PEC - ENTORNOS DE ANÁLISIS DE DATOS R

Para la realización de esta prueba de evaluación continua he escogido un dataset que recoge durante 4 meses y 10 días vuelos internos que se produjeron con salida en cualquiera de los 3 aeropuertos de Nueva York. Debido a mi formación académica como ingeniero aeroespacial, siempre he tenido un gran interés hacia la industria aeronáutica, y me parece una buena oportunidad aprovechar esta prueba para poder analizar durante ese periodo de tiempo el tráfico aéreo de los aeropuertos de una de las ciudades más importantes del mundo. Mencionar que tanto las librarías usadas para la realización de los ejercicios como la url del dataset se encuentran en el fichero readme que se puede encontrar en el zip.

1. ¿Qué te ha parecido R y Tidyverse? ¿Te ha gustado?

Mi única experiencia en programación, y el único lenguaje que conozco a un nivel avanzado, es Python, por lo que mi opinión tras esta asignatura acerca de R está claramente influenciada por Python y por el hecho de no ser ni programador ni matemático.

El lenguaje R es muy útil para todo el mundo estadístico y matemático, ya que proporciona un amplio abanico de herramientas y una gran facilidad para el uso de vectores, aunque la sintaxis es un poco tediosa para mi gusto. Por lo tanto, entiendo su uso debido a la amplia variedad de posibilidades que ofrece, y a otra de las grandes ventajas que le he encontrado, que es la facilidad de visualización que ofrece R. Dentro del paquete tidyverse se encuentra ggplot2, un paquete que permite generar gráficos de alta calidad, de forma muy sencilla y que en otros lenguajes sería más difícil de realizar, o conllevaría un mayor número de sentencias. El resto de los paquetes de tidyverse que he llegado a utilizar son muy útiles a la hora de programar con R y facilitan la programación, pero no considero que marquen una diferencia sustancial con otros lenguajes de programación.

Por lo tanto, como conclusión final, me parece un lenguaje muy interesante y necesario para el mundo de la ciencia de datos porque es otra herramienta que puede llegar a ser muy útil para análisis rápidos dada la facilidad de visualización y el manejo de vectores que ofrece, y me ha gustado mucho dedicar una asignatura a aprender su uso, e incluso me hubiera gustado una asignatura previa para aprender la base, y haber utilizado esta asignatura para profundizar más en R orientado a la ciencia de datos.

2. Importa el dataset a R con readr, exporta un subconjunto del dataset consistente en las 5 columnas que más te gusten del mismo a formato csv. Interpreta estas columnas.

En el código, se importan las librerías readr y dplyr, ya que se va a hacer uso de read_csv, write_csv, pertenecientes a readr, y de select, que pertenece a dplyr. Para leer el dataset, es necesario utilizar la función read_csv pasando como argumento el path donde se encuentra el dataset, y para guardar el dataset en el mismo formato, se utiliza la función write_csv.

El dataset original se almacena en la variable *df*.

La función select permite hacer una selección de las variables generando un nuevo dataframe, por lo que es ideal para el ejercicio. Las 5 columnas escogidas son:

- *dep_delay* y *arr_delay*: estas dos variables representan en minutos el retraso del avión en la salida y en la llegada respectivamente. Si el valor aparece en negativo significa que se ha adelantado. Permite analizar si el avión ha tardado más o menos de lo planificado en hacer el recorrido.
- *air_time*: es el tiempo que el avión ha estado en el aire. Es una variable que siempre es positiva.
- *distance*: representa la distancia entre el origen y el destino. Es una variable que siempre es positiva, y que se espera que tenga una correlación positiva con la anterior.
- *origin*: es el aeropuerto de origen. Es una variable categórica con 3 posibles opciones que se corresponden con los 3 aeropuertos de Nueva York, y permite analizar el tráfico aéreo de salida de estos 3 aeropuertos y compararlo entre ellos.

El resultado se almacena en *df_reduced*.

3. Se deben elegir 5 variables reales o enteras y se debe calcular su media, mediana y moda. Filtra los valores ausentes.

Para el cálculo de la mediana y la media se puede hacer uso de las funciones mean() y median() filtrando los valores ausentes con na.rm = TRUE, y como R no dispone de ninguna función estándar que genere la moda, es necesario crearla. Para ello, filtramos los valores ausentes, calculamos las veces que aparece cada valor en la variable, y el valor que más veces aparezca es la moda. Las 5 variables escogidas son *arr_delay*, *dep_delay*, *distance*, *air_time* y *hour*.

Los resultados se almacenan en variables denominadas por el nombre seguido de media, mediana o moda en función del valor que recojan.

Para interpretar los resultados, hay que conocer que la media de un conjunto de datos es el promedio, resultado de sumar todos los números del conjunto y dividir por el número de valores. La mediana es el valor medio al ordenar los datos de menor a mayor, y la moda es el número que se presenta con mayor frecuencia en los datos.

En primer lugar, se puede analizar, por ejemplo, la variable *hour*, donde se observa que la hora a la que salen más aviones es de 8 a 9 AM. La mediana es 1PM, por lo que tiene sentido que, al tratarse de vuelos, haya vuelos a todas las horas del día, siendo el resultado una mediana igual a 13. Por último, si nos fijamos en la media, vemos que es muy similar a la mediana, lo que indica que la distribución de dicha variable debe ser bastante simétrica.

Las variables *arr_delay* y *dep_delay* se rigen por un mismo patrón. Grandes diferencias entre estas variables indicarían variaciones en el tiempo de vuelo estimado de forma muy frecuente, cuando, en general, el retraso aéreo se produce en el aeropuerto de salida, mientras que el tiempo de vuelo estimado suele ser superior al que realmente se necesita.

Se observa que la media y la mediana son bastante diferentes en ambas variables, lo que indica que las variables son asimétricas, siendo la media del retraso en la salida mayor que la media del retraso en la llegada, que es algo esperable porque los aviones suelen intentar recuperar tiempo en el trayecto en caso de que el avión tenga un gran retraso para evitar penalizaciones y porque el tiempo estimado de vuelo es superior al real, lo que permite que el retraso en la llega sea algo inferior. La moda en ambos casos es -1, lo que indica que hay más casos en los que el avión llega sin retraso, y que la media sea tan superior puede entenderse porque que el vuelo salga adelantado puede pasar, pero siempre serán pocos minutos, mientras que un retraso puede conllevar mucho tiempo, lo que hace que la media suba.

En las variables *air_time* y *distance*, al esperar que tengan una correlación positiva, tiene sentido que su media y mediana tengan un patrón similar, como así sucede. En ambos casos, la media es superior a la mediana, que puede suceder porque haya una serie de vuelos con una distancia lejana y que por lo tanto necesiten un tiempo mayor de vuelo, lo que afecta más a la media que a la mediana. La moda es diferente porque un mismo trayecto, por ejemplo, Nueva York – Los Ángeles, siempre se considera que tiene la misma distancia, porque esta no varía, pero el tiempo de vuelo si que puede verse modificado fácilmente en un mismo trayecto, por lo que tiene sentido que en este caso no sigan un patrón similar.

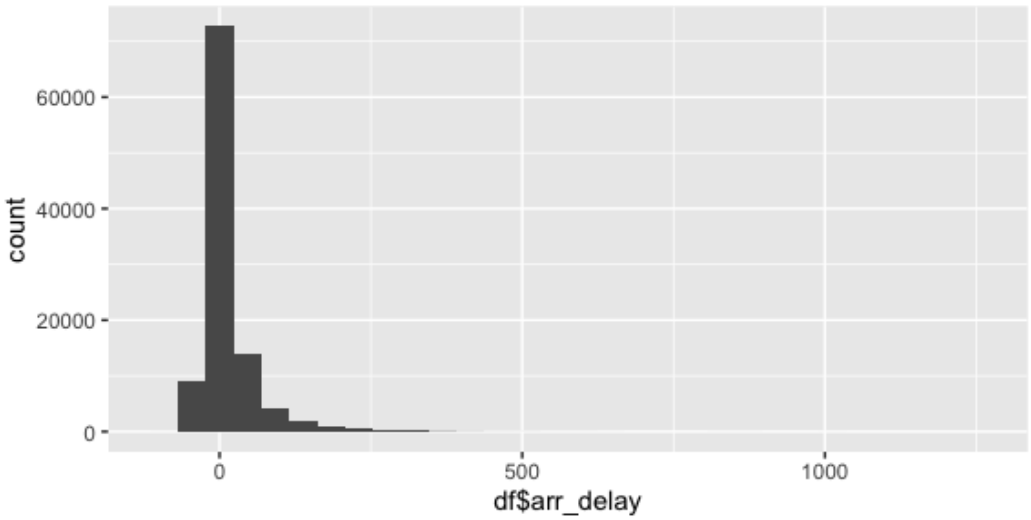
Finalmente, hay que comentar que es recomendable usar la media en distribuciones normarles, con cantidad baja de valores típicos, mientras que la mediana se utiliza para devolver la tendencia central en el caso de distribuciones numéricas sesgadas.

