Housing Price Analysis Appendix

Yangsu Roy Wang

1) Import Data

Establish a housing price dataset by reading the excel data file and assign the contents to a variable `housing_prices`.

```
require(tidyverse)
```

```
## Loading required package: tidyverse
```

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

```
## v ggplot2 3.3.1     v purrr   0.3.4
## v tibble  3.0.1     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
require(readxl)
```

```
## Loading required package: readxl
```

```
housing_prices <- read_excel("data/Housing_prices_revised.xlsx", col_types = c("numeric", "numeric", "nu
                                          "numeric"))
```

2) Models and Statistics

a) Perform a multiple regression analysis to test whether house selling price can be effectively predicted by number of rooms and location.

```
model <- lm(Selling.Price ~ Location + Number.of.Rooms, housing_prices)
summary(model)
```

```
##
## Call:
## lm(formula = Selling.Price ~ Location + Number.of.Rooms, data = housing_prices)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -249.47  -52.99  -11.03   67.73  159.16
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       155.29     107.88   1.439 0.168191
## Location         -222.04      52.59  -4.222 0.000574 ***
## Number.of.Rooms    54.90      10.60   5.177 7.58e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 102.9 on 17 degrees of freedom
## Multiple R-squared:  0.8348, Adjusted R-squared:  0.8153
## F-statistic: 42.95 on 2 and 17 DF,  p-value: 2.257e-07
```

b) Construct an ANOVA table using the aforementioned multiple regression analysis.

```
anova(model)
```

```
## Analysis of Variance Table
##
## Response: Selling.Price
##                 Df Sum Sq Mean Sq F value    Pr(>F)
## Location         1 625872  625872  59.090 6.258e-07 ***
## Number.of.Rooms  1 283900  283900  26.804 7.576e-05 ***
## Residuals       17 180062   10592
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

c) Calculate the prediced selling price for a house with 9 rooms and is located in Dandenong.

```
model$coefficients[1] + model$coefficients[2] * 0 + model$coefficients[3] * 9
```

```
## (Intercept)
##    649.3713
```

d) Add a joint term between number of rooms and locations. Build a multiple regression model with interaction effects between two independent variables.

```
model_joint <- lm(Selling.Price ~ Location + Number.of.Rooms + Location : Number.of.Rooms, housing_pric
summary(model_joint)
```

