Análisis al dataframe nycflights

Alejandro Zavala

2023-08-22

Contents


```
# Clear environment
rm(list = ls())
# Librerias a ocupar
library("knitr")
library("statsr")
library("dplyr")
library("ggplot2")
library("gridExtra")
```

Análisis a los datos de vuelos que salieron de Nueva York en el año 2013

Se hara un analisis al dataframe "nycflights". Contiene los datos de en tiempo para una muestra aleatoria de vuelos que salieron de Nueva York. Podemos encontrar información acerca de sus variables usando el comando **?nycflights**

Describiendo las variables del dataframe

Veamos los campos del dataframe

```
names(nycflights)
```

```
##  [1] "year"      "month"     "day"       "dep_time"  "dep_delay" "arr_time"
##  [7] "arr_delay" "carrier"   "tailnum"   "flight"    "origin"    "dest"
## [13] "air_time"  "distance"  "hour"      "minute"
```

Veamos los primeros registros de este dataframe (lo partiremos en dos para que se puedan visualizar)

```
kable(head(nycflights[,0:8]))
```

year	month	day	dep_time	dep_delay	arr_time	arr_delay	carrier
2013	6	30	940	15	1216	-4	VX
2013	5	7	1657	-3	2104	10	DL
2013	12	8	859	-1	1238	11	DL
2013	5	14	1841	-4	2122	-34	DL
2013	7	21	1102	-3	1230	-8	9E
2013	1	1	1817	-3	2008	3	AA

