P8160 - Breast Cancer Diagnosis

Hongjie Liu, Xicheng Xie, Jiajun Tao, Shaohan Chen, Yujia Li

3/31/2023

Contents

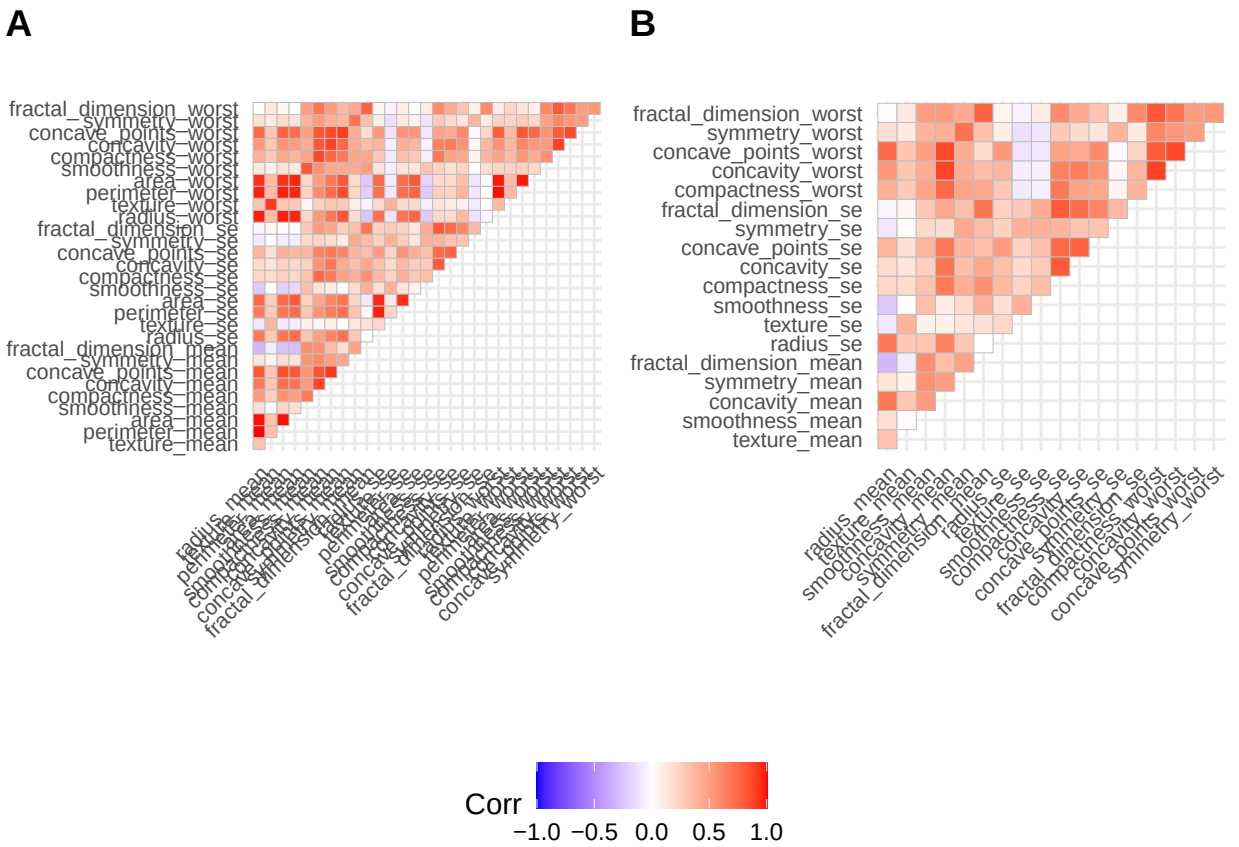
1. Objectives

Mammography is recognized as the most effective screening method for early breast cancer detection, but its accuracy remains limited. And as the number of variables that help predict breast cancer increases, doctors are forced to rely more on their subjective experiences to make decisions. The use of computer models can detect abnormalities in mammograms to aid radiologists in breast cancer diagnosis (Freer et al. 2001). The purpose of this study was to predict breast cancer benign/malignant status by quantitative modeling based on logistic regression, which may help radiologists manage the large amount of available information, make effective decisions to detect breast cancers and reduce unnecessary biopsy.

