Stats506 hw2_1

David Li

October 22, 2017

Stats506: Problem 1 Code
Data Used: http://www.eia.gov/consumption/residential/data/2009/csv/recs2009_public.csv
Submitted by: David Li

```
# Read in the RECS.csv file for data parsing from local drive
recs_tib = read.csv("~/Desktop/Stats506/Datasets/recs2009_public.csv")
```

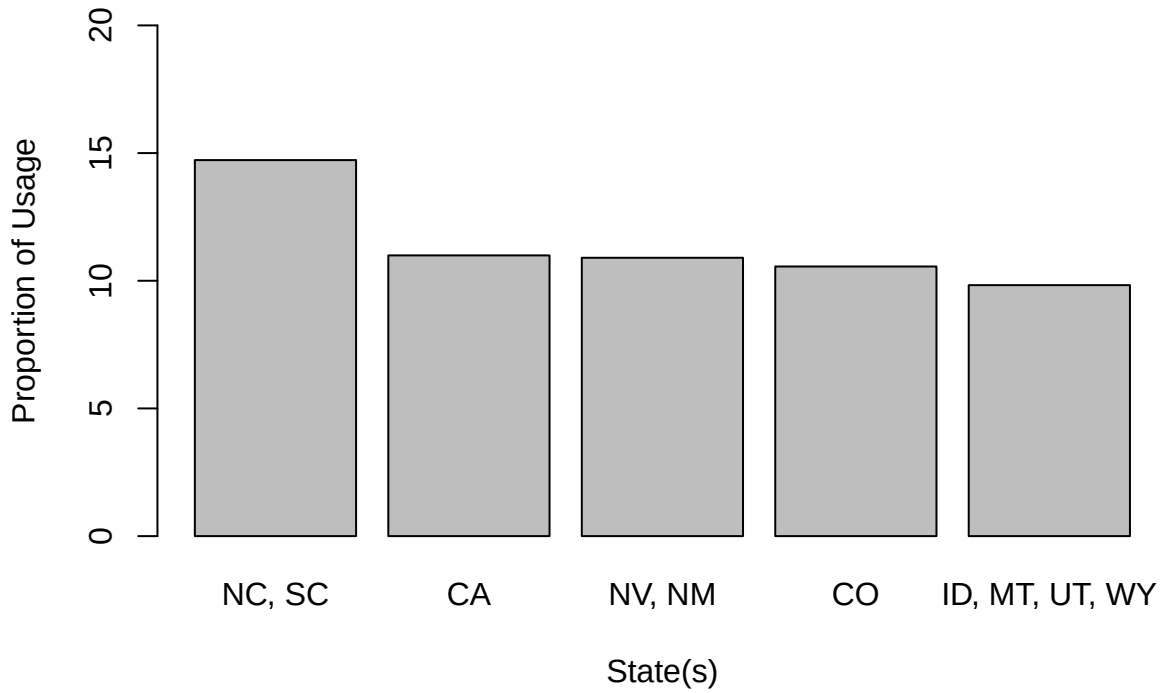
```
# Computing Roof Type Proportions
roof_type_prop = recs_tib %>%
  # Some data management including: subsetting, filtering, grouping, transforming from long to wide dat
  transmute(UniqueId = DOEID, State = REPORTABLE_DOMAIN, RoofType = ROOFTYPE, YearMade = YEARMADE,
    YearMadeDecade = YEARMADERANGE, Weight = NWEIGHT) %>%
  filter(RoofType != -2) %>% # Discarding N/A Data
  mutate(State = decode_all_states(State), RoofType = decode_all_roof_types(RoofType),
    YearMadeDecade = decode_all_decade_ranges(YearMadeDecade)) %>%
  group_by(State, RoofType) %>%
  summarize(Homes = sum(Weight)) %>%
  tidyr::spread(RoofType,Homes) %>%
  rowwise() %>%
  # Computing Proportions
  mutate(Total = sum(Asphalt, CeramicClay, CompShing, Concrete_Tiles, Metal, Other, Slate,
    WoodShing, na.rm = TRUE),
      Asphalt = 100*Asphalt/Total,
      CeramicClay = 100*CeramicClay/Total,
      CompShing = 100*CompShing/Total,
      Concrete_Tiles = 100*Concrete_Tiles/Total,
      Metal = 100*Metal/Total,
      Other = 100*Other/Total,
      Slate = 100*Slate/Total,
      WoodShing = 100*WoodShing/Total
  ) %>%
  select(-Total) %>% # Not needed in displaying
  # Arranging by greatest proportion of wood shingles roofs
  arrange(desc(WoodShing))
# Table
kable(roof_type_prop, digits = 2, caption='Proportion of roof types by State(s).')
```

Table 1: Proportion of roof types by State(s).

