HW10

Kuo, Chien-Yu

以過去修過的課程為主，整理使用過的R code

- 敘述統計
- 分配
- 檢定
- 分析

一、敘述統計

1. 資料的基本型態

```
#以R內建的資料"airquality"為例
#觀察前幾筆資料
head(airquality)
```

```
##   Ozone Solar.R Wind Temp Month Day
## 1    41     190  7.4   67     5   1
## 2    36     118  8.0   72     5   2
## 3    12     149 12.6   74     5   3
## 4    18     313 11.5   62     5   4
## 5    NA      NA 14.3   56     5   5
## 6    28      NA 14.9   66     5   6
```

```
#計算資料的相關統計量
summary(airquality)
```

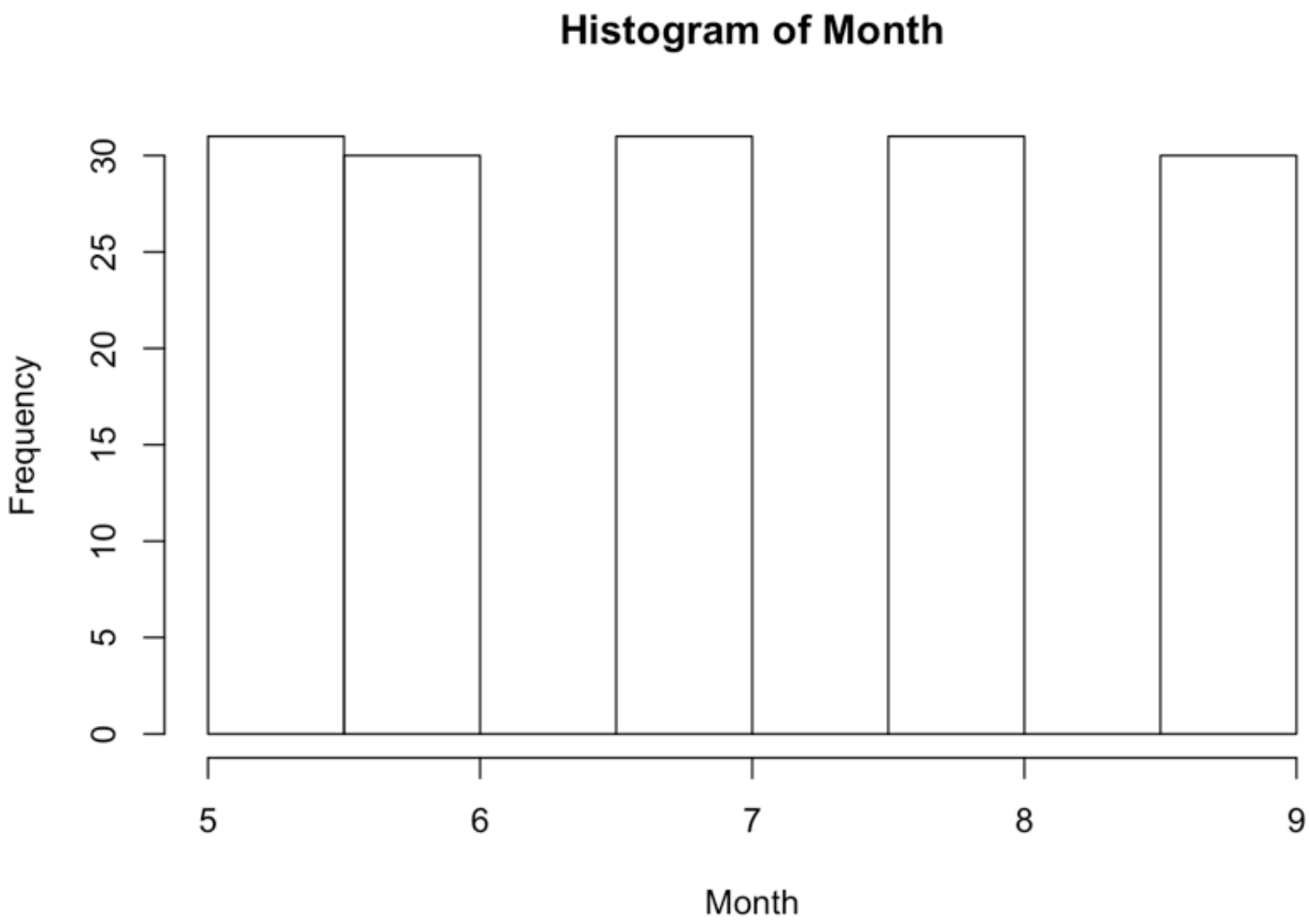
```
##      Ozone           Solar.R          Wind             Temp      
##  Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :56.00  
##  1st Qu.: 18.00   1st Qu.:115.8   1st Qu.: 7.400   1st Qu.:72.00  
##  Median : 31.50   Median :205.0   Median : 9.700   Median :79.00  
##  Mean   : 42.13   Mean   :185.9   Mean   : 9.958   Mean   :77.88  
##  3rd Qu.: 63.25   3rd Qu.:258.8   3rd Qu.:11.500   3rd Qu.:85.00  
##  Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00  
##  NA's   :37       NA's   :7                                       
##      Month            Day       
##  Min.   :5.000   Min.   : 1.0  
##  1st Qu.:6.000   1st Qu.: 8.0  
##  Median :7.000   Median :16.0  
##  Mean   :6.993   Mean   :15.8  
##  3rd Qu.:8.000   3rd Qu.:23.0  
##  Max.   :9.000   Max.   :31.0  
##
```

2. 數據視覺化

同樣拿“airquality”為例

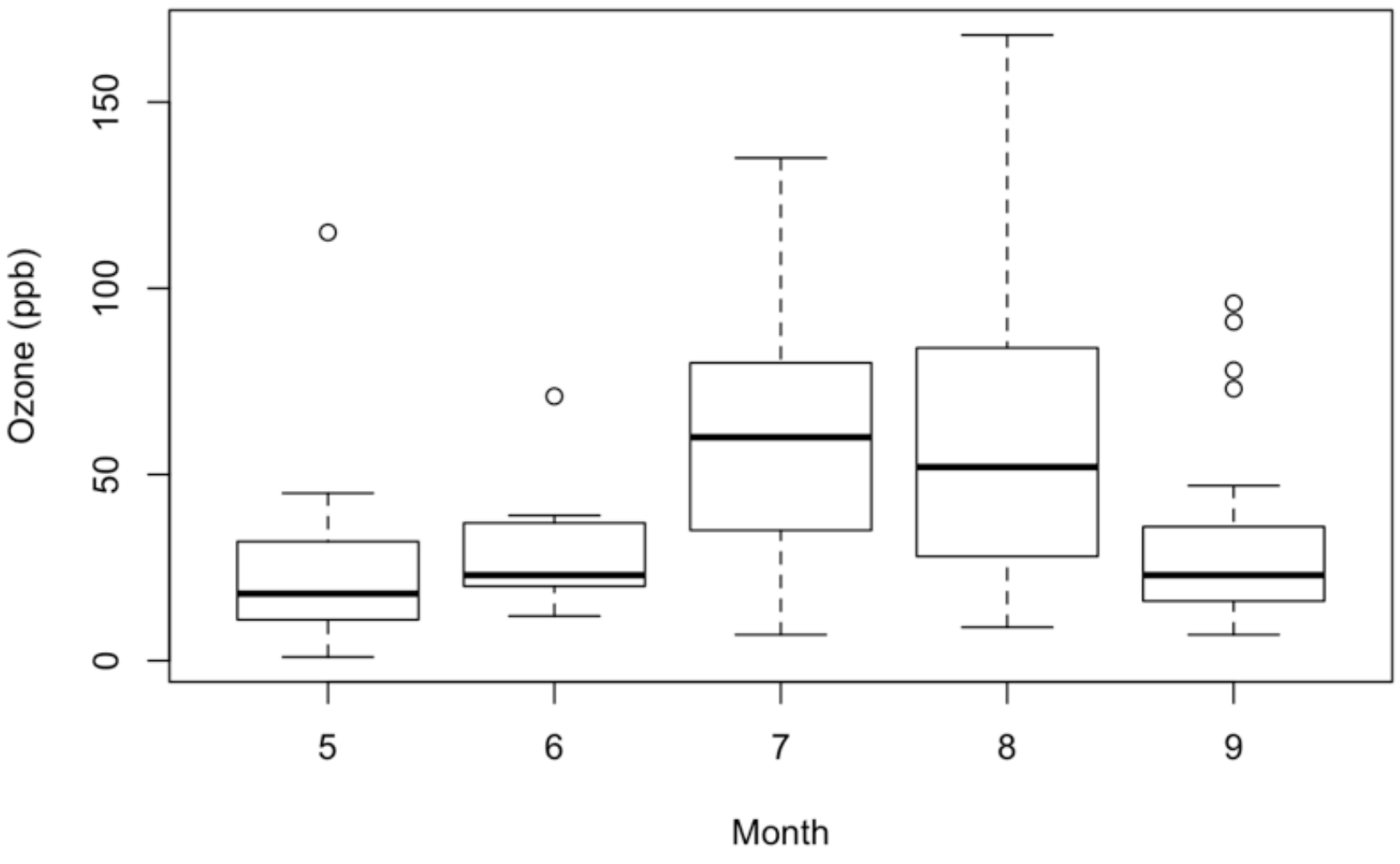
a. histogram

```
#想知道該筆data每個月份分別有幾筆資料：
hist(x=airquality$Month,
     main="Histogram of Month",          # 圖片的名稱
     xlab="Month",                       # X軸的名稱
     ylab="Frequency")                   # Y軸的名稱
```



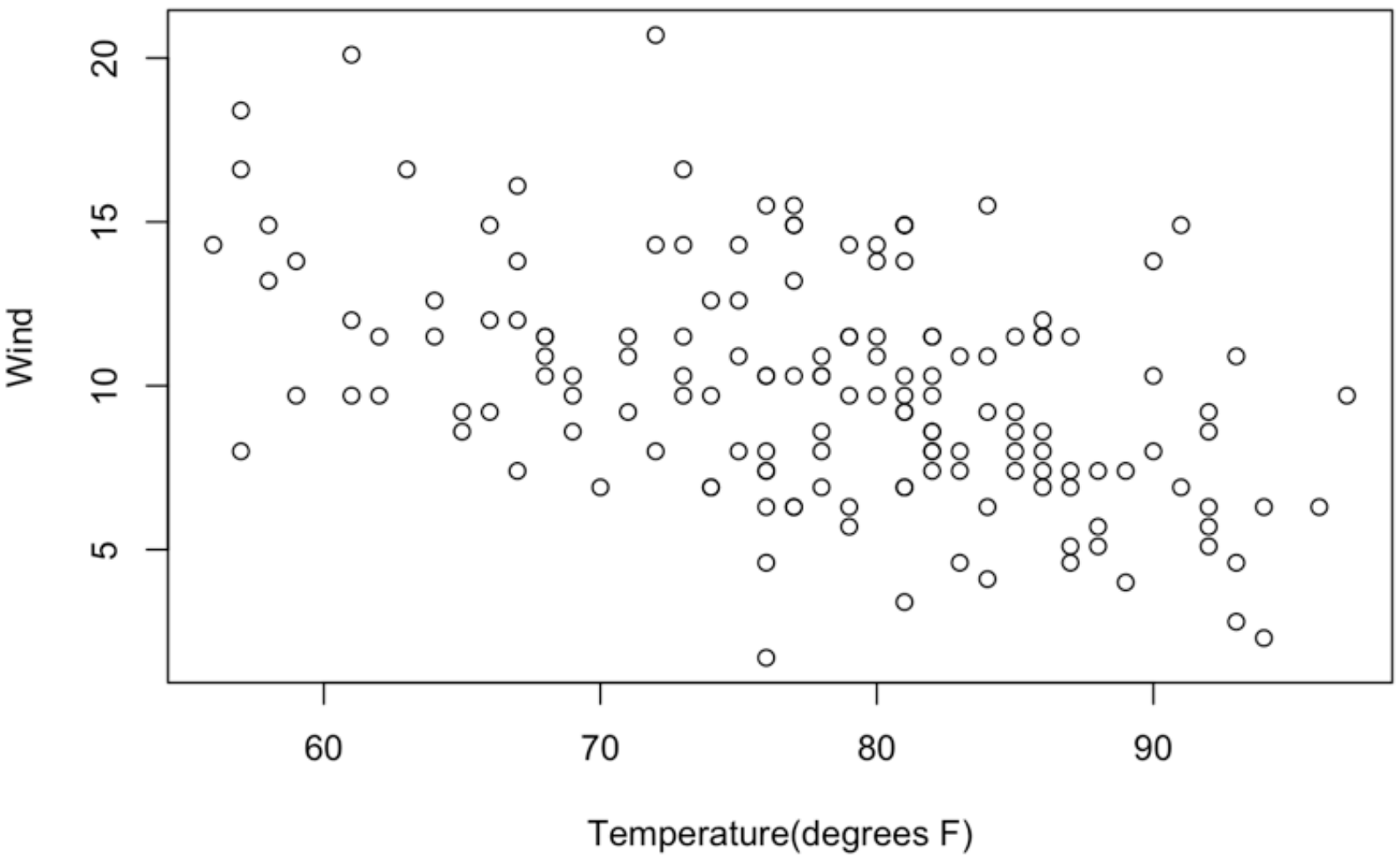
b. boxplot

```
#想知道該筆data不同月份的臭氧(Ozone)數值的分布情況：
boxplot(formula = Ozone ~ Month, # Y ~ X (代表X和Y軸要放的數值)
        data = airquality,       # 資料
        xlab = "Month",          # X軸名稱
        ylab = "Ozone (ppb)")    # Y軸名稱
```



c. plot：為散佈圖

```
#想知道溫度(Temp)和風速(Wind)之間的關係：
plot(x=airquality$Temp,                # X軸的值
     y=airquality$Wind,                # Y軸的值
     main="Temp to Wind",              # 圖片名稱
     xlab="Temperature(degrees F)", # X軸名稱
     ylab="Wind")                      # Y軸名稱
```



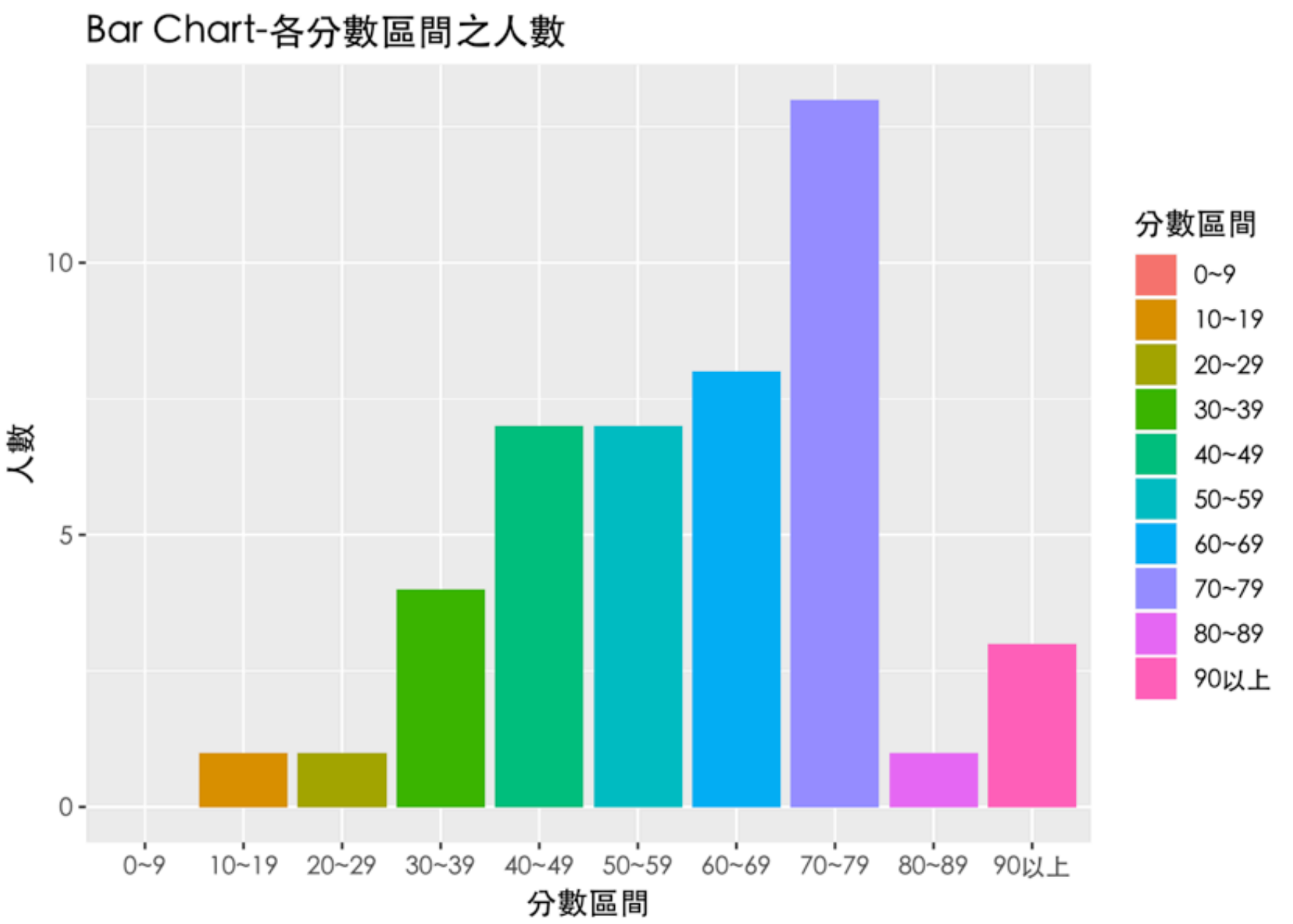
3. 利用ggplot2呈現

install.packages(“ggplot2”)

