Data visualization: flights at ABIA

Bao Doquang, Dhwanit Agarwal, Akksay Singh and Shristi Singh

February 14th, 2019

We were looking to answer, if someone has to take a flight, what is most likely and ideal (one that minimizes the total delay) and worst conditions to fly. In this regard, we have looked at three parameters: i) the average total delay for a given day of the week, ii) the average total delay for month of the travel, and iii) lastly the average delay based on the departure times of the flight, on any given day.
Here Average Total Delay is Sum of all the Delays divided by Number of Parameter, ???(Delays Given)/(Number of Parameter) Data for a given parameter was excluded if any one of the day's delay information was N/A ??? (Delays Given)= the sum of all the 7 delays given (arrival, departure, security, carrier, weather, late aircraft, and NAS delays) So average total delay by day of the week plot was calculated by summing the delays in minutes of each of the flight that took off on that day, divided by the number of that particular day in the dataset.
It turns out that flights in September, on Thursdays that depart at 12 to 1 am are on average going to be the least delayed, and 5 to 6 P.M flights in December on Sundays are going to be the most delayed flights. This makes logical sense, in that it is the Christmas and winter break season. People often also prefer to fly on the weekends, and especially in the evenings, as it is often an opportune time to prepare well (e.g pack luggage, drive to the airport, etc). The weather in December is also inclement around the United States, increasing delays as well. Similarly, in direct contraposition, it also makes logical sense that flights on Thursdays in September at 12 - 1 AM get delayed the least. Attempts were made to observe correlations between the average distance of the flights and if that was a factor in determining the expected delay, (i.e do flights that simply fly more, are they delayed more as well?). The average distance travelled by flights (computed using the same methodology as average delay) on any given day was also analyzed. Interestingly, people also took the shortest flights on Thursday as well, but apart from that, no general trend was found. As expected though, people on average took the longest flights on Saturdays. The relevant graphs are given below.

```
library(mosaic)
```

```
## Warning: package 'mosaic' was built under R version 3.6.2
```

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

```
## Loading required package: lattice
```

```
## Loading required package: ggformula
```

```
## Loading required package: ggplot2
```

```
## Loading required package: ggstance
```

```
## Warning: package 'ggstance' was built under R version 3.6.2
```

```
##
## Attaching package: 'ggstance'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     geom_errorbarh, GeomErrorbarh
```

```
##
## New to ggformula?  Try the tutorials:
##  learnr::run_tutorial("introduction", package = "ggformula")
##  learnr::run_tutorial("refining", package = "ggformula")
```

```
## Loading required package: mosaicData
```

```
## Warning: package 'mosaicData' was built under R version 3.6.2
```

```
## Loading required package: Matrix
```

```
## Registered S3 method overwritten by 'mosaic':
##   method                           from
##   fortify.SpatialPolygonsDataFrame ggplot2
```

```
##
## The 'mosaic' package masks several functions from core packages in order to add
## additional features.  The original behavior of these functions should not be affected by thi
s.