```
##
## Call:
## lm(formula = Selling.Price ~ Location + Number.of.Rooms + Location:Number.of.Rooms,
##     data = housing_prices)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -233.12  -57.50   -5.01   47.19  168.49
##
```

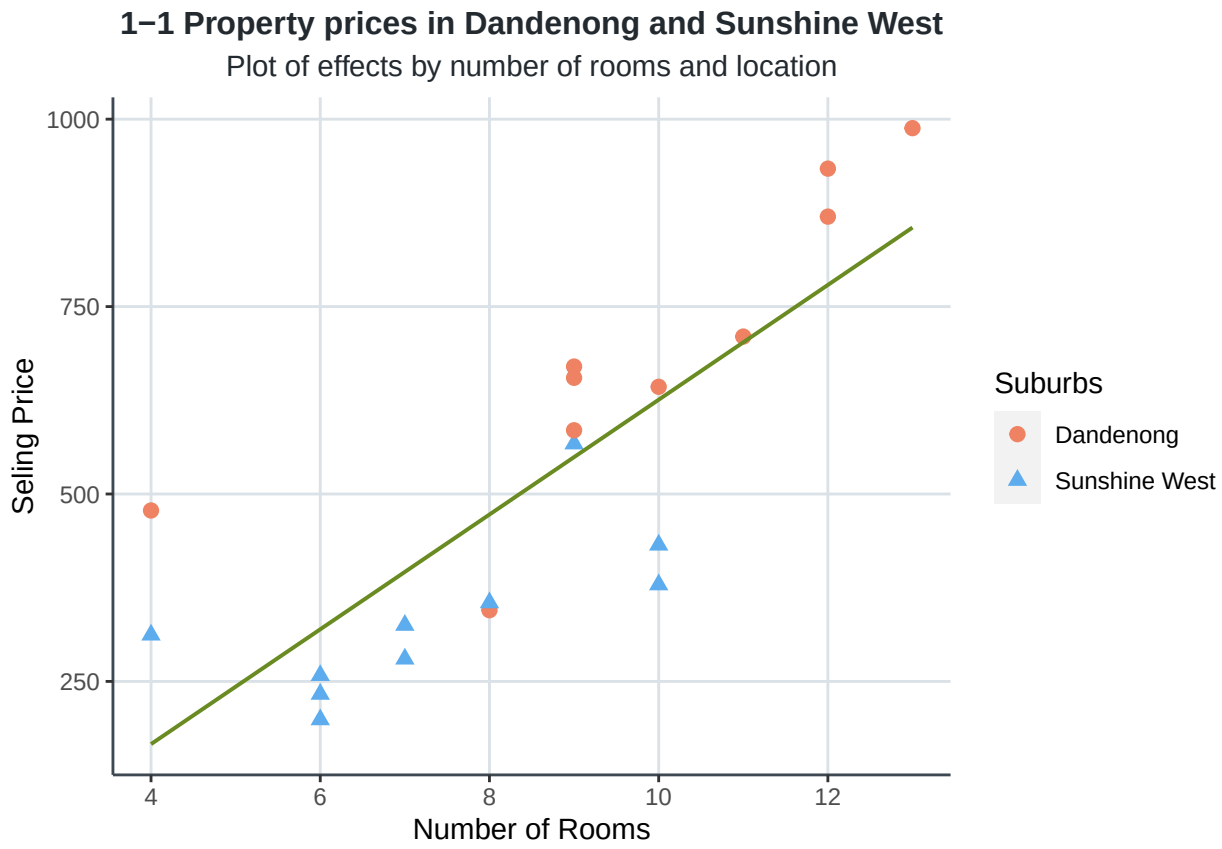
```
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  61.997    130.940   0.473 0.642271
## Location                     -5.011    185.099  -0.027 0.978736
## Number.of.Rooms              64.516     13.087   4.930 0.000151 ***
## Location:Number.of.Rooms    -26.569     21.752  -1.221 0.239620
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 101.5 on 16 degrees of freedom
## Multiple R-squared:  0.8489, Adjusted R-squared:  0.8205
## F-statistic: 29.96 on 3 and 16 DF,  p-value: 8.452e-07
```

3) Visualisation

a) Use a single linear regression model to find the relationship between house *selling price* and *number of rooms*. Plot the regression line. Differentiate the plots by properties in two different suburbs (Dandenong and Sunshine West) using different colour and shape.

