SQL Week2 Assignment Answer

Yu Bai

July 12, 2015

Q1: What weather conditions are associated with New York City departure delays?

Solution:

1. Found outlier data in *wind_speed*, *wind_gust* columns in the *weather* table.
query *wind_speed* column:

```
SELECT avg(wind_speed) as avg_wind_speed,
       stddev(wind_speed) as std_wind_speed,
       min(wind_speed) as min_wind_speed,
       max(wind_speed) as max_wind_speed
FROM weather
WHERE wind_speed is not null and origin='EWR'
```

The result is

```
avg_wind_speed | std_wind_speed | min_wind_speed | max_wind_speed
-------------- | -------------- | -------------- | --------------
9.33           | 12.33          | 0              | 1048.36
```

query *wind_gust* column:

```
SELECT avg(wind_gust) as avg_wind_gust,
       stddev(wind_gust) as std_wind_gust,
       min(wind_gust) as min_wind_gust,
       max(wind_gust) as max_wind_gust
FROM weather
WHERE wind_gust is not null and origin='EWR'
```

The result is

```
avg_wind_gust | std_wind_gust | min_wind_gust | max_wind_gust
--------------| --------------| ------------- | --------------
10.74         | 14.19         | 0             | 1206.43
```

Similar queries executed on all other parameter columns in the *weather* table (date not shown) and no notable outliers need to be removed.

2. Run query to join the *weather* and the *flights* tables at EWR airport (using EWR airport to represent the NYC airports) on matching time points[&]. Then obtain mean duration of the delays, mean percetage of the delayed flights, and mean values of all parameters in the *weather* table averaged over for each time point[#].

[&] The caveat here is that the originally scheduled departure time of the flights, not the actual departure time (the *hour* column), should be used to match the time when the weather conditions were measured. The delay ended at the time the flights actually departed, by then the weather condition might be no longer be a cause.

[#] The available resolution for the time point is hour. However, there are relatively small number (1-38) of flights per hour. Thus, instead of data per hour, I generated data per day for the subsequent regression analysis. I also analyzed the per-hour data and got consistent conclusions.

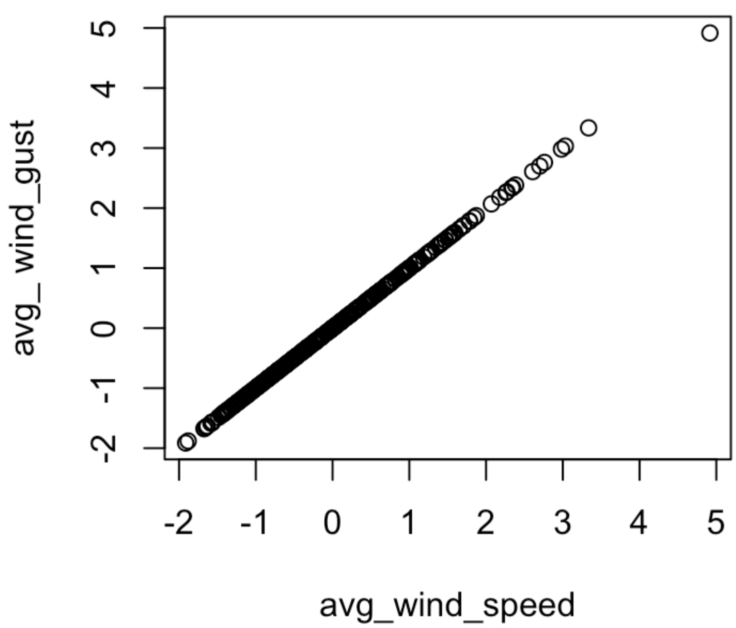
```
--Note negative dep_delay should not be considered as "less delay"" but rather no delay (delay=0)
COPY(
SELECT concat(f.year,'-',f.month,'-',f.day) as timepoint,
       count(*) as num_flights,
       round(cast(sum(case when f.dep_delay >0 then 1 else 0 end ) as numeric) / count(*),2) as delay_rate,
       avg(case when f.dep_delay <0 then 0 when f.dep_delay >0 then f.dep_delay end) as delay_length,
       avg(case when w.visib is not null then w.visib end) as avg_visibility,
       avg(case when w.wind_gust is not null and w.wind_gust<1000 then w.wind_gust end) as avg_wind_gust,
       avg(case when w.wind_speed is not null and w.wind_speed<1000 then w.wind_speed end) as avg_wind_speed,
       avg(case when w.wind_dir is not null then w.wind_dir end) as avg_wind_dir,
       avg(case when w.precip is not null then w.precip end) as avg_precipitation,
       avg(case when w.temp is not null then w.temp end) as avg_temperature,
       avg(case when w.dewp is not null then w.dewp end) as avg_dew_point,
       avg(case when w.humid is not null then w.humid end) as avg_humidity,
       avg(case when w.pressure is not null then w.pressure end) as avg_pressure
FROM weather w join flights f on f.year=w.year and f.month=w.month and f.day=w.day and f.hour=w.hour
WHERE f.origin in ('EWR') and f.origin=w.origin
GROUP BY timepoint
ORDER BY delay_rate desc
)to 'weather_delay.csv' with CSV HEADER
```

