Introduction to Computing Technologies
MSBA 2023-2024

This session will be recorded

Prof Vincent Nijs // Rady @ UCSD

Get all the ICT files from GitHub (extract the zip file!)

Get all the ICT files from GitHub (extract the zip file!)

https://github.com/rady-msba/rsm-ict-2023/

Open the ICT-2023-links.html and ict.pdf files in your browser

3.1 Update and launch the RSM-JUPYTER computing environment

- Windows 10 or 11 from an Ubuntu Shell (WSL2)

Rerun step 4 in the install process. Get the code for Window from section 3.1 in the “Links” document

Re-run step 4 in the install process by running the code below. From an Ubuntu terminal run the below to get your username:

```
USERNAME=$(powershell.exe '$env:UserName'|tr -d '\r');
echo $USERNAME;
```

Then run the code below to set up connections to various folder:

```
if [ -d "/mnt/c/Users/$USERNAME/OneDrive/Desktop/" ]; then
  echo "Using Desktop backed up in OneDrive" >&2
  DTOP="/OneDrive/Desktop";
elif [ -d "/mnt/c/Users/$USERNAME/Desktop/" ]; then
  echo "Using Desktop folder in user home directory" >&2
  DTOP="/Desktop";
else
  echo "Unable to determine Desktop folder location" >&2
  exit 1
fi
echo "wt.exe wsl.exe ~/git/docker/launch-rsm-msba-intel.sh -v ~" > /mnt/c
chmod 755 /mnt/c/Users/"$USERNAME$DTOP"/launch-rsm-msba.bat;
cd ~;
ln -s /mnt/c/Users/"$USERNAME$DTOP"/ ./Desktop;
ln -s /mnt/c/Users/"$USERNAME"/Dropbox ./Dropbox;
ln -s /mnt/c/Users/"$USERNAME"/Downloads ./Downloads;
ln -s "/mnt/c/Users/$USERNAME/Google Drive" "./Google Drive";
ln -s /mnt/c/Users/"$USERNAME"/OneDrive ./OneDrive;
ln -s /mnt/c/Users/"$USERNAME" ./win_home;
/mnt/c/Users/"$USERNAME$DTOP"/launch-rsm-msba.bat;
```

3.1 Update and launch the RSM-JUPYTER computing environment

Turn OFF VPN for now!

- Windows 10 or 11 from an Ubuntu Shell (WSL2)

```
docker pull vnijs/rsm-msba-intel;
rm -rf ~/git/docker;
git clone https://github.com/radiant-rstats/docker.git ~/git/docker;
~/git/docker/launch-rsm-msba-intel.sh -v ~;
```

3.1 Update and launch the RSM-JUPYTER computing environment

- macOS (M1 or M2)

```
docker pull vnijs/rsm-msba-arm;
rm -rf ~/git/docker;
git clone https://github.com/radiant-rstats/docker.git ~/git/docker;
cp -p ~/git/docker/launch-rsm-msba-arm.sh ~/Desktop/launch-rsm-msba-arm.command;
~/Desktop/launch-rsm-msba-arm.command;
```

- macOS (Intel)

```
rm -rf ~/git/docker;
git clone https://github.com/radiant-rstats/docker.git ~/git/docker;
cp -p ~/git/docker/launch-rsm-msba-intel.sh ~/Desktop/launch-rsm-msba-intel.command;
~/Desktop/launch-rsm-msba-intel.command;
```

3.1 What you will see when you “launch” …

```
--------------------------------------------------------------------
Docker is not installed. Download and install Docker from
https://download.docker.com/mac/stable/Docker.dmg
--------------------------------------------------------------------
```

```
Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the
 docker daemon running?
--------------------------------------------------------------------
Waiting for docker to start ...
When docker has finished starting up press [ENTER} to continue
--------------------------------------------------------------------
```

./launch-rsm-jupyter.sh -v ~

```
11e23ac719b3: Already exists
40ccc697c028: Already exists
93172d0b50ab: Already exists
4f4fb700ef54: Already exists
792e4741d040: Already exists
1db80232d608: Already exists
564e7df72d9f: Already exists
bbedb275afc1: Already exists
ef812cd8aab7: Already exists
f1515db70833: Already exists
dd5e25eec1f3: Already exists
f211ebb6ad92: Already exists
12f05adaf45c: Already exists
586ca2f19596: Already exists
07723b48ac50: Already exists
ed0b9a89d281: Already exists
70ff6b62f501: Already exists
c5903559a6fb: Already exists
55732ad24869: Already exists
02fafe6680b2: Already exists
ea04a88c1ad6: Already exists
eaaaa22cffb1: Already exists
014b9bc7f735: Already exists
db28f56941e6: Already exists
39e9b6edb817: Already exists
```

~/Desktop/launch-rsm-msba-arm.command

```
--------------------------------------------------------------------------
Starting the rsm-msba-arm computing environment on macOS (ARM64)
Version   : 2.7.0
Build date: 2023-06-29
Base dir. : /Users/vnijs
Cont. name: rsm-msba-arm
--------------------------------------------------------------------------
Press (1) to show Jupyter Lab, followed by [ENTER]:
Press (2) to show Rstudio, followed by [ENTER]:
Press (3) to show Radiant, followed by [ENTER]:
Press (4) to show GitGadget, followed by [ENTER]:
Press (5) to show a (ZSH) terminal, followed by [ENTER]:
Press (6) to update the rsm-msba-arm container, followed by [ENTER]:
Press (7) to update the launch script, followed by [ENTER]:
Press (8) to clear Rstudio sessions and packages, followed by [ENTER]:
Press (9) to clear local Python packages, followed by [ENTER]:
Press (10) to start a Selenium container, followed by [ENTER]:
Press (h) to show help in the terminal and browser, followed by [ENTER]:
Press (c) to commit changes, followed by [ENTER]:
Press (q) to stop the docker process, followed by [ENTER]:
--------------------------------------------------------------------------
Note: To start, e.g., Jupyter on a different port type 1 8991 [ENTER]
Note: To start a specific container version type, e.g., 6 2.7.0 [ENTER]
Note: To commit changes to the container type, e.g., c myversion [ENTER]
--------------------------------------------------------------------------
```

Latest: 2.7.0 (2023-06-29)

3.1.1 Update Advanced Settings as needed (macOS)

1. Introduction

Prof Vincent Nijs // Rady @ UCSD

Get ready for the Rady MSBA program!

