

Using Python Jupyter Notebooks + nbgrader as a Homework System

EAMS 2022

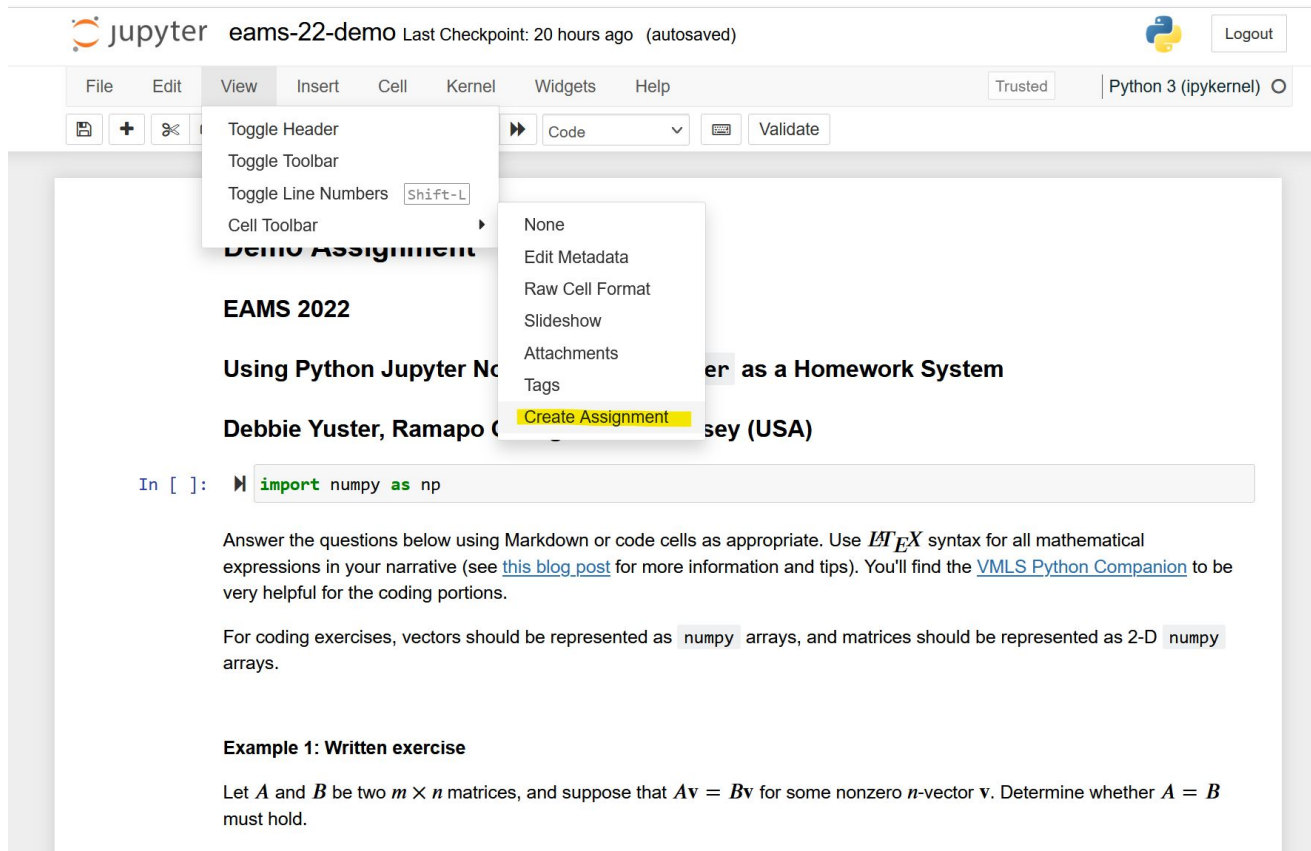
Debbie Yuster, Ph.D. - dyuster@ramapo.edu
Assistant Professor of Data Science and Mathematics
Ramapo College of New Jersey (USA)

