A APPENDIX: VARIATIONAL LOWER BOUND

In this appendix we give some more details on the computation of the variational lower bound for the variationally constrained model.

The augmented joint probability density (after introducing the inducing points) takes the form,

$$\begin{aligned} &p(\mathbf{Y}, \mathbf{F}, \mathbf{U}, \mathbf{X}|\mathbf{X}_u) \\ &\quad = p(\mathbf{Y}|\mathbf{F})p(\mathbf{F}|\mathbf{U}, \mathbf{X}, \mathbf{X}_u)p(\mathbf{U}|\mathbf{X}_u)p(\mathbf{X}) \\ &\quad = \left(\prod_{j=1}^{p} p(\mathbf{y}_j|\mathbf{f}_j)p(\mathbf{f}_j|\mathbf{u}_j, \mathbf{X}, \mathbf{X}_u)p(\mathbf{u}_j|\mathbf{X}_u)\right) p(\mathbf{X}). \end{aligned}$$

In the r.h.s above, the observed inputs $\mathbf{Z}$ do not appear, exactly because we introduce them through the variational constraint, which does not constitute a probabilistic mapping. In the above equations we have

$$p(\mathbf{f}_j|\mathbf{u}_j, \mathbf{X}, \mathbf{X}_u) = \mathcal{N}\left(\mathbf{f}_j|\mathbf{a}_j, \mathbf{\Sigma}_f\right),$$

being the conditional GP prior with

$$\mathbf{a}_j = \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}\mathbf{u}_j \quad \text{and} \quad \mathbf{\Sigma}_f = \mathbf{K} - \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}\mathbf{K_{uf}}$$

and

$$p(\mathbf{u}_j|\mathbf{X}_u) = \mathcal{N}(\mathbf{u}_j|\mathbf{0}, \mathbf{K_{uu}}),$$

is the marginal GP prior over the inducing variables. In the above expressions, $\mathbf{K_{uu}}$ denotes the covariance matrix constructed by evaluating the covariance function on the inducing points, $\mathbf{K_{uf}}$ is the cross-covariance between the inducing and the latent points and $\mathbf{K_{fu}} = \mathbf{K_{uf}^{\top}}$.

In order to perform variational inference in this expanded probability model, we introduce the variational distributions $q(\mathbf{X}|\mathbf{Z})$ and $q(\mathbf{U})$, which are both taken to be Gaussian. For convenience, we drop the inducing points $\mathbf{X}_u$ from our expressions for the remainder of the Appendix, for convenience. We now have:

$$\begin{aligned} &\log p(\mathbf{Y}|\mathbf{X}_u) = \\ &\qquad \log \int_{\mathbf{U},\mathbf{X}} p(\mathbf{U})p(\mathbf{X}) \int_{\mathbf{F}} p(\mathbf{Y}|\mathbf{F})p(\mathbf{F}|\mathbf{U}, \mathbf{X}). \end{aligned}$$

By applying Jensen's inequality, we obtain a lower bound $\mathcal{F}(q(\mathbf{X}), q(\mathbf{U}))$ on the above marginal likelihood, where:

$$\begin{aligned} &\mathcal{F}(q(\mathbf{X}|\mathbf{Z}), q(\mathbf{U})) = \\ &\quad \int_{\mathbf{U},\mathbf{X}} q(\mathbf{U})q(\mathbf{X}|\mathbf{Z}) \log \frac{p(\mathbf{U})p(\mathbf{X}) \int_{\mathbf{F}} p(\mathbf{Y}|\mathbf{F})p(\mathbf{F}|\mathbf{U}, \mathbf{X})}{q(\mathbf{U})q(\mathbf{X}|\mathbf{Z})} \\ &= \int_{\mathbf{U},\mathbf{X}} q(\mathbf{U})q(\mathbf{X}|\mathbf{Z}) \log \frac{p(\mathbf{U}) \int_{\mathbf{F}} p(\mathbf{Y}|\mathbf{F})p(\mathbf{F}|\mathbf{U}, \mathbf{X})}{q(\mathbf{U})} \\ &- \mathrm{KL}\left(q(\mathbf{X}|\mathbf{Z}) \,\|\, p(\mathbf{X})\right) \\ &\coloneqq \hat{\mathcal{F}} - \mathrm{KL}\left(q(\mathbf{X}|\mathbf{Z}) \,\|\, p(\mathbf{X})\right). \end{aligned}$$

