Методы частичного обучения (semi-supervised learning)

К. В. Воронцов
vokov@forecsys.ru

Этот курс доступен на странице вики-ресурса
http://www.MachineLearning.ru/wiki
«Машинное обучение (курс лекций, К.В.Воронцов)»

8 мая 2012

Постановка задачи частичного обучения

Дано:

множество объектов X, множество классов Y;

$X^\ell = \{x_1, \ldots, x_\ell\}$ — размеченная выборка (labeled data);
$\{y_1, \ldots, y_\ell\}$

$X^k = \{x_{\ell+1}, \ldots, x_{\ell+k}\}$ — неразмеченная выборка (unlabeled data).

Два варианта постановки задачи:

- *Частичное обучение* (semi-supervised learning):
  построить алгоритм классификации $a\colon X \to Y$.
- *Трансдуктивное обучение* (transductive learning):
  зная все $\{x_{\ell+1}, \ldots, x_{\ell+k}\}$, получить метки $\{y_{\ell+1}, \ldots, y_{\ell+k}\}$.

Типичные приложения:

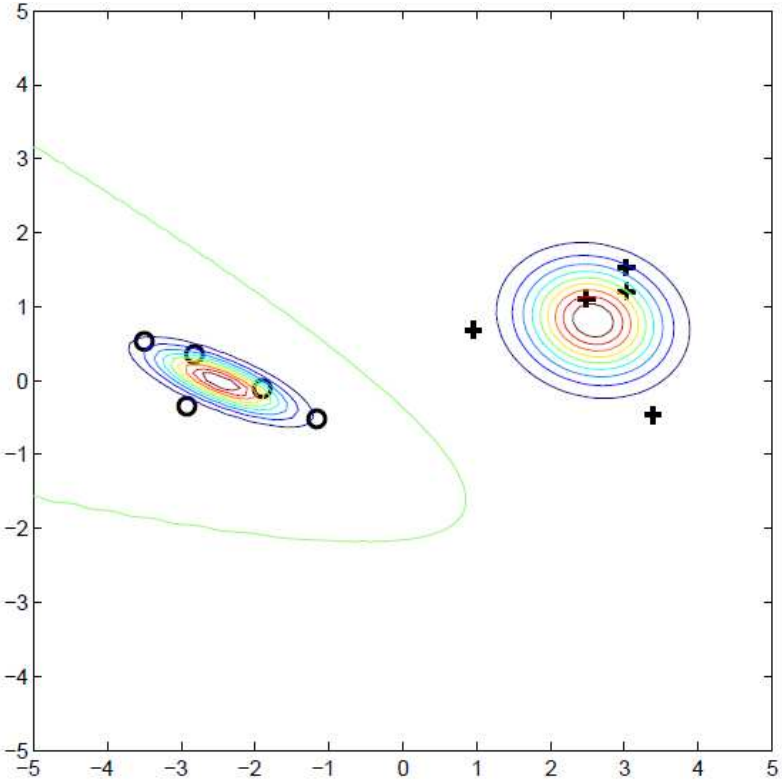
классификация и каталогизация текстов, изображений, и т. п.

