Final Project

Ruohui Chen

4/23/2018

Contents


```
#total 63 people in the AWRL dataset
AWRL<-read_excel("STRAIN_Noise_and_Control.xlsx", 1)
colnames(AWRL)[6]<-c("Noise_Level")
AWRL$Night<-paste("AWRL",AWRL$Night,sep = "_")
AWRL$Plane<-c("1")
AWRL$Plane<-as.numeric(as.character(AWRL$Plane))

#same 63 people in the AWR(control) dataset
AWR<-read_excel("STRAIN_Noise_and_Control.xlsx", 2)
AWR$Night<-paste("AWR",AWR$Night,sep = "_")
AWR$Noise_Level<-0
AWR$Noise_Duration_sec<-0
AWR$Noise_Slope<-0
AWR$Plane<-c("0")
AWR$Plane<-as.numeric(as.character(AWR$Plane))

Total <- rbind(AWRL, AWR)
Total$ID<-as.factor(Total$ID)
#Total$Awakening<-as.factor(Total$Awakening)
Total<-Total[,c("ID","Awakening","Noise_Level","Background_Noise","Night")]
```

Introduction

The demand for mobility in general and air traffic in particular has been strongly increasing over the past few years. As a minimum interval between two starting or two landing planes is necessary for safety reasons,

evasion of air traffic to shoulder hours and even the night time has been observed in the past and will even increase in the future. Therefore, the strain of residents living in the vicinity of airports is likely to increase due to noise emitted from nocturnal air traffic. Previous study has been done to estimate the sample size needed for field studies on the effects of aircraft noise on sleep, but not much research has been done to investigate the effect of single aircraft event on people's sleep.

Methods

Study design

There are total 63 participants in the dataset, each participant's awakening status had been moniterd for nights that had flights over the house, and nights that didn't have flights over the house, and every participant was measured repeated times per night. For nights had flights over the house, indoor noise level of single aircraft event was monitored (Noise_Level). Participants' awakening status were determined by the cardiology wave, which was detected by a device participants wear during their sleep. The first night's data was disgarded to let participants get used to the device they wearing during the sleep.

Inferential statistical analyse

For this specific dataset, we decided to fit mixed effect models with random intercept for the dataset best. Here let X_1 denotes the background noise, X_2 denotes the noise level caused by airplane.

We first fit the dataset with model that has background noise, interaction term between noise level and plane, and the interaction term between background noise and noise level caused by airplane. When the interaction term is significant, the interpretation for this model will be, controll background noise, if there is one unit increase in the noise_level of airplane, the odds of awakenign will be increased by $exp(\beta_2 + \beta_3 * X_1)$

$$Y_{ij} = \beta_0 + \beta_1 * X_{1j} + \beta_2 * X_{2j} + \beta_3 * X_{1j} * X_{2j} + b_i + e_{ij}$$

After fitting the dataset with the random intercept model, we then fit the dataset with the GEE(exchangable) model to see the difference between random intercept model and GEE(exchangeable) model.

Review of the strengths and weakness

The key advantage to longitudinal studies is the ability to show the patterns of a variable over time. This is one powerful way in which we come to learn about cause-and-effect relationships. Depending on the scope of the study, longitudinal observation can also help to discover "sleeper effects" or connections between different events over a long period of time; events that might otherwise not be linked.

There are, of course, drawbacks to longitudinal studies, panel attrition being one of them. If you are dependent on the same group of 63 subjects for a study that takes place once every year, for twenty years, obviously some of those subjects will no longer be able to participate, either due to death, refusal, or even changes in contact information and address. That cuts down on usable data we can draw conclusions from.

Another weakness is that while longitudinal data is being collected at multiple points, those observation periods are pre-determined and cannot take into account whatever has happened in between those touch points.

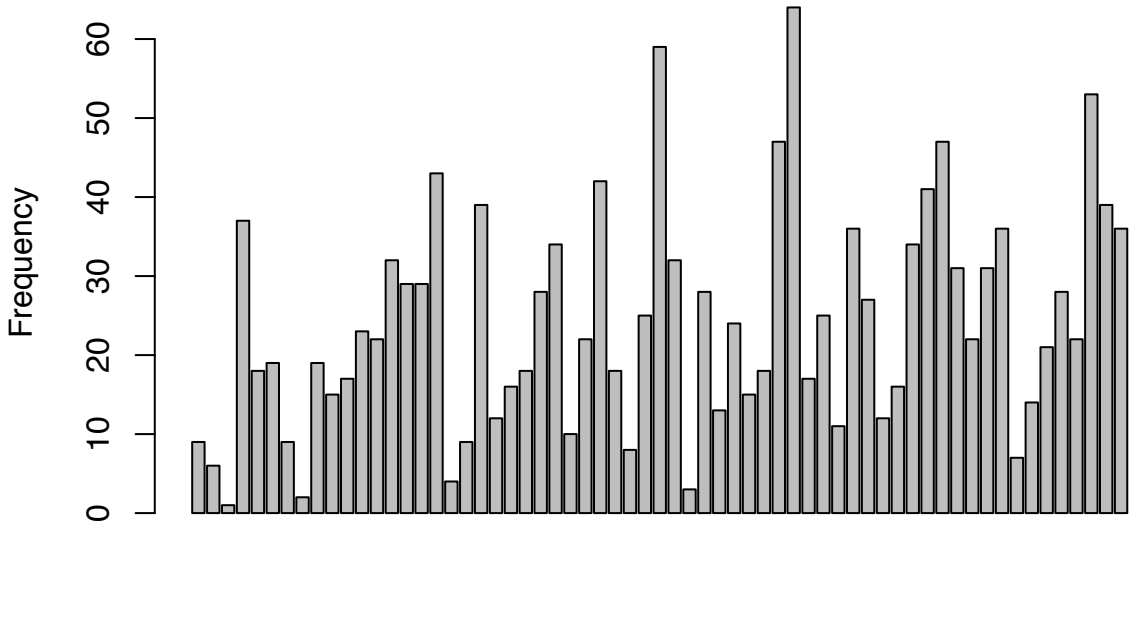
A third disadvantage is the idea of panel conditioning, where over time, respondents can often unknowingly change their qualitative responses to better fit what they consider to be the observer's intended goal. The process of the study itself has changed how the subject or respondent views the questions.

