Deep Learning

Supervised Learning

Optimization

GD + momentum: best 1st order conv. in theory
Adagrad: invariant to κ, vanishing η
Preconditioned GD/adaptive algorithms: can use a larger $\eta < 1/a$ for curvature $A \succeq \nabla^2 f \succeq aA$
Adam = Adagrad + momentum, can fail to conv.
Lottery ticket hypothesis. Small subnetwork that wins initialization lottery can have same performance.
Mode connectivity. Local minima are connected by simple paths of near-same cost.
Generalization Can't explain double descent. Modern view: flat minima

Unsupervised Learning

Distri. learning, poor classification. Let $KL(Q, P) \leq \varepsilon$. For classification, $TV(Q_\theta(\cdot \mid x)||P_\theta(\cdot \mid x)) = \sup_\Omega |Pr_{h\sim Q}[\Omega] - Pr_{h\sim P}[\Omega]|$.
From Pinsker's inequality,
$TV(Q||P) \leq \sqrt{\frac{1}{2}KL(Q||P)} \leq \sqrt{\frac{1}{2}\epsilon}$.

K-Means (Structure Learning)

Distance metrics include Euclidean, Manhattan, Minkowski. Criteria: intra-cluster cohesion, inter-cluster separation. Cons: sensitive to outliers, bad for non-spherical clusters.

Representation Learning

Want: interpretive, have downstream use, hierarchy, semantic clusterability, linear interpolation, disentangled
Disentangled representations formalized as $p_\theta(\mathbf{z}) = \prod_i p_\theta(z_i)$ for the prior and $\int_x q_\theta(z|x)p(x)dx$ is approximately a prod. dist. for the posterior. Weak evidence seems to correlate with performance on downstream tasks. Suffers from parametrization variance. Measures (are all correlated). Ex: BetaVAE metric assuming ground truth variation factors:

1. generate v_1, v_2 where k factor is the same, generate x_1, x_2
2. infer latents z_1, z_2 using model
3. calculate (z_{avg}, k) and train linear predictor, evaluate.

Sparse Coding

Learn dictionary D of features s.t. sample $x \approx Dh$ where $\|h\|_0$ is small. *Learning* objective is $\min_D \frac{1}{T}\sum_{t=1}^{T} \min_{h^{(t)}} \frac{1}{2}\|x^{(t)} - Dh^{(t)}\|_2^2 + \lambda\|h^{(t)}\|_1$ (prefer not L0 for convexity) with ISTA (LASSO) algorithm and warm starts.

Autoencoders

Learn features s.t. input is reconstructable from them (encoder, decoder). Add constraints such that identity is not learned (like sparsity).

Variant.	*Pros*	*Cons*
weight tying		
undercomplete rep.	cannot memorize	bad with alt. inp
overcomplete rep.		could memorize
denoising autoenc.	robust to noise	
variational	learn dist.	

Bayesian Networks (Distribution Learning)

BNs are **DAGs** where directed links correspond to conditional dependence. The probability distribution can be factorized (each node conditioned on its parents). Easy to sample from.

Sigmoid Belief Network

Bipartite latent-variable model (directed RBM), sigmoid activations.

Latent Dirichlet Allocation

Given $\alpha \in \mathbb{R}^K$, $\beta \in \mathbb{R}_+^{N\times K}$ for K topics and N vocabulary words:

1. Sample $\theta \sim \text{Dir}(\alpha)$ (get proportion of topics).
2. For each word x, sample topic $z \sim \text{Cat}(\theta)$, then sample $x \sim \text{Cat}(\beta_z)$.

Variational Autoencoders

BNs with Gaussian layers (assume diagonal covariance for tr. eff.).
Is an **encoder** as ELBO likelihood can be rewritten as $\mathbb{E}_{q_\theta}[\log \frac{p_\theta(x,h)}{q_\theta(h|x)}] = obj - \beta R$ where $obj = \mathbb{E}_{q_\theta} \log p_\theta(x|h)$ and $R = KL(q_\theta||p_\theta(h))$ for each layer. $\uparrow \beta = \uparrow$ disentanglement.
Learning: Reparam $\nabla_\theta \mathbb{E}_\theta$ to reduce variance

Markov Random Fields (Distribution Learning)

undirected, $p(x) = \frac{1}{Z}\prod_{(i,j)\in E(G)} \phi_{ij}(x_i, x_j)$ factors into potential functions over maximal cliques.

Jaynes principle. Distr. $p = \arg\max_p H(p)$ s.t. mean of each clique is μ_C is $p(x) \propto \exp\left(\sum_C w_C \phi_C(x_C)\right)$.

