Deliverable 3

Carlson Smith

10/15/2020

Introduction

In this project I will be studying climate change, deforestation, fires, and more with their effects on carbon emissions on earth. Using these predictions or data I will try to look to the future as well as find other correlations between the climate and how we live. The reason I'm interested in this field of research is because of the fires and reduction of snow in California along with the fear of how our actions might effect future generations. My parents have talked about how much snow there used to be in Tahoe, and how the fires have just gotten bigger.
I want to know what we can change, and what causes the climate to warm and change over time

The Domains I will specifically be looking into are:

- Fire Emissions
- Fire frequency
- Temperature History
- Carbon Emissions
- Clean Energy
- Electrical Usage

Data Sets

```r
co2.data <- read_csv("owid-co2-data.xlsx.csv")
```

```
## Parsed with column specification:
## cols(
##   iso_code = col_character(),
##   country = col_character(),
##   year = col_double(),
##   co2 = col_double(),
##   co2_growth_prct = col_double(),
##   co2_growth_abs = col_double(),
##   share_global_co2 = col_double(),
##   cumulative_co2 = col_double(),
##   share_global_cumulative_co2 = col_double(),
##   cement_co2 = col_double(),
##   coal_co2 = col_double(),
##   flaring_co2 = col_double(),
##   gas_co2 = col_double(),
##   oil_co2 = col_double(),
##   population = col_double(),
##   gdp = col_double()
## )
```

co2.data comes from https://github.com/owid/co2-data that is maintained by Our World in Data. It is updated regularly and includes data on CO2 emissions along with other helpful metrics. The original data set has a few more continuous variables that I didn't think I would need for what I want to look at.

The categorical variables for co2.data are:

- iso_code - ISO 3166-1 alpha-3 – three-letter country codes
- country - Geographic location
- year - Year of observation

The continuous variables for co2.data are:

- co2 - Annual production-based emissions of carbon dioxide (CO2), measured in million tonnes per year.
- co2_growth_prct - Percentage change in CO2 emissions from one year relative to the previous year.
- co2_growth_abs - Annual change in CO2 emissions from one year relative to the previous year, measured in million tonnes.
- share_global_co2 - National or regional annual CO2 emissions, measured as a percentage of the global total
- cumulative_co2 - Cumulative emissions of CO2 from 1751 through to the given year, measured in million tonnes.
- share_global_cumulative_co2 - National or regional annual cumulative CO2 emissions, measured as a share of the global total
- cement_co2 - CO2 emissions from cement production, measured in million tonnes.
- coal_co2 - CO2 emissions from coal production, measured in million tonnes.
- flaring_co2 - CO2 emissions from gas flaring, measured in million tonnes.
- gas_co2 - CO2 emissions from gas production, measured in million tonnes.
- oil_co2 - CO2 emissions from oil production, measured in million tonnes.
- population - Total population
- gdp - Total real gross domestic product, inflation-adjusted

co2.data comes from Our World in Data and will have some limitations, but should be a very trustworthy source as it does come from a source with a good reputation. You also know that the website doesn't make money off of what they do since it's a .org site. The one downside of this data set is that they don't split up co2 emissions to all the possible sources which means the countries actual co2 release could be different.

```
tree.data <- read_csv("global_country_tree_cover_loss.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   country = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

tree.data comes from https://www.globalforestwatch.org that is made through a partnership with World Resources Institute. It is updated regularly and keeps track of the forests starting from around 2001. Since it's hard to keep track of forests before the initial photos in 2001 were taken, the data is a little more limited to recent years.

The categorical variables for tree.data are:

- country - Geographic location
- threshold - Percent canopy cover levels

The continuous variables for tree.data are:

- area_ha - Hectares of tree cover
- extent_2000_ha - Hectares of tree cover in 2000
- extent_2010_ha - Hectares of tree cover in 2010
- gain_2000-2012_ha - From 2001 to 2012, hectares of tree cover gained
- tc_loss_ha_2001 - Hectares of tree cover loss for 2001
- All years in between show loss for different years
- tc_loss_ha_2019 - Hectares tree cover loss for 2019

tree.data is another organization who monitors the forests of our world with satellite data. Again, they are an .org so the website isn't made to get money and the data they get is going to be quite accurate. Using high resolution satellite imagery to look at our forests is a great way to get a generic summary of how they're doing without too much work.

One worry for this data set is that comparison between 2001-2009 and 2011-2019 is not supposed to be very accurate or show patterns. The authors of the data said to preform this with caution making me think that they are comparing new data to extent 2000 or extent 2010 to find the loss of tree cover. If comparing the two, tree growth and forest regrowth will have to be considered.

```
fire.data <- read_csv("US_Fires.csv")
```

```
## Parsed with column specification:
## cols(
##   Year = col_double(),
##   Fires = col_double(),
##   Acres = col_double()
## )
```

fire.data comes from https://www.nifc.gov/fireInfo/fireInfo_stats_totalFires.html. The data for this data set comes from the National Interagency Fire Center who handles multi agency corrdination for fire fighting in the United States. The data that it provides is a general summary of the US and the fires over years.

The categorical variables for fire.data are:

- Year - The year fire data relates to

The continuous variables for fire.data are:

- Fires - Number of fires
- Acres - Acres burned

fire.data does come from a .org and national agency, but I dont know how reliable it is going to be. I have never heard of this group until I was looking for data so I might run into issues later if I realize this data isn't reliable.

Another worry with this data is there is a chance deforestation is the US and the acres burned are connected, but also who knows, more research is going to be needed to figure out if they track the similar areas.

```r
car.data <- read_csv("Annual_Emissions_per_Vehicle.csv")
```

```
## Parsed with column specification:
## cols(
##   car_type = col_character(),
##   pounds_of_c02 = col_double(),
##   miles_driven = col_double()
## )
```

```r
electric.data <- read_csv("Electricity_Sources.csv")
```

```
## Parsed with column specification:
## cols(
##   electric_sources = col_character(),
##   percentage_grid = col_double()
## )
```

car.data and electric.data is grabbed from https://afdc.energy.gov/vehicles/electric_emissions.html that breaks down the data into a pie chart and bar chart. This data source is to fill in some blanks of data in our previous data sets so we can better explore our questions

The categorical variables for car.data are:

- car_type - Style of engine for cars

The continuous variables for car.data are:

- pounds_of_c02 - pounds of co2 released by car over a year
- miles_driven - miles driven by car over a year

The categorical variables for electric.data are:

- electric_sources - Electric source for the grid

The continuous variables for electric.data are:

- percentage_grid - percentage of the grid supported by type of source

These two sets of data are from the United States Department of Energy who also listed where they got their estimations and data. The sources they used and the results of the summarization all come from quite trustworthy sources. Though the one downside of this data set is that it is only for the United States and not global.

Data Prep and Analysis

co2.data has data that covers generic land regions that overlap with countries.
This is just added up from each country in it so I figured by deleting the data that didn't have an iso_code, we could remove some of this overlap. I'm also going to delete any data that doesn't have co2 output since that's what we really
are about when looking at this data

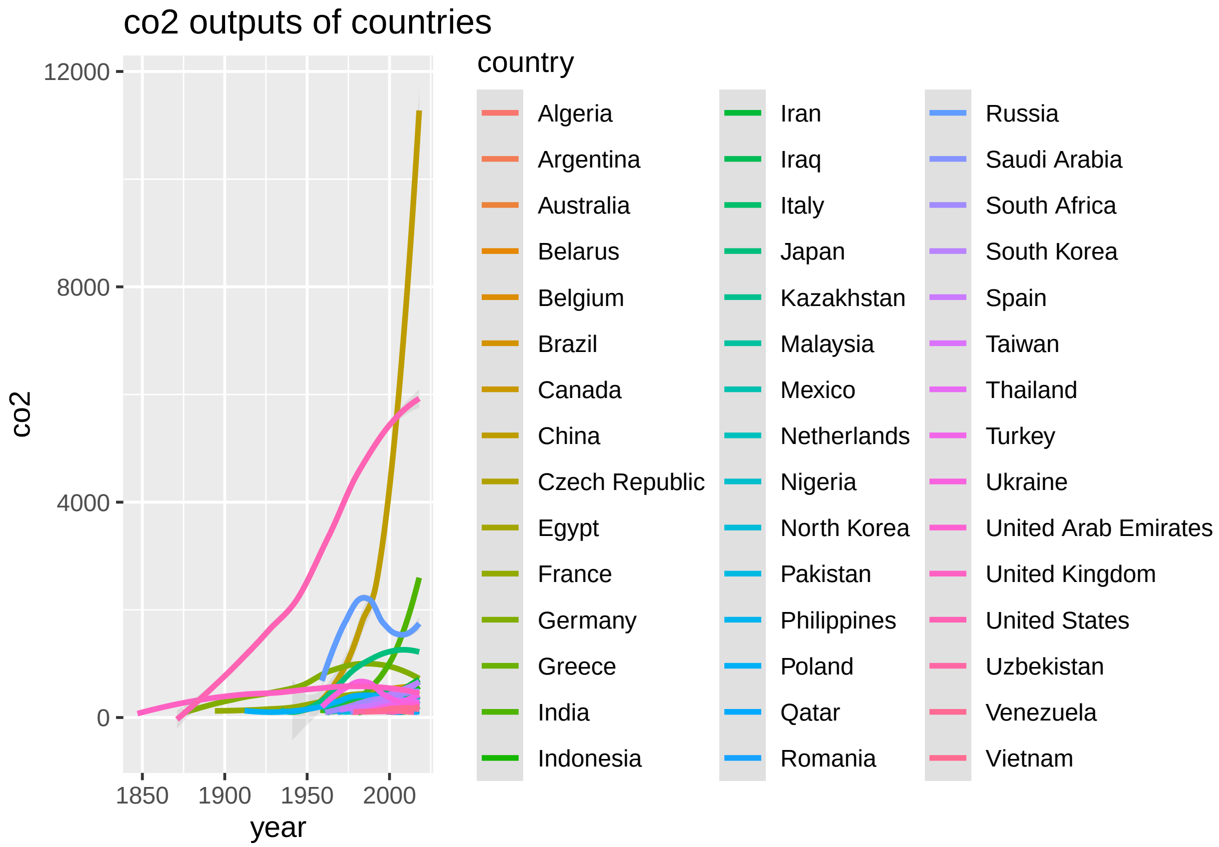
```
co2.data <- filter(co2.data, iso_code != is.na(iso_code), co2 > 0)
head(co2.data)
```

```
## # A tibble: 6 x 16
##   iso_code country  year   co2 co2_growth_prct co2_growth_abs share_global_co2
##   <chr>    <chr>   <dbl> <dbl>           <dbl>          <dbl>            <dbl>
## 1 AFG      Afghan~  1949 0.015            NA             NA                0
## 2 AFG      Afghan~  1950 0.084           475              0.07             0.001
## 3 AFG      Afghan~  1951 0.092             8.70           0.007            0.001
## 4 AFG      Afghan~  1952 0.092             0              0                0.001
## 5 AFG      Afghan~  1953 0.106            16              0.015            0.002
## 6 AFG      Afghan~  1954 0.106             0              0                0.002
## # ... with 9 more variables: cumulative_co2 <dbl>,
## #   share_global_cumulative_co2 <dbl>, cement_co2 <dbl>, coal_co2 <dbl>,
## #   flaring_co2 <dbl>, gas_co2 <dbl>, oil_co2 <dbl>, population <dbl>,
## #   gdp <dbl>
```

This graph shows the carbon emissions for countries based on the year, though the first time I tried this, it didn't work since there were too many countries with very low carbon output to draw lines for each one. For ease of visualization, I'm going to set the co2 to above 100 million tonnes per year.

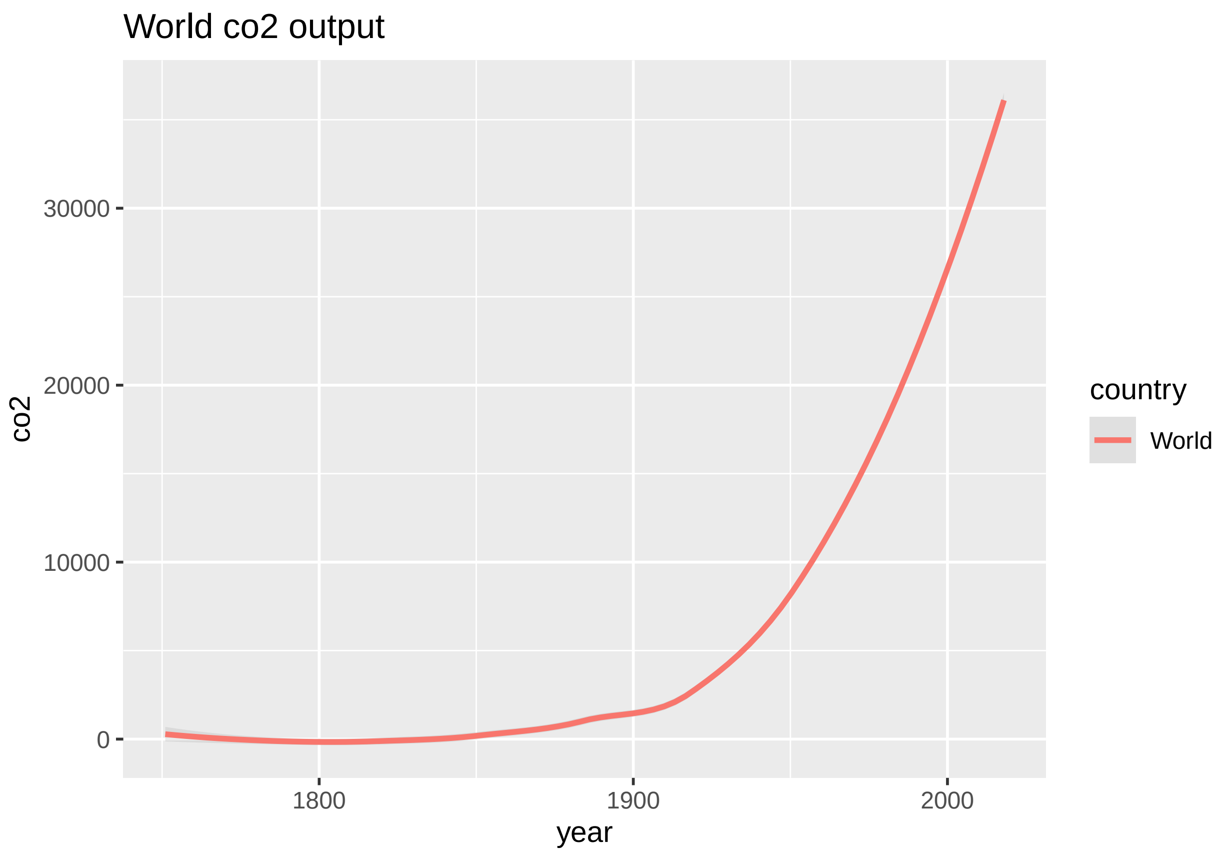
```
ggplot(data = filter(co2.data, co2 > 100, country != "World"), mapping = aes(x=year, y=co2, color = cour
  geom_smooth(alpha = 0.2)+
  ggtitle("co2 outputs of countries")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



```r
ggplot(data = filter(co2.data, country == "World"), mapping = aes(x=year, y=co2, color = country))+
  geom_smooth(alpha = 0.2)+
  ggtitle("World co2 output")
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
```



You can see from this data set, the overall growth of co2 emissions and outputs with the World summary line. Admittedly there are too many countries to see all of this added up on the graph, but you can see the general idea with the visible lines. There are also lots more in the data set that are not put in the graph since we wouldn't be able to graph each line, these are just the biggest emitters. Though you can see each line with their growth over the years, some moving faster than others.

