Actuarial Valuation of Pullanta Carbon Credits

Team UT Boundless

Faculty Advisor: Ian Genno

Changhao Liu
Zhiya Lou
Songqi Wang
Pei Yang
Luolin Chen

UNIVERSITY OF TORONTO

SOCIETY OF ACTUARIES

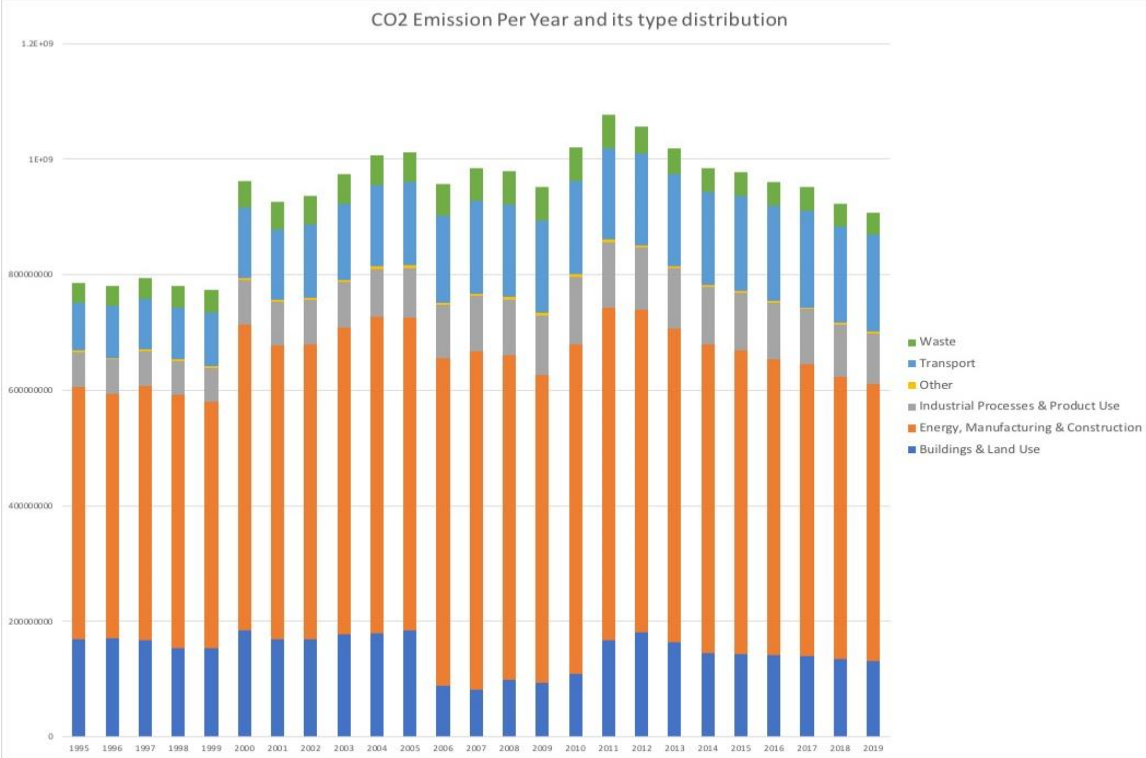
Table of Contents

1.Exectuive Summary

In order to reduce CO_2 emission, we are asked to help Pullanta to design a carbon credit instrument. Carbon credit is a government-issued permit that gives the holder, such as a company, the legal right to emit one metric tonne (one thousand kilograms) of carbon dioxide or an equivalent amount of other greenhouse gases. In order to reduce the carbon emissions of Pullanta, a specific program is necessary. First, figure out the composition of carbon emission among the different sectors and the relationship between carbon emission. Secondly, design a couple of carbon credit financial instruments. Finally, estimate the revenue Pullanta government can gain from this program.

2.Background

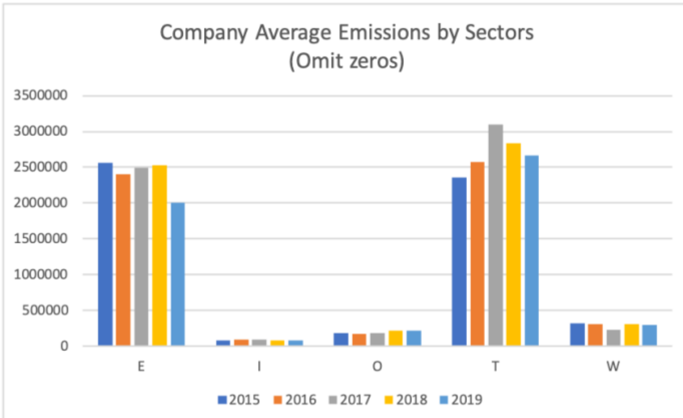
Pullanta is an economically developed country whose government recently set a goal of reducing carbon emissions to 25% below the 2018 level by the end of the year 2030.



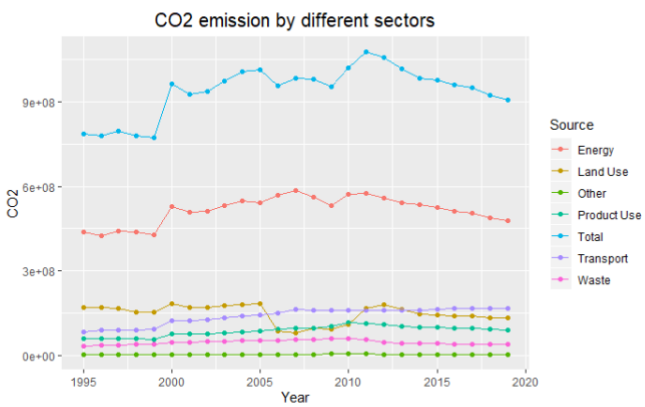
Graph 2-1

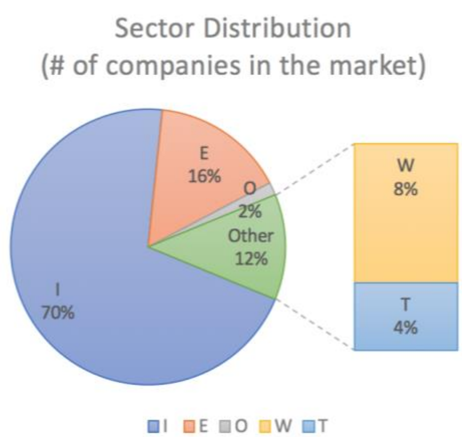
In our data (Graph 2-1), the carbon emission amount fluctuates along with the time and the carbon emission reaches the peak in 2011 with an obvious downward trend in the following years which implies that it is reasonable for us to expect a carbon credit program to further control the carbon emissions.

Graph 2-2 shows the company average of carbon emission for each sector from 2015 to 2019 omitted zero entries of the Company Data. We can see the sectors take most weights are Energy and Transport and they both show decreasing trends in later years.



