COHORT REVENUE & RETENTION ANALYSIS: A BAYESIAN APPROACH

JUAN CAMILO ORDUZ

Abstract. We present a bayesian approach to model cohort-level retention rates and revenue over time. We use bayesian additive regression trees (BART) to model the retention component which we couple with a linear model to model the revenue component. This method is flexible enough to allow adding additional covariates to both model components. This bayesian model allows us to quantify the uncertainty in the estimation, understand the effect of the covariates on the retention through partial dependence plots (PDP), individual conditional expectation (ICE) plots and last but not least forecast the future revenue and retention rates.

Contents

1. Introduction

Retention and customer lifetime value estimation are one of the most important aspects to understand customer behavior. There are many ways to model retention and revenue by modeling individual-level purchase behavior for both the contractual and the non-contractual setting, see for example the work by Fader and Herdie [2] and [1] respectively[1]. In real cases, one is interested in modeling retention and revenue at cohort-level. There are (at least) three options to use the techniques mentioned above to model cohort-level retention:

(1) Pool the cohorts together and model the retention and revenue as a whole.
(2) Un-pool the cohorts and model each cohort separately.
(3) Try to model the cohorts jointly.

Date: May 23, 2023.

[1]Our definition of retention is what they call survival curve. See precise definitions below.

See [4] for more details on these approaches. One of the limitations of those approaches is that they are not flexible enough to model seasonality and add external regressors[2]. One can argue that seasonality is not that important when trying to estimate the customer-lifetime-value. Nevertheless, in practice, there are business models on which the customer base is very seasonal.

In this work, we present a bayesian approach to model cohort-level retention rates from a top-down perspective. We do not model the individual-level purchase behavior[3] but rather the retention and revenue at cohort matrices. This approach allows for modeling non-linear relationships between cohorts, adding seasonality and external regressors. Concretely, we use bayesian additive regression trees (BART, see [10]) to model the retention component and we couple it with a linear model to model the revenue. The following are the main ingredients behind the model:

Features. The following are the main features used to model retention and revenue:

- **Cohort age**: Age of the cohort in months.
- **Age**: Age of the cohort with respect to the observation time. This feature is a numerical encoder for the cohort.
- **Month**: Month of the observation time (period).

In Figure 1 we show an example of a retention matrix. Note that we are removing the diagonal as it is not informative since it has just ones. As an example, let us assume our observation month is *2022-11* and consider the cohort *2022-09*. In this case, the age of this cohort is 2 months as it is always relative to the observation period. This cohort was two observation periods *2022-10* and *2022-11* with cohort age 1 and 2 respectively.

Note that all of these features are accessible for out-of-sample predictions. As we will see below, in practice we can add more covariates to the model. The only requirement for out-of-sample predictions is that the covariates are available for the new observation period.

Model Specification.

- We model the number of active users in the cohort as a binomial random variable $\text{Binomial}(N_{\text{total}}, p)$, where the parameter p represents the retention. We model the latent variable p using a BART model with features cohort age, age and month.
- We model the revenue as a gamma random variable $\text{Gamma}(N_{\text{active}}, \lambda)$. We model the latent variable λ through a linear model with features cohort age, age and, a multiplicative interaction (to be precise using a log as a link function). Note that we do not add a seasonality component as we often see most of the seasonality coming from the retention itself. This of course can be added as a feature (plus any other covariates!) to the model.

[2]Actually, one can add regressors in some cases as described in [3] in the non-contractual case.

[3]This is a clear limitation if one is interested at customer-level parameters and predictions.

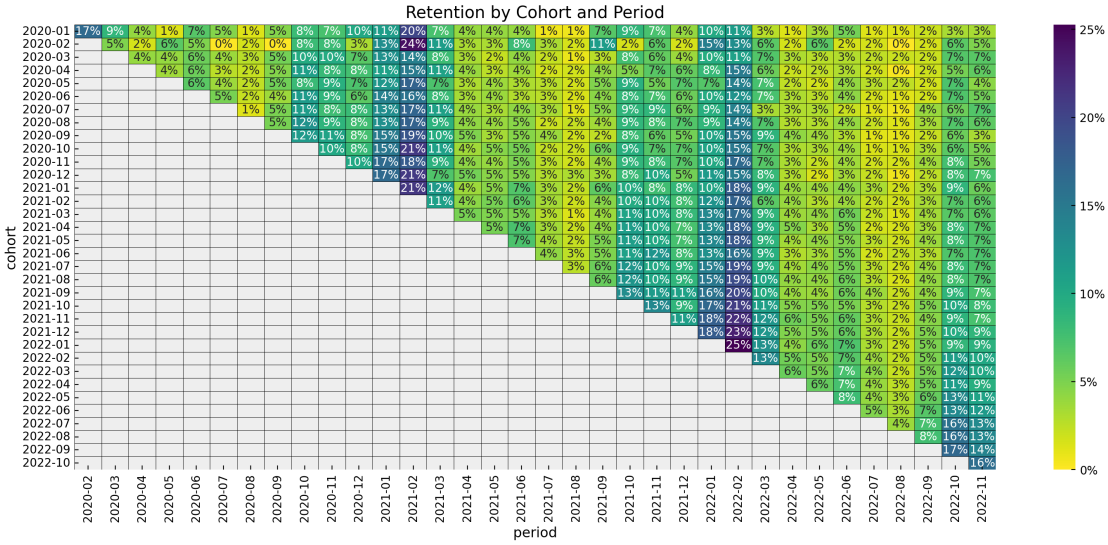


FIGURE 1. Retention matrix example.

- Here is a summary of how the retention and revenue components are coupled together:

$$
\begin{aligned}
\text{Revenue} &\sim \text{Gamma}(N_{\text{active}}, \lambda) \\
\log(\lambda) &= (\text{intercept} \\
&\quad + \beta_{\text{cohort age}}\text{cohort age} \\
&\quad + \beta_{\text{age}}\text{age} \\
&\quad + \beta_{\text{cohort age}\times\text{age}}\text{cohort age} \times \text{age}) \\
N_{\text{active}} &\sim \text{Binomial}(N_{\text{total}}, p) \\
\text{logit}(p) &= \text{BART}(\text{cohort age}, \text{age}, \text{month})
\end{aligned}
$$

We are interested in estimating the BART parameters and the beta coefficients (plus the intercept) of the linear component simultaneously. Figure 2 summarizes the model structure.

Remark 1. This work is the result of a sequence of blog posts where all the details on the code and implementation are presented, see [5], [6] and [7].

2. Synthetic Data

We illustrate the definitions, concepts and model using a synthetic data set (you can get it as *csv* from [9]). The code to (deterministically!) generate the data set is publicly available in [8].

