SC-MD_Nov-14_1

Srijan Kundu

2022-11-14

Using the package stargazer:

```
library(stargazer)
```

```
##
## Please cite as:

##  Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
beaver1
```

```
##     day time  temp activ
## 1   346  840 36.33     0
## 2   346  850 36.34     0
## 3   346  900 36.35     0
## 4   346  910 36.42     0
## 5   346  920 36.55     0
## 6   346  930 36.69     0
## 7   346  940 36.71     0
## 8   346  950 36.75     0
## 9   346 1000 36.81     0
## 10  346 1010 36.88     0
## 11  346 1020 36.89     0
```

```
## 12  346 1030 36.91     0
## 13  346 1040 36.85     0
## 14  346 1050 36.89     0
## 15  346 1100 36.89     0
## 16  346 1110 36.67     0
## 17  346 1120 36.50     0
## 18  346 1130 36.74     0
## 19  346 1140 36.77     0
## 20  346 1150 36.76     0
## 21  346 1200 36.78     0
## 22  346 1210 36.82     0
## 23  346 1220 36.89     0
## 24  346 1230 36.99     0
## 25  346 1240 36.92     0
## 26  346 1250 36.99     0
## 27  346 1300 36.89     0
## 28  346 1310 36.94     0
## 29  346 1320 36.92     0
## 30  346 1330 36.97     0
## 31  346 1340 36.91     0
## 32  346 1350 36.79     0
## 33  346 1400 36.77     0
## 34  346 1410 36.69     0
## 35  346 1420 36.62     0
## 36  346 1430 36.54     0
## 37  346 1440 36.55     0
## 38  346 1450 36.67     0
## 39  346 1500 36.69     0
## 40  346 1510 36.62     0
## 41  346 1520 36.64     0
## 42  346 1530 36.59     0
## 43  346 1540 36.65     0
## 44  346 1550 36.75     0
## 45  346 1600 36.80     0
## 46  346 1610 36.81     0
## 47  346 1620 36.87     0
## 48  346 1630 36.87     0
## 49  346 1640 36.89     0
## 50  346 1650 36.94     0
## 51  346 1700 36.98     0
## 52  346 1710 36.95     0
## 53  346 1720 37.00     0
## 54  346 1730 37.07     1
## 55  346 1740 37.05     0
## 56  346 1750 37.00     0
## 57  346 1800 36.95     0
## 58  346 1810 37.00     0
## 59  346 1820 36.94     0
## 60  346 1830 36.88     0
## 61  346 1840 36.93     0
## 62  346 1850 36.98     0
## 63  346 1900 36.97     0
## 64  346 1910 36.85     0
## 65  346 1920 36.92     0
```

```
## 66  346 1930 36.99     0
## 67  346 1940 37.01     0
## 68  346 1950 37.10     1
## 69  346 2000 37.09     0
## 70  346 2010 37.02     0
## 71  346 2020 36.96     0
## 72  346 2030 36.84     0
## 73  346 2040 36.87     0
## 74  346 2050 36.85     0
## 75  346 2100 36.85     0
## 76  346 2110 36.87     0
## 77  346 2120 36.89     0
## 78  346 2130 36.86     0
## 79  346 2140 36.91     0
## 80  346 2150 37.53     1
## 81  346 2200 37.23     0
## 82  346 2210 37.20     0
## 83  346 2230 37.25     1
## 84  346 2240 37.20     0
## 85  346 2250 37.21     0
## 86  346 2300 37.24     1
## 87  346 2310 37.10     0
## 88  346 2320 37.20     0
## 89  346 2330 37.18     0
## 90  346 2340 36.93     0
## 91  346 2350 36.83     0
## 92  347    0 36.93     0
## 93  347   10 36.83     0
## 94  347   20 36.80     0
## 95  347   30 36.75     0
## 96  347   40 36.71     0
## 97  347   50 36.73     0
## 98  347  100 36.75     0
## 99  347  110 36.72     0
## 100 347  120 36.76     0
## 101 347  130 36.70     0
## 102 347  140 36.82     0
## 103 347  150 36.88     0
## 104 347  200 36.94     0
## 105 347  210 36.79     0
## 106 347  220 36.78     0
## 107 347  230 36.80     0
## 108 347  240 36.82     0
## 109 347  250 36.84     0
## 110 347  300 36.86     0
## 111 347  310 36.88     0
## 112 347  320 36.93     0
## 113 347  330 36.97     0
## 114 347  340 37.15     1
```

```
stargazer(beaver1, type = "text", out = "table1.txt", title = "Summary output", digits = 1, covariate.la
```

```
##
## Summary output
```

```
## ==================================
## Statistic  x1     x2     x3  activ
## ----------------------------------
## N          114    114   114   114
## Mean      346.2 1,312.0 36.9  0.1
## St. Dev.   0.4   701.9  0.2   0.2
## Min        346     0    36.3   0
## Max        347   2,350  37.5   1
## ----------------------------------
```

- First argument of `stargazer` has to be a dataframe.

