Diagnosing bacteremia - Statistical analysis plan (SAP)

Michael Kammer

July 31, 2024

1 Structure of the SAP

In this document we outline a Frequentist analysis of the bacteremia task example. For a comprehensive initial data analysis (IDA) we refer to `https://stratosida.github.io/regression-regrets/Bact_intro.html`. The dataset itself is available from `https://zenodo.org/records/7554815`.

2 Statistical focus

Our main research question is: "How well can we diagnose the presence of bacteremia (bacteria in the bloodstream) with a parsimonious set of predictors?".

Our statistical focus areas are representative of a typical predictive model building task:

1. Develop a parsimonious model for the outcome using modern penalized regression.
2. Handle missing data using multiple imputation, and account for that during model development.
3. Internally validate the model using resampling to estimate the out-of-sample error and its variability, also accounting for the multiple imputation.

The key difficulty here is the presence of missing data, which is often ignored or handled in a very simple manner in practice. We aim to adopt a workflow that is sound but pragmatic, i.e. implementable using software packages available for statistical practice. For simplicity we assume the presence of missing data only during the model development stage, but we expect no missing data during application of the model. This is likely not required for Bayesian methods that directly allow to accommodate missing data during application as well. We use multiple imputation for handling missing data as it is very frequently

used in all kinds of predictive tasks, even if it has strong ties to the Bayesian framework. It requires some simplifications (i.e. using Rubin's rules) to adapt to the Frequentist framework. Lastly, combining imputation and validation requires sophisticated resampling methodology in a Frequentist framework, and may be streamlined in a Bayesian setting.

3 Variables

The outcome is "presence of bacteremia" as indicated by the variable "BloodCulture" (yes/no).

As predictors we aim to include as many variables as possible, with two reasons for exceptions to reduce extreme collinearities between predictors. First, we will only use the absolute measurements for white blood cells (MONO, LYM, NEU, EOS, BASO), not the ratios (MONOR, LYMR, NEUR, EOSR, BASOR), and also exclude the total amount (WBC). These variables are closely related as they are different measurements of the same kind of cells, and add up to the total amount. Second, we pragmatically exclude variables with extremely high variance inflation factors:

- we will exclude MCV based on its extremely high correlation with MCH.
- we will exclude HCT based on its extremely high correlation with HGB.

4 Missing data handling

We will use multiple imputation by chained equations (MICE) to impute missing data. The MICE approach is motivated by Bayesian methodology, but nevertheless very popular and used in many prediction tasks. The imputation model for a specific predictor will comprise all other predictors as well as the outcome variable. Note that there are no missing values in the outcome variable itself. As model for imputing data we will employ predictive mean matching. In our setting with a large sample size and at most moderate missingness this method is expected to work well. As it is robust against transformations of the variable to be imputed we will not seek to symmetrize their distributions before imputation. The model for predictive mean matching will comprise linear-additive effects only for simplicity. Given the overall level of missingness which is below 20% for most predictors we aim to create 20 imputed datasets. However, given the high computational demand for the combination with internal validation (see below), we will likely reduce the number of imputations within the validation procedure. This has been shown to have no impact on the precision of performance estimates in Wahl et al. (2016). For imputation models on the full dataset we will check trace plots to evaluate the mixing of the Markov chains of the MICE procedure. We will also do so for a selected number of imputations models for resamples.

5 Model development

To obtain a parsimonious prediction model we will use the adaptive Lasso to obtain a parsimonious prediction model. The adaptive Lasso is a weighted variant of the commonly used Lasso regression model known to improve selection capabilities. In detail, we will fit an adaptive Lasso model in each imputed dataset $\{D_j, j = 1, \ldots, m\}$ obtained from the full dataset D, and derive the final model M by averaging the coefficients from each run as outlined in Musoro et al. (2014). While this approach may result in a less parsimonious model compared to other approaches such as thresholding or stacking, it is very simple to do and reflects the key principles of working with multiple imputation. We expect it to capture essential variables for prediction while shrinking the coefficients of unimportant predictors strongly towards zero, especially if they are only selected in few imputations. The penalization strength will be tuned using 5-fold cross-validation. We define penalization weights for the adaptive Lasso as the reciprocals of the absolute values of the coefficients estimated from an ordinary logistic regression. Fitting this full model will be feasible due to the large sample size and the sufficiently high outcome prevalence - this should also hold for tuning within the resampling procedure below. These weights may also include an exponent γ, which we will set to 0.5 for simplicity. We will ensure that the weights are estimated within the training folds during cross-validation.

