Causal-Based Feature Selection with Kernel-Based Conditional Independence Tests - Applied in Biostatistics

Ioannis Maris
University of Crete, Heraklion
`math5666@math.uoc.gr`

Contents

Introduction 2

1 Improving the Use of `P-Values` 2

1.1 Converting `p-values` into Bayes Factors . . . 2

1.2 Use 0.005 Instead of 0.05 as a Threshold . . . 2

2 Feature Selection 3

2.1 AIC, AICc, BIC Information Criteria . . . 3

2.2 Constraint-Based vs Score-Based vs Hybrid Algorithms . . . 3

2.3 Forward Stepwise Ridge Logistic Selection . . . 3

2.4 Forward-Backward Selection with Kernel-Based Conditional Independence Test . . 4

2.5 Max-Min Parent & Children (MMPC): Conditioning on Subsets . . . 5

2.6 Graphical Lasso with a Kernel-Based Conditional Independence Test . . . 5

3 Model Selection 6

3.1 Pipelines & Grid-Search Cross-Validation . . . 7

3.2 Why Logistic Regression? . . . 7

3.3 Ridge-Lasso-ElasticNet Logistic Regression Classifier . . . 7

4 ROC AUC Performance & Data Visualization 7

4.1 Shapley Additive Explanations Plots . . . 7

4.2 Comparing with JADBio . . . 9

5 Conclusions 9

Abstract

This project presents an approach for binary classification using both causal-based & simple stepwise model building based approaches. The main goal of the study is to build an efficient model that contains the "important features" of the dataset by implementing three different causal-based feature selection algorithms and compare them with a simple Ridge Regression. Furthermore, we are interested in estimating the probabilities, to do that, logistic regression is the best way to go.
Keywords: Logistic Regression, Causal-Based Feature Selection, Markov Blanket Discovery

Introduction

The Breast Cancer Wisconsin (Diagnostic) dataset on Kaggle[1] is a collection of medical data that pertains to breast cancer diagnosis. The dataset contains a total of 569 observations, each representing a patient with a biopsied breast cell. Each observation includes various features such as the size of the cell nucleus, texture, smoothness, perimeter, area and more, as well as a diagnosis of whether the cell is benign (not cancerous or negative class) or malignant (cancerous or positive class). This dataset is a great resource for researchers, students and data scientists who are interested in developing machine learning models for early detection of breast cancer, feature engineering and classification tasks using the size of the cell nucleus as a feature.

1. Improving the Use of P-Values

P-values are commonly used by researchers to assess the strength of evidence against a null hypothesis, but they can be misinterpreted. This recommendation can help guard against harmful p-value misinterpretations and move away from reliance on 'p < 0.05'.

1.1 Converting p-values into Bayes Factors

The p-value quantifies the discrepancy between the data and a null hypothesis of interest, usually the assumption of no difference or no effect. A Bayesian approach allows the calibration of p-values by transforming them to direct measures of the evidence against the null hypothesis, so-called Bayes factors.

$$\mathbf{BF} = \frac{\text{average likelihood of the observed data under the alternative hypothesis}}{\text{likelihood of the observed data under the null hypothesis}}$$

Unfortunately, there is no unique mapping between p-values and Bayes factors because, unlike calculating the p-value, calculating the Bayes factor requires specifying an alternative hypothesis (more specifically, a prior distribution for the parameter values under the alternative hypothesis). What can we do? Several methods have been developed for using the p-value to calculate an upper bound on BF, called the Bayes factor bound (BFB), namely:

$$\mathbf{BF} \leq \mathbf{BFB} = \frac{1}{-e\, p \log p}$$

1.2 Use 0.005 Instead of 0.05 as a Threshold

"Use 0.005 Instead of 0.05 as a Threshold". The following table shows the value of BFB for a wide range of p-values. The table also gives the corresponding upper bound on the posterior probability of H_1. When the prior probabilities of H_0 and H_1 are equal, this upper bound on the posterior probability of H_1 is given by $P^U(H_1 \mid p) = BFB/(1 + BFB)$.

