Midterm Project: County-level oil and gas production

Zichun Liu

10/11/2016

Part One: Data Description

1.1 Background and Data sources

US has large quantities of oil and gas trapped in shale and other formations.

Prior to the application of unconventional drilling methods, U.S. natural gas production had been growing slowly, and U.S. oil production had been in decline since the mid-1980s. Both trends reversed during the 2000s.

Most policies regarding oil and gas development occur at the State level. Using data from State agencies, the Energy Information Administration (EIA) publishes oil and gas production totals by State, but more local data for the entire United States had not been widely available.

By acquiring disaggregated oil and gas production data from State agencies, a national county-level database was created, providing yearly estimates of onshore production for counties in the lower 48 States. Nationwide county-level data permit a more comprehensive assessment of the geography of oil and gas development. These county-level data also allow researchers to assess changes in rural production over the past decade.

1.2 Description of Dataset

County-level data from oil and/or natural gas producing States—for onshore production in the lower 48 States only—are compiled on a State-by-State basis.

Most States have production statistics available by county, field, or well, and these data were compiled at the county level to create a database of county-level production, annually for 2000 through 2011.

Raw data for natural gas is for gross withdrawals, and oil data almost always include natural gas liquids.

In the data file, counties with increases or decreases in excess of $20 million in oil and/or natural gas production during 2000-11 are also identified. See Documentation and Maps for more details.

1.3 How it was collected, assembled and maintained

ERS researchers created this national county-level database providing yearly estimates of onshore production for counties in the lower 48 States. And was maintained by ERS researchers as well.

Part Two: Data Cleaning and Preparation

2.1 Manipulate oilgas

```
oilgas <- read.csv2("oilgascounty.csv",header = TRUE,sep = ",")
oilgas$County_Name <- str_replace_all(oilgas$County_Name, " County","")
for (i in 1:length(letters)) {
  oilgas$County_Name <- str_replace_all(oilgas$County_Name,
                                        LETTERS[i],letters[i])}
```

```
for (i in 1:length(state.abb)) {
  oilgas$Stabr <- str_replace_all(oilgas$Stabr,
                                  state.abb[i],state.name[i])}
for (i in 1:length(letters)) {
  oilgas$Stabr <- str_replace_all(oilgas$Stabr,
                                  LETTERS[i],letters[i])}
oilgas$oil_change_group <- factor(oilgas$oil_change_group,
                                  levels = c("H_Growth", "H_Decline", "Status Quo"))
oilgas$gas_change_group <- factor(oilgas$gas_change_group,
                                  levels = c("H_Growth", "H_Decline", "Status Quo"))
oilgas$oil_gas_change_group <- factor(oilgas$oil_gas_change_group,
                                      levels = c("H_Growth", "H_Decline", "Status Quo"))

head(oilgas)
```

```
##   FIPS geoid   Stabr County_Name Rural_Urban_Continuum_Code_2013
## 1 1001  1001 alabama     autauga                               2
## 2 1003  1003 alabama     baldwin                               3
## 3 1005  1005 alabama     barbour                               6
## 4 1007  1007 alabama        bibb                               1
## 5 1009  1009 alabama      blount                               1
## 6 1011  1011 alabama     bullock                               6
##   Urban_Influence_2013 Metro_Nonmetro_2013 Metro_Micro_Noncore_2013
## 1                    2                   1                        2
## 2                    2                   1                        2
## 3                    6                   0                        0
## 4                    1                   1                        2
## 5                    1                   1                        2
## 6                    6                   0                        0
##   oil2000 oil2001 oil2002 oil2003 oil2004 oil2005 oil2006 oil2007 oil2008
## 1       0       0       0       0       0       0       0       0       0
## 2  138072  134666  138011  127985  130763  118043  103992  112303   97623
## 3       0       0       0       0       0       0       0       0       0
## 4       0       0       0       0       0       0       0       0       0
## 5       0       0       0       0       0       0       0       0       0
## 6       0       0       0       0       0       0       0       0       0
##   oil2009 oil2010 oil2011  gas2000  gas2001   gas2002   gas2003  gas2004
## 1       0       0       0        0        0         0         0        0
## 2   84982  101955   94638 72543902 98699994 107142655 101510068 90146850
## 3       0       0       0        0        0         0         0        0
## 4       0       0       0        0        0         0         0        0
## 5       0       0       0        0        0         0         0        0
## 6       0       0       0        0        0         0         0        0
##    gas2005  gas2006  gas2007  gas2008  gas2009  gas2010  gas2011
## 1        0        0        0        0        0        0        0
## 2 84536875 83951640 82876786 78547145 68525628 63069025 51041072
## 3        0        0        0        0        0        0        0
## 4     8301    98853   480015   684143   551719   453132   400504
## 5        0        0        0    20516    61054     3594    21496
## 6        0        0        0        0        0        0        0
##   oil_change_group gas_change_group oil_gas_change_group
## 1       Status Quo       Status Quo           Status Quo
## 2       Status Quo        H_Decline            H_Decline
```

```
## 3       Status Quo       Status Quo           Status Quo
## 4       Status Quo       Status Quo           Status Quo
## 5       Status Quo       Status Quo           Status Quo
## 6       Status Quo       Status Quo           Status Quo
```

```
#see if id is unique
length(unique(oilgas$geoid)) == length(oilgas$geoid)
```

```
## [1] TRUE
```

```
#see if FIPS is same as geoid
sum(oilgas$FIPS - oilgas$geoid)
```

```
## [1] 0
```

```
#remove FIPS
oilgas$FIPS <- NULL

#calculate total withdraw in 12 years and level the sum
for (i in 1:length(oilgas$geoid)) {
  oilgas$ttoil[i] <- sum(as.numeric(oilgas[i,8:19]))
  a <-oilgas$ttoil[i]
  if (a < 10) {oilgas$oil_level[i] <- (0)}
  else if (a >= 10 & a < 1000) {oilgas$oil_level[i] <- (1)}
  else if (a >= 1000 & a < 10000) {oilgas$oil_level[i] <- (2)}
  else if (a >= 10^4 & a < 10^5) {oilgas$oil_level[i] <- (3)}
  else if (a >= 10^5 & a < 10^6) {oilgas$oil_level[i] <- (4)}
  else if (a >= 10^6 & a < 10^7) {oilgas$oil_level[i] <- (5)}
  else if (a >= 10^7 & a < 10^8) {oilgas$oil_level[i] <- (6)}
  else if (a >= 10^8 & a < 10^9) {oilgas$oil_level[i] <- (7)}
  else {oilgas$oil_level[i] <- (8)}
  oilgas$ttgas[i] <- sum(as.numeric(oilgas[i,20:31]))
  a <- oilgas$ttgas[i]
  if (a < 10 | is.na(a) == TRUE) {oilgas$gas_level[i] <- (0)}
  else if (a >= 10 & a < 1000) {oilgas$gas_level[i] <- (1)}
  else if (a >= 1000 & a < 10000) {oilgas$gas_level[i] <- (2)}
  else if (a >= 10^4 & a < 10^5) {oilgas$gas_level[i] <- (3)}
  else if (a >= 10^5 & a < 10^6) {oilgas$gas_level[i] <- (4)}
  else if (a >= 10^6 & a < 10^7) {oilgas$gas_level[i] <- (5)}
  else if (a >= 10^7 & a < 10^8) {oilgas$gas_level[i] <- (6)}
  else if (a >= 10^8 & a < 10^9) {oilgas$gas_level[i] <- (7)}
  else if (a >= 10^9){oilgas$gas_level[i] <- (8)}
}
```