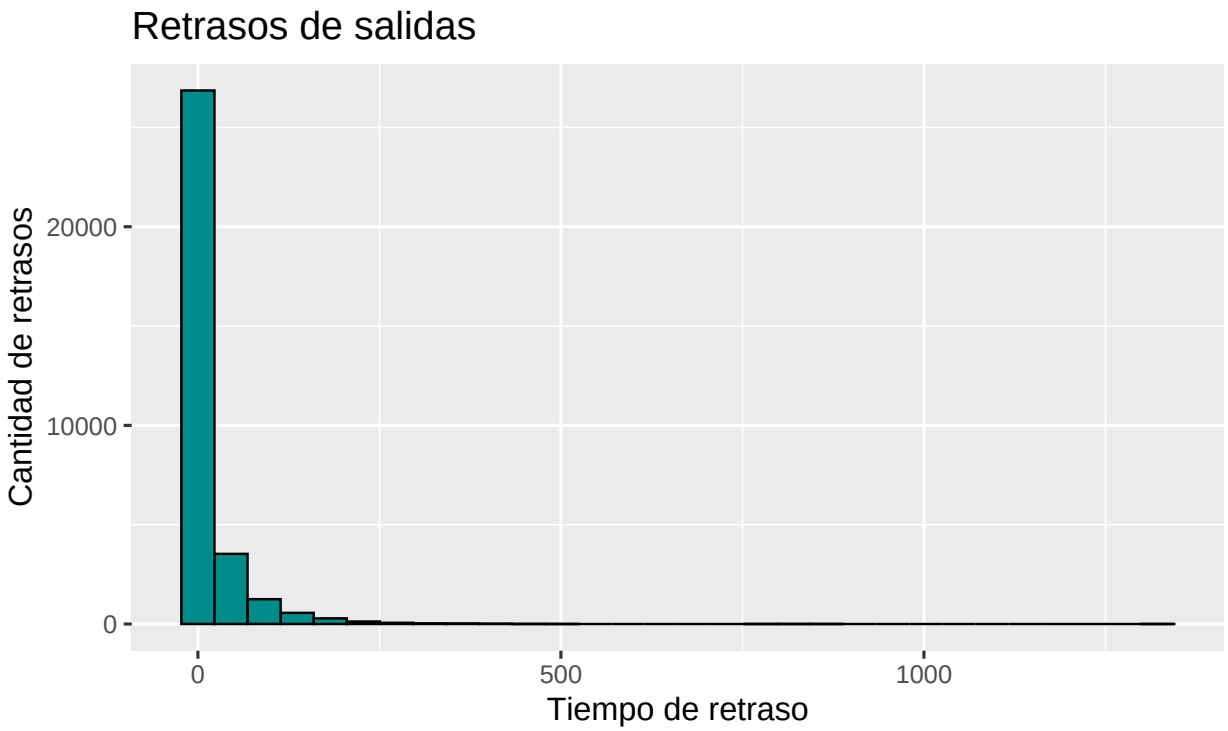
```
kable(head(nycflights[,9:16]))
```

tailnum	flight	origin	dest	air_time	distance	hour	minute
N626VA	407	JFK	LAX	313	2475	9	40
N3760C	329	JFK	SJU	216	1598	16	57
N712TW	422	JFK	LAX	376	2475	8	59
N914DL	2391	JFK	TPA	135	1005	18	41
N823AY	3652	LGA	ORF	50	296	11	2
N3AXAA	353	LGA	ORD	138	733	18	17

Podemos examinar la distribución de los retrasos en las salidas de todos los vuelos con un histograma.

