WP\_analysis

library('RSQLite')  
mydb <- dbConnect(RSQLite::SQLite(), "flights.sqlite")   
dbListTables(mydb)

## [1] "airlines" "airports" "cities" "ny\_flights" "stocks"

airlines<-dbGetQuery(mydb, 'SELECT \* FROM airlines')   
airports<-dbGetQuery(mydb, 'SELECT \* FROM airports')  
cities<-dbGetQuery(mydb, 'SELECT \* FROM cities')  
ny\_flights<-dbGetQuery(mydb, 'SELECT \* FROM ny\_flights')

ds=list(airlines=airlines, airports=airports, cities=cities, ny\_flights=ny\_flights)  
ds2=list("airlines", "airports", "cities", "ny\_flights")  
for (i in 1:length(ds)){  
 print(ds2[[i]])  
 print(c('rows:', nrow(ds[[i]])))  
 print(c('cols:', length(ds[[i]])))   
 print(c('missings:', sum(!complete.cases(ds[[i]]))))  
 print(c('duplicates:', anyDuplicated(ds[[i]])))  
 print(' ')   
}

## [1] "airlines"  
## [1] "rows:" "1665"   
## [1] "cols:" "3"   
## [1] "missings:" "0"   
## [1] "duplicates:" "0"   
## [1] " "  
## [1] "airports"  
## [1] "rows:" "6520"   
## [1] "cols:" "3"   
## [1] "missings:" "1"   
## [1] "duplicates:" "0"   
## [1] " "  
## [1] "cities"  
## [1] "rows:" "5898"   
## [1] "cols:" "2"   
## [1] "missings:" "0"   
## [1] "duplicates:" "0"   
## [1] " "  
## [1] "ny\_flights"  
## [1] "rows:" "825413"  
## [1] "cols:" "23"   
## [1] "missings:" "25158"   
## [1] "duplicates:" "0"   
## [1] " "

#(2) data cleaning;  
nrow(ny\_flights)

## [1] 825413

ny\_flights<-ny\_flights[which(ny\_flights$cancelled==0 & ny\_flights$'diverted'==0), ]  
nrow(ny\_flights)

## [1] 801595

citylt<-list()   
for (i in 1:nrow(cities)){   
 citylt[[i]]<-cities[i, 2]  
}  
names(citylt)<-cities[, 1]  
ny\_flights$origin\_city\_market\_nm<-citylt[as.character(ny\_flights$origin\_city\_market\_id)]  
ny\_flights$dest\_city\_market\_nm<-citylt[as.character(ny\_flights$dest\_city\_market\_id)]  
  
#merge airline names;   
airlns<-list()  
for (i in 1:nrow(airlines)){  
 airlns[[i]]<-airlines[i, "airline\_name"]  
}   
names(airlns)<-airlines[, 'id']   
ny\_flights$op\_carrier\_airline\_nm<-airlns[as.character(ny\_flights$op\_carrier\_airline\_id)]  
  
#merge airport names;   
airpts<-list()  
for (i in 1:nrow(airports)){  
 airpts[[i]]<-airports[i, "airport\_name"]  
}   
names(airpts)<-airports[, 'id']   
ny\_flights$origin\_airport\_nm<-airpts[as.character(ny\_flights$origin\_airport\_id)]  
ny\_flights$dest\_airport\_nm<-airpts[as.character(ny\_flights$dest\_airport\_id)]   
  
#add date features;   
ny\_flights$fl\_wkday <- weekdays(as.Date(ny\_flights$fl\_date))  
# unique(ny\_flights$fl\_wkday)  
ny\_flights$fl\_yr<-as.numeric(format(as.Date(ny\_flights$fl\_date, format="%Y-%m-%d"), '%Y'))  
ny\_flights$fl\_mon<-as.numeric(format(as.Date(ny\_flights$fl\_date, format="%Y-%m-%d"), '%m'))   
#define the route for analysis;   
orig\_arpts=c(12953, 12478, 11618)  
dest\_arpts=c(13930)  
routes<-ny\_flights[which(ny\_flights$origin\_airport\_id %in% orig\_arpts & ny\_flights$dest\_airport\_id %in% dest\_arpts), ]   
  
table(routes$fl\_yr)

##   
## 2018 2019   
## 10285 8702

table(routes$fl\_yr, routes$fl\_mon)

