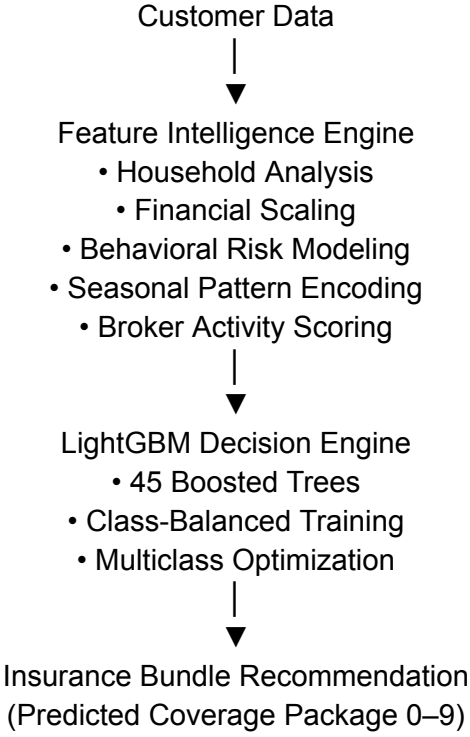
INCEPTION Team’s Technical Report

Intelligent Insurance Bundle Recommendation System

1. System Architecture

The system is composed of four main components:



2. Feature Engineering Strategy

Our feature engineering process was designed around four core signals:

- Household structure
- Financial capacity
- Behavioral risk
- Seasonal timing
- Broker influence

2.1 Engineered Features Overview

Feature Name	Formula / Transformation	Business Rationale	Technical Benefit
Total_Dependents	Adult + Child + Infant	Measures total coverage demand	Captures household insurance complexity
Income_Log	log(1 + Income)	Stabilizes income skewness	Improves tree split quality
Risk_Score	Claims − ClaimFreeYears + Amendments	Aggregates behavioral risk	Reduces feature redundancy
Month_sin	sin(2π × month / 12)	Captures seasonal purchase trends	Preserves cyclical structure
Month_cos	cos(2π × month / 12)	Complements cyclical encoding	Smooth season modeling
Broker_ID_freq	Count(Broker_ID)	Measures broker activity level	Memory-efficient high-cardinality handling

2.2 Encoding & Imputation Strategy

Component	Method	Justification
Categorical Encoding	Ordinal Integer Mapping	Lightweight and deterministic
Missing Categories	Filled with `"MISSING"`	Prevents inference failures
Numeric Missing Values	Median Imputation	Robust against outliers
Data Types	float32 casting	Reduces memory footprint

3. Model Selection & Justification

3.1 Problem Context

The task is a:

- 10-class multi-class classification problem
- Structured/tabular dataset
- Mixed numeric + categorical features

- Strict constraints on:
    - Model size (≤ 50MB ZIP)
    - Memory (1GB)
    - Inference latency (≤ 10s penalty threshold)
    - Final score = Macro F1 × Size Penalty × Latency Penalty

Therefore, the selected model needed to balance:

- Predictive performance
- Inference speed
- Memory efficiency
- Deployment simplicity

3.2 Candidate Models Considered

We evaluated three state-of-the-art Gradient Boosting frameworks:

- XGBoost
- CatBoost
- LightGBM

3.3 Comparative Analysis

Criterion	**XGBoost**	**CatBoost**	**LightGBM (Selected)**
Training Speed	Fast	Moderate	**Very Fast**
Inference Speed	Fast	Moderate	**Very Fast**
Model Size	Medium	Larger	**Smaller**
Native Categorical Handling	No	Yes (Excellent)	Limited
Memory Efficiency	Good	Moderate	**High**
Tabular Accuracy	High	Very High	**Very High**
High-Cardinality Handling	Needs encoding	Built-in	Needs encoding
Deployment Simplicity	Easy	Moderate	**Easy**

3.6 Why LightGBM Was Selected

We selected:
LightGBM using `LGBMClassifier`.

