MLHW6

Arjun Natarajan, Daniel Sealand

11 March 2020

1 Extract Features

b. When simply considering the mean of these time-sensitive measurements, we lose an understanding of the distribution of these measurements over time. It would be useful to look at the standard deviation of measurements, and thus identify the number of measurements that fell outside of this standard deviation, which would indicate an anomaly.

c. This imputation not only assumes that each feature is independent, but that each feature follows a normal distribution. It also assumes that the sampled features represent the distribution of the entire dataset. These assumptions are likely appropriate.

   We could handle missing data by sampling from the existing values instead of replacing with the mean. We could also replace the missing data with the mode of existing values.

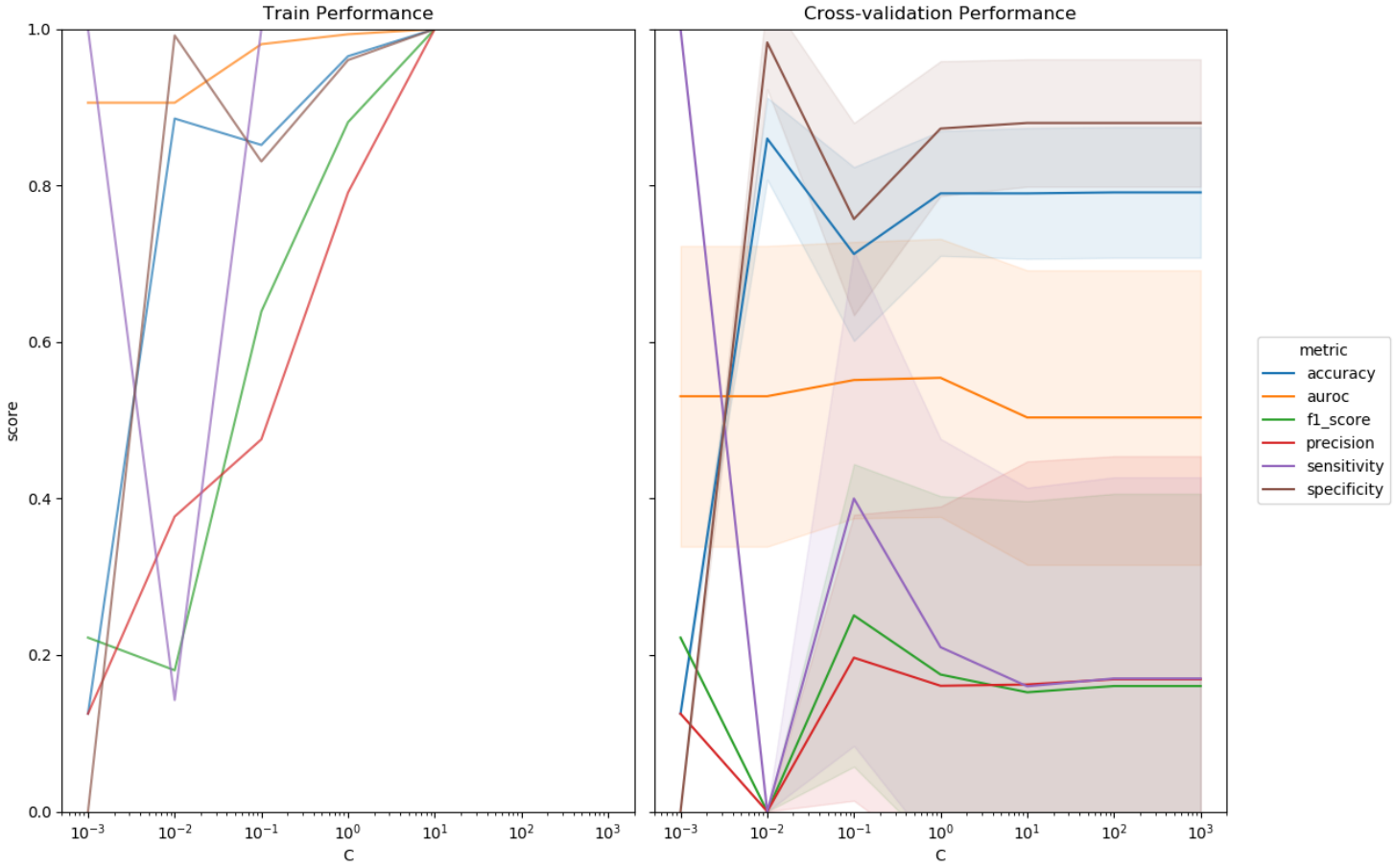
d. One advantage of scaling all of the feature values to a fixed interval is that we have standardization across all features. Specifically, when considering SVM, features with larger values will tend to dominate the weights of the resulting perceptron as a result of the formulation of the update step in the perceptron algorithm. Thus, fixing all features to a standardized scale will ensure that all features have equal weight.

