hw4

Github:https://github.com/yyhken/stats-506

Question1

a.

```
library(nycflights13)
library(tidyverse)
```

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

```
# Join flights data with airport names for departure delays
departure_delays <- flights %>%
  left_join(airports, by = c("origin" = "faa")) %>%
  group_by(name) %>%
  summarise(
    mean_dep_delay = mean(dep_delay, na.rm = TRUE),
    median_dep_delay = median(dep_delay, na.rm = TRUE),
    n_flights = n()
  ) %>%
  filter(n_flights >= 10) %>%
  arrange(desc(mean_dep_delay)) %>%
  select(name, mean_dep_delay, median_dep_delay)

# Print the departure delay table
print(departure_delays, n = Inf)
```

```
# A tibble: 3 × 3
  name                mean_dep_delay median_dep_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
```

```
# Join flights data with airport names for arrival delays
arrival_delays <- flights %>%
  left_join(airports, by = c("dest" = "faa")) %>%
```

```
  group_by(name) %>%
  summarise(
    mean_arr_delay = mean(arr_delay, na.rm = TRUE),
    median_arr_delay = median(arr_delay, na.rm = TRUE),
    n_flights = n()
  ) %>%
  filter(n_flights >= 10) %>%
  arrange(desc(mean_arr_delay)) %>%
  select(name, mean_arr_delay, median_arr_delay)

# Print the arrival delay table
print(arrival_delays, n = Inf)
```

```
# A tibble: 99 × 3
   name                                mean_arr_delay median_arr_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	"Bob Hope"	8.18	-3
52	"Fort Lauderdale Hollywood Intl"	8.08	-3
53	"Bangor Intl"	8.03	-9
54	"Asheville Regional Airport"	8.00	-1
55	"Pittsburgh Intl"	7.68	-5
56	"Gallatin Field"	7.6	-2
57	"NW Arkansas Regional"	7.47	-2
58	"Tampa Intl"	7.41	-4
59	"Charlotte Douglas Intl"	7.36	-3
60	"Minneapolis St Paul Intl"	7.27	-5
61	"William P Hobby"	7.18	-4
62	"Bradley Intl"	7.05	-10
63	"San Antonio Intl"	6.95	-9
64	"South Bend Rgnl"	6.5	-3.5
65	"Louis Armstrong New Orleans Intl"	6.49	-6
66	"Key West Intl"	6.35	7
67	"Eagle Co Rgnl"	6.30	-4
68	"Austin Bergstrom Intl"	6.02	-5
69	"Chicago Ohare Intl"	5.88	-8
70	"Orlando Intl"	5.45	-5
71	"Detroit Metro Wayne Co"	5.43	-7
72	"Portland Intl"	5.14	-5
73	"Nantucket Mem"	4.85	-3
74	"Wilmington Intl"	4.64	-7
75	"Myrtle Beach Intl"	4.60	-13
76	"Albuquerque International Sunport"	4.38	-5.5
77	"George Bush Intercontinental"	4.24	-5
78	"Norman Y Mineta San Jose Intl"	3.45	-7
79	"Southwest Florida Intl"	3.24	-5
80	"San Diego Intl"	3.14	-5
81	"Sarasota Bradenton Intl"	3.08	-5
82	"Metropolitan Oakland Intl"	3.08	-9
83	<NA>	3.01	-5
84	"General Edward Lawrence Logan Intl"	2.91	-9
85	"San Francisco Intl"	2.67	-8
86	"Yampa Valley"	2.14	2
87	"Phoenix Sky Harbor Intl"	2.10	-6
88	"Montrose Regional Airport"	1.79	-10.5
89	"Los Angeles Intl"	0.547	-7
90	"Dallas Fort Worth Intl"	0.322	-9
91	"Miami Intl"	0.299	-9
92	"Mc Carran Intl"	0.258	-8

```
93 "Salt Lake City Intl"                    0.176             -8
94 "Long Beach"                            -0.0620           -10
95 "Martha\\\\'s Vineyard"                 -0.286            -11
96 "Seattle Tacoma Intl"                   -1.10             -11
97 "Honolulu Intl"                         -1.37              -7
98 "John Wayne Arpt Orange Co"             -7.87             -11
99 "Palm Springs Intl"                    -12.7              -13.5
```

b.

