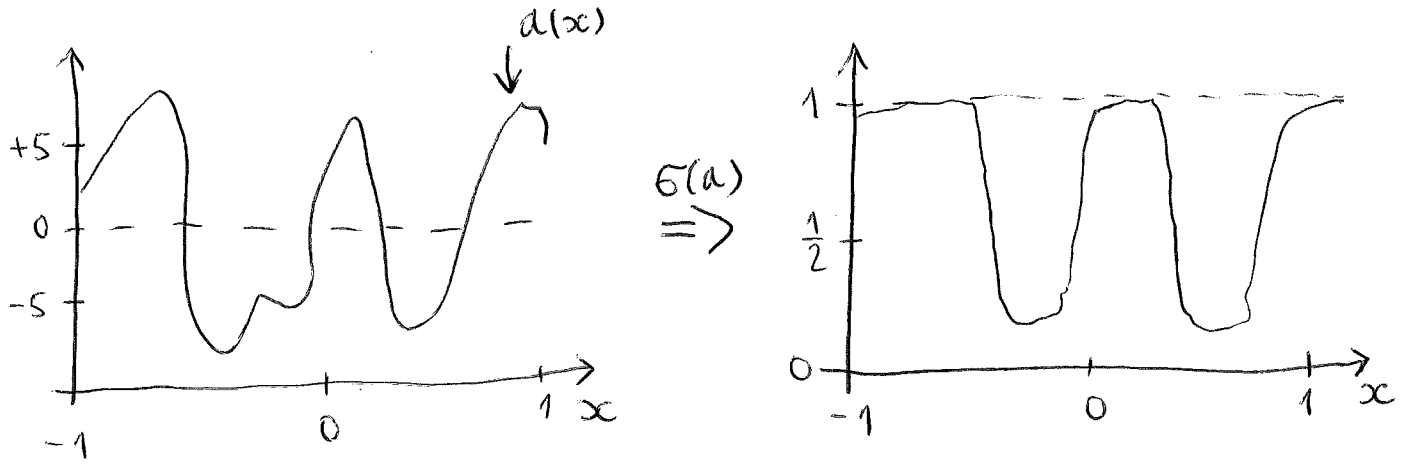
Gaussian processes for classification

Consider $K = 2$, s.t. $t \in \{0, 1\}$.

Idea: define a gaussian process over $a(\vec{x})$ & then use $y = \sigma(a)$ to obtain a non-gaussian stochastic process over functions $y(\vec{x})$, s.t. $y(\vec{x}) \in (0, 1)$, $\forall x$.

For example, in the 1D case



Recall that $p(t|a) = \sigma(a)^t (1 - \sigma(a))^{1-t}$.

Training dataset: $\vec{x}_1 \dots \vec{x}_N \Rightarrow \vec{t} = t_1 \dots t_N$

also, $\vec{x}_{N+1} \Rightarrow t_{N+1}$

We need predictive distr'n $p(t_{N+1} | \vec{t})$

Define $\vec{a}_{N+1} = a(\vec{x}_1) \dots a(\vec{x}_{N+1})$, then

$p(\vec{a}_{N+1}) \overset{\text{gaussian process}}{\nearrow} \mathcal{N}(\vec{a}_{N+1} | \vec{0}, C_{N+1})$, where

$C_{N+1, nm} = C_{N+1}(\vec{x}_n, \vec{x}_m) =$

$= \underbrace{k(\vec{x}_n, \vec{x}_m)}_{\text{pos. semidefinite kernel f'n}} + \underbrace{\nu \delta_{nm}}_{\text{small pos. const}}$

In general, the kernel f'n $k(\vec{x}, \vec{x}')$ depends on hyperparameters $\vec{\theta}$.

Since $K=2$, we can focus on $p(t_{N+1}=1 \mid \vec{t}_N)$ since $p(t_{N+1}=0 \mid \vec{t}_N) = 1 - p(t_{N+1}=1 \mid \vec{t}_N)$.

