# Prediction of Patient Outcomes after Renal Replacement Therapy in the Intensive Care Unit

Siegfried Horschig
Hasso Plattner Institute (HPI)
Enterprise Platform and Integration Concepts
Potsdam, Germany
Email: siegfried.horschig@student.hpi.de

***Abstract*—In order to compensate impairments of the renal system in the human body, artificial methods in the form of Renal Replacement Therapy (RRT), called dialysis, have to be introduced. Many parameters of the dialysis can be adjusted and the outcome of the procedure may change with different patient characteristics. In this paper, we introduce a clinical decision support system to predict the effect of a given dialysis on a patient while in the Intensive Care Unit (ICU).**
**For this purpose, we employ two kinds of machine learning models: Bayesian Rule Lists (BRL) and Deep Neural Network (DNN). Although the DNN may provide better accuracy, its decision making is not easily interpretable for humans. For this reason, we use mimic learning as a method to make the DNN interpretable.**
**Results show us that the DNN outperforms our BRL classifier as expected, but by a rather small margin. For the mimic learning process, we used a Bayesian Ridge Regression model. Even though the regression model performs worse when training as a mimic model as opposed to directly on the data, it provides some insight into the inner workings of the DNN.**

## I. Introduction

The renal system in the human body has the purpose to eliminate wastes from the body and control levels of certain substances in the blood. If this system is impaired, for example due to Acute Kidney Injury (AKI), artificial methods in the form of RRT have to be introduced, more commonly known under the term of dialysis.
There are different options for dialysis available. One example is the hemodialysis, where the patient's blood is pumped through a dialyzer, inside of which is a liquid called dialysate. This liquid's composition determines which substances should be filtered out of the blood. The dialyzer separates the blood and the dialysate though a partially permeable membrane, allowing for the filtering of the blood through osmosis. Another example for the dialysis is the peritoneal dialysis, which uses the peritoneal cavity inside the patient as a container for the dialysate.

The dialysis outcomes are highly dependent on both the patients characteristics and the parameters as well as the type of the dialysis. So, usually, patients undergoing the peritoneal dialysis experience lesser health issues related to the dialysis than those undergoing hemodialysis, as there is less pressure on the circulatory system [1]. On the other hand, the hemodialysis is more efficient in such a way that it needs less time for the same amount of filtration.
Especially the hemodialysis is a costly process which needs specialized equipment and therefore has many parameters to be tuned. These include, but are not limited to the duration of the process, the filtration rate and flow rates of the blood and dialysate. There is no consensus about the optimal duration for a dialysis procedure. Studies exist proposing both longer durations [2] as well as shorter procedures [3]. It stands to reason that a Clinical Decision Support System (CDSS) is needed in order to predict optimal parameters based on patients' characteristics.

Aside from usual criteria like accuracy or recall, when employing a machine learning model in the medical context one especially important factor is the interpretability of the model [4]. This is due to the fact that the doctors want to make decisions based on those predictions and take full responsibility for the consequences, so they have to validate the decision making process. Oftentimes, models just show correlations between various parameters, and those correlations have to be manually checked for causal relations. Additionally, the European Union introduced regulations (taking effect 2018) that give consumers in any sector a "right to explanation" [5]. Essentially, this means that in any decision making process that is done by machines, be it a credit application or a diagnosis, the individual has the right to access meaningful information about the logic involved.

This way, we can roughly separate machine learning algorithms in two categories: interpretable and non-interpretable. One example for interpretable models are BRL [6]. By presenting itself as *if...then...else* lists, it is easy for humans to comprehend both the decision making and the individual influence of each parameter on the outcome.
DNN on the other hand are more accurate, but non-interpretable. This is because the weights of the nodes in the hidden layers is everything they expose to the outside. Due to

the fact that different loss and activation functions take effect when updating those weights, the abstraction to the original input data is just too large for a human to grasp.
At this point, we have a tradeoff between interpretability and accuracy. In order to overcome this tradeoff, we employ a strategy called mimic learning [7]. By training an interpretable model on the predictions of the more accurate, non-interpretable model, we gain insight into its decision process and can therefore make the non-interpretable model interpretable.