Содержание: методы кластеризации

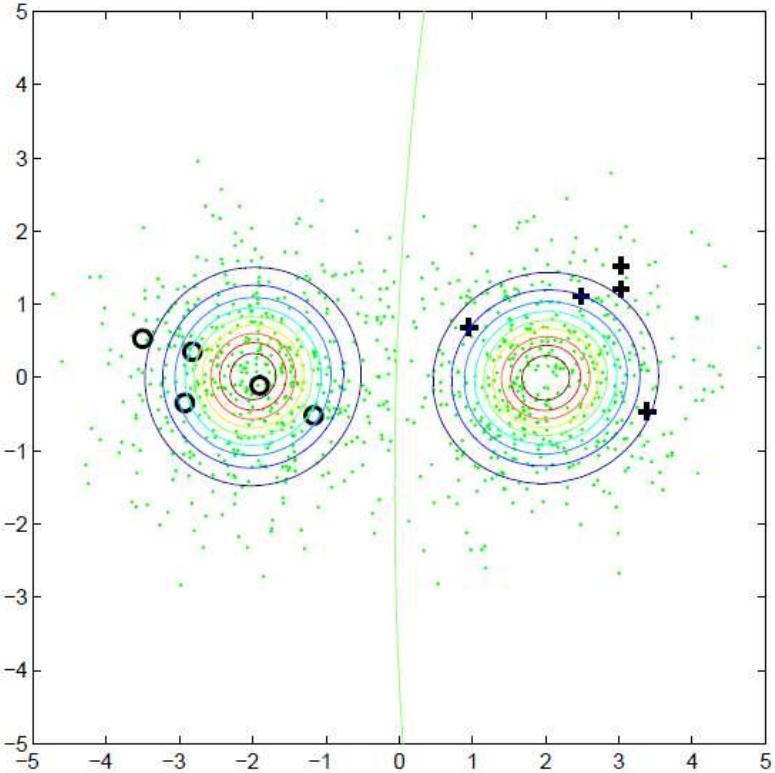
SSL не сводится к классификации

Пример 1. плотности классов, восстановленные:

по размеченным данным X^ℓ

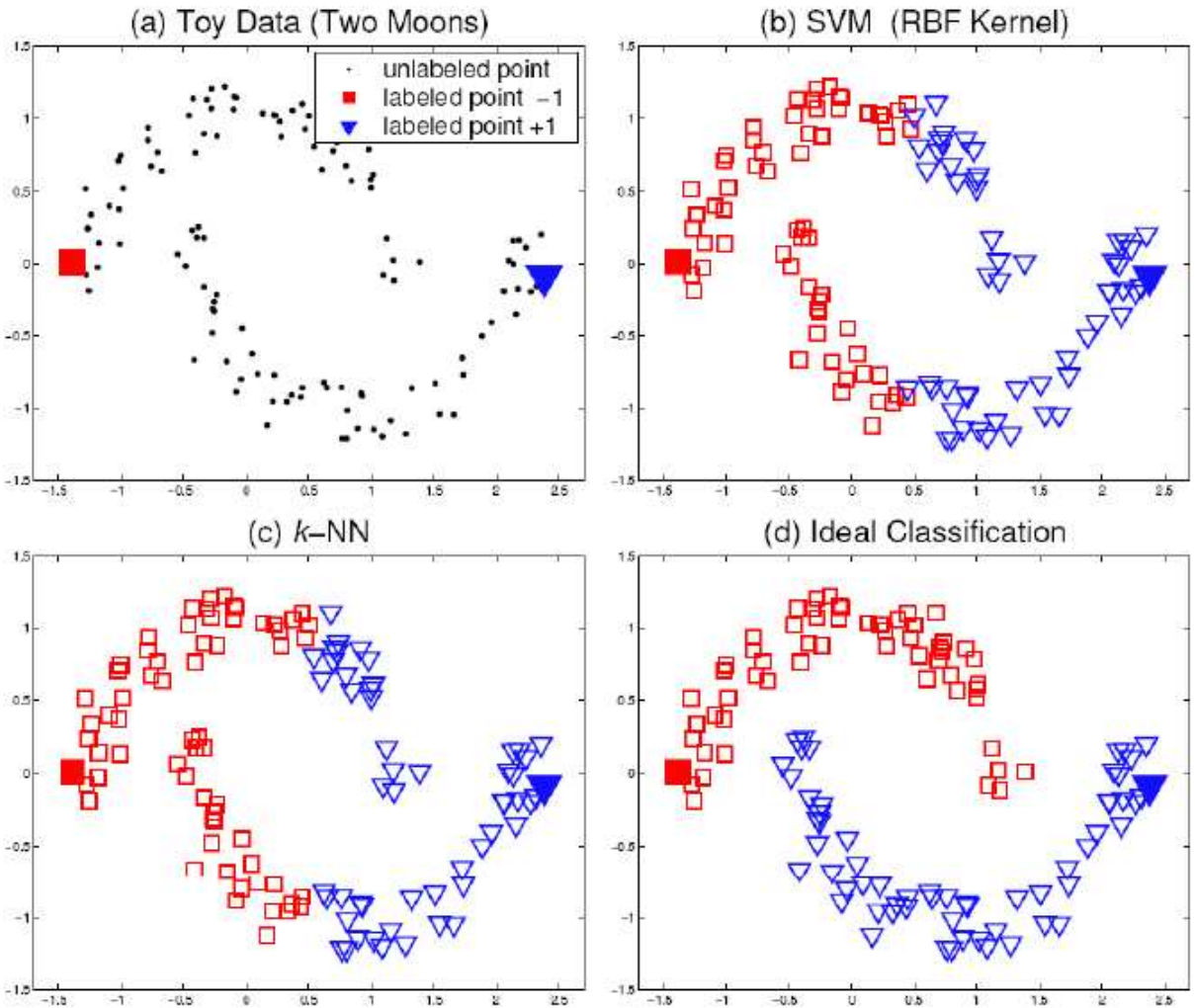


по полным данным $X^{\ell+k}$



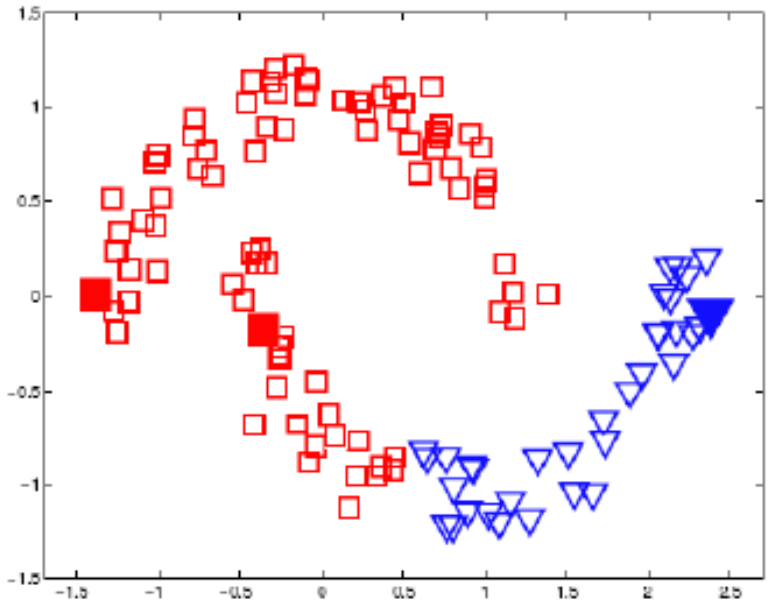
SSL не сводится к классификации

Пример 2.



Однако и к кластеризации SSL также не сводится

Пример 3.



Метод self-training (1965-1970)

Пусть $\mu\colon X^\ell \to a$ — произвольный метод обучения;
классификаторы имеют вид $a(x) = \arg\max\limits_{y\in Y} \Gamma_y(x)$;

Отступ объекта — степень доверия классификации $a(x_i)$:

$$M_i(a) = \Gamma_{y_i}(x_i) - \max_{y\in Y\setminus y_i} \Gamma_y(x_i).$$

Алгоритм self-training — обёртка (wrapper)
над произвольным методом обучения классификатора:

1: $Z := X^\ell$;
2: **пока** $|Z| < \ell + k$
3: $\quad a := \mu(Z)$;
4: $\quad \Delta := \left\{x_i \in X^k\setminus Z \;\middle|\; M_i(a) \geqslant M_0\right\}$;
5: $\quad y_i := a(x_i)$ для всех $x_i \in \Delta$;
6: $\quad Z := Z \cup \Delta$;

Метод co-training (Blum, Mitchell, 1998)

Пусть $\mu_1 \colon X^\ell \to a_1$, $\mu_2 \colon X^\ell \to a_2$ — два существенно различных метода обучения, использующих
— либо разные наборы признаков;
— либо разные парадигмы обучения (inductive bias);
— либо разные источники данных $X_1^{\ell_1}$, $X_2^{\ell_2}$.

