# Washington State Domestic Violence Fatalities

I am interested in understanding trends in domestic violence fatalities in Washington state, specifically:

- Have rates of domestic violence homicides, suicides, and fatal police interventions changed over time?
- Which counties have the highest numbers of domestic violence homicides and suicides (particularly after normalizing for population)?

To address these questions I am using data from the Washington State Coalition Against Domestic Violence's Domestic Violence Fatality Review and population data from the Office of Financial Management. The Domestic Violence Fatality Review provides county-level data on domestic violence homicides, suicides, and fatal police interventions from 1997 through 2015 (though I will only be using data through 2014 as the second half of 2015 is not yet available). I used Tabula to convert the data PDF into a CSV. The resulting CSV is still quite messy and required some significant data cleaning.

```
## First load the required packages
library("readxl")
library("tidyr")
library("dplyr")
library("maps")
library("ggplot2")
library("viridis")

## Load data without the header row for now -- we'll add that back in later
dv <- read.table(
  "../data/original/tabula-fatalities-by-county-through-6-30-2015.csv",
  sep = ",",
  skip = 1,
  header = FALSE,
  stringsAsFactors = FALSE
)

## Fix row of fatality types (homicide, suicide, police intervention)
dv[1, ] <- gsub("\n", "", dv[1, ])     # remove \n's
dv[1, grep("(?=.*h)(?=.*o)(?=.*m)(?=.*i)(?=.*c)(?=.*d)(?=.*e)",
           dv[1, ], perl = TRUE)] <- "homicide"
dv[1, grep("(?=.*s)(?=.*u)(?=.*i)(?=.*c)(?=.*d)(?=.*e)",
           dv[1, ], perl = TRUE)] <- "suicide"
dv[1, grep("(?=.*p)(?=.*o)(?=.*l)(?=.*i)(?=.*c)(?=.*e)",
           dv[1, ], perl = TRUE)] <- "police.intervention"

## Get years from first row of CSV
years <- scan(
  "../data/original/tabula-fatalities-by-county-through-6-30-2015.csv",
  nlines = 1, sep = ",", what = character()
)

## Fill in empty strings. There should be 3 entries for each year. This creates
## 6 for 2015; the last three are actually columns of totals that are going to
## be removed anyway
for (i in seq_along(years)) {
```

```
  if (i == 1) {
    years[i] <- years[i]
  }
  else if (years[i] == "") {
    years[i] <- years[i - 1]
  }
}

## Paste together year and fatality type into one column header -- these will be
## separated later after reshaping
newnames <- paste(years, dv[1, ], sep = "_")
newnames[1] <- "county"

## Use year and fatality type as column names
names(dv) <- newnames

## Now get rid of first row of data that contains fatality types, the last row
## of data that contains state-wide totals, and last three columns which are
## totals for the time series
dv <- dv[-c(1, nrow(dv)), -c(ncol(dv), ncol(dv) - 1, ncol(dv) - 2)]
```

It would be helpful to have the data in a long format, so we'll reshape it and do a little final tidying.

```
dv_long <- dv %>%
  ## Gather into long format
  gather(key = yeartype, value = fatalities, -county) %>%
  ## Separate yeartype column into year and fatality type
  separate(yeartype, c("year", "type"), sep = "_") %>%
  ## Based on the original PDF it looks like NAs should be true zeroes
  mutate(fatalities = ifelse(fatalities == "", "0", fatalities)) %>%
  ## Convert fatalities to numeric
  mutate(fatalities = as.numeric(fatalities))
```

