日报8

Rnn的核心：用上上一刻的结果（之前），与这一刻的信息。

Transformer中用self-attention来得到输入（之前的结果总结）。

（BPTT）

SRN

Lstm：输入输出遗忘，记得输入输出都要有tanh。

Gru：两个门由x与h-1决定，重置门（输出门），更新门（）门都直接由h-1（不重置），x决定，而用以预测输出的值是要tanh的。

Seq2seq：（rnn式）：每一刻需要上一刻的某个输出与这一刻的特征

编码器：输出注释向量（k，v）的组合c（可能有若干个）（相同于总结）常规rnn

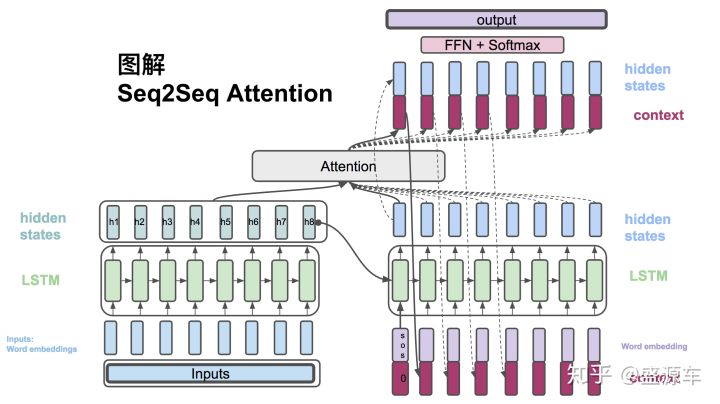
解码器：每一刻用c做输入（与上一刻的隐藏，可能有其他。）

Attention：（翻译）

编码器：输出注释向量（k，v）的组合c（可能有若干个）（相同于总结）常规rnn

解码器：先用句首预测出q0，计算score（q，k），组合v得到c0，将c0做下一刻的输入。这个例子中还有上一刻的词语做输入。第一个元有编码器的输入。

如果解码器的输入也是已知的：那可以直接用rnn读入输入再qkv。



Transformer：不递归，直接输完一个句子。

编码器

Self attention

句子中的词向量=》W映射到低维，qkv=》（qkv）n得到输出=》组合这些输出。

解码器：注意遮挡

注意self-attention编码器解码器容易让人误解为有一个尽力满足的输入的输出，这里几乎没有这样的倾向，只是为了总结上下文。

记得屏蔽右侧。

编码-解码——attention中

当然最后一个编码器（6个中的最后一个）做输入，得到kv，而解码器本身提供的最开始的输入则需要不断遮挡。且刚开始是句首标记，这样做q，实验表明这样可以更多关注整体信息，但弱化了序列信息。

Rnn：

时间神经网络（时间递归）（循环）

结构神经网络（结构递归）（递归）

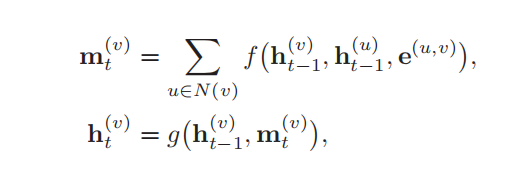
百度：递归神经网络包含循环神经网络（RNN），以及bolzman机

某些：

递归神经网络（recursive neural network）是具有树状阶层结构且网络节点按其连接顺序对输入信息进行[递归](https://baike.baidu.com/item/%E9%80%92%E5%BD%92/1740695" \t "https://baike.baidu.com/item/%E9%80%92%E5%BD%92%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C/_blank)的[人工神经网络](https://baike.baidu.com/item/%E4%BA%BA%E5%B7%A5%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C/382460" \t "https://baike.baidu.com/item/%E9%80%92%E5%BD%92%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C/_blank)

结构递归神经网络：RBM,BM,HOP-Field，

Vae，LDA.递归自编码机

图神经网络： 

自适应学习率

Adagrad：梯度平方和

**RMSprop：**梯度幅值指数加权平均（均方根调整）

**AdaDelta：**梯度幅值指数加权平均**（1-t）**，**参数更改量的指数平均（1-t-1）**

1. **beta）\*x（t-1）+（beta）\*sum（t-1）递归的一般乘的最多。**

**Adam梯度幅值平方加权平均，梯度加权平均，乘以α，这是为了自适应动量。**

方向调整：

动量：梯度方向的指数加权平均，

可以看作先一次惯性部分。（ro）（seta中）

再一次新的部分。（α）（这里用的seta（t-1））（似乎有些不合理）

Nesterov：新部分用seta（seta中）

Adam：指数加权平均前期小

梯度的指数加权平均/正则（beta1）

梯度幅值的指数加权平均/正则（beta2）（beta2会大一点）

还有一个α

# 深度网络中的超参数：

优化类：

批量（与batch-normalization）

学习率（如指数加权平均）（初始值）

更新方向相关（如指数加权平均的加权系数）

梯度截断：按值截断，（分量大小）

按模截断。（模截断到超参数b）

初始化超参数。全初始化为0是可行的（从w梯度意义上甚至是很好的初始值），但对更深层初期会出现对称权值现象。后面才会慢慢改动，所以不是很好。

一般截断高斯分布

初始化类：

随机区间

如果参数太小，

一是会导致神经元的输入过小，经过多层之后信号就慢慢消失了；二是还会使得

Sigmoid型激活函数丢失非线性的能力。以Logistic函数为例，在0附近基本上是

近似线性的。这样多层神经网络的优势也就不存在了。如果参数取得太大，会导

致输入状态过大。对于 Sigmoid 型激活函数来说，激活值变得饱和，从而导致梯

度接近于0。

**Xavier** 初始化

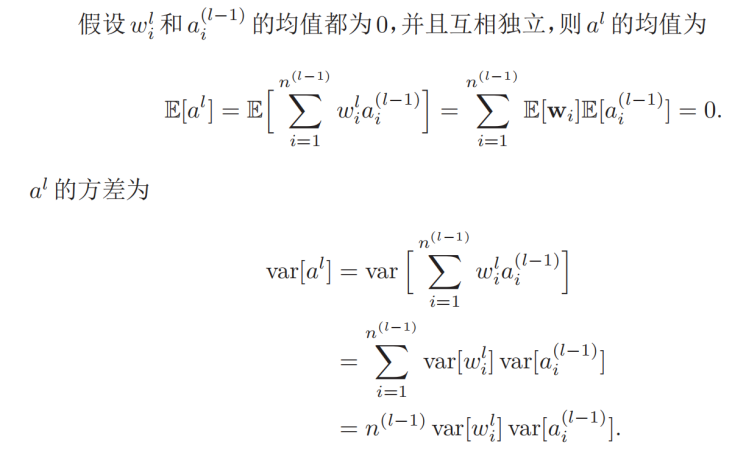
输入信号的方差在经过该神经元后被放大或缩小了*n*

(*ll*1)