```
flights
```

```
# A tibble: 336,776 × 19
    year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
   <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
 1  2013     1     1      517            515         2      830            819
 2  2013     1     1      533            529         4      850            830
 3  2013     1     1      542            540         2      923            850
 4  2013     1     1      544            545        -1     1004           1022
 5  2013     1     1      554            600        -6      812            837
 6  2013     1     1      554            558        -4      740            728
 7  2013     1     1      555            600        -5      913            854
 8  2013     1     1      557            600        -3      709            723
 9  2013     1     1      557            600        -3      838            846
10  2013     1     1      558            600        -2      753            745
# ℹ 336,766 more rows
# ℹ 11 more variables: arr_delay <dbl>, carrier <chr>, flight <int>,
#   tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>,
#   hour <dbl>, minute <dbl>, time_hour <dttm>
```

```
planes
```

```
# A tibble: 3,322 × 9
   tailnum  year type              manufacturer model engines seats speed engine
   <chr>   <int> <chr>             <chr>        <chr>   <int> <int> <int> <chr>
 1 N10156   2004 Fixed wing multi… EMBRAER      EMB-…       2    55    NA Turbo…
 2 N102UW   1998 Fixed wing multi… AIRBUS INDU… A320…       2   182    NA Turbo…
 3 N103US   1999 Fixed wing multi… AIRBUS INDU… A320…       2   182    NA Turbo…
 4 N104UW   1999 Fixed wing multi… AIRBUS INDU… A320…       2   182    NA Turbo…
 5 N10575   2002 Fixed wing multi… EMBRAER      EMB-…       2    55    NA Turbo…
 6 N105UW   1999 Fixed wing multi… AIRBUS INDU… A320…       2   182    NA Turbo…
 7 N107US   1999 Fixed wing multi… AIRBUS INDU… A320…       2   182    NA Turbo…
 8 N108UW   1999 Fixed wing multi… AIRBUS INDU… A320…       2   182    NA Turbo…
 9 N109UW   1999 Fixed wing multi… AIRBUS INDU… A320…       2   182    NA Turbo…
10 N110UW   1999 Fixed wing multi… AIRBUS INDU… A320…       2   182    NA Turbo…
# ℹ 3,312 more rows
```

```
fastest_aircraft <- flights %>%
  mutate(speed_mph = distance/ (air_time/60)) %>%
  filter(!is.na(speed_mph)) %>%
  left_join(planes, by = "tailnum") %>%
  group_by(model) %>%
```

```
  summarise(
    avg_speed_mph = mean(speed_mph, na.rm = TRUE),
    n_flights = n()
  ) %>%
  arrange(desc(avg_speed_mph)) %>%
  slice(1)

fastest_aircraft
```

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

Question2

a.

```
data <- read.csv("C:/Users/ken/Desktop/stats 506/hw4/recs2020_public_v7.csv")
nnmaps <- read.csv("C:/Users/ken/Desktop/stats 506/hw4/chicago-nmmaps.csv")
```

```
#' Get average monthly temperature
#' @param month Numeric or string month
#' @param year Year
#' @param data Data set containing `month_numeric`, `year`, and `temp` columns
#' @param average_fn Function to compute average. Default is `mean`.
#' @param celsius Logical, default `FALSE` (return Fahrenheit instead)
#' @return Average temperature
get_temp <- function(month, year, data, average_fn = mean, celsius = FALSE) {
  if (is.numeric(month)) {
    if (month < 1 | month > 12) {
      warning("Invalid month provided. Returning NA.")
      return(NA)
    }
  } else if (is.character(month)) {
    months <- c("January", "February", "March", "April", "May", "June", "July",
                "August", "September", "October", "November", "December")
    match_month <- match.arg(month, months, several.ok = FALSE)
    if (is.na(match_month)) {
      warning("Invalid month name provided. Returning NA.")
      return(NA)
    }
    month <- which(months == match_month)
  } else {
    stop("Month must be numeric or character")
  }

  if (!is.numeric(year)) {
    stop("Year must be numeric")
  }
  if (year < 1997 | year > 2000) {
```

```
    warning("Year out of range. Returning NA.")
    return(NA)
  }

  if (!is.function(average_fn)) {
    stop("average_fn must be a function")
  }

  data %>%
    select(temp, month_numeric, year) %>%
    rename(year_col = year) %>% # Avoid conflict with `year` parameter
    filter(year_col == year,
           month_numeric == month) %>%
    summarize(avgtmp = average_fn(temp)) %>%
    mutate(avgtmp = ifelse(isTRUE(celsius), 5/9 * (avgtmp - 32), avgtmp)) %>%
    pull(avgtmp) -> out # Convert to numeric for result

  return(out)
}
```

```
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
```

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

```
Warning in get_temp(13, 1998, data = nnmaps): Invalid month provided. Returning
NA.

