***SPIDER TRONIX***

Signal processing and Machine Learning

Name :- Anuj Agrawal

Roll.no :- 107119020

IMAGE PROCESSING

1.B

2. D

3. a) C

b)C

4. B

5. B

6.C

7.A

Audio processing

1.C

The most common **standard sampling rate** for digital **audio** (the one used for CDs) is 44.1 kHz, giving us a **Nyquist frequency** (defined as half the **sampling rate**) of 22.05 kHz. If we use lower **sampling** rates, for example, 20 kHz, we can't represent a sound whose **frequency** is above 10 kHz.

2. D

Machine learning

1. A) Logistic Regression

B) Linear Regression

C) Linear Regression

D) Logistic Regression

1. theta=[-0.0373,4.0576]

X = [1,1;1,2;1,6;1,4]

Y = [4;8.1;24.3;16.2]

theta = [0;0]

m = length(X)

J = (1/(2\*m))\*(sum((X\*(theta)-Y).^2))

theta1 =[600;600];

alpha=0.01;

i=1

while (theta1 ~= theta)

theta1 =theta;

theta = theta - ((alpha)/m)\*X'\*(X\*(theta)-Y);

J(i)= (1/(2\*m))\*(sum((X\*(theta)-Y).^2));

end

theta

J = (1/(2\*m))\*(sum((X\*(theta)-Y).^2))

1. A) Multi-Class Classification

B) Multi-Label Classification

C) Multi-Class Classification

D) Multi-Label Classification

1. A) It is case of under fitting as most of point are far away from hypothesis curve
   1. B) It is case of perfect fitting because there is small distance between point and our hypothesis .
   2. c) It is case of over fitting because our hypothesis pass through every test case and seems to parameter equal to test case so it is not perfect case
2. 1. a) Tokenizer Class - This class allows to vectorize a text corpus, by turning each text into either a sequence of integers (each integer being the index of a token in a dictionary) or into a vector where the coefficient for each token could be binary, based on word count, based on tf-idf...

b) one\_hot function - Keras provides the one\_hot() function that you can use to tokenize and integer encode a text document in one step. The name suggests that it will create a one-hot encoding of the document, which is not the case.

1. Yes, the lengths need to be the same. This is done by padding and truncating.
   * 1. Padding: The pad\_sequences() function in the Keras deep learning library can be used to pad variable length sequences.
     2. Trimming: The length of sequences can also be trimmed to a desired length.The desired length for sequences can be specified as a number of timesteps with the “maxlen” argument.
2. Yes, Scikit learn only handles real numbers. So you need to do something like one hot encoding where n numerical dimensions are used to represent membership in the categories. If you just pass in strings they'll get cast to floats in unpredictable ways. There are many ways to encode categorical variables for modeling, although the three most common are as follows:
3. **Integer Encoding**:​ Where each unique label is mapped to an integer.
4. **One Hot Encoding**:​ Where each label is mapped to a binary vector.
5. **Learned Embedding**:​ Where a distributed representation of the categories is learned.

In Neural Networks, the variable class is found by the systematic firing of neurons and also compared with the dataset which backtracking during training.

In SVMs, and Logistic Regression, the Class ID is used and the correct target class is found. Hence, Encoding the labels is always a good pratice.

7.

import numpy as np

from numpy.linalg import inv

A =np.matrix([[2, 1],[2, -1],[7, 3]]) B=np.matrix([-1,-1,8])

y = np.dot( np.dot( inv(np.dot( A, np.transpose(A), out=None)),np.transpose(A), out=None))

,np.transpose(B), out=None) print(y)

8 . Matlab code

X = [1,0;1,1;1,2;1,3]

Y = [1;2;5;6]

theta = [0;0]

m = length(X)

J = (1/(2\*m))\*(sum((X\*(theta)-Y).^2))

theta1 =[600;600];

alpha=0.01;

i=1

while (theta1 ~= theta)

theta1 =theta;

theta = theta - ((alpha)/m)\*X'\*(X\*(theta)-Y);

J(i)= (1/(2\*m))\*(sum((X\*(theta)-Y).^2));

end

theta

J = (1/(2\*m))\*(sum((X\*(theta)-Y).^2))

Theta =0.8,1.8