Problem Statement

Torqata, a wholly owned subsidiary of American Tire Distributors (ATD), is in prime position to help coordinate a new sustainable future for the automotive aftermarket. As a connector with data visibility and analytics capability across the supply chain, Torqata can help manufacturers, distributors, and retailers accomplish sustainability goals through a coordinated effort to recycle scrap tires.

However, there are obstacle in the way – even for the largest tire distributor in North America. Despite ATD being a juggernaut in the industry, delivering over 44MM tires annually across 80K retail locations through a network of 120 Distribution Centers (DC), ATD's foray into scrap tire recycling logistics faces two challenges:

- **Scrap Tire Forecast:** How many tires of a given size_code will be collected by a given Distribution Center every day?
- **Logistics Optimization:** How to efficiently move the scrap tires downstream to the recycler?

More importantly, ATD needs to accomplish this *while minimizing* those Green House Gas (GHG) emissions *per tire*! Are you up to the task?

Sustainability is the future of the automotive aftermarket. And to help tackle these challenges, we have invited some of the best data science teams from across the US and the world to compete in this year's ***Reinvent the Wheel 2.0*** Hackathon.

You will have 24 hours to find a solution to both challenges. May the best team win!

1. Scrap Tire Forecast

1.1 OVERVIEW

When a customer visits a tire shop for tire replacement, the tires removed (known as scrap tires) must be disposed of. ATD is exploring the logistics of scrap tire recycling – acquiring and delivering them to downstream recyclers. To effectively do this, ATD needs to know the future demand for scrap tires at their warehouses on any given day. Predicting this volume is the goal of the first challenge. We are challenging you to build a model that will forecast the number of scrap tires in each size that a DC can expect to receive from its customers daily, one to seven days out.

1.2 THE QUESTION

Given a distribution center and size_code combination, can you tell us the number of tires that will be received at the distribution center for any given day, seven days out?

1.3 DATA

1.3.1 Historic Tires Received (Challenge1_train_data.csv)

A CSV containing details about all the tires received at the distribution centers. You are given their historical data from September 20, 2020, to September 19, 2022.

date	dc_name	zip_code	size_code	retail_price	total_tires
2020-12-05	Sacramento	81235	2253521	143.56	5
2020-12-05	Sacramento	81235	2653519	281.59	14
2020-12-05	Bakersfield	31256	2253521	145.62	2
……	……	……	……	……	……
2020-12-07	San Jose	56734	2253521	151.07	12

Feel free to use any other external data sources (for example, weather data), on top of the data sources provided.

1.4 RESULTS FORMAT

Your task is to forecast the number of tires received at the distribution centers for each size_code for each of the days from September 20, 2022, to September 26, 2022. As an example, your predictions should look like:

date	dc_name	size_code	total_tires
2020-09-20	Sacramento	2253521	1
2020-09-20	Sacramento	2653519	5
2020-09-20	Bakersfield	2253521	12
……	……	……	……
2020-09-22	San Jose	2253521	2

Make sure your column names are the same as shown above when submitting, including case. Also, don't include any additional columns in your submission. Note that there is no time component for the date (e.g., the date is "2022-09-20", not "2022-09-20-2022").

1.5 EVALUATION

We will calculate the WAPE (Weighted Absolute Percentage Error) as our metric to calculate the final score. WAPE is calculated across DC-Size Code for each horizon, i.e., a single WAPE number per day. The final score will be an average WAPE across the horizons; lower is better. Here's the equation to calculate the score.

$$\text{Score} = \frac{1}{7} \sum_{September\ 20}^{September\ 22} \left(\frac{\sum(abs(actual - predicted))}{\sum(abs(actual))} \right)$$

Here is an example of how WAPE is calculated. We will also provide you with a python function to calculate WAPE. This will be available along with the dataset you will receive.

A WAPE score of 10 implies an error of 1000%. If your score is more than 10, we will cap it to 10 to normalize the scores for the leaderboard.

Check the SUBMISSION section for submission procedure.

2. Logistics Optimization

2.1 OVERVIEW

In Challenge #1 we forecasted the number of tires we expect to be at each DC. In this challenge we would like to determine the best method to move these tires from the DCs to the recycler for a single day, such that we minimize the amount of emitted greenhouse gas (GHG).