Then

$$\underbrace{p(t_{N+1}=1 \mid \vec{t}_N)}_{\text{predictive distr'n}} = \int da_{N+1} \overbrace{p(t_{N+1}=1 \mid a_{N+1})}^{\sigma(a_{N+1})} \times \underbrace{p(a_{N+1} \mid \vec{t}_N)}_{\text{non-gaussian, need an approx'n}}$$

Use Laplace:

$$p(a_{N+1} \mid \vec{t}_N) = \int d\vec{a}_N \, p(a_{N+1}, \vec{a}_N \mid \vec{t}_N) =$$

$$= \int d\vec{a}_N \, \frac{p(\vec{t}_N \mid a_{N+1}, \vec{a}_N) \, p(a_{N+1}, \vec{a}_N)}{p(\vec{t}_N)} =$$

(Bayes theorem)

$$= \frac{1}{p(\vec{t}_N)} \int d\vec{a}_N \underbrace{p(\vec{t}_N \mid \vec{a}_N)}_{\text{does not depend on } a_{N+1}} p(a_{N+1} \mid \vec{a}_N) \, p(\vec{a}_N) \overset{\text{Bayes again}}{=}$$

$$\frac{p(\vec{t}_N \mid \vec{a}_N) \, p(\vec{a}_N)}{p(\vec{t}_N)} = p(\vec{a}_N \mid \vec{t}_N)$$

$$= \int d\vec{a}_N \underbrace{p(a_{N+1} \mid \vec{a}_N)}_{\text{gaussian}} \underbrace{p(\vec{a}_N \mid \vec{t}_N)}_{\text{non-gaussian}}$$

Just as with regression,

$$p(a_{N+1} \mid \vec{a}_N) = \mathcal{N}(a_{N+1} \mid \vec{k}^T C_N^{-1} \vec{a}_N, \; c - \vec{k}^T C_N^{-1} \vec{k}).$$

Now, use Laplace on $p(\vec{a}_N|\vec{t}_N)$:

go back to $p(\vec{a}_N|\vec{t}_N) = \dfrac{p(\vec{t}_N|\vec{a}_N)\,p(\vec{a}_N)}{p(\vec{t}_N)}$, where

$p(\vec{a}_N) = \mathcal{N}(\vec{a}_N|\vec{0}, C_N)$ and

$$p(\vec{t}_N|\vec{a}_N) = \prod_{n=1}^{N} \sigma(a_n)^{t_n}(1-\sigma(a_n))^{1-t_n} = \prod_{n=1}^{N} e^{a_n t_n}\sigma(-a_n)$$

$$\underbrace{\sigma(a)^t}_{\frac{1}{(1+e^{-a})^t}}\underbrace{(1-\sigma(a))^{1-t}}_{\left(\frac{e^{-a}}{1+e^{-a}}\right)^{1-t}} = \frac{e^{a(t-1)}}{1+e^{-a}} = \underbrace{\frac{e^{-a}}{1+e^{-a}}}_{\sigma(-a)=\frac{1}{1+e^{a}}} e^{at}$$

Then $\log p(\vec{a}_N|\vec{t}_N) = \overbrace{\log p(\vec{a}_N) + \log p(\vec{t}_N|\vec{a}_N)}^{\text{“}\Psi(\vec{a}_N)} -$

$$- \log p(\vec{t}_N) = -\frac{1}{2}\vec{a}_N^T C_N^{-1}\vec{a}_N - \frac{N}{2}\log(2\pi) -$$

$$- \frac{1}{2}\log|C_N| + \vec{t}_N^T\vec{a}_N - \sum_{n=1}^{N}\log(1+e^{a_n}) - \log p(\vec{t}_N).$$

$$\nabla_{\vec{a}}\log p(\vec{a}_N|\vec{t}_N) = \nabla_{\vec{a}}\Psi(\vec{a}_N) = \vec{t}_N - C_N^{-1}\vec{a}_N - \vec{\sigma}_N.$$

$$\frac{\partial}{\partial a_i}\sum_{n=1}^{N}\log(1+e^{a_n}) = \sum_n \frac{1}{1+e^{a_n}} e^{a_n}\delta_{ni} = \underbrace{\frac{e^{a_i}}{1+e^{a_i}}}_{\sigma(a_i)}$$

