Regression

Code

Hide

```
library(car)
library(QuantPsyc)
library(boot)
library(dplyr) # data mainpulation
library(cowplot)
library(ggplot2)
```

Hide

```
df1<- read.delim('/home/atrides/Desktop/R/statistics_with_R/07_Regression/Da
ta_Files/Album Sales 1.dat', header=TRUE)
```

Hide

```
albumsales1<-lm(formula=sales~adverts, data=df1)


# Interpreting a simple regression
summary(albumsales1)
```

```

Call:
lm(formula = sales ~ adverts, data = df1)

Residuals:
     Min       1Q   Median       3Q      Max
-152.949  -43.796   -0.393   37.040  211.866

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.341e+02  7.537e+00  17.799   <2e-16 ***
adverts     9.612e-02  9.632e-03   9.979   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 65.99 on 198 degrees of freedom
Multiple R-squared:  0.3346,    Adjusted R-squared:  0.3313
F-statistic: 99.59 on 1 and 198 DF,  p-value: < 2.2e-16
```

Hide

```
df2<- read.delim('/home/atrides/Desktop/R/statistics_with_R/07_Regression/Data_Files/Album Sales 2.dat', header=TRUE)

head(df2)
```

	adverts <dbl>	sales <int>	airplay <int>	attract <int>
1	10.256	330	43	10
2	985.685	120	28	7
3	1445.563	360	35	7
4	1188.193	270	33	7
5	574.513	220	44	5
6	568.954	170	19	5

6 rows

Hide

```
albumsales2<-lm(sales~adverts, data=df2)
albumsales3<-lm(sales~adverts+airplay+attract, data=df2) # or use update(albumsales2, .~.+attract+airplay)
```

Hide

```
summary(albumsales2)
```

```

Call:
lm(formula = sales ~ adverts, data = df2)

Residuals:
     Min       1Q   Median       3Q      Max 
-152.949  -43.796   -0.393   37.040  211.866 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1.341e+02  7.537e+00  17.799   <2e-16 ***
adverts     9.612e-02  9.632e-03   9.979   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 65.99 on 198 degrees of freedom
Multiple R-squared:  0.3346,    Adjusted R-squared:  0.3313 
F-statistic: 99.59 on 1 and 198 DF,  p-value: < 2.2e-16
```

Hide

```
summary(albumsales3)
```

```

Call:
lm(formula = sales ~ adverts + airplay + attract, data = df2)

Residuals:
     Min       1Q   Median       3Q      Max 
-121.324  -28.336   -0.451   28.967  144.132 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -26.612958  17.350001  -1.534    0.127    
adverts       0.084885   0.006923  12.261  < 2e-16 ***
airplay       3.367425   0.277771  12.123  < 2e-16 ***
attract      11.086335   2.437849   4.548 9.49e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 47.09 on 196 degrees of freedom
Multiple R-squared:  0.6647,    Adjusted R-squared:  0.6595 
F-statistic: 129.5 on 3 and 196 DF,  p-value: < 2.2e-16
```

Hide

```
# model parameters
print(lm.beta(albumsales3))  # standarized b balues
```

```
  adverts   airplay   attract 
0.5108462 0.5119881 0.1916834 
```

Hide

```
print(confint(albumsales3))
```

```
                  2.5 %      97.5 %
(Intercept) -60.82960967  7.60369295
adverts       0.07123166  0.09853799
airplay       2.81962186  3.91522848
attract       6.27855218 15.89411823
```

Hide

```
# comparing Models
anova(albumsales2, albumsales3)
```

```
Analysis of Variance Table

Model 1: sales ~ adverts
Model 2: sales ~ adverts + airplay + attract
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1    198 862264
2    196 434575  2    427690 96.447 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Hide

```
cat("AIC_model2: ", AIC(albumsales2),"\nBIC_model2: ",  BIC(albumsales2))
```

```
AIC_model2:  2247.375
BIC_model2:  2257.27
```

Hide

```
cat("AIC_model3: ", AIC(albumsales3),"\nBIC_model3: ",  BIC(albumsales3))
```

```
AIC_model3:  2114.337
BIC_model3:  2130.828
```

Hide

```
# outliers and influential cases

df2$residuals <-resid(albumsales3)
df2$standarized.residuals <- rstandard(albumsales3)
df2$studentized.residuals <- rstudent(albumsales3)
df2$cooks <- cooks.distance(albumsales3)
df2$dfbeta <- dfbeta(albumsales3)
df2$dffit <- dffits(albumsales3)
df2$leverage <- hatvalues(albumsales3)
df2$covratio <- covratio(albumsales3)


