Question 0

0.1

- $\phi^{(1)}$ should be a trigonometric feature transform in order to capture the periodicity of the time-temperature relationship.
- $\phi^{(2)}$ should be a linear feature transform to capture the linear relationship between latitude and temperature without introducing the overfitting that could potentially arise from a polynomial transform.
- $\phi^{(4)}$ could either be a polynomial or an RBF transform since these are both very flexible thus would be able to capture a wide range of possible CO2-temperature relationships. One (small) drawback of the RBF transform is that the bandwidth hyperparameter (σ) has to be chosen beforehand, though heuristics do exist to help decide this value.

0.2 Accidentally including $x^{(2)}$ could lead to (**A**) overfitting (i.e. (**C**) a decreased training error and (**E**) increased testing error) since the model will try to fit to the peculiarities of the training set's (useless) longitudinal information, which is unlikely to match up with the pattern of longitudinal data in the training set, thus causing an increased testing error. Hence the outcome that will **not** happen is (**B**)—the model underfitting.

Question 1

1.1 We have that:

$$p(f_1(\mathbf{x}_1)...f_1(\mathbf{x}_n)|\mathbf{K}) = \mathcal{N}_{f_1(\mathbf{x}_1)...f_1(\mathbf{x}_n)}(\mathbf{0}, \mathbf{K}) \tag{1.1}$$

$$p(y_1...y_n|f_1(\mathbf{x}_1)...f_1(\mathbf{x}_n), \sigma) = \mathcal{N}_{y_1...y_n}(f_1(\mathbf{x}_1)...f_1(\mathbf{x}_n), \sigma^2\mathbf{I}). \tag{1.2}$$

From Pattern Recognition & Machine Learning (PRML) [1] equations 2.115-2.117 we also have that, given a marginal Gaussian distribution for $\hat{\mathbf{x}}$ and a conditional Gaussian distribution for $\hat{\mathbf{y}}$ of the form:

$$p(\hat{\mathbf{x}}) = \mathcal{N}_{\hat{\mathbf{x}}}(\boldsymbol{\mu}, \Lambda^{-1}) \tag{1.3}$$

$$p(\hat{\mathbf{y}}|\hat{\mathbf{x}}) = \mathcal{N}_{\hat{\mathbf{y}}}(\mathbf{A}\hat{\mathbf{x}} + \mathbf{b}, \mathbf{L}^{-1}). \tag{1.4}$$

Then the marginal distribution of $\hat{\mathbf{y}}$ is given by:

$$p(\hat{\mathbf{y}}) = \mathcal{N}_{\hat{\mathbf{y}}}(\mathbf{A}\boldsymbol{\mu}, \mathbf{L}^{-1} + \mathbf{A}\Lambda^{-1}\mathbf{A}^T). \tag{1.5}$$

First note that eq. 1.1 corresponds to eq. 1.3 with $\hat{\mathbf{x}} = f_1(\mathbf{x}_1)...f_1(\mathbf{x}_n)$, $\boldsymbol{\mu} = \mathbf{0}$ and $\Lambda^{-1} = \mathbf{K}$. Next observe that then eq. 1.2 corresponds to eq. 1.4 with $\hat{\mathbf{y}} = \mathbf{y}|\sigma$, $\mathbf{A} = \mathbf{I}$, $\mathbf{b} = \mathbf{0}$ and $\mathbf{L}^{-1} = \sigma^2\mathbf{I}$, then eq. 1.5 gives us the desired result:

$$p(\mathbf{y}) = \mathcal{N}_{\mathbf{y}}(\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}).$$

1.2 First let us partition $\mathbf{y}$ into:

$$\mathbf{y} = \begin{pmatrix} y_1 \\ \tilde{\mathbf{y}} \end{pmatrix}$$

(so $\tilde{\mathbf{y}} = (y_2, ..., y_n)^T$) and further partition $\Sigma := \sigma^2\mathbf{I} + \mathbf{K}$ into the following submatrices:

$$\Sigma = \begin{pmatrix} c & \mathbf{k}^T \\ \mathbf{k} & \mathbf{C} \end{pmatrix}$$

where $c = \sigma^2 + k(x_1, x_1)$, $\mathbf{k} = (k(x_1, x_2), k(x_1, x_3), ..., k(x_1, x_n))^T$ and $\mathbf{C}$ is an $(n-1) \times (n-1)$ matrix with the (i, j)th entry given by $\sigma^2\mathbf{I}_{i,j} + \mathbf{K}_{i+1,j+1}$. Note here that we are defining the kernel function $k(x_i, x_j) := \mathbf{K}_{i,j}$ as the (i, j)th element of $\mathbf{K}$.

