New York University: Machine Learning in Economics

Lecture 13: Generative Models

Dr. George Lentzas

Introduction

Restricted Boltzman Machines

- The **Restricted Boltzmann Machine (RBM)** is an unsupervised learning model that tries to approximate the probability density function of the data.
- Like other "generative" models its estimation involves learning the probability distribution of the data in order to reconstruct it.
- The term "restricted" refers to the fact that there are no connections between units in the same layer.
- The probability of a specific arrangement of inputs and hidden units is proportional to the exponential of the negative of a so called"energy function".
- Connection between layers are symmetric and bi-directional, which allows information flow in both directions.

Restricted Boltzman Machines

The Probability Function

- Specifically, for each (binary) input/hidden vector pair the probability of the pair (v, h) is assumed to be

$$\Pr(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp(-E(\mathbf{v}, \mathbf{h})).$$

- Here the energy function E is given by

$$E(\mathbf{v}, \mathbf{h}) = -\sum_{i=1}^{m}\sum_{j=1}^{n} w_{ij} v_i h_i - \sum_{i=1}^{m} v_i b_i - \sum_{j=1}^{n} h_j d_j$$

  where w_{ij} is the weight between the input v_i and the hidden unit h_j, m and n are the number of visible and hidden nodes respectively and the parameters b_i and d_j are the biases.

- The normalizing constant Z is given by

$$Z = \sum_{\mathbf{v}}\sum_{\mathbf{h}} (\exp(-E(\mathbf{v}, \mathbf{h})))$$

Restricted Boltzman Machines

The Conditional Probabilities

- Since the hidden units of an RBM are connected only to units outside the specific layer ("restricted") they are -conditionally on the visible units- mutually independent.
- This allows us to factorize the conditional distributions of the hidden units as follows:

$$\Pr(\mathbf{h}|\mathbf{v}) = \prod_j \Pr(h_j|\mathbf{v})$$

with

$$\Pr(h_j = 1|\mathbf{v}) = \sigma\left(\sum_i w_{ij} v_i + d_j\right)$$

- The above generalizes to non binary hidden/output units, for a review of the different models see KMML.

Restricted Boltzman Machines

The Conditional Probabilities

- Similarly, for the conditional distributions of the visible units

$$\Pr(\mathbf{v}|\mathbf{h}) = \prod_i \Pr(v_i|\mathbf{h})$$

with

$$\Pr(v_i = 1|\mathbf{h}) = \sigma\left(\sum_j w_{ij} h_j + b_i\right)$$

- We use these conditional probabilities for calculating iterative updates between hidden and visible layers in estimation.

Restricted Boltzman Machines

Maximum Likelihood

- An RBM is typically estimated by log-likelihood. The gradient of the log-likelihood is given by

$$\frac{\partial}{\partial \boldsymbol{\theta}} L(\boldsymbol{\theta}) = -\langle \frac{\partial E(v;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \rangle_{data} + \langle \frac{\partial E(v;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \rangle_{model}$$

  where:
  $\langle . \rangle_{data}$ is the expectation under the conditional distribution of the hidden given the visible units and
  $\langle . \rangle_{model}$ is the expectation under the model distribution $\Pr(v, h)$.
- Intuitively, this says that when the model expectations are equal or close to the empirical expectations, the model has converged (the gradient is close to zero).
- Here is a good article with the proof if you want to read about this http://image.diku.dk/igel/paper/AItRBM-proof.pdf.
- These expectations tend to be intractable to calculate so we typically use a technique called Gibbs sampling to estimate them. This is a Bayesian technique that produces a simulated sample which approximates the joint probability distribution of interest.

Restricted Boltzman Machines

A (Very) Brief Intro to MCMC

- MCMC methods allow us to draw samples from very complex/intractable distribution, which are then used to obtain estimates of functions of the distribution, such as an expectation. Underlying this concept is the idea that if we want to estimate $\mathrm{E}(X)$ and we are able to draw (simulate) a sample from $f(X)$ then we could use the sample mean as an estimate of the expected value.
- Gibbs sampling, which is commonly used MCMC method) takes this one step further by allowing us to simulate sample from complex/intractable multivariate distributions by repeatedly simulating the conditional distributions. For example, lets assume we want to simulate a sample from $\Pr(\mathbf{v}, \mathbf{h})$. Then we can simulate samples from $\Pr(\mathbf{v_{i+1}}|\mathbf{h_i})$ and then from $\Pr(\mathbf{h_{i+2}}|\mathbf{v_{i+1}})$ and if we iterate long enough the resulting sample (after potentially throwing out the first observations) will have converged to the joint distribution of interest.

