dplyr

October 22, 2020

0.1 Dplyr

Instalar paquetes. Para empezar:

[7]:
```
#Jalamos los paquetes que vamos a utilizar
if(!require(dplyr, quietly = TRUE, warn.conflicts = FALSE) ){
    install.packages('dplyr',
        dependencies = TRUE,
        repos = "http://cran.us.r-project.org")
}
if(!require(nycflights13, quietly = TRUE, warn.conflicts = FALSE) ){
    install.packages('nycflights13',
        dependencies = TRUE,
        repos = "http://cran.us.r-project.org")
}
```

[8]:
```
install.packages('tidyverse')
```

```
Updating HTML index of packages in '.Library'

Making 'packages.html' …
 done
```

[9]:
```
library(tidyverse)
```

[ ]:
```
library(lubridate)
```

¡Exploremos un poco!

[10]:
```
library(dplyr)
library(nycflights13)
head(flights)
```

A tibble: 6 × 19

year <int>	month <int>	day <int>	dep_time <int>	sched_dep_time <int>	dep_delay <dbl>	arr_time <int>	sched_arr_ti
2013	1	1	517	515	2	830	819
2013	1	1	533	529	4	850	830
2013	1	1	542	540	2	923	850
2013	1	1	544	545	-1	1004	1022
2013	1	1	554	600	-6	812	837
2013	1	1	554	558	-4	740	728

¡Exploremos un poco!

[11]:
```
class(flights)
```

1. 'tbl_df' 2. 'tbl' 3. 'data.frame'

¡Exploremos un poco!

[12]:
```
str(flights)
#Compactly display the internal structure of an R object
```

```
tibble [336,776 × 19] (S3: tbl_df/tbl/data.frame)
 $ year          : int [1:336776] 2013 2013 2013 2013 2013 2013 2013 2013 2013
2013 …
 $ month         : int [1:336776] 1 1 1 1 1 1 1 1 1 …
 $ day           : int [1:336776] 1 1 1 1 1 1 1 1 1 …
 $ dep_time      : int [1:336776] 517 533 542 544 554 554 555 557 557 558 …
 $ sched_dep_time: int [1:336776] 515 529 540 545 600 558 600 600 600 600 …
 $ dep_delay     : num [1:336776] 2 4 2 -1 -6 -4 -5 -3 -3 -2 …
 $ arr_time      : int [1:336776] 830 850 923 1004 812 740 913 709 838 753 …
 $ sched_arr_time: int [1:336776] 819 830 850 1022 837 728 854 723 846 745 …
 $ arr_delay     : num [1:336776] 11 20 33 -18 -25 12 19 -14 -8 8 …
 $ carrier       : chr [1:336776] "UA" "UA" "AA" "B6" …
 $ flight        : int [1:336776] 1545 1714 1141 725 461 1696 507 5708 79 301
…
 $ tailnum       : chr [1:336776] "N14228" "N24211" "N619AA" "N804JB" …
 $ origin        : chr [1:336776] "EWR" "LGA" "JFK" "JFK" …
 $ dest          : chr [1:336776] "IAH" "IAH" "MIA" "BQN" …
 $ air_time      : num [1:336776] 227 227 160 183 116 150 158 53 140 138 …
 $ distance      : num [1:336776] 1400 1416 1089 1576 762 …
 $ hour          : num [1:336776] 5 5 5 5 6 5 6 6 6 6 …
 $ minute        : num [1:336776] 15 29 40 45 0 58 0 0 0 0 …
 $ time_hour     : POSIXct[1:336776], format: "2013-01-01 05:00:00" "2013-01-01
05:00:00" …
```

[13]:
```
?str
```

Verbos Dplyr tiene verbos que usualmente aplicamos a las bases de datos.

- `filter()`

- `slice()`
- `select()`
- `rename()`
- `distinct()`
- `mutate()`
- `transmute()`
- `summarise()`
- `sample_n()`
- `sample_frac()`

0.2 Verbos

Usos y costumbres

0.2.1 `filter`

Filtra el data frame con base en las distintas variables que tengas.