##
## Note: If you use the Matrix package, be sure to load it BEFORE loading mosaic.
```

```
##
## Attaching package: 'mosaic'
```

```
## The following object is masked from 'package:Matrix':
##
##     mean
```

```
## The following object is masked from 'package:ggplot2':
##
##     stat
```

```
## The following objects are masked from 'package:dplyr':
##
##     count, do, tally
```

```
## The following objects are masked from 'package:stats':
##
##     binom.test, cor, cor.test, cov, fivenum, IQR, median, prop.test,
##     quantile, sd, t.test, var
```

```
## The following objects are masked from 'package:base':
##
##     max, mean, min, prod, range, sample, sum
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.6.2
```

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

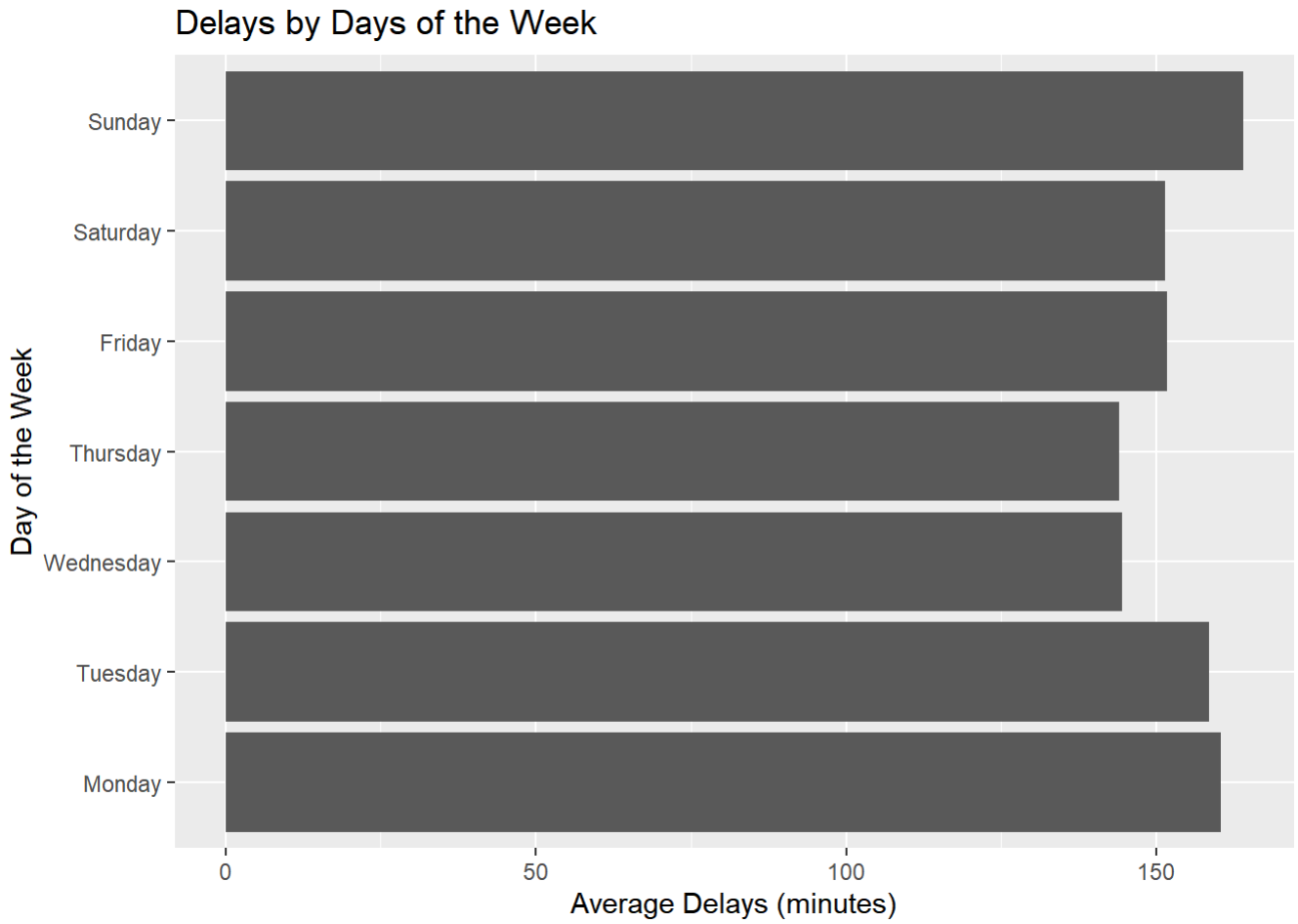
```
## <U+2713> tibble  2.1.3     <U+2713> purrr   0.3.3
## <U+2713> tidyr   1.0.0     <U+2713> stringr 1.4.0
## <U+2713> readr   1.3.1     <U+2713> forcats 0.4.0
```

```
## -- Conflicts ------------------------------------------------------------------------
----------- tidyverse_conflicts() --
## x mosaic::count()            masks dplyr::count()
## x purrr::cross()             masks mosaic::cross()
## x mosaic::do()               masks dplyr::do()
## x tidyr::expand()            masks Matrix::expand()
## x dplyr::filter()            masks stats::filter()
## x ggstance::geom_errorbarh() masks ggplot2::geom_errorbarh()
## x dplyr::lag()               masks stats::lag()
## x tidyr::pack()              masks Matrix::pack()
## x mosaic::stat()             masks ggplot2::stat()
## x mosaic::tally()            masks dplyr::tally()
## x tidyr::unpack()            masks Matrix::unpack()
```

