

# R Coding Sample

Yuqi Shi

2024-10-30

```
# Set working directory and libraries
setwd('C:/Users/shiyq/Downloads/IntercessionR')

library(dplyr) # For any additional data manipulation

# Read the dataset
real_estate <- read.csv('realestatedata.csv')

# Overview of the dataset
head(real_estate)
```

```
##   sale_price year_built yr_sold month_sold size_sqf floor hallway_type
## 1    141592     2006    2007         8      814     3    terraced
## 2     51327     1985    2007         8      587     8    corridor
## 3     48672     1985    2007         8      587     6    corridor
## 4    380530     2006    2007         8     2056     8    terraced
## 5    221238     1993    2007         8     1761     3      mixed
## 6     35840     1992    2007         8      355     5    corridor
##      heating_type    apt_manage_type n_parkinglot_ground
## 1 individual_heating management_in_trust             111
## 2 individual_heating    self_management              80
## 3 individual_heating    self_management              80
## 4 individual_heating management_in_trust             249
## 5 individual_heating management_in_trust             523
## 6 individual_heating management_in_trust             200
##   n_parkinglot_basement time_to_bus_stop time_to_subway n_apt n_manager
## 1                   184      5min~10min    10min~15min     3         3
## 2                   76       0~5min      5min~10min     1         2
## 3                   76       0~5min      5min~10min     1         2
## 4                  536       0~5min       0~5min       6         5
## 5                  536       0~5min     15min~20min     8         8
## 6                   0      5min~10min    10min~15min     3         5
##   n_elevators n_facilities_in_apt n_facilities_near_by_total
## 1           0                 5                      6
## 2           2                 3                      12
## 3           2                 3                      12
## 4          11                 5                       3
## 5          20                 4                      14
## 6          10                 3                      16
##   n_school_near_by_total
## 1                      9
## 2                      4
```

```
## 3          4
## 4          7
## 5         17
## 6         17
```

```
colnames(real_estate)
```

```
## [1] "sale_price"      "year_built"
## [3] "yr_sold"         "month_sold"
## [5] "size_sqf"        "floor"
## [7] "hallway_type"    "heating_type"
## [9] "apt_manage_type" "n_parkinglot_ground"
## [11] "n_parkinglot_basement" "time_to_bus_stop"
## [13] "time_to_subway"  "n_apt"
## [15] "n_manager"       "n_elevators"
## [17] "n_facilities_in_apt" "n_facilities_near_by_total"
## [19] "n_school_near_by_total"
```

```
summary(real_estate)
```

```
##   sale_price      year_built      yr_sold      month_sold
##   Min.   : 32743   Min.   :1978   Min.   :2007   Min.   : 1.00
##   1st Qu.:144247   1st Qu.:1993   1st Qu.:2010   1st Qu.: 3.00
##   Median :207964   Median :2006   Median :2013   Median : 6.00
##   Mean   :221218   Mean   :2003   Mean   :2013   Mean   : 6.16
##   3rd Qu.:291150   3rd Qu.:2008   3rd Qu.:2015   3rd Qu.: 9.00
##   Max.   :585840   Max.   :2015   Max.   :2017   Max.   :12.00
##   size_sqf      floor      hallway_type      heating_type
##   Min.   : 135.0   Min.   : 1.00   Length:5891   Length:5891
##   1st Qu.: 644.0   1st Qu.: 6.00   Class :character   Class :character
##   Median : 910.0   Median :11.00   Mode  :character   Mode  :character
##   Mean   : 955.6   Mean   :12.03
##   3rd Qu.:1149.0   3rd Qu.:17.00
##   Max.   :2337.0   Max.   :43.00
##   apt_manage_type  n_parkinglot_ground n_parkinglot_basement
##   Length:5891      Min.   : 0.0      Min.   : 0.0
##   Class :character  1st Qu.: 11.0      1st Qu.: 184.0
##   Mode  :character  Median :100.0      Median : 536.0
##                      Mean   :195.9      Mean   : 570.8
##                      3rd Qu.:249.0      3rd Qu.: 798.0
##                      Max.   :713.0      Max.   :1321.0
##   time_to_bus_stop  time_to_subway      n_apt      n_manager
##   Length:5891      Length:5891      Min.   : 1.000   Min.   : 1.00
##   Class :character  Class :character  1st Qu.: 3.000   1st Qu.: 5.00
##   Mode  :character  Mode  :character  Median : 7.000   Median : 6.00
##                      Mean   : 5.614   Mean   : 6.31
##                      3rd Qu.: 8.000   3rd Qu.: 8.00
##                      Max.   :13.000   Max.   :14.00
##   n_elevators  n_facilities_in_apt n_facilities_near_by_total
##   Min.   : 0.00   Min.   : 1.00   Min.   : 0.000
##   1st Qu.: 5.00   1st Qu.: 4.00   1st Qu.: 8.000
##   Median :11.00   Median : 5.00   Median : 9.000
##   Mean   :11.15   Mean   : 5.81   Mean   : 9.871
```

```
## 3rd Qu.:16.00 3rd Qu.: 7.00 3rd Qu.:13.000
## Max. :27.00 Max. :10.00 Max. :16.000
## n_school_near_by_total
## Min. : 0.00
## 1st Qu.: 7.00
## Median :10.00
## Mean :10.86
## 3rd Qu.:15.00
## Max. :17.00
```

*#Choose predictors by plotting the relationship between the categorical variables and the dependent variable, and the relationship between the numeric variables and the dependent variable*

*# Correlation matrix*

```
cor(real_estate[, c(1:6, 10:11,14:19)))[1,]
```

```
##          sale_price          year_built
##          1.0000000          0.4478751
##          yr_sold          month_sold
##          0.3576821          0.0819240
##          size_sqf          floor
##          0.6971990          0.3367288
##          n_parkinglot_ground  n_parkinglot_basement
##          -0.1305495          0.4732516
##          n_apartment          n_manager
##          0.1626843          0.3583294
##          n_elevators          n_facilities_in_apartment
##          0.2082483          0.5052823
##          n_facilities_near_by_total  n_school_near_by_total
##          -0.4197625          -0.3779044
```

*# year\_built, yr\_sold, size\_sqf, floor, n\_parkinglot\_basement, n\_facilities\_near\_by\_total have relatively strong correlation with the dependent variable.  
#I chose one of the three variables related to facilities (total facilities).*

*# Boxplots*

```
par(mfrow = c(2, 3))
boxplot(sale_price ~ hallway_type, data = real_estate)
boxplot(sale_price ~ heating_type, data = real_estate)
boxplot(sale_price ~ apt_manage_type, data = real_estate)
boxplot(sale_price ~ time_to_bus_stop, data = real_estate)
boxplot(sale_price ~ time_to_subway, data = real_estate)
# apt_manage_type and hallway_type seem to have a strong impact on price.
```

*# Create dummy variables*

```
manage_dummies <- model.matrix( ~ ., real_estate['apt_manage_type'])
real_estate <- cbind(real_estate, manage_dummies[, 2])
colnames(real_estate)[20] <- "self_management"
```

```
hallway_dummies <- model.matrix( ~ ., real_estate['hallway_type'])
real_estate <- cbind(real_estate, hallway_dummies[, 2:3])
```

