Credit Risk Modeling: From Model to Business Value

Abstract

This project builds a credit risk model using Home Credit data, aiming to achieve both accuracy and stability over time. Using systematic preprocessing, feature engineering, and a LightGBM model optimized for PR-AUC under class imbalance, the model shows strong predictive power (ROC-AUC above 0.85) and stable results across different periods. From a business view, the model can be used with confidence for credit approval and monitoring, keeping portfolio risk under control across different bad-rate caps. Most top features are stable, but two unstable drivers were found, which should be monitored to avoid future model drift.

1. Data

1.1 Source

The dataset used in this project originates from the Kaggle competition *Home Credit – Credit Risk Model Stability*. It contains historical loan application records from Home Credit, with the target variable indicating whether the client experienced payment difficulties (1 = default, `0` = non-default). All experiments are based solely on the train set provided by the competition in Parquet format.

1.2 Structure

- Observation unit: each credit case. (case_id)
- Target variable: `target` (binary default indicator)
- Temporal index: `WEEK_NUM`, used for time-based model evaluation and stability checks
- Feature origins: application-level static features, historical behavioral features, and derived transformations
- Feature suffixes denote transformation types. **P** (DPD (Days past due)), **A** (amounts), **D** (dates), **M** (masked categorical variables) and **T** and **L**(Unspecified Transform).

1.3 Data Characteristics

- **Imbalance**: Defaults form only a small proportion of applications
- **Temporal drift**: Feature distributions and default rates change across time
- **Missing values**: Many features have high missingness, requiring domain-specific imputation
- **Heterogeneity**: Data come from multiple historical tables, mixing numeric, categorical, and date-derived features

2. Data Preprocessing – Stage 1 (Before Merge)

This stage is performed before merge and splitting, when the dataset still contains multiple historical records (rows) per case_id from depth=1 and depth=2 tables. All processing here is row-level within historical records of case_id.

2.1 Missing Value Imputation

Applied suffix-based imputation rules:

- Numeric amount or DPD variables → fill with 0.
- Masked categorical variables → fill with `"missing"` (string) or `-1` (numeric category codes).
- Date and certain mixed-type variables → left unfilled to preserve raw information.

2.2 Additional Feature Creation

Generated domain-driven combination features at the historical record level, such as delinquency ratios, debt-to-income measures, credit utilization, repayment progress, and overdue structure indicators.

Constructed row-level domain-driven combination features within each historical record, including:

- **Delinquency behavior**:
  - `Overdue Days to Balance Ratio, Recent Delinquency Worsening Indicator, Severe Delinquency (≥60 Days) in Past Year/Two-Year, Chronic Severe Delinquency (≥60 Days) over 24 Months, Delinquency Improvement Indicator, Early-Stage Delinquency Indicator`
- **Debt & income capacity**:
  - `Debt-to-Income Ratio (DTI), Disposable Income Ratio, Total Debt Burden Ratio, High Leverage Indicator (DTI > 60%)`
- **Credit usage**:
  - `Credit Utilization Ratio, Credit Card Balance-to-Limit Ratio`
- **Repayment & overpayment**:
  - `Repayment Progress Ratio, Overpayment Indicator`
- **Overdue structure**:
  - `Active Account Overdue Ratio, Closed Account Overdue Ratio, Active Account Overdue Indicator, Closed Account Overdue Indicator, Payment-Level Overdue Indicator`
- **Other financial behavior**:
  - `Payment Ability Ratio, Residual Balance-to-Price Ratio (Active Accounts), Residual Balance Indicator (Closed Accounts), Revolving Account Ownership Indicator, Delinquency Tolerance Ratio, Recent-to-Historical Delinquency Relief Ratio, Short-Term vs. Long-Term Delinquency Delta, Severe Current Delinquency Indicator(≥60 Days), Negative Delinquency Change Indicator`

These features capture payment history, credit utilization, repayment patterns, and overdue structures at the historical record level.

2.3 Aggregation of Historical Data

For depth=1 and depth=2 tables, aggregated historical records per `case_id` according to suffix/type-specific rules:

- **Numeric amount / DPD features (P, A):** max, last, mean.
- **Date features (D):** converted to days from `date_decision`, then max, last, mean.
- **Categorical (masked) features (M):** max, last.
- **Other transformed features (T, L):** max, last.
- **Index count features (`num_group` columns):** max, last.
- **Flag / Ratio / Delta features（combo feature）:** mean, max, last.

