ICTS 4111/5111: Introduction to NLP

Razvan Bunescu

Lecture notes, March 21, 2024

1 Binary and Multinomial Logistic Regression

In binary LR, we have only two classes C_1 and C_2. To do classification of a vector of features $\mathbf{x} = [x_1, x_2, ..., x_F]$, we first need to train a vector of parameters $\mathbf{w} = [w_1, w_2, ..., w_F]$ and bias b. These parameters are used to compute the probability that $\mathbf{x}$ belongs to class C_1 (the positive class), as follows:

1. We first compute the *logit* score $z = \mathbf{w}^T\mathbf{x} + b$.
2. Then we squash it between 0 and 1 using the logistic sigmoid $p(C_1|\mathbf{x}) = \sigma(z) = \dfrac{1}{1+e^{-z}}$.
3. Then we can do classification by saying that $\mathbf{x}$ is in class C_1 (positive) if and only if $p(C_1|\mathbf{x}) = \sigma(z) \geq 0.5$.
   - We showed in class that this is equivalent with $z \geq 0$.

There are many *multiclass* classification problems where the number of classes is $K \geq 2$, for example LLMs predict one token auto-regresively at each step. Each token is a class, therefore there are $|V|$ classes, where V is the vocabulary.

In general, we have a set of classes $\mathcal{C} = \{C_1, C_2, ..., C_K\}$, where $K \geq 2$. For this multiclass classification case, we use *multinomial LR*, where we train a set of parameters $\mathbf{w^{(k)}}$ and $b^{(k)}$ for each class $1 \leq k \leq K$. To do classification, we precoeed as follows:

1. We compute a logit score $z_k = \mathbf{w^{(k)}}^T\mathbf{x} + b^{(k)}$ for each class k.
2. Now we have a vector of logit scores $\mathbf{z} = [z_1, z_2, ..., z_K]$. We want to transform these scores into probabilities $\mathbf{p} = [p_1, p_2, ..., p_K]$, where each p_k is interpreted as representing the probability the example belongs to class C_k, i.e. $p_k = p(C_k|\mathbf{x})$. For this, we will use the *softmax* function, which means:

$$
\begin{aligned}
\mathbf{p} &= softmax(\mathbf{z}) \\
p_1, p_2, ..., p_K &= softmax(z_1, z_2, ..., z_K] \\
p_k &= \frac{exp(z_k)}{Z} = \frac{exp(z_k)}{\displaystyle\sum_{j=1}^{K} exp(z_j)}
\end{aligned}
$$

   where Z is the normalization required to make all probabilities sum up to 1, also called the *partition* function.

Some exercises:

1. If the total number of features is $F = 512$ in $\mathbf{x}$, and the number of classes is $K = 30,000$, then the total number of parameters in the multinomial LR model will be (the number of classes) $\times$ (the number of params for each class) $= K \times (F + 1) = 30,000 \times 513 = 15.4M$ params.

2. If the total number of features is $F = 5000$ in $\mathbf{x}$ and the number of classes is $K = 3$ (positive, negative, neutral sentiment), then the total number of params will be $3 \times 5001 = 15,003$ params.

Classification is done by selecting the class with the highest computed probability:

$$\hat{k} = \underset{1 \le k \le K}{\arg\max}\, p(C_k|\mathbf{x}) = \underset{1 \le k \le K}{\arg\max}\, exp(z_k) = \underset{1 \le k \le K}{\arg\max}\, z_k \tag{1}$$