Introduction to Computing Technologies (ICT)

WORKSHOP OBJECTIVES

- Get ready to use Python, VS Code, Co-Pilot, and ChatGPT+ with Code Interpreter
- Get ready to use Postgres (SQL and ETL class)
- Get ready to use R and Radiant (for both R and Python)
- Get ready to use Git and GitHub for version control and Canvas, GitHub Classroom, and GradeScope for assignment submission

→ Goal for ICT: Other instructors won't have to spend as much time on setup for their classes, so they focus on content

#1 tip for the MSBA program: Be prepared, plan your time, and start early

- “The assignment is due tomorrow, and I haven’t gotten a response from the TA on the problem I posted on Piazza, like, 5 minutes ago!”
- Your assignments and cases will be challenging and time-consuming
- Do NOT wait until the last minute to complete your work!
- Make sure your computing environment is ready to complete required work!

Word of warning about this workshop … It may get messy

Who am I?

- MS from the University of Groningen (Netherlands)
- Ph.D. from the University of Leuven (Belgium)
- Research: Marketing Effectiveness
- At Rady since 2010
  - Co-founder and Faculty-director of the Master of Science in Business Analytics program at Rady
  - Associate Dean of Academic Programs
  - Associate Professor of Business Analytics and Marketing
- How to pronounce “Nijs”?
- Almost exactly like you would say the word “Nice”

1.1. How-to-pronounce-your-name form (before or after workshop)

- Go to the link below and login with your @ucsd.edu email
  - https://rsm-compute-01.ucsd.edu:4443/rady-profiles/
- Phonetic pronunciation of your full name (first and last name)
  - E.g., Vincent Nijs :: Fin-cent Nice
  - E.g., Vidya Chockalingam :: We-dya Chock-ligam
  - E.g., Huiyu Zheng :: Hui-U Djuhng
- Provide a short bio
- Use your phone to create and post an audio recording with the correct pronunciation of your full name (**convert to mp3**)
- Use your phone to create and post a (normal) selfie (**convert to png**). Do NOT use an "official" picture!

1.1 Student profile (before or after workshop)

Rady Profiles | Profile Submission Form

Profile Submission Form

Note: Fields with an * are required

First Name*

Vincent

Preferred Name (if different from first name)

Vincent

Last Name*

Nijs

Phonetic spelling of full name (first and last)

Fin-cent Nice

Select your Rady cohort*

Rady Prof

LinkedIn URL

https://www.linkedin.com/in/rady-msba-pro

Audio clip of full name (first and last) pronunciation (mp3)*

Browse... vnijs-name-pronunciation.mp3

Upload complete

0:02 / 0:03

You can create the audio file using your phone or at https://vocaroo.com/
If needed, convert your audio file to mp3 format using https://online-audio-converter.com/

A picture that clearly shows your face (png, max 10MB)*

Browse... vnijs-smaller.png

Upload complete

If needed, convert your image to png format using https://image.online-convert.com/convert-to-png
If needed, crop your image to 1:1 aspect ratio https://croppola.com/

Submit

https://rsm-compute-01.ucsd.edu:4443/rady-profiles/

1.2 Open the ICT-2023-links.html document from email or canvas

- Please download this HTML file and the slides for the ICT session to your Desktop
- After the workshop move these files to a location you can remember as you will likely need to access the files over time

Introduction to Computing Technologies: Links

AUTHORS
Professor Vincent R. Nijs
Rady School of Management @ UCSD

1 Introduction

The Introduction to Computing Technolgies (ICT) workshop will ensure everyone is set up with the tools needed to start and complete the Rady MSBA program successfully.

Use the links and required documentation in this document before, during, and after the ICT workshop as stated next to each header. Each section header indicates when you should take the suggested steps.

When we mention your MSBA program or cohort site on Canvas we are refering to the sites linked below:

- FT MSBA 2023 cohort site: https://rady.instructure.com/courses/1229
- FW (part-time) MSBA 2023 cohort site: https://rady.instructure.com/courses/1228

During the ICT session we will ask you to sit with your assigned study group. The MSBA ICT support team will be walking around to help out. You can also post screenshots and what happend on your computer to the Google Sheet below or to the RADY499 piazza site linked below and we will provide support as quickly as possible.

https://docs.google.com/presentation/d/1Fd-EPFC9f8ZGWSZJzRoE8-6chDj-vrYh3nICQV0I5no/edit?usp=sharing

Table of contents

1 Introduction
1.1 Provide your student profile (before or after workshop)
1.2 Store this HTML file on your Desktop (before workshop)
1.3 Install the Zoom app on our computer (before workshop)
1.4 Install the iClicker Student app on your phone (before workshop)
1.5 Browser (before workshop)
1.6 Programming and technical support through Piazza (before or during workshop)
1.7 Piazza site for the AI-Assisted Math and Programming for Business Analtyics class (before or during workshop)
1.8 Install VPN to access the MSBA server from off-campus (before workshop)
1.9 Download and install Google Drive for Desktop (any time)