This aggregation logic compresses multiple historical rows per `case_id` into one case-level row while preserving both recent and long-term behavioral signals.

2.4 Merge & Conflict Resolution

Sequentially merged aggregated tables into the base table on `case_id`. For conflicting columns, values from the left table were prioritized, with nulls filled from the right table.

2.5 Date Features

Extracted decision month and weekday, and converted all `_D` date variables into days relative to `date_decision`.

3. Data Split

To keep the evaluation realistic, the dataset was split by decision week. After cleaning data types and sorting records by `WEEK_NUM` and `weekday_decision`, I built three non-overlapping subsets:

- **Training set (Weeks 0–69):** used to train the model and perform feature selection.
- **Validation set (Weeks 70–74):** used to search for the business threshold.
- **Test set (Weeks 75–91):** used to evaluate both model and business performance.

This chronological split makes sure the model is tested in the same way it would face real future cases.

4. Data Preprocessing – Stage 2 (After Split)

This stage is performed after aggregation and chronological splitting, when the dataset is already in one row per `case_id` format. All processing here is application-level with aggregated case-level features.

4.1 Missing Value Imputation

Applied a second round of suffix-based imputation to handle any remaining missing values in the aggregated case-level dataset, after applying a hard filter (removing features with missing value ratio > 97% or cardinality > 5000).

4.2 Additional Post-Split Features

Constructed post-split case-level combination features using only data available up to each dataset's time period, including:

- `High Utilization with Severe Delinquency Indicator` – flag for high credit card utilization and severe delinquency co-occurring.
- `Recent Delinquency Change` – recent actual DPD minus its historical mean.
- `Credit Card Burden Relief Ratio` – relative improvement in credit card burden.
- `Excess Debt-to-Income Ratio` – annuity relative to main occupation income.
- `Arrears Progression Indicator` – interaction of repayment progress and delinquency severity.
- `Residual Balance Indicator` – combined residual amount and residual closed account indicators.
- `Credit Limit Change` – change in credit limit from historical mean.
- No Recent Installment Payment Indicator
- `Zero Debt-to-Income Indicator`

These features are designed to capture recent shifts and interactions in credit behavior that may not be visible from pre-split row-level combos.

4.3 Winsorization of Extreme Values

To reduce the influence of outliers, numeric variables were winsorized at the 99.8th percentile, based on the training set distribution. The same upper limits were applied to the validation and test sets to keep the datasets comparable to prevent any information leakage. In the credit risk setting, winsorization reduces the effect of extreme amounts or credit exposures, while keeping the relative order of cases' risk levels.

4.4 Encoding of Categorical Variables

Categorical predictors were processed with two approaches:

- **One-hot encoding** was applied to low-cardinality features (with ≤*20 distinct categories*).
- **Target encoding with temporal ordering** was applied to high-cardinality variables (with > *20 distinct categories*). For each record, the encoding was calculated only from past data available up to that point, using a rolling time window, which keeps the time-dependent nature of credit risk, prevents forward-looking bias, and helps the model reflect changing default risks across borrower groups.

4.5 Correlation Pruning

Removed highly correlated features (higher than 0.97) based on train set only using a preference hierarchy (mean = 1, max = 2, last = 3) that retained more stable and interpretable aggregations.

4.6 Feature Selection (Gain + SHAP Union)

Since the modeling approach is based on **GBDT**, I did not use traditional credit risk methods such as WOE and IV, which are mainly designed to capture linear relationships for logistic regression. Instead, I applied **gain-based feature importance** and **SHAP** for feature selection. I performed supervised feature selection on the training set only. A gradient-boosted tree model (LightGBM) was fitted to obtain two important views: (i) gain-based feature importance and (ii) mean absolute SHAP values. I then retained the smallest feature sets that cumulatively explain 95% of total gain and 90% of total mean absolute SHAP values, respectively, and took their union. The resulting union was applied unchanged to the

validation and test sets, preventing temporal leakage while balancing model-driven relevance and attribution robustness.

