

# Statistics 452: Statistical Learning and Prediction

## Case Study

*Brad McNeney*

### Flights dataset

This is the dataset being analyzed by the Stat 652 class for their final project. The data are on flights from three New York City airports in 2013, from the `nycflights13` package. Data were combined from four datasets from this package:

- `flights`
- `weather`
- `airports`, and
- `planes`

You can read about the variables in each dataset by typing `help(datasetname)` from the R console. Our goal is to predict departure delays (variable `dep_delay` in minutes).

```
library(tidyverse)
library(nycflights13)
#help(flights)
#help(weather)
#help(airports)
#help(planes)
fltrain <- read_csv("../Project652/fltrain.csv.gz")
fltrain

## # A tibble: 200,000 x 43
##   year.x month   day dep_time sched_dep_time dep_delay arr_time
##   <dbl>   <dbl> <dbl>    <dbl>          <dbl>      <dbl>      <dbl>
## 1 2013     11     7      600            600        0       826
## 2 2013     10    30     1252           1250        2      1356
## 3 2013     12     18     1723           1715        8      2008
## 4 2013     11    20     2029           2030       -1      2141
## 5 2013     10    21     1620           1625       -5      1818
## 6 2013     11     7      852            900       -8      1139
## 7 2013      9    29     1519           1529      -10      1639
## 8 2013     12    21     1526           1530       -4      1654
## 9 2013     11     7     1650           1650        0      1910
## 10 2013     3    31     1652           1700      -8      1810
## # ... with 199,990 more rows, and 36 more variables: sched_arr_time <dbl>,
## #   arr_delay <dbl>, carrier <chr>, flight <dbl>, tailnum <chr>,
## #   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dttm>, temp <dbl>, dewp <dbl>, humid <dbl>,
## #   wind_dir <dbl>, wind_speed <dbl>, wind_gust <dbl>, precip <dbl>,
## #   pressure <dbl>, visib <dbl>, name <chr>, lat <dbl>, lon <dbl>,
## #   alt <dbl>, tz <dbl>, dst <chr>, tzone <chr>, year.y <dbl>, type <chr>,
## #   manufacturer <chr>, model <chr>, engines <dbl>, seats <dbl>,
## #   speed <dbl>, engine <chr>
dim(fltrain)

## [1] 200000     43
```

There are 43 variables measured on 200,000 flights.

## Missing data

Handling of missing data is an important topic, but one that we did not consider in class. Two common ways to deal with missing data are to (i) remove observations with any missing data (complete-case analysis) and (ii) impute missing data. Both have their strengths and limitations. For simplicity we will remove observations. The danger is that our inference and/or predictions could be biased, which happens when the chance of a missing observation depends on the (unobserved) value of the missing data. However, a complete-case analysis is the most straightforward.

To ensure we are not discarding too many data points, we limit the **variables** to those with only a small proportion of missing values. One rule of thumb is to discard variables with more than 5% missing values, which is 10,000 for these data. To this end we count the number of missing values in each variable. The character variables have NA interpreted as a character string. This will be converted to the missing code NA if we coerce to a factor.

```
f1 <- fltrain
for(i in 1:ncol(f1)) {
  if(typeof(f1[[i]]) == "character") {
    f1[[i]] <- factor(f1[[i]])
  }
}
```

Now count the missing values in each variable.

```
num_miss <- function(x) { sum(is.na(x)) }
sapply(f1,num_miss)
```

	year.x	month	day	dep_time	sched_dep_time
##	0	0	0	4898	0
##	dep_delay	arr_time	sched_arr_time	arr_delay	carrier
##	4898	5169	0	5584	0
##	flight	tailnum	origin	dest	air_time
##	0	1492	0	0	5584
##	distance	hour	minute	time_hour	temp
##	0	0	0	0	948
##	dewp	humid	wind_dir	wind_speed	wind_gust
##	948	948	5862	982	152260
##	precip	pressure	visib	name	lat
##	937	23092	937	4484	4484
##	lon	alt	tz	dst	tzone
##	4484	4484	4484	4484	4484
##	year.y	type	manufacturer	model	engines
##	34298	31163	31163	31163	31163
##	seats	speed	engine		
##	31163	199415	31163		

Some of the variables, particularly those taken from the `planes` dataset (`year.y` to `engine`), have many missing values. In what follows I'll discard all of the variables from `planes`, plus `wind_gust` and `pressure`.

```
f1 <- f1%>%
  select(-year.y,-type,-manufacturer,-model,-engines,-seats, -speed, -engine, -wind_gust, -pressure)
summary(f1)

##      year.x          month           day        dep_time
##  Min.   :2013   Min.   : 1.000   Min.   : 1.0   Min.   : 1
```

```

## 1st Qu.:2013   1st Qu.: 4.000   1st Qu.: 8.0   1st Qu.: 907
## Median :2013   Median : 7.000   Median :16.0   Median :1401
## Mean    :2013   Mean   : 6.553   Mean   :15.7   Mean   :1349
## 3rd Qu.:2013   3rd Qu.:10.000   3rd Qu.:23.0   3rd Qu.:1745
## Max.    :2013   Max.   :12.000   Max.   :31.0   Max.   :2400
##                                     NA's   :4898
## sched_dep_time  dep_delay      arr_time  sched_arr_time
## Min.   : 106   Min.   :-43.0   Min.   : 1   Min.   : 1
## 1st Qu.: 905   1st Qu.:-5.0   1st Qu.:1104   1st Qu.:1124
## Median :1359   Median :-2.0   Median :1535   Median :1557
## Mean   :1344   Mean   :12.7   Mean   :1502   Mean   :1537
## 3rd Qu.:1729   3rd Qu.:11.0   3rd Qu.:1941   3rd Qu.:1945
## Max.   :2359   Max.   :1301.0  Max.   :2400   Max.   :2359
##                                     NA's   :4898   NA's   :5169
## arr_delay       carrier       flight     tailnum
## Min.   :-79.000  UA   :34734   Min.   : 1   N725MQ : 350
## 1st Qu.:-17.000  B6   :32355   1st Qu.: 561   N723MQ : 300
## Median :-5.000   EV   :32217   Median :1499   N722MQ : 294
## Mean   : 6.969   DL   :28731   Mean   :1975   N711MQ : 290
## 3rd Qu.: 14.000  AA   :19415   3rd Qu.:3470   N713MQ : 260
## Max.   :1272.000 MQ   :15608   Max.   :8500   (Other):197014
## NA's   :5584     (Other):36940   NA's   :        NA's   : 1492
## origin          dest         air_time   distance
## EWR:71658   ATL   : 10319   Min.   :20.0   Min.   : 17
## JFK:65951   ORD   : 10186   1st Qu.:82.0   1st Qu.: 502
## LGA:62391   LAX   :  9472   Median :129.0   Median : 872
##           BOS   :  9217   Mean   :150.5   Mean   :1038
##           MCO   :  8425   3rd Qu.:191.0   3rd Qu.:1389
##           CLT   :  8319   Max.   :695.0   Max.   :4983
##           (Other):144062  NA's   :5584
## hour          minute      time_hour
## Min.   : 1.00   Min.   : 0.00   Min.   :2013-01-01 10:00:00
## 1st Qu.: 9.00   1st Qu.: 8.00   1st Qu.:2013-04-04 20:00:00
## Median :13.00   Median :29.00   Median :2013-07-03 15:00:00
## Mean   :13.18   Mean   :26.22   Mean   :2013-07-03 12:05:05
## 3rd Qu.:17.00   3rd Qu.:44.00   3rd Qu.:2013-10-01 12:00:00
## Max.   :23.00   Max.   :59.00   Max.   :2014-01-01 04:00:00
##
## temp          dewp        humid      wind_dir
## Min.   :10.94   Min.   :-9.94   Min.   :12.74   Min.   : 0.0
## 1st Qu.:42.08   1st Qu.:26.06   1st Qu.:43.99   1st Qu.:130.0
## Median :57.20   Median :42.80   Median :57.69   Median :220.0
## Mean   :56.98   Mean   :41.62   Mean   :59.57   Mean   :201.5
## 3rd Qu.:71.96   3rd Qu.:57.92   3rd Qu.:75.33   3rd Qu.:290.0
## Max.   :100.04  Max.   :78.08   Max.   :100.00  Max.   :360.0
## NA's   :948     NA's   :948     NA's   :948     NA's   :5862
## wind_speed    precip      visib
## Min.   : 0.000  Min.   :0.0000  Min.   : 0.000
## 1st Qu.: 6.905  1st Qu.:0.0000  1st Qu.:10.000
## Median :10.357  Median :0.0000  Median :10.000
## Mean   :11.107  Mean   :0.0045  Mean   : 9.252
## 3rd Qu.:14.960  3rd Qu.:0.0000  3rd Qu.:10.000
## Max.   :42.579  Max.   :1.2100  Max.   :10.000
## NA's   :982     NA's   :937     NA's   :937

