Positive and Unlabeled Data

jojonki

Feb 2020

1 はじめに

このまとめは，Learning Classifiers on Positive and Unlabeled Data with Policy Gradient という論文で提案されている，Positive and Unlabeled Data の学習手法を勉強した際の jojonki のメモである．誤り箇所があれば Twitter 等で@jojonki までメンションを投げてもらえると助かる．

2 イントロ

まず Positive and Unlabeled な設定とは，事例 x に対して，2 値のラベル $y \in \{0, 1\}$ を予測する問題であるが，ラベルデータが正例（Positive），つまり $y = 1$ のみが振られており，更にすべての正例にラベルが振られていない設定である．ラベルが振られているかどうかは $s \in \{0, 1\}$ で表現する．この状況下での最終的な目的は $f(x) = p(y = 1|x)$ を求めることである．これを traditional な分類器と呼ぶ．しかしラベルが振られていないデータには正例と負例が含まれており，ラベルが振られていないからといって，すべてを負例にするわけにはいかない．これをうまいことベイズの定例で解いていくのが今回紹介する手法．この設定を nontraditional な学習と表現する．

3 大まかな流れ

まず目的の $f(x) = p(y = 1|x)$ を手に入れるまでの流れを先に整理しておく．これは後で見返したときに初めて意味がわかると思うので，今はわからなくて問題ない．

1. 学習データ (X, s) から，ラベルが振られてるかどうかの s を予測するモデル g を鍛える
2. 学習済み g を使って，dev セットにおいて $c = e_1 = \frac{1}{n} \sum_{x \in P} g(x) = \frac{1}{n} \sum_{x \in P} p(s = 1|x)$ を求める
3. ラベルなし事例が正例である場合と負例である場合の重み付を計算する
4. ラベルなし事例の重みを考慮して，ラベル y を予測するモデルを学習

4 アルゴリズム

まず最初に正例のみがラベル付けされているということは，負例がラベル付けされている確率は 0 である．

$$p(s = 1|x, y = 0) = 0 \tag{1}$$

また「ラベル付けされる正例は全正例から完全にランダムに選ばれる」という想定を置く（at random 想定）．これにより正例にラベルが振られている確率は，x に無関係となる．正例がラベル付けされている確率を c とすると下記のようになる..

$$c = p(s = 1|x, y = 1) = p(s = 1|y = 1) \tag{2}$$

ところで，事例に対してラベル付けされている確率 $g(x)$ は下記のように展開でき，重要な補題が導ける．下式は，求めたい式 $f(x)$ が $g(x)$ とその $g(x)$ から計算できる c から求められることを指す．

$$\begin{aligned} g(x) = p(s=1|x) &= p(y=1 \wedge s=1|x) \\ &\quad \text{ベイズの定理} \\ &= p(y=1|x)p(s=1|y=1,x) \\ &\quad \text{at random 想定} \\ &= p(y=1|x)p(s=1|y=1) \\ &= f(x)c \end{aligned} \tag{3}$$

よって traditional な分類器 $f(x)$ は，下記のように表せる．更にこれは確率である．

$$\begin{aligned} f(x) &= \frac{g(x)}{c} \\ &= \frac{\text{事例のラベル付けされている割合}}{\text{正例のラベル付けされている割合}} \\ &\leq 1 \end{aligned} \tag{4}$$

4.1 $c = e_1 = p(s=1|y=1)$ を求める

c は，正例に対してどの程度ラベルが振られているかの確率である．その c の求め方として論文では e_1, e_2, e_3 の 3 つの方法を示しているが，もっとも筋が良い？とされる e_1 を説明する．c は，学習済み g と dev セットから求めることができる．まず，学習データを利用して，$g(x) = p(s=1|x)$ を学習する（つまり学習データがラベル付けされているかを予測するモデル）．そして dev セットの $x \in P$(正例) における $g(x)$ の平均予測結果 $(\frac{1}{n}\sum_{x \in P} g(x))$ を $c = e_1$ と求めることができる．

この証明は下記のようになる．証明の条件として，$x \in P$ と「ラベル付けされる正例は全正例から完全にランダムに選ばれる」を使っている．

$$\begin{aligned} g(x) &= p(s=1|x) \\ &\quad y \text{ に対して周辺化} \\ &= p(s=1,y=1|x)p(s=1,y=0|x) \\ &= \frac{1}{p(x)}\{p(s=1,y=1,x) + p(s=1,y=0,x)\} \\ &= \frac{1}{p(x)}\{p(s=1|x,y=1)p(y=1|x)p(x) + p(s=1|x,y=0,x)p(y=0|x)p(x)\} \\ &= p(s=1|x,y=1)p(y=1|x) + p(s=1|x,y=0)p(y=0|x) \\ &\quad x \in P \text{ であるから} \\ &= p(s=1|x,y=1)1 + 0 \cdot 0 \\ &= p(s=1|y=1) \\ &= c \end{aligned} \tag{5}$$