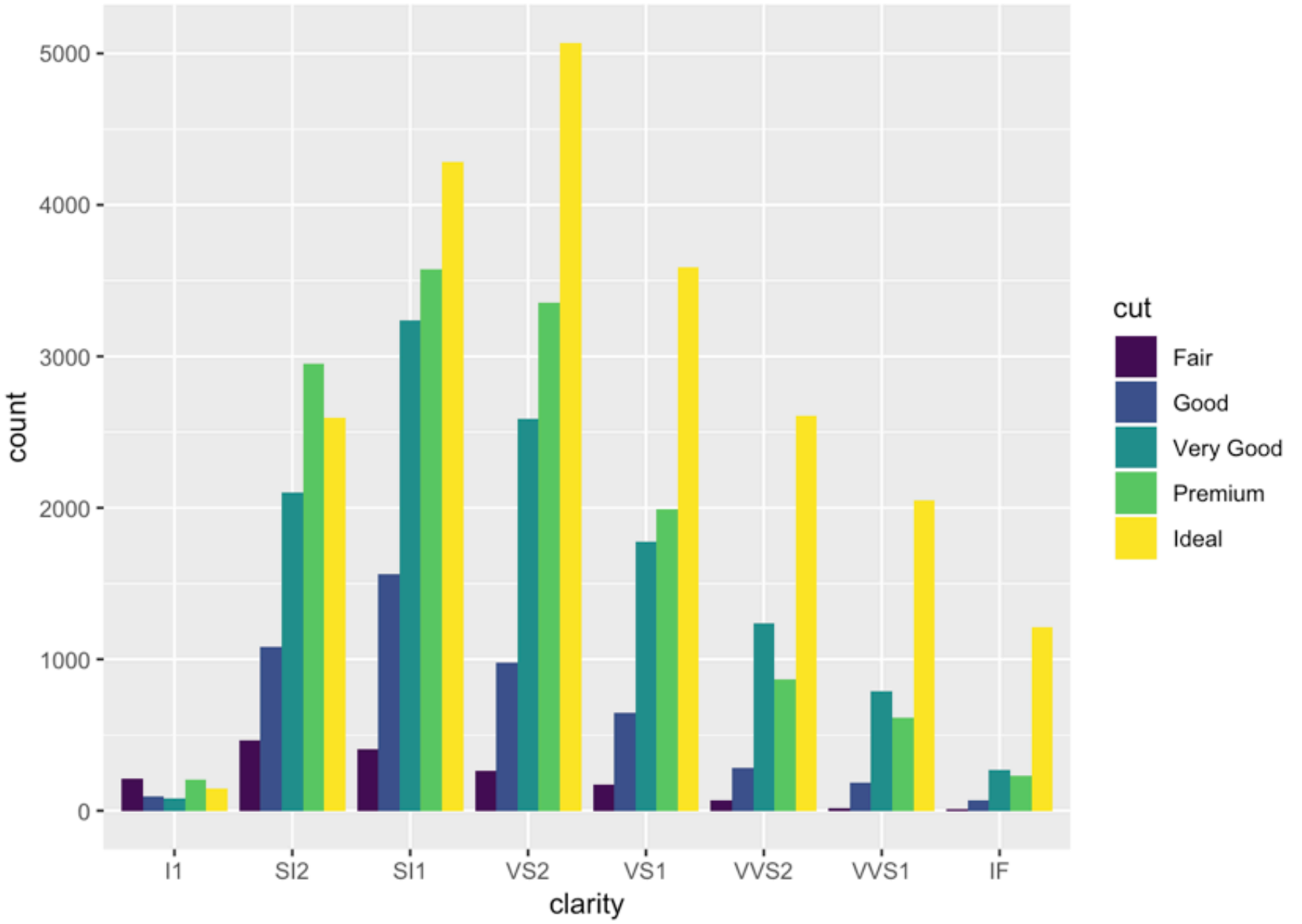
```
library("ggplot2")
```

```
## Warning: package 'ggplot2' was built under R version 3.5.2
```

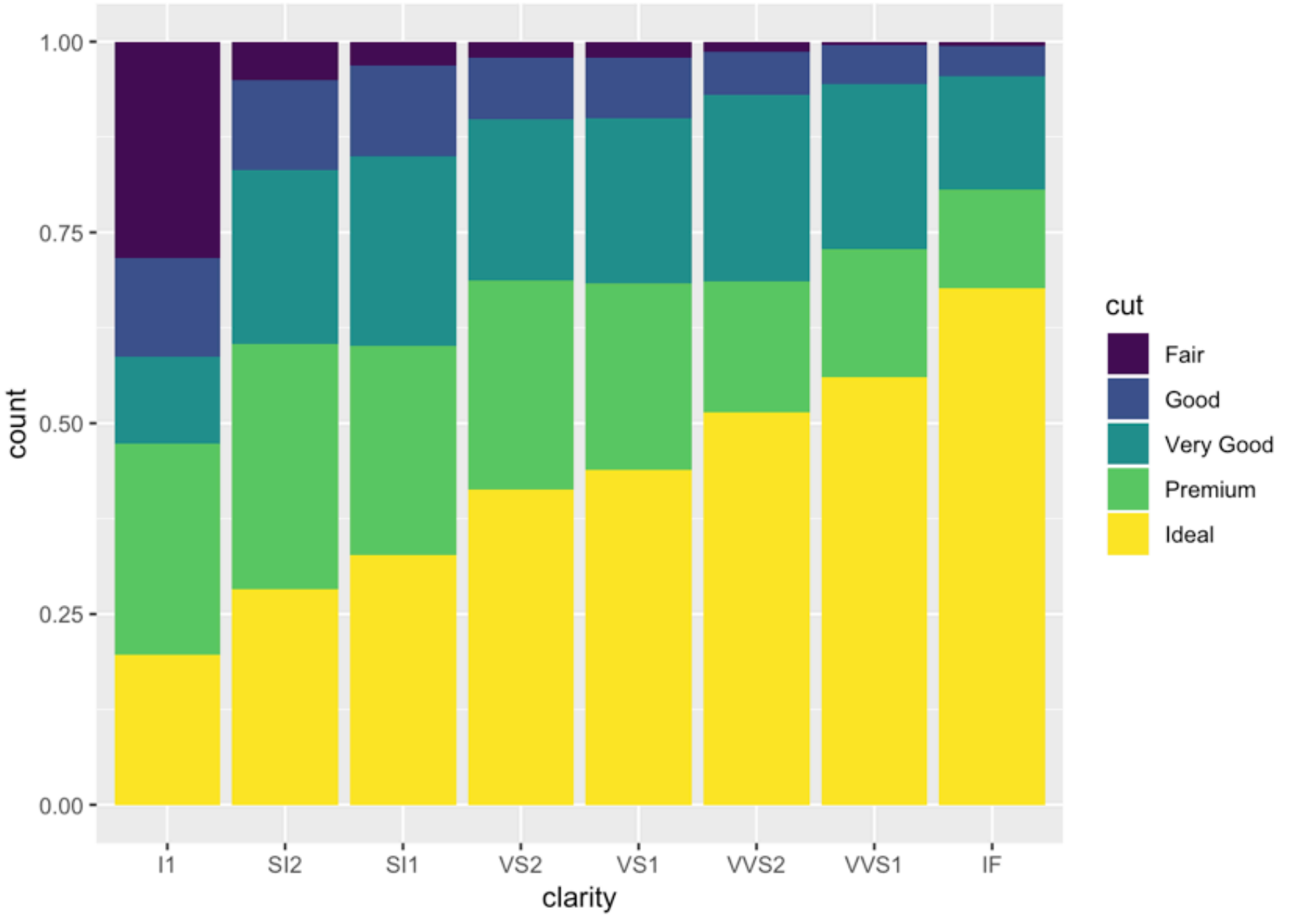
```
#以某次數統期中考成績為例進行分析
#畫出成績分布的直方圖
score <- data.frame(分數區間=c("0~9", "10~19", "20~29","30~39","40~49","50~59","60~69","70~79",
"80~89","90以上"), 人數=c(0,1,1,4,7,7,8,13,1,3))
pp<-ggplot(data=score, aes(x=分數區間, y=人數,fill=分數區間),family="黑體-繁 中黑") +geom_bar(stat="identity") +theme(text=element_text(family="黑體-繁 中黑", size=12)) +labs(title="Bar Chart-各分數區間之人數", x="分數區間", y="人數")
pp
```




```
#以內建的dataset "diamonds"為例
#並列直方圖
 p1<-ggplot(diamonds, aes(clarity, fill=cut)) + geom_bar(position="dodge")
p1
```




```
#以內建的dataset "diamonds"為例
#堆疊直方圖
 p2<-ggplot(diamonds, aes(clarity, fill=cut)) + geom_bar(position="fill")
p2
```



二、分配

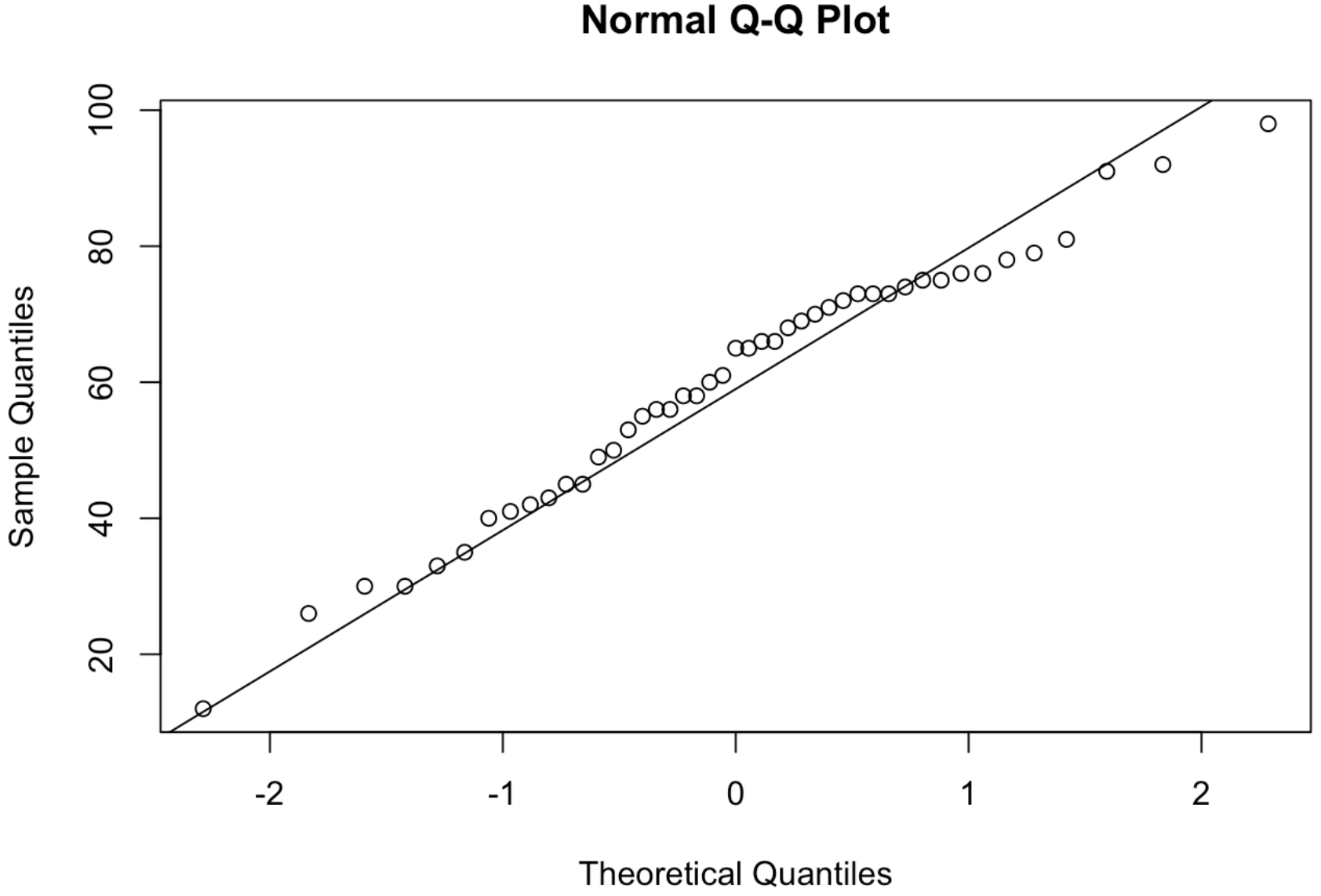
1. 常態檢定

```
#同樣取某次數統期中考成績，猜測其為何者分配
#猜測為常態，進行shapiro.test
grade<- read.csv("grade.csv",fileEncoding = "big5")
shapiro.test(grade$第一次期中考)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  grade$第一次期中考
## W = 0.97569, p-value = 0.4567
```

```
#p-value=0.4567，大於0.05，不拒絕虛無假設，資料為常態
```

```
#觀察qqplot
qqnorm(grade$第一次期中考)
qqline(grade$第一次期中考)
```




```
#分佈為常態
```

三、檢定

1. 單組樣本檢定

(a)平均值檢定－student t test

```
x = rnorm(100, mean=7, sd=5)
x
```

```
##   [1] 11.7164861 -2.6917843  3.4942224  3.4109055  2.0809861  5.3729268
##   [7]  1.9462714  8.0479425  4.5293100 14.7123037 11.7359573 10.7435717
##  [13]  1.5825996  8.6131329  4.4107578  5.9155051 -1.2902184 -0.6165122
##  [19]  5.5187184  8.0080936  1.7570976  5.5163757  4.4190956  1.7070402
##  [25]  9.1272204  8.3115432 10.1421440 12.7483046 14.2985122  9.1723196
##  [31] 11.3008476  1.9030646  4.0504307  2.2252794  7.4951309 13.1911549
##  [37]  8.8199698  1.0334012  9.4034941  8.3120655  7.0095016  9.9106562
##  [43] -1.6701468  4.0466713  8.5062583 10.6007837  5.6962846  6.5202432
##  [49]  9.6312284  3.9947928 11.1033672  3.2185887  4.0568771  3.9517716
##  [55] -0.5406143  2.4940272  5.3275316 11.0795598  3.7608902 14.4732761
##  [61]  7.1567201 12.7079903  0.7349331  7.6330714 -0.4634109  8.4052107
##  [67] 13.2219417  5.2445121  4.0822290  6.4227440 -6.2477534  2.4660698
##  [73]  2.4129829 10.5990510  5.8586444  2.7402463  5.2685322 10.3220955
##  [79]  5.8133482 18.6041287  8.2096949  4.2919996  6.7793231 -1.1635291
##  [85] 12.9510391  4.9260399 12.0146516  3.9548343  7.8830528  3.9237658
##  [91]  6.3821908  3.0918955  9.3309084  6.4805206 14.5742451  7.1363955
##  [97] 16.8893843  7.8836064 10.9905043  5.7212463
```

```
mean(x)
```

```
## [1] 6.505823
```

```
sd(x)
```

```
## [1] 4.547329
```

```
t.test(x,alternative= "two.sided", mu=6)
```

```
##
##  One Sample t-test
##
## data:  x
## t = 1.1124, df = 99, p-value = 0.2687
## alternative hypothesis: true mean is not equal to 6
## 95 percent confidence interval:
##  5.603534 7.408111
## sample estimates:
## mean of x
##  6.505823
```

```
#p-value 小於0.05, 拒絕虛無假設
```

