```
weather <- read.csv("C:/Users/nguye/OneDrive/Classes/8. Winter 2025/STATS
101A/Project/weatherHistory.csv")

# Rearranging data
weather <- weather |> select(Temperature..C., Humidity, Wind.Speed..km.h., Visibility..km.,
Pressure..millibars., Loud.Cover)
weather <- weather |>
  rename(
    Temperature = Temperature..C.,
    Wind = Wind.Speed..km.h.,
    Visibility = Visibility..km.,
    Pressure = Pressure..millibars.,
    LoudCover = Loud.Cover
  ) |> sample_n(1000) |> filter(Pressure > 0)
weather

# Create model
weather_model <- lm(Wind~Temperature+Humidity+Visibility+Pressure, weather)
summary(weather_model)

# One variable model
small_model <- lm(Wind~Pressure, weather)
ggplot(weather, aes(x = Pressure, y = Wind)) +
   geom_point() +  # Adds scatter plot points
   geom_smooth(method = "lm", col = "blue")

summary(small_model)

avPlot(weather_model, variable = Temperature, ask = FALSE)

######### FIVE-NUM ###########

numeric_weather <- weather |> select(Wind, Temperature, Humidity, Visibility, Pressure)
fivenum_summary <- apply(numeric_weather, 2, fivenum)

# Convert the result to a data frame and transpose to swap rows and columns
fivenum_df <- as.data.frame(fivenum_summary)
rownames(fivenum_df) <- c("Min", "Q1", "Median", "Q3", "Max")
fivenum_df


################ ANOVA #################
```

```
weather_model_with_visibility <- lm(Temperature ~ Humidity + Wind + Visibility + Pressure, data
= weather)

anova_with_visibility <- anova(weather_model_with_visibility)

print(anova_with_visibility)

weather_model_without_visibility <- lm(Temperature ~ Humidity + Wind + Pressure, data =
weather)

anova_without_visibility <- anova(weather_model_without_visibility)
print(anova_without_visibility)

model <- lm(Wind~Temperature+Humidity+Visibility+Pressure, weather)
reduced_model <- lm(Wind~Temperature+Humidity+Pressure, weather)
anova(reduced_model, model)

library(GGally)
ggpairs(weather[, c("Wind", "Temperature", "Humidity", "Visibility", "Pressure")])

anova(reduced_model)

################# PLOTS ##################

for (var in explanatory_vars) {
  model <- lm(Wind ~ get(var), data = weather)  # Create model for each variable
  par(mfrow = c(2,2))  # 2x2 plot layout
  plot(model, main = paste("Diagnostics for", var))  # Generate diagnostics
  readline(prompt = "Press [Enter] to continue")  # Wait before moving to the next variable
}

############### TESTS ####################

library(lmtest)

boxcox(weather_model)
hist(weather$Wind, main = "Histogram of Wind", col = "orange", breaks = 20)
hist(weather$Visibility, main = "Histogram of Visibility", col = "orange", breaks = 20)
shapiro.test(weather$Temperature)
bptest(weather_model)



# TRANSFORMATIONS
```

```
#doing this so that all variables are positive
weather$Temperature <- weather$Temperature+21.83
weather$Wind <- weather$Wind + 0.01

#box-cox
summary(tranxy<-powerTransform(cbind(weather$Wind,weather$Temperature,weather$Humidity,weather$Pressure)~1))

#transformed variables
wind_sqrt <- weather$Wind^0.5
humidity_squared <- weather$Humidity^2
pressure_inv3 <- weather$Pressure^(-3)

#transformed model
model_transformed <- lm(wind_sqrt ~ Temperature + humidity_squared + pressure_inv3, data = weather)
summary(model_transformed)

#looking at residual plots for transformed
par(mfrow = c(2, 2))
plot(model_reduced)
plot(model_transformed)
```