Ride Like the Wind Without Getting Winded: The Growth of E-bike Use

Team #16412

March 2, 2023

Executive Summary

The increasing affordability and efficiency of electric vehicles in the US in recent years have led to a significant surge in their sales — and this trend will continue [1]. Contrary to popular belief, the primary contributor to this surge was not electric cars, but rather, electric bicycles (e-bikes) [2]. These e-bikes have revolutionized the transportation industry and provided a new alternative to transit. Our team aims to predict the number of e-bikes that will be sold both two and five years from now. We also aim to determine which factors contribute to this growth in sales, and quantify the impacts that e-bikes will have on other modes of transportation, carbon emissions, health, and traffic conditions.

We first predicted the number of electric bike sales in the US in 2 years (2025) and 5 years (2028) by creating an equation while taking inspiration from a system dynamics model. We identified the primary factors contributing to the growth of e-bike sales: number of cars, number of bikes, environmental perception, disposable income, price of gas, price of electricity, price of lithium ion batteries, temperature, obesity rate, CPI transportation, EPI, and the number of e-bike Google searches. We found the best-fit regression curve (linear, sinusoidal, or inverse) of each factor and further classified them under three indices: location, word of mouth, and price. Using a minimal loss function to regress for coefficients, we created an equation. We concluded that the number of bike sales will be approximately 2.06 million in 2 years (2025) and approximately 3.42 million in 5 years (2028).

Next, we determined the relative importance of each factor contributing to the growth in e-bike sales in the US. We used the same factors as Part 1, however, we replaced the price of electricity with the price of lithium ion batteries and omit the transportation CPI factor. Using the random forest feature importance algorithm, we determined that the following factors hold relative importance: temperature (0.16), number of Google searches (0.15), rate of obesity (0.14), and environmental perception (0.13). A sensitivity analysis determined that word of mouth was very sensitive compared to the other factors.

Finally, we quantified the impacts of e-bikes will have on American carbon emissions, traffic congestion, and calories burnt to represent health. Recognizing that our Part 1 model could not be generalized for long term use, we used a predator-prey model in which we modeled bikes and pedestrians as prey, and e-bikes and cars as predators. Using this data, we created equations regarding carbon emissions per car, traffic conditions in cities, and calories burnt while cycling. We determined that carbon emissions will reduce to almost zero in 20 years, traffic congestion will generally decrease over time, and Americans will become healthier by burning more calories due to the growth of e-bike sales.

Contents

Global Assumptions

1. *Everyone who can buy an e-bike can ride an e-bike.* Within the U.S. population, people who are physically unfit to ride an e-bike are minimal. Thus, even though these people would be unable to use an e-bike, we will assume that this population are negligible and will not account for them.

2. *There will be no major legislation put into place in the next 5 years that will impact transportation use.* Legislation can impact what type of transportation people use with actions such as instant rebates and cheap loans. However, we assume that this legislation does not change, as it is beyond the scope of our paper to account for political decisions and politician plans.

3. *There will be no major shifts in US economic landscape.* It is possible for the economic structure of the US to completely change, thus changing the ways in which people travel. However, as this is unlikely to change greatly, we will assume the general economic makeup of the US remains constant, and no major inflationary or recessionary periods occur.

4. *All economic decisions made by the consumer are financially optimal and businesses will cater towards their needs.* The average consumer always has their financial interests in mind. It is reasonable to assume that such a consumer makes the most optimal decision. Thus, businesses need to adhere to the demands of these consumers to sustain themselves.

5. *There will be no supply chain shortages that affect the availability and manufacturing of vehicles and their parts.* There is no evidence of an upcoming supply chain breach, shortage or any form of malfunction. It is reasonable to assume the supply of vehicles and their parts will be sufficient to address consumer demand.

6. *Google searches of the key word "e-bike" are representative of the "coolness factor."* New topics, trends, and ideas are generally searched up on the world's most popular search engine, Google. It is reasonable to assume searches with the key word "e-bike" represent the relevancy of e-bikes.

7. *Consumers of public transport will not switch to electric bikes, traditional bikes, cars, or walking.* While some may do this switch, for the large part most consumers who select public transport due so because of already existing cost efficiency and convenience. Therefore, they are unlikely to voluntarily switch off of public transport.

1 Part I: The Road Ahead

1.1 Restatement of the Problem

In this problem, we are tasked with finding the sales of electric bikes in two years (2025) and in five years (2028). We interpret this as finding a bike sales index, which converts population to overall electric bike sales. To do this with dynamic factors, we develop our own indices to account for each component to estimate this index.

1.2 Assumptions

1. *The population changes according to census trends.* We account for average birth rates and death rates in our analysis by using census data. There is no evidence that new developments or trends will cause birth and death rates to change significantly, thus, it is reasonable to assume trends will follow an established census.

1.3 Variables

Variable	Description	Unit
N_C[3]	Number of cars	Cars
N_B[4]	Number of bikes	Bikes
E_p [5]	Environmental perception	Proportion
N_I [5]	Disposable income	US Dollars
P_G [5]	Price of gas	US Dollars
P_E[6]	Price of electricity	Cents per kilowatt hour
P_L[7]	Price of Lithium Batteries	Cents per kilowatt hour
T[6]	Temperature	Degrees Celsius
R_O[8]	Rate of obesity	Proportion
C [9]	CPI Transportation	Index
E_{pi}[10]	Environmental Performance Index	Index
N_G[11]	Number of e-bike Google searches	Searches
N_E[5] [12]	E-bike sales	E-bikes
P_{US}[13]	Population of the US	People
I_b	E-bike Index	Unitless
I_l	Location Index	Unitless
I_w	Word of Mouth Index	Unitless
I_p	Price Index	Unitless

Table 1: Variables for Part II

1.4 Model Development

In our model, we develop a e-Bike Index which can be multiplied by the current US population in order to predict the number of e-bike sale in that year.

$$N_E = I_b \cdot P_{US} \tag{1}$$

From historical data, we calculate the values for I_b using known values of N_E and P_{US} from years $2012 - 2022$. We identify the three main causes of e-bike sales to be the status of location factors, word of mouth factors, and price factors which have their own indexes I_l, I_w, and I_p, respectively. We group each factor under the indices as such:

$$I_l = T \cdot R_O \cdot N_C \cdot N_B \tag{2}$$

$$I_w = N_G \cdot E_p \cdot E_{pi} \tag{3}$$

$$I_p = N_i \cdot P_G \cdot P_L \cdot C \cdot P_E \tag{4}$$

We multiply each factor to determine the proportionality and change of each. From here, we use historical data for each of the factors to calculate an index value for I_l, I_w, and I_p from years $2012 - 2022$. We are then able to combine those three indexes to calculate our final I_b,

$$I_b = aI_l + bI_w + cI_p + d \tag{5}$$

where we use Python to regress for constants a, b, c, and d using a mean square loss function. We choose a mean square loss function due to the lack of outliers present in our data set. The main weakness of mean square error is its sensitivity to outliers, but we are able to largely avoid this weakness. Furthermore, the benefits of using the mean square error are great as we require extreme accuracy on our coefficients. We do not need much graphical clarity, as this function does not need to be visualized. Therefore, we overall decide upon the mean absolute error loss function. Following this, we use the Gradient Descent function. We make this decision as the gradient descent technique is efficient. Given our complex three-variable regression, the gradient descent is the best choice as it can easily handle multiple variables. We conclude the minimal loss to be 4% which is a fairly small error percentage.

