Minji Song

Table of Contents

# Explantory Data Analysis

library(ggplot2)  
library(cowplot)

## Warning: package 'cowplot' was built under R version 3.6.3

##   
## \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## Note: As of version 1.0.0, cowplot does not change the

## default ggplot2 theme anymore. To recover the previous

## behavior, execute:  
## theme\_set(theme\_cowplot())

## \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

library(ggcorrplot)

## Warning: package 'ggcorrplot' was built under R version 3.6.3

library(dplyr)

## Warning: package 'dplyr' was built under R version 3.6.3

##   
## Attaching package: 'dplyr'

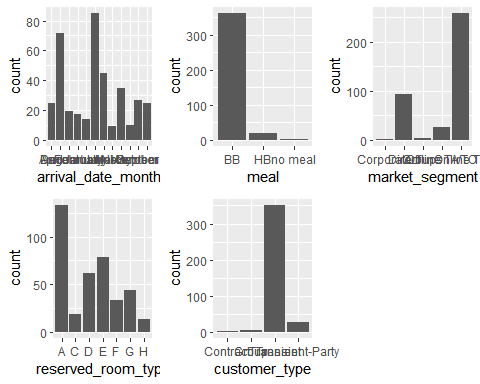
## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

data = read.csv("C:/Users/sec/Desktop/HW UIUC/SP 20/STAT 425/stat425\_fpdata.csv")  
  
data$is\_canceled <- as.factor(ifelse(data$is\_canceled==1, 'Yes', 'No'))  
  
  
data$meal = recode(data$meal, SC = 'no meal', Undefined = 'no meal')  
  
data = data[, -1]  
set.seed(2)  
trn\_idx = sample(nrow(data), size = 0.8 \* nrow(data))  
trn = data[trn\_idx, ]  
tst = data[trn\_idx, ]

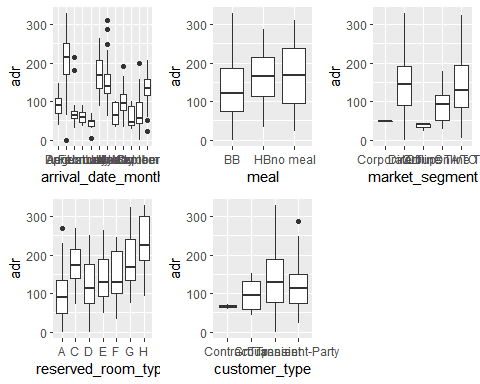
I changed some values of the is\_canceled variable and meal variable. is\_canceled variable is a categorical variable with 0 and 1, so I changed it to 0 to No and 1 to Yes to identify values easily. For meal variables, SC and Undefined have the same meaning but they are different values, thus I combined both values to name it as no meal.

plot\_grid( ggplot(trn, aes(x=arrival\_date\_month))+ geom\_bar() , ggplot(trn, aes(x=meal))+ geom\_bar() , ggplot(trn, aes(x=market\_segment))+ geom\_bar() , ggplot(trn, aes(x=reserved\_room\_type, ))+ geom\_bar(), ggplot(trn, aes(x=customer\_type, ))+ geom\_bar())



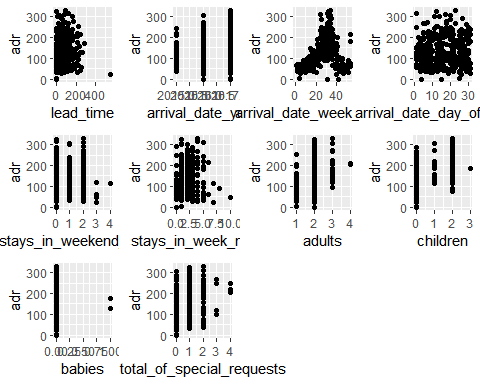
For categorical variables, I performed Explanatory Data Analysis by plotting each categorical variable to see the relationship between adr and other variables. These bar plots show the proportion of each variable. Meal and reserved room types are right-skewed and the market segment is left-skewed. There is no variable that has a perfect normal graph.

plot\_grid(ggplot(trn,aes(x=arrival\_date\_month, y=adr))+ geom\_boxplot(), ggplot(trn,aes(x=meal, y=adr))+ geom\_boxplot(), ggplot(trn,aes(x=market\_segment, y=adr))+ geom\_boxplot(), ggplot(trn,aes(x=reserved\_room\_type, y=adr))+ geom\_boxplot(), ggplot(trn,aes(x=customer\_type, y=adr))+ geom\_boxplot())



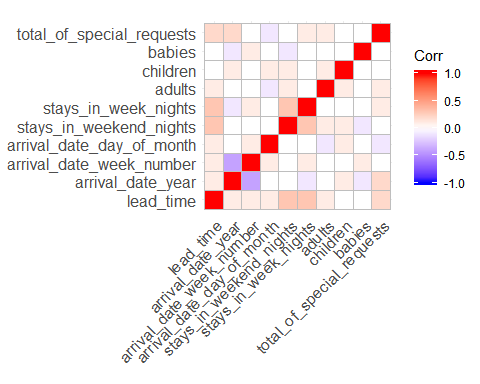
For the arrival date month variable, we can see that August is associated with high adr, and January has low adr. Reserved room type H has a higher adr with about 180 adr. Other variables in this plot have similar trend.

plot\_grid(ggplot(trn,aes(x=lead\_time, y=adr))+ geom\_point(),  
ggplot(trn,aes(x=arrival\_date\_year, y=adr))+ geom\_point(),  
ggplot(trn,aes(x=arrival\_date\_week\_number, y=adr))+ geom\_point(),  
ggplot(trn,aes(x=arrival\_date\_day\_of\_month, y=adr))+ geom\_point(),  
ggplot(trn,aes(x=stays\_in\_weekend\_nights, y=adr))+ geom\_point(),  
ggplot(trn,aes(x=stays\_in\_week\_nights, y=adr))+ geom\_point(),  
ggplot(trn,aes(x=adults, y=adr))+ geom\_point(),  
ggplot(trn,aes(x=children, y=adr))+ geom\_point(),  
ggplot(trn,aes(x=babies, y=adr))+ geom\_point(),  
ggplot(trn,aes(x=total\_of\_special\_requests, y=adr))+ geom\_point())



It seems lead time and arrival date week numer have trend in each plot so we might think the relationship between the two.

