1 Fundamentals

- **Normal**: $\frac{\exp(-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu))}{\sqrt{(2\pi)^k\det(\Sigma)}}$
- **Beta**: $\mathrm{Beta}(\theta;\alpha,\beta)\propto\theta^{\alpha-1}(1-\theta)^{\beta-1}$
- **Laplace**: $\frac{1}{2l}\exp(-\frac{|x-\mu|}{l})$
- Gaussian CDF has no closed-form
- Gaussian can be repr. using $O(n^2)$ params.; $\Sigma\in\mathbb{R}^{n\times n}$

Expectation
- $\mathbb{E}[\mathbf{AX}+\mathbf{b}]=\mathbf{A}\mathbb{E}[\mathbf{X}]+\mathbf{b}$; $\mathbb{E}[\mathbf{X}+\mathbf{Y}]=\mathbb{E}[\mathbf{X}]+\mathbb{E}[\mathbf{Y}]$
- $\mathbb{E}[\mathbf{XY}^\top]=\mathbb{E}[\mathbf{X}]\cdot\mathbb{E}[\mathbf{Y}]^\top$ (if $\mathbf{X}$, $\mathbf{Y}$ indep.)
- LOTUS: $\mathbb{E}[\mathbf{g}(\mathbf{X})]=\int_{\mathbf{X}(\Omega)}\mathbf{g}(\mathbf{x})\cdot p(\mathbf{x})dx$ (if $\mathbf{g}$ nice and $\mathbf{X}$ cont.)
- Tower rule: $\mathbb{E}_\mathbf{Y}[\mathbb{E}_\mathbf{X}[\mathbf{X}|\mathbf{Y}]]=\mathbb{E}[\mathbf{X}]$

Variance
- $\mathrm{Var}[\mathbf{X}]\doteq\mathbb{E}[(\mathbf{X}-\mathbb{E}[\mathbf{X}])(\mathbf{X}-\mathbb{E}[\mathbf{X}])^\top]$ $=\mathbb{E}[\mathbf{XX}^\top]-\mathbb{E}[\mathbf{X}]\cdot\mathbb{E}[\mathbf{X}]^\top=\mathrm{Cov}[\mathbf{X},\mathbf{X}]$
- $\mathrm{Var}[\mathbf{AX}+\mathbf{b}]=\mathbf{A}\mathrm{Var}[\mathbf{X}]\mathbf{A}^\top$
- $\mathrm{Var}[\mathbf{X}+\mathbf{Y}]=\mathrm{Var}[\mathbf{X}]+\mathrm{Var}[\mathbf{Y}]+2\mathrm{Cov}[\mathbf{X},\mathbf{Y}]$
- $\mathrm{Var}[\mathbf{X}+\mathbf{Y}]=\mathrm{Var}[\mathbf{X}]+\mathrm{Var}[\mathbf{Y}]$ (if $\mathbf{X}$, $\mathbf{Y}$ indep.)
- Law of total variance, LOTV: $\mathrm{Var}[\mathbf{X}]=\mathbb{E}_\mathbf{Y}[\mathrm{Var}_\mathbf{X}[\mathbf{X}|\mathbf{Y}]]+\mathrm{Var}_\mathbf{Y}[\mathbb{E}_\mathbf{X}[\mathbf{X}|\mathbf{Y}]]$

Covariance
- $\mathrm{Cov}[\mathbf{X},\mathbf{Y}]\doteq\mathbb{E}[(\mathbf{X}-\mathbb{E}[\mathbf{X}])(\mathbf{Y}-\mathbb{E}[\mathbf{Y}])^\top]$ $=\mathbb{E}[\mathbf{XY}^\top]-\mathbb{E}[\mathbf{X}]\cdot\mathbb{E}[\mathbf{Y}]^\top$
- $\mathrm{Cov}[\mathbf{X},\mathbf{Y}]=\mathrm{Cov}[\mathbf{Y},\mathbf{X}]$; $\mathrm{Cov}[\mathbf{X},\mathbf{X}]\geq\mathbf{0}$
- $\mathrm{Cov}[\mathbf{AX}+\mathbf{c},\mathbf{BY}+\mathbf{d}]=\mathbf{A}\mathrm{Cov}[\mathbf{X},\mathbf{Y}]\mathbf{B}^\top$
- *Correlation* is normalized covariance: $\mathrm{Cor}[\mathbf{X},\mathbf{Y}](i,j)\doteq\frac{\mathrm{Cov}[X_i,Y_j]}{\sqrt{\mathrm{Var}[X_i]\mathrm{Var}[Y_j]}}\in[-1,1]$
- *Uncorrelated* iff $\mathrm{Cov}[\mathbf{X},\mathbf{Y}]=\mathbf{0}$.
- **Change of variables**: Let $\mathbf{g}$ be diff. and inv. Then for $\mathbf{Y}=\mathbf{g}(\mathbf{X})$: $p_\mathbf{Y}(\mathbf{y})=p_\mathbf{X}(\mathbf{g}^{-1}(\mathbf{y}))\cdot|\det(\mathbf{Dg}^{-1}(\mathbf{y}))|$ where $\mathbf{Dg}^{-1}(\mathbf{y})$ is the Jacobian of $\mathbf{g}^{-1}$ at $\mathbf{y}$.

Posterior $p(\mathbf{x}|\mathbf{y})$: updated belief about $\mathbf{x}$ after observing $\mathbf{y}$. **Prior** $p(\mathbf{x})$: initial belief about $\mathbf{x}$.
Conditional likelihood $p(\mathbf{y}|\mathbf{x})$: how likely the observations $\mathbf{y}$ are under a given value $\mathbf{x}$.
Joint likelihood $p(\mathbf{x},\mathbf{y})=p(\mathbf{y}|\mathbf{x})p(\mathbf{x})$
Marginal likelihood $p(\mathbf{y})$: how likely the observations $\mathbf{y}$ are across all values of $\mathbf{x}$. Can be computed with $p(\mathbf{y})=\int_{\mathbf{X}(\Omega)}p(\mathbf{y}|\mathbf{x})\cdot p(\mathbf{x})d\mathbf{x}$.

Bayes' rule: $p(\mathbf{x}|\mathbf{y})=\frac{p(\mathbf{y}|\mathbf{x})\cdot p(\mathbf{x})}{p(\mathbf{y})}$
- If prior $p(\mathbf{x})$ and posterior $p(\mathbf{x}|\mathbf{y})$ from same family of distr., prior is **conjugate prior** to the likelihood $p(\mathbf{y}|\mathbf{x})$.
- Beta distr. is a conjugate prior to binomial likelihood.
- Under some conditions, the **Gaussian is self-conjugate** (Gaussian prior and likelihood $\rightarrow$ posterior Gaussian).

Choice of prior
- Choosing non-informative prior in absence of evidence is **principle of indifference/insufficient reason**.
- **Improper prior:** It is not required that the prior is a valid distr. (i.e., integrates to 1). We can still derive meaning from the posterior if it's valid.
- **Maximum entropy principle:** choose a prior from all possible distributions that are consistent with prior knowledge, s.t. one that makes the least "additional assumptions", i.e., the prior that is least "informative".

Gaussian properties
- **Gaussians have max. entropy among all distr.** with known mean and variance: $1/2\cdot\log((2\pi e)^d\det(\boldsymbol{\Sigma}))$
- Jointly Gaussian random vectors, $\mathbf{X}$ and $\mathbf{Y}$, are independent iff $\mathbf{X}$ and $\mathbf{Y}$ are uncorrelated.
- Closed under marginalization and conditioning.

Let $\mathbf{X}$ be Gaussian and index sets $A,B\subseteq[n]$.
For any **marginal distr.** $\mathbf{X}_A\sim\mathcal{N}(\mu_A,\boldsymbol{\Sigma}_{AA})$ and for any **conditional distr.**:
$\mathbf{X}_A|\mathbf{X}_B=\mathbf{x}_B\sim\mathcal{N}(\mu_{A|B},\boldsymbol{\Sigma}_{A|B})$ where
$\mu_{A|B}=\mu_A+\boldsymbol{\Sigma}_{AB}\boldsymbol{\Sigma}_{BB}^{-1}(\mathbf{x}_B-\mu_B)$
$\boldsymbol{\Sigma}_{A|B}=\boldsymbol{\Sigma}_{AA}-\boldsymbol{\Sigma}_{AB}\boldsymbol{\Sigma}_{BB}^{-1}\boldsymbol{\Sigma}_{BA}$
Observe that the variance can only shrink.

- Additive and closed under affine transformations.
  - $\mathbf{M}\cdot\mathcal{N}(\mu,\boldsymbol{\Sigma})=\mathcal{N}(\mathbf{M}\mu,\mathbf{M}^\top\boldsymbol{\Sigma}\mathbf{M})$
  - $\mathcal{N}(\mu_A,\boldsymbol{\Sigma}_A)+\mathcal{N}(\mu_B,\boldsymbol{\Sigma}_B)=\mathcal{N}(\mu_A+\mu_B,\boldsymbol{\Sigma}_A+\boldsymbol{\Sigma}_B)$
  - $\mathcal{N}(\mu_A,\boldsymbol{\Sigma}_A)\cdot\mathcal{N}(\mu_B,\boldsymbol{\Sigma}_B)\propto\mathcal{N}(\cdot,\cdot)$

Maximum likelihood estimate (MLE):
$\hat\theta_{\mathrm{MLE}}\doteq\underset{\theta\in\Theta}{\operatorname{argmax}}\,p(y_{1:n}|\mathbf{x}_{1:n},\theta)=\underset{\theta\in\Theta}{\operatorname{argmax}}\sum_{i=1}^n\log p(y_i|\mathbf{x}_i,\theta)$
- **Consistent** if: $\hat\theta_{\mathrm{MLE}}\xrightarrow{\mathbb{P}}\theta^\star$ as $n\to\infty$.
- **Asymptotically normal** if $\hat\theta_{\mathrm{MLE}}\xrightarrow{\mathcal{D}}\mathcal{N}(\theta^\star,\mathbf{S}_n)$ as $n\to\infty$ where $\mathbf{S}_n$ is the asymptotic covariance of MLE.
- MLE is **asymptotically efficient** (there exists no other consistent estimator with a "smaller" asymptotic var.).
- For the finite sample regime, the MLE need not be unbiased, and it is susceptible to overfitting to the (finite) training data.

