Tại Sao Logistic Regression Sử Dụng Hàm Sigmoid?

Khoa KT (AI VIETNAM)

July 2021

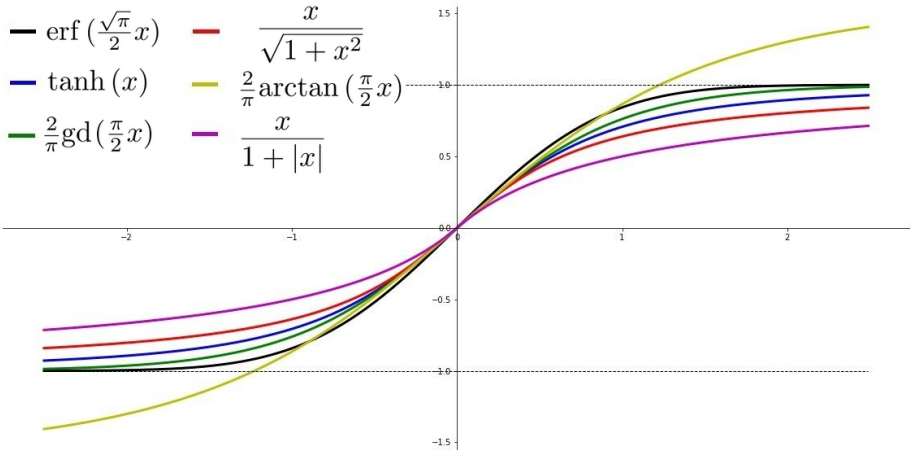
1 Giới Thiệu

Mọi người khi học AI đều sẽ phải tìm hiểu về Logistic Regression và Sigmoid. Bài viết này sẽ giải thích về một khía cạnh tại sao Sigmoid lại được sử dụng cho Logistic Regression mà không phải là các hàm activation nào khác dù có nhiều hàm tốt hơn.

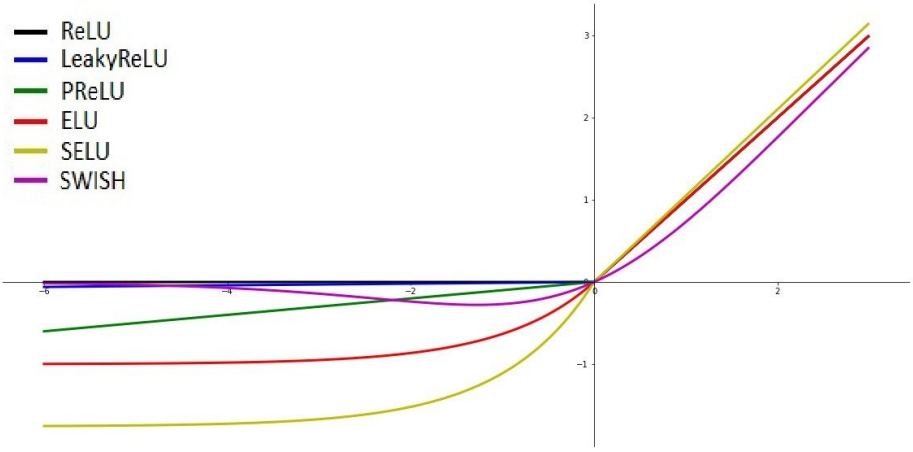
2 Mục Tiêu Của Logistic Regression

Bài toán mà Logistic Regression thực hiện là việc predict giá trị của y (binary values chỉ có giá trị 0 hoặc 1) được gọi là binary classification. Về bản chất Logistic regression là linear predictor và kết quả y không chỉ cho biết về class đó là gì mà còn có thêm thông tin về xác suất của class đó.
Mục tiêu:

- Thực hiện bài toán phân loại bằng một linear model $z = \mathbf{w}^T\mathbf{x} + b$. Tuy nhiên do $z \in (-\infty, +\infty)$, không thể thỏa mãn yêu cầu output range của bài toán.
- Sử dụng thêm một hàm bao bên ngoài để ánh xạ range của z từ $(-\infty, +\infty)$ sang $[0, 1]$. Điều này cũng đồng nghĩa với việc giả sử hàm bao này $g(z) \in [0, 1]$ **(1)** tuân theo Bernoulli distribution.
- Bernoulli distribution được định nghĩa chỉ bởi 1 số, model chỉ cần predict xác suất positive class $P(y = 1|z)$ do $P(Y = 0|z) = 1 - P(Y = 1|z)$.