[14]:
```
filter(flights, month == 10, day == 31)
```

	year	month	day	dep_time	sched_dep_time	dep_delay	arr_time	sched_arr_
	<int>	<int>	<int>	<int>	<int>	<dbl>	<int>	<int>
	2013	10	31	458	500	-2	638	651
	2013	10	31	513	517	-4	824	757
	2013	10	31	542	545	-3	818	855
	2013	10	31	543	545	-2	852	827
	2013	10	31	550	600	-10	824	854
	2013	10	31	552	600	-8	703	659
	2013	10	31	553	600	-7	649	701
	2013	10	31	553	600	-7	925	943
	2013	10	31	554	600	-6	713	711
	2013	10	31	554	600	-6	834	856
	2013	10	31	555	600	-5	752	749
	2013	10	31	555	600	-5	707	717
	2013	10	31	555	600	-5	658	716
	2013	10	31	555	600	-5	730	730
	2013	10	31	556	600	-4	721	721
	2013	10	31	556	600	-4	926	923
	2013	10	31	556	600	-4	830	851
	2013	10	31	556	600	-4	657	658
	2013	10	31	556	600	-4	758	758
	2013	10	31	558	600	-2	734	715
	2013	10	31	559	600	-1	819	828
	2013	10	31	559	600	-1	848	905
	2013	10	31	602	600	2	839	815
	2013	10	31	603	610	-7	800	811
	2013	10	31	606	615	-9	736	750
	2013	10	31	606	610	-4	851	855
	2013	10	31	607	610	-3	804	745
	2013	10	31	608	615	-7	755	818
	2013	10	31	609	615	-6	757	821
A tibble: 922 × 19	2013	10	31	610	615	-5	802	817
	2013	10	31	2141	2130	11	16	18
	2013	10	31	2142	2130	12	5	2359
	2013	10	31	2145	2106	39	2247	2213
	2013	10	31	2146	2150	-4	26	36
	2013	10	31	2151	2140	11	2307	2250
	2013	10	31	2151	2110	41	10	2341
	2013	10	31	2154	2059	55	2254	2211
	2013	10	31	2155	2159	-4	2248	2306
	2013	10	31	2156	2159	-3	2258	2308
	2013	10	31	2202	2159	3	2317	2327
	2013	10	31	2204	2124	40	23	2337
	2013	10	31	2216	2110	66	2353	2255
	2013	10	31	2225	2159	26	2317	2304
	2013	10	31	2235	2245	-10	2342	3
	2013	10	31	2236	1910	206	103	2215
	2013	10	31	2238	2245	-7	2346	2353
	2013	10	31	2240	2245	-5	2342	2355
	2013	10	31	2240	2250	-10	2353	8
	2013	10	31	2242	2030	132	2357	2150
	2013	10	31	2245	2250	-5	2348	2356

0.2.2 `slice`

Filtra y **selecciona** en función del **número de renglón**.

[15]:
```
slice(flights, 1:10)
```

A tibble: 10 × 19

year <int>	month <int>	day <int>	dep_time <int>	sched_dep_time <int>	dep_delay <dbl>	arr_time <int>	sched_arr_ <int>
2013	1	1	517	515	2	830	819
2013	1	1	533	529	4	850	830
2013	1	1	542	540	2	923	850
2013	1	1	544	545	-1	1004	1022
2013	1	1	554	600	-6	812	837
2013	1	1	554	558	-4	740	728
2013	1	1	555	600	-5	913	854
2013	1	1	557	600	-3	709	723
2013	1	1	557	600	-3	838	846
2013	1	1	558	600	-2	753	745

0.2.3 `arrange`

Ordena los renglones del data frame en función de distintas variables a elegir.

