Preliminary Exploration

Load the data, and take a look.

```
train <- read.csv("../data/train.csv")
test <- read.csv("../data/test.csv")
dim(train)
```

```
## [1] 10506    12
```

```
dim(test)
```

```
## [1] 116293     11
```

```
head(train)
```

```
##         Date                                              Address
## 1 2007-05-29 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 2 2007-05-29 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 3 2007-05-29  6200 North Mandell Avenue, Chicago, IL 60646, USA
## 4 2007-05-29    7900 West Foster Avenue, Chicago, IL 60656, USA
## 5 2007-05-29    7900 West Foster Avenue, Chicago, IL 60656, USA
## 6 2007-05-29   1500 West Webster Avenue, Chicago, IL 60614, USA
##                   Species Block          Street Trap
## 1 CULEX PIPIENS/RESTUANS    41  N OAK PARK AVE T002
## 2         CULEX RESTUANS    41  N OAK PARK AVE T002
## 3         CULEX RESTUANS    62   N MANDELL AVE T007
## 4 CULEX PIPIENS/RESTUANS    79    W FOSTER AVE T015
## 5         CULEX RESTUANS    79    W FOSTER AVE T015
## 6         CULEX RESTUANS    15   W WEBSTER AVE T045
##              AddressNumberAndStreet Latitude Longitude AddressAccuracy
## 1 4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 2 4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 3  6200  N MANDELL AVE, Chicago, IL 41.99499 -87.76928               9
## 4   7900  W FOSTER AVE, Chicago, IL 41.97409 -87.82481               8
## 5   7900  W FOSTER AVE, Chicago, IL 41.97409 -87.82481               8
## 6  1500  W WEBSTER AVE, Chicago, IL 41.92160 -87.66645               8
##   NumMosquitos WnvPresent
## 1            1          0
## 2            1          0
## 3            1          0
## 4            1          0
## 5            4          0
## 6            2          0
```

```
head(test, 10)
```

```
##    Id       Date                                              Address
## 1   1 2008-06-11 4100 North Oak Park Avenue, Chicago, IL 60634, USA
```

```
## 2   2 2008-06-11 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 3   3 2008-06-11 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 4   4 2008-06-11 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 5   5 2008-06-11 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 6   6 2008-06-11 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 7   7 2008-06-11 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 8   8 2008-06-11 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 9   9 2008-06-11  6200 North Mandell Avenue, Chicago, IL 60646, USA
## 10 10 2008-06-11  6200 North Mandell Avenue, Chicago, IL 60646, USA
##                   Species Block          Street Trap
## 1  CULEX PIPIENS/RESTUANS    41  N OAK PARK AVE T002
## 2          CULEX RESTUANS    41  N OAK PARK AVE T002
## 3           CULEX PIPIENS    41  N OAK PARK AVE T002
## 4         CULEX SALINARIUS    41  N OAK PARK AVE T002
## 5          CULEX TERRITANS    41  N OAK PARK AVE T002
## 6           CULEX TARSALIS    41  N OAK PARK AVE T002
## 7        UNSPECIFIED CULEX    41  N OAK PARK AVE T002
## 8          CULEX ERRATICUS    41  N OAK PARK AVE T002
## 9  CULEX PIPIENS/RESTUANS    62   N MANDELL AVE T007
## 10         CULEX RESTUANS    62   N MANDELL AVE T007
##               AddressNumberAndStreet Latitude Longitude AddressAccuracy
## 1  4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 2  4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 3  4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 4  4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 5  4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 6  4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 7  4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 8  4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 9   6200  N MANDELL AVE, Chicago, IL 41.99499 -87.76928               9
## 10  6200  N MANDELL AVE, Chicago, IL 41.99499 -87.76928               9
```

```
str(train, strict.width="cut")
```

```
## 'data.frame':    10506 obs. of  12 variables:
##  $ Date                   : Factor w/ 95 levels "2007-05-29","2007-06-05",..
##  $ Address                : Factor w/ 138 levels "1000 East 67th Street, "..
##  $ Species                : Factor w/ 7 levels "CULEX ERRATICUS",..: 3 4 4..
##  $ Block                  : int  41 41 62 79 79 15 25 11 11 11 ...
##  $ Street                 : Factor w/ 128 levels " E 105TH ST",..: 33 33 2..
##  $ Trap                   : Factor w/ 136 levels "T001","T002",..: 2 2 7 1..
##  $ AddressNumberAndStreet: Factor w/ 138 levels "1000  E 67TH ST, Chicag"..
##  $ Latitude               : num  42 42 42 42 42 ...
##  $ Longitude              : num  -87.8 -87.8 -87.8 -87.8 -87.8 ...
##  $ AddressAccuracy        : int  9 9 9 8 8 8 8 8 8 8 ...
##  $ NumMosquitos           : int  1 1 1 1 4 2 1 1 2 1 ...
##  $ WnvPresent             : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
str(test, strict.width="cut")
```

```
## 'data.frame':    116293 obs. of  11 variables:
##  $ Id                     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Date                   : Factor w/ 95 levels "2008-06-11","2008-06-17",..
```

```
##  $ Address               : Factor w/ 151 levels "1000 East 67th Street, "..
##  $ Species               : Factor w/ 8 levels "CULEX ERRATICUS",..: 3 4 2..
##  $ Block                 : int  41 41 41 41 41 41 41 41 62 62 ...
##  $ Street                : Factor w/ 139 levels " E 105TH ST",..: 37 37 3..
##  $ Trap                  : Factor w/ 149 levels "T001","T002",..: 2 2 2 2..
##  $ AddressNumberAndStreet: Factor w/ 151 levels "1000  E 67TH ST, Chicag"..
##  $ Latitude              : num  42 42 42 42 42 ...
##  $ Longitude             : num  -87.8 -87.8 -87.8 -87.8 -87.8 ...
##  $ AddressAccuracy       : int  9 9 9 9 9 9 9 9 9 9 ...
```

And there is no missing value.

```
any(is.na(train) == TRUE)
```

```
## [1] FALSE
```

```
any(is.na(test) == TRUE)
```

```
## [1] FALSE
```