1.2 Set up all the tools

Look for “ICT-2023-links.html” in your @ucsd.edu inbox or canvas announcements on the program site

MSBA Program 2024
Flex MSBA Program 2024

1.3 Download and Install the Zoom app on your computer

- You must have the zoom app installed on your computer to have access to all relevant features (i.e., don't use Zoom through your browser)
  - https://ucsd.zoom.us/download

- Click the link below to claim your UC San Diego Zoom Pro account using Single Sign-On (SSO)
  - https://ucsd.zoom.us/signin

- Hopefully, you won't need Zoom for classes but be prepared! We may use zoom for work-sessions, office hours, group work, etc.

1.3 Learning in a hybrid classroom (FW MSBA only)

- “Roomies” and “Zoomies”

- Speak clearly so everyone can hear the question

- Zoomies can chat with a TA, raise their hand, and unmute to talk

- Zoomies may need to use UCSD VPN to connect to services from off-campus:
  - https://blink.ucsd.edu/technology/network/connections/off-campus/VPN/index.html
  - Try “2-Step Secured – split” so that only UCSD traffic goes through VPN

1.4 iClicker Student App

Create an account:

https://www.iclicker.com/students/apps-and-remotes/apps

Install the app on your phone:

https://www.youtube.com/watch?v=liozvhPhtJA

Add a class:

- "University of CA San Diego Rady School of Management"
- "Introduction to Computing Technologies (FT)" or "Introduction to Computing Technologies (FW)"

1.5 What browser do you use?

1.5 Let's test out iClicker app

What is the default browser on your laptop?

A. Chrome
B. Firefox
C. Safari
D. Edge
E. Other

1.6 Piazza site for general programming and tech support

https://piazza.com/rady.ucsd/summer2023/rady499

School: UCSD Rady School of Management
Term: Summer 2023
Course Number: Rady 499
Course Name: Technical and Programming Support

TAGS

- ChatGPT
- Python
- Docker
- Git
- SQL
- Postgres
- VS Code
- Windows
- macOS

1. Ask questions

The best way to get answers is to ask questions! Ask questions on Piazza rather than emailing teaching staff so everyone can benefit from the response (and so you can get answers from classmates who are up as late as you are). Before you post a new question do a search to see if your question has already been asked (and perhaps answered).

1.6 Piazza site for general programming and tech support

2. Edit questions and answers wiki-style
   - Think of Piazza as a Q&A wiki for your class. Every question has one answer that students can edit collectively and one instructor answer
   - For more information on how to ask questions see: editing answer on Piazza
3. Add a follow-up to comment or ask further questions
4. Make your posts **public** in most cases. Post **anonymously** to classmates if you prefer (**public but anonymous**)

1.7 Piazza site for Math and Programming for Business Analytics

https://piazza.com/ucsd/summer2023/mgta403ft

School: UCSD
Term: Summer 2023
Course Number: MGTA 403 (FT)
Course Name: AI-Assisted Math and Programming for Business Analytics (FT)

https://piazza.com/ucsd/summer2023/mgta403fw

School: UCSD
Term: Summer 2023
Course Number: MGTA 403 (FW)
Course Name: AI-Assisted Math and Programming for Business Analytics (FW)

TAGS

- Logistics
- Git
- GitHub
- DataCamp
- Math Stats Bootcamp
- ChatGPT
- CodeInterpreter
- Python
- VS Code
- Radiant for R
- Radiant for Python

1.7 Reporting issues and asking questions (create a “reprex”)

- Post (public) on Piazza. You can post anonymously if you prefer
- Provide a clear description detailed information about what you did and what happened
- Help others to help you by providing enough information so they can reproduce the problem (i.e., help us help you). E.g., what OS, etc. etc.
- What error or warning messages are shown on screen?
- **Provide screen shots**:
- **Learn the keyboard shortcuts**
  macOS (Shift + Cmd + 4 or 5)
  Windows (Shift + Win + S)

macOS: Screenshot

Windows:
Snip & Sketch

1.7 Reporting issues and asking questions (create a “reprex”)

Learn the keyboard shortcuts:

macOS (Shift + Cmd + 4 or 5)

Windows (Shift + Win + S)

1.8 VPN with 2FA required to connect to (most) services on MSBA server

For off-campus connection to the server you will need VPN with 2-factor authentication

Cisco AnyConnect VPN

CISCO

http://blink.ucsd.edu/technology/network/connections/off-campus/VPN/windows10.html

http://blink.ucsd.edu/technology/network/connections/off-campus/VPN/mac6.html

2. Program Preparation

Prof Vincent Nijs // Rady @ UCSD

2.1 DataCamp (ChatGPT, Python, R, SQL, Shell)

- Online platform focused on introduction to technical skills
- Free access for 6 months (will be renewed)

datacamp

2.2 Math and Stats Bootcamp

Math & Stats Boot camp | Introduction | Cases | Your answers | Logged in as vnijs Logout

RADY MATH & STATS BOOT CAMP

Getting ready to launch your Rady career

Today more than ever analytical skills are required to leverage "big data" and enhance decision-making. Solving business problems requires mathematics and statistics. For example, specific calculations are needed to price and market products without exceeding budget restrictions. Maximizing the return on a stock portfolio requires knowledge of optimization concepts. Managing inventory requires an understanding of probability to deal with demand uncertainty.