State	Asphalt	CeramicClay	CompShing	Concrete_Tiles	Metal	Other	Slate	WoodShing
NC, SC	20.5	NA	51	0.32	11.17	0.35	1.60	14.7
CA	8.8	16.34	52	5.38	3.15	1.10	2.52	11.0
NV, NM	18.1	24.29	23	2.65	11.20	8.12	1.29	10.9
CO	19.8	0.65	55	2.05	9.80	1.77	0.28	10.6
ID, MT, UT, WY	45.1	NA	34	0.60	8.83	0.69	0.60	9.8
TX	2.9	0.91	77	0.42	8.21	0.32	0.53	9.3
FL	18.6	7.60	41	3.16	18.05	1.42	1.59	8.1

State	Asphalt	CeramicClay	CompShing	Concrete_Tiles	Metal	Other	Slate	WoodShing
AK, HI, OR, WA	6.5	1.23	74	0.61	9.12	1.32	0.19	7.5
IN, OH	19.8	0.50	63	NA	8.50	0.30	1.63	6.8
DE, DC, MD, WV	13.2	0.59	65	NA	9.81	0.86	4.10	6.8
PA	19.0	NA	59	NA	5.92	6.66	3.00	6.7
GA	13.9	1.85	72	0.24	4.13	0.50	0.74	6.6
AZ	7.4	31.87	23	14.76	11.94	4.12	0.56	6.1
MA	54.6	0.54	34	NA	1.62	1.58	1.09	6.1
NY	36.4	NA	49	0.34	4.46	2.54	1.79	5.9
MO	13.0	0.71	68	0.56	9.78	0.92	1.26	5.7
KS, NE	20.6	0.61	69	NA	4.15	0.25	0.33	5.2
CT, ME, NH, RI, VT	33.5	NA	48	0.25	11.37	0.22	1.66	4.8
IA, MN, ND, SD	44.6	0.16	45	NA	4.42	0.52	0.42	4.6
IL	30.7	NA	61	0.44	1.93	0.50	0.93	4.3
VA	14.8	0.38	66	NA	14.43	NA	1.00	3.8
WI	51.0	2.20	39	NA	2.33	0.51	0.82	3.8
AR, LA, OK	14.2	1.19	68	NA	12.08	NA	1.48	3.5
AL, KY, MS	10.0	0.34	61	0.24	25.26	0.28	NA	3.4
MI	16.7	0.41	67	NA	9.48	1.42	1.99	3.4
NJ	19.4	NA	74	2.62	0.72	NA	NA	2.9
TN	25.0	NA	58	0.45	13.96	0.68	0.46	1.8

```
# Barplot of Top Wood Shingle Usage
top5 = roof_type_prop[1:5,]
barplot(top5$WoodShing, main = "Top 5 States that use Wood Shingles",
        ylim = c(0, 20), names.arg=c("NC, SC", "CA", "NV, NM", "CO", "ID, MT, UT, WY"), xlab = "State(s)
```