[16]:
```
arrange(flights, year, desc(month), day)
#desc --> indica orden descendente
```

year <int>	month <int>	day <int>	dep_time <int>	sched_dep_time <int>	dep_delay <dbl>	arr_time <int>	sched_
2013	12	1	13	2359	14	446	445
2013	12	1	17	2359	18	443	437
2013	12	1	453	500	-7	636	651
2013	12	1	520	515	5	749	808
2013	12	1	536	540	-4	845	850
2013	12	1	540	550	-10	1005	1027
2013	12	1	541	545	-4	734	755
2013	12	1	546	545	1	826	835
2013	12	1	549	600	-11	648	659
2013	12	1	550	600	-10	825	854
2013	12	1	554	600	-6	827	849
2013	12	1	554	601	-7	748	811
2013	12	1	554	600	-6	712	719
2013	12	1	554	600	-6	645	705
2013	12	1	555	600	-5	805	827
2013	12	1	556	600	-4	846	846
2013	12	1	556	600	-4	742	756
2013	12	1	557	600	-3	733	754
2013	12	1	557	600	-3	828	834
2013	12	1	558	600	-2	841	856
2013	12	1	558	600	-2	721	717
2013	12	1	558	600	-2	718	725
2013	12	1	558	600	-2	724	743
2013	12	1	559	600	-1	709	719
2013	12	1	600	600	0	1041	1043
2013	12	1	600	600	0	717	736
2013	12	1	602	600	2	724	738
2013	12	1	603	605	-2	731	735
2013	12	1	604	608	-4	818	829
2013	12	1	604	600	4	827	840
2013	1	31	NA	1245	NA	NA	1600
2013	1	31	NA	1240	NA	NA	1540
2013	1	31	NA	1200	NA	NA	1304
2013	1	31	NA	1415	NA	NA	1724
2013	1	31	NA	825	NA	NA	1026
2013	1	31	NA	1130	NA	NA	1334
2013	1	31	NA	1500	NA	NA	1653
2013	1	31	NA	600	NA	NA	703
2013	1	31	NA	700	NA	NA	807
2013	1	31	NA	800	NA	NA	908
2013	1	31	NA	1200	NA	NA	1305
2013	1	31	NA	1300	NA	NA	1406
2013	1	31	NA	1500	NA	NA	1608
2013	1	31	NA	2100	NA	NA	2207
2013	1	31	NA	700	NA	NA	807
2013	1	31	NA	800	NA	NA	917
2013	1	31	NA	900	NA	NA	1022
2013	1	31	NA	1510	NA	NA	1650
2013	1	31	NA	1940	NA	NA	2100
2013	1	31	NA	1435	NA	NA	1559

A tibble: 336776 × 19

0.2.4 `select`

Selecciona columnas de un data frame, para quedarnos con un subconjunto de las mismas (como en un select de SQL)

[17]:
```
select(flights, year, month, day, carrier, origin, dest)
```

year	month	day	carrier	origin	dest
<int>	<int>	<int>	<chr>	<chr>	<chr>
2013	1	1	UA	EWR	IAH
2013	1	1	UA	LGA	IAH
2013	1	1	AA	JFK	MIA
2013	1	1	B6	JFK	BQN
2013	1	1	DL	LGA	ATL
2013	1	1	UA	EWR	ORD
2013	1	1	B6	EWR	FLL
2013	1	1	EV	LGA	IAD
2013	1	1	B6	JFK	MCO
2013	1	1	AA	LGA	ORD
2013	1	1	B6	JFK	PBI
2013	1	1	B6	JFK	TPA
2013	1	1	UA	JFK	LAX
2013	1	1	UA	EWR	SFO
2013	1	1	AA	LGA	DFW
2013	1	1	B6	JFK	BOS
2013	1	1	UA	EWR	LAS
2013	1	1	B6	LGA	FLL
2013	1	1	MQ	LGA	ATL
2013	1	1	B6	EWR	PBI
2013	1	1	DL	LGA	MSP
2013	1	1	MQ	LGA	DTW
2013	1	1	AA	EWR	MIA
2013	1	1	DL	JFK	ATL
2013	1	1	UA	EWR	MIA
2013	1	1	MQ	EWR	ORD
2013	1	1	UA	JFK	SFO
2013	1	1	B6	JFK	RSW
2013	1	1	B6	JFK	SJU