*# Fit the model. Chose OLS model because the dependent variable sales\_price is a quantitative variable.*

```
model1 <- lm(sale_price ~ year_built + yr_sold + size_sqf + floor + n_parkinglot_basement + n_facilities_near_by_total + self_management + hallway_typedmixed + hallway_typedterraced, data = real_estate)

summary(model1)
```

```
##
## Call:
## lm(formula = sale_price ~ year_built + yr_sold + size_sqf + floor +
##     n_parkinglot_basement + n_facilities_near_by_total + self_management +
##     hallway_typedmixed + hallway_typedterraced, data = real_estate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -208987  -20496    2801   25192  159346
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.631e+07  4.618e+05 -56.960 < 2e-16 ***
## year_built      1.425e+03  1.100e+02  12.953 < 2e-16 ***
## yr_sold         1.166e+04  2.113e+02  55.198 < 2e-16 ***
## size_sqf        1.684e+02  1.891e+00  89.068 < 2e-16 ***
## floor          1.343e+03  8.251e+01  16.279 < 2e-16 ***
## n_parkinglot_basement  4.266e+01  2.203e+00  19.360 < 2e-16 ***
## n_facilities_near_by_total -1.341e+03  2.966e+02  -4.520 6.29e-06 ***
## self_management -6.493e+03  3.059e+03  -2.122  0.0338 *
## hallway_typedmixed -1.351e+04  2.387e+03  -5.660 1.59e-08 ***
## hallway_typedterraced  2.924e+04  3.121e+03   9.368 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 44230 on 5881 degrees of freedom
## Multiple R-squared:  0.8274, Adjusted R-squared:  0.8271
## F-statistic: 3132 on 9 and 5881 DF, p-value: < 2.2e-16
```

```
# All the coefficients included in this model are statistically significant predictors of
#sales price as their p-values are all less than 0.05. The adjusted R-squared value is 8.271,
#showing that the model has a strong predictive ability of sales price. Year sold,
#management type and hallway type have the strongest impact on sale price by checking their coefficient.
#For example, for one unit increase in year sold, the price will increase by 11660 on average,
#indicating the apartment selling price has been increasing over the years.
```

```
# Validate the model using 5-fold cross validation
```

```
# Compute RMSE to evaluate the performance of regressions out-of-sample
```

```
rmse <- function(predicted, observed) {
  return(sqrt(sum((predicted - observed)^2)/length(observed)))
}
```

```
folds <- sample(rep(1:5, length = nrow(real_estate)))
```

```
rep(1:5, length=nrow(real_estate))
```

```
##      [1] 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2
```

##	[38]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
##	[75]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	
##	[112]	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	
##	[149]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	
##	[186]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1		
##	[223]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	
##	[260]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5		
##	[297]	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	
##	[334]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	
##	[371]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1		
##	[408]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	
##	[445]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5		
##	[482]	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	
##	[519]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	
##	[556]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1		
##	[593]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	
##	[630]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5		
##	[667]	2	3	4	5	1																																	

##	[2036]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2
##	[2073]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4
##	[2110]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	
##	[2147]	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	
##	[2184]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	
##	[2221]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5		
##	[2258]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	
##	[2295]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4		
##	[2332]	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1		
##	[2369]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	
##	[2406]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5		
##	[2443]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	
##	[2480]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4		
##	[2517]	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1		
##	[2554]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	
##	[2591]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5		
##	[2628]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	
##	[2665]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3</																		

##	[4034]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
##	[4071]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2
##	[4108]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4
##	[4145]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1
##	[4182]	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3
##	[4219]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
##	[4256]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2
##	