Results

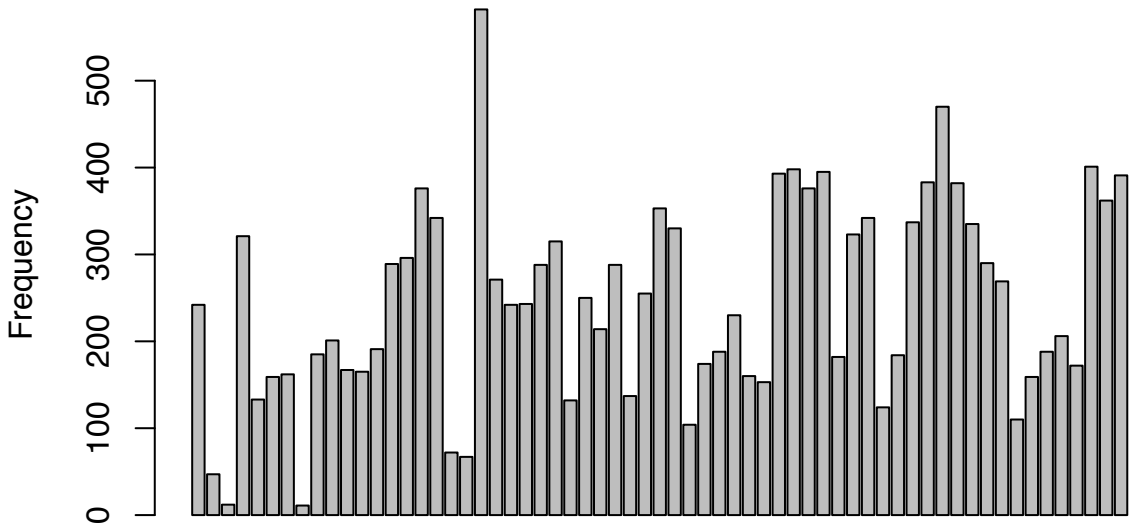
Descriptive statistics

For nights have flights over the house(AWRL)

```
#Descriptive statistics

#For AWRL(nights have flights over the house)
#number of participants
num.AWRL<-length(unique(AWRL$ID))
#summary statistics for indoor noise level of aircraft in the nights have flights over the house
Lmax.AWRL<-AWRL$Noise_Level
Background.AWRL<-AWRL$Background_Noise
Plane_Noise<-matrix(c(mean(Lmax.AWRL), sd(Lmax.AWRL), median(Lmax.AWRL),min(Lmax.AWRL), max(Lmax.AWRL))
Plane_Noise<-as.data.frame(Plane_Noise)
colnames(Plane_Noise)<-c("Mean", "Sd","Median","Minimum","Maximum")
row.names(Plane_Noise)<-c("Plane_Noise")
#print(Plane_Noise)
Bg_Noise<-matrix(c(mean(Background.AWRL), sd(Background.AWRL), median(Background.AWRL),
                   min(Background.AWRL),max(Background.AWRL)), nrow = 1)
Bg_Noise<-as.data.frame(Bg_Noise)
colnames(Bg_Noise)<-c("Mean", "Sd","Median","Minimum","Maximum")
row.names(Bg_Noise)<-c("Background_Noise")
#print(Bg_Noise)

#Number of awakes per individual for AWRL
CtAWRL_Awake<-plyr::count(AWRL, vars = "ID", wt_var = "Awakening")
number_Awakes<-matrix(c(mean(CtAWRL_Awake$freq), sd(CtAWRL_Awake$freq), median(CtAWRL_Awake$freq),
                        min(CtAWRL_Awake$freq), max(CtAWRL_Awake$freq)), nrow = 1)
number_Awakes<-as.data.frame(number_Awakes)
colnames(number_Awakes)<-c("Mean","Sd","Median","Minimum","Maximum")
row.names(number_Awakes)<-c("Awakes/person")
#print(number_Awakes)

barplot(CtAWRL_Awake$freq,
main = "barplot for # of awakes/person during nights have flights over house", ylab = "Frequency",
xlab = "people")
```




```
#Number of observations per individual for nights have flights over house
CtAWRL_Obs<-plyr::count(AWRL, vars = "ID", wt_var = NULL)
number_Obs<-matrix(c(mean(CtAWRL_Obs$freq), sd(CtAWRL_Obs$freq), median(CtAWRL_Obs$freq),
                     min(CtAWRL_Obs$freq), max(CtAWRL_Obs$freq)), nrow = 1)
number_Obs<-as.data.frame(number_Obs)
colnames(number_Obs)<-c("Mean","Sd","Median","Minimum","Maximum")
row.names(number_Obs)<-c("Observations/person")
#print(number_Obs)

barplot(CtAWRL_Obs$freq,
main = "barplot for # of obs/person during nights have flights over house", ylab = "Frequency",
xlab = "people")
```



