Lab 2: Fitting linear anf generalized linear models

Peter Diggle & Emanuele Giorgi

Lancaster Medical School, Lancaster University, Lancaster, UK

Model-based geostatistics: geospatial statistical methods for public health applications, 5-9 October 2015

Overview

- Time: 09:00-10:30.

- Contents:
  1. formula expressions and contrasts;
  2. linear models fitting, extraction of information and testing of regression effects;
  3. fitting of binomial models, from log-odds to odds.

Formula

Linear regression:

$$Y_i = \overbrace{\sum_{j=1}^{p} \underbrace{\beta_j}_{\text{regression coefficient}} \underbrace{x_{ij}}_{\text{covariate}}}^{\text{linear predictor}} + \overbrace{Z_i}^{\text{error term}}$$

$$Z_i \sim N(0, \sigma^2).$$

Formula

Linear regression:

$$Y_i = \overbrace{\sum_{j=1}^{p} \underbrace{\beta_j}_{\text{regression coefficient}} \underbrace{x_{ij}}_{\text{covariate}}}^{\text{linear predictor}} + \overbrace{Z_i}^{\text{error term}}$$

$$Z_i \sim N(0, \sigma^2).$$

Define a linear predictor through a `formula` obbject.

```
y~1 # Intercept only
y~x # Linear effect of x
y~x-1 # Removal of the intercept
y~x+I(x^2) # Quadratic effect of x
```

Contrasts: unordered factors

```
> state <- c(rep("Malawi",5),rep("Italy",2),rep("Madagascar",4))
> state <- factor(state, levels=c("Italy", "Madagascar", "Malawi"))
> contrasts(state)
           Madagascar Malawi
Italy               0      0
Madagascar          1      0
Malawi              0      1
```

Contrasts: unordered factors

```
> state <- c(rep("Malawi",5),rep("Italy",2),rep("Madagascar",4))
> state <- factor(state, levels=c("Italy", "Madagascar", "Malawi"))
> contrasts(state)
           Madagascar Malawi
Italy               0      0
Madagascar          1      0
Malawi              0      1
```

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + Z_i$$

with

$$x_{i1} = \begin{cases} 1 & \text{if Madagascar} \\ 0 & \text{otherwise} \end{cases}, \; x_{i2} = \begin{cases} 1 & \text{if Malawi} \\ 0 & \text{otherwise} \end{cases}$$

Contrasts: unordered factors

```
> state <- c(rep("Malawi",5),rep("Italy",2),rep("Madagascar",4))
> state <- factor(state, levels=c("Italy", "Madagascar", "Malawi"))
> contrasts(state)
           Madagascar Malawi
Italy               0      0
Madagascar          1      0
Malawi              0      1
```

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + Z_i$$

with

$$x_{i1} = \begin{cases} 1 & \text{if Madagascar} \\ 0 & \text{otherwise} \end{cases}, \quad x_{i2} = \begin{cases} 1 & \text{if Malawi} \\ 0 & \text{otherwise} \end{cases}$$

- Italy: $E(Y_i) = \beta_0$.
- Madagascar: $E(Y_i) = \beta_0 + \beta_1$.
- Malawi: $E(Y_i) = \beta_0 + \beta_2$.

Contrasts: ordered factors

```
> income <- c(rep(1,3),rep(2,5),rep(3,4))
> income <- factor(income,levels=1:3)
> contrasts(income)[lower.tri(contrasts(income))] <- 1
> contrasts(income)
  2 3
1 0 0
2 1 0
3 1 1
```

Contrasts: ordered factors

```
> income <- c(rep(1,3),rep(2,5),rep(3,4))
> income <- factor(income,levels=1:3)
> contrasts(income)[lower.tri(contrasts(income))] <- 1
> contrasts(income)
  2 3
1 0 0
2 1 0
3 1 1
```

