HW3

Yoonseo Mok

Github link https://github.com/mokys1213/STATS-506-FA-2024/blob/main/HW4/HW4.pdf

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggplot2)
```

Problem 1

```
# Installing and loading nycflights13
# install.packages("nycflights13")
library(nycflights13)
```

A

```
# Left joining airports dataset to get the name of the airport
tmp1=flights %>% left_join(airports %>% rename(origin = faa))
```

```
## Joining with `by = join_by(origin)`
```

```
# Generating a table reporting the mean and median departure delay per airport
tmp1 %>% group_by(name) %>% summarize(mean_delay=mean(dep_delay,na.rm = T),
                                      median_delay=median(dep_delay,na.rm=T),n=n()) %>%
  filter(n>10) %>% arrange(desc(mean_delay)) %>% select(-n)
```

```
## # A tibble: 3 x 3
##   name                mean_delay median_delay
##   <chr>                    <dbl>        <dbl>
## 1 Newark Liberty Intl       15.1           -1
## 2 John F Kennedy Intl       12.1           -1
## 3 La Guardia                10.3           -3
```

```
# Generating a table reporting the mean and median arrival delay per airport
tmp2=flights %>% group_by(dest) %>% summarize(mean_delay=mean(arr_delay,na.rm = T),
                                              median_delay=median(arr_delay,na.rm=T),n=n()) %>%
                 ungroup() %>% filter(n>=10) %>% rename(faa = dest)
tmp2 %>% left_join(airports, by = "faa") %>% select(name, mean_delay, median_delay) %>%
  arrange(desc(mean_delay)) %>% print(n = 1e3)
```

```
## # A tibble: 102 x 3
##     name                                  mean_delay median_delay
##     <chr>                                      <dbl>        <dbl>
##   1 "Columbia Metropolitan"                     41.8           28
##   2 "Tulsa Intl"                                33.7           14
##   3 "Will Rogers World"                         30.6           16
##   4 "Jackson Hole Airport"                      28.1           15
##   5 "Mc Ghee Tyson"                             24.1            2
##   6 "Dane Co Rgnl Truax Fld"                    20.2            1
##   7 "Richmond Intl"                             20.1            1
##   8 "Akron Canton Regional Airport"             19.7            3
##   9 "Des Moines Intl"                           19.0            0
##  10 "Gerald R Ford Intl"                        18.2            1
##  11 "Birmingham Intl"                           16.9           -2
##  12 "Theodore Francis Green State"              16.2            1
##  13 "Greenville-Spartanburg International"      15.9           -0.5
##  14 "Cincinnati Northern Kentucky Intl"         15.4           -3
##  15 "Savannah Hilton Head Intl"                 15.1           -1
##  16 "Manchester Regional Airport"               14.8           -3
##  17 "Eppley Afld"                               14.7           -2
##  18 "Yeager"                                    14.7           -1.5
##  19 "Kansas City Intl"                          14.5            0
##  20 "Albany Intl"                               14.4           -4
##  21 "General Mitchell Intl"                     14.2            0
##  22 "Piedmont Triad"                            14.1           -2
##  23 "Washington Dulles Intl"                    13.9           -3
##  24 "Cherry Capital Airport"                    13.0          -10
##  25 "James M Cox Dayton Intl"                   12.7           -3
##  26 "Louisville International Airport"          12.7           -2
##  27 "Chicago Midway Intl"                       12.4           -1
##  28 "Sacramento Intl"                           12.1            4
##  29 "Jacksonville Intl"                         11.8           -2
##  30 "Nashville Intl"                            11.8           -2
##  31 "Portland Intl Jetport"                     11.7           -4
##  32 "Greater Rochester Intl"                    11.6           -5
##  33 "Hartsfield Jackson Atlanta Intl"           11.3           -1
##  34 "Lambert St Louis Intl"                     11.1           -3
##  35 "Norfolk Intl"                              10.9           -4
##  36 "Baltimore Washington Intl"                 10.7           -5
##  37 "Memphis Intl"                              10.6           -2.5
```

