Chapter 4 Notes: Classification

Statistical Learning with R

Jon Geiger

February 3, 2022

Classification is a statistical method used with a *qualitative* response variable.

Some classifiers covered in this chapter include logistic regression, linear discriminant analysis, quadratic discriminant analysis, naive Bayes, and K-nearest neighbors. These topics are used to segue into Generalized Linear Models and Poisson Regression.

4.1: An Overview of Classification

The `Default` data set will be used extensively, in predicting whether an individual will default on their credit card payment, on the basis of their annual income and monthly credit card balance.

We'll be building a model to predict `default` from `balance` (X_1) and `income` (X_2).

4.2: Why Not Linear Regression?

Linear regression doesn't work for classification problems because in order to predict an outcome, we need to *numerically encode* the categories as a quantitative response variable. This doesn't work because the categories rarely ever have any logical order.

In the case of predicting medical diagnoses, a response variable might look like:

$$Y = \begin{cases} 1 & \text{if } \texttt{stroke}, \\ 2 & \text{if } \texttt{drug overdose}, \\ 3 & \text{if } \texttt{epileptic seizure}. \end{cases}$$

The situation improves slightly if we choose to use the *dummy variable* approach, where we code a response variable which looks like:

$$Y = \begin{cases} 0 & \text{if } \texttt{stroke}; \\ 1 & \text{if } \texttt{drug overdose}. \end{cases}$$

In this case, we could have some estimates outside of the $[0, 1]$ range, which leads to difficulty interpreting probabilities.

Overall: (1) regression cannot accommodate non-binary classification, and (b) regression methods will not provide meaningful estimates of $\Pr(Y|X)$, even with just two classes.

4.3: Logistic Regression

Consider the binary classification problem, namely with the `Default` data set.

Logistic regression models the *probability* that Y belongs to a particular class, rather than modeling the classification directly.

The probability of defaulting given a certain balance can be expressed as

$$\Pr(\texttt{balance} = \texttt{Yes}|\texttt{balance}) \equiv p(\texttt{balance})$$

In general, we notate that $p(X) = \Pr(Y = 1|X)$. In logistic regression, we use the logistic function, which we can rearrange to create a linear regression problem.

$$\begin{aligned} p(X) &= \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}} && \text{Logistic Function} \\ \therefore \quad \frac{p(X)}{1-p(X)} &= e^{\beta_0+\beta_1 X} && \text{Odds} \\ \log\left(\frac{p(X)}{1-p(X)}\right) &= \beta_0+\beta_1 X && \text{Log Odds / Logit} \end{aligned}$$

Odds close to zero indicate low probabilities of default, and values close to ∞ indicate high probabilities of default. Interpreting this final equation is a bit tricky, as a one-unit increase in X will yield a β_1 increase in the log odds.

The coefficients β_0 and β_1 must be estimated to best fit our training data. This is done using *maximum likelihood estimation*. The likelihood function is:

$$\ell(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1-p(x_{i'}))$$

The estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ are chosen to maximize this likelihood function.

In a problem of inference looking for association between `default` and `balance`, our null hypothesis would be $H_0 : \beta_1 = 0$, which makes sense because a one-unit increase in X should not affect Y at all. So the null logistic function will be $p(X) = \frac{e^{\beta_0}}{1+e^{\beta_0}}$. Similarly to linear regression, the z-statistic associated with β_1 is $z = \hat{\beta}_1/\text{SE}(\hat{\beta}_1)$.

For making predictions, we plug our estimates into the logistic function, so we have the equation:

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 X}}{1+e^{\hat{\beta}_0+\hat{\beta}_1 X}}$$

Multiple Logistic Regression extends nicely out from simple logistic regression, where we can generalize our log odds equation from earlier:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0+\beta_1 X_1+\cdots+\beta_p X_p$$

Where we have p predictors. This also means that our logistic equation for making predictions becomes:

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

We would also use maximum likelihood estimation to estimate all the coefficients $\beta_0, \beta_1, \ldots, \beta_p$.