2. Background

The data given has 569 observations. The column 'Diagnosis' which identifies if the image is coming from cancer tissue or benign cases (M=malignant, B = benign) would be used as outcome for modeling. We denote malignant as 1 and benign cases as 0 for prediction. The other 30 columns correspond to mean, standard deviation and the largest values (points on the tails) of the distributions of the following 10 features computed for the cell nuclei:

- radius: mean of distances from center to points on the perimeter
- texture: standard deviation of gray-scale values
- perimeter: mean size of the core tumor
- area: mean area of the core tumor
- smoothness: local variation in radius lengths
- compactness: perimeter^2/area - 1.0
- concavity: severity of concave portions of the contour
- concave points: number of concave portions of the contour
- symmetry: symmetry of the tumor
- fractal dimension: "coastline approximation" - 1



As exploring the breast cancer data, there are many predictors are highly correlated to each other as shown in the Figure A below, such as 'area_mean', 'perimeter_worst', 'radius_mean' and so on, that could cause unstable parameter estimation as well as perplexing the interpretation of logistic model. While we could eliminate some correlated features for modeling as shown in the Figure B, regularized logistic regression also help to tackle with it, which would be further explore in the following parts.

2. Methods

2.1. Logistic Model

Logistic model measure the probability of an event taking place by having the log-odds for the event be a linear combination of one or more independent variables, and commonly used in classifying binary response variables.

Hereby, the variable "Diagnosis" is a binary response variable indicating if the image is coming from cancer tissue or benign cases (M = malignant, B = benign). In the following logistic regression model, the "Diagnosis" variable will be coded as 1 for malignant cases and 0 for benign cases.

Given n i.i.d. observations with p predictors, we consider a logistic regression model

$$P(Y_i = 1 \mid \mathbf{x}_i) = \frac{e^{\mathbf{x}_i^\top \beta}}{1 + e^{\mathbf{x}_i^\top \beta}}, \ i = 1, \dots, n \tag{1}$$

where $\beta = (\beta_0, \beta_1, \dots, \beta_p)^\top \in \mathbb{R}^{p+1}$ is the parameter vector, $\mathbf{x}_i = (1, X_{i1}, \dots, X_{ip})^\top$ is the vector of predictors in the i-th observation, and $Y_i \in \{0, 1\}$ is the binary response in the i-th observation. Let $\mathbf{y} = (Y_1, Y_2, \dots, Y_n)^\top$ denote the response vector, $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)^\top \in \mathbb{R}^{n \times (p+1)}$ denote the design matrix. The observed

likelihood of $\{(Y_1, \mathbf{x}_1), (Y_2, \mathbf{x}_2) \dots, (Y_n, \mathbf{x}_n)\}$ is

$$L(\beta; \mathbf{y}, \mathbf{X}) = \prod_{i=1}^{n} \left[\left(\frac{e^{\mathbf{x}_i^\top \beta}}{1 + e^{\mathbf{x}_i^\top \beta}} \right)^{Y_i} \left(\frac{1}{1 + e^{\mathbf{x}_i^\top \beta}} \right)^{1 - Y_i} \right].$$

Maximizing the likelihood is equivalent to maximizing the log-likelihood function:

$$f(\beta; \mathbf{y}, \mathbf{X}) = \sum_{i=1}^{n} \left[Y_i \mathbf{x}_i^\top \beta - \log\left(1 + e^{\mathbf{x}_i^\top \beta}\right) \right]. \tag{2}$$

The estimates of model parameters are

$$\hat{\beta} = \arg\max_{\beta} \ f(\beta; \mathbf{y}, \mathbf{X}),$$

and the optimization problem is

$$\max_{\beta} \ f(\beta; \mathbf{y}, \mathbf{X}). \tag{3}$$

Denote $p_i = P(Y_i = 1 \mid \mathbf{x}_i)$ as given in (1) and $\mathbf{p} = (p_1, p_2, \dots, p_n)^\top$. The gradient of f is

$$\begin{aligned} \nabla f(\beta; \mathbf{y}, \mathbf{X}) &= \mathbf{X}^\top (\mathbf{y} - \mathbf{p}) \\ &= \sum_{i=1}^{n} (Y_i - p_i) \mathbf{x}_i \\ &= \begin{pmatrix} \sum_{i=1}^{n} (Y_i - p_i) \\ \sum_{i=1}^{n} (Y_i - p_i) X_{i1} \\ \vdots \\ \sum_{i=1}^{n} (Y_i - p_i) X_{ip} \end{pmatrix}. \end{aligned}$$

