FRANKFURT UNIVERSITY OF APPLIED SCIENCES

credit shelf

Semi-Automatic Risk Analysis Interfaces

TRAN BAO KHANH DANG

Content

1. Introduction

Business context:

- Development of economic globalization
- Small and Medium-sized Enterprises (SMEs) play a crucial role for economic growth

⟶ Business lending becomes a major target for banks and investors

⟶ The need of increasing the power of Risk Analytics

1. Introduction

Risk Analytics:

- Investors don’t give credit to anyone who asks for it
- They need to consider the risks associated with their investments:
    - Business default or not be able to pay back the loan
    - Markets collapse

⟶ Analyze the borrower's background & behavior

⟶ Calculate the creditworthiness:

- Credit Scoring: risk categories (e.g. “Good” or “Bad”)
- Credit Rating: grades (e.g. {A, B, C, D})

1. Introduction

Challenges:

- Require lots of time and implicit knowledge from credit experts
- Some qualitative aspects cannot easily be explained: human feelings, opinions, common sense…
- Lack of information

1. Introduction

Solution:

Preference Learning: Giving relative estimation is easier than absolute estimation

⟶ Implement a tool that:

1. Allows credit experts to do relative estimation (i.e. comparing A with B)
2. Achieve the final ranking of preference for all cases

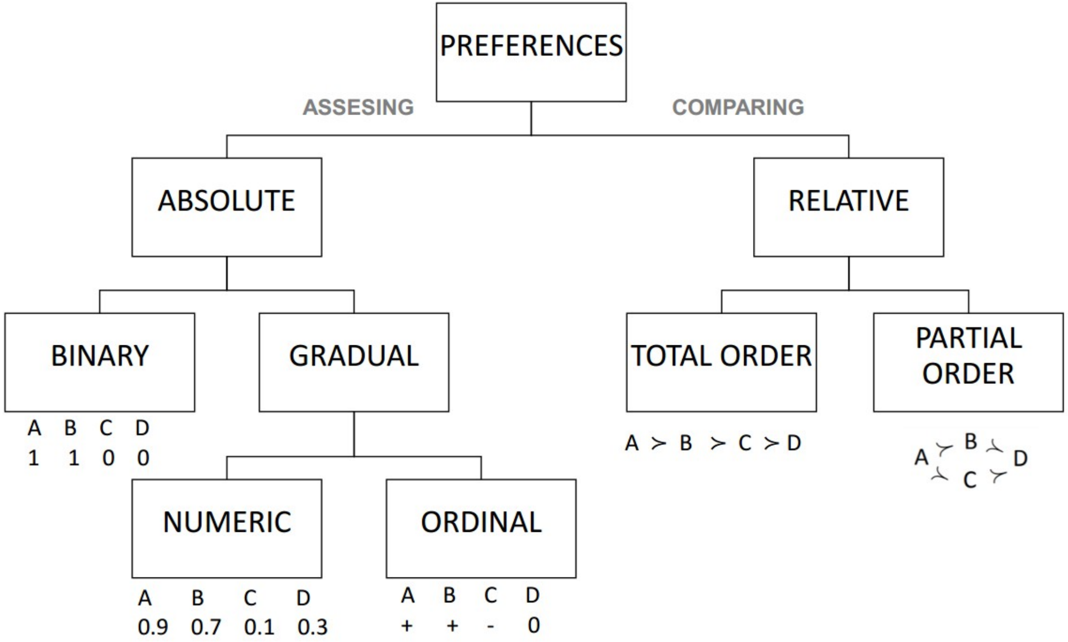
2. The concept of Preference Learning

Preference Learning:

- *Preference:* the choice for one alternative over another or others
- A subfield of Machine Learning: classification method based on observed preference information
- Output of Preference Learning system:
  - A ranking list of inputs based
  - A comparison to whether A is better than B
- Applications: recommender systems, learning-to-rank search results...

