1 Math

$$\sigma^s = \mathbb{E}[X^2] - \mathbb{E}[X]^2 \tag{1}$$

$$\frac{\partial}{\partial q_k} soft(q)_i = soft(q)_i(\delta_{i,k} - soft(q)_k) \tag{2}$$

$$KL(p||q) = \int p(x)\log(\frac{p(x)}{q(x)})dx \tag{3}$$

$$\tag{4}$$

2 Deep Generative Models

Boltzman dist: $p(x) = \frac{1}{Z}\exp(-E(x))$. Comp. of normal. Const. Z difficult. **Boltzmann machine** $E(x) = -x^T W x - b^T x$ x is 256^2 big. Instead **RBM** $E(x) = -x^T W h - b^T x - c^T h$ with latent h.

2.1 Variational Inference

How est. posterior: MCMC or var. infer.: $\phi^* = \arg\min_\phi KL(q(\theta|\phi)||p(\theta|x))$, rev divergence (underestimate var, overest. with forward). **ELBO** $\mathbb{E}_{q_\phi(\theta)}[\log p(x|\theta)] - KL(q_\phi(\theta)||p(\theta)) = \mathbb{E}_{q_\phi(\theta)}[\log p(x|\theta)] + \mathbb{E}_{q_\phi(\theta)}[\log p(\theta)] - \mathbb{E}_{q_\phi(\theta)}[\log q(\theta)]$ with that $\log p(x) = ELBO_{\theta,\phi}(x) + KL(q_\phi(\theta)||p(\theta|x))$. ELBO is vari. free Enrgy.

2.2 Normalizing Flows

$\log p(x) = \log \pi_o(z_o - \sum_i^K |\det \frac{df_i}{dz_{i-1}}|)$

3 Bayesian Deep Learning

Benefits of Bayesian: ensemble makes better accuracies, uncertainty estimates, sparsity makes model compression, active learning, distributed learning. **Epistemnic uncertainty** ignorance which model generated the data. More data reduces this. For safety critical stuff, small datasets. **Aleatoric uncertainty** ignorance about the nature of the data. *Heteroscedastic* uncertainty about specific data $\mathcal{L} = \frac{||y_i - \hat{y}_i||^2}{s\sigma_i^2} + \log\sigma_i$, *homoscedastic* uncertainty about the task, we might reduce by combining tasks. $\mathcal{L}$ same but without idx. **MC Dropout** have d. during inference (by Bernoulli as vari. dist.) Then model prec. $\tau = \frac{l^2 p}{2N\lambda}$.

4 Deep Sequential models

4.1 Autoregressive models

With sequential data we have: $x = [x_1, \ldots, x_k] \implies p(x) = \prod_{k=1}^D p(x_k|x_{j<k})$ thus no param sharing and no ∞ chains $\implies p(x)$ is tractable.
NADE: fixed masks, conditionals modeled as MoG. **MADE**: masked conv.
PixelRNN seq. order over rows and channel R,G and B. Conditionals modeled with LSTM. Slow train and gen, but good gen. **PixelCNN** model conds with masked convs. Is worse than RNN cause blind spot. Fix by having convs for left row and everything above cascading. **PixelCNN++** dropout/whole pixels/discr log mix likelihood.
PixelVAE VAE+PixelCNN as the networks