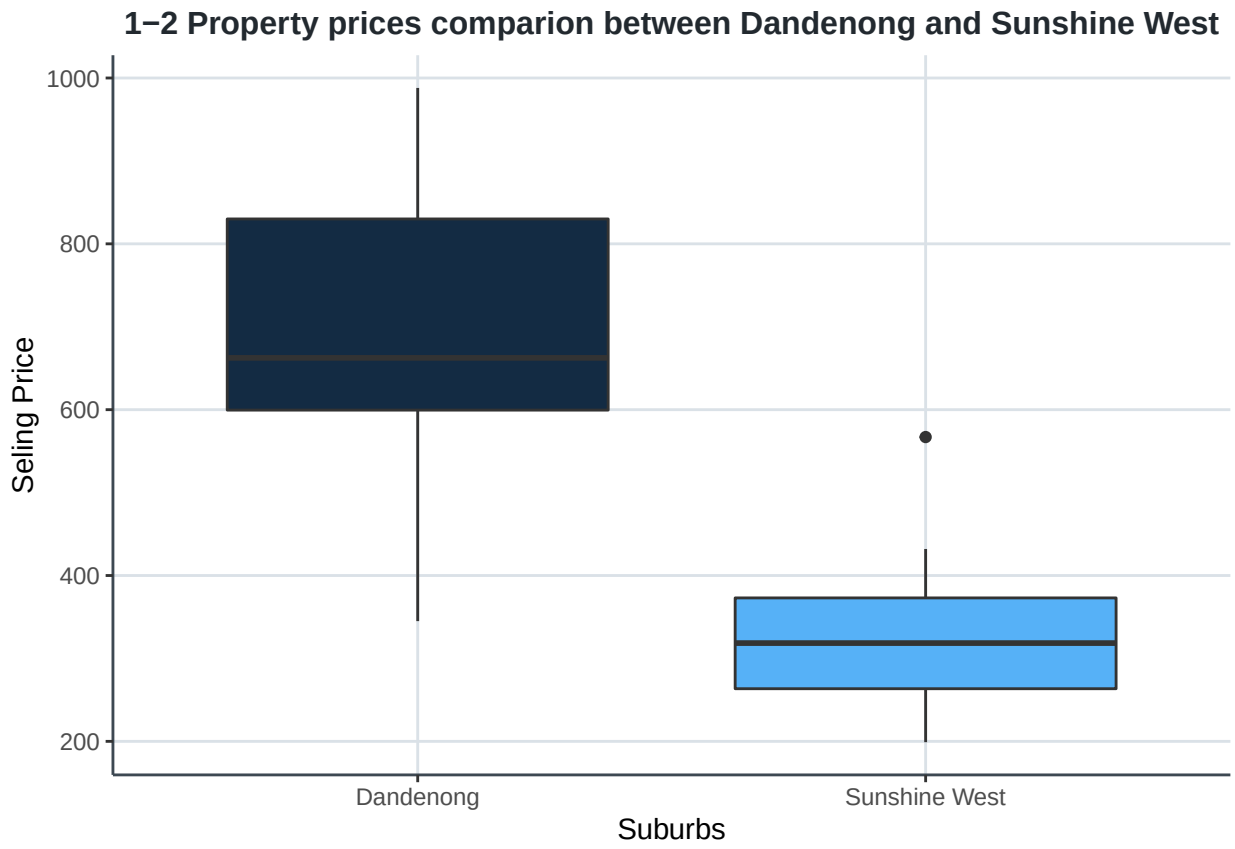
```
ggplot(housing_prices) +
  geom_point(size = 2.2, aes(x = Number.of.Rooms, y = Selling.Price, colour = as.factor(Location),
             shape = as.factor(Location))) +
  geom_smooth(size = 0.7, aes(x = Number.of.Rooms, y = Selling.Price), method = lm, se = FALSE,
              colour = "olivedrab4") +
  labs(title="1-1 Property prices in Dandenong and Sunshine West",
       subtitle = "Plot of effects by number of rooms and location",
       x="Number of Rooms", y = "Seling Price") +
  scale_shape_manual("Suburbs", values = c(19,17), labels = c("Dandenong", "Sunshine West")) +
  scale_colour_manual("Suburbs", values = c("salmon2","steelblue2"), labels = c("Dandenong", "Sunshine
  theme(plot.title = element_text(color = "#22292F", size = 12, face = "bold", hjust = 0.5),
        plot.subtitle = element_text(color = "#22292F", hjust = 0.5),
        panel.background = element_blank(),
        panel.grid.major = element_line(color = "#DAE1E7"), panel.grid.major.x = element_line(color = "
        axis.line = element_line(color = "#3D4852"))
```