```
library(knitr)

ABIA <- read.csv("~/GitHub/SDS323_Spring2020/ex1/ABIA.csv")
#new variable total delay
ABIA = ABIA %>% mutate(totdelay = ArrDelay + DepDelay +
                         CarrierDelay + WeatherDelay + NASDelay + SecurityDelay +LateAircraftDel
ay)

#omit NA
ABIA_edit <- na.omit(ABIA)

#fix labels
ABIA_edit <- mutate(ABIA_edit, DayOfWeek =
                      factor(DayOfWeek,levels = c(1, 2, 3, 4, 5, 6, 7),
                             labels=c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "S
aturday", "Sunday")))
ABIA_edit <- mutate(ABIA_edit, Month =
                      factor(Month,levels = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12),
                             labels=c("Jan", "Feb", "Mar", "Apr", "May", "June", "July", "Aug",
"Sept", "Oct", "Nov", "Dec")))

#Cut CRSDepTime in Factors
ABIA_edit = ABIA_edit %>%
  mutate(tod_cat = cut(CRSDepTime,
                       c(0000, 0100, 0200, 0300, 0400, 0500, 0600, 0700, 0800, 0900, 1000, 1100,
1200, 1300, 1400, 1500, 1600, 1700, 1800, 1900, 2000, 2100, 2200, 2300, 2400),
                       labels = c("12am-1am", "1am-2am", "2am-3am", "3am-4am", "4am-5am", "5am-6
am", "6am-7am", "7am-8am", "8am-9am", "9am-10am", "10am-11am", "11am-12pm", "12pm-1pm", "1pm-2p
m", "2pm-3pm", "3pm-4pm", "4pm-5pm", "5pm-6pm", "6pm-7pm", "7pm-8pm", "8pm-9pm", "9pm-10pm", "10
