ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν Πανεπιστήμιον Αθηνών
ΙΔΡΥΘΕΝ ΤΟ 1837
ATHENA Research & Innovation Information Technologies
ΙΕΛ LSP

M902
Βασικές Μαθηματικές Έννοιες στη Γλωσσική Τεχνολογία

Project Recap

Κυλάφη Χριστίνα-Θεανώ

LT1200012

January, 2021

TABLE OF CONTENTS

Question **1**

1. Let $X_{1xn} = [\ x_1,\ x_2,\ \ldots\ ,\ x_n\ ]$ the input feature values of the neural network model and $W_{1xn} = [\ w_1,\ w_2,\ \ldots\ ,\ w_n\ ]$, the learnt weights during the training process (and w_0 a bias to be added in the weighted sum).

Then the weighted sum is calculated as follows:

$$z_{1x1} = X\ W^T = [\ X\ ]_{1xn}\ \left[\ W^T\ \right]_{nx1} = [\ x_1\ x_2\ \ldots\ x_{n-1}\ x_n\ ] \begin{bmatrix} w_1 \\ w_2 \\ \cdots \\ w_{n-1} \\ w_n \end{bmatrix}$$

$$= [\ x_1 w_1\ +\ x_2 w_2\ +\ \ldots\ +\ x_n w_n\ ]$$

$$= \left[\ \sum_1^n x_i w_i\ \right]$$

where W^T the transpose of the weight matrix. The previous output could also be calculated by transposing the input values matrix and the formula $z_{1x1} = W\ X^T = \left[\ \sum_1^n w_i x_i\ \right] = \left[\ \sum_1^n x_i w_i\ \right]$.

2. Let σ the sigmoid function, where $\sigma(x) = \dfrac{1}{1+e^{-x}}$.

A bias w_0 is added to the weighted sum z:

$$z' \rightarrow \text{bias}\ +\ z = w_0\ +\ \sum_1^n x_i w_i = w_0\ +\ x_1 w_1\ +\ x_2 w_2\ +\ \ldots\ +\ x_n w_n$$

Then, in order to calculate the derivative of the output, we convert z' into a function of the $f(x) = a\,x + c$ form :

$$f(w_1) = z'(w_1) = w_0\ +\ x_1 w_1\ +\ x_2 w_2\ +\ \ldots\ +\ x_n w_n = x_1 w_1\ +\ c\ ,$$

where:
$x_1,\ c = w_0\ +\ x_2 w_2\ +\ \ldots\ +\ x_n w_n$, constants.

Then:

$$\sigma(f(w_1)) = \frac{1}{1+e^{-f(w_1)}} = \frac{1}{1+e^{-(x_1w_1+c)}} = \frac{1}{1+\frac{1}{e^{(x_1w_1+c)}}} = \frac{1}{\frac{e^{(x_1w_1+c)}+1}{e^{(x_1w_1+c)}}}$$

$$= \frac{e^{(x_1w_1+c)}}{e^{(x_1w_1+c)}+1} = \frac{e^{x_1w_1}e^{c}}{e^{x_1w_1}e^{c}+1} \qquad (1)$$

$\sigma(f(w_1))$ is differentiable in $Dom(\sigma)$, as a composition of differentiable functions $\sigma,\ f,$ with its derivative $\sigma'(f(w_1))$ as follows:

$$\frac{d(\sigma(f(w_1)))}{dw_1} = \sigma(\ f(x)\ )\ (\ 1-\sigma(\ f(x)\ )\ )\ = \sigma'(f(w_1))$$

$$= \left(\frac{1}{1+e^{-f(w_1)}}\right)' \overset{(1)}{=} \left(\frac{e^{x_1w_1}e^{c}}{e^{ax}e^{b}+1}\right)'$$

$$= \frac{(e^{x_1w_1}e^{c})'(e^{x_1w_1}e^{c}+1)\ -\ (e^{x_1w_1}e^{c})(e^{x_1w_1}e^{c}+1)'}{(e^{x_1w_1}e^{c}+1)^2}$$

$$= \frac{(x_1e^{x_1w_1}e^{c})(e^{x_1w_1}e^{c}+1)\ -\ (e^{x_1w_1}e^{c})(x_1e^{x_1w_1}e^{c})}{(e^{x_1w_1}e^{c}+1)^2}$$

