SIRE516

Basic data analyses

Disclaimer

- This is not a statistics class so I will not cover all statistical aspects behind each analysis.
- I will cover regularly used analyses.
- However, I can provide detailed explanations if time permits.

Exploratory data analysis

Understanding your data

- Before any analysis
  - Read the data dictionary
  - Consult the source (e.g. ‘wet lab’ people, physicians)
- Type of data
  - Transformation?
  - Type of test and analysis?

Understanding your data

- General data types
  - Categorical
    - Nominal
    - Ordinal
  - Numerical
    - Arbitrary (uniform distribution?)
    - Normal distribution
    - Other: Binomial, Poisson, Beta, etc

Histogram

```
1  ###Histogram
2  library(MASS)
3  ?iris
4
5  #Which parameter is normally distributed?
6  truehist(iris$Sepal.Length)
7  truehist(iris$Sepal.Width)
8  truehist(iris$Petal.Length)
9  truehist(iris$Petal.Width)
10
```

Quantile-Quantile plot

```
#### Q-Q plot ####
library(car)

#Which parameter is normally distributed?
qqPlot(iris$Sepal.Length)
qqPlot(iris$Sepal.Width)
qqPlot(iris$Petal.Length)
qqPlot(iris$Petal.Width)
```

Test of normality

```
20  #### Test of Normality
21
22  #Which parameter is normally distributed?
23  shapiro.test(iris$Sepal.Length)
24  shapiro.test(iris$Sepal.Width)
25  shapiro.test(iris$Petal.Length)
26  shapiro.test(iris$Petal.Width)
```

What to do if your data is not normally distributed?

- Transformation
  - Log10: Viral load
- Non-parametric analyses
  - No transformation
  - Arbitrary values

```
#### Data transformation ####
dat <- rnorm(100,mean = 5,sd = 2) # Generate 100 numbers with mean = 5 and sd = 2 (Normal distribution)
truehist(dat)

expdat <- exp(dat) #e^x ~ 2.718282^x
truehist(expdat)

shapiro.test(dat)
shapiro.test(expdat)

tdat <- log10(expdat + 1) # + 1 is important because log10(0) is undefined!
truehist(tdat)
shapiro.test(tdat)

```

How to present values summarizing your data

- Categorical
  - Count (Percent of count of all category)

```r
#### Summarize categorical value ####
rows <- sample(nrow(iris),91, replace = F) # Sample 91 rows from the iris dataset
IR <- iris[rows,]

tb <- table(IR$Species)
print(tb)

percent <- round(100 * tb/nrow(IR),1)
print(percent)

paste0(tb," (",percent,")")
```

How to present values summarizing your data

- Numerical
  - Mean ± Standard deviation
  - Normal distribution only

```
54 #### Summarize numerical value (Mean SD) ####
55 m <- round(mean(IR$Sepal.Length),1)
56 s <- round(sd(IR$Sepal.Length),1)
57 paste0(m, " ± ", s)
58
59 lm <- round(mean(exp(IR$Sepal.Length)),1)
60 ls <- round(sd(exp(IR$Sepal.Length)),1)
61 paste0(lm, " ± ", ls) # SD >= mean implying deviation from normal distribution
62
63 llm <- round(mean(log10(1+ exp(IR$Sepal.Length))),1)
64 lls <- round(sd(log10(1+exp(IR$Sepal.Length))),1)
65 paste0(llm, " ± ", lls)
```

How to present values summarizing your data

- Numerical
  - Median (Interquartile range, IQR, Q25 to Q75)
  - Median (Range, Min to Max)
  - Almost always suitable for numerical data

```
67  #### Summarize numerical value (Median) ####
68  q <- round(quantile(IR$Sepal.Length,c(0.5,0.25,0.75)),1)
69  paste0(q[1]," (",q[2],"-",q[3],")")
70  paste0(median(IR$Sepal.Length)," (",min(IR$Sepal.Length),"-",max(IR$Sepal.Length),")")
71
72  lq <- round(quantile(exp(IR$Sepal.Length),c(0.5,0.25,0.75)),1)
73  paste0(lq[1]," (",lq[2],"-",lq[3],")")
74  paste0(round(median(exp(IR$Sepal.Length)),1)," (",round(min(exp(IR$Sepal.Length)),1),
75         "-",round(max(exp(IR$Sepal.Length)),1),")")
```

Practical 1

Practical 1

- We will create a “baseline” characteristic table of the IRIS data set
- The table will contain four columns one for variable and three for each species
- The first row will be count (%) for each species

```
1  ?iris
2  summary(iris)
```

Hypothesis testing

Why?