Before the start of fall classes, all incoming Rady students must complete a self-paced, online math & stats boot camp to ensure everyone is adequately prepared to handle the rigor of the Rady program. The boot camp is an essential review of mathematical and statistical concepts relevant in a business context.

In this online course you will review optimization, probability, and statistics in a series of business cases. The case context is a business venture started by a Rady alumnus in 2014. We expect that you will already be familiar with the mathematical concepts. However, the goal is to develop an understanding of how these concepts are applied in business. Topics covered include:

- Demand and profit functions
- Optimization
- Break-even analysis
- Compounding of interest
- Demand uncertainty
- Describing and visualizing business data

https://rsm-compute-01.ucsd.edu:4443/msbabootcamp2023/

3. Set up required tools for the MSBA program

Prof Vincent Nijs // Rady @ UCSD

3.1 Install Docker and the RSM-MSBA computing environment

Windows:

https://github.com/radiant-rstats/docker/blob/master/install/rsm-msba-windows.md

macOS (M1 and M2):

https://github.com/radiant-rstats/docker/blob/master/install/rsm-msba-macos-m1.md

macOS (Intel):

https://github.com/radiant-rstats/docker/blob/master/install/rsm-msba-macos.md

Linux (Ubuntu 22.04):

https://github.com/radiant-rstats/docker/blob/master/install/rsm-msba-linux.md

ChromeOS:

https://github.com/radiant-rstats/docker/blob/master/install/rsm-msba-chromeos.md

3.1 Update and launch the RSM-JUPYTER computing environment

- Windows 10 or 11 from an Ubuntu Shell (WSL2)

Rerun step 4 in the install process. Get the code for Window from section 3.1 in the “Links” document

Re-run step 4 in the install process by running the code below. From an Ubuntu terminal run the below to get your username:

```
USERNAME=$(powershell.exe '$env:UserName'|tr -d '\r');
echo $USERNAME;
```

Then run the code below to set up connections to various folder:

```
if [ -d "/mnt/c/Users/$USERNAME/OneDrive/Desktop/" ]; then
  echo "Using Desktop backed up in OneDrive" >&2
  DTOP="/OneDrive/Desktop";
elif [ -d "/mnt/c/Users/$USERNAME/Desktop/" ]; then
  echo "Using Desktop folder in user home directory" >&2
  DTOP="/Desktop";
else
  echo "Unable to determine Desktop folder location" >&2
  exit 1
fi
echo "wt.exe wsl.exe ~/git/docker/launch-rsm-msba-intel.sh -v ~" > /mnt/c
chmod 755 /mnt/c/Users/"$USERNAME$DTOP"/launch-rsm-msba.bat;
cd ~;
ln -s /mnt/c/Users/"$USERNAME$DTOP"/ ./Desktop;
ln -s /mnt/c/Users/"$USERNAME"/Dropbox ./Dropbox;
ln -s /mnt/c/Users/"$USERNAME"/Downloads ./Downloads;
ln -s "/mnt/c/Users/$USERNAME/Google Drive" "./Google Drive";
ln -s /mnt/c/Users/"$USERNAME"/OneDrive ./OneDrive;
ln -s /mnt/c/Users/"$USERNAME" ./win_home;
/mnt/c/Users/"$USERNAME$DTOP"/launch-rsm-msba.bat;
```

3.1 Update and launch the RSM-JUPYTER computing environment

Turn OFF VPN for now!

- Windows 10 or 11 from an Ubuntu Shell (WSL2)

```
docker pull vnijs/rsm-msba-intel;
rm -rf ~/git/docker;
git clone https://github.com/radiant-rstats/docker.git ~/git/docker;
~/git/docker/launch-rsm-msba-intel.sh -v ~;
```

3.1 Update and launch the RSM-JUPYTER computing environment

- macOS (M1 or M2)

```
docker pull vnijs/rsm-msba-arm;
rm -rf ~/git/docker;
git clone https://github.com/radiant-rstats/docker.git ~/git/docker;
cp -p ~/git/docker/launch-rsm-msba-arm.sh ~/Desktop/launch-rsm-msba-arm.command;
~/Desktop/launch-rsm-msba-arm.command;
```

- macOS (Intel)

```
docker pull vnijs/rsm-msba-intel;
rm -rf ~/git/docker;
git clone https://github.com/radiant-rstats/docker.git ~/git/docker;
cp -p ~/git/docker/launch-rsm-msba-intel.sh ~/Desktop/launch-rsm-msba-intel.command;
~/Desktop/launch-rsm-msba-intel.command;
```

3.1 What you will see when you “launch” …

```
--------------------------------------------------------------------
Docker is not installed. Download and install Docker from
https://download.docker.com/mac/stable/Docker.dmg
--------------------------------------------------------------------
```

```
Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the
 docker daemon running?
--------------------------------------------------------------------
Waiting for docker to start ...
When docker has finished starting up press [ENTER} to continue
--------------------------------------------------------------------
```

./launch-rsm-jupyter.sh -v ~

```
11e23ac719b3: Already exists
40ccc697c028: Already exists
93172d0b50ab: Already exists
4f4fb700ef54: Already exists
792e4741d040: Already exists
1db80232d608: Already exists
564e7df72d9f: Already exists
bbedb275afc1: Already exists
ef812cd8aab7: Already exists
f1515db70833: Already exists
dd5e25eec1f3: Already exists
f211ebb6ad92: Already exists
12f05adaf45c: Already exists
586ca2f19596: Already exists
07723b48ac50: Already exists
ed0b9a89d281: Already exists
70ff6b62f501: Already exists
c5903559a6fb: Already exists
55732ad24869: Already exists
02fafe6680b2: Already exists
ea04a88c1ad6: Already exists
eaaaa22cffb1: Already exists
014b9bc7f735: Already exists
db28f56941e6: Already exists
39e9b6edb817: Already exists
```