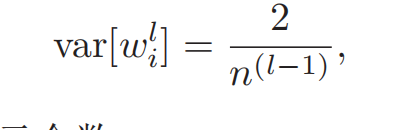
var[*w*

*l*

*i*]倍。



误差信号反向传播又会放大nl所以折中。

Relu时候：

当多变量求导时出现在同一级，求一个时其他当常，比如求偏k时其他的偏都是常数。

预处理：（结构类。）

白化，（pca），标准化，缩放

Batch-normalization在仿射函数之后做，因为我们最关心激活函数。（激活函数丢到0-1之间）这样分布也不会被仿射函数移动的太厉害。方差均值是整个的

注意对会造成影响。。并且引入了一些序列网络的性质（因为普通网络样本之间不会有影响这里是有的）

训练完成后用整个训练集的统计回去。。。

Layer-normalization（相对少。）

权值normalization

侧抑制

*Dropout*

2选1：

经过上面屏蔽掉某些神经元，使其激活值为0以后，我们还需要对向量y1……y1000进行缩放，也就是乘以1/(1-p)。如果你在训练的时候，经过置0后，没有对y1……y1000进行缩放（rescale），那么在测试的时候，就需要对权重进行缩放，操作如下。

1. 在测试模型阶段预测模型的时候，每一个神经单元的权重参数要乘以概率p。

超参数优化：

网格搜索

随机搜索

贝叶斯优化：高斯过程，收益，期望改善：时序模型优化，期望改善。

神经架构搜索

动态资源分配

在训练时，激活神经元的平均数量为原来的*p*倍。而在测试时，所有的神经元都是可以激活的，这会造成训练和测试时网络的输出不一致。为了缓解这个问题，在测试时需要将神经层的输入**x** 乘以*p*

*Nlp：总体领域。*

*再重点从文本挖掘入手。*

*文本挖掘：  
几个领域*

*机器学习下的领域。。。：迁移学习。集成学习。深度学习。*

*推荐书籍。*

*领域：*

*信息抽取：*

*知识图谱：NER，RE*

*事件检测：ED*

*半监督，有监督，无监督，*

*中文还是英文。*

*策略：基于规则。*

*端到端。如果这个任务本身有特点，自己加相应的结构。*

*强化学习：*

*有向图最长路径问题（只许你走多少步）。*

*概率（非确定性）有限自动机。（当让r=p即语法解析。）*

*Keras*

*从应用上看。从数据上看。*

线索：目标强化学习，多任务学习。

现有运算体系对有些逻辑不友好

大头：对话系统。

Tensor("activation\_188/Relu:0", shape=(?, 7, 7, 1056), dtype=float32)

769

Press any key to continue . . .

Tensor("activation\_188/Relu:0", shape=(?, 7, 7, 1056), dtype=float32)

769

Press any key to continue . . .

Zero-shot learning ，常识，life-long learning

一阶逻辑。

Few-shot learning

转导迁移学习。

Vae

不要企图在单机上跑大型模型

Acl Anthology

Kdd被AI统治的比acl还纯粹。。。

VAE，GAN，zero-shot挖掘

常识，life-long learning

一阶逻辑。

Few-shot learning

转导迁移学习。

* 机器学习顶级会议：NIPS, ICML, UAI, AISTATS（期刊：JMLR, ML, Trends in ML, IEEE TNN）
* 计算机视觉和图像识别：ICCV, CVPR, ECCV（期刊：IEEE T-PAMI, IJCV, IEEE TIP）
* 人工智能：IJCAI, AAAI（期刊AI）
* 另外相关的还有SIGGRAPH, KDD, ACL, SIGIR, WWW等
* 特别是，如果做机器学习，必须地，把近4年的NIPS, ICML翻几遍；如果做计算机视觉，要把近4年的ICCV, CVPR, NIPS, ICML翻几遍

完全自力更生 –读发表在重要刊物和会议上的有关你的topic的文献 • 2、30篇读下来，你大概能知道有哪些问题是没解决的了 • 关注这个topic上活跃的leading expert的工作、他们的文章中可能 会指出一些需要解决的重要问题 –兴趣 –有价值的问题 –知识结构 –资源 –宜“小题大做”，忌“大题小做

## 在带inception结构的网络迁移学习只剩下最后一层时会出现伪过拟合现象acc可以逐步增高，但val——acc降低。

#### 49是Xception，而50是res

坑：

多类别。

有些真的很像。

Effcientnet一个大系列，建议穷举（以及所有你能见到的好网络。）

模型融合。集成学习那一套，或多任务学习那一套。

标签平滑，

TTA，

数据增强，

Batch-norm在预训练的法则，

基本是爬数据集大赛。。

投票。（可能腐败。）（因为事实上有些难认得样本是大家都难认。）

（而一个模型能认得集合可能是多数好认+一些独立得难认得集合。）

（核心在于高度不想关）（典型得是k折交叉验证。）

分类：

投票，加权投票。

平均

Layer得build(input\_shape):

这是你定义权重的地方。这个方法必须设self.built = True，可以通过调用super([Layer], self).build()完成。build，这个函数用来确立这个层都有哪些参数，哪些参数是可训练的哪些参数是不可训练的。call(x): 这里是编写层的功能逻辑的地方。你只需要关注传入call的第一个参数：输入张量，除非你希望你的层支持masking。这个函数在调用层对象时自动使用，里面就是该层的计算逻辑，或计算图了。显然，这个层的核心应该是一段符号式的输入张量到输出张量的计算过程。compute\_output\_shape(input\_shape): 如果你的层更改了输入张量的形状，你应该在这里定义形状变化的逻辑，这让Keras能够自动推断各层的形状。

写个layer。。。

Build：哪些参数

Call：如何计算

compute\_output\_shape：

Keras.save

keras.utils.multi\_gpu\_model(model, gpus)模型并行。

### **multi\_gpu\_model**

类变量，所有方法外，实例变量，\_\_init\_\_内

\_\_foo\_\_:特殊.

\_\_foo:私有。

\_foo:protected

第二维度，第一维，

Sklearn可以采用keras作为底层模型。

惊了那是集成学习的套路。/。。

cdf*×*1(*ξ*)