```

```

##                               name          lat
## Hartsfield Jackson Atlanta Intl : 10319 Min.   :21.32
## Chicago Ohare Intl           : 10186 1st Qu.:32.90
## Los Angeles Intl             :  9472 Median :36.10
## General Edward Lawrence Logan Intl: 9217 Mean   :36.02
## Orlando Intl                :  8425 3rd Qu.:41.41
## (Other)                      :147897 Max.   :61.17
## NA's                         :  4484 NA's   :4484
##          lon          alt          tz          dst
## Min.   :-157.92   Min.   : 3.0   Min.   :-10.000 A   :192358
## 1st Qu.:-95.28   1st Qu.: 26.0   1st Qu.: -6.000 N   : 3158
## Median :-83.35   Median :433.0   Median : -5.000 NA's: 4484
## Mean   :-89.44   Mean   :582.5   Mean   : -5.748
## 3rd Qu.:-80.15   3rd Qu.:748.0   3rd Qu.: -5.000
## Max.   :-68.83   Max.   :6602.0  Max.   : -5.000
## NA's   :4484     NA's   :4484    NA's   :4484
##          tzone
## America/New_York   :114518
## America/Chicago    : 44400
## America/Los_Angeles: 27368
## America/Denver     : 6069
## America/Phoenix    : 2759
## (Other)            :  402
## NA's               : 4484

```

When we omit rows with any missing values we end up with 184,316 rows out of the original 200,000.

```

f1 <- na.omit(f1)
summary(f1)

```

```

##      year.x       month       day      dep_time
## Min.   :2013   Min.   : 1.000   Min.   : 1.00   Min.   : 1
## 1st Qu.:2013   1st Qu.: 4.000   1st Qu.: 8.00   1st Qu.: 910
## Median :2013   Median : 7.000   Median :16.00   Median :1408
## Mean   :2013   Mean   : 6.553   Mean   :15.67   Mean   :1353
## 3rd Qu.:2013   3rd Qu.:10.000   3rd Qu.:23.00   3rd Qu.:1747
## Max.   :2013   Max.   :12.000   Max.   :31.00   Max.   :2400
##
##      sched_dep_time dep_delay      arr_time      sched_arr_time
## Min.   : 500   Min.   :-43.00   Min.   : 1   Min.   : 1
## 1st Qu.: 905   1st Qu.:- 5.00   1st Qu.:1106   1st Qu.:1123
## Median :1359   Median : -2.00   Median :1545   Median :1602
## Mean   :1342   Mean   : 12.67   Mean   :1511   Mean   :1544
## 3rd Qu.:1729   3rd Qu.: 11.00   3rd Qu.:1946   3rd Qu.:1950
## Max.   :2345   Max.   :1301.00   Max.   :2400   Max.   :2359
##
##      arr_delay      carrier      flight      tailnum
## Min.   :-79.000   UA   :32252   Min.   : 1   N725MQ : 322
## 1st Qu.:-17.000   B6   :29282   1st Qu.: 544   N723MQ : 271
## Median : -5.000   EV   :29137   Median :1499   N711MQ : 268
## Mean   :  7.014   DL   :26998   Mean   :1966   N722MQ : 268
## 3rd Qu.: 14.000   AA   :17742   3rd Qu.:3448   N351JB : 247
## Max.   :1272.000  MQ   :14382   Max.   :8500   N258JB : 244
## (Other) :34523
##
##      origin        dest      air_time      distance

```

```

## EWR:65512 ATL : 9726 Min. : 20.0 Min. : 80
## JFK:60327 ORD : 9443 1st Qu.: 81.0 1st Qu.: 502
## LGA:58477 LAX : 9185 Median :127.0 Median : 866
## BOS : 8674 Mean :149.5 Mean :1035
## MCO : 8131 3rd Qu.:184.0 3rd Qu.:1372
## CLT : 7822 Max. :695.0 Max. :4983
## (Other):131335

##      hour       minute      time_hour
## Min. : 5.00   Min. : 0.00   Min. :2013-01-01 10:00:00
## 1st Qu.: 9.00   1st Qu.: 8.00   1st Qu.:2013-04-04 13:00:00
## Median :13.00   Median :29.00   Median :2013-07-03 17:00:00
## Mean :13.15   Mean :26.06   Mean :2013-07-03 11:34:16
## 3rd Qu.:17.00   3rd Qu.:43.00   3rd Qu.:2013-10-02 10:00:00
## Max. :23.00   Max. :59.00   Max. :2013-12-30 23:00:00
##
##      temp        dewp       humid      wind_dir
## Min. : 10.94   Min. : -9.94   Min. : 13.00   Min. : 0.0
## 1st Qu.: 42.08   1st Qu.: 26.06   1st Qu.: 43.71   1st Qu.:130.0
## Median : 57.02   Median : 42.08   Median : 57.14   Median :220.0
## Mean : 56.86   Mean : 41.33   Mean : 59.12   Mean :201.9
## 3rd Qu.: 71.96   3rd Qu.: 57.20   3rd Qu.: 74.29   3rd Qu.:290.0
## Max. :100.04   Max. : 78.08   Max. :100.00   Max. :360.0
##
##      wind_speed     precip      visib
## Min. : 0.000   Min. :0.000000   Min. : 0.000
## 1st Qu.: 6.905   1st Qu.:0.000000   1st Qu.:10.000
## Median :10.357   Median :0.000000   Median :10.000
## Mean :11.202   Mean :0.004059   Mean : 9.294
## 3rd Qu.:14.960   3rd Qu.:0.000000   3rd Qu.:10.000
## Max. :42.579   Max. :1.210000   Max. :10.000
##
##      name       lat
## Hartsfield Jackson Atlanta Intl : 9726 Min. :21.32
## Chicago Ohare Intl           : 9443 1st Qu.:32.90
## Los Angeles Intl             : 9185 Median :36.08
## General Edward Lawrence Logan Intl: 8674 Mean :35.97
## Orlando Intl                 : 8131 3rd Qu.:41.41
## Charlotte Douglas Intl       : 7822 Max. :61.17
## (Other)                      :131335

##      lon       alt      tz      dst
## Min. : -157.92   Min. : 3.0   Min. : -10.000 A:181313
## 1st Qu.: -95.34   1st Qu.: 26.0   1st Qu.: -6.000 N: 3003
## Median : -83.35   Median : 433.0   Median : -5.000
## Mean : -89.58   Mean : 582.3   Mean : -5.757
## 3rd Qu.: -80.15   3rd Qu.: 748.0   3rd Qu.: -5.000
## Max. : -68.83   Max. :6602.0   Max. : -5.000
##
##      tzone
## America/Anchorage : 3
## America/Chicago : 41444
## America/Denver : 5819
## America/Los_Angeles: 26461
## America/New_York :107586
## America/Phoenix : 2629