Denote $w_i = p_i(1 - p_i) \in (0, 1)$ and $\mathbf{W} = \mathrm{diag}(w_1, \dots, w_n)$. The Hessian matrix of f is given by

$$\begin{aligned} \nabla^2 f(\beta; \mathbf{y}, \mathbf{X}) &= -\mathbf{X}^\top \mathbf{W} \mathbf{X} \\ &= -\sum_{i=1}^{n} w_i \mathbf{x}_i \mathbf{x}_i^\top \\ &= -\begin{pmatrix} \sum_{i=1}^{n} w_i & \sum_{i=1}^{n} w_i X_{i1} & \cdots & \sum_{i=1}^{n} w_i X_{i1} \\ \sum_{i=1}^{n} w_i X_{i1} & \sum_{i=1}^{n} w_i X_{i1}^2 & \cdots & \sum_{i=1}^{n} w_i X_{i1} X_{ip} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} w_i X_{ip} & \sum_{i=1}^{n} w_i X_{in} X_{i1} & \cdots & \sum_{i=1}^{n} w_i X_{ip}^2 \end{pmatrix}. \end{aligned}$$

Next, we show that the Hessian matrix $\nabla^2 f(\beta; \mathbf{y}, \mathbf{X})$ is a negative-definite matrix if $\mathbf{X}$ has full rank.

Proof. For any $(p+1)$-dimensional nonzero vector α, given that $\mathbf{X}$ has full rank, $\mathbf{X}\alpha$ is also a nonzero vector. Since $\mathbf{W}$ is positive-definite, we have

$$\begin{aligned} \alpha^\top \nabla^2 f(\beta; \mathbf{y}, \mathbf{X}) \alpha &= \alpha^\top (-\mathbf{X}^\top \mathbf{W} \mathbf{X}) \alpha \\ &= -(\mathbf{X}\alpha)^\top \mathbf{W} (\mathbf{X}\alpha) \\ &< 0. \end{aligned}$$

Thus, $\nabla^2 f(\beta; \mathbf{y}, \mathbf{X})$ is negative-definite. □

Hence, the optimization problem (3) is a well-defined problem.

2.2. Newton-Raphson Algorithm

While derivative of the likelihood function with respect to the parameters is nonlinear and difficult to solve analytically for maximum likelihood, it requires Newton-Raphson iterations.The target function f given in task 1:

$$f(\beta; \mathbf{y}, \mathbf{X}) = \sum_{i=1}^{n} \left[Y_i \mathbf{x}_i^\top \beta - \log\left(1 + e^{\mathbf{x}_i^\top \beta}\right) \right]. \tag{4}$$

We develop a modified Newton-Raphson algorithm including a step-halving step. *(we probably don't need to ensure that the direction of the step is an ascent direction, since in this example Hessian is always negative-definite. but Hessian could be computationally singular when the starting points are bad)*

Algorithm 1 Newton-Raphson algorithm including a step-halving step

Require: $f(\beta)$ - target function as given in (5); β_0 - starting value
Ensure: $\widehat{\beta}$ such that $\widehat{\beta} \approx \arg\max_\beta \ f(\beta)$
  $i \leftarrow 0$, where i is the current number of iterations
  $f(\beta_{-1}) \leftarrow -\infty$
  while convergence criterion is not met **do**
    $i \leftarrow i + 1$
    $\mathbf{d}_i \leftarrow -[\nabla^2 f(\beta_{i-1})]^{-1} \nabla f(\beta_{i-1})$, where $\mathbf{d}_i$ is the direction in the i-th iteration
    $\lambda_i \leftarrow 1$, where λ_i is the multiplier in the i-th iteration
    $\beta_i \leftarrow \beta_{i-1} + \lambda_i \mathbf{d}_i$
    while $f(\beta_i) \leq f(\beta_{i-1})$ **do**
      $\lambda_i \leftarrow \lambda_i / 2$
      $\beta_i \leftarrow \beta_{i-1} + \lambda_i \mathbf{d}_i$
    end while
  end while
  $\widehat{\beta} \leftarrow \beta_i$

