ISEP
Machine Learning
November 4th 2024

Tutorial course 5 : Introduction to deep learning

1 Part I : Exercises

Exercise 1. Consider a neural network with the following characteristics : 4 input features, 2 units in the first hidden layer, 3 units in the second hidden layer, and a single unit in the output layer.

1. Sketch the neural network. You will indicate the number of layers L, number of units in each layer $n^{[\ell]}$, the dimensions of the parameter matrices $W^{[\ell]}$ and $b^{[\ell]}$ for each layer ℓ.
2. How many parameters to estimate are there ?
3. You are fitting your model to perform binary classification. Calculate the value of $\hat{y} = P(Y/X)$ considering the following information :
   — $X = [4, 0, 1, 3]$
   — the parameters are initialized to : $\mathbf{W}^{[1]} = \begin{bmatrix} 7 & -10 & 3 & -1 \\ -1 & -3 & -9 & -10 \end{bmatrix}$; $\mathbf{b}^{[1]} = \begin{bmatrix} 0 \\ -2 \end{bmatrix}$;
   $\mathbf{W}^{[2]} = \begin{bmatrix} -5 & 1 \\ 2 & -2 \\ -1 & 1 \end{bmatrix}$; $\mathbf{b}^{[2]} = \begin{bmatrix} 0 \\ 4 \\ -3 \end{bmatrix}$; $\mathbf{W}^{[3]} = \begin{bmatrix} -5 & 1 & 2 \end{bmatrix}$; $\mathbf{b}^{[3]} = \begin{bmatrix} -3 \end{bmatrix}$.
   Consider the `ReLU` activation function for the first hidden layers and the `tanh` function for the second one.
4. Calculate the loss function if $y = 1$.

Exercise 2. Consider the simple function $R(\beta) = \sin(\beta) + \beta/10$.

1. Draw a graph of this function over the range $\beta \in [-6, 6]$ increment by 0.1.
2. What is the derivative of this function ?
3. Initialize β to 2.3, run 100 iteration of the gradient descent algorithm to find a local minimum of $R(\beta)$ using a learning rate of $\alpha = 0.1$. Show each obtained value of β after each iteration in your plot, as well as the final answer.
4. Repeat using as initial value $\beta = 1.4$. Compare the results and conclude.

Exercise 3.* When the target variable Y has more than one class, the activation function used in the output layer is called softmax and it is a generalization of the sigmoid function to κ classes. This is called *multinomial classification*. The softmax function has the following expression :

$$P(Y = k|X) = \frac{e^{z_k}}{\sum_{l=1}^{\kappa} e^{z_l}} \quad k \in 1, ..., \kappa$$

1. Demonstrate that if we add a real constant c to each of the z_j, then the probability is unchanged.
2. Consider the formula given in Lecture 2 :

$$P(Y = k|X) = \frac{e^{\beta_{0k} + \beta_{1k} X_1 + ... + \beta_{pk} X_p}}{\sum_{l=1}^{\kappa} e^{\beta_{0l} + \beta_{1l} X_1 + ... + \beta_{pl} X_p}} \quad k \in 1, ..., \kappa$$

   Demonstrate that if we add constants c_j, $j = 1, ..., p$, to each of the corresponding coefficients of each feature for each of the classes, then the predictions at any new point x are unchanged.

2 Part II : Practical application

From moodle download the folder *Lab5 - Introduction to neural networks*. You will work on 2 labs described in the following jupyter notebook files :

— `Building+your+DeepNN+Step+by+Step.ipynb`

— `Deep-NN-Application.ipynb`

Previously, you will need to install Jupyter notebook.

`https://jupyter.readthedocs.io/en/latest/install.html`

To open a Jupyter notebook you have to open a terminal and type :

jupyter notebook

Then, click on the file name.

References

— James, Gareth ; Witten, Daniela ; Hastie, Trevor and Tibshirani, Robert. "An Introduction to Statistical Learning with Applications in R", 2nd edition, New York : "Springer texts in statistics", 2021. Site web : `https://hastie.su.domains/ISLR2/ISLRv2_website.pdf`.

— Mooc Coursera : `https://fr.coursera.org` , Âń Deep Learning specialization Âż. Course 1 : Neural Networks and Deep Learning. Site web : `https://fr.coursera.org/specializations/deep-learning` , 2018.