For example, a p-value of 0.05 corresponds to a Bayes factor of at most 2.46:1. That is, the data imply odds in favor of the alternative hypothesis relative to the null hypothesis of at most 2.46 to 1. So if the null and alternative hypotheses were originally equally likely, there remains, at least, a 29% chance that the null hypothesis is true. In the other hand, a p-value of **0.005** indicates the alternative hypothesis is at most ≈**14** times as likely as the null hypothesis.

[1] https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data

Table 1: Evidence against the null hypothesis

p-value	0.1	0.05	0.01	0.005	0.001	0.0001	1e-05
BFB	1.6	2.46	7.99	13.89	53.26	399.42	3195.36
$\mathbf{P}^U(\mathrm{H}_1\|\mathbf{p})$	0.615	0.7107	0.8887	0.9328	0.9816	0.9975	0.9997

2. Feature Selection

In this paper, we are going to use four different feature selection algorithms using both `p-values` and other model selection information criteria such as: Corrected Akaike and Bayesian Information Criterion.

2.1 AIC, AICc, BIC Information Criteria

AIC and BIC are both MLE driven and penalize free parameters in an effort to combat overfitting, they do so in ways that result in significantly different behavior.

$$\textbf{AIC} = -2\log(\mathcal{L}) + 2p, \ \textbf{BIC} = -p\log(\mathcal{L}) + 2p,$$

$$\mathrm{AIC_c} = \mathrm{AIC} + \frac{2p(p+1)}{n-p-1} = -2\log(\mathcal{L}) + 2p + \frac{2p(p+1)}{n-p-1},$$

where $\mathcal{L}$ is the maximized value of the likelihood function, p is the number of parameters estimated by the model and n the number of observations.
The best model in the group compared is the one that minimizes these scores. Clearly, AIC does not depend directly on sample size. Moreover, generally speaking, AIC presents the danger that it might overfit, whereas BIC presents the danger that it might underfit. Furthermore, BIC is more tolerant of free parameters than AIC, but less tolerant at higher n (as the natural log of n overcomes 2). In general, AIC is best for prediction as it is asymptotically equivalent to cross-validation and BIC is best for explanation as it is allows consistent estimation of the underlying data generating process.
When the sample size is small, there is a substantial probability that AIC will select models that have too many parameters, i.e. that AIC will overfit. To address such potential overfitting, AICc was developed, AICc is essentially AIC with an extra penalty term. Note that as $n \to \infty$, the extra penalty term converges to 0, and thus AICc converges to AIC.

2.2 Constraint-Based vs Score-Based vs Hybrid Algorithms

Bayesian Network Structure Learning is defined by the combination of a statistical criterion and an algorithm that determines how the criterion is applied to the data.
Constraint-based algorithms identify conditional independence constraints with statistical tests, and link nodes that are not found to be independent. In the other hand, **Score-based** algorithms are applications of general optimisation techniques, each candidate DAG[2] is assigned a network score maximise as the objective function. Constraint-based algorithms produce BNs with the highest log-likelihood, hybrid have the worst log-likelihood values and includes only a few teleconnections. Although score-based algorithms are faster than both hybrid and constraint-based, constraint-based algorithms (i.e. PC, Inter-IAMB), are more accurate than score-based for small sample sizes.

2.3 Forward Stepwise Ridge Logistic Selection

Forward stepwise selection is a computationally efficient alternative to best subset selection. While the best subset selection procedure considers all 2^p possible models containing subsets of the p predictors, forward stepwise considers a much smaller set of models. More formally, the FSS procedure is given in Algorithm 1

[2] (DAG) is a directed graph with no directed cycles.