[1] NA
```

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

```
Warning in get_temp(2, 2005, data = nnmaps): Year out of range. Returning NA.

[1] NA
```

```
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
```

Question3

```
data2 <- read.csv("C:/Users/ken/Desktop/stats 506/hw4/df_for_ml_improved_new_market.csv")
head(data2)
```

```
  id case_id year height width size_inchsqr price_usd meanprice_year min_price
1  0   57649 1997     29    24          696      4160     246.772000 0.7753623
2  1   30468 1997     17    14          238      2340      13.870345 7.5610860
3  2   85464 1997     28    22          616      3640      26.460571 1.3283702
4  3   27308 1997     32    39         1248     10832      18.206057 1.4957265
5  4   82202 1997     46    37         1702     13210       5.255666 0.7423530
6  5   60932 1997     50    43         2150      3434      30.392677 0.9310700
    max_price medianprice_year  cnt_mean   cnt_max cnt_median  cot_mean
1 4680.250000        18.515385  4.854934  6.393678   4.854934  4.854934
2   23.111110        12.780289  9.038946 10.516807   9.038946 10.088465
3   86.531040        18.425526  6.321429  6.321429   6.321429 26.259993
4  123.551500        10.004489 17.954947 26.624638  17.954947 22.126005
5    8.303173         5.572877  4.100771  8.303173   3.066840  5.255666
6  683.472200         4.892097  3.827314  6.385563   3.626819 12.564496
     cot_max cot_median ranking fest_biennal private_inst public_inst solo_show
1   6.393678   4.854934      38            0            2           7         4
2  12.187500  10.516807     550            0            3           3         3
3  86.531040  18.425526     857            0            1           2         1
4 123.551500  11.151515      62            0            6           5         3
5   8.303173   5.572877      89            1            2          11         5
6  95.500000   4.628543     201            0            6           3         1
  group_show age estimate_min_usd estimate_max_usd estimate_center_usd
1          5  67         5200.000         6500.000            5850.000
2          3  41         1950.000         2600.000            2275.000
3          2  70         3900.000         5200.000            4550.000
4          8  60        13209.756        15851.707           14530.732
5          9  59        10568.000        13210.000           11889.000
6          7  62         2377.385         2641.538            2509.462
  log10_estimate_geo_mean_usd estimate_range_usd Country...Austria
1                    3.764458          1300.0000                 0
2                    3.352504           650.0000                 0
3                    3.653534          1300.0000                 0
4                    4.160485          2641.9512                 0
5                    4.072448          2642.0000                 0
6                    3.398978           264.1538                 0
  Country...China Country...Dutch Country...France Country...Germany
1               0               0                0                 0
2               0               0                0                 0
3               0               0                0                 0
4               0               0                0                 0
5               0               0                0                 0
6               0               0                0                 0
  Country...Japan Country...Others Country...Swiss Country...UK Country...USA
1               0                1               0            0             0
2               0                1               0            0             0
```

