Machine Learning in Home Insurance

Loss Modeling ▪ Geospatial Engineering ▪ SageMaker Pipelines

yuyaow@bu.edu

Core Themes

Satellite Imagery ▪ XGBoost Tweedie ▪ Street View LLM

PySpark + SQL ▪ AWS SageMaker

Agenda

Home Insurance Loss Modelling

A **loss model** estimates expected claim cost per policy:

$$\mathbb{E}[\text{Loss}] = \underbrace{P(\text{claim})}_{\text{frequency}} \times \underbrace{\mathbb{E}[\text{cost} \mid \text{claim}]}_{\text{severity}}$$

Accurate models enable:

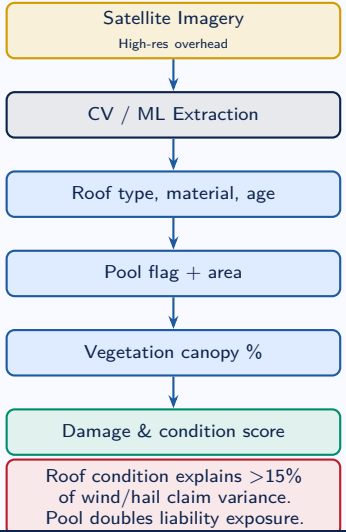
- Risk-adequate **pricing**
- Avoiding **adverse selection**
- Better **reinsurance** optimisation
- Sharper **underwriting** segmentation

The Core Challenge

- Policies run 12 months — feedback is *slow*
- Physical risk is *invisible* in traditional data
- Legacy GLMs miss non-linear interactions
- Geospatial signals were **untapped**

Our Goal Combine **satellite imagery**, **Street View LLM**, and **policyholder data** in a production-grade ML loss model on 50M+ rows.

Category	Signals
Roof	Type (gable/hip/flat), material, condition score, age estimate
Structure	Footprint area, shape complexity, outbuildings
Amenities	Pool presence & area, hot tub, trampoline, deck/patio
Vegetation	Tree canopy density, slope, proximity to water
Construction	Siding type, visible damage, solar panels
Lot	Lot size, impervious surface ratio, fence presence



Demographic & Lifestyle Signals

Domain	Features
Household	Marital status, household size
Children	Count, age range of dependents
Income	Estimated band, wealth proxy
Employment	Occupation type, tenure
Tenure	Years at property, policy age
Credit	Financial stability index
Prior Claims	Frequency, severity, recency

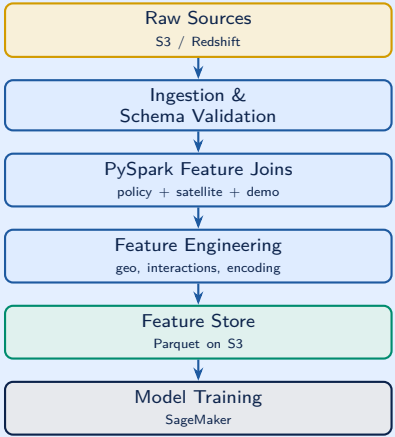
Policy & Coverage Features

- Dwelling replacement cost, coverage limits
- Deductible: all-peril, wind, hurricane
- Policy type: HO-3, HO-5, condo, renters
- Endorsements: water backup, scheduled items
- Channel and agent segment

Feature Store Architecture

Online: AWS Feature Store — real-time scoring
Offline: S3 Parquet lake — Spark training
Registry: Point-in-time joins, no leakage

Pipeline Architecture



Scale Facts

- **50M+ rows** per training cohort
- EMR Spark with **auto scaling**
- Point-in-time joins via **Spark SQL** windows
- Geo joins with **H3 hex indexing**
- Parquet partitioned by **state** × **year**

Engineering Choices

- **Broadcast joins** for lookup tables
- **Salting** to reduce skew in FL, TX, CA
- **Delta Lake** for versioned features
- SQL `QUALIFY` dedup on policy snapshots

Feature	Construction
Peril Zones	NOAA wind / hail / flood grids joined by lat-lon
Wildfire	Distance to WUI boundary; USFS fire-risk tile
Flood Plain	FEMA 100/500-year zone flag; flood-way distance
Micro-climate	Elevation, coastal proximity, precip normals
Neighbourhood	H3 claim frequency with empirical Bayes smoothing
Crime Index	FBI / local crime index aggregated to H3 hex

Key Interaction Features

Roof age × Hail zone

Pool × Children

H3 Spatial Smoothing

Claim rate intensity

H3 Resolution 8 $\approx$ 0.74 km^2 per hex. Empirical Bayes shrinkage stabilizes low-exposure hexagons.

