

Nikita Volkov

13 April 2017

1.

$$a(x) = \begin{cases} 1 & f(x) > 0 \\ -1 & f(x) < 0 \end{cases}$$

2.

$$M_i = y_i * f(x_i)$$

Если $M_i > 0$, то классификатор сработал правильно на объекте x_i , если же $M_i \leq 0$, то неправильно.

3. К векторам x и w добавим фиктивные первые переменные $x_1 = 1$ и $w_1 = w_0$, обозначим расширенные векторы за \tilde{x} и \tilde{w} соответственно. Тогда:

$$\forall w_0 : (w, x) + w_0 = (\tilde{w}, \tilde{x})$$

4.

$$Q(w) = \sum_{i=1}^l [M_i \leq 0]$$

Наилучший алгоритм не ошибается, поэтому для него $Q(w) = 0$.

5. Положим w нулевым вектором, тогда $(w, x_i) = 0$ для любого объекта x_i , и $M_i = 0$. Значит, $Q(w) = 0$.

6.

$$Q(w) = \sum_{i=1}^l L(M_i)$$

7. Функция потерь $L(y_i, a(x_i))$ характеризует ошибку при неправильном ответе алгоритма a . Обычно для линейной классификации функция потерь неотрицательна и невозрастает.

8.

$$\max(0, 1 - M(w))$$

9. Регуляризатор штрафует за большие веса для признаков в линейной модели. Обычно используют l_1 — регуляризатор:

$$\gamma * \sum_{i=1}^l |w_i|,$$

или l_2 — регуляризатор:

$$\gamma * \sum_{i=1}^l |w_i|^2,$$

где γ — коэффициент регуляризации.

10. Если обобщающая способность алгоритма высока, то есть мы знаем, что ошибка на тестовой выборке не сильно отличается от ошибки на трейне, то переобучение отсутствует. Регуляризация помогает не переобучиться на конкретные признаки из тренировочной выборки, тем самым уменьшая возможность переобучения.
- 11.
- 12.
13. Для построенного с регуляризацией, так как регуляризация не позволяет переобучиться на конкретные параметры, минимизирующие функцию риска.
14. Скорее всего для построенного без регуляризации, так как алгоритм может переобучиться на тренировочной выборке и плохо сработать на тестовой.
15. Будем использовать обозначения TP, TN, FP, FN, где первая буква T обозначает, как классифицировали, T — правильно, F — неправильно, а вторая, к какому классу отнесли. Тогда:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

16. Метрика качества AUC — это площадь под ROC-кривой. Каждая точка на ROC-кривой соответствует некоторому алгоритму, x -ая координата показывает долю неправильных положительных классификаций — FPR, y -ая координата показывает долю правильных положительных классификаций — TPR. Положение точки зависит от выбранного в алгоритме порога w_0 для решающего правила.
17. Заметим, что для построения ROC-кривой не надо перебирать все значения w_0 , достаточно рассмотреть значения $f(x) = (x, w)$, для x из выборки. На этой идее опишем алгоритм построения ROC-кривой:

```

1:  $l_- = \sum_{i=1}^l [y_i = -1]$  {количество эл-ов класса -1}
2:  $l_+ = \sum_{i=1}^l [y_i = +1]$  {количество эл-ов класса +1}
3: Упорядочить выборку по значениям  $f(x, w)$ 
4:  $(FPR_0, TPR_0) = (0, 0)$ 
5: for  $i = 1, \dots, l$  do
6:   if  $y_i == -1$  then
        $(FPR_i, TPR_i) = (FPR_{i-1} + \frac{1}{l_-}, TPR_{i-1})$ 
7:   else
        $(FPR_i, TPR_i) = (FPR_{i-1}, TPR_{i-1} + \frac{1}{l_+})$ 

```