[4293]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4
##	[4330]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1
##	[4367]	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3
##	[4404]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
##	[4441]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2
##	[4478]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4
##	[4515]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1
##	[4552]	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3
##	[4589]	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
##	[4626]	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2
##	[4663]	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4
##	[4700]	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1
##	[4737]	2	3	4	5	1	2	3	4	5	1	2																					

```

## [1] 1 1 4 3 2 5 1 2 4 5 3 4 1 2 4 5 2 1 4 2 5 2 2 1 5 3 4 2 4 3 3 5 2 4 2 3 1
## [38] 3 2 1 2 1 5 5 1 5 5 5 4 4 4 3 2 4 3 3 4 2 3 3 1 2 3 4 1 5 4 3 3 2 2 3 2 4
## [75] 5 5 4 4 4 1 3 3 2 5 5 2 5 2 4 1 3 3 2 3 3 2 1 4 1 4 5 2 4 2 1 4 4 2 5 3 4
## [112] 4 2 1 4 4 1 5 5 1 5 2 5 4 5 3 2 4 5 5 2 2 1 5 4 4 1 3 2 4 1 2 4 5 3 2 3 3
## [149] 2 3 5 3 2 4 4 2 3 2 5 4 2 1 3 5 1 3 1 5 2 5 5 1 4 4 5 2 4 3 5 3 3 3 3 3 4
## [186] 3 2 2 3 1 3 1 1 4 1 1 3 5 2 3 3 2 3 4 5 1 5 4 3 5 4 5 5 4 3 1 1 2 5 1 3 4
## [223] 4 4 4 1 1 5 4 5 4 4 3 4 1 5 4 4 1 1 5 3 4 5 4 5 5 3 1 1 4 5 3 5 5 5 3 4 3
## [260] 1 5 1 5 3 5 3 2 3 5 2 2 1 3 2 1 5 5 5 1 4 3 4 4 1 3 5 2 4 3 4 1 2 4 1 5 1
## [297] 4 5 5 4 5 1 5 3 3 3 4 4 3 3 1 3 4 1 5 3 1 1 3 5 4 2 1 4 1 1 1 5 3 2 2 5 4
## [334] 3 2 3 5 3 5 2 2 4 2 2 3 4 5 2 1 3 4 3 4 4 3 4 2 1 1 4 4 2 4 4 5 2 4 5 5 4
## [371] 5 1 4 4 4 5 2 5 2 5 2 5 5 2 4 5 2 4 5 1 1 2 3 4 5 3 1 5 3 2 3 3 1 5 5 3 1
## [408] 5 1 1 4 4 5 1 5 2 3 3 5 2 3 3 5 1 1 3 5 1 4 5 1 5 1 5 1 5 1 4 1 4 1 5 3 5 3 1
## [445] 5 3 5 1 5 2 5 3 4 5 1 4 1 5 5 1 1 5 1 4 5 3 5 3 1 5 3 3 3 1 4 5 5 3 3 1 2
## [482] 1 5 5 3 2 2 5 4 1 2 3 2 3 5 4 1 2 1 4 1 3 1 4 5 3 5 2 1 2 5 3 3 2 3 3 2 3
## [519] 3 3 2 2 1 3 5 1 1 1 3 1 3 3 2 3 4 2 4 5 2 4 1 2 2 1 3 3 4 3 5 1 5 5 4 1 5
## [556] 4 3 4 2 1 5 2 4 2 2 4 5 3 5 1 2 3 1 3 5 2 5 2 1 5 2 1 4 4 5 5 2 3 5 5 2 2
## [593] 2 2 3 1 5 4 4 2 3 2 1 4 3 5 1 1 1 1 2 1 5 3 4 5 5 4 4 2 1 3 2 5 3 4 2 4 2
## [630] 5 4 3 2 4 3 5 5 2 3 3 2 4 2 2 2 3 3 1 2 3 5 3 1 3 1 1 4 4 3 1 4 5 2 5 1 5
## [667] 2 4 1 2 2 2 1 5 3 3 1 2 4 4 5 4 2 1 5 4 1 5 1 2 3 3 3 4 3 4 4 2 3 3 1 2 2
## [704] 1 5 5 1 1 5 2 1 1 2 3 3 5 5 4 3 4 4 3 4 4 2 3 1 5 2 5 5 1 2 3 2 2 4 3 1 1
## [741] 5 2 5 1 2 5 3 1 2 1 4 2 3 3 2 4 2 2 4 2 5 1 1 3 5 1 3 2 1 4 4 1 2 4 4 1 2
## [778] 5 3 5 2 4 5 1 3 4 3 4 1 1 4 2 2 2 4 4 4 3 2 2 1 2 3 1 4 4 3 5 5 5 1 4 4 3
## [815] 1 5 1 3 2 5 4 2 4 5 4 4 1 5 4 5 5 2 5 4 2 1 1 2 2 1 4 5 2 3 4 1 3 4 1 1 4
## [852] 5 4 1 2 5 3 4 2 1 3 4 2 3 2 4 2 3 2 2 4 1 4 2 3 4 5 2 5 4 1 5 5 3 3 4 5 1
## [889] 1 5 5 2 4 4 3 2 3 5 5 5 5 3 4 5 5 2 4 3 5 5 3 2 1 4 3 3 5 2 2 4 4 5 5 4 4
## [926] 3 2 3 2 5 4 4 4 4 4 2 5 4 2 1 5 2 4 1 1 4 4 2 4 3 2 3 1 4 5 5 4 4 1 2 1 1
## [963] 4 3 2 1 3 4 2 4 5 3 4 4 2 4 3 3 2 1 4 5 3 4 2 1 4 4 1 3 2 3 3 5 3 2 4 4 5
## [1000] 1 2 2 3 2 1 4 5 4 5 1 5 1 3 4 1 1 1 5 5 4 3 4 1 5 2 2 5 5 5 5 5 4 4 5 4 4
## [1037] 1 1 4 2 2 5 5 3 2 4 3 5 1 2 2 4 2 5 4 5 4 5 4 3 5 2 5 5 5 1 4 1 4 1 5 4 1
## [1074] 3 5 2 1 5 1 3 1 3 5 3 4 4 1 2 3 3 5 2 4 1 1 4 5 2 2 5 1 1 5 5 3 5 1 3 1 3
## [1111] 2 5 4 2 3 1 4 4 5 1 4 1 2 1 3 4 1 3 1 1 2 4 2 5 1 2 3 4 4 4 4 1 4 4 2 5 1
## [1148] 2 4 4 4 1 1 2 1 5 2 3 2 5 5 5 1 5 1 3 1 3 3 1 2 5 2 5 4 4 4 5 3 4 1 1 2 5
## [1185] 4 2 2 3 1 1 3 2 1 4 2 5 4 2 4 4 3 3 2 1 5 3 1 1 5 5 4 3 2 2 3 5 3 4 2 5 2
## [1222] 3 4 1 1 1 2 3 2 2 2 5 4 3 2 4 5 1 5 4 2 3 4 5 4 2 5 5 4 1 5 3 1 5 3 4 3 5
## [1259] 5 3 3 3 2 3 2 5 3 3 1 5 3 1 4 5 5 1 2 1 1 4 5 2 3 2 3 2 1 5 4 2 3 1 5 3 4
## [1296] 5 1 4 3 5 4 1 3 3 5 4 5 5 5 2 4 1 1 5 5 4 4 1 1 3 3 5 2 5 3 4 3 4 1 5 3 2
## [1333] 2 2 3 3 2 5 3 1 4 4 3 3 3 2 5 4 5 4 3 4 1 3 2 3 2 2 1 1 4 2 1 3 3 1 2 5 1
## [1370] 3 2 4 5 3 5 5 2 4 2 5 1 5 5 1 1 2 2 4 3 5 1 5 3 4 3 2 5 5 3 2 3 1 3 3 5 3
## [1407] 3 4 2 4 1 4 2 5 1 3 2 5 1 2 3 5 3 2 5 2 1 2 5 4 2 3 1 2 5 5 3 4 5 2 2 5 5
## [1444] 4 3 3 1 4 4 5 2 4 3 1 1 4 4 3 4 3 5 4 2 5 1 2 5 5 3 2 5 4 2 5 2 5 2 3 1 3
## [1481] 4 1 2 5 5 1 3 2 5 1 5 2 2 5 2 2 1 3 5 2 5 3 3 2 1 5 4 4 2 1 4 4 4 5 3 2 4
## [1518] 3 4 1 2 5 1 3 2 2 4 4 4 2 2 5 2 4 1 4 3 1 2 1 1 3 4 3 4 1 1 4 2 4 1 1 1 4
## [1555] 1 1 4 4 4 1 2 2 3 1 3 3 3 3 2 2 4 4 3 3 2 1 1 2 4 3 1 3 5 2 1 2 3 4 4 2 3
## [1592] 5 4 5 2 4 5 4 3 3 1 4 2 2 5 5 1 5 1 3 5 1 3 5 1 1 4 3 4 1 2 2 1 1 4 2 1 5
## [1629] 5 4 3 2 5 1 4 3 2 5 1 2 1 2 1 3 1 2 4 5 5 1 2 3 4 2 4 3 4 2 4 5 5 5 1 1 5
## [1666] 3 3 3 1 1 1 1 5 3 2 3 4 3 5 5 2 5 2 3 5 2 2 5 4 5 4 5 1 3 1 1 2 1 1 1 3 1
## [1703] 1 3 4 4 1 2 5 4 5 3 2 5 3 5 5 5 2 4 5 5 4 3 2 4 5 3 1 2 2 1 2 3 3 5 2 3 4
## [1740] 5 2 1 3 4 2 5 2 3 4 5 5 1 3 2 2 1 4 4 1 1 2 2 5 3 1 1 4 4 5 3 1 4 1 5 2 1
## [1777] 2 2 4 5 1 2 4 1 4 4 3 3 3 5 2 5 4 4 5 1 2 2 4 4 1 2 2 4 4 1 3 3 1 1 1 5
## [1814] 5 1 3 3 1 1 2 1 3 5 2 4 1 2 3 3 1 1 4 5 1 5 5 3 3 1 4 4 2 4 2 2 2 1 2 2 4
## [1851] 2 2 3 4 4 3 5 5 1 5 1 1 2 3 5 3 4 1 1 4 3 3 1 3 2 3 2 1 2 3 1 2 4 5 5 5 1
## [1888] 3 4 4 2 4 1 4 2 2 5 5 4 3 3 4 5 5 4 2 4 3 5 4 2 5 1 5 5 3 3 2 2 5 3 4 2 4
## [1925] 4 1 5 4 2 3 3 5 2 1 1 3 3 1 4 4 3 1 5 1 4 5 3 3 3 1 2 3 3 1 2 5 5 4 1 3 3
## [1962] 5 4 3 3 2 3 5 1 5 2 3 2 1 4 3 1 3 2 4 2 3 1 5 2 2 3 1 4 4 5 2 3 4 4 5 4 1