At this point, our variational bound is similar to the one of equation (7), but the first term, here denoted as $\hat{\mathcal{F}}$, refers to the expanded probability space and, thus, involves the inducing inputs and the additional variational distribution $q(\mathbf{U})$. Since the second term (the KL term) is tractable (because it only involves Gaussian distributions), we are now going to focus on the $\hat{\mathcal{F}}$ term. By breaking the logarithm again, we can further write this term as:

$$\begin{aligned} \hat{\mathcal{F}} = &\int_{\mathbf{U},\mathbf{X}} q(\mathbf{U})q(\mathbf{X}|\mathbf{Z}) \log \left(\int_{\mathbf{F}} p(\mathbf{Y}|\mathbf{F})p(\mathbf{F}|\mathbf{U}, \mathbf{X})\right) \\ &- \mathrm{KL}\left(q(\mathbf{U}) \,\|\, p(\mathbf{U})\right) \quad \text{(A.1)}. \end{aligned}$$

We notice that we can make use of Jensen's inequality once more, because:

$$\log \left(\int_{\mathbf{F}} p(\mathbf{Y}|\mathbf{F})p(\mathbf{F}|\mathbf{U}, \mathbf{X})\right) \geq \int_{\mathbf{F}} p(\mathbf{F}|\mathbf{U}, \mathbf{X}) \log p(\mathbf{Y}|\mathbf{F}).$$

This expectation is analytically tractable. Indeed, for a single dimension j, we can find this expectation as:

$$\begin{aligned} &\int_{\mathbf{f}_j} p(\mathbf{f}_j|\mathbf{u}_j, \mathbf{X}) \log p(\mathbf{y}_j|\mathbf{f}_j) = \\ &\qquad \log \mathcal{N}\left(\mathbf{y}_j|\mathbf{a}_j, \beta^{-1}\mathbf{I}\right) - \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}\right) \\ &\qquad + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K_{uu}^{-1}}\mathbf{K_{uf}}\mathbf{K_{fu}}\right), \end{aligned}$$

where $\mathbf{K}$ is the covariance matrix constructed by evaluating the covariance function on the training inputs $\mathbf{X}$. The full expression can be found by taking the appropriate product with respect to dimensions; indeed, since the joint probability factorises with respect to output dimensions j, then a bound to the logarithm of the marginal likelihood can be written as a sum over terms, where every term considers a single dimension j. Notice that to obtain this tractable bound we did not explicitly make the assumption of equation (9) about the form of the variational distribution. However, this assumption is still made implicitly and the equivalence of the two derivations is rather instructive with respect to the effect of a variational constraint.

We also notice that in the above expression, the covariance matrix $\mathbf{K}$ is no longer inverted. Therefore, by writting the term $\hat{\mathcal{F}}$ in this form, we manage to obtain an expression which allows the uncertainty in $\mathbf{X}$ to be propagated through the GP mapping.

It is possible to also obtain a "tighter" variational bound $\mathcal{F}(q(\mathbf{X}|\mathbf{Z})) \geq \mathcal{F}(q(\mathbf{U}), q(\mathbf{X}|\mathbf{Z}))$ which does not depend on $q(\mathbf{U})$. To do so, we need to "collect" all terms that contain $p(\mathbf{U})$ from equation (A.1) and find the stationary point with respect to the distribution $q(\mathbf{U})$ (by computing the gradient w.r.t $q(\mathbf{U})$ and setting it to zero). By doing so, we are then able to replace $q(\mathbf{U})$ with its optimal value back to the variational bound. Titsias and Lawrence [2010] further explain this trick.