Graph 2-2

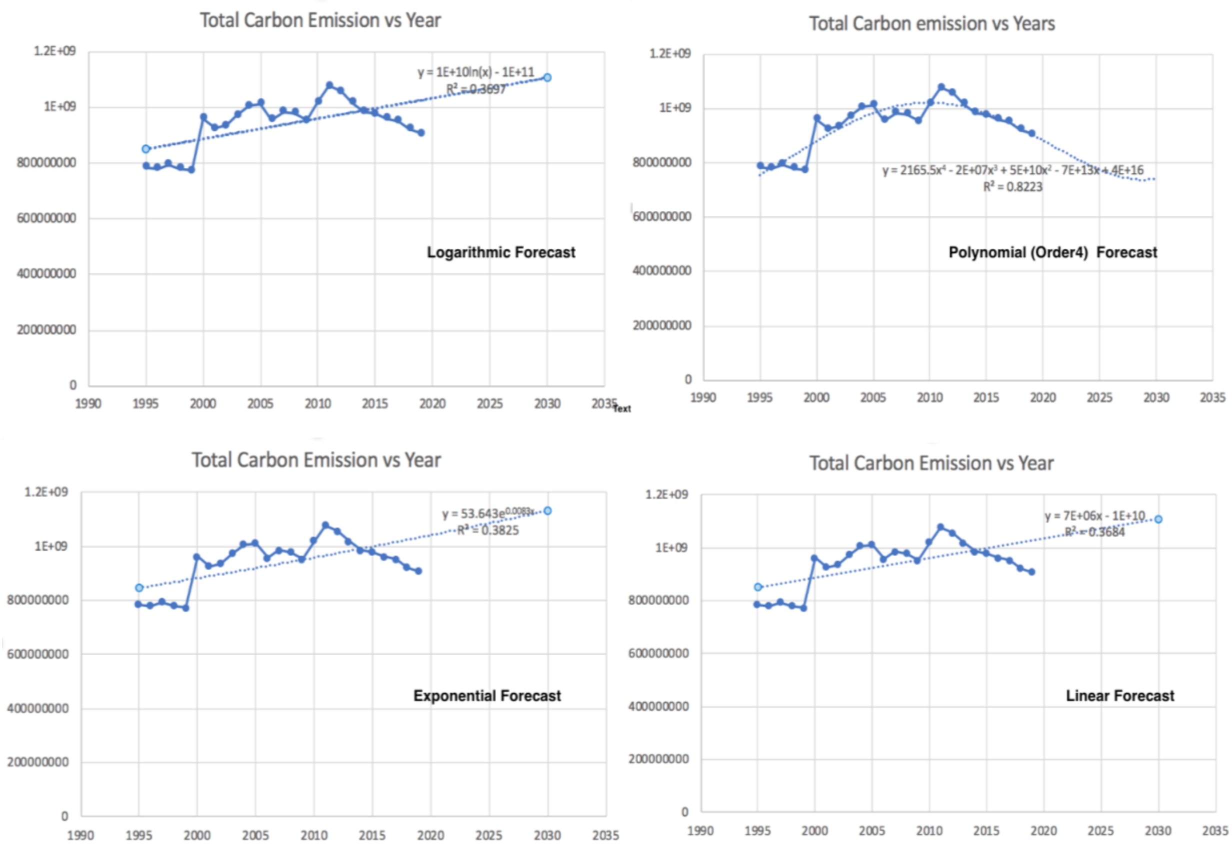




Graph 2-3

Graph 2-4

In terms of future projection, we used four various models to forecast the next 11 year's Total Carbon Emission until 2030.



Graph 2-5: Total Carbon Emissions Forecast by different models

3. Data Limitation

limitation 1: zero entries of company data

From the Pullanta's individual company dataset, we see there are zero entries of individual company's emissions in different years which caused by various reasons (i.e. the company didn't report or the company actually emitted zero units of CO_2). However, our analysis for the average CO_2 emission only considers the non-zero entries which generate the average emissions of the non-zero CO_2 emission companies.

limitation 2: estimated data of 2019

According to the Pullanta government, the data for 2019 are estimated values. However, there are increasing number of zeros entries for individual

limitation 3: inconsistency of data collection criteria prior to 2000

From the graph of total CO_2 emission from 1995 to 2019, we see there is an abnormal increase of CO_2 emission in the year of 2000. We assume it is caused by the different criteria of data collection and pay less attention to the data prior to 2000.

Limitation 4: Lack of data on company's ability to execute the carbon bonds

The probability of the Pullanta companies' violation of the CO_2 emission rule is not given. Therefore, we have to set a reasonable probability that the company will emit excess amount CO_2 after reaching their carbon credit limit.

Limitation 5: Company dataset doesn't have "building sector"

We don't have information on the building sector (B) in the "company dataset". Hence, we assume this source (buildings and land use) is emitted by the government hold company and carbon credit will not be implemented on the "Building Sector".

4. Assumptions

Assumption 1: Stable Interest rate

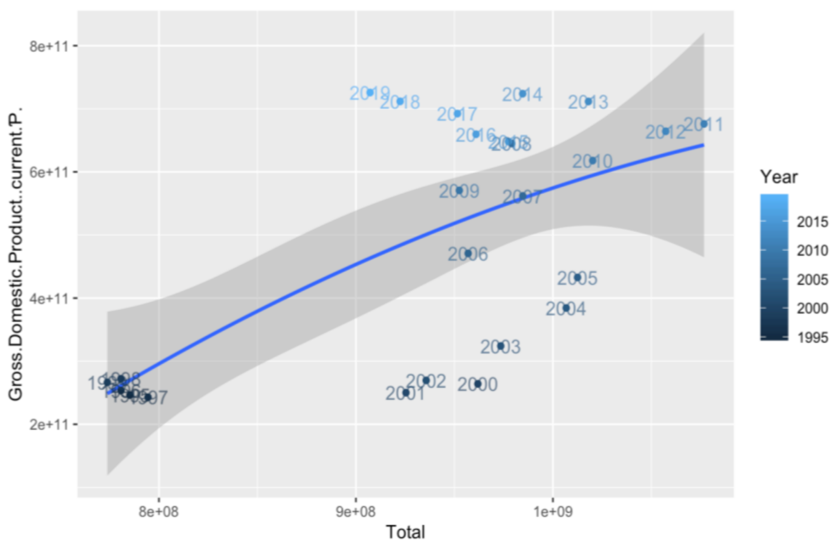
We assume Pullanta has a stable long-term interest rate which is similar to the 10-year US Treasury Bond interest rate. In our analysis, we use the arithmetic mean of three years (2017: 2.33%, 2018: 2.91%, 2019: 2.14%) and get the result of 2.46%

Assumption 2: Companies can only emit carbon dioxide with valid carbon bond after reaching free carbon emission threshold

We assume Pullanta has strict market supervision such that if an entity emitted more than the free amount, it can only purchase carbon credit in secondary market (i.e. From other carbon credit holders) and cannot legally emit CO_2 without valid carbon credit.

Assumption 3: The percent change of carbon dioxide and percent change of GDP have linear relationship

By using ggplot2 function geom_smooth in R with the method of linear regression with 90% confidence interval (grey area), we can observe a linear relationship between Gross Domestic Product with Total Carbon emissions from the raw data. Hence, we assume this relationship exists and discover the deeper analysis in later sections. Note that, however, there is a big cluster around the left corner with all the years before 2000, that may affect our assumption's accuracy. Indeed, there is a huge jump from 1999 to 2000 without any reasonable prediction as we mentioned in data limitation.



