An Investigation into the Effects of Multiple Kernel Combinations on Solutions Spaces in Support Vector Machines

Paul Kelly, Luca Longo

An investigation into the effects of Multiple Kernel combinations on solutions spaces in Support Vector Machines.

Paul Kelly[1] and Luca Longo[2]

[1] National College of Ireland, Dublin, Ireland
x15006956@student.ncirl.ie
[2] School of Computing, College of Health and Sciences, Dublin Institute of Technology, Dublin, Ireland
luca.longo@dit.ie

Abstract. The use of Multiple Kernel Learning (MKL) for Support Vector Machines (SVM) in Machine Learning tasks is a growing field of study. MKL kernels expand on traditional base kernels that are used to improve performance on non-linearly separable datasets. Multiple kernels use combinations of those base kernels to develop novel kernel shapes that allow for more diversity in the generated solution spaces. Customising these kernels to the dataset is still mostly a process of trial and error. Guidelines around what combinations to implement are lacking and usually they requires domain specific knowledge and understanding of the data. Through a brute force approach, this study tests multiple datasets against a combination of base and non-weighted MKL kernels across a range of tuning hyperparameters. The goal was to determine the effect different kernels shapes have on classification accuracy and whether the resulting values are statistically different populations. A selection of 8 different datasets are chosen and trained against a binary classifier. The research will demonstrate the power for MKL to produce new and effective kernels showing the power and usefulness of this approach.

1 Introduction

Support Vector Machines are one of the many methods in Machine Learning used as a discriminative classification algorithm. They are a form of supervised learning that allows the categorisation of inputs based on their position in a feature space. They traditionally address binary classification problems by separating data points in the solution hyperspace by means of a hyperplane. The goal is to generate the maximal margin between the hyperplane and the closest points of each category being classified. When a hyperplane cannot be found, which can adequately separate the data points, a kernel can be used [16]. Kernels are used in SVMs to allow non-linearly separable data points to be mapped to a higher, and potentially infinitely higher dimensional space [22]. Using the kernel trick this can be done with relatively low computation cost as the simple inner product of vectors. In the case of Mercer compliant kernels [16] , which are the

focus of this study, the kernels have the property of being positive semi-definite (PSD) that guarantees a convex solution space. A PSD matrix is defined as one that has only positive eigenvalues. It's resultant convex space guarantees a search minimum and it reduces the search requirements of non-convex solution spaces associated with methods such as Artificial Neural Networks (ANNs). Multiple Kernel Learning is an expansion on these base kernels and used when the traditional kernels, such as Radial Based Functions or Polynomial expansions, are insufficient. Properties of PSD kernels is that the product of two will always produce another semi-definite kernel. This allows for a broad range of combinations and shapes of kernel to be used against data that is difficult to classify. Previous studies on Multiple Kernel Learning have focused on a few themes, the use of a genetic algorithm for hyperparameters selection [8], genetic algorithms for kernel selection, the weighted combination of multiple kernels to determine kernel shape [23], and large scale kernel combinations [1] [6]. These approaches focus on specifics of the MKL implementation for a chosen task such as image identification [21] or landslide detection [13]. They are also generally more specific in the choice of kernel combinations and commonly focus on combinations of two different base kernel types. The computation time of constructing the kernels and testing them against the data is a prohibitive feature of the MKL research and the sparse knowledge around global kernel shape approaches makes the scope of research quite broad. The fundamental question of whether the MKL kernel will produce novel results is not explicitly addressed and that is the goal of this research.

This paper is structured as follows. Section 2 will address the previous work done in this field and the varying approaches used in relation to their strength and weaknesses regard computation time and performance. Section 3 will detail the approach put forward in the paper and address the choice for experimental design, evaluation methods used, and datasets chosen. Sections 4 and 5 will deal with the results, and what conclusions can be drawn from them, respectively.