Deepth，分辨率，通道数。

满足一个分布的随机数。你拥有一个随机数。

积分太难算：

采样。

随机采样，

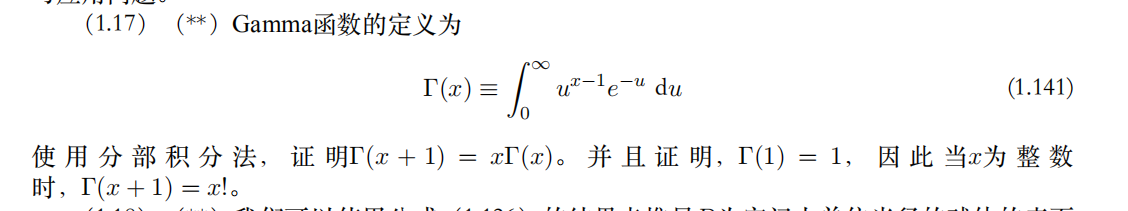
拒绝采样，（未归一的概率）接受概率：p（x）/k q（x）

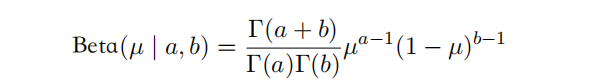
重要性采样。（需要重要性）w=p（x）/q（x）（思想很重要）

高维空间中，其采样率会非常低，导致很难应用到实际问题中。

MCMC方法的关键是如何构造出平稳分布为*p*(**x**)的马尔可夫链

生成事件。



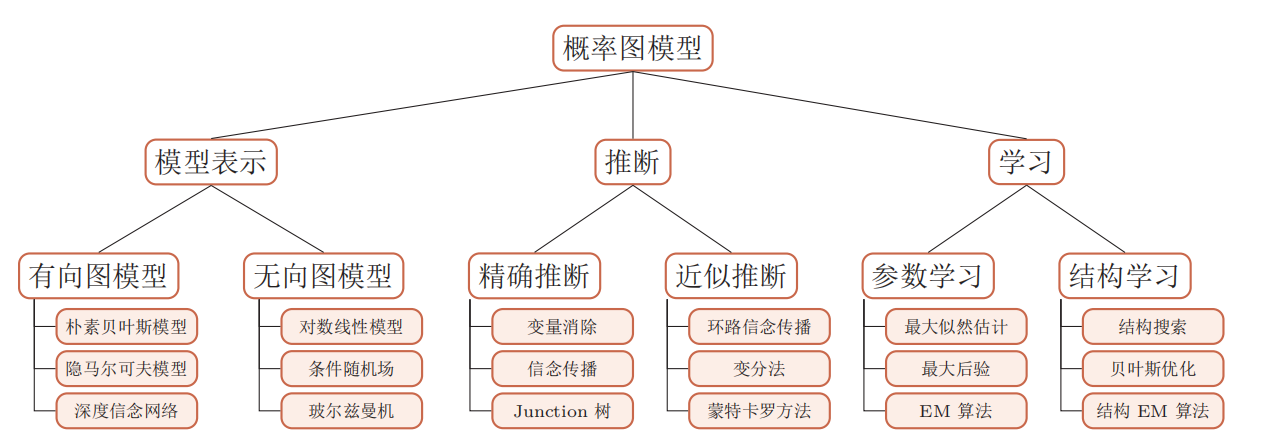


近似推断，采样。

有向图：信念网络

无向图模型

# 群论。



我们的目的其实是找到最优的selta。

泛函是将一个函数映射到一个数值的东西。典型的是积分。

常说的函数一般指的是数映射到数。

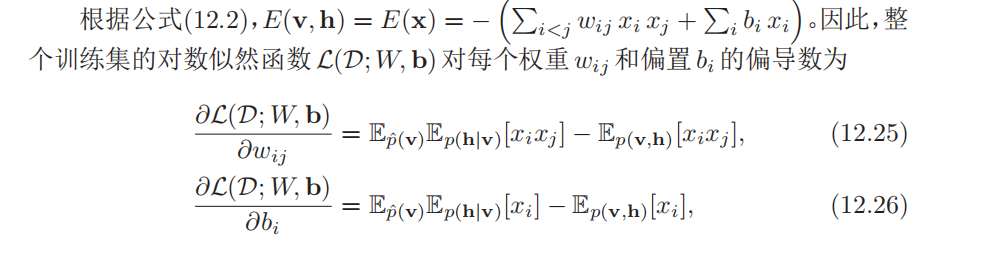
深度信念网络

深度玻尔兹曼机

玻尔兹曼机生成的数据：（样本集）

有些时候我们不是要生成一个真的数据。对于一个判断模型而言，它可能更需要一个真的分布，即使其他有相对没有那么真的数据。（出现的概率少）

玻尔兹曼机的训练：



当经验边缘期望乘以参数条件期望=参数全期望收敛（这里由于h不知道）

而那边是经验期望乘以参数条件期=经验全期望（这里由于y知道）

采样是拿来算期望的。一般无向图的不是很配直接算得出来。。

为了近似这个期望，我们可以固定住可观测变量 **v**，只对**h**进行吉布斯采样。当玻尔兹曼机达到热平衡状态时，采样*xixj* 的值。在 训练集上所有的训练样本上重复此过程，得到*xixj* 的近似期望*⟨xixj ⟩*data。公式 (12.26) 中的第二项为玻尔兹曼机在没有任何限制条件下，*xixj* 的期望。这时可 以对所有变量进行吉布斯采样。当玻尔兹曼机达到热平衡状态时，采样 *xixj* 的值，得到近似期望*⟨xixj ⟩*model。

