Gimme Some Credit!

Armando Rangel
Mariam Dayoub
Sander Iwase
Solano Jacon

04/12/2020

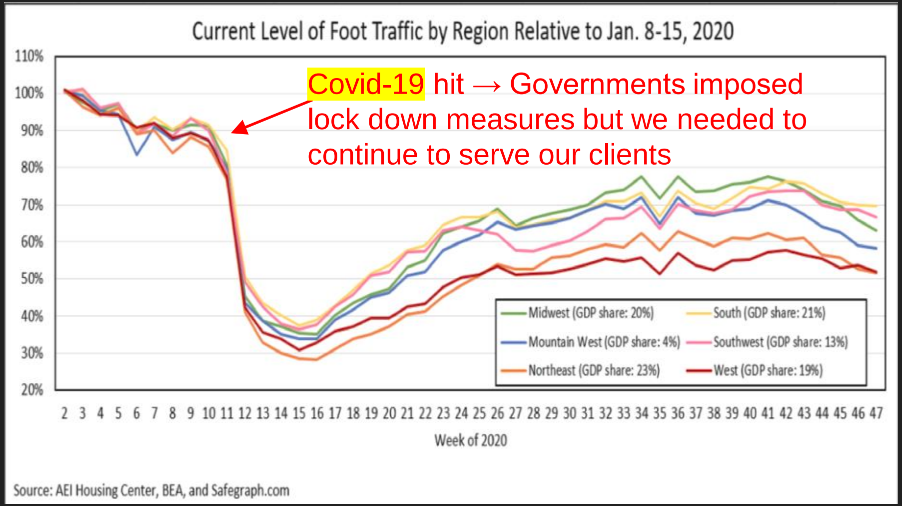
Roadmap

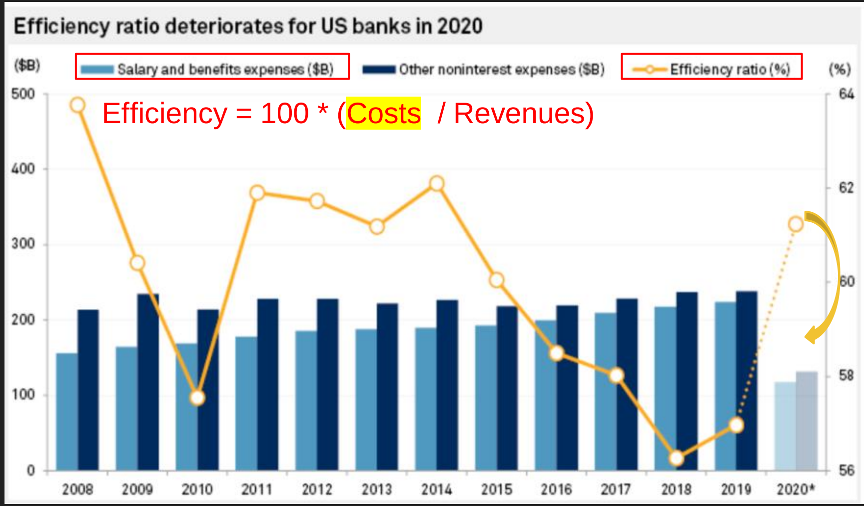
1. Who We Are & Project’s Goals
2. Dataset
3. Data Visualization
4. The Model
5. Final Product
6. Conclusions & Recommendations

1. Who We Are & Project's Goals

"Banking is essential, but banks are not"
(Bill Gates)

- We are a group of data scientists hired by a small US bank
  - Covid-19 pandemic
  - Digitalization trend
  - Efficiency (costs), productivity (workload) & client satisfaction (response time)
  - Machine learning algorithm
  - Diversify loan portfolio