A tibble: 336776 × 6

2013	1	1	DL	EWR	ATL
2013	9	30	EV	LGA	CHO
2013	9	30	EV	EWR	CLT
2013	9	30	B6	JFK	DEN
2013	9	30	EV	LGA	RIC
2013	9	30	MQ	JFK	DCA
2013	9	30	AA	JFK	LAX
2013	9	30	EV	EWR	PWM
2013	9	30	B6	JFK	SJU
2013	9	30	B6	LGA	FLL
2013	9	30	UA	EWR	BOS
2013	9	30	EV	EWR	MHT
2013	9	30	9E	JFK	BUF
2013	9	30	EV	LGA	BGR
2013	9	30	MQ	LGA	BNA
2013	9	30	EV	EWR	STL
2013	9	30	B6	JFK	PWM
2013	9	30	UA	EWR	SFO
2013	9	30	B6	JFK	MCO
2013	9	30	B6	JFK	BTV
2013	9	30	B6	JFK	SYR

0.2.5 `select`

[18]:
```
select(flights, year:day)
```

A tibble: 336776 × 3

year	month	day
<int>	<int>	<int>
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	1	1
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30
2013	9	30

0.2.6 select

[19]:
```
select(flights, -year)
#Selecciona todos menos year
```

month	day	dep_time	sched_dep_time	dep_delay	arr_time	sched_arr_time
<int>	<int>	<int>	<int>	<dbl>	<int>	<int>
1	1	517	515	2	830	819
1	1	533	529	4	850	830
1	1	542	540	2	923	850
1	1	544	545	-1	1004	1022
1	1	554	600	-6	812	837
1	1	554	558	-4	740	728
1	1	555	600	-5	913	854
1	1	557	600	-3	709	723
1	1	557	600	-3	838	846
1	1	558	600	-2	753	745
1	1	558	600	-2	849	851
1	1	558	600	-2	853	856
1	1	558	600	-2	924	917
1	1	558	600	-2	923	937
1	1	559	600	-1	941	910
1	1	559	559	0	702	706
1	1	559	600	-1	854	902
1	1	600	600	0	851	858
1	1	600	600	0	837	825
1	1	601	600	1	844	850
1	1	602	610	-8	812	820
1	1	602	605	-3	821	805
1	1	606	610	-4	858	910
1	1	606	610	-4	837	845
1	1	607	607	0	858	915
1	1	608	600	8	807	735
1	1	611	600	11	945	931
1	1	613	610	3	925	921
1	1	615	615	0	1039	1100
1	1	615	615	0	833	842

A tibble: 336776 × 18

9	30	2123	2125	-2	2223	2247
9	30	2127	2129	-2	2314	2323
9	30	2128	2130	-2	2328	2359
9	30	2129	2059	30	2230	2232
9	30	2131	2140	-9	2225	2255
9	30	2140	2140	0	10	40
9	30	2142	2129	13	2250	2239
9	30	2145	2145	0	115	140
9	30	2147	2137	10	30	27
9	30	2149	2156	-7	2245	2308
9	30	2150	2159	-9	2250	2306
9	30	2159	1845	194	2344	2030
9	30	2203	2205	-2	2339	2331
9	30	2207	2140	27	2257	2250
9	30	2211	2059	72	2339	2242
9	30	2231	2245	-14	2335	2356
9	30	2233	2113	80	112	30
9	30	2235	2001	154	59	2249
9	30	2237	2245	-8	2345	2353
9	30	2240	2245	-5	2334	2351

Se pueden usar funciones para *matchear* como `contains()`, `starts_with()`, etc. También se pueden renombrar variables en el proceso.

