Data Storm - Storming Round

Team Name - OptiMinders

EDA

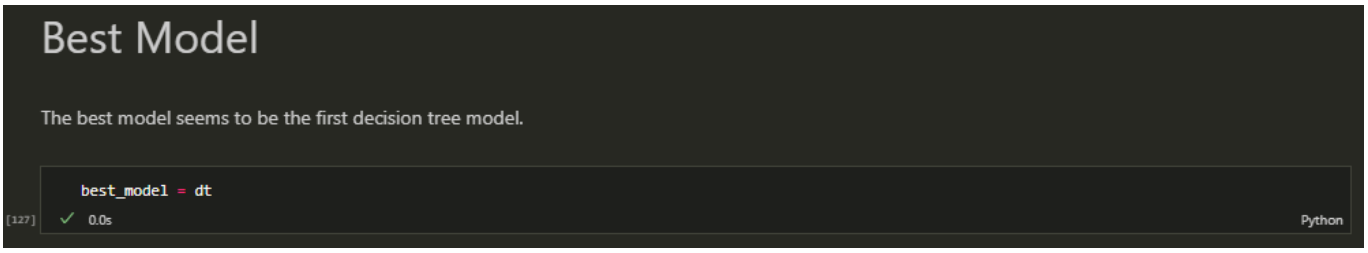
Since the 5 questions in “EDA Part” are answered in the “EDA.ipynb” Jupyter notebook, they are not discussed here.
For Question 4, since there are more than 900 agents, we decided not to visualise the evolution of agent trajectories directly within the Jupyter notebook. Instead, they are saved in a separate folder named “trajectories” in the SVG format when the notebook is run.
To generate the SVG files, the last code block in the section “Individual Agent Trajectories Over Time” must be executed. It is shown below. For its correct execution, there must be a folder named “trajectories” in the same folder as the “EDA.ipynb” Jupyter notebook.

```
# For each agent, visualize the new_policy_count over time
for agent in unique_agents:
    agent_data = train[train['agent_code'] == agent]

    plt.figure(figsize=(12, 6))
    plt.plot(agent_data.index, agent_data['new_policy_count'], marker='o')
    plt.title(f'New Policy Count Over Time for Agent {agent}')
    plt.xlabel('Month')
    plt.ylabel('New Policy Count')
    plt.xticks(rotation=45)
    plt.grid()
    plt.tight_layout()

    plt.savefig(f'trajectories/{agent}.svg', format='svg')
    plt.close()
```

Part 1

1. **Forecasting whether an agent is going to be NILL or not**

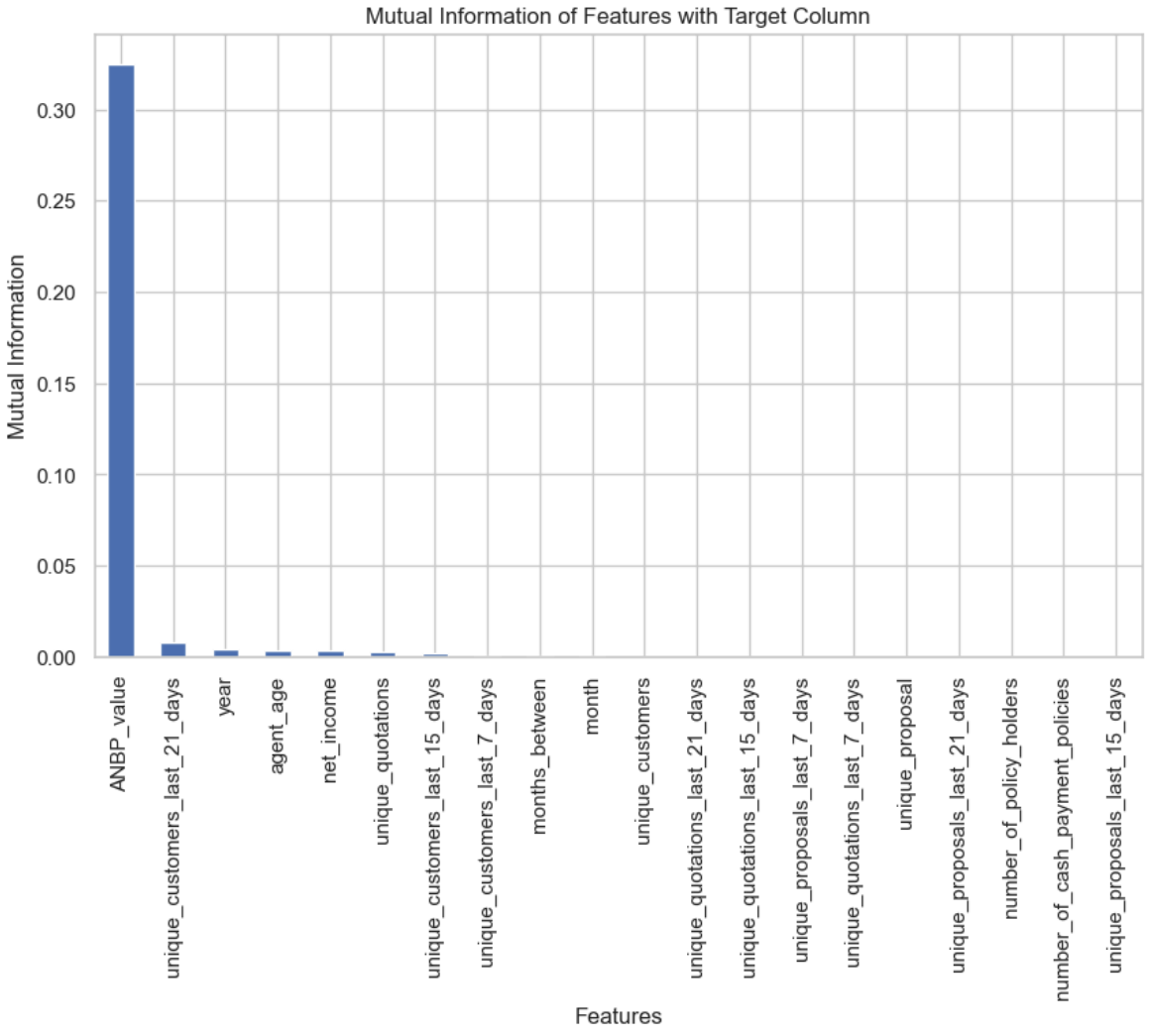
   The model to forecast whether an agent will become One NILL can be found in the “Part 1.ipynb” Jupyter notebook. It is a classification model based on decision trees. The model has been called separately at the end of the Jupyter notebook.