Data的积分是通过固定v对h采样得到的

Model的积分是都不固定得到的。。

它的导数需要对状态穷举求和。。

概率图模型

深度信念网络

生成网络

EM算法的初衷是整体分布形式已知。但缺省变量。这样完整数据的分布是很容易写出的，可观测变量的极大似然一般很难写，因为涉及积分。

任务是找到合适的selta使得极大观测变量的似然。

不能很纯粹的从无到有的生成分布。

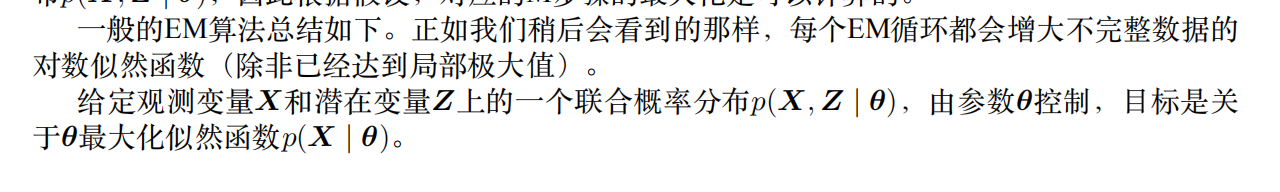
数据的极大似然 = KL散度 + 证据下界

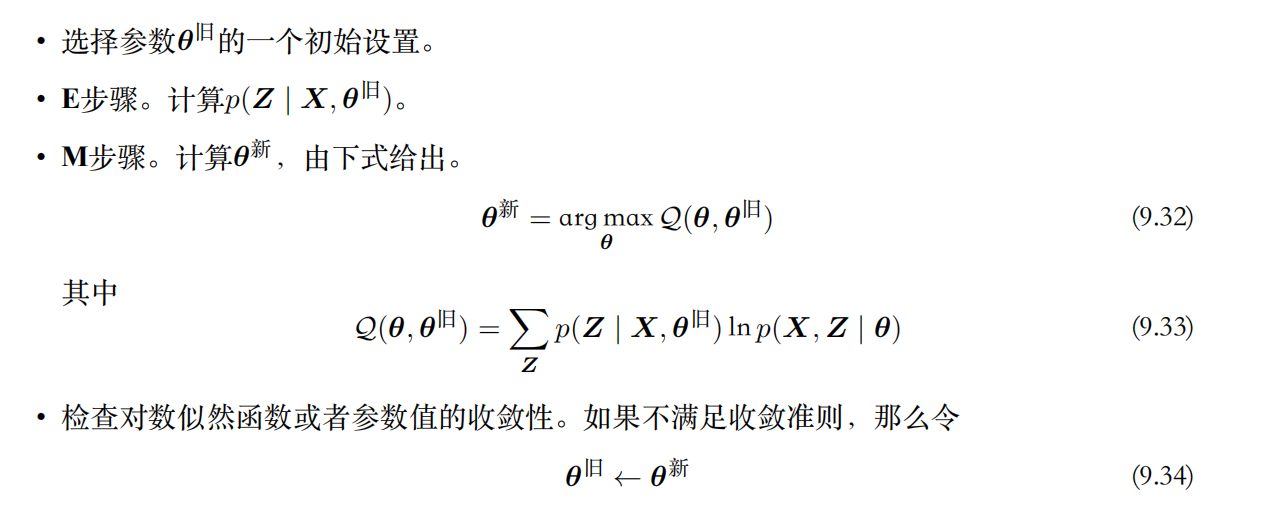
玻尔兹曼分布（0，1）的全条件概率就是sigmoid函数

EM证据下界最后收敛于真实分布。

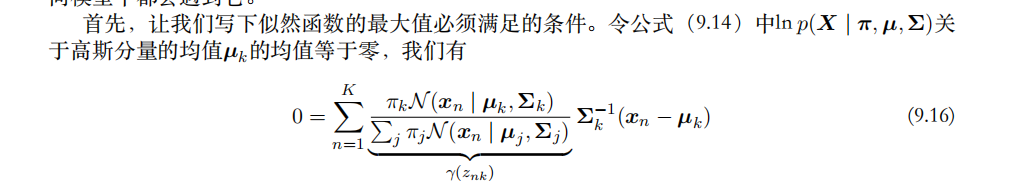
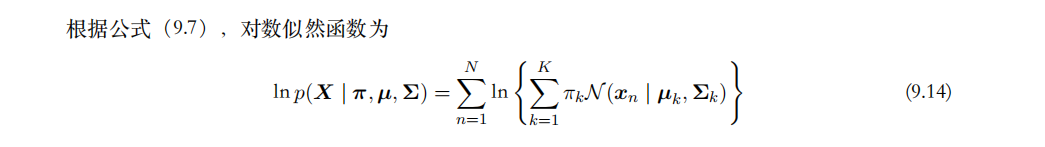
E步找到q（Z）最大化证据下界（也是一个泛函。）

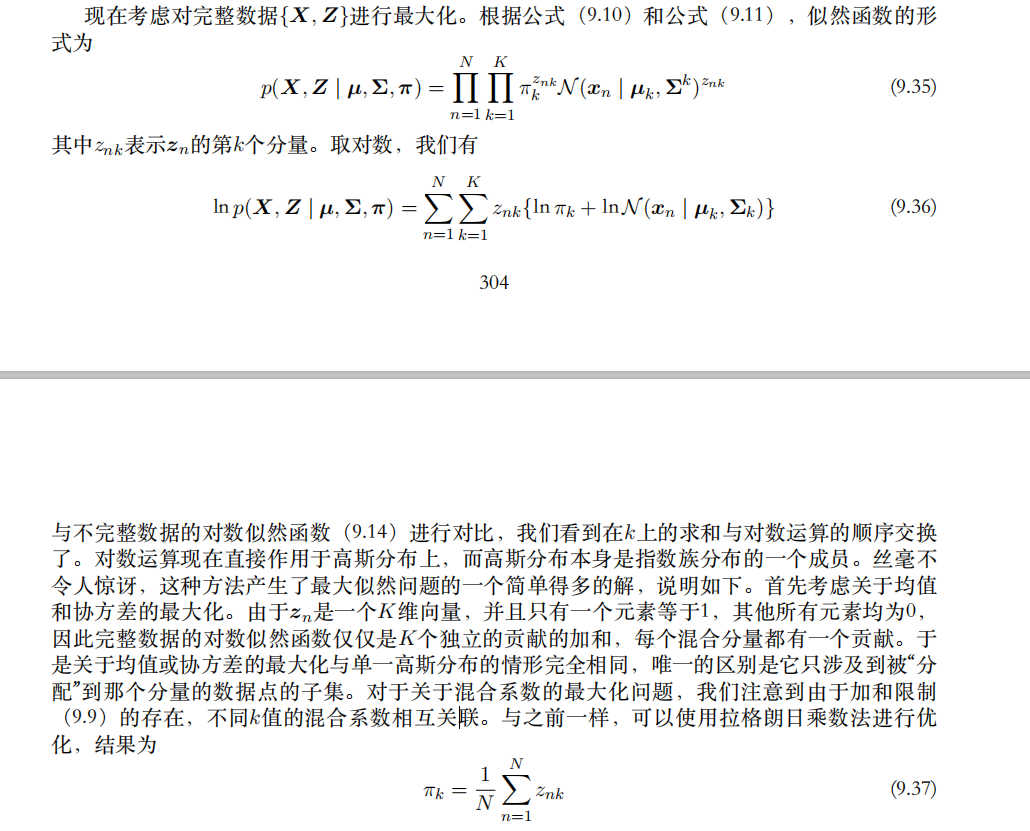
M步找到selta最大化这个泛函。（这里其实就是完整数据的学习问题。有些情况可以直接解出来，否则梯度上升）





我们确实是想极大化Q，但实际上我们最大化的是完整数据的极大似然。





变分分布，变分推断。

图模型的全概率是相对其他状态的概率。。。

观测样本来估计其分布。所以是随意一张真的图的概率，或者说任意

p（x|z）p（z）这个来求p（z|x），z可以很容易算p（x|z）

卡点：看似最优化完整数据的对数似然，而不是证据下界或者说交叉熵。。

GEM很多时候E或M会很难算，这种时候请选择梯度上升在内的其他方法。

一般都只是不知道参数而非连分布形式都不知道。如果分布未知至少要有采样能力。

变分函数，我们不断的修改q（z）来修改完整数据的分布。

变分推断：

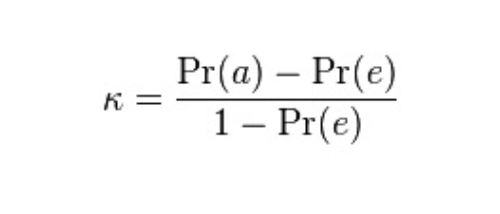
由于主题模型是个聚类问题，所以特征选区或者说学习表征是个很核心的问题。基本说每一个新的类别的（神经）词向量，新的概率推断模型，都可以衍生一个主题模型。

缺点就是后者太硬核了。。

Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. In *Proceedings of the 2017 International Conference on Learning Representations*.

