
Learning Exponential Families

Anonymous Author(s)

Affiliation

Address

email

Abstract

[SLOPPY NOTES STAGE JUST TO GET THOUGHTS DOWN]

Recently much attention has been paid to probabilistic models defined by a deep neural network transformation of a simpler random variable; these implicit generative models have been used to great success across variational inference, generative modeling of complex data types, and more. In essentially all of these settings, the model is specified by the network architecture, and a particular member of that model is chosen to minimize some loss (be it adversarial or information divergence)

We treat the problem of learning an exponential family – the model itself, rather than the typical setting of learning a particular member of that model.

Many intractable distributions encountered in machine learning belong to exponential families. In rare cases these distributions are tractable due to either known conjugacy in the problem setup (such as the normal-inverse-Wishart), or due to careful numerical work historically that has made these distributions computationally indistinguishable from tractable (eg the Dirichlet).

1 Introduction

People use lots of implicit generative models:

Across machine learning, including ABC [?], GANs [1], VAEs [2, 3], and their many follow-ons (too numerous to cite in any detail), models that specify a distribution via the nonlinear transformation of latent random variable. We prefer and use the terminology of [4], calling such a distribution an *implicit generative model*, defined as:

something like eq 1 and 2 in Mohamed:2016aa, defining $q_{\theta}(z)$

Also use the proper notation of the density implied by the pushforward measure of the function $f_{\theta\sharp}$ if useful. Also reference to this being super standard and widespread [5]. The two central uses are at present generative distributions of interesting data types (as in GANs), and for variational inference. Regardless, all of these use cases specify a *model* (or variational family) $\mathcal{Q} = \{q_{\theta} : \theta \in \Theta\}$, and then minimize a suitable loss $\mathcal{L}(q, p)$ over $q \in \mathcal{Q}$. In the case of VI p is the posterior (or the unnormalized log joint) and \mathcal{L} is the KL divergence (or so called ELBO), in GAN p is the sample density of a (large) dataset and \mathcal{L} is the adversarial objective whose details do not matter here.

All these learn a single member of a family

Inherent in all the above approaches is an algorithmic procedure to select a *single* distribution $q_{\theta}(z)$ from among the *model* \mathcal{Q} . Implicit in this effort is the belief that \mathcal{Q} is suitably general to contain the true distribution of interest, or at least an adequately close approximation.

Here we learn the family

We leverage the natural parameterization of exponential families to derive a novel objective that is amenable to stochastic optimization.

31 *A note on amortization*

32 Several have pointed out that these IGMs are in fact strictly less expressive than a mean field, at
33 least in the conventional VI setting. See for example [http://dustintran.com/blog/variational-auto-](http://dustintran.com/blog/variational-auto-encoders-do-not-train-complex-generative-models)
34 [encoders-do-not-train-complex-generative-models](http://dustintran.com/blog/variational-auto-encoders-do-not-train-complex-generative-models) (here I like the line “The neural network used in
35 the encoder (variational distribution) does not lead to any richer approximating distribution. It is a
36 way to amortize inference such that the number of parameters does not grow with the size of the data
37 (an incredible feat, but not one for expressivity!) (Stuhlmüller et al., 2013)”). You have to optimize
38 for every data point individually, or instead you get to do so in aggregate once in advance (at a much
39 higher cost) and then recover that cost over future data points within that distribution (and hence the
40 term amortization, though perhaps there is shared statistical power as well) Etc etc what we are doing
41 here is *amortized* amortized inference, in the sense that we are amortizing not the data points, but the
42 distribution itself.

43 REparameterization trick (Kingma and Welling (2013), Rezende et al. (2014) and Titsias and
44 Lazaro-Gredilla 2014).. See also Archer 2015 / Gao 2016 for clean explanation.

45 Key for obvious norm flow connection but also a good bibliography and some good historical views
46 to Dayan and Gershman and other people who did norm flows. <https://arxiv.org/pdf/1505.05770.pdf>

47 *Our contributions include:*

48 ...

49 This should not be confused with "Learning to learn by gradient descent by gradient descent"
50 (Andrychowicz et. al. 2016) and similar works.

51 ...

52 Important to distinguish carefully from VI. In a sense VI does parameterize a family: given data,
53 you get local variational parameters and that parameterizes a density (like a regular VAE). Inference
54 networks are exclusively used to data to amortize with a global set of parameters a variational
55 distribution, not a model. Of course it is in a sense a model, but that's a bunch of normals. The
56 sampling mechanism is easy (Gaussian).

57 *Our results demonstrate*

58 ...