0.2.7 `rename`

La manera más limpia de **renombrar variables**.

[20]:
```
rename(flights, mes = month)
#El de la izquiera es el nuevo nombre
```

year <int>	mes <int>	day <int>	dep_time <int>	sched_dep_time <int>	dep_delay <dbl>	arr_time <int>	sched_a <int>
2013	1	1	517	515	2	830	819
2013	1	1	533	529	4	850	830
2013	1	1	542	540	2	923	850
2013	1	1	544	545	-1	1004	1022
2013	1	1	554	600	-6	812	837
2013	1	1	554	558	-4	740	728
2013	1	1	555	600	-5	913	854
2013	1	1	557	600	-3	709	723
2013	1	1	557	600	-3	838	846
2013	1	1	558	600	-2	753	745
2013	1	1	558	600	-2	849	851
2013	1	1	558	600	-2	853	856
2013	1	1	558	600	-2	924	917
2013	1	1	558	600	-2	923	937
2013	1	1	559	600	-1	941	910
2013	1	1	559	559	0	702	706
2013	1	1	559	600	-1	854	902
2013	1	1	600	600	0	851	858
2013	1	1	600	600	0	837	825
2013	1	1	601	600	1	844	850
2013	1	1	602	610	-8	812	820
2013	1	1	602	605	-3	821	805
2013	1	1	606	610	-4	858	910
2013	1	1	606	610	-4	837	845
2013	1	1	607	607	0	858	915
2013	1	1	608	600	8	807	735
2013	1	1	611	600	11	945	931
2013	1	1	613	610	3	925	921
2013	1	1	615	615	0	1039	1100
2013	1	1	615	615	0	833	842
2013	9	30	2123	2125	-2	2223	2247
2013	9	30	2127	2129	-2	2314	2323
2013	9	30	2128	2130	-2	2328	2359
2013	9	30	2129	2059	30	2230	2232
2013	9	30	2131	2140	-9	2225	2255
2013	9	30	2140	2140	0	10	40
2013	9	30	2142	2129	13	2250	2239
2013	9	30	2145	2145	0	115	140
2013	9	30	2147	2137	10	30	27
2013	9	30	2149	2156	-7	2245	2308
2013	9	30	2150	2159	-9	2250	2306
2013	9	30	2159	1845	194	2344	2030
2013	9	30	2203	2205	-2	2339	2331
2013	9	30	2207	2140	27	2257	2250
2013	9	30	2211	2059	72	2339	2242
2013	9	30	2231	2245	-14	2335	2356
2013	9	30	2233	2113	80	112	30
2013	9	30	2235	2001	154	59	2249
2013	9	30	2237	2245	-8	2345	2353
2013	9	30	2240	2245	-5	2334	2351

A tibble: 336776 × 19

0.2.8 `distinct`

Quita los duplicados del data frame.

[21]:
```
distinct(select(flights, origin, dest))
#Quitamos los duplicados de las columnas origin y dest
```

origin	dest
<chr>	<chr>
EWR	IAH
LGA	IAH
JFK	MIA
JFK	BQN
LGA	ATL
EWR	ORD
EWR	FLL
LGA	IAD
JFK	MCO
LGA	ORD
JFK	PBI
JFK	TPA
JFK	LAX
EWR	SFO
LGA	DFW
JFK	BOS
EWR	LAS
LGA	FLL
EWR	PBI
LGA	MSP
LGA	DTW
EWR	MIA
JFK	ATL
JFK	SFO
JFK	RSW
JFK	SJU
EWR	ATL
EWR	PHX
LGA	MIA

A tibble: 224 × 2

EWR	MSP
JFK	ACK
LGA	BGR
LGA	MSN
LGA	ORF
JFK	IAH
JFK	MCI
LGA	OMA
LGA	DSM
LGA	GSP
JFK	ABQ
LGA	ILM
LGA	SYR
JFK	MVY
LGA	SBN
JFK	STL
LGA	LEX
EWR	SBN
LGA	MHT
LGA	CAE
JFK	JAC

