ML–DS I: Project V Report

Sebastijan Trojer
ML-DS I 2024/25 , FRI, UL
st5804@student.uni-lj.si

I. Introduction

In this assignment we implemented kernelized ridge regression and support vector regression. We implemented both polynomial and RBF (Gaussian) kernels and compared the advantages of using each. We applied the implemented methods on both one-dimensional sinusoidal data and multidimensional housing dataset.

II. Methodology and results

A. Kernelized Ridge Regression

First we implemented the kernelized ridge regression, using both kernels. We did so by solving the closed-form equation $\alpha = (K + \lambda I)^{-1} y$, where $K = kernel(X, X)$. Before fitting the model, the data was standardized. Figure 1 shows the original and the predicted data. Notice that the model was able to fit to the data relatively well, however the degree of the polynomial had to be very high. Similarly, Figure 2 shows how the model fit to that same data using the RBF kernel. In this case, the fit is arguably a bit better.

B. Support Vector Regression

We also implemented support vector regression (SVR). For this, we had to compute 6 matrices, so we could use a KKT solver. Since the output of the function had to be in a required format, we computed them regularly and then permuted them.

$$P = \begin{bmatrix} K_n & -K_n \\ -K_n & K_n \end{bmatrix}, \quad q = \begin{bmatrix} (\epsilon - y)_n & (\epsilon + y)_n \end{bmatrix}$$

$$G = \begin{bmatrix} -I_{2n} \\ I_{2n} \end{bmatrix}, \quad h = \begin{bmatrix} 0_{2n} & C_{2n} \end{bmatrix}$$

$$A = \begin{bmatrix} I_n & -I_n \end{bmatrix}, \quad b = 0$$

and $C = \frac{1}{\lambda}$

As mentioned, the matrices listed above would return the result $x = [\alpha_1, \alpha_2, ..., \alpha_1^*, \alpha_2^*, ...]$ so the matrices were permuted before being passed to the solver. Figures 1 and 2 show the model predictions on the data using polynomial and RBF kernel respectively. The support vectors are shown in green. Both ridge regression and SVR performed well on the sine data and parameter fitting was easier when using a RBF kernel in both cases.

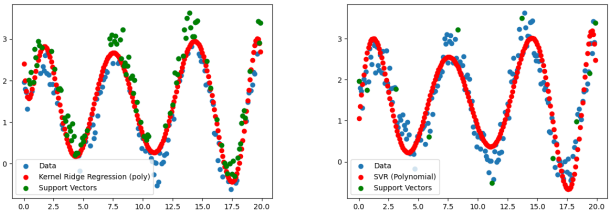

Fig. 1. Ridge regression and SVR with a polynomial kernel with degree 11 and λ 0.001 and 0.00001 respectively.

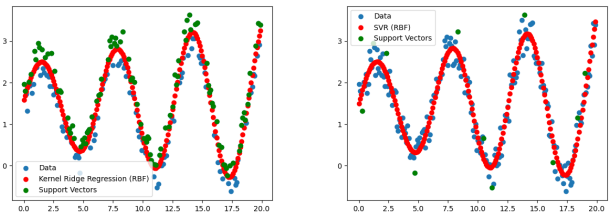

Fig. 2. Ridge regression and SVR with RBF kernel with $\sigma = 0.5$ and $\epsilon = 0.6$ for the SVR. λ was the same as in Figure 1

C. Regression on housing dataset

We evaluated both models with both kernels on a housing dataset. For the kernels, we evaluated multiple kernel parameter values, namely the degree for the polynomial and σ for the RBF. However we did this in two different ways – one model was evaluated with λ fixed to 1 and another had λ selected using nested cross validation (CV). Both inner and outer CV used 10 folds. Figure 3 shows the error against the polynomial kernel of various degrees for both fixed and optimized λ. Similarly, Figure 4 shows the same curves, but for the RBF kernel and different values of σ. ϵ was fixed to 0.5, since that is a loose enough margin that allows us to have a simpler model, that can still perform well, while keeping the solution sparse. The variance shown on error bars was estimated using asymptotic normality, meaning we computed the SE over the splits and divided by $\sqrt{n}$ where n is the number of splits. The main takeaway from these figures is that the models with empirically selected λ performed better, however Table I shows that such models did not necessarily produce more sparse models. The reason for that is that optimizing λ was done in a way that optimized model performance, which often means producing more complex models, therefore the solution is

less sparse. Such an observation can be made especially in the case of the RBF kernel.

