Deliverable 2

Carlson Smith

10/15/2020

Data from part 1 to be used in part 2

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

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.

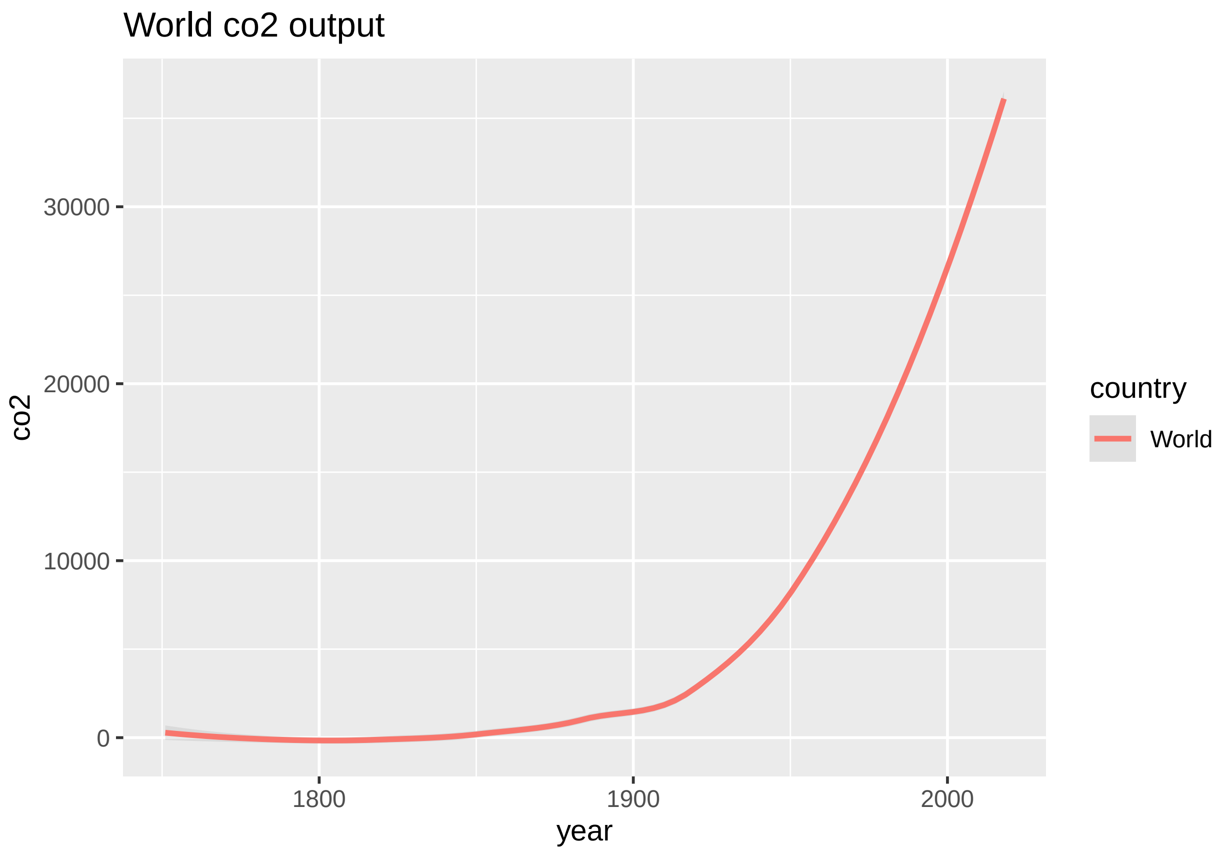
Power grids in a country might also have an affect on this area as some countries make their way towards a 0 carbon life style. This might now actually push much around though as construction and vehicles are also another big factor in this area.

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.

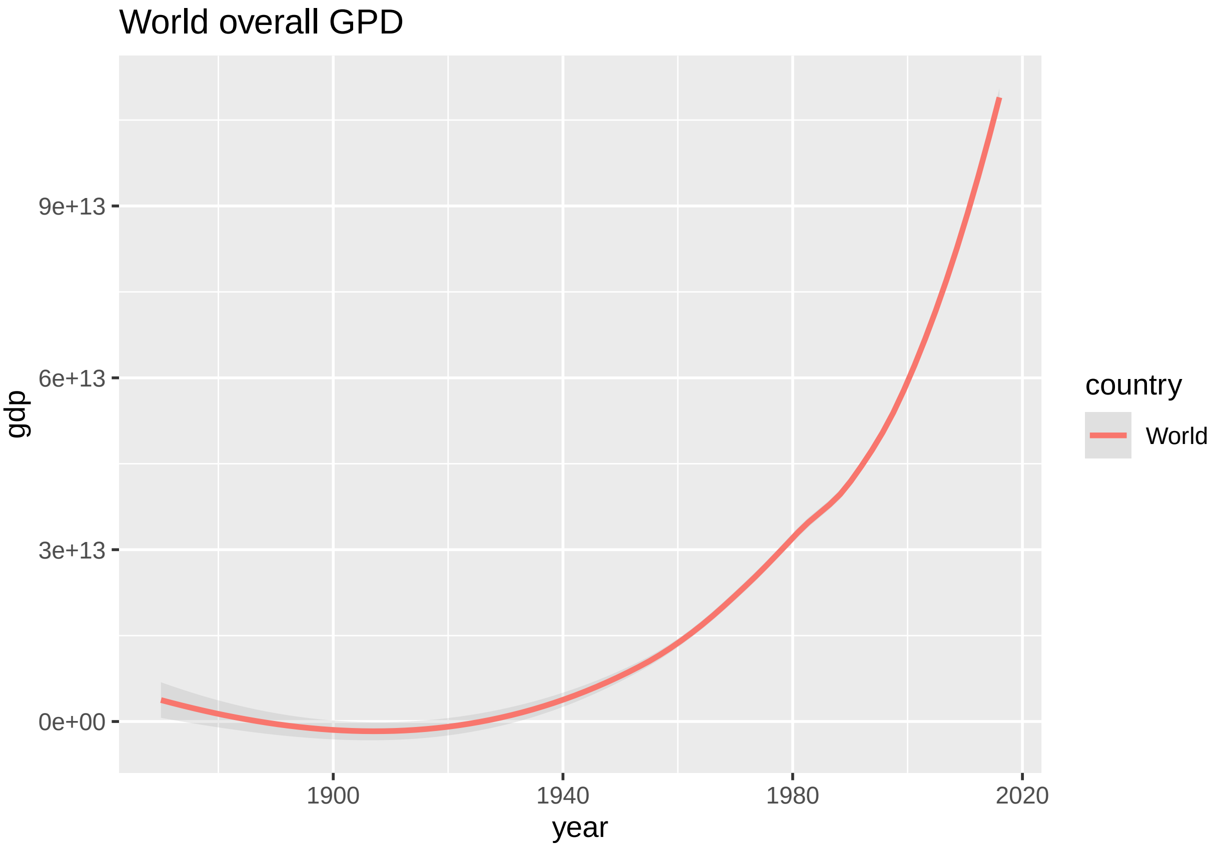
```
ggplot(data = filter(co2.data, country == "World"), mapping = aes(x=year, y=co2, color = country))+
  geom_smooth(alpha = 0.2)+
  ggtitle("World co2 output")
```

```
## ‘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’
```