Feature Engineering

Date

```
train$Date <- as.Date(train$Date)
test$Date <- as.Date(test$Date)
```

Naturally, we create four features Year, Month, Week and Weekday based on Date.

```
library(lubridate)
## train set
train$Year <- as.integer(year(train$Date))
train$Month <- factor(months(train$Date),
                      levels=c("May", "June", "July", "August", "September", "October"))
train$Week <- week(train$Date)
train$Weekday <- factor(weekdays(train$Date),
                        levels=c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday"))
## test set
test$Year <- as.integer(year(test$Date))
test$Month <- factor(months(test$Date),
                     levels=c("June", "July", "August", "September", "October"))
test$Week <- week(test$Date)
test$Weekday <- factor(weekdays(test$Date),
                       levels=c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday"))
## take a look
head(train[,c("Date", "Year", "Month", "Week", "Weekday")])
```

```
##         Date Year Month Week Weekday
## 1 2007-05-29 2007   May   22 Tuesday
```

```
## 2 2007-05-29 2007   May   22 Tuesday
## 3 2007-05-29 2007   May   22 Tuesday
## 4 2007-05-29 2007   May   22 Tuesday
## 5 2007-05-29 2007   May   22 Tuesday
## 6 2007-05-29 2007   May   22 Tuesday
```

```
head(test[,c("Date", "Year", "Month", "Week", "Weekday")])
```

```
##         Date Year Month Week   Weekday
## 1 2008-06-11 2008  June   24 Wednesday
## 2 2008-06-11 2008  June   24 Wednesday
## 3 2008-06-11 2008  June   24 Wednesday
## 4 2008-06-11 2008  June   24 Wednesday
## 5 2008-06-11 2008  June   24 Wednesday
## 6 2008-06-11 2008  June   24 Wednesday
```

```
str(train[,c("Date", "Year", "Month", "Week", "Weekday")])
```

```
## 'data.frame':    10506 obs. of  5 variables:
##  $ Date   : Date, format: "2007-05-29" "2007-05-29" ...
##  $ Year   : int  2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 ...
##  $ Month  : Factor w/ 6 levels "May","June","July",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Week   : num  22 22 22 22 22 22 22 22 22 22 ...
##  $ Weekday: Factor w/ 5 levels "Monday","Tuesday",..: 2 2 2 2 2 2 2 2 2 2 ...
```

```
str(test[,c("Date", "Year", "Month", "Week", "Weekday")])
```

```
## 'data.frame':    116293 obs. of  5 variables:
##  $ Date   : Date, format: "2008-06-11" "2008-06-11" ...
##  $ Year   : int  2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
##  $ Month  : Factor w/ 5 levels "June","July",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Week   : num  24 24 24 24 24 24 24 24 24 24 ...
##  $ Weekday: Factor w/ 5 levels "Monday","Tuesday",..: 3 3 3 3 3 3 3 3 3 3 ...
```

Species

As a factor, Species has 7 levels in train set, but 8 levels in test set. The extra one in test set is "UNSPECIFIED CULEX".

```
table(train$Species)
```

```
##
##        CULEX ERRATICUS          CULEX PIPIENS CULEX PIPIENS/RESTUANS
##                      1                   2699                   4752
##         CULEX RESTUANS       CULEX SALINARIUS         CULEX TARSALIS
##                   2740                     86                      6
##        CULEX TERRITANS
##                    222
```

```
table(test$Species)
```

```
##
##        CULEX ERRATICUS          CULEX PIPIENS CULEX PIPIENS/RESTUANS
##                  14345                  14521                  15359
##         CULEX RESTUANS       CULEX SALINARIUS         CULEX TARSALIS
##                  14670                  14355                  14347
##        CULEX TERRITANS      UNSPECIFIED CULEX
##                  14351                  14345
```

The "UNSPECIFIED CULEX" does not appear in the train set. We need a mechanism to specify it to another level, which is one of the 7 levels in train set. Right now we use the following strategy.

```
test[test$Species=="UNSPECIFIED CULEX", "Species"] <- "CULEX ERRATICUS"
```

NumMosquitos

These train results are organized in such a way that when the number of mosquitos exceed 50, they are split into another record (another row in the dataset), such that the number of mosquitos are capped at 50. For the test set, it is the same, while NumMosquitos does not appear as a predictor.
This section we combine the same record into one row, instead of several rows since its NumMosquitos > 50.

```
library(plyr)
```

```
##
## Attaching package: 'plyr'
##
## The following object is masked from 'package:lubridate':
##
##     here
```

