

# Statewide Suburban Trends

John McCormick

2023-11-14

## Unemployment in Ohio, Minnesota and Michigan Suburbs

```
load("totalPopulationPreliminary.RData")

load("OH_MI_MN_UnemploymentData.RData")

all_years <- seq(1990, 2020)

# Function to interpolate data for one county
interpolate_county_data <- function(subset_data) {
  # Use 'approx' for linear interpolation of population
  interpolated <- approx(subset_data$YEAR, subset_data$totalPop, xout = all_years)

  # Return a data frame with interpolated values
  data.frame(County = unique(subset_data$County),
             State = unique(subset_data$State),
             Year = all_years,
             Population = interpolated$y)
}

population <- data.frame(data) %>%
  mutate(State = case_when(STATE == "Ohio" ~ "OH",
                           STATE == "Minnesota" ~ "MN",
                           STATE == "Michigan" ~ "MI")) %>%
  mutate(County = COUNTY) %>%
  filter(YEAR %in% c(1990, 2000, 2010, 2020))

# Apply the interpolation function to each county/state combination
interpolated_data <- population %>%
  group_by(County, State) %>%
  do(interpolate_county_data(.))

unemployment <- data.frame(combinedData)
unemployment <- unemployment%>%
  mutate(Date = as.Date(paste0(Label, " 01"), format = "%Y %b %d")) %>%
  filter(State %in% c("OH", "MI", "MN")) %>%
  arrange(Date) %>%
  group_by(State, Date)

merged_data <- full_join(unemployment, population, by = "County")
```