Maximum a posterior (MAP) estimate:
$\hat\theta_{\mathrm{MAP}}\doteq\underset{\theta\in\Theta}{\operatorname{argmax}}\,p(\theta|\mathbf{x}_{1:n},y_{1:n})$
$=\underset{\theta\in\Theta}{\operatorname{argmin}}\underbrace{-\log p(\theta)}_{\text{regularization}}+\underbrace{\ell_{\mathrm{nll}}(\theta;\mathcal{D}_n)}_{\text{quality of fit}}$
The **log-prior** $\log p(\theta)$ acts as a regularizer. Common:
- $p(\theta)=\mathcal{N}(\theta;\mathbf{0},\lambda\mathbf{I})$ gives $-\log p(\theta)=\frac{\lambda}{2}\|\theta\|_2^2+\mathrm{const}$
- $p(\theta)=\mathrm{Laplace}(\theta;\mathbf{0},\lambda)$ gives $-\log p(\theta)=\lambda\|\theta\|_1+\mathrm{const}$
- Uniform prior gives const (no regularization, MAP is equivalent to the MLE)

2 Bayesian Linear Regression (BLR)

$\hat{\mathbf{w}}_{\mathrm{ls}}\doteq\underset{\mathbf{w}\in\mathbb{R}^d}{\operatorname{argmin}}\|\mathbf{y}-\mathbf{Xw}\|_2^2=\underset{\mathbf{w}\in\mathbb{R}^d}{\operatorname{argmin}}\sum_{i=1}^n(y_i-\mathbf{w}^\top\mathbf{x}_i)^2$
$\hat{\mathbf{w}}_{\mathrm{ridge}}\doteq\underset{\mathbf{w}\in\mathbb{R}^d}{\operatorname{argmin}}\|\mathbf{y}-\mathbf{Xw}\|_2^2+\lambda\|\mathbf{w}\|_2^2$
$\hat{\mathbf{w}}_{\mathrm{ls}}=(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}$; $\hat{\mathbf{w}}_{\mathrm{ridge}}=(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}$

Gaussian prior on weights $\mathbf{w}\sim\mathcal{N}(\mathbf{0},\sigma_p^2\mathbf{I})$:
- Yields Gaussian posterior $\mathbf{w}|\mathbf{x}_{1:n},y_{1:n}\sim\mathcal{N}(\mu,\boldsymbol{\Sigma})$, as $\log p(\mathbf{w}|\mathbf{x}_{1:n},y_{1:n})=-\frac{1}{2}[\mathbf{w}^\top\boldsymbol{\Sigma}^{-1}\mathbf{w}-2\mu]+\mathrm{const}$, with $\boldsymbol{\Sigma}\doteq(\sigma_n^{-2}\mathbf{X}^\top\mathbf{X}+\sigma_p^{-2}\mathbf{I})^{-1}$ and $\mu\doteq\sigma_n^{-2}\boldsymbol{\Sigma}\mathbf{X}^\top\mathbf{y}$.
- MAP is *identical to ridge regression* with $\lambda=\sigma_n^2/\sigma_p^2$.
- **Bayesian inference**: Distr. for a test point $\mathbf{x}^\star$ is: $y^\star|\mathbf{x}^\star,\mathbf{x}_{1:n},y_{1:n}\sim\mathcal{N}(\mu^\top\mathbf{x}^\star,\mathbf{x}^{\star\top}\boldsymbol{\Sigma}\mathbf{x}^\star+\sigma_n^2)$

Laplace prior on weights $\mathbf{w}\sim\mathrm{Laplace}(\mathbf{0},h)$:
- MAP is *identical to lasso regression* with $\lambda=\sigma_n^2/h$.

Heteroscedastic noise ϵ_i may depend on $\mathbf{x}_i$, while **Homoscedastic** may not.
$\mathrm{Var}[y^\star|\mathbf{x}^\star]=\underbrace{\mathbb{E}_\theta[\mathrm{Var}_{y^\star}[y^\star|\mathbf{x}^\star,\theta]]}_{\text{aleatoric uncertainty}}+\underbrace{\mathrm{Var}_\theta[\mathbb{E}_{y^\star}[y^\star|\mathbf{x}^\star,\theta]]}_{\text{epistemic uncertainty}}$
Aleatoric: noise in data; **Epistemic**: noise in model.

Applying linear regression to non-linear functions: use non-linear transformation ϕ to $\mathbf{X}$. Define $\boldsymbol{\Phi}=\phi(\mathbf{X})$. With Gaussian prior and $\mathbf{K}=\sigma_p^2\boldsymbol{\Phi}\boldsymbol{\Phi}^\top$ we get: $\mathbf{f}|\mathbf{X}\sim\mathcal{N}(\boldsymbol{\Phi}\mathbb{E}[\mathbf{w}],\boldsymbol{\Phi}\mathrm{Var}[\mathbf{w}]\boldsymbol{\Phi}^\top)=\mathcal{N}(\mathbf{0},\mathbf{K})$.
Kernel: $k(\mathbf{x},\mathbf{x}')\doteq\sigma_p^2\phi(\mathbf{x})^\top\phi(\mathbf{x}')=\mathrm{Cov}[f(\mathbf{x}),f(\mathbf{x}')]$
- Choice of kernel implicitly determines the function class that $\mathbf{f}$ is sampled from, which encodes our prior beliefs.
- Kernel matrix has shape $n\times n$ (input space dimension) instead of $e\times e$ (feature space dimension).

For **inference**, define $\tilde{\boldsymbol{\Phi}}\doteq\begin{bmatrix}\boldsymbol{\Phi}\\ \phi(\mathbf{x}^{\star\top})\end{bmatrix}$, $\tilde{\mathbf{y}}\doteq\begin{bmatrix}\mathbf{y}\\ y^\star\end{bmatrix}$, $\tilde{\mathbf{f}}\doteq\begin{bmatrix}\mathbf{f}\\ f^\star\end{bmatrix}$.
For $\tilde{\mathbf{f}}=\tilde{\boldsymbol{\Phi}}w$ we have: $\tilde{\mathbf{y}}|\mathbf{X},\mathbf{x}^\star\sim\mathcal{N}(\mathbf{0},\tilde{\mathbf{K}}+\sigma_n^2\mathbf{I})$

- **Linear kernel**: $k(\mathbf{x},\mathbf{x}')=\lambda\mathbf{x}^\top\mathbf{x}'$
- **RBF/Gaussian kernel**: $k(\mathbf{x},\mathbf{x}')=\exp(-\frac{(\mathbf{x}-\mathbf{x}')^2}{2l^2})$ (larger length scale l results in smoother fns.; cannot model under the weight-space view of BLR; feature space are polynomials of infinite degree)
- **Polynomial kernel** $k(\mathbf{x},\mathbf{x}')=(1+\mathbf{x}^\top\mathbf{x}')^d$ (feature space are polynomials of degree d)
- **Laplacian kernel**: $k(\mathbf{x},\mathbf{x}')=\exp(-\alpha\|\mathbf{x}-\mathbf{x}'\|)$ (non-smooth)
- **Matérn kernel**: (For $v=\frac{1}{2}$ equiv. to Laplace kernel, for $v\to\infty$ equal to RBF kernel)

Properties of kernels
- Symmetric: $k(\mathbf{x},\mathbf{x}')=k(\mathbf{x}',\mathbf{x})$
- $\mathbf{K}_{AA}$ is p.s.d.
- **Stationary** if there exists $\tilde k$ s.t. $\tilde k(\mathbf{x}-\mathbf{x}')=k(\mathbf{x},\mathbf{x}')$ (only relative location of points matters)
- **Isotropic** if there exists $\tilde k$ s.t. $\tilde k(\|\mathbf{x}-\mathbf{x}'\|_2)=k(\mathbf{x},\mathbf{x}')$ (only distance between points matters)

Composition of kernels
- Addition: $k(\mathbf{x},\mathbf{x}')\doteq k_1(\mathbf{x},\mathbf{x}')+k_2(\mathbf{x},\mathbf{x}')$ (OR)
- Multiplication: $k(\mathbf{x},\mathbf{x}')\doteq k_1(\mathbf{x},\mathbf{x}')\cdot k_2(\mathbf{x},\mathbf{x}')$ (AND)
- Mult. with const.: $k(\mathbf{x},\mathbf{x}')\doteq c\cdot k_1(\mathbf{x},\mathbf{x}')$ for any $c\geq0$
- Composition. with poly.: $k(\mathbf{x},\mathbf{x}')\doteq f(k_1(\mathbf{x},\mathbf{x}'))$ for any poly. f with positive coefficients.
- Composition. with exp.: $k(\mathbf{x},\mathbf{x}')\doteq\exp(k_1(\mathbf{x},\mathbf{x}'))$

Efficient online BLR ($O(d^2)$ instead of $O(d^3)$):
- $\mathbf{X}^{(t+1)\top}\mathbf{X}^{(t+1)}=\mathbf{X}^{(t)\top}\mathbf{X}^{(t)}+\mathbf{x}^{(t)\top}\mathbf{x}^{(t)}\in\mathbb{R}^{d\times d}$
- $\mathbf{X}^{(t+1)\top}\mathbf{y}^{(t+1)}=\mathbf{X}^{(t)\top}\mathbf{y}^{(t)}+\mathbf{x}^{(t)\top}\mathbf{y}^{(t)}\in\mathbb{R}^d$
- Since $\mathbf{X}^\top\mathbf{X}=\sum_{i=1}^t\mathbf{x}_i\mathbf{x}_i^\top$ and $\mathbf{X}^\top\mathbf{y}=\sum_{i=1}^t y_i x_i$