Algorithm 1 Forward Stepwise Selection

Input: Dataset **D**, predictors **p**
Output: Selected Variables

```
Let $\mathbf{M}_0$ denote the null model, which contains no predictors.
for $k = 0, ..., p{-}1$ do
    Consider all $\mathrm{p{-}k}$ models that augment the predictors in $\mathrm{M_k}$ with one additional predictor.
    Choose the best among these $\mathrm{p{-}k}$ models, and call it $\mathrm{M_{k+1}}$ //best is defined as having highest AUC
end for
Select a single best model from among $\mathbf{M}_0, \ldots, \mathbf{M}_p$ using AICc/BIC
```

This amounts to a total of $1 + \sum_{k=0}^{p-1}(p - k) = 1 + p(p + 1)/2$ models. This is a substantial difference: i.e. if $p = 20$, best subset selection requires fitting $1.048.576$ models, whereas forward stepwise requires fitting only 211 models.

2.4 Forward-Backward Selection with Kernel-Based Conditional Independence Test

Algorithm 2 Forward Selection

Input: Dataset **D**, Target Variable **T**, Variables **V**, Significance Threshold $a \rightarrow 0.005$
Output: Selected Variables **S**

```
$R \leftarrow V \setminus S$    //Remaining Variables
while $S$ changes do
    //Identify $V^*$ with min p-value given $S$
    $V^* \leftarrow \underset{V \in R}{\operatorname{argmin}} \;\; \texttt{pvalue}(T; V|S)$
        $R \leftarrow R \setminus V^*$
        if pvalue$(T; V^*|S) \leq a$ then:
            $S \leftarrow S \cup V^*$
        end if
end while
return S
```

Algorithm 3 Backward Selection

Input: FS Selected Variables **S**

```
while $S$ changes do
    //Identify $V^*$ with max p-value given $S \setminus V$
    $V^* \leftarrow \underset{V \in S}{\operatorname{argmax}} \;\; \texttt{pvalue}(T; V|S \setminus V)$
        if pvalue$(T; V^*|S \setminus V) > a$ then:
            $S \leftarrow S \setminus V^*$
        end if
end while
return S
```

The Forward-Backward selection algorithm is a straightforward and efficient method that uses a greedy approach (hill-climbing) to select features. It is suitable when a sufficient sample size is available to condition on all selected features. The algorithm is based on the theorem that it returns the Markov Blanket[3] of T in distributions that conform to a Bayesian Network with latent variables, assuming perfect Conditional Independence (CI) tests. Its complexity, measured by the number of CI tests, is $O(n \cdot s)$ where n is the total number of features and s is the number of selected features. When enough sample size is available, Backward-Search algorithm can be used instead. Also, using `p-values` unifies algorithms that appear different depending on the outcome type.

[3] A Markov Blanket in a graph is a set of nodes that renders all other nodes conditionally independent of a target given its direct parents, children, and children's parents.

2.5 Max-Min Parent & Children (MMPC): Conditioning on Subsets

Algorithm 4 Max-Min Parents and Childern (MMPC)

Input: Target **T**, Variable **V**, Max Conditioning Size k
Output: Selected Variables **S**

```
S ← $\emptyset$ // Selected variables
R ← V // Remaining variables
while $S$ changes do
    //Maximum p-value over subsets Z of S with size $\leq k$
    for V∈R do
        $\texttt{MAXPVALUE}(T;V) \leftarrow \max_{Z} \texttt{PVALUE}(T;V \mid Z)$
        s.t. Z∈S∧|Z|≤ $k$
    end for
    //Identify $V^*$ with minimum MAXPVALUE
    $V^* \leftarrow \operatorname*{argmin}_{V \in R} \texttt{MAXPVALUE}(T;V)$
    //Remove from R
    R←R$\backslash V^*$
    R←R$\backslash \{V : V \in \mathbf{R} \wedge \texttt{MAXPVALUE}(T;V|S) > a\}$
    //Select $V^*$ if dependent
    if $\texttt{MAXPVALUE}(T;V^*) \leq a$ then
        S←S$\cup V^*$
    end if
end while
return S
```

When one has limited statistical power to consider only a certain number of features, the MMPC (with symmetry correction and a large enough value of k) is a suitable method. It will return the parent and child variables of a target variable T, in a way that is consistent with a Bayesian Network that includes latent variables. Without the symmetry correction, it will return a slightly larger set of variables. By using simple extensions, such as MMMB, the full Markov Blanket can be obtained. The computational complexity of this method is $O(n \cdot s^k)$, and typically using a value of $k = 3$ or $k = 4$ yields good results. The Max-Min Heuristic can be used to select the most important variable by choosing the one with the strongest minimum conditional association or the smallest maximum `p-value`, when considering all subsets of the selected features. MMPC works better with small sample size ($< 10^3$) and guarantees $\subseteq$ **MB**.