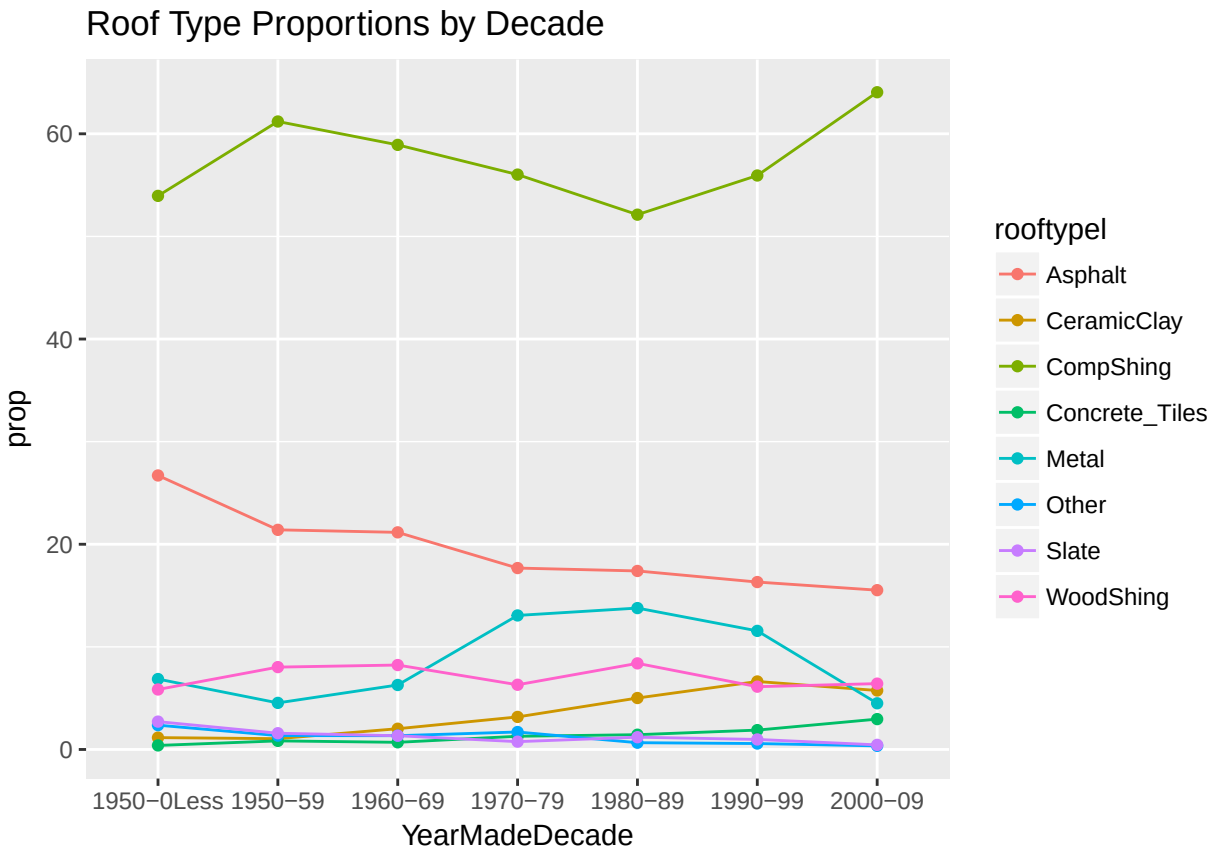
From our computed proportions, we see that North Carolina and South Carolina have the highest rooftype usage of Wood Shingles at 14.7%.

```r
# Computing Roof Type Proportions conditioned by Decade
roof_type_decade = recs_tib %>%
  # Some data management including: subsetting, filtering, grouping, transforming from long to wide dat
  transmute(UniqueId = DOEID, State = REPORTABLE_DOMAIN, RoofType = ROOFTYPE, YearMade = YEARMADE, Yearl
  filter(RoofType != -2) %>% # Discarding N/A Data
  mutate(State = decode_all_states(State), RoofType = decode_all_roof_types(RoofType), YearMadeDecade =
  group_by(YearMadeDecade, RoofType) %>%
  summarize(Homes = sum(Weight)) %>%
  tidyr::spread(RoofType, Homes) %>%
  rowwise() %>%
  # Computing Proportions
  mutate(Total = sum(Asphalt, CeramicClay, CompShing, Concrete_Tiles, Metal, Other, Slate,
      WoodShing, na.rm = TRUE),
         Asphalt = 100*Asphalt/Total,
         CeramicClay = 100*CeramicClay/Total,
         CompShing = 100*CompShing/Total,
         Concrete_Tiles = 100*Concrete_Tiles/Total,
         Metal = 100*Metal/Total,
         Other = 100*Other/Total,
         Slate = 100*Slate/Total,
         WoodShing = 100*WoodShing/Total
  ) %>%
  select(-Total) # Not needed in displaying
roof_type_decade_rorder = roof_type_decade[c(1,2,3,4,5,6,7),] # Rearranging columns since numbers are c
# Table
kable(roof_type_decade_rorder, digits=2, caption='Proportion of roof types by Decade(s).')
```

Table 2: Proportion of roof types by Decade(s).