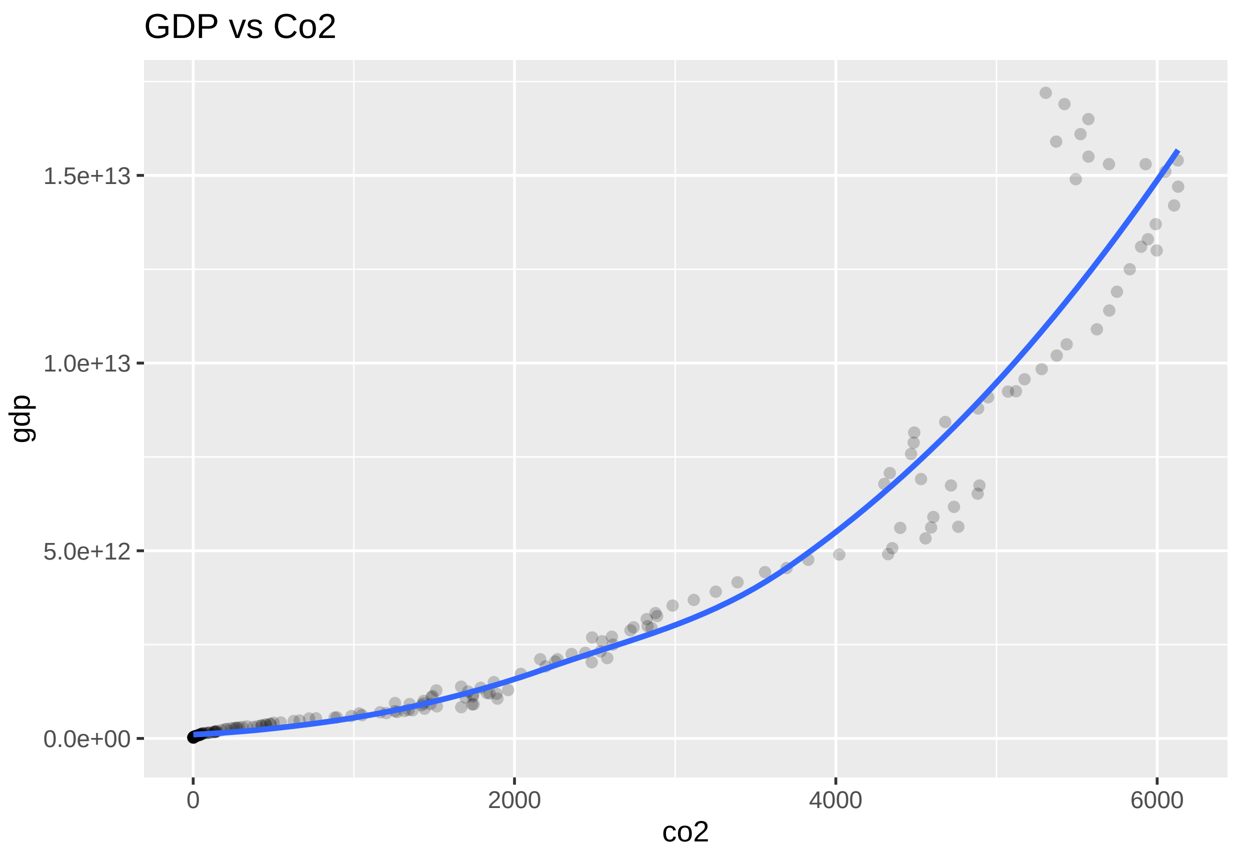


From the data science questions I wanted to look into from the initial data analysis, it seems like gdp and co2 might have a very close relation based on the general look of the world. THough I do want to look into specific countries since I know the USA has curbed their co2 outputs but their gdp is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

```
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, ]
```

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)

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

```
## [1] 0.9436283
```

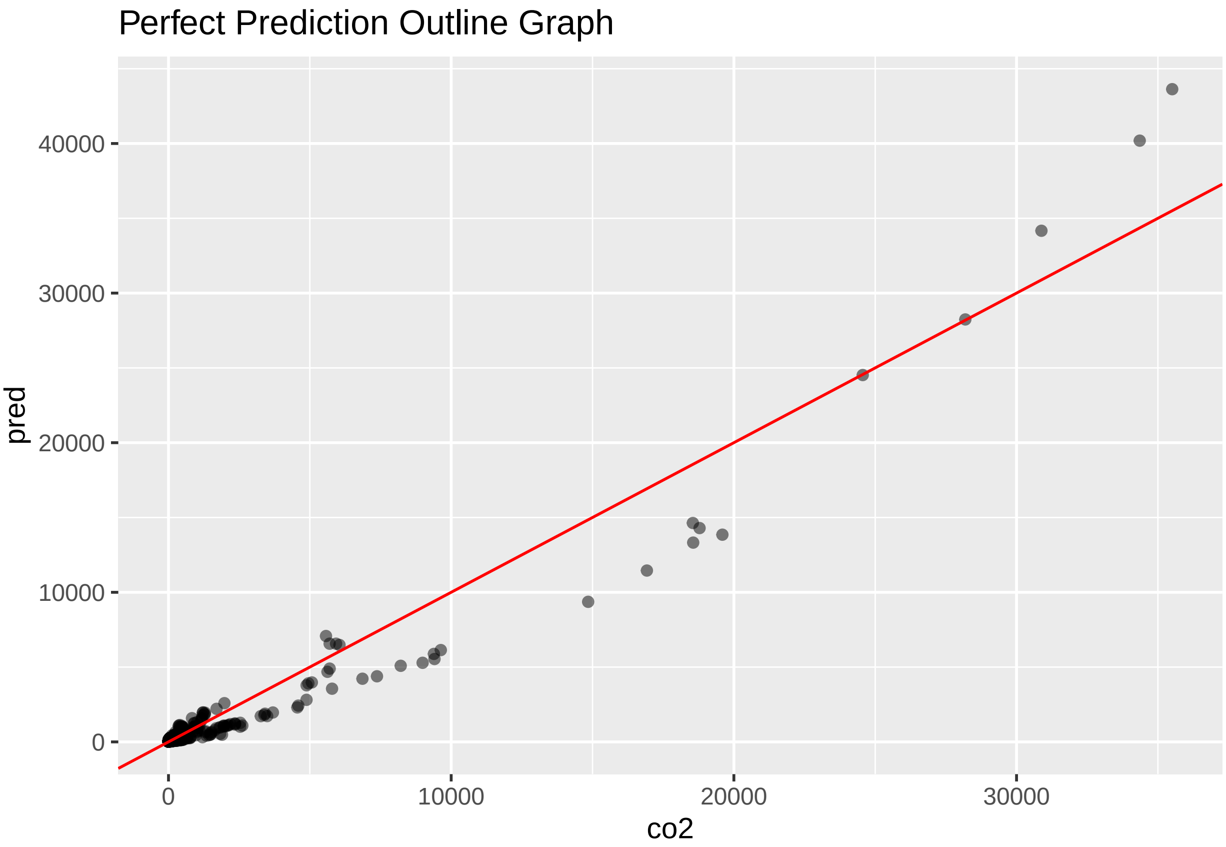
```
MAE(predictions$pred, predictions$co2)
```

```
## [1] 97.04375
```

