

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, OBESITY_CrudePrev)

# 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")
```

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

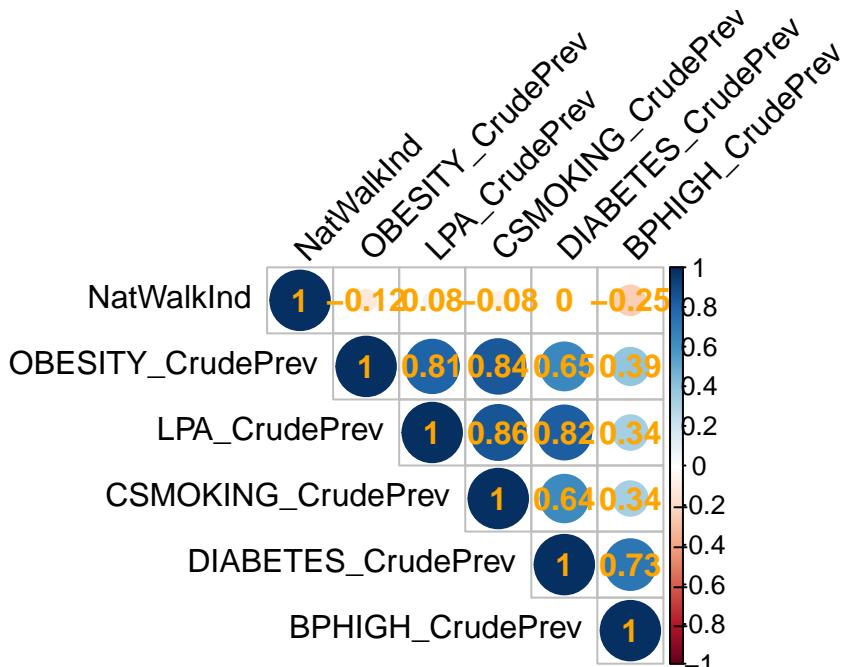
# Output variable names (column names) used for correlation matrix
cat("Variables used for correlation matrix:\n")
```

Variables used for correlation matrix:

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

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

```
# 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

```
library(GWmodel)
```

Loading required package: robustbase

Loading required package: sp

Loading required package: Rcpp

Welcome to GWmodel version 2.3-2.

```
library(sf)
```

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

```
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/ca_tract.shp'  
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
```

```
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 + CSMC_CrudePrev,  
                           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 + CSMOKING_CrudePrev,
                           data = merged_spatialdf,
                           bw = merged_gwr_bw,
                           kernel = "exponential",
                           )

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

```

```
summary(gwr_results_sf)
```

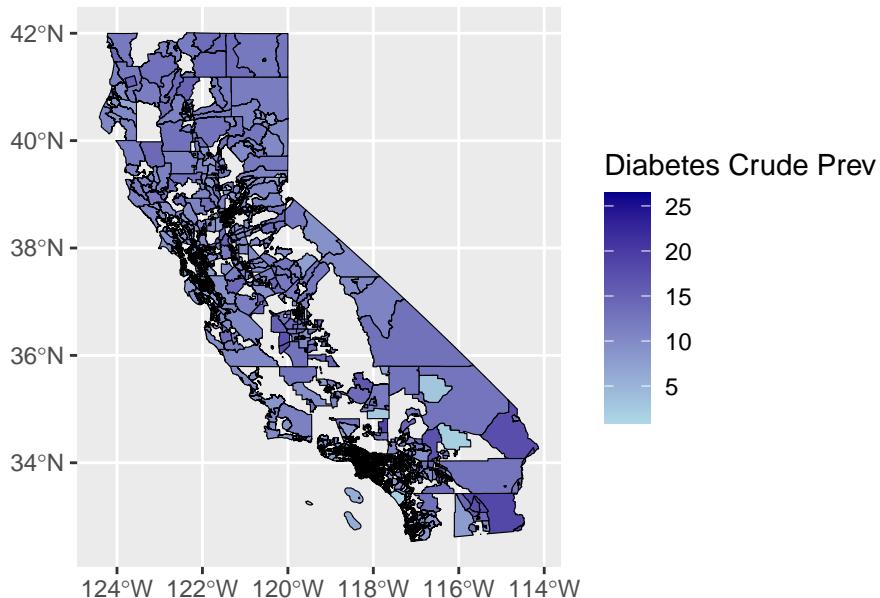
	Intercept	OBESITY_CrudePrev	BPHIGH_CrudePrev	LPA_CrudePrev
Min.	-50.449	-8.31242	-0.1448	-0.3606
1st Qu.	-5.239	-0.04769	0.2927	0.2544
Median	-4.242	-0.01005	0.3209	0.3309
Mean	-4.648	0.03475	0.3233	0.3126
3rd Qu.	-3.258	0.09442	0.3517	0.3830
Max.	52.917	2.87757	2.6076	1.4275
CSMOKING_CrudePrev	y	yhat	residual	
Min.	-3.5116	0.900	-22.266	-80.69782
1st Qu.	-0.3397	8.100	8.128	-0.09385
Median	-0.2447	9.600	9.649	0.00213
Mean	-0.2219	9.913	9.917	-0.00378
3rd Qu.	-0.1193	11.500	11.535	0.11349
Max.	5.1472	26.400	92.998	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
```