```
ggplot(data = nycflights, aes(x = dep_delay)) +
  geom_histogram(color="black", fill="darkcyan") +
  ggtitle(label="Retrasos de salidas") +
  xlab("Tiempo de retraso") +
  ylab("Cantidad de retrasos") +
  theme_grey(base_size = 14)
```

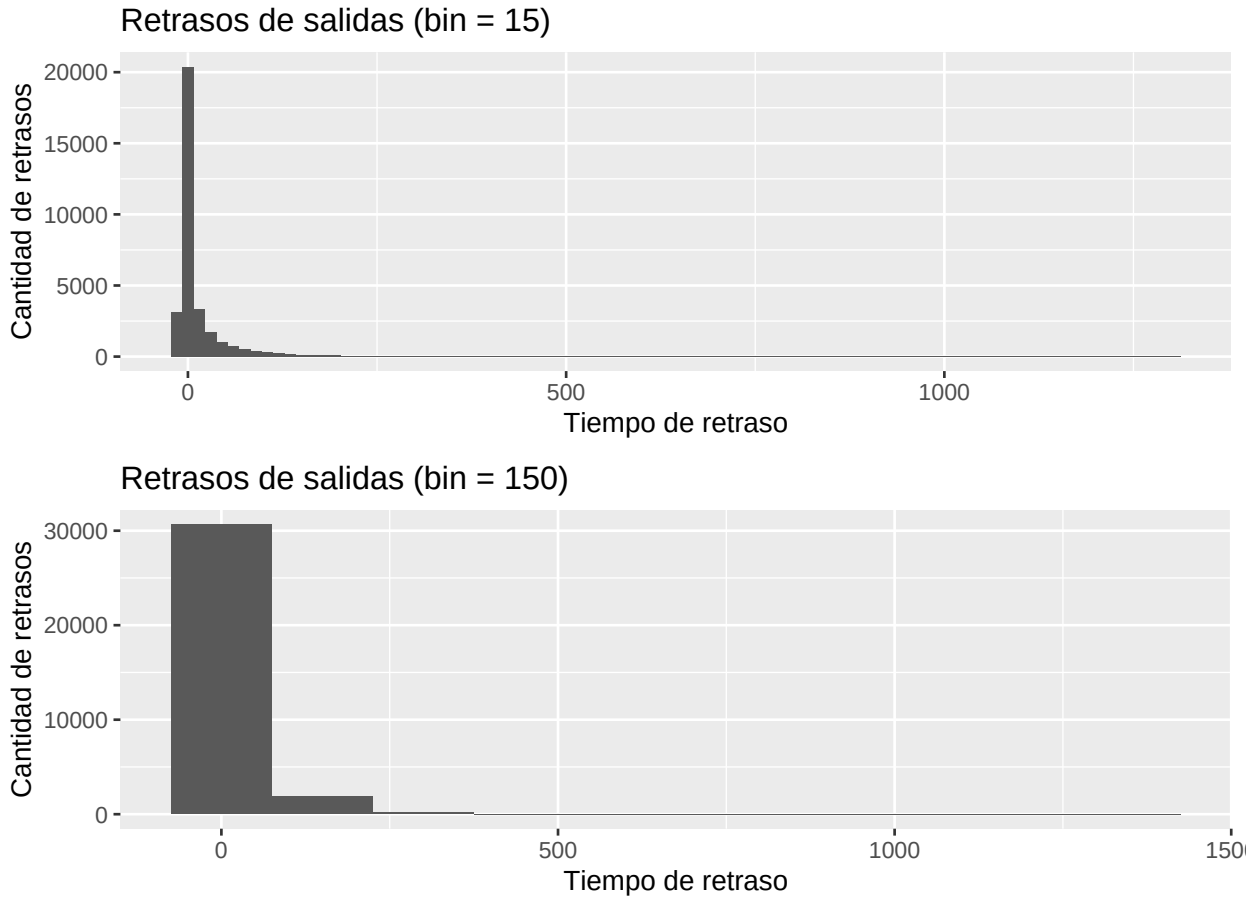
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Distribución de datos