Slides: bit.ly/yuster-eams-22

Introduction

- Applied Linear Algebra course for Masters of Science in Data Science students
- Homework assigned entirely within Jupyter notebooks (Python kernel)
- Mix of written, computational, and coding problems

Creating an assignment



The screenshot shows the JupyterLab interface for a notebook named 'eams-22-demo'. The top bar includes the Jupyter logo, the notebook name, and a 'Logout' button. The main menu bar contains 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. The 'View' menu is open, showing options like 'Toggle Header', 'Toggle Toolbar', 'Toggle Line Numbers', and 'Cell Toolbar'. A sub-menu is open under 'Cell Toolbar', listing options such as 'None', 'Edit Metadata', 'Raw Cell Format', 'Slideshow', 'Attachments', 'Tags', and 'Create Assignment', which is highlighted in yellow. The notebook content includes a title 'EAMS 2022', a subtitle 'Using Python Jupyter Notebook as a Homework System', and an author 'Debbie Yuster, Ramapo College of New Jersey (USA)'. Below the title is a code cell with the text 'In []: import numpy as np'. The main text of the notebook asks the user to answer questions using Markdown or code cells, and provides instructions on representing vectors and matrices using numpy arrays.

Jupyter eams-22-demo Last Checkpoint: 20 hours ago (autosaved) Python 3 (ipykernel)

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Toggle Header
Toggle Toolbar
Toggle Line Numbers Shift-L
Cell Toolbar

None
Edit Metadata
Raw Cell Format
Slideshow
Attachments
Tags
Create Assignment

EAMS 2022

Using Python Jupyter Notebook as a Homework System

Debbie Yuster, Ramapo College of New Jersey (USA)

```
In [ ]: import numpy as np
```

Answer the questions below using Markdown or code cells as appropriate. Use $LaTeX$ syntax for all mathematical expressions in your narrative (see [this blog post](#) for more information and tips). You'll find the [VMLS Python Companion](#) to be very helpful for the coding portions.

For coding exercises, vectors should be represented as `numpy` arrays, and matrices should be represented as 2-D `numpy` arrays.

Example 1: Written exercise

Let A and B be two $m \times n$ matrices, and suppose that $A\mathbf{v} = B\mathbf{v}$ for some nonzero n -vector \mathbf{v} . Determine whether $A = B$ must hold.



ID: cell-4c5be924e56fb862

Read-only



Example 3: Multiple choice exercise

When the Gram-Schmidt algorithm is run on a particular list of 10 15-vectors, it terminates in iteration 5 (since $\tilde{q}_5 = 0$). Which of the following must be true? Assign the variable `ans3` to either "a", "b", or "c", according to your answer choice.

- (a) a_2, a_3, a_4 are linearly independent.
- (b) a_1, a_2, a_5 are linearly dependent.
- (c) a_1, a_2, a_3, a_4, a_5 are linearly dependent.

In []: ▶

-

-

Manually graded answer

Manually graded task

Autograded answer

Autograder tests

Read-only

In []: ▶



ID: cell-4c5be924e56fb862

Read-only



Example 3: Multiple choice exercise

When the Gram-Schmidt algorithm is run on a particular list of 10 15-vectors, it terminates in iteration 5 (since $\tilde{q}_5 = 0$). Which of the following must be true? Assign the variable `ans3` to either "a", "b", or "c", according to your answer choice.

- (a) a_2, a_3, a_4 are linearly independent.
- (b) a_1, a_2, a_5 are linearly dependent.
- (c) a_1, a_2, a_3, a_4, a_5 are linearly dependent.

In []: ▶

ID: cell-7327ce4e63e7422f

Autograded answer



In []: ▶

-

-

Manually graded answer

Manually graded task

Autograded answer

Autograder tests

Read-only



ID: cell-4c5be924e56fb862

Read-only



Example 3: Multiple choice exercise

When the Gram-Schmidt algorithm is run on a particular list of 10 15-vectors, it terminates in iteration 5 (since $\tilde{q}_5 = 0$). Which of the following must be true? Assign the variable `ans3` to either "a", "b", or "c", according to your answer choice.