2. Dataset: Original

- The dataset has information about loan requests made to the bank, available in the website https://www.kaggle.com/vineeth1999/loanapplication
- The dataset comprises 17 features and 1 target for 60,804 loan requests

```
Loan.ID
Current.Loan.Amount
Term
Credit.Score
Years.in.current.job
Home.Ownership
Annual.Income
Purpose
Monthly.Debt
Years.of.Credit.History
Months.since.last.delinquent
Number.of.Open.Accounts
Number.of.Credit.Problems
Current.Credit.Balance
Maximum.Open.Credit
Bankruptcies
Tax.Liens
Loan.Status
```

Target

2. Dataset: Data Cleaning & Removal of Outliers

- Annual Income and Credit Score: deleted rows with NA
- Current Loan Amount: deleted lines with 99999999
- Home Ownership: substituted ‘HaveMortgage’ for ‘Home Mortgage’
- Years in Current Job: in the case of NA, conservatively filled the gaps with ‘less than 1 year’
- Bankruptcies and Tax Liens: substituted NA for 0

CATEGORY	SCORE
Excellent (30% of People)	750 - 850
Good (13% of People)	700 - 749
Fair (18% of People)	650 - 699
Poor (34% of People)	550 - 649
BAD (16% of People)	350 - 549

- Credit Score: for very high values (e.g., 7,500), it was assumed that those were typos and the values were divided by 10
- Outliers removed: Annual income > $4,000,000 (1 row) and Loan vs. Income > 1 (1 row)

2. Dataset: Encoding & Feature Engineering

Feature	Encoding / Feature Engineering
Number of open accounts	Buckets: < 10, 11-20 and > 20 → Label encoder (0, 1, 2)
Number of credit problems	Buckets: 0, 1 or more than 1 → Label encoder (0, 1, 2)
Years of credit history	Buckets: (0-10], (10, 20], (20, 30] and > 30 → Label encoder (0, 1, 2, 3)
Credit score*	Buckets: below average (< 703), average (703 a 729) and good (> 729) → Label encoder (0, 1, 2)
Years in the current job	Strings (e.g., ‘1 year’) converted to int. If ‘less than 1 year’, then, 0. If, ‘10+ years’, then 10
Tax liens	If 0, then 0. If > = 1, then 1

* The average credit score of US consumers is 703. Source: https://www.cnbc.com/select/average-fico-score-hits-record-high-703/#:~:text=The%20average%20FICO%20score%20in,credit%20(670%20to%20739)

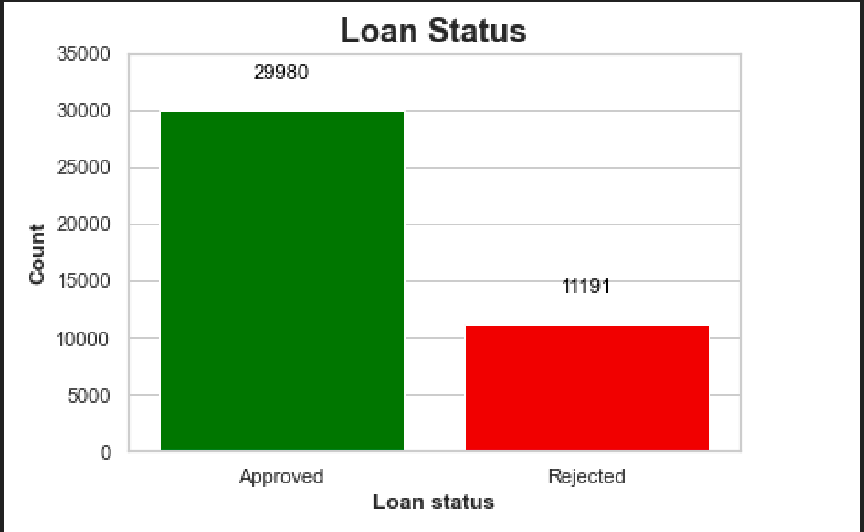
2. Dataset: Encoding & Feature Engineering (cont.)