```
stargazer(beaver1, type = "html", out = "table1.html", title = "Summary output", digits = 1, covariate.l
```

```
##
## <table style="text-align:center"><caption><strong>Summary output</strong></caption>
## <tr><td colspan="5" style="border-bottom: 1px solid black"></td></tr><tr><td style="text-align:left"
## <tr><td colspan="5" style="border-bottom: 1px solid black"></td></tr><tr><td style="text-align:left"
## <tr><td style="text-align:left">Mean</td><td>346.2</td><td>1,312.0</td><td>36.9</td><td>0.1</td></tr
## <tr><td style="text-align:left">St. Dev.</td><td>0.4</td><td>701.9</td><td>0.2</td><td>0.2</td></tr>
## <tr><td style="text-align:left">Min</td><td>346</td><td>0</td><td>36.3</td><td>0</td></tr>
## <tr><td style="text-align:left">Max</td><td>347</td><td>2,350</td><td>37.5</td><td>1</td></tr>
## <tr><td colspan="5" style="border-bottom: 1px solid black"></td></tr></table>
```

```
#default type is latex
stargazer(beaver1, out = "table1.tex", title = "Summary output", digits = 1, covariate.labels = c("x1",
```

```
##
## % Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac a
## % Date and time: Mon, Nov 14, 2022 - 15:56:05
## \begin{table}[!htbp] \centering
##   \caption{Summary output}
##   \label{}
## \begin{tabular}{@{\extracolsep{5pt}}lcccc}
## \\[-1.8ex]\hline
## \hline \\[-1.8ex]
## Statistic & x1 & x2 & x3 & activ \\
## \hline \\[-1.8ex]
## N & 114 & 114 & 114 & 114 \\
## Mean & 346.2 & 1,312.0 & 36.9 & 0.1 \\
## St. Dev. & 0.4 & 701.9 & 0.2 & 0.2 \\
## Min & 346 & 0 & 36.3 & 0 \\
## Max & 347 & 2,350 & 37.5 & 1 \\
## \hline \\[-1.8ex]
## \end{tabular}
## \end{table}
```

Question: Prepare a table taking a subset data from `beaver1` such that `temperature` is greater than 36.70**.**

```r
data = beaver1[beaver1$temp>36.7, ]
data
```

```
##     day time  temp activ
## 7   346  940 36.71     0
## 8   346  950 36.75     0
## 9   346 1000 36.81     0
## 10  346 1010 36.88     0
## 11  346 1020 36.89     0
## 12  346 1030 36.91     0
## 13  346 1040 36.85     0
## 14  346 1050 36.89     0
## 15  346 1100 36.89     0
## 18  346 1130 36.74     0
## 19  346 1140 36.77     0
## 20  346 1150 36.76     0
## 21  346 1200 36.78     0
## 22  346 1210 36.82     0
## 23  346 1220 36.89     0
## 24  346 1230 36.99     0
## 25  346 1240 36.92     0
## 26  346 1250 36.99     0
## 27  346 1300 36.89     0
## 28  346 1310 36.94     0
## 29  346 1320 36.92     0
## 30  346 1330 36.97     0
## 31  346 1340 36.91     0
## 32  346 1350 36.79     0
## 33  346 1400 36.77     0
## 44  346 1550 36.75     0
## 45  346 1600 36.80     0
## 46  346 1610 36.81     0
## 47  346 1620 36.87     0
## 48  346 1630 36.87     0
## 49  346 1640 36.89     0
## 50  346 1650 36.94     0
## 51  346 1700 36.98     0
## 52  346 1710 36.95     0
## 53  346 1720 37.00     0
## 54  346 1730 37.07     1
## 55  346 1740 37.05     0
## 56  346 1750 37.00     0
## 57  346 1800 36.95     0
## 58  346 1810 37.00     0
## 59  346 1820 36.94     0
## 60  346 1830 36.88     0
## 61  346 1840 36.93     0
## 62  346 1850 36.98     0
## 63  346 1900 36.97     0
## 64  346 1910 36.85     0
## 65  346 1920 36.92     0
## 66  346 1930 36.99     0
## 67  346 1940 37.01     0
```