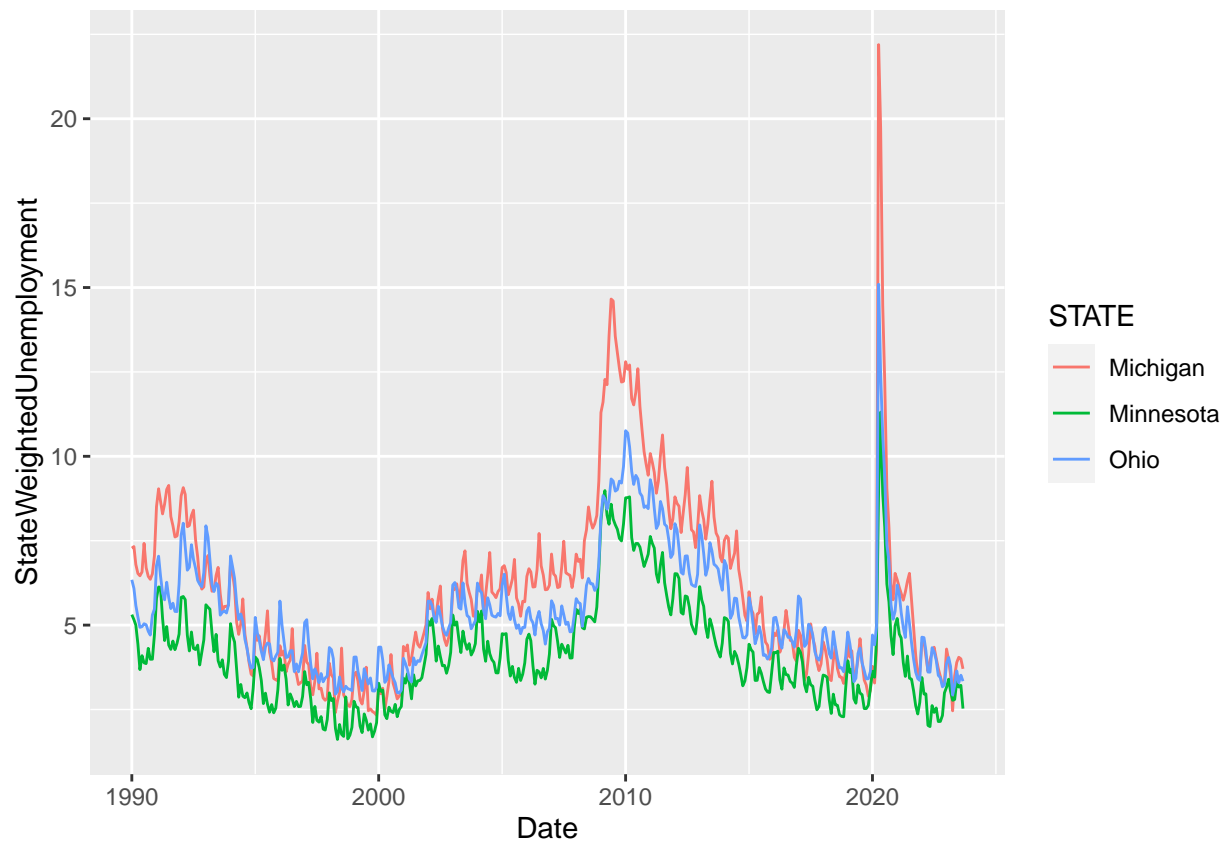
```
## Warning in full_join(unemployment, population, by = "County"): Detected an unexpected many-to-many r
## i Row 1 of 'x' matches multiple rows in 'y'.
## i Row 18 of 'y' matches multiple rows in 'x'.
## i If a many-to-many relationship is expected, set 'relationship =
##   "many-to-many"' to silence this warning.
```

```
merged_data <- merged_data %>%
  group_by(STATE, Date) %>%
  mutate(TotalStatePop = sum(totalPop)) %>%
  ungroup() %>%
  mutate(PopShare = totalPop / TotalStatePop)

# Calculate the weighted average unemployment rate for each state
state_weighted_avg <- merged_data %>%
  group_by(STATE, Date) %>%
  summarize(StateWeightedUnemployment = sum(Value * PopShare), .groups = 'drop')

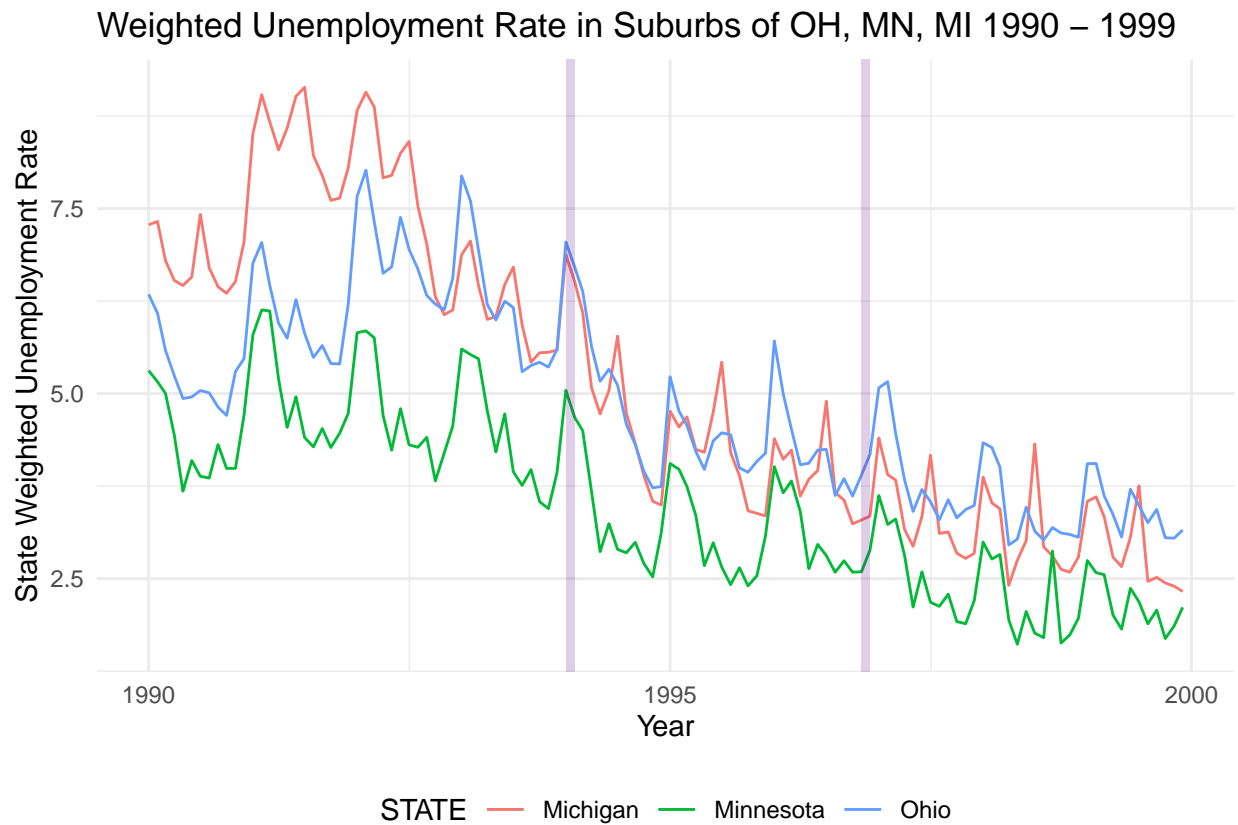
# Convert the results to a data frame
final_data_frame <- data.frame(state_weighted_avg)

ggplot(final_data_frame, aes(Date, StateWeightedUnemployment, color = STATE)) +
  geom_line()
```