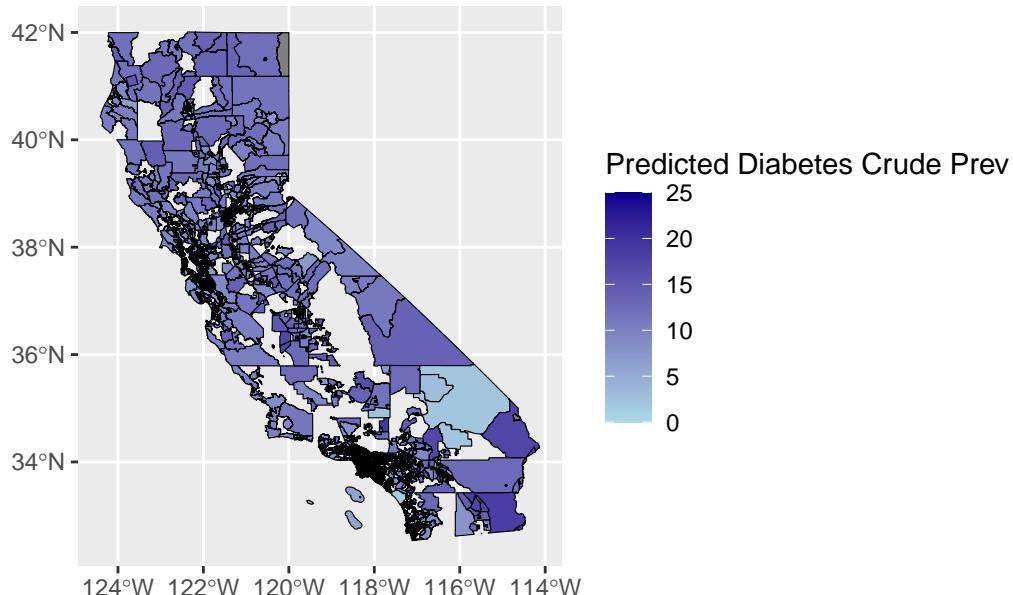
```
ggplot() +
  geom_sf(data = merged_sf, aes(fill = DIABETES_CrudePrev), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Spatial Distribution of Diabetes Crude Prevalence")
```

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")
```

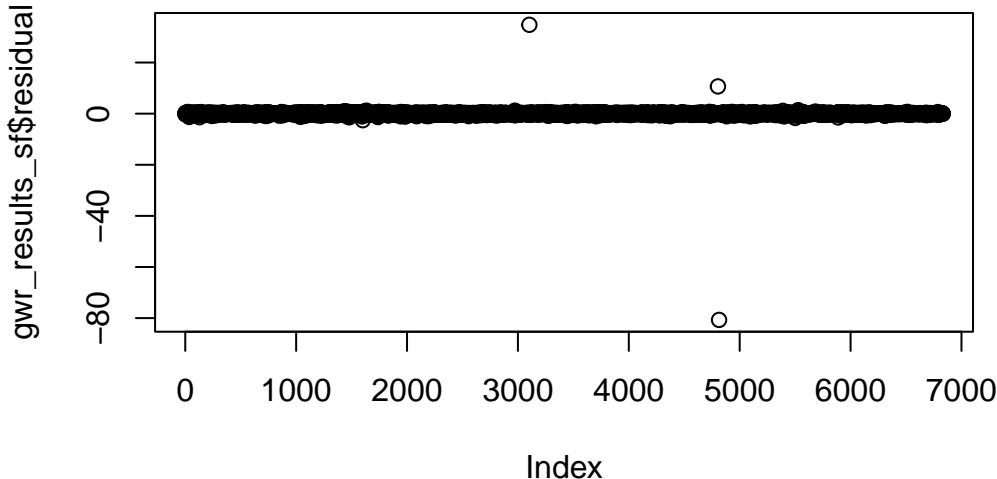
GWR Predicted Diabetes Crude Prevalence using GWR



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':
```