```
## 68  346 1950 37.10     1
## 69  346 2000 37.09     0
## 70  346 2010 37.02     0
## 71  346 2020 36.96     0
## 72  346 2030 36.84     0
## 73  346 2040 36.87     0
## 74  346 2050 36.85     0
## 75  346 2100 36.85     0
## 76  346 2110 36.87     0
## 77  346 2120 36.89     0
## 78  346 2130 36.86     0
## 79  346 2140 36.91     0
## 80  346 2150 37.53     1
## 81  346 2200 37.23     0
## 82  346 2210 37.20     0
## 83  346 2230 37.25     1
## 84  346 2240 37.20     0
## 85  346 2250 37.21     0
## 86  346 2300 37.24     1
## 87  346 2310 37.10     0
## 88  346 2320 37.20     0
## 89  346 2330 37.18     0
## 90  346 2340 36.93     0
## 91  346 2350 36.83     0
## 92  347    0 36.93     0
## 93  347   10 36.83     0
## 94  347   20 36.80     0
## 95  347   30 36.75     0
## 96  347   40 36.71     0
## 97  347   50 36.73     0
## 98  347  100 36.75     0
## 99  347  110 36.72     0
## 100 347  120 36.76     0
## 102 347  140 36.82     0
## 103 347  150 36.88     0
## 104 347  200 36.94     0
## 105 347  210 36.79     0
## 106 347  220 36.78     0
## 107 347  230 36.80     0
## 108 347  240 36.82     0
## 109 347  250 36.84     0
## 110 347  300 36.86     0
## 111 347  310 36.88     0
## 112 347  320 36.93     0
## 113 347  330 36.97     0
## 114 347  340 37.15     1
```

```
beaver1
```

```
##     day time  temp activ
## 1   346  840 36.33     0
## 2   346  850 36.34     0
## 3   346  900 36.35     0
## 4   346  910 36.42     0
```

```
## 5   346  920 36.55     0
## 6   346  930 36.69     0
## 7   346  940 36.71     0
## 8   346  950 36.75     0
## 9   346 1000 36.81     0
## 10  346 1010 36.88     0
## 11  346 1020 36.89     0
## 12  346 1030 36.91     0
## 13  346 1040 36.85     0
## 14  346 1050 36.89     0
## 15  346 1100 36.89     0
## 16  346 1110 36.67     0
## 17  346 1120 36.50     0
## 18  346 1130 36.74     0
## 19  346 1140 36.77     0
## 20  346 1150 36.76     0
## 21  346 1200 36.78     0
## 22  346 1210 36.82     0
## 23  346 1220 36.89     0
## 24  346 1230 36.99     0
## 25  346 1240 36.92     0
## 26  346 1250 36.99     0
## 27  346 1300 36.89     0
## 28  346 1310 36.94     0
## 29  346 1320 36.92     0
## 30  346 1330 36.97     0
## 31  346 1340 36.91     0
## 32  346 1350 36.79     0
## 33  346 1400 36.77     0
## 34  346 1410 36.69     0
## 35  346 1420 36.62     0
## 36  346 1430 36.54     0
## 37  346 1440 36.55     0
## 38  346 1450 36.67     0
## 39  346 1500 36.69     0
## 40  346 1510 36.62     0
## 41  346 1520 36.64     