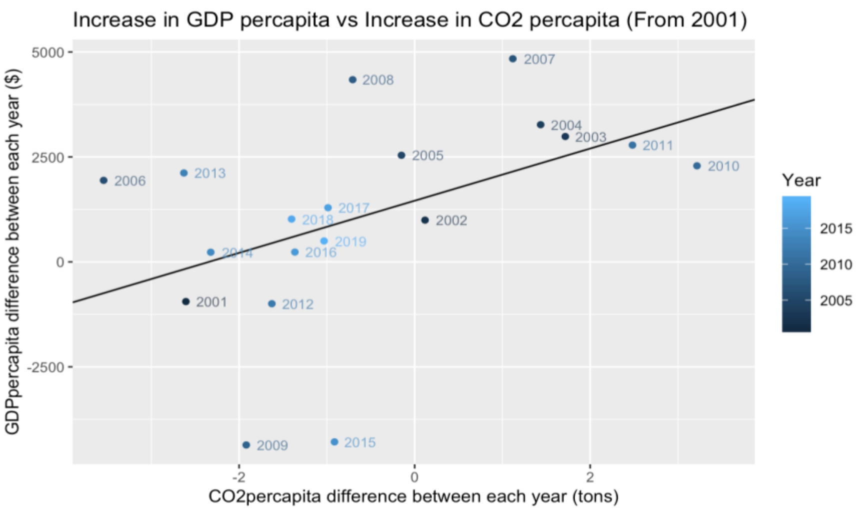
Graph 4-1: Relationship between GDP and total emission

Therefore, we remove the data points before 2000, and continue discovering the linear relationship starting from Year 2001 and on.

5.Design

5.1 methodology

Usually, a bond has coupons and maturity. For the carbon bond, we can simply change the coupon from money to the amount of the CO_2 emission. Therefore, to price the bond, we continue the study with the potential linear relationship between Gross Domestic Product with Total Carbon emissions:



In this Linear Regression, we found the relationship can be written:

$$\Delta GDP \text{ per capita} = 1457.3 + 621.5*(\Delta CO_2 \text{ emission per capita})$$

When the country increases 1 unit (metric tonnes) of CO_2 emission, the GDP will increase by $621.5 correspondingly.

In the macro view, there are still lots of other significant factors that contribute to GDP, so we approximately assume that CO_2 emission has the contribution of 0.1% to GDP in Pullanta:

That is treating \$0.6215 as the Price for 1 metric tonne CO_2. We will use this to price our carbon bond in the later section.

Next, why do companies want to buy our bond instead of periodically purchasing carbon credits? Because we have an additional policy:

The face amount can be considered as a deposit for this carbon credit program.
If the company's actual usage in that year is less than the carbon credit they bought, these extra carbon credits can be converted into bonus money, and the company can redeem the bonus with maturity at the end of year. In this way, companies will have incentives to emit less carbon, and earn more interest from this carbon credit program, instead of buying carbon credits with market volatility.

5.2 implementation

Recall the assumption:

The annual interest rate is 2.46% and assume it will remain the same over the next 10 years.

This is our **Annual Goal** and also long term goal to 2030 that will achieve the 25% reduction in Carbon emission (calculating linearly, reducing 19217522 tonnes each year):

	Carbon Reduction Plan	
	%	estimated CO2 emission
2018	1	922441064.3
2019	0.979167	903223542.1
2020	0.958333	884006020
2021	0.9375	864788497.8
2022	0.916667	845570975.6
2023	0.895833	826353453.5
2024	0.875	807135931.3
2025	0.854167	787918409.1
2026	0.833333	768700886.9
2027	0.8125	749483364.8
2028	0.791667	730265842.6
2029	0.770833	711048320.4
2030	0.75	691830798.2

Bond 1,2 Design:

Large companies always prefer long term management and small companies have lower demand for long term carbon emission plans. So we design 2 types of bond here, In order to control the amount of CO_2 emission:

1) 2 year bond : can only be purchased every 2 years
2) 5 year bond: can only be purchased in 2020 and 2025.

The government can set the price base on the Price for 1 metric tonne CO_2. If the price of 1 metric tonne CO_2 is $0.6215, and the 1 annual yield is 2.46%. Therefore, if the 5-year carbon bond face value is $1000, redeem at par, then the price is $3776.29. And the 2-year carbon is $2151.32.

30% of the total amount would be 5-year, 70% would be 2-year bond. The annual coupon of bonds would be 1000 metric tonnes of CO_2 emission.

In the range of our annual target, based on the previous history CO_2 carbon emission from company data, we allow 80% of their amount to be free carbon credit if the company is implementing this bond program.

For the rest of the amount, the company has to purchase the bond to get the permission of CO_2 emission. If the company is planning for a higher CO_2 emission than the previous year, then the company has to pay the higher price for the extra carbon bond.

Based on our projection, we would sell *53061* 5-year carbon bonds in 2020 and *47189* in 2025.Then for the rest amount, we would sell *123808* 2 -year carbon bonds in 2020, *118328* in 2022, *112847* in 2024, *107366* in 2026, *101885.4* in 2028.

Bond amount after company limit

co2 <dbl>	Source <chr>	Year <int>	bond <dbl>	2y <dbl>	5y <dbl>
884349576	Total	2020	176869.9	123808.9	53060.97
864774960	Total	2021	172955.0	123808.9	53060.97
845200345	Total	2022	169040.1	118328.0	53060.97
825625730	Total	2023	165125.1	118328.0	53060.97
806051115	Total	2024	161210.2	112847.2	53060.97
786476500	Total	2025	157295.3	112847.2	47188.59
766901885	Total	2026	153380.4	107366.3	47188.59
747327269	Total	2027	149465.5	107366.3	47188.59
727752654	Total	2028	145550.5	101885.4	47188.59
708178039	Total	2029	141635.6	101885.4	47188.59

The present value of 2-year bonds revenue would be 617992030.

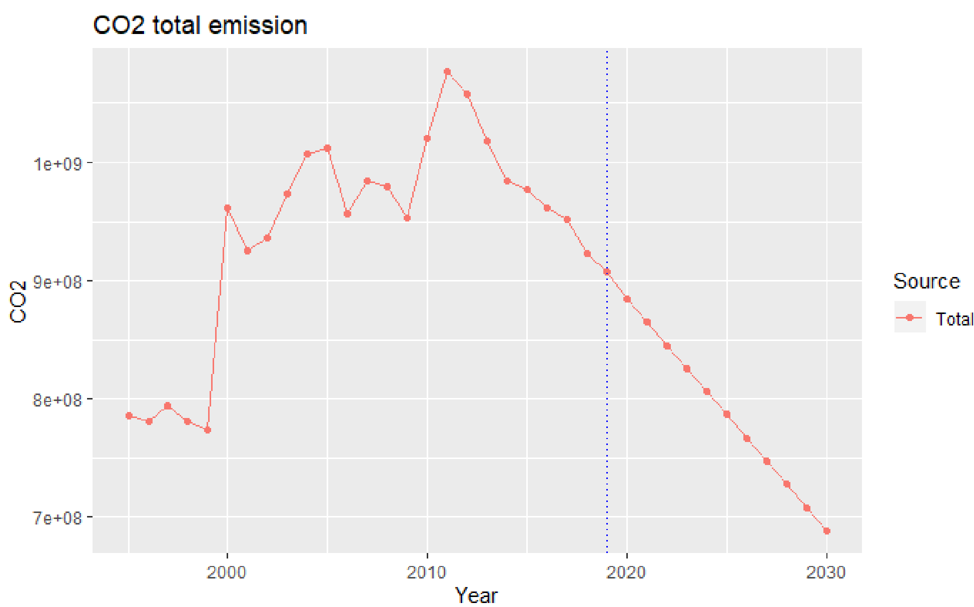
The present value of 5-year bonds revenue would be 274184665.

The total revenue would be 892176696.

