Logistic Regression

Dereje T. Abzaw

Logistic Regression

- LR is the algorithm suited for discovering the link between **features** and a **particular outcome.**
- In NLP, logistic regression is the baseline **supervised machine learning algorithm** for classification
- LR has a very close relationship with **neural networks.**
- **Logistic regression** can be used to classify an observation into one of two classes or into one of many classes.

Discriminative - Generative

- What is the main difference between LR and Naive Bayes?
  - LR is a **discriminative classifier** while naive Bayes is a **generative classifier.**
- A discriminative classifier is a type of model that focuses on modeling the decision boundary between classes.
- Instead of modeling how data is generated, as generative models do, discriminative classifiers model the probability of the target class given the features directly.
- Essentially, they distinguish between different classes by learning what makes each class unique.

Collars?

- For Example: We're trying to distinguish dog images from cat images.
  - A generative model would have the goal of understanding what dogs look like and what cats look like and will 'generate', i.e., draw, a dog.
  - Given a test image, the system then asks whether it's the cat model or the dog model that better fits the image, and chooses that as its label.
  - A **discriminative model**, by contrast, is only trying to learn to distinguish the classes usually without learning much about them.
  - So maybe all the dogs in the training data are wearing collars and the cats aren't. If that one feature neatly separates the classes, the model is satisfied.
    - If you ask such a model what it knows about cats all it can say is that they don't wear collars.

Contd ...

Collars?

No Collar

Wear Collars

Formally …

- Assigns a class c to a document d **not by directly computing P(c|d)**

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} \; \overbrace{P(d|c)}^{\text{likelihood}} \; \overbrace{P(c)}^{\text{prior}}$$

- A discriminative model attempts to **directly compute P(c|d)**.
- Perhaps it will learn to assign a high weight to document features that directly improve its ability **to discriminate** between possible classes

Sigmoid Function

- The goal of binary logistic regression is to train a classifier that can make a binary decision about the class of a new **input observation**.
- Consider a single input observation **x**, which we will represent by a vector of features [x1, x2, ..., xn].
  - The classifier output y can be 0 or 1
  - The probability **P(y = 1|x)** ?
- Logistic regression solves this task by learning, from a training set, **a vector of weights** and a **bias term (Intercept).**

Contd ...

- The weight represents how important that input feature is to the classification decision
- Can be positive or negative
  - Awesome - Positive weight
  - Horrible - Negative weight

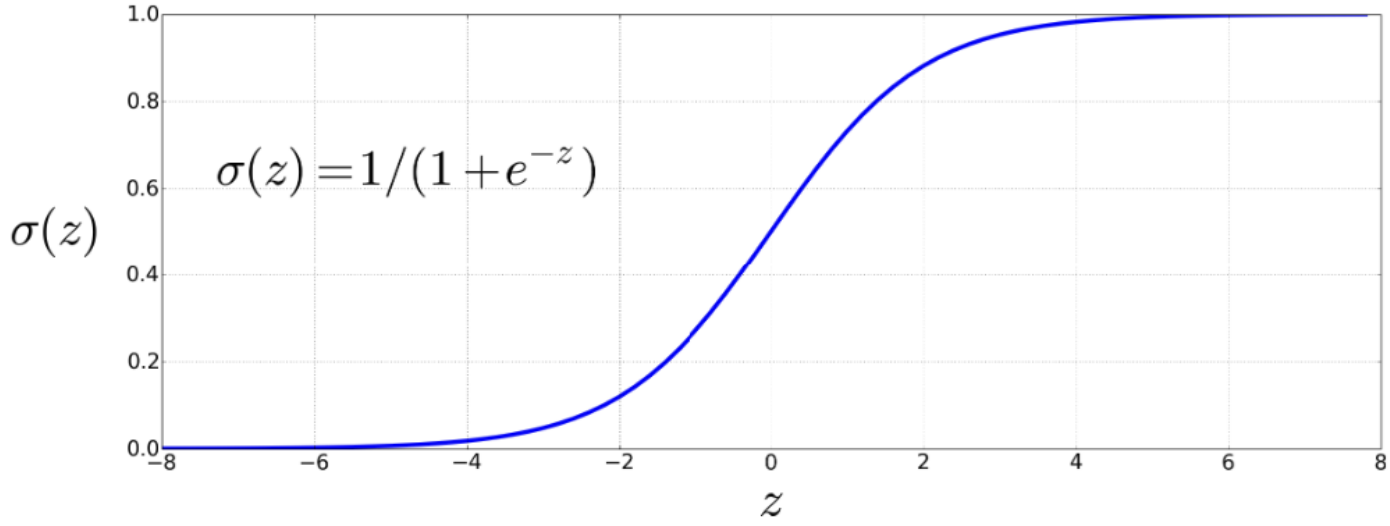
$$z = \left(\sum_{i=1}^{n} w_i x_i \right) + b \quad \xrightarrow{\text{Dot Product}} \quad z = \mathbf{w} \cdot \mathbf{x} + b$$

Legal Probability

- The probability should lie between 0 or 1.
- To create a probability of that type , the sigmoid function, **σ (z)** is used.
- Sigmoid function(**logistic function**) has the following equation
- Input to Sigmoid function, the scores of Z , is referred to as **logits**

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

Graphically …



Contd ….

- The sigmoid function takes a real value and maps it to the range (0, 1).
- Since weights are real-valued, the logits , z , ranges from −∞ to ∞