Dr Karol Przanowski

kprzan@sgh.waw.pl

Project Description for the course: 220311

„ Credit Scoring - business process automation”

General description of the tasks

The aim of the project is to familiarize the student with the construction of scoring models and with difficulties in their implementation. The project also allows the student to observe how significantly rejected applications can influence the results. Finally, the final reports and results allow you to develop intuition and develop the experience of influencing the list of decision engine rules with financial indicators of the entire acceptance process, which consists of two processes: acquisition - instalment loan and cross selling - cash loan. It is a unique experience of building a fully automatic process, where decisions are made based on predictive models completely automatically. It is a very important lesson for the student, because in the near future only such processes will be in business and you need to be able to configure and maintain them.

To-do List

The project may be carried out by a team of up to five people.

The goal of the project is to build an optimal strategy for the loan approval process, in order to maximize final profit on both products in the 1975-87 period. At a minimum, the results should be an improvement to the generic strategy which brings a profit of PLN 663,327 in the period 1975-87.

The list of individual tasks are as follows:

1. Improving any of the stages of building a model by writing or adding your own code in SAS 4GL. This can be, for example: counting additional statistics, making an additional graphic report, etc. You can also correct or propose additional Python codes for building a scorecard or other model building elements, including AI model codes, or sophisticated variable selection methods. Elements of XAI are also welcome, i.e. interpreting models, creating good and visualized documentation of the AI model.
2. Building a risk model measuring the probability of default12 = 1 for instalment loan (product = 'ins'), designation of the PD Ins model.
3. Building a risk model measuring the probability of default12 = 1 for a cash loan (product = 'css'), designation of the PD Css model.
4. Building a risk model measuring the probability of default_cross12 = 1 for a cash loan at the time of applying for instalment loan, designation of the PD Css Cross model.
5. Building a marketing model measuring the probability of cross_response = 1 event when applying for instalment and / or cash loan, designation of the PR Css Cross model.

6. Create rules for the decision engine and implement all described models. Launching the strategy of the credit acceptance process with new rules and calculating the final profit and loss reports for the period 1975-87.

Models can be built either via ASB_SAS or via ASB_PYTHON. It is allowed that one model can be an AI model, i.e. not a scorecard, but a good and interpretable description of the model must be presented. It also requires writing your own scoring_code.sas tool, which will run the Python class directly from SAS.

It is also possible to build more than four models by proper segmentation of the applicant's portfolio.

The division of tasks can be chosen freely, with only one restriction: the first task cannot be the only task of one student.

Each task should end with appropriate electronic documentation sent then to lecturer.

The project is subject to defence, during which all results are evaluated, and the documentation is the most important material for discussion and final evaluation.

The defence consists in convincing the tutor to the obtained results, the given form of the model, inequalities in categories, patterns in the rules of the strategy, etc. What counts here will be mainly the list of arguments raised by the students. Ideally, each of the project participants should talk about their part.

Further chapters describe the tasks listed above in detail. In addition, two chapters help launch the project for the first time, with regards to a master strategy and model building.

Final assessment and points

The following criteria are considered in assessment of a project defence by the lecturer, below are provided maximal points per category:

Criterion	Points
Profit – financial result, max 6 pts	6
The method of arguing for the adopted acceptance strategy, max 5 pts	5
Flaws, strengths and weaknesses of the models, max 4 pts	4
Flaws of statistics and KPI of the models, max 3 pts	3
Flaws, strengths and weaknesses of the reports, max 2 pts	2
Sum	**20**

How to run a project with a benchmark strategy

In the directory ...\CS-AUT\software\PROCSS_SIMULATION\process\codes\ there is a model code decision_engine.sas (100% acceptance), which should be modified by the student. This code refers to directory of subdirectories ...\CS-AUT\software\PROCSS_SIMULATION\process\calibration\, in which there are configuration files of built models, which the students should also replace with new ones built by them. In each folder of the model there is the most important code called scoring_code.sas, which by the %include statement is included in the code decision_engine.sas. In this code, the entire processed month is scored with four models, adding not only the values of scores, but also the probability values of modelled events. Finally, the final datastep at the end of this code calculates the correct rules by defining the final acceptance decisions and the reasons for the rejects.