5. Model Development

I applied a **time-series expanding window cross-validation**, where each validation set followed the chronological order of loan origination. The validation windows had a fixed length and were arranged sequentially near the end of the training period. For example, with an end point at week 69 and a validation span of 4 weeks in this project, the validation windows were [54–57], [58–61], [62–65], and [66–69]. In each case, the training window always covered all weeks from the start up to the week before the validation period, which ensured an expanding training set.

The model was optimized for **PR-AUC**, highlighting the importance of ranking performance under severe class imbalance. Hyperparameters were tuned automatically with **Optuna**, and class imbalance was handled by **re-weighting positive samples**.

The final model was then trained on the full training period (weeks 0–69) with the selected parameters and stored for later stability checks and business evaluation.

6. Model Evaluation

6.1 Model Performance

I evaluated the model's predictive performance using multiple metrics, including ROC-AUC, PR-AUC, and KS statistics.

- **Overall Performance**: The aggregated metrics are summarized in Table 1 (Overall Model *Information*), which shows consistently high ROC-AUC values across validation and test sets, reliable KS statistics, and stable PR-AUC despite class imbalance.
- **Temporal Stability**: Weekly performance plots (Figure 1 Weekly Model Performance – Validation) and (Figure 2 Weekly Model Performance – Test) — show that all three main metrics remained relatively stable. Importantly, no downward trend was observed, suggesting that the model remains robust when applied to future data.

split	the number of case	Positive rate	pr_auc	roc_auc	ks	brier	ece
TRAIN	1302727	0.0331	0.2593	0.9013	0.6495	0.1429	0.2682
VAL	40554	0.0236	0.1756	0.8569	0.5550	0.1165	0.2380
TEST	183378	0.02158	0.1845	0.8718	0.5855	0.1488	0.2869

Table 1. Overall Model Information

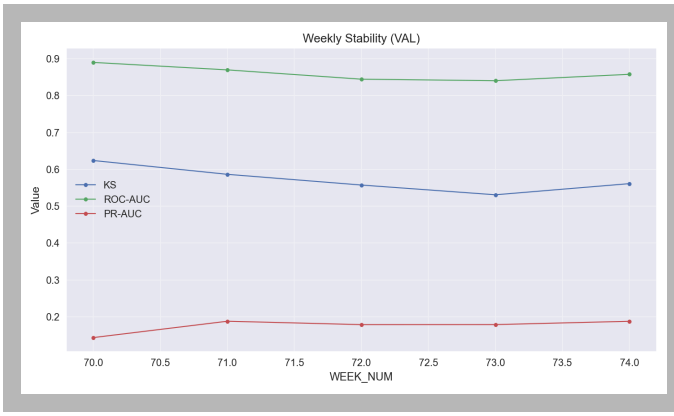


Figure 1 Weekly Model Performance – Validation

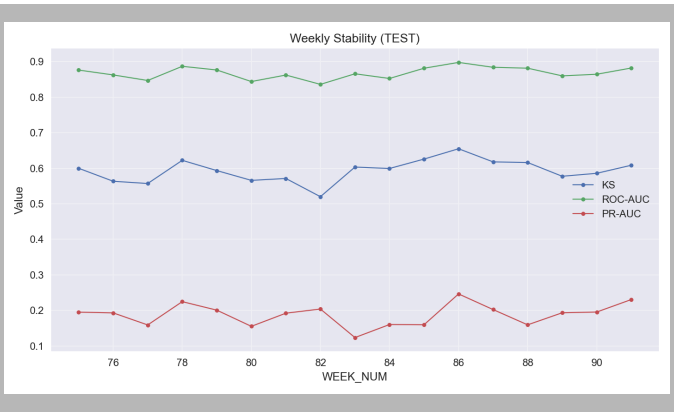


Figure 2 Weekly Model Performance – Test

From a business perspective, the results suggest that the model can be deployed with confidence for credit approval and monitoring. Its stable performance over time reduces concerns about sudden drops in accuracy, helping to maintain portfolio quality and support long-term risk management goals. If the model's KS declines in the later part of the test period, this may indicate potential overfitting and should be investigated further.

6.2 Stability Analysis

To assess the temporal robustness of the model, I computed **Population Stability Index (PSI)** by comparing score distributions between training, validation, and test datasets. Results are presented in two layers: overall stability and weekly stability.