- (a) a_2, a_3, a_4 are linearly independent.
- (b) a_1, a_2, a_5 are linearly dependent.
- (c) a_1, a_2, a_3, a_4, a_5 are linearly dependent.

In []:



ID: cell-7327ce4e63e7422f

Autograded answer



```
### BEGIN SOLUTION
ans3 = "c"
### END SOLUTION
```

In []:



Points: 1

ID: cell-c39a3a88ac744e27

Autograder tests



```
# This test checks that your answer has the right format
# It does not check for correctness
assert ans3 == "a" or ans3 == "b" or ans3 == "c" or ans3 == "d"

### BEGIN HIDDEN TESTS
assert ans3 == "c"
### END HIDDEN TESTS
```



ID: cell-916b50345035d32c

Read-only



Example 4: Autograded answer (numeric vector, unique solution)

Find the vector \mathbf{a} such that $\mathbf{a}^T \mathbf{v}$ extracts (is equal to) the 5th entry of the 10-vector \mathbf{v} . Store the vector \mathbf{a} in the variable `a`.

In []:



ID: cell-b4699b230f177703

Autograded answer



```
### BEGIN SOLUTION
a = np.array([0, 0, 0, 0, 1, 0, 0, 0, 0, 0])
### END SOLUTION
```

In []:



Points:

1

ID: cell-293281c68751a39c

Autograder tests



```
assert isinstance(a, np.ndarray)
assert len(a) == 10

### BEGIN HIDDEN TESTS
np.testing.assert_allclose(a, np.array([0, 0, 0, 0, 1, 0, 0, 0, 0, 0]))
### END HIDDEN TESTS
```



ID: cell-3148db653f7a660f

Read-only



Example 5: Autograded answer (numeric matrix, multiple solutions)

Find a nonzero 2×2 matrix B such that $B^2 = 0$. Store the matrix B in the variable `B`.

In []: ▶

ID: cell-3f08658a77c24558

Autograded answer



(Writing the solution is optional)

In []: ▶



Points: 1

ID: cell-12efc9e5a9cef395

Autograder tests



```
# In this case I've chosen to expose all the tests  
assert np.sum(abs(B)) != 0  
np.testing.assert_allclose(B @ B, np.zeros((2, 2)) )
```


Assignment manager

← → ↻ localhost:8888/formgrader#

nbgrader Manage Assignments



Manage Assignments

Manual Grading

Manage Students

Assignments

Instructions (click to expand)

The exchange directory at [does not exist and could not be created](#). The "release" and "collect" functionality will not be available. Please see the documentation on [Setting Up The Exchange](#) for instructions.

Search:

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
eams-22-demo	None	draft	✎	🔄	🔍			0		
Homework 1	None	draft	✎	🔄				0		
hw1	None	draft	✎	🔄	🔍			7	🗨	✕
hw11	None	draft	✎	🔄	🔍			5	🗨	✕
hw12	None	draft	✎	🔄	🔍			5	🗨	✕
hw15	None	draft	✎	🔄				0		
hw2	None	draft	✎	🔄	🔍			8	🗨	✕
hw3	None	draft	✎	🔄	🔍			8	🗨	✕
hw4	None	draft	✎	🔄	🔍			7	🗨	✕
hw5	None	draft	✎	🔄	🔍			7	🗨	✕
hw7	None	draft	✎	🔄	🔍			7	🗨	✕
hw8	None	draft	✎	🔄	🔍			7	🗨	✕
hw9	None	draft	✎	🔄	🔍			4	🗨	✕

+ Add new assignment...