Here, $\vec{\sigma}_N = \overbrace{\sigma(a_1)\dots\sigma(a_N)}$

$$\nabla_{\vec{a}}\Psi(\vec{a}_N) = 0 \Rightarrow \underbrace{\vec{\sigma}_N + C_N^{-1}\vec{a}_N}_{\text{nonlinear in } a_i} = \vec{t}_N$$

Need to find the solution by e.g. Newton-Raphson method:

$$\nabla_{\vec{a}} \nabla_{\vec{a}} \Psi(\vec{a}_N) = -C_N^{-1} - W_N$$

$$\frac{\partial}{\partial a_j} \sigma(a_i) = \sigma(a_i)(1-\sigma(a_i))\,\delta_{ij}$$

$$W_N = \begin{pmatrix} \sigma(a_1)(1-\sigma(a_1)) & & 0 \\ & \ddots & \\ 0 & & \sigma(a_N)(1-\sigma(a_N)) \end{pmatrix}$$

W_N pos. def.

C_N pos. def. $\Rightarrow C_N^{-1}$ pos. def., yielding

Hessian $A = -\nabla_{\vec{a}} \nabla_{\vec{a}} \Psi = W_N + C_N^{-1}$ pos. def.

That means that $\log p(\vec{a}_N | \vec{t}_N)$ is convex everywhere $\rightarrow$ there is a single global maximum, so NR should work very well.

Specifically,

$$\vec{a}_N^{\,new} = \vec{a}_N^{\,old} + A^{-1} \underbrace{\nabla_{\vec{a}} \log p(\vec{a}_N | t_N)}_{\nabla_{\vec{a}} \Psi(\vec{a}_N)} =$$

$$= \vec{a}_N^{\,old} + \underbrace{(W_N + C_N^{-1})^{-1}}_{C_N(\mathbb{I} + W_N C_N)^{-1}} \left[\vec{t}_N - \vec{\sigma}_N - C_N^{-1} \vec{a}_N^{\,old} \right] =$$

$$C_N C_N^{-1} (W_N + C_N^{-1})^{-1} = C_N \left((W_N + C_N^{-1}) C_N \right)^{-1}$$

$$= C_N (\mathbb{I} + W_N C_N)^{-1} \left[\vec{t}_N - \vec{\sigma}_N + (\mathbb{I} + W_N C_N) C_N^{-1} \vec{a}_N^{\,old} - C_N^{-1} \vec{a}_N^{\,old} \right] =$$

(=) $C_N(\mathbb{I} + W_N C_N)^{-1}[\vec{t}_N - \vec{\sigma}_N + W_N \vec{a}_N^{old}]$.

Iterate to convergence $\Rightarrow$ find $\vec{a}_N^*$ s.t.

$$\nabla_{\vec{a}} \Psi(\vec{a}_N)\Big|_{\vec{a}_N^*} = \vec{0} \Rightarrow \underbrace{\vec{a}_N^* = C_N(\vec{t}_N - \vec{\sigma}_N)}_{\text{non-linear eq'n in } \vec{a}_N^*}$$

Now, $H = -\nabla_{\vec{a}}\nabla_{\vec{a}} \Psi(\vec{a}_N)\Big|_{\vec{a}_N^*} = W_N + C_N^{-1}$.

