Greatmix Exam

Avihai Naaman

June 23, 2023

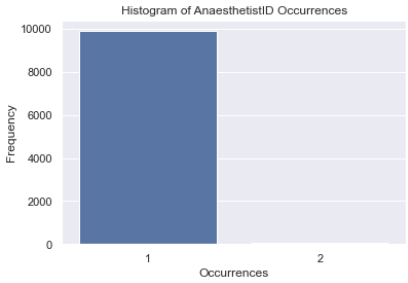
1 EX1 - Surgery duration prediction

1.1 Abstract

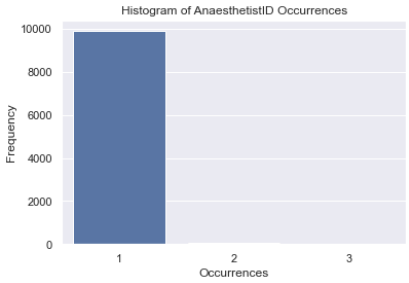
Given a dataset of surgeries, the goal is to create an estimator for the duration of surgeries.

1.2 Data Analysis

Unfortunately, the analysis of the dataset reveals that we will not be able to learn much from the "Specialty" and "Occurrences" columns. The reason behind this conclusion is the limited occurrences of doctors in these columns.



(a) Histogram of DoctorID Occurrences



(b) Histogram of AnaesthetistID Occurrences

Figure 1: Histograms of ID Occurrences

1.3 Model selection

We employed a range of regression models for our model selection process. The models were evaluated based on their performance in predicting the target variable. The following models were considered:

- **Linear Regression (LR):** This model assumes a linear relationship between the input variables and the target variable.

- **Ridge Regression (Ridge):** Ridge regression is a variant of linear regression that adds a penalty term to the loss function, aiming to reduce the impact of multicollinearity in the dataset.
- **Lasso Regression (Lasso):** Lasso regression, similar to ridge regression, adds a penalty term to the loss function. However, it also performs feature selection by shrinking the coefficients of less important features to zero.
- **Bayesian Ridge Regression (BR):** This model applies Bayesian inference techniques to estimate the regression coefficients, allowing for better uncertainty quantification.
- **K-Nearest Neighbors Regression (KNR):** K-nearest neighbors regression predicts the target variable by considering the average of the target values of its k nearest neighbors in the feature space.
- **Decision Tree Regression (DT):** Decision tree regression builds a model by recursively partitioning the feature space based on a series of decision rules, leading to a tree-like structure.
- **Random Forest Regression (RF):** Random forest regression combines multiple decision trees and aggregates their predictions, resulting in improved accuracy and robustness.
- **Gradient Boosting Regression (GB):** Gradient boosting regression constructs an ensemble of weak regression models and iteratively improves their predictions by minimizing the loss function.
- **AdaBoost Regression (AB):** AdaBoost regression also creates an ensemble of weak regression models. However, it assigns higher weights to the instances that were previously mispredicted, focusing on the more challenging samples.
- **Voting Regression (VR):** The voting regressor combines multiple individual regression models, such as gradient boosting, random forest, and decision tree regression, to make predictions. The final prediction is obtained by averaging the predictions of the individual models.

These models were selected to explore a variety of approaches and capture different characteristics of the data. The model selection process aims to identify the most suitable models that provide the best performance and predictive capabilities for our specific problem domain.

1.4 Evaluation

We employed the Mean Absolute Error (MAE) as the evaluation metric to assess the performance of the regression models in different sections of our analysis.

