## General definitions

$\mathcal{N}(y;\Sigma,\mu) = \frac{1}{\sqrt{(2\pi)^n|\Sigma|}} \exp\Big( -\frac{1}{2}(y-\mu)^T\Sigma^{-1}(y-\mu)\Big)$
need only $n^2$ params for joint instead of $2^n - 1$
entropy $H(q) = -\int q(\theta)\log q(\theta)d\theta = \mathbb{E}_{\theta\sim q}[-\log q(\theta)]$
mutual info $I(X;Y) = H(X) - H(X|Y)$ (symmetric)
KL div. $KL(q||p) = \int q(\theta)\log\frac{q(\theta)}{p(\theta)}d\theta = \mathbb{E}_{\theta\sim q}[\log\frac{q(\theta)}{p(\theta)}]$
non-negative, zero iff q and p agree a.e., not symmetr.
Jensen's inequality: $g(\mathbb{E}[X]) \leq \mathbb{E}[g(X)]$ for g convex, else flipped (e.g. $\log(\mathbb{E}[X]) \geq \mathbb{E}[\log(X)]$
Hoeffding's inequality: for $f$ bounded in $[0,C]$
$p\big(|\mathbb{E}_p[f(X)] - \frac{1}{N}\sum_i f(x_i)| > \epsilon\big) \leq 2\exp(-2N\epsilon^2/C^2)$
Robins-Monro conditions: $\sum_t \epsilon_t = \infty, \sum_t \epsilon_t^2 < \infty$

## Bayesian linear regression (BLR)

BLR makes same assumptions as ridge regression: cond. idd Gaussian noise, Gaussian prior
RR = MAP estimation for LR ($y = w^Tx$), i.e. returns single model, no uncertainty qualification (collapses all uncertainty onto mode of posterior $p(w|X,y)$)
BLR reasons about full $p(w|X,y) \sim \mathcal{N}(y;\mu,\Sigma)$
$\mu = (X^TX + \sigma_n^2I)^{-1}X^Ty$ ; $\Sigma = (\sigma_n^{-2}X^TX + I)^{-1}$
prediction: $p(y^*|X,y,x^*) \sim \mathcal{N}(\mu^Tx^*, x^{*T}\Sigma x^* + \sigma_n^2)$
$\Rightarrow$ separation of epistemic uncertainty (about $f^*$/model due to lack of data) and aleatoric uncertainty (irreducible noise from $y^* = f^* + \epsilon$)
independent noise $\Rightarrow$ recursive Bayesian updates, i.e. use posterior from last iteration as prior:
$p(w|y_{1:j+1}) = \frac{1}{Z}p(w|y_{1:j})p(y_{j+1}|w,y_{1:j})$
$w^{(j+1)} = f(w^{(j)}, y_{j+1}, x_{j+1})$

## Kalman/Bayesian filtering

motion model: $x_{t+1} = Fx_t + \epsilon_t$ ; $\epsilon_t \sim \mathcal{N}(0,\Sigma_x)$
sensor model: $y_t = Hx_t + \eta_t$ ; $\eta_t \sim \mathcal{N}(0,\Sigma_y)$
$F, H$ known and deterministic, KF resembles HMM
Kalman update: $\mu_{t+1} = F\mu_t + K_{t+1}(y_{t+1} - HF\mu_t)$
$\Sigma_{t+1} = (I - K_{t+1})(F\Sigma_tF^T + \Sigma_x)$
Kalman gain: $K_{t+1} = (F\Sigma_tF^T + \Sigma_x)H^T(H(F\Sigma_tF^T + \Sigma_x)H^T + \Sigma_y)^{-1}$, compute $\Sigma_t, K_t$ offline (indep. of obs.)
BLR = KF with $w$ as hidden vars., $F = I, \sigma_x^2 = 0$
KF special case of GP with cond. indep. structure