0.2.9 `mutate`

Genera nuevas variables, se pueden usar el resto de los renglones para crear **nuevas variables**:

[22]:
```
#Usamos el pipe para asignarle otra funcion
flights %>%
    mutate(speed=distance/ air_time * 60, speed2 = speed*2)
```

year <int>	month <int>	day <int>	dep_time <int>	sched_dep_time <int>	dep_delay <dbl>	arr_time <int>	sched_
2013	1	1	517	515	2	830	819
2013	1	1	533	529	4	850	830
2013	1	1	542	540	2	923	850
2013	1	1	544	545	-1	1004	1022
2013	1	1	554	600	-6	812	837
2013	1	1	554	558	-4	740	728
2013	1	1	555	600	-5	913	854
2013	1	1	557	600	-3	709	723
2013	1	1	557	600	-3	838	846
2013	1	1	558	600	-2	753	745
2013	1	1	558	600	-2	849	851
2013	1	1	558	600	-2	853	856
2013	1	1	558	600	-2	924	917
2013	1	1	558	600	-2	923	937
2013	1	1	559	600	-1	941	910
2013	1	1	559	559	0	702	706
2013	1	1	559	600	-1	854	902
2013	1	1	600	600	0	851	858
2013	1	1	600	600	0	837	825
2013	1	1	601	600	1	844	850
2013	1	1	602	610	-8	812	820
2013	1	1	602	605	-3	821	805
2013	1	1	606	610	-4	858	910
2013	1	1	606	610	-4	837	845
2013	1	1	607	607	0	858	915
2013	1	1	608	600	8	807	735
2013	1	1	611	600	11	945	931
2013	1	1	613	610	3	925	921
2013	1	1	615	615	0	1039	1100
2013	1	1	615	615	0	833	842

A tibble: 336776 × 21

2013	9	30	2123	2125	-2	2223	2247
2013	9	30	2127	2129	-2	2314	2323
2013	9	30	2128	2130	-2	2328	2359
2013	9	30	2129	2059	30	2230	2232
2013	9	30	2131	2140	-9	2225	2255
2013	9	30	2140	2140	0	10	40
2013	9	30	2142	2129	13	2250	2239
2013	9	30	2145	2145	0	115	140
2013	9	30	2147	2137	10	30	27
2013	9	30	2149	2156	-7	2245	2308
2013	9	30	2150	2159	-9	2250	2306
2013	9	30	2159	1845	194	2344	2030
2013	9	30	2203	2205	-2	2339	2331
2013	9	30	2207	2140	27	2257	2250
2013	9	30	2211	2059	72	2339	2242
2013	9	30	2231	2245	-14	2335	2356
2013	9	30	2233	2113	80	112	30
2013	9	30	2235	2001	154	59	2249
2013	9	30	2237	2245	-8	2345	2353
2013	9	30	2240	2245	-5	2334	2351

0.2.10 `sumarize`

Sirve para aplicar **funciones a los renglones** de la base de datos, particularmente útil con `group_by` para agrupaciones.

[23]:
```
summarise(flights, delay = mean(dep_delay, na.rm = TRUE))
```

A tibble: 1 × 1

delay
<dbl>
12.63907

0.3 Agrupaciones

0.3.1 `Group by`

Los verbos por si solos ya responden preguntas, sin embargo si los juntamos con **agrupaciones** puede llegar a ser bastante interesante.

