carlearn / dsc-mod-3-project-v2-1 Public

forked from learn-co-curriculum/dsc-mod-3-project-v2-1

0 stars 17 forks

Star Watch

Code Pull requests Actions Projects Wiki Security Insights

master

This branch is 12 commits ahead of learn-co-curriculum:master. Contribute Fetch upstream

carlearn Delete Jupyter Notebook_Mod3_Project.pdf ... 2 minutes ago 18

View code

README.md

Module 3 Final Project

Introduction

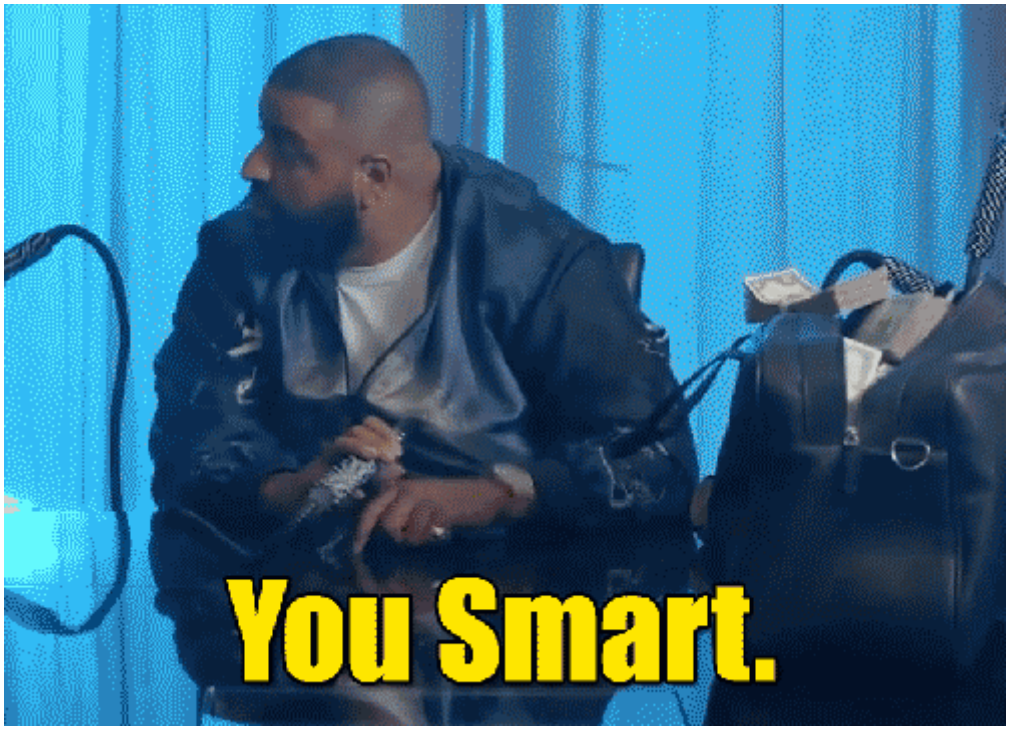
In this lesson, we'll review all the guidelines and specifications for the final project for Module 3.

Objectives

- Understand all required aspects of the Final Project for Module 3
- Understand all required deliverables
- Understand what constitutes a successful project

Final Project Summary

Congratulations! You've made it through another *intense* module, and now you're ready to show off your newfound Machine Learning skills!



All that remains for Module 3 is to complete the final project!

The Project

The main goal of this project is to create a classification model. For this project you have the choice to either:

- choose a data set from a curated list
- choose your own data set *outside* of the curated list.

The data guidelines for either option are shown below

For this project, you're going to select a dataset of your choosing and create a classification model. You'll start by identifying a problem you can solve with classification, and then identify a dataset. You'll then use everything you've learned about Data Science and Machine Learning thus far to source a dataset, preprocess and explore it, and then build and interpret a classification model that answers your chosen question.

a. Choosing the data from a curated list

You are allowed to select one of the four data sets described below. Each comes with its own advantages and disadvantages, and, of course, its own associated business problem and stakeholders. It may be desirable to flesh out your understanding of the audience or the business proposition a little more than sketched out here. If you select one of these four data sets, you **need no further approval from your instructor**.