2 Related Work

SVM optimisation is based on the configuration and adjustment of hyperparameters along with the kernel choice when the data is non-linearly separable. Additional hyperparameters also come into play regarding the kernels themselves with specific values needed based on type. Multiple approaches to tuning these hyperparameters have been used to overcome what was originally described by Boser [2] as needing direct configuring and altering. This problem of tuning is described by Diosan [5] as an empirical one and should be treated in similar ways to other problems of scale. As an example, the weight of any one hyperparameter, such as the slack variable, has a direct effect on the influence of each support vector [20]. Slack in SVMs is the allowance given to the algorithm to accept misclassified or anomalous data points in the training data. Similarly, the gamma value for the kernel defines the nonlinear mapping to the higher feature space [14]. Weighting these vital parameters, and avoiding over-weighting, then

becomes a trade-off between maximising the support vector width and reducing errors in classification [4]. The following sections will describe the different approaches used to tackle the tuning of hyperparameters, the use of genetic algorithms in previous MKL work, and what considerations computational complexity plays in the implementation of MKLs.

2.1 Hyperparameter Tuning

One approach used by both Shermeh [20] and Lessmann [12] to find optimal hyperparameter values for a kernel is the grid search approach where a range of values are iterated through to find the best solution. Alternatively, Genetic Algorithms (GA) whereby a fitness measure is used to assess a population of randomly chosen parameter sets applied to the SVM is also popularly used to determine the best values. The computation cost, however, of Genetic Algorithm Support Vector Machines (GASVMs) makes them less useful, particularly over larger datasets, as the execution time for all the permutations becomes high. While GASVM approaches have gained popularity in recent years, they have not addressed general rules for the effectiveness or differentiation from base kernel outcomes that the specific kernel combinations produce. While SVM and kernel specific hyperparameter tuning are needed to ensure robust modeling, the use of which kernel combinations determines what shape and eventual properties the MKL kernel will have.

2.2 Genetic Algorithm approach to Kernel weights

The question of which kernels to combine and their effect on robust models is another imprtant decision after hyperparameter selection and can also use GA-SVMs to solve for more effective combinations. Polynomial and Radial Based Function(RBF) kernels were combined by Howley [10] to identify hand-written characters, which is an extremely well-studied problem, to see the effect of their different weighting. The RBF kernel, or Gaussian Kernel, is known for its effectiveness at handling small, high dimensionality datasets which are both notoriously difficult properties for a classification algorithm. Li [13] demonstrated the RBF's ability to perform well at nonlinear mapping while Diosan [5] highlights its effectiveness at nonparametric classification functions. Both papers, however, point out the conflicting trade-off that optimisation brings in gains of training accuracy over the complexity of the model. Other approaches have also used a weighted method to determine the role that each of the component kernels has in the resulting MKL kernel. For example, Deng took this approach to include higher weight based on individual features of the dataset but highlighted the resultant search space as being a prohibitive factor [3]. Similarly to the previous approaches, they suggested a possible approach being the use of a GASVM (Genetic Algorithm SVM) to calculate the optimal solution for kernels as opposed to hyperparameter choice, but again this comes with the high cost of computation expense and time required. Weighting can be counter productive also as

strong weighting of less optimal kernels in the MKL arrangement can undermine performance, as documented by by Li et al [13] in their EEG classification MKL-SVM, and Hao's [9] research into primal MKL.

2.3 Computational Cost

Computational cost is not only restricted to the choice of kernel combinations or hyperparameter settings but also in either maintaining Mercer compliance to allow simple gradient descent search or employing a more advanced search algorithm to account for the non-convex solution space that non-Mercer kernels produce. Ensuring the Mercer compliance can be maintained easily by making kernels the inner product of existing Mercer compliant kernels and removing the need for these tests that prove positive semi-definiteness. This combination always results in a continuous, positive semi-definite kernel and due to the commutative property of Mercer kernels the order of base kernels has no bearing on the output. MKL kernels can be tested to ensure that the combinations of kernels are compliant with the positive semi-definite properties of a Mercer kernel as highlighted by Howley et al in their use of GASVMs [10].

3 Design and Methodology

The approach for this paper is to show that over a broad range of configurations, MKL kernels are valid and novel in shape. To take in a large range of possibilities, a broad range of parameters for the individual kernels will be selected. These will be a choice of kernel specific parameters and tested against the various datasets. Each of the starting, or base, kernels will be first tested against the datasets and then each combined to give the new MKL kernels. This combination will be guaranteed PSD, as discussed earlier, as the base kernels will be chosen from preexisting PSD kernels. The aim will be to test the outputs of the base kernels compared to that of the MKL kernels derived from those specific base kernels. The goal is to move beyond most MKL approaches which use fixed or heuristic based techniques identified by Gonen et al [7] and demonstrate the power of combinations.
The approach used will be the CRISP DM method (Cross Industry Standard Process for Data Mining) [19] and involves six major stages: 1)Business understanding, 2)Data understanding, 3)Data preparation, 4)Modelling, 5)Evaluation, and 6)Deployment.

