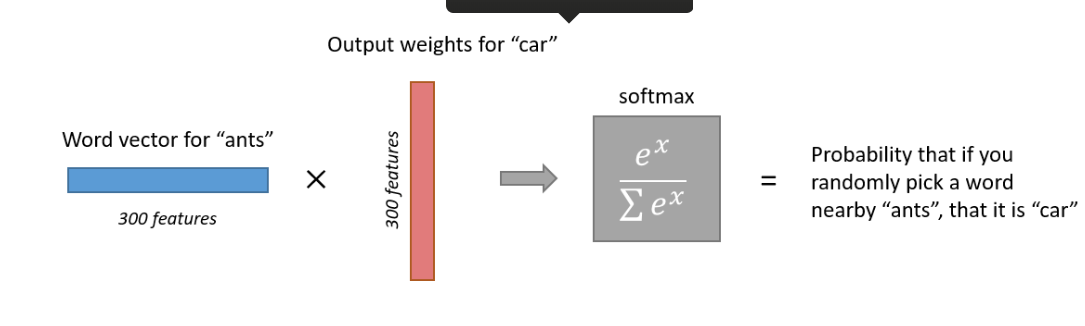
Assignment 2



Notations and Notes -

• Both the input vector x and the output y are one-hot encoded

• vw and v’w are two representations of the input word w

• vw comes from the rows of W (Here V)

• v’w comes from the columns of W’ (Here U)

• vw is usually called the input vector (Here vw)

• v’w is usually called the output vector (Here uw)

A good doc to understand the inner workings - <http://www.1-4-5.net/~dmm/ml/how_does_word2vec_work.pdf>

Objectives –

Understanding Word2Vec (Skip Gram) Model

Understanding the Architecture of the model

Understanding the loss, error, various gradients, input layer, output layer, input vector and output vector, probability of context vector given centre vector.

And

Implementing Sigmoid Function

Implementing Softmax and Negative Sampling loss and their gradients

Implementing the loss and gradients for Skip-gram model

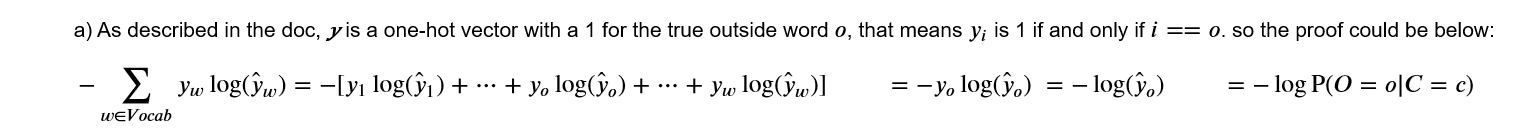
Implementing SGD Optimizer for back propagation

Lets start -

(a)

This problem can be simply solved looking at the 1 hot encoded nature of output word( context vector in this case).

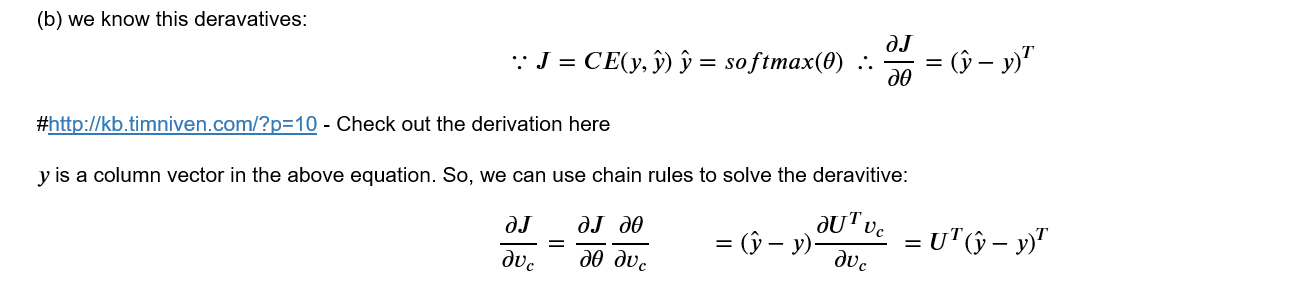
I have elaborated the same below -



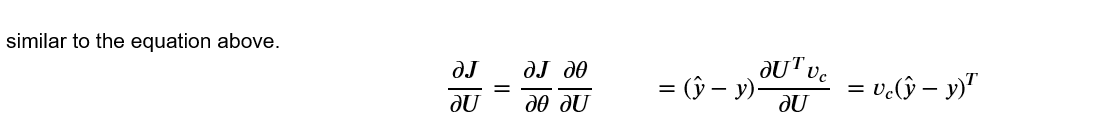
(b) J is the softmax function of theta which is applied on the output vector to get the predicted probabilities of context vectors.