The main purpose of this paper is to develop a CDSS to determine patient-specific outcomes after RRT in the ICU. We evaluate the performance and interpretability of two different models, BRL as the interpretable one and DNN as the non-interpretable one. Afterwards, we employ mimic learning to overcome the tradeoff between accuracy and interpretability and give some insight into the decision making process of the DNN. We then ask an expert in the field of renal medicine for further evaluation and validation of the models.

After evaluating related work in section II, we present the tools, data and models used in section III. Thereafter, we present the results in section VII and discuss them in section VIII.

## II. Related Work

In the field of prediction concerning the renal system, a lot of work has already been done. For example, Schwenger et al. researched the mortality of patients undergoing dialysis with different procedures, thus making patient mortality a fitting target for prediction by our model [8]. Likewise, Ricci et al. researched dialysis duration and showed impact of blood values, making them suitable features to train the model upon [9].
When it comes to model selection, Baby et al. found a bayes algorithm suitable [10], while Greco et al. proposed decision trees [11]. The issue with those models is that they lack accuracy when compared to more advanced models. Vijayarani et al. and Sinha et al. used Support Vector Machine (SVM) for the prediction in the renal context and achieved satisfying results, but lack interpretablity [12] [13]. In an equal manner, Lakshmi et al. compared the three models regression, random forest and artificial neural networks and proposed the latter for better performance and accuracy, while still lacking interpretability [14].
Lipton et al. discussed the term *interpretability* in the context of machine learning [15]. Particular focus was on the definition of the decision boundaries as well as influence of specific features, which is important for our evaluation. Katuwal et al. worked on achieving interpretability of machine learning models in precision medicine and used the locally-interpretable model- agnostic explanations (LIME) technique, achieving accuracies of 80% [4]. Ultimately, we decided on using the mimic learning technique as proposed by Che et al. for usage in the context of patients in the ICU [7]. While Che et al. used Gradient Boosting Trees as mimic learning model, we wanted to employ a different model and compare the results, especially when transferred into the renal context.

In the context of this work, we try to unify both the prediction of patient outcomes in the renal context as well as the interpretability of the models used. We compare the results of mimic learning as described by Che et al. with the results of an interpretable classifier, BRL as proposed by Letham et al [7] [6].

## III. Tools

For quick prototyping, we used *RapidMiner* [16], allowing us to prepare data, develop and cross-validate first models. For the final models, we implemented them with the *scikit-learn* library [17] in Python.
The Data we used was provided by the *MIMICIII* dataset [18] stored in a *HANA* database [19].

## IV. Data

The *MIMICIII* dataset contains hospital admission data for patient collected over an eleven-year period in a Boston hospital. As seen in figure IV, out of the approximately 46,000 patients present in the dataset, we extracted 925 relevant patients for us, totaling to approximately 3,000 dialysis procedures we can train our models upon.

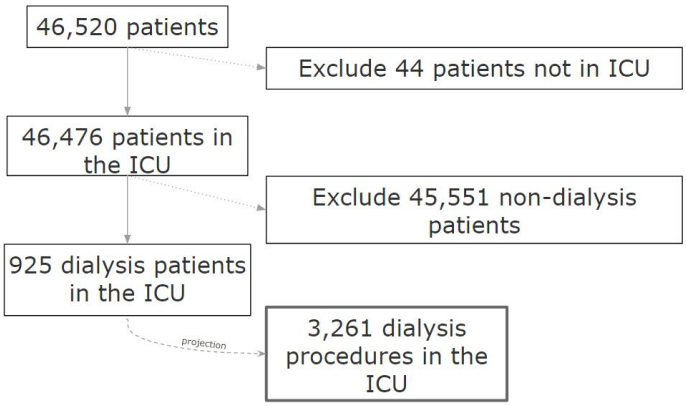


Fig. 1. Cohort selection of the dialysis procedures based on the patients.