3.1 Business Understanding

The research hypothesis behind this study is that the accuracy of models induced by SVM classifiers by employing base kernels is statistically significantly less than the accuracy of models induced by employing multiple base kernels, at a 95% confidence level. Formally: $H : acc(SVM-baseKernel) <> acc(SVM-MKL)$, with alpha=0.05

3.2 Data Understanding

To test the research hypothesis 8 testing datasets have been taken from the UCI repository and are exclusively binary output variables. The datasets selected are the Prognosis, Heart, Ionosphere, Student Math, Diagnosis, Indian Liver, Student Portuguese, and Pima Indian sets. The datasets were chosen to be relatively small binary sets that have a been previously used in SVM papers and could be run against the full range of hyperparameters and kernels within the allotted time. Variation of the dimensionality, post normalised ranges for standard deviation and variance, and proportion of nominal to categorical variables were also selected for to give a more diverse range of datasets. The focus will be on the individual dataset responses to the different kernel structures and not the specifics of the data.

3.3 Data Preparation

All of the datasets were formatted in the folowing ways to maintain consistency and to make them suitable for the SVM implementation.

- Formatted to CSV file type to allow for import and to standardise to a common file type
- Column values added from Metadata description files for easier referencing
- Categorical and non-numeric fields encoded to numeric values representing each unique entry. This is done to allow them to be used by the SVM.
- Normalised between values of 0 and 1 across all dimension excluding on the output/ classification variable to minimise the influence of large number fields.
- The output variable is converted to a -1 and 1 encoding which will be used as factors in the R code.
- Training and testing datasets are both merged as the k-fold method described later will construct multiple training and testing groups later in the process to ensure the quality of accuracy measurement.

3.4 Modeling

The 10-fold cross validation technique has been adopted to induce models using the original Support Vector Machine learning algorithm. This validation technique has been chosen because it will ensure a measure of accuracy that prevents over-fitting and maximises the variation in the relatively small datasets used.

Base Kernel Details

1. Linear Kernel
   Inner product of [x,y] and c value for offset

$$[k(x, y) = x * y + c]$$

2. Polynomial Kernel (Degrees 2,3,and 4)
   Polynomial kernels expand the examination of similarity between entries in the data. The hyperparameters for the polynomial kernel are the slope, the constant C, as in the Linear Kernel , and the degree, d, of the Polynomial used aginst the combined transposed x and y input vectors.

$$[k(x, y) = (x^T y + c)^d]$$

3. Gaussian Kernel
   The Gaussian Kernel, also known as the Radial Based Function Kernel, is essentially a weighted linear combination that results in a smooth function of the Hilbert space that is defined as a Euclidean space which is complete, separable and infinitely-dimensional.

$$[k(x, y) = e^{\frac{-\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}}]$$

   The sigma parameter determines the width from the classifying points with smaller sizes trending towards local classification and larger sigma for more general classification. It is specifically important for an effective implementation as if underestimated the model becomes particularly susceptible to noise in the training data whereas, if overestimated, it reduces the non-linear power that is associated with Gaussian kernels.

4. Hyperbolic Tangent (Sigmoid) Kernel
   The Hyperbolic Tangent, or Sigmoid Kernel, has its origins in Artificial Neural Networks (ANNs) where the sigmoid function is used by neurons in the ANN for an activation function. It is also referred to as the Multilayer Perceptron Kernel.

$$[k(x, y) = tanh(\alpha x^T y + c)]$$

The Sigmoid Kernel is very effective but is not guaranteed to be positive semi definite at higher levels. These higher levels will be avoided for the purposes of this study in order not to affect Mercer conditions.