~/Desktop/launch-rsm-msba-arm.command

```
--------------------------------------------------------------------------
Starting the rsm-msba-arm computing environment on macOS (ARM64)
Version   : 2.7.0
Build date: 2023-06-29
Base dir. : /Users/vnijs
Cont. name: rsm-msba-arm
--------------------------------------------------------------------------
Press (1) to show Jupyter Lab, followed by [ENTER]:
Press (2) to show Rstudio, followed by [ENTER]:
Press (3) to show Radiant, followed by [ENTER]:
Press (4) to show GitGadget, followed by [ENTER]:
Press (5) to show a (ZSH) terminal, followed by [ENTER]:
Press (6) to update the rsm-msba-arm container, followed by [ENTER]:
Press (7) to update the launch script, followed by [ENTER]:
Press (8) to clear Rstudio sessions and packages, followed by [ENTER]:
Press (9) to clear local Python packages, followed by [ENTER]:
Press (10) to start a Selenium container, followed by [ENTER]:
Press (h) to show help in the terminal and browser, followed by [ENTER]:
Press (c) to commit changes, followed by [ENTER]:
Press (q) to stop the docker process, followed by [ENTER]:
--------------------------------------------------------------------------
Note: To start, e.g., Jupyter on a different port type 1 8991 [ENTER]
Note: To start a specific container version type, e.g., 6 2.7.0 [ENTER]
Note: To commit changes to the container type, e.g., c myversion [ENTER]
--------------------------------------------------------------------------
```

Latest: 2.7.0 (2023-06-29)

3.1 Trouble shooting on Windows

- Type ‘whoami’ from an Ubuntu shell – You should NOT be “root”. If you are, remove Ubuntu, re-install, and set a username
- Check what version of Windows you have (Type “winver” in Powershell)
- Must be Windows 11 or Windows 10 Professional, Education, or Enterprise
- Windows 10, version 2004 or higher

```
PS C:\WINDOWS\system32> wsl --list
Windows Subsystem for Linux Distributions:
Ubuntu-22.04 (Default)
docker-desktop-data
docker-desktop
```

3.1 Press (h) for help, and (q) to stop

Notice the URL: **localhost:8989**

3.1.1 Update Docker Resource settings as needed (macOS)

3.1.2 Update Docker Advanced Settings as needed (macOS)

3.2 Setting up your terminal (install Nerd Fonts)

https://github.com/romkatv/powerlevel10k#manual-font-installation

Manual font installation

1. Download these four ttf files:
   - MesloLGS NF Regular.ttf
   - MesloLGS NF Bold.ttf
   - MesloLGS NF Italic.ttf
   - MesloLGS NF Bold Italic.ttf
2. Double-click on each file and click "Install". This will make `MesloLGS NF` font available to all applications on your system.

3.2 Setting up your terminal on macOS

https://iterm2.com/

3.2 Right click on launcher Icon on your Desktop and “Get Info”

3.2 Windows WSL2: Setting some defaults

Press
Ctrl + , or use the down-carrot and select settings

3.2 Windows WSL2: Setting some defaults

~

3.2 Finalize docker setup for Zsh, Rstudio, TinyTex and Radiant on your laptop

- Open a terminal from the Launch Menu using 5 + Enter
- Press q + Enter if prompted
- Type "**setup;**" and follow any prompts
- When done, type "**exit;**"
- Open a new terminal from the Launch Menu using 5 + Enter
- If all went well, the new font and icons will be available

3.2 OhMyZsh with icon in your macOS (iterm2) or Windows Terminal (Optional)

Pin Windows Terminal to your Taskbar

Add iTerm2 to your Dock

To add icons to your OS termina follow the instructions linked below

https://github.com/radiant-rstats/docker/blob/master/install/setup-ohmyzsh.md

3.2 Setting up your terminal on macOS

https://iterm2.com/

iTerm2: Preferences → Profiles → Text and set Font to "MesloLGS NF"

Windows WSL2: Setting some defaults

Scroll down in Ubuntu (22.04) settings and click on Appearance

Set the font face to MesloGS NF

4. Tools for Business Analytics

What about a spreadsheet (e.g., Excel)?

Hospital Operations - solution.xlsx

	A	B	C	D	E	F	G	H
1					Required			
2	Location	New Patients	Maint. Patients		Physicians		Physician Costs	Objective Function
3	La Jolla	75	318.75		5.390625		$72,000.00	$38,584.48
4	Hillcrest	147.5	307.5		6.609375			
5					12	Note: Number of doctors cannot exceed 12		
6								
7	Patient Type	La Jolla %	Hillcrest %	Rev $ (NP)	Rev $ (Maint.)			
8	PPO	65%	21%	214	175			
9	HMO	19%	32%	143	121			
10	Medicare	9%	24%	125	103			
11	Medicaid	5%	9%	45	31			
12	Uninsured	2%	14%	0	0			
13								
14								
15		Available	Available					
16	Location	New Patients	Maint. Patients	Max NP Con.	Max Maint Con.	Note: Cannot see more patients than are available		
17	La Jolla	150	475	75	318.75			
18	Hillcrest	295	615	147.5	307.5			
19								
20								
21		Fraction Seen	Fraction Seen	50% Con.	50% Con.			
22	Location	New Patients	Maint. Patients	New Patients	Maint. Patients			
23	La Jolla	0.5	0.671052632	50%	50%	Note: Must serve at least 50% of available demand		
24	Hillcrest	0.5	0.5	50%	50%			
25								
26								
27		Total Seen	Minimum					
28	Location	Maint. Patients	Maint. Patients					
29	La Jolla	318.75	150	Note: Number of maintanance patients must be 2 X the number of new patients				
30	Hillcrest	307.5	295					
31								

Sheet1 | Sensitivity Report

Stata?