To start the analysis, let's do some exploratory data analysis. Figure 1 shows the retention matrix per cohort and period. There are two things to note at first glance:

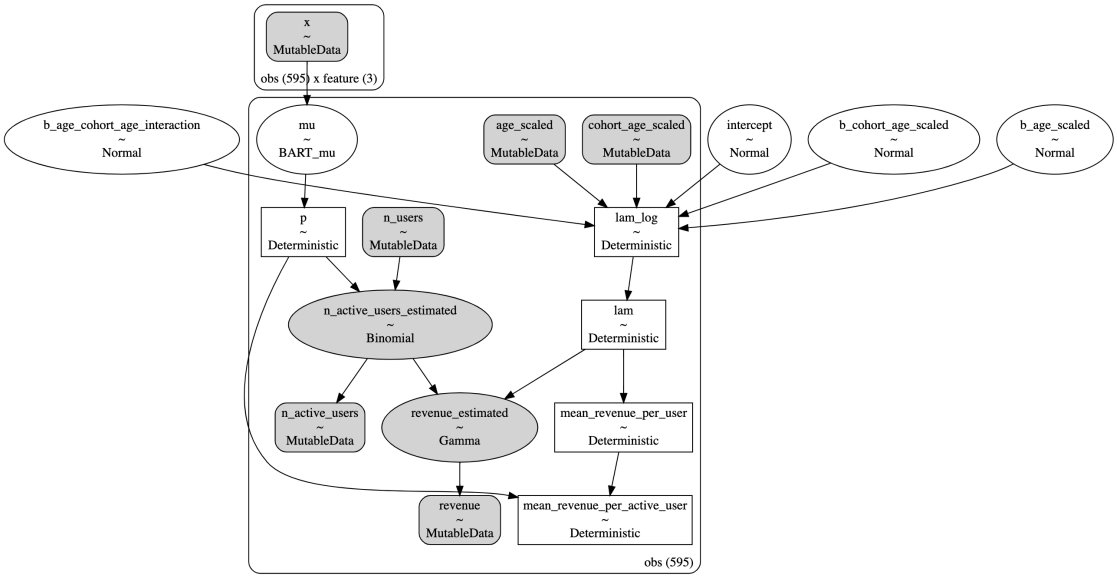


FIGURE 2. Cohort-revenue-retention model structure.

(1) The retention has a clear seasonal pattern concerning the period. Note it is higher in the last months of the year and lower in the middle of the year. To see the seasonality pattern more clearly see Figure 3.
(2) The retention seems to be increasing as the age decreases. You could see this by comparing the retention values when the period month is November and the cohort age is one.

It is important to keep in mind that the retention metric is a quotient, and therefore the cohort size matters. For example, a retention rate of 0.4 could come from $4/10$ or $4 \times 10^5/10^6$. One could argue that the former case carries more uncertainty regarding the estimation. This motivates looking into the number of active users' values. These are shown in Figure 4. We can see that the number of active users increases considerably for newer cohorts. We would like this information to be encoded in the model.

Now, let us look at the revenue. Figure 5 shows the revenue per cohort. We see from the plots how the revenue correlates with the number of active users. This hints that the revenue per user does not change dramatically over time. To see this, we can compute the revenue per user as a function of the age and the period (see Figure 6). We also can compute the revenue per *active* user (see Figure 7). The main difference between the two is that for the former we divide by the cohort size, while for the latter we divide by the number of active users in the given period. Here are some observations:

- The revenue per user shows a clear seasonality pattern. This is expected as the retention has a seasonality pattern.

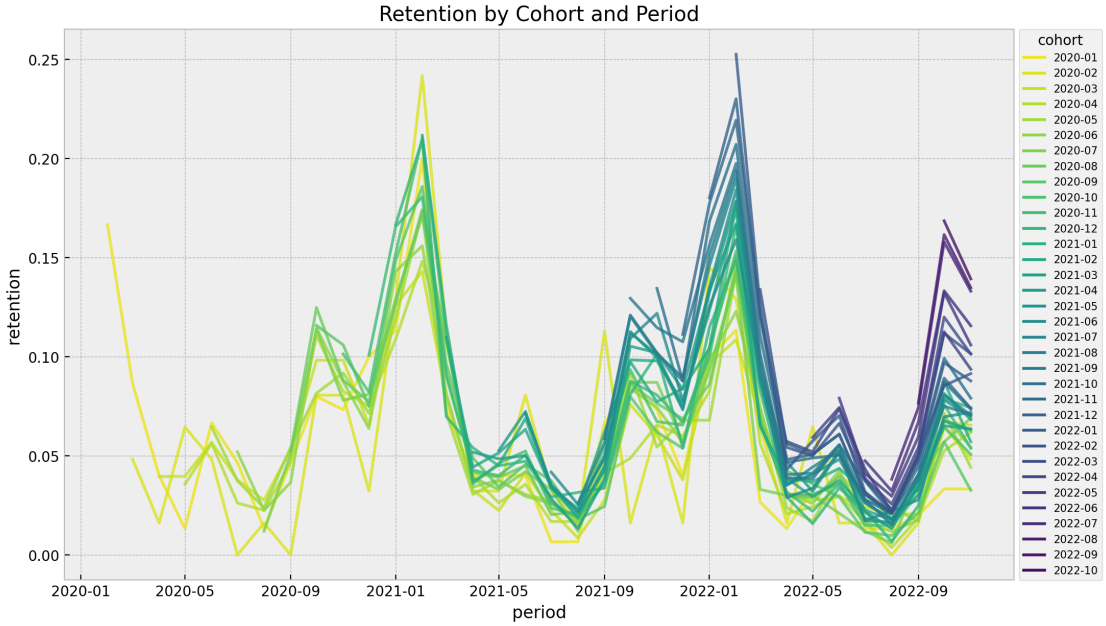


FIGURE 3. Retention as a function of the period. This plot clearly shows the yearly seasonality pattern of the retention values.

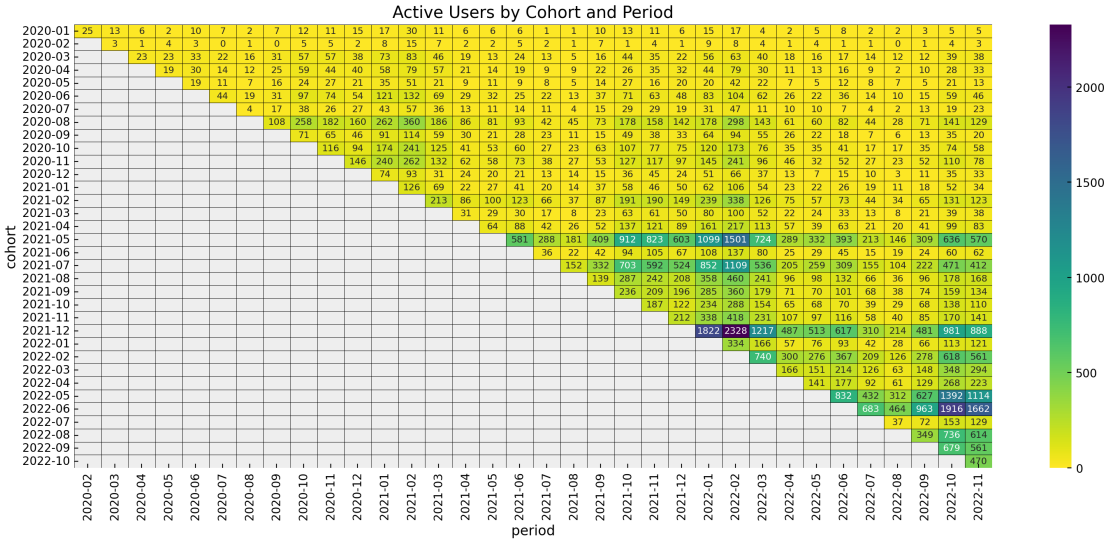


FIGURE 4. Active users' values.

