标准化流(Normalizing Flow)教程（一）

<https://www.jianshu.com/p/66393cebe8ba>

<https://zhuanlan.zhihu.com/p/59615785>

**element-wise 相乘的直观意义是什么 https://www.zhihu.com/question/65866370**

elementwise multiplication 直白翻译过来就是元素的智能乘积。例如    表示对每一个输入向量 v 乘以一个给定的“权重” w 向量。换句话说，就是通过一个乘子对数据集的每一列进行缩放。这个转换可以表示为如下的形式：

# 对角矩阵的行列式为对角线元素乘积

* 一对角矩阵 diag(a1, ..., an) 的[行列式](https://zh.wikipedia.org/wiki/%E8%A1%8C%E5%88%97%E5%BC%8F" \o "行列式" \t "_blank)为a1...an的乘积。

# 上三角矩阵的行列式为对角线元素相乘

# 理解 Word2Vec 之 Skip-Gram 模型

<https://zhuanlan.zhihu.com/p/27234078>

**softmax交叉熵损失函数及其求导**

<https://blog.csdn.net/fsdfasfawre/article/details/80586123#comments_10451692>

# skip-gram 梯度求导

https://blog.csdn.net/u010665216/article/details/78721354

# word2vec求导

<https://zhuanlan.zhihu.com/p/53425736>

<https://plmsmile.github.io/2017/11/02/cs224n-lecture2-word2vec/>

textbook 求导：

<http://d2l.ai/chapter_natural-language-processing-pretraining/word2vec.html#the-skip-gram-model>

<http://d2l.ai/chapter_natural-language-processing-pretraining/approx-training.html>

Because the softmax operation has considered that the context word could be any word in the dictionary V, the loss mentioned above actually includes the sum of the number of items in the dictionary size.

From the last section, we know that for both the skip-gram model and CBOW model, because they both get the conditional probability using a softmax operation, the gradient computation for each step contains the sum of the number of items in the dictionary size.

Huffman softmax

<https://www.cnblogs.com/pinard/p/7160330.html>

最先优化使用的数据结构是用霍夫曼树来代替隐藏层和输出层的神经元，霍夫曼树的叶子节点起到输出层神经元的作用，叶子节点的个数即为词汇表的小大。 而内部节点则起到隐藏层神经元的作用

一般得到霍夫曼树后我们会对叶子节点进行霍夫曼编码，由于权重高的叶子节点越靠近根节点，而权重低的叶子节点会远离根节点，这样我们的高权重节点编码值较短，而低权重值编码值较长。这保证的树的带权路径最短，也符合我们的信息论，即我们希望越常用的词拥有更短的编码。

在word2vec中，约定编码方式和上面的例子相反，即约定左子树编码为1，右子树编码为0，同时约定左子树的权重不小于右子树的权重。

<https://www.cnblogs.com/pinard/p/7243513.html>

word2vec里面最大的问题在于从隐藏层到输出的softmax层的计算量很大，因为要计算所有词的softmax概率，再去找概率最大的值。这个模型如下图所示。其中𝑉是词汇表的大小

由于我们把之前所有都要计算的从输出softmax层的概率计算变成了一颗二叉霍夫曼树，那么我们的softmax概率计算只需要沿着树形结构进行就可以了。如下图所示，我们可以沿着霍夫曼树从根节点一直走到我们的叶子节点的词𝑤2

使用霍夫曼树有什么好处呢？首先，由于是二叉树，之前计算量为𝑉,现在变成了𝑙𝑜𝑔2𝑉。第二，由于使用霍夫曼树是高频的词靠近树根，这样高频词需要更少的时间会被找到，这符合我们的贪心优化思想。

GAN:

<https://analyticsindiamag.com/hands-on-guide-to-deep-convolutional-gan-for-fashion-apparel-image-generation/>

<https://github.com/SK7here/DCGAN-Fashion-MNIST/blob/master/DCGAN-Model.ipynb>