Logistic BLR:
$\hat{\mathbf{w}}_{\mathrm{MAP}}=\underset{\mathbf{w}\in\mathbb{R}^d}{\operatorname{argmin}}\frac{1}{2\sigma_p^2}\|\mathbf{w}\|_2^2+\sum_{i=1}^n\log(1+\exp(-y_i\mathbf{w}^\top\mathbf{x}_i))$
- For $\lambda=1/(2\sigma_p^2)$ this is equiv. to standard logistic regression where $\ell_{\log}(\mathbf{w}^\top\mathbf{x};y)\doteq\log(1+\exp(-y\mathbf{w}^\top\mathbf{x}))$ and $\nabla_\mathbf{w}\ell_{\log}(\mathbf{w}^\top\mathbf{x};y)=-y\mathbf{x}\cdot\sigma(-y\mathbf{w}^\top\mathbf{x})$.
- Posterior is not Gaussian or closed-form, but its log. density is convex

3 Gaussian Processes (GPs)

An infinite set of random variables s.t. any finite number of them are jointly Gaussian. We use set $\mathcal{X}$ to index the collection of random variables. It is def. by **mean fn.** $\mu:\mathcal{X}\to\mathbb{R}$ and **covar./kernel fn.** $k:\mathcal{X}\times\mathcal{X}\to\mathbb{R}$ s.t. for any $A\doteq\{\mathbf{x}_1,\dots,\mathbf{x}_m\}\subseteq\mathcal{X}$, we have $\mathbf{f}_A=[f_{\mathbf{x}_1}\cdots f_{\mathbf{x}_m}]^\top\sim\mathcal{N}(\mu_A,\mathbf{K}_{AA})$. We write $f\sim\mathcal{GP}(\mu,k)$. Using homoscedastic noise assumption: $y^\star|\mathbf{x}^\star,\mu,k\sim\mathcal{N}(\mu(\mathbf{x}^\star),k(\mathbf{x}^\star,\mathbf{x}^\star)+\sigma_n^2)$

New point: Joint distribution of the observations $y_{1:n}$ and the noise-free prediction $f^\star$ at a test point $\mathbf{x}^\star$ as $\begin{bmatrix}\mathbf{y}\\ f^\star\end{bmatrix}|\mathbf{x}^\star,\mathbf{x}_{1:n}\sim\mathcal{N}(\tilde\mu,\tilde{\mathbf{K}})$ where $\tilde\mu=\begin{bmatrix}\mu_A\\ \mu(\mathbf{x}^\star)\end{bmatrix}$,
$\tilde{\mathbf{K}}=\begin{bmatrix}\mathbf{K}_{AA}+\sigma_n^2\mathbf{I} & \mathbf{k}_{\mathbf{x}^\star,A}\\ \mathbf{k}_{\mathbf{x}^\star,A}^\top & k(\mathbf{x}^\star,\mathbf{x}^\star)\end{bmatrix}$, $\mathbf{k}_{\mathbf{x},A}\doteq[k(\mathbf{x},\mathbf{x}_1)\dots k(\mathbf{x},\mathbf{x}_n)]^\top$

GP posterior update: $f|\mathbf{x}_{1:n},y_{1:n}\sim\mathcal{GP}(\mu',k')$ where $\mu'(\mathbf{x})=\mu(\mathbf{x})+\mathbf{k}_{\mathbf{x},A}^\top(\mathbf{K}_{AA}+\sigma_n^2\mathbf{I})^{-1}(\mathbf{y}_A-\mu_A)$ and $k'(\mathbf{x},\mathbf{x}')\doteq k(\mathbf{x},\mathbf{x}')-\mathbf{k}_{\mathbf{x},A}^\top(\mathbf{K}_{AA}+\sigma_n^2\mathbf{I})^{-1}\mathbf{k}_{\mathbf{x}',A}$
- The posterior covariance can only decrease when conditioning on more data, and is independent of $\mathbf{y}_i$.
- GP posterior takes $\mathcal{O}(n^3)$ because of mat. inversion.

Maximizing marginal likelihood: optimizes the θ across all realizations of $\mathbf{f}$ (somewhat regularization, avoids overfitting without train and validation split):
$\hat\theta_{\mathrm{MLE}}\doteq\underset{\theta}{\operatorname{argmax}}\,p(y_{1:n}|\mathbf{x}_{1:n},\theta)$
$=\underset{\theta}{\operatorname{argmax}}\int p(y_{1:n}|\mathbf{x}_{1:n},f,\theta)p(f|\theta)df$
Intuition:
- *Underfit* models: likelihood is mostly small as data cannot be well described; prior is large as there are "fewer" fns. to choose from.
- *Overfit* models: likelihood is large for "some" fns. but small for "most" fns.; prior is small, as probability mass has to be distributed among "more" fns.

Maximizing encourages trading between a large likelihood and large prior, as one product term will be small.

For GP regression: $y_{1:n}|\mathbf{x}_{1:n},\theta\sim\mathcal{N}(\mathbf{0},\mathbf{K}_{f,\theta}+\sigma_n^2\mathbf{I})$, write $\mathbf{K}_{\mathbf{y},\theta}\doteq\mathbf{K}_{f,\theta}+\sigma_n^2\mathbf{I}$, and obtain:
$\hat\theta_{\mathrm{MLE}}=\operatorname{argmin}_\theta\underbrace{1/2\cdot\mathbf{y}^\top\mathbf{K}_{\mathbf{y},\theta}^{-1}\mathbf{y}}_{\text{Goodness of fit}}+\underbrace{1/2\cdot\log\det(\mathbf{K}_{\mathbf{y},\theta})}_{\text{"Volume" of model class}}$
The loss can be expressed in closed-form with $\alpha=\mathbf{K}_{\mathbf{y},\theta}^{-1}\mathbf{y}$:
$\frac{\partial}{\partial\theta_j}\log p(y_{1:n}|\mathbf{x}_{1:n},\theta)=\frac{1}{2}\mathrm{tr}((\alpha\alpha^\top-\mathbf{K}_{\mathbf{y},\theta}^{-1})\frac{\partial\mathbf{K}_{\mathbf{y},\theta}}{\partial\theta_j})$
This optimization problem is, in general, non-convex.

Reproducing kernel Hilbert space (RKHS):
Given kernel $k:\mathcal{X}\times\mathcal{X}\to\mathbb{R}$, its corresponding RKHS is the space of functions f defined as:
$\mathcal{H}_k(\mathcal{X})=\{f(\cdot)=\sum_{i=1}^n\alpha_ik(\mathbf{x}_i,\cdot)|n\in\mathbb{N},\mathbf{x}_i\in\mathcal{X},\alpha_i\in\mathbb{R}\}$
The inner product of the RKHS is defined as: $\langle f,g\rangle_k=\sum_{i=1}^n\sum_{j=1}^n\alpha_i\alpha'_jk(\mathbf{x}_i,\mathbf{x}'_j)$ where $g(\cdot)=\sum_{j=1}^n\alpha'_jk(\mathbf{x}'_j,\cdot)$.
The RKHS induces norm $\|\mathbf{f}\|_k=\sqrt{\langle\mathbf{f},\mathbf{f}\rangle_k}$ measuring the "smoothness" or "complexity" of $\mathbf{f}$.
- For all $\mathbf{x}\in\mathcal{X}$ and $\mathbf{f}\in\mathcal{H}_k(\mathcal{X})$: $\mathbf{f}(\mathbf{x})=\langle\mathbf{f}(\cdot),k(\mathbf{x},\cdot)\rangle_k$
- **Representer theorem:** Kernel k, $\lambda>0$, $f\in\mathcal{H}_k(\mathcal{X})$, and train data $\{(\mathbf{x}_i,f(\mathbf{x}_i))\}_{i=1}^n$. Let loss fn. $\mathcal{L}(f(\mathbf{x}_1),\dots,f(\mathbf{x}_n))\in\mathbb{R}\cup\{\infty\}$ depend on f only through its eval. at train points. Then, any minimizer $\hat f\in\underset{f\in\mathcal{H}_k(\mathcal{X})}{\operatorname{argmin}}\mathcal{L}(f(\mathbf{x}_1),\dots,f(\mathbf{x}_n))+\lambda\|f\|_k^2$ admits a repr. of form $\hat f(\mathbf{x})=\hat{\mathbf{a}}\mathbf{k}_{\mathbf{x},\{\mathbf{x}_i\}_{i=1}^n}=\sum_{i=1}^n\hat\alpha_ik(\mathbf{x},\mathbf{x}_i)$
- MAP estimate of a GP corresponds to the solution of a regularized linear regression problem in the RKHS of the kernel fn.: $\hat f=\underset{f\in\mathcal{H}_k(\mathcal{X})}{\operatorname{argmin}}-\log(y_{1:n}|\mathbf{x}_{1:n},f)+\frac{1}{2}\|f\|_k^2$
- **GPs remain comp. tractable** even though they can model fns. over "infinite-dim." feat. spaces.

Approximations: GP need to invert mat ($\mathcal{O}(n^3)$).
- **Local method**: When sampling at $\mathbf{x}$ only condition on samples $\mathbf{x}'$, that are close: $|k(\mathbf{x},\mathbf{x}')|\geq\tau$ for some $\tau>0$.
Problem: τ has to be chosen carefully: if τ is chosen too large, samples become essentially independent. Still expensive if "many" points are close.
- **Kernel approximation**: Construct a low dim. feat. map $\phi:\mathbb{R}^d\to\mathbb{R}^m$ s.t.: $k(\mathbf{x},\mathbf{x}')\approx\phi(\mathbf{x})^\top\phi(\mathbf{x}')$.
  - Transforms function-space view (GP) back into a tractable weight-space view (BLR, $\mathcal{O}(nm^2+m^3)$).
  - Can be done with **Random Fourier features**: a *stationary* kernel k can be interpreted as fn. in one variable, and has an associated Fourier transform.

