Zipline data challenge

JIASHU MIAO

5/1/2019

Introduction

- This is the R markdown file that contains most of my code doing this project.
- First I look at the summary data and pick several individual flight data to have a glimpse of the datasets.
- Then start my question which could be categorized as:
- Type of variables and summary, dimensions, etc.
- Any special values in the dataset (NA, Inf, NaN, Null, Unexplained value)
- Any potential outlier launches, outlier components, or mal-functional parts, or fault data entries that do not make sense or different from the majority.
- Any patterns/models, interpretations and conclusions from study the data with manipulation, modeling and data mining.
- Any further study by creating new variables or relate with other sources (eg. Physics, Aviation Journals, Articles, etc.)
- Any insights/suggestions that are actionable and practicable for Zipline to improve the business and overall drone quality.

Environmental Setup and Data Loading

```
# set up environment for online plot you may check and play with
Sys.setenv("plotly_username"="michaelmiaomiao")
Sys.setenv("plotly_api_key"="BNIZiiSEJ4LqoRuJbZ3a")

# load the package possibly needed for the data analysis and processing.
pkg <- c("readr","readxl","dplyr","stringr","ggplot2","tidyr","car","lubridate","caret","randomForest
")
pkgload <- lapply(pkg, require, character.only = TRUE)
```

```
## Loading required package: readr
```

```
## Loading required package: readxl
```

```
## Loading required package: dplyr
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
## Loading required package: stringr
```

```
## Loading required package: ggplot2
```

```
## Loading required package: tidyr
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## Loading required package: lubridate
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
## Loading required package: caret
```

```
## Loading required package: lattice
```

```
## Loading required package: randomForest
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
library(lubridate) #time
library(plotly) #graph
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```r
# write the function for ploting graphs effectively used later.
ggplotRegression <- function (fit) {
   require(ggplot2)

   ggplot(fit$model, aes_string(x = names(fit$model)[2], y = names(fit$model)[1])) +
   geom_point() +
   stat_smooth(method = "lm", col = "red") +
   labs(title = paste(
   "Adj R2 = ",
   signif(summary(fit)$adj.r.squared, 5),
   "Intercept =",
   signif(fit$coef[[1]], 5),
   " Slope =",
   signif(fit$coef[[2]], 5),
   " P =",
   signif(summary(fit)$coef[2, 4], 5)
   ))
}
# Load the data (incase the computer crashes, I looked up the data manually before load into R)
summary_data <- read.csv("summary_data.csv") %>% glimpse()
```

```
## Observations: 447
## Variables: 14
## $ flight_id             <int> 16951, 16952, 16954, 16955, 16957, 16959, …
## $ air_temperature       <dbl> 20.55000, 20.50000, 24.47502, 27.30000, 26…
## $ battery_serial_number <fct> 15SPJJJ09036021, 15SPJJJ10029029, 15SPJJJ1…
## $ body_serial_number    <dbl> 5.773501e+17, 5.772096e+17, 5.772096e+17, …
## $ commit                <fct> 5c504d9a16, 5c504d9a16, 5c504d9a16, 5c504d…
## $ launch_airspeed       <dbl> 32.45345, 32.14121, 34.70188, 34.36900, 32…
## $ launch_groundspeed    <dbl> 30.16466, 30.53525, 29.87261, 29.87762, 30…
## $ launch_timestamp      <fct> 2018-09-06 07:43:59 CAT, 2018-09-06 07:51:…
## $ preflight_voltage     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA…
## $ rel_humidity          <dbl> 74.15000, 71.17504, 66.37498, 59.00000, 63…
## $ static_pressure       <dbl> 80662.08, 80708.07, 80774.27, 80805.14, 80…
## $ wind_direction        <dbl> -49.434555, -4.408768, -23.458781, -46.747…
## $ wind_magnitude        <dbl> 1.9493382, 0.9173566, 3.7883831, 3.9216052…
## $ wing_serial_number    <fct> 15SPJJJ11024054, 15SPJJJ09011032, 15SPJJJ0…
```

```
dim(summary_data)
```

```
## [1] 447  14
```

```
summary(summary_data)
```

```
##    flight_id     air_temperature     battery_serial_number
##  Min.   :16951   Min.   :16.50   15SPJJJ10012034: 31
##  1st Qu.:17170   1st Qu.:22.04   15SPJJJ10029029: 27
##  Median :17359   Median :24.95   15SPJJJ09036021: 26
##  Mean   :17373   Mean   :25.23   15SPJJJ10050016: 26
##  3rd Qu.:17590   3rd Qu.:28.32   15SPJJJ09018015: 24
##  Max.   :17745   Max.   :34.60   15SPJJJ11059037: 23
##                                  (Other)        :290
##  body_serial_number         commit    launch_airspeed launch_groundspeed
##  Min.   :5.772e+17   1ecbc27833: 65   Min.   :28.03   Min.   :27.55
##  1st Qu.:5.773e+17   38bf99b15a: 60   1st Qu.:30.76   1st Qu.:29.93
##  Median :5.774e+17   4d9468bd3c: 12   Median :31.89   Median :30.10
##  Mean   :5.773e+17   5c504d9a16:310   Mean   :31.98   Mean   :30.11
##  3rd Qu.:5.774e+17                    3rd Qu.:33.20   3rd Qu.:30.28
##  Max.   :5.774e+17                    Max.   :36.93   Max.   :31.21
##
##                  launch_timestamp preflight_voltage  rel_humidity
##  2018-09-06 07:43:59 CAT:  1     Min.   :31.54     Min.   :35.50
##  2018-09-06 07:51:49 CAT:  1     1st Qu.:32.06     1st Qu.:51.20
##  2018-09-06 09:56:37 CAT:  1     Median :32.19     Median :56.20
##  2018-09-06 10:27:04 CAT:  1     Mean   :32.15     Mean   :56.29
##  2018-09-06 11:09:39 CAT:  1     3rd Qu.:32.27     3rd Qu.:61.35
##  2018-09-06 11:31:07 CAT:  1     Max.   :32.52     Max.   :74.15
##  (Other)                :441     NA's   :16
##  static_pressure wind_direction    wind_magnitude
##  Min.   :80010   Min.   :-176.13   Min.   :0.1888
##  1st Qu.:80324   1st Qu.: -78.53   1st Qu.:1.7033
##  Median :80445   Median : -51.63   Median :2.3077
##  Mean   :80456   Mean   : -45.29   Mean   :2.3595
##  3rd Qu.:80590   3rd Qu.: -25.95   3rd Qu.:3.0070
##  Max.   :80844   Max.   : 179.70   Max.   :7.4662
##
##       wing_serial_number
##  15SPJJJ09008034: 65
##  15SPJJJ09025064: 58
##  15SPJJJ09052035: 51
##  15SPJJJ09024061: 45
##  15SPJJJ09040032: 44
##  15SPJJJ09031032: 27
##  (Other)        :157
```

```
# Randomly Load two individual flight data to check and explore
f16951 <- read.csv("flight_16951.csv") %>%  glimpse()
```

```
## Observations: 1,001
## Variables: 19
## $ seconds_since_launch       <dbl> -4.99846, -4.97846, -4.95833, -4.9384…
## $ position_ned_m.0.          <dbl> 5.143372, 5.143372, 5.143372, 5.14354…
## $ position_ned_m.1.          <dbl> 8.170100, 8.170100, 8.170100, 8.16881…
## $ position_ned_m.2.          <dbl> -4.561916, -4.561916, -4.561916, -4.5…
## $ velocity_ned_mps.0.        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
## $ velocity_ned_mps.1.        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
## $ velocity_ned_mps.2.        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
## $ accel_body_mps2.0.         <dbl> 2.200169, 2.027999, 2.051773, 2.17316…
## $ accel_body_mps2.1.         <dbl> -0.059349830, -0.053380057, -0.264372…
## $ accel_body_mps2.2.         <dbl> -9.497843, -9.611131, -9.632382, -9.6…
## $ orientation_rad.0.         <dbl> 0.007623033, 0.007616369, 0.007601701…
## $ orientation_rad.1.         <dbl> 0.2143962, 0.2144144, 0.2144239, 0.21…
## $ orientation_rad.2.         <dbl> 2.741056, 2.741054, 2.741051, 2.74105…
## $ angular_rate_body_radps.0. <dbl> 0.0020107578, 0.0049842047, 0.0008864…
## $ angular_rate_body_radps.1. <dbl> -6.391027e-04, -7.657856e-04, 2.61283…
## $ angular_rate_body_radps.2. <dbl> 4.947464e-04, 1.945693e-03, -1.268762…
## $ position_sigma_ned_m.0.    <dbl> 0.1865353, 0.1865353, 0.1865353, 0.18…
## $ position_sigma_ned_m.1.    <dbl> 0.3408245, 0.3408245, 0.3408245, 0.34…
## $ position_sigma_ned_m.2.    <dbl> 0.4286826, 0.4286826, 0.4286826, 0.42…
```

```
dim(f16951)
```

```
## [1] 1001   19
```

```
#summary(f16951)

f17326 <- read.csv("flight_17326.csv")
head(f17326,3)
```

```
##   seconds_since_launch position_ned_m.0. position_ned_m.1.
## 1             -4.99752          4.837937          4.269466
## 2             -4.97847          4.837937          4.269466
## 3             -4.95848          4.836390          4.270962
##   position_ned_m.2. velocity_ned_mps.0. velocity_ned_mps.1.
## 1         -4.093463                   0                   0
## 2         -4.093463                   0                   0
## 3         -4.091085                   0                   0
##   velocity_ned_mps.2. accel_body_mps2.0. accel_body_mps2.1.
## 1                   0           2.186458        -0.08728751
## 2                   0           1.983590        -0.21528484
## 3                   0           2.148795        -0.04404519
##   accel_body_mps2.2. orientation_rad.0. orientation_rad.1.
## 1          -9.567879        0.007625219          0.2184191
## 2          -9.503726        0.007612888          0.2184156
## 3          -9.568281        0.007613972          0.2183988
##   orientation_rad.2. angular_rate_body_radps.0. angular_rate_body_radps.1.
## 1           2.741071               1.940164e-05               0.0007292685
## 2           2.741068               2.804355e-03               0.0002513833
## 3           2.741068              -3.879838e-03              -0.0013232076
##   angular_rate_body_radps.2. position_sigma_ned_m.0.
## 1                0.001375442               0.4753959
## 2               -0.001871372               0.4753959
## 3               -0.001022377               0.4754192
##   position_sigma_ned_m.1. position_sigma_ned_m.2.
## 1               0.7821813                1.021483
## 2               0.7821813                1.021483
## 3               0.7821956                1.021601
```

```
dim(f17326)
```

```
## [1] 1001   19
```

```
#summary(f17326)

#After check the individual data, I load all 447 flight at one time :
temp = list.files(pattern="*.csv")
myfiles = lapply(temp, read.csv)
# myfiles[[448]] # is the summary data and I drop it here.
myfiles <- myfiles[-448]
    myfiles %>% length()
```

```
## [1] 447
```

Data Cleaning and Manipulation

```
# Now I would like to check the special values in the dataset (NA, Inf, NaN, Null, Unexplained value)
# Create functions that more powerfully go through all datasets at one time



# summary_data %>% glimpse() %>% dim()
attach(summary_data,warn.conflicts = F)

# before start, I would like to conduct check for NA, NULL, Inf, NaN.

#NA
(colSums(is.na(summary_data)) !=0)
```

```
##             flight_id       air_temperature battery_serial_number
##                 FALSE                 FALSE                 FALSE
##    body_serial_number                commit       launch_airspeed
##                 FALSE                 FALSE                 FALSE
##    launch_groundspeed      launch_timestamp     preflight_voltage
##                 FALSE                 FALSE                  TRUE
##          rel_humidity       static_pressure        wind_direction
##                 FALSE                 FALSE                 FALSE
##        wind_magnitude    wing_serial_number
##                 FALSE                 FALSE
```

```
sum(is.na(preflight_voltage)) # we found 16 minssing values in preflight_voltage variable.
```

```
## [1] 16
```

```
tr_NA <- 0
for (i in 1:14) {
   tr_NA= sum(is.na(summary_data[,i]))
   if (!tr_NA==0)
      cat(tr_NA,"Missing values (NULL)for",names(summary_data[i]),"\n")
}
```

```
## 16 Missing values (NULL)for preflight_voltage
```

```
#NULL (empty, NULL)

tr_NULL <- 0
for (i in 1:14) {
   tr_NULL= sum(is.null(summary_data[,i]))
   if (!tr_NULL==0)
      cat(tr_NULL,"Null values (NULL)for",names(summary_data[i]),"\n")
}

#NaN (not a number)

tr_NaN <- 0
for (i in 1:14) {
   tr_NaN= sum(is.nan(summary_data[,i]))
   if (!tr_NaN==0)
      cat(tr_NaN,"not a number (NaN) for",names(summary_data[i]),"\n")
}

# Inf (infinite)

tr_inf <- 0
for ( i in 1:14 ) {
   tr_inf = sum(is.infinite(summary_data[,i]))
   if (!tr_inf==0)
   cat(tr_inf,"infinite values (INF) for",names(summary_data[i]),"\n")
}

# which is the one missing volatage:
which(is.na(summary_data$preflight_voltage))
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 16 18
```

```
summary_data[which(is.na(summary_data$preflight_voltage)), ]
```

```
##    flight_id air_temperature battery_serial_number body_serial_number
## 1      16951        20.55000       15SPJJJ09036021       5.773501e+17
## 2      16952        20.50000       15SPJJJ10029029       5.772096e+17
## 3      16954        24.47502       15SPJJJ10012034       5.772096e+17
## 4      16955        27.30000       15SPJJJ10054027       5.772096e+17
## 5      16957        26.95000       15SPJJJ10050049       5.773488e+17
## 6      16959        28.57495       15SPJJJ09018015       5.773501e+17
## 7      16960        27.55000       15SPJJJ09017016       5.772096e+17
## 8      16961        28.25000       15SPJJJ10023027       5.773501e+17
## 9      16962        28.60000       15SPJJJ10052026       5.773501e+17
## 10     16965        32.25000       15SPJJJ10029029       5.772096e+17
## 11     16967        32.40000       15SPJJJ09036021       5.773501e+17
## 12     16980        18.20000       15SPJJJ10052026       5.773488e+17
## 13     16983        18.40000       15SPJJJ10050049       5.773501e+17
## 14     16984        18.20000       15SPJJJ09013015       5.773501e+17
## 16     16986        18.30000       15SPJJJ09036021       5.773488e+17
## 18     16988        18.37652       15SPJJJ10008029       5.772096e+17
##        commit launch_airspeed launch_groundspeed        launch_timestamp
## 1  5c504d9a16        32.45345           30.16466 2018-09-06 07:43:59 CAT
## 2  5c504d9a16        32.14121           30.53525 2018-09-06 07:51:49 CAT
## 3  5c504d9a16        34.70188           29.87261 2018-09-06 09:56:37 CAT
## 4  5c504d9a16        34.36900           29.87762 2018-09-06 10:27:04 CAT
```

```
## 5  5c504d9a16        32.89898           30.02718 2018-09-06 11:09:39 CAT
## 6  5c504d9a16        33.25801           30.17881 2018-09-06 11:31:07 CAT
## 7  5c504d9a16        33.93734           30.06319 2018-09-06 12:55:23 CAT
## 8  5c504d9a16        33.59898           29.96951 2018-09-06 13:09:51 CAT
## 9  5c504d9a16        31.63985           30.26374 2018-09-06 13:43:05 CAT
## 10 5c504d9a16        32.74496           30.35478 2018-09-06 14:56:25 CAT
## 11 5c504d9a16        33.74804           30.14972 2018-09-06 15:02:27 CAT
## 12 5c504d9a16        28.26758           31.02285 2018-09-06 17:46:38 CAT
## 13 5c504d9a16        30.38840           31.12986 2018-09-06 18:04:04 CAT
## 14 5c504d9a16        28.82763           30.50900 2018-09-06 17:56:06 CAT
## 16 5c504d9a16        30.60393           30.11974 2018-09-06 18:25:40 CAT
## 18 5c504d9a16        28.43547           30.47430 2018-09-06 18:59:13 CAT
##    preflight_voltage rel_humidity static_pressure wind_direction
## 1                 NA     74.15000        80662.08     -49.434555
## 2                 NA     71.17504        80708.07      -4.408768
## 3                 NA     66.37498        80774.27     -23.458781
## 4                 NA     59.00000        80805.14     -46.747881
## 5                 NA     63.90000        80768.97     -29.293360
## 6                 NA     65.07495        80621.20     -68.360838
## 7                 NA     61.25000        80599.90     -27.822443
## 8                 NA     53.50000        80552.49       7.094333
## 9                 NA     60.37498        80445.02     -46.053006
## 10                NA     49.60000        80379.65     -17.594640
## 11                NA     57.62499        80382.99      -6.229944
## 12                NA     67.80000        80473.49     173.524053
## 13                NA     65.90000        80371.51     177.288807
## 14                NA     65.75000        80554.22     157.407334
## 16                NA     69.47499        80468.89     -38.575222
## 18                NA     64.87652        80579.96     163.843576
##    wind_magnitude wing_serial_number
## 1       1.9493382    15SPJJJ11024054
## 2       0.9173566    15SPJJJ09011032
## 3       3.7883831    15SPJJJ09011032
## 4       3.9216052    15SPJJJ11049056
## 5       2.9758809    15SPJJJ09031032
## 6       2.7503460    15SPJJJ11024054
## 7       1.5563404    15SPJJJ09031032
## 8       2.3786070    15SPJJJ11049056
## 9       1.1619245    15SPJJJ09011032
## 10      2.7420269    15SPJJJ11049056
## 11      2.6763300    15SPJJJ09031032
## 12      2.3755740    15SPJJJ11024054
## 13      1.7803189    15SPJJJ09025064
## 14      1.9940298    15SPJJJ11049056
## 16      0.3140002    15SPJJJ11024054
## 18      2.3874066    15SPJJJ11049056
```

```
# They are flight: 16951 16952 16954 16955 16957 16959 16960 16961 16962 16965 16967 16980 16983 1698
4 16986 16988
# We find missing values in preflight_voltage colum with 16 NA's.




# I also check the 447 myfiles data sets for each individual and it's cleaned now. (Did't put that co
de here similarly algorithm as above)
na_check <- NULL
for (i in 1:447) {
   na_check[i] <- sum(as.vector(colSums(is.na(myfiles[[i]]))))
}
   any(!na_check==0)
```

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

```
   # No missing values in each flight data

detach(summary_data)
```

`Anwser` - I find missing values in the pre_voltage column and it has 16 NA's. - They are flight: **16951 16952 16954 16955 16957 16959 16960 16961 16962 16965 16967 16980 16983 16984 16986 16988.** The first 18 flights expect for No. 15 and 17. 0

Data Manipulation

```
### Select the individual data from each flight at time = 0 (the moment launching) and combine with s
ummary data.

filter_fun2 <- function(x){




   myfiles[[x]][which(myfiles[[x]][1]==0),]
   myfiles[[x]][which(myfiles[[x]][1]==0),]=as.vector(myfiles[[x]][which(myfiles[[x]][1]==0),])
  return(myfiles[[x]][which(myfiles[[x]][1]==0),])
}

fil <- list(NULL)
for (i in 1:447){

   fil[[i]] <- filter_fun2(i)
}
new_0 <- bind_rows(fil)
sumcom <- cbind(summary_data,new_0) %>% glimpse()
```

```
## Observations: 447
## Variables: 33
## $ flight_id                  <int> 16951, 16952, 16954, 16955, 16957, 16…
## $ air_temperature            <dbl> 20.55000, 20.50000, 24.47502, 27.3000…
## $ battery_serial_number      <fct> 15SPJJJ09036021, 15SPJJJ10029029, 15S…
## $ body_serial_number         <dbl> 5.773501e+17, 5.772096e+17, 5.772096e…
## $ commit                     <fct> 5c504d9a16, 5c504d9a16, 5c504d9a16, 5…
## $ launch_airspeed            <dbl> 32.45345, 32.14121, 34.70188, 34.3690…
## $ launch_groundspeed         <dbl> 30.16466, 30.53525, 29.87261, 29.8776…
## $ launch_timestamp           <fct> 2018-09-06 07:43:59 CAT, 2018-09-06 0…
## $ preflight_voltage          <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, N…
## $ rel_humidity               <dbl> 74.15000, 71.17504, 66.37498, 59.0000…
## $ static_pressure            <dbl> 80662.08, 80708.07, 80774.27, 80805.1…
## $ wind_direction             <dbl> -49.434555, -4.408768, -23.458781, -4…
## $ wind_magnitude             <dbl> 1.9493382, 0.9173566, 3.7883831, 3.92…
## $ wing_serial_number         <fct> 15SPJJJ11024054, 15SPJJJ09011032, 15S…
## $ seconds_since_launch       <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
## $ position_ned_m.0.          <dbl> -6.076665, -5.291423, -5.921773, -5.0…
## $ position_ned_m.1.          <dbl> 13.07089, 12.70803, 12.29018, 10.7320…
## $ position_ned_m.2.          <dbl> -7.098896, -7.163928, -6.849090, -5.9…
## $ velocity_ned_mps.0.        <dbl> -27.52789, -28.02120, -27.42528, -27.…
## $ velocity_ned_mps.1.        <dbl> 11.91206, 11.58430, 11.50415, 11.2015…
## $ velocity_ned_mps.2.        <dbl> -5.593383, -5.581750, -5.598970, -5.9…
## $ accel_body_mps2.0.         <dbl> -0.9511417, -1.9985299, -1.0178263, 1…
## $ accel_body_mps2.1.         <dbl> 0.3976415, 1.1861557, 0.7624364, 4.03…
## $ accel_body_mps2.2.         <dbl> -6.674903, -5.974786, -4.019612, -11.…
## $ orientation_rad.0.         <dbl> 0.012297876, 0.035127785, 0.017262662…
## $ orientation_rad.1.         <dbl> 0.1602600, 0.1692261, 0.1617596, 0.19…
## $ orientation_rad.2.         <dbl> 2.752593, 2.754875, 2.747974, 2.74753…
## $ angular_rate_body_radps.0. <dbl> -0.01260387, 0.27340308, 0.28942248, …
## $ angular_rate_body_radps.1. <dbl> -0.2363039, -0.3474914, -0.2299277, -…
## $ angular_rate_body_radps.2. <dbl> 0.04230096, 0.12799662, 0.04695929, 0…
## $ position_sigma_ned_m.0.    <dbl> 0.26747534, 0.31797993, 0.36143770, 0…
## $ position_sigma_ned_m.1.    <dbl> 0.4943953, 0.5187752, 0.6756144, 0.44…
## $ position_sigma_ned_m.2.    <dbl> 0.6233013, 0.7073184, 0.8630315, 0.36…
```

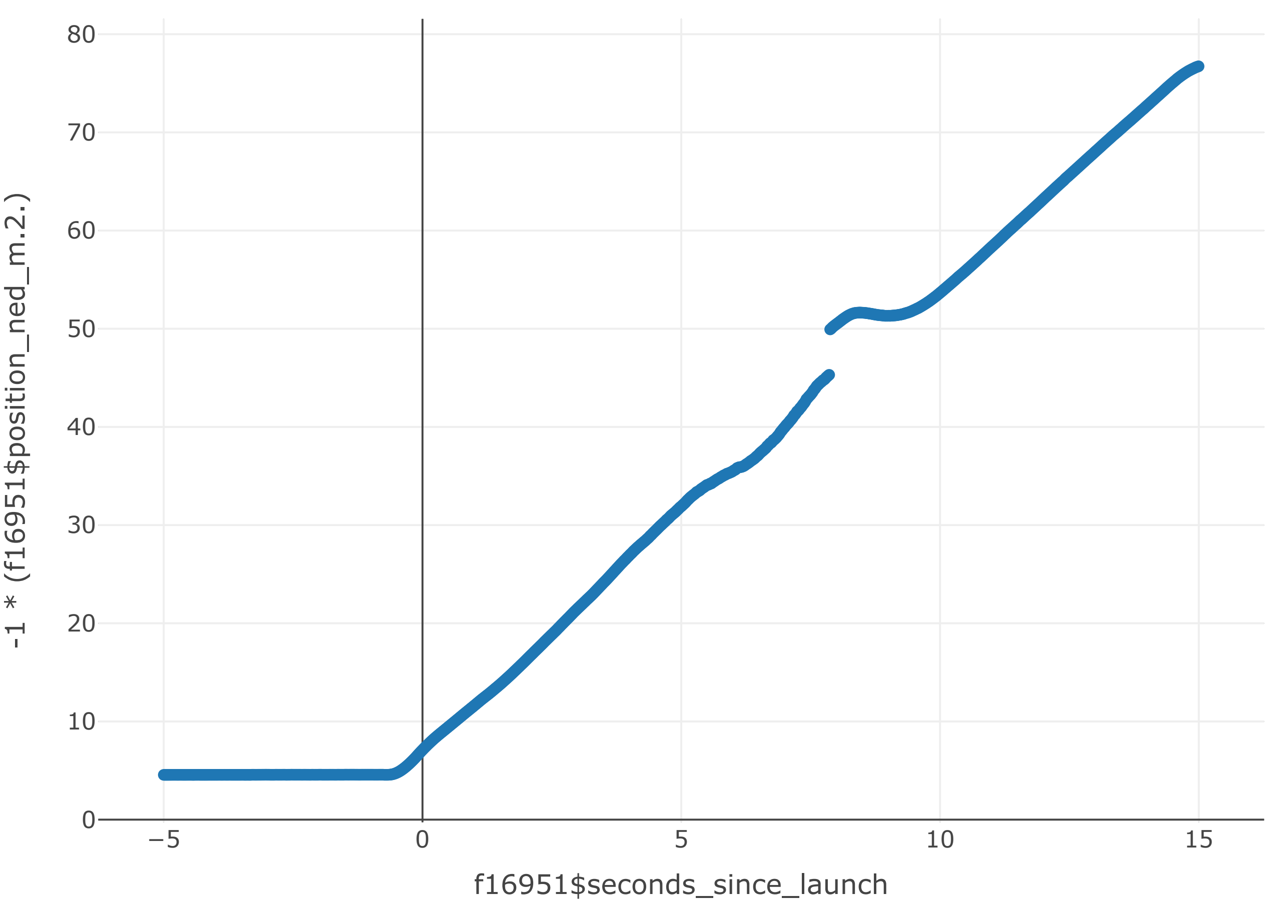
```
sumcom %>% head(.,3) %>% dim() # 33 variables andfor each flight 0 second (lauch moment).
```

```
## [1]  3 33
```