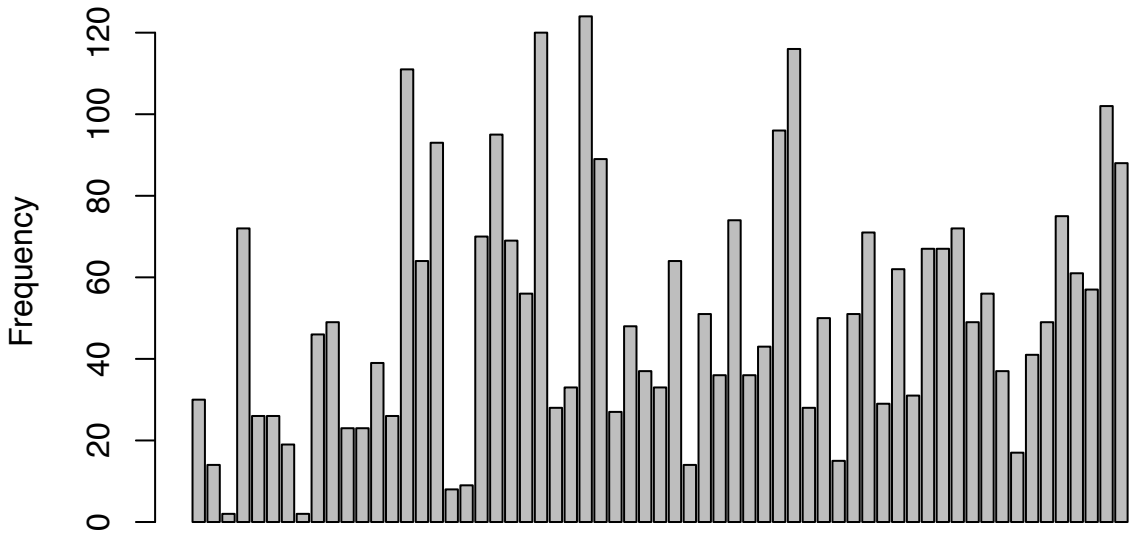

```
rbind(Plane_Noise, Bg_Noise, number_Awakes, number_Obs)
```

```
##                           Mean         Sd Median Minimum Maximum
## Plane_Noise           43.70025   8.709769   43.8    13.8    73.2
## Background_Noise      28.76758   6.609283   27.8    16.4    58.8
## Awakes/person         24.19048  14.005595   22.0     1.0    64.0
## Observations/person 246.33333 115.787486  242.0    11.0   582.0
```