1. Chicago Car Crash Data. Note this links also to Vehicle Data and to Driver/Passenger Data.

Build a classifier to predict the primary contributory cause of a car accident, given information about the car, the people in the car, the road conditions etc. You might imagine your audience as a Vehicle Safety Board who's interested in reducing traffic accidents, or as the City of Chicago who's interested in becoming aware of any interesting patterns. Note that there is a **multi-class** classification problem. You will almost certainly want to bin or trim or otherwise limit the number of target categories on which you ultimately predict. Note e.g. that some primary contributory causes have very few samples.

2. Terry Stops Data. In *Terry v. Ohio*, a landmark Supreme Court case in 1967-8, the court found that a police officer was not in violation of the "unreasonable search and seizure" clause of the Fourth Amendment, even though he stopped and frisked a couple of suspects only because their behavior was suspicious. Thus was born the notion of "reasonable suspicion", according to which an agent of the police may e.g. temporarily detain a person, even in the absence of clearer evidence that would be required for full-blown arrests etc. Terry Stops are stops made of suspicious drivers.

Build a classifier to predict whether an arrest was made after a Terry Stop, given information about the presence of weapons, the time of day of the call, etc. Note that this is a **binary** classification problem.

Note that this dataset also includes information about gender and race. You **may** use this data as well. You may, e.g. pitch your project as an inquiry into whether race (of officer or of subject) plays a role in whether or not an arrest is made.

If you **do** elect to make use of race or gender data, be aware that this can make your project a highly sensitive one; your discretion will be important, as well as your transparency about how you use the data and the ethical issues surrounding it.

3. Customer Churn Data

Build a classifier to predict whether a customer will ("soon") stop doing business with SyriaTel, a telecommunications company. Note that this is a **binary** classification problem.

Most naturally, your audience here would be the telecom business itself, interested in losing money on customers who don't stick around very long. Are there any predictable patterns here?

4. Tanzanian Water Well Data (*active competition*!) Tanzania, as a developing country, struggles with providing clean water to its population of over 57,000,000. There are many waterpoints already established in the country, but some are in need of repair while others have failed altogether.

Build a classifier to predict the condition of a water well, using information about the sort of pump, when it was installed, etc. Note that this is a **ternary** classification problem.

b. Selecting a Data Set *Outside* of the Curated List

We encourage you to be very thoughtful when identifying your problem and selecting your data set--an overscoped project goal or a poor data set can quickly bring an otherwise promising project to a grinding halt. **If you are going to choose your own data set, you'll need to run it by your instructor for approval**.

To help you select an appropriate data set for this project, we've set some guidelines:

1. Your dataset should work for classification. The classification task can be either binary or multiclass, as long as it's a classification model.

2. Your dataset needs to be of sufficient complexity. Try to avoid picking an overly simple dataset. Try to avoid extremely small datasets, as well as the most common datasets like titanic, iris, MNIST, etc. We want to see all the steps of the Data Science Process in this project--it's okay if the dataset is mostly clean, but we expect to see some preprocessing and exploration. See the following section, ***Data Set Constraints***, for more information on this.

3. On the other end of the spectrum, don't pick a problem that's too complex, either. Stick to problems that you have a clear idea of how you can use machine learning to solve it. For now, we recommend you stay away from overly complex problems in the domains of Natural Language Processing or Computer Vision--although those domains make use of Supervised Learning, they come with a lot of other special requirements and techniques that you don't know yet (but you'll learn soon!). If you're chosen problem feels like you've overscoped, then it probably is. If you aren't sure if your problem scope is appropriate, double check with your instructor!

Data Set Constraints

When selecting a data set, be sure to take into consideration the following constraints:

1. Your data set can't be one we've already worked with in any labs.
2. Your data set should contain a minimum of 1000 rows.
3. Your data set should contain a minimum of 10 predictor columns, before any one-hot encoding is performed.
4. Your instructor must provide final approval on your data set.

Problem First, or Data First?

There are two ways that you can about getting started: ***Problem-First*** or ***Data-First***.

Problem-First: Start with a problem that you want to solve with classification, and then try to find the data you need to solve it. If you can't find any data to solve your problem, then you should pick another problem.

Data-First: Take a look at some of the most popular internet repositories of cool data sets we've listed below. If you find a data set that's particularly interesting for you, then it's totally okay to build your problem around that data set.

There are plenty of amazing places that you can get your data from. We recommend you start looking at data sets in some of these resources first:

- UCI Machine Learning Datasets Repository
- Kaggle Datasets
- Awesome Datasets Repo on Github
- New York City Open Data Portal
- Inside AirBNB

The Deliverables

For online students, your completed project should contain the following four deliverables:

1. A ***Jupyter Notebook*** containing any code you've written for this project. This work will need to be pushed to a public GitHub repository dedicated for this project.