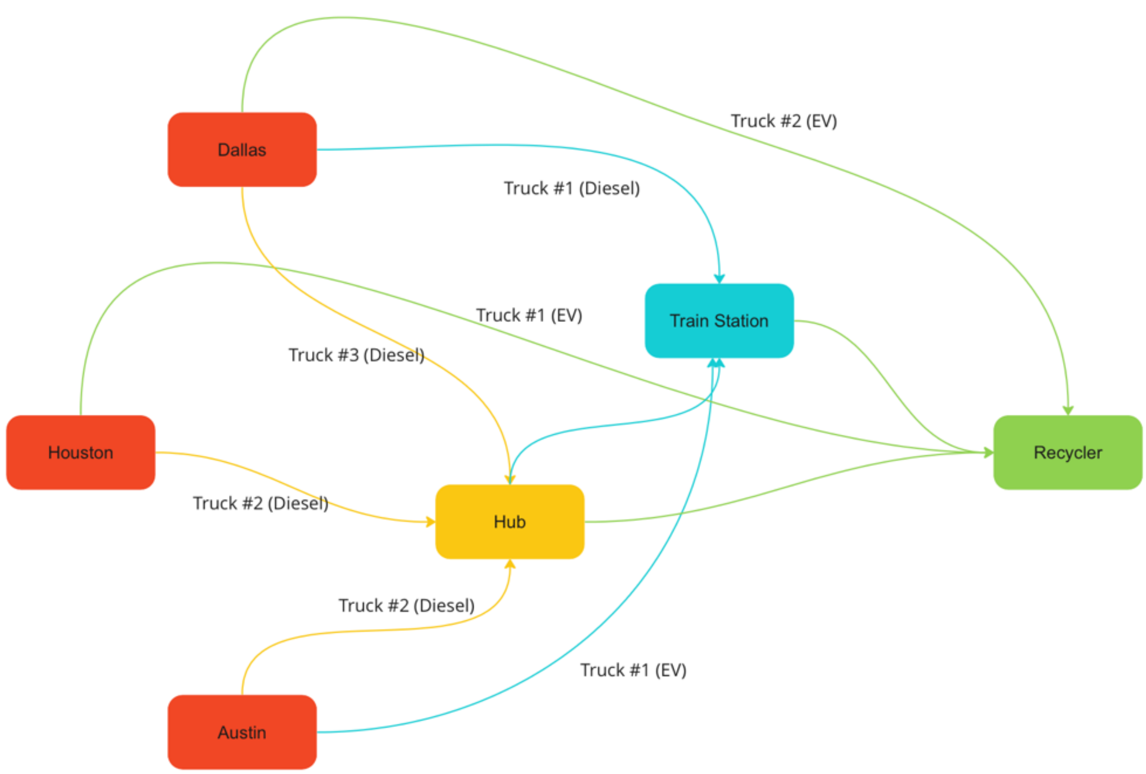

Figure 1 – routes to deliver scrap tires

DCs collect and store scrap tires from retailers daily. Depending on the location of the DC, tires may be shipped to the final recycler via one or multiple of the following methods:

- ***DC to Hub to Recycler***
  From the originating DC, the scrap tires are transported via truck to a larger warehouse known as a "Hub". A Hub collects tires from multiple DCs which are then moved via rail or truck to the recycler
- ***DC to Train Station to Recycler***
  From the originating DC, the scrap tires are transported via truck to a train station, and then make the last leg of their journey to the recycler via rail.
- ***DC to Recycler***
  From the originating DC, the scrap tires are transported via truck directly to the recycler.

For each of the above three methods of the journey, different types of trucks can be used. Each of these trucks have limitations and GHG/tire/mile associated with them.

Keep in mind that for your optimization, **you are only deciding which truck(s) to take for the first leg of the journey for each DC**. For example, if you decide to transport some of the scrap tires from the originating DC to a Hub or the Train Station, you don't need to worry about how those tires will be transported to the recycler. We will calculate the additional GHG required to do so and add that to your final GHG value (see details in section 2.3.5).

The Hub, Train Station, and the Recycler are shared among the DCs, i.e., all the tires transported from any originating DC to a recycler will be shipped to the same recycler. This is important when you want to account for the capacity of each corresponding destination (see details in section 2.3.4).

2.2 THE QUESTION

For this challenge, we have three DCs, one hub, one train station, and a recycler. Your task is to minimize the amount of GHG produced while transporting tires from DCs to the Recycler for a single day, subject to the constraints listed under the data tables below.

