Milestone 4 Individual Progress on Analysis

Dawei Wang

Preliminary work: I was in charge of database pretreatment including datab ase searching and splicing, transforming data into a readable format, and make correspondent annotations.

In this stage, I conduct missing data analysis and shared the result to the group and then perform data transformation conduct LASSO regression and PCA and Factor analysis to the dataset.

Missing data analysis:
By substituting all missing values to an enormous number compared to the variable range, and then plot each variable against all the other variables to check the distribution of missing value. 8 variables in our dataset has missing values, and their distribution are random, and no pattern were dis covered over 176 plots. Since there are too many missing values (about 2/3 of total observations), it's unreasonable to find a value to fill the blank. S o, observations with missing values were eliminated.

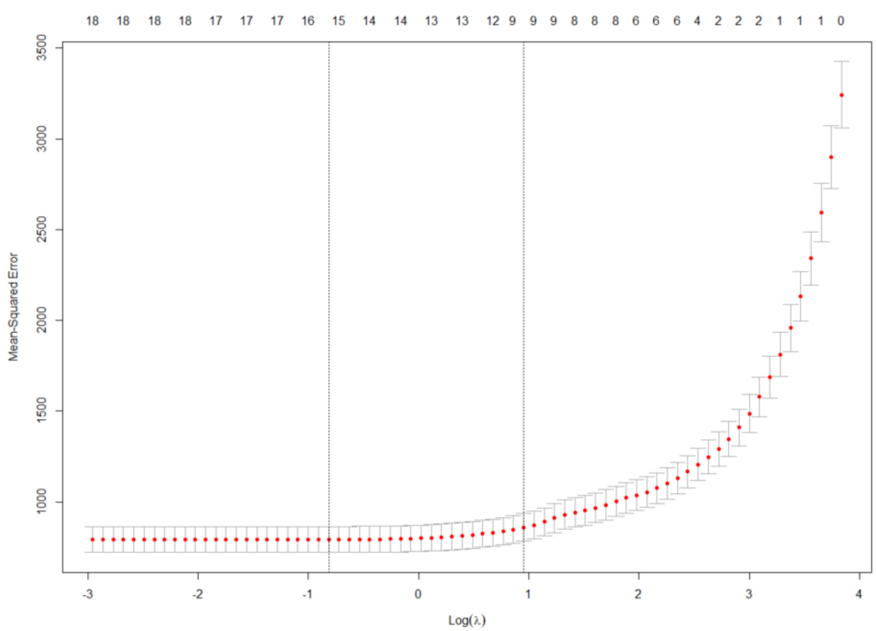
Data transformation:
The variable names were too complex to use in the R command, so all the names were substituted by the new names. The corresponding table listed below:

New name	Original column name
mort	heart_disease_mortality_per_100k
V1	econ__pct_civilian_labor
V2	econ__pct_unemployment
V3	econ__pct_uninsured_adults
V4	econ__pct_uninsured_children
V5	demo__pct_below_18_years_of_age
V6	demo__pct_aged_65_years_and_older
V7	demo__birth_rate_per_1k
V8	demo__death_rate_per_1k
V9	health__pct_adult_obesity

V10	health__pct_adult_smoking
V11	health__pct_diabetes
V12	health__pct_low_birthweight
V13	health__pct_excessive_drinking
V14	health__pct_physical_inacticity
V15	health__air_pollution_particulate_matter
V16	health__homicides_per_100k
V17	health__motor_vehicle_crash_deaths_per_100k
V18	health__pop_per_dentist
V19	health__pop_per_primary_care_physician
V20	Area_Rucc
V21	Econ_Economic_typology
V22	Area_Urban_Influence

In the previous analysis, we find the log and square root transformation could make some of our variables normally distribute. To avoid infinite or NA, here, I did log (Variable +1) for V2, V4, V7, V12, V16, V17, V18, V19 and square root for V8 and V11. The dataset after all these treatments was saved as clean_hd.csv

LASSO:
To practice cross-validation, the dataset was split into a training set and a testing set randomly with the portion of 8:2. Matrixes for training set and testing set was build for LASSO.