What happens when we want to classify something which has more than one outcome? In the previous section we dealt with the three classes being `stroke`, `drug overdose`, and `epileptic seizure`. This is a problem of **multinomial logistic regression**. We select a single class to act as the *baseline*, then we say that:

$$\Pr(Y=k|X=x) = \begin{cases} \dfrac{e^{\beta_{k0}+\beta_{k1}x_1+\cdots+\beta_{kp}x_p}}{1+\sum_{l=1}^{K-1} e^{\beta_{l0}+\beta_{l1}x_1+\cdots+\beta_{lp}x_p}} & \text{for } k=1,\ldots,K-1, \\ \dfrac{1}{1+\sum_{l=1}^{K-1} e^{\beta_{l0}+\beta_{l1}x_1+\cdots+\beta_{lp}x_p}} & \text{for } k=K. \end{cases}$$

Additionally, it can be shown that

$$\log\left(\frac{\Pr(Y=k|X=x)}{\Pr(Y=K|X=x)}\right) = \beta_{k0}+\beta_{k1}x_1+\cdots+\beta_{kp}x_p$$

An alternative coding for multinomial logistic regression is known as *softmax* coding, where, rather than selecting a baseline class, we treat all K classes symmetrically, and assume that for $k=1,\ldots,K$,

$$\Pr(Y=k|X=x) = \frac{e^{\beta_{k0}+\beta_{k1}x_1+\cdots+\beta_{kp}x_p}}{1+\sum_{l=1}^{K} e^{\beta_{l0}+\beta_{l1}x_1+\cdots+\beta_{lp}x_p}}$$

So, we actually estimate coefficients for all K classes rather than just for $K-1$ classes. As a result of this, the log odds ratio between the kth and k'th classes is

$$\log\left(\frac{\Pr(Y=k|X=x)}{\Pr(Y=k'|X=x)}\right) = (\beta_{k0}-\beta_{k'0})+(\beta_{k1}-\beta_{k'1})x_1+\cdots+(\beta_{kp}-\beta_{k'p})x_p$$

4.4: Generative Models for Classification

An alternative model for classification makes use of *Bayes' Theorem*, which states that

$$\Pr(Y|X) = \frac{P(X|Y)\cdot P(Y)}{P(X)}$$

This is not how the textbook defines Bayes' Theorem, so we will derive the textbook definition from this representation to make sense of the book's definition.

When talking about multinomial classification, let us assume that we have K total classes, and we want to find the probability that an observation belongs to a class k given information about that observation X. We can express this with Bayes' Theorem as:

$$\Pr(Y=k|X=x) = \frac{\Pr(X=x|Y=k)\cdot\Pr(Y=k)}{\Pr(X=x)}$$

Cleaning this up a bit, we can rewrite some of the probabilities:

$$\Pr(k|x) = p_k(x) = \frac{\Pr(x|k) \cdot \Pr(k)}{\Pr(x)}$$

If we have classes $k = 1, 2, 3, \ldots, K$, we can use the law of total probability to expand this formula a bit in order to figure out what $\Pr(x)$ is.

$$\begin{aligned}
p_k(x) &= \frac{\Pr(x|k) \cdot \Pr(k)}{\Pr(x)} \\
&= \frac{\Pr(x|k) \cdot \Pr(k)}{\Pr(x \cap 1) + \Pr(x \cap 2) + \cdots + \Pr(x \cap K)} \\
&= \frac{\Pr(x|k) \cdot \Pr(k)}{\Pr(x|1) \cdot \Pr(1) + \Pr(x|2) \cdot \Pr(2) + \cdots + \Pr(x|K) \cdot \Pr(K)} \\
&= \frac{\Pr(x|k) \cdot \Pr(k)}{\sum_{l=1}^{K} \Pr(x|l) \cdot \Pr(l)} \\
&= \frac{\Pr(k) \cdot \Pr(x|k)}{\sum_{l=1}^{K} \Pr(l) \cdot \Pr(x|l)}
\end{aligned}$$

If we let $f_k(x)$ be the *(probability) density function* of X for an observation which comes from class k, and we let π_k represent the probability that an observation comes from class k (also called the **prior probability**), then we have:

$$p_k(x) = \frac{\pi_k f_k(x)}{\sum_{\ell=1}^{K} \pi_\ell f_\ell(x)}$$

Rather than directly modeling $p_k(x)$ (also called the **posterior probability**) as we do with logistic regression, we can estimate the different prior probabilities $\pi_1, \ldots, \pi_K$, and the various density functions for the K classes $f_1(x), \ldots, f_K(x)$. The prior probabilities are relatively easy to estimate by taking a random sample, but estimating $f_k(x)$ proves to be a slightly bigger challenge.

We'll talk about three classifiers that all use different estimates of $f_k(x)$: *linear discriminant analysis*, *quadratic discriminant analysis*, and *naive Bayes*.

