**Delivery date prediction**

**Problem statement:**

The logistics team at Olist uses heuristics to provide an estimated delivery date for the orders placed. It is very conservative about the delivery dates. As a result, it is able to deliver the products much in advance. Although this is beneficial for the logistics team’s 'on time delivery' KPI, it is not favourable for the CMO. He found that on average, the estimated time to deliver products that are given to customers is twice that of the actual delivery time. Such a high expected delivery time is driving away Olist's customers. So, the CMO is looking to use ML to get a far more accurate expected delivery date.

**Proposed Solutions:**

1. ML solutions -   
   1. Use one regression model using features like destination location and source location, and historic delivery date, item size, proximity to distribution hubs, seller, etc to predict the estimated time of delivery. Add the delivery time to the order date and calculate the estimated delivery.
   2. Divide the entire delivery process into individual stages of transport.
      1. Estimate time for getting the goods from the vendor.
      2. Estimate the time that the item will stay in the warehouse.
      3. Estimate the time required to send the item from the warehouse to the delivery location.  
           
         Build individual models for estimating these times, add the time to get the overall delivery time and add it to the order time to get the estimated delivery date.   
         The advantage of building separate models is more fine control over the processes and better prediction.
2. Non - ML solutions
   1. A rule-based approach to predicting delivery times. There are set times for each step of the delivery process such as set time for getting goods from the warehouse to the transport hub, from the transport hub to the next transport hub etc.   
        
      Each of these times is mapped to the nearest shipping cutoffs wherever appropriate. Fixed additional processing times are added for weekends and holidays. Thus, the rule-based model doesn’t adapt based on recent performance changes and is designed based on heuristics.

Selecting First ML solution - because the data points for individual steps are not present.

**Benefits of Proposed Solution**

**Process improvements -**

Predicting delivery dates more accurately will not impact any process in the e-commerce setup.

Reduction in Inventory costs.

**Monetary benefits -**

The monetary benefits of accurate delivery can be divided into two streams

* Reducing customer churn because of accurate delivery date prediction. Hence, increasing the revenue of the company.   
  1. Reduction in the number of people who drop out - 10% (Assumed)
  2. Number of customers daily - 272.44   
     (total number of orders in 1 year = 99442 / number of days in a year 365)
  3. Average order value - 100 Brazilian real (Assumed)
  4. Revenue increased by = 0.1 \* 272.44 \* 100 = 2724.4 per day.
* Inventory management   
    
  Predicting the delivery date correctly will reduce the pressure on the supply chain. The through out rate of deliveries of the entire system increases but predicting the delivery date correctly. The warehouses will have to hold lesser inventory.
  1. Reduction of time items stay in warehouse = 40% (assumed)
  2. In the same time frame warehouse will be able to hold more items = 40% (assumed)
  3. Warehouse cost per item delivered will reduce = 25% (assumed)
  4. Initial warehouse cost per item delivered = 5 BR (assumed)
  5. Total savings per day = 272.44 \* 0.75 \* 5 BR = 1021.65 BR

**Summarise the DS approach**

The estimation of accurate delivery dates is a regression problem to be solved. You use various data to estimate the time needed for delivery, then add the time to order date to get the right delivery date.

Delivery date prediction is also a kind of balancing act between competitiveness and accuracy.   
You can always have long delivery dates and always be accurate, but might lose on sale to some competitor who can deliver quickly.

Or you can have extremely short delivery time promises and disturb the customer sentiment. Hence add a buffer  
  
[There is a constant trade-off between being accurate and being competitive and, of course, we would aim to optimize both.](https://towardsdatascience.com/delivery-date-estimation-5aff1a0ff8dc)

**Limitation**

1. RMSE might be high
2. High complexity model - non linearity , thus non linear model might be needed - is the team capable or not.
3. Think of edge cases - data for special cases is not available

**Define Appropriate Success Metrics**

1. Early delivery rates
2. Late delivery rates
3. Churn rate after estimated dates is shown

**Prioritising Use Case**

Refer to the use case prioritisation framework linked [here](https://docs.google.com/spreadsheets/d/1EGoVubzdetsv8YhrKeenSq_uaBSubtsgZm2WydL1U1I/edit#gid=704154787).

**Cite references**

1. Similar delivery date estimation application form example on amazon blogs. <https://aws.amazon.com/blogs/industries/how-to-predict-shipments-time-of-delivery-with-cloud-based-machine-learning-models/>
2. Paper suggesting various solutions for delivery date production <https://arxiv.org/pdf/2105.00315.pdf>

**Sentiment Analysis**

**Problem statement:**

The Chief Marketing Officer at Olist wanted to understand the experience of the customers based on the reviews received after the delivery of the orders. He also wanted to identify the areas of improvement based on these reviews. He had heard that NLP can be used for sentiment analysis and topic modeling, which will be useful in finding topics in customer reviews. However, he was also cognizant of the fact the customer reviews are in Portuguese, whereas the NLP algorithms are not so sophisticated in Portuguese.

**Proposed Solutions:**

1. ML Solution
2. Non ML solution

**Benefits of Proposed Solution**

**Summarise the Solution**

**Prioritising Use Case**

**Cite references**