# Kernel Generations for a Diagnosis Model with GP

Jongseong Kim and Hoo-Gon Choi*
*Department of Systems Management Engineering, Sungkyunkwan University, Suwon, Republic of Korea*

Keywords: Diagnosis Model, Genetic Programming, Absorbing Evolution, Accuracy and Recall Rate, Computing Time.

Abstract: An accurate diagnosis model is required to diagnose the medical subjects. The subjects should be diagnosed with high accuracy and recall rate by the model. The laboratory test data are collected from 953 latent subjects having type 2 diabetes mellitus. The results are classified into patient group and normal group by using support vector machine kernels optimized through genetic programming. Genetic programming is applied for the input data twice with absorbing evolution, which is a new approach. The result shows that new approach creates a kernel with 80% accuracy, 0.794 recall rate and 28% reduction of computing time comparing to other typical methods. Also, the suggested kernel can be easily utilized by users having no and little experience on large data.



## 1 INTRODUCTION

The number of latent subjects with type 2 diabetes mellitus in Korea has rapidly increased over the past three decades. In general, a laboratory test is taken for the latent subject to figure out seriousness of the disease. Many specialized diabetes clinics in Korea utilize fasting glucose level as the main parameter to diagnose type 2 diabetes mellitus although it is changed on daily basis and correlated with other testing parameters. If the fasting glucose level is > 120 mg/dL, type 2 diabetes mellitus is diagnosed for latent subjects. However, a level of fasting glucose is correlated with other laboratory test parameters. Therefore, it is necessary for accurate diagnosis by figuring out the relationship of between fasting glucose level and various test parameters. If the accurate diagnosis model is developed for type 2 diabetes mellitus by using good kernel functions, it is helpful for the clinics to diagnose the disease with low misdiagnosing errors and high recall rate. The diagnosis model classifies the subjects into either patient group or normal group, and the relationship between the principal parameters among testing parameters and subject groups can be specified. The purpose of this study is to generate an excellent kernel for an accurate diagnosis model with high recall rate of type 2 diabetes mellitus by which the test results are accurately grouped or classified. For this purpose, a new method “absorbing evolution (AE)” is developed by optimizing support vector machine (SVM) kernel through genetic programming (GP) in this study.

When SVM is used to classify the input data into two classes with a linear function in a hyperplane, two conditions should be established: one is setting for the amount of classifying errors using cost parameter and the other is defining kernel functions. The cost parameter is required for determining classification accuracy and fitness, and kernel function decides whether the test data are linearly separable or not. The kernel function has its own cost parameter. In this study, the accuracy of kernel function is the main concern along with fixed cost parameter.

GP has a complex tree structure consisting of both function node and terminal node in each program as an object. The tree structure is presented by S-Expression formats of Lisp language. Comparing with GA, GP has different representation scheme for genes. GP operations including crossover, mutation, inversion or permutation, edit, capsulization, and elimination perform on the basis of each tree or program. To obtain the optimal solution for a given problem, five components should be determined. They are a set of terminals, a set of functions, fitness functions, algorithm parameters, and terminating condition. This study focuses on a new method for constructing the initial population and defining a fitness function for

*Correspondig author

Kim J. and Choi H..
Kernel Generations for a Diagnosis Model with GP.
DOI: 10.5220/0004028500570062
In *Proceedings of the International Conference on Data Technologies and Applications* (DATA-2012), pages 57-62
ISBN: 978-989-8565-18-1

clinical data, and fast GP evolution for generating SVM kernel.

