Predictive Modeling

Department of Mathematical Sciences
Qiuying Sha, Professor

Michigan Tech

Chapter 12. Discriminant Analysis and Other Linear Classification Models

Discriminant or classification techniques seek to categorize samples into groups based on the predictor characteristics

We will discuss:

- Logistic regression
- Linear discriminant analysis (LDA)
- Partial least squares discriminant analysis (PLSDA)
- Penalized models
- Nearest shrunken centroids

12.1 Case Study: Predicting Successful Grant Applications

These data are from a 2011 Kaggle competition sponsored by the University of Melbourne.

Goal:

- Predict whether or not a grant application would be accepted.
- The university sought to understand factors that were important in predicting success.

Data:

- In the contest, 8,708 grants between the years 2005 and 2008 are for training and the test set contained applications from 2009 to 2010.
- Only the training set data contain the outcomes for the grants.

- Predictors:
  - The role of each individual listed on the grant.
  - Several characteristics of each individual on the grant.
  - One or more codes related to Australia's research fields, courses and disciplines (RFCD) classification.
  - The submission date of the grant.
  - The monetary value of the grant, binned into 17 groups.
  - A grant category code which describes the type sponsor as well as a code for the specific sponsor.

Data pre-procession:

- How to encode these data?
- How to deal with missing?
- How to deal with collinearity?
  - A significant number of predictors had pair-wise absolute correlations that were larger than 0.99.
  - A high-correlation filter was used on the predictor set to remove these highly redundant predictors from the data.
  - 56 predictors were eliminated from the data for this reason.

- How to deal with sparse and unbalanced predictors
  - The binary nature of many of predictors also resulted in many cases where the data were very sparse and unbalanced.
  - Many of the predictors could be classified as *near-zero variance predictors*.

Two different sets of predictors were used:

- "full set"-- including all the variables regardless of their distribution (1,070 predictors).
- "reduced set" -- was developed for models that are sensitive to sparse and unbalanced predictors and contained 252 predictors. .

Results from univariate data analysis:

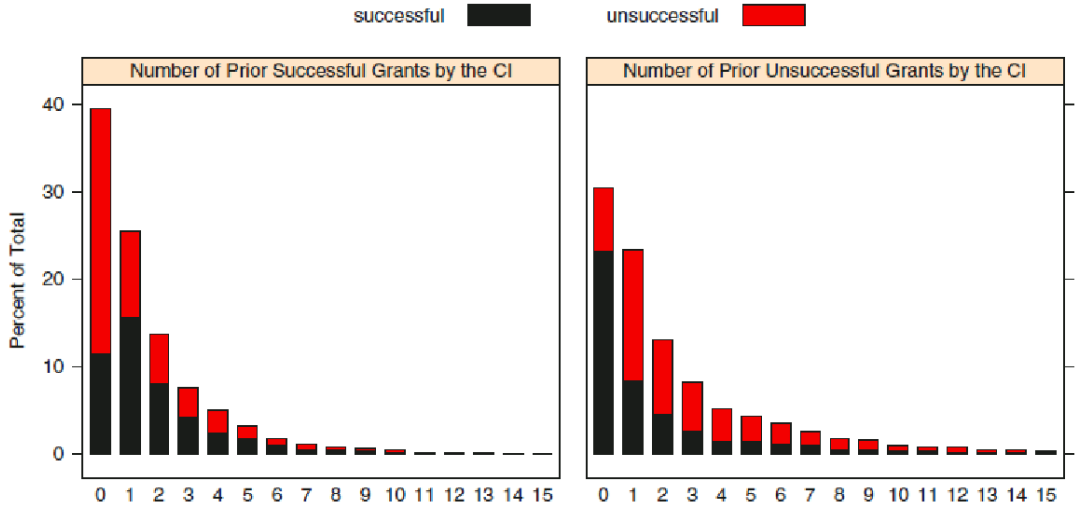


Fig. 12.1: The top two continuous predictors associated with grant success based on the pre-2008 data. Prior success in receiving a grant by the chief investigator as well as prior failure in receiving a grant are most highly associated with the success or failure of receiving a future grant. The x-axis is truncated to 15 grants so that the long tails of the distribution do not obfuscate the differences

- Two continuous predictors, were highly associated with grant application success.
  - ➢ We would expect these predictors to play significant roles for most classification models.

Michigan Tech

Table 12.1: Statistics for the three categorical predictors with highest univariate association with the success funding of a grant

	Grant success					
	Yes	No	N	Percent	Odds	Odds ratio
Contract value band						
A	1,501	818	2,319	64.7	1.835	2.84
Other bands	2,302	3,569	5,871	39.2	0.645	
Sponsor						
Unknown	732	158	890	82.2	4.633	6.38
Known	3,071	4,229	7,300	42.1	0.726	
Month						
January	480	45	525	91.4	10.667	13.93
Other months	3,323	4,342	7,665	43.4	0.765	

- Three categorical predictors had the highest univariate associations with grant application success.
  - Odds is the ratio of the probability of a success grant over the probability of an unsuccessful grant.

- One common method for quantifying the predictive ability of a binary predictor is the *odds ratio* (The ratio of the odds).

