# **Support Vector Machine Regressor**

Krishna Kumar Veeraputhiran

Grand Canyon University

DSC 540 – O500 – Machine Learning for Data Science

Dr. Aiman Darwiche

September 1,2021

# **Support Vector Machine Regressor**

Support vector machines are widely used for classification problem in Machine Learning, but one of the important functionality of SVM is for solving Regression problem too. In this report we will see more on the functionality of the regression and its potential uses.

**Regression**

Support Vector Regression is a type of SVM that supports both Linear and Non Linear regression. With Support vector Regression, the goal is to find a function that approximates mapping from an input domain to real numbers on the basis of the training data. With respect to SVR, the main objective is to define the decision boundary and identify the hyperplane and identify all data points that are within the decision boundary margin. The line of best fit here is the hyperplane, which has maximum number of points. The steps to build a SVR can be shown below,

1. Collect a training data {X,y}
2. Choose the SVR parameters like Kernel, Regularization parameter and the epsilon (which is the distance between the decision boundary and the hyperplane).
3. Form the correlation matrix which can be shown below,

**Kij = exp(∑k θk|xki – xkj|2) + εδij**

1. The machine algorithm trains to get the contraction coefficient using both the correlation matrix and the input data.

̂**a= K̅ -1 ̅y**

1. Using the above coefficient an estimator is created, which is then used to predict the output.

**Parameters Used**

A SVM regression can be created in Python using the Scikit library using the svr package which is under svm. The most important parameters used in the SVM regression in Python programming are shown below.

*Kernel*: This specifies the kernel type that needs to be used in the algorithm. For a linear regression we can use the ‘linear’ kernel. The default is ‘rbf’ (Radial basis function). The other kernels are poly, sigmoid etc.

*Regularization Parameter (C)* : The regularization parameter serves as a degree of importance that is given to misclassification. When the value of C increases the rate of misclassification falls down.

*Gamma*: This parameter defines how far the influence of a single training example reaches, with low gamma value means a far reach and a high means low reach. It can be seen as the inverse of the radius of influence of samples selected as support vectors.

*Epsilon*: This defines the epsilon tube (Decision boundary). This defines the margin of tolerance within which no penalty is given for errors.

**Regressor Accuracy**

The support vector machine regression algorithm can be tested for accuracy on how well the regression has been made. The accuracy of the SVM regression can be obtained from the R-squared value of the model. The R-squared can be obtained by calculating 1- (u/v), where u is the residual sum of squares and v is the total sum of squares. The best possible score that a model can get is 1 and the worst being 0. In python programming the accuracy score can be calculated using the *score* function available in the SVR package. In the model (with the Gaussian noise) that has been calculated for this weeks exercise we can see the accuracy being calculated as 74.7%. The model that has been used is to predict the sinc function for fifty data points added with Gaussian noise.

As Nelson et al. (2008) suggests, the choice of Kernel is crucial when trying to tackle a non-linear function like sinc function. According to the author there can be any sort of Kernels used to tackle these problems and may yield multiple solutions. This is where understanding the relationship between the data space and the Kernel space becomes crucial. Considering that we have selected the standard rbf kernel for our non-linear sinc function. The author illustrate how a dedicated sinc kernel can be used to tackle the signal theory.

One of the biggest advantages of the SVM regressor model is that it is a robust and immune to outliers. Also the decision model can be easily updated and has an excellent generalization capability with higher prediction accuracy.

# **References**

Gopal, M. (2019). *Applied machine learning*. McGraw-Hill Education.

Nelson, J. D. B., Damper, R. I., Gunn, S. R., & Guo, B. (2009). A signal theory approach to support vector classification: The sinc kernel. *Neural Networks*, *22*(1), 49–57. https://doi.org/10.1016/j.neunet.2008.09.016

[Scikit-learn: Machine Learning in Python](http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html), Pedregosa *et al.*, JMLR 12, pp. 2825-2830, 2011

Raj, A. (2020, October 3). *Unlocking the true power of support vector regression*. Medium. https://towardsdatascience.com/unlocking-the-true-power-of-support-vector-regression-847fd123a4a0.

Sharif. (n.d.). *Support vector regression made easy(with python code)*. www.aionlinecourse.com. https://www.aionlinecourse.com/tutorial/machine-learning/support-vector-regression.