Parcel Taxpayers

Tyler Hart

March 31, 2019

Spokane county makes data available about who the taxpayer is for a given parcel of land and what their address is. It's presented in an Excel-formatted file as "taxpayer_info.xlsx". For those who might not be aware, Spokane County encompasses the following cities and towns:

- Airway Heights
- Cheney
- Deer Park
- Fairfield
- Latah
- Liberty Lake
- Medical Lake
- Millwood
- Rockford
- Spangle
- Spokane Valley
- Waverly

The parcel data includes much more than just land in the city of Spokane. While most of the data is complete and usable, some of it is not. Since the data isn't yet complete, we'll set the *stringsAsFactors* option to *FALSE* before importing from the CSV:

```
options("stringsAsFactors" = FALSE)
```

First, we should open the Excel file and do a "Save As" into CSV format. While R can do file format conversions via the Rio package, I find it easier to do the conversion manually for one-off analysis. Next, we'll import data from the provided CSV file:

```
taxpayer_info <- read_csv(
  "taxpayer_info.csv",
  col_types = cols(
    parcel = col_character(),
    taxpayer = col_character(),
    address_1 = col_character(),
    address_2 = col_character(),
    city = col_character(),
    state = col_character(),
    zip = col_character(),
    country = col_character(),
    role_percentage = col_double(),
    in_care_of = col_character()
    )
  )
```

The raw data provided by Spokane County looks like this:

```
head(taxpayer_info)
```

```
## # A tibble: 6 x 10
##   parcel taxpayer address_1 address_2 city  state zip   country
##   <chr>  <chr>    <chr>     <chr>     <chr> <chr> <chr> <chr>
## 1 00.00~ A & D Q~ 15205 E ~ <NA>      VERA~ WA    99037 <NA>
## 2 00.00~ O'CONNO~ 905 W RI~ <NA>      SPOK~ WA    99201 USA
## 3 00.00~ COLUMBI~ 701 MILL~ <NA>      GREE~ SC    29607 <NA>
## 4 00.00~ ONE BRI~ 1817 N D~ <NA>      SPOK~ WA    99207 USA
## 5 00.00~ HALL OF~ PO BOX 3~ <NA>      LIBE~ WA    99019 <NA>
## 6 00.00~ ACC LIN~ 501 E SP~ <NA>      SPOK~ WA    99202 <NA>
## # ... with 2 more variables: role_percentage <dbl>, in_care_of <chr>
```

Now that the data has been pulled into R it's time to do some wrangling.

Data Wrangling

We'll go through a number of steps to update the data, mostly filling in missing fields and fixing spelling. We didn't want the parcel or address character-type columns to be imported automatically as factors, hence the option setting *stringsAsFactors* to *FALSE*. Later we'll convert some of the columns like *city* and *state* to factors for easier analysis, but only after cleaning up the data.

Not all rows have a country specified, even though their address is clearly in the United States. Here are some examples of WA addresses not being in the USA:

```
head(taxpayer_info %>%
       filter(state == "WA" & is.na(country)) %>%
       select(parcel, taxpayer, state, country))
```

```
## # A tibble: 6 x 4
##   parcel    taxpayer                           state country
##   <chr>     <chr>                              <chr> <chr>
## 1 00.000014 A & D QUALITY HYDROSEEDING         WA    <NA>
## 2 00.000019 HALL OF BOOKS                      WA    <NA>
## 3 00.000024 ACC LINEN SALES & GARMENT PRINTING WA    <NA>
## 4 00.000026 A1 ELECTRONIC ELECTRICIAN          WA    <NA>
## 5 00.000027 EVERGREEN APPRAISAL SERVICE        WA    <NA>
## 6 00.000028 LANA'S HAIR & NAIL DESIGN          WA    <NA>
```

We can safely assume that an address in WA state - or any other American state - is in the USA. Other state designations used for military mail like "AA", "AE", and "AP" should also be considered in the USA. Territories like Puerto Rico ("PR") and Guam ("GM") are also in the United States for our purposes, even if they aren't part of the continental United States or officially given full statehood. The District of Columbia ("DC") is obviously in the United States as well. We'll fill in "USA" country values for more complete data:

```
# Offical states
taxpayer_info$country <- ifelse(
  taxpayer_info$state %in% state.abb,
  "USA",
  taxpayer_info$country)

# Territories, military, and DC
taxpayer_info$country <- ifelse(
  taxpayer_info$state %in% c("AA", "AE", "AP", "DC", "PR", "GM"),
  "USA",
  taxpayer_info$country)
```