(b)比例檢定－常態Z test

```
prop.test(25, 100, correct=T, p=0.3)
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  25 out of 100, null probability 0.3
## X-squared = 0.96429, df = 1, p-value = 0.3261
## alternative hypothesis: true p is not equal to 0.3
## 95 percent confidence interval:
##  0.1711755 0.3483841
## sample estimates:
##    p
## 0.25
```

```
#p-value 大於0.05,不拒絕虛無假設
```

(c)中位數檢定—Wilcoxon Sign-Rank test(無母數方法)

```
wilcox.test(x, mu=6)
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  x
## V = 2817, p-value = 0.3162
## alternative hypothesis: true location is not equal to 6
```

```
#p-value 小於0.05,拒絕虛無假設
```

2. 兩組樣本的檢定

(a)比較兩平均數的差值—Two Sample t-test

```
#同樣取某次數統成績進行分析，想知道成績與年級是否相關
#H0：三年級與四年級的平均成績相等
#Ha：三年級與四年級的平均成績不相等
t.test(grade$第一次期中考~grade$年級,var.equal=TRUE)
```

```
##
##  Two Sample t-test
##
## data:  grade$第一次期中考 by grade$年級
## t = 0.12242, df = 43, p-value = 0.9031
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -10.83108  12.23108
## sample estimates:
## mean in group 3 mean in group 4
##            60.4            59.7
```

```
#p-value 大於0.05，不拒絕虛無假設，因此成績與年級無關
```

(b)比較兩變異數的差異

```
x=rnorm(100, mean=5.0, sd=3)
y=rnorm(100, mean=5.5, sd=3)
var.test(x,y)
```

```
##
##  F test to compare two variances
##
## data:  x and y
## F = 0.83481, num df = 99, denom df = 99, p-value = 0.3706
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.5616941 1.2407209
## sample estimates:
## ratio of variances
##          0.8348087
```

#p-value 小於0.05，拒絕虛無假設，兩者變異數不相等

(c)比較兩組比例的差異

#設有一筆資料顯示共有56處導致地下水與土地污染，其中24處為工廠，25處為加油站，而我們想檢定工廠所佔比例是否等於加油站所佔比例

```
prop.test(c(24, 25), c(56, 56), alternative = "two.sided", conf.level = 0.95)
```

```
##
##  2-sample test for equality of proportions with continuity
##  correction
##
## data:  c(24, 25) out of c(56, 56)
## X-squared = 0, df = 1, p-value = 1
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.2194311  0.1837169
## sample estimates:
##    prop 1    prop 2
## 0.4285714 0.4464286
```

#p-value大於0.999，不拒絕虛無假設，兩者比例相等

(d)比較兩組中位數的差異（無母數方法）

```
x=rnorm(100, mean=5.0, sd=3)
y=rnorm(100, mean=5.5, sd=3)
wilcox.test(x,y,exact=F,correst=F)
```

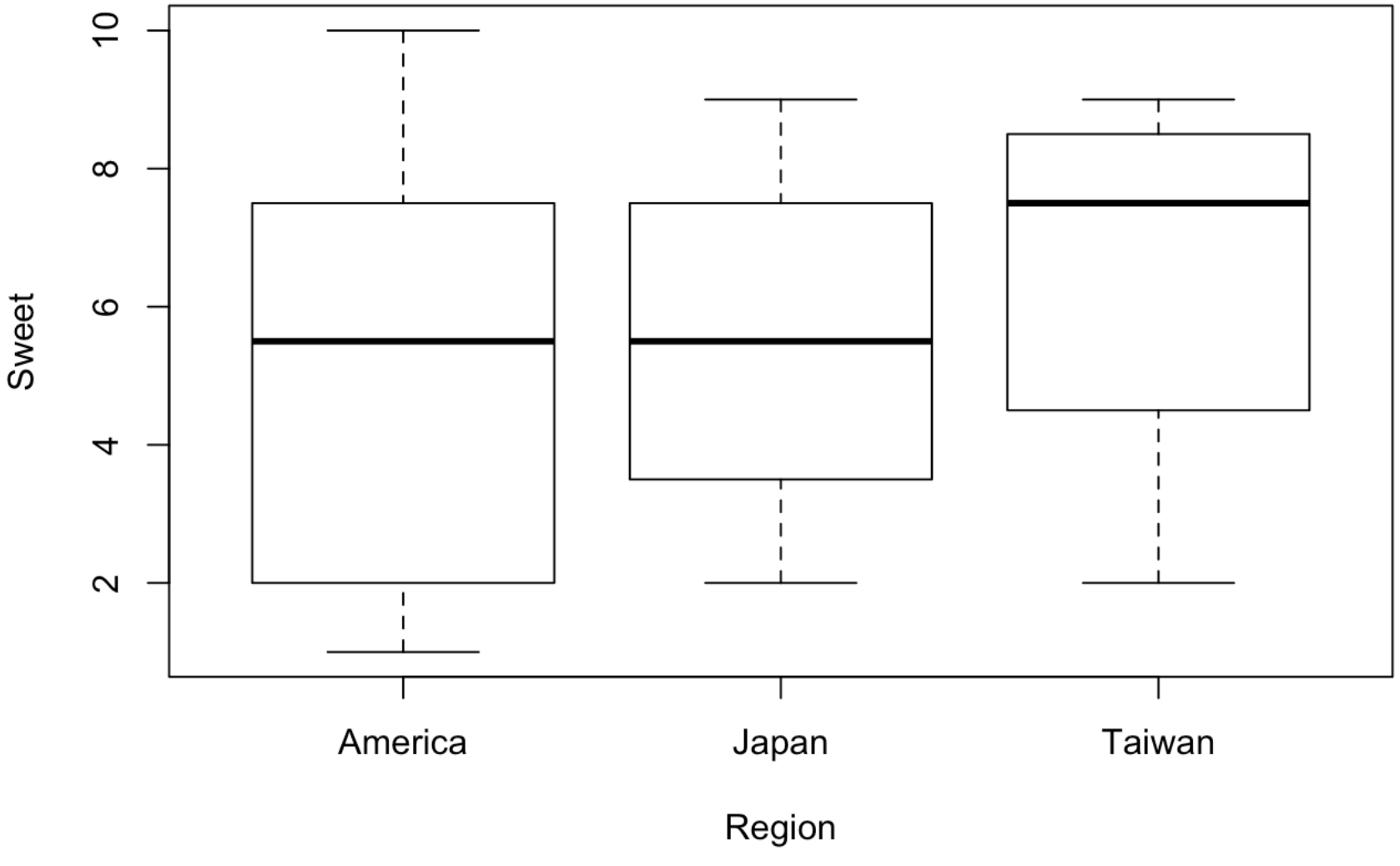
```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  x and y
## W = 4366, p-value = 0.1216
## alternative hypothesis: true location shift is not equal to 0
```

```
#p-value 大於0.05，不拒絕虛無假設，兩者中位數相等
```

3. 三組以上的樣本檢定

(a)ANOVA分析

```
#假設想知道日本進口(Japan)、美國進口(America)與台灣本地(Taiwan)的蘋果甜度是否有差異
#輸入data
Sweet=c(3,5,8,6,4,9,2,7,1,5,10,6,7,8,3,1,6,9,9,8,7,3,2,8)
Region=c(rep("Japan",8),rep("America",8),rep("Taiwan",8))
SweetTest=data.frame(Sweet,Region)
plot(Sweet~Region,data=SweetTest)
```




```
result=aov(Sweet~Region,data=SweetTest)
summary(result)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Region       2   8.08   4.042   0.509  0.609
## Residuals   21 166.88   7.946
```

```
#p-value大於0.05,不拒絕虛無假設，三地區的蘋果甜度沒有顯著差異。
```

四、分析

1. 迴歸分析—線性迴歸

```
#以一筆聯合國World Happiness Report的資料為例，僅挑選2018年的資料進行分析，欲從這些指標中挑選出影響人民幸福指數的變數
#資料共136筆
#變數解釋
#Log GDP per capita：根據世界銀行WDI資料庫而得的人均GDP
#Social support：社會支持
#Healthy life expectancy at birth：零歲時健康平均餘命
#Freedom to make life choices：自由程度
#Generosity：人民慷慨程度
#Perceptions of corruption：對政府貪腐程度的看法
#Positive affect：正面情緒量表
#Negative affect：負面情緒量表
#Confidence in national government：對政府的信任程度

#匯入資料
happiness<- read.csv("2018world.csv", header=T)
#敘述統計
summary(happiness)
```

```
##       Country.name Log.GDP.per.capita Social.support
##  Afghanistan:  1   Min.   : 6.541     Min.   :0.4847
##  Albania    :  1   1st Qu.: 8.346     1st Qu.:0.7397
##  Algeria    :  1   Median : 9.416     Median :0.8366
##  Argentina  :  1   Mean   : 9.250     Mean   :0.8105
##  Armenia    :  1   3rd Qu.:10.167     3rd Qu.:0.9056
##  Australia  :  1   Max.   :11.454     Max.   :0.9845
##  (Other)    :130   NA's   :9
##  Healthy.life.expectancy.at.birth Freedom.to.make.life.choices
##  Min.   :48.20                    Min.   :0.3735
##  1st Qu.:59.08                    1st Qu.:0.7182
##  Median :66.35                    Median :0.7956
##  Mean   :64.67                    Mean   :0.7845
##  3rd Qu.:69.08                    3rd Qu.:0.8770
##  Max.   :76.80                    Max.   :0.9699
##  NA's   :4
##    Generosity       Perceptions.of.corruption Positive.affect
##  Min.   :-0.33638   Min.   :0.09656           Min.   :0.4241
##  1st Qu.:-0.15049   1st Qu.:0.69107           1st Qu.:0.6393
##  Median :-0.03820   Median :0.79309           Median :0.7353
##  Mean   :-0.02909   Mean   :0.73174           Mean   :0.7096
##  3rd Qu.: 0.06307   3rd Qu.:0.85138           3rd Qu.:0.7940
##  Max.   : 0.49938   Max.   :0.95201           Max.   :0.8836
##  NA's   :10         NA's   :7                 NA's   :1
##  Negative.affect  Confidence.in.national.government Happiness.score
##  Min.   :0.0927   Min.   :0.07971                  Min.   :3.203
##  1st Qu.:0.2191   1st Qu.:0.33120                  1st Qu.:4.556
##  Median :0.2874   Median :0.46884                  Median :5.356
##  Mean   :0.2937   Mean   :0.49512                  Mean   :5.441
##  3rd Qu.:0.3600   3rd Qu.:0.62847                  3rd Qu.:6.204
##  Max.   :0.5438   Max.   :0.98812                  Max.   :7.769
##  NA's   :1        NA's   :13
##     X             X.1
##  Mode:logical   Mode:logical
##  NA's:136       NA's:136
##
##
##
##
##
```

```
#建立模型–放入感興趣的變數
lm.fitall<-lm(Happiness.score~happiness$Log.GDP.per.capita+happiness$Freedom.to.ma
ke.life.choices+happiness$Positive.affect,data=happiness)
summary(lm.fitall)
```

```
##
## Call:
## lm(formula = Happiness.score ~ happiness$Log.GDP.per.capita +
##     happiness$Freedom.to.make.life.choices + happiness$Positive.affect,
##     data = happiness)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3035 -0.3386  0.1038  0.4396  1.0955
##
## Coefficients:
##                                       Estimate Std. Error t value
## (Intercept)                           -3.02573    0.50491  -5.993
## happiness$Log.GDP.per.capita           0.66494    0.04751  13.995
## happiness$Freedom.to.make.life.choices  1.43874    0.60442   2.380
## happiness$Positive.affect              1.65025    0.63156   2.613
##                                       Pr(>|t|)
## (Intercept)                           2.15e-08 ***
## happiness$Log.GDP.per.capita           < 2e-16 ***
## happiness$Freedom.to.make.life.choices   0.0188 *
## happiness$Positive.affect                0.0101 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6005 on 122 degrees of freedom
##   (10 observations deleted due to missingness)
## Multiple R-squared:  0.7135, Adjusted R-squared:  0.7064
## F-statistic: 101.3 on 3 and 122 DF,  p-value: < 2.2e-16
```

```
#篩選變數
#forward：從空集合開始，每次進入p-value最小的解釋變數，直到未進入的解釋變數都不顯著
step(lm.fitall,direction= "forward")
```

```
## Start:  AIC=-124.57
## Happiness.score ~ happiness$Log.GDP.per.capita + happiness$Freedom.to.make.life
.choices +
##     happiness$Positive.affect
```

```
##
## Call:
## lm(formula = Happiness.score ~ happiness$Log.GDP.per.capita +
##     happiness$Freedom.to.make.life.choices + happiness$Positive.affect,
##     data = happiness)
##
## Coefficients:
##                                 (Intercept)
##                                     -3.0257
##           happiness$Log.GDP.per.capita
##                                      0.6649
## happiness$Freedom.to.make.life.choices
##                                      1.4387
##              happiness$Positive.affect
##                                      1.6503
```

```
#backward：從包含所有變數的模型開始，每次剔除p-value最大的解釋變數，直到保留的解釋變數都顯著
step(lm.fitall,direction= "backward")
```

```
## Start:  AIC=-124.57
## Happiness.score ~ happiness$Log.GDP.per.capita + happiness$Freedom.to.make.life
.choices +
##     happiness$Positive.affect
##
##                                          Df Sum of Sq     RSS      AIC
## <none>                                                  43.997 -124.572
## - happiness$Freedom.to.make.life.choices  1     2.043  46.040 -120.852
## - happiness$Positive.affect               1     2.462  46.459 -119.711
## - happiness$Log.GDP.per.capita            1    70.634 114.631   -5.915
```

```
##
## Call:
## lm(formula = Happiness.score ~ happiness$Log.GDP.per.capita +
##     happiness$Freedom.to.make.life.choices + happiness$Positive.affect,
##     data = happiness)
##
## Coefficients:
##                                 (Intercept)
##                                     -3.0257
##           happiness$Log.GDP.per.capita
##                                      0.6649
## happiness$Freedom.to.make.life.choices
##                                      1.4387
##              happiness$Positive.affect
##                                      1.6503
```

```
#forward和backward的改良結合版
step(lm.fitall,direction= "both")
```

```
## Start:  AIC=-124.57
## Happiness.score ~ happiness$Log.GDP.per.capita + happiness$Freedom.to.make.life
.choices +
##     happiness$Positive.affect
##
##                                          Df Sum of Sq     RSS      AIC
## <none>                                                43.997 -124.572
## - happiness$Freedom.to.make.life.choices  1     2.043  46.040 -120.852
## - happiness$Positive.affect               1     2.462  46.459 -119.711
## - happiness$Log.GDP.per.capita            1    70.634 114.631   -5.915
```

```
##
## Call:
## lm(formula = Happiness.score ~ happiness$Log.GDP.per.capita +
##     happiness$Freedom.to.make.life.choices + happiness$Positive.affect,
##     data = happiness)
##
## Coefficients:
##                             (Intercept)
##                                 -3.0257
##           happiness$Log.GDP.per.capita
##                                  0.6649
## happiness$Freedom.to.make.life.choices
##                                  1.4387
##              happiness$Positive.affect
##                                  1.6503
```