- The revenue per active user does now the seasonality pattern as it s already encoded in the denominator. In addition, we see that the revenue per active user seems to be decreasing as the cohort age increases.

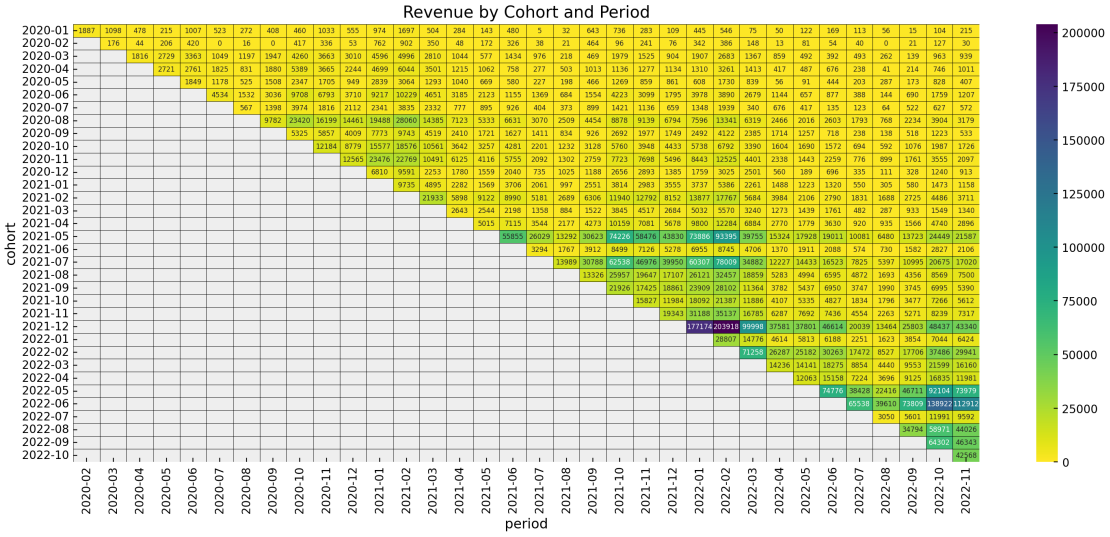


FIGURE 5. Revenue per cohort.

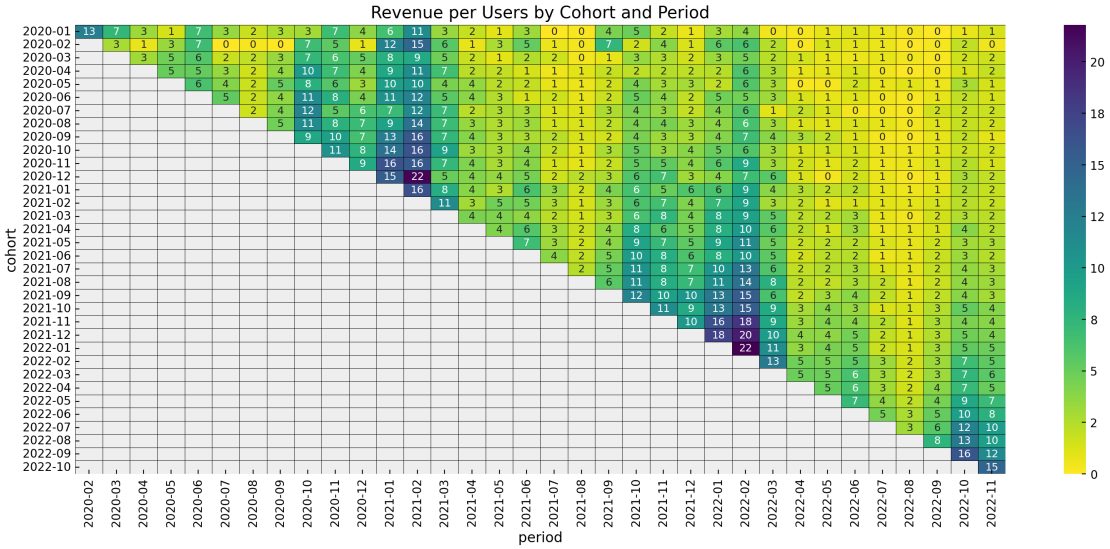


FIGURE 6. Revenue per cohort.

After this exploratory analysis, we can start the modeling phase.

3. Model Specification and Diagnostics

Now we expand on the model structure described in the introduction. The main idea is to model the number of active users as a binomial random variable $\text{Binomial}(N_{\text{total}}, p)$, where the parameter p represents the retention. We use bayesian additive regression trees

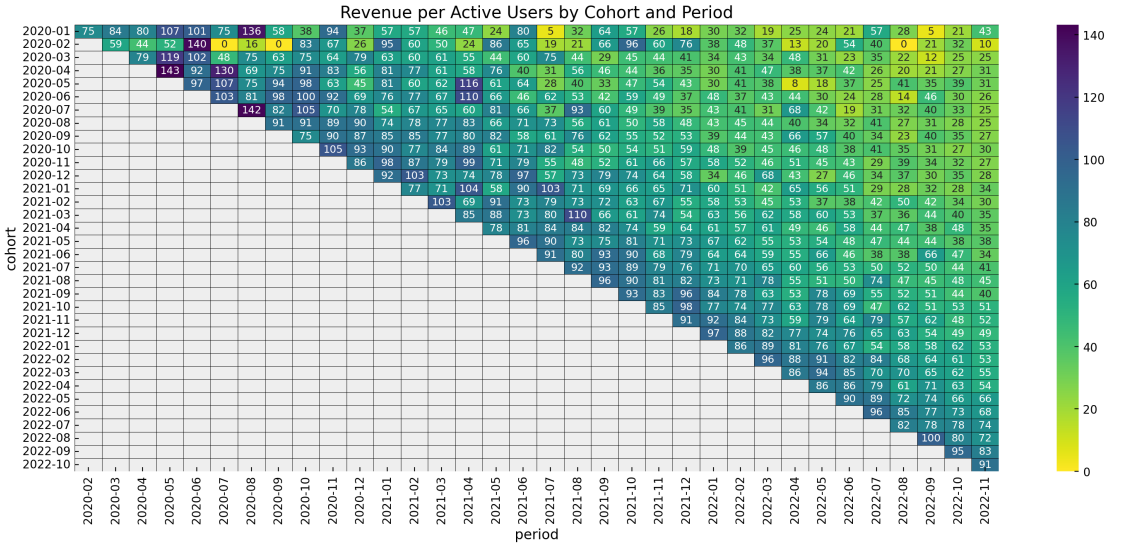


FIGURE 7. Revenue per cohort.