2. The concept of Preference Learning

Types of Preference:



→ (ORDINAL) REGRESSION

→ CLASSIFICATION/ RANKING

3. Application UI implementation

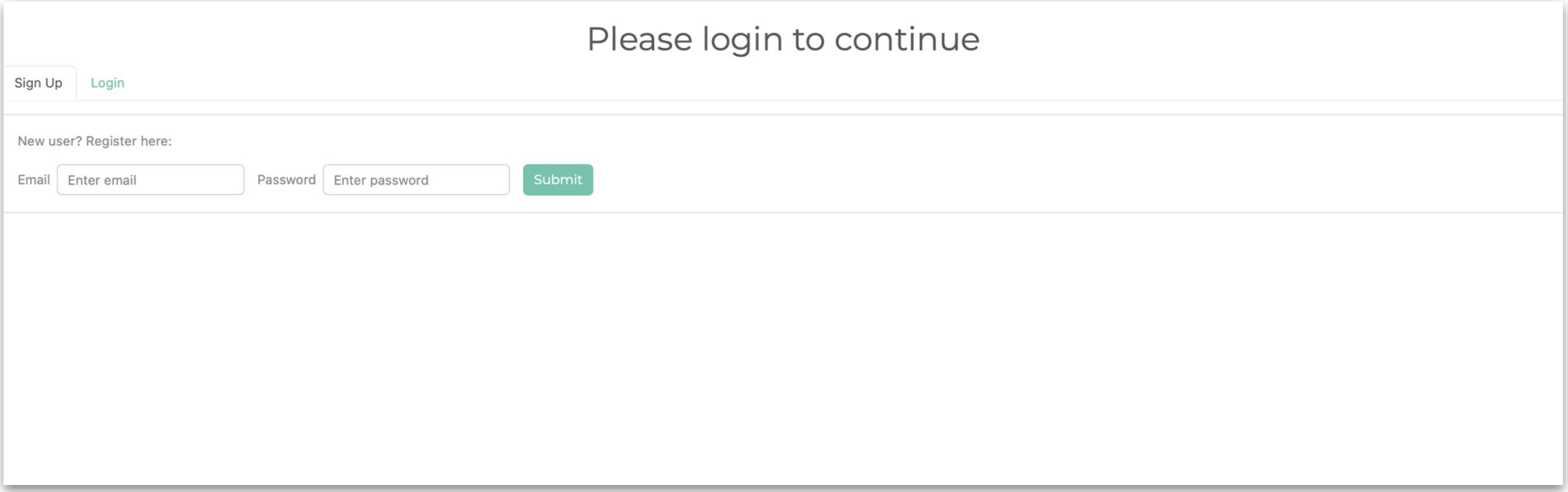
- Language: Python 3
- Frameworks: Flask & Dash

⟶ Runs in the web browser

- Main layouts:
  - Login
  - Company voting

3. Application UI implementation

1. Login:



3. Application UI implementation

2. Company Voting:

Financial dataset

(Millions Euros)

Logout

Company 1:

Company Name	Year	Revenues	Depreciation and Amortization	Operating Profits	Interest Expense	EBITDA	Net Profit	EBITDA %	Op Prof %	Op Prof Growth	Rev growth	Net Profit %
Redo Water Systems GmbH	2014	€2.7400	€0.0000	€0.3850	€0.0800	€0.385	€0.305	14.0511%	14.0511%	0.0%	0.0%	11.1314%
Redo Water Systems GmbH	2015	€2.8100	€0.0000	€0.4230	€0.0800	€0.423	€0.343	15.0534%	15.0534%	0.0713%	2.5547%	12.2064%
Redo Water Systems GmbH	2016	€7.7040	€0.0000	€0.2620	€0.0800	€0.262	€0.182	3.4008%	3.4008%	-0.7741%	174.1637%	2.3624%

Company 2:

Company Name	Year	Revenues	Depreciation and Amortization	Operating Profits	Interest Expense	EBITDA	Net Profit	EBITDA %	Op Prof %	Op Prof Growth	Rev growth	Net Profit %
WEMARO Tools GmbH	2014	€2.6460	€0.0220	€0.0670	€0.0250	€0.089	€0.042	3.3636%	2.5321%	0.0%	0.0%	1.5873%
WEMARO Tools GmbH	2015	€2.5630	€0.0160	€0.0430	€0.0240	€0.059	€0.019	2.302%	1.6777%	-0.3374%	-3.1368%	0.7413%
WEMARO Tools GmbH	2016	€2.5720	€0.0120	€0.1560	€0.0280	€0.168	€0.128	6.5319%	6.0653%	2.6152%	0.3512%	4.9767%

Which company would rank better?