## 2 POTENTIAL RESEARCH ON KERNEL AND PARAMETERS

The well-known evolutionary algorithms (EA) such as GP, genetic algorithm (GA), particle swarm optimization (PSO), and evolutionary strategy (ES) have been combined into SVM to evolve kernel functions or kernel parameters. Howley and Madden (2005), and Gagne, et al. (2006) evolved a kernel function using GP. Their methods produced good results for practical cases, but there was no guarantee for the final genetic kernel to be positive-semidefinite (PSD). Sullivan and Nuke (2007) proposed combinatory kernel-bounded operations to develop new complex kernels from basic predefined kernels. Methasate and Theeramunkong (2007) proposed a weighted tree in which weight of an edge becomes a parameter of the children connected to their parent nodes. Also, the weight is adjusted by gradient descent using GP. Simian (2008) introduced a multiple kernel based on simple polynomial kernels using GP. Friedrichs and Igel (2005) proposed a covariance matrix adaptation evolutional strategy (CMAES) to extend the radial basis function (RBF) kernel with scaling and rotation. Then, invariance of linear transformation was realized within the space of SVM parameters. A mechanism, which is similar with reverse singular value decomposition (SVD), was used to guarantee that the final kernel is semi-positive definite. Phienthrakul and Kilsirikul (2005 and 2008) used ES for learning the weights in a weighted linear combination of Gaussian radial basis functions. Souza et al. (2006) used PSO to obtain optimal parameters for multi-class classification in a Gaussian kernel function. In a similar way, Huang and Wang (2006), and Lessmann et al. (2006) used GA for classification. Runarsson and Sigurdsson (2006) used a parallel ES to generate optimal parameters in a Gaussian kernel. Mierswa (2006) combined PSO into ES to solve the constrained optimization problem related to SVM. Keerthi et al. (2007) investigated a method to tune parameters in SVM models based on minimizing a smooth performance validation function. Simaian and Stoicar (2009) proposed a stationary tree structure having a combination of known kernels in which each kernel parameters were encoded in a single chromosome. These parameters are then optimized using GA. Kernel functions should be PSD and typically meet with Mercer conditions (Refer to Section 4.1).

The previous studies have not considered computing time for selecting or generating kernels. However, faster selection and generation of kernels are necessary for reducing the time since both data sizes and dimensions have been big and large. This study proposes AE algorithm to generate kernels with fast speed in GP. This new approach is applied for laboratory test data collected from latent subjects of type 2 diabetes mellitus.

## 3 TARGET DATA DESCRIPTIONS

In this study, laboratory test results were collected from a specialized diabetes mellitus clinic in Seoul, Korea in 2009. The total number of data items was 953.The laboratory test includes 47 different parameters related to liver function, hematology, urinalysis, blood sugar, kidney profile, and lipid profile. Such parameters having either identical values or many missing values are removed regardless of gender. A total of 32 parameters including gender type are selected for analysis. Also, test values located outside of the upper and lower limits are removed from each parameter regardless of subject to reduce any possible measuring errors.

In this study, LIBSVM (Chang and Lin, 2011) is used for separating the subjects into normal group (false) and patient group (true) in SVM learning.

## 4 SVM KERNEL OPTIMIZATION USING GP

In this study, SVM is used for classification of laboratory test results collected from 953 latent subjects. For this purpose, either linear classification or nonlinear classification is appropriate for such big data. Maximum margin SVM is available for linear classification while soft margin SVM is good for it with penalties given to misclassification. When both SVMs are still inappropriate, kernel trick is utilized by which the data can be linearly classified after mapping the original data into high dimensional space. Nonlinear classification adapts both misclassification penalty and kernel trick. Accuracy of a kernel affects the performance of nonlinear classification. Therefore, it is important to select better kernels. There are two ways to select SVM kernels: one is based on expert's experience and the other is using popular methods such as grid search,

gradient search, GA, and PSO. In this study, GP is adapted as a new approach. GP needs high computing time to obtain the kernel and no standard procedure is available. However, GP does not require the expert's experience or knowledge to generate kernels. This study proposes a new approach to reduce its computing time in searching kernels.

## 4.1 Mercer's Theorem and Initial Population

The kernel should be positive-semidefinite (PSD) to satisfy Mercer's theorem. If all kernel matrices generated by kernels are symmetry and have eigenvalues which are greater than 0, the kernel become PSD. All evolved programs generated by GP should meet the theorem to be kernels. However, only a few programs satisfy the theorem at early evolution stage, even when evolutionary process for given data is terminated. A typical way of resolving this problem, the number of programs and generations can be increased. However, this resolution requires higher computing time. To reduce the time, the initial population is made of programs through GP instead of using random population in this study. Then, more programs satisfying Mercer's theorem are generated as shown in Table 1. Since the initial population is obtained, it proceeds again to generate the optimize SVM kernel through GP. In other words, highly accurate SVM kernel to define a diagnosis model is found by using GP twice.