## Gaussian processes

instead of random $w$, think of random responses
$f = Xw \sim \mathcal{N}(0, \sigma_p^2XX^T)$ s.t. $XX^T = K$, $K_{ij} = x_i^Tx_j$
Gaussians over functions instead of RVs/points
prior $p(f)$ encodes smoothness ass. on functions
posterior $p(f|data)$ encodes agreement with data
uncertainty, tractable inference for finite marginals
mean func. $\mu$, covariance func. $k$ (BLR for lin. kernel)
prediction: closed form, posterior cov. $k'$ indep. of $y_A$
$\mu'(x) = \mu(x) + k_{x,A}(K_{AA} + \sigma^2I)^{-1}(y_A - \mu_A)$
$k'(x,x') = k(x,x') - k_{x,A}(K_{AA} + \sigma^2I)^{-1}k_{x,A}^T$
sampling from GP: $f = [f_1, ..., f_n] \sim \mathcal{N}(0, K_x)$
product rule $\Rightarrow$ forward sampling (fully sequential): sampling from univariate Gaussians $f_n \sim p(f_n|f_{1:n-1})$
opt. kernel params: 1) CV on predictive performance
2) Bayesian, i.e. max. marg. likelihood:
$\hat\theta = \arg\max_\theta \int p(y|f,X)p(f|\theta)df$ (general)
$\hat\theta = \arg\min_\theta \frac{1}{2}\log|K_y(\theta)| + \frac{1}{2}y^TK_y(\theta)y$ (Gaussians)
solve using GD, i.e. $\theta^{(t+1)} = \theta^{(t)} - y_t\nabla L(\theta)$
reduces overfitting, but depends heavily on prior
comp. cost: LSE in $|A|$ unknowns $\Rightarrow \mathcal{O}(|A|^3)$
acceleration methods: 1) parallelization (still $\mathcal{O}(|A|^3)$)
2) local GP methods: only consider $x'$ if $|k(x,x')| > \tau$
3) kernel approx.: Fourier for stationary kernels
4) inducing point: ignore points (e.g. in clusters)

## Approximate inference

Variational inference:
for BLR and GPR everything closed form, generally not the case $\Rightarrow$ need approximations
can evaluate joint $p(y,\theta)$ but not normalizer $Z$
replace high-dim. integrals by optimization
$p(\theta|y) = \frac{1}{Z}p(y,\theta) \approx q(\theta|\lambda)$
$q^* = \arg\min_q KL(q||p) = \arg\min_\lambda KL(q_\lambda||p)$
prefer $\arg\min_q KL(p||q)$(p in q), but harder to opt.
$q^* = \arg\max_q \mathbb{E}_{\theta\sim q}[\log p(y|\theta)] - KL(q||p(\cdot))$
regularizer: want $q$ close to prior $p(\cdot)$
Jensen's inequality $\Rightarrow$ ELBO $L(q) \leq \log p(y)$
to use SGD to max. $L(\lambda)$, need reparameterization:
$q(\theta|\lambda) = \phi(\epsilon)|\nabla_\epsilon g(\epsilon;\lambda)|^{-1}$ ; $\epsilon \sim \phi, \theta = g(\epsilon,\lambda)$
$\nabla_\lambda\mathbb{E}_{\theta\sim q_\lambda}[f(\theta)] = \nabla_\lambda\mathbb{E}_{\epsilon\sim\phi}[f(g(\epsilon;\lambda))] = \mathbb{E}_{\epsilon\sim\phi}[\nabla_\lambda f(g)]$

Laplace approximation:
2nd-order Taylor expansion around $\hat\theta$ to construct Gaussian: $q(\theta) \sim \mathcal{N}(\theta;\hat\theta,\Lambda^{-1})$ ; $\Lambda = -\nabla\nabla\log p(\hat\theta|y)$
$Z$ const. in optimization for $\hat\theta$ and calculation for $\Lambda$
overconfident, does not consider cov. when seeking $\hat\theta$