Restricted Boltzmann Machines

A RBM is a MRF latent-variable model where graph is bipartite. Assume $E(v,h) = -a'h - b'v - h'Wv$ then $P(v,h) = \frac{1}{Z}e^{-E(v,h)}$. $p(x)$ hard, but easy to sample posterior:
$P(h_j = 1|v) = \frac{1}{1+\exp -W_{,j}v - a}$, $P(v_j = 1|h) = \frac{1}{1+\exp -W_{i,}h - b}$

GANS (Distribution Learning)

Train a metric for semantic image similarity instead of ℓ_2 loss.

W-GAN. $\min_{g\in G} \underbrace{\max_{f\in F} |\mathbb{E}_{P_g}[f] - \mathbb{E}_{P_{samples}}[f]|}_{d_F(P_{samples}, P_g)}$

Also $\phi(f)$ variants for monotone ϕ e.g. log for DC-GAN

d_F is TV distance for $F = \{f : |f_\infty| \leq 1\}$, and Wasserstein distance for $F = \{f : Lip(f) \leq q\}$, JS divergence for F unconstrained. F has *distinguishing power* against G if $\forall g, h \in G \ : d_F(P_g, P_h) \gtrsim W_1(P_g, P_h)$.

Statistical Considerations (discriminator choice)

Weak discr. Generator with support size m fools NN discr. with $\leq m$ parameters. Weak f leads to mode collapse (cannot distinguish between small-support distribution and real distribution).

Large discr. Large discr. leads to poor generalization (overfitting).

Discr. for 1-to-1 G. $\exists$ small F with distinguishing power against G (1-to-1 NN) s.t. w poly(d) samples, $d_F \leq \epsilon \to W_1 \leq O(\sqrt{\epsilon})$.

Algorithmic Considerations (discriminator choice)

Take multiple steps for f for every g. Clip weights for Lipschitzness.

Problems: *unstable training* (saddle point problem), *vanishing gradient* (if the discriminator is too good, generator gradients are small), *mode collapse* (unclear if stat or alg problem).

Evaluation

No $p(x)$. Diagnose small support w/ birthday paradox. Not memorizing if *interpolation in latent* gives meaningful images without sharp transitions. *Inception score*: inception network $p(y|x)$ should be sure of labels and generate a good mix of labels

Invertible Models (Distribution Learning)

Marry likelihood based approach with GANs by assuming $g^{-1} = f$ is invertible and $P_g(x) = \phi(f(x))|\det(J_x(f(x)))|$. Max likelihood is $\max_\theta \sum_i^N \log P_g(x)$.

Transform.	*Pros*	*Cons*
linear	det easy	poor rep.
elementwise	J diag	poor rep.
NICE	J lower triag. (det prod of diag)	

Variational Methods

Inference: Partition Function

For *self-reducible* problems, we can compute marginals by computing partition functions.

Gibbs variat. principle For $p = \frac{1}{Z}\psi(x)$, $\log Z = \max_q[\mathbb{E}_q \log \psi(x) + H(q)]$ (Gibbs free energy) since $0 \leq KL(q\|p) = \mathbb{E}_q \log q - \mathbb{E}_q \log p$.

Mean-field approx. assume $q = \prod_i q_i$.

Inference: Posterior

Using $p(z|x) = \frac{p(z)p(x|z)}{p(x)}$ and formulae for KL, $\arg\min_{q(z|x)} KL(q(z|x)||p(z|x)) = \arg\min\{-H(q) - \mathbb{E}_q \log$
Can use coordinate ascent if using the mean-field approx.

KL order. Minimizing $KL(q||p)$ will have $q = 0$ where $p = 0$ whereas $KL(p||q)$ will have $q \neq 0$ where $p \neq 0$.

ELBO Using Bayes rule and Gibbs if given $p(z,x)$ then $\log p(x) = \max_{q(z|x)} H(q|z) + \mathbb{E}_q \log p(x,z)$.

Learning: Params for BNs

$\max_{\theta\in\Theta} \sum \log p(x_i) = \max_{\theta\in\Theta} \max_{q(z|x)} \sum H(q) + \mathbb{E}_q[\log p_\theta(x_i, z)] \Rightarrow$ EM

MCMC

Markov chain monte-carlo methods for sampling (inference).

Markov if $P(X_t|X_{<t}) = P(X_t|X_{t-1})$. **Homogenous** if $P(X_t|X_{t-1}$ does not depend on t. A **stationary distribution** satisfies $\pi T = \pi$. Unique if graph is irreducible (connected) and aperiodic (acyclic).

Detailed balance. A sufficient condition for π is $\pi_i T_{ij} = \pi_j T_{ji}$.