It will be useful to have county-level population data. The Office of Financial Management publishes yearly postcensal population estimates that will be good enough for our purposes. It too requires some cleaning.

```
## Download OFM data
## download.file(
##   url = "http://www.ofm.wa.gov/pop/april1/hseries/ofm_april1_postcensal_estimates_pop_1960-present.x
##   destfile = "../data/original/postcensal_pop_estimates.xlsx"
## )

## Load population estimates
pop <- read_excel(
  "../data/original/postcensal_pop_estimates.xlsx",
  sheet = 2, skip = 3
)

## Make syntactically valid column names
names(pop) <- make.names(names(pop), unique = TRUE)

## Remove empty and unnecessary columns and rows
pop <- pop %>%
```

```
  select(-starts_with("NA")) %>%
  filter(Filter == 1 & !County %in% c("State", ".")) %>%
  select(-Line, -Filter, -Jurisdiction)

## Remove periods from column names
names(pop) <- gsub("\\.", "", names(pop))

## Convert data to long format
pop_long <- pop %>%
  gather(key = year, value = pop, -County) %>%
  rename(county = County) %>%
  ## Remove postcensal estimates in years where there is an actual census
  filter(!year %in% c("X1960PostcensalEstimateofTotalPopulation",
                      "X1970PostcensalEstimateofTotalPopulation",
                      "X1980PostcensalEstimateofTotalPopulation",
                      "X1990PostcensalEstimateofTotalPopulation",
                      "X2000PostcensalEstimateofTotalPopulation",
                      "X2010PostcensalEstimateofTotalPopulation")) %>%
  mutate(year = substr(year, 2, 5)) %>%
  ## Remove data from before 1997 (the start of DV data)
  filter(year >= 1997) %>%
  ## Make population and year numeric
  mutate(pop = as.numeric(pop),
         year = as.numeric(year))
```

Finally, we'll want to merge the fatality and population data.

```
dat <- merge(dv_long, pop_long, by = c("county", "year"))
```

