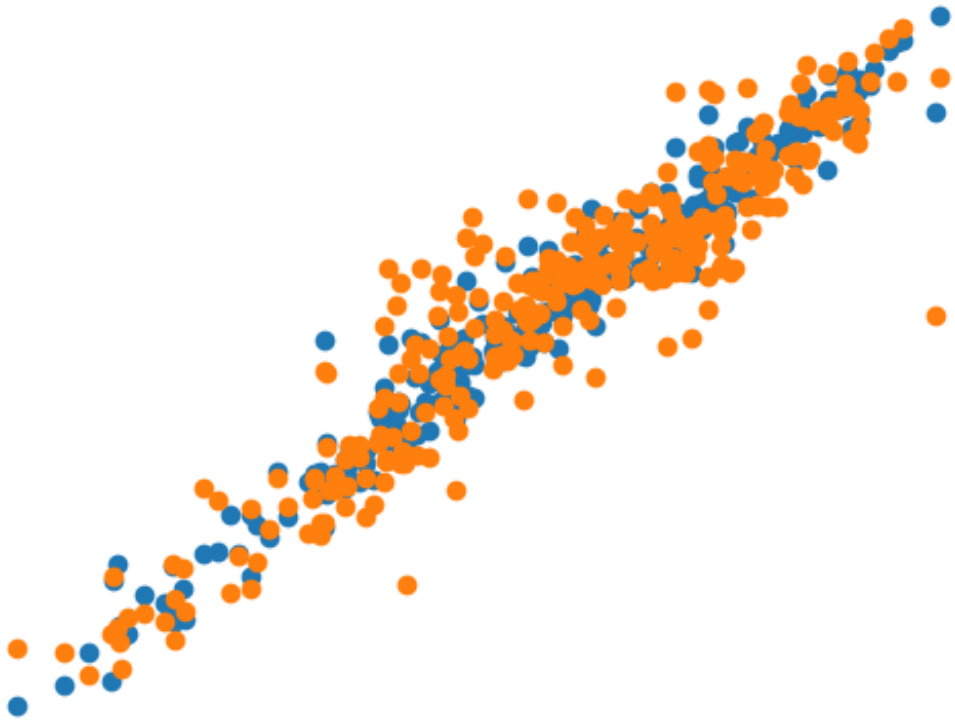
Linear Regressions and Posterior Gaussian Processes

Axel Orrhede

October 2024



1 Comparison of polynomial regression and kernel ridge regression

In this task, we are going to predict Scaled sound pressure levels, in decibels based on a NASA data set called Airfoil Self-Noise[1]. The 1503 data points were obtained from aerodynamic and acoustic tests of two and three-dimensional airfoil blade sections in an anechoic wind tunnel. The data is presented as a pair plot in figure 1.