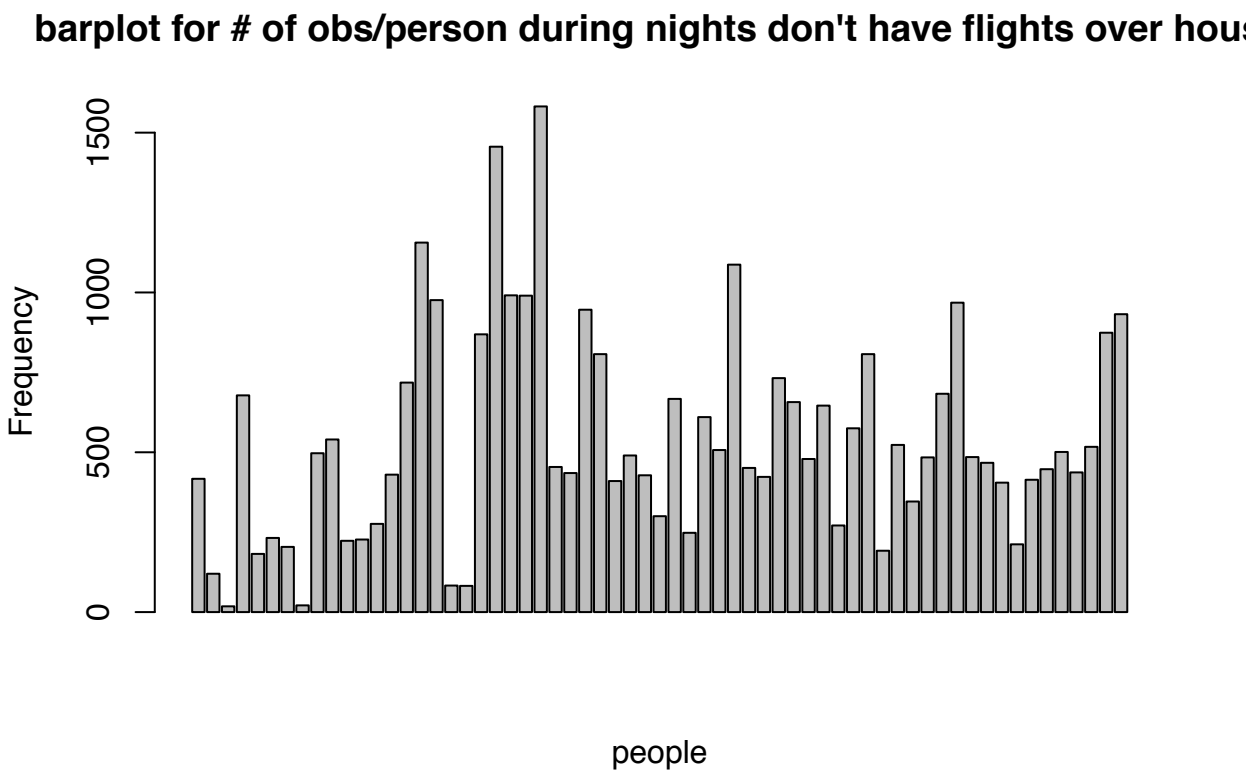
For nights don't have flights over the hosue(AWR)

```
#For AWR(nights don't have flights over the house)
num.AWR<-length(unique(AWR$Night))
#summary statistics for indoor noise level of aircraft in the nights don't have flights over the house
Background.AWR<-na.omit(AWR$Background_Noise)
Background_Noise<-matrix(c(mean(Background.AWR), sd(Background.AWR), median(Background.AWR),
                           min(Background.AWR),max(Background.AWR)), nrow = 1)
Background_Noise<-as.data.frame(Background_Noise)
colnames(Background_Noise)<-c("Mean", "Sd", "Median","Minimum","Maximum")
row.names(Background_Noise)<-c("Background_Noise")
#print(Background_Noise)


#number of awakes per individual for AWR
CtAWR_Awake<-plyr::count(AWR, vars = "ID", wt_var = "Awakening")
number_Awakes<-matrix(c(mean(CtAWR_Awake$freq), sd(CtAWR_Awake$freq), median(CtAWR_Awake$freq),
                        min(CtAWR_Awake$freq), max(CtAWR_Awake$freq)), nrow = 1)
number_Awakes<-as.data.frame(number_Awakes)
colnames(number_Awakes)<-c("Mean","Sd","Median","Minimum","Maximum")
```

```
row.names(number_Awakes)<-c("Awakes/person")
#print(number_Awakes)

barplot(CtAWR_Awake$freq,
main = "barplot for # of awakes/person during nights don't have flights over house", ylab = "Frequency"
xlab = "people")
```




```
#Number of observations per individual for nights have flights over house
CtAWR_Obs<-plyr::count(AWR, vars = "ID", wt_var = NULL)
number_Obs<-matrix(c(mean(CtAWR_Obs$freq), sd(CtAWR_Obs$freq), median(CtAWR_Obs$freq),
                     min(CtAWR_Obs$freq), max(CtAWR_Obs$freq)), nrow = 1)
number_Obs<-as.data.frame(number_Obs)
colnames(number_Obs)<-c("Mean","Sd","Median","Minimum","Maximum")
row.names(number_Obs)<-c("Observations/person")
#print(number_Obs)

barplot(CtAWR_Obs$freq,
main = "barplot for # of obs/person during nights don't have flights over house", ylab = "Frequency",
xlab = "people")
```




```
rbind(Background_Noise, number_Awakes, number_Obs)
```

```
##                            Mean         Sd Median Minimum Maximum
## Background_Noise      27.80545   6.369305   26.9    16.1    61.7
## Awakes/person         51.04762  30.096312   49.0     2.0   124.0
## Observations/person 544.20635 326.594903  484.0    18.0  1582.0
```

From the above barplots, we call tell that the background noise is almost the same for both nights that have planes fly over the house and nights that dont't have planes fly over house. For the nights that have planes fly over the house, there are about 246 observations per participant, with a standard deviation around 116. And There are average 24 awakes per person, with a standard deviation around 14. On the contrary, there are about 544 observations per person, with a standard deviation around 327 for the nights that don't have planes fly over the house. And there are average 51 awakes per person, with a standard deviation around 30.