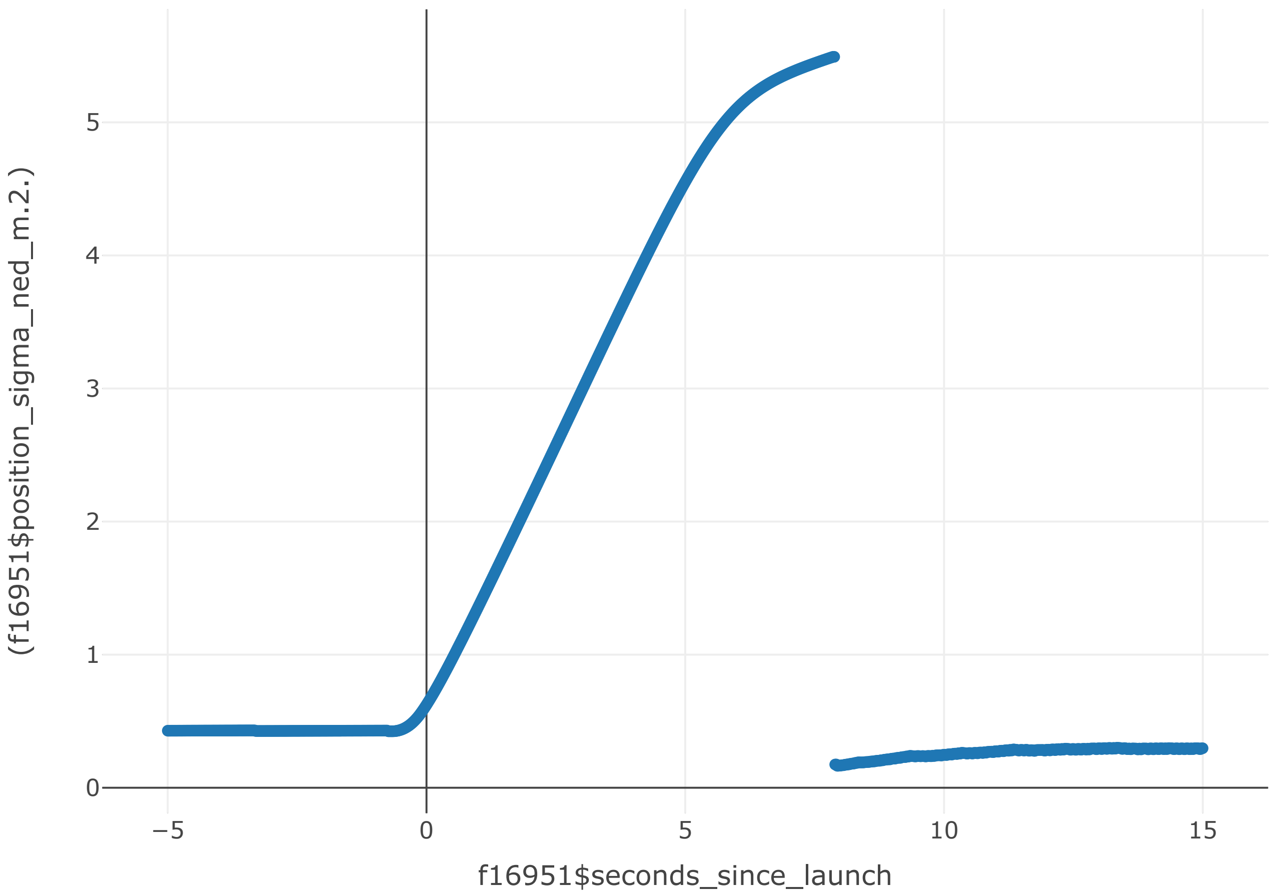
```r
### Select the individual data from each flight at the last time in record (timely last record for ea
ch flight) and combine with summary data.
filter_fun_last <- function(x) {
   myfiles[[x]][which(myfiles[[x]][1] == max(myfiles[[x]][[1]])), ]
   myfiles[[x]][which(myfiles[[x]][1] == max(myfiles[[x]][[1]])), ] = as.vector(myfiles[[x]][which(my
files[[x]][1] ==
   max(myfiles[[x]][[1]])), ])
   return(myfiles[[x]][which(myfiles[[x]][1] == max(myfiles[[x]][[1]])), ])
}

fil_last <- list(NULL)
for (i in 1:447){

   fil_last[[i]] <- filter_fun_last(i)
}
new_last <- bind_rows(fil_last)
sumcom_last <- cbind(summary_data,new_last)
glimpse(sumcom_last)   # 33 variables and for each flight at around 15 seconds.
```

```
## Observations: 447
## Variables: 33
## $ flight_id                   <int> 16951, 16952, 16954, 16955, 16957, 16…
## $ air_temperature             <dbl> 20.55000, 20.50000, 24.47502, 27.3000…
## $ battery_serial_number       <fct> 15SPJJJ09036021, 15SPJJJ10029029, 15S…
## $ body_serial_number          <dbl> 5.773501e+17, 5.772096e+17, 5.772096e…
## $ commit                      <fct> 5c504d9a16, 5c504d9a16, 5c504d9a16, 5…
## $ launch_airspeed             <dbl> 32.45345, 32.14121, 34.70188, 34.3690…
## $ launch_groundspeed          <dbl> 30.16466, 30.53525, 29.87261, 29.8776…
## $ launch_timestamp            <fct> 2018-09-06 07:43:59 CAT, 2018-09-06 0…
## $ preflight_voltage           <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, N…
## $ rel_humidity                <dbl> 74.15000, 71.17504, 66.37498, 59.0000…
## $ static_pressure             <dbl> 80662.08, 80708.07, 80774.27, 80805.1…
## $ wind_direction              <dbl> -49.434555, -4.408768, -23.458781, -4…
## $ wind_magnitude              <dbl> 1.9493382, 0.9173566, 3.7883831, 3.92…
## $ wing_serial_number          <fct> 15SPJJJ11024054, 15SPJJJ09011032, 15S…
## $ seconds_since_launch        <dbl> 14.99538, 14.99540, 14.99546, 14.9955…
## $ position_ned_m.0.           <dbl> -389.3163, -397.0894, -382.1655, -375…
## $ position_ned_m.1.           <dbl> 176.8091, 180.9998, 172.1376, 168.742…
## $ position_ned_m.2.           <dbl> -76.73076, -77.16143, -75.26395, -74.…
## $ velocity_ned_mps.0.         <dbl> -23.90914, -24.96563, -22.71749, -22.…
## $ velocity_ned_mps.1.         <dbl> 13.43833, 13.98824, 12.05545, 11.0792…
## $ velocity_ned_mps.2.         <dbl> -1.99261320, -0.75729960, -3.81525350…
## $ accel_body_mps2.0.          <dbl> 1.46991420, 1.74591980, 1.04803550, 1…
## $ accel_body_mps2.1.          <dbl> 0.24691114, 0.24927872, 0.39077517, 0…
## $ accel_body_mps2.2.          <dbl> -4.1565680, -5.3124440, -5.7719555, -…
## $ orientation_rad.0.          <dbl> -0.2851829, -0.3962580, -0.2641048, -…
## $ orientation_rad.1.          <dbl> 0.06852871, 0.07136085, 0.15472068, 0…
## $ orientation_rad.2.          <dbl> 2.599779, 2.666603, 2.659648, 2.60035…
## $ angular_rate_body_radps.0. <dbl> -0.145041330, -0.143309470, -0.103264…
## $ angular_rate_body_radps.1. <dbl> -0.1135008200, -0.0498964450, -0.1549…
## $ angular_rate_body_radps.2. <dbl> -0.08767550, -0.06086628, -0.11480521…
## $ position_sigma_ned_m.0.    <dbl> 0.129550590, 0.440122700, 0.544847550…
## $ position_sigma_ned_m.1.    <dbl> 0.225514100, 0.450459360, 0.602896870…
## $ position_sigma_ned_m.2.    <dbl> 0.296766100, 0.468056860, 0.531991300…
```

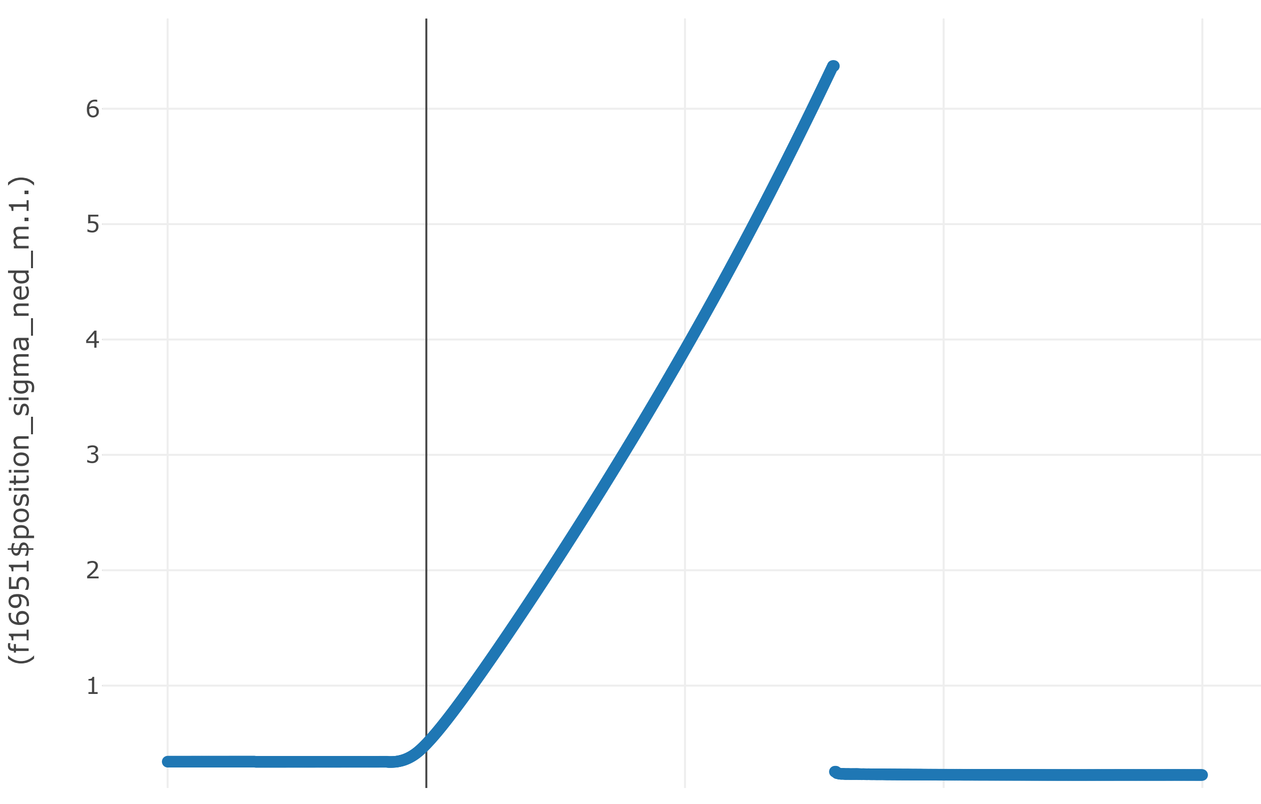
```
### Create the calculated speed based on velovity at time 0 , and time(last) respectively and add to the data frame.  speed = square root of square sum for velocity in three directions

sumcom$calculated_speed <- sqrt(sumcom$velocity_ned_mps.0.^2+sumcom$velocity_ned_mps.1.^2+sumcom$velocity_ned_mps.2.^2)

sumcom_last$calculated_speed <- sqrt(sumcom_last$velocity_ned_mps.0.^2+sumcom_last$velocity_ned_mps.1.^2+sumcom_last$velocity_ned_mps.2.^2)

### Create the distance travel (horizontally : only east/south/north/west directions are considered.) for entire period in record. Distance = square root of square sum for position change in horizontal directions

sumcom_last$distance_travel <- sqrt((sumcom$position_ned_m.0. - sumcom_last$position_ned_m.0.) ^ 2 + (sumcom$position_ned_m.1. - sumcom_last$position_ned_m.1.) ^
   2
   )
sumcom$distance_travel <- sumcom_last$distance_travel


### Create Position Errors for the entire period. Error = sum for error rate in three direction. Here I just pick one end moment for calculating the error represent the average for each trip because when you draw the graph to see errors with respect to time, the error before  is distinctly large than after around 8 seconds (From the position plot I could figure out the plane climb around 8 seconds than start flat flying.)

## Postion:
 plot_ly(f16951,x = ~f16951$seconds_since_launch, y = ~-1*(f16951$position_ned_m.2.)) %>%  add_markers() # I flip the direction to intuitive show the path of flight in  vertical direction WRT time.
```

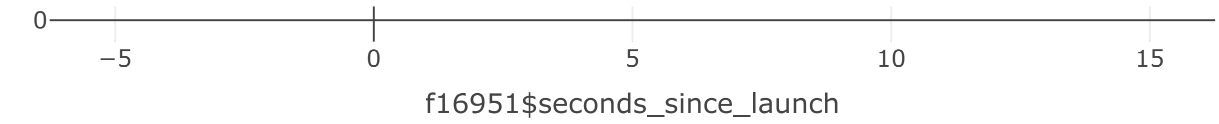


```
## Position Error:
   plot_ly(f16951,x = ~f16951$seconds_since_launch, y = ~(f16951$position_sigma_ned_m.2.)) %>%  add_m
arkers()
```

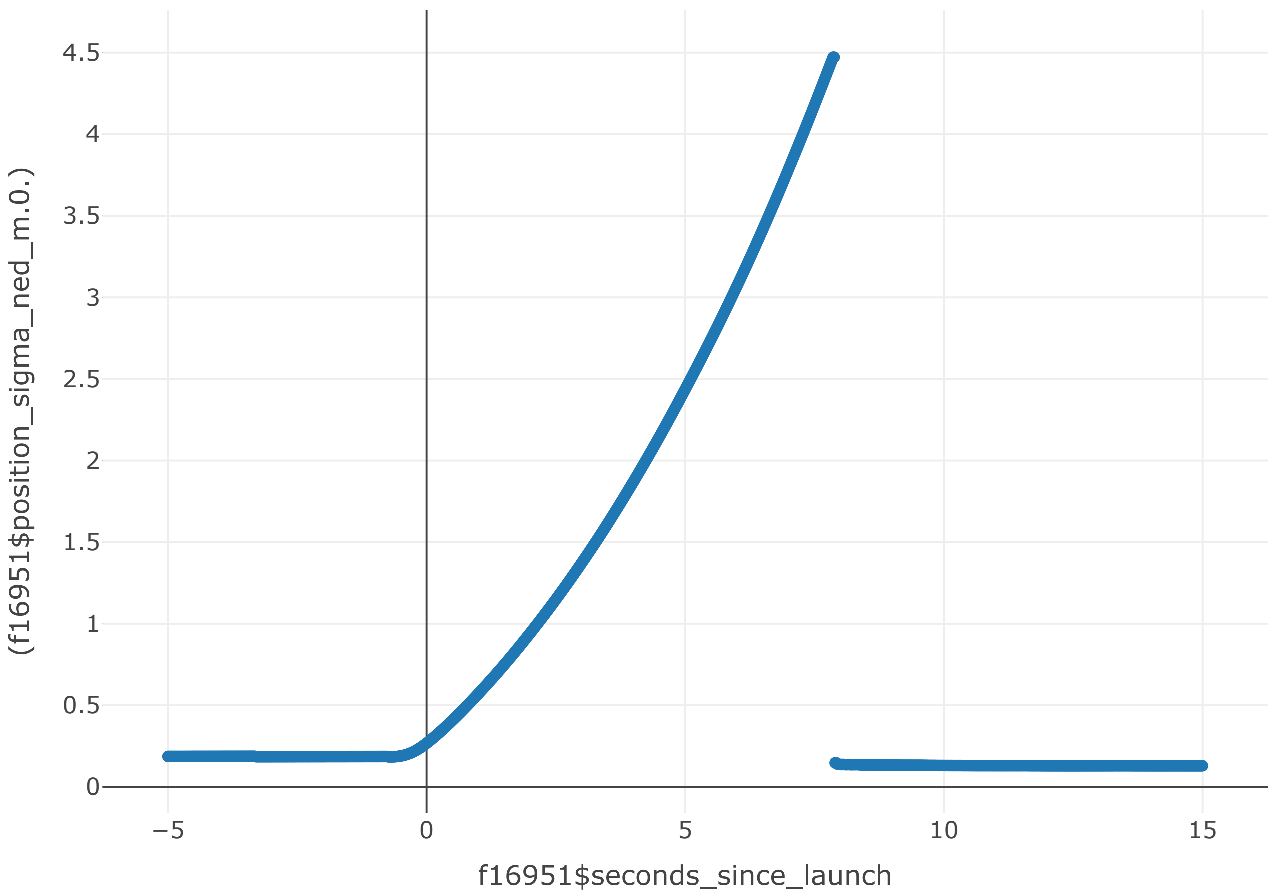



```
   plot_ly(f16951,x = ~f16951$seconds_since_launch, y = ~(f16951$position_sigma_ned_m.1.)) %>%  add_m
arkers()
```






```
plot_ly(f16951,x = ~f16951$seconds_since_launch, y = ~(f16951$position_sigma_ned_m.0.)) %>%  add_m
arkers() # You can see the error decreases a lot after around 8 secons from launching.
```




```
sumcom_last$error <- sumcom_last$position_sigma_ned_m.0.+sumcom_last$position_sigma_ned_m.1.+sumcom_l
ast$position_sigma_ned_m.2.

## Also create error at the laucnhing
sumcom$error <- sumcom$position_sigma_ned_m.0.+sumcom$position_sigma_ned_m.1.+sumcom$position_sigma_n
ed_m.2.



## Create datetime contains date only to check pattern by day.
sumcom_last$datetime <- as.Date(as.character(sumcom_last$launch_timestamp))
sumcom$datetime <- sumcom_last$datetime
glimpse(sumcom_last$datetime)
```

```
##  Date[1:447], format: "2018-09-06" "2018-09-06" "2018-09-06" "2018-09-06" "2018-09-06" ...
```

```
dim(sumcom_last)
```

```
## [1] 447  37
```

```
dim(sumcom)
```

```
## [1] 447  37
```

`Anwswer` - I create two new data sets that combine the summary_data with flights at launch time and ending time in record repectively. - Create variables that calculated speed at launching, ending time, distance travelled by each flight and overall average position errors for each flight and error at launch, I also group the time by date. - Something interest, there are sudden postion changes in the record at around 8 seconds after lauching, I guess it's probably position adjustment and the position error decreases a lot then.

Outlier and fault exploration.

```
### I would like to consider the outlier by looking at the components that makes up the main part for each flight and it's my focus to give insights.

### frequency for each components used:


table(summary_data$battery_serial_number)
```

```
##
## 15SPJJJ09010022 15SPJJJ09013015 15SPJJJ09017016 15SPJJJ09018015
##              21               4               8              24
## 15SPJJJ09036021 15SPJJJ10005031 15SPJJJ10007045 15SPJJJ10008029
##              26               3              15               7
## 15SPJJJ10012034 15SPJJJ10018016 15SPJJJ10019016 15SPJJJ10021047
##              31              10              20              21
## 15SPJJJ10022048 15SPJJJ10023027 15SPJJJ10027028 15SPJJJ10029029
##              21              19              10              27
## 15SPJJJ10030028 15SPJJJ10040016 15SPJJJ10048030 15SPJJJ10050016
##              11              22              18              26
## 15SPJJJ10050049 15SPJJJ10052026 15SPJJJ10054027 15SPJJJ10056048
##              13              12              19              18
## 15SPJJJ10060032 15SPJJJ11059037
##              18              23
```

```
table(summary_data$body_serial_number)
```

```
##
## 577209618523054080 577209618523082752 577348835878129664
##                  3                 37                 23
## 577348835962032128 577348835962105856 577348835962150912
##                 54                 14                 13
## 577348835962155008 577350132790489088 577350132790558720
##                 22                 23                 69
## 577350132807348224 577350132807356416 577350132807368704
##                 63                  3                 20
## 577350132807389184 577350132840857600 577350132840894464
##                 26                 45                 32
```

```
table(summary_data$wing_serial_number)
```

```
##
## 15SPJJJ09008034 15SPJJJ09010032 15SPJJJ09011032 15SPJJJ09019061
##              65               5              11              23
## 15SPJJJ09021032 15SPJJJ09024061 15SPJJJ09025064 15SPJJJ09028034
##               8              45              58              14
## 15SPJJJ09028064 15SPJJJ09031032 15SPJJJ09032034 15SPJJJ09036063
##               4              27              15              17
## 15SPJJJ09040032 15SPJJJ09043062 15SPJJJ09052035 15SPJJJ11024054
##              44              22              51              13
## 15SPJJJ11048054 15SPJJJ11049056
##               1              24
```

`Answer` - I would suggest try using the components with similar frequencies. Wing: 15SPJJJ11048054 only used one time and wing: 15SPJJJ09008034 was used 65 time might cause overuse.

```
### which has NA's and NA frequency in each part
cat("battery")
```

```
## battery
```

```
battery <- summary_data$battery_serial_number[is.na(summary_data$preflight_voltage)] %>% table() %>%
print()
```

```
## .
## 15SPJJJ09010022 15SPJJJ09013015 15SPJJJ09017016 15SPJJJ09018015
##               0               1               1               1
## 15SPJJJ09036021 15SPJJJ10005031 15SPJJJ10007045 15SPJJJ10008029
##               3               0               0               1
## 15SPJJJ10012034 15SPJJJ10018016 15SPJJJ10019016 15SPJJJ10021047
##               1               0               0               0
## 15SPJJJ10022048 15SPJJJ10023027 15SPJJJ10027028 15SPJJJ10029029
##               0               1               0               2
## 15SPJJJ10030028 15SPJJJ10040016 15SPJJJ10048030 15SPJJJ10050016
##               0               0               0               0
## 15SPJJJ10050049 15SPJJJ10052026 15SPJJJ10054027 15SPJJJ10056048
##               2               2               1               0
## 15SPJJJ10060032 15SPJJJ11059037
##               0               0
```

```
cat("body")
```

```
## body
```

```
body <- summary_data$body_serial_number[is.na(summary_data$preflight_voltage)] %>% table() %>% print(
)
```

```
## .
## 577209618523054080 577209618523082752 577348835962150912
##                  3                  3                  3
## 577350132807348224 577350132840857600
##                  4                  3
```

```
cat("wing")
```

```
## wing
```

```
wing <- summary_data$wing_serial_number[is.na(summary_data$preflight_voltage)] %>% table() %>% print(
)
```

```
## .
## 15SPJJJ09008034 15SPJJJ09010032 15SPJJJ09011032 15SPJJJ09019061
##               0               0               3               0
## 15SPJJJ09021032 15SPJJJ09024061 15SPJJJ09025064 15SPJJJ09028034
##               0               0               1               0
## 15SPJJJ09028064 15SPJJJ09031032 15SPJJJ09032034 15SPJJJ09036063
##               0               3               0               0
## 15SPJJJ09040032 15SPJJJ09043062 15SPJJJ09052035 15SPJJJ11024054
##               0               0               0               4
## 15SPJJJ11048054 15SPJJJ11049056
##               0               5
```

`Anwswer` - body as following have create missing values in pre-flight voltage : - 577209618523054080, 577209618523082752, 577348835962150912, 577350132840857600 = 3 times - 577350132807348224 = 4 times.

```
#### Percentage NA for each component during use
cat("battery",'\n')
```

```
## battery
```

```
per_battery <- as.numeric(battery)/as.numeric(table(summary_data$battery_serial_number))
print(per_battery)
```

```
##  [1] 0.00000000 0.25000000 0.12500000 0.04166667 0.11538462 0.00000000
##  [7] 0.00000000 0.14285714 0.03225806 0.00000000 0.00000000 0.00000000
## [13] 0.00000000 0.05263158 0.00000000 0.07407407 0.00000000 0.00000000
## [19] 0.00000000 0.00000000 0.15384615 0.16666667 0.05263158 0.00000000
## [25] 0.00000000 0.00000000
```

```
cat("body",'\n')
```

```
## body
```

```
per_body <- as.numeric(body)/as.numeric(table(summary_data$body_serial_number))
print(per_body)
```

```
##  [1] 1.00000000 0.08108108 0.13043478 0.07407407 0.21428571 0.23076923
##  [7] 0.13636364 0.13043478 0.05797101 0.04761905 1.00000000 0.15000000
## [13] 0.11538462 0.08888889 0.09375000
```