Why Tweedie?

Losses follow a compound Poisson–Gamma model:

$$L = \sum_{i=1}^{N} C_i, \qquad N \sim \text{Pois}, \; C_i \sim \text{Gamma}$$

Tweedie with $p \in (1, 2)$ naturally handles:

- excess zeros (most policies have no claim)
- heavy right tail (large severities)
- pure premium in a single model

Why not a two-part model?

A single Tweedie objective lets us model **frequency and severity jointly**, while preserving the business target of **expected loss cost per policy**.

Interpretation

- N captures whether claims occur
- C_i captures claim size
- their sum gives total policy loss
- Tweedie is well aligned with insurance pricing

Practical fit for P&C pricing

Zero-inflated outcomes, skewed severities, and exposure-weighted training all arise naturally in home insurance loss modeling.

XGBoost Config

`objective`	`reg:tweedie`
`tweedie_var_power`	1.5 (CV-tuned)
`tree_method`	`hist` (GPU)
`n_estimators`	1,200
`max_depth`	6
`learning_rate`	0.05
`subsample`	0.80

Training Strategy

- stratified k-fold by **state + AY**
- earned exposure as **sample weight**
- **Optuna** HPO with 200 trials
- monotonic constraints on intuitive features

Model Comparison

Model	Gini	RMSE
Legacy GLM	42.1	1,842
XGB (no geo)	44.8	1,790
XGB + geo	48.3	1,731
XGB + all	**49.1**	**1,708**

Adding geospatial and engineered features improved discrimination by roughly **+7 Gini points** over the legacy GLM, while also reducing RMSE.

Why Street View?

Satellite imagery is top-down. Street View adds frontal cues that are often highly relevant for risk:

- façade condition and deferred maintenance
- visible structural damage or weathering
- garage type: attached, detached, or carport
- driveway material and condition
- clutter and outdoor storage

Why it matters for underwriting

These signals are difficult to recover from roofline or parcel data alone, but they can materially improve property-level assessment of condition, inspection need, and loss propensity.

Complementary modality

Satellite explains parcel context. Street View explains visible property condition.

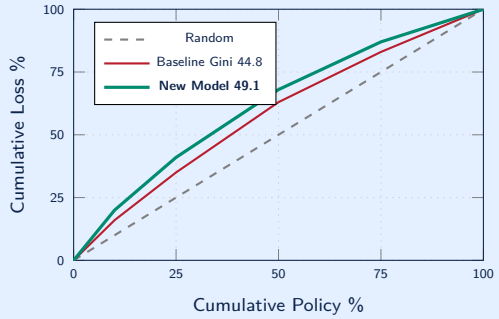
Typical examples

peeling exterior, damaged roof edge, deteriorated driveway, cluttered yard, visible maintenance neglect

Model Performance: +4.3 pt Gini Improvement

Out-of-time holdout evaluation

Lorenz Curve — Baseline vs. Challenger



Satellite + LLM + geospatial features account for ≈**55%** of total model gain.

Feature Importance (Top 10 by Gain)

#	Feature	Gain
1	H3 neighbourhood loss rate	14.2%
2	Roof age (satellite)	11.8%
3	Peril zone index (wind)	9.4%
4	Replacement cost	8.1%
5	Roof type (satellite)	6.7%
6	Facade condition (LLM)	5.3%
7	Prior claim frequency	4.9%
8	Tree canopy density	4.1%
9	Income band	3.8%
10	Pool flag × children	3.2%

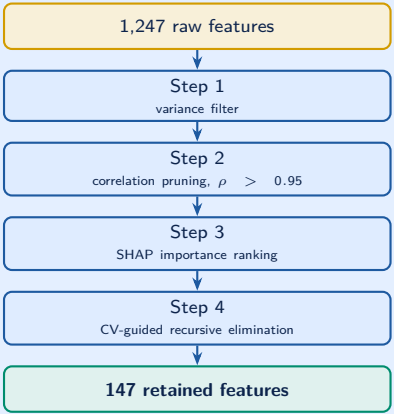
The Problem

- more than **1,200** raw columns after encoding and expansions
- collinearity and noisy features increase training cost
- inference latency budget was $<$ **50 ms** at p99

Goal

Reduce dimensionality aggressively while preserving ranking power and keeping model behavior stable across refreshes.

4-Step Compression Pipeline



Feature Compression: Results and Automation

Large latency gains with minimal loss in model lift

Results

Metric	Before	After
Features	1,247	**147**
Training time	4.2 h	**38 min**
Inference p99	210 ms	**42 ms**
Gini change	—	−0.3 pt

Compression reduced inference latency by about **80%**, with only a marginal loss in Gini.