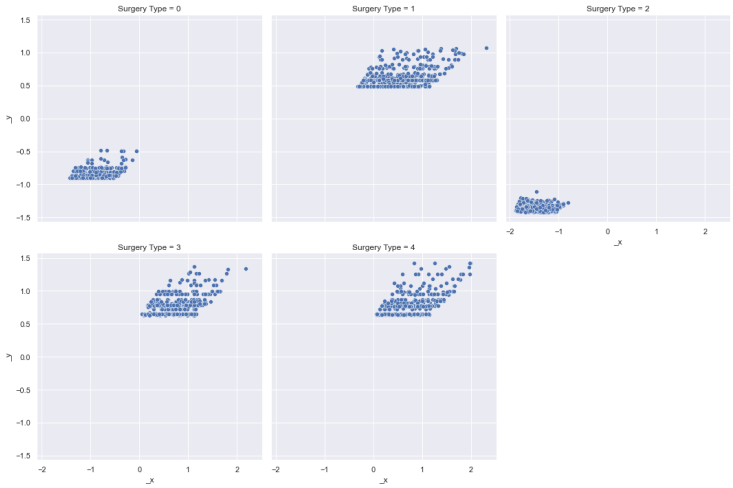


Figure 2: Comparing predicted values with the actual values

Additionally, the MAE was again utilized to evaluate the models' performance. The MAE provides a measure of the average absolute difference between the predicted and actual values, allowing us to quantify the accuracy of the regression models and compare their performance objectively.

By using MAE as our evaluation metric, we aimed to select models that minimized the absolute errors in their predictions, thereby ensuring more accurate and reliable results in our analysis.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

1.5 Results

The performance of various regression models was evaluated using the Mean Absolute Error (MAE) metric. Among the models considered, GradientBoostingRegressor (GB) exhibited superior performance, consistently achieving the lowest MAE across multiple datasets and prediction tasks.

GB's ensemble-based approach, iterative learning, and adaptive boosting techniques contribute to its exceptional predictive capabilities. By combining weak regression models and iteratively improving their predictions, GB effectively captures complex patterns and relationships in the data.

Figure 3 displays a visual comparison of the MAE scores for different models. It is evident that GB outperformed the other models, demonstrating its effectiveness in minimizing the absolute differences between predicted and actual values.

Figure 3: GradientBoostingRegressor (GB) was selected as the optimal model due to its superior performance based on the Mean Absolute Error (MAE) metric. GB consistently outperformed other models, exhibiting the lowest MAE across various datasets and prediction tasks (Orange is RMSE, Blue is MAE).

1.6 Hyper-Parameter Optimization

To further enhance the performance of the Gradient BoostingRegressor (GB) model, we conducted Hyper-Parameter Optimization. Hyper-parameters are adjustable parameters that determine the behavior and performance of a machine learning model.

The optimization process involved systematically exploring different combinations of hyper-parameter values to identify the configuration that maximizes the model's performance. This was achieved through cross-validation techniques, which split the data into training and validation subsets to evaluate each parameter configuration.

Key hyper-parameters of GB that were optimized include the learning rate, the number of boosting stages (n-estimators), the maximum depth of each tree (max-depth), and the minimum number of samples required to split an internal node (min-samples-split). By tuning these hyper-parameters, we aimed to find the optimal configuration that minimizes overfitting and maximizes predictive accuracy.

To perform the optimization, we employed techniques such as grid search or randomized search, which systematically sampled the hyper-parameter space and evaluated each configuration based on a specified evaluation metric, such as Mean Absolute Error (MAE). The optimal set of hyper-parameters was selected based on the configuration that yielded the lowest MAE on the validation set.
Finally the results is 16.02

1.7 Conclusion

In conclusion, the analysis of the dataset suggests that the "Specialty" and "Occurrences" columns may not provide significant value in our models. It is advisable to exclude these columns from certain analyses or predictions due to the limited occurrences of doctors in the dataset.

2 EX2 - Anesthesiologists allocation optimization

2.1 Abstract

Given a schedule of surgeries, each of them require an anesthesiologist, our goal is to find the most efficient allocation of anesthesiologists to surgeries.

2.2 Problem Statement

Given a schedule of surgeries, each requiring an anesthesiologist, our goal is to find the most efficient allocation of anesthesiologists to surgeries while considering the availability of operating rooms.