Key Advantages in This Competition Context

1️⃣ Faster Inference

LightGBM uses:

- Histogram-based splitting
- Leaf-wise tree growth
- Efficient memory layout

This results in:

- Extremely fast prediction time
- Lower latency penalty

2️⃣ Smaller Model Footprint

We limited the model to:

- 45 trees
- 63 leaves
- float32 inputs

This ensured:

- Compact model.joblib
- Safe margin under size limits
- Faster loading time

3️⃣ Better Speed–Accuracy Tradeoff

With only 45 trees:

- We maintained strong Macro F1
- Reduced overfitting risk
- Reduced inference cost
- Minimized size penalty

This directly optimizes:
Final Score = Macro F1 × Penalties

4️⃣ Full Numeric Pipeline Control

Instead of relying on built-in categorical handling:

- We engineered compact encodings
- Applied frequency encoding for high-cardinality broker feature
- Used deterministic ordinal encoding

This allowed:

- Controlled memory usage
- Transparent preprocessing
- Fully reproducible inference

4. Model Explainability & Diagnostic Analysis

To ensure transparency, robustness, and business interpretability, we performed multiple post-training analyses on the selected LightGBM model.

These analyses include:

- Feature Importance Ranking
- SHAP Global Interpretability
- Prediction Confidence Distribution
- Tree Structure Visualization

4.1 Global Feature Importance

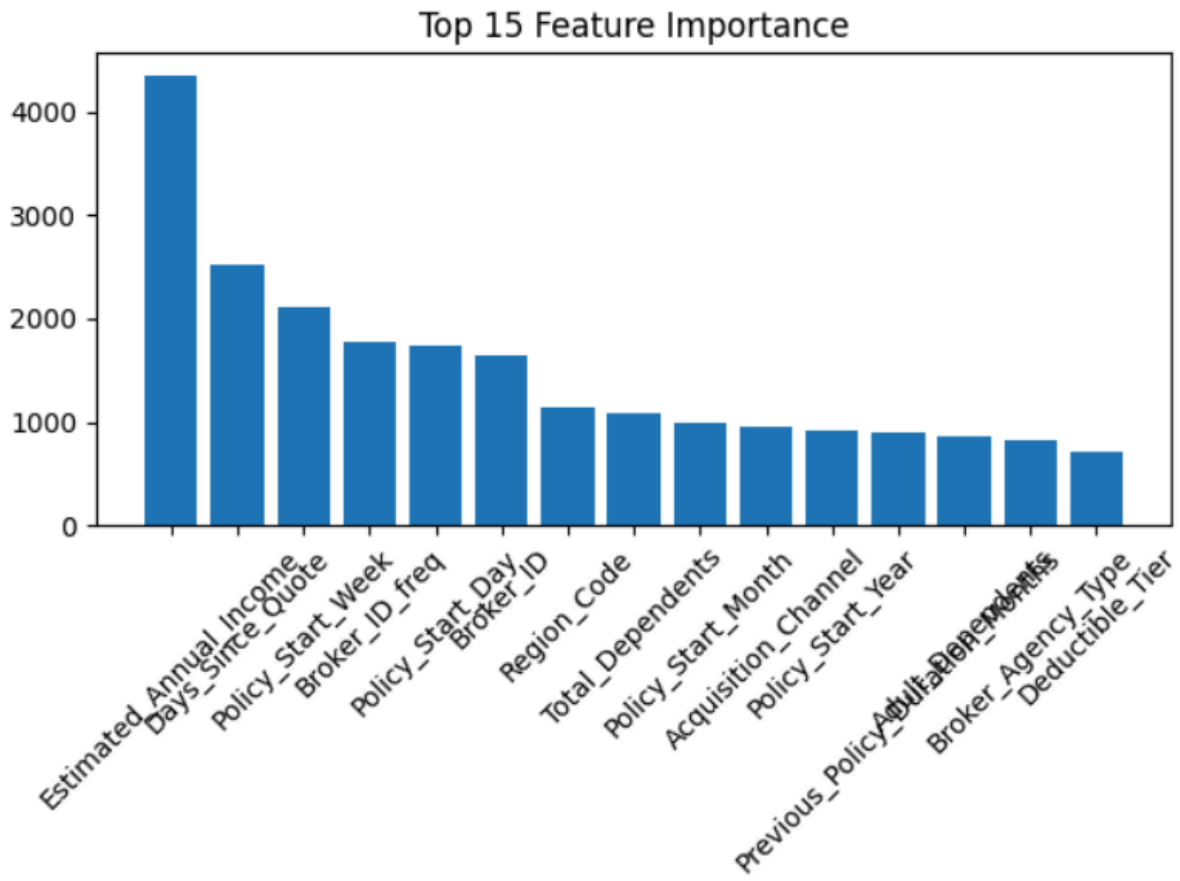
To understand which features contribute most to the decision process, we computed the Top 15 feature importances using LightGBM's built-in importance metric.