Now since $p(\mathbf{y}) = \mathcal{N}_\mathbf{y}(\mathbf{0}, \Sigma)$, using equations 2.81 and 2.82 in PRML [1] we find that $p(y_1|\tilde{\mathbf{y}}, \mathbf{K}, \sigma) = \mathcal{N}_{y_1}(\mu_{y_1|\tilde{\mathbf{y}}}, \Sigma_{y_1|\tilde{\mathbf{y}}})$ where

$$\begin{aligned} \mu_{y_1|\tilde{\mathbf{y}}} &= \mathbf{0} + \mathbf{k}^T\mathbf{C}^{-1}(\tilde{\mathbf{y}} - \mathbf{0}) &&= \mathbf{k}^T\mathbf{C}^{-1}\tilde{\mathbf{y}} \\ \Sigma_{y_1|\tilde{\mathbf{y}}} &= c - \mathbf{k}^T\mathbf{C}^{-1}\mathbf{k} &&= \sigma^2 + k(x_1, x_1) - \mathbf{k}^T\mathbf{C}^{-1}\mathbf{k}. \end{aligned}$$

Hence,

$$p(y_1|\tilde{\mathbf{y}}, \mathbf{K}, \sigma) = \mathcal{N}_{y_1}(\mathbf{k}^T\mathbf{C}^{-1}\tilde{\mathbf{y}}, \sigma^2 + k(x_1, x_1) - \mathbf{k}^T\mathbf{C}^{-1}\mathbf{k}).$$

1.3 First we partition $\mathbf{K}$ in the following submatrices:

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{pmatrix}$$

where $\mathbf{K}_{11} = k(x_1, x_1)$, $\mathbf{K}_{12} = (k(x_2, x_1), k(x_3, x_1), ..., k(x_n, x_1)) = \mathbf{K}_{21}^T$ and $\mathbf{K}_{22}$ is an $(n-1)\times(n-1)$ matrix with the (i, j)th entry given by $\mathbf{K}_{i+1,j+1}$. Then using slide 23 from lecture 8, the kernel regression using $\{(y_i, \mathbf{x}_i)\}_{i=2}^n$ as training data would lead to the predictive function at $\mathbf{x}_1$:

$$\mathbf{f}(\mathbf{x}_1, \mathbf{w}_{LS}) = \mathbf{K}_{21}^T(\mathbf{K}_{22} + \lambda\mathbf{I})^{-1}\tilde{\mathbf{y}}^T$$

with regularization coefficient $\lambda > 0$.

Notice that by the definition of $\mathbf{C}$ and $\mathbf{k}$ in the previous question we have that

$$\begin{aligned} \mathbf{C} &= \mathbf{K}_{22} + \sigma^2\mathbf{I} \\ \mathbf{k} &= \mathbf{K}_{21}^T. \end{aligned}$$

This shows that the regularised kernel regression arises naturally from the probabilistic approach in Q1.2, i.e. with regularisation coefficient $\lambda = \sigma^2$:

$$\mathbf{f}(\mathbf{x}_1, \mathbf{w}_{LS}) = \mathbb{E}[y_1|\tilde{\mathbf{y}}, \mathbf{K}, \sigma].$$

1.4 Unlike in classic least squares with a linear model, a Gaussian Process does not require that the data points $(\mathbf{x}_i, y_i)$ be iid, therefore having the advantage of being applicable to a wider range of datasets (such as time series in which the iid assumption often doesn't hold).