Hyperparameter Ranges Hyperparameters used across the kernels are:

- Slack: it determines the cost of improperly labelled data points. Range: (0.1, 1, 10, 100, 1000)
- Tolerance: required for termination of SVM. Range:(0.1, 0.01, 0.001, 0.0001, 0.00001)
- Sigma: parameter for Gaussian kernel. Range: (.01,0.1,0.2,0.5,0.7,0.9).
- Offset: add an additional offset to data points. Range: (1,2,3)
- Scale: used by the Sigmoid and Polynomial. Range: (1,2,3)
- Degree: used as the power for the Polynomial kernel. Range: (2,3,4)

Novel kernels will be named based on their basic kernel components and will total 11 unique combinations all tested again every configuration of hyperparameters. They will be referred to going forward as LinSigGauPoly(LSGP), LinSigGau(LSG), LinSigPoly(LSP), LinGauPoly(LGP), SigGauPoly(SGP), LinSig(LS), LinGau(LG), LinPoly(LP), SigGau(SG), SigPoly(SP), and GauPoly(GP).

3.5 Evaluation

Due to the fact that 10-fold cross validation has been used in the training phase, 10 surrogate models have been produced. Therefore, a distribution of 10 classification accuracies for each combination of dataset, hyper-parameters and kernel, is available for comparison purposes [17]. The Kruskal Wallis test has been chosen to compare obtained distributions of accuracies and to test the research hypothesis. In particular, the distributions obtained when using combination of multiple simple kernels is compared against the distributions of accuracies obtained when employing these individual kernels alone. This method allows for an analysis of variance that is not restricted to normal distributions of equal size. The Kruskal Wallis technique allows for non-parametric analysis of variance [11]. This means that the distribution of values does not need to be normal and also accommodates differing sample sizes for comparison. Both characteristics are needed in the case of the SVM outputs as there are non-normal, and occasionally bimodal, distributions of cross validation results for certain kernels across the hyperparameter range. The range of hyperparameters themselves are also kernel specific which will result in differing population sizes that need to be compared. As an example the Linear/Vanilla kernel doesn't take in kernel specific hyperparameters but instead just requires the slack and tolerance values to be tested which totals only 24 combinations. Other kernels, however, such as Lin, Sig, Gau, Poly will require the entire range of combinations resulting in over 4000 results per dataset. The Kruskal Wallis test is run across each of the 8 datasets and compares each base kernel with the kernels that have it as a component part. An example would be all the result for Polynomial Kernel against the Ionosphere dataset compared to those of the LinPoly kernel which is composed, in part, by the Polynomial kernel. A threshold of 95 percent confidence is required to establish if the distributions match and therefore show a lack of a novel solution space

4 Results

4.1 Cross Validation

Results show (Table 1) the classification accuracies obtained across selected datasets, base kernels and combined kernels. It also placed these accuracies in context by comparing them with those obtained in another similar studies in the literature (GMKL) [15]. The SimpleMKL [18] package, which is very popular method, has also been included with comparable results although over a different set of hyperparameters and completion criteria.

Cross validation is used as it gives an indication of a kernels tendency to overfit. Within our results we see multiple 'error' values for the SVM that represent the best outcome achieved in fitting to the dataset which return a zero value. This represents one hundred percent classification accuracy for the training values. The cross validation being much higher than this zero value indicated that the kernel shape has been over fitted to the training set and therefore under-performs

Dataset	GMKL	SimpleMKL	Base Kernel	MKL
			Gaussian	SigGau
WPBC	79.0	76.7	83.9	82.9
			Gaussian	SigGauPoly
Iono	93.0	91.5	96.2	96.2
			Polynomial	LinGau
Liver	72.7	65.9	72.5	72.0
			Polynomial	LinPoly
Pima	77.2	76.5	78.4	78.4

Table 1. Base Kernel and MKL Kernel performance

when presented with novel data. The role of particular kernels and hyperparameters is not in the scope of this specific research but there are trends visible in the data showing a tendency to overfit when the slack value is set too high. This makes intuitive sense as a high punishment for incorrect value will tend to force the SVM to over accommodate mislabels or outliers. The cross validated SVMs results will be using the second evaluation stage as the populations, per kernel and dataset, that need to be compared to the base kernel values.