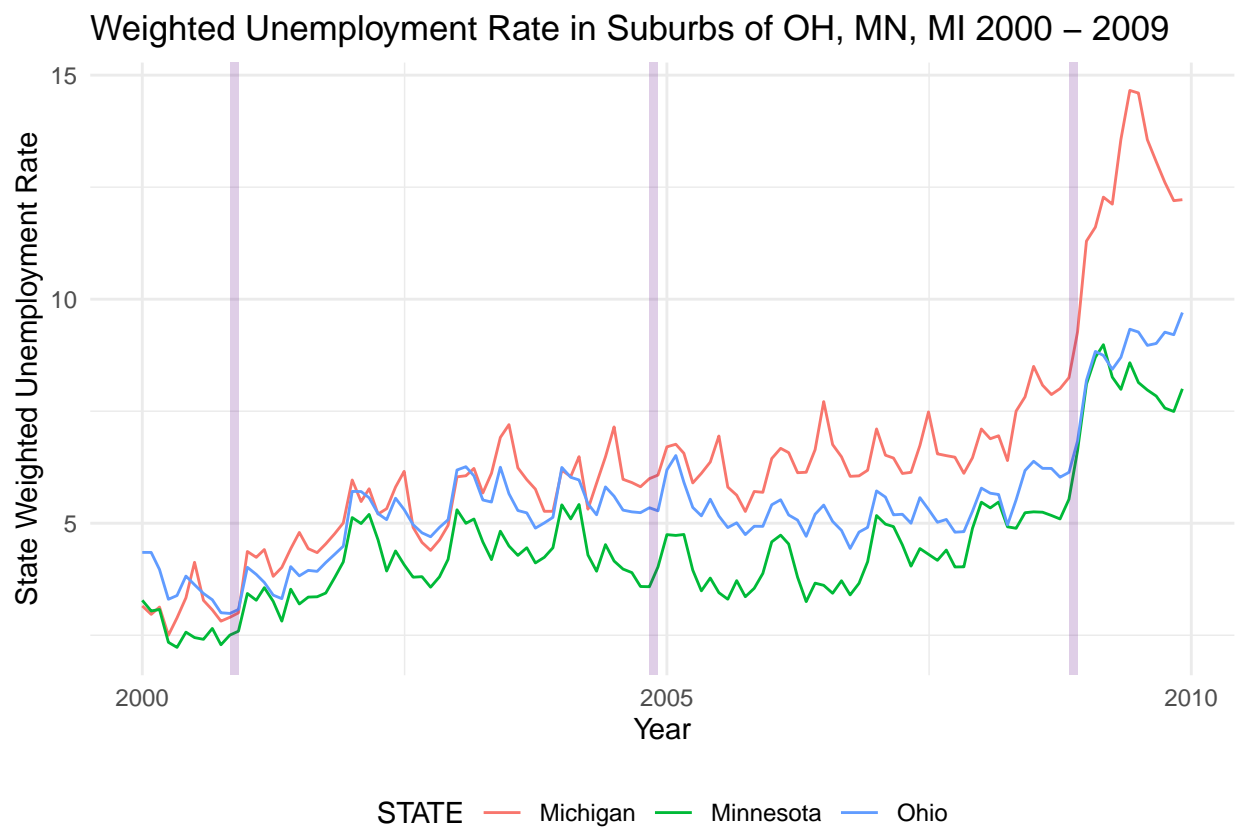
## Looking by decade

```
events <- data.frame(  
  start = as.Date(c("1994-01-01", "1996-11-01")),  
  end = as.Date(c("1994-01-31", "1996-11-30"))  
)  
  
pre2000 <- final_data_frame %>%  
  filter(Date < as.Date("2000-01-01"))  
pre2010 <- final_data_frame %>%  
  filter(Date < as.Date("2010-01-01") & Date > as.Date("1999-12-31"))  
post2010 <- final_data_frame %>%  
  filter(Date > as.Date("2009-12-31"))  
  
ggplot(pre2000, aes(x = Date, StateWeightedUnemployment, color = STATE)) +  
  geom_line() +  
  geom_rect(data = events, aes(xmin = start, xmax = end, ymin = -Inf, ymax = Inf),  
    fill = "#5F0A87", alpha = 0.2, inherit.aes = FALSE) +  
  labs(title = paste("Weighted Unemployment Rate in Suburbs of OH, MN, MI 1990 - 1999"),  
    x = "Year",  
    y = "State Weighted Unemployment Rate") +  
  theme_minimal() +  
  theme(legend.position = "bottom")
```



```
events <- data.frame(
  start = as.Date(c( "2000-11-01", "2004-11-01",
                    "2008-11-01")),
  end = as.Date(c( "2000-11-30", "2004-11-30",
                  "2008-11-30")))

ggplot(pre2010, aes(x = Date, StateWeightedUnemployment, color = STATE)) +
  geom_line() +
  geom_rect(data = events, aes(xmin = start, xmax = end, ymin = -Inf, ymax = Inf),
            fill = "#5F0A87", alpha = 0.2, inherit.aes = FALSE) +
  labs(title = paste("Weighted Unemployment Rate in Suburbs of OH, MN, MI 2000 - 2009"),
       x = "Year",
       y = "State Weighted Unemployment Rate") +
  theme_minimal() +
  theme(legend.position = "bottom")
```