```
## `geom_smooth()` using formula 'y ~ x'
```



b) Use a simple boxplot to demonstrate effects of house prices differentiated by locations.

```
ggplot(housing_prices) +
  geom_boxplot(aes(x = as.factor(Location), y = Selling.Price, fill = Location)) +
  labs(title="1-2 Property prices comparion between Dandenong and Sunshine West",
       x="Suburbs", y = "Seling Price") +
  theme(legend.position = "none", plot.title = element_text(color = "#22292F",size = 12, face = "bold",
        hjust = 0.5), plot.subtitle = element_text(hjust = 0.5),
        panel.background = element_blank(),
        panel.grid.major = element_line(color = "#DAE1E7"), panel.grid.major.x = element_line(color = "
        axis.line = element_line(color = "#3D4852")) +
  scale_x_discrete(labels = c("0" = "Dandenong", "1" = "Sunshine West"))
```

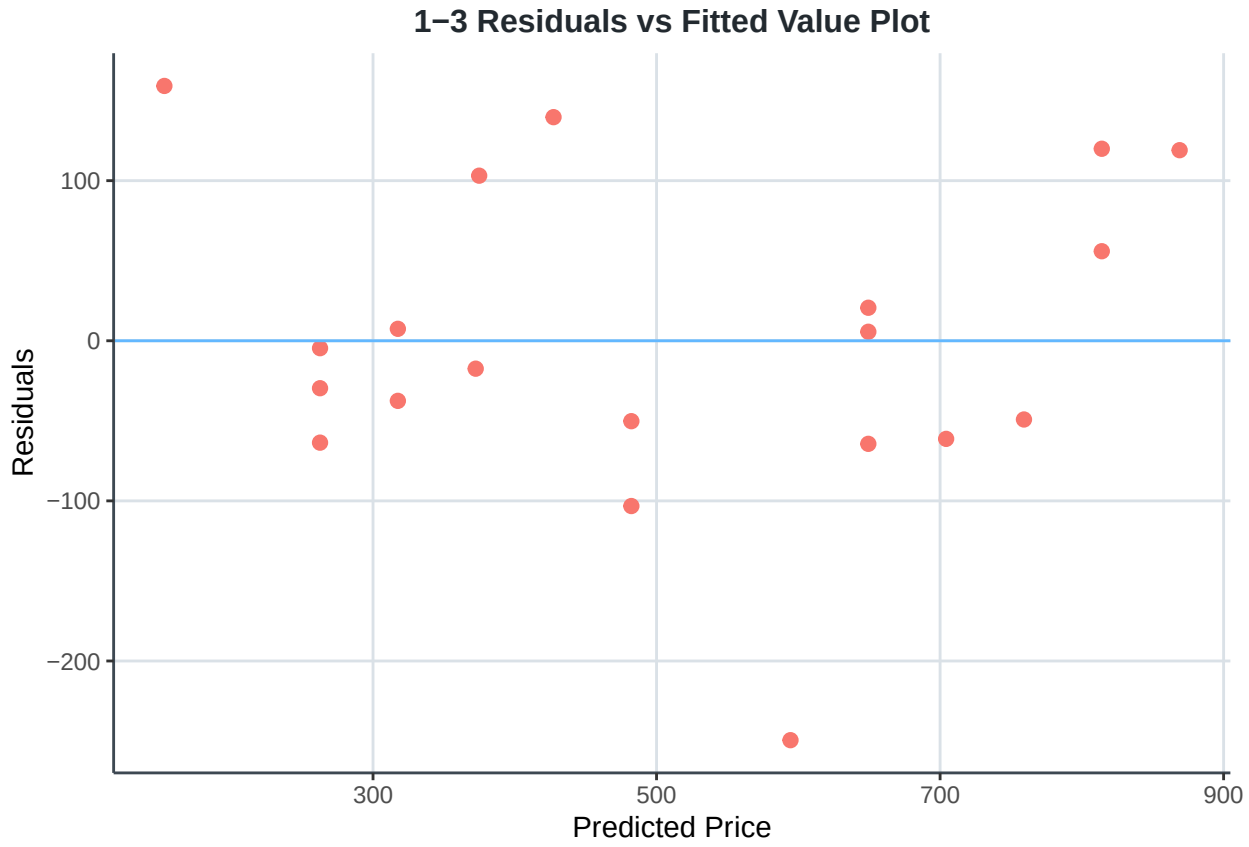


c) Assign a new dataset `housing_prices_output` by adding columns *Predicted Price* and *Residuals*.

