Correlation Matrix

```
# Load necessary libraries
library(dplyr)
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
library(corrplot) # correlation matrix visualization
```

```
corrplot 0.92 loaded
```

```
library(ggcorrplot) # ggplot2 style visualization of correlation matrix
```

```
Loading required package: ggplot2
```

```
# Read datasets
data <- read.csv("../../Data/data.csv")
data2 <- read.csv("../../Data/500cities.csv")

# Prepare 'data' by padding and merging columns to create a unique identifier
data$STATEFP <- sprintf("%02d", as.numeric(data$STATEFP))
data$COUNTYFP <- sprintf("%03d", as.numeric(data$COUNTYFP))
data$TRACTCE <- sprintf("%06d", as.numeric(data$TRACTCE))
data$TractFIPS <- as.numeric(paste0(data$STATEFP, data$COUNTYFP, data$TRACTCE))

# Summarize 'data' to get mean of 'NatWalkInd' by 'TractFIPS'
averaged_data <- data %>%
  group_by(TractFIPS) %>%
  summarise(NatWalkInd = mean(NatWalkInd, na.rm = TRUE)) %>%
  ungroup()

# Select relevant columns from 'data2'
data2_relevant <- data2 %>%
  select(TractFIPS, StateAbbr, DIABETES_CrudePrev, BPHIGH_CrudePrev, OBESITY_CrudePrev, LPA_CrudePrev, C

# Merge 'averaged_data' with 'data2_relevant' on 'TractFIPS'
merged_data <- merge(averaged_data, data2_relevant, by = "TractFIPS", all.x = TRUE, all.y = TRUE)

# Filter for specific condition
merged_data <- merged_data %>% filter(StateAbbr == "CA")
```

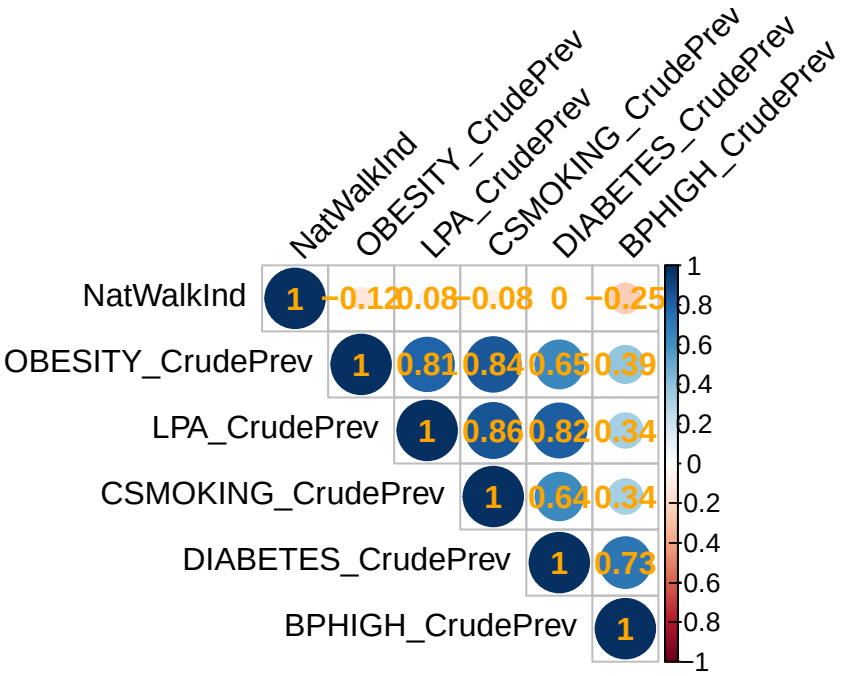
```r
# Calculate correlation matrix
cor_matrix <- cor(merged_data %>% select(-TractFIPS, -StateAbbr), use = "complete.obs")  # Handling miss

# Output variable names (column names) used for correlation matrix
cat("Variables used for correlation matrix:\n")
```

```
Variables used for correlation matrix:
```

```r
print(names(merged_data %>% select(-TractFIPS, -StateAbbr)))
```

```
[1] "NatWalkInd"         "DIABETES_CrudePrev" "BPHIGH_CrudePrev"
[4] "OBESITY_CrudePrev"  "LPA_CrudePrev"      "CSMOKING_CrudePrev"
```

```r
# Make correlation matrix
corrplot(cor_matrix, method = "circle", type = "upper", order = "hclust",
         tl.col = "black", tl.srt = 45,  # Text label color and rotation
         addCoef.col = "orange")  # Add correlation coefficients to plot
```



The correlation matrix above indicates strong positive correlations between obesity, physical activity, smoking, diabetes, and high blood pressure prevalence, with less correlation to the national walking index. Given these relationships, my team decided that a spatial autoregressive model (SAR) or a geographically weighted regression (GWR) could be appropriate to account for spatial dependencies and variations in the data.

