final_project

Tianqi Wu

4/28/2020

```r
library(lmtest)
library(corrplot)
library(randomForest)
library(leaps)
library(car)
library(splines)
library(faraway)
library(nlme)
library(MASS)
```

```r
## read data
data_all = read.csv('stat425_fpdata.csv')
data = data_all[data_all$hotel=='City Hotel',]
data = data[, -1] ## delete variable hotel

## rename variables
colnames(data)[which(names(data) == "arrival_date_year")] <- "year"
colnames(data)[which(names(data) == "arrival_date_week_number")] <- "week"
colnames(data)[which(names(data) == "arrival_date_month")] <- "month"
colnames(data)[which(names(data) == "arrival_date_day_of_month")] <- "day"
colnames(data)[which(names(data) == "stays_in_weekend_nights")] <- "weekend_night"
colnames(data)[which(names(data) == "stays_in_week_nights")] <- "week_night"
colnames(data)[which(names(data) == "reserved_room_type")] <- "room_type"
colnames(data)[which(names(data) == "total_of_special_requests")] <- "requests"
```

Section 2: Exploratory Data Analysis

```r
## check missing value
sum(is.na(data))
```

```
## [1] 0
```

```r
dim(data)
```

```
## [1] 1618   17
```

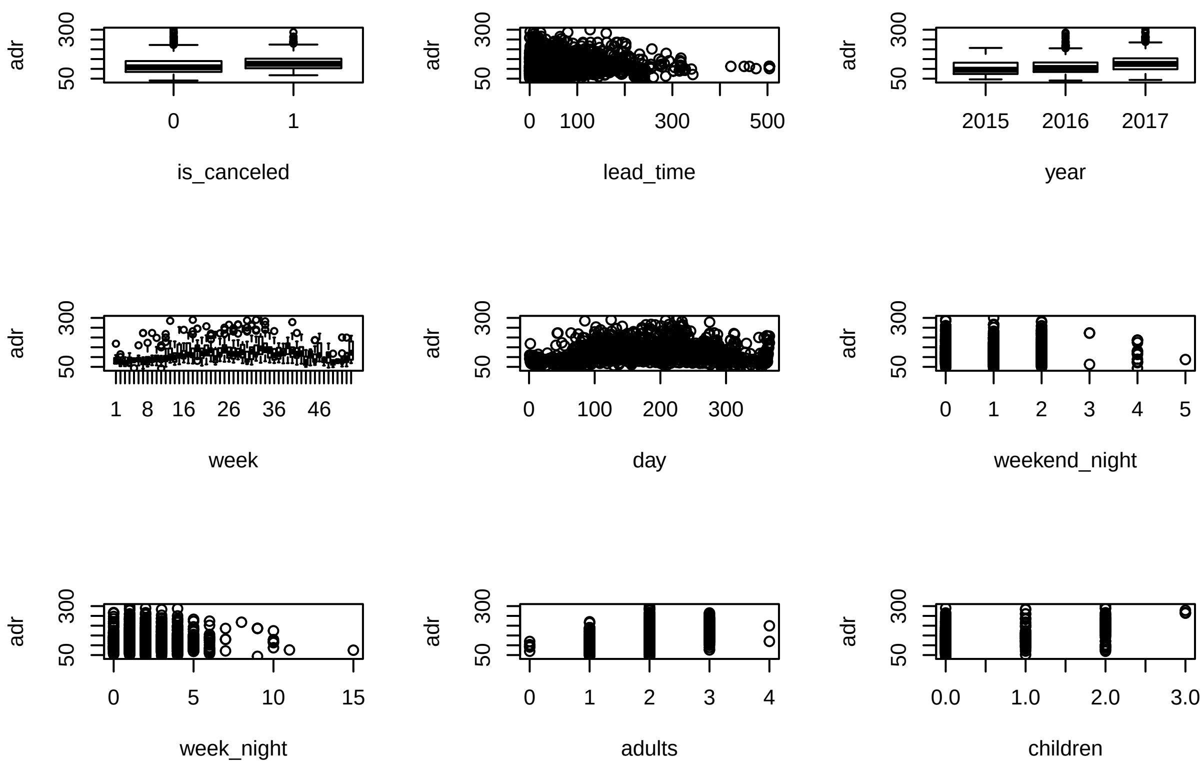
```r
## remove some apparent unusual obersvations
data = data[which(data$adr>12),]
data = data[which(data$market_segment!='Aviation'),]
data = data[which(data$market_segment!='Complementary'),]
data = data[which(data$room_type!='C'),]
data = droplevels(data)
```

```r
## numeric to categoric
data$is_canceled = as.factor(data$is_canceled)
data$week = as.factor(data$week)
data$year = as.factor(data$year)


df.month = data.frame(month = format(ISOdate(2015,1:12,1),"%B"))
data$month_number = mapply(function(x){which(df.month==as.character(x))}, data$month)
dayofyear = function(month, day){as.POSIXlt(paste(day, month, sep='.'), format = "%d.%m")$yday+1}
data$day = mapply(dayofyear, data$month_number, data$day)

## week and month are redundant
# chi-squared test: week and month are dependent
tbl = table(as.factor(data$week), data$month)
chisq.test(tbl)
```

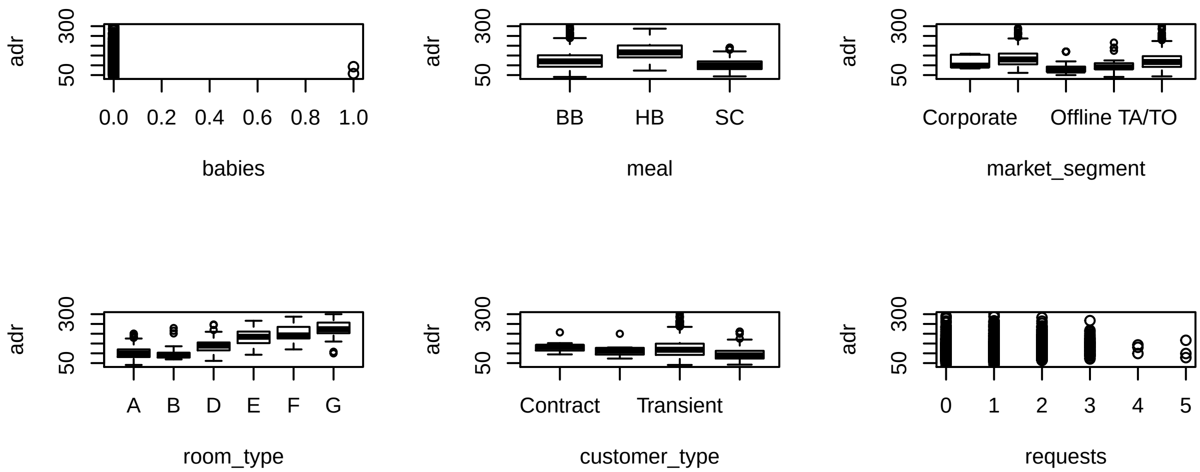
```
## Warning in chisq.test(tbl): Chi-squared approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  tbl
## X-squared = 16081, df = 572, p-value < 2.2e-16
```

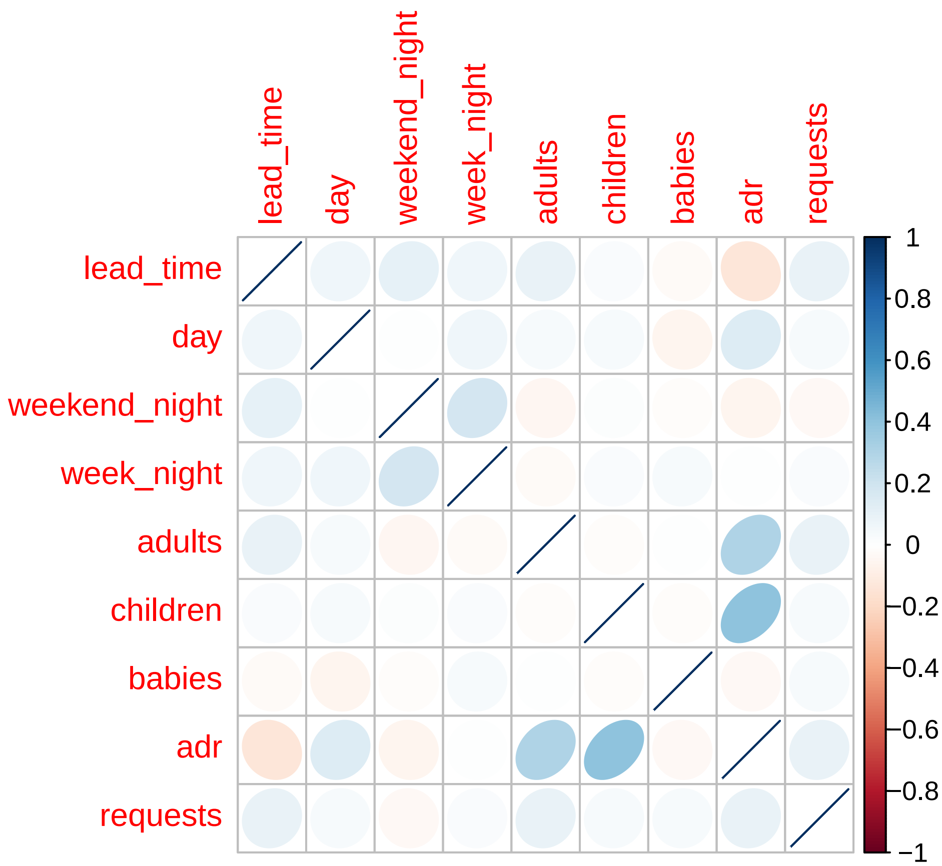
```r
# Remove day, month,month_number
data = subset(data,select=-c(month, month_number))

## Generate Sec 2, Figure 1: Graphic display
par(mfrow = c(3, 3))
plot(adr~.,data)
```



```r
## Generate Sec 2, Figure 2:  check collinearity
par(mfrow = c(1, 1))
```




```r
numeric = unlist(lapply(data, is.numeric))
corrplot(cor(data[,numeric]), method="ellipse")
```




```r
round(cor(data[,numeric]),1)
```

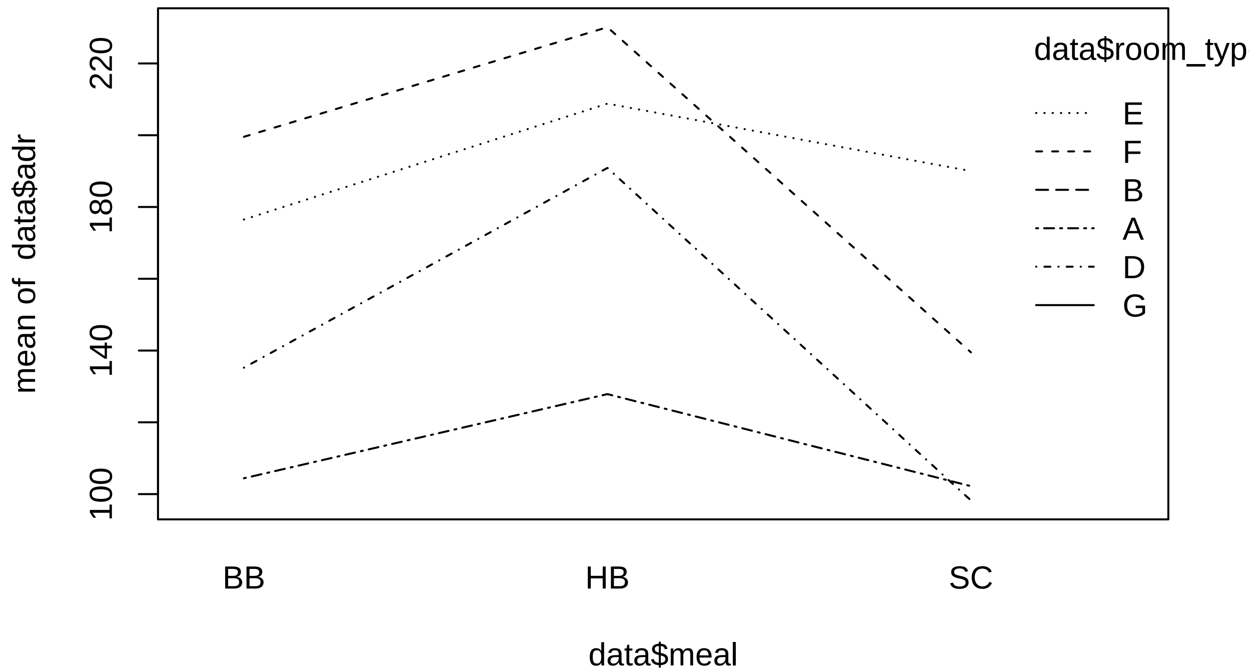
```
##               lead_time  day weekend_night week_night adults children babies
## lead_time           1.0  0.1           0.1        0.1    0.1      0.0    0.0
## day                 0.1  1.0           0.0        0.1    0.0      0.0   -0.1
## weekend_night       0.1  0.0           1.0        0.2    0.0      0.0    0.0
```

```
## week_night          0.1  0.1           0.2        1.0    0.0       0.0     0.0
## adults              0.1  0.0           0.0        0.0    1.0       0.0     0.0
## children            0.0  0.0           0.0        0.0    0.0       1.0     0.0
## babies              0.0 -0.1           0.0        0.0    0.0       0.0     1.0
## adr                -0.1  0.1          -0.1        0.0    0.3       0.4     0.0
## requests            0.1  0.0           0.0        0.0    0.1       0.0     0.0
##                adr requests
## lead_time     -0.1      0.1
## day            0.1      0.0
## weekend_night -0.1      0.0
## week_night     0.0      0.0
## adults         0.3      0.1
## children       0.4      0.0
## babies         0.0      0.0
## adr            1.0      0.1
## requests       0.1      1.0
```

