CSCD01 Assignment 2

Group cswasamistake

2021-03-06

Bug #1: “Regression in GP standard deviation where y_train.std() == 0”

(Source: https://github.com/scikit-learn/scikit-learn/issues/18318)

Overview:

The first bug we have selected to fix is an issue regarding an unexpected error that occurs in the Gaussian Process module, when creating a GaussianProcessRegressor object with the parameter normalize_y set to True. The parameter normalize_y indicates that the dataset provided by the user will be normalized. Inspecting the code shows that the formula used to do this is done by subtracting by the mean of the dataset, and then dividing by the standard deviation of the dataset. The problem arises when the standard deviation of the dataset is zero; there is a division by zero error. This occurs when there is only one point in the dataset, or when all points in the dataset are the same.

Solution:

The solution implemented will check for the edge cases, and then raise a clear and concise error. Note that we considered an alternate solution of adding an insignificant value to the standard deviation to avoid division by zero. However, this hides the error from the user, changing the expected behaviour of the normalization done by the module. We have also considered simply automatically setting normalize_y to False, and continuing without normalizing the dataset. This also hides the error from the user, and so we decided to raise an appropriate error to form the user instead.

Changes:

In the file gaussian_process/_gpr.py, approximately line 203; a case is added to check for when standard deviation is equal to zero.

Testing:

The test cases are implemented within the existing test suite of the library, under the gaussian_process module.

In the file gaussian_process/test_gpr.py, approximately line 276; two test cases were added under the function named test_no_standard_deviation. The first test case constructs a NumPy array with only one data point. The second test case constructs a NumPy array with three identical data points. The two test cases together exhaust the possibilities of when there is only one datapoint, and when there are multiple datapoints (and all of them are identical).

Bug #2: “Pipeline requires both fit and transform method to be available instead of only fit_transform”

(Source: https://github.com/scikit-learn/scikit-learn/issues/16710)

Overview:

The second bug we decided to fix is located in the pipeline module, where creating a pipeline with a non-parametric function caused an unexpected type error to be thrown. A non-parametric function, an example being TSNE, is a function whose arguments can infinitely grow with the sample size. It is trivial to see why pipelines created with non-parametric functions do not have a dedicated transform method. The pipeline module requires both fit and transform methods to be available to create a pipeline, instead of only fit_transform, which is the only method available when creating a pipeline with a

non-parametric function. As a result, when making a pipeline with a non-parametric function such as TSNE, the condition is not satisfied, thus an error is thrown.

Solution:

The solution implemented was to modify the if condition located on line 166 in the pipeline.py file. A truth table for the non-modified condition can be represented as such:

fit	fit_transform	transform	¬(¬(fit ∨ fit_transform) ∨ ¬transform)
F	F	F	**F**
F	F	T	**F**
F	T	F	**F**
F	T	T	**T**
T	F	F	**F**
T	F	T	**T**
T	T	F	**F**
T	T	T	**T**

(Figure 1)

One can see on line 3 that the truth assignments with fit = false, transform = false and fit_transform = true, evaluates to false, which causes the error to be thrown in line 167-168.

We modified the if condition to account for the aforementioned case, as well as the cases where fit_transform = true. These cases can be seen in the truth table below:

fit	fit_transform	transform	(fit_transform ∨ (fit ∧ transform))
F	F	F	**F**
F	F	T	**F**
F	T	F	**T**
F	T	T	**T**
T	F	F	**F**
T	F	T	**T**
T	T	F	**T**
T	T	T	**T**

(Figure 2)

Changes:

In the file sklearn/pipeline.py, approximately line 166; the if condition is modified to reflect the behaviour represented in figure 2.

Testing:

The test cases are implemented within the existing test suite of the library, under the manifold module.

In the file sklearn/manifold/tests/test_t_sne.py, approximately line 122; a test case was added under the function named test_TSNE_pipeline. The test case constructs a degenerate pipeline with a non-parametric function (TSNE) and runs fit_transform with some training data. This test case ensures non-parametric functions can successfully create a pipeline.