Geographically Weighted Regression Model

```r
library(GWmodel)
```

```
Loading required package: robustbase
```

```
Loading required package: sp
```

```
Loading required package: Rcpp
```

```
Welcome to GWmodel version 2.3-2.
```

```r
library(sf)
```

```
Linking to GEOS 3.11.0, GDAL 3.5.3, PROJ 9.1.0; sf_use_s2() is TRUE
```

```r
spatial_data <- st_read("../../Data/shapes/ca_tract")
```

```
Reading layer `tl_2021_06_tract' from data source
  `/Users/sebastian/Documents/School/stat489/Spatiotemporal_Analysis_of_Diabetes_Incidence/data/shapes/c
  using driver `ESRI Shapefile'
Simple feature collection with 9129 features and 12 fields
Geometry type: MULTIPOLYGON
Dimension:     XY
Bounding box:  xmin: -124.482 ymin: 32.52883 xmax: -114.1312 ymax: 42.0095
Geodetic CRS:  NAD83
```

```r
spatial_data$TractFIPS <- as.numeric(spatial_data$GEOID)
merged_data <- merge(merged_data, spatial_data, by = "TractFIPS", all.x = TRUE, all.y = TRUE)

merged_data <- na.omit(merged_data)
merged_sf <- st_as_sf(merged_data)
merged_spatialdf <- as(merged_sf, "Spatial")

merged_gwr_bw <- bw.gwr(DIABETES_CrudePrev ~ OBESITY_CrudePrev + BPHIGH_CrudePrev + LPA_CrudePrev + CSMO
                        data = merged_spatialdf,
                        kernel = "exponential",
                        )
```

```
Take a cup of tea and have a break, it will take a few minutes.
          -----A kind suggestion from GWmodel development group
Fixed bandwidth: 8.41394 CV score: 2720.139
Fixed bandwidth: 5.201141 CV score: 2630.17
Fixed bandwidth: 3.215522 CV score: 2514.129
Fixed bandwidth: 1.988342 CV score: 2370.523
Fixed bandwidth: 1.229903 CV score: 2185.328
Fixed bandwidth: 0.7611615 CV score: 1953.585
Fixed bandwidth: 0.4714636 CV score: 1685.891
Fixed bandwidth: 0.2924204 CV score: 1431.498
Fixed bandwidth: 0.1817656 CV score: 1226.298
Fixed bandwidth: 0.1133772 CV score: 1075.896
Fixed bandwidth: 0.07111088 CV score: 963.9296
Fixed bandwidth: 0.04498883 CV score: 894.579
Fixed bandwidth: 0.02884452 CV score: 891.5145
Fixed bandwidth: 0.01886678 CV score: 20643.8
Fixed bandwidth: 0.0350111 CV score: 876.0943
Fixed bandwidth: 0.03882225 CV score: 881.6854
Fixed bandwidth: 0.03265567 CV score: 874.9931
Fixed bandwidth: 0.03119994 CV score: 875.3238
Fixed bandwidth: 0.03355536 CV score: 874.9161
Fixed bandwidth: 0.0341114 CV score: 875.3624
Fixed bandwidth: 0.03321171 CV score: 875.0572
Fixed bandwidth: 0.03376775 CV score: 875.0916
Fixed bandwidth: 0.0334241 CV score: 874.8954
Fixed bandwidth: 0.03334298 CV score: 874.9637
Fixed bandwidth: 0.03347424 CV score: 874.8823
Fixed bandwidth: 0.03350523 CV score: 874.9585
```

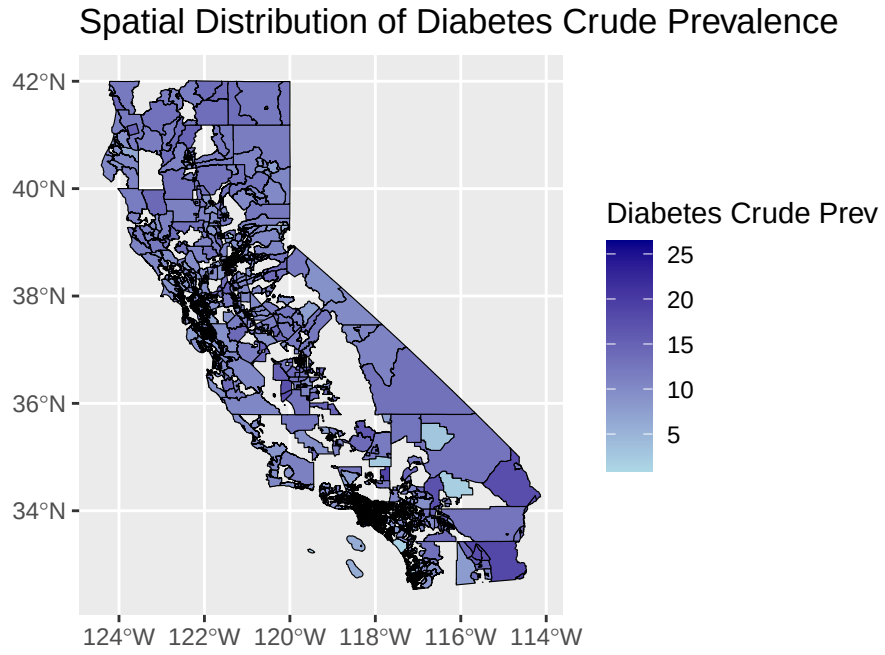
```
merged_gwr <- gwr.basic(DIABETES_CrudePrev ~ OBESITY_CrudePrev + BPHIGH_CrudePrev + LPA_CrudePrev + CSMO
                        data = merged_spatialdf,
                        bw = merged_gwr_bw,
                        kernel = "exponential",
                        )

gwr_results_sf <- merged_gwr$SDF %>% as("sf")
```

```
summary(gwr_results_sf)
```