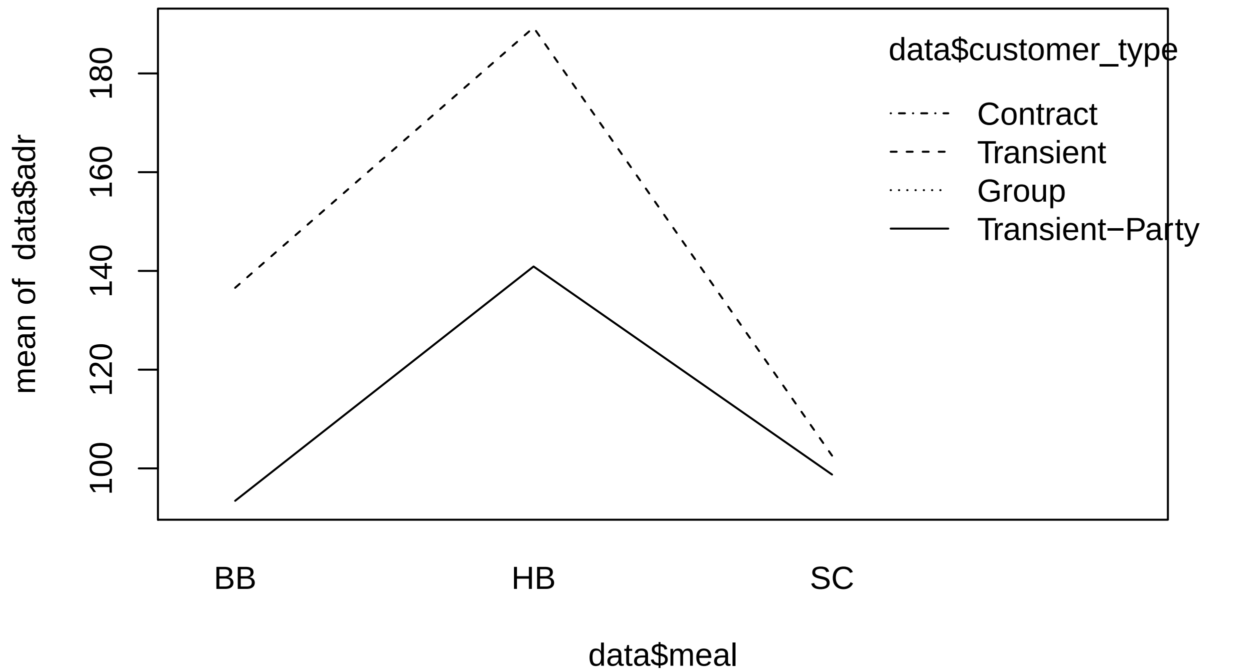
```r
summary(data)
```

```
##  is_canceled   lead_time        year          week           day
##  0:1173      Min.   :  0.00   2015:114   18     :  66   Min.   :  1.0
##  1: 425      1st Qu.: 17.00   2016:735   27     :  61   1st Qu.:121.0
##              Median : 56.00   2017:749   29     :  55   Median :180.5
##              Mean   : 82.68              22     :  52   Mean   :180.4
##              3rd Qu.:127.00              32     :  52   3rd Qu.:240.0
##              Max.   :504.00              25     :  51   Max.   :366.0
##                                          (Other):1261
##  weekend_night      week_night         adults         children
##  Min.   :0.0000   Min.   : 0.000   Min.   :0.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.: 1.000   1st Qu.:2.000   1st Qu.:0.0000
##  Median :1.0000   Median : 1.000   Median :2.000   Median :0.0000
##  Mean   :0.8673   Mean   : 1.876   Mean   :1.869   Mean   :0.1471
##  3rd Qu.:2.0000   3rd Qu.: 3.000   3rd Qu.:2.000   3rd Qu.:0.0000
##  Max.   :5.0000   Max.   :15.000   Max.   :4.000   Max.   :3.0000
##
##      babies          meal            market_segment room_type
##  Min.   :0.000000   BB:1094   Corporate    :  24   A:1047
##  1st Qu.:0.000000   HB:  31   Direct       : 233   B:  30
##  Median :0.000000   SC: 473   Groups       : 126   D: 345
##  Mean   :0.001252             Offline TA/TO: 109   E:  70
##  3rd Qu.:0.000000             Online TA    :1106   F:  68
##  Max.   :1.000000                                  G:  38
##
##          customer_type       adr            requests
##  Contract       :  17   Min.   : 40.67   Min.   :0.0000
##  Group          :   9   1st Qu.: 89.10   1st Qu.:0.0000
##  Transient      :1324   Median :112.67   Median :1.0000
##  Transient-Party: 248   Mean   :121.05   Mean   :0.7735
##                         3rd Qu.:144.86   3rd Qu.:1.0000
##                         Max.   :300.00   Max.   :5.0000
##
```

```r
## Generate Sec 2, Figure 3: interaction plots
interaction.plot(data$meal, data$room_type, data$adr)
```




```
interaction.plot(data$meal, data$customer_type, data$adr)
```



Section 3: Method

```
## train_test_split
set.seed(123)
index = sample(1:nrow(data),size=floor(0.8*nrow(data)))
train_data = data[index,]
test_data = data[-index,]
```

```r
test_x = subset(test_data, select = -c(adr))
test_y = test_data$adr
```

```r
## 3.1 simple model
model.3.1 = lm(adr~.-day, train_data)
#summary(model.3.1)

## Training R^2
summary(model.3.1)$r.squared
```

```
## [1] 0.7701231
```

```r
## Training RMSE
sqrt(sum((model.3.1$fitted.values-train_data$adr)^2)/nrow(train_data))
```

```
## [1] 20.907
```

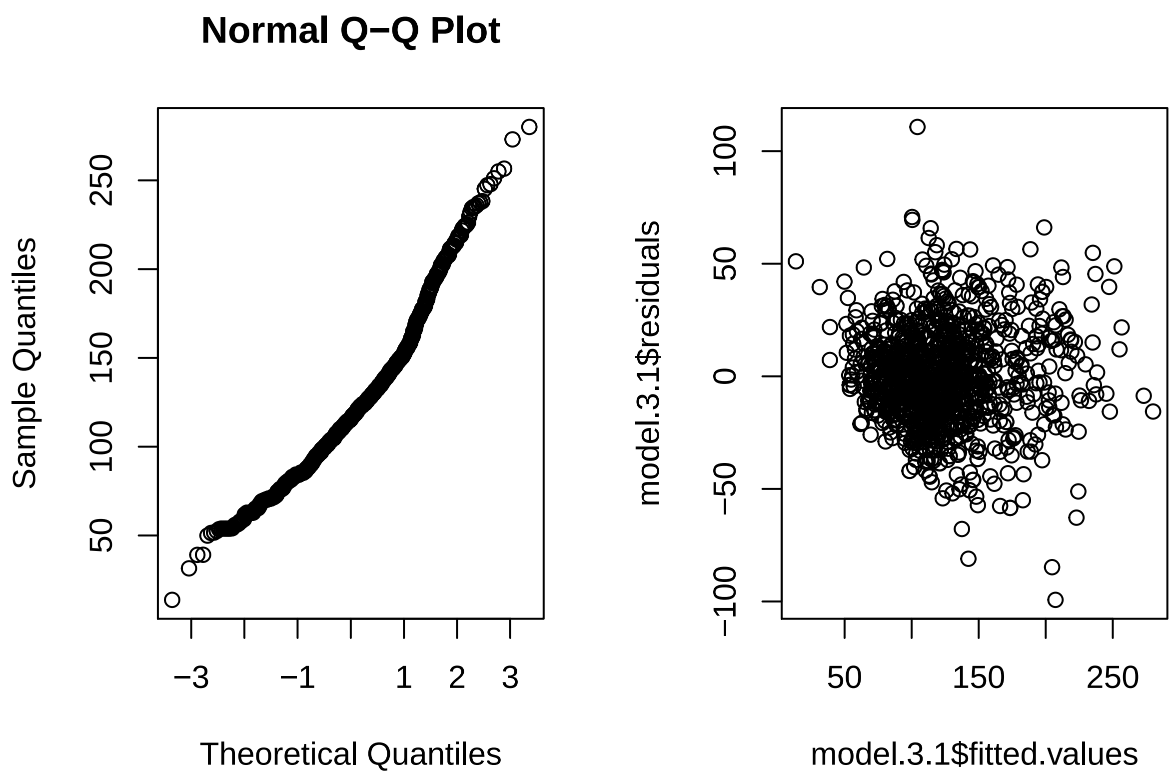
```r
## testing R^2 squared
predicted.adr = predict(model.3.1, newdata=test_x)
1-sum((predicted.adr-test_y)^2)/sum((test_y-mean(test_y))^2)
```

```
## [1] 0.725059
```

```r
## testing RMSE
sqrt(sum((predicted.adr-test_y)^2)/nrow(test_data))
```

```
## [1] 23.35453
```

```r
## Generate Sec 3.2, Figure 4: diagnostic plots
par(mfrow=c(1,2))
qqnorm(model.3.1$fitted.values)
plot(model.3.1$fitted.values, model.3.1$residuals)
```



```r
## constant variance test
bptest(model.3.1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model.3.1
## BP = 261.13, df = 76, p-value < 2.2e-16
```

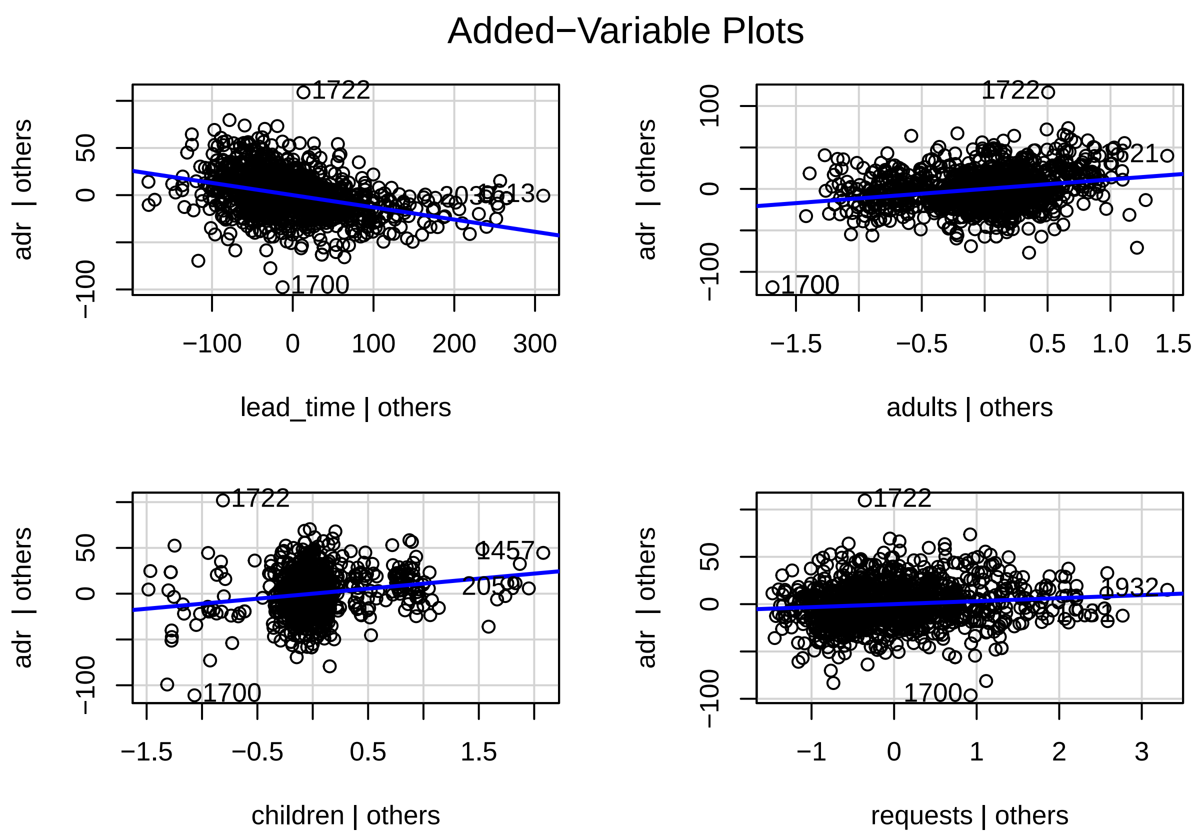
```r
## normality
shapiro.test(residuals(model.3.1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model.3.1)