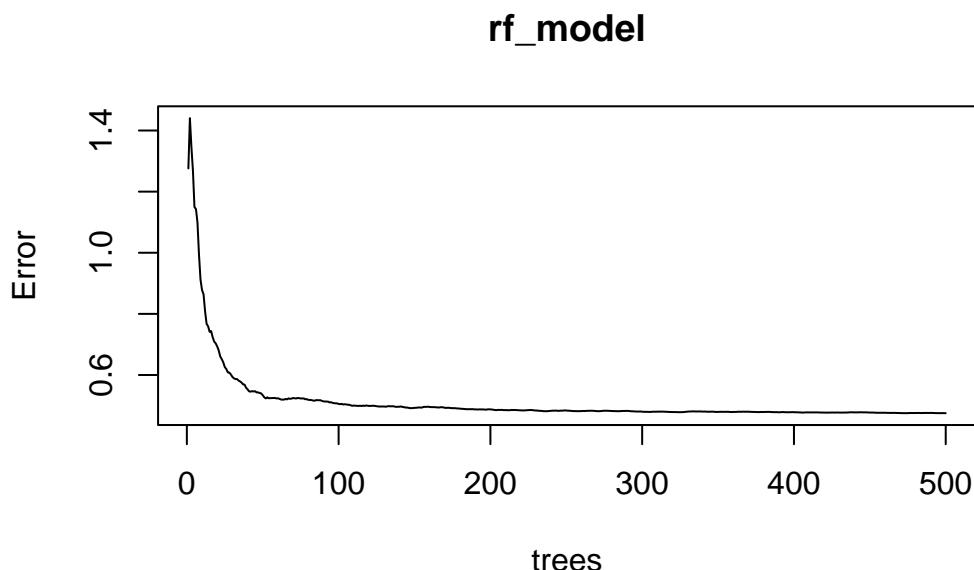
```
combine
```

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

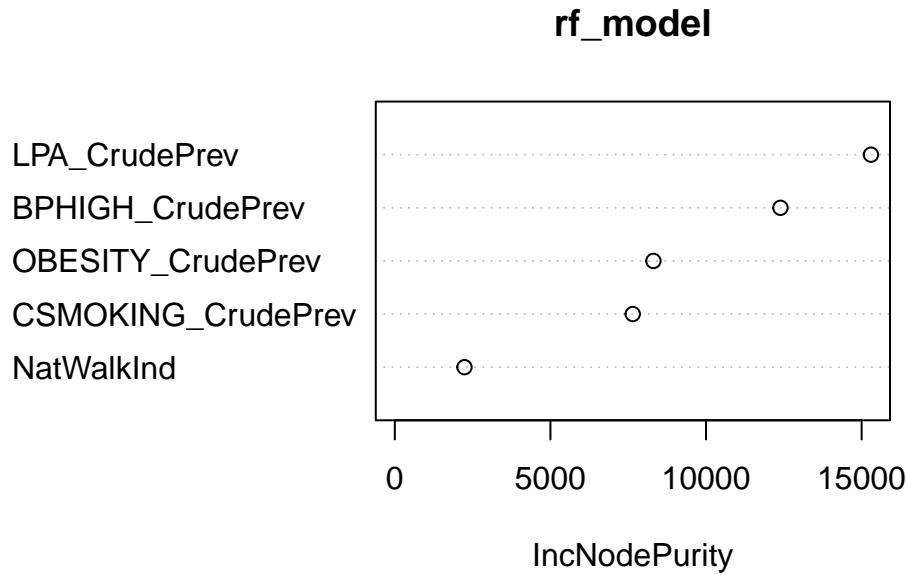
rf_model <- randomForest(DIABETES_CrudePrev ~ NatWalkInd+OBESITY_CrudePrev + BPHIGH_CrudePrev + LPA_CrudePrev)
print(rf_model)
```

Call:
randomForest(formula = DIABETES_CrudePrev ~ NatWalkInd + OBESITY_CrudePrev + BPHIGH_CrudePrev + LPA_CrudePrev)
Type of random forest: regression
Number of trees: 500
No. of variables tried at each split: 1
Mean of squared residuals: 0.4749132