## Preliminary Graphs

What is the distribution of fatalities by county in a given year, normalized by population?

```
dat %>%
  group_by(county, year) %>%
  summarize(tot_fatal_pop = sum(fatalities, na.rm = TRUE) / (unique(pop) / 10000)) %>%
  ggplot(aes(x = tot_fatal_pop)) +
  geom_histogram(binwidth = 0.1) +
  facet_wrap(~ year, ncol = 3) +
  ylab("Count") +
  xlab("Number of fatalities per 10,000 population") +
  ggtitle("Domestic Violence Fatalities")
```
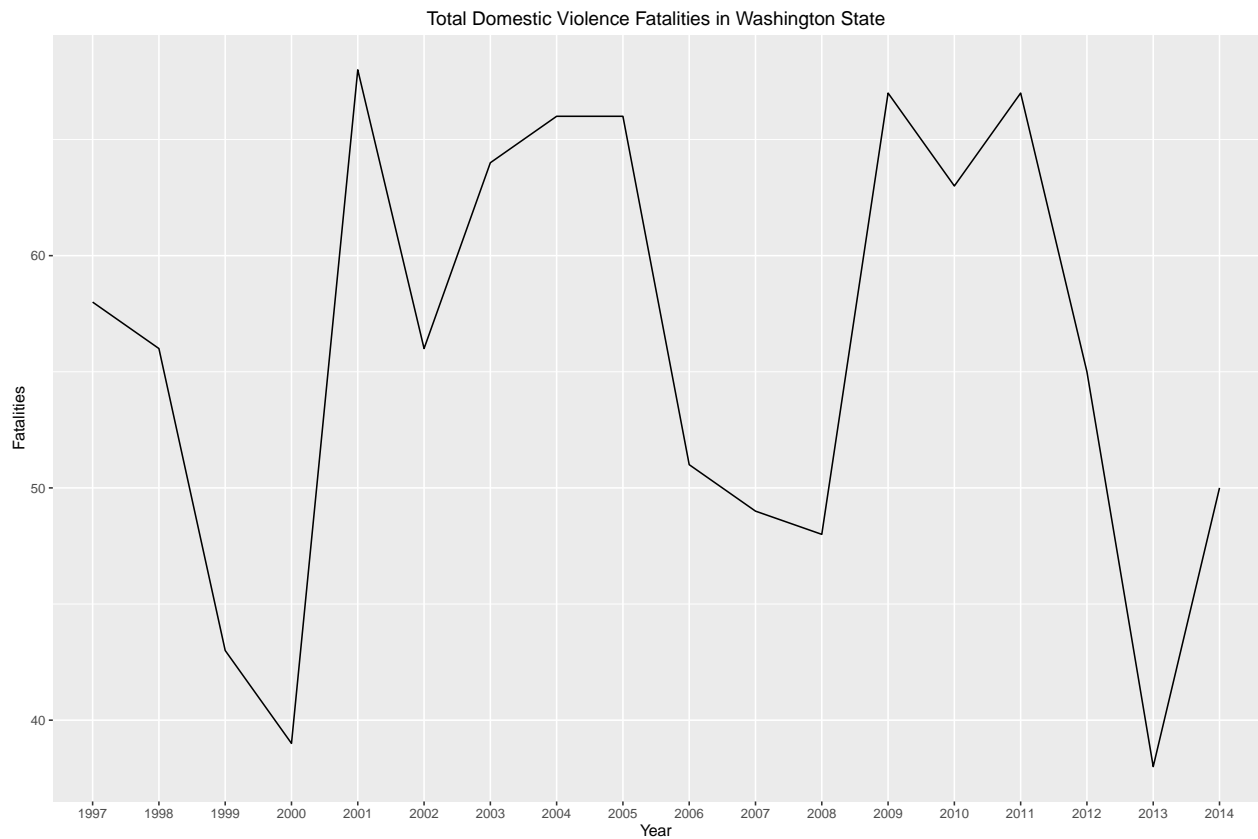
Domestic Violence Fatalities
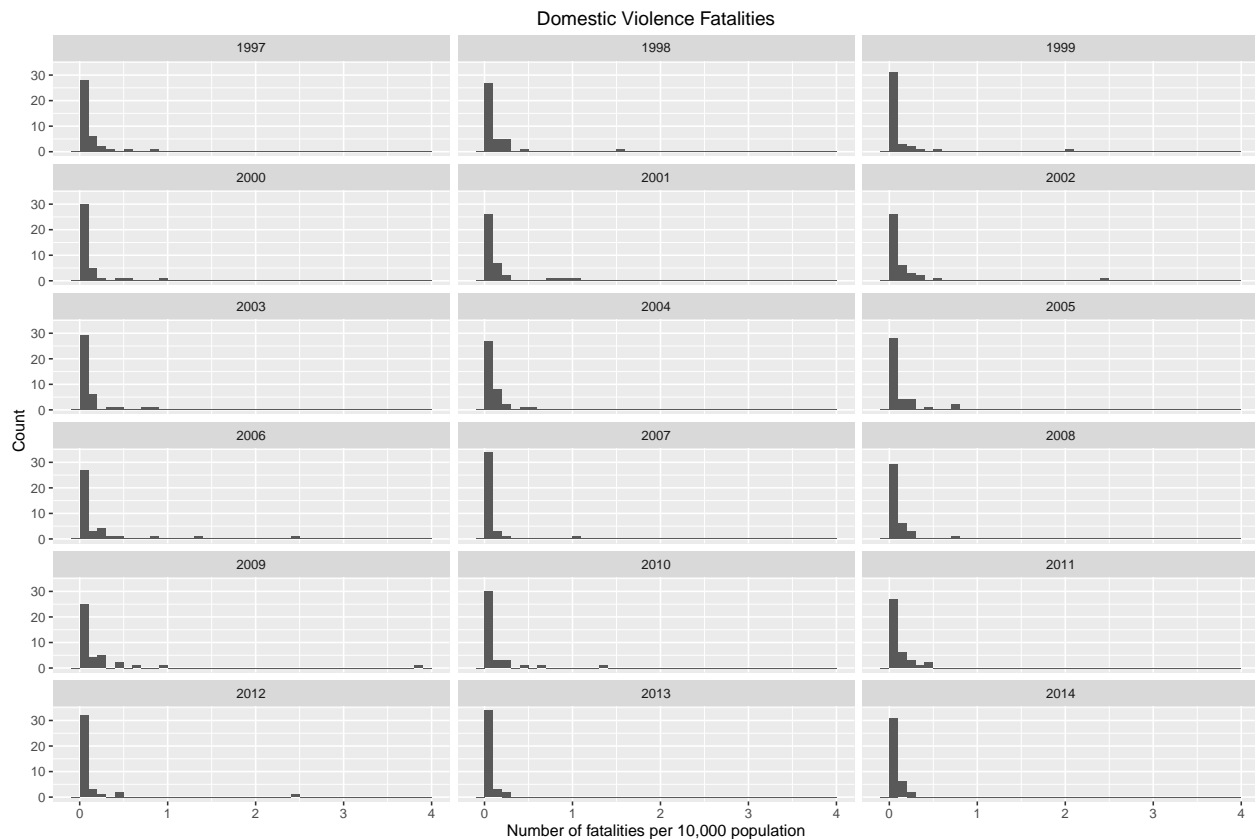
## Time series and temporal trends