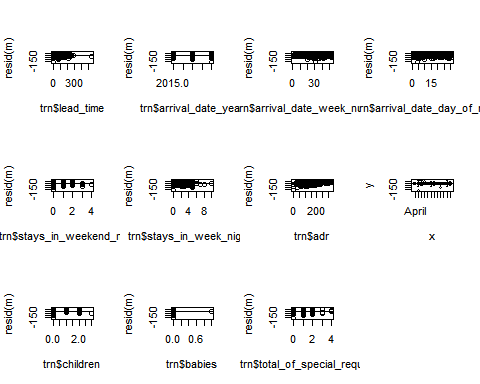
correl = round(cor(trn[,c("lead\_time", "arrival\_date\_year", "arrival\_date\_week\_number", "arrival\_date\_day\_of\_month", "stays\_in\_weekend\_nights", "stays\_in\_week\_nights", "adults", "children", "babies", "total\_of\_special\_requests")]),1)  
  
ggcorrplot(correl)



we see that arrival date week number and arrival date year tend to have a negative correlation more than other variables.

m = lm(adr~. , trn)

par(mfrow=c(3,4))  
  
plot(trn$lead\_time ,resid(m))  
abline(0,0)  
plot(trn$arrival\_date\_year ,resid(m))  
abline(0,0)  
plot(trn$arrival\_date\_week\_number ,resid(m))  
abline(0,0)  
plot(trn$arrival\_date\_day\_of\_month,resid(m))  
abline(0,0)  
plot(trn$stays\_in\_weekend\_nights,resid(m))  
abline(0,0)  
plot(trn$stays\_in\_week\_nights,resid(m))  
abline(0,0)  
plot(trn$adr,resid(m))  
abline(0,0)  
plot(trn$arrival\_date\_month ,resid(m))  
abline(0,0)  
plot(trn$children ,resid(m))  
abline(0,0)  
plot(trn$babies ,resid(m))  
abline(0,0)  
plot(trn$total\_of\_special\_requests ,resid(m))  
abline(0,0)



All plots are well distributed and there is no clear trend. Even though lead time plot looks like to be a heteroscedasticity a bit in a little plot, it is distributed well enough. From the residual plots, none of the numerical variables support the nonlinear trend.

# Linear Regression

m = lm(adr~. +lead\_time\*arrival\_date\_week\_number,trn)  
summary(m)

##   
## Call:  
## lm(formula = adr ~ . + lead\_time \* arrival\_date\_week\_number,   
## data = trn)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -190.04 -14.98 -0.91 14.74 79.58   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -4.508e+04 6.685e+03 -6.742 6.59e-11 \*\*\*  
## is\_canceledYes 1.691e+01 4.540e+00 3.725 0.000228 \*\*\*  
## lead\_time -2.042e-02 7.980e-02 -0.256 0.798136   
## arrival\_date\_year 2.234e+01 3.300e+00 6.769 5.60e-11 \*\*\*  
## arrival\_date\_monthAugust 8.902e+01 7.010e+01 1.270 0.205008   
## arrival\_date\_monthDecember -5.809e+01 1.412e+02 -0.411 0.681046   
## arrival\_date\_monthFebruary -1.789e+01 3.589e+01 -0.498 0.618549   
## arrival\_date\_monthJanuary -3.416e+00 5.341e+01 -0.064 0.949043   
## arrival\_date\_monthJuly 6.395e+01 5.246e+01 1.219 0.223629   
## arrival\_date\_monthJune 3.771e+01 3.575e+01 1.055 0.292212   
## arrival\_date\_monthMarch -1.844e+01 2.178e+01 -0.847 0.397729   
## arrival\_date\_monthMay -1.841e+00 1.897e+01 -0.097 0.922748   
## arrival\_date\_monthNovember -5.433e+01 1.236e+02 -0.440 0.660473   
## arrival\_date\_monthOctober -3.343e+01 1.062e+02 -0.315 0.753034   
## arrival\_date\_monthSeptember 3.147e+01 8.830e+01 0.356 0.721747   
## arrival\_date\_week\_number 1.980e+00 4.055e+00 0.488 0.625596   
## arrival\_date\_day\_of\_month 2.021e-01 6.072e-01 0.333 0.739484   
## stays\_in\_weekend\_nights 8.356e-01 2.071e+00 0.403 0.686859   
## stays\_in\_week\_nights 2.667e+00 1.245e+00 2.142 0.032913 \*   
## adults 1.714e+01 4.016e+00 4.267 2.57e-05 \*\*\*  
## children 1.466e+01 3.426e+00 4.279 2.44e-05 \*\*\*  
## babies -3.037e+01 2.160e+01 -1.406 0.160647   
## mealHB 2.457e+01 7.410e+00 3.315 0.001013 \*\*   
## mealno meal 4.663e+01 2.481e+01 1.879 0.061052 .   
## market\_segmentDirect 2.316e+01 2.353e+01 0.984 0.325674   
## market\_segmentGroups -1.500e+01 2.928e+01 -0.512 0.608763   
## market\_segmentOffline TA/TO -1.225e+00 2.337e+01 -0.052 0.958231   
## market\_segmentOnline TA 1.470e+01 2.329e+01 0.631 0.528179   
## reserved\_room\_typeC 2.559e+01 8.523e+00 3.002 0.002876 \*\*   
## reserved\_room\_typeD 1.734e+01 4.584e+00 3.782 0.000184 \*\*\*  
## reserved\_room\_typeE 3.683e+01 4.493e+00 8.196 4.97e-15 \*\*\*  
## reserved\_room\_typeF 4.863e+01 6.202e+00 7.841 5.65e-14 \*\*\*  
## reserved\_room\_typeG 5.592e+01 6.545e+00 8.545 4.24e-16 \*\*\*  
## reserved\_room\_typeH 6.380e+01 9.802e+00 6.509 2.67e-10 \*\*\*  
## customer\_typeGroup -1.602e+01 2.713e+01 -0.591 0.555216   
## customer\_typeTransient 4.838e+00 2.295e+01 0.211 0.833188   
## customer\_typeTransient-Party 1.965e+01 2.367e+01 0.830 0.407006   
## total\_of\_special\_requests 4.801e+00 2.183e+00 2.199 0.028530 \*   
## lead\_time:arrival\_date\_week\_number -5.599e-03 2.619e-03 -2.138 0.033197 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 28.68 on 344 degrees of freedom  
## Multiple R-squared: 0.8472, Adjusted R-squared: 0.8304   
## F-statistic: 50.21 on 38 and 344 DF, p-value: < 2.2e-16