We will use all predictors outlined above, with linear functional forms for simplicity. We will symmetrize distributions for all continuous laboratory parameters using pseudolog transformations. These are given by $t(x) = \sinh^{-1}(x/2\sigma)/\log(10)$ in Johnson (1949) and are well defined also for values of 0. To find an "optimal" value for σ we will search a sequence of values defined as $2^x, x \in \{-10, -9, \ldots, 9, 10\}$ and use the value with the highest Pearson correlation to a normal distribution.

6 Model fit and evaluation

As evaluation measures for our research question we will focus on the c-index for discrimination and the calibration intercept and calibration slope for calibration. We will compute the latter two measures using a logistic regression model of observed against predicted outcomes. For the estimation of all measures respecting multiple imputation, including corresponding confidence intervals, we follow the recommendations in Wahl et al. (2016). The overall strategy will be to first draw bootstrap resamples, then use multiple imputation in each resample, and then compute performance measures employing the 0.632+ bootstrap approach outlined in Efron & Tibshirani (1997).

First we fit and tune the model of interest M as outlined above on the multiply imputed full dataset D. Here we use 20 multiple imputations $\{D_j, j = 1, \ldots, 20\}$, regardless of the number of imputations used in the following resampling step. Hence, we obtain apparent, optimistic performance estimates $\hat{p}_M^{(D,D)} = 1/20 \sum_j \hat{p}_M^{(D,D_j)}$ by averaging the performance estimates from each imputation. Here the notation (X, Y) refers to performance estimates obtained

by fitting a model using a dataset X and evaluating it on dataset Y.

Next, we draw a number of bootstrap resamples $\{D_i, i = 1, \ldots, b\}$, each of which contains approximately 63.2% of the original dataset D. In each bootstrap, we use MICE to obtain multiply imputed datasets $\{D_{ij}, j = 1, \ldots, m\}$ (with m possibly lower than 20). Similarly, we obtain multiply imputed versions D_{ij}^c of the remaining 36.8% of "out of bag" samples by applying MICE separately for the samples D_i^c not in the bootstrap sample D_i. In each D_i we fit the prediction model M_i in the same way as the final model M outlined above. These models are then used for predictions in D_i^c to obtain performance estimates $\hat{p}_{M_i}^{(D_i,D_i^c)} = 1/m \sum_j \hat{p}_{M_i}^{(D_i,D_{ij})}$. In the next step we obtain $\hat{p}_M^{BS} = 1/b \sum_i \hat{p}_{M_i}^{(D_i,D_i^c)}$ by averaging the results from all bootstrap resamples.

Now we can obtain the 0.632+ estimate for performance by $\hat{p}_M^{0.632+} = (1 - w)\hat{p}_M^{(D,D)} + w\hat{p}_M^{BS}$ with weights $w = \frac{0.632}{1-0.368R}$. Here R is a relative overfitting rate given by $R = \frac{\hat{p}_M^{BS} - \hat{p}_M^{(D,D)}}{\hat{p}_M^0 - \hat{p}_M^{(D,D)}}$. The performance value $\hat{p}_M^0$ refers to the performance of the model in the absence of an effect. For our evaluation measures this is known from theory as 0.5 for the c-index, and 0 and 1 for the calibration intercept and calibration slope, respectively.

For confidence intervals we will proceed following Jiang et al. (2008), Wahl et al. (2016). First, we compute the apparent performance measures $\hat{p}_{M_i}^{(D_i,D_i)}$ in each bootstrap resample by averaging the performance measures when predicting values from each imputation D_{ij}. Then we subtract the overall apparent performance to define $w_i = \hat{p}_{M_i}^{(D_i,D_i)} - \hat{p}_M^{(D,D)}$. Next we compute the $\alpha/2$ and $1-\alpha/2$ percentiles $\hat{\xi}_{\alpha/2}$ and $\hat{\xi}_{1-\alpha/2}$ of the distribution of $\{w_i, i = 1, \ldots, b\}$. Hence we obtain confidence intervals for $\hat{p}_M^{0.632+}$ as $[\hat{p}_M^{0.632+} - \hat{\xi}_{1-\alpha/2}, \hat{p}_M^{0.632+} + \hat{\xi}_{\alpha/2}]$. We use $\alpha = 0.05$ to obtain 95% confidence intervals. This procedure was shown to have adequate type 1 error and reasonable power, and has the great advantage that it can be incorporated in the resampling procedure directly. Thus it accounts for the multiple imputation and needs minimal additional computational resources.