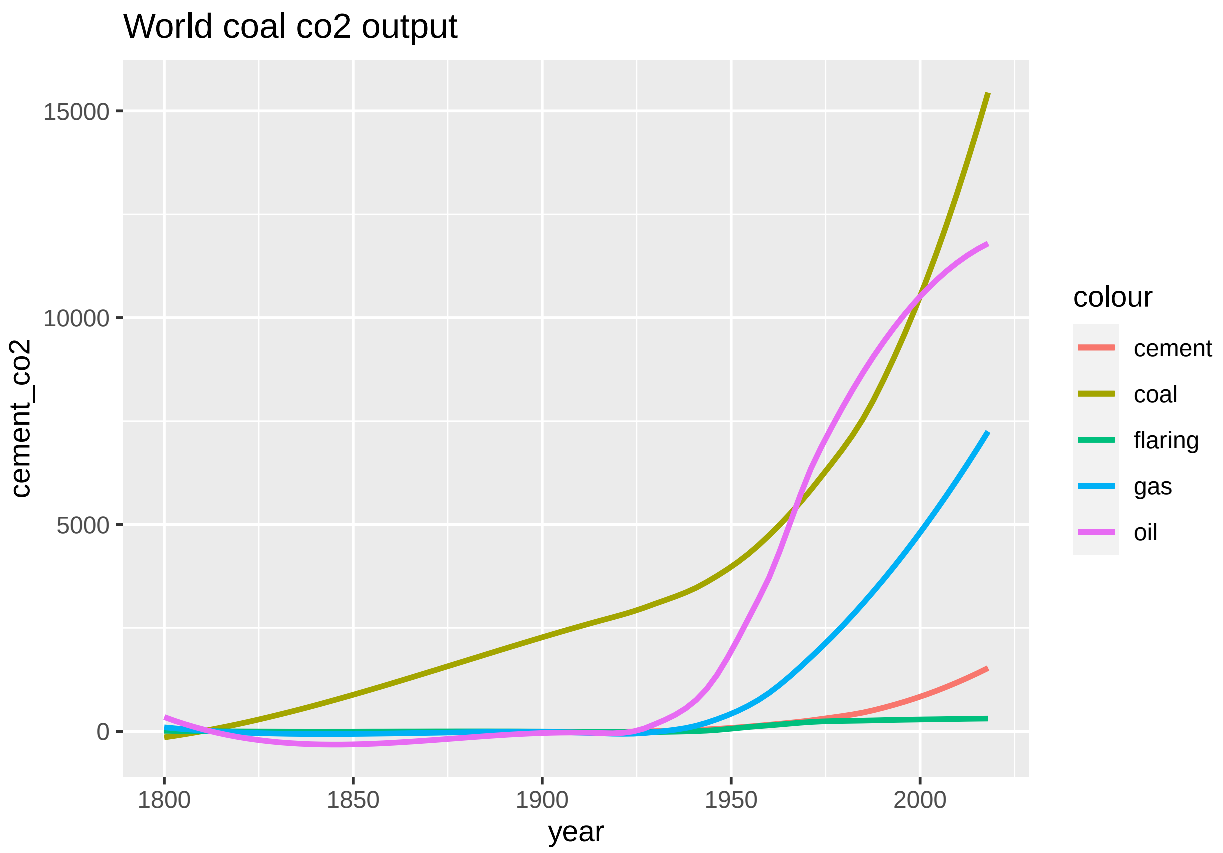
```
summary(co2.data)
```

```
##    iso_code           country               year           co2          
##  Length:18439       Length:18439       Min.   :1751   Min.   :    0.00  
##  Class :character   Class :character   1st Qu.:1950   1st Qu.:    0.52  
##  Mode  :character   Mode  :character   Median :1975   Median :    4.39  
##                                        Mean   :1965   Mean   :  171.67  
##                                        3rd Qu.:1997   3rd Qu.:   33.09  
##                                        Max.   :2018   Max.   :36572.75  
##                                                                         
##  co2_growth_prct     co2_growth_abs     share_global_co2  cumulative_co2     
##  Min.   :-2835.714   Min.   :-847.729   Min.   :  0.000   Min.   :     -1.2  
##  1st Qu.:   -1.097   1st Qu.:  -0.015   1st Qu.:  0.004   1st Qu.:      6.6  
##  Median :    3.578   Median :   0.051   Median :  0.045   Median :     77.9  
##  Mean   :   18.359   Mean   :   3.864   Mean   :  2.864   Mean   :   6259.4  
##  3rd Qu.:   10.680   3rd Qu.:   0.869   3rd Qu.:  0.322   3rd Qu.:    770.6  
##  Max.   :27336.247   Max.   :1543.508   Max.   :100.000   Max.   :1611817.1  
##  NA's   :229         NA's   :214                                            
##  share_global_cumulative_co2   cement_co2          coal_co2       
##  Min.   :0.0000000           Min.   :   0.000   Min.   :    0.000  
##  1st Qu.:0.0000000           1st Qu.:   0.000   1st Qu.:    0.110  
##  Median :0.0000000           Median :   0.179   Median :    2.176  
##  Mean   :0.0002734           Mean   :   5.591   Mean   :  110.922  
##  3rd Qu.:0.0000000           3rd Qu.:   1.147   3rd Qu.:   16.693  
##  Max.   :0.0100000           Max.   :1506.762   Max.   :15043.240  
##                              NA’s   :4215       NA’s   :5362       
##   flaring_co2        gas_co2           oil_co2            population       
##  Min.   :  0.000   Min.   :  -0.007   Min.   :   -1.748   Min.   :2.000e+03  
##  1st Qu.:  0.000   1st Qu.:   0.000   1st Qu.:    0.136   1st Qu.:1.308e+06  
##  Median :  0.000   Median :   0.059   Median :    1.432   Median :5.198e+06  
##  Mean   :  3.222   Mean   :  38.168   Mean   :   60.063   Mean   :4.858e+07  
##  3rd Qu.:  0.275   3rd Qu.:   5.888   3rd Qu.:   10.681   3rd Qu.:1.612e+07  
##  Max.   :402.003   Max.   :7485.188   Max.   :12425.536   Max.   :7.631e+09  
##  NA’s   :8779      NA’s   :6582       NA’s   :556         NA’s   :350        
##       gdp            
##  Min.   :6.378e+07  
##  1st Qu.:9.080e+09  
##  Median :3.022e+10  
##  Mean   :4.434e+11  
##  3rd Qu.:1.230e+11  
##  Max.   :1.066e+14  
##  NA’s   :5548       
```

You can also see from this summary of co2.data that the mean of co2_growth_prct is on an upward trend even though the min occurrence is negative. Same thing with the co2_growth_abs.

```r
ggplot()+
  geom_smooth(data = filter(co2.data, country == "World"), se = FALSE, mapping = aes(x=year, y=cement_c
  geom_smooth(data = filter(co2.data, country == "World"), se = FALSE, mapping = aes(x=year, y=coal_co2
  geom_smooth(data = filter(co2.data, country == "World"), se = FALSE, mapping = aes(x=year, y=flaring_
  geom_smooth(data = filter(co2.data, country == "World"), se = FALSE, mapping = aes(x=year, y=gas_co2,
  geom_smooth(data = filter(co2.data, country == "World"), se = FALSE, mapping = aes(x=year, y=oil_co2,
  ggtitle("World coal co2 output")
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
```



This graph splits up all the sources of co2 in the world to really show who is doing the most and what really needs to be fixed if we want to reduce our emissions. These could all be things fixed in the future to reverse the damage we have done so far.

```
head(tree.data)
```

```
## # A tibble: 6 x 25
##   country threshold area_ha extent_2000_ha extent_2010_ha ‘gain_2000-2012~
##   <chr>       <dbl>   <dbl>          <dbl>          <dbl>            <dbl>
## 1 Afghan~        10  6.44e7         432115         126247              304
## 2 Afghan~        15  6.44e7         302660         106867              304
## 3 Afghan~        20  6.44e7         284357         105733              304
## 4 Afghan~        25  6.44e7         254867          72395              304
## 5 Afghan~        30  6.44e7         205791          71797              304
## 6 Afghan~        50  6.44e7         148430          46242              304
## # ... with 19 more variables: tc_loss_ha_2001 <dbl>, tc_loss_ha_2002 <dbl>,
## #   tc_loss_ha_2003 <dbl>, tc_loss_ha_2004 <dbl>, tc_loss_ha_2005 <dbl>,
## #   tc_loss_ha_2006 <dbl>, tc_loss_ha_2007 <dbl>, tc_loss_ha_2008 <dbl>,
## #   tc_loss_ha_2009 <dbl>, tc_loss_ha_2010 <dbl>, tc_loss_ha_2011 <dbl>,
## #   tc_loss_ha_2012 <dbl>, tc_loss_ha_2013 <dbl>, tc_loss_ha_2014 <dbl>,
## #   tc_loss_ha_2015 <dbl>, tc_loss_ha_2016 <dbl>, tc_loss_ha_2017 <dbl>,
## #   tc_loss_ha_2018 <dbl>, tc_loss_ha_2019 <dbl>
```

tree.date looks at all of the tree cover, and to make it a little easier to find patterns in the data, I’m going to cut down the canopy density percentage to 30% and up. The reason for cutting at 30% is that its the better visualization of tree loss, you can’t choose 100%, but you cant choose 10%, as acres lost at that point would do very little to the number of trees. Along with that, I'm going to rotate the data to make year a new variable so I can easily graph tree loss over the years as well as join co2 and trees later on.

```
tree.30threshold <- filter(tree.data, threshold == 30)
names(tree.30threshold)[7] <- "2001"
names(tree.30threshold)[8] <- "2002"
names(tree.30threshold)[9] <- "2003"
names(tree.30threshold)[10] <- "2004"
names(tree.30threshold)[11] <- "2005"
names(tree.30threshold)[12] <- "2006"
names(tree.30threshold)[13] <- "2007"
names(tree.30threshold)[14] <- "2008"
names(tree.30threshold)[15] <- "2009"
names(tree.30threshold)[16] <- "2010"
names(tree.30threshold)[17] <- "2011"
names(tree.30threshold)[18] <- "2012"
names(tree.30threshold)[19] <- "2013"
names(tree.30threshold)[20] <- "2014"
names(tree.30threshold)[21] <- "2015"
names(tree.30threshold)[22] <- "2016"
names(tree.30threshold)[23] <- "2017"
names(tree.30threshold)[24] <- "2018"
names(tree.30threshold)[25] <- "2019"
tree.pivot <- pivot_longer(tree.30threshold,c('2001','2002','2003','2004','2005','2006','2007','2008','
head(tree.pivot)
```

```
## # A tibble: 6 x 8
##   country threshold area_ha extent_2000_ha extent_2010_ha `gain_2000-2012~ year
##   <chr>       <dbl>   <dbl>          <dbl>          <dbl>            <dbl> <chr>
## 1 Afghan~        30  6.44e7         205791          71797              304 2001
## 2 Afghan~        30  6.44e7         205791          71797              304 2002
## 3 Afghan~        30  6.44e7         205791          71797              304 2003
## 4 Afghan~        30  6.44e7         205791          71797              304 2004
## 5 Afghan~        30  6.44e7         205791          71797              304 2005
## 6 Afghan~        30  6.44e7         205791          71797              304 2006
## # ... with 1 more variable: tc_loss_ha <dbl>
```

Using the same method I did with co2 emissions for each country, you can see the deforestation for multiple countries here. I chose the highest ones since I have too many countries to graph them all. I would have done geom_smooth but the unpredictable data made it so geom smooth didn't work, this might be an issue in the future.

```
summarize(tree.pivot, mean(tc_loss_ha))
```

```
## # A tibble: 1 x 1
##   `mean(tc_loss_ha)`
##                <dbl>
## 1            126616.
```

Though I couldn't make a graph you can see the mean tc_loss_ha over all the years. I'm going to have to the size of this data and look at singular countries or make the data easier to look at some point. Maybe I could come up with a sum total to look at the world deforestation?

Using this pivot, we can combine with the co2 data with the years and countries that appear in both data sets. This gives us a generic and overarching data set with deforestation rates and co2 emissions. I'm going to filter as well since if I don't want to join and have tree data being empty, I want the related years only, which is limited my tree.pivot which goes 2001-2019.

```
tree.pivot$year <- as.double(tree.pivot$year)
tree.co2.data <- left_join(co2.data, tree.pivot) %>%
  filter(year > 2000)
```

```
## Joining, by = c("country", "year")
```

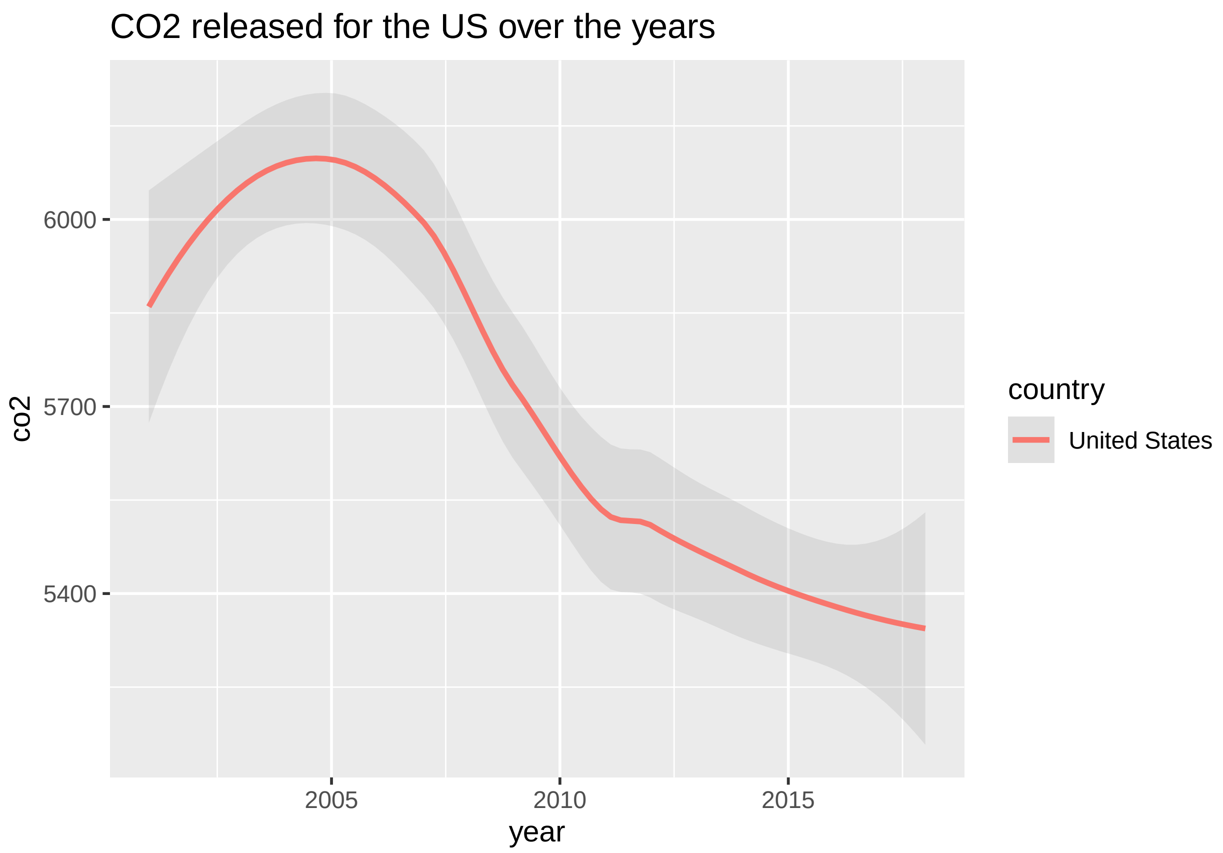
```
head(tree.co2.data)
```

```
## # A tibble: 6 x 22
##   iso_code country  year   co2 co2_growth_prct co2_growth_abs share_global_co2
##   <chr>    <chr>   <dbl> <dbl>           <dbl>          <dbl>            <dbl>
## 1 AFG      Afghan~  2001 0.812            5.71          0.044            0.003
## 2 AFG      Afghan~  2002 1.06            31.0           0.252            0.004
## 3 AFG      Afghan~  2003 1.20            13.3           0.141            0.004
## 4 AFG      Afghan~  2004 0.908          -24.6          -0.297            0.003
## 5 AFG      Afghan~  2005 1.32            45.3           0.412            0.005
## 6 AFG      Afghan~  2006 1.64            24.5           0.323            0.005
## # ... with 15 more variables: cumulative_co2 <dbl>,
## #   share_global_cumulative_co2 <dbl>, cement_co2 <dbl>, coal_co2 <dbl>,
## #   flaring_co2 <dbl>, gas_co2 <dbl>, oil_co2 <dbl>, population <dbl>,
## #   gdp <dbl>, threshold <dbl>, area_ha <dbl>, extent_2000_ha <dbl>,
## #   extent_2010_ha <dbl>, `gain_2000-2012_ha` <dbl>, tc_loss_ha <dbl>
```