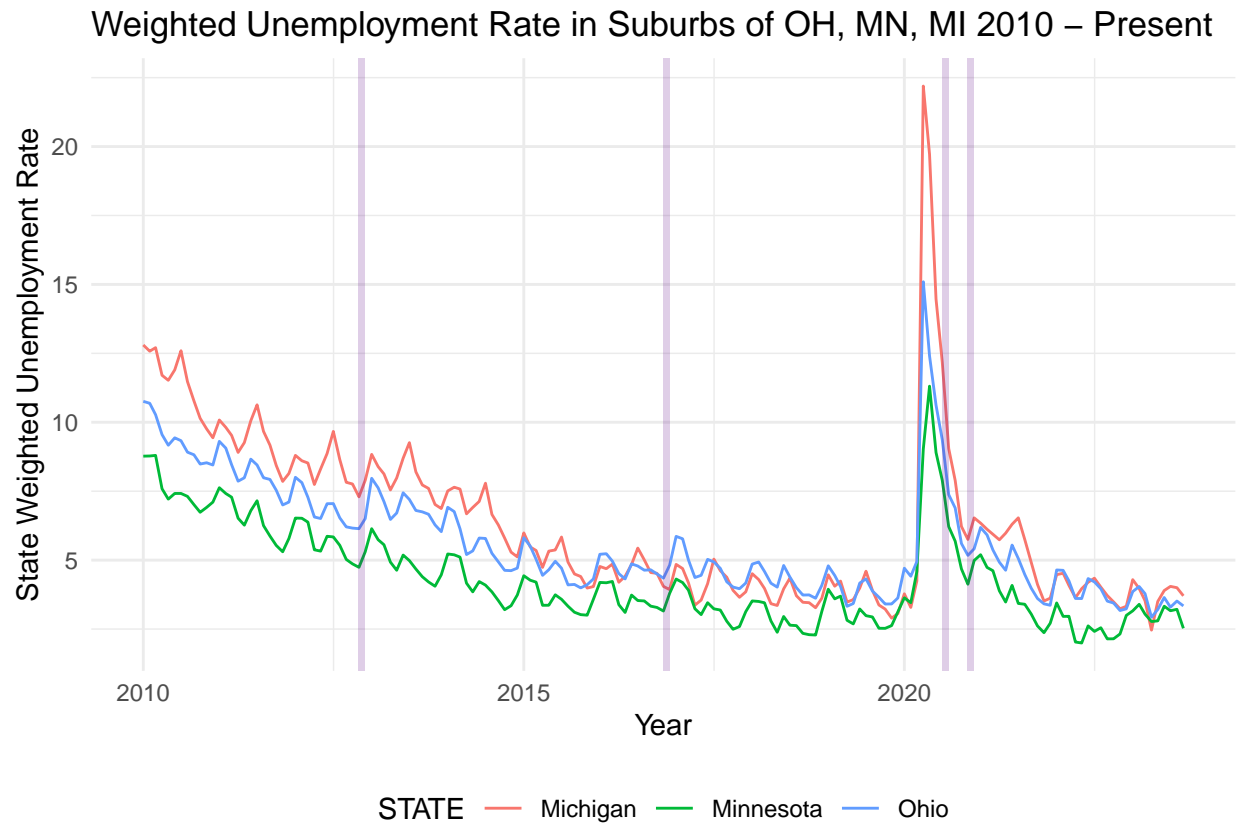
```
events <- data.frame(
  start = as.Date(c("2012-11-01", "2016-11-01", "2020-11-01", "2020-07-01")),
  end = as.Date(c("2012-11-30", "2016-11-30", "2020-11-30", "2020-07-31"))
)

ggplot(post2010, aes(x = Date, StateWeightedUnemployment, color = STATE)) +
  geom_line() +
  geom_rect(data = events, aes(xmin = start, xmax = end, ymin = -Inf, ymax = Inf),
```

```

    fill = "#5F0A87", alpha = 0.2, inherit.aes = FALSE) +
  labs(title = paste("Weighted Unemployment Rate in Suburbs of OH, MN, MI 2010 - Present"),
    x = "Year",
    y = "State Weighted Unemployment Rate") +
  theme_minimal() +
  theme(legend.position = "bottom")

```



## Looking at Voting Trends in 3 states suburbs

```

load("Presidential Votes County 2000-2020.RData")

ohioCounties = c("Lorain", "Medina", "Summit", "Geauga", "Lake", "Delaware", "Union", "Licking", "Farif

michiganCounties = c("Monroe", "Lenawee", "Jackson", "Ingham", "Livingston", "Macomb", "Washtenaw")

minnesotaCounties = c("Dakota", "Scott", "Carver", "Wright", "Sherburne", "Anoka", "Chisago")

michiganData <- data %>%
  mutate(county_name = str_replace(county_name, " County", "")) %>%
  filter(state_name == "Michigan", county_name %in% michiganCounties) %>%
  mutate(republicanPercent = votes_gop / total_votes) %>%
  mutate(democratPercent = votes_dem / total_votes) %>%
  mutate(repUpPoints = republicanPercent - democratPercent)

