ELL409 Assignment 3

Support Vector Regression

Isha Chaudhary

2018EE30614

1 Dataset:

- **Name**: Boston Housing Price dataset
- **URL**: http://lib.stat.cmu.edu/datasets/boston
- **Number of examples:** 506
- **Number of Features:** 13
- **Features:**
  1. CRIM: per capita crime rate by town
  2. ZN: proportion of residential land zoned for lots over 25,000 sq.ft.
  3. INDUS: proportion of non-retail business acres per town
  4. CHAS: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
  5. NOX: nitric oxides concentration (parts per 10 million)
  6. RM: average number of rooms per dwelling
  7. AGE: proportion of owner-occupied units built prior to 1940
  8. DIS: weighted distances to five Boston employment centres
  9. RAD: index of accessibility to radial highways
  10. TAX: full-value property-tax rate per $10,000
  11. PTRATIO: pupil-teacher ratio by town
  12. B: $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town
  13. LSTAT: lower status of the population
- **Target variable:** MEDV - Median value of owner-occupied homes in $1000's

2 ϵ SVR Using CVXOPT library

2.1 Procedure

- The data was imported and the features were standardized.
- Function Compute_kernel() is used to find the kernel matrix for the input given the type of kernel demanded.
  1. **RBF Kernel:** $k(x_1, x_2) = exp^{-\gamma||(x_1-x_2)||^2}$
  2. **Polynomial kernel**: $k(x_1, x_2) = (1 + x_1 x_2')^p$
  3. **Linear kernel:** $k(x_1, x_2) = x_1 x_2'$
- The optimization (Primal) involved here is:

$$\min_{W,\xi,\hat{\xi},b} C\sum_{n=1}^{N}(\xi_n + \hat{\xi_n}) + \frac{1}{2}||W||^2$$

subject to

$$\xi_n \geq 0;\ \hat{\xi_n} \geq 0$$
$$y_n \leq f(X_n) + \epsilon + \xi_n$$
$$y_n \geq f(X_n) - \epsilon - \hat{\xi_n}$$

- Dual problem:

$$\min_{a,\hat{a}} \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}(a_n - \hat{a_n})(a_m - \hat{a_m})k(X_n, X_m) - \epsilon\sum_{n=1}^{N}(a_n + \hat{a_n}) + \sum_{n=1}^{N}(a_n - \hat{a_n}y_n)$$

subject to

$$0 \leq a_n \leq C$$
$$0 \leq \hat{a_n} \leq C$$

- $f(X) = \sum_{n=1}^{N}(a_n^* - \hat{a_n}^*)k(X, X_n) + b$
- $b = y_n - \epsilon - \sum_{m=1}^{N}(a_m^* - \hat{a_m}^*)k(X_n, X_m)$; where $a_n^*, \hat{a_n}^*$ form the optimal solution of the dual problem. Here n is such that $0 < a_n < C$ *or* $0 < \hat{a_n} < C$ (in the cvxopt implementation, due to the infinite precision of floating point numbers the lower bound is kept to be 1e-3)
- To find the optimal solution of the dual problem using CVXOPT, it is formulated in the form:

$$\min_{x} \frac{1}{2}x^T P x + q^T x$$

subject to

$$Gx \leq h$$
$$Ax = b$$

- Here $x = \binom{a_n}{\hat{a_n}}$,

$$P = \begin{pmatrix} I_n \\ -I_n \end{pmatrix} k(X,X) \begin{pmatrix} I_n & -I_n \end{pmatrix},\ q = \begin{pmatrix} \epsilon - y_1 \\ ... \\ \epsilon - y_n \\ \epsilon + y_1 \\ ... \\ \epsilon + y_n \end{pmatrix}$$

$$G = \begin{pmatrix} I_{2n} \\ -I_{2n} \end{pmatrix},\ h = \begin{pmatrix} C \\ .. \\ (2n\ entries) \\ .. \\ C \\ 0 \\ .. \\ (2n\ entries) \\ .. \\ 0 \end{pmatrix}$$

$A = \begin{pmatrix} 1 & .. & (n\ entries) & .. & 1 & -1 & .. & (n\ entries) & .. & -1 \end{pmatrix}$, $b = [0.0]$
where I_n is the identity matrix with n rows.