```
3               0               1              0           0             0
4               0               1              0           0             0
5               0               1              0           0             0
6               0               1              0           0             0
  Continent...East.Asia Continent...Europe Continent...North.America
1                     0                  1                         0
2                     0                  1                         0
3                     0                  1                         0
4                     0                  1                         0
5                     0                  1                         0
6                     0                  1                         0
  Continent...Others Tier...1 Tier...2 Tier...3 Tier...4 Living...Argentina
1                  0        0        0        0        1                  0
2                  0        0        0        0        1                  0
3                  0        0        0        0        1                  0
4                  0        0        0        0        1                  0
5                  0        0        0        0        1                  0
6                  0        0        0        0        1                  0
  Living...Australia Living...Austria Living...Brazil Living...China
1                  0                 0               0              0
2                  0                 0               0              0
3                  0                 0               0              0
4                  0                 0               0              0
5                  0                 0               0              0
6                  0                 0               0              0
  Living...Denmark Living...Dutch Living...France Living...Germany
1                0              0               0                0
2                0              0               0                0
3                0              0               1                0
4                0              0               0                0
5                1              0               0                0
6                0              0               0                0
  Living...India Living...Italia Living...Japan Living...Mexico Living...Norway
1              0               0              0               0               0
2              0               0              0               0               0
3              0               0              0               0               0
4              0               0              0               0               0
5              0               0              0               0               0
6              0               0              0               0               0
  Living...Others Living...South.Africa Living...Sweden Living...Swiss
1               0                     0              0             0
2               0                     0              0             0
3               0                     0              0             0
4               0                     0              0             0
5               0                     0              0             0
6               0                     0              0             0
  Living...Taiwan Living...UK Living...USA price_usd_prev_5_mean
1               0           0            1                    NA
2               0           0            1                    NA
3               0           0            0                    NA
4               0           0            1               32823.0
5               0           0            0               33234.0
6               0           0            1                7456.4
  price_usd_prev_5_median price_usd_prev_5_max price_usd_prev_10_mean
```

```
1                      NA                   NA                      NA
2                      NA                   NA                      NA
3                      NA                   NA                      NA
4                    8155               133336                25191.14
5                   33234                33234                33234.00
6                    5284                13008                 7456.40
  price_usd_prev_10_median price_usd_prev_10_max price_same_size_prev_5_max
1                       NA                    NA                         NA
2                       NA                    NA                         NA
3                       NA                    NA                         NA
4                     8155                133336                   77262.79
5                    33234                 33234                   12982.45
6                     5284                 13008                   13728.96
  price_same_size_prev_5_mean price_same_size_prev_5_median
1                          NA                            NA
2                          NA                            NA
3                          NA                            NA
4                   27815.086                     17532.290
5                   12982.451                     12982.451
6                    8252.103                      7335.083
  price_same_size_prev_10_max price_same_size_prev_10_mean
1                          NA                           NA
2                          NA                           NA
3                          NA                           NA
4                    77262.79                    23102.915
5                    12982.45                    12982.451
6                    13728.96                     8252.103
  price_same_size_prev_10_median matched_genre matched_country gender_NA
1                             NA             0               0         0
2                             NA             0               0         0
3                             NA             0               0         0
4                      14228.535             0               0         0
5                      12982.451             1               0         0
6                       7335.083             0               0         0
  gender_female gender_male edu_NA edu_abroad edu_both edu_domestic degree_NA
1             0           1      0          0        0            1         0
2             0           1      0          0        0            1         0
3             0           1      0          0        0            1         0
4             0           1      0          0        0            1         0
5             0           1      0          0        0            1         0
6             0           1      0          0        0            1         0
  grad no_degree undergrad elite_school elite_school_NA non_elite_school
1    0         0         1            1               0                0
2    1         0         0            0               0                1
3    0         0         1            0               0                1
4    1         0         0            1               0                0
5    0         0         1            0               0                1
6    0         0         1            0               0                1
  elite_award_not_received elite_award_received artwork_order   cnt_min
1                        0                    1            46 3.3161905
2                        1                    0             3 7.5610860
3                        1                    0             8 6.3214290
4                        1                    0            40 9.2852560
5                        1                    0             6 0.9323006
```