Coefficient	Value
a	$0.11 \cdot 10^{-15}$
b	$0.71 \cdot 10^{-3}$
c	$0.01 \cdot 10^{-8}$
d	-130

Table 2: Coefficients for Equation 5

In order to obtain data for 2025 and 2028, we regress each factor using either simple linear, quadratic, cubic, sinusoidal, or exponential regression. We want to choose the simplest one possible since we only had 11 years of data available. We visualize the data

plots and then applied the best fitting regression based on the points. Knowing that the mean square error loss function punishes outliers greatly, we run it on each data set iteration, removing one outlier between each iteration until the loss function calmed to a regulated and reasonable level. We apply this for each regression, taking the overall lowest error value. One factor that did not have a reasonable fit was price of lithium ion batteries. The best fit ended up being quadratic and kept increasing unreasonably in the short-term future. We instead hold the lowest price that had previously existed, 127 cents per kilowatt hour, for all future years because society would not adopt an economically less advantageous technology if there was a cheaper one available. This is also to assume the worst case scenario. Another exception to this was fitting for the gas prices. The fitting comes out to be best with cubic, but we find that implementing a cubic function resulted in unreasonable gas prices in the future. Using reasoning combined with error numbers, we decide to utilize the sinusoidal function. From here, we are able to get the following trends and regression.

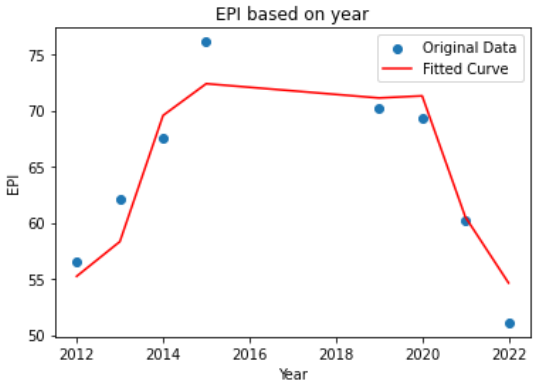

Figure 1:
$y = -9.26\ \sin(-1.30x + 4649.28) + 63.85$

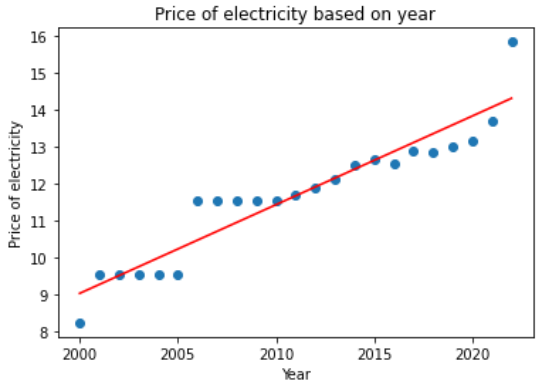

Figure 2: $y = 0.2402371541501977x - 471.4395256916997$

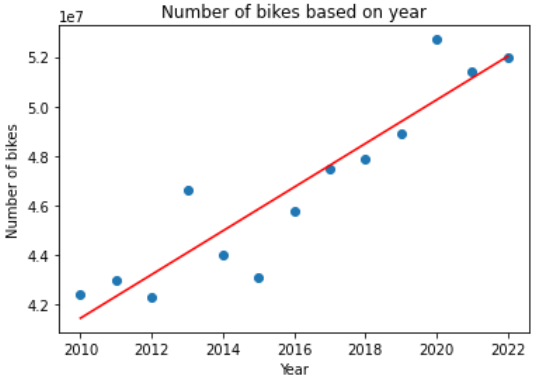

Figure 3: $y = 880769.2307692309x - 1728892307.692308$

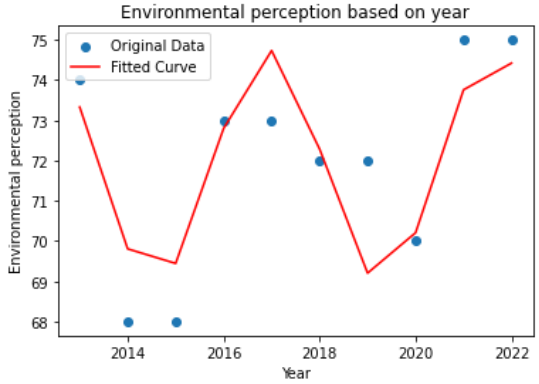

Figure 4:
$y = 2.88\ \sin(1.33x - 672.44) + 71.87$

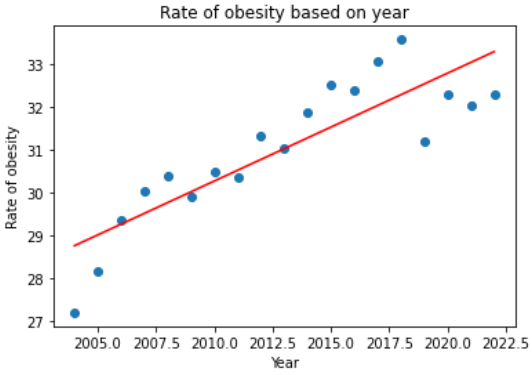

Figure 5: $y = 0.2523684210526314x - 476.99131578947333$

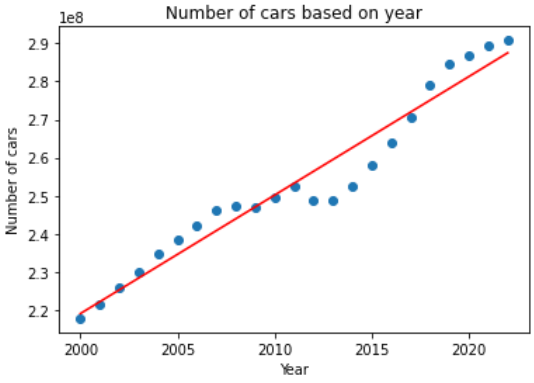

Figure 6: $y = 3112252.964426878x + -6005445059.288539$

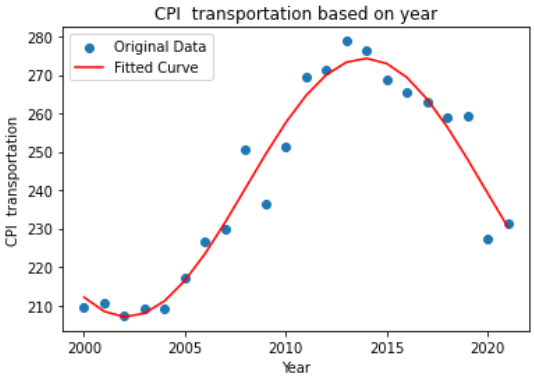

Figure 7: $y = -33.67 \sin(0.27x + 1479.8) + 240.66$

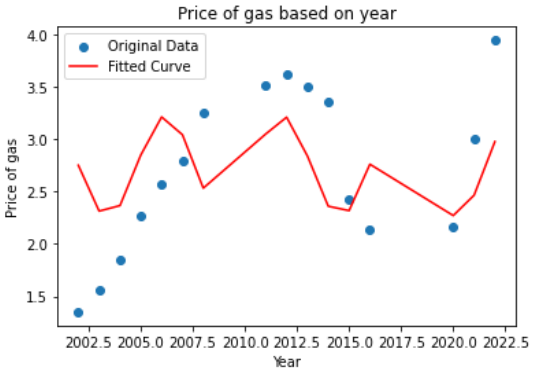

Figure 8: $y = 107.67080745341613x - 217090.47204968942$

[t]