Finally, Laplace:

$$p(\vec{a}_N|\vec{t}_N) \Rightarrow q(\vec{a}_N) = \mathcal{N}(\vec{a}_N|\vec{a}_N^*, H^{-1}).$$

So, $p(a_{N+1}|\vec{t}_N) = \int d\vec{a}_N \mathcal{N}(a_{N+1}|\vec{k}^T C_N^{-1}\vec{a}_N, c - \vec{k}^T C_N^{-1}\vec{k}) \times$

$$\times \mathcal{N}(\vec{a}_N|\vec{a}_N^*, H^{-1})$$

(2.115): $p(\vec{y}) = \int d\vec{x}\, p(\vec{y}|\vec{x})\, p(\vec{x}) =$

$$= \int d\vec{x}\, \mathcal{N}(\vec{y}|A\vec{x}+\vec{b}, L^{-1})\, \mathcal{N}(\vec{x}|\vec{\mu}, \Lambda^{-1}) =$$

$$= \mathcal{N}(\vec{y}|A\vec{\mu}+\vec{b},\ L^{-1} + A\Lambda^{-1}A^T)$$

$$\begin{cases} \vec{y} \to a_{N+1} \\ \vec{x} \to \vec{a}_N \end{cases} \Rightarrow \begin{cases} A\vec{x}+\vec{b} \to \vec{k}^T C_N^{-1}\vec{a}_N \\ \vec{\mu} = \vec{a}_N^*,\ \Lambda^{-1} = H^{-1} \end{cases}$$

$\left[\begin{matrix} A \to \vec{k}^T C_N^{-1} \\ \vec{b} \to 0 \end{matrix}\right]$ gives vector of length N, $L^{-1} \to c - \vec{k}^T C_N^{-1}\vec{k}$

$$p(a_{N+1}|\vec{t}_N) = \mathcal{N}(a_{N+1}|\vec{k}^T C_N^{-1}\vec{a}_N^*,\ c - \vec{k}^T C_N^{-1}\vec{k} + \vec{k}^T C_N^{-1} H^{-1} C_N^{-1}\vec{k})$$

$(C_N^{-1})^T = C_N^{-1}$ (11)

$$\vec{k}^T C_N^{-1} \underbrace{\vec{a}_N^*}_{C_N(\vec{t}_N - \vec{\sigma}_N)} = \vec{k}^T(\vec{t}_N - \vec{\sigma}_N),$$

$$c - \vec{k}^T \left[C_N^{-1} - C_N^{-1}(W_N + C_N^{-1})^{-1} C_N^{-1} \right] \vec{k} =$$

[illegible]

$$= c - \vec{k}^T \left[C_N^{-1} - C_N^{-1}(W_N C_N + \mathbb{I})^{-1} \right] \vec{k} =$$

$$= c - \vec{k}^T \left[\left(C_N^{-1} \cdot (W_N C_N + \mathbb{I}) - C_N^{-1} \right) (W_N C_N + \mathbb{I})^{-1} \right] \vec{k} =$$

$$= c - \vec{k}^T \left[W_N (W_N C_N + \mathbb{I})^{-1} \right] \vec{k} = c - \vec{k}^T (C_N + W_N^{-1})^{-1} \vec{k}.$$

$$\boxed{\exists}\ \mathcal{N}\left(a_{N+1} \mid \underbrace{\vec{k}^T(\vec{t}_N - \vec{\sigma}_N)}_{\mu}, \underbrace{c - \vec{k}^T(C_N + W_N^{-1})^{-1}\vec{k}}_{\sigma^2}\right).$$

Finally,

$$p(t_{N+1} = 1 \mid \vec{t}_N) = \int da_{N+1}\, \sigma(a_{N+1})\, \mathcal{N}(a_{N+1} \mid \mu, \sigma^2) \simeq$$

$$\overset{(4.153)}{\simeq} \sigma(\kappa(\sigma^2)\mu), \quad \text{where}$$

$$\kappa(\sigma^2) = \frac{1}{\sqrt{1 + \frac{\pi\sigma^2}{8}}}.$$

What about hyperprms $\vec{\theta}$?