## W = 0.98562, p-value = 6.241e-10
```

```r
## error independence
dwtest(model.3.1)
```

```
##
##  Durbin-Watson test
##
## data:  model.3.1
## DW = 2.0233, p-value = 0.6617
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
## model structure
avPlots(model.3.1,~lead_time+adults+children+requests)
```




```r
## variable selection
step(model.3.1, scope=list(upper=~., lower=~1), trace=0)
```

```
##
## Call:
## lm(formula = adr ~ is_canceled + lead_time + year + week + adults +
##     children + meal + market_segment + room_type + customer_type +
##     requests, data = train_data)
##
## Coefficients:
##                  (Intercept)                  is_canceled1
##                      36.2632                        9.7298
##                    lead_time                      year2016
##                      -0.1303                       15.7103
##                     year2017                         week2
##                      36.9027                       -5.2799
##                        week3                         week4
##                       2.4971                       13.0280
##                        week5                         week6
##                      11.6415                        3.9487
##                        week7                         week8
##                      12.8500                        8.7973
##                        week9                        week10
##                      14.3766                       11.1962
##                       week11                        week12
##                      17.7930                       22.8931
##                       week13                        week14
##                      31.3177                       27.9713
```

```
##                     week15                     week16
##                    46.5512                    39.9628
##                     week17                     week18
##                    46.7183                    52.3862
##                     week19                     week20
##                    62.3351                    47.9892
##                     week21                     week22
##                    57.3110                    56.3333
##                     week23                     week24
##                    49.1404                    48.1587
##                     week25                     week26
##                    54.6654                    39.6424
##                     week27                     week28
##                    40.8335                    56.3686
##                     week29                     week30
##                    49.0983                    50.8093
##                     week31                     week32
##                    51.2737                    63.2271
##                     week33                     week34
##                    62.0573                    60.9497
##                     week35                     week36
##                    51.0325                    48.9721
##                     week37                     week38
##                    50.6476                    66.0920
##                     week39                     week40
##                    92.0967                    67.8230
##                     week41                     week42
##                    65.3058                    64.2772
##                     week43                     week44
##                    51.7369                    38.8379
##                     week45                     week46
##                    37.4612                    57.6967
##                     week47                     week48
##                    30.8480                    24.6154
##                     