```
6                       1                    0            21 1.7088372
   cot_min Genre___Photography Genre___Print Genre___Sculpture Genre___Painting
1 3.316190                   0             0                 0                1
2 7.561086                   0             0                 1                0
3 1.328370                   0             0                 1                0
4 1.495726                   0             0                 0                1
5 0.742353                   1             0                 0                0
6 0.931070                   0             0                 0                1
  Genre___Others price_usd_prev_5_min price_usd_prev_10_min
1              1                 3800                  3800
2              0                 3250                  3250
3              0                  735                   735
4              1                 4000                  4000
5              0                 1840                  1840
6              1                 1692                  1692
  price_same_size_prev_5_min price_same_size_prev_10_min  eventdate train
1                  2265.2726                   2265.2726 24/04/1997     1
2                  2041.9710                   2041.9710 24/04/1997     1
3                   996.2469                    996.2469 24/04/1997     1
4                  3268.1713                   3268.1713 22/10/1997     1
5                  8566.5881                   8566.5881 22/10/1997     1
6                  2001.8004                   2001.8004 22/10/1997     1
```

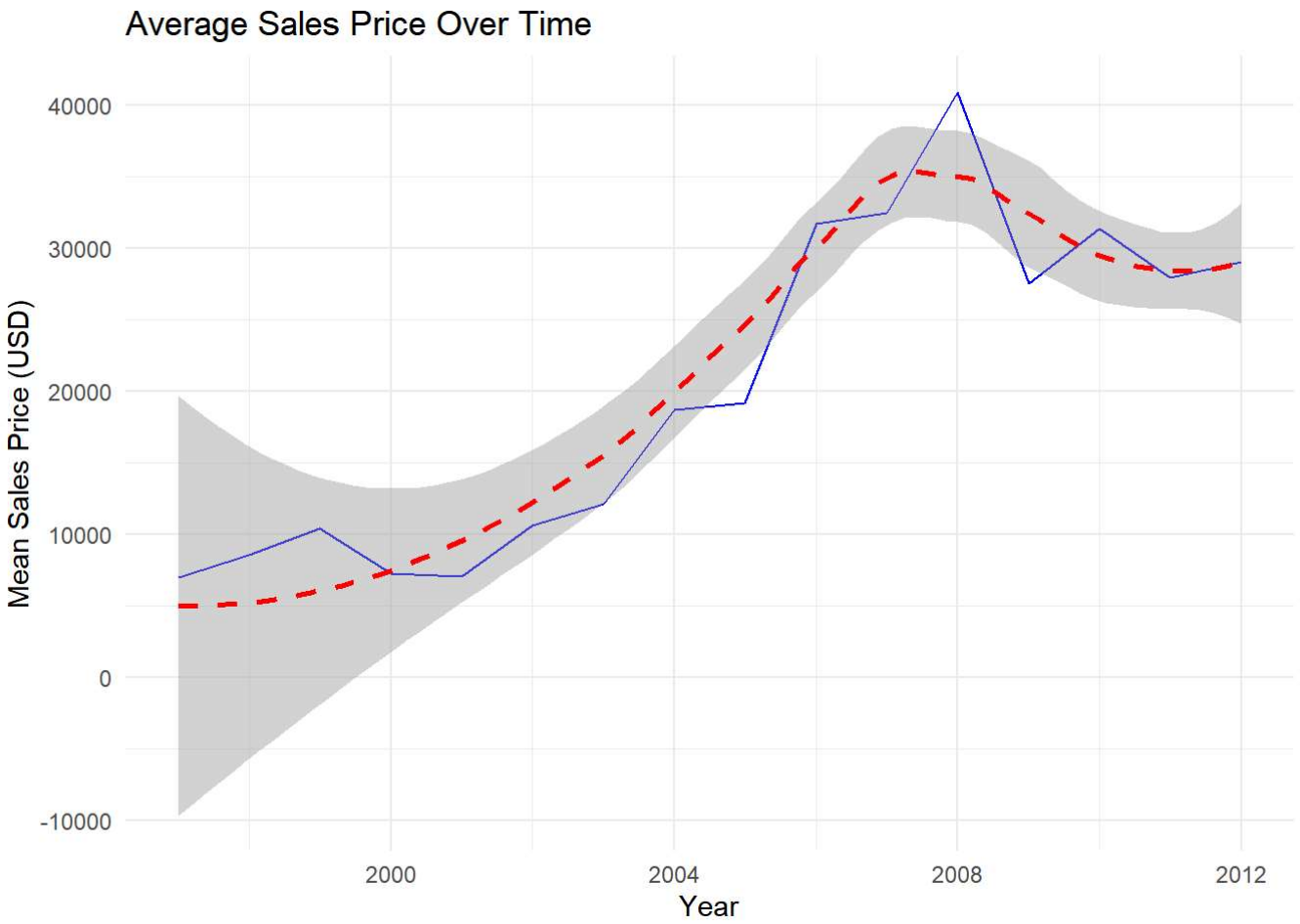
```
names(data2)
```