- These matrices are fed into the qp solver as cvxopt.solvers.qp(P, q, G, h, A, b) and the optimal values for a_n, $\hat{a_n}$ are found out. Following this the regression function is formulated as that given in the equations above to begin off with the prediction.

- The Correlation of the output with the labels, both for the training and test sets is found out using the R2 score. The Mean square error values are also calculated for further insight.

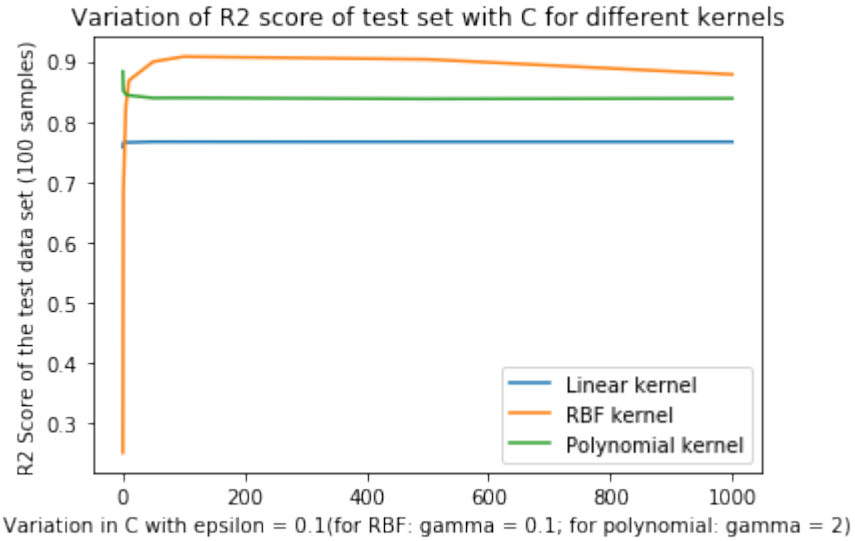
- An implementation of the Grid Search Cross Validation was also tried out. In this algorithm, a set of best hyperparameters for the model are selected by running k-fold cross validation over all combinations of the hyperparameter values input input into the function. The set of hyperparameter values which give the least mean square error are selected. In this implementation, only one set of hyperparameter values can be specified for the kernel function(s), under the name, gamma.

2.2 Results:

(test set consists of 100 examples, training on remaining 406 examples)

2.2.1 Observing the effects of Hyperparameter values on the R2 score of the test data for various kernel functions

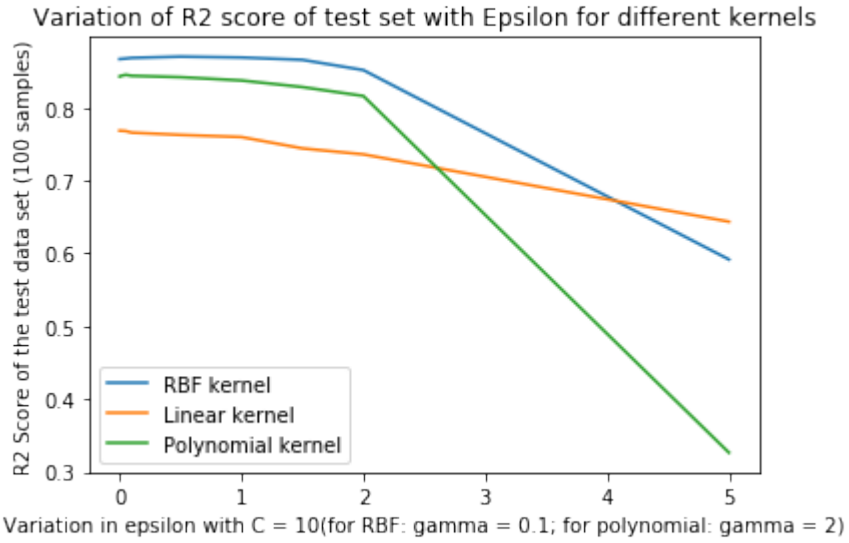
- The variation of the R2 score of test set with C for various kernel functions is depicted in the figure.



