Data analysis on $PM_{2.5}$ exposure factors with environmental and survey data

Presenter: Dr. Ming-Chien Mark Tsou

Research Center for Environmental Changes

Academia Sinica

Outline

Part 1:
PM data
processing

Part 2:
Time-
activity
diary (TAD)
data
processing

Part 3:
Multiple
linear
regression
for exposure
factors

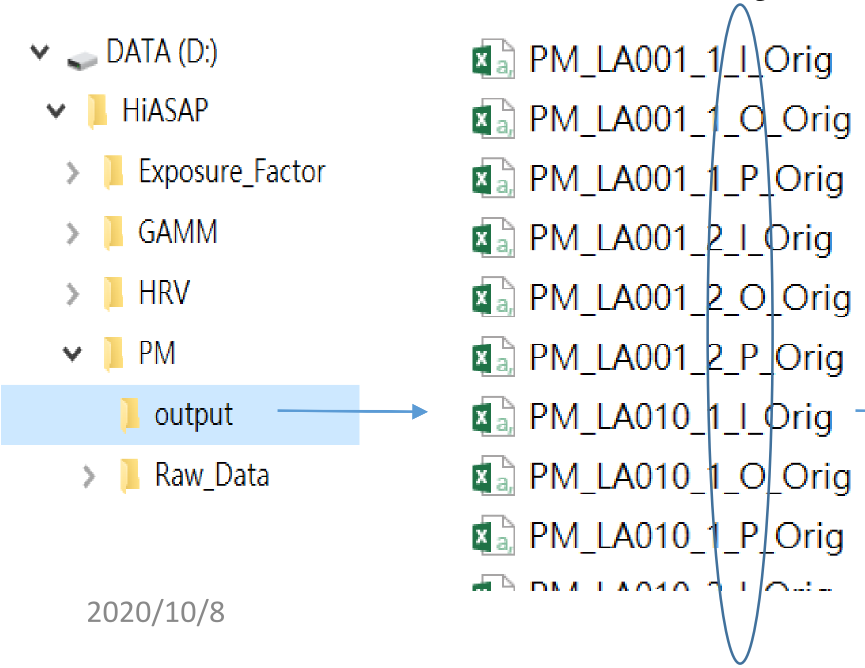
Objectives

- To evaluate important factors of personal $PM_{2.5}$ exposure during at-home period
  - Outdoor $PM_{2.5}$ concentration
  - Household exposure sources
  - Ventilations

Part 1:
PM data processing

PM data processing

- From Line 1 to 63
- The same procedures as the PM data processing for GAMM
- To export the PM data obtained from AS-LUNG-O, AS-LUNG-I and AS-LUNG-P for each subject



	A	B	C	D	E	F	G	H	I	J	K
1	Season	AL_Type	datatime	date	time	sht_t	sht_h	co2	cPM1	cPM2.5	
2	0	2	2018/10/8 00:00	2018/10/8	00:00:00	26.7	71.4	420	14.307	15.182	
3	0	2	2018/10/8 00:00	2018/10/8	00:00:15	26.7	71.4	419	13.708	14.278	
4	0	2	2018/10/8 00:00	2018/10/8	00:00:30	26.7	71.3	417	13.708	14.278	
5	0	2	2018/10/8 00:00	2018/10/8	00:00:45	26.8	71.3	417	13.708	13.826	
6	0	2	2018/10/8 00:01	2018/10/8	00:01:00	26.8	71.2	415	13.708	13.826	
7	0	2	2018/10/8 00:01	2018/10/8	00:01:15	26.8	71.2	414	13.708	13.826	
8	0	2	2018/10/8 00:01	2018/10/8	00:01:30	26.8	71.2	414	13.708	13.826	
9	0	2	2018/10/8 00:01	2018/10/8	00:01:45	26.8	71.2	414	13.109	13.826	
10	0	2	2018/10/8 00:02	2018/10/8	00:02:00	26.8	71.1	414	13.109	13.826	
11	0	2	2018/10/8 00:02	2018/10/8	00:02:15	26.8	71	414	13.109	13.826	

1. To get the data time directly from AS-LUNG data for creating the 5-min intervals

```r
# To calculate 5-min average PM data for exposure factor (based on ASLUNG time)
library('lubridate')
date_1<-c(ymd(as.character(ASLUNGt$date)))
time<-substr(ASLUNGt$time,1,5)
date_2<-paste(date_1,time)
mm<-c(as.numeric(substr(date_2,16,16)))
mm5<-c()
for (l in 1:length(mm)) {
    if(mm[l]<5){
        mm5[l]<-"0~4"
    }else{
        mm5[l]<-"5~9"
    }
}
date_3 <- paste0(substr(date_2,1,15),mm5)
date_4<-as.data.frame(table(date_3))
```

	date_3	Freq
1	2018-10-08 00:00~4	20
2	2018-10-08 00:05~9	20
3	2018-10-08 00:10~4	20
4	2018-10-08 00:15~9	20
5	2018-10-08 00:20~4	20
6	2018-10-08 00:25~9	20
7	2018-10-08 00:30~4	20
8	2018-10-08 00:35~9	20
9	2018-10-08 00:40~4	20
10	2018-10-08 00:45~9	20
11	2018-10-08 00:50~4	20
12	2018-10-08 00:55~9	20