```
  [1] "id"                          "case_id"
  [3] "year"                        "height"
  [5] "width"                       "size_inchsqr"
  [7] "price_usd"                   "meanprice_year"
  [9] "min_price"                   "max_price"
 [11] "medianprice_year"            "cnt_mean"
 [13] "cnt_max"                     "cnt_median"
 [15] "cot_mean"                    "cot_max"
 [17] "cot_median"                  "ranking"
 [19] "fest_biennal"                "private_inst"
 [21] "public_inst"                 "solo_show"
 [23] "group_show"                  "age"
 [25] "estimate_min_usd"            "estimate_max_usd"
 [27] "estimate_center_usd"         "log10_estimate_geo_mean_usd"
 [29] "estimate_range_usd"          "Country...Austria"
 [31] "Country...China"             "Country...Dutch"
 [33] "Country...France"            "Country...Germany"
 [35] "Country...Japan"             "Country...Others"
 [37] "Country...Swiss"             "Country...UK"
 [39] "Country...USA"               "Continent...East.Asia"
 [41] "Continent...Europe"          "Continent...North.America"
 [43] "Continent...Others"          "Tier...1"
 [45] "Tier...2"                    "Tier...3"
 [47] "Tier...4"                    "Living...Argentina"
 [49] "Living...Australia"          "Living...Austria"
 [51] "Living...Brazil"             "Living...China"
 [53] "Living...Denmark"            "Living...Dutch"
 [55] "Living...France"             "Living...Germany"
```

```
 [57] "Living...India"                "Living...Italia"
 [59] "Living...Japan"                "Living...Mexico"
 [61] "Living...Norway"               "Living...Others"
 [63] "Living...South.Africa"         "Living...Sweden"
 [65] "Living...Swiss"                "Living...Taiwan"
 [67] "Living...UK"                   "Living...USA"
 [69] "price_usd_prev_5_mean"         "price_usd_prev_5_median"
 [71] "price_usd_prev_5_max"          "price_usd_prev_10_mean"
 [73] "price_usd_prev_10_median"      "price_usd_prev_10_max"
 [75] "price_same_size_prev_5_max"    "price_same_size_prev_5_mean"
 [77] "price_same_size_prev_5_median" "price_same_size_prev_10_max"
 [79] "price_same_size_prev_10_mean"  "price_same_size_prev_10_median"
 [81] "matched_genre"                 "matched_country"
 [83] "gender_NA"                     "gender_female"
 [85] "gender_male"                   "edu_NA"
 [87] "edu_abroad"                    "edu_both"
 [89] "edu_domestic"                  "degree_NA"
 [91] "grad"                          "no_degree"
 [93] "undergrad"                     "elite_school"
 [95] "elite_school_NA"               "non_elite_school"
 [97] "elite_award_not_received"      "elite_award_received"
 [99] "artwork_order"                 "cnt_min"
[101] "cot_min"                       "Genre___Photography"
[103] "Genre___Print"                 "Genre___Sculpture"
[105] "Genre___Painting"              "Genre___Others"
[107] "price_usd_prev_5_min"          "price_usd_prev_10_min"
[109] "price_same_size_prev_5_min"    "price_same_size_prev_10_min"
[111] "eventdate"                     "train"
```