```

```
## Pacific/Honolulu : 374
```

## Summaries of the response variable dep\_delay

The departure delays variable is highly right-skewed.

```
range(f1$dep_delay)  
  
## [1] -43 1301  
  
fivenum(f1$dep_delay)  
  
## [1] -43 -5 -2 11 1301  
  
quantile(f1$dep_delay,probs = c(0.01,0.05,0.1,0.25,.5,.75,.90,.95,.99))  
  
## 1% 5% 10% 25% 50% 75% 90% 95% 99%  
## -12 -9 -7 -5 -2 11 49 88 193  
  
mean(f1$dep_delay >= 60) # about 15,000 or 8% of flights
```

```
## [1] 0.08210356
```

Top 10 delays.

```
f1 %>% arrange(desc(dep_delay)) %>% head(10)
```

```
## # A tibble: 10 x 33  
##   year.x month day dep_time sched_dep_time dep_delay arr_time  
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2013     1     9     641      900    1301    1242  
## 2 2013     9    20    1139     1845    1014    1457  
## 3 2013     3    17    2321      810     911     135  
## 4 2013     7    22    2257      759     898     121  
## 5 2013    12     5     756     1700     896    1058  
## 6 2013     5    19     713     1700     853    1007  
## 7 2013     2    10    2243      830     853     100  
## 8 2013    12    19     734     1725     849    1046  
## 9 2013    12    17     705     1700     845    1026  
## 10 2013    12    14     830     1845     825    1210  
## # ... with 26 more variables: sched_arr_time <dbl>, arr_delay <dbl>,  
## #   carrier <fct>, flight <dbl>, tailnum <fct>, origin <fct>, dest <fct>,  
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,  
## #   time_hour <dttm>, temp <dbl>, dewp <dbl>, humid <dbl>, wind_dir <dbl>,  
## #   wind_speed <dbl>, precip <dbl>, visib <dbl>, name <fct>, lat <dbl>,  
## #   lon <dbl>, alt <dbl>, tz <dbl>, dst <fct>, tzone <fct>
```

Summaries of departure delay by NYC airport:

```
Q3 <- function(x) { quantile(x,probs=.75) }  
f1 %>% group_by(origin) %>%  
  summarize(n=n(),med_d = median(dep_delay),Q3_d = Q3(dep_delay), max_d = max(dep_delay)) %>%  
  arrange(desc(Q3_d)) %>% head(10)  
  
## # A tibble: 3 x 5  
##   origin     n med_d  Q3_d max_d  
##   <fct> <int> <dbl> <dbl> <dbl>  
## 1 EWR     65512    -1     15    896  
## 2 JFK     60327    -1     10    1301
```

```

## 3 LGA      58477     -3      7   911

Summaries of departure delay by airline (carrier).

f1 %>% group_by(carrier) %>%
  summarize(n=n(),med_d = median(dep_delay),Q3_d = Q3(dep_delay), max_d = max(dep_delay)) %>%
  arrange(desc(Q3_d)) %>% head(10)

## # A tibble: 10 x 5
##   carrier     n med_d   Q3_d max_d
##   <fct>   <int> <dbl> <dbl> <dbl>
## 1 EV        29137    -1   25    536
## 2 WN        6897     1   18    471
## 3 F9        388      0  17.2   853
## 4 9E       10179    -2   16    430
## 5 FL        1832     1   16    602
## 6 YV        312     -3   13    387
## 7 B6       29282    -1   12    502
## 8 UA       32252     0   11    483
## 9 MQ       14382    -3    9    486
## 10 VX       2991     0    7    653

f1 %>% group_by(origin,carrier) %>%
  summarize(n=n(),med_d = median(dep_delay),Q3_d = Q3(dep_delay), max_d = max(dep_delay)) %>%
  arrange(desc(Q3_d)) %>% head(10)

## # A tibble: 10 x 6
## # Groups:   origin [3]
##   origin carrier     n med_d   Q3_d max_d
##   <fct>  <fct>   <int> <dbl> <dbl> <dbl>
## 1 EWR     00        3     4   67.5   131
## 2 EWR     EV        23565   -1   26    443
## 3 LGA     EV        4769    -2   22    473
## 4 JFK     9E        8126    -1   20    430
## 5 JFK     EV        803     -2   19    536
## 6 EWR     WN        3487     2   18    440
## 7 LGA     WN        3410     1   18    471
## 8 LGA     F9        388      0  17.2   853
## 9 EWR     MQ        1156    -2   17    381
## 10 LGA    FL        1832     1   16    602

f1 %>% group_by(dest,carrier) %>%
  summarize(n=n(),med_d = median(dep_delay),Q3_d = Q3(dep_delay), max_d = max(dep_delay)) %>%
  arrange(desc(Q3_d)) %>% head(10)

## # A tibble: 10 x 6
## # Groups:   dest [10]
##   dest  carrier     n med_d   Q3_d max_d
##   <fct> <fct>   <int> <dbl> <dbl> <dbl>
## 1 STL   UA        2  77.5  116.   155
## 2 DTW   00        2   61    96    131
## 3 TYS   EV        183    8   68.5   285
## 4 PBI   EV        3   50    67.5   85
## 5 ORD   00        1   67    67    67
## 6 RDU   UA        1   60    60    60
## 7 TUL   EV        185    3   53    251
## 8 OKC   EV        184   8.5   51.5   207