week49                     week50
##                    18.0433                    26.4556
##                     week51                     week52
##                    20.7015                    20.4298
##                     week53                     adults
##                    48.9224                    11.5074
##                   children                     mealHB
##                    10.9959                    24.4304
##                     mealSC        market_segmentDirect
##                   -16.9748                    11.6751
##       market_segmentGroups  market_segmentOffline TA/TO
##                    -2.3870                    -3.7843
##    market_segmentOnline TA                  room_typeB
##                    13.5307                    -7.3926
##                 room_typeD                  room_typeE
##                    14.4313                    50.1592
##                 room_typeF                  room_typeG
##                    59.0946                    91.5094
##          customer_typeGroup      customer_typeTransient
##                   -14.3458                   -20.9452
```

```
## customer_typeTransient-Party                            requests
##                        -11.9170                          3.1702
```

```r
## 3.2 Linear Regression with Interaction and Qudratic Terms
model.3.2.1 = lm(adr ~ is_canceled + lead_time + year + week + adults +
                  children + meal + market_segment + room_type + customer_type +
                  requests + meal:market_segment + meal:room_type +
                  meal:requests + market_segment:room_type  +
                  I(lead_time^2) + I(children^2) + I(adults^2), data = train_data)

## model diagnostics
## check leverage
n=nrow(train_data); p=ncol(train_data);
lev=influence(model.3.2.1)$hat
sort(lev, decreasing = TRUE)[1:6]
```

```
##       510      1061      1487      1696       633      1988
## 1.0000000 1.0000000 1.0000000 0.7668225 0.7668225 0.6288907
```

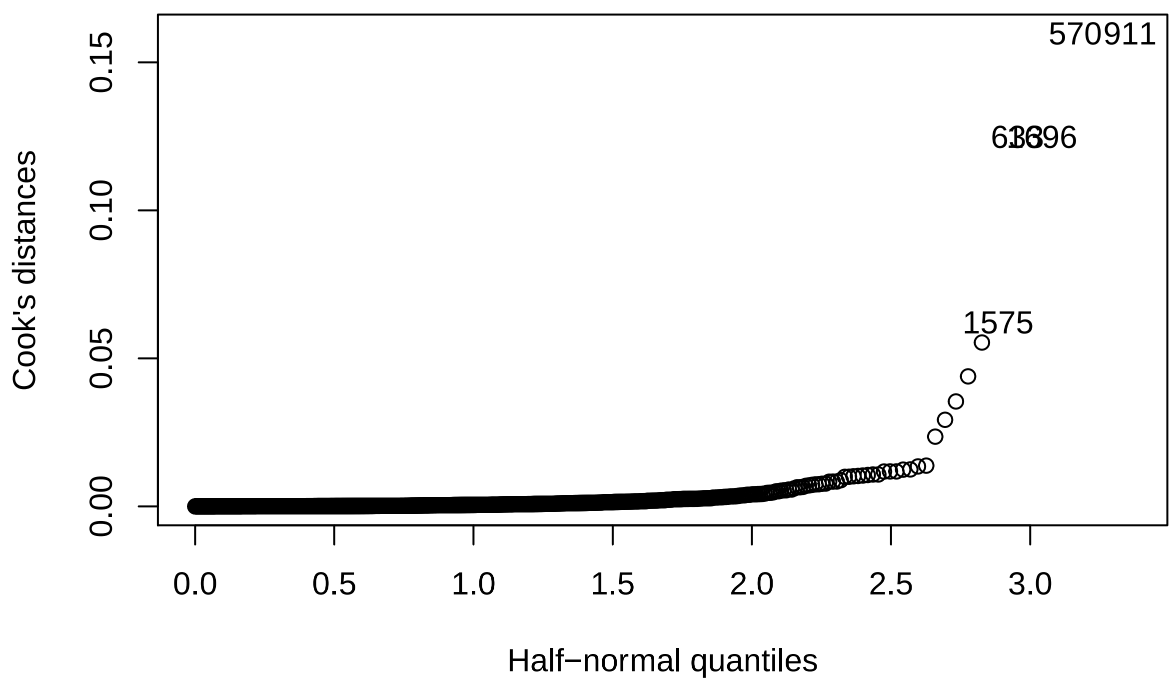
```r
## check outliers
jack=rstudent(model.3.2.1);
qt(.05/(2*n), n-p-1)
```

```
## [1] -4.127247
```

```r
sort(abs(jack), decreasing=TRUE)[1:5]
```

```
##     1722     1575     1700     1463      911
## 6.388524 5.789680 5.078372 4.135798 3.604173
```

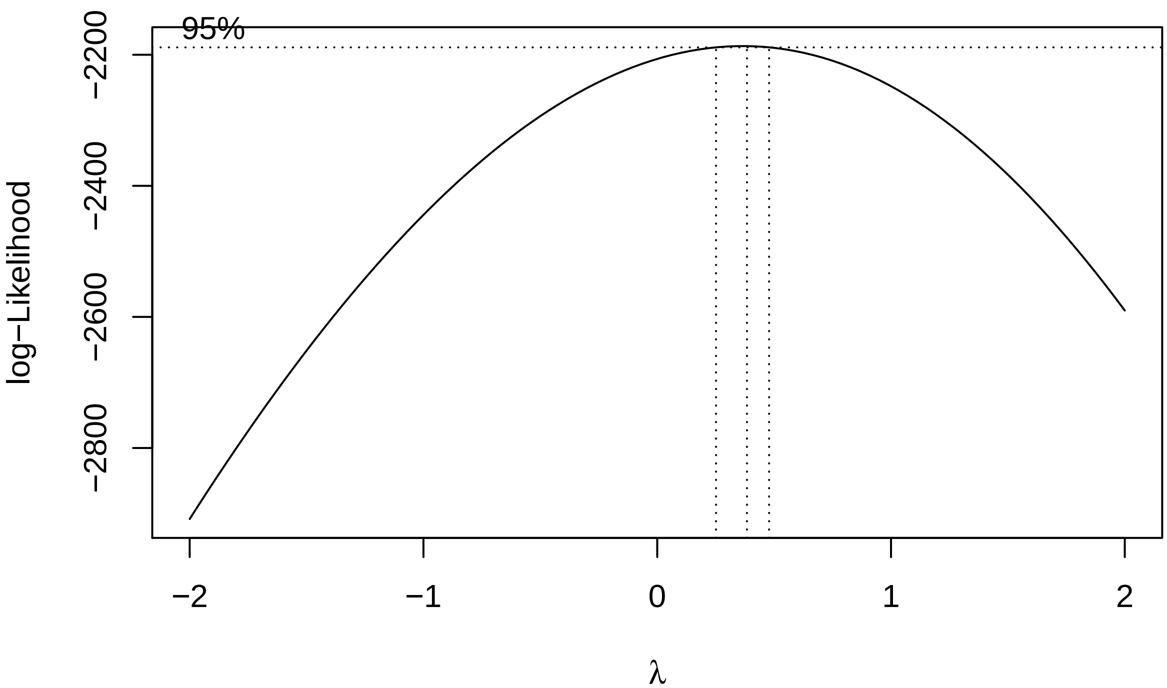
```r
## Influential observations
cook = cooks.distance(model.3.2.1)
halfnorm(cook, nlab=5, labs=row.names(train_data), ylab="Cook's distances")
```



```r
sort(abs(cook), decreasing=TRUE)[1:5]
```

```
##        911        570       1696        633       1575
## 0.15974305 0.15974305 0.12478496 0.12478496 0.06212536
```

```r
#max(cook)

deleted = c('510', '1061', '1487', '1722', '1700', '1575')
train_data.new = train_data[!row.names(train_data) %in% deleted,]

## refit with new train_data
model.3.2.2 = lm(adr ~ is_canceled + lead_time + year + week + adults +
                  children + meal + market_segment + room_type + customer_type +
                  requests + meal:market_segment + meal:room_type +
                  meal:requests + market_segment:room_type  +
                  I(lead_time^2) + I(children^2) + I(adults^2), data = train_data.new)

## Section 3.2 Figure 5: boxcox
par(mfrow=c(1,1))
bc = boxcox(model.3.2.2)
```




```r
bc$x[bc$y == max(bc$y)]
```

```
## [1] 0.3838384
```

```r
# Check for collinearlity
# conditonal number
x = model.matrix(model.3.2.2)[,c('lead_time','adults','children','requests')]
x = x - matrix(apply(x,2, mean), nrow(x),ncol(x), byrow=TRUE)
x = x / matrix(apply(x, 2, sd), nrow(x),ncol(x), byrow=TRUE)
apply(x,2,mean)
```

```