- We cannot measure or test all existing data.
- We measure and test a small part (i.e. sample) of the data (i.e. population).
- How can we be sure the small part represents all the data?
  - Sample size → Type 1 & 2 errors
  - Hypothesis testing → Type 1 errors
- We **reject** a "Null" hypothesis
  - Sufficient evidence

Table of error types		Null hypothesis (H_0) is	
		True	**False**
Decision about null hypothesis (H_0)	**Don't reject**	Correct inference (true negative) (probability = $1-\alpha$)	Type II error (false negative) (probability = β)
	Reject	Type I error (false positive) (probability = α)	Correct inference (true positive) (probability = $1-\beta$)

https://en.wikipedia.org/wiki/Type_I_and_type_II_errors

One-sample testing

- Population mean or proportion was previously estimated
- We test whether the value found in our sample agrees with these values

One-sample test of proportion

- Example: Flipping a normal coin would result in a head in 50% of the trials
- You flip your coin 100 times and get a head 71 times. Is your coin normal or loaded?

```
77 #### One-sample proportion test ####
78 binom.test(x = 71, n = 100, p = 0.5) # Exact test
79
80 prop.test(x = 71, n = 100, p = 0.5) # Estimation
```

One-sample test of mean

- Example: A survey in 2000 found that Thai male first graders had a mean height of 105 cm with a standard deviation of 5 cm.
- You conduct a similar survey in 100 first graders and found that the mean height is now 107 cm with the same standard deviation. Are the male first graders getting taller?

```r
#### One-sample mean test ####
dat <- round(rnorm(n = 100, mean = 107, sd = 5),1) # Simulate the survey data
truehist(dat)

t.test(dat, mu = 105) # Parametric test
wilcox.test(dat, mu = 105) # Non-parametric test

dat2 <- round(runif(n = 100, min = 92, max = 122 ),1) # Simulate the survey data. Not normal distribution
truehist(dat2)
mean(dat2)

t.test(dat2, mu = 105) # Parametric test
wilcox.test(dat2, mu = 105) # Non-parametric test
```

Test difference in proportions

- Example: Proportion of patients with cancer for two populations
- chisq.test() and prop.test() for large samples (require different inputs)
- Exact test for smaller samples → We will focus on this test in Categorical data analysis

```
96  #### Difference in proportions ####
97  smoke <- 510
98  smoke_CA <- 400
99  non_smoke <- 540
100 non_smoke_CA <- 300
101
102 smoke_not_CA <- smoke - smoke_CA
103 non_smoke_not_CA <- non_smoke - non_smoke_CA
104
105 prop.test(x = c(non_smoke_CA,smoke_CA), n = c(non_smoke,smoke))
106
107 m <- matrix(data = c(non_smoke_CA, non_smoke_not_CA, smoke_CA, smoke_not_CA) , nrow = 2)
108 print(m)
109
110 chisq.test(m)
```

Test difference in means

- Example: Heights between male and female first graders.
- Parametric test: t.test
- Non-parametric test: wilcox.test

```
#### Difference in means ####
male <- round(rnorm(n = 100 ,mean = 107, sd = 5),1)
female <- round(rnorm(n = 100, mean = 105, sd = 4.5),1)

truehist(male)
truehist(female)

t.test(male,female, var.equal = F, paired = F)
wilcox.test(male,female, paired = F)

#paired = T for dependent samples (e.g. Before-After mearsurements)
```

Practical 2

Practical 2

- We will perform hypothesis testing with the “Melanoma” dataset.
- We will compare between males and females for any difference in each parameter
- For ‘status’, regroup to 1 vs 2+3

```
1  library(MASS)
2
3  ?Melanoma
4
5  female <- Melanoma[which(Melanoma$sex == 0),]
6  male  <- Melanoma[which(Melanoma$sex == 1),]
```

Categorical Data Analysis

Contingency table

- Help for categorical data analysis
- Example:

	Cancer (Case)	No cancer (Control)
Smoker	400	150
Non smoker	100	320
TOTAL	500	470