事件提取本身更关注一些动词，但本身可以是动词性质的名词，有些轻动词（have，make）也一般会造成困扰（take effect，take a flight，make，have）

# Cohen's kappa评估两个标注者的差异



如何显示的表达一个网络的分布。。。并用其聚类。。。或者调节部分参数使其能拟合更多的问题

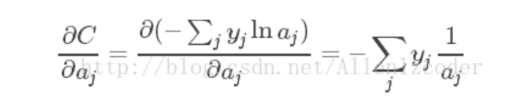
Yi是真实样本，而a是预测结果。

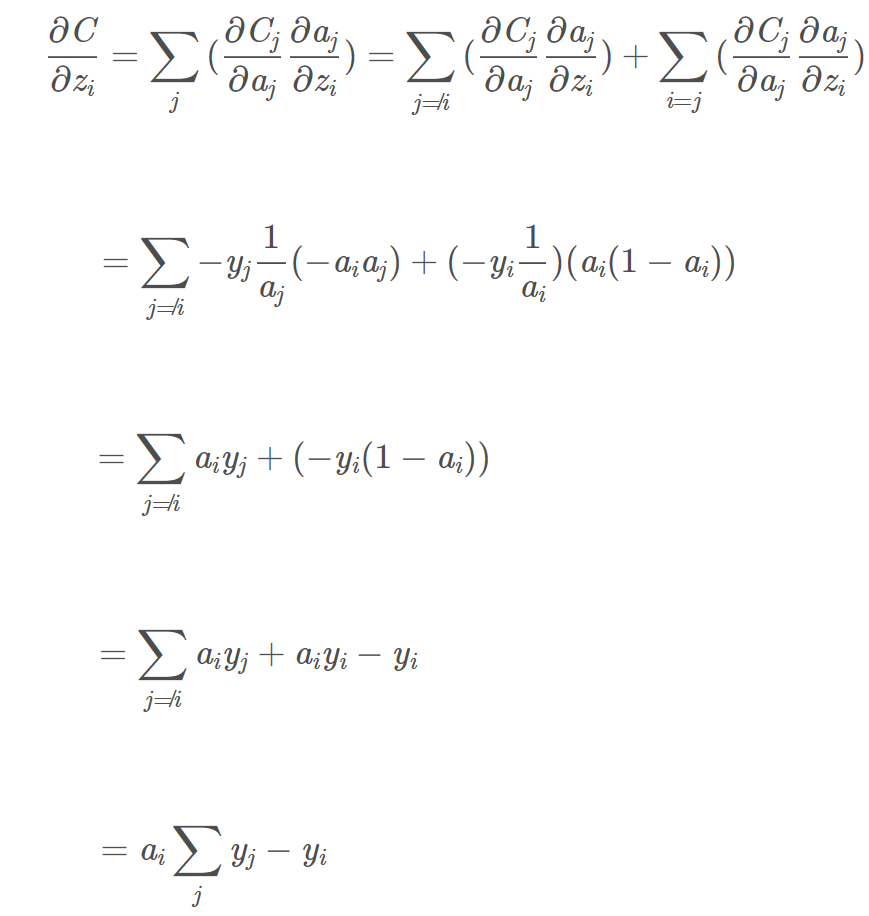
## 文本界的对抗样本。文本界的数据增强。

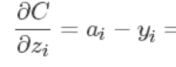
## Distill learning

## 图语料库。

## GCN，不同的分布。







Iclr上有许多。

大于0.9就不修饰的损失函数。强行写个条件应该是可行的。写的好的话也许可以投个iclr。

给论文，给数据集，去拿洗发水。

ACE2005 dataset

TAC KBP 2014 Data(LDC)

[https://tac.nist.gov//2014/KBP/Event/](https://tac.nist.gov/2014/KBP/Event/)

多次映射变换后的。

可以设想其实可以用若干张图片还原。所有这些图片都是符合的。

无偏有效一致。

无偏意味着：损失函数最低时，你的模型应该是最优或接近最优的。(收敛)

有效：最好不要让你的模型有太多的梯度弥散点。

一致：你的模型理论上用这个模型无穷样本时依概率收敛于最佳结果。（依概率收敛。）

BPP,RP,

IP（图同构问题：）

伪逆矩阵（wt\*w）-1\*wt对w而言就是伪逆矩阵

在线性方程组左乘（非满秩序的矩阵）的话解会变多。

列数大于行秩则无数解。

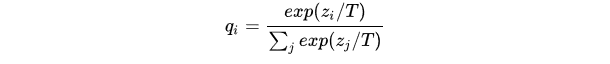
ICLR

朱：识云82，数学不擅长。。不应该指派数学任务给他。。

杨子宁：预训练网络

时：数据集。

蒸馏学习：

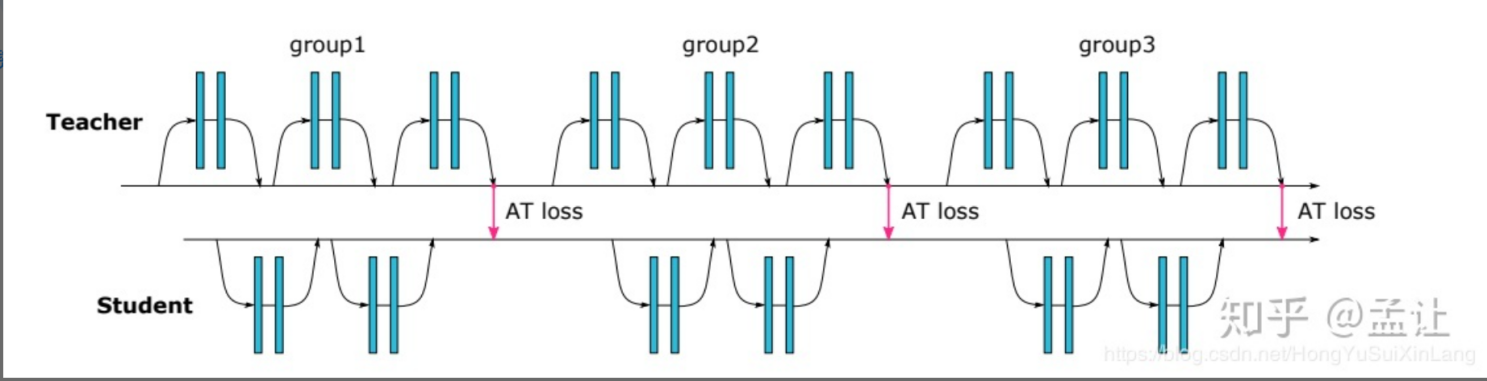


这是soft目标的输出。

硬目标：真实的label

软目标：（一个或一群）teacher教的。（能比硬 target提过更多的信息，权重应该更大。）T^2

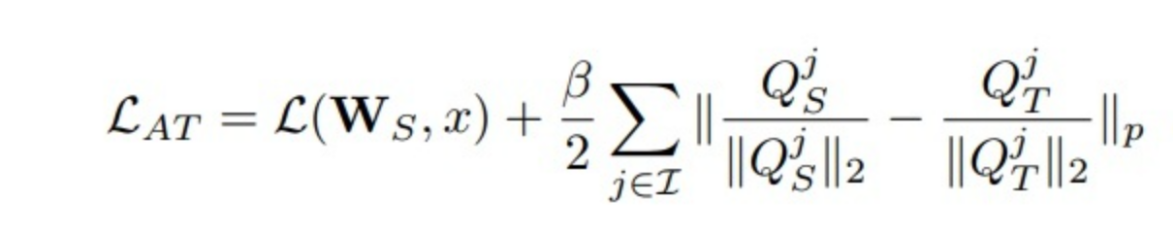
一般是加权统一。



学习特征的也是有的。名为attention transfer

现在序列网络还没看到过这种transfer

可以看成是正则，模型浓缩，半监督。



只要大于0.9就不惩罚。

CRF，NER,POS，语法解析器等。

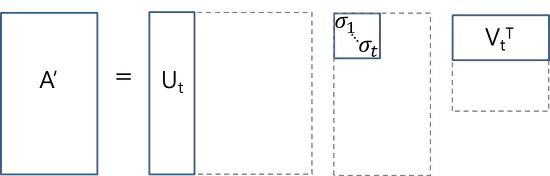
TempEx： reference date, some text, and parts of speech