```

```

minnesotaData <- data %>%
  mutate(county_name = str_replace(county_name, " County", "")) %>%
  filter(state_name == "Minnesota", county_name %in% minnesotaCounties) %>%
  mutate(republicanPercent = votes_gop / total_votes) %>%
  mutate(democratPercent = votes_dem / total_votes) %>%
  mutate(repUpPoints = republicanPercent - democratPercent)

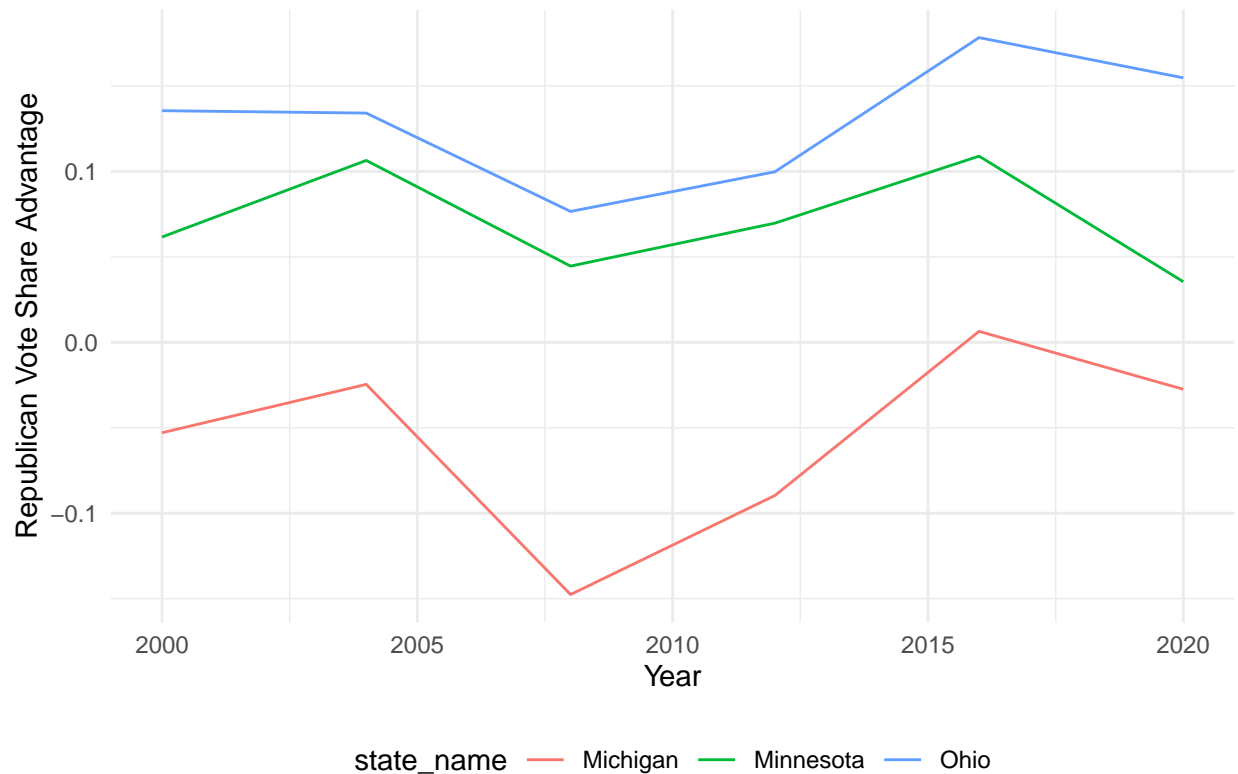
ohioData <- data %>%
  mutate(county_name = str_replace(county_name, " County", "")) %>%
  filter(state_name == "Ohio", county_name %in% ohioCounties) %>%
  mutate(republicanPercent = votes_gop / total_votes) %>%
  mutate(democratPercent = votes_dem / total_votes) %>%
  mutate(repUpPoints = republicanPercent - democratPercent)
vote_data <- bind_rows(michiganData, minnesotaData, ohioData)

state_vote_data <- vote_data %>%
  group_by(state_name, year) %>%
  summarize(TotalRepublicanVotes = sum(votes_gop),
            TotalStateVotes = sum(total_votes),
            RepublicanVoteShare = TotalRepublicanVotes / TotalStateVotes,
            TotalDemocraticVotes = sum(votes_dem),
            DemocraticVoteShare = TotalDemocraticVotes / TotalStateVotes,
            VoteDifference = RepublicanVoteShare - DemocraticVoteShare,
            .groups = 'drop')

ggplot(state_vote_data, aes(year, VoteDifference, color = state_name)) +
  geom_line() +
  labs(title = paste("Republican Vote Share Advantage in OH, MI, MN Suburban Counties"),
       x = "Year",
       y = "Republican Vote Share Advantage") +
  theme_minimal() +
  theme(legend.position = "bottom")