Los histogramas son una buena manera de ver la distribución de los datos, en R se puede cambiar dependiendo de cómo se dividen los datos entre los diferentes contenedores (bins). Puede definir fácilmente el ancho del contenedor que desea utilizar (por ejemplo 15 y 150):

```
h_1 <- ggplot(data = nycflights, aes(x = dep_delay)) +
  geom_histogram(binwidth = 15) +
  ggtitle(label="Retrasos de salidas (bin = 15)") +
  xlab("Tiempo de retraso") +
  ylab("Cantidad de retrasos") +
  theme_grey(base_size = 12)

h_2 <- ggplot(data = nycflights, aes(x = dep_delay)) +
  geom_histogram(binwidth = 150) +
  ggtitle(label="Retrasos de salidas (bin = 150)") +
  xlab("Tiempo de retraso") +
  ylab("Cantidad de retrasos") +
  theme_grey(base_size = 12)

grid.arrange(h_1, h_2,ncol = 1) # Multiple plots in one
```

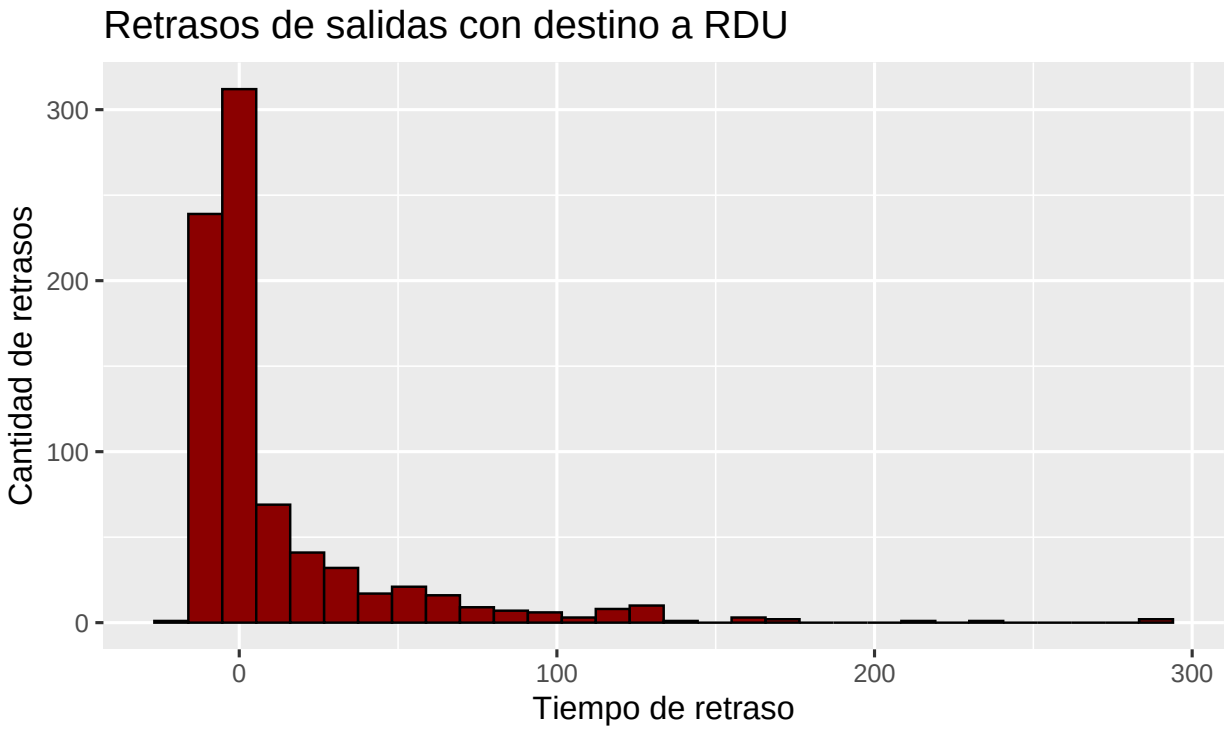


Filtrando información

Si queremos centrarnos únicamente en los retrasos en las salidas de los vuelos con destino a RDU (Aeropuerto Internacional de Raleigh-Durham), debemos primero filtrar los datos de los vuelos dirigidos a RDU (dest == "RDU") y despues haremos un histograma de retrasos en las salidas únicamente de esos vuelos. Usaremos la libreria "dplyr" para hacer filtrados de la información

```
rdu_flights <- nycflights %>% filter(dest == "RDU")
ggplot(data = rdu_flights, aes(x = dep_delay)) +
  geom_histogram(color="black", fill="darkred") +
  ggtitle(label="Retrasos de salidas con destino a RDU") +
  xlab("Tiempo de retraso") +
  ylab("Cantidad de retrasos") +
  theme_grey(base_size = 14)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Obteniendo medidas por agrupaciones

Obteniendo ahora una tabla de resumen con la media, desviación estandar y numero de datos de este dataframe

De manera manual

```
kable(data.frame(mean = c(mean(rdu_flights$dep_delay)),
                 sd = c(sd(rdu_flights$dep_delay)),
                 length = c(length(rdu_flights$dep_delay))))
```

mean	sd	length
11.69913	35.55567	801

O tambien pudiendo ocupar la libreria **dplyr**

```
agg_flights <- rdu_flights %>% summarise(mean_dd = mean(dep_delay), sd_dd = sd(dep_delay), n = n())
kable(agg_flights)
```

mean_dd	sd_dd	n
11.69913	35.55567	801

Ahora tambien podemos obtener medidas por agrupaciones, para los vuelos a RDU viendo grupos por origen y obteniendo su media, desviación estandar y el numero de datos

```
agg_flights_or <- rdu_flights %>%
  group_by(origin) %>%
  summarise(mean_dd = mean(dep_delay), sd_dd = sd(dep_delay), n = n())