##     lead_time        adults      children      requests
## -4.264955e-18  4.593727e-17 -2.280660e-18 -3.541395e-17
```

```r
apply(x,2,var)
```

```
## lead_time    adults  children  requests 
##         1         1         1         1
```

```r
e = eigen(t(x) %*% x)
sqrt(e$val[1]/e$val)
```

```
## [1] 1.000000 1.078520 1.159356 1.169910
```

```r
# VIF
round(faraway::vif(x), dig=2)
```

```
## lead_time    adults  children  requests 
##      1.02      1.02      1.01      1.02
```

```r
## refit box-cox transformation
model.3.2.3 = lm(adr^(0.38) ~ is_canceled + lead_time + year + week + adults +
                  children + meal + market_segment + room_type + customer_type +
                  requests + meal:market_segment + meal:room_type +
                  meal:requests + market_segment:room_type  +
                  I(lead_time^2) + I(children^2) + I(adults^2), data = train_data.new)


summary(model.3.2.3)
```

```
## 
## Call:
## lm(formula = adr^(0.38) ~ is_canceled + lead_time + year + week + 
##     adults + children + meal + market_segment + room_type + customer_type + 
##     requests + meal:market_segment + meal:room_type + meal:requests + 
##     market_segment:room_type + I(lead_time^2) + I(children^2) + 
##     I(adults^2), data = train_data.new)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.77570 -0.20800  0.00538  0.22009  1.27380 
## 
## Coefficients: (19 not defined because of singularities)
##                                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)                             5.270e+00  2.028e-01  25.986  < 2e-16
## is_canceled1                            2.242e-01  2.836e-02   7.905 6.14e-15
## lead_time                              -4.389e-03  3.333e-04 -13.166  < 2e-16
## year2016                                3.467e-01  5.789e-02   5.989 2.81e-09
## year2017                                7.472e-01  6.632e-02  11.267  < 2e-16
## week2                                  -1.039e-01  1.178e-01  -0.882 0.377881
## week3                                  -2.325e-01  1.594e-01  -1.459 0.144810
## week4                                   2.120e-02  1.813e-01   0.117 0.906904
## week5                                  -2.937e-02  1.940e-01  -0.151 0.879735
## week6                                   6.053e-02  1.721e-01   0.352 0.725166
## week7                                  -8.586e-03  1.417e-01  -0.061 0.951707
## week8                                  -5.840e-02  1.595e-01  -0.366 0.714306
## week9                                   9.386e-02  1.426e-01   0.658 0.510667
## week10                                  8.779e-02  1.318e-01   0.666 0.505618
## week11                                  2.449e-01  1.201e-01   2.038 0.041726
## week12                                  3.570e-01  1.252e-01   2.851 0.004437
```

```
## week13                              6.589e-01  1.332e-01   4.946 8.66e-07
## week14                              4.228e-01  1.404e-01   3.011 0.002656
## week15                              7.693e-01  1.242e-01   6.196 7.98e-10
## week16                              6.896e-01  1.208e-01   5.706 1.46e-08
## week17                              8.261e-01  1.268e-01   6.512 1.09e-10
## week18                              9.045e-01  1.123e-01   8.054 1.95e-15
## week19                              1.086e+00  1.201e-01   9.046  < 2e-16
## week20                              8.511e-01  1.168e-01   7.286 5.86e-13
## week21                              1.008e+00  1.172e-01   8.601  < 2e-16
## week22                              1.028e+00  1.182e-01   8.700  < 2e-16
## week23                              8.559e-01  1.188e-01   7.207 1.02e-12
## week24                              8.810e-01  1.164e-01   