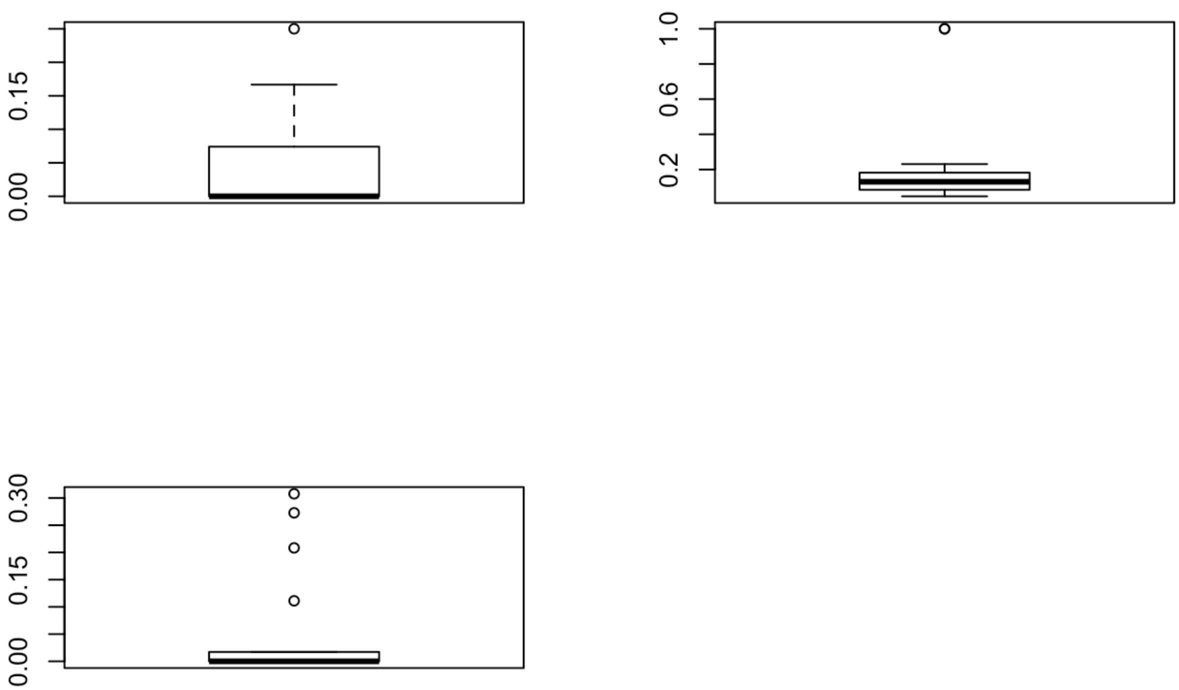
```
cat("wing",'\n')
```

```
## wing
```

```
per_wing <- as.numeric(wing)/as.numeric(table(summary_data$wing_serial_number))
print(per_wing)
```

```
##  [1] 0.00000000 0.00000000 0.27272727 0.00000000 0.00000000 0.00000000
##  [7] 0.01724138 0.00000000 0.00000000 0.11111111 0.00000000 0.00000000
## [13] 0.00000000 0.00000000 0.00000000 0.30769231 0.00000000 0.20833333
```

```
par(mfrow=c(2,2))
boxplot(per_battery)
boxplot(per_body)
boxplot(per_wing)
```

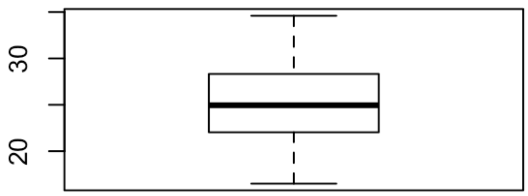


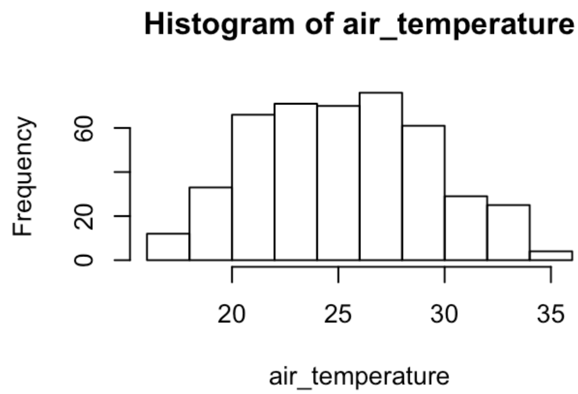
`Answer` - battery:15SPJJJ09013015 has higher NA percentage around 25% - body: 577209618523054080 is used 3 times and all has NA pre-flight voltage 100%. - wings also have but I assume there is no relation between wings and voltage.

```
### Explore each variable
### air temperature
attach(summary_data,warn.conflicts = F)
par(mfrow=c(2,2))
summary(air_temperature) # no outlier temp
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   16.50   22.04   24.95   25.23   28.32   34.60
```

```
boxplot(air_temperature)
hist(air_temperature)
```



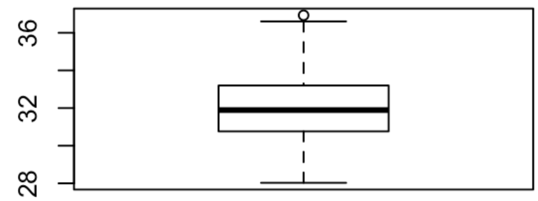


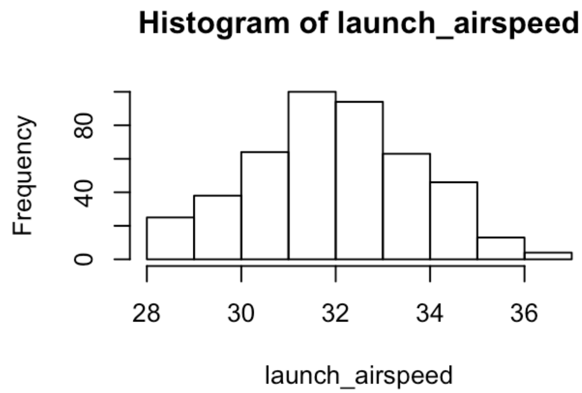

```
### airspeed
par(mfrow=c(2,2))
summary(launch_airspeed) # one possible outlier
```

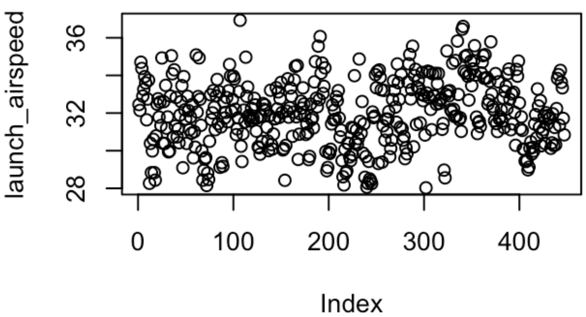
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   28.03   30.76   31.89   31.98   33.20   36.93
```

```
boxplot(launch_airspeed)
hist(launch_airspeed)
plot(launch_airspeed)
summary_data[(max(launch_airspeed)),] # flight_id 17028 has highest airspeed and possible outlier. It
has battery_serial_number 15SPJJJ10012034; body_serial_number 5.773488e+17 wing_serial_number 15SPJJJ
09011032
```

```
##    flight_id air_temperature battery_serial_number body_serial_number
## 36     17028           26.25       15SPJJJ10012034       5.773488e+17
##        commit launch_airspeed launch_groundspeed        launch_timestamp
## 36 5c504d9a16        34.08525           29.86218 2018-09-08 10:47:11 CAT
##    preflight_voltage rel_humidity static_pressure wind_direction
## 36           31.7047        58.85        80676.95      -13.32951
##    wind_magnitude wing_serial_number
## 36       3.191452    15SPJJJ09011032
```



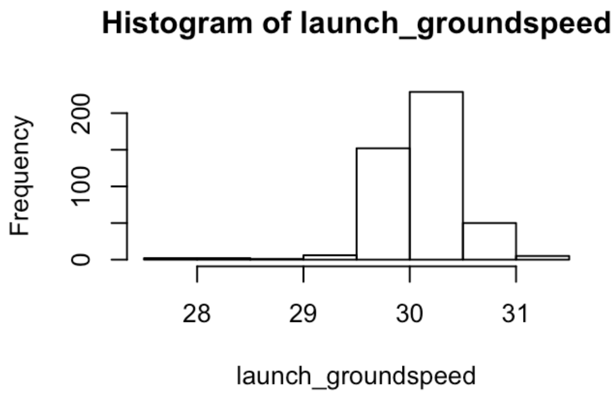


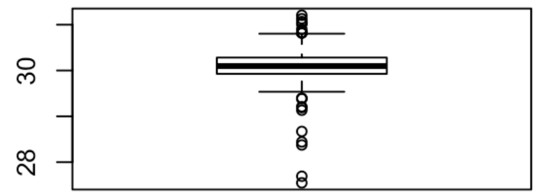



```
### groundaspeed
par(mfrow=c(2,2))
hist(launch_groundspeed)
summary(launch_groundspeed) # one possible outlier
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   27.55   29.93   30.10   30.11   30.28   31.21
```

```
boxplot(launch_groundspeed) # more potential outliers because of the wind
```






```
### launch_timestamp
launch_timestamp %>% class()
```

```
## [1] "factor"
```

```
levels(launch_timestamp ) %>% length() # time early -- late inorder with flight number in ascending o
rder
```

```
## [1] 447
```

```
### preflight_voltage
par(mfrow=c(2,2))
summary(preflight_voltage)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   31.54   32.06   32.19   32.15   32.27   32.52      16
```

```
boxplot(preflight_voltage)
summary_data[which(summary_data$preflight_voltage<32.06),c(1,3,4,14)]
```

```
##     flight_id battery_serial_number body_serial_number wing_serial_number
## 19      16989       15SPJJJ09013015       5.773501e+17    15SPJJJ11024054
## 21      16991       15SPJJJ10060032       5.773488e+17    15SPJJJ11024054
## 22      16994       15SPJJJ09036021       5.772096e+17    15SPJJJ09031032
## 28      17013       15SPJJJ10018016       5.773501e+17    15SPJJJ09021032
## 32      17020       15SPJJJ10005031       5.773501e+17    15SPJJJ11049056
```

```
## 34     17025   15SPJJJ10029029   5.773488e+17   15SPJJJ11049056
## 36     17028   15SPJJJ10012034   5.773488e+17   15SPJJJ09011032
## 37     17029   15SPJJJ10052026   5.772096e+17   15SPJJJ11049056
## 43     17043   15SPJJJ10050049   5.773501e+17   15SPJJJ09032034
## 45     17045   15SPJJJ09036021   5.772096e+17   15SPJJJ09021032
## 47     17056   15SPJJJ10027028   5.772096e+17   15SPJJJ11024054
## 48     17057   15SPJJJ10048030   5.773501e+17   15SPJJJ11049056
## 57     17084   15SPJJJ09013015   5.772096e+17   15SPJJJ09032034
## 62     17095   15SPJJJ10030028   5.773501e+17   15SPJJJ11024054
## 70     17103   15SPJJJ10060032   5.773501e+17   15SPJJJ09043062
## 71     17105   15SPJJJ11059037   5.773501e+17   15SPJJJ11024054
## 75     17110   15SPJJJ10027028   5.773488e+17   15SPJJJ09032034
## 76     17112   15SPJJJ10050016   5.773501e+17   15SPJJJ09021032
## 77     17113   15SPJJJ10012034   5.773488e+17   15SPJJJ11024054
## 78     17114   15SPJJJ10048030   5.773501e+17   15SPJJJ09052035
## 79     17115   15SPJJJ10005031   5.773501e+17   15SPJJJ09043062
## 90     17134   15SPJJJ10048030   5.773501e+17   15SPJJJ09052035
## 91     17136   15SPJJJ10027028   5.773501e+17   15SPJJJ09043062
## 100    17150   15SPJJJ09010022   5.773501e+17   15SPJJJ09043062
## 106    17161   15SPJJJ09017016   5.773501e+17   15SPJJJ09010032
## 108    17163   15SPJJJ10040016   5.773501e+17   15SPJJJ11049056
## 110    17165   15SPJJJ10012034   5.773501e+17   15SPJJJ09052035
## 115    17174   15SPJJJ10027028   5.773488e+17   15SPJJJ09052035
## 116    17175   15SPJJJ09018015   5.773501e+17   15SPJJJ09008034
## 117    17176   15SPJJJ10050016   5.773501e+17   15SPJJJ09028034
## 120    17179   15SPJJJ10012034   5.773501e+17   15SPJJJ09025064
## 124    17184   15SPJJJ10040016   5.773501e+17   15SPJJJ09025064
## 126    17190   15SPJJJ10012034   5.773488e+17   15SPJJJ09008034
## 127    17191   15SPJJJ10022048   5.773501e+17   15SPJJJ09028034
## 128    17192   15SPJJJ10050016   5.773501e+17   15SPJJJ09028064
## 129    17193   15SPJJJ10019016   5.773501e+17   15SPJJJ09025064
## 130    17195   15SPJJJ10056048   5.773501e+17   15SPJJJ09028064
## 133    17201   15SPJJJ10008029   5.773501e+17   15SPJJJ09043062
## 134    17202   15SPJJJ10023027   5.773501e+17   15SPJJJ09008034
## 139    17224   15SPJJJ10012034   5.773501e+17   15SPJJJ09043062
## 142    17231   15SPJJJ10060032   5.773501e+17   15SPJJJ09036063
## 144    17233   15SPJJJ10054027   5.773501e+17   15SPJJJ09025064
## 146    17235   15SPJJJ10048030   5.773488e+17   15SPJJJ09008034
## 147    17236   15SPJJJ09036021   5.773501e+17   15SPJJJ09043062
## 149    17239   15SPJJJ11059037   5.773501e+17   15SPJJJ09052035
## 150    17240   15SPJJJ10018016   5.772096e+17   15SPJJJ09008034
## 156    17251   15SPJJJ10060032   5.773501e+17   15SPJJJ09025064
## 159    17256   15SPJJJ10054027   5.773501e+17   15SPJJJ09024061
## 169    17278   15SPJJJ09017016   5.773501e+17   15SPJJJ09043062
## 175    17286   15SPJJJ10050016   5.773501e+17   15SPJJJ09024061
## 176    17287   15SPJJJ10021047   5.773488e+17   15SPJJJ09024061
## 177    17289   15SPJJJ10012034   5.772096e+17   15SPJJJ09025064
## 178    17292   15SPJJJ10018016   5.773501e+17   15SPJJJ09019061
## 179    17298   15SPJJJ10060032   5.773501e+17   15SPJJJ09043062
## 192    17316   15SPJJJ10054027   5.773501e+17   15SPJJJ09024061
## 201    17327   15SPJJJ10029029   5.773501e+17   15SPJJJ09008034
## 210    17341   15SPJJJ10012034   5.773488e+17   15SPJJJ09025064
## 213    17345   15SPJJJ10021047   5.773501e+17   15SPJJJ09008034
## 216    17349   15SPJJJ11059037   5.772096e+17   15SPJJJ09052035
## 218    17351   15SPJJJ10012034   5.773488e+17   15SPJJJ09008034
## 220    17354   15SPJJJ09010022   5.773501e+17   15SPJJJ09024061
## 221    17355   15SPJJJ10012034   5.773488e+17   15SPJJJ09008034
## 222    17356   15SPJJJ10056048   5.773501e+17   15SPJJJ09019061
## 235    17398   15SPJJJ10060032   5.773488e+17   15SPJJJ09024061
```

```
## 236     17399      15SPJJJ09018015      5.773501e+17    15SPJJJ09028064
## 241     17411      15SPJJJ10018016      5.772096e+17    15SPJJJ09008034
## 246     17418      15SPJJJ10060032      5.773488e+17    15SPJJJ09040032
## 251     17429      15SPJJJ10029029      5.773488e+17    15SPJJJ09040032
## 254     17438      15SPJJJ10022048      5.773501e+17    15SPJJJ09040032
## 255     17439      15SPJJJ10023027      5.773488e+17    15SPJJJ09019061
## 265     17461      15SPJJJ10023027      5.773488e+17    15SPJJJ09028034
## 266     17462      15SPJJJ09036021      5.773488e+17    15SPJJJ09008034
## 271     17476      15SPJJJ10048030      5.773488e+17    15SPJJJ09036063
## 274     17480      15SPJJJ10022048      5.773501e+17    15SPJJJ09008034
## 275     17483      15SPJJJ10012034      5.773488e+17    15SPJJJ09025064
## 282     17502      15SPJJJ09036021      5.773501e+17    15SPJJJ09008034
## 285     17508      15SPJJJ10019016      5.773488e+17    15SPJJJ09036063
## 300     17532      15SPJJJ10048030      5.773488e+17    15SPJJJ09025064
## 310     17553      15SPJJJ10048030      5.773488e+17    15SPJJJ09008034
## 311     17554      15SPJJJ09036021      5.773501e+17    15SPJJJ09025064
## 312     17556      15SPJJJ10040016      5.773501e+17    15SPJJJ09025064
## 319     17568      15SPJJJ10012034      5.773501e+17    15SPJJJ09008034
## 323     17576      15SPJJJ11059037      5.773501e+17    15SPJJJ09008034
## 324     17577      15SPJJJ10050016      5.773488e+17    15SPJJJ09040032
## 325     17578      15SPJJJ10022048      5.773488e+17    15SPJJJ09052035
## 332     17586      15SPJJJ11059037      5.773501e+17    15SPJJJ09008034
## 334     17589      15SPJJJ10056048      5.773501e+17    15SPJJJ09040032
## 335     17590      15SPJJJ10019016      5.773501e+17    15SPJJJ09040032
## 339     17594      15SPJJJ10056048      5.773501e+17    15SPJJJ09040032
## 343     17599      15SPJJJ10012034      5.773488e+17    15SPJJJ09040032
## 344     17600      15SPJJJ11059037      5.773501e+17    15SPJJJ09008034
## 347     17603      15SPJJJ10023027      5.773488e+17    15SPJJJ09052035
## 377     17648      15SPJJJ09018015      5.773501e+17    15SPJJJ09008034
## 383     17654      15SPJJJ10050016      5.773488e+17    15SPJJJ09040032
## 386     17657      15SPJJJ10060032      5.773501e+17    15SPJJJ09031032
## 393     17666      15SPJJJ10023027      5.773488e+17    15SPJJJ09040032
## 405     17681      15SPJJJ10052026      5.773488e+17    15SPJJJ09028034
## 406     17682      15SPJJJ10022048      5.773488e+17    15SPJJJ09019061
## 410     17687      15SPJJJ10023027      5.773501e+17    15SPJJJ09031032
## 411     17688      15SPJJJ10056048      5.773501e+17    15SPJJJ09025064
## 414     17691      15SPJJJ10048030      5.773501e+17    15SPJJJ09028034
## 417     17698      15SPJJJ10012034      5.773488e+17    15SPJJJ09024061
## 418     17699      15SPJJJ10060032      5.773501e+17    15SPJJJ09031032
## 419     17700      15SPJJJ10019016      5.773501e+17    15SPJJJ09031032
## 423     17705      15SPJJJ10060032      5.773501e+17    15SPJJJ09019061
## 427     17714      15SPJJJ10023027      5.773501e+17    15SPJJJ09040032
## 430     17717      15SPJJJ10012034      5.773501e+17    15SPJJJ09019061
## 433     17723      15SPJJJ10060032      5.773501e+17    15SPJJJ09024061
## 434     17724      15SPJJJ10007045      5.773501e+17    15SPJJJ09031032
## 435     17725      15SPJJJ09018015      5.773501e+17    15SPJJJ09024061
## 436     17726      15SPJJJ10012034      5.773501e+17    15SPJJJ09008034
```

```
table(summary_data[which(summary_data$preflight_voltage<32.06),c(3)] )
```

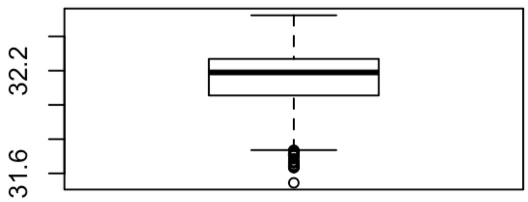
```
##
## 15SPJJJ09010022 15SPJJJ09013015 15SPJJJ09017016 15SPJJJ09018015
##               2               2               2               4
## 15SPJJJ09036021 15SPJJJ10005031 15SPJJJ10007045 15SPJJJ10008029
##               6               2               1               1
## 15SPJJJ10012034 15SPJJJ10018016 15SPJJJ10019016 15SPJJJ10021047
##              16               4               4               2
## 15SPJJJ10022048 15SPJJJ10023027 15SPJJJ10027028 15SPJJJ10029029
##               5               7               4               3
## 15SPJJJ10030028 15SPJJJ10040016 15SPJJJ10048030 15SPJJJ10050016
##               1               3               8               6
## 15SPJJJ10050049 15SPJJJ10052026 15SPJJJ10054027 15SPJJJ10056048
##               1               2               3               5
## 15SPJJJ10060032 15SPJJJ11059037
##              11               6
```

```
table(summary_data[which(summary_data$preflight_voltage<32.06),c(4)] )
```

```
##
## 577209618523082752 577348835878129664 577348835962032128
##                  9                  7                 14
## 577348835962105856 577348835962150912 577348835962155008
##                  6                  2                  3
## 577350132790489088 577350132790558720 577350132807348224
##                  7                 10                 13
## 577350132807356416 577350132807368704 577350132807389184
##                  1                  8                  7
## 577350132840857600 577350132840894464
##                 12                 12
```

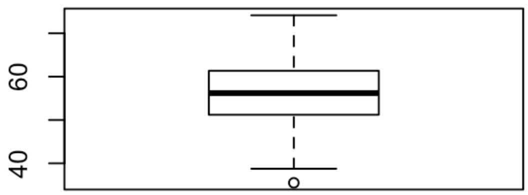
```
table(summary_data[which(summary_data$preflight_voltage<32.06),c(14)] )
```

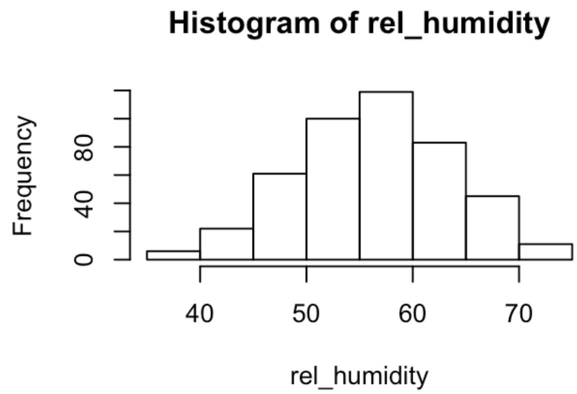
```
##
## 15SPJJJ09008034 15SPJJJ09010032 15SPJJJ09011032 15SPJJJ09019061
##              20               1               1               6
## 15SPJJJ09021032 15SPJJJ09024061 15SPJJJ09025064 15SPJJJ09028034
##               3               9              12               5
## 15SPJJJ09028064 15SPJJJ09031032 15SPJJJ09032034 15SPJJJ09036063
##               3               6               3               3
## 15SPJJJ09040032 15SPJJJ09043062 15SPJJJ09052035 15SPJJJ11024054
##              11               9               8               6
## 15SPJJJ11048054 15SPJJJ11049056
##               0               5
```



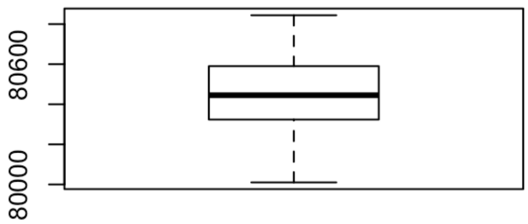

```
### rel_humidity
# summary(rel_humidity)
par(mfrow=c(2,2))
boxplot(rel_humidity)
hist(rel_humidity)
summary_data[which(summary_data$rel_humidity==35.50),]
```

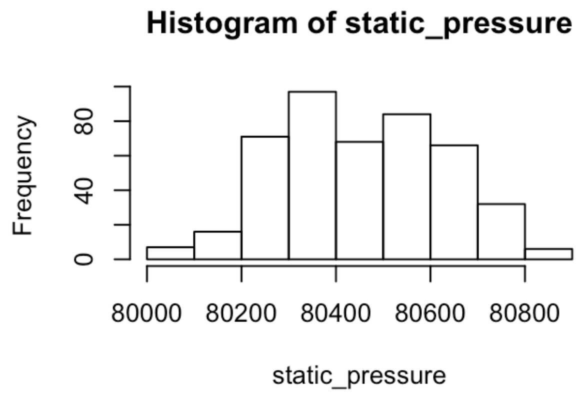
```
##     flight_id air_temperature battery_serial_number body_serial_number
## 258     17450           32.55       15SPJJJ10030028       5.773488e+17
##         commit launch_airspeed launch_groundspeed        launch_timestamp
## 258 4d9468bd3c        33.20337           30.01733 2018-09-25 16:46:28 CAT
##     preflight_voltage rel_humidity static_pressure wind_direction
## 258          32.37127         35.5         80269.3      -29.25491
##     wind_magnitude wing_serial_number
## 258        2.34973    15SPJJJ09008034
```



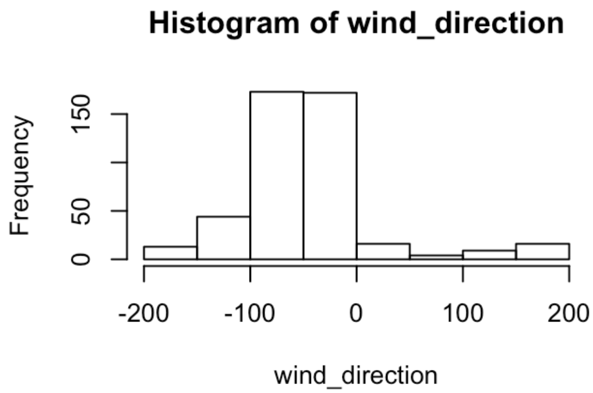



```
### static_pressure
# summary(static_pressure)
par(mfrow=c(2,2))
boxplot(static_pressure)  # no outlier
hist(static_pressure)
```






```
# wind_direction
par(mfrow=c(2,2))
# summary(wind_direction)
hist(wind_direction)
# boxplot(wind_direction) that boxplot doesn't make sense

# wind_magnitude
par(mfrow=c(2,2))
```




```
# summary(wind_magnitude)
boxplot(wind_magnitude)
hist(wind_magnitude)
summary_data[which(summary_data$wind_magnitude>5),]
```

```
##     flight_id air_temperature battery_serial_number body_serial_number
## 96      17145           24.05       15SPJJJ09036021       5.773501e+17
## 107     17162           25.80       15SPJJJ09018015       5.773501e+17
## 167     17274           31.35       15SPJJJ10048030       5.773501e+17
##         commit launch_airspeed launch_groundspeed        launch_timestamp
## 96  5c504d9a16        34.58137           29.84380 2018-09-12 08:55:56 CAT
## 107 5c504d9a16        36.92920           29.61042 2018-09-12 16:58:38 CAT
## 167 5c504d9a16        34.53072           29.98681 2018-09-17 16:23:47 CAT
##     preflight_voltage rel_humidity static_pressure wind_direction
## 96           32.12171        64.35        80563.51      -12.85778
## 107          32.19193        58.50        80252.84      -56.52143
## 167          32.16600        52.70        80111.57      -87.36706
##     wind_magnitude wing_serial_number
## 96        5.275389    15SPJJJ09052035
## 107       7.466193    15SPJJJ09052035
## 167       5.486348    15SPJJJ09052035
```