## Markov Chain Monte Carlo (MCMC)

vs. VI: returns accurate result, higher comp. cost
seek to approx. p using samples constructed by a markov chain (law of large numbers, need $\theta^{(i)}$ indep.)
$p(y^*|X,y,x^*) = \mathbb{E}_{\theta\sim p(\cdot|X,y)}[p(y^*|x^*,\theta)] \approx \frac{1}{N}\sum_i f(\theta^{(i)})$
need $N \geq \frac{C^2}{2\epsilon^2}\log\frac{2}{\delta}$ for error $\leq \epsilon$ with prob. $\geq 1-\delta$
create MC with $\pi = P(x) = \frac{1}{Z}P(y)P(x|y) = \frac{1}{Z}Q(x)$
guaranteed by detailed balance:
$\frac{1}{Z}Q(x)P(x'|x) = \frac{1}{Z}Q(x')P(x|x')$
Metropolis-Hastings: (perf. highly dependent on R!)
1) given $X_t = x$, sample proposal $x' \sim R(X'|X=x)$
2) set $X_{t+1} = x'$ with prob. $\alpha$, else $X_{t+1} = x$
$\alpha = \min\{1, \frac{Q(x')R(x|x')}{Q(x)R(x'|x)}\}$
Gibbs:
1) init. assignment $x^{(0)}$ to all variables
2) fix observed vars. $X_B$ to their observed values $x_B$
3) either random order (detailed balance): pick $i$ unif. at random, update $x_i \sim P(X_i|v_i)$ or practical variant (no det. bal. but has correct $\pi$): set $x^{(t)} = x^{(t-1)}$, then update all $x_i$ except those in $B$
$Z = \sum_x Q(X_i = x, v_i)$ is easy to calculate $\Rightarrow$ sampling from $X_i$ given assignment to all other vars. is efficient
$x^{(t)}$ dep. on $x^{(t-1)} \Rightarrow$ loln, Hoeffding's no longer hold
only ergodic MC $\lim_{N\to\infty}\frac{1}{N}\sum_i f(x_i) = \mathbb{E}_{x\sim\pi}[f(x)]$
MCMC for continuous RVs:
proposal distr. either random (simple, uninformed) or in gradient direction (MALA):
$R(x'|x) \sim \mathcal{N}(x'; x - \tau\nabla f(x), 2\tau I)$
$\alpha = \min\{1, e^{f(x)-f(x')}\}$ for $p = \frac{1}{Z}e^{-f(x)}$
converges to $\pi$ for f convex $\Leftrightarrow$ p log-concave
requires access to full energy func. f
$\Rightarrow$ SGLD:
replace full gradient by unbiased estimate (mini-batch), always accept but reduce step size $\eta_t$ over time
$\Rightarrow$ SGD + Gaussian noise, converges for $\eta_t \in \mathcal{O}(t^{-1/3})$

**Bayesian deep learning**
heteroscedastic noise: noise depends on input
$\Rightarrow$ model mean and (log) var as outputs of NN
$p(y|x,\theta) = \mathcal{N}(y; f_1(x,\theta), e^{f_2(x,\theta)})$
MAP est.: $\hat{\theta} = \arg\min_\theta - \log p(\theta) - \sum_i \log p(y_i|x_i,\theta)$
prediction $p(y^*|X,y,x^*) = \int p(y^*|x^*,\theta)p(\theta|X,y)d\theta$
integrals intractable $\Rightarrow$ approximate inference:
Bayes by backprop:
$llh \overset{VI}{\approx} \mathbb{E}_{\theta\sim q(\cdot|\lambda)}[p(y^*|x^*,\theta)] \overset{MC}{\approx} \frac{1}{m}\sum_j p(y^*|x^*,\theta^{(j)})$
$\Rightarrow$ mixture of Gaussians
$\mathbb{E}[llh] \approx \bar{\mu}(x^*) = \frac{1}{m}\sum_j \mu(x^*,\theta^{(j)})$
$\text{Var}(llh) = \text{Var}(\mathbb{E}_y[y^*|x^*,\theta]) + \mathbb{E}_\theta[\text{Var}(y^*|x^*,\theta)]$
$\approx \frac{1}{m}\sum_j(\mu(x^*,\theta^{(j)}) - \bar{\mu}(x^*))^2 + \frac{1}{m}\sum_j \sigma(x^*,\theta^{(j)})$
MCMC for BNNs:
apply SGLD, MALA (only need stoch. grads of joint)
$\Rightarrow$ produce sequence $\theta^{(1)},...,\theta^{(T)}$, impossible to store all samples/models, hard to determine burn-in
1) subsampling: keep only a subset of $m < T$ models
2) Gaussian approx.: running averages for $\mu_i, \sigma_i^2$
specialised inference techniques for BNNs:
dropout regularization: randomly ignore hidden units during each SGD iteration (forward and backprop.)
view as VI: $q(\theta|\lambda) = \prod_j p\delta_0(\theta_j) + (1-p)\delta_{\lambda_j}(\theta_j)$
probabilistic ensembles of NNs: variation of $\theta^{(j)}$ shows uncertainty $\Rightarrow$ bootstrap, get MAP on $D_j$ to get $\theta^{(j)}$

**Active learning**
use epistemic and aleatoric uncertainty to decide which data to collect (e.g. where to place sensors)
want points $S$ which max. info gain (NP-hard)
greedy algo./uncertainty sampling: choose $x_{t+1} = \arg\max_x \sigma_t^2(x)$ (only considers epistemic uncertainty)
for heteroscedastic case, need $x_{t+1} = \arg\max_x \frac{\sigma_f^2(x)}{\sigma_n^2(x)}$ as aleatoric uncertainty no longer const. in $x$