I performed linear regression using all variables and interaction terms with lead time and arrival date week numbers. In a summary, the p-value of interaction term is lower than 0.05, so we keep this significant variable in the model.

new\_mod = step(m, direction="both")

## Start: AIC=2607.74  
## adr ~ is\_canceled + lead\_time + arrival\_date\_year + arrival\_date\_month +   
## arrival\_date\_week\_number + arrival\_date\_day\_of\_month + stays\_in\_weekend\_nights +   
## stays\_in\_week\_nights + adults + children + babies + meal +   
## market\_segment + reserved\_room\_type + customer\_type + total\_of\_special\_requests +   
## lead\_time \* arrival\_date\_week\_number  
##   
## Df Sum of Sq RSS AIC  
## - arrival\_date\_day\_of\_month 1 91 283063 2605.9  
## - stays\_in\_weekend\_nights 1 134 283106 2605.9  
## <none> 282972 2607.7  
## - babies 1 1626 284598 2607.9  
## - customer\_type 3 5088 288060 2608.6  
## - lead\_time:arrival\_date\_week\_number 1 3761 286733 2610.8  
## - stays\_in\_week\_nights 1 3773 286746 2610.8  
## - total\_of\_special\_requests 1 3978 286950 2611.1  
## - market\_segment 4 10612 293585 2613.8  
## - meal 2 11407 294379 2618.9  
## - is\_canceled 1 11415 294387 2620.9  
## - adults 1 14974 297946 2625.5  
## - children 1 15061 298033 2625.6  
## - arrival\_date\_year 1 37694 320666 2653.6  
## - reserved\_room\_type 6 121770 404742 2732.8  
## - arrival\_date\_month 11 688128 971100 3058.0  
##   
## Step: AIC=2605.86  
## adr ~ is\_canceled + lead\_time + arrival\_date\_year + arrival\_date\_month +   
## arrival\_date\_week\_number + stays\_in\_weekend\_nights + stays\_in\_week\_nights +   
## adults + children + babies + meal + market\_segment + reserved\_room\_type +   
## customer\_type + total\_of\_special\_requests + lead\_time:arrival\_date\_week\_number  
##   
## Df Sum of Sq RSS AIC  
## - stays\_in\_weekend\_nights 1 107 283171 2604.0  
## <none> 283063 2605.9  
## - babies 1 1660 284723 2606.1  
## - customer\_type 3 5042 288106 2606.6  
## + arrival\_date\_day\_of\_month 1 91 282972 2607.7  
## - lead\_time:arrival\_date\_week\_number 1 3940 287003 2609.2  
## - stays\_in\_week\_nights 1 4011 287074 2609.2  
## - total\_of\_special\_requests 1 4095 287158 2609.4  
## - market\_segment 4 10535 293599 2611.9  
## - meal 2 11439 294502 2617.0  
## - is\_canceled 1 11448 294511 2619.1  
## - adults 1 14927 297991 2623.6  
## - children 1 15626 298689 2624.4  
## - arrival\_date\_year 1 59410 342473 2676.8  
## - reserved\_room\_type 6 121735 404798 2730.9  
## - arrival\_date\_month 11 689062 972125 3056.4  
##   
## Step: AIC=2604.01  
## adr ~ is\_canceled + lead\_time + arrival\_date\_year + arrival\_date\_month +   
## arrival\_date\_week\_number + stays\_in\_week\_nights + adults +   
## children + babies + meal + market\_segment + reserved\_room\_type +   
## customer\_type + total\_of\_special\_requests + lead\_time:arrival\_date\_week\_number  
##   
## Df Sum of Sq RSS AIC  
## <none> 283171 2604.0  
## - babies 1 1719 284890 2604.3  
## - customer\_type 3 4938 288109 2604.6  
## + stays\_in\_weekend\_nights 1 107 283063 2605.9  
## + arrival\_date\_day\_of\_month 1 65 283106 2605.9  
## - lead\_time:arrival\_date\_week\_number 1 3903 287074 2607.2  
## - total\_of\_special\_requests 1 4036 287207 2607.4  
## - stays\_in\_week\_nights 1 4448 287618 2608.0  
## - market\_segment 4 10551 293721 2610.0  
## - meal 2 11346 294516 2615.1  
## - is\_canceled 1 11416 294586 2617.2  
## - adults 1 15278 298448 2622.1  
## - children 1 15837 299008 2622.8  
## - arrival\_date\_year 1 59403 342574 2674.9  
## - reserved\_room\_type 6 122558 405729 2729.8  
## - arrival\_date\_month 11 689087 972257 3054.5

summary(new\_mod)