There are 20 data between 00:00 to 04:59

2. To calculate the 5-min average PM data

```r
ALFinal<-data.frame()
for(j in 1:dim(date_4)[1]){
    ALFinal[j,1]<-date_4[j,1]
    ALFinal[j,2]<-substr(bb[k],10,14)
    ALFinal[j,3]<-mean(as.numeric(ASLUNGt[which(date_3==date_4[j,1]),1]),na.rm = TRUE) # 0=ho
    ALFinal[j,4]<-mean(as.numeric(ASLUNGt[which(date_3==date_4[j,1]),2]),na.rm = TRUE) # 1=ou
    ALFinal[j,5]<-mean(as.numeric(ASLUNGt[which(date_3==date_4[j,1]),6]),na.rm = TRUE)
    ALFinal[j,6]<-mean(as.numeric(ASLUNGt[which(date_3==date_4[j,1]),7]),na.rm = TRUE)
    ALFinal[j,7]<-mean(as.numeric(ASLUNGt[which(date_3==date_4[j,1]),9]),na.rm = TRUE)
    ALFinal[j,8]<-mean(as.numeric(ASLUNGt[which(date_3==date_4[j,1]),10]),na.rm = TRUE)
    ALFinal[j,9]<-mean(as.numeric(ASLUNGt[which(date_3==date_4[j,1]),8]),na.rm = TRUE)
    ALFinal[j,10]<-date_4[j,2]
}
colnames(ALFinal)<-c("Date","S_no","Season","AL_Type","TEM","HUM","PM1","PM2.5","CO2","Freq")
```

To create a variable for the number of data in this 5-min interval

3. To exclude the time with insufficient data (half of default number of data)

```r
    # To exclude the 5-min intervals which contained less than half of expected number of data
    if(substr(aa1[p],9,9)=="O"){
        ALFinal2 <- ALFinal[which(ALFinal$Freq>=3),]
    }else{
        ALFinal2 <- ALFinal[which(ALFinal$Freq>=10),]
    }
```

	The default number of data in each 5-min interval
AS-LUNG-O	1x5=5
AS-LUNG-I	4x5=20
AS-LUNG-P	4x5=20

```r
    ALFinal2 <- data.frame(subset(ALFinal2,select=c(Date,S_no,Season,AL_Type,TEM,HUM,PM1,PM2.5,CO2)))
    outputname<-paste0("PM_",substr(bb[k],10,14),"_",Season[i]+1,"_",substr(aa1[p],9,9),"_5 min_ASLUNG_Time.csv")
    write.csv(ALFinal2,outputname,row.names=FALSE,na="")
}
```

4. To export the dataset for 5-min average data for each subject

PM_LA010_1_I_5 min_ASLUNG_Time
PM_LA014_1_P_5 min_ASLUNG_Time
PM_LA001_1_O_5 min_ASLUNG_Time
PM_LA021_1_O_5 min_ASLUNG_Time
PM_LA010_1_O_5 min_ASLUNG_Time

5-min average PM data combination for all subjects

- From Line 108 to 159
- The same procedures as the PM data processing for GAMM from Line 133 to 184
- To export the final dataset for PM data for exposure factors evaluation

1	Date	Year	Month	Day	Hour	Minute	Minute_3(	S_no	Season	AL_Type	TEM	HUM	PM1	PM2.5	CO2
2	2018/10/8 13:11	2018	10	8	13	11	1	LA001	0	3	26.305	78.895	2.95625	3.3784	449.95
3	2018/10/8 13:16	2018	10	8	13	16	1	LA001	0	3	26.71	78.17	2.9818	3.4876	448.05
4	2018/10/8 13:21	2018	10	8	13	21	1	LA001	0	3	27.01	77.315	3.1351	3.6332	440.65
5	2018/10/8 13:26	2018	10	8	13	26	1	LA001	0	3	27.165	76.05	3.1862	3.6332	428.65
6	2018/10/8 13:31	2018	10	8	13	31	2	LA001	0	3	27.38	75.31	3.31395	3.797	426.5
7	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Part 2:
Time-activity diary (TAD) data processing