The key observations are:

1. The R2 score of the RBF kernel function is higher for higher values of C, as compared to the other kernel functions. So using the RBF kernel appears to be a better choice.
2. For Polynomial and linear kernels, the R2 score does not change much with very high values of C.
3. For very low values of C (tending to 0), the performance of the RBF kernel is not so good as compared to the other kernels.
4. The R2 score of the RBF kernel function drops for higher values of C, as the ϵ tube becomes harder.

- The variation of the R2 score of test set with Epsilon for various kernel functions is depicted in the figure.

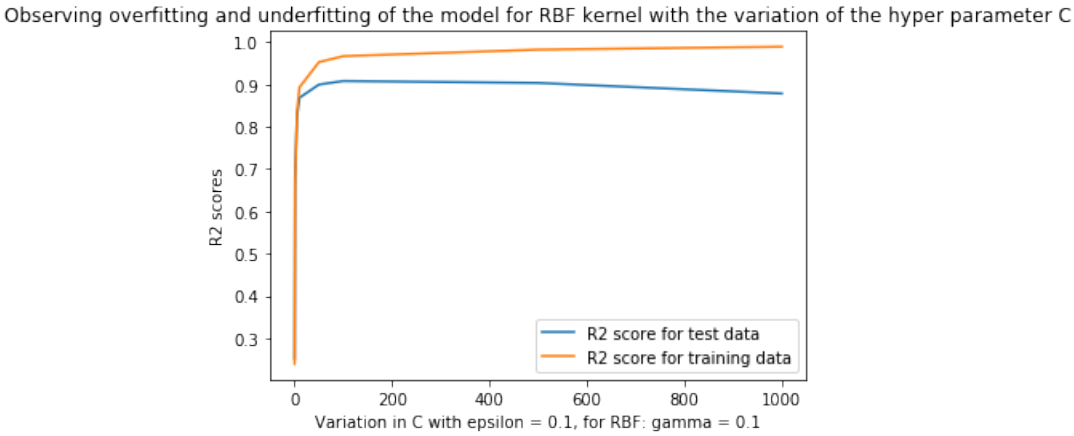


The key observations are:

1. The R2 score for RBF kernel function stays quite high as compared to the other kernels for the majority of the range. As the ϵ tube gets wider, linear kernel takes over and has the highest R2 score.
2. The R2 score falls for all the kernel functions as the ϵ tube grows wider as we are allowing for a lower accuracy for the fitting model.
3. So the best correlation between the fitted model and the actual target function is seen to be achieved with a smaller value of ϵ, upto a certain limit.

