LECTURE 10: BASICS OF LOGISTIC REGRESSION

Text reference: Shalizi Chapter 11

Overview

So far we have assumed that the outcome Y is real-valued. **Logistic regression** is a generalization of regression that is used when the outcome Y is *binary* (which is most common, in practice, for categorical data) or *binomial*. Suppose, for example, that $Y_i \in \{0, 1\}$ is a binary variable, and we want to relate Y to some covariate x. Examples:

- Y : whether a patient will develop breast cancer or remain healthy;
  X: genetic information
- Y: whether or not a user will like a new product;
  X: user covariates and a history of the user's previous ratings

The usual regression model for relating Y to X is not appropriate since it does not constrain Y to be binary and $0 \leq \mathbb{E}[Y|X] \leq 1$.

In logistic regression, we assume that

$$Y_i|X_i \sim \text{Binomial}(n_i, p_i)$$

(where for binary data $n_i = 1$). Instead of modeling Y_i as a function of X_i directly, we model the **probability** p_i that Y_i is equal to class 1, given X_i.

Remark: Note that if were were only trying to predict Y_i (that is, come up

with a rule (or a function $h : \mathcal{X} \to \mathcal{Y} = \{0, 1\}$) which guesses the binary output from the input variables) then that would be a **classification** problem. To cite Shalizi: *"However, guessing yes or no is pretty crude especially if there is no perfect rule (Why should there be a perfect rule?). Something which takes noise into account, and doesn't just give a binary answer, will often be useful. In short, we want probabilities which means we need to fit a stochastic model."* Of course, once we have an estimate $\widehat{p}(x)$ of $p(x) = \mathbb{P}(Y = 1|X = x)$, we can predict the associated class according to

$$\widehat{h}(x) = \begin{cases} 1 & \widehat{p}(x) > 0.5 \\ 0 & \widehat{p}(x) \leq 0.5 \end{cases}$$

Follow-up questions:

- What is the assumed model for the conditional probability $p(x)$?
- How do you fit such models to data?
- How do you carry out inference?

R Demo 10.1: Motivating Example

Our first example concerns the probability of extinction as a function of island size (Sleuth case study 21.1). The data provide island size, number of bird species present in 1949 and the number of these extinct by 1959.

island	area	atrisk	extinctions
Ulkokrunni	185.80	75	5
Maakrunni	105.80	67	3
Ristikari	30.70	66	10
Isonkivenletto	8.50	51	6
Hietakraasukka	4.80	28	3
Kraasukka	4.50	20	4
Lansiletto	4.30	43	8
Pihlajakari	3.60	31	3
Tyni	2.60	28	5
Tasasenletto	1.70	32	6
Raiska	1.20	30	8
Pohjanletto	0.70	20	2
Toro	0.70	31	9
Luusiletto	0.60	16	5
Vatunginletto	0.40	15	7
Vatunginnokka	0.30	33	8
Tiirakari	0.20	40	13
Ristikarenletto	0.07	6	3

Let n_i be the number of species at risk and Y_i be the number of extinctions out of n_i. We assume $Y_i|x_i \sim \text{Binomial}(n_i, p_i)$ where p_i is a function of the area. Define $x_i = \log(\text{area}_i)$, If we plot $\widehat{p}_i = Y_i/n_i$ as a function of x_i we see

an s-shaped decline in the response variable, but $\log[\widehat{p}_i/(1-\widehat{p}_i)]$ declines linearly with x_i. This example motivates logistic regression.

The Logistic Model

Throughout this lecture, we will for simplicity work with binary data ($n_i = 1$), which is most common in practice. Later on, we will revisit the binomial with $n_i > 1$. Suppose that $Y_i \in \{0, 1\}$ and we want to relate Y to some covariate x.

With the **logistic regression** model we assume that the regression

Note that since Y_i is binary, $\mathbb{E}(Y_i|X_i) = \mathbb{P}(Y_i = 1|X_i)$. We assume this probability follows the logistic function $e^{\beta_0+\beta_1 x}/(1+e^{\beta_0+\beta_1 x})$. The parameter β_1 controls the steepness of the curve. The parameter β_0 controls the horizontal shift of the curve (see Fig 1).