pm-11pm", "11pm-12am")))


#plot avg total delay by day of week
bydow = ABIA_edit %>%
  group_by(DayOfWeek) %>%
  summarize(avg.delay = sum(totdelay)/n())
bydow
```

```
## # A tibble: 7 x 2
##   DayOfWeek avg.delay
##   <fct>         <dbl>
## 1 Monday         160.
## 2 Tuesday        158.
## 3 Wednesday      145.
## 4 Thursday       144.
## 5 Friday         152.
## 6 Saturday       151.
## 7 Sunday         164.
```

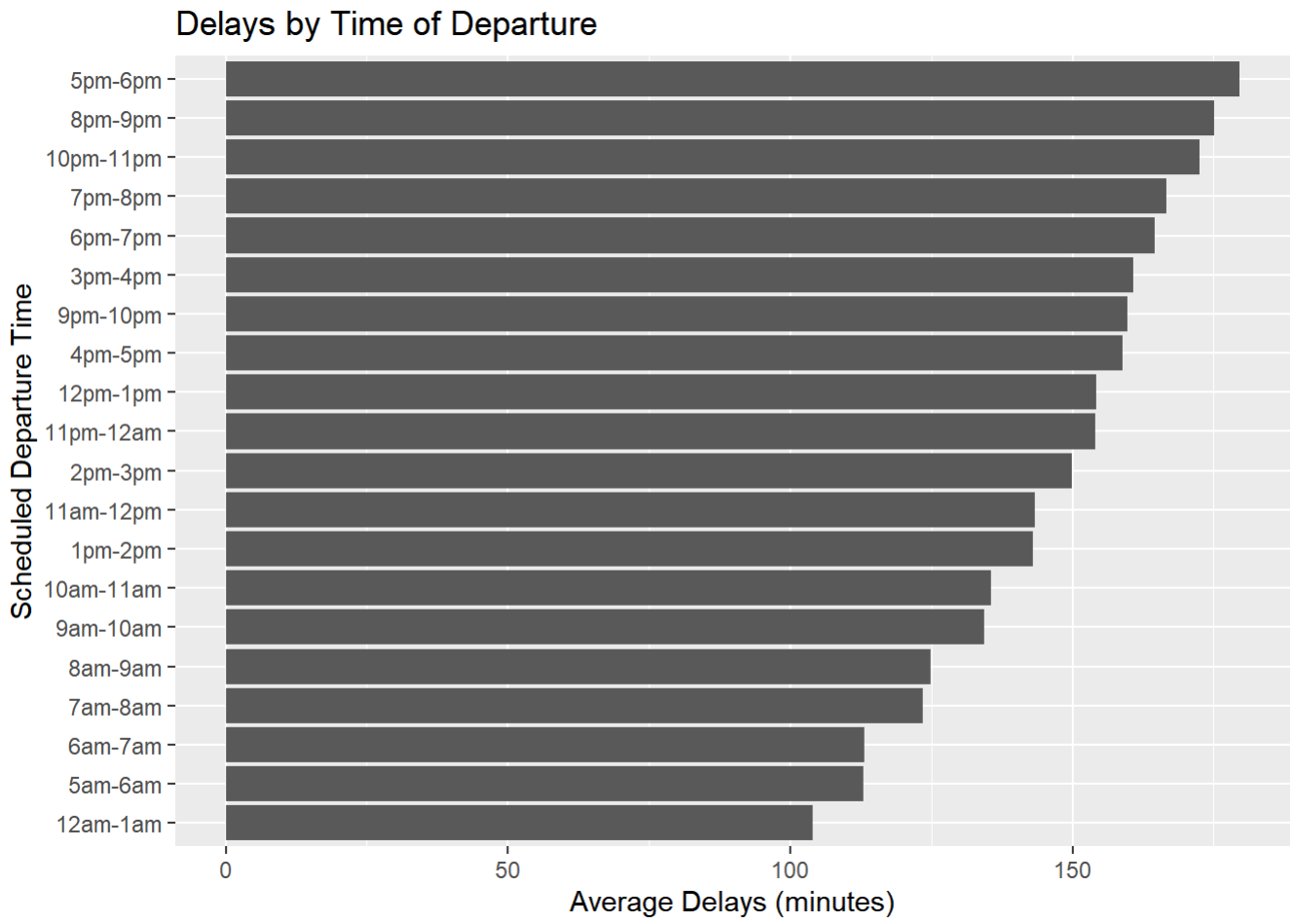
```
ggplot(data=bydow, aes(x=DayOfWeek, y=avg.delay)) +
  geom_bar(stat='identity') +
  labs(title= "Delays by Days of the Week",
       x = "Day of the Week",
    y = "Average Delays (minutes)") +
  coord_flip()
```




```
#plot avg total delay by time of day
bytod = ABIA_edit %>%
  group_by(tod_cat) %>%
  summarize(avg.delay = sum(totdelay)/n())
bytod
```

```
## # A tibble: 20 x 2
##    tod_cat   avg.delay
##    <fct>         <dbl>
##  1 12am-1am       104
##  2 5am-6am        113.
##  3 6am-7am        113.
##  4 7am-8am        123.
##  5 8am-9am        125.
##  6 9am-10am       134.
##  7 10am-11am      135.
##  8 11am-12pm      143.
##  9 12pm-1pm       154.
## 10 1pm-2pm        143.
## 11 2pm-3pm        150.
## 12 3pm-4pm        161.
## 13 4pm-5pm        159.
## 14 5pm-6pm        180.
## 15 6pm-7pm        165.
## 16 7pm-8pm        167.
## 17 8pm-9pm        175.
## 18 9pm-10pm       160.
## 19 10pm-11pm      172.
## 20 11pm-12am      154.
```

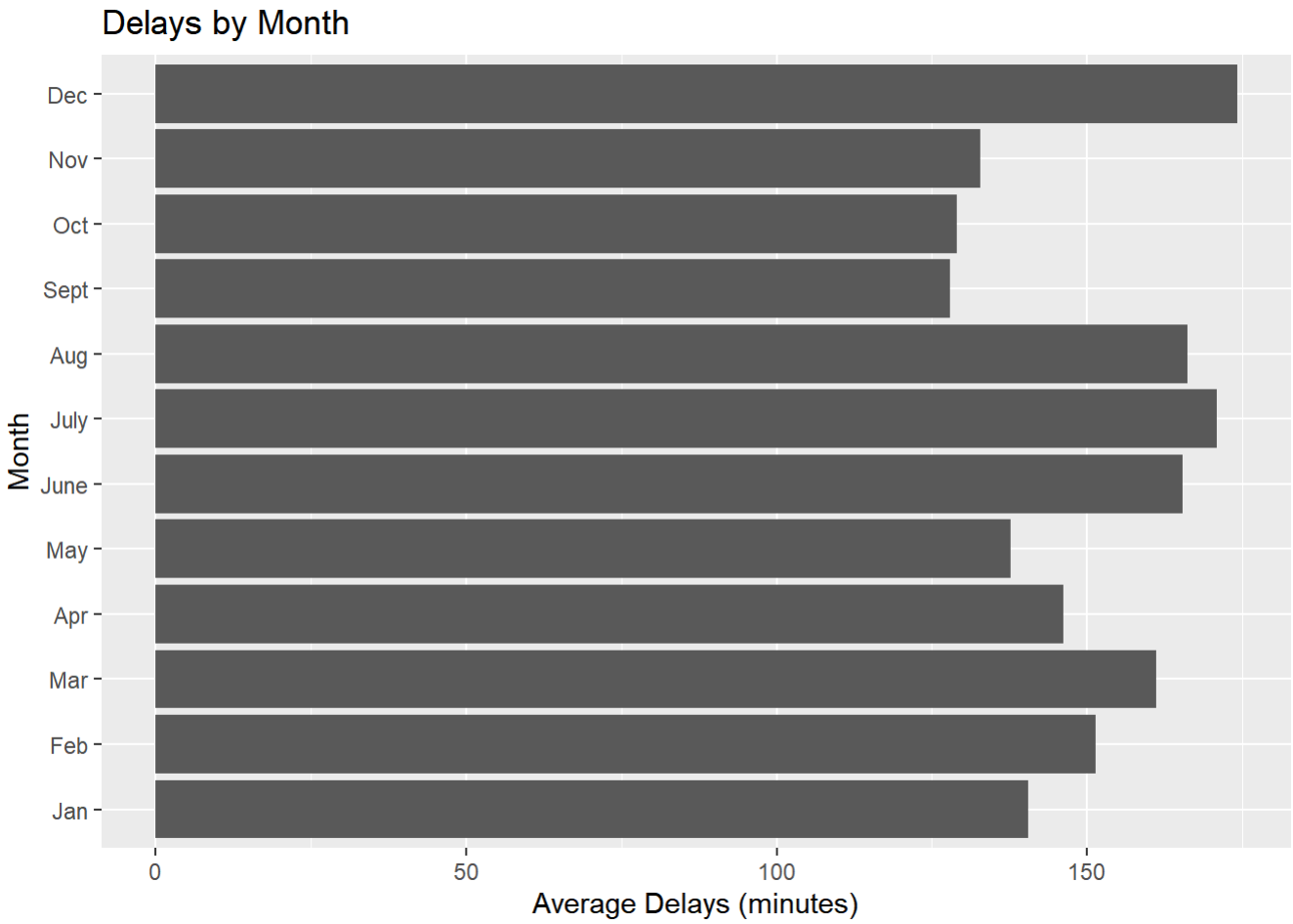
```
ggplot(data=bytod, aes(x=reorder(tod_cat, avg.delay), y=avg.delay)) +