   One limitation of this scaling approach is that every feature now has equal importance and we cannot emphasize features that may be more indicative of a patient's health. One way that we may handle scaling, instead then, is to assign larger scales to features that we may believe are more important and to assign smaller scales to those that we believe are less important based on domain knowledge.

2 Practice Pipelines

b. So, we can see the plots below. Generally, we see a better performance in both the training and cross-validation tests, regardless of metric, as C increases. There is some noise to this data such as specificity taking a large

dip when C increases from 10^{-3} to 10^{-2}, but otherwise this general trend holds. Recalling that C is the regularization factor, we notice that larger C should lead to larger regularization and therefore better generalization and smaller C should lead to less generalization. It then, makes sense that as we increase C we see an improvement in our cross-validation performance, up to some threshold as with larger C our model is better generalized and performs better on the held-out test set. This also explains why we see dips when increasing C in our training performance graph. Larger generalization penalizes overfitting to the training data, and so we expect to see a worse training performance but better cross-validation performance as C increases.



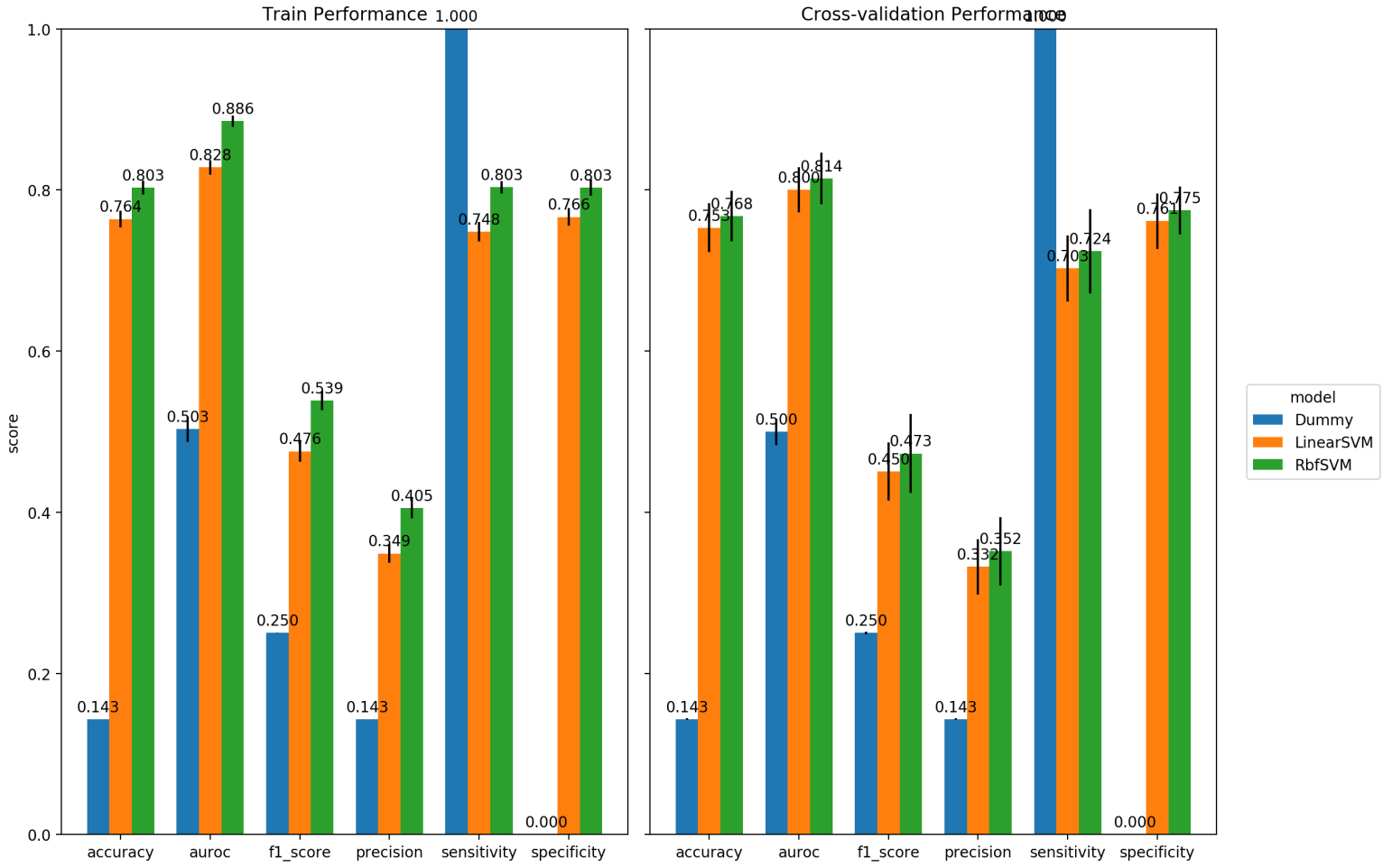
c. We observe the table.