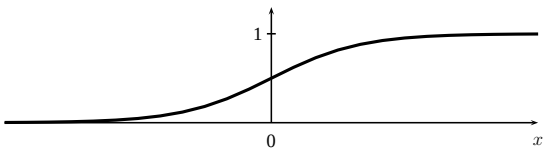

Figure 1: The logistic function $p = e^x/(1 + e^x)$.

Define the **logit function**

$$\text{logit}(z) = \log\left(\frac{z}{1-z}\right).$$

Also, define

$$p_i = \mathbb{P}(Y_i = 1|X_i).$$

Then we can rewrite the logistic model as

[1]

The extension to several covariates is straightforward:

The **logit** is the **log odds function**.

Interpreting the Coefficients

- How can we interpret the role of the coefficients $\beta = (\beta_1, \ldots \beta_p) \in \mathbb{R}^p$ in the logistic model? One nice feature of the logistic model is that it comes equipped with a useful interpretation for these coefficients.

- Write
$$\frac{p(X)}{1-p(X)} = e^{\beta^T X} = e^{\beta_1 X_1 + \ldots + \beta_p X_p}.$$
The left-hand side above is the odds of class 1 (conditional on X). We can see that **increasing X_j by one unit, while keeping all other predictors fixed, multiplies the odds by e^{β_j}.** This is because
$$e^{\beta_1 X_1 + \ldots + \beta_j (X_j+1) + \ldots + \beta_p X_p} = e^{\beta_1 X_1 + \ldots + \beta_j X_j + \ldots + \beta_p X_p} \cdot e^{\beta_j}$$

- Equivalently, write
$$\log\Big(\frac{p(X)}{1-p(X)}\Big) = \beta^T X = \beta_1 X_1 + \ldots + \beta_p X_p.$$
Now **increasing X_j by one unit, and keeping all other predictors fixed, changes the log odds by β_j.**

- It will help to get comfortable with the concept of odds, and log odds, if you haven't done so already in another class. Note that probabilities q close to 0 or 1 have odds $q/(1-q)$ close to 0 or ∞, respectively. And probabilities q close to 0 or 1 have log odds $\log(q/(1-q))$ close to $-\infty$ or ∞, respectively.

Estimation of Parameters by MLE

How do we estimate the parameters of the logistic regression function? Usually we use maximum likelihood.

As before: Given i.i.d. data $(x_i, Y_i) \in \mathbb{R}^p \times \{0, 1\}$, $i = 1, \ldots n$, we let $p_i \equiv p(x_i) \equiv \mathbb{P}(Y_i = 1|x_i)$, and assume

$$\log\left(\frac{p(x_i)}{1 - p(x_i)}\right) = \beta^T x_i, \quad i = 1, \ldots n.$$

To construct an estimate $\widehat{\beta}$ of the coefficients, we will use the principle of *maximum likelihood*.

The probability function for n independent tosses, $Y_1, \ldots, Y_n$, is

The **likelihood function** is just the probability function regarded as a function of the parameter β and treating the data as fixed:

The **maximum likelihood estimator** is the value $\widehat{\beta}$ that maximizes $\mathcal{L}(\beta)$. Maximizing the likelihood is equivalent to maximizing the loglikelihood function

It helps to rearrange this as

$$\begin{aligned}\ell(\beta) &= \sum_{i=1}^{n} Y_i \big[\log p(x_i) - \log\big(1 - p(x_i)\big)\big] + \log\big(1 - p(x_i)\big) \\ &= \sum_{i=1}^{n} Y_i \log\Big(\frac{p(x_i)}{1 - p(x_i)}\Big) + \log\big(1 - p(x_i)\big).\end{aligned}$$

Finally, plugging in for $\log(p(x_i)/(1 - p(x_i))) = x_i^T \beta$ and using $1 - p(x_i) = 1/(1 + \exp(x_i^T \beta))$, $i = 1, \ldots n$,

$$\ell(\beta) = \sum_{i=1}^{n} Y_i (x_i^T \beta) - \log\big(1 + \exp(x_i^T \beta)\big). \tag{1}$$