% Var explained: 93.07

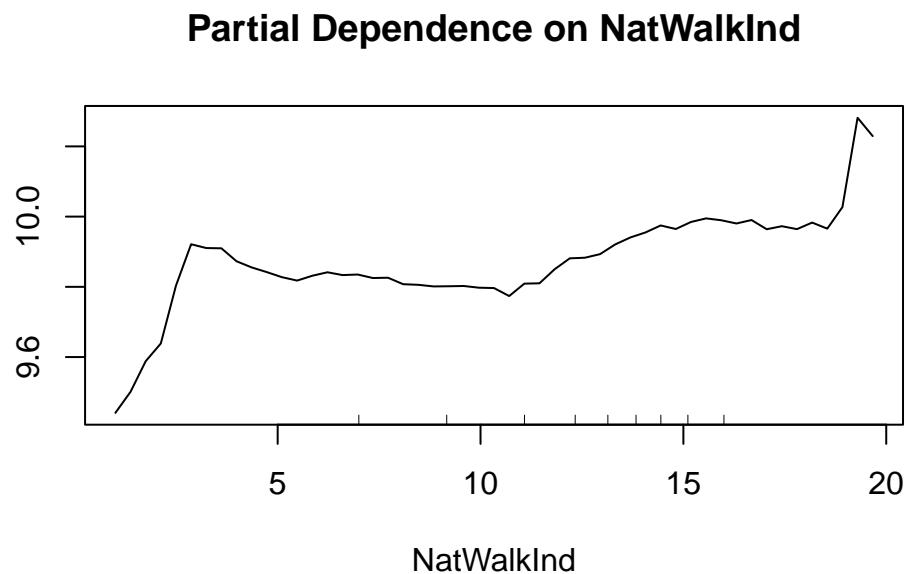
```
plot(rf_model)
```



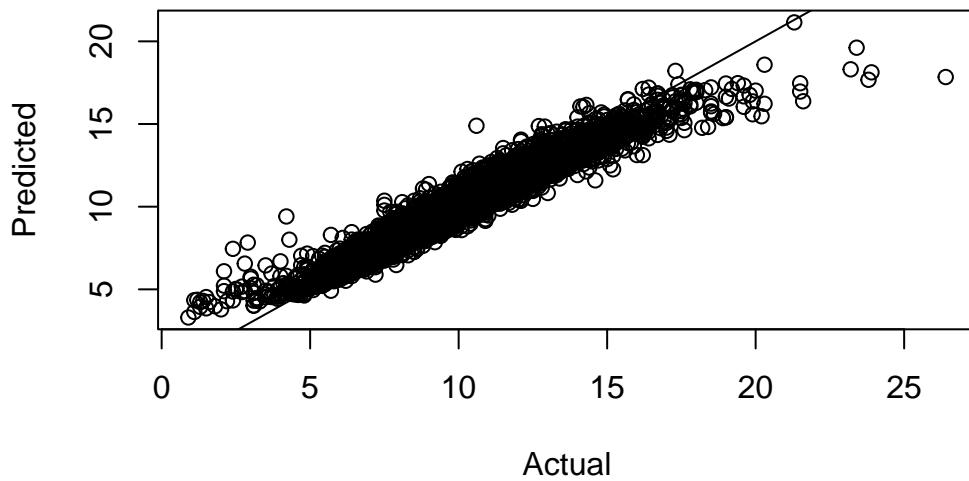
```
varImpPlot(rf_model)
```



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



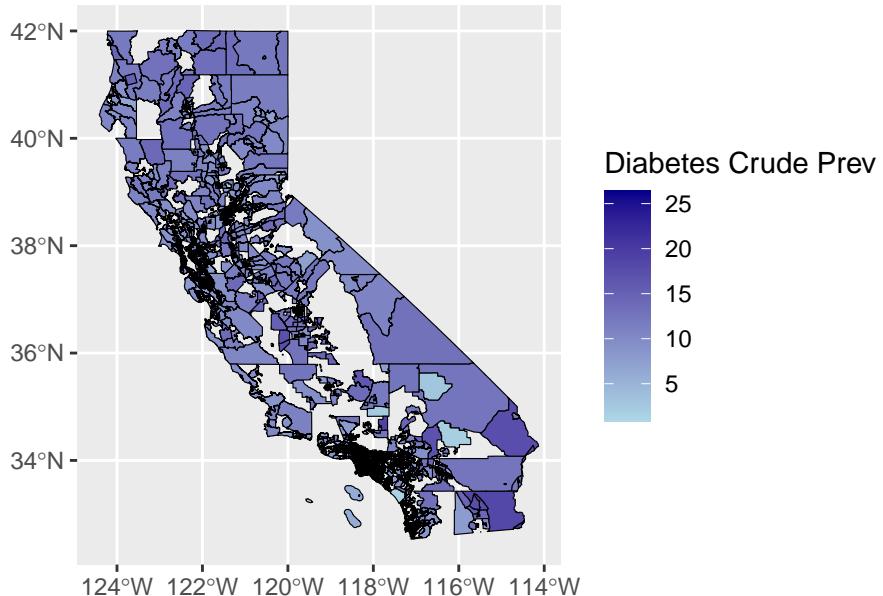
```
plot(data_df$DIABETES_CrudePrev, predict(rf_model), xlab = "Actual", ylab = "Predicted")  
abline(0,1)
```



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

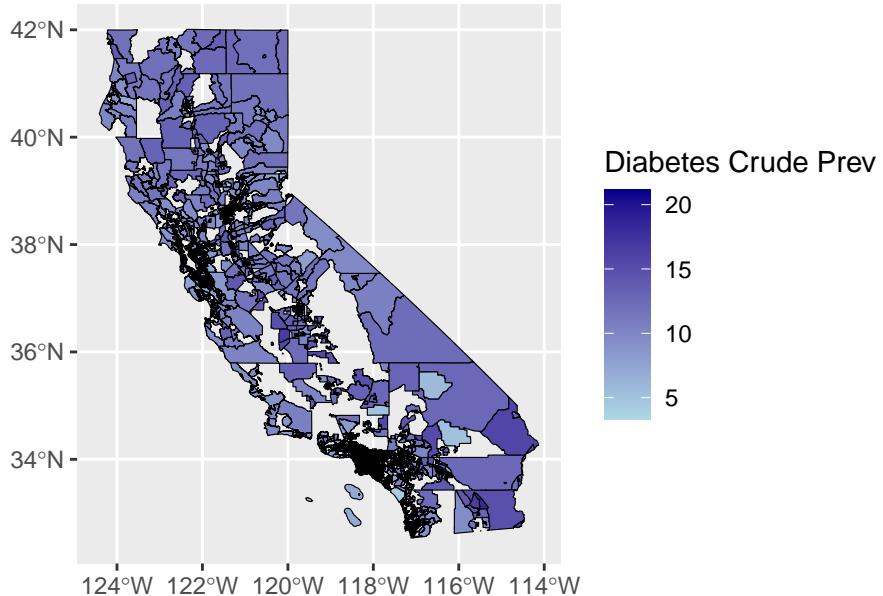
```
ggplot() +
