

# Klasyfikacja za pomocą regresji logistycznej

Techniki Optymalizacji

13.11.2018

## 1 Problem klasyfikacji

Przewidywania/wyjaśnienie zmian jednej zmiennej *dyskretnej* ( $Y$ ) pod wpływem zmian innych zmiennych ( $\mathbf{X} = (X_1, \dots, X_m)$ ). Zwykle  $Y$  przyjmuje tylko kilka możliwych wartości, które nazywa się *klasami*, *etykietami* lub *kategoriami*. Często  $Y \in \{-1, +1\}$ , mamy wtedy do czynienia z *klasyfikacją binarną*. Zajmiemy się tylko tą ostatnią, ponieważ każdy problem z więcej niż dwoma klasami da się sprowadzić do klasyfikacji binarnej (np. metodą „jeden przeciwko wszystkim”). Przykłady:

- $\mathbf{X}$  – wyniki testów medycznych,  $Y \in \{\text{chory}, \text{zdrowy}\}$ .
- $\mathbf{X}$  – jasność pikseli w obrazie  $Y \in \{0, \dots, 9\}$  – cyfra na obrazie.
- $\mathbf{X}$  – słowa w dokumencie tekstowym,  $Y \in \{\text{spam}, \text{niesпам}\}$ .
- $\mathbf{X}$  – treść strony internetowej + słowa kluczowe zapytania,  $Y \in \{\text{odwiedzona}, \text{nieodwiedzona}\}$  – czy strona została(by) odwiedzona po pokazaniu w wyszukiwarce.

Dzisiaj zajmiemy się *klasyfikacją liniową*. Kodując klasy jako  $\{-1, +1\}$ , klasyfikator liniowy składa się z dwóch części:

- Funkcji liniowej od  $\mathbf{X}$ :

$$f(\mathbf{X}) = w_0 + \sum_{j=1}^m w_j X_j = \mathbf{w}^\top \mathbf{X}$$

- Klasyfikacji poprzez progowanie w zerze:

$$\hat{Y}(\mathbf{X}) = \begin{cases} +1 & \text{jeśli } f(\mathbf{X}) \geq 0 \\ -1 & \text{jeśli } f(\mathbf{X}) < 0 \end{cases} = \text{sgn}(f(\mathbf{X})).$$

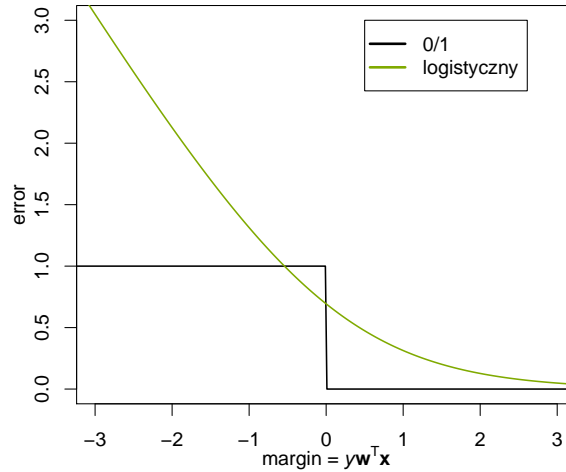
Naturalnym błędem klasyfikacji jest błąd 0/1 („zero-jedynkowy”), zapisany jako:

$$\ell(Y, \hat{Y}) = \mathbb{1}[Y \neq \hat{Y}],$$

gdzie  $\mathbb{1}[C]$  to *funkcja indykatorowa*:  $\mathbb{1}[C] = 1$  : 0. Chcielibyśmy więc wytrenować klasyfikator minimalizując jego błąd na zbiorze uczącym:

$$L(\mathbf{w}) = \sum_{i=1}^n \mathbb{1}[y_i \neq \hat{y}_i] = \sum_{i=1}^n \mathbb{1}[y_i \mathbf{w}_i^\top \mathbf{x}_i < 0].$$

Niestety, jest to problem NP-trudny, stąd przybliżamy błąd 0/1 innym typem błędu. Dzisiaj zajmiemy się błędem logistycznym.



Rysunek 1: Błąd logistyczny w porównaniu do błęd 0/1 jako funkcja *marginu*  $Y\mathbf{w}^\top \mathbf{X}$

## 2 Regresja logistyczna

*Uwaga:* regresja logistyczna jest procedurą klasyfikacji! Nazwa „regresja” ma tu tylko znaczenie historyczne. Zastępujemy błąd 0/1 błędem logistycznym:

$$\ell(Y, \mathbf{w}) = \log(1 + e^{-Y\mathbf{w}^\top \mathbf{X}}).$$

Wyrażenie  $Y\mathbf{w}^\top \mathbf{X}$  nazywamy często *marginem*.