Slides: bit.ly/yuster-eams-22

Grading/marking interface

In [18]:

Student's answer

```
##  
# (a) The goal of clustering a set of vectors is to choose the best vectors from the set  
a44 = False  
# (b) The goal of clustering a set of vectors is to divide them into groups of vectors that are near each other  
b44 = True  
# (c) The goal of clustering a set of vectors is to determine the nearest neighbors of each of the vectors  
c44 = False  
# (d) The k-means algorithm always converges to a clustering that minimizes the mean-square vector-representative d  
d44 = False  
# (e) The k-means algorithm can converge to different final clusterings, depending on the initial choice of represe  
e44 = True  
# (f) The k-means algorithm is widely used in practice  
f44 = True  
# (g) The choice of k, the number of clusters to partition a set of vectors into, depends on why you are clustering  
g44 = True  
# (h) The choice of k, the number of clusters to partition a set of vectors into, should always be as large as your  
h44 = False  
##
```

< >

Type any comments here (supports Markdown and MathJax)

← Autograded exercise

In [19]:

cell-d5eb3033421ae1d6

Full credit No credit 0.5 / 0.5 + 0 (extra credit)

```
assert a44 == True or a44 == False  
### BEGIN HIDDEN TESTS  
assert a44 == False  
### END HIDDEN TESTS
```

Grading/marking interface

3.10 Nearest neighbor document

Student's answer

Full credit

No credit

1

/ 1.0

+

0

(extra credit)

Consider the 5 Wikipedia pages in table 3.1 on page 51. What is the nearest neighbor of (the word count histogram vector of) 'Veterans Day' among the others? Does the answer make sense?

When looking at the Pairwise word count histogram distances between the five Wikipedia articles, the nearest neighbor of "Veterans Day" is "Memorial Day" since it has the smallest distance value of 0.095. Intuitively, this makes sense when comparing Veterans Day with the other entries such as the Academy Awards, the Golden Globes and the Super Bowl. Articles about Veterans Day and Memorial Day would naturally contain many of the same words since they are both US holidays dealing with veterans. These articles would not have much overlap with entertainment award shows or a big sporting event like the Super Bowl.

Type any comments here (supports Markdown and MathJax)

← **Manually graded exercise**

Returning graded/marked work

hw5 (Score: 10.7 / 11.0)

1. Written response (Score: 2.0 / 2.0)
2. Written response (Score: 1.0 / 1.0)
3. Written response (Score: 1.0 / 1.0)
4. Written response (Score: 1.7 / 2.0)
5. Comment
6. Written response (Score: 1.0 / 1.0)
7. Test cell (Score: 0.5 / 0.5)
8. Test cell (Score: 0.5 / 0.5)
9. Test cell (Score: 1.0 / 1.0)
10. Written response (Score: 1.0 / 1.0)
11. Coding free-response (Score: 0.5 / 0.5)
12. Written response (Score: 0.5 / 0.5)

Applied Linear Algebra

Dr. Yuster

Homework 5 (Textbook Ch. 6-7)

Name: ██████████

```
In [1]: import numpy as np
```

Answer the questions below using Markdown or code cells as appropriate. Use LaTeX syntax for all mathematical expressions in your narrative (see [this blog post](#) for more information and tips). You'll find the [VMLS Python Companion](#) to be very helpful for the coding portions.

Questions marked with an exercise number (like 1.2) can be found in the [textbook](#) at the end of that chapter. Questions marked with the AE prefix (like AE 1.2) can be found in the [Additional Exercises](#).

For coding portions, vectors should be represented as `numpy` arrays, and matrices should be represented as 2-D `numpy` arrays.

6.1 Matrix and vector notation. Answer all parts in the cell below.

Student's answer

Score: 2.0 / 2.0 (Top)

- (a) Yes, this expression uses valid notation. The dimensions would be an $(m \cdot n) \times 1$ matrix or an m -vector because you are stacking n m -vectors.
- (b) Yes, this expression uses valid notation. The dimensions would be an $n \times m$ matrix because the m -vectors are transposed (creating row vectors) and being stacked.
- (c) Yes, this expression uses valid notation. The dimensions would be an $m \times n$ matrix the vectors are being concatenated, creating a matrix with height m and width n .
- (d) Yes, this expression uses valid notation. The dimensions would be a $1 \times (mn)$ matrix or a (mn) row vector.