2.6 Graphical Lasso with a Kernel-Based Conditional Independence Test

The graphical lasso[4] is a method for estimating the inverse covariance matrix of a multivariate Gaussian distribution when some of the variables are conditionally dependent. It is a type of regularized inverse covariance estimation, meaning that it tries to find a sparse solution[5] by adding a penalty term to the likelihood function. The graphical lasso algorithm can be used for variable selection in high-dimensional problems, and it has been applied in a wide range of fields, including genetics, neuroscience, and finance. In statistics, the graphical lasso is a sparse penalized maximum likelihood estimator for the concentration or precision matrix[6] (inverse of covariance matrix) of a multivariate elliptical distribution.

Consider observations $X_1, \ldots, X_n$ from multivariate Gaussian distribution $X \sim \mathcal{N}(0, \Sigma)$. We are interested in estimating the precision matrix $\Theta = \Sigma^{-1}$. A systematic approach by Friedman, Hastie

[4] R package: glasso, python: cdt.independence.graph.HSICLasso

[5] i.e. a solution where many of the entries in the inverse covariance matrix are exactly zero

[6] The precision matrix or concentration matrix is the matrix inverse of the covariance matrix or dispersion matrix, $P = \Sigma^{-1}$.

and Tibshirani :

$$\hat{\Theta} = \underset{\Theta \geq 0}{\text{argmin}} \quad \left(\underbrace{\text{tr}(S\Theta)}_{\langle S,\Theta \rangle} - \log\det(\Theta) + \lambda \underbrace{\sum_{j \neq j} |\Theta_{jk}|}_{\|\Theta\|_1} \right)$$

(Convex optimization) where $S = \dfrac{1}{n}\sum_{i=1}^{n} X_i X_i^T$ (sample covariance matrix). The gradient equation:

$$\Theta^{-1} - S - \lambda \cdot \text{Sign}(\Theta) = 0$$

Let $W = \Theta^{-1}$ and

$$\begin{bmatrix} W_{11} & w_{12} \\ w_{21}^T & w_{22} \end{bmatrix} \begin{bmatrix} \Theta_{11} & \theta_{12} \\ \theta_{21}^T & \theta_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & 0 \\ 0^T & 1 \end{bmatrix} \Rightarrow w_{12} = -W_{11}\theta_{12}/\theta_{22} = W_{11}\beta,$$

where $\beta = -\theta_{12}/\theta_{22}$. The upper right block of the gradient equation:

$$W_{11}\beta - s_{12} + \lambda \cdot \text{Sign}(\beta) = 0$$

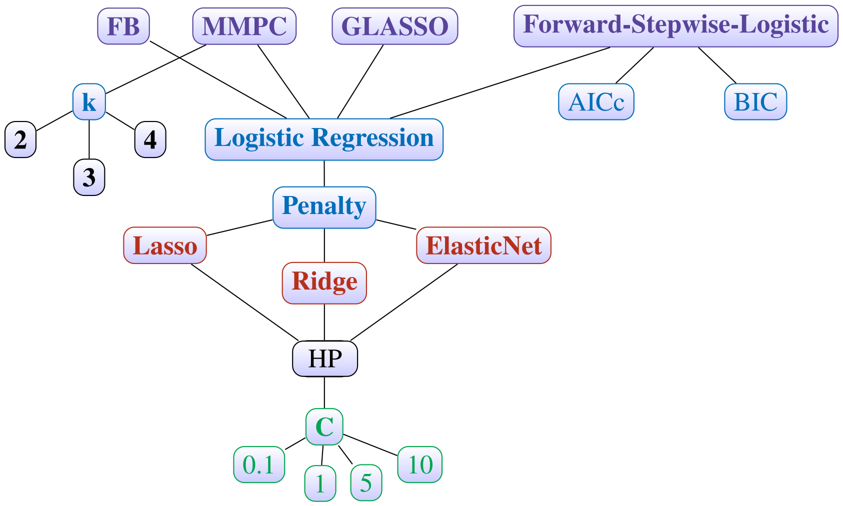
Which is recognized as the estimation equation for the Lasso regression.