0
## 42  346 1530 36.59     0
## 43  346 1540 36.65     0
## 44  346 1550 36.75     0
## 45  346 1600 36.80     0
## 46  346 1610 36.81     0
## 47  346 1620 36.87     0
## 48  346 1630 36.87     0
## 49  346 1640 36.89     0
## 50  346 1650 36.94     0
## 51  346 1700 36.98     0
## 52  346 1710 36.95     0
## 53  346 1720 37.00     0
## 54  346 1730 37.07     1
## 55  346 1740 37.05     0
## 56  346 1750 37.00     0
## 57  346 1800 36.95     0
## 58  346 1810 37.00     0
```

```
## 59  346 1820 36.94     0
## 60  346 1830 36.88     0
## 61  346 1840 36.93     0
## 62  346 1850 36.98     0
## 63  346 1900 36.97     0
## 64  346 1910 36.85     0
## 65  346 1920 36.92     0
## 66  346 1930 36.99     0
## 67  346 1940 37.01     0
## 68  346 1950 37.10     1
## 69  346 2000 37.09     0
## 70  346 2010 37.02     0
## 71  346 2020 36.96     0
## 72  346 2030 36.84     0
## 73  346 2040 36.87     0
## 74  346 2050 36.85     0
## 75  346 2100 36.85     0
## 76  346 2110 36.87     0
## 77  346 2120 36.89     0
## 78  346 2130 36.86     0
## 79  346 2140 36.91     0
## 80  346 2150 37.53     1
## 81  346 2200 37.23     0
## 82  346 2210 37.20     0
## 83  346 2230 37.25     1
## 84  346 2240 37.20     0
## 85  346 2250 37.21     0
## 86  346 2300 37.24     1
## 87  346 2310 37.10     0
## 88  346 2320 37.20     0
## 89  346 2330 37.18     0
## 90  346 2340 36.93     0
## 91  346 2350 36.83     0
## 92  347    0 36.93     0
## 93  347   10 36.83     0
## 94  347   20 36.80     0
## 95  347   30 36.75     0
## 96  347   40 36.71     0
## 97  347   50 36.73     0
## 98  347  100 36.75     0
## 99  347  110 36.72     0
## 100 347  120 36.76     0
## 101 347  130 36.70     0
## 102 347  140 36.82     0
## 103 347  150 36.88     0
## 104 347  200 36.94     0
## 105 347  210 36.79     0
## 106 347  220 36.78     0
## 107 347  230 36.80     0
## 108 347  240 36.82     0
## 109 347  250 36.84     0
## 110 347  300 36.86     0
## 111 347  310 36.88     0
## 112 347  320 36.93     0
```

```
## 113 347  330 36.97     0
## 114 347  340 37.15     1
```

```
stargazer(data, type = "text", out = "table2.txt", title = "Summary output", flip = TRUE, notes = "data
```

```
##
## Summary output
## ========================================
## Statistic   day     time     temp  activ
## ----------------------------------------
## N           95       95       95    95
## Mean      346.232 1,338.000 36.921 0.063
## St. Dev.   0.424   749.249  0.145  0.245
## Min         346       0     36.710   0
## Max         347     2,350   37.530   1
## ----------------------------------------
## data taken from R
```

Question: For each destination, calculate average arrival delay and average distance and present it in a table. Remove Honolulu airport, and retain those destinations for which number of flights arriving is greater than 20.