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

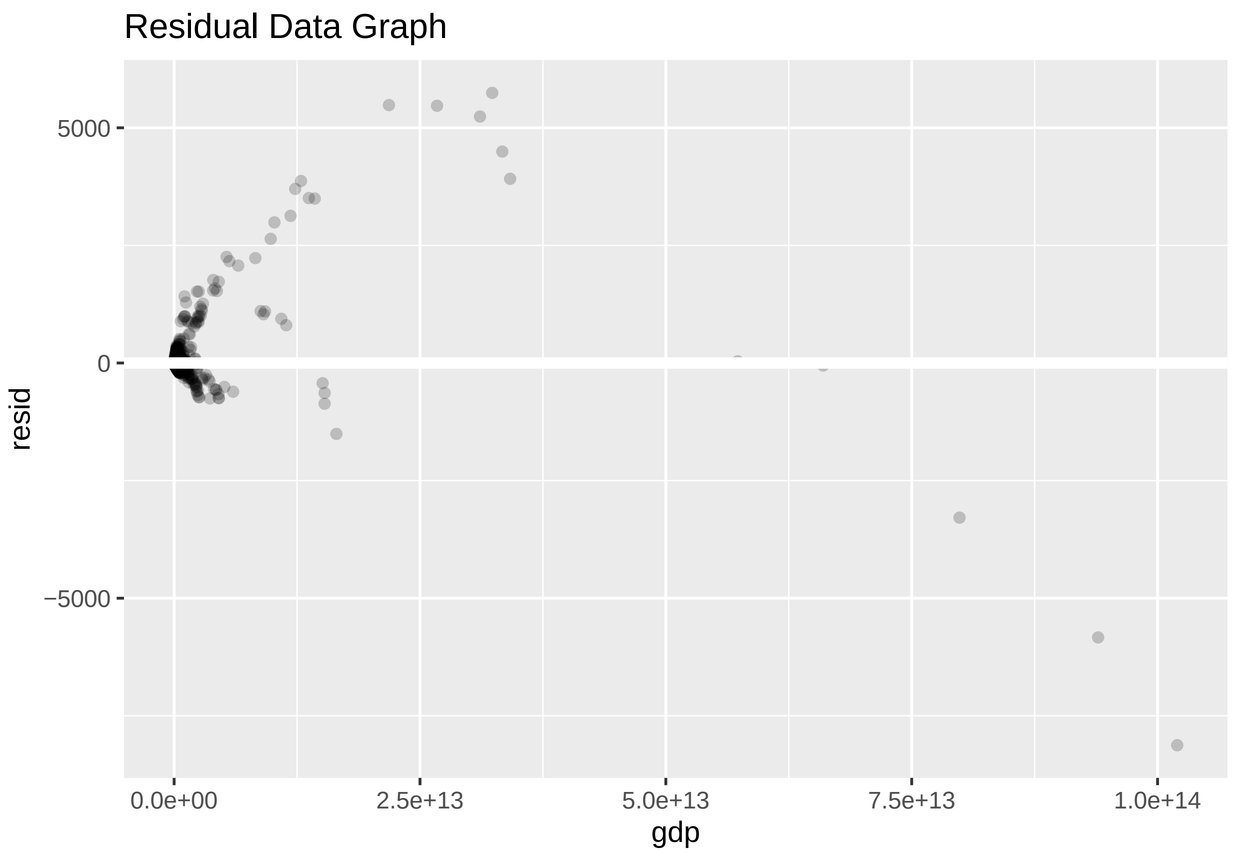
```
## [1] 412.2577
```

```
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")
```

