Exercise Sheet 5

Robin Schmidt
Probabilistic Inference & Learning

November 18, 2018

Gaussian Process Regression

1. (a)

please see provided code in jupyter notebook.

Adding the second running period:

```
# time frame of running
runstart    = date2num(datetime(2009,7,1));
runend      = date2num(datetime(2009,12,5));
runstart2    = date2num(datetime(2013,8,1));
runend2      = max(X);
```

Modifying phi_run:

```
phi_run  = lambda t : ( (t > runstart) * (t < runend) * (t -
   runstart) + (t >= runend) * (runend - runstart) \
    + (t > runstart2) * (t < runend2) * (t - runstart2) + (t >=
       runend2) * (runend2 - runstart2) ) / 1000.
```

Plotting the posterior mean and standard deviation with:

```
# Plotting the standard deviation and the posterior mean
fig2 = plt.figure();
plt.plot_date(Xd,Y, '.', color=dark);
plt.plot(t,mpost, '-', color=red, lineWidth=4)
plt.plot(t,spost, '-', color=dark, lineWidth=4)
```

This plots the posterior mean in *mpost* which is constant at 31 Dec 2013 and afterwards of -5.63411469 and the standard deviation in *spost* which grows with time onwards is around 1.6.

1. (b)

-

2. (a)

Using the properties from lecture 06 for Gaussian processes the marginal prior distribution for f_X is:

$$p(f_X) = \mathcal{N}(f_X; \mu_X, k_{XX})$$

With k_{XX} being a gaussian like kernel:

$$k(X_i, X_j) = \theta^2 exp(-\frac{(X_i - X_j)^2}{2\lambda^2})$$

2. (b)

Recalling the lecture "Gaussian distributions" and the fact that Gaussians are closed under linear maps gives:

$$\begin{aligned} p(f_X) &= \mathcal{N}(f_X; \mu_X, k_{XX}) \\ f_A &= Af_X \\ p(f_A) &= \mathcal{N}(Af_X; A\mu_X, Ak_{XX}A^T) \end{aligned}$$

Also recalling that Gaussians are closed under conditioning from lecture $03 - 17$ gives:

$$\begin{aligned} p(f_B) &= \mathcal{N}(Bf_X; B\mu_X, Bk_{XX}B^T) \\ p(Y|f_X) &= \mathcal{N}(Y; Af_X, \sigma^2 I_m) \\ p(f_B|Y) &= \frac{p(f_B, Y)}{p(Y)} = \mathcal{N}(f_B; B\mu_{f_B} + k_{f_B}YK_{YY}^{-1}(Y - Af_X), k_{f_B f_B} - k_{f_B Y}k_{YY}^{-1}k_{f_B Y}) \end{aligned}$$

2. (c)

Because the derivation is a linear operation we can just use the method from above:

$$\begin{aligned} f'(x) &= \frac{d}{dx}f(x) \\ Tf(x) &= \frac{d}{dx}f(x) \\ p(f) &= GP(f; 0, k) \\ p(f') &= GP(f'; 0, k') \\ p(f') &= GP(f'; T0, TkT^{-1}) = GP(f'; 0, \nabla k) \end{aligned}$$

The marginal is a Gaussian process, because marginals over Gaussians are also Gaussians (recalled from lecture 03-17):

$$\int \mathcal{N}\left[\begin{pmatrix} x \\ y \end{pmatrix}; \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}, \begin{pmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{pmatrix}\right] = \mathcal{N}(x; \mu_x, \Sigma_{xx})$$

With the following gradient:

$$
\begin{aligned}
\nabla k &= \frac{\partial k}{\partial a} + \frac{\partial k}{\partial b} \\
&= \frac{\partial}{\partial a}(\theta^2 exp\Big(-\frac{(a-b)^2}{2\lambda^2}\Big)) + \frac{\partial}{\partial b}(\theta^2 exp\Big(-\frac{(a-b)^2}{2\lambda^2}\Big)) \\
&= -\frac{\theta^2(a-b)exp(-\frac{(a-b)^2}{2\lambda^2})}{\lambda^2} + \frac{\theta^2(a-b)exp(-\frac{(a-b)^2}{2\lambda^2})}{\lambda^2} \\
&= 0
\end{aligned}
$$

2. (d)