```



```

## [1999] 5 1 1 3 1 4 5 2 4 1 4 4 2 1 1 5 3 1 1 5 4 4 1 2 1 2 2 1 3 2 4 1 5 3 2 1 2
## [2036] 3 3 3 3 5 1 5 3 3 2 1 1 4 2 3 3 3 5 4 2 2 5 4 5 4 5 5 4 5 4 1 5 3 4 1 2 1
## [2073] 5 2 5 3 1 5 1 2 1 4 2 5 4 5 1 2 4 2 2 5 1 2 2 4 5 3 4 2 1 3 5 3 3 3 2 5 4
## [2110] 2 1 5 4 2 4 2 1 4 3 4 5 5 1 3 4 5 2 4 1 3 4 2 5 2 1 3 3 5 3 4 1 2 2 5 5 4
## [2147] 3 2 3 4 5 3 3 1 1 4 3 4 5 3 5 3 1 5 2 2 5 1 2 1 5 5 4 5 1 2 1 5 2 5 1 3 2
## [2184] 2 2 2 1 2 4 2 4 5 1 3 2 5 3 1 2 4 1 1 2 1 5 2 1 2 3 3 4 2 4 4 3 3 5 1 3 1
## [2221] 3 1 3 5 2 5 1 1 3 4 3 2 5 3 1 1 1 1 5 2 5 3 4 4 5 4 1 3 5 2 4 5 2 5 5 2 5
## [2258] 4 4 1 4 1 3 3 2 2 2 3 4 2 4 3 2 1 1 4 5 3 1 4 1 4 1 4 4 4 3 3 4 3 3 2 5 2
## [2295] 3 2 5 4 2 1 1 1 1 5 3 2 4 2 5 4 2 3 2 1 2 1 5 3 1 4 3 3 2 5 1 3 1 1 3 4 4
## [2332] 2 2 2 5 4 3 1 1 2 4 3 4 4 5 3 2 2 3 5 3 3 4 2 5 5 1 2 3 4 1 5 1 4 4 1 4 4
## [2369] 2 2 2 4 2 4 1 5 3 1 3 2 2 3 1 4 5 5 1 2 4 5 5 3 3 2 4 1 5 5 1 4 4 3 3 4 4
## [2406] 5 2 5 5 3 4 2 2 5 4 5 2 2 1 1 5 4 4 3 3 3 5 4 5 5 5 3 3 3 5 1 5 5 2 3 1 3
## [2443] 5 5 1 1 3 3 5 3 2 5 3 4 2 5 3 3 1 1 3 2 1 1 1 4 5 1 1 4 5 2 3 2 3 4 2 5 3
## [2480] 3 1 4 2 3 1 5 2 4 5 2 3 3 1 5 3 2 2 4 5 5 2 2 5 1 1 1 5 5 2 3 5 4 4 3 3 2
## [2517] 5 1 5 4 2 5 4 1 2 5 3 4 2 5 4 5 2 4 3 1 5 5 5 2 5 3 3 2 3 3 4 5 5 2 5 3 1
## [2554] 1 4 4 2 3 3 2 4 4 2 5 2 4 4 1 4 1 3 5 3 4 5 2 3 3 4 3 3 1 3 1 2 5 4 5 2 3
## [2591] 1 5 4 3 3 1 3 5 5 4 4 4 3 3 3 2 2 5 5 2 2 1 3 5 1 2 2 5 2 4 2 1 1 2 3 2 3
## [2628] 2 1 5 5 3 1 4 2 5 5 1 5 1 5 2 3 3 4 3 2 4 1 3 4 3 3 1 3 2 1 3 1 4 5 1 4 1
## [2665] 1 4 2 3 5 1 4 1 3 2 1 3 2 3 4 3 1 1 3 5 2 2 2 3 2 1 4 2 4 4 2 3 2 2 3 4 5
## [2702] 3 2 3 4 3 1 5 3 5 2 2 4 3 4 3 4 2 1 2 5 3 1 3 3 3 5 3 4 1 1 2 1 4 2 2 2 1
## [2739] 1 1 5 5 1 1 5 4 3 1 2 3 4 1 5 4 1 4 2 5 2 4 4 3 2 1 1 5 2 4 4 2 1 4 4 4 5
## [2776] 1 3 2 5 2 4 3 5 2 4 2 4 1 4 3 2 4 4 1 4 2 5 3 5 4 5 5 2 4 5 1 3 4 1 3 2 5
## [2813] 5 5 2 4 3 3 1 5 2 5 5 5 5 3 4 4 1 3 5 1 2 4 3 4 1 2 1 5 2 4 1 4 2 3 2 4 2
## [2850] 1 2 2 2 2 4 5 1 1 2 2 2 2 3 3 1 5 4 4 1 2 1 5 1 3 1 2 4 4 5 4 3 2 5 1 4 5
## [2887] 3 2 1 4 2 3 1 4 3 3 4 2 5 2 5 2 5 2 1 2 3 1 4 1 2 2 3 2 5 1 3 3 1 5 2 5 2
## [2924] 3 3 5 1 3 5 1 3 2 5 1 2 3 2 5 1 2 3 5 5 5 3 1 4 1 2 4 2 3 4 4 2 3 5 2 1 3
## [2961] 5 1 4 4 2 4 2 2 3 3 4 5 4 5 5 1 2 5 1 2 1 3 4 5 4 4 2 5 5 3 5 1 2 5 4 5 5
## [2998] 1 5 5 2 1 3 2 1 3 2 5 2 4 4 2 3 1 5 2 1 4 4 1 1 4 3 5 4 4 3 5 2 3 3 4 4 1
## [3035] 5 4 1 2 2 3 5 5 3 4 4 2 5 3 4 5 3 1 5 4 3 5 4 3 4 2 3 4 2 5 4 2 2 5 3 1 4
## [3072] 4 4 1 5 2 4 4 3 3 1 2 5 3 1 2 5 4 3 5 3 4 3 5 4 5 4 3 3 3 3 5 1 1 3 2 4 3
## [3109] 3 5 1 3 3 3 3 2 5 1 5 2 1 3 2 1 4 2 2 4 1 2 4 1 2 3 5 4 4 3 5 4 4 1 3 4 1
## [3146] 1 3 5 5 1 2 5 3 4 2 3 5 2 1 1 5 1 4 3 3 1 1 2 4 3 2 3 3 3 2 3 1 3 4 5 4 3
## [3183] 4 1 4 5 3 4 5 5 1 1 5 3 1 1 5 1 1 4 1 5 5 1 5 1 1 1 2 5 2 2 1 3 2 2 4 5 3
## [3220] 1 3 2 1 4 2 1 1 1 4 3 3 3 5 4 2 4 2 3 1 1 4 3 1 2 1 2 5 1 5 3 4 4 2 1 2 1
## [3257] 4 4 3 3 3 2 5 3 3 1 3 3 3 5 2 3 1 4 3 3 1 5 3 4 2 4 5 2 4 1 2 2 2 4 5 3 5
## [3294] 4 2 5 2 4 5 5 2 4 4 5 1 2 2 4 5 1 2 3 5 5 3 5 1 3 2 4 4 4 3 5 3 2 4 5 5 3
## [3331] 4 2 2 2 5 4 1 1 4 4 3 4 3 5 2 4 2 1 1 3 1 3 3 2 3 3 2 1 2 3 4 5 3 4 1 3 5
## [3368] 4 4 3 2 2 2 1 5 1 4 1 5 2 2 3 1 4 1 1 5 4 5 1 3 3 5 5 2 5 4 3 1 5 4 1 2 3
## [3405] 1 4 4 1 3 2 3 4 3 4 1 1 1 3 1 2 4 3 5 5 1 2 4 3 1 2 4 3 3 5 1 1 4 3 2 4 4
## [3442] 5 4 5 5 5 1 4 3 4 1 1 3 5 2 5 2 1 1 5 4 1 4 1 2 1 1 1 4 4 3 3 2 5 2 1 1 2
## [3479] 5 1 4 3 2 5 4 1 3 1 1 2 3 4 1 2 1 5 2 2 3 2 3 5 3 1 4 2 4 1 2 5 5 5 1 2 2
## [3516] 3 1 2 5 5 3 3 1 4 2 4 5 2 1 2 2 3 3 3 4 2 2 3 2 5 2 2 2 2 3 5 4 2 5 2 2 3
## [3553] 2 5 1 3 3 3 3 2 1 4 3 5 1 1 4 3 5 5 4 5 4 1 3 4 5 5 2 3 4 1 4 2 4 2 4 4 4
## [3590] 2 2 1 4 1 4 4 3 2 5 4 3 5 1 1 1 5 3 1 5 2 5 2 1 2 3 1 2 1 5 3 5 3 5 4 5 3
## [3627] 5 1 5 2 1 4 4 3 4 5 2 4 1 1 3 5 1 1 2 3 5 5 3 4 1 4 2 1 5 4 4 5 5 1 3 1 1
## [3664] 2 4 2 4 2 1 3 5 4 3 5 4 5 5 2 1 5 5 5 4 4 5 5 1 4 2 5 1 1 5 4 4 1 1 2 4 3
## [3701] 3 3 2 5 3 2 2 4 5 4 2 5 3 5 5 4 2 2 2 3 4 5 1 4 3 2 5 1 5 5 2 5 4 5 1 2 4
## [3738] 4 5 5 1 2 5 1 2 3 2 1 4 3 4 2 1 5 2 5 2 1 3 1 2 4 4 2 3 1 4 1 3 2 4 4 5 1
## [3775] 4 1 4 1 2 1 3 3 4 1 5 1 5 5 4 1 5 3 3 2 2 3 4 1 1 3 1 1 5 5 2 2 2 2 1 2 4
## [3812] 1 1 1 2 2 1 5 5 4 1 3 2 1 5 3 2 5 1 3 1 2 5 3 1 1 4 2 2 4 3 1 5 5 4 1 3 4
## [3849] 1 3 4 4 1 3 3 4 1 1 5 3 2 1 2 4 3 4 2 5 2 1 2 1 3 5 2 5 3 3 1 4 5 1 3 1 1
## [3886] 2 2 3 4 2 2 4 4 4 3 5 4 2 3 2 1 4 5 4 5 5 5 4 5 1 5 4 4 1 3 4 1 3 1 1 2 1
## [3923] 2 1 3 4 1 3 2 3 4 5 2 2 4 2 2 3 5 2 1 1 1 3 4 4 1 1 4 3 5 3 2 2 1 3 1 2 3
## [3960] 1 1 3 5 2 1 1 1 4 5 4 1 5 2 4 5 2 5 1 2 4 4 4 5 4 3 4 5 4 5 1 5 5 5 2 5 2