2.3. Logistic-LASSO Model

Regularization is the common approach for variable selection, in which LASSO is to add L-1 penalty to the objective function. In the context of logistic regression, we turn to maximize the penalized loglikelihood as following:

- Log-likelihood f of logistic regression:

$$f(\beta; \mathbf{y}, \mathbf{X}) = \sum_{i=1}^{n} \left[Y_i \mathbf{x}_i^\top \beta - \log\left(1 + e^{\mathbf{x}_i^\top \beta}\right) \right]. \tag{5}$$

- LASSO estimates the logistic model parameters β by optimizing a penalized loss function:

$$\min_\beta \ -\frac{1}{n} f(\beta) + \lambda \sum_{k=1}^{p} |\beta_k|. \tag{6}$$

where $\lambda \geq 0$ is the tuning parameter. Note that the intercept is not penalized and all predictors are standardized.

Then we could develop a path-wise coordinate descent algorithm to the penalized weighted-least-squares-problem with a sequence of nested loops: **OUTER LOOP** In the outer loop, we compute the solutions of

the optimization problem (6) for a decreasing sequence of values for λ: $\{\lambda_1, \dots, \lambda_m\}$, starting at the smallest value $\lambda_1 = \lambda_{max}$ for which the estimates of all coefficients $\hat{\beta}_j = 0,\ j = 1, 2, \dots, p$, which is

$$\lambda_{max} = \frac{1}{n} \max_j \left| \langle \mathbf{x}_{\cdot j}, \mathbf{y} \rangle \right|, \tag{7}$$

where $\mathbf{x}_{\cdot j}$ is the j-th column of the design matrix $\mathbf{X}$, for $j = 1, \dots, p$.
For tuning parameter value λ_{k+1}, we initialize coordinate descent algorithm at the computed solution for λ_k (warm start). Apart from giving us a path of solutions, this scheme exploits warm starts, and leads to a more stable algorithm.

MIDDLE LOOP In the middle loop, we find the estimates of β by solving the optimization problem (6) for a fixed λ. For each iteration of the middle loop, based on the current parameter estimates $\tilde{\beta}$, we form a quadratic approximation to the log-likelihood f using a Taylor expansion:

$$\begin{aligned}
f(\beta) \approx \ell(\beta) &= f(\tilde{\beta}) + (\beta - \tilde{\beta})^\top \nabla f(\tilde{\beta}) + \frac{1}{2}(\beta - \tilde{\beta})^\top \nabla^2 f(\tilde{\beta})(\beta - \tilde{\beta}) \\
&= f(\tilde{\beta}) + [\mathbf{X}(\beta - \tilde{\beta})]^\top (\mathbf{y} - \tilde{\mathbf{p}}) - \frac{1}{2}[\mathbf{X}(\beta - \tilde{\beta})]^\top \tilde{\mathbf{W}} \mathbf{X}(\beta - \tilde{\beta}) \\
&= f(\tilde{\beta}) + \sum_{i=1}^n (Y_i - \tilde{p}_i) \mathbf{x}_i^\top (\beta - \tilde{\beta}) - \frac{1}{2} \sum_{i=1}^n \tilde{w}_i \left[\mathbf{x}_i^\top (\beta - \tilde{\beta}) \right]^2 \\
&= -\frac{1}{2} \sum_{i=1}^n \tilde{w}_i \left\{ \left[\mathbf{x}_i^\top (\tilde{\beta} - \beta) \right]^2 + 2 \frac{Y_i - \tilde{p}_i}{\tilde{w}_i} \left[\mathbf{x}_i^\top (\tilde{\beta} - \beta) \right] \right\} + f(\tilde{\beta}) \\
&= -\frac{1}{2} \sum_{i=1}^n \tilde{w}_i \left[\mathbf{x}_i^\top (\tilde{\beta} - \beta) + \frac{Y_i - \tilde{p}_i}{\tilde{w}_i} \right] + \frac{1}{2} \sum_{i=1}^n \tilde{w}_i \left(\frac{Y_i - \tilde{p}_i}{\tilde{w}_i} \right)^2 + f(\tilde{\beta}),
\end{aligned}$$