  geom_sf(data = merged_sf, aes(fill = DIABETES_CrudePrev), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Spatial Distribution of Diabetes Crude Prevalence")
```

Spatial Distribution of Diabetes Crude Prevalence



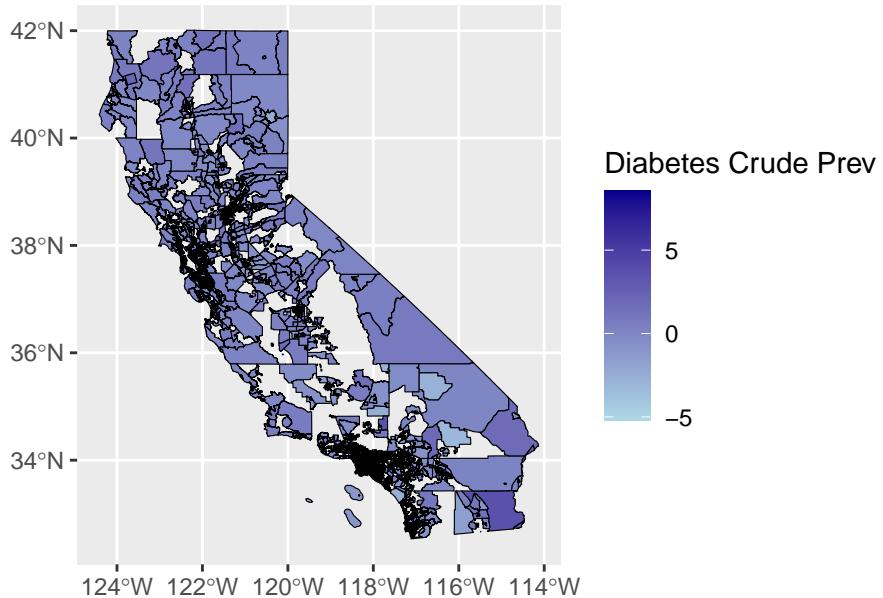
```
ggplot() +
  geom_sf(data = merged_sf, aes(fill = rfModelValues), color="black",size=0.2) +
  scale_fill_gradient(name = "Diabetes Crude Prev", low = "lightblue", high = "darkblue") +
  labs(title = "Spatial Distribution of Diabetes Crude Prevalence us Random Forest Model")
```

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")
```