Feature	Encoding / Feature Engineering
Purpose of the loan	Debt consolidation, home improvement, business loan, buy a car, medical bills, buy a home, and other → One Hot Encoder
Bankruptcies	If client has ever gone bankrupt, then 1. Else, 0
Any delinquency in the past 3 years?	If yes, then 1. Else, 0
Term	Short-term = 0 and long-term = 1
Loan vs. Income	Current loan amount / Annual income
Credit minus loan	Take the difference between current credit balance and current loan amount. If it is greater than 0, then 1. Else, 0
Leverage	(Monthly debt x 12) / Annual income

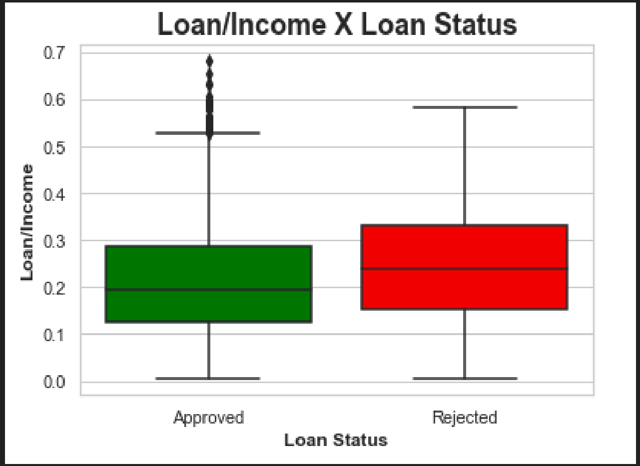
2. Dataset: Final

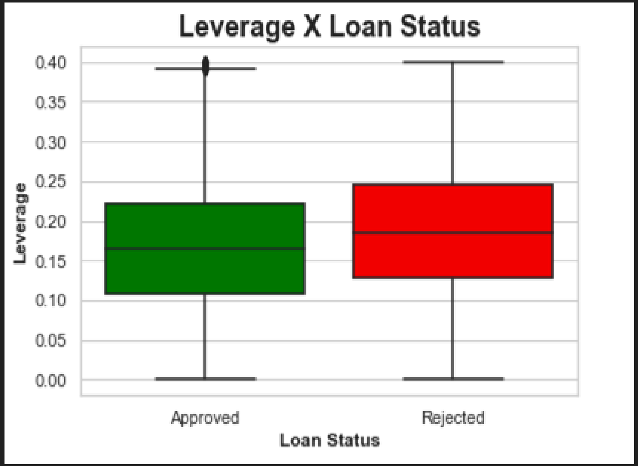
- The final dataset was comprised of 41,171 samples
- The target is unbalanced with **29,980 loans approved (1)** and **11,191 loans rejected (0)**

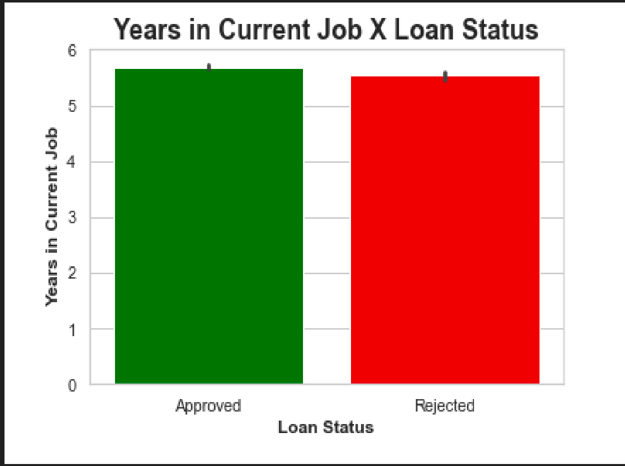
Loan.Status	Target	int64
Credit.Minus.Loan		int64
Years.current_job_enc		int64
Tax.Liens.Enc		int64
Leverage		float64
Bankruptcies.enc		int64
Years.since.last.delinquent		int64
Loan.vs.Income		float64
Term.Encoded		float64
Number.of.Open.Accounts.Labeled		int64
Number.of.Credit.Problems.Labeled		int64
Credit.Score.Labeled		int64
Year.Credit.History.Labeled		int64
H.O.Home Mortgage		float64
H.O.Own Home		float64
H.O.Rent		float64
Purp.Business Loan		float64
Purp.Buy House		float64
Purp.Buy a Car		float64
Purp.Debt Consolidation		float64
Purp.Home Improvements		float64
Purp.Medical Bills		float64
Purp.Other		float64