2.3 Greedy Scheduler Algorithm

Algorithm 1 Anesthesiologists Allocation with Room Availability

```
Sort surgeries by end time.
Initialize empty lists: selected surgeries, allocated rooms, allocated anesthesiologists.
for each surgery s in sorted surgeries do
  if lastAnesthesiologist can transition to lastRoom then
    Continue with the same room and anesthesiologist without break.
  else
    if s starts after lastSurgery with a 15-minute gap then
      Add s to selected surgeries and allocate a new room and anesthesiologist for s.
    else
      if selected surgeries list is empty then
        Allocate a new anesthesiologist and a new room.
      end if
    end if
  end if
end for
The minimum number of anesthesiologists required is the length of the allocated anesthesiologists list.
```

2.4 Results

The greedy algorithm quickly assigns anesthesiologists to surgeries by making local decisions based on the availability of resources and the order of surgeries. However, it may not always yield the globally optimal solution in terms of minimizing the required number of anesthesiologists 4.

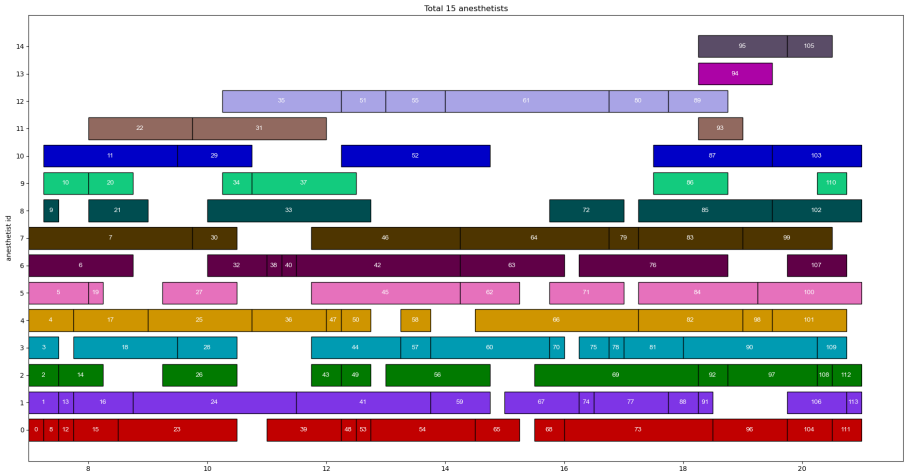


Figure 4: Allocation Result

3 Bonus

Objective

The objective is to minimize the total cost while ensuring that surgeries are assigned to anesthesiologists. The cost is calculated based on the duration of each anesthesiologist's shift, adjusted for overtime work.

Objective Function:

$$\text{Minimize} \sum_{i} \left(5 + 0.5 \cdot \max \left(0, \sum_{j=1}^{n} (D_j - 9) \right) \right)$$

where D_j represents the duration of surgery j.

Constraints

Anesthesiologist Constraint:

$$\sum_j x_{ij} \leq 1 \quad \forall i$$

Surgery Assignment Constraint:

$$\sum_i x_{ij} = 1 \quad \forall j$$

Room Constraint:

$$\sum_j x_{ij} = 1 \quad \forall i$$

Gap Constraint:

$\text{Gap}_i \geq 15 \quad \forall i$ with surgeries in different rooms

Shift Duration Constraint:

$\text{Duration}_i = \text{Latest\_Surgery}_i - \text{Earliest\_Surgery}_i \quad \forall i$

$\text{Duration}_i \geq 5 \quad \forall i$

$\text{Duration}_i \leq 12 \quad \forall i$

Settings

- Number of anesthesiologists: Unlimited
- Number of operating rooms: 20

4 Source code

The code available in `https://github.com/AvihaiNaa/coding_assignment_gmix.git`.