Hindawi Publishing Corporation

Science and Technology of Nuclear Installations
Volume 2008 (2008), Article ID 695153, 6 pages
doi:10.1155/2008/695153

Research Article

Machine Learning of the Reactor Core Lo Pattern Critical Parameters

Krešimir Trontl,[1] Dubravko Pevec,[1] and Tomislav Šmuc[2]

[1]Department of Applied Physics, Faculty of Electrical Engineering a 3, 10000 Zagreb, Croatia
[2]Division of Electronics, Ruđer Bošković Institute, Bijenička 54, 10(

Received 11 March 2008; Accepted 23 June 2008

Academic Editor: Igor Jencic

Abstract

The usual approach to loading pattern optimization involves high rules, an optimization algorithm, and a computer code used for e the optimization process is highly dependent on the computer investigate the applicability of a machine learning model which co employ a recently introduced machine learning technique, suppor kernel based, nonlinear modeling paradigm, in which model para quadratic optimization problem. The main objective of the work re of applying SVR method for reactor core loading pattern modeling discuss its applicability, that is, complexity, speed, and accuracy.

1. Introduction

Decrease of the fuel cycle costs is an important factor in nuclear p cycle can strongly benefit from the optimization of the reactor amount of enriched uranium and burnable absorbers placed in operational and safety characteristics.

The usual approach to loading pattern optimization involves high
rules, an optimization algorithm, and a reactor physics computer c
Since the loading pattern optimization problem is of combinatoı
numbers of core modeling calculations (e.g., genetic algorithms oı
for one full optimization run is essentially determined by the com
pattern.

The aim of the work reported in this paper was to investigate the a
loading pattern evaluation. We employed a recently introduced ma
(SVR), which has a strong theoretical background in statistical lea
in which model parameters are automatically determined by solvin

This paper reports on the possibility of applying SVR method for re
of the learning data set, as a function of targeted accuracy, influer
definition were studied.

In Section 2, the support vector regression method is discussec
optimization as well as the methodology applied for the investiç
loading pattern evaluation are presented in Section 3. Results and
5 the conclusions based on this work are drawn.

2. Support Vector Regression

Machine learning is, by its definition, a study of computer algorith
One of machine learning techniques is the support vector machin
background in statistical learning theory [1]. The method pro
classification and regression problems. Although, historically sp
classification problems [2, 3], in the last decade, the applicati
noticeable in different fields of science and technology [4 – 10
generalization properties of the method.

In the upcoming paragraphs, we will give a short introduction int
only the most important theoretical and practical aspects of the
referenced literature.

In general, the starting point of the machine learning problem is a
model (training set) and a separate set to test the learned model
regression model, we will consider a training data set, as well
input/output pairs, representing the experimental relationship bet
output value (y_i):

$$\{(\vec{x}_1, y_1), (\vec{x}_2, y_2), \ldots, (\vec{x}_n, y_n$$

In our case, the input vector defines the characteristics of the load
as a target value, denotes the parameter of interest.

The modeling objective is to find a function $y = f(\vec{x})$ such that it acc
(y) corresponding to a new input vector ($\vec{x}$), yet unseen by the
particular input vector) [11].

Due to the high complexity of underlying physical process that

expected to have high nonlinear properties. In the support vect
mapped into a higher dimensional feature space F using a nonlin
performed in that space. Therefore, a problem of nonlinear regre
linear regression in high-dimensional feature space.