Table 1: Percent of programs by population types.

| Population Type | Percent of programs satisfying Mercer's Theorem (%) |
|---|---|
| Random | 0.5 |
| Initial Population | 5 |

## 4.2 Selected Primitives in GP

In a GP structure, program trees need primitives or operators. Each kernel has scalar outputs from input vectors and it is used for combining programs in evolutionary process. The selection of appropriate primitives is important for GP in terms of speed and simplification of process. In addition to four basic operators (+, -, *, /), several operators such as power and log can make faster evolutionary speed. Other primitives such as *p-norm* and *mhnorm* can make easier to figure out characteristics of raw data in a space mapped by kernels. Also, such primitives as L2 norm and *p-norm* can be used to determine the features of a multi-dimension space. Table 2 shows those primitives with their argument and return types adapted in this study. The combination between primitives is done by Automatic Defined Function (ADF) method which allows strict primitive operations only proved mathematically.

Table 2: Selected primitives used for GP.

| Name | Args. And return types | Description |
|---|---|---|
| ssadd, sssub, ssmul, ssdiv, sspow | (scalar, scalar) → (scalar) | arithmetic operations |
| vvadd, vvsub, dot | (vector, vector) → (vector) | |
| vsmul, svmul, vsdiv, vspow | (vector, scalar) → (scalar) | |
| ssin,scos,stan | (scalar) → (scalar) | triangular functions |
| vsin,vcos,vtan | (vector) → (vector) | |
| sexp, slog, sneg, sabs, ssqrt | (scalar) → (scalar) | exponential, log, negative, absolute |
| vexp, vlog, vneg, vsbs, vsqrt | (vector) → (vector) | |
| p_norm, norm2 | (vector, vector) → (scalar) | p-norm, L2 norm |
| p_normdist,norm2dist, mhdist | (vector,vector) → (scalar) | p-norm distance, L2 norm distance, Mahalanobis distance |
| random scalar | (scalar) | |
| random vector | (vector) | |

## 4.3 Fitness Function

Fitness function is used as the major criterion to select better kernels. An appropriate fitness function should be established to make ensure whether kernel functions meet with required conditions or not. In this study, laboratory test results are analysed to develop an accurate diagnosis model. The model can classify the results into patient group and normal group. The conditions required for kernel are the overall precision of classification and recall rate. If a kernel classifies correctly real patients as true patients and normal subjects as normal, the model has high recall rate. Large costs would be paid if the model misclassifies real patients as normal subjects or vice versa. In addition to precision and recall rate, the number of support vectors should be low in the fitness function. The data mapped into kernel functions would be linearly separable better if the number of vectors is low. Howley and Madden (2005) presents that overfitting risks are reduced with low number of vectors. In this study, the developed fitness function includes the number of

support vectors as shown Eq. 1.

$$\text{Fitness} = \text{AVG}\left(\frac{\text{SV} * \text{R}^2}{\text{AC} \times (\text{PC} \times 100 \times \alpha + \text{RC} \times 100 \times \beta)}\right) \quad (1)$$

SV is the number of support vectors, R is the radius of the smallest hypersphere (Cristianini, 2000), AC is the accuracy, PC is the precision rate, RC is the recall rate in SVM training/validation, and α and β are weights given for precision and recall rate. In this study, α and β are set as 0.2 and 0.8, respectivelysince the recall rate is regarded as more important measurement in developing diagnosis models.