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + Z_i$$

with

$$x_{i1} = \begin{cases} 1 & \text{if ``Medium Income'' or ``High income''} \\ 0 & \text{otherwise} \end{cases}, \; x_{i2} = \begin{cases} 1 & \text{if ``High income''} \\ 0 & \text{otherwise} \end{cases}$$

Contrasts: ordered factors

```
> income <- c(rep(1,3),rep(2,5),rep(3,4))
> income <- factor(income,levels=1:3)
> contrasts(income)[lower.tri(contrasts(income))] <- 1
> contrasts(income)
  2 3
1 0 0
2 1 0
3 1 1
```

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + Z_i$$

with

$$x_{i1} = \begin{cases} 1 & \text{if ``Medium Income'' or ``High income''} \\ 0 & \text{otherwise} \end{cases}, \quad x_{i2} = \begin{cases} 1 & \text{if ``High income''} \\ 0 & \text{otherwise} \end{cases}$$

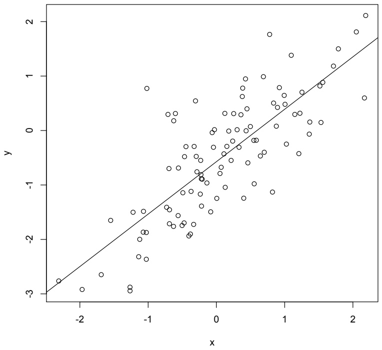
- Low income: $E(Y_i) = \beta_0$.
- Medium income: $E(Y_i) = \beta_0 + \beta_1$.
- High income: $E(Y_i) = \beta_0 + \beta_1 + \beta_2$.

Linear regression: a simulated example (1)

```
> beta0 <- -0.5
> beta1 <- 1
>
> sigma2 <- 0.5
>
> n <- 100
>
> set.seed(123)
> x <- rnorm(n)
> y <- beta0+beta1*x+rnorm(n,sd=sqrt(sigma2))
> data.sim <- data.frame(y=y,x=x)
>
> fit.lm <- lm(y~x,data=data.sim)
> summary(fit.lm)

Call:
lm(formula = y ~ x, data = data.sim)

Residuals:
     Min       1Q   Median       3Q      Max
-1.34869 -0.48331 -0.06187  0.41057  2.32666

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.57269    0.06898  -8.302 5.72e-13 ***
x            0.96290    0.07557  12.741  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.6864 on 98 degrees of freedom
Multiple R-squared:  0.6236, Adjusted R-squared:  0.6197
F-statistic: 162.3 on 1 and 98 DF,  p-value: < 2.2e-16
```

Linear regression: a simulated example (2)

```
> plot(x,y)
> abline(fit.lm)
>
> z1 <- rnorm(n)
> z2 <- rnorm(n)
> data.sim2 <- data.frame(y=y,x=x,z1=z1,z2=z2)
> fit.lm2 <- lm(y~x+z1+z2,data=data.sim2)
>
> anova(fit.lm,fit.lm2)
Analysis of Variance Table

Model 1: y ~ x
Model 2: y ~ x + z1 + z2
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     98 46.172
2     96 46.059  2   0.11307 0.1178  0.889
>
> F.test <- ((46.172-46.059)/2)/((46.059)/96)
> F.test
[1] 0.117762
> p.value.F.test <- 1-pf(F.test,2,96)
> p.value.F.test
[1] 0.8890358
```



Logistic regression (1)

Ingredients of a generalized linear model:

- Y_i are mutually independent and $E(Y_i) = m_i g^{-1}(\eta_i)$;
- $g(\cdot)$ is the link-function;
- $\eta_i = \sum_{j=1}^{p} \beta_j x_{ij}$ is the linear predictor.

Logistic regression (1)

Ingredients of a generalized linear model:

- Y_i are mutually independent and $E(Y_i) = m_i g^{-1}(\eta_i)$;
- $g(\cdot)$ is the link-function;
- $\eta_i = \sum_{j=1}^{p} \beta_j x_{ij}$ is the linear predictor.

