NYOHAKI

NYOHAKI GROUP

Lead Generation AI

Insurance Lead Scoring Models Documentation

Table of Contents

Overview

The Insurance Lead Scoring System is a comprehensive machine learning platform designed to evaluate and score insurance leads across multiple product lines. The system provides accurate, fair, and compliant lead scoring with specialized models for different insurance types.

System Architecture

- Insurance Lead Scoring System
- Base Insurance Model (Foundation)
- Healthcare Insurance Model (Specialized)
- Life Insurance Model (Specialized)
- API Layer (FastAPI).
- Evaluation Framework
- Compliance Engine

Key Features

- Multi-Product Support: Base, Healthcare, and Life Insurance
- Real-time Scoring: Sub-second response times
- Compliance-First: GDPR, CCPA, TCPA compliant
- Bias Detection: Automated fairness monitoring
- Interpretable Results: Feature importance and reasoning
- Scalable Architecture: Batch and real-time processing

Base Insurance Lead Scoring Model

Purpose

The base model serves as the foundation for all insurance lead scoring, providing core functionality and common features across insurance products.

Model Details

- Algorithm: XGBoost Regressor
- Output: Conversion probability score (0-100)
- Training Data: General insurance leads with consent
- Update Frequency: Monthly retraining

Core Feature

python

```python
feature_columns = [
    'age', 'income', 'location', 'employment_status', 'credit_score',
    'previous_insurance', 'contact_method', 'lead_source', 'time_of_contact',
    'interaction_history', 'demographic_segment', 'risk_profile'
]
```

Feature Engineering

- Risk Assessment: Credit score normalization and risk categorization
- Demographic Scoring: Age-income interaction effects
- Behavioral Patterns: Contact timing and interaction history
- Lead Quality: Source quality and engagement metrics

Model Performance

- Target Accuracy: $R^2 > 0.85$
- Response Time: < 100ms per lead
- Batch Processing: 10,000+ leads per minute
- Fairness Threshold: Demographic parity < 0.1

Usage Example

Python

```python
from models.insurance_lead_scoring.inference import InsuranceLeadScorer

scorer = InsuranceLeadScorer()
result = scorer.score_lead({
    'lead_id': 'BASE_001',
    'age': 35,
    'income': 75000,
    'employment_status': 'employed',
    'credit_score': 720,
    'consent_given': True
})
```

Healthcare Insurance Lead Scoring Model

Purpose
Specialized model for healthcare insurance leads, incorporating health-specific factors and regulatory compliance requirements.

Model Details
- Algorithm: XGBoost Regressor (Healthcare-optimized)
- Output: Healthcare conversion score (0-100) + health insights
- Training Data: Healthcare insurance leads with HIPAA compliance
- Update Frequency: Bi-weekly retraining

Healthcare-Specific Features

```python
healthcare_features = [
    'age', 'income', 'family_size', 'employment_status', 'current_insurance_status',
    'health_conditions', 'prescription_medications', 'healthcare_utilization',
    'preferred_provider_network', 'coverage_preferences', 'deductible_preference',
```

'chronic_conditions', 'preventive_care_usage', 'specialist_needs'
]

Advanced Feature Engineering

- Health Risk Scoring: Chronic conditions and medication analysis
- Network Preferences: Provider network alignment scoring
- Cost Sensitivity: Deductible and premium preference modeling
- Utilization Patterns: Healthcare usage prediction
- Family Coverage: Dependent coverage needs assessment

Healthcare-Specific Outputs

```python
{
    'score': 87.5,
    'health_risk_category': 'MODERATE',
    'recommended_plan_type': 'PPO',
    'estimated_annual_cost': 8400,
    'network_preference_match': 0.92,
    'chronic_condition_coverage': ['diabetes', 'hypertension'],
    'preventive_care_score': 8.2,
    'urgency_level': 'HIGH'
}
```

Compliance Features

- HIPAA Compliance: Health data anonymization and encryption
- ACA Compliance: Non-discrimination in health status
- State Regulations: State-specific healthcare requirements
- Privacy Protection: PII masking and secure processing

Performance Metrics

- Overall Accuracy: $R^2 > 0.88$
- High-Risk Accuracy $R^2 > 0.85$ for chronic conditions
  Family Plan Accuracy: $R^2 > 0.90$ for family coverage
- Network Match Accuracy: 94% provider preference alignment

Life Insurance Lead Scoring Model

Purpose

Specialized model for life insurance leads, incorporating mortality risk assessment, coverage adequacy analysis, and life stage considerations.