2. **Identifying key factors for agent performance**

   As we have shown in the “Mutual information” section in “Part 1.ipynb” Jupyter notebook, there are several factors affecting early performance.

ANBP_value	0.325016
unique_customers_last_21_days	0.007889
year	0.004619
agent_age	0.003825
net_income	0.003478
unique_quotations	0.002655
unique_customers_last_15_days	0.002134
unique_customers_last_7_days	0.000731
months_between	0.000619
month	0.000438
unique_customers	0.000231



The top features affecting early performance are given below.

- **ANBP_value** - Annualized New Business Premium value per agent
- **unique_customers_last_21_days -** Number of unique customers contacted in the last 15 to 21 days
- **year -** The year of the month for which we are predicting whether the agent is going to be NILL or not

- **agent_age -** Age of the agent
- **net_income -** Net income earned by the agent
- **unique_quotations -** Total number of unique quotations in the recorded month
- **unique_customers_last_15_days -** Number of unique customers contacted in the last 7 to 15 days

3. **SMART action plans for at-risk agents**

   As we have now identified the key factors affecting the agent performance, we can suggest several SMART action plans according to those key factors.

   - Agents with low ANBP value can attend a weekly session with a senior agent. In the sessions, they can focus on improving the explanation of insurance plans and sales techniques. The aim is to increase ANBP by 20% in one month. Real examples from top-performing agents can be used during training. Progress can be checked at the end of each week.
   - Agents who contact only a few new customers should aim to speak to at least 5 new people each day. They can use lead lists and call scripts to make this easier. The goal is to reach 100 unique customer contacts in 3 weeks. Weekly progress checks can help keep them on track.
   - Agents earning low income may need help to identify what is blocking their progress. A performance coach can guide them in changing their sales approach. The aim can be to raise income by 25% over one month. Small weekly goals and regular income checks can support this plan.
   - Younger agents or those new to the job can learn faster by having a mentor. Following an experienced agent for 3 weeks can help build skills. This gives them a chance to watch how successful agents talk to customers and close sales. A review after 3 weeks can show how much they've learned.

Part 2

1. **Method to classify current agent performance**

   After considering several methods, we decided to use the following method to calculate a performance score. First we get the change rate of new_policy_count (CN) and change rate of unique_customers (CU).

$$change\ rate = (value\ in\ current\ month - value\ in\ previous\ month) / value\ in\ previous\ month$$

- If both CN and CU are positive, the agent is a high performing agent.
- If only one of CN or CU is positive, the agent is a medium performing agent..
- Otherwise, the agent is a low performing agent.

Results after classification can be found in the "agent_performance.csv" file.