TAD raw data

Locations

Ventilations

Activities

Sources

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	S_no	Year	Month	Day	Date	WD	Hour	Minute_3(	Place1	Place2	Ventilatio	Ventilatio	Activity1	Activity2	AP1	AP2	AP3	Sensor	
2	LA001	2018	10	8	1	1	13	1	1	0	2	0	10	0	0	0	0	0	
3	LA001	2018	10	8	1	1	13	2	1	0	2	0	10	0	0	0	0	1	
4	LA001	2018	10	8	1	1	14	1	3	0	3	0	3	0	0	0	0	1	
5	LA001	2018	10	8	1	1	14	2	3	0	3	0	3	0	0	0	0	1	
6	LA001	2018	10	8	1	1	15	1	3	0	3	0	3	0	0	0	0	1	
7	LA001	2018	10	8	1	1	15	2	3	0	3	0	3	0	0	0	0	1	
8	LA001	2018	10	8	1	1	16	1	3	0	3	0	3	0	0	0	0	1	
9	LA001	2018	10	8	1	1	16	2	3	0	3	0	3	0	0	0	0	1	
10	LA001	2018	10	8	1	1	17	1	3	0	3	0	3	0	0	0	0	1	
11	LA001	2018	10	8	1	1	17	2	3	0	3	0	3	0	0	0	0	1	
12	LA001	2018	10	8	1	1	18	1	3	0	3	0	3	0	0	0	0	1	
13	LA001	2018	10	8	1	1	18	2	3	0	3	0	3	0	0	0	0	1	
14	LA001	2018	10	8	1	1	19	1	3	0	3	0	3	0	0	0	0	1	
15	LA001	2018	10	8	1	1	19	2	16	0	9	0	2	0	0	0	0	1	
16	LA001	2018	10	8	1	1	20	1	1	0	3	0	8	4	2	0	0	1	
17	LA001	2018	10	8	1	1	20	2	1	0	3	0	10	0	0	0	0	1	
18	LA001	2018	10	8	1	1	21	1	1	0	3	0	9	0	0	0	0	1	

TAD data processing

```r
##TAD data processing
TAD <- read.csv(paste0(way,"/TAD_Raw.csv"))
```