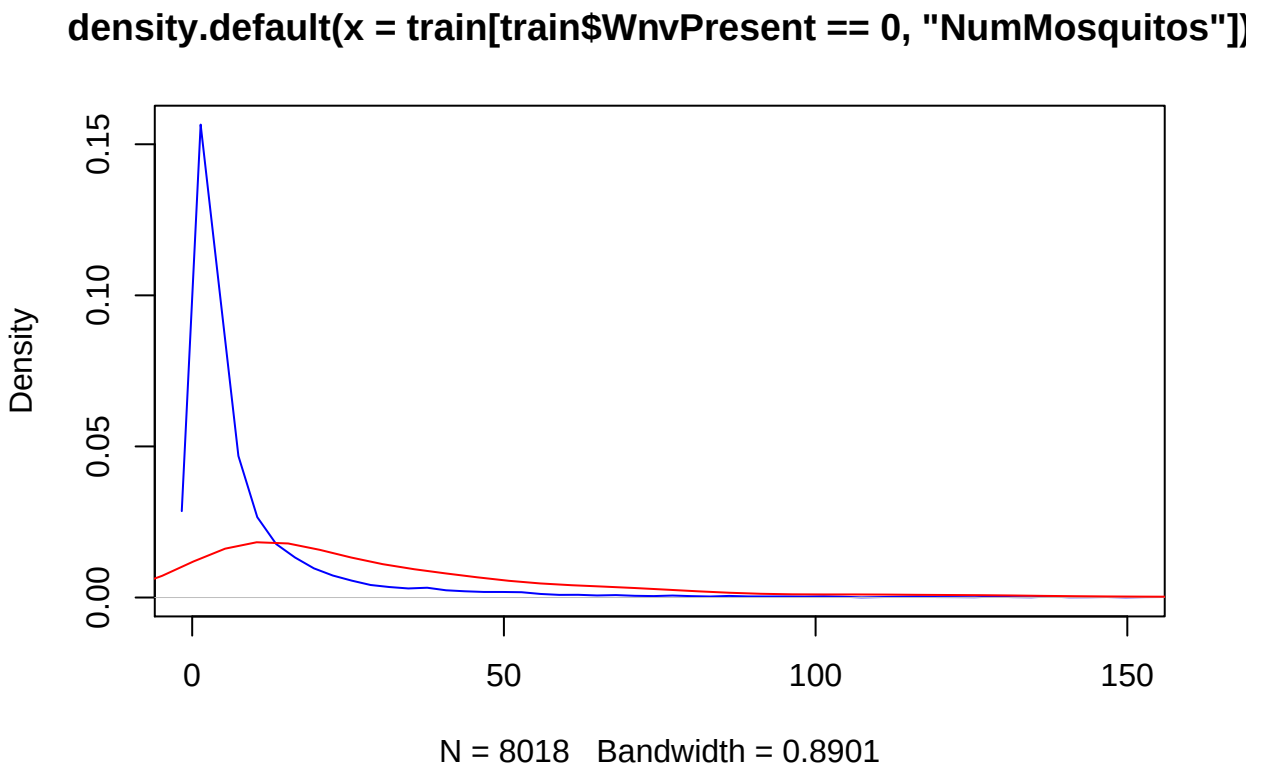
```
train <- ddply(train,
               .(Date, Address, Species, Block, Street, Trap,
                 AddressNumberAndStreet, Latitude, Longitude,
                 AddressAccuracy, Year, Month, Week, Weekday),
               summarize,
               NumMosquitos = sum(NumMosquitos),
               WnvPresent = as.integer(as.logical(sum(WnvPresent)))
               )
test[,"NumMosquitos"] <- 1
NumMosquitosCount <- ddply(test,
                           .(Date, Address, Species, Block, Street, Trap,
                             AddressNumberAndStreet, Latitude, Longitude,
                             AddressAccuracy, Year, Month, Week, Weekday),
                           summarize,
                           NumMosquitos = sum(NumMosquitos)
                           )[,"NumMosquitos"]
test[,"NumMosquitos"] <- rep(NumMosquitosCount, NumMosquitosCount)
## change scale
est.num <- function(num){
    n1 <- (num-1)*50
```

```
    n2 <- num*50
    result <- median(train[train$NumMosquitos>=n1 & train$NumMosquitos<=n2, "NumMosquitos"])
    return(result)
}
median.num <- sapply(1:26, est.num)
median.num[18] <- 870
median.num[23:26] <- c(1120, 1170, 1220, 1270)
test$NumMosquitos <- median.num[test$NumMosquitos]
summary(test$NumMosquitos)
```

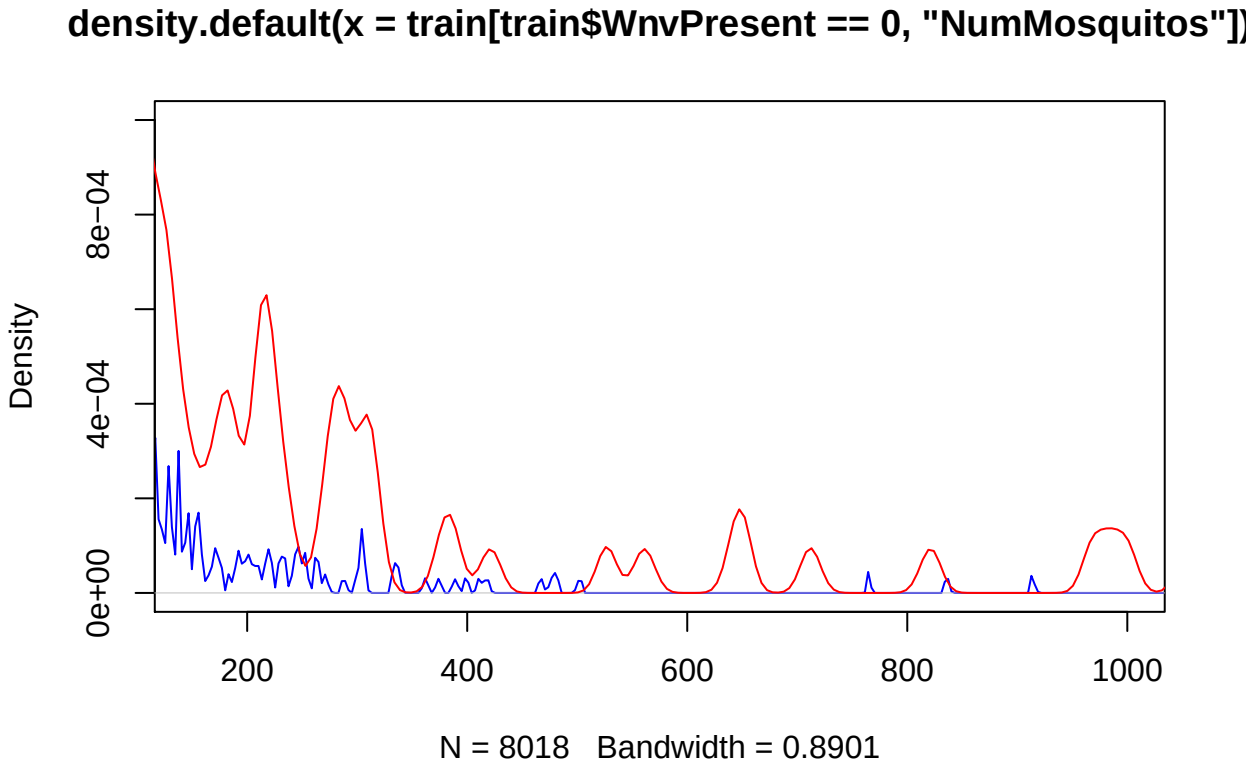
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    3.00    3.00    3.00   22.32   64.00 1270.00
```

20394->20416 79229->79254 Then we compare the density of NumMosquitos with $WnvPresent = 0 or 1$

```
n0 <- length(train[train$WnvPresent==0,])
n1 <- length(train[train$WnvPresent==1,])
weight0 <- rep(1/(n0+n1), n0)
weight1 <- rep(1/(n0+n1), n1)
density0 <- density(train[train$WnvPresent==0, "NumMosquitos"])
density1 <- density(train[train$WnvPresent==1, "NumMosquitos"])
plot(density0, col="blue", xlim=c(0, 150))
lines(density1, col="red")
```




```
plot(density0, col="blue", xlim=c(150,1000), ylim=c(0, 0.001))
lines(density1, col="red")
```



Trap

First, check out the names of all traps.