The ordinary least squares regression was conducted as a baseline. Then the lambda selection was conducted. At lambda.min, the LASSO gives the mean square error with most variables left.

RMSE for OSL training set is 26.43769 and for testing set is 22.13102. RMSE for testing set of LASSO model at lambda.min is 25.21313

Minimum lambda is 0.3639 and the corresponding R-squared is 76.66% (below left) the coefficient of each variables are listed below (right)

	Df	%Dev	Lambda
1	0	0.0000	46.370
2	1	0.4104	28.560
3	2	0.5781	17.590
4	6	0.6527	10.830
5	6	0.6900	6.671
6	8	0.7157	4.108
7	9	0.7427	2.530
8	13	0.7562	1.558
9	13	0.7623	0.960
10	15	0.7648	0.591
11	16	$\boxed{0.7666 \quad 0.364}$	
12	17	0.7675	0.224
13	17	0.7680	0.138
14	18	0.7682	0.085
15	18	0.7682	0.052
16	18	0.7683	0.032
17	18	0.7683	0.020
18	19	0.7683	0.012

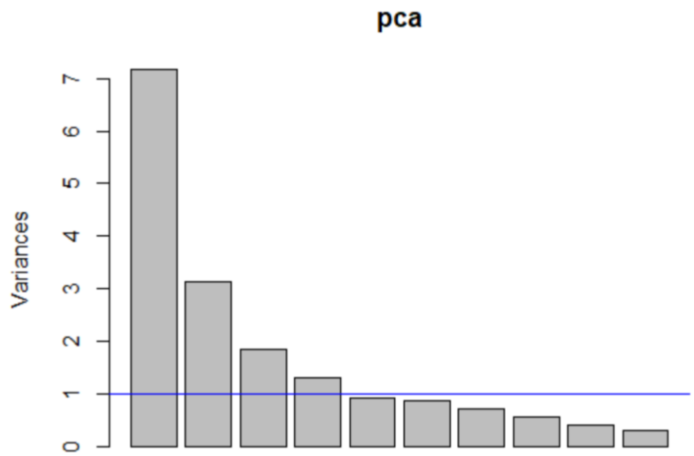
```
> coef(fitLasso, s=fitLasso$lambda.min)
20 x 1 sparse Matrix of class "dgCMatrix"
                       1
(Intercept)    -5.535459
v1            -81.587351
v2             33.083097
v3              .
v4              .
v5              .
v6           -508.631955
v7             14.885851
v8             65.531789
v9             65.950091
v10             4.066561
v11            47.947056
v12           157.435089
v13             6.625477
v14           342.049234
v15            -1.201901
v16             .
v17            10.903107
v18             .
v19             .
```

This model shows good prediction ability with selected feature.

PCA analysis:

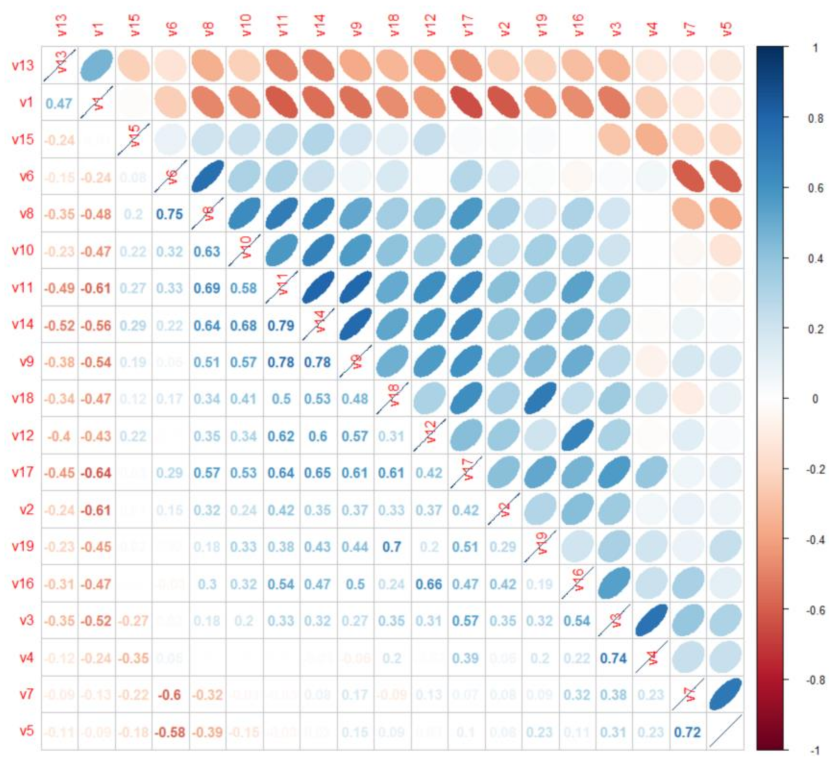
After checking the attribute of the different variables, it makes sense to use scaled PCA:

```
> summary(pca)
Importance of components:
                          PC1    PC2     PC3     PC4     PC5     PC6     PC7     PC8     PC9    PC10
Standard deviation     2.6773 1.7691 1.35660 1.14116 0.95711 0.93246 0.85074 0.74620 0.63360 0.55864
Proportion of Variance 0.3772 0.1647 0.09686 0.06854 0.04821 0.04576 0.03809 0.02931 0.02113 0.01643
Cumulative Proportion  0.3772 0.5420 0.63883 0.70737 0.75559 0.80135 0.83944 0.86875 0.88988 0.90630
                          PC11    PC12    PC13    PC14    PC15    PC16    PC17    PC18    PC19
Standard deviation     0.55262 0.52263 0.49677 0.49243 0.43554 0.39332 0.37562 0.37084 0.29914
Proportion of Variance 0.01607 0.01438 0.01299 0.01276 0.00998 0.00814 0.00743 0.00724 0.00471
Cumulative Proportion  0.92237 0.93675 0.94974 0.96250 0.97248 0.98063 0.98805 0.99529 1.00000
```



After conducted scaled PCA, we can say the knee is about starting at fifth principal component. The first 4 components will cover 70.7% variances. Since we are trying to find the hidden information here, it's good enough to have first 4 components.

The correlation matrix of numeric variables of our dataset listed below:



Four groups can be found in this plot.

Then the testing of the correlation matrix was applied

Here I used $p < 0.05$ as the standard of significant and here's the matrix:

	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15	v16	v17	v18	v19
v1	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE
v2	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE
v3	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v4	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
v5	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
v6	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE
v7	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v8	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v9	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v10	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v11	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v12	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v13	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v14	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v15	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE
v16	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE
v17	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE
v18	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
v19	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE

Then find the true number of each variable:

v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15	v16	v17	v18	v19
17	16	17	11	15	12	16	17	17	16	15	15	18	16	13	16	17	18	16

If we consider >90% correlated to other variables as over correlated, we can find v13 and v19 are corelated to any other variables. These two variables were removed from the factor analysis.

The factor analysis has chosen 4 as factor number and if choose 0.4 as cutoff value, the result listed here:

Loadings:

	RC1	RC4	RC2	RC3
v1	-0.592	-0.461		
v2		0.519		
v3		0.421		0.753
v4				0.872
v5			0.843	
v6			-0.890	
v7			0.809	
v8	0.539		-0.650	
v9	0.705	0.498		
v10	0.690			
v11	0.652	0.589		
v12		0.828		
v14	0.744	0.490		
v15				-0.650
v16		0.856		
v17	0.720			
v19	0.781			

	RC1	RC4	RC2	RC3
SS loadings	4.104	3.315	2.798	2.266
Proportion Var	0.241	0.195	0.165	0.133
Cumulative Var	0.241	0.436	0.601	0.734

Analyzing the factors could get some very interesting conclusion. The RC1 explains variances that higher percentage of some bad health index, like diabetes, smoking, obesity, etc. This factor also have higher harmful objective rates, such as lower physician per capita rate and motor crash death rate. Civilian labor percentage is negative coefficient here. The other factor analysis requires a better understanding of data itself since RC4 shows some practical significance similarity with RC1 but the variables are somewhat different.