```

```
##   9 BHM   EV      175   3    50    325
## 10 CAE   EV      57   10    48    163
```

Summaries of departure delay by date:

```
f1 %>% group_by(month,day) %>%
  summarize(n=n(),med_d = mean(dep_delay),max_d = max(dep_delay)) %>%
  arrange(desc(med_d)) %>% head(10) # what happened on march 8?
```

```
## # A tibble: 10 x 5
## # Groups: month [7]
##   month day n med_d max_d
##   <dbl> <dbl> <int> <dbl> <dbl>
## 1     3     8   461  79.5  470
## 2     7     1   505  58.1  363
## 3     7    10   471  56.6  576
## 4     9     2   438  53.7  696
## 5    12     5   458  52.2  896
## 6     5    23   453  51.5  410
## 7     4    19   511  50.4  812
## 8     9    12   444  50.4  602
## 9     6    13   469  50.3  388
## 10    7    22   476  49.9  898
```

Summaries of departure delay by precipitation:

```
f1 %>% mutate(haveprecip = factor(precip>0)) %>% group_by(haveprecip) %>%
  summarize(n=n(),med_d = median(dep_delay),Q3_d = Q3(dep_delay), max_d = max(dep_delay)) %>%
  arrange(desc(med_d)) %>% head(10)

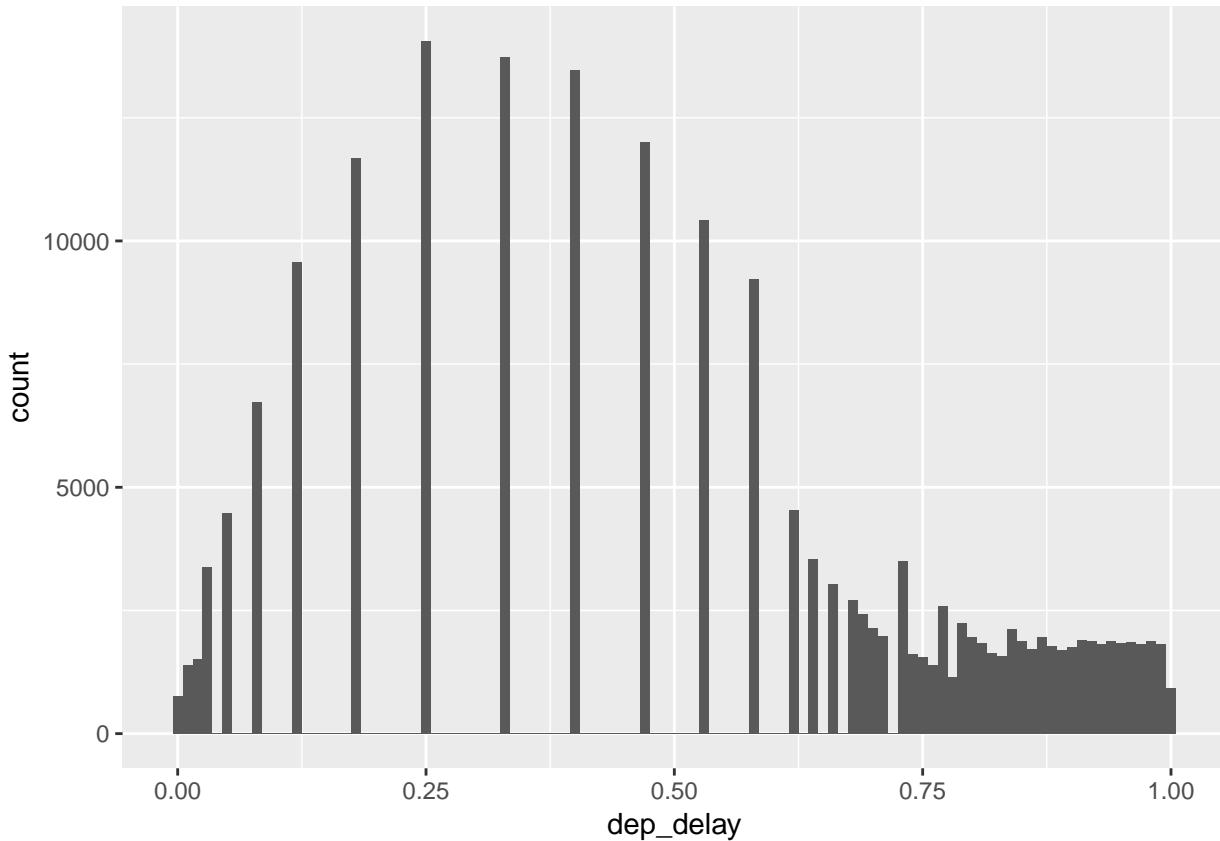
## # A tibble: 2 x 5
##   haveprecip n med_d   Q3_d max_d
##   <fct>     <int> <dbl> <dbl> <dbl>
## 1 TRUE       11804     5     41    853
## 2 FALSE      172512    -2     9   1301
```

## What can we predict?

Extremes seem to be caused by phenomena not in our data, such as snow storms, mechanical breakdowns (?), etc.

Perhaps we should map these extremes to something less extreme. Consider mapping to quantiles of the standard normal (like grading departure delays on a “curve”), or mapping to ranks. We will scale the ranks by  $n + 1$  to get the empirical quantiles, which will be comparable to those in the test dataset.

```
#f1 <- f1 %>% mutate(dep_delay = qnorm(dep_delay)$x)
den <- nrow(f1)+1
f1 <- f1 %>% mutate(dep_delay = rank(dep_delay)/den)
ggplot(f1,aes(x=dep_delay)) + geom_histogram(binwidth=.01)
```