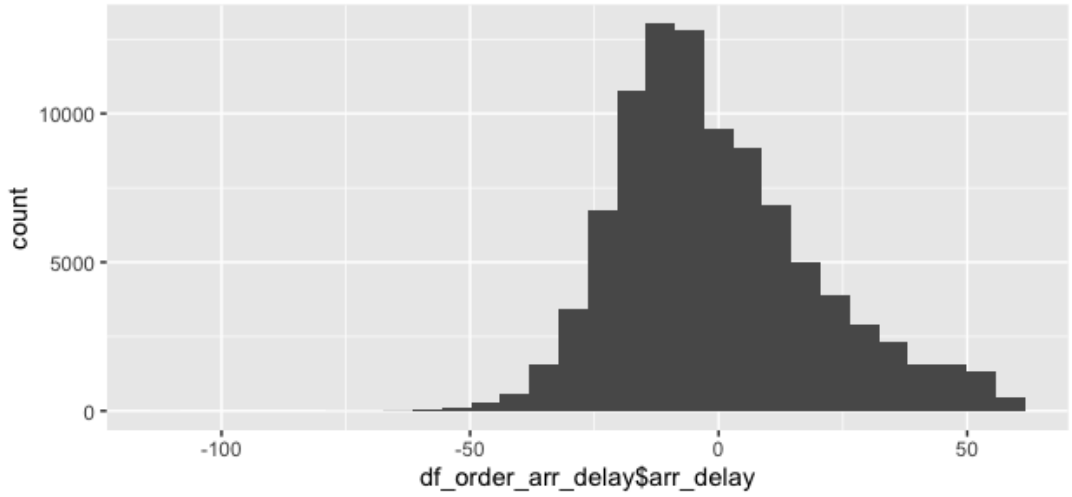
4. **Se debe ordenar los valores de una variable real y eliminar el 10% de valores más elevados y bajos (eliminación de outliers) donde esta operación pueda tener sentido.**

De forma adicional a lo pedido en el enunciado, para escoger la variable se hace uso de gráficas para identificar las variables con outliers que son más adecuadas para este ejercicio.

Como se pide ordenar a una variable, se hace uso de la función arrange del paquete dplyr, ya que este es el cometido de dicha función. Para eliminar los outliers, se ha decidido eliminar solo los más elevados ya que se consideran los más representativos, puesto que los más bajos son muy escasos. Para ello, se seleccionan las filas correspondientes al 90% del set de datos empezando desde la fila 1. El resultado se recoge en *df_order_arr_delay*.

Para comparar los resultados, se adjuntan los dos histogramas generados a partir de la librería ggplot2:

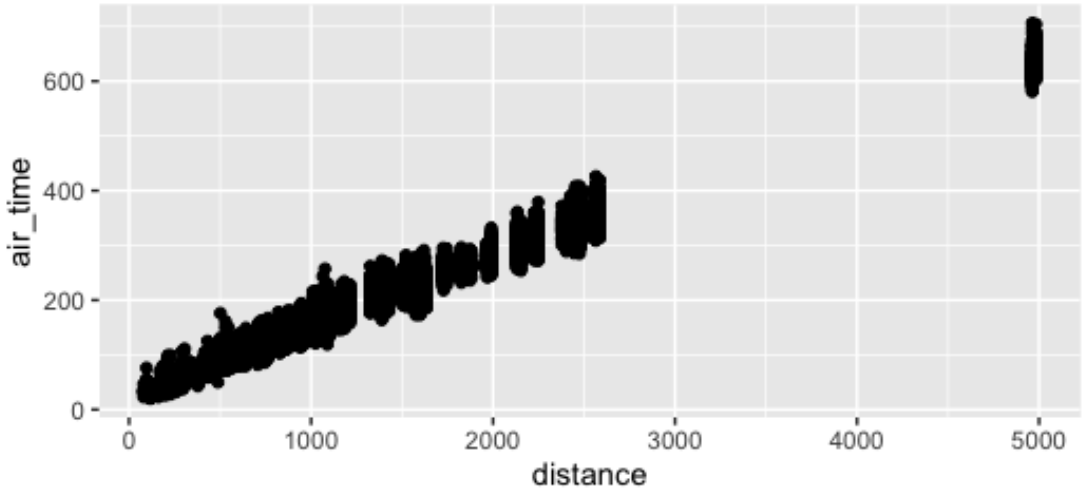




La variable escogida es *arr_delay*. Como se ha mencionado en el anterior ejercicio, el retraso de un vuelo puede suponer grandes diferencias con la media debido a valores elevados, pero no es muy frecuente que un vuelo se adelante mucho tiempo, por lo que solo se ha decidido eliminar el 10% de los outliers más elevados. Se puede apreciar como la eliminación de outliers modifica la representación de los datos en un histograma, aportando el segundo más información acerca de los valores más comunes.

5. **Se debe adjuntar un gráfico de puntos de dos variables reales donde se pueda apreciar una correlación entre ambas (a mayor valor de una variable, menor de otra o viceversa). Sino existen 2 variables así en tu dataset, comenta que observas en el gráfico con 2 variables cualquiera.**

Mediante el uso de la librería ggplot2, se puede crear un diagrama de puntos de una forma muy simple gracias al uso de geom_point. Como ya se ha comentado anteriormente, se espera que las variables *distance* y *air_time* tengan una correlación positiva, por lo que he escogido dichas variables para este ejercicio, siendo el resultado:



Como era de esperar, se muestra una correlación positiva, que se entiende de forma que cuando una variable aumenta, la otra también, algo fácilmente apreciable en el diagrama. Como observación curiosa, se muestra que a partir de una distancia entorno a 2600 no hay ningún vuelo hasta una distancia cercana a los 5000. Estos valores cercanos a los 5000 son los que provocan que la media sea superior a la mediana en estas variables.

6. **Crea una variable adicional con respecto a las que tenga tu dataset que sea el resultado de dividir una variable real entre otra variable real.**

Se genera la variable *speed* a partir de la división de las variables *distance* y *air_time*. El resultado final es que se modifica *df* añadiendo la variable speed al final.

7. **Filtra los datos de tu dataset para solo tener las instancias correspondientes a los meses de enero, febrero y marzo. Si tu dataset no tiene fechas, crea una variable adicional fecha que ponga una fecha aleatoria para cada instancia y filtra tu dataset por la fecha.**

En el dataframe los meses aparecen de forma numérica por lo que se hace uso de la función mapvalues del paquete plyr para crear una nueva variable, denominada *new_month*, con los meses escritos tal que Enero, Febrero, Marzo, Abril y Mayo, que será de utilidad en siguientes ejercicios y que se puede usar para resolver este. Por lo tanto, se modifica *df* añadiendo al final de este la variable *new_month*.

Para la resolución del ejercicio se utiliza la función filter del paquete dplyr aplicando un filtro de forma que solo obtenemos los valores de la variable new month que se encuentran en el vector especificado. El resultado se recoge en *df_ene_feb_mar*.

8. **Devuelve todas las instancias que cumplan que una variable alfanumérica tenga la letra 'a'. Si tu dataset no tiene una variable alfanumérica, genera una variable cuyos valores alfanuméricos sean letras ('a',…,'z') aleatorias y devuelve solo las instancias cuyo valor para la variable alfanumérica sea 'a'.**

En este caso, con una sentencia se puede llevar a cabo el cometido del ejercicio. Es necesario destacar que se está haciendo uso de la variable generada en el ejercicio anterior con los meses de enero a mayo.