2.2 Get first tidy dataset

```
a <- oilgas %>% gather(key = year,value = oil, oil2000:oil2011)
b <- oilgas %>% gather(key = year,value = gas, gas2000:gas2011)
b <- b[c("geoid","year","gas")]
target <- cbind(a,b[-1])
```

```
target <- target[-c(8:19,29)]
target$year <- factor(str_replace_all(target$year, "oil",""))
head(target)
```

```
##   geoid   Stabr County_Name Rural_Urban_Continuum_Code_2013
## 1  1001 alabama     autauga                               2
## 2  1003 alabama     baldwin                               3
## 3  1005 alabama     barbour                               6
## 4  1007 alabama        bibb                               1
## 5  1009 alabama      blount                               1
## 6  1011 alabama     bullock                               6
##   Urban_Influence_2013 Metro_Nonmetro_2013 Metro_Micro_Noncore_2013
## 1                    2                   1                        2
## 2                    2                   1                        2
## 3                    6                   0                        0
## 4                    1                   1                        2
## 5                    1                   1                        2
## 6                    6                   0                        0
##   oil_change_group gas_change_group oil_gas_change_group   ttoil oil_level
## 1       Status Quo       Status Quo           Status Quo       0         0
## 2       Status Quo        H_Decline            H_Decline 1383033         5
## 3       Status Quo       Status Quo           Status Quo       0         0
## 4       Status Quo       Status Quo           Status Quo       0         0
## 5       Status Quo       Status Quo           Status Quo       0         0
## 6       Status Quo       Status Quo           Status Quo       0         0
##       ttgas gas_level year    oil      gas
## 1         0         0 2000      0        0
## 2 982591640         7 2000 138072 72543902
## 3         0         0 2000      0        0
## 4   2676667         5 2000      0        0
## 5    106660         4 2000      0        0
## 6         0         0 2000      0        0
```

```
saveRDS(target,"tidydataoilgas.rds")
rm(a,b,target)
tidy <- readRDS("tidydataoilgas.rds")
```

2.3 Get another tidy dataset

```
tidy2 <- oilgas[,-(8:31)]
saveRDS(tidy2,"data_without_year.rds")
```

Part Three: Exploratory Data Analysis

3.1 Explore on first tidy dataset(oil&gas data from 2000 to 2011)

1) on nation level

```r
#on nation level
Nationoil <- NULL
inrateoil <- NULL
Nationgas <- NULL
inrategas <- NULL
for (i in 1:12) {
  Nationoil[i] <- sum(as.numeric(subset(tidy, subset = year == 1999+i)$oil))
  Nationgas[i] <- sum(as.numeric(subset(tidy, subset = year == 1999+i)$gas))
  if (i == 1) {
    inrateoil[i] <- 0
    inrategas[i] <- 0
  }
  else{
    inrateoil[i] <- (Nationoil[i]-Nationoil[i-1])/Nationoil[i-1]
    inrategas[i] <- (Nationgas[i]-Nationgas[i-1])/Nationgas[i-1]
  }
}
Nation <- data.frame(year=2000:2011,
                     Nationoil = Nationoil,
                     oilinrate = inrateoil,
                     Nationgas = Nationgas,
                     gasinrate = inrategas)

rm(Nationoil,Nationgas,inrategas,inrateoil)

summary(Nation)
```

```
##       year        Nationoil           oilinrate
##  Min.   :2000   Min.   :1.078e+09   Min.   :-0.036228
##  1st Qu.:2003   1st Qu.:1.091e+09   1st Qu.:-0.018507
##  Median :2006   Median :1.124e+09   Median :-0.002476
##  Mean   :2006   Mean   :1.145e+09   Mean   : 0.009753
##  3rd Qu.:2008   3rd Qu.:1.176e+09   3rd Qu.: 0.012760
##  Max.   :2011   Max.   :1.346e+09   Max.   : 0.141568
##    Nationgas           gasinrate
##  Min.   :1.572e+10   Min.   :-0.007303
##  1st Qu.:1.602e+10   1st Qu.: 0.008583
##  Median :1.699e+10   Median : 0.024923
##  Mean   :1.806e+10   Mean   : 0.033057
##  3rd Qu.:1.999e+10   3rd Qu.: 0.047637
##  Max.   :2.310e+10   Max.   : 0.093644
```

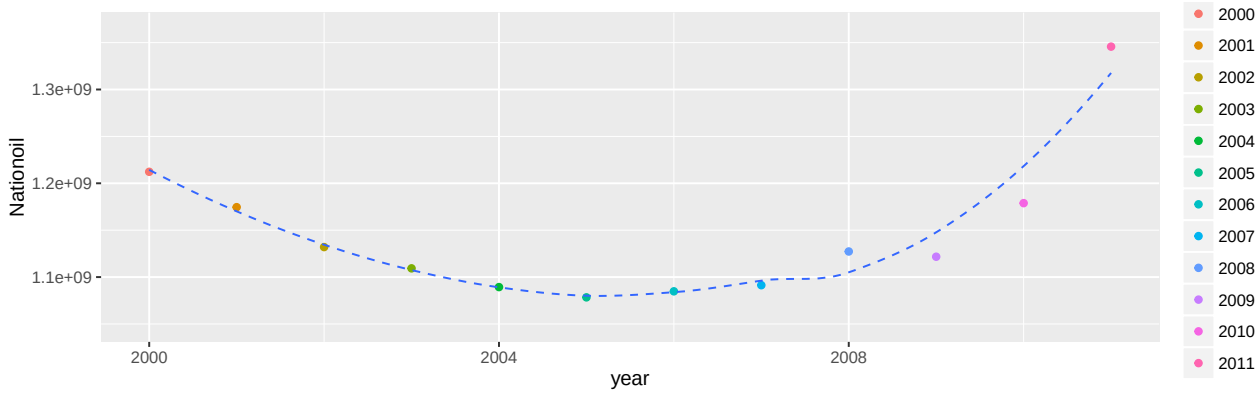
```r
#average increasing rate of oil production
sum(Nation$oilinrate)/11
```

```
## [1] 0.01063942
```