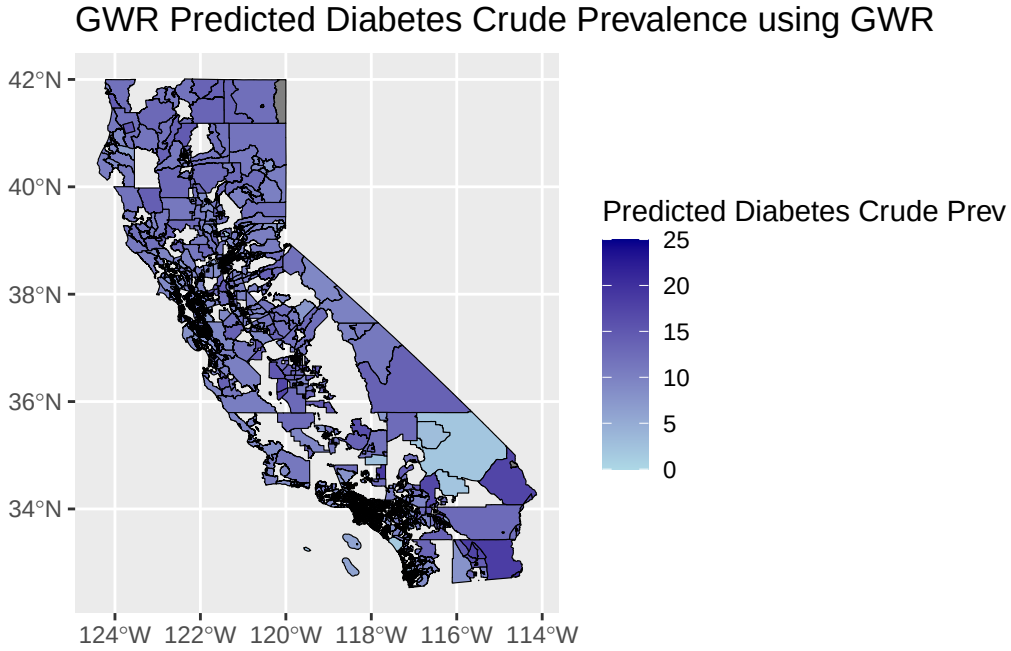
```
  Intercept       OBESITY_CrudePrev  BPHIGH_CrudePrev  LPA_CrudePrev
 Min.   :-50.449   Min.   :-8.31242   Min.   :-0.1448   Min.   :-0.3606
 1st Qu.: -5.239   1st Qu.:-0.04769   1st Qu.: 0.2927   1st Qu.: 0.2544
 Median : -4.242   Median :-0.01005   Median : 0.3209   Median : 0.3309
 Mean   : -4.648   Mean   : 0.03475   Mean   : 0.3233   Mean   : 0.3126
 3rd Qu.: -3.258   3rd Qu.: 0.09442   3rd Qu.: 0.3517   3rd Qu.: 0.3830
 Max.   : 52.917   Max.   : 2.87757   Max.   : 2.6076   Max.   : 1.4275
 CSMOKING_CrudePrev       y               yhat            residual
 Min.   :-3.5116    Min.   : 0.900   Min.   :-22.266   Min.   :-80.69782
 1st Qu.:-0.3397    1st Qu.: 8.100   1st Qu.:  8.128   1st Qu.: -0.09385
 Median :-0.2447    Median : 9.600   Median :  9.649   Median :  0.00213
 Mean   :-0.2219    Mean   : 9.913   Mean   :  9.917   Mean   : -0.00378
 3rd Qu.:-0.1193    3rd Qu.:11.500   3rd Qu.: 11.535   3rd Qu.:  0.11349
 Max.   : 5.1472    Max.   :26.400   Max.   : 92.998   Max.   : 34.76556
    CV_Score Stud_residual        Intercept_SE      OBESITY_CrudePrev_SE
 Min.   :0   Min.   :-54.38814   Min.   :  0.4058   Min.   : 0.01850
 1st Qu.:0   1st Qu.: -0.09849   1st Qu.:  1.1421   1st Qu.: 0.04873
 Median :0   Median :  0.00878   Median :  1.6974   Median : 0.10193
 Mean   :0   Mean   : -0.01825   Mean   :  4.8746   Mean   : 0.29319
 3rd Qu.:0   3rd Qu.:  0.11249   3rd Qu.:  4.3050   3rd Qu.: 0.29059
 Max.   :0   Max.   : 31.32936   Max.   :757.4998   Max.   :65.68896
 BPHIGH_CrudePrev_SE LPA_CrudePrev_SE   CSMOKING_CrudePrev_SE  Intercept_TV
 Min.   : 0.01605    Min.   : 0.02229   Min.   : 0.05539      Min.   :-11.086
 1st Qu.: 0.03880    1st Qu.: 0.04799   1st Qu.: 0.10995      1st Qu.: -3.745
 Median : 0.05654    Median : 0.08324   Median : 0.18846      Median : -1.893
 Mean   : 0.10491    Mean   : 0.17709   Mean   : 0.34669      Mean   : -2.749
 3rd Qu.: 0.09259    3rd Qu.: 0.16659   3rd Qu.: 0.35787      3rd Qu.: -1.020
 Max.   :18.19498    Max.   :10.08113   Max.   :41.56039      Max.   :  1.999
 OBESITY_CrudePrev_TV BPHIGH_CrudePrev_TV LPA_CrudePrev_TV
 Min.   :-5.06151     Min.   :-0.03597    Min.   :-0.4077
 1st Qu.:-0.74986     1st Qu.: 3.25034    1st Qu.: 1.4223
 Median :-0.09819     Median : 5.67225    Median : 3.8258
 Mean   :-0.22545     Mean   : 6.61666    Mean   : 5.1327
 3rd Qu.: 0.35848     3rd Qu.: 8.34267    3rd Qu.: 7.6321
 Max.   : 2.95653     Max.   :23.64133    Max.   :18.1830
 CSMOKING_CrudePrev_TV    Local_R2                    geometry
 Min.   :-7.4841       Min.   :-1141.7257   MULTIPOLYGON :6831
 1st Qu.:-2.4507       1st Qu.:    0.9879   epsg:4269    :   0
 Median :-1.0289       Median :    0.9925   +proj=long...:   0
 Mean   :-1.6313       Mean   :    0.7953
 3rd Qu.:-0.3797       3rd Qu.:    0.9958
 Max.   : 1.9432       Max.   :    1.0000
```

```
library(ggplot2)

# Plot the spatial distribution of DIABETES_CrudePrev
```

```
ggplot() +
  geom_sf(data = merged_sf, aes(fill = DIABETES_CrudePrev), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Spatial Distribution of Diabetes Crude Prevalence")
```



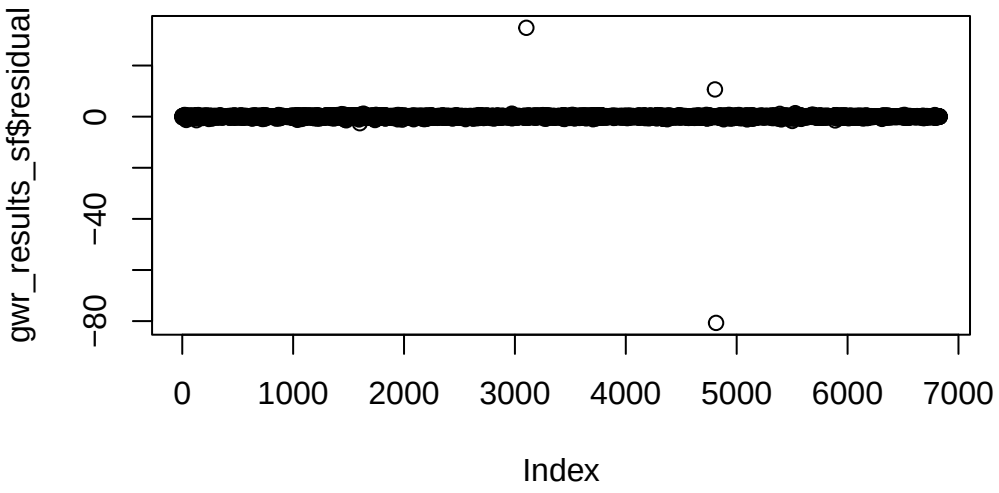

```
# Plot the spatial distribution of GWR results
ggplot() +
  geom_sf(data = gwr_results_sf, aes(fill = yhat), color = "black", size = 0.2) +
  scale_fill_gradient(name = "Predicted Diabetes Crude Prev", low = "lightblue", high = "darkblue",limit
  labs(title = "GWR Predicted Diabetes Crude Prevalence using GWR")
```