We want to find the gradients of J wrt to vc which is the embedded representation of input word( center word in this case).

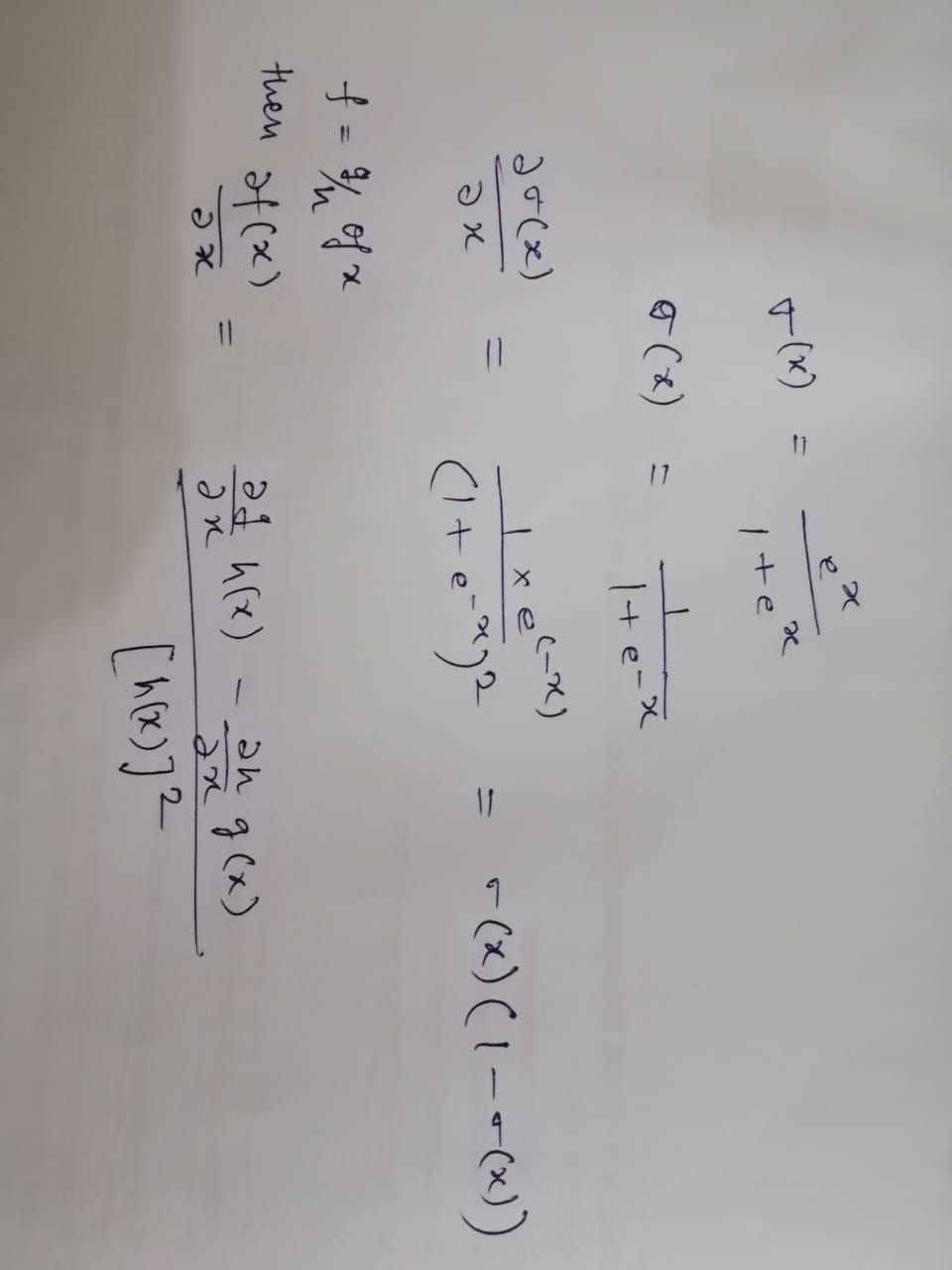
We can use chain rule to get the gradients related to both input vector here.