RESULT AFTER IMPLEMENTATION:

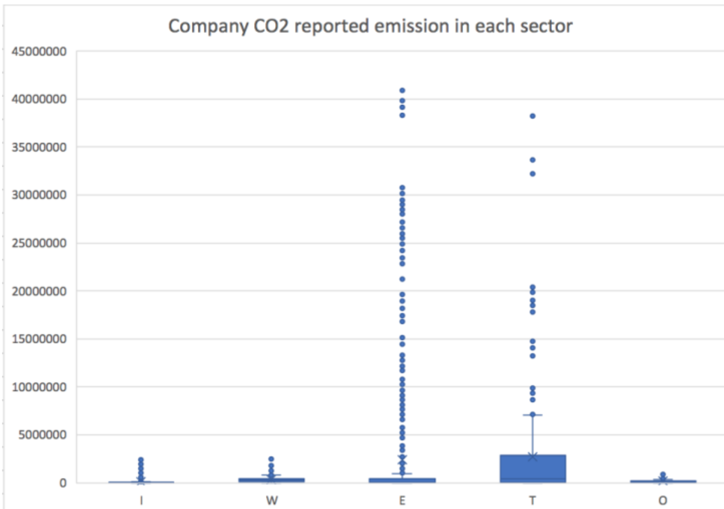
We can see under this control, CO_2 emission is going downwards consistently, and it achieves our goal successfully. There is no other new carbon bond to be issued in any other years. Under this policy, the total amount of CO_2 emission in each year is easily limited, therefore we allow the secondary market to exist. So if a company has to increase its carbon emission, it can adjust through the market.

This is the graph that visualizes our implementation result with decreasing trend until our target.



Bond 3,4 Design

We will divide companies according to their annual carbon emission level. First, let's look at their carbon emission distribution in each sector from Company data after



removing all the "0": Here are some statistics:

Company's CO2 Emission Distribution by Sector

Sector	E	I	W	O	T
MIN	7	5	8	501	7
20th	6208	5988.4	101737	23229.6	48734
50th	32082	13117	235583	45499	442626
75th	391219	36880	395269	160618	2846997
AVG	2394827	86864	311750	194315	2696176
MAX	40893269	2619734	2444295	1271326	38242274

Note: There are lots of large outliers in Energy and Transportation sector, so we should be aware that the most of companies in those sectors are emitting at the normal level, regulating those high emission companies is in the first place.

We set the Starting point as 20th percentile of each sector, that is, for each company in the corresponding sector will have some free carbon credit that is not counting towards their total carbon emission. For example, each Energy Company doesn't need to pay 6208 tonnes of carbon per year. So that small companies in that sector will not take too much burden to afford this cost, while the big companies also have these amount of free carbon credit, creating a fair market environment.

We are calculating the percentile of each sector because we want to find a fixed unit of CO_2 for each bond we are selling in the market to that sector. If we are using the same initial amount of CO_2, those small companies with low emission will have extra carbon credit left. To make the policy more efficient, we will use the **20th percentile** in each sector as the amount of **1 carbon credit for** the corresponding sector. We will offer two types:

3) 1 Year Short term Bond: buying 1 carbon credit with corresponding sector per bond in that year, with face value 1000 and redeem at par. (It is designed more expensive than the 10 year one)
4) 10 Year Long term bond: giving 1 carbon credit each year, with face value 1000 and redeem at par.

The price of bonds is based on the following factors, the price of 1 carbon credit is based on the price of 1 ton of CO_2, which is $0.6125. And the annual yield is 2.46%.

3) Bond Price in each Sector $P				
E	I	O	W	T
3502.83329	3395.92106	49926.5516	11774.4378	24168.9665

And then, dividing it with our annual goal.

This is the table we designed for the number of available carbon credits each year vary by sector:

number of carbon credit issue	B	E	I	O	T	W	total
2018							
2019							
2020		87311.5292	18587.8675	141.700007	3763.95479	431.90043	110236.952
2021		85052.0232	18345.625	135.974379	3722.52108	416.295198	107672.439
2022		82811.7526	18094.7691	130.38953	3678.93773	401.02079	105116.87
2023		80590.4596	17835.4155	124.943571	3633.23354	386.07277	102570.125
2024		78387.891	17567.6774	119.634651	3585.43682	371.446784	100032.087
2025		76203.7981	17291.6662	114.460949	3535.57536	357.138555	97502.6392
2026		74037.9364	17007.4914	109.420676	3483.67646	343.143878	94981.6689
2027		71890.066	16715.2605	104.512076	3429.76697	329.458626	92469.0641
2028		69759.9509	16415.079	99.7334212	3373.87323	316.078742	89964.7153
2029		67647.3594	16107.0508	95.0830168	3316.02114	303.000241	87468.5146
2030		65552.0637	15791.2778	90.5591955	3256.23617	290.219204	84980.3561

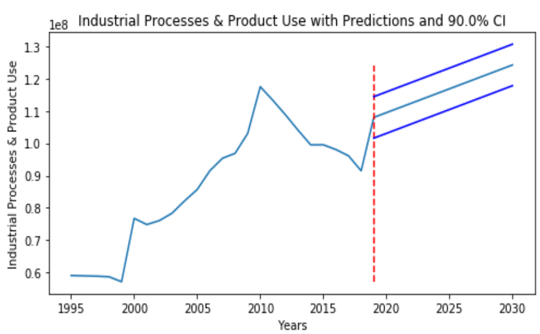
(Assume Sector Buildings is owned by government because it doesn't contain in Company Data. Hence, we will not issue bond to Sector Building)

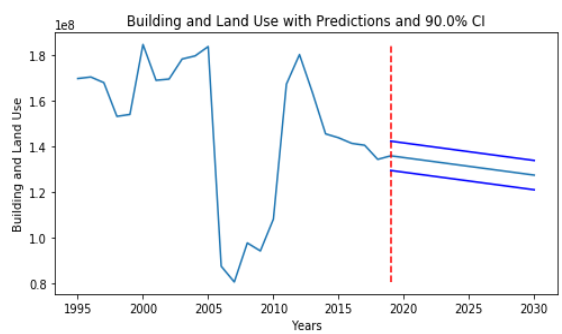
RESULT AFTER IMPLEMENTATION:

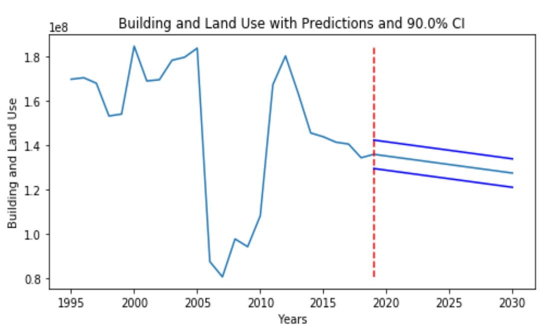
Assumptions to consider:

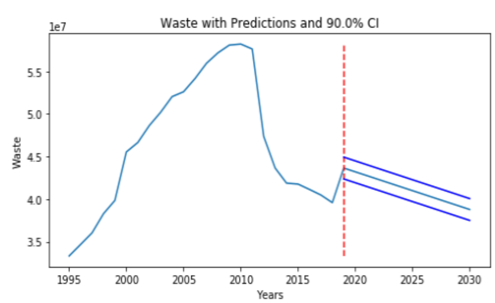
With different numbers of companies in each sector, we assume that each company will have a Normal distributed total emission, that is centered at their emission allowance with a fixed fluctuation. It is reasonable to make such assumption since it is rational for companies to optimize their emission credit, but errors can occur, which leads to some variance away from their target allowance.