59 **2 Learning exponential families**

60 *Why this is important*

61 Exp fams are awesome and fundamental []. Also [?] rightly point out that many many inference
62 problems can be cast as exponential families. Can we cast the VAE encoder network as a suitable
63 exp fam... sure I think that's right; the network parameters of z form the statistics, and then the
64 observations are η 's.

65 *Why this is coherent*

66 Θ defines quite a big \mathcal{Q} , and indeed the subject of compressibility, generalization, etc is of keen
67 interest to many [?]. So actually the space of distributions is quite large, and in many cases certainly
68 larger than it needs be. Why? Well, we know precisely the parameter space of the exponential family;
69 it is defined by the *natural* parameters $\eta \in \mathbb{R}^p$ (or whatever we choose there).

70 *Figure 1*

71 Figure of model space. Yeah that's good. Then graphical model. Note that perhaps \mathcal{Q} is too big,
72 and a simpler model space (the $\|\eta\|$ dimensional subspace of Θ) would be better for the usual
73 robustness/generalization reasons.

74 *Aside*

75 A neat idea is to ask if learning the $\theta(\eta)$ network leads to better VI in terms of inference networks,
76 since it is apparently appropriately regularized and can just take suff stats. That's testable if we have
77 time.

78 *Why Flow Networks*

We choose flow networks [] and [] because duh. And "implicit generative models aka density networks" (or rather, density networks are the instantiation of an IGM with deep nets, which is effectively synonymous these days. Gibbs and MacKay Density Networks 1997! And invertible networks In that vein probably definitely cite invertible deep nets in general: Baird et al IJCAI 2005, Ripple and Adams 2013 . Note that what norm flows (the Rezende/Mohamed stuff specifically) did is make it tractable and scalable and in the modern VAE style. That makes these comparisons legitimate and apples to apples. Any generalization of this is also dandy though, so could use a mean field approach (standard) or any of the things that go beyond mean field, either classically (Saul and Jordan, 1996; Barber and Wiering, 1999); this is called structured variational inference. Another way to expand the family is to consider mixtures of variational densities, i.e., additional latent variables within the variational family (Bishop et al., 1998). or newer stuff [] [Tran Copula VI, Hoffman and Blei 2015].

As noted in norm flows paper: "The true posterior distribution will be more complex than this assumption allows for, and defining multi-modal and constrained posterior approximations in a scalable manner remains a significant open problem in variational inference."

Couch this in terms of normalizing flows though point out this is not strictly necessary. Note in particular Tabak, E. G. and Turner, C. V. A family of nonparametric density estimation algorithms. Communications on Pure and Applied Mathematics, 66(2):145-164, 2013. Tabak, E. G and VandenEijnden, E. Density estimation by dual ascent of the log-likelihood. Communications in Mathematical Sciences, 8(1):217-233, 2010. A nice line from Rezende and Mohamed is: Thus, an ideal family of variational distributions $q(z|x)$ is one that is highly flexible, preferably flexible enough to contain the true posterior as one solution. One path towards this ideal is based on the principle of normalizing flows (Tabak Turner, 2013; Tabak VandenEijnden, 2010).

Related work / How close is this to norm flows or VAE

In a restricted technical sense, rather close: VAE and other black box VI that uses reparameterization results in a conditional density $q_\phi(z|x)$. If we consider η as x , then sure yes the previous stuff specifies a model $\mathcal{Q}_{VAE} = \{q_\phi(z|x) : x \in X\}$. But that's a little silly, and any way that is very often a normal family with variational parameters specified by (a deep function of) x . Much closer is Figure 2 in Rezende and Mohamed, where like here they use a network to index the parameters of the normalizing flow. In that case it's a function of x the observation, and as such that network is an inference network; here it's a function of η and as such is a parameter network. That's just nomenclature, so naturally the next question is do they differ at some other level. Yes, distinctly. The other term implied in a VI (or norm flow VAE style as they use) is the expected log joint $E_{q_\phi(x)}(\log p_\theta(x, z))$. Now sure that's a loss function on x, z , so then when we look at that same term in EFN we see $E_{q_\phi(\eta)}(\eta^\top t(z))$, which sure also looks like a loss function on η, z . And yes, they are both unnormalized (in the sense that VI is an ELBO / joint $p(x, z)$ and EFN lacks the normalizer because it's constant, so we're not getting a KL estimate). A picky difference is that the exp family doesn't really correspond to a proper unnormalized log joint (though I suppose it could), as there is not a prior on η in the objective (but is that just ignoring $p(\eta)$ in our sampling scheme?). But yes if we want to be reductionist and pedantic [use nicer words] in general we could see this as a specific case where $x = \eta$ and thus we are learning a family just as in the inference case. And for example we are giving in the bayesian logistic regression example full datasets for inference instead of single data points. To make this as close as possible, we write $p(\eta|z) = \frac{1}{A(t(z))} \exp\{\eta^\top t(z)\}$. That's the "likelihood" of an EFN in some wonky sense. So this reveals the mechanical differences: first, $t(z)$ is not a deep generative model with parameters θ , but rather it is a fixed set of sufficient statistics that define the exp fam. Next, there is no clear prior $p(z)$, which is critical to understanding how VI behaves (see Hoffman and Johnson ELBO surgery paper). So yes there is a hand wavy sense in which EFN is a specific case of norm flow, but of course it is. And anyway norm flow is a specific case of a DNN architecture or Helmholtz machine or deep density network (Ripple and Adams). This is just rambling but good to have all perspective here. Ok so what to do? First, then we need to produce really compelling results focusing on when learning an exp fam is key. Second we need some very tight language to draw this distinction without seeming a small tweak on normalizing flows. One way to do this is the restricted model class argument, a la Fig 7.2 in Hastie and Tibshirani. Another is to actually produce a conditional exp fam, as in something indexed on both x and η . Third, possible novelties in norm flows, like triple spinners or other better choices than planar flows (yuck).

