Techniki eksploracji danych

Krzysztof Gajowniczek

Rok akademicki: 2020/2021

Section 1

Sztuczne sieci neuronowe

Subsection 1

Związek z regresją logistyczną

- **Sieci neuronowe** są powszechnie uważane jako modele będące czarną skrzynką.
- W pewnym sensie natomiast sieć neuronowa to niewiele więcej niż kilka połączonych ze sobą modeli regresji logistycznej.
- Regresja logistyczna opiera się na specyficznym sposobie wyrażania prawdopodobieństwa, zwanym szansą (ang. odds).
- Zamiast określać prawdopodobieństwo klasycznie, za pomocą stosunku liczby sukcesów do liczby wszystkich prób, oblicza się szansę, czyli stosunek prawdopodobieństwa sukcesu do prawdopodobieństwa porażki.
- Można ją łatwo wyliczyć ze zwykłego prawdopodobieństwa:

$$Odds = \frac{P}{1-P}$$

- Regresja logistyczna zakłada, że zmienna objaśniana ma rozkład dwupunktowy:

$$y_i \sim B(P_i, n_i)$$

- gdzie liczba prób w procesie Bernoulliego n_i jest znana, a prawdopodobieństwo sukcesu P_i jest nieznane.
- Załóżmy dalej, że rozwiązujemy problem klasyfikacji binarnej $y \in \{0, 1\}$.

- Model zakłada, że dla każdej próby Bernoulliego, istnieje zbiór p zmiennych objaśniających, które niosą pewną informację na temat prawdopodobieństwa sukcesu.
- Model przyjmuje wówczas postać:

$$P_i = E\left(\frac{y_i}{n_i} | \boldsymbol{x}_i^T\right)$$

- Logit nieznanego prawdopodobieństwa sukcesu P_i jest modelowany jako liniowa funkcja $\boldsymbol{x}_i^T$:

$$Logit(P_i) = \log\left(\frac{P_i}{1 - P_i}\right) = \boldsymbol{\theta}^T \boldsymbol{x}_i$$

$$\hat{y}_i = \log\left(\frac{P(y_i = 1)}{1 - P(y_i = 1)}\right) = \boldsymbol{\theta}^T \boldsymbol{x}_i$$

- Funkcja celu przyjmuję nasstepującą postać:

$$L = \prod_{i=1}^{n} P(y_i = 1)^{y_i} P(y_i = 0)^{1-y_i}$$

- a jej logarytm to:

$$\log L = \sum_{i=1}^{n} y_i \log P(y_i = 1) + (1 - y_i) \log P(y_i = 0)$$

Subsection 2

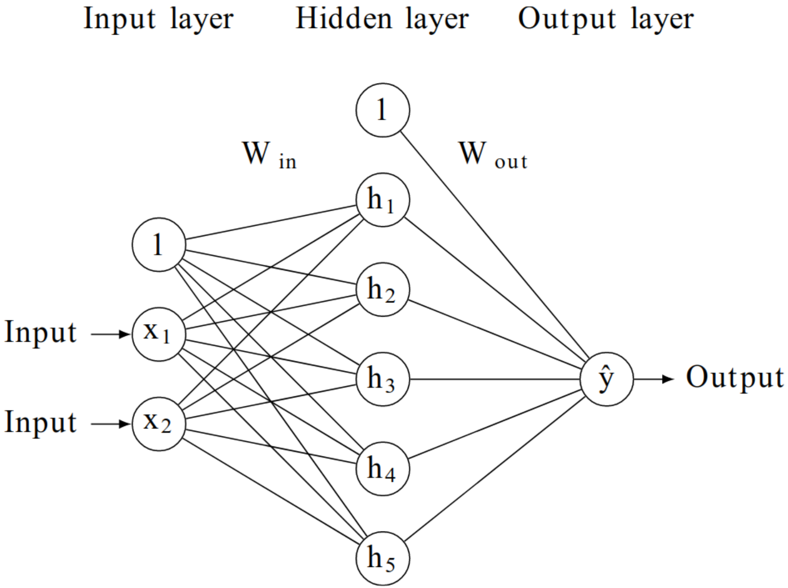
Architektura

- Sztuczna sieć neuronowa jest tak nazywana, ponieważ kiedyś uważano, że jest dobrym modelem działania neuronów w mózgu.
- Sieć neuronowa zorganizowana jest w warstwy.
- Warstwa wyjściowa Y to nasze oszacowanie prawdopodobieństwa, że obiekty należą do każdej klasy.
- Pośrodku znajduje się ukryta warstwa H (o wymiarze h), która jest przekształceniem przestrzeni wejściowej X.
- Następnie wykonujemy regresję logistyczną na tej przekształconej przestrzeni, aby oszacować klasę.

- Algorytm działania wygląda następująco:

1. Wygeneruj h różnych kombinacji liniowych zmiennych wejściowych X.
2. Zastosuj funkcję aktywacji, która dla każdej obserwacji „włącza” lub „wyłącza” każdy ukryty węzeł H.
3. Oszacuj h modeli regresji logistycznej dla przetransformowanych zmiennych z pkt. 1.
4. Dostosuj parametry zarówno wejścia, jak i wyjścia, aby zmaksymalizować prawdopodobieństwo.
5. Powtórz.