Random effect Model with random intercept

```
#fit the random intercept model
fit1<-glmer(Awakening~Noise_Level+Background_Noise+Noise_Level*Background_Noise+(1|ID/Night),
                  data = Total,
                 family = binomial,
                 na.action = na.omit)
summary(fit1)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## Awakening ~ Noise_Level + Background_Noise + Noise_Level * Background_Noise +
##     (1 | ID/Night)
```

```
##    Data: Total
##
##      AIC      BIC   logLik deviance df.resid
##  31034.6  31087.5 -15511.3  31022.6    49769
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -0.5487 -0.3458 -0.3078 -0.2747  4.6565
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Night:ID (Intercept) 0.05650  0.2377
##  ID       (Intercept) 0.07301  0.2702
## Number of obs: 49775, groups:  Night:ID, 960; ID, 63
##
## Fixed effects:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -2.2674667  0.1094589 -20.715  < 2e-16 ***
## Noise_Level                 -0.0089497  0.0031830  -2.812 0.004927 **
## Background_Noise            -0.0012768  0.0037007  -0.345 0.730079
## Noise_Level:Background_Noise  0.0003601  0.0001055   3.413 0.000642 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) Ns_Lvl Bckg_N
## Noise_Level -0.464
## Backgrnd_Ns -0.923  0.463
## Ns_Lvl:Bc_N  0.483 -0.963 -0.519
## convergence code: 0
## Model failed to converge with max|grad| = 0.179561 (tol = 0.001, component 1)
## Model is nearly unidentifiable: very large eigenvalue
##  - Rescale variables?
## Model is nearly unidentifiable: large eigenvalue ratio
##  - Rescale variables?
```

```
fit2<-geeglm(Awakening~Noise_Level+Background_Noise+Noise_Level*Background_Noise,
             data = na.omit(Total),
             id = ID,
             corstr = "exchangeable",
             family = binomial(link = "logit"))
summary(fit2)
```

```
##
## Call:
## geeglm(formula = Awakening ~ Noise_Level + Background_Noise +
##     Noise_Level * Background_Noise, family = binomial(link = "logit"),
##     data = na.omit(Total), id = ID, corstr = "exchangeable")
##
##  Coefficients:
##                               Estimate    Std.err    Wald Pr(>|W|)
## (Intercept)                 -2.1892541  0.1448714 228.364  < 2e-16 ***
## Noise_Level                 -0.0085446  0.0042439   4.054  0.04408 *
## Background_Noise            -0.0038796  0.0050267   0.596  0.44023
## Noise_Level:Background_Noise  0.0004156  0.0001360   9.332  0.00225 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Estimated Scale Parameters:
##             Estimate Std.err
## (Intercept)   0.9969 0.06907
##
## Correlation: Structure = exchangeable  Link = identity
##
## Estimated Correlation Parameters:
##       Estimate  Std.err
## alpha 0.007591 0.001933
## Number of clusters:   126   Maximum cluster size: 1582
```

```
fit3<-geeglm(Awakening~Noise_Level+Background_Noise+Noise_Level*Background_Noise,
             data = na.omit(Total),
             id = ID,
             corstr = "independence",
             family = binomial(link = "logit"))
summary(fit3)
```

```
##
## Call:
## geeglm(formula = Awakening ~ Noise_Level + Background_Noise +
##     Noise_Level * Background_Noise, family = binomial(link = "logit"),
##     data = na.omit(Total), id = ID, corstr = "independence")
##
##  Coefficients:
##                               Estimate   Std.err   Wald Pr(>|W|)
## (Intercept)                 -1.994032  0.149617 177.62   <2e-16 ***
## Noise_Level                 -0.010417  0.004643   5.03   0.0249 *
## Background_Noise            -0.010276  0.005269   3.80   0.0512 .
## Noise_Level:Background_Noise  0.000427  0.000150   8.13   0.0044 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Estimated Scale Parameters:
##             Estimate Std.err
## (Intercept)        1  0.0693
##
## Correlation: Structure = independenceNumber of clusters:   126   Maximum cluster size: 1582
```

```
Random_intercept<-as.data.frame(coef(summary(fit1)))
GEE_exchangeable<-as.data.frame(coef(summary(fit2)))
GEE_independence<-as.data.frame(coef(summary(fit3)))

Random_intercept
```

```
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -2.26747   0.109459 -20.715 2.52e-95
## Noise_Level                 -0.00895   0.003183  -2.812 4.93e-03
## Background_Noise            -0.00128   0.003701  -0.345 7.30e-01
## Noise_Level:Background_Noise  0.00036   0.000105   3.413 6.42e-04
```

```
GEE_exchangeable
```

```
##                                Estimate  Std.err    Wald Pr(>|W|)
## (Intercept)                   -2.189254 0.144871 228.364  0.00000
## Noise_Level                   -0.008545 0.004244   4.054  0.04408
## Background_Noise              -0.003880 0.005027   0.596  0.44023
## Noise_Level:Background_Noise   0.000416 0.000136   9.332  0.00225
```

```
GEE_independence
```

```
##                                Estimate Std.err   Wald Pr(>|W|)
## (Intercept)                   -1.994032 0.14962 177.62  0.00000
## Noise_Level                   -0.010417 0.00464   5.03  0.02486
## Background_Noise              -0.010276 0.00527   3.80  0.05116
## Noise_Level:Background_Noise   0.000427 0.00015   8.13  0.00436
```