```
unique(train$Trap)
```

```
##   [1] T002  T007  T015  T045  T046  T048  T049  T050  T054  T086  T091
##  [12] T094  T096  T129  T143  T148  T153  T159  T009  T011  T016  T019
##  [23] T025  T028  T031  T033  T089  T090  T092  T135  T141  T142  T145
##  [34] T146  T147  T149  T150  T151  T152  T154  T158  T162  T218  T220
##  [45] T001  T003  T006  T008  T012  T034  T037  T040  T043  T047  T051
##  [56] T085  T088  T161  T219  T013  T014  T018  T030  T084  T144  T160
##  [67] T005  T017  T044  T095  T004  T035  T036  T039  T060  T061  T062
##  [78] T065  T066  T067  T069  T070  T071  T073  T074  T075  T076  T077
##  [89] T079  T080  T081  T082  T083  T114  T155  T063  T115  T138  T200
## [100] T206  T209  T212  T215  T107  T128  T072  T078  T097  T099  T100
## [111] T102  T103  T027  T156  T157  T221  T900  T903  T222  T223  T225
## [122] T227  T224  T226  T229  T230  T228  T232  T231  T235  T233  T236
## [133] T237  T238  T094B T054C
## 136 Levels: T001 T002 T003 T004 T005 T006 T007 T008 T009 T011 T012 ... T903
```

```
unique(test$Trap)
```

```
##   [1] T002  T007  T015  T045  T046  T048  T049  T050  T054  T086  T091
##  [12] T094  T096  T129  T143  T148  T153  T159  T009  T011  T016  T019
##  [23] T025  T028  T031  T033  T089  T090  T092  T135  T141  T142  T145
##  [34] T146  T147  T149  T150  T151  T152  T154  T158  T162  T218  T220
```

```
##  [45] T001  T003  T006  T008  T012  T034  T037  T040  T043  T047  T051
##  [56] T085  T088  T161  T219  T013  T014  T018  T030  T084  T144  T160
##  [67] T005  T017  T044  T095  T004  T035  T036  T039  T060  T061  T062
##  [78] T065  T066  T067  T069  T070  T071  T073  T074  T075  T076  T077
##  [89] T079  T080  T081  T082  T083  T114  T155  T063  T115  T138  T200
## [100] T206  T209  T212  T215  T107  T128  T072  T078  T097  T099  T100
## [111] T102  T103  T027  T156  T157  T221  T900  T903  T090A T090B T090C
## [122] T200A T128A T200B T218A T218C T218B T222  T223  T225  T227  T224
## [133] T226  T229  T230  T228  T231  T232  T002A T002B T233  T234  T235
## [144] T236  T237  T238  T065A T094B T054C
## 149 Levels: T001 T002 T002A T002B T003 T004 T005 T006 T007 T008 ... T903
```

Through the observation, we know that initial "T" is shared by all traps' names. The single letter after the 3 digits means this trap is a satellite trap of the main trap with the same 3 digits in the name.
Two new features can be naturally generated from this Trap feature:

- The main trap number (TrapNumber): the 3 digits;
- Main or satellite trap (TrapMS): the single letter after digits in Trap. If it is the main trap, there is no single letter. We label it "M".

```
## train set
train$TrapNumber <- as.integer(substr(as.character(train$Trap), 2, 4))
train$TrapMS <- as.factor(substr(as.character(train$Trap), 5, 5))
levels(train$TrapMS) <- c("M", "B", "C")
## test set
test$TrapNumber <- as.integer(substr(as.character(test$Trap), 2, 4))
test$TrapMS <- as.factor(substr(as.character(test$Trap), 5, 5))
levels(test$TrapMS) <- c("M", "A", "B", "C")
## show
head(train[,c("Trap", "TrapNumber", "TrapMS")])
```

```
##   Trap TrapNumber TrapMS
## 1 T002          2      M
## 2 T002          2      M
## 3 T007          7      M
## 4 T015         15      M
## 5 T015         15      M
## 6 T045         45      M
```

```
head(test[,c("Trap", "TrapNumber", "TrapMS")])
```

```
##   Trap TrapNumber TrapMS
## 1 T002          2      M
## 2 T002          2      M
## 3 T002          2      M
## 4 T002          2      M
## 5 T002          2      M
## 6 T002          2      M
```

Distances to the two weather stations

- Weather station 1: Lat: 41.995 Lon: -87.933

- Weather station 2: Lat: 41.786 Lon: -87.752

```
## Calculate distance in kilometers between two points
## (Unit is km, but it doesn't matter)
## From https://conservationecology.wordpress.com/2013/06/30/distance-between-two-points-in-r/
earth.dist <- function (long1, lat1, long2, lat2)
{
rad <- pi/180
a1 <- lat1 * rad
a2 <- long1 * rad
b1 <- lat2 * rad
b2 <- long2 * rad
dlon <- b2 - a2
dlat <- b1 - a1
a <- (sin(dlat/2))^2 + cos(a1) * cos(b1) * (sin(dlon/2))^2
c <- 2 * atan2(sqrt(a), sqrt(1 - a))
R <- 6371
d <- R * c
return(d)
}
long.stn1 <- -87.933
lat.stn1 <- 41.995
long.stn2 <- -87.752
lat.stn2 <- 41.786
train[, "DisStn1"] <- earth.dist(train$Longitude, train$Latitude,
                                 rep(long.stn1, nrow(train)),
                                 rep(lat.stn1, nrow(train)))
train[, "DisStn2"] <- earth.dist(train$Longitude, train$Latitude,
                                 rep(long.stn2, nrow(train)),
                                 rep(lat.stn2, nrow(train)))
test[, "DisStn1"] <- earth.dist(test$Longitude, test$Latitude,
                                 rep(long.stn1, nrow(test)),
                                 rep(lat.stn1, nrow(test)))
test[, "DisStn2"] <- earth.dist(test$Longitude, test$Latitude,
                                 rep(long.stn2, nrow(test)),
                                 rep(lat.stn2, nrow(test)))
train[, "ClosestStn"] <- ifelse(train$DisStn1 < train$DisStn2, 1, 2)
test[, "ClosestStn"] <- ifelse(test$DisStn1 < test$DisStn2, 1, 2)
summary(train[,19:21])
```