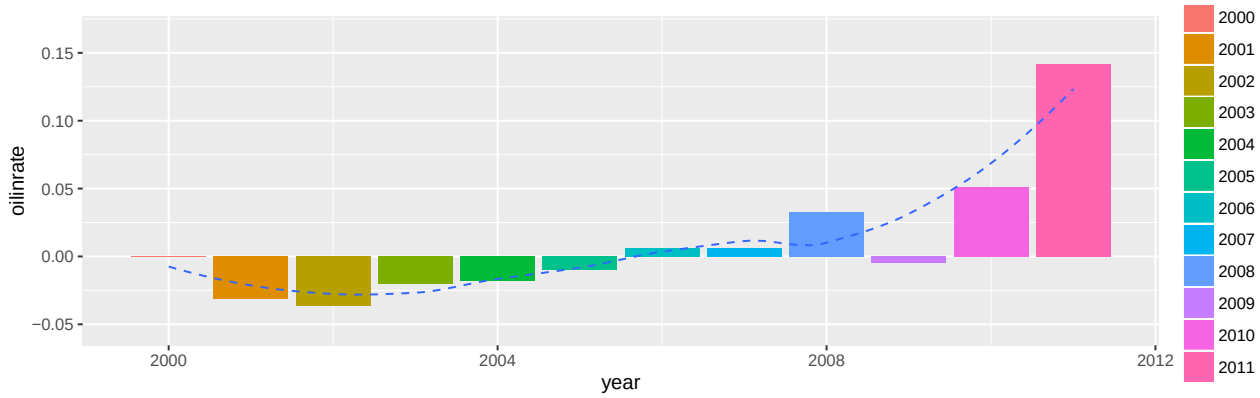
```r
#average increasing rate of gas production
sum(Nation$gasinrate)/11
```

```
## [1] 0.03606225
```

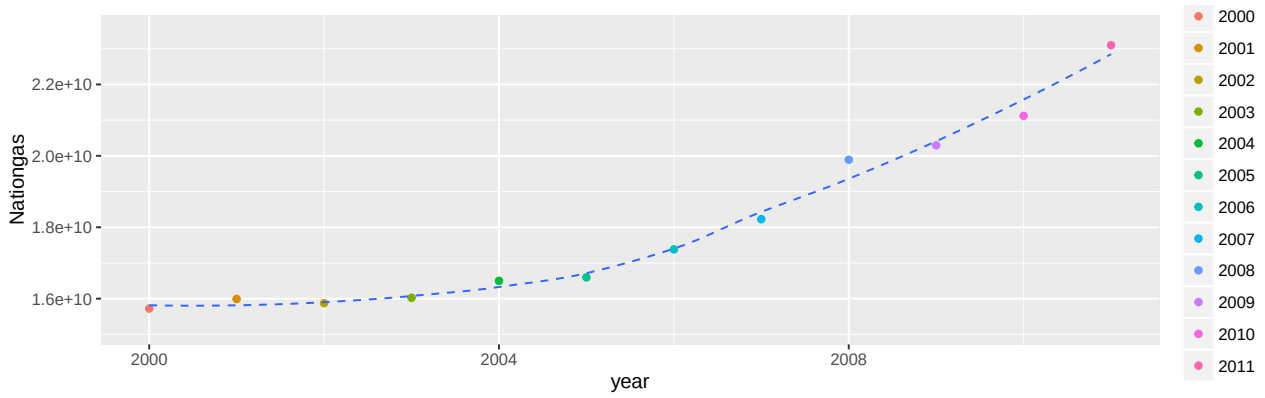
```
#nation <- Nation %>% gather(key = resource, value = volume, c(Nationoil,Nationgas))

ggplot(data = Nation, aes(x = year, y = Nationoil)) +
  geom_point(aes(color = factor(year))) +
  geom_smooth(linetype = 2, size = 0.5, alpha = 0)
```



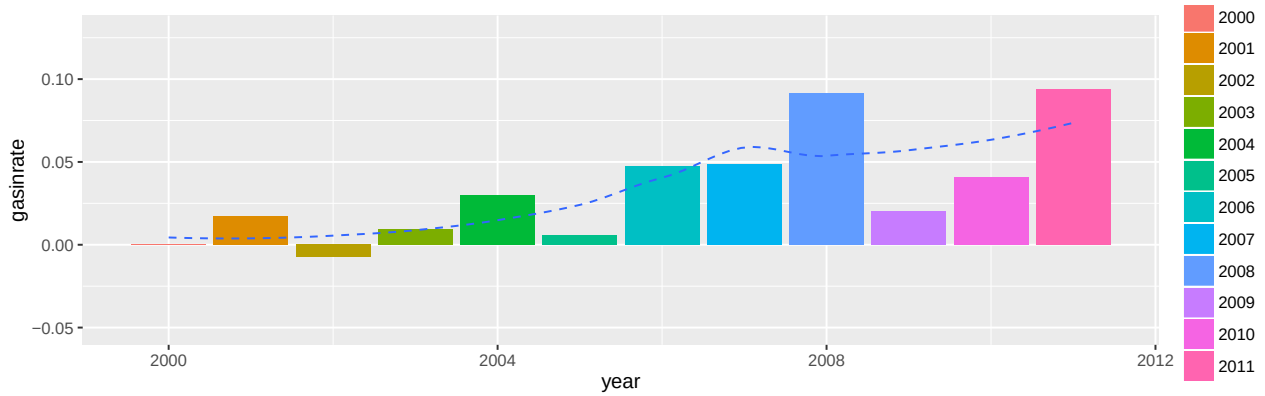

```
ggplot(data = Nation, aes(x = year, y = oilinrate)) +
  geom_bar(stat = "identity", aes(fill = factor(year))) +
  geom_smooth(linetype = 2, size = 0.5, alpha = 0)
```




```
ggplot(data = Nation) +
  geom_point(aes(x = year, y = Nationgas, color = factor(year))) +
  geom_smooth(aes(x = year, y = Nationgas),linetype = 2, size = 0.5, alpha = 0)
```

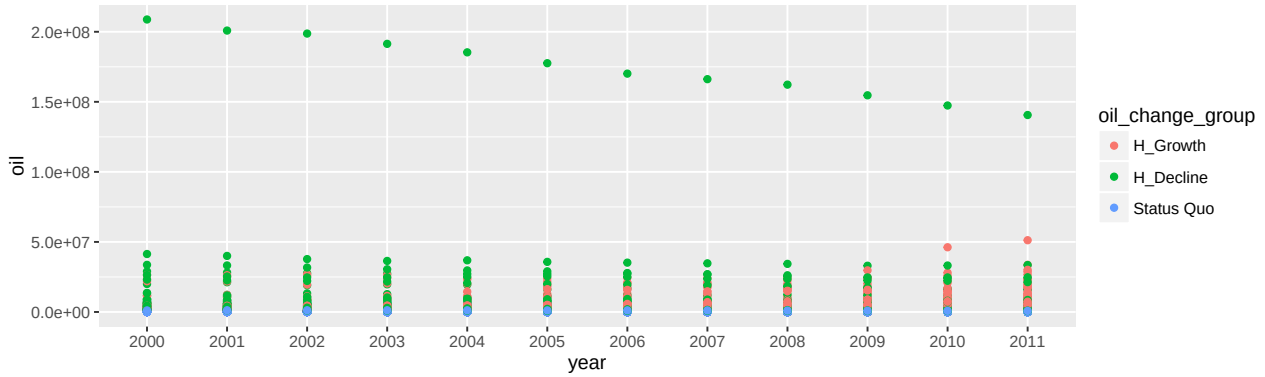



```
ggplot(data = Nation, aes(x = year, y = gasinrate)) +
  geom_bar(stat = "identity", aes(fill = factor(year))) +
  geom_smooth(linetype = 2, size = 0.5, alpha = 0)
```