1: $Z_1 := X_1^{\ell_1}$; $Z_2 := X_2^{\ell_2}$;
2: **пока** $|Z_1 \cup Z_2| < \ell + k$
3: $\quad a_1 := \mu_1(Z_1)$; $\Delta_1 := \left\{x_i \in X^k \setminus Z_1 \setminus Z_2 \;\middle|\; M_i(a_1) \geqslant M_{01}\right\}$;
4: $\quad y_i := a(x_i)$ для всех $x_i \in \Delta_1$;
5: $\quad Z_2 := Z_2 \cup \Delta_1$;
6: $\quad a_2 := \mu_2(Z_2)$; $\Delta_2 := \left\{x_i \in X^k \setminus Z_1 \setminus Z_2 \;\middle|\; M_i(a_2) \geqslant M_{02}\right\}$;
7: $\quad y_i := a(x_i)$ для всех $x_i \in \Delta_2$;
8: $\quad Z_1 := Z_1 \cup \Delta_2$;

Метод co-learning (deSa, 1993)

Пусть $\mu_t \colon X^\ell \to a_t$ — разные методы обучения, $t = 1, \ldots, T$.

Алгоритм co-learning — это self-training для композиции — простого голосования базовых алгоритмов $a_1, \ldots, a_T$:

$$a(x) = \arg\max_{y \in Y} \Gamma_y(x), \quad \Gamma_y(x_i) = \sum_{t=1}^{T} \big[a_t(x_i) = y\big].$$

тогда $M_i(a)$ — перевес голосов в пользу правильного класса y_i.

1: $Z := X^\ell$;
2: **пока** $|Z| < \ell + k$
3: $\quad a := \mu(Z)$;
4: $\quad \Delta := \big\{x_i \in X^k \backslash Z \;\big|\; M_i(a) \geqslant M_0\big\}$;
5: $\quad y_i := a(x_i)$ для всех $x_i \in \Delta$;
6: $\quad Z := Z \cup \Delta$;

Кластеризация как задача дискретной оптимизации

Пусть $\rho(x, x')$ — *функция расстояния* между объектами.
Веса на парах объектов (близости): $w_{ij} = \exp(-\beta\rho(x_i, x_j))$,
где β — параметр.

Задача кластеризации:

$$\sum_{i=1}^{\ell+k}\sum_{j=i+1}^{\ell+k} w_{ij}\left[a_i \neq a_j\right] \to \min_{\{a_i \in Y\}}.$$

Задача частичного обучения:

$$\sum_{i=1}^{\ell+k}\sum_{j=i+1}^{\ell+k} w_{ij}\left[a_i \neq a_j\right] + \lambda\sum_{i=1}^{\ell}\left[a_i \neq y_i\right] \to \min_{\{a_i \in Y\}}.$$

где λ — ещё один параметр.

Алгоритм КНП: кластеризация

Графовый алгоритм КНП (кратчайший незамкнутый путь)

1: Найти пару вершин $(x_i, x_j) \in X^{\ell+k}$ с наименьшим $\rho(x_i, y_j)$ и соединить их ребром;
2: **пока** в выборке остаются изолированные точки
3: найти изолированную точку, ближайшую к некоторой неизолированной;
4: соединить эти две точки ребром;
5: удалить $K - 1$ самых длинных рёбер;

Задача частичного обучения: земенить только шаг 5...

Алгоритм КНП: частичное обучение

Графовый алгоритм КНП (кратчайший незамкнутый путь)

1: Найти пару вершин $(x_i, x_j) \in X^{\ell+k}$ с наименьшим $\rho(x_i, y_j)$ и соединить их ребром;
2: **пока** в выборке остаются изолированные точки
3: найти изолированную точку, ближайшую к некоторой неизолированной;
4: соединить эти две точки ребром;
5: ~~удалить $K-1$ самых длинных рёбер;~~
6: **пока** есть путь между двумя вершинами разных классов
7: удалить самое длинное ребро на этом пути.

Алгоритм Ланса-Уильямса: кластеризация

Алгоритм иерархической кластеризации (Ланс, Уильямс, 1967)

1: $C_1 := \{\{x_1\}, \ldots, \{x_{\ell+k}\}\}$ — все кластеры 1-элементные;
$R_{\{x_i\}\{x_j\}} := \rho(x_i, x_j)$ — расстояния между ними;
2: **для всех** $t = 2, \ldots, \ell + k$ (t — номер итерации):
3: найти в C_{t-1} пару кластеров (U, V) с минимальным R_{UV};
4: слить их в один кластер:
$W := U \cup V$;
$C_t := C_{t-1} \cup \{W\} \setminus \{U, V\}$;
5: **для всех** $S \in C_t$
6: вычислить R_{WS} по формуле Ланса-Уильямса:
$R_{WS} := \alpha_U R_{US} + \alpha_V R_{VS} + \beta R_{UV} + \gamma \left| R_{US} - R_{VS} \right|$;