```
library(nycflights13)
```

```
## Warning: package 'nycflights13' was built under R version 4.2.2
```

```
flights
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
##    <int> <int> <int>    <int>      <int>   <dbl>   <int>   <int>   <dbl> <chr>
##  1  2013     1     1      517        515       2     830     819      11 UA
##  2  2013     1     1      533        529       4     850     830      20 UA
##  3  2013     1     1      542        540       2     923     850      33 AA
##  4  2013     1     1      544        545      -1    1004    1022     -18 B6
##  5  2013     1     1      554        600      -6     812     837     -25 DL
##  6  2013     1     1      554        558      -4     740     728      12 UA
##  7  2013     1     1      555        600      -5     913     854      19 B6
##  8  2013     1     1      557        600      -3     709     723     -14 EV
##  9  2013     1     1      557        600      -3     838     846      -8 B6
## 10  2013     1     1      558        600      -2     753     745       8 AA
## # ... with 336,766 more rows, 9 more variables: flight <int>, tailnum <chr>,
## #   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dttm>, and abbreviated variable names
## #   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
## #   5: arr_delay
```

```
data = flights %>%
  group_by(dest) %>%
  select(dest, arr_delay, distance) %>%
```

```
  summarise(count = n(), avg_arr_delay = mean(arr_delay, na.rm = TRUE), avg_distance = mean(distance, na
  filter(dest != 'HNL' & count> 20) %>%
  select(dest, avg_arr_delay, avg_distance) %>%
  format.data.frame(digits = 3)
data
```

```
##    dest avg_arr_delay avg_distance
## 1   ABQ         4.382       1826.0
## 2   ACK         4.852        199.0
## 3   ALB        14.397        143.0
## 4   ATL        11.300        757.1
## 5   AUS         6.020       1514.3
## 6   AVL         8.004        583.6
## 7   BDL         7.049        116.0
## 8   BGR         8.028        378.0
## 9   BHM        16.877        866.0
## 10  BNA        11.812        758.2
## 11  BOS         2.914        190.6
## 12  BQN         8.245       1579.0
## 13  BTV         8.951        265.1
## 14  BUF         8.946        296.8
## 15  BUR         8.176       2465.0
## 16  BWI        10.727        179.4
## 17  BZN         7.600       1882.0
## 18  CAE        41.764        603.6
## 19  CAK        19.698        397.0
## 20  CHO         9.500        305.0
## 21  CHS        10.593        632.9
## 22  CLE         9.182        414.2
## 23  CLT         7.360        538.0
## 24  CMH        10.601        476.6
## 25  CRW        14.672        444.0
## 26  CVG        15.365        575.2
## 27  DAY        12.680        537.1
## 28  DCA         9.067        211.0
## 29  DEN         8.607       1614.7
## 30  DFW         0.322       1383.0
## 31  DSM        19.006       1020.9
## 32  DTW         5.430        498.1
## 33  EGE         6.304       1735.7
## 34  FLL         8.082       1070.1
## 35  GRR        18.190        605.8
## 36  GSO        14.113        449.8
## 37  GSP        15.935        596.0
## 38  HOU         7.176       1420.2
## 39  IAD        13.864        224.8
## 40  IAH         4.241       1407.2
## 41  ILM         4.636        500.0
## 42  IND         9.940        652.3
## 43  JAC        28.095       1875.6
## 44  JAX        11.845        824.7
## 45  LAS         0.258       2241.0
## 46  LAX         0.547       2468.6
```

```
## 47  LGB        -0.062       2465.0
## 48  MCI        14.514       1097.7
## 49  MCO         5.455        943.1
## 50  MDW        12.364        718.0
## 51  MEM        10.645        954.2
## 52  MHT        14.788        207.0
## 53  MIA         0.299       1091.6
## 54  MKE        14.167        733.4
## 55  MSN        20.196        804.0
## 56  MSP         7.270       1017.4
## 57  MSY         6.490       1177.7
## 58  MVY        -0.286        173.0
## 59  MYR         4.603        550.7
## 60  OAK         3.078       2576.0
## 61  OKC        30.619       1325.0
## 62  OMA        14.699       1135.6
## 63  ORD         5.877        729.0
## 64  ORF        10.949        288.5
## 65  PBI         8.563       1028.8
## 66  PDX         5.142       2445.6
## 67  PHL        10.127         94.3
## 68  PHX         2.097       2141.3
## 69  PIT         7.681        334.1
## 70  PSE         7.872       1617.0
## 71  PVD        16.235        160.0
## 72  PWM        11.660        276.1
## 73  RDU        10.052        426.8
## 74  RIC        20.111        281.4
## 75  ROC        11.561        259.3
## 76  RSW         3.238       1072.9
## 77  SAN         3.139       2437.3
## 78  SAT         6.945       1578.3
## 79  SAV        15.130        709.2
## 80  SDF        12.669        646.0
## 81  SEA        -1.099       2412.7
## 82  SFO         2.673       2577.9
## 83  SJC         3.448       2569.0
## 84  SJU         2.521       1599.8
## 85  SLC         0.176       1987.0
## 86  SMF        12.110       2521.0
## 87  SNA        -7.868       2434.0
## 88  SRQ         3.082       1044.7
## 89  STL        11.078        878.7
## 90  STT        -3.836       1627.0
## 91  SYR         8.904        205.9
## 92  TPA         7.409       1003.9
## 93  TUL        33.660       1215.0
## 94  TVC        12.968        652.4
## 95  TYS        24.069        638.8
## 96  XNA         7.466       1142.5
```

```
stargazer(data,
          summary = FALSE,
          type = "text",
```

```
          out = "table3.txt",
          covariate.labels = c("", "Destination", "Average-arrival-delay", "Average-distance"),
          digits = 3,
