p$
- control $(x_i^a, y_i^a)$’s participation in the learning $w$
- If $(x_i^a, y_i^a)$ and $D^p$ are mismatched, $w$ cannot make
  $$\sum_{i=1}^{N_i^p} \ln \sigma(y_i^p w^T x_i^p)$$ and
  $$\ln \sigma(y_i^a w^T x_i^a)$$ large at the same time.
- $\mu_i$ causes $x_i^a$ to migrate towards class $y_i^a$
- $w$ is less sensitive to $(x_i^a, y_i^a)$ and can focus more on fitting $D_i^p$. 

---

Let $\mu$’s be free?

Overriding $w$ and $D^a$ will not participate in learning $w$

Constrained maximization

\[
\max_{\omega, \mu} \quad \ell(\omega, \mu; D^p \cup D^a) \\
\text{subject to} \\
\frac{1}{N^a} \sum_{i=1}^{N^a} y_i^a \mu_i \leq C, \quad C \geq 0 \\
y_i^a \mu_i \geq 0, \quad i = 1, 2, \cdots, N^a
\]

The $\mu_i$ always exerts a positive influence in fitting the data.

The classifier resulting from solving the problem in (5)-(7) is referred to as “Migratory-Logit” or “M-Logit”.

---

What is $C$?

$C = \text{average mismatch}$

An intuition

Mismatch ratio $N^m / N^a$: the fraction of $D^a$ mismatched with $D^p$

The sigmoid function $\sigma(\mu)$ saturates at $\mu = \pm 12$

By letting $C = 12N^m / N^a$, we distribute $N^a C$ only to that part of $D^a$ that is mismatched with $D^p$

As $0 \leq N^m \leq N^a$, letting $0 \leq C \leq 12$ is usually a reasonable choice.

Insensitivity
Outline

- What is a classification problem
- training data and test data are mismatched ...
- Migratory-Logit

**Fast learning of Migratory-Logit**

- Active selection of labelled data
- A toy example
- An example on UC Irvine Data
Fast Learning of Migratory-Logit

- Traditional learning
  - Is concave
  - Number of $\mu$ proportional to the size $D^a$
  - When $D^a$ is large, estimation of $\mu$’s will consume most of the computational time

Fast Learning of Migratory-Logit

- New idea
  - Alternately solve for \( w \) and \( \mu \), keeping one fixed when solving the other
  - Analytic solution of \( \mu \) for given \( w \)
- A theorem
Theorem 1 Let $f(z)$ be a twice continuously differentiable function and its second derivative $f''(z) < 0$ for any $z \in \mathbb{R}$. Let $b_1 \leq b_2 \leq \cdots \leq b_N$, $R \geq 0$, and

$$n = \max\{m : mb_m - \sum_{i=1}^{m} b_i \leq R, 1 \leq m \leq N\}$$  \hspace{1cm} (8)

Then the problem

$$\max \{z_i\} \quad \sum_{i=1}^{N} f(b_i + z_i)$$  \hspace{1cm} (9)

subject to

$$\sum_{i=1}^{N} z_i \leq R, \quad R \geq 0$$  \hspace{1cm} (10)

$$z_i \geq 0, \quad i = 1, 2, \cdots, N$$  \hspace{1cm} (11)

has a unique global solution

$$z_i = \begin{cases} \frac{1}{n} \sum_{j=1}^{n} b_j + \frac{1}{n} R - b_i, & 1 \leq i \leq n \\ 0, & n < i \leq N \end{cases}$$  \hspace{1cm} (12)
For a fixed $w$, the problem in (5)-(7) is simplified to maximizing

$$\max \sum_{i=1}^{N^a} \ln \sigma(y_i^aw^T \mathbf{x}_i^a + y_i^a \mu_i)$$  \hspace{1cm} (13)$$

subjectto

$$\frac{1}{N^a} \sum_{i=1}^{N^a} y_i^a \mu_i \leq C, \quad C \geq 0$$  \hspace{1cm} (14)$$

$$y_i^a \mu_i \geq 0, \quad i = 1, 2, \cdots, N^a$$  \hspace{1cm} (15)$$
Fast Learning of Migratory-Logit

