ISYE 6501 | Final Project

Introduction

The ORION (On-Road Integrated Optimization and Navigation) Project that UPS (United Postal Service) has implemented has greatly improved the P&D (pickup and delivery) operations at UPS. Using ORION, UPS is able to provide its drivers with an efficient route in which packages should be delivered each day. This strategy allows UPS to save money, reduce fuel consumption, reduce CO_2 emissions, increase the amount of serviced customers each day, and reduce the amount of drivers and vehicles needed to service its daily customer base.

But, how exactly was this ORION system devised? Here, I will provide a potential solution including what data was used and collected, what three models were used and combined, how those models were implemented, and how UPS may be maintaining said models.

This ORION system can be broken down into three parts/models. First, use a clustering model to split destination assignments into zones. Next, use a time series forecasting model to determine how long it will take to travel between destinations in zones. Lastly, use an optimization model to determine both how many drivers are needed for each zone and the schedule for each driver (driver assignment for each destination and destination order for each driver).

Stage One: Clustering Package Delivery Zones

First, we should use clustering to group the destinations that UPS services into zones. This will allow each zone to be treated independently and effective optimization to be performed. For example, if UPS services all the destinations in Atlanta, it doesn't make sense to treat the entirety of Atlanta as one zone and optimize for the entire city. Rather, a zoning system (i.e., each neighborhood in Atlanta is a zone) should be used to split up work effectively and reduce time and distance between deliveries.

But, how do we create these zones? The best way to do so is to cluster the homes based on various data about the locations and delivery history.

Data Needed

- Location: A way to quantify information about each destination is converting each address to latitude and longitude. This way, a distance metric between destinations can be easily computed.
- Delivery volume: We also want to determine how much volume that each destination needs delivered on average daily. Since each driver can only carry a fixed amount of volume in the delivery truck at one time, it is important to understand how much volume

each destination requires so that zones do not require an overflow of volume in each truck or an excess number of trucks to be used.
- Destination type: Is this destination a residential, business, or other type? This allows us to make imputation decisions for missing volume.
- Ideal number of zones: Because we are using clustering, we must determine exactly how many clusters to use when determining zones. Of course, this can be optimized or determined using some model or algorithm. However, the best way to decide this may actually simply be using an expert choice or heuristic given by business professionals within the company.

Data Collection

UPS has been servicing clients since 1907 and thus have decades of prior customer data available for use. So, data about delivery volume based on location is easy to gather from the archives. However, location data (specifically latitude and longitude), destination type, and road information must be gathered using some sort of map library such as OpenStreetMaps or Google Map API. Ideal number of zones should be gathered by either surveying or specifically asking experts at UPS for their pick. Another option is to try the overall model (combining all three stages) using various ideal number of zones, however this would be more computationally and time expensive.

Of course there are scenarios where new customers are picked up or new locations are built, which would require data imputation or some way to handle missing data. The only feature that would actually require data imputation is actually delivery volume. However, we can use a regression model using location and destination type to impute predicted delivery volume. This is the smartest method since it allows us to use information about similar destinations within the area in order to determine what the delivery volume might be. It should be noted that this may increase the bias of the clustering model quite a lot (and may either wildly over or underestimate the volume), however it would allow us to make some sort of assumption based on facts rather than simply providing a mean or removing the data point altogether.

Model Creation

Other than the initial model of regression used for data imputation of data volume, which uses location and destination type as features and delivery volume as the response, the main model for this stage is clustering. Of course, there are different types of clustering that you can use. For this stage, we should use hierarchical clustering. This is the ideal type because it allows us to cluster without defining the number of clusters. If it turns out that the expert opinion for the ideal number of zones is incorrect or we wanted to test multiple values, we could easily do so without having to rerun stage one from scratch. When clustering we want to use location and delivery volume as features and use the ideal number of zones as the number of clusters.

Model Maintenance

The amount and locations of the destinations that UPS services changes very rarely. However, it is always possible that a consumer or business decides to switch between using or not using UPS or a new destination is built. Therefore, it would be important to run this stage at least a couple times throughout the year. Perhaps a good suggestion would be once at the end of every fiscal quarter, since the end goal is to in fact reduce cost for UPS and customer information may have changed during the quarter as those customers made adjustments following the previous fiscal quarter.

Furthermore, it should be noted that this stage should be run once per midsize region. For example, we wouldn't want to run it on the whole world or country at once. Instead, maybe run it once per state or even county if that is the preference.

Stage Two: Time Series Forecasting of Travel Times

Next, it is important to understand exactly how long it takes to travel from place to place. Even though we have determined zones in stage one, we still want to determine, daily, the most efficient route to take within each zone. Based on weather patterns, traffic information, road information, etc, exactly how long does it take to travel from destination to destination?

But, how do we predict travel time each day? The best way to do so is using time series forecasting in order to predict based on historical data. Since we have historical and timestamped data, we can forecast travel time based on a variety of features and the specific timestamp to predict.