2. 多變量分析－主成份分析

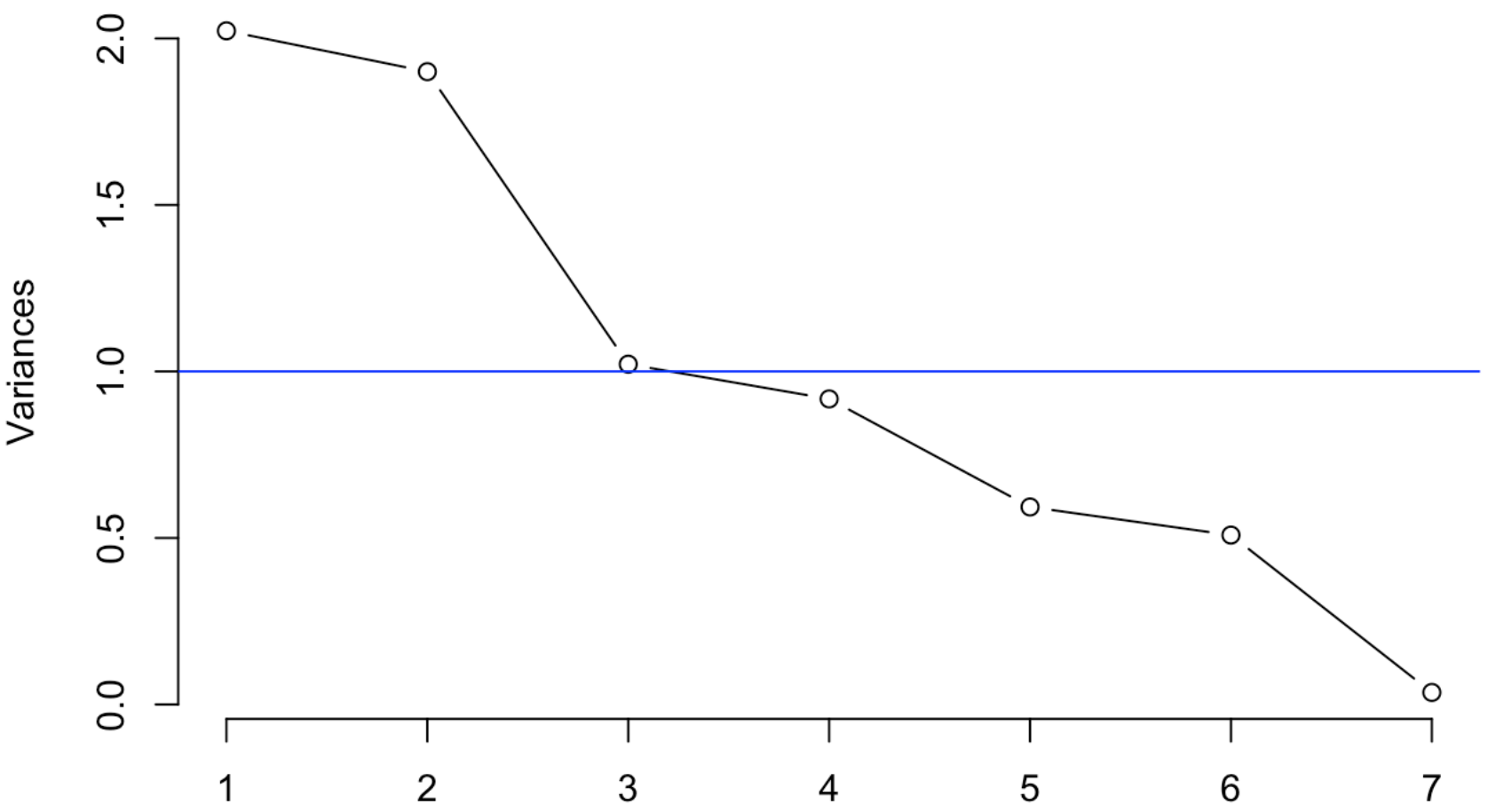
```
#以2012年美國職棒MLB的資料為例，此為網路上的公開資料
#匯入資料
MLB<- read.csv("2012MLB.csv",header=T,sep=",")
head(MLB)
```

```
##                   Team   G   R    H H1B H2B H3B  HR RBI  BB   SO  SB   AVG
## 1        Texas Rangers 152 764 1444 941 283  32 188 738 448 1030  89 0.275
## 2   Los Angeles Angels 153 726 1424 972 252  20 180 691 423 1041 125 0.273
## 3     Colorado Rockies 152 718 1425 932 286  49 158 680 432 1129  96 0.272
## 4  St. Louis Cardinals 153 723 1447 983 277  35 152 691 503 1128  89 0.272
## 5 San Francisco Giants 153 683 1419 997 273  54  95 642 453 1032 111 0.270
## 6       Detroit Tigers 152 689 1379 922 267  37 153 663 490 1042  53 0.268
##     OBP
## 1 0.335
## 2 0.331
## 3 0.329
## 4 0.338
## 5 0.327
## 6 0.336
```

```r
#進行主成份分析
pca <- prcomp(formula = ~ H1B+H2B+H3B+HR+RBI+SB+BB,  #選擇七個變數
              data = MLB,                           # 資料
              scale = TRUE)                          # 正規化資料
pca
```

```
## Standard deviations (1, .., p=7):
## [1] 1.4222856 1.3785035 1.0108522 0.9578441 0.7700729 0.7131148 0.1897347
##
## Rotation (n x k) = (7 x 7):
##              PC1        PC2          PC3           PC4          PC5
## H1B -0.40991503 -0.4681242  0.07174689  0.056704066 -0.07882016
## H2B -0.51441491 -0.2004156  0.01669591  0.255448162 -0.46809834
## H3B  0.01853759 -0.5595940 -0.19427151 -0.004051477  0.71490431
## HR  -0.34336124  0.5417488 -0.03416307 -0.394140194  0.26396281
## RBI -0.64629912  0.1016251 -0.25396353 -0.156840299  0.20084751
## SB   0.16722000 -0.2741655 -0.52853255 -0.679207860 -0.39181790
## BB   0.05866272  0.2203673 -0.78218985  0.538744635 -0.00676713
##              PC6         PC7
## H1B -0.66701643 -0.39159028
## H2B  0.62315846 -0.14911690
## H3B  0.34921449 -0.12535585
## HR   0.10991843 -0.59189466
## RBI -0.12239863  0.65387482
## SB   0.04037564 -0.03239357
## BB  -0.12695740 -0.17252886
```

```r
#該選擇幾個主成份，繪製「陡坡圖」(scree plot)及「累積解釋圖」(Pareto plot)
#使用plot()函式
plot(pca,          # 放pca
     type="line", # 用直線連結每個點
     main="Scree Plot for 2012MLB") # 主標題

#用藍線標示出特徵值=1的地方
abline(h=1, col="blue") # Kaiser eigenvalue-greater-than-one rule
```




```
vars <- (pca$sdev)^2  # 從pca中取出標準差(pca$sdev)後再平方，計算variance(特徵值)
vars
```

```
## [1] 2.02289644 1.90027181 1.02182222 0.91746533 0.59301228 0.50853268
## [7] 0.03599925
```

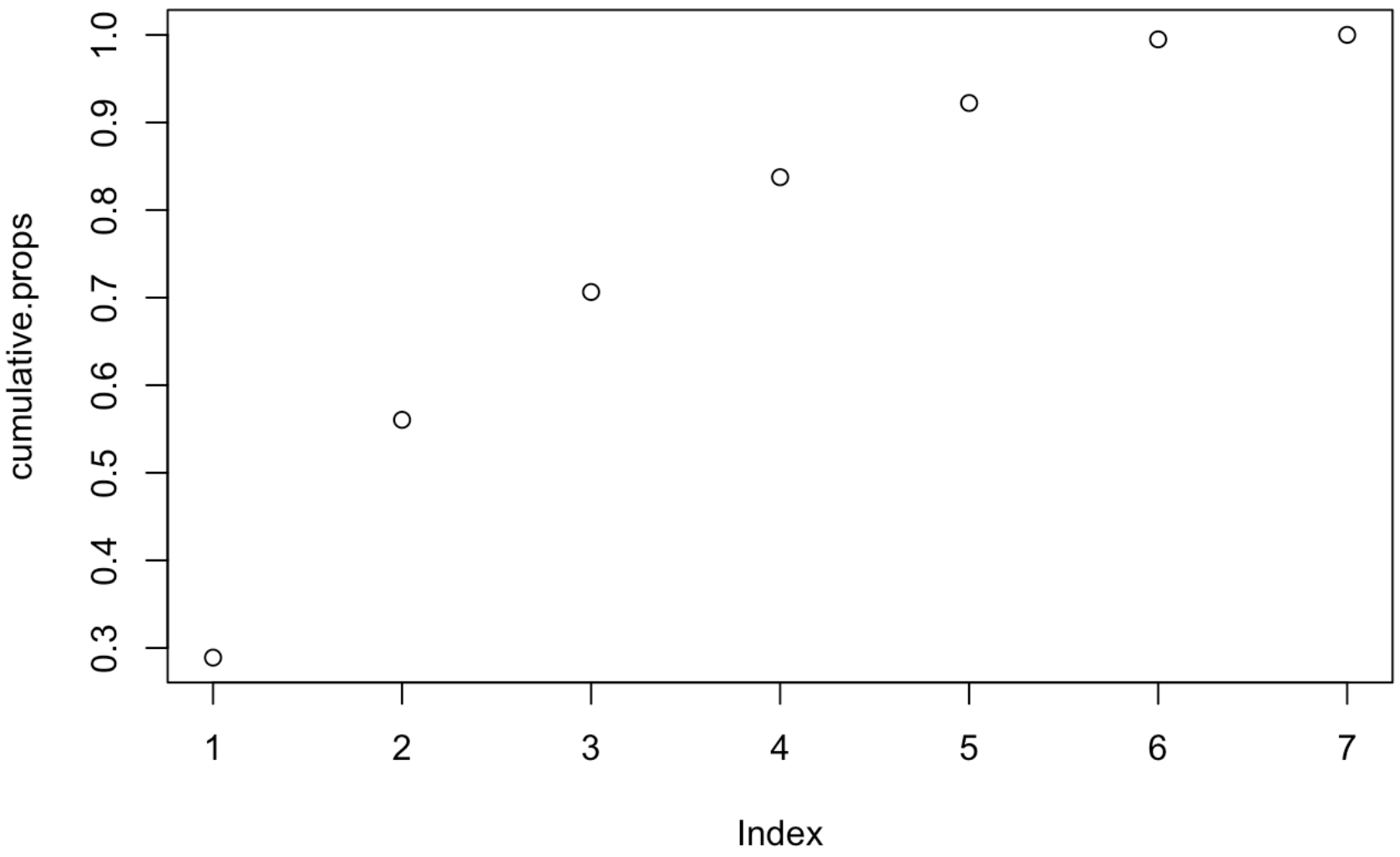
```
#計算每個主成分的解釋比例 = 各個主成分的特徵值/總特徵值
props <- vars / sum(vars)
props
```

```
## [1] 0.288985205 0.271467401 0.145974603 0.131066475 0.084716040 0.072647526
## [7] 0.005142749
```

```
cumulative.props <- cumsum(props)  # 累加前n個元素的值
cumulative.props
```

```
## [1] 0.2889852 0.5604526 0.7064272 0.8374937 0.9222097 0.9948573 1.0000000
```

```
#累積解釋比例圖
plot(cumulative.props)
```




```
#原本的資料經主成份分析後，將其轉換成新的以主成份代替的資料集(pca$x)。
#以下步驟是取前三個主成份，作為新的資料集：
# pca$rotation
top3_pca.data <- pca$x[, 1:3]
top3_pca.data
```

```
##             PC1         PC2         PC3
## 1  -2.65536140  0.04641055  0.05124254
## 2  -1.37712847  0.01360254  0.24495540
## 3  -1.57754875 -1.72554295  0.14807629
## 4  -1.76751032 -0.84074064 -0.75258974
## 5  -0.44097214 -3.66454431 -0.36109275
## 6  -1.08166263 -0.10861369  0.27429500
## 7  -0.45474791 -2.65730709  1.13595923
## 8  -2.46449798  0.03093137  1.32250201
## 9  -0.11216989 -1.29948488 -0.83784413
## 10 -1.01441489  0.55336109  0.51565842
## 11 -1.53519975  3.14421085 -1.02043498
## 12 -1.20736887 -0.28285945 -1.24157398
## 13 -1.06901074  0.22513361 -0.72630319
## 14  0.01063384 -0.25128338  0.55678238
## 15 -0.25896225  1.04428266  0.46364492
## 16 -0.32397454  0.64019528  0.36177440
## 17  0.46087245  0.61660868  0.67675736
## 18  0.41732553  0.71111234 -1.21591003
## 19  1.04120451  0.11986429 -0.63306531
## 20  1.39709979 -0.53060810  0.67003668
## 21  1.64046682 -1.54053971 -1.66879107
## 22 -0.51860317  2.45956726  1.36247833
## 23  1.10286061  0.60454851  1.70393647
## 24  1.91924719 -1.14973539 -0.64690007
## 25  0.84451316  1.55315562  0.06346943
## 26  1.74614534 -0.60978067  1.36484924
## 27  1.23431051  0.91240050 -2.16643033
## 28  2.60386762  0.08178357  0.87401532
## 29  1.00595356  1.50672401 -1.44179251
## 30  2.43463278  0.39714752  0.92229467
```

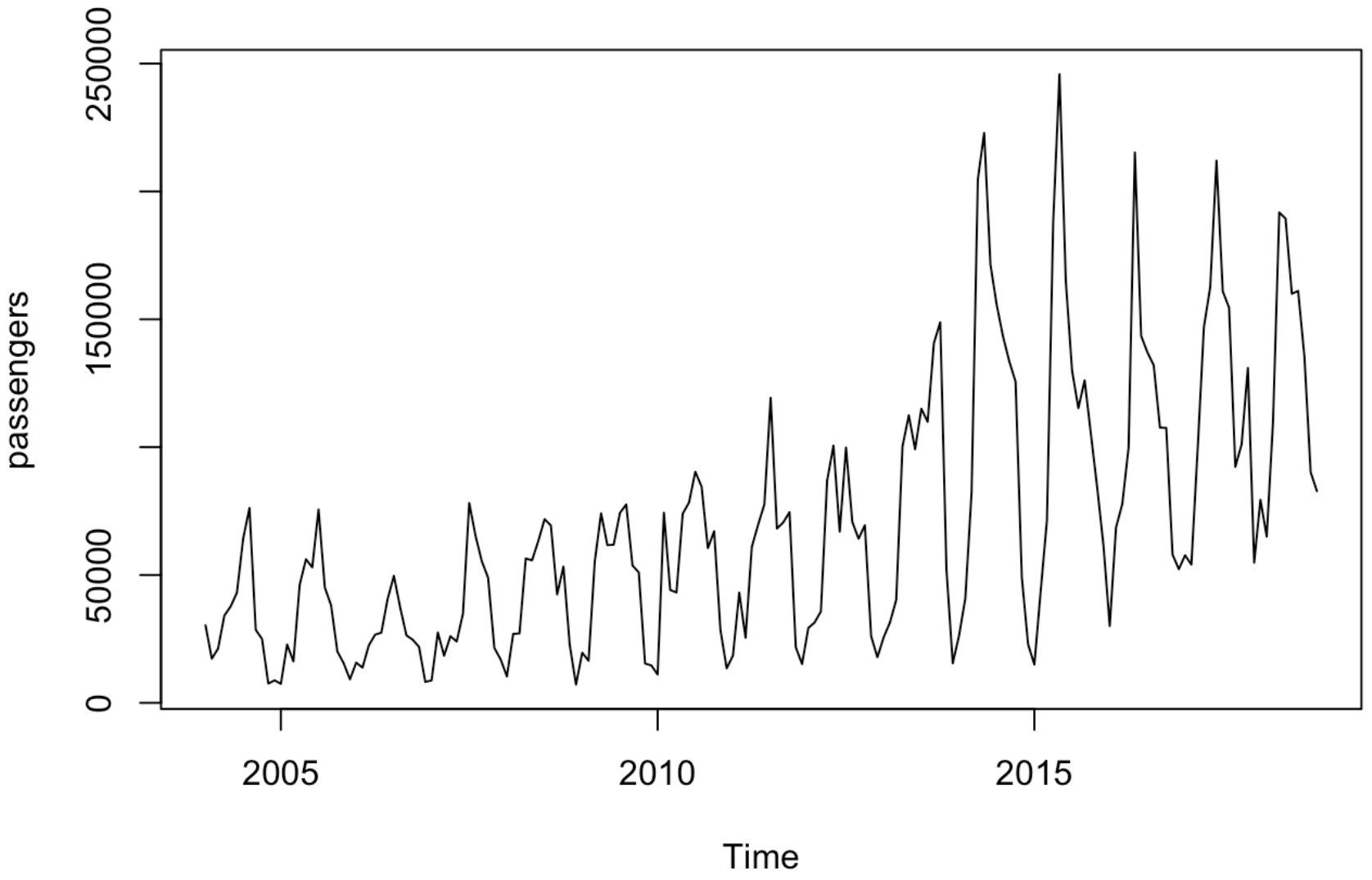
3. 時間序列分析

安裝相關套件 library(TSA)

library(forecast)