6.2.1 Overall PSI

- **Validation (Weeks 70–74 vs TRAIN):** Average PSI = 0.0567
- **Test (Weeks 75–91 vs TRAIN):** Average PSI = *0.0373*

All values are well below 0.1, indicating stable score distributions without evidence of systematic drift.

6.2.2 Weekly PSI

- **Validation Period:** Weekly PSI ranges from 0.047 to 0.067, consistently stable across Weeks 70–74. (Figure 3 *Weekly PSI(VAL to TRAIN)*)
- **Test Period:** Weekly PSI fluctuates between 0.03 and 0.101, with a temporary spike near Week 79 but no persistent drift. (Figure 4 *Weekly PSI(TEST to TRAIN)*).

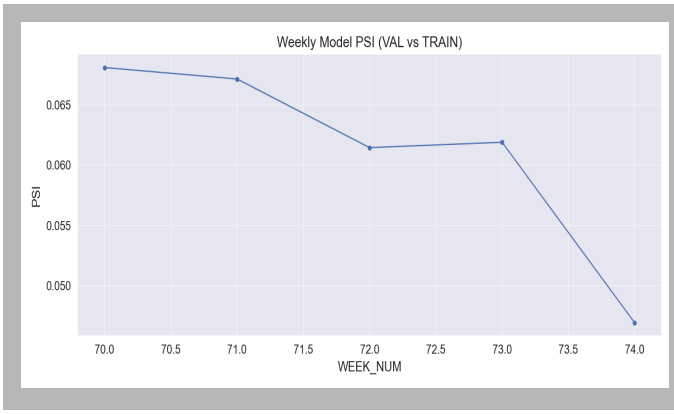


Figure 3 Weekly PSI(VAL to TRAIN)

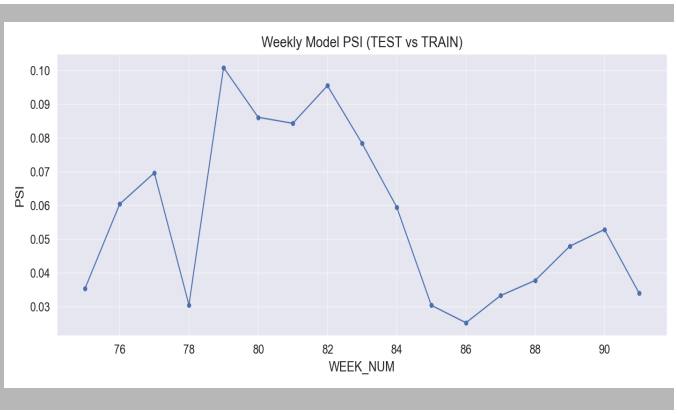


Figure 4 Weekly PSI(TEST to TRAIN)

Overall, weekly stability analysis confirms that the model maintains acceptable consistency, with only short-term variations observed.

7. Business Evaluation

7.1 Threshold & Approval Analysis

To translate model scores into actionable credit decisions, I analyzed the relationship between threshold, approval rate, and bad rate.

- On the **validation set**, the threshold–approval–bad-rate curve (Figure 5 Threshold-VAL) was used to identify a cut-off that balances portfolio growth with risk control.
- On the **test set**, the same cut-off was applied, and its performance was validated (Figure 6 Threshold-TEST).