```

```

## [3997] 4 2 4 3 5 3 2 3 4 2 4 5 1 2 4 4 5 2 5 1 1 4 1 2 1 5 1 3 4 2 3 2 3 2 2 1 5
## [4034] 2 3 4 4 2 5 3 3 1 4 4 1 4 3 4 5 2 1 1 3 4 4 2 3 1 3 4 3 4 2 3 1 2 1 3 2 2
## [4071] 5 1 2 1 5 5 3 4 2 4 2 1 5 2 2 2 5 3 5 2 1 1 1 3 5 4 3 2 2 1 1 2 1 4 1 5 5
## [4108] 5 2 5 3 3 4 5 5 3 4 5 3 3 2 4 3 1 1 2 3 2 1 2 3 1 1 1 4 4 3 3 1 1 5 5 4 4
## [4145] 5 1 2 1 1 3 4 2 1 1 2 1 2 1 5 3 3 5 5 3 3 3 1 3 4 3 5 1 2 2 5 1 1 1 5 4 2
## [4182] 5 3 1 4 5 3 1 3 2 1 4 5 5 4 5 5 5 5 4 3 2 2 1 5 2 4 3 1 1 1 1 2 4 4 1 2 1
## [4219] 5 1 1 4 1 4 1 3 2 4 5 1 3 2 4 3 1 4 4 1 4 5 2 1 4 2 3 3 1 3 4 3 5 5 2 2 5
## [4256] 5 4 5 3 4 3 4 5 1 1 3 1 1 2 3 2 3 3 1 5 1 3 5 2 3 3 1 2 2 1 3 5 2 5 5 5 2
## [4293] 4 5 5 4 2 1 1 2 2 3 3 3 3 1 3 2 4 4 5 5 2 3 5 3 4 4 1 3 5 5 2 4 4 1 5 5 1
## [4330] 4 3 2 1 1 5 2 5 1 2 3 1 2 5 2 3 1 4 3 4 1 3 4 4 5 4 5 5 3 1 2 5 1 1 3 2 1
## [4367] 5 5 5 1 5 4 3 3 1 4 4 5 1 2 4 2 5 1 3 3 1 5 2 2 3 4 4 4 4 2 5 3 3 1 1 1 1
## [4404] 1 2 1 5 5 1 3 2 4 5 2 5 1 4 1 1 5 4 4 3 1 3 2 4 2 3 3 3 2 4 5 2 5 3 1 5 1
## [4441] 3 5 1 2 5 4 4 2 1 3 2 1 2 3 3 1 3 3 4 1 4 5 3 5 4 2 4 1 2 2 5 4 2 4 4 2 5
## [4478] 4 4 1 5 2 5 4 3 4 5 3 4 5 5 3 2 2 5 2 4 5 4 2 1 5 2 2 4 4 1 2 3 3 5 4 4 3
## [4515] 4 5 1 4 1 5 4 1 1 3 4 2 1 2 3 2 3 2 5 4 5 1 4 4 2 3 5 2 3 5 1 3 5 1 5 2 5
## [4552] 5 2 3 3 2 1 2 4 3 2 4 5 3 4 5 5 3 2 2 4 3 3 3 5 1 3 3 1 1 5 1 5 2 1 1 2 3
## [4589] 3 1 5 3 4 5 1 1 1 4 4 1 2 4 3 3 1 2 4 2 5 1 4 3 5 3 2 2 2 4 2 5 2 4 2 2 3
## [4626] 2 2 2 4 3 4 5 4 2 1 5 4 1 3 1 1 4 4 1 3 1 5 2 5 1 3 3 1 5 2 3 3 2 3 5 4 5
## [4663] 1 3 3 1 1 4 1 4 2 3 4 3 4 1 1 5 4 1 5 4 4 5 4 2 3 1 5 5 5 5 4 2 4 2 4 2 3
## [4700] 3 5 3 4 4 5 2 1 2 4 1 3 2 2 2 4 4 3 1 4 1 1 2 1 5 4 1 1 3 4 3 3 2 3 4 2 5
## [4737] 3 1 3 1 5 4 3 4 5 2 3 3 5 5 4 5 5 2 2 4 3 4 2 2 5 4 3 3 3 1 4 4 5 3 2 4 1
## [4774] 1 3 3 4 4 2 3 4 5 4 2 1 1 5 5 1 2 3 4 3 4 5 1 2 5 5 5 5 1 4 3 3 2 3 4 2 2
## [4811] 3 1 2 5 1 4 5 4 1 2 1 1 1 5 1 4 5 1 5 3 4 4 1 2 5 4 4 1 4 2 2 1 2 3 2 4 4
## [4848] 3 3 2 3 5 5 4 5 5 1 3 4 4 3 5 5 5 2 4 4 4 5 3 4 3 2 4 2 3 5 5 5 3 1 3 1 4
## [4885] 4 5 4 4 2 3 5 2 1 1 5 3 2 5 4 1 4 3 2 1 3 5 2 5 4 3 1 2 5 3 3 2 2 2 1 4 3
## [4922] 3 3 1 5 4 1 3 3 5 3 3 3 3 4 3 3 4 3 5 4 5 4 1 3 5 5 5 5 1 4 2 5 4 2 4 3 5
## [4959] 3 2 1 5 5 3 4 1 1 1 1 1 5 4 3 5 2 5 1 2 5 2 5 3 5 5 2 3 1 2 4 5 4 4 5 3 1
## [4996] 3 2 4 2 3 1 3 5 4 5 1 4 3 4 1 4 3 5 1 1 3 3 1 1 1 4 2 5 5 5 5 3 4 1 2 3 2
## [5033] 2 2 3 5 1 3 1 2 4 1 3 4 1 2 2 5 2 5 1 2 5 4 4 4 2 1 4 4 1 1 4 2 3 5 5 3 3
## [5070] 3 1 4 2 3 1 4 5 4 5 4 2 3 5 1 5 3 4 3 5 3 3 2 1 1 4 2 1 3 2 2 1 4 3 1 2 5
## [5107] 4 3 3 2 4 5 2 5 2 4 4 2 1 4 2 1 4 2 2 3 1 2 1 4 3 1 3 2 2 1 5 2 2 4 1 1 4
## [5144] 3 2 2 1 5 3 5 4 1 2 1 1 4 4 4 1 5 3 1 4 5 2 4 2 2 4 5 1 5 3 1 2 1 3 1 2 2
## [5181] 5 5 3 2 5 1 1 2 1 3 3 1 1 1 2 3 3 1 2 5 3 3 3 5 3 1 2 5 5 2 5 3 1 4 4 3 1
## [5218] 5 1 1 4 2 5 3 2 5 4 4 5 2 5 2 3 4 4 5 3 1 3 1 2 3 2 5 2 4 3 5 1 4 5 3 2 4
## [5255] 4 2 4 4 2 1 5 1 5 2 1 5 1 3 3 4 1 4 3 5 1 3 1 4 2 1 5 1 4 3 3 2 3 5 1 3 5
## [5292] 3 4 3 3 5 4 2 4 5 4 3 3 2 4 1 1 1 3 1 1 3 2 1 2 1 2 1 2 3 1 4 4 1 5 1 4 2
## [5329] 1 2 4 1 3 1 2 2 4 4 5 2 4 3 3 4 5 1 4 1 5 5 4 2 4 2 2 3 5 3 2 1 2 4 3 5 5
## [5366] 3 3 2 5 4 2 2 2 2 5 4 4 1 3 2 2 2 4 5 1 1 1 5 3 1 3 3 3 2 3 5 5 1 4 2 3 5
## [5403] 1 2 4 3 5 2 5 1 3 3 5 2 3 1 4 5 3 3 5 2 2 1 3 4 2 5 3 4 1 1 2 4 3 4 1 4 2
## [5440] 1 1 2 1 5 4 4 1 4 1 4 1 1 1 1 5 1 4 3 5 3 2 2 5 5 2 5 2 2 1 3 3 3 1 2 4 3
## [5477] 5 5 5 1 5 1 2 1 1 1 1 5 5 3 3 4 5 1 5 1 5 2 2 4 2 4 1 2 3 3 2 4 4 3 4 1 5
## [5514] 1 4 5 4 2 2 2 4 3 1 3 2 1 5 5 2 1 4 2 3 5 5 4 3 3 5 2 4 1 2 2 3 4 4 5 5 1
## [5551] 3 3 2 1 1 4 3 5 3 3 3 5 2 3 5 2 2 1 2 5 2 2 5 2 2 5 5 5 2 3 3 4 4 5 1 4 5
## [5588] 4 3 4 4 4 4 3 4 3 2 5 3 1 1 3 4 3 1 2 1 3 3 2 3 2 5 1 4 5 1 4 1 4 1 2 5 5
## [5625] 3 1 5 4 5 3 3 2 2 3 2 2 2 4 4 2 1 4 4 1 4 5 2 5 5 1 3 1 2 1 1 4 2 3 5 1 2
## [5662] 1 3 3 3 2 3 5 2 5 5 5 1 5 2 4 4 4 2 1 5 1 4 4 5 4 4 1 5 5 1 4 4 4 3 2 3 4
## [5699] 4 3 4 1 3 3 5 5 1 1 5 2 3 5 3 2 4 4 5 4 2 2 4 5 1 3 3 2 5 4 5 2 2 5 5 5 5
## [5736] 5 3 4 3 3 3 3 1 2 3 1 5 4 3 4 4 2 5 5 4 3 3 3 4 2 2 5 1 5 4 3 1 2 1 2 4 1
## [5773] 3 2 4 3 3 3 5 3 4 5 2 1 3 2 3 5 3 4 1 4 5 1 2 1 4 2 3 2 2 3 5 3 3 5 3 2 4
## [5810] 5 3 4 3 5 1 2 2 5 2 1 5 5 3 2 5 2 2 2 4 5 4 2 2 2 5 5 3 3 4 4 2 2 5 2 3 2
## [5847] 4 4 3 2 1 4 4 4 5 4 2 2 1 3 3 2 3 4 1 3 3 4 1 4 1 1 1 3 2 2 5 5 1 5 2 4 4
## [5884] 5 1 1 4 2 4 1 5