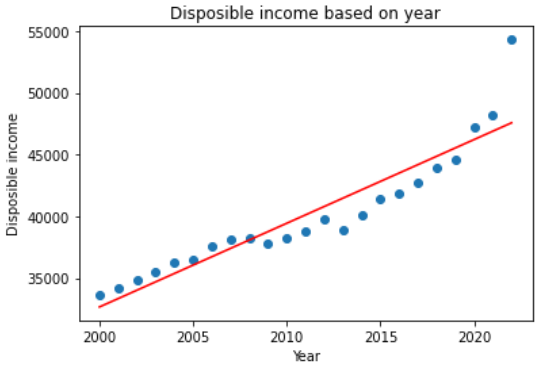

Figure 9: $y = 677.5612648221344x - 1322454.2252964426$

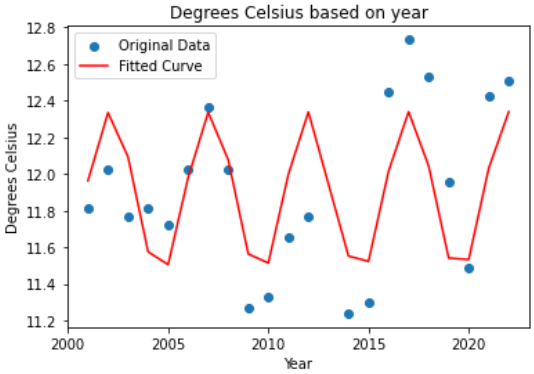

Figure 10: $y = 0.44 \sin(1.26x - 532.19) + 11.90$

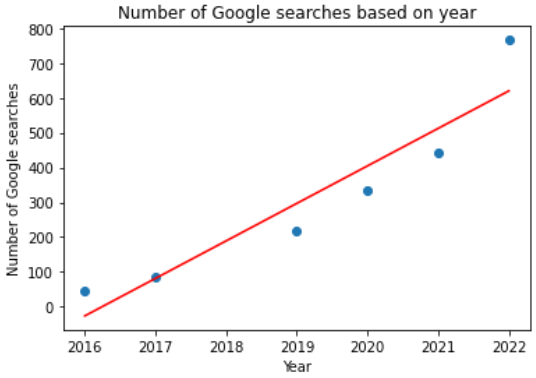

Figure 11: $y = 107.67080745341613x - 217090.47204968942$

1.5 Results

Substituting in 2025 and 2028 into each regression equation, we are able to get the the following values:

	2025	**2028**
N_C	296867193.7	306203952.6
N_B	54665384.62	57307692.31
E_p	69.87357515	74.74727123
N_I	49607.33597	51640.01976
P_G	2.87245234	2.517198047
P_E	15.04071146	15.76142292
P_L	127	127
T	12.19475671	11.85796986
R_O	34.05473684	34.81184211
C	249.7296763	270.3988659
E_{pi}	64.98136439	56.70770419
N_G	1665.967997	3013.341586

Table 3: Predicted factor values in 2025 and 2028

We then substitute these calculated factors into Equation 2, Equation 3, and Equation 4. We obtain I_l, I_l, I_l, I_l and substitute these values into Equation 5. Finally, we take US Census[13] estimation of the population at 2025 and 2028 to be 344,234,000 and 350,872,000 people, and combine it with Equation 5 to calculate e-bike sales in Equation 1. We determine that in 2025, there will be $\boxed{2.06 \text{ million} \pm 4\ \%}$ e-bike sales in the US and in 2028, $\boxed{3.42 \text{ million} \pm 4\ \%}$ e-bike sales in the US.

1.6 Strengths and Weaknesses

A strength to our model is that it is relatively not sensitive to changes due to the use of addition of our three main indices and multiplication within the indices. Adding can help to normalize the scale of the index and to adjust for known relationships between the variable. This can make it easier to compare changes in the index over time or between different groups.

A weakness of the model is that the regressions developed, especially those cubic, will generate unreasonable numbers the farther in the future we try to estimate. Since e-bike are a relatively new product for the global market, data on sales was relatively unavailable. Long-term trends that could reveal different regression behaviors weren't apparent enough to determine. However, since we were only used to estimate sales for short-term trends, our higher powered polynomial regressions would be locally accurate although behavior past five years could deviate from the real trends.

2 Part II: Shifting Gears

2.1 Restatement of the Problem

In this problem, we are tasked with determining the factors that significantly contribute to e-bike growth. We implement the random forest feature importance algorithm on each factor that contributed to this growth in sales in order to determine the relative importance of each factor.

2.2 Assumptions

1. *The Environmental Performance Index (EPI) of a given year is the average of the EPIs of the preceding and succeeding year if it is not explicitly stated in the data set.* The environment is constantly changing. Taking the average of the preceding and succeeding years is a reasonable estimate of the environmental performance of the year between.

2. *A 20% increase in e-bike sales from the preceding year can be classified as a sufficient increase.* For large-cap companies, a 5-10% increase in sales is considered sufficient. For mid-cap and small-cap companies, a 10% increase in sales is considered sufficient. We assume the worst-case scenario and set the sufficient condition as a 20% increase in sales between years [14].

2.3 Variables

Variable	Description	Unit
N_C	Number of cars	Cars
N_B	Number of bikes	Bikes
E_p	Environmental perception	Proportion
N_I	Disposable income	US Dollars
P_G	Price of gas	US Dollars
P_L	Price of lithium ion batteries	US Dollars
T	Temperature	Degrees Celsius
R_O	Rate of obesity	Proportion
E_{pi}	Environmental Performance Index	Index
N_G	Number of e-bike Google searches	Searches
N_E	E-bike sales	E-bikes
I	Bike sales indicator	0 or 1

Table 4: Variables for Part II

2.4 Model Development

In our model, we seek to determine the relative importance of the factors contributing to a growth in e-bike sales. We first determine the bike sales indicator I by calculating the increase in e-bike sales between each year from 2012-2022. As entailed by Assumption 2, if e-bike sales increase by 20% or more, we set the indicator as 1 and if they increase by less than 20%, we set the indicator as 0.

We then use the random forest feature importance algorithm on the factors listed above in Table 4. We use a 90-10 train test split to implement this algorithm. This means we use 90% of the data set to train the decision trees and 10% of the data set to test the model and achieve the relative feature importance.