Metropolis-Hasting (Inference: Sampling)

Want to sample from stationary distribution up to a (unknown) constant of proportionality $\pi(x = i) = \frac{b(i)}{Z}$.

For $\alpha(i,j) = \min\left(\frac{\pi_j q(j,i)}{\pi_i q(i,j)}, 1\right) = \min\left(\frac{b(j)q(j,i)}{b(i)q(i,j)}, 1\right)$ then π is the stationary distribution of the walk (proven by detailed balance).

Algorithm

$P(X_n = j | X_{n-1} = i) =$

1. i to j with probability $q(i,j)$
2. w.p. $1 - \alpha(i,j)$ go back to state i, otherwise stay in j

Gibbs Sampling (Inference: Sampling)

When $P(x_i|\mathbf{x}_{-i})$ is easy, just do coordinate-wise updates (= MHs with appropriate kernel).

Mixing time. High if there's poor conductance $\phi(S) = \frac{\sum_{i\in S, j\notin S} T_{ij}}{\sum_{i\in S}\pi_i}$.

Langevin Dynamics (Inference: Sampling)

Want to sample from $p(x) = \frac{1}{Z}\exp(-f(x))$ over differentiable continuous domain. Algorithm is gradient descent with Gaussian noise $x_{t+1} = x_t - \eta\nabla f(x_t) + \sqrt{2\eta}\xi_k$. Works well for unimodal functions, hardness comes from multimodality (hard to climb hills).

Tempering

Potential solution for multimodality. Algorithm involves $p_k(x) \propto e^{-f(x)/c}$ for different k, and swapping occasionally. Algorithm is (1) stay on chain w.p. 1/2, (2) switch to chain k' w.p. $\min(\frac{p_{k'})(x)}{p_k)(x)}, 1)$ where x is current point. Stat. distri. is $P(x,k) = \frac{1}{K}p_k(x)$.

Langevin Tempering Runtime. For $p(x) \propto e^{-f(x)}$ be K shifts of d dimension log-concave distrib., then runtime is poly(K,d) until stat. distrib (works as take the road less hilly, for same shape modes).

Learning: Energy Models

Solve $\max_{\theta\in\Theta}\sum_{i=1}^n \log p_\theta(x_i)$ where $p_\theta(x) \propto \exp(-E_\theta(x))$.

$\nabla_\theta \log Z_\theta = \frac{1}{Z_\theta}\int_x \exp(-E_\theta(x))\nabla_\theta(-E_\theta(x))dx = \mathbb{E}_{p_\theta}[-\nabla_\theta E_\theta(x)]$. Hence, gradient of objective is

$$\nabla_\theta f \approx \mathbb{E}_{p_{data}}[-\nabla_\theta E_\theta(x)] - \mathbb{E}_{p_\theta}[-\nabla_\theta E_\theta(x)],$$

where we use Langevin to sample from p_θ.

Or minimize $\mathbb{E}_{p_{data}}\|\nabla_x \log p_{data}(x) - (-E_\theta(x))\|^2$ (which is friendly to gradients) w/ Gaussian conv. for smoothing out bad estimates and tempering for multimodality.

Learning: RBMs

Can factorize $p(x) = \exp(F(\mathbf{x})/Z$ such that gradient is $\frac{1}{n}\left(\sum_i -\nabla_\theta F_\theta(x_i)\right) - \mathbb{E}_{p_\theta}[-\nabla_\theta F_\theta(x)]$. Sample $x \sim p_\theta$ in second part using Gibbs. Algorithm **CD-k** is GD + Gibbs with k steps. In general, greater k gives less biased gradients, in practice $k = 1$ works well. **Persistent CD** is warm-starting the Gibbs chain.

Learning: DBNs

DBNs are stacked SBNs (directed, conditional restricted probabilities $P(h^{(1)} = 1|\mathbf{h}^{(2)}) = \sigma(\mathbf{b}^{(1)} + {W^{(2)}}^\top \mathbf{h}^{(2)})$ with an RBM at the top. Joint distribution factorizes.

The **variational intuition** involves ELBO and noting that the joint distribution can untied and then viewed as maximizing expectation of log probabilities.

Algorithm is layer-wise training. Train bottom up assuming undirected (Gibbs sampling) freeze weights after convg. and move up. Sample using Gibbs on top layer but conditional probs on others.

Self-Supervised Learning

No labels, but train on auxiliary supervised tasks so model learns good representation for downstream tasks

Autoregressive Models (Sequential Learning)

Factor joint distribution of data as $p(x_1, x_2, \ldots, x_t) = p(x_i \mid x_{<i})$.