kable(agg_flights_or)
```

origin	mean_dd	sd_dd	n
EWR	13.365517	32.08492	145
JFK	15.396667	40.30535	300
LGA	7.904494	32.18620	356

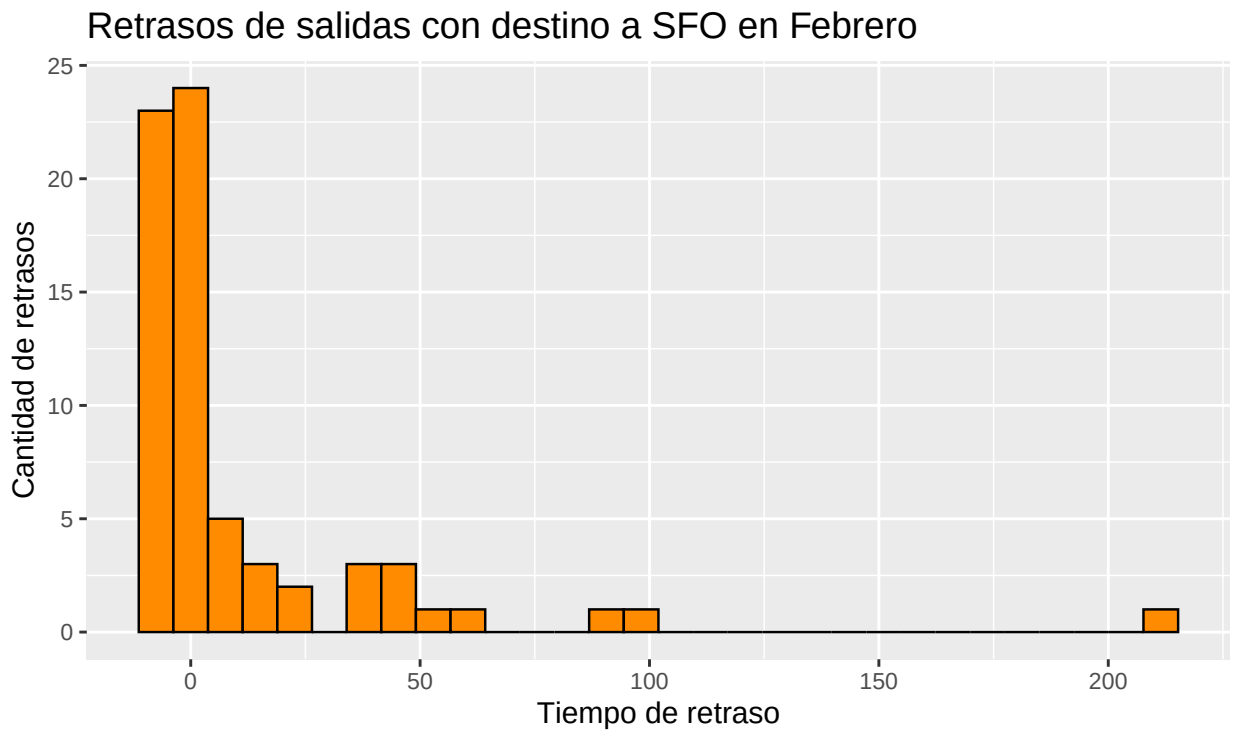
Ahora si queremos obtener los retrasos de vuelos con destino a SFO (San Francisco) en el mes de febrero. Recordando que el tiempo negativo representa salidas o llegadas anticipadas.

```
sfo_feb_flights <- nycflights %>%
  filter(dest == "SFO", month == 2)

sfo_feb_flights_agg <- sfo_feb_flights %>%
  summarise(mean_dd = mean(dep_delay), sd_dd = sd(dep_delay), n = n())

kable(sfo_feb_flights_agg)
```

mean_dd	sd_dd	n
10.5	33.27968	68

```
ggplot(data = sfo_feb_flights, aes(x = dep_delay)) +
  geom_histogram(color="black", fill="darkorange") +
  ggtitle(label="Retrasos de salidas con destino a SFO en Febrero") +
  xlab("Tiempo de retraso") +
  ylab("Cantidad de retrasos") +
  theme_grey(base_size = 13)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Calculemos ahora la mediana y el rango intercuartil para los vuelos de llegada en el dataframe, agrupado por transportista.

```
tbl_sfo_feb_flights <- sfo_feb_flights %>%
  group_by(carrier) %>%
  summarise(median_ad = median(arr_delay),
            iqr_ad = IQR(arr_delay),
            percent25 = quantile(arr_delay, probs = 0.25),
            percent75 = quantile(arr_delay, probs = 0.75),
            n_flights = n()
            )
kable(tbl_sfo_feb_flights)
```

carrier	median_ad	iqr_ad	percent25	percent75	n_flights
AA	5.0	17.50	-2.00	15.50	10
B6	-10.5	12.25	-12.50	-0.25	6
DL	-15.0	22.00	-27.50	-5.50	19
UA	-10.0	22.00	-20.00	2.00	21
VX	-22.5	21.25	-32.25	-11.00	12