We take into account nuanced factors and their trends from 2012-2022 that are the most likely to contribute to the increase in e-bike sales within our model. The trends of an increase in the number of cars and bikes are representative of the increasing necessity of transportation. The trends of an increase in environmental perception, or the proportion of individuals who are aware of declining environmental conditions, and the EPI are representative of the increasing inclination to transition to zero emission machinery. The trend of an increase in disposable income is representative of the increasing capability of purchasing e-bikes. The trends of an increase in gas prices and decrease in lithium ion batteries are representative of increasing long-term financial efficiency. The trend of an increase in temperature is representative of the increasing ability to ride an e-bike in suitable weather conditions. The trend of an increase in obesity rates is representative of the increasing necessity of health benefits. The trend of an increase in Google searches regarding e-bikes is representative of the increasing "coolness factor." The random forest feature importance algorithm will help us determine which of these factors contribute to e-bike sales.

2.5 Results

The results we return from the random forest feature importance algorithm are depicted in the following graph:

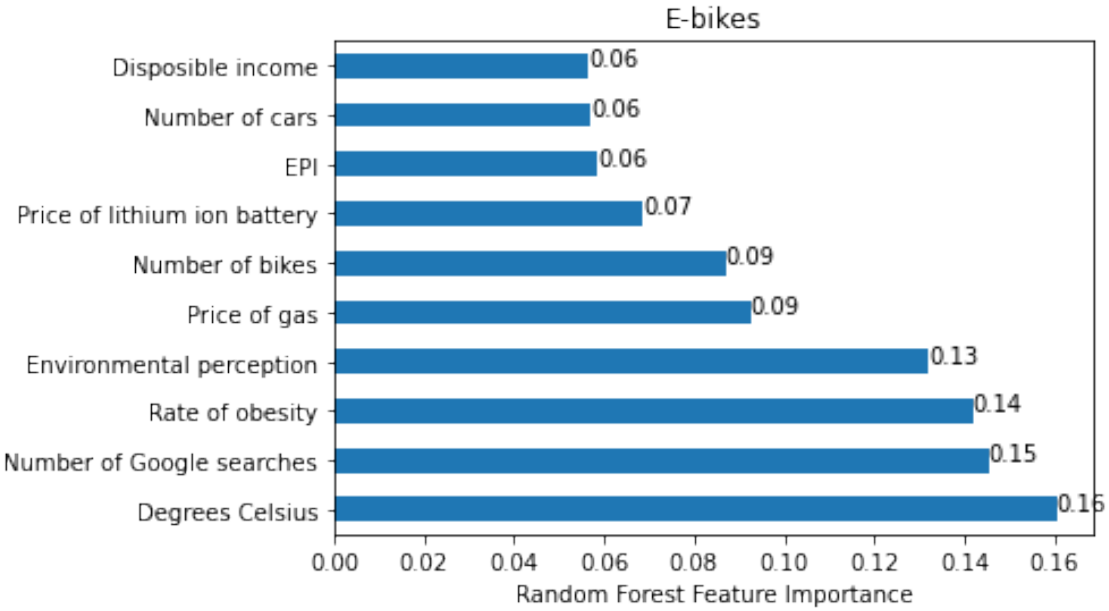

Figure 12: Random Forest on factors contributing to e-bike sales

In a trivial scenario in which each of the factors have an equal contribution in the growth of e-bike sales, each of the feature importance values would be 0.1 because $1/10$ factors $= 0.1$. In other words, each of the ten factors would have a weight of 10% towards the growth in e-bike sales.

However, the factors do not truly have equal importance. According to Figure 12, the temperature, number of Google searches, rate of obesity, and environmental perception, all have a relative importance greater than 0.1 at 0.16, 0.15, 0.14, and 0.13, respectively. We conclude that these features contribute to the growth in e-bike sales significantly more than the expected contribution. Thus, the disposable income, number of cars, EPI, price of lithium batteries, number of bikes, and the price of gas are not significant reasons for the growth in e-bike usage. On the other hand, the temperature, number of Google searches regarding e-bikes, rate of obesity, and environmental perception are significant reasons for the growth in e-bike usage.

2.6 Sensitivity Analysis

%Δ in Independent Variable		%Δ in Dependent Variable
IV 1 (Degrees Celsius)	+10%	+1.24%
IV 1 (Degrees Celsius)	−10%	−1.24%
IV 2 (Number of Google searches)	+10%	+8.97%
IV 2 (Number of Google searches)	−10%	−8.97%
IV 3 (Rate of obesity)	+10%	+1.24%
IV 3 (Rate of obesity)	−10%	−1.24%

Table 5: Sensitivity analysis for Part II

To determine the sensitivity of the dependent variable, we change the top 3 most important independent variables by 10 percent. We use the values from our predicted year 2025 and recalculate the percent change in e-bike sales. As seen in Table 5, the obesity rate and temperature have little effect on the e-bike sales. Google searches have a large impact on our dependent variable because the popularity of products has a huge effect on the consumer's demand, and thus, heavily changes the number of sales. For example, communities in Brompton used mass advertisements to attract new users to e-biking, which proved succesful [15]. As we can see, the word of mouth spreads popularity amongst communities which snowballs — heavily impacting the e-bike industry in regards to demand and the amount of sales.

2.7 Strengths and Weaknesses

A strength of the random forest feature importance algorithm is its ability to remain consistent as it produces the same results regardless of the specific subset of the data used for training and testing. The algorithm is also very robust to noisy data, allowing it to accurately calculate results despite the existence of irrelevant and redundant data. Furthermore, the algorithm is very resistant to over fitting - a problem in which a model fits the training data too closely - resulting in poor performance when applied to new data. This is because it aggregates the predictions of multiple decision trees, thus reducing the impact of over fitting and improving the model's generalization performance.

A weakness of the random forest feature is its tendency to be biased towards features with a large number of unique values because such features tend to have more opportunities for split points. Additionally, this algorithm ignores the effects of features on each other as it is a univariate method that evaluates the importance of each feature in isolation. This may result in inaccurate outcomes as this factors overlap and influence one another in reality. This algorithm is also vulnerable to outliers within our data. Although we cleaned our data

3 Part III: Off the Chain

3.1 Restatement of the Problem

In this problem, we are tasked with quantifying the impact of the shift to electric bikes. To do so, we recognize that our model in Part I made many short term assumptions, and cannot be generalized to the long term. In the long term, we implement the predator-prey model that interprets electric bikes and cars as predators and bikes and pedestrians as prey. We justify this by claiming that electric bikes and cars will decrease users of traditional bikes and the number of individuals who walk.

3.2 Assumptions

1. *All cars emit 4.6 metric tons of carbon dioxide per year.* Though cars have great variability, and their usage differs based on the user, the U.S. government has found an average of 4.6 metric tons of carbon dioxide per car. Analyzing further and looking into specifics is beyond the scope of this model [16].

2. *Americans spend equal hours on electric bikes as compared to normal bikes.* As the two bike types travel at roughly the same speed, it is reasonable to assume Americans spend equal time on each.