2.2.2 Observing Overfitting and Underfitting by the variation of hyperparameters

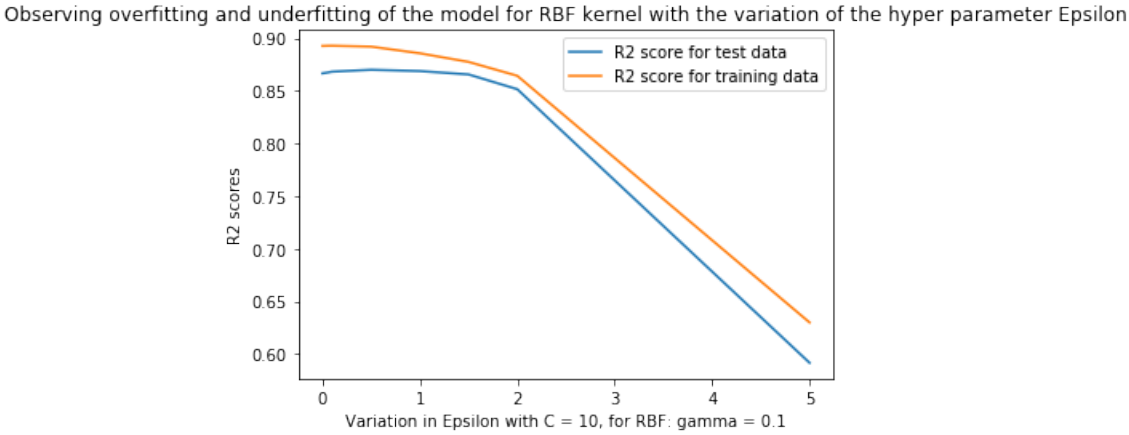
- Variation of training and test data R2 scores for the values of C for RBF kernel.



The key observations are:

1. The model begins to slightly overfit the training data for larger values of C.
2. For very small values of C, the model underfits the data. (Although not so apparent due to the large range of the parameter C, for lower values of C, both the training and test R2 scores are small.)

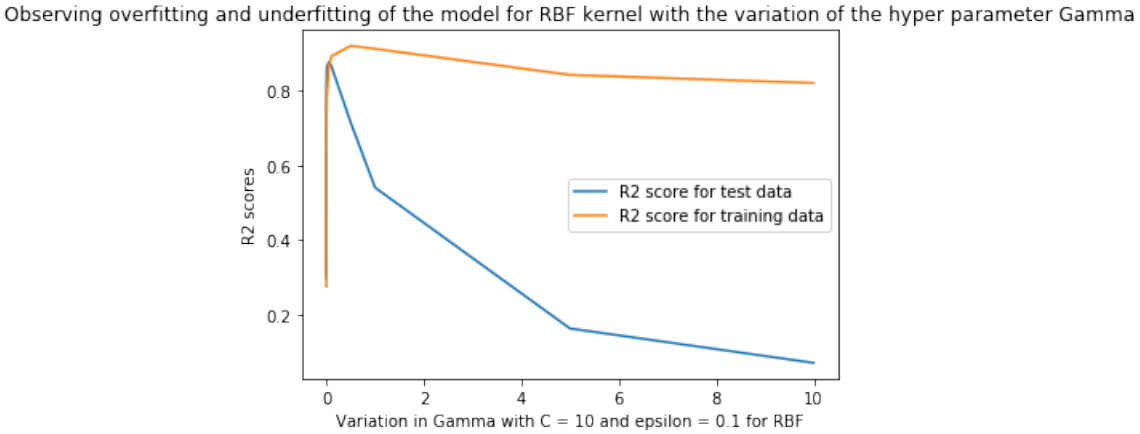
- Variation of training and test data R2 scores for the values of Epsilon for RBF kernel.



The key observations are:

1. The model underfits the data for large values of ϵ, as both the training and test R2 scores have become very low. The regression function is not getting learnt when we allow for a very wide ϵ tube.
2. No strong tendency of overfitting is observed. Only for small values of ϵ, the training score exceeds the test score by the largest amount.