##   
## Call:  
## lm(formula = adr ~ is\_canceled + lead\_time + arrival\_date\_year +   
## arrival\_date\_month + arrival\_date\_week\_number + stays\_in\_week\_nights +   
## adults + children + babies + meal + market\_segment + reserved\_room\_type +   
## customer\_type + total\_of\_special\_requests + lead\_time:arrival\_date\_week\_number,   
## data = trn)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -190.055 -14.856 -0.897 14.997 79.381   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -4.620e+04 5.419e+03 -8.526 4.78e-16 \*\*\*  
## is\_canceledYes 1.691e+01 4.527e+00 3.735 0.000220 \*\*\*  
## lead\_time -1.731e-02 7.934e-02 -0.218 0.827428   
## arrival\_date\_year 2.288e+01 2.686e+00 8.520 5.00e-16 \*\*\*  
## arrival\_date\_monthAugust 6.672e+01 2.139e+01 3.119 0.001968 \*\*   
## arrival\_date\_monthDecember -1.035e+02 4.160e+01 -2.488 0.013327 \*   
## arrival\_date\_monthFebruary -6.267e+00 1.343e+01 -0.467 0.641046   
## arrival\_date\_monthJanuary 1.362e+01 1.803e+01 0.756 0.450416   
## arrival\_date\_monthJuly 4.726e+01 1.622e+01 2.914 0.003797 \*\*   
## arrival\_date\_monthJune 2.654e+01 1.210e+01 2.193 0.028986 \*   
## arrival\_date\_monthMarch -1.205e+01 1.302e+01 -0.925 0.355363   
## arrival\_date\_monthMay -6.976e+00 9.356e+00 -0.746 0.456390   
## arrival\_date\_monthNovember -9.390e+01 3.621e+01 -2.593 0.009919 \*\*   
## arrival\_date\_monthOctober -6.771e+01 3.133e+01 -2.161 0.031377 \*   
## arrival\_date\_monthSeptember 3.017e+00 2.650e+01 0.114 0.909397   
## arrival\_date\_week\_number 3.287e+00 1.164e+00 2.824 0.005016 \*\*   
## stays\_in\_week\_nights 2.812e+00 1.206e+00 2.331 0.020318 \*   
## adults 1.723e+01 3.989e+00 4.321 2.04e-05 \*\*\*  
## children 1.488e+01 3.383e+00 4.399 1.45e-05 \*\*\*  
## babies -3.115e+01 2.149e+01 -1.449 0.148147   
## mealHB 2.425e+01 7.352e+00 3.298 0.001074 \*\*   
## mealno meal 4.768e+01 2.466e+01 1.934 0.053955 .   
## market\_segmentDirect 2.232e+01 2.334e+01 0.956 0.339599   
## market\_segmentGroups -1.591e+01 2.911e+01 -0.546 0.585082   
## market\_segmentOffline TA/TO -1.902e+00 2.316e+01 -0.082 0.934606   
## market\_segmentOnline TA 1.385e+01 2.309e+01 0.600 0.548916   
## reserved\_room\_typeC 2.526e+01 8.476e+00 2.981 0.003079 \*\*   
## reserved\_room\_typeD 1.745e+01 4.566e+00 3.821 0.000157 \*\*\*  
## reserved\_room\_typeE 3.701e+01 4.465e+00 8.290 2.54e-15 \*\*\*  
## reserved\_room\_typeF 4.848e+01 6.177e+00 7.849 5.29e-14 \*\*\*  
## reserved\_room\_typeG 5.600e+01 6.518e+00 8.591 2.99e-16 \*\*\*  
## reserved\_room\_typeH 6.373e+01 9.745e+00 6.539 2.22e-10 \*\*\*  
## customer\_typeGroup -1.745e+01 2.688e+01 -0.649 0.516662   
## customer\_typeTransient 3.476e+00 2.271e+01 0.153 0.878451   
## customer\_typeTransient-Party 1.782e+01 2.328e+01 0.766 0.444344   
## total\_of\_special\_requests 4.814e+00 2.168e+00 2.221 0.027016 \*   
## lead\_time:arrival\_date\_week\_number -5.667e-03 2.595e-03 -2.184 0.029643 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 28.61 on 346 degrees of freedom  
## Multiple R-squared: 0.8471, Adjusted R-squared: 0.8312   
## F-statistic: 53.26 on 36 and 346 DF, p-value: < 2.2e-16

I performed the AIC model selection to get rid of some useless variables based on the small AIC value. Through the selection, the final model has 14 variables: is canceled, lead time, arrival date year, arrival date month, arrival date week number, stay weeknights, adults, children, meal, market segment, reserved room type, customer type, a total of special requests and the interaction terms.

library(faraway)  
vif(new\_mod)

## is\_canceledYes lead\_time   
## 1.301232 13.944653   
## arrival\_date\_year arrival\_date\_monthAugust   
## 1.533816 32.694813   
## arrival\_date\_monthDecember arrival\_date\_monthFebruary   
## 38.192252 3.580339   
## arrival\_date\_monthJanuary arrival\_date\_monthJuly   
## 5.355276 21.251669   
## arrival\_date\_monthJune arrival\_date\_monthMarch   
## 7.110077 1.820790   
## arrival\_date\_monthMay arrival\_date\_monthNovember   
## 3.401439 15.604355   
## arrival\_date\_monthOctober arrival\_date\_monthSeptember   
## 30.100962 20.045836   
## arrival\_date\_week\_number stays\_in\_week\_nights   
## 81.384677 1.405875   
## adults children   
## 1.216240 2.304569   
## babies mealHB   
## 1.122733 1.251979   
## mealno meal market\_segmentDirect   
## 1.477858 46.868723   
## market\_segmentGroups market\_segmentOffline TA/TO   
## 3.081207 16.449344   
## market\_segmentOnline TA reserved\_room\_typeC   
## 54.858598 1.585010   
## reserved\_room\_typeD reserved\_room\_typeE   
## 1.323983 1.527565   
## reserved\_room\_typeF reserved\_room\_typeG   
## 1.405789 2.021917   
## reserved\_room\_typeH customer\_typeGroup   
## 1.457283 3.495078   
## customer\_typeTransient customer\_typeTransient-Party   
## 18.484308 16.043321   
## total\_of\_special\_requests lead\_time:arrival\_date\_week\_number   
## 1.409475 15.867242