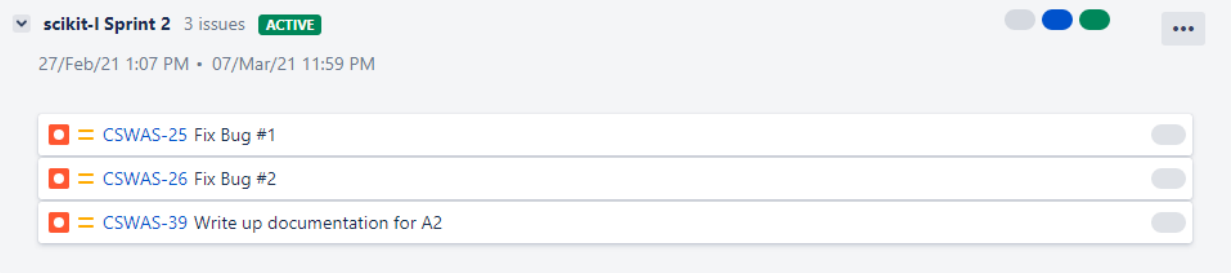
Use of tools:

Github:

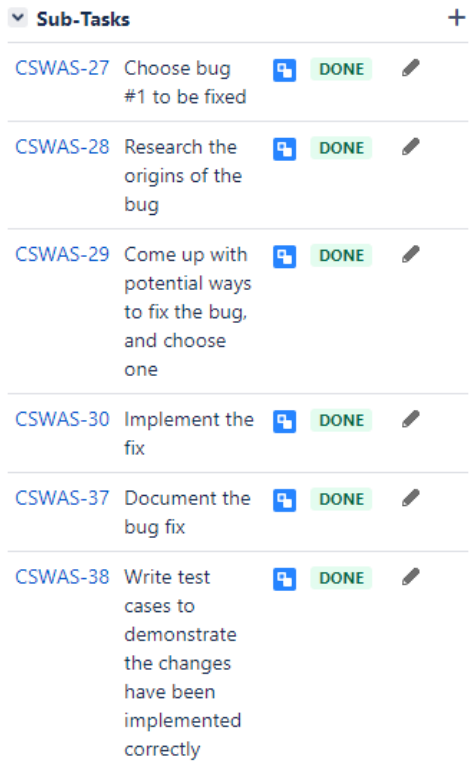
Each assignment is under the correct folder, and one copy of the scikit-learn library is cloned. The changes are all made in the cloned library, and each bug fix is done on a separate branch before being merged with master.

Jira:

We have planned our workflow of assignment 2 as the second sprint of our project. Each bug fix is shown as an issue:



Also, each issue (bug fix) has six sub-tasks assigned to it:



The bug fixes do not have assignees on Jira. This is because Jira does not allow multiple people to work on the same issue. As such, we have assigned groups of three to work on each bug through Discord and a shared Google docs file:

A2 Part 2

Group 1:
Anthony
Jeremy
Carlos

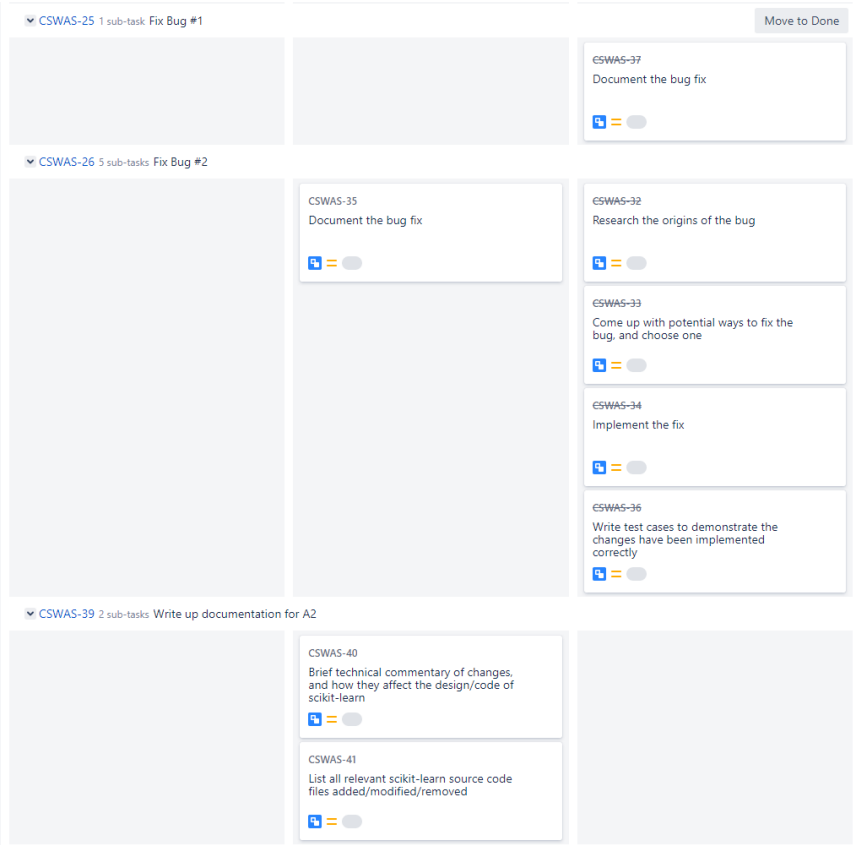
Bug:
Regression in GP standard deviation where y_train.std() == 0
https://github.com/scikit-learn/scikit-learn/issues/18318

Bug fix:
gaussian_process/_gpr.py line 203; added test for when std=0

Tes cases:
gaussian_process/tests/test_gpr.py line 276; added test case for when sample size = 1, or when sample size > 1 but sample points are all the same.