The whole process starts with the batch.bat file from the directory ...\CS-AUT\software\PROCSS_SIMULATION\codes\. The batch command refers to the code all_contents.sas. The final report with financial indicators is made in the folder ...\CS-AUT\software\PROCSS_SIMULATION\process\reports\ and is named profit_1975_1987.html.

How to calibrate models for PD and PR and how to determine cut-off points

If all models are already built, there can be more than four. Each model, with its code scoring_code.sas, should be copied to the appropriate subdirectory in the directory ...\CS-AUT\software\PROCSS_SIMULATION\process\calibration\. The folder also contains the code calibration.sas, which shows how to determine the probabilities of modelled events and how to determine the appropriate cut-off points on a standard strategy. Many codes should be commented on here. The basic idea of determining the PD or PR parameter is to run the appropriate logistic procedure with the outest = beta option, and then copy from the beta dataset their values to the formula with the exp() function in the appropriate place. Setting cut-off points is an interesting challenge. You can approach this task with one or two products. In the second case, the rank procedure is used and various groups of applications are analysed for the first and second product based on probability parameters.

Model construction and documentation

The model should be built on a properly selected sample. Always build on data from the period 1975-87 (condition in SAS 4GL '197501'<=period<='198712'), and on properly selected product criteria (type product='css' or product='ins') and decision (decision='A'). In case of Python the same criteria should be used. ABT variables are built only on the basis of information about accepted customer applications in the process, and thus only on approved credit histories by our process, this may cause a bias of estimation and model building that takes place on a portfolio that was otherwise accepted.

Each processed strategy creates a set of abt_app.sas7bdat in the directory ...\CS-AUT\software\PROCSS_SIMULATION\process\data\, which should then be copied to the appropriate modelling directory, e.g. to ...\CS-AUT\software\ASB_SAS\inlib\ or to …\CS-AUT\software\ASB_PYTHON\. The selection of the initial strategy for building the model is one of the tasks of this stage.

The final model with its parameterization files and reports should be copied to many appropriate places:

1. Copy the correct files to the processing directory of the new strategy, i.e. in case of PD INS model, to ...\CS-AUT\software\PROCSS_SIMULATION\process\calibration\model_ins_risk\.
2. Copy to the appropriate structures as in the directory ...\CS-AUT\materials_all\example_of_project_documentation\.

3. Modify the file model_documentation_PD_INS.xlsx, which is the defence material from the folder ...\sas_cs_en\project\example_of_project_documentation\example_of_model\ or use a proper Python code to get the same documentation form like in case of building by Python, see directory ...\CS-AUT\materials_all\example_of_project_documentation\example_of_model_PYTHON\.

Strategy building and documentation

At the beginning, you need to collect all models, place the codes for scoring in the appropriate folders and perform calibration. Then define the correct rules and save them in the decision_engine.sas code. The rule "998 not active customer" must not be changed. Rules may be time-dependent, may be more or less complicated, as long as they are based on available information at the time of application, i.e. based on data from ABT and probabilities from models. After starting processing, copy the result files to the directory as in the example ...\CS-AUT\materials_all\example_of_project_documentation\example_of_strategy\ and modify the file strategy_documentation.xlsx and send it to the lecturer.

You should also prepare the proper structure of all files and directories as in the directory ...\CS-AUT\materials_all\example_of_project_documentation\example_of_strategy\ to forward to the teacher.

Improved codes and documentation

Just write the code and send it in the package of other documents: model documentations and strategy report. An example is in the folder ...\CS-AUT\materials_all\example_of_project_documentation\example_of_extra_code\.