1. To read the TAD data

2. To create dummy variables by categorizing different locations, ventilations, activities or sources into different groups

```r
Loc_Home <- c()
Loc_In <- c()
Loc_Out <- c()
Loc_Trans_In <- c()
Loc_Trans_Out <- c()
Loc_Trans <- c()
Loc_In_All <- c()
Loc_Out_All <- c()
Vent_Closed <- c()
Vent_Opened <- c()
Vent_AC <- c()
Act_Sleep <- c()
Act_Commute <- c()
Act_Work <- c()
Act_Cook <- c()
Act_Worship <- c()
Act_Shopping <- c()
Act_Exercise <- c()
Act_Eat <- c()
Act_Bath <- c()
Act_Sedentary <- c()
Act_Housework <- c()
Act_Other <- c()
S_Exhaust <- c()
S_Cooking <- c()
S_ETS <- c()
S_Dust <- c()
S_Incense <- c()
S_MosquitoCoil <- c()
S_Aromatic <- c()
S_Josspaper <- c()
S_OpenedBurning <- c()
S_Factory <- c()
S_Garbage <- c()
S_Other <- c()
S_Incense_JPaper <- c()
```

```r
for(i in 1:dim(TAD)[1]){
    ifelse(TAD$Place1[i]==1 | TAD$Place2[i]==1,Loc_Home[i]<-1,Loc_Home[i]<-0)
    ifelse(TAD$Place1[i]==1 | TAD$Place1[i]==2 | TAD$Place1[i]==3 | TAD$Place1[i]==4 | TAD$Place1[i]==5 | TAD$Place1[i]==6 | TAD$Place1[i]==7 | TAD$Plac
    ifelse(TAD$Place1[i]==17 | TAD$Place1[i]==18 | TAD$Place1[i]==19 | TAD$Place1[i]==20 | TAD$Place1[i]==21 | TAD$Place1[i]==22 | TAD$Place1[i]==23 |
    ifelse(TAD$Place1[i]==11 | TAD$Place1[i]==12 | TAD$Place1[i]==13 | TAD$Place1[i]==14 | TAD$Place2[i]==11 | TAD$Place2[i]==12 | TAD$Place2[i]==13 |
    ifelse(TAD$Place1[i]==16 | TAD$Place1[i]==9 | TAD$Place1[i]==10 | TAD$Place1[i]==15 | TAD$Place1[i]==62 | TAD$Place2[i]==16 | TAD$Place2[i]==9 | TAD
    ifelse(TAD$Place1[i]==11 | TAD$Place1[i]==12 | TAD$Place1[i]==13 | TAD$Place1[i]==14 | TAD$Place1[i]==9 | TAD$Place1[i]==10 | TAD$Place1[i]==15 | TA
    ifelse(TAD$Place1[i]==1 | TAD$Place1[i]==2 | TAD$Place1[i]==3 | TAD$Place1[i]==4 | TAD$Place1[i]==5 | TAD$Place1[i]==6 | TAD$Place1[i]==7 | TAD$Plac
    ifelse(TAD$Place1[i]==16 | TAD$Place1[i]==17 | TAD$Place1[i]==18 | TAD$Place1[i]==19 | TAD$Place1[i]==20 | TAD$Place1[i]==21 | TAD$Place1[i]==22 |
    ifelse(TAD$Ventilation1[i]==1 | TAD$Ventilation2[i]==1,Vent_Closed[i]<-1,Vent_Closed[i]<-0)
    ifelse(TAD$Ventilation1[i]==2 | TAD$Ventilation1[i]==3 | TAD$Ventilation2[i]==2 | TAD$Ventilation2[i]==3,Vent_Opened[i]<-1,Vent_Opened[i]<-0)
    ifelse(TAD$Ventilation1[i]==4 | TAD$Ventilation1[i]==5 | TAD$Ventilation2[i]==4 | TAD$Ventilation2[i]==5,Vent_AC[i]<-1,Vent_AC[i]<-0)
    ifelse(TAD$Activity1[i]==1 | TAD$Activity2[i]==1,Act_Sleep[i]<-1,Act_Sleep[i]<-0)
    ifelse(TAD$Activity1[i]==2 | TAD$Activity2[i]==2,Act_Commute[i]<-1,Act_Commute[i]<-0)
    ifelse(TAD$Activity1[i]==3 | TAD$Activity2[i]==3,Act_Work[i]<-1,Act_Work[i]<-0)
    ifelse(TAD$Activity1[i]==4 | TAD$Activity2[i]==4,Act_Cook[i]<-1,Act_Cook[i]<-0)
    ifelse(TAD$Activity1[i]==5 | TAD$Activity2[i]==5,Act_Worship[i]<-1,Act_Worship[i]<-0)
    ifelse(TAD$Activity1[i]==6 | TAD$Activity2[i]==6,Act_Shopping[i]<-1,Act_Shopping[i]<-0)
    ifelse(TAD$Activity1[i]==7 | TAD$Activity2[i]==7,Act_Exercise[i]<-1,Act_Exercise[i]<-0)
    ifelse(TAD$Activity1[i]==8 | TAD$Activity2[i]==8,Act_Eat[i]<-1,Act_Eat[i]<-0)
    ifelse(TAD$Activity1[i]==9 | TAD$Activity2[i]==9,Act_Bath[i]<-1,Act_Bath[i]<-0)
    ifelse(TAD$Activity1[i]==10 | TAD$Activity2[i]==10,Act_Sedentary[i]<-1,Act_Sedentary[i]<-0)
    ifelse(TAD$Activity1[i]==11 | TAD$Activity2[i]==11,Act_Housework[i]<-1,Act_Housework[i]<-0)
    ifelse(TAD$Activity1[i]==64 | TAD$Activity2[i]==64,Act_Other[i]<-1,Act_Other[i]<-0)
    ifelse(TAD$AP1[i]==1 | TAD$AP2[i]==1 | TAD$AP3[i]==1,S_Exhaust[i]<-1,S_Exhaust[i]<-0)
    ifelse(TAD$AP1[i]==2 | TAD$AP2[i]==2 | TAD$AP3[i]==2,S_Cooking[i]<-1,S_Cooking[i]<-0)
    ifelse(TAD$AP1[i]==3 | TAD$AP2[i]==3 | TAD$AP3[i]==3,S_ETS[i]<-1,S_ETS[i]<-0)
    ifelse(TAD$AP1[i]==4 | TAD$AP2[i]==4 | TAD$AP3[i]==4,S_Dust[i]<-1,S_Dust[i]<-0)
    ifelse(TAD$AP1[i]==5 | TAD$AP2[i]==5 | TAD$AP3[i]==5,S_Incense[i]<-1,S_Incense[i]<-0)
    ifelse(TAD$AP1[i]==6 | TAD$AP2[i]==6 | TAD$AP3[i]==6,S_MosquitoCoil[i]<-1,S_MosquitoCoil[i]<-0)
    ifelse(TAD$AP1[i]==7 | TAD$AP2[i]==7 | TAD$AP3[i]==7,S_Aromatic[i]<-1,S_Aromatic[i]<-0)
    ifelse(TAD$AP1[i]==8 | TAD$AP2[i]==8 | TAD$AP3[i]==8,S_Josspaper[i]<-1,S_Josspaper[i]<-0)
    ifelse(TAD$AP1[i]==9 | TAD$AP2[i]==9 | TAD$AP3[i]==9,S_OpenedBurning[i]<-1,S_OpenedBurning[i]<-0)
    ifelse(TAD$AP1[i]==10 | TAD$AP2[i]==10 | TAD$AP3[i]==10,S_Factory[i]<-1,S_Factory[i]<-0)
    ifelse(TAD$AP1[i]==11 | TAD$AP2[i]==11 | TAD$AP3[i]==11,S_Garbage[i]<-1,S_Garbage[i]<-0)
    ifelse(TAD$AP1[i]==65 | TAD$AP2[i]==65 | TAD$AP3[i]==65,S_Other[i]<-1,S_Other[i]<-0)
    ifelse(TAD$AP1[i]==5 | TAD$AP2[i]==5 | TAD$AP3[i]==5 | TAD$AP1[i]==8 | TAD$AP2[i]==8 | TAD$AP3[i]==8,S_Incense_JPaper[i]<-1,S_Incense_JPaper[i]<-0)
```

Dummy variables

At home	Loc_Home (the dummy variable)
Yes	1
No	0

- For example:

```r
Loc_Home <- c()
```

```r
for(i in 1:dim(TAD)[1]){
    ifelse(TAD$Place1[i]==1 | TAD$Place2[i]==1,Loc_Home[i]<-1,Loc_Home[i]<-0)
```

