Final Report – Data Storm v6.0

Preliminary Round

Team Name: *NaN*

Team Members:

Sachintha Lakruwan

Sathsarani Amarasinghe

Methmi Rathnayaka

---

1. Overview

Insurance companies rely heavily on agent performance to drive customer acquisition and policy sales. However, identifying which new agents are likely to succeed, and how to support existing agents to perform better, remains a challenging task.

In this competition, we were provided with a rich, time-series dataset capturing monthly activity of insurance agents, including proposals, quotations, customer interactions, and policy sales. Our objective was divided into three main goals:

1. **Exploratory Data Analysis (EDA)**
2. **Predictive Modeling**
3. **Performance Improvement**

By combining insights, models, and tracking systems, we aim to build a scalable, data-driven framework to support agent success.

2. Our Approach

Our approach combined data analysis, predictive modeling, and performance improvement into a complete solution. We began with exploratory data analysis to understand agent behavior and identify key patterns.

A classification model was then developed to predict early performance risks in new agents. For existing agents, we created agent-level summaries with engineered features like performance trend and anomaly count, applied clustering to categorize them, and designed targeted intervention strategies.

Finally, we developed a simple dashboard to visualize predictions, performance categories, and agent progress, making the system interactive and easy to interpret.

3. Exploratory Data Analysis

The Exploratory Data Analysis phase involved examining the dataset to identify underlying patterns, relationships, and potential anomalies. In this step, we performed a thorough review of the available features, visualized distributions, and assessed correlations among variables.

All steps, insights, and visualizations from the exploratory data analysis phase are thoroughly documented in the notebook: `data-storm-v6-0-EDAipynb`.

4. Predictive Modeling (Part 1)

We developed and trained a **regression-based machine learning model** to predict which insurance agents will sell policies in the upcoming month. The model uses historical agent performance data to classify agents into two categories: those who will sell at least one policy in the following month (class 1) and those who won't (class 0).

- **Model Used:** Support Vector Machine
- **Target Variable:** next_month_policy_count
- **Evaluation Metrics:** MAE, RMSE, R^2

The full implementation for this section is available in the submitted notebook: `data-storm-v6-0-part-1.ipynb`.

Agent Clustering

Before the main prediction pipeline, agents are grouped into performance clusters:

- Calculate summary statistics for each agent (means, sums of key metrics)
- Extract performance trends by fitting a linear regression to each agent's sales over time
- Identify anomalies in agent performance using standard deviation thresholds
- Apply K-Means clustering (K=3) to categorize agents into performance segments
- Cluster assignments are added as a feature to enhance prediction quality

Date Preprocessing

- Convert date columns (agent_join_month, first_policy_sold_month, year_month) to datetime format.
- Extract year and month components from dates for temporal analysis.

Target Engineering

- Created the target variable, **next_month_policy_count**, by linking each agent's performance in the current month to their sales in the following month.
- This transformed our task into a supervised learning problem; allowing the model to learn from past behavior to predict future outcomes.
- For each historical month, the model identifies whether the agent sold any policies in the subsequent month. The target is binary: 1 if the agent sold at least one policy in the following month, 0 otherwise.

Feature Engineering

Apply logarithmic transformations to handle skewed numerical features:

log_net_income = log(net_income + 1)
log_number_of_cash_payment_policies = log(number_of_cash_payment_policies + 1)

Selected features include:

- Temporal features (year, month)
- Activity metrics (unique proposals, quotations, customers)
- Recent activity (7-day and 21-day metrics)
- Current performance indicators (new policy count)
- Financial metrics (transformed net income)
- Performance cluster assignments from the clustering algorithm

Model Training and Prediction

- Use **Support Vector Machine (SVC)** with probability estimation
- Train on all historical data except the current month
- Generate probability scores for the current month's agents

Implementation Details

- The data is split temporally, using all historical months for training
- The current month (most recent in the dataset) is used for prediction
- A threshold mechanism ensures the distribution of predictions matches business expectations
- Agent performance clusters are integrated to improve prediction accuracy
- The final output is a binary prediction for each agent in the current month

Model Applications

This prediction model enables:

- Resource allocation optimization
- Targeted intervention for agents at risk of not selling
- Performance forecasting for business planning

Intervention Logic

- **Defining At-Risk Agents**

Agents were classified as **"at-risk"** if their predicted next_month_policy_count was zero or very close to zero. These agents are likely to become inactive and require timely action.