For example, when a grant is submitted in January the odds are much higher (10.7) than other months (0.8). The ratio of the odds for this predictor suggests that grants submitted in January are 13.9 times more likely to be successful than the other months.

How to split the data?

- Two facts:
  - The percentage of successful grants varied over the years: 45% (2005), 51.7% (2006), 47.2% (2007), and 36.6% (2008).
  - The purpose is to create a predictive model to quantify the likelihood of success for new grants
- If the grant success rate were relatively constant over the years: take all the available data from 2005 to 2008, reserve some data for a test set, and use resampling with the remainder of the samples for tuning the various models.

- An alternative strategy would be to create models using the data before 2008, but tune them based on how well they fit the 2008 data. Essentially, the 2008 data would serve as a single test set that is more relevant in time to the original test set of data from 2009 to 2010.

Problems on this strategy:

- This is a single “look” at the data that do not provide any real measure of uncertainty for model performance.
- May lead to substantial over-fitting to this particular set of 2008 data and may not generalize well to subsequent years.

How do these two approaches compare for these data?

Figure 12.2 shows the results for a support vector machine classification model.

Using the radial basis function kernel previously discussed, the tuning parameters are the kernel parameter, σ, and the cost value, C, used to control for over-fitting.

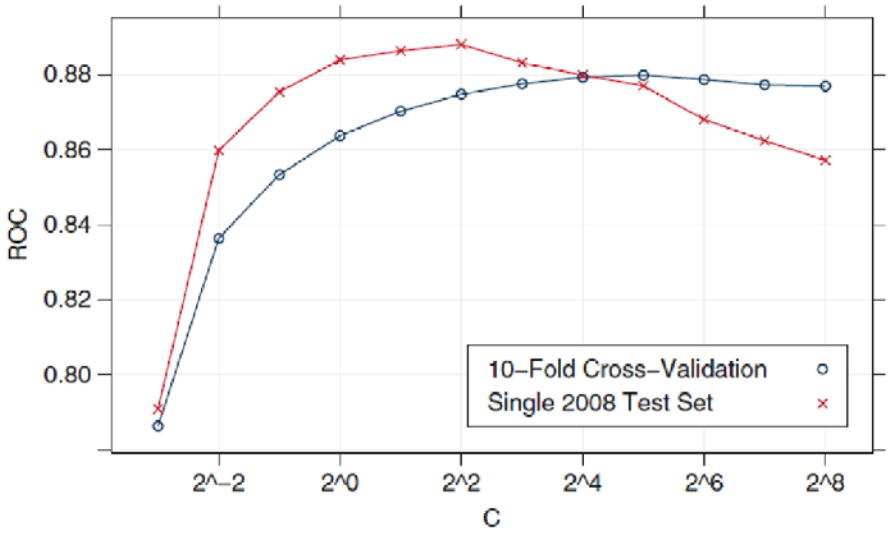


Fig. 12.2: Two models for grant success based on the pre-2008 data but with different data sets used to tune the model

- The first model:
  - Built on 8,189 grants that include all pre-2008 data and 25% of the 2008 data (*n* = 290).

- To choose the regularization and kernel parameter(s), 10-fold cross-validation is used.
- A set of 2008 grants (n = 1, 785) is held back to validate the choice of the final tuning parameter (blue profile).

- The second model:
  - Built on pre-2008 data.
  - The value of the tuning parameter is chosen to maximize the area under the ROC curve for the 2008 grants.
  - No additional samples are held back for verifying the parameter choice (red profile).

- Firstly, given the amount of data to evaluate the model, it is problematic that the curves suggest different tuning parameters.

- Secondly, when the cross-validated model is evaluated on the 2008 data, the area under the ROC curve is substantially smaller (0.83) than the cross-validation results indicate.

The compromise taken here is to

- Build models on the pre-2008 data.
- Tune them by evaluating a random sample of 2,075 grants from 2008.
- Once the optimal parameters are determined, final model is built using these parameters and the entire training set (i.e., the data prior to 2008 and the additional 2,075 grants).
- A small holdout set of 518 grants from 2008 will be used to ensure that no gross methodology errors occur from repeatedly evaluating the 2008 data during model tuning.
  - This set of samples is called the *2008 holdout set*. This small set of year 2008 grants will be referred to as the *test set* and will not be evaluated until set of candidate models are identified.

12.2 Logistic

Like in linear regression models, use *maximum likelihood estimates (MLEs)* to estimate parameters in logistic regression.

We will discuss:

- For two classes, we assume a binomial distribution for the response variable.
- *p* is the probability of an event or a specific class;

  n is the total sample points;

  r is the number of sample points in the specific class.
- Then, the likelihood function is

$$L(p) = \binom{n}{r} p^r (1-p)^{n-r}$$

In the pre-2008 grants, n = 6,633, r = 3,233.

- The likelihood function is

$$L(p) = \binom{6633}{3233} p^{3233}(1-p)^{6633-3233}$$

- The MLE would find a value of *p* that produces the largest value for *L*(*p*).
- It turns out that the sample proportion, r/n = 3233/6633 = 0.487, is the MLE in this situation.