A. Location: Please select where you stay most of the time in this specific period

[Indoor] Space with ceiling, wall or door

1. Home
2. Office, classroom, cram school, etc.
3. Factory (repair car garage, furnishing

[Outdoor] hall, road, outdoors, open ground

16. On the road(walking on the road, waiting for bus or traffic light)
17. Traditional market

```r
S_Cooking <- c()
```

```r
    ifelse(TAD$AP1[i]==2 | TAD$AP2[i]==2 | TAD$AP3[i]==2,S_Cooking[i]<-1,S_Cooking[i]<-0)
```

D. Air quality: Please choose at most two items you have smelled.

1. Exhaust gas from cars or motorcycles.
2. Smoke of lampblack from homes, restaurants and peddlers.
3. Smell from smoking or second-hand smoke.
4. Sweep dust.

Source of cooking	S_Cooking (the dummy variable)
Yes	1
No	0

Variables for TAD data

You can create variables according your requirements

Location (microenvironments)		Ventilation	Activities		Sources	
Home	Indoor environments (including transportation)	Window open	Sleeping	Taking bath	Vehicle exhaust	Open burning
Indoor environments		Window closed	Commuting	Sedentary activities	Cooking	Factories
Outdoor environments	Outdoor environments (including transportation)	AC-on	Working	Doing housework	ETS	Garbage
Transportation (indoor)			Cooking	Other	Cleaning/dust	Other
Transportation (outdoor)			Worshipping		Burning of incense	Burning of incense/joss-paper
Transportation (all)			Shopping		Burning of mosquito coil	
			Doing exercise		Aromatic products	
			Eating		Burning of joss-paper	

3. To combine all dummy variables created in Step 2 with subjects' ID and data time

```
TADr <- data.frame(subset(TAD,select=c(S_no,Year,Month,Day,Date,WD,Hour,Minute_30)),Loc_Home,Loc_In,Loc_Out,Loc_Trans_In,Loc_Trans_Out,Loc_Trans,
```

4. To group the sources with less frequency (cleaning/dust, burning of mosquito coil, aromatic products, open burning, factories, garbage and other) into "Other" group.

```
S_Other2<-c()
for(i in 1:dim(TADr)[1]){
    if(TADr$S_Dust[i]==1|TADr$S_MosquitoCoil[i]==1|TADr$S_Aromatic[i]==1|TADr$S_OpenedBurning[i]==1|TADr$S_Factory[i]==1|TADr$S_Garbage[i]==1|TAD
        S_Other2[i]<-1
    }else{
        S_Other2[i]<-0
    }
}
```

5. To create a variable for the time with more than one source

```r
S_Multiple<-c()
for(i in 1:dim(TADr)[1]){
    if(TADr$S_Exhaust[i]+TADr$S_Cooking[i]+TADr$S_ETS[i]+TADr$S_Incense_JPaper[i]+S_Other2[i]<=1){
        S_Multiple[i]<-0
    }else{
        S_Multiple[i]<-1
    }
}
```

6. To create a variable for the time with no recorded source

```r
S_None<-c()
for(i in 1:dim(TADr)[1]){
    if(TADr$S_Exhaust[i]==0&TADr$S_Cooking[i]==0&TADr$S_ETS[i]==0&TADr$S_Incense_JPaper[i]==0&S_Other2[i]==0){
        S_None[i]<-1
    }else{
        S_None[i]<-0
    }
}
```

7. To create a dummy variable for ventilations

```
Vent_D3_1<-c()
for(i in 1:dim(TADr)[1]){
    if(TADr$Vent_AC[i]==0 & TADr$Vent_Closed[i]==1 & Vent_Opened[i]==0){
        Vent_D3_1[i]<-1
    }else{
        Vent_D3_1[i]<-0
    }
}

Vent_D3_2<-c()
for(i in 1:dim(TADr)[1]){
    if(TADr$Vent_AC[i]==1 & TADr$Vent_Closed[i]==0 & Vent_Opened[i]==0){
        Vent_D3_2[i]<-1
    }else{
        Vent_D3_2[i]<-0
    }
}
```

	Vent_D3_1	Vent_D3_2
AC-on	0	1
Window closed	1	0
Window open (reference group)	0	0

```r
Season <- c()
for(i in 1:dim(TADr)[1]){
    if(TADr$Year[i]==2018){
        Season[i]<-0
    }else{
        Season[i]<-1
    }
}

TADr <- data.frame(TADr,S_Other2,S_Multiple,S_None,Vent_D3_1,Vent_D3_2)

write.csv(TADr,file="2020_Training_Course_TAD.csv",row.names=FALSE)
```

7. To create a "Season" variable

8. To combine variables created with TAD data

9. To export the final TAD dataset

1	S_no	Year	Month	Day	Date	WD	Hour	Minute_3(	Loc_Hom	Loc_In	Loc_Out	Loc_Trans	Loc_Trans	Loc_T
2	LA001	2018	10	8	1	1	13	1	1	1	0	0	0	
3	LA001	2018	10	8	1	1	13	2	1	1	0	0	0	
4	LA001	2018	10	8	1	1	14	1	0	1	0	0	0	
5	LA001	2018	10	8	1	1	14	2	0	1	0	0	0	

