Kenytt Avery, Computer Science Department, California State University, Fullerton

CPSC 483 - Introduction to Machine Learning

Project 4, Fall 2021

due Friday, December 17 at 9:45 pm PST

Last updated Monday November 29, 5:05 pm PST

In this project you will work with some toy datasets to compare the decision boundaries determined by several classifier algorithms.

The project may be completed individually, or in a team of up to four students as long as all students are enrolled in the same section of the course.

Platforms

The platform requirements for this project are the same as for Project 1.

Libraries and Code

In addition to NumPy, pyplot, Pandas, and scikit-learn, you may wish to use Seaborn to draw more attractive graphs when visualizing data. You may not use any other library except the Python Standard Library.

Code from the Python documentation, *A Whirlwind Tour of Python*, the Jupyter notebooks accompanying the textbook, the library documentation, and the tutorial article referenced below may be reused. All other code and the results of experiments must be your own original work or the original work of other members of your team.

Dataset

Download `dataset1.csv`, `dataset2.csv`, and `dataset3.csv`. These datasets have three columns: features x_1 and x_2, and label $t \in \{0, 1\}$.

Experiments

Begin by adding a Markdown cell introducing your project. Be sure to include the names of the members of your team, the semester, section, and project number.

Run the following experiments in a Jupyter notebook, running Python code in code cells and describing your results in Markdown cells. Be sure to mark clearly in the notebook where each experiment begins.

1. Load and examine each dataset.
2. Fit and score binary classifiers for dataset 1 using logistic regression, Gaussian naive Bayes, and K-Nearest Neighbors for dataset 1. How accurate are the model's predictions?

   Note: since these are toy datasets, and we are interested in the behavior of the classifiers themselves rather than making predictions, do not split the data into training and test sets. Train with the entire dataset.
3. Repeat experiment *(2)* for dataset 2. How do your results compare?
4. Create scatterplots for datasets 1 and 2, plotting points from class 0 with a different color and marker from points in class 1. What accounts for the discrepancies between experiments *(2)* and *(3)*?
5. Repeat experiments *(3)* and *(4)* for dataset 3. What do you observe about the differences in behavior between the classifiers?
6. Use the code from KV Subbaiah Setty's tutorial How To Plot A Decision Boundary For Machine Learning Algorithms in Python as a guide, plot the decision boundaries for each classifier and dataset. Does this help to explain the behavior you saw in experiment *(5)*?
7. Repeat experiments *(3)*, *(4)*, and *(6)*, this time using Support Vector Machines with linear, polynomial, radial basis function, and sigmoid kernels. Which kernels do the best and worst and finding appropriate decision boundaries?
8. Since dataset 3 seems the hardest to fit, let's focus on that. If you look closely at the documentation, you'll see that the polynomial kernel uses a default degree of 3, but it seems as if it ought to be possible to fit this dataset better.

   Use `sklearn.model_selection.GridSearchCV` to try polynomial kernels of degree up to 10. Which leads to the best performance?
9. Now plot the decision boundary of the polynomial kernel you found in experiment *(8)*, and compare it to the decision boundary you found in experiment *(7)*.

Submission

As described above, the first cell in your notebook should include the names of the members of your team, the semester, section, and project number. Only one submission is required.

Since you may be actively editing and making changes to the code cells in your notebook, be certain that each of your code cells still runs correctly before submission by selecting *Run All* from the drop-down menu bar.

Submit your Jupyter `.ipynb` notebook file through Canvas before 9:45 pm PST on the due date.

The Canvas submission deadline includes a grace period of an hour. Canvas will mark submissions after the first submission deadline as late, but your grade will not be penalized. If you miss the second deadline, you will not be able to submit and will not receive credit for the project.

Note: do not attempt to submit projects via email. Projects must be submitted via Canvas, and instructors cannot submit projects on students' behalf.

Note: In order to submit a project as a team, you must join a group for that project in Canvas.

- Teams are specific to the project, so you must join a new group even if you worked with the same team on a previous project.
- Teams must be created by the instructor; you cannot submit projects using a "Student Group."
- Several groups have been pre-created by the instructor, and your team may join one of those for your submission. The first team member to join will automatically be assigned as the group leader.
- If there are no groups available, email the instructor immediately to request a new group to be created. Do not wait until the due date.

See the following sections of the Canvas documentation for instructions on group submission:

- How do I join a group as a student?
- How do I submit an assignment on behalf of a group?

Grading

The grade for the project will be assigned on the following five-point scale:

Exemplary (5 points)

Results are correct and clearly presented; explanatory text clearly and concisely tells the story with appropriate context and analysis; organization makes it easy to review.

Basically Correct (4 points)

The analysis comes to correct (or defensible) results and conclusions, but the presentation is not easy to follow and/or portions are not clear or lack context.

Right Idea (3 points)

The approach is appropriate, but the work has mistakes in code, analysis, or presentation that undermine the correctness of conclusions.

Solid Start (2 points)

The work makes a good start, but has fundamental conceptual problems in code, analysis, or presentation such that it will not produce legitimate results.

Did Something (1 point)

The solution began an attempt, but is either insufficiently complete to assess correctness or is on entirely the wrong track.

Did Nothing (0 points)

Project was not submitted, submitted code belonging to someone other than the members of the team, or submission was of such low quality that there is nothing to assess.

Acknowledgements: this grading scale is drawn from the general rubric used by Professor Michael Ekstrand at Boise State University.