new\_mod = lm(formula = adr ~ is\_canceled + lead\_time + arrival\_date\_year + arrival\_date\_month + arrival\_date\_week\_number + stays\_in\_week\_nights + adults + children + babies + meal + market\_segment + reserved\_room\_type + customer\_type + total\_of\_special\_requests + lead\_time:arrival\_date\_week\_number, data = trn)

Before we confirm the final model, I performed the VIF test to see if there is any multicollinearity with this model. It seems like there is multicollinearity in this model because some values have high VIF values. The highest VIF is arrival date week number with 81.3846, there are few more variables that have over 10 VIFs. However, those variables that have high VIF values can be ignored since those are either categorical variables with more than two levels or part of interaction terms. Other than those variables, all other variables look good.

test = trn[c(1,1,1),]  
test$arrival\_date\_week\_number= c(min(trn$arrival\_date\_week\_number), median(trn$arrival\_date\_week\_number), max(trn$arrival\_date\_week\_number))  
test

## is\_canceled lead\_time arrival\_date\_year arrival\_date\_month  
## 341 No 65 2017 February  
## 341.1 No 65 2017 February  
## 341.2 No 65 2017 February  
## arrival\_date\_week\_number arrival\_date\_day\_of\_month  
## 341 1 16  
## 341.1 29 16  
## 341.2 53 16  
## stays\_in\_weekend\_nights stays\_in\_week\_nights adults children babies meal  
## 341 2 3 2 0 0 BB  
## 341.1 2 3 2 0 0 BB  
## 341.2 2 3 2 0 0 BB  
## market\_segment reserved\_room\_type customer\_type adr  
## 341 Online TA A Transient 48  
## 341.1 Online TA A Transient 48  
## 341.2 Online TA A Transient 48  
## total\_of\_special\_requests  
## 341 1  
## 341.1 1  
## 341.2 1

predict(new\_mod, test)

## 341 341.1 341.2   
## 21.73755 103.46875 173.52406

A large number of arrival date week number has a higher average daily rate than a small number of arrival date week number. When the arrival date week number is 1, 29 and 53, the predicted adr are 21.74, 103.47, and 173.52.

test = trn[c(1,1,1),]  
test$lead\_time= c(min(trn$lead\_time), median(trn$lead\_time), max(trn$lead\_time))  
test

## is\_canceled lead\_time arrival\_date\_year arrival\_date\_month  
## 341 No 0 2017 February  
## 341.1 No 36 2017 February  
## 341.2 No 542 2017 February  
## arrival\_date\_week\_number arrival\_date\_day\_of\_month  
## 341 7 16  
## 341.1 7 16  
## 341.2 7 16  
## stays\_in\_weekend\_nights stays\_in\_week\_nights adults children babies meal  
## 341 2 3 2 0 0 BB  
## 341.1 2 3 2 0 0 BB  
## 341.2 2 3 2 0 0 BB  
## market\_segment reserved\_room\_type customer\_type adr  
## 341 Online TA A Transient 48  
## 341.1 Online TA A Transient 48  
## 341.2 Online TA A Transient 48  
## total\_of\_special\_requests  
## 341 1  
## 341.1 1  
## 341.2 1

predict(new\_mod, test)

## 341 341.1 341.2   
## 42.95474 40.90365 12.07437

The shorter lead time has a higher average daily rate by 30% than 537 days longer lead time. The shorter time between booking date and arrival date have a higher rate.

test = trn[c(1,1,1),]  
test$arrival\_date\_year = c(min(trn$arrival\_date\_year ), median(trn$arrival\_date\_year ), max(trn$arrival\_date\_year ))  
test

## is\_canceled lead\_time arrival\_date\_year arrival\_date\_month  
## 341 No 65 2015 February  
## 341.1 No 65 2016 February  
## 341.2 No 65 2017 February  
## arrival\_date\_week\_number arrival\_date\_day\_of\_month  
## 341 7 16  
## 341.1 7 16  
## 341.2 7 16  
## stays\_in\_weekend\_nights stays\_in\_week\_nights adults children babies meal  
## 341 2 3 2 0 0 BB  
## 341.1 2 3 2 0 0 BB  
## 341.2 2 3 2 0 0 BB  
## market\_segment reserved\_room\_type customer\_type adr  
## 341 Online TA A Transient 48  
## 341.1 Online TA A Transient 48  
## 341.2 Online TA A Transient 48  
## total\_of\_special\_requests  
## 341 1  
## 341.1 1  
## 341.2 1

predict(new\_mod, test)

## 341 341.1 341.2   
## -6.517773 16.366801 39.251375

This data set is for between 2015 and 2017. In 2015, the average daily rate is much lower than in 2017. In 2017, the rate has increased by about 45% compared to 2015 and it was -6.5 in 2015.

test = trn[c(1,1,1),]  
test$stays\_in\_week\_nights = c(min(trn$stays\_in\_week\_nights ), median(trn$stays\_in\_week\_nights ), max(trn$stays\_in\_week\_nights ))  
test