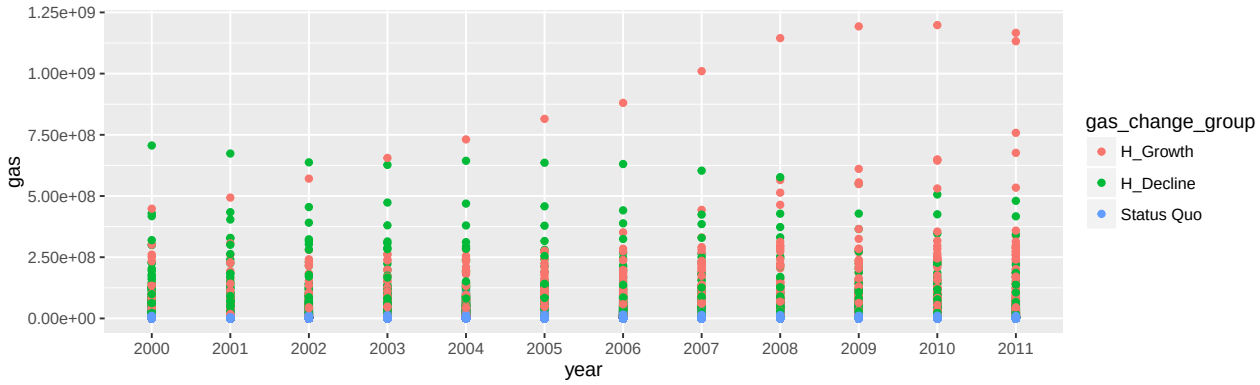


2) overall

```
#overall
ggplot(tidy) +
  geom_point(aes(x = year, y = oil, color = oil_change_group))
```

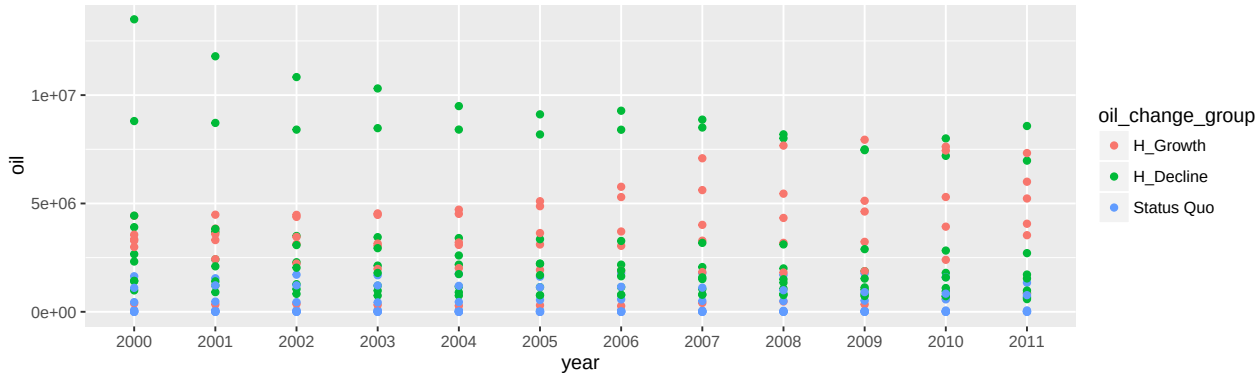


```
ggplot(tidy) +
  geom_point(aes(x = year, y = gas, color = gas_change_group))
```

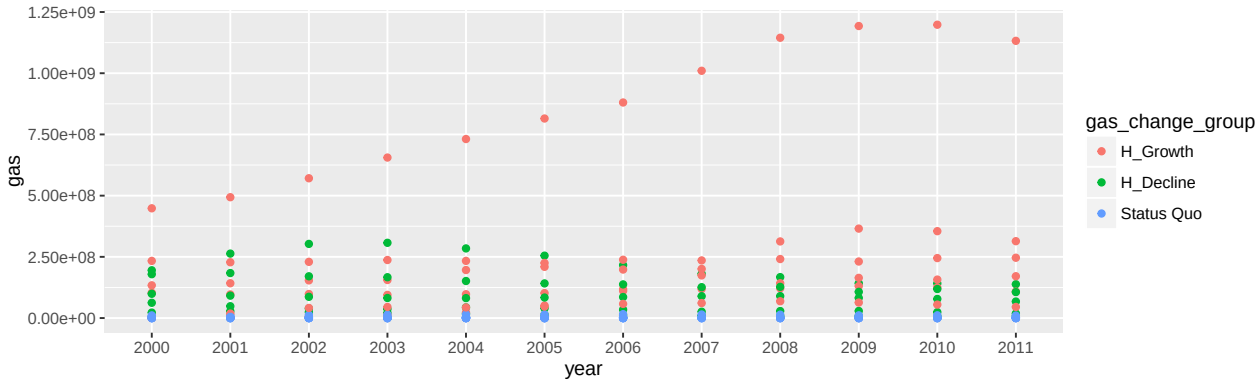


3) on state level(use example of Wyoming)

```
#by state(use example of Wyoming)
ggplot(data = subset(tidy,subset = Stabr=="wyoming"))+
  geom_point(aes(x = year, y = oil, color = oil_change_group))
```

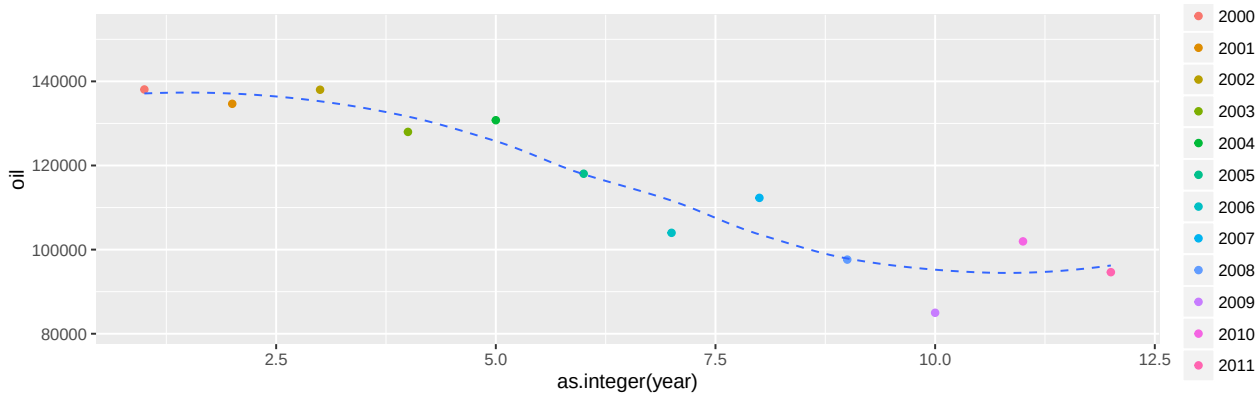



```
ggplot(data = subset(tidy,subset = Stabr=="wyoming"))+
  geom_point(aes(x = year, y = gas, color = gas_change_group))
```