Bochner's Theorem A continuous Kernel on $\mathbb{R}^d$ is p.s.d iff its Fourier transform $p(\omega)$ is non-negative.

Randomized feature map: $z_{\omega,b}(\mathbf{x})\doteq\sqrt{2}\cos(\omega^\top\mathbf{x}+b)$, with $\omega^{(i)}\overset{\text{iid}}{\sim}p$ and $b^{(i)}\overset{\text{iid}}{\sim}\mathrm{Unif}([0,2\pi])$. Inner product $z(x)^\top z(x')$ is unbiased estimator of $k(x-x')$. Error prob. decays exp. in dim. of Fourier feature space m.
- **Inducing point methods**: Idea is to summarize data around inducing pts. $U\doteq\{\bar{\mathbf{x}}_1,\dots,\bar{\mathbf{x}}_n\}$. Let $\mathbf{f}\doteq[f(\mathbf{x}_1)\dots,f(\mathbf{x}_n)]^\top$, $f^\star\doteq f(\mathbf{x}^\star)$, $\mathbf{u}\doteq[f(\hat{\mathbf{x}}_1)\dots f(\hat{\mathbf{x}}_n)]^\top$.
Original GP recoverable with marginalization: $p(f^\star|\mathbf{f})=\int_{\mathbb{R}^k}p(f^\star,\mathbf{f}|\mathbf{u})p(\mathbf{u})d\mathbf{u}$. Approx. the joint prior, assuming $f^\star$, $\mathbf{f}$ are cond. indep. given $\mathbf{u}\sim\mathcal{N}(\mathbf{0},\mathbf{K}_{UU})$
Train conditional: $p(\mathbf{f}|\mathbf{u})\sim\mathcal{N}(\mathbf{f};\mathbf{K}_{AU}\mathbf{K}_{UU}^{-1}\mathbf{u},\mathbf{K}_{AA}-\mathbf{Q}_{AA})$
Test conditional: $p(f^\star|\mathbf{u})\sim\mathcal{N}(f^\star;\mathbf{K}_{\star U}\mathbf{K}_{UU}^{-1}\mathbf{u},\mathbf{K}_{\star\star}-\mathbf{Q}_{\star\star})$
w. $\mathbf{Q}_{ab}\doteq\mathbf{K}_{aU}\mathbf{K}_{UU}^{-1}\mathbf{K}_{Ub}$. $\mathbf{K}_{AA}$ represents the prior covar. and $\mathbf{Q}_{AA}$ represents covar. from inducing pts. Covar. mat. comp. is expensive; need to approx.:
  - **Subset of regressors (SoR)**: Forgets about all var. and covar. $q_{\mathrm{SoR}}(\mathbf{f}|\mathbf{u})\doteq\mathcal{N}(\mathbf{f};\mathbf{K}_{AU}\mathbf{K}_{UU}^{-1}\mathbf{u},\mathbf{0})$
  $q_{\mathrm{SoR}}(f^\star|\mathbf{u})\doteq\mathcal{N}(f^\star;\mathbf{K}_{\star U}\mathbf{K}_{UU}^{-1}\mathbf{u},\mathbf{0})$
  - **Fully independent training conditional (FITC)**: Keeps track of variances but forgets about covariance
  $q_{\mathrm{FITC}}(\mathbf{f}|\mathbf{u})\doteq\mathcal{N}(\mathbf{f};\mathbf{K}_{AU}\mathbf{K}_{UU}^{-1}\mathbf{u},\mathrm{diag}\{\mathbf{K}_{AA}-\mathbf{Q}_{AA}\})$
  $q_{\mathrm{FITC}}(f^\star|\mathbf{u})\doteq\mathcal{N}(f^\star;\mathbf{K}_{\star U}\mathbf{K}_{UU}^{-1}\mathbf{u},\mathrm{diag}\{\mathbf{K}_{\star\star}-\mathbf{Q}_{\star\star}\})$
Comp. cost SoR/FITC is dom. by mat. inv. of $\mathbf{K}_{UU}$, so cubic in num. inducing pts. and linear in data pts.

4 Variational Inference

Idea: approximate true posterior distribution with a simpler posterior that is easy to sample:
$p(\theta|\mathbf{x}_{1:n},y_{1:n})=\frac{1}{Z}p(\theta,y_{1:n}|\mathbf{x}_{1:n})\approx q(\theta|\lambda)=q_\lambda(\theta)$, where λ are params. of the **variational posterior** q_λ.
Laplace approx.: find a Gaussian approx. (i.e. second-order Taylor) of the posterior around its mode: $q(\theta)=\mathcal{N}(\theta;\hat\theta,\boldsymbol{\Lambda}^{-1})\propto\exp(\hat\psi(\theta))$, with $\hat\theta$ the mode (i.e. MAP estimate) and with $\mathbf{H}$ the Hessian: $\boldsymbol{\Lambda}\doteq-\mathbf{H}_\psi(\hat\theta)=-\mathbf{H}_\theta\log p(\theta|\mathbf{x}_{1:n},y_{1:n})|_{\theta=\hat\theta}$.
Perform inference using the variations approximation: $p(y^\star|\mathbf{x}^\star,\mathbf{x}_{1:n},y_{1:n})\approx\int p(y^\star|\mathbf{x}^\star,\theta)q_\lambda(\theta)d\theta$ $=\mathbb{E}_{\theta\sim q_\lambda}[p(y^\star|\mathbf{x}^\star,\theta)]$
- Matches shape of true posterior around its mode but may not represent it accurately elsewhere.
- Leads to extremely overconfident predictions, often unsuitable for approximate probabilistic inference.
- Preserves MAP point estimate as its mean and just "adds" a little uncertainty around it.

Surprise of an event with probability u: $\mathrm{S}[u]\doteq-\log u$. $\mathrm{S}[u]$ is convex in u. For a discrete RV X: $\mathrm{S}[p(x)]\geq0$. Axiomatic characterization up to pos. const. factor:
- $\mathrm{S}[u]>\mathrm{S}[v]\implies u<v$ (anti-monotonicity)
- S continuous
- $\mathrm{S}[uv]=\mathrm{S}[u]+\mathrm{S}[v]$ for independent events

Entropy of distr. p is avg. surprise of samples from p: $\mathrm{H}[p]\doteq\mathbb{E}_{x\sim p}[\mathrm{S}[p(x)]]=\mathbb{E}_{x\sim p}[-\log p(x)]$. Can be negative but if p is discrete then $\mathrm{H}[p]\geq0$.

Jensen's inequality: For convex fn. g we have: $g(\mathbb{E}[X])\leq\mathbb{E}[g(X)]$; if h concave: $h(\mathbb{E}[X])\geq\mathbb{E}[h(X)]$

The **cross-entropy** of q relative to p is: $\mathrm{H}[p\|q]\doteq\mathbb{E}_{x\sim p}[\mathrm{S}[q(x)]]=\mathbb{E}_{x\sim p}[-\log q(x)]$.

KL-div.: measures additional expected surprise when observing samples from p that is due to assuming (wrong) q and which is not inherent in p already. $\mathrm{KL}(p\|q)=\mathrm{H}[p\|q]-\mathrm{H}[p]=\mathbb{E}_{\theta\sim p}[\log\frac{p(\theta)}{q(\theta)}]$
- $\mathrm{KL}(p\|q)\geq0$; $\mathrm{KL}(p\|q)=0$ iff $p=q$ almost surely
- There exist distr. p and q s.t. $\mathrm{KL}(p\|q)\neq\mathrm{KL}(q\|p)$

Note that: $\mathrm{H}[p\|q]=\mathrm{H}[p]+\mathrm{KL}(p\|q)\geq\mathrm{H}[p]$.
- $\mathrm{KL}(\mathrm{Bern}(p)\|\mathrm{Bern}(q))=p\log\frac{p}{q}+(1-p)\log\frac{(1-p)}{(1-q)}$
- For $p=\mathcal{N}(\mu_p,\boldsymbol{\Sigma}_p)$ and $q=\mathcal{N}(\mu_q,\boldsymbol{\Sigma}_q)$: $\mathrm{KL}(p\|q)=$ $\frac{1}{2}\left(\mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_p)+(\mu_p-\mu_q)^\top\boldsymbol{\Sigma}_q^{-1}(\mu_p-\mu_q)-d+\log\frac{\det(\boldsymbol{\Sigma}_q)}{\det(\boldsymbol{\Sigma}_p)}\right)$
- **Forward KL**: $q_1^\star\doteq\operatorname{argmin}_{q\in\mathcal{Q}}\mathrm{KL}(p\|q)$(mode avg., more conservative, yields more "desired" approx.)
- **Reverse KL**: $q_2^\star\doteq\operatorname{argmin}_{q\in\mathcal{Q}}\mathrm{KL}(q\|p)$ (greedily mode seeking, underestimate var., overconfident preds.)

Evidence lower bound (ELBO), for given data $\mathcal{D}_n$:
$L(q,p;\mathcal{D}_n)=\underbrace{\log p(y_{1:n}|\mathbf{x}_{1:n})}_{\text{const}}-\mathrm{KL}(q\|p(\cdot|\mathbf{x}_{1:n},y_{1:n}))$
$=\underbrace{\mathbb{E}_{\theta\sim q}[\log p(y_{1:n}|\mathbf{x}_{1:n},\theta)]}_{\text{log-likelihood}}-\underbrace{\mathrm{KL}(q\|p(\cdot))}_{\text{proximity to prior}}$
Max the ELBO coincides with min. reverse-KL. Since KL-div. is non-negative: $\log p(y_{1:n}|\mathbf{x}_{1:n})\geq L(q,p;\mathcal{D}_n)$
- Max. ELBO selects a var. distr. q that is close to prior $p(\cdot)$ while also max. avg. data likelihood $p(y_{1:n}|\mathbf{x}_{1:n},\theta)$ for $\theta\sim q$. Contrast to MAP, which picks single mode θ that max. the likelihood and proximity to the prior.
- ELBO gradient is gen. **intractable** (use reparam. trick).