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/gbm3/gbm3>

```

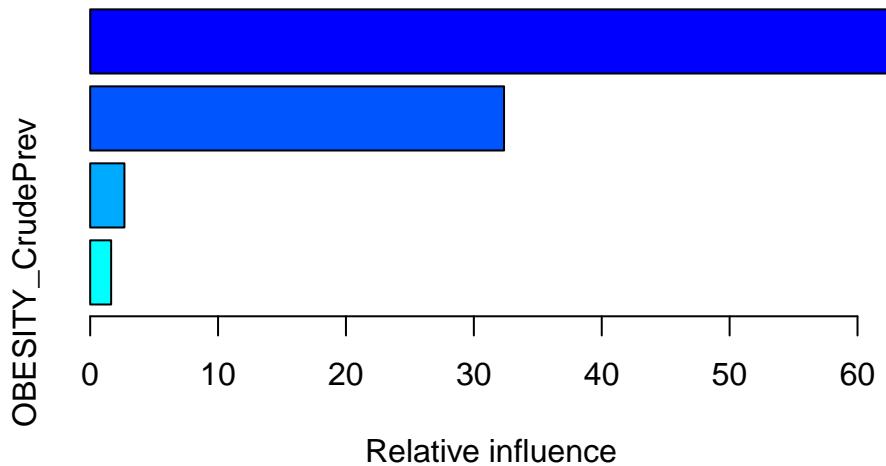
merged_spatialdf <- as(merged_sf, "Spatial")
data_df <- as.data.frame(merged_spatialdf)

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