Algorithm 5 Graphical Lasso

```
Initialize $\mathbf{W} = \mathbf{S} + \lambda\mathbf{I}$. The diagonal of $\mathbf{W}$ remain unchaged in what follows.
Repeat for $j = 1, 2, \ldots, p, 1, 2, \ldots, p, \ldots$ until convergence:
    Partition the matrix $W$ into part 1: all but the jth row and column, and part 2: the jth row and column.
    Solve the estimating equations $W_{11}\beta - s_{12} + \lambda \cdot \text{Sign}(\beta) = 0$ using the cyclical coordinate-descent algorithm for modified lasso.
    Update $w_{12} = W_{11}\hat{\beta}$.
In the final cycle (for each j) solve for $\hat{\theta}_{12} = -\hat{\beta} \cdot \hat{\theta}_{22}$, with $1/\hat{\theta}_{22} = w_{22} - w_{12}^T\hat{\beta}$. =0
```

It is important to note that the output of the graphical lasso algorithm is an estimate of the inverse covariance matrix, which can be used to identify the conditional dependencies between variables. But, it is not guaranteed that the output will be the exact Markov blanket or a subset of it.

3. Model Selection



Tree 1: Configuration structure

3.1 Pipelines & Grid-Search Cross-Validation

For each feature selection algorithm, we construct a pipeline and employ a 5-Fold stratified cross-validation method with a scoring metric of ROC AUC. To optimize the process, we utilize the ‘saga‘ optimizer[7], which requires that the features have a similar scale for guaranteed fast convergence. Therefore, scaling is necessary. Ultimately, we place each pipeline into a grid, and fit it to the training data[8]. After the training is complete, we refit the best model selected using all the (training) data.

3.2 Why Logistic Regression?

Logistic regression is a widely used method for binary classification in the field of bio-statistics, especially when the estimation of probability is a crucial goal. This method provides several advantages over other algorithms, such as a clear probabilistic interpretation of the results, a linear relationship between the dependent and independent variables, the option to incorporate regularization techniques, efficient computation, and the capability to incorporate multiple independent variables in the analysis. These features make logistic regression a compelling choice for many binary classification problems in bio-statistics

3.3 Ridge-Lasso-ElasticNet Logistic Regression Classifier

Lasso will eliminate many features, and reduce overfitting in our linear model. Ridge will reduce the impact of features that are not important in predicting your y values. Elastic Net combines feature elimination from Lasso and feature coefficient reduction from the Ridge model to improve our model’s predictions. Cost functions:

$$J_{\textbf{Ridge}}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(y-\hat{y})^2 + \lambda\sum_{j=1}^{n} w_j^2. \quad J_{\textbf{Lasso}}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(y-\hat{y})^2 + \lambda\sum_{j=1}^{n}|w_j|$$

$$J_{\textbf{ElasticNet}}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(y-\hat{y})^2 + r\lambda\sum_{j=1}^{n}|w_j| + \frac{1-r}{2}\lambda\sum_{j=1}^{n} w_j^2$$

