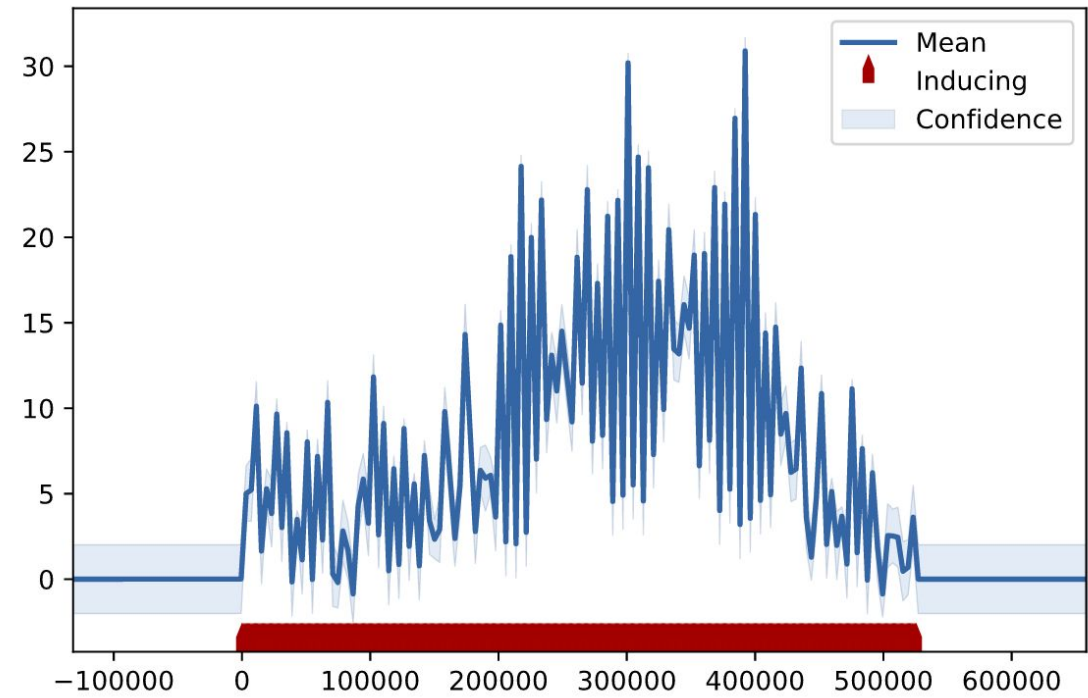


OUTPUTS:

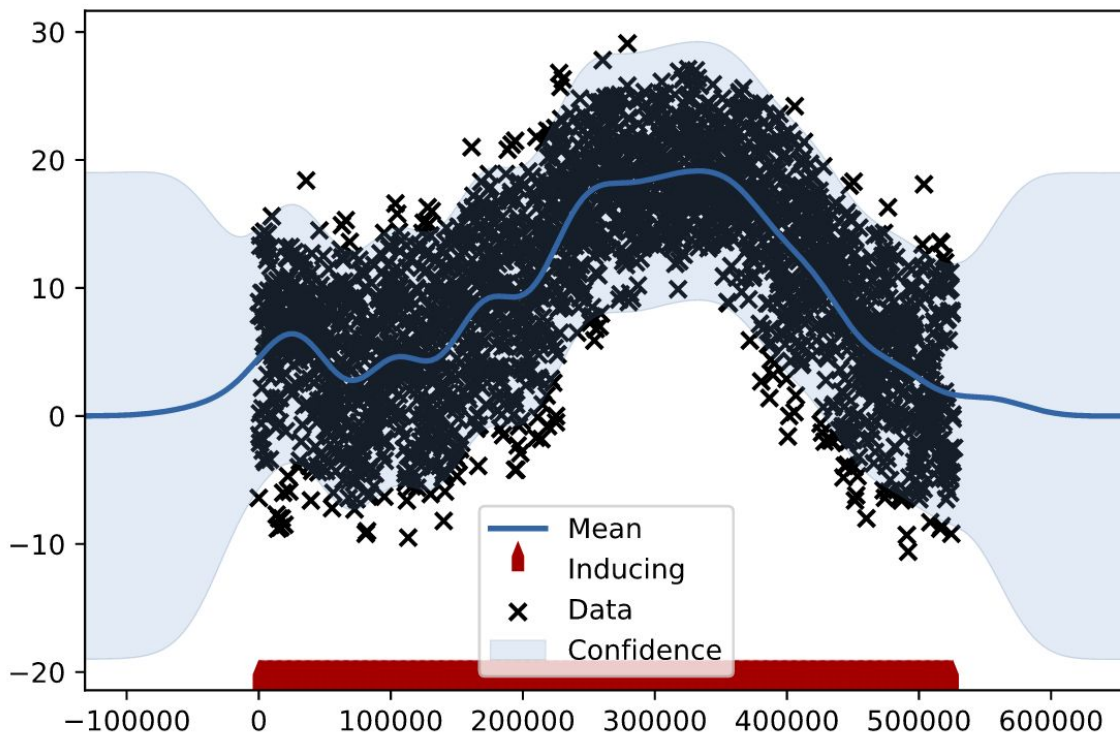
Name : sparse_gp
Objective : 1293395.61811
Number of Parameters : 603
Number of Optimization Parameters : 603
Updates : True
Parameters:

sparse_gp.	value	constraints	priors
inducing_inputs	(600, 1)		
rbf.variance	1.0	+ve	
rbf.lengthscale	200.0	+ve	
Gaussian_noise.variance	0.05	+ve	



Name : sparse_gp
 Objective : 8071.08734089
 Number of Parameters : 603
 Number of Optimization Parameters : 3
 Updates : True
 Parameters:

sparse_gp.	value	constraints	priors
inducing_inputs	(600, 1)	fixed	
rbf.variance	67.6623326878	+ve	
rbf.lengthscale	37600.8834062	+ve	
Gaussian_noise.variance	26.3557945084	+ve	



Describe in detail the covariance function you chose, and why. Did you fit any hyperparameters, and if so, how?

I chose an RBF as the covariance function because temperature usually changes very smoothly and does not drop off quickly like stock markets. RBF will be good for this compared to some exponential kernel. Ideally I'd use a periodic kernel to account for the periodic element of daily changes in temperature, however the documentation in GPy is hard for me to understand how to use Periodic kernels. Will try to improve that for next time I use Sparse GPs.

Did you fit any hyperparameters, and if so, how?

I used chose an appropriate lengthscale and high number of inducing inputs for best results. I figure temperature doesn't change very drastically over a couple hours so I made my lengthscale about 200 minutes and I used 600 inducing inputs to spread over all the data.

MSE = 26.1280558332

Train time = 0.392135858536

Cross-validation time (on the 2011-2014 data): 3.1 seconds

Fit time (best cross-validated choice on 2011-2014 data): 0.35 seconds

Prediction time (validation set, 2015 data): 0.025 seconds

Validation set MSE: 25.57

Validation MAE (mean absolute error): 4.21

Validation median absolute error: 3.84

Validation R^2 : 0.324

My test set MSE is 26.1280945874. Guy's validation set MSE from a few weeks earlier using Linear Parameter model is 25.57. They are very similar.

My training time for GP model is 0.3921 seconds while Guy's train time for the linear parameter model is 0.35 seconds. Again very very similar.