```

```
table(folds)
```

```
## folds
##      1      2      3      4      5
## 1179 1178 1178 1178 1178
```

```
metrics <- c()
for(i in unique(folds)) {
  test <- which(folds == i)
  fit <- model1
  preds <- predict(fit, newdata = real_estate[test,], type = 'response')
  err <- rmse(predicted = preds, observed = real_estate$sale_price[test])
  metrics <- c(metrics, err)
}

mean(metrics)
```

```
## [1] 44137.74
```

*# The RMSE for this model is 44188.59. This is relatively large given the price range  
# of around 30000 ~ 140000. This means that the model has a relatively large predicative uncertainty.*

*# Set up the data as a model.matrix  
# and run a lasso regression with 5-fold cross-  
# validation on the model. Then we report the accompanying metric  
# that used to determine model performance, based on the 'lambda.min'  
# value.*

```
# Set up data as a model.matrix
?everything()
real_estate_alt <- dplyr::select(real_estate, sale_price, everything())
real_estate_alt$real_estate <- NULL
head(real_estate_alt)
```

```
##   sale_price year_built yr_sold month_sold size_sqf floor hallway_type
## 1    141592      2006   2007         8       814     3    terraced
## 2     51327      1985   2007         8       587     8    corridor
## 3     48672      1985   2007         8       587     6    corridor
## 4    380530      2006   2007         8      2056     8    terraced
## 5    221238      1993   2007         8      1761     3      mixed
## 6     35840      1992   2007         8       355     5    corridor
##      heating_type    apt_manage_type n_parkinglot_ground
## 1 individual_heating management_in_trust             111
## 2 individual_heating      self_management              80
## 3 individual_heating      self_management              80
## 4 individual_heating management_in_trust             249
## 5 individual_heating management_in_trust             523
## 6 individual_heating management_in_trust             200
##   n_parkinglot_basement time_to_bus_stop time_to_subway n_apt n_manager
## 1                   184      5min~10min   10min~15min     3         3
## 2                   76         0~5min     5min~10min     1         2
## 3                   76         0~5min     5min~10min     1         2
```

```
## 4          536          0~5min          0-5min          6          5
## 5          536          0~5min          15min~20min          8          8
## 6           0          5min~10min          10min~15min          3          5
##   n_elevators n_facilities_in_apt n_facilities_near_by_total
## 1           0           5           6
## 2           2           3          12
## 3           2           3          12
## 4          11           5           3
## 5          20           4          14
## 6          10           3          16
##   n_school_near_by_total self_management hallway_typedmixed hallway_typedterraced
## 1           9           0           0           1
## 2           4           1           0           0
## 3           4           1           0           0
## 4           7           0           0           1
## 5          17           0           1           0
## 6          17           0           0           0
```

```
predictors <- model.matrix(sale_price ~ . -1, data = real_estate_alt)
head(predictors)
```

```
##   year_built yr_sold month_sold size_sqf floor hallway_typecorridor
## 1      2006    2007         8      814     3           0
## 2      1985    2007         8      587     8           1
## 3      1985    2007         8      587     6           1
## 4      2006    2007         8     2056     8           0
## 5      1993    2007         8     1761     3           0
## 6      1992    2007         8      355     5           1
##   hallway_typedmixed hallway_typedterraced heating_typeindividual_heating
## 1           0           1           1
## 2           0           0           1
## 3           0           0           1
## 4           0           1           1
## 5           1           0           1
## 6           0           0           1
##   apt_manage_typeself_management n_parkinglot_ground n_parkinglot_basement
## 1           0           111          184
## 2           1           80           76
## 3           1           80           76
## 4           0          249          536
## 5           0          523          536
## 6           0          200           0
##   time_to_bus_stop10min~15min time_to_bus_stop5min~10min
## 1           0           1
## 2           0           0
## 3           0           0
## 4           0           0
## 5           0           0
## 6           0           1
##   time_to_subway10min~15min time_to_subway15min~20min time_to_subway5min~10min
## 1           1           0           0
## 2           0           0           1
## 3           0           0           1
## 4           0           0           0
```

```

## 5          0          1          0
## 6          1          0          0
##   time_to_subwayno_bus_stop_nearby n_apt n_manager n_elevators
## 1          0          3          3          0
## 2          0          1          2          2
## 3          0          1          2          2
## 4          0          6          5          11
## 5          0          8          8          20
## 6          0          3          5          10
##   n_facilities_in_apt n_facilities_near_by_total n_school_near_by_total
## 1          5          6          9
## 2          3          12         4
## 3          3          12         4
## 4          5          3          7
## 5          4          14         17
## 6          3          16         17
##   self_management hallway_typedmixed hallway_typedterraced
## 1          0          0          1
## 2          1          0          0
## 3          1          0          0
## 4          0          0          1
## 5          0          1          0
## 6          0          0          0