Maximize $p(\vec{t}_N|\vec{\theta}) \Rightarrow \vec{\theta}_{ML}$

Use $p(\vec{t}_N|\vec{\theta}) = \int d\vec{a}_N\, p(\vec{t}_N|\vec{a}_N)\, p(\vec{a}_N|\vec{\theta})$,

apply Laplace: $\int \underbrace{d\vec{z}}_{M\text{ dim}} f(\vec{z}) \simeq f(\vec{z}_0)\, \frac{(2\pi)^{M/2}}{|A|^{1/2}}$

$$\log p(\vec{t}_N|\vec{\theta}) \simeq \underbrace{\log p(\vec{t}_N|\vec{a}_N^*) + \log p(\vec{a}_N^*|\vec{\theta})}_{\Psi(\vec{a}_N^*)} +$$

(first term depends on $\vec{a}_N^*$; second term depends on $\vec{\theta}$)

$$+ \frac{N}{2}\log(2\pi) - \frac{1}{2}\underbrace{\log|W_N + C_N^{-1}|}_{\text{depends on } \vec{\theta}}.$$

To maximize this, we need to find

$\frac{\partial \log p(\vec{t}_N|\vec{\theta})}{\partial \theta_j}$ in terms of $\frac{\partial C_N}{\partial \theta_j}$ & $\frac{\partial \vec{a}_N^*}{\partial \theta_j}$

Note that $\begin{cases} C_N = C_N(\vec{\theta}), \\ \vec{a}_N^* = \vec{a}_N(\vec{\theta}). \end{cases}$

Recall that $\begin{cases} p(\vec{t}_N|\vec{a}_N^*) = \prod_{n=1}^{N} e^{a_n^* t_n}\, \sigma(-a_n^*), \\ p(\vec{a}_N^*|\vec{\theta}) = \mathcal{N}(\vec{a}_N^*|\vec{0}, C_N) \end{cases}$

Finally, note that

$$\frac{\partial \vec{a}_N^*}{\partial \theta_j} = \frac{\partial C_N}{\partial \theta_j}(\vec{t}_N - \vec{\sigma}_N) - C_N \underbrace{\frac{\partial \vec{\sigma}_N}{\partial \theta_j}}_{\text{vector with elements}}, \text{ or}$$

$$\frac{\partial \sigma(a_n^*)}{\partial \theta_j} = \underbrace{\sigma(a_n^*)(1-\sigma(a_n^*))}_{W_{N,nn}} \frac{\partial a_n^*}{\partial \theta_j}$$

$$\Downarrow$$

$$\frac{\partial \vec{\sigma}_N}{\partial \theta_j} = W_N \frac{\partial \vec{a}_N^*}{\partial \theta_j}$$

$$\frac{\partial \vec{a}_N^*}{\partial \theta_j} = \frac{\partial C_N}{\partial \theta_j}(\vec{t}_N - \vec{\sigma}_N), \text{ or}$$

$(\mathbb{I} + C_N W_N)$

$$\frac{\partial \vec{a}_N^*}{\partial \theta_j} = (\mathbb{I} + \underbrace{W_N C_N}_{= C_N W_N})^{-1} \frac{\partial C_N}{\partial \theta_j}(\vec{t}_N - \vec{\sigma}_N).$$

So $\frac{\partial \vec{a}_N^*}{\partial \theta_j}$ can be expressed through $\frac{\partial C_N}{\partial \theta_j}$

Differentiating C_N wrt θ_j is relatively straightforward as $C(\vec{x}_n, \vec{x}_m) = k(\vec{x}_n, \vec{x}_m) + \nu \delta_{nm}$ (depends on $\vec{\theta}$)

with $\frac{\partial \log p(\vec{t}_N | \vec{\theta})}{\partial \theta_j}$ available, can use e.g. conjugate gradient to maximize $\log p(\vec{t}_N | \vec{\theta})$ & find $\vec{\theta}_{ML}$.