Data Needed

- Travel time: Historically, how long does it take to travel between any two destinations. This should be collected on a daily basis.
- Estimated travel time: It is important to understand how long it takes to travel from destination to destination based map estimates. This is especially useful when destinations are new, haven't been traveled on a certain route, or are just missing data about actual travel time.
- Datetime of travel: What time did the driver leave destination one in order to travel to destination two. Depending on the two destinations, the time of travel could make the travel time wildly different. For example, traveling on certain roads immediately prior or following rush hour could greatly increase the travel time of that specific stop or the overall route. Additionally, certain days or times of the year can also affect travel time.

Travel during the summer may take longer than travel during the winter due to the increased number of people leaving their house.

- Road information: Various road information such as construction, accidents, closings, or new roads should be considered. This should be marked as whether there was a

Data Collection

Of course, it may be the case that all of this information may not already be collected together. It may be the case that travel time is already collected, but maybe not previous weather information for those specific days. So, prior to using this system for the drivers, we should allow the drivers to drive for a certain amount of weeks (whatever the company can allow based on their budget) in order to collect information about routes. Alternatively, we can simply use the model with default values and update on the fly day by day as we collect more information. The latter model may actually provide better performance and allow the company to use the impact of the efficient routes sooner rather than later. However, it should be noted that the latter method may fall victim to having a buffer of model performance and thus slightly inefficient routes whenever certain trends start or stop. This issue can be negated using one of the two following options: using expert advice on trend/seasonality dates or waiting for the trends to occur multiple times and learning when the trends/seasons start and stop.

Travel time should be collected in the above manner. Estimated travel time can be collected each day or every so often by using an API such as Google Map API. Google Map API even has the functionality of estimating travel time at certain scheduled times of the day. The datetime of travel will be inherently collected when collecting the travel times. Road information should also be collected using an API as it would be difficult to collect the information for all roads manually.

Model Creation

For this stage, we can use the features listed above and an ARIMA model to predict future values of travel time. It should be noted for data collection purposes that ARIMA requires at least forty past date points.

This stage should be used to create an overall graph between each destination. Using the response from the forecasting (travel time between destinations) as the weight, the graph will contain useful information for stage three.

Model Performance

The ideal scenario would be to run this stage in real-time, continuously, to constantly provide each driver with the ideal route and constantly adapt to any change in weather or road information. However, this method would be wildly inefficient as it would require data collection and model creation to constantly occur. On the other hand, if we run it every so often, such as once a week or month, we run into the issue of not having updated information about roads, weather, etc.

So, we should use the middle ground of running this stage once every day. Given that stage three should be run once every day, we want to run it the same amount in order to provide updated information to stage three.

Furthermore, it should be noted that this stage should be run once per zone and thus the number of graphs generated should be equivalent to the number of zones.

Stage Three: Optimize Driver Routes and Assignments

In stage three, we need to put everything together. Using the zones from stage one and the graph for each zone from stage two, we need to determine how many drivers are required for each zone, which destinations in the zone that those drivers have on their route, and the most efficient route they should take.

Data Needed

- Travel time information: This is simply the graph generated during the computation of stage two. This useful graph will allow us to calculate time of travel for the route.
- Number of drivers available for each zone: How many drivers can carry the burden of each zone?
- Delivery volumes: What is the delivery volume for each destination?
- Max delivery volume: What is the maximum volume each driver can carry?
- Hours working: How many hours is each driver working that day?

Data Collection

For this stage, the data collection is very easy and should be mostly contained within UPS's HR system. The travel time information is held within the graph generated in stage two. The number of drivers available and hours working of each driver will need to be accessed from the HR system. The max delivery volume should be manually or automatically depending on what truck the driver is driving. The delivery volumes of each destination should be help in a UPS system that is already implemented when tracking packages.

Model Creation

For this stage, we need to use an optimization model with the following variables, constraints, and objective function. The general idea is that reducing the travel time for the zone overall will reduce fuel consumption, CO_2 emissions, and save UPS money in the long run.

Variables:

- $T_{i,j}$: travel time between destination i and j
- V_i: delivery volume of destination i
- TT_x: total travel time for driver x
- DV_x: volume carried by driver x
- DD_i: driver assignment of destination i

Constraints:

- The total travel time for driver x should be less than or equal to daily hours for driver x for every driver x
- Sum of driver assignments should equal to number of destinations for the zone
- Sum of all assignments' volumes should be less than or equal to max driver volume

Objective Function:

- Minimize the total travel time for the zone

Model Maintenance

As discussed above, this stage (following stage two) should be run once a day. Specifically, this stage should be run immediately in the morning prior to each delivery driver starting their routes so that they have updated information and can efficiently complete their route.

Furthermore, it should be noted that this stage should be run once per zone.

Closing

Integrating all these models together, once per day, will allow an effective ORION system to be implemented and provide each UPS driver with efficient routes.

References:

Postal Service, U. S. (2021, January 14). *UPS on-road integrated optimization and Navigation (Orion) Project*. INFORMS. https://www.informs.org/Impact/O.R.-Analytics-Success-Stories/Optimizing-Delivery-Routes