Contingency table

- Create a table from counts

```
124 #### Contingency table from counts ####
125 ctable <- matrix(c(400,150,100,320), nrow = 2 , byrow = T) # Matrix fill data by column by default
126 dimnames(ctable) <- list(c("Smoker","Non-smoker"),c("Cancer","No cancer"))
127 print(ctable)
128
129 matrix(c(400,150,100,320), nrow = 2)
```

Contingency table

- Create a table from raw data

```
131 #### Contingency table from raw data ####
132
133 #Simulate the data
134 rawdat <- data.frame(patientID = 1:(400+150+100+320),
135                      smoking = c(rep("Smoker",400+150),rep("Non-smoker",100+320)),
136                      status = c(rep("Cancer",400),rep("No cancer",150),rep("Cancer",100),rep("No cancer",320)))
137 View(rawdat)
138 summary(rawdat)
139
140 ctable2 <- table(rawdat[,c("smoking","status")])
141 print(ctable2)
```

Test of difference proportions

- chisq.test() → Estimation
  - **Caution:** if >20% of cells (e.g. 1 in 4 cells of a 2x2 contingency table) is <= 5, Chi-square will not be accurate.
- fisher.test() → Exact test
  - Can be used in most cases. But a large sample size will require intensive computation.

```
143 #### Test of difference proportions ####
144
145 chisq.test(ctable2)
146 fisher.test(ctable2)
147
148 small <- matrix(data = c(1,39,2,20), nrow = 2, byrow = T)
149 print(small)
150
151 chisq.test(small)
152 fisher.test(small)
```

Measure of association

- Question: If you decide to start smoking, how much more likely you will get cancer?
- Relative risk (RR)
  - *** Specifically for cohort or cross-sectional study***
  - **Not** for a case-control study
    - Cases and Controls were specifically recruited
    - Cases from a “rare” disease
    - True prevalence and incidence are lost

$$RR = \frac{\Pr(Disease|Exposed)}{\Pr(Disease|Not\ exposed)}$$

Measure of association

- Odds ratio (OR)
  - Can be used with cross-sectional, cohort and case-control studies
  - Logistic regression

$$Odds\ of\ an\ event = \frac{\Pr(event\ will\ occur)}{\Pr(event\ will\ not\ occur)}$$

$$OR = \frac{odds\ that\ an\ exposed\ person\ develops\ a\ disease}{odds\ that\ a\ non-exposed\ person\ develops\ a\ disease}$$

Measure of association

	Disease	No disease	Total
Exposed	a	b	a + b
Non-exposed	c	d	c + d
TOTAL	a + c	b + d	a + b + c + d

- $RR = \frac{\frac{a}{a+b}}{\frac{c}{c+d}}$

- $OR = \frac{\frac{a}{b}}{\frac{c}{d}}$

= 1 → No association

< 1 → Negative association (protection)

> 1 → Positive association (elevation)

Measure of association

- Package “epitools”

```
143  #### Measure of association
144  library(epitools)
145
146  #Re-arrange data to match epitools requirement
147  rawdat$status <- factor(rawdat$status, levels = c("No cancer","Cancer"))
148  rawdat$smoking <- factor(rawdat$smoking, levels =c("Non-smoker","Smoker"))
149
150  ctable3 <- table(rawdat[,c("smoking","status")])
151  print(ctable3)
152
153  res <- oddsratio(ctable3, method = "wald")
154  print(res)
155
156  res2 <- riskratio(ctable3, method = "wald")
157
158  print(res2)
```

Practical 3

Practical 3

- We will calculate odds ratios of alcohol and tobacco consumption in the “esoph” dataset
- Alcohol consumption → 0-39g/day versus >=40g/day
- Tobacco consumption → 0-9g/day versus >=10g/day

```
1  library(MASS)
2  library(epitools)
3
4  ?esoph
5  esoph$alc <- factor(esoph$alcgp, levels(esoph$alcgp), labels = c("0-39g/day",rep(">=40g/day",3)))
6  esoph$tob <- factor(esoph$tobgp, levels(esoph$tobgp), labels = c("0-9g/day",rep(">=10g/day",3)))
```