$$= \frac{(x_1e^{x_1w_1}e^{c})(e^{x_1w_1}e^{c}+1\ -\ e^{x_1w_1}e^{c})}{(e^{x_1w_1}e^{c}+1)^2}$$

$$= \frac{x_1e^{x_1w_1+c}}{(e^{x_1w_1}e^{c}+1)^2} \ \rightarrow\ \text{derivative of the network output } z' \text{ (} z \text{ with added bias } w_0 \text{)}$$

with respect to the parameter w_1

Question **2**

Let

$D = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N) \}$
$y_i \in \{ 0, 1 \}, i \in [1, 2, \ldots, N]$

1. In order to perform density estimation on the given data, we use the Maximum Likelihood Estimation method (**MLE**). Assuming that the observations are independent and identically distributed - **I.I.D.** (following the Bernoulli distribution $Bern(Y = y; \theta)$, where), the likelihood of all the data is the product of the respective **likelihood** of each point
$(x_i , y_i), i \in [1, \ldots, N]$:

$$\begin{aligned}\mathscr{L}(\theta) &= P(x_1 \to y_1, x_2 \to y_2, \ldots, x_n \to y_n | \theta) \\ &= \prod_{i=1}^{n} P(x_i \to y_i | \theta) = \prod_{i=1}^{n} \hat{y}_i^{\,y_i} \, (1 - \hat{y}_i)^{\,1-y_i} \\ &= \prod_{i=1}^{n} f(x_i | \theta)^{y_i} \, (1 - f(x_i | \theta))^{1-y_i} \quad (1)\end{aligned}$$

2. The expression above is **not** part of the **optimal** solution for the task. First of all, it is not so easy to differentiate (during the maximisation process). Also, the sequential multiplications can lead to considerably **small** values, a situation that involves the risk of **underflow** (running out of floating point precision which means lack of enough bits to represent those really small numbers, thus rounding them to ~0) . Therefore, the natural logarithm of this expression is preferred (transition from **product** to **sum**). Summing is less expensive than multiplication and the logarithm is a monotonically increasing function, ensuring that the maximum value occurs at the same point as the original probability function. For the aforementioned reasons, we alter (1) using the logarithm function and calculate the log-likelihood as follows:

$$\mathscr{LL}(\theta) = \log(\mathscr{L}(\theta))$$

$$\mathscr{LL}(\theta) = \log(\mathscr{L}(\theta)) = \log\left(\prod_{i=1}^{n} f(x_i | \theta)^{y_i} \, (1 - f(x_i | \theta))^{1-y_i} \right)$$

$$= \sum_{i=1}^{n} \log\left(f(x_i|\theta)^{y_i} \, (1 - f(x_i|\theta))^{1-y_i} \right)$$

$$= \sum_{i=1}^{n} \left(y_i \log\big(f(x_i|\theta)\big) + (1 - y_i) \log\big(1 - f(x_i|\theta)\big) \right)$$

$$= \sum_{i=1}^{n} y_i \log\big(f(x_i|\theta)\big) + \sum_{i=1}^{n} (1 - y_i) \log\big(1 - f(x_i|\theta)\big)$$

$$= \left(\sum_{i=1}^{n} y_i \right) \log\big(f(x_i|\theta)\big) + \left(\sum_{i=1}^{n} (1 - y_i) \right) \log\big(1 - f(x_i|\theta)\big)$$

$$= Y \log\big(f(x_i|\theta)\big) + (n - Y) \log\big(1 - f(x_i|\theta)\big),$$

where

$$Y = \sum_{i=1}^{n} y_i$$

Question **3**

1. The set $\{-4, -3, -2, -1, 0, 1, 2, 3, \dots\}$ can also be notated as:

$b \rightarrow \{\ x \in \mathbb{Z} : -4 \leq x\ \}$, which means that the set consists of integer numbers that are greater or equal than the integer -4.