Hình 1: Activations tương tự như sigmoid



Hình 2: Các họ ReLu

Hình 1 chỉ cần shift trục tung lên 1 đơn vị và scale 0.5 thì output của các activations này sẽ trong range $[0, 1]$ để thỏa mãn điều kiện ở trên.

Tuy nhiên, khi xét điều kiện output **(1)** $[0, 1]$ thì các họ của ReLu activations (Hình 1) đã vị phạm yêu cầu range của output do tại các vị trí $z > 0$ hàm vẫn tăng dần mà không có giới hạn.

3 Tại Sao Chọn Sigmoid (High-lelvel)

Từ công thức $P(y = 1|z)$ có thể được triển khai thành $P(y = 1|x; \mathbf{w})$ được gọi là ditribution của y khi biết trước x (đã được tham số hóa bởi w).
Tiếp theo, $\mathbf{w}$ là tham số cần học và x, y được xem là các biến ngẫu nhiên. Mục tiêu của bài toán là model sẽ phải học $\mathbf{w}$ sao cho distribution của model gần nhất với distribution của training data. Hay nói cách khác, phải tìm được bộ

tham số $\mathbf{w}$ để có xác suất cao nhất đối với tập training (là bài toán Maximum Likelihood Estimation)

$$\mathbf{w} = \arg\max_{\mathbf{w}} P(\mathbf{y}|\mathbf{X}; \mathbf{w})$$

Đi cùng với giả định dữ liệu được sinh ra ngẫu nhiên và độc lập nên:

$$P(\mathbf{y}|\mathbf{X}; \mathbf{w}) = \prod_{i=1}^{N} P(y_i|\mathbf{x}_i; \mathbf{w})$$

Và để tránh việc tích của các xác suất sẽ là con số rất nhỏ và tăng độ phức tạp trong tính toán, hàm likelyhood sẽ được lấy logarit cơ số e để chuyển thành các phép tính tổng.

$$\log(P(\mathbf{y}|\mathbf{X}; \mathbf{w})) = \sum_{i=1}^{N} P(y_i|\mathbf{x}_i; \mathbf{w})$$

Và thay vì đi tìm max likelihood cho function này, thì sẽ đi tìm negative log likelihood và hàm này được gọi là hàm loss

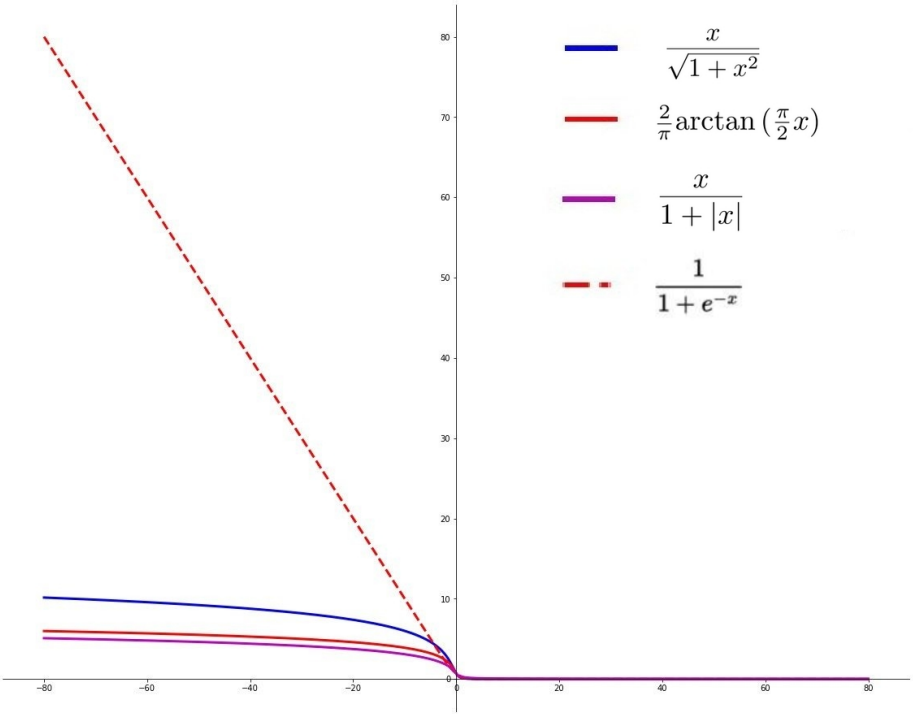
$$J(\mathbf{w}) = -\log(P(\mathbf{y}|\mathbf{X}; \mathbf{w}))$$

Bởi vì model sẽ học bằng cách tối ưu hóa dựa vào gradient (đạo hàm), do đó hàm activation được chọn phải thỏa mãn thêm điều kiện continuous, differentiable. Do đó gradient của output model với các parameter không được bằng 0 **(2)**. Bởi vì khi gradient bằng 0 model sẽ không còn biết học tiếp như thế nào để cải thiện các tham số.

