

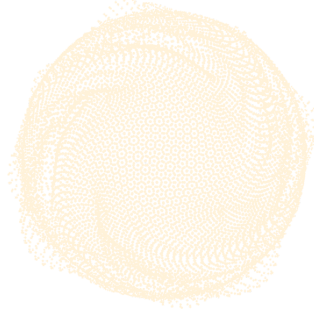
baswe.Ai Engineer Accelerator™

Companion Materials

Project brief + data dictionary for the churn pipeline build

This package contains the materials you need before starting Part 1 of the Module 5 Checkpoint:

- Document 1: Project brief from Meridian's retention team
- Document 2: Data dictionary (every feature in the dataset, with governance status)
- CSV files: train.csv (40K rows) and test.csv (15K rows) — download separately from your dashboard



baswe's

AiEngineerAccelerator

Document 1 Project Brief

From: Robin Castillo, Head of Claims (acting as Retention Sponsor)
To: ML Engineering team

Business context

Meridian Health currently loses ~24% of its members at renewal each year roughly 200,000 of our 850,000 members do not renew. Acquisition cost per new member is materially higher than retention cost, but our retention team has no systematic way to identify at-risk members early enough to do anything about it.

Today, the retention team's outreach is mostly reactive: they call members AFTER cancellation paperwork has been started, which is too late for the majority of cases. The retention director has asked for a system that surfaces likely non-renewers 60–90 days BEFORE the renewal decision, so the team can intervene proactively.

What we are asking you to build

A monthly batch model that scores all members approaching renewal in the next 90 days and produces a ranked list of the top ~5–10% most likely to not renew. The retention team's outreach capacity is constrained. They can productively contact about 7,000 members per month so ranking matters more than calibrated probability.

This is a V1. We do not need real-time inference. We do need:

- Predictions the retention team can trust enough to act on
- Honest performance numbers, including how the model does on subgroups (Medicare vs Marketplace vs Commercial members)
- A documented monitoring plan so we will know if the model degrades after launch
- A reproducible pipeline an engineer can pick up and retrain next quarter

Success metrics

Primary metric	Precision at the operating threshold the retention team can actually contact. With ~7,000 monthly contacts and ~70,000
----------------	--

	renewals per month, we are operating at roughly the top 10% risk band. Precision at that band is what matters.
Secondary metric	Recall of true churners in the top band. We want to know how many actual non-renewers we are capturing.
Subgroup parity	Performance must be reasonable across plan types. A model that works well for Commercial but poorly for Medicare Advantage is not deployable.
Calibration	Predicted probabilities should be roughly calibrated. The retention team may eventually want to act on probability bands, not just rankings.

Constraints from the AI Safety & Governance team

(This is YOU. The constraints you applied to ClaimsAssist in Module 4 carry forward to this project. The data dictionary on the next page reflects these constraints.)

- Race, gender, mental health diagnosis, and dual-eligibility status may not be used as model features under our internal Responsible AI policy.
- State and rural-ness ARE allowed as features (geography is operationally relevant), but subgroup performance must be reported across these dimensions.
- Any feature where we cannot articulate WHY it should be predictive of churn should not be used.
- The model must be auditable: a retention rep should be able to ask "why was this member flagged?" and get a defensible answer, not a black-box score.

Deployment expectations

- Monthly batch run, not real-time.
- Output is a CSV with member_id, predicted probability, and the top 3 contributing feature reasons per prediction.
- The retention team will pull the CSV from a known location
- Pipeline must be re-runnable end-to-end by another engineer with only the repo and the dataset.

What we are NOT asking for

- A deep learning model. scikit-learn is sufficient. Pick the simplest model that gets the job done and defend the choice.
- A web UI. CSV in, CSV out.
- Real-time serving. Monthly batch is the operational pattern.
- A causal model. We want a predictive model that ranks well. Causal interpretation is out of scope for V1.

Document 2 Data Dictionary

Below is every column in train.csv and test.csv. Read the Governance status column carefully as it determines what you may and may not use.