```

```
head(real_estate_alt[-1])
```

```

##   year_built yr_sold month_sold size_sqf floor hallway_type heating_type
## 1      2006    2007          8     814     3    terraced individual_heating
## 2      1985    2007          8     587     8    corridor individual_heating
## 3      1985    2007          8     587     6    corridor individual_heating
## 4      2006    2007          8    2056     8    terraced individual_heating
## 5      1993    2007          8    1761     3      mixed individual_heating
## 6      1992    2007          8     355     5    corridor individual_heating
##   apt_manage_type n_parkinglot_ground n_parkinglot_basement
## 1 management_in_trust          111          184
## 2 self_management          80          76
## 3 self_management          80          76
## 4 management_in_trust          249          536
## 5 management_in_trust          523          536
## 6 management_in_trust          200           0
##   time_to_bus_stop time_to_subway n_apt n_manager n_elevators
## 1      5min~10min    10min~15min     3         3          0
## 2           0~5min      5min~10min     1         2          2
## 3           0~5min      5min~10min     1         2          2
## 4           0~5min           0~5min     6         5          11
## 5           0~5min     15min~20min     8         8          20
## 6      5min~10min    10min~15min     3         5          10
##   n_facilities_in_apt n_facilities_near_by_total n_school_near_by_total
## 1          5          6          9
## 2          3          12         4
## 3          3          12         4
## 4          5          3          7
## 5          4          14         17
## 6          3          16         17

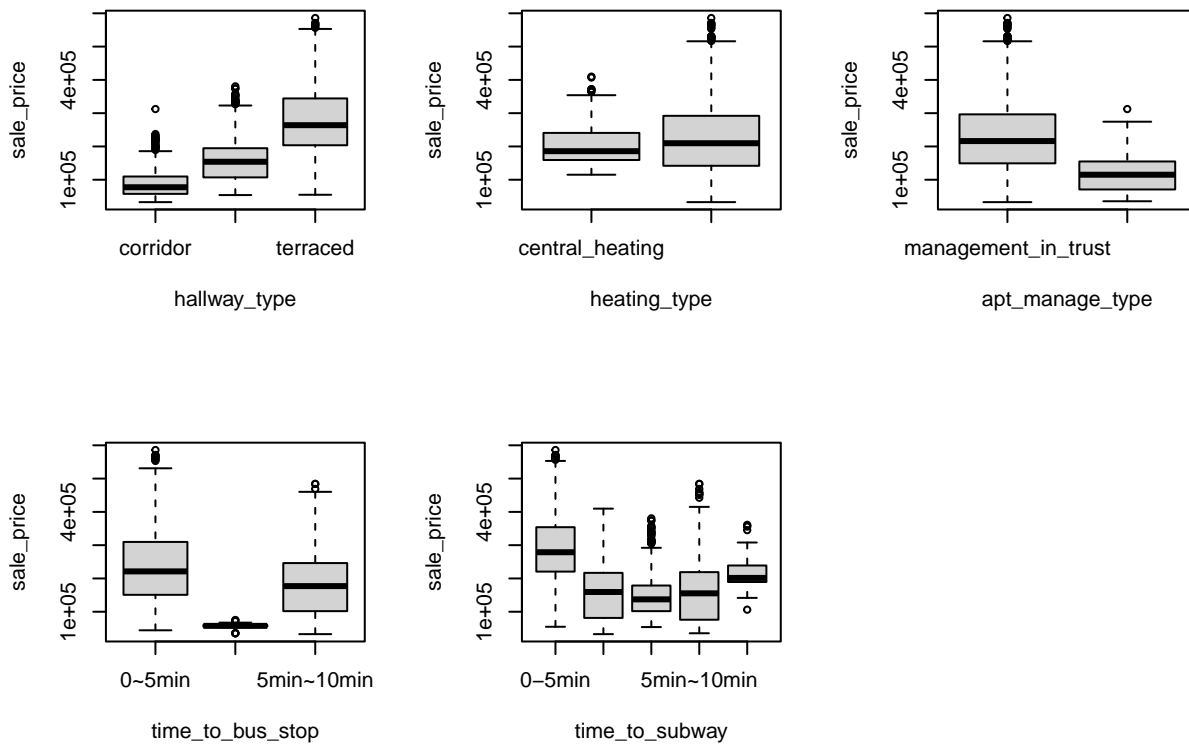
```

```
## self_management hallway_typedmixed hallway_typedterraced
## 1 0 0 1
## 2 1 0 0
## 3 1 0 0
## 4 0 0 1
## 5 0 1 0
## 6 0 0 0
```