3.3 Variables

Variable	Description
α	Intrinsic growth rate of the bikes
β	Predation rate of electric bikes on bikes
γ	Intrinsic growth rate of pedestrians
δ	Predation rate of the cars on pedestrians
E	Carbon emissions
C	Number of Cars (in millions)
L	Congestion index
K	Maximum flow rate of the average US lane
D	Maximum capacity of the average US lane
B	Number of traditional bikes (in millions)
E_l	Number of electric bikes (in millions)
C_b	Calories burnt by all Americans due to biking

Table 6: Variables for Part III

3.4 Model Development

We use a predator-prey model over other alternatives such as the SIR or mutualism model. This is due to the nature of the transportation industry; in the transportation industry, there are numerous forms of transportation which compete with one another. Though the SIR model could be utilized, it would be largely inefficient as the spread of electric bikes will be highly contested by other modes of transportation, such as traditional bikes or cars. Furthermore, the predator-prey model fits our scenario much better than models such as the mutualism model, as there is significant competition between many transportation types. For example, electric bike growth could negatively effect normal bike sales, which in turn, could improve car sales. This relationship is efficiently modeled with a predator-prey model.

For this predator-prey model, we take the 6 year expected values for electric bikes, cars, and traditional bikes from Part 1. First, we set the initial populations as the data we had for 2022. Following this, we set up equations for two predators and two prey, using the aforementioned variables as the rates in the model. Then, we optimize this predator-prey model utilizing a mean absolute error loss function. We use this loss function due to its efficiency and economy given our data. We rule out other loss functions due to some of the following issues. We rule out the mean square error loss function due to its severe punishment of outliers, and its tendency to reveal biased results. As we are optimizing essentially four curves simultaneously, there are bound to be outliers. Therefore, we want a more even spread of error, which the mean absolute error loss function provides, as it lacks sensitivity to wild outliers and would create more recognizable fitting.

Utilizing our data in this loss function, we decide to create forced optimization bounds on α, β, γ, and δ. We form these bounds by looking at reasonable growth and predation rates. We force $\alpha \in (0, 2)$, $\beta \in (0, 0.5)$, $\gamma \in (0, 1)$, and $\delta \in (0, 0.5)$. This allows for the fit to be more accurate, as it provides a general guess to where the optimized variables were, allowing the SciPy optimization to build off of it. This decision was made after seeing fits without this guessed range, as the model will occasionally create fits with unrealistically high electric bike counts and other transportation type counts. From all this, we obtain the following variable values and graph.

Variable	Value
α	The intrinsic growth rate of the bikes.
β	The predation rate of electric bikes on bikes.
γ	The intrinsic growth rate of pedestrians.
δ	The predation rate of the cars on pedestrians.

Table 7: Variables for Part III

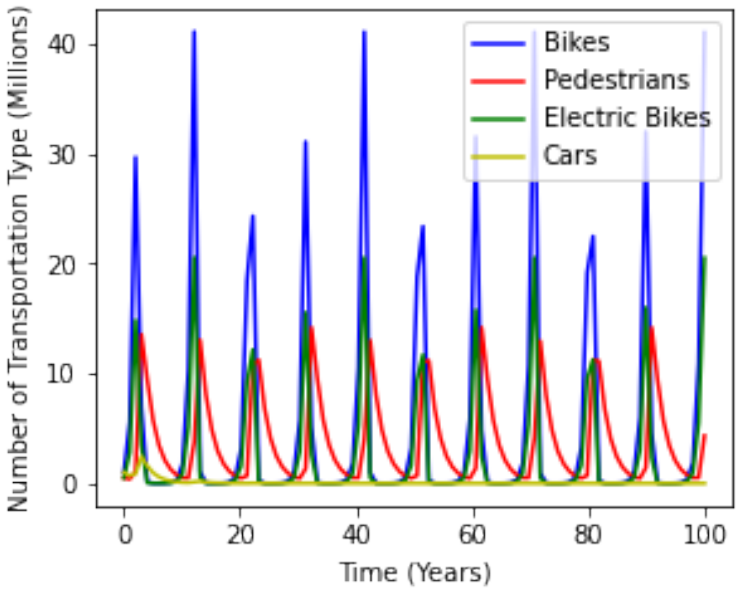

Figure 13: Predator-Prey Model

From this, we visualize the C. From our assumption regarding carbon emissions per car, we find that

$$E = 4.6C$$

to get the total metric tons of carbon emissions per year. Graphing this, we see that the total carbon emissions will follow the below trend.

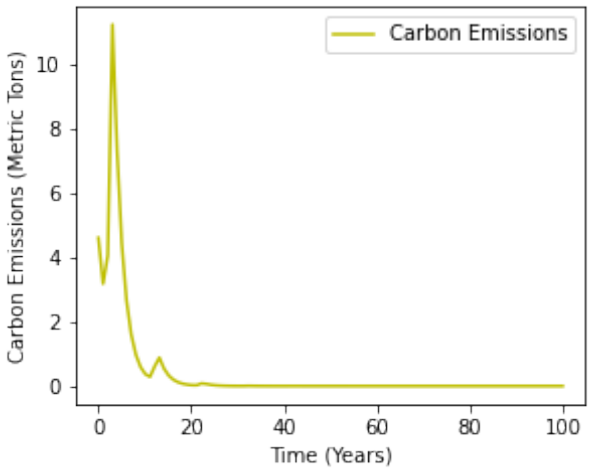

Figure 14: Carbon Emissions Emitted

Furthermore, the traffic congestion in the US can generally be modeled using the following congestion equation, which gives a congestion level index.

$$L = \frac{KD}{C + B + E_l}$$

The higher this index is, the better the traffic conditions are in a given city. For this index, we account for all vehicles which will be present on US roads, which is bikes, cars, and electric bikes. We can compare this index to today's index, such that we can see if

congestion levels will decrease or increase in the coming years. Utilizing current data, we find $K = 1900$ and $D = 2200$.[17][18]. We get the following graph from this.

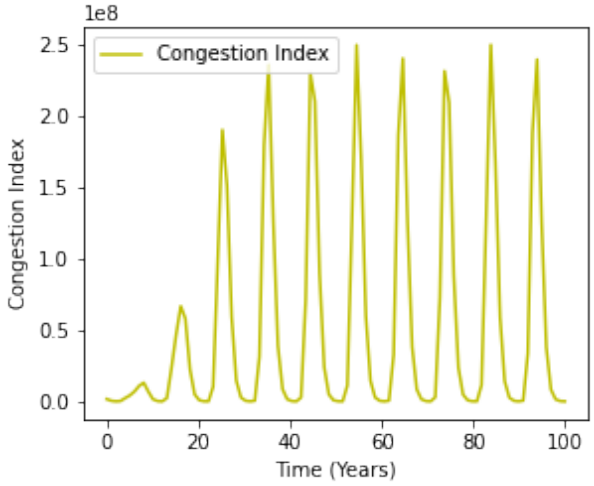

Figure 15: Traffic Congestion

Therefore, it is clear that over the next 40 years, traffic congestion will slowly get better, maintaining a oscillating pattern. After that, it should continue oscillating, but peak to a mostly consistent amount of traffic congestion.