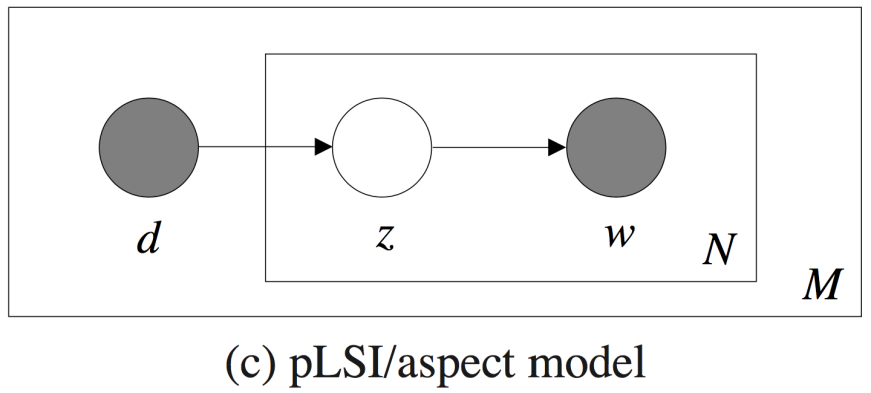
.

 LSA，PLSA,LDA

LSA（词向量流）之流都是向量相似度越高，共现数目越高。这里其实相当于选了若干主题。每个词语与若干主题相关。解决了一义多词，但不能解决一词多义。

（其实并没有对主题显示建模，而是隐式语义空间。。）（缺乏解释性。）

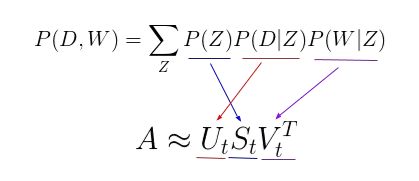


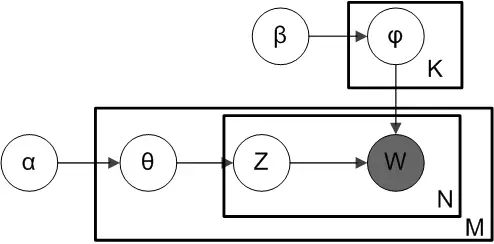
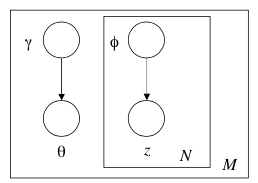


* 给定文档 d，主题 z 以 P(z|d) 的概率出现在该文档中
* 给定主题 z，单词 w 以 P(w|z) 的概率从主题 z 中提取出来

IMG_256

也有从Z开始的。



* 因为我们没有参数来给 P(D) 建模，所以不知道如何为新文档分配概率
* pLSA 的参数数量随着我们拥有的文档数线性增长，因此容易出现过度拟合问题
* 
* 

阿尔法是生成0.7，0.2，0.1的迪利克雷先验，Selta被阿尔法抽出来是0.7，0.2，

0.1，Z通过selta抽出来是0.7对应的（如对应主题是狗）

phi是被beta抽出来的0.4，0.5，0.1（狗毛，狗尾巴，狗眼）

Z找主题，phi是对应主题下词语的分布。

LDA2vector

实际上就是skipgram连上lda的两种向量，即文档向量，以及主题矩阵

### **Blocked Gibbs sampler多于一个**

### **Collapsed Gibbs sampler**

自由度。显变量确定隐变量的自由度是经常遇到的问题，（变分假设通常认为互相独立。）

平均场理论：一堆电子对一个东西的作用互相独立而叠加得到最后的结果。

图模型中大概即为独立假设，每个隐变量对结果互相独立。

对称迪利克雷分布，即每一个阿尔法相等。（各向同性)

易犯的错误：图模型求解时不要用实际情况代理求解图模型的已知变量。

问题：

Lda两种推断：变分推断，gibbs采样推断

问题：

CRF公式由来还是说是硬假设。

回归树是什么

<https://www.cnblogs.com/pinard/p/6873703.html>

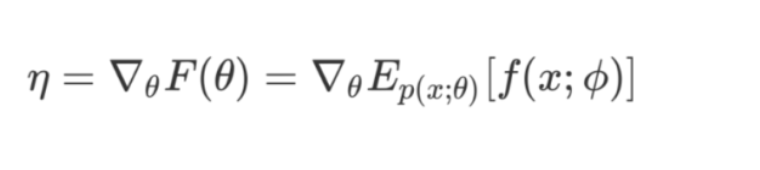
https://www.cnblogs.com/pinard/p/6867828.html

我希望有一个人能盯着icml，与iclr或者说机器学习纯度更高的会议。

由于显示对分布建模。。似然很容易计算。。。（）

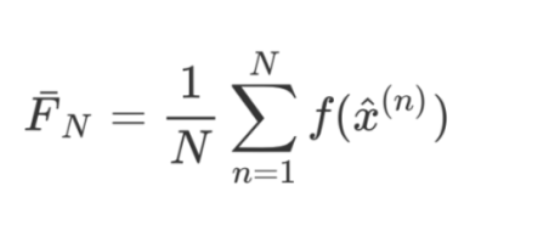
因此使用变分法引入mean field assumption，假设所有的隐变量都是通过各自独立的分布生成的，即去掉隐变量之间的连线和w结点，并赋予β、z、θ各自独立分布，λ、Φ、γ为变分参数。