Model.	$p(x_i \mid x_{<i})$
FV Sigmoid BN	$p(x_i \mid x_{<i}) = \sigma\left(\sum_{j=1}^i A_i x_j + c_i\right)$ extra layer
NADE	$h_i = \sigma(W_{\cdot,i}x_{<i} + c)$ s.t. $p(x_i\|x_{<i}) = \sigma(\alpha_i^T h_i + b_i)$
MADE	use autoencoder to specify $p(\hat{x}\|x)$, then mask for $p(x_i\|x_{<i})$ (fixed seq length)
PixelCNN	Conv. version of MADE, layers after first are not masked.

RNNs

Sequential model for arbitrary length sequences. Specified as $h_i = \tanh(W_{hh}h_{i-1} + W_{xh}x_i$, $o_i = W_{hy}h_i$ and $p(x_i|x_{<i}) \sim \text{softmax}(o_i)$.
Solve exploding/vanishing gradient problem (lack of theory) using *LSTMs*: train gate on the input, gate on hidden layer update, and when to forget the previous state.

NLP

Big models. Large models to better and improvements have not asymptoted.
Multi-step Q&A not solved – involves semantic understanding instead of just pattern matching.

Word embeddings

Want semantically meaningful vector reps. Objective is $\max_\theta \sum_t \log p_\theta(x_t|x_{t-1}, \ldots, x_{t-L})$ (generative model, can use cross-entropy).
Related tasks: pred. middle x (table below), predict subset of words. No generative model, need to evaluate using indirect means: *intrinsic* similar words (cosine), analogies or *extrinsic* train for downstream tasks.

Model.	p_θ
CBOW	$p_\theta(x_t\|x_{t-1}, \ldots, x_{t-L}) \propto \exp(v_{x_t}, \sum w_{x_i})$
Skip-Gram	$p_\theta(x_i\|x_t) \propto \exp(v_{x_t}, w_{x_t})$
ELMo	train LSTM on $p_\theta(x_t\|x_{<t})$, and $p_\theta(x_t\|x_{>t})$ then concat.

Distri. hypothesis. Words are defined by its context, cosine sim. correlates with human similarity. Can try to find low-dimension approx of similarities.
SkipGram/CBOW reduce dim. Obj. can be rewritten as maximization of inner products for words that co-occur: $\langle v_i, w_j \langle \approx PMI$.

Generative Model

Let $P(w) \propto \exp(v_w \cdot c_t)$ where c_t is a discourse vector that does a random walk (s.t. $\|c_t - c_{t-1}\| \ll \frac{1}{\sqrt{d}}$ and $\|v_w\| = \Theta(\sqrt{d})$).
Co-occurence capture PMI. Let $P(w, w')$ be the co-occurrence, then $\log p(w, w') = \frac{1}{2d}\|v_w + v_{w'}\|^2 - 2\log Z \pm \epsilon$ (norm of vec. is freq; spatial orient is meaning).

Transformers

Can use for machine translation (RNN-based encoder/decoders).

1. involves *attention* $A(Q, K, V) = \text{softmax}(QK^TV)$ (query q, key k and value v + similarity).
2. positional encoding (add sinusoid)
3. non linearities

Learn word embeddings. Obv. method: *GPT2* use decoder on words to learn classes. *BERT* uses encoder to predict random % of words given rest.

Vision

Task	*Comments*
Inpainting	L2 loss (blurry), GANs loss typically Region: fixed (not gen.), random (square borders problem), rand. silhouette (ill-defined)
Jigsaw	shortcuts (boundary, lines, chromatic abb.)
Rotation	no obv. cheats
CD	"distortion" should still have sim. features.

Arch. matters. Different self-supervising tasks need different architectures.

Adversarial Robustness

Perturbations imperceptible to humans cause NN to fail.
Attacks: Literature focuses on white box attacks: adversary produces perturbation δ with ϵ max norm (fast gradient sign method (take largest ϵ step), PGD ($\arg\max_\delta L(x + \delta; \theta)$ s.t. $\|\delta\| \leq \epsilon$).

Defenses

Failed defenses: non-differentiability, add randomness, very deep networks (gradient obfuscation).
Adversarial Training: Min max training on defender and adversary (empirical). Not yet broken but slow.
Provable Defenses: decision boundary is not contained within an ε-norm ball:

- Convex Polytope: track a ball around the input to the output
- Interval Bound Propagation: axis-aligned polytope is faster
- Randomized smoothing: add noise, integrate (via sampling), scalable, smooth decision boundary, reduces accuracy but might improve generalization

Adding robustness typically decreases standard accuracy.

Train w/o Non-Robust Features Networks might use non-robust features which correlate with label on average but can flip within ε-ball. Removing non-robust features leads to adversarially robust generalization, with accuracy tradeoff.