Part 3: Multiple linear regression for exposure factors

You can modify by yourself

```r
# To remove previous memory in R.
rm(list=ls())

location <- "D:/"

# To set the output file
cmd1 <- paste0("setwd('",location,"HiASAP/Exposure_Factor/output')")
eval(parse(text=cmd1))

outputname <- "Exposure_Factor"
```

1. To remove all objects from current workspace

2. To set the default drive of your data

3. To set the path and filename of output file

4. To create variables for personal PM data

```r
# To create variables of outdoor, indoor and personal data of ASLUNG
PM <- read.csv(paste0(location,"HiASAP/PM/output/PM_5 min_ASLUNG_Time_All.csv"))
ALP <- PM[which(PM$AL_Type==3),]
colnames(ALP)[names(ALP) == c("TEM","HUM","PM1","PM2.5","CO2")]<-c("ALP_TEM","ALP_HUM","ALP_PM1","ALP_PM2.5","ALP_CO2")
ALP_2 <- ALP[,c(1:9,11:15)]
```

5. To create variables for indoor PM data

```r
ALI <- PM[which(PM$AL_Type==2),]
colnames(ALI)[names(ALI) == c("TEM","HUM","PM1","PM2.5","CO2")]<-c("ALI_TEM","ALI_HUM","ALI_PM1","ALI_PM2.5","ALI_CO2")
ALI_2 <- ALI[,c(2:6,8,11:15)]
```

6. To create variables for outdoor PM data

```r
ALO <- PM[which(PM$AL_Type==1),]
colnames(ALO)[names(ALO) == c("TEM","HUM","PM1","PM2.5","CO2")]<-c("ALO_TEM","ALO_HUM","ALO_PM1","ALO_PM2.5","ALO_CO2")
ALO_2 <- ALO[,c(2:6,8,11:15)]
```

7. To merge (1) personal PM, (2) indoor PM , (3) outdoor PM , (4) TAD and (5) meteorological data for multiple linear regression

(1) Personal PM data

(2) Indoor PM data

(3) Outdoor PM data

(4) TAD data

(5) Meteorological data

```r
# To combine data
PMall <- data.frame()
PMall <- merge(ALP_2,ALI_2, by=c("Year","Month","Day","Month","Hour","Minute","S_no"))
PMall <- merge(PMall,ALO_2, by=c("Year","Month","Day","Month","Hour","Minute","S_no"))

PMall_2 <- data.frame()
TAD <- read.csv(paste0(location,"HiASAP/Questionnaire_TAD/output/2020_Training_Course_TAD.csv"))
PMall_2 <- merge(PMall,TAD, by=c("Year","Month","Day","Month","Hour","Minute_30","S_no"))

Meteor <- read.csv(paste0(location,"HiASAP/Meteor/Meteor_Hourly_All.csv"))
Meteor <- Meteor[,c(1:4,14)]
PMfinal <- merge(PMall_2,Meteor, by=c("Year","Month","Day","Month","Hour"))

write.csv(PMfinal,file=paste0(outputname,".csv"),row.names=FALSE)
```

8. To select data at no-raining, awake and at-home period

```r
# To select no-raining, awake and at-home period
PMfinal_A_NR_Home <- PMfinal[which(PMfinal$Precp==0 & PMfinal$Act_Sleep==0 & PMfinal$Loc_Home==1),]

# To determine a final regression model by the stepwise regression method
# To remain the variables which you are interested
aa <- PMfinal_A_NR_Home[,c(9,13,23,50,51,52,62,63,66,67)]
```

9. To select variables you want to add to the model

- In this case, we select outdoor $PM_{2.5}$ concentration, source of vehicle exhaust, source of cooking, source of ETS, source of burning incense/joss-paper, other sources and dummy variables of ventilation to the models

Stepwise regression method

- Forward stepwise selection
- Backward stepwise selection
- Both forward and backward selection