4. ROC AUC Performance & Data Visualization

Table 2: Performance, Hyperparameter tuning, Features selected out of 30

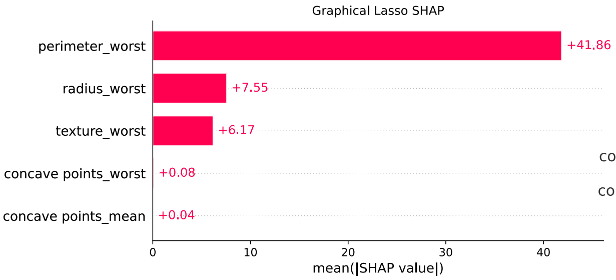
	AUC Performance			
	Mean CV	**Hold-Out**	**Best HP**	**# Features Selected**
Graph LASSO	0.989	0.9994	C: 1, penalty: ℓ^2	5
MMPC	0.983	0.9935	k: 2, C: 10, penalty: ℓ^1	3
FB	0.984	0.9975	C: 1, penalty: ℓ^1	3
FS	0.995	0.9975	metric: AICc, C: 10, penalty: ℓ^1	9

4.1 Shapley Additive Explanations Plots

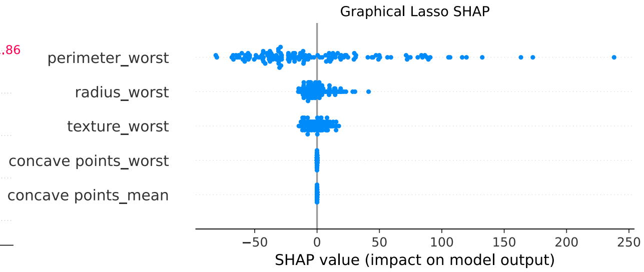
SHAP plots are important for the prediction of cancer because they provide insight into the factors driving a model’s prediction, which is crucial in medical applications. With SHAP plots, the importance of each feature in the prediction can be understood, helping in identifying important features for diagnosis and treatment. SHAP plots offer numerous benefits, including improved interpretability of the model’s decision-making process, evaluation of prediction fairness, and enhanced feature selection. Let’s now take a look at them :

[7] https://www.di.ens.fr/∼fbach/Defazio_NIPS2014.pdf

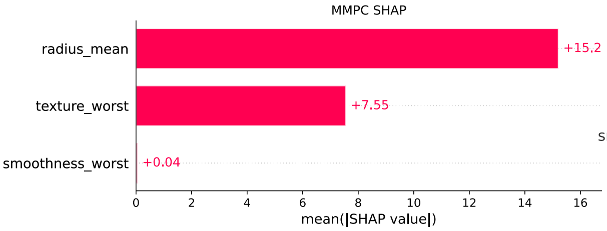
[8] We split the dataset into a training set (75%) - hold-out-set (25%)