134 Another related work is that this is somehow the dual of MEFN, or a generalization of the dual
135 problem. In the wainwright and jordan sense of forward and backward mappings.

136 3 Results

137 *Chapter 1, Fig 1*

138 Toy figure that demonstrates what we are doing and a simple example. Note this should probably not
139 be in Results but in the EFN section or similar. Ideas:

- 140 • value of a restricted model, see hastie tibshirani fig 7.2, or porbanz's batman version from
141 4400 slides. ... well that's a bit off topic. At least worth a mention in motivation.
- 142 • graphical model. yeah probably needed.
- 143 • network model. yeah probably needed.
- 144 • cartoon example three sets of natural parameters in, three dirichlet distributions out. Or
145 similar.

146 *Chapter 2: Fig 2 and 3 and 4* Ground truth toy examples, etc.

147 Fig 2. This says, it works in a sensible case (maybe Dirichlet 25 or NIW 10x10):

- 148 • training and test curves (probably both of these are needless since by def train equals test
149 here.
- 150 • both in r^2 and in KL to ground truth.
- 151 • MMD p values? Can this help give specifics?

152 Fig 3. This says, it works across a range of Dirichlet and NIW (and perhaps not Gaussian). Sensitivi-
153 ties

- 154 • dimensionality of dir and NIW
- 155 • choices of K, M (is this really needed)
- 156 • key comparison to baseline of VI (doing this specifically for $K=1$ but a fixed K , possibly
157 with no parameter network because it's unnecessary). The key is to show that it works
158 almost as well as VI in the specific case, but of course VI can't generalize. And when I say
159 VI I don't really mean VI, I mean just learning a specific member of the exp fam.

160 Fig 4. One great virtue of EFNs is is learning a restricted model, which should demonstrate the usual
161 bias-variance tradeoff. We can show this. Suppose we have only a limited amount of data from a
162 specific exp fam, but we don't know what member. Yeah this is cool. So we can either explore all
163 of ϕ space in the usual way (the full flow network model \mathcal{Q}), or we can have previously learned
164 the exp fam, which operates on the restricted coordinates of the natural parameters. Then, we take
165 gradients over natural parameter space to find the best ϕ , which is really $\phi^* = \phi(\eta^*)$, and hopefully
166 that distribution is closer in KL to the true posterior than had we just learned straight from \mathcal{Q} . Well
167 this isn't quite fair, as we don't even need to take gradients in η space, since in the simplest case we
168 just plug in η and we've got the exact member. But yes that's a good example too. Possibly better. As
169 a function of available data, can you fit the posterior? Well, you can learn directly from \mathcal{Q} , always an
170 option. That shows something probably unhelpful: that you can learn a point mass or something like
171 that. Hmm not quite. Ok use a Norm Flow with a bayesian logistic regression generative model
172 and limited data. Learn the posterior on β for a given dataset. That maps a dataset X to a posterior.
173 That will run into scaling problems as X grows, because apparently the network has to follow suit.
174 Instead cheat and give it some sufficient statistics. And ok that's not so much of a cheat because you
175 would know from the model. So that's fair I suppose. Another choice is to just model the data, so
176 write the generative model and ask the norm flow to learn the posterior. That's the same thing and
177 I'm talking in circles. There is a good idea here.