# saving this data
write.table(df2,'albumSalesWithDiagnosticsData.dat', sep='\t', row.names = F
ALSE)

large_resid <- dplyr::filter(df2, standarized.residuals>2 | standarized.resi
duals< -2)
# these cases are to be analyzed coz they have somewhat large residuals
large_resid
```

Hide

```
# now lets see cooks distance , leverage , covariance ratio for 'these' case
s
k = 3 #number of predictors
n = 200 #number of objervations

average_leverage = (k+1)/n
average_leverage
```

```
[1] 0.02
```

Hide

```
cvr_low<- 1-3*average_leverage
cvr_high<- 1+3*average_leverage

large_resid$cov_ideal_low <- cvr_low
large_resid$cov_ideal_high <- cvr_high

large_resid
```

Hide

```
# from this large residual model we conclude that
# Most of our 12 potential outliers have CVR values within or just outside t
he boundaries.
# none of them has a Cook’s distance greater than 1,  so none of the cases i
s having an undue influence on the model.


# So , Note:

# i) Look at standardized residuals and check that no more than 5% of cases
have absolute values above 2,
#    and that no more than about 1% have absolute values above 2.5. Any case
with a value above about 3 could be an outlier.

# ii)Look at the values of Cook’s distance: any value above 1 indicates a ca
se that might be influencing the model.

# iii)Calculate the average leverage (the number of predictors plus 1, divid
ed by the sample size)
#      and then look for values greater than twice or three times this avera
ge value

# iv)Calculate the upper and lower limit of acceptable values for the covari
ance ratio, CVR.
#        The upper limit is 1 plus three times the average leverage, whereas
#        the lower limit is 1 minus three times the average leverage.
#        Cases that have a CVR falling outside these limits may be problemat
ic
```

Hide

```
#
------------------------------------------------------------------------------
-------------------------------------------------
#
------------------------------------------------------------------------------
-------------------------------------------------
#
------------------------------------------------------------------------------
-------------------------------------------------
#
------------------------------------------------------------------------------
-------------------------------------------------


# Testing various assumptions

# i) Assumptions of Independent Errors
car::durbinWatsonTest(albumsales3)  # hence assumption is valid here
```

```
 lag Autocorrelation D-W Statistic p-value
   1       0.0026951      1.949819   0.738
 Alternative hypothesis: rho != 0
```

Hide

```
# ii) Assumption of no multicollinearity
vif_<- car::vif(albumsales3)
print(vif_)
```

```
 adverts  airplay  attract
1.014593 1.042504 1.038455
```

Hide

```
print(mean(vif_))
```

```
[1] 1.03185
```

Hide

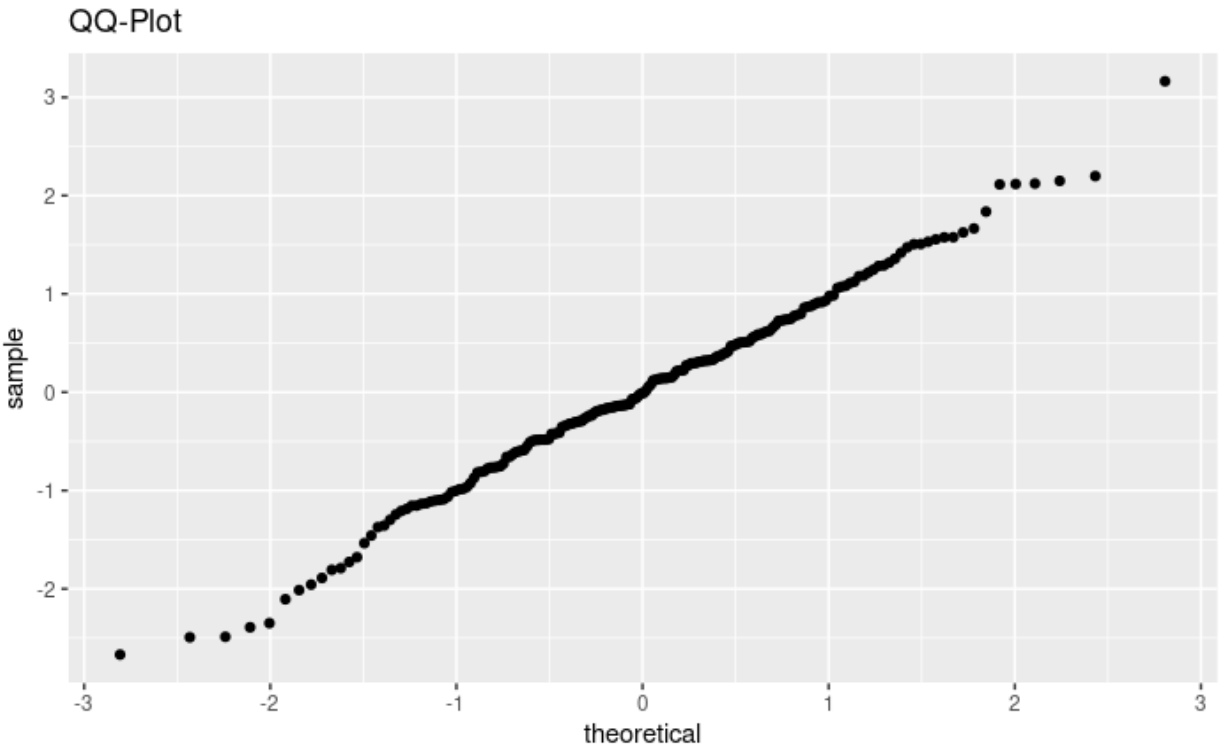
```
# the assumption of multicollinearity is followed too
```

Hide