Statistics
User
Window
Help
Stata/SE 14.0
Viewer
Graph
Do-file Editor
Data Editor
Data Browser
More
Break
2015 StataCorp LP
ive
, Texas 77845 USA
Summaries, tables, and tests
Linear models and related
Binary outcomes
Ordinal outcomes
Categorical outcomes
Count outcomes
Fractional outcomes
Generalized linear models
Treatment effects
Endogenous covariates
Sample-selection models
Exact statistics
Nonparametric analysis
Time series
Multivariate time series
Longitudinal/panel data
Multilevel mixed-effects models
Survival analysis
Epidemiology and related
SEM (structural equation modeling)
IRT (item response theory)
Survey data analysis
Multiple imputation
Multivariate analysis
Power and sample size
Bayesian analysis
Resampling
Postestimation
Other
Exact logistic regression
Exact Poisson regression
Centiles with CIs
Two-sample Kolmogorov-Smirnov test
K-sample equality-of-medians test
Binomial probability test
Binomial probability test calculator
Symmetry and marginal homogeneity test
Symmetry and marginal homogeneity test calculator
Two-way table with measures of association
All possible two-way tables
Table calculator
Large one-way ANOVA
Tetrachoric correlations
Indirect standardization
Tables for epidemiologists
Incidence-rate ratio
Incidence-rate ratio calculator
Cohort study risk-ratio etc.
Cohort study risk-ratio etc. calculator
Case-control odds ratio
Case-control odds-ratio calculator
Matched case-control studies
Matched case-control calculator

SAS?

Open-source tools are key to a successful career in analytics

- Python and R are the most powerful tools for data analysis in existence today
- Python and R are open source and run on Mac, PC, and Linux
- Great graphics (e.g., ggplot for R), help resources, online distribution system, etc.
- R has an excellent package management system (CRAN). Python has something like 15 package manage systems that are decent
- R is the best tool for statistics and is used by the majority of statisticians
- Python is used by many engineers, computer scientists, and data scientists
- Both Python and R have bindings to fast computational libraries in C and Fortran
- Python has the most powerful DeepLearning libraries
  - Pytorch
  - Tensorflow and Keras

PyTorch

K + TensorFlow

tool
“One ~~ring~~ to rule them all”?

PyTorch

APACHE Spark

Coding for reproducible analysis

"A minimal standard for data analysis and other scientific computations is that they be reproducible: that the code and data are assembled in a way so that another group can re-create all of the results (e.g., the figures in a paper). Adopting a workflow that will make your results reproducible will ultimately make your life easier; if a problem (or question) arises somewhere down the line, it will be much easier to correct (or explain)."

Source: http://kbroman.org/steps2rr/
Additional discussion: https://www.slideshare.net/RevolutionAnalytics/reproducible-data-science-with-r

Can you just give managers and decision makers code?

Key tools for (1) reproducible research and (2) clear communication of results

Easier
Radiant: ~~Easy~~ access to the power of R

Code: https://github.com/radiant-rstats

Documentation: https://radiant-rstats.github.io/docs

Easier
Radiant: ~~Easy~~ access to the power of Python

Code: https://github.com/vnijs/pyrsm

Power skill: Story telling with data and analytics (MGTA 402)

https://www.fusioncharts.com/blog/storytelling-data-visualization-marketing/

5. Why use Docker?

How to create a consistent computing platform?

Minimal requirement for reproducible analysis is …

a. Code used to generate results
b. + Data used in analysis
c. + Analysis tool
   (e.g., R or Python)
d. + Analysis tool versions
   (e.g., R version, package version)
e. + OS used

MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.

https://xkcd.com/1987/

How to create a consistent computing environment?

On the importance of a consistent computing environment

https://medium.com/kredaro-engineering/ai-tales-building-machine-learning-pipeline-using-kubeflow-and-minio-4b88da30437b

What is Docker? Why use it?

- Think of it as a “light weight Virtual Machine”
- But you don’t have to learn the GUI for a new OS (i.e., Linux)
- Provides consistency in the computing environment across students, groups, TAs, and instructors, regardless of host operating system they use
- Isolated computing environment ensures analysis results are reproducible

What is docker: https://www.youtube.com/watch?v=YFl2mCHdv24

On the importance of docker (and Kubernetes)

Other tools

More respondents want to continue using Cargo next year than the top competitors (top 6 tools that respondents want to use next year), however, Docker has almost double the proportion of respondents that want to use it next year compared to all other options.

What tools are in the RSM-MSBA Docker image?

- Python 3.11.4 + Python packages (incl. pyrsm and Radiant for Python)
- Conda 23.5.0
- JupyterLab
- R 4.2.3 + R packages (incl. Radiant)
- Rstudio Server
- PostgreSQL 14
- Spark
- Bash (ZSH)
- PyTorch
- VS Code (on localhost)

Source code: https://github.com/radiant-rstats/docker

Images:

- https://hub.docker.com/r/vnijs/rsm-msba-arm
- https://hub.docker.com/r/vnijs/rsm-msba-intel

Accessible on the MSBA server and available for your laptop and/or desktop

How are the tools connected?