Se hace uso de la función grepl, que recibe como argumentos el patrón a buscar, en este caso la letra 'a', y la columna de la variable en cuestión, aunque se transforma todo a minúsculas, ya que, por ejemplo, el mes Abril no se detectaría si no se hiciera este cambio.

Finalmente, hay que destacar que la función grepl genera un vector de tipo lógico de forma que, si la letra a no aparece en el valor, se devuelve False, y si aparece se devuelve True, y con este vector se puede filtrar el dataframe inicial obteniendo el resultado deseado, que se almacena en *df_a*.

9. **Ordena una variable real y devuelve las 10 instancias con mayor valor para esa variable.**

De nuevo, se hace uso de la función arrange para ordenar un dataframe, con la diferencia de que se hace de forma descendente en la columna escogida. Una vez ordenado, simplemente se guardan los 10 primeros valores, que son los 10 más elevados, en una variable, denominada, *order_arr_delay_desc*, y se imprime por pantalla.

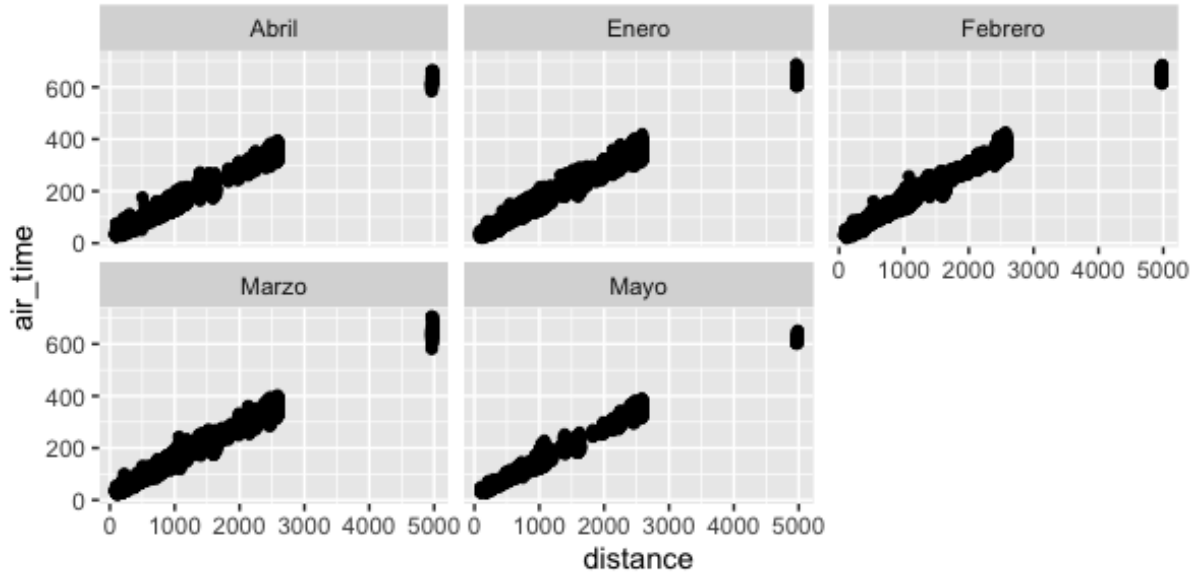
10. **A partir de una variable real, haz una variable categórica factor cuyo valor sea A si el valor es superior a la mediana y B si el valor es inferior a la mediana.**

Para la realización de este ejercicio, se ha escogido la variable *distance*, cuya mediana está almacenada en la variable *mediana_distance*. Aplicamos una estructura if else de forma que el valor de la nueva variable, *factor_variable*, sea A en caso de que el valor de distance sea superior a la mediana, o B en caso contrario, es decir, inferior o igual a la mediana.

Se añade la nueva variable al final de *df*.

11. Imprime un gráfico de facetas en rejilla cuyas gráficas sean gráficas de puntos que muestre los datos de 2 variables real y una categórica.

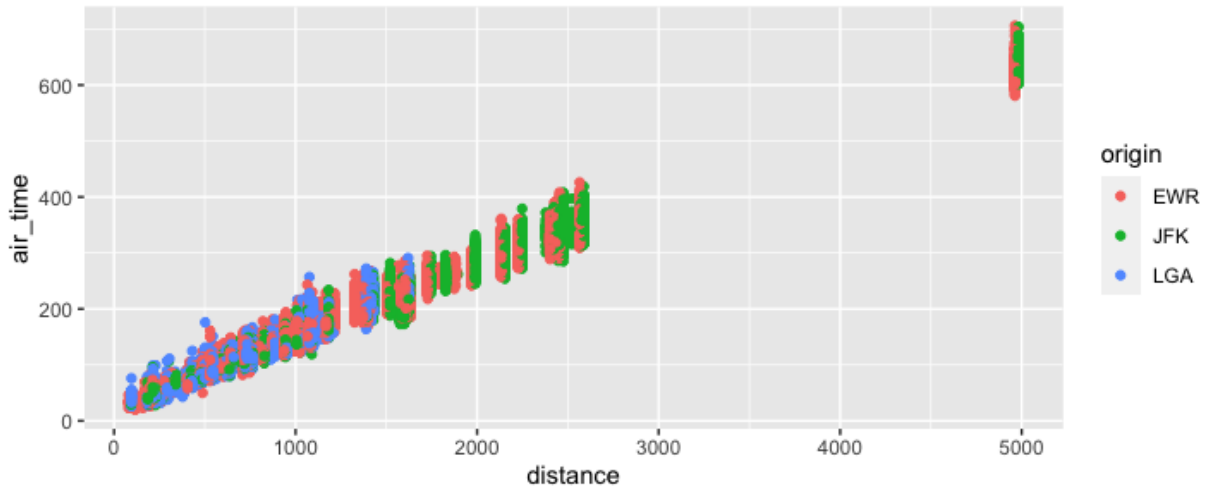
Las dos variables reales escogidas son *air_time* y *distance*, mientras que la variable categórica es *new_month*, cuyos valores son Enero, Febrero, Marzo, Abril y Mayo.



Para la realización de un gráfico de facetas es necesario añadir facet_wrap con la variable categórica en cuestión. En cuanto al resultado, se observa como la correlación entre ambas variables es similar en todos los meses, pero en el mes de mayo, se aprecia un menor número de vuelos, lo cual tiene sentido porque el dataframe recoge los datos de todos los meses completos, excepto de mayo, cuyos datos alcanzan hasta el día 11 de dicho mes.

12. Se debe adjuntar un gráfico de puntos de 2 variables reales en el cuál se muestren con colores los valores de una variable categórica o factor o una variable alfanumérica.

Para la realización de este ejercicio se han vuelto a elegir las variables *air_time* y *distance* para el gráfico de puntos, y la variable categórica que se ha escogido para el color es el origen, que se corresponde con los tres aeropuertos de Nueva York, que son EWR, JFK y LGA.



Se puede apreciar que los aeropuertos JFK y EWR disponen de vuelos de todas las distancias, aunque su presencia aumenta a partir de la distancia 1500, mientras que el aeropuerto LGA alcanza hasta una distancia de 1700. Buscando información en internet, tiene sentido puesto que el aeropuerto LGA se trata del principal aeropuerto doméstico de la ciudad debido a su localización centralizada y su proximidad a Manhattan, pero no dispone de aduanas, ni servicios de inmigración, y no se permiten vuelos de entrada ni de salida que excedan los 2400 km. Esta información explica que no haya vuelos correspondientes a dicho aeropuerto para distancias elevadas, y que en distancias cortas sea el más representativo, puesto que es principal aeropuerto doméstico.

13. Se debe calcular la media y desviación típica de los 10 valores más altos de una variable real filtrada por un valor de una variable categórica de un día de cada mes a elegir por el alumno de una variable que represente una fecha.