The first plot shows the actual geographical distribution of diabetes crude prevalence throughout California, with the intensity of the blue shade meaning the prevalence rates and the darker areas signaling higher diabetes prevalence. This is shown mostly in the central and southern regions, which highlights the spatial

variability in diabetes prevalence and suggests that lifestyle factors, healthcare access, and demographics might be influential here. The second plot presents predictions from our Geographically Weighted Regression (GWR) model, showing a spatial gradient of diabetes prevalence. The model shows higher rates in similar central areas as observed in the actual data. This consistency between the observed and predicted patterns emphasizes the potential impact of local factors on diabetes prevalence. These maps give a complete view of both the current state and modeled predictions of diabetes distribution, which could be very insightful when considering health strategies and specific geographic needs.

```
plot(gwr_results_sf$residual)
```



The plot above is a residual plot from the Geographically Weighted Regression (GWR) analysis. The distribution of points along the horizontal line at zero suggests that the model has no systematic error across the range of the data. However, there are several outliers, which can be seen by points above or below the horizontal line, which probably represents areas where the model's predictions are less accurate. In general, the residuals are relatively evenly scattered, which implies it is a good fit for most of the data points, although the outliers suggest areas that might need further examination or could be due to unique local factors not captured by the model.

Random Forest Model

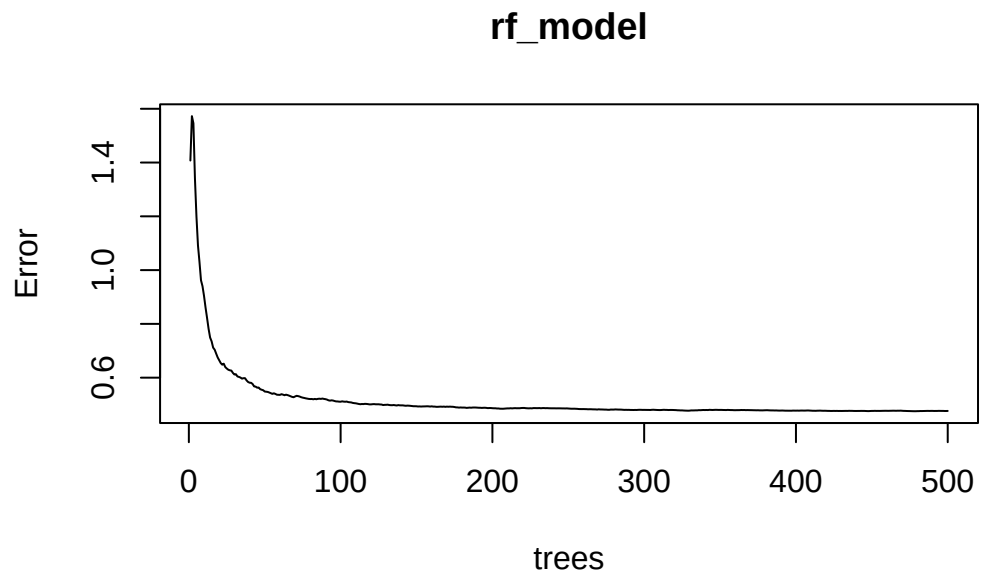
```
library(randomForest)
```

```
randomForest 4.7-1.1

Type rfNews() to see new features/changes/bug fixes.


Attaching package: 'randomForest'

The following object is masked from 'package:ggplot2':

    margin

The following object is masked from 'package:dplyr':

    combine
```

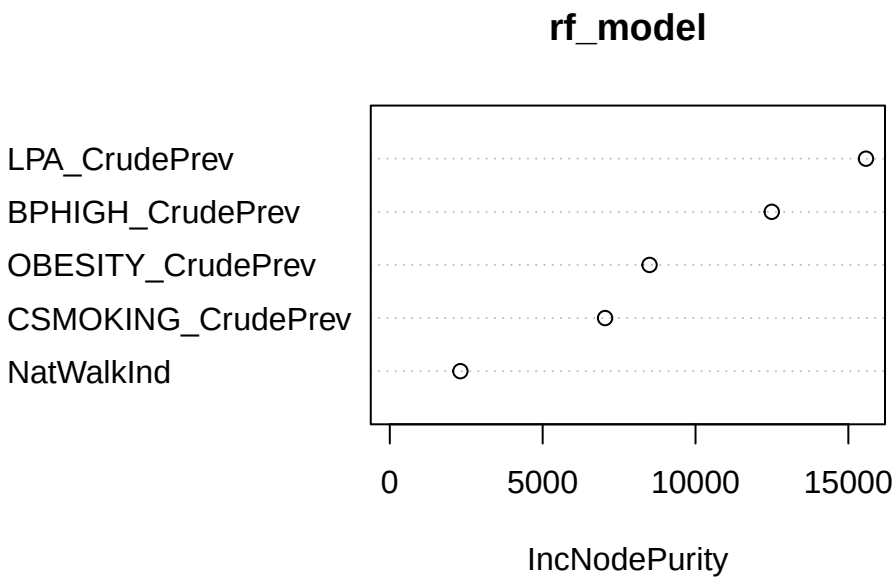
```r
data_df <- as.data.frame(merged_spatialdf)

rf_model <- randomForest(DIABETES_CrudePrev ~ NatWalkInd+OBESITY_CrudePrev + BPHIGH_CrudePrev + LPA_Crud
print(rf_model)
```

```
Call:
 randomForest(formula = DIABETES_CrudePrev ~ NatWalkInd + OBESITY_CrudePrev +      BPHIGH_CrudePrev + LP
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 1

          Mean of squared residuals: 0.4761803
                    % Var explained: 93.05
```