```
summary_data[92:109,] # look if that day has stronger wind 2018-09-12
```

```
##     flight_id air_temperature battery_serial_number body_serial_number
## 92      17139           22.65       15SPJJJ10022048       5.773501e+17
## 93      17140           23.25       15SPJJJ09017016       5.773501e+17
## 94      17141           25.00       15SPJJJ10018016       5.773488e+17
## 95      17144           22.80       15SPJJJ09010022       5.773501e+17
```

```
## 96     17145          24.05      15SPJJJ09036021      5.773501e+17
## 97     17146          24.50      15SPJJJ10048030      5.773501e+17
## 98     17147          23.85      15SPJJJ10040016      5.773501e+17
## 99     17148          26.55      15SPJJJ10012034      5.773488e+17
## 100    17150          26.75      15SPJJJ09010022      5.773501e+17
## 101    17151          29.65      15SPJJJ10022048      5.773501e+17
## 102    17152          30.90      15SPJJJ09018015      5.773501e+17
## 103    17155          27.70      15SPJJJ10022048      5.773501e+17
## 104    17157          27.35      15SPJJJ09010022      5.773501e+17
## 105    17160          27.25      15SPJJJ10012034      5.773501e+17
## 106    17161          28.50      15SPJJJ09017016      5.773501e+17
## 107    17162          25.80      15SPJJJ09018015      5.773501e+17
## 108    17163          20.35      15SPJJJ10040016      5.773501e+17
## 109    17164          19.85      15SPJJJ10050016      5.773501e+17
##        commit launch_airspeed launch_groundspeed       launch_timestamp
## 92  5c504d9a16        32.03880           30.21231 2018-09-12 07:34:11 CAT
## 93  5c504d9a16        32.82465           30.15264 2018-09-12 07:40:49 CAT
## 94  5c504d9a16        31.53479           30.02592 2018-09-12 07:53:36 CAT
## 95  5c504d9a16        33.18635           30.12064 2018-09-12 08:15:15 CAT
## 96  5c504d9a16        34.58137           29.84380 2018-09-12 08:55:56 CAT
## 97  5c504d9a16        33.96659           29.57980 2018-09-12 09:03:07 CAT
## 98  5c504d9a16        32.08388           29.83460 2018-09-12 09:13:46 CAT
## 99  5c504d9a16        32.73053           29.78764 2018-09-12 10:22:00 CAT
## 100 5c504d9a16        33.95885           29.77808 2018-09-12 11:57:41 CAT
## 101 5c504d9a16        33.81445           30.09764 2018-09-12 12:01:31 CAT
## 102 5c504d9a16        32.20170           30.20247 2018-09-12 12:09:06 CAT
## 103 5c504d9a16        30.84962           29.92542 2018-09-12 15:24:56 CAT
## 104 5c504d9a16        31.15663           30.35397 2018-09-12 16:03:41 CAT
## 105 5c504d9a16        31.74609           29.95693 2018-09-12 16:07:26 CAT
## 106 5c504d9a16        33.14270           30.05171 2018-09-12 16:25:56 CAT
## 107 5c504d9a16        36.92920           29.61042 2018-09-12 16:58:38 CAT
## 108 5c504d9a16        30.23961           30.21643 2018-09-12 17:23:27 CAT
## 109 5c504d9a16        29.42163           30.53867 2018-09-12 17:31:46 CAT
##     preflight_voltage rel_humidity static_pressure wind_direction
## 92           32.11091     66.50000        80524.66      -1.646400
## 93           32.30321     65.55000        80687.19      -1.896772
## 94           32.11523     61.55000        80692.21     -14.649196
## 95           32.25027     66.45000        80590.36      -5.493722
## 96           32.12171     64.35000        80563.51     -12.857783
## 97           32.10118     64.12499        80690.82     -10.239765
## 98           32.18221     61.40000        80777.44     -16.475939
## 99           32.12928     58.02499        80751.77     -34.904701
## 100          31.87864     62.35000        80643.68     -43.332546
## 101          32.19302     60.47499        80474.69     -39.403119
## 102          32.29132     57.00000        80470.61     -66.839478
## 103          32.23515     57.65000        80391.63     -52.287143
## 104          32.35614     60.85000        80377.89     -76.884744
## 105          32.11307     52.57501        80395.21     -56.920556
## 106          32.02772     45.80000        80282.11     -39.383782
## 107          32.19193     58.50000        80252.84     -56.521431
## 108          31.73603     55.10000        80493.76    -109.681997
## 109          32.07418     60.70000        80509.40     166.625103
##     wind_magnitude wing_serial_number
## 92       2.0215997    15SPJJJ09052035
## 93       2.6392558    15SPJJJ09043062
## 94       2.9268354    15SPJJJ11049056
## 95       3.5281443    15SPJJJ09052035
## 96       5.2753888    15SPJJJ09052035
## 97       3.7830189    15SPJJJ09043062
```

```
## 98       4.1461344    15SPJJJ11049056
## 99       4.0830937    15SPJJJ11049056
## 100      2.9852745    15SPJJJ09043062
## 101      3.2225291    15SPJJJ09052035
## 102      2.8555161    15SPJJJ09025064
## 103      1.6849935    15SPJJJ09028034
## 104      2.2172968    15SPJJJ09043062
## 105      2.3098278    15SPJJJ11049056
## 106      2.4970023    15SPJJJ09010032
## 107      7.4661926    15SPJJJ09052035
## 108      0.9403838    15SPJJJ11049056
## 109      1.7613523    15SPJJJ09028034
```

```
summary_data[which(summary_data$wind_magnitude>6),] # wind_magnitude 7.466193 for flight 17162   with index 107
```

```
##     flight_id air_temperature battery_serial_number body_serial_number
## 107     17162            25.8       15SPJJJ09018015       5.773501e+17
##         commit launch_airspeed launch_groundspeed        launch_timestamp
## 107 5c504d9a16         36.9292           29.61042 2018-09-12 16:58:38 CAT
##     preflight_voltage rel_humidity static_pressure wind_direction
## 107          32.19193         58.5        80252.84      -56.52143
##     wind_magnitude wing_serial_number
## 107       7.466193    15SPJJJ09052035
```

```
# sumcom_last$distance_travel
plot(y=as.numeric(tapply(sumcom_last$distance_travel,sumcom_last$datetime,mean)),x=c(1:30),type = "bar")
```

```
## Warning in plot.xy(xy, type, ...): plot type 'bar' will be truncated to
## first character
```

```
tapply(sumcom_last$distance_travel,sumcom_last$datetime,mean)
```

```
## 2018-09-06 2018-09-07 2018-09-08 2018-09-09 2018-09-10 2018-09-11
##   436.5852   420.8661   420.5463   421.7589   435.7925   417.3416
## 2018-09-12 2018-09-13 2018-09-14 2018-09-15 2018-09-16 2018-09-17
##   414.0976   424.0795   428.0676   430.4087   433.3155   424.7380
## 2018-09-18 2018-09-19 2018-09-20 2018-09-21 2018-09-22 2018-09-23
##   428.5770   427.0411   438.8954   428.8350   426.1259   465.2436
## 2018-09-24 2018-09-25 2018-09-26 2018-09-27 2018-09-28 2018-09-29
##   424.8297   433.0503   427.0050   419.1462   428.1857   420.3933
## 2018-09-30 2018-10-01 2018-10-02 2018-10-03 2018-10-04 2018-10-05
##   423.2996   416.7704   422.9500   430.6660   429.2154   422.7038
```

```
tapply(sumcom_last$distance_travel,sumcom_last$datetime,mean) %>% max() # 2018-09-23 travels averagely longest distance.
```

```
## [1] 465.2436
```

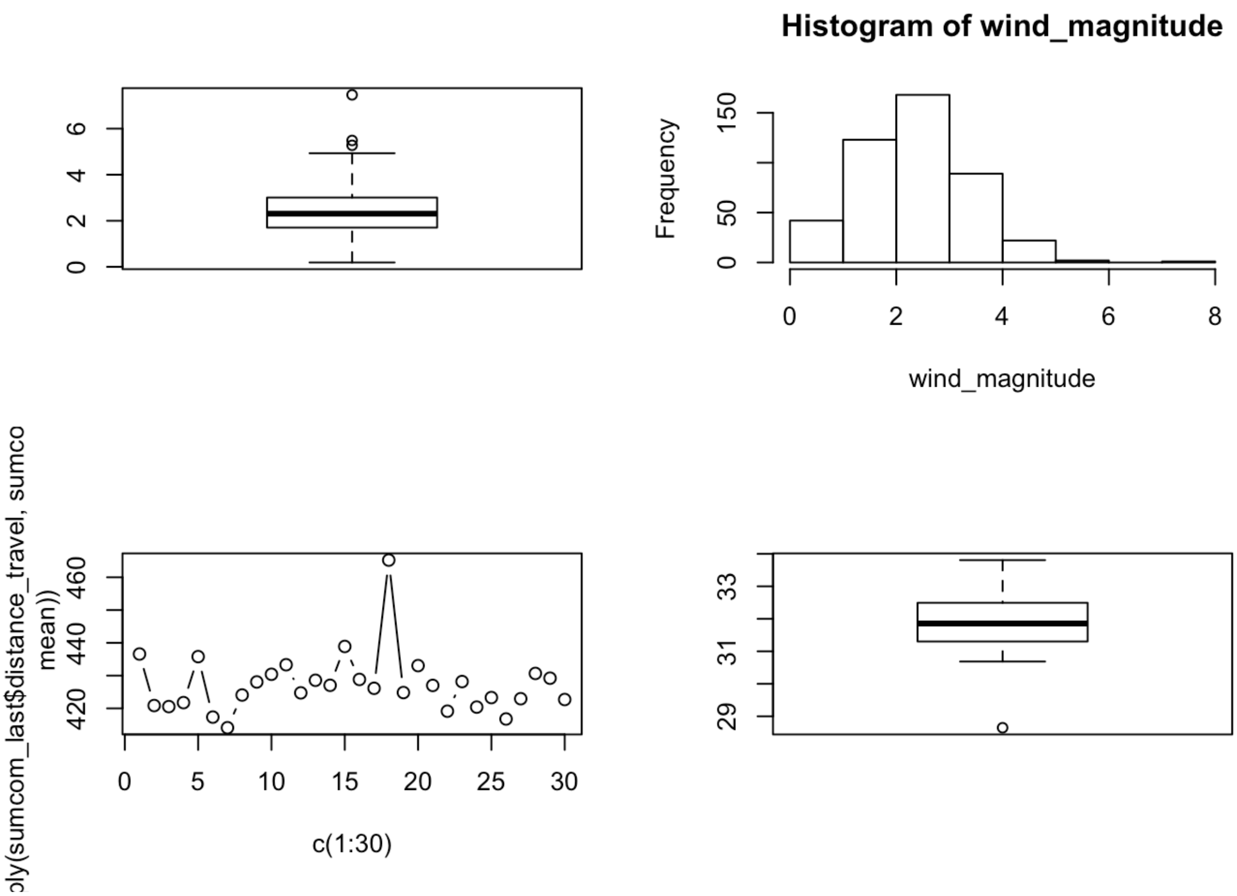
```
range(tapply(sumcom_last$distance_travel,sumcom_last$datetime,mean))  # 2018-09-12 has strong wind and quite one of the shortist distance travled.
```

```
## [1] 414.0976 465.2436
```

```r
tapply(sumcom_last$air_temperature,sumcom_last$datetime,mean)
```

```
## 2018-09-06 2018-09-07 2018-09-08 2018-09-09 2018-09-10 2018-09-11
##   23.65981   25.87500   26.26458   28.02917   22.76111   25.60782
## 2018-09-12 2018-09-13 2018-09-14 2018-09-15 2018-09-16 2018-09-17
##   25.37500   27.83167   23.58542   24.33750   25.68542   29.70834
## 2018-09-18 2018-09-19 2018-09-20 2018-09-21 2018-09-22 2018-09-23
##   26.60909   25.22727   19.87059   22.31458   29.58333   20.73750
## 2018-09-24 2018-09-25 2018-09-26 2018-09-27 2018-09-28 2018-09-29
##   23.27000   24.40250   27.88971   29.81000   24.37000   25.54774
## 2018-09-30 2018-10-01 2018-10-02 2018-10-03 2018-10-04 2018-10-05
##   25.89583   26.80875   26.93478   22.71552   25.10595   21.91000
```

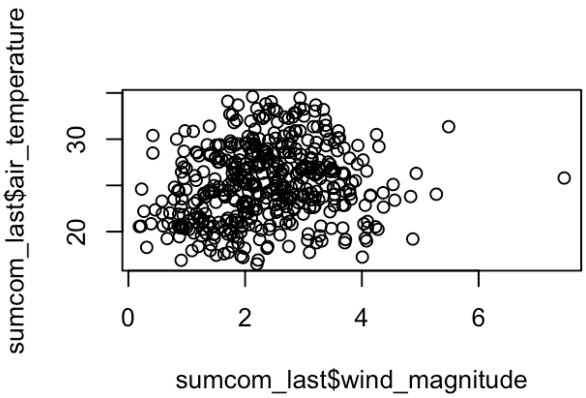
```r
tapply(sumcom_last$launch_airspeed,sumcom_last$datetime,mean) %>% boxplot()
```




```r
tapply(sumcom_last$launch_airspeed,sumcom_last$datetime,mean) %>% min() # 2018-09-23 has lowest launc
h speed averagely
```

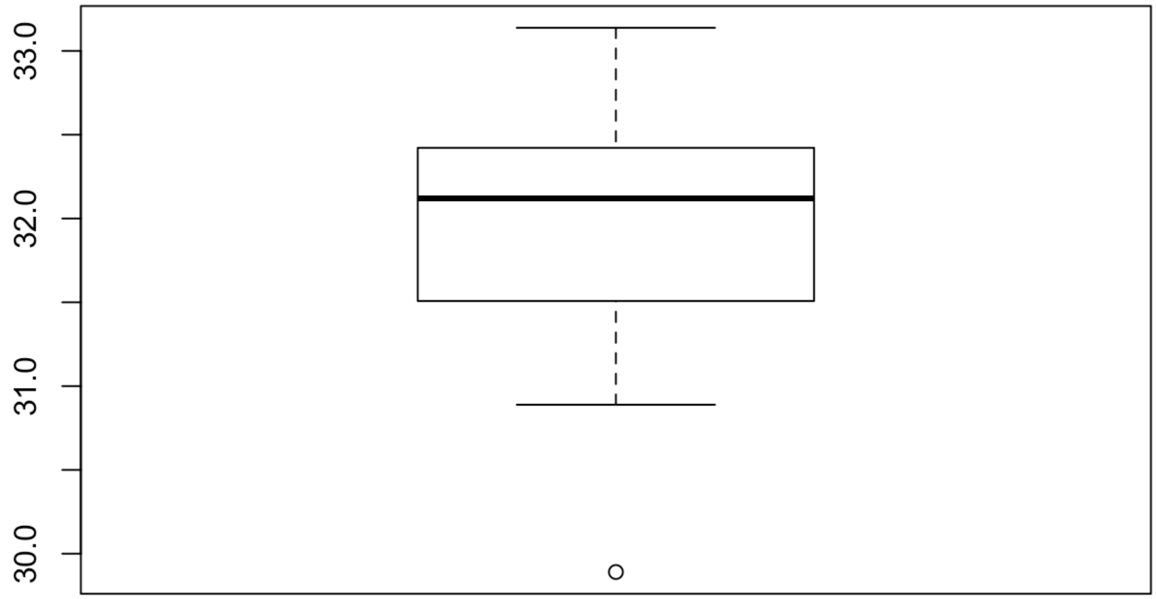
```
## [1] 28.64822
```

```r
plot(x=sumcom_last$wind_magnitude,y=sumcom_last$air_temperature)
```



`Answer` - I find that wind magnitude could possibly affect the distrance travel in 15 seconds, - The air_temperature is affecting wind magnitude. - The launch speed is also affected by air_temp and wind magnitude - I found some potential outliers in some of the variable and noted in commment also I find certain - 2018-09-12 has strongest wind and one of the shortest distance travelled averagely in that day.
- So it is clear that weather makes an important role in speed of drone and distance they could travel in certain time and it’s worth to get proved by setting up models and do more statistical analysis then.

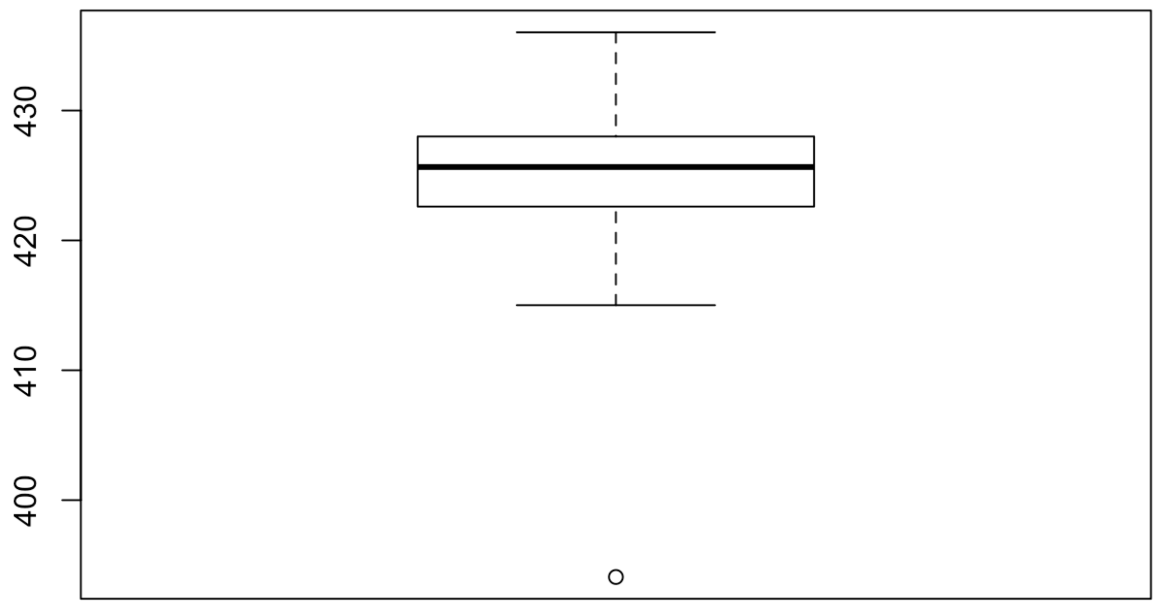
```
## CHECK the average air speed based on the components
# WING

tapply(summary_data$launch_airspeed,summary_data$wing_serial_number,mean) %>% boxplot()
```




```
min(tapply(summary_data$launch_airspeed,summary_data$wing_serial_number,mean) )
```

```
## [1] 29.89082
```

```
tapply(sumcom_last$distance_travel,summary_data$wing_serial_number,mean) %>% boxplot()
```



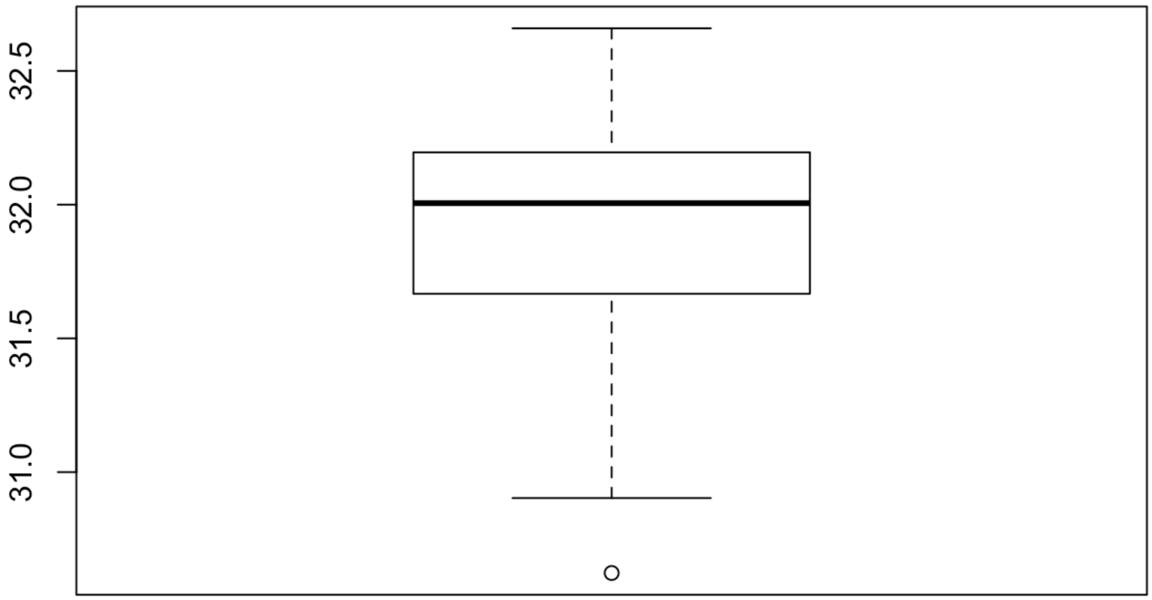

```
min(tapply(summary_data$launch_airspeed,summary_data$wing_serial_number,mean) )
```

```
## [1] 29.89082
```

```
summary_data[summary_data$wing_serial_number=="15SPJJJ09028064",]
```

```
##     flight_id air_temperature battery_serial_number body_serial_number
## 128     17192            21.2       15SPJJJ10050016       5.773501e+17
## 130     17195            24.0       15SPJJJ10056048       5.773501e+17
## 236     17399            30.9       15SPJJJ09018015       5.773501e+17
## 256     17441            21.0       15SPJJJ10040016       5.773488e+17
##         commit launch_airspeed launch_groundspeed        launch_timestamp
## 128 5c504d9a16        29.81584           30.21607 2018-09-14 07:54:02 CAT
## 130 5c504d9a16        30.48896           30.11618 2018-09-14 08:51:02 CAT
## 236 5c504d9a16        29.86724           30.16005 2018-09-22 17:28:37 CAT
## 256 4d9468bd3c        29.39124           30.32662 2018-09-25 07:09:11 CAT
##     preflight_voltage rel_humidity static_pressure wind_direction
## 128          31.70578     69.07494        80716.22     -102.76020
## 130          31.92941     66.90000        80737.08     -106.73234
## 236          32.01368     44.70000        80249.49      -66.56073
## 256          32.12819     63.65000        80645.02       53.98928
##     wind_magnitude wing_serial_number
## 128      1.6804769    15SPJJJ09028064
## 130      1.2947793    15SPJJJ09028064
## 236      0.9689003    15SPJJJ09028064
## 256      1.5526905    15SPJJJ09028064
```

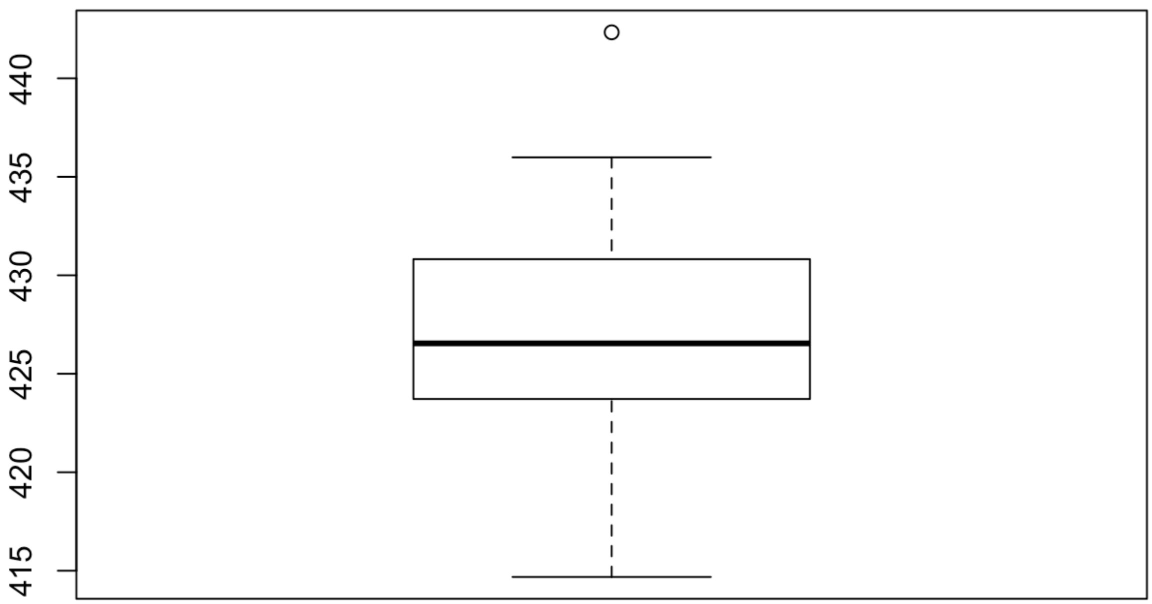
```
# 15SPJJJ09028064 min # they come from different day