```

## Republican Vote Share Advantage in OH, MI, MN Suburban Counties



## Personal Income

```
load("IncomeByCounty1990_2020.RData")

weighted_income <- incomeData %>%
  group_by(State, Year) %>%
  summarize(
    TotalIncome = sum(PerCapitaIncome * Population),
    TotalPopulation = sum(Population),
    WeightedAverageIncome = TotalIncome / TotalPopulation,
    WeightedAverageIncomeThousands = WeightedAverageIncome / 1000,
    .groups = 'drop' # Add this line to drop the grouping
  )

weighted_income1 <- weighted_income %>%
  filter(Year < 2000) %>%
  mutate(Year = as.Date(paste0(Year, "-01-01")))

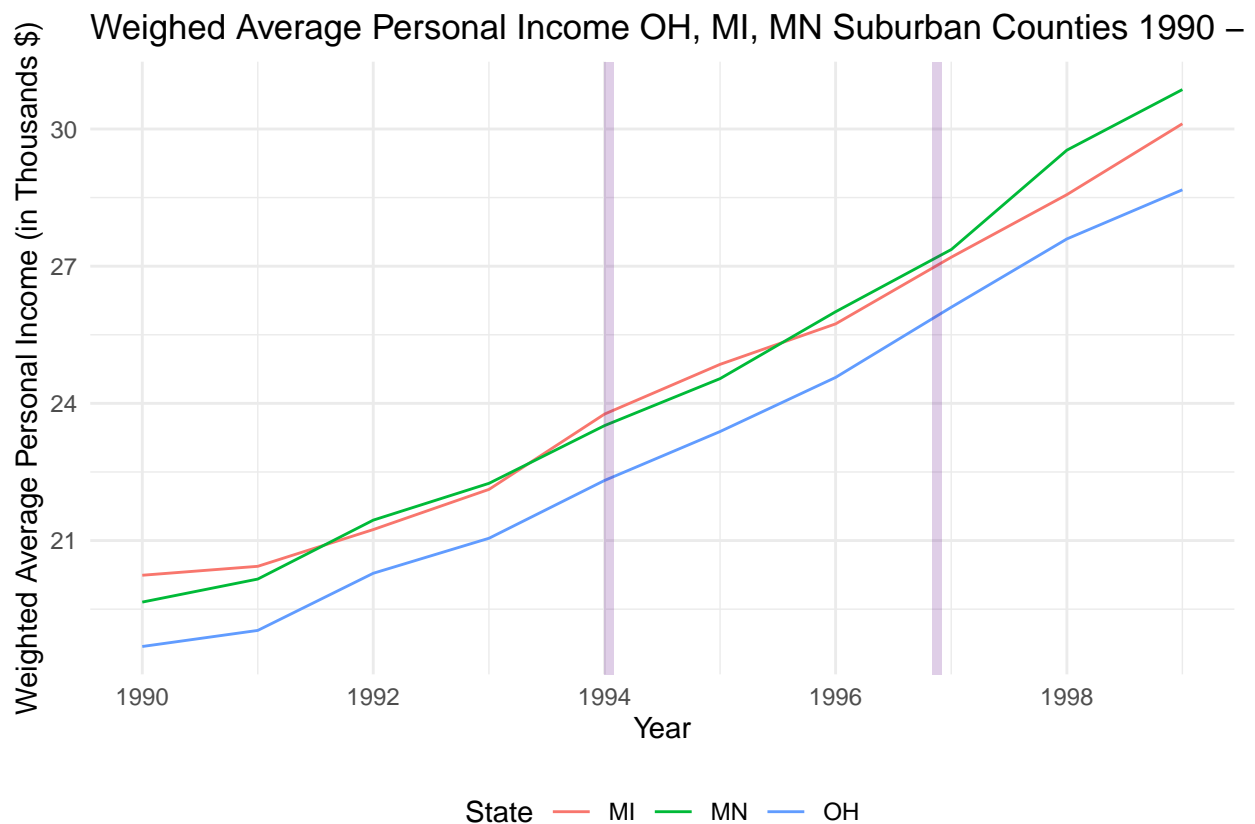
weighted_income2 <- weighted_income %>%
  filter(Year > 1999 & Year < 2010) %>%
  mutate(Year = as.Date(paste0(Year, "-01-01")))

weighted_income3 <- weighted_income %>%
  filter(Year > 2009) %>%
```

```
mutate(Year = as.Date(paste0(Year, "-01-01")))

events <- data.frame(
  start = as.Date(c("1994-01-01", "1996-11-01")),
  end = as.Date(c("1994-01-31", "1996-11-30")))

ggplot(weighted_income1, aes(x=Year, y=WeightedAverageIncomeThousands, color = State, group = State)) +
  geom_line() +