## is\_canceled lead\_time arrival\_date\_year arrival\_date\_month  
## 341 No 65 2017 February  
## 341.1 No 65 2017 February  
## 341.2 No 65 2017 February  
## arrival\_date\_week\_number arrival\_date\_day\_of\_month  
## 341 7 16  
## 341.1 7 16  
## 341.2 7 16  
## stays\_in\_weekend\_nights stays\_in\_week\_nights adults children babies meal  
## 341 2 0 2 0 0 BB  
## 341.1 2 2 2 0 0 BB  
## 341.2 2 10 2 0 0 BB  
## market\_segment reserved\_room\_type customer\_type adr  
## 341 Online TA A Transient 48  
## 341.1 Online TA A Transient 48  
## 341.2 Online TA A Transient 48  
## total\_of\_special\_requests  
## 341 1  
## 341.1 1  
## 341.2 1

predict(new\_mod, test)

## 341 341.1 341.2   
## 30.81557 36.43944 58.93493

The more nights staying in a week increases the average daily rate. The difference between 0 nights to 10 nights is 28.1% .

test = trn[c(1,1,1),]  
test$adults = c(min(trn$adults ), median(trn$adults ), max(trn$adults ))  
test

## is\_canceled lead\_time arrival\_date\_year arrival\_date\_month  
## 341 No 65 2017 February  
## 341.1 No 65 2017 February  
## 341.2 No 65 2017 February  
## arrival\_date\_week\_number arrival\_date\_day\_of\_month  
## 341 7 16  
## 341.1 7 16  
## 341.2 7 16  
## stays\_in\_weekend\_nights stays\_in\_week\_nights adults children babies meal  
## 341 2 3 1 0 0 BB  
## 341.1 2 3 2 0 0 BB  
## 341.2 2 3 4 0 0 BB  
## market\_segment reserved\_room\_type customer\_type adr  
## 341 Online TA A Transient 48  
## 341.1 Online TA A Transient 48  
## 341.2 Online TA A Transient 48  
## total\_of\_special\_requests  
## 341 1  
## 341.1 1  
## 341.2 1

predict(new\_mod, test)

## 341 341.1 341.2   
## 22.01702 39.25137 73.72007

test = trn[c(1,1,1),]  
test$children = c(min(trn$children ), median(trn$children ), max(trn$children ))  
test

## is\_canceled lead\_time arrival\_date\_year arrival\_date\_month  
## 341 No 65 2017 February  
## 341.1 No 65 2017 February  
## 341.2 No 65 2017 February  
## arrival\_date\_week\_number arrival\_date\_day\_of\_month  
## 341 7 16  
## 341.1 7 16  
## 341.2 7 16  
## stays\_in\_weekend\_nights stays\_in\_week\_nights adults children babies meal  
## 341 2 3 2 0 0 BB  
## 341.1 2 3 2 0 0 BB  
## 341.2 2 3 2 3 0 BB  
## market\_segment reserved\_room\_type customer\_type adr  
## 341 Online TA A Transient 48  
## 341.1 Online TA A Transient 48  
## 341.2 Online TA A Transient 48  
## total\_of\_special\_requests  
## 341 1  
## 341.1 1  
## 341.2 1

predict(new\_mod, test)

## 341 341.1 341.2   
## 39.25137 39.25137 83.89169

More adults and children also increase the average daily rate.

test = trn[c(1,1,1),]  
test$total\_of\_special\_requests = c(min(trn$total\_of\_special\_requests), median(trn$total\_of\_special\_requests), max(trn$total\_of\_special\_requests))  
test

## is\_canceled lead\_time arrival\_date\_year arrival\_date\_month  
## 341 No 65 2017 February  
## 341.1 No 65 2017 February  
## 341.2 No 65 2017 February  
## arrival\_date\_week\_number arrival\_date\_day\_of\_month  
## 341 7 16  
## 341.1 7 16  
## 341.2 7 16  
## stays\_in\_weekend\_nights stays\_in\_week\_nights adults children babies meal  
## 341 2 3 2 0 0 BB  
## 341.1 2 3 2 0 0 BB  
## 341.2 2 3 2 0 0 BB  
## market\_segment reserved\_room\_type customer\_type adr  
## 341 Online TA A Transient 48  
## 341.1 Online TA A Transient 48  
## 341.2 Online TA A Transient 48  
## total\_of\_special\_requests  
## 341 0  
## 341.1 1  
## 341.2 4

predict(new\_mod, test)

## 341 341.1 341.2   
## 34.43697 39.25137 53.69459

a lot of special requests increase the daily average rate. A special request increases the rate by about 4.5%.

# Regression Tree

library(rpart)

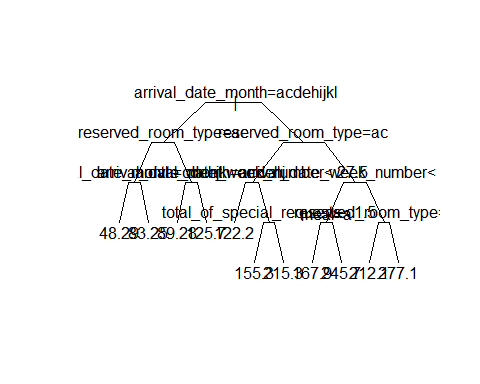
##   
## Attaching package: 'rpart'

## The following object is masked from 'package:faraway':  
##   
## solder

rpmod = rpart(adr ~ is\_canceled + lead\_time + arrival\_date\_year + arrival\_date\_month + arrival\_date\_week\_number + stays\_in\_week\_nights + adults + children + babies + meal + market\_segment + reserved\_room\_type + customer\_type + total\_of\_special\_requests,cp=0.01, data = trn)  
rpmod