Let's assume the following:

- We have $p = 1$ predictors.
- $f_k(x)$ is *normal* or *Gaussian*. In order to approximate $f_k(x)$, we have to make some assumptions about its shape.
- The variances of all the classes are equal. That is, that there is a shared variance term among all the classes, σ^2.

Then,

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_k)^2\right)$$

and

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_k)^2\right)}{\sum_{\ell=1}^{K} \pi_\ell \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_\ell)^2\right)}$$

Our goal with classification is to choose the class k such that $p_k(x)$ is maximized. Maximizing $p_k(x)$ is equivalent to maximizing its logarithm, thus we want to assign x to a class k for which $\delta_k(x)$ is maximized, where $\delta_k(x)$ is given by:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

In reality, we do not know the parameters of the Gaussian distribution from which we assume the observations come, we use *Linear Discriminant Analysis* to replace the parameters with statistics. We use the following estimations:

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i \qquad \hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2 \qquad \hat{\pi}_k = \frac{n_k}{n}$$

The LDA classifier then assigns an observation $X = x$ to the class for which the *discriminant function* $\hat{\delta}_k(x)$ is the largest:

$$\hat{\delta}_k(x) = x \cdot \frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}^2} + \log(\hat{\pi}_k)$$

Linear Disciminant Analysis is *linear* because the discriminant functions are linear functions of x.

When we have $p > 1$, we need to assume a multivariate Gaussian distribution with a class-specific mean vector and a common covariance matrix.

Without going through all the details, the Bayes classifier assigns an observation $X = x$ to the class for which $\delta_k(x)$ is the largest, where we have:

$$\delta_k(x) = x^T \mathbf{\Sigma}^{-1} \mu_k - \frac{1}{2} \mu_k^T \mathbf{\Sigma}^{-1} \mu_k + \log \pi_k$$

Here, x is an observation vector with p components, $\mathbf{\Sigma}$ is a common covariance matrix.

Quadratic Discriminant Analysis makes the same assumption as LDA about the Gaussian shape of f, but assumes that each class has its own covariance matrix. The discriminant function is then

$$\delta_k(x) = -\frac{1}{2}(x - \mu_k)^T \mathbf{\Sigma}_k^{-1}(x - \mu_k) - \frac{1}{2} \log |\mathbf{\Sigma}_k| + \log \pi_k$$

Naive Bayes takes a different approach than LDA and QDA in estimating $f_k(x)$. Rather than worrying about the covariances between the different predictors, we can assume that they are independent of one another. This means, then, that for f_{kj} being the density function of the jth predictor among observations in the kth class,

$$f_k(x) = f_{k1}(x_1) \times f_{k2}(x_2) \times \cdots \times f_{kp}(x_p)$$

This is a big convenience for modeling, since we don't need to worry about including marginal or joint distributions of the predictors. This assumption doesn't always hold, but is extremely convenient much of the time.

Now, the posterior probability is given by:

$$\Pr(Y = k | X = x) = p_k(x) = \frac{\pi_k \times f_{k1}(x_1) \times f_{k2}(x_2) \times \cdots \times f_{kp}(x_p)}{\sum_{\ell=1}^{K} \pi_\ell \times f_{\ell 1}(x_1) \times f_{\ell 2}(x_2) \times \cdots \times f_{\ell p}(x_p)}$$

The question now becomes, how do we model or estimate f_{kj}? We have a few options.

- If X is quantitative, we could simpply assume that each X_j is normally distributed, such that $(X_j|Y = k) \sim N(\mu_{jk}, \sigma^2_{jk})$. In other words, we would assume that within each class, the jth predictor is drawn from a univariate normal distribution with mean μ_{jk} and variance σ^2_{jk}.
- Another option for a quantitative X is to use a non-parametric method such as creating a histogram or using a kernel density estimator (essentially a smoothed histogram) as an estimate for $f_{kj}(x_j)$.
- For qualitative X_j, we can use a sample proportion according to each class to estimate f_{kj}. Let's say that $X_j \in \{1, 2, 3\}$, and we have 100 observations in the kth class. Suppose that the jth predictor takes on values of 1, 2, and 3 in 32, 55, and 13 of those observations, respectively. Then we have:

$$\hat{f}_{kj}(x_j) = \begin{cases} 0.32 & \text{if } x_j = 1 \\ 0.55 & \text{if } x_j = 2 \\ 0.13 & \text{if } x_j = 3 \end{cases}$$

To recap, with K classes and p predictors, we are estimating $K \times p$ density functions. In other words, for each class k, there will be p density estimates. Applying the observation X to each of these density estimates in a class k should yield a probability (posterior probability) that X belongs to that class. See Figure 4.10 and its caption for a good visual explanation.