## 4.4 Absorbing Evolution (AE) Algorithm

There exist two major problems in using GP to obtain optimized SVM kernels: one is high computing time and the other is lack of PSD programs. The latter is more serious in terms of evolutionary speed and dropping into optimal solution locally. Parallel evolutionary algorithm (PEA) can solve these problems in which the original data is divided into partial populations, each population evolves independently and its result is shared by migrating its objects each other with specific operators or primitives under given conditions (Kim, 2011). Yet, the algorithm has the possibility to generate too small number of programs during evolution process. In this study, the island algorithm as a typical PEA is modified and utilized to minimize this problem. The modified island algorithm is defined as "absorbing evolution (AE)" algorithm. The AE algorithm composes a target population consisting of the desired objects and defines it as the initial partial population. Then, the target population absorbs new objects from other populations.If some objects aremigrated from a partial population, the same number of new objects migrates into the partial population in the island algorithm. Therefore, the population size is always same. In AE algorithm, however, the size of the target population is increased through migration of objects selected from other partial populations.

The procedure of AE algorithm is as follows:

*Step 1: Initialization*
Compose target population with desired objects and the rest of objects are grouped into other partial populations

*Step 2: Termination*
Establish terminating conditions

*Step 3: Migration*

*Step 3.1: Selection of migrating objects*
Select migrating objects based on both migration rate defined for each partial population and fitness function values, and copy

*Step 3.2: Transferring migrating objects into neighbouring populations*

*Step 3.3: Receiving migrated objects*

*Step 3.4: Exchanging the existing objects in the current population with new migrating objects to maintain the same size of population*

*Step 4: Absorption*

*Step 4.1: Selection of absorbing objects except migrating objects*
Select absorbing object based on absorbing conditions

*Step 4.2: Target population absorbs selected objects*

*Step 5: Deportation and elimination*
Target population deports and removes those objects on the basis of deportation criteria

*Step 6: Executing standard GP and back to Step 2.*

## 4.5 Setting Algorithm Parameters in GP

In this study, the initial population is made of programs through GP (GP-1) to reduce computing time. Then, more programs satisfying Mercer's theorem are generated. Since the initial population is obtained, it proceeds again to generate the optimize SVM kernel through GP (GP-2) with AE. In other words, highly accurate SVM kernel to define a diagnosis model is found by using GP twice.

Table 3: Algorithm parameter settings.

| | GP-1 | GP-2 |
|---|---|---|
| Number of iterations | 30 | 10 |
| Number of populations | 1000 | 5000 (including from GP-1's best results) |
| Number of generations | 1000 | 1000 |
| Crossover probability | 0.8 | 0.8 |
| Mutation probability | 0.4 | 0.1 |
| Selection algorithm | Tournament t= 3 | Tournament t= 3 |
| SVM cross validation | 10 fold validation | 10 fold validation |
| Terminate condition | None | None |
| Additional condition | Mercer's theorem | Mercer's theorem or not matter and migration rate, absorption condition |

Table 3 shows GP algorithm parameters and

terminate condition. GP-1 focuses on kernel function satisfied Mercer's theorem quickly. So, mutation probability is set high. GP-2, all parameters are general except migration rate, absorbing condition and elimination rate. Migration rate is 0.1. Absorbing condition is whether a function satisfiesMercer's theorem or not. Finally, elimination rate is 0.05.

# 5 RESULTS

Table 4: The 13 kernels obtained for laboratory test results.