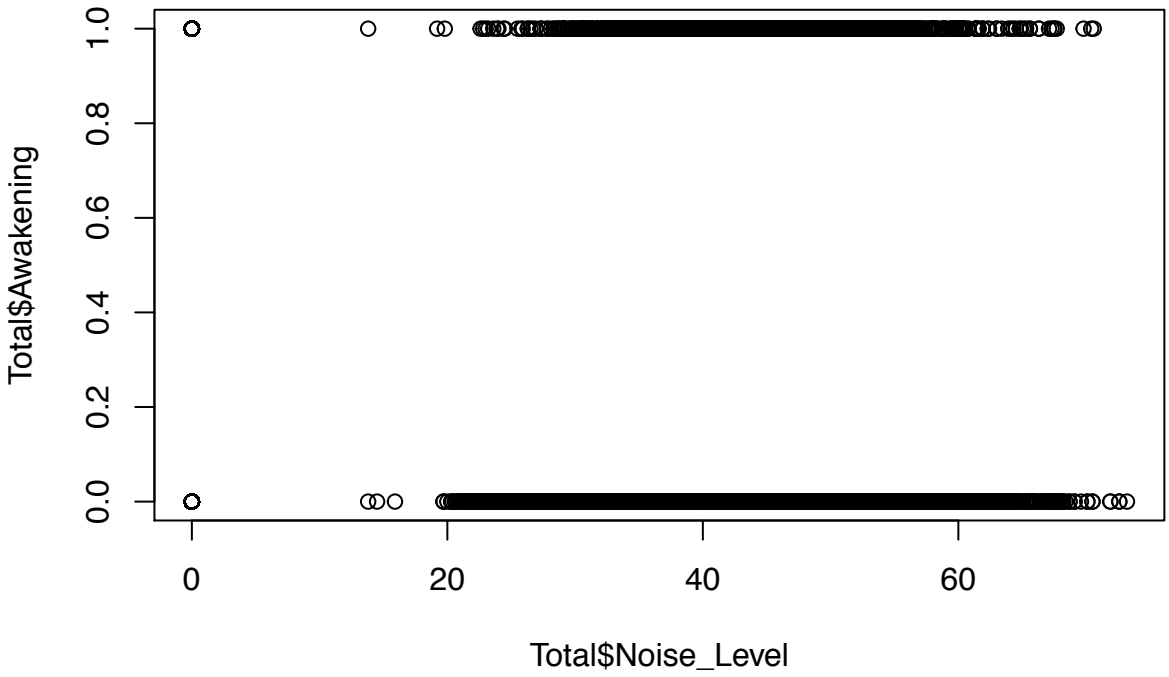
From the result of random intercept model above, we can say that controll for the background noise, one db increase in Noise_level, the odds of awakening will increase by $exp(0.00036 * BackgroundNoise - 0.00895)$, with the mean of Background noise around 28 db. Besides that, from the results we got from the random intercept model and gee models with exchangeable and independence structures, we found the random intercept model fits the dataset best. Even though the estimated β_i are similar across those three different models, but GEE models gave much larger P-value and larger standardard deviation for the $\hat{\beta}_i$ compare to the random intercept model. Another thing worth to note is that with small clusters, imbalanced design, and incomplete within-cluster confounder adjustment, exchangeable correlation can be more inefficient and biased relative than independence GEE. Those assumptions can be rather strong, too. However, when those assumptions are met, we can get more efficient inference with the exchangeable.

Sensitivity analyses: investigating the assumption of linearity between noise level and log odds of awakening

When we include a continuous variable as a covariate in a regression model, it's important that we include it using the correct (or something approximately correct) functional form. For example, with a continuous outcome Y and continuous covariate X, it may be the case that the expected value of Y is a linear function of X and X^2, rather than a linear function of X. For linear regression there are a number of ways of assessing what the appropriate functional form is for a covariate. A simple but often effective approach is simply to look at a scatter plot of Y against X, to visually assess the shape of the association.

With a binary outcome which we typically model using logistic regression things are not quite as easy (at least when trying to use graphical methods). For a start, the scatter plot of Y against X is now entirely uninformative about the shape of the association between Y and X, and hence how X should be include in the logistic regression model.

```
plot(x = Total$Noise_Level, y = Total$Awakening)
```