Here's the new data with filled-in country column:

```
head(taxpayer_info %>% select(parcel, taxpayer, state, country))
```

```
## # A tibble: 6 x 4
##   parcel    taxpayer                           state country
##   <chr>     <chr>                              <chr> <chr>
## 1 00.000014 A & D QUALITY HYDROSEEDING         WA    USA
## 2 00.000016 O'CONNOR, MONAGHAN & SOMERS PS     WA    USA
## 3 00.000017 COLUMBIA LIGHTING MFG COMPANY      SC    USA
## 4 00.000018 ONE BRIDGE NORTH TAVERN            WA    USA
## 5 00.000019 HALL OF BOOKS                      WA    USA
## 6 00.000024 ACC LINEN SALES & GARMENT PRINTING WA    USA
```

That takes care of states in the USA, but not all taxpayers have American addresses. Unfortunately, not all non-USA countries are formatted consistently in the data. The ISO 3166 standard sets out standardized two and three-letter abbreviations for countries, and is the preferred way of storing location information. Data from Spokane County sometimes has country abbreviations with two letters, while other countries like Canada are entered as "CANDA". Australia is abbreviated as "AUSTR" in the data, which could be confused with "Austria". We need to fix non-USA country names so they are usable as factors and for visualizations. We'll do selective replacements using abbreviations within the data and their respective country names:

```
country_names <- data.frame(
  "country" = c(
    "CANDA",
    "AUSTR",
    "HK",
    "DEN",
    "FIN",
    "GERMY",
    "JAPAN",
    "MEXCO",
    "PHILI",
    "SAUDI",
    "USA",
    "SWLAN",
    "ROC"
    ),
  "fixed_country" = c(
    "Canada",
    "Australia",
    "Hong Kong",
    "Denmark",
    "Finland",
    "Germany",
    "Japan",
    "Mexico",
    "Phillipines",
    "Saudi Arabia",
    "USA",
    "Switzerland",
    "Republic of China"
    )
)

taxpayer_info <- left_join(taxpayer_info, country_names, by = "country")
taxpayer_info$country <- taxpayer_info$fixed_country # Replace old data with new
taxpayer_info$fixed_country <- NULL # Clean up
```

Unlike rows that we just fixed with missing countries, dozens of rows assigned "Spokane" *city*, "99XYZ" *zip*, and "USA" *country* are missing *state* ("WA") data. We'll update these rows with "WA" values for *state*:

```
taxpayer_info$state[
  which(
    taxpayer_info$city == "SPOKANE" & taxpayer_info$country == "USA" & substr(taxpayer_info$zip, 1, 2) =
    )
  ] <- "WA"
```

Almost a dozen other rows have their city set to "Chicago, IL" but no state data - they are obviously in Illinois so we can fix them too:

```
taxpayer_info$state[which(taxpayer_info$city == "CHICAGO, IL")] <- "IL"
```

A number of other rows have "city, STATE" data for Washington locations in the *city* column but no state. For example, there are a few rows with "Spokane, WA" for *city* and "NA" values for *state*. We'll assign "WA" for *state*, then strike the ", WA" characters from *city*. We'll also do the same for ", IL" data that we already fixed:

```
taxpayer_info$state[which(str_sub(taxpayer_info$city, start = -4) == ", WA")] <- "WA"

for (state_str in c(", IL", ", WA")) {
  taxpayer_info$city <- gsub(state_str, "", taxpayer_info$city)
}
```

Now that city, state, and country names are squared-up, we'll remove columns that aren't useful for analysis. For example, the taxpayer's name isn't necessary for analyzing data by country or state. A parcel's street number and name aren't needed either. Most parcels have no data in the second address line field, so we'll drop that as well. The *roll_percentage* field could express multiple levels of ownership, but all 338315 rows have a value of "100", so we'll drop it too:

```
taxpayer_info$taxpayer <- NULL # Names
taxpayer_info$address_1 <- NULL # Street address
taxpayer_info$address_2 <- NULL # Unit number
taxpayer_info$in_care_of <- NULL # Notes
taxpayer_info$role_percentage <- NULL # Ownership percentage?
```

About a half-dozen rows out of hundreds-of-thousands don't have city, state, or country filled in. Rather than guessing we'll just filter out any remaining rows that don't have a country assigned:

```
taxpayer_info <- taxpayer_info %>% filter(!is.na(country))
```