The success rate, p, is affected by multiple factors.

Build a model that uses those factors to produce a more refined probability estimate.

Logistic regression models the log odds of the event as a linear function

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_P x_P.$$

P is the number of predictors.

This model is equivalent to

$$p = \frac{1}{1 + \exp\left[-(\beta_0 + \beta_1 x_1 + \cdots + \beta_P x_P)\right]}$$

We then relate our model to the parameter of the binomial distribution,

- We can find candidate values of the parameters (*β*), compute a value of the likelihood function.
- Once we find *β* values that maximize the likelihood for our data, these values would be used to predict sample outcomes.
- Logistic regression and ordinary linear regression fall into a larger class of techniques called generalized linear models (GLMs).

We could fit a simple logistic regression model to the grant data using a single predictor, such as the numeric day of the year

- $\widehat{\beta_0} = 0.919$, $\widehat{\beta_1} = -0.0042$, and $\widehat{\beta_2} =$. This means that there was a per day *decrease* in the log odds of 0.0042.

Another model can be created where a third parameter corresponds to a squared day term.

- $\widehat{\beta_0} = 1.88$, $\widehat{\beta_1} = -0.019$, and $\widehat{\beta_2} =$-0.000038

For the grant data, the full set of predictors was used in a logistic regression model.

- ROC curve is 0.78, sensitivity is 77%, and specificity is 76.1% on the 2008 holdout set.

Many of the categorical predictors have sparse and unbalanced distributions.

We would expect that a model using the full set of predictors would perform worse than the set that has near-zero variance predictors removed.

- For the reduced set of 253 variables, the ROC curve is 0.87, the sensitivity is 80.4%, and the specificity is 82.2%.

There was a substantial improvement gained by removing these predictors.

For logistic regression, formal statistical hypothesis tests can be conducted to assess whether the slope coefficients for each predictor are statistically significant.

- A *Z* statistic is commonly used for these models, and is essentially a measure of the signal-to-noise ratio: the estimated slope is divided by its corresponding standard error.
- Using this statistic, the predictors can be ranked to understand which terms had the largest effect on the model.

The **five most important predictors** were

- The number of unsuccessful grants by chief investigators,
- The number of successful grants by chief investigators,
- Contract value band F,
- Contract value band E, and
- Numeric day of the year (squared).

Logistic model requires the user to identify effective representations of the predictor data that yield the best performance.

There are other classification models that empirically derive these relationships in the course of model training.

If the model will only be utilized for prediction, these techniques may be more advantageous.

12.3 Linear Discriminant Analysis (LDA)

The roots of LDA date back to Fisher (1936) and Welch (1939).

- Each of these researchers took a different perspective on the problem of obtaining optimal classification rules.
- Each came to find the same rule in the two-group classification setting.

Welch (1939)'s approach: minimizing the total probability of misclassification

- $\Pr(Y = C_l)$ is the probability in class C_l , known as the *prior probability*.
- $\Pr(X|Y = C_l)$ is the *conditional probability* of observing predictors *X*, given that the data stem from class C_l

Then the *posterior probability* is given by

$$Pr[Y = C_\ell | X] = \frac{Pr[Y = C_\ell] Pr[X | Y = C_\ell]}{\sum_{l=1}^{C} Pr[Y = C_l] Pr[X | Y = C_l]}$$

For a two-group classification problem, the rule that minimizes the total probability of misclassification would be to

- Classify *X* into group 1 if

$$\Pr(Y = C_1 \mid X) > \Pr(Y = C_2 \mid X)$$

- Into group 2 if the inequality is reversed.

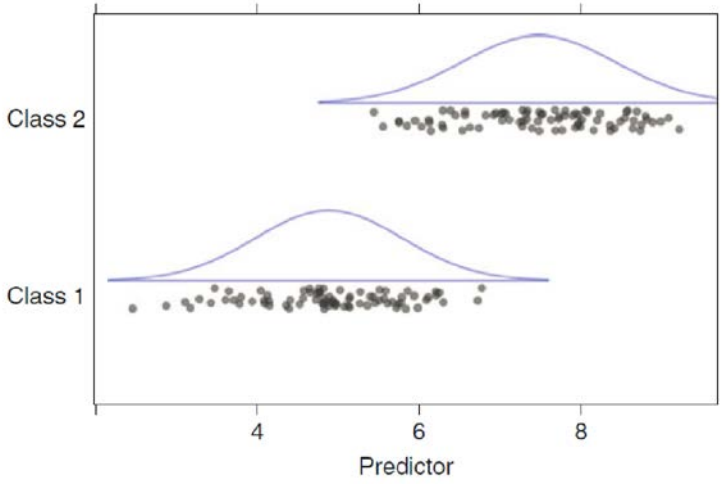
Using the above equation, this rule directly translates to classifying *X* into group 1 if

$$\Pr(Y = C_1) \Pr(X \mid Y = C_1) > \Pr(Y = C_2) \Pr(X \mid Y = C_2)$$

We can easily extend this rule to the more-than-two group case.