Stepwise regression method

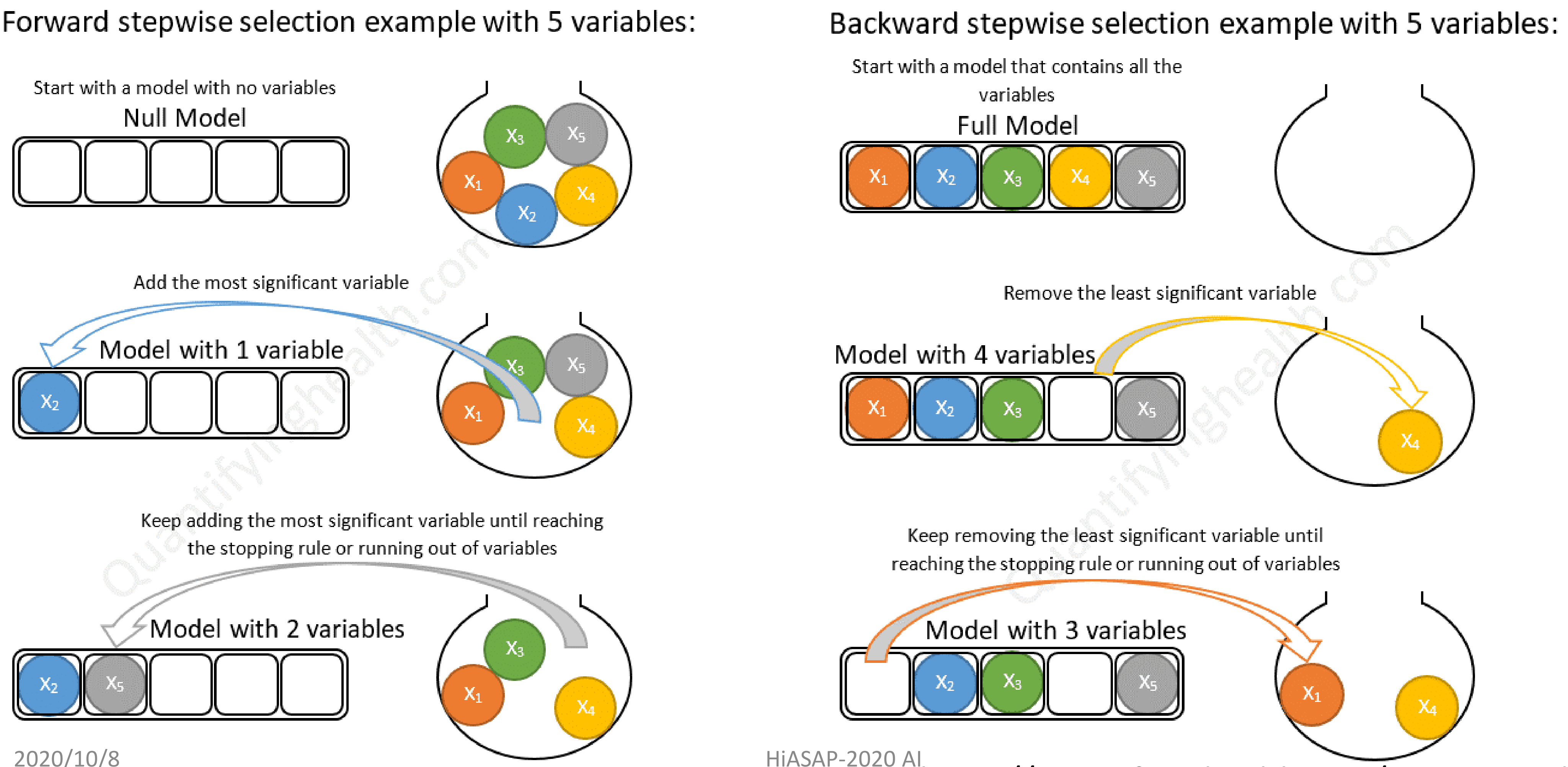


Figure source: https://quantifyinghealth.com/stepwise-selection/

Multiple linear regression with stepwise regression method

10. To use the stepwise regression method to select the most important variables

```r
# To use the stepwise regression method to identify and select the most useful explanatory variables from a list of
# several plausible independent variables
Final_Model <- step(lm(ALP_PM2.5~.,data=aa),direction="both")
summary(Final_Model)
```

- Here “ALP_PM2.5~.” means all variates except personal $PM_{2.5}$ concentration are added to the model

AIC: a mathematical method for evaluating how well a model fits the data it was generated from

Results of stepwise regression method

```
Step:  AIC=16315.58
ALP_PM2.5 ~ Season + ALO_PM2.5 + S_Exhaust + S_Cooking + S_Incense_JPaper +
    S_Other2 + Vent_D3_1 + Vent_D3_2

                 Df Sum of Sq     RSS   AIC
- S_Exhaust       1       2.1 1467950 16314
- S_Cooking       1      72.8 1468020 16314
- Vent_D3_2       1     651.8 1468599 16315
<none>                        1467948 16316
- S_Other2        1    1967.8 1469915 16317
- S_Incense_JPaper 1   3697.4 1471645 16320
- Season          1    5321.9 1473269 16323
- Vent_D3_1       1    6632.8 1474580 16325
- ALO_PM2.5       1   27556.5 1495504 16361
```

① Model 1 (Full model)

```
Step:  AIC=16313.59
ALP_PM2.5 ~ Season + ALO_PM2.5 + S_Cooking + S_Incense_JPaper +
    S_Other2 + Vent_D3_1 + Vent_D3_2

                   Df Sum of Sq     RSS   AIC
- S_Cooking         1      72.7 1468022 16312
- Vent_D3_2         1     651.0 1468601 16313
<none>                          1467950 16314
- S_Other2          1    1972.2 1469922 16315
+ S_Exhaust         1       2.1 1467948 16316
- S_Incense_JPaper  1    3705.2 1471655 16318
- Season            1    5486.5 1473436 16321
- Vent_D3_1         1    6653.0 1474603 16323
- ALO_PM2.5         1   27614.4 1495564 16359
```

