Deep Learning - Homework 0

Brendon Boldt

Sep/11/2017

1 Course Preparation

1.1 Python Requirements

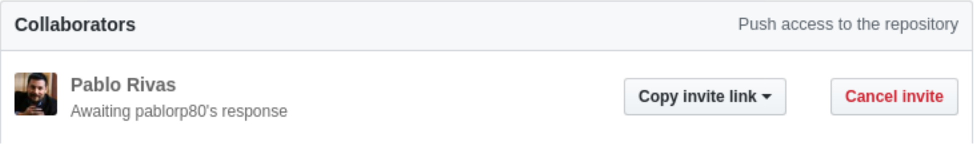
The following source code demonstrates the installation of the required Python libraries:

```
# file: hw0.libraries.py
# author: Brendon Boldt
# version: 1.1
# date: Sep/11/2017
#
# This file demonstrates the installation of the Python
# libraries required for deep learning.

import sys
import numpy
import scipy
import sklearn
import matplotlib
import pandas
import tensorflow

# Print the version number of the modeles as proof-of-installation
print(sys.version)
print(numpy.__version__)
print(scipy.__version__)
print(sklearn.__version__)
print(matplotlib.__version__)
print(pandas.__version__)
print(tensorflow.__version__)
```

```
> python3 hw0.libraries.py
3.5.2 (default, Nov 17 2016, 17:05:23)
[GCC 5.4.0 20160609]
1.13.1
0.19.1
0.19.0
2.0.2
0.20.3
1.3.0
```

1.2 GitHub Repository

My GitHub profile can be found at `https://github.com/brendon-boldt`, and the class repository at `https://github.com/brendon-boldt/cmpt-469`. The following screenshot is not fabricated



Collaborators

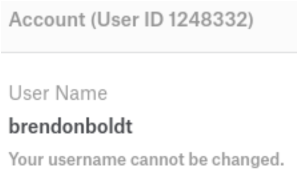
Push access to the repository

Pablo Rivas

Awaiting pablorp80's response

1.3 Kaggle Account

My Kaggle username is `brendonboldt`, and the following screenshot is taken from my account page:



Account (User ID 1248332)

User Name

brendonboldt

Your username cannot be changed.

2 Solution to Problem 1

For the function $g(x) = -3x^2 + 24x - 30$, find the value for x that maximizes $g(x)$.

We know that $g'(x) = -6x+24$; we can then see that $g(x)$ has one critical point at $x = 4$. Since $g'(x) > 0$ for $x < 4$ and $g'(x) < 0$ for $x > 4$ we know that the critical point at $x = 4$ is a local maximum. Since $g(x)$ is of degree 2, it has only one local maximum or minimum; we also know that $\lim_{x\to-\infty} = -\infty$ and $\lim_{x\to\infty} = -\infty$. Thus, the local maximum at $x = 4$ is the global maximum.

3 Solution to Problem 2

Consider the following function:

$$f(x) = 3x_0^3 - 2x_0x_1^2 + 4x_1 - 8$$

what are the partial derivatives of $f(x)$ with respect to x_0 and x_1?

It is the case that $\frac{\partial f}{\partial x_0} = 9x_0^2 - 2x_1^2$ and $\frac{\partial f}{\partial x_1} = -4x_0x_1 + 4$.

4 Solution to Problem 3

Consider the matrix $A = \begin{bmatrix} 1 & 4 & -3 \\ 2 & -1 & 3 \end{bmatrix}$ and $B = \begin{bmatrix} -2 & 0 & 5 \\ 0 & -1 & 4 \end{bmatrix}$, then answer the following and verify your answers in Python:

(a) *can you multiply the two matrices? Elaborate on your answer.*

(b) *multiply A^T and B and give its rank.*

(c) *let $C = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$ be a new matrix; what is the result of $AB^T + C^{-1}$*

(a) No; it is not possible to multiply a 2 × 3 and a 2 × 3 matrix.

(b)

$$
\begin{aligned}
A^T B &= \begin{bmatrix} 1 & 2 \\ 4 & -1 \\ -3 & 3 \end{bmatrix} \begin{bmatrix} -2 & 0 & 5 \\ 0 & -1 & 4 \end{bmatrix} \\
&= \begin{bmatrix} -2+0 & 0+-2 & 5+8 \\ -8+0 & 0+1 & 20+-4 \\ 6+0 & 0+-3 & -15+12 \end{bmatrix} \\
&= \begin{bmatrix} -2 & -2 & 13 \\ -8 & 1 & 16 \\ 6 & -3 & -3 \end{bmatrix}
\end{aligned}
$$

(c)

$$
\begin{aligned}
AB^T + C^{-1} &= \begin{bmatrix} 1 & 4 & -3 \\ 2 & -1 & 3 \end{bmatrix} \begin{bmatrix} -2 & 0 \\ 0 & -1 \\ 5 & 4 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} -17 & -16 \\ 11 & 13 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \\
&= \begin{bmatrix} -16 & -16 \\ 11 & \frac{27}{2} \end{bmatrix}
\end{aligned}
$$

The results can be confirmed by running the following code:

```
# file: hw0.math.py
# author: Brendon Boldt
# version: 1.1
# date: Sep/11/2017
#
# This file demonstrates using Numpy and TensorFlow for basic
# computations in linear algebra

import numpy as np
import tensorflow as tf

# Create the necessary arrays in Numpy format
a = np.array([[1,4,-3],[2,-1,3]], dtype=np.float64)
b = np.array([[-2,0,5],[0,-1,4]], dtype=np.float64)
c = np.array([[1,0],[0,2]], dtype=np.float64)

# Demonstrate the computation using Numpy
print("numpy\n")
print("a^T␣b␣=\n%s\n" % (np.matmul(np.transpose(a), b)))
print("a␣b^T␣+␣c^-1␣=\n%s\n" % (np.matmul(a, np.transpose(b)) + np.linalg.
    ↪ inv(c)))

# Create the TensorFlow graph
# TF graph can be created directly from Numpy arrays
atb = tf.matmul(tf.transpose(a), b)
abt = tf.matmul(a, tf.transpose(b))
with tf.Session() as sess:
        # Demonstrate the computation using TensorFlow
        print("tensorflow\n")
        print("a^T␣x␣b␣=\n%s\n" % (atb.eval()))
        print("a b^T + c^-1 =\n%s\n" % ((abt+tf.matrix_inverse(c)).eval())
            ↪ )
```

```
> python3 hw0.math.py
numpy

a^T b =
[[ -2.   -2.   13.]
 [ -8.    1.   16.]
 [  6.   -3.   -3.]]

a b^T + c^-1 =
[[-16.   -16. ]
 [ 11.     13.5]]

tensorflow

a^T x b =
[[ -2.   -2.   13.]
 [ -8.    1.   16.]
 [  6.   -3.   -3.]]

a b^T + c^-1 =
[[-16.   -16. ]
 [ 11.     13.5]]
```

5 Solution to Problem 4

Suppose that random variable X $N(2,3)$. What is the expected value of X.

The expected value of X is 2 since the normal distribution, which is symmetric, described by X is centered about 2.