2. **Custom Interventions**

   For high performing agents
   - Celebrating performance through incentives and leaderboard features
   - Pairing them with medium performing or new agents as mentors

   For medium performing agents
   - Having group sessions with high performing agents to share techniques
   - Analysing whether lead quality is affecting performance
   - Running short-term performance challenges to boost motivation

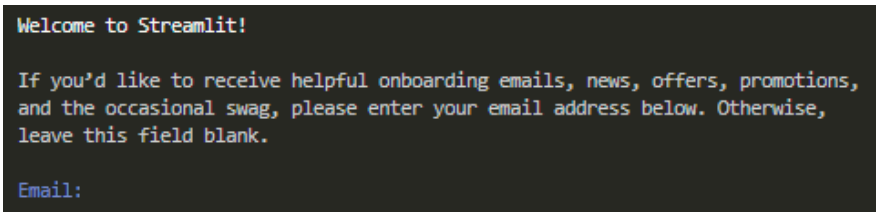
   For low performing agents
   - Giving personalised help
   - Providing structured call scripts and pitch frameworks
   - Reviewing results closely and adjust interventions monthly

3. **Progress Tracker**

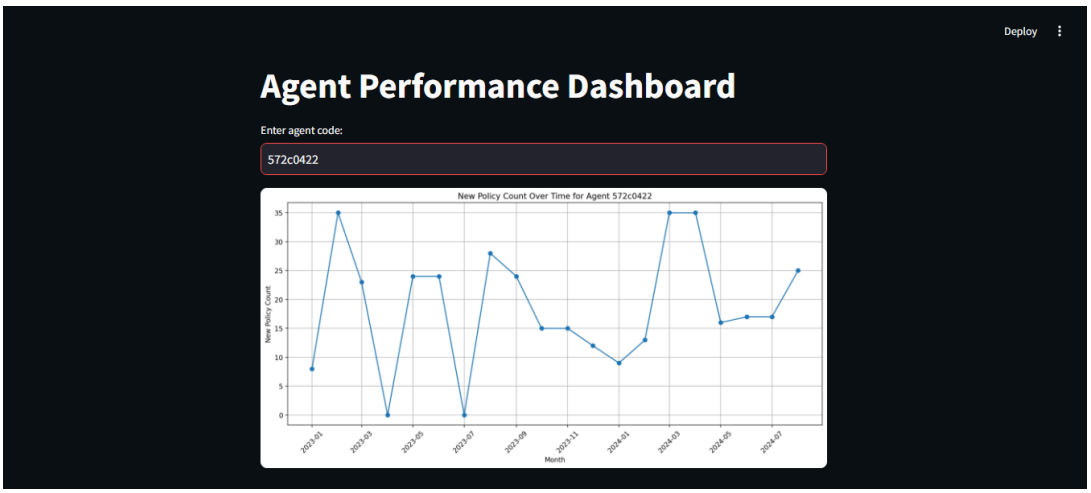
   Instructions for running the progress tracker are given in the next section

Instructions for Running the Progress Tracker

1. Install the necessary packages using the command
   pip install pandas matplotlib streamlit
2. Place the “dashboard.py” file in the same folder as “train_storming_round.csv” file.
3. In the command prompt (after going to the folder containing the “dashboard.py” file), run
   streamlit run dashboard.py
4. If an email address is asked, just press the Enter key.

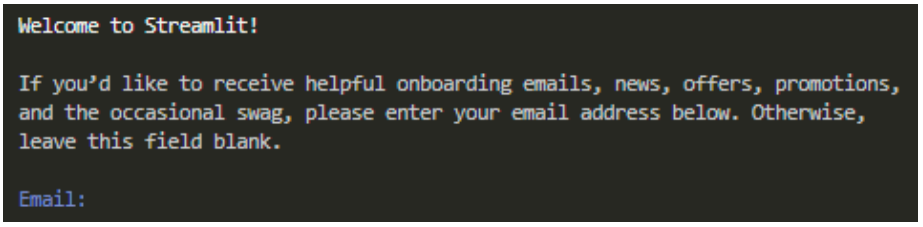


5. In the search bar in the following screen, type a valid agent code and press the Enter key.



Bonus: Dashboard for Classification and Recommendation

1. Install the necessary packages using the command
   pip install pandas streamlit
2. Place the “recommendor.py” file in the same folder as “agent_performance.csv” file.
3. In the command prompt (after going to the folder containing the “dashboard.py” file), run
   streamlit run recommendor.py
4. If an email address is asked, just press the Enter key.



5. In the search bar in the following screen, type a valid agent code and press the Enter key.