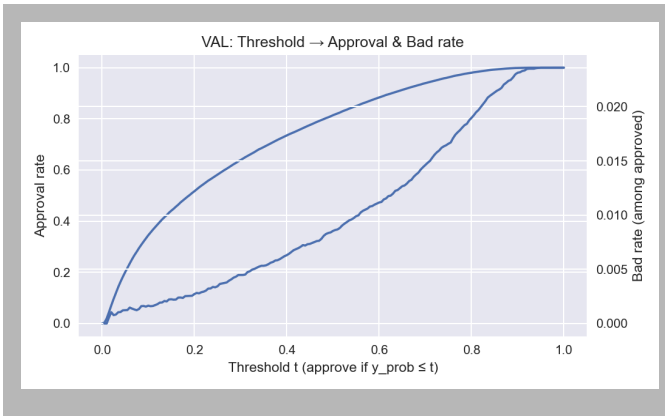


Figure 5 Threshold-VAL

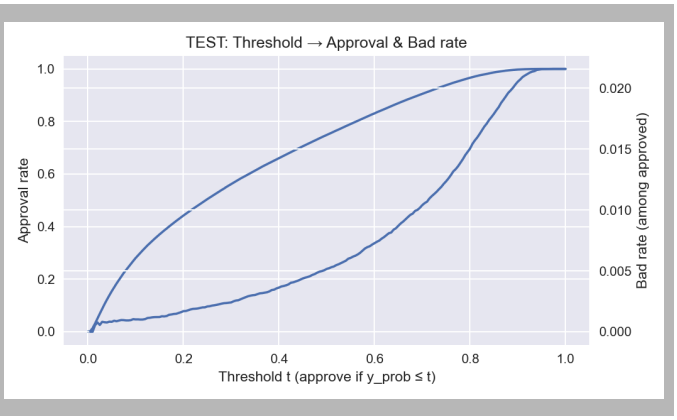


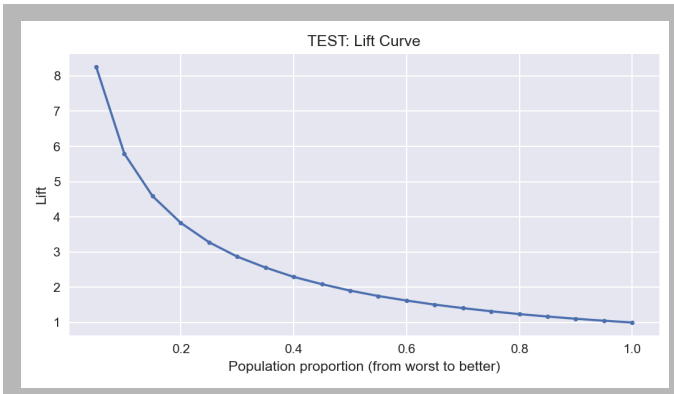
Figure 6 Threshold-TEST

This approach ensures that threshold selection is not overly tuned to historical data and remains effective when generalized to unseen periods.

7.2 Discriminatory Power (Lift & Gain)

To evaluate business value, I assessed the model's ability to prioritize high-risk clients.

- The **Lift Curve** (Figure 7 Lift Curve) demonstrates that the model achieves significantly higher default capture rates in the top deciles compared to random selection, confirming its discriminatory power.
- The **Cumulative Gain Chart** (Figure 8 Cumulative Gain) further shows that a relatively small fraction of the population accounts for a disproportionately large share of defaults, enabling more efficient allocation of risk exposure.



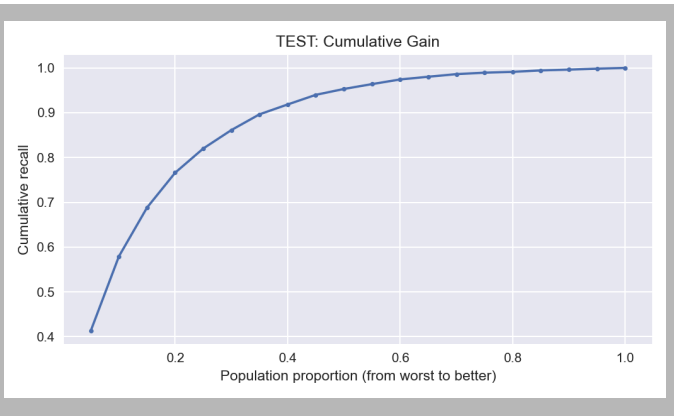


Figure 7 Lift Curve

Figure 8 Cumulative Gain

To complement the lift curve, we evaluated the model's performance at the top 5%, 10%, and 15% of predicted risk scores. Table 2 (test _PR_at_K) indicates that the model is highly effective at identifying risky customers in the top-ranked groups.

K%	topN	precision_at_K	recall_at_K
5%	9168	0.1783	0.4132
10%	18337	0.1249	0.5790
15%	27506	0.0990	0.6881

Table 2. test _PR_at_K

These results highlight that the model can materially improve credit decisioning by concentrating risk detection in the early scored segments.