(BART) to model the latent variable p using as features cohort age, age and (period) month.

$$N_{\text{active}} \sim \text{Binomial}(N_{\text{total}}, p)$$
$$\text{logit}(p) = \text{BART}(\text{cohort age}, \text{age}, \text{month})$$

The only parameter we need to specify is ht number of trees in the BART model. We generally start with a small number of trees and increase them by checking the posterior predictive distribution (see below).

Remark 2. Using a BART model is very easy to add new covariates to the model. For example, we have experimented with adding certain customer segmentation features (e.g. media channel acquisition from an attribution model) in some real business applications. This turns to provide key insights into the whole media channel return-on-investment (ROI) to not only consider the cost per acquisition but also the estimated revenue per user over a given period (that is, blend with customer lifetime value estimations through this model).

Remark 3. One could of course start with a simpler model, say a linear model as described in [7]. Nevertheless, in real datasets, this approach does not fit the data well.

On the other hand, we model the revenue component using a gamma random variable $\text{Gamma}(N_{\text{active}}, \lambda)$ (this was inspired in the work [11]). Note that the mean of this gamma distribution is $N_{\text{active}}/\lambda$, thus we can interpret $1/\lambda$ as the *average revenue per active user*. We model $\log(\lambda)$ through a linear model using the cohort age and age (and potentially higher-order interactions) as features.

$$\begin{aligned}
\text{Revenue} &\sim \text{Gamma}(N_{\text{active}}, \lambda) \\
\log(\lambda) &= (\text{intercept} \\
&\quad + \beta_{\text{cohort age}}\text{cohort age} \\
&\quad + \beta_{\text{age}}\text{age} \\
&\quad + \beta_{\text{cohort age}\times\text{age}}\text{cohort age} \times \text{age})
\end{aligned}$$

One of the key observations we have seen in many real applications (and also in this synthetic data set is that we do not need to add a seasonality component into the retention model as it is already captured by the retention itself.

Remark 4. Note that the *age* feature characterizes the cohort itself. One could try to replace the age numerical encoding with a one-hot encoding of the cohort. This would allow the addition of a hierarchical structure to the model to pool information across cohorts. However, under the assumption that close cohorts are more similar than distant cohorts, the numerical encoding is more appropriate and results in a simpler model.

As part of the pre-processing step, we standardize the features for the linear models. The main benefit of this is to be able to specify priors for the coefficients of the regression which could be interpreted as the effect of a one standard deviation change. That is, we could regularize by taking standard normal priors for the coefficients (see [6]).

In summary, the cohort-revenue-retention model is specified as follows:

$$\begin{aligned}
\text{Revenue} &\sim \text{Gamma}(N_{\text{active}}, \lambda) \\
\log(\lambda) &= (\text{intercept} \\
&\quad + \beta_{\text{cohort age}}\text{cohort age} \\
&\quad + \beta_{\text{age}}\text{age} \\
&\quad + \beta_{\text{cohort age}\times\text{age}}\text{cohort age} \times \text{age}) \\
N_{\text{active}} &\sim \text{Binomial}(N_{\text{total}}, p) \\
\text{logit}(p) &= \text{BART}(\text{cohort age}, \text{age}, \text{month}) \\
\text{intercept} &\sim \text{Normal}(0, 1) \\
\beta_{\text{cohort age}} &\sim \text{Normal}(0, 1) \\
\beta_{\text{age}} &\sim \text{Normal}(0, 1) \\
\beta_{\text{cohort age}\times\text{age}} &\sim \text{Normal}(0, 1)
\end{aligned}$$

Remark 5. Remember that in the linear model, we standardize the features. We do not add it to the specification above to simplify the notation.

Once we have the model specification, we can implement it in PyMC (see Appendix A and [7]). Figure 8 shows the posterior predictive distribution of both of the components. The results look quite good. In addition, we can check the trace of the linear terms (see Figure 9). The model did not present any divergences or warnings.

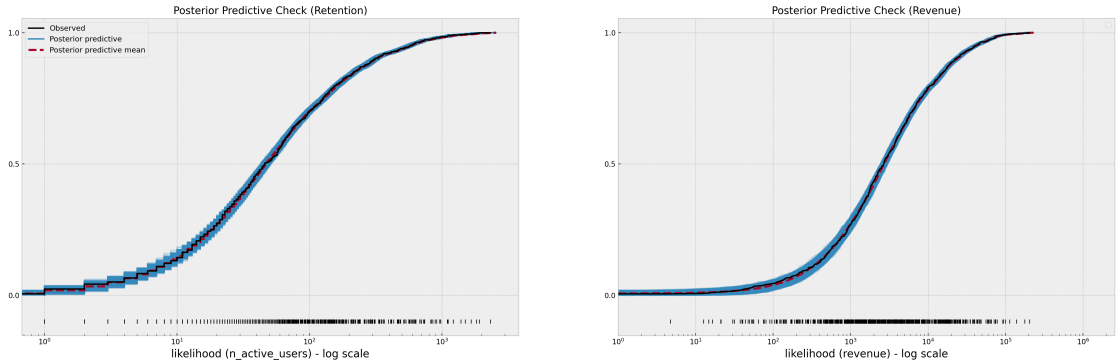


FIGURE 8. Posterior predictive distribution of the retention (left) and revenue (right) per cohort.