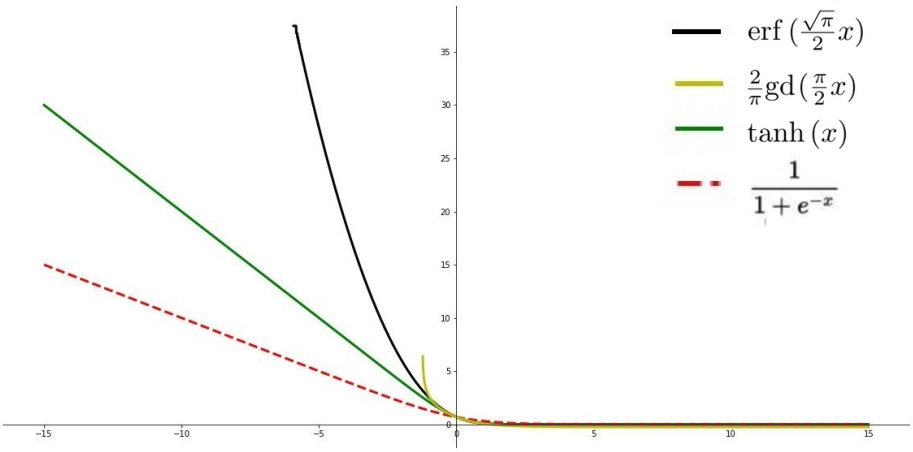
Chú ý: Hình 3, Hình 4, và Hình 5 là negative log likelihood của function tương ứng.

Từ Hình 3, các function màu tím, đỏ và xanh dương gần như có slope bằng 0 khi tiến x tiến về phía trái (sẽ không có đạo hàm). Chỉ có hàm sigmoid (đỏ nét đứt) là tương đối giảm đều. Do đó 3 hàm màu tím, đỏ và xanh dương không thỏa mãn điều kiện **(2)** nên bị loại.

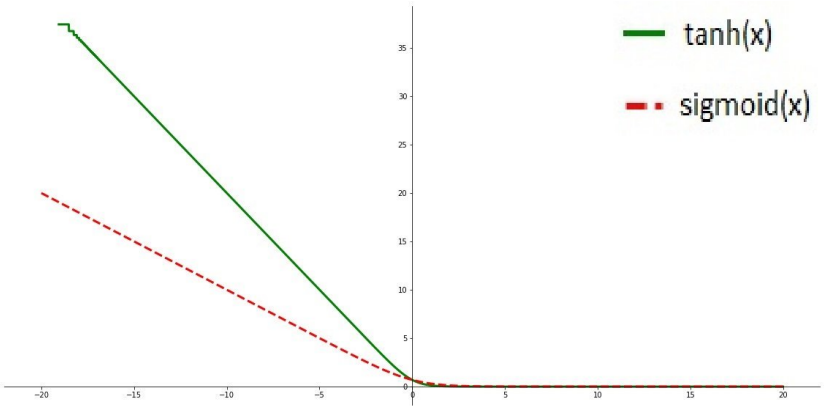
Từ Hình 4, function màu vàng không ổn định, và hàm màu đen slope gần như thẳng đứng dẫn đến việc các step khi học sẽ không ổn định. Do đó 2 activation này bị loại. Chỉ còn 2 activation là màu sigmoid (đỏ đứt nét) và tanh (xanh lá cây, hàm đã shift 1 và scale 0.5) Hình 5 là vẫn còn đủ điều kiện.



Hình 3: Negative log likelihood của các activation



Hình 4: Negative log likelihood của các activation



Hình 5: Negative log likelihood của tanh và sigmoid

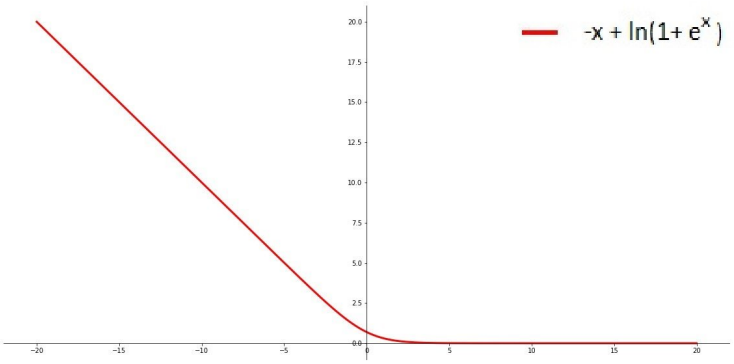
4 Tại Sao Chọn Sigmoid (Low-level)

