ML core lecture SS2024

Project Description

Prof. Dr. Isabel Valera, Jonas Klesen, Ayan Majumdar

Version 1.0

1 Introduction

In the following, we describe the practical project that we have prepared for the students of the Machine Learning (ML) core lecture of Summer 2024. The idea of this project is that the students bring into practice the theoretical concepts introduced during the lectures.

This year the project consists on predicting the healthcare usage of individuals, and will involve both regression and classification tasks. The evaluation of the healthcare project is divided into two main parts that contribute differently to your final grade for the course: the report and the challenge. Details on grading are provided in section 5.

1.1 Participation in the project

The participation to the project is **fully voluntary but highly recommended**, as it provides an opportunity for students to gain hands-on experience in ML. Students are expected to work in groups of up 3 members and must register their group via CMS (already possible). **Participating in the project means submitting both the report and the predictions for the challenge**. It is not possible to participate in the challenge but not submitting the report, or the other way around. Refer to Section 6 for the expected timeline.

2 Problem Description

In healthcare, effective allocation of limited resources is crucial to adequately address the needs of all patients optimally. To better inform healthcare allocation, predictive analytics have become crucial in recent times to predict the healthcare usage of individuals.

Your objective for this project is to develop machine learning models capable of forecasting patients' healthcare utilization and the total cost incurred by each individual. To enable learning such models, you are provided with a dataset compiled from a survey of patients in the United States. This dataset encompasses various features related to health data, demographics, and specific services utilized by individuals.

The specific target features in this dataset and the respective tasks pertaining to these target features are as follows.

1. **Total Medical Expenditure**: This is a continuous feature measuring the total healthcare expenditure in US dollars. Your goal is to build a regression model that, when deployed, can take the features of previously unseen patients as input and predict the total medical expenditure.

2. **Healthcare Utilization**: This is a binary categorical feature that can either indicate LOW usage or HIGH usage. Your goal is to build a binary classification model that, when deployed, can take the features of previously unseen patients as input and predict the utilization category of the healthcare system.

2.1 Datasets

We share three datasets with you, one training set and two test sets. For each set, the data will contain 108 features and, in the case of the training set, two target features. The targets are `TOT_MED_EXP` indicating the continuous healthcare expenditure in US dollars and `UTILIZATION` indicating the binary utilization class (`LOW` or `HIGH`). The data features include:

- **Demographics**: `RACE`, `SEX`, `AGE`, `PANEL` (survey panel number), `REGION` (Census region), `MARITAL_STAT` (marital status), `POVRTY_CAT` (poverty category), `POVRTY_LEV` (poverty level), `EDU_YRS` (education years), `EDU_DEG` (highest education degree), `SPOUSE_PRSNT` (marital status with spouse present), `STUDENT_STAT` (student status), `UNION_STAT` (union status), `NUM_DEP_OUT_REP_UNT` (number of dependents outside survey

reporting unit), EMPLYMT (employment status), OCCUP (occupation), NON_ENG_LANG (non-English language spoken).

- **Personal**: PUB_ASST (public assistance money), TAX_FORM_TYP (tax form type submitted), FOOD_STMP_MNTHS (number of months food stamps purchased), FOOD_STMP_VAL (value of food stamps), MIL_ACTIV_DUTY (military active duty), HON_DISCHARGE (honorary discharge from army), INSUR_COV (insurance coverage), TOT_INCOME (total income), EMPLYR_INS (employer offers insurance), CHILD_SUPP (child support), PROB_WKIDS (problem with kids), FAM_INCOME (family income), PROB_BILL_PAY (problem with bill payments), DELAY_PRESCT_MED (delay getting prescription medication), DAYS_CAREOTHR_NOWORK (days not working due to care for others), PENSN_PLAN (pension plan), NO_WORK_WHY (reason for not working).