Reparm. trick: Let $\epsilon\sim\phi$ be indep. of λ, $\mathbf{g}:\mathbb{R}^d\to\mathbb{R}^d$ be a diff. and inv. fn, $\theta=\mathbf{g}(\epsilon;\lambda)$, and $\mathbf{f}$ a *nice* fn. We get: $q_\lambda(\theta)=\phi(\epsilon)\cdot|\det(\mathbf{D}_\epsilon\mathbf{g}(\epsilon;\lambda))|^{-1}$; $\mathbb{E}_{\theta\sim q_\lambda}[\mathbf{f}(\theta)]=\mathbb{E}_{\epsilon\sim\phi}[\mathbf{f}(\mathbf{g}(\epsilon;\lambda))]$
- For ELBO $\boldsymbol{\nabla}_\lambda\mathbb{E}_{\theta\sim q_\lambda}[\mathbf{f}(\theta)]=\mathbb{E}_{\epsilon\sim\phi}[\boldsymbol{\nabla}_\lambda\mathbf{f}(\mathbf{g}(\epsilon;\lambda))]$. If we can find $\mathbf{g}$ and suitable reference density ϕ which is indep. of λ, we say q_λ is **reparametrizable**.
- For Gaussian: $q_\lambda(\theta)=\mathcal{N}(\theta;\mu,\boldsymbol{\Sigma})$; $\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})$, set: $\theta=\mathbf{g}(\epsilon;\lambda)\doteq\boldsymbol{\Sigma}^{1/2}\epsilon+\mu$, then: $\phi(\epsilon)=q_\lambda(\theta)\cdot|\det(\boldsymbol{\Sigma}^{1/2})|$ and $\epsilon=\mathbf{g}^{-1}(\theta;\lambda)=\boldsymbol{\Sigma}^{-1/2}(\theta-\mu)$

5 Bayesian Deep Learning

Universal approx. theorem: Any ANN with a single hidden layer (arbitrary width) and non-poly. activation fn. can approx. any cont. fn. to an arbitrary accuracy.
- **Softmax**: $\sigma_i(\mathbf{f})\doteq\frac{\exp(f_i)}{\sum_{j=1}^c\exp(f_j)}$ (classification)
$\nabla_z\sigma_i(\mathbf{f})=\sigma_i(\mathbf{f})(1-\sigma_i(\mathbf{f}))$
- **Hyperbolic tangent**: $\mathrm{Tanh}(z)\doteq\frac{\exp(z)-\exp(-z)}{\exp(z)+\exp(-z)}$
$\nabla_z\mathrm{Tanh}(z)=1-\mathrm{Tanh}^2(z)$; $\mathrm{Tanh}(z)=2\sigma(2z)-1$;
- **Rectified linear unit**: $\mathrm{ReLU}(z)\doteq\max\{z,0\}\in[0,\infty)$
$\nabla_z\mathrm{ReLU}(z)=\mathbf{1}_{\{z\geq0\}}$

For linear regression, minimizing MSE/CE **corresponds to MLE** under a Gaussian likelihood.
- **MSE loss**: $\ell_{\mathrm{mse}}(\theta;\mathcal{D})\doteq\frac{1}{n}\sum_{i=1}^n(f(\mathbf{x}_i;\theta)-y_i)^2$
- **CE loss**: $\ell_{\mathrm{ce}}(\theta;\mathcal{D})\doteq-\frac{1}{n}\sum_{i=1}^n\log q_\theta(\mathbf{y}_i|\mathbf{x}_i)$

BNNs: Gaussian prior on weights $\theta\sim\mathcal{N}(\mathbf{0},\sigma_p^2\mathbf{I})$, and Gaussian likelihood to describe how well data is described by the model: $y\,|\,\mathbf{x},\theta\sim\mathcal{N}(f(\mathbf{x};\theta),\sigma_n^2)$.
The **MAP estimate** is:
$\hat\theta_{\text{MAP}}=\arg\min_\theta \frac{1}{2\sigma_p^2}\|\theta\|_2^2+\frac{1}{2\sigma_n^2}\sum_{i=1}^n(y_i-f(\mathbf{x}_i;\theta))^2$.
Update rule: $\theta\leftarrow\theta(1-\frac{\eta_t}{\sigma_p^2})+\eta_t\sum_{i=1}^n\boldsymbol{\nabla}\log p(y_i\,|\,\mathbf{x}_i,\theta)$.

Also modeling heteroscedastic noise: Use a neural network with 2 outputs f_1,f_2, and define:
$y\,|\,\mathbf{x},\theta\sim\mathcal{N}(\mu(\mathbf{x};\theta),\sigma^2(\mathbf{x};\theta))$ where $\mu(\mathbf{x};\theta)\doteq f_1(\mathbf{x};\theta)$ and $\sigma^2(\mathbf{x};\theta)\doteq\exp(f_2(\mathbf{x};\theta))$. Likelihood term:
$\log p(y_i\,|\,\mathbf{x}_i,\theta)=\text{const}-\frac{1}{2}[\log\sigma^2(\mathbf{x}_i;\theta)+\frac{(y_i-\mu(\mathbf{x}_i;\theta))^2}{\sigma^2(\mathbf{x}_i;\theta)}]$.

- BNN learning and inference are **generally intractable** when the noise is not assumed to be homoscedastic and known. Thus, we need approx. inference.
- Goal: approx. true posterior $p(\theta\,|\,\mathcal{D})$ with simpler variational distr. q_λ typically family of indep. Gaussians.
- Achieved by max. ELBO with SGD and reparm. trick.
- We can approx. the predictive distr. by sampling from the variational posterior $p(y^\star\,|\,\mathbf{x}^\star,\mathbf{x}_{1:n},\mathbf{y}_{1:n})\approx\mathbb{E}_{\theta\sim q_\lambda}[p(y^\star\,|\,\mathbf{x}^\star,\theta)]\approx\frac{1}{m}\sum_{i=1}^m p(y^\star\,|\,\mathbf{x}^\star,\theta^{(i)})$.
- VI in BNNs can be seen as avg. preds. of multiple NNs drawn acc. to the variational posterior q_λ.
- Using Monte Carlo samples estimate mean and var.:

$\mathbb{E}[y^\star\,|\,\mathbf{x}^\star,\mathbf{x}_{1:n},\mathbf{y}_{1:n}]\approx\frac{1}{m}\sum_{i=1}^m\mu(\mathbf{x}^\star;\theta^{(i)})$
$\text{Var}[y^\star\,|\,\mathbf{x}^\star,\mathbf{x}_{1:n},\mathbf{y}_{1:n}]\approx\mathbb{E}_\theta[\text{Var}_{y^\star}[y^\star\,|\,\mathbf{x}^\star,\theta]]+\text{Var}_\theta[\mathbb{E}_{y^\star}[y^\star\,|\,\mathbf{x}^\star,\theta]]$
$\approx\mathbb{E}_\theta[\sigma^2(\mathbf{x}^\star;\theta)]+\text{Var}_\theta[\mu(\mathbf{x}^\star;\theta)]$

$$\approx\underbrace{\frac{1}{m}\sum_{i=1}^m\sigma^2(\mathbf{x}^\star;\theta^{(i)})}_{\text{aleatoric}}+\underbrace{\frac{1}{m-1}\sum_{i=1}^m(\mu(\mathbf{x}^\star;\theta^{(i)})-\bar\mu(\mathbf{x}^\star))^2}_{\text{epistemic}}$$

Alternative inference techniques:
- **Dropout/Dropconnect** randomly select/omits vertices/edges of the comp. graph. For valid interpretation of this as variational inference, we also need to perform dropout/dropconnect during inference.
- Dropout masks will overlap, making predictions highly correlated, leading to underestimation of epistemic uc. **Masksembles** mitigate by choosing fixed set of pre-defined dropout masks (controlled overlap).
- **Probabilistic ensembles:** learn m different NN over random chosen subsets of train data for each network.

Evidence of val. set; How well model desc. val. set?:
$\log p(y_{1:m}^{\text{val}}\,|\,\mathbf{x}_{1:m}^{\text{val}},\mathbf{x}_{1:n}^{\text{train}},y_{1:n}^{\text{train}})\gtrsim\frac{1}{k}\sum_{j=1}^k\sum_{i=1}^m\log p(y_i^{\text{val}}\,|\,\mathbf{x}_i^{\text{val}},\theta^{(j)})$

- **Frequency:** Proportion of samples in bin m that belong to 1: $\text{freq}(B_m)\doteq\frac{1}{|B_m|}\sum_{i\in B_m}\mathbb{1}\{\mathcal{Y}_i=1\}$
- **Confidence:** Avg. conf. of samples in bin m belonging to 1: $\text{conf}(B_m)\doteq\frac{1}{|B_m|}\sum_{i\in B_m}\mathbb{P}\{\mathcal{Y}_i=1\,|\,\mathbf{x}_i\}$

A model is **well-calibrated** if its confidence coincides with its acc. across many preds.: $\text{freq}(B_m)\approx\text{conf}(B_m)$

- **ECE:** $\ell_{\text{ECE}}\doteq\sum_{m=1}^M\frac{|B_m|}{n}|\text{freq}(B_m)-\text{conf}(B_m)|$
- **MCE:** $\ell_{\text{ECE}}\doteq\max_{m\in[M]}\frac{|B_m|}{n}|\text{freq}(B_m)-\text{conf}(B_m)|$

6 Active Learning

Cond. entropy: $\text{H}[\mathbf{X}\,|\,\mathbf{Y}]\doteq\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}[\text{H}[\mathbf{X}\,|\,\mathbf{Y}=\mathbf{y}]]$
$=\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim p(\mathbf{x},\mathbf{y})}[-\log p(\mathbf{x}\,|\,\mathbf{y})]$