where $\tilde{\mathbf{p}} = (\tilde{p}_1, \dots, \tilde{p}_n)^\top$ and $\tilde{\mathbf{W}} = \mathrm{diag}(\tilde{w}_1, \dots, \tilde{w}_n)$ are the estimates of $\mathbf{p}$ and $\mathbf{W}$ based on $\tilde{\beta}$.
We rewrite the function $\ell(\beta)$ as follows:

$$\ell(\beta) = -\frac{1}{2} \sum_{i=1}^n \tilde{w}_i (\tilde{z}_i - \mathbf{x}_i^\top \beta)^2 + C(\tilde{\beta}), \tag{8}$$

where

$$\tilde{z}_i = \mathbf{x}_i^\top \tilde{\beta} + \frac{Y_i - \tilde{p}_i}{\tilde{w}_i}$$

is the working response, $\tilde{w}_i$ is the working weight, and C is a function that does not depend on β.

INNER LOOP. In the inner loop, we find the estimates of β by solving a modified optimization problem of (6). With fixed $\tilde{w}_i$'s, $\tilde{z}_i$'s, and a fixed form of ℓ based on the estimates of β in the previous iteration of the middle loop, we use coordinate descent to solve the penalized weighted least-squares problem

$$\min_\beta \ -\frac{1}{n} \ell(\beta) + \lambda \sum_{k=1}^p |\beta_k|, \tag{9}$$

and update the estimates of β. For each iteration of the inner loop, suppose we have the current estimates $\tilde{\beta}_k$ for $k \neq j$ and we wish to partially optimize with respect to β_j:

$$\min_{\beta_j} \ \frac{1}{2n} \sum_{i=1}^n \tilde{w}_i \left(\tilde{z}_i - x_{ij} \beta_j - \sum_{k \neq j} x_{ik} \tilde{\beta}_k \right)^2 + \lambda |\beta_j| + \lambda \sum_{k \neq j} |\tilde{\beta}_k|.$$

Updates:

$$\begin{aligned}
\tilde{\beta}_0 &\leftarrow \frac{\sum_{i=1}^n \tilde{w}_i (\tilde{z}_i - \sum_{k=1}^p x_{ik} \tilde{\beta}_k)}{\sum_{i=1}^n \tilde{w}_i}, \\
\tilde{\beta}_j &\leftarrow \frac{S\left(\frac{1}{n} \sum_{i=1}^n \tilde{w}_i x_{ij} (\tilde{z}_i - \sum_{k \neq j} x_{ik} \tilde{\beta}_k), \lambda \right)}{\frac{1}{n} \sum_{i=1}^n \tilde{w}_i x_{ij}^2}, \ j = 1, \dots, p
\end{aligned}$$

where $S(z,\gamma)$ is the soft-thresholding operator with value

$$S(z,\gamma) = \text{sign}(z)(|z|-\gamma)_+ = \begin{cases} z-\gamma, & \text{if } z>0 \text{ and } \gamma<|z| \\ z+\gamma, & \text{if } z<0 \text{ and } \gamma<|z| \\ 0, & \text{if } \gamma \geq |z| \end{cases}$$

We can then update estimates of β_j's repeatedly for $j = 0, 1, 2, ..., p, 0, 1, 2, ...$ until convergence.

- Note: Care is taken to avoid coefficients diverging in order to achieve fitted probabilities of 0 or 1. When a probability is within $\epsilon = 10^{-5}$ of 1, we set it to 1, and set the weights to ϵ. 0 is treated similarly.

Algorithm 2 Path-wise coordinate-wise optimization algorithm

Require: $g(\beta,\lambda) = -\frac{1}{n}f(\beta) + \lambda\sum_{k=1}^{p}|\beta_k|$ - target function, where $f(\beta)$ is given in (5); β_0 - starting value; $\{\lambda_1,\dots,\lambda_m\}$ - a sequence of descending λ's, where $\lambda_1 = \lambda_{max}$ is given in (7); ϵ - tolerance; N_s, N_t - maximum number of iterations of the middle and inner loops

Ensure: $\widehat{\beta}(\lambda_r)$ such that $\widehat{\beta}(\lambda_r) \approx \arg\min_\beta\ g(\beta,\lambda_r),\ r = 1,\dots,m$