Để hiểu rõ hơn vì sao sigmoid hoạt động tốt hơn hàm loss với sigmoid sẽ được triển khai với $y = 1$ (tương tự cho $y = 0$) như sau

$$J(\mathbf{w}) = -\log P(\mathbf{y} = 1|\mathbf{X}; \mathbf{w})) = -\log(\frac{1}{1 + e^{-z}}) = -z + \log(1 + e^{z})$$

Có thể thấy rằng $J(\mathbf{w})$ có chứa thành phần linear là $-z$. Bây giờ sẽ có 2 trường hợp:

- **z lớn:** model đã dự đoán đúng vì $y = 1$. Có thể thấy rằng $\log(1 + e^{z})$ sẽ tiệm cận với z khi z lớn dẫn đến việc 2 thành phần này triệt tiêu nhau do trái dấu. Điều này làm cho hàm loss là gần bằng 0. Đây là hoàn toàn hợp lý do model đã học đúng nên gradient cũng yếu dần và việc học chậm dần cho đến khi gần như không học nữa.
- **z nhỏ:** model dự đoán sai. $\log(1 + e^{z})$ sẽ gần bằng 0 dẫn đến $J(\mathbf{w})$ gần bằng $-z$. Điều này đồng nghĩa với việc slope gần bằng -1, tạo ra một độ dốc khá đều như Hình 6



Hình 6: Kết quả của hàm loss

Đi sâu hơn nữa vào vấn đề, Bayesian rule sẽ được sử dụng cho việc phân loại. Giả sử bài toán binary classification trên data có 1 chiều, mỗi class sẽ có variance giống nhau và mean lần lượt là 3 và 5, và $P(class_1) = P(class_2) = 0.5$. Ta có xác suất prior $P(class_1)$, $P(class_2)$, likelihood của data $P(X|class_1)$, $P(X|class_2)$. Áp dụng Bayes rule để tìm posterior:

$$P(class_1|X) = \frac{P(X|class_1)P(class_1)}{P(X|class_1)P(class_1) + P(X|class_2)P(class_2)}$$

Nếu chia mẫu cho posterior thì thu được:

$$P(class_1|X) = \frac{1}{1 + \frac{P(X|class_1)P(class_1)}{P(X|class_2)P(class_2)}}$$

Trong đó do tuân theo Gauss distribution:

$$P(X|class_1) = N(3,1) \approx e^{(\frac{(x-3)^2}{2})}$$

$$P(X|class_2) = N(5,1) \approx e^{(\frac{(x-5)^2}{2})}$$

$$\frac{P(X|class_1)}{P(X|class_2)} = e^{(\frac{(x-3)^2}{2} - \frac{(x-5)^2}{2})} = e^{(}2x - 8)$$

Nếu như xem $z = 2x - 8$, và để posterior và z trở thành monotonic trong cùng một hướng do đó:

$$P(class_1|X) = \frac{1}{1 + e^{-z}}$$

5 Kết Luận

Có 3 lý do hàm sigmoid được chọn cho logistic regression:

- Output region là xác suất và trong range $[0, 1]$.
- Negative log likelihood của sigmoid cho thấy hàm có slope đều thuận lợi cho việc đọa hàm.
- Sigmoid đến từ Bayes rule cho việc binary classification. Hơn nữa, nó không yêu cầu distribution là Gaussian, chỉ cần distribution có họ tương tự hàm mũ và là binary classification.

6 References

- https://stats.stackexchange.com/questions/162988/why-sigmoid-function-instead-of-anything-else?fbclid=IwAR3yBqPnMFK0eCAwHdc4JePaCKkJkSZGC3kIGs5f6ajYZAri8loKWLnU1Vw
- https://towardsdatascience.com/why-sigmoid-a-probabilistic-perspective-42751d82686
- https://www.youtube.com/watch?v=WsFasV46KgQ
- https://machinelearningcoban.com/2017/01/27/logisticregression/mo-hinh-logistic-regression
- https://www.quora.com/Logistic-Regression-Why-sigmoid-function
- https://sebastianraschka.com/faq/docs/logistic-why-sigmoid.html
- https://towardsdatascience.com/why-sigmoid-a-probabilistic-perspective-42751d82686