- We classify *X* into C_l if $\Pr(Y = C_l|X) = \Pr(Y = C_l)\Pr(X|Y = C_l)$ *has the largest value across all of the* C_l classes.

Figure 12.5 illustrates this with a single predictor and two classes.



How does one compute quantities such as $\Pr(X|Y = C_l)$ in many dimensions?

What multivariate probability distributions can be used to this effect?

- Often used scenario is to assume that the distribution of the predictors is multivariate normal.
- This distribution has two parameters: the multidimensional mean vector μ_l and covariance matrix Σ_l
- We assume that the means of the groups are unique (i.e., a different μ_l for each group), but the covariance matrices Σ_l are identical across groups.

Notations:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \\ x_p \end{bmatrix}, \ \boldsymbol{\mu} = E(\mathbf{x}) = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \\ \mu_p \end{bmatrix} \text{ and } \Sigma = \text{Cov}(\mathbf{x}) = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2p} \\ & & \dots & \\ \sigma_{p1} & \sigma_{p2} & \dots & \sigma_{pp} \end{bmatrix}$$

The density function of the MVN distribution $MVN(\boldsymbol{\mu}, \Sigma)$ is:

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}} exp[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})'\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})];$$

$$\log f(\mathbf{x}) = \log\left(\frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}}\right) - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})'\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})$$

$$= \text{constant} - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})'\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}).$$

If we assume that $\mathbf{x} \mid Y = C_l \sim MVN(\boldsymbol{\mu}_l, \Sigma)$, then

$$\log\left[\Pr(Y = C_l)\Pr(\mathbf{x} \mid Y = C_l)\right]$$

$$= \log\left[\Pr(Y = C_l)\right] + \log\left[\Pr(\mathbf{x} \mid Y = C_l)\right]$$

$$= \log\left[\Pr(Y = C_l)\right] + \text{constant} - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_l)'\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_l)$$

$$= \log\left[\Pr(Y = C_l)\right] + \text{constant} - \frac{1}{2}\left[\mathbf{x}'\Sigma^{-1}\mathbf{x} - \mathbf{x}'\Sigma^{-1}\boldsymbol{\mu}_l - \boldsymbol{\mu}_l'\Sigma^{-1}\mathbf{x} + \boldsymbol{\mu}_l'\Sigma^{-1}\boldsymbol{\mu}_l\right]$$

$$= \log\left[\Pr(Y = C_l)\right] + \text{constant} - \frac{1}{2}\left[\mathbf{x}'\Sigma^{-1}\mathbf{x} - 2\mathbf{x}'\Sigma^{-1}\boldsymbol{\mu}_l + \boldsymbol{\mu}_l'\Sigma^{-1}\boldsymbol{\mu}_l\right]$$

$$= \log\left[\Pr(Y = C_l)\right] + \text{constant} - \frac{1}{2}\mathbf{x}'\Sigma^{-1}\mathbf{x} + \mathbf{x}'\Sigma^{-1}\boldsymbol{\mu}_l - \frac{1}{2}\boldsymbol{\mu}_l'\Sigma^{-1}\boldsymbol{\mu}_l$$

To compare $\Pr(Y = C_l)\Pr(\mathbf{x} \mid Y = C_l)$ for different l, we only need to compare

$$\log\left[\Pr(Y = C_l)\right] + \mathbf{x}'\Sigma^{-1}\boldsymbol{\mu}_l - \frac{1}{2}\boldsymbol{\mu}_l'\Sigma^{-1}\boldsymbol{\mu}_l.$$

- The **linear discriminant function of l^{th} group**.

$$X'\mathbf{\Sigma}^{-1}\mu_\ell - 0.5\mu_\ell^{'}\mathbf{\Sigma}^{-1}\mu_\ell + \log\left(Pr[Y = C_\ell]\right).$$

- We classify X to l^{th} group if it is the largest one.

In practice,

- The theoretical means, μ_l are estimated by the class-specific means ($\bar{X}_l$).
- The theoretical covariance matrix, Σ_l, is estimated by the observed covariance matrix of the data, S, and
- *X* is replaced with an observed sample.

Note: the linear discriminant function of l^{th} group is a linear function in *X.*

If we assume that the covariance matrices are not identical across the groups, it will lead to quadratic discriminant analysis described in Sect. 13.1.

Fisher's approach:

Fisher formulated the classification problem in a different way.

- Find the linear combination of the predictors such that the between-group variance was maximized relative to the within-group variance.

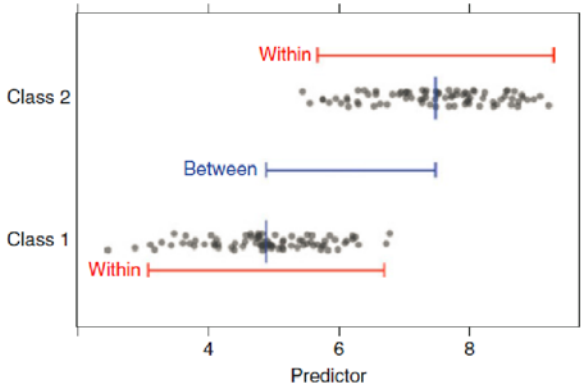


Fig. 12.6: The same data as shown in Fig. 12.5. Here, the between- and within-class variances are illustrated. The within-class ranges are based on the mean ± two standard deviations

- Find the combination of the predictors that gave maximum separation between the centers of the data while at the same time minimizing the variation within each group of data.