Model Details

- Algorithm: XGBoost Regressor (Life Insurance-optimized)
- Output: Life insurance score (0-100) + actuarial insights
- Training Data: Life insurance leads with actuarial data
- Update Frequency: Monthly retraining with mortality table updates

Life Insurance Features

```python
life_insurance_features = [
    'age', 'income', 'marital_status', 'dependents_count', 'employment_status',
    'health_status', 'smoking_status', 'coverage_amount_requested', 'policy_term',
    'existing_life_insurance', 'beneficiary_count', 'debt_obligations',
    'mortgage_balance', 'education_level', 'occupation_risk_level',
    'life_stage', 'financial_dependents', 'estate_planning_needs'
```

]
```

**Actuarial Feature Engineering**

- Mortality Risk Scoring: Age, health, smoking, occupation risk
- Coverage Adequacy: Income replacement and debt coverage analysis
- Life Stage Analysis: Young professional to estate planning stages
- Financial Responsibility: Dependents and debt obligations
- Estate Planning: High-net-worth and inheritance considerations

**Life Insurance Outputs**
```python
{
    'score': 92.3,
    'life_stage': 'family_building',
    'mortality_risk_score': 3.2,
    'recommended_coverage': 750000,
    'coverage_adequacy': 'adequate',
    'coverage_gap': 0,
    'recommended_policy_type': 'TERM_LIFE',
    'urgency_level': 'CRITICAL',
    'estate_planning_urgency': 4.5
}
```

Policy Recommendations

- Term Life: Young families, temporary needs
- Whole Life: Estate planning, permanent coverage
- Universal Life: Flexible premiums, investment growth
- Variable Life: Investment-minded, high-income clients

Actuarial Compliance

- Mortality Tables: Industry-standard life expectancy data
- Non-Discrimination: Fair underwriting practices
- State Regulations: Insurance commissioner requirements
- Actuarial Fairness: Evidence-based risk assessment

Performance Metrics

- Overall Accuracy: $R^2 > 0.87$
- Family Building Accuracy: $R^2 > 0.92$ for family stage
- High Coverage Accuracy: $R^2 > 0.89$ for $500k+ policies
- Policy Recommendation: 91% accuracy in policy type matching

- API Documentation
- Base Insurance API
- Endpoint: `/score-lead`

```http
POST /score-lead
Content-Type: application/json

{
  "lead_id": "BASE_001",
  "age": 35,
  "income": 75000,
  "employment_status": "employed",
  "credit_score": 720,
  "consent_given": true,
  "consent_timestamp": "2024-01-15T10:30:00Z"
}
```

Response:
```json
{
   "lead_id": "BASE_001",
   "score": 78.5,
   "confidence": 0.92,
   "risk_category": "MEDIUM",
   "timestamp": "2024-01-15T10:30:15Z",
   "model_version": "1.0_base",
   "compliance_status": "PASSED"
}
```

Healthcare Insurance API

Endpoint: `/score-healthcare-lead`
```http
POST /score-healthcare-lead
Content-Type: application/json

{
   "lead_id": "HEALTH_001",
   "age": 42,
   "income": 85000,
   "family_size": 4,
   "current_insurance_status": "uninsured",
   "health_conditions": ["diabetes", "hypertension"],
   "preferred_provider_network": "PPO",
   "consent_given": true
}
```

Response:

