# HW_1_week2

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.6.3
```

```
## -- Attaching packages -----------------------------------------------------------------------
- tidyverse 1.3.0 --
```

```
## v ggplot2 3.2.1     v purrr   0.3.3

## v tibble  2.1.3     v dplyr   0.8.3

## v tidyr   1.0.0     v stringr 1.4.0

## v readr   1.3.1     v forcats 0.4.0
```

```
## -- Conflicts ---------------------------------------------------------------------------- tidy
verse_conflicts() --

## x dplyr::filter() masks stats::filter()

## x dplyr::lag()    masks stats::lag()
```

```
library(readr)
```

```
# Load data and print head

direct_marketing <- read_csv("direct_marketing.csv",

    col_types = cols(Catalogs = col_integer(),

        Children = col_integer()))

head(direct_marketing)
```

```
## # A tibble: 6 x 10

##   Age   Gender OwnHome Married Location Salary Children History Catalogs

##   <chr> <chr>  <chr>   <chr>   <chr>     <dbl>    <int> <chr>      <int>

## 1 Old   Female Own     Single  Far       47500        0 High           6

## 2 Midd~ Male   Rent    Single  Close     63600        0 High           6

## 3 Young Female Rent    Single  Close     13500        0 Low           18

## 4 Midd~ Male   Own     Married Close     85600        1 High          18

## 5 Midd~ Female Own     Single  Close     68400        0 High          12

## 6 Young Male   Own     Married Close     30400        0 Low            6

## # ... with 1 more variable: AmountSpent <dbl>
```

# Question 1

Creating indicator variables for the ‘History’ column. Considering the base case as None (i.e create Low, Medium and High variables with 1 denoting the positive case and 0 the negative).

Creating variables LowSalary, MediumSalary and HighSalary based on the customer history type i.e., Medium Salary = Medium*Salary

```
direct_marketing <- direct_marketing %>%

  mutate(Low = ifelse(History == "Low",1,0))%>%

  mutate(Medium = ifelse(History == "Medium",1,0))%>%

  mutate(High = ifelse(History == "High",1,0))%>%

  mutate(LowSalary = Low*Salary)%>%

  mutate(MediumSalary = Medium*Salary)%>%

  mutate(HighSalary = High*Salary)
```

# Part a: Fit a multiple linear regression model using AmountSpent as the response variable and the indicator variables along with their salary variables as the predictors

```
# create model using 'lm' and print summary

model = lm(AmountSpent~Salary + Low + Medium + High + LowSalary + MediumSalary + HighSalary,data
= direct_marketing)

summary(model)
```

```
##
## Call:
## lm(formula = AmountSpent ~ Salary + Low + Medium + High + LowSalary +
##     MediumSalary + HighSalary, data = direct_marketing)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -214.33  -25.47   -6.46   20.64  352.50
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.9622199  6.3880253   0.307 0.758777
## Salary        0.0023641  0.0001071  22.083  < 2e-16 ***
## Low          25.4466733  8.9203292   2.853 0.004426 **
## Medium       79.2984388 12.8982169   6.148 1.14e-09 ***
## High         72.6735221 15.2270169   4.773 2.09e-06 ***
## LowSalary    -0.0021069  0.0001890 -11.150  < 2e-16 ***
## MediumSalary -0.0021153  0.0002182  -9.693  < 2e-16 ***
## HighSalary   -0.0006408  0.0001926  -3.328 0.000908 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 55.79 on 992 degrees of freedom
## Multiple R-squared:  0.6654, Adjusted R-squared:  0.6631
## F-statistic: 281.9 on 7 and 992 DF,  p-value: < 2.2e-16
```

Statistically significant variables: All the variables are statistically significant within a 95% confidence interval

# Part b: What is the amount spent by a customer for each historic type provided his/her salary is $10,000?