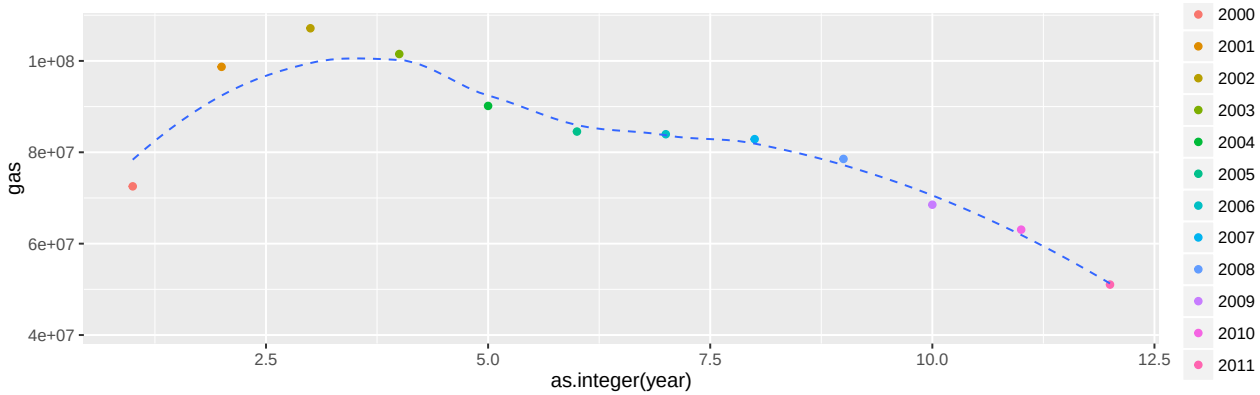


4) on County_Name (use example of baldwin, Alabama)

```
ggplot(data = subset(tidy,subset = geoid==1003),
       aes(x = as.integer(year), y = oil))+
  geom_point(aes(color = factor(year))) +
  geom_smooth(linetype = 2, size = 0.5, alpha = 0)
```

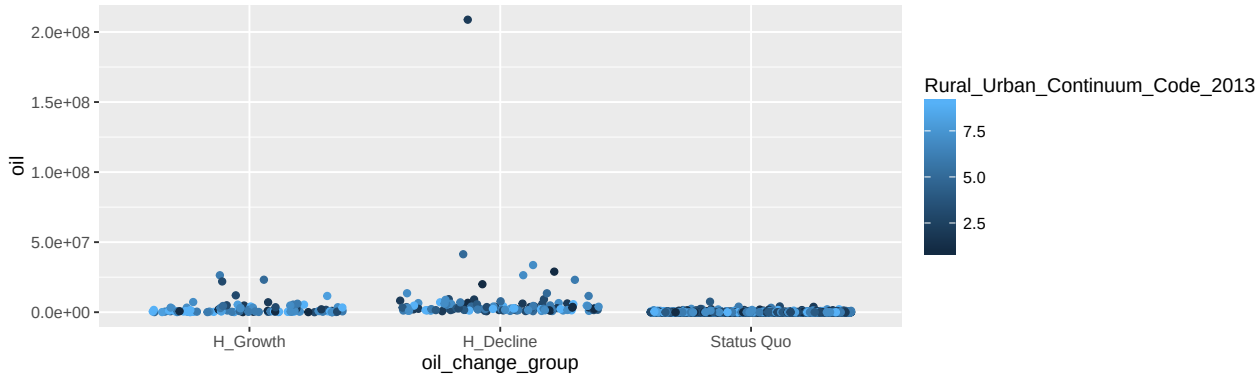



```
ggplot(data = subset(tidy,subset = geoid==1003),
       aes(x = as.integer(year), y = gas)) +
  geom_point(aes(color = factor(year))) +
  geom_smooth(linetype = 2, size = 0.5, alpha = 0)
```

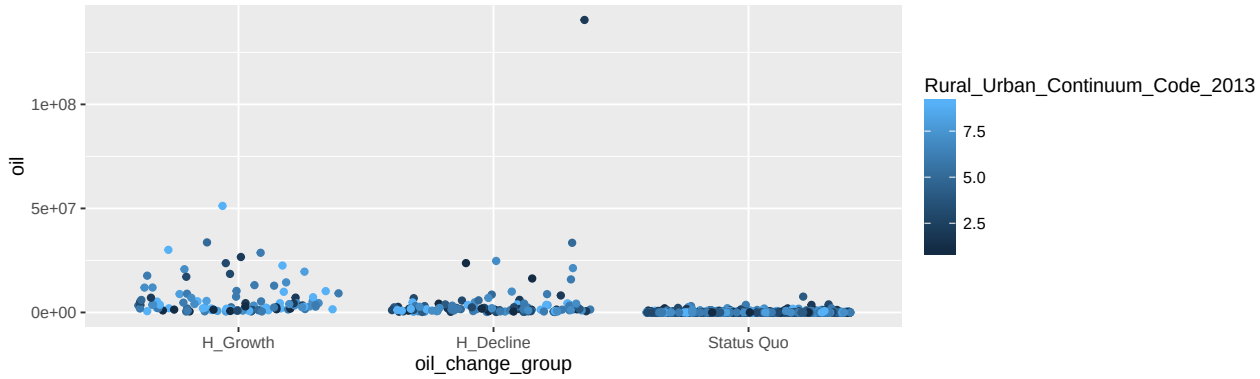


5) on year level(use example of first year(2000) and last year(2011))

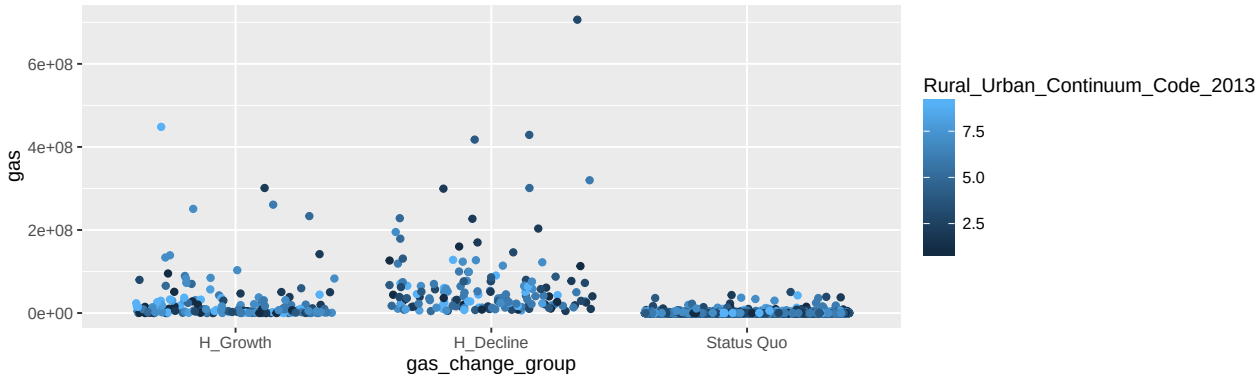
```
#oid
ggplot(data = subset(tidy,subset = (year == 2000)),
       aes(x = oil_change_group, y = oil,
           color = Rural_Urban_Continuum_Code_2013)) +
  geom_jitter()
```