What, if any, temporal trends are there in total fatalities and the three types of fatalities?

```
dat %>%
  group_by(year) %>%
  summarize(tot_fatal_state = sum(fatalities, na.rm = TRUE)) %>%
  ggplot(aes(x = year, y = tot_fatal_state, group = 1)) +
  geom_line() +
  ylab("Fatalities") +
  xlab("Year") +
  ggtitle("Total Domestic Violence Fatalities in Washington State")
```

4

**Total Domestic Violence Fatalities in Washington State**



```
dat %>%
  group_by(year, type) %>%
  summarize(tot_fatal_state = sum(fatalities, na.rm = TRUE)) %>%
    ggplot(aes(x = year, y = tot_fatal_state, group = 1)) +
  geom_line() +
  facet_wrap(~ type, ncol = 1) +
  ylab("Fatalities") +
  xlab("Year") +
  ggtitle("Domestic Violence Fatalities in Washington State by Type")
```

Domestic Violence Fatalities in Washington State by Type
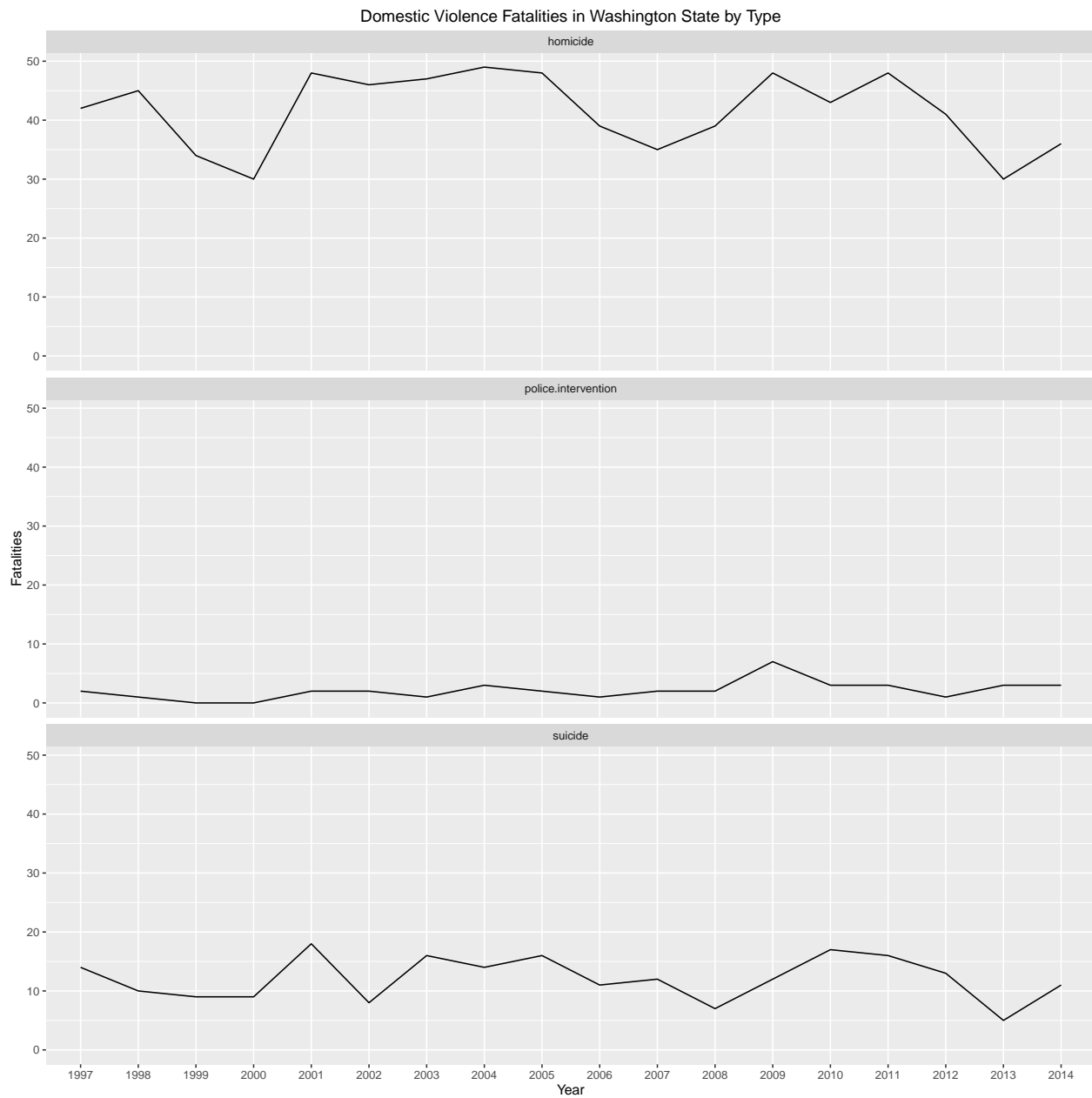