- Gdy $h = 1$ wtedy istnieje tylko jedna liniowa kombinacja zmiennych objaśniających, co w rzeczywistości oznacza brak jakiejkolwiek warstwy ukrytej, tj. zwykłą regresję logistyczną.


Input layer
Hidden layer
Output layer
1
W in
W out
h1
1
h2
Input
x1
h3
ŷ
Output
Input
x2
h4
h5

- Warstwa wejściowa to macierz X.
- Warstwa wyjściowa jest wektorem szacowanych prawdopodobieństw $\hat{\boldsymbol{y}}$.
- Sieć neuronowa dodaje warstwę ukrytą, o której można by myśleć jako pośrednią macierz projektową między wejściami a wyjściami.
- Uczenie głębokie to po prostu nauka sieci neuronowej z wieloma ukrytymi warstwami.
- Funkcja aktywacji jest często (nie zawsze) wybierana jako funkcja sigmoidalna:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Subsection 3

Propagacja sygnału

- Zaczynając od danych wejściowych X, przesyłamy sygnał dalej przez sieć w następujący sposób.
- Po pierwsze, obliczamy liniową kombinację zmiennych objasniających, używając macierzy wag $\boldsymbol{W}_{in} \in \mathbb{R}^{(p+1)\times h}$.

$$\boldsymbol{z}_1 = X\boldsymbol{W}_{in}$$

- Następnie wykorzystujemy funkcję aktywacji, aby uzyskać węzły w ukrytej warstwie H.

$$H = \sigma(\boldsymbol{z}_1)$$

- Dla warstwy wyjściowej obliczmy liniową kombinację ukrytych zmiennych H, tym razem używając innej macierzy wag $\boldsymbol{W}_{out} \in \mathbb{R}^{(h+1)\times(k-1)}$.

$$\boldsymbol{z}_2 = H\boldsymbol{W}_{out}$$

- Na końcu stosujemy jeszcze jedną funkcję aktywacji, aby uzyskać wynik:

$$\hat{\boldsymbol{y}} = \sigma(\boldsymbol{z}_2)$$

Subsection 4

Propagacja wsteczna sygnału

- Podobnie jak parametry w regresji liniowej, musimy wybrać wagi, które czynią nasz model „lepszym” według pewnych kryteriów, np:

$$L = f(\boldsymbol{W}) = \sum_{i=1}^{n} y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)$$

- Optymalizacja może być wykonana dzięki różnym algorytmom gradientowym:

$$\boldsymbol{W}_{t+1} = \boldsymbol{W}_t - \gamma \nabla f(\boldsymbol{W}_t)$$

- gdzie $\boldsymbol{W}$ jest macierzą wag synaptycznych w kroku t, ∇ jest gradientem funkcji L oraz γ jest parametrem szybkości uczenia.

- Korzystając z reguły łańcuchowej, gradient logarytmicznej wiarygodności L w odniesieniu do wag wyjściowych $\boldsymbol{W}_{out}$ jest określony przez:

$$\frac{\partial L}{\partial \boldsymbol{W}_{out}} = \frac{\partial L}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{W}_{out}}$$

- gdzie:

$$\frac{\partial L}{\partial \hat{\boldsymbol{y}}} = \frac{\boldsymbol{y}}{\hat{\boldsymbol{y}}} - \frac{1-\boldsymbol{y}}{1-\hat{\boldsymbol{y}}} = \frac{\hat{\boldsymbol{y}}-\boldsymbol{y}}{\hat{\boldsymbol{y}}(1-\hat{\boldsymbol{y}})}$$

$$\frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{W}_{out}} = H^T \sigma(H\boldsymbol{W}_{out})(1-\sigma(H\boldsymbol{W}_{out})) = H^T \hat{\boldsymbol{y}}(1-\hat{\boldsymbol{y}})$$

- Korzystając z reguły łańcuchowej, gradient logarytmicznej wiarygodności L w odniesieniu do wag wyjściowych $\boldsymbol{W}_{in}$ jest określony przez:

$$\frac{\partial L}{\partial \boldsymbol{W}_{in}} = \frac{\partial L}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial H} \frac{\partial H}{\partial \boldsymbol{W}_{in}}$$

- gdzie:

$$\frac{\partial \hat{\boldsymbol{y}}}{\partial H} = \sigma(H\boldsymbol{W}_{out})(1 - \sigma(H\boldsymbol{W}_{out}))\boldsymbol{W}_{out}^T = \hat{\boldsymbol{y}}(1 - \hat{\boldsymbol{y}})\boldsymbol{W}_{out}^T$$

$$\frac{\partial H}{\partial \boldsymbol{W}_{in}} = X^T\sigma(X\boldsymbol{W}_{in})(1 - \sigma(X\boldsymbol{W}_{in}))$$

Section 2

Literatura

- *Friedman, J., Hastie, T., Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, No. 10). New York: Springer series in statistics.*