To take a peek for relations in the tree/co2 data set, I'm going to reduce it to just the USA where I can look at both the growth rates of deforestation and how much co2 we make

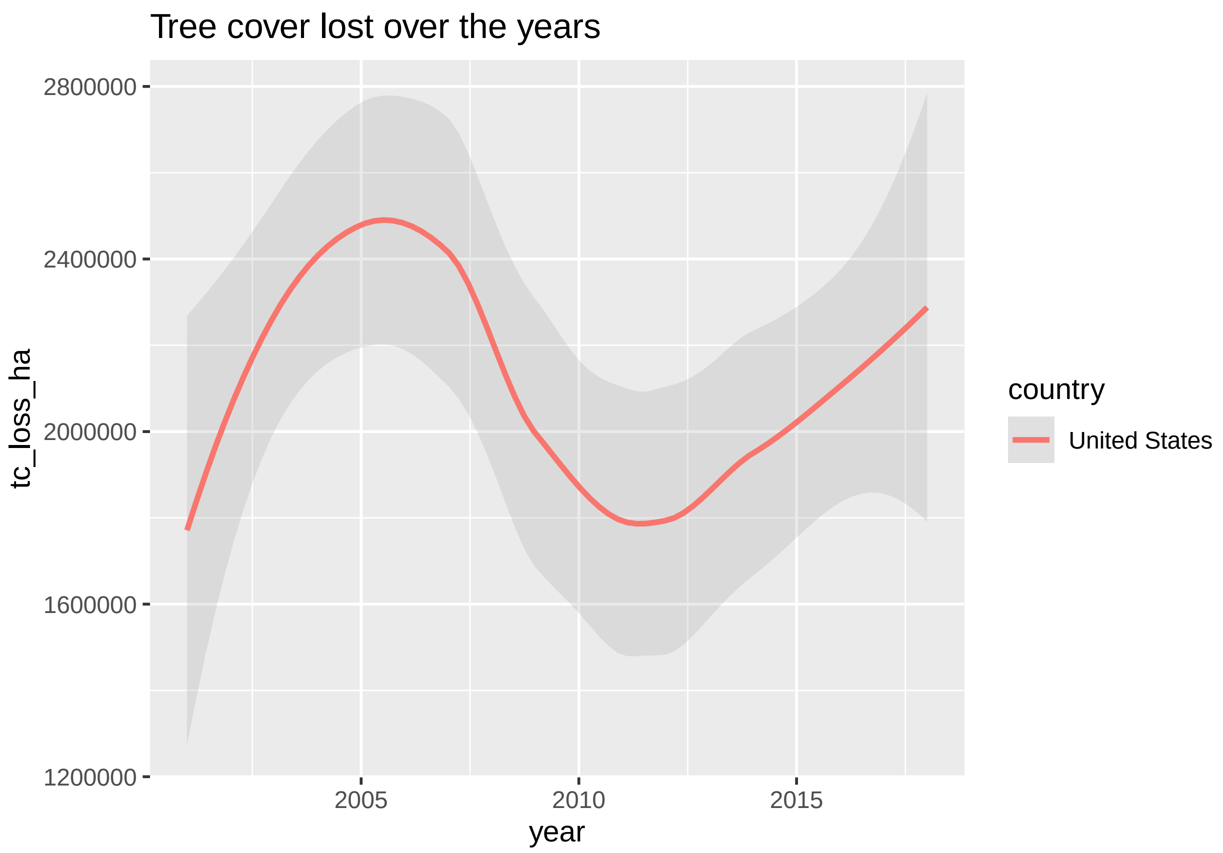
```
ggplot(data = filter(tree.co2.data, iso_code == 'USA'), mapping = aes(x=year, y=co2, color = country))+
  geom_smooth(alpha = 0.2)+
  ggtitle("CO2 released for the US over the years")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



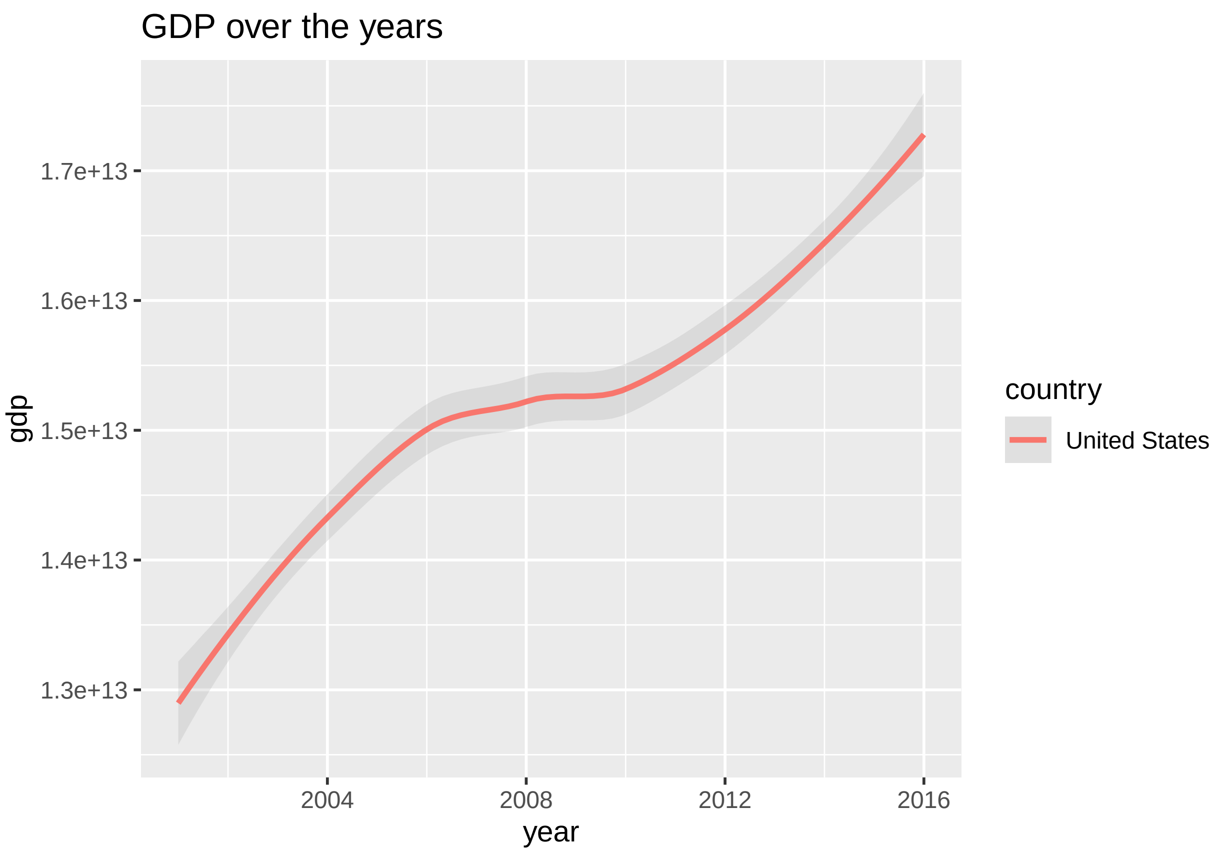
```
ggplot(data = filter(tree.co2.data, iso_code == 'USA'), mapping = aes(x=year, y=tc_loss_ha, color = cou
  geom_smooth(alpha = 0.2)+
  ggtitle("Tree cover lost over the years")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



```
ggplot(data = filter(tree.co2.data, iso_code == 'USA'), mapping = aes(x=year, y=gdp, color = country))+
  geom_smooth(alpha = 0.2)+
  ggtitle("GDP over the years")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



while it does look like deforestation and co2 output seem to have very similar curves on this graph pair at first, it could be explored more to see if this is true with other countries as well. If this is true then I would have to look into the idea of co2 output and hectares lost being connected on the co2 charts.

```
head(fire.data)
```

```
## # A tibble: 6 x 3
##    Year Fires    Acres
##   <dbl> <dbl>    <dbl>
## 1  2019 50477  4664364
## 2  2018 58083  8767492
## 3  2017 71499 10026086
## 4  2016 67743  5509995
## 5  2015 68151 10125149
## 6  2014 63312  3595613
```

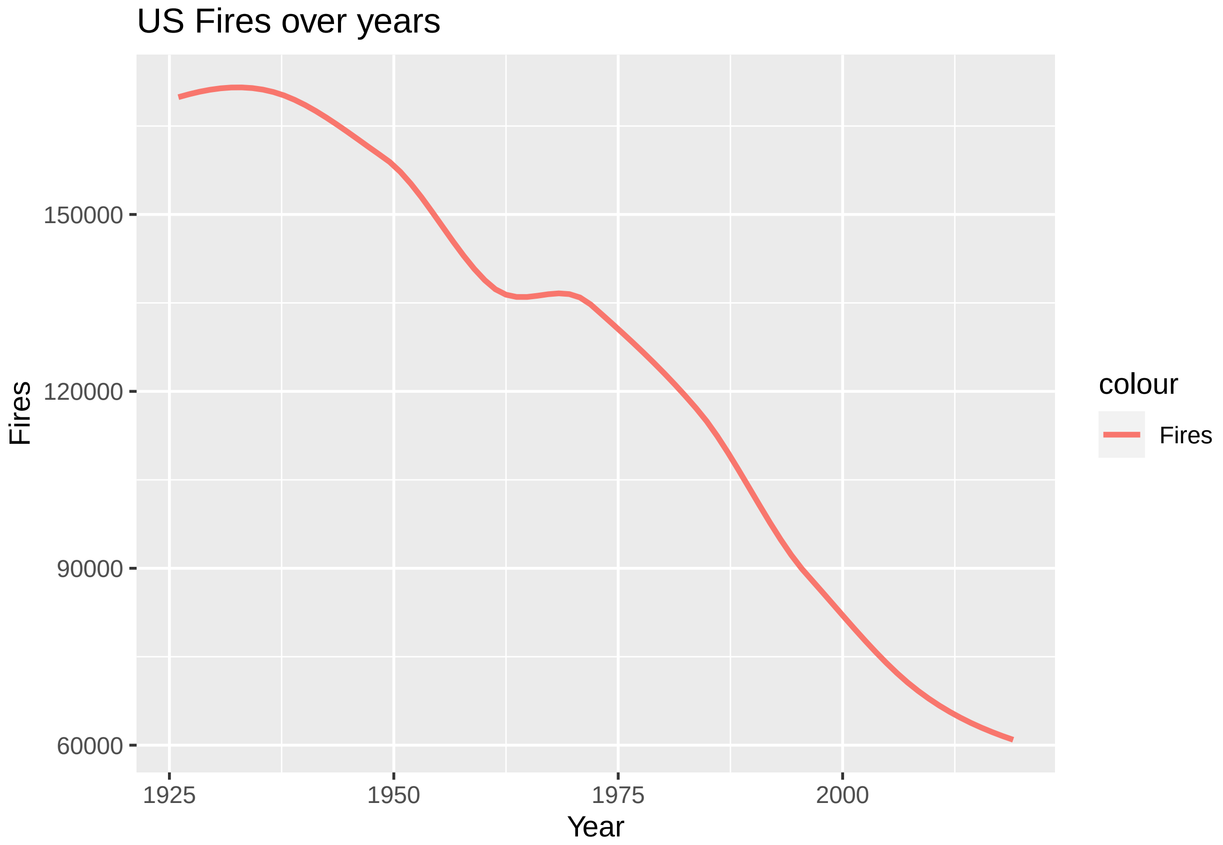
```
summary(fire.data)
```

```
##       Year          Fires             Acres
##  Min.   :1926   Min.   : 18229   Min.   : 1148409
##  1st Qu.:1949   1st Qu.: 78839   1st Qu.: 3413181
##  Median :1972   Median :115852   Median : 5568044
##  Mean   :1972   Mean   :124500   Mean   :12062963
##  3rd Qu.:1996   3rd Qu.:170808   3rd Qu.:15488500
##  Max.   :2019   Max.   :249370   Max.   :52266000
```

The Fire data for the US is mainly to see if there is a connection with the co2 emissions and rates of fire as well. There is a chance that deforestation reduces fires as well which could be something explored in the future as well. This could be explored further if I look into the Australian data with their recent mega fire.

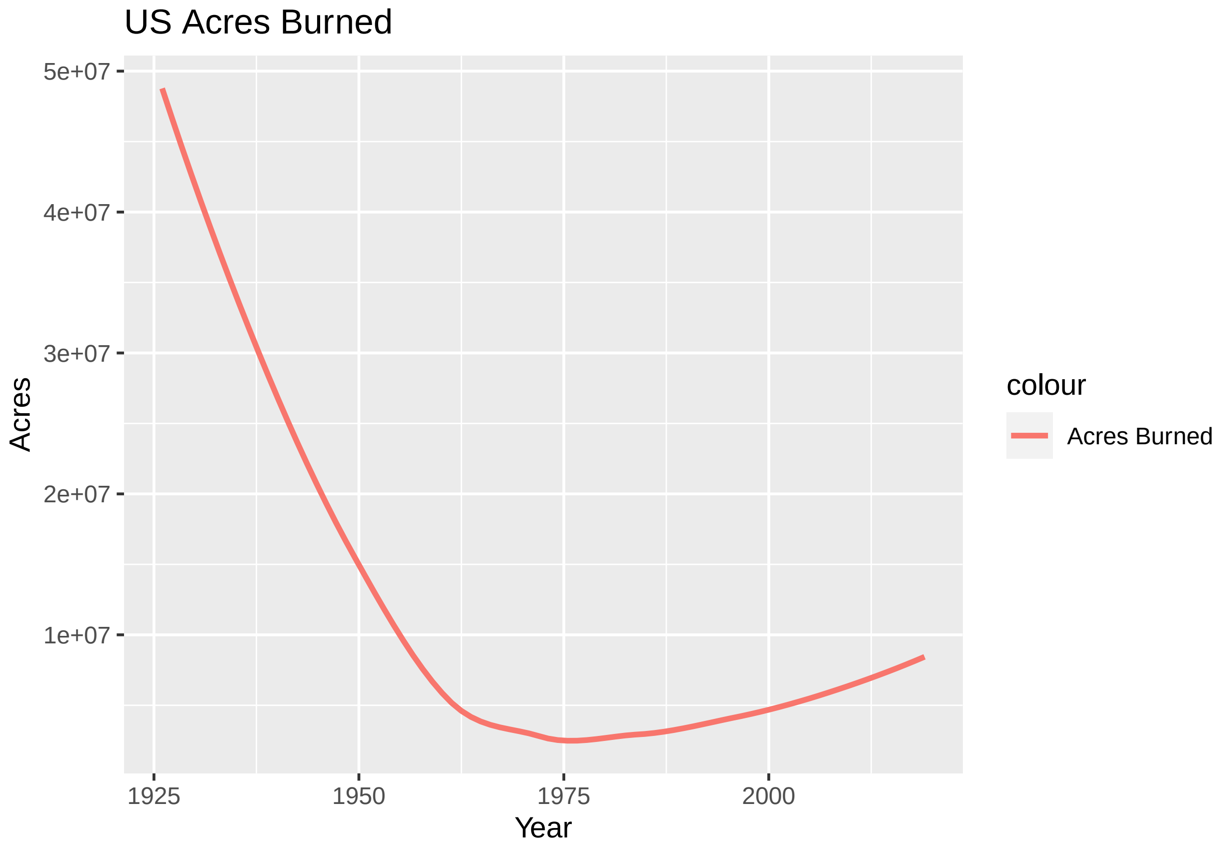
```
ggplot()+
  geom_smooth(data = fire.data, se=FALSE, mapping = aes(x=Year, y=Fires, color = 'Fires'))+
  ggtitle("US Fires over years")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



```
ggplot()+
  geom_smooth(data = fire.data, se=FALSE, mapping = aes(x=Year, y=Acres, color = 'Acres Burned'))+
  ggtitle("US Acres Burned")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Though this data is only the United States, it is interesting seeing the dip end in 1975, and slowly go back up while the number of fires still goes down, it seems like the fires are just getting bigger in more recent years, but we have less fires starting overall.