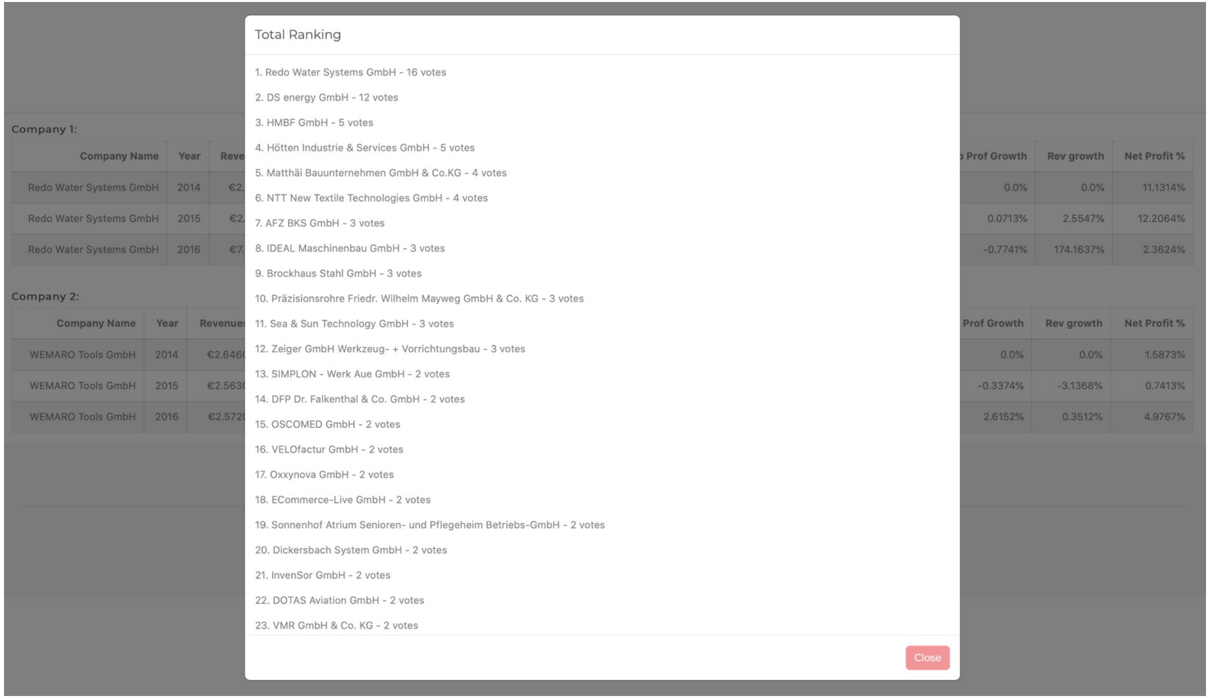
Redo Water Systems GmbH | WEMARO Tools GmbH

You have completed 0/ 251 votes

Total Ranking

3. Application UI implementation

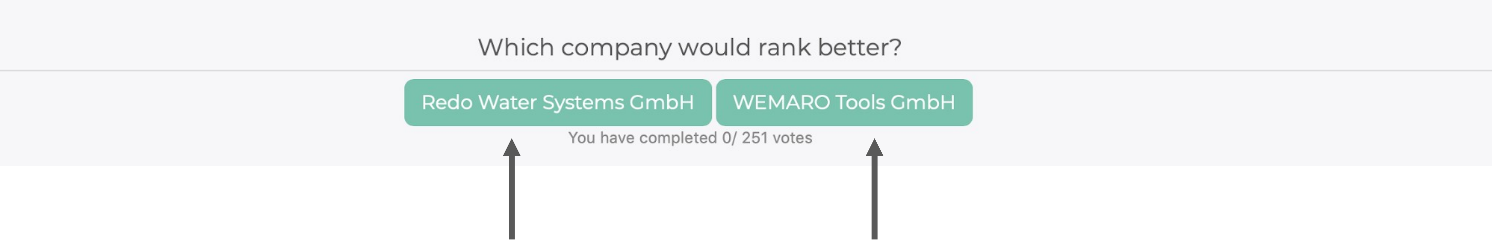
2. Company Voting:



3. Application UI implementation

Collected data:

```
mysql> select * from action_capture;
+-----+---------+-------------------+-------------------------+---------------------+-------------------------------------+-----------------+------+---------------------+
| id  | pair_id | company_id_voted  | company_name_voted      | company_id_compared | company_name_compared               | button1_clicked | user | timestamp           |
+-----+---------+-------------------+-------------------------+---------------------+-------------------------------------+-----------------+------+---------------------+
| 443 | 279     | 1192              | Redo Water Systems GmbH | 1501                | WEMARO Tools GmbH                   |               1 | 1    | 2020-11-02 13:36:39 |
| 444 | 822     | 1128              | DS energy GmbH          | 14220               | IDEAL Maschinenbau GmbH             |               1 | 1    | 2020-11-02 13:36:40 |
| 445 | 1190    | 1710              | HMBF GmbH               | 22165               | Hötten Industrie & Services GmbH    |               1 | 1    | 2020-11-02 13:36:41 |
| 446 | 1282    | 276               | AFZ BKS GmbH            | 64984               | Matthäi Bauunternehmen GmbH & Co.KG |               1 | 1    | 2020-11-02 13:36:47 |
+-----+---------+-------------------+-------------------------+---------------------+-------------------------------------+-----------------+------+---------------------+
```



Button 1 **Button 2**

Company 1 **Company 2**

4. Machine Learning methods and models

A pair of company A and B

A or B is better?

Input → **Model** → **Output**

{company A, company B} → **company A**

{company C, company D} → **company D**

4. Machine Learning methods and models

Training Set

X: an array of 66 dimensions

pair_id	year_1	revenue_1	depreciation_ amortization_1	...	year_2	revenue_2	...	user	button1_ clicked
552	2016	60.2990	0.7730	...	2016	27.5460	...	1	1
552	2017	60.8780	0.7990		2017	18.1310		1	1
552	2018	59.4390	0.8580		2018	23.3190		1	1
738	2016	17.1080	0.0940		2016	3.9500		15	0
738	2017	28.7580	0.2110		2017	4.1900		15	0
738	2018	32.7630	0.3100		2018	4.5700		15	0

Company 1

Company 2

Y

4. Machine Learning methods and models

Models used for classification problem:

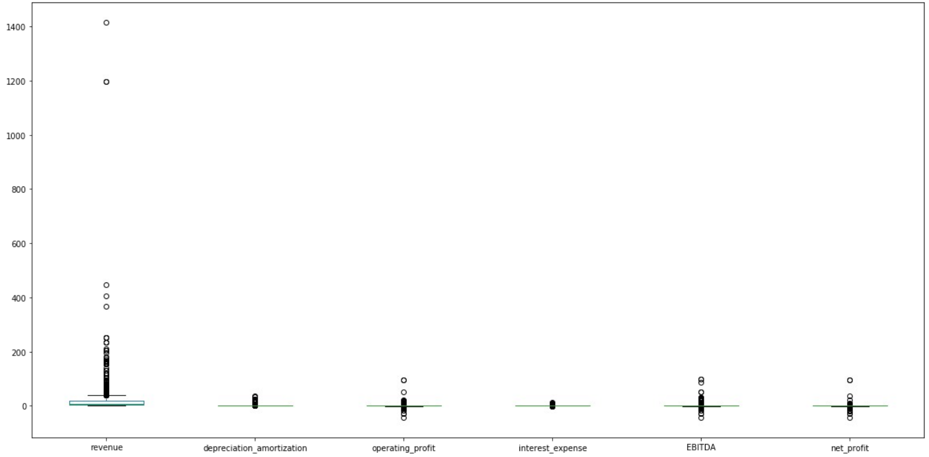
- Support Vector Machine (SVM)
- k-Nearest Neighbors (kNN)
- Using *scikit-learn* from Python library