          title = "Average arrival delay and average distance and present it in a table, removed Honolul
          notes = "(data taken from nycflights13 package)")
```

```
##
## Average arrival delay and average distance and present it in a table, removed Honolulu airport, and
## ====================================================
##    Destination Average-arrival-delay Average-distance
## ----------------------------------------------------
## 1      ABQ             4.382              1826.0
## 2      ACK             4.852              199.0
## 3      ALB            14.397              143.0
## 4      ATL            11.300              757.1
## 5      AUS             6.020              1514.3
## 6      AVL             8.004              583.6
## 7      BDL             7.049              116.0
## 8      BGR             8.028              378.0
## 9      BHM            16.877              866.0
## 10     BNA            11.812              758.2
## 11     BOS             2.914              190.6
## 12     BQN             8.245              1579.0
## 13     BTV             8.951              265.1
## 14     BUF             8.946              296.8
## 15     BUR             8.176              2465.0
## 16     BWI            10.727              179.4
## 17     BZN             7.600              1882.0
## 18     CAE            41.764              603.6
## 19     CAK            19.698              397.0
## 20     CHO             9.500              305.0
## 21     CHS            10.593              632.9
## 22     CLE             9.182              414.2
## 23     CLT             7.360              538.0
## 24     CMH            10.601              476.6
## 25     CRW            14.672              444.0
## 26     CVG            15.365              575.2
## 27     DAY            12.680              537.1
## 28     DCA             9.067              211.0
## 29     DEN             8.607              1614.7
## 30     DFW             0.322              1383.0
## 31     DSM            19.006              1020.9
## 32     DTW             5.430              498.1
## 33     EGE             6.304              1735.7
## 34     FLL             8.082              1070.1
## 35     GRR            18.190              605.8
## 36     GSO            14.113              449.8
## 37     GSP            15.935              596.0
## 38     HOU             7.176              1420.2
## 39     IAD            13.864              224.8
## 40     IAH             4.241              1407.2
## 41     ILM             4.636              500.0
## 42     IND             9.940              652.3
```

```
## 43     JAC           28.095            1875.6
## 44     JAX           11.845             824.7
## 45     LAS            0.258            2241.0
## 46     LAX            0.547            2468.6
## 47     LGB           -0.062            2465.0
## 48     MCI           14.514            1097.7
## 49     MCO            5.455             943.1
## 50     MDW           12.364             718.0
## 51     MEM           10.645             954.2
## 52     MHT           14.788             207.0
## 53     MIA            0.299            1091.6
## 54     MKE           14.167             733.4
## 55     MSN           20.196             804.0
## 56     MSP            7.270            1017.4
## 57     MSY            6.490            1177.7
## 58     MVY           -0.286             173.0
## 59     MYR            4.603             550.7
## 60     OAK            3.078            2576.0
## 61     OKC           30.619            1325.0
## 62     OMA           14.699            1135.6
## 63     ORD            5.877             729.0
## 64     ORF           10.949             288.5
## 65     PBI            8.563            1028.8
## 66     PDX            5.142            2445.6
## 67     PHL           10.127              94.3
## 68     PHX            2.097            2141.3
## 69     PIT            7.681             334.1
## 70     PSE            7.872            1617.0
## 71     PVD           16.235             160.0
## 72     PWM           11.660             276.1
## 73     RDU           10.052             426.8
## 74     RIC           20.111             281.4
## 75     ROC           11.561             259.3
## 76     RSW            3.238            1072.9
## 77     SAN            3.139            2437.3
## 78     SAT            6.945            1578.3
## 79     SAV           15.130             709.2
## 80     SDF           12.669             646.0
## 81     SEA           -1.099            2412.7
## 82     SFO            2.673            2577.9
## 83     SJC            3.448            2569.0
## 84     SJU            2.521            1599.8
## 85     SLC            0.176            1987.0
## 86     SMF           12.110            2521.0
## 87     SNA           -7.868            2434.0
## 88     SRQ            3.082            1044.7
## 89     STL           11.078             878.7
## 90     STT           -3.836            1627.0
## 91     SYR            8.904             205.9
## 92     TPA            7.409            1003.9
## 93     TUL           33.660            1215.0
## 94     TVC           12.968             652.4
## 95     TYS           24.069             638.8
## 96     XNA            7.466            1142.5
```

```
## ---------------------------------------------------
## (data taken from nycflights13 package)
```

Question: Fit a linear regression on average arrival delay on average distance. Also fit a second degree regression on average arrival delay on distance. Present the output in a tabular form.