```
head(car.data)
```

```
## # A tibble: 4 x 3
##   car_type       pounds_of_c02 miles_driven
##   <chr>                  <dbl>        <dbl>
## 1 All Electric            4100        11824
## 2 Plug-in Hybrid          5885        11824
## 3 Hybrid                  6258        11824
## 4 Gasoline               11435        11824
```

```
head(electric.data)
```

```
## # A tibble: 6 x 2
##   electric_sources percentage_grid
##   <chr>                      <dbl>
## 1 Natrual Gas                38.5
## 2 Coal                       23.5
## 3 Nuclear                    19.7
## 4 Wind                        7.31
## 5 Hydro                       6.54
## 6 Solar                       1.76
```

These last two data sets are just to fill in later when we look at renewable energy and what would happen if we managed to switch to a green house gas free life style which would include changing the power grid and switching to electric cars. This data set needs to be worked on with more tables if I pursue this path further.

Need to look for data of electric energy use on the power grid to see if that might have a relation to any of the co2 growth charts in a country. Electric cars might also have an impact since they get gas cars off the roads, but data for that topic is quite hard to find.

Data Science Questions

I really want to look into GPD and co2 output, i think that they don't have to be connected, but could also be in close relation when your economy is based off of machinery that isn't the cleanest. For example a country that has a higher oil co2 release might have a higher GPD and be having more deforestation as well. This would all point to a mindset though rather than an actual mathematical pattern that you can follow.

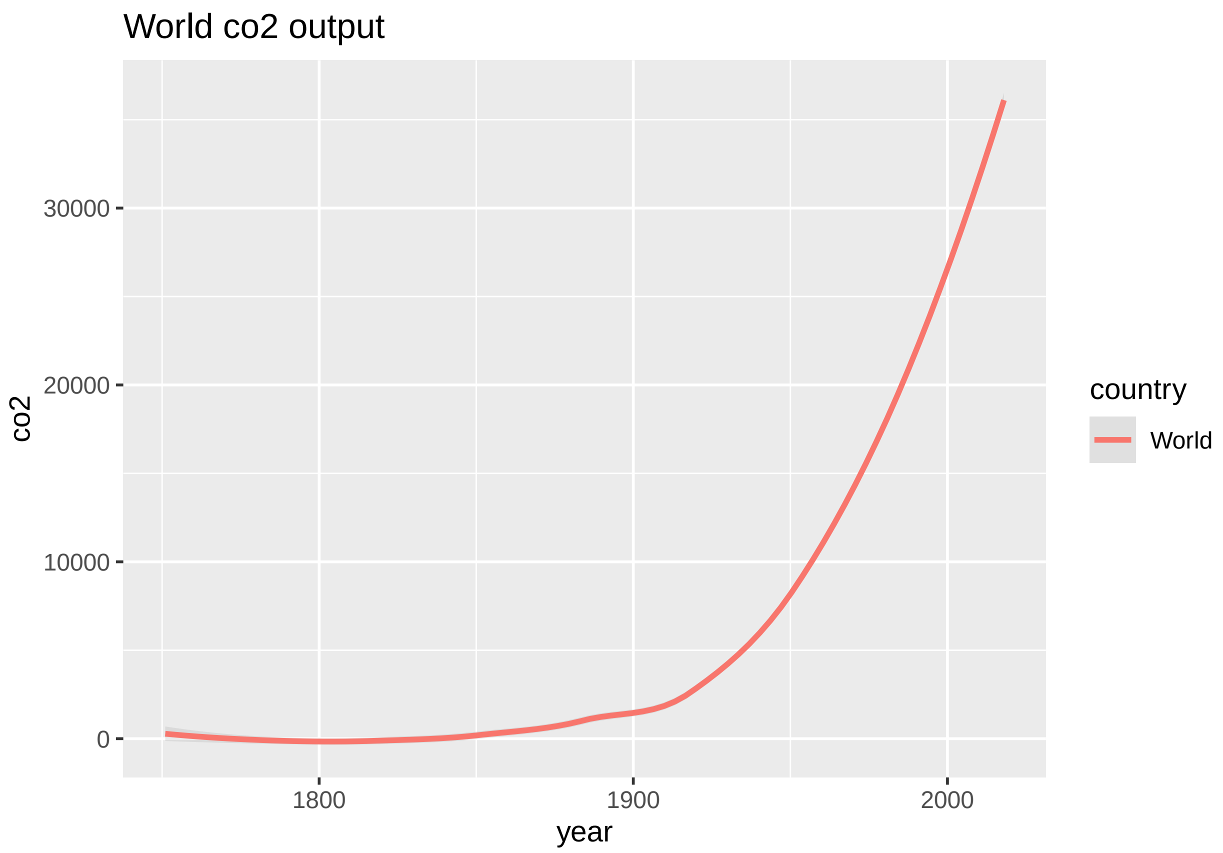
Power grids in a country might also have an affect on this area as some countries make their way towards a 0 carbon life style. This might now actually push much around though as construction and vehicles are also another big factor in this area.

A success for me is finding a relation between co2 and gdp that a linear model can follow. I feel like a R2 value above 0.90 is going to sufficient enough to say that there was a pattern that was found between the two. Though better values are always good. If I can I will improve this model with energy or forest data later on.

Possible Model Topics.

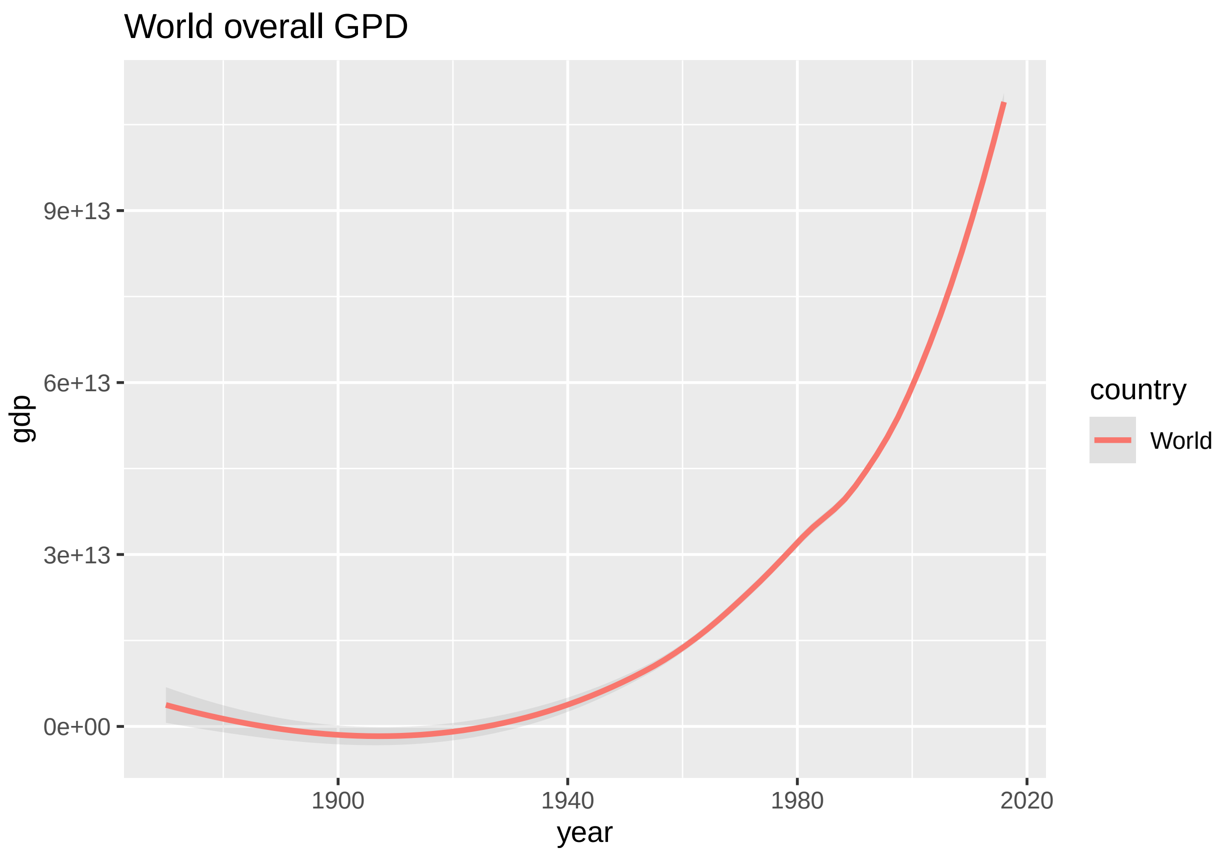
```r
ggplot(data = filter(co2.data, country == "World"), mapping = aes(x=year, y=co2, color = country))+
  geom_smooth(alpha = 0.2)+
  ggtitle("World co2 output")
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
```



```
ggplot(data = filter(co2.data, country == "World"), mapping = aes(x=year, y=gdp, color = country))+
  geom_smooth(alpha = 0.2)+
  ggtitle("World overall GPD")
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
```

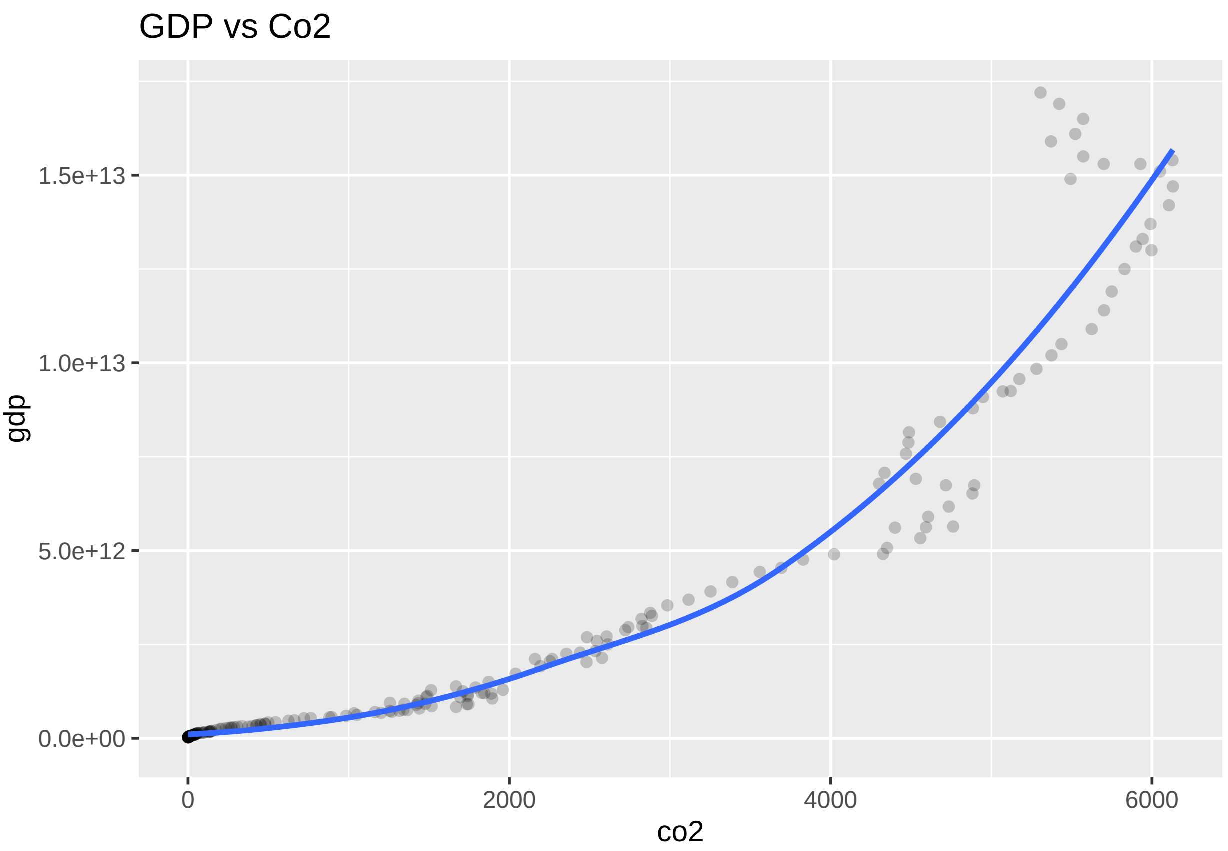


From the data science questions I wanted to look into from the initial data analysis, it seems like gdp and co2 might have a very close relation based on the general look of the world. THough I do want to look into specific countries since I know the USA has curbed their co2 outputs but their gdp is still rising.

THe model would be a simple relation between gdp and the co2 outputs

```
ggplot(data = filter(co2.data, country == "United States"), mapping = aes(x=co2, y=gdp))+
  geom_point(alpha = 0.2)+
  geom_smooth(alpha = 0)+
  ggtitle("GDP vs Co2")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



While its quite hard to see much in this point graph, looking into individual co2 emissions might provide better results.

To make a model off of gdp to attempt finding a prediction on co2 output, I'm first going to split the data into an 80% 20% split for training and testing.

Setting a seed is also going to help streamline the process in this case it makes sure the random data split is the same each time. In the data split though, I need to get rid of NA values that I will be making the model off of.

