Machine Learning - Homework 4 Report

Lukas Johannes Ruettgers (2023372403)

October 25, 2023

1 Objective

The experiment's objective is to examine and evaluate the performance of a Support Vector Machine (SVM) on a multi-feature dataset. The dataset originates from https://www.kaggle.com/c/widsdatathon2020/data and contains medical information of patients at the Intensive Care Unit (ICU) of US-American hospitals. The patients x_i were classified into two groups depending on their survival s_i during their stay at the ICU.

The MLP is supposed to estimate the probability of survival ($s_i = 1$) or death ($s_i = 0$) respectively for any patient x_i represented by the feature schema of the given dataset.

2 Experiment setup

2.1 Model

Support Vector Machines arose from the problem to find an optimal hyperplane inside a solution region of linear separable, multi-dimensional data. However, this method nowadays is already applied to nonlinear input data by mapping the nonlinear data with a kernel to a feature spaces, where the classes are in the optimal case better linearly separable to avail to the fundamental strengths of support vector machines. Common kernel choices are the linear, polynomial, Gaussian and sigmoid kernel. Usually, the kernel operation is normalized by dividing through the amount of samples. This is referred to as the `auto` kernel coefficient. However, another method divides the term not only through the amount of training data, but also the variance of training data, which is referred to as the `scale` kernel coefficient. Similarly to the multilayer-perception, a regularization term can be added to the objective function which penalizes the weight magnitude in the L2 norm.

2.2 Preprocessing

The training dataset and the testing dataset comprises 5000 and 1097 samples respectively. Each sample is described by 108 features subject to binary, numeric and integer domains. While the binary values were solely converted into integers, the values of numeric and integer features were all normalized to $[0, 1]$, since their values varied largely in size and units. This was achieved by linearly rescaling each feature value x to $\frac{x - x_{min}}{x_{max} - x_{min}}$, where x_{min} and x_{max} are the maximum and minimum values of the respective feature.

2.3 Hyperparameters

For the experiment, the SVM model provided by the `sklearn.svm` package is used. For all parameters which are not subject to the experiment, the default values were taken. As kernels, the linear, Gaussian, polynomial aswell as the sigmoid kernel are examined. In particular, the relation between the degree of the polynomial kernel and the training,

validation and test error is examined. Furthermore, we investigate if the kernel coefficient that takes the variance of the data into account indeed improves the model performance. Over and above, we test how the regularization rate affects the final decision hyperplane by comparing the values $C = 1, 0.5, 0.1$.

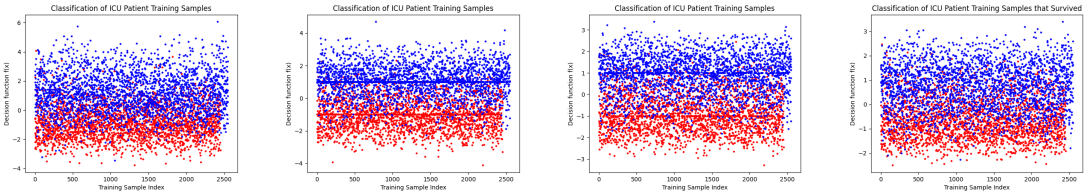
2.4 Cross Validation

Besides training the model with the entire training dataset, the bias of the models was assessed via 10-fold cross validation (CV). The training data set was split into ten equally sized intervals and the 10-fold CV error was computed as the mean accuracy of the predictions of the models trained on their respective testing interval.