```r
ggplot(data2, aes(x = year, y = price_usd)) +
  geom_line(stat = "summary", fun = mean, color = "blue") +
  geom_smooth(method = "loess", se = TRUE, color = "red", linetype = "dashed") +
  labs(
    title = "Average Sales Price Over Time",
    x = "Year",
    y = "Mean Sales Price (USD)"
  ) +
  theme_minimal()
```

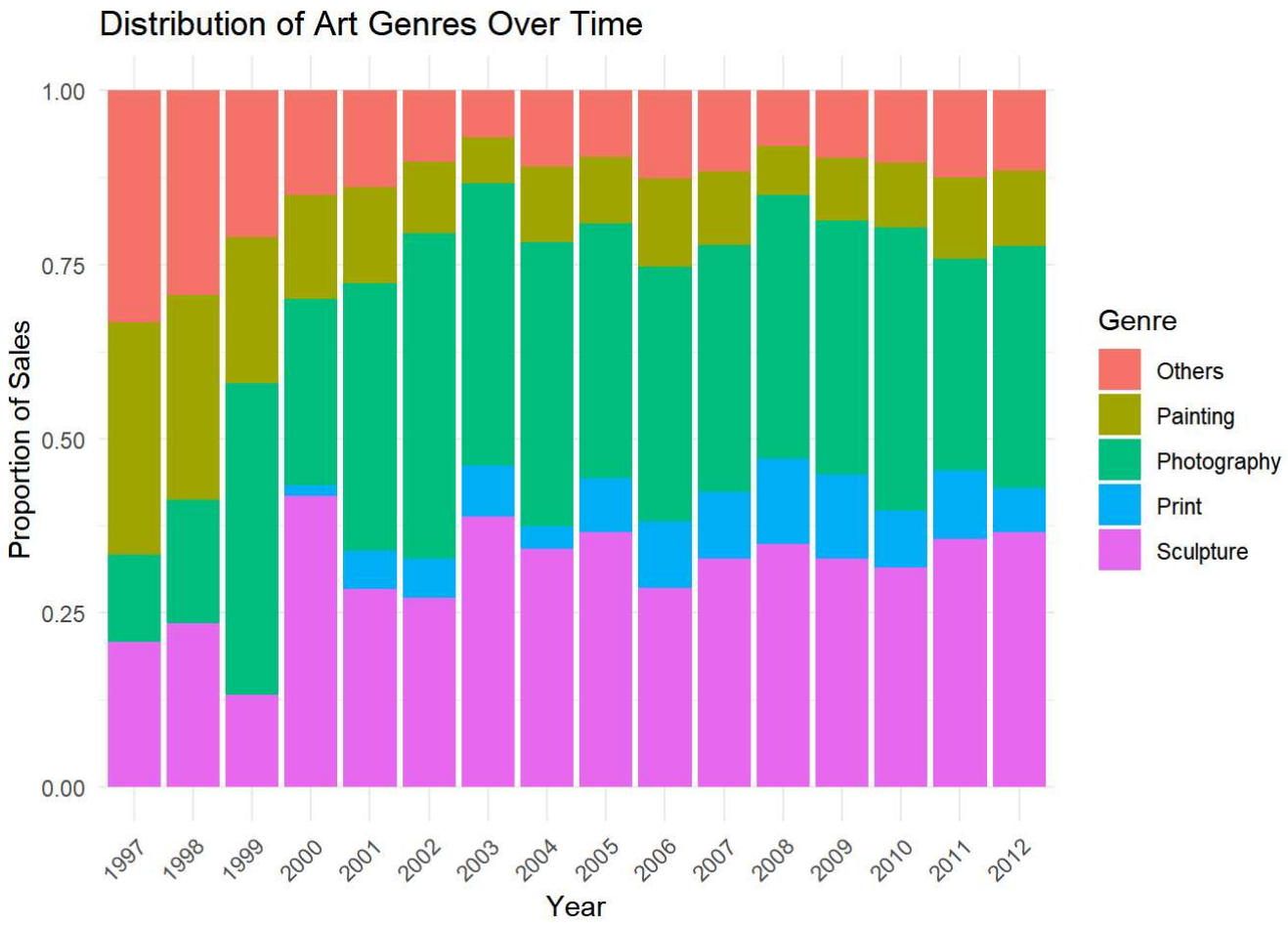
```
`geom_smooth()` using formula = 'y ~ x'
```




```r
# Create a melted dataset for genres
genre_data <- data2 %>%
  select(year, matches("Genre___")) %>%
  pivot_longer(cols = starts_with("Genre___"), names_to = "genre", values_to = "count") %>%
  filter(count == 1) %>%
  mutate(genre = gsub("Genre___", "", genre))

# Plot the distribution of genres over years
genre_plot <- genre_data %>%
  ggplot(aes(x = factor(year), fill = genre)) +
  geom_bar(position = "fill") +
  labs(title = "Distribution of Art Genres Over Time",
       x = "Year",
       y = "Proportion of Sales",
       fill = "Genre") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