Linear Regression

Correlation between two numerical variables

- By knowing X → Estimate Y
- Parametric: Pearson (default)
- Non-parametric: Spearman and Kendall

```
160 #### Correlation between numerical variables ####
161 ?iris
162
163 cor.test(iris$Sepal.Length,iris$Petal.Width) #Paired values
164 cor.test(~Sepal.Length + Petal.Width, iris) #Data.frame
165
166 cor.test(iris$Sepal.Length,iris$Petal.Width, method = "s") #Non-paramatric
167 cor.test(iris$Sepal.Length,iris$Petal.Width, method = "k") #Non-paramatric
```

Correlation between two numerical variables

- P-value:
  - Null hypothesis: the coefficient of correlation = 0
- Interpretation of the coefficient of correlation

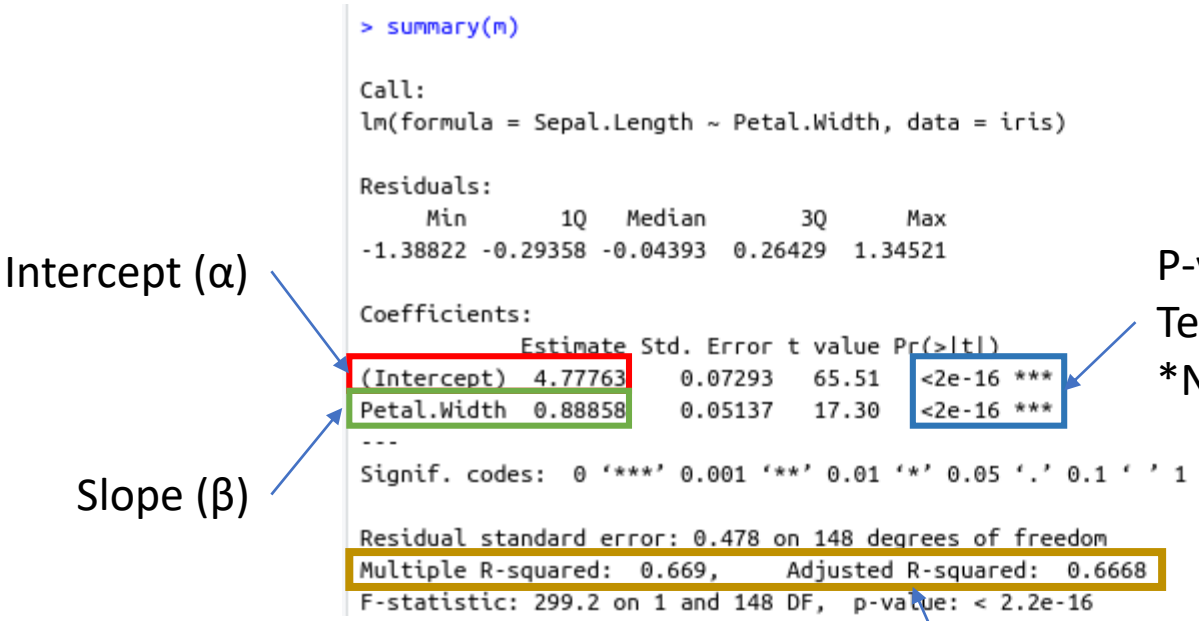
Correlation Coefficient Value	Direction and Strength of Correlation
−1.0	Perfectly negative
−0.8	Strongly negative
−0.5	Moderately negative
−0.2	Weakly negative
0.0	No association
+0.2	Weakly positive
+0.5	Moderately positive
+0.8	Strongly positive
+1.0	Perfectly positive

By knowing X → Estimate Y

- Simple linear regression: $y = \alpha + \beta x + \varepsilon$
  - α = intercept
  - β = slope
  - ε = error (mean of ε = 0)
  - $\hat{y} = \alpha + \beta x$

```
169 #### Simple linear regression ####
170
171 m <- lm(Sepal.Length ~ Petal.Width   , iris)
172 print(m)
173 summary(m)
```

Interpreting regression result



```
> summary(m)

Call:
lm(formula = Sepal.Length ~ Petal.Width, data = iris)

Residuals:
     Min       1Q   Median       3Q      Max
-1.38822 -0.29358 -0.04393  0.26429  1.34521

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.77763    0.07293   65.51   <2e-16 ***
Petal.Width  0.88858    0.05137   17.30   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.478 on 148 degrees of freedom
Multiple R-squared:  0.669,     Adjusted R-squared:  0.6668
F-statistic: 299.2 on 1 and 148 DF,  p-value: < 2.2e-16
```


