1 Approach

Clean the data, convert categorical values, try to get the best model using AutoML by Flaml and LazyPredict. And use the model with the best r2_score.

2. Data Pre-processing/Feature Engineering

Converting categorical values to numerical ones to get better predictions using label encoding and one hot coding.

Tried Log transform for cltv, didn't work out. Reduced the r2_score, so didn't use it.

Also, tried to remove outliers, did nothing to r2_score. There were no significant outliers

Tried various feature engineering claim_amount/income. claim_amount/area, claim_amount/vintage, 1/claim_amount, (policyA, policy, policy) * type_of_policy.

Initially, I tried predicting without any feature engineering and used the values with the most importance to form up new values. Tried combinations of two or more alterations in the feature. Finally, Claim_amount/income gave the highest r2_score, So, I used that only.

Also, used sweetviz to get various insights of the data.

3. Final Model

Final model is lgbm regression by flaml with 5 layers

I tried xgboost, lgbm using flaml

Also, used lazypredict module. With 42 models, Gradient Boosting Regressor gave the highest r2_score.

Automl lgbm r2_score is higher than that of lazypredict. So, I went with Automl lgbm with r2_score around 0.158863