YearMadeDecade	Asphalt	CeramicClay	CompShing	Concrete_Tiles	Metal	Other	Slate	WoodShing
1950-0Less	27	1.2	54	0.39	6.9	2.37	2.71	5.8
1950-59	21	1.1	61	0.84	4.5	1.35	1.58	8.0
1960-69	21	2.0	59	0.70	6.3	1.35	1.34	8.2
1970-79	18	3.2	56	1.29	13.1	1.70	0.76	6.3
1980-89	17	5.0	52	1.44	13.8	0.66	1.20	8.4
1990-99	16	6.6	56	1.89	11.6	0.58	0.98	6.1
2000-09	16	5.8	64	2.96	4.5	0.35	0.45	6.4

```r
# Converting to long data form to prepare for line plot display
roof_type_decade_long = roof_type_decade_rorder %>%
  gather(rooftypel, prop, Asphalt:WoodShing)
ggplot(roof_type_decade_long, aes(x = YearMadeDecade, y = prop, group = rooftypel, colour = rooftypel))
  geom_line() + geom_point() + ggtitle("Roof Type Proportions by Decade")
```



A View of rooftype usage by decade; good to thoroughly investigate the data.

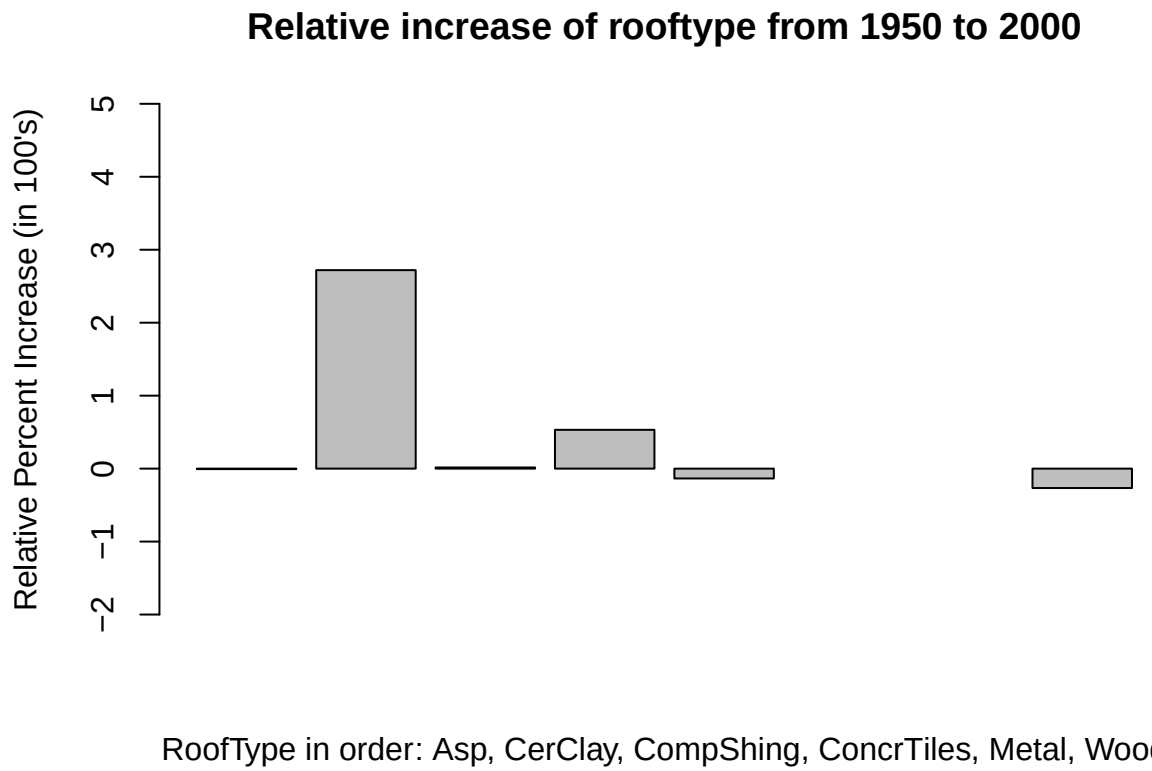
```
# Computing Relative Roof Type Usage between 1950 and 2000
roof_type_1950_2000 = recs_tib %>%
  # Some data management including: subsetting, filtering, grouping, transforming from long to wide dat
  transmute(UniqueId = DOEID, State = REPORTABLE_DOMAIN, RoofType = ROOFTYPE, YearMade = YEARMADE, Yearl
  filter(RoofType != -2) %>% # Discarding N/A Data
  filter(YearMade == c(1950, 2000)) %>% # Only care about years 1950 and 2000
  mutate(State = decode_all_states(State), RoofType = decode_all_roof_types(RoofType), YearMadeDecade =
  group_by(YearMade, RoofType) %>%
  summarize(Homes = sum(Weight)) %>%
  tidyr::spread(RoofType, Homes) %>%
  rowwise() %>%
  # Computing Relative Increase amount
  mutate(Total = sum(Asphalt, CeramicClay, CompShing, Concrete_Tiles, Metal, Other, Slate,
                     WoodShing, na.rm = TRUE),
         Asphalt = 100*Asphalt/Total,
         CeramicClay = 100*CeramicClay/Total,
         CompShing = 100*CompShing/Total,
         Concrete_Tiles = 100*Concrete_Tiles/Total,
         Metal = 100*Metal/Total,
         Other = 100*Other/Total,
         Slate = 100*Slate/Total,
         WoodShing = 100*WoodShing/Total
  ) %>%
  select(-Total) # Not needed in displaying
  for(i in 2:9){
```

```
    roof_type_1950_2000[3,i] = Increase_percent(roof_type_1950_2000[1,i], roof_type_1950_2000[2,i])
  }
  roof_type_1950_2000[3,1] = "Relative Increase from 1950 to 2000"
  # Arranging by rooftype with greatest relative increase from 1950 to 2000
  #arrange(desc(Relative_increase_percent))
# Table
kable(roof_type_1950_2000, digits=2, caption='Relative Increase of RoofType Usage from 1950 to 2000.')
```