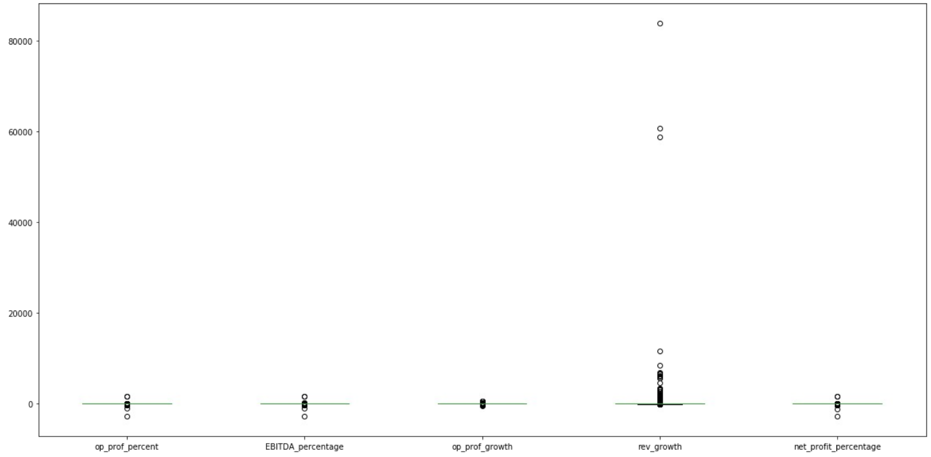
5. Results & Discussion

- 4 end users testing
- 760 data points/ votes collected

5. Results & Discussion

Data distribution





→ requires normalization before model training

5. Results & Discussion

SVM Model Performance Optimization:

- Tuning SVM parameter (C, gamma)
- Remove duplicate X in training data
- Using NuSVC with advantage of using a parameter *nu* to effectively control the number of support vectors
- Scale data

5. Results & Discussion

SVM Model Performance Optimization:

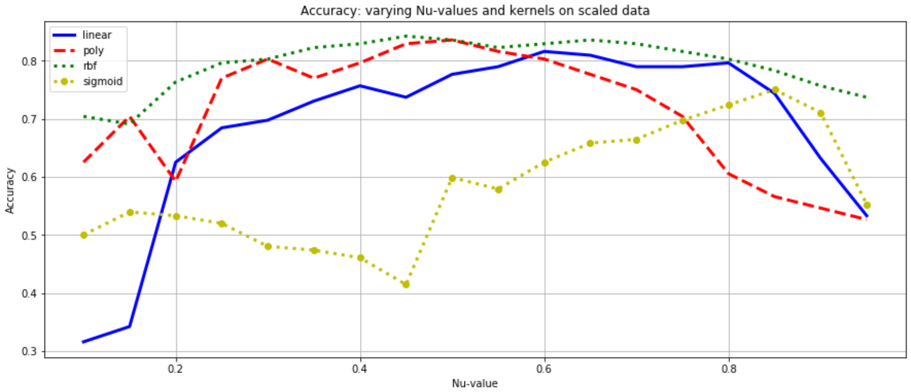
- Accuracy: increases and more consistent with repeatability test
- Precision & Recall: improve

```
Accuracy score SVM: 0.5526315789473685
Confusion matrix SVM:
 [[81  1]
 [67  3]]
Precision: 0.75
Recall: 0.04285714285714286
```