## n= 383   
##   
## node), split, n, deviance, yval  
## \* denotes terminal node  
##   
## 1) root 383 1852467.00 134.49170   
## 2) arrival\_date\_month=April,December,February,January,March,May,November,October,September 181 310527.70 84.19547   
## 4) reserved\_room\_type=A,D 94 85575.85 62.04979   
## 8) arrival\_date\_month=December,February,January,March,November,October 57 19266.29 48.28842 \*  
## 9) arrival\_date\_month=April,May,September 37 38885.98 83.24973 \*  
## 5) reserved\_room\_type=C,E,F,G,H 87 129041.60 108.12300   
## 10) arrival\_date\_month=April,December,February,January,March,November 42 44756.32 89.28357 \*  
## 11) arrival\_date\_month=May,October,September 45 55465.44 125.70640 \*  
## 3) arrival\_date\_month=August,July,June 202 673784.50 179.55920   
## 6) reserved\_room\_type=A,D 101 240234.20 152.31550   
## 12) arrival\_date\_week\_number< 27.5 29 21127.08 122.20520 \*  
## 13) arrival\_date\_week\_number>=27.5 72 182224.70 164.44330   
## 26) total\_of\_special\_requests< 1.5 61 134380.00 155.26590 \*  
## 27) total\_of\_special\_requests>=1.5 11 14215.91 215.33640 \*  
## 7) reserved\_room\_type=C,E,F,G,H 101 283623.00 206.80280   
## 14) arrival\_date\_week\_number< 29.5 48 108385.30 179.26150   
## 28) meal=BB 41 55722.41 167.91590 \*  
## 29) meal=HB,no meal 7 16473.43 245.71430 \*  
## 15) arrival\_date\_week\_number>=29.5 53 105854.20 231.74580   
## 30) reserved\_room\_type=C,E,F 37 39411.82 212.12700 \*  
## 31) reserved\_room\_type=G,H 16 19268.31 277.11440 \*

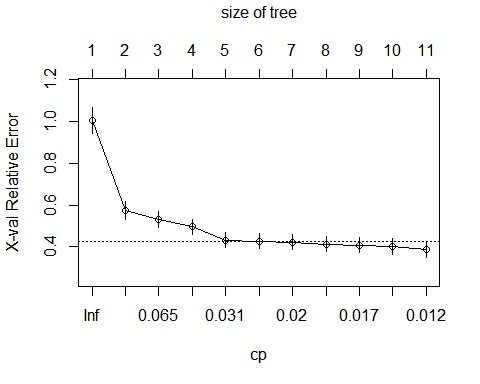
plot(rpmod,compress=T,uniform=T,branch=0.4,margin=.10)  
text(rpmod)



printcp(rpmod)

##   
## Regression tree:  
## rpart(formula = adr ~ is\_canceled + lead\_time + arrival\_date\_year +   
## arrival\_date\_month + arrival\_date\_week\_number + stays\_in\_week\_nights +   
## adults + children + babies + meal + market\_segment + reserved\_room\_type +   
## customer\_type + total\_of\_special\_requests, data = trn, cp = 0.01)  
##   
## Variables actually used in tree construction:  
## [1] arrival\_date\_month arrival\_date\_week\_number   
## [3] meal reserved\_room\_type   
## [5] total\_of\_special\_requests  
##   
## Root node error: 1852467/383 = 4836.7  
##   
## n= 383   
##   
## CP nsplit rel error xerror xstd  
## 1 0.468648 0 1.00000 1.00596 0.062974  
## 2 0.080934 1 0.53135 0.57566 0.043383  
## 3 0.051774 2 0.45042 0.53109 0.036354  
## 4 0.037455 3 0.39864 0.49549 0.035560  
## 5 0.025466 4 0.36119 0.43074 0.035600  
## 6 0.019910 5 0.33572 0.42619 0.035552  
## 7 0.019536 6 0.31581 0.42262 0.035522  
## 8 0.018154 7 0.29628 0.41243 0.035487  
## 9 0.015558 8 0.27812 0.40657 0.035933  
## 10 0.014804 9 0.26257 0.40179 0.039373  
## 11 0.010000 10 0.24776 0.38747 0.038920

plotcp(rpmod)

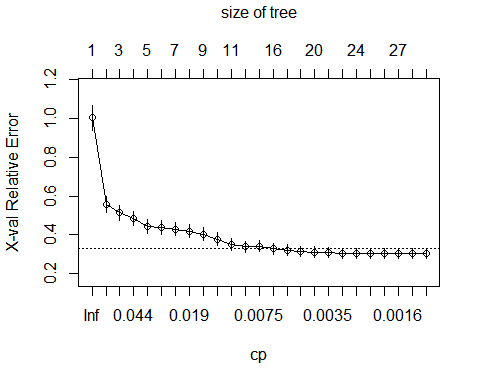


This regression split the model into several steps and the first node of the variable we split on is arrival date month with 181 observations. The second branch is also an arrival date month with 202 observations. It is telling that the most important variable in this regression is the arrival date month.

rpmods = rpart(adr ~ is\_canceled + lead\_time + arrival\_date\_year + arrival\_date\_month + arrival\_date\_week\_number + stays\_in\_week\_nights + adults + children + babies + meal + market\_segment + reserved\_room\_type + customer\_type + total\_of\_special\_requests,   
 cp=0.001, data = trn)  
  
printcp(rpmods)