## More data wrangling

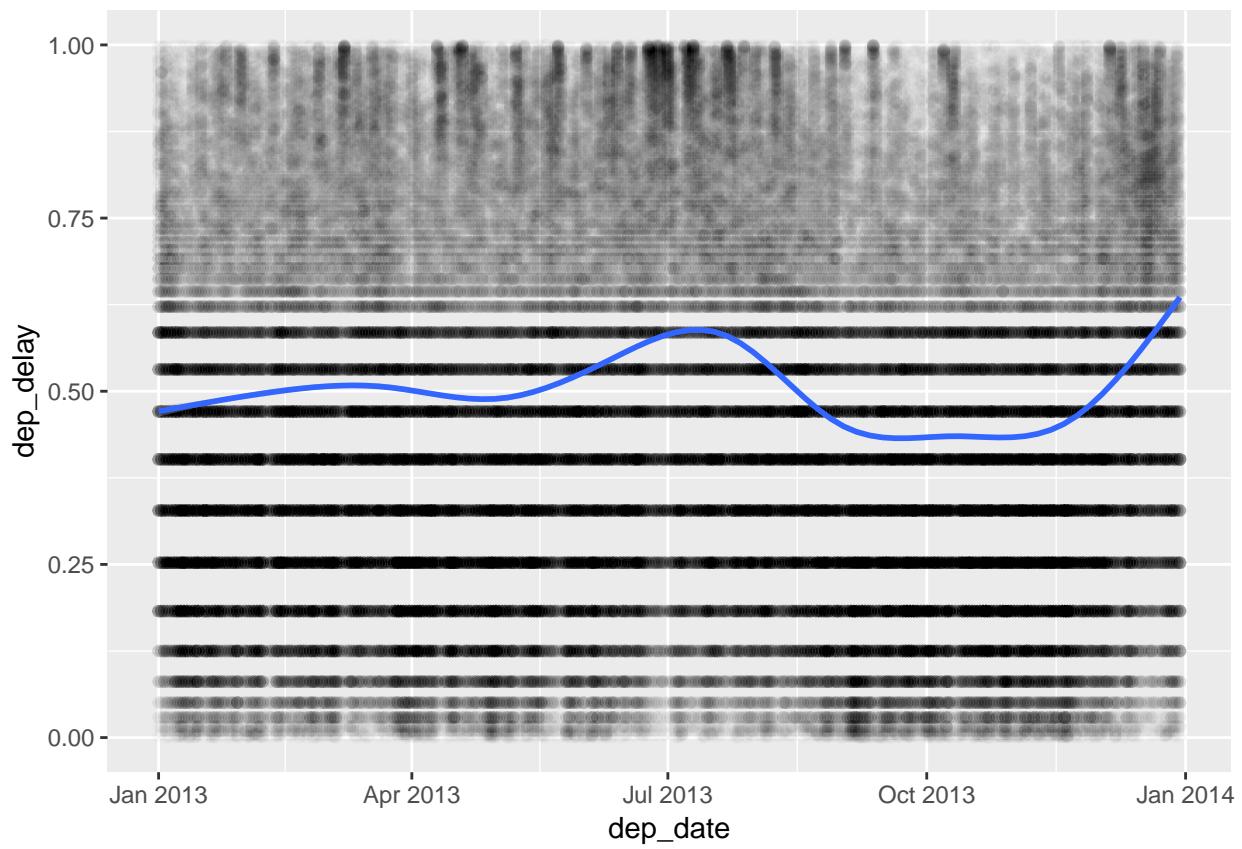
- Convert year/month/day to a date object.
- Remove `dep_time`, `arr_time` and `arr_delay`. If we are interested in predicting departure delay, these are not known to us. Also, they may be associated with `dep_delay`, but if so the causal effect is likely in reverse.
- Remove `sched_arr_time` (basically departure time + air time), `tailnum` (4000 planes), `flight` (flight number), `name` (captured by `dest`), `air_time` (highly correlated with `distance`), `hour` and `minute` (in `sched_dep_time`), `time_hour` (same as `sched_dep_time`), `tz`, `dst`, `tzone` (time zone of destination),
- Replace numeric `precip` with indicator of precipitation/none.

```
library(lubridate)
fl <- fl %>%
  mutate(dep_date = make_date(year.x,month,day)) %>%
  select(-year.x,-month,-day,-dep_time,-arr_time,-arr_delay,
         -sched_arr_time,-tailnum,-flight,-name,-air_time,
         -hour,-minute,-time_hour,-tz,-dst,-tzone) %>%
  mutate(precip = as.numeric(precip>0))
```

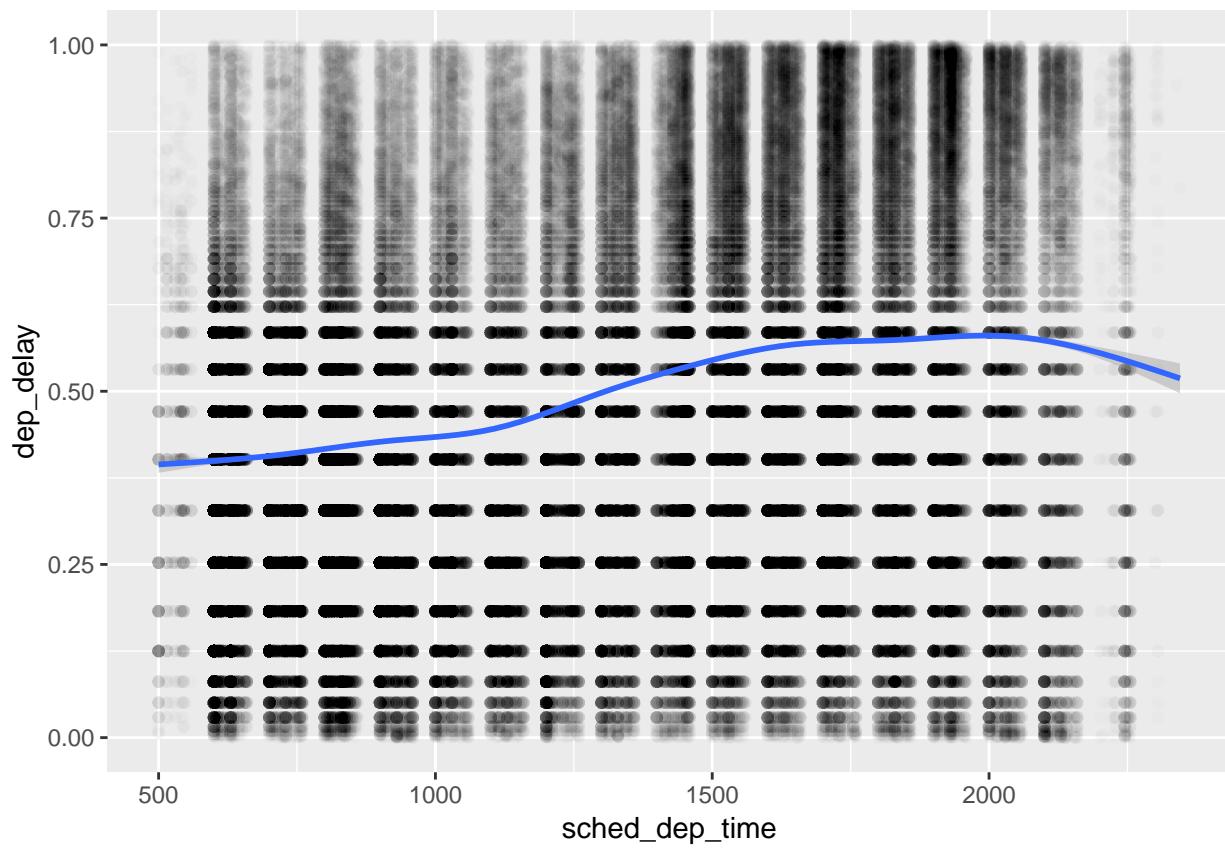
## Associations between `dep_delay` and quantitative predictors

Here we look at associations between `dep_delay` rank and other variables one-at-a-time. The presentation here is not exhaustive