With a low number of predictors, naive Bayes will not necessarily outperform LDA or QDA because the reduction in variance is not super important. In scenarios with many predictors or with few training examples, though, naive Bayes will outperform LDA or QDA because of the assumption of independence.

4.5: A Comparison of Classification Methods

Without writing out all the derivations, we can look at a mathematical comparison between the different classification settings. In a setting with K classes, we look to maximize

$$\log\left(\frac{\Pr(Y = k|X = x)}{\Pr(Y = K|X = x)}\right)$$

for $k = 1, \ldots, K$.

For the LDA setting, we end up with:

$$\begin{aligned} \log\left(\frac{\Pr(Y = k|X = x)}{\Pr(Y = K|X = x)}\right) &= \log\left(\frac{\pi_k f_k(x)}{\pi_K f_K(x)}\right) \\ &= \cdots \\ &= a_k + \sum_{j=1}^{p} b_{kj} x_j \end{aligned}$$

Where $a_k = \log\left(\frac{\pi_k}{\pi_K}\right) - \frac{1}{2}(\mu_k + \mu_K)^T \mathbf{\Sigma}^{-1}(\mu_k - \mu_K)$ and b_{kj} is the jth component of $\mathbf{\Sigma}^{-1}(\mu_k - \mu_K)$. So LDA assumes that the log odds of the posterior probabilities is linear in x, just like logistic regression.

In the QDA setting, we have:

$$\begin{aligned} \log\left(\frac{\Pr(Y = k|X = x)}{\Pr(Y = K|X = x)}\right) &= \log\left(\frac{\pi_k f_k(x)}{\pi_K f_K(x)}\right) \\ &= \cdots \\ &= a_k + \sum_{j=1}^{p} b_{kj} x_j + \sum_{j=1}^{p} \sum_{l=1}^{p} c_{kjl} x_j x_l \end{aligned}$$

where a_k, b_kj, and c_kjl are functions of π_k, π_K, μ_k, μ_K, $\mathbf{\Sigma}_k$, and $\mathbf{\Sigma}_K$

Finally, in the naive Bayes setting, we have:

$$\begin{aligned}\log\left(\frac{\Pr(Y=k|X=x)}{\Pr(Y=K|X=x)}\right) &= \log\left(\frac{\pi_k f_k(x)}{\pi_K f_K(x)}\right)\\ &= \cdots\\ &= a_k + \sum_{j=1}^{p} g_{kj}(x_j)\end{aligned}$$

where $a_k = \log\left(\frac{\pi_k}{\pi_K}\right)$ and $g_{kj}(x_j) = \log\left(\frac{f_{kj}(x_j)}{f_{Kj}(x_j)}\right)$. This takes the form of a *generalized additive model*, which is covered in chapter 7.

We can notice a few things about these results:

- LDA is a special case of QDA with $c_{kjl} = 0$ for all k, j, l.
- Any classifier with a linear decision boundary is a special case of naive Bayes with $g_{kj}(x_j) = b_{kj}x_j$. This also means that LDA is a specific case of naive Bayes.
  - This is not directly intuitive, as for LDA we assumed a shared within-class covariance matrix, and for naive Bayes we assumed independence across all features.
- Modeling f with naive Bayes using a one-dimensional Gaussian distribution gives us the LDA classifier where $\mathbf{\Sigma}$ is a diagonal matrix with the jth diagonal element is equal to σ_j^2.
- QDA and naive Bayes are completely separate from one another, but LDA is a special case of both. Because QDA has interaction terms, it has the potential to be more accurate in scenarios with high interactions between classes.

There is no one classifier that can do it all. In six scenarios, six different classifiers were tested on each scenario. In addition to Logistic Regression, LDA, QDA, and naive Bayes, K-nearest neighbors with $K = 1$ and with K chosen automatically from a cross-validation set were compared. When the decision boundary is highly non-linear and we have many observations, the nonparametric approaches such as KNN tend to work much better than parametric models.

4.6: Generalized Linear Models

For this section, consider the `Bikeshare` data, which describes the number of hourly users of a bike sharing program in DC. The response is `bikers`, which is a *count* variable, and the covariates are `mnth`, `hr`, `workingday`, `temp`, and `weathersit`.