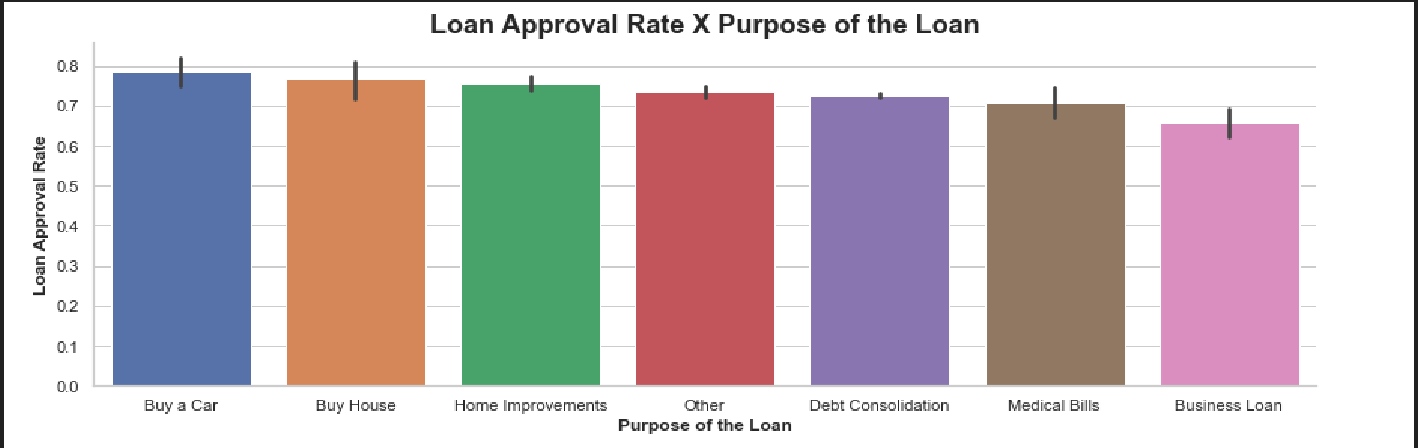
3. Data Visualization

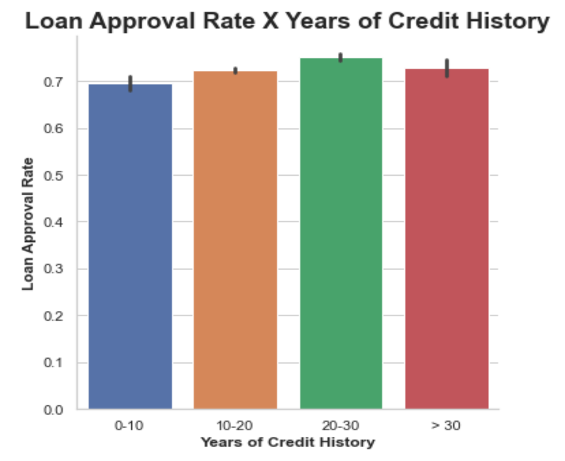


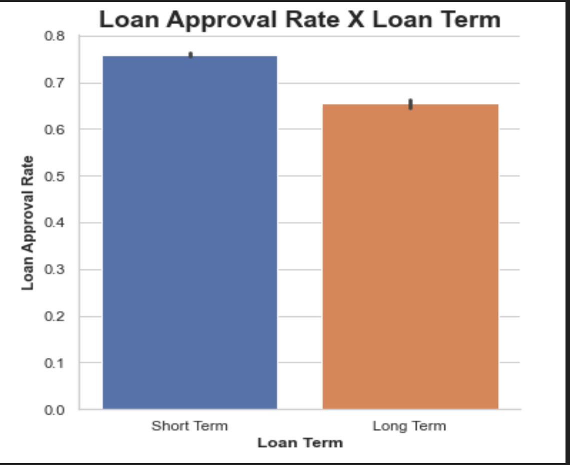


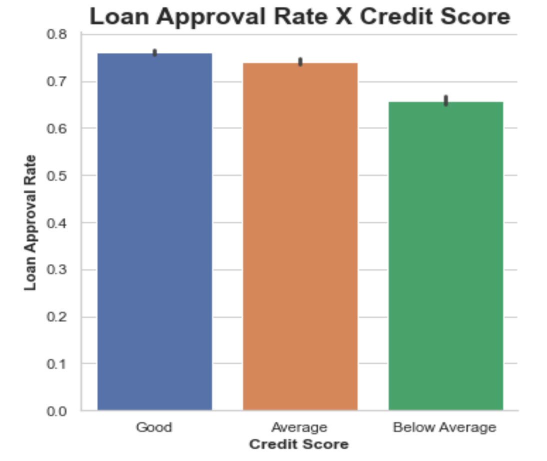


3. Data Visualization (cont.)









4. The Model: Background & Performance Metrics

- Target variable: 0 (loan rejected) or 1 (loan approved) → Classification models
- Applied SMOTE technique to the training set
- Performance metrics

Metric	Formula	Chosen because…
Precision	$\frac{TP}{TP+FP}$	We want to be right to with a high degree of confidence in our predictions in order to automate the human decision-making process
F1	$2\frac{precision \times recall}{precision + recall}$	It is a metric that is appropriate to unbalanced datasets with different costs of false negatives and false positives

