Identify churned/inconsistent people:

Preprocessing

* Time split for the data (train vs test): ignore 2020
* Looking for loyal turned inconsistent
* How to define consistency? If person A used to come every month in the first half of 2018, but in the second half of 2018 comes every 2 months (best case) or time interval is undefined (worst case), they are considered inconsistent.
* We will drop customers who only have transaction frequency of less than say 10% (x%) of the population frequency distribution.
* Drop top and bottom 10% of items (give the reasoning)
  + Top 10% should not be recommended as there is little profit margin to earn from it if they were recommended
  + Bottom 10% as it is very rarely bought.
* Why new and churned customers?:
  + Minimise loss as much as possible (churned) → give the percentage of people who we have currently lost
  + For new customers → we have to find from the net, why new customers are impt

Decision Tree (if decision tree still good):

* Factors that make people churn/inconsistent: identify those features that fine a churned group are the nodes in the tree
  + Picture of the tree
  + Statistics: confusion matrix, highlight FNR
  + Top node determines that this item causes a split between the churned and not churned, it should not be recommended
  + Recoup our losses: we increase the basket size by recommending items that are usually bought by non-churners in the same bmi and income level
  + Have to identify out of the 12 clusters, high income + high bmi (show our budget and calorie constraint). Another group, suggestion, is that we identify a group that is very different (in terms of transaction data) from its corresponding non-churner cluster.

Logistic Regression

* + Remove items that are not as significant (feature selection methods: RFE, chi^2, fisher score)
  + Statistics: confusion matrix, highlight FNR
  + Factors that make people churn: identified by the coefficients
  + Recoup our losses: we increase the basket size by recommending items that are usually bought by non-churners in the same bmi and income level
  + Have to identify out of the 12 clusters, high income + high bmi (show our budget and calorie constraint). Another group, suggestion, is that we identify a group that is very different (in terms of transaction data) from its corresponding non-churner cluster.

Neural Networks

* + Show either of the first two examples (logistic/decision tree), compare the FNR/accuracy etc metrics and say its better
  + Use wide & deep learning IF it is actually better, we have to compare FNR.
  + Recoup our losses: we increase the basket size by recommending items that are usually bought by non-churners in the same bmi and income level
  + Have to identify out of the 12 clusters, high income + high bmi (show our budget and calorie constraint). Another group, suggestion, is that we identify a group that is very different (in terms of transaction data) from its corresponding non-churner cluster.

New Customers:

* A recommendation for a new customer (which was pulled out from the system)
* After the first recommendation, we feed the first transaction into the system, and then churn out the second recommendation.
* 1st vs 2nd vs 5th transaction: recommendation system show the change in recommendation (won’t change most likely if we use an item based recc system?)
* If there’s a possibility for RNN, focus on it

NEW DIRECTION (if all RNN/ new customer doesn’t work out)

* Loyal customers:
  + Interesting service to keep them hooked on to our supermarket / app
  + Using a regression tree, to predict how much they are willing to spend?
* New customers:
  + Hard to justify better find good reason
  + Using a **regression tree on non-essentials**, predict how much a customer with a certain profile will spend.
  + Split the train/test: train is based on 2017/2018 data, test are NEW customers in 2019.
  + Metric: Mean Squared Error for the money spent in a transaction.
  + The y labels (the amount of money they spend in 1 transaction) for the existing customers (need to ensure that the staples have been removed beforehand)
  + Report the MSE test loss and say its good if it is
  + Once we know how much they are willing to spend, then we come up with the profile based recommendation system → find the top n items that can fit into their basket.
  + Average pack size / volume generate from the dataset, and then suggest it as part of their basket. Constrained the money based on that
  + Suggest another item as part of their basket to have another basket.
    - Eg. first basket based on top 10 items
    - Eg. bmi based, second basket is a calorie constrained basket, if it doesn’t hit the money constrained.

For a churned customer, let’s assume that the customer behavior may change before they churn (i.e. they use to order monthly, but starts to order at irregular intervals before they dropped off, this may be a sign that they are churning)

Can we come up with a function, that given a customer id and their recent frequency, can we predict it they will be likely to churn in the following weeks?

Logistic regression ---> predict if someone will churn

If that someone is about to

New Customers:

* Explain why focussing on them is important:
  + Trying to solve the Cold-start problem → wherein a recommender system isnt provided with enough information
  + Hence, to embark on this problem we use new customers whose data is severely limited and see what is the best way to solve a problem with limited information
  + The number of new customers seems to increase over time, as well the proportion they make up of the entire population (graph)
  + We can state some statistics that shows how important new customers are for business growth (??)
  + Introduce the idea:
    - Predict firstly the number of items they would possibly buy (i.e basket size) based on demographics
    - Predict the what items they would buy based on customers that are similar to them
  + How do we do this?:
    - Use 2 decision trees
    - For number of unique item categories in a regression tree
      * Choose a customer ID that we want to present
      * If depth is 4 and below then we put it there
      * If its more than 10, dont put the picture, just show the bottom leaf nodes leading to the regression value
      * Put the graph for max depth to explain how we choose max depth to prevent overfitting.
      * Final R squared or MSE statistic (whichever is better)
    - And for the other one its a one vs many multilabel decision tree
      * We cant have a picture
      * Explain the statistics and explain how it works → explain how the algo works with a bunch of trees
      * F1 score the fact that we remove certain items from the list → algo is not very accurate. We believe that part of the reason is due to the low support of these items.
      * Examples of low support/f1 score items → probably due to data sparsity
      * Given the unique items constraint and the calories constraint, we can then recommend 2 baskets of items. One based on highest probability and one that is more focused on lowering calories.
      * Show an example using StreamLit. We can have an overweight BMI person and a normal BMI person.
      * **Last important pointer: WHAT ARE THE PROFIT MARGINS.**
  + Show the branch leading to the basket size for the high bmi customer, show the branch leading to the

Buy too many cooking sauces = most likely you will churn, so change the cooking sauces? We have to conduct experiments on the different cooking sauces to see if this factor becomes less important over time.