```
housing_prices_output <- cbind(housing_prices)
housing_prices_output <-
  housing_prices_output %>%
  mutate("Predicted.Price" = model$fitted.values) %>%
  mutate("Residuals" = model$residuals)
```

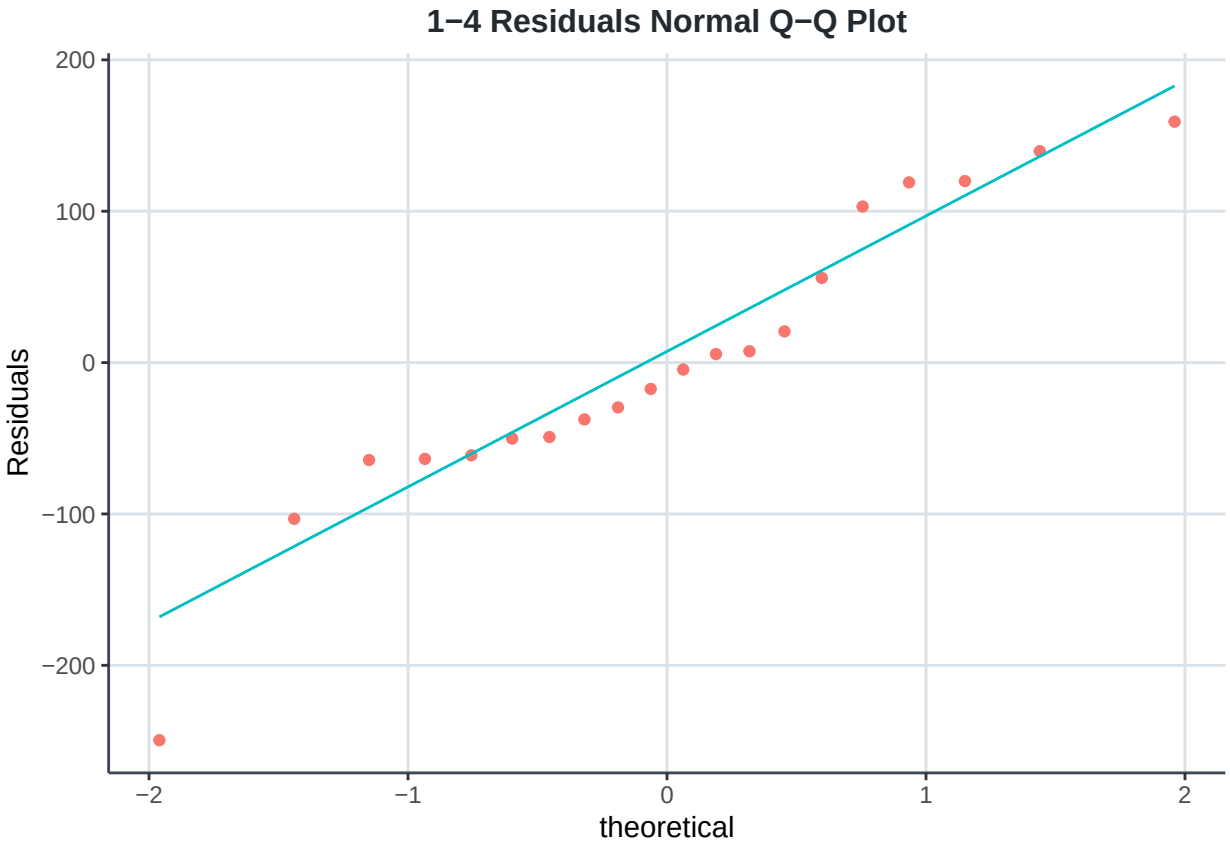
d) Create a residuals plot to check the pattern of residuals for the regression model.

```
ggplot(housing_prices_output, aes(x = Predicted.Price, y = Residuals)) +
  geom_point(size = 2.2, aes(colour = "salmon2")) +
  geom_hline(yintercept = 0, color = "steelblue1") +
  labs(title="1-3 Residuals vs Fitted Value Plot", x = "Predicted Price", y = "Residuals") +
  theme(legend.position = "none", plot.title = element_text(color = "#22292F",size = 12, face = "bold",
        hjust = 0.5), panel.background = element_blank(), panel.grid.major = element_line(color = "#DAE
        panel.grid.major.x = element_line(color = "#DAE1E7"), axis.line = element_line(color = "#3D4852
```



e) Create a Normal Quantile-Quantile (Q-Q) Plot to test the normality of the data residuals.

```
ggplot(housing_prices_output, aes(sample = Residuals)) +
  stat_qq(aes(colour = "salmon2")) +
  stat_qq_line(aes(colour = "steelblue1")) +
  labs(title="1-4 Residuals Normal Q-Q Plot", y = "Residuals") +
  theme(legend.position = "none", plot.title = element_text(color = "#22292F",size = 12, face = "bold",
        hjust = 0.5), panel.background = element_blank(),
        panel.grid.major = element_line(color = "#DAE1E7"), panel.grid.major.x = element_line(color = "
        axis.line = element_line(color = "#3D4852"))
```


1−4 Residuals Normal Q−Q Plot
Residuals
200
100
0
−100
−200
−2
−1
0
1
2
theoretical