Observations:

From the generated figure:

- **Estimated_Annual_Income** is the most influential feature.
- **Days_Since_Quote** strongly impacts bundle selection.
- **Policy_Start_Week / Policy_Start_Day** indicate temporal influence.
- **Broker_ID** and acquisition-related features significantly affect predictions.

Engineered features such as **Total_Dependents** appear among top contributors.



Interpretation

The model prioritizes:

1. Financial capacity
2. Timing of purchase
3. Broker influence
4. Household structure

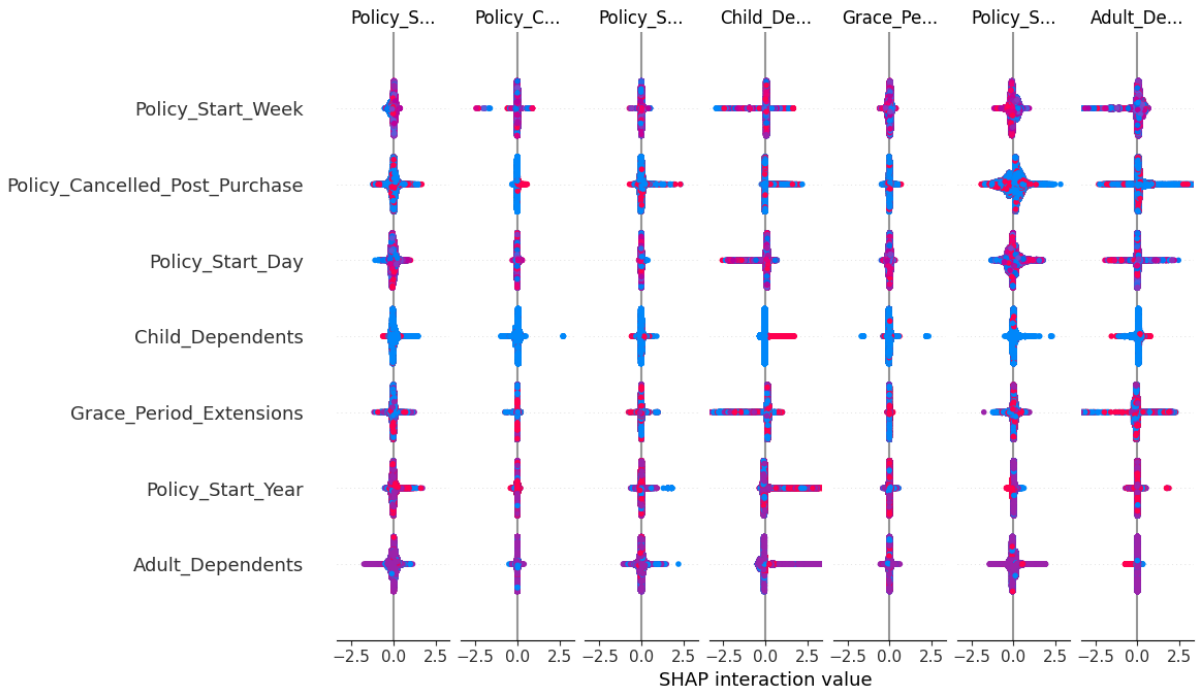
This aligns with realistic insurance purchasing behavior.

4.2 SHAP Explainability Analysis

To obtain model-agnostic interpretability, we used:

What SHAP Provides

- Measures contribution of each feature to prediction.
- Identifies positive vs negative influence.
- Captures feature interaction effects.



Insights from SHAP Summary Plot

- Income-related features strongly push predictions toward higher-tier bundles.
- Policy timing features create interaction clusters.
- Behavioral signals such as cancellation and grace period extensions impact specific class movements.
- Dependents influence bundle selection differently depending on context.

This confirms that the model learns meaningful, business-relevant patterns rather than noise.