```

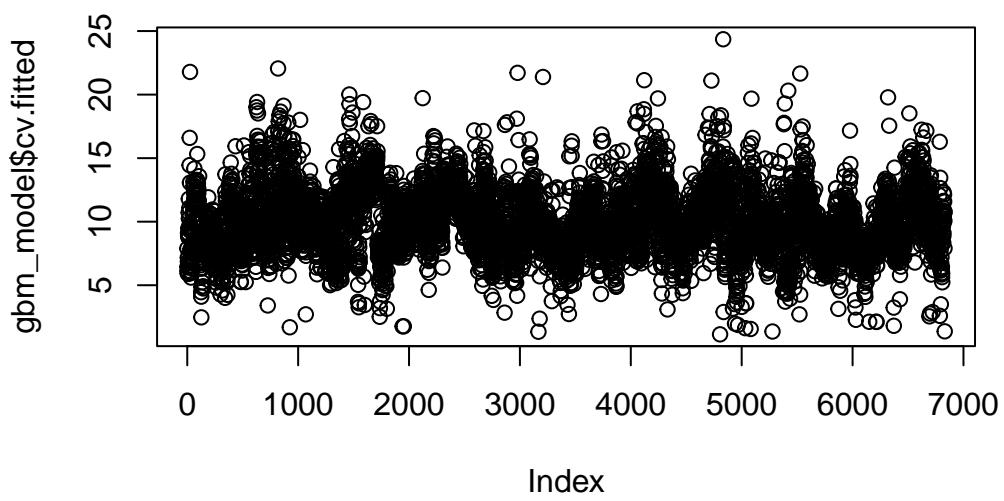
Iter	TrainDeviance	ValidDeviance	StepSize	Improve
1	6.0156	nan	0.1000	0.8196
2	5.3086	nan	0.1000	0.6942
3	4.7231	nan	0.1000	0.5872
4	4.2375	nan	0.1000	0.4758
5	3.8086	nan	0.1000	0.4207
6	3.4405	nan	0.1000	0.3712
7	3.1290	nan	0.1000	0.3072
8	2.8438	nan	0.1000	0.2813
9	2.6058	nan	0.1000	0.2188
10	2.3929	nan	0.1000	0.2053
20	1.1906	nan	0.1000	0.0678
40	0.6367	nan	0.1000	0.0099
60	0.4783	nan	0.1000	0.0069
80	0.3946	nan	0.1000	0.0022
100	0.3483	nan	0.1000	0.0009
120	0.3187	nan	0.1000	0.0014
140	0.3016	nan	0.1000	-0.0005
160	0.2873	nan	0.1000	0.0001
180	0.2815	nan	0.1000	-0.0002
200	0.2759	nan	0.1000	-0.0002
220	0.2705	nan	0.1000	0.0001
240	0.2664	nan	0.1000	-0.0003
260	0.2623	nan	0.1000	-0.0002
280	0.2591	nan	0.1000	0.0001
300	0.2560	nan	0.1000	-0.0001
320	0.2532	nan	0.1000	-0.0004
340	0.2507	nan	0.1000	-0.0004
360	0.2485	nan	0.1000	-0.0004
380	0.2463	nan	0.1000	-0.0003
400	0.2440	nan	0.1000	-0.0001
420	0.2420	nan	0.1000	0.0000
440	0.2404	nan	0.1000	-0.0001
460	0.2388	nan	0.1000	0.0001
480	0.2373	nan	0.1000	-0.0001
500	0.2358	nan	0.1000	-0.0004

```
summary(gbm_model)
```



	var	rel.inf
LPA_CrudePrev	LPA_CrudePrev	63.323407
BPHIGH_CrudePrev	BPHIGH_CrudePrev	32.359800
CSMOKING_CrudePrev	CSMOKING_CrudePrev	2.677051
OBESITY_CrudePrev	OBESITY_CrudePrev	1.639743

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



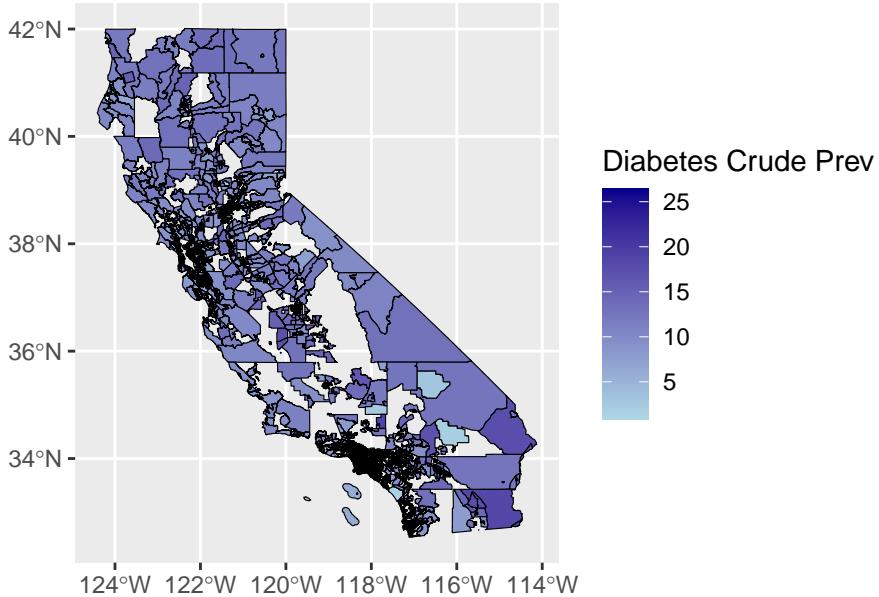
Plots for RFM and GBM

```

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")

