Case Study

Bankruptcy prediction of Polish companies

Nikola Miszalska, Grzegorz Zakrzewski

June 9, 2022

Overview

1. Introduction

2. EDA

3. Preprocessing

4. Models

5. XAI

Introduction

Introduction

A short introduction to our project

The dataset is about bankruptcy prediction of Polish companies. The bankrupt companies were analyzed in the period 2000-2012, while the still operating companies were evaluated from 2007 to 2013.

Basing on the collected data five classification cases were distinguished, that depends on the forecasting period:

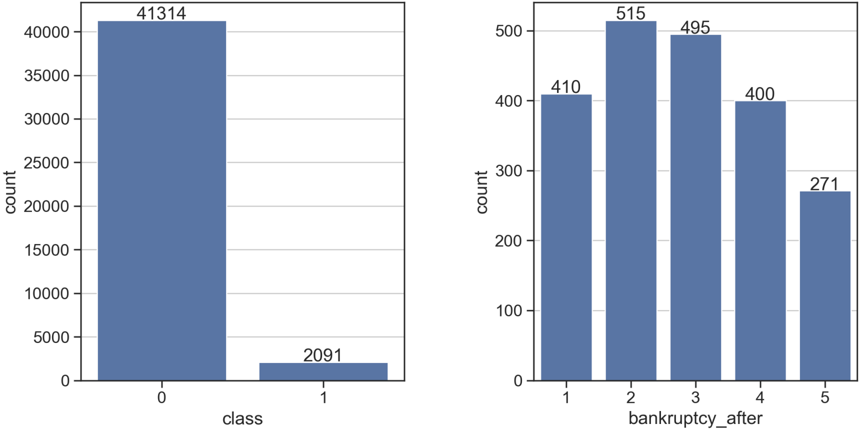
- 1st Year : 271 bankrupted companies, 6756 firms that did not bankrupt
- 2nd Year: 400 bankrupted companies, 9773 firms that did not bankrupt
- 3rd Year: 495 bankrupted companies, 10008 firms that did not bankrupt
- 4th Year: 515 bankrupted companies, 9277 firms that did not bankrupt
- 5th Year: 410 bankrupted companies, 5500 firms that did not bankrupt

EDA

EDA

Distribution of target classes

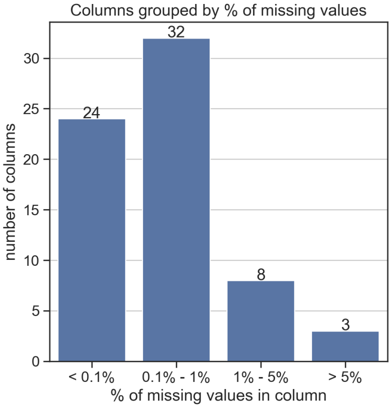
- We had to deal with strongly unbalanced classes