3. Regression analysis

The weather parameters are standardrized to make the beta weights from the regression comparable.

```
weather_delay[1:3,]
##    timepoint num_flights delay_rate delay_length avg_visibility
## 1 2013-12-23         347       0.85     43.96471        6.32853
## 2 2013-12-22         299       0.83     47.76451       10.00000
## 3   2013-3-8         266       0.82    100.98069        2.62782
##   avg_wind_gust avg_wind_speed avg_wind_dir avg_precipitation
## 1      7.888521       6.854934     218.2540       0.016974063
## 2     16.511606      14.348187     209.1639       0.001337793
## 3     14.139085      12.286523     320.3759       0.022744361
##   avg_temperature avg_dew_point avg_humidity avg_pressure
## 1        58.09896      55.96127     92.61749     1015.147
## 2        66.88388      59.14027     76.46896     1011.507
## 3        33.95226      31.26579     90.15120     1017.639

#The weather parameters are standarized such that beta weights from fitting can be
comparable.
for(each in 3:13){  weather_delay[,each] <- scale(weather_delay[,each]) }
```

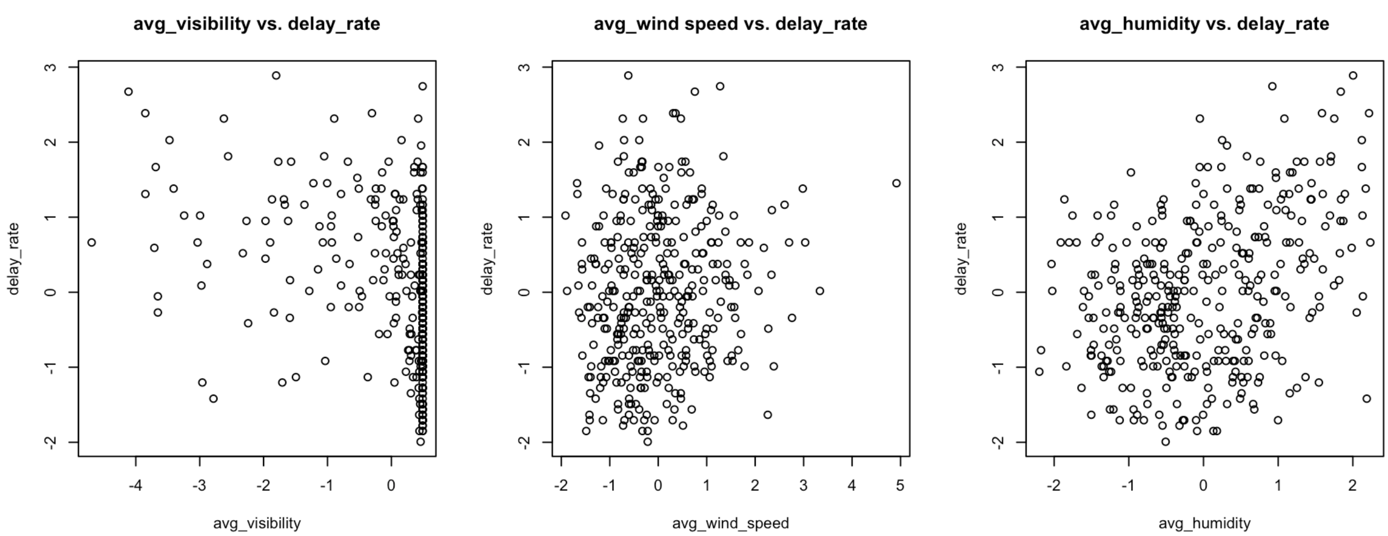
The *avg_wind_speed* and *avg_wind_gust* were closely correlated with each other and thus I dropped one (*avg_wind_gust*) from the predictor variables



