DTSC 670: Foundations of Machine Learning

Final Project Student Instructions

Overview

In this final project, you will be given a dataset and will be asked to complete an end-to-end machine learning project similar to the **California Housing Prices** example. This project will demonstrate all the knowledge and skills that you have acquired throughout the course. You will have a lot of leeway in how you complete your final project as long as you follow the instructions and rubric in this document.

You will start by reviewing the data description below, read through the assignment instructions, and make sure that you understand the requirements in the project rubric. Next, you will choose between working on a regression or classification task making sure that the data preparation, machine learning algorithms, and metrics make sense for your type of task. As you work through your project, you will follow the **Machine Learning Project Checklist** (found in the appendix of your textbook) and structure your project into the following sections:

- Frame the Problem and Look at the Big Picture.
- Get the Data.
- Explore the data to gain insights.
- Prepare the data to better expose the underlying data patterns to machine learning algorithms.
- Explore many different models and shortlist the best ones.
- Fine-tune your models and combine them into a great solution.
- Present your solution (in your Jupyter notebook)

It is important that you carefully read and follow all instructions in this document. These instructions need to be followed closely to allow for an easier time to grade your project. Major points may be deducted for not following directions.

Please ensure that you take your time to work on this project as this represents a large portion of your grade. Your final grade will reflect the level of analysis that you perform on the data, the attention to detail when working on the data preparation steps and creation of your machine learning models, along with your ability to communicate your project appropriately.

Objectives

- Apply your acquired skills to create an end-to-end machine learning project
- Combine, clean, analyze, and prepare the data for machine learning models
- Clearly communicate the machine learning process and the ultimate results

Data

This dataset was downloaded from UC Irvine's machine learning repository and has been modified for our use in this assignment. It contains student performance data from two Portuguese schools and was originally collected from various school reports and questionnaires. Please make sure that you are using the files from Brightspace since this data has been adapted for our use.

Attributes

1. school - student's school (binary: "GP" - Gabriel Pereira or "MS" - Mousinho da Silveira)
2. sex - student's sex (binary: "F" - female or "M" - male)
3. age - student's age (numeric: from 15 to 22)
4. address - student's home address type (binary: "U" - urban or "R" - rural)
5. famsize - family size (binary: "LE3" - less or equal to 3 or "GT3" - greater than 3)
6. Pstatus - parent's cohabitation status (binary: "T" - living together or "A" - apart)
7. Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
8. Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
9. Mjob - mother's job (nominal: "teacher", "health" care related, civil "services" (e.g. administrative or police), "at_home" or "other")
10. Fjob - father's job (nominal: "teacher", "health" care related, civil "services" (e.g. administrative or police), "at_home" or "other")
11. reason - reason to choose this school (nominal: close to "home", school "reputation", "course" preference or "other")
12. guardian - student's guardian (nominal: "mother", "father" or "other")
13. traveltime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
14. studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
15. failures - number of past class failures (numeric: n if 1<=n<3, else 4)
16. schoolsup - extra educational support (binary: yes or no)
17. famsup - family educational support (binary: yes or no)
18. paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
19. activities - extra-curricular activities (binary: yes or no)
20. nursery - attended nursery school (binary: yes or no)
21. higher - wants to take higher education (binary: yes or no)

22. internet - Internet access at home (binary: yes or no)
23. romantic - with a romantic relationship (binary: yes or no)
24. famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
25. freetime - free time after school (numeric: from 1 - very low to 5 - very high)
26. goout - going out with friends (numeric: from 1 - very low to 5 - very high)
27. Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
28. Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
29. health - current health status (numeric: from 1 - very bad to 5 - very good)
30. absences_G1 - number of school absences for G1 term (numeric)
31. absences_G2 - number of school absences for G2 term (numeric)
32. absences_G3 - number of school absences for G3 term (numeric)

these grades are related with the course math subject

33. G1 - first term grade (numeric: from 0 to 20)
34. G2 - second term grade (numeric: from 0 to 20)
35. G3 - final grade (numeric: from 0 to 20, ← **this is your output target**)