3 Results

3.1 Training

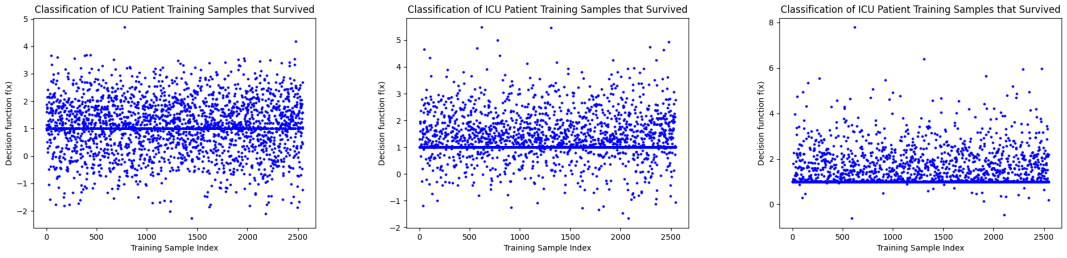
While the training error does not differ largely between the four kernels, the linear projection of the classification hyperplane takes on different shapes for different kernels, as Figure 1 indicates. This implies that the hyperplanes themselves must differ from each other. While the decision function produced by the model with the sigmoid kernel creates a greater variance in both classes (Fig. 1d), the Gaussian kernel (Fig. 1c) and the polynomial kernel (Fig. 1b) seem to yield a decision function that compresses the training data more at the margin and therefore also more support vectors. It is important to remark that for the sigmoid kernel, the performance exarcerbated significantly if the default kernel coefficient `scale` that takes the variance of the training samples into account was chosen. Instead, the `auto` kernel which only averages over the number of training samples was observed to yield smaller errors and was therefore also taken for a fairer comparison with the other kernels. Both the polynomial and the Gaussian kernel seem to yield the most discriminating decision function. However, when it comes to the polynomial kernel, the



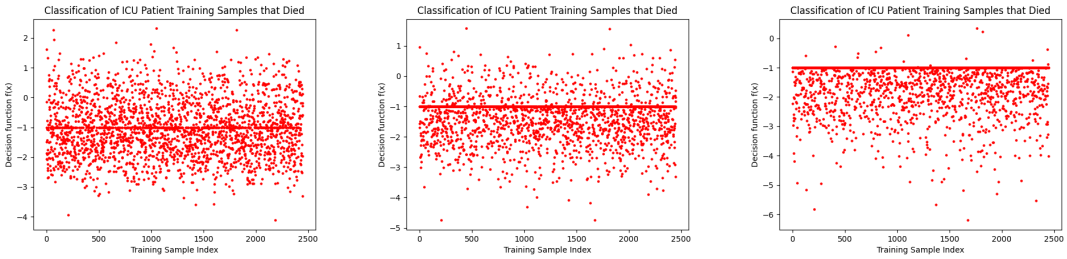
(a) Linear kernel (b) Polynomial kernel (c) Gaussian kernel (d) Sigmoid kernel

Figure 1: Comparison of the linear kernel (a), polynomial kernel with default degree 3 (b), Gaussian kernel (c) and the sigmoid kernel (d) on the decision function f. While blue dots indicate the ICU patient samples that survived, the red dots represent the ones that died.

degree of the polynomial is another hyperparameter whose influence on the generalization of the hyperplane should not be neglected. As Figure 2 demonstrates, a few additional degrees in freedom cause the polynomial kernel to yield a hyperplane that overfits the training data. With rising degrees of freedom, more vector collapse to the margin line and become support vectors. For the model with the polynomial kernel of degree 7, the accuracy on the training dataset culminates in 0.999.



(a) Poly kernel with degree 3 (b) Poly kernel with degree 5 (c) Poly kernel with degree 7

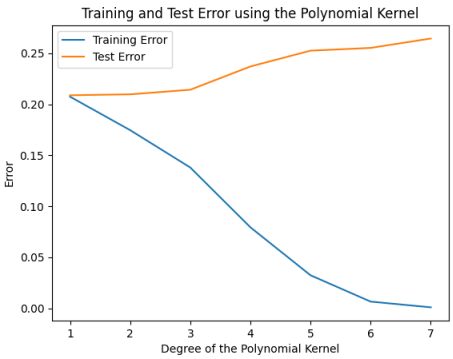


(d) Poly kernel with degree 3 (e) Poly kernel with degree 5 (f) Poly kernel with degree 7

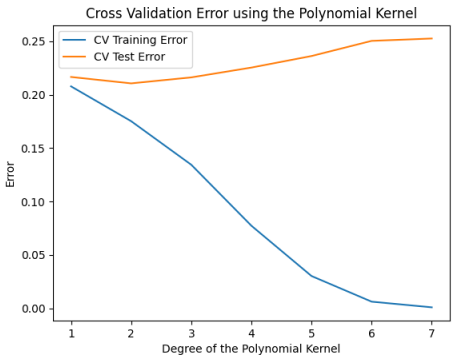
Figure 2: Comparing how the decision function f discriminates the testing data for different degrees of the polynomial kernel illustrates how the increasing degrees of freedom of the kernel lead to overfitting the training data. The first row shows the decision function values for the survived class and the second row depicts the values for the class of patients that died. For degree 7, the model discriminates the training samples almost perfectly.