```json
{
    "lead_id": "HEALTH_001",
    "score": 87.5,
    "health_risk_category": "MODERATE",
    "recommended_plan_type": "PPO",
    "estimated_annual_cost": 8400,
    "network_preference_match": 0.92,
    "urgency_level": "HIGH",
    "compliance_status": "PASSED"
}
```

Life Insurance API

Endpoint: `/score-life-insurance-lead`
```http
POST /score-life-insurance-lead
Content-Type: application/json

{
    "lead_id": "LIFE_001",
    "age": 35,
    "income": 85000,
    "marital_status": "married",
    "dependents_count": 2,
    "health_status": "good",
    "smoking_status": "non_smoker",
    "coverage_amount_requested": 750000,
    "consent_given": true
}
```

Response:

```json
{
    "lead_id": "LIFE_001",
    "score": 92.3,
    "life_stage": "family_building",
    "mortality_risk_score": 3.2,
    "recommended_coverage": 750000,
    "recommended_policy_type": "TERM_LIFE",
    "urgency_level": "CRITICAL",
    "compliance_status": "PASSED"
}
```

Batch Processing APIs

Endpoint: `/score-leads` (All Models)
```http
POST /score-leads
Content-Type: application/json

{
    "leads": [
        {"lead_id": "001", "age": 35, ...},
        {"lead_id": "002", "age": 42, ...}
    ]
}
```

Utility APIs

Coverage Calculator
```http
GET /life-insurance-coverage-calculator?income=85000&dependents=2&mortgage=300000
```

Health Risk Assessment

```http
GET /healthcare-risk-assessment?age=42&conditions=diabetes,hypertension
```

Model Evaluation

Evaluation Framework

Each model includes comprehensive evaluation across multiple dimensions:

Accuracy Metrics

- R² Score: Coefficient of determination
- RMSE: Root Mean Square Error
- MAE: Mean Absolute Error
- Precision/Recall: For binary classification thresholds

Fairness Metrics

- Demographic Parity: Equal positive rates across groups
- Equalized Odds: Equal TPR/FPR across groups
- Individual Fairness: Similar individuals receive similar scores

Business Metrics

- Conversion Accuracy: Actual vs predicted conversions
- Revenue Impact: Lead value optimization
- Cost Efficiency: Reduced manual review requirements

Evaluation Reports

Base Insurance Evaluation

python

```
from models.insurance_lead_scoring.evaluate import InsuranceModelEvaluator

evaluator = InsuranceModelEvaluator()
report = evaluator.generate_evaluation_report('data/test_leads.csv')
```

Compliance & Ethics
Data Privacy

GDPR Compliance

- Consent Management: Explicit consent tracking
- Right to Erasure: Data deletion capabilities
- Data Minimization: Only necessary data collection
- Anonymization: PII masking in processing

CCPA Compliance

- Consumer Rights: Data access and deletion
- Opt-out Mechanisms: Marketing communication controls
- Transparency: Clear data usage disclosure

HIPAA Compliance (Healthcare Model)

- Health Data Protection: Encrypted health information
- Access Controls: Role-based data access
- Audit Trails: Complete data access logging

Fairness & Bias Prevention
Protected Classes
- Age, Gender, Race, Religion, Disability Status
- Geographic Location, Socioeconomic Status

- Health Status (where legally permissible)

Bias Detection

python

 Automated bias monitoring

```
fairness_metrics = {
    'age_bias': 0.05,  # < 0.1 threshold
    'income_bias': 0.03,
    'location_bias': 0.07,
    'overall_bias_alert': False
}
```

Mitigation Strategies

- Pre-processing: Bias-aware feature selection

- In-processing: Fairness constraints during training

- Post-processing: Score adjustment for fairness

- Monitoring: Continuous bias detection

Regulatory Compliance

Insurance Regulations

- State Insurance Codes: Compliance with state requirements

- NAIC Guidelines: National Association of Insurance Commissioners

- Fair Credit Reporting Act: Credit data usage compliance

- Equal Credit Opportunity Act: Non-discriminatory practices

Model Governance

- Model Documentation: Complete model lineage

- Validation Framework: Independent model validation

- Risk Management: Model risk assessment

- Audit Trail: Complete decision audit logs