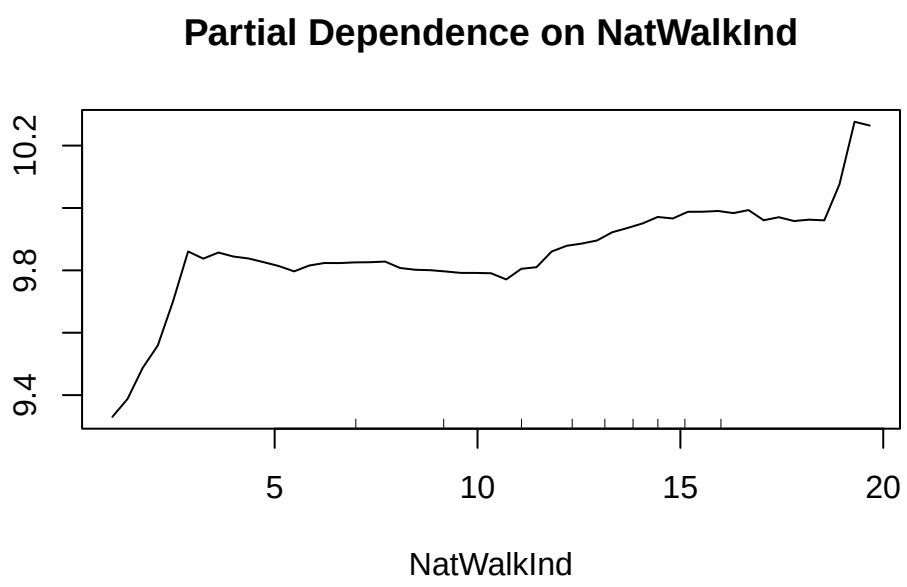
```r
plot(rf_model)
```



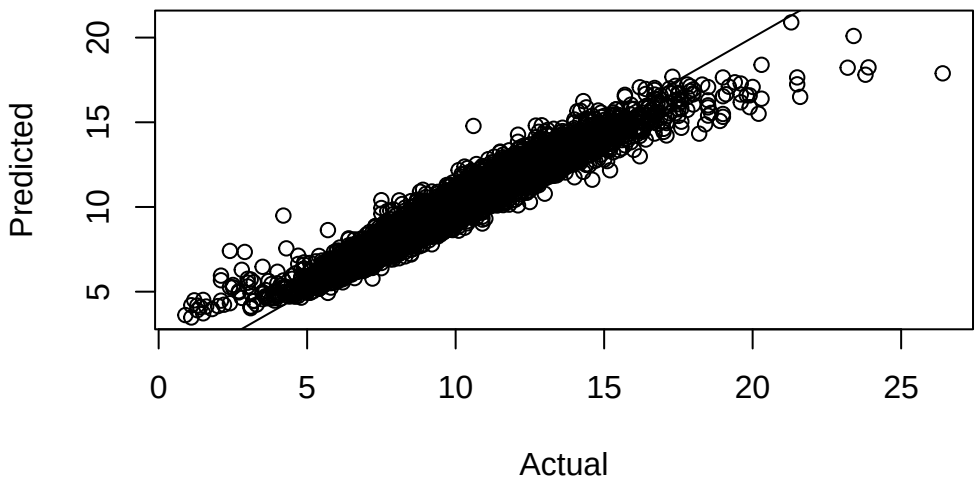

```r
varImpPlot(rf_model)
```




```
partialPlot(rf_model, data_df, NatWalkInd)
```

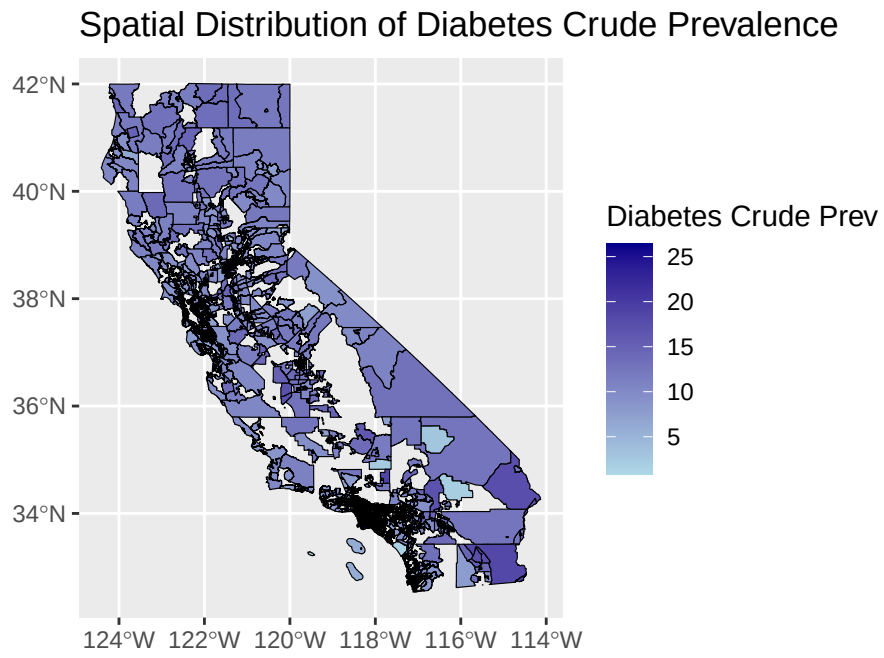



```
plot(data_df$DIABETES_CrudePrev, predict(rf_model), xlab = "Actual", ylab = "Predicted")
abline(0,1)
```