Important notes

- You do not need to use all of these features in your final model. One important part of creating robust models is to analyze your data and determine the important features.
- G3 will be your target column but please read through the following note from UC Irvine's website: "*Important note: the target attribute G3 has a strong correlation with attributes G2 and G1. This occurs because G3 is the final year grade (issued at the 3rd period), while G1 and G2 correspond to the 1st and 2nd period grades. It is more difficult to predict G3 without G2 and G1, but such prediction is much more useful (see paper source for more details)*."
- You can choose to perform a regression task or classification task:
    - Regression task: predict the numeric final G3 grade
    - Classification task: predict whether a student failed the course or not based on their final G3 grade. According to this website, the Portugal grading system corresponds with the following US grading scale:

Portuguese Grade	US Grade Equivalent
18.00 - 20.00	A+
16.00 - 17.99	A
14.00 - 15.99	B
10.00 - 13.99	C
0.00 - 9.99	F

Final Project Instructions

Frame the Problem

Suppose that you work in the Advising Team for a large Portuguese school system, and your school director has asked you to analyze student data and attempt to create a machine learning model to predict a student's performance based on select features. Your director hopes to use this information to identify students who might need additional assistance and interventions to improve their grade in the course.

Present Your Findings

Your director has experience with machine learning and has asked to see your Jupyter notebook when you are finished with your project. However, she plans to also discuss the project with the school's board of directors who have limited technical knowledge. She has asked you to ensure that your notebook is understandable to someone with a limited machine learning background. Make sure that your notebook is well organized, includes easy to understand markdown comments throughout the notebook, along with good comments in your code for your teammates to be able to revise your analysis for other school classes in the future.

Required Elements of Your Project (75% of total grade)

Your project in your notebook must contain or use the following items. **Items marked with a red asterisk (*) is code that must be fully run and your output must be shown in the notebook before uploading to CodeGrade to get credit for the rubric item. Assignment submissions without output shown will receive a grade of 0 with the opportunity to resubmit one time. There will be an automatic grade reduction of 20 points on your second submission.**

Required Item	Percentage of Grade
Problem Framing & Big Picture	5%
Include an opening overview of your project that discusses the following. 1) Clearly communicate the problem and objective in business terms and how your solution will be used. 2) How should you frame this problem (supervised/unsupervised, online/offline, etc)? Briefly explain these terms since part of your audience is non-technical.	

3) Discuss the specific machine learning task that you are working on (regression/classification) and how it could solve the business problem. Briefly explain the difference since part of your audience is non-technical. 4) Identify the metrics that you will use to measure the model's performance. 5) Is there anything else that your director or board of directors need to know about this project?	
Get the Data	5%
1) Correctly import your data (CodeGrade will assume that your data is in the same folder as your notebook just like with your other assignments) *2) Check the size and type of data 3) List the available features and their data descriptions so that your director/board of directors can understand your work 4) Identify the target or label attribute 5) Correctly split your data into a training and test set	
Explore the Data	
*Thoroughly study the attributes and their characteristics	3%
*Produce at least four visualizations to assist in exploring the data. You should ensure that your visuals are informative and visually appealing, not purely using the default plots. (matplotlib and seaborn are the only packages available in CodeGrade)	4%
*Study the correlations between discrete and continuous numerical attributes	3%

Prepare the Data	
Based on your exploration of the data above, perform feature selection to narrow down your data. (While not required, we would suggest that you create a function or custom transformer to handle this step so that you can more easily transform your test data.)	4%
Create at least one data pipeline to handle the data preparation steps	5%
Fill in missing values or drop the rows or columns with missing values	2%
Create a custom transformer in your pipeline that: ● has a parameter that when equal to True, drops the G1 and G2 columns, and when False, leaves the columns in the data ● creates a new column in the data that sums the absences_G1, absences_G2, and absences_G3 data and then drops those three columns.	5%
Perform feature scaling on continuous numeric data	2%
Ordinal encode features that are either binary or that are ordinal in nature	1%
One-hot encode nominal or categorical data	1%
*Implement and use a Column Transformer to transform your numeric and categorical data	5%
*Correctly transform your training data using the above data preparation steps	5%