7.3 Strategy Performance (Threshold & Bad-Rate Caps)

To turn the model results into practical business rules, I tested threshold-based approval strategies under different bad rate limits. The **threshold–approval–bad-rate curve** (Figure 5 Threshold-VAL) shows the balance between approval rate and bad rate across score cutoffs on the validation set. This curve helped determine operating points by matching risk tolerance with business needs.

On the test set, the same simulation (Figure 6 Threshold-TEST) shows that the chosen thresholds work well in practice, keeping bad rate under control while still allowing enough approvals.

A summary of the main scenarios is given in Table 3 (Overall business information). Each row represents a candidate threshold, with the related approval rate, bad rate, and default capture rate. The table makes the trade-offs clear: stricter thresholds lower portfolio risk but reduce approval coverage, while looser thresholds expand coverage but increase credit losses.

These findings provide business stakeholders with a clear and practical framework for setting decision rules in line with portfolio-level risk appetite. In addition, the weekly stability analysis attached to the table shows that the strategy's performance stays consistent over time, reducing concerns about temporal drift.

bad_cap	threshold	val_approve_rate	val_bad_rate_in_approved	val_bad_capture_rate	test_approve_rate	test_bad_rate_in_approved	test_bad_capture_rate
0.016	0.7318311	0.9542	0.0160	0.3532	0.9267	0.0116	0.5021
0.018	0.778016	0.9734	0.0180	0.2581	0.9547	0.0138	0.3907
0.02	0.8200516	0.9857	0.0200	0.1651	0.9756	0.0162	0.2661

Table 3. Overall business information

7.4 Stability Analysis

Across the three business caps (1.6%, 1.8%, 2.0%), the weekly bad rate in validation and test sets stayed under control, as shown in Table 4 (Strategy Table with Weekly Stability under Bad-rate Caps).

- At 1.6% cap, stability was strongest: validation went over the cap in 40% of weeks, but the test set never crossed it. The worst single-week bad rate was about 1.58%, with very low variation (std ≈ 0.20%).
- At 1.8% and 2.0% caps, stability became looser. The test set breached the cap in 5.9% and 17.6% of weeks. Even then, the highest bad rate (~1.8–2.1%) were only slightly higher than the cap.

bad_cap	threshold	val_weeks_over_cap	val_worst_week	val_std	test_weeks_over_cap	test_worst_week	test_std
0.016	0.7318	0.4	0.0184	0.0017	0	0.0158	0.0021
0.018	0.7780	0.2	0.0207	0.0017	0.0588	0.0181	0.0023
0.02	0.8201	0.2	0.0231	0.0018	0.1765	0.0214	0.0027

Table 4. Strategy Table with Weekly Stability under Bad-rate Caps

The stability of the approved population was also assessed using PSI, as summarized in Table 5 (Approved PSI Stability Table). Median weekly PSI stayed below 0.06 across all caps, which is within the "no significant shift" level (<0.1). The maximum weekly PSI values were also below 0.10 (0.087–0.093), showing that even in the worst weeks, the distribution did not change much compared with validation. Overall PSI values (~0.028–0.035) confirm that the approved population kept stable patterns over time.

bad_cap	threshold	overall_PSI_approved	median_week_PSI_approved	max_week_PSI_approved
0.016	0.7318	0.0277	0.0480	0.0873
0.018	0.7780	0.0310	0.0518	0.0895

0.02	0.8201	0.0346	0.0544	0.0928

Table 5. Approved PSI Stability Table

The weekly stability analysis attached to the table shows that the strategy's performance stays consistent over time, reducing concerns about temporal drift.

Based on the current analysis, I recommend using a 1.8% cap, as it delivers the highest return under the assumption of equal loan amounts per case. This recommendation holds as long as the per-case profit remains within a normal range. If detailed loan amount data were available, a more refined profit and risk analysis could be carried out.

8. Feature Analysis

8.1 Overall Feature Analysis

First, I used SHAP to identify the ten most important features and carried out an overall analysis. In most cases, the top ten already explain the majority of the model. The combined stability and predictive diagnostics are summarized in Table 6 (Top10 SHAP Feature PSI Stability Table).