# BATTERY
tapply(summary_data$launch_airspeed,summary_data$battery_serial_number,mean) %>% boxplot()
```




```
summary_data[summary_data$battery_serial_number=="15SPJJJ10052026",]
```

```
##     flight_id air_temperature battery_serial_number body_serial_number
## 9       16962        28.60000       15SPJJJ10052026       5.773501e+17
## 12      16980        18.20000       15SPJJJ10052026       5.773488e+17
## 17      16987        18.40000       15SPJJJ10052026       5.773501e+17
## 37      17029        28.45000       15SPJJJ10052026       5.772096e+17
## 74      17109        21.25000       15SPJJJ10052026       5.773501e+17
## 114     17173        27.97501       15SPJJJ10052026       5.773501e+17
## 168     17277        28.00000       15SPJJJ10052026       5.773501e+17
## 184     17303        23.50000       15SPJJJ10052026       5.773501e+17
## 243     17413        20.15000       15SPJJJ10052026       5.773488e+17
## 364     17630        26.30000       15SPJJJ10052026       5.773501e+17
## 387     17659        26.40000       15SPJJJ10052026       5.773501e+17
## 405     17681        20.45000       15SPJJJ10052026       5.773488e+17
##         commit launch_airspeed launch_groundspeed        launch_timestamp
## 9   5c504d9a16        31.63985           30.26374 2018-09-06 13:43:05 CAT
## 12  5c504d9a16        28.26758           31.02285 2018-09-06 17:46:38 CAT
## 17  5c504d9a16        28.83412           30.82901 2018-09-06 18:53:48 CAT
## 37  5c504d9a16        30.73261           30.04209 2018-09-08 13:36:27 CAT
## 74  5c504d9a16        28.46939           30.35008 2018-09-10 18:12:25 CAT
## 114 5c504d9a16        32.89062           30.10885 2018-09-13 12:02:52 CAT
## 168 5c504d9a16        29.53809           30.45948 2018-09-17 17:21:21 CAT
## 184 5c504d9a16        31.21466           29.97484 2018-09-19 07:38:16 CAT
## 243 5c504d9a16        28.49784           30.64442 2018-09-23 18:03:02 CAT
## 364 38bf99b15a        35.10431           29.87290 2018-10-02 08:39:42 CAT
## 387 1ecbc27833        32.22424           30.01476 2018-10-03 09:23:50 CAT
## 405 1ecbc27833        30.06057           30.06587 2018-10-03 17:20:05 CAT
##     preflight_voltage rel_humidity static_pressure wind_direction
## 9                  NA     60.37498        80445.02      -46.05301
## 12                 NA     67.80000        80473.49      173.52405
## 17           32.31293     68.25000        80493.81      144.95530
## 37           31.66149     47.80000        80487.15      -81.29874
## 74           32.26540     54.85000        80471.00     -159.07861
## 114          32.20058     62.97499        80677.81      -54.22348
## 168          32.32698     42.95000        80255.95     -137.49245
## 184          32.20166     61.67501        80642.25      -16.71810
## 243          32.32698     60.25000        80242.11     -169.32404
## 364          32.13792     50.45000        80554.17      -51.30995
## 387          32.21678     50.80000        80582.61      -47.20048
## 405          31.64636     67.70000        80498.11       34.94068
##     wind_magnitude wing_serial_number
## 9        1.1619245    15SPJJJ09011032
## 12       2.3755740    15SPJJJ11024054
## 17       1.1997710    15SPJJJ09011032
## 37       2.1239333    15SPJJJ11049056
## 74       1.8301288    15SPJJJ11049056
## 114      1.4933019    15SPJJJ09043062
## 168      3.1248559    15SPJJJ09008034
## 184      0.7840549    15SPJJJ09043062
## 243      2.2790694    15SPJJJ09052035
## 364      4.9308502    15SPJJJ09025064
## 387      1.8334581    15SPJJJ09008034
## 405      0.8833876    15SPJJJ09028034
```

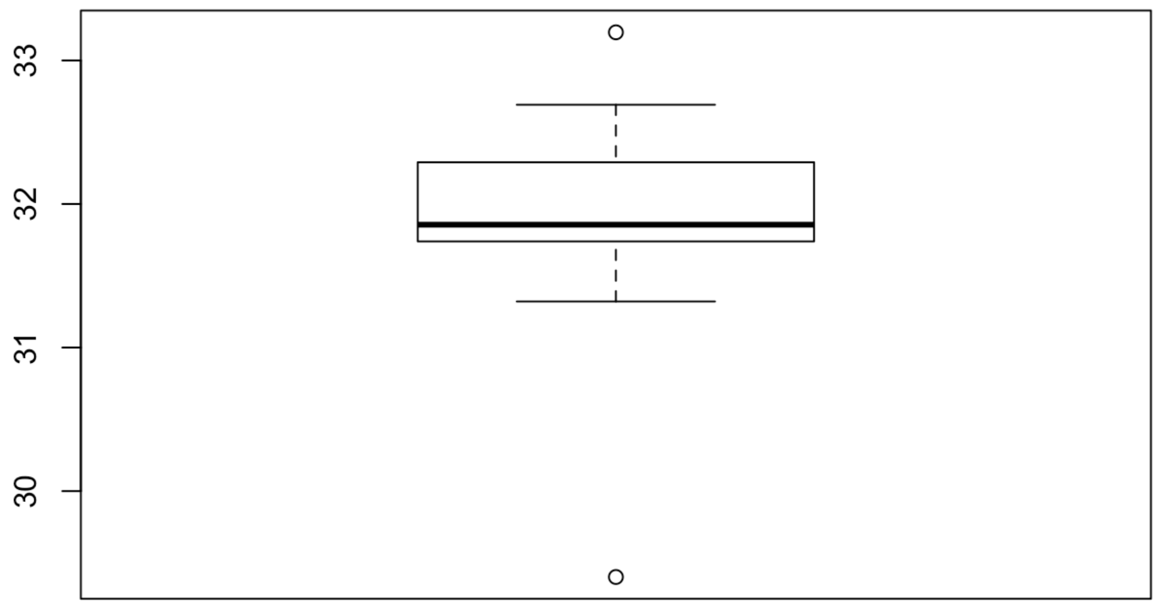
```
# 15SPJJJ10052026 min # they come from diffrent day
tapply(sumcom_last$distance_travel,summary_data$battery_serial_number,mean) %>% boxplot()
```




```
tapply(sumcom_last$distance_travel,summary_data$battery_serial_number,mean) %>% max()
```

```
## [1] 442.3365
```

```
# 15SPJJJ10052026  max



# BODY
tapply(summary_data$launch_airspeed,summary_data$body_serial_number,mean,na.rm = T) %>% boxplot()
```




```
tapply(sumcom_last$distance_travel,summary_data$body_serial_number,mean,na.rm = T)%>% max()
```

```
## [1] 443.9354
```

```
# 577348835962155008 max dis
tapply(summary_data$launch_airspeed,summary_data$body_serial_number,mean,na.rm = T) %>% range()
```

```
## [1] 29.40176 33.19601
```

```
# 577350132807356416 min
# 577209618523054080 max
summary_data[summary_data$body_serial_number=="577350132807356416",] # they come from same day
```

```
##    flight_id air_temperature battery_serial_number body_serial_number
## 60     17093           26.85       15SPJJJ09018015       5.773501e+17
## 70     17103           22.20       15SPJJJ10060032       5.773501e+17
## 74     17109           21.25       15SPJJJ10052026       5.773501e+17
##        commit launch_airspeed launch_groundspeed        launch_timestamp
## 60 5c504d9a16        30.18534           29.74227 2018-09-10 09:55:58 CAT
## 70 5c504d9a16        29.55054           30.45524 2018-09-10 17:02:05 CAT
## 74 5c504d9a16        28.46939           30.35008 2018-09-10 18:12:25 CAT
##    preflight_voltage rel_humidity static_pressure wind_direction
## 60          32.26972        60.65        80810.46      -55.33610
## 70          32.00719        57.05        80378.29      -85.38741
## 74          32.26540        54.85        80471.00     -159.07861
##    wind_magnitude wing_serial_number
## 60      0.9909951    15SPJJJ11049056
## 70      0.2673694    15SPJJJ09043062
## 74      1.8301288    15SPJJJ11049056
```

```
summary_data[summary_data$body_serial_number=="577209618523054080",] # same day as well.
```

```
##    flight_id air_temperature battery_serial_number body_serial_number
## 2      16952        20.50000       15SPJJJ10029029       5.772096e+17
## 3      16954        24.47502       15SPJJJ10012034       5.772096e+17
## 10     16965        32.25000       15SPJJJ10029029       5.772096e+17
##        commit launch_airspeed launch_groundspeed        launch_timestamp
## 2  5c504d9a16        32.14121           30.53525 2018-09-06 07:51:49 CAT
## 3  5c504d9a16        34.70188           29.87261 2018-09-06 09:56:37 CAT
## 10 5c504d9a16        32.74496           30.35478 2018-09-06 14:56:25 CAT
##    preflight_voltage rel_humidity static_pressure wind_direction
## 2                 NA     71.17504        80708.07      -4.408768
## 3                 NA     66.37498        80774.27     -23.458781
## 10                NA     49.60000        80379.65     -17.594640
##    wind_magnitude wing_serial_number
## 2       0.9173566    15SPJJJ09011032
## 3       3.7883831    15SPJJJ09011032
## 10      2.7420269    15SPJJJ11049056
```

`Answer` - I mark the potential outliers (components that work not well) for each by comparing the air speed and distance travelled. - **I mark they are from same day means the low/hight speed might not due to components but weather which we previously assume they affect the speed and distance intuitively for example the potential outliers by body_serie_number.** - **I mark different day, for example the wing 15SPJJJ09028064 higly possible that works bad because in different days (different environment) they always perform badly for speed.** - **For the battery 15SPJJJ10052026 it gives both lowest airspeed and highest distance, so I categorize it as unexplained behavior. The records are comming from different environment so i cannot conclude it's due to weather conditions.**

Data exploriation using Plotly, GGPLOT and Tableau

```
# Before really build my models to prove and quantify my previous findings.
# I see up several graphs using different packages in R, Panda and also from Tableau based on need.
p_track<- plot_ly(myfiles[[1]],x = ~myfiles[[1]]$position_ned_m.0., y = ~myfiles[[1]]$position_ned_m.
1., z = ~-1*(myfiles[[1]]$position_ned_m.2.)) %>%  add_markers()
p_track
```

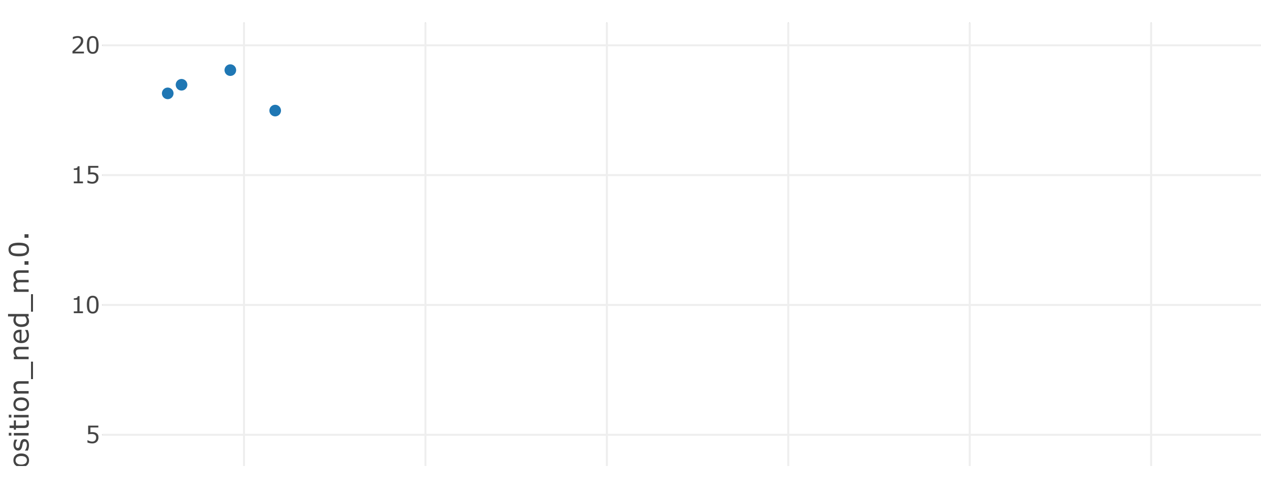
```
# api_create(p_track , filename = "position_track - one random flight")
# We can see the drone climb up into sky and fly.
```

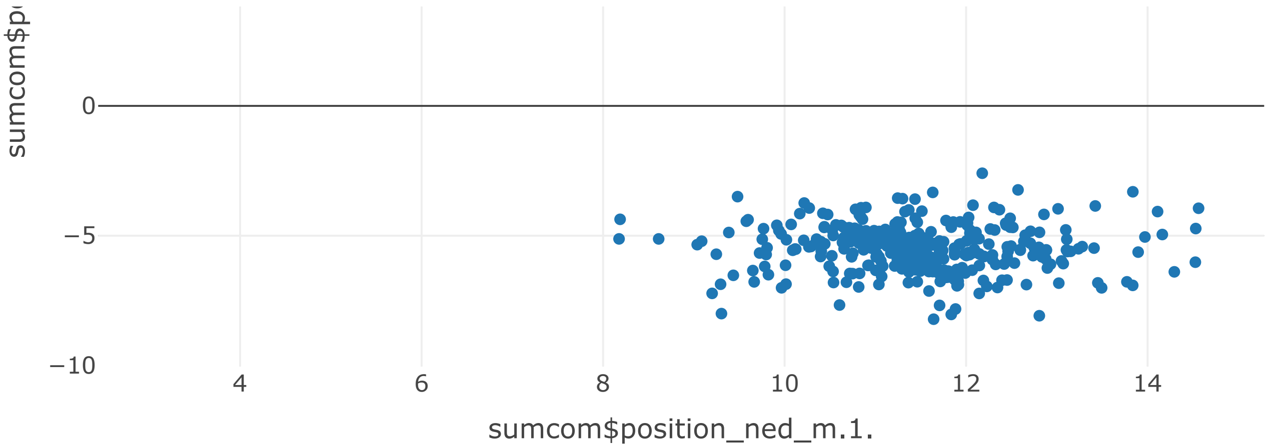
`Answer` - We can see the drone climb up into sky and fly. - Check the link to see more: https://plot.ly/~michaelmiaomiao/3/ (https://plot.ly/~michaelmiaomiao/3/)

```
## map for launching positions for all flights x=east, y= north
position_distribution <- plot_ly(data = sumcom,y=~sumcom$position_ned_m.0.,x=~sumcom$position_ned_m.1
.)
position_distribution
```

```
## No trace type specified:
##   Based on info supplied, a 'scatter' trace seems appropriate.
##   Read more about this trace type -> https://plot.ly/r/reference/#scatter
```

```
## No scatter mode specifed:
##   Setting the mode to markers
##   Read more about this attribute -> https://plot.ly/r/reference/#scatter-mode
```






```
# Based on similar plot in Tableau I realy lize four flight starts from quite different position --le
gt upper corners points  which are flight
# api_create(position_distribution , filename = "position_distribution - scatter plot")

# Also talbleau graph are attached here.
```

`Answer` : - This is link to https://plot.ly/~michaelmiaomiao/1/ (https://plot.ly/~michaelmiaomiao/1/) to see more detail about the positions at launch. - The four points have weired postions at upper left corner. - Flight Id Position Ned M.0. Position Ned M.1. - 17439 19.043728 3.8491414 - 17438 17.48529 4.343673 - 17437 18.148804 3.1589265 - 17136 18.479887 3.3110466 - Also talbleau graphs are included in the submission folder.

```
# Also I draw plots based on day for errors in Tableau in sheet 12 and find errors > 3(aroud 3rd Q),
as follow:
```

`Answer` - The following flights have large position errors averagely. - error flight_id - 3.6561384 17727 –3.105732 17726 - 5.3882611 17702 - 4.36668944 17699 - 5.8239182 17635 - 4.2272231 17593 - 3.84838555 17586 - 3.4442441 17460 - 3.2004311 17399 - 4.6499484 17326 - 3.53671036 17311 - 3.8611856 17309 - 3.62672143 17181 - 3.3146722 17125

More data exploration based on date

```
####distance
travel_average_byday <- tapply(sumcom_last$distance_travel,sumcom_last$datetime,mean)
par(mfrow=c(2,2))
barplot(travel_average_byday,ylim = c(400,460),col = "light green")
travel_average_byday %>% print() #2018-09-23
```

```
## 2018-09-06 2018-09-07 2018-09-08 2018-09-09 2018-09-10 2018-09-11
##   436.5852   420.8661   420.5463   421.7589   435.7925   417.3416
## 2018-09-12 2018-09-13 2018-09-14 2018-09-15 2018-09-16 2018-09-17
##   414.0976   424.0795   428.0676   430.4087   433.3155   424.7380
## 2018-09-18 2018-09-19 2018-09-20 2018-09-21 2018-09-22 2018-09-23
##   428.5770   427.0411   438.8954   428.8350   426.1259   465.2436
## 2018-09-24 2018-09-25 2018-09-26 2018-09-27 2018-09-28 2018-09-29
##   424.8297   433.0503   427.0050   419.1462   428.1857   420.3933
## 2018-09-30 2018-10-01 2018-10-02 2018-10-03 2018-10-04 2018-10-05
##   423.2996   416.7704   422.9500   430.6660   429.2154   422.7038
```

```
summary(travel_average_byday)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   414.1   422.0   426.6   427.4   430.1   465.2
```

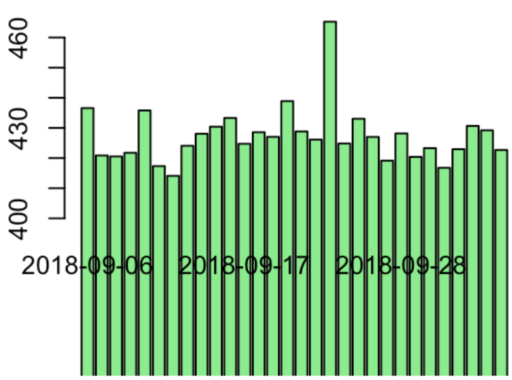
```
####temp
temp_byday <- tapply(sumcom_last$air_temperature,sumcom_last$datetime,mean) %>% print()
```

```
## 2018-09-06 2018-09-07 2018-09-08 2018-09-09 2018-09-10 2018-09-11
##   23.65981   25.87500   26.26458   28.02917   22.76111   25.60782
## 2018-09-12 2018-09-13 2018-09-14 2018-09-15 2018-09-16 2018-09-17
##   25.37500   27.83167   23.58542   24.33750   25.68542   29.70834
## 2018-09-18 2018-09-19 2018-09-20 2018-09-21 2018-09-22 2018-09-23
##   26.60909   25.22727   19.87059   22.31458   29.58333   20.73750
## 2018-09-24 2018-09-25 2018-09-26 2018-09-27 2018-09-28 2018-09-29
##   23.27000   24.40250   27.88971   29.81000   24.37000   25.54774
## 2018-09-30 2018-10-01 2018-10-02 2018-10-03 2018-10-04 2018-10-05
##   25.89583   26.80875   26.93478   22.71552   25.10595   21.91000
```

```
summary(temp_byday)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   19.87   23.60   25.46   25.26   26.76   29.81
```

```
par(mfrow=c(2,2))
```



```r
barplot(tapply(sumcom_last$air_temperature,sumcom_last$datetime,mean),ylim = c(20,30),col = "light bl
ue")  # 2018-09-23

####wind
wind_byday <- tapply(sumcom_last$wind_magnitude,sumcom_last$datetime,mean) %>% print()
```

```
## 2018-09-06 2018-09-07 2018-09-08 2018-09-09 2018-09-10 2018-09-11
##   2.127035   2.432881   2.434992   2.194056   1.960421   2.141916
## 2018-09-12 2018-09-13 2018-09-14 2018-09-15 2018-09-16 2018-09-17
##   3.130213   1.958277   1.178155   2.050966   2.601785   2.356680
## 2018-09-18 2018-09-19 2018-09-20 2018-09-21 2018-09-22 2018-09-23
##   1.823037   2.443068   1.971643   1.460815   2.229876   1.981061
## 2018-09-24 2018-09-25 2018-09-26 2018-09-27 2018-09-28 2018-09-29
##   2.647857   2.144133   2.758542   2.844207   2.331601   3.079609
## 2018-09-30 2018-10-01 2018-10-02 2018-10-03 2018-10-04 2018-10-05
##   2.693475   3.033240   3.157236   2.059061   2.234983   2.245979
```

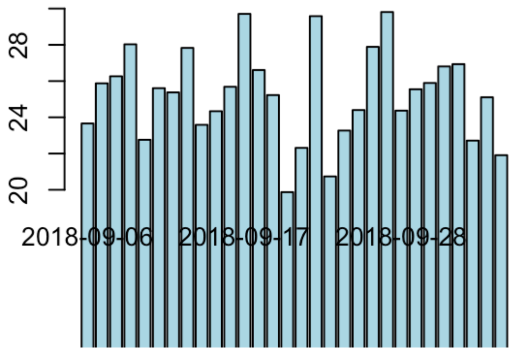
```r
wind_byday
```


```
## 2018-09-06 2018-09-07 2018-09-08 2018-09-09 2018-09-10 2018-09-11
##   2.127035   2.432881   2.434992   2.194056   1.960421   2.141916
## 2018-09-12 2018-09-13 2018-09-14 2018-09-15 2018-09-16 2018-09-17
##   3.130213   1.958277   1.178155   2.050966   2.601785   2.356680
## 2018-09-18 2018-09-19 2018-09-20 2018-09-21 2018-09-22 2018-09-23
##   1.823037   2.443068   1.971643   1.460815   2.229876   1.981061
## 2018-09-24 2018-09-25 2018-09-26 2018-09-27 2018-09-28 2018-09-29
##   2.647857   2.144133   2.758542   2.844207   2.331601   3.079609
## 2018-09-30 2018-10-01 2018-10-02 2018-10-03 2018-10-04 2018-10-05
##   2.693475   3.033240   3.157236   2.059061   2.234983   2.245979
```


```r
wind_byday %>% summary()
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.178   2.053   2.240   2.324   2.636   3.157
```

```r
par(mfrow=c(2,2))
```



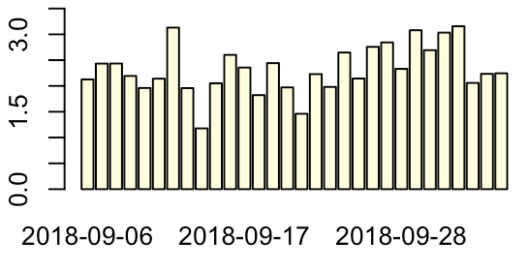

```
barplot(wind_byday,ylim = c(0,3.5),col = "light yellow")
####  2018-09-23




####airspeed
lauchsp_byday <- tapply(sumcom_last$launch_airspeed,sumcom_last$datetime,mean) %>% print()  #### 2018
-09-23
```

```
## 2018-09-06 2018-09-07 2018-09-08 2018-09-09 2018-09-10 2018-09-11
##   31.71309   31.76144   32.33448   31.83676   30.68589   31.73630
## 2018-09-12 2018-09-13 2018-09-14 2018-09-15 2018-09-16 2018-09-17
##   32.57819   32.19213   31.27357   31.06345   31.73752   32.28619
## 2018-09-18 2018-09-19 2018-09-20 2018-09-21 2018-09-22 2018-09-23
##   31.54510   32.38319   30.92289   30.71962   31.87632   28.64822
## 2018-09-24 2018-09-25 2018-09-26 2018-09-27 2018-09-28 2018-09-29
##   32.79854   31.07110   31.90055   32.91724   32.49327   33.20705
## 2018-09-30 2018-10-01 2018-10-02 2018-10-03 2018-10-04 2018-10-05
##   32.51199   33.80750   33.12042   31.77000   31.30251   31.97298
```

```
summary(lauchsp_byday)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   28.65   31.36   31.86   31.87   32.47   33.81
```

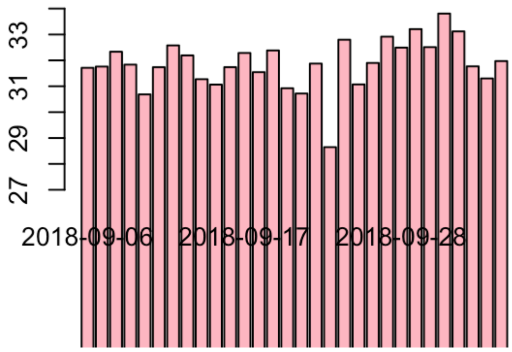
```
par(mfrow=c(2,2))
```




```
barplot(lauchsp_byday,ylim = c(27,34),col="light pink")
# sumcom_last[sumcom_last$datetime=="2018-09-23",] min

####


lauchsp_error <- tapply((sumcom_last$error+sumcom$error),sumcom_last$datetime,mean) %>% print()
```

```
## 2018-09-06 2018-09-07 2018-09-08 2018-09-09 2018-09-10 2018-09-11
##   1.928264   2.888232   3.342021   2.765032   2.235653   2.597486
## 2018-09-12 2018-09-13 2018-09-14 2018-09-15 2018-09-16 2018-09-17
##   2.417590   2.854329   2.686957   3.485458   3.294210   2.229066
## 2018-09-18 2018-09-19 2018-09-20 2018-09-21 2018-09-22 2018-09-23
##   2.409463   3.214554   2.711266   3.524681   2.978122   3.150650
## 2018-09-24 2018-09-25 2018-09-26 2018-09-27 2018-09-28 2018-09-29
##   2.712581   3.507661   3.026100   2.413297   2.264341   3.029075
## 2018-09-30 2018-10-01 2018-10-02 2018-10-03 2018-10-04 2018-10-05
##   5.131051   2.883129   2.714246   2.901203   4.100110   3.613741
```

```
par(mfrow=c(2,2))
```




```
lauchsp_error %>% boxplot()
range(lauchsp_error)
```

```
## [1] 1.928264 5.131051
```

```
# 2018-09-17 min error
# 2018-09-30 max error

# 2018-09-23 has relatively average error

sumcom_last[sumcom_last$datetime=="2018-09-30",]
```

```
##     flight_id air_temperature battery_serial_number body_serial_number
## 334     17589          20.550       15SPJJJ10056048       5.773501e+17
## 335     17590          22.850       15SPJJJ10019016       5.773501e+17
## 336     17591          28.050       15SPJJJ10056048       5.773501e+17
## 337     17592          28.550       15SPJJJ10050016       5.773488e+17
## 338     17593          30.125       15SPJJJ10022048       5.773501e+17
## 339     17594          25.250       15SPJJJ10056048       5.773501e+17
##         commit launch_airspeed launch_groundspeed        launch_timestamp
## 334 38bf99b15a        31.83897           29.89783 2018-09-30 07:58:11 CAT
## 335 38bf99b15a        35.82533           29.74221 2018-09-30 09:35:47 CAT
## 336 38bf99b15a        32.98829           29.83806 2018-09-30 12:52:56 CAT
## 337 38bf99b15a        31.64530           30.09586 2018-09-30 12:56:28 CAT
## 338 38bf99b15a        31.74863           29.78376 2018-09-30 13:29:54 CAT
## 339 38bf99b15a        31.02546           30.28819 2018-09-30 17:06:59 CAT
##     preflight_voltage rel_humidity static_pressure wind_direction
## 334          31.80517        65.75        80758.43      -49.91983
## 335          32.05797        64.50        80809.27      -32.98361
```