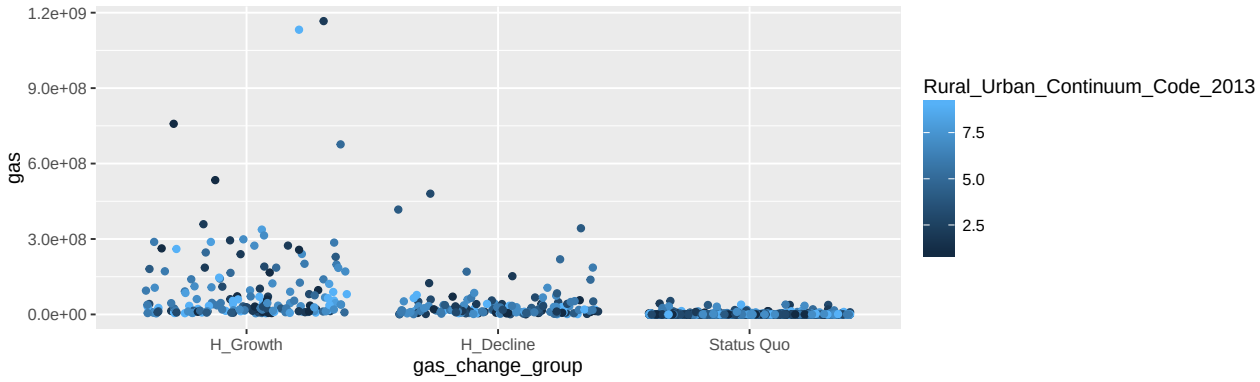

```
ggplot(data = subset(tidy,subset = (year == 2011)),
       aes(x = oil_change_group, y = oil,
           color = Rural_Urban_Continuum_Code_2013)) +
  geom_jitter()
```




```
#gas
ggplot(data = subset(tidy,subset = (year == 2000)),
       aes(x = gas_change_group, y = gas,
           color = Rural_Urban_Continuum_Code_2013)) +
  geom_jitter()
```

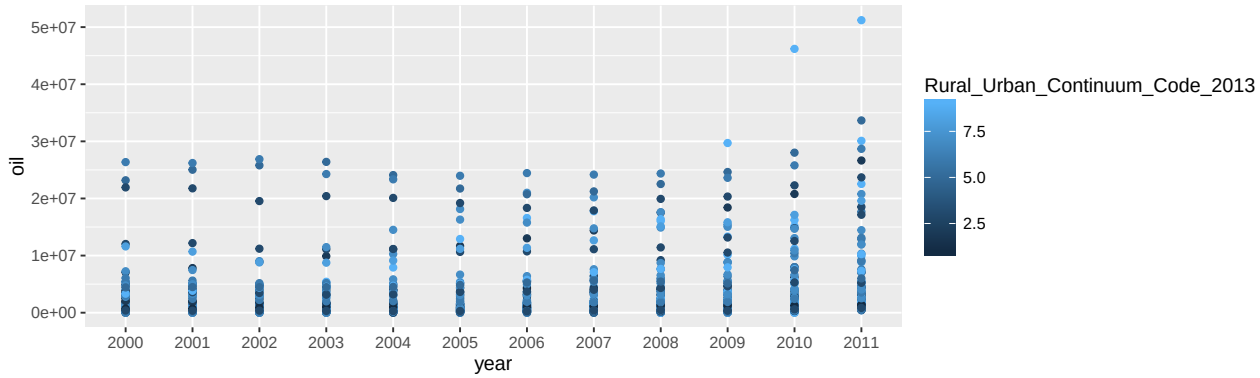


```
ggplot(data = subset(tidy,subset = (year == 2011)),
       aes(x = gas_change_group, y = gas,
           color = Rural_Urban_Continuum_Code_2013)) +
  geom_jitter()
```

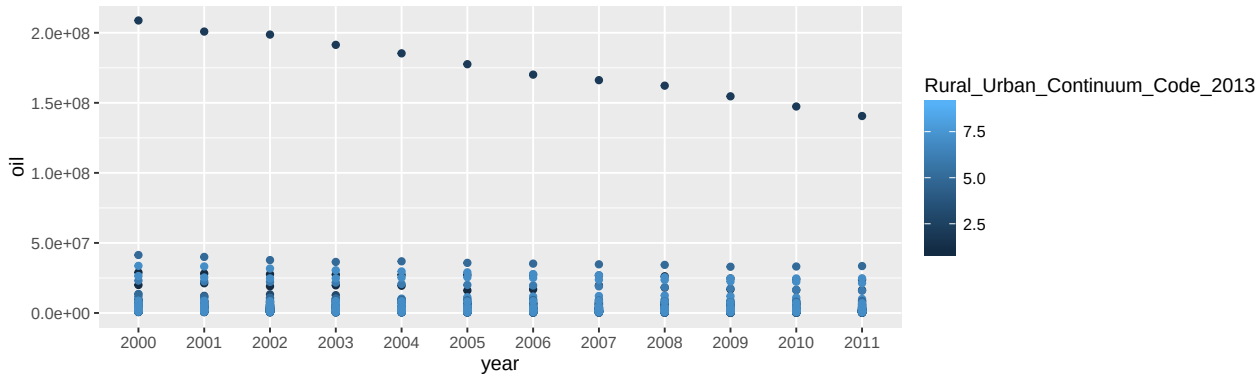


6) on change_group

```
#oil
ggplot(data = subset(tidy,subset = oil_change_group=="H_Growth"))+
  geom_point(aes(x = year, y = oil,
                 color = Rural_Urban_Continuum_Code_2013))
```

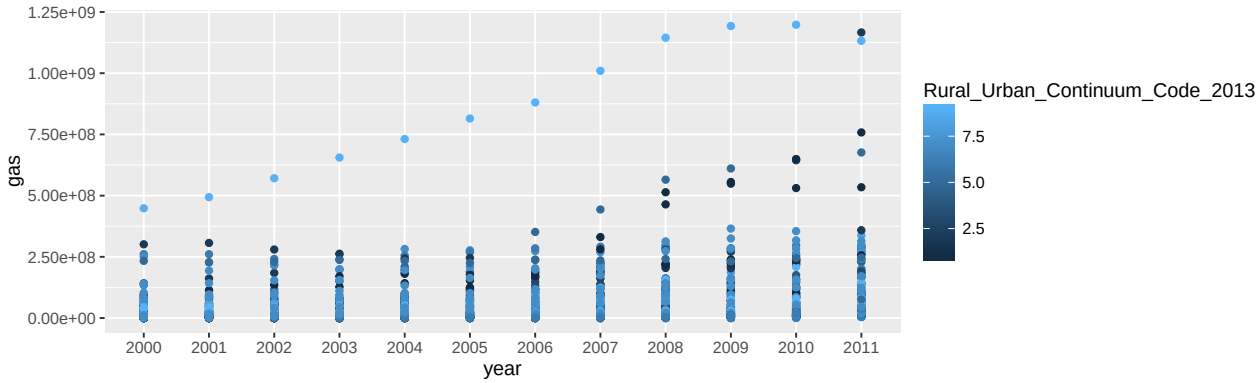



```
ggplot(data = subset(tidy,subset = oil_change_group=="H_Decline"))+
  geom_point(aes(x = year, y = oil,
                 color = Rural_Urban_Continuum_Code_2013))
```