metric	C	train	test
accuracy	0.01	0.891	0.858
AUROC	1.0	0.993	0.576
F1_Score	0.1	0.643	0.263
Precision	0.1	0.479	0.204
Sensitivity	0.001	1.0	1.0
Specificity	0.001	0.991	0.98

3 Tune and Compare Models through Cross-Validation

b. We include the figure below. It is important to look at training scores because this indicates if our model is under or overfitting. For example, we can see that the Dummy classifier has a very low training accuracy, indicating that it is underfitting to our data. If it was able to learn a more complex model, it would have, and therefore would have had a higher training accuracy.

We see that across all metrics, except sensitivity, the dummy classifier is much worse than both SVM classifier. This makes sense, as if the classifier always classifies as positive, then the sensitivity will always be 1. Additionally, we see across all metrics that the RBF Support Vector Machine is better than the Linear Support Vector Machine. This makes sense, as using a kernel allows the machine to learn a non-linear decision boundary, allowing for a more complicated model. We see that the RBF SVM is able to achieve a balance of high sensitivity and specificity, leading to a relatively high area under the ROC curve, of 08.14 on the cross-validated data.



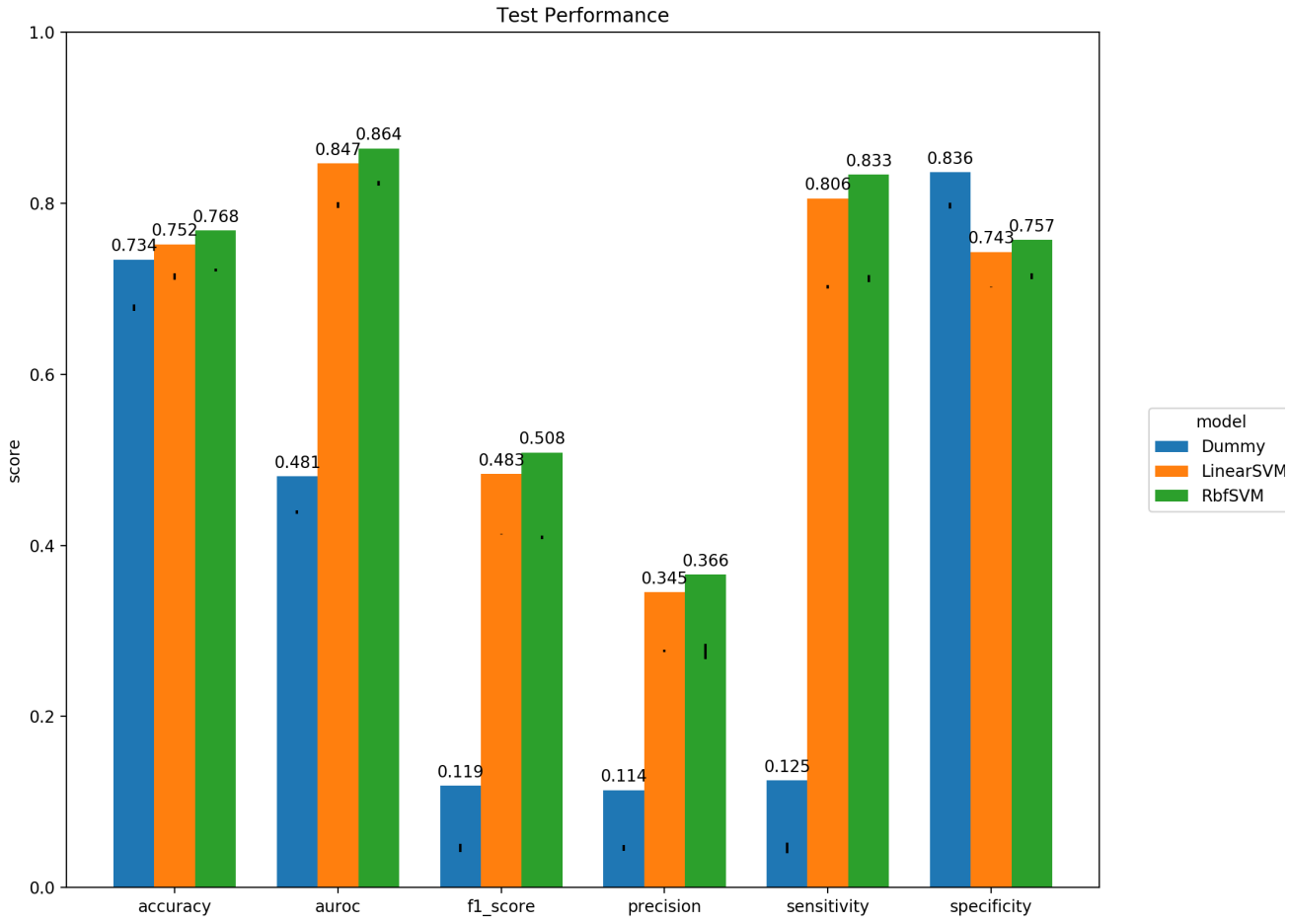
c. We consider the area under the ROC curve over other metrics as when considering the AUROC metric, we are able to weigh and consider the different misclassification costs that come with this specific dataset.