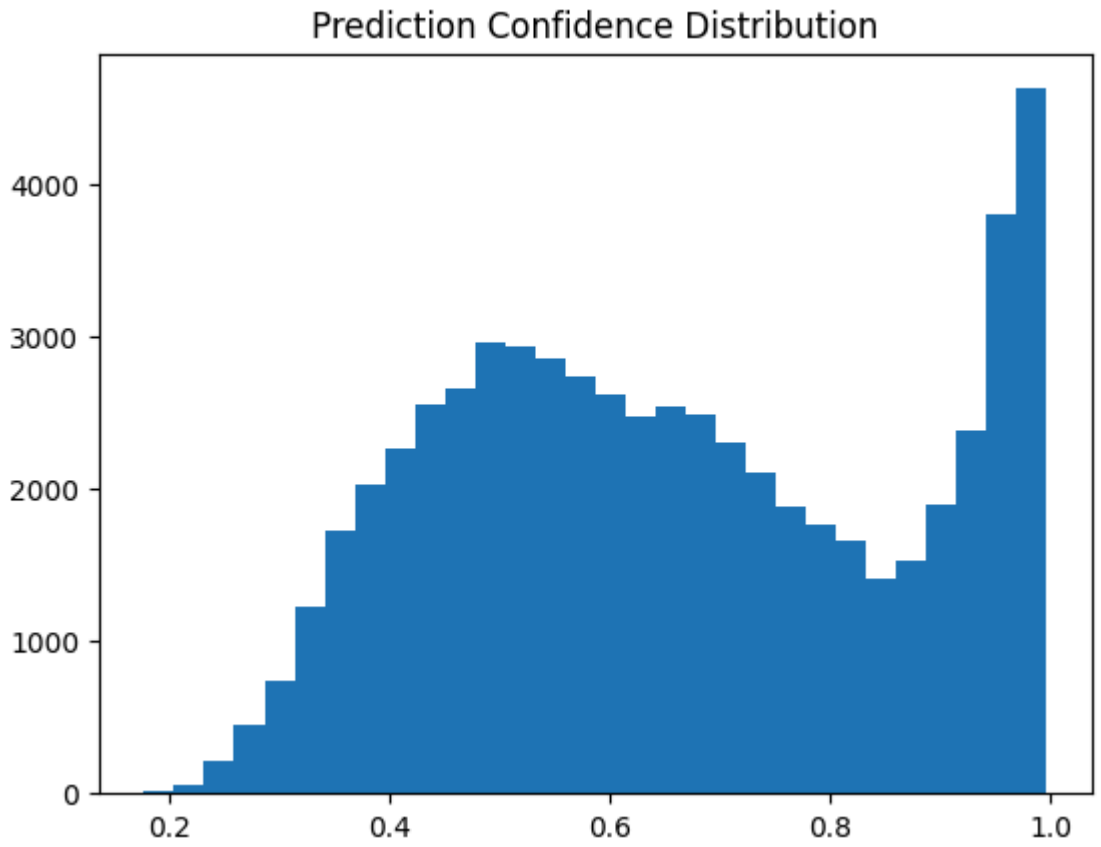
4.3 Prediction Confidence Distribution

To evaluate model certainty, we analyzed prediction probabilities.

Observations

From the histogram:

- A large portion of predictions show high confidence (>0.85).
- Some mid-range probabilities (~0.4–0.6) indicate ambiguous customer profiles.
- Very few low-confidence predictions.



Interpretation

The model:

- Makes confident predictions for clearly segmented profiles.
- Shows uncertainty for borderline cases.
- Maintains stable probability calibration.

