Logistic Regression

Li Xing

June 13, 2023

Categorical Variables

- **Nominal Data** (no intrinsic order)

  gender, hair color, name, etc

- **Ordinal Data**

  -*Age group*: baby, child, teenager, adult, middle-aged person, & senior;

  -*Weight*: heavy, average weight, & thin/slim;

  -*Level of Depression*: mild, moderate, & severe.

Classification

It is the process of predicting categorical responses.

Examples of simple classifiers: **logistic regression**, **linear discriminant analysis (LDA)**; **K-nearest neighbors**, etc.

Example: Default Data

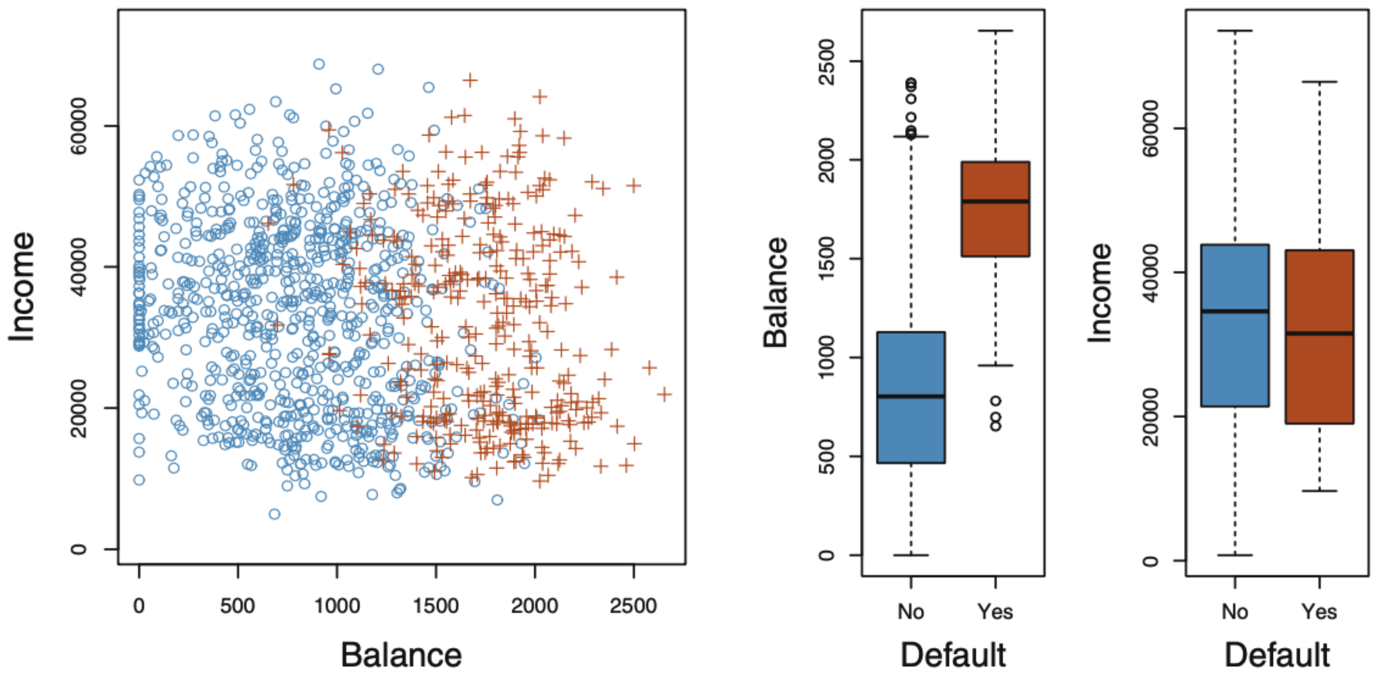


FIGURE 4.1. *The* `Default` *data set.* Left: *The annual incomes and monthly credit card balances of a number of individuals. The individuals who defaulted on their credit card payments are shown in orange, and those who did not are shown in blue.* Center: *Boxplots of* `balance` *as a function of* `default` *status.* Right: *Boxplots of* `income` *as a function of* `default` *status.*

About the Link Function

A generalized linear model format

$$E(Y|X) = g^{-1}(\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p)$$

- g links a linear combination of predictors with the expectation of the response.
- In a linear regression, it is the identity function.