Can we see any trends in homicides by county?

```
## Line graph of homicides
dat %>%
  filter(type == "homicide") %>%
  ggplot(aes(x = year, y = fatalities, group = county, color = county)) +
  geom_line()
```
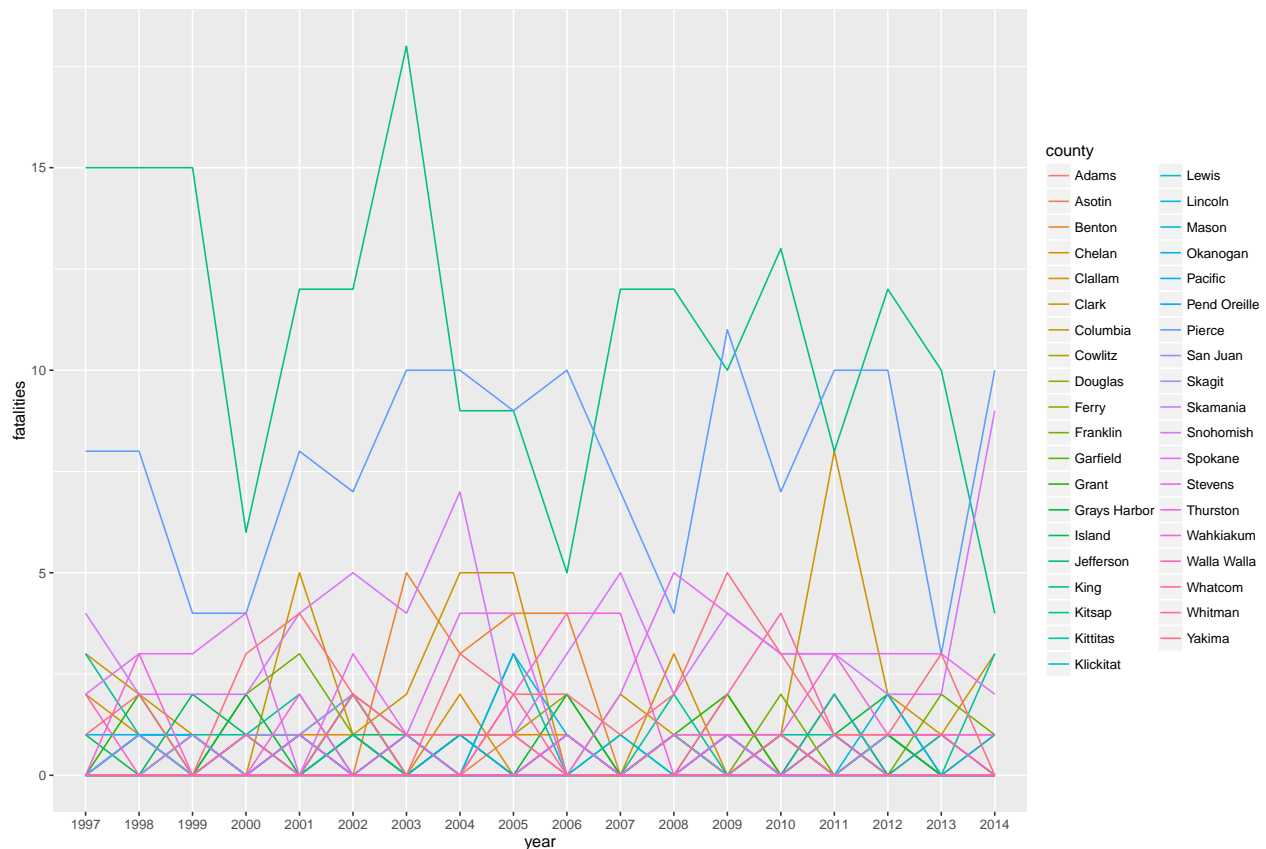
Okay not that way... how about faceting?

```
## Faceted by county
dat %>%
  filter(type == "homicide") %>%
  ggplot(aes(x = year, y = fatalities)) +
  facet_wrap(~ county, ncol = 5) +
  geom_line(group = 1) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  ylab("Number of fatalities") +
  xlab("Year") +
  ggtitle("Domestic Violence Homicide Fatalities")
```