The SVR technique considers the following linear estimation functic

$$f(\vec{x}) = \langle \vec{w}, \Phi(\vec{x}) \rangle +$$

where $\vec{w}$ denotes the weight vector, b is a constant known as bias
dot product in feature space, such that $\Phi : \vec{x} \rightarrow F$, $\vec{w} \in F$ [12]. The
the data points in the training set. To avoid overfitting and m
regularized form of the functional, following principles of structural

$$R_{\mathrm{reg}}[f] = \sum_{i=1}^{M} C\left(f(\vec{x}_i) - y_i\right)$$

where $R_{\mathrm{reg}}[f]$ denotes regression risk (possible test set error), bas
cost function C determined on the points of the training set, and
model. Minimization task thus involves simultaneous minimization
complexity of the model. Most commonly used cost function (loss
"ε insensitive loss function":

$$C(f(\vec{x}_i) - y_i) = \begin{cases} \| f(\vec{x}) - y \| - \varepsilon, & \text{fc} \\ 0, & \text{o} \end{cases}$$

where ε is a parameter representing radius of the tube around r
position the tube around the data, as depicted in Figure 1 [7], and
which calculated values (y) lie inside this tube. The deviations of p
function are penalized in the optimization through their positive
variables.

Figure 1: The schematic illustration of the SVI

It was shown that the following function minimizes the regularized

$$f\left(\vec{x}, \vec{w}\right) = f\left(\vec{x}, \vec{a}, \vec{a}^*\right) = \sum_{i=1}^{n} \left(a_i^* -\right.$$

where $a_i^* a_i$ are Lagrange multipliers describing $\vec{w}$, and are estima
quadratic programming algorithm, and $K(\vec{x}_i, \vec{x})$ is a so called *kernel*
feature space. A number of kernel functions exist [13]. Kernel f
details in the following section.

Due to the character of the quadratic optimization, only some
corresponding input vectors $\vec{x}$ are called *support vectors* (SVs). I
positioned inside the ε tolerance tube and are therefore, not intere
vectors that are determined in the training (optimization) phase a

the information content of the training set. In most of the SVR for
by the user: C-cost of the penalty for data-model deviation, and
the chosen form of the kernel function and its corresponding pa
performance of the regression model.

3. Methodology

One of the key processes of both, safe and economical operations
to be more precise, fuel loading pattern determination and optim
loading pattern determination and optimization tasks, whether
genetic algorithms, or a combination of stated approaches, require
evaluation. The evaluation is normally performed using a more o
such codes is time consuming. Therefore, in this work, we are inv
as a fast tool for loading pattern evaluation.

However, taking into account that the SVR method is to be used,
creating a model. The first is the setting of the loading pattern t
which the experimental data points are to be generated, the def
target values. The second is the choice of the kernel function and a
Finally, SVR modeling tools have to be addressed.

3.1. Computational Experiment Setup

Taking into account the preliminary and inquiring characteristics of
inventory for a single loading pattern optimization as a basis fo
Krško Cycle 22 loading pattern has been used as a reference on
were used for core loading in Cycle 22 have been used for gene
loading patterns, which were then divided into training and
development process. The global core calculations of each of the
MCRAC code of the FUMACS code package, which also includes
preparation [14]. The calculation is based on quarter core symm
concentration curve.

The generation phase, that is, the definition of the loading pattern
order to narrow the investigated input space as much as possible,
of available fuel assemblies per batch, we introduced a limitation fo
it can be placed: fuel assemblies originally placed on axes position
and vice versa. The central location fuel assembly was fixed for eve

The most important issue in the regression model development is
model development. Since in a quarter core symmetry setup, the
and having in mind the inquiring nature of the work, we decided to
core symmetry, resulting in 21 fuel assemblies defining the co
enrichment, number of IFBAs, and reactor history, or at least bur
number of potential parameters defining the input space is 63. Th
increases the number of training points and time required for the
properties. Therefore, we decided to reduce the number of param
cycle as a new parameter and representing fuel assembly only by
64, 92, and 116 for fresh fuel). Thus, the final number of paramete

The SVR model would eventually be used in an optimization algo
Therefore, the target parameters which we want to model should b

evaluation is based. In this work, we used the global core effectiv
end of the cycle (k_{effBOC} and k_{effEOC}), as well as power peak
separate SVR models were built.