About the Link Function for a Binary Outcome

When Y is binary, we will model

$$E(Y|X) = Pr(Y = 1|X) = p.$$

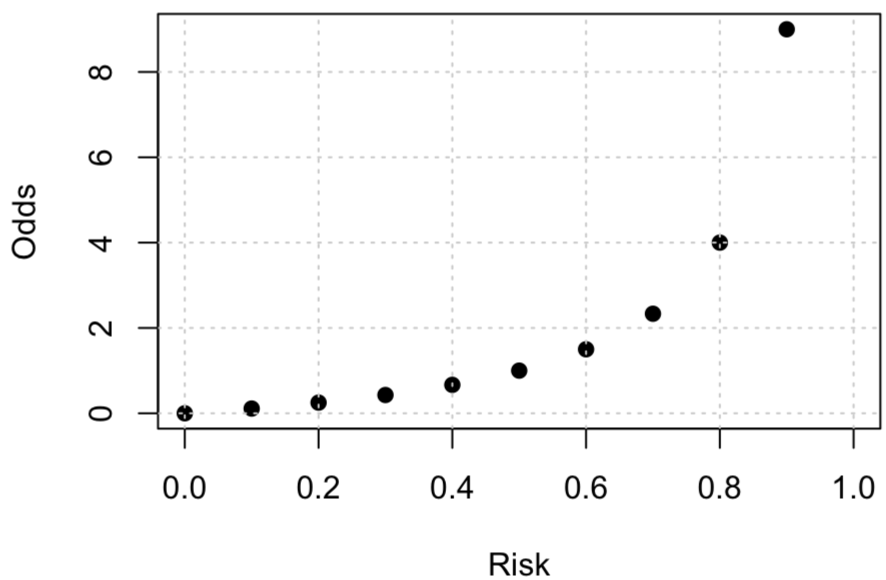
Now the link function need to link a value in $(0, 1)$ with a value in $(-\infty, \infty)$.

Logistic regression model

- the link function is the logit function.
- We model $\log\left(\frac{p}{1-p}\right)$ as a linear combination of the predictors.

Terms I

- **Risk** p: probability of $Y = 1$ (Disease, Death, Default, etc)
- **Odds** $\frac{p}{1-p}$: An alternative for risk measurement.



Terms II

- **Relative Risk** $\frac{p_1}{p_2}$: Ratio of two risks used to comparing risks of two groups.
- **Odds Ratio** $\frac{\frac{p_1}{1-p_1}}{\frac{p_2}{1-p_2}}$: The alternative for comparing risks of two groups.
- For a rare disease, Odds Ratio can be approximated by Relative Risk.

Comparison Between a Linear Regression Model and a Logistic Regression Model

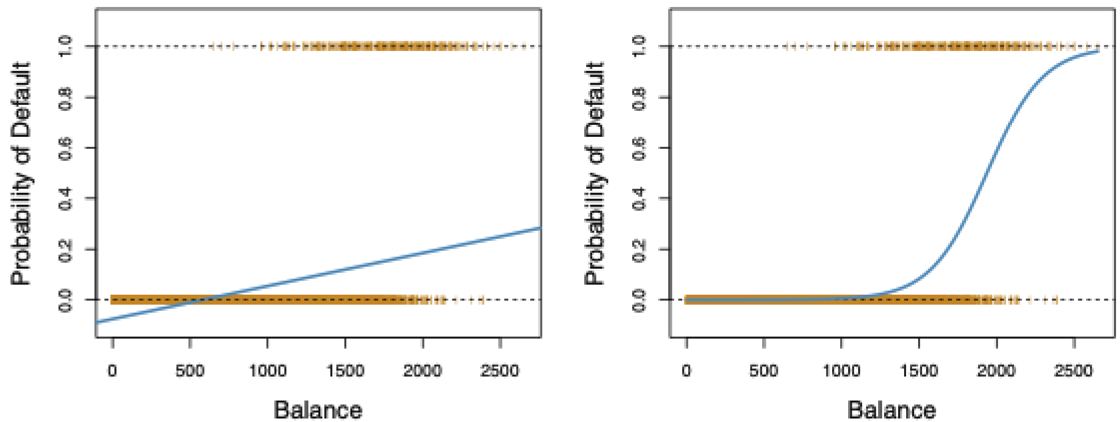


FIGURE 4.2. *Classification using the* `Default` *data.* Left: *Estimated probability of* `default` *using linear regression. Some estimated probabilities are negative! The orange ticks indicate the 0/1 values coded for* `default`*(*`No` *or* `Yes`*).* Right: *Predicted probabilities of* `default` *using logistic regression. All probabilities lie between* 0 *and* 1.