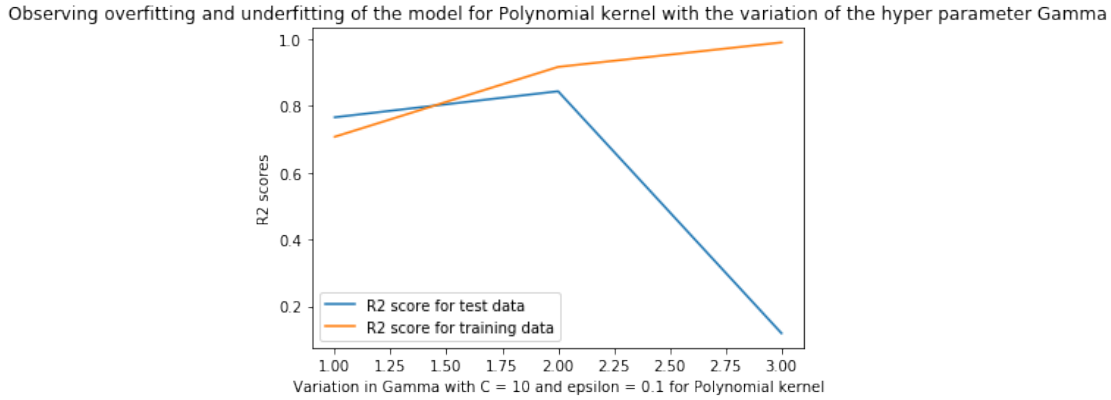
- Variation of training and test data R2 scores for the values of Gamma for RBF kernel.



The key observations are:

1. For higher values of γ, the model overfits the data, as the test R2 score becomes quite small.
2. It appears that the model underfits the data for very small values of γ.

- Variation of training and test data R2 scores for the values of Gamma for Polynomial kernel.



The key observations are:

1. The model has overfit the training data at $\gamma = 3$.
2. The most suitable value for γ in this case appears to be 2.

- Results of the self-implemented GridCV() function for ϵ SVR are: 'C': 10.0, 'epsilon': 0.1, 'kernel': 'rbf', 'gamma': 0.1

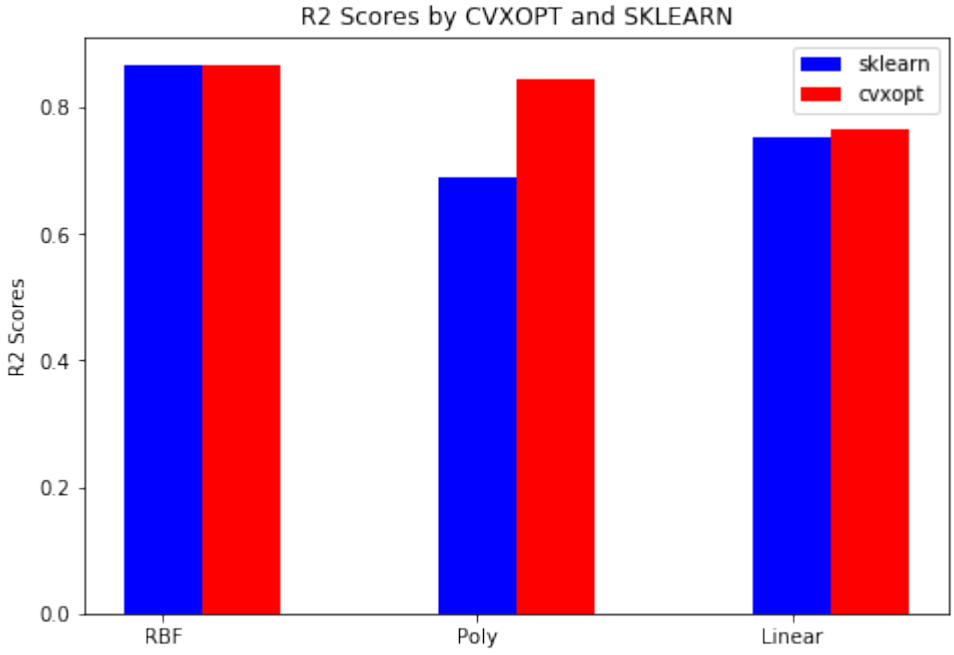
3 ϵ SVR implementation using sklearn library

3.1 Procedure:

- The data is input and the features are standardized.
- From sklearn.model_selection, GridSearchCV() function is used to obtain the optimal values for the hyperparameters C, ϵ, kernel function to be used and the constants used in the kernel function. This optimization is performed using the negative mean squared loss function
- From sklearn.svm, SVR class is imported and the model formed by the optimal hyperparameters is fit over the training data. The training data is 80% of the actual data set (406 examples).
- The R2 score and the mean squared loss are found out for the training and test data and the number of support vectors formed are also investigated.