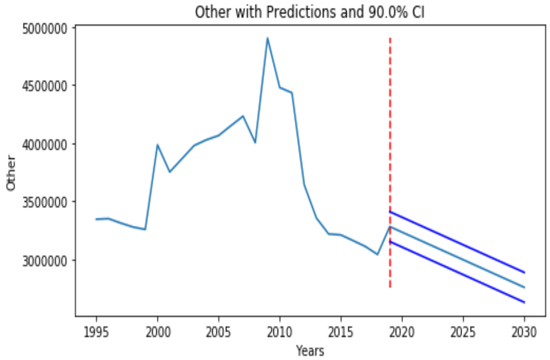
Here is our projections after implementing bond 3 and 4 in each sector, the darker blue line above and below is the 90% Confidence Interval we constructed based on the assumption, indicating that Pullanta expects with 90% certainty to result in aggregate carbon emission staying within the goals.

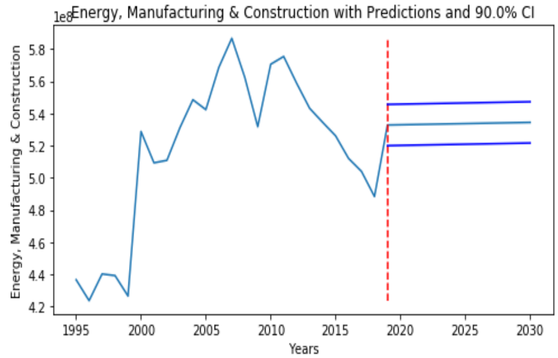









Other with Predictions and 90.0% CI
5000000
4500000
4000000
3500000
3000000
Other
1995
2000
2005
2010
2015
2020
2025
2030
Years


Energy, Manufacturing & Construction with Predictions and 90.0% CI
1e8
5.8
5.6
5.4
5.2
5.0
4.8
4.6
4.4
4.2
Energy, Manufacturing & Construction
1995
2000
2005
2010
2015
2020
2025
2030
Years

5.3 REVENUE AND EXPENSE

Revenue = Quantity x Price

Since we already computed the price and the number of carbon credit issued in each year and each sector, simply multiply them with discount factor correspondingly would be our expectation revenue from the program.

Aggregate Emission:

	generated from python code (please refer our python code in the appendix section)						
		B	E	I	O	T	W
expected weight over time for all sectors	2019	13.58%	53.22%	10.78%	0.33%	17.73%	4.36%
	2020	13.46%	53.06%	10.90%	0.32%	17.96%	4.30%
	2021	13.34%	52.91%	11.01%	0.32%	18.18%	4.24%
	2022	13.22%	52.76%	11.12%	0.31%	18.40%	4.19%
	2023	13.10%	52.61%	11.23%	0.31%	18.62%	4.13%
	2024	12.99%	52.46%	11.34%	0.30%	18.84%	4.07%
	2025	12.87%	52.31%	11.45%	0.29%	19.05%	4.02%
	2026	12.76%	52.16%	11.56%	0.29%	19.27%	3.96%
	2027	12.64%	52.02%	11.67%	0.28%	19.48%	3.91%
	2028	12.53%	51.87%	11.77%	0.28%	19.69%	3.85%
	2029	12.42%	51.73%	11.88%	0.27%	19.91%	3.80%
	2030	12.30%	51.58%	11.99%	0.27%	20.12%	3.74%

		E	I	O	T	W
expected weight without B	2019	61.58%	12.48%	0.38%	20.52%	5.04%
	2020	61.32%	12.59%	0.37%	20.75%	4.97%
	2021	61.06%	12.70%	0.37%	20.98%	4.90%
	2022	60.80%	12.81%	0.36%	21.20%	4.82%
	2023	60.54%	12.92%	0.35%	21.43%	4.75%
	2024	60.29%	13.03%	0.34%	21.65%	4.68%
	2025	60.04%	13.14%	0.34%	21.87%	4.61%
	2026	59.79%	13.25%	0.33%	22.09%	4.54%
	2027	59.55%	13.36%	0.32%	22.30%	4.47%
	2028	59.30%	13.46%	0.32%	22.52%	4.40%
	2029	59.06%	13.57%	0.31%	22.73%	4.34%
	2030	58.82%	13.67%	0.30%	22.94%	4.27%

Graph 5.3-1

Based on our pricing strategy, Pullanta should issue different amounts of carbon credits to different sectors because the CO_2 emission trends are different (graph.Moreover, we also set the threshold of free carbon emissions as 20th percentile of various sectors in order to make sure that the small companies won't purchase too many carbon credits that exceed their needs and also can reduce their financial stress.

Social Cost of Carbon

Based on the assumption of the linear relationship of CO_2 and GDP, Pullenta's GDP increases Ᵽ621.5 by increasing 1 metric tonne of CO_2 emission. Therefore, by limiting CO_2, Pullanta also gives up part of its GDP which needs to be compensated by the carbon credit. Considering the transition problem might occur at the beginning period of carbon credit, we assume the price of the carbon credit should be cheaper at the beginning and gets more expensive at later time.

Secondary Market

The small companies (lower than 20th percentile) are more likely to trade their excess amount of carbon credit to the big companies because big companies take more time to make adjustment to the new carbon bond program. We also recommend Pullanta allow free trading of carbon credit on secondary market with reasonable taxes. However, because most companies will use up all their carbon credits due to transition issues in the early stage, the supply of carbon credit will be low which will result in a much higher price of carbon credit in the secondary market and it can be viewed as a kind of penalty to a large extent.

Consequences

If the companies exceed carbon emissions limits, the government can deduct the face amount of the bonds which can be seen as a penalty. In this situation, the revenue of the government increases but it is harder for the government to meet its goal of getting a carbon emission reduction of 25% by 2030 if there are a lot of companies exceeding the carbon emissions limits. Also, the extra amount of emission will damage the environment which cannot be eliminated in a short period of time.

6. Risk Analysis

Social Transition Risk:

After implementing the carbon credit, entities will carry more burdens than before. Their production may be limited by total emission limit and they also face an increase of expenses due to cost of carbon credits. Therefore, at the beginning of the carbon credit program, we issue a similar amount of units to the units emitted last year so that the companies can get used to it year by year instead of being charged a large amount of money immediately. They can adjust to new technologies with less carbon emission.

Considering that some of the smaller companies may not be able to afford a sudden cost from carbon credit, we set up a certain amount for each sector to allow companies to emit without purchasing carbon credits. Therefore, for the companies with extreme low emissions, they do not need to pay anything.

Due to the reduction of carbon emission, the production of the manufactures may also decrease. Then some of the suppliers may provide less commodities for the consumers to buy. With unchanged demand, the price of those commodities will increase which resulting in higher expenses for consumers.

After implementing the carbon credit program, the government needs to hire more employees to assist the implementation which results in extra expenses. However, with the help of this program, the government can get funding from the carbon credits and use it to further help dealing with climate change problems.

The other approach to reduce carbon emission is to give rewards or allowances to the companies in renewable energy area and the companies with less carbon emission than before. This approach stimulates the development of renewable energy and reduction of carbon emission without causing extra expenses for the companies. Therefore, the economic growth may not be influenced. However, the rewards and allowances become a huge burden for the government.