For each procedure, we have about 80 features defining it. Those include patient demographics such as age or BMI, dialysis parameters such as the duration of the procedure, comorbidities as well as lab values. Additionally, we include patient vitals and measurements on how ”well” the patient is doing such as 90-day mortality, the number of days he/she spent without mechanical ventilation and in the ICU in general.

### A. Missing Data

Due to the manually curated nature of the *MIMICIII* dataset, aside from the inconsistencies within, a lot of data is missing. *TODO: Retrieve numbers!!!* As the scikit-learn models need a complete dataset for training, we decided to impute the

missing values. We evaluated both mean/median imputation and K-Nearest-Neighbour imputation and decided on the latter, giving us an increase of about 2% in accuracy.

## V. Models

In order to compare performance of both interpretable and non-interpretable models, we trained one model for each category. In the following section, we describe the models and strategies used as well as the parameters chosen for training.

### A. Interpretable - Bayesian Rule Lists

For the interpretable model, we chose the existing Python 2 implementation of Bayesian Rule Lists (BRL) as described in [6]. They position themselves as a direct competitor to decision tree approaches, as they have a high accuracy for classification while still being easy to read for humans.
This algorithm tries to find *if...then...else* statements over a dataset with the important criteria of them being sparse for better human readability. It achieves this goal by mining antecendents from the data and afterwards computing the posterior distribution over the antecedent lists.
The current implementation of BRL has the shortcoming of only being able to classify binary targets. Thus, we have to adjust the target features accordingly.

*1) Parameters:* The sole adjustable parameter in the used implementation is the maximum number of iterations. Multiple adjustments to this parameter - including changes by factor 10 - did not result in a significant change, neither for the runtime nor for the accuracy. For the evaluation, we chose a value of 50,000 maximum iterations.

### B. Non-Interpretable - Deep Neural Network

As non-interpretable model, we chose the powerful and widely used Deep Neural Network (DNN). Specifically, the scikit-learn implementation Multi-Layer Perceptron (MLP). Just as other implementations, this network consists of multiple layers of so-called "neurons": one input layer with as many neurons as there are inputs, one output layer with the size of the number of target features and hidden layers varying in size and quantity. The weights of each neuron is updated after each iteration of training, optimizing the log-loss function.
Scikit-learn offers implementations for both regression and classification tasks.

*1) Parameters:* Neural networks have a wealth of parameters to be adjusted. Doing a grid search over some of the parameters, we found the default ones from the library to perform the best.
This means the learning rate - determining the speed and accuracy of convergence - is set to 0.001. The activation function, determining the output of the neurons in the hidden layer, is the rectifier linear unit "relu". The network consists of one hidden layer with 100 neurons. The maximum number of iterations before convergence is set to 200.

### C. Interpretablity Approach - Mimic Learning

The large amount of neurons in the DNN and the many parameters influencing their weights and output make it very difficult - if not impossible - for a human to understand the influence of each feature on the training.
That is why we wanted to give some insight into the workings of the DNN by applying a method called *mimic learning*. Building upon the approach of [7], we train an interpretable model - the so-called "mimic model" - on the outputs of the non-interpretable model. The mimic model takes the same input features as the non-interpretable model. For classification tasks, the output of the non-interpretable model are called "soft scores", because as they are probabilites, they are continuous variables, coming close to the actual target features. In theory, the training of the mimic model on the soft scores allows to create a much smaller, thus understandable, faster but still equally accurate model. Using the principle of knowledge distillation, it is even possible for the mimic model to perform better than the non-interpretable model.
In our case, the DNN is the non-interpretable model. For the mimic model, we need a model which is able to predict continuous scores. We decided on Bayesian Ridge Regression.

*1) Bayesian Ridge Regression:* The Bayesian Ridge Regression, like common linear regression, tries to find coefficients for each input feature so that they map to the target feature, minimizing loss. In addition to common linear regression, it includes regularization parameters to control the growth of the coefficients. Therefore, this model is less prone to overfitting while still being as fast as linear regression.

