Advanced Coding For Data Analytics (A.Y. 24/25)

BSc in Management and Computer Science
Luiss Guido Carli

Project:
Data Analytics for a Credit Score

Teachers:
Andrea Coletta (acoletta@luiss.it), Alessio Martino (amartino@luiss.it)

Guest Speaker:
Mattia Villani (https://www.linkedin.com/in/mattiajacopovillani/)

Abstract

Credit scoring plays a crucial role in financial decision-making, helping institutions assess risk and make informed lending decisions. With the increasing availability of credit-related data, machine learning and AI algorithms offer a powerful tool to improve the accuracy and efficiency of credit score classification.

In this project, students will tackle a real-world problem: using a historical credit-score dataset, they will apply AI techniques to automate credit risk assessment. However, working with real financial data comes with challenges, such as handling missing values and addressing potential biases in the model.

The project will guide students through key steps, including: a) data preprocessing, b) training an initial model, c) analyzing biases, d) implementing mitigation strategies, and e) refining the final model. By the end of the project, students will gain hands-on experience in building fair and effective credit scoring models, ensuring a more transparent and responsible approach to financial decision-making.

Project data:

We use anonymised, randomized and synthetic data from a Credit Score dataset. A **credit score** is a numerical representation of an individual's financial reliability, often used by banks and lenders to assess creditworthiness. In this project, the credit score is categorized into four levels: **Good (3), Standard (2), Poor (1)** and **Zero (0).** The credit score serves as the target variable, which can be treated as a numerical value for general purposes.

Disclaimer: The biases present in the dataset are artificially generated and do not reflect any actual beliefs or real-world facts.

You can download the data from luiss-learn:

credit_score.csv

An example is provided below:

	Month	Name	Age	...	Occupation	Annual_Income	Credit_Score
0	NaN	Aaron Maashoh	NaN	...	Manager	19114.12	3
1	February	Aaron Maashoh	23.0	...	Manager	19114.12	3
2	NaN	NaN	-500.0	...	Manager	19114.12	3
3	April	Aaron Maashoh	23.0	...	Manager	19114.12	3
4	May	Aaron Maashoh	23.0	...	Manager	19114.12	3

The dataset includes details about various system users, with each user having historical data that may reflect changes in their credit score and other information. **Note**: For simplicity, you can exclude the time dimension and focus solely on each individual row.

Each row provides information about a user, such as:

**** Customer_ID ****
Represents a unique identification of a person

**** Month ****

Represents the month of the year

**** Name ****
Represents the name of a person

**** Age ****
Represents the age of the person

**** SSN ****
Represents the social security number of a person

**** Occupation ****
Represents the occupation of the person

**** Annual_Income ****
Represents the annual income of the person

**** Monthly_Inhand_Salary ****
Represents the monthly base salary of a person

**** Num_Bank_Accounts ****
Represents the number of bank accounts a person holds

**** Num_Credit_Card ****
Represents the number of other credit cards held by a person

**** Interest_Rate ****
Represents the interest rate on credit card

**** Num_of_Loan ****
Represents the number of loans taken from the bank

**** Type_of_Loan ****
Represents the types of loan taken by a person

**** Delay_from_due_date ****
Represents the average number of days delayed from the payment date

**** Num_of_Delayed_Payment ****
Represents the average number of payments delayed by a person

**** Changed_Credit_Limit ****

Represents the percentage change in credit card limit

**** Num_Credit_Inquiries ****
Represents the number of credit card inquiries

**** Credit_Mix ****
Represents the classification of the mix of credits

**** Outstanding_Debt ****
Represents the remaining debt to be paid (in USD)

**** Credit_Utilization_Ratio ****
Represents the utilization ratio of credit card

**** Credit_History_Age ****
Represents the age of credit history of the person

**** Payment_of_Min_Amount ****
Represents whether only the minimum amount was paid by the person

**** Amount_invested_monthly ****
Represents the monthly amount invested by the customer (in USD)

**** City ****
Indicates the location where the person resides.

**** Street ****
Specifies the specific street address of the person's current residence.

While the last column indicates the **Credit Score value.**

NOTE:

- The datasets contains several missing values (that's happen a lot in real life)
- The datasets contains wrong values (e.g., negative ages, to handle)
- The datasets contains possible biases, which is imperative to study and drop in a sensitive application. Not always historical trends should predict or alter future choices.
- The datasets may contain unblanaced classes.