Some rows remain that are missing state or zip code information, but that shouldn't affect our analysis in a material way. This leaves us with a total of 338304 rows to analyze. Finally, we'll set the *country* column as a factor for easier analysis:

```
taxpayer_info$country <- factor(taxpayer_info$country, levels = country_names$fixed_country)
```

Let's look at a table of parcel counts broken up by country:

```
sort(table(taxpayer_info$country), decreasing = TRUE)
```

```
## 
##                USA            Canada            Mexico        Phillipines 
##             337023              1191                39                 19 
##          Australia           Denmark             Japan            Germany 
##                 10                 6                 6                  5 
##          Hong Kong           Finland      Saudi Arabia        Switzerland 
##                  1                 1                 1                  1 
## Republic of China
##                    1
```

It looks like the vast majority of parcels are owned by taxpayers (people or organizations) with addresses in the United States. The next largest number of parcels is owned by taxpayers with addresses in Canada. Here's the count of Canadian taxpayer addresses:

```
sum(taxpayer_info$country == "Canada")
```

```
## [1] 1191
```

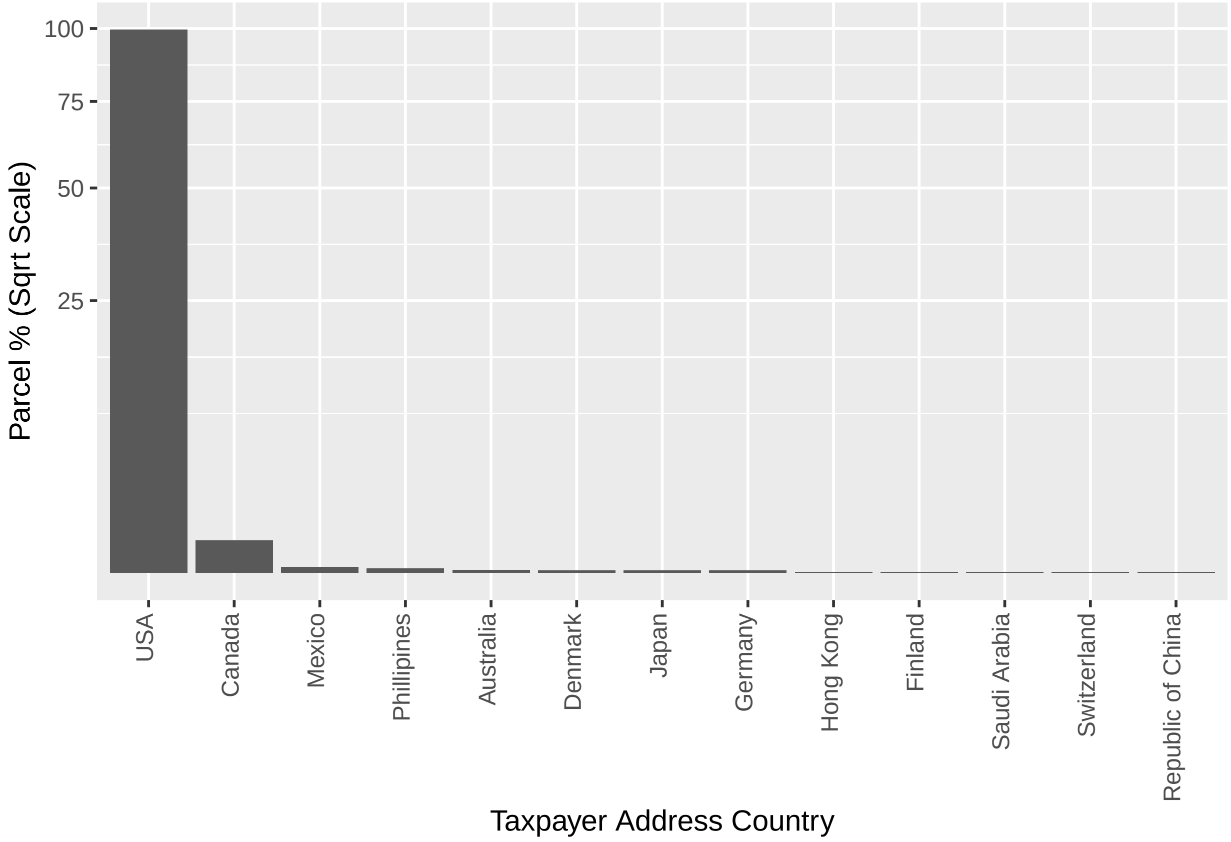
Canadian taxpayers seem to own more parcels than all of the other non-USA countries combined - but is that a lot? Here's the overall *percentage* of parcels with Canadian taxpayers:

```
(sum(taxpayer_info$country == "Canada") / length(taxpayer_info$country)) * 100
```

```
## [1] 0.3520502
```