2.3 DATA

You will receive two sets of data for the optimization challenge: "Public Data", and "Private Data".

- "Public Data" – which you can use to build your solution and submit the results used for the public leaderboard.
- "Private Data" – which you can only access during the final two hours of the challenge, used for the private leaderboard (and thus winning result).

Make sure that your code for optimization is generic enough to accommodate for any number of DCs for the "Private Data"; i.e., don't assume that you will have the same number of DCs in the "Private" data as compared to the "Public" data.

2.3.1 Forecast Scrap Tires (Challenge2_total_tires_public.csv)

A CSV containing the forecasted scrap tire volume data for each DC.

dc_name	tires
KANSAS CITY	1126
ST LOUIS	929
SPRINGFIELD	436

2.3.2 DC Fleet (Challenge2_vehicles_public.csv)

A CSV containing the vehicle information for all the vehicles available at each DC to transport the scrap tires to any of the first leg destinations. A vehicle can be diesel powered or electric.

- Each vehicle emits GHG per mile depending on factors like type, size & weight of the vehicle. We have labeled this as *base_ghg_per_mile*.
- As we start loading tires onto a vehicle, its weight increases and hence will produce extra GHG. Column *extra_ghg_per_tire_per_mile* represents this additional cost.
- Each vehicle has a maximum tire capacity that it can transport (*max_tire_capacity*). Make sure the number of tires shipped on your result is less than or equal to this capacity.
- Furthermore, each vehicle can only be used once.

dc_name	vehicle_name	vehicle_type	base_ghg_per_mile	extra_ghg_per_tire_per_mile	max_tire_capacity
KANSAS CITY	DESDAMONA	EV	26	0.00405	260
KANSAS CITY	ELEANOR	EV	26	0.00405	260
ST LOUIS	GINA	EV	26	0.00405	260
……	……	……	……	……	……
SPRINGFIELD	ROCKY	DIESEL	23	0.03915	335

2.3.3 Distance Matrix (Challenge2_distances_public.csv)

A CSV containing distance between the originating DCs and the different destinations. The distance is in miles. You will need this table to calculate total GHG in combination with table present in section 2.3.2 (where GHG numbers are in terms of per mile).

dc_name	destination	distance
KANSAS CITY	HUB	5
ST LOUIS	HUB	248
SPRINGFIELD	HUB	163
……	……	……
SPRINGFIELD	RECYCLER	869

2.3.4 Destination Capacity (Challenge2_capacity_public.csv)

A CSV containing the maximum number of tires a location can hold. Make sure total tires shipped from any any/all DCs do not exceed the *max_tire_capacity* of the location. Also be mindful that Hub, Train, and Recycler is shared among all DCs. To further clarify, if you ship 500 tires from

Kansas City and 300 from St. Louis to TRAIN, then TRAIN is full. Your tires from SPRINGFIELD cannot go to TRAIN.

location	max_tire_capacity
HUB	2000
TRAIN	800
RECYCLER	1500

2.3.5 Second Leg GHG Cost (Challenge2_second_leg_GHG _public.csv)

The solution you submit only takes tires to the first leg: Recycler, or Train station, or a Hub. Tires that were moved to the recycler do not incur second leg GHG cost as it has reached its destination. However, if the tire ends up at a train station, or hub, or remains at the DC, then we will have to move it to recycler eventually and it will incur some additional GHG for this second leg of the journey.

The CSV contains base GHG per mile and additional GHG per mile per tire you will incur for any tire you leave at any location other than the recycler.

location	base_ghg	additional_ghg_per_tire
KANSAS CITY	1016640	252.59
ST LOUIS	1183680	310.5
SPRINGFIELD	1122240	289.2
……	……	……
RECYCLER	0	0

2.4 RESULTS FORMAT

Your task is to decide, for each DC, what vehicle(s) you will use to ship how many tires to what destination. As an example, your submission may look like:

dc_name	vehicle_name	destination	number_of_tires_shipped
KANSAS CITY	DESDAMONA	RECYCLER	260
KANSAS CITY	ELEANOR	TRAIN	230
……	……	……	……
ST LOUIS	GINA	HUB	200

2.5 CONSTRAINTS

For challenge #2 there are some scenarios where your submission would be invalid (i.e., they violate the constraints). Here is the list of those constrains:

- You transport more tires than the capacity of the destination. (Section 2.3.4)
- You transport more tires than the capacity of the vehicle. (Section 2.3.2)
- You use the same vehicle multiple times. (Section 2.3.2)
- You use a vehicle from a DC that is not assigned to the DC. (Section 2.3.2)

- You use a vehicle but do not ship any tires.