Project goal:

For simplicity we are breaking down the project goal in three main steps, but feel free to work directly with your own schedule.

Task 1 – Load the datasets

Setup your python environment and download the credit-score data. Remember you have one file, look at the whole structure and the data.

Hint: You can use pandas!

Task 2 – Analyzing the credit-score data

NOTE: the following steps have increasingly complexity, and we evaluate your project accordingly to them. But if you complete correctly only 4/5 steps, you can still get a good evaluation and pass the project.

Step 1) Perform Exploratory Data Analysis:

- Analyze and visualize the distribution of Credit Scores.
- Explore client features to identify useful data that might influence the credit score.
- Gain insights into the relationship between each feature and the Credit Score.
- Identify any missing or erroneous data.

Step 2) Handle Missing and Incorrect Values:

- Select an appropriate method (e.g., mean, median, mode, forward/backward filling, or advanced AI techniques) to handle missing values.
- Use your background knowledge and choose a method to detect and correct incorrect values (e.g., validating that age cannot be negative).
- *Tip: Consider visualizing the data distribution before and after addressing missing and incorrect values.*

Step 3) Address Class Imbalance:

- If you find an imbalanced distribution of Credit Scores, decide how to handle it: re-balance the dataset using over-sampling, under-sampling, or synthetic data generation.

Step 4) Train the Model:

- Choose an appropriate method for prediction (e.g., regression or classification, such as Random Forest) to predict the Credit Score based on client features.

Step 5) Analyze Results:

- Evaluate and report the model's performance.
- Identify the most important features and their interaction with the Credit Score, using simple *explainability techniques (optional)*.

Step 6) Address Bias and Re-train:

- If you identify any biases in the model (e.g., the Credit Score being influenced by ethnicity), resolve the issue (e.g., by dropping the biased feature) and re-train the model.
- Re-analyze and report the model's performance after adjustments.

Task 3 – Visualization and reporting:

Please present all your discoveries in the presentation, using the approach and visualization methods you prefer.

Extra: You can also build an interactive tool, using streamlit, to predict the credit-score for a new user at run-time.

Project Report

The project is MANDATORY and accounts for 70% of the final grade, and we expect you to work in a group of 3 students.

Deliverable:
You must provide a presentation (in case of ppt or pdf at most 8-10 slides), describing your project, few implementation details, and the analysis done with the results.

You should send us (Coletta and Martino) an e-mail with a zip file containing your presentation/slides and the code. The zip should be named using your group id.

When you send us your project, please, use as subject "Coding2025: group X", where X is your group ID assigned by the instructors.

AI policy:
Most of the AI tools work really well and produce high quality code and software, however hallucination is around the corner (especially in state-of-art data analysis tasks). Coders should be able to understand – **and explain** -- the output of these AI tools, and use them in a conscious way. In this course we kindly ask to code and try by yourself, made errors, and eventually learn. Copilot and chat-GPT will be much better when you can better evaluate their answers!

Note:
If you deliver the project during the last lesson of the course you will get 1 additional point. More info on luiss-learn.

References:

We refer to the slides of our guest speaker *Mattia Villani* for a more extensive introduction to biases and fairness in AI systems.

A pdf version will be available through the course website on luiss-learn.

We really hope you enjoyed the lecture from our guest speaker! As always, if you have any comments/questions feel free to send us an email!

Good luck!!!