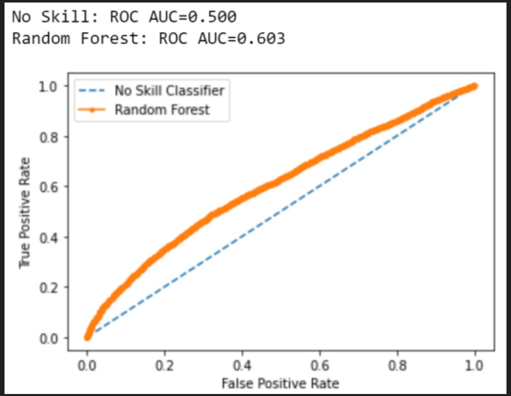
Weighted between 0 and 1

4. Models Tests and Their Performance Metrics

With feature selection and grid search for hyperparameter tuning

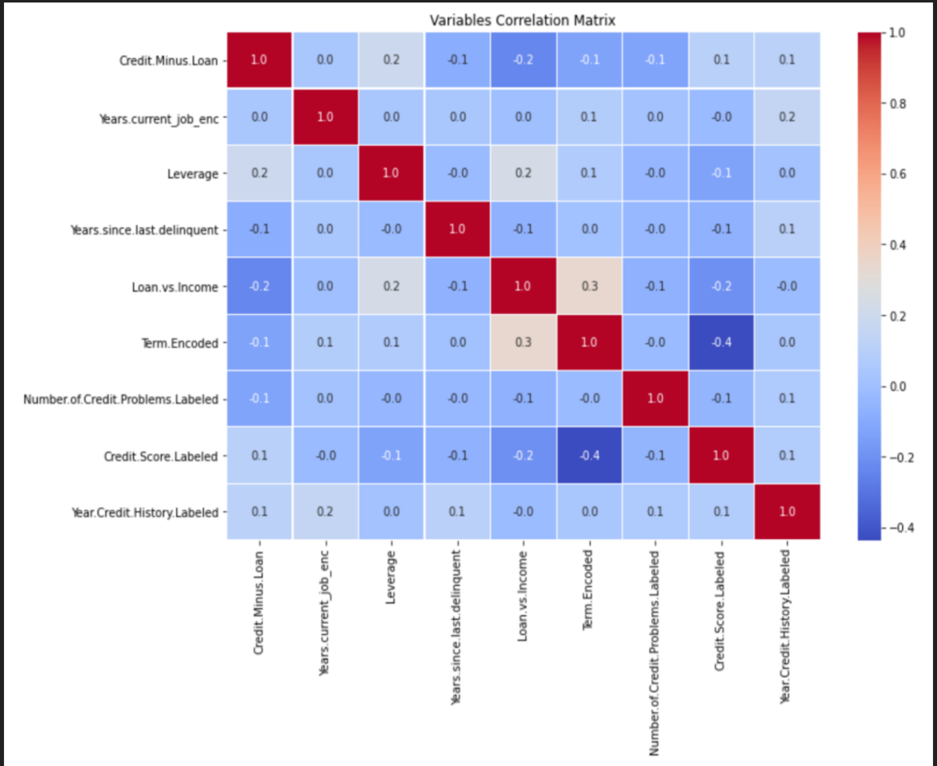
Model	Precision_weighted	F1_weighted
AdaBoost Classifier	0.64	0.61
KNN Classifier	0.66	0.35
Linear SVM	0.66	0.62
Logistic	0.67	0.62
Neural network	0.66	0.57
Random Forest Classifier	**0.65**	**0.64**
SGD Classifier	0.62	0.12
XGBoost	0.64	0.63

- Random Forest Classifier was chosen: Simple and explainable
- ROC-AUC curve: Random Forest is better than a simple random model