2.6 EVALUATION

Let us consider a single DC which has the following number of scrap tires:

dc_name	total_tires
Houston	2000

After optimization, let's assume you sent us the following result (and all the constraints are already satisfied).

dc_name	vehicle_name	destination	number_of_tires_shipped
KANSAS CITY	DESDAMONA	RECYCLER	260
KANSAS CITY	ELEANOR	RECYCLER	260
KANSAS CITY	TERMINATOR	TRAIN	335

For the first row in the results above, the total GHG can be calculated as:

*Distance from Kansas City to Recycler (759) * base_ghg_per_mile (26) +*
*Distance from Kansas City to Recycler (759) * extra_ghg_per_tire_per_mile (0.00405) **
*number_of_tires (260) = 759 * 26 + 759 * 0.00405 * 260*
= 20533.227 GHG

We can calculate GHG similarly for all the rows above:

dc_name	vehicle_name	destination	number_of_tires_shipped	ghg
KANSAS CITY	DESDAMONA	RECYCLER	260	20533.227
KANSAS CITY	ELEANOR	RECYCLER	260	20533.227
KANSAS CITY	TERMINATOR	TRAIN	335	794.535

Total GHG emitted based on the table above = 41860.989

On top of this, you will also incur additional GHG based on number of tires at the DC and the Train station. (Section 2.3.5)

Tires left at the Kansas City DC: 2000 – (260 + 260 + 335) = 1145
*Additional GHG for tires left at the Kansas City DC: 1145 * 252.59 + 1016640 = 1305861.5*
*Additional GHG for tires left at the Train Station: 335 * 66.56 + 224000 = 246279.6*

Hence, total GHG for the solution = *41860.989 + 1305861.5 + 246279.6 = 1594020 GHG units*

To normalize the scores for challenge #2 for the public leaderboard we have capped the minimum and the maximum GHG from between 0.5 million and 10 million GHG units. If your submission is outside of these bounds, we will cap it to the nearest threshold value.

2.7 LAST HOUR RUSH!

Two hours before the Hackathon deadline, you will be provided with a password to unlock the private data via Slack. You are supposed to run your code based on this new dataset and submit your solution.

The solution submission format will be the same as public data. The only thing different is that all files on the folder will be labeled "*_private.csv" compared to "*_public.csv".

SUBMISSION

For any submission, you will be using REST endpoints to submit your solutions. For each of the challenges, there will be two types of submission routes available:

1. Validity: This route should be used to make sure your submissions are valid to be scored. This route checks whether your column names are correct, you have all the columns present, if there are nulls present, among many other checks. You can use this route as many times as you wish.
2. Scoring: This route should be used to submit a solution to be scored. **Once your submission is successfully scored for a challenge, you will not be able to submit another solution for that challenge for another 30 minutes.** Thus, please make sure your result passes the validity checks before submitting it for scoring. You will need to submit challenge #1 and #2 results separately.

Your team will receive a unique API key in your private Slack channel that must be passed alongside the data for any submission using the REST endpoints. **If you run into any issues, then you can reach out to any volunteer at the event either via Slack or at the Torqata booth.**

Python instructions:

If you are using Pandas in python to submit your results within a dataframe called *df_submission*, then you may submit using the code below. Note in this example we are submitting our dataframe to be validated for the forecast problem.

```
import requests

url = "https://scoring-app-uuzeqpiufa-ue.a.run.app/forecast/validate"
payload = {
      "team_key": "abcd123",
      "data": df_submission.to_dict(orient="records")
}
```

```
response = requests.post(url, json=payload)
print(response.status_code, response.content)
```

In case of a successful submission where all the validation checks have passed, you will get a (200, {“message”: “success”}). If any check fails, you will see an appropriate error within the message.

Challenge #1 vs Challenge #2:

The submission procedure for both challenges is identical. You will receive URLs to validate and submit your result. Specifically for challenge #2, the URLs you will be using for the first 22 hours of the challenge will be used to calculate public score. For private score you are provided with another two separate URLs. **You are allowed to submit your results for the private data for challenge #2 for a maximum 2 times; and there is no 30 minutes restriction between those 2 submissions.**

URLs for submission:

Just prepend https://scoring-app-uuzeqpiufa-ue.a.run.app/ before each endpoint and you should be good to submit, e.g., https://scoring-app-uuzeqpiufa-ue.a.run.app/forecast/validate.