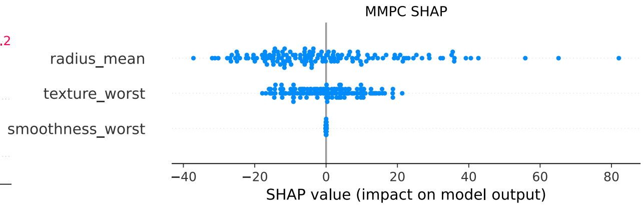
(a) Plot 1: Graph Lasso bar SHAP



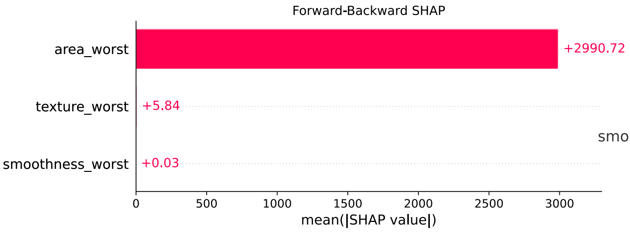
(b) Plot 2: Graph Lasso dot SHAP



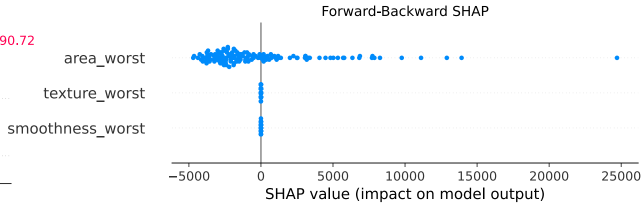
(a) Plot 3: MMPC bar SHAP



(b) Plot 4: MMPC dot SHAP



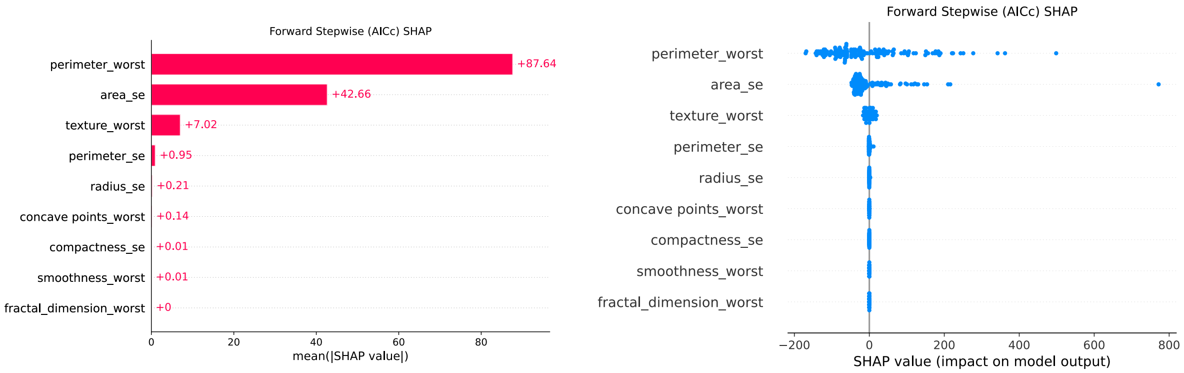
(a) Plot 5: Graph Lasso bar SHAP



(b) Plot 6: FB dot SHAP

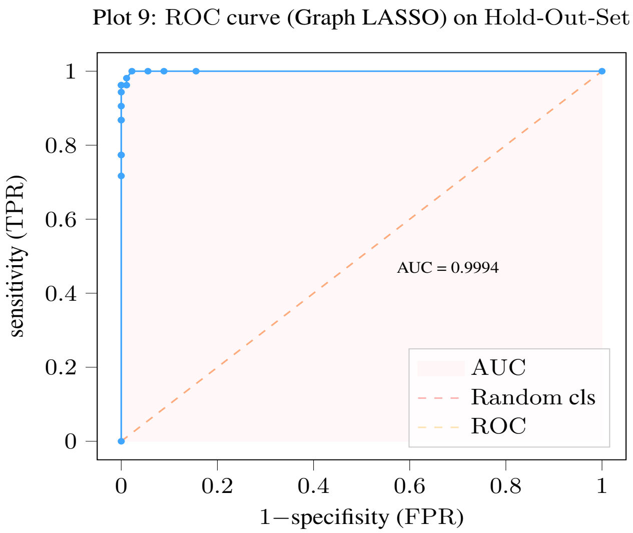
Based on the best out-of-sample performance, Graphical Lasso returns the highest AUC score. Let's examine the features selected by each algorithm by looking the SHAP[9] plots: We can compare the distribution of the SHAP values by calculating the mean absolute value (MAV) and choose the algorithm with the smallest MAV. In our case, the algorithm with the smallest MAV value is the MMPC ($\texttt{MMPC} < \texttt{GLASSO} < \texttt{Forward-Stepwise} << \texttt{Forward-Backward}$) Hence, the final model based on out-of-sample AUC score is the Graphical Lasso, however, based on MAV, the MMPC seems the best choice. We can compare the distribution of the SHAP values by calculating the median absolute deviation (MAD), the smaller, the better.