```
##  38 "Port Columbus Intl"                     10.6          -3
##  39 "Charleston Afb Intl"                    10.6          -4
##  40 "Philadelphia Intl"                      10.1          -3
##  41 "Raleigh Durham Intl"                    10.1          -3
##  42 "Indianapolis Intl"                       9.94         -3
##  43 "Charlottesville-Albemarle"               9.5          -5
##  44 "Cleveland Hopkins Intl"                  9.18         -5
##  45 "Ronald Reagan Washington Natl"           9.07         -2
##  46 "Burlington Intl"                         8.95         -4
##  47 "Buffalo Niagara Intl"                    8.95         -5
##  48 "Syracuse Hancock Intl"                   8.90         -5
##  49 "Denver Intl"                             8.61         -2
##  50 "Palm Beach Intl"                         8.56         -3
##  51  <NA>                                     8.25         -1
##  52 "Bob Hope"                                8.18         -3
##  53 "Fort Lauderdale Hollywood Intl"          8.08         -3
##  54 "Bangor Intl"                             8.03         -9
##  55 "Asheville Regional Airport"              8.00         -1
##  56  <NA>                                     7.87          0
##  57 "Pittsburgh Intl"                         7.68         -5
##  58 "Gallatin Field"                          7.6          -2
##  59 "NW Arkansas Regional"                    7.47         -2
##  60 "Tampa Intl"                              7.41         -4
##  61 "Charlotte Douglas Intl"                  7.36         -3
##  62 "Minneapolis St Paul Intl"                7.27         -5
##  63 "William P Hobby"                         7.18         -4
##  64 "Bradley Intl"                            7.05        -10
##  65 "San Antonio Intl"                        6.95         -9
##  66 "South Bend Rgnl"                         6.5          -3.5
##  67 "Louis Armstrong New Orleans Intl"        6.49         -6
##  68 "Key West Intl"                           6.35          7
##  69 "Eagle Co Rgnl"                           6.30         -4
##  70 "Austin Bergstrom Intl"                   6.02         -5
##  71 "Chicago Ohare Intl"                      5.88         -8
##  72 "Orlando Intl"                            5.45         -5
##  73 "Detroit Metro Wayne Co"                  5.43         -7
##  74 "Portland Intl"                           5.14         -5
##  75 "Nantucket Mem"                           4.85         -3
##  76 "Wilmington Intl"                         4.64         -7
##  77 "Myrtle Beach Intl"                       4.60        -13
##  78 "Albuquerque International Sunport"       4.38         -5.5
##  79 "George Bush Intercontinental"            4.24         -5
##  80 "Norman Y Mineta San Jose Intl"           3.45         -7
##  81 "Southwest Florida Intl"                  3.24         -5
##  82 "San Diego Intl"                          3.14         -5
##  83 "Sarasota Bradenton Intl"                 3.08         -5
##  84 "Metropolitan Oakland Intl"               3.08         -9
##  85 "General Edward Lawrence Logan Intl"      2.91         -9
##  86 "San Francisco Intl"                      2.67         -8
##  87  <NA>                                     2.52         -6
##  88 "Yampa Valley"                            2.14          2
##  89 "Phoenix Sky Harbor Intl"                 2.10         -6
##  90 "Montrose Regional Airport"               1.79        -10.5
##  91 "Los Angeles Intl"                        0.547        -7
```

```
##  92 "Dallas Fort Worth Intl"                  0.322          -9
##  93 "Miami Intl"                              0.299          -9
##  94 "Mc Carran Intl"                          0.258          -8
##  95 "Salt Lake City Intl"                     0.176          -8
##  96 "Long Beach"                             -0.0620        -10
##  97 "Martha\\\\'s Vineyard"                  -0.286         -11
##  98 "Seattle Tacoma Intl"                    -1.10          -11
##  99 "Honolulu Intl"                          -1.37           -7
## 100  <NA>                                    -3.84           -9
## 101 "John Wayne Arpt Orange Co"              -7.87          -11
## 102 "Palm Springs Intl"                     -12.7           -13.5
```

B

```
# Identifying how many flights did the aircraft model with the fastest average speed take
flights %>% left_join(planes,by ="tailnum") %>% mutate(time = air_time/60, mph = distance/time) %>%
  group_by(model) %>% summarize(average_speed=mean(mph, na.rm=TRUE),n_flights = n()) %>%
  arrange(desc(average_speed))  %>% slice(1)
```