Naszą nową funkcją do minimalizacji jest więc sumaryczny błąd logistyczny:

$$L(\mathbf{w}) = \sum_{i=1}^n \log(1 + e^{-y_i \mathbf{w}^\top \mathbf{x}_i}).$$

W celu uniknięcia osobliwości hesjanu dodaje się, podobnie jak w przypadku regresji liniowej, dodatkowy człon z *regularyzacją*:

$$L(\mathbf{w}) = \sum_{i=1}^n \log(1 + e^{-y_i \mathbf{w}^\top \mathbf{x}_i}) + \frac{1}{2} \lambda \|\mathbf{w}\|^2.$$

W celu minimalizacji  $L(\mathbf{w})$ , wyznaczamy gradient i hesjan:

$$\nabla L(\mathbf{w}) = \lambda \mathbf{w} - \sum_{i=1}^n y_i \mathbf{x}_i \beta_i, \quad \mathbf{H}(\mathbf{w}) = \lambda \mathbf{I} + \sum_{i=1}^n \beta_i (1 - \beta_i) \mathbf{x}_i \mathbf{x}_i^\top,$$

gdzie:

$$\beta_i = \frac{1}{1 + e^{y_i \mathbf{w}^\top \mathbf{x}_i}}.$$

Dokładne wyprowadzenie znajduje się na slajdach z wykładu. Metoda Newtona-Raphsona prowadzi do następującego algorytmu:

1. Zaczynamy od  $\mathbf{w}_0 = \mathbf{0}$ ,
2. W kolejnych iteracjach  $t = 1, 2, \dots$  aż do zbieżności:
  - (a) Wyznaczamy współczynniki  $\beta_i = \frac{1}{1 + e^{y_i \mathbf{w}_{t-1}^\top \mathbf{x}_i}}$ .
  - (b) Wyznaczamy gradient  $\nabla L(\mathbf{w}_{t-1}) = \lambda \mathbf{w}_{t-1} - \sum_{i=1}^n y_i \mathbf{x}_i \beta_i$ .

- (c) Wyznaczamy hesjan  $\mathbf{H}(\mathbf{w}_{t-1}) = \lambda \mathbf{I} + \sum_{i=1}^n \beta_i (1 - \beta_i) \mathbf{x}_i \mathbf{x}_i^\top$ .
- (d) Robimy krok Newtona-Rapshona uaktualniając wektor wag:

$$\mathbf{w}_t := \mathbf{w}_{t-1} - \mathbf{H}(\mathbf{w}_{t-1})^{-1} \nabla L(\mathbf{w}_{t-1}).$$

Należy zauważyć, że problem bardzo przypomina problem regresji liniowej, z wyjątkiem tego, że pojawiają się współczynniki  $\beta_i$ , oraz tego, że optymalizację trzeba powtórzyć wielokrotnie (w przypadku regresji liniowej wystarczyła jedna iteracja).

Jak wiele iteracji robimy? Sensowne, a zarazem łatwe do implementacji kryterium stopu to wyznaczanie długości (normy) gradientu  $\|\nabla L(\mathbf{w}_{t-1})\|$  (funkcja `norm` w Octave) i zatrzymanie pętli, gdy będzie ona bardzo bliska zeru, np. poniżej  $\epsilon = 10^{-6}$ .

### 3 Uwagi o implementacji

Kolejne zadania laboratorium będą wykorzystywały napisaną przez Was implementację metody regresji logistycznej. Przy implementacji należy zwrócić uwagę na następujące rzeczy:

- Należy napisać program tak, aby przyjmował na wejściu dowolny zbiór danych w określonym formacie (można użyć skryptu do wczytywania danych z poprzednich zajęć).
- Tak jak w przypadku regresji liniowej, naturalne jest wczytanie danych jako macierz  $\mathbf{X}$  o rozmiarze  $n \times m$ , której poszczególne wiersze to obserwacje  $\mathbf{x}_1, \dots, \mathbf{x}_n$  (tak wczytuje to powyższy skrypt).
- Warto również zdefiniować macierz diagonalną  $\mathbf{B}$  o rozmiarze  $n \times n$ , gdzie  $B_{ii} = \beta_i$  (poza diagonalą są same zera).
- Wtedy hesjan można zapisać jako:

$$\mathbf{H}(\mathbf{w}_{t-1}) = \lambda \mathbf{I} + \mathbf{X}^\top \mathbf{B} (\mathbf{I} - \mathbf{B}) \mathbf{X} \dots$$

- ... ale uwaga: do macierzy  $\mathbf{X}$  po wczytaniu należy dodać kolumnę jedynek aby uwzględnić wyraz wolny!
- Podobnie, naturalne jest wczytanie wartości zmiennej wyjściowej jako wektora  $\mathbf{y}$  (tak wczytuje to powyższy skrypt). Wtedy gradient można zapisać jako jako:

$$\nabla L(\mathbf{w}_{t-1}) = \lambda \mathbf{w}_{t-1} - \mathbf{X}^\top \mathbf{B} \mathbf{y}.$$

### 4 Sztuczne zbiory danych

Celem zadania jest implementacja metody regresji logistycznej zgodnie z uwagami wymienionymi powyżej, a następnie użycie jej na dwóch małych, sztucznych zbiorach danych: [pierwszy zbiór](#) oraz [drugi zbiór](#). Format obu plików jest taki sam: pierwszy wiersz zawiera nazwy zmiennych oddzielone spacją, a każdy kolejny wiersz to opis jednej obserwacji z listą wartości zmiennych (również oddzielone spacją). *Wskazówka:* Dla pierwszego zbioru danych powinniście dostać współczynniki (przy  $\lambda = 0$ ):  $w_0 = -0.5873, w_1 = 1.1723, w_2 = 0.7065$ .

Odpowiedz na następujące pytania:

1. Jakie są wartości współczynników  $\mathbf{w}$ ?
2. Czy współczynniki zależą od małej regularyzacji w sposób istotny? Tzn. zmniejszając regularyzację do zera, czy współczynniki zmieniają się mocno?