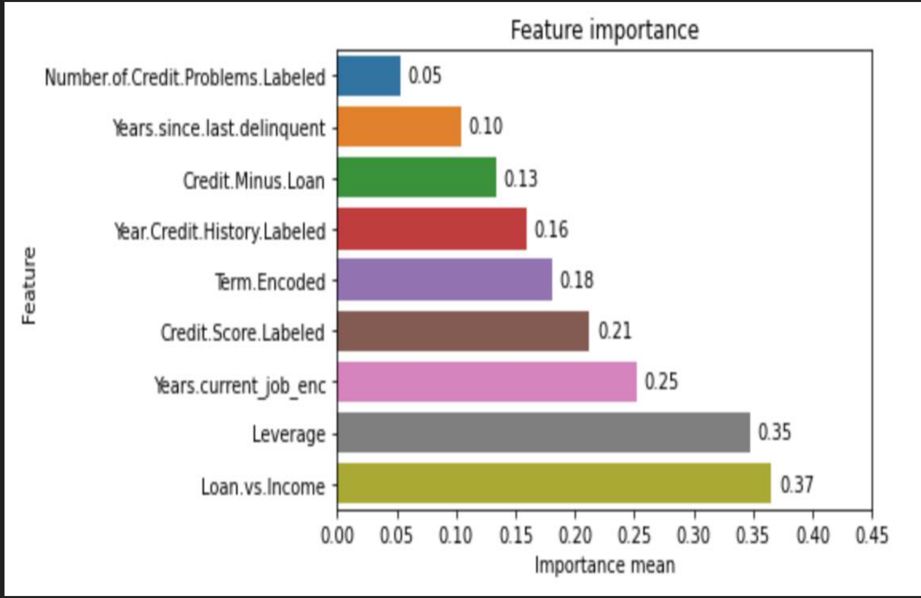


4. The Model: Correlation Matrix & Feature Selection

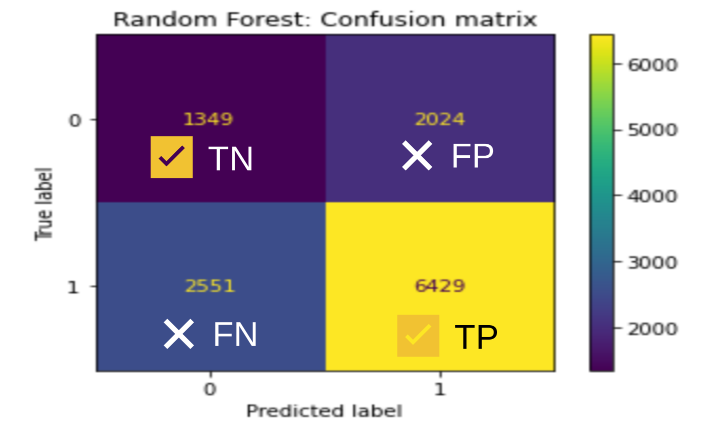
Multicollinearity is not an issue



Feature importance ranking



4. The Model: Random Forest Classifier



Precision(1): out of 100 loan requests **approved** by the model, it was actually correct about 76