7.571 7.45e-14
## week25                              9.731e-01  1.207e-01   8.060 1.86e-15
## week26                              6.809e-01  1.169e-01   5.825 7.35e-09
## week27                              7.242e-01  1.125e-01   6.437 1.78e-10
## week28                              9.685e-01  1.228e-01   7.888 6.96e-15
## week29                              8.489e-01  1.158e-01   7.328 4.32e-13
## week30                              8.732e-01  1.191e-01   7.332 4.22e-13
## week31                              8.073e-01  1.214e-01   6.648 4.55e-11
## week32                              1.066e+00  1.151e-01   9.260  < 2e-16
## week33                              1.094e+00  1.197e-01   9.139  < 2e-16
## week34                              1.091e+00  1.164e-01   9.373  < 2e-16
## week35                              9.170e-01  1.195e-01   7.676 3.43e-14
## week36                              8.843e-01  1.362e-01   6.493 1.24e-10
## week37                              8.895e-01  1.295e-01   6.870 1.04e-11
## week38                              1.206e+00  1.199e-01  10.060  < 2e-16
## week39                              1.611e+00  1.300e-01  12.397  < 2e-16
## week40                              1.212e+00  1.297e-01   9.348  < 2e-16
## week41                              1.153e+00  1.215e-01   9.495  < 2e-16
## week42                              1.113e+00  1.233e-01   9.025  < 2e-16
## week43                              8.134e-01  1.327e-01   6.128 1.21e-09
## week44                              7.107e-01  2.131e-01   3.336 0.000878
## week45                              6.047e-01  1.445e-01   4.185 3.07e-05
## week46                              1.018e+00  1.401e-01   7.272 6.47e-13
## week47                              4.365e-01  1.552e-01   2.813 0.004994
## week48                              3.516e-01  2.173e-01   1.618 0.105958
## week49                              9.751e-02  1.956e-01   0.498 0.618303
## week50                              3.581e-01  1.832e-01   1.955 0.050825
## week51                              1.327e-01  1.674e-01   0.793 0.428041
## week52                              2.271e-01  1.472e-01   1.543 0.123091
## week53                              9.133e-01  1.342e-01   6.805 1.60e-11
## adults                             -3.118e-01  8.746e-02  -3.565 0.000379
## children                            4.835e-01  8.796e-02   5.496 4.75e-08
## mealHB                              8.319e-01  2.505e-01   3.320 0.000927
## mealSC                             -4.189e-01  4.228e-02  -9.907  < 2e-16
## market_segmentDirect                1.258e-01  1.165e-01   1.080 0.280379
## market_segmentGroups               -6.241e-02  1.260e-01  -0.495 0.620573
## market_segmentOffline TA/TO        -5.789e-02  1.197e-01  -0.484 0.628753
## market_segmentOnline TA             2.349e-01  1.124e-01   2.089 0.036963
## room_typeB                         -4.657e-01  1.025e-01  -4.542 6.16e-06
## room_typeD                         -4.775e-02  2.489e-01  -0.192 0.847889
## room_typeE                          8.157e-01  8.339e-02   9.781  < 2e-16
## room_typeF                          1.032e+00  8.520e-02  12.118  < 2e-16
## room_typeG                          1.380e+00  1.048e-01  13.171  < 2e-16
```

```