Needless to say, less than half of one percent isn't a lot. The ownership percentage of parcels by taxpayers with addresses in other countries besides Canada are so small they aren't worth mentioning. Here's a graph:

```
taxpayer_info %>%
  count(country) %>%
  mutate("perc" = (n / nrow(taxpayer_info)) * 100) -> country_percs
country_percs %>% ggplot(aes(reorder(country, -perc), perc)) +
  geom_bar(stat = "identity") +
  scale_y_sqrt() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.25)) +
  xlab("Taxpayer Address Country") +
  ylab("Parcel % (Sqrt Scale)")
```



Non-USA numbers are so small that the Y-axis scale had to be switched to square root (sqrt) to make the data visible. Let's move on to more interesting analysis - how is parcel ownership distributed within the United States? First we'll filter all parcels with *country* "USA". Then we'll create a table with state abbreviations and lowercase names, and join it to the real data. Those lower-case proper names will be used later to draw a map. Finally, we'll set *state* as a factor and take a quick peek at the data:

```
usa_state_taxpayers <- taxpayer_info %>% filter(country == "USA") %>% na.omit()

# State abbrev, names
state_table <- data.frame(
  "state" = state.abb,
  "state_name" = str_to_lower(state.name)
)

# Add proper state names
usa_state_taxpayers <- left_join(usa_state_taxpayers, state_table, by = "state")

# Factor states
usa_state_taxpayers$state <- as.factor(usa_state_taxpayers$state)
head(usa_state_taxpayers)
```

```
## # A tibble: 6 x 6
##   parcel    city         state zip   country state_name
##   <chr>     <chr>        <fct> <chr> <fct>   <chr>
## 1 00.000014 VERADALE     WA    99037 USA     washington
## 2 00.000016 SPOKANE      WA    99201 USA     washington
## 3 00.000017 GREENVILLE   SC    29607 USA     south carolina
## 4 00.000018 SPOKANE      WA    99207 USA     washington
## 5 00.000019 LIBERTY LAKE WA    99019 USA     washington
## 6 00.000024 SPOKANE      WA    99202 USA     washington
```

With each state (and territory) being a factor in R we can easily see how many parcels there are for taxpayers in each state:

```
table(usa_state_taxpayers$state)
```

```
##
##     AA     AE     AK     AL     AP     AR     AZ     BC     CA     CO
##      1     57    284    148     16    137   1088      1   4734   1182
##     CT     DC     DE     FL     GA     GM     HI     IA     ID     IL
##    671     34    127    839    550      4    253    275   3871   3072
##     IN     KS     KY     LA     MA     MD     ME     MI     MN     MO
##    282    397    104    297    222    121     21    285    735    998
##     MS     MT     NC     ND     NE     NH     NJ     NM     NV     NY
##     17    664    343     39    486     44    805    101   3871    482
##     OH     OK     OR     PA     PR     RI     SC     SD     TN     TX
##    537    388   2320    729      1     64    173    178    243   3949
##     UT     VA     VT     WA     WI     WV     WY
##    520    200      9 288753    162     12     56
```

