Backpropagation  
The softmax function has the desirable property that it outputs a probability distribution, and is often used as an activation function in many classification neural networks. Consider a 2-layer neural network for K-class classification using softmax activation and cross-entropy loss, as defined below:

where the model is given input of shape , and one-hot encoded label . Assume that the hidden layer has nodes, i.e. is a vector of size . Recall, the softmax function is computed as follows:

where   
2.1  
What are the shapes of ? If we were vectorizing across examples, i.e. using a batch of samples as input, what would be the shape of the output of the hidden layer?  
2.2  
What is ? Simplify your answer in terms of element(s) of .  
2.3  
What is , for ? Simplify your answer in terms of element(s) of .

Backpropagation  
The softmax function has the desirable property that it outputs a probability distribution, and is often used as an activation function in many classification neural networks. Consider a 2-layer neural network for K-class classification using softmax activation and cross-entropy loss, as defined below:

where the model is given input of shape , and one-hot encoded label . Assume that the hidden layer has nodes, i.e. is a vector of size . Recall, the softmax function is computed as follows:

where

2.4  
Assume that the label has 1 at its entry, and 0 elsewhere. What is ? Simplify your answer in terms of . (Consider both cases where and ).  
2.5  
What is ? Refer to this result as .  
2.6  
What is ? Refer to this result as .  
2.7  
Denote with . What is and ? You can reuse notations from previous parts.  
2.8  
To avoid running into issues with numerical stability, one trick may be used when implementing the softmax function. Let be the of , then

What is the numerical problem with the initial softmax computation? Why the modified formula would help to resolve that problem?