## 一、理论结果

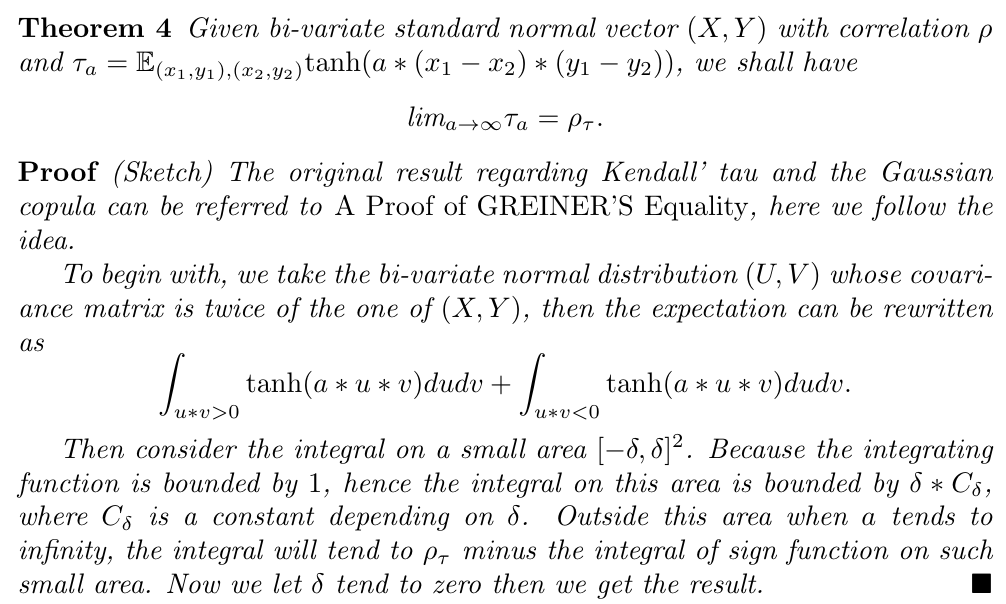
### 1、copula的估计

我们之前提出用Kendall’s tau来估计Meta Gaussian distribution的Gaussian copula，在实际应用中，对于变量X=(X_1,X_2)的Kendall’s tau的定义为

\rho_\tau = \mathbb{E}_{(x_1,x_2), (x_1^\prime, x_2^\prime)} \text{sign}((x_1 - x_1^\prime) * (x_2 - x_2^\prime))，

其中sign函数虽然能在Pytorch中直接调用，但是其梯度几乎处处为零，因此不适用于神经网络。

为了克服这一点，我们采用tanh()来改进。特别地，可以证明，对于正实数a，其增加时，积分\mathbb{E}_{(x_1,x_2),(x_1^\prime,x_2^\prime)} \text{tanh}(a*(x_1-x_1^\prime)*(x_2-x_2^\prime))将收敛于Kendall’s tau。

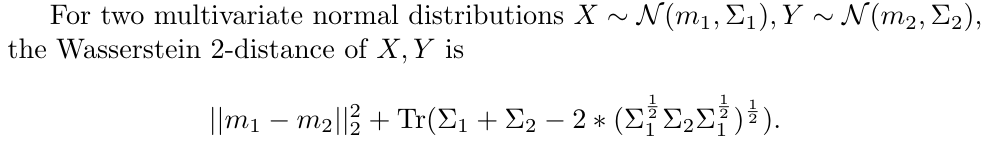


### 2、以高斯分布为例的distance计算

### KL divergence

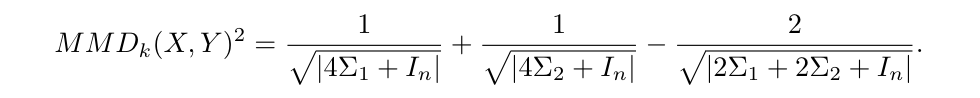
### Wasserstein distance

Wasserstein 1-distance 没有关于高斯分布distance的显示表达式；但是对于Wasserstein 2-distance,我们有



### Gaussian kernel MMD

假设kernel function为k(x,y) = \text{exp}(-||x-y||_2^2)，对于两个均值相同的高斯分布X \sim \mathcal{N}(0,\Sigma_1), Y \sim \mathcal{N}(0,\Sigma_2)，我们有



## 二、实验

我们对Meta Gaussian distribution做了一些实验。具体地，首先对于源域用三维高斯分布生成三个相关的变量，再按照不同的边缘分布，生成边缘分布分别为正态分布a、t-分布b以及Gamma分布c的Meta Gaussian分布。源域的标签是norm.cdf(2-a-b-c)。之后对于目标域，用同样的三维高斯分布生成a、b、c后，a乘以一个常数，b不变，c变为5\*e^{-c}，之后标签函数不变。