```
y = as.numeric(data$avg_arr_delay)
x = as.numeric(data$avg_distance)
linmod1 = lm(y~x)
linmod2 = lm(y~x+I(x^2))
linmod1
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Coefficients:
## (Intercept)            x
##   13.536849    -0.003781
```

```
linmod2
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2))
##
## Coefficients:
## (Intercept)            x       I(x^2)
##   1.053e+01    3.473e-03   -2.778e-06
```

```
stargazer(linmod1, linmod2,
          type = "html",
          out = "table4.html",
          dep.var.labels = "Arrival delay",
          covariate.labels = c("distance", "distance**2", "constant-term"),
          title = "Regression on average delay in arrival vs average distance",
          notes = "(data taken from nycfights pakage)")
```

```
##
## <table style="text-align:center"><caption><strong>Regression on average delay in arrival vs average d
## <tr><td colspan="3" style="border-bottom: 1px solid black"></td></tr><tr><td style="text-align:left">
## <tr><td></td><td colspan="2" style="border-bottom: 1px solid black"></td></tr>
## <tr><td style="text-align:left"></td><td colspan="2">Arrival delay</td></tr>
## <tr><td style="text-align:left"></td><td>(1)</td><td>(2)</td></tr>
## <tr><td colspan="3" style="border-bottom: 1px solid black"></td></tr><tr><td style="text-align:left">
## <tr><td style="text-align:left"></td><td>(0.001)</td><td>(0.004)</td></tr>
## <tr><td style="text-align:left"></td><td></td><td></td></tr>
## <tr><td style="text-align:left">distance**2</td><td></td><td>-0.00000<sup>*</sup></td></tr>
## <tr><td style="text-align:left"></td><td></td><td>(0.00000)</td></tr>
## <tr><td style="text-align:left"></td><td></td><td></td></tr>
## <tr><td style="text-align:left">constant-term</td><td>13.537<sup>***</sup></td><td>10.531<sup>***</su
```

```
## <tr><td style="text-align:left"></td><td>(1.238)</td><td>(1.967)</td></tr>
## <tr><td style="text-align:left"></td><td></td><td></td></tr>
## <tr><td colspan="3" style="border-bottom: 1px solid black"></td></tr><tr><td style="text-align:left"
## <tr><td style="text-align:left">R<sup>2</sup></td><td>0.135</td><td>0.169</td></tr>
## <tr><td style="text-align:left">Adjusted R<sup>2</sup></td><td>0.126</td><td>0.151</td></tr>
## <tr><td style="text-align:left">Residual Std. Error</td><td>7.096 (df = 94)</td><td>6.993 (df = 93)</
## <tr><td style="text-align:left">F Statistic</td><td>14.715<sup>***</sup> (df = 1; 94)</td><td>9.474<s
## <tr><td colspan="3" style="border-bottom: 1px solid black"></td></tr><tr><td style="text-align:left"
## <tr><td style="text-align:left"></td><td colspan="2" style="text-align:right">(data taken from nycfi
## </table>
```

- `dpois(x, lambda)` gives $P[X = x]$.
- `ppois(x, lambda)` gives $P[X \leq x]$, where $X \sim P(\lambda)$.
- `qpois(x, lambda)` gives such a value of x for which $P[X \leq x]$ is *at least* p.
- In case we want to find $P[X > 2]$, we can do `ppois(2, 2, lowwer.tail = FALSE)`
- `rpois(n, lambda)` is to get n random values from the $P(\lambda)$ distribution.

Distribution name	Density function	Distribution function	Quantile function	Random generation