- Applying Theorem

Assume \( y_{k_1}^a w^T x_{k_1}^a \leq y_{k_2}^a w^T x_{k_2}^a \leq \cdots \leq y_{k_N^a}^a w^T x_{k_N^a}^a \), where \( k_1, k_2, \cdots, k_{N^a} \) is a permutation of \( 1, 2, \cdots, N^a \). Then

\[
\mu_{k_i} = \begin{cases} 
\frac{1}{n} y_{k_i}^a \sum_{j=1}^{n} y_{k_j}^a w^T x_{k_j}^a + \frac{N^a}{n} y_{k_i}^a C - w^T x_{k_i}^a, & 1 \leq i \leq n \\
0, & n < i \leq N^a
\end{cases}
\]  

(16)

where

\[
n = \max \left\{ m : m y_{k_m}^a w^T x_{k_m}^a - \sum_{i=1}^{m} y_{k_i}^a w^T x_{k_i}^a \leq N^a C, 1 \leq m \leq N^a \right\}
\]  

(17)

- For a fixed \( \mu \), we solve \( w \) by gradient ascent

- The complete algorithm
Table 1: Fast Learning Algorithm of M-Logit

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Initialize ( \mathbf{w} ) and ( \mu_i = 0 ) for ( i = 1, 2, \ldots, N^a ).</td>
</tr>
<tr>
<td>2.</td>
<td>Compute the gradient ( \nabla_{\mathbf{w}} \ell ) and Hessian matrix ( \nabla^2_{\mathbf{w}} \ell ).</td>
</tr>
<tr>
<td>3.</td>
<td>Compute the ascent direction ( \mathbf{d} = - (\nabla^2_{\mathbf{w}} \ell)^{-1} \nabla_{\mathbf{w}} \ell ).</td>
</tr>
<tr>
<td>4.</td>
<td>Do a linear search for the step-size ( \alpha^* = \arg \max_{\alpha} \ell(\mathbf{w} + \alpha \mathbf{d}) ).</td>
</tr>
<tr>
<td>5.</td>
<td>Update ( \mathbf{w} ): ( \mathbf{w} \leftarrow \mathbf{w} + \alpha^* \mathbf{d} ).</td>
</tr>
<tr>
<td>6.</td>
<td>Sort ( { y_i^a \mathbf{w}^T \mathbf{x}<em>i^a }</em>{i=1}^{N^a} ) in ascending order. Assume the result is ( y_{k_1}^a \mathbf{w}^T \mathbf{x}<em>{k_1}^a \leq y</em>{k_2}^a \mathbf{w}^T \mathbf{x}<em>{k_2}^a \leq \cdots \leq y</em>{k_{N^a}}^a \mathbf{w}^T \mathbf{x}<em>{k</em>{N^a}}^a ), where ( k_1, k_2, \ldots, k_{N^a} ) is a permutation of ( 1, 2, \ldots, N^a ).</td>
</tr>
<tr>
<td>7.</td>
<td>Find the ( n ) using (17).</td>
</tr>
<tr>
<td>8.</td>
<td>Update the auxiliary variables ( { \mu_i }_{i=1}^{N^a} ) using (16).</td>
</tr>
<tr>
<td>9.</td>
<td>Check the convergence of ( \ell ): exit and output ( \mathbf{w} ) and ( { \mu_i }_{i=1}^{N^a} ) if converged; go back to 2 otherwise.</td>
</tr>
</tbody>
</table>
Outline

- What is a classification problem
- training data and test data are mismatched ...
- Migratory-Logit
- Fast learning of Migratory-Logit

**Active selection of labelled data**

- A toy example
- An example on UC Irvine Data
Active Selection of $\mathcal{D}_l^p$

- Let $Q$ denote the Fisher information matrix of $\mathcal{D}_l^p \cup \mathcal{D}^a$ about $w$.

- By definition, $Q = \mathbb{E}\{y_i^p, y_i^a\} \frac{\partial \ell}{\partial w} \frac{\partial \ell}{\partial w}^T$, and substituting (4) into this equation gives

$$Q = \sum_{i=1}^{N_l^p} \sigma_i^p (1 - \sigma_i^p) x_i^p x_i^p^T + \sum_{i=1}^{N_a} \sigma_i^a (1 - \sigma_i^a) x_i^a x_i^a^T$$

where $\sigma_i^p = \sigma(w^T x_i^p)$ for $i = 1, 2, \ldots, N_l^p$, and $\sigma_i^a = \sigma(w^T x_i^a + \mu_i)$ for $i = 1, 2, \ldots, N_a$.