You can see that, unlike the least squares criterion for regression, this criterion $\ell(\beta)$ does not have a closed-form expression for its maximizer (e.g., try taking its partial derivatives and setting them equal to zero). Hence we have to run an optimization algorithm to find $\widehat{\beta}$

Somewhat remarkably, we can maximize $\ell(\beta)$ by running repeated weighted least squares regressions! For those of you who have learned a little bit of optimization, this is actually just an instantiation of Newton's method. Applied to the maximum likelihood criterion, we refer to it as **iterative reweighted least squares** or IRLS. [For details, see HTF Section 4.4.1, "Fitting Logistic Regression Models"]

In summary: Estimation of $\widehat{\beta}$ in logistic regression is more involved than it is in linear regression, but it is possible to do so by iteratively using linear

regression software (by a fast numerical algorithm called reweighted least squares) .

Inference

Much of the standard machinery for inference in linear regression carries over to logistic regression. Recall that we can solve for the logistic regression coefficients $\widehat{\beta}$ by performing repeated weighted linear regressions; hence we can simply think of the logistic regression estimates $\widehat{\beta}$ as the result of a single weighted linear regression—the last one in this sequence (upon convergence). *Confidence intervals for $\widehat{\beta}_j$, $j = 1, \ldots p$, and so forth, are then all obtained from this weighted linear regression perspective.* For example, the standard error of the coefficients are given by

---

---

where W is the $n \times n$ diagonal weight matrix whose i^{th} diagonal element is $\widehat{p}_i(1 - \widehat{p}_i)$, the estimated variance of Y_i given X_i. [1] We will not go into detail here, but such inferential tools for weighted linear regression are implemented in software, and it helps to be aware of where they come from.

[1] Notice that the standard error of Y_i is a function of the mean p_i. This is a key difference between linear and logistic regression. In the latter there is no unknown σ^2. The variance of the model is determined by the mean.

Model Selection and Deviance Tests

When fitting logistic models, one has to address the following model selection problem: which $\beta_j x_j$ terms in the model for $\text{logit}(p)$ should we include? This is essentially the same as the model selection problem in linear regression. One approach is to use **AIC** (which can be thought of as a "goodness of fit" minus "complexity" score). Let M denote some logistic model. Different models correspond to setting different $\beta_j x_j$ terms to 0. For AIC, we choose the model M which maximizes ("goodness of fit" minus "complexity")

---

or, alternatively, minimizes ("lack of fit" plus "complexity")

---

where $|M|$ is the number of parameters in model M, and $\widehat{\ell}(M)$ is the value of the log-likelihood of that model evaluated at the MLE. Usually the model search is done with **forward or backward stepwise regression**. In forward stepwise regression, we start with no covariates in the model. We then add the one variable that leads to the best score. We continue adding variables one at a time until the score does not improve. Backwards stepwise regression is the same except that we start with the biggest model and drop one variable at a time.

A different approach is based on **hypothesis testing** where $M_{red} \subset M_{full}$ and we test the hypothesis

$$H_0 : \text{the true model is } M_{red} \text{ vs. } H_1 : \text{the true model is } M_{full}$$

Recall from Lecture 9: For linear models if we wish to compare the fit of a "full" model with a "reduced" model we examine the difference in residual sum of squares via an F test:

$$F = \frac{(RSS_{red} - RSS_{full})/df_1}{RSS_{full}/df_2} \sim F_{df_1, df_2},$$

where $df_1 = p_{full} - p_{red}$ and $df_2 = n - (p_{full} + 1)$. Now, assuming normality, $RSS_{full} \sim \sigma^2 \chi^2_{df_2}$, so $E[MSE = RSS_{full}/df_2] = \sigma^2$, and $(RSS_{red} - RSS_{full}) \sim \sigma^2 \chi^2_{df_1}$. Consequently, σ^2 cancels out of the equation in the F test. The test assesses whether the difference in RSS in the full and reduced models is bigger than expected with degrees of freedom equal to df_1. The denominator of the F test is included simply as an estimate σ^2.

For glm models a **deviance test** plays the same role for comparing the fit of a "full" model with a "reduced" model. If the log likelihood is $\ell(\theta) = \sum_i \log f(y_i|\theta)$, then the deviance is defined as

---

For a linear model the deviance reduces to

For Poisson and Binomial model the mean determines the variance of the model, so there is no unknown σ^2 to be estimated.