First, let's explore multiple linear regression on the data. We fit the model:

$$Y = \sum_{j=1}^{p} X_j\beta_j + \epsilon$$

Performing multiple linear regression on the `Bikeshare` data is problematic for two reasons:

1. The `bikers` variable is fundamentally integer-valued, but a linear model assumes a continuous, normally distributed error term ϵ.
2. With just multiple linear regression, it turns out that 9.6% of the fitted values are negative, which doesn't make sense at all.

To remedy the second issue, we could perhaps fit a model to the log of bikers, which would yield all positive predictions:

$$\log(Y) = \sum_{j=1}^{p} X_j \beta_j + \epsilon$$

This, however, leads to difficulty in interpretation, namely that we would say that *a one-unit increase in X_j is associated with an increase in the mean of the log of Y by an amount β_j*. Also, this technique cannot be applied where 0 is in the range of Y.

To combat these issues, we can use **Poisson regression**.

If a random variable Y follows a Poisson distribution, it has a probability mass function given by:

$$\Pr(Y = k) = \frac{e^{-\lambda}\lambda^k}{k!} \quad \text{for } k = 0, 1, 2, ...$$

With a Poisson distribution, we know that the expected value *and* variance are both equal to λ.

The Poisson distribution is used to model *counts*. Let's consider:

- A particular hour of the day
- A particular set of weather conditions
- During a particular month of the year

For these specific parameters, we might have a Poisson distribution with $E(Y) = \lambda = 5$. So, for example, we would have $P(Y = 2) = \frac{e^{-5}5^2}{2!} = 0.084$. We expect this value of λ to vary according to each of our covariates (or predictors), so we write λ as a function of each of the predictors:

$$\lambda = \lambda(X_1, X_2, \ldots, X_p)$$

Because λ can only take on values greater than zero, we model the logarithm of lambda as being a linear function of the covariates:

$$\log(\lambda(X_1, ..., X_p)) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Or equivalently,

$$\lambda(X_1, ..., X_p) = e^{\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p}$$

Now that we have a form for λ as a function of the covariates, we can write out our likelihood function:

$$L(\beta_0, \beta_1, \ldots, \beta_p) = \prod_{i=1}^{n} \frac{e^{-\lambda(x_i)}\lambda(x_i)^{y_i}}{y_i!}$$

And we can use this along with maximum likelihood estimation to find the values of our coefficients $\beta_0, \beta_1, \ldots, \beta_p$ that fit our model.

In order to interpret the Poisson model, we should notice that increasing X_j by one unit changes $\lambda = E(Y)$ by a factor of $\exp(\beta_j)$. With regards to the model itself, we also assume that the mean bike usage in an hour equals the variance of bike usage during that same hour. This relationship holds up nicely. Additionally, there are no negative fitted values since the Poisson distribution only holds for values ≥ 0.

Linear, logistic, and Poisson regression all share a few common characteristics:

1. Each approach uses predictors $X_1, \ldots, X_p$ to predict a response Y. We assume that Y belongs to a family of distributions. For linear, we assume Gaussian; for logistic, we assume Bernoulli; for Poisson, we assume a Poisson distribution.
2. Each approach models the mean of Y as a function of the predictors.
   For Linear Regression, we have:

$$E(Y|X_1, \ldots, X_p) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p.$$

   For Logistic Regression, we have:

$$\begin{aligned} E(Y|X_1, \ldots, X_p) &= \Pr(Y = 1|X_1, \ldots, X_p) \\ &= \frac{e^{\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p}}{1 + e^{\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p}}. \end{aligned}$$

   For Poisson Regression, we have:

$$E(Y|X_1, \ldots, X_p) = e^{\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p}.$$

The previous three equations can be expressed in terms of a *link function*, given by η. That is, that

$$\eta(E(Y|X_1, \ldots, X_p)) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

The link functions for these distributions are:

$$\begin{aligned} \text{Linear:} \quad & \eta(\mu) = \mu \\ \text{Logistic:} \quad & \eta(\mu) = \log\left(\frac{\mu}{1-\mu}\right) \\ \text{Poisson:} \quad & \eta(\mu) = \log(\mu) \end{aligned}$$

These three distributions (Gaussian, Bernoulli, and Poisson) all come from the *exponential family* of distributions, of which the Exponential, Gamma, and Negative Binomial Distributions are also members.

We create a regression problem by modeling Y as coming from a member of the exponential family, then transforming the mean of the response so that it's a linear function of the predictors, then we bunch those transformations up into the link function η. A regression approach that follows this idea is called a **generalized linear model (GLM)**.