- Determines linear combinations of the predictors to maximize the signal-to-noise ratio.

- The between group variance is the square of the difference in these means.
- The within-group variance would be estimated by a variance that pools the variances of the predictor within each group.
- Taking a ratio of these two quantities is a signal-to-noise ratio.

Let

- **B** represent the between-group covariance matrix
- **W** represent the within-group covariance matrix.

Then Fisher's approach: find the value of *b* to maximize

$$\frac{b'\mathbf{B}b}{b'\mathbf{W}b} \qquad (12.7)$$

- The solution is the eigenvector corresponding to the largest eigenvalue of $\mathbf{W}^{-1}\mathbf{B}$.
- This vector is a linear discriminant.

Let's consider the two-group setting.

- Solving Eq. 12.7 for two groups gives the discriminant function of $S^{-1}(\overline{X_1} - \overline{X_2})$, where S^{-1} is the inverse of the covariance matrix of the data and is multiplied by the difference between the mean vectors of predictors for each group.

In practice, a new sample, **u**, is projected onto the discriminant function as $u'S^{-1}(\overline{X_1} - \overline{X_2})$, which returns a discriminant score.

A new sample is then classified into group 1 if the sample is closer to the group 1 mean than the group 2 mean in the projection:

$$\left|b'\left(\mathbf{u} - \overline{\mathbf{x}}_1\right)\right| - \left|b'\left(\mathbf{u} - \overline{\mathbf{x}}_2\right)\right| < 0.$$

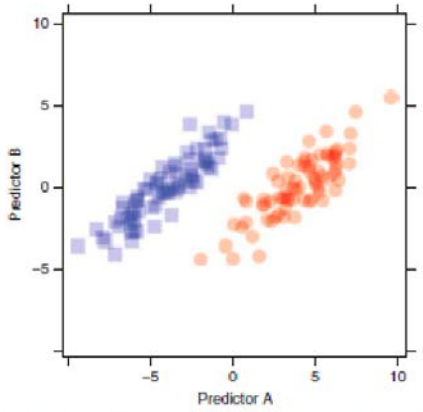


Fig. 12.7: A simple example of two groups of samples that are clearly separable

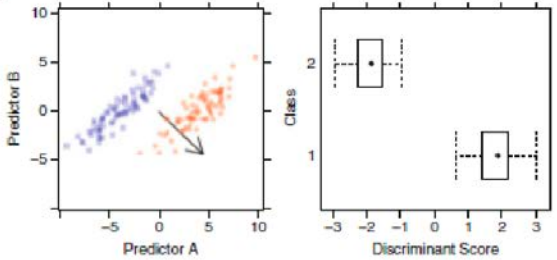


Fig. 12.8: The line at approximately $A = B$ is the vector that visually separates the two groups. Assessing class membership is determined by projecting a sample onto the discriminant vector (*red arrow*) and then calculating its distance from the mean for each group. The sample is then classified into the group which mean is closer. The *box plots* are the distribution of the samples for each class after LDA has been performed, illustrating the maximization of between-to-within group variation

Pre-processing:

Practitioners should be particularly rigorous in pre-processing data before using LDA.

The authors recommend that

- predictors be centered and scaled and
- near-zero variance predictors be removed.
- If the covariance matrix is still not invertible, then they recommend using PLS or a regularization approach.

The usual LDA approach cannot be directly used when

- Predictors are highly correlated or
- The number of predictors exceeds the number of samples collected.

12.4 Partial Least Squares Discriminant Analysis (PLSDA)

The usual LDA approach cannot be directly used when

- Predictors are highly correlated or
- The number of predictors exceeds the number of samples collected.

To solve these problems, we can attempt to pre-process our predictors.

- Removes highly correlated predictors.
- If more complex correlation structure exist in the data or if the number of predictors still exceeds the number of samples (or the ratio of samples to predictors is too low), PCA can be used to reduce the predictor-space dimension.
  - However, PCA may not identify the predictor combinations that optimally separate samples into groups, because PCA does not take into consideration any of the response classification information.

- Instead of taking this stepwise approach (PCA-then-LDA) to the overdetermined problem, PLS is recommended.

Extension of PLS to the classification setting is called PLS discriminant analysis (or PLSDA).

Recall PLS in chapter 6, PLS finds latent variables that simultaneously reduce dimension and maximize correlation with a continuous response value (see Fig. 6.9).

In the classification setting for a two-group problem, we could use the samples' class value (represented by 0's and 1's) as the response for this model.