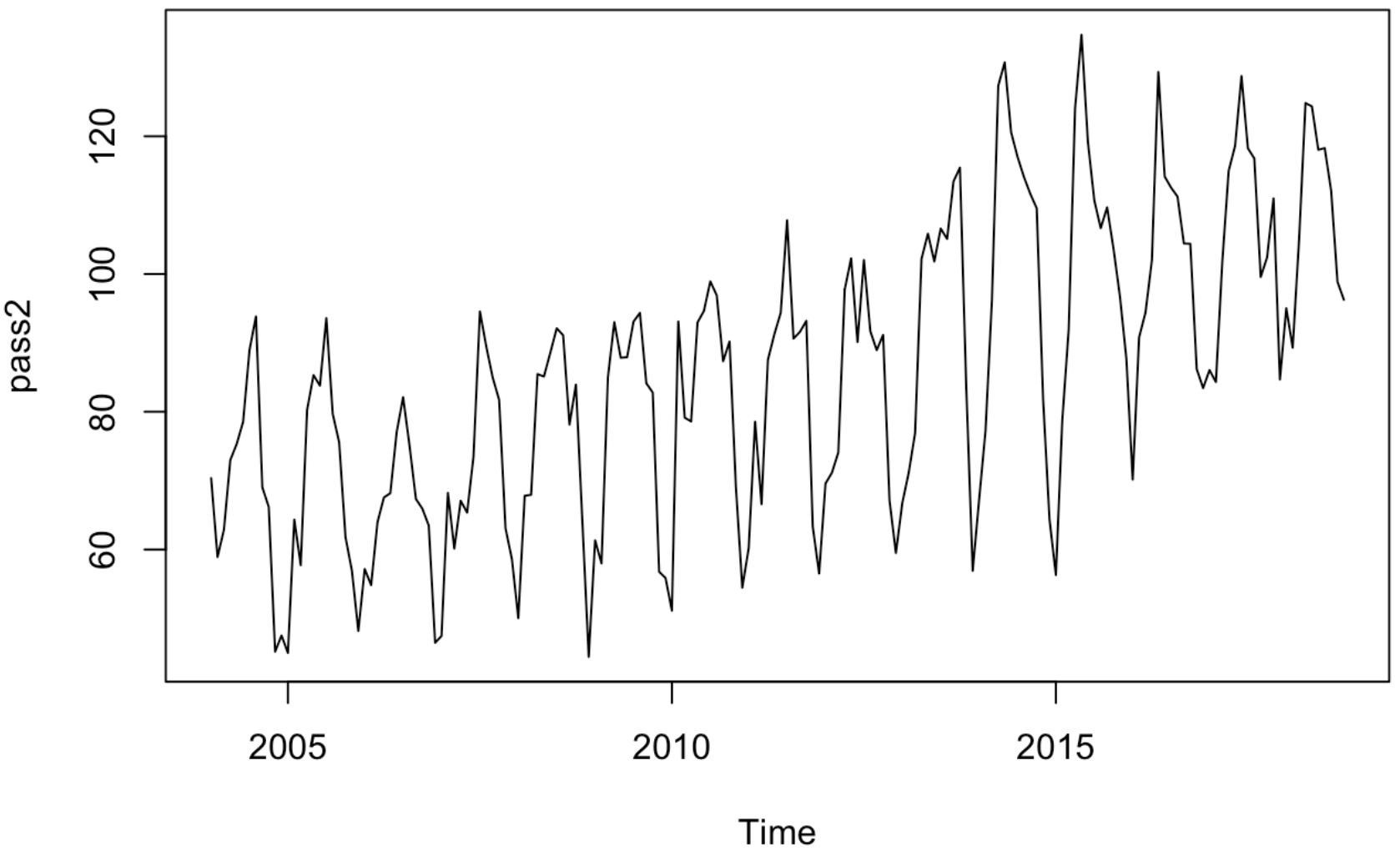
library(tseries)

```
#讀取檔案
data<-read.csv("cruise passengers.csv",fileEncoding="big5")
#畫出時間序列圖
passengers<- ts(data$total, frequency=12, start=c(2004,1))
plot.ts(passengers)
```



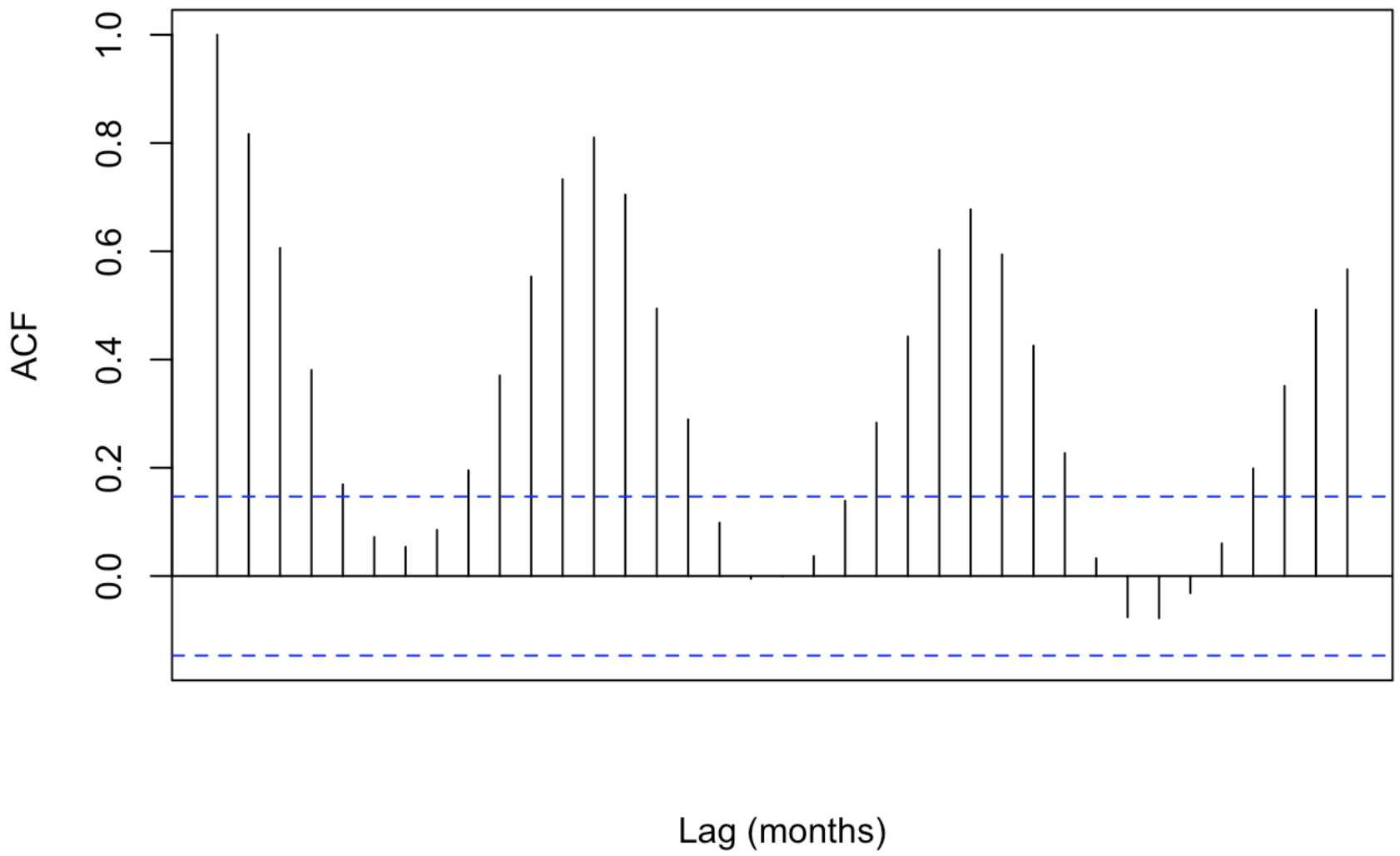

```
#畫出BoxCox
#BoxCox.ar(passengers,lambda = seq(0,0.5,0.01))
#判斷lambda後進行轉換
#lambda=0.3
```

```
#進行轉換
data2<-(((data$total^(0.3))-1)/0.3)
pass2<-ts(data2,frequency=12,start=c(2004,1))
#畫圖觀察趨勢
plot.ts(pass2)
```

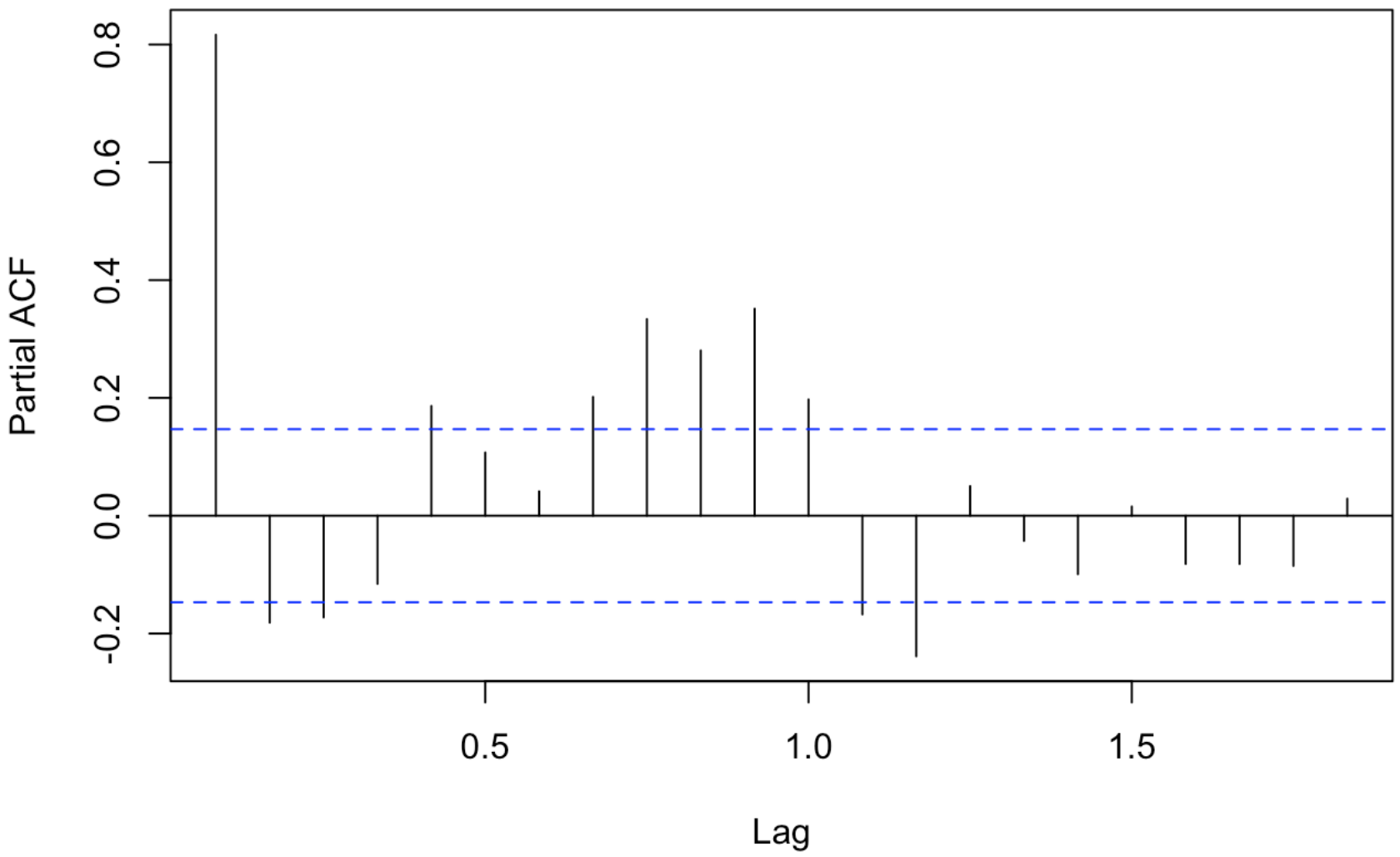



```
#仍有些微的上升趨勢
```

```
#觀察ACF圖與PACF圖
mx=36
acf(pass2,lag.max = mx,xaxt="n",xlab="Lag (months)")
```