Question 2

2.1 In the Week 4 Additional Questions (Q2) we showed that under these same assumptions, with $\mathbf{h}_i := \boldsymbol{\phi}^T(\mathbf{x}_i)(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\boldsymbol{\Phi}$ and $\text{Var}[\varepsilon_i] = \sigma^2$, we have that:

$$\text{Var}[f(\mathbf{x}_i; \mathbf{w}_{LS})|\mathbf{x}_i] = \langle \mathbf{h}_i, \mathbf{h}_i \rangle \sigma^2.$$

When we apply the above result to this question we see that:

$$\begin{aligned}
\frac{1}{n}\sum_{i\in D} \text{Var}[f(\mathbf{x}_i; \mathbf{w}_{LS})|\mathbf{x}_i] &= \frac{1}{n}\sum_{i\in D} \langle \mathbf{h}_i, \mathbf{h}_i \rangle \sigma^2 \\
&= \frac{\sigma^2}{n}\sum_{i\in D} \text{tr}(\mathbf{h}_i\mathbf{h}_i^T) \\
&= \frac{\sigma^2}{n}\sum_{i\in D} \text{tr}(\boldsymbol{\phi}^T(\mathbf{x}_i)(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\boldsymbol{\phi}(\mathbf{x}_i)) \\
&= \frac{\sigma^2}{n}\sum_{i\in D} \text{tr}(\boldsymbol{\phi}^T(\mathbf{x}_i)(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\boldsymbol{\phi}(\mathbf{x}_i)) \\
&= \frac{\sigma^2}{n}\sum_{i\in D} \text{tr}(\boldsymbol{\phi}(\mathbf{x}_i)\boldsymbol{\phi}^T(\mathbf{x}_i)(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}) \\
&= \frac{\sigma^2}{n}\text{tr}(\sum_{i\in D} \boldsymbol{\phi}(\mathbf{x}_i)\boldsymbol{\phi}^T(\mathbf{x}_i)(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}) \\
&= \frac{\sigma^2}{n}\text{tr}(\boldsymbol{\Phi}\boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}) \\
&= \frac{\sigma^2}{n}\text{tr}(\mathbf{I}_b) \\
&= \frac{\sigma^2}{n}b
\end{aligned}$$

(since $\boldsymbol{\Phi}$ is a $b\times n$ matrix and so $\boldsymbol{\Phi}\boldsymbol{\Phi}^T$ is $b\times b$). From this it is clear that $\frac{1}{n}\sum_{i\in D} \text{Var}[f(\mathbf{x}_i; \mathbf{w}_{LS})|\mathbf{x}_i]$ increases as b increases.

2.2 Using the bias-variance decomposition of the in-sample error $\frac{1}{n}\sum_{i\in D} \mathbb{E}_D[[y_i - f(\mathbf{x}_i; \mathbf{w}_{LS})]^2|\mathbf{x}_i]$, notice that

$$\frac{1}{n}\sum_{i\in D} \mathbb{E}_D[[y_i - f(\mathbf{x}_i; \mathbf{w}_{LS})]^2|\mathbf{x}_i] = \frac{1}{n}\sum_{i\in D}\left[\text{Var}[\varepsilon] + [g(\mathbf{x}_i) - \mathbb{E}[f_{LS}(\mathbf{x}_i)]|\mathbf{x}_i]^2 + \text{Var}[f_{LS}(\mathbf{x}_i)|\mathbf{x}_i]\right].$$

In Q1 of the Week 4 Additional Questions we showed that (with our assumptions) the second term inside the summation (the bias) is equal to zero. Combining this with the previous question's result we see that

$$\begin{aligned}
\frac{1}{n}\sum_{i\in D} \mathbb{E}_D[[y_i - f(\mathbf{x}_i; \mathbf{w}_{LS})]^2|\mathbf{x}_i] &= \frac{1}{n}\sum_{i\in D}\left[\sigma^2 + 0 + \text{Var}[f_{LS}(\mathbf{x}_i)|\mathbf{x}_i]\right] \\
&= \frac{1}{n}\sum_{i\in D}\sigma^2 + \frac{1}{n}\sum_{i\in D}\text{Var}[f_{LS}(\mathbf{x}_i)|\mathbf{x}_i] \\
&= \sigma^2 + \frac{\sigma^2}{n}b.
\end{aligned}$$

Clearly as n increases, this value decreases.

2.3 We can see that as $\boldsymbol{n}$ decreases the variance of the prediction increases, meaning that the prediction is highly dependent on the particular training dataset (and the noise therein), leading to overfitting. The variance also increases when we increase $\boldsymbol{b}$, similarly leading to overfitting.

References

[1] Christopher M. Bishop. *Pattern recognition and machine learning*. Information science and statistics. Springer, New York, 2006.