Appendix

1 Rank One optimal Equivalent Kernel

In this section we present a different approach to show that the optimal equivalent kernel $\underline{\boldsymbol{Z}}^*$ is a rank one matrix. As discussed in the main paper, this stems from the fact that ∇f has at least one non zero eigenvalue, and therefore the eigenvector $\boldsymbol{u}$ corresponds to a non zero eigenvalue.

First, note that each element in the gradient matrix ∇f is non

2 Experimental Methodology

In this section we describe in details the experimental methodology of the Multi Kernel Learning benchmark [1].

On the large datasets (ADVERT and MULTIFEAT) the data was divided (with stratification) into three equal parts (denoted by segments A,B & C). Segment C was reserved as a testing set. Every algorithm was trained on segment A, with the hyperparameter C values $0.01, 0.1, 1, 10, 100$. The optimal value, as evaluated on the segment B (the validation segment) was picked, and the algorithm was retrained on both segment A and segment B. Then, the algorithm was tested on segment C. This process was repeated with reversed roles for segments A and B, namely with segment B as the training set and segment A as the validation.

This procedure was repeated five times (5×2 cross validation), and concluded with 10 trials. The mean and the standard deviations of the performance metrics in these trials are presented in the main paper.

The PROTEIN database includes a given partition to a train and test sets. The reported results were obtained by performing a 30 fold cross validation on the train set: The train set was partitions into 30 folds (with stratification). At each trial, one fold was reserved for verification of the the optimal hyperparameter C (with candidate values, as before, $0.01, 0.1, 1, 10, 100$), while the rest were used to train the various algorithms. In the main paper, we report the mean and the standard deviations of the various metrics in these 30 trials are presented.

Finally, note that the active kernel account is the number of base kernels that were used by the learner. In other words, "A kernel is *active*, if it needs to be calculated to make a prediction for an unseen test instance" [1].

	3 Kernels	5 Kernels	7 Kernels
RBMKL	75.03 ± 4.65	76.22 ± 4	76.68 ± 5.23
SimpleMKL	74.53 ± 4.94	76.88 ± 5.65	75.83 ± 6.02
GMKL	76.77 ± 5.93	73.37 ± 6.15	75.73 ± 5.44
NLMKL ($p = 1$)	76.29 ± 4.26	74.62 ± 7.83	76.56 ± 4.32
NLMKL ($p = 2$)	73.21 ± 5.3	76.35 ± 3.67	75.85 ± 4.99
Nuc-MKL	84.27 ± 3.79	84.39 ± 3.75	83.91 ± 3.44

Table 1: The classification error on the SONAR database.

3 Additional Experiments

We performed additional experiments and compared the performance of Nuc-MKL to a selected subset, state-of-the-art MKL algorithms on additional databases, SONAR, BREAST and IONOSPHERE. These databases are some of the most frequently used datasets in machine learning.

Our test methodology followed precisely the experimental procedure for the ADVERT and MULTIFEAT datasets. However, since the features were not divided into feature categories or feature sets, we randomly divided the feature into disjoint sets. This partition was, obviously, was performed only once and was fixed in all experiments. Note that each feature set generates a kernel. We experienced with different number of feature sets, and report the results in this section.

3.1 SONAR

The SONAR[1] database contain 208 samples, with 59 features, corresponding to different reflected sonar signals. The classification task is to identify which object is a rock and which is a metallic cylinder. In our experiment, NuC-MKL came on top, with a rather large gap than the other algorithms. The errors of the different algorithms are presented in Table 1.

3.2 BREAST

The BREAST[2] database contain 686 samples (after removing samples with missing entries) of categorical analysis of breast cancer cells. Since the are only nine features, we limited ourselves to a smaller number of kernels compared to the SONAR database. The goal in this task is to classify whether a tumor is malignant or benign. In our experiment, the NuC-MKL performed well, similarly to the linear MKL algorithms, and there is no clear winner since the difference in performance is statistically insignificant. It is important to note that the non-linear MKL NLMKL algorithm fell behind since the data features of this task are apparently very close to the features in the feature plane. Nevertheless,

[1] https://archive.ics.uci.edu/ml/datasets/Connectionist+Bench+%28Sonar%2C+Mines+vs.+Rocks%29
[2] https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29

	2 Kernels	3 Kernels	4 Kernels
RBMKL	96.46 ± 0.99	97.16 ± 1	96.83 ± 0.9
SimpleMKL	96.75 ± 1.28	97.27 ± 0.9	$96.93 \pm 0,77$
GMKL	96.6 ± 1.09	97.09 ± 1.29	96.63 ± 1.09
NLMKL ($p = 1$)	85.67 ± 2.01	85.59 ± 2.34	83.34 ± 2.28
NLMKL ($p = 2$)	86.33 ± 2.31	84.42 ± 2.37	83.29 ± 2.55
Nuc-MKL	96.75 ± 3.44	96.84 ± 1.01	96.87 ± 1.32

Table 2: The classification error on the BREAST database.