On stability (PSI), most features such as **mean_dateofcredstart_739D**, **avgdpdtolclosure24_3658938P**, and **mobilephncnt_593L** remain highly stable across validation and test. In contrast, **mean_refreshdate_3813885D** and **max_amount_4527230A** show extremely high PSI, suggesting severe distribution drift.

On discriminatory power (KS), features like **mean_refreshdate_3813885D** reach very high KS (>0.95), but this strength may mislead based on their instability. More reliable drivers such as **mean_dateofcredstart_739D** and **pmtnum_254L** maintain moderate KS while also being stable.

On information value (IV), most features retain positive and consistent contributions across samples, though **max_sex_738L_M** shows inconsistent IV between validation and test, hinting at sample imbalance rather than genuine signal.

Taken together, the majority of top drivers combine low PSI with steady KS and IV, confirming both stability and predictive reliability. A small number of features, however, are unstable or inconsistent, especially, two features, **mean_refreshdate_3813885D** and **max_amount_4527230A**, which show clear distribution shifts with very high PSI, making them risk signals that require close monitoring or replacement.

feature	psi_train_to_val	psi_train_to_test	ks(train-val)	ks(train-test)	iv(val-train)	iv(test-train)
mean_dateofcredstart_739D	0.0332	0.0448	0.0443	0.0730	0.0610	0.0705
max_sex_738L_M	5.0558	0.0031	0.3795	0.0269	-0.0390	0.0233
mean_refreshdate_3813885D	11.1747	11.1946	0.9577	0.9589	-0.0192	-0.0128
avgdpdtolclosure24_3658938P	0.0018	0.0020	0.0188	0.0274	0.0534	0.0954
mobilephncnt_593L	0.0234	0.0229	0.0679	0.0675	-0.0189	0.0076

mean_dpdmax_139P	0.0029	0.0040	0.0150	0.0306	-0.0141	-0.0140
pmtnum_254L	0.0625	0.0305	0.0879	0.0605	0.0314	0.0468
mean_overdueamountmax_155A	0.0005	0.0011	0.0010	0.0261	0.0199	-0.0282
mean_birth_259D	0.1013	0.1023	0.1360	0.1301	0.1883	0.1107
max_amount_4527230A	1.0493	1.1101	0.3360	0.3381	-0.0054	-0.0163

Table 6. Top10 SHAP Feature PSI Stability Table

8.2 Unstable Feature Analysis

Within the SHAP top 10, we focused on mean_refreshdate_3813885D and max_amount_4527230A for deeper investigation, as their instability indicates potential model drift and higher business risk if left unmonitored.

8.2.1 mean_refreshdate_3813885D

The feature *mean_refreshdate_3813885D* represents the average date when the credit bureau's public sources were last updated, which reflects the recency of external credit data availability. According to Figure 7 (mean SHAP per bin of mean_refreshdate_3813885D), the longer the relative time since the last refresh, the lower the risk contribution. This aligns with business logic: if a long period has passed since the last update, it suggests no new overdue events have been recorded, indicating lower risk. In addition, Figure 8 (mean weekly SHAP of mean_refreshdate_3813885D) shows that its contribution in validation and test sets remains stable, proving that the feature continues to be effective. The abnormal PSI, KS, and IV values are therefore more likely caused by differences in the data distribution, possibly due to irregular updates in the credit bureau's public sources, rather than by a real loss of predictive power. Thus, this feature should not be discarded but closely monitored in future use.

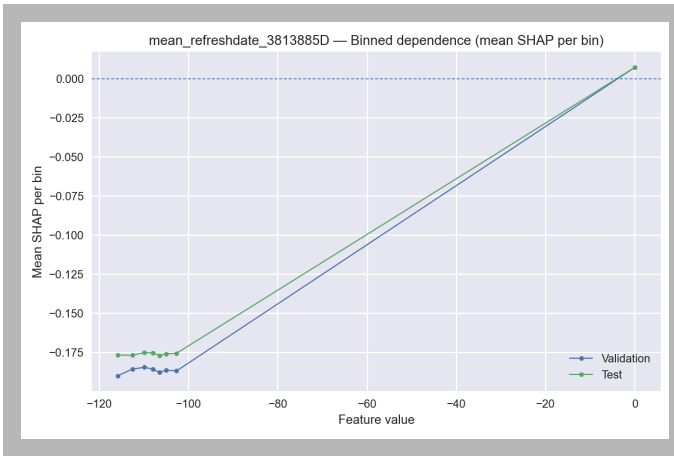


Figure 7 mean SHAP per bin(mean_refreshdate_3813885D)