**Bayesian optimization** (exploration-exploitation)
use that similar alternatives have similar performance
multi-armed bandits: pick $x_t$, observe $y_t = f(x_t) + \epsilon_t$
cum. regr. $R_T = \sum_t \max_x f(x) - f(x_t)$ ; want $\frac{R_T}{T} \to 0$

acquisition functions:
GP-UCB: focus exploration on regions where upper conf. bound $\geq$ best lower conf. bound
$x_t = \arg\max_x \mu_{t-1}(x) + \beta_t\sigma_{t-1}(x)$ (gen. non-convex)
how to choose $\beta_t$?, naturally trades off e-e
Thompson: $x_t = \arg\max_x \tilde{f}(x)$ ; $\tilde{f} \sim p(f|x_{1:t}, y_{1:t})$
randomness in $\tilde{f}$ enough to trade off e-e

**Markov decision processes (MDPs)**
states, actions, transition probas. and reward function
$V^\pi(x) = r(x,\pi(x)) + \gamma\sum_{x'} P(x'|x,\pi(x))V^\pi(x')$
can compute $V^\pi = r^\pi + \gamma T^\pi V^\pi$ exactly by solving LSE
approx. by fixed point iteration: $V_t^\pi = r^\pi + \gamma T^\pi V_{t-1}^\pi$
converges exponentially
every $V$ induces a (greedy) $\pi$ and vice versa:
$V \rightsquigarrow \pi_g(x) = \arg\max_a r(x,a) + \gamma\sum_{x'} P(x'|x,a)V(x')$
Bellman thm: $\pi$ optimal $\Leftrightarrow$ greedy w.r.t. induced $V$
policy iteration: init $\pi$, until convergence:
1) comp. $V^\pi(x)$ 2) comp. $\pi_g$ w.r.t. $V^\pi$ 3) $\pi = \pi_g$
$V^\pi$ monotonically increases, converges to optimal $\pi$
complexity: need to solve LSE for $V$
value iteration: Bellman+FPI $V_0(x) = \max_a r(x,a)$
$Q_t(x,a) = \max_a r(x,a) + \gamma\sum_{x'} P(x'|x,a)V_t(x')$
$V_t(x) = \max_a Q_t(x,a)$, break if $\|V_t - V_{t-1}\|_\infty \leq \epsilon$
$\rightsquigarrow \pi_g$, converges to $\epsilon$-optimal $\pi$ in $\mathcal{O}(ln^1/_\epsilon)$ iterations
POMDP: (control. HMM); $P(X_{t+1}|X_t,A_t)$, $P(Y_t|X_t)$
very powerful but generally extremely intractable
$\Rightarrow$ belief-state MDPs (use Bayesian filtering):
beliefs $P(X_t|y_{1:t})$ given noisy observations $y$
$b_{t+1}(x) = P(X_{t+1} = x|y_{1:t+1}) = \frac{1}{Z}P(y_{t+1}|x)P(X_{t+1} = x|y_{1:t})$ ; $r(b_t,a_t) = \sum_x b_t(x)r(x,a_t)$
most belief states never reached
dyn. progr., point based methods, policy grads

**Reinforcement learning**
credit assignment problem: which $a_i$ got me to this $r$?
data not iid, depends on our actions $\Rightarrow$ e-e dilemma
model-based RL: learn MDP from data
estimate $P(x'|x,a), r(x,a)$ e.g. by MLE (counts)
store r,P; solve est. MDP up to $|X|\cdot|A|$ times
$\epsilon$-greedy: random $a_t$ with prob. $\epsilon_t$, else best $a_t$
conv. to optimal $\pi$, considers suboptimal actions