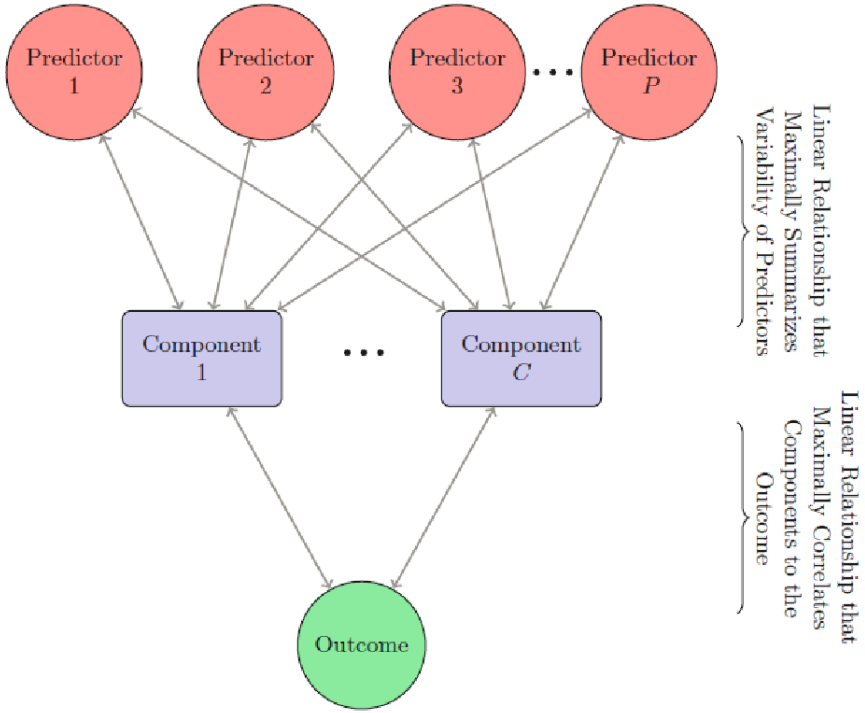


Fig. 6.9: A diagram depicting the structure of a PLS model. PLS finds components that simultaneously summarize variation of the predictors while being optimally correlated with the outcome

Given what we know about PLS for regression, we would then expect that the latent variables would be selected to reduce dimension while optimizing correlation with the categorical response vector.

- Of course, optimizing correlation isn't the natural objective if classification is the goal—rather, minimizing misclassification error or some other objective related to classification would seem like a better approach.

- Despite this fact, PLSDA should do better since the group information is being considered while trying to reduce the dimension of the predictor space.

Even though a correlation criterion is being used by PLS for dimension reduction with respect to the response, it turns out that this criterion happens to be doing the right thing.

- One tuning parameter: the number of latent variables to be retained.

- PLS can be viewed as a *supervised* dimension reduction procedure; PCR is an *unsupervised* procedure.

- Prior to performing PLS, the predictors should be centered and scaled.

12.5 Penalized Models

Many classification models utilize penalties (or regularization) to improve the fit to the data.

For example, one might include a penalty term for the logistic regression model in a manner that is very similar to ridge regression.

- Logistic regression finds parameter values that maximizes the likelihood function, $L(p)$.
- A simple approach to regularizing this model would be to add a squared penalty function to the log likelihood and find parameter estimates that maximize

$$\log L(p) - \lambda \sum_{j=1}^{P} \beta_j^2.$$

  - ➢ When there are a large number of predictors and a small training set sample, the penalty term can stabilize the logistic regression model coefficients.

- When there are highly correlated predictors, adding a penalty can also provide a countermeasure against highly correlated predictors.

The glmnet models uses **ridge and lasso** penalties simultaneously, like the elastic net, but structures the penalty slightly differently:

$$\log L(p) - \lambda \left[(1-\alpha)\frac{1}{2}\sum_{j=1}^{P}\beta_j^2 + \alpha\sum_{j=1}^{P}|\beta_j| \right].$$

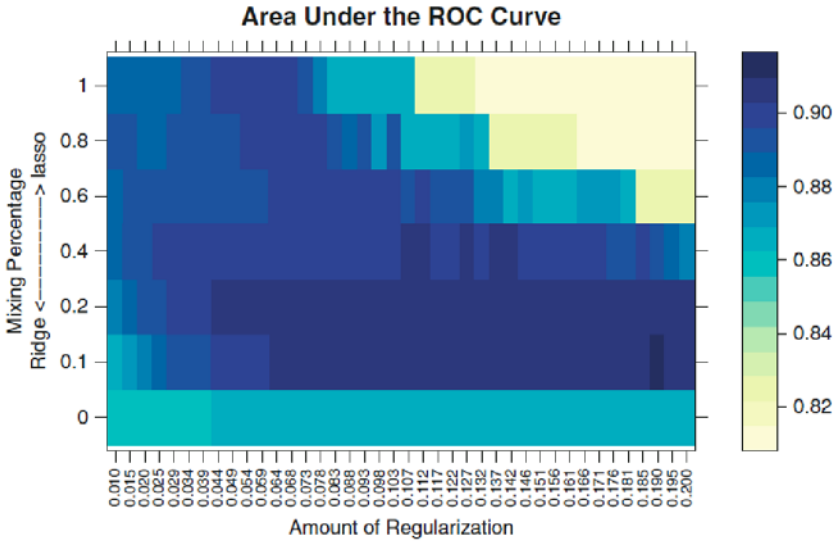
- The α value is the "mixing proportion" that toggles between the pure lasso penalty (when $\alpha = 1$) and a pure ridge-regression-like penalty ($\alpha = 0$).