```
# iii) Assumption about the Residuals
df2$fitted.values<- fitted.values(albumsales3)
df2$std_fitted.values<- (fitted.values(albumsales3)-mean(fitted.values(album
sales3)))/sd(fitted.values(albumsales3))
resid_plot<- ggplot(df2, aes(standarized.residuals,std_fitted.values))
resid_plot<- resid_plot+geom_point()+geom_smooth(formula='y~x',method = "l
m",alpha=0.1)



resid_qq<- ggplot(df2, aes(sample=studentized.residuals))
resid_qq<- resid_qq+stat_qq()+ggtitle('QQ-Plot')
resid_qq
```

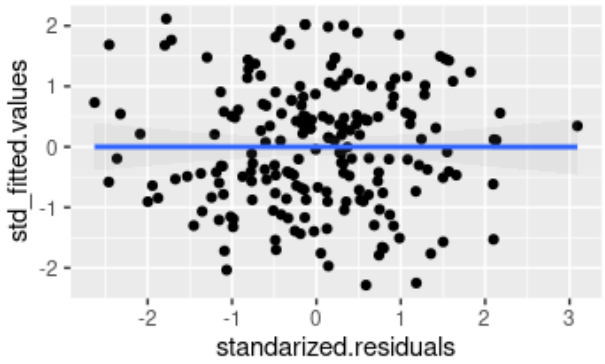


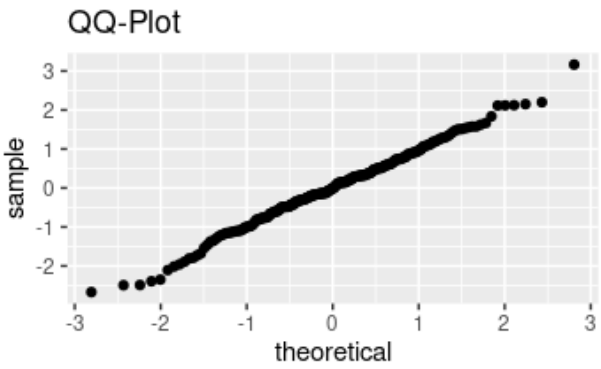
Hide

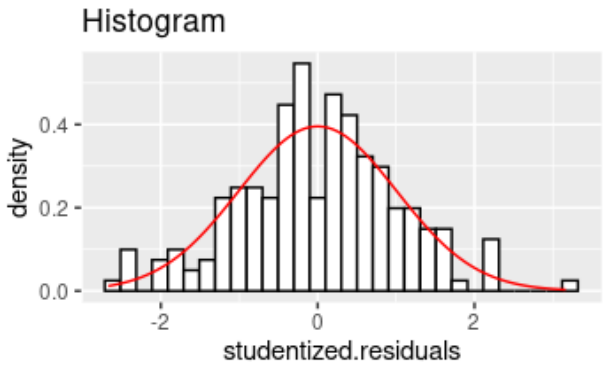
```
histresid<- ggplot(df2, aes(studentized.residuals))
histresid<- histresid+geom_histogram(aes(y=..density..),colour='black', fill
='white')+
  ggtitle('Histogram')+
  stat_function(fun = dnorm, args = list(mean=0, sd=sd(df2$studentized.resid
uals, na.rm = TRUE)), colour='red')


plot_grid(resid_plot, resid_qq, histresid,ncol=2, nrow=2 )
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```







Hide

```
# this assumption was also met
```