P-values:
Test whether any coefficient = 0
*Normally, the intercept is kept regardless of p-value

R^2 → % of Y explained (predicted) by your model
Used adjusted R^2 for multivariate regression

Multivariate linear regression

- Add more independent variables for prediction
- Categorical variables can be added
  - R does not need recoding of categorical variables for regression

```
175 #### Multivariate linear regression ####
176
177 mm <- lm(Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width + Species , iris)
178 print(mm)
179 summary(mm)
```

Multivariate linear regression

-0.723 is added, if and only if, species = versicolor

-1.023 is added, if and only if, species = virginica

```
> summary(mm)

Call:
lm(formula = Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width +
    Species, data = iris)

Residuals:
     Min       1Q   Median       3Q      Max
-0.79424 -0.21874  0.00899  0.20255  0.73103

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        2.17127    0.27979   7.760 1.43e-12 ***
Sepal.Width        0.49589    0.08607   5.761 4.87e-08 ***
Petal.Length       0.82924    0.06853  12.101  < 2e-16 ***
Petal.Width       -0.31516    0.15120  -2.084  0.03889 *
Speciesversicolor -0.72356    0.24017  -3.013  0.00306 **
Speciesvirginica  -1.02350    0.33373  -3.067  0.00258 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3068 on 144 degrees of freedom
Multiple R-squared:  0.8673,    Adjusted R-squared:  0.8627
F-statistic: 188.3 on 5 and 144 DF,  p-value: < 2.2e-16
```

These values are not added if species = setosa (i.e. setosa is set as a reference)

Inhibit Interpretation/Conversion of Objects

- Function I() in the regression formula allows you to modify/calculate a variable before being fed into the model

```
181 #### Inhibit Interpretation/Conversion of Objects ####
182 mlog <- lm(Sepal.Length ~ I(log10(1+Petal.Width)), iris)
183 print(mlog)
184 summary(mlog)
185 
186 msq <- lm(Sepal.Length ~ I(Petal.Width^2), iris) # Quadratic model
187 print(msq)
188 summary(msq)
```

Interaction

- Interaction between variables could be added to the model
- For example: male and female might have difference slope of height ~ weight
- Any interaction must be determined whether it is important or relevant first. Do not rely solely on statistics/statisticians!
- If a interaction is kept in the model, all interacting variables must also be kept regardless of p-values.

```
190 #### Interaction ####
191 mint <- lm(Sepal.Length ~ Petal.Width*Species, iris)
192 print(mint)
193 summary(mint)
```

Model selection

- Keep adding variables with limited improvement in performance is not optimal.
  - Simplest model with optimal performance
- Akaike's An Information Criterion (AIC) measures a trade-off between adding variables and performance improvement

```
195 #### Model selection ####
196 library(MASS)
197
198 sm <- stepAIC(mint)
199 print(sm)
200 summary(sm)
```

Model selection

- ANOVA could be used to compare two models whether a sufficient improvement is observed in the more complex model.
- P < 0.05 → Improvement

```
202 #### Model selection ANOVA ####
203
204 anova(m,mint)
205 anova(m,mm)
```

Prediction with linear regression

- New dataset must have the identical structure as the training dataset

```
218 #### Prediction with linear regression ####
219 newdat <- iris[4:9,]
220 print(newdat)
221 
222 predict(mm, newdata = newdat)
```

Practical 4

Practical 4

- We will perform multivariate regression analysis on the “Davis” dataset
- We will determine how sex, height and interaction between sex and height affect the weight
- Select your best model

```
1  library(car)
2
3  ?Davis
4
5  #Correct errors in the dataset
6  print(Davis[10:13,])
7
8  nDavis <- Davis
9  nDavis[12,2:3] <- Davis[12,3:2]
10
11 #Use nDavis dataset, not Davis
```

Logistic Regression

Binary outcomes

- Linear regression → numerical outcome
- Logistic regression → binary outcome (e.g. cancer vs no cancer)
- Regression results in $(-\infty, \infty)$
- Probability of binary outcome is [0,1]
- Convert probability to a real number scale
  - Log of odds → Logit