Алгоритм Ланса-Уильямса: частичное обучение

Алгоритм иерархической кластеризации (Ланс, Уильямс, 1967)

1: $C_1 := \{\{x_1\}, \ldots, \{x_{\ell+k}\}\}$ — все кластеры 1-элементные;
$R_{\{x_i\}\{x_j\}} := \rho(x_i, x_j)$ — расстояния между ними;
2: **для всех** $t = 2, \ldots, \ell + k$ (t — номер итерации):
3: найти в C_{t-1} пару кластеров (U, V) с минимальным R_{UV},
при условии, что в $U \cup V$ нет объектов с разными метками;
4: слить их в один кластер:
$W := U \cup V$;
$C_t := C_{t-1} \cup \{W\} \setminus \{U, V\}$;
5: **для всех** $S \in C_t$
6: вычислить R_{WS} по формуле Ланса-Уильямса:
$R_{WS} := \alpha_U R_{US} + \alpha_V R_{VS} + \beta R_{UV} + \gamma \left| R_{US} - R_{VS} \right|$;

Метод k-средних: кластеризация

1: начальное приближение центров μ_y, $y \in Y$;
2: **повторять**
3: **E-шаг:**
отнести каждый x_i к ближайшему центру:
$y_i := \arg\min_{y \in Y} \rho(x_i, \mu_y)$, $i = 1, \ldots, \ell + k$;
4: **M-шаг:**
вычислить новые положения центров:

$$\mu_y := \frac{\sum_{i=1}^{\ell+k} [y_i = y]\, x_i}{\sum_{i=1}^{\ell+k} [y_i = y]}, \quad \text{для всех } y \in Y;$$

5: **пока** y_i не перестанут изменяться;

Метод k-средних: частичное обучение

1: начальное приближение центров μ_y, $y \in Y$;
2: **повторять**
3: **E-шаг:**
отнести каждый $x_i \in X^k$ к ближайшему центру:
$y_i := \arg\min_{y \in Y} \rho(x_i, \mu_y)$, $i = \ell+1, \ldots, \ell+k$;
4: **M-шаг:**
вычислить новые положения центров:
$$\mu_y := \frac{\sum_{i=1}^{\ell+k} [y_i = y]\, x_i}{\sum_{i=1}^{\ell+k} [y_i = y]}, \text{ для всех } y \in Y;$$
5: **пока** y_i не перестанут изменяться;

SVM: классификация

Линейный классификатор на два класса $Y = \{-1, 1\}$:

$$a(x) = \text{sign}\big(\langle w, x\rangle - w_0\big), \quad w, x \in \mathbb{R}^n,\ w_0 \in \mathbb{R}.$$

Отступ объекта x_i:

$$M_i(w, w_0) = \big(\langle w, x_i\rangle - w_0\big) y_i.$$

Задача обучения весов w, w_0 по размеченной выборке:

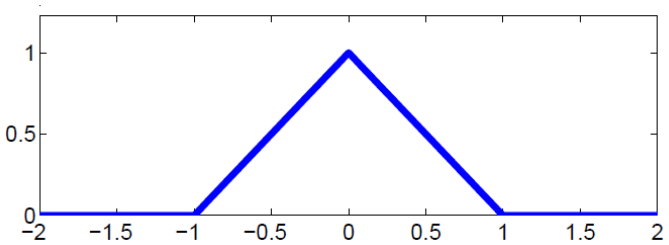
$$Q(w, w_0) = \sum_{i=1}^{\ell} \big(1 - M_i(w, w_0)\big)_+ + \frac{1}{2C}\|w\|^2 \ \to \ \min_{w, w_0}.$$