For validation and exploration, I'm also going to include

```
set.seed(12345)
co2.data <- filter(co2.data, gdp != is.na(gdp), co2 != is.na(co2))
train_rows <- as.vector(createDataPartition(co2.data$co2, p = 0.8, list = FALSE))
tv_data <- co2.data[train_rows, ]
test <- co2.data[-train_rows, ]

train_rows <- as.vector(createDataPartition(tv_data$co2, p = 0.75, list = FALSE))
valid <- tv_data[-train_rows, ]
train <- tv_data[train_rows, ]
```

Since I only have two data sets with no validation split, I will be using the k-fold cross validation method to make a linear model on the training set.

```
train.control <- trainControl(method = "cv", number = 5)
model <- train(co2 ~ gdp, data = tv_data, method = "lm", trControl = train.control)
```

```r
predictions <- add_predictions(valid, model)
R2(predictions$pred, predictions$co2)
```

```
## [1] 0.9498204
```

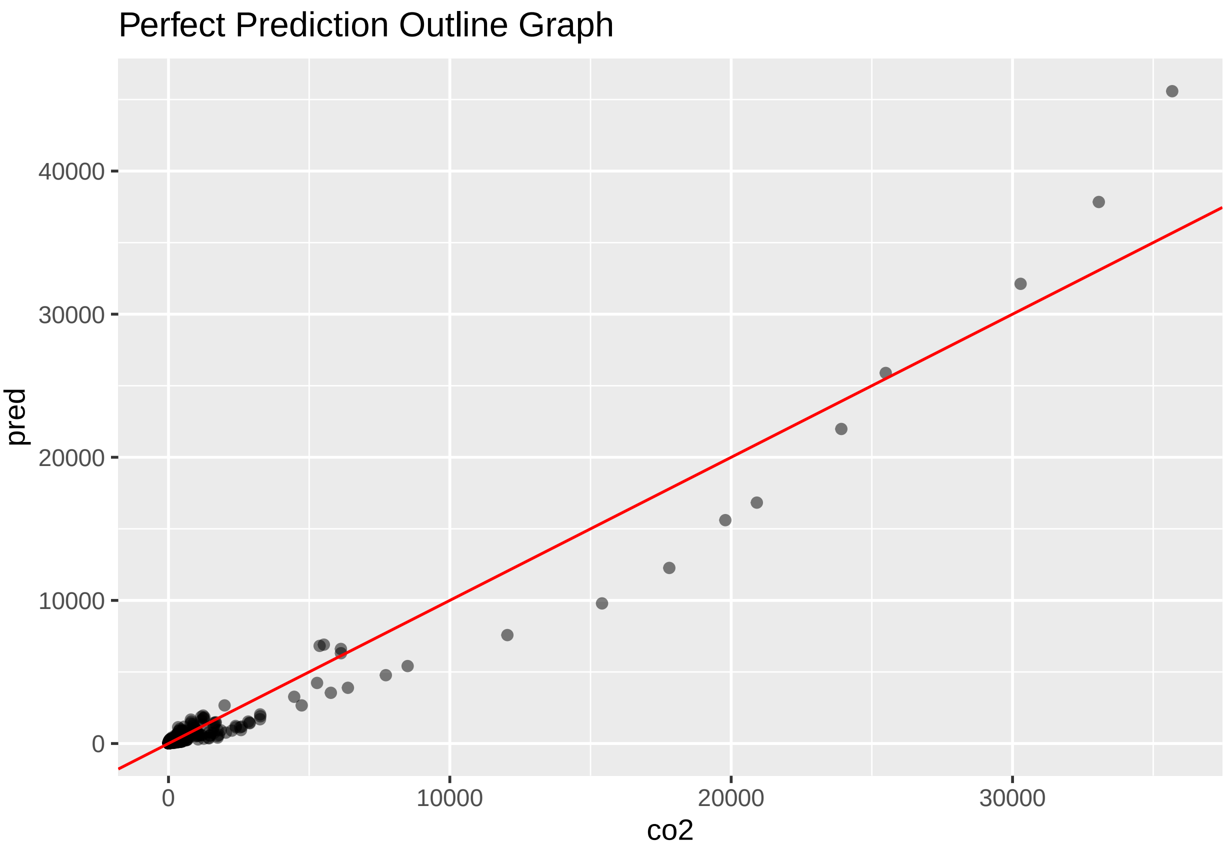
```r
MAE(predictions$pred, predictions$co2)
```

```
## [1] 84.05828
```

```r
RMSE(predictions$pred, predictions$co2)
```

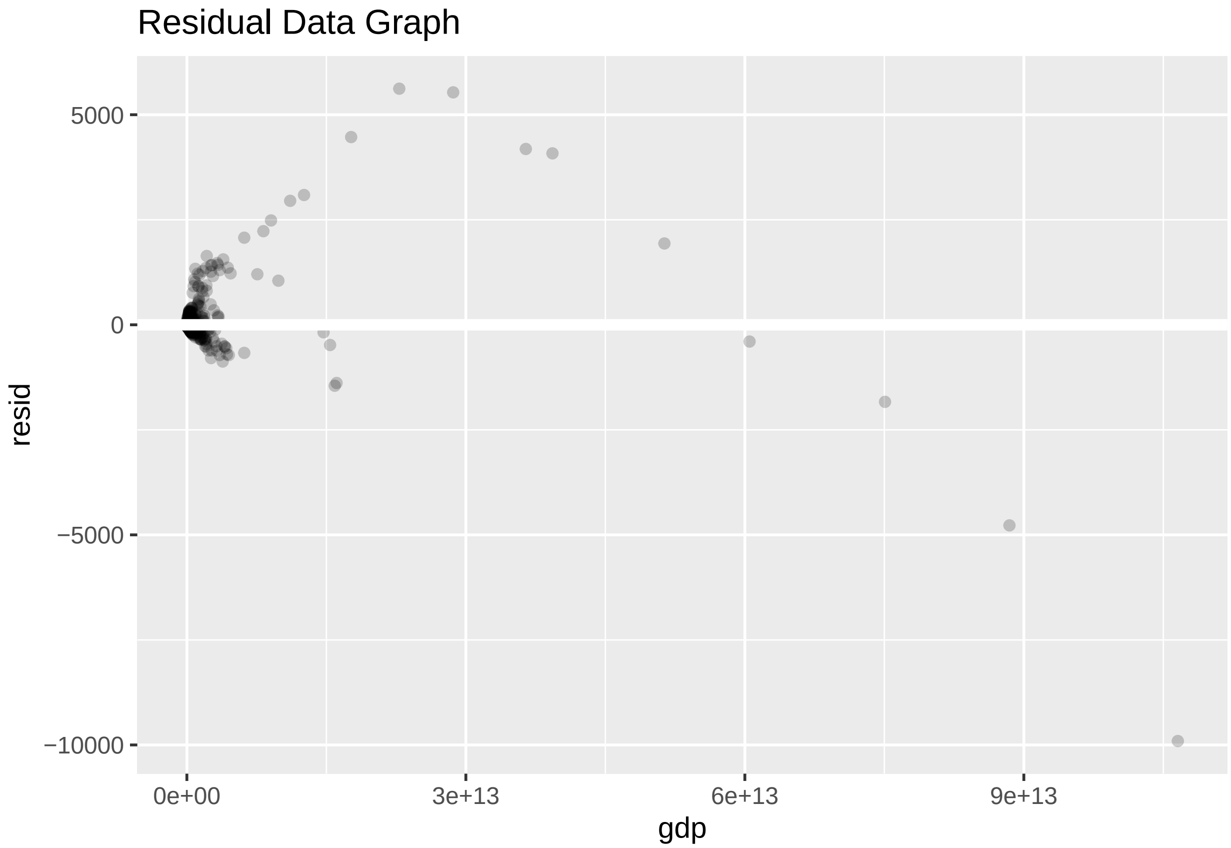
```
## [1] 365.1683
```

```r
ggplot(data = predictions, mapping=(aes(x = co2, y = pred)))+
  geom_point(alpha = 0.5)+
  geom_abline(intercept = 0, slope = 1, color = "red")+
  ggtitle("Perfect Prediction Outline Graph")
```

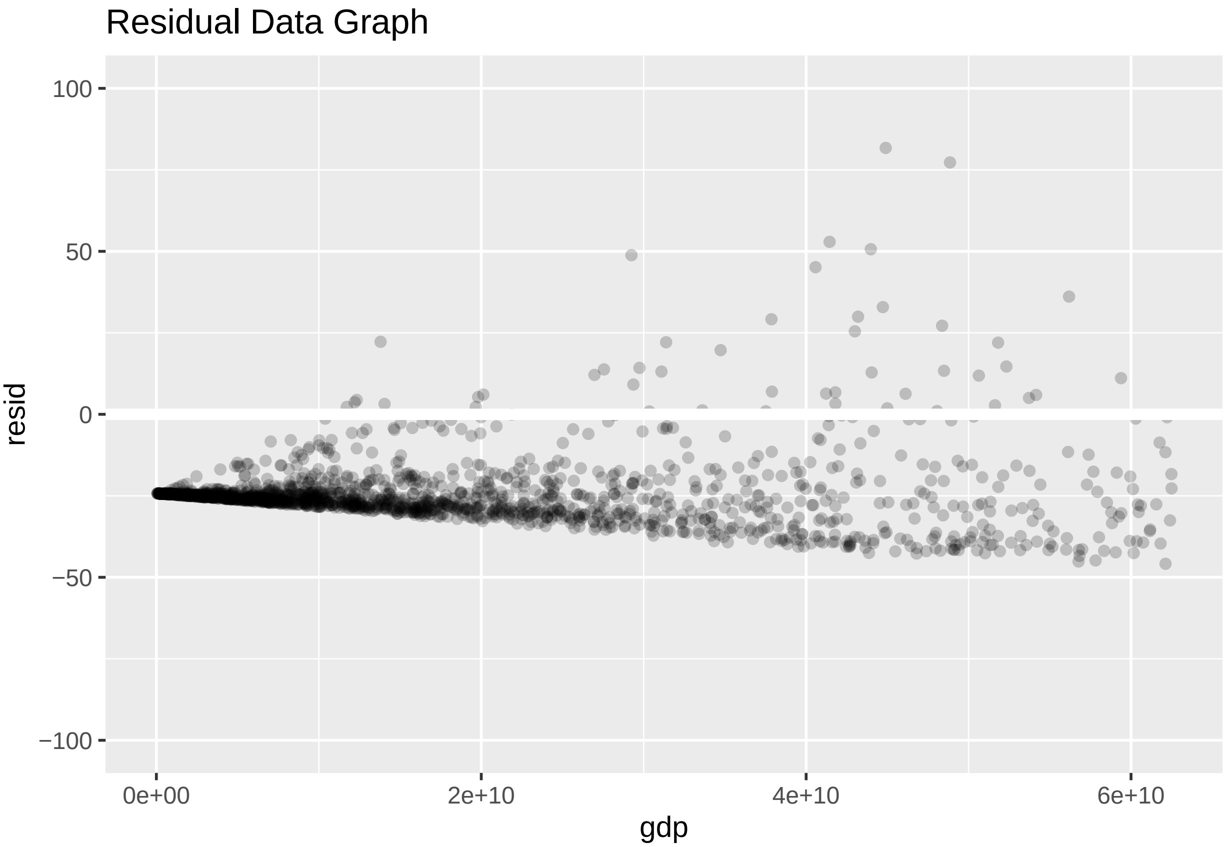


Looking at these data pieces it looks like our linear model found some pattern it could make into an eq as the R2 value isn't too far from 1. But looking at the prediction graph, it seems like a linear model might not be able to capture the exact values.

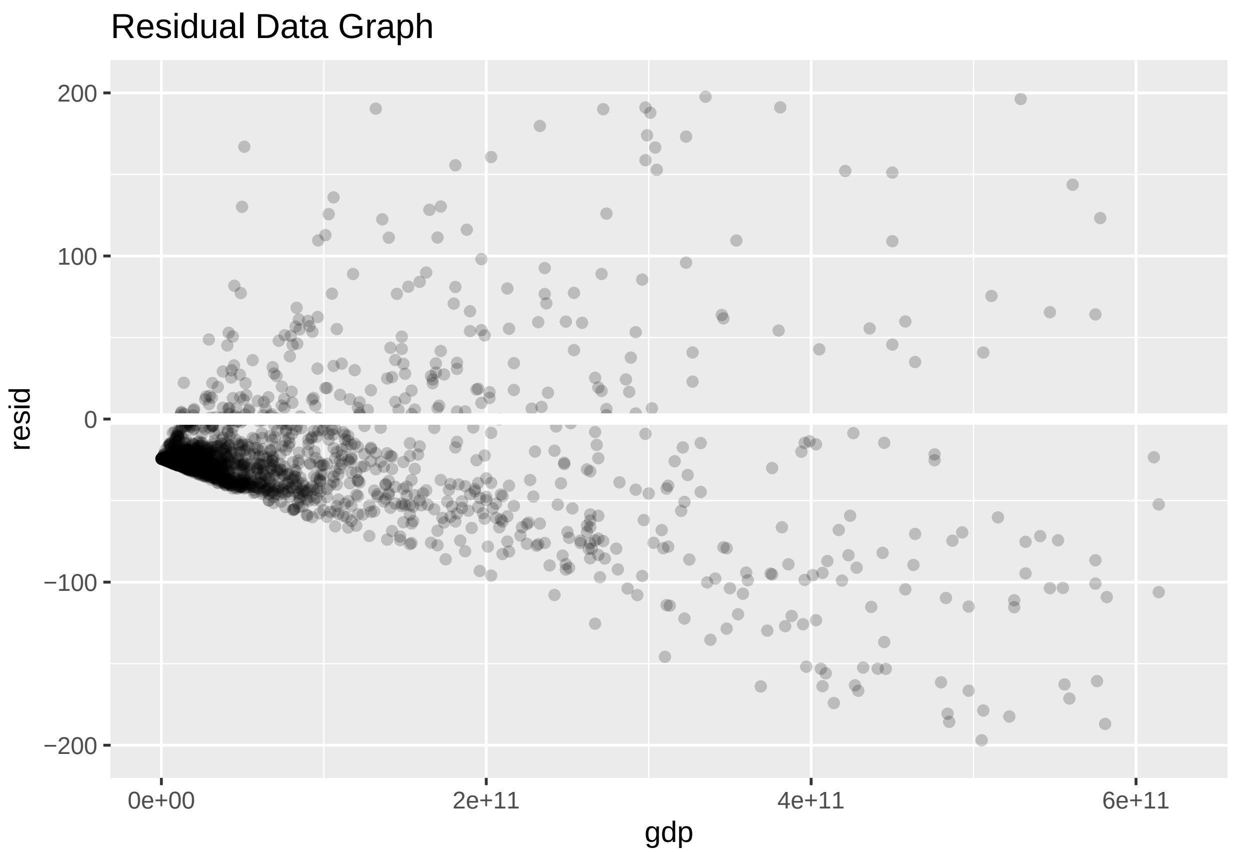
```
resid <- add_residuals(valid, model)
ggplot(data = resid, mapping = aes(x=gdp, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  ggtitle("Residual Data Graph")
```



```
ggplot(data = resid, mapping = aes(x=gdp, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  xlim(0,62500000000)+
  ylim(-100,100)+
  ggtitle("Residual Data Graph")
```



```
ggplot(data = resid, mapping = aes(x=gdp, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  xlim(0,625000000000)+
  ylim(-200,200)+
  ggtitle("Residual Data Graph")
```



The first graph is no x and y constraints, but I soon realized that world is an outlier in this so I should probably remove that from the data set next time.
Cutting down the x and y to look at the graph lets us see the patterns better in graph 2 and 3.

Looking at these residual graphs based on the gdp, it seems like there might be a pattern in the residual graph which shouldn't be there. If this model is effective, then we should see a random scattering of dots in the residual graph.