Binary outcomes

- Convert probability to a real number scale
  - Log of odds → Logit

```
#### Logit ####
library(ggplot2)

pt <- 10^4 #Try changing number of point

dat <- data.frame(id = 1:pt, prob = seq(0,1,length.out = pt))
ggplot(dat,aes(x = id, y = prob))+
  geom_line()

dat$odds <- dat$prob/(1-dat$prob)
ggplot(dat,aes(x = id, y = odds))+
  geom_line()

dat$logit <- log(dat$odds)
ggplot(dat,aes(x = id, y = logit))+
  geom_line()
```

Setting logistic regression model

- Setting and selecting logistic regression models are similar to linear regression
- The difference is the interpretation of the result

```
235 #### Setting a logistic regression model ####
236 #From https://towardsdatascience.com/how-to-do-logistic-regression-in-r-456e9cfec7cd
237 library(AER)
238 data(Affairs)
239 ?Affairs
240
241 Affairs$ynaffair[Affairs$affairs >  0] <- 1
242 Affairs$ynaffair[Affairs$affairs == 0] <- 0
243 Affairs$ynaffair <- factor(Affairs$ynaffair,levels=c(0,1), labels=c("No","Yes"))
244 table(Affairs$ynaffair)
245
246 lgm <- glm(ynaffair ~ gender + age + yearsmarried + children
247            + religiousness + education + occupation +rating,
248            data=Affairs, family="binomial")
249 print(lgm)
250 summary(lgm)
```

Setting logistic regression model

- The difference is the interpretation of the result

```
> summary(lgm)

Call:
glm(formula = ynaffair ~ gender + age + yearsmarried + children +
    religiousness + education + occupation + rating, family = "binomial",
    data = Affairs)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5713  -0.7499  -0.5690  -0.2539   2.5191

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.37726    0.88776   1.551 0.120807
gendermale    0.28029    0.23909   1.172 0.241083
age          -0.04426    0.01825  -2.425 0.015301 *
yearsmarried  0.09477    0.03221   2.942 0.003262 **
childrenyes   0.39767    0.29151   1.364 0.172508
religiousnes -0.32472    0.08975  -3.618 0.000297 ***
education     0.02105    0.05051   0.417 0.676851
occupation    0.03092    0.07178   0.431 0.666630
rating       -0.46845    0.09091  -5.153 2.56e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 675.38  on 600  degrees of freedom
Residual deviance: 609.51  on 592  degrees of freedom
AIC: 627.51

Number of Fisher Scoring iterations: 4
```

For reporting, coefficients must be converted to odds ratio

P-values are still used to determine the significance of each variable

Reporting odds ratio from regression results

```
252 #### Reporting odds ratio from regression results ####
253
254 exp(cbind(OR = coef(lgm), confint(lgm)))
```

```
                    OR     2.5 %     97.5 %
(Intercept)   3.9640180 0.7013467 22.9148790
gendermale    1.3235091 0.8294178  2.1209988
age           0.9567099 0.9223032  0.9908864
yearsmarried  1.0994093 1.0326203  1.1718726
childrenyes   1.4883560 0.8451473  2.6584834
religiousness 0.7227292 0.6049441  0.8605325
education     1.0212740 0.9254481  1.1284991
occupation    1.0314027 0.8964089  1.1884105
rating        0.6259691 0.5227302  0.7470069
```

Prediction with logistic regression

- New dataset must have the identical structure as the training dataset
- Use type = “response” argument to predict a probability of the event

```
256 #### Logistic regression prediction ####
257
258 newdat <- Affairs[c(4,5,9,10),]
259 predict(lgm, newdata = newdat, type = "response")
```

Practical 5

Practical 5

- We will analyze “Affairs” dataset
- Select the best logistic regression model that predict an affair (e.g. anova, stepAIC)
- Format the type of each variable (e.g. factor, numeric)

```r
#From https://towardsdatascience.com/how-to-do-logistic-regression-in-r-456e9cfec7cd
library(AER)
data(Affairs)
?Affairs

Affairs$ynaffair[Affairs$affairs >  0] <- 1
Affairs$ynaffair[Affairs$affairs == 0] <- 0
Affairs$ynaffair <- factor(Affairs$ynaffair,levels=c(0,1), labels=c("No","Yes"))
table(Affairs$ynaffair)
```