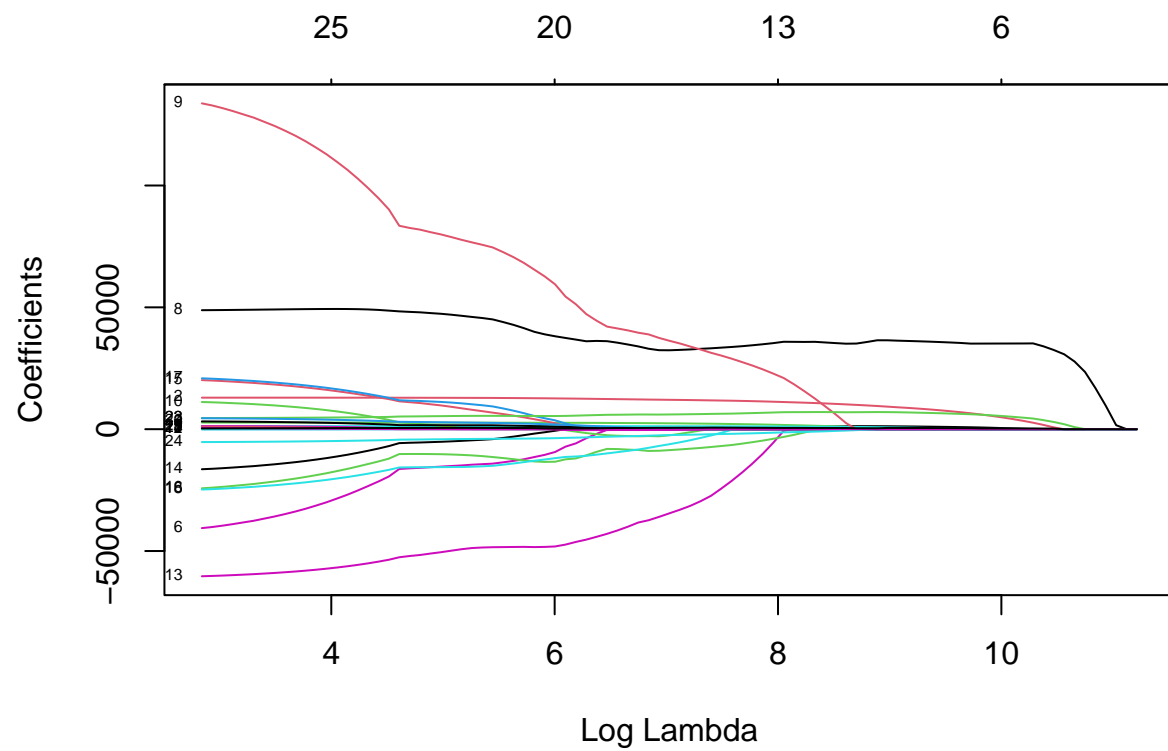
```
library(glmnet)

y <- real_estate_alt$sale_price
lasso <- glmnet(predictors, y, alpha = 1)

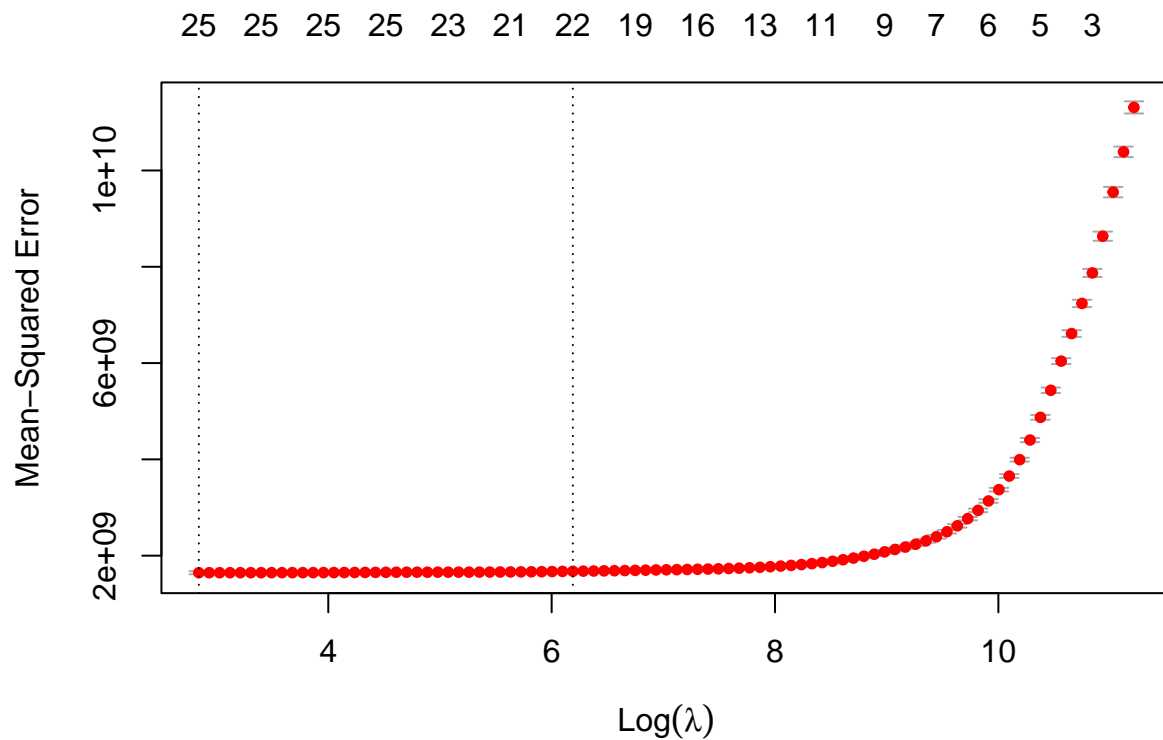
par(mfrow = c(1, 1))
```



```
plot(lasso, xvar='lambda', label = T)
```



```
# Validate the model's performance with cross-validation and pick an optimal lambda
cv_lasso <- cv.glmnet(predictors, y, alpha = 1, nfolds = 5)
plot(cv_lasso)
```



```
# Get optimal parameters
cv_lasso
```

```
##
## Call:  cv.glmnet(x = predictors, y = y, nfolds = 5, alpha = 1)
##
## Measure: Mean-Squared Error
##
##      Lambda Index  Measure      SE Nonzero
## min   17.1    91 1.646e+09 31355118      25
## 1se  488.0    55 1.675e+09 32119111      22
```

```
# Both min/1se RMSE are smaller comparing to the OLS model.
#The sparser model does improve the predictive performance.
sqrt(cv_lasso$cvm[which(cv_lasso$lambda %in% cv_lasso$lambda.min)])
```

```
## [1] 40566.42
```

```
sqrt(cv_lasso$cvm[which(cv_lasso$lambda %in% cv_lasso$lambda.1se)])
```

```
## [1] 40932.56
```



```
# Given the coefficient, the size_sqf is most associated with the value of lambda.min.
#A couple of variables should be dropped.
coef(cv_lasso, s = cv_lasso$lambda.min)
```

```
## 28 x 1 sparse Matrix of class "dgCMatrix"
##                                     s1
## (Intercept)                      -2.697993e+07
## year_built                       3.956744e+02
## yr_sold                          1.294193e+04
## month_sold                       2.732492e+03
## size_sqf                         1.605767e+02
## floor                           1.274654e+03
## hallway_typecorridor             -4.059554e+04
## hallway_typemixed                .
## hallway_typeterraced             4.883511e+04
## heating_typeindividual_heating   1.337152e+05
## apt_manage_typeself_management   1.115640e+04
## n_parkinglot_ground              -2.798602e+01
## n_parkinglot_basement            -3.995974e+01
## time_to_bus_stop10min~15min      -6.034732e+04
## time_to_bus_stop5min~10min       -1.643619e+04
## time_to_subway10min~15min        2.013700e+04
## time_to_subway15min~20min        -2.425536e+04
## time_to_subway5min~10min         2.087142e+04
## time_to_subwayno_bus_stop_nearby -2.476366e+04
## n_apt                            1.294901e+03
## n_manager                        3.200082e+03
## n_elevators                      1.057086e+03
## n_facilities_in_apt              4.567245e+03
## n_facilities_near_by_total       4.490724e+03
## n_school_near_by_total          -5.285943e+03
## self_management                  5.583369e+02
## hallway_typemixed                .
## hallway_typeterraced             2.988434e+02
```

```
library(boot)
full_fit <- glm(sale_price ~ ., data = real_estate_alt)
k_fold_mse <- cv.glm(real_estate_alt, full_fit, K = 5)$delta[1]
sqrt(k_fold_mse)
```

```
## [1] 40504.32
```

```
# Looking at the performance of the full model, it looks like regularization
#is helping us in reducing the predicative uncertainty with a smaller MSE.
```

```
library(randomForest)
set.seed(101)
dim(real_estate)
```

```
## [1] 5891 22
```

```

train <- sample(nrow(real_estate), 300)

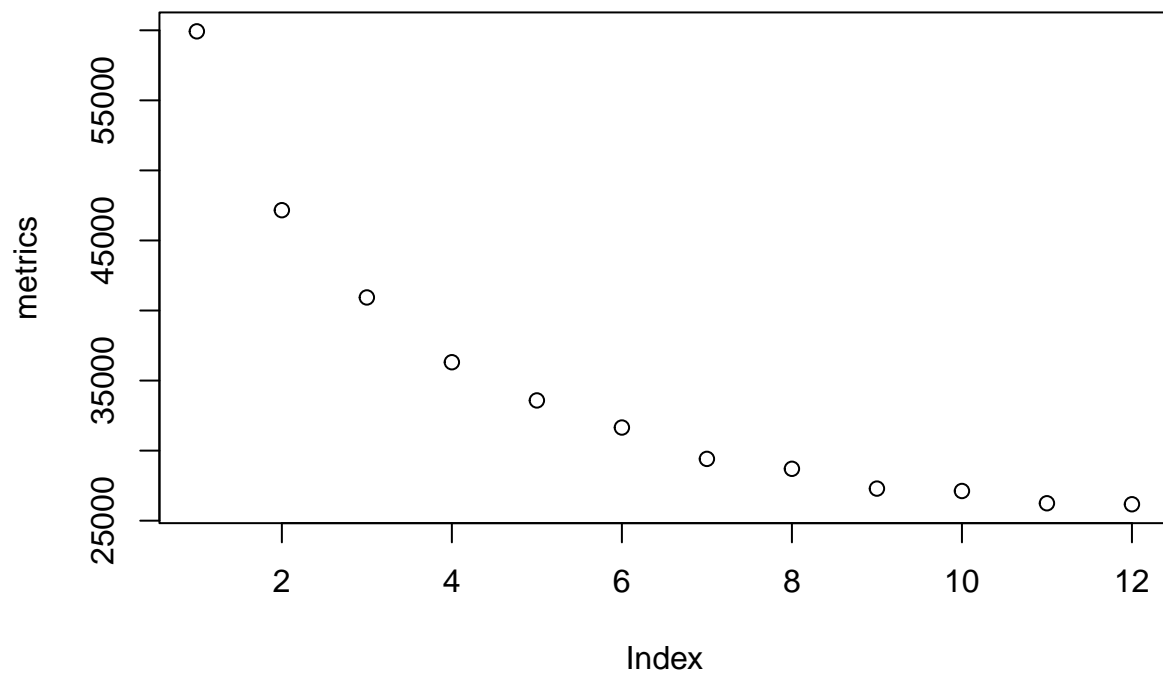
rf_model <- randomForest(sale_price ~ ., data = real_estate, subset = train)
rf_model

##
## Call:
## randomForest(formula = sale_price ~ ., data = real_estate, subset = train)
##               Type of random forest: regression
##               Number of trees: 500
## No. of variables tried at each split: 7
##
##               Mean of squared residuals: 1109306873
##               % Var explained: 90.46

# Vary our forest across the `mtry` parameter
metrics <- numeric(length=12L)
for (mtry in seq_along(metrics)) {
  message(glue::glue('Fitting randomForest models with {mtry} variable(s) per split'))
  fit <- randomForest(
    sale_price ~ .,
    data = real_estate,
    subset = train,
    mtry = mtry,
    ntree = 400
  )
  pred <- predict(fit, real_estate[-train, ])
  metrics[mtry] <- rmse(predicted = pred, observed = real_estate$sale_price[-train])
}

plot(metrics)

```



```
min(metrics)
```

```
## [1] 26172.17
```

```
# Compare against a linear model  
real_estate_lm <- glm(sale_price ~ ., data = real_estate, subset = train)  
preds <- predict(real_estate_lm, newdata = real_estate[-train,])  
rmse(preds, real_estate$sale_price[-train])
```

```
## [1] 42807.1
```

```
# The random Forest model is doing a better job than the general linear model
```