Removing world and making this a polynomial equation instead should fit the data more and allow for better predictions. Looking back at the graph of co2 vs gdp, you can see that the predicted trend from geom_smooth() doesn't look very linear in the slightest with a good exponential curve in it. A polynomial function should be able to capture the trend seen.

```
co2.data <- filter(co2.data, country != "world")
model <- train(co2 ~ ns(gdp, 100), data = tv_data, method = "lm", trControl = train.control)

predictions <- add_predictions(valid, model)
R2(predictions$pred, predictions$co2)
```

```
## [1] 0.9807677
```

```
MAE(predictions$pred, predictions$co2)
```

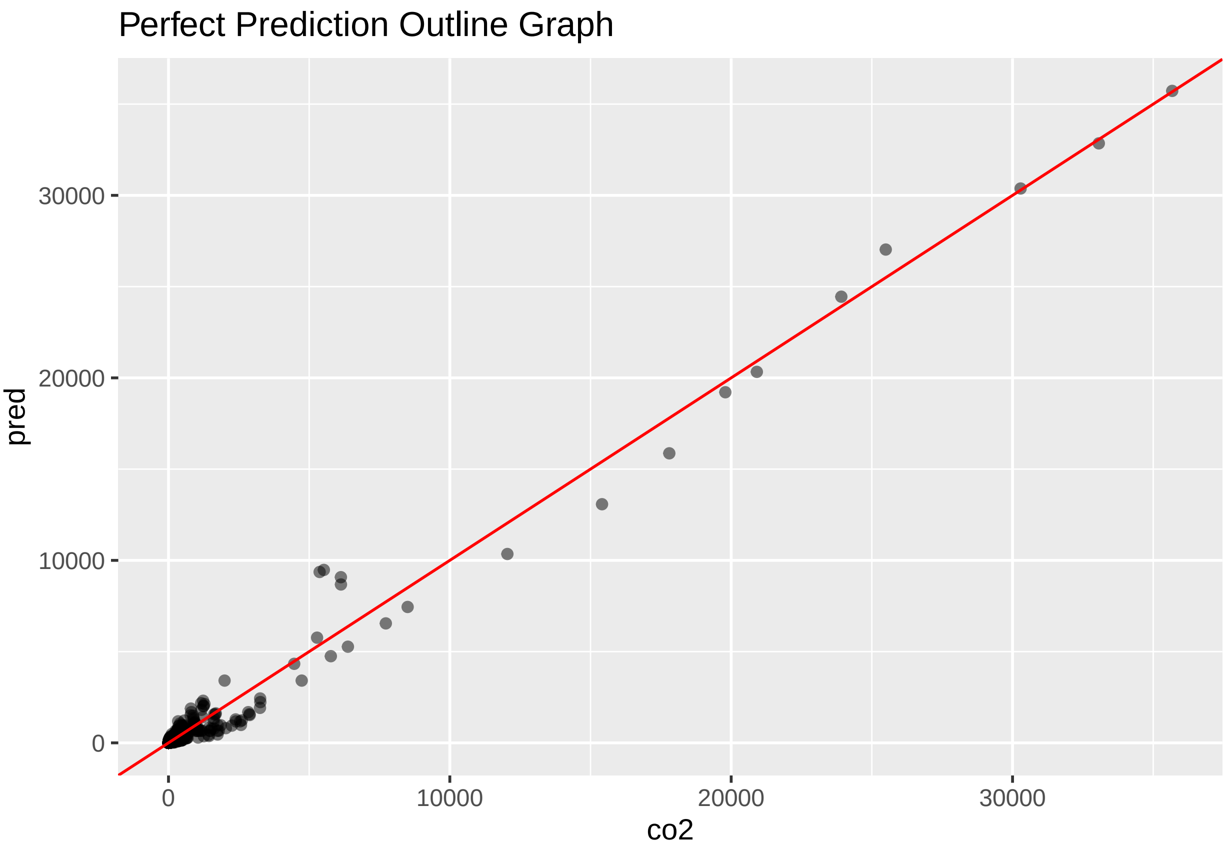
```
## [1] 58.42826
```

```
RMSE(predictions$pred, predictions$co2)
```

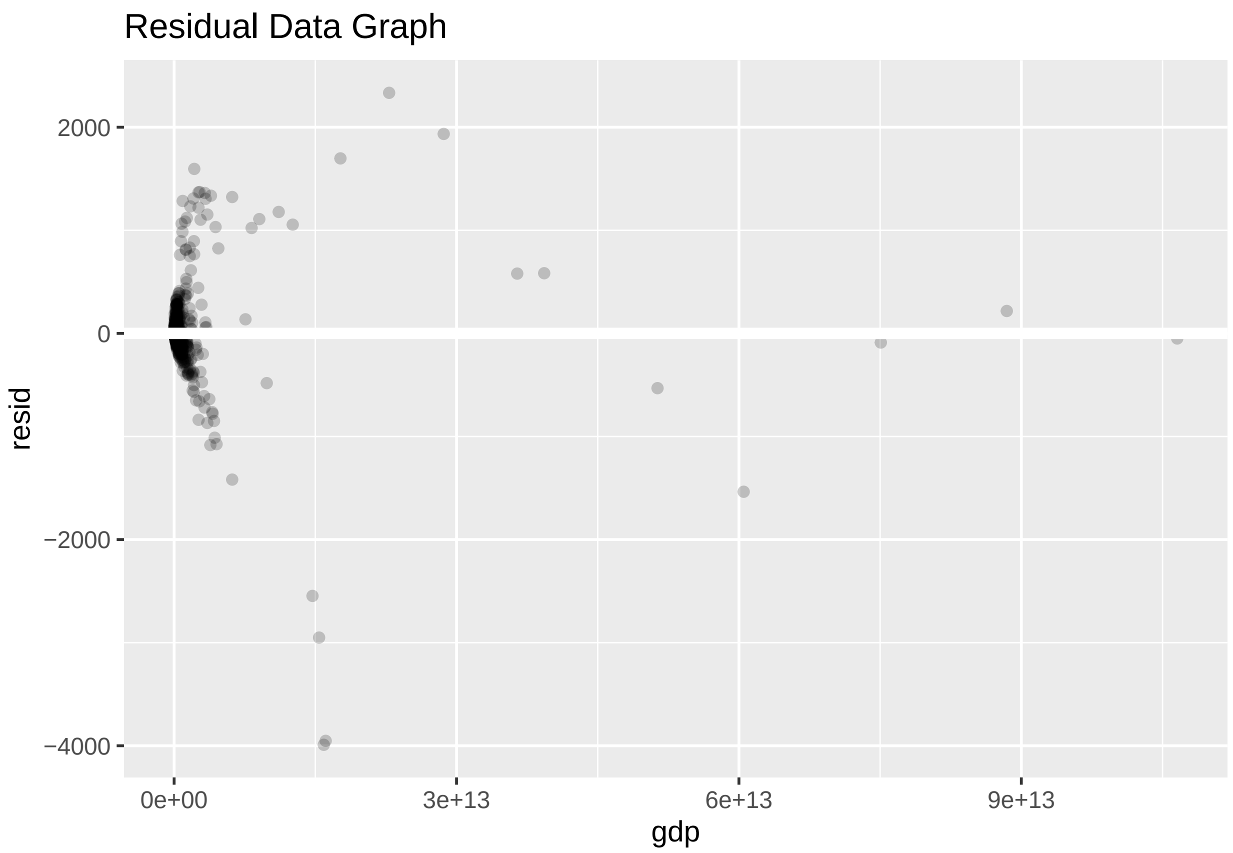
```
## [1] 221.4866
```

Using a polynomial function for making a model seems to produce a much better result than just a linear model. The R2 value is a lot closer to 1 and MAE/RMSE got lower values.

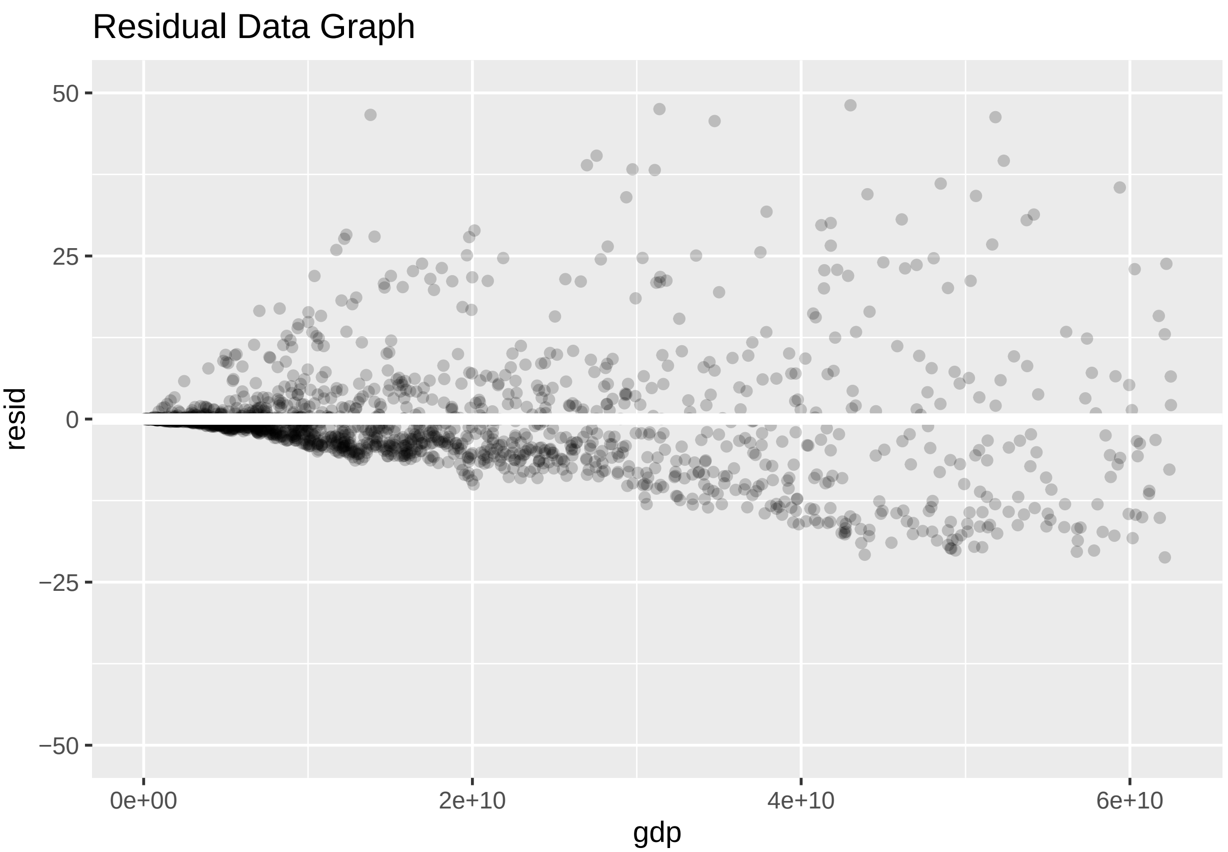
```
ggplot(data = predictions, mapping=(aes(x = co2, y = pred)))+
  geom_point(alpha = 0.5)+
  geom_abline(intercept = 0, slope = 1, color = "red")+
  ggtitle("Perfect Prediction Outline Graph")
```



```
resid <- add_residuals(valid, model)
ggplot(data = resid, mapping = aes(x=gdp, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  ggtitle("Residual Data Graph")
```



```
ggplot(data = resid, mapping = aes(x=gdp, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  xlim(0,62500000000)+
  ylim(-50,50)+
  ggtitle("Residual Data Graph")
```



Using a polynomial and graphing out the residual/predictions shows that this model is getting a much better picture of patterns that can be found in the data

Though I feel like I might be able to get better results with more data including renewable energy and maybe some trends of clean energy that different countries are taking.

```
energy.data <- read_csv("clean_energy_data.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   ‘2 019‘ = col_logical()
## )
```

```
## See spec(...) for full column specifications.
```

energy.data comes from https://www.irena.org/Statistics/View-Data-by-Topic/ Capacity-and-Generation/Country-Rankings and is maintained on a yearly basis from 2000 until now. It splits up the data on a source and year level giving us a wide range of angles to look at for renewable energy on a country basis.

This data was grabbed from a .xlsm file through Excel. It allowed you to query the data base from International Renewable Energy Agency with different calls. I I ended up calling the data base looking only at renewable energy made and downloaded the .csv file after

The categorical variables for energy.data are:

- Country/area - location on the globe the data is from

- Technology - source of renewable energy in the power grid

The continuous variables for energy.data are:

- 2000 - clean electricity generated that year
- ...
- 2019 - each year is the same measurements

While this data does have some empty spaces and NULL values in spots, I'm not going to remove those variables. The main reason I want to keep them is that I need to tidy up the data with the original number of rows first, and filtering isn't too hard later on

Though the .csv file didn't quite fill in all of the country and place names so I'm going to have to supplement that and fill out the column.

```
for (i in 0:4749){
  if(i %% 19 == 0){
    word <- toString(energy.data[i+1,1])
  }else{
    energy.data[i+1,1] <- word
  }
}
```

Along with fixing the data, I want to be able to combine the energy data with co2 data so I'm going to have to trim this all down and make it join-able according to the co2 ordering.

```
energy.data <- filter(energy.data, Technology == "Total renewable energy")
energy.data <- select(energy.data, -c(Technology))
names(energy.data)[1] <- "country"
names(energy.data)[2] <- "2000"
names(energy.data)[3] <- "2001"
names(energy.data)[4] <- "2002"
names(energy.data)[5] <- "2003"
names(energy.data)[6] <- "2004"
names(energy.data)[7] <- "2005"
names(energy.data)[8] <- "2006"
names(energy.data)[9] <- "2007"
names(energy.data)[10] <- "2008"
names(energy.data)[11] <- "2009"
names(energy.data)[12] <- "2010"
names(energy.data)[13] <- "2011"
names(energy.data)[14] <- "2012"
names(energy.data)[15] <- "2013"
names(energy.data)[16] <- "2014"
names(energy.data)[17] <- "2015"
names(energy.data)[18] <- "2016"
names(energy.data)[19] <- "2017"
names(energy.data)[20] <- "2018"
names(energy.data)[21] <- "2019"
energy.data <- select(energy.data, -c('2019'))
energy.data <- pivot_longer(energy.data ,c('2000','2001','2002','2003','2004','2005','2006','2007','2008

energy.data$year <- as.double(energy.data$year)
```

```r
for (i in 0:4749){
  energy.data[i+1,3] <- sub(" ","",toString(energy.data[i+1,3]))
  energy.data[i+1,3] <- sub(" ","",toString(energy.data[i+1,3]))
  energy.data[i+1,3] <- sub(" ","",toString(energy.data[i+1,3]))
}
energy.data$total_clean_energy <- as.numeric(energy.data$total_clean_energy)
energy.co2.data <- inner_join(co2.data, energy.data) %>%
  filter(year > 1999, country != "World", country != "China")
```

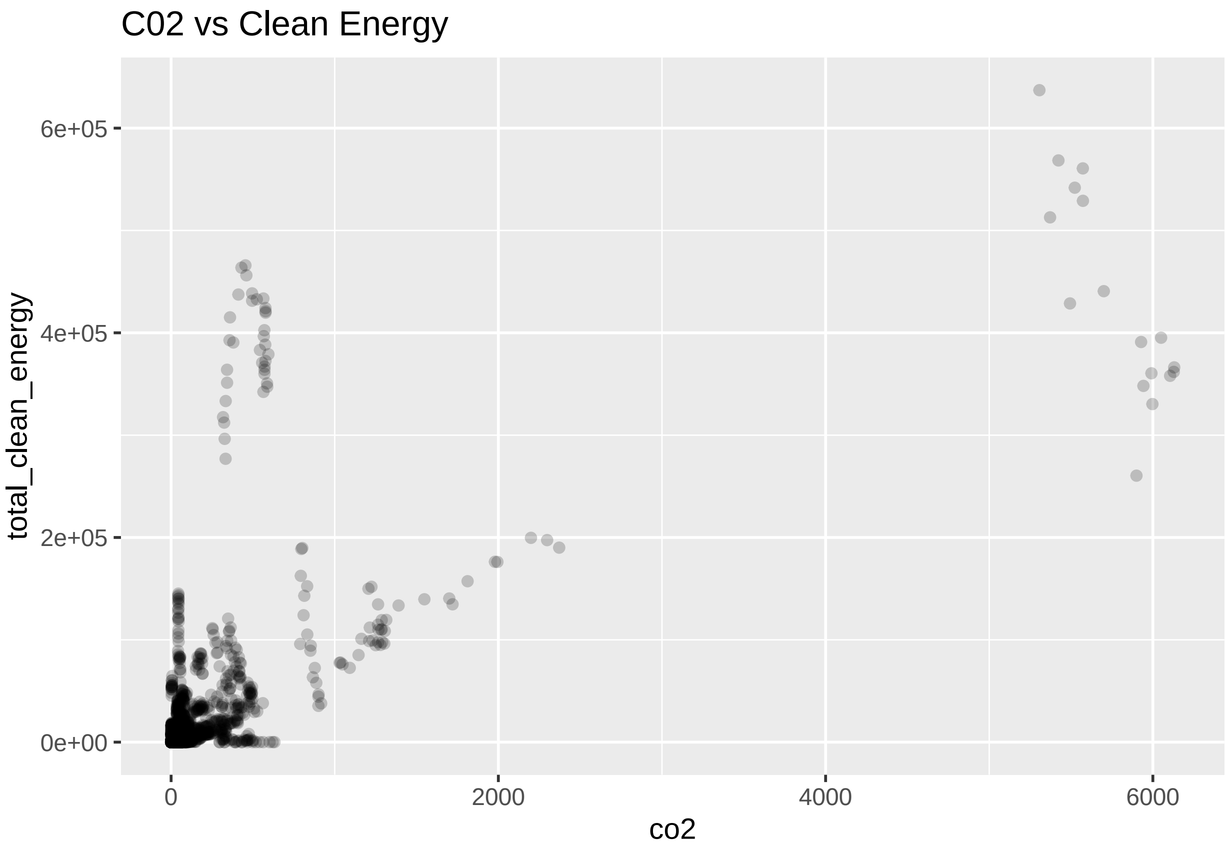
```
## Joining, by = c("country", "year")
```

To combine the two I had to rotate the columns of years to one row, as well as making the clean energy measurements numbers instead of strings. String wont work for models later on.