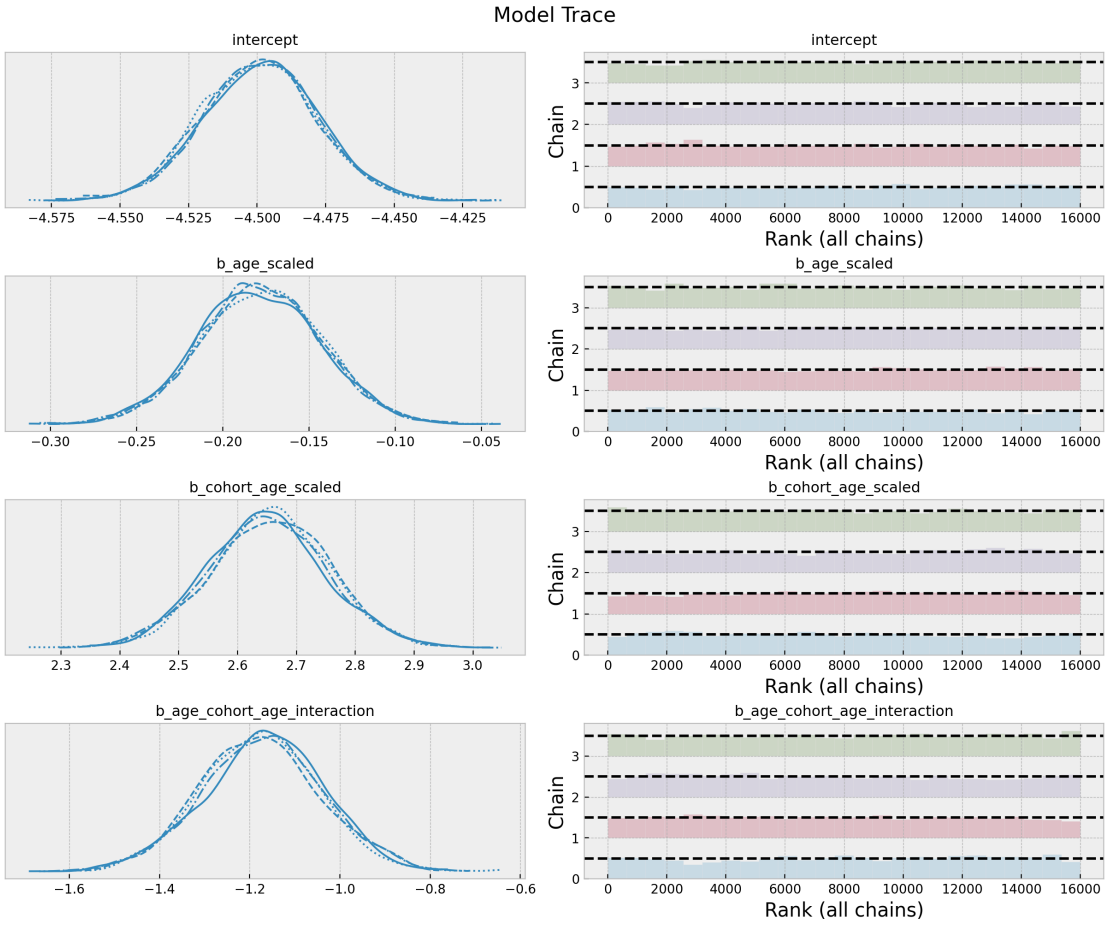


FIGURE 9. Linear model trace plots.

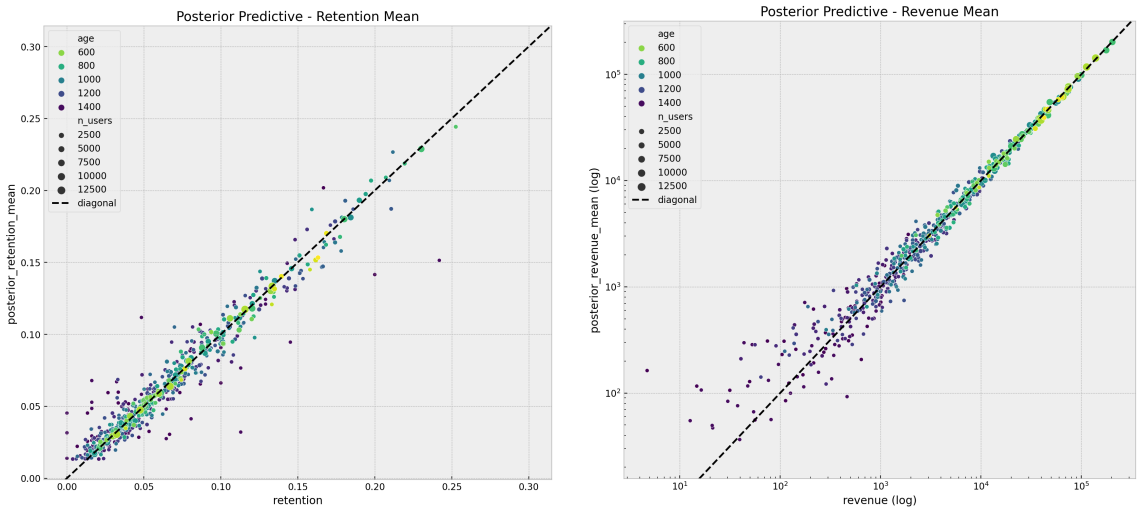


FIGURE 10. Retention (left) and revenue (right) in-sample posterior predictive mean against the actuals.

4. PREDICTIONS

In this section, we present the in-sample and out-of-sample predictions of the model.

4.1. **In-Sample Predictions.** The first thing we can do is to check the in-sample mean posterior predictions of the model. We can compare the in-sample predictions with the actuals in Figure 10. The results look quite good.

We can also visualize the in-sample posterior predictive distribution of the retention for a specific subset of cohorts as in Figure 11. Observe how the credible intervals are quite narrow for the cohorts with more data (more recent ones). Overall, the in-sample predictions capture the behavior of the data quite well. We can also visualize the in-sample posterior predictive distribution of the revenue values, see Figure 12. Again, the model posterior predictive captures most of the variability of the data.

4.2. **Out-of-Sample Predictions.** We can generate analogous plots for the out-of-sample predictions for retention and revenue, see Figure 13 and Figure 14 respectively. The credible intervals match the holdout set consistently well. Note in particular how we can generate very good predictions for the cohort *2022-07-01* for we just have 4 data points in the training set. We are successfully pooling information (trend and seasonality) from previous cohorts.