7. Appendix

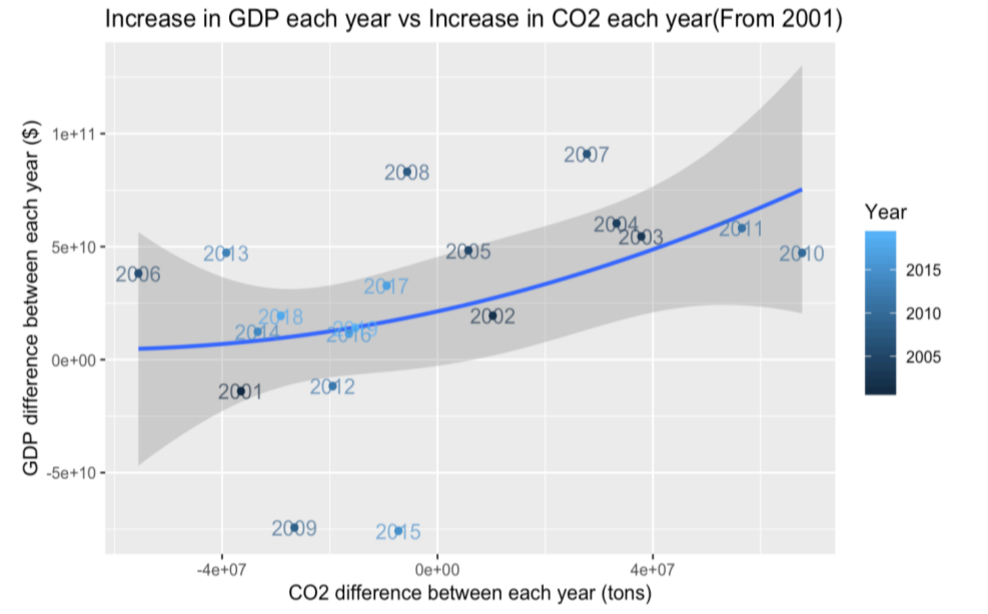
Appendix A: Supporting Calculation on Linear Model

The amount of Carbon emission each year may be affected by various factors. With fast growth in recent decades, Pullanta's economy and market structure must have significant changes. So, such comparing between different intervals of time period would be inefficient (e.g. compare the carbon emission with 2001 and 2011, obviously in 2011 with better technology, the way and the amount of carbon emission is totally different as a decade ago). Therefore, we want to "standardize" this comparison by analyzing the changes in carbon emission each year.

That is, Δ CO_2 in year N = (total CO_2 in year N) - (total CO_2 in year N-1)

For the same reason, we also calculate the changes in GDP each year, ΔGDP.

And Here is the linear relationship after re-computation:



graph 5.1-1

Into next step, we also want to consider the other factors that make up CO_2 emission. From the dataset, Population is considered as an important component that affects CO_2 emission, since growing population will result a higher emission.

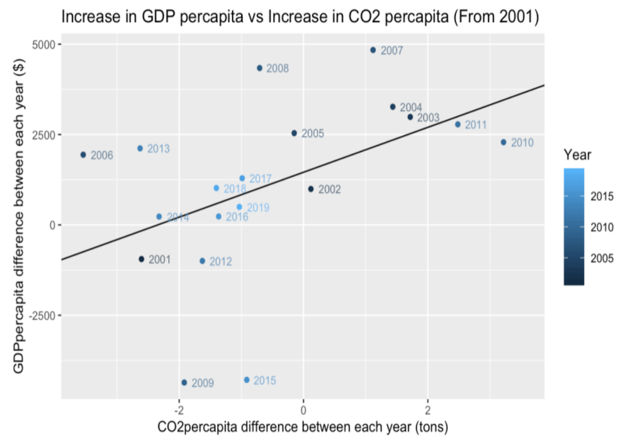
Dividing GDP and CO_2 emission by the total population in that year, we can get the increase in GDP and CO_2 per capita. In this situation, we limit the variation of CO_2 emission possibly caused by population growth. Keeping this as a fixed variable, we can get more accurate predictions for the relationship between GDP and CO_2, and pricing carbon emission.

```
Call:
lm(formula = Yeardivgdp ~ cpcdiv, data = gdpco1)

Residuals:
    Min      1Q  Median      3Q     Max 
-5174.5  -660.2   211.3  1048.0  3327.8 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)  
(Intercept)   1457.3      544.0   2.679   0.0159 *
cpcdiv         621.5      287.6   2.161   0.0453 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2255 on 17 degrees of freedom
Multiple R-squared:  0.2155,	Adjusted R-squared:  0.1693 
F-statistic: 4.669 on 1 and 17 DF,  p-value: 0.04527
```

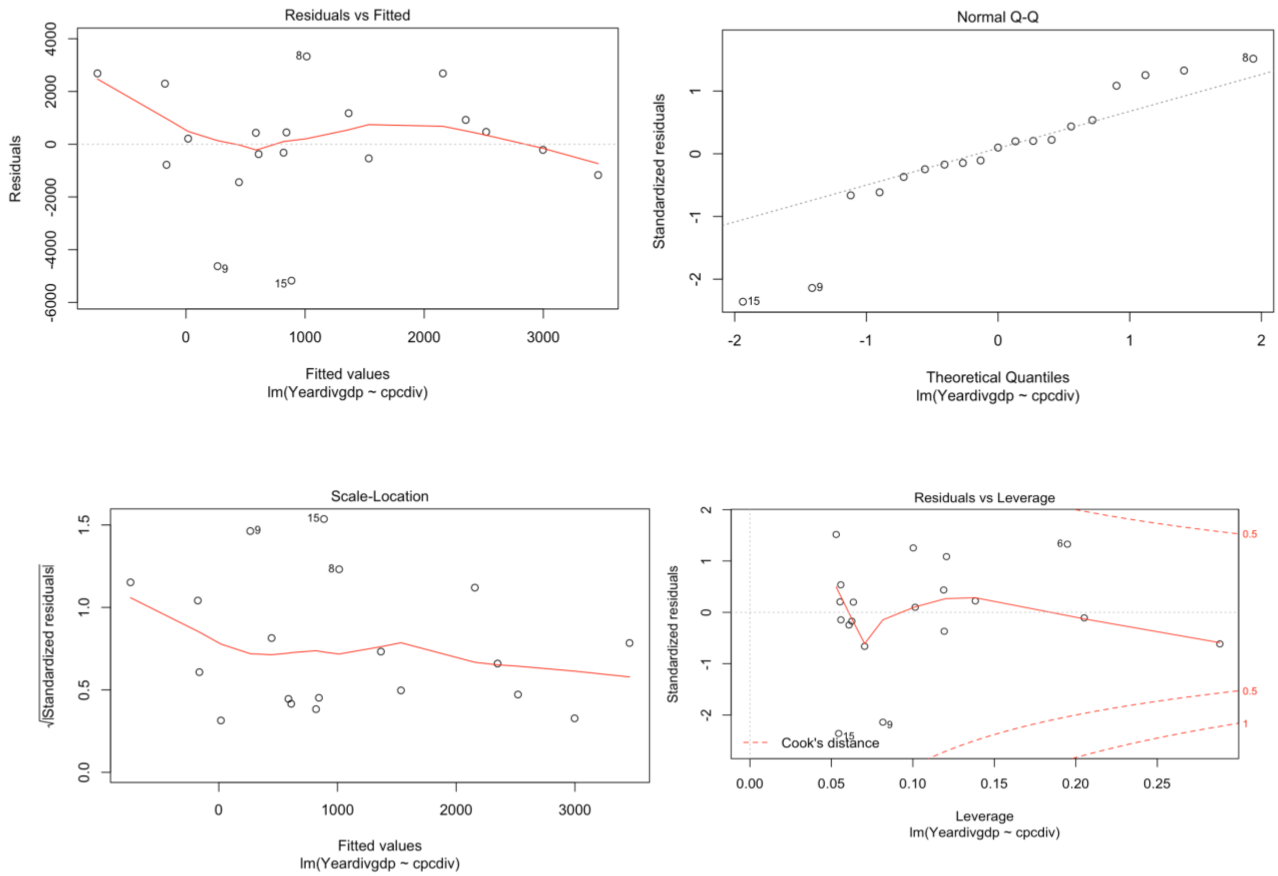


graph 5.1-2: result of linear regression

In this Linear Regression, we found the relationship can be written:

$$\Delta \text{GDP per capita} = 1457.3 + 621.5*(\Delta CO_2 \text{ emission per capita})$$