```
## # A tibble: 1 x 3
##   model   average_speed n_flights
##   <chr>           <dbl>     <int>
## 1 777-222          483.         4
```

Problem 2

```
# Loading chicago-nmmaps data
nnmaps <- read_csv("chicago-nmmaps.csv")
```

```
## Rows: 1461 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): city, season, month
## dbl  (7): temp, o3, dewpoint, pm10, yday, month_numeric, year
## date (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
##' Function for getting average monthly temperature
##' @param month numeric 1-12 or a string.
##' @param year Year
##' @param data The data set to obtain data from.
##' @param average_fn A function with which to compute the mean. Default is mean.
##' @param celsius Logically indicating whther the results should be in celsius. Default FALSE.
##' @return Average temperature
get_temp=function(month, year, data, average_fn=mean, celsius=FALSE) {
```

```
  # Error message for month
  if (is.character(month) %in% TRUE) {
    fullmonth=c("January","February","March","April","May","June","July",
                "August","September","October","November","December")
    abbr=substr(fullmonth, 1, 3)
    if(nchar(month) %in% 3){
      month=match(tolower(month), tolower(abbr))
    } else {
      month=match(tolower(month), tolower(fullmonth))
    }
  } else if (is.numeric(month) %in% TRUE) {
    if (month<1 | month>12) {
      stop("Invalid month")
    }
  } else {
    stop("month must be numeric or character")
  }

  # Error message for year
  if (year<1997 | year>2000) {
    stop("year is not in the data")
  }
  if (!is.numeric(year)) {
    stop("year must be numeric")
  }

  # Error message for average_fn
  if (!(is.function(average_fn))) {
    stop("average_fn must be a function")
  }

  tmp=data %>% select(year,temp,month_numeric) %>% rename(data_year=year) %>%
    filter(data_year %in% year, month_numeric %in% month) %>% # subsetting matching data
    summarize(averagetmp = average_fn(temp)) %>% # calculate average temperature based on function
    mutate(averagetmp=ifelse(isTRUE(celsius), 5/9*(averagetmp-32), averagetmp))

  tmp=as.numeric(tmp)

  return(tmp)
}
```

```
get_temp("Apr", 1999, data = nnmaps)
```

```
## [1] 49.8
```

```
get_temp("Apr", 1999, data = nnmaps, celsius = TRUE)
```

```
## [1] 9.888889
```

```
get_temp(10, 1998, data = nnmaps, average_fn = median)
```

```
## [1] 55
```

```r
get_temp(13, 1998, data = nnmaps)
```

```
## Error in get_temp(13, 1998, data = nnmaps): Invalid month
```

```r
get_temp(2, 2005, data = nnmaps)
```

```
## Error in get_temp(2, 2005, data = nnmaps): year is not in the data
```

```r
get_temp("November", 1999, data =nnmaps, celsius = TRUE,
         average_fn = function(x) {
           x %>% sort -> x
           x[2:(length(x) - 1)] %>% mean %>% return
         })
```