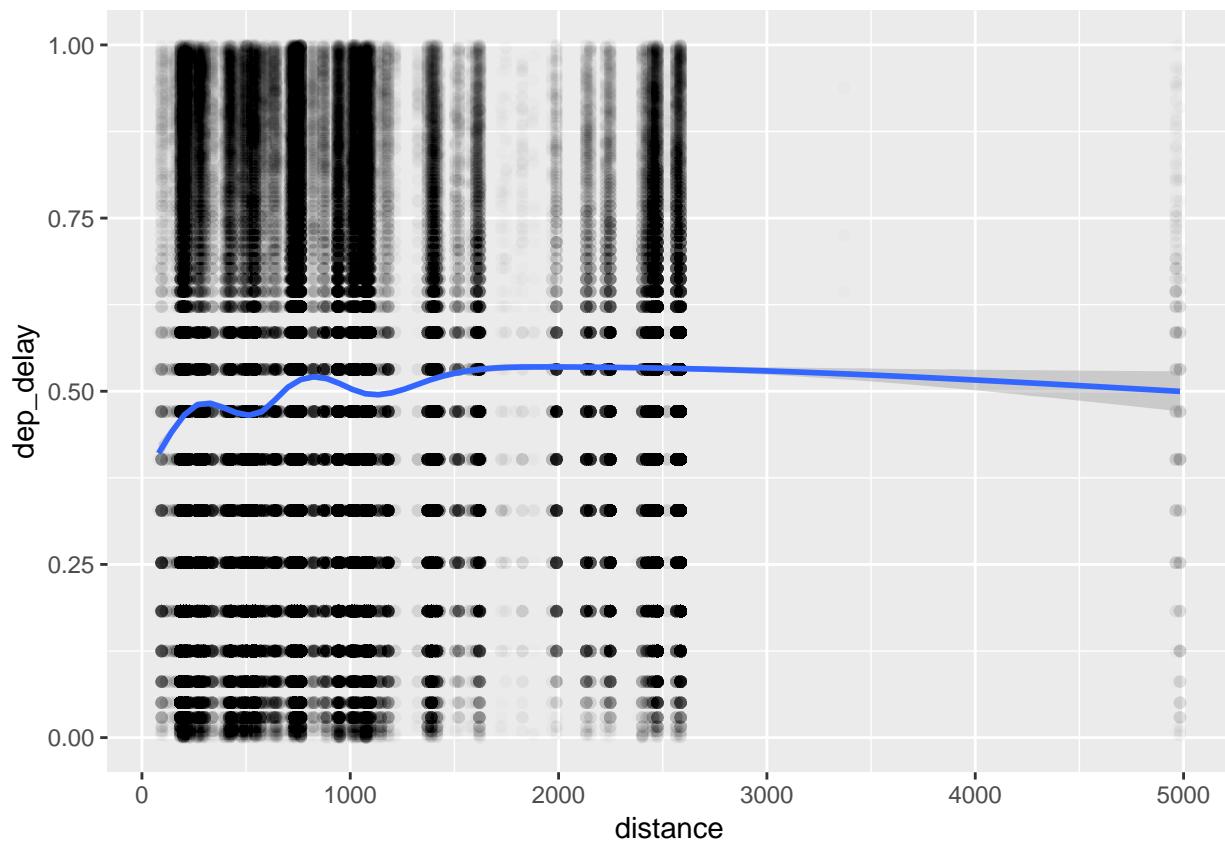
```
ggplot(fl,aes(x=dep_date,y=dep_delay)) + geom_point(alpha=.01) + geom_smooth()
```

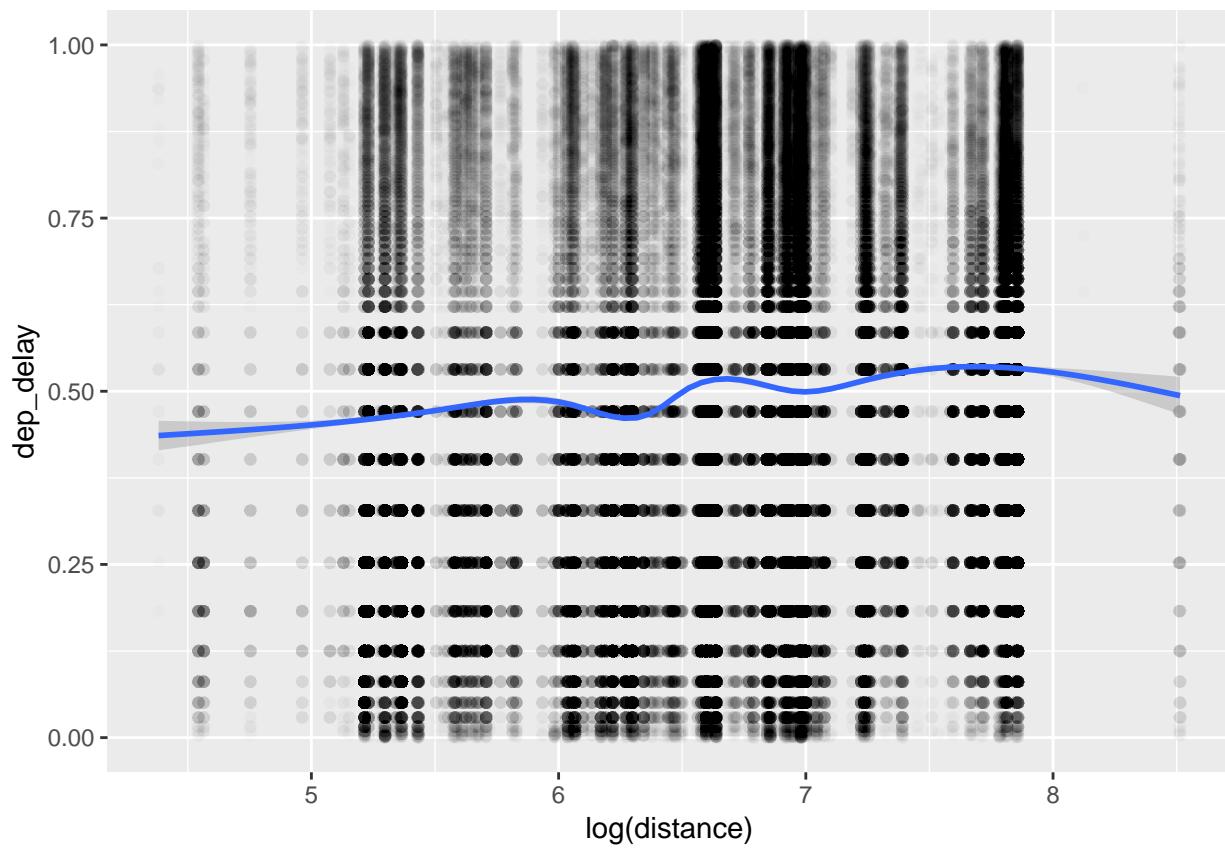


```
# Definitely non-linear. High in summer, low in fall. Not sure about winter. Looks like
# some sort of event around the end of 2013, but could just be an end effect.
ggplot(f1,aes(x=sched_dep_time,y=dep_delay)) + geom_point(alpha=0.01) + geom_smooth()
```