  geom_rect(data = events, aes(xmin = start, xmax = end, ymin = -Inf, ymax = Inf),
            fill = "#5F0A87", alpha = 0.2, inherit.aes = FALSE) +
  labs(title = paste("Weighed Average Personal Income OH, MI, MN Suburban Counties 1990 - 1999"),
       x = "Year",
       y = "Weighted Average Personal Income (in Thousands $)") +
  theme_minimal() +
  theme(legend.position = "bottom")
```



```
events <- data.frame(
  start = as.Date(c("2000-11-01", "2004-11-01",
                    "2008-11-01")),
  end = as.Date(c("2000-11-30", "2004-11-30",
                    "2008-11-30")))

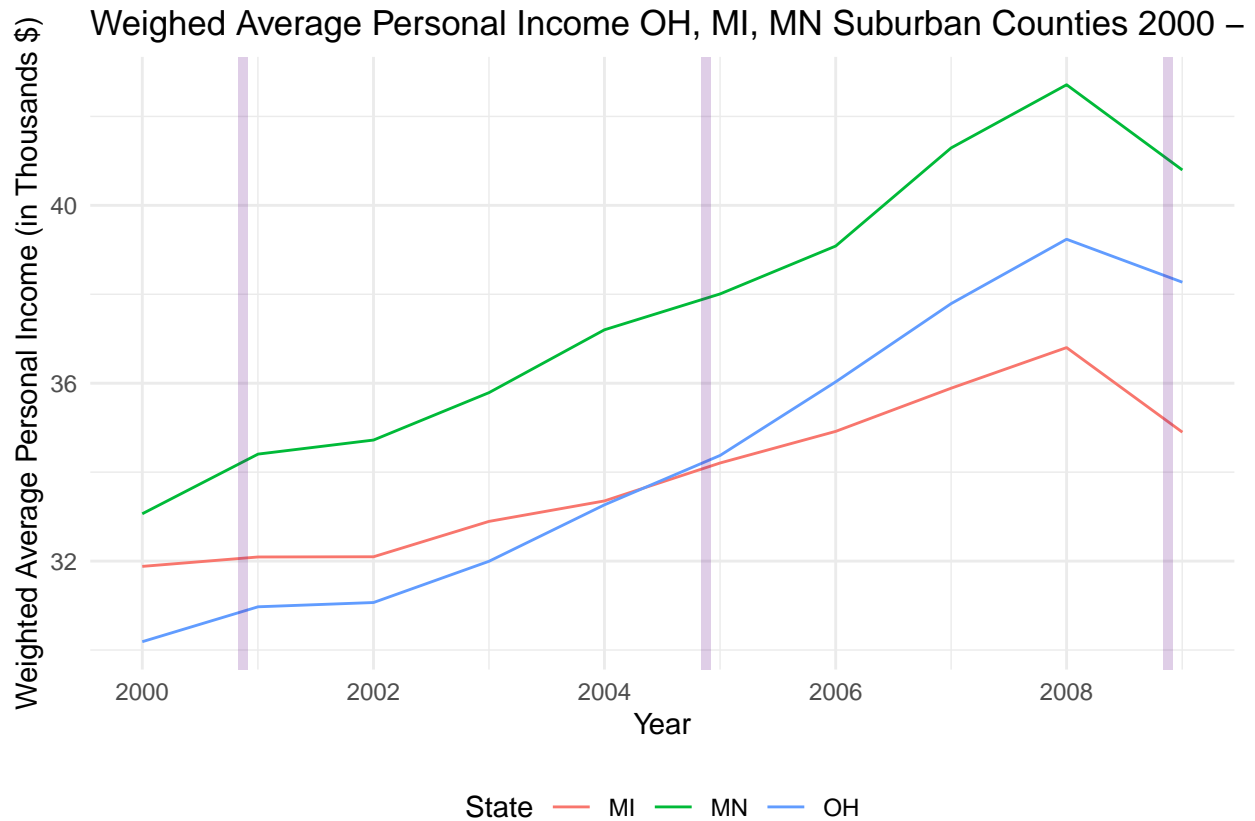
ggplot(weighted_income2, aes(x=Year, y=WeightedAverageIncomeThousands, color = State, group = State)) +
  geom_line() +
  geom_rect(data = events, aes(xmin = start, xmax = end, ymin = -Inf, ymax = Inf),
```



```

    fill = "#5F0A87", alpha = 0.2, inherit.aes = FALSE) +
  labs(title = paste("Weighed Average Personal Income OH, MI, MN Suburban Counties 2000 - 2009"),
    x = "Year",
    y = "Weighted Average Personal Income (in Thousands $)") +
  theme_minimal() +
  theme(legend.position = "bottom")