Furthermore, we measure overall healthiness of Americans by analyzing the number of calories burned yearly by the entire US population. From sources, we know the average calories burned in an hour of electric biking is 300[19], and the average calories burned in an hour of regular biking is 480[20]. From another source, we know that the average American spends 0.183 hours a day cycling[21]. Therefore, we know that

$$C_b = (480B + 300E_l) * 0.183 * 365$$

and we get the following graph

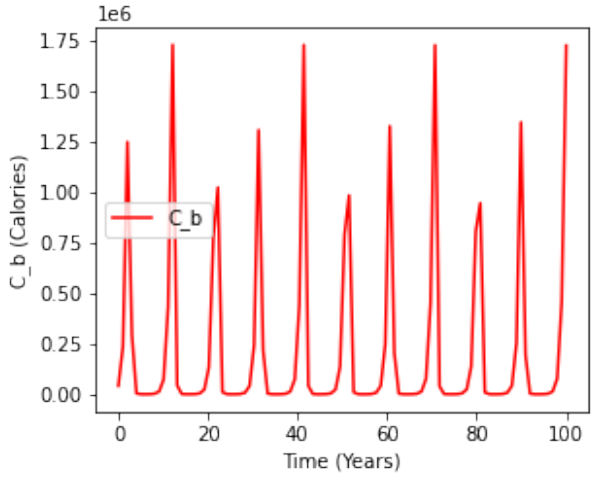

Figure 16: Calories Burnt by All Americans by Biking in One Year

As calories are directly related to health, this graph is generally representative of the impacts of biking on health of the average American. For example, in 12 years, we expect Americans to get significantly healthier due to biking, as calories burned peaks. However, in 18 years, we expect a much smaller effect on American health from biking, as biking rates decrease in that time.

3.5 Results

- *Carbon Emissions* - Will slowly reduce to 0 in about 20 years.
- *Traffic Congestion* - Over the next 40 years, traffic congestion will slowly get better, maintaining a oscillating pattern. After that, it should continue oscillating, but peak to a mostly consistent amount of traffic congestion.
- *Calories Burnt* - In 12 years, calories burnt by biking will peak, in 18 years, calories burnt by biking will be minimal again, and the cycle will repeat.

3.6 Strengths and weaknesses

The model in extremely strong in its non-peak estimates. Mean absolute error provides a very good fit on the bottom of the curves, and since the data the curves were fitted to were at the bottom of the curves. However, this causes a significant weakness of the graph, in that the peaks are not necessarily correct. This is again due to the fitting, as the known data location causes inaccuracy in the estimation of overall height of the peaks. These two lead to identical strengths and weaknesses in each of our derivative models.
Another weakness of this model is its lacking data set. Due to the limited time scope of our part one model, it would be extremely skewed if we utilized further future years past 2028 in this model. However, due to this limitation, we lacked sufficient data to regulate the peaks and maintain overall modeling accuracy. In the future, if we had more data for the past, we could also use that to model. Currently, our past data is mostly unusable here, as electric bikes are a recent development which has not impacted car, bike, and pedestrian numbers greatly. If we had more data, we could even implement a weighting system which multiplied the loss function in certain regions by differing values, so that we could ensure our curve fit more in critical portions, such as the peaks and the descent areas.

References

[1] US electric vehicle sales surge in 2022, gain on Tesla . [Online]. Available: https://thehill.com/policy/technology/3802179-us-electric-vehicle-sales-surge-in-2022-gain-on-tesla/#:~:text=Fully%20electric%20vehicles%20jumped%20in,to%20The%20Wall%20Street%20Journal.

[2] Electric bicycles are now outselling electric cars and plug-in hybrids combined in the US . [Online]. Available: https://electrek.co/2022/01/26/electric-bicycles-are-now-outselling-electric-cars-and-plug-in-hybrids-combined-in-the-us/

[3] Number of cars. [Online]. Available: bts.gov

[4] Number of bicycling participants in the united states from 2010 to 2021. [Online]. Available: https://www.statista.com/statistics/191204/participants-in-bicycling-in-the-us-since-2006/

[5] Ride like the wind, mathworks math modeling challenge 2023. [Online]. Available: https://m3challenge.siam.org/node/596.

[6] Average retail electricity prices in the united states from 1990 to 2021. [Online]. Available: https://www.statista.com/statistics/183700/us-average-retail-electricity-price-since-1990/#:~:text=The%20retail%20price%20for%20electricity,per%20kilowatt%20hour%20in%202021.

[7] Lithium battery pack prices go up for first time since bloombergnef began annual survey. [Online]. Available: https://www.energy-storage.news/lithium-battery-pack-prices-go-up-for-first-time-since-bloombergnef-began-annual-survey/

[8] Percent of adults with obesity. [Online]. Available: https://usafacts.org/data/topics/people-society/health/health-risk-factors/obesity/?utm_source=bing&utm_medium=cpc&utm_campaign=ND-StatsData&msclkid=80748f50ec941a592d0feb56d8dd2c52

[9] Consumer price index for transportation. [Online]. Available: https://www.bts.gov/consumer-price-index-transportation

[10] United states of america. [Online]. Available: https://sedac.ciesin.columbia.edu/data/collection/epi/sets/browsea

[11] Google trends e-bikes. [Online]. Available: https://trends.google.com/trends/explore?date=all&geo=US&q=ebikes

[12] Opinion: E-bikes may be the greenest form of transportation in human history. why aren't cities taking advantage? [Online]. Available: https://www.latimes.com/opinion/livable-city/la-ol-e-bike-cities-climate-change-transportation-20170126-story.html

[13] US Census. [Online]. Available: https://www.census.gov

[14] Sales Growth TTM. [Online]. Available: https://www.stockopedia.com/ratios/sales-growth-ttm-838/

[15] E-bike ad campaign targets the masses: Brompton tackles ageism. [Online]. Available: https://cyclingindustry.news/cannondale-e-bike-ad-campaign-targets-the-masses-brompton-tackles-ageism/

[16] Greenhouse gas emissions, typical passenger vehicle. [Online]. Available: https://www.epa.gov/greenvehicles/greenhouse-gas-emissions-typical-passenger-vehicle

[17] Numbers every traffic engineer should know. [Online]. Available: https://www.mikeontraffic.com/numbers-every-traffic-engineer-should-know/#:~:text=Roundabout%20controlled%3A%2035%20seconds%2Fvehicle%20Traffic%20Signal%20controlled%3A%2055,big%20events%29%201%2C900%20vehicles%20per%20hour%20per%20lane

[18] Highway capacity: Definition, importance, factors and formula. [Online]. Available: https://www.engineeringenotes.com/transportation-engineering/traffic-engineering/highway-capacity-definition-importance-factors-and-formula/48457#:~:text=%28According%20to%20HCM%2C%20the%20theoretical%20capacity%20under%20ideal,multi-lane%20highways.%20Level%20of%20Service%20Concept%20%28HCM%2C%20USA%29%3A

[19] Biking calories burned: How many calories burned cycling? [Online]. Available: https://www.bostonbikes.org/advice/how-many-calories-do-you-burn-riding-a-bike/

[20] This is how many calories you burn when e-biking. [Online]. Available: https://www.flyer-bikes.com/en/this-is-how-many-calories-you-burn-when-e-biking#:~:text=On%20an%20e-bike%2C%20therefore%2C%20you%27re%20always%20moving%2C%20and,effort%20on%20an%20e-bike%20burns%20around%20300%20calories.