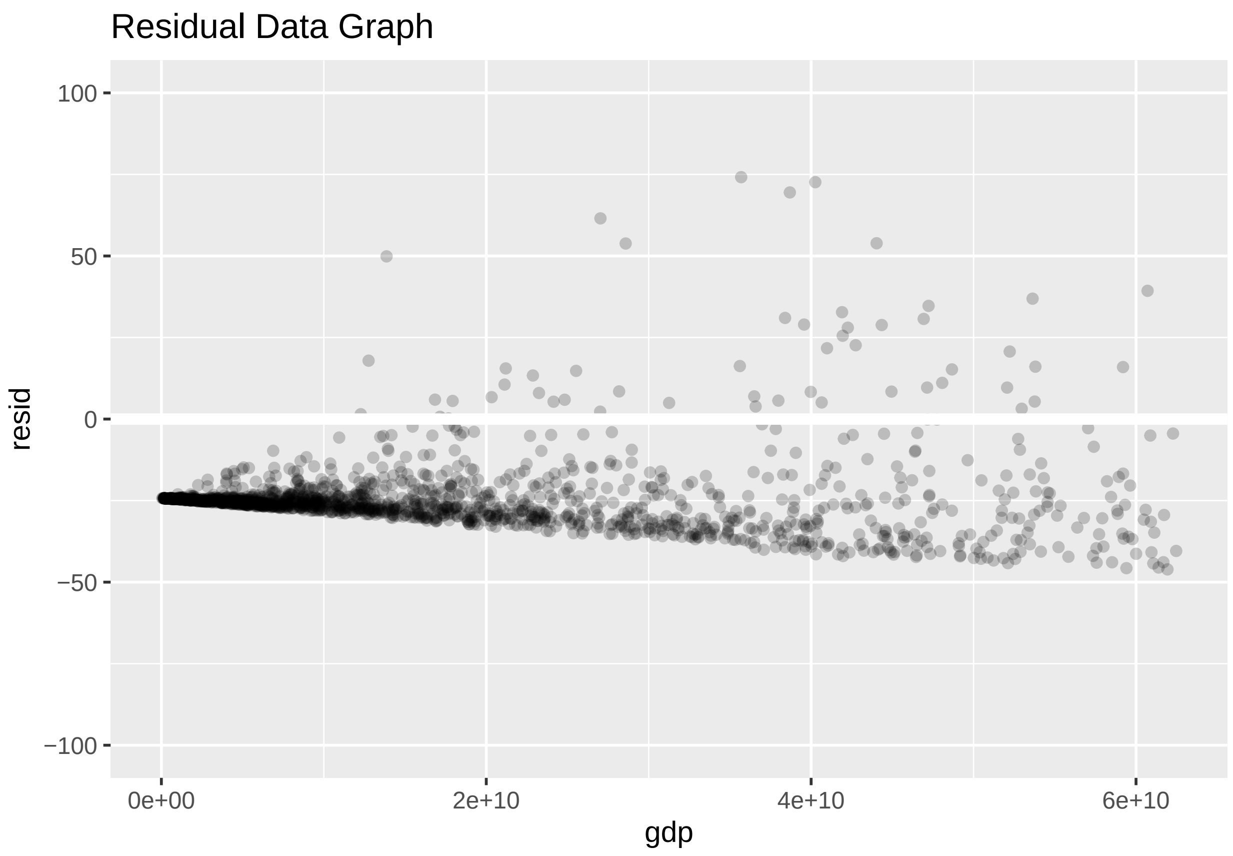


Looking at these data pieces it looks like our linear model found some pattern it could make into an eq as the R2 value isn't too far from 1. But looking at the prediction graph, it seems like a linear model might not be able to capture the exact values.

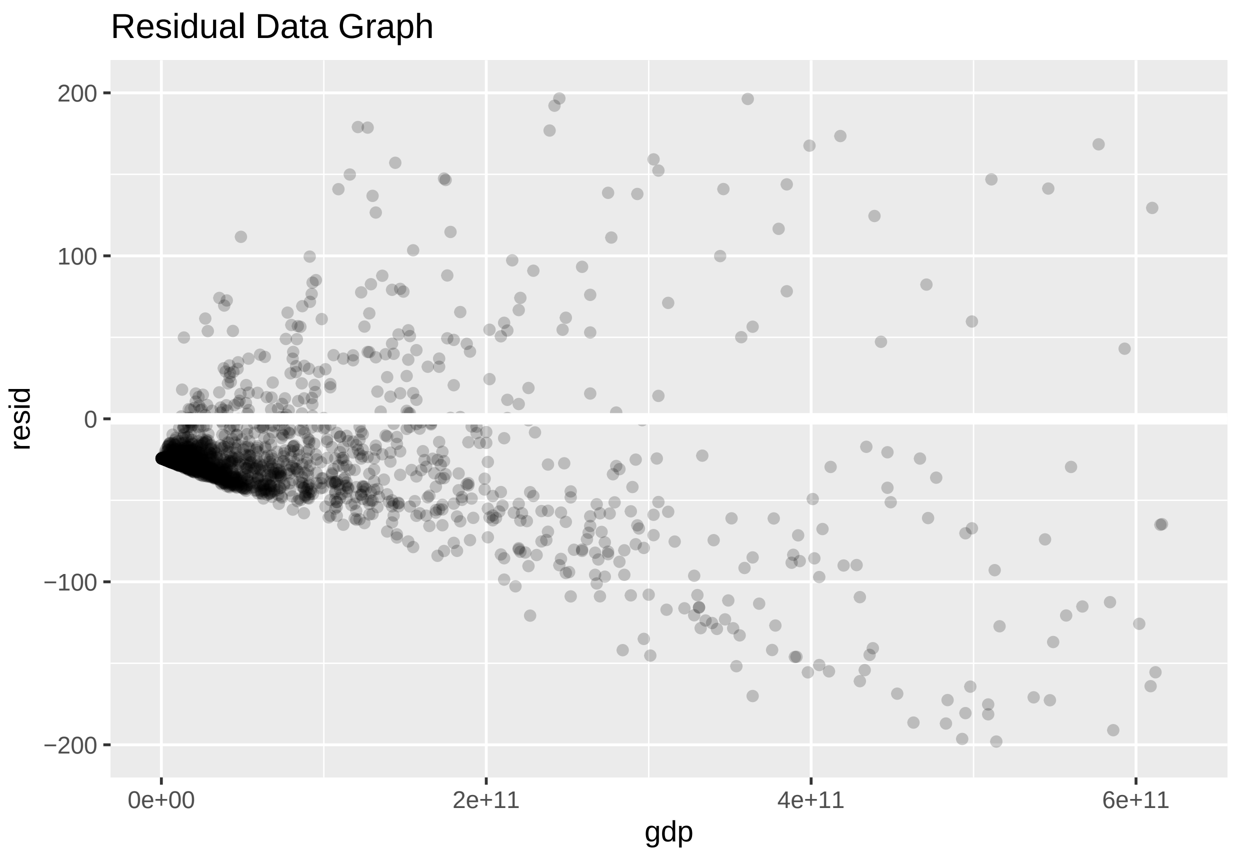
```
resid <- add_residuals(test, model)
ggplot(data = resid, mapping = aes(x=gdp, y=resid))+
  geom_point(alpha = 0.2)+
  geom_ref_line(h=0)+
  ggtitle("Residual Data Graph")
```