```
## [1] 7.301587
```

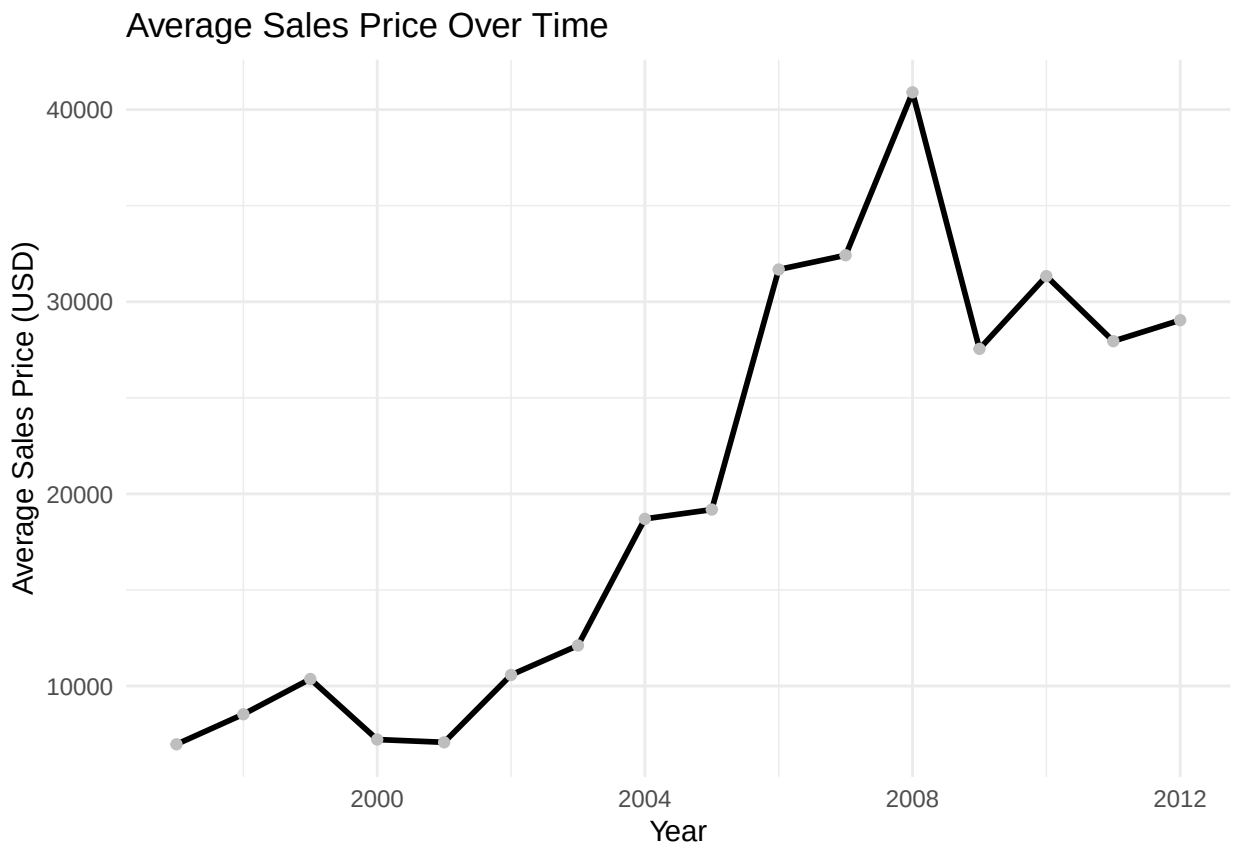
Problem 3

```r
# Loading the dataset
p3=read_csv("df_for_ml_improved_new_market.csv")
```

```
## Rows: 4347 Columns: 112
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (1): eventdate
## dbl (111): id, case_id, year, height, width, size_inchsqr, price_usd, meanpr...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

A

```r
# Generating dataframe to calculate the mean price by year
tmpa <- p3 %>%
  group_by(year) %>%
  summarize(average_price = mean(price_usd, na.rm = TRUE))

# Creating line plot
ggplot(tmpa, aes(x = year, y = average_price)) +
  geom_line(color = "black", linewidth = 1) +
  geom_point(color = "grey") +
  labs(title = "Average Sales Price Over Time")+ xlab("Year")+
  ylab("Average Sales Price (USD)") +
  theme_minimal()
```



Looking at the plot, we see a continuous increase in average sales price from year 2001 to 2008 (being the highest), and sharp decrease in 2009. This suggests that there was a change in average sales price over time.