3.2 Testing

Computing the training and testing error for all degrees from 1 to 7 (Fig. 3a) underscores the negative impact of the increasing degrees in freedom. While the training error monotonously decreases, the test error conversely increases. Surprisingly, the model with a linear polynomial achieves the best performance with a test accuracy of 0.7912. However, the difference to the model with a polynomial of degree 2 – which achieves a test accuracy of 0.7903 – is only small. Performing a cross validation on the training dataset leads to the same conclusion (Fig. 3b). Here, the model with polynomial degree 2 performs slightly better on the test interval, but still lacking behind with regard to the training error. A closer look on the prediction outcomes for different choices for the kernel coefficient γ and the regularization rate in Figure 4 reveals both the impact of regularization and the variance-dependent kernel coefficient `scale`. First, we observe how the models with the variance-independent kernel coefficient switches the bias of the Gaussian and polynomial kernel models from positive samples to negative ones. The opposite is the case for the sigmoid kernel. This indicates that the different results for the variance-based and the variance-independent kernel coefficient do not primarily arise from different variance properties among the two classes in the underlying data, but that rather the kernel functions profit more or less from considering the variance. While the Gaussian kernel already considers the data as a normal Gaussian distribution and therefore integrates the variance well into its kernel expression, the sigmoid kernel takes another approach which is rather spoiled by the normalization through the variance. Secondly, an increasing regularization rate seems to push the polynomial kernel towards negative predictions aswell. However, this inclination is more subtle than the one caused by the variance-independent kernel coefficient and even achieves a slightly better test performance. Considering that the regularization rate was tremendously decreased to 0.1 – which means that the regularization penalty receives *more* weight – the effect of regularization is rather neglegible.



(a) Training and test error



(b) Cross validation error

Figure 3: Development of the training error (blue), cross validation error and test error (red) for increasing degrees of the polynomial kernel. As expected, the decreasing training error comes at the cost of a increasing test error. Surprisingly, the polynomial kernel with degree 1 receives the best error scores in both training and testing.

This indicates that the model on the one side not optimizes its weight magnitude without regularization, but that the prediction shift caused by the decreasing weight magnitude as a consequence of the decreasing regularization parameter does not significantly improve the model's predictions. The test accuracy suggests that indeed the Gaussian kernel with the variance-based kernel coefficient achieves the best performance among all models.

Model Parameters			**Prediction Results**				
Kernel	RR	γ	TP	FN	FP	TN	Accuracy
Linear	1.0	-	**458**	**89**	148	402	0.7840
Poly(3)	1.0	scale	456	91	145	405	0.7849
Gaussian	1.0	scale	453	94	130	420	**0.7958**
Sigmoid	1.0	scale	377	170	140	410	0.7174
Poly(3)	1.0	auto	372	175	**108**	**442**	0.7420
Gaussian	1.0	auto	428	119	123	427	0.7794
Sigmoid	1.0	auto	416	131	115	435	0.7758
Poly(3)	1.0	scale	**458**	**89**	146	404	0.7858
Poly(3)	0.5	scale	457	90	143	407	0.7876
Poly(3)	0.1	scale	440	107	**124**	**426**	**0.7894**

Figure 4: Test results of the investigated models. Models are categorized by their kernel function (Kernel), regularization rate (RR) and their kernel coefficient γ.

To summarize, this experiment demonstrated the relation between increasing domains of freedom in the polynomial kernel function and an overfitting on the training data set. Moreover, it illustrated how the variance-based kernel coefficient matches with the Gaussian and polynomial model but spoils the sigmoid model, which in contrast to the Gaussian model does not assume a isotropic distribution of the data. Lastly, regularization on the polynomial kernel indeed affects the magnitude of the weights and adjusts the predictions, but does not further increase the model's predicitions, which indicates that the model's capactity to discriminate the data has already been exhausted.