AI-Powered Intelligent Insurance Risk Assessment and Customer Insights System

AI & Machine Learning

- With the advancement of AI and machine learning, automated risk assessment and customer analysis can improve efficiency, accuracy, and customer experience.
- This project aims to build a comprehensive AI-powered system integrating various machine learning and deep learning techniques to optimize insurance processes.

Objectives:

- Risk Classification & Claim Prediction
- Customer Segmentation
- Fraud Detection
- Sentiment Analysis
- Policy Translation and Summarization
- AI Assistant – Chatbot

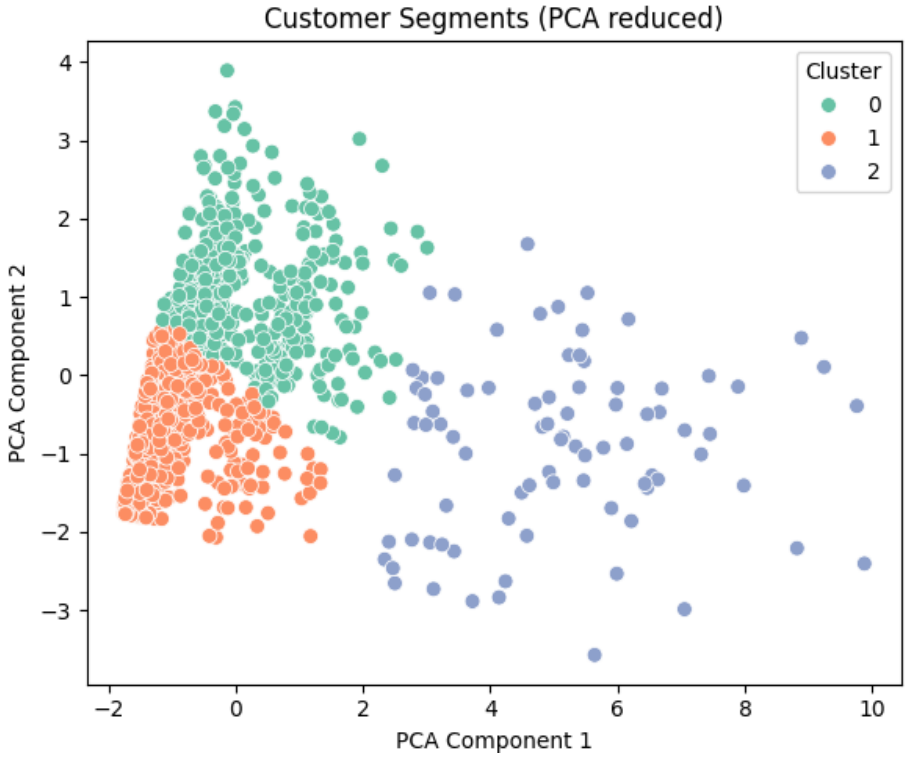
Risk Classification & Claim Prediction

The steps followed in project are as follows:

1. Load the data and perform Exploratory Data Analysis to explore the numerical and categorical columns
2. Feature engineering:
   - One-Hot-Encoding
   - Label Encoder
   - Use MinMaxScaler() for Feature Scaling
3. Fraud Detection: Anomaly Tagging - Isolation Forest
4. Correlation – Heatmap
5. Train Test Split: Train size is 70%
6. Claim Prediction: Model Training - LinearRegression(), Ridge(), XGBRegressor(), RandomForestRegressor(). Based on the Test RMSE and Test R^2 Score: XGBRegressor() model was used.
7. Saved claim prediction model and registered the model in MLFlow
8. Risk Classification: Model Training - LogisticRegression(), RandomForestClassifier(), XGBClassifier(). Based on the classification report, XGBClassifier() was used.
9. ROC_AUC Curve, Confusion Matrix were plotted for classifcation.
10. Feature importance was plotted
11. Saved risk prediction model and registered the model in MLFlow

Customer Segmentation

- Load the data and perform Exploratory Data Analysis to explore the numerical and categorical columns
- Feature Scaling: StandardScaler()
- Applied PCA(n_components=2)
- Model Training and Prediction was done using the following methods:
  - Kmeans with optimal_k = 3
  - DBScan
- Based on the Silhouette Score and scatter plot visual for customer segments, the Kmeans() model was used.
- Model, pca and scaler was saved using joblib



Fraud Detection

- Load the dataset and perform Exploratory Data Analysis to explore the numerical and categorical columns
- Feature Engineering: From the given dataset, generate 2 new feature's:
  - "Claim processing days"
  - "Claim-to-Income ratio"
- Implement Anomaly Detection using the following methods:
  - Elliptic Envelope
  - Isolation Forest
  - Local Outlier Factor
- Use One Hot Encoding for categorical columns
- Data normalization using StandardScaler().
- Apply Train test split method
- Use SMOTE technique to handle class imbalance as the number of 'not fraud' is high when compared to "Fraud" label.
- Model Training and Evaluation: RadomForestClassifier and FeedForwardNeuralNetworks(Deep Learning).
- FFNN was saved and used, as the classification report values were high when compared to RandomForestClassifier.
- SHAP to detect feature importance and MLFlow to register the model were also implemented.

Sentiment Analysis

- Load the dataset and preprocess the data using spacy
- Create “NLP” pipeline object using spacy and English Language Model.
- Use the “NLP” to Tokenize, Tag POS, NER, Lemmatize, Parse
- Implement TF-IDF with RandomForestClassifier and LogisticRegression.
- Train the Sentiment Analysis data using both models.
- Use Classification Report to check the accuracy of Sentiment Analysis, the accuracy was little high when using LogisticRegression
- Implement Hugging Face transformer : The model used was "bert-base-uncased“.
- Use AutoTokenizer on the dataset, and load the model using AutoModelForSequenceClassification
- Train and evaluate the transformer model using Sentiment Analysis dataset.
- The accuracy was high when compared to statistical machine learning models.
- Hence the transformer model was saved and used for the Sentiment Analysis prediction.

Policy Translation and Summarization

- The dataset was created using mBart model, it was used to translate English Insurance Policies into French and Spanish languages.
- A relatively light weight language model Helsinki-NLP was trained on the dataset and the model was saved to predict the language translations.
- The dataset was also used to train the summarizer model facebook/bart-large-cnn.
- The facebook/bart-large-cnn summarizer model was then used to predict the summary of the English language policy from the streamlit app.

AI Assistant – Chatbot

- Load and clean policy data.
- Create embeddings and build a FAISS similarity index.
  - Loads a **lightweight transformer model** (all-MiniLM-L6-v2) to convert policy text to vectors (embeddings).
  - Creates a **FAISS index** (IndexFlatL2) for fast similarity search using L2 distance.
  - Caches the model and index for reuse.
- Accept user questions.
- Find top-matching policy chunks.
  - Converts the **user's question** into an embedding.
  - Uses FAISS to find the top-k most similar chunks from the policy.
  - Returns the top matches as context.
- Use BART to generate a natural-language answer.
  - Loads the **BART model and tokenizer**.
  - This version (facebook/bart-large-cnn) is trained for **summarization**, and works well for Q&A too.

Thank you