To test a full vs. reduced model for glm we look at the drop in deviance

rather than an F test; this is basically the **likelihood ratio test** for the null hypothesis H_0 : the model is M_{red} vs. H_1 : the model is M_{full}. Under the null hypothesis, the difference in deviances is asymptotically distributed $\chi^2_{df_1}$. If $df_1 = 1$ then an alternative test is Wald's test $\widehat{\beta}_j / se(\beta_j)$. The Wald's test is the natural analog to the t-test in linear models. The deviance test and Wald's test will give similar, but not identical results.

R Demo 10.2

The Coronary Risk-Factor Study (CORIS) data involve 462 males between the ages of 15 and 64 from three rural areas in South Africa. The outcome Y is the presence ($Y = 1$) or absence ($Y = 0$) of coronary heart disease. There are 9 covariates: systolic blood pressure, cumulative tobacco (kg), ldl (low density lipoprotein cholesterol), adiposity, famhist (family history of heart disease), typea (type-A behavior), obesity, alcohol (current alcohol consumption), and age. A logistic regression yields the following estimates and Wald statistics W_j for the coefficients:

Covariate	$\widehat{\beta}_j$	$\widehat{\text{se}}$	W_j	p-value
Intercept	-6.145	1.300	-4.738	0.000
sbp	0.007	0.006	1.138	0.255
tobacco	0.079	0.027	2.991	0.003
ldl	0.174	0.059	2.925	0.003
adiposity	0.019	0.029	0.637	0.524
famhist	0.925	0.227	4.078	0.000
typea	0.040	0.012	3.233	0.001
obesity	-0.063	0.044	-1.427	0.153
alcohol	0.000	0.004	0.027	0.979
age	0.045	0.012	3.754	0.000

Are you surprised by the fact that systolic blood pressure is not significant or by the minus sign for the obesity coefficient? If yes, then you are confusing association and causation. The fact that blood pressure is not significant does not mean that blood pressure is not an important *cause* of heart disease. It means that it is not an important *predictor* of heart disease relative to the other variables in the model. ■

Model selection can be done using AIC or BIC:

$$AIC_S = -2\ell(\widehat{\beta}_S) + 2|S|$$

where S is a subset of the covariates.

When $n_i = 1$ it is not possible to examine residuals to evaluate the fit of our regression model.

```
> attach(sa.data)
> out     = glm(chd ~ ., family=binomial,data=sa.data)
> print(summary(out))



Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -6.1482113  1.2977108  -4.738 2.16e-06 ***
sbp          0.0065039  0.0057129   1.138 0.254928
tobacco      0.0793674  0.0265321   2.991 0.002777 **
ldl          0.1738948  0.0594451   2.925 0.003441 **
adiposity    0.0185806  0.0291616   0.637 0.524020
famhist      0.9252043  0.2268939   4.078 4.55e-05 ***
typea        0.0395805  0.0122417   3.233 0.001224 **
obesity     -0.0629112  0.0440721  -1.427 0.153447
alcohol      0.0001196  0.0044703   0.027 0.978655
age          0.0452028  0.0120398   3.754 0.000174 ***
---



> out2 = step(out)

Start:  AIC= 492.14
 chd ~ sbp + tobacco + ldl + adiposity + famhist + typea + obesity +
    alcohol + age
```

```
etc.


Step:  AIC= 487.69
 chd ~ tobacco + ldl + famhist + typea + age

          Df Deviance    AIC
<none>         475.69 487.69
- ldl      1   484.71 494.71
- typea    1   485.44 495.44
- tobacco  1   486.03 496.03
- famhist  1   492.09 502.09
- age      1   502.38 512.38
>

> p = out2$fitted.values
> names(p) = NULL
> n = nrow(sa.data)
> predict  = rep(0,n)
> predict[p > .5] = 1
> print(table(chd,predict))

   predict
chd 0   1
  0 256  46
  1  73  87
> error = sum( ((chd==1)&(predict==0)) | ((chd==0)&(predict==1)) )/n
> print(error)

[1] 0.2575758
```