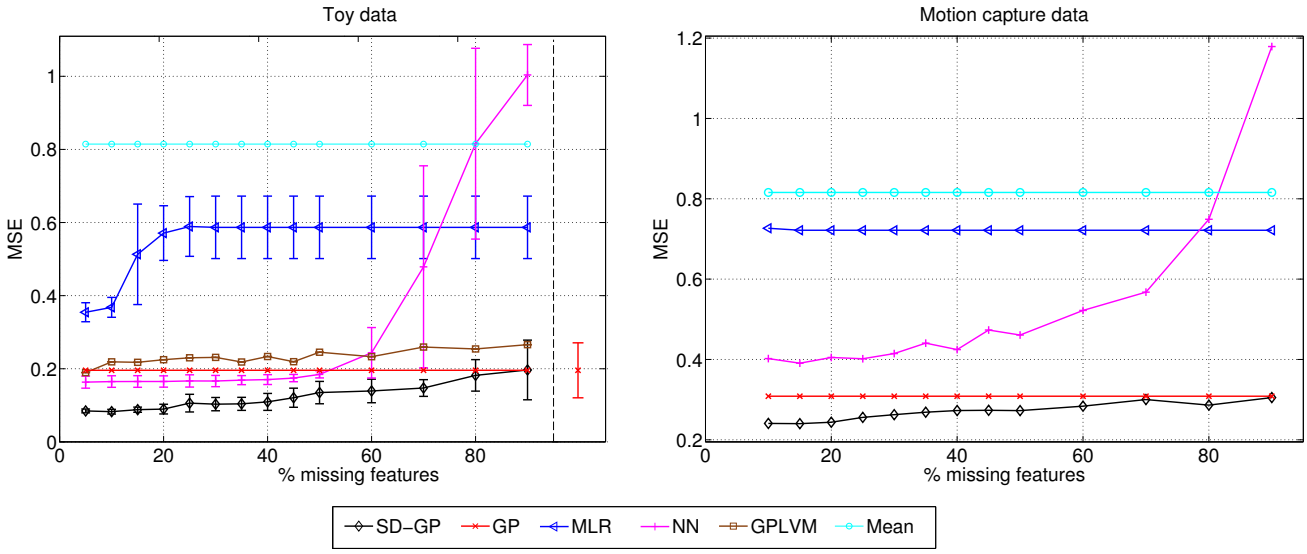

Figure 5: MSE for predictions obtained by different methods on semi-described learning (full version of figure 2). Comparing our method (SD-GP), the standard GP method, multiple linear regression (MLR), nearest neighbour regression on the input space (NN), the data-imputation method based on GP-LVM and the mean predictor (mean). The results for simulated data are obtained from 4 trials. The GP method cannot handle partial observations, thus the uncertainty (2σ) is constant; for clarity, the errorbar is plotted separately on the right of the dashed vertical line (for nonsensical x values). The GP-LVM method produced huge errorbars (about 3.5 times larger than thos of MLR), thus we don't plot them here, for clarity.

B APPENDIX: MORE DETAILS FOR THE SEMI-DESCRIBED LEARNING EXPERIMENT

In Section 3.1 we looked at performing predictions with Gaussian processes trained from partially observed inputs. Our method (semi-described GP or SD-GP) was compared to other approaches in figure 2, but the limit in the $y-$axis was fixed to a smaller value to show the comparison with the standard GP method more clearly. For the same reason, methods which produced very large errors were omitted. In this appendix we show the full figure from all the obtained results – figure 5.

The conclusion drawn from figure 5 is that our method is very efficient in taking into account the extra, partially observed input set $\mathbf{Z}^u$. This is true even if this extra set only has a small proportion of features observed. On the other hand, nearest neighbour runs into difficulties when real data are considered and, even worse, produces huge errors when more than 60% of the features are missing in $\mathbf{Z}^u$. Finally, the baseline which uses the standard GP-LVM as a means of imputing missing values produces bad results, in fact worse compared to if the extra set $\mathbf{Z}^u$ is just ignored (i.e. the GP baseline). This is because the baseline using GP-LVM treats the input space as single point estimates; by not incorporating (and optimising jointly) the uncertainty for each input location, the model has no way of ignoring "bad" imputed values.

C APPENDIX: MORE DETAILS FOR THE AUTO-REGRESSIVE EXPERIMENT

This appendix refers to the auto-regressive Gaussian process model developed in Section 3.2. In figure 3 we showed the results from the last 310 steps of the iterative forecasting task. Here (figure 6) we show the rest of the predictive sequence, obtained for extrapolating up until 1110 steps. The corresponding quantification of the error is shown in Table 1.