Identifier and target

Feature	Description	Governance status
<code>member_id</code>	Unique identifier for each Meridian member. Format: M0123456.	ID — not a feature
<code>churned</code>	Binary outcome. 1 = member did not renew at their renewal decision date. 0 = member renewed. This is your training label.	Target variable

Demographic features

Feature	Description	Governance status
<code>age</code>	Member age in years at time of renewal decision. Range: 19–92.	Allowed as feature
<code>gender</code>	M / F / X. Self-reported.	BLOCKED do not use as feature. Report subgroup performance only.
<code>race</code>	Self-reported race category.	BLOCKED do not use as feature. Report subgroup performance only.
<code>state</code>	Two-letter state code: OH, IN, KY, WV.	Allowed as feature
<code>is_rural</code>	Boolean. True if member's primary residence county is classified as rural by HRSA.	Allowed as feature but check subgroup performance

Plan and enrollment features

Feature	Description	Governance status
<code>plan_type</code>	One of: <code>commercial_group</code> , <code>individual_marketplace</code> , <code>medicare_advantage</code> .	Allowed as feature
<code>dual_eligible</code>	Boolean. True if member qualifies for both Medicare and Medicaid.	BLOCKED do not use as feature. Considered a sensitive socioeconomic indicator per Module 4 governance.

Feature	Description	Governance status
mental_health_diagnosis	Boolean. True if member has a documented serious mental illness diagnosis in their claims history.	BLOCKED do not use as feature. Considered a sensitive health indicator per Module 4 governance.
tenure_years	Number of years the member has been continuously enrolled with Meridian.	Allowed as feature
chronic_conditions	Boolean. True if the member has at least one documented chronic condition.	Allowed as feature

Engagement and claims features

Feature	Description	Governance status
claims_last_12mo	Total number of claims submitted in the 12 months preceding the renewal decision.	Allowed as feature
denied_claims_last_12mo	Subset of claims_last_12mo that were denied.	Allowed as feature
premium_increase_pct	Year-over-year percent change in member's monthly premium. Can be negative if their premium decreased.	Allowed as feature
service_calls_last_12mo	Number of times the member called Meridian customer service in the 12 months preceding renewal.	Allowed as feature
last_satisfaction_score	Member's most recent satisfaction survey response, 1–5. Missing for members who did not respond to surveys.	Allowed as feature but mind the missingness

Timing features

Feature	Description	Governance status
renewal_decision_date	ISO date when the member's renewal decision was finalized. Useful for time-based splits.	Allowed for time-based splitting; not a feature directly
days_to_renewal_decision	Days between the latest member event in our records and the renewal_decision_date. Can be negative.	LEAKAGE this is computed post-outcome and is not available at prediction time. Do not use.

Feature	Description	Governance status
<code>cancellation_for_m_started</code>	Boolean. True if the member began a cancellation form before their renewal decision date.	LEAKAGE this is the outcome leaking. By the time this is True, the retention team has already lost them. Do not use.

On the BLOCKED and LEAKAGE features

These columns are in the dataset on purpose. In real Ai/ML jobs you will receive datasets with both governance landmines and leakage traps in them, and you will be expected to spot them without being told. The rubric specifically rewards students who identify, document, and exclude them. **We specifically penalize students who use them even unintentionally.** A high-AUC model trained with leakage features will be penalized harder than a lower-AUC model trained honestly.

Data drift you should know about

The train and test splits are NOT a random split of the same underlying distribution. They are time-based:

- train.csv covers renewal decisions from January 2024 through December 2024 (40,000 rows).
- test.csv covers renewal decisions from January 2025 through June 2025 (15,000 rows).

Real-world distributions shift over time. Some of that shift is in this dataset, and some of your evaluation work should account for it. If you cross-validate as if test.csv is i.i.d. with train.csv, you will produce numbers that look better than the real future will be.

Class imbalance

Roughly 24–30% of rows are positive (churned = 1) depending on the split. This is high enough that you do not need exotic techniques (SMOTE, etc.), but it is low enough that accuracy is a misleading metric..

Missingness

Several columns have missing values, most notably last_satisfaction_score (~30–38% missing). Missingness is not random — it correlates with member engagement, which in turn correlates with churn. Your imputation strategy matters; document it in the README.