- The $w$ and $\{\mu_i\}$ represent the true classifier and auxiliary variables.

- Replacing truth with estimates
Active Selection of $\mathcal{D}^p_l$

- The inverse Fisher information $Q^{-1}$ lower bounds the covariance matrix of the estimated $w$.

- In particular, $[\det(Q)]^{-1}$ lower bounds the product of variances of the elements in $w$.

- The goal in selecting $\mathcal{D}^p_l$ is to reduce the variances, or uncertainty, of $w$.

- Thus we seek the $\mathcal{D}^p_l$ that maximize $\det(Q)$. 
Active Selection of $\mathcal{D}_l^p$

### Sequential selection

- Initially $\mathcal{D}_u^p = \mathcal{D}^p$, $\mathcal{D}_l^p$ is empty, and $Q = \sum_{i=1}^{N^a} \sigma_i^a (1 - \sigma_i^a) x_i^a x_i^a^T$

- Then one at a time, a data point $x_i^p \in \mathcal{D}_u^p$ is selected and moved from $\mathcal{D}_u^p$ to $\mathcal{D}_l^p$.

- This causes $Q$ to be updated as: $Q \leftarrow Q + \sigma_i^p (1 - \sigma_i^p) x_i^p (x_i^p)^T$.

- The $w$ and $\{\mu_i\}$ are re-estimated

- At each iteration, the selection is based on

\[
\max_{x_i^p \in \mathcal{D}_u^p} \det \left\{ Q + \sigma_i^p (1 - \sigma_i^p) x_i^p (x_i^p)^T \right\} \\
= \max_{x_i^p \in \mathcal{D}_u^p} \left\{ 1 + \sigma_i^p (1 - \sigma_i^p) (x_i^p)^T Q^{-1} x_i^p \right\}
\]

(19)
Outline

- What is a classification problem
- training data and test data are mismatched ...
- Migratory-Logit
- Fast learning of Migratory-Logit
- Active selection of labelled data
- A toy example
- An example on UC Irvine Data
Experimental Setup

Four classifiers are compared:

- Migratory-Logit trained on $D^a \cup D^p_l$
- Standard Logit trained on $D^a \cup D^p_l$
- Standard Logit trained on $D^p_l$
- Standard Logit trained on $D^a$

The $D^p_l$ are either randomly selected from $D^p$, or actively selected from $D^p$

For random $D^p_l$, the test error rates are an average from 50 independent trials
Experimental Setup

- The four classifiers are tested on $\mathcal{D}_u^p = \mathcal{D}^p \setminus \mathcal{D}_l^p$, using the rule: declare $y^p = -1$ if $\sigma(w^T x^p) \leq 0.5$ and $y^p = 1$ otherwise, for any $x^p \in \mathcal{D}_u^p$.

- $C = 6$ for Migratory-Logit
A toy Example

\( \mathcal{D}^p \) are two bivariate Gaussian distributions:
\[
\Pr(x^p | y^p = -1) = \mathcal{N}(x^p; \mu_0, \Sigma) \quad \text{and} \quad \Pr(x^p | y^p = +1) = \mathcal{N}(x^p; \mu_1, \Sigma),
\]
with \( \mu_0 = [0, 0]^T \), \( \mu_1 = [2.3, 2.3]^T \), and \( \Sigma = \begin{bmatrix} 1.75 & -0.433 \\ -0.433 & 1.25 \end{bmatrix} \).