```
##     DisStn1         DisStn2          ClosestStn
##  Min.   : 4.14   Min.   : 0.8572   Min.   :1.000
##  1st Qu.:17.77   1st Qu.:10.3361   1st Qu.:1.000
##  Median :26.17   Median :14.4021   Median :2.000
##  Mean   :26.96   Mean   :14.6058   Mean   :1.709
##  3rd Qu.:37.49   3rd Qu.:19.1911   3rd Qu.:2.000
##  Max.   :49.62   Max.   :26.2772   Max.   :2.000
```

```
summary(test[,19:21])
```

```
##     DisStn1         DisStn2          ClosestStn
##  Min.   : 4.14   Min.   : 0.8572   Min.   :1.000
##  1st Qu.:17.27   1st Qu.:10.3297   1st Qu.:1.000
```

```
##  Median :26.15   Median :15.3222   Median :2.000
##  Mean   :26.72   Mean   :14.7680   Mean   :1.696
##  3rd Qu.:36.21   3rd Qu.:19.5879   3rd Qu.:2.000
##  Max.   :50.42   Max.   :26.2772   Max.   :2.000
```

Save this data set

```
write.csv(train, "../data/train2.csv", row.names=F)
write.csv(test, "../data/test2.csv", row.names=F)
```

Data Exploration

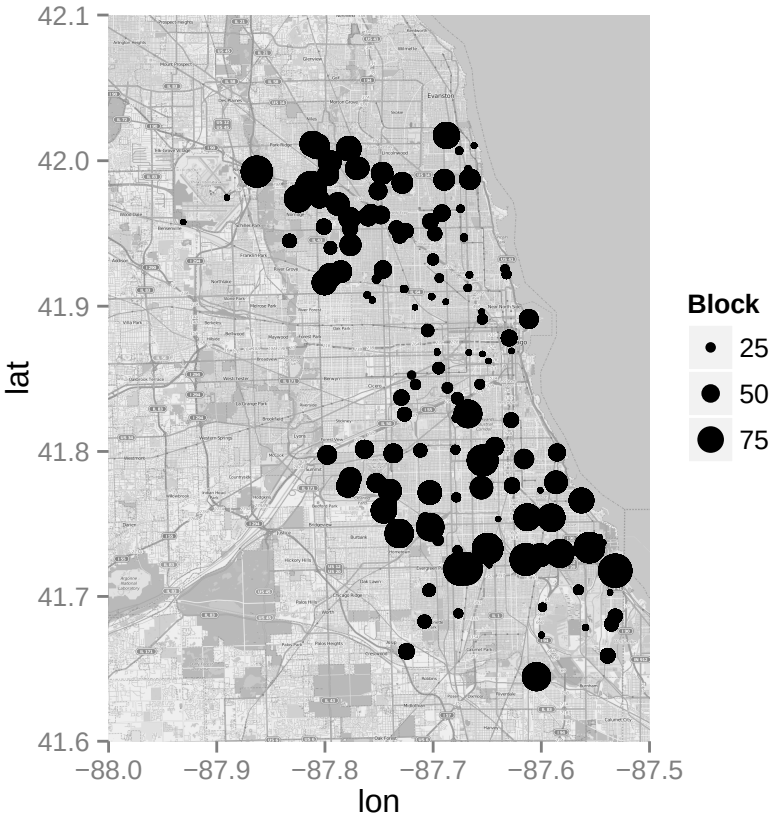
The distribution of blocks

It seems like the distribution of blocks does not have a very regular sequence.

```
library(ggplot2)
library(ggmap)
```

```
## Google Maps API Terms of Service: http://developers.google.com/maps/terms.
## Please cite ggmap if you use it: see citation('ggmap') for details.
```

```
mapdata <- readRDS("../data/mapdata_copyright_openstreetmap_contributors.rds")
ggmap(mapdata) + geom_point(data=train, aes(x=Longitude,y=Latitude, size=Block))
```



Weather data

> *It is believed that hot and dry conditions are more favorable for West Nile virus than cold and wet. We provide you with the dataset from NOAA of the weather conditions of 2007 to 2014, during the months of the tests. (From https://www.kaggle.com/c/predict-west-nile-virus/data)*

Read the weather data

```
weather.data <- read.csv("../data/weather.csv", na.strings=c("M", "-", "", " "))
dim(weather.data)
```