- The other tuning parameter λ controls the total amount of penalization.

Applications:

The glmnet model is applied to the grant data:

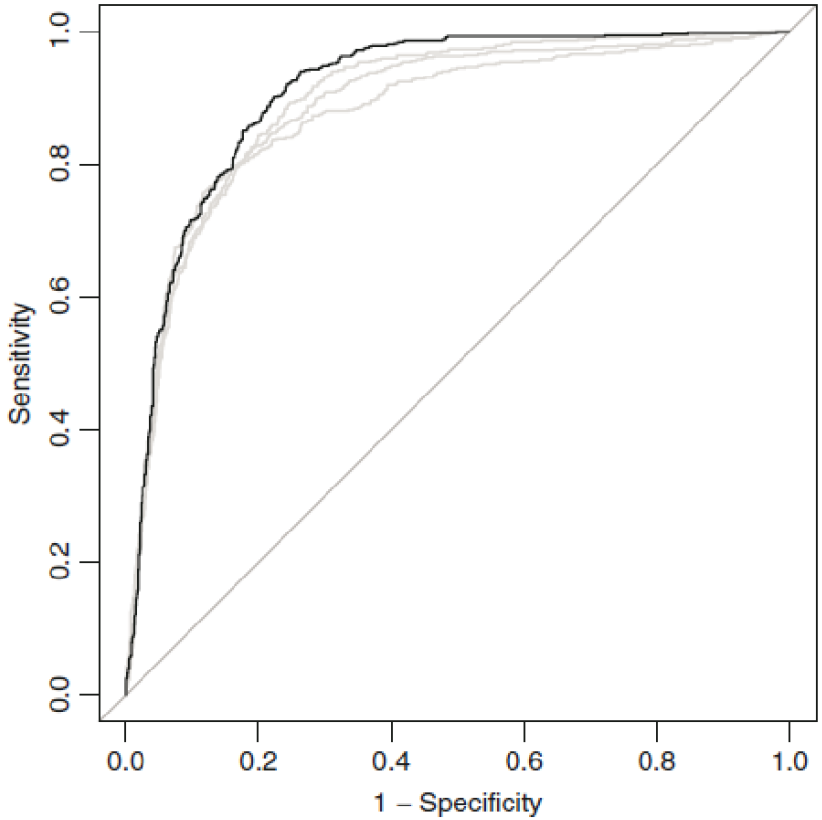
- The glmnet model was tuned over seven values of the mixing parameter *α* and 40 values of the overall amount of penalization.



- The full set of predictors was used in the model.

Results:

- A heat map of the area under ROC curve for these models.



- The data favor models with a larger mix of the ridge penalty than the lasso penalty, although there are many choices in this grid that are comparable.
- In the end, the numerically optimal settings are a mixing percentage of 0.1 and a value of 0.19 for the regularization amount.
- These settings had the effect of using **only 44 predictors out of 1,070** in the final glmnet model, which achieved an area under the ROC curve of **0.91**.

Notes:

- The previous logistic regression model which used the reduced set of predictors resulted in an **AUC of 0.87**, indicating that the methodical removal of noninformative predictors increased the effectiveness of the model.

- Penalization strategies can be applied to LDA models.

- The same lasso penalty has also been applied to PLS discriminant models so that some of the PLS loadings are also eliminated.

12.6 Nearest Shrunken Centroids

The nearest-shrunken centroid model

- is also known as PAM, for predictive analysis for microarrays.
- is a linear classification model that is well suited for high-dimensional problems (Tibshirani et al. 2002, 2003; Guo et al. 2007).

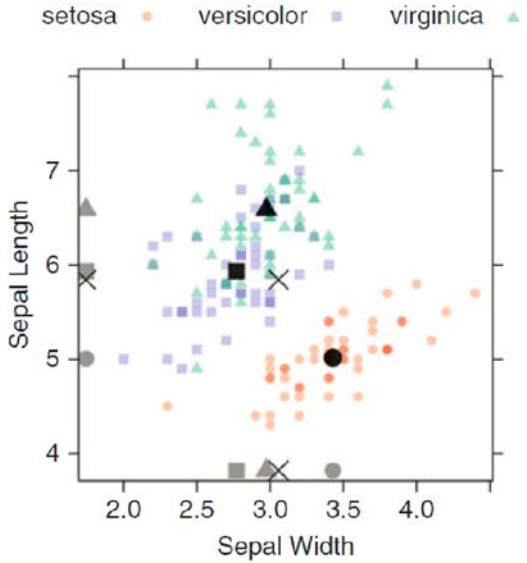
For each class, the centroid of the data is found by taking the average value of each predictor (per class) in the training set.

The overall centroid is computed using the data from all of the classes.

If a predictor does not contain much information for a particular class, its centroid for that class is likely to be close to the overall centroid.

Consider the three class data.

- This data set is the famous Fisher/Anderson iris data.
- Four measurements of iris sepals and petals are used to classify flowers into one of three different iris species: setosa, versicolor, and virginica.