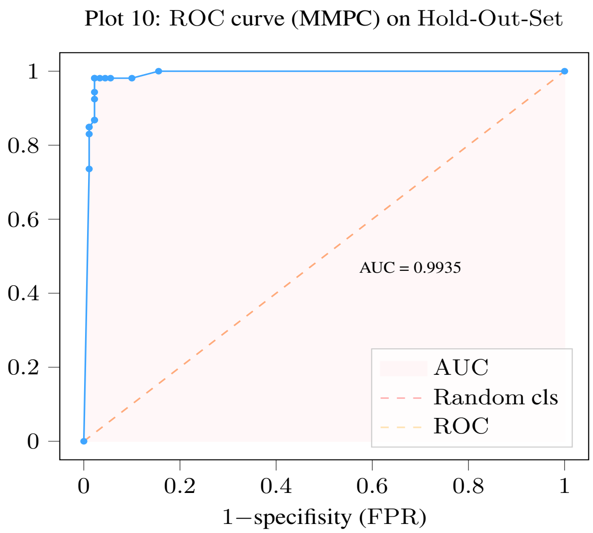
[9] SHAP (SHapley Additive exPlanations) plots display feature importance in ML models by visualizing the contribution of each feature towards a prediction for a specific instance. It uses Shapley values from game theory.



(a) Plot 7: Forward-Stepwise bar SHAP

(b) Plot 8: Forward-Stepwise dot SHAP





We have calculated the confidence interval of the best model (Graph Lasso) by bootstrapping the hold-out set. **95%** Confidence Interval → **[0.99731, 1]**. After the bootstrapping, we shall retrain the model using all the data.

4.2 Comparing with JADBio

JADBio selected 5 out of 30 features (just like Graphical Lasso), namely: {`perimeter worst`, `smoothness worst`, `texture worst`, `concavity mean`, `symmetry worst`}. JADBio reports an AUC performance of 0.989 and a mean average precision[10] of 0.99 .

5. Conclusions

In conclusion, this paper compares causal-based and simpler linear models for feature selection in biostatistics, highlighting the importance of considering feature importance in biology and medicine, as it plays a crucial role in accurately predicting and diagnosing diseases. The results show that all models had similar AUC performances, but the selected features differed depending on the algorithm used. Notably, "texture worst" was consistently selected by all algorithms. The forward-backward selection algorithm had 2 out of 3 common features with the MMPC algorithm, namely "texture worst" and "smoothness worst." The forward-stepwise algorithm selected 9 features, but only a few of them were deemed important. It is emphasized that incorporating prior knowledge of the physical meaning of the features through interaction terms can lead to a more efficient models.

[10] Precision is defined as TP/(TP + FP) and Recall as TP/(TP+FN)

References

[1] Robert Tibshirani, Trevor Hastie, Gareth James, Daniela Witten. An Introduction to Statistical Learning: with Applications in R (Springer Texts in Statistics) 1st ed. 2013, Corr. 7th printing 2017 Edition

[2] Peter D. Hoff. A First Course in Bayesian Statistical Methods

[3] Ioannis Tsamardinos, Giorgos Borboudakis, Gnosis Data Analysis, Forward-Backward Selection with Early Dropping

[4] Trevor Hastie, Robert Tibshirani, Jerome Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics) 2nd Edition

[5] Kenneth P. Burnham, David R. Anderson. Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach 2nd Edition, Kindle Edition

[6] Daniel J. Benjamin James O. Berger. The American Statistician. Three Recommendations for Improving the Use of P-Values

[7] Kun Zhang, Jonas Peters Dominik Janzing Bernhard Schölkopf. Max Planck Institute for Intelligent Systems Spemannstr. 38, 72076 Tübingen Germany. Kernel-based Conditional Independence Test and Application in Causal Discovery

[8] Marco Scutari, Catharina Elisabeth, Graafland Jose , Manuel Gutierrez. Who Learns Better Bayesian Network Structures Constraint-Based, Score-based or Hybrid Algorithms? Department of Statistics University of Oxford, UK

[9] Rahul Mazumder, Trevor Hastie, Massachusetts Institute of Technology Cambridge, and Department of Statistics Stanford University. Departments of Statistics and Health Research and Policy Stanford University. The graphical lasso: New insights and alternatives