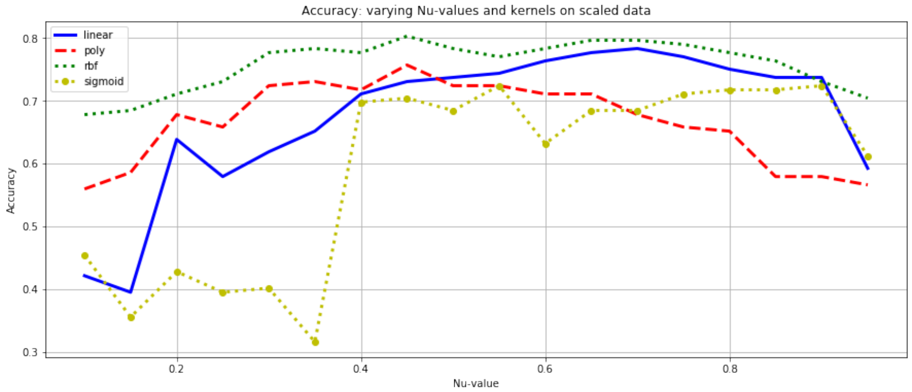
```
Accuracy score SVM: 0.8355263157894737
Confusion matrix SVM:
 [[71 14]
 [11 56]]
Precision: 0.8
Recall: 0.835820895522388
```

5. Results & Discussion

SVM Model Performance Optimization:



Test run 1



Test run 2

⟶ Best performance using RBF kernel

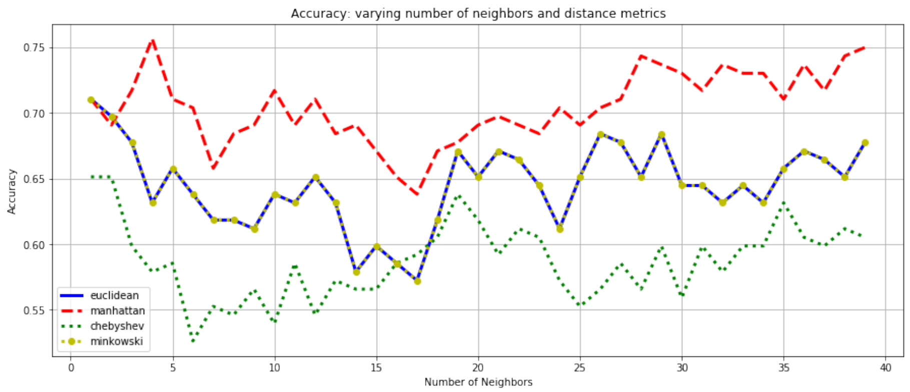
5. Results & Discussion

kNN Model Performance Optimization:

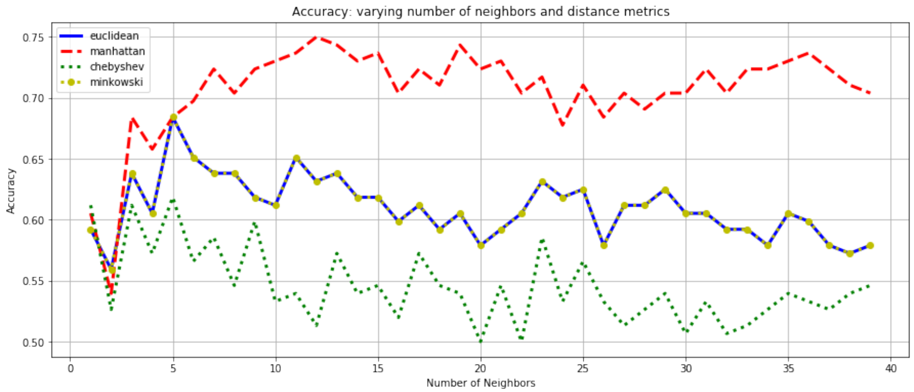
- Selection of distance function
- Data scaling
- Choosing the right k-value

5. Results & Discussion

kNN Model Performance Optimization:



Test run 1



Test run 2

→ Best performance using Manhattan distance function

5. Results & Discussion

SUMMARY:

- Highly adaptable models
- An accuracy of 78% on average
- SVM is a better choice of model than kNN:
  - kNN performance depends on training set and k-value
  - SVM is more reliable

- Withdraws:
  - Lack of training data
  - Invisible factors

6. Conclusion

- Developed a web-application based on the concept of Preference Learning to collect and study the preferences of risk experts
- A practical framework for the implementation of credit decision making using Machine Learning
- Models with highly adaptable features

Future perspectives:

- Develop comprehensive Machine Learning algorithms
- The data’s non-linearity properties need to be studied and accounted for