Restricted Boltzman Machines

A (Very) Brief Intro to MCMC

- In the context of RBMs Gibbs sampling is a method where the visible nodes are sampled all together given values for the hidden nodes and similarly hidden nodes are sampled all together given the visible nodes. As the number of iterations becomes increasingly large the derived sample will converge to its joint distribution $\Pr(v, h)$.
- Fortunately, it has been shown that just running a few iterations produces estimates that work well for the purposes of training the RBM. This process is called "Contrastive Divergence" and its details are beyond the scope of this class. The paper above is good read on the details for the motivated reader.
- The standard RBM uses binary units for both hidden and visible layers. Real valued data in the $[0, 1]$ range can easily be used with essentially no changes to the above approach.

Restricted Boltzman Machines

The ML Gradient

- Let us consider in more detail how Maximum Likelihood (ML) estimation works for RBMs. We are interested in the total probability of the visible variables $\mathbf{V}$ given by:

$$\Pr(\mathbf{v}) = \sum_{\mathbf{h}} \Pr(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \sum_{\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}))$$

and where

$$Z = \sum_{\mathbf{v},\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h})).$$

- The log-likelihood of the sample is given by

$$\log L(\boldsymbol{\theta}|\boldsymbol{v}) = \log \Pr(\mathbf{v}|\theta) = \log \left[\frac{1}{Z} \sum_{\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h})) \right]$$

$$= \log \sum_{\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h})) - \log \sum_{\mathbf{h},\mathbf{v}} \exp(-E(\mathbf{v}, \mathbf{h}))$$

Restricted Boltzman Machines

The ML Gradient

- This gives the following gradient

$$\frac{\partial}{\partial \boldsymbol{\theta}} L\left(\boldsymbol{\theta}|\mathbf{v}\right) = \frac{\partial}{\partial \boldsymbol{\theta}}\left[\log \sum_{\mathbf{h}} \exp\left(-E\left(\mathbf{v},\mathbf{h}\right)\right)\right] - \frac{\partial}{\partial \boldsymbol{\theta}}\left[\log \sum_{\mathbf{h},\mathbf{v}} \exp\left(-E\left(\mathbf{v},\mathbf{h}\right)\right)\right]$$

$$= \frac{1}{\sum_{\mathbf{h}} \exp\left(-E\left(\mathbf{v},\mathbf{h}\right)\right)} \sum_{\mathbf{h}} \exp\left(-E\left(\mathbf{v},\mathbf{h}\right)\right) \frac{\partial}{\partial \boldsymbol{\theta}} E\left(\mathbf{v},\mathbf{h}\right)$$

$$+ \frac{1}{\sum_{\mathbf{h},\mathbf{v}} \exp\left(-E\left(\mathbf{v},\mathbf{h}\right)\right)} \sum_{\mathbf{h},\mathbf{v}} \exp\left(-E\left(\mathbf{v},\mathbf{h}\right)\right) \frac{\partial}{\partial \boldsymbol{\theta}} E\left(\mathbf{v},\mathbf{h}\right)$$

Restricted Boltzman Machines

The ML Gradient

- Using Bayes Rule for conditional probabilities

$$\Pr(\mathbf{h}|\mathbf{v}) = \frac{\Pr(\mathbf{v},\mathbf{h})}{\Pr(\mathbf{v})} = \frac{\frac{1}{Z}\exp(-E(\mathbf{v},\mathbf{h}))}{\frac{1}{Z}\sum_{\mathbf{h}}\exp(-E(\mathbf{v},\mathbf{h}))} = \frac{\exp(-E(\mathbf{v},\mathbf{h}))}{\sum_{\mathbf{h}}\exp(-E(\mathbf{v},\mathbf{h}))}$$

and the definition

$$\Pr(\mathbf{v},\mathbf{h}) = \frac{1}{\sum_{\mathbf{v},\mathbf{h}}\exp(-E(\mathbf{v},\mathbf{h}))}\exp(-E(\mathbf{v},\mathbf{h}))$$

- By substitution it follows that

$$\frac{\partial}{\partial\boldsymbol{\theta}}L(\boldsymbol{\theta}|\mathbf{v}) = \sum_{\mathbf{h}}\Pr(\mathbf{h}|\mathbf{v})\frac{\partial}{\partial\boldsymbol{\theta}}E(\mathbf{v},\mathbf{h}) + \sum_{\mathbf{h},\mathbf{v}}\Pr(\mathbf{v},\mathbf{h})\frac{\partial}{\partial\boldsymbol{\theta}}E(\mathbf{v},\mathbf{h})$$

which gives us the difference of expectation equation for the gradient. Directly calculating these sums (which iterate over all the hidden and observed variables) is of exponential complexity and needs to be approximated, typically by MCMC.