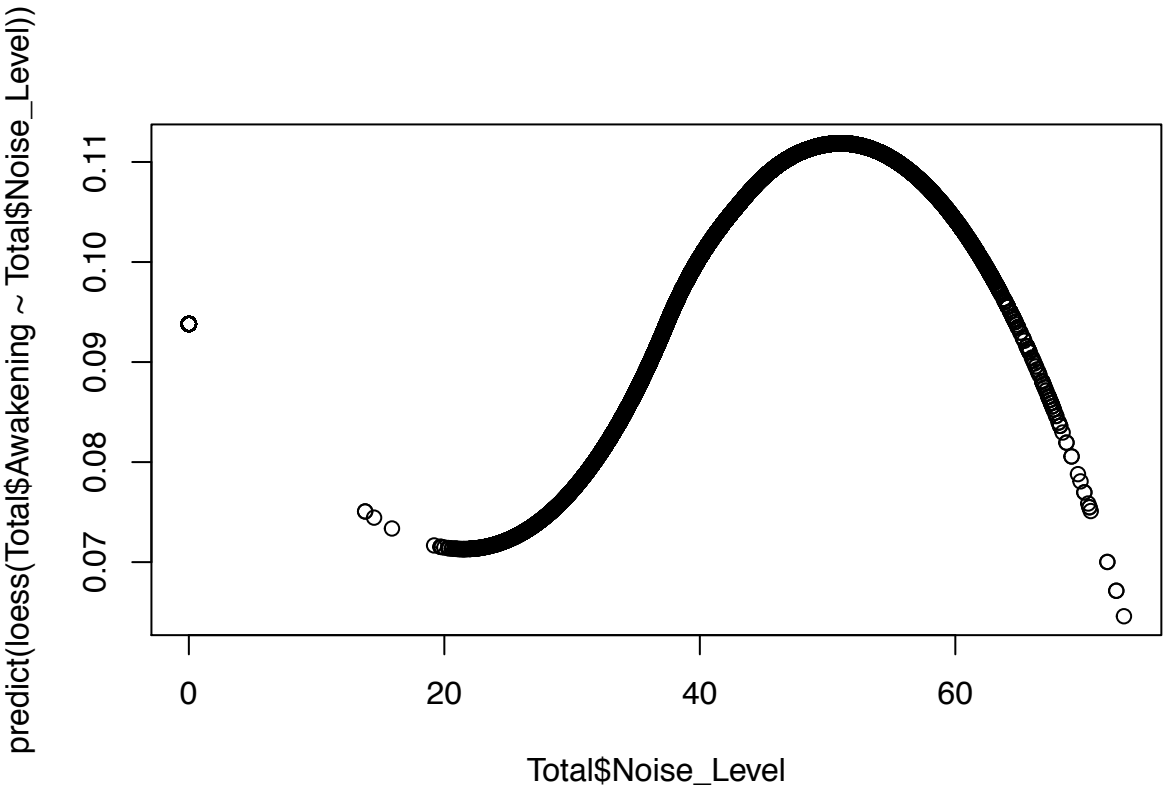


The above plot is uninformative regarding how Y depends on X, due to the binary nature of Y.

One approach to overcome this problem is rather than plotting individual (Y,X) values, to plot a smoothed line of how the average value of Y changes with X. The simplest type of smoother is a running mean, where at a given value X=x, the line is equal to the mean (possibly weighted somehow) of the Y values. The mean values at each value of X are then joined up to give a smoothed line. The amount of smoothing is controlled by the width of the window used in the averaging, or how quickly (in X) the weighting drops to zero.

The $*loess/lowess*$ is a slightly more complicated version, where instead of calculating a (possibly weighted) mean of the Y values in a neighbourhood of X=x, we fit a regression line (e.g. linear) to the data around X=x. By doing this, we assume that locally the Y-X association is linear, but without assuming that it is globally linear. An advantage of this over taking a simple mean is that we need less data to obtain a good estimate of how Y depends on X. Whereas with the running mean we may calculate a weighted mean, where the weight of an observation is higher the closer its value of X is to x, with loess we fit the local linear model using weighted least square, giving less weight to observations with X values further away from X=x.

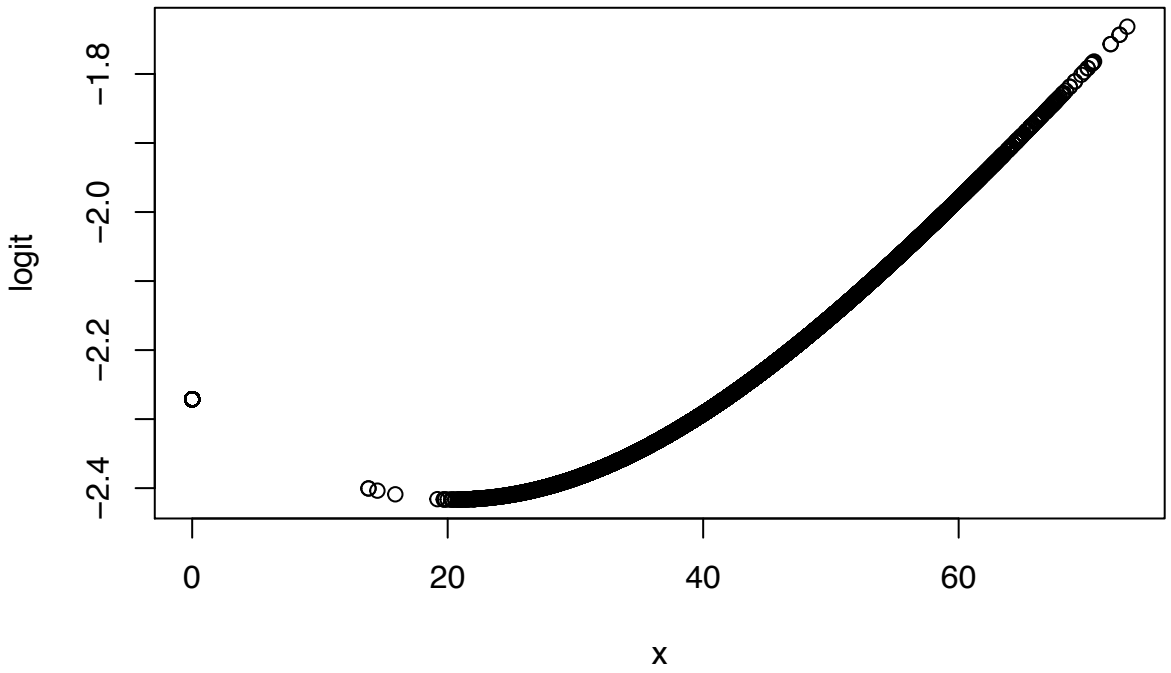
```
plot(Total$Noise_Level,predict(loess(Total$Awakening~Total$Noise_Level)))
```