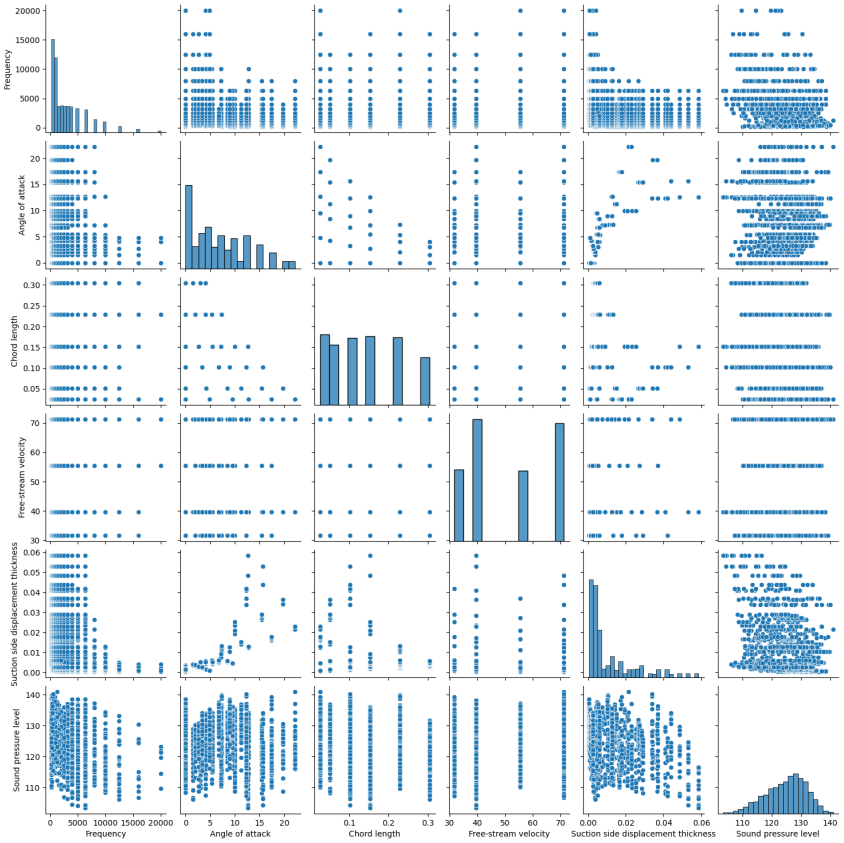


Figure 1: A pair plot of the different features to show their correlations.

From figure 1 we find that there seem to be no obvious correlations between sound pressure and the other features, and we will therefore use two of the more complex linear regression models, one based on kernels and one based on polynomials. We will tune them slightly and then compare their performances and try to explain why they perform as they do.

The data is then normalized and split into 1202 training data points and 301 data

points for testing.

1.1 Polynomial Regression

In testing, the unregularized polynomial model with degree 4 turns out to have the best performance with an MSE of 0.0056, while any polynomial over the degree 7 overfits because it has 1288 parameters according to equation 1 fitting to the 1202 data points in the training set.

$$\text{Number of parameters} = \binom{n+d}{d} + 1 = \frac{(n+d)!}{n!d!} + 1 \qquad [3] \tag{1}$$

If we regularize this problem by adding an $L2$ coefficient with the value of 2.78e-06 and using a polynomial of degree 5 instead (both values found by a grid search), we can get the MSE down to 0.00456.

1.2 Kernel regression

A grid search over the $\nu = 0.5, 1.5, 2.5$ Matérn kernels (choice motivated in Appendix) and the RBF kernel found that the optimal values for this task without regularization were $\sigma = 0.0599$ and $\nu = 0.5$, resulting in an MSE of 0.0013. Which is way better than even the regularized version of polynomial regression. However, one must note that finding the optimal kernel settings is a longer and more finicky process. Even the simple grid search made here with 4 kernels and 10 different values for σ took 34 minutes, which is more than one order of magnitude more than the optimization used for the regularized polynomial kernels.

1.3 Performance comparison

In figure 2 we can see that the optimized kernel significantly outperforms the regularized regression model. It is difficult to explain why this is the case, but we can make a few educated guesses by looking back at figure 1. There are no apparent polynomial relations between any of the features and the sound pressure level, we instead have to hope that the interactions between features allows us to predict the sound pressure level. However and those are not apparent either. Further on, we also note that many features are discretely distributed in only a few values, making it hard to fit a polynomial of a higher degree.[2]

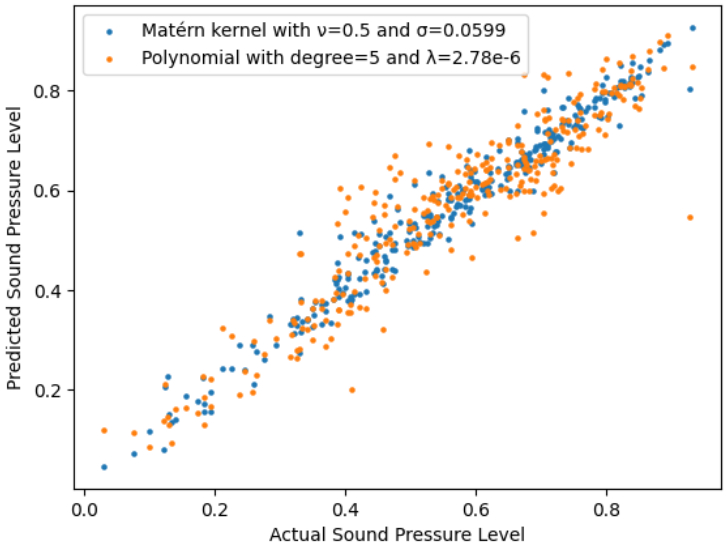


Figure 2: The model predictions over the target values, points resembling a line mean good models and low MSE while dots off the diagonal line are faulty predictions.