6. VS Code

On the importance of VS Code as a technology

6.1 Set up VS Code on macOS

- Microsoft's open-source integrated development environment (IDE), VS Code or Visual Studio Code, was the most popular development environment based on a recent Stack Overflow developer survey. VS Code is widely used by Google developers and is the default development environment at Facebook.

- VS Code is great for Python, R, SQL, and many other languages

- Get VS Code: https://code.visualstudio.com/download

- To install a set of extensions from a terminal on macOS open up the **Command Palette** in VS Code by pressing **Shift + Cmd + P** to install the "code" shell command

- Next type the below in a terminal on macOS

```
cd ~/git/docker/vscode;
./extension-install.sh;
cd -;
```

6.1 Set up VS Code on Windows

- On Windows you should already have VS Code installed
- To install a set of extensions, enter the commands below in PowerShell
- Copy-and-Paste the command from the links document (section 6.1).

```
Invoke-WebRequest -Uri https://raw.githubusercontent.com/radiant-
rstats/docker/master/vscode/extensions.txt -OutFile extensions.txt;
cat extensions.txt |% { code --install-extension $_ --force};
del extensions.txt;
```

VS Code: Attach to Running Container

VS Code: Click on the extensions icon, scroll down, and then click “Install in Container …” for all packages with such a button

VS Code: Set default python to use from Command Palette

* If Python: Select Interpreter is not listed, return to the previous slide and make sure you completed that step

VS Code: Change Terminal Font through Command Palette

Open the Command Palette and type “settings”

VS Code: Always set the “Kernel” to use for a notebook

VS Code resources

- Python in VS Code: https://code.visualstudio.com/docs/languages/python
- R in VS Code: https://code.visualstudio.com/docs/languages/r
- Attach to running docker container: https://code.visualstudio.com/docs/remote/attach-container
- Customizing settings.json:
  - Shift + CTRL (CMD) + P > Preferences: Open settings (JSON)
  - Copy-and-paste from https://github.com/radiant-rstats/docker/blob/master/vscode/settings-vscode.json

7. Using the RSM-MSBA docker container

7.1 Start the RSM-MSBA (ARM or INTEL) computing environment

- Start the Docker container by double-clicking the launch icon on your Desktop

For macOS (Intel) and Windows you can always use the command below from a (Ubuntu) terminal

```
~/git/docker/launch-rsm-msba-intel.sh -v ~
```

For macOS (M1 or M2) you can always use the command below from a terminal

```
~/git/docker/launch-rsm-msba-arm.sh -v ~
```

7.1 What you will see when you “launch” …

```
--------------------------------------------------------------------
Docker is not installed. Download and install Docker from
https://download.docker.com/mac/stable/Docker.dmg
--------------------------------------------------------------------
```

```
Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the
 docker daemon running?
--------------------------------------------------------------------
Waiting for docker to start ...
When docker has finished starting up press [ENTER} to continue
--------------------------------------------------------------------
```

./launch-rsm-jupyter.sh -v ~

```
11e23ac719b3: Already exists
40ccc697c028: Already exists
93172d0b50ab: Already exists
4f4fb700ef54: Already exists
792e4741d040: Already exists
1db80232d608: Already exists
564e7df72d9f: Already exists
bbedb275afc1: Already exists
ef812cd8aab7: Already exists
f1515db70833: Already exists
dd5e25eec1f3: Already exists
f211ebb6ad92: Already exists
12f05adaf45c: Already exists
586ca2f19596: Already exists
07723b48ac50: Already exists
ed0b9a89d281: Already exists
70ff6b62f501: Already exists
c5903559a6fb: Already exists
55732ad24869: Already exists
02fafe6680b2: Already exists
ea04a88c1ad6: Already exists
eaaaa22cffb1: Already exists
014b9bc7f735: Already exists
db28f56941e6: Already exists
39e9b6edb817: Already exists
```

~/Desktop/launch-rsm-msba-arm.command

```
--------------------------------------------------------------------------
Starting the rsm-msba-arm computing environment on macOS (ARM64)
Version   : 2.7.0
Build date: 2023-06-29
Base dir. : /Users/vnijs
Cont. name: rsm-msba-arm
--------------------------------------------------------------------------
Press (1) to show Jupyter Lab, followed by [ENTER]:
Press (2) to show Rstudio, followed by [ENTER]:
Press (3) to show Radiant, followed by [ENTER]:
Press (4) to show GitGadget, followed by [ENTER]:
Press (5) to show a (ZSH) terminal, followed by [ENTER]:
Press (6) to update the rsm-msba-arm container, followed by [ENTER]:
Press (7) to update the launch script, followed by [ENTER]:
Press (8) to clear Rstudio sessions and packages, followed by [ENTER]:
Press (9) to clear local Python packages, followed by [ENTER]:
Press (10) to start a Selenium container, followed by [ENTER]:
Press (h) to show help in the terminal and browser, followed by [ENTER]:
Press (c) to commit changes, followed by [ENTER]:
Press (q) to stop the docker process, followed by [ENTER]:
--------------------------------------------------------------------------
Note: To start, e.g., Jupyter on a different port type 1 8991 [ENTER]
Note: To start a specific container version type, e.g., 6 2.7.0 [ENTER]
Note: To commit changes to the container type, e.g., c myversion [ENTER]
--------------------------------------------------------------------------
```

Latest: 2.7.0 (2023-06-29)

