CREDIT ANALYSIS: LOAN DEFAULTS

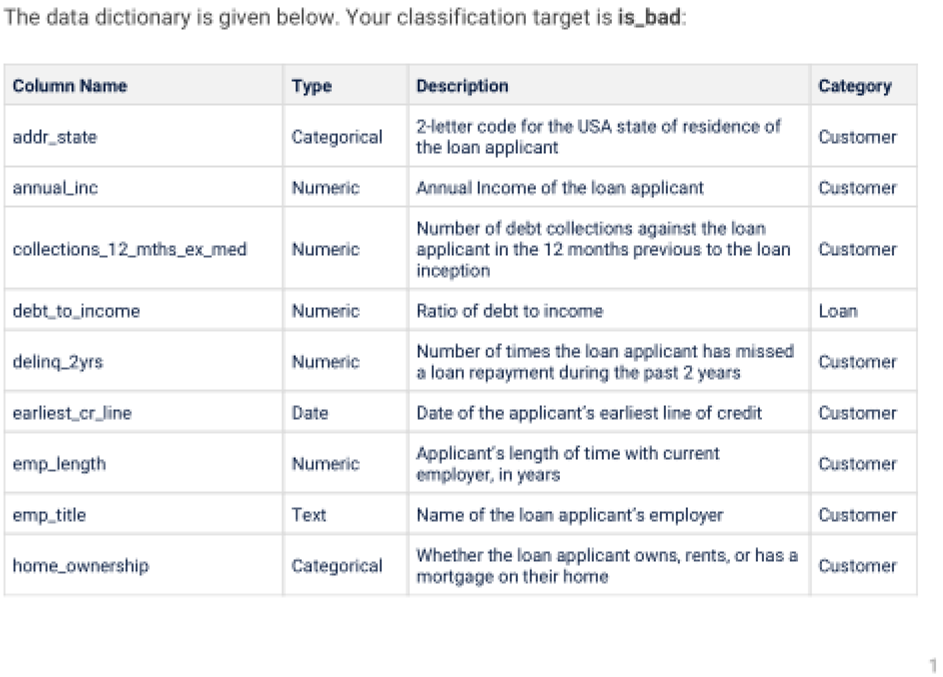
Machine Learning Engineer Take Home Test

Problem Overview

Predicting loan defaults is an extremely common use case for machine learning in banking. As a loan officer, you are responsible for determining which loans are going to be the most profitable and worthy of lending money to. Based on a loan application from a potential client, you would like to predict whether the loan will be paid back in time.

Data

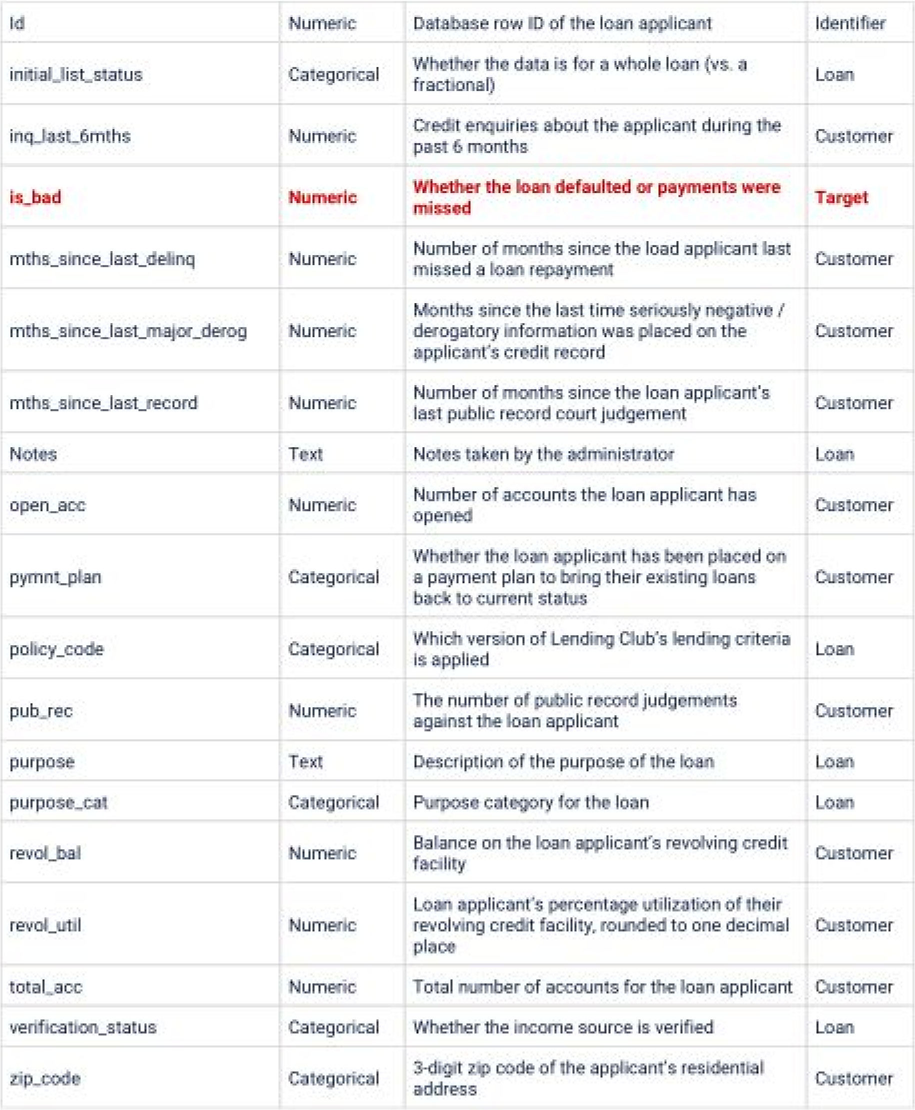
You will be working with a loan dataset from LendingClub.com (source), a US peer-to-peer lending company. Download the dataset from the following URL:
.../DR_Demo_Lending_Club.csv