```
## [1] 2944   22
```

```
str(weather.data)
```

```
## 'data.frame':    2944 obs. of  22 variables:
##  $ Station    : int  1 2 1 2 1 2 1 2 1 2 ...
##  $ Date       : Factor w/ 1472 levels "2007-05-01","2007-05-02",..: 1 1 2 2 3 3 4 4 5 5 ...
##  $ Tmax       : int  83 84 59 60 66 67 66 78 66 66 ...
##  $ Tmin       : int  50 52 42 43 46 48 49 51 53 54 ...
##  $ Tavg       : int  67 68 51 52 56 58 58 NA 60 60 ...
##  $ Depart     : int  14 NA -3 NA 2 NA 4 NA 5 NA ...
##  $ DewPoint   : int  51 51 42 42 40 40 41 42 38 39 ...
##  $ WetBulb    : int  56 57 47 47 48 50 50 50 49 50 ...
##  $ Heat       : int  0 0 14 13 9 7 7 NA 5 5 ...
##  $ Cool       : int  2 3 0 0 0 0 0 NA 0 0 ...
##  $ Sunrise    : int  448 NA 447 NA 446 NA 444 NA 443 NA ...
##  $ Sunset     : int  1849 NA 1850 NA 1851 NA 1852 NA 1853 NA ...
##  $ CodeSum    : Factor w/ 97 levels "BCFG BR","BR",..: NA NA 2 3 NA 19 23 NA NA NA ...
##  $ Depth      : int  0 NA 0 NA 0 NA 0 NA 0 NA ...
##  $ Water1     : logi  NA NA NA NA NA NA ...
##  $ SnowFall   : Factor w/ 3 levels "0.0","0.1","  T": 1 NA 1 NA 1 NA 1 NA 1 NA ...
##  $ PrecipTotal: Factor w/ 167 levels "0.00","0.01",..: 1 1 1 1 1 1 167 1 167 167 ...
##  $ StnPressure: num  29.1 29.2 29.4 29.4 29.4 ...
##  $ SeaLevel   : num  29.8 29.8 30.1 30.1 30.1 ...
##  $ ResultSpeed: num  1.7 2.7 13 13.3 11.7 12.9 10.4 10.1 11.7 11.2 ...
##  $ ResultDir  : int  27 25 4 2 7 6 8 7 7 7 ...
##  $ AvgSpeed   : num  9.2 9.6 13.4 13.4 11.9 13.2 10.8 10.4 12 11.5 ...
```

```
head(weather.data)
```

```
##   Station       Date Tmax Tmin Tavg Depart DewPoint WetBulb Heat Cool
## 1       1 2007-05-01   83   50   67     14       51      56    0    2
## 2       2 2007-05-01   84   52   68     NA       51      57    0    3
## 3       1 2007-05-02   59   42   51     -3       42      47   14    0
## 4       2 2007-05-02   60   43   52     NA       42      47   13    0
## 5       1 2007-05-03   66   46   56      2       40      48    9    0
## 6       2 2007-05-03   67   48   58     NA       40      50    7    0
##   Sunrise Sunset CodeSum Depth Water1 SnowFall PrecipTotal StnPressure
## 1     448   1849    <NA>     0     NA      0.0        0.00       29.10
## 2      NA     NA    <NA>    NA     NA     <NA>        0.00       29.18
```

```
## 3     447   1850       BR     0     NA      0.0        0.00       29.38
## 4      NA     NA    BR HZ    NA     NA     <NA>        0.00       29.44
## 5     446   1851     <NA>     0     NA      0.0        0.00       29.39
## 6      NA     NA       HZ    NA     NA     <NA>        0.00       29.46
##   SeaLevel ResultSpeed ResultDir AvgSpeed
## 1    29.82         1.7        27      9.2
## 2    29.82         2.7        25      9.6
## 3    30.09        13.0         4     13.4
## 4    30.08        13.3         2     13.4
## 5    30.12        11.7         7     11.9
## 6    30.12        12.9         6     13.2
```

Notice of the weather data

From the file: QUALITY CONTROLLED LOCAL CLIMATOLOGICAL DATA

1. The dry bulb, dew point and wet bulb temperatures were originally reported to the nearest tenth of a degree Fahrenheit. The **Automated Surface Observing System (ASOS)** records temperatures and dew points in whole degrees Fahrenheit and converts these values to the nearest tenth of a degree Celsius for observation transmission. Until this date, these values online have incorrectly been converted back to the nearest tenth of a degree Fahrenheit, implying a level of precision that is not present at the instrument level.
2. Two stations.

- Whole degree Celsius temperature values for **AWOS** stations;
- Tenths degrees Celsius temperature values for **ASOS** stations.

Their location:

- Station 1: CHICAGO O'HARE INTERNATIONAL AIRPORT Lat: 41.995 Lon: -87.933 Elev: 662 ft. above sea level
- Station 2: CHICAGO MIDWAY INTL ARPT Lat: 41.786 Lon: -87.752 Elev: 612 ft. above sea level

From https://www.kaggle.com/c/predict-west-nile-virus/data

Note of some features

- **WetBulb**: Wet-bulb temperature is largely determined by both actual air temperature (dry-bulb temperature) and the amount of moisture in the air (humidity)

Data engineering

Separate the data set by station

```
weather.data.split <- split(weather.data, weather.data$Station)
weather.stn1 <- weather.data.split[[1]]
weather.stn2 <- weather.data.split[[2]]
dim(weather.stn1)
```

```
## [1] 1472   22
```

```
dim(weather.stn2)
```

```
## [1] 1472   22
```

Sunrise and Sunset

Only station 1 has such data, but they should be the same for the two stations

```
weather.stn2$Sunrise <- weather.stn1$Sunrise
weather.stn2$Sunset <- weather.stn1$Sunset
```

Depart

Depart means "DEPARTURE FROM NORMAL".

```
summary(weather.stn1$Depart)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -17.000  -3.000   2.000   1.954   7.000  23.000
```

```
summary(weather.stn2$Depart)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##      NA      NA      NA     NaN      NA      NA    1472
```

```
normal.tmp <- weather.stn1$Tavg - weather.stn1$Depart
weather.stn2$Depart <- weather.stn2$Tavg - normal.tmp
summary(weather.stn1$Depart)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -17.000  -3.000   2.000   1.954   7.000  23.000
```

```
summary(weather.stn2$Depart)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
## -17.000  -2.000   3.000   3.207   8.000  25.000      11
```

CodeSum

First find out all the unique code for different weather.

```
code.A <- unique(unlist(strsplit(unique(as.character(weather.stn1$CodeSum)), " ")))
code.B <- unique(unlist(strsplit(unique(as.character(weather.stn2$CodeSum)), " ")))
code <- union(code.A, code.B)[-1]
code
```

```
##  [1] "BR"   "RA"   "HZ"   "VCTS" "TSRA" "FU"   "DZ"   "TS"   "FG+"  "BCFG"
## [11] "MIFG" "FG"   "SQ"   "SN"   "VCFG" "GR"
```

```
code.name <- paste(rep("Code.", length(code)), code, sep="")
code.name
```

```
##  [1] "Code.BR"   "Code.RA"   "Code.HZ"   "Code.VCTS" "Code.TSRA"
##  [6] "Code.FU"   "Code.DZ"   "Code.TS"   "Code.FG+"  "Code.BCFG"
## [11] "Code.MIFG" "Code.FG"   "Code.SQ"   "Code.SN"   "Code.VCFG"
## [16] "Code.GR"
```

Then add new columns indicating distinct weather code in the data frames. Observe all levels in CodeSum, "FG", "TS" and "RA" are the tricky ones, since these letter pairs appear in more than one code.