4.2 ラベルなし事例への重み付け

任意の関数 h に対して，$p(x,y,s)$ の期待値 $E_p(x,y,s)[h(x,y)] = E[h]$ を考えてみる．

$$
\begin{aligned}
E[h] &= \int_{x,y,s} h(x,y)p(x,y,s) \\
&\quad \text{ベイズの定理} \\
&= \int_{x,y,s} h(x,y)p(y|x,s)p(s|x)p(x) \\
&= \int_x p(x) \sum_{s=0}^{1} p(s|x) \sum_{y=0}^{1} p(y|x,s)h(x,y) \\
&= \int_x p(x) \Big(p(s=1|x) \sum_{y=0}^{1} p(y|x,s=1)h(x,y) + p(s=0|x) \sum_{y=0}^{1} p(y|x,s=0)h(x,y) \Big) \\
&\quad s=1 \text{ であれば } p(y|x)=1 \text{ であるのでカッコ内の第 1 項は } y=1 \text{ のみ残る} \\
&= \int_x p(x) \Big(p(s=1|x)h(x,1) + p(s=0|x) \sum_{y=0}^{1} p(y|x,s=0)h(x,y) \Big) \\
&\quad \text{カッコ内の第 2 項を } y=0,1 \text{ で展開．ラベルなしデータにラベルがある時とない時の重み付け和となる} \\
&= \int_x p(x) \Big(p(s=1|x)h(x,1) + p(s=0|x) \left[p(y=1|x,s=0)h(x,1) + p(y=0|x,s=0)h(x,0) \right] \Big)
\end{aligned}
\tag{6}
$$

よって，ラベルなしデータが正例である確率（重み）を $w(x) = p(y=1|x,s=0) = \frac{1-c}{c} \frac{p(s=1|x)}{1-p(s=1|x)}$ と置くと (あとで証明)，

$$
E[h] = \frac{1}{m} \left(\sum_{x,s=1} h(x,1) + \sum_{x,s=0} \left[w(x)h(x,1) + (1-w(x))h(x,0) \right] \right) \tag{7}
$$

と表せた．m は学習データセットのサイズである．これによりラベルが振られていない事例に対して，正例，負例の場合を重み付けでサンプリングして学習できる．

ちなみに $w(x)$ は下記のように証明できる．

$$
\begin{aligned}
w(x) = p(y=1|x,s=0) &= \frac{p(s=0|x,y=1)p(y=1|x)}{p(s=0|x)} \\
&= \frac{[1-p(s=1|x,y=1)]]p(y=1|x)}{1-p(s=1|x)} \\
&= \frac{(1-c)p(y=1|x)}{1-p(s=1|x)} \\
&\quad \text{f(x) は g(x) と c で表せるので} \\
&= \frac{(1-c)\frac{p(s=1|x)}{c}}{1-p(s=1|x)} \\
&= \frac{1-1}{c} \frac{p(s=1|x)}{1-p(s=1|x)}
\end{aligned}
\tag{8}
$$

5 Python で書いてみる

2 つの異なるガウス分布から擬似データを算出し，一部を意図的にマスクした．この実験コードは，`https://github.com/jojonki/pu-learning` を参照してほしい．

正例（青色）の一部しかラベル付けされていないのにも関わらず（右上），それなりの精度の境界面を引けていることがわかる（右下）．

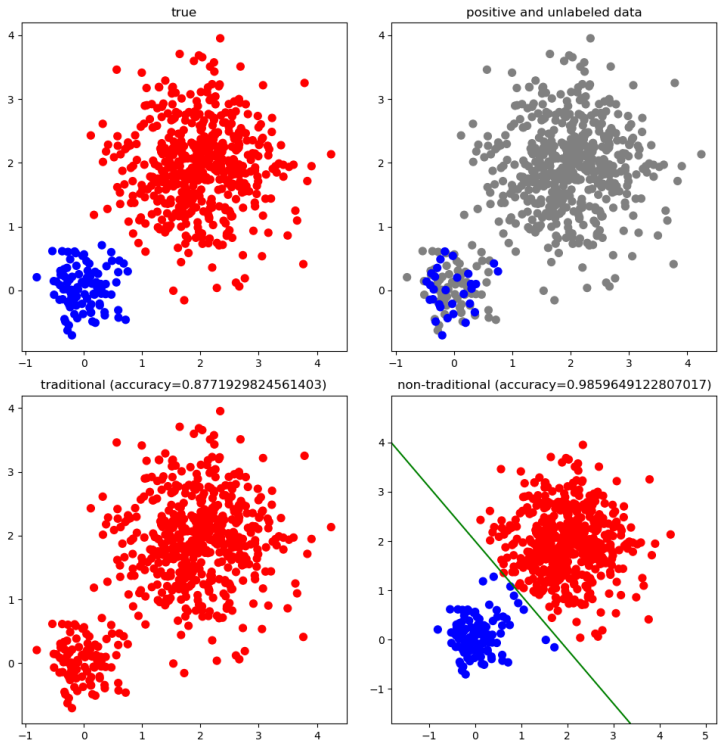

Figure 1: 実験結果

6 まとめ

正直かなり端折って書いているため，論文の情報を読み解けているかというと実は NO なのだが，よくある設定状況下においても基本的にベイズの定理を展開するだけで，このような事例に対処できるのは非常に興味深い．

7 参考情報

- 正例とラベル無しデータからの学習 (PU classification)
  https://www.pillyshi.net/2015/10/29/%E6%AD%A3%E4%BE%8B%E3%81%A8%E3%83%A9%E3%83%99%E3%83%AB%E7%84%A1%E3%81%97%E3%83%87%E3%83%BC%E3%82%BF%E3%81%8B%E3%82%89%E3%81%AE%E5%AD%A6%E7%BF%92-pu-classification/