````
---
title: "Data Analysis for House Sales"

## Importing required packages
library(tidyverse)
library(lmtest)
library(ggpubr)
library(broom)
library(ggfortify)
library(skimr)


## Import the built-in R data set
data("txhousing")

## View and check the dimension of the data set
View(txhousing)
dim(txhousing)

## Check the column names for the data set
names(txhousing)
```


## Use R functions to describe the data

```{r}
## Take a peek using the head and tail functions
head(txhousing)
tail(txhousing)

## Check the internal structure of the data frame
glimpse(txhousing)

## Create a broad overview of a data set
skim(txhousing)

## Drop the missing values in sales, volume, median
tx_data <- txhousing %>%
  drop_na(sales,volume,median)

## Create the age variable
tx_data$age <- 2023 - tx_data$year

## Create a broad overview of a data set
skim(tx_data)

```


## Create data visualization using ggplot
# A scatter plot to visualize the variables for model building
#```{r}
## Find the correlation between the variables
cor(tx_data$sales, tx_data$volume)

## Plot a scatter plot for the variables with sales on the x-axis
## volume on the y-axis
ggplot(tx_data, aes(x = sales, y = volume)) +
  geom_point() +
  stat_smooth(se = FALSE)

```


## Build a regression model
````

```
## Create a simple linear regression model using the variables
simple_model <- lm(volume ~ sales, data = tx_data )
simple_model

## Plot the regression line for the model
ggplot(tx_data, aes(x = sales, y = volume)) +
  geom_point() +
  stat_smooth(method = lm) +
  labs(title = "Regression of Sales on Volume of Housing sales in TX",
       x = "Sales", y = "Volume")
       scale_y_continuous(labels = scales::comma)
#```

## Perform diagnostic checks on fitted model

## Plotting the fitted model
plot(simple_model)

## Return the first diagnostic plot for the model
plot(simple_model,3)

## Create all four plots at once


```

Perform model fit assessment

Assess the summary of the fitted model
summary(simple_model)

Calculate the confidence interval for the coefficients
confint(simple_model)

Make predictions using the fitted model

Find the fitted values of the simple regression model
fitted_values <- predict.lm(simple_model)
head(fitted_values, 3)

Return the model metrics
model_metrics <- augment(simple_model)
model_metrics

Predict new values using the model

predict(simple_model,
  newdata = data.frame(sales = c(210, 27, 140)))

Multiple Regression

Build the multiple regression model with volume as the y variable and sales, median and
age on the x variables
multiple_reg <- lm(volume ~ sales + median + age, data = tx_data)

This prints the result of the model
multiple_reg

Check the summary of the multiple regression model
```
```

```
summary(multiple_reg)

## Plot the fitted multiple regression model
autoplot(multiple_reg)
```