```
# prepare prediction data and use 'predict' to find the value

pred_data = data.frame(Salary = 10000, High = 1, Medium = 0, Low = 0, HighSalary = 10000, Medium
Salary = 0, LowSalary = 0)%>%

add_row(Salary = 10000, High = 0, Medium = 1, Low = 0, HighSalary = 0, MediumSalary = 10000, Low
Salary = 0)%>%

add_row(Salary = 10000, High = 0, Medium = 0, Low = 1, HighSalary = 0, MediumSalary = 0, LowSala
ry = 10000)%>%

add_row(Salary = 10000, High = 0, Medium = 0, Low = 0, HighSalary = 0, MediumSalary = 0, LowSala
ry = 0)

predict(model, pred_data)
```

```
##        1        2        3        4

## 91.86874 83.74909 29.98157 25.60347
```

It is $91.87 for a customer with ‘High’ history

It is $83.74 for a customer with ‘Medium’ history

It is $29.98 for a customer with ‘Low’ history

It is $25.60 for a customer with ‘No’ history i.e., History = None

Performing Log transformation for the variables Price and overall_satisfaction and create new variables log_Price, log_OverallSatisfaction respectively.Note: Since few values of the overall_satisfaction are zero take the log transformation for overall_satisfaction+1

```
airbnb_data <- read_csv("airbnb_data.csv")
```

```
## Parsed with column specification:

## cols(

##   room_id = col_double(),

##   survey_id = col_double(),

##   host_id = col_double(),

##   room_type = col_character(),

##   city = col_character(),

##   reviews = col_double(),

##   overall_satisfaction = col_double(),

##   accommodates = col_double(),

##   bedrooms = col_double(),

##   price = col_double()

## )
```

```
airbnb_data <- airbnb_data %>%

  mutate(log_OverallSatisfaction = log(overall_satisfaction+1))%>%

  mutate(log_Price = log(price))
```

# Part c: Fit all four models i.e., linear-linear, linear-log, log-linear and log-log regression models using price as the response variable and overall_satisfaction as the predictor.

```
#Linear-Linear Model

model2 = lm(price~overall_satisfaction,data = airbnb_data)

summary(model2)
```

```
##
## Call:
## lm(formula = price ~ overall_satisfaction, data = airbnb_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -167.0  -51.3  -24.2   16.8 4805.0
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           194.967     17.698  11.016  < 2e-16 ***
## overall_satisfaction  -16.353      3.903  -4.189 3.09e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 200.4 on 852 degrees of freedom
## Multiple R-squared:  0.02018,    Adjusted R-squared:  0.01903
## F-statistic: 17.55 on 1 and 852 DF,  p-value: 3.088e-05
```

```
#Linear-Log Model
model3 = lm(price~log_OverallSatisfaction,data = airbnb_data)
summary(model3)
```

```
##
## Call:
## lm(formula = price ~ log_OverallSatisfaction, data = airbnb_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -168.5  -50.7  -24.7   16.3 4803.5
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)               196.46      17.76  11.062  < 2e-16 ***
## log_OverallSatisfaction   -46.20      10.84  -4.263 2.24e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 200.4 on 852 degrees of freedom
## Multiple R-squared:  0.02089,    Adjusted R-squared:  0.01974
## F-statistic: 18.18 on 1 and 852 DF,  p-value: 2.239e-05
```

```
#Log-Linear Model
model4 = lm(log_Price~overall_satisfaction,data = airbnb_data)
summary(model4)
```

```
##
## Call:
## lm(formula = log_Price ~ overall_satisfaction, data = airbnb_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6234 -0.3525 -0.0432  0.3302  3.7220
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           4.79515    0.05083  94.339  < 2e-16 ***
## overall_satisfaction -0.04401    0.01121  -3.926 9.33e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5757 on 852 degrees of freedom
## Multiple R-squared:  0.01777,    Adjusted R-squared:  0.01662
## F-statistic: 15.41 on 1 and 852 DF,  p-value: 9.331e-05
```

```
#Log-Log Model
model5 = lm(log_Price~log_OverallSatisfaction,data = airbnb_data)
summary(model5)
```

```
##
## Call:
## lm(formula = log_Price ~ log_OverallSatisfaction, data = airbnb_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6030 -0.3551 -0.0327  0.3298  3.7132
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)              4.80396    0.05098  94.228  < 2e-16 ***
## log_OverallSatisfaction -0.12750    0.03111  -4.099 4.55e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5752 on 852 degrees of freedom
## Multiple R-squared:  0.01934,    Adjusted R-squared:  0.01819
## F-statistic:  16.8 on 1 and 852 DF,  p-value: 4.547e-05
```

# Part d: Which of the four models has the best R^2? Do you have any comments on the choice of the dependent variable.

The linear -log model has the highest R^2, Note that R^2 value are very insignificant as this is not a very good predictor variable.