Análisis de interes

¿En qué mes esperaría usted tener el mayor retraso promedio en la salida desde un aeropuerto de Nueva York?

```
agg_month_meandd <- nycflights %>%
  group_by(month) %>%
  summarise(mean_dd = mean(dep_delay)) %>%
  arrange(desc(mean_dd))

kable(agg_month_meandd)
```

month	mean_dd
7	20.754559
6	20.350293
12	17.368188
4	14.554477
3	13.517602
5	13.264800
8	12.619097
2	10.687227
1	10.233333
9	6.872436
11	6.103183
10	5.880375

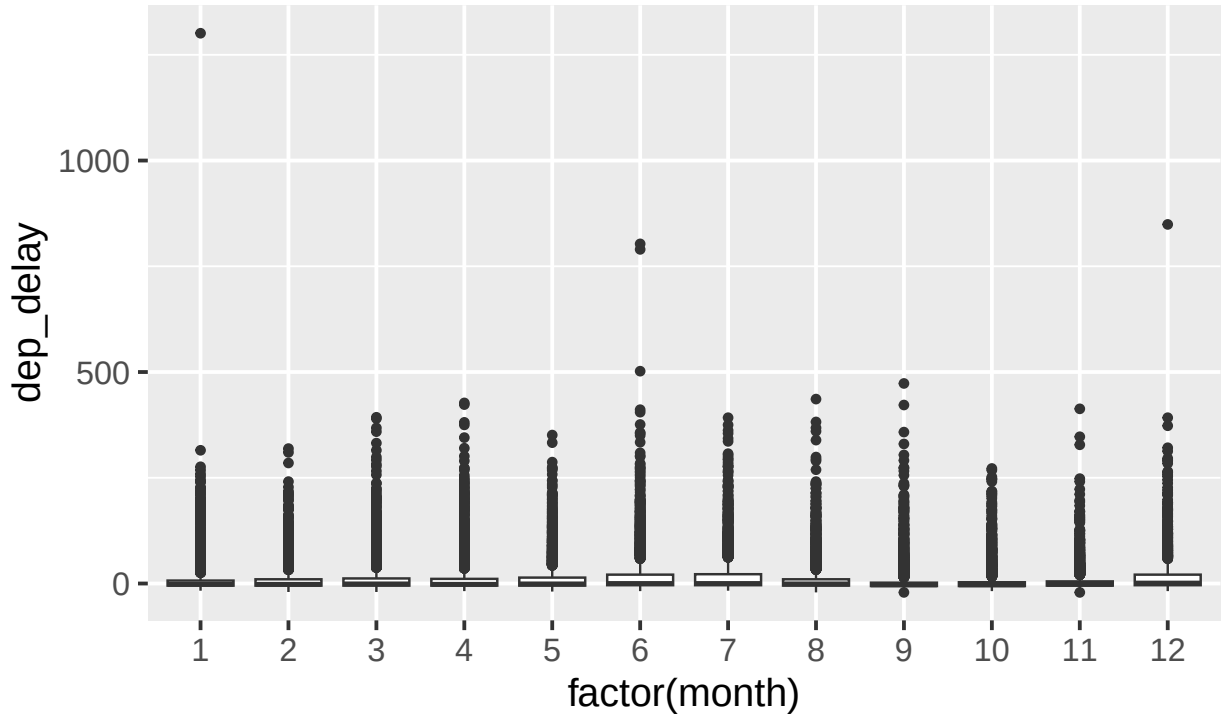
¿Qué mes tiene el mayor retraso medio en las salidas desde un aeropuerto de Nueva York?

```
agg_month_mediandd <- nycflights %>%
  group_by(month) %>%
  summarise(median_dd = median(dep_delay)) %>%
  arrange(desc(median_dd))

kable(agg_month_mediandd)
```

month	median_dd
12	1
6	0
7	0
3	-1
5	-1
8	-1
1	-2
2	-2
4	-2
11	-2
9	-3
10	-3

Veamos los diagramas de caja por mes

```
ggplot(nycflights, aes(x = factor(month), y = dep_delay)) +
  geom_boxplot() +
  theme_grey(base_size = 17)
```