```
## 336          32.27728        58.75        80574.44      -60.10718
## 337          32.20922        59.50        80534.49      -52.41973
## 338          32.11199        49.30        80274.28      -66.45707
## 339          31.91213        56.70        80343.66      -86.93097
##     wind_magnitude wing_serial_number seconds_since_launch
## 334       1.364852    15SPJJJ09040032             14.99542
## 335       3.707514    15SPJJJ09040032             14.99537
## 336       2.774602    15SPJJJ09031032             14.99545
## 337       2.361319    15SPJJJ09040032             14.99544
## 338       3.617039    15SPJJJ09008034             14.99541
## 339       2.335525    15SPJJJ09040032             14.99538
##     position_ned_m.0. position_ned_m.1. position_ned_m.2.
## 334         -402.0740          185.1781         -77.73494
## 335         -386.9127          176.6132         -76.06919
## 336         -386.8881          176.7912         -76.82001
## 337         -394.9049          180.9773         -77.84484
## 338         -387.2269          177.1193         -78.00999
## 339         -400.6611          185.8711         -77.09568
##     velocity_ned_mps.0. velocity_ned_mps.1. velocity_ned_mps.2.
## 334           -25.56490            15.06070          -0.6343892
## 335           -24.38108            13.46235          -2.3123183
## 336           -24.87971            12.79982          -1.9018772
## 337           -25.04969            13.92336          -1.3045157
## 338           -24.39591            12.85232          -2.7935214
## 339           -26.16767            15.25441          -0.7306841
##     accel_body_mps2.0. accel_body_mps2.1. accel_body_mps2.2.
## 334           1.361574         0.10797766         -7.2831430
## 335           1.046137         0.21138728         -4.9615216
## 336           1.509822        -0.06897218          0.5650798
## 337           1.409641         0.24184518         -6.6773510
## 338           1.092700         0.11886739         -3.5589173
## 339           1.442826         0.41567930         -7.5023675
##     orientation_rad.0. orientation_rad.1. orientation_rad.2.
## 334         -0.4153132        0.010195659           2.579648
## 335         -0.2657863        0.073913700           2.624076
## 336         -0.2638269        0.045947980           2.575331
## 337         -0.3393584        0.018981304           2.522569
## 338         -0.3186988       -0.008205542           2.497834
## 339         -0.4195814        0.021884360           2.510503
##     angular_rate_body_radps.0. angular_rate_body_radps.1.
## 334                -0.14698726               -0.044434140
## 335                -0.08734535               -0.095107734
## 336                 0.12373587               -0.152833210
## 337                -0.01786421               -0.020641468
## 338                -0.26049173               -0.161836640
## 339                -0.07766923                0.003879925
##     angular_rate_body_radps.2. position_sigma_ned_m.0.
## 334                -0.11925533              0.62462090
## 335                -0.09523319              0.39031634
## 336                -0.05745659              0.26675197
## 337                -0.10268874              0.72938424
## 338                -0.17397588              2.03900430
## 339                -0.10243278              0.01618769
##     position_sigma_ned_m.1. position_sigma_ned_m.2. calculated_speed
## 334              1.01354200              0.76509994         29.67813
## 335              0.98768900              0.89499550         27.94671
## 336              0.39624876              0.52346600         28.04375
## 337              0.53472316              0.41664058         28.68882
## 338              1.17660260              1.01161620         27.71545
```

```
## 339               0.01809392               0.04577897         30.29815
##     distance_travel      error   datetime
## 334        433.4741 2.40326284 2018-09-30
## 335        417.0441 2.27300084 2018-09-30
## 336        415.6861 1.18646673 2018-09-30
## 337        425.9945 1.68074798 2018-09-30
## 338        416.3629 4.22722310 2018-09-30
## 339        431.2360 0.08006057 2018-09-30
```

```
table(sumcom_last[sumcom_last$datetime=="2018-09-30",("body_serial_number")])
```

```
## 
## 577348835962032128 577350132790489088 577350132807389184 
##                  1                  3                  2
```

```
table(sumcom_last[sumcom_last$datetime=="2018-09-30",("battery_serial_number")])
```

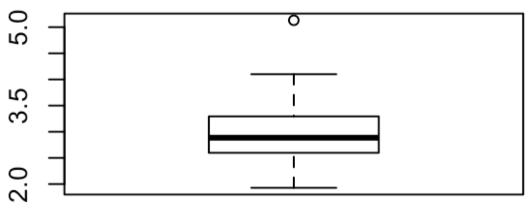
```
## 
## 15SPJJJ09010022 15SPJJJ09013015 15SPJJJ09017016 15SPJJJ09018015 
##               0               0               0               0 
## 15SPJJJ09036021 15SPJJJ10005031 15SPJJJ10007045 15SPJJJ10008029 
##               0               0               0               0 
## 15SPJJJ10012034 15SPJJJ10018016 15SPJJJ10019016 15SPJJJ10021047 
##               0               0               1               0 
## 15SPJJJ10022048 15SPJJJ10023027 15SPJJJ10027028 15SPJJJ10029029 
##               1               0               0               0 
## 15SPJJJ10030028 15SPJJJ10040016 15SPJJJ10048030 15SPJJJ10050016 
##               0               0               0               1 
## 15SPJJJ10050049 15SPJJJ10052026 15SPJJJ10054027 15SPJJJ10056048 
##               0               0               0               3 
## 15SPJJJ10060032 15SPJJJ11059037 
##               0               0
```

```
table(sumcom_last[sumcom_last$datetime=="2018-09-30",("wing_serial_number")])
```

```
## 
## 15SPJJJ09008034 15SPJJJ09010032 15SPJJJ09011032 15SPJJJ09019061 
##               1               0               0               0 
## 15SPJJJ09021032 15SPJJJ09024061 15SPJJJ09025064 15SPJJJ09028034 
##               0               0               0               0 
## 15SPJJJ09028064 15SPJJJ09031032 15SPJJJ09032034 15SPJJJ09036063 
##               0               1               0               0 
## 15SPJJJ09040032 15SPJJJ09043062 15SPJJJ09052035 15SPJJJ11024054 
##               4               0               0               0 
## 15SPJJJ11048054 15SPJJJ11049056 
##               0               0
```

```
# - I find  2018-09-30 has large position errors so it might due to batterys that day only uses batte
ry 15SPJJJ10056048 with highes frequency, body 577350132790489088, and wing 15SPJJJ09040032

tapply(sumcom_last$wind_magnitude,sumcom_last$datetime,mean)[["2018-09-23"]]
```

```
## [1] 1.981061
```

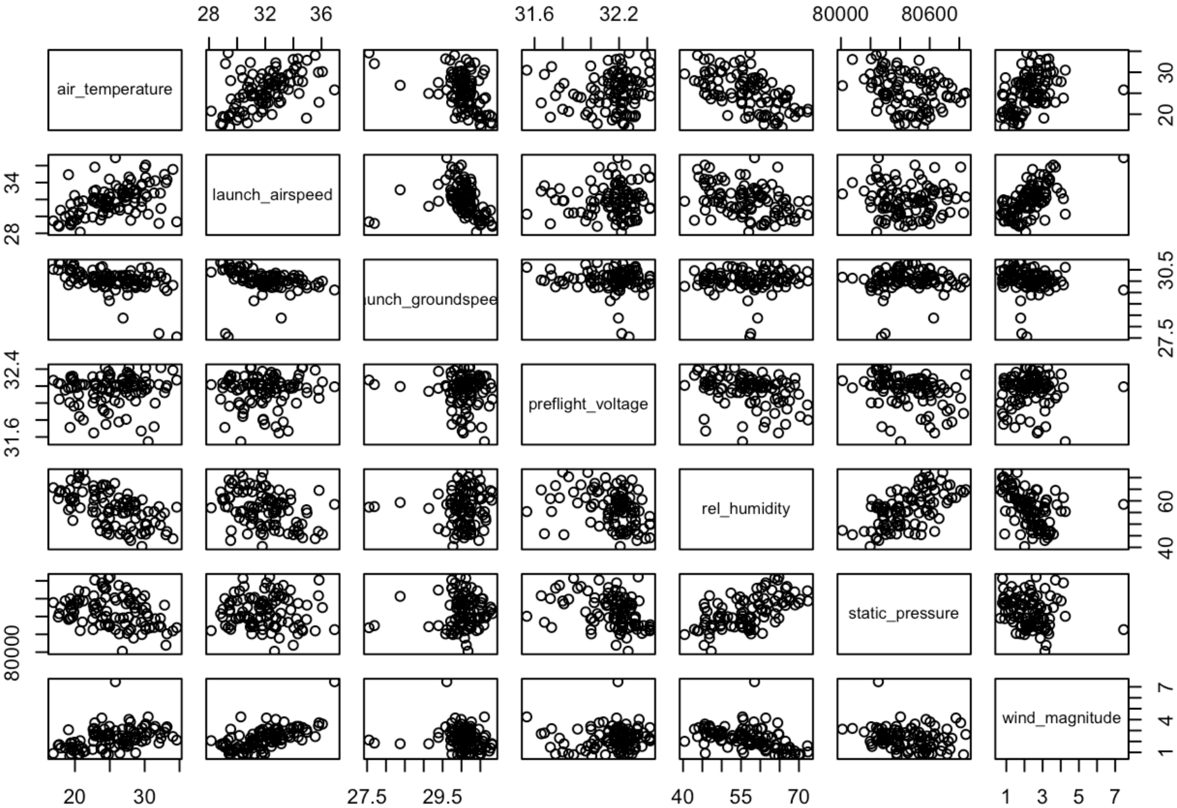


`Answer` - I find 2018-09-30 has large position errors so it might due to batterys that day only uses battery 15SPJJJ10056048 with highes frequency, body 577350132790489088, and wing 15SPJJJ09040032

- The previous unexplained 2018-09-23 does NOT hace high error, so it might due to wind and other weather conditions wind relatively low:1.981061

Modeling and validation

```
# First glimpse of any possible linear regresion (by sampling out some from whole data for clear grap
h)
pairs(summary_data[sample(1:nrow(summary_data),100,replace = F),c(2,6,7,9,10,11,13)])
```



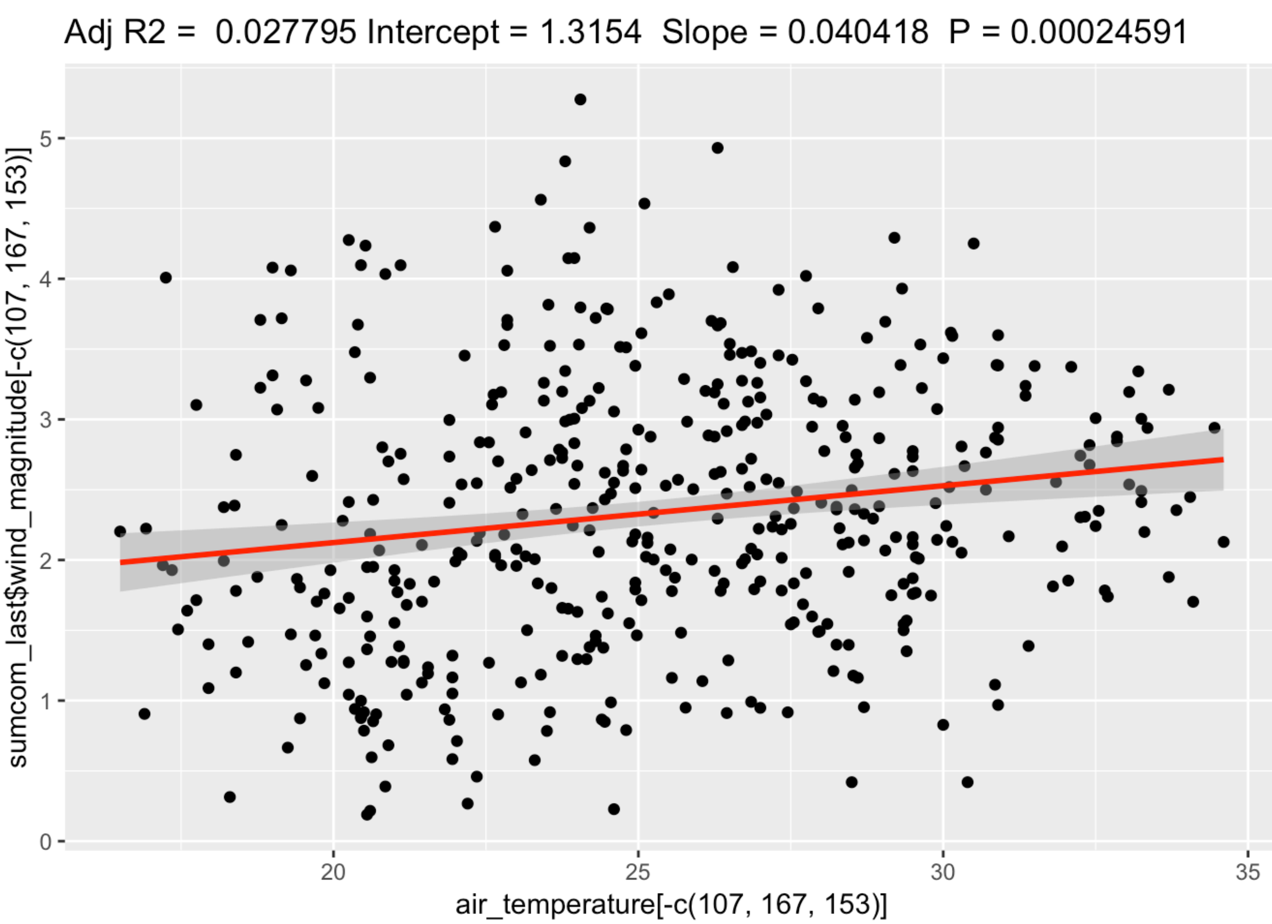

```
### Then I start budiling models based on my previous findings through outlier exploration, component
s check and plots
attach(sumcom_last, warn.conflicts = F)
###   regression association between air temp and wind magnitude.
sumcom_last[sumcom_last$wind_magnitude>5,] # NO. 107 :: 17162   went through largest wind
```

```
##     flight_id air_temperature battery_serial_number body_serial_number
## 96      17145           24.05       15SPJJJ09036021       5.773501e+17
## 107     17162           25.80       15SPJJJ09018015       5.773501e+17
## 167     17274           31.35       15SPJJJ10048030       5.773501e+17
##         commit launch_airspeed launch_groundspeed        launch_timestamp
## 96  5c504d9a16        34.58137           29.84380 2018-09-12 08:55:56 CAT
## 107 5c504d9a16        36.92920           29.61042 2018-09-12 16:58:38 CAT
## 167 5c504d9a16        34.53072           29.98681 2018-09-17 16:23:47 CAT
##     preflight_voltage rel_humidity static_pressure wind_direction
## 96           32.12171        64.35        80563.51      -12.85778
## 107          32.19193        58.50        80252.84      -56.52143
## 167          32.16600        52.70        80111.57      -87.36706
##     wind_magnitude wing_serial_number seconds_since_launch
## 96        5.275389    15SPJJJ09052035             14.99549
## 107       7.466193    15SPJJJ09052035             14.99544
## 167       5.486348    15SPJJJ09052035             14.99549
##     position_ned_m.0. position_ned_m.1. position_ned_m.2.
## 96          -360.5559          163.4982         -72.37028
## 107         -326.6844          146.3029         -65.44482
## 167         -387.0159          176.6228         -76.55714
##     velocity_ned_mps.0. velocity_ned_mps.1. velocity_ned_mps.2.
## 96            -21.28304            10.11100           -4.504086
## 107           -19.06580             8.62150           -2.815421
## 167           -23.15765            12.74663           -2.274825
##     accel_body_mps2.0. accel_body_mps2.1. accel_body_mps2.2.
## 96           0.8286435         0.02298691          -8.457809
## 107          1.8803368         0.03061696         -11.432279
## 167          1.0948108         0.13712817          -4.895300
##     orientation_rad.0. orientation_rad.1. orientation_rad.2.
## 96         -0.12401899         0.17141998           2.762434
## 107        -0.07291253         0.14846751           2.482747
## 167        -0.27646154         0.05733163           2.366463
##     angular_rate_body_radps.0. angular_rate_body_radps.1.
## 96                  0.01284266               -0.002646205
## 107                 0.10115400                0.091850370
## 167                -0.02348123               -0.079776675
##     angular_rate_body_radps.2. position_sigma_ned_m.0.
## 96                 -0.07080932              0.39671758
## 107                -0.03830098              0.02391988
## 167                -0.11645035              0.00523444
##     position_sigma_ned_m.1. position_sigma_ned_m.2. calculated_speed
## 96              0.177394520              0.39050934         23.98931
## 107             0.028533220              0.05714971         21.11307
## 167             0.006849676              0.01092471         26.53165
##     distance_travel      error   datetime
## 96         385.9088 0.96462144 2018-09-12
## 107        348.5531 0.10960281 2018-09-12
## 167        415.9838 0.02300883 2018-09-17
```

```
model_windmag_temp<- lm(sumcom_last$wind_magnitude[-c(107,167,153)]~sumcom_last$air_temperature[-c(10
7,167,153)])
summary(model_windmag_temp)
```

```
##
## Call:
## lm(formula = sumcom_last$wind_magnitude[-c(107, 167, 153)] ~
##     sumcom_last$air_temperature[-c(107, 167, 153)])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.12460 -0.64893 -0.08462  0.62424  2.98795
##
## Coefficients:
##                                                Estimate Std. Error t value
## (Intercept)                                     1.31540    0.27945   4.707
## sumcom_last$air_temperature[-c(107, 167, 153)]  0.04042    0.01093   3.697
##                                                Pr(>|t|)
## (Intercept)                                    3.37e-06 ***
## sumcom_last$air_temperature[-c(107, 167, 153)] 0.000246 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9373 on 442 degrees of freedom
## Multiple R-squared:  0.02999,    Adjusted R-squared:  0.0278
## F-statistic: 13.67 on 1 and 442 DF,  p-value: 0.0002459
```

```
ggplotRegression(lm(sumcom_last$wind_magnitude[-c(107,167,153)]~air_temperature[-c(107,167,153)], dat
a = summary_data))
```



```r
### no regression between windirection and groundspeed
model_grdspeed_win2 <- lm(wind_magnitude[which(wind_direction>0)]~launch_groundspeed[which(wind_direc
tion>0)])

summary(model_grdspeed_win2)
```

```
##
## Call:
## lm(formula = wind_magnitude[which(wind_direction > 0)] ~ launch_groundspeed[which(wind_direction >
##     0)])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.28816 -0.54671  0.03424  0.46188  1.46052
##
## Coefficients:
##                                              Estimate Std. Error t value
## (Intercept)                                   11.5407     9.8720   1.169
## launch_groundspeed[which(wind_direction > 0)]  -0.3211     0.3246  -0.989
##                                              Pr(>|t|)
## (Intercept)                                     0.249
## launch_groundspeed[which(wind_direction > 0)]    0.328
##
## Residual standard error: 0.7267 on 43 degrees of freedom
## Multiple R-squared:  0.02226,    Adjusted R-squared:  -0.0004785
## F-statistic: 0.979 on 1 and 43 DF,  p-value: 0.328
```

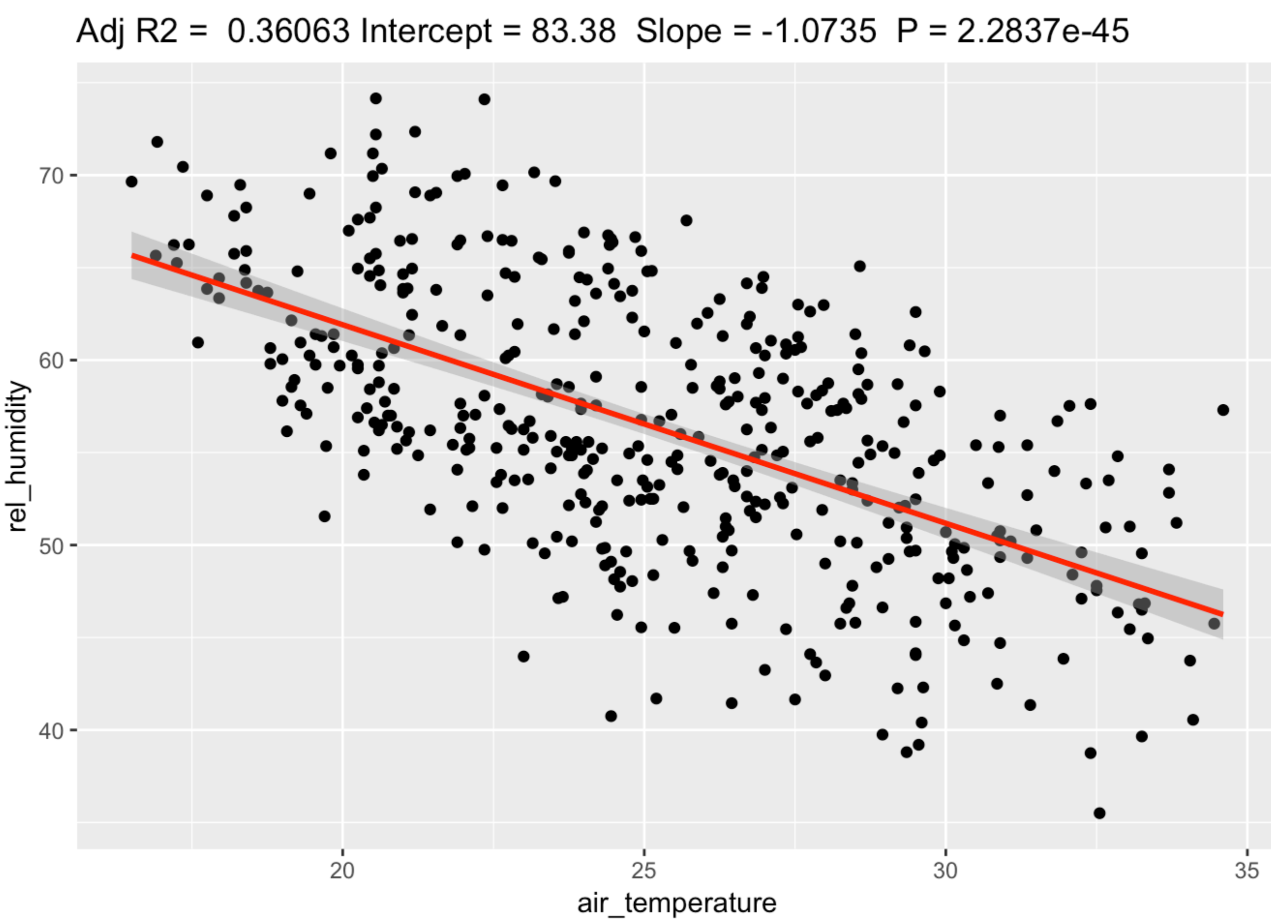
```r
model_grdspeed_win1 <- lm(wind_magnitude[which(wind_direction<0)]~launch_groundspeed[which(wind_direc
tion<0)])
summary(model_grdspeed_win1)
```

```
##
## Call:
## lm(formula = wind_magnitude[which(wind_direction < 0)] ~ launch_groundspeed[which(wind_direction <
##     0)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5460 -0.7023 -0.0394  0.6972  4.8866
##
## Coefficients:
##                                              Estimate Std. Error t value
## (Intercept)                                   12.3594     4.1587   2.972
## launch_groundspeed[which(wind_direction < 0)]  -0.3303     0.1383  -2.389
##                                              Pr(>|t|)
## (Intercept)                                   0.00314 **
## launch_groundspeed[which(wind_direction < 0)]  0.01736 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9959 on 400 degrees of freedom
## Multiple R-squared:  0.01407,    Adjusted R-squared:  0.0116
## F-statistic: 5.707 on 1 and 400 DF,  p-value: 0.01736
```

```r
# ggplotRegression(lm(launch_groundspeed~wind_direction, data = summary_data))


### regreesion between humidity and air temperature
model_hum_temp <- lm(rel_humidity~air_temperature)
summary(model_hum_temp)
```