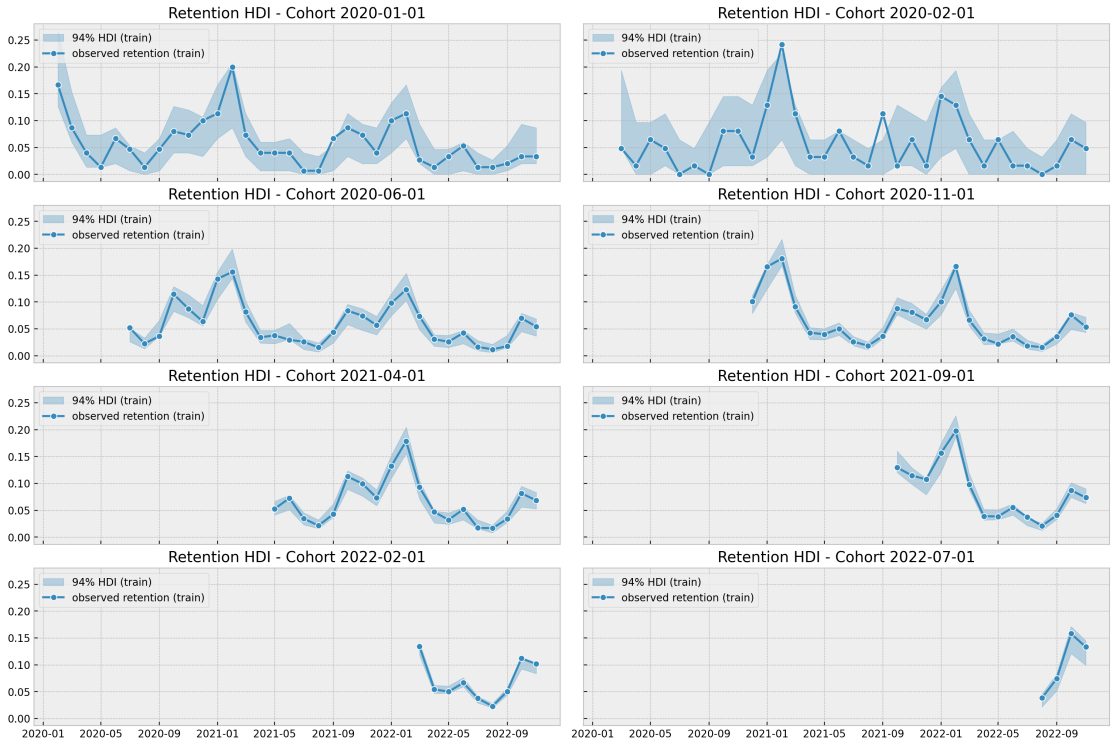


FIGURE 11. Retention in-sample posterior predictive distribution for a subset of cohorts.

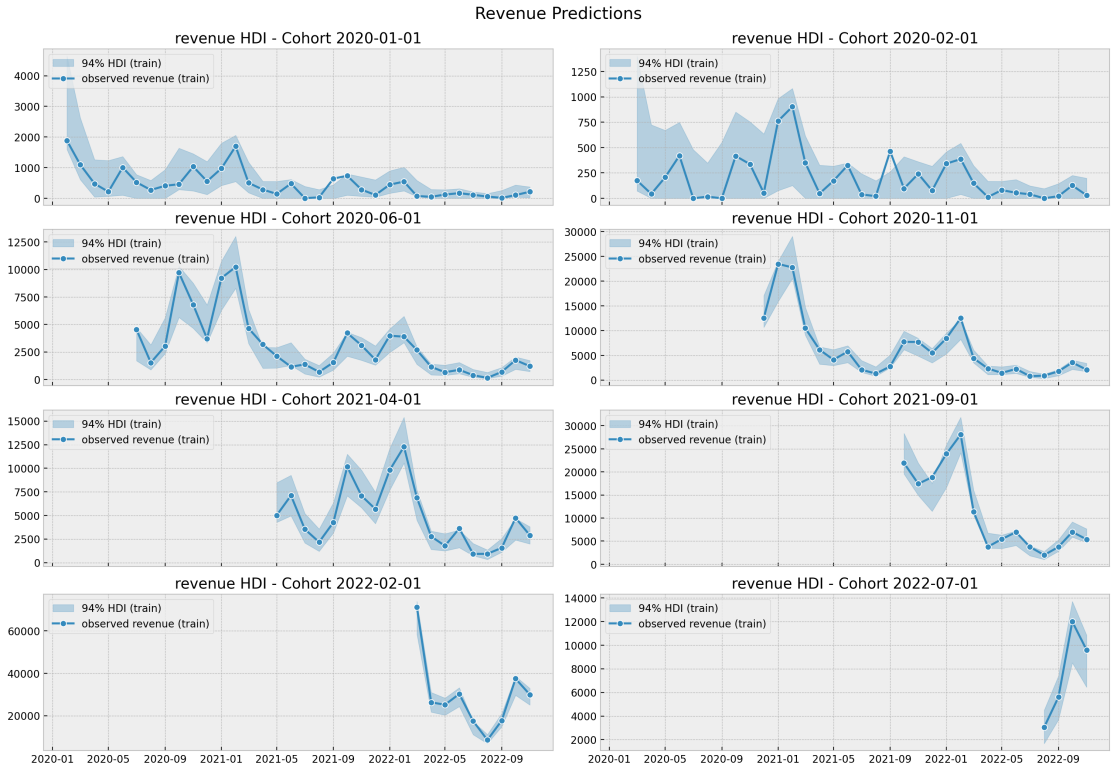
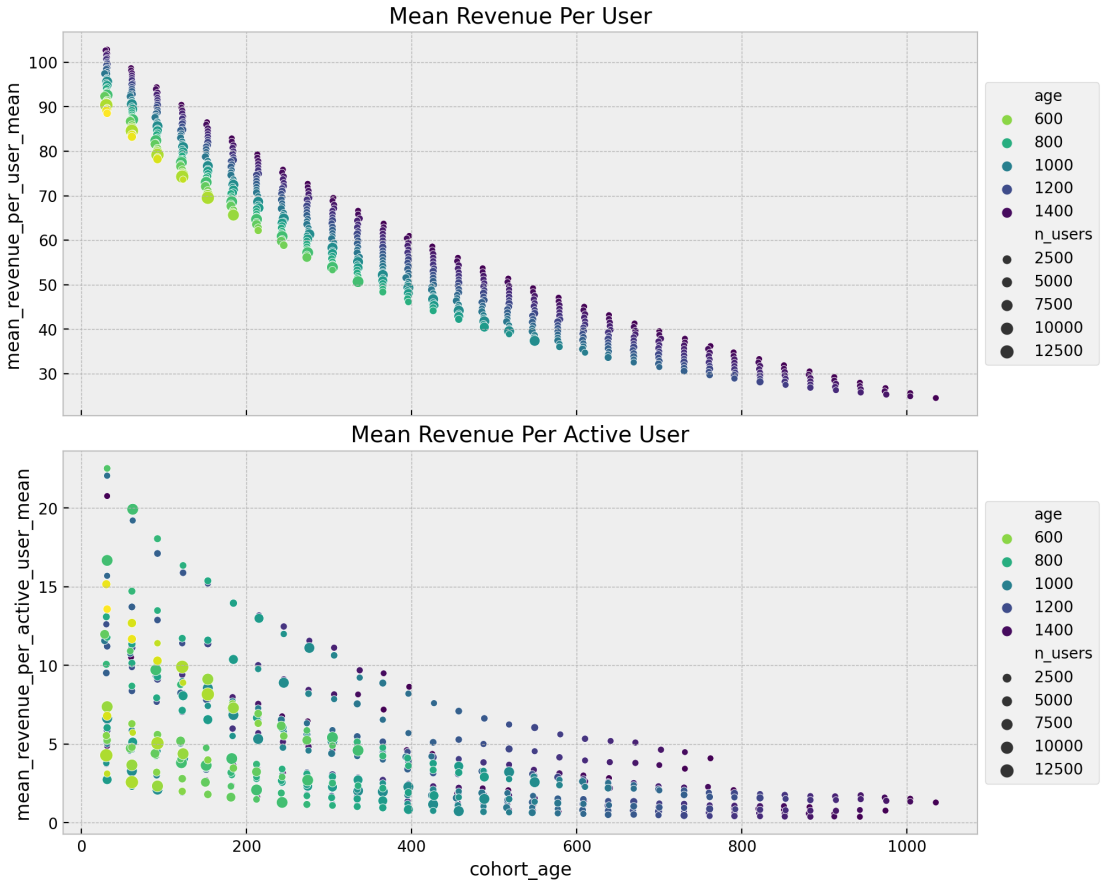


FIGURE 12. Revenue in-sample posterior predictive distribution for a subset of cohorts.