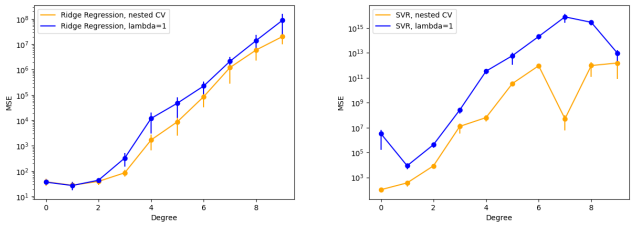

Fig. 3. Ridge regression and SVR with polynomial kernel on the housing dataset. The blue line shows the mean MSE with respect to the degree of the polynomial in 10-fold CV and λ set to 1, and the orange line a similar score, with the exception that the λ was chosen using nested CV.

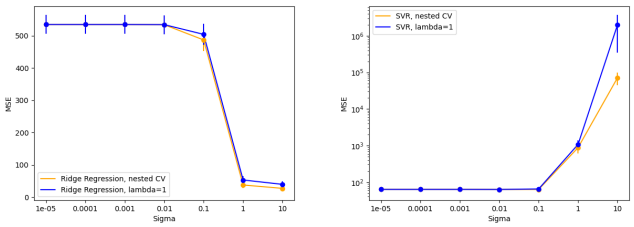

Fig. 4. Ridge regression and SVR with RBF kernel on the housing dataset. The experiment was the same as the one described on Figure 3

Another interesting thing to note when looking at the results of I is that as the kernel parameter increased, the amount of support vectors grew. For example, the SVR with polynomial kernel and fixed and non-fixed λ had 22 and 8 support vectors at degree 2, and 131 and 169 support vectors at degree 10, respectively. Similarly for the RBF kernel, the models had around 25 and 60 support vectors at lower σ values and up to 100 at large σ values. This indicates that model performance and model sparsity are not necessarily directly related, in a sense, that a more sparse model does not necessarily mean a better model, and that it depends on the use-case. The lowest MSE was achieved by ridge regression with empirical λ and RBF kernel, with MSE of 27.

Model	Kernel	(Avg.) λ	Avg. support vectors
SVR	Polynomial	1	98
	Polynomial	6.33	92
SVR	RBF	1	57
	RBF	0.676	99

TABLE I
NUMBER OF SUPPORT VECTORS IN EACH MODEL ACROSS DIFFERENT KERNEL PARAMETERS. THE THRESHOLD FOR SUPPORT VECTORS WAS SET TO 1×10^{-5}. NOTABLY, THE AMOUNT OF SUPPORT VECTORS INCREASED WITH INCREASED DEGREE OF POLYNOMIAL AND σ.

Overall both models are viable for real-life use case, however we prefer the kernelized ridge regression, due to its simplicity and ability to perform comparably to an SVR, as discussed above. SVR is still a good model, however it is less intuitive, namely its implementation requires solving a Lagrange problem, compared to ridge regression, the unkernelized version of which is well known and also has a closed-form solution. The biggest advantage SVR has over ridge regression is that it offers a sparse solution, therefore inference can be faster. Regarding the kernels, both polynomial and RBF kernel are viable choices, however the polynomial kernel tends to overfit faster while RBF appears more robust. Similarly, setting the σ parameter seems more intuitive (smoothness of fit, data point influence range) than setting the degree of polynomial (higher degree = ability to learn more complex data, but how to quantify data complexity), where we might also have to deal with extremely large values at high degrees.

III. Conclusion

We implemented kernelized ridge regression and support vector regression, using 2 different kernels. We evaluated all approaches and compared them, outlining the advantages of each and discussed their performance in different scenarios and ease-of-use. Overall we choose kernelized ridge regression as the better model, since it has a closed-form solution and shorter fitting time, while providing similar performance to the support vector regression.