- forecast/validate
  - Validation route for the forecast challenge
- forecast/submit
  - Submission route for the forecast challenge
- optimization/validate
  - Validation route for “public” optimization challenge
- optimization/submit
  - Submission route for “public” optimization challenge
- optimization2/validate:
  - Validation route for “private” optimization challenge – open after 22 hours
- optimization2/submit
  - Validation route for “private” optimization challenge – open after 22 hours

LEADERBOARD

A leaderboard will be maintained throughout the competition. The leaderboard will show the **latest score** for the “Public Leaderboard Set”. The leaderboard will refresh every 5 seconds.

IMPORTANT: The top 5 teams for final presentations will be selected based on the **LATEST** scores of the “Private Leaderboard Set” – you must keep track of your best scores for both challenges and re-submit before the end of the event!

- **Public Leaderboard Set:**
  - **Challenge #1 (forecast)**: Your submission will be equally split into “Public” and “Private” component and then compared against the true values. The scores will then

be normalized to a scale of 0 to 1 across all the teams. This will weigh 60% of your final score.

- **Challenge #2 (optimization):** We will use the “Public” optimization dataset for public leaderboard. The submission values will be normalized to a scale of 0 to 1 across all the teams. This will weigh 40% of your final score.
- **Overall Rank:** The final score (and rank) will be calculated based on the weightage mentioned above. Higher score = Better predictions.

- **Private Leaderboard Set:**
    - **Challenge #1 (forecast)**: We will use the “Private” component of the split data. All other processes to score will be the same as for the public leaderboard.
    - **Challenge #2 (optimization):** We will use the “Private” optimization dataset for the private leaderboard which is only allowed to be used for the final 2 hours of the competition. All other processes to score will be the same as for the public leaderboard.
    - **Overall Rank:** The final score (and rank) will be calculated based on the weightage mentioned above. Higher score = Better predictions.

DELIVERABLES

As the competition closes out on Saturday, you are asked to submit the components laid out below to be considered by the judges. We will collect all the source code and presentation through each team’s private slack channels.

1) 3-page PowerPoint deck:
   Every team is asked to provide a 3-page PowerPoint deck. The PowerPoint deck should answer the following prompt: “Tim Eisenmann, the CEO of Torqata, just asked you in his office and wants to get a 3-minute update on the scrap tire forecast and the optimization project. Tim is not a data scientist and does not have a technical background, he prefers clean and concise communication in language that he can understand with visualizations where it makes sense.”

   Please cover the following content:
   Page-1: Approach taken
   Page-2: Key insights and results
   Page-3: Improvement opportunities and recommendations
   Pages 4,5: Supporting visualizations for pages 1-3 (this could be pictures, gifs, graphs – whatever gets your point across)

   You will be provided with a PowerPoint template that you can use for your deck.

2) All the code, notebooks:

Please include all the relevant code, notebooks that you produced and used to get your predictions for both the challenges. This will be very important for us to understand the approach you have taken and decide on who has the most unique approach (check details below)

Based on the private scoreboard, the judging panel will announce the TOP 5 finalists without disclosing the ranking of the teams. Each finalist team will get the opportunity to present their work using the power point deck to the judging panel as well as the evening guests. Presentation time is five total minutes from on to off stage and will be strictly enforced. Order of the presenting teams will be random. And remember, use language that is suitable for a non-technical, executive audience.

What success looks like

The top 5 teams based on mathematical accuracy will present in no specific order for eligibility on the Best Presentation award. The **top 3 winning teams** will submit an accurate forecasting model for scrap tire supply, and an optimization strategy that minimizes emitted GHG. These awards will be given based purely on mathematical accuracy.

Additional awards

In addition to the main challenge, we will be awarding prizes to teams for categories.

1) **Most unique approach**:
   The team with the most unique approach to create the solution for both the forecasting and optimization challenge will get this award.

2) **Best presentation**:
   To award the team that was able to articulate their approach and results in a very clean and concise language that anyone from a technical or non-technical person can understand.

3) **Best use of Cloud**:
   Judged by Google employees themselves, they would like to see how you utilize/integrate various cloud services provided by GCP to create/present your solution.