Prefer the highlighted

Traffic Volume - https://archive.ics.uci.edu/ml/datasets/Metro+Interstate+Traffic+Volume

Air Quality - https://archive.ics.uci.edu/ml/datasets/Beijing+Multi-Site+Air-Quality+Data

Forest Fire - https://archive.ics.uci.edu/ml/datasets/Forest+Firess

Preprocessing:

- **Background:** Briefly explain the topic and what are we trying to solve
  - http://www.dot.state.mn.us/traffic/data/coll-methods.html#ATR
- **Categorical/Numerical approche:** predictor
- **General distribution:** Describe the predictor
- **Skewness:** pca, centering, scaling, Box-Cox
- **Comparison Before and After:** predictor Distribution
- **Missing Value:** Knn(most accurate) imputation, if most

1. Goal: To understand how do weather, holidays, temperatures, and time affect the metro interstate traffic volume from 2012-2018.
2. Data Structures:
   a. Categorical:
      i. holiday
      ii. weather_main
      iii. weather_description
      iv. date_time
   b. Numerical:
      i. temp
      ii. rain_1h
      iii. snow_1h
      iv. clouds_all
      v. traffic_volume
   c. Sample size: 48204 observations & 9 variables
   d. Response variable: traffic_volume
3. Preprocess data:
   a. Dummy variables for categorical data
   b. Delete predictor if:
      i. Highly correlated
      ii. Near-zero predictor
   c. Imputation:
      i. Check if we have missing value (we do not have missing values)
      ii. Box-cox transformation (prefer not to use since some predictors have 0s)
      iii. PC

iv. Center and Scale
v. Spatial Sign for outliers?? (we can do box plot to identify outliers)

d. How to spend data?
i. ? % training set and ? % testing set
ii. Resampling? Which method is appropriate?

```
## 3.1 Summary Statistics ##
library(e1071) #install for skewness
library(gridExtra)
library(grid)
frame()
summarystats <- rbind(mean = apply(df3_1_pred, 2, mean),
                      median = apply(df3_1_pred, 2, median),
                      sd = apply(df3_1_pred, 2, sd),
                      apply(df3_1_pred, 2, quantile),
                      IQR = apply(df3_1_pred, 2, IQR),
                      skewness = apply(df3_1_pred, 2, skewness))
tt=ttheme_default()
grid.table(round(summarystats,digits = 4), theme=tt)

## 3.1 Boxplots for each Predictor ##
sbplts = length(1:ncol(df3_1_pred))
clplts = 3
rwplts = ceiling(sbplts/clplts)
par(mfrow=c(rwplts,clplts))
for (col in 1:ncol(df3_1_pred)) {
  boxplot(df3_1_pred[,col] ~ df3_1_resp[,1],  main = paste("Boxplot of" ,
colnames(df3_1_pred)[col],"by Type"),
          xlab="Type", ylab="Percentage")
  #Sys.sleep(1) #Pause between plots
}
dev.off() #reset plots

## 3.1 Histograms for each Predictor ##
frame()
sbplts = length(1:ncol(df3_1_pred))
clplts = 3
rwplts = ceiling(sbplts/clplts)
```

```
par(mfrow=c(rwplts,clplts))
for (col in 1:ncol(df3_1_pred)) {
  hist(df3_1_pred[,col],  main = paste("Histogram of" , colnames(df3_1_pred)[col]),
       xlab="Bin", ylab="Frequency")
  #Sys.sleep(1) #Pause between plots
}
dev.off() #reset plots

## 3.2 Frequency Plots for each Predictor ##
frame()
sbplts = length(1:ncol(df3_2_pred))
clplts = 7
rwplts = ceiling(sbplts/clplts)
par(mfrow=c(rwplts,clplts))
for (col in 1:ncol(df3_2_pred)) {
  barplot(table(df3_2_pred[,col]),  main = paste("Frequency Plot of" ,
colnames(df3_2_pred)[col]),
          xlab="Level", ylab="Frequency")
  #Sys.sleep(1) #Pause between plots
}
dev.off() #reset plots
```

Model fit:

- Linear model (PCR)
- Lasso (feature selection) note: before and after znv
- Ridge, e-net
- Rank correlation stuff (SAS, R?)
- Find the best model -> comparison -> make tables and plots(overfit OR underfit?) say with 10% loss??