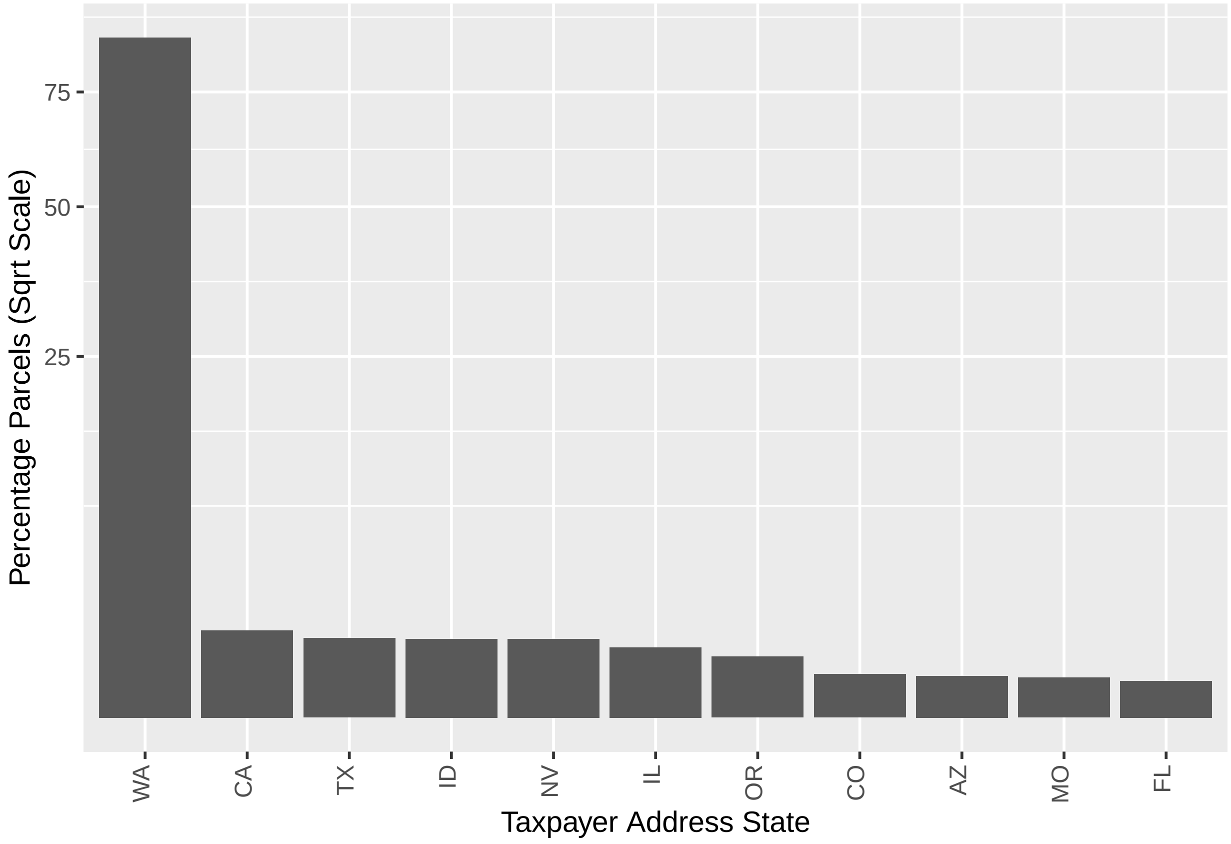
Note that there is one parcel listed under the state BC. Unfortunately in the data entry process for the original CSV there appears to be an error with one row having a Canadian address marked with "USA" for the country. We'll look at the same type of bar chart as we did for countries already, but this time for states in the US:

```
# Count, percentage by state
usa_state_taxpayers_perc <- usa_state_taxpayers %>%
  count(state) %>%
  mutate("perc" = (n / nrow(usa_state_taxpayers)) * 100)

# Add proper names for easier mapping later
usa_state_taxpayers_perc <- left_join(usa_state_taxpayers_perc, state_table, by = "state")
```

```
## Warning: Column `state` joining factor and character vector, coercing into
## character vector
```

```
# Plot parcel taxpayers by state, percentages greater than 0.25%
usa_state_taxpayers_perc %>%
  filter(perc > 0.25) %>%
  ggplot(aes(reorder(state, -perc), perc)) +
  geom_bar(stat = "identity") +
  scale_y_sqrt() +
  xlab("Taxpayer Address State") +
  ylab("Percentage Parcels (Sqrt Scale)") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.25))
```