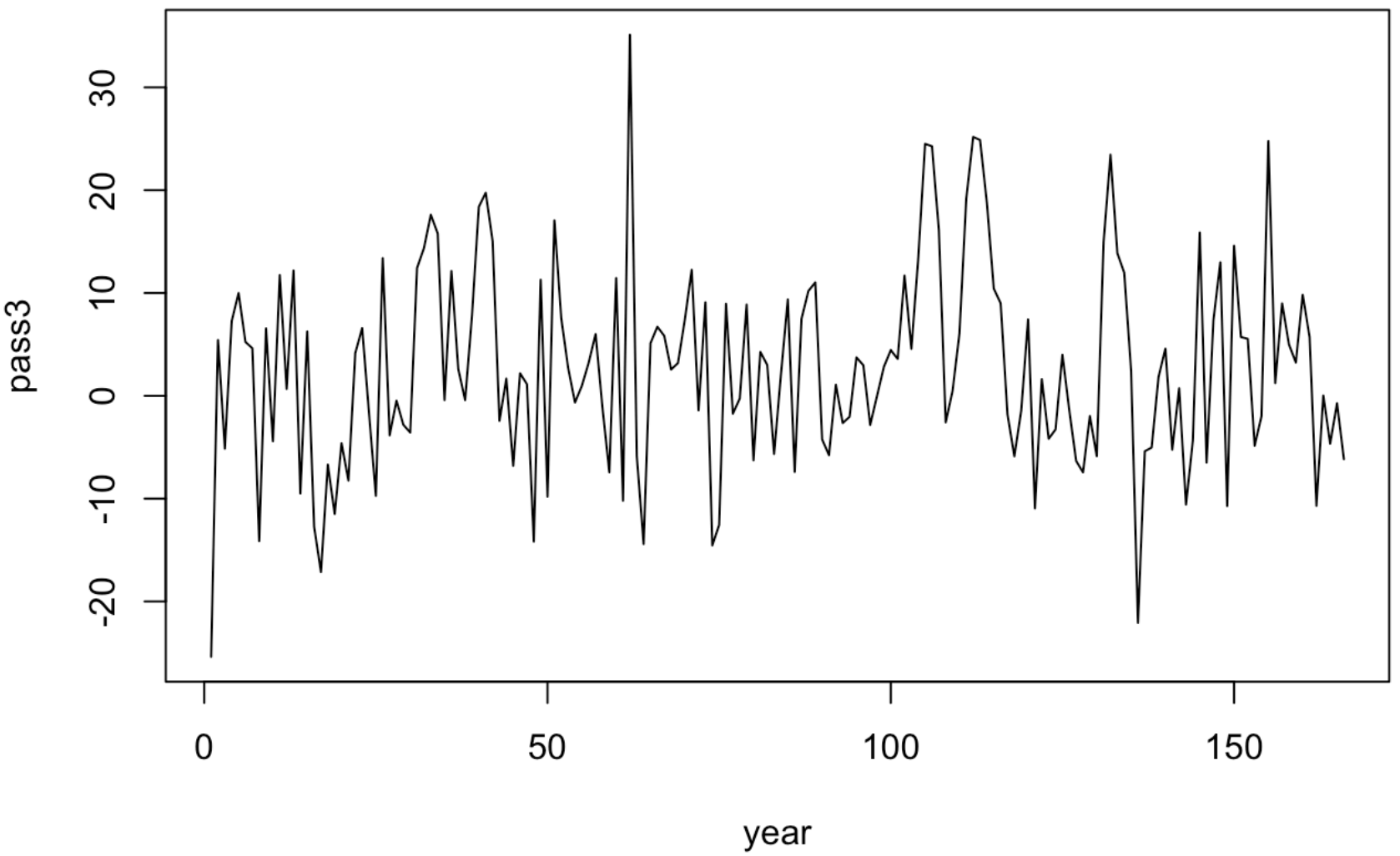



```
pacf(pass2)
```

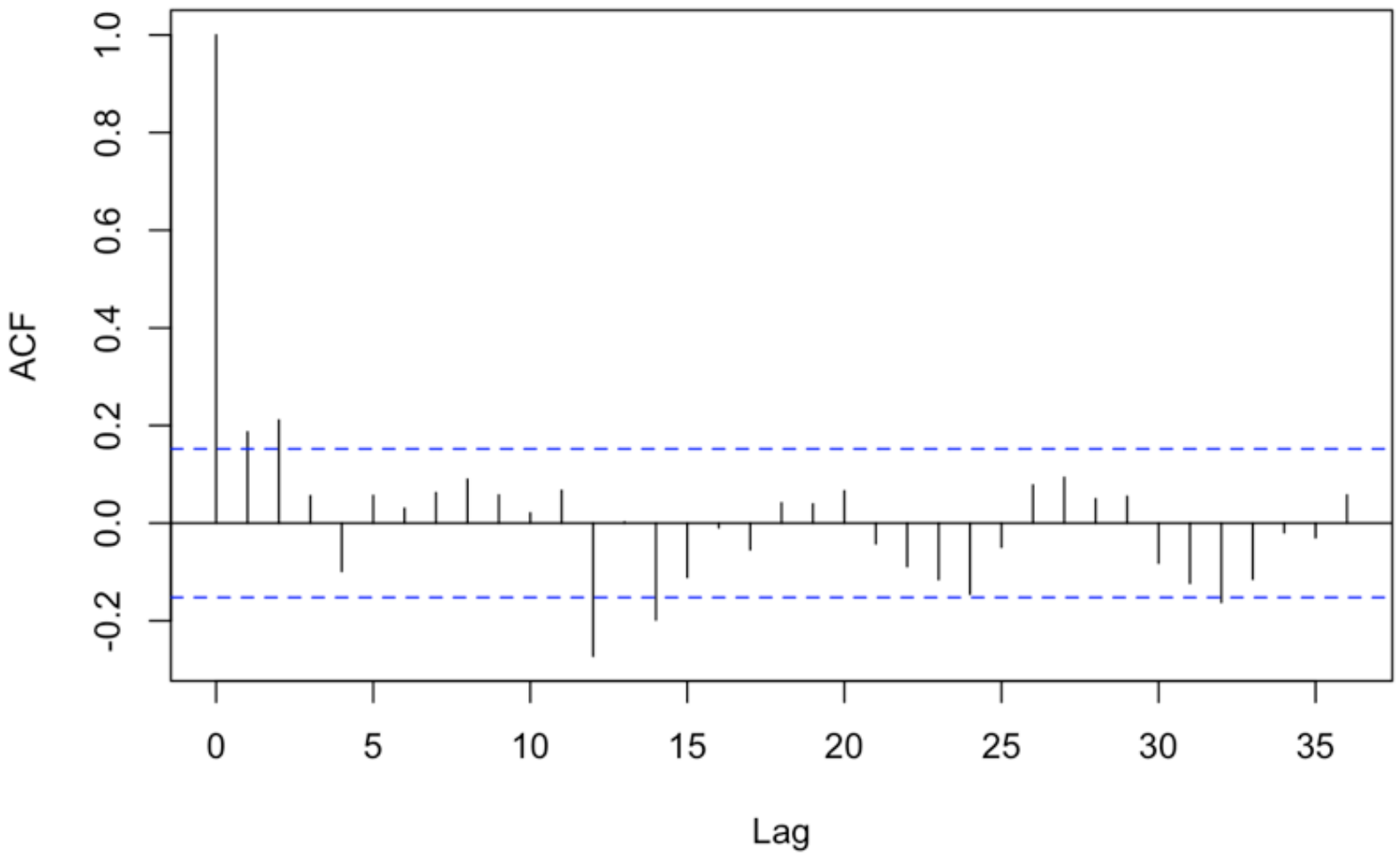



```
#由ACF圖可看出明顯的季節趨勢，週期=12。另外，從ACF圖中看出有拖尾性，考慮做一次差分。
```

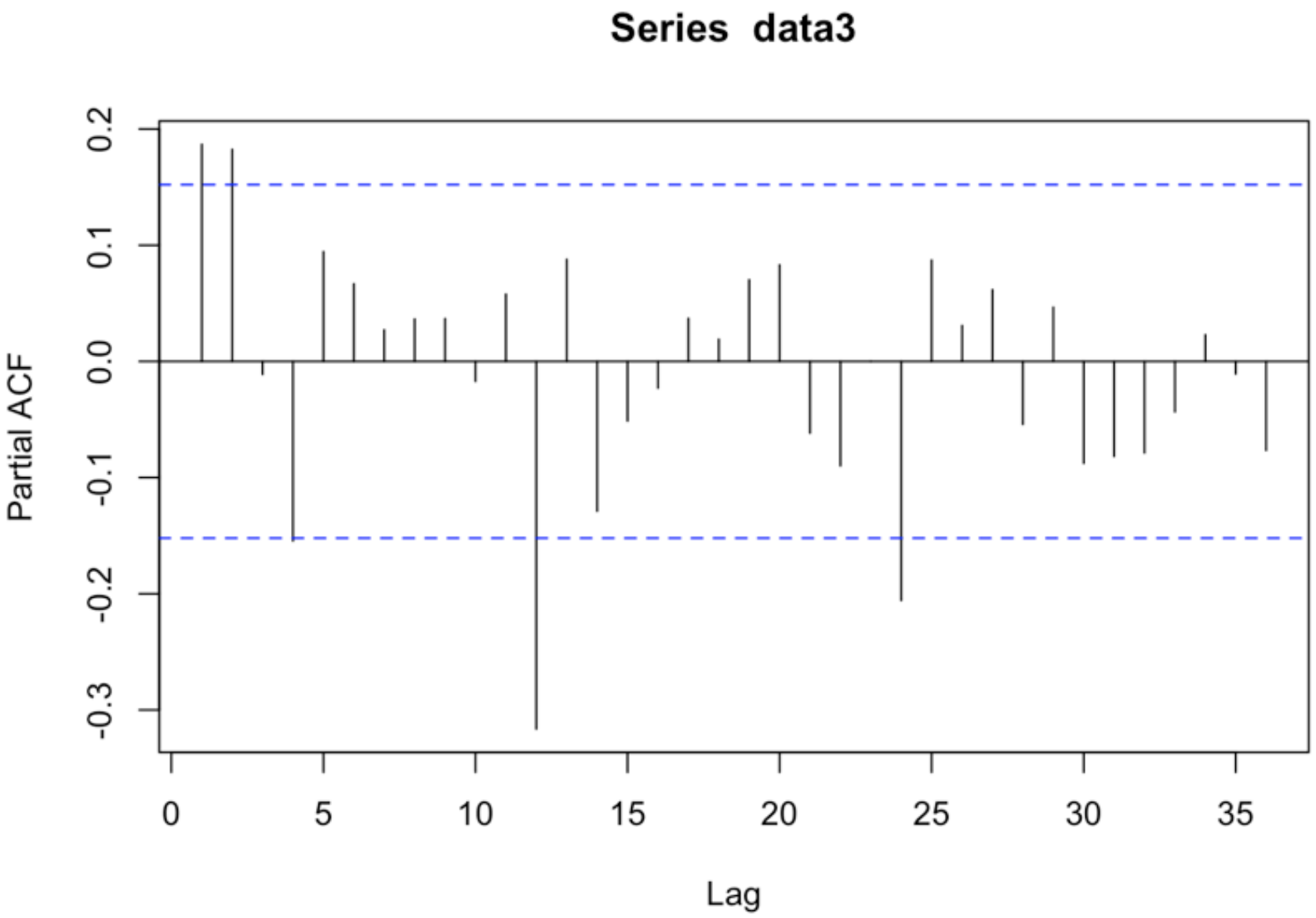
```
#對轉換過的資料做一次季節差分
#D=1
data3=diff(data2,lag=12)
plot.ts(data3,xlab="year",ylab="pass3")
```




```
#時間序列圖大致平穩，無明顯趨勢
#觀察ACF圖與PACF圖
acf(data3,lag.max= 36 )
```