Findings:

- The virginica centroid (for sepal width) is very close to the overall centroid.
- The versicolor centroid is slightly closer to the overall centroid than the setosa flowers.
- This indicates that the sepal width predictor is most informative for distinguishing the setosa species from the other two.

- For sepal length, the versicolor centroid is very close to the center of the data and the other two species are on the extremes.

One approach to classifying unknown samples would be

- to find the closest class centroid in the full dimensional space and choose that class for prediction (i.e., a "nearest centroid" model).

- It turns out that this approach would result in linear class boundaries.

The approach taken by Tibshirani et al. (2002) is to shrink the class centroids closer to the overall centroid.

- Centroids that start off closer to the overall centroid move to that location before others.

  - For example, in the sepal width dimension, the virginica centroid will reach the center before the other two.

- For this model, once the class centroid meets the overall centroid, it no longer influences the classification of samples for that class.
- The nearest shrunken centroid model also conducts feature selection during the model training process.
- The nearest shrunken centroid method has one tuning parameter: shrinkage.

Nearest centroid classification

- Training procedure: given labeled training $\{(\overrightarrow{x_1}, y_1), \ldots, (\overrightarrow{x_n}, y_n)\}$ with class labels $y_1 \in Y$, compute the per-class centroids $\overrightarrow{u_l} = \frac{1}{|C_l|}\sum_{i \in C_l} \overrightarrow{x_i}$, where C_l is the set of indices of samples belonging to class $l \in Y$.
- Prediction function: the class assigned to an observation $\vec{x}$ is

$$\hat{y} = \arg\min_{l \in Y} \| \overrightarrow{\mu_l} - \overrightarrow{\mathrm{x}} \|$$

Briefly,

- The method computes a standardized centroid for each class.
- Nearest centroid classification takes a new sample, and compares it to each of these class centroids. The class whose centroid that it is closest to, in squared distance, is the predicted class for that new sample.

Nearest shrunken centroid classification

Nearest shrunken centroid classification makes one important modification to standard nearest centroid classification.

- It "shrinks" each of the class centroids toward the overall centroid for all classes by an amount we call the **threshold**.
- This shrinkage consists of moving the centroid towards zero by **threshold**, setting it equal to zero if it hits zero.

- For example if **threshold** was 2.0, a centroid of 3.2 would be shrunk to 1.2, a centroid of -3.4 would be shrunk to -1.4, and a centroid of 1.2 would be shrunk to zero.

- After shrinking the centroids, the new sample is classified by the usual nearest centroid rule, but using the shrunken class centroids.

This shrinkage has two advantages:

1) It can make the classifier more accurate by reducing the effect of noisy predictors,

2) It does automatic selection of predictors.

In particular, if a predictor is shrunk to zero for all classes, then it is eliminated from the prediction rule.

The user decides on the value to use for **threshold**.

- Typically one examines a number of different choices.
  - ➢ To guide in this choice, PAM does K-fold cross-validation for a range of threshold values.
  - ➢ The cross-validated misclassification error rate is reported for each threshold value.
- Typically, the user would choose the threshold value giving the minimum cross-validated misclassification error rate.
- Pre-Processing: center and scale predictors.

Hints:

12.2
For more than two classes, you can use

```
ctrl <- trainControl(summaryFunction = defaultSummary)
```

or not use anything.

For logistic model, change glm to multinom

```
Logit <- train(xTrain, yTrain,
               method = "multinom",
               metric = "Accuracy", trControl = ctrl)
Logit
summary(Logit)
```

###12.3:
There are four predictors that are not numeric. You need to add dummy variables (dummyVar in R) for these predictors.

Create dummy vars for state, area_code, international_plan, voice_mail_plan

```
library(caret)
dummRes<-dummyVars("~state+area_code+international_plan+
                                voice_mail_plan", data=data name, fullRank=TRUE)
churn_pred_test_dumm<-data.frame(predict(dummRes, newdata=data name))
```

If you can not figure out how to add dummy variables, you can use as.integer to code predictor values as integer (This is not recommended in practice)

```
# Converting string predictors to numbers
churnTrain_1$state <- as.integer(unclass(factor(churnTrain_1$state)))
table(churnTrain_1$state)   ## to check for the distribution
churnTrain_1$area_code <- as.integer(unclass(factor(churnTrain_1$area_code)))
churnTrain_1$international_plan <- as.integer(churnTrain_1$international_plan)
churnTrain_1$voice_mail_plan <- as.integer(churnTrain_1$voice_mail_plan)
```

13.1

For continuous predictors, you can plot parallel boxplots for “yes” and “no” to compare the means, using

```
for(i in 6:17)
{
        Sys.sleep(0.1);
        boxplot(pre.x[,i]~pre.x[,19], main = names(pre.x)[i])
}
```

For categorical predictors, you can use histograms to compare the distributions between “yes” and “no”, using

```
counts <- table(catePredictor[,i], Classes)
barplot(counts, main=names(catePredictor)[i],
# col=c(3,5,6,7,8,2),
legend = rownames(counts), beside=TRUE)
```