2. Let

$A = \{\text{Greek articles}\} = \{\text{οι, η, τους, ένας, των, ... }\}$
$B = \{\text{Greek vowels}\} = \{\ \text{α, ε, η, ι, ο, υ, ω }\}$

Then:

$A \cap B = \{\text{Greek articles}\} \cap \{\text{Greek vowels}\}$

$= \{\text{οι, η, τους, ένας, των, ... }\} \cap \{\ \text{α, ε, η, ι, ο, υ, ω }\}$

$= \{\ \text{α, ε, η, ι, ο, υ, ω }\} = \{\ \text{Greek vowels }\}$

$= B$

3. Let

$A = \{\ \text{Words in Document A }\}$
$B = \{\ \text{Words in Document B }\}$
$A \subset B$

Then:

a. $B - A$ **:** The set consisting of words contained in Document B that are not contained in Document A .

b. $A \subset B \implies |A| \leq |B|$**,** where $|A|$, $|B|$ the respective cardinalities of sets A, B .

c. Sets A and B, carry information about the unique words that occur in the respective documents (a set cannot contain duplicates by definition). No information is given about the frequency of each word, thus the total words comprising the documents. Therefore, no safe conclusion can be made concerning the lengths of the two Documents A and B .

Question **4**

Let **2, 4, 15, 8, 7, 7, 6, 3, 3, 3, 40** observations (x_i , $i = 1, 2, \ldots, 11$) of a random variable X.

Then the mean and median values are calculated as follows:

a. Mean value:

$$\frac{\sum_1^{11} x_i}{11} = \frac{2 + 4 + 15 + 8 + 7 + 7 + 6 + 3 + 3 + 3 + 40}{11} = 8.91$$

Median value:

2, 3, 3, 3, 4, 6, 7, 7, 8, 15, 40

b. The **mean** value is the most commonly used statistical measure of central tendency with regard to an observation list. However, sometimes due to the structure of the observations' distribution, the **median** is more representative and a more appropriate metric of the data. For example, **not normally distributed** values or the existence of **outliers**, considerably affect the value of the distribution's mean. In our case, there is an **outlier** of value **40** in the dataset, so even though the two values, mean and median, are not too different from one another (**8.91** VS **6**), the **mean** value exhibits **sensitivity** to the **outlier**, **proving** that the **median** is a **better** choice for getting a proper **idea** of the **data distribution** given above.

Question **5**

A **statistical measure** for the evaluation of a **word**'s importance in a document, is **TF-IDF**, that stands for term frequency-inverse document frequency. It can successfully been used for the tasks of relevance scoring and ranking (recommendation/information retrieval/search) as well as filtering (text cleaning) to name a few.

The calculated value represents the **weight** of each word, consisting of two parts that measure the importance of the word in a particular document with consideration to both the exact document and the corpus (total collection of documents) factors. This way, the **significance** of the word is estimated in a more accurate manner.
Specifically, the **TF-IDF** of a word $w_i \in W$ in a document $d_i \in D$, where W the total number of unique words contained in all the corpus comprising D documents, is **calculated** as follows:

$$TF(w_i, d_i) = \frac{occ_{w_i d_i}}{\sum_{i=1}^{W} occ_{w_i d_i}}$$

where
$occ_{w_i d_i}$, the **occurrences** of the **word** w_i in document d_i

$\sum_{i=1}^{W} occ_{w_i d_i}$, the **sum** of all the **words** in **document** d_i (length of d_i)

$$IDF(w_i, D) = \log\left(\frac{D}{docs_{w_i}}\right)$$

where
D , the **total** number of **documents** in the collection
$docs_{w_i}$, the number of the **documents** in the corpus that **contain** the **word** w_i

Then, the **final** value/weight of the word, is the **product** of the values of the two respective statistics above:

$$TF - IDF(w_i, d_i) = TF(w_i, d_i) \ * \ IDF(w_i, D)$$

The significance of a word increases proportionally to its occurrences in the target document, while at the same time decreases as its frequency in the rest of the collection rises.

To **conclude**, **TF-IDF** is a rather practical and accurate measure in calculating the importance of a word as an independent occurrence with respect to a certain document, by exploiting the hybrid model of the two aforementioned statistics. **However**, it is unable to capture conditional dependencies, semantics, positions, co-occurrences of words, which might be important and needed depending on the task at hand. Therefore, this method is suggested to be employed as only a part of the process (e.g. feature extraction / classification / etc.).