Domestic Violence Homicide Fatalities
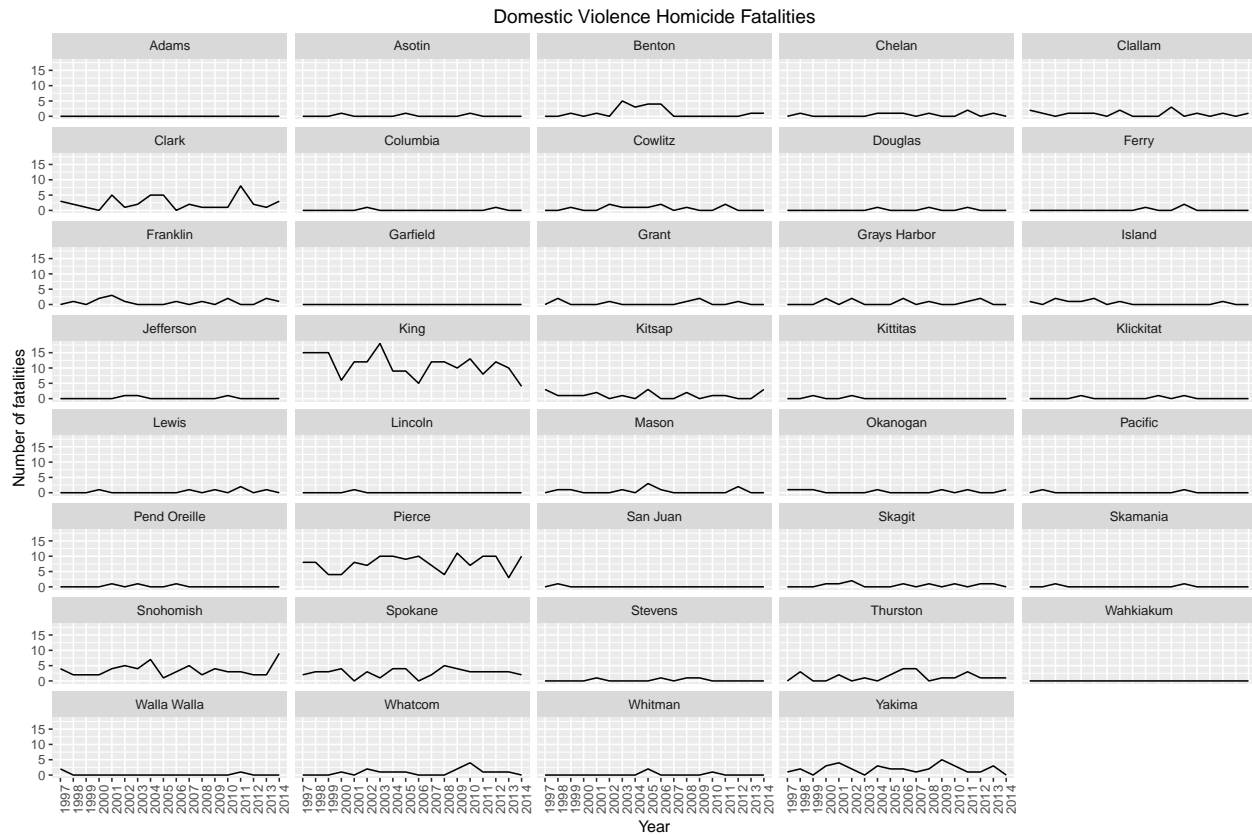
**Maps of fatalities by county**