$R_{max}$: "optimism in the face of uncertainty"
init $r(x,a) = R_{max}, P(x^*|x,a) = 1$, $\pi$ opt. w.r.t. $r,P$
repeat: exec. $\pi$, obs. $(x,a)$, update $r$, est. $P(x'|x,a)$, recompute $\pi$ w.r.t. $r,P$ after $n \in \mathcal{O}(\frac{R_{max}^2}{\epsilon^2}\log\frac{1}{\delta})$ obs.
model-free RL: est. $V^\pi$ directly given $\pi$
TD-learning: (on-policy), init $V_0^\pi$, $\pi \rightsquigarrow (x,a,r,x')$
$V_{t+1}^\pi(x) = (1-\alpha_t)V_t^\pi(x) + \alpha_t(r + \gamma V_t^\pi(x'))$
i.e. use bootstrapping, one-sample est. of long-term $r$
Q-learning: (off-policy) $a \rightsquigarrow (x,a,r,x')$
$Q_{t+1}(x,a) = (1-\alpha_t)Q_t(x,a) + \alpha_t(r + \gamma\max_{a'} Q_t(x',a'))$
choose $Q_0(x,a) = \frac{R_{max}}{1-\gamma}\prod_t(1-\alpha_t)^{-1}$ for e-e tradeoff
large state spaces: learn approx. $V(x;\theta), Q(x,a;\theta)$
neural-fitted Q-iteration (DQN): collect dataset D
$L(\theta) = \sum_{\cdot\in D}(r + \gamma\max_{a'} Q(x',a';\theta^{old}) - Q(x,a;\theta))^2$
max. bias, too optimistic about noisy est. of Q
DDQN: decouple max.: $a^*(\theta) = \arg\max_{a'} Q(x',a';\theta)$
$L(\theta) = \sum_{\cdot\in D}(r + \gamma Q(x',a^*(\theta);\theta^{old}) - Q(x,a;\theta))^2$
large action spaces: policy search, learn $\pi(x;\theta)$
1) policy gradients: $J(\theta) = \frac{1}{m}\sum_j r(\tau^{(j)})$ (on-policy)
$\nabla J(\theta) = \nabla\mathbb{E}_{\tau\sim\pi_\theta}[r(\tau)] = \nabla\mathbb{E}_{\tau\sim\pi_\theta}[r(\tau)\nabla\log\pi_\theta(\tau)]$
MDP structure $\Rightarrow \nabla\mathbb{E}_{\tau\sim\pi_\theta}[r(\tau)\sum_t \nabla\log\pi(a_t|x_t;\theta)]$
unbiased but very large variance $\Rightarrow$ baselines:
$\nabla J(\theta) = \nabla\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_t \gamma^t(G_t - b_t)\nabla\log\pi(a_t|x_t;\theta)\right]$
e.g. $G_t = \sum_{t'=t}\gamma^{t'-t}r_{t'}$ rews-to-go, $b_t = \frac{1}{T}\sum_t G_t$
2) actor-critic: (non-episodic)
$\nabla J(\theta_\pi) = \mathbb{E}_{(x,a)\sim\pi_\theta}[Q(x,a;\theta_Q)\nabla\log\pi(a|x;\theta_\pi)]$
$\theta_\pi \leftarrow \theta_\pi + \eta_t Q(x,a;\theta_Q)\nabla\log\pi(a|x;\theta_\pi)$ ; $\theta_Q \leftarrow \theta_Q - \eta_t(Q(x,a;\theta_Q) - r - \gamma Q(x',\pi(x';\theta_\pi);\theta_Q))\nabla Q(x,a;\theta_Q)$
off-policy AC: (DDPG, resp. TD3 to avoid max. bias)
$L(\theta_Q) = \sum_{\cdot\in D}(r + \gamma Q(x',\pi(x';\theta_\pi);\theta_Q^{old}) - Q(x,a;\theta_Q))^2$
$\nabla J(\theta_\pi) = \mathbb{E}_{x\sim\mu}[\nabla Q(x,\pi(x;\theta);\theta_Q)]$ (i.e. w.r.t. $\pi_G$)
only for determin. $\pi$, add action noise for exploration
random. $\pi$: A use reparam. to pull $\nabla_{\theta_\pi}$ into $\mathbb{E}_{a\sim\pi(x,\theta_\pi)}$
soft AC: $J_\lambda(\theta) = J(\theta) + \lambda H(\pi_\theta)$ (entropy regulariz.)
model-based deep RL: smaller sample complexity
MPC: $\max_{a_{0:\infty}}\sum_t \gamma^t r(x_t,a_t)$ s.t. $x_{t+1} = f(x_t,a_t)$
finite horizon, unroll: $\max_{a_{t:t+H-1}}\sum_\tau \gamma^\tau r(x_\tau(a_{t:\tau-1}),a_\tau)$
analytic grads local min., exploding/vanishing grads
use heuristics, e.g. random shooting
sparse r, add (off-policy) V estimate $+\gamma^H V(x_{t+H})$
unknown (f, r): regression (Bayesian learning, e-e)