Table 1: Mean squared and mean absolute error obtained when extrapolating in the chaotic time-series data. $\text{GP}_{\text{uncert}}$ refers to the basic (moment matching) method of Girard et al. [2003] and the "naive" autoregressive GP approach is the one which does not propagate uncertainties.

Method	MAE	MSE
ours	**0.529**	**0.550**
$\text{GP}_{\text{uncert}}$	0.700	0.914
"naive" GP approach	0.799	1.157

D APPENDIX: THE EFFECT OF q, p, n IN SEMI-DESCRIBED LEARNING

As mentioned in Section 3.1, we found that when q is large compared to p and n, then the data imputation step of our

algorithm can be problematic as the percentage of missing features in $\mathbf{Z}^{\mathcal{U}}$ approaches 100%. This is somehow a corner-case, but it still shows that the method is reliant on having some covariates available. To investigate further this issue we created simulated data as explained in Section 3.1, but this time multiple datasets were generated with different input and output dimensions, q and p respectively. In figure 7 we show the comparison of SD-GP and the standard GP (which ignores $\mathbf{Z}^{\mathcal{U}}$) for different selections of q, p and percentage of missing features in $\mathbf{Z}^{\mathcal{U}}$.

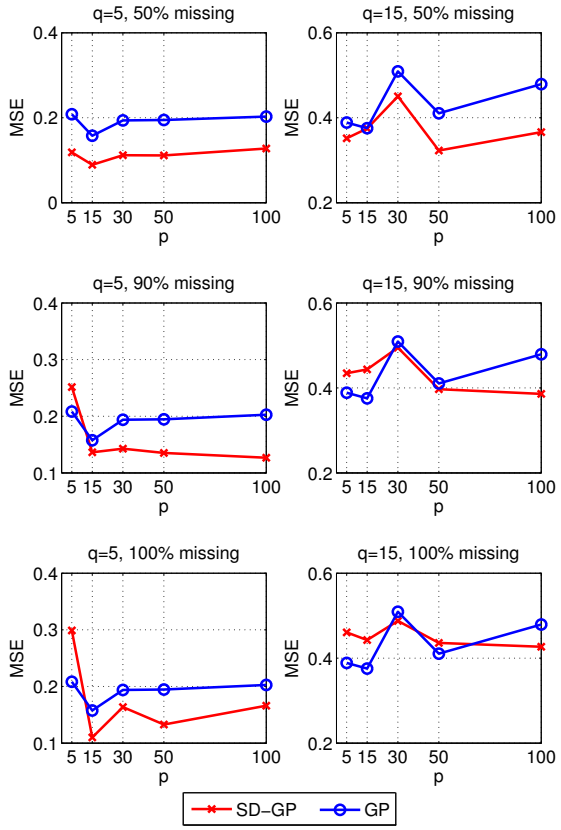

Figure 7: Comparison of our method (SD-GP) and the standard GP (which ignores $\mathbf{Z}^{\mathcal{U}}$) for different selections of q, p and percentage of missing features in $\mathbf{Z}^{\mathcal{U}}$.

The summary of this experiment is that:

- For the most usual scenarios, i.e. when the percentage of features missing is not too high, SD-GP performs very well, but as p and n become small compared to q, then the performance of the method seems to deteriorate.

- Even if 100% of the features are missing in $\mathbf{Z}^{\mathcal{U}}$, using our SD-GP can still be advantageous compared to using a standard GP. This is because SD-GP can utilise the extra information in the fully observed outputs, $\mathbf{Y}^{\mathcal{U}}$, which correspond to the fully missing set $\mathbf{Z}^{\mathcal{U}}$. However, when the percentage of missing features is very large and the relative size of p and n is small compared to q, then the method can produce worst results compared to the standard GP.

To explain the challenge of handling missing values with SD-GP, consider that a separate variational parameter exists for every input, namely the parameters $\mu^{\mathcal{U}}_{i,j}, S^{\mathcal{U}}_{i,j}, i = 1, ...n, j = 1, ..., q$ of step 7 in Algorithm 1. In the extreme cases mentioned in the previous paragraph, the number of variational parameters remains large but the available covariates to learn from are too few. This renders the optimisation of the parameters very difficult.