Example: The Stock Market Data

```r
library(ISLR)
names(Smarket)
```

```
## [1] "Year"      "Lag1"      "Lag2"      "Lag3"      "Lag4"      "Lag5"
## [7] "Volume"    "Today"     "Direction"
```

```r
dim(Smarket)
```

```
## [1] 1250    9
```

Viewing Data

```
head(Smarket)
```

```
##   Year   Lag1   Lag2   Lag3   Lag4   Lag5 Volume  Today Direction
## 1 2001  0.381 -0.192 -2.624 -1.055  5.010 1.1913  0.959        Up
## 2 2001  0.959  0.381 -0.192 -2.624 -1.055 1.2965  1.032        Up
## 3 2001  1.032  0.959  0.381 -0.192 -2.624 1.4112 -0.623      Down
## 4 2001 -0.623  1.032  0.959  0.381 -0.192 1.2760  0.614        Up
## 5 2001  0.614 -0.623  1.032  0.959  0.381 1.2057  0.213        Up
## 6 2001  0.213  0.614 -0.623  1.032  0.959 1.3491  1.392        Up
```

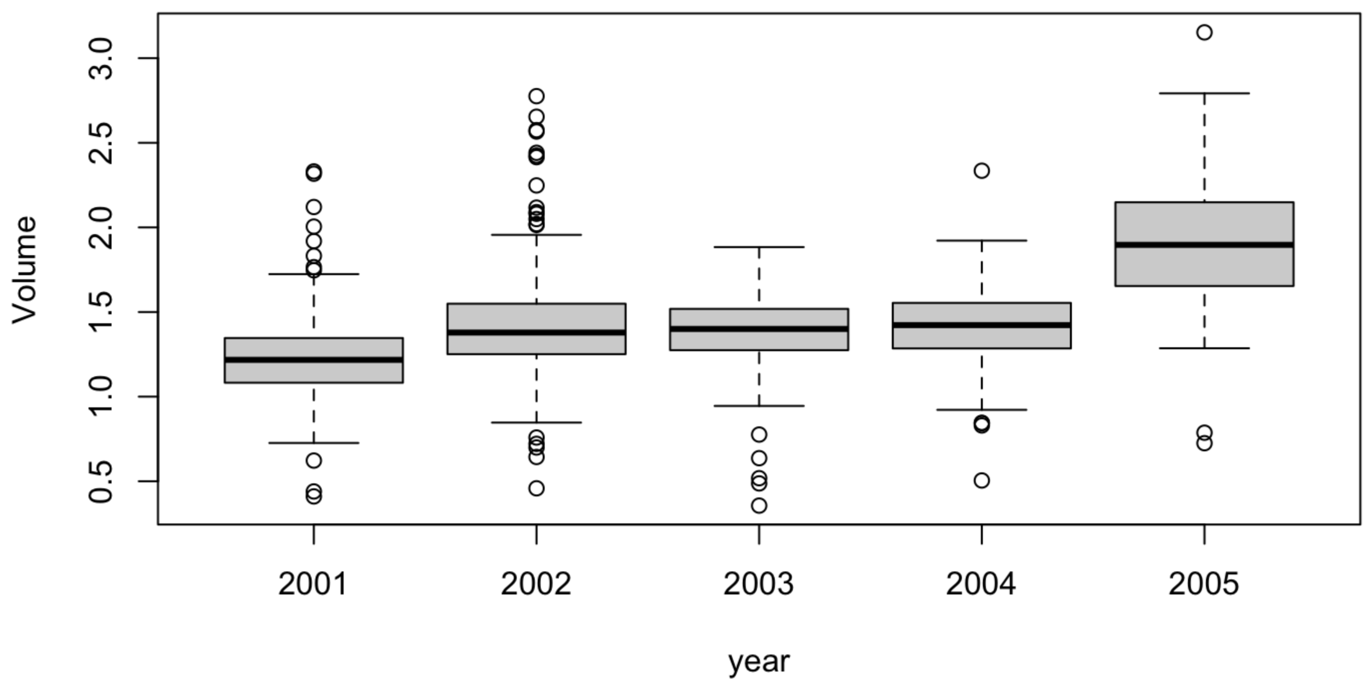
EDA

```r
cor(Smarket[, -9])
```

```
##              Year         Lag1         Lag2         Lag3         Lag4
## Year   1.00000000  0.029699649  0.030596422  0.033194581  0.035688718
## Lag1   0.02969965  1.000000000 -0.026294328 -0.010803402 -0.002985911
## Lag2   0.03059642 -0.026294328  1.000000000 -0.025896670 -0.010853533
## Lag3   0.03319458 -0.010803402 -0.025896670  1.000000000 -0.024051036
## Lag4   0.03568872 -0.002985911 -0.010853533 -0.024051036  1.000000000
## Lag5   0.02978799 -0.005674606 -0.003557949 -0.018808338 -0.027083641
## Volume 0.53900647  0.040909908 -0.043383215 -0.041823686 -0.048414246
## Today  0.03009523 -0.026155045 -0.010250033 -0.002447647 -0.006899527
##                Lag5      Volume        Today
## Year    0.029787995  0.53900647  0.030095229
## Lag1   -0.005674606  0.04090991 -0.026155045
## Lag2   -0.003557949 -0.04338321 -0.010250033
## Lag3   -0.018808338 -0.04182369 -0.002447647
## Lag4   -0.027083641 -0.04841425 -0.006899527
## Lag5    1.000000000 -0.02200231 -0.034860083
## Volume -0.022002315  1.00000000  0.014591823
## Today  -0.034860083  0.01459182  1.000000000
```