3.2 Results:

The test set (100 examples) R2 scores of the implementations using CVXOPT and sklearn are compared in the following bar graph for different kernel functions (Poly stands for the Polynomial kernel function):



The values for the hyperparameters used are:
C = 10, $\epsilon = 0.1$, γ (for RBF kernel) = 0.1, degree of the polynomial kernel = 2.
Thus we can conclude that the implementation using *CVXOPT* gives better

performance for all kernels as compared to the implementation using *SCIKIT LEARN*, with respect to the test set R2 scores.

4 Bonus problem: RH-SVR using the CVXOPT library

4.1 Procedure:

- The data was imported and the features were standardized.
- Function Compute_RBF_kernel_matrix() is used to find the kernel matrix for the input given the type of kernel demanded.
- Dual problem: (reference: A geometric approach to support vector regression)

$$\min_{u,v} \frac{1}{2}(u-v)'(K+y.y')(u-v) - 2\epsilon y'(u-v)$$

subject to

$$\sum_{i=1}^{l} u_i = 1 = \sum_{i=1}^{l} v_i$$

$$0 \leq u \leq D$$

$$0 \leq v \leq D$$

where K is the kernel matrix of shape (no. of features of X_input, no. of features of X_input)

- $f(X) = \sum_{i=1}^{l}(\bar{v_i} - \bar{u_i})k(X_i, X) + \bar{b}$
- Here, the optimal solution of the above dual problem is $(\hat{u}, \hat{v})$.
- $\hat{\delta} = (\hat{u} - \hat{v})'y + 2\epsilon$
  $\bar{u} = \hat{u}/\hat{\delta}$
  $\bar{v} = \hat{v}/\hat{\delta}$
- $\bar{b} = (\hat{u} - \hat{v})'K(\hat{u} + \hat{v})/2\hat{\delta} + (\hat{u} + \hat{v})'y/2$, where K is the Kernel matrix.
- To find the optimal solution of the dual problem, it is formulated in the form:

$$\min_{x} \frac{1}{2}x^T Px + q^T x$$

subject to

$$Gx \leq h$$

$$Ax = b$$

- Here $x = \binom{u}{v}$,
  $P = \binom{I_n}{-I_n}(K + y.y')\begin{pmatrix} I_n & -I_n \end{pmatrix},\ q = 2\epsilon\binom{I_n}{-I_n}y'$
  $G = \binom{-I_{2n}}{I_{2n}},\ h = \begin{pmatrix} 0 \\ .. \\ (2n\ entries) \\ .. \\ 0 \\ D \\ .. \\ (2n\ entries) \\ .. \\ D \end{pmatrix}$
  $A = \begin{pmatrix} 1 & .. & (n\ entries) & .. & 1 & 0 & .. & (n\ entries) & .. & 0 \\ 0 & .. & (n\ entries) & .. & 0 & 1 & .. & (n\ entries) & .. & 1 \end{pmatrix},\ b = \binom{1.0}{1.0}$
  where I_n is the identity matrix with n rows.

- These matrices are fed into the qp solver as cvxopt.solvers.qp(P, q, G, h, A, b) and the optimal values for $a_n,\ \hat{a_n}$ are found out. Following this the regression function is formulated as that given in the equations above to begin off with the prediction.

- The Correlation of the output with the labels, both for the training and test sets is found out using the R2 score. The Mean square error values are also calculated for further insight.

- Currently, this implementation is made only for RBF kernel.

4.2 Results:

R2 Score on test data (100 examples): 0.851,
R2 Score on training data (406 examples): 0.911
with D = 10, $\epsilon = 0.001$, $\gamma = 0.0001$
This was the best result obtained after trying several combinations of model hyperparameters.