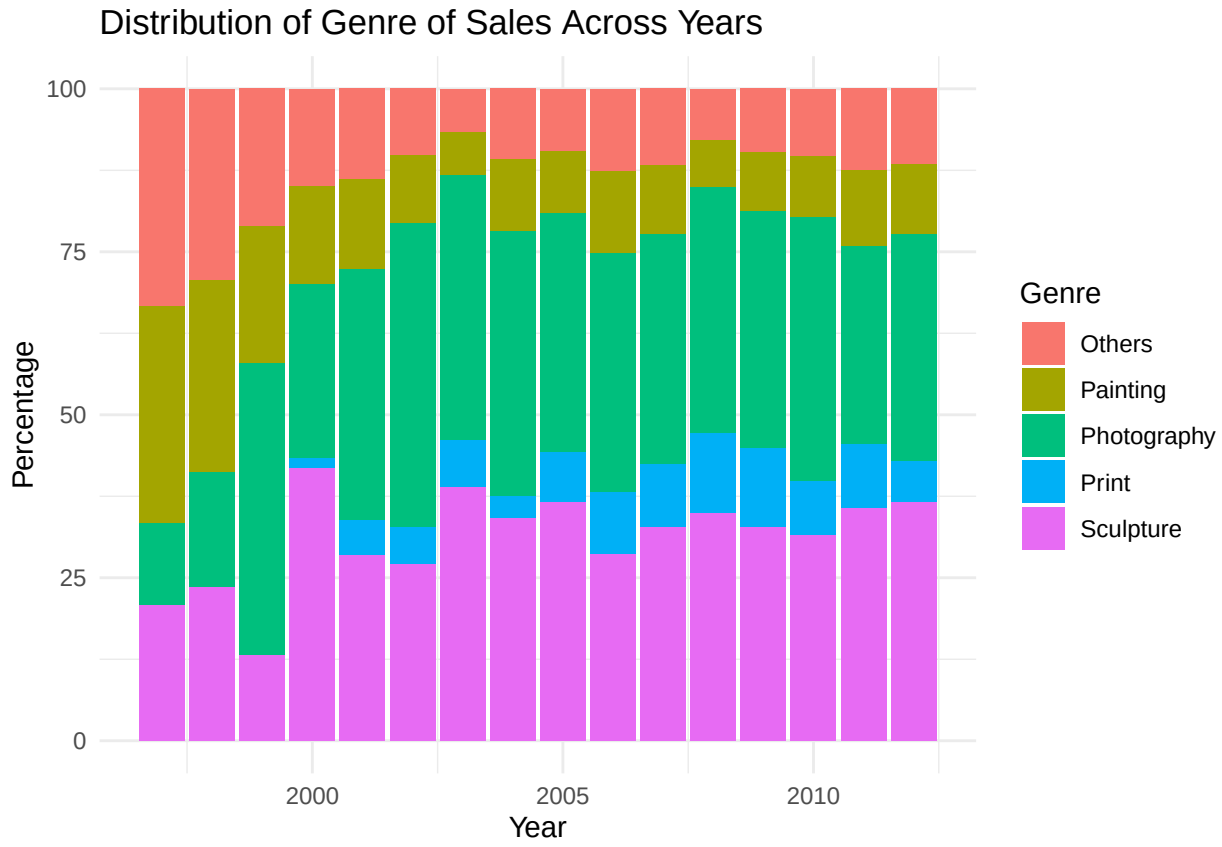
B

```
# Renaming genre related columns
p3=p3 %>% rename(Photography=Genre___Photography,Painting=Genre___Painting,
                 Sculpture=Genre___Sculpture,Print=Genre___Print,Others=Genre___Others)

# Making a long dataset that indicates genre for each painting
p3b=p3 %>% pivot_longer(cols=c(Photography,Painting,Sculpture,Print,Others),
                        names_to = "genre",values_to = "count") %>% filter(count %in% 1)

# Generating percentage within each year for each genre
p3b2=p3b %>% group_by(year, genre) %>% summarise(count = n()) %>% ungroup() %>%
  group_by(year) %>% mutate(percent = count / sum(count) * 100)
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```
# Generating stacked bar plot
ggplot(p3b2, aes(x = year, y=percent, fill = genre)) +
  geom_bar(stat = "identity", position = "stack") +
```

```
  labs(title = "Distribution of Genre of Sales Across Years",fill="Genre")+
  xlab("Year")+ylab("Percentage")+
  theme_minimal()
```