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

```
## [1] 0.9801988
```

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

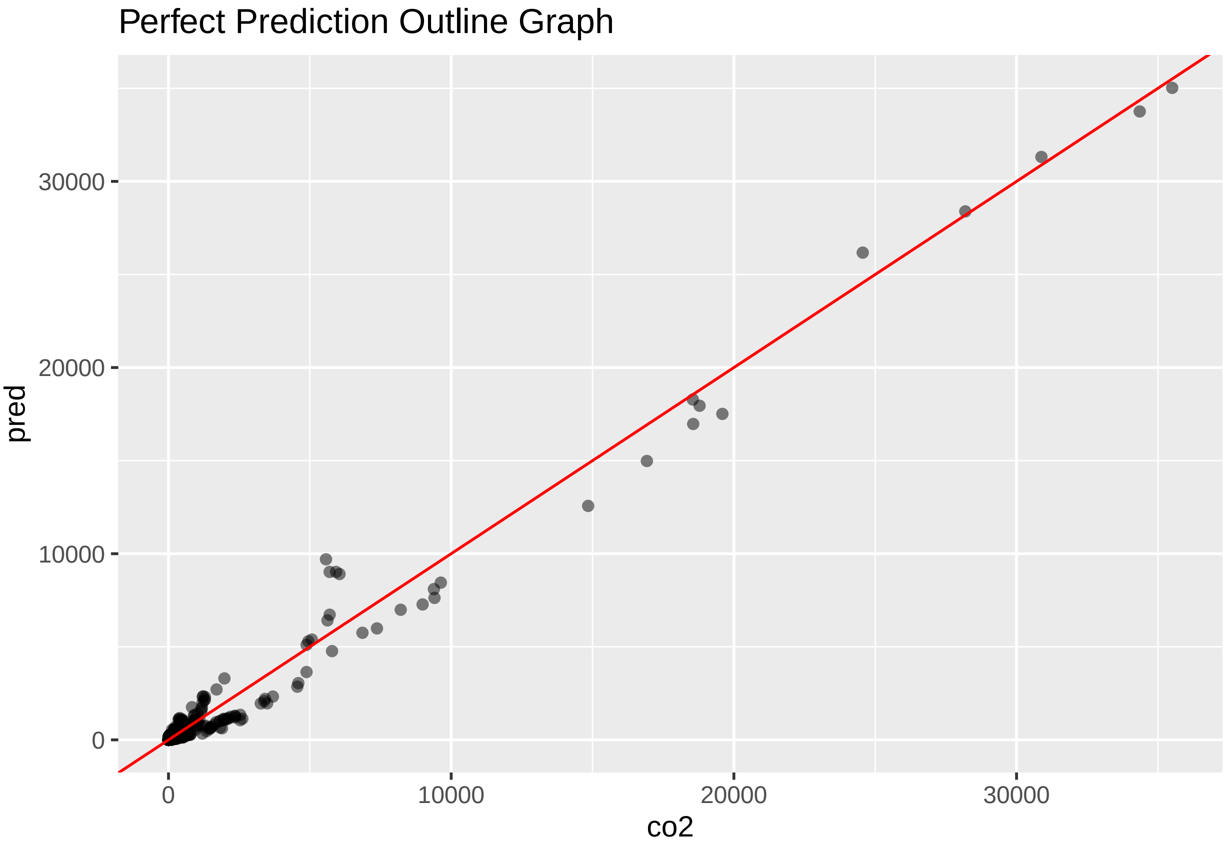
```
## [1] 67.34041
```

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

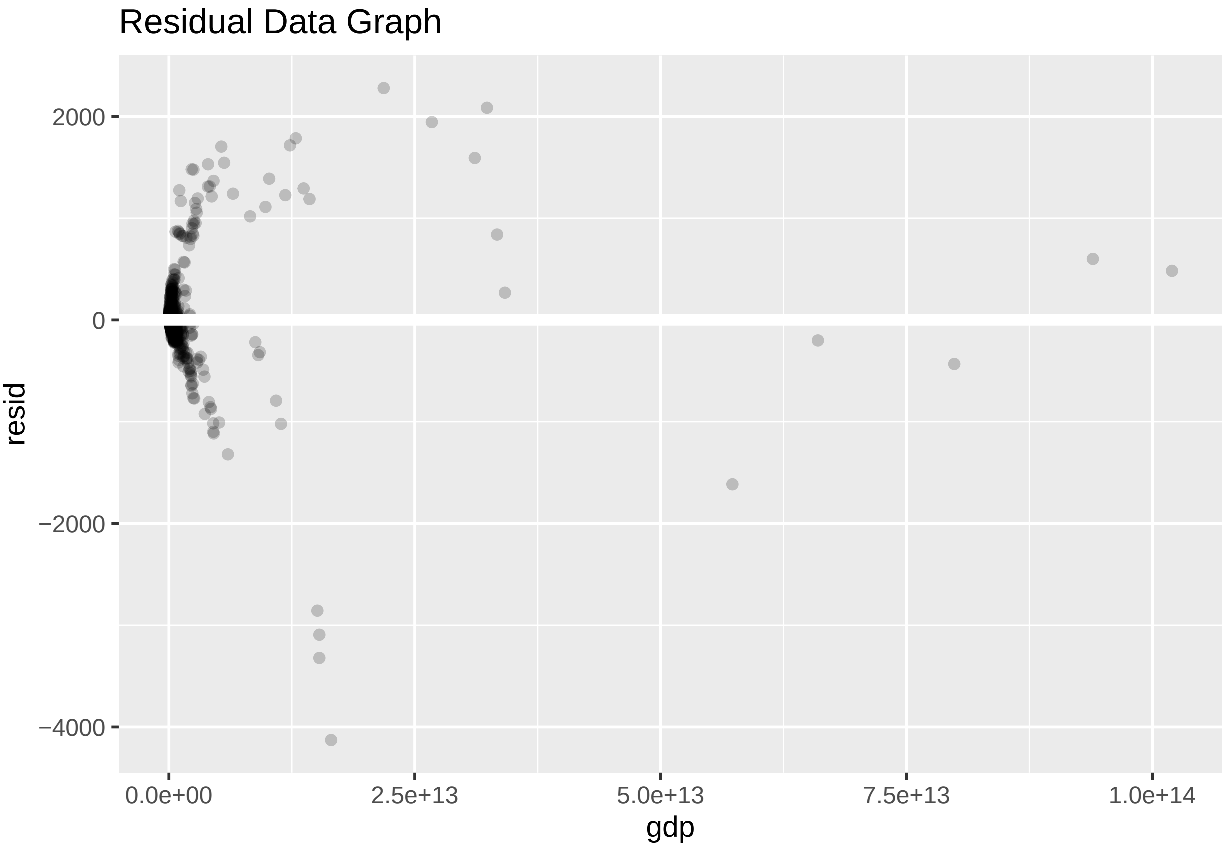
```
## [1] 243.7527
```

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.