[24]:
```
flights %>%
    group_by(month, day) %>%
        summarise(delay = mean(dep_delay, na.rm = TRUE)) %>%
            arrange(desc(delay))
#Agrupamos --> funccion para delay --> ordenamos descendente
#Así tenemos el delay por mes y dia
```

```
`summarise()` regrouping output by 'month' (override with `.groups` argument)
```

month	day	delay
<int>	<int>	<dbl>
3	8	83.53692
7	1	56.23383
9	2	53.02955
7	10	52.86070
12	5	52.32799
5	23	51.14472
9	12	49.95875
6	28	48.82778
6	24	47.15742
7	22	46.66705
4	19	46.12783
6	13	45.79083
7	23	44.74169
6	30	44.18818
8	8	43.34995
5	8	43.21778
6	25	43.06303
6	27	40.89123
12	17	40.70560
8	28	40.52689
10	7	39.14671
2	11	39.07360
2	27	37.76327
7	28	37.71016
7	8	37.29665
7	7	36.61745
6	18	35.95077
4	18	34.91536
4	12	34.83843
12	9	34.80022
⋮	⋮	⋮
11	16	1.69620253
10	20	1.61092896
1	12	1.59649123
9	14	1.38888889
10	23	0.97940268
10	21	0.95445344
9	9	0.79474216
9	28	0.70250368
9	4	0.60233298
10	16	0.60206186
11	15	0.58697864
10	30	0.56553148
9	25	0.47412008
11	6	0.46376812
11	19	0.43904959
11	2	0.24486804
5	26	0.24142661
11	29	0.14523449
1	15	0.12372304
9	17	-0.09707724

A grouped_df: 365 × 3

Preguntas:

- ¿Hay algún día de la semana que sea considerablemente mejor para volar?

[48]:
```
flightsW <-flights %>%
               mutate(week_day = wday(mdy(paste0(month,'-',day,'-',year))))
```

year <int>	month <int>	day <int>	dep_time <int>	sched_dep_time <int>	dep_delay <dbl>	arr_time <int>	sched_
2013	1	1	517	515	2	830	819
2013	1	1	533	529	4	850	830
2013	1	1	542	540	2	923	850
2013	1	1	544	545	-1	1004	1022
2013	1	1	554	600	-6	812	837
2013	1	1	554	558	-4	740	728
2013	1	1	555	600	-5	913	854
2013	1	1	557	600	-3	709	723
2013	1	1	557	600	-3	838	846
2013	1	1	558	600	-2	753	745
2013	1	1	558	600	-2	849	851
2013	1	1	558	600	-2	853	856
2013	1	1	558	600	-2	924	917
2013	1	1	558	600	-2	923	937
2013	1	1	559	600	-1	941	910
2013	1	1	559	559	0	702	706
2013	1	1	559	600	-1	854	902
2013	1	1	600	600	0	851	858
2013	1	1	600	600	0	837	825
2013	1	1	601	600	1	844	850
2013	1	1	602	610	-8	812	820
2013	1	1	602	605	-3	821	805
2013	1	1	606	610	-4	858	910
2013	1	1	606	610	-4	837	845
2013	1	1	607	607	0	858	915
2013	1	1	608	600	8	807	735
2013	1	1	611	600	11	945	931
2013	1	1	613	610	3	925	921
2013	1	1	615	615	0	1039	1100
2013	1	1	615	615	0	833	842

A tibble: 336776 × 20

2013	9	30	2123	2125	-2	2223	2247
2013	9	30	2127	2129	-2	2314	2323
2013	9	30	2128	2130	-2	2328	2359
2013	9	30	2129	2059	30	2230	2232
2013	9	30	2131	2140	-9	2225	2255
2013	9	30	2140	2140	0	10	40
2013	9	30	2142	2129	13	2250	2239
2013	9	30	2145	2145	0	115	140
2013	9	30	2147	2137	10	30	27
2013	9	30	2149	2156	-7	2245	2308
2013	9	30	2150	2159	-9	2250	2306
2013	9	30	2159	1845	194	2344	2030
2013	9	30	2203	2205	-2	2339	2331
2013	9	30	2207	2140	27	2257	2250
2013	9	30	2211	2059	72	2339	2242
2013	9	30	2231	2245	-14	2335	2356
2013	9	30	2233	2113	80	112	30
2013	9	30	2235	2001	154	59	2249
2013	9	30	2237	2245	-8	2345	2353
2013	9	30	2240	2245	-5	2334	2351