Automation Details

- triggered **nightly** on refreshed feature snapshots
- SHAP computed on a held-out **validation set**
- all decisions logged to **MLflow**
- alert raised if retained count shifts by more than **10%**

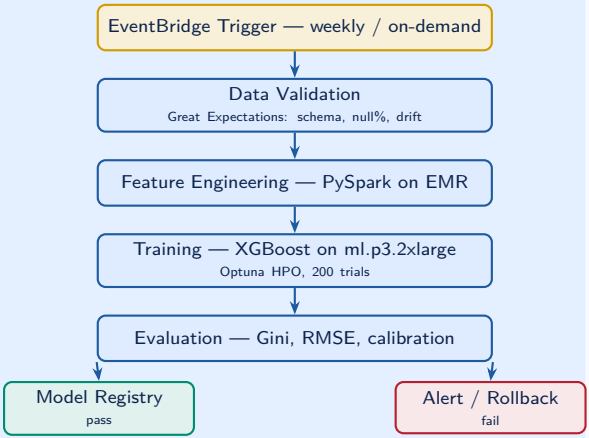
Operational Benefit

The compressed feature set made the model cheaper to retrain, faster to serve, and easier to govern in production.

AWS SageMaker Retraining Pipeline

Fully automated weekly retraining with champion/challenger

Pipeline DAG



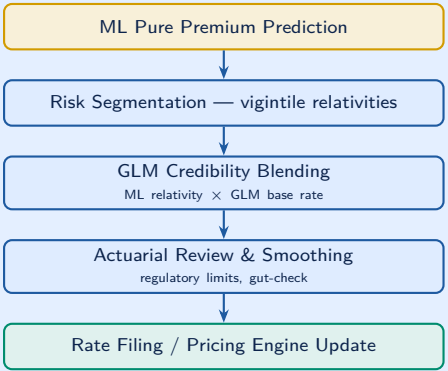
Key Design Decisions

- **Shadow deploy**: 4-week parallel run before cutover
- **A/B traffic**: 5% to challenger via feature flag
- **Auto rollback** if Gini drops > 0.5 pt
- **PSI drift** monitoring weekly on key features
- All artefacts versioned in S3 + MLflow

Infrastructure Stack

EMR Spark 3.4 · SageMaker Pipelines
Step Functions · Lambda · SNS
S3 Data Lake · Redshift Spectrum
CloudWatch · MLflow

ML Output → Rate Filing



LLM Score → Pricing Factor

Score	Condition	Factor
5	Excellent	0.90
4	Good	0.97
3	Fair	1.00
2	Poor	1.12
1	Very poor	1.28

Regulatory Requirements

All ML factors **filed with state DOI**. Explainability via **SHAP additive decomposition** — each factor traceable to an input feature.

Quantitative Gains

Metric	Before	After
XGBoost Gini	44.8	**49.1**
Gini vs. GLM	+2.7 pt	+7.0 **pt**
Calibration slope	0.91	**0.98**
RMSE (pure prem.)	1,790	**1,708**
Training cadence	Quarterly	**Weekly**
Retraining time	4.2 h	**38 min**
Feature count	1,247	**147**

Business Outcomes

- **Loss ratio**: better-priced book reduces adverse selection
- **Inspection targeting**: LLM flags route only high-risk properties
- **Underwriting automation**: ML score enables straight-through processing
- **Actuarial velocity**: weekly refresh replaces annual GLM cycle

Lessons Learned & Future Directions

Lessons Learned

1. **Geospatial = biggest unlock.** Satellite + geo added ≈3.5 Gini pts alone.
2. **Compression is non-optional.** 1,200+ features tripled cost; SHAP pruning cost < 0.3 pt Gini.
3. **LLM quality requires QA gates.** Confidence thresholds + audit sample review mandatory.
4. **Explainability must be designed in,** not bolted on — SHAP from day one for DOI filings.
5. **Shadow deployment prevents catastrophe** in live pricing — always run champion/challenger.

Future Directions

- Foundation model embeddings from satellite (geospatial transformers)
- Multi-peril sub-models: wind, water, fire, liability — stacked
- Real-time feature refresh via Kinesis + online feature store
- Causal inference to disentangle selection bias
- Federated learning across state books for low-exposure markets

Key Takeaway

Satellite + Street View LLM + geospatial engineering in a fully automated SageMaker pipeline delivered **+4.3 pt Gini** and a **6× faster** retraining cycle — moving actuarial from annual to **continuous learning**.