Scenario Questions

Ans 1

First, Let's see the nature of the problem. We have an input article, as a feature, and we have a corresponding category, as a label.

Since the input article is a string, we need to find a way to change it to a ***numeric***.

Method A:

- What if we used a ***Label Encoding*** technique and change both the features and labels into a numeric? The problem with this is that since the dimension of the input is high, we might ended up from 0 to billion and billions of numbers to encode it, which would result in ***high variance, hence skewed distribution of the feature set.***

Method B:

- What if we use ***tokenization*** technique and encode the whole articles as 0s and 1s in the corresponding word match and use that as an input feature. This is a typical method in NLP to eliminate variance. This would result in a normalized feature set hence improved accuracy.

Ans 2

Here are the steps to follow when dealing with such problems

- Initially Logistic regression is used as our model and we believe the performance could be improved. In this step we have to do two things
  - Use multiple performance measurement techniques and reason out why it is not performing well.
    - If the model is underfitting – try to gather up additional data set if available
    - If the model is overfitting – try to:
      - Use regularization
      - Pre-process the data to lower the variance

- If we found out that the model is underfitting the data even after adding some data set, it is time to use non-linear models to capture the non-linear relationship among the data set and labels.
  - In such case, we might try
    - KNeighborsClassifier
    - SVM
    - NN
- For production environment, we might need to use an online learning techniques so that the model would learn on the fly
- The problem in such case is that the model might suffer from catastrophic forgetting - **abruptly and drastically forget previously learned information upon learning new information.** We need to tune the hyper-parameter learning rate to an optimal value. If it is high, it will ended up forgetting previous information and adapting quickly whereas setting it to a low value might favor previous information while ignoring current trends. So, what to do? Set up a feedback loop and change the value of learning rate accordingly.

***** Read and summarize Lesson 4**

In lesson 4, the following main points about ANN is covered

- Applications of ANN
  - image and speech recognition
  - NLP
- Basics of NN
  - Input layer, Hidden layer, Output layer
- Perceptron
- Gradient based Learning
- Common cost function in ANN – MSE, Cross-entropy
- Convergence criteria
- Loss function
- Activation function – sigmoid, tanh, Relu, softmax
- Backpropagation