4.2 Kruskal Wallis Analysis

The Kruskal Wallins test is performed against each kernel/dataset combination to determine statistically significant differences in the combined kernels output results and that of the base kernels used in their creation

$$H = (N-1)\frac{\sum_{i=1}^{g} n_i(\bar{r}_{i\cdot} - \bar{r})^2}{\sum_{i=1}^{g}\sum_{j=1}^{n_i}(r_{ij} - \bar{r})^2},$$

This method allows for an analysis of variance (ANOVA) that is not restricted to normal distributions of equal size. The test will result in a p_value which will allow a rejection or acceptance of the null or alternate hypothesis set out in the business understanding section.
Of the 224 results there were only 8 populations which show a distribution unchanged through the addition of another kernel. The 8 ranges of accuracies, produced by the new kernels, that didn't show a new distribution are highlighted in **bold** below (Table 2). This shows that for the Maths dataset the gaussian kernel was the only one unchanges in some MKl instance. For the Iono, Liver, and Pima the Sigmoid base kernel was the one without changes in some cases and for the Diag dataset the Polynomial had this property. All other 216 combinations produced a new solution space as a result of being combined with another Mercer Kernel.

4.3 Summary of findings, strengths and limitations

The findings of the study means that the expansion of the base kernels using additional base kernel(s) does result in a unique range of cross validated results

	Base	LS**G**P	LSG	LGP	SGP	LG	SG	GP
Math	Gau	<0.01	**0.66685**	0.01632	**0.73788**	**0.80290**	0.03859	**0.73731**
		LSGP	LSG	LSP	SGP	LS	SG	SP
Iono	Sig	<0.01	<0.01	**0.86221**	<0.01	<0.01	<0.01	<0.01
Liver	Sig	0.02753	**0.05394**	<0.01	0.01759	**0.21516**	0.01934	**0.08873**
Pima	Sig	0.02017	<0.01	0.01120	<0.01	**0.08193**	<0.01	<0.01
		LSGP	LSP	LGP	SGP	LP	SP	GP
Diag	Poly	<0.01	<0.01	<0.01	<0.01	<0.01	<0.01	**0.12087**

Table 2. Base Kernel and MKL Kernel performance

in the vast majority of cases. Most of the p-values related to the comparison of models trained with a base kernels and models trained by adding another base kernel were less than 0.00001. Despite findings strongly support the research hypothesis, these do not give any insight on the impact of multiple kernel on the enhancement of classification accuracies. The confidence in the results can only be stated for the small datasets used relatively balanced target variables. While characteristics of the dataset were noted at the start phase for dimensionality, proportion of categorical fields, and range of values pre and post normalisation, these factors have not been analysed as part of this research and a further investigation is needed to understand how the classification accuracies change and whether dataset properties have a predictable response to the MKL expansions. However, this empirical study does have an impact in validating kernels as viable and distinct kernels for use in SVMs.

5 Conclusion

The conclusion of this study finds that for the vast majority of base kernel result across the varying datasets, the combination with an additional base kernel or base kernel combination resulted in a statistically significant different range of cross validated accuracies. This helps to further validate the use of multiple kernel learning as a solution to inseparable data through the use of combined conventional kernels. It further opens up investigation into how certain data sets might respond to different kernel shapes based on the dataset properties. Future work should be around investigating the specific hyperparameters role, alongside the dataset characteristics. While this study concentrated on accuracy as the measure of the populations of output, additional work could investigate the ways in which the distributions were altered, positively or negatively, to determine patterns and trends that could help with future MKL creation. The datasets in this study were also relatively small, however, later implementations should address the computational cost using higher performing machines to incorporate larger datasets. Differing proportions of output variables should also be tested and, with it, an examination into the role of kernel shape in accommodating datasets with high sensitivity and specificity requirements. As noted before, when increasing the data size and the balance of outcomes, seed values should be set in

the code to ensure the SVM runs against the same partitions of data when using the k-fold cross-validation approach. The scale parameter can also be ignored in future work as the data was pre-normalised but was included for completeness. Finally, the number of base kernels should be expanded to add to the diversity of MKL kernels along with a weighting approach that can consider multiple additions of a specific kernel type rather than MKL kernel composed of just one instance of a particular base kernel.