\( \mathcal{D}^a \) are a selective draw from the two Gaussian distributions, with the selection probability
\[
\Pr(s | x^p, y^p = -1) = \sigma(w_0 + w_1 K(x^p, \mu_s^0; \Sigma)) \quad \text{and} \quad \Pr(s | x^p, y^p = +1) = \sigma(w_0 + w_1 K(x^p, \mu_s^1; \Sigma)),
\]
where \( \sigma \) is the sigmoid function, \( w_0 = -1 \), \( w_1 = \exp(1) \),
\[
K(x^p, \mu_s^0; \Sigma) = \exp\{-0.5(x^p - \mu_s^0)^T \Sigma^{-1}(x^p - \mu_s^0)\} \quad \text{with} \quad \mu_s^0 = [2, 1]^T,
\]
and
\[
K(x^p, \mu_s^1; \Sigma) = \exp\{-0.5(x^p - \mu_s^1)^T \Sigma^{-1}(x^p - \mu_s^1)\} \quad \text{with} \quad \mu_s^1 = [0, 3]^T.
\]

We obtain 150 samples of \( \mathcal{D}^p \) and 150 samples of \( \mathcal{D}^a \), which are shown in Figure 2.
A toy Example

The size of $x^a_i \in D^a$ is displayed in proportion to $\exp(-y^a_i \mu_i / 12)$

Figure 1: Illustration of active data selection by Migratory-Logit.
A toy Example

The size of $x_i^a \in D^a$ is displayed in proportion to $\exp(-y_i^a \mu_i/12)$

Figure 2: Illustration of active data selection by Migratory-Logit.
A toy Example

movie of data selection
A toy Example

Figure 3: Error rates of Migratory-Logit and logistic regression on the toy data, as a function of size of $D^p_l$. The primary labeled data $D^p_l$ are actively selected from $D^p$. 

A toy Example

Figure 4: Test error rates of Migratory-Logit and logistic regression on the toy data, as a function of size of $D_P^p$. The primary labeled data $D_P^p$ are randomly selected from $D_P$. The error rates are an average over 50 independent trials of random selection of $D_P^p$. 

Outline

- What is a classification problem
- training data and test data are mismatched ...
- Migratory-Logit
- Fast learning of Migratory-Logit
- Active selection of labelled data
- A toy example
- An example on UC Irvine Data
569 instances with feature dimensionality 30

We randomly partition the data set into 228 training data \( \mathcal{D}^a \) and 341 test data \( \mathcal{D}^p \).

We artificially make \( \mathcal{D}^a \) mismatched with \( \mathcal{D}^p \) by making changing to 50\% randomly chosen \((x^a_i, y^a_i) \in \mathcal{D}^a:\) change the signs of \( y^a_i \) and add 0 dB white Gaussian noise to \( x^a_i \).

The test errors are summarized in Figure 5 for actively selected \( \mathcal{D}^p_i \) and Figure 6 for randomly selected \( \mathcal{D}^p_i \).
Results on Wisconsin Breast Cancer Data

Figure 5: The $D^p_i$ are randomly selected from $D^p$. The error rates are an average over 50 independent trials of random selection of $D^p_i$. 

Results on Wisconsin Breast Cancer Data

Figure 6: The $\mathcal{D}_l^p$ are actively selected from $\mathcal{D}^p$. 

Using wdbc data from UCI

- M–Logit (C=6) trained on $\mathcal{D}^a +$ active $\mathcal{D}_l^p$
- Logistic regression trained on $\mathcal{D}^a +$ active $\mathcal{D}_l^p$
- Logistic regression trained on active $\mathcal{D}_l^p$
- Logistic regression trained on $\mathcal{D}^a$
Migratory-Logit with different $C$’s, using the Wisconsin Breast Cancer Databases of UCI. The primary labeled data $D^p_i$ are actively selected from $D^p$.

Figure 7: Comparison of Migratory-Logit with different $C$’s, using the Wisconsin Breast Cancer Databases of UCI. The primary labeled data $D^p_i$ are actively selected from $D^p$. 
Summary

- We have proposed Migratory-Logit to learn in the presence of mismatch between the training data $D^a$ and the testing data $D^p$.

- The basic idea is to introduce an auxiliary variable $\mu_i$ for each example $(x^a_i, y^a_i) \in D^a$.

- We have developed a fast learning algorithm to enhance the ability of M-Logit to handle large auxiliary data sets.

- The primary labeled data $D^p_l$ is actively selected to enhance adaptivity.

- The experimental results demonstrate that if the classifier trained on $D^a$ is to generalize well to $D^p$, the mismatch between $D^a$ and $D^p$ must be compensated.