Precision(0): out of 100 loan requests that were **rejected** by the model, it was correct about 35 → BIASES

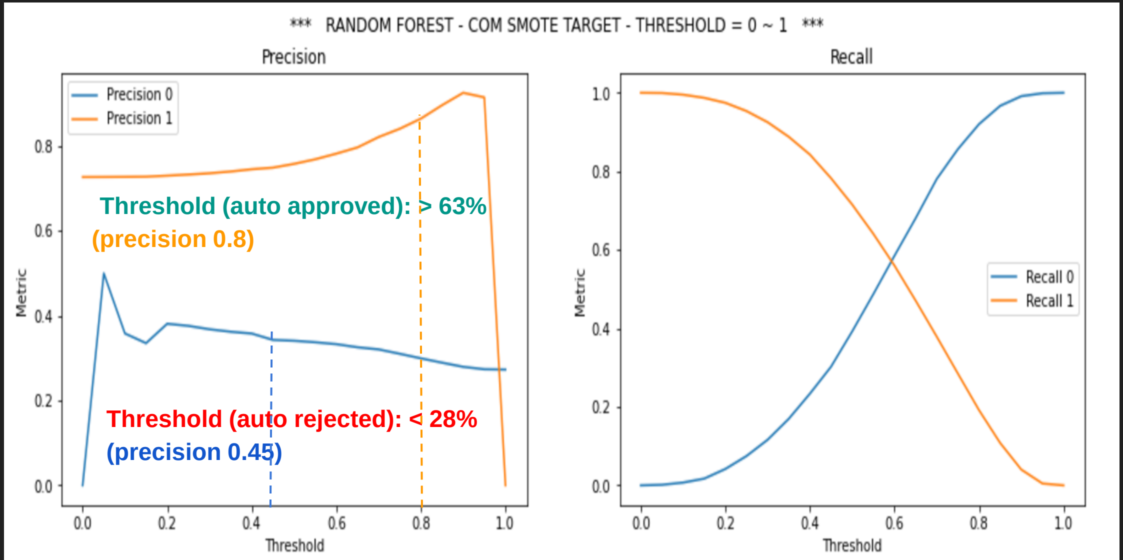
RANDOM FOREST W/ FEATURE SELECTION & GRID SEARCH - CLASSIFICATION REPORT

	precision	recall	f1-score	support
0	0.35	0.40	0.37	3373
1	0.76	0.72	0.74	8980
accuracy			0.63	12353
macro avg	0.55	0.56	0.55	12353
weighted avg	0.65	0.63	0.64	12353

