

Comandes d'R per fer la vida més senzilla

Models Lineals

Inicialització de coses

Suposant que existeix un fitxer `./data.csv` d'on treiem les dades, això és el preàmbul a qualsevol `.rmd`:

```
setwd('.')
library(car)
library(emmeans)
library(tables)
library(RcmdrMisc)
dd<-read.csv2("data.csv")
```

Suposem que tenim un factor (variable categòrica) A amb nivells

$$A = [a_1, a_2, a_3, \dots]$$

Aleshores, si es troba a `dd` farem `dd$A<-as.factor(dd$A)` per llegir-la com un factor.

Per pekejar unes dades farem `head(dd)`.

Creació d'un model lineal

Per fer un model lineal amb la fórmula

$$Y_i = \beta_0 + x_{i1}\beta_1 + \dots x_{i(p-1)}\beta_{p-1} + e_i = \mu_i + e_i$$

farem

```
mod <- lm(Y~x_1+...+x_p, dd)
m0 <- lm(Y~1, dd) #sempre posem el model nul, per si de cas
```

Per comprovar els paràmetres i els p-valors i més, fem

```
summary(mod)
summary(mod)$sigma^2
Anova(mod) #tests de tipus II
anova(mod, mod0) #comparació dels models mod i m0
```

Per fer els `prediction intervals` (la regió de predicció) d'un model sobre unes dades

$$D_1 = [d_{11}, d_{12}, \dots, d_{1n_1}], \quad D_2 = [d_{21}, d_{22}, \dots, d_{2n_2}], \quad (\dots), \quad D_m = [d_{m1}, \dots, d_{mn_m}]$$

fem

```
predict(mod, data.frame(D1=c(d11,d12,...,d1n), D2=c(d21,d22,...,d2n), ...),  
interval="prediction")
```

Per fer `residuals vs. predictions` fem

```
plot(predict(mod),resid(mod))  
abline(h=0,lt=2)
```

Pels `studentized residuals`,

```
plot(rstudent(mod))  
abline(h=c(-3,-2,0,2,3),lt=2)
```

Comparació entre models

Tenim els models `ma,mb,mc,md`, i els volem comparar

```
rbind("logLik"=c("Model A"=logLik(ma),"Model B"=logLik(mb),"Model  
C"=logLik(mc),"Model D"=logLik(md)),  
"AIC"=c(AIC(ma),AIC(mb),AIC(mc),AIC(md)),  
"BIC"=c(BIC(ma),BIC(mb),BIC(mc),BIC(md)),  
"R2"=c(summary(ma)$adj.r.squared,summary(mb)$adj.r.squared,summary(mc)$adj.r.squ  
ared,summary(md)$adj.r.squared))
```

Test d'homocedasticitat

Per comprovar si un model compleix la homocedasticitat en els residuals, fem

```
leveneTest(resid(mod),A)
```

on A és un factor que defineix els grups d'on volem comprovar les variàncies.

Podria haver-hi interacció

Aleshores, es fa un

```
emmip(mod, formula | variable_explicativa_condicionant, cis=T)  
#l'última part és pels intervals de confiança
```

i així es pot veure si la interacció és molt evident. En cas que sigui convenient, aleshores cal fer

```
emm <- emmeans(mod, formula | variable_explicativa_condicionant)
```

També existeix `emtrends`, que calcula les tendències en un model ajustat quan tenim una variable numèrica interactuant amb una altra de categòrica. Per tant, com que hi ha interacció, mirem entre quines n'hi ha, i si és el cas, considerem el model amb interacció.

Per veure quines variables tenen possiblement multicolinearietat, fem `pairs(emm)`. Aquesta comanda ens donarà informació sobre quines variables tenen diferència esperada igual a zero (hipòtesi nul·la)

Errors

Els errors han de ser lineals amb els percentils d'una $N(0, \sigma^2)$. Això es pot comprovar amb

```
qqnorm(resid(mod))
qqline(resid(mod), lty = 2)
```

Si no s'observa relació lineal, alguna cosa hem fet malament. A més, **residuals vs. predits ha de no tenir cap tendència**.

Models Lineals Generalitzats

Descriptiva de les dades

```
print(tabular(Y~((Dades=1)+(Mitjana=mean)+
(Desv.Tipus=sd)),dd),justification="r")

with(dd,plotMeans(GMD,Y,error.bars = "sd"))
```

Models i resums

```
summary(mg <- glm(Y~X, family=gaussian, dd))
mg0 <- glm(GMD~1, family=gaussian, dd)
emmip(mg,~X,CIs=T) #aquí s'hi pot afegir si volem que ens doni el gràfic tipus
resposta amb ty="response"
```

Cal notar que `family` pot ser qualsevol d'entre:

- `[binomial(link="logit)`
- `gaussian(link="identity)`
- `Gamma(link="inverse")`
- `inverse.gaussian(link = "1/mu^2")`
- `poisson(link = "log")`

Per comprovar les estadístiques del model, fem

```
Anova(mg, test.statistic="F") #test.statistic pot ser Chisq o F en general
```

Per comparar dos models, fem

```
anova(mg0,mg,test="F") #test pot ser Chisq o F en general
```

Pel paràmetre de dispersió, `summary(mg)$dispersion`. Si es vol estimar les esperances marginals, o fer comparacions múltiples,

```
emm <- emmeans(mg,~X)
pairs(emm)
CLD(emm)
```

Residuals

`residuals vs. predicted:`

```
plot(resid(mg,ty="pearson")~X)
abline(h=0,lty=2)
plot(mg,1)
residualPlot(mg)
```

Residus d'un model: `resid(mg,ty="pearson")`