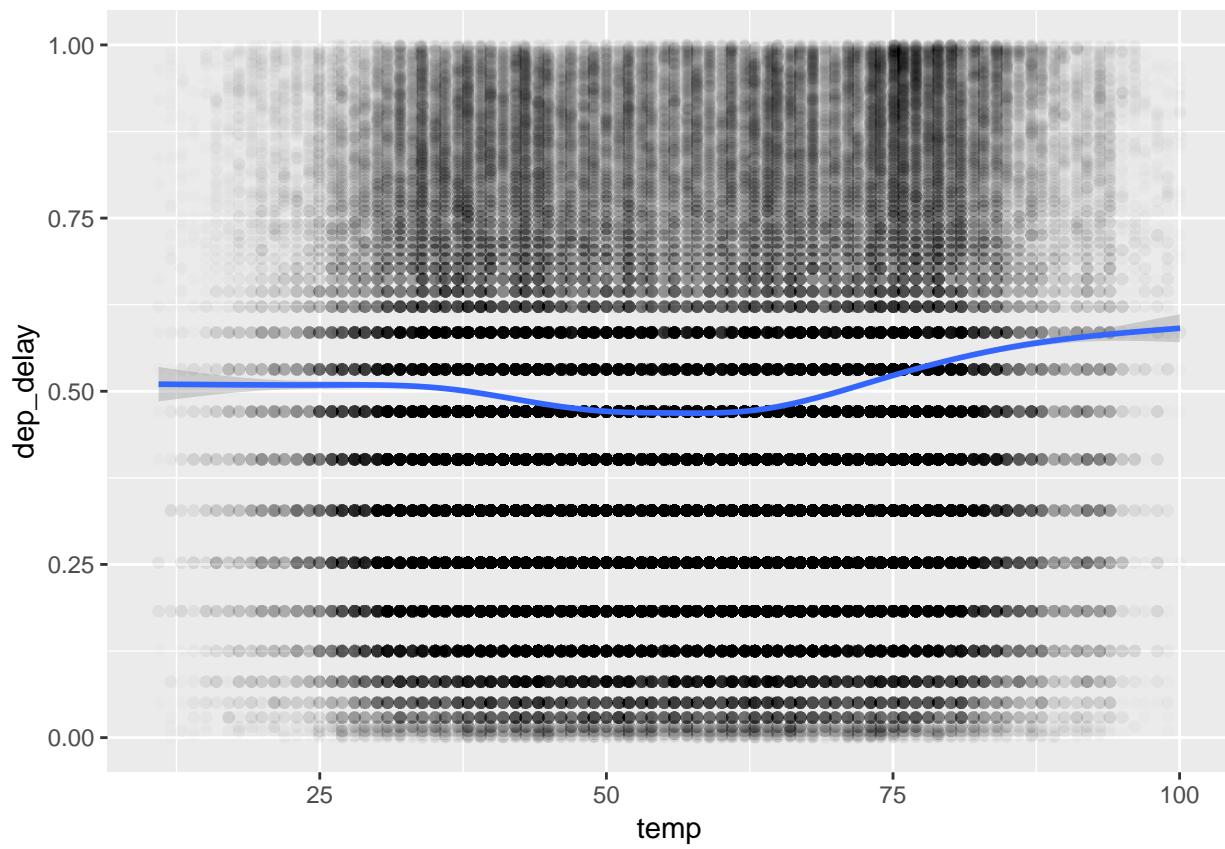


```
# delays increase throughout the day  
ggplot(f1,aes(x=distance,y=dep_delay)) + geom_point(alpha=0.01) + geom_smooth()
```

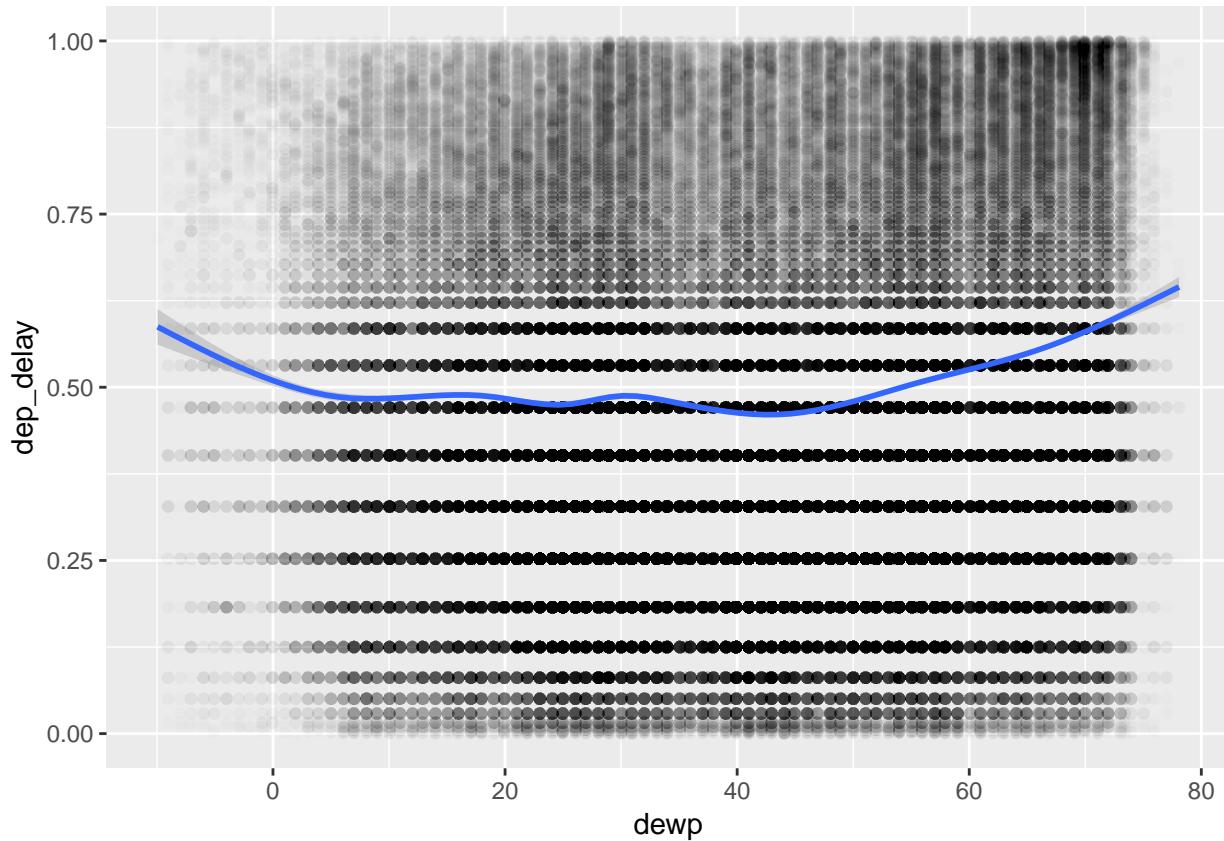




```
# increases with distance -- use log distance
fl <- mutate(f1, logdistance = log(distance)) %>% select(-distance)
ggplot(f1, aes(x=temp, y=dep_delay)) + geom_point(alpha=0.01) + geom_smooth()
```



```
# delays when too hot or too cold  
ggplot(f1,aes(x=dewp,y=dep_delay)) + geom_point(alpha=0.01) + geom_smooth()
```



```
# similar to temp
# Etc.
# Replace alt with log(alt)
fl <- mutate(f1, logalt = log(alt)) %>% select(-alt)
```

We will likely need to include non-linear terms in the quantitative predictors in our models.

### Split training set in two for tuning

- We have lots of data.
- Methods like cross validation can be used to select tuning parameters, but approaches like boosting are best with a training/test set.
- Evaluate all learning methods on the test set.

```
set.seed(123)
tr_size <- ceiling(2*nrow(f1)/3)
train <- sample(1:nrow(f1), size=tr_size)
f1_tr <- f1[train,]
f1_te <- f1[-train,]

# baseline to compare learning methods to:
var_dd <- var(f1_te$dep_delay)
var_dd

## [1] 0.08311941
```

## Learning methods

In the interest of time I'll just consider gam and boosting.