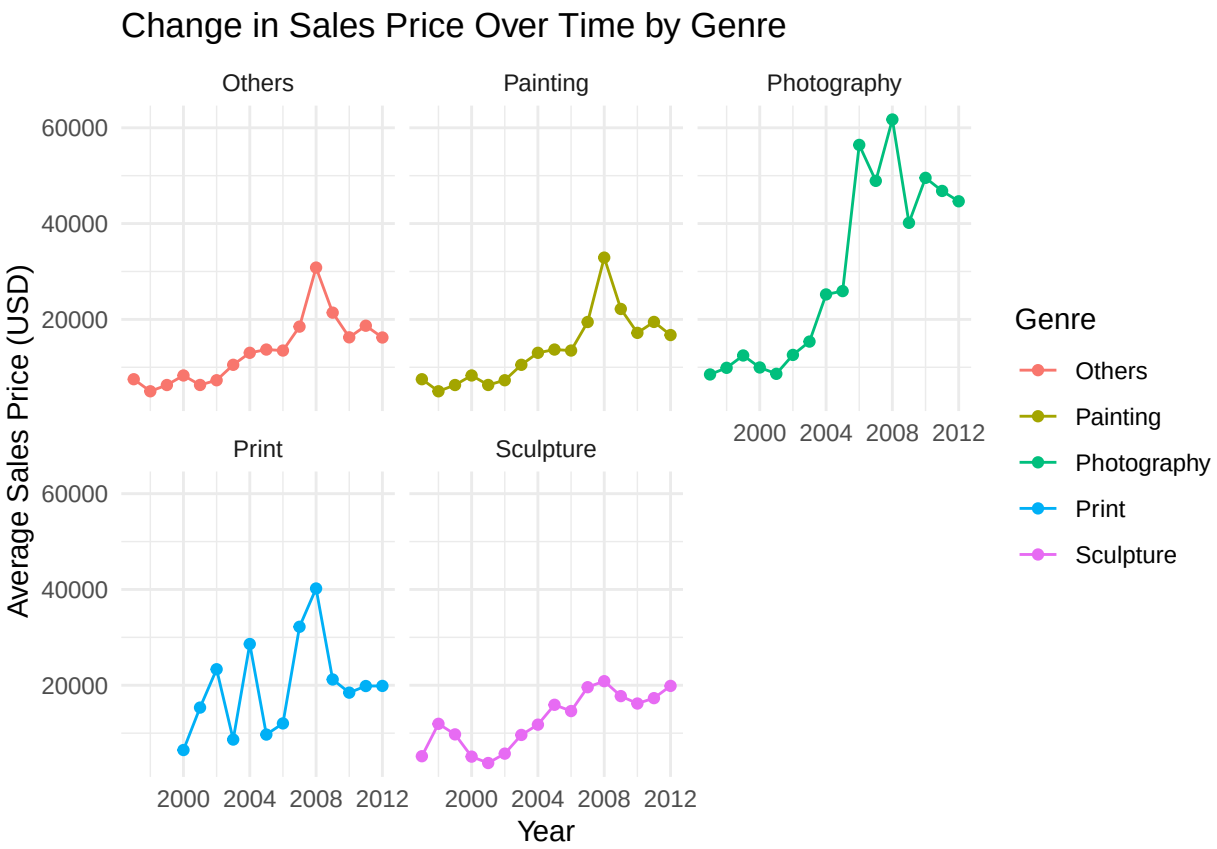
Looking at the plot, the distribution of genre of sales across years appear to change. There was no print genre in early years, but it started to come out from year 2000. Painting genre had the most percentage in year 1997, but it started to decrease. More of the photography genre came out from year 1999.

C

```
# Calculate average price per genre per year
p3c=p3b %>% group_by(year, genre) %>% summarize(average_price = mean(price_usd, na.rm = TRUE)) %>% rena
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```
# Generating line plot for each genre
ggplot(p3c, aes(x = year, y = average_price, color = Genre)) +
  geom_line()+geom_point() +
  labs(title = "Change in Sales Price Over Time by Genre",
       x = "Year",
       y = "Average Sales Price (USD)") +
  facet_wrap(~Genre)+
  theme_minimal()
```



For photography genre, it had a sharp increase in average sales price in 2006, and slight decrease in 2007 but reached the peak in 2008. Painting and others genre has a similar pattern, having increasing trend and reaching its peak in 2008. Print genre has up and downs in average sales price. Sculpture genre has an increase in average sales price across the years except the early years.