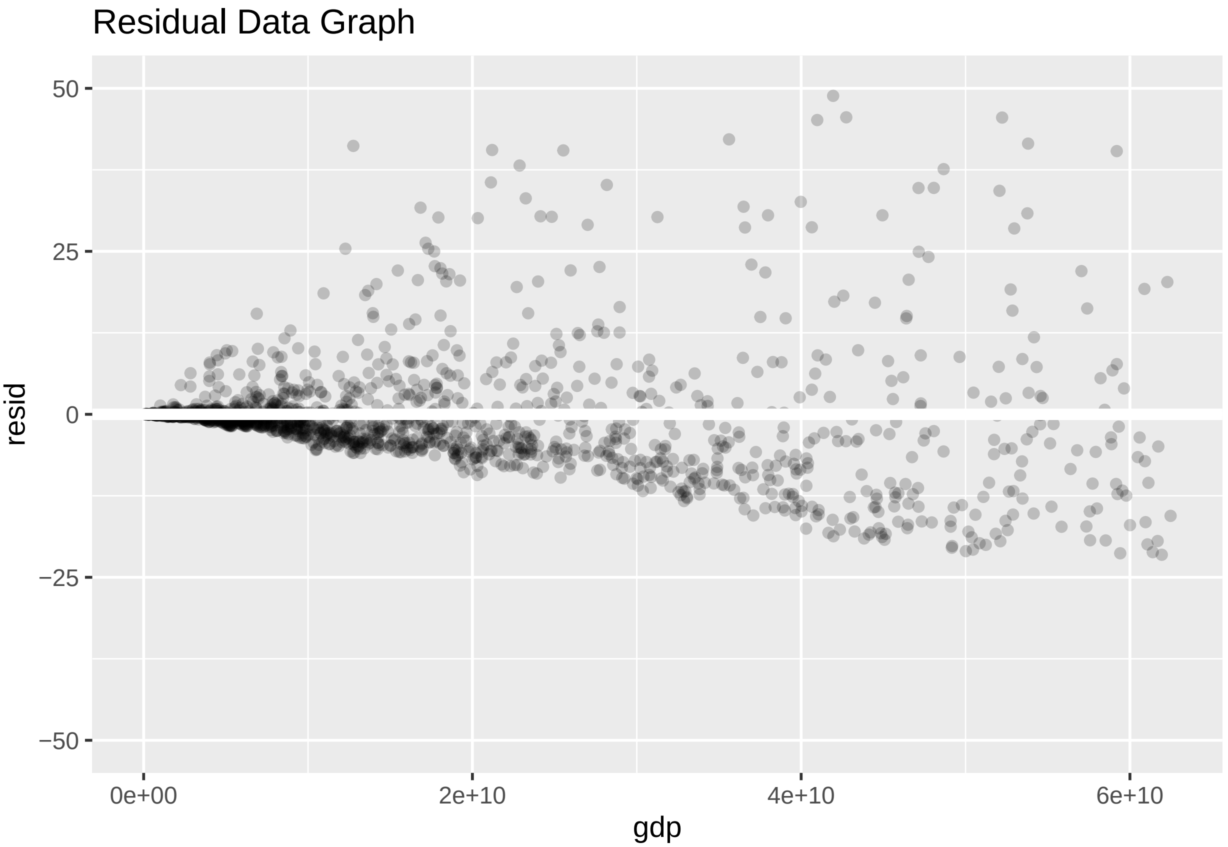
```
ggplot(data = predictions, mapping=(aes(x = co2, y = pred)))+
  geom_point(alpha = 0.5)+
  geom_abline(intercept = 0, slope = 1, color = "red")+
  ggtitle("Perfect Prediction Outline Graph")
```



```
resid <- add_residuals(test, 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")
```

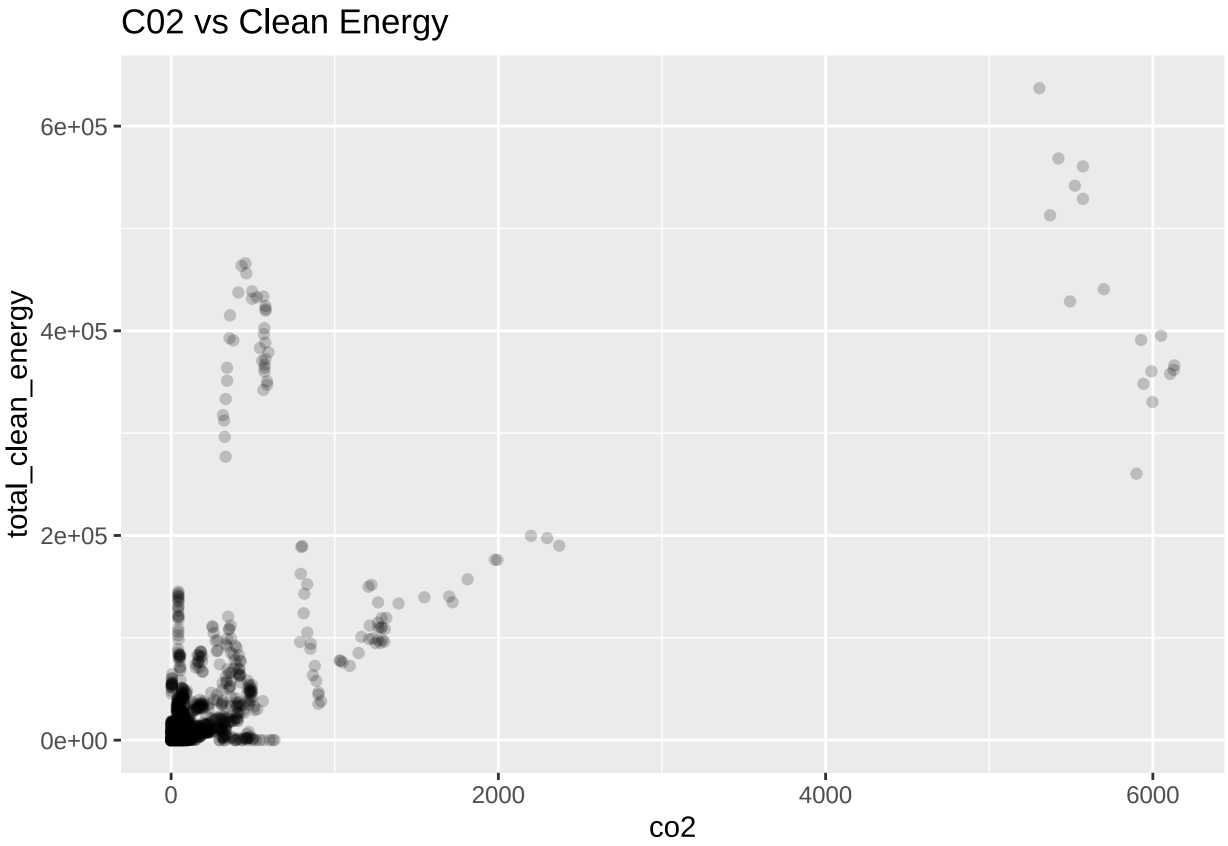
```
## Joining, by = c("country", "year")
```

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.