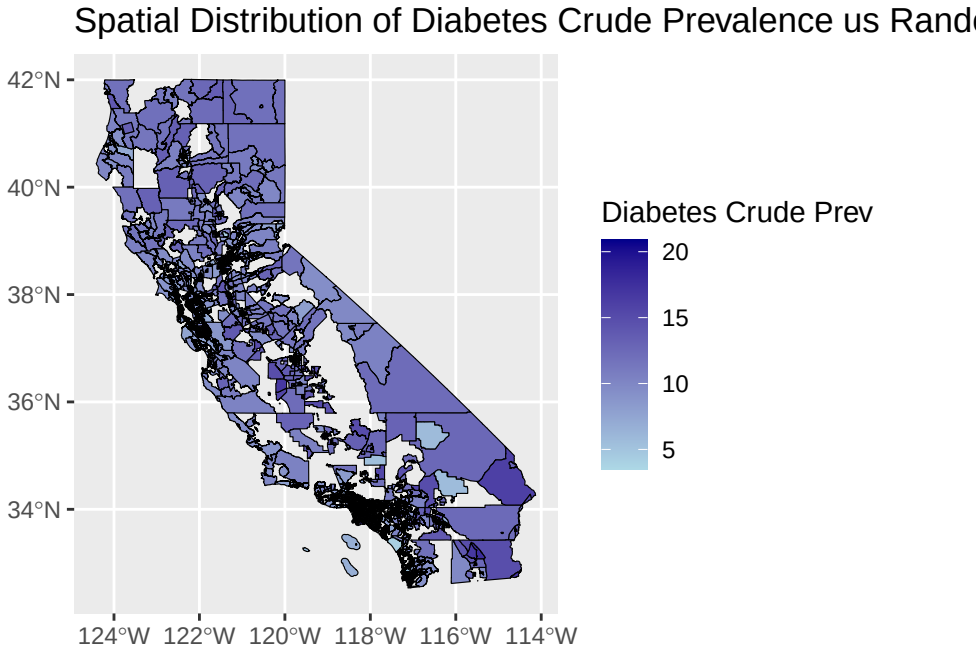



```
modelPrediction <- predict(rf_model)
merged_sf$rfModelValues <- modelPrediction
merged_sf$rf_residuals <- merged_sf$DIABETES_CrudePrev - merged_sf$rfModelVal
```

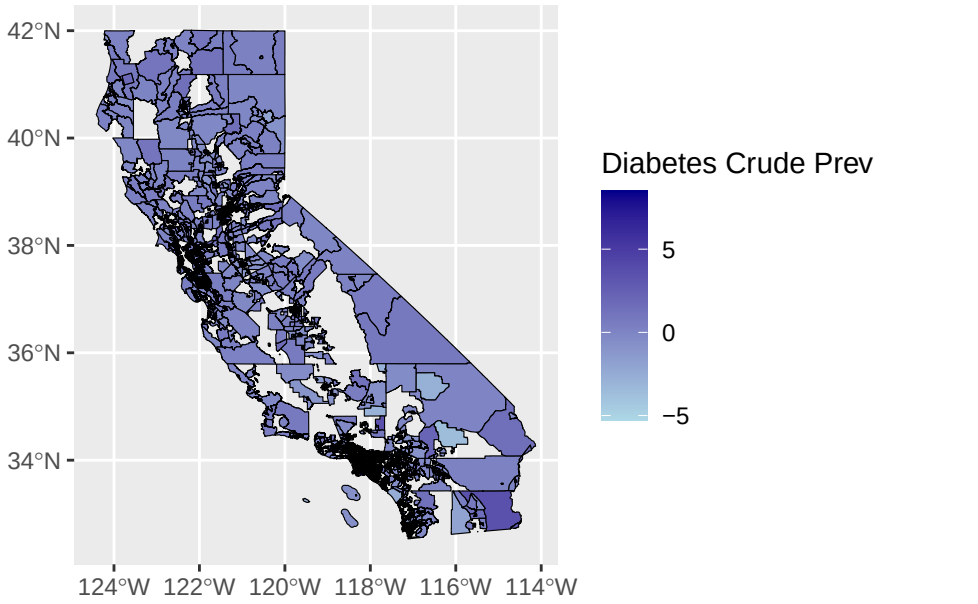
```
ggplot() +
  geom_sf(data = merged_sf, aes(fill = DIABETES_CrudePrev), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Spatial Distribution of Diabetes Crude Prevalence")
```




```
ggplot() +
  geom_sf(data = merged_sf, aes(fill = rfModelValues), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Spatial Distribution of Diabetes Crude Prevalence us Random Forest Model")
```




```
ggplot() +
  geom_sf(data = merged_sf, aes(fill = rf_residuals), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Residuals of Random Forest Model")
```



Gradient Boosted Model

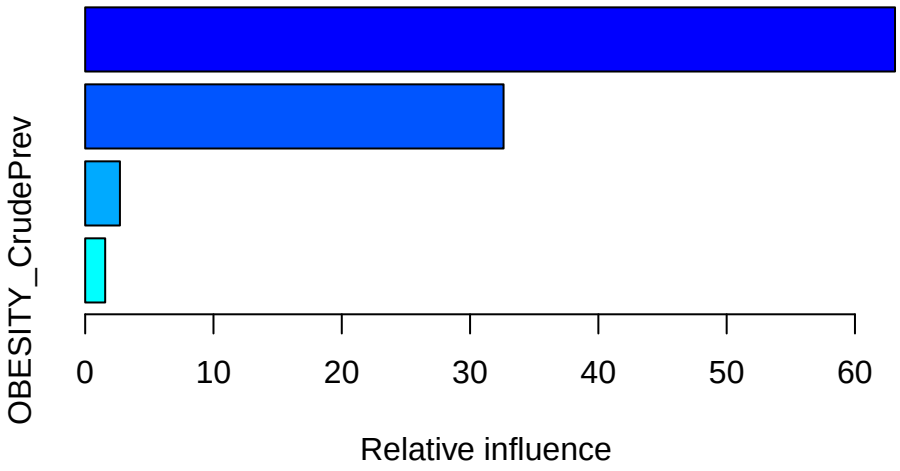
```
library(gbm)
```

```
Loaded gbm 2.1.9

This version of gbm is no longer under development. Consider transitioning to gbm3, https://github.com/g
```

```r
merged_spatialdf <- as(merged_sf, "Spatial")
data_df <- as.data.frame(merged_spatialdf)

gbm_model <- gbm(DIABETES_CrudePrev ~ OBESITY_CrudePrev + BPHIGH_CrudePrev + LPA_CrudePrev + CSMOKING_Cr
                 data = data_df,
                 distribution = "gaussian",
                 n.trees = 500,
                 interaction.depth = 3,
                 shrinkage = 0.1,
                 bag.fraction = 0.5,
                 cv.folds = 5,
                 verbose = TRUE)
```