```
##
## Call:
## lm(formula = rel_humidity ~ air_temperature)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.3840  -3.9299  -0.5219   4.6187  14.7118
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     83.38033    1.72659   48.29   <2e-16 ***
## air_temperature -1.07347    0.06755  -15.89   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.82 on 445 degrees of freedom
## Multiple R-squared:  0.3621, Adjusted R-squared:  0.3606
## F-statistic: 252.6 on 1 and 445 DF,  p-value: < 2.2e-16
```

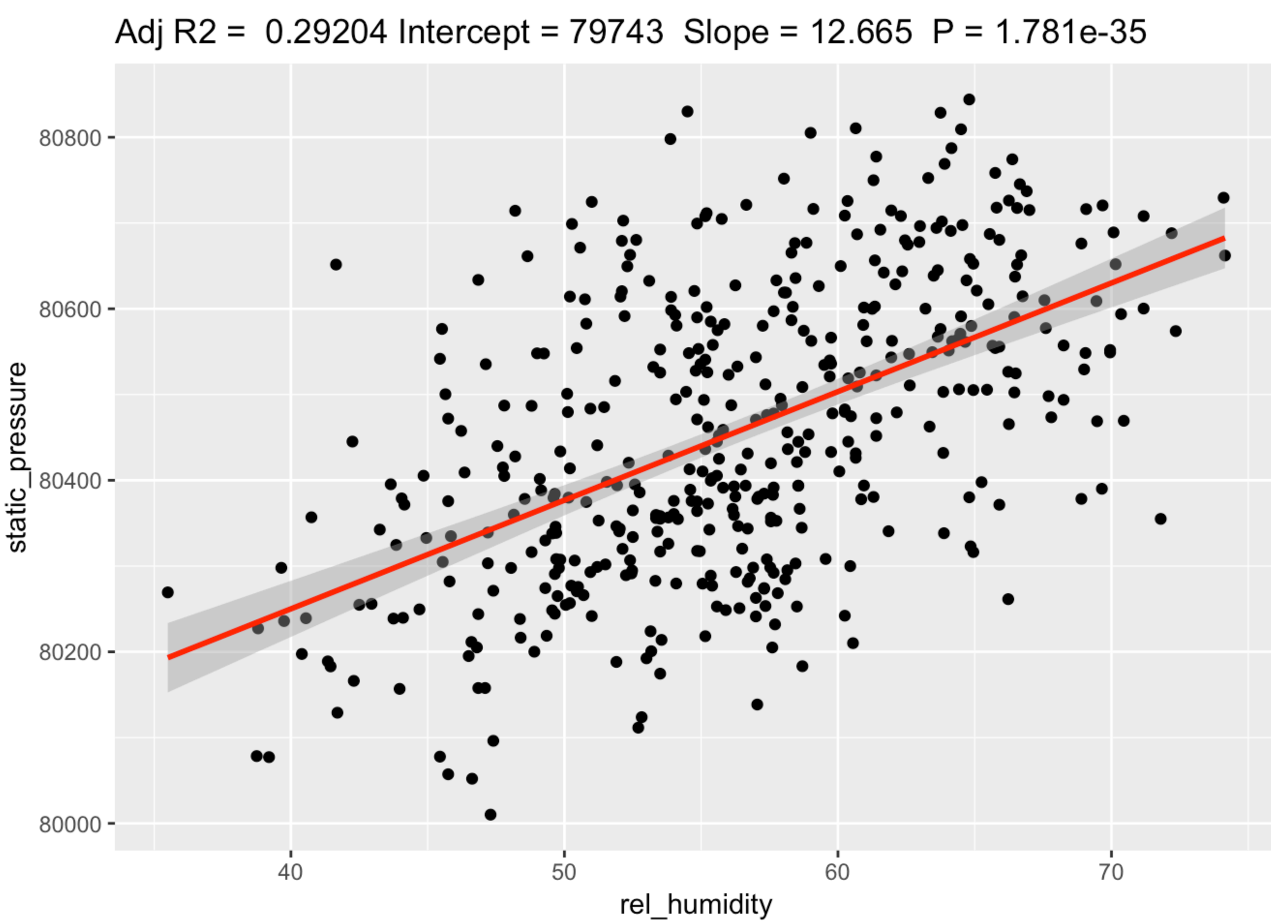
```r
ggplotRegression(lm(rel_humidity~air_temperature, data = summary_data))
```



```r
### regreesion between humidity and static pressure
model_statpre_hum <- lm(static_pressure~rel_humidity)
summary(model_statpre_hum)
```

```
##
## Call:
## lm(formula = static_pressure ~ rel_humidity)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -332.43  -93.78  -14.13   98.61  396.30
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.974e+04  5.285e+01  1508.7   <2e-16 ***
## rel_humidity 1.267e+01  9.312e-01    13.6   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 143.1 on 445 degrees of freedom
## Multiple R-squared:  0.2936, Adjusted R-squared:  0.292
## F-statistic:   185 on 1 and 445 DF,  p-value: < 2.2e-16
```

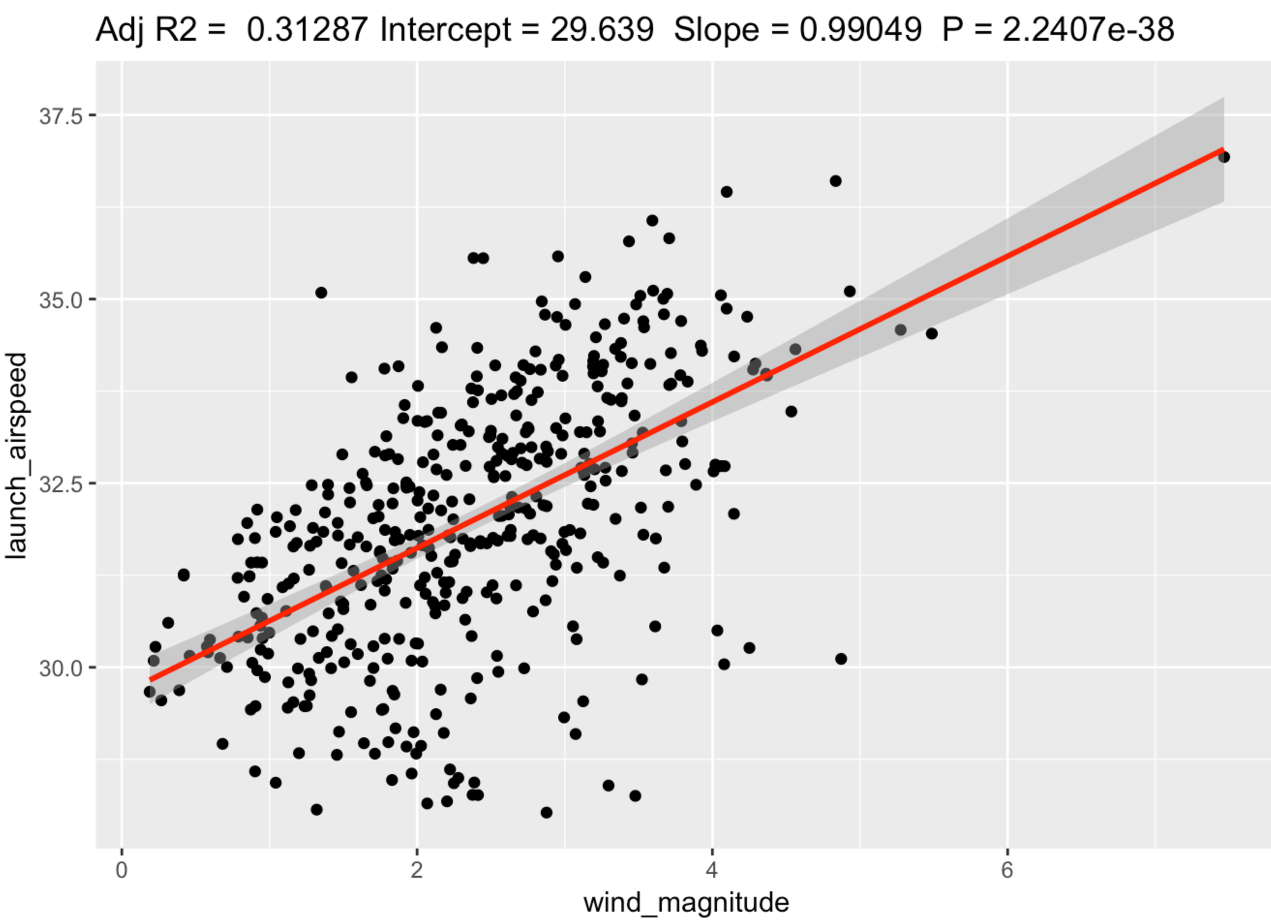
```r
ggplotRegression(lm(static_pressure~rel_humidity, data = summary_data))
```



```
par(mfrow=c(2,2))

### regression association between air speed and wind magnitude.
model_speedair_windmag <- lm(launch_airspeed~wind_magnitude)
model_speedair_windmag %>% summary()
```

```
##
## Call:
## lm(formula = launch_airspeed ~ wind_magnitude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8303 -0.8506  0.0955  0.9702  4.1091
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    29.63947    0.17755  166.94   <2e-16 ***
## wind_magnitude  0.99049    0.06933   14.29   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.459 on 445 degrees of freedom
## Multiple R-squared:  0.3144, Adjusted R-squared:  0.3129
## F-statistic: 204.1 on 1 and 445 DF,  p-value: < 2.2e-16
```

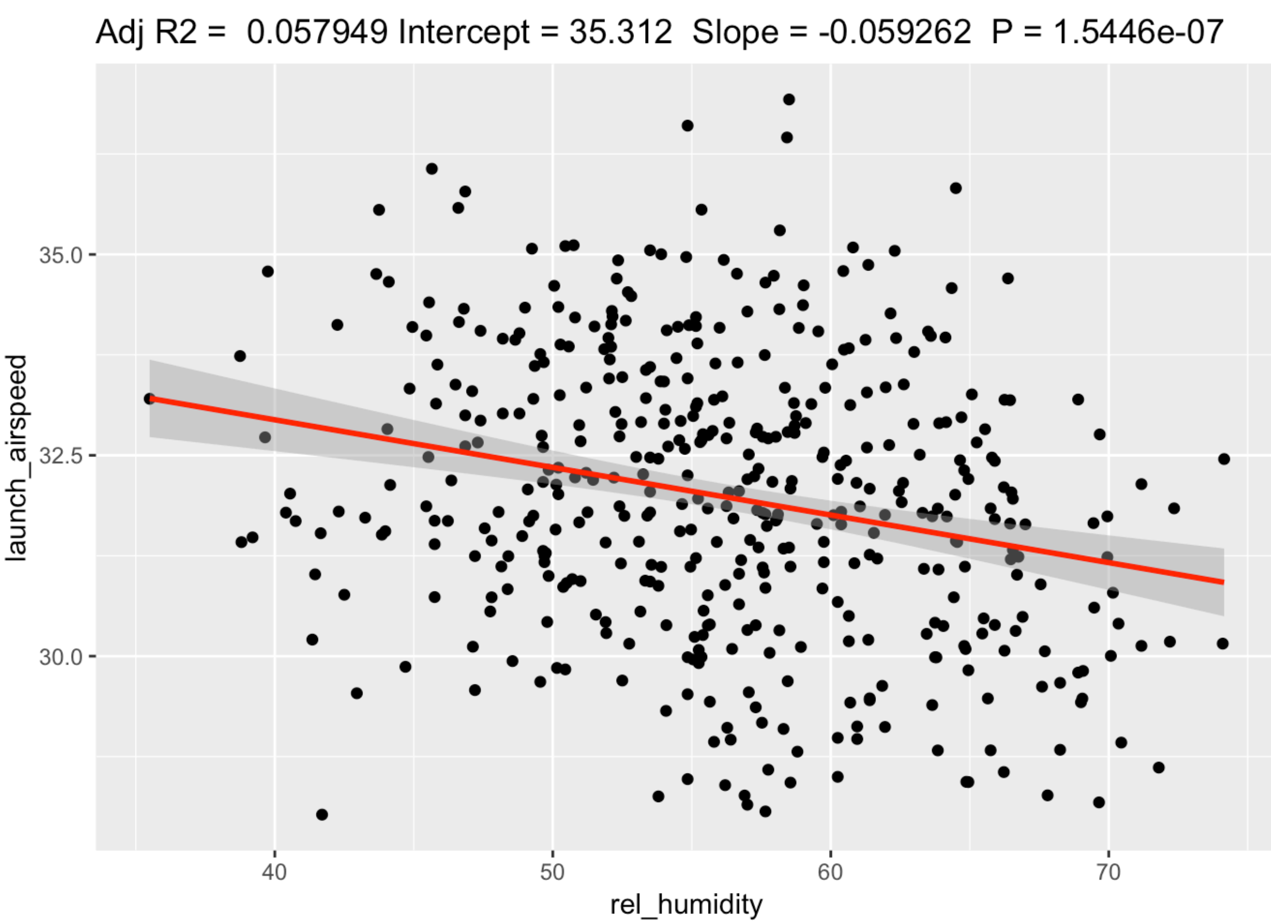
```
ggplotRegression(lm(launch_airspeed~wind_magnitude, data = summary_data))
```



```
### regreesion between air speed and humidity
model_speedair_hum <- lm(launch_airspeed~rel_humidity)
summary(model_speedair_hum)
```

```
##
## Call:
## lm(formula = launch_airspeed ~ rel_humidity)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8141 -1.2147  0.0033  1.2590  5.0836
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.31247    0.63080  55.981  < 2e-16 ***
## rel_humidity -0.05926    0.01111  -5.332 1.54e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.708 on 445 degrees of freedom
## Multiple R-squared:  0.06006,    Adjusted R-squared:  0.05795
## F-statistic: 28.44 on 1 and 445 DF,  p-value: 1.545e-07
```

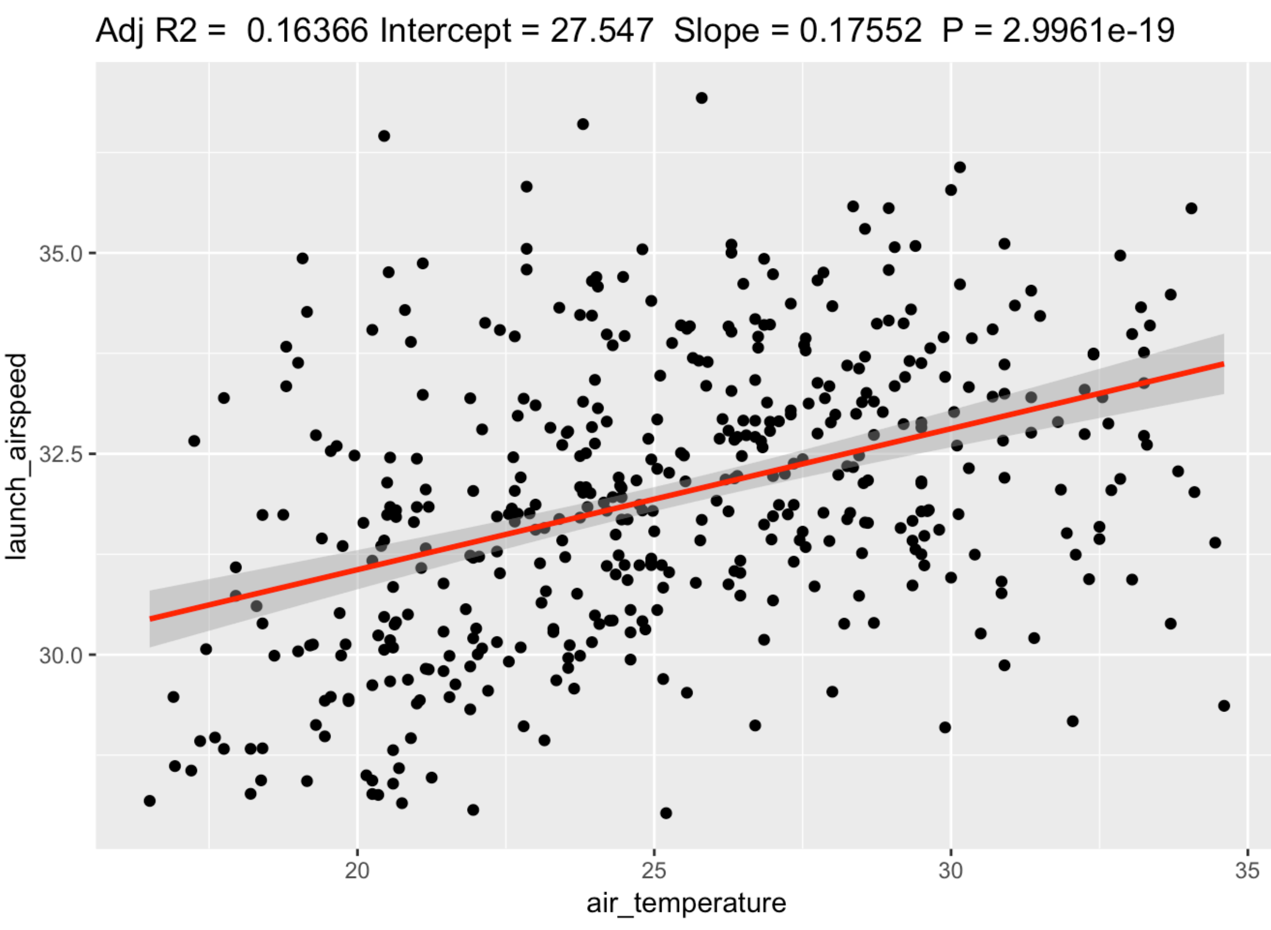
```
ggplotRegression(lm(launch_airspeed ~ rel_humidity, data = summary_data))
```




```
### regreesion between air speed and air temperature
model_speedair_temp <- lm(launch_airspeed~air_temperature)
summary(model_speedair_temp)
```

```
##
## Call:
## lm(formula = launch_airspeed ~ air_temperature)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2570 -1.1815 -0.0032  0.9742  5.3194
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     27.54733    0.47752  57.689   <2e-16 ***
## air_temperature  0.17552    0.01868   9.396   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.61 on 445 degrees of freedom
## Multiple R-squared:  0.1655, Adjusted R-squared:  0.1637
## F-statistic: 88.28 on 1 and 445 DF,  p-value: < 2.2e-16
```

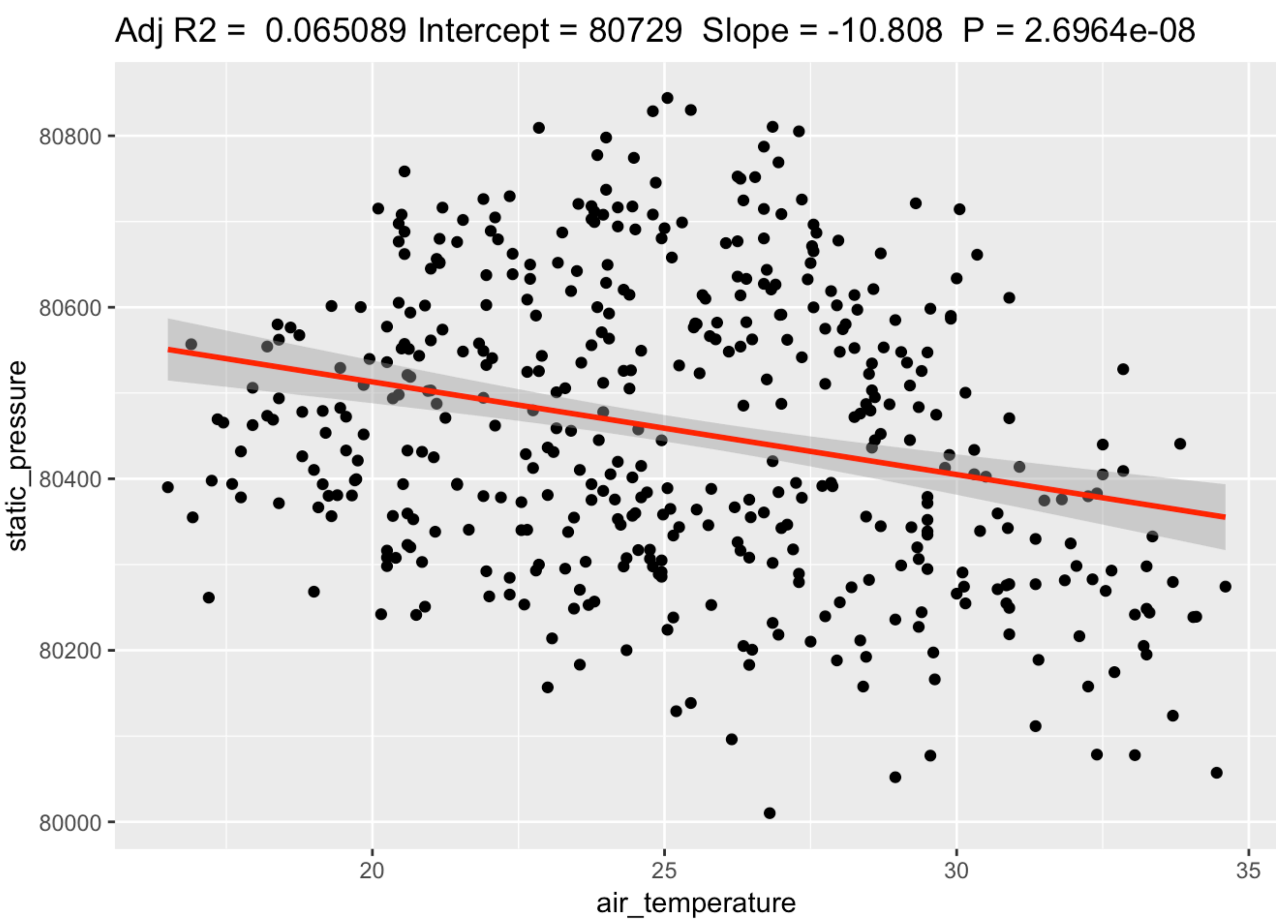
```
ggplotRegression(lm(launch_airspeed~air_temperature, data = summary_data))
```




```
### regression between static pressue and air temperature
model_stpre_temp <- lm(static_pressure~air_temperature)
summary(model_stpre_temp)
```

```
##
## Call:
## lm(formula = static_pressure ~ air_temperature)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -429.39 -124.40  -15.49  133.79  385.63
##
## Coefficients:
##                  Estimate Std. Error  t value Pr(>|t|)
## (Intercept)     80729.186     48.800 1654.303  < 2e-16 ***
## air_temperature   -10.808      1.909   -5.661  2.7e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 164.5 on 445 degrees of freedom
## Multiple R-squared:  0.06719,    Adjusted R-squared:  0.06509
## F-statistic: 32.05 on 1 and 445 DF,  p-value: 2.696e-08
```

```
ggplotRegression(lm(static_pressure~air_temperature, data = summary_data))
```




```
### NO clear regreesion between air speed and static pressure
model_speedair_stpre <- lm(launch_airspeed~static_pressure)
summary(model_speedair_stpre)
```

```
##
## Call:
## lm(formula = launch_airspeed ~ static_pressure)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7299 -1.2150 -0.0326  1.2818  5.2031
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -66.964282  39.179645  -1.709   0.0881 .
## static_pressure   0.001230   0.000487   2.525   0.0119 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.749 on 445 degrees of freedom
## Multiple R-squared:  0.01413,    Adjusted R-squared:  0.01191
## F-statistic: 6.377 on 1 and 445 DF,  p-value: 0.01191
```

```
# ggplotRegression(lm(launch_airspeed~static_pressure, data = summary_data))



### No regreesion between air speed and wind direction
model_speedair_windir <- lm(launch_airspeed~wind_direction)
summary(model_speedair_windir)
```

```
##
## Call:
## lm(formula = launch_airspeed ~ wind_direction)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9146 -1.2032 -0.0731  1.2144  4.9568
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.199e+01  1.024e-01  312.43   <2e-16 ***
## wind_direction 3.675e-04  1.314e-03    0.28     0.78
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.762 on 445 degrees of freedom
## Multiple R-squared:  0.0001757,  Adjusted R-squared:  -0.002071
## F-statistic: 0.07821 on 1 and 445 DF,  p-value: 0.7799
```

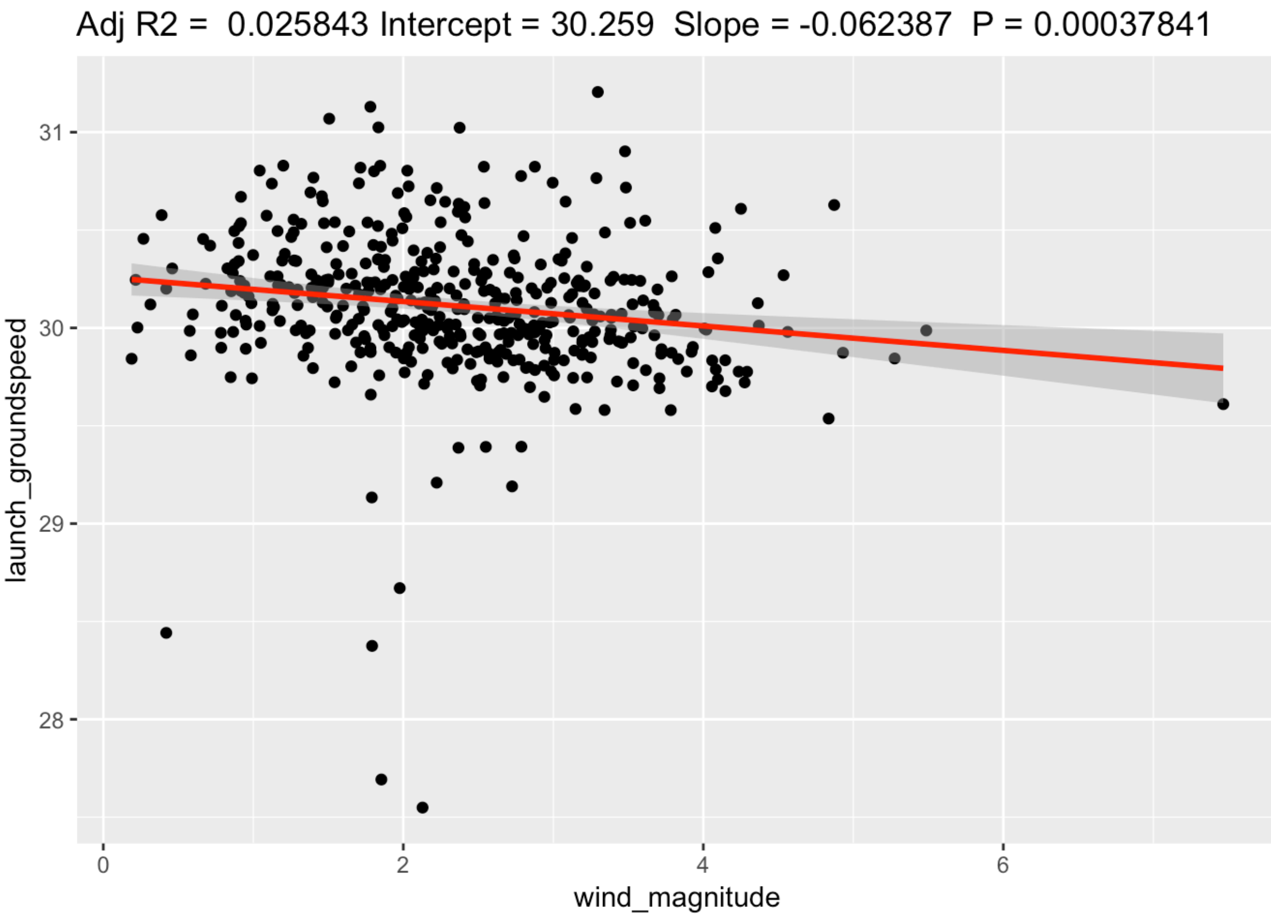
```
# ggplotRegression(lm(launch_airspeed~wind_direction
#                      , data = summary_data))