## VI. Performance Metrics

In order to compare the models' performance to each other, we have to specify performance metrics, in this case for binary classification. All of them are working with the terms of true and false positives and negatives. The terms *positives* and *negatives* refer to the prediction of the model, while the terms of *true* and *false* refer to the fact if the prediction of the model was correct. Additionally, all targets that are true are *relevant* elements.
**Sensitivity / Recall** specifies the number of relevant items that have been selected. This means the number of true positives divided by the number of all true targets.
**Precision** specifies the number of relevant instances in the result set. This means, out of all as positively classified targets, how many have actually been positive.
**Specificity** specifies the number of correctly classified negative instances.
The **Diagnostic Odds Ratio** (DOR) is defined as the ratio of the odds of the prediction being positive if the target is positive against the odds that the prediction is positive if the target is negative. It range from zero to infinity, with a DOR of 1 meaning that the prediction is equally likely to give a positive prediction no matter the true status. A higher value indicates a better prediction.
The **Receiver Operating Characteristics curve** (ROC) plots the true positive rate against the false positive rate at various

thresholds. The important measure for this plot is the **Area Under the Curve** (AUC), a value ranging from zero to one, with an AUC of 0.5 describing a random classifier. Higher values indicate better classification performance.

## VII. Results

In the following section, we want to compare the performance of our interpretable model, the BRL, and our non-interpretable model, the DNN. Although there are continuous values for our target variables in the dataset, we had to transform them into a binary format in order for the BRL classifier to work. This is why the target variable for this sample is mortality within 90 days of the procedure.

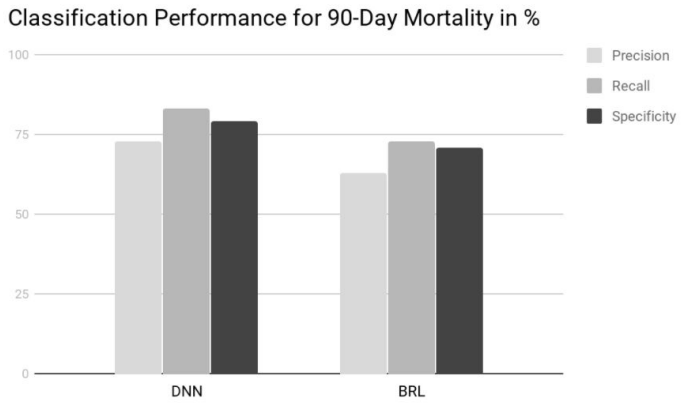


Fig. 2. Performance comparison of the classifiers when predicting 90-day mortality.

Figure VII shows general performance of the classifiers. We can see that, as expected, the DNN outperforms the BRL classifier in every performance criteria. The difference is about ten percent for every performance measure.

IF **GFR_48_B : -inf to 6** AND **ELIXHAUSER_VANWALRAVEN : -inf to 11.5** THEN
probability of DIED_90DAYS: **3.7%** (1.9%-6.2%)

ELSE IF **LACTATE : 4.2 to inf** AND **BICARBONATE : -inf to 21.9** THEN
probability of DIED_90DAYS: **77.4%** (71.2%-83.0%)

ELSE IF **LENGTH_OF_STAY_HOURS : 1030 to 1475** AND **OBESITY : All** THEN
probability of DIED_90DAYS: **98.5%** (95.9%-99.8%)

ELSE ....

Fig. 3. Excerpt of the rules from the BRL classifier when prediction 90-day mortality.

Due to the interpretable nature of the BRL, we can look at the importance of single features. Figure VII shows the influence of some features and their values on the prediction of 90-day mortality. For example, we can see that a lower *Elixhauser-van Walraven* score - a score combining several comorbidities - along with a lower *Glomerular Filtration Rate* two days before the procedure leads to a lower possibility of death within 90 days.