En el dataset escogido, la fecha está dividida en tres columnas, year, month y day. Se ha decidido filtrar por el valor day igual a 1, de forma que obtenemos todos los vuelos del primer día de cada mes. La variable real escogida es *air_time*, y a la hora de seleccionar los 10 valores más altos, se observa que el décimo tiene una gran diferencia con los otros 9, por lo que se ha decidido calcular también la media y desviación estándar de los 9 primeros valores, sin incluir el décimo, para observar como afecta este al resultado.

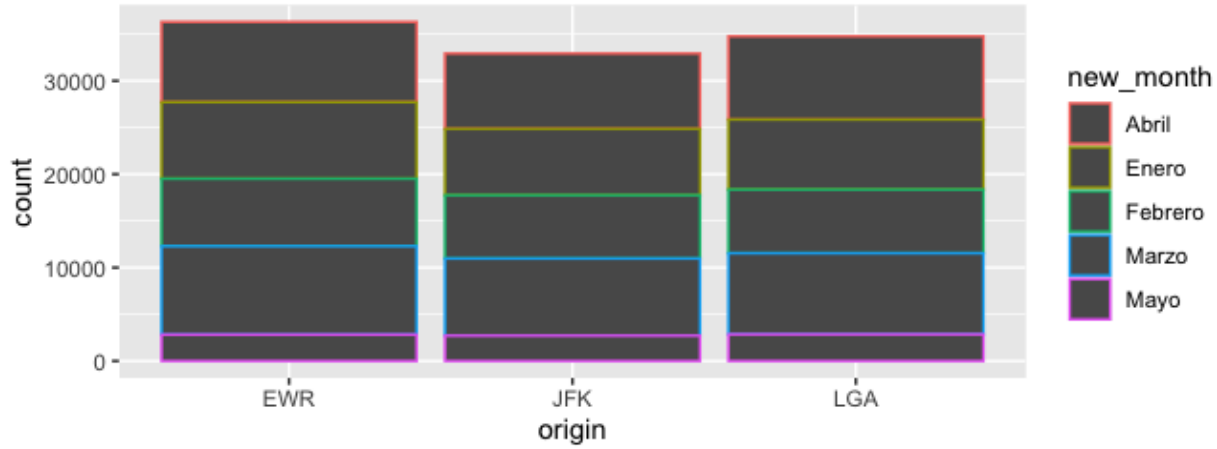
El calculo tanto de la media como de la desviación estándar se hacen con las funciones mean() y sd(), y se recoge el resultado en *media_air_time_10*, *media_air_time_9*, *sd_air_time_10* y *sd_air_time_9*.

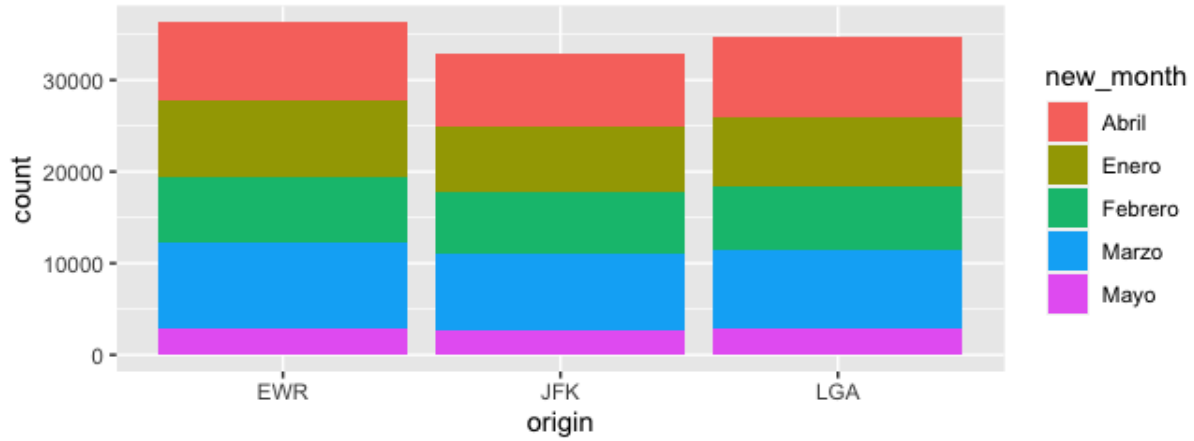
Para interpretar los resultados, es necesario conocer que la desviación estándar es una medida de dispersión que indica qué tan dispersos están los datos con respecto a la media. Cuanto más cercano a 0 sea dicho valor, menos dispersión habrá en el conjunto de datos.

Comparando los resultados, se aprecia que el dato correspondiente a la décima fila genera una gran dispersión, ya que hace aumentar en gran medida la desviación estándar, mientras que la variación de la media no es tan grande.

14. Se debe generar un gráfico de barras similar al mostrado en la primera figura del apartado 3.8 del libro de Data Science de las variables de tu dataset que prefieras.

Para generar el gráfico correspondiente, se hace uso de las variables *origin* para el eje x y de *new_month* para el color, con el objetivo de contabilizar el número de vuelos que salen de cada aeropuerto en cada mes.





Las gráficas nos permiten observar el menor número de vuelos en el mes de mayo, pero como ya se ha comentado, se debe a la falta de datos en dicho mes. Se puede comprobar que el aeropuerto con más vuelos durante este periodo de tiempo en este dataset fue EWR, y la distribución por meses en cada mes permite observar que, en todos los aeropuertos, tanto marzo como abril tienen un número de vuelos mayor, lo que tiene sentido puesto que enero y febrero son meses de temporadas más bajas para el tráfico aéreo.

15. Describe para que sirven. los pipes en R y pon un ejemplo con tu conjunto de datos en el que uses, para una operación, 3 pipes consecutivos.

Los pipes son una herramienta para expresar claramente una secuencia de múltiples operaciones. El pipe se expresa mediante la simbología *%>%* y se carga del paquete magrittr, que también está incluido en tidyverse. Permite expresar de forma clara una secuencia de operaciones de forma que el operador pipe coge la salida de una sentencia de código y la convierte en el argumento de una nueva sentencia.

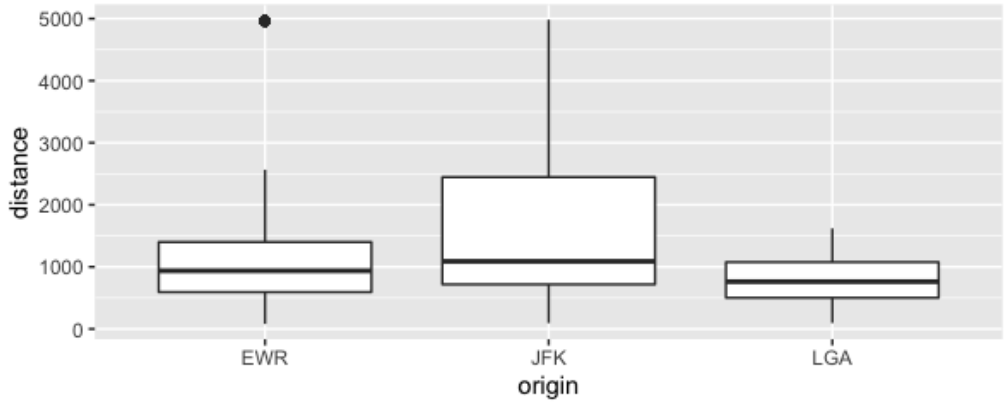
Para este ejercicio, se pretende utilizar los pipes para averiguar los 10 días de todos los recogidos en los cuales se realizó una cantidad mayor de vuelos, y almacenando el resultado en *df_day_freq*.

Para ello, el primer elemento es el dataframe inicial que sirve de entrada a la siguiente sentencia, donde se pretende juntar las columnas year, month y day en una sola, para lo cuál se usan conjuntamente mutate, para transformar el dataframe, y make_date del paquete lubridate para generar la fecha a partir de las columnas anteriores. De esta forma tenemos de entrada para la siguiente sentencia el mismo dataframe con la columna añadida, y ahora se busca obtener las veces que aparece cada fecha en dicha columna, cuyo resultado se obtiene con count del paquete dplyr. Finalmente, se escogen las 10 primeras filas, y tenemos nuestro resultado.