Joint entropy: $\text{H}[\mathbf{X},\mathbf{Y}]\doteq\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim p(\mathbf{x},\mathbf{y})}[-\log p(\mathbf{x},\mathbf{y})]$
- $\text{H}[\mathbf{X}\,|\,\mathbf{Y}]\neq\text{H}[\mathbf{Y}\,|\,\mathbf{X}]$ in general; but $\text{H}[\mathbf{X},\mathbf{Y}]=\text{H}[\mathbf{Y},\mathbf{X}]$
- $\text{H}[\mathbf{X},\mathbf{Y}]=\text{H}[\mathbf{Y}]+\text{H}[\mathbf{X}\,|\,\mathbf{Y}]=\text{H}[\mathbf{X}]+\text{H}[\mathbf{Y}\,|\,\mathbf{X}]$
- $\text{H}[\mathbf{X}\,|\,\mathbf{Y}]=\text{H}[\mathbf{Y}\,|\,\mathbf{X}]+\text{H}[\mathbf{X}]-\text{H}[\mathbf{Y}]$ (Bayes Rule)
- $\text{H}[\mathbf{X}\,|\,\mathbf{Y}]\le\text{H}[\mathbf{X}]$ (Gibbs; Information never hurts) $\Leftrightarrow 0\le\text{H}[\mathbf{X}]-\text{H}[\mathbf{X}\,|\,\mathbf{Y}]=\text{I}(\mathbf{X};\mathbf{Y})$

Mutual info: $\text{I}(\mathbf{X};\mathbf{Y})=\text{H}[\mathbf{X}]+\text{H}[\mathbf{Y}]-\text{H}[\mathbf{X},\mathbf{Y}]$
- $\text{I}(\mathbf{X};\mathbf{Y})=\text{I}(\mathbf{Y};\mathbf{X})=\mathbb{E}_{\mathbf{y}\sim p}[\text{KL}(p(\mathbf{x}\,|\,\mathbf{y})\|p(\mathbf{x}))]$

Cond. mutual info:
- $\text{I}(\mathbf{X};\mathbf{Y}\,|\,\mathbf{Z})=\text{H}[\mathbf{X}\,|\,\mathbf{Z}]-\text{H}[\mathbf{X}\,|\,\mathbf{Y},\mathbf{Z}]$
$=\text{H}[\mathbf{X},\mathbf{Z}]+\text{H}[\mathbf{Y},\mathbf{Z}]-\text{H}[\mathbf{Z}]-\text{H}[\mathbf{X},\mathbf{Y},\mathbf{Z}]$
$=\text{I}(\mathbf{X};\mathbf{Y},\mathbf{Z})-\text{I}(\mathbf{X};\mathbf{Z})$
- $\text{I}(\mathbf{X};\mathbf{Y}\,|\,\mathbf{Z})=\text{I}(\mathbf{Y};\mathbf{X}\,|\,\mathbf{Z})$
- $\text{I}(\mathbf{X};\mathbf{Y};\mathbf{Z})\doteq\text{I}(\mathbf{X};\mathbf{Y})-\text{I}(\mathbf{X};\mathbf{Y}\,|\,\mathbf{Z})$, so the "information never hurts" principle does not hold for MI. Information about Z may reduce the MI between $\mathbf{X}$ and $\mathbf{Y}$

- Given (discrete) fn. $F:\mathcal{P}(\mathcal{X})\to\mathbb{R}$, the **marginal gain** of $\mathbf{x}\in\mathcal{X}$ given $A\subseteq\mathcal{X}$ is: $\Delta_F(\mathbf{x}\,|\,A)\doteq F(A\cup\{\mathbf{x}\})-F(A)$.
- The fn. is **submodular** iff for any $\mathbf{x}\in\mathcal{X}$ and any $A\subseteq B\subseteq\mathcal{X}$: $F(A\cup\{\mathbf{x}\})-F(A)\ge F(B\cup\{\mathbf{x}\})-F(B)$ or equally $\Delta_F(\mathbf{x}\,|\,A)\ge\Delta_F(\mathbf{x}\,|\,B)$. Submodularity can be interpreted as notion of "concavity" for discrete fns.
- It is called **monotone** if $F(A)\le F(B)$.

Maximization objective: monotone submodular function: $I(S)\doteq\text{I}(\mathbf{f}_S;\mathbf{y}_S)=\text{H}[\mathbf{f}_S]-\text{H}[\mathbf{f}_S\,|\,\mathbf{y}_S]$. $\text{H}[\mathbf{f}_S]$: uc in $\mathbf{f}_S$ before observing $\mathbf{y}_S$. $\text{H}[\mathbf{f}_S\,|\,\mathbf{y}_S]$ uc in $\mathbf{f}_S$ after observing $\mathbf{y}_S$. Max. MI is in general NP-hard.

- **Greedy**: Pick the locations $\mathbf{x}_1$ through $\mathbf{x}_n$ individually by greedily finding the location with the maximal MI, this provides a $(1-1/e)$-approximation of the optimum.
- **Uncertainty sampling**: Have already picked $S_t=\{\mathbf{x}_1,\dots,\mathbf{x}_t\}$; Solve the following:
$\mathbf{x}_{t+1}\doteq\arg\max_{\mathbf{x}\in\mathcal{X}}\Delta_I(\mathbf{x}\,|\,S_t)=\arg\max_{\mathbf{x}\in\mathcal{X}}\text{I}(f_\mathbf{x};y_\mathbf{x}\,|\,\mathbf{y}_{S_t})$.
Doesn't work with heteroscedastic noise: large aleatoric uc may dominate epistemic uc. In classification corresponds to selecting label that max. entropy of predicted label: $\mathbf{x}_{t+1}\doteq\arg\max_{\mathbf{x}\in\mathcal{X}}\text{H}[y_\mathbf{x}\,|\,\mathbf{x}_{1:t},y_{1:t}]$.

Bayesian active learning by disagreement (BALD): Identifies points $\mathbf{x}$ where models *disagree* about label $y_\mathbf{x}$ (each model is *confident* but predict different labels):
$\mathbf{x}_{t+1}\doteq\arg\max_{\mathbf{x}\in\mathcal{X}}\text{I}(\theta;y_\mathbf{x}\,|\,\mathbf{x}_{1:t},y_{1:t})=$
$\arg\max_{\mathbf{x}\in\mathcal{X}}\text{H}[y_\mathbf{x}\,|\,\mathbf{x}_{1:t},y_{1:t}]-\mathbb{E}_{\theta|\mathbf{x}_{1:t},y_{1:t}}\text{H}[y_\mathbf{x}\,|\,\theta]$

- **Inductive learning** extract general rules from data. Typically, we can directly observe $f(\mathbf{x})$ at any $\mathbf{x}$.
- **Transductive learning** make best pred. at particular $\mathbf{x}^\star$. Typically, cannot directly observe $f(\mathbf{x}^\star)$. Require gen. $f(\mathbf{s})$ from the behavior of f at other locations.

7 Bayesian Optimization

Cumulative regret for time horizon T associated with choices $\{\mathbf{x}_t\}_{t=1}^T$ is: $R_T\doteq\sum_{t=1}^T\underbrace{(\max_\mathbf{x}f^\star(\mathbf{x})-f^\star(\mathbf{x}_t))}_{\textit{instantaneous regret}}$.

Goal: Achieve sublinear regret: $\lim_{T\to\infty}R_T/T=0$ (requires balancing exploration and exploitation).

Algorithm 9.2: Bayesian optimization (with GPs)

```
initialize f ~ GP(μ_0, k_0)
for t = 1 to T do
    choose x_t = argmax_{x∈X} F(x; μ_{t−1}, k_{t−1})
    observe y_t = f(x_t) + ε_t
    perform a probabilistic update to obtain μ_t and k_t
```

- Common to use an **acquisition fn.** to greedily pick the next point to sample based on the current model.
- **Upper confidence bound (UCB)**:
$\mathbf{x}_{t+1}\doteq\arg\max_{\mathbf{x}\in\mathcal{X}}\mu_t(\mathbf{x})+\beta_{t+1}\sigma_t(\mathbf{x})$, where $\sigma_t(\mathbf{x})\doteq\sqrt{k_t(\mathbf{x},\mathbf{x})}$. If $\beta_t=0$ then UCB is purely exploitative; if $\beta_t\to\infty$, UCB recovers uc sampling. UCB fn. generally non-convex.

When choosing β_t appropriately: $R_T=\mathcal{O}(\sqrt{T\gamma_T})$, with $\gamma_T\doteq\max_{\substack{S\subseteq\mathcal{X}\\|S|=T}}\text{I}(\mathbf{f}_S;\mathbf{y}_S)=\max_{\substack{S\subseteq\mathcal{X}\\|S|=T}}\frac{1}{2}\log\det(\mathbf{I}+\sigma_n^{-2}\mathbf{K}_{SS})$, is the maximum information gain after T rounds.
- Linear: $\gamma_T=\mathcal{O}(d\log T)$
- Gaussian: $\gamma_T=\mathcal{O}((\log T)^{d+1})$
- Matérn for $\nu>\frac{1}{2}$: $\gamma_T=\mathcal{O}(T^{\frac{d}{2\nu+d}}(\log T)^{\frac{2\nu}{2\nu+d}})$

Thompson Sampling: At time $t+1$, we sample a fn. $\tilde f_{t+1}\sim p(\cdot\,|\,\mathbf{x}_{1:t},y_{1:t})$ from our posterior distr. Then, we simply max. $\tilde f_{t+1}$, $\mathbf{x}_{t+1}\doteq\arg\max_{\mathbf{x}\in\mathcal{X}}\tilde f_{t+1}(\mathbf{x})$.