② Model 2

```
Step:  AIC=16311.71
ALP_PM2.5 ~ Season + ALO_PM2.5 + S_Incense_JPaper + S_Other2 +
    Vent_D3_1 + Vent_D3_2

                   Df Sum of Sq     RSS   AIC
- Vent_D3_2         1     652.2 1468675 16311
<none>                          1468022 16312
- S_Other2          1    1954.7 1469977 16313
+ S_Cooking         1      72.7 1467950 16314
+ S_Exhaust         1       2.1 1468020 16314
- S_Incense_JPaper  1    3649.8 1471672 16316
- Season            1    5527.4 1473550 16319
- Vent_D3_1         1    6964.9 1474987 16322
- ALO_PM2.5         1   27811.8 1495834 16358
```

③ Model 3

Final model

```
Step:  AIC=16310.85
ALP_PM2.5 ~ Season + ALO_PM2.5 + S_Incense_JPaper + S_Other2 +
    Vent_D3_1

                   Df Sum of Sq     RSS   AIC
<none>                          1468675 16311
+ Vent_D3_2         1     652.2 1468022 16312
- S_Other2          1    1966.8 1470641 16312
+ S_Cooking         1      73.9 1468601 16313
+ S_Exhaust         1       1.4 1468673 16313
- S_Incense_JPaper  1    3023.0 1471698 16314
- Season            1    5732.4 1474407 16319
- Vent_D3_1         1    6915.9 1475591 16321
- ALO_PM2.5         1   27282.4 1495957 16356
```

smallest

④ Model 4

Results of final model

11. To run the final model for personal $PM_{2.5}$ exposure factors evaluation

```
# In order to explain the set of dummy variables of trinary variable (window open, window closed and AC-on),
# we add the dummy variable of "Vent_D3_2" into the final model
Final_Model_2 <- lm(ALP_PM2.5 ~ ALO_PM2.5+S_Incense_JPaper+S_Other2+Vent_D3_1+Vent_D3_2+Season, data = aa)
summary(Final_Model_2)
```

```
Call:
lm(formula = ALP_PM2.5 ~ ALO_PM2.5 + S_Incense_JPaper + S_Other2 +
    Vent_D3_1 + Vent_D3_2 + Season, data = aa)

Residuals:
   Min     1Q Median     3Q    Max
-55.32  -9.41  -6.92  -0.77 415.84

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       12.45380    1.05795  11.772  < 2e-16 ***
ALO_PM2.5          0.35080    0.05037   6.964 4.18e-12 ***
S_Incense_JPaper   9.41805    3.73313   2.523  0.01170 *
S_Other2          -4.12492    2.23419  -1.846  0.06497 .
Vent_D3_1         -6.78389    1.94656  -3.485  0.00050 ***
Vent_D3_2        -11.19743   10.49940  -1.066  0.28631
Season             3.05936    0.98541   3.105  0.00193 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 23.95 on 2560 degrees of freedom
Multiple R-squared:  0.03389,	Adjusted R-squared:  0.03162
F-statistic: 14.97 on 6 and 2560 DF,  p-value: < 2.2e-16
```

SE (standard error)

β (coefficients)

p value

Partial R^2 estimation

12. To install (for the first time to use) and load the R packages for partial R^2 estimation

```r
# To determine the partial R2 of each independent variable
# install.packages("rsq") (only for first time to use)
library("rsq")

Partial_R2 <- rsq.partial(Final_Model_2)
```

13. To calculate the partial R^2 of each independent variable in the model

```
[1] "Partial R2"
$adjustment
[1] FALSE

$variable
[1] "ALO_PM2.5"        "S_Incense_JPaper" "S_Other2"         "Vent_D3_1"        "Vent_D3_2"        "Season"

$partial.rsq
[1] 0.0185928613 0.0024800368 0.0013297616 0.0047219976 0.0004440941 0.0037510554
```

Results export

```r
# To print out final model results to txt file
sink("Exposure_Factor_Results.txt") # redirect console output to a file
print("Stepwise")
print(step(lm(ALP_PM2.5~.,data=aa),direction="both"))
print("Final model")
print(summary(Final_Model_2))
print("Partial R2")
print(Partial_R2)
sink()  # return output to the terminal
```

14. To export results of the final model

Results of personal $PM_{2.5}$ exposure factors evaluation

	Personal $PM_{2.5}$ exposure (μg/m^3)			
Variables	Coefficient	95% CI	Partial R^2	*p*-value
Outdoor $PM_{2.5}$ concentration (μg/m^3)	0.351	0.252, .450	0.0186	<0.001
Burning of incense/joss-paper	9.42	2.10, 16.7	0.0025	0.012
Other exposure sources	-4.12	-8.50, 0.254	0.0013	0.065
Window-closed	-6.78	-10.6, -2.97	0.0047	<0.001
AC-on	-11.2	-31.8, 9.38	0.0004	0.286
Season	3.06	1.13, 4.99	0.0038	0.002

- Burning of incense/joss-paper had on average the highest 5-min $PM_{2.5}$ increments (9.42 μg/m^3) to personal $PM_{2.5}$ exposure.

Thank you for your attention