Analizando el resultado, de todas las fechas, las 7 primeras coincidieron con Lunes o Jueves, y las tres últimas en un Martes, lo que podría indicar que son viajes de trabajo, siendo este uno de los motivos del gran tráfico aéreo interno de los aeropuertos de Nueva York.

16. Haz un gráfico de cajas de una variable real y otra categórica.

Para la realización del gráfico se ha escogido como variable categórica el origen, es decir, los tres aeropuertos de la ciudad de Nueva York, y como variable real la distancia recorrida en los vuelos que salieron de dichos aeropuertos.



En primer lugar, es necesario describir un diagrama de cajas para poder interpretarlo. Las líneas superior e inferior que delimitan la caja representan los cuartiles Q1 y Q3 respectivamente mientras que la línea gruesa de su interior representa el Q2, que es la mediana. Los cuartiles se calcula de forma que el Q1 representa hasta el 25% de la muestra, el Q2 hasta el 50% y el q· hasta el 75%, de forma que la caja en sí representa el 50%.

Para entender las líneas que se sitúan fuera de la caja es necesario conocer que el rango intercuartil es la distancia entre Q1 y Q3, que los valores atípicos leves son los que se sitúan

fuera de la caja, pero en el intervalo *Q1 – 1,5*Rango Intercuartil* o *Q3 + 1,5*Rango Intercuartil*, mientras que, si los valores se encuentran fuera de dicho rango, se consideran valores atípicos.

Una vez comentadas las características del diagrama de cajas, analizando el gráfico, se pueden sacar conclusiones que o bien se esperaban, o que ya se habían comprobado con otros gráficos ya realizados. Por ejemplo, el aeropuerto LGA no recoge datos a partir de una distancia de 1700 y vemos como el rango de datos que abarca es inferior al del resto de aeropuertos, sin incluir valores atípicos y con una distribución de vuelos por distancia similar, que cada 25% de la muestra ocupa un rango de distancia parecido.

En el aeropuerto EWR vemos como la mayoría de los vuelos se produce en un rango de distancias pequeños, lo que hace que aparezcan los outliers en la distancia de 5000, que se encuentra muy alejada de la media. El aeropuerto JFK, por ejemplo, nos ofrece una mediana similar a la del EWR. Se observa como una gran cantidad de vuelos se encuentra en un rango muy pequeño de distancias que hace que la mediana sea baja, pero el otro 50% no está concentrado en una distancia cercana a la mediana lo que hace que la parte superior de la caja sea más grande, siendo el Q3 bastante superior, y no generando outliers.

17. Devuelve la moda de letras 'a' de una variable alfanumérica de todas sus instancias.

Para calcular la moda de letras 'a' de una variable se ha escogido new_month. Se hace uso de la función get_mode que se había calculado para el ejercicio 3, la cual calculaba la moda de un vector numérico. Por lo tanto, es necesario calcular dicho vector para la cual se hace uso de la función str_count, la cual recibe como parámetros la columna en cuestión transformada a minúsculas y el patrón a buscar que es la letra a.

El resultado final se recoge en *month_mode_a*, y su valor es 1, es decir, los valores de la columna *new_month* contienen de forma más frecuente 1 vez la letra a que ninguna vez, o que dos veces, por ejemplo. Al conocer los valores de dicha variable ya que es categórica, y son Enero, Febrero, Marzo, Abril y Mayo, el resultado es el esperado ya que 3 de los 5 meses contienen la letra a, y aunque el número de datos del mes de mano es menor, también sabemos que los meses de marzo y abril son los que contienen un mayor número de vuelos, por lo que el resultado tiene sentido.

18. Crea una columna que contenga un periodo de tiempo en semanas entre dos columnas previas. Sino dispones de esta información en tu dataset, crea columnas sintéticamente.

Para la realización del ejercicio, es necesario crear las columnas con las fechas. La primera, se crea a partir de las columnas year, month y day, uniéndolas y aplicando el formato ymd y de tipo fecha, que se añade a *df* con el nombre *date*. La segunda columna se crea de forma aleatoria con valores entre el 1 de enero de 2015 y el 1 de enero de 2016, de forma que cada registro puede tener un valor igual a cualquiera de esos 365 días disponibles. Esta segunda columna se añade a df con el nombre *date_random*.

Para la diferencia entre columnas se hace uso de difftime especificando que la unidad es weeks, y mediante select se modifica el orden de las columnas de forma que obtenemos el resultado deseado, tal que se añade la diferencia en la columna *diff_in_weeks*.

Respecto a la variable *diff_in_weeks* que representa la diferencia en semanas entre ambas variables, es necesario mencionar que la resta se calcula de forma decimal, es decir, que la parte entera son las semanas completas, mientras que la parte decimal son los días que no llegan a completar una semana total.

19. Devuelve el sumatorio de los valores de una variable real agrupados por una variable categórica o por mes de una variable de fecha.

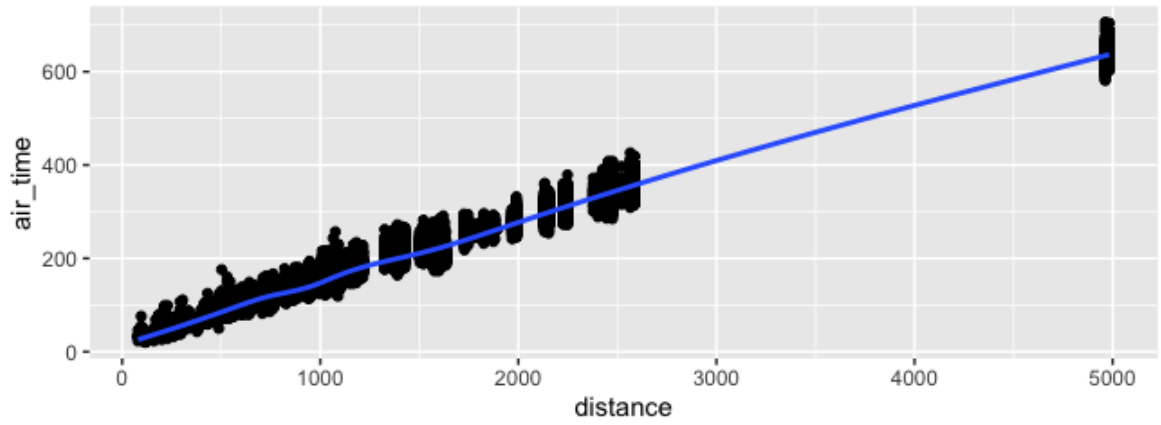
Para la variable categórica se ha escogido los meses recogidos en la variable *new_month*, y la variable real es *air_time*, de forma que se devuelve el tiempo de vuelo empleado en cada mes.

Para ello, se hace uso de la función aggregate a la cual hay que especificar la variable que se va a modificar, la función que se va a aplicar a dicha variable y en función de que variable se va a agrupar. Finalmente se ordena el resultado de mayor a menor y se recoge en la variable *df_sum_air_time*. Se observa que el mes donde se emplea más tiempo de vuelo es marzo, seguido de abril, y que el mes completo con menos tiempo de vuelo es febrero.

20. ¿Podrías predecir una variable real en base a otras variables de tu dataset? Razona como lo harías.

Para predecir una variable en función de otra podemos fijarnos en el ejercicio 5, donde se demuestra que las *distance* y *air_time* están correlacionadas de forma positiva, por lo que sabiendo como es la correlación podemos predecir el valor de una variable a partir de la otra.

Por ejemplo, con los conocimientos vistos hasta ahora en R, se podría generar un gráfico como el siguiente:



En dicho gráfico, hemos generado un diagrama de puntos idéntico al del ejercicio 5 añadiendo un diagrama geom_smooth() que te permite calcular una curva que se ajusta a los datos. Crea una tendencia de los datos. En este caso, usa el modelo gam para generar dicha curva debido

a que el número de registros del dataframe se puede considerar elevado para hacer uso de otros métodos como podría ser loess. De esta forma, a partir de un valor de air_time o de distance, podemos obtener el valor de la otra variable a partir de la curva generada.