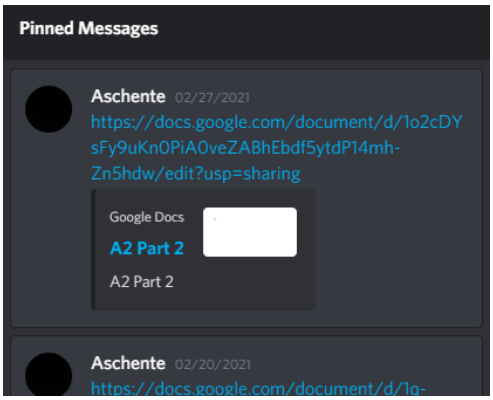
Group 2:
Evan
Ryan
Hunter

As of the creation of this document, our workflow progress is as follows:

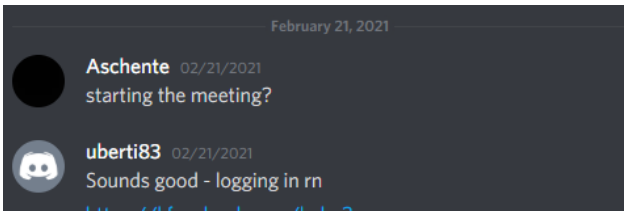


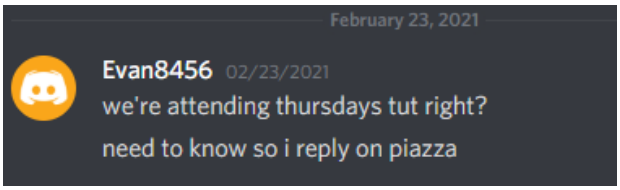
Discord:

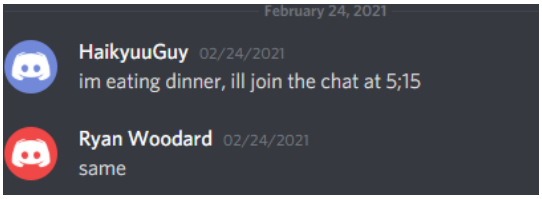
We use pins to link to shared Google docs files for referral:

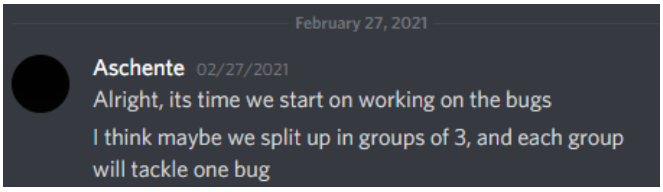


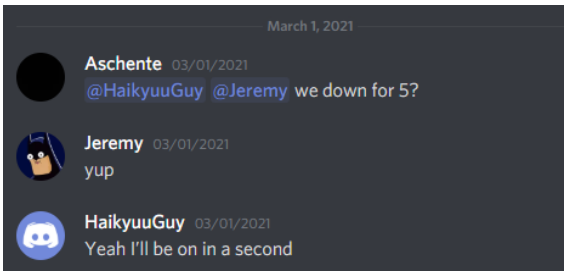
Our Discord server is the main means of communication, and previous chat logs show that we meet up every 2-3 days. Below are snippets of our chat logs.

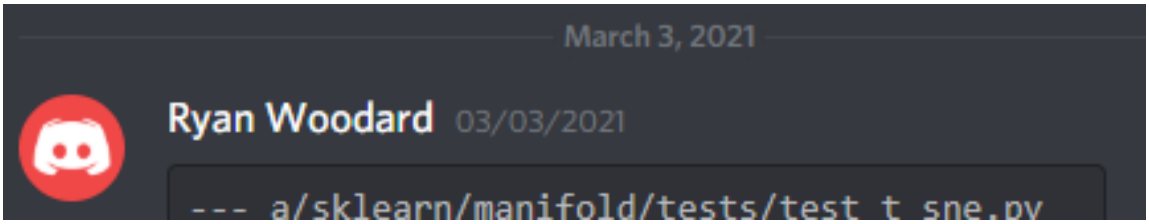









March 1, 2021
Aschente 03/01/2021
@HaikyuuGuy @Jeremy we down for 5?
Jeremy 03/01/2021
yup
HaikyuuGuy 03/01/2021
Yeah I'll be on in a second


March 3, 2021
Ryan Woodard 03/03/2021
--- a/sklearn/manifold/tests/test_t_sne.py