| | Kernels | | | | | |
|---|---|---|---|---|---|---|
| 1 | scos(mhdist(vsdiv(VE1, 1.0), VE0)) | | | | | |
| 2 | scos(mhdist(VE1, vneg(svmul(-1.0, VE0)))) | | | | | |
| 3 | scos(mhdist(VE1, vspow(VE0, 1.0))) | | | | | |
| 4 | scos(mhdist(VE1, vabs(vabs(VE0)))) | | | | | |
| 5 | scos(mhdist(VE1, vsdiv(VE0, 1.0))) | | | | | |
| 6 | scos(mhdist(vspow(VE1, 1.0), VE0)) | | | | | |
| 7 | scos(mhdist(svmul(1.0, VE1), VE0)) | | | | | |
| 8 | scos(mhdist(vsmul(VE1, 1.0), VE0)) | | | | | |
| 9 | scos(mhdist(VE1, vvsub(vvsub(VE1, VE1), vneg(VE0)))) | | | | | |
| 10 | scos(mhdist(vabs(vspow(VE1, 1.0)), VE0)) | | | | | |
| 11 | scos(ssadd(slog(-1.0), mhdist(VE1, VE0))) | | | | | |
| 12 | scos(mhdist(vvsub(svmul(1.0, VE1), vvsub(VE1, VE1)), VE0)) | | | | | |
| 13 | scos(mhdist(VE0, vsmul(VE1, 1.0))) | | | | | |
| | Fitness | Kernel Making Time | SVM Learning Time | SVM Accuracy | Precision | Recall |
| 1 | 0.269 | 0.878 | 0.005 | 79.412 | 1.000 | 0.794 |
| 2 | 0.269 | 0.854 | 0.004 | 79.412 | 1.000 | 0.794 |
| 3 | 0.269 | 0.902 | 0.004 | 79.412 | 1.000 | 0.794 |
| 4 | 0.269 | 0.840 | 0.004 | 79.412 | 1.000 | 0.794 |
| 5 | 0.269 | 0.964 | 0.004 | 79.412 | 1.000 | 0.794 |
| 6 | 0.269 | 0.901 | 0.004 | 79.412 | 1.000 | 0.794 |
| 7 | 0.269 | 0.895 | 0.007 | 79.412 | 1.000 | 0.794 |
| 8 | 0.269 | 0.897 | 0.006 | 79.412 | 1.000 | 0.794 |
| 9 | 0.269 | 0.984 | 0.004 | 79.412 | 1.000 | 0.794 |
| 10 | 0.269 | 1.002 | 0.004 | 79.412 | 1.000 | 0.794 |
| 11 | 0.269 | 0.998 | 0.005 | 79.412 | 1.000 | 0.794 |
| 12 | 0.269 | 1.227 | 0.004 | 79.412 | 1.000 | 0.794 |
| 13 | 0.269 | 1.146 | 0.005 | 79.412 | 1.000 | 0.794 |

*VE0, VE1 are single data vector (like X, Y).

Table 4 presents top 13 kernels which have identical fitness values obtained through GP using AE algorithm. These kernels have the identical fitness value. Among the kernels, the second kernel has the least computing time and can be selected finally. When the kernel is applied for the original data, that is, laboratory test results with 10 fold cross validation, 80% accuracy of classification is shown while other classifiers such as RBF kernel and linear kernel are shown about 81%. Also, the recall rate is 0.794 for GP with AE algorithm, and it is 0.644 for linear kernel. Even though other kernels present similar performance in terms of accuracy and recall rate, the new method suggested in this study shows the highest recall rate and needs less computing time compared to standard GP algorithms. The time is reduced as much as 28%.

The selected kernel has the expression as shown in Eq. 2.

$$\text{Kernel} = \cos\left((\mathrm{X}-\mathrm{Y})\sum^{-1}(X-Y)^{T}\right) \quad (2)$$

where $\sum^{-1}$ is an inverse of the covariance matrix of the original data, e.g. the laboratory test results. Both X and Y are single data vectors in the original space.

# 6 CONCLUSIONS AND DISCUSSION

This study focuses on generating an accurate diagnosis model from laboratory test results obtained by type 2 diabetes mellitus subjects. The accurate model should be able to classify the subjects into patient group and normal group with high precision. There are 32 test parameters including fasting glucose level in a laboratory test. Because these parameters are correlated and have complex relations, a diagnosis model cannot classify the subjects clearly. In other words, misclassification of normal subjects into patient group or vice versa can be occurred. This study suggests a new approach to optimize SVM kernels with GP. Especially, GP is utilized twice along with AE algorithm. Then, the accuracy of the best kernel is 80% and the recall rate is 0.794. Other typical kernel like RBF shows similar accuracy. However, the method suggested by this study shows the highest recall rate and needs less computing time although it utilizes GP twice. In addition to this achievement, other advantage of this study is easiness to generate a good diagnosis model from the developed kernel function without expert's experience on clinical data. Yet, the time should be minimized by better approaches. Also, further research should be followed by investigation of advanced

methodologies to generate more PSD programs.