##   
## 1 2 3 4 5 6 7 8 9 10 11 12  
## 2018 0 0 0 0 0 1509 1490 1509 1462 1569 1396 1350  
## 2019 1333 1357 1531 1488 1527 1466 0 0 0 0 0 0

routes<-routes[routes$fl\_yr==2018, ]  
nrow(routes)

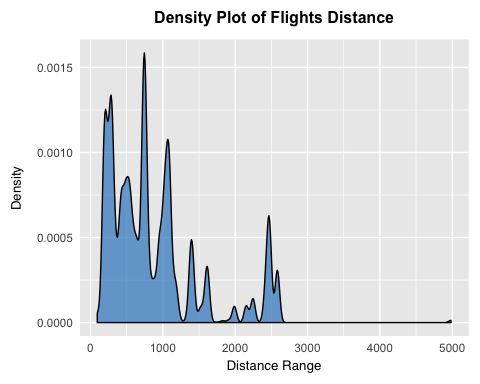
## [1] 10285

# duplicated(routes)   
  
#(3) Visualizations;   
library("ggplot2")   
install.packages('ggthemes')

##   
## There is a binary version available but the source version is later:  
## binary source needs\_compilation  
## ggthemes 3.4.0 4.2.0 FALSE

## installing the source package 'ggthemes'

library('ggthemes')  
test<-ny\_flights  
  
#plot3;   
b<-ggplot(data=test, aes(x=distance))   
b2<-b+geom\_density(color="black", fill="#0073C2FF", alpha=0.6,size=0.5, kernel='gaussian') #alpha: color depth; linetype='dotted',  
b2<-b2+labs(title="Density Plot of Flights Distance", y="Density", x="Distance Range")   
b2<-b2+theme(plot.title = element\_text(color = "black", size = 12, face = "bold", hjust = 0.5, margin=margin(0,0,10,0)),  
 axis.title.x = element\_text(family = "sans", size = 10, margin=margin(5,0,0,0)),  
 axis.title.y = element\_text(family = "sans", size = 10, margin=margin(0,10,0,0)),   
 plot.margin=unit(c(0.3,0.3,0.3,0.3),"cm"), axis.ticks=element\_blank())  
b2

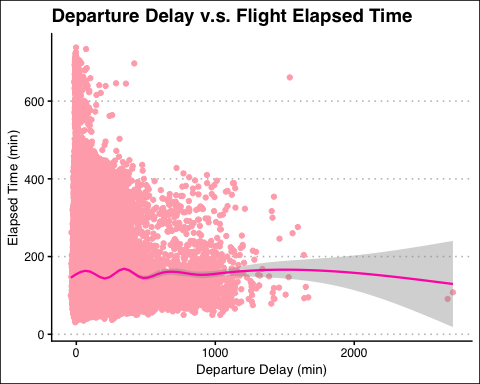


#plot5;   
d<-ggplot(data=test, aes(x=dep\_delay, y=actual\_elapsed\_time))  
d<-d+geom\_point(color="lightpink1")+geom\_smooth(se=TRUE, linetype="solid", color="maroon1", size=0.8)   
d<-d+labs(title="Departure Delay v.s. Flight Elapsed Time", y="Elapsed Time (min)", x="Departure Delay (min)")   
d<-d+theme(plot.title = element\_text(size = 10))  
d+theme\_clean()

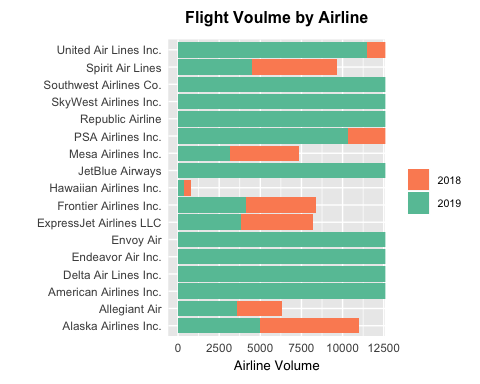
## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'

## Warning: Removed 936 rows containing non-finite values (stat\_smooth).

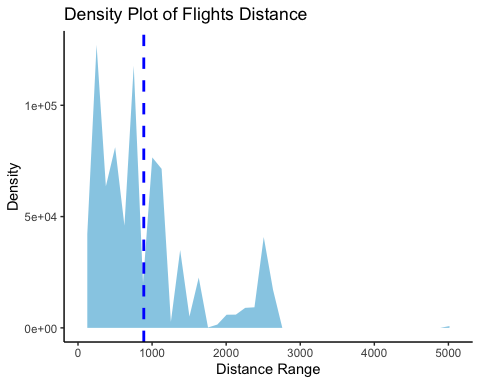
## Warning: Removed 936 rows containing missing values (geom\_point).