Otra forma de hacer una predicción, aunque no vista en clase, es usar la función *lm* que es usada para ajustar modelos lineales. En este caso, generaría un simple modelo de regresión lineal entre las variables distance y air_time, que se almacena en model, cuyo resultado sería:

```
Call:
lm(formula = air_time ~ distance, data = df)

Coefficients:
(Intercept)     distance
      19.12         0.13
```

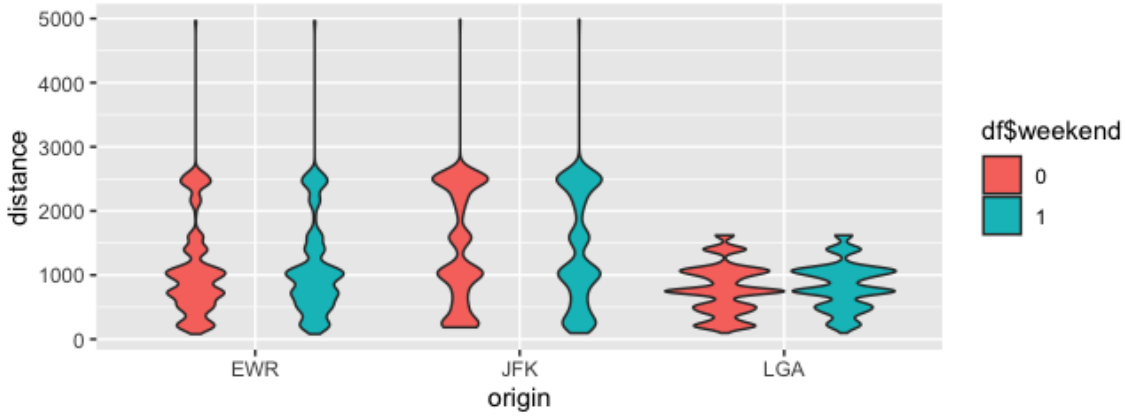
El modelo genera la ecuación:

$$air\_time = \ 19.12 \ + distance * 0.13$$

De esta forma, haciendo uso de la función predict, podemos pasar como argumentos el modelo, y el nuevo dato, que serían las distancias, para predecir el tiempo de vuelo.

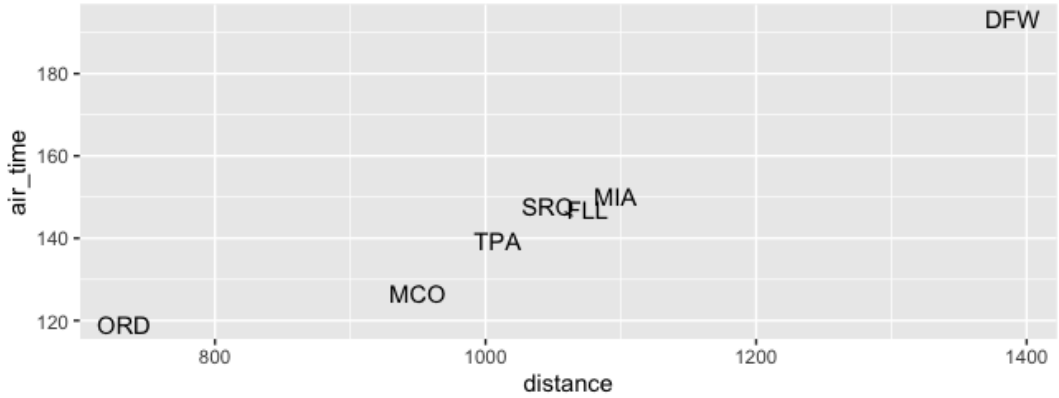
21. Realiza dos visualizaciones no vistas en clase que involucren a un mínimo de 3 variables. Consulta la cheatsheet de R de ggplot2. Interprétalas.

En la primera visualización se pretende analizar la distribución de los vuelos en cada aeropuerto en función de la distancia, y dependiendo de si los vuelos se realizan en fin de semana o no. Para ello, se añade una nueva variable a *df* denominada *weekday* que indica el día de la semana correspondiente y otra variable denominada *weekend* de forma que el valor es igual a No o a Si en función de si es fin de semana o no. Una vez calculada la variable, se puede generar el gráfico:



Se ha escogido el diagrama de violín, donde las figuras representan la distribución de los vuelos en cada aeropuerto en función de la distancia. Es un diagrama similar al de cajas, pero ofrece más detalle en cuanto a la distribución de valores. Se puede apreciar como en los tres casos, las figuras son bastante similares para los días entre semana y para el fin de semana, aunque las figuras que representan los días entre semana son más anchas que las del fin de semana lo que evidencia un mayor número de vuelos en estos días. Por lo demás, considero que la distribución es bastante similar, y no se puede obtener ninguna diferencia clara entre la distancia realizada entre semana con respecto a los sábados y domingos.

Para el segundo gráfico, se ha escogido un gráfico de texto que muestra los destinos de los vuelos que salieron el 11 de mayo desde el aeropuerto LGA. Para la realización del dataframe a partir del cual se genera el gráfico se han realizado 5 pipes, por lo que se comentará en el ejercicio 24. El gráfico generado es el siguiente:



En el gráfico se muestra el destino en formato texto en función de la distancia y del tiempo de vuelo. Se ha generado un poco de ruido aleatorio para que se pueda observar mejor el gráfico. También se ha decidido hacer uso de la fecha 11 mayo y exclusivamente el aeropuerto LGA porque el número de vuelos es menor y permite una mejor visualización de los datos.

Se observa que el vuelo más corto que se realizó fue al aeropuerto ORD, que se encuentra en Chicago, relativamente cerca de Nueva York, mientras que el más lejano fue al aeropuerto de Dallas.

22. Devuelve un resumen de estadísticos de las variables que consideras más importantes e interprétalo.

Para la obtención del resumen de estadísticos de las variables se ha usado el paquete fBasics, que contiene la función basicStats que devuelve el siguiente resumen estadístico al pasar como argumento un dataframe. El resumen se recoge en la variable *stats* y se muestra a continuación:

	dep_delay	arr_delay	air_time	distance	hour
Mean	14.42526	11.85486	161.16327	1092.7308	13.172755
Stdev	44.70504	48.96861	96.33674	734.5159	4.894809
Variance	1998.54074	2397.92511	9280.76674	539513.6277	23.959155
1. Quartile	-5.00000	-13.00000	92.00000	541.0000	9.000000
Median	-1.00000	-1.00000	141.00000	950.0000	13.000000
3. Quartile	13.00000	18.00000	202.00000	1411.0000	17.000000
Minimum	-112.00000	-112.00000	20.00000	80.0000	0.000000
Maximum	1241.00000	1223.00000	706.00000	4983.0000	24.000000

En el ejercicio 3 ya se comentaron los valores de media, mediana y moda para las cinco variables que se han incluido, por lo que se van a comentar otros resultados estadísticos.

En el caso de las variables *dep_delay* y *arr_delay* vemos la principal diferencia en la distribución de los valores, ya que los cuartiles Q1 y Q3 son claramente diferentes, estando la variable *dep_delay* más concentrada en unos valores cercanos a la mediana que la variable *arr_delay*. Que la desviación estándar sea superior en el caso de *arr_delay* nos indica que los valores de toda la variable son más dispersos que los de la variable *dep_delay*. Por lo tanto, podemos concluir que la variable *dep_delay* contiene más retrasos cercanos al valor medio que la variable *arr_delay*, por lo que la dispersión es menor.