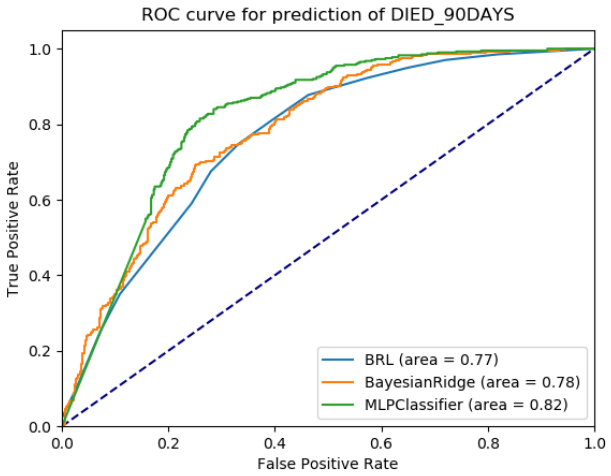


Fig. 4. ROC curve comparing the classifiers for prediction of 90-day mortality.

If we want to see how the DNN actually makes its decisions, we have to apply the mimic learning strategy. First, we have to evaluate if the performance of the mimic model is reasonably good when only being trained on the outputs of the DNN. Figure VII presents the performance of the mimic model - our Bayesian Ridge Regression - relative to the other classifiers. We can see that, while the regression is still worse than the DNN, it performs better than the BRL, if only by a small margin.

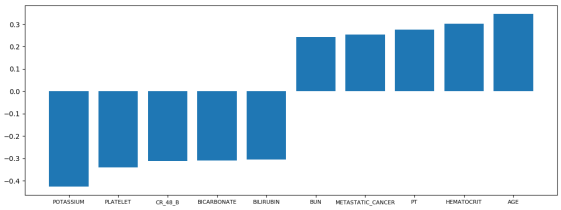


Fig. 5. Coefficients of the most important features for the Bayesian Ridge Regression trained as mimic model when predicting for a true outcome of 90-day mortality.

If we now look into the most important coefficients of the regression in figure VII, we can see the influence of single features on a positive prediction of 90-day mortality. So, for example, the higher the rightmost feature, the age of the patient, the higher is the probability of the patient to die within 90 days. On the other hand side, the higher the leftmost feature, the potassium value in the blood of the patient, the less likely the patient is to die within 90 days.

## VIII. Discussion

The classification results of the models can be improved. While providing a sufficiently good performance for researching the interpretability, their confidence has to be higher for the practical use in the medical context.
Looking at the influence of single features, the results are mixed. While some of them make sense intuitively, like a higher age correlating with a higher chance of mortality, some just appear to be coincidental correlations. For example, there is most probably not a linear correlation between certain blood values and the mortality, as either too low or too large values can be influencing the patient's health negatively. In the case of potassium, values between 3.5 and 5.5 µg/l would be fine values, but higher or lower values can lead to heart arrhythmia.
The same observation holds true for the output of the BRL. While obesity and a longer stay in the hospital are most likely indeed factors for a higher mortality, other associations with the lactate value in the blood may be random coincidence. Especially because higher lactate values usually lead to other complications, so the upper bound of "infinity" is very rough. In order to refine and validate those assumptions, it is necessary to go further with the data analysis. Finding actual upper and lower bounds in the dataset can provide some insight to the actual values the model considers when making predictions.

By training the regression as a mimic model, we can make assumptions on how the DNN *may* make its decisions. There still is a gap between the performance of the regression model and the DNN, which makes it difficult to say how close those coefficients are to the actual influence of features in the DNN. The mimic model performs worse when being trained on the outputs of the DNN as opposed to being trained on the real targets, because it most probably also learns the errors of the DNN. This can be a resolvable issue by improving the performance of the DNN through further parameter tuning and data preparation.

## IX. Conclusion

In this paper, we compared the performance of different models when being used in the prediction in the renal context. An important part is the interpretability of said models to validate the decision making. We used a mimic learning approach to make a Deep Neural Network interpretable and compared this output to that of the interpretable model, the Bayesian Rule Lists. Preliminary results for prediction of 90-day mortality enable the exploration of interpretability, showing the influence of single features. While some of the features' influence can intuitively be classified as correct, others' influence needs further data examination to prove or disprove.