Crearemos un nuevo dataframe el cual tendra una columna nueva donde el tiempo de retraso si es menor a 5 minutos sera en tiempo (on time) y si es mayor lo marcaremos como retraso (delayed)

```
nycflights_on_d <- nycflights %>%
  mutate(dep_type = ifelse(dep_delay < 5, "on time", "delayed"))
```

Veamos la nueva variable que agrego

```
kable(head(nycflights_on_d[,0:8]))
```

year	month	day	dep_time	dep_delay	arr_time	arr_delay	carrier
2013	6	30	940	15	1216	-4	VX
2013	5	7	1657	-3	2104	10	DL
2013	12	8	859	-1	1238	11	DL
2013	5	14	1841	-4	2122	-34	DL
2013	7	21	1102	-3	1230	-8	9E
2013	1	1	1817	-3	2008	3	AA

```
kable(head(nycflights_on_d[,9:17]))
```

tailnum	flight	origin	dest	air_time	distance	hour	minute	dep_type
N626VA	407	JFK	LAX	313	2475	9	40	delayed
N3760C	329	JFK	SJU	216	1598	16	57	on time
N712TW	422	JFK	LAX	376	2475	8	59	on time
N914DL	2391	JFK	TPA	135	1005	18	41	on time
N823AY	3652	LGA	ORF	50	296	11	2	on time
N3AXAA	353	LGA	ORD	138	733	18	17	on time

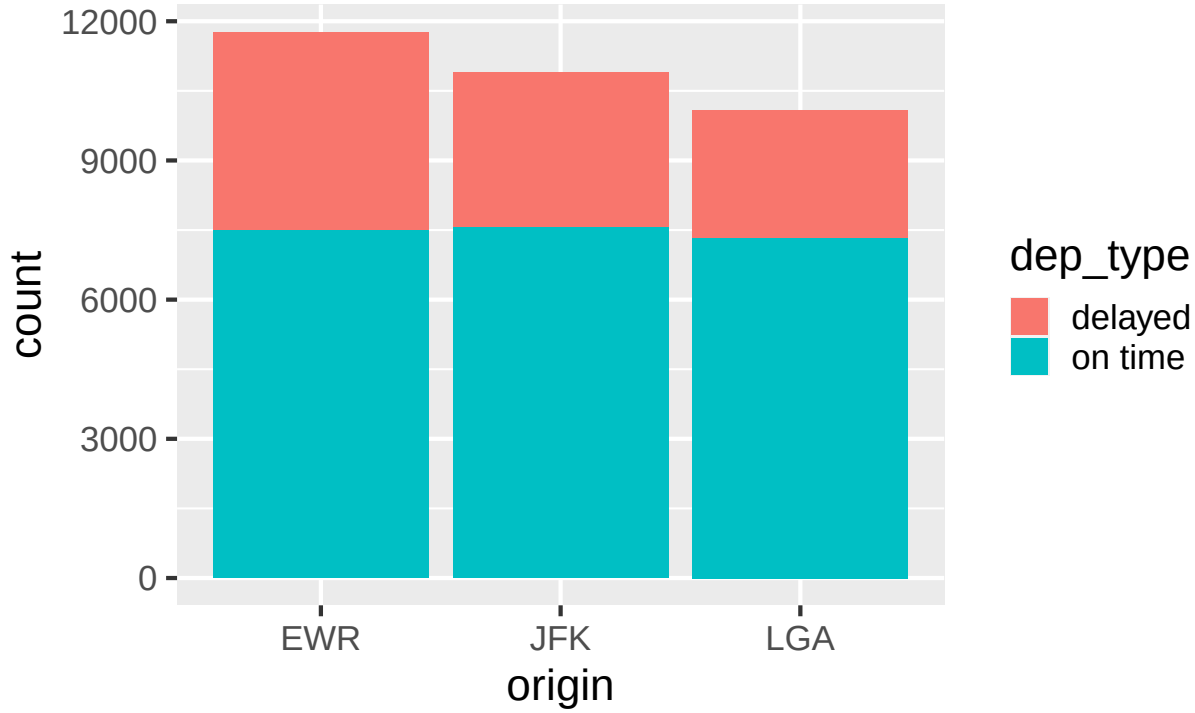
Haremos un agrupado por origen y tomando la proporción si es un viaje en tiempo o anticipado comparandolo con algun retardo

```
agg_or_dept <- nycflights_on_d %>%
  group_by(origin) %>%
  summarise(ot_dep_rate = sum(dep_type == "on time") / n() , dl_dep_rate = sum(dep_type != "on time") /
  arrange(desc(ot_dep_rate))

kable(agg_or_dept)
```