This plot suggests that the mean of Y is not linear in X, but is perhaps quadratic. The logit function is actually very close to linear for probabilities that are not close to zero or one, and in datasets where the probabilities are not close to zero or one, this is less of an issue.

We can overcome this by plotting the logit of the estimated probabilities (mean of Y) which loess is calculating for us.

```
logitloess <- function(x, y, s) {

logit <- function(pr) {
    log(pr/(1-pr))
}

if (missing(s)) {
    locspan <- 3
} else {
    locspan <- s
}

loessfit <- predict(loess(y~x,span=locspan))
pi <- pmax(pmin(loessfit,0.9999),0.0001)
logitfitted <- logit(pi)

plot(x, logitfitted, ylab="logit")

}

logitloess(Total$Noise_Level,Total$Awakening)
```



We now have a plot that looks a lot more linear. Because the loess does not assume any particular functional form, there will always be some noise in the estimated regression line. The amount of smoothing can be controlled in loess using the span argument, and in the logitloess function we have just defined, this can be controlled using the third argument s. A further thing to note is that the estimated logit will be much more imprecise in regions where there are few X values.

Dicussion

In conclusion, we found that there is a significance relationship between aircraft noise level and people's awakening probability. As the result of the fitted random intercept model, we can tell that controlling the background noise, 1 db increase in the aircraft noise, the odds of awakening will increase by $exp(0.00036 * BackgroundNoise - 0.00895)$.

However, the current state of the art of predicting aircraft noise-induced awakenings on the basis of absolute indoor sound exposure levels leaves much to be desired. Over a very wide range of common indoor sound exposure levels, dosage-response relationships typically predict very small probabilities of awakening. Most of these predictions do not differ significantly from zero, nor from one another. The uncertainty of the predictions is difficult to estimate. Besides that, intruding noises with which sleepers are familiar only rarely awaken them and are tolerated at levels far higher than those with which they are unfamiliar. Residential populations appear to self-select for tolerance to nighttime aircraft noise, so that community-wide behavioral awakening rates vary directly with median noise levels of nighttime aircraft operations. Improved efforts to predict noise-induced awakenings must explicitly address their strong dependence on habituation.

References

LS. (2010) Michaud DS (2007) Berglund B (1990) Honaker.J and G.King (2010) Holmes (2006) Ishwaran and L.F.James. (2001) Kropko and J.Hill. (2014) Reiter and S.Kidneey. (2006) Si and J.P.Reiter (2014)

Berglund B, Nordin S., Lindvall T. 1990. "Adverse Effects of Aircraft Noise." *Environment International.*, no.

54(16): 44–50.

Holmes, and L.Held, C.C. 2006. "Bayesian Auxiliary Variable Models for Binary and Multinomial Regression." *Bayesian Analysis*, no. 1(1).

Honaker.J, and G.King. 2010. "What to Do About Missing Values in Time-Series Cross-Section Data." *American Journal of Political Science*, no. 54(2).

Ishwaran, H., and L.F.James. 2001. "Gibbs Sampling for Stick-Breaking Priors." *Journal of the American Statistical Association*, no. 96:161-73.

Kropko, B.Goodrich, J., and J.Hill. 2014. "Multiple Imputation for Continuous and Categorical Data: Comparing Joint Multivariate Normal and Conditional Approaches." *Political Analysis*.

LS., Finegold. 2010. "Sleep Disturbance Due to Aircraft Noise Exposure." *Noise & Health.*, no. 47(12): 88–94.

Michaud DS, Pearsons K, Fidell S. 2007. "Review of Field Studies of Aircraft Noise-Induced Sleep Disturbance." *J Acoust Soc Am.*, no. 50(12): 32–34.

Reiter, T.E.Raghunathan, J.P., and S.Kidneey. 2006. "The Importance of Modeling the Sampling Design in Multiple Imputation for Missing Data." *Survey Methodology*, no. 32(2).

Si, Y., and J.P.Reiter. 2014. "Nonparametric Bayesian Multiple Imputation for Incomplete Categorical Variables in Largescale Assessment Surveys." *Journal of Educational and Behavioral Statistics*, no. 38(5): 499–521.