Table 3: Relative Increase of RoofType Usage from 1950 to 2000.

YearMade	Asphalt	CeramicClay	CompShing	Concrete_Tiles	Metal	Other	Slate	W
1950	20.44	2.0	58.42	1.41	5.70	1.4	2.7	
2000	20.30	7.5	59.34	2.16	4.93	NA	NA	
Relative Increase from 1950 to 2000	-0.01	2.7	0.02	0.53	-0.13	NA	NA	

```
# Barplot of Increases
ex_roof_type_1950_2000 = as.numeric(roof_type_1950_2000[-c(1, 2),-c(1)])
barplot(ex_roof_type_1950_2000, main = "Relative increase of rooftype from 1950 to 2000",
        ylim = c(-2, 5), xlab = "RoofType in order: Asp, CerClay, CompShing, ConcrTiles, Metal, Wood", y
```



Subsetting by years of 1950 and 2000 and rooftype usage in that particular year, it seems that Ceramic and Clay Tiles had the largest jump at a 270% increase from 1950 to 2000.

```
## Appendix ##
## Packages Utilized ##
library("dplyr", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")
library("ggplot2", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")
library("knitr", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")
```

```r
library("rmarkdown", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")
library("stringdist", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")
library("stringr", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")
library("tidyr", lib.loc="~/R/x86_64-pc-linux-gnu-library/3.4")

# Begin Functions Section #
decode_state = function(x){ # Decodes the numeric representation of states into their actual names
  if(!is.numeric(x)){
    stop('decode_state expects numeric input indexed from 1!')
  }
  switch(x,
         "CT, ME, NH, RI, VT", "MA", "NY", "NJ", "PA", "IL", "IN, OH", "MI", "WI",
         "IA, MN, ND, SD", "KS, NE", "MO", "VA", "DE, DC, MD, WV", "GA",
         "NC, SC" , "FL", "AL, KY, MS", "TN", "AR, LA, OK",
         "TX", "CO", "ID, MT, UT, WY", "AZ", "NV, NM",
         "CA", "AK, HI, OR, WA"
  )
}
decode_all_states = function(x){ # Applies decoding to a vector instead of a single value
  sapply(x, decode_state)
}
decode_roof_type = function(x){ # Decodes the numeric representation of rooftypes into their actual nam
  if(!is.numeric(x)){
    stop('decode_roof_type expects numeric input indexed from 1!')
  }
  # Not Applicable values will be filtered out later
  switch(x,
         "CeramicClay", "WoodShing", "Metal",
         "Slate", "CompShing", "Asphalt",
         "Concrete_Tiles", "Other"
  )
}
decode_all_roof_types = function(x){ # Applies decoding to a vector instead of a single value
  sapply(x, decode_roof_type)
}
decode_decade_range = function(x){ # Decodes the numeric representation of decades into their actual na
  if(!is.numeric(x)){
    stop('decode_decade_range expects numeric input indexed from 1!')
  }
  switch(x,
         "1950Less", "1950-59", "1960-69", "1970-79",
         "1980-89", "1990-99", "2000-09", "2000-09" # This effectively combines '00 - '04 and '05 - '09
  )
}
decode_all_decade_ranges = function(x){ # Applies decoding to a vector instead of a single value
  sapply(x, decode_decade_range)
}
Increase_percent = function(a,b){ # Calculate relative increase percentage
  return((b-a) / a)
}
# End Functions Section #
```