Logistic regression

- $Y_i \sim$ Binomial(m_i, p_i), hence $E(Y_i) = m_i p_i$.
- $g(\eta_i) = \log\{p_i/(1 - p_i)\}$

Logistic regression (1)

Ingredients of a generalized linear model:

- Y_i are mutually independent and $E(Y_i) = m_i g^{-1}(\eta_i)$;
- $g(\cdot)$ is the link-function;
- $\eta_i = \sum_{j=1}^{p} \beta_j x_{ij}$ is the linear predictor.

Logistic regression

- $Y_i \sim$ Binomial(m_i, p_i), hence $E(Y_i) = m_i p_i$.
- $g(\eta_i) = \log\{p_i/(1 - p_i)\}$

```
> beta0 <- -0.5
> beta1 <- 1
>
> sigma2 <- 0.5
>
> n <- 100
>
> set.seed(123)
> x <- rnorm(n)
> eta <- beta0+beta1*x+rnorm(n,sd=sqrt(sigma2))
> m <- rep(10,n)
> y <- rbinom(n,size=m,prob=exp(eta)/(1+exp(eta)))
> data.bin.sim <- data.frame(y=y,m=m,x=x)
>
> fit.glm <- glm(cbind(y,m-y)~x,data=data.bin.sim,
+             family=binomial(link="logit"))
```

Logistic regression (2)

```
> summary(fit.glm)

Call:
glm(formula = cbind(y, m - y) ~ x, family = binomial(link = "logit"),
    data = data.bin.sim)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9524  -1.0052  -0.1079   0.8931   4.3163

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.63056    0.07223  -8.731   <2e-16 ***
x            0.90179    0.08587  10.502   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 314.85  on 99  degrees of freedom
Residual deviance: 183.13  on 98  degrees of freedom
AIC: 416.72

Number of Fisher Scoring iterations: 4
```

Logistic regression (3)

```
> z1 <- rnorm(n)
> z2 <- rnorm(n)
> data.bin.sim2 <- data.frame(y=y,m=m,x=x,z1=z1,z2=z2)
> fit.glm2 <- glm(cbind(y,m-y)~x+z1+z2,
+                 data=data.bin.sim2,
+                 family=binomial(link="logit"))
> logLik(fit.glm)
'log Lik.' -206.3582 (df=2)
> logLik(fit.glm2)
'log Lik.' -204.9515 (df=4)
> stat.test <- as.numeric(2*(logLik(fit.glm2)-logLik(fit.glm)))
> p.value <- 1-pchisq(stat.test,2)
> p.value
[1] 0.2449481
>
> exp(coef(fit.glm2))
(Intercept)           x          z1          z2
  0.5394988   2.4910965   0.9218388   1.0812969
>
> exp(confint(fit.glm2))
Waiting for profiling to be done...
                2.5 %    97.5 %
(Intercept) 0.4670885 0.6213581
x           2.1102030 2.9601158
z1          0.8016522 1.0599070
z2          0.9459411 1.2369409
```

Exercise

Admissions to graduate school

Read the data-frame ``admissions.txt''. The data-frame contains the following variables.

- `amdit`: 1 - admitted; 0 - not admitted.
- `gre`: Graduate Record Exam score.
- `gpa`: Grade Point Average score.
- `rank`: Institution prestige (1 - very low; 4 - very high).

Questions.

1. Use the function `prop.table` to obtain the distribution of `amdit` for each given `rank`. What is the proportion of admitted students in each rank?

Exercise

Admissions to graduate school (continue)

2. Use the function `cut` to define interval classes for `gre` and `gpa`. Plot the logit of the proportion of admitted against the average value within each class of the two variables. What type of relationship do you observe?
3. Fit a logistic regression which includes `gre`, `gpa` and `rank` to model the probability of admission. Use the variable `rank` as a factor using the contrasts in Slide 4. How do you interpret the estimated regression coefficients?
4. What is the probability of a student with average `gre` and `gpa` to be admitted in an institution of highest prestige?