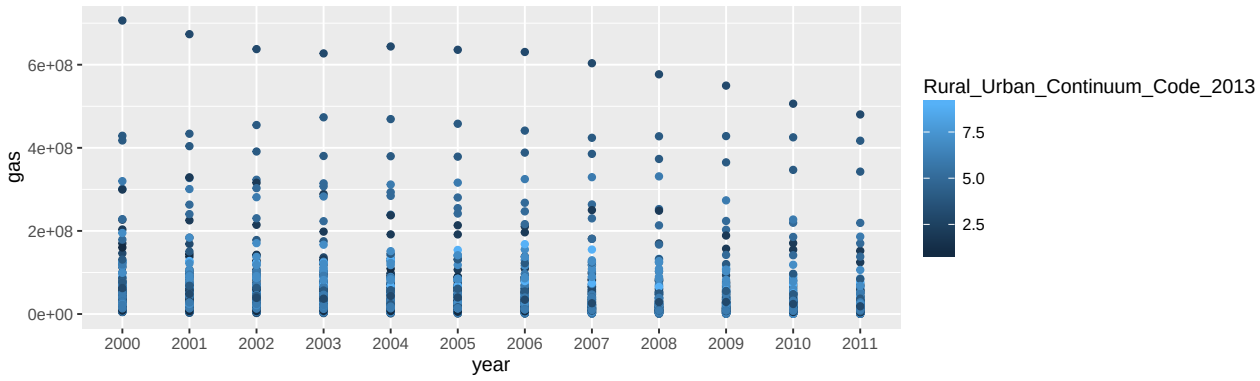



```
ggplot(data = subset(tidy,subset = oil_change_group=="Status Quo"))+
  geom_point(aes(x = year, y = oil,
                 color = Rural_Urban_Continuum_Code_2013))
```

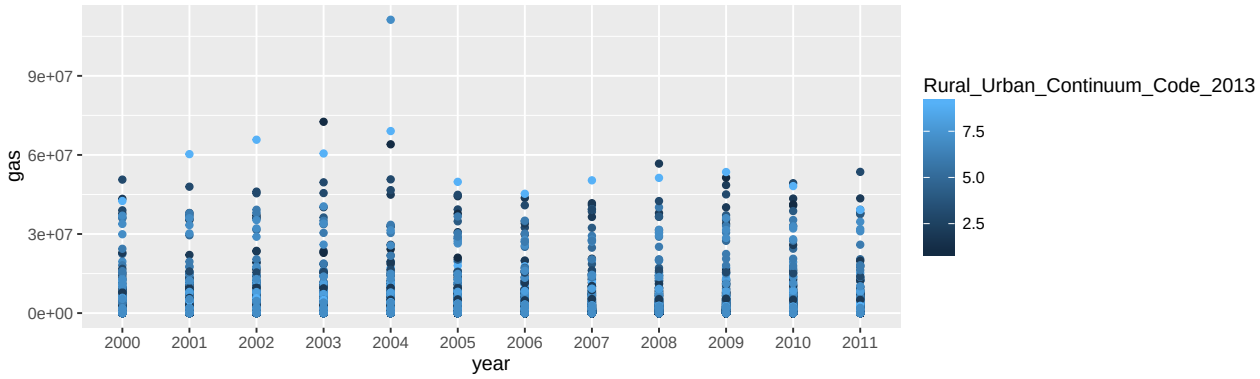

```
#gas
ggplot(data = subset(tidy,subset = gas_change_group=="H_Growth"))+
  geom_point(aes(x = year, y = gas,
                 color = Rural_Urban_Continuum_Code_2013))
```




```
ggplot(data = subset(tidy,subset = gas_change_group=="H_Decline"))+
  geom_point(aes(x = year, y = gas,
                 color = Rural_Urban_Continuum_Code_2013))
```



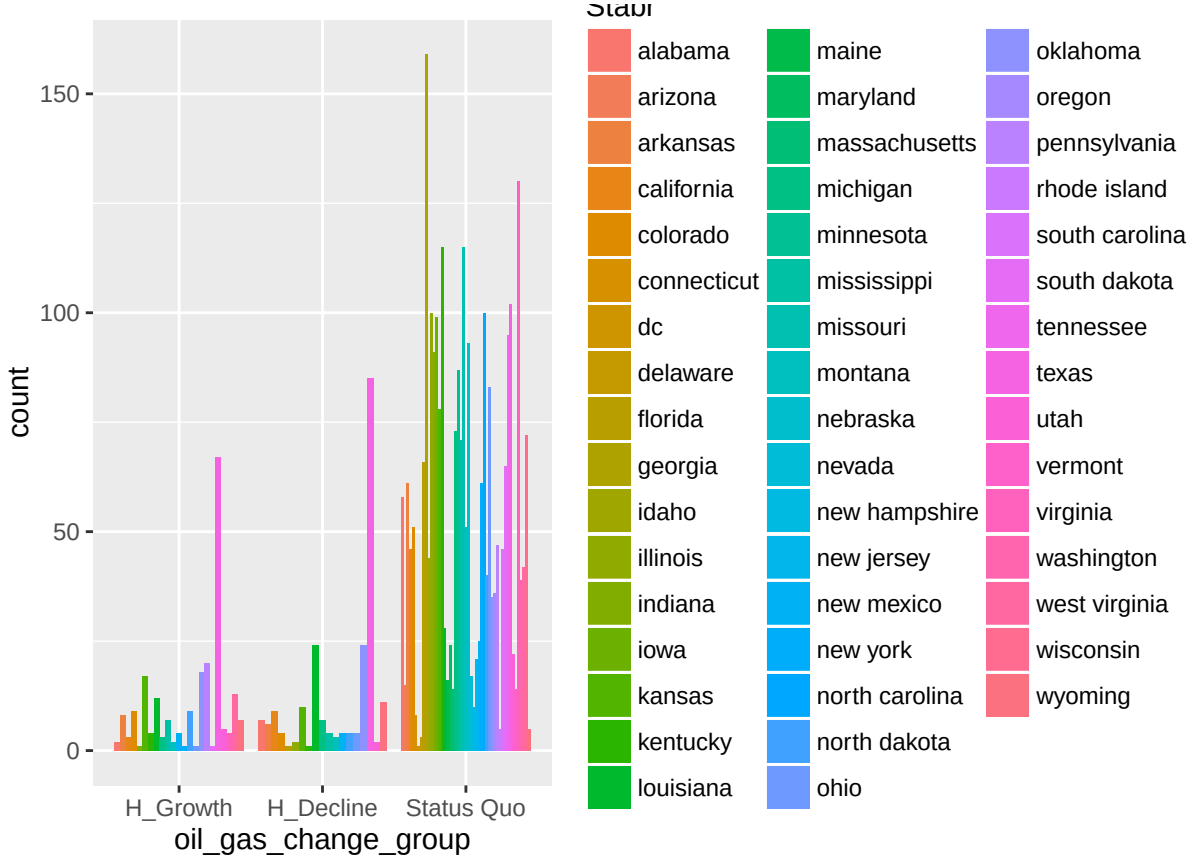

```
ggplot(data = subset(tidy,subset = gas_change_group=="Status Quo"))+
  geom_point(aes(x = year, y = gas,
                 color = Rural_Urban_Continuum_Code_2013))
```