For this approach to model evaluation we have to choose a number of bootstrap samples b. Following the recommendations in Wahl et al. (2016) we will seek to maximize the number of bootstrap samples, rather than the number of imputations m in each bootstrap. The data presented by the authors suggests that at most 1000 bootstrap resamples should provide performance estimates with very low standard deviations, and that there is little improvement between 500 and 1000 resamples. This is particularly the case with a large sample size (2000 in the paper, which is smaller than the sample size in our case). Hence, we will aim to use 1000 resamples. If the running times become prohibitive (i.e. longer than 24 hours of parallel computation) we will first reduce m to 10 or 5, and if that does not reduce running times only then will we reduce b to 500.

7 Implementation details

All analyses will be performed in the statistical software R. For multiple imputation we will use the package `mice`. To implement predictive mean matching we will use the default values used in the package. For the implementation of the adaptive Lasso we will use the `glmnet` library with mostly default settings. Tuning will be conducted with the function `cv.glmnet()`. Penalty factors will be estimated using `stats::glm`.

8 Additional model descriptives

Besides the estimated performance measures, we will provide a calibration plot for the final model M showing results from each imputed dataset overlayed. We will also provide prediction and calibration stability plots adapted from Riley & Collins (2023). Here we will plot the estimated probability for bacteremia of the final model M (averaged over imputations) for all individuals in D against their estimated probability of the models M_i (averaged over imputations) from each bootstrap resample. We will do a similar plot for calibration by overlaying all calibration lines obtained for M and the models M_i (each obtained by averaging predictions over imputations). Finally, we will provide histograms for the coefficients of all fitted models M_i, highlighting the estimated value for M to assess the stability of the set of selected predictors.

We will further report all issues (e.g. failure to fit penalty weight model, failure to fit adaptive Lasso model, failure to fit imputation model) that occur during fit of the final model and the resampling procedure. We expect only very few issues to happen.

9 Potential problems

- Computational demand: the multiple imputation and resampling may require a lot of computational resources. We may have to reduce the number of subsamples and imputations. In Wahl et al. (2016) the authors even use only a single imputation per subsample.
- Model diagnostics: we do hardly any model checks for the models in the resampling procedure due to the large number of models fitted. We may check a subsample of them.
- Pessimistic evaluation: our model evaluation scheme requires imputation of the validation out-of-bag samples. This is normally not necessary as we assume data during application of the model to be complete, and introduces additional variability into the model evaluation. We expect this to widen the confidence intervals.

10 Version history

1. 27.5.2024: New version based on Georgs and Marianas comments. Refined description.
2. 9.4.2024: Removed gbm model and model choice after discussion to simplify
3. 16.2.2024: Initial version

References

Efron, B. & Tibshirani, R. (1997), 'Improvements on cross-validation: The 632+ bootstrap method', *Journal of the American Statistical Association* **92**(438), 548–560. doi: 10.1080/01621459.1997.10474007.
URL: *https://doi.org/10.1080/01621459.1997.10474007*

Jiang, B., Zhang, X. & Cai, T. (2008), 'Estimating the confidence interval for prediction errors of support vector machine classifiers', *Journal of Machine Learning Research* **9**(17), 521–540.
URL: *http://jmlr.org/papers/v9/jiang08a.html*

Johnson, N. L. (1949), 'Systems of frequency curves generated by methods of translation', *Biometrika* **36**(1/2), 149–176.
URL: *http://www.jstor.org/stable/2332539*

Musoro, J., Zwinderman, A., Puhan, M., ter Riet, G. & Geskus, R. (2014), 'Validation of prediction models based on lasso regression with multiply imputed data', *BMC Medical Research Methodology* **14**(1), 116.

Riley, R. D. & Collins, G. S. (2023), 'Stability of clinical prediction models developed using statistical or machine learning methods', *Biom J* **65**(8), e2200302.

Wahl, S., Boulesteix, A. L., Zierer, A., Thorand, B. & van de Wiel, M. A. (2016), 'Assessment of predictive performance in incomplete data by combining internal validation and multiple imputation', *BMC Med Res Methodol* **16**(1), 144.
URL: *https://www.ncbi.nlm.nih.gov/pubmed/27782817*