8 Diffusion generative models

Let $\beta_t\in(0,1]$, $\bar\alpha_t=\prod_{s=1}^t\alpha_s$, and $\alpha_s=1-\beta_s$. Typically, β_t is monotonically increases, which implies that $\bar\alpha_t\to0$ and thus $x_T\to\mathcal{N}(0,\mathbf{I})$ for $T\to\infty$.
1. **Forward process:** Transform data points into (Gaussian) noise by using a fixed noising MC q:
$q(x_{1:T}\,|\,x_0)=\prod_{t=1}^T q(x_t\,|\,x_{t-1})$
$q(x_t\,|\,x_{t-1})=\mathcal{N}(x_t;\sqrt{1-\beta_t}x_{t-1},\beta_t\mathbf{I})$
$q(x_t\,|\,x_0)=\mathcal{N}(x_t;\sqrt{\bar\alpha_t}x_{t-1},(\bar\alpha-1)\mathbf{I})$
2. **Backward process:** Learn a denoising MC p matching the reversed forward process.
$p_\lambda(x_{t-1}\,|\,x_t)=\mathcal{N}(x_{t-1};\mu_\lambda(x_t,t),\boldsymbol{\Sigma}_\lambda(x_t,t))$
$p_\lambda(x_{0:T})=p_\lambda(x_T)\prod_{t=1}^T p_\lambda(x_{t-1}\,|\,x_t)$
$p_\lambda(x_0)=\int p_\lambda(x_{0:T})dx_{1:T}$ where $x_{1:T}$ latent vars.
3. **Generation:** Now generate novel data points by simulating the learned denoising MC p.
(1) Sample $x_1\sim p(X_1)$, (2) Sample $x_2\sim p(X_2\,|\,X_1=x_1)$,
(T) Sample $x_T\sim p(X_T\,|\,X_{T-1}=x_{T-1})$

Note $p_\lambda(x_0)$ is intractable. Idea: use VI. ELBO:
$\log p_\lambda(x_0)\ge\log p_\lambda(x_0)-\mathcal{D}_{\text{KL}}(q(\cdot\,|\,x_0)\|p_\lambda(\cdot\,|\,x_0))$
$=\mathbb{E}_{"}[\log p_\lambda(x_T)-\sum_{t=2}^T\log\frac{q(x_t\,|\,x_{t-1})}{p_\lambda(x_{t-1}\,|\,x_t)}-\log\frac{q(x_1\,|\,x_0)}{p_\lambda(x_0\,|\,x_1)}]$
$=\text{const}+\mathbb{E}_{"}[-\sum_{t=2}^T\underbrace{\mathcal{D}_{\text{KL}}(q(\cdot\,|\,x_t,x_0)\|p_\lambda(\cdot\,|\,x_t))}_{L_t}+\underbrace{\log p_\lambda(x_0\,|\,x_1)}_{L_1}]$
with $"\doteq x_{1:T}\sim q(\cdot\,|\,x_0)$. Now optimize this via **stochastic VI** using closed-form expression of this loss/the KL-divergence term with const. var. schedule:
$D_{\text{KL}}(q(\cdot\,|\,x_t,x_0)\|p_\lambda(\cdot\,|\,x_t))=\frac{1}{2\sigma_t^2}\|\mu_t'(x_t,x_0)-\mu_\lambda(x_t,t)\|_2^2+\text{const}$
with $\mu_t'(x_t,x_0)=\frac{\sqrt{\overline\alpha_t}\beta_t}{1-\overline\alpha_t}x_0+\frac{\sqrt{\alpha_t}(1-\overline\alpha_{t-1})}{(1-\overline\alpha_t)}x_t$

9 Markov Decision Processes (MDPs)

- A policy induces a MC $(X_t^\pi)_{t\in\mathbb{N}_0}$: $p^\pi(x'\,|\,x)\doteq\mathbb{P}(X_{t+1}^\pi=x'\,|\,X_t^\pi=x)=\sum_{a\in A}\pi(a\,|\,x)p(x'\,|\,x,a)$.
- The **discounted payoff** from time t is: $G_t\doteq\sum_{m=0}^\infty\gamma^mR_{t+m}$, for **discount factor** $\gamma\in[0,1)$.

- **State value fn.**: $v_t^\pi\doteq\mathbb{E}_\pi[G_t\,|\,X_t=x,A_t=a]$ measures avg. discounted payoff from time t starting from $x\in X$.
- **State-action value fn. (Q-fn.)**: $q_t^\pi(x,a)\doteq\mathbb{E}_\pi[G_t\,|\,X_t=x,A_t=a]=r(x,a)+\gamma\sum_{x'\in X}p(x'\,|\,x,a)\cdot v_{t+1}^\pi(x')$ measures avg. discounted payoff from time t starting from $x\in X$ and with playing action $a\in A$.

Bellman Expectation Equation:
- $v^\pi(x)=r(x,\pi(x))+\gamma\mathbb{E}_{x'|x,\pi(x)}[v^\pi(x')]$
- If stochastic policy: $v^\pi(x)=\mathbb{E}_{a\sim\pi(x)}[q^\pi(x,a)]$
$q^\pi(x,a)=r(x,a)+\gamma\mathbb{E}_{x'|x,a}\mathbb{E}_{a'\sim\pi(x')}[q^\pi(x',a')]$
- For deterministic: $v^\pi(x)=q^\pi(x,\pi(x))$.

Can be used to find v^π given policy π, by solving linear system of eq. in cubic time in size of state space. Can also be solved using fixed pt. iter: $\mathbf{B}^\pi\mathbf{v}\doteq\mathbf{r}^\pi+\gamma\mathbf{P}^\pi\mathbf{v}$.
$\|\mathbf{v}_t^\pi-\mathbf{v}^\pi\|_\infty=\|\mathbf{B}^\pi\mathbf{v}_{t-1}^\pi-\mathbf{B}^\pi\mathbf{v}^\pi\|_\infty\le\gamma\|\mathbf{v}_{t-1}^\pi-\mathbf{v}^\pi\|_\infty\le\gamma\|\mathbf{v}_0^\pi-\mathbf{v}^\pi\|_\infty$

Bellman's theorem: A policy $\pi^\star$ is optimal iff it's greedy w.r.t its own value fn. In other words, $\pi^\star$ is optimal iff $\pi^\star(x)$ is a distr. over set $\arg\max_{a\in A}q^\star(x,a)$.

- If for every state there is a unique action that max. the q-fn., $\pi^\star$ is deter. and unique, $\pi^\star(x)=\arg\max_{a\in A}q^\star(x,a)$.
- For finite MDPs, PI converges to $\pi^\star$ in poly. num. of iter. Each step takes cubic comp. in the num. of states.

Algorithm 10.14: Policy iteration

```
initialize π (arbitrarily)
repeat
    compute v^π
    compute π_{v^π}
    π ← π_{v^π}
until converged
```

Monotonic improvement of PI:
- $v^{\pi_{t+1}}(x)\ge v^{\pi_t}(x)$ for all $x\in X$
- $v^{\pi_{t+1}}(x)>v^{\pi_t}(x)$ for at least one $x\in X$, unless $v^{\pi_t}\equiv v^\star$

Algorithm 10.17: Value iteration

```
initialize v(x) ← max_{a∈A} r(x,a) for each x ∈ X
for t = 1 to ∞ do
    v(x) ← (B*v)(x) = max_{a∈A} q(x,a) for each x ∈ X
choose π_v
```

- VI converges to an optimal policy, as $v^\star$ and $q^\star$ are a fixed-points of the Bellman update $\mathbf{B}^\star$.
- For any $\epsilon>0$, VI converges to an ϵ-optimal solution in poly time. However, unlike PI, VI does not generally reach the exact optimum in a finite num. of iter.

POMDP: Markov process, with **observations** Y, and **observation probs.** $o(y\,|\,x)\doteq\mathbb{P}(Y_t=y\,|\,X_t=x)$. Hard to solve in gen., can conv. to MDP with larger state space.

10 Tabular Reinforcement Learning

Markovian property of the underlying MDP:
$X_{t=1}\perp X_{t'+1}\,|\,X_t,X_{t'+1},A_t,A_{t'}$ $R_t\perp X_{t'}\,|\,R_t,X_{t'+1},A_t,A_{t'}$

Bootstrapping: approx. a true quantity by using an empirical quantity, which itself is constructed using samples from the true quantity that is to be approx.

For model-based approaches MLE yields:
$\hat p(x'\,|\,x,a)=\frac{N(x'\,|\,x,a)}{N(a\,|\,x)}$ and $\hat r(x,a)=\frac{1}{N(a\,|\,x)}\sum_{t=0,x_t=x,a_t=a}^\infty r_t$
Both unbiased as they correspond to a sample mean.
- $N(x'\,|\,x,a)$ num. trans. from x to x' when play a
- $N(a\,|\,x)$ nums trans. from x and play a.

Greedy in the limit with inf. exploration (GLIE):
1. All state-action pairs are explored infinitely many times: $\lim_{t\to\infty}N_t(x,a)=\infty$
2. The policy converges to a greedy policy: $\lim_{t\to\infty}\pi_t(a\,|\,x)=\mathbf{1}\{a=\arg\max_{a'\in A}Q_t^\star(x,a')\}$

Robbins-Montro (RM) conditions: for a sequence $(\alpha)_{t\in\mathbb{N}_0}$ if: $\alpha_t\ge0$, $\sum_{t=0}^\infty\alpha_t=\infty$, $\sum_{t=0}^\infty\alpha_t^2<\infty$.

Algorithm 11.2: ε-greedy

```
for t = 0 to ∞ do
    sample u ∈ Unif([0,1])
    if u ≤ ε_t then pick action uniformly at random among all actions
    else pick best action under the current model
```

- Ignores all past experience. • Will eventually converge.
- ϵ-greedy is GLIE with prob. 1 if $(\epsilon_t)_{t\in\mathbb{N}_0}$ satisfies the **Robbins-Montro (RM)** conditions (e.g., $\epsilon_t=1/t$).

Softmax/Boltzmann exploration: alt. to ϵ-greedy $\pi_\lambda(a\,|\,x)\propto\exp(\frac{1}{\lambda}Q^\star(x,a))$ (Gibbs). For $\lambda\to0$ greedily max. Q-fn. For $\lambda\to\infty$ uniform random exploration.