To verify this is a strong relationship, we checked by using hypothesis testing with null hypothesis of they don't have linear relationship (slope = 0). It is notable that we get a very small p-value (<0.05), providing strong evidence that this linear relationship is valid. Also, there are no extreme leverage points and residuals. (Shown in below graphs)



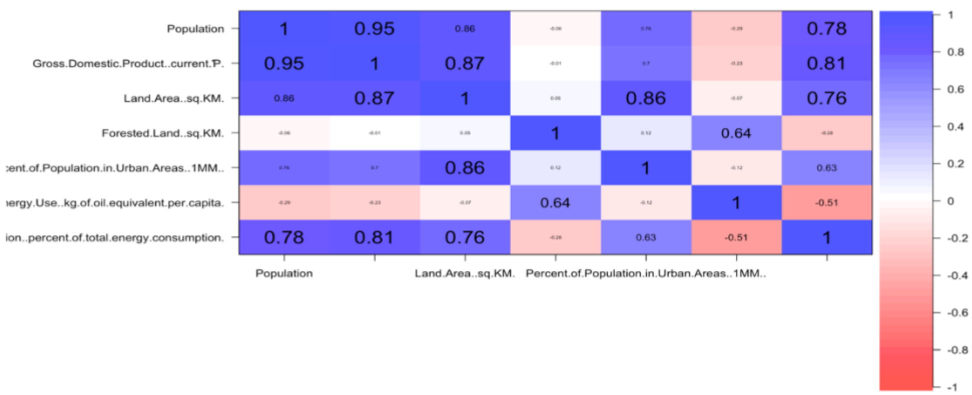
Graph 5.1-3: Regression Model Diagnose

Since we get a strong relationship between the ΔGDP and Δ CO_2, we can interpret this as:

When the country increases 1 unit (metric tonnes) of CO_2 emission, the GDP will increase by $621.5 correspondingly.

In the macro view, there are still lots of other significant factors that contribute to GDP, so we approximately assume that CO_2 emission has the contribution of 0.1% to GDP in Pullanta:

Appendix B: Correlation Matrix for Independent Variables



Appendix C: List of Enclosed Files

LinearModel-UTBoundless.Rmd

Pricing-UTBoundless.Rmd

BondPricing.xlsx

Appendix D (Python codes)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.pylab as pylab
import os
import statsmodels.api as sm
from sklearn import linear_model
from sklearn.metrics import mean_squared_error, r2_score
import scipy.stats as stats

#import matplotlib as mpl
#mpl.rcParams.update(mpl.rcParamsDefault)

#params = {'legend.fontsize': 'x-large',
#          'figure.figsize': (10, 4),
#         'axes.labelsize': 'x-large',
#         'axes.titlesize':'x-large',
#         'xtick.labelsize':'x-large',
#         'ytick.labelsize':'x-large'}
params = {'figure.figsize': (8, 4)}
pylab.rcParams.update(params)

path = r'C:\Patrick\MFI\SOA case study'
os.chdir(path)
dfs = pd.read_excel('2020-pullanta-total-c02e-emissions.xlsx', sheet_name=None)


#%%
aggData = dfs['AggregateData']
aggData = aggData.drop(aggData.columns[0], axis=1) # delete first empty column
colNames = list(aggData.iloc[8])
aggData = aggData[9:34]
aggData.columns = colNames
aggData.index = aggData.Year
aggData = aggData.drop('Year', axis=1)
#aggData = aggData.reset_index(drop=True)

emiData = aggData.iloc[:,0:7].copy()
countryData = aggData.iloc[:,7:14].copy()
biocapData = aggData.iloc[:,14:20].copy()
ecoData = aggData.iloc[:,20:].copy()

#%%
compData = dfs['CompanyData']
compData = compData.drop(compData.columns[0], axis=1)
colNames = list(str(x) for x in compData.iloc[10])
compData = compData[11::]
compData.columns = colNames
compData = compData.reset_index(drop=True)
compData = compData.rename(columns={'2018.0':'2018','2017.0':'2017',
                         '2016.0':'2016','2015.0':'2015',})

#%%
```

```
#comp_emi_data = compData[['2019','2018','2017','2016','2015']]
#comp_avg = np.mean(comp_emi_data, axis=1)
recent_years = ['2019','2018','2017','2016','2015']
compData_2 = compData.copy()
compData_2['comp_avg'] = np.mean(compData_2[recent_years], axis=1)
compData_2 = compData_2[(compData_2['comp_avg']!=0)].reset_index(drop=True)

 # Average emission of active companies
annual_avg = np.mean(compData_2[recent_years],axis=0)

plt.figure()
plt.bar(recent_years, annual_avg)
plt.title('Average Emission of Active Companies')
plt.ylabel('Avg emission (metric tonnes)')
plt.xlabel('Year')
plt.show()
# Note that 2019 data are estimated based on mid-year reporting
# There are potentially more emission as weather becomes colder in winter.

#%%
plt.figure()
plt.bar(emiData.index[-5:], emiData.Total.iloc[-5:])
plt.show()


#%%
cto = np.sum(compData_2[recent_years], axis=0)
plt.figure()
plt.bar(cto.index, cto)
plt.show()


#%%
#emiData.columns
plt.figure()
for sector in emiData.columns[:-1]:
#    plt.figure()
    plt.plot(emiData.index, emiData[sector], label=sector)
#    plt.title(str(sector))
#    plt.show()
plt.legend()
plt.show()

#%%
plt.figure()
GDPname = countryData.columns[1]
plt.plot(countryData.index, countryData[GDPname])
plt.title(GDPname)
plt.show()

#%%


# Predict future factors that might effect carbon emission
def predict_factor(data, title, plot_past=False, plot_all=True):
```

```
    pastYears = np.array(data.index)
    futureYears = np.array(range(pastYears[-1]+1,2031))
    X_train = pastYears.reshape(-1,1)
    X_pred = futureYears.reshape(-1,1)

    if plot_past:
        plt.figure()
        plt.plot(data)
        plt.title('Historical '+title)
        plt.show()

    regr = linear_model.LinearRegression()
    weight = np.array(range(1,len(data)+1))
    regr.fit(X_train, data,weight)
#    r2score = regr.score(X_train, data,weight)
#    print('R2 is ' + str(round(r2score,4)))
    Y_pred = regr.predict(X_pred)