Mean Revenue Per User
mean_revenue_per_user_mean
100
90
80
70
60
50
40
30
age
600
800
1000
1200
1400
n_users
2500
5000
7500
10000
12500
Mean Revenue Per Active User
mean_revenue_per_active_user_mean
20
15
10
5
0
0
200
400
600
800
1000
cohort_age

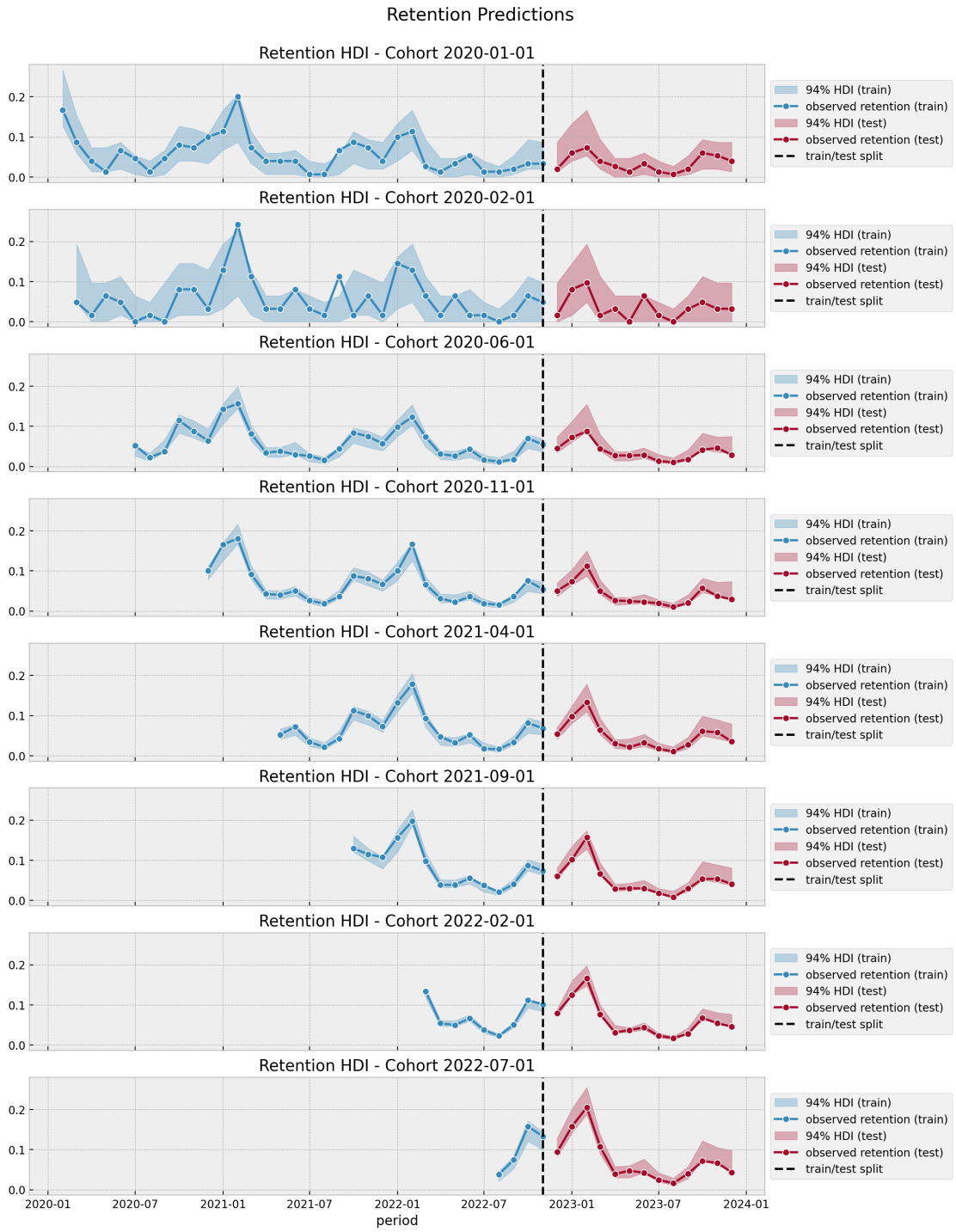


FIGURE 13. Retention out-of-sample posterior predictive distribution for a subset of cohorts.