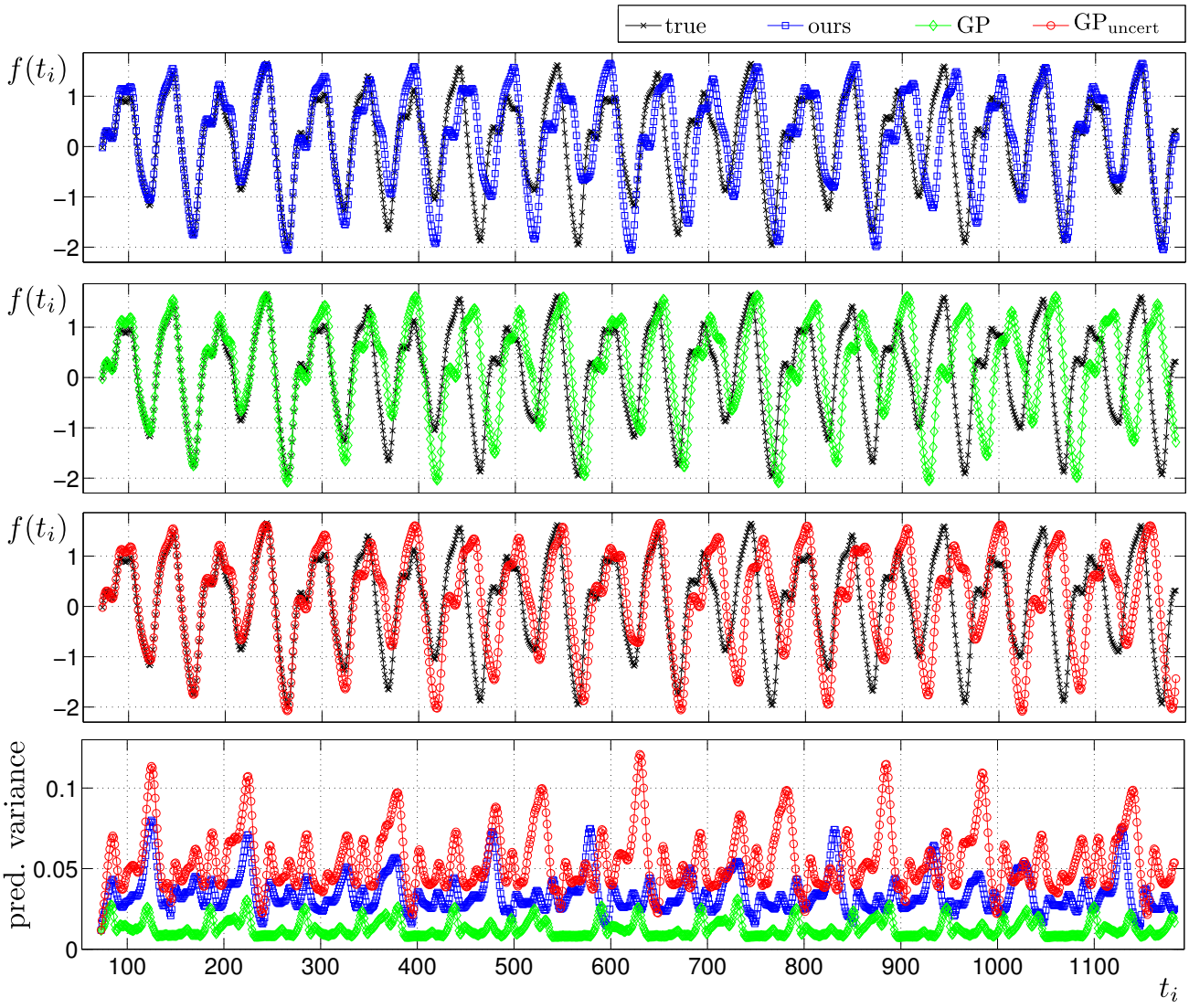

Figure 6: The full predictions obtained by the competing methods for the chaotic time-series data. The top 3 plots show the values obtained in each predictive step for each of the compared methods; the plot on the bottom shows the corresponding predictive uncertainties (2σ). GP$_{\text{uncert}}$ refers to the basic (moment matching) method of Girard et al. [2003] and the GP is the "naive" autoregressive GP which does not propagate uncertainties.