- **Health-related**: WEIGHT, HEALTH_STAT (perceived health status), MENTAL_HLTH (perceived mental health), CHRON_BRONCH (chronic bronchitis), JNT_PAIN (joint pain), PREGNT (pregnant), WALK_LIM (walking limitation), ACTIV_LIM (activity limitation), SOCIAL_LIM (social limitation), COGNTV_LIM (cognitive limitation), BM_IDX (BMI), MULT_HIGHBP (multiple high blood pressure readings), HOUSEWRK_LIM (housework limitation), SCHOOL_LIM (school limitation), ADV_NO_FAT_FOOD (advised to restrict high-fat food), ADV_EXERCISE_MORE (advised to exercise), ADV_DNTL_CKP (advised dental checkup), FREQ_DNTL_CKP (frequency of dental checkup), RSN_NO_DNTL_CKP (reason for no dental checkup), RSN_NO_MED_CKP (reason for no medical checkup), DOC_CK_BP (doctor checked blood pressure), TAKE_RISK (prone to taking risks), ADV_BOOST_SEAT (advised booster seat), WHEN_ADV_BOOST_SEAT (when advised booster seat), FEEL_DEPRS (feels depressed), ADV_NO_SMKG (advised no smoking), AGE_DIAG_ADHD (age when diagnosed ADHD), PROB_WBHV (problem with home behavior), WEAR_SEATBLT (wear seat belt), WHEN_ADV_LAP_BLT (when advised lap belt), WHEN_LST_ASTHMA (when last asthma episode), ADV_LAP_BLT (advised lap belt), ADV_EAT_HLTHY (advised to eat healthy), DOC_TIM_ALN (doctor spent time alone), APPT_REG_MEDCARE (made routine appointment medical care), LOST_ALL_TEETH, ASPRN_REG (regular aspirin usage), DIFF_ERRND_ALN (difficulty doing errands alone), DIAB_KIDNY (diabetes-related kidney issue), DIAB_INSLN (diabetes insulin use), DIAB_MED (diabetes medicine use), DISPSN_STAT (patient disposition status), TIME_LAST_PSA (last PSA test), WHEN_ADV_EXERCISE (when advised to exercise more), DEAF, BLIND, LAST_FLU_VAC (last flu vaccine), UNABL_PRES_MED (unable to get proper medicine), HEAR_AID (need hearing aid), LAST_REG_CKP (last regular checkup), DAYS_ILL_NOWORK (days miss work for illness), DAYS_ILL_NOSCHL (days miss school for illness), HIGH_BP_DIAG, COR_HRT_DIAG (coronary heart disease diagnosis), ANGINA_DIAG, HRT_ATT_DIAG (heart attack diagnosis), OTH_HRT_DIAG (other heart-related diagnosis), STROKE_DIAG (stroke diagnosis), EMPHYM_DIAG (emphysema diagnosis), HIGHCHOL_DIAG (high cholesterol diagnosis), CANCER_DIAG, DIAB_DIAG (diabetes), ARTHR_DIAG (arthritis), ARTHR_TYPE (arthritis type), ASTHM_DIAG (asthma), ADHD_DIAG, NUM_PRESCR_MEDS (number of prescription medicines), DIFFIC_HEAR (difficulty hearing), DIFFIC_SEE (difficulty seeing), SMOK (smoking), OVR_FEEL_14 (overall feeling rating 14 days), MENTAL_HLTH_SCR (mental health score), PHY_HLTH_SCR (physical health score), OVR_FEEL_30 (overall feeling rating 30 days).

The training data has information and labels for 15000 patients. The first test dataset, named **test_public.csv**, has features of 4791 patients without labels. You can access it right now on CMS. The second test dataset, named **test_private.csv** has features of 5000 patients without labels. This dataset will be accessible closer to the final submission deadline and will be used to compute the final leaderboard of the challenge.

Note that the datasets we provide you have been only partially cleaned. Thus, it is important to try different pre-processing approaches for feature selection, feature encoding, outliers, etc. You are also encouraged to explore different trustworthy aspects for your ML modeling and deployment, e.g., exploring fairness and explainability aspects (see point 5 in Section 3 below).

2.2 Jupyter Notebook Example

We have uploaded a Jupyter Notebook in CMS that contains a description of the data and an example on how to load and preprocess the data, visualize it, and use it to train a linear regression and classification model. The notebook also includes instructions on how to format and save your predictions for submission to CMS (both for the leaderboard and the final ranking). We highly recommend that you use this notebook as a starting point for your solution.

3 Report Instructions

Every student group/team participating in the project should submit a report (.pdf file) using the LaTeX template provided in CMS. The report must be **at most 6 pages long (references excluded)** and contain detailed information on the methodology applied to select the final model and make the necessary predictions to participate in the challenge. More specifically, the report should contain information on the following aspects:

1. **Data analysis & preprocessing:** The report should describe any considered approach used for data analysis and prepocessing to prepare the input data (features) to the ML model.
2. **ML modeling:** The report should include a short description of the different models (i.e. the classifiers and regression models) applied to the data, specifying the used python libraries (if any).
3. **Model selection:** The report should detail the methodology followed to compare the different ML models (and, if applicable, data preprocessing approaches), as well as to select the final model used to make the predictions for the challenge.
4. **Empirical results:** The report should provide a summary and description of the empirical results that have led the students to select the final classification and regression models for the challenge.
5. **Others:** The report may contain any additional analysis performed by the students that may be interesting from a practitioner point of view. Examples of such analysis may i) provide a thorough data analysis (for example, data visualization using unsupervised learning techniques); or account for the robustness, explainability or fairness considerations of the different models explored by the students.