The first fit is a gam with default df for smooths of quantitative variables. As expected, lat, lon and alt of the destination contribute very little (not shown) and are removed.

```
library(gam)
form <- formula(dep_delay ~ s(dep_date) + s(sched_dep_time) + carrier + origin + dest + s(logdistance) +
                  s(temp) + s(dewp) + s(humid) + s(wind_dir) + s(wind_speed) + precip + s(visib))
gam_fit <- gam(form, data=fl_tr,family=gaussian)
summary(gam_fit)

##
## Call: gam(formula = form, family = gaussian, data = fl_tr)
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.848656 -0.209769  0.004745  0.211186  0.763180
##
## (Dispersion Parameter for gaussian family taken to be 0.07)
##
## Null Deviance: 10216.89 on 122877 degrees of freedom
## Residual Deviance: 8585.173 on 122725 degrees of freedom
## AIC: 22023.21
##
## Number of Local Scoring Iterations: 2
##
## Anova for Parametric Effects
##                               Df Sum Sq Mean Sq F value    Pr(>F)
## s(dep_date)                 1    0.3    0.26   3.757 0.0525882 .
## s(sched_dep_time)          1  572.4  572.44 8183.025 < 2.2e-16 ***
## carrier                      15   340.9   22.73  324.885 < 2.2e-16 ***
## origin                       2    48.4   24.18  345.701 < 2.2e-16 ***
## dest                         98   90.6    0.92   13.211 < 2.2e-16 ***
## s(logdistance)               1     1.3    1.35   19.231 1.159e-05 ***
## s(temp)                      1     5.1    5.06   72.302 < 2.2e-16 ***
## s(dewp)                      1   287.5  287.53 4110.195 < 2.2e-16 ***
## s(humid)                     1   37.0   36.99  528.811 < 2.2e-16 ***
## s(wind_dir)                  1     9.4    9.36  133.854 < 2.2e-16 ***
## s(wind_speed)                1   22.9   22.92  327.609 < 2.2e-16 ***
## precip                        1   16.2   16.15  230.903 < 2.2e-16 ***
## s(visib)                      1     0.9    0.89   12.733 0.0003594 ***
## Residuals                   122725 8585.2    0.07
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##                               Npar Df Npar F    Pr(F)
## (Intercept)                  3 491.42 < 2e-16 ***
## s(dep_date)                  3 296.70 < 2e-16 ***
## s(sched_dep_time)           3  0.37  0.77273
## s(temp)                      3 255.55 < 2e-16 ***
```

```

## s(dewp)           3  97.58 < 2e-16 ***
## s(humid)          3   3.33  0.01872 *
## s(wind_dir)        3  51.39 < 2e-16 ***
## s(wind_speed)      3  12.50  3.6e-08 ***
## precip
## s(visib)          3  26.86 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
gam_pred <- predict(gam_fit,newdata=fl_te)
mse_gam <- mean((fl_te$dep_delay-gam_pred)^2)
mse_gam

## [1] 0.07047458
abs(mse_gam - var_dd)/var_dd

## [1] 0.1521285

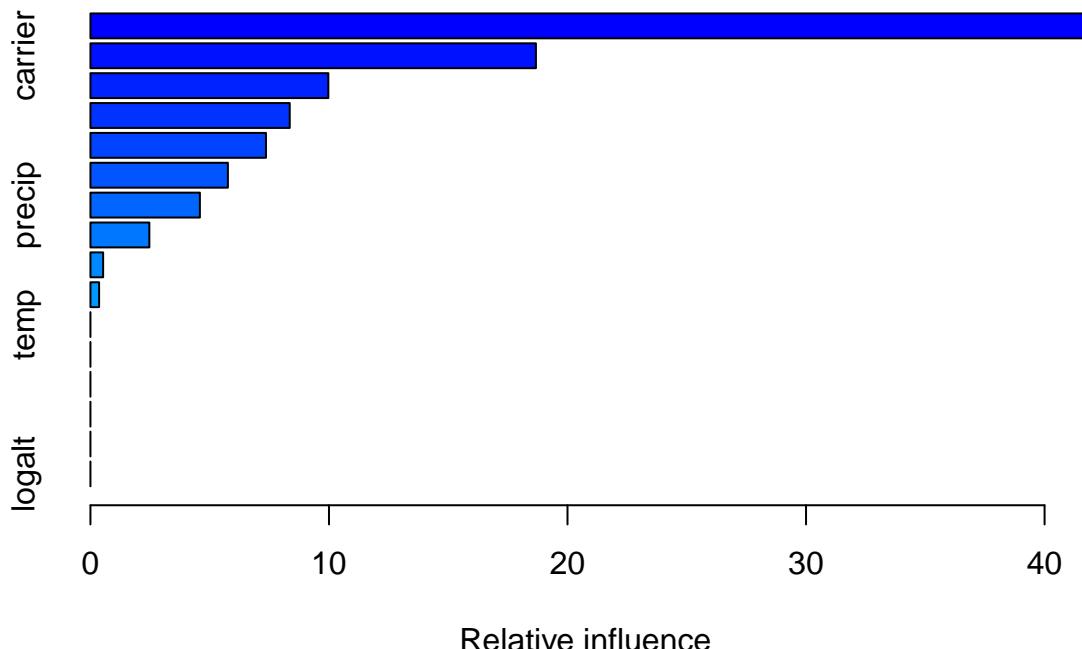
```

The more trees and slower learning, the better a boosted model will do. I only have time to go as far as 1000 trees and shrinkage 0.01.

```

library(gbm)
dep_date_numeric <- as.numeric(fl_tr$dep_date)
dep_date_numeric <- dep_date_numeric - mean(dep_date_numeric)
fl_tr_tem <- mutate(fl_tr,dep_date = dep_date_numeric)
gbm_fit <- gbm(dep_delay ~ .,data=fl_tr_tem,distribution="gaussian",
                 n.trees = 1000, shrinkage = 0.01)
summary(gbm_fit)

```



```

##                   var    rel.inf
## sched_dep_time sched_dep_time 41.9210129
## carrier         carrier 18.6758728
## dep_date        dep_date  9.9734373
## humid           humid  8.3543544
## dewp            dewp  7.3585086

```

```

## dest          dest  5.7623968
## precip       precip 4.5884783
## origin       origin 2.4700180
## wind_speed   wind_speed 0.5323563
## logdistance  logdistance 0.3635646
## temp          temp  0.0000000
## wind_dir     wind_dir 0.0000000
## visib         visib 0.0000000
## lat           lat   0.0000000
## lon           lon   0.0000000
## logalt        logalt 0.0000000

#
dep_date_numeric <- as.numeric(fl_te$dep_date)
dep_date_numeric <- dep_date_numeric - mean(dep_date_numeric)
fl_te_tem <- mutate(fl_te,dep_date = dep_date_numeric)
#
gbm_pred <- predict(gbm_fit,newdata=fl_te_tem,n.trees = 1000)
mse_gbm <- mean((fl_te$dep_delay-gbm_pred)^2)
mse_gbm

## [1] 0.07206016
abs(mse_gbm - var_dd)/var_dd

## [1] 0.1330526

```