```
code
```

```
##  [1] "BR"   "RA"   "HZ"   "VCTS" "TSRA" "FU"   "DZ"   "TS"   "FG+"  "BCFG"
## [11] "MIFG" "FG"   "SQ"   "SN"   "VCFG" "GR"
```

```
## rewrite the reguar expression of "FG", "TS" and "RA" to avoid wrong matching
code[2] <- "^RA | RA$| RA |^RA$"
code[8] <- "^TS | TS$| TS |^TS$"
code[12] <- "^FG | FG$| FG |^FG$"
```

```
for(i in 1:length(code)) {
     new.code <- code[i]
    new.code.name <- code.name[i]
    weather.stn1[, new.code.name] <- grepl(new.code, weather.stn1$CodeSum)
    weather.stn2[, new.code.name] <- grepl(new.code, weather.stn2$CodeSum)
}
## show the resultant features
head(weather.stn1[,c(13, 23:38)])
```

```
##    CodeSum Code.BR Code.RA Code.HZ Code.VCTS Code.TSRA Code.FU Code.DZ
## 1     <NA>   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 3       BR    TRUE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 5     <NA>   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 7       RA   FALSE    TRUE   FALSE     FALSE     FALSE   FALSE   FALSE
## 9     <NA>   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 11    <NA>   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
##    Code.TS Code.FG+ Code.BCFG Code.MIFG Code.FG Code.SQ Code.SN Code.VCFG
## 1    FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 3    FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 5    FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 7    FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 9    FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 11   FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
##    Code.GR
## 1    FALSE
## 3    FALSE
## 5    FALSE
## 7    FALSE
## 9    FALSE
## 11   FALSE
```

```
head(weather.stn2[,c(13, 23:38)])
```

```
##    CodeSum Code.BR Code.RA Code.HZ Code.VCTS Code.TSRA Code.FU Code.DZ
## 2     <NA>   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 4    BR HZ    TRUE   FALSE    TRUE     FALSE     FALSE   FALSE   FALSE
## 6       HZ   FALSE   FALSE    TRUE     FALSE     FALSE   FALSE   FALSE
## 8     <NA>   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 10    <NA>   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 12    <NA>   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
##    Code.TS Code.FG+ Code.BCFG Code.MIFG Code.FG Code.SQ Code.SN Code.VCFG
## 2    FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 4    FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 6    FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 8    FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 10   FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 12   FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
##    Code.GR
## 2    FALSE
## 4    FALSE
## 6    FALSE
## 8    FALSE
## 10   FALSE
## 12   FALSE
```

Water1

```
summary(weather.stn1$Water1)
```

```
##    Mode    NA's
## logical    1472
```

```
summary(weather.stn2$Water1)
```

```
##    Mode    NA's
## logical    1472
```

This Water1 is useless, remove it from data frames.

```
weather.stn1 <- weather.stn1[,-15]
weather.stn2 <- weather.stn2[,-15]
```

SnowFall

SnowFall is not a good predictor, since the time is from May to October.

```
summary(weather.stn1["SnowFall"])
```

```
##  SnowFall
##  0.0:1459
##  0.1:   1
##    T:  12
```

```
summary(weather.stn2["SnowFall"])
```

```
##  SnowFall
##  0.0 :   0
##  0.1 :   0
##    T :   0
##  NA's:1472
```

```
weather.stn1 <- weather.stn1[,-15]
weather.stn2 <- weather.stn2[,-15]
```

Date

```
weather.stn1$Date <- as.Date(weather.stn1$Date)
weather.stn2$Date <- as.Date(weather.stn2$Date)
```

PrecipTotal

```
weather.stn1$PrecipTotal <- as.numeric(weather.stn1$PrecipTotal)
weather.stn2$PrecipTotal <- as.numeric(weather.stn2$PrecipTotal)
```

Combined Main and Weather Data Set

Merge train/test and weather.stn1/weather.stn2

Each row in main data set is merged to the weather record by the closer station.

```
## combine the weather.stn1 and weather.stn2 to one data frame first
weather.stn <- rbind(weather.stn1, weather.stn2)
## merge
train <- merge(train, weather.stn, by.x=c("Date", "ClosestStn"), by.y=c("Date", "Station"))
test <- merge(test, weather.stn, by.x=c("Date", "ClosestStn"), by.y=c("Date", "Station"))
## show the new data sets
str(train, strict.width="cut")
```

```
## 'data.frame':    8475 obs. of  55 variables:
##  $ Date                   : Date, format: "2007-05-29" "2007-05-29" ...
##  $ ClosestStn             : num  1 1 1 1 1 1 1 2 2 2 ...
##  $ Address                : Factor w/ 138 levels "1000 East 67th Street, "..
##  $ Species                : Factor w/ 7 levels "CULEX ERRATICUS",..: 3 4 4..
##  $ Block                  : int  41 41 62 79 79 75 65 25 11 15 ...
##  $ Street                 : Factor w/ 128 levels " E 105TH ST",..: 33 33 2..
##  $ Trap                   : Factor w/ 136 levels "T001","T002",..: 2 2 7 1..
##  $ AddressNumberAndStreet: Factor w/ 138 levels "1000  E 67TH ST, Chicag"..
##  $ Latitude               : num  42 42 42 42 42 ...
##  $ Longitude              : num  -87.8 -87.8 -87.8 -87.8 -87.8 ...
##  $ AddressAccuracy        : int  9 9 9 8 8 8 8 8 8 8 ...
```