Classifier	C	γ
Dummy	-	-
Linear SVM	0.1	-
RBF SVM	1	0.1

4 Evaluate Performance and Gain Insight

c. So, again we see that on our test data the Dummy classifier tends to have the lowest score for most metrics, and the RBF Support Vector Machine has the

highest score across all metrics. We see that the dummy stratified classifier has a surprisingly high accuracy that is close to that of both other classifiers and has a higher specificity, but is far worse in every other metric. The Linear SVM and the RBF SVM are quite close in every metric, although the RBF SVM is likely slightly better in each metric due to its increased complexity. For the RBF SVM we observe an AUROC of 0.864, which is relatively close to 1.



d. The coefficients of the learned classifier, θ, indicate which features are more important. Larger absolute values for θ_i indicate that feature i is more important, and smaller absolute values indicate that a feature is less important. If a feature i has a negative value for θ_i it indicates a decreased risk and if it has a positive value it indicates an increased risk.

rank	increased risk feature	coefficient	decreased risk feature	coefficient
1	mean_BUN	1.677	mean_GCS	-1.1915
2	Age	0.8927	mean_Temp	-1.0538
3	mean_HR	0.7091	mean_Na	-0.7075
4	mean_Glucose	0.5751	Weight	-0.5393
5	mean_Lactate	0.5340	mean_K	-0.4684

e. Let us consider, mean_BUN, which is the mean Blood Urea Nitrogen concentration. It makes sense that this is a strong feature for increased risk, as a higher BUN level means that the patient is more likely to have kidney or liver failure.

f. It is not possible to use learned coefficients to determine feature importance for a non-linear kernel. This is because the non-linear kernel has a feature mapping that inherently changes the structure of the data. Therefore, the coefficients that are learned to not have a direct correlation to the features, as they do with a linear kernel, and so we cannot determine feature importance in the same way.

5 Explain to a Client

Our goal with this project was to develop a machine learning model that consistently and sufficiently predicts whether or not a patient will survive while in intensive care. We used real patient data, collected from Beth Israel Deaconess Medical Center in Boston, with 42 parameters based on measurements that were taken while the patient was in intensive care, to teach our model. Based on the measurements of the 42 parameters, or features, and the resulting outcomes for the patients in this dataset, our model learns the relative importance of each feature. In this way, for a certain combination of these features, our model is able to predict whether or not that combination will lead to an in-hospital death, or not.

We were able to conclude that the most 5 important features that were likely to lead to an increased risk of in-hospital death were a high Blood Urea Nitrogen level, old age, high heart rate, high blood sugar level, and high lactic acid levels. The 5 most important features that were likely to lead to a decreased risk of in-hospital death were a high Glasgow Coma Score, which measures the patient's consciousness, a high temperature, a high blood sodium level, high weight, and high blood potassium level. Our model is able to make predictions with an accuracy of 76.8% and has a sensitivity of 83.3%. This means that of all the patients who actually will die in the hospital, our model only catches 83.3% of these, and therefore misses 16.7% of positive cases. Therefore, we recommend for doctors to use this model as a baseline assessment as it performs relatively well, but to not solely consult our model as it does miss a considerable number of cases where the patient would, in fact, die in the hospital. In examining

patients, we recommend considering the 10 measurements previously listed, as these seem to be the most indicative of a patient's likely of survival.