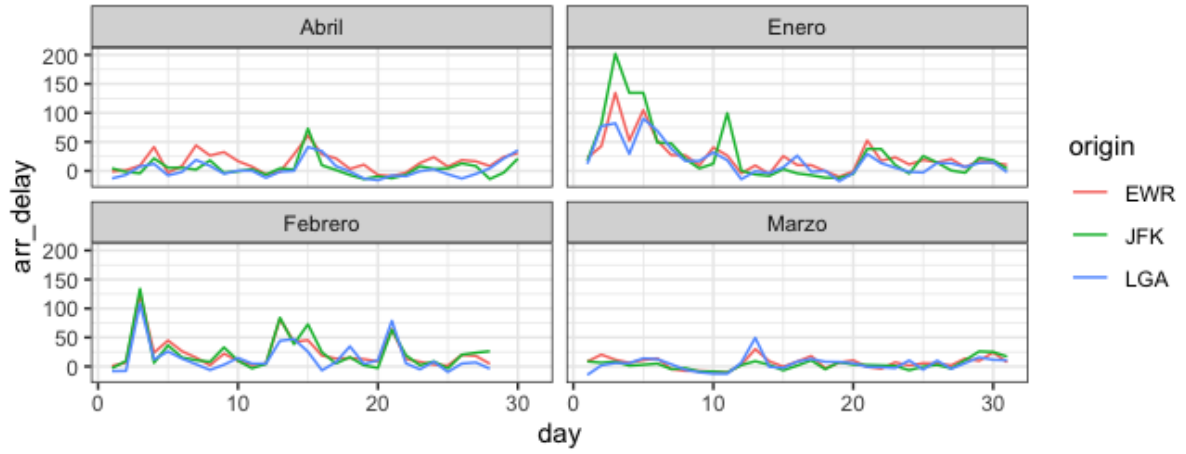
Las variables *air_time* y *distance* siguen un patrón similar con los cuartiles, donde encontramos que hay más vuelos concentrados entre Q2 y Q1 que entre Q3 y Q2. Al ver el máximo, parece que el valor mostrado por la dispersión podría ser muy bajo, ya que la diferencia entre el máximo y la media es muy alta, pero la dispersión no es tan alta en ambos casos. Si recordamos de otros gráficos, los vuelos llegaban hasta una distancia de 2750 más o menos, y no había más vuelos hasta una distancia cercana a los 5000 que se refleja en el máximo de la variable *distance* con el valor 4983. Una dispersión baja a pesar de esta diferencia entre máximo y media indica que los valores cercanos a la distancia 5000 no deben de ser muchos por lo que no se genera tanta dispersión en la variable. Para comprobarlo podemos filtrar por los valores superiores a una distancia 4000 y comprobaremos cuantos existen. El resultado se observa en la variable *df_distance_5000* y se comprueba que son solo 232 registros, que comparados con los 103.997 que componen el dataset, tiene sentido que no generen una gran dispersión. Como curiosidad, ya se ha comentado que el dataset recoge solo vuelos internos, y el destino que se corresponde con los 232 registros de una distancia tan elevada es el aeropuerto de Honolulu en el estado de Hawái.

Finalmente, para comparar los valores de la variable *hour*, si nos fijamos en los cuartiles, vemos de nuevo que la diferencia entre ellos es idéntica, lo que de nuevo refuerza la idea de una variable bastante simétrica que habíamos comentado al principio debido a la similitud de los valores de la media y la mediana. En cuanto a la desviación estándar cercana a 5 es un valor razonable teniendo en cuenta que el 50% de los valores se encuentra en un rango superior o inferior de 4 horas a la media, ya que su valor es similar al de la mediana.

23. Realiza una visualización no vista en clase y distinta de la del apartado 21 que involucre un mínimo de 4 variables. Interprétala.

Para la realización de la visualización se ha escogido un gráfico de líneas con la idea de representar para cada día de cada mes en cada aeropuerto la media del retraso de los vuelos en la llegada.

Para su elaboración, se ha generado un nuevo dataframe denominado *df_delay*, en el que excluimos el mes de mayo ya que no está completo. Se escogen las 4 variables mencionadas y se agrupa por mes, día y aeropuerto, de forma que obtenemos la media del retraso que se produjo en cada aeropuerto para todos los días. Para la realización del gráfico se hace uso de geom_line, de facet_wrap ya comentada anteriormente y se le añade theme_bw para mejorar la visualización del gráfico de líneas.



El gráfico permite observar la evolución del retraso medio en cada mes de una forma muy simple. El principal motivo de los retrasos en los aeropuertos es la meteorología, ya que afecta a todos los vuelos, por lo que grandes picos de retrasos en determinados días se pueden deber a condiciones climatológicas muy adversas como se observa principalmente en enero y en febrero, que suele haber más nevadas. Otro tipo de retrasos se pueden deber a problemas mecánicos, pero no afecta a todo el aeropuerto, por lo que la media no se muy alterada. Otro de los motivos que puede conllevar retrasos es la alta afluencia en determinados días, como sucede en salidas o vuelta de vacaciones, y puede suceder con el pico que se observa a principios de enero, que coincide con la vuelta de Navidades.

La diferencia entre los retrasos de cada aeropuerto se debe al cometido de estos, debido que aeropuertos destinados a un mayor número de vuelos y de mayor distancia como el JFK pueden sufrir más retrasos.

24. Haz un ejemplo que involucre 5 pipes seguidos. Interprétalo.

Como se ha comentado en el ejercicio 21, el ejemplo que involucra 5 pipes se ha realizado para obtener los vuelos correspondientes al día 11 de mayo en el aeropuerto LGA, y que posteriormente se ha utilizado para el gráfico de dicho ejercicio.

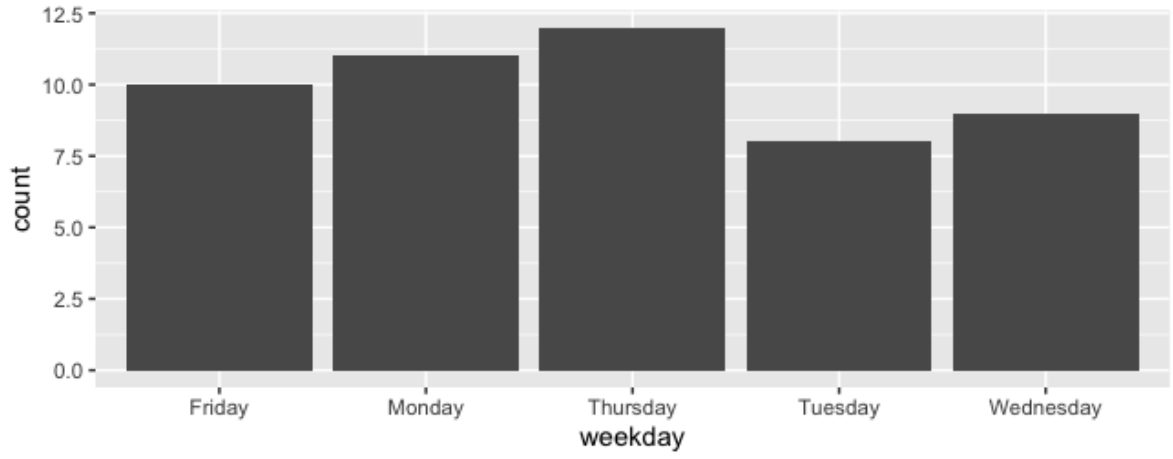
El dataframe se almacena en df_may_11 y el primer pipe se corresponde con el dataframe original en el cual se filtra para obtener los valores deseados que son el 11 de mayo y el aeropuerto LGA. Se hace una transformación de la variable factor_variable para tener valores numéricos, representando si el registro es superior o inferior a la mediana de la distancia. La transformación es necesaria para poder usar la función summarise posteriormente.

Tras las transformaciones, escogemos las variables deseadas, agrupamos por el destino con la media de los valores, con un resultado final que recoge la media tiempo de vuelo empleado en los trayectos a cada aeropuerto, la distancia entre el aeropuerto LGA y el destino, y si dicha distancia es superior a la mediana o no.

25. En función a tu conjunto de datos, presenta las 5 conclusiones que, en tu opinión, son mas relevantes y pueden aportar mas información del conjunto de datos que has elegido. Apoya tu decisión con gráficos, análisis univariantes y multivariantes de variables y con otras técnicas que conozcas y que consideres necesarias. No dudes en emplear conocimiento propio no visto durante el curso y que pueda encontrarse en el libro R for Data Science o que conozcas previamente.