To visualize homicides on a map, we first need to acquire county boundary data from `ggplot2`'s `map_data()` function and merge it with our homicide data.

```
## Gather county spatial data
county_map <- ggplot2::map_data("county") %>%
  filter(region == "washington") %>%
  ## Rename subregion column to county
  rename(county = subregion) %>%
  ## Remove region column (washington)
  select(-region) %>%
  ## Capitalize counties
  mutate(county = gsub("(^|[[:space:]])([[:alpha:]])", "\\1\\U\\2",
                       county, perl = TRUE))

## Merge homicide data with county map
hom_county <- dat %>%
  filter(type == "homicide") %>%
  merge(county_map, by = "county")
```

Then we'll map the number of domestic violence homicides by county using colors from the `viridis` package.

```
## Map of number of homicides in each county by year
ggplot(hom_county, aes(x = long, y = lat, group = group)) +
  facet_wrap(~ year, ncol = 3) +
```
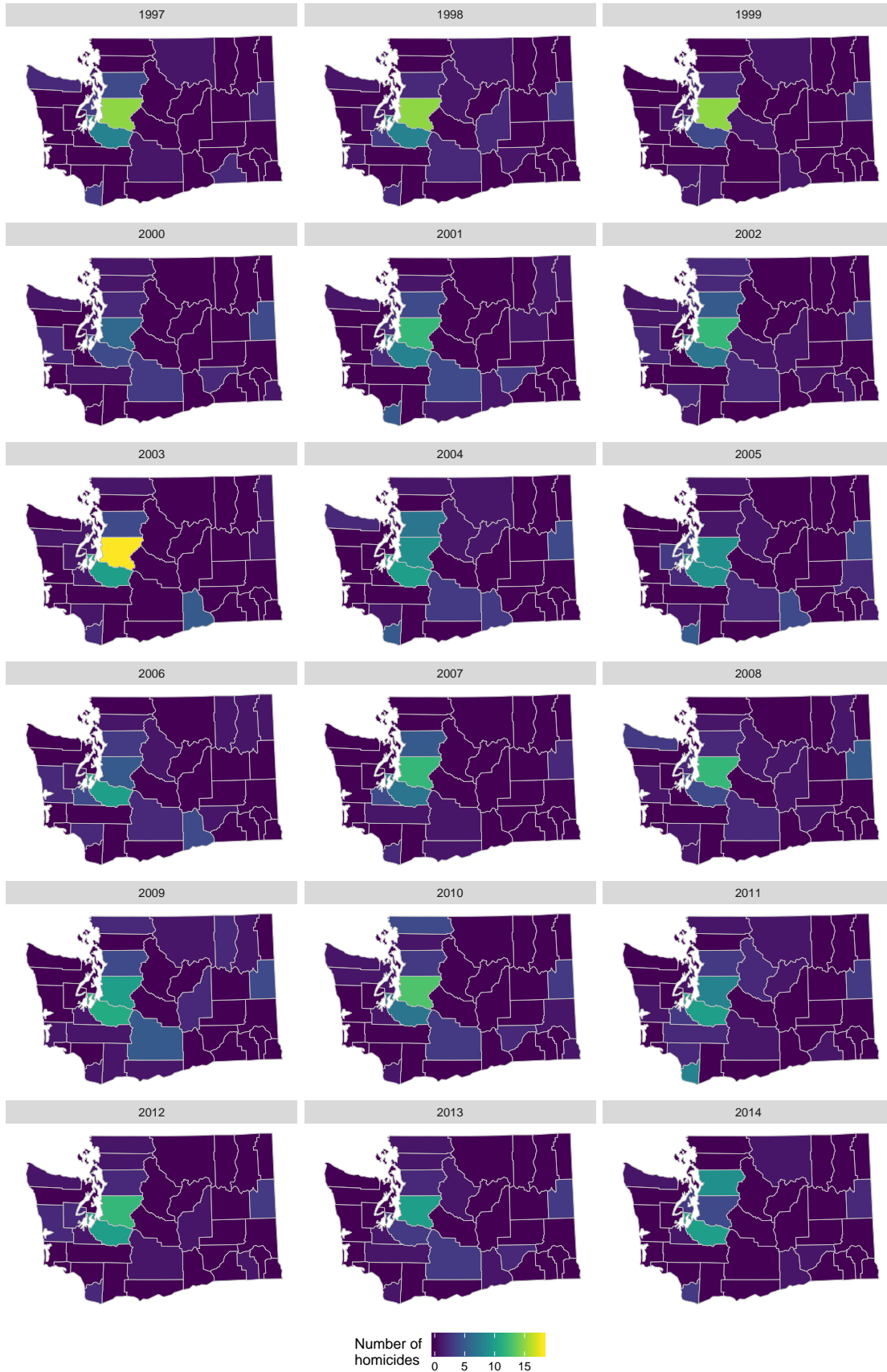
```r
geom_polygon(colour = "grey", size = 0.3, aes(fill = fatalities)) +
scale_fill_viridis(alpha = 1, begin = 0, end = 1, discrete = FALSE,
                   option = "D") +
theme(axis.text = element_blank(),
      axis.ticks = element_blank(),
      axis.title = element_blank(),
      panel.background = element_blank(),
      panel.grid = element_blank(),
      legend.position = "bottom") +
expand_limits(x = hom_county$long, y = hom_county$lat) +
coord_map("polyconic") +
labs(fill="Number of \nhomicides") +
ggtitle("Domestic Violence Homicides")
```

Domestic Violence Homicides

Predictably, this basically looks like a population map of Washington. So instead let's look at the number of homicides per 10,000 people.

```
## Map of number of homicides/population by year
ggplot(hom_county, aes(x = long, y = lat, group = group)) +
  facet_wrap(~ year, ncol = 3) +
  geom_polygon(colour = "grey", size = 0.3, aes(fill = fatalities / (pop / 10000))) +
  scale_fill_viridis(alpha = 1, begin = 0, end = 1, discrete = FALSE,
                     option = "D") +
  theme(axis.text = element_blank(),
        axis.ticks = element_blank(),
        axis.title = element_blank(),
        panel.background = element_blank(),
        panel.grid = element_blank(),
        legend.position = "bottom") +
  expand_limits(x = hom_county$long, y = hom_county$lat) +
  coord_map("polyconic") +
  labs(fill="Homicides per \n10,000 population") +
  ggtitle("Domestic Violence Homicides")
```

# Domestic Violence Homicides



Homicides per
10,000 population  0.0 0.5 1.0 1.5 2.0 2.5