- **Flagging At-Risk Agents**

After running predictions on current data, we filtered and flagged low-performing agents based on the predicted values. This provided a prioritized list for intervention.

Top Factors Affecting Early Performance

The following features were identified as key indicators influencing the early performance of insurance agents:

- **Unique Customers Last 7 Days**: The number of distinct customers an agent interacted with in the past week.
- **New Policy Count**: The total number of policies the agent has sold in the current month.
- **Number of Policy Holders**: The total number of active policyholders under the agent's portfolio.
- **Unique Quotations Last 7 Days**: The number of unique quotes generated by the agent in the past week.
- **ANBP Value**: The Average Net Premium Base, representing the agent's sales and policy volume over time.
- **Unique Customers Last 15 Days**: The number of distinct customers an agent interacted with in the past two weeks.

Understanding these factors allows targeted interventions based on specific risk indicators rather than generic assumptions.

Personalized Action Plan Recommendation System for At-Risk Agents

In addition to predicting agents at risk of becoming One Month NILL, we developed a personalized recommendation system to support their improvement. Initially designed as a **rule-based suggestion engine** inspired by the behavior patterns of high-performing agents, the system now integrates **OpenAI's GPT-4 API** to generate dynamic, SMART (Specific, Measurable, Achievable, Relevant, Time-bound) action plans.

How It Works:

- The agent's latest statistics (e.g., proposals, quotations, customers, ANBP) are compared against the average values of top-performing agents.
- A carefully designed prompt sends this data to GPT-4.
- The model returns **three tailored action items**, categorized into:
    1. **Training**
    2. **Mentoring**
    3. **Goal Setting / Motivation**

Example Behavioral Patterns Used:

- **Low engagement** → Assign to focused training sessions
- **Fluctuating activity** → Pair with a mentor for consistency
- **New agents** → Provide structured onboarding and allow shadowing of top performers

Example Rules Translated into Prompts:

- *If unique_proposals < 15* → Recommend outreach target setting
- *If net_income is high but new policies = 0* → Suggest funnel review and follow-ups

5. Performance Improvement (Part 2)

To enhance overall productivity and guide managerial interventions, we developed a clustering-based system to categorize agents by performance levels. Using historical metrics such as income, sales consistency, and anomaly frequency, agents were grouped into **High**, **Medium**, and **Low** performers. This segmentation allows targeted strategy design for each group.

- **Technique Used:** K-Means Clustering
- **Features Used:** Aggregated performance metrics (e.g., ANBP, new policy count, performance slope)
- **Output Labels:** High, Medium, Low

This classification enables strategic decision-making. It also lays the groundwork for ongoing tracking and evaluation of intervention success.

The full implementation for this section is available in the submitted notebook:
`data-storm-v6-0-part-2.ipynb`.

Data Preparation

- Cleaned and validated monthly agent records.
- Converted date fields and ensured data consistency across months.
- No null values or duplicate entries were present.
- Outliers were retained as they reflect natural behavior; tracked via an engineered anomaly_count feature.

Feature Engineering

- **performance_slope**: Captures the linear trend of monthly policy sales for each agent.
- **anomaly_count**: Represents the number of months with unusually low or high performance, helping flag instability.

Aggregation & Feature Selection

- Aggregated monthly records to generate a **single row per agent** with meaningful summary statistics.
- Used a correlation heatmap to identify and eliminate redundant fields.
- Retained only the most informative features for clustering and classification.

Clustering & Categorization

- Applied **KMeans clustering** on scaled features to group agents by performance.
- Interpreted the resulting clusters and mapped them to:
  - **High Performers**: Consistently strong metrics with high income.
  - **Medium Performers**: Decent overall metrics but some inconsistency.
  - **Low Performers**: Weak metrics and a negative performance trend over time.

Intervention strategy

After categorizing agents into High, Medium, and Low performers through clustering, we designed thoughtful, targeted strategies to improve outcomes within each group. These strategies focus on personalized development, performance stability, and long-term growth.

i. High Performers (Consistent, High Income)

Objective: Retain top talent and maintain high performance.