origin	ot_dep_rate	dl_dep_rate
LGA	0.7279229	0.2720771
JFK	0.6935854	0.3064146
EWR	0.6369892	0.3630108

También podemos visualizar la distribución de la tasa de salidas puntuales en los tres aeropuertos utilizando un gráfico de barras segmentadas.

```
ggplot(data = nycflights_on_d, aes(x = origin, fill = dep_type)) +
  geom_bar() + theme_grey(base_size = 19)
```



Ahora, si no queremos los retrasos en la salida y deseamos programar un viaje en un mes que minimice el posible retraso en la salida al salir de Nueva York. Una alternativa podria ser elegir el mes con el menor retraso medio en la salida. Otra opción es elegir el mes con el retraso de mediana de salida más bajo. ¿Cuáles son los pros y los contras de estas dos opciones?

Media: representa el promedio general, teniendo en cuenta el efecto de cada retraso y dando una idea de cómo se distribuyen los datos. Desventaja: esto puede verse sesgado por valores atípicos.

Mediana: toma el valor medio de todo el conjunto de datos, por lo que los valores atípicos no sesgan la mediana. Desventaja: no representa cómo se distribuyen los datos.

Ahora, crearemos una variable llamada velocidad promedio, se puede calcular como la distancia dividida por el número de horas de viaje, y tenga en cuenta que el tiempo_aire se expresa en minutos. Una vez aplicando ese filtro y ademas ordenando descendentemente veremos los vuelos con la velocidad mas alta.

```
nycflights_sp <- nycflights %>%
  mutate(avg_speed = 60*(distance / air_time)) %>%
  arrange(desc(avg_speed))

kable(head(nycflights_sp[,0:8]))
```

year	month	day	dep_time	dep_delay	arr_time	arr_delay	carrier
2013	5	25	1709	9	1923	-14	DL
2013	2	21	2355	-3	412	-26	B6
2013	11	16	2349	-10	402	-38	B6
2013	2	22	831	4	1258	-18	B6
2013	12	5	1628	-2	2045	-45	AA
2013	3	9	2214	79	229	43	B6

```
kable(head(nycflights_sp[,9:17]))
```

tailnum	flight	origin	dest	air_time	distance	hour	minute	avg_speed
N666DN	1499	LGA	ATL	65	762	17	9	703.3846
N779JB	707	JFK	SJU	172	1598	23	55	557.4419
N571JB	1503	JFK	SJU	173	1598	23	49	554.2197
N568JB	403	JFK	SJU	175	1598	8	31	547.8857
N5EHAA	95	JFK	SJU	175	1598	16	28	547.8857
N656JB	701	JFK	SJU	175	1598	22	14	547.8857

Ahora, haciendo un gráfico de dispersión entre la distancia y velocidad, se obtiene:

```
ggplot(data = nycflights_sp, aes(x = distance, y = avg_speed)) +
  geom_point() +
  ggtitle(label="Velocidad promedio de vuelos de NY") +
  xlab("Distancia") +
  ylab("Velocidad promedio") +
  theme_grey(base_size = 19)
```