178 *Chapter 3: Fig 5 and 6*

179 Fig 5. Bayesian logistic regression. Learn the full BLR posterior (BLRP) family for some dimension-
180 ality. Then get some data (see links in slack, look for nickish and all that). Then find the posterior.

181 etc, etc, etc. Then use MF VI or EP or whatever in an attempt to do same. Show outperformance.
 182 Really scrutinize the comparison point here. What really is VI in this setting?

183 Fig 6. The Killer real data. Perhaps Gibbs or Markov Random Field. Learn it, then pick some η , then
 184 show samples from it. Can this look interesting? Some thoughts...

185 Criteria:

- 186 • Needs to be an exp fam.
- 187 • Needs to be a forward exp fam. As in, not fit to data, because we don't have μ parameters,
 188 we have η parameters.
- 189 • "real data" is a misnomer, since we are not doing VI or similar. Really we want an exp fam
 190 that is real and somehow useful in its own right, and that people want to sample from.
- 191 • Reminder: we will *always* be comparing to "well normally you can do this with learning
 192 a *single* distribution in the $\min KL(q||p)$ sense. That's fine. The point is we can learn the
 193 whole family, then choose and sample, vs just one by one.
- 194 • something hard to sample will be key, since the "toy" results will have used things we
 195 already "know" how to sample, like NIW or Dirichlet.

196 Ideas:

- 197 • Fancy Exp Fam like Marshall-Olkin. Yeah but who really cares about this esoteric distribu-
 198 tion? It doesn't look cool visually either.
- 199 • Ising models: classic, bw images
- 200 • Potts model: great because failure of MCMC (Gibbs sampling) here is at least locally
 201 well known from Geman and Geman 1984 through Sudderth correcting this (see Gibbs
 202 sampler slides from Advanced ML, Peter's part). But that is kind of a failure example, not
 203 an interesting one (MRFs are smoothness prior, not segmentation prior).
- 204 • Markov Random Fields / Gibbs Random Fields (same, by Hammersley Clifford theorem).
 205 Yes this is cool: image distributions, texture distributions. Can show wild diff sets of textures,
 206 none of which require any sampling or any such thing. Can we make this super intractable
 207 from an MCMC perspective? Need to read on how sampling is done there. Erik Sudderth
 208 and his phd thesis are likely good resources.
- 209 • Gatys and Simoncelli texture stuff (see for example MEFN paper for refs); those are
 210 interesting distributions on textures, or specified moments. Can then just sample from this
 211 family.

212 4 To Do

213 4.1 SRB

- 214 • set up submission at <https://cmt.research.microsoft.com/NIPS2018/>
- 215 • review and conform to style requirements (see website with template); 8 pages not including
 216 refs and acks and appendices.

217 4.2 JPC

- 218 • Outline
- 219 • Write

220 \usepackage[pdftex]{graphicx} ...
 221 \includegraphics[width=0.8\linewidth]{myfile.pdf}

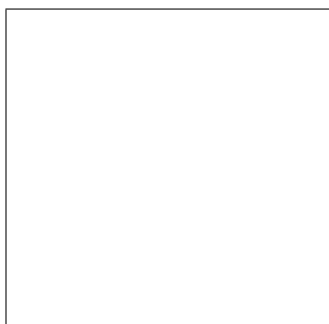


Figure 1: Figure 1: possibly Fig 7.2 bias-variance tradeoff and then benefit of a restricted model from Hastie Book, or similar from W4400 (ask PO for batman permission).

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	~ 100
Axon	Output terminal	~ 10
Soma	Cell body	up to 10^6

Acknowledgments

Use unnumbered third level headings for the acknowledgments. All acknowledgments go at the end of the paper. Do not include acknowledgments in the anonymized submission, only in the final paper.