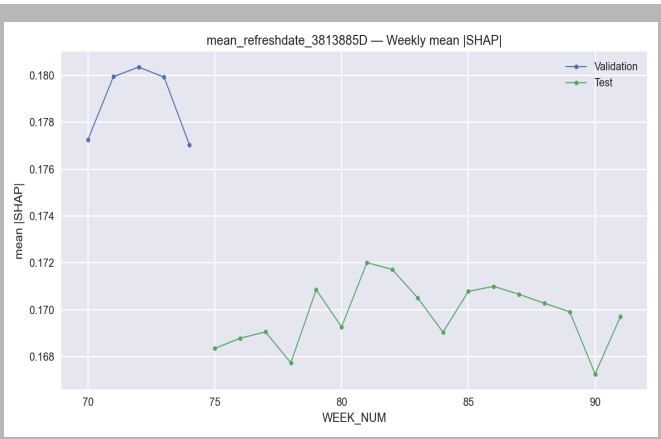


Figure 8 mean SHAP weekly(mean_refreshdate_3813885D)

8.2.2 max_amount_4527230A

max_amount_4527230A represents the largest tax deduction amount tracked by the government registry. According to Figure 9 (mean SHAP per bin), the feature contributes consistently to risk prediction, but the lack of dispersion in values limits its discriminative power. In practice, the largest deduction is often dominated by housing-related items, which explains why most values are concentrated in a narrow range. Figure 10 (weekly mean SHAP) further shows that, although there are fluctuations across test weeks, the contribution remains within a relatively stable band. The high PSI and KS indicate dataset-level

distributional shifts, likely reflecting structural changes in how the government registry reports or records deduction amounts, or social factors such as housing price changes, rather than a breakdown of the feature's effect. Thus, this variable should be treated as a risk signal that requires monitoring for reporting consistency and market sensitivity, but not immediately discarded.

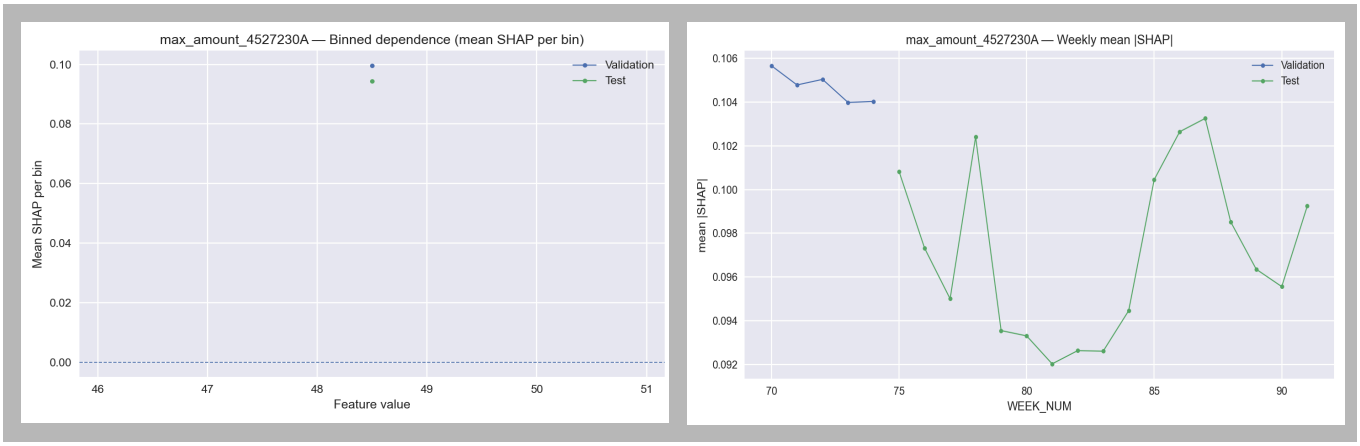


Figure 9 mean SHAP per bin(max_amount_4527230A)

Figure 10 mean SHAP weekly(max_amount_4527230A)