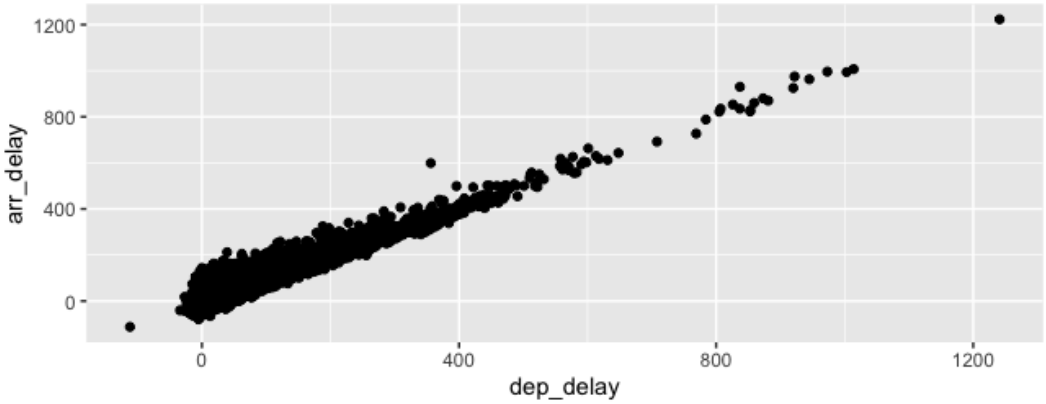
Debido al análisis llevado a cabo para la realización del resto de ejercicios, muchas de las conclusiones vienen derivadas de resultados ya obtenidos.

- La primera conclusión que considero relevante está relacionada con el día de la semana en el que se producen más vuelos sumando todos los aeropuertos de Nueva York. Como se muestra en la siguiente gráfica, entre los 50 días donde se produjeron un mayor número de vuelos, no se encuentra ningún domingo ni sábado, sorprendiendo más el primero, ya que es un día donde se suele esperar que haya una gran afluencia de vuelos.



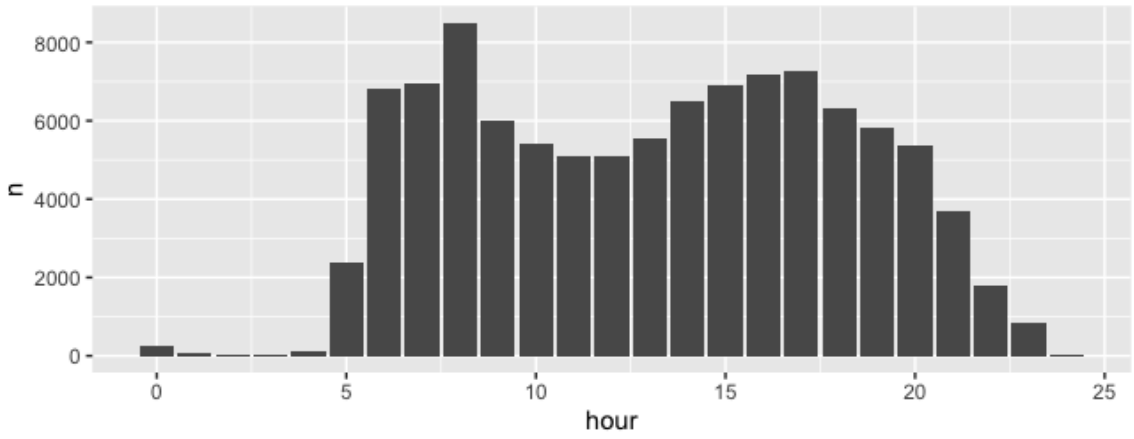
- La segunda conclusión que obtengo es la correlación entre las variables distance y air_time que se ha mostrado ya en más de un ejercicio. Aunque esta conclusión pueda resultar bastante obvia, me parece que una de las funciones principales de un análisis exploratorio es la búsqueda de variables que estén relacionadas entre sí, porque pueden aportar mucha información ya sea para predecir variables o para eliminar variables si se consideran redundantes. Aparte de los gráficos ya mostrados, se ha calculado el valor de la correlación a partir de la función cor y se ha almacenado en *corr_dist_air_time*. El resultado es de 0.99, lo que significa casi una correlación positiva perfecta, cuyo valor sería 1. Un valor 0 indicaría que no existe correlación, y un valor -1 que es una correlación negativa perfecta.

- La tercera conclusión es de nueva una correlación entre dos variables, en este caso, el retraso en la salida respecto al retraso en la llegada. La correlación entre estas dos variables indica que el tiempo estimado de vuelo se suele cumplir ya que variaciones significativas en el mismo producirían una correlación menor. Esto también nos permite conocer que la mayoría de los retrasos se producen en el aeropuerto de salida, puesto que, si a veces se produjera en el de salida, tras en el de llegada y otras durante el trayecto, no existiría una correlación entre estas variables como la que se aprecia en el siguiente gráfico:



A modo de comparación con el punto anterior, se ha calculado el valor exacto de la correlación entre estas variables almacenado en *corr_delay*, con un resultad de 0.935, que muestra una menor correlación que la existente entre el tiempo de vuelo y la distancia.

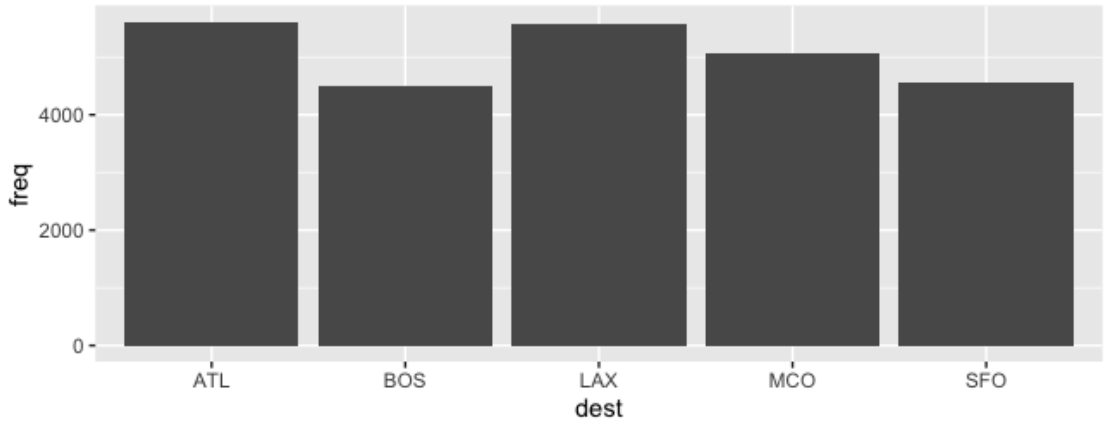
- La cuarta conclusión está relacionada con la hora a la que es más frecuente que salgan los vuelos. Los valores de los cuartiles, la desviación estándar, la media, la mediana y la moda, indicaban que la gran mayoría de los vuelos se registrarían entre las 8AM y las 18PM, y mediante el uso de una representación gráfica podemos ver la distribución de los vuelos y confirmar las hipótesis realizadas:



Durante la realización de otros ejercicios se ha comentado que la variable *hour* sería bastante simétrica, y en el gráfico anterior se demuestra que, si se puede considerar una gráfica simétrica, aunque evidentemente no del todo. Observamos como la moda es claramente las 8AM, una hora perfecta para vuelos internos con un objetivo laboral.

Para la realización de este apartado se ha modificado el dataframe original y almacenado en la variable *df_hour* para poder realizar el gráfico.

- La quinta conclusión tiene que ver con los vuelos más frecuentes desde Nueva York. Es interesante analizar los destinos a los que se voló de forma más frecuente durante el periodo de tiempo que recoge el dataset. En el siguiente gráfico se muestran los aeropuertos que recibieron más vuelos procedentes de Nueva York:



Se observa que el aeropuerto que recibió más vuelos fue el de Atlanta, que puede ser extraño al no ser una ciudad tan conocida como otras de EEUU, pero el aeropuerto de Atlanta es el de mayor tráfico del mundo, lo que ayuda a comprender que sea el que más vuelos tuviera con Nueva York. También tenemos 3 aeropuertos de 3 ciudades muy importantes como son Boston, San Francisco y Los Ángeles, por lo que se entiende que haya una gran frecuencia de vuelos, y por último el aeropuerto MCO, es el aeropuerto de Orlando, en el estado de Florida que, a parte de ser una ciudad importante, es un habitual sitio de vacaciones en los EEUU.