  geom_bar(stat='identity') +
  labs(title= "Delays by Time of Departure",
       x = "Scheduled Departure Time",
       y = "Average Delays (minutes)") + coord_flip()
```




```
#plot avg total delay by month
bym = ABIA_edit %>%
  group_by(Month) %>%
  summarize(avg.delay = sum(totdelay)/n())
bym
```

```
## # A tibble: 12 x 2
##    Month avg.delay
##    <fct>     <dbl>
##  1 Jan        140.
##  2 Feb        151.
##  3 Mar        161.
##  4 Apr        146.
##  5 May        138.
##  6 June       165.
##  7 July       171.
##  8 Aug        166.
##  9 Sept       128.
## 10 Oct        129.
## 11 Nov        133.
## 12 Dec        174.
```

```
ggplot(data=bym, aes(x=Month, y=avg.delay)) +
  geom_bar(stat='identity') +
  labs(title= "Delays by Month",
       x = "Month",
       y = "Average Delays (minutes)") + coord_flip()
```




```
#plot flight distance by days of the week
distance = ABIA_edit %>%
  group_by(DayOfWeek) %>%
  summarize(avg.distance = sum(Distance)/n())
distance
```

```
## # A tibble: 7 x 2
##   DayOfWeek avg.distance
##   <fct>            <dbl>
## 1 Monday            731.
## 2 Tuesday           730.
## 3 Wednesday         726.
## 4 Thursday          699.
## 5 Friday            724.
## 6 Saturday          819.
## 7 Sunday            727.
```

```
ggplot(data=distance, aes(x=DayOfWeek, y=avg.distance)) +
  geom_bar(stat='identity') +
  labs(title= "Average Distance Traveled by Days of the Week",
       x = "Day of the Week",
       y = "Average Distance (miles)") + coord_flip()
```