Next, I will find more information about what each variable really means and finish up the factor analysis. Then conduct linear discriminant analysis and maybe apply multidimensional scaling or cluster analysis to find more information from the dataset.

R command

```
# data cleaning
hd = heart_disease
dt = na.omit(hd)
dt = dt[,-1]
# rename column names
cnames=paste("v",1:22,sep="")
cnames=c('mort',cnames)
cnames
colnames(dt)=cnames
colnames(hd)
# data transfromation
dt$v2 = log(dt$v2+1)
dt$v4 = log(dt$v4+1)
dt$v7 = log(dt$v7+1)
dt$v12= log(dt$v12+1)
dt$v16= log(dt$v16+1)
dt$v17= log(dt$v17+1)
dt$v18= log(dt$v18+1)
dt$v19= log(dt$v19+1)
dt$v8 = sqrt(dt$v8)
dt$v11 = sqrt(dt$v11)
hist(dt$v16)

# save the dataset
write.csv(dt,"D:\\clean_hd.csv",row.names = FALSE)

# LASSO
# split the dataset to training and testing sets
set.seed(166)
partition = sample(2,nrow(dt),replace=T,prob=c(0.80,0.20))
train = dt[partition==1,]
test = dt[partition==2,]

# Separate the X's and Y's as matrices
xTrain = as.matrix(train[, -c(1,21:23)])   # Take out column 1 and cate col 21:23
yTrain = as.matrix(train[, 1])    # Take only column 1
xTest = as.matrix(test[, -c(1,21:23)])   # Take out column 1
yTest = as.matrix(test[, 1])    # Take only column 1
```

```r
#OLS
OLS = lm (mort ~ ., data = train)
summary(OLS)
#find RMSE
rmseTrain = sqrt(mean(OLS$residuals^2))
rmseTrain
#predict on the test set and RMSE of test set
olsPredict = predict(OLS, test)
rmseTest = sqrt(mean((olsPredict - test$mort)^2))
rmseTest
library(car)

#LASSO
library(glmnet)
fitLasso = cv.glmnet(xTrain, yTrain, alpha=1, nlambda = 20)
fitLasso
plot(fitLasso)
summary(fitLasso)
fitLasso$lambda.1se
fitLasso$lambda.min

# select minimum lambda
lassoPred = predict(fitLasso, xTest, s="lambda.min")
rmseLasso = sqrt(mean((lassoPred - yTest)^2))
rmseLasso

# coef and R-square
coef(fitLasso, s=fitLasso$lambda.min)
fit = glmnet(xTrain, yTrain, alpha=1, nlambda = 20)
print(fit)

# PCA
summary(dt)
pca = prcomp(dt[,2:20],scale. = T)
summary(pca)
plot(pca)
abline(h=1,lwd=1,col="blue")
```

```
# correlation plot
cor = cor(dt[,-c(1,21:23)])
corrplot(cor, order="AOE",method="ellipse")
corrplot(cor,method = "ellipse",tl.pos = NULL, tl.cex = 0.65,order="AOE")
corrplot(cor,type="upper",order="AOE",method = "ellipse")
corrplot(cor,add=TRUE, type="lower", method="number",diag=FALSE, cl.pos="n",order="AOE")
library(psych)
p2 = principal(dt[,2:20],nfactor =4, rotate="varimax")
print(p2$loadings,cutoff=.4)

# correlation test
round(cor_em,2)
corTest = corr.test(dt[,2:20],adjust="none")
round(corTest$p,2)
MTest=ifelse(corTest$p<0.05, T, F)
MTest
colSums(MTest)-1
# delete v13 v18 and factor analysis
fa = dt[,-c(1,14,19,21:23)]
faA = principal(fa,nfactor =4, rotate="varimax")
print(faA$loadings,cutoff=.4)
```