### NO clear regreesion between air speed and preflight voltage
model_speedair_prevol <- lm(launch_airspeed~preflight_voltage)
summary(model_speedair_prevol)
```

```
##
## Call:
## lm(formula = launch_airspeed ~ preflight_voltage)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9787 -1.2077  0.0059  1.2209  4.9241
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        11.4414    14.4659   0.791    0.429
## preflight_voltage   0.6388     0.4500   1.420    0.156
##
## Residual standard error: 1.746 on 429 degrees of freedom
##   (16 observations deleted due to missingness)
## Multiple R-squared:  0.004675,   Adjusted R-squared:  0.002355
## F-statistic: 2.015 on 1 and 429 DF,  p-value: 0.1565
```

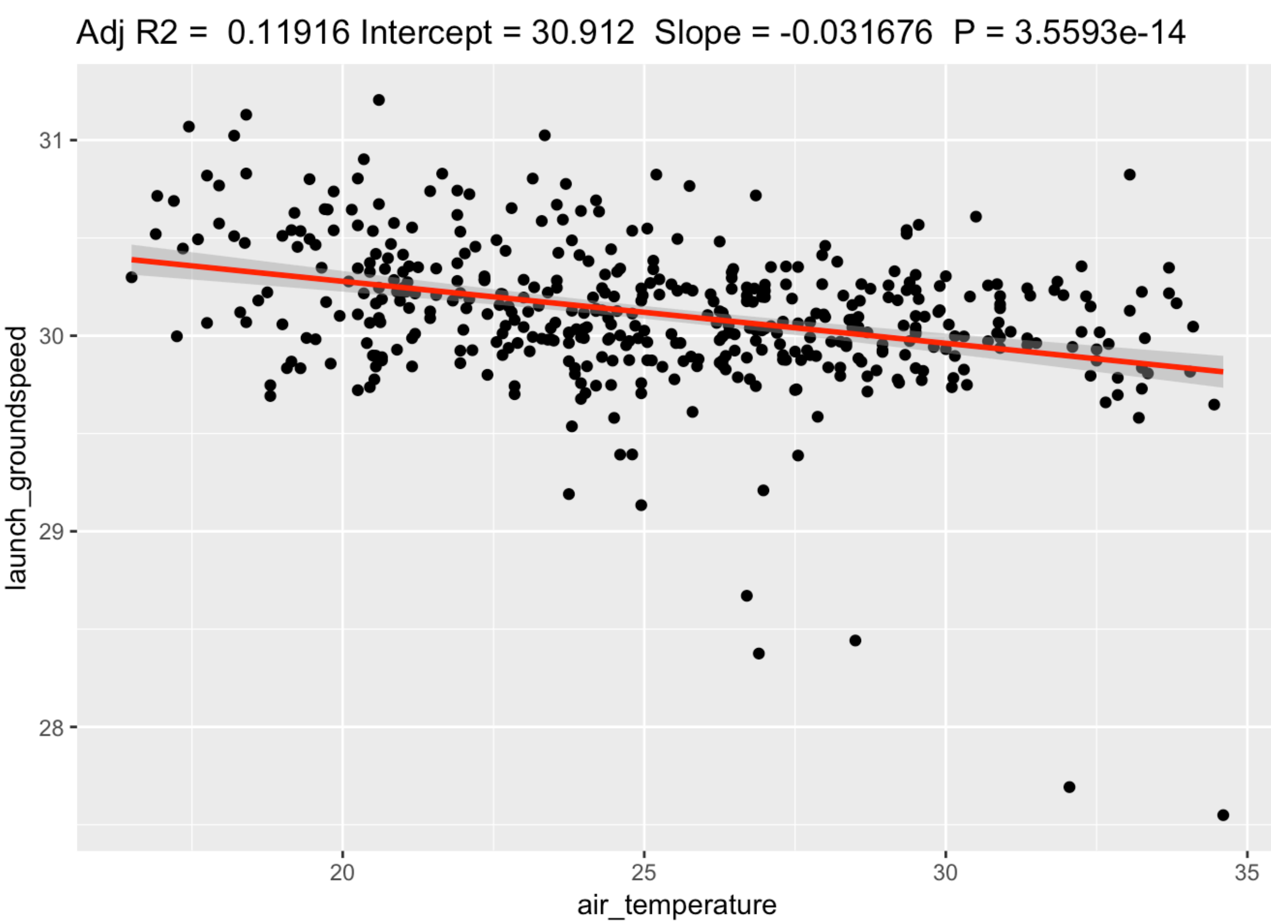
```
# ggplotRegression(lm(launch_airspeed~preflight_voltage, data = summary_data))


### regreesion between ground speed and wind magnitude
model_grdspeed_winmag <- lm(launch_groundspeed~wind_magnitude,data=summary_data)
summary(model_grdspeed_winmag)
```

```
##
## Call:
## lm(formula = launch_groundspeed ~ wind_magnitude, data = summary_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.57771 -0.17948 -0.01253  0.17235  1.15158
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    30.25938    0.04460 678.473  < 2e-16 ***
## wind_magnitude -0.06239    0.01742  -3.582 0.000378 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3665 on 445 degrees of freedom
## Multiple R-squared:  0.02803,    Adjusted R-squared:  0.02584
## F-statistic: 12.83 on 1 and 445 DF,  p-value: 0.0003784
```

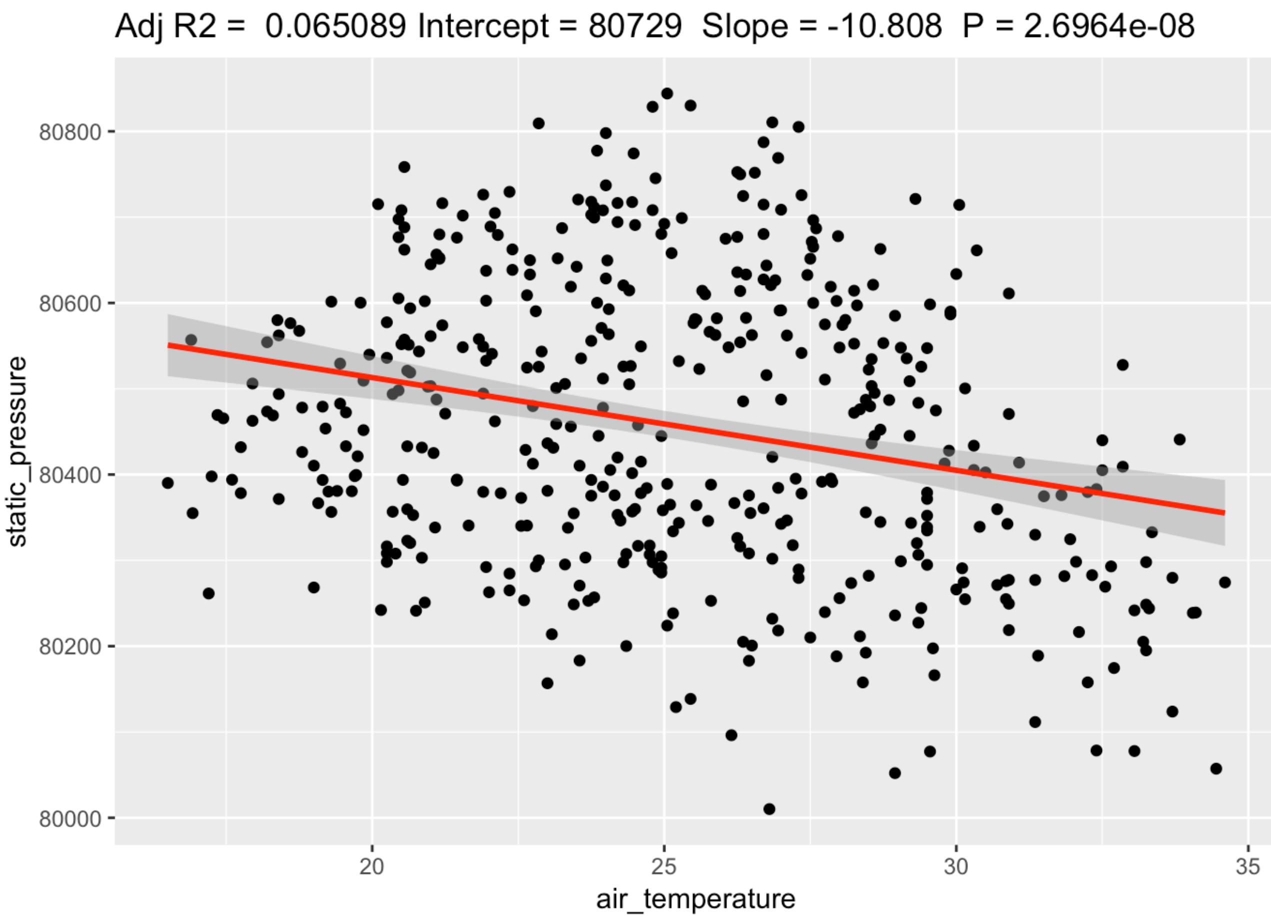
```
ggplotRegression(lm(launch_groundspeed~wind_magnitude, data = summary_data))
```



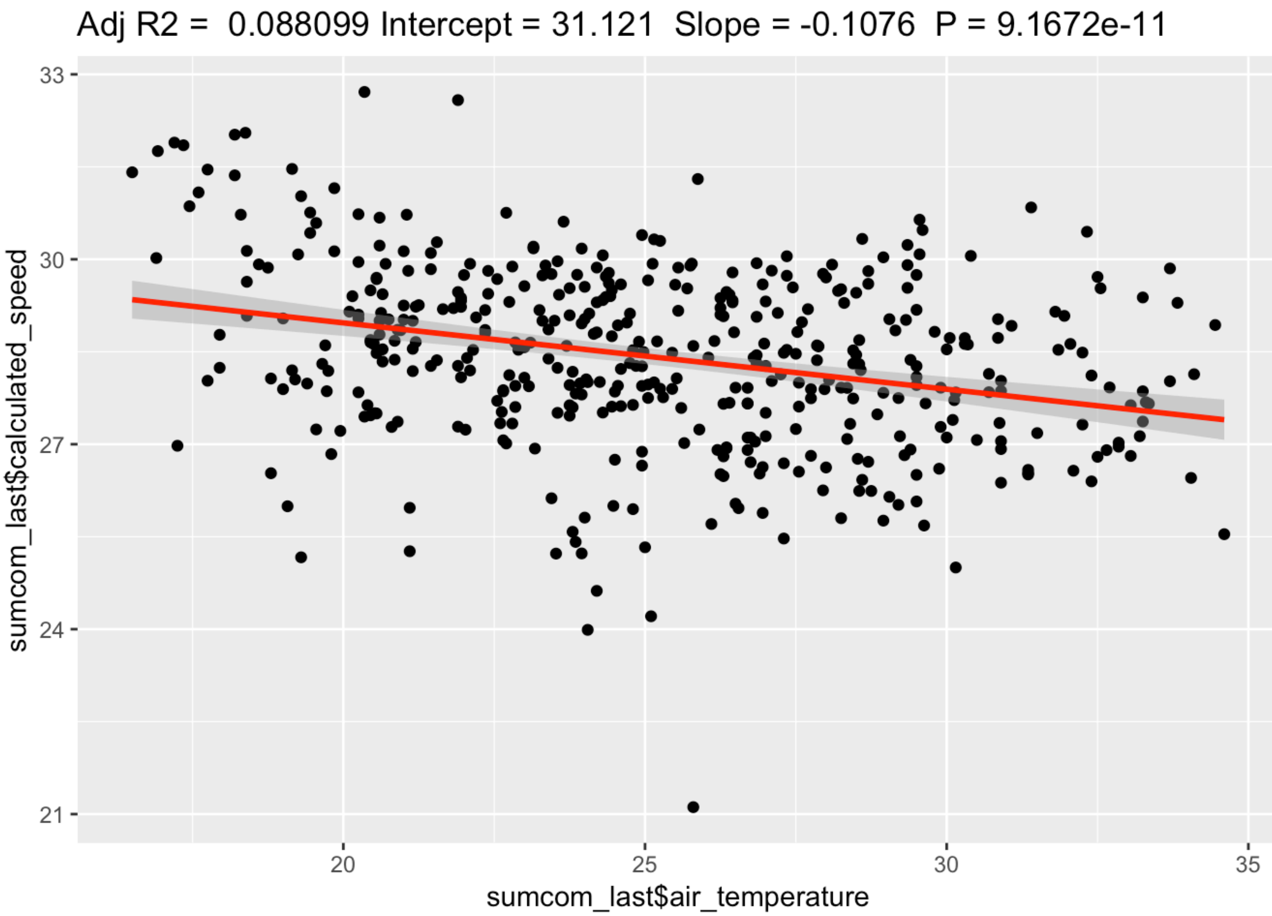

```
ggplotRegression(lm(launch_groundspeed~air_temperature, data = summary_data))
```



```
#ggplotRegression(lm(launch_groundspeed~rel_humidity, data = summary_data))

ggplotRegression(lm(static_pressure~air_temperature, data = summary_data))
```



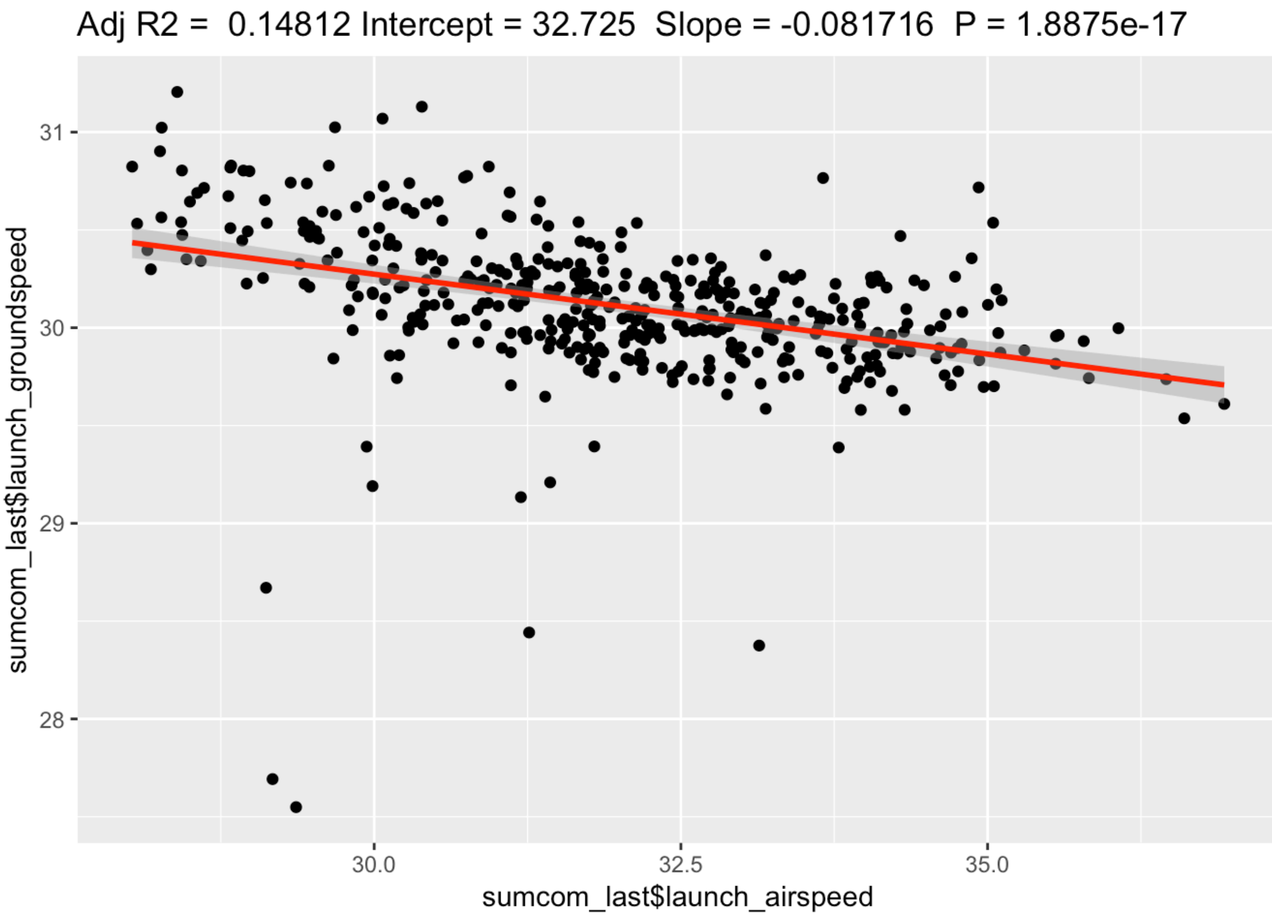

```
ggplotRegression(lm(sumcom_last$calculated_speed~sumcom_last$air_temperature, data = sumcom_last))##
```




```
#### regression between ground and air speed
lm(sumcom_last$launch_groundspeed~sumcom_last$launch_airspeed) %>% summary()
```

```
##
## Call:
## lm(formula = sumcom_last$launch_groundspeed ~ sumcom_last$launch_airspeed)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.77682 -0.13570  0.01542  0.15860  0.88791
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 32.72517    0.29527 110.832   <2e-16 ***
## sumcom_last$launch_airspeed -0.08172    0.00922  -8.863   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3427 on 445 degrees of freedom
## Multiple R-squared:   0.15,  Adjusted R-squared:  0.1481
## F-statistic: 78.55 on 1 and 445 DF,  p-value: < 2.2e-16
```

```
plot(sumcom_last$launch_airspeed,sumcom_last$launch_groundspeed) # seems the higher airspeed lower th
e ground speed

ggplotRegression(lm(sumcom_last$launch_groundspeed~sumcom_last$launch_airspeed))
```



`Answer:` - I find 17162 went through 7.466193 with airspeed 36.9292!!
- From the several regression models and plots I find - the higher the temp, higher the wind magnitude - the higher the temp, lower the humidity - the higher the temp, lower the static pressure - the higher the humidity,higher static pressure - the higher the wind magnitude, higher the airspeed - the lower the humidity, higher the air speed - the higher the temp, higher the air speed (not strong) - the higher the temp, the lower the calculated speed. - the higher the wind magnitude, the lower the ground speed.

- **It seems wind and temp really affects the lauch_speed, and humidity**

Modelling for distance

```
model_dis_wind <- lm(data = sumcom_last, distance_travel~ wind_magnitude)
summary(model_dis_wind)
```

```
##
## Call:
## lm(formula = distance_travel ~ wind_magnitude, data = sumcom_last)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.931  -8.993  -2.854   4.184  74.545
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    441.6552     1.9271 229.182  < 2e-16 ***
## wind_magnitude  -6.3179     0.7525  -8.395 6.25e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.83 on 445 degrees of freedom
## Multiple R-squared:  0.1367, Adjusted R-squared:  0.1348
## F-statistic: 70.48 on 1 and 445 DF,  p-value: 6.253e-16
```

```
model_dis_air <- lm(data = sumcom_last, distance_travel~ air_temperature)
summary(model_dis_air)
```

```
##
## Call:
## lm(formula = distance_travel ~ air_temperature, data = sumcom_last)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -77.220  -9.226  -1.415   6.675  59.056
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     470.2583     4.6044 102.132   <2e-16 ***
## air_temperature  -1.7242     0.1801  -9.572   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.52 on 445 degrees of freedom
## Multiple R-squared:  0.1707, Adjusted R-squared:  0.1689
## F-statistic: 91.62 on 1 and 445 DF,  p-value: < 2.2e-16
```

```
model_dis_hum <- lm(data = sumcom_last, distance_travel~ rel_humidity)
summary(model_dis_hum)
```

```
##
## Call:
## lm(formula = distance_travel ~ rel_humidity, data = sumcom_last)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -79.052  -9.274  -2.531   6.076  68.445
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  404.9030     6.2061  65.243  < 2e-16 ***
## rel_humidity   0.3881     0.1093   3.549 0.000428 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.81 on 445 degrees of freedom
## Multiple R-squared:  0.02753,    Adjusted R-squared:  0.02534
## F-statistic:  12.6 on 1 and 445 DF,  p-value: 0.0004275
```

```
model_dis_pre <- lm(data = sumcom_last, distance_travel~ static_pressure)
summary(model_dis_pre)
```

```
##
## Call:
## lm(formula = distance_travel ~ static_pressure, data = sumcom_last)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -81.052  -9.880  -1.997   5.491  66.155
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.556e+03  3.779e+02   4.116 4.59e-05 ***
## static_pressure -1.403e-02  4.697e-03  -2.987  0.00297 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.87 on 445 degrees of freedom
## Multiple R-squared:  0.01966,    Adjusted R-squared:  0.01745
## F-statistic: 8.923 on 1 and 445 DF,  p-value: 0.002971
```

```
model_dis_la <- lm(data = sumcom_last, distance_travel~ launch_airspeed)
summary(model_dis_la)
```

```
##
## Call:
## lm(formula = distance_travel ~ launch_airspeed, data = sumcom_last)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.199  -7.132  -0.702   5.790  40.559
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     657.9912     9.7517   67.47   <2e-16 ***
## launch_airspeed  -7.2317     0.3045  -23.75   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.32 on 445 degrees of freedom
## Multiple R-squared:  0.559,  Adjusted R-squared:  0.558
## F-statistic:   564 on 1 and 445 DF,  p-value: < 2.2e-16
```

```
model_dis_lg <- lm(data = sumcom_last, distance_travel~ launch_groundspeed)
summary(model_dis_lg)
```

```
##
## Call:
## lm(formula = distance_travel ~ launch_groundspeed, data = sumcom_last)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -64.862  -8.157  -1.461   5.530  70.844
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -373.395     53.339      -7 9.46e-12 ***
## launch_groundspeed   26.572      1.771      15  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.89 on 445 degrees of freedom
## Multiple R-squared:  0.3359, Adjusted R-squared:  0.3344
## F-statistic: 225.1 on 1 and 445 DF,  p-value: < 2.2e-16
```

```
# ggplotRegression(lm(data = sumcom_last, distance_travel~ air_temperature))
```

`Answer` - It is actually more acurate to study distance travelled than lanch air speed because lauch air speed might due to batter effects and power given manually. - higher wind, lower distance - higher temp, lower distance - higher humdity, higher distance - lower pressure, higher distance.

```
model_full <- lm(data=sumcom_last, distance_travel~air_temperature+wind_magnitude+rel_humidity+static
_pressure)
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:plotly':
##
##     select
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
step_model <- stepAIC(model_full, direction = "both",trace = F)

summary(step_model)
```

```
##
## Call:
## lm(formula = distance_travel ~ air_temperature + wind_magnitude +
##     static_pressure, data = sumcom_last)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -55.097  -8.887  -1.640   6.389  62.007
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2.698e+03  3.245e+02   8.312 1.16e-15 ***
## air_temperature -1.800e+00  1.696e-01 -10.613  < 2e-16 ***
## wind_magnitude  -5.420e+00  6.723e-01  -8.063 7.02e-15 ***
## static_pressure -2.750e-02  4.019e-03  -6.842 2.60e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.94 on 443 degrees of freedom
## Multiple R-squared:  0.334,  Adjusted R-squared:  0.3295
## F-statistic: 74.07 on 3 and 443 DF,  p-value: < 2.2e-16
```

```
step_model
```

```
##
## Call:
## lm(formula = distance_travel ~ air_temperature + wind_magnitude +
##     static_pressure, data = sumcom_last)
##
## Coefficients:
##     (Intercept)  air_temperature   wind_magnitude  static_pressure
##       2697.5054          -1.7995          -5.4204          -0.0275
```

```
 # and lauch_airspeed is correlction with wing series a little bit, but some wings usef very few, can not represent, while wing  15SPJJJ09019061 used 45 times could represent statistically sig.
```

`Answer` - I predict a model that explain quantatively the distance travel due to weather condition.

$$distance = -1.7995 * air.temperature - 5.4204 * wind.magnitude - 0.0275 * static_pressure + 2.698e + 03$$

Conclusion and insights

- This analysis gives me several fidings though need further study for validation:
- The voltage has missing values
- There are fours flights that their location is weird when looking at map at launch (upper left corner, need check)
- specific battery and body series affects the creating of missing voltage, especially body 577209618523054080 always gives missing values ever since use it. **Avoid from using 577209618523054080 to check error**
- The components are not used equally frequenly which possibly causes overuse to affects the quality of drone system.
- The wing series 15SPJJJ09028064 affects the launch_airspeed and need check.
- on 2018-09-30, only battery 15SPJJJ10056048 used the whole day and causes largest average error, and it needs to be checked, the wing and body also a affect a little bit but not as serious as this battery.

- Some possible relation:
- the higher the air temperature, the higher the wind, and the lower the static pressue and lower humidity.
- I think it's better to study the distance travel in same time other than launch airspeed and have fidings that explain the distance travel , and therefore can calculate the average speed for the trip.
- `distance = -1.7995 * air.temperature -5.4204 * wind.magnitude  -0.0275 * static_pressure + 2.698e+03`
- the unexplained behavior then make sense why `2018-09-23` has low lauch_airspeed but highest distance: the wind is small and it's pretty cold with high humidity and low pressure
- *(physics: rainy/cloudy has lower pressure, and lower wind and colder, but lower pressure makes techinician depressed mood therefore might cause some mistake when choosing and installing the components and monitoring the positions. history:2018-09-23 weather in Rwanda was cloudy and high humidity.)*
- So to travel longer distance in same period, beside checking the components to used at best performance, the weather matters too, and it is preferable that to fly at lower temperature with higher humidity, which might counter my common sense.

- And hence people at Zipline can use weahter to best perform the fast delivery and power-saving drone system.

- Please refer to tableau visualizaed plots and python for more.