## ACKNOWLEDGEMENTS

This research is supported by Samsung Research Fund (2010-0683-000) of Sungkyunkwan University, Suwon, Republic of Korea.

## REFERENCES

Chang, C. C. and Lin, C. J., 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, Vol. 2, No. 27, pp. 1-27, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

Cristianini, N. and Shawe-Taylor, J., 2000.*An Introduction to Support Vector Machines*. Cambridge University Press.

Friedrichs, F. and Igel, C., 2005.Evolutionary tuning of multiple SVM parameters, *Neurocomputing*, Vol.64, pp.107–117.

Gagne, C., Schoenauer, M., Sebag, M. and Tomassini, M., 2006. Genetic programming for kernel based learning with co-evolving subsets selection, *Proceedings of Parallel Problem Solving in Nature*, LNCS, No. 4193, pp. 1006–1017.

Howley, T. and Madden, M. G., 2005. The genetic kernel support vector machine: Description and evaluation, *Artif. Intell. Rev*., Vol.24, No.3–4, pp.379–395.

Hsu, C. W., Chang, C. C. and Li, C. J., 2010, A Practical Guide to Support Vector Classification*, National Taiwan University,* Taiwan.

Huang, C. L. and Wang,C. J., 2006. A GA-based feature selection and parameter optimization for support vector machines, *Expert Systems with Applications*, Vol. 31, No. 2, pp. 231–240.

Huang, C. L., Chen, M. C.and Wang, C. J., 2007. Credit scoring with a data mining approach based on support vector machines, *Expert Systems with Applications,* Vol. 33, No. 4, pp. 847–856.

Kim, Y. K, 2011. Evolution Algorithms, *Chonnam National University Press*, Republic of Korea.

Lessmann, S., Stahlbock, R. and Sven, F., 2006.Genetic algorithms for support vector machine model selection, *Proceedings of International Joint Conference on Neural Networks*, pp. 3063–3069.

Methasate, I. and Theeramunkong, T., 2007.Kernel Trees for Support Vector Machines, *IEICE Trans. Inf. & Syst.*, Vol. E90-D, No 10, pp. 1550–1556.

Mierswa, I., 2006. Evolutionary learning with kernels: A generic solution for large margin problems, *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 1553–1560.

Phienthrakul, T. and Kijsirikul, B., 2005.Evolutionary strategies for multiscale radial basis function kernels in support vector machines, *Proceedings of Conference on Genetic and Evolutionary Computation*, pp.905–911.

Phienthrakul, T and Kijsirikul, B., 2008. Adaptive stabilized multi-RBF kernel for support vector regression, *Proceedings of International Joint Conference on Neural Networks*, pp. 3545–3550.

Runarsson, T. P. and Sigurdsson, S., 2004.Asynchronous parallel evolutionary model selection for support vector machines,*Neural Information Processing – Letters and Reviews*, Vol. 3, No. 3, pp. 59–67.

Keerthi, S., Sindhwani, V. and Chapelle, O., 2007.An efficient method for gradient-based adaptation of hyperparameters in SVM models, *Advances in Neural Information Processing Systems 19*, MIT Press, Cambridge, MA, pp. 674–480.

Simian, D., 2008.A model for a complex polynomial SVM kernel, *Proceeding of the 8th WSEAS International Conference on Simulation, Modelling and optimization*, pp. 164–169

Simian, D. and Stoica, F., 2009.An evolutionary method for constructing complex SVM kernels, *Proceedings of the 10th WSEAS International Conference on Mathematics and Computers in Biology and Chemistry*, pp. 172–177.

Souza, B. F., Carvalho, A. C., R. Calvo and Ishii, R. P., 2006.Multiclass SVM model selection using particle swarm optimization, *Proceedings of the Sixth International Conference on Hybrid Intelligent Systems*, pp. 31–34.

Sullivan, K. and Luke, S., 2007. Evolving Kernels for Support Vector Machine Classification, *Proceedings of the 9th annual conference on Genetic and evolutionary computation, GECCO*, pp. 1702–1707.