[52]:
```
flightsW %>%
    group_by(week_day) %>%
        summarise(delay = mean(dep_delay, na.rm = TRUE)) %>%
            arrange(desc(delay))
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

A tibble: 7 × 2

week_day <dbl>	delay <dbl>
5	16.148920
2	14.778937
6	14.696057
4	11.803512
1	11.589532
3	10.631683
7	7.650502

- ¿Hay alguna aerolínea que tenga algún problema a nivel mes?

[55]:
```
flights %>%
    group_by(carrier, month) %>%
        summarise(delay = mean(dep_delay, na.rm =TRUE)) %>%
            arrange(desc(delay))
```

```
`summarise()` regrouping output by 'carrier' (override with `.groups` argument)
```

A grouped_df: 185 × 3

carrier <chr>	month <int>	delay <dbl>
OO	1	67.00000
OO	8	64.00000
OO	6	61.00000
HA	1	54.38710
YV	6	42.79487
FL	7	41.16270
FL	6	38.80658
F9	5	35.94828
VX	7	35.26337
YV	3	31.88889
F9	7	31.81034
9E	7	31.39883
WN	6	30.51473
F9	2	29.77083
F9	6	29.43636
9E	6	28.95298
VX	6	28.41250
EV	12	27.88718
YV	4	27.11111
EV	7	26.50472
EV	3	26.16982
FL	12	26.10577
EV	6	25.49683
B6	7	24.90232
WN	12	24.89479
F9	4	24.63158
WN	7	24.62817
EV	1	24.22888
FL	8	23.41016
EV	4	22.76755
⋮	⋮	⋮
AA	11	3.1020328
AS	11	3.0769231
AA	10	3.0022173
B6	10	2.9630649
AS	8	2.8709677
DL	11	2.8539121
US	3	2.7226942
AS	7	2.4193548
FL	1	1.9722222
US	9	1.9625830
US	1	1.8173633
HA	8	1.6774194
HA	6	1.4666667
HA	3	1.1612903
VX	1	1.0634921
US	2	0.9801642
OO	11	0.8000000
AS	2	0.7222222
AS	10	0.6774194
US	11	0.5761391

- ¿Hay algún avión problemático?

[56]:
```
flights %>%
    group_by(tailnum) %>%
        summarise(delay = mean(dep_delay, na.rm = TRUE)) %>%
            arrange(desc(delay))
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

tailnum	delay
<chr>	<dbl>
N844MH	297.00000
N922EV	274.00000
N587NW	272.00000
N911DA	268.00000
N851NW	233.00000
N654UA	227.00000
N928DN	203.00000
N7715E	186.00000
N665MQ	177.00000
N136DL	165.00000
N633AW	164.00000
N790SK	154.00000
N670US	132.00000
N427SW	131.00000
N305AS	112.50000
N78003	111.00000
N7ASAA	104.00000
N828AW	97.00000
N657UA	91.00000
N937DN	90.00000
N521SW	85.00000
N762SK	85.00000
N276AT	84.83333
N303AS	83.00000
N309AS	81.50000
N586AS	81.50000
N652SW	79.50000
N919FJ	78.00000
N162PQ	77.00000
A tibble: 4044 × 2 N7AEAA	74.00000
N627AW	-8.0
N693CA	-8.0
N907DA	-8.0
N941DN	-8.0
N287AT	-8.5
N518AS	-8.5
N585AS	-8.5
N789SK	-8.5
N509AA	-9.0
N632AW	-9.0
N661UA	-9.0
N778SK	-9.0
N913EV	-9.0
N926LR	-9.0
N583AS	-9.5
N14628	-10.0
N794SK	-10.0
N17627	-10.5
N701SK	-11.0
N726SK	-11.0

- ¿Hay alguna correlación entre distancia y retrasos?
- ¿Hay alguna correlación entre distancia y retrasos?