```

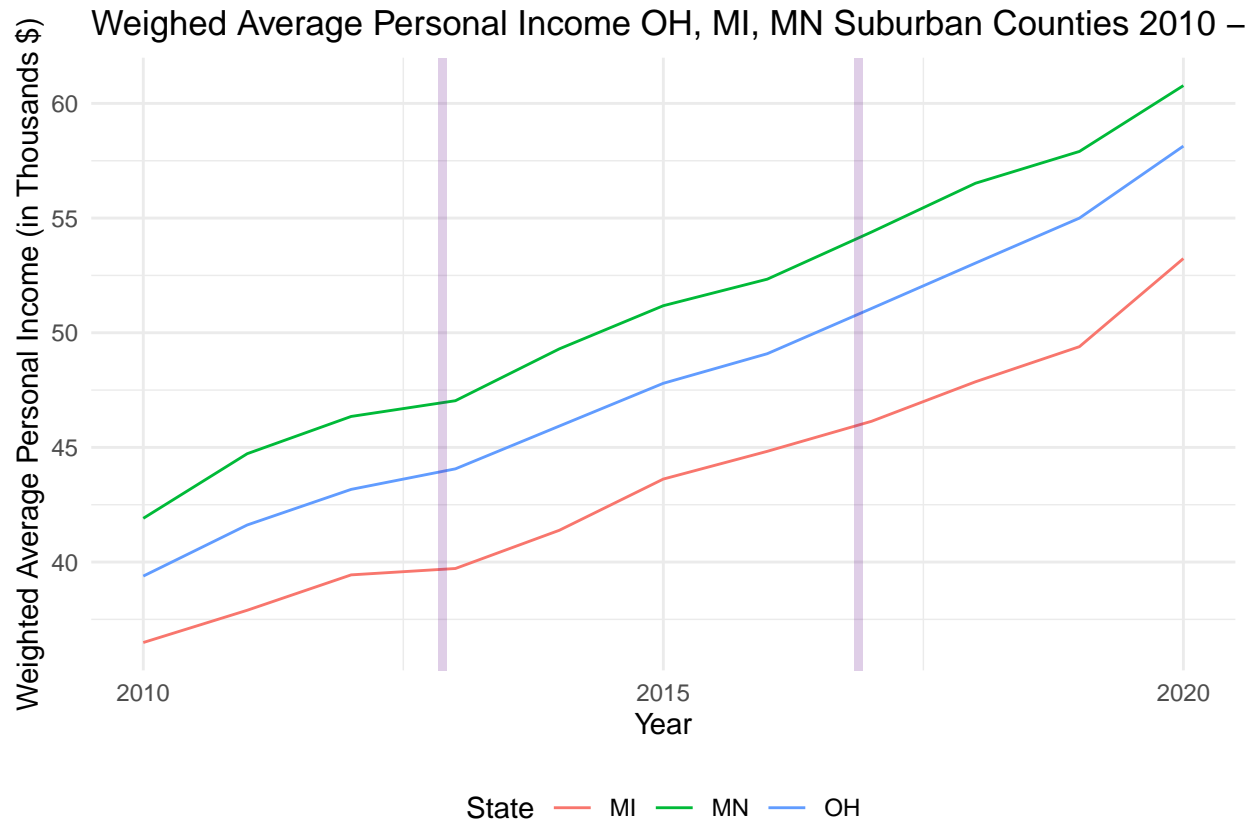


```

events <- data.frame(
  start = as.Date(c("2012-11-01", "2016-11-01")),
  end = as.Date(c("2012-11-30", "2016-11-30"))
)

ggplot(weighted_income3, aes(x=Year, y=WeightedAverageIncomeThousands, color = State, group = State)) +
  geom_line() +
  geom_rect(data = events, aes(xmin = start, xmax = end, ymin = -Inf, ymax = Inf),
    fill = "#5F0A87", alpha = 0.2, inherit.aes = FALSE) +
  labs(title = paste("Weighed Average Personal Income OH, MI, MN Suburban Counties 2010 - Present"),
    x = "Year",
    y = "Weighted Average Personal Income (in Thousands $)") +
  theme_minimal() +
  theme(legend.position = "bottom")

```



```
stateClean <- state_vote_data %>%
  mutate(State = case_when(state_name == "Michigan" ~ "MI",
                           state_name == "Ohio" ~ "OH",
                           state_name == "Minnesota" ~ "MN"),
         Year = as.character(year))

weighted_incomeVote <- weighted_income %>%
  filter(Year %in% c(2000, 2004, 2008, 2012, 2016, 2020))

voting_income <- inner_join(weighted_incomeVote, stateClean, by = join_by(State == State, Year == Year))

ggplot(voting_income, aes(WeightedAverageIncomeThousands, VoteDifference, color = State)) +
  geom_line() +
  labs(title = paste("Weighed Average Personal Income vs Voting Trends OH, MI, MN Suburban Counties 2010 –",
                    x = "Weighted Average Personal Income (in Thousands $)",
                    y = "Republican Vote Share Advantage") +
        theme_minimal() +
        theme(legend.position = "bottom"))
```

Weighed Average Personal Income vs Voting Trends OH, MI, MN Suburba