3.2. Kernel Functions

The idea of the kernel function is to enable mathematical operatior
high-dimensional feature space [15]. The theory is based upon rep

A number of kernel functions have been proposed in the literature
be used for mapping nonlinear input data into a linear feature sp
representing the problem. It is up to the modeller to select the ap
placed on two widely used kernel functions, namely, radial ba
polynomial function (PF), which are defined by (6)

$$K_{\text{RBF}}\left(\vec{x}_i,\vec{x}_j\right)=\exp\left(\frac{-\|\vec{x}\ldots}{\ldots}\right.$$

$$K_{\text{PF}}\left(\vec{x}_i,\vec{x}_j\right)=\left(\vec{x}_i^T\vec{x}_j\ldots\right.$$

In the case of RBF kernel, parameter σ represents the radius of th
represents the degree of the polynomial kernel.

As already mentioned, the behaviour of the SVR technique strong
its corresponding parameters, and general SVR "free" paramete
were determined by a combination of engineering judgement and
genetic algorithms [17].

3.3. SVR Modeling Tools

Excellent results in SVR application to a wide range of classifica
science and technology, initiated creation of a number of impleme
some of which are freely available software packages. In this wo
SVMTorch [18], LIBSVM [19], and WEKA [20].

As stated in the previous subsection, RBF and PF kernel functions
given in (6). However, practical parameterisation of the functions,
from code to code. For example, parameter g in LIBSVM notatio
comparison of codes has been performed, general kernel param
parameters were modified to reflect on these values.

4. Results and Discussion

4.1. Comparison of Code Packages

The comparison of three code packages for SVR modeling, na
conducted using a maximum training set size of 15 000 data poir
The number of data points for learning models is typically enlarg
are achieved. In this subsection, only the results of final models co

Preliminary analyses revealed that preprocessing of the input data
fast operation of all SVR code packages. Mainly, due to the fact th
scaling of the input data has been performed, including the scaling
one of LIBSVM codes: SVMSCALE.

Models for three target values (k_{effBOC}, k_{effEOC} and $F^{N}_{\Delta H}$) were
implementation times (Pentium 4 Mobile CPU 1.7 GHz, 256 MB
support vectors as the measure of model generalization characteri
on 5000 data points. The accuracy of the model was determined
average deviation (RAD) defined as

$$\mathrm{RMSE} = \sqrt{\frac{\Sigma_{i=1}^{n}(y_i -}{n}}$$

$$\mathrm{RAD} = \frac{\Sigma_{i=1}^{n}(|\,y_i - f_i\,|/y_i}{n}$$

where f_i stands for predicted value corresponding to the targe
percentage of tested data points which had the predicted value dev

$$\frac{|y_i - f_i|}{y_i} \times 100\% > 2$$

In the case of RBF kernel function, the initial values of free parame
on the LIBSVM code. The ranges for every parameter (C, ε, and σ)
to 1000 for C and 0.001 to 2.0, and 1 to 7.07 ($\sqrt{50}$) for ε and
populations each consisting of 100 members. The training set co
500 data points. The best result was obtained for $C = 371.725$, $\varepsilon =$

In the case of the PF kernel function, we decided to set the d par
simplicity reasons C and ε were set to 371.725 and 0.05154, resp
are given in Table 1 while in Table 2 comparison results for PF ker

Table 1: Comparison of results for RBF kernel

Table 2: Comparison results for PF kernel func

The results of preliminary tests suggest that appropriate regressic
all target values regardless of the applied code package. The on
model to be developed. The implementation or deployment time
seconds for 5000 calculations) is not the issue. The accuracy for th
while additional effort has to be placed on developing the $F^{N}_{\Delta H}$ mod
training set size.

4.2. Training Set Size Influence on SVR Model Quality

SVR model quality can be interpreted as the time required for
generalization characteristics of the model. As shown in the previ
time is not the key issue.

As discussed previously, the size of the training set influences all f
analysis of that influence is necessary. Here, we present the resul
development using LIBSVM code package (see Figure 2). The cha
target values are qualitatively very similar.