Kernel models, on the other hand, who thrive in complex relations, should have an easier time fitting to the data. There is also something to be said about finding $\nu = 0.5$ to be the optimal value, this means that the underlying relations are probably not that smooth, and therefore a better fit for this kernel. Another note is that I lack prior knowledge about the data, allowing me to fully leverage the non-assumptive nature of the kernel method.[2]

Seeing as we only had 1202 data points in the training set, kernel methods also seemed feasible. If we were to have more data points, kernel methods could become too computationally expensive, making this a task for smaller regression models if speed is of the essence or larger neural networks optimized with gradient descent if precision is more important.

2 Calculating the distribution of Gaussian processes

2.1 Brownian motion

Brownian motion is defined by a mean function, that is equal to zero and a covariance function defined by the minimum.

$$K(t, t') = \min\{t, t'\} \tag{2}$$

If we then observe the value 0 at $t = 0$ and x_1 at $t = 1$, we can update our process to fit this information. We start by constructing the covariance matrix, its pseudo

inverse, and two vectors with the covariance function.

$$\mathbf{K} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \qquad \mathbf{K}^{+} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \qquad \mathbf{c} = \begin{bmatrix} \min(0,t) \\ \min(1,t) \end{bmatrix} \qquad \mathbf{c}' = \begin{bmatrix} \min(0,t') \\ \min(1,t') \end{bmatrix} \tag{3}$$

This can be simplified further since t is non-negative. The formula for posterior predictive distributions allows us to calculate the mean and covariance functions for the posterior process.

$$m'(x) = \mathbf{c}^T \mathbf{K}^{+} \begin{bmatrix} 0 \\ x_1 \end{bmatrix} \text{ and } \kappa'(t,t') = \kappa(t,t') - \mathbf{c}^T \mathbf{K}^{+} \mathbf{c}' [4] \tag{4}$$

And we get the following posterior mean and covariance functions.

$$m'(t) = x_1 \min(1,t) \text{ and } \kappa'(t,t') = \min\left(t,t'\right) - \min(1,t)\min\left(1,t'\right) \tag{5}$$

The distribution of a posterior Gaussian process at any given time t will still be Gaussian, meaning that we can describe the distribution D as.

$$D(t) \sim \mathcal{N}(\min(1,t), t - \min(1,t)^2) \tag{6}$$

The posterior distribution is simulated in figure 3

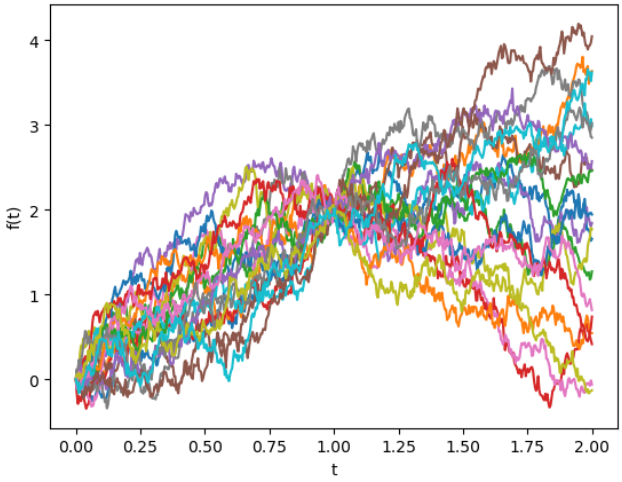


Figure 3: 20 simulated posterior Brownian motions with $x_1 = 2$

2.2 The Ornstein-Uhlenbeck process

The Ornstein-Uhlenbeck process is also fully specified by its mean and covariance function. Again, the mean function will be 0 in the prior distribution, but the covariance will now instead be:

$$K(t,t') = e^{-|t-t'|} \tag{7}$$

The observations remain the same, except that it now takes the value of x_0 when $t = 0$. The same process for calculating the posterior distribution is repeated, this time with the following matrices:

$$\mathbf{K} = \begin{bmatrix} 1 & e^{-1} \\ e^{-1} & 1 \end{bmatrix} \qquad \mathbf{K}^+ = \begin{bmatrix} -\frac{e^2}{1-e^2} & -\frac{1}{2\sinh(1)} \\ -\frac{1}{2\sinh(1)} & -\frac{e^2}{1-e^2} \end{bmatrix} \qquad \mathbf{c} = \begin{bmatrix} e^{-t} \\ e^{-|t-1|} \end{bmatrix} \qquad \mathbf{c}' = \begin{bmatrix} e^{-t'} \\ e^{-|t'-1|} \end{bmatrix} \tag{8}$$

Once again we calculate

$$m'(x) = \mathbf{c}^T \mathbf{K}^+ \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} \text{ and } \kappa'(t,t') = \kappa(t,t') - \mathbf{c}^T \mathbf{K}^+ \mathbf{c}' \tag{9}$$

Getting the following results

$$m'(t) = x_0 \left(-\frac{e^{-|t-1|}}{2\sinh(1)} - \frac{e^2 e^{-t}}{1-e^2} \right) + x_1 \left(-\frac{e^2 e^{-|t-1|}}{1-e^2} - \frac{e^{-t}}{2\sinh(1)} \right) \tag{10}$$

$$\kappa'(t,t') = -\left(-\frac{e^{-|t-1|}}{2\sinh(1)} - \frac{e^2 e^{-t}}{1-e^2} \right) e^{-t'} - \left(-\frac{e^2 e^{-|t-1|}}{1-e^2} - \frac{e^{-t}}{2\sinh(1)} \right) e^{-|t'-1|} + e^{-|t-t'|} \tag{11}$$

The distribution D at a time t will be as follows.

$$D(t) \sim \mathcal{N}(m'(t), \kappa'(t,t)) \tag{12}$$

Our posterior process is simulated in figure 4.

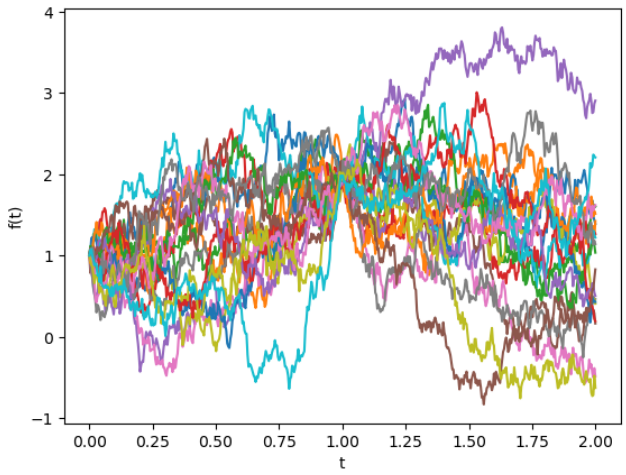


Figure 4: 20 simulated posterior Ornstein-Uhlenbeck processes with $x_0 = 1$ and $x_1 = 2$ is

References

[1] T. Brooks, D. Pope, and M. Marcolini. Airfoil self-noise, 2014.

[2] Nur'eni, M. Fajri, and S. Astuti. Comparison of kernel regression model with a polynomial regression model on financial data. *Journal of Physics: Conference Series*, 1763(1):012017, 2021.

[3] R. Webber. Lecture 3 in math216a, 2024.

[4] R. Webber. Lecture 6-7 in math216a, 2024.

[5] Wikipedia. Bertil matérn (the article is way more comprehensive in swedish), 2024.

3 Appendix

3.1 Why choose the Matérn kernel?

You should obviously always choose the Matérn kernel because of its Swedish Heritage! [5]

Figure 5: För kung och fosterland