```

Spatial Distribution of Diabetes Crude Prevalence

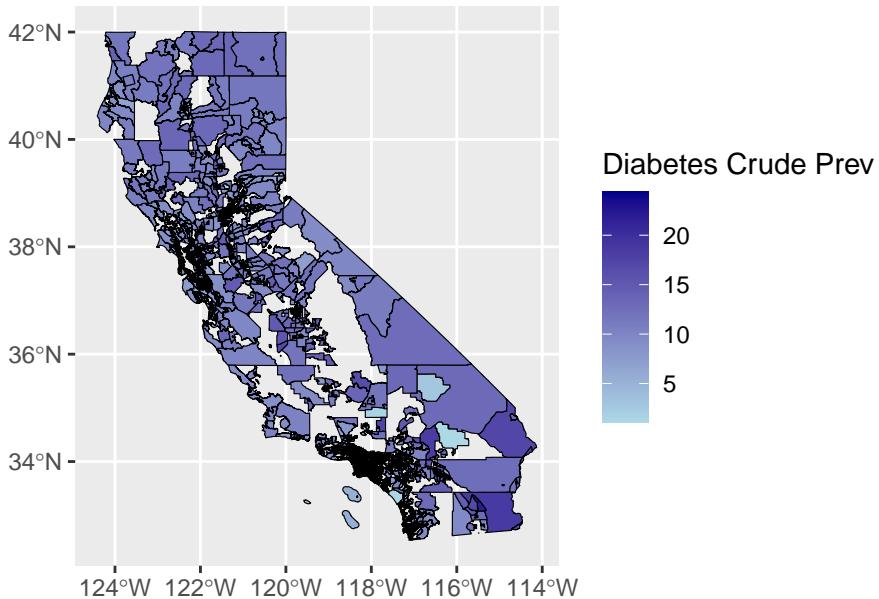


```

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")

```

Spatial Distribution of Diabetes Crude Prevalence us Gradient Boosted Model

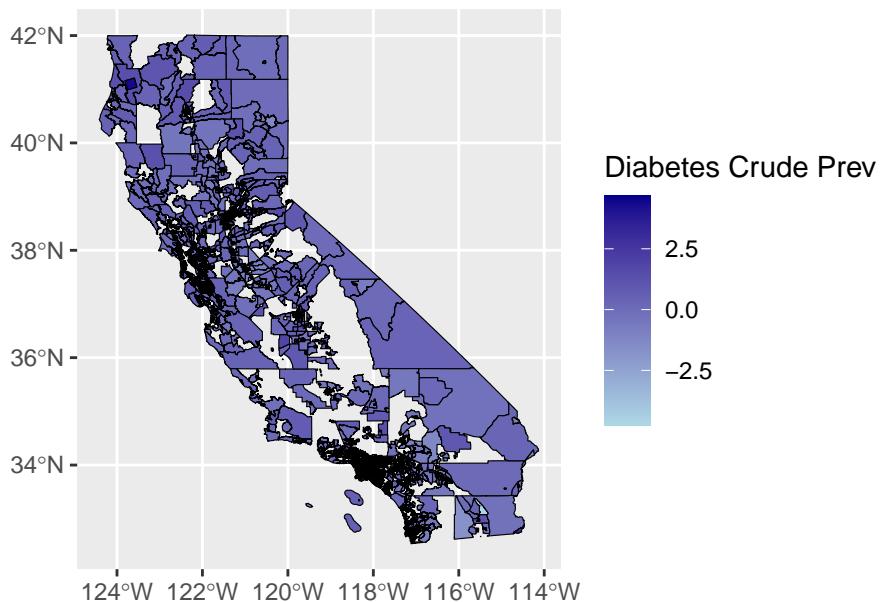


```

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")

```

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, like the Random Forest model, shows room for refinement in capturing local variance.

Comparative Analysis:

The Geographically Weighted Regression (GWR) model effectively captures local variations in diabetes prevalence, highlighting the impact of regional factors. In comparison, the Gradient Boosted Model (GBM) provides a more generalized overview, potentially smoothing over local nuances. GWR's strength lies in its

spatially detailed predictions, while GBM's smoothed predictions may be beneficial for broader trend analysis. For public health strategies, GWR's detailed spatial insights could be more preferred. On the other hand, the GBM's general predictions across larger areas could be more appropriate for overarching health policies.