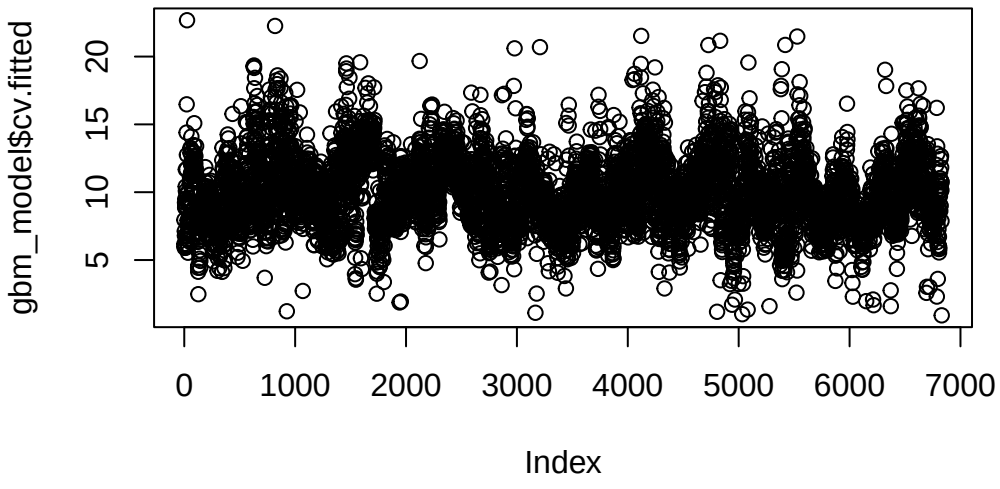
```
Iter   TrainDeviance   ValidDeviance   StepSize   Improve
     1        6.0116             nan     0.1000    0.8146
     2        5.3240             nan     0.1000    0.6610
     3        4.7310             nan     0.1000    0.5821
     4        4.2411             nan     0.1000    0.4973
     5        3.7982             nan     0.1000    0.4185
     6        3.4171             nan     0.1000    0.3776
     7        3.0993             nan     0.1000    0.3102
     8        2.8123             nan     0.1000    0.2834
     9        2.5698             nan     0.1000    0.2457
    10        2.3682             nan     0.1000    0.1951
    20        1.2060             nan     0.1000    0.0631
    40        0.6505             nan     0.1000    0.0103
    60        0.4897             nan     0.1000    0.0023
    80        0.4039             nan     0.1000    0.0013
   100        0.3539             nan     0.1000    0.0010
   120        0.3231             nan     0.1000    0.0006
   140        0.3056             nan     0.1000    0.0001
   160        0.2926             nan     0.1000   -0.0001
   180        0.2857             nan     0.1000   -0.0002
   200        0.2791             nan     0.1000    0.0003
   220        0.2734             nan     0.1000   -0.0000
   240        0.2692             nan     0.1000   -0.0002
   260        0.2639             nan     0.1000    0.0002
   280        0.2602             nan     0.1000   -0.0001
   300        0.2566             nan     0.1000    0.0000
   320        0.2541             nan     0.1000   -0.0006
   340        0.2518             nan     0.1000    0.0001
   360        0.2489             nan     0.1000   -0.0003
   380        0.2468             nan     0.1000   -0.0001
   400        0.2443             nan     0.1000   -0.0000
   420        0.2421             nan     0.1000   -0.0002
   440        0.2400             nan     0.1000   -0.0001
   460        0.2387             nan     0.1000   -0.0001
   480        0.2372             nan     0.1000   -0.0001
   500        0.2356             nan     0.1000   -0.0001
```

```r
summary(gbm_model)
```



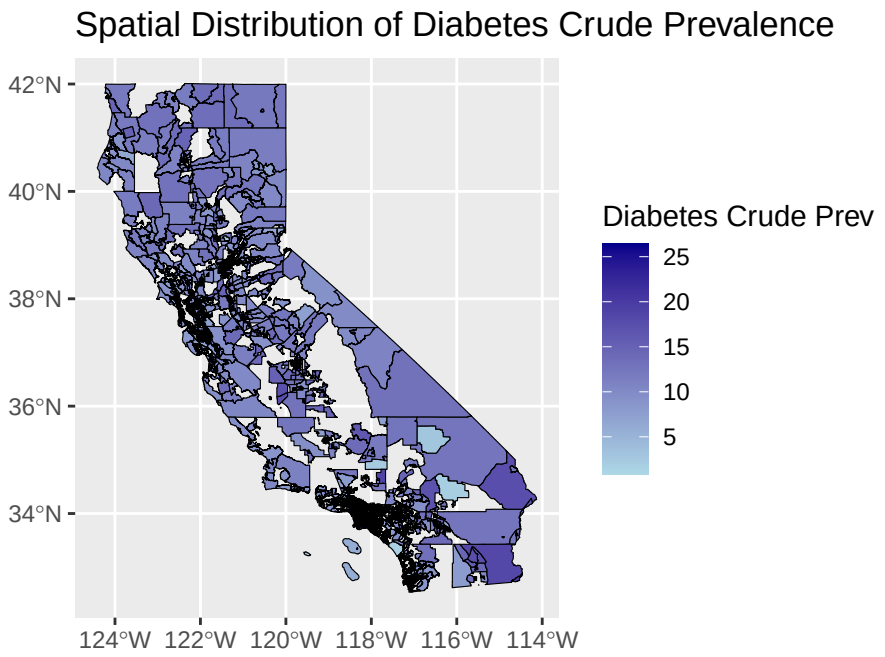

```
                                  var   rel.inf
LPA_CrudePrev           LPA_CrudePrev 63.122269
BPHIGH_CrudePrev     BPHIGH_CrudePrev 32.610751
CSMOKING_CrudePrev CSMOKING_CrudePrev  2.708896
OBESITY_CrudePrev   OBESITY_CrudePrev  1.558084
```

```
plot(gbm_model$cv.fitted)
```