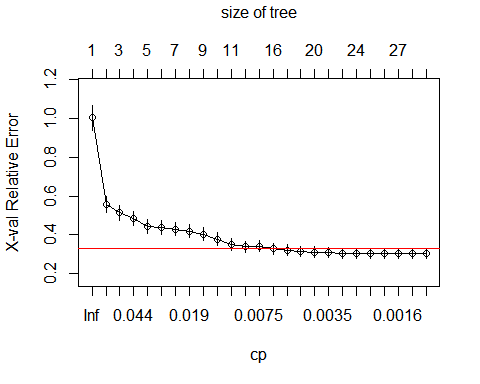
##   
## Regression tree:  
## rpart(formula = adr ~ is\_canceled + lead\_time + arrival\_date\_year +   
## arrival\_date\_month + arrival\_date\_week\_number + stays\_in\_week\_nights +   
## adults + children + babies + meal + market\_segment + reserved\_room\_type +   
## customer\_type + total\_of\_special\_requests, data = trn, cp = 0.001)  
##   
## Variables actually used in tree construction:  
## [1] adults arrival\_date\_month   
## [3] arrival\_date\_week\_number arrival\_date\_year   
## [5] children lead\_time   
## [7] market\_segment meal   
## [9] reserved\_room\_type total\_of\_special\_requests  
##   
## Root node error: 1852467/383 = 4836.7  
##   
## n= 383   
##   
## CP nsplit rel error xerror xstd  
## 1 0.4686479 0 1.00000 1.00218 0.062561  
## 2 0.0809339 1 0.53135 0.55649 0.042265  
## 3 0.0517744 2 0.45042 0.51285 0.037279  
## 4 0.0374546 3 0.39864 0.48359 0.036853  
## 5 0.0254655 4 0.36119 0.44158 0.035134  
## 6 0.0199099 5 0.33572 0.43804 0.034193  
## 7 0.0195358 6 0.31581 0.42923 0.034723  
## 8 0.0181535 7 0.29628 0.41922 0.034290  
## 9 0.0155575 8 0.27812 0.40433 0.034013  
## 10 0.0148038 9 0.26257 0.37839 0.033289  
## 11 0.0096474 10 0.24776 0.34935 0.030431  
## 12 0.0085356 13 0.21882 0.33808 0.028171  
## 13 0.0066706 14 0.21029 0.34112 0.028474  
## 14 0.0058331 15 0.20361 0.32790 0.028353  
## 15 0.0053710 16 0.19778 0.32138 0.027719  
## 16 0.0047816 17 0.19241 0.31391 0.027523  
## 17 0.0035575 19 0.18285 0.31140 0.026983  
## 18 0.0033983 20 0.17929 0.30934 0.026991  
## 19 0.0030461 21 0.17589 0.30600 0.026059  
## 20 0.0028928 23 0.16980 0.30418 0.025887  
## 21 0.0020060 24 0.16691 0.30366 0.026201  
## 22 0.0018369 25 0.16490 0.30212 0.025980  
## 23 0.0014659 26 0.16306 0.30227 0.025998  
## 24 0.0011027 27 0.16160 0.30364 0.025946  
## 25 0.0010000 28 0.16049 0.30186 0.025946

plotcp(rpmods)



We can fit a bigger tree by pruning the tree using cp=0.001 instead of cp=0.01.

tb=rpmods$cptable  
id.min = which.min(tb[, 'xerror'])  
err = tb[id.min,'xerror'] +tb[id.min, 'xstd']  
plotcp(rpmods)  
abline(h=err, col="red")

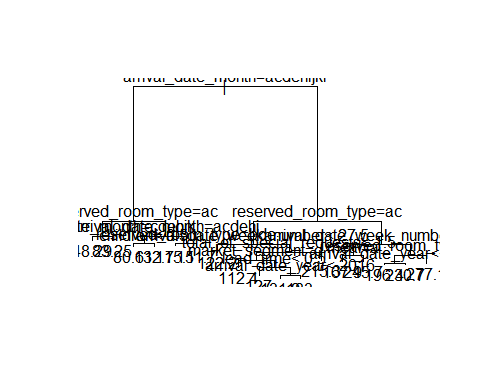


I chose smallest tree size with CV error within 1 standard error of the smallest CV error to determine the optimal tree size and the red line of the plot indicates the CV error within 1 standard error of the smallest CV error.

id = min(which(tb[, 'xerror'] < err))  
cp = (tb[id, 'CP'] + tb[id-1, 'CP'])/2  
cp

## [1] 0.005602056

cpp= prune.rpart(rpmods, cp)  
  
plot(cpp)  
text(cpp)



library(rattle)

## Warning: package 'rattle' was built under R version 3.6.3

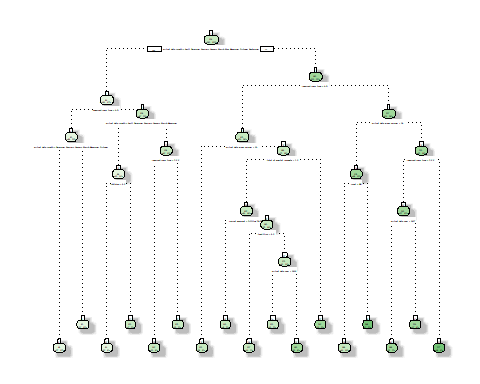
## Rattle: A free graphical interface for data science with R.  
## Version 5.3.0 Copyright (c) 2006-2018 Togaware Pty Ltd.  
## Type 'rattle()' to shake, rattle, and roll your data.

library(rpart.plot)

## Warning: package 'rpart.plot' was built under R version 3.6.3

library(RColorBrewer)  
  
fancyRpartPlot(cpp, caption=NULL)

## Warning: labs do not fit even at cex 0.15, there may be some overplotting



This is the final optimal regression tree size based on optimal cv error.

library(randomForest)

## Warning: package 'randomForest' was built under R version 3.6.3

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:rattle':  
##   
## importance

## The following object is masked from 'package:dplyr':  
##   
## combine

## The following object is masked from 'package:ggplot2':  
##   
## margin

rfm = randomForest(adr ~ is\_canceled + reserved\_room\_type + market\_segment + lead\_time + arrival\_date\_year + arrival\_date\_week\_number + stays\_in\_week\_nights + adults + children + total\_of\_special\_requests, trn)  
  
p = predict(rfm, trn)  
  
#training error  
sum((trn$adr - p)^2)

## [1] 101966.9

#test(CV) error  
sum((trn$adr - rfm$predicted)^2)

## [1] 381974.4

Using the same predictors as linear regression in a tree model, I fitted the random forests to see how training error and cross-validation test errors are different. The training error is 101966.9 and the test error is 381974.4.