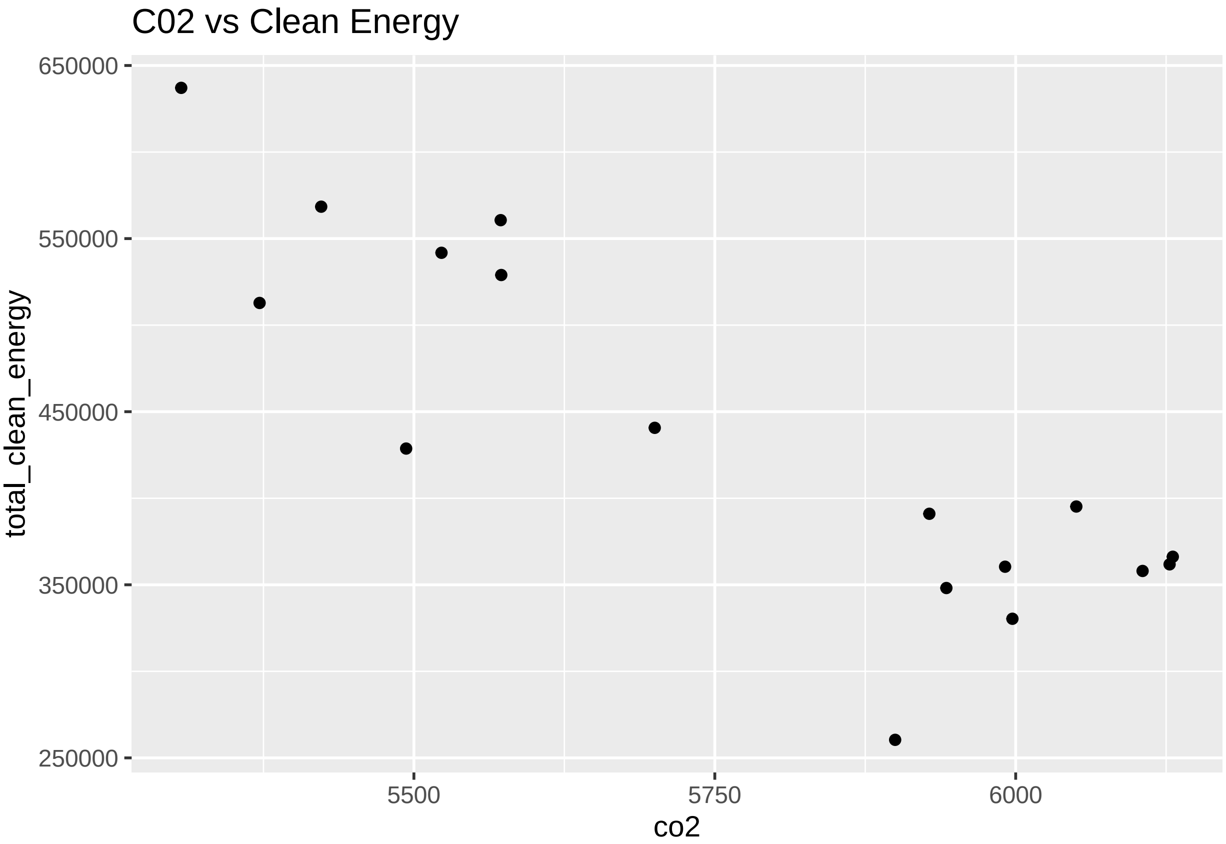
Filtered out China and World as the data for China is highly debated in the data set, and World isn't going to help since it's has such extreme numbers and wont add much to the model.

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_e
  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.

```
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, ]

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] 0.9670262
```