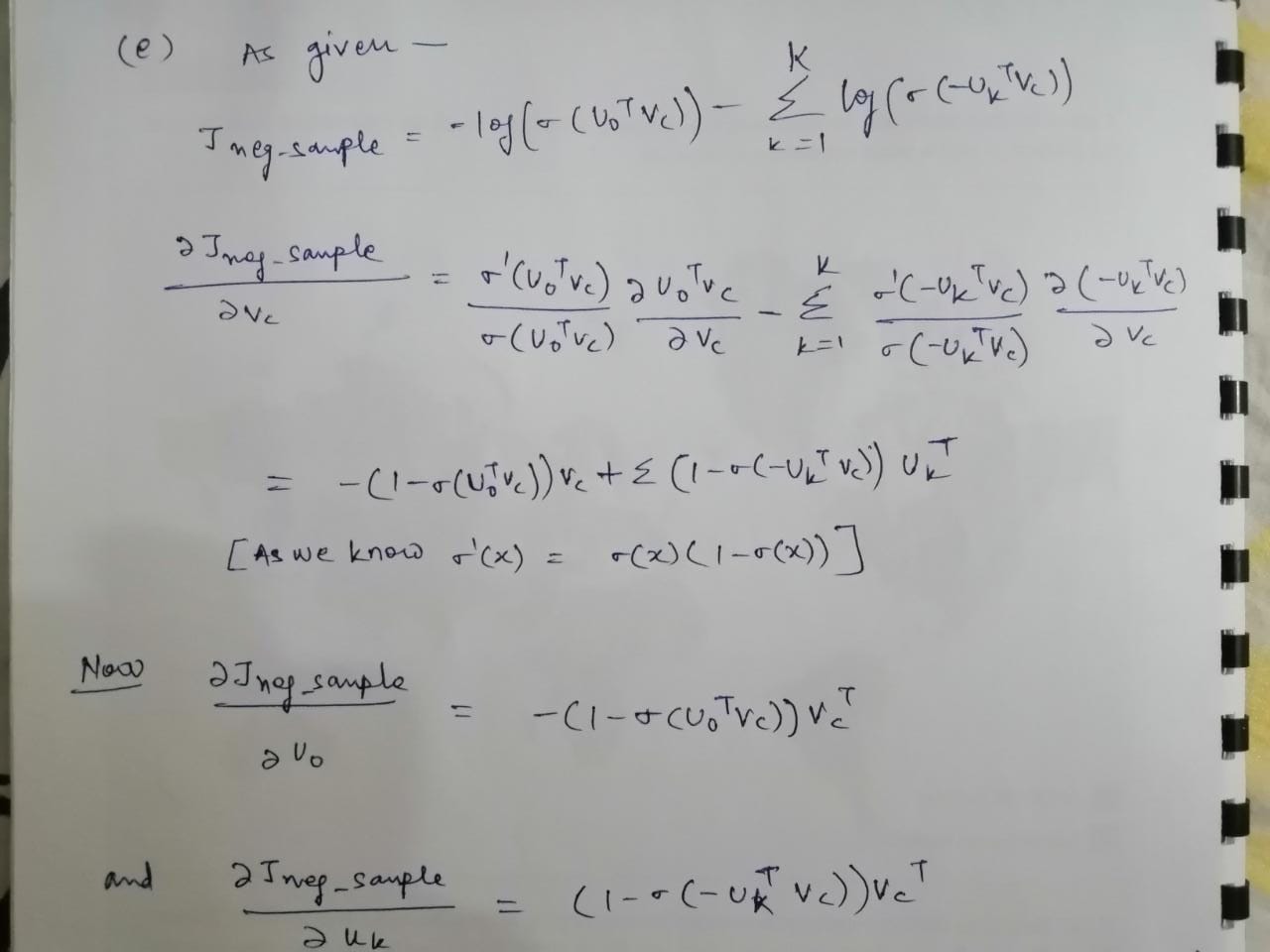


(c) In the similar way we can apply chain rule to get the gradient wrt to output vector matrix(U).