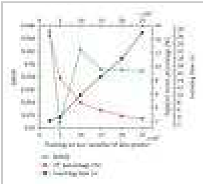

Figure 2: Training set size influence on model

Apart from the anomaly observed for the RMSE curve at the tr
statistical and random characteristic of the training and testing dat
properties (low SV percentage) of the models increase with the in
also increased exhibiting a nearly linear trend.

5. Conclusions

This work introduces a novel concept for fast evaluation of reac
regression model relying on the state of the art research in the fiel

Preliminary tests were conducted on the NPP Krško reactor co
reference data. Three support vector regression code packages w
creating regression models of effective multiplication factor at
multiplication factor at the end of the cycle (k_{effEOC}), and power p

The preliminary tests revealed a great potential of the SVR met
loading pattern evaluation. However, prior to the final conclusior
codes, additional tests and analyses are required, mainly focuse
influencing its size, the required size of the training set and param

In the case of the scenario involving machine learning from the
code, we do not anticipate any major changes in the learning s
implementation. However, generation of training and testing data
and requiring more hardware resources).

These are the issues that are within the scope of our future researc

References

1. V. N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons
2. E. Osuna, R. Freund, and F. Girosi, "Training support vecto *Proceedings of the IEEE Computer Society Conference on Cc* pp. 130–136, San Juan, Puerto Rico, USA, June 1997.

3. B. Schölkopf, *Support vector learning*, Ph.D. thesis, R. Older

4. S. M. Clarke, J. H. Griebsch, and T. W. Simpson, “Analysis complex engineering analyses,” in *Proceedings of the Desig Computers and Information in Engineering Conference (DET* 2003.

5. T. Gu, W. Lu, X. Bao, and N. Chen, “Using support vector r melting point of binary and ternary compound semiconducto 136, 2006.

6. E. Myasnikova, A. Samsonova, M. Samsonova, and J. Reinit determination of the developmental age of a *Drosophila* emb patterns,” *Bioinformatics*, vol. 18, pp. S87 – S95, 2002.

7. S. Nandi, Y. Badhe, J. Lonari, et al., “Hybrid process model networks/support vector regression and genetic algorithms: catalyst,” *Chemical Engineering Journal*, vol. 97, no. 2-3, p

8. D. J. Strauß, G. Steidl, and U. Welzel, “Parameter detectior support vector machines,” *Applied Numerical Mathematics*,

9. D. O. Whiteson and N. A. Naumann, “Support vector regres physics,” *Neurocomputing*, vol. 55, no. 1-2, pp. 251 – 264,

10. K. Trontl, T. Šmuc, and D. Pevec, “Support vector regressic for multi-layer shields,” *Annals of Nuclear Energy*, vol. 34,

11. N. Cristianini and J. Shawe-Taylor, *An Introduction to Suppo Learning Methods*, Cambridge University Press, Cambridge,

12. A. J. Smola and B. Schölkopf, “A tutorial on support vector no. 3, pp. 199 – 222, 2004.

13. A. J. Smola, *Learning with kernels*, Ph.D. thesis, Technische

14. B. Petrović, D. Pevec, T. Šmuc, and N. Urli, “FUMACS (FUel Institute, Zagreb, Croatia, 1991.

15. S. R. Gunn, “Support vector machines for classification and Mathematics, University of Southampton, Southampton, UK,

16. N. Aronszajn, “Theory of reproducing kernels,” *Transactio* no. 3, pp. 337 – 404, 1950.

17. B. Üstün, W. J. Melssen, M. Oudenhuijzen, and L. M. C. Buy regression parameters by genetic algorithms and simplex op 1-2, pp. 292 – 305, 2005.

18. R. Collobert and S. Bengio, “SVMTorch: support vector mac *Journal of Machine Learning Research*, vol. 1, no. 2, pp. 143

19. C.-C. Chang and C.-J. Lin, “LIBSVM: a library for support v

20. I. H. Witten and E. Frank, *Data Mining: Practical Machine Le* San Fransisco, Calif, USA, 2nd edition, 2005.