When using *avg_delay_rate* as response, a

LASSO based variable selection (*glmnet* package) resulted in *avg_visibility*, *avg_speed* and *avg_humidity* as the main parameters (judging by their beta coefficient values) associated with the delay rate, *avg_precipitation* and *avg_pressure* may also have an impact but to a less extent.
Among the main predictors, the *avg_visibility* value is negatively correlated with the delay rate, i.e. the lower visibility and higher the delay rate. On the other hand, the higher and *avg_wind_speed* and *avg_humidity*, the higher the delay rate.

```
library(glmnet)
preds <- as.matrix(weather_delay[,c("avg_visibility","avg_wind_speed","avg_wind_di
r","avg_precipitation","avg_temperature", "avg_dew_point","avg_humidity","avg_pres
sure")])
response <- as.matrix(weather_delay$delay_rate)
delay_rate_fit <-cv.glmnet(preds, response)
coef(delay_rate_fit,s=delay_rate_fit$lambda.1se)
## 9 x 1 sparse Matrix of class "dgCMatrix"
##                              1
## (Intercept)        1.091339e-15
## avg_visibility    -1.297993e-01
## avg_wind_speed     2.488400e-02
## avg_wind_dir       .
## avg_precipitation  5.834611e-02
## avg_temperature    .
## avg_dew_point      .
## avg_humidity       9.779463e-02
## avg_pressure      -5.423394e-02
```



If use *avg_delay_length* as response, regression yielded similar results: *avg_visibility* and *avg_humidity* are the main contributors, followed by *avg_precipitation* and *avg_pressure*. However, *avg_wind_speed* is not a significant predictor in this case. It could be due to noise in the data, or *avg_wind_speed* affected whether the flights were delayed but not necessarily how long the delay would be.

```
response <- as.matrix(weather_delay$delay_length)
delay_length_fit <-cv.glmnet(preds, response)
coef(delay_length_fit,s=delay_rate_fit$lambda.1se)
## 9 x 1 sparse Matrix of class "dgCMatrix"
##                               1
## (Intercept)       -2.490674e-16
## avg_visibility    -1.497911e-01
## avg_wind_speed     .
## avg_wind_dir       .
## avg_precipitation  3.088984e-02
## avg_temperature    .
## avg_dew_point      .
## avg_humidity       1.308297e-01
## avg_pressure      -6.588910e-02
```

4. Conclusion to Q1: the *avg_visibility*, *avg_humidity* and *avg_speed* appear to be the main parameters that are associated with the percentage of flights delayed per day

Q2: Are older planes more likely to be delayed?

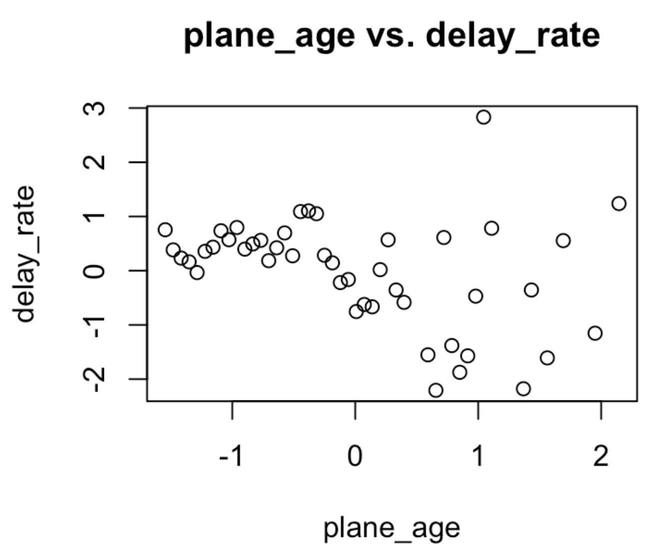
Solution:

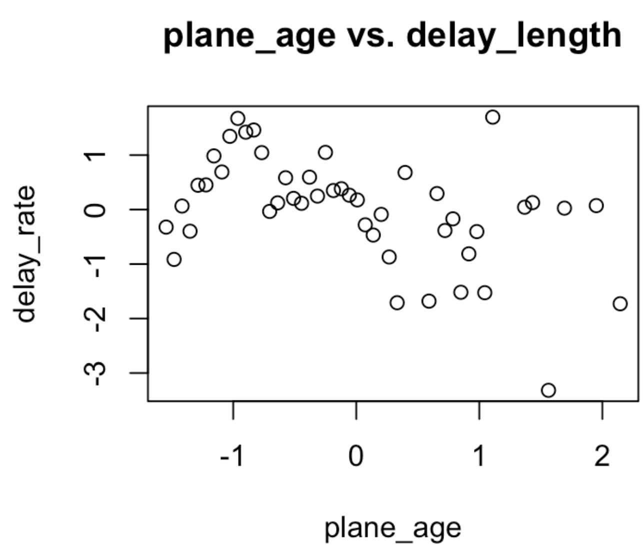
1. After joining *flights* and *planes* tables on *tailnum*, two steps of aggregations were applied to obtain the delay measurements (length and rate) with respect to the plane ages. The first step was to average the delays over the flights from the same plane. This is because the number of flights per plane vary a lot, directly averaging over all flights is not a fair sampling among planes but biased towards those with many flights. A virtual VIEW table was created to hold the results from the step 1 query. The second step is to query on the virtual table to obtain the average delays and plane ages for subsequent statistical analysis

```
DROP VIEW IF EXISTS plane_delay;
CREATE VIEW plane_delay AS
(SELECT p.tailnum as tailnum, (2015-p.year) as plane_age ,
       count(*) as num_flights,
       round(cast(sum(case when f.dep_delay >0 then 1 else 0 end ) as numeric) / c
ount(*), 2) as delay_rate,
       avg(case when f.dep_delay <0 then 0 when f.dep_delay >0 then f.dep_delay en
d) as delay_length
FROM flights f join planes p on f.tailnum=p.tailnum
WHERE f.origin in ('LGA','JFK','EWR') and f.dep_delay is not null and p.year is no
t null
GROUP BY p.tailnum, (2015-p.year)
ORDER BY plane_age desc
)

COPY(
SELECT plane_age, count(*) as num_planes,
     avg(delay_rate) as delay_rate, avg(delay_length) as delay_length
FROM plane_delay
GROUP BY plane_age
ORDER BY plane_age desc
)to 'plane_delay.csv' with CSV HEADER
```

2. Linear regression

```
head(plane_delay)
##   plane_age num_planes delay_rate delay_length
## 1        59          1       0.50     7.809524
## 2        56          2       0.29    15.247909
## 3        52          2       0.44    15.055556
## 4        50          1       0.25     1.250000
## 5        48          1       0.36    15.473684
## 6        47          1       0.20    15.125000

#standardrize variables
for(each in c("plane_age","delay_rate","delay_length")){plane_delay[,each] <- scal
e(plane_delay[,each])}

#lm() fitting on response:delay_rate and predictor:plane_age
fit=lm(delay_rate~plane_age,data=plane_delay)
summary(fit)$coefficients
##                  Estimate Std. Error       t value   Pr(>|t|)
## (Intercept) -3.009198e-16  0.1388054 -2.167925e-15 1.00000000
## plane_age   -3.652607e-01  0.1403392 -2.602698e+00 0.01255845

# lm() fitting on response:delay_length and predictor:plane_age
fit=lm(delay_length~plane_age,data=plane_delay)
summary(fit)$coefficients
##                  Estimate Std. Error       t value    Pr(>|t|)
## (Intercept)  7.856142e-18  0.1323954  5.933849e-17 1.000000000
## plane_age   -4.600058e-01  0.1338584 -3.436512e+00 0.001297947
```