HTML document -
Attach to student's
assignment
submission in
VLE/LMS

What I liked

- Homework in Jupyter notebooks - loved the idea for this course!!!
- Easily combine written, computational, and coding problems
- Gentle introduction to LaTeX and markdown
- All free and open source tools
- Lightweight: Homework distributed as a single notebook file, no additional software installation for students

What I didn't like

- Solutions were automatically displayed in returned notebooks
 - [Configurable](#) in newest version of nbgrader?
- Imperfect LaTeX rendering in grading view
- Executing student code locally - yikes!
 - It's possible to [grade inside Docker containers](#)
- Clunky workflow
 - Smoother with JupyterHub

3.20 Regression model sensitivity. (Hint: Use the Cauchy-Schwarz inequality)

Student's answer

Score: 1.0 / 1.0 (Top)

Consider the regression model $\hat{y} = x^T \beta + v$, where \hat{y} is the prediction, x is a feature vector, β is a coefficient vector, and v is the offset term. If x and \tilde{x} are feature vectors with corresponding predictions \hat{y} and \tilde{y} show that $|\hat{y} - \tilde{y}| \leq \|\beta\| \|x - \tilde{x}\|$. This means that when $\|\beta\|$ is small, the prediction is not very sensitive to a change in the feature vector.

$$\begin{aligned} & |\hat{y} - \tilde{y}| \\ &= |(x^T \beta + v) - (\tilde{x}^T \beta + v)| \\ &= |x^T \beta + v - \tilde{x}^T \beta - v| \\ &= |x^T \beta - \tilde{x}^T \beta| \\ &= |(x - \tilde{x})^T \beta| \end{aligned}$$

Using the Cauchy-Schwarz inequality which says $|a^T b| \leq \|a\| \|b\|$ we will get the following:

$$|(x - \tilde{x})^T \beta| \leq \|\beta\| \|x - \tilde{x}\|$$

Comments:

Great work!

Student submission

LaTeX (non-) Rendering when Grading



Grading view

3.20 Regression model sensitivity. (Hint: Use the Cauchy-Schwarz inequality)

Student's answer

Full credit

No credit

1

/ 1.0

+

0

(extra credit)

Consider the regression model $\hat{y} = x^T \beta + v$, where \hat{y} is the prediction, x is a feature vector, β is a coefficient vector, and v is the offset term. If x and \tilde{x} are feature vectors with corresponding predictions \hat{y} and \tilde{y} show that $|\hat{y} - \tilde{y}| \leq \|\beta\| \|x - \tilde{x}\|$. This means that when $\|\beta\|$ is small, the prediction is not very sensitive to a change in the feature vector.

$|\hat{y} - \tilde{y}|$

$$= |(x^T \beta + v) - (\tilde{x}^T \beta + v)|$$

$$= |x^T \beta + v - \tilde{x}^T \beta - v|$$

$$= |x^T \beta - \tilde{x}^T \beta|$$

$$= |(x - \tilde{x})^T \beta|$$

Using the Cauchy-Schwarz inequality which says $|a^T b| \leq \|a\| \|b\|$ we will get the following:

$$|(x - \tilde{x})^T \beta| \leq \|\beta\| \|x - \tilde{x}\|$$

Great work!

Changes for next time

Plan to try [Otter Grader](#) + [Gradescope](#)

Resources

- Project Jupyter (<https://jupyter.org/>)
- Online book: Teaching and Learning with Jupyter (<https://jupyter4edu.github.io/jupyter-edu-book/>)
- nbgrader (<https://nbgrader.readthedocs.io/en/stable/>)
- canvas2nbgrader (<https://github.com/patrickwalls/canvas2nbgrader>)
- Textbook: Introduction to Applied Linear Algebra- Vectors, Matrices, and Least Squares by Boyd and Vandenberghe (<https://web.stanford.edu/~boyd/vmls/>)
- Otter Grader (<https://otter-grader.readthedocs.io/en/latest/>)
- Gradescope (<https://www.gradescope.com/>)