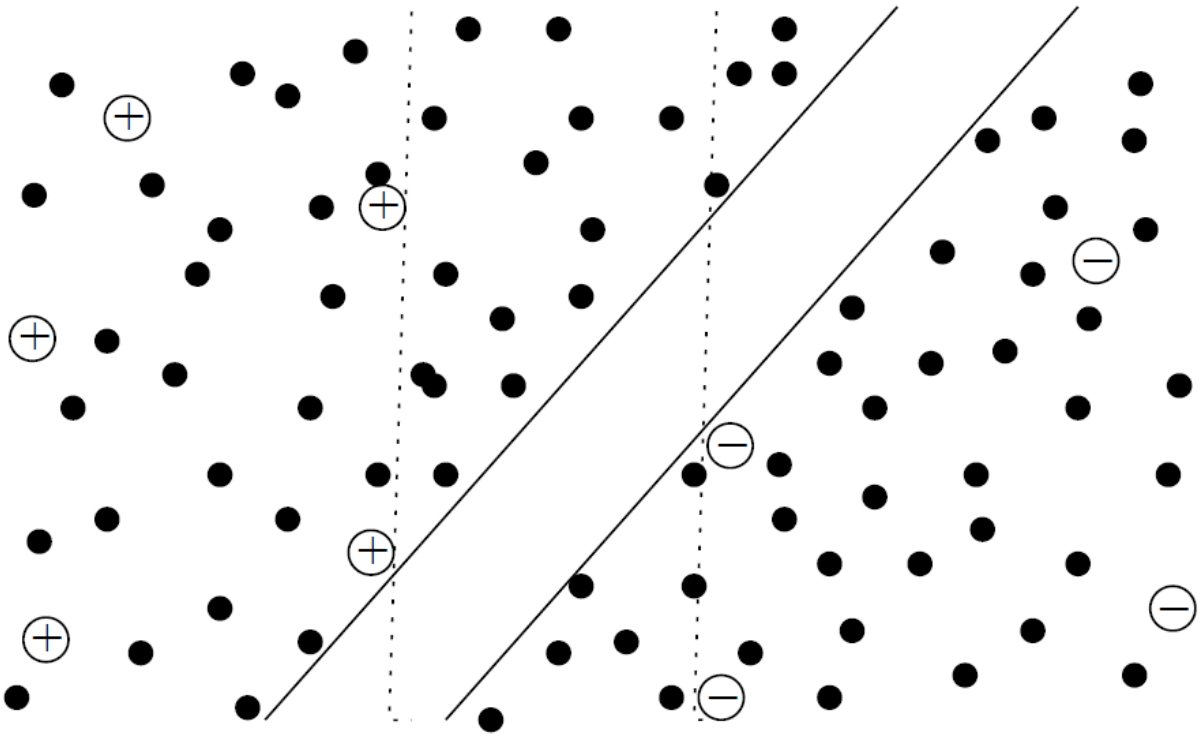
Функция $\mathscr{L}(M) = (1 - M)_+$ штрафует за уменьшение отступа.

Идея!
Функция $\mathscr{L}(M) = \big(1 - |M|\big)_+$ штрафует за попадание объекта внутрь разделяющей полосы.

Функция потерь для трансдуктивного SVM

Функция потерь $\mathscr{L}(M) = \big(1 - |M|\big)_+$ штрафует за попадание объекта внутрь разделяющей полосы.





Transductive SVM: частичное обучение

Обучение весов w, w_0 по частично размеченной выборке:

$$Q(w, w_0) = \sum_{i=1}^{\ell} \big(1 - M_i(w, w_0)\big)_+ + \frac{1}{2C}\|w\|^2 +$$

$$+ \gamma \sum_{i=\ell+1}^{\ell+k} \big(1 - |M_i(w, w_0)|\big)_+ \;\to\; \min_{w, w_0}.$$

Эффективная реализация:

Sindhwani, Keerthi. Large scale semisupervised linear SVMs. SIGIR 2006.

Гауссовская функция штрафа:

Chapelle, Zien. Semi-supervised classification by low density separation. AISTAT 2005.

Недостатки TSVM:

- решение неустойчиво, если нет области разреженности;
- требуется настройка двух параметров C, γ;

Логистическая регрессия: классификация на 2 класса

Линейный классификатор на два класса $Y = \{-1, 1\}$:

$$a(x) = \text{sign}\langle w, x\rangle, \quad x, w \in \mathbb{R}^n.$$

Вероятность того, что объект x_i относится к классу y:

$$P(y|x_i, w) = \frac{1}{1 + \exp(-\langle w, x_i\rangle y)}.$$

Задача максимизации регуляризованного правдоподобия:

$$Q(w) = \sum_{i=1}^{\ell} \log P(y_i|x_i, w) - \frac{1}{2C}\|w\|^2 \;\to\; \max_w,$$