	3 Kernels	4 Kernels	5 Kernels
RBMKL	81.35 ± 2.66	81.96 ± 3.12	79.3 ± 3.9
SimpleMKL	80.64 ± 2.87	81.4 ± 2.57	80.68 ± 3.12
GMKL	79.94 ± 3.3	80.35 ± 3.1	80.06 ± 2.36
NLMKL ($p = 1$)	85.18 ± 2.06	83.02 ± 2.44	82.77 ± 3.62
NLMKL ($p = 2$)	84.77 ± 3.63	82.86 ± 2.89	83.03 ± 2.63
Nuc-MKL	92.4 ± 1.76	91.13 ± 2.26	92.35 ± 1.73

Table 3: The classification error on the IONOSPHERE database.

although NuC-MKL is a non-linear algorithm of the base kernels, it allows a linear combination of the base kernels as an outcome as discussed in the main paper, and therefore performed well in this experiment too. The errors of the different algorithms are presented in Table 2.

3.3 IONOSPHERE

This database[3] contain 361 samples, each having 34 features. Some samples contains a large number of null values (more than 6 values) and were removed. In this test, NuC-MKL was the clear winner. The errors of the different algorithms are presented in Table 3.

3.4 PASCAL VOC 2007

Next, we tested NuC-MKL in the classification task of PASCAL VOC 2007. A recent analysis had compared a few state-of-the-art Deep Neural Networks performance in this challenge [2]. In this review, classification was performed by extracting the features from the last fully connected layer, generating a linear kernel, and using a SVM classifier. Table 4 restates the reported mean average precision (mAP), derived according to the experimental methodology of PASCAL 2007 challenge. We tested NuC-MKL performance on three kernels, corresponding the a set of three extracted features set. We tested the NuC-MKL performance using three kernels, corresponding to the three extracted feature

[3] https://archive.ics.uci.edu/ml/datasets/Ionosphere

Categories	**CNN-M128 [3]**	**CNN-S** [4]	**CNN-F [5]**	**NuC-MKL (3)**	**NuC-MKL (6)**
Aeroplane	91.3	90.7	88.7	96.1	96.3
Bicycle	83.9	85.7	83.9	90.8	90.9
Bird	89.2	88.9	87.0	93.8	93.8
Boat	86.9	86.6	84.7	90.9	90.9
Bottle	52.1	50.5	46.9	52.0	53.2
Bus	81.0	80.1	77.5	86.2	84.8
Car	86.6	87.8	86.3	91.6	91.4
Cat	87.5	88.3	85.4	93.1	93.3
Chair	59.1	61.3	58.6	66.1	65.8
Cow	70.0	74.8	71.0	79.1	79.6
Dining table	72.9	74.7	72.6	76.9	77.3
Dog	84.6	87.2	82.0	91.6	91.7
Horse	86.7	89.0	87.9	93.5	93.7
Motorbike	83.6	83.7	80.7	88.4	88.9
Person	89.4	92.3	91.8	95.1	95.4
Plant	57.0	58.8	58.5	59.4	59.3
Sheep	81.5	80.5	77.4	85.4	85.3
Sofa	64.8	90.5	66.3	74.3	74.0
Train	90.4	74.0	89.1	96.8	96.7
TV Monitor	73.4	75.34	71.3	77.3	76.9
mAP	**78.6**	**79.74**	**77.38**	**83.92**	**83.95**

Table 4: The precision and mean average precision (mAP) in PASCAL VOC 2007 classification test. The NuC-MKL (3) is applied on the three linear kernels, correponding to the extracted features of CNN-M128, CNN-S and CNN-F.

set Table 4 shows $\approx 4-6\%$ improvement over the classification result of a linear SVM classifier based on a single DNN features. Finally, we analyzed the algorithm's performance under noisy conditions. The networks CNN-M, CNN-M2048 and CNN-M4096 are minor variations of CNN-M128, and the latter's kernel was included in the set kernel set of NuCMKL (3). Hence, the resulting kernels add very little information, and can be regarded as realistic noisy versions of CNN-M128 kernel. Table 4 shows that the performance of NuC-MKL does not deteriorate is the present of redundant information, showing it is disinclined to overfitting in such scenarios.

References

[1] M. Gönen and E. Alpaydin, "Multiple Kernel Learning Algorithms," *JMLR*, vol. 12, pp. 2211–2268, Feb. 2011.

[2] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *BMVC*, 2014.

[3] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," Nov. 2013.

[4] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks," Dec. 2013.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS 2012*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.