- **Recognition and Incentives**: Introduce reward programs (e.g., bonuses, public recognition) to sustain motivation.
- **Leadership Opportunities**: Offer mentorship roles or leadership in internal training to promote career growth.
- **Skill Sharpening**: Provide access to advanced sales techniques, financial planning tools, and exclusive workshops.

ii. Medium Performers (Decent Metrics, Slightly Unstable)

Objective: Stabilize performance and boost consistency.

- **Performance Feedback Loops**: Introduce monthly review sessions with actionable insights and coaching.
- **Personalized Goal Setting**: Break down targets into short-term, achievable milestones based on past trends.
- **Peer Learning**: Match with high-performing mentors to share strategies and motivation.

iii. Low Performers (Low Metrics, Declining Trend)

Objective: Address root causes and support recovery or reassignment.

- **Intensive Training Programs**: Assign targeted training in product knowledge, client communication, and time management.
- **Closer Supervision**: Increase manager touchpoints (e.g., weekly check-ins) to monitor progress and provide immediate feedback.
- **Motivation Boosters**: Set smaller goals with quick wins to help agents rebuild confidence.
- **Role Re-evaluation**: If performance remains stagnant despite interventions, consider reassignment to a more suitable role.

Progress Tracking

i. Visual Tracker: Per Agent Plot

The **Visual Progress Tracker** generates individual line plots for each agent, illustrating how their **monthly performance** (specifically, the number of new policies sold) **evolves over time** within the given dataset. These plots are saved as image files, allowing for easy review and comparison.

Currently, the tracker visualizes the agents' progress during the available historical period. However, the same code is **fully reusable** for future datasets — enabling ongoing performance monitoring in the months to come. As new data is collected, this tool can help track whether interventions or training strategies are leading to real improvements in individual agent performance over time.

This tracker thus serves both as a **diagnostic tool** and a **long-term monitoring solution** for agent growth.

ii. Performance Slope Tracker

The **performance slope** calculates the trend of each agent's monthly policy sales using linear regression. A positive slope indicates improvement over time, while a negative slope reflects declining performance.

slope > 0 → improving agent
slope < 0 → declining agent
slope ≈ 0 → stable agent

This score provides a quick, numeric representation of the agent's trajectory, making it easier to assess whether their performance is stabilizing, rising, or falling over time.

The Performance Slope is a versatile metric that can be applied to future datasets, helping monitor ongoing improvements and identify agents who may require further intervention.

iii. Categorical Progress Summary

The **Categorical Progress Summary** provides a simplified view of each agent's performance trajectory by categorizing their performance slope into 05 trend categories: **Strongly Improving, Improving, Stable, Strongly Declining** or **Declining**.

This categorization makes it easier to track and understand individual agent progress, providing a clear snapshot of how each agent is performing relative to their historical performance. By grouping agents into these categories, it becomes more manageable to identify who needs additional support and who is excelling.

iv. Visual Tracker: Category-Level Trend Plot

The **Visual Tracker: Category-Level Trend Plot** shows how the average monthly performance of each agent group (High, Medium, Low) changes over time. This plot offers a visual representation of how the performance of each category evolves, enabling the assessment of whether targeted interventions or training programs are leading to improvements.

By comparing trends across categories, this plot helps determine the overall effectiveness of strategies designed for specific performance levels. It provides a clear and insightful overview of group-wide performance changes, making it easier to assess the broader impact of interventions on agent performance.

7. Dashboard

To make our solution more interactive and actionable, we built a web-based dashboard using **Next.js, Tailwind CSS, and TypeScript**, integrated with our predictive model and OpenAI's API. The dashboard allows users to view agent predictions, inspect individual performance trends, and generate smart, personalized action plans for at-risk agents.

You can access the dashboard here: https://data-storm-6-0-nan.vercel.app/

A separate document outlining the full functionality and features of the dashboard is attached with this submission for reference.

8. Conclusion

Our team approached the Data Storm challenge with a comprehensive strategy that combined exploratory data analysis, predictive modeling, clustering, and progress tracking. By transforming raw performance data into actionable insights, we built a scalable framework for identifying at-risk agents, categorizing performance levels, and delivering personalized intervention plans. Grounded in real behavioral patterns, our solution empowers proactive decision-making and sets the foundation for continuous agent development and long-term performance improvement.