Restricted Boltzman Machines

The ML Gradient

- But first some more detail; let us consider the gradient w.r.t. the weight w_{ij} given by

$$\frac{\partial}{\partial w_{ij}} L(\boldsymbol{\theta}|\mathbf{v}) = \sum_{\mathbf{h}} \Pr(\mathbf{h}|\mathbf{v}) \frac{\partial}{\partial w_{ij}} E(\mathbf{v},\mathbf{h}) + \sum_{\mathbf{h},\mathbf{v}} \Pr(\mathbf{v},\mathbf{h}) \frac{\partial}{\partial w_{ij}} E(\mathbf{v},\mathbf{h})$$

$$= \sum_{\mathbf{h}} \Pr(\mathbf{h}|\mathbf{v}) h_i v_j + \sum_{\mathbf{h},\mathbf{v}} \Pr(\mathbf{v},\mathbf{h}) h_i v_j$$

- The first term we can calculate using the factorization trick we saw above. The second term however is intractable and is what we need to evaluate via MCMC. Specifically we can approximate this by simulation samples from the model distribution using Gibbs sampling. This requires simulating long enough for the Markov Chain to reach the stationary distribution! In practice this is too slow so we have to introduce an additional approximation; this is where CD comes in.

Restricted Boltzman Machines

Contrastive Divergence

- The idea behind k-step CD is to approximate the second term in the gradient by a sample from the model distribution derived from a short (k=step) Markkov Chain. This works as follows:
  1. Initialize the MCMC with a training example $\mathbf{v^0}$ (from the mini-batch)
  2. Sample $\mathbf{h^1}$ from $\Pr\left(\mathbf{h}|\mathbf{v^0}\right)$ and then sample $\mathbf{v^1}$ from $\Pr\left(\mathbf{v}|\mathbf{h^1}\right)$. This is one (of the k) pass. You can now see why the conditional independence is such a crucial property!
  3. Approximate the gradient using

$$CD_1\left(\boldsymbol{\theta}, \mathbf{v^0}\right) = \sum_{\mathbf{h}} \Pr\left(\mathbf{h}|\mathbf{v^0}\right) h_i v_j^0 + \sum_{\mathbf{h},\mathbf{v}} \Pr\left(\mathbf{v^k}, \mathbf{h^k}\right) h_i^k v_j^k$$

- Since typically a small (k=1) number of steps is used the approximation is biased, however in practice it works sufficiently well. Extensions of CD (e.g. Persistent CD) deal with issues of bias and convergence and are typically preferred in practice.

Deep Belief Nets

DBNs

- A DBN is a multilayer neural network comprised of several stacked RBMs! As with deep auto-encoders the output of one RBM becomes the input of the next RBM.
- Stacked RBMs can thus be interpreted as deterministic feed-forward Neural Networks, with each successive payer extracting increasingly relevant features from the data and the resulting RBM initialization (tantamount to unsupervised pre-training) helps the top layer Neural Network to perform better.
- As with a deep auto-encoder models training is achieved in two steps, Pre-Training and Fine-Tuning.
- Pre-training: use layer-by-layer RBM training in which each layer is trained individually by contrastive divergence. The *activation probabilities* of the hidden units in the estimated RBM are used as inputs for the next RBM.
- Fine-tuning: use back-propagation in the entire network to make small adjustments to perform classification tasks.

Reading and Homework

Reading

- These notes are based on "Deep Learning Made Easy with R", by N.D Lewis, Chapters 7, 8 and on "An Inrtoduction to Restricted Boltzmann Machines" by A. Fischer and C. Igel.
- "Deep Learning Methods and Applications". Deng and Yu, Foundations and Trends in Signal Processing.
- "Machine Learning, A Probabilistic Perspective", K. Murphy, Chapter 28.
- "A Practical Guide to Training Restricted Boltzmann Machines", G. Hinton.
- "Deep Learning", Y. LeCun, Y Bengio and G. Hinton, Nature.
- "Deep Learning Made Easy with R", by N.D Lewis
- "Deep Learning Methods and Applications". Deng and Yu, Foundations and Trends in Signal Processing.
- "Machine Learning, A Probabilistic Perspective", K. Murphy, Chapter 28.
- "Deep Learning", Y. LeCun, Y Bengio and G. Hinton, Nature.