## customer_typeGroup                      -4.812e-01  2.118e-01  -2.271 0.023308
## customer_typeTransient                  -5.556e-01  1.227e-01  -4.528 6.57e-06
## customer_typeTransient-Party            -3.967e-01  1.330e-01  -2.983 0.002917
## requests                                 4.164e-02  1.717e-02   2.426 0.015433
## I(lead_time^2)                           6.099e-06  9.489e-07   6.427 1.89e-10
## I(children^2)                           -1.496e-01  4.323e-02  -3.460 0.000560
## I(adults^2)                              1.413e-01  2.390e-02   5.912 4.43e-09
## mealHB:market_segmentDirect              1.338e-01  4.400e-01   0.304 0.761172
## mealSC:market_segmentDirect              3.806e-01  1.075e-01   3.541 0.000414
## mealHB:market_segmentGroups             -5.610e-01  3.531e-01  -1.589 0.112376
## mealSC:market_segmentGroups                     NA         NA      NA       NA
## mealHB:market_segmentOffline TA/TO      -9.537e-01  3.429e-01  -2.781 0.005506
## mealSC:market_segmentOffline TA/TO       1.011e-01  1.465e-01   0.690 0.490328
## mealHB:market_segmentOnline TA          -2.565e-01  3.836e-01  -0.669 0.503786
## mealSC:market_segmentOnline TA                  NA         NA      NA       NA
## mealHB:room_typeB                               NA         NA      NA       NA
## mealSC:room_typeB                               NA         NA      NA       NA
## mealHB:room_typeD                        2.548e-01  3.748e-01   0.680 0.496718
## mealSC:room_typeD                       -7.340e-01  1.772e-01  -4.143 3.68e-05
## mealHB:room_typeE                       -1.369e-01  3.721e-01  -0.368 0.712984
## mealSC:room_typeE                        7.464e-01  3.769e-01   1.980 0.047895
## mealHB:room_typeF                       -3.326e-01  4.192e-01  -0.793 0.427653
## mealSC:room_typeF                               NA         NA      NA       NA
## mealHB:room_typeG                               NA         NA      NA       NA
## mealSC:room_typeG                               NA         NA      NA       NA
## mealHB:requests                         -1.266e-01  1.579e-01  -0.802 0.422633
## mealSC:requests                          1.117e-01  3.134e-02   3.565 0.000379
## market_segmentDirect:room_typeB                 NA         NA      NA       NA
## market_segmentGroups:room_typeB                 NA         NA      NA       NA
## market_segmentOffline TA/TO:room_typeB          NA         NA      NA       NA
## market_segmentOnline TA:room_typeB              NA         NA      NA       NA
## market_segmentDirect:room_typeD          4.101e-01  2.592e-01   1.583 0.113783
## market_segmentGroups:room_typeD         -1.260e-01  2.798e-01  -0.450 0.652616
## market_segmentOffline TA/TO:room_typeD -1.493e-01  2.968e-01  -0.503 0.615051
## market_segmentOnline TA:room_typeD       3.620e-01  2.526e-01   1.433 0.152158
## market_segmentDirect:room_typeE          2.673e-02  1.207e-01   0.221 0.824747
## market_segmentGroups:room_typeE                 NA         NA      NA       NA
## market_segmentOffline TA/TO:room_typeE -2.927e-01  2.441e-01  -1.199 0.230787
## market_segmentOnline TA:room_typeE              NA         NA      NA       NA
## market_segmentDirect:room_typeF          1.946e-01  1.546e-01   1.258 0.208469
## market_segmentGroups:room_typeF                 NA         NA      NA       NA
## market_segmentOffline TA/TO:room_typeF          NA         NA      NA       NA
## market_segmentOnline TA:room_typeF              NA         NA      NA       NA
## market_segmentDirect:room_typeG          5.416e-01  1.675e-01   3.234 0.001254
## market_segmentGroups:room_typeG                 NA         NA      NA       NA
## market_segmentOffline TA/TO:room_typeG          NA         NA      NA       NA
## market_segmentOnline TA:room_typeG              NA         NA      NA       NA
##
## (Intercept)                            ***
## is_canceled1                           ***
## lead_time                              ***
## year2016                               ***
## year2017                               ***
## week2
```