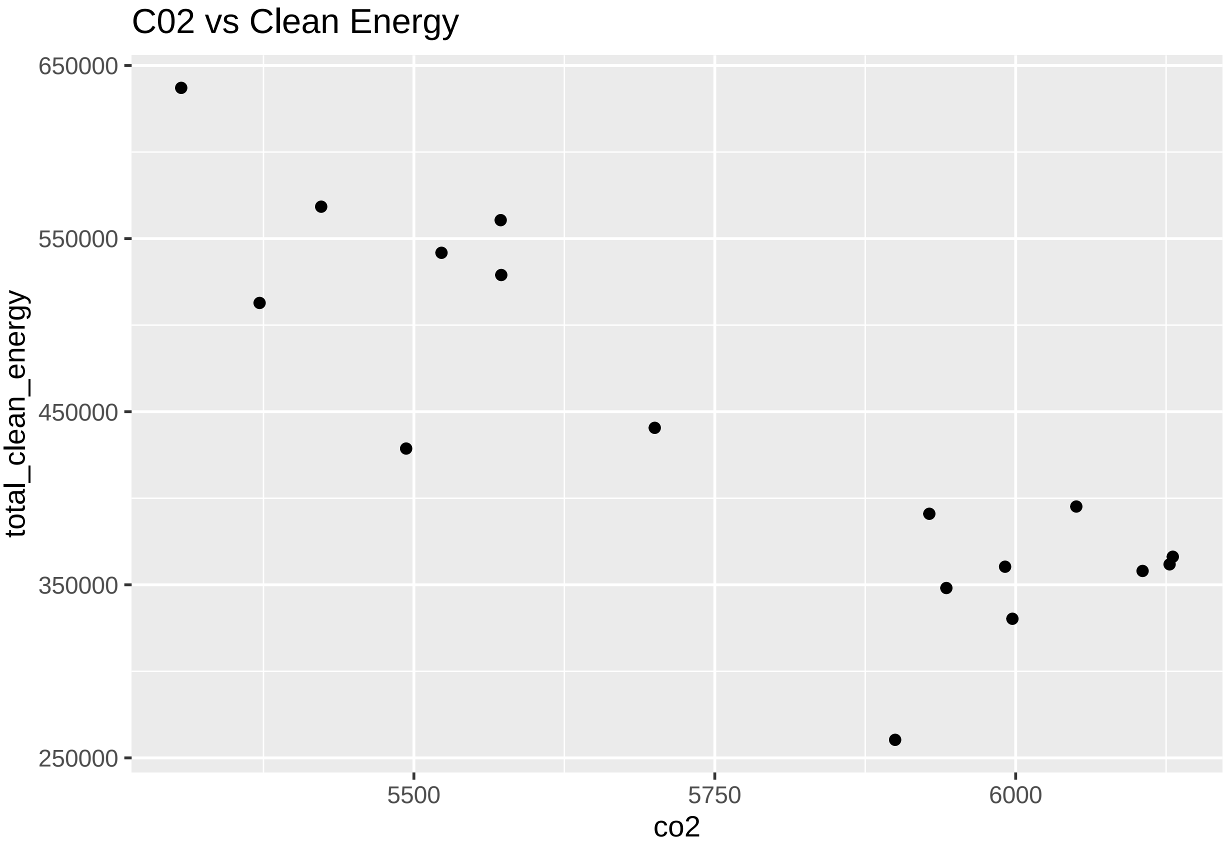
Filtered out China and World as the data for China is highly debated in the data set, and World isn't going to help since it's has such extreme numbers and wont add much to the model.

With the two data sets joined, we can split up the data and try making a linear model again but with clean energy involved. While this data will also be limited in the year span, it might still have good results to consider.

```r
ggplot(data = filter(energy.co2.data),  mapping = aes(x=co2, y=total_clean_energy))+
  geom_point(alpha = 0.2)+
  ggtitle("C02 vs Clean Energy")
```



```
ggplot(data = filter(energy.co2.data, country == "United States"),  mapping = aes(x=co2, y=total_clean_
  geom_point()+
  ggtitle("C02 vs Clean Energy")
```



While this data set seems to be somewhat interesting with the correspondence of clean energy having higher co2 output, it seems like the graph is having a rough time graphing it all out. Though when looking at only the United States, it seems like there might be a good correspondence there with more clean energy resulting in less co2 output.

We are now going to make a basic linear model with gdp and clean energy in mind to predict the co2 of a country. This is also going to take the same data splitting ratio as the model before.

```
energy.co2.data <- filter(energy.co2.data, gdp != is.na(gdp), co2 != is.na(co2))
train_rows <- as.vector(createDataPartition(energy.co2.data$co2, p = 0.8, list = FALSE))
tv_data <- energy.co2.data[train_rows, ]
test <- energy.co2.data[-train_rows, ]

train_rows <- as.vector(createDataPartition(tv_data$co2, p = 0.75, list = FALSE))
valid <- tv_data[-train_rows, ]
train <- tv_data[train_rows, ]

model <- train(co2 ~ gdp + total_clean_energy, data = tv_data, method = "lm", trControl = train.control

predictions <- add_predictions(valid, model)
R2(predictions$pred, predictions$co2)
```

```
## [1] 0.9681292
```

```
MAE(predictions$pred, predictions$co2)
```

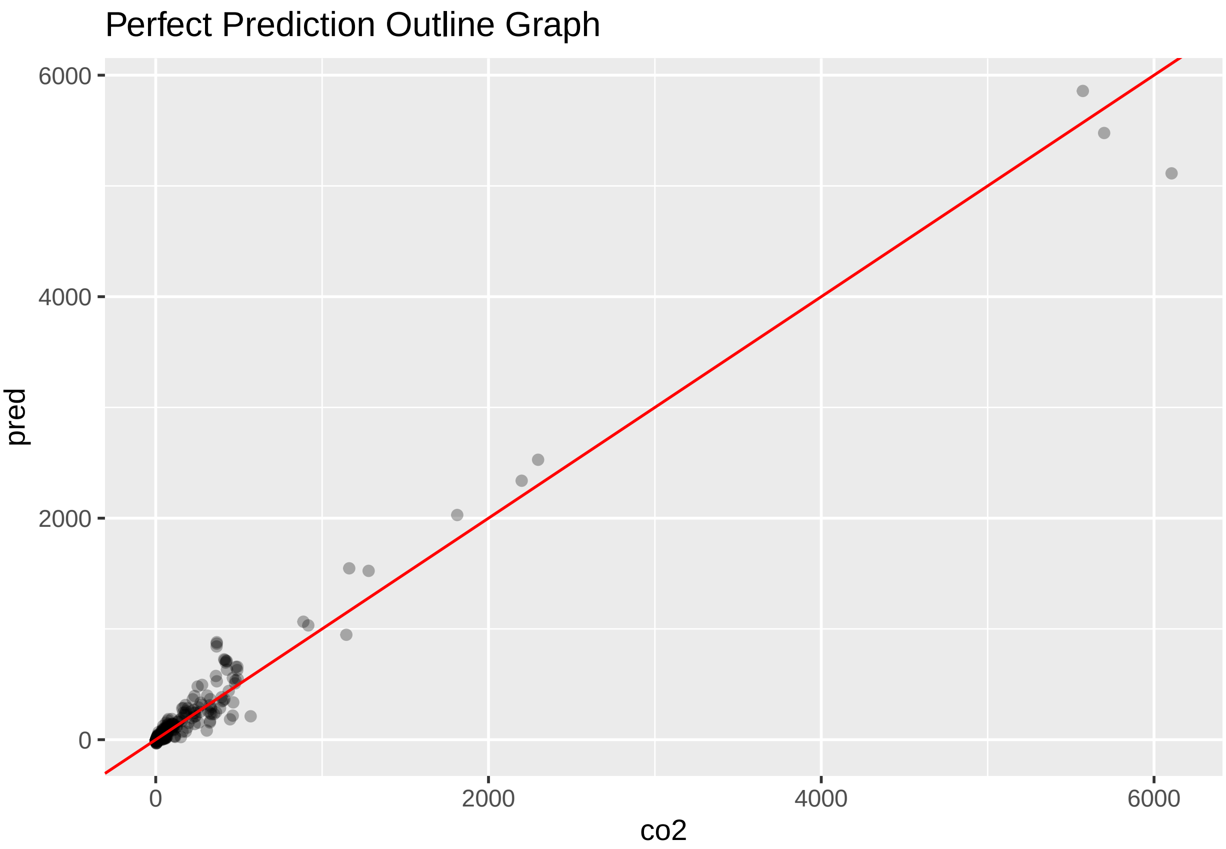
```
## [1] 39.93718
```

```
RMSE(predictions$pred, predictions$co2)
```

```
## [1] 90.12624
```

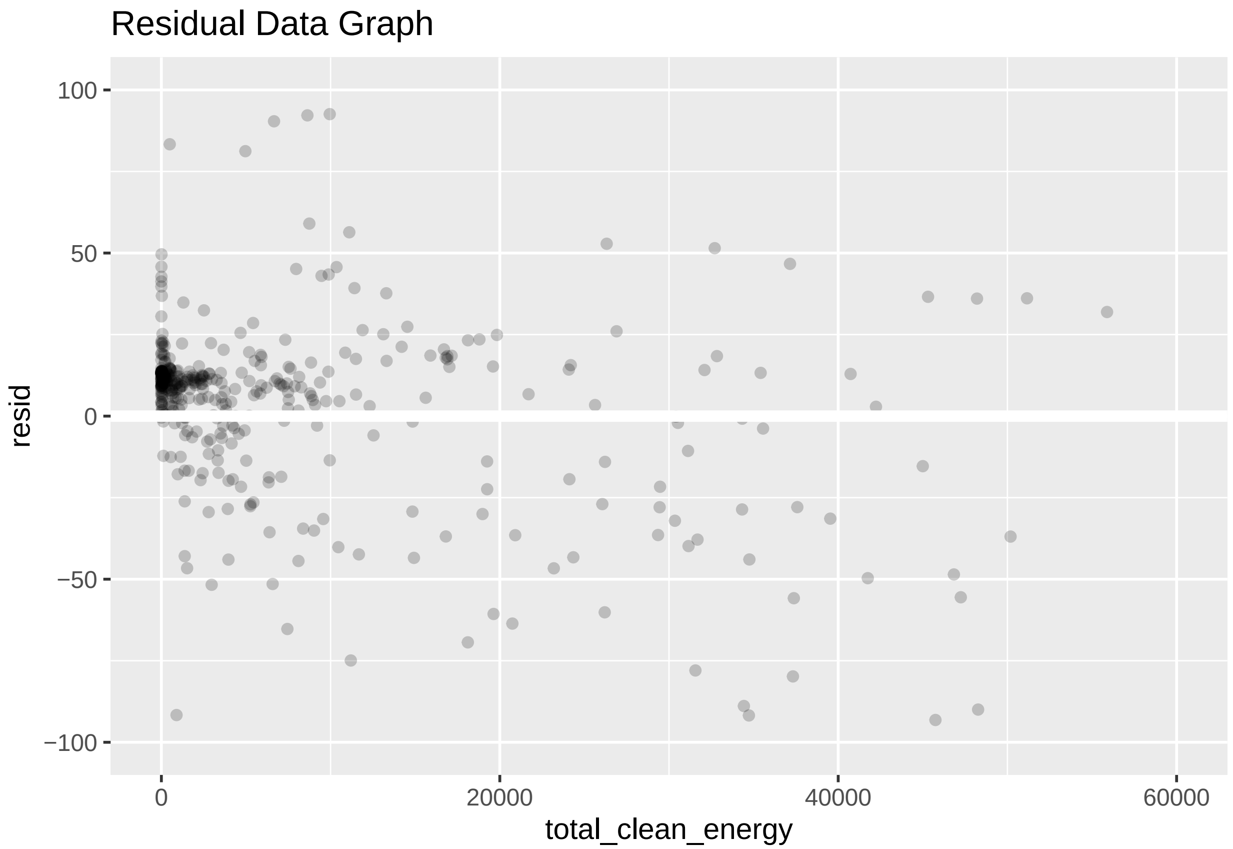
From the R2 value, this seems to have made a very interesting result and actually increased the value of the regular linear model test I first did with just co2 and gdp. This I did not expect as the graph for co2 and clean energy seems to show both going up at similar rates.

```
ggplot(data = predictions, mapping=(aes(x = co2, y = pred)))+
  geom_point(alpha = 0.3)+
  geom_abline(intercept = 0, slope = 1, color = "red")+
  ggtitle("Perfect Prediction Outline Graph")
```

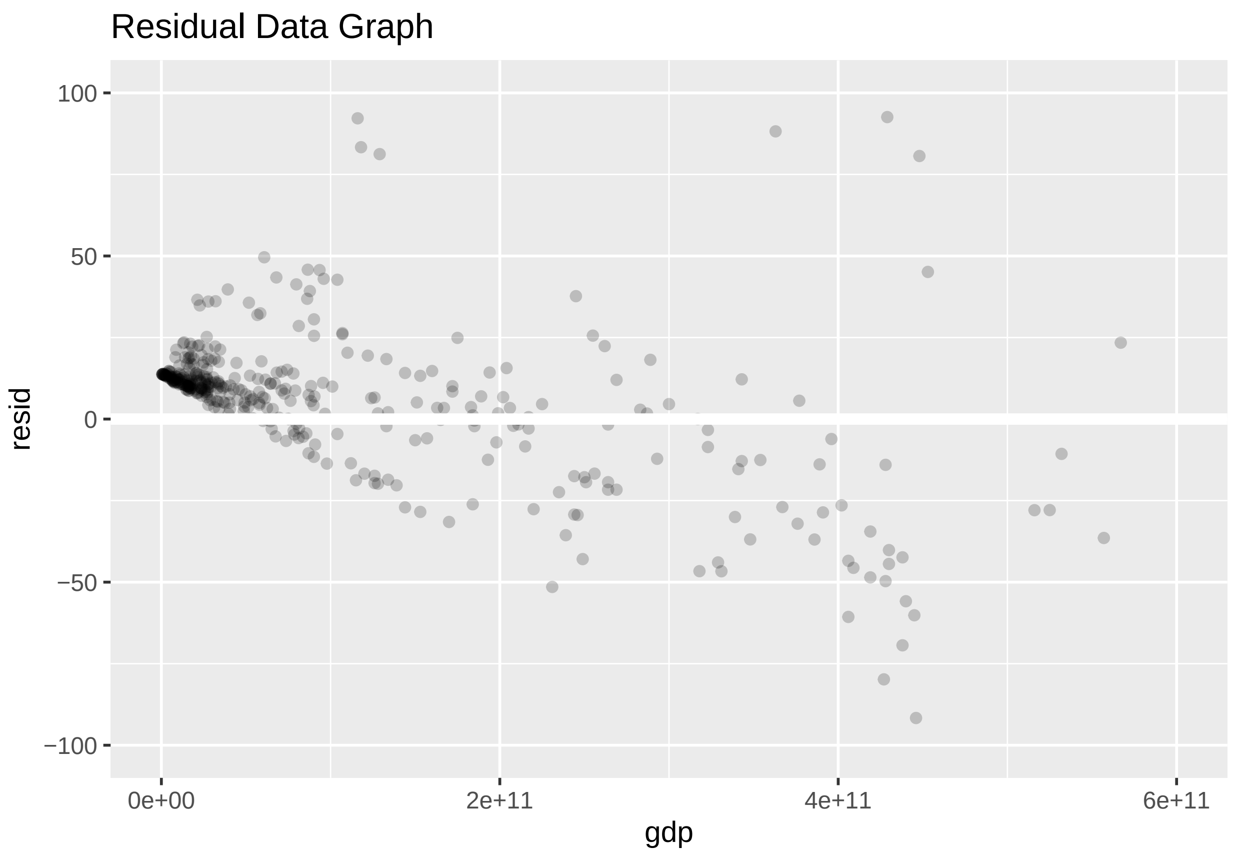


The predictions seem very clumped around the perfect prediction mid-line. This is a very good sign for the model accuracy.

```
resid <- add_residuals(valid, model)
ggplot(data = resid, mapping = aes(x=total_clean_energy, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  xlim(0,60000)+
  ylim(-100,100)+
  ggtitle("Residual Data Graph")
```



```
resid <- add_residuals(valid, model)
ggplot(data = resid, mapping = aes(x=gdp, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  xlim(0,600000000000)+
  ylim(-100,100)+
  ggtitle("Residual Data Graph")
```



Both residual graphs are focused on the chunk of the data ignoring the outliers that exist at very high extremes. Ignoring the outliers though, you can see a trend of under fitting here with the majority of dots being just a tad over the 0 line.