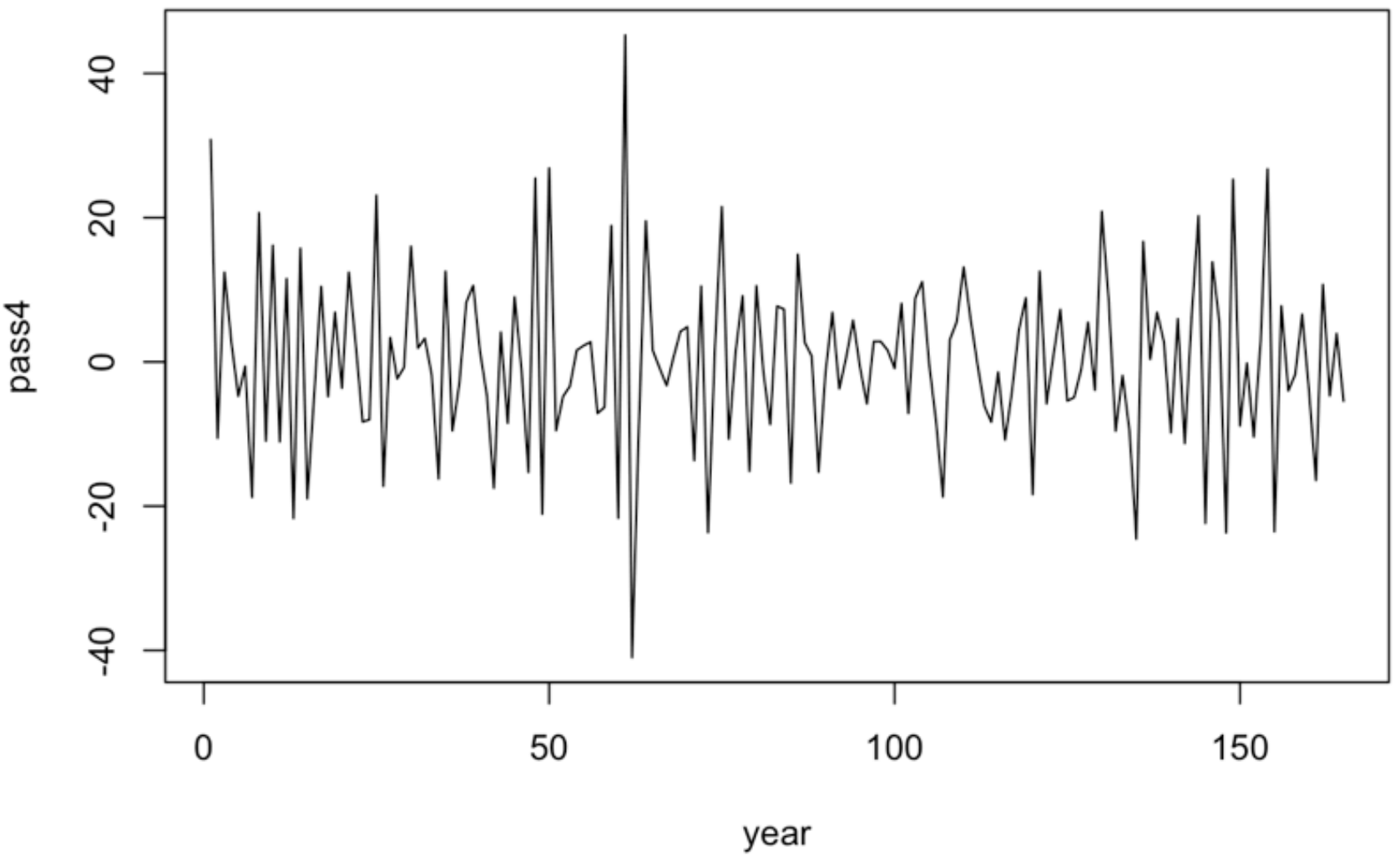



```
pacf(data3,lag.max= 36 )
```

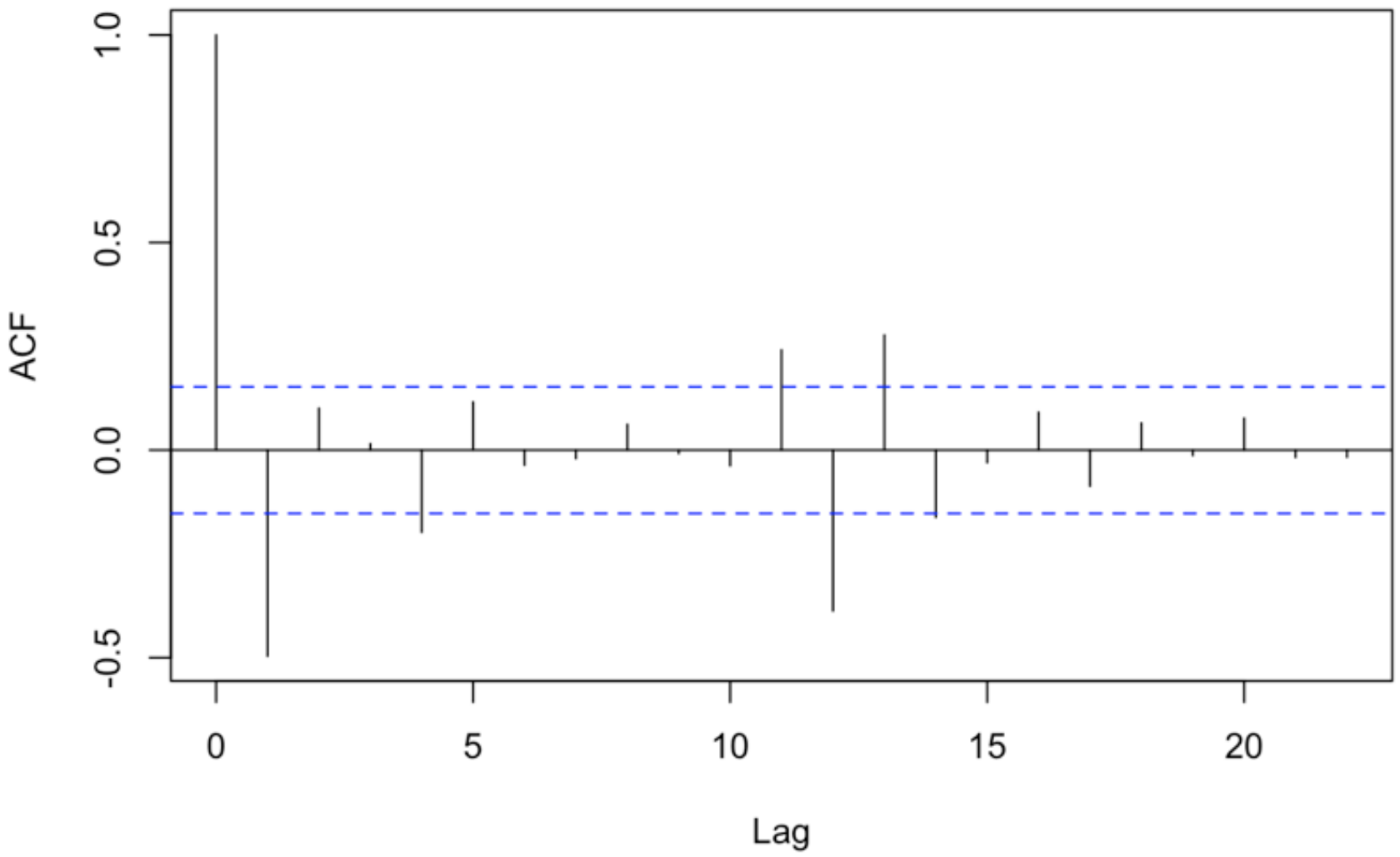



```
#趨於平穩
```

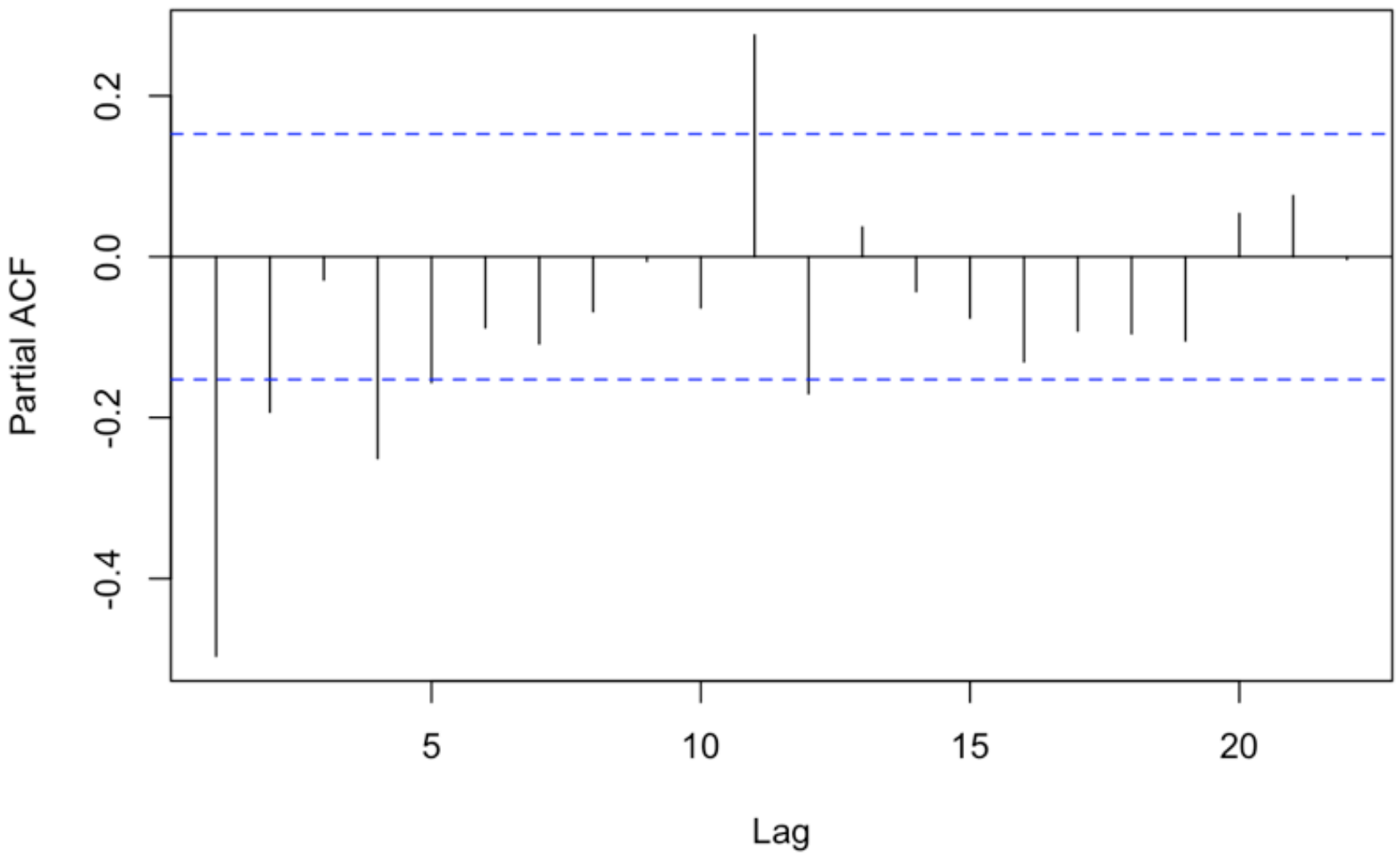
```
#判斷轉換過經一次季節差分的資料是否需要差分，以KPSS 檢定序列的穩定性。
#kpss.test(data3)
#進行一次差分
#D=1 d=1
data4=diff(data3)
#觀察趨勢、ACF圖、PACF圖
plot.ts(data4,xlab="year",ylab="pass4")
```




```
acf(data4)
```




```
pacf(data4)
```




```
#預測模型
model1=arima(passengers,order=c(4,1,2),seasonal=list(order=c(1,1,1),period=12))
model2=arima(passengers,order=c(4,1,4),seasonal=list(order=c(1,1,2),period=12))

#對候選模型進行殘差檢定
t.test(residuals(model1))
```

```
##
##  One Sample t-test
##
## data:  residuals(model1)
## t = 0.525, df = 177, p-value = 0.6002
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -2260.244  3898.726
## sample estimates:
## mean of x
##  819.2409
```

```
shapiro.test(residuals(model1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model1)
## W = 0.94652, p-value = 3.114e-06
```

```
Box.test(residuals(model1), type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  residuals(model1)
## X-squared = 0.00024219, df = 1, p-value = 0.9876
```

```
t.test(residuals(model2))
```

```
##
##  One Sample t-test
##
## data:  residuals(model2)
## t = 0.69054, df = 177, p-value = 0.4908
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -1923.638  3994.465
## sample estimates:
## mean of x
##  1035.413
```

```
shapiro.test(residuals(model2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model2)
## W = 0.93353, p-value = 2.64e-07
```

```
Box.test(residuals(model2), type = "Ljung-Box")
```

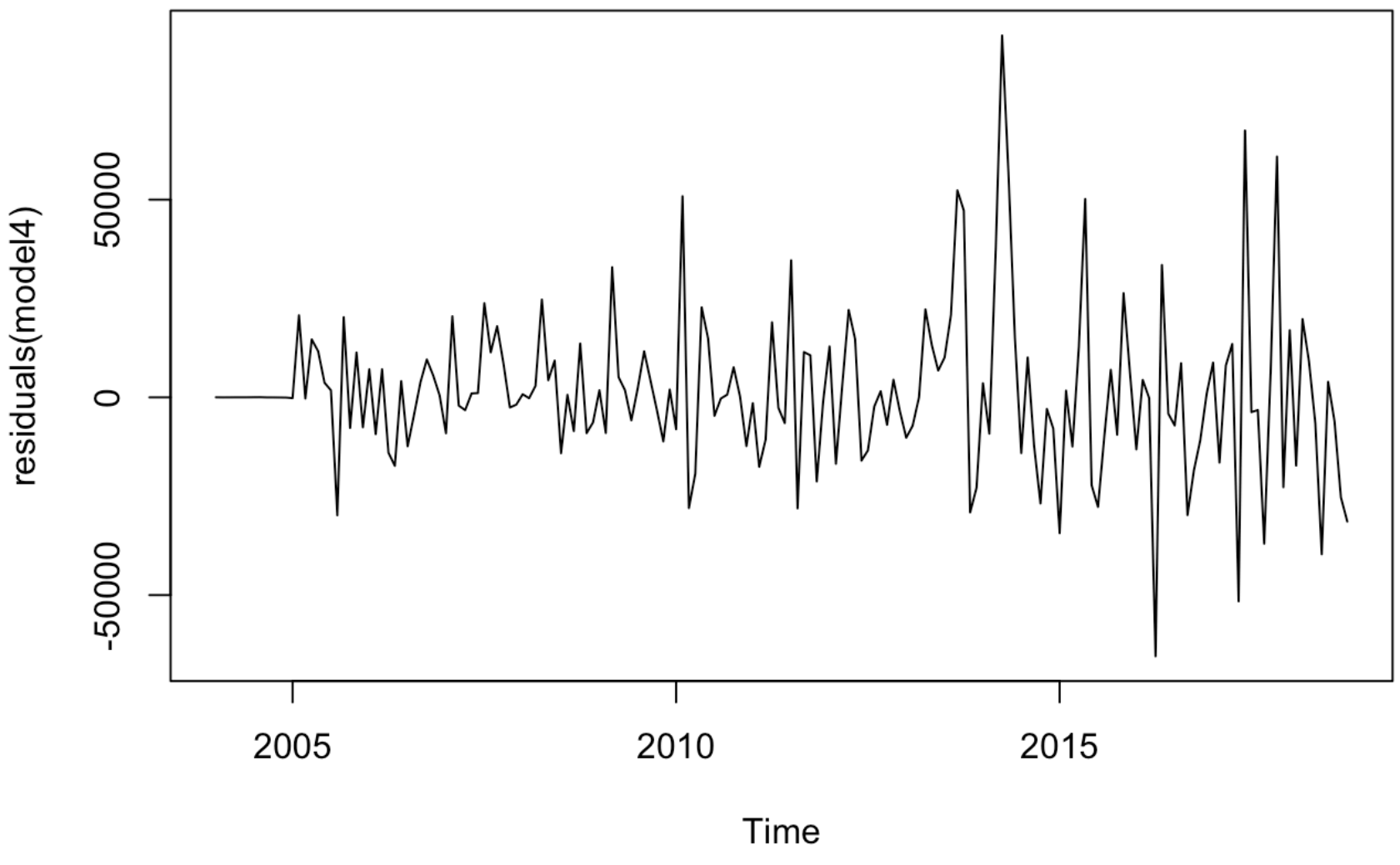
```
##
##  Box-Ljung test
##
## data:  residuals(model2)
## X-squared = 0.0017608, df = 1, p-value = 0.9665
```

```
#選擇AIC較小的，因此選擇model2
```

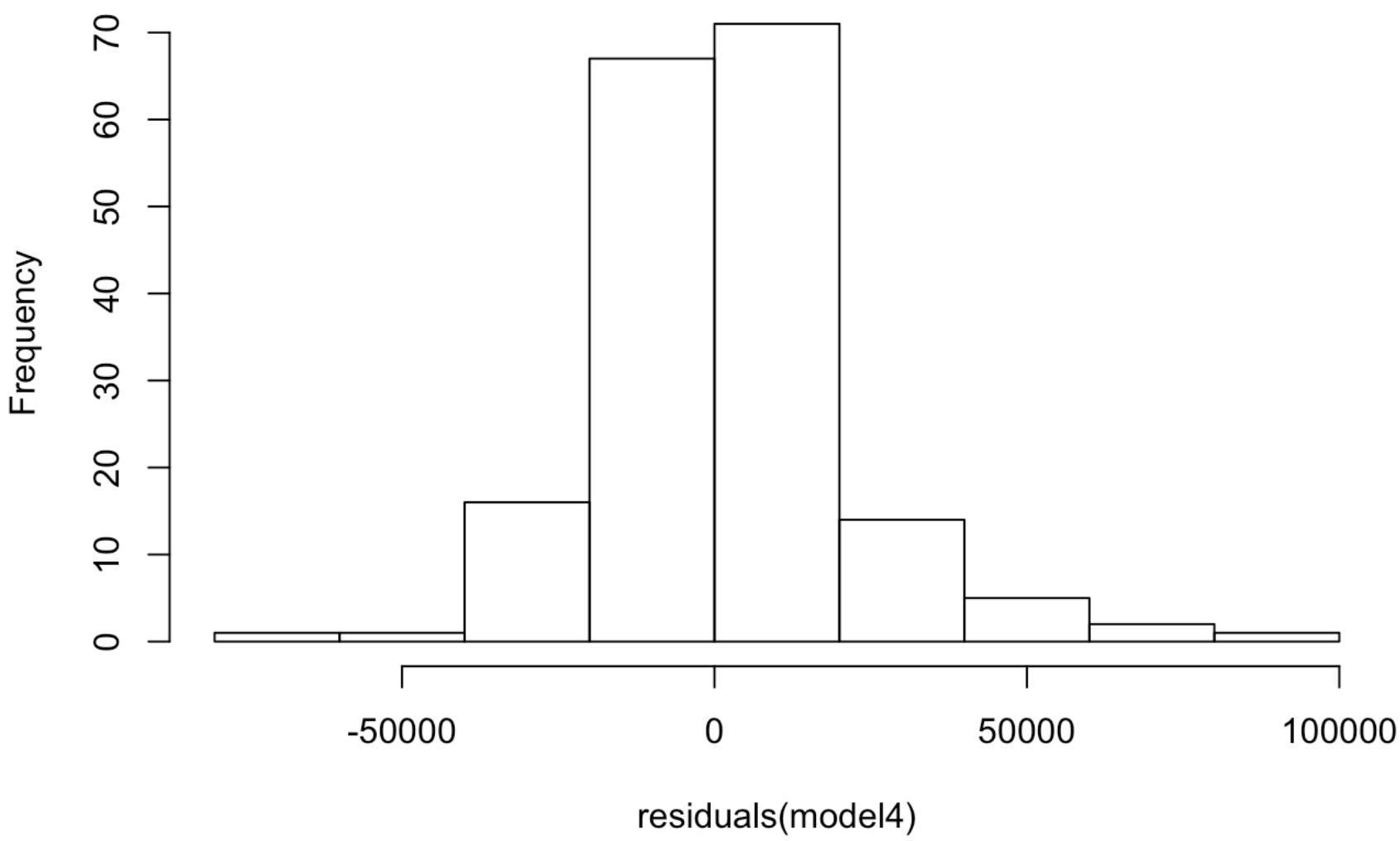
```
#修正模型為ARIMA(3,1,3)X(1,1,2)12 ，並進行參數估計
model3=arima(passengers,order=c(3,1,3),seasonal=list(order=c(1,1,2),period=12))

#除估計出 Φ1 外，其餘參數皆與 0 距離介在兩個標準差外，與 0 有顯著差異，故假設其為 0，再重新估計參數
model4=arima(passengers,order=c(3,1,3),seasonal=list(order=c(1,1,2),period=12),tra
nsform.pars = FALSE,fixed=c(NA,NA,NA,NA,NA,NA,NA,0,NA))
```

```
## Warning in log(s2): 產生了 NaNs
```