3. Conclusion for Q2: Above statistical analysis suggests that plane age is a sigificant factor to either delay rate (percentage of flights delayed) or delay length (duration of the delay). Nevertheless, the correlation is negative. That is older planes are less delayed, not more. A possible underlying explanation could be older planes fly less and thus have less occassions to get delayed.

Q3: Dose plane capacity affect its flight distance?

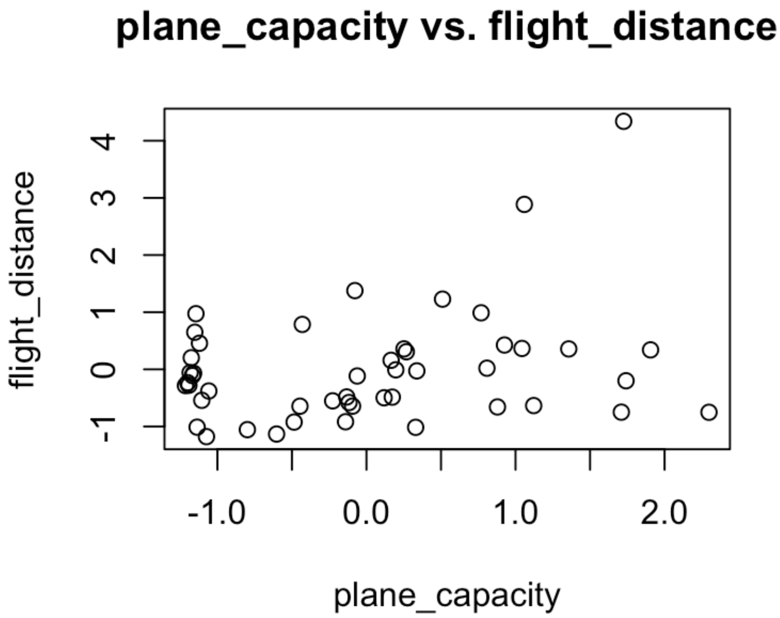
Solution:

1. Obtain average flight distance with respect to the plane capacity. Again, the flight distances were averaged first per plane and then per plane capacity

```
DROP VIEW IF EXISTS size_distance;
CREATE VIEW size_distance AS
(SELECT p.tailnum as tailnum, p.seats as plane_capacity,
       count(*) as num_flights,
       avg(case when f.distance>0 then f.distance end) as flight_distance
FROM flights f join planes p on f.tailnum=p.tailnum
WHERE f.origin in ('LGA','JFK','EWR') and p.seats is not null
GROUP BY p.tailnum, p.seats
ORDER BY p.seats desc
)

COPY(
SELECT plane_capacity, count(*) as num_planes,
        avg(flight_distance) as flight_distance
FROM size_distance
GROUP BY plane_capacity
ORDER BY plane_capacity desc
)to 'size_distance.csv' with CSV HEADER
```

2. linear regression analysis

```
head(size_distance)
##   plane_capacity num_planes flight_distance
## 1            450          1         760.000
## 2            400         12        1665.708
## 3            379         55        1217.278
## 4            377         14        4983.000
## 5            375          1         762.000
## 6            330        114        1679.393

#standardrize variables
for(each in c("plane_capacity","flight_distance")){size_distance[,each] <- scale(s
ize_distance[,each])}

#lm() fitting on response: flight_distance and predictor:plane_capacity
fit=lm(flight_distance~plane_capacity,data=size_distance)
summary(fit)$coefficients
##                     Estimate Std. Error       t value   Pr(>|t|)
## (Intercept)    -9.614813e-17  0.1399986 -6.867790e-16 1.00000000
## plane_capacity  2.814869e-01  0.1414802  1.989586e+00 0.05260104
```



3. Conclusion to Q3: there is no significant association between plane capacity and flight distance.