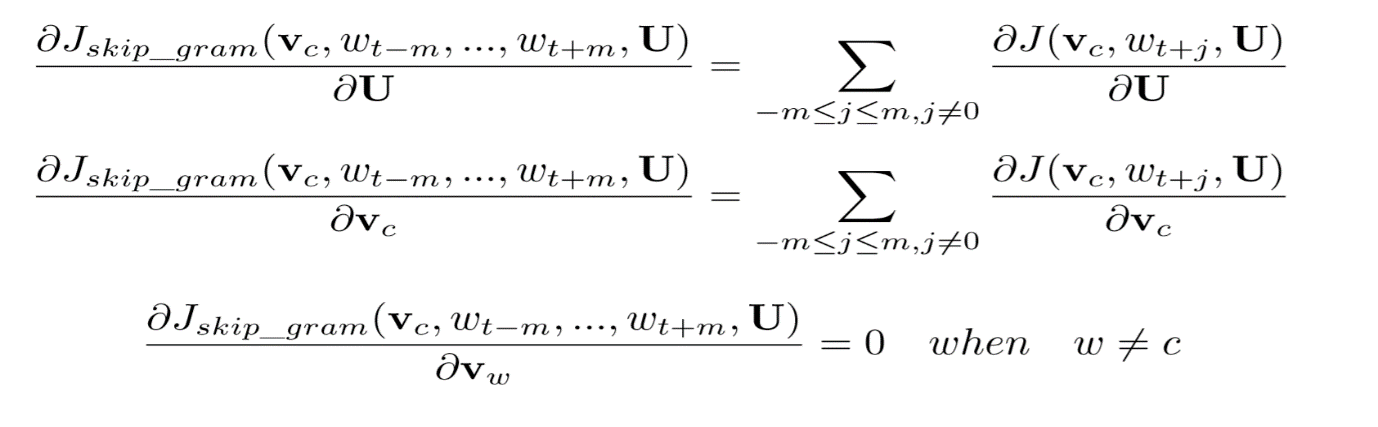


(d)



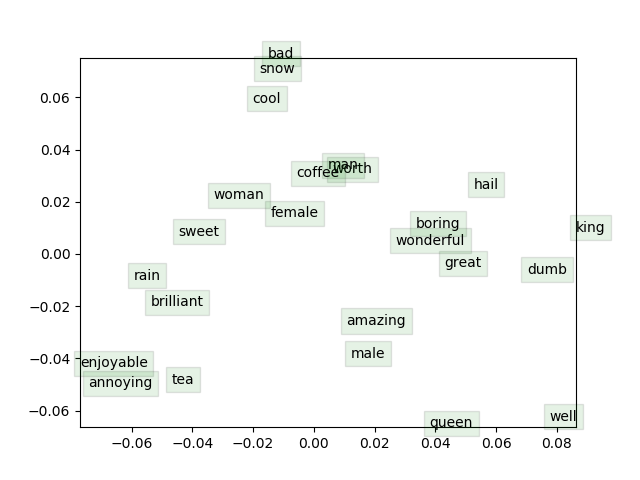
(e) 

(f)



**Coding Implementation** -

c) Last Part of code after only 1000 iterations out of 40000 iterations -



Helpful Links - <https://stats.stackexchange.com/questions/253244/gradients-for-skipgram-word2vec>

<https://courses.cs.ut.ee/MTAT.03.277/2015_fall/uploads/Main/word2vec.pdf>

<https://deepnotes.io/softmax-crossentropy>

<https://math.stackexchange.com/questions/945871/derivative-of-softmax-loss-function>

<http://www.claudiobellei.com/2018/01/06/backprop-word2vec/>

<https://medium.com/explore-artificial-intelligence/word2vec-a-baby-step-in-deep-learning-but-a-giant-leap-towards-natural-language-processing-40fe4e8602ba>

<https://towardsdatascience.com/an-implementation-guide-to-word2vec-using-numpy-and-google-sheets-13445eebd281>

<https://nathanrooy.github.io/posts/2018-03-22/word2vec-from-scratch-with-python-and-numpy/>

<https://cambridgespark.com/4046-2/>

<http://kb.timniven.com/?p=10>