References

References

- [1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.
- [2] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. 12 2013.
- [3] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.
- [4] Shakir Mohamed and Balaji Lakshminarayanan. Learning in implicit generative models. 10 2016.
- [5] Luc Devroye. *Non-uniform random variate generation*. Springer-Verlag, New York, 1986.
- Stuff on wake sleep and the Helmholtz machine
- Stuff on sampling from Gibbs distributions (max ent models), and sampling from exp fams generally, with MCMC and such.
- Flow networks
- Devroye’s book.
- Hoffman et al 2013 SVI
- From Blei review on VI. The development of variational techniques for Bayesian inference followed two parallel, yet separate, tracks. Peterson and Anderson (1987) is arguably the first variational procedure for a particular model: a neural network. This paper, along with insights from statistical mechanics (Parisi, 1988), led to a flurry of variational inference procedures for a wide class of models (Saul et al., 1996; Jaakkola and Jordan, 1996, 1997; Ghahramani and Jordan, 1997; Jordan et al., 1999). In parallel, Hinton and Van Camp (1993) proposed a variational algorithm for a similar neural network model. Neal and Hinton (1999) (first published in 1993) made important connections to the expectation maximization (EM) algorithm (Dempster et al., 1977), which then led to a variety of variational inference algorithms for other types of models (Waterhouse et al., 1996; MacKay, 1997).
- Salimans, T. and Knowles, D. (2014). On using control variates with stochastic approximation for variational Bayes. *arXiv preprint arXiv:1401.1022*.
- Salimans, T., Kingma, D., and Welling, M. (2015). Markov chain Monte Carlo and variational inference: Bridging the gap. In *International Conference on Machine Learning*, pages 1218–1226.
- Ranganath, R., Gerrish, S., and Blei, D. (2014). Black box variational inference. In *Artificial Intelligence and Statistics*.
- Hoffman, M. D. and Blei, D. M. (2015). Structured stochastic variational inference. In *Artificial Intelligence and Statistics*.
- Possibly some of Burda, Y., Grosse, R., & Salakhutdinov, R. (2016). Importance Weighted Autoencoders. In *International Conference on Learning Representations*. Damianou, A. C., & Lawrence, N. D. (2013). Deep Gaussian Processes. In *Artificial Intelligence and Statistics*. Dayan, P., Hinton, G. E., Neal, R. M., & Zemel, R. S. (1995). The Helmholtz Machine. *Neural Computation*, 7(5), 889–904. <http://doi.org/10.1162/neco.1995.7.5.889> Dinh, L., Sohl-Dickstein, J., & Bengio, S. (2016). Density estimation using Real NVP. *arXiv.org*. Harville, D. A (1977). Maximum likelihood approaches to variance component estimation and to related problems. *Journal of the American Statistical Association*, 72(358):320–338. Hinton, G. and Van Camp, D (1993). Keeping the neural networks simple by minimizing the description length of the weights. In *Computational Learning Theory*,

269 pp. 5713. ACM. Johnson, M. J., Duvenaud, D., Wiltchko, A. B., Datta, S. R., & Adams, R. P.
 270 (2016). Composing graphical models with neural networks for structured representations and fast
 271 inference. arXiv.org. Kingma, D. P., & Welling, M. (2014). Auto-Encoding Variational Bayes. In
 272 International Conference on Learning Representations. Kingma, D. P., Salimans, T., & Welling, M.
 273 (2016). Improving Variational Inference with Inverse Autoregressive Flow. arXiv.org. Louizos, C.,
 274 & Welling, M. (2016). Structured and Efficient Variational Deep Learning with Matrix Gaussian
 275 Posteriors. In International Conference on Machine Learning. Maaloe, L., Sonderby, C. K., Sonderby,
 276 S. K., & Winther, O. (2016). Auxiliary Deep Generative Models. In International Conference
 277 on Machine Learning. MacKay, D. J., & Gibbs, M. N. (1999). Density networks. Statistics and
 278 neural networks: advances at the interface. Oxford University Press, Oxford, 129-144. Mnih, A., &
 279 Rezende, D. J. (2016). Variational inference for Monte Carlo objectives. In International Conference
 280 on Machine Learning. Ranganath, R., Tran, D., & Blei, D. M. (2016). Hierarchical Variational
 281 Models. In International Conference on Machine Learning. Rezende, D. J., Mohamed, S., & Wierstra,
 282 D. (2014). Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In
 283 International Conference on Machine Learning. Salimans, T., Kingma, D. P., & Welling, M. (2015).
 284 Markov Chain Monte Carlo and Variational Inference: Bridging the Gap. In International Conference
 285 on Machine Learning. Salakhutdinov, R., Tenenbaum, J. B., and Torralba, A (2013). Learning with
 286 hierarchical-deep models. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 35
 287 (8):1958-1971. Stuhlmüller, A., Taylor, J., & Goodman, N. (2013). Learning Stochastic Inverses.
 288 In Neural Information Processing Systems. Tran, D., Blei, D. M., & Airoldi, E. M. (2015). Copula
 289 variational inference. In Neural Information Processing Systems. Tran, D., Ranganath, R., & Blei, D.
 290 M. (2016). The Variational Gaussian Process. International Conference on Learning Representations.
 291 Waterhouse, S., MacKay, D., and Robinson, T (1996). Bayesian methods for mixtures of experts. In
 292 Neural Information Processing Systems.