References

1. Aiolli, F., Donini, M.: Easy multiple kernel learning. ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (April), 23–25 (2014)
2. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A Training Algorithm for Optimal Margin Classifiers. Proceedings of the fifth annual workshop on Computational learning theory pp. 144–152 (1992)
3. Deng, S., Sakurai, A.: Integrated Model of Multiple Kernel Learning and Differential Evolution for EUR / USD Trading. The Scientific World Journal 2014 (2014)
4. Deng, S., Yoshiyama, K., Mitsubuchi, T., Sakurai, A.: Hybrid Method of Multiple Kernel Learning and Genetic Algorithm for Forecasting Short-Term Foreign Exchange Rates. Computational Economics pp. 1–41 (2013)
5. Diosan, L., Rogozan, A., Pecuchet, J.P.: Improving classification performance of Support Vector Machine by genetically optimising kernel shape and hyper-parameters. Applied Intelligence 36(2), 280–294 (2012)
6. Duvenaud, D., Lloyd, J.R., Grosse, R., Tenenbaum, J.B., Ghahramani, Z.: Structure Discovery in Nonparametric Regression through Compositional Kernel Search. Proceedings of the 30th International Conference on Machine Learning 28, 1166–1174 (2013)
7. Gonen, M., Alpaydin, E.: Multiple kernel learning algorithms. Journal of Machine Learning Research. ICML, 2011 pp. 2211–2268 (2011)
8. Gupta, A.K., Guntuku, S.C., Desu, R.K., Balu, A.: Optimisation of turning parameters by integrating genetic algorithm with support vector regression and artificial neural networks. The International Journal of Advanced Manufacturing Technology 77(1-4), 331–339 (2014)
9. Hao, Z., Yuan, G., Yang, X., Chen, Z.: A primal method for multiple kernel learning. Neural Computing and Applications 23(3-4), 975–987 (2012)
10. Howley, T., Madden, M.: The genetic evolution of kernels for support vector machine classifiers. 15th Irish Conference on Artificial Intelligence pp. 445–453 (2004)
11. Kruskal, W.H., W.W.: Use of ranks in one-criterion variance analysis. Am. Stat. Assoc. pp. 583–621 (1952)
12. Lessmann, S., Stahlbock, R., Crone, S.: Genetic Algorithms for Support Vector Machine Model Selection. The 2006 IEEE International Joint Conference on Neural Network Proceedings pp. 3063–3069 (2006)
13. Li, X.Z., Kong, J.M.: Application of GA-SVM method with parameter optimization for landslide development prediction. Natural Hazards and Earth System Sciences 14(3), 525–533 (2014)
14. Lu, D., Qiao, W.: A GA-SVM hybrid classifier for multiclass fault identification of drivetrain gearboxes. 2014 IEEE Energy Conversion Congress and Exposition, ECCE 2014 pp. 3894–3900 (2014)

15. Manik Varma, B.R.B.: More Generality in Efficient Multiple Kernel Learning Manik pp. 1065–1072 (2009)
16. Mercer: Functions of positive and negative type, and their connection the theory of integral equations. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 209(441-458), 415–446 (1909)
17. Mosteller, Tukey: Data analysis, including statistics, In Handbook of Social Psychology. Addison-Wesley. ICML, 2011 (1968)
18. Rakotomamonjy, Bach, C.G.: SimpleMKL. Journal of Machine Learning Research 9 pp. 2491–2521 (2008)
19. Shearer: The CRISP-DM Model: The New Blueprint for Data Mining. Journal of Data Warehousing pp. 13–22 (2000)
20. Shermeh, A.E., Ghazalian, R.: Recognition of communication signal types using genetic algorithm and support vector machines based on the higher order statistics. Digital Signal Processing 20(6), 1748–1757 (2010)
21. Siddiquie, B., Vitaladevuni, S.N., Davis, L.S.: Combining multiple kernels for efficient image classification. Applications of Computer Vision WACV 2009 Workshop on pp. 1–8 (2009)
22. Vladimir Vapnik, C.C.: Support-Vector Networks . Kluwer Academic Publishers, Boston (1995)
23. Xu, Z., Jin, R., Yang, H., King, I., Lyu, M.R.: Simple and efficient multiple kernel learning by group lasso. International Conference on Machine Learning (ICML) pp. 1191–1198 (2010)