The data dictionary is given below. Your classification target is **is_bad**:

Column Name	Type	Description	Category
addr_state	Categorical	2-letter code for the USA state of residence of the loan applicant	Customer
annual_inc	Numeric	Annual Income of the loan applicant	Customer
collections_12_mths_ex_med	Numeric	Number of debt collections against the loan applicant in the 12 months previous to the loan inception	Customer
debt_to_income	Numeric	Ratio of debt to income	Loan
delinq_2yrs	Numeric	Number of times the loan applicant has missed a loan repayment during the past 2 years	Customer
earliest_cr_line	Date	Date of the applicant's earliest line of credit	Customer
emp_length	Numeric	Applicant's length of time with current employer, in years	Customer
emp_title	Text	Name of the loan applicant's employer	Customer
home_ownership	Categorical	Whether the loan applicant owns, rents, or has a mortgage on their home	Customer

1



Id	Numeric	Database row ID of the loan applicant	Identifier
initial_list_status	Categorical	Whether the data is for a whole loan (vs. a fractional)	Loan
inq_last_6mths	Numeric	Credit enquiries about the applicant during the past 6 months	Customer
is_bad	**Numeric**	**Whether the loan defaulted or payments were missed**	**Target**
mths_since_last_delinq	Numeric	Number of months since the load applicant last missed a loan repayment	Customer
mths_since_last_major_derog	Numeric	Months since the last time seriously negative / derogatory information was placed on the applicant's credit record	Customer
mths_since_last_record	Numeric	Number of months since the loan applicant's last public record court judgement	Customer
Notes	Text	Notes taken by the administrator	Loan
open_acc	Numeric	Number of accounts the loan applicant has opened	Customer
pymnt_plan	Categorical	Whether the loan applicant has been placed on a payment plan to bring their existing loans back to current status	Customer
policy_code	Categorical	Which version of Lending Club's lending criteria is applied	Loan
pub_rec	Numeric	The number of public record judgements against the loan applicant	Customer
purpose	Text	Description of the purpose of the loan	Loan
purpose_cat	Categorical	Purpose category for the loan	Loan
revol_bal	Numeric	Balance on the loan applicant's revolving credit facility	Customer
revol_util	Numeric	Loan applicant's percentage utilization of their revolving credit facility, rounded to one decimal place	Customer
total_acc	Numeric	Total number of accounts for the loan applicant	Customer
verification_status	Categorical	Whether the income source is verified	Loan
zip_code	Categorical	3-digit zip code of the applicant's residential address	Customer

2
Task
1. Partition your data into a holdout set and 5 stratified CV folds.
2. Pick any two machine learning algorithms from the list below, and build a binary classification model with each of them:
○ Regularized Logistic Regression (scikit-learn)
○ Gradient Boosting Machine (scikit-learn, XGBoost or LightGBM)
○ Neural Network (Keras), with the architecture of your choice
3. Both of your models must make use of numeric, categorical, text, and date features.
4. Compute out-of-sample LogLoss and F1 scores on cross-validation and holdout.
5. Which one of your two models would you recommend to deploy? Explain your decision.
6. (Advanced, optional) Which 3 features are the most impactful for your model? Explain

your methodology.

Submission
Implement your solution as a Python script using Python 3.6 or above. Make sure the results are reproducible. Alternatively, you can use a Jupyter notebook.
...