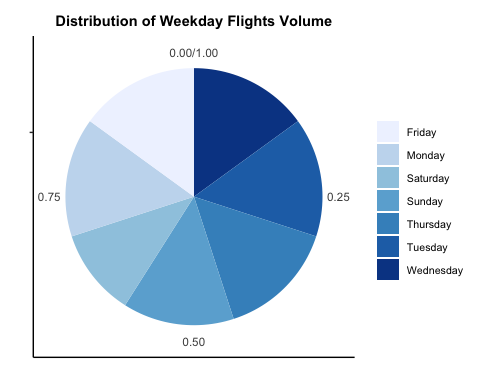
#plot6;   
temp1<-as.data.frame(table(ny\_flights[ny\_flights$fl\_yr==2018, "op\_carrier\_airline\_id"]))  
temp2<-as.data.frame(table(ny\_flights[ny\_flights$fl\_yr==2019, "op\_carrier\_airline\_id"]))  
temp1['year']<-2018  
temp2['year']<-2019  
temp<-rbind(temp1, temp2)  
temp<-merge(x=temp, y=airlines[airlines$id %in% temp1$Var1, ], by.x="Var1", by.y="id", all.x=TRUE)  
temp<-temp[, c("airline\_name", "Freq", "year")]  
f<-ggplot(data=temp, aes(x=airline\_name, y=Freq, fill=as.factor(year), group=as.factor(year)))  
f<-f+geom\_bar(stat="identity") #width=0.5, position="dodge"  
f<-f+coord\_flip(ylim=c(5, 12000))+scale\_color\_hue(l=40, c=35)  
f<-f+labs(title="Flight Voulme by Airline", y="Airline Volume", x="")   
f<-f+theme(legend.title = element\_blank(),   
 legend.text=element\_text(size=8),   
 plot.title = element\_text(color = "black", size = 12, face = "bold", hjust = 0.5, margin=margin(0,0,10,0)),  
 axis.title.x = element\_text(family = "sans", size = 10, margin=margin(5,0,0,0)),  
 axis.title.y = element\_text(family = "sans", size = 10, margin=margin(0,10,0,0)),   
 plot.margin=unit(c(0.3,0.3,0.3,0.3),"cm"), axis.ticks=element\_blank())  
f+scale\_fill\_brewer(palette="Set2", direction = -1)



#plot4;   
b3<-b+geom\_area(stat="bin", fill="deepskyblue3", alpha=0.5, bins=40)+theme\_classic()  
b3<-b3+geom\_vline(aes(xintercept=mean(distance)), color="blue", linetype="dashed", size=1)  
b3<-b3+labs(title="Density Plot of Flights Distance", y="Density", x="Distance Range")   
b3



#plot1;   
a1<-table(test$fl\_wkday)  
a1<-round(a1/sum(a1), 2)   
a1<-as.data.frame(a1)   
a2<-ggplot(data=a1, aes(x="", y=Freq, fill=Var1))   
a2<-a2+geom\_bar(stat='identity', width=3)+coord\_polar(theta="y", start=0)+scale\_fill\_brewer(palette=1)  
a2<-a2+labs(title="Distribution of Weekday Flights Volume", y="", x="")+theme\_classic()   
a2<-a2+theme(legend.title = element\_blank(),   
 legend.text=element\_text(size=8),   
 legend.background = element\_rect(color = NA),   
 plot.title = element\_text(color = "black", size = 11, face = "bold", hjust = 0.5),   
 plot.margin=unit(c(0.4,0.2,0.01,0.2),"cm"))  
print(a2)



#plot2;   
b<-ggplot(data=test, aes(x=distance))   
b1<-b+geom\_freqpoly(bins=20, color='royalblue4')+theme\_economist()   
b1<-b1+labs(title="Majority of Flights Distance Stay between [0, 2000]", y="# of Flights", x="Distance Range")  
b1<-b1+theme(plot.title = element\_text(color = "black", size = 13, face = "bold", hjust = 0.5, margin=margin(0,0,20,0)),  
 axis.title.x = element\_text(family = "sans", size = 10, margin=margin(10,0,0,0)),  
 axis.title.y = element\_text(family = "sans", size = 10, margin=margin(0,10,0,0)),   
 plot.margin=unit(c(0.3,0.3,0.3,0.3),"cm"), axis.ticks=element\_blank())   
print(b1)