$\widetilde{\beta}_0(\lambda_1) \leftarrow \beta_0$
OUTER LOOP
for $r \in \{1,\dots,m\}$, where r is the current number of iterations of the outer loop, **do**
  $s \leftarrow 0$, where s is the current number of iterations of the middle loop
  $g(\widetilde{\beta}_{-1}(\lambda_r),\lambda_r) \leftarrow \infty$
  MIDDLE LOOP
  while $t \geq 2$ and $s < N_s$ **do**
    $s \leftarrow s+1$
    Update $\tilde{w}_i^{(s)}$, $\tilde{z}_i^{(s)}$ $(i = 1,\dots,n)$, and thus $\ell_s(\beta)$ as given in (8) based on $\widetilde{\beta}_{s-1}(\lambda_r)$
    $t \leftarrow 0$, where t is the current number of iterations of the inner loop
    $\widetilde{\beta}_s^{(0)}(\lambda_r) \leftarrow \widetilde{\beta}_{s-1}(\lambda_r)$
    $h_s(\widetilde{\beta}_s^{(-1)}(\lambda_r),\lambda_r) \leftarrow \infty$, where $h_s(\beta,\lambda) = -\frac{1}{n}\ell_s(\beta) + \lambda\sum_{k=1}^{p}|\beta_k|$
    INNER LOOP
    while $\left|h_s(\widetilde{\beta}_s^{(t)}(\lambda_r),\lambda_r) - h_s(\widetilde{\beta}_s^{(t-1)}(\lambda_r),\lambda_r)\right| > \epsilon$ and $t < N_t$ **do**
      $t \leftarrow t+1$
      $\widetilde{\beta}_0^{(t)}(\lambda_r) \leftarrow \sum_{i=1}^{n}\tilde{w}_i^{(s)}\left(\tilde{z}_i^{(s)} - \sum_{k=1}^{p}x_{ik}\widetilde{\beta}_k^{(t-1)}(\lambda_r)\right)\Big/\sum_{i=1}^{n}\tilde{w}_i^{(s)}$
      for $j \in \{1,\dots,p\}$ **do**
        $\widetilde{\beta}_j^{(t)}(\lambda_r) \leftarrow S\left(\frac{1}{n}\sum_{i=1}^{n}\tilde{w}_i^{(s)}x_{ij}\left(\tilde{z}_i^{(s)} - \sum_{k<j}x_{ik}\widetilde{\beta}_k^{(t)}(\lambda_r) - \sum_{k>j}x_{ik}\widetilde{\beta}_k^{(t-1)}(\lambda_r)\right),\lambda_r\right)\Big/\frac{1}{n}\sum_{i=1}^{n}\tilde{w}_i^{(s)}x_{ij}^2$
      end for
    end while
    $\widetilde{\beta}_s(\lambda_r) \leftarrow \widetilde{\beta}_s^{(t)}(\lambda_r)$
  end while
  $\widehat{\beta}(\lambda_r) \leftarrow \widetilde{\beta}_s(\lambda_r)$
  $\widetilde{\beta}_0(\lambda_{r+1}) \leftarrow \widehat{\beta}(\lambda_r)$
end for

2.4. Five-fold Cross Validation for LASSO

Since the Logistic-Lasso model depend on penalty term λ for selection, we further develop 5-fold cross validation to select the parameter λ to obtain the optimized result.

To select the optimal tuning parameter , the original data is randomly shuffled and split into five equally sized groups. One of the groups is used as a test set, while the remaining groups are used as training data. The path-wise coordinate-wise optimization algorithm is then applied to the training data, and AUC scores are calculated for each λ using the test data. This process is repeated until each of the five groups has been used as the test set, and the mean AUC for each λ is computed.

3. Results

5-fold CV Results for Logistic-LASSO

Model Comparison

4. Discussion

4.1. Summary

4.2. Limitations

4.3. Group Contributions

References

Freer, Timothy W., and Michael J. Ulissey. "Screening mammography with computer-aided detection: prospective study of 12,860 patients in a community breast center." Radiology 220.3 (2001): 781-786.

Friedman J, Hastie T, Tibshirani R. Regularization Paths for Generalized Linear Models via Coordinate Descent. J Stat Softw. 2010;33(1):1-22. PMID: 20808728; PMCID: PMC2929880.

Appendices