```
## week3
## week4
## week5
## week6
## week7
## week8
## week9
## week10
## week11                                    *
## week12                                    **
## week13                                    ***
## week14                                    **
## week15                                    ***
## week16                                    ***
## week17                                    ***
## week18                                    ***
## week19                                    ***
## week20                                    ***
## week21                                    ***
## week22                                    ***
## week23                                    ***
## week24                                    ***
## week25                                    ***
## week26                                    ***
## week27                                    ***
## week28                                    ***
## week29                                    ***
## week30                                    ***
## week31                                    ***
## week32                                    ***
## week33                                    ***
## week34                                    ***
## week35                                    ***
## week36                                    ***
## week37                                    ***
## week38                                    ***
## week39                                    ***
## week40                                    ***
## week41                                    ***
## week42                                    ***
## week43                                    ***
## week44                                    ***
## week45                                    ***
## week46                                    ***
## week47                                    **
## week48
## week49
## week50                                    .
## week51
## week52
## week53                                    ***
## adults                                    ***
## children                                  ***
## mealHB                                    ***
```

```
## mealSC                                  ***
## market_segmentDirect
## market_segmentGroups
## market_segmentOffline TA/TO
## market_segmentOnline TA                 *
## room_typeB                              ***
## room_typeD
## room_typeE                              ***
## room_typeF                              ***
## room_typeG                              ***
## customer_typeGroup                      *
## customer_typeTransient                  ***
## customer_typeTransient-Party            **
## requests                                *
## I(lead_time^2)                          ***
## I(children^2)                           ***
## I(adults^2)                             ***
## mealHB:market_segmentDirect
## mealSC:market_segmentDirect             ***
## mealHB:market_segmentGroups
## mealSC:market_segmentGroups
## mealHB:market_segmentOffline TA/TO      **
## mealSC:market_segmentOffline TA/TO
## mealHB:market_segmentOnline TA
## mealSC:market_segmentOnline TA
## mealHB:room_typeB
## mealSC:room_typeB
## mealHB:room_typeD
## mealSC:room_typeD                       ***
## mealHB:room_typeE
## mealSC:room_typeE                       *
## mealHB:room_typeF
## mealSC:room_typeF
## mealHB:room_typeG
## mealSC:room_typeG
## mealHB:requests
## mealSC:requests                         ***
## market_segmentDirect:room_typeB
## market_segmentGroups:room_typeB
## market_segmentOffline TA/TO:room_typeB
## market_segmentOnline TA:room_typeB
## market_segmentDirect:room_typeD
## market_segmentGroups:room_typeD
## market_segmentOffline TA/TO:room_typeD
## market_segmentOnline TA:room_typeD
## market_segmentDirect:room_typeE
## market_segmentGroups:room_typeE
## market_segmentOffline TA/TO:room_typeE
## market_segmentOnline TA:room_typeE
## market_segmentDirect:room_typeF
## market_segmentGroups:room_typeF
## market_segmentOffline TA/TO:room_typeF
## market_segmentOnline TA:room_typeF
## market_segmentDirect:room_typeG         **
```

```
## market_segmentGroups:room_typeG
## market_segmentOffline TA/TO:room_typeG
## market_segmentOnline TA:room_typeG
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3733 on 1174 degrees of freedom
## Multiple R-squared:  0.8073, Adjusted R-squared:  0.7914
## F-statistic: 50.71 on 97 and 1174 DF,  p-value: < 2.2e-16
```

```r
## Training R^2
summary(model.3.2.3)$r.squared
```

```
## [1] 0.8073231
```

```r
## Training RMSE
sqrt(sum((model.3.2.3$fitted.values^(1/0.38)-train_data.new$adr)^2)/nrow(train_data.new))
```

```
## [1] 18.49513
```

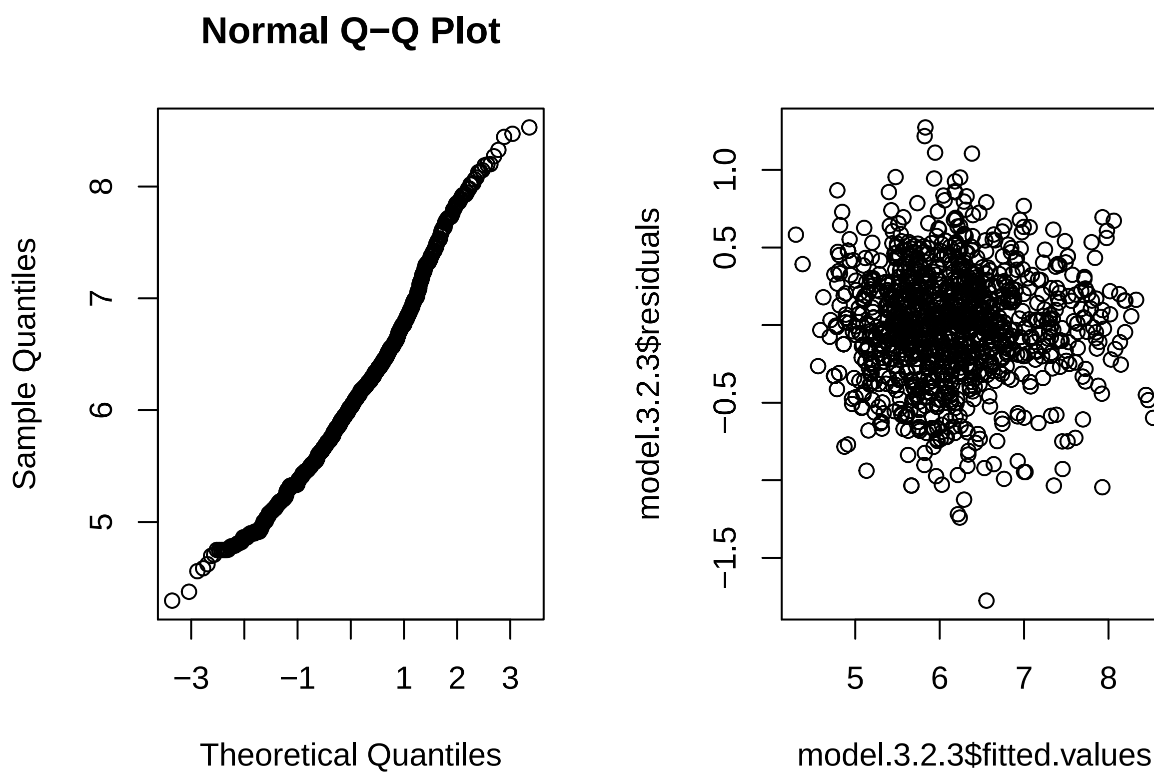
```r
## testing R^2 squared
predicted.adr = predict(model.3.2.3, newdata=test_x)^(1/0.38)
1-sum((predicted.adr-test_y)^2)/sum((test_y-mean(test_y))^2)
```

```
## [1] 0.7426901
```

```r
## testing RMSE
sqrt(sum((predicted.adr-test_y)^2)/nrow(test_data))
```

```
## [1] 22.5933
```

```r
## Generate Sec 3.2, Figure 6: diagnostic plots
par(mfrow=c(1,2))
qqnorm(model.3.2.3$fitted.values)
plot(model.3.2.3$fitted.values, model.3.2.3$residuals)
```




```
## constant variance test
bptest(model.3.2.3)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model.3.2.3
## BP = 297.23, df = 97, p-value < 2.2e-16
```

```
## normality
shapiro.test(residuals(model.3.2.3))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model.3.2.3)
## W = 0.99115, p-value = 6.206e-07
```

```
## 3.3 Random Forest
set.seed(123)
model.3.3 = randomForest(adr~.-day, train_data)

## train R^2 squared
1-sum((model.3.3$predicted-train_data$adr)^2)/sum((train_data$adr-mean(train_data$adr))^2)
```

```
## [1] 0.7550102
```

```
## Training RMSE
sqrt(sum((model.3.3$predicted-train_data$adr)^2)/nrow(train_data))
```

```
## [1] 21.5833
```

```r
## test R^2 squared
predicted.adr = predict(model.3.3, newdata=test_x)
1-sum((predicted.adr-test_y)^2)/sum((test_y-mean(test_y))^2)
```

```
## [1] 0.7245936
```

```r
## testing RMSE
sqrt(sum((predicted.adr-test_y)^2)/nrow(test_data))
```

```
## [1] 23.37429
```