2. An organized **README.md** file in the GitHub repository that describes the contents of the repository. This file should be the source of information for navigating through the repository.

3. A ***Blog Post***.

4. An ***"Executive Summary" PowerPoint Presentation*** that gives a brief overview of your problem/dataset, and each step of the OSEMN process.

Note: On-campus students may have different deliverables, please speak with your instructor.

Jupyter Notebook Must-Haves

For this project, your Jupyter Notebook should meet the following specifications:

Organization/Code Cleanliness

- The notebook should be well organized, easy to follow, and code is commented where appropriate.
    - Level Up: The notebook contains well-formatted, professional looking markdown cells explaining any substantial code. All functions have docstrings that act as professional-quality documentation.
- The notebook is written to technical audiences with a way to both understand your approach and reproduce your results. The target audience for this deliverable is other data scientists looking to validate your findings.

Process, Methodology, and Findings

- Your notebook should contain a clear record of your process and methodology for exploring and preprocessing your data, building and tuning a model, and interpreting your results.
- We recommend you use the OSEMN process to help organize your thoughts and stay on track.

Blog Post Must-Haves

Refer back to the Blogging Guidelines for the technical requirements and blog ideas.

The Process

These steps are informed by Smart Vision's[1] description of the CRISP-DM process.

1. Business Understanding

Start by reading this document, and making sure that you understand the kinds of questions being asked. In order to narrow your focus, you will likely want to make some design choices about your specific audience, rather than addressing all of the "many people" mentioned in the background section. Do you want to emphasize affordability, investment, or something else? This framing will help you choose which stakeholder claims to address.

Three things to be sure you establish during this phase are:

1. **Objectives:** what questions are you trying to answer, and for whom?
2. **Project plan:** you may want to establish more formal project management practices, such as daily stand-ups or using a Trello board, to plan the time you have remaining. Regardless you should determine the division of labor, communication expectations, and timeline.
3. **Success criteria:** what does a successful project look like? How will you know when you have achieved it?

2. Data Understanding

Write a script to download the data (or instructions for future users on how to manually download it), and explore it. Do you understand what the columns mean? How do the three data tables relate to each other? How will you select the subset of relevant data? What kind of data cleaning is required?

It may be useful to generate visualizations of the data during this phase.

3. Data Preparation

Through SQL and Pandas, perform any necessary data cleaning and develop a query that pulls in all relevant data for analysis in a linear regression model, including any merging of tables. Be sure to document any data that you choose to drop or otherwise exclude. This is also the phase to consider any feature scaling or one-hot encoding required to feed the data into a classification model.

4. Modeling

The focus this time is on prediction. Good prediction is a matter of the model generalizing well. Steps we can take to assure good generalization include: testing the model on unseen data, cross-validation, and regularization. What sort of model should you build? A diverse portfolio is probably best. Classification models we've looked at so far include logistic regression, decision trees, bagging, and boosting, each of these with different flavors. You are encouraged to try any or all of these.

5. Evaluation

Recall that there are many different metrics we might use for evaluating a classification model. Accuracy is intuitive, but can be misleading, especially if you have class imbalances in your target. Perhaps, depending on you're defining things, it is more important to minimize false positives, or false negatives. It might therefore be more appropriate to focus on precision or recall. You might also calculate the AUC-ROC to measure your model's *discrimination*.

6. Deployment

In this case, your "deployment" comes in the form of the deliverables listed above. Make sure you can answer the following questions about your process:

- "How did you pick the question(s) that you did?"
- "Why are these questions important from a business perspective?"
- "How did you decide on the data cleaning options you performed?"
- "Why did you choose a given method or library?"
- "Why did you select those visualizations and what did you learn from each of them?"
- "Why did you pick those features as predictors?"
- "How would you interpret the results?"
- "How confident are you in the predictive quality of the results?"
- "What are some of the things that could cause the results to be wrong?"

Grading Rubric

Online students can find a PDF of the grading rubric for the project here. *Note: On-campus students may have different requirements, please speak with your instructor.*

Citation

1. "What is the CRISP-DM Methodology?" Smart Vision Europe. Available at: https://www.sv-europe.com/crisp-dm-methodology/

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

- **Jupyter Notebook** 100.0%