Future work includes more elaborate use of the data. This means the inclusion of more features, more elaborate imputation and collection of information about the patients. Additionally, more target features can be added to the existing experiments. The binary classification limitation of the current implementation of BRL can be overcome by using a more advanced algorithm proposed by Yang [20]. When a higher precision is achieved, interpretable models can be used as a Clinical Decision Support System, allowing the doctors to validate the decisions and giving patients insight into their treatment.

## References


[1] S. S. Fenton *et al.*, "Hemodialysis versus peritoneal dialysis: A comparison of adjusted mortality rates," *American Journal of Kidney Diseases*, vol. 30, no. 3, pp. 334–342, 1997.

[2] P. Jungers *et al.*, "Longer duration of predialysis nephrological care is associated with improved long-term survival of dialysis patients," *Nephrology Dialysis Transplantation*, vol. 16, no. 12, pp. 2357–2364, 2001.

[3] C. Ronco *et al.*, "Technical and clinical evaluation of different short, highly efficient dialysis techniques." Karger Publishers, 1988, vol. 61, pp. 46–68.

[4] G. J. Katuwal and R. Chen, "Machine Learning Model Interpretability for Precision Medicine," 2016.

[5] B. Goodman and S. Flaxman, "European union regulations on algorithmic decision-making and a" right to explanation"," *arXiv preprint arXiv:1606.08813*, 2016.

[6] B. Letham *et al.*, "Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model," *Annals of Applied Statistics*, vol. 9, no. 3, pp. 1350–1371, 2015.

[7] Z. Che *et al.*, "Interpretable Deep Models for ICU Outcome Prediction." *AMIA Annual Symposium proceedings. AMIA Symposium*, vol. 2016, pp. 371–380, 2016.

[8] V. Schwenger *et al.*, "Sustained low efficiency dialysis using a single-pass batch system in acute kidney injury - a randomized interventional trial: the Renal Replacement Therapy Study in Intensive Care Unit Patients," *Critical Care*, vol. 16, no. 4, p. R140, 2012.

[9] Z. Ricci, R. Bellomo, and C. Ronco, "Dose of dialysis in acute renal failure." *Clinical journal of the American Society of Nephrology : CJASN*, vol. 1, no. 3, pp. 380–388, 2006.

[10] P. S. Baby and P. Vital, "Statistical Analysis and Predicting Kidney Diseases using Machine Learning Algorithms," *International Journal of Engineering Research & Technology*, vol. 4, no. 07, pp. 206–210, 2015.

[11] R. Greco *et al.*, "Decisional Trees in Renal Transplant Follow-up," *Transplantation Proceedings*, vol. 42, no. 4, pp. 1134–1136, may 2010.

[12] S. Vijayarani and S. Dhayanand, "Kidney Disease Prediction Using SVM and ANN," *International Journal of Computing and Business Research (IJCBR)*, vol. 6, no. 2, 2015.

[13] P. Sinha and P. Sinha, "Comparative Study of Chronic Kidney Disease Prediction using KNN and SVM," vol. 4, no. 12, pp. 608–612, 2015.

[14] K. R. Lakshmi, Y. Nagesh, and M. Veerakrishna, "Performance Comparison of Three Data Mining Techniques for Predicting Kidney Dialysis Survivability," *International Journal of Advances in Engineering & Technology*, vol. 7, no. 1, pp. 242–254, 2014.

[15] Z. C. Lipton, "The Mythos of Model Interpretability," no. Whi, 2016.

[16] M. Hofmann and R. Klinkenberg, *RapidMiner: Data mining use cases and business analytics applications*. CRC Press, 2013.

[17] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[18] A. E. W. Johnson *et al.*, "MIMIC-III, a freely accessible critical care database," *Scientific data*, vol. 3, 2016.

[19] F. Färber *et al.*, "SAP HANA database," *ACM SIGMOD Record*, 2012.

[20] H. Yang, C. Rudin, and M. Seltzer, "Scalable Bayesian Rule Lists," *arXiv preprint arXiv:1602.08610*, 2016.