Algorithm 11.6: $R_{\max}$ algorithm On-policy, Model-based

```
add the fairy-tale state x* to the Markov decision process
set r̂(x,a) = R_max for all x ∈ X and a ∈ A
set p̂(x* | x,a) = 1 for all x ∈ X and a ∈ A
compute the optimal policy π̂ for r̂ and p̂
for t = 0 to ∞ do
    execute policy π̂ (for some number of steps)
    for each visited state-action pair (x,a), update r̂(x,a)
    estimate transition probabilities p̂(x' | x,a)
    after observing "enough" transitions and rewards, recompute the
     optimal policy π̂ according the current model p̂ and r̂.
```

- Optimism in the face of uncertainty. Init. with max rew.
- Every T steps, with high prob., either obtains near-optimal reward; or visits one unknown state-action pair.
- With prob. at least $1-\delta$, $R_{\max}$ reaches ϵ-optimal π in poly. num. steps in $|X|$, $|A|$, T, $1/\epsilon$, $1/\delta$, and $R_{\max}$.

Algorithm 11.9: Temporal-difference (TD) learning On-policy, Model-free

```
initialize V^π arbitrarily (e.g., as 0)
for t = 0 to ∞ do
    follow policy π to obtain the transition (x,a,r,x')
    V^π(x) ← (1 − α_t)V^π(x) + α_t(r + γV^π(x'))
```

- If α_t satisfies RM conditions and all state-action pairs are chosen inf. often, then V^π conv. to v^π w. prob 1.
- For estimates V^π to converge true v^π, the transitions that are used for the estimation must follow policy π.

SARSA: Same as TD but estimate Q with update:
$Q^\pi(x,a)\leftarrow(1-\alpha_t)Q^\pi(x,a)+\alpha_t(r+\gamma Q^\pi(x',a'))$
Same convergence guarantees as TD. On-policy, Model-free

Algorithm 11.12: Q-learning Off-policy, Model-free

```
initialize Q*(x,a) arbitrarily (e.g., as 0)
for t = 0 to ∞ do
    observe the transition (x,a,r,x')
    Q*(x,a) ← (1 − α_t)Q*(x,a) + α_t(r + γ max_{a'∈A} Q*(x',a'))
```

- If α_t satisfies RM conditions and all state-action pair are visited inf. often, then $Q^\star$ conv. to $q^\star$ with prob. 1.
- With prob. at least $1-\delta$, conv. to ϵ-optimal policy in num. steps that is poly. in $\log|X|$, $\log|A|$, $\frac{1}{\epsilon}$ and $\log\frac{1}{\delta}$.

Optimistic Q-learning: Similar to $R_{\max}$ Init.
$Q^\star(x,a)=V_{\max}\prod_{t=1}^{T_{\text{init}}}(1-\alpha_t)^{-1}$ w. $V_{\max}\doteq\frac{R_{\max}}{1-\gamma}\ge\max q^\star(x,a)$.
- With prob. at least $1-\delta$, ϵ-optimal π after num. steps that is poly. in $|X|$, $|A|$, $\frac{1}{\epsilon}$, $\log\frac{1}{\delta}$, and $R_{\max}$ where init. time T_{init} is upper bounded by a poly. in same coeff.
- If T_{init} chosen large enough, converges quickly to $\pi^\star$.

11 Model-free Reinforcement Learning

In tabular methods:
- Storing val. fn., we need at least $O(|\mathcal{X}|)$ space.
- Storing Q-fn, we even need $O(|\mathcal{X}|\cdot|\mathcal{A}|)$ space.

Time req. to compute value fn. for every state-action pair exactly will grow poly. in size of the state-action space.

Can view TD-/Q-learning as SGD on the squared loss: $\ell(\theta;x,r,x')\doteq\frac{1}{2}(r+\gamma\theta^{\text{old}}(x')-\theta(x))^2$ and learn param. approx. of $V(\mathbf{x};\theta)$ or $Q(\mathbf{x},\mathbf{a};\theta)$ using Monte Carlo est. and bootstrapping; Main reason model-free methods (TD-/Q-learning) are usually **sample inefficient**:
- Bootstrapping leads to "(initially) incorrect" and "unstable" targets of the optimization
- Monte Carlo est. with single sample leads to large var.

Q-learning with fn. approx.: (1) Observe $\mathbf{x}'$ and r from picking a in $\mathbf{x}$. (2) Update $\theta\leftarrow\theta+\alpha_t\delta_{\text{B}}\phi(\mathbf{x},\mathbf{a})$, where $\delta_{\text{B}}\doteq r+\gamma\max_{\mathbf{a}'\in\mathcal{A}}Q^\star(\mathbf{x}',\mathbf{a}';\theta^{\text{old}})-Q^\star(\mathbf{x},\mathbf{a};\theta)$.
- In the tabular setting, this is identical to Q-learning
- Converges to the true Q-function $q^\star$.

"Tricks of the trade" to improve SGD:
- **Stabilizing opti. targets:** Bootstrapping est. changes after each iteration, leading to stability issues. **DQN** updates NN used for bootstrapping infrequently and maintains const. opti. target across multiple episodes. E.g. clone: hanging/online NN and fixed/target NN.
- **Max. bias:** Estimates $Q^\star$ are noisy (biased) estimates of $q^\star$. **DDQN**: instead of picking optimal action w.r.t. old network, it picks w.r.t. new network:

Policy val. fn.: measures discounted payoff of policy: $J(\pi)\doteq\mathbb{E}_\pi[G_0]=\mathbb{E}_\pi[\sum_{t=0}^\infty\gamma^tR_t]$, and bounded ver.: $J_T(\pi)\doteq\mathbb{E}_\pi[G_{0:T}]=\mathbb{E}_\pi[\sum_{t=0}^{T-1}\gamma^tR_t]$. Non-convex.

Score grad. est.: $\nabla_\varphi\mathbb{E}_{\tau\sim\Pi_\varphi}[G_{0:T}]=\mathbb{E}[G_{0:T}\nabla_\varphi\log\Pi_\varphi(\tau)]$
Typically var. of est. is very large. Reduce w. **baselines:** $\mathbb{E}[G_{0:T}\nabla_\varphi\log\Pi_\varphi(\tau)]=\mathbb{E}[(G_{0:T}-b)\nabla_\varphi\log\Pi_\varphi(\tau)]$

Algorithm 12.8: REINFORCE algorithm On-policy, Model-free

```
initialize policy weights φ
repeat
    generate an episode (i.e., rollout) to obtain trajectory τ
    for t = 0 to T − 1 do
        set g_{t:T} to the downstream return from time t
        φ ← φ + ηγ^t g_{t:T} ∇_φ log π_φ(a_t | x_t)
until converged
```

- SGD with score grad est. and downstream returns.
- Not guaranteed to find an optimal policy. Can get stuck in local optima even for very small domains.

Advantage fn.: $a^\pi(\mathbf{x},\mathbf{a})\doteq q^\pi(\mathbf{x},\mathbf{a})-v^\pi(\mathbf{x})$
$=q^\pi(\mathbf{x},\mathbf{a})-\mathbb{E}_{\mathbf{a}'\sim\pi(\mathbf{x})}[q^\pi(\mathbf{x},\mathbf{a}')]$
- π is optimal $\Longleftrightarrow\forall\mathbf{x}\in\mathcal{X},\mathbf{a}\in\mathcal{A}:a^\pi(\mathbf{x},\mathbf{a})\le0$

Policy gradient theorem: Max. $J(\varphi)$ corresponds to incr. the prob. of actions with large and decr. the prob. of actions with small value, taking into account how often the resulting policy visits certain states.
$\nabla_\varphi J(\varphi)=\sum_{t=0}^\infty\mathbb{E}_{\mathbf{x}_t,\mathbf{a}_t}[\gamma^tq^{\pi_\varphi}(\mathbf{x}_t,\mathbf{a}_t)\nabla_\varphi\pi_\varphi(\mathbf{a}_t,\mathbf{x}_t)]$

Algorithm 12.11: Online actor-critic On-policy, Online, Model-free

```
initialize parameters φ and
repeat
    use π_φ to obtain transition (x,a,r,x') and the next a
     a' ~ π_φ(· | x')
    δ = r + γQ(x',a';θ) − Q(x,a;θ)
    // actor update
    φ ← φ + ηQ(x,a;θ)∇_φ log π_φ(a | x)
    // critic update
    θ ← θ + ηδ∇_θ Q(x,a;θ)
until converged
```

- Use SARSA for learning critic; SGD for gradient est.
- Actor is not guaranteed to improve

- **TRPO:** optimizes via KL-div. constraints for stable, monotonic improv. Uses 2nd-order natural grad. to prevent performance collapse. (on-policy, model-free, policy-gradient)
- **PPO:** heuristic variants of TRPO; replace constrained opti. prob. by unconstr. opti. of regularized objective (on-policy, model-free, actor-critic) • **GRPO:** normalizes rewards across group samples to eliminate the critic network and reduce memory (on-policy,model-free, critic-less) • **DDPG:** uses det. policy grad. and experience replay for cont. action spaces. Combines Q-lean. stability with AC arch. (off-policy, model-free, deterministic) • **SAC:** Max. reward plus policy entropy to ensure robust exploration and stability. Prevents premature convergence in complex continuous control tasks (off-policy, model-free, stochastic) • **DPO:** Maps RLHF objectives directly to a cross-entropy loss without explicit reward modeling or RL loops. (offline, model-free, ref.-based)
- **PETS:** combines probabilistic ensembles to capture uncertainty with MPC for planning. (model-based, stochastic, MPC) • **UCRL:** implements optimistic exploration via plausible models to min. regret. (model-based, exploration, opt. i.t.f.o. uc) • **H-UCRL:** extends UCRL to hierarchical structures to manage exploration. (model-based, opt. i.t.f.o. uc)

Adapted by Samuel Stante (HS2025) from Nils Jensen (HS2023)