[21] Time spent cycling, walking, running, standing and sedentary: a cross-sectional analysis of accelerometer-data from 1670 adults in the copenhagen city heart study. [Online]. Available: https://bmcpublichealth.biomedcentral.com/articles/10.1186/s12889-019-7679-z#:~:text=Results,579.1%20min%2Fday%2C%20respectively.

Code

```
# Import packages
import graphviz
import math
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from scipy.optimize import curve_fit, fsolve
from sklearn.ensemble import RandomForestClassifier,
    RandomForestRegressor
from sklearn.feature_selection import SelectFromModel
from sklearn.linear_model import LinearRegression
from sklearn import metrics
from sklearn.metrics import accuracy_score, r2_score
from sklearn.model_selection import train_test_split
from sklearn.tree import export_graphviz
from scipy.stats import linregress
from scipy.optimize import minimize
from scipy.optimize import Bounds



# Import data and remove the last 12 rows, which have holes in them
df = pd.read_csv("ebikerff.csv")
df = df.drop(df.index[-12:])

# The independent variables and dependent variable. Is good is based on
    the sales increases from the previous year must be > 20% for it to
    be labeled as 1.
target_list = ['Number of cars','Number of bikes','Environmental
    perception','Disposible income','Price of gas','Price of lithium ion
    battery','Degrees Celsius','Rate of obesity','EPI','Number of
    Google searches']
X = df[target_list]
y = df["is good"]
X = X.astype('float')
y = y.astype('float')


# Split into training and testing
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size
    =0.1, random_state=245)


# Train the random forest model
clf = RandomForestClassifier(n_estimators=100, random_state=14)
```

```
clf.fit(X_train, y_train)

# Visualize the forest
dot_data = export_graphviz(clf.estimators_[0], out_file=None,
                           feature_names=target_list,
                           class_names=[str(i) for i in clf.classes_],
                           filled=True, rounded=True,
                           special_characters=True)
graph = graphviz.Source(dot_data)
graph.render('Decision_Tree')

# Predict the class for each X value
y_pred = clf.predict(X_test)


# Print the accuracy of the model
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)


# Determine the feature importances and display them
importance = clf.feature_importances_
feat_importances = pd.Series(clf.feature_importances_, index=X.columns)
feat_importances.nlargest(20).plot(kind='barh')
plt.xlabel("Random Forest Feature Importance")
feature_importances = [(feature, round(importance, 2)) for feature,
    importance in zip(target_list, importance)]

# Print the error values
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test,
    y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, y_pred)
    )
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(
    y_test, y_pred)))
print("Data Size:", df.shape)
plt.title('E-bikes')

# Annotations
for index, value in enumerate(feat_importances.nlargest(20)):
  plt.text(value, index, str(round(value, 2)))

plt.show()



def dispLin(col):
```

```
df = pd.read_csv('ebike.csv')
# Single out one IV and the DV
df = df.dropna(subset=[col])
x = df['US']
y = df[col]
# The columns needing linear regression
if col == 'Number of cars' or col == 'Number of bikes' or col == '
 Number of Google searches' or col == 'Disposible income' or col == '
 Price of electricity' or col == 'Rate of obesity':
  # Linear regression values
  slope, intercept, r_value, p_value, std_err = linregress(x, y)

  # Create a scatter plot
  plt.scatter(x, y)

  # Create a line plot with the linear regression line
  plt.plot(x, slope*x + intercept, color='red')

  # Add labels
  plt.xlabel('Year')
  plt.ylabel(col)
  plt.title(col + " based on year")

  # Print the needed values, equation, slope, intercept, and
 predicted values
  plt.show()
  print("Equation: y= ", slope,"x +",intercept)
  print("Slope:", slope)
  print("Intercept:", intercept)
  y_2028 = slope * 2028 + intercept
  y_2025 = slope * 2025 + intercept
  print("Predicted value for 2028: ", y_2028)
  print("Predicted value for 2025: ", y_2025)

# The columns needing sinusoid regression
elif col == 'Environmental perception' or col == 'Degrees Celsius' or
   col == 'CPI  transportation' or col == 'EPI' or col == 'Price of
 gas':

  # Sinusoid function
  def sinusoid(x, a, b, c, d):
    return a * np.sin(b * x + c) + d

  # Fit the sinusoidal function to the data
  popt, pcov = curve_fit(sinusoid, x, y)

  # Print the coefficients of the parameters
```

```
    print("Coefficients: a = {:.2f}, b = {:.2f}, c = {:.2f}, d = {:.2f}".format(*popt))

    #Label and display values and graph
    plt.xlabel('Year')
    plt.ylabel(col)
    plt.title(col + " based on year")
    plt.scatter(x, y, label='Original Data')
    plt.plot(x, sinusoid(x, *popt), 'r-', label='Fitted Curve')
    plt.legend()
    plt.show()

    x_new = np.array([2025, 2028])
    y_new = sinusoid(x_new, *popt)
    print("Predicted values at 2025 and 2028: ", y_new)

  # The columns needing inverse regression
  elif col == 'Price of lithium ion battery':

    # Inverse function
    def inverse(x, a, b, c):
      return a / (b + x**(3/2)) + c

    # Fit the inverse function to the data
    popt, pcov = curve_fit(inverse, x, y, maxfev = 10000)

    # Print the coefficients of the parameters
    print("Optimized values: a = {:.2f}, b = {:.2f}, c = {:.2f}".format(*popt))

    # Label and display values and graph
    plt.xlabel('Year')
    plt.ylabel(col)
    plt.title(col + " based on year")
    plt.scatter(x, y, label='Original Data')
    plt.plot(x, inverse(x, *popt), 'r-', label='Fitted Curve')
    plt.legend()
    plt.show()

    x_new = np.array([2025, 2028])
    y_new = inverse(x_new, *popt)
    print("Predicted values at 2025 and 2028: ", y_new)

# Go through each column
df = pd.read_csv('ebike.csv')
for column_name in df.columns:
  if column_name != "US":
```

```
    print(column_name)
    dispLin(column_name)

# Similar function for the USA csv
def dispLinUSA(col):
  df = pd.read_csv('ebikeUSA.csv')
  df = df.dropna(subset=[col])
  x = df['US']
  y = df[col]

  # Linear regression
  slope, intercept, r_value, p_value, std_err = linregress(x, y)

  # Create a scatter plot
  plt.scatter(x, y)

  # Create a line plot with the linear regression line
  plt.plot(x, slope*x + intercept, color='red')

  # Add labels
  plt.xlabel('Year')
  plt.ylabel(col)
  plt.title(col + " based on year")

  # Display needed values
  plt.show()
  print("Equation: y= ", slope,"x +",intercept)
  print("Slope:", slope)
  print("Intercept:", intercept)
  for i in range(2028,2022,-1):
    ypred = slope * i + intercept
    print("Predicted value for ",i,': ', ypred)

# Same process
df = pd.read_csv('ebikeUSA.csv')
for column_name in df.columns:
  if column_name != "US" and not df[column_name].isnull().all():
    print(column_name)
    dispLinUSA(column_name)




# Input data of indexes to determine the coefficients for computing the
    final index
x1 = np.array
   ([6.108,5.925,5.610,5.189,5.624,5.408,4.875,4.083,3.982,4.656,3.876],
```

```
    dtype='float64')
x2 = np.array
   ([2.936,2.292,1.615,1.117,0.364,0.520,0.291,0.170,0.142,0.092,0.056],
    dtype='float64')
x3 = np.array
   ([1.197,0.608,0.414,0.611,0.716,0.767,0.873,1.302,2.743,3.066,3.343],
    dtype='float64')
y = np.array([2.78,2.26,1.25,1.29,1.13,0.81,0.47,0.41,0.61,0.50,0.22],
   dtype='float64')

# The loss function we are using
def loss(w):
    y_pred = w[0] * x1 + w[1] * x2 + w[2] * x3 + w[3]
    return np.mean((y - y_pred) ** 2).astype(y.dtype)

# The gradient function
def gradient(w):
    y_pred = w[0] * x1 + w[1] * x2 + w[2] * x3 + w[3]
    dw1 = -2 * np.mean(x1 * (y - y_pred))
    dw2 = -2 * np.mean(x2 * (y - y_pred))
    dw3 = -2 * np.mean(x3 * (y - y_pred))
    dw4 = -2 * np.mean(y - y_pred)
    return np.array([dw1, dw2, dw3, dw4], dtype=w.dtype)

# Gradient descent function
def gradient_descent(w_start, learning_rate, num_iterations):
    w = w_start
    losses = []
    for i in range(num_iterations):
        dw = gradient(w)
        w -= learning_rate * dw
        losses.append(loss(w))
    return w, losses

# Parameters
w_start = np.array([0, 0, 0, 0], dtype='float64')
learning_rate = 0.01
num_iterations = 1000

# Running and outputting the optimal coefficients
w_opt, losses = gradient_descent(w_start, learning_rate, num_iterations
   )
print("Optimized values: w1 = {:.2f}, w2 = {:.2f}, w3 = {:.2f}, w4 =
   {:.2f}".format(w_opt[0], w_opt[1], w_opt[2], w_opt[3]))
print("Final loss: {:.2f}".format(loss(w_opt)))
print("Optimized values: w1 = {:.2f}, w2 = {:.2f}, w3 = {:.2f}, w4 =
   {:.2f}".format(w_opt[0], w_opt[1], w_opt[2], w_opt[3]))
```

```
252
253
254
255
256
257
258
259 # Our predator prey model inspired by the lotka-volterra equation, but
       adapted to include multiple prey and predators.
260 def predator_prey_model(y, t, alpha, beta, delta, gamma):
261     x1, y1, x2, y2 = y
262     dx1dt = x1 * (alpha - beta * y1 - delta * y2)
263     dy1dt = y1 * (-gamma + delta * x1)
264     dx2dt = x2 * (alpha - beta * y1 - delta * y2)
265     dy2dt = y2 * (-gamma + delta * x2)
266     return [dx1dt, dy1dt, dx2dt, dy2dt]
267
268 # Our determined parameters
269 alpha = 1.2
270 beta = 0.6
271 delta = 0.8
272 gamma = 0.3
273
274 # Initial values
275 x1_0 = 1.0
276 y1_0 = 0.5
277 x2_0 = 0.5
278 y2_0 = 1.0
279 y0 = [x1_0, y1_0, x2_0, y2_0]
280
281 # Time span we want to simulate it in
282 t = np.linspace(0, 100, 100)
283
284
285 # Previous data from part 1
286 x1_obs = np.array([52.904, 53.785, 54.665, 55.546, 56.430, 57.308])
287 y1_obs = np.array([0.674, 0.676, 0.678, 0.680, 0.681, 0.683])
288 x2_obs = np.array([0.989, 1.404, 2.061, 2.847, 3.336, 3.418])
289 y2_obs = np.array([290.642, 293.754, 296.867, 299.979, 303.091,
       306.203])
290
291 # Our loss function
292 def loss_function(theta):
293     # Simulate the model with the given parameters
294     y_sim = odeint(predator_prey_model, y0, t, args=tuple(theta))
295
296     # Solve the differential equation over time span
```

```
    t_sim = np.linspace(0, 100, len(x1_obs))
    y_sim_obs = odeint(predator_prey_model, y0, t_sim, args=tuple(theta
    ))

    # Calculate the squared error between the simulated and observed
    data
    se = np.mean(np.abs(y_sim_obs[:,0] - x1_obs)
           + np.abs(y_sim_obs[:,1] - y1_obs)
           + np.abs(y_sim_obs[:,2] - x2_obs)
           + np.abs(y_sim_obs[:,3] - y2_obs))

    return se




# The bounds for the parameters
bounds = Bounds([0, 0, 0, 0], [2, 0.5, 1, 0.5])

# The optimization function
res = minimize(loss_function, theta0, method='L-BFGS-B', bounds=bounds)

# Optimal theta
theta_opt = res.x
print('Optimized parameters: alpha = {:.3f}, beta = {:.3f}, delta =
    {:.3f}, gamma = {:.3f}'.format(*theta_opt))
y_sim = odeint(predator_prey_model, y0, t, args=tuple(theta_opt))

# Plot the results
plt.figure(figsize=(10,8))
plt.subplot(221)
plt.plot(t, y_sim[:,0], 'b-', label='Bikes')
plt.plot(t, y_sim[:,1], 'r-', label='Pedestrians')
plt.plot(t, y_sim[:,2], 'g-', label='Electric Bikes')
plt.plot(t, y_sim[:,3], 'y-', label='Cars')
plt.xlabel('Time (Years)')
plt.ylabel('Number of Transportation Type (Millions)')
plt.legend()
plt.show()

plt.figure(figsize=(10,8))
plt.subplot(222)


plt.plot(t, 4.6*y_sim[:,3], 'y-', label='Carbon Emissions')
plt.xlabel('Time (Years)')
plt.ylabel('Carbon Emissions (Metric Tons)')
```

```
plt.legend()
plt.show()




plt.figure(figsize=(10,8))
plt.subplot(223)


plt.plot(t, (4180000/(y_sim[:,3]+y_sim[:,0]+y_sim[:,2])), 'y-', label='
    Congestion Index')
plt.xlabel('Time (Years)')
plt.ylabel('Congestion Index')
plt.legend()
plt.show()


plt.figure(figsize=(10,8))
plt.subplot(224)


plt.plot(t, (480*y_sim[:,0]+300*y_sim[:,2])*0.183*365, 'r-', label='C_b
    ')
plt.xlabel('Time (Years)')
plt.ylabel('C_b (Calories)')
plt.legend()
plt.show()
```