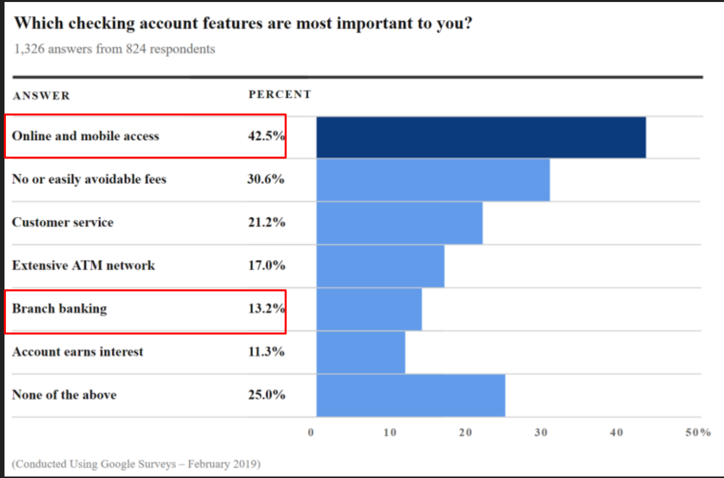
Recall(1): the model correctly identified 72% of all loans that were **approved**

Recall(0): the model correctly identified 40% of all loans that were **rejected**

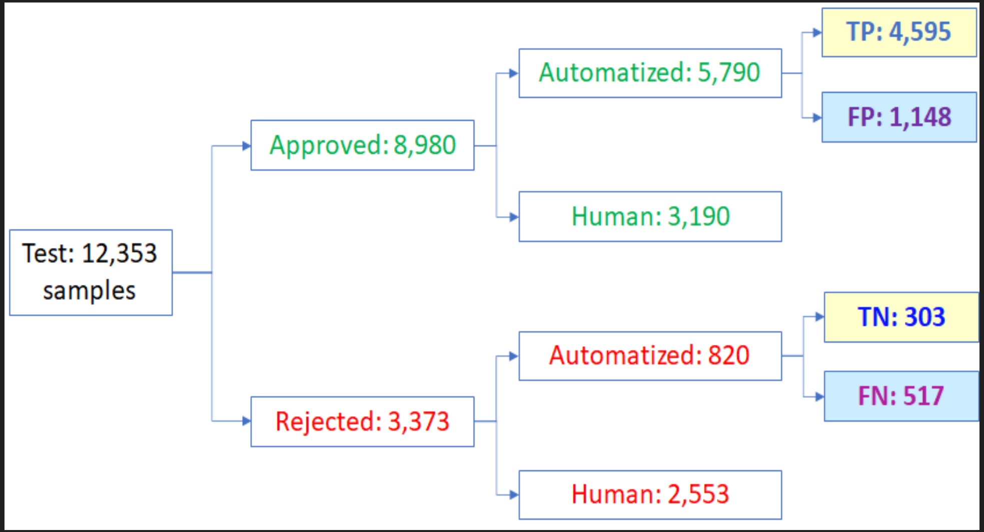
4. Looking for Thresholds for Automation



Customers value digitalization more than branch banking



4. Test: 53% of Loan Requests Were Automated



- 53% of loan requests were automated, using thresholds
- The other 47% of requests required human intervention
- 40% of loan requests were automatized and classified correctly

5. Final Product: The Website

https://gimmecredit-solano.herokuapp.com

Now, let’s do a few demos

6. Conclusions & Recommendations

- The model allowed for the automation of 53% of loan requests, correctly classifying 75% of the automated requests
- For requests in which the model has lower precision rates, its output would be used as an input for the human intervention
    - Given that human interactions involve biases, the goal in these cases is to focus on the decision-making process so that similar requests have similar decisions made by different humans
- To improve the model's precision rates, data for new loan requests will be collected and fed into the model for continuous computations
- The loan request form for new loans should be filled online in order to improve the data collection process (e.g., avoiding typos in credit score)
- We propose redesigning the loan request form to include additional information, such as the client's zip code, marital status, income of co-signers, gender, etc. in an attempt to capture some biases

Thank you very much

Give us some credit, if you liked it...