EDA (continued)

```
boxplot(Smarket$Volume~Smarket$Year, xlab = "year", ylab = "Volume")
```



Fitting the Logistic Regression Model

We would like to fit a logistic regression model in order to predict **Direction** using **Lag1** through **Lag5** and **Volume**.

```
myfit <- glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume, data = Smarket,
             family = "binomial")
```

The Fitted Model

```
summary(myfit)
```

```
##
## Call:
## glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 +
##     Volume, family = "binomial", data = Smarket)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.446  -1.203   1.065   1.145   1.326
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.126000   0.240736  -0.523    0.601
## Lag1        -0.073074   0.050167  -1.457    0.145
## Lag2        -0.042301   0.050086  -0.845    0.398
## Lag3         0.011085   0.049939   0.222    0.824
## Lag4         0.009359   0.049974   0.187    0.851
## Lag5         0.010313   0.049511   0.208    0.835
## Volume       0.135441   0.158360   0.855    0.392
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

The Estimated Coefficients

```
coef(myfit)
```

```
##  (Intercept)         Lag1         Lag2         Lag3         Lag4         Lag5 
## -0.126000257 -0.073073746 -0.042301344  0.011085108  0.009358938  0.010313068 
##       Volume 
##  0.135440659
```

The Fitted Probabilities

```
glm.probs = predict(myfit, type="response")
glm.probs[1:10]
```

```
##         1         2         3         4         5         6         7         8
## 0.5070841 0.4814679 0.4811388 0.5152224 0.5107812 0.5069565 0.4926509 0.5092292
##         9        10
## 0.5176135 0.4888378
```

Classifications

```
contrasts(Smarket$Direction)
```

```
##      Up
## Down  0
## Up    1
```

```
glm.pred = rep("Down", 1250)
glm.pred[glm.probs > 0.5] = "Up"
table(glm.pred, Smarket$Direction)
```

```
##
## glm.pred Down  Up
##     Down  145 141
##     Up    457 507
```

Two Rates

```r
(507+145)/1250   # true classification rate
```

```
## [1] 0.5216
```

```r
(141+457)/1250   # mis-classification rate
```

```
## [1] 0.4784
```

Or alternatively

```r
mean(glm.pred==Smarket$Direction)
mean(glm.pred!=Smarket$Direction)
```