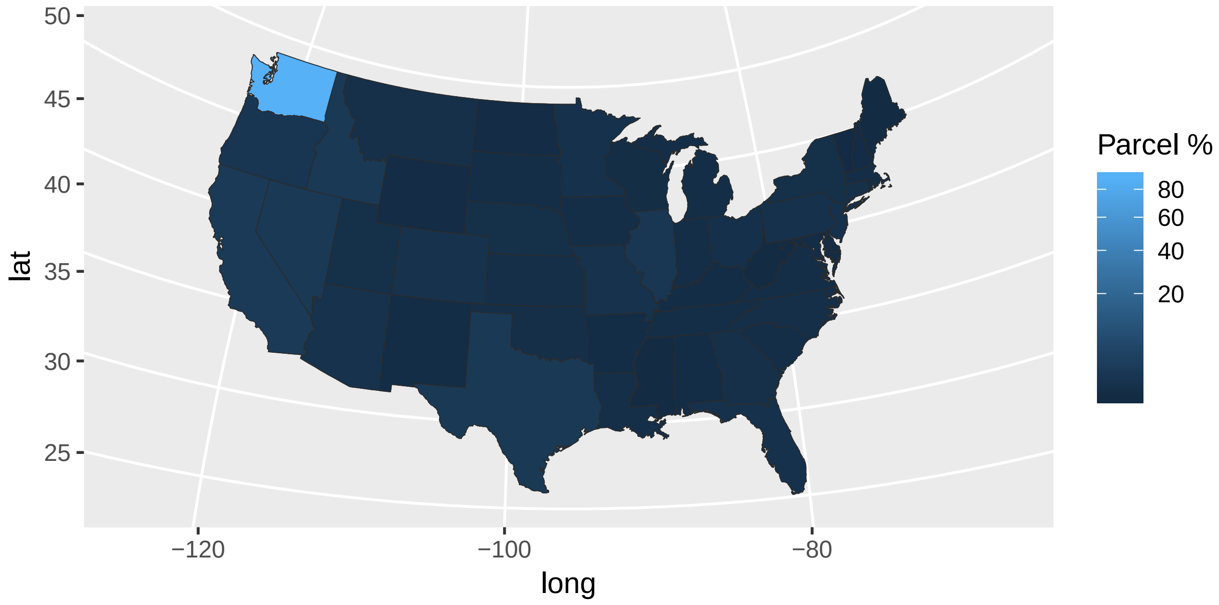


We can build a map of parcel taxpayer states as well:

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

usa_state_taxpayers_perc %>%
  ggplot() +
  geom_map(data = state_map,
         map = state_map,
         aes(
           long,
           lat,
           map_id = region
           ),
         fill="#ffffff", color="#2b2b2b", size=0.15
         ) +
  geom_map(data = usa_state_taxpayers_perc,
           map = state_map,
           aes(
             fill = perc,
             map_id = state_name,
             color = factor(state_name)
             ),
           color="#2b2b2b", size=0.15
           ) +
  scale_fill_continuous(name = "Parcel %", type = "gradient", trans = "sqrt") +
  coord_map("polyconic")
```

```
## Warning: Ignoring unknown aesthetics: x, y
```



By an overwhelming majority it appears that parcels in Spokane County are owned by taxpayers with WA addresses. Total WA taxpayer addresses make up 88.59% of parcels. Next in line is California, which may give weight to anecdotal observations about people from California making local property purchases. However, CA only makes up 1.45% of parcel taxpayer addresses. All other states have parcel taxpayer percentages so small they are difficult to visualize on a colored map.

Let's go one level deeper and look at taxpayers within WA state - how many are from Spokane? First, we'll whittle the data down more and filter by taxpayers in Washington state:

```
wa_taxpayers <- usa_state_taxpayers %>% filter(state == "WA")
head(wa_taxpayers)
```

```
## # A tibble: 6 x 6
##   parcel    city         state zip   country state_name
##   <chr>     <chr>        <fct> <chr> <fct>   <chr>
## 1 00.000014 VERADALE     WA    99037 USA     washington
## 2 00.000016 SPOKANE      WA    99201 USA     washington
## 3 00.000018 SPOKANE      WA    99207 USA     washington
## 4 00.000019 LIBERTY LAKE WA    99019 USA     washington
## 5 00.000024 SPOKANE      WA    99202 USA     washington
## 6 00.000025 GREENACRES   WA    99016 USA     washington
```

The data is complete across all the columns, but I noticed while glancing at the rows in data viewer that there are some spelling and data entry errors. In some cases there are many different ways the same data, like "Spokane", was entered. Here are all the right (and wrong) spellings of "SPOKANE" in the data:

```
unique(wa_taxpayers$city[which(wa_taxpayers$city %like% "SPOKANE")])
```

```
##  [1] "SPOKANE"                        "SPOKANE VALLEY"
##  [3] "SPOKANE, VALLEY"                "SSPOKANE VALLEY"
##  [5] "SPOKANE-"                       "EX15-08390-DTD-6/26/15-SPOKANE"
##  [7] "SPOKANE VALELY"                 "SPOKANE-VALLEY"
##  [9] "SPOKANE VALLY"                  "SPOKANE  WA"
## [11] "SPOKANE VALLLEY"                "SPOKANE VLY"
## [13] "SPOKANEVALLEY"                  "SPOKANE AVLLEY"
```

"NINE MILE FALLS" ends up looking like this in the data:

```
unique(wa_taxpayers$city[which(wa_taxpayers$city %like% "NINE")])
```

```
## [1] "NINE MILE FALLS"  "NINE MILES FALLS" "NINE MILE FALLS*"
## [4] "NINE MILES FLS"   "NINE MILE FALL"   "NINE MILE FLS"
```

There is a bit of work to be done on the names before we can Washington data properly. First, we'll convert all city names to lowercase for easier comparison. Then, we'll correct spellings. Fortunately much of this work can be automated if we provide a list of correct spellings:

```
# Lowercase
wa_taxpayers$city <- str_to_lower(wa_taxpayers$city, locale = "en")

# City names that were misspelled
misspelled_cities <- c(
  "airway heights",
  "bainbridge island",
  "bothell",
  "bellevue",
  "chattaroy",
  "chehalis",
  "cheney",
  "centralia",
  "clarkston",
  "cle elum",
  "clyde hill",
  "colbert",
```

```r
  "deer park",
  "east wenatchee",
  "elk",
  "fairchild",
  "fairfield",
  "friday harbor",
  "gig harbor",
  "greenacres",
  "kirkland",
  "liberty lake",
  "marshall",
  "mead",
  "medical lake",
  "metaline",
  "metaline falls",
  "mountlake terrace",
  "nine mile falls",
  "normandy park",
  "otis orchards",
  "port orchard",
  "rochester",
  "rolling bay",
  "rosalia",
  "spokane",
  "spokane valley",
  "sprague",
  "sumner",
  "tukwila",
  "university place",
  "washougal",
  "west richland",
  "woodinville"
)

# Fix names within 1:3 char of the real name, incrementally
for (iter_count in 1:3) {
  for (city_name in misspelled_cities) {
    wa_taxpayers$city[
      which(stringdist(wa_taxpayers$city, city_name) %in% c(1:iter_count))
      ] <- city_name
  }
}
```

With city names being corrected we can capitalize the names properly and create factors:

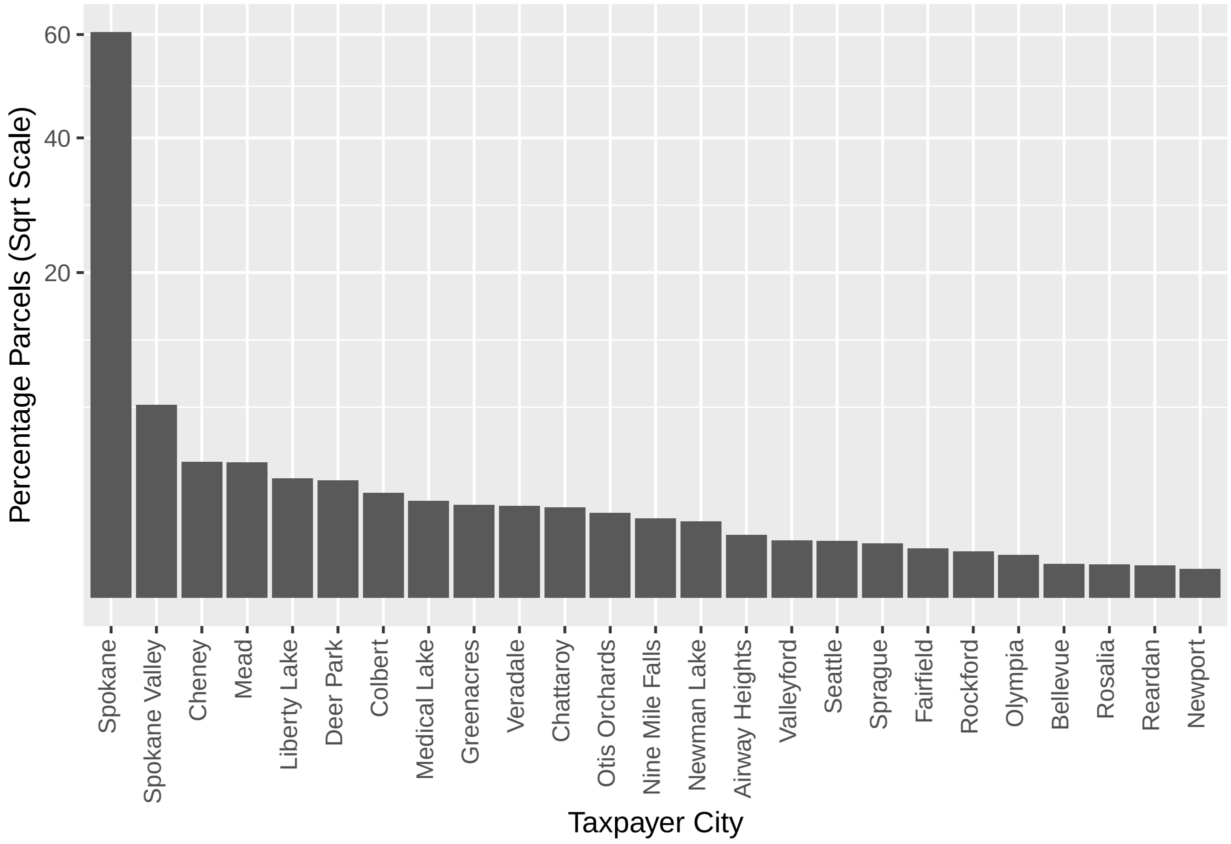
```r
wa_taxpayers$city <- tools::toTitleCase(as.character(wa_taxpayers$city)) # Fix name case
wa_taxpayers$city <- as.factor(wa_taxpayers$city) # Create factor levels
```

The automated fixes aren't perfect, but we only have a few city names that weren't corrected because their spellings or formatting are really out-of-whack. Now we can graph the number of parcels in the Spokane County data with taxpayer addresses in Washington. In this case we'll only show the top 25 cities in WA listed as parcel taxpayer addresses:

```r
# Counts, percentages by city
wa_state_taxpayers_perc <- wa_taxpayers %>%
  count(city) %>%
  mutate("perc" = (n / nrow(wa_taxpayers)) * 100) %>%
  top_n(., 25)
```

```
## Selecting by perc
```

```
# Plot parcel taxpayers by city
wa_state_taxpayers_perc %>% ggplot(aes(reorder(city, -perc), perc)) +
  geom_bar(stat = "identity") +
  scale_y_sqrt() +
  xlab("Taxpayer City") +
  ylab("Percentage Parcels (Sqrt Scale)") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.25))
```



So many parcels have Spokane-addressed taxpayers that we need to use a different scale (Sqrt) for the Y-axis to draw up the other cities for comparison. Taxpayers with Spokane addresses have the lion's share of parcels in Spokane county, following by Spokane Valley and Cheney addresses. The numbers diminish in cities further aware from Spokane.

Percentages

Most people will want to know overall percentages, so we'll break those down by country, state, and city. First, the breakdown by country:

```
country_percentage <- taxpayer_info %>%
  group_by(country) %>%
  summarise(n = n()) %>%
  mutate("percentage" = round((n / sum(n)) * 100, 6)) %>%
  arrange(desc(percentage))

head(country_percentage)
```

```
## # A tibble: 6 x 3
##   country          n percentage
##   <fct>        <int>      <dbl>
## 1 USA         337023   99.6
## 2 Canada        1191    0.352
## 3 Mexico          39    0.0115
## 4 Phillipines     19    0.00562
## 5 Australia       10    0.00296
## 6 Denmark          6    0.00177
```

Now, the breakdown by state:

```
state_percentage <- usa_state_taxpayers %>%
  group_by(state) %>%
  summarise(n = n()) %>%
  mutate("percentage" = round((n / sum(n)) * 100, 6)) %>%
  arrange(desc(percentage))

head(state_percentage)
```

```
## # A tibble: 6 x 3
##   state      n percentage
##   <fct>  <int>      <dbl>
## 1 WA    288753     88.6
## 2 CA      4734      1.45
## 3 TX      3949      1.21
## 4 ID      3871      1.19
## 5 NV      3871      1.19
## 6 IL      3072      0.942
```

Finally, the breakdown by city within Washington state:

```
wa_city_percentage <- wa_taxpayers %>%
  group_by(city) %>%
  summarise(n = n()) %>%
  mutate("percentage" = round((n / sum(n)) * 100, 6)) %>%
  arrange(desc(percentage))

head(wa_city_percentage)
```

```
## # A tibble: 6 x 3
##   city                n percentage
##   <fct>           <int>      <dbl>
## 1 Spokane        174618      60.5
## 2 Spokane Valley  20349       7.05
## 3 Cheney          10088       3.49
## 4 Mead             9989       3.46
## 5 Liberty Lake     7749       2.68
## 6 Deer Park        7514       2.60
```

Summary

We can summarize the data above with one statement: For the most part, taxpayers owning parcels in Spokane County have local addresses. We can assume most parcel owners are locals based on those addresses. Here are the other high-level points:

1. We analyzed data from Spokane County on 338,304 parcels
2. Almost all Spokane County parcels are owned by USA-addressed taxpayers (99.62%)
3. 88.59% of USA-addressed taxpayers have Washington state addresses. Next is California at 1.45%, and it's downhill from there.
4. Within WA state, most taxpayers who own parcels in Spokane County have addresses in Spokane (60.47%), Spokane Valley (7.05%), or Cheney (3.49%)