Логистическая регрессия: классификация при произвольном Y

Линейный классификатор при произвольном числе классов $|Y|$:

$$a(x) = \arg\max_{y \in Y} \langle w_y, x \rangle, \quad x, w_y \in \mathbb{R}^n, \quad w \equiv (w_y)_{y \in Y}.$$

Вероятность того, что объект x_i относится к классу y:

$$P(y|x_i, w) = \frac{\exp\langle w_y, x_i \rangle}{\sum_{c \in Y} \exp\langle w_c, x_i \rangle}.$$

Задача максимизации регуляризованного правдоподобия:

$$Q(w) = \sum_{i=1}^{\ell} \log P(y_i|x_i, w) - \frac{1}{2C}\|w\|^2 \to \max_w,$$

Логистическая регрессия: частичное обучение

Теперь учтём неразмеченные данные $X^k = \{x_{\ell+1}, \ldots, x_{\ell+k}\}$.
Пусть $b_j(x)$ — бинарные признаки, $j = 1, \ldots, m$.

Оценим вероятности $\mathrm{P}(y|b_j(x) = 1)$ двумя способами:

1) эмпирическая оценка по размеченным данным X^ℓ:

$$p_j(y) = \frac{\sum\limits_{i=1}^{\ell} b_j(x_i)\big[y_i = y\big]}{\sum\limits_{i=1}^{\ell} b_j(x_i)};$$

2) оценка по неразмеченным данным X^k и линейной модели w:

$$p_j(y, w) = \frac{\sum\limits_{i=\ell+1}^{\ell+k} b_j(x_i) P(y|x_i, w)}{\sum\limits_{i=\ell+1}^{\ell+k} b_j(x_i)}.$$

Построение функционала качества

Кросс-энтропия — мера согласованности двух оценок, $p_j(y)$ и $p_j(y,w)$ одной и той же вероятности $\mathrm{P}(y|b_j(x)=1)$:

$$H_j(w) = \sum_{y\in Y} p_j(y)\log p_j(y,w) \to \max_w$$

(максимум достигается при $p_j(y) \equiv p_j(y,w)$).

Добавим суммарную согласованность по всем m признакам к функционалу регуляризованного правдоподобия:

$$Q(w) = \sum_{i=1}^{\ell} \log P(y_i|x_i,w) - \frac{1}{2C}\sum_{y\in Y}\|w_y\|^2 + \\ + \lambda\sum_{j=1}^{m}\sum_{y\in Y} p_j(y)\log\left(\frac{\sum_{i=\ell+1}^{\ell+k} b_j(x_i)P(y|x_i,w)}{\sum_{i=\ell+1}^{\ell+k} b_j(x_i)}\right) \to \max_w,$$

Замечания про метод XR (Expectation Regularization)

1. Оптимизация $Q(w)$ — методом стохастического градиента.
2. Возможные варианты задания переменных b_j:
   1. $b_j(x) \equiv 1$, тогда $\mathrm{P}(y|b_j(x)=1)$ — априорная вероятность класса y (label regularization) — хорошо подходит для задач с несбалансированными классами;
   2. $b_j(x) = \big[\text{термин } j \text{ содержится в тексте } x\big]$ — для задач классификации и каталогизации текстов.
3. XR слабо чувствителен к выбору C и γ.
4. XR очень устойчив к погрешностям оценивания $p_j(y)$.
5. XR не требователен к числу размеченных объектов ℓ.
6. XR хорошо подходит для категоризации текстов.
7. XR показывает в экспериментах очень высокую точность.

Mann, McCallum. Simple, robust, scalable semi-supervised learning via expectation regularization. ICML 2007.

Резюме в конце лекции

- Задача SSL занимает промежуточное положение между классификацией и кластеризацией, но не сводится к ним.
- Простые методы-обёртки требуют многократного обучения, что вычислительно неэффективно.
- Методы кластеризации легко адаптируются к SSL путём введения ограничений (constrained clustering), но, как правило, вычислительно трудоёмки.
- Адаптация методов классификации реализуется сложнее, но приводит к более эффективным методам.
- Expectation Regularization — быстрый и точный метод, позволяющий учитывать дополнительную информацию.