# Print the plot
print(genre_plot)
```

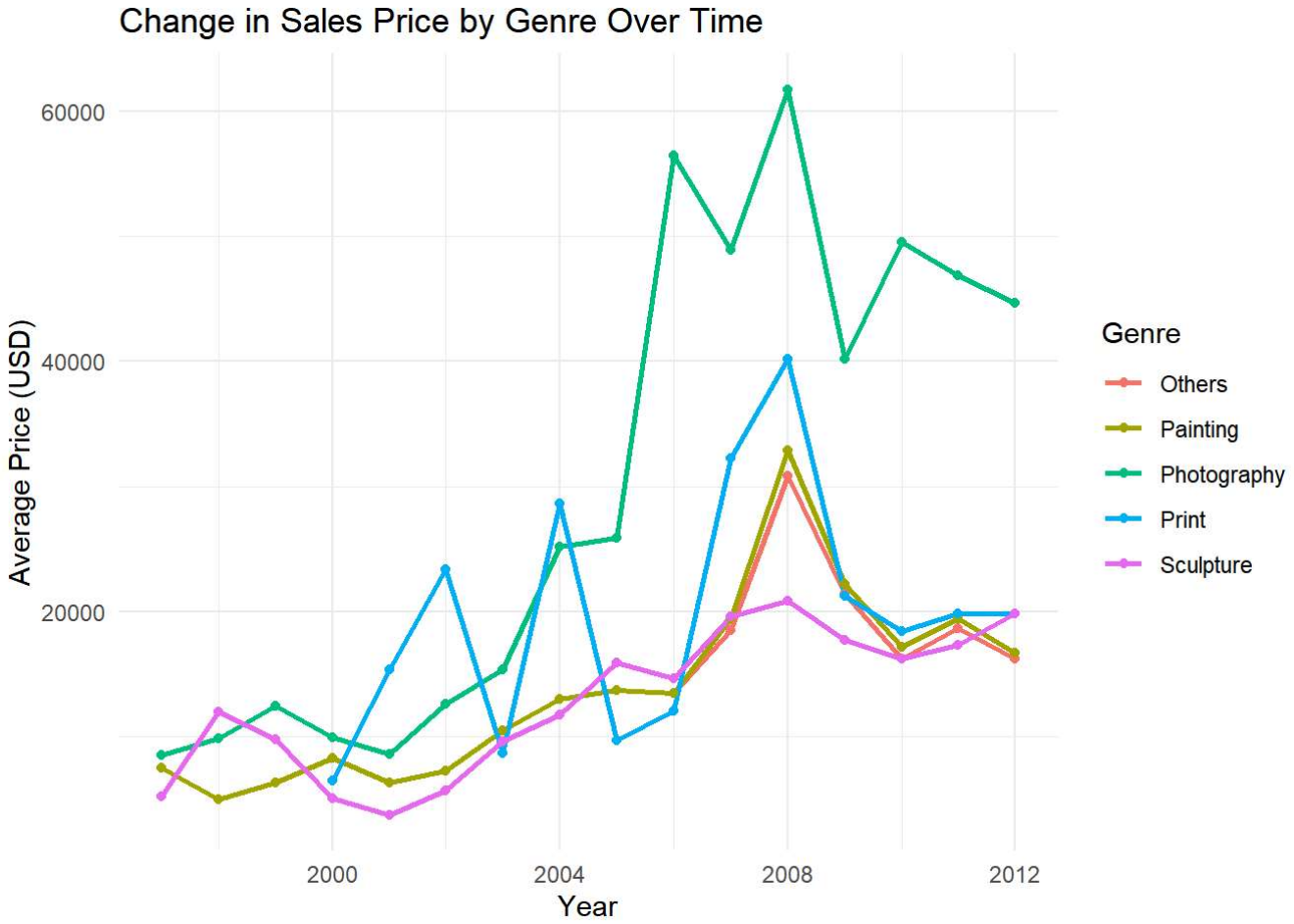



```
# Create a melted dataset for genres with prices
genre_price_data <- data2 %>%
  select(year, price_usd, matches("Genre___")) %>%
  pivot_longer(cols = starts_with("Genre___"), names_to = "genre", values_to = "count") %>%
  filter(count == 1) %>%
  mutate(genre = gsub("Genre___", "", genre)) %>%
  group_by(year, genre) %>%
  summarise(mean_price_usd = mean(price_usd, na.rm = TRUE), .groups = "drop")

# Plot showing the change in average sales price by genre over time
genre_price_plot <- genre_price_data %>%
  ggplot(aes(x = year, y = mean_price_usd, color = genre)) +
  geom_line(size = 1) +
  geom_point() +
  labs(title = "Change in Sales Price by Genre Over Time",
       x = "Year",
       y = "Average Price (USD)",
       color = "Genre") +
  theme_minimal()
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead.
```

```
# Print the plot
print(genre_price_plot)
```


Change in Sales Price by Genre Over Time
Average Price (USD)
60000
40000
20000
2000
2004
2008
2012
Year
Genre
Others
Painting
Photography
Print
Sculpture