```
##  $ Year                 : int  2007 2007 2007 2007 2007 2007 2007 2007 2..
##  $ Month                : Factor w/ 6 levels "May","June","July",..: 1 1..
##  $ Week                 : num  22 22 22 22 22 22 22 22 22 22 ...
##  $ Weekday              : Factor w/ 5 levels "Monday","Tuesday",..: 2 2 ..
##  $ NumMosquitos         : int  1 1 1 1 4 1 1 1 1 2 ...
##  $ WnvPresent           : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ TrapNumber           : int  2 2 7 15 15 148 143 46 48 45 ...
##  $ TrapMS               : Factor w/ 3 levels "M","B","C": 1 1 1 1 1 1 1 ..
##  $ DisStn1              : num  11.8 11.8 13.53 9.24 9.24 ...
##  $ DisStn2              : num  19.2 19.2 23.3 21.8 21.8 ...
##  $ Tmax                 : int  88 88 88 88 88 88 88 88 88 88 ...
##  $ Tmin                 : int  60 60 60 60 60 60 60 65 65 65 ...
##  $ Tavg                 : int  74 74 74 74 74 74 74 77 77 77 ...
##  $ Depart               : int  10 10 10 10 10 10 10 13 13 13 ...
##  $ DewPoint             : int  58 58 58 58 58 58 58 59 59 59 ...
##  $ WetBulb              : int  65 65 65 65 65 65 65 66 66 66 ...
##  $ Heat                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Cool                 : int  9 9 9 9 9 9 9 12 12 12 ...
##  $ Sunrise              : int  421 421 421 421 421 421 421 421 421 421 ...
##  $ Sunset               : int  1917 1917 1917 1917 1917 1917 1917 1917 1..
##  $ CodeSum              : Factor w/ 97 levels "BCFG BR","BR",..: 3 3 3 3..
##  $ Depth                : int  0 0 0 0 0 0 0 NA NA NA ...
##  $ PrecipTotal          : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ StnPressure          : num  29.4 29.4 29.4 29.4 29.4 ...
##  $ SeaLevel             : num  30.1 30.1 30.1 30.1 30.1 ...
##  $ ResultSpeed          : num  5.8 5.8 5.8 5.8 5.8 5.8 5.8 5.8 5.8 5.8 ...
##  $ ResultDir            : int  18 18 18 18 18 18 18 16 16 16 ...
##  $ AvgSpeed             : num  6.5 6.5 6.5 6.5 6.5 6.5 6.5 7.4 7.4 7.4 ...
##  $ Code.BR              : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ Code.RA              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.HZ              : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ Code.VCTS            : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.TSRA            : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.FU              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.DZ              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.TS              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.FG+             : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.BCFG            : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.MIFG            : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.FG              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.SQ              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.SN              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.VCFG            : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ Code.GR              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
```

```
head(test)
```

```
##         Date ClosestStn Id
## 1 2008-06-11          1  1
## 2 2008-06-11          1  2
## 3 2008-06-11          1  3
## 4 2008-06-11          1  4
## 5 2008-06-11          1  5
## 6 2008-06-11          1  6
```

```
##                                           Address
## 1 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 2 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 3 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 4 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 5 4100 North Oak Park Avenue, Chicago, IL 60634, USA
## 6 4100 North Oak Park Avenue, Chicago, IL 60634, USA
##                  Species Block          Street Trap
## 1 CULEX PIPIENS/RESTUANS    41  N OAK PARK AVE T002
## 2         CULEX RESTUANS    41  N OAK PARK AVE T002
## 3          CULEX PIPIENS    41  N OAK PARK AVE T002
## 4       CULEX SALINARIUS    41  N OAK PARK AVE T002
## 5        CULEX TERRITANS    41  N OAK PARK AVE T002
## 6         CULEX TARSALIS    41  N OAK PARK AVE T002
##              AddressNumberAndStreet Latitude Longitude AddressAccuracy
## 1 4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 2 4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 3 4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 4 4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 5 4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
## 6 4100  N OAK PARK AVE, Chicago, IL 41.95469 -87.80099               9
##   Year Month Week   Weekday NumMosquitos TrapNumber TrapMS  DisStn1
## 1 2008  June   24 Wednesday            3          2      M 11.79739
## 2 2008  June   24 Wednesday            3          2      M 11.79739
## 3 2008  June   24 Wednesday            3          2      M 11.79739
## 4 2008  June   24 Wednesday            3          2      M 11.79739
## 5 2008  June   24 Wednesday            3          2      M 11.79739
## 6 2008  June   24 Wednesday            3          2      M 11.79739
##   DisStn2 Tmax Tmin Tavg Depart DewPoint WetBulb Heat Cool Sunrise Sunset
## 1 19.1911   86   61   74      7       56      64    0    9     416   1926
## 2 19.1911   86   61   74      7       56      64    0    9     416   1926
## 3 19.1911   86   61   74      7       56      64    0    9     416   1926
## 4 19.1911   86   61   74      7       56      64    0    9     416   1926
## 5 19.1911   86   61   74      7       56      64    0    9     416   1926
## 6 19.1911   86   61   74      7       56      64    0    9     416   1926
##   CodeSum Depth PrecipTotal StnPressure SeaLevel ResultSpeed ResultDir
## 1    <NA>     0           1       29.28    29.99         8.9        18
## 2    <NA>     0           1       29.28    29.99         8.9        18
## 3    <NA>     0           1       29.28    29.99         8.9        18
## 4    <NA>     0           1       29.28    29.99         8.9        18
## 5    <NA>     0           1       29.28    29.99         8.9        18
## 6    <NA>     0           1       29.28    29.99         8.9        18
##   AvgSpeed Code.BR Code.RA Code.HZ Code.VCTS Code.TSRA Code.FU Code.DZ
## 1       10   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 2       10   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 3       10   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 4       10   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 5       10   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
## 6       10   FALSE   FALSE   FALSE     FALSE     FALSE   FALSE   FALSE
##   Code.TS Code.FG+ Code.BCFG Code.MIFG Code.FG Code.SQ Code.SN Code.VCFG
## 1   FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 2   FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 3   FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 4   FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
```

```
## 5   FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
## 6   FALSE    FALSE     FALSE     FALSE   FALSE   FALSE   FALSE     FALSE
##   Code.GR
## 1   FALSE
## 2   FALSE
## 3   FALSE
## 4   FALSE
## 5   FALSE
## 6   FALSE
```

Save this data set

```
write.csv(train, "../data/train3.csv", row.names=F)
write.csv(test, "../data/test3.csv", row.names=F)
```