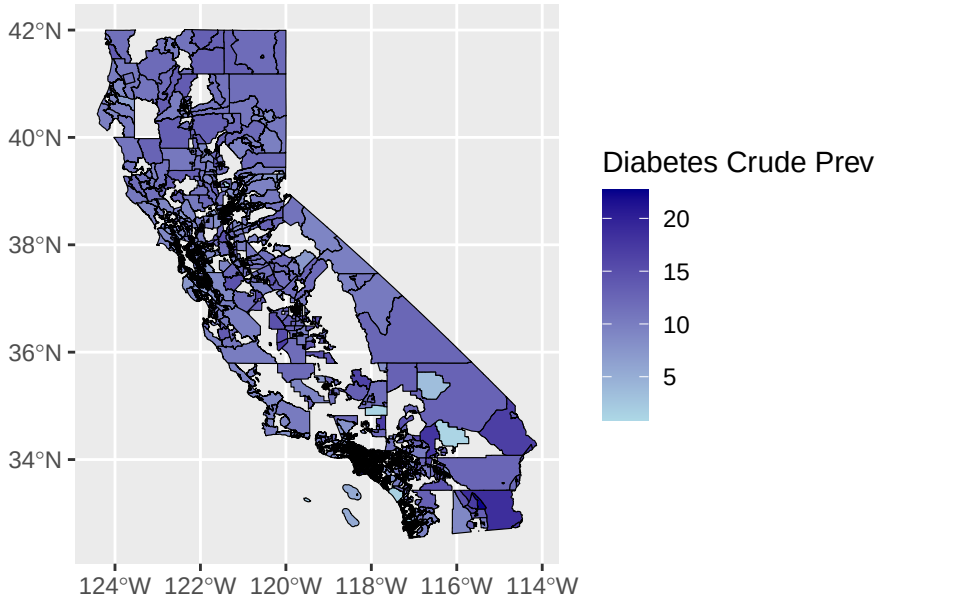


Plots for RFM and GBM

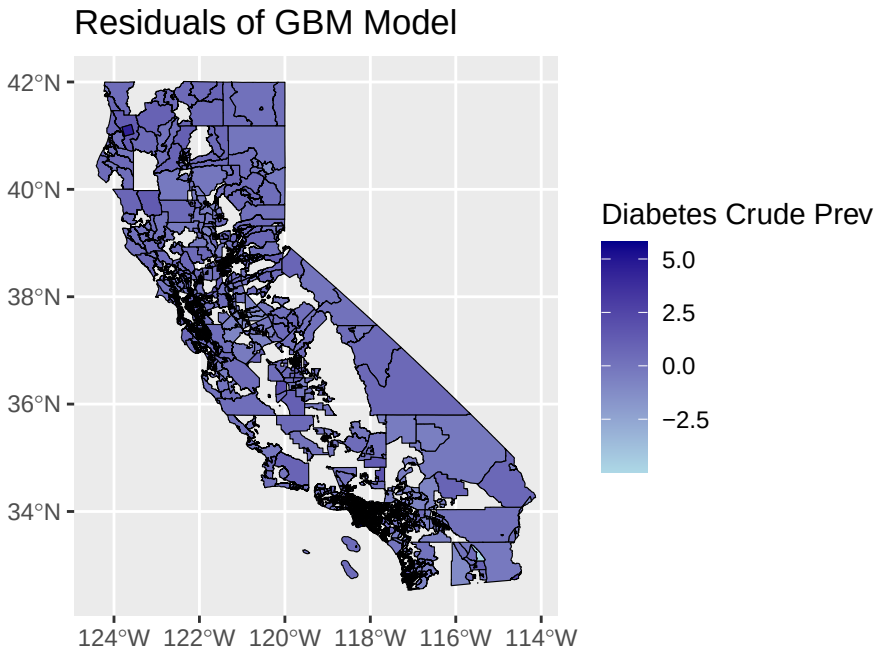
```
merged_sf$gbmModelValue <- gbm_model$cv.fitted
merged_sf$gbmResiduals <- merged_sf$DIABETES_CrudePrev-merged_sf$gbmModelValue
ggplot() +
  geom_sf(data = merged_sf, aes(fill = DIABETES_CrudePrev), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Spatial Distribution of Diabetes Crude Prevalence")
```




```
ggplot() +
  geom_sf(data = merged_sf, aes(fill = gbmModelValue), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Spatial Distribution of Diabetes Crude Prevalence us Gradient Boosted Model")
```



```r
ggplot() +
  geom_sf(data = merged_sf, aes(fill = gbmResiduals), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Residuals of GBM Model")
```



Summary of Plots

Random Forest Model:

The Random Forest model's spatial plot suggest the highest rates are concentrated in the south. The variation in prevalence across counties may imply that local factors, potentially including lifestyle, access to healthcare, or socio-economic conditions, significantly influence diabetes rates. The coastal regions show a lower prevalence, hinting at possible variations in health behaviors or environmental factors. The model seems to effectively capture these regional differences, which offers detailed insights.

Residuals of Random Forest Model:

The residuals from the Random Forest model indicate a generally well-fitting model since most residuals are clustered around zero. However, the presence of outliers with large residuals suggests that there are areas where the model doesn't align perfectly with the observed data. This could potentially be due to factors not accounted for in the model. These outliers highlight specific regions where additional variables or model adjustments may be necessary to improve predictive accuracy.

Gradient Boosted Model:

The Gradient Boosted Model shows a tendency to smooth out the spatial variability seen in the Random Forest outputs. While this may imply a less nuanced capture of local occurrence, it could also indicate a more generalized understanding of the data, which might be useful for broader public health planning. The GBM's residuals also display a good fit overall but, but like the Random Forest model, shows room for refinement in capturing local variance.

Comparative Analysis:

When comparing the two models, the Random Forest appears to describe finer spatial details of diabetes prevalence, while the Gradient Boosted Model provides a more generalized prediction across the state. The choice between these models may depend on the specific needs of the analysis—if the goal is to understand detailed local patterns, then the Random Forest might probably be preferable. Contrarily, for more general

trends, the GBM could satisfy demands. Both models, however, point to the multifactorial nature of diabetes prevalence and underscore areas where public health efforts might be targeted to reduce it.