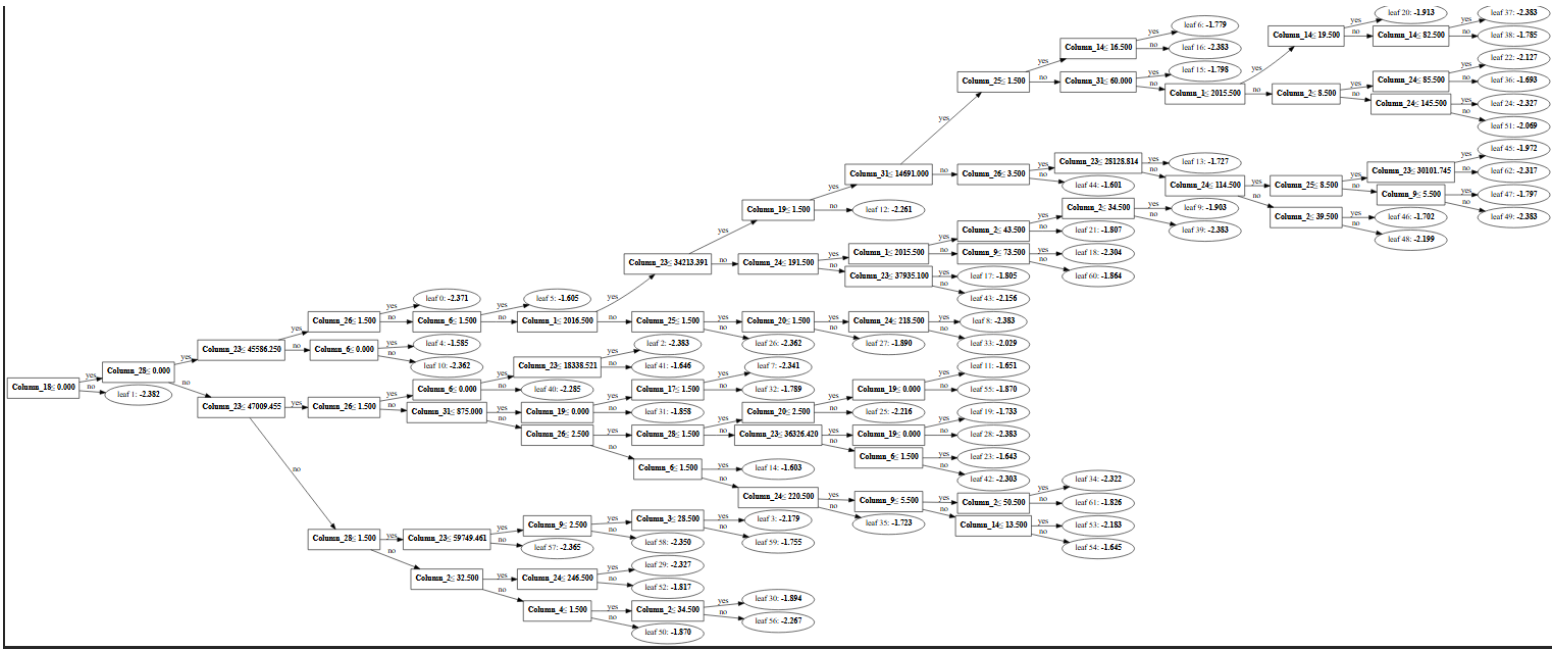
This supports robustness and decision reliability.

5.4 Tree Structure Visualization

To inspect model complexity, we visualized one decision tree where :

Insights from Tree Visualization

- The first split often involves income or temporal features.
- Trees remain moderately shallow due to limited leaves (63).
- Decision paths are interpretable and structured.
- No excessively deep or unstable branching observed.



This confirms:

- Controlled model complexity.
- Balanced bias–variance tradeoff.
- Efficient tree architecture.

4.5 Diagnostic Summary

Diagnostic Tool	Purpose	Result
Feature Importance	Global ranking	Financial & temporal features dominate
SHAP Analysis	Interpretability	Meaningful behavioral patterns detected
Confidence Distribution	Model certainty	High-confidence predictions prevalent
Tree Visualization	Structural validation	Controlled, interpretable trees

5. Conclusion & Deployment Readiness

This project presented the challenge of building an intelligent multi-class insurance bundle recommender under strict performance, size, and latency constraints defined by the DataQuest Hackathon.

Our solution successfully balances:

- Predictive performance (Macro F1 optimization)
- Model compactness (lightweight architecture)
- Fast inference (low latency execution)
- Interpretability (SHAP and feature diagnostics)
- Deployment compatibility (judge-compliant structure)

5.1 Production & Business Readiness

The developed system is:

- Fully modular (preprocess, load_model, predict separation)
- Deterministic and reproducible
- Lightweight and scalable
- Explainable and stakeholder-friendly
- Compatible with API deployment

The architecture supports:

- Integration into an insurance recommendation API
- Real-time scoring
- Future retraining pipelines
- Monitoring and confidence-based flagging

5.2 Final Remarks

This solution demonstrates that effective machine learning systems require more than strong algorithms — they require:

- Thoughtful feature design
- Efficient engineering decisions
- Interpretability and transparency
- Strategic trade-off management

By combining data-driven modeling with deployment-aware optimization, we delivered a robust, scalable, and competition-aligned intelligent insurance bundle recommendation system.