3.1 Report grading

The report will be graded. There are four possible grades for the project report:

- [**0 (out of 10) points**] If a major methodological mistake (e.g., selecting the ML model on the data used to train it) is detected.
- [**5 points**] If a subset of the models introduced in the lectures and tutorials are correctly applied, evaluated and reported.
- [**7.5 points**] If a comprehensive application of the techniques covered in the lectures are correctly applied, evaluated and reported.
- [**10 points**] If the students go one step beyond the course material. They may, e.g., provide additional content in the report covering data analysis or robustness/fairness aspects (see point 5. above) and/or apply methodology that goes beyond what has been introduced in the lectures to train excellent regression and classification models.

4 Challenge

The ML project will be maintained in a challenge format similar to a Kaggle-like competition. This means you will see your performance and ranking in a leaderboard which will be updated at multiple time stamps during the semester. To begin with, we plan to update the leaderboard about once every week, and towards the end of the semester (i.e. after the main exam until the project deadline) we will update the leaderboard about once every 2 days. After each update of the leaderboard you will be able to see your model's performance on the public test set and your ranking in the whole competition. In the following part, we will deliver the details about the challenge.

Throughout the semester, you are encouraged to work with your training data, **train.csv**, to update your models, try different approaches, or perform better model selection and hyper-parameter tuning. Your participation in the challenge will be evaluated in two folds.

1. **Leaderboard:** For the first part of the challenge, we have a **leaderboard** in CMS which will be updated multiple times during the semester. Throughout the semester, you can submit your predictions on the test set, **test_public.csv**, in CMS. In this leaderboard, you will be able to see the performance evaluation of your team's model and its ranking among the models of other teams (for both prediction tasks). The idea of the leaderboard is twofold: i) give you a realistic estimate of the team raking for each of the tasks to incentivize healthy competition; and ii) get you familiar with the challenge evaluation process.
2. **Final Assessment:** For the second and final evaluation of the challenge, you are supposed to submit your predictions on the private test set, **test_private.csv**, which will be shared with you towards the end of the semester. You need to submit your predictions again through CMS. Your participation in the challenge will be assessed by your model's performance in this second test set alone, independent of the performance on the data from **test_public.csv**.

4.1 Performance evaluation

The teams will be ranked for the the classification and regression tasks based on the following metrics:

- For the regression task

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i^{true} - y_i^{pred})^2}. \tag{1}$$

- For the classification task

$$\text{(macro) Average f1-score} = \frac{1}{2}(F_1(Y=0) + F_1(Y=1)). \tag{2}$$

  Above, $F_1(Y=k)$ denotes the f1-score computed for class k.

Bonus points:
The top 5% of the teams, ranked according to the above metrics evaluated on the test_private.csv dataset for both tasks, will get a bonus (one extra point in the German grading system) on their final grade.

4.2 Submission

To submit your predictions for both the leaderboard and final submission, put your .npy files containing the predictions for both regression and classification tasks in the **same zip file**. Please name files inside the .zip as stated at the end of the provided Jupyter notebook.

5 Effect on overall course grade

The project is **fully voluntary** but it can help your grade, and did so for many students of previous iterations. Thus, it is highly recommended to do the project. To pass the course, you will have to pass either the main or re-exam. If you do not submit a project report, then your exam grade (i.e. the better grade of main exam and re-exam) will be your final grade. If your team submits the report, your overall course grade will be the better of the following two grades: your exam grade alone, or a 75%exam+25%report mixture grade. To put in in math terms:

$$\text{Course grade} = \max(\text{Exam grade};\ 0.75 * \text{Exam grade} + 0.25 * \text{project report grade})$$

All members of a team get an identical report grade. Note again that to participate in the project (and get a potential grade boost), your team must submit **both** the report and the predictions for the challenge.

Bonus points:
As noted at the end of section 4.1, the best-performing teams in the challenge will get a bonus, which increases your grade by 1 point in the German grade system. An example, taken from the previous iteration: if you have a grade of 2.0 in the exam, and a good report (10 out of 10 points), and you also do well enough in the challenge to get the bonus, your final course grade would be a 1.3.

6 Timeline

In the following, we detail the key dates that should not be missed if interested in joining the ML project:

- **Team registration:** via CMS due by 20.06.24. Teams of up to 3 students should be registered in CMS by then. Later changes in a team will only be possible under request.
- **Final project submission** via CMS due by 19.08.24. the students will need to submit both their report and final predictions (on `TWEETS_TEST_2`) by this date. We will not accept any late submissions.