Sampling Approach

We took the following steps to create sample datasets. These steps were the same for both submodel 1 and submodel 2. No global sampling was performed, and we began with the entire campaign dataset for these submodels, which comprises solicitees who were assigned to cells 1 and 2 (submodel 1), and cells 3 and 4 (submodel 2). The target variable is a binary class: 0 (non-responders), 1 (responders).

1. **Population split:** We used the population split step in the pipeline to randomly put eligible customers from our dataset into training, validation, and test holdout segments based on a specified proportion of responders and non-responders. We used 60% of the population for training, 20% for validation, and 20% for in-time test holdout. Since this is done randomly, the proportion of responders in these datasets roughly follows this distribution after splitting.

2. **Training and validation down-sampling:** A 20% down-sampling ratio was applied to the training and validation datasets. This is done to rebalance the ratio of the responders to non-responders in the target class. The training and validation datasets are from May 2021. The training and validation datasets are created by selecting 100% of the minority class and down-sampling the majority class to achieve a ratio of 80% non-response and 20% response. This is shown in *Table 1.1*, where the ratio of responders from the training and validation datasets is roughly 8:2.

   The Pozzolo method was used for probability calibration during down-sampling. The Pozzolo calibration method attempts to reduce/remove bias caused by random down-sampling by keeping the probability distribution of the classes consistent between the original dataset and the sampled dataset. Essentially, data was chosen from the majority class to be in the down-sampled population based on a probability calibration. This adjustment for down-sampling bias in the predicted probabilities by the Pozzolo method may result in better performance compared to Platt scaling in some cases [3]. *Table 1.1* shows the down-sampled training and validation data used for modeling with ~20% response rate. The seed was set at 42. Models were developed on the training data.

Table 1.1: Eligible population and response rate after down-sampling

Submodel	Population Split	Period	Eligible	Response	Response Rate
1	Training	202105	5,596	1,135	20.28%
	Validation	202105	1,991	435	21.85%
	In-time holdout (not down-sampled)	202105	240,394	392	0.16%
2	Training	202105	13,604	2,723	20.02%

	Validation	202105	4,515	901	19.96%
	In-time holdout (not down-sampled)	202105	512,062	867	0.17%

Tables 1.2 and *1.3* provide the summary of the populations before and after down-sampling. Before down-sampling, the training and validation datasets have a response rate of ~0.17%. After down-sampling, the response rate is ~20% for both datasets. Note that the number of responders didn't change, because we kept all responders and only down-sampled the non-response class. In-time test holdout is not down-sampled.

Table 1.2: Population partition before and after down-sampling – summary – submodel 1

Submodel 1	**Before down-sampling**			**After down-sampling**		
Population	# of Eligible	# of Response	Response Rate	# of Eligible	# of Response	Response Rate
Starting	1,204,778	1,962	0.16%			
Training	722,015	1,135	0.16%	5,596	1,135	20.28%
Validation	242,369	435	0.18%	1,991	435	21.85%
In-time Holdout	240,394	392	0.16%	Down-sampling not applied		

(etc.)