**蒙特卡洛梯度估计方法（MCGE）**

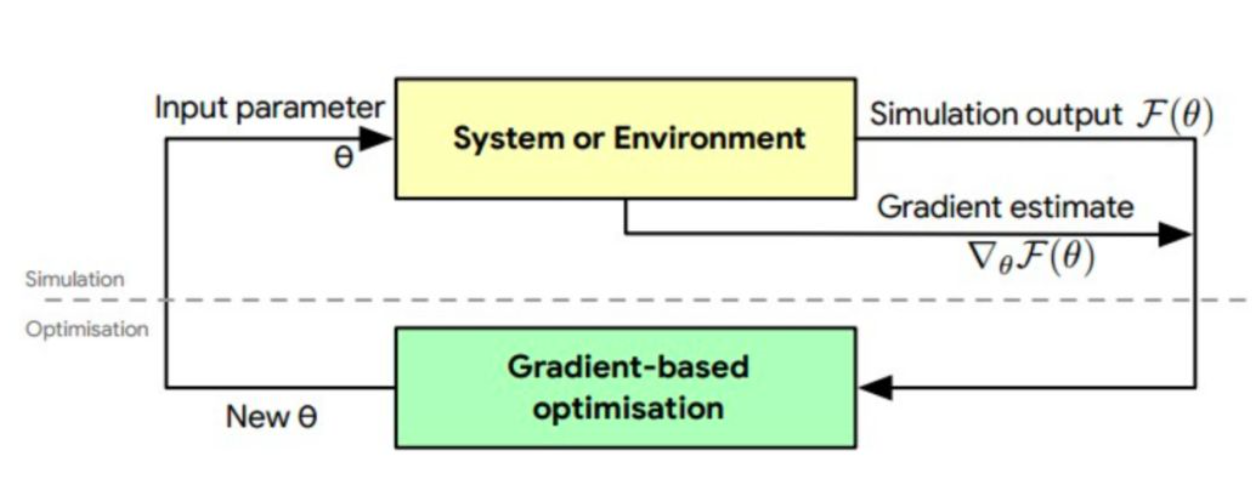


经常需要计算这个来更新分布。

背景知识是蒙特卡洛采样。

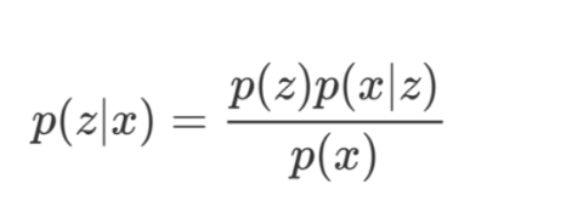


与随机优化

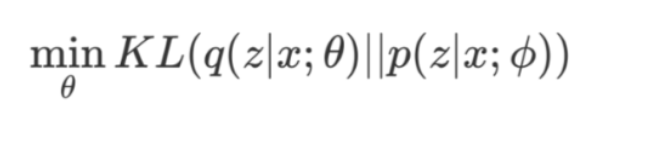


输入一个值输出确是随机的。

变分推断一般是最大化证据下界。



算不出来

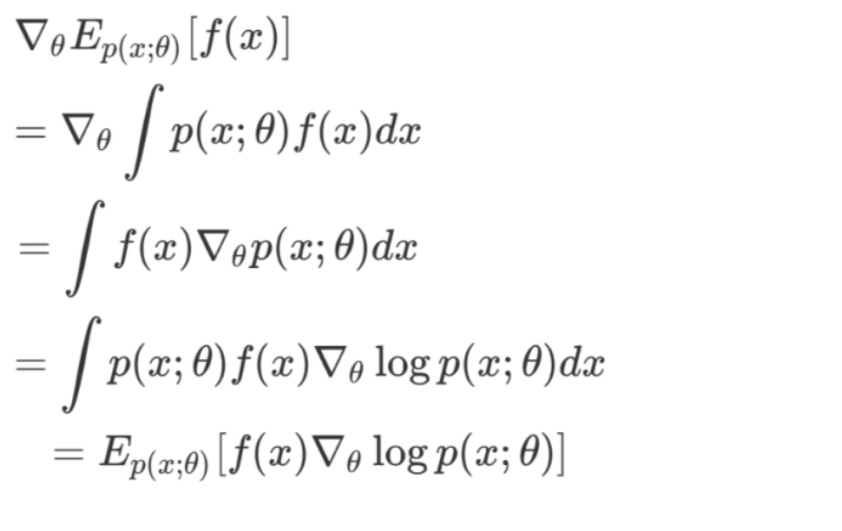


用q近似

要么p必须是已知的，（即使是像em中的迭代未收敛值）

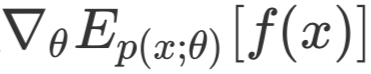
要么p是可以采样的。（黑箱实验。）

## Score Function Gradient Estimator (SFGE)



这类任务中经常会遇到一个问题，如何选择下一个更好的配置，使得选择之后比选择之前性能的概率会有所提升。因此需要优化如下问题：

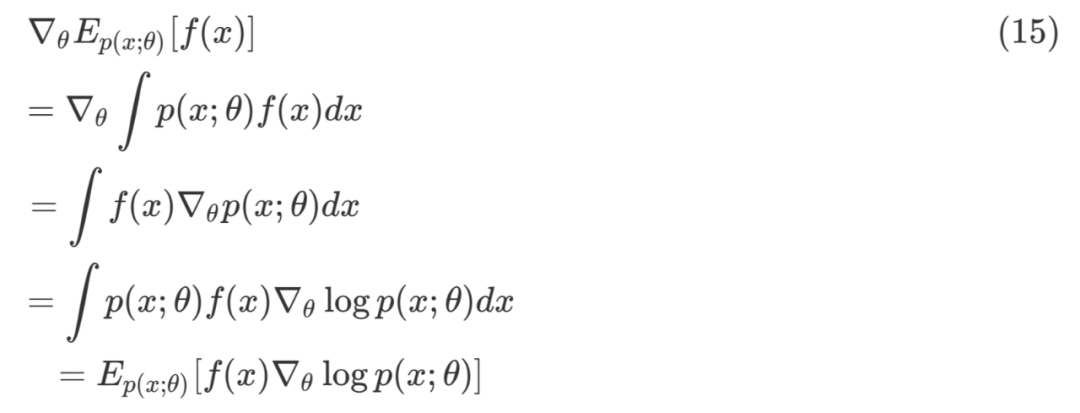




Fisher 信息量

score function是

统计学中：score是对[数似然函数](https://en.wikipedia.org/wiki/Log-likelihood_function" \o "Log-likelihood function)相对于[参数向量](https://en.wikipedia.org/wiki/Statistical_parameter" \o "Statistical parameter)的[梯度](https://en.wikipedia.org/wiki/Gradient" \o "梯度)



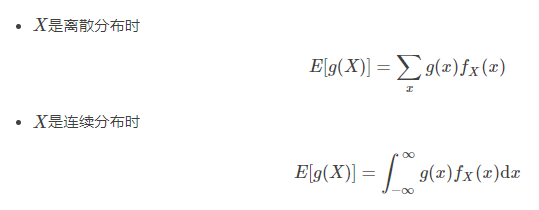
好求不代表好收敛。

## Pathwise Gradient Estimator (PGE)

不同于 SFGE 对代价函数没有任何约束，PGE 要求代价函数可微，虽然 SFGE 更具一般性，但 PGE 会有更好的性质。PGE在机器学习领域有一个重要的方法是

## Law of the Unconscious Statistician (LOTUS)

已知随机变量X的概率密度函数为This is the rendered form of the equation. You can not edit this directly. Right click will give you the option to save the image, and in most browsers you can drag the image onto your desktop or another program.，但不知道g(X)的概率分布，此时用LOTUS公式能计算出函数g(X)的数学期望。



联合分布时同理。

从频率的角度，叠加的大小可以是概率的大小，只需要对应放大的步长。

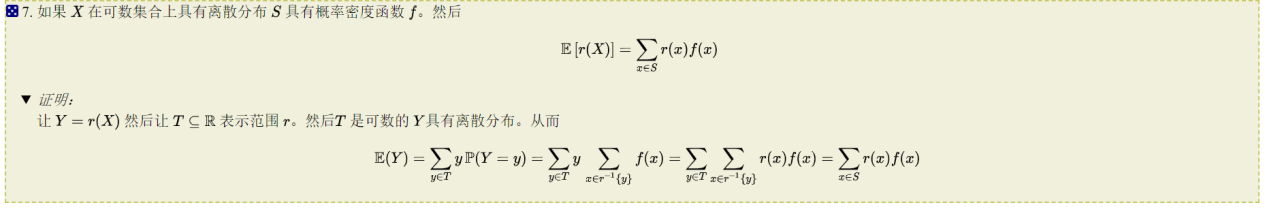
求解分布测度的导数，包括本文介绍的 score function gradient estimator

