upGrad

LEAD SCORING CAPSTONE PROJECT

Predictive Model to Prioritize High-Potential Leads

Navneet Kumar

BUSINESS PROBLEM

- The company aimed to improve lead conversion rates by identifying leads more likely to convert.
- Current approach treated all leads equally, causing inefficient use of sales resources.
- Objective: Build a predictive model to estimate conversion probabilities and prioritize leads accordingly.

DATA OVERVIEW

- Dataset included features such as:
  - Total Time Spent on Website
  - Total Visits
  - Lead Origin and Source
  - Engagement channels like Olark Chat

- Some optional dropdown fields had a default placeholder "Select" indicating no choice.

- Dataset split into approximately 70% training and 30% testing sets for modeling and evaluation.

DATA PREPROCESSING

- Handled Missing Values
  - Checked with df.isnull().sum()
  - Treated “Select” in dropdowns as missing
- Removed Irrelevant/Low-variance Columns
- Encoded Categorical Variables (One-Hot Encoding)
- Scaled Numerical Features (Standardization)
- Split Data into Training (~70%) & Test (~30%) sets
- Added Constant for statsmodels logistic regression

MODEL BUILDING

- Logistic Regression was chosen for:
  - Interpretability — stakeholders can understand model coefficients.
  - Suitability for binary classification tasks.
  - Generating probabilistic scores to allow threshold tuning.

CUTOFF SELECTION USING TRAINING DATA

- **Cutoff threshold determines the probability above which a lead is classified as a converter.**
- **Default is 0.5, but our analysis showed an optimal cutoff around 0.42 balancing key metrics.**

Metric	Value
Accuracy	0.624
Sensitivity	0.608
Specificity	0.638

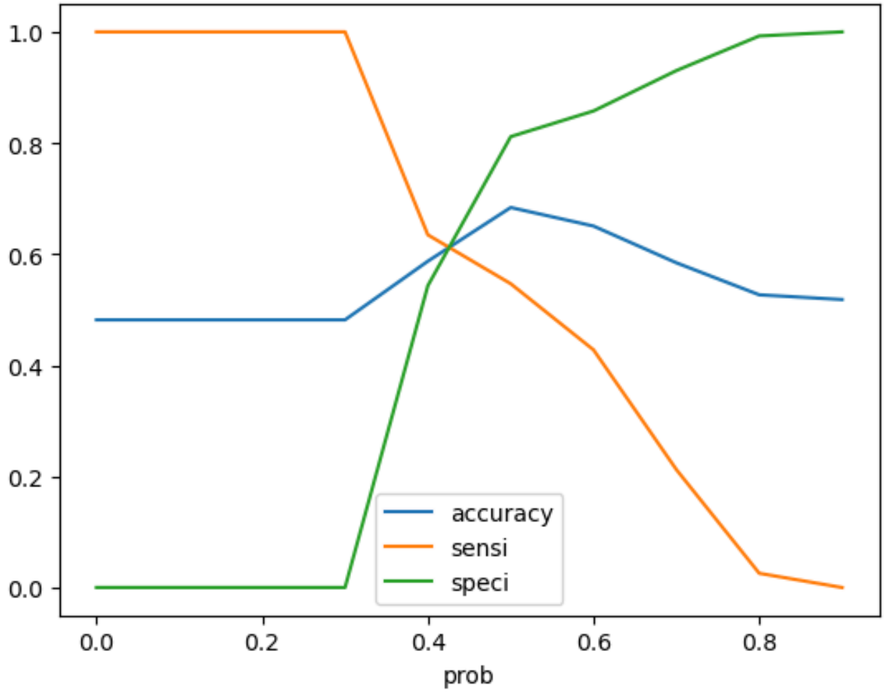
- **Confusion matrix at 0.42 cutoff:**

	Predicted No	Predicted Yes
Actual No	1476	836
Actual Yes	842	1307

ROC CURVE ANALYSIS

- **Purpose: Shows trade-off between Sensitivity (TPR) & 1 – Specificity (FPR) at various thresholds.**
- **AUC: Indicates good model discrimination between converters & non-converters.**
- **Observation: Optimal cutoff ≈ 0.42 gives best balance of sensitivity (0.608) & specificity (0.638).**
- **Interpretation: Model can rank leads effectively; moving from default 0.5 improves business outcomes.**

```
cutoff_df.plot.line(x='prob', y=['accuracy','sensi','speci'])
plt.show()
```



TESTING SET EVALUATION AND ROC CURVE

- **Applied a cutoff of 0.45 on test data.**
- **Metrics matched training performance, with:**

Metric	Value
Accuracy	0.624
Sensitivity	0.618
Specificity	0.63

- **Confusion matrix on test data:**

	Predicted No	Predicted Yes
Actual No	627	369
Actual Yes	350	566

- **Interpretation: Consistency of training and testing ROC curves confirms model generalization.**

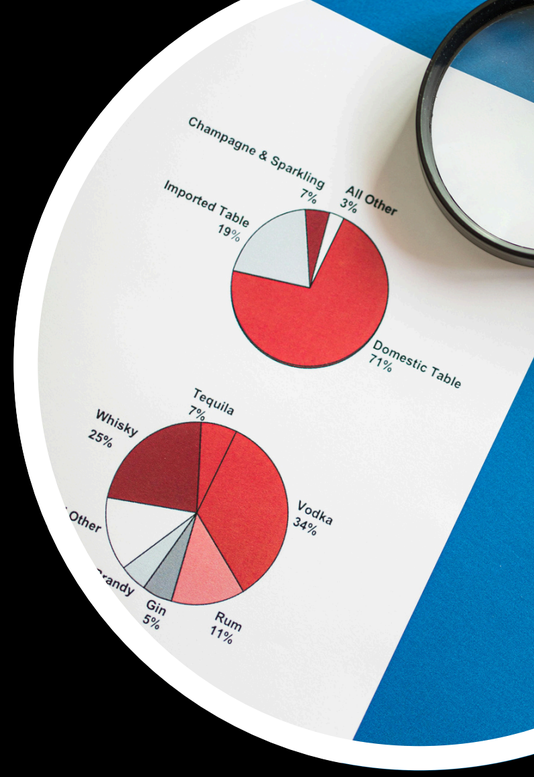
PRECISION-RECALL OPTIMIZATION AND IMPACT

- **Precision-recall tradeoff is crucial to reduce missed conversions without flooding sales with false positives.**
- **Initial precision ≈ 0.729 and recall ≈ 0.547.**
- **Adjusted cutoff (~0.40) improved recall to 0.591 with minimal precision loss (0.660).**

- **Confusion matrix after tuning:**

	Predicted No	Predicted Yes
Actual No	1657	655
Actual Yes	880	1269

KEY INSIGHTS & INTERPRETATIONS



- **Total Time Spent on Website is a strong predictor — more engaged leads convert more.**
- **The logistic regression model is interpretable and performs consistently across datasets.**
- **Cutoff tuning balances false positives and false negatives according to business priorities.**

RECOMMENDATIONS

- Integrate the model into CRM systems for real-time lead scoring.
- Prioritize leads by probability buckets for efficient sales focus.
- Monitor model performance regularly and adjust cutoffs as needed.
- Enrich future models with behavioral and campaign data.

CONCLUSION

- The logistic regression model with cutoff tuning (0.42–0.45) provides effective lead prioritization.
- Enables better alignment of sales efforts with conversion likelihood.
- Supports data-driven decision-making for improved sales efficiency.


THANK YOU