    # The coefficients
    #print('Coefficients: \n', regr.coef_)
    #plt.plot(X_pred, urbanPopPred)
    #plt.show()

    if plot_all:
        plt.figure()
        plt.plot(np.append(X_train,X_pred), np.append(data, Y_pred))
        dataMin = np.min(np.append(data, Y_pred))
        dataMax = np.max(np.append(data, Y_pred))
        plt.plot(np.ones(100)*2019, np.linspace(dataMin,dataMax,num=100), 'b--')
        plt.title(title + ' with Predictions')
        plt.ylabel(title)
        plt.xlabel('Years')
        plt.show()
    return Y_pred
def predict_factor_CI(data, title, plot_CI=False, CI=0.9, sigma=1, n=1):
    Y_pred = predict_factor(data, title, False, False)
    pastYears = np.array(data.index)
    futureYears = np.array(range(pastYears[-1]+1,2031))
    X_train = pastYears.reshape(-1,1)
    X_pred = futureYears.reshape(-1,1)
    z = stats.norm.ppf(CI)
    upper = Y_pred + z * sigma / np.sqrt(n)
    lower = Y_pred - z * sigma / np.sqrt(n)

    if plot_CI:
        plt.figure()
        plt.plot(np.append(X_train,X_pred), np.append(data, Y_pred))
        dataMin = np.min(np.append(data, Y_pred))
        dataMax = np.max(np.append(data, Y_pred))
        plt.plot(np.ones(100)*2019, np.linspace(dataMin,dataMax,num=100), 'r--')
        plt.plot(X_pred, upper, 'b-')
        plt.plot(X_pred, lower, 'b-')
        plt.title(title + ' with Predictions and ' + str(CI*100)+'% CI')
        plt.ylabel(title)
        plt.xlabel('Years')
        plt.show()
```

```
    return upper, lower


#%%
popName = countryData.columns[0]
urbanPrecName = countryData.columns[4]
urbanPop = countryData[popName][:-1] * countryData[urbanPrecName][:-1]

#%%

pastYears = np.array(urbanPop.index)
futureYears = np.array(range(pastYears[-1]+1,2031))
X_train = pastYears.reshape(-1,1)
X_pred = futureYears.reshape(-1,1)

plt.figure()
plt.plot(urbanPop)
plt.title('Historical Population')
plt.show()

regr = linear_model.LinearRegression()
regr.fit(X_train, urbanPop)
urbanPopPred = regr.predict(X_pred)

# The coefficients
#print('Coefficients: \n', regr.coef_)
#plt.plot(X_pred, urbanPopPred)
#plt.show()


plt.figure()
plt.plot(np.append(X_train,X_pred), np.append(urbanPop, urbanPopPred))
plt.title('Urban Population with Predictions')
plt.ylabel('Population')
plt.xlabel('Years')
plt.show()


#%%
y = np.array([emiData.Total.iloc[-2], emiData.Total.iloc[-2]*0.75])
x = np.array([2018, 2030])
xq = np.array(range(2019,2031))
emissionGoal = np.interp(xq, x, y)
plt.plot(xq, emissionGoal)

y1 = np.array([1, 0.75])
emissionPercGoal = np.interp(xq,x,y1)
emissionPercGoal = pd.DataFrame(emissionPercGoal, index=xq)



#%%
#emissionPred = pd.DataFrame(index=futureYears)
emissionPred = []
```

```
buildingEmiPast = emiData[emiData.columns[0]][:-1]
buildingEmiPred = predict_factor(buildingEmiPast, 'Building and Land Use', False)
#emissionPred['B'] = buildingEmiPred
emissionPred.append(buildingEmiPred)


emcEmiPast = emiData[emiData.columns[1]][:-1]
emcEmiPred = predict_factor(emcEmiPast, 'Energy, Manufacturing & Construction', False)
#emissionPred['E'] = emcEmiPred
emissionPred.append(emcEmiPred)

industyEmiPast = emiData[emiData.columns[2]][:-1]
industyEmiPred = predict_factor(industyEmiPast, 'Industrial Processes & Product Use', False)
#emissionPred['I'] = industyEmiPred
emissionPred.append(industyEmiPred)

otherEmiPast = emiData[emiData.columns[3]][:-1]
otherEmiPred = predict_factor(otherEmiPast, 'Other', False)
#emissionPred['O'] = otherEmiPred
emissionPred.append(otherEmiPred)

transportEmiPast = emiData[emiData.columns[4]][:-1]
transportEmiPred = predict_factor(transportEmiPast, 'Transport', False)
#emissionPred['T'] = transportEmiPred
emissionPred.append(transportEmiPred)

wasteEmiPast = emiData[emiData.columns[5]][:-1]
wasteEmiPred = predict_factor(wasteEmiPast, 'Waste', False)
#emissionPred['W'] = wasteEmiPred
emissionPred.append(wasteEmiPred)

emissionPred = np.array(emissionPred).T
#emiPredTotal = np.sum(emissionPred, axis=1)
emissionPredPerc = np.true_divide(emissionPred, emissionPred.sum(axis=1, keepdims=True))
#emissionPredPerc = pd.DataFrame(emissionPredPerc, columns=['B', 'E', 'I', 'O', 'T', 'W'],
index=futureYears)

sectorEmiGoal = np.multiply(emissionPredPerc.T, emissionGoal).T
sectorEmiGoal = pd.DataFrame(sectorEmiGoal, columns=['B', 'E', 'I', 'O', 'T', 'W'], index=futureYears)

#%%

BbondCC = np.zeros(len(futureYears)) # annual carbon coupon of bond in Building and Land Use
BbondCC[0] = 5000
BbondNum = int(np.ceil(sectorEmiGoal['B'].iloc[0] / BbondCC[0]))
#print('Total number of bond issued for Building and Land sector is ' + str(BbondNum))
BbondNum

EbondCC = np.zeros(len(futureYears))
EbondCC[0] = 5000
EbondNum = int(np.ceil(sectorEmiGoal['E'].iloc[0] / EbondCC[0]))
EbondNum

IbondCC = np.zeros(len(futureYears))
IbondCC[0] = 5000
IbondNum = int(np.ceil(sectorEmiGoal['I'].iloc[0] / IbondCC[0]))
```

```
IbondNum

ObondCC = np.zeros(len(futureYears))
ObondCC[0] = 5000
ObondNum = int(np.ceil(sectorEmiGoal['O'].iloc[0] / ObondCC[0]))
ObondNum

TbondCC = np.zeros(len(futureYears))
TbondCC[0] = 5000
TbondNum = int(np.ceil(sectorEmiGoal['T'].iloc[0] / TbondCC[0]))
TbondNum

WbondCC = np.zeros(len(futureYears))
WbondCC[0] = 5000
WbondNum = int(np.ceil(sectorEmiGoal['W'].iloc[0] / WbondCC[0]))
WbondNum


#%%
# change sigma and n based on assumptions
buildingUpper, buildingLower = predict_factor_CI(buildingEmiPast, 'Building and Land Use', True,
sigma=5e7, n=100)

emcUpper, emcLower = predict_factor_CI(emcEmiPast, 'Energy, Manufacturing & Construction', True,
sigma=1e8, n=100)

industyUpper, industyLower = predict_factor_CI(industyEmiPast, 'Industrial Processes & Product Use',
True, sigma=5e7, n=100)

otherUpper, otherLower = predict_factor_CI(otherEmiPast, 'Other', True, sigma=1e6, n=100)

transportUpper, transportLower = predict_factor_CI(transportEmiPast, 'Transport', True, sigma=1e8,
n=100)

wasteUpper, wasteLower = predict_factor_CI(wasteEmiPast, 'Waste', True, sigma=1e7, n=100)
```

8. References

10 Year Treasury Rate - 54 Year Historical Chart

https://www.macrotrends.net/2016/10-year-treasury-bond-rate-yield-chart