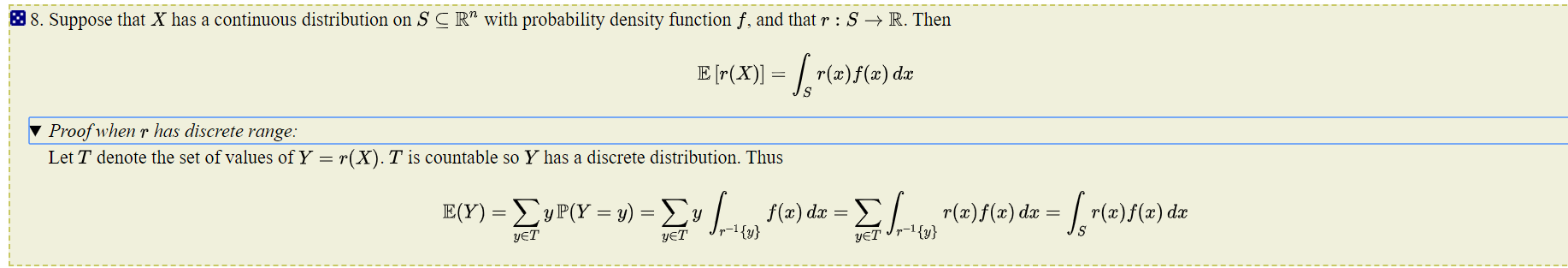
求解代价函数的导数，包括本文介绍的 pathwise gradient estimator

测度论换元。已知x=g（α）

Ex（f（x））=Eα（f（g（α）））

最夸张的莫过于被折成





反过来偷梁换柱也是可以的。

测度论换元的示范。结果还不大看得清楚。。

# Location–scale family Y=aX+b还是这个分布。

Vae发现了一个事情采样采一个，但很多x生成的z效果一样会很好。

LDA的

策略梯度算法与Q-learning

等于说我工作绕一圈还是到了如何看图模型推断带有监督信息的结果。。那种简单的想法最后会归结到算高斯分布上，而这个最后会被归为混合高斯模型聚类。。。

（理论上无限高斯模型可以近似任何分布，泛函意义上。。）

迪利克雷的后验即是偏观测向量。

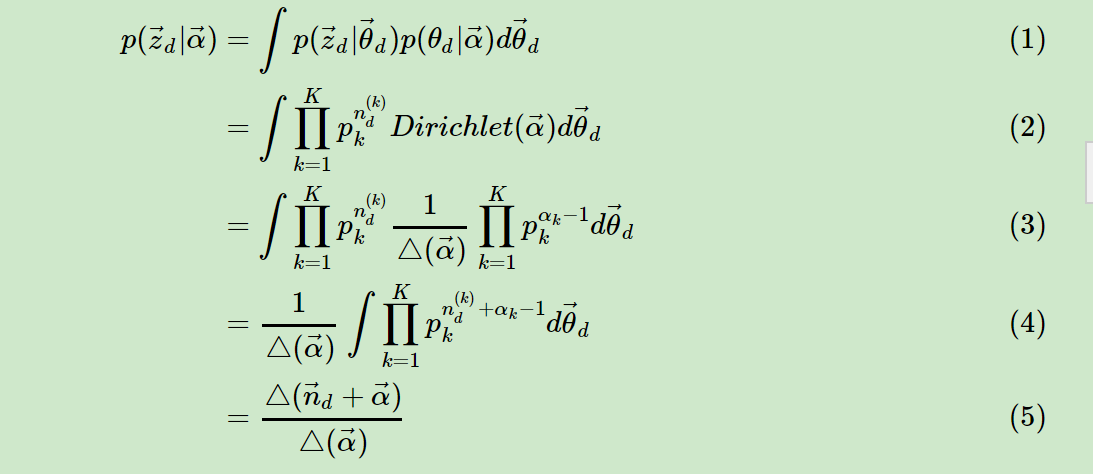
大概可以理解为先观测了阿尔法-1次。所以当阿尔法取1时是均匀分布，阿尔法必须大于0.。。在阿尔法小于1时会更偏0，1否则偏中间。

一般我们会倾向于把不变的变量作为正比分布组件。

##### 问题：

##### 假设中的αβ一般偏小（小于1）。

##### P（z|α）的分布是什么样子。



注意nd其实只有一维是1这其实已经是分布律了。。离散值请自行穷举。。但是这个是不是分母更大。。

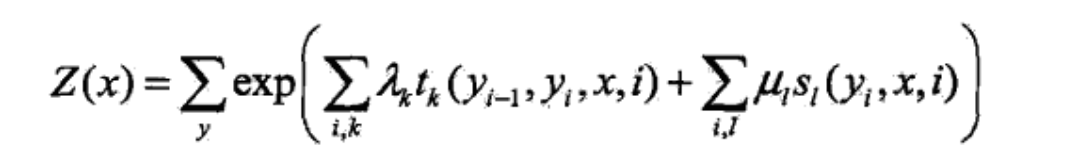
注意delta（α）是归一系数，gamma大的那头在下面。。。后面那个积分积出来总是小于1所以delta（α）也跟着小于1。且向量越大越小。。

条件随机场：CRF：特征函数，

每个（三元）团会有k个特征。

而（二元）团会有l个

当然有些规则只会对特定团有效。遍历一个团放入各个特征函数时可能会遇到根本无关或部分无关的团这时只会输出0



## What I cannot create, I do not understand.

只有自己能想象一个东西如何产生，才算是真正的理解。

Gibbs执行贝叶斯推断的特殊情况。

似然函数可以指的是未归一的概率函数。

共轭的选择就是看似然函数对应的参数形式。

细粒度的任务才容易涉及ool，oov。

耐心弄没等下一个人看懂了再交流吧。

时间序列分析

沃瑟斯坦距离

高斯过程，sne

凭什么能积分。。。是什么因素

回归树。

设备目前暂时考虑上云，aws，google drive，华为云，应该都可以，主要是每次要适配云的格式会把我心态炸。个人比较推荐的是10w的3titan rtx加t7920或t640加

因为人工智能的比赛基本是模型写好跑一下休息半天，有奖的话就是把比赛拿了十万一般还是有的。正好把0.5必修学分拿了。

杨子宁这周应该是组会分享。正常来说有时间的话应该弄个代码阅读。如何迁移学习。甚至如何放在keras里，结构，主要模块是怎么构成的。。

时昕我看下truth discovery的问题，以及解决方法，词向量能干什么。

朱晓祥的话有时间说下你看的论文。我们需要讨论，有必要的话可以吵起来。同理吴洁师姐。