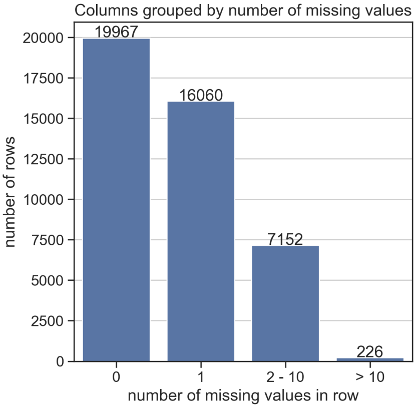


EDA

Missing values

- Half of rows contained some missing values

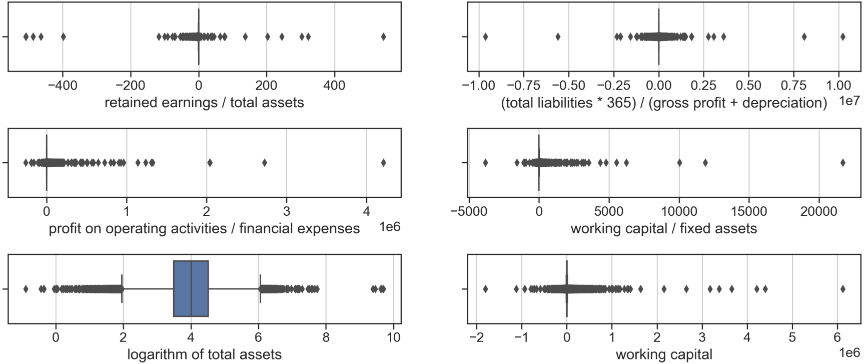

Columns grouped by % of missing values


Columns grouped by number of missing values

EDA

Outliers

- Every single feature had strongly skew distribution



Preprocessing

Preprocessing

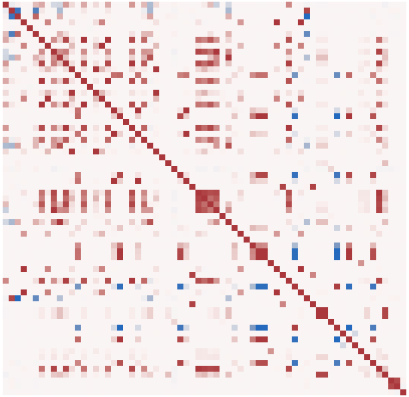
What we did?

- Drop 3 column, which had more than 5% of missing values
- Delete rows, which had more than 7 missing values (about 200 rows)
- Cut outliers to quantiles : 0.025 from left and 0.975 from right
- Impute missing values with column medians
- Standardize features by removing the mean and scaling to unit variance

Preprocessing

What we tried?

- No column was correlated with target variable
- But there were groups of strongly correlated columns
- We generated all strong correlated groups and keep only one column from group
- That didn't have positive impact on models



Models

Models

Overview

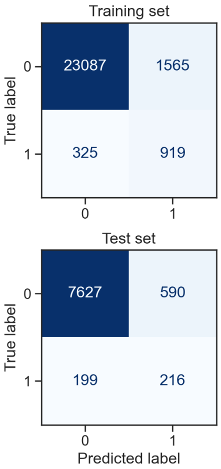
- Our goal was to maximize `f1-score`
- We tried:
  - logistic regression
  - support vector machine
  - random forest
  - xgboost
- First two models gave us very poor results
- We performed hyper-parameter tuning on random forest and xgboost