Shortlist Promising Models	
*1) Fit three or more promising models to your data *2) Use your custom transformer to see how all three (or more) of your models perform with both the G1 and G2 columns removed and with them remaining *3) Compare all three models both with and without the G1/G2 columns with cross validation	15%
Fine-Tune the System	
*Pick one model and use at least one grid search to fine-tune hyperparameters	3%
*Correctly transform your testing data using your data preparation pipeline(s)	4%
*Select your final model and measure its performance on the test set	3
Present Your Solution	See below
See the below additional rubric categories for the items related to presenting your solution to your executive team (in other words, presenting your work in this Jupyter notebook). Your project should include an overview and concluding section.	

Analytical Insight (10% of final grade)

A well-done machine learning project should analyze the data thoroughly and share the insights gleaned from the data. Your key findings should be communicated clearly, use beautiful visualizations, and easy-to-remember statements (e.g. “the median income is the number-one predictor of housing prices”) as described in the Machine Learning checklist from our textbook.

10%	5%	0%
Comprehensive and thorough data analysis conducted. Critical insights are clearly and effectively communicated through easy-to-remember statements. Visualizations are well-designed, beautiful, and highly informative, enhancing understanding. Findings are presented in a way that even non-technical stakeholders can grasp their significance.	Some level of data analysis attempted but lacks depth and thoroughness. Insights are present but may not be clearly stated or are not the most critical findings. Visualizations are used, but they could be more effective in conveying information. Default visuals were used without any attempt at making them more appealing or understandable.	No analysis of the data performed or only very superficial analysis. No insights shared or insights are vague, unclear, or inaccurate. Visualizations are missing, unclear, or irrelevant.

Communication (15% of final grade)

Effectively communicating your work, important insights, and solutions are a vital but frequently neglected aspect of being a Data Scientist. To ensure clarity and coherence in your work, consider the following:

- **Notebook Organization**: Your project documentation should be structured logically, making it easy for others to follow. Utilize markdown headers to separate sections and provide context (see next section).
- **Comments in Code**: Include clear and comprehensive comments within your code. These comments should not only explain what the code is doing but also why it's being done in that way.
- **Overview Section**: Start your project with an overview section. Here, you briefly introduce the problem, the data, and your approach. It sets the stage for what follows.
- **Concluding Section**: At the end of your project, create a concluding section. In this part, restate the big picture of your work, emphasizing its significance. Explain whether your solution aligns with the business objectives and provide reasons for your assessment based on your work.

- **Evaluation of What Worked and What Didn't**: Reflect on your project. Describe what aspects of your project were successful and why. Also, acknowledge and analyze what didn't work as expected, along with potential reasons. What are potential next steps based on your findings?

The ability to present your work in a clear, organized, and understandable manner is crucial for a Data Scientist. It ensures that your findings and solutions are not only valuable but accessible to stakeholders and peers.

15%	10%	5%	0%
Exceptional organization with well-structured markdown headers, making it easy to navigate. Comprehensive and clear comments throughout the code, offering insights and justifications. A thorough overview section that sets the context effectively. An insightful concluding section that eloquently restates the big picture, demonstrates the alignment of the solution with business objectives, and offers a detailed analysis of what worked and what didn't. The next steps based on your findings were clearly outlined. .	Decent organization with clear markdown headers separating sections. Reasonable comments in the code, providing insights into the process. An informative overview section that introduces the problem and approach. A concluding section that effectively restates the big picture and aligns the solution with business objectives. Some analysis of what worked and what didn't. Basic next steps were mentioned based on your findings.	Some attempt at organization with limited use of markdown headers. Basic comments in the code, explaining some parts of the process. An overview section is present but may lack depth or clarity. A rudimentary concluding section that touches upon the big picture and business objectives but lacks analysis of what worked or didn't work. Next steps were lacking or not included.	The work lacks organization and structure; there are no markdown headers or sections. Minimal or no comments in the code, making it difficult to understand. Absence of an overview and concluding section. No reflection on the alignment of the solution with business objectives, and no analysis of what worked or didn't work. There were no next steps based on your findings.