It's hard to tell if these results are better than the polynomial data that was made earlier, but getting some feedback on the polynomial model, there's a good chance it overfit. This means that using these two data points were really a much better choice for modeling.

To improve upon just using gdp to make a model though, I want to now look into using exp(x) instead of n(x,100) and just x.

Since exp() doesn't actually work, I'm going to log(y) instead to provide a similar equation to exponential.

```
model <- train(log(co2) ~ gdp, data = tv_data, method = "lm", trControl = train.control)

predictions <- add_predictions(valid, model)
R2(predictions$pred, predictions$co2)
```

```
## [1] 0.9662758
```

```
MAE(predictions$pred, predictions$co2)
```

```
## [1] 128.3766
```

```
RMSE(predictions$pred, predictions$co2)
```

```
## [1] 518.8304
```

comparing these values to just the straight linear model with no variation, it seems to have done worse with the error margins. The r2 is very similar but that doesn't mean too much with the difference in error.

It still seems like using the second data set to make a multi variable linear model got the most accurate results out of all the model tests. Though I'm going to also test with a smaller number than 100 for a polynomial model before switching just to be thorough.

```
co2.data <- filter(co2.data, country != "world")
model <- train(co2 ~ ns(gdp, 3), data = tv_data, method = "lm", trControl = train.control)

predictions <- add_predictions(valid, model)
R2(predictions$pred, predictions$co2)
```

```
## [1] 0.9757895
```

```
MAE(predictions$pred, predictions$co2)
```

```
## [1] 33.25704
```

```
RMSE(predictions$pred, predictions$co2)
```

```
## [1] 79.02233
```

This polynomial function on just gdp, isnt going to be overfit, but does seem to still be a little worse than the multi variable lm model. It seems like just gdp isn't quite enough data to determine an as accurate result when comparing to using two data points.

I'm going to be switching my model back to using clean energy data as well to validate on the test data and see what kind of results we will get with that.

```
model <- train(co2 ~ gdp + total_clean_energy, data = tv_data, method = "lm", trControl = train.control

predictions <- add_predictions(test, model)
R2(predictions$pred, predictions$co2)
```

```
## [1] 0.9653536
```

```
MAE(predictions$pred, predictions$co2)
```

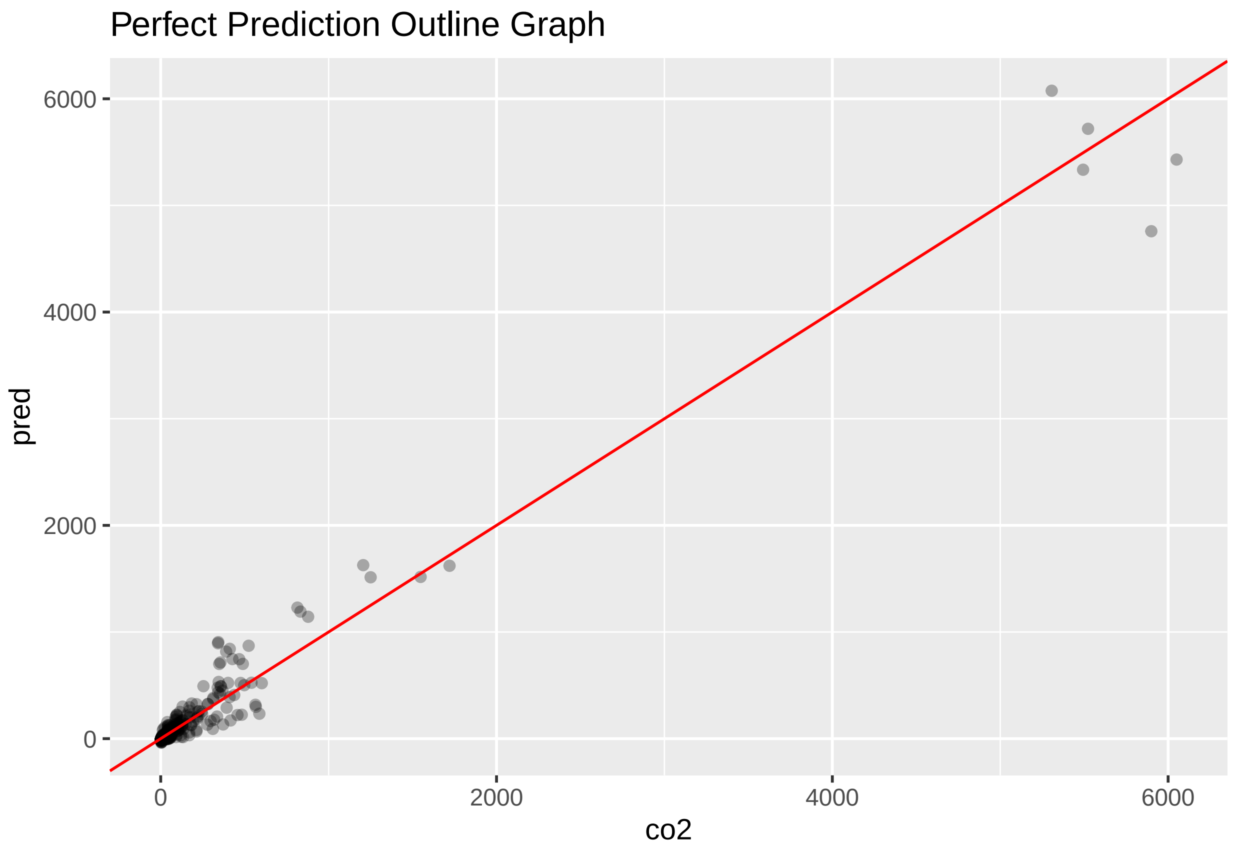
```
## [1] 46.72996
```

```
RMSE(predictions$pred, predictions$co2)
```

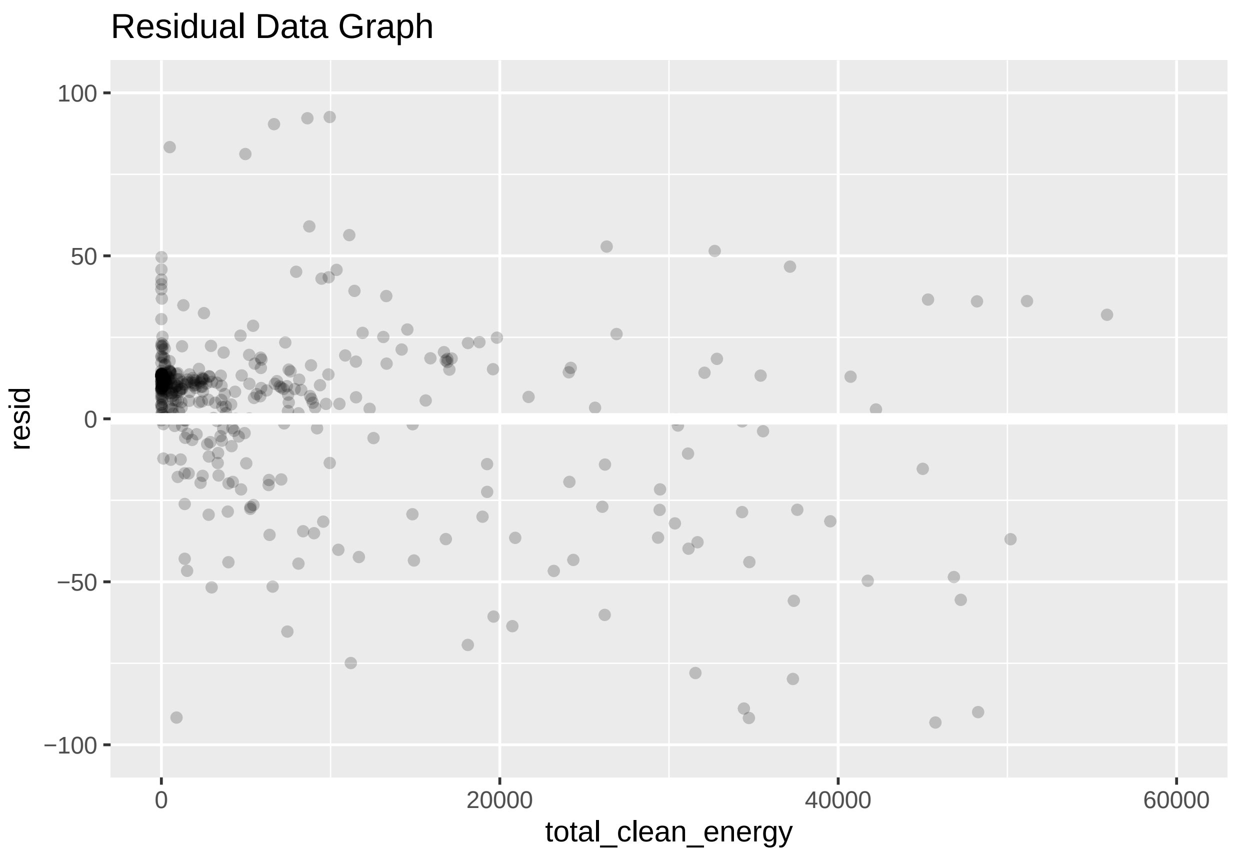
```
## [1] 111.5247
```

While the test data seems to have a worse error value and slightly lower r2, it looks like we still have a pattern found with the test data. Considering valid was trained on, it makes sense that it would do better than test.

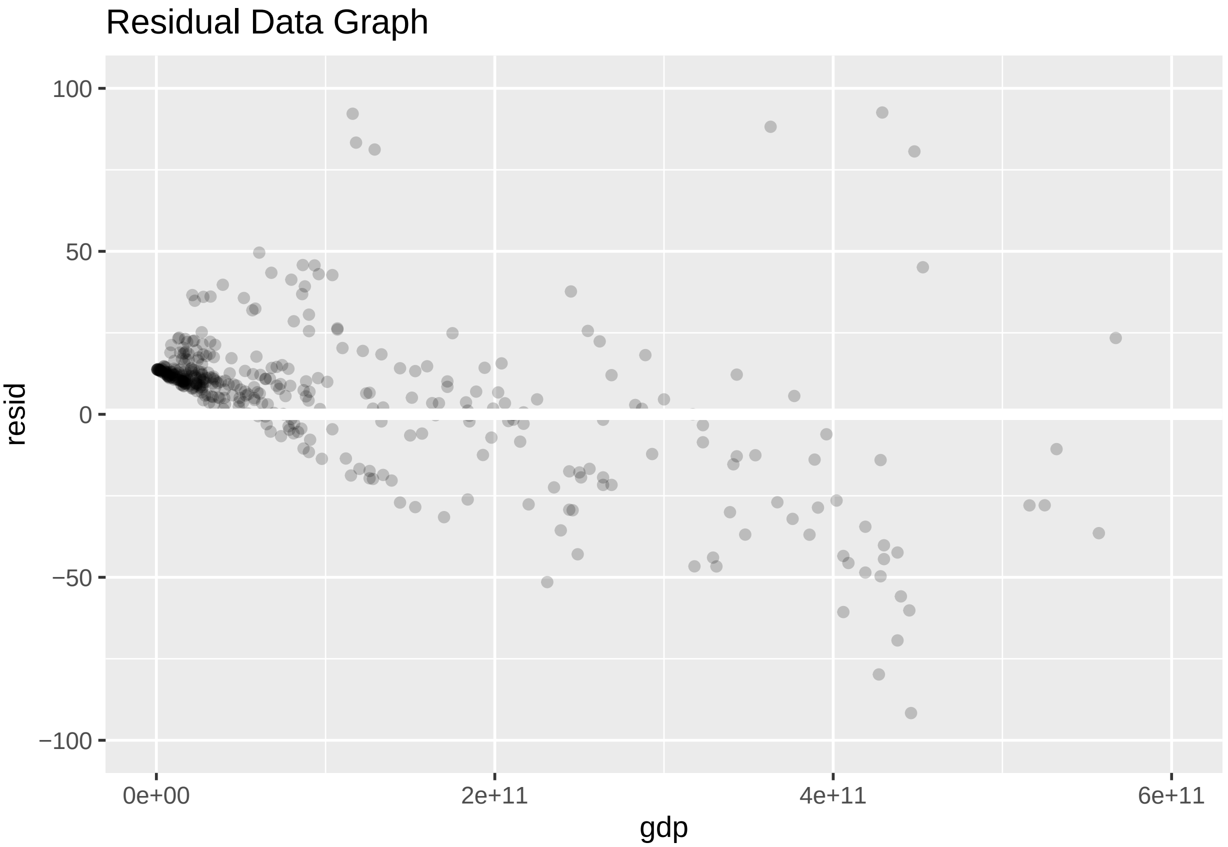
```
ggplot(data = predictions, mapping=(aes(x = co2, y = pred)))+
  geom_point(alpha = 0.3)+
  geom_abline(intercept = 0, slope = 1, color = "red")+
  ggtitle("Perfect Prediction Outline Graph")
```



```
resid <- add_residuals(valid, model)
ggplot(data = resid, mapping = aes(x=total_clean_energy, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  xlim(0,60000)+
  ylim(-100,100)+
  ggtitle("Residual Data Graph")
```



```
resid <- add_residuals(valid, model)
ggplot(data = resid, mapping = aes(x=gdp, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  xlim(0,600000000000)+
  ylim(-100,100)+
  ggtitle("Residual Data Graph")
```



The prediction and residual graphs look very similar to the previous graphs on valid data. Looks like the same patterns were picked up in residual again with an under-prediction on lower co2, and maybe an over-prediction on higher co2

Wrapping it all up

Overall these models surprised me in multiple ways. I honesty expected the first linear model to work pretty well and didn't see the failure coming. although switching to the polynomial equation really helped match the curve with a much better result than before, it was probably over fitting with 100 points. Testing out 3 points instead seemed to get decent results while avoiding overfitting though.

Looking into log(co2) = gdp was an interesting test to see if I could find patterns without overfitting, but it seems like there was still a little too much variance giving me a result that was worse than regular lm()

Grabbing the electrical grid information seemed to make a difference as well with the R2 being very similar and error values being lower. The one fair point that was true with most models I looked into though is that they under or over predicted. In the electric-co2 model we do end up under predicting lower values, while over predicting on higher values. It will find a decent summary of co2 outputs but overall isn't the most accurate.

I still don't think there is an ethical or social issue with the research I'm working on. Any improvement on co2 predictions or help that we can get will help society in the long run by getting global warming blame out there. If the United Nations could see the worst countries and act on that, we would probably be able to curb the warming.