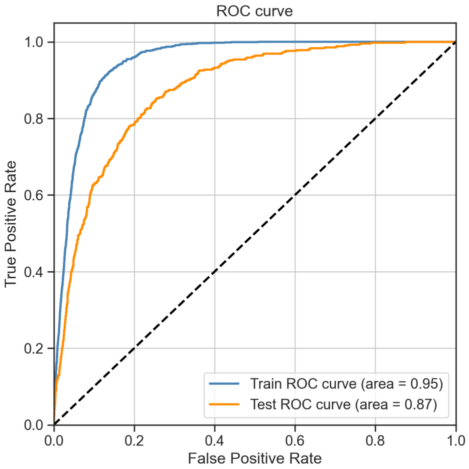
Models

RandomForest

Training set	
precision	0.3699
recall	0.7387
f1	0.4930

Test set	
precision	0.2679
recall	0.5204
f1	0.3538

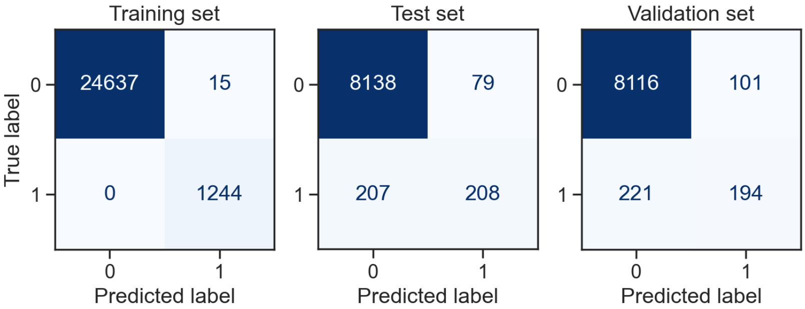




Models

XGBoost

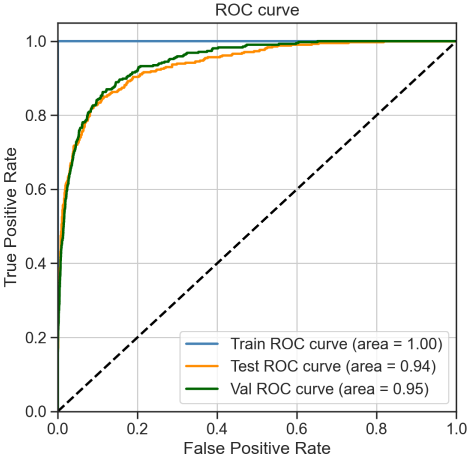
	Training	Test	Validation
precision	0.9881	0.7247	0.6576
recall	1.000	0.5012	0.4675
f1	0.9940	0.5926	0.5465



Models

XGBoost

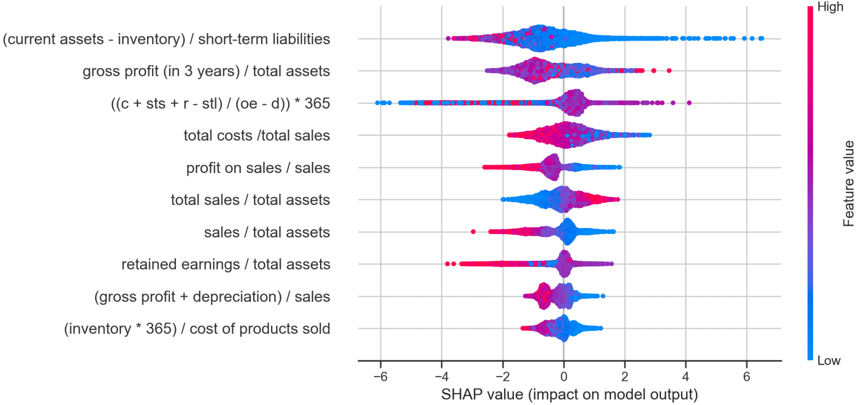
- XGBoost model seems a little bit overfitted
- Attempts to prevent overfitting had negative impact on test model score



XAI

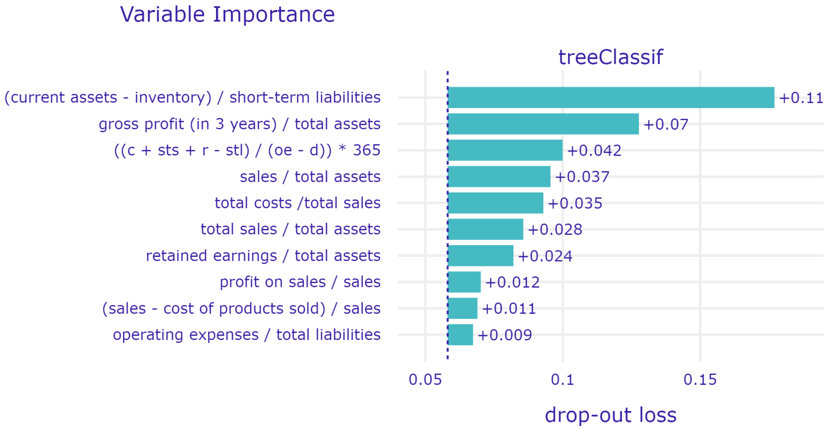
XAI

Shap summary plot



XAI

Variable importance



Summary

- Problems with data:
  - unbalanced classes
  - missing values
  - outliers
  - domain-specific language
- Objective: maximize `f1-score`
- Best model: XGBoost with 0.55 score on validation set
- Important features:
  - `current assets - inventory) / short-term liabilities`
  - `gross profit (in 3 years) / total assets`
  - `total costs / total sales`
  - `sales / total assets`