3.2 Explore on the second tidy dataset(non-year variables)

1) State Status

```
ggplot(data = tidy2, aes(x = oil_gas_change_group, fill = Stabr)) +
  geom_bar(position = "dodge")
```



2) Choropleth map

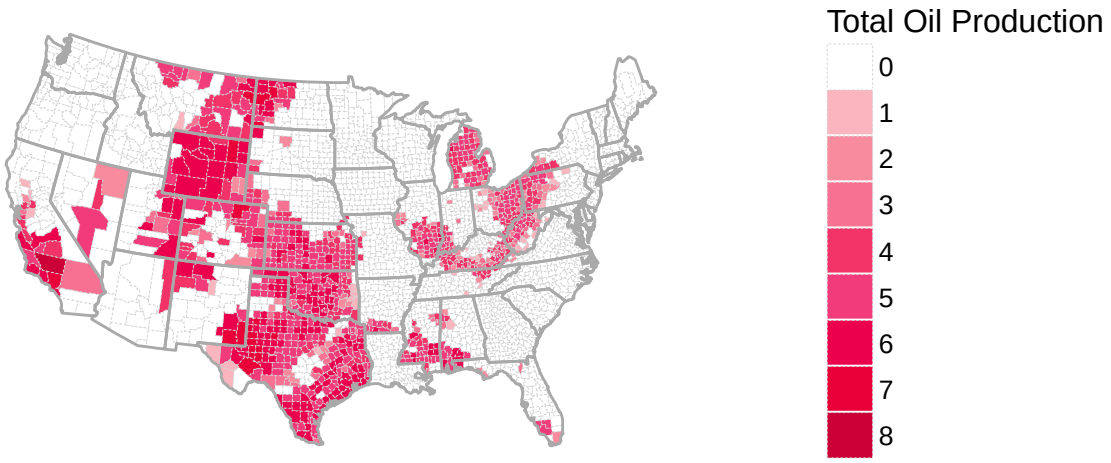
```
#pullout the level and region information
pullout <- oilgas[,c(2,3,36,38)]
colnames(pullout)[1] <- "region"
colnames(pullout)[2] <- "subregion"

#get map data for US counties and states
county_map <- map_data("county")
```

```
##
## Attaching package: 'maps'

## The following object is masked from 'package:plyr':
##
##     ozone
```

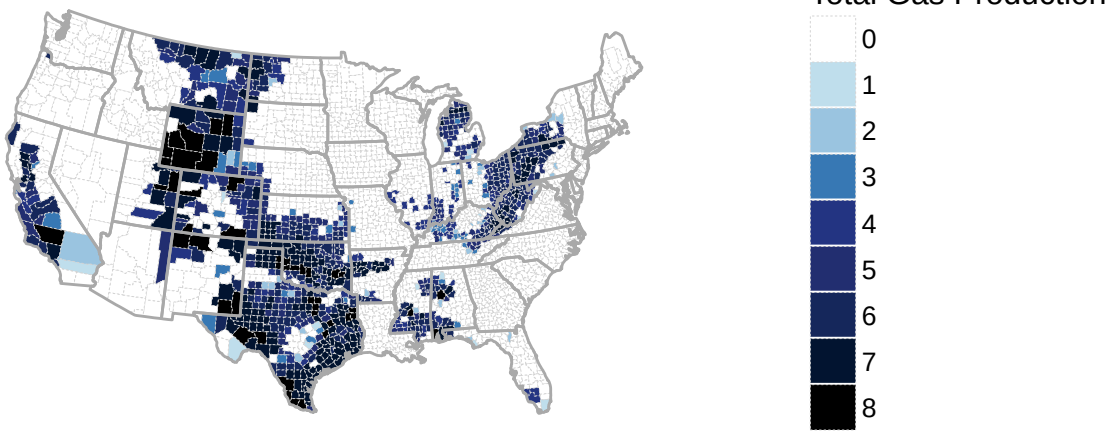
```
state_map <- map_data("state")

#merge pullout and county_map
pullout_map <- merge(county_map, pullout,
                     by.x=c("region", "subregion"),
                     by.y=c("region", "subregion"),  all.x=TRUE)
```

```r
#resort merged data
pullout_map <- arrange(pullout_map, group, order)

#relpace NA with 0's
pullout_map[is.na(pullout_map)] <- 0

#generate a disctrete color pallette
pal_red <- c("#ffffff","#fbb5bf","#f88b9e","#f67192",
             "#f13367","#f23b7a","#eb004e","#ea0039","#cb0036")
pal_blue <- c("#ffffff","#bfdeec","#9ac4e0","#3778b4",
              "#233482","#222e70","#15275b","#021430","#000000")

theme_clean <- function(base_size = 12) {
  require(grid)
  theme_grey(base_size) %+replace%
    theme(
      axis.title      =   element_blank(),
      axis.text       =   element_blank(),
      panel.background    =   element_blank(),
      panel.grid      =   element_blank(),
      axis.ticks.length   =   unit(0,"cm"),
      axis.ticks.margin   =   unit(0,"cm"),
      panel.margin    =   unit(0,"lines"),
      plot.margin     =   unit(c(0,0,0,0),"lines"),
      complete = TRUE
    )
}

#choropleth map on oil level
ggplot( pullout_map, aes( x = long , y = lat , group=group ) ) +
  geom_polygon(linetype = 2, size = 0.1, colour = "lightgrey",
               aes( fill = factor(oil_level) ) ) +
  scale_fill_manual( values = pal_red ) +
  expand_limits( x = pullout_map$long, y = pullout_map$lat ) +
  coord_map( "polyconic" ) +
  labs(fill="Total Oil Production") +
  theme_clean( )+
  geom_path( data = state_map, color = "darkgrey")
```

```
## Loading required package: grid
```




```
#choropleth map on gas level
ggplot( pullout_map, aes( x = long , y = lat , group=group ) ) +
  geom_polygon(linetype = 2, size = 0.1, colour = "lightgrey" ,
               aes( fill = factor(gas_level) ) ) +
  scale_fill_manual( values = pal_blue ) +
  expand_limits( x = pullout_map$long, y = pullout_map$lat ) +
  coord_map( "polyconic" ) +
  labs(fill="Total Gas Production") +
  theme_clean( )+
  geom_path( data = state_map, color = "darkgrey")
```




```
rm(a,i)
```