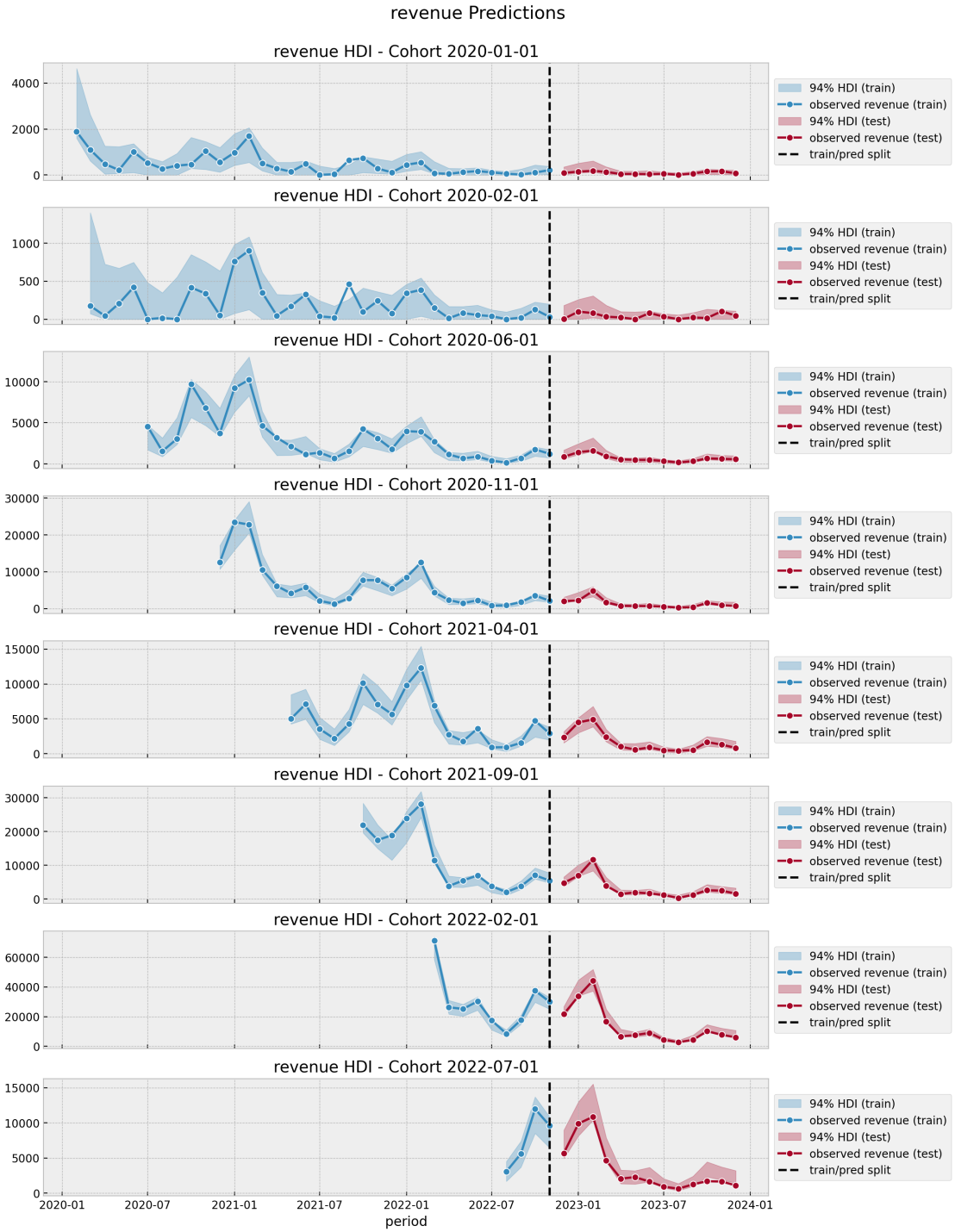


FIGURE 14. Revenue out-of-sample posterior predictive distribution for a subset of cohorts.

Appendix A. Python Code

In this appendix, we present the Python code core used to implement the model in PyMC. The detailed implementation can be found in [9].

Listing 1. PyMC model implementation.

```
import pymc_bart as pmb
import pymc as pm


with pm.Model(coords={"feature": features}) as model:

    # --- Data ---
    model.add_coord(name="obs", values=train_obs_idx, mutable=True)
    age_scaled = pm.MutableData(
        name="age_scaled", value=train_age_scaled, dims="obs"
    )
    cohort_age_scaled = pm.MutableData(
        name="cohort_age_scaled", value=train_cohort_age_scaled, dims="obs"
    )
    x = pm.MutableData(name="x", value=x_train, dims=("obs", "feature"))
    n_users = pm.MutableData(name="n_users", value=train_n_users, dims="obs")
    n_active_users = pm.MutableData(
        name="n_active_users", value=train_n_active_users, dims="obs"
    )
    revenue = pm.MutableData(name="revenue", value=train_revenue, dims="obs")

    # --- Priors ---
    intercept = pm.Normal(name="intercept", mu=0, sigma=1)
    b_age_scaled = pm.Normal(name="b_age_scaled", mu=0, sigma=1)
    b_cohort_age_scaled = pm.Normal(name="b_cohort_age_scaled", mu=0, sigma=1)
    b_age_cohort_age_interaction = pm.Normal(
        name="b_age_cohort_age_interaction", mu=0, sigma=1
    )

    # --- Parametrization ---
    # The BART component models the image of the retention rate under the
    # logit transform so that the range is not constrained to [0, 1].
    mu = pmb.BART(name="mu", X=x, Y=train_retention_logit, m=50, dims="obs")
    # We use the inverse logit transform to get the retention rate
    # back into [0, 1].
    p = pm.Deterministic(name="p", var=pm.math.invlogit(mu), dims="obs")
    # We add a small epsilon to avoid numerical issues.
    p = pt.switch(pt.eq(p, 0), eps, p)
    p = pt.switch(pt.eq(p, 1), 1 - eps, p)

    # For the revenue component we use a Gamma distribution where we
    # combine the number of estimated active users with the average
    # revenue per user.
```

```
    lam_log = pm.Deterministic(
        name="lam_log",
        var=intercept
        + b_age_scaled * age_scaled
        + b_cohort_age_scaled * cohort_age_scaled
        + b_age_cohort_age_interaction * age_scaled * cohort_age_scaled,
        dims="obs",
    )

    lam = pm.Deterministic(name="lam", var=pm.math.exp(lam_log), dims="obs")

    # --- Likelihood ---
    n_active_users_estimated = pm.Binomial(
        name="n_active_users_estimated",
        n=n_users,
        p=p,
        observed=n_active_users,
        dims="obs",
    )

    x = pm.Gamma(
        name="revenue_estimated",
        alpha=n_active_users_estimated + eps,
        beta=lam,
        observed=revenue,
        dims="obs",
    )

    # --- Derived Quantities ---
    mean_revenue_per_user = pm.Deterministic(
        name="mean_revenue_per_user", var=(1 / lam), dims="obs"
    )
    pm.Deterministic(
        name="mean_revenue_per_active_user",
        var=p * mean_revenue_per_user,
        dims="obs"
    )
```

References

[1] Fader, P., Hardie, B., and Lee, K. "Counting Your Customers" the Easy Way: An Alternative to the Pareto/NBD Model. *Marketing Science 24* (05 2005), 275–284.

[2] Fader, P. S., and Hardie, B. G. How to project customer retention. *Journal of Interactive Marketing 21*, 1 (2007), 76–90.

[3] Fader, P. S., and Hardie, B. G. Incorporating Time-Invariant Covariates into the Pareto/NBD and BG/NBD Models. `http://brucehardie.com/notes/019/`, 2007.

[4] Fader, P. S., and Hardie, B. G. Fitting the sBG Model to Multi-Cohort Data. `http://brucehardie.com/notes/017/`, 2017.

[5] Orduz, J. A Simple Cohort Retention Analysis in PyMC. https://juanitorduz.github.io/retention/, 12 2022.
[6] Orduz, J. Cohort Retention Analysis with BART. https://juanitorduz.github.io/retention_bart/, 01 2023.
[7] Orduz, J. Cohort Revenue & Retention Analysis: A Bayesian Approach. https://juanitorduz.github.io/revenue_retention/, 01 2023.
[8] Orduz, J. Cohort Revenue & Retention Analysis: A Bayesian Approach - Code to generate data. https://github.com/juanitorduz/website_projects/blob/master/Python/retention_data.py, 01 2023.
[9] Orduz, J. Cohort Revenue & Retention Analysis: A Bayesian Approach - Data (csv). https://github.com/juanitorduz/website_projects/blob/master/data/retention_data.csv, 01 2023.
[10] Quiroga, M., Garay, P. G., Alonso, J. M., Loyola, J. M., and Martin, O. A. Bayesian additive regression trees for probabilistic programming, 2022.
[11] Stucchio, C. Bayesian a/b testing at vwo. https://vwo.com/downloads/VWO_SmartStats_technical_whitepaper.pdf, 2015.

Email address: juanitorduz@gmail.com
URL: https://juanitorduz.github.io/

Berlin, Germany