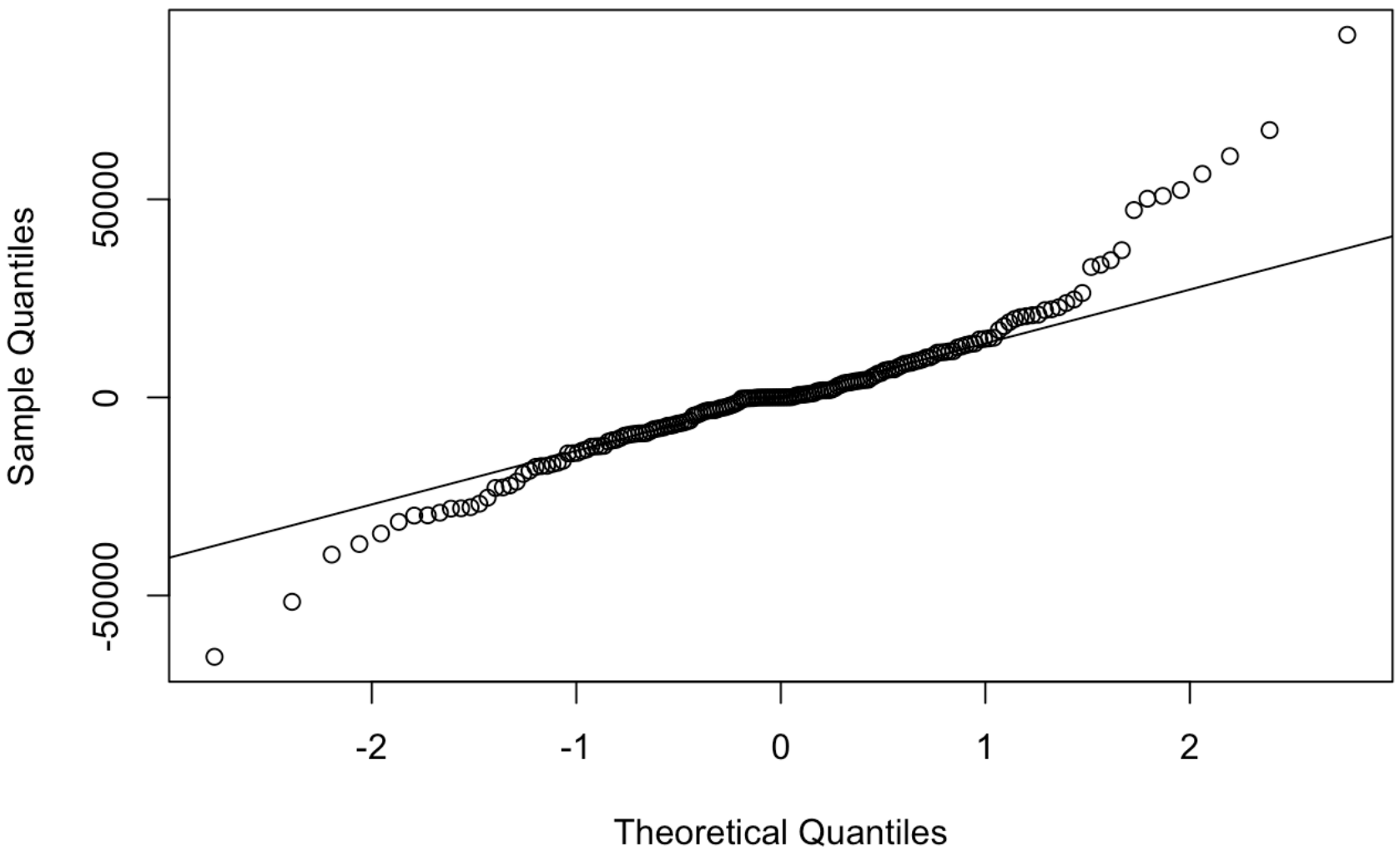
```
#對模型進行殘差檢定
plot(residuals(model4))
```



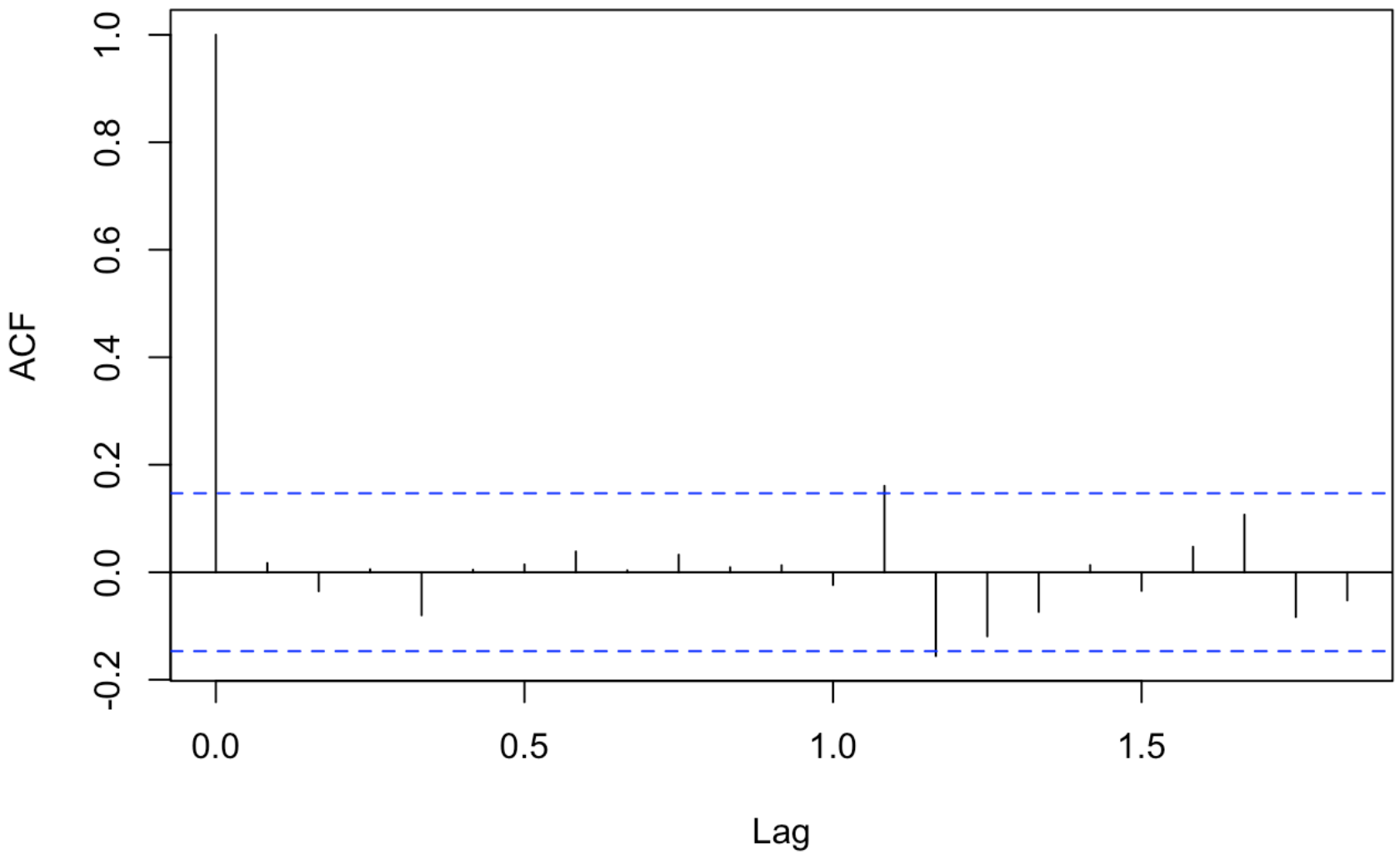

```
#殘差大致為隨機飄移

#觀察相關圖形
hist(residuals(model4))
```




```
qqnorm(residuals(model4))
qqline(residuals(model4))
```




```
acf(residuals(model4))
```




```
#觀察殘差之Q-Q plot，可看出在上下有飄移的情形，殘差的ACF值在lag=12超出信賴區間，表示模型的殘差可能有自我相關，故再由T test、Shapiro-Wilk normality Test和Box-Ljung test檢定
t.test(residuals(model4))
```

```
##
##  One Sample t-test
##
## data:  residuals(model4)
## t = 0.8615, df = 177, p-value = 0.3901
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -1668.207  4253.131
## sample estimates:
## mean of x
##  1292.462
```

```
shapiro.test(residuals(model4))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model4)
## W = 0.92863, p-value = 1.117e-07
```

```
Box.test(residuals(model4))
```

```
##
##  Box-Pierce test
##
## data:  residuals(model4)
## X-squared = 0.052004, df = 1, p-value = 0.8196
```

```
#得出結論
```

4. 類別資料分析

(a)fisher's exact test

適用於小樣本分析(特別是小於20的小樣本)在Fisher著名的茶實驗中，有人自稱可以分辨到底是茶先倒入杯中，或是牛奶先倒入杯中

```
fisher<-read.csv("fishertea.csv", header=T, sep=",")
fisher
```

```
##   real guess
## 1    1     1
## 2    1     1
## 3    1     1
## 4    1     2
## 5    2     2
## 6    2     2
## 7    2     1
## 8    2     2
```

```
fisher.test(table(fisher$real,fisher$guess))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  table(fisher$real, fisher$guess)
## p-value = 0.4857
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##    0.2117329 621.9337505
## sample estimates:
## odds ratio
##   6.408309
```

```
#p-value 大於0.05，不拒絕虛無假設，代表猜測與實際未有顯著關聯。
```

(b)2×2×K交叉表

```
#以一份佛羅里達（Florida）的研究為例，在151件白人殺害白人的案例中被判死刑的有19件；在9件白人殺害黑人的案例中被判死刑的有0件；在63件黑人殺害白人的案例中被判死刑的有11件；在103件黑人殺害黑人的案例中被判死刑的有6件
#建立表格
dp <- c(19, 132, 0, 9, 11, 52, 6, 97)
dp <- array(dp, dim=c(2,2,2))
dimnames(dp) <- list(DeathPen=c("yes","no"),Defendant=c("white","black"),Victim=c("white","black"))
dpflat = ftable(dp, row.vars=c("Victim","Defendant"), col.vars="DeathPen")
deathpenalty = as.data.frame(dpflat)
library(reshape2)
dp <- melt(deathpenalty)
```

```
## Using Victim, Defendant, DeathPen as id variables
```

```
#Using Victim, Defendant, DeathPen as id variables
dpwide <- dcast(dp, ... ~ DeathPen)
dpwide
```

```
##   Victim Defendant variable yes  no
## 1  white     white     Freq  19 132
## 2  white     black     Freq   0   9
## 3  black     white     Freq  11  52
## 4  black     black     Freq   6  97
```

```
#建立模型
dep.fit1 <- glm(cbind(yes,no) ~ Defendant + Victim, family=binomial, data=dpwide)
summary(dep.fit1)
```

```
##
## Call:
## glm(formula = cbind(yes, no) ~ Defendant + Victim, family = binomial,
##     data = dpwide)
##
## Deviance Residuals:
##       1        2        3        4
##  0.0803  -0.8145  -0.1072   0.1394
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -1.9581     0.2451  -7.991 1.34e-15 ***
## Defendantblack  -1.3242     0.5193  -2.550   0.0108 *
## Victimblack      0.4402     0.4009   1.098   0.2722
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8.13161  on 3  degrees of freedom
## Residual deviance: 0.70074  on 1  degrees of freedom
## AIC: 19.015
##
## Number of Fisher Scoring iterations: 4
```

```
#likelihood ratio test
#H0：victim’s race has no effect
#Ha：victim’s race has effect
#若p-value < 0.05, reject H0, so we can say victim’s has effect.
```

(c)logistic regression

利用crabs的資料進行分析

變數解釋

color：2 - light medium, 3 - medium, 4 - dark medium, 5 - dark

spine：1 - both good, 2 - one worn or broken, 3 - both worn or broken

width：carapace width in cm

weight：weight in g

```
crabs <- read.table("http://www.math.montana.edu/shancock/courses/stat539/data/hor
seshoe.txt",header=T)
y<- as.numeric(crabs$satell> 0)
data<-cbind(crabs,cbind(y))
head(data)
```

```
##   color spine width satell weight y
## 1     2     3  28.3      8   3.05 1
## 2     3     3  22.5      0   1.55 0
## 3     1     1  26.0      9   2.30 1
## 4     3     3  24.8      0   2.10 0
## 5     3     3  26.0      4   2.60 1
## 6     2     3  23.8      0   2.10 0
```

```
#建立logistic regression model for the probability of a satellite, using color alon
e as the predictor.Treat color as nominal scale (qualitative).
color=data$color
color.level <- factor(color)
model1<-glm(data$y ~ color.level, family = binomial(link=logit), data=data)
summary(model1)
```

```
##
## Call:
## glm(formula = data$y ~ color.level, family = binomial(link = logit),
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6651  -1.3370   0.7997   0.7997   1.5134
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.0986     0.6667   1.648   0.0994 .
## color.level2  -0.1226     0.7053  -0.174   0.8620
## color.level3  -0.7309     0.7338  -0.996   0.3192
## color.level4  -1.8608     0.8087  -2.301   0.0214 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 212.06  on 169  degrees of freedom
## AIC: 220.06
##
## Number of Fisher Scoring iterations: 4
```

```
#logit[P (Y= 1)] = 1.0986 + (-0.1226)c1 + (-0.7309)c2 + (-1.8608)c3

#conduct a likelihood-ratio test of the hypothesis that color has no effect.
#p-value < 0.05, reject H0, so color has effect.

#Treating color in a quantitative manner
model2<-glm(data$y ~ data$color, family = binomial(link=logit), data=data)
summary(model2)
```

```
##
## Call:
## glm(formula = data$y ~ data$color, family = binomial(link = logit),
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9103  -1.2719   0.8142   0.8142   1.3937
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.3635     0.5551   4.257 2.07e-05 ***
## data$color   -0.7147     0.2095  -3.412 0.000645 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 213.30  on 171  degrees of freedom
## AIC: 217.3
##
## Number of Fisher Scoring iterations: 4
```

```
#logit[ π̂(x)]=3.0781 − 0.7147x

#conduct a likelihood-ratio test of the hypothesis that color has no effect.
#p-value < 0.05, reject H0： color has effect.

#檢測weight與width是否有交互作用(interaction)
#H0：no interaction
#Ha：interaction
model4<-glm(data$y ~ data$weight+data$width, family = binomial(link=logit), data=d
ata)
#conduct a likelihood-ratio test of the hypothesis that color has no effect.
#p-value=0.2236 > 0.05→do not reject H0, for easier interpretation, use simpler mo
del (no interaction)
```

五、文字探勘

安裝套件

install.packages(“jiebaR”)

library(jiebaR)

install.packages(“wordcloud”)

library(wordcloud)

install.packages(“wordcloud2”)

library(wordcloud2)

```
#中文斷詞
#mixseg<-worker()
#mixseg["那一年我們望著星空有那麼多的燦爛的夢"]
```

[1] “那” “一年” “我們” “望著” “星空” “有” “那麼” “多” “的” “燦爛” “的” “夢”

```
#讀入文字檔
#md<-scan("Desktop/Mayday.txt",sep="\n",what="",encoding="UTF-8")
#head(md)
#mixseg<-worker()
#segment(head(md),mixseg)
```

[1] “脫下長” “日” “的” “假面” “奔” “向” “夢幻”

[8] “的” “疆界” “南瓜” “馬車” “的” “午夜” “換上”

[15] “童話” “的” “玻璃” “鞋” “讓” “我” “享受”

```
#斷詞
#md2<-segment(md,mixseg)
#檢視md2的前30的詞頻統計
#sort(table(md2),decreasing=T)[1:30]
```

```
md2
 的  我  你  了  是  在  和  都 有我們  著  不  誰  像  讓  要  就  卻  再
449 317 212 143  96  81  80  74  65  62  61  57  53  45  44  39  38  38  38
快樂 世界  又  那 自己  為  這  只 回憶  到 一天
 37  37  36  35  32  31  30  30  29  28  28
```

Caption for the picture.

```
#篩選字串長度介於2-6的詞並進行前30的詞頻統計
#md3<-md2[nchar(md2)>1 & nchar(md2)<7]
#sort(table(md3),decreasing=T)[1:30]
```

```
md3
  我們  快樂  世界  自己  回憶  一天  一個  如果  動次  沒有
  62    37    37    32    29    28    27    25    24    24
 什麼  天使  突然  真的  不能  不要  煩惱  就是  依賴  永遠
  23    23    22    21    20    20    20    20    20    20
 那麼  未來  眼淚  一次  自由  不會  後來 日日夜夜 有沒有  最後
  18    16    16    16    16    15    15    15    15    15
```

Caption for the picture.

```
#輸出前50個md3的高頻統計詞
#wordFreq50=sort(table(md3),decreasing=T)[1:50]
#wordFreq50
```

```
#匯出文字雲
#wordcloud2(wordFreq50, size = 1,shape = 'pentagon')
```

Caption for the picture.

```
#col設置字體顏色為彩虹色
#shape為形狀 circle為預設值
#shape='circle','cardioid','star','diamond','triangle-foward','pentagon'
#wordcloud2(wordFreq50, size = 1,col=rainbow(12),shape = 'star')
```

Caption for the picture.

參考資料：

https://rpubs.com/skydome20/Table (https://rpubs.com/skydome20/Table)

https://sites.google.com/site/rlearningsite/ (https://sites.google.com/site/rlearningsite/)

http://ccckmit.wikidot.com/r:main (http://ccckmit.wikidot.com/r:main)