Markdown Headers (required)

Assignment submissions without these exact markdown headers will receive a grade of 0 with the opportunity to resubmit one time. There will be an automatic grade reduction of 20 points on your second submission. This is an essential piece of the assignment to make grading fair and easier.

- Each respective section from the machine learning checklist (the bullet points from the **Overview** section of these instructions) should be clearly separated with the use of Heading 2 (##) markdown headers.
- In addition, each rubric item from the **Required Elements of Your Project** should be marked with Heading 3 (###) markdown headers for the subsections. We need these elements to be easily graded, so without these headers, an element will not be graded even if it is in your notebook.
- Take a look at this documentation for how to use Markdown comments.

Watch the **Final Project Assignment Walkthrough** video in Brightspace for more details and an example of how your headers should be set up.

CodeGrade & PDF Submission

In order to get full credit for the assignment, you need to submit the following:

- the .ipynb file for your project named **final_project.ipynb** that runs without errors
- any code blocks that generate output must be fully executed with the output shown within the notebook prior to submission. Otherwise, we will not see your output, and you will not get credit for the rubric item!
- a PDF of your project that will be run through Turnitin plagiarism detection

CodeGrade will attempt to run your submission to verify that your code works and there are no errors in the file. Any CodeGrade submission with an error will have an automatic deduction of 20 points. There are two CodeGrade submission links in the **Final Project** folder on Brightspace.

The first submission link is called **Final Project Practice Submission**. You will have unlimited submission attempts through this link to check for any errors in CodeGrade. We will not check these submissions and these will not be graded.

The second submission link is called **Final Project Notebook Submission**. You will have only one submission attempt allowed for your actual project. These submissions will be manually graded according to the rubric items in this document. Please remember that only rubric items that employ the appropriate Markdown headers and display code output will be considered for grading.

You will submit a PDF of your project that will be run through Turnitin through the **Final Project PDF Submission** link. Assignments without matching PDF submissions will not be graded.

Miscellaneous

Additional points can be taken off at the discretion of the grader if the following details are not followed.

- Your file should be named **final_project.ipynb**
- Upload PDF of your entire project named **DTSC670_FinalProject_<Your Name>.pdf**
- Your notebook should be organized and easy to follow.
- Your work should be free of spelling and grammar errors.
- Your work should reflect that a meaningful amount of work went into it. It is fairly obvious those that take their time with this project and those that pull it together last minute.
- Late submissions will not be accepted unless you have an approved Incomplete request that was requested and approved **BEFORE** the term ends. We highly recommend that you avoid waiting until the final few days of the class to submit your project, as "technical issues" will not be accepted as a valid reason for a late submission.
- Points can be deducted for other items not included in this document for any items that make grading more difficult or for projects that do not follow directions.

Plagiarism

While you may look online for inspiration for your project, visualizations, and analytical analysis concepts on sites like Kaggle, **everything in your project must be your own**. You may not directly copy ideas from online sources, nor can you collaborate in any way with classmates or other students. If plagiarism or collaboration is found, you will receive a zero grade for the assignment. You may be dismissed from the MSDS program.

Students may not post online or share in any way solutions or answers to any assignments. This includes posting solutions to public GitHub repositories or other similar sites. If students want to build a portfolio, please create a private repository to share as needed. Penalties for violation of this policy can range from failing the class to dismissal from the program.

Next Steps

Once you complete all the steps above, you will:

1. Upload your **final_project.ipynb** to the **Final Project Practice Submission** link in Brightspace to check for errors.
2. Once you are sure that your notebook will run without errors, upload your final submission to the **Final Project Notebook Submission** link.
3. Finally, upload a PDF copy of your complete project to the **Final Project PDF Submission** link.
4. **Important: Your submission will only be graded if both the PDF and the .ipynb notebook are submitted, you have clear markdown headers for rubric items, and your code is fully rendered and your output is shown before uploading to CodeGrade.**