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

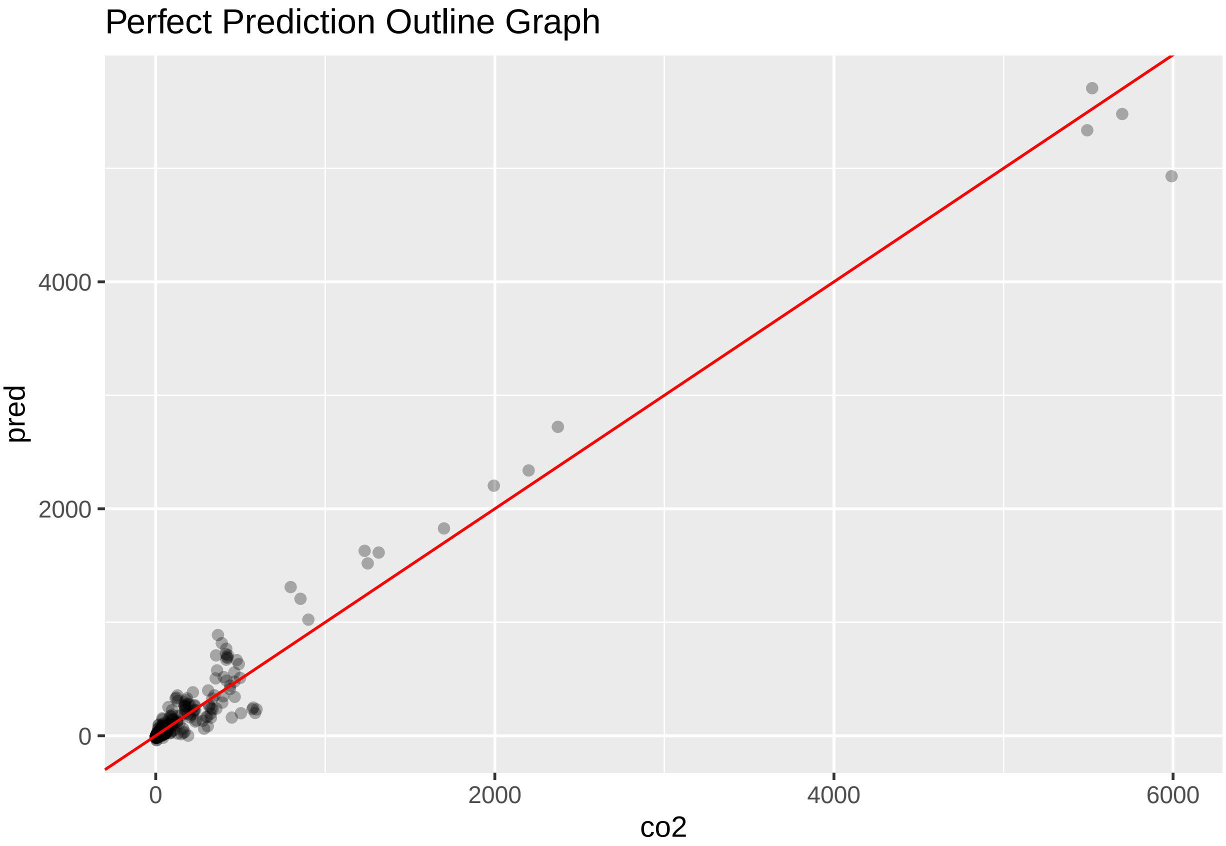
```
## [1] 45.97456
```

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

```
## [1] 103.0282
```

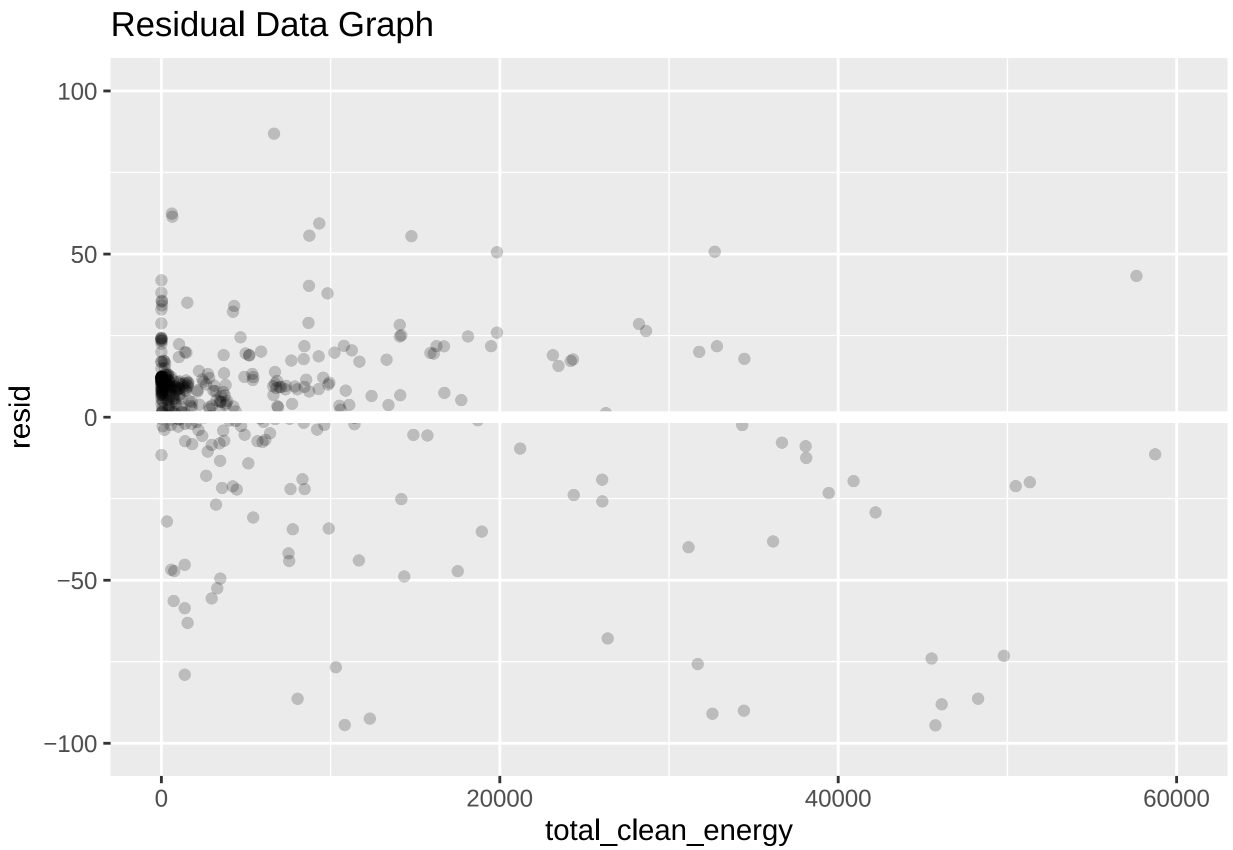
From the R2 value, this seems to have made a very interesting result and actually increased the value of the regular linear model test I first did with just co2 and gdp. This I did not expect as the graph for co2 and clean energy seems to show both going up at similar rates.

```
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")
```

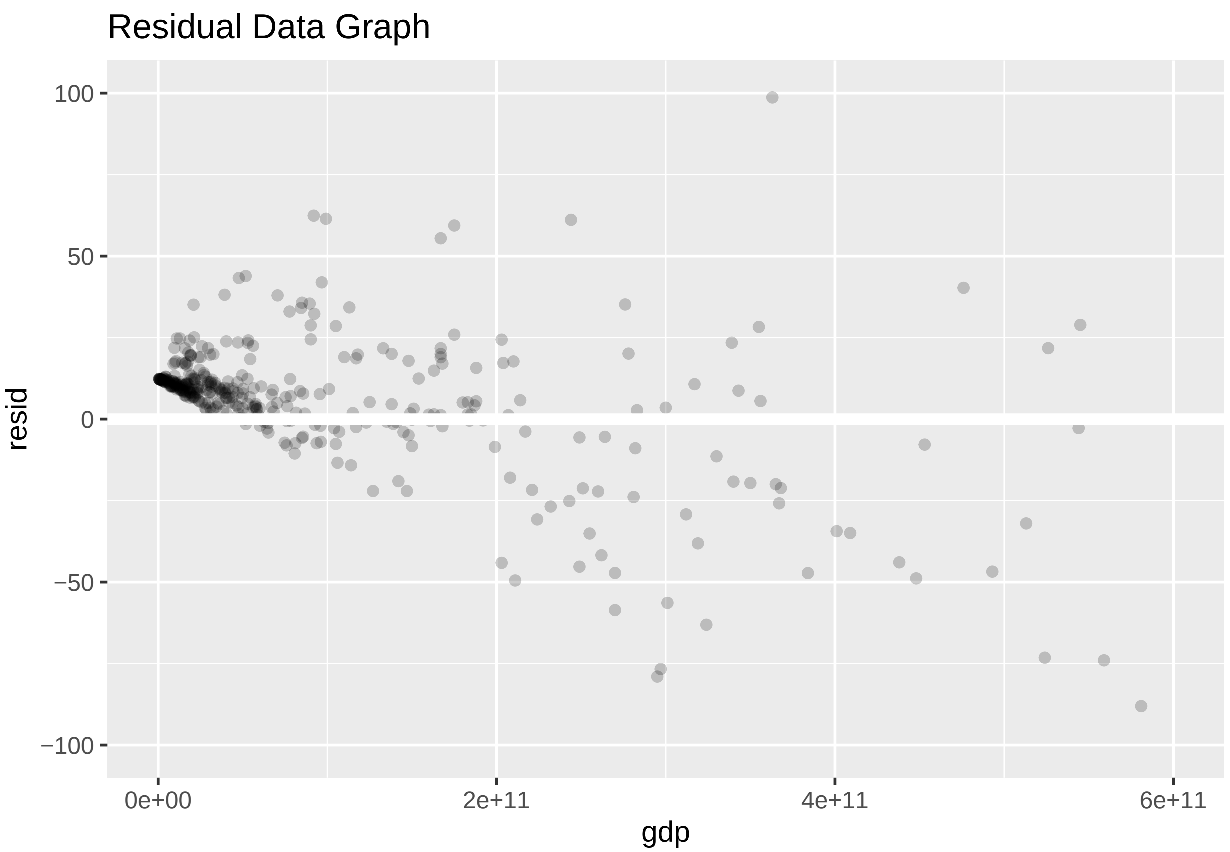


The predictions seem very clumped around the perfect prediction mid-line. This is a very good sign for the model accuracy.

```
resid <- add_residuals(test, 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(test, 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 it does seem to have found a pattern with both variables in this case. The one big thing is that MAE and RMSE are both lower than before making me think this model might be a tad better with error values.

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. Though switching to the polynomial equation really helped match the curve with a much beter result than before.

Grabbing the electrical grid information seemed to make a difference as well with the R2 being very similar and error values being lower.

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.