7.1 Updating the computing environment and launch-script

```
~/Desktop/launch-rsm-msba-arm.command
--------------------------------------------------------------------------
Starting the rsm-msba-arm computing environment on macOS (ARM64)
Version   : 2.7.0
Build date: 2023-06-29
Base dir. : /Users/vnijs
Cont. name: rsm-msba-arm
--------------------------------------------------------------------------
Press (1) to show Jupyter Lab, followed by [ENTER]:
Press (2) to show Rstudio, followed by [ENTER]:
Press (3) to show Radiant, followed by [ENTER]:
Press (4) to show GitGadget, followed by [ENTER]:
Press (5) to show a (ZSH) terminal, followed by [ENTER]:
Press (6) to update the rsm-msba-arm container, followed by [ENTER]:
Press (7) to update the launch script, followed by [ENTER]:
Press (8) to clear Rstudio sessions and packages, followed by [ENTER]:
Press (9) to clear local Python packages, followed by [ENTER]:
Press (10) to start a Selenium container, followed by [ENTER]:
Press (h) to show help in the terminal and browser, followed by [ENTER]:
Press (c) to commit changes, followed by [ENTER]:
Press (q) to stop the docker process, followed by [ENTER]:
--------------------------------------------------------------------------
Note: To start, e.g., Jupyter on a different port type 1 8991 [ENTER]
Note: To start a specific container version type, e.g., 6 2.7.0 [ENTER]
Note: To commit changes to the container type, e.g., c myversion [ENTER]
--------------------------------------------------------------------------
```

- Press 6 to update the rsm-jupyter container
- Press 7 to update the launch script (or "**git pull**" from the ~/git/docker directory)
- **Press h to show help and documentation**

7.2 Jupyter Lab on the MSBA server (VPN required off-campus)

https://rsm-compute-01.ucsd.edu:8000 login with UCSD id and password

7.2 WIP Connect to docker container through Kubernetes (VPN required off-campus) WIP

7.3 Open rsm-ict-2023/rsm-icit-init-notebook/python-notebook.ipynb

Run shell commands Using the terminal in VS Code (open zsh-script.sh)

8. Where is my data?

8.1. Accessing files from macOS

8.2 Windows WSL2: Where are my files?

In an Ubuntu Terminal type:

cd ~
explorer.exe ..

In the file explorer window that opens, right click on the folder you see with your username and select "Pin to Quick Access"

8.3 What is a ”path”? Or, how to find your data

… a **path** defines the location of a **file** or folder in a computer's **file** system. **Paths** are also called "directory **paths**" because they often include one or more directories that describe the **path** to the **file** or folder.

8.3 What is a ”path”? Or, how to find your data

8.3 Absolute and Relative paths

- simpsons1.png
- simpsons2.png
- simpsons3.png
- …

Absolute path: /home/Roger/Photos/simpsons1.png

Relative path: Photos/simpsons1.png

8.3 Absolute and Relative paths

If the first line of your R script is
setwd(“/home/jenny/path/that/only/I/have")
I will come into your office and SET YOUR COMPUTER ON FIRE 🔥.

Source: https://www.tidyverse.org/articles/2017/12/workflow-vs-script/

If you load your data using
houseprices <- readr::read_rds("/home/vnijs/path/that/only/I/have/houseprices.rds")
your TA, Prof, or collaborators, will SET YOUR COMPUTER ON FIRE 🔥.

8.3 Absolute and Relative paths: How to find your data

8.3 Always use relative paths!

Navigate to a project directory using “File > Open Folder” or CMD + O (macOS) or CTRL + K O (Windows)

In Jupyter notebook, access data relative to the location of the notebook. **This overrules a VS Code project directory setting**

Start Radiant-for-Python apps from a notebook in VS Code

In Rstudio, always use ‘projects’ to set the base directory

Start Radiant-for-R from an Rstudio project

Download and install cloud drive apps on your laptop

As a UCSD student you already have a Google Drive account. To install the Desktop software, use the instructions linked below:

- Google Drive for Desktop: https://www.google.com/drive/download/
- This is useful when professors want to share files with you or when you want to share data and files with your study group

9. Connecting to PostgreSQL (aka postgres)

On the importance of PostgreSQL

9.1 Accessing the PostgreSQL (aka postgres) server from a notebook

Use the file browser in JupyterLab to open rsm-ict-2023/rsm-icit-init-notebook/postgres-connect-py.ipynb

If there is a connection error, stop the container, and then run the following from an Ubuntu or macOS terminal: **docker volume rm pg_data**

9.2 Accessing the PostgreSQL from pgweb

Enter the below in the Scheme tab in pgweb
postgresql://jovyan:postgres@127.0.0.1:8765/rsm-docker

9.2 Accessing the PostgreSQL (aka postgres) server

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
21	6	160	110	3.9	2.62	16.46	0	1	4	4
21	6	160	110	3.9	2.875	17.02	0	1	4	4
22.8	4	108	93	3.85	2.32	18.61	1	1	4	1
21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
18.7	8	360	175	3.15	3.44	17.02	0	0	3	2

9.3 Accessing the PostgreSQL (aka postgres) server from VS Code

- Click on on the icon in the left navigation bar. If you don't see it, click on "…" in the left navigation bar and then on "PostgreSQL Explorer"

- Click on the + in top left of your VS Code window to Add Database Connection

- Use:
  - 127.0.0.1 as the hostname
  - "jovyan" as the PostgreSQL user
  - "postgres" as the password
  - 8765 as the port number
  - Standard connection
  - "rsm-docker" as the database
  - "rsm-docker" as the display name

9.3 Accessing the rsm-docker database from VS Code

Open rsm-ict-2023/rsm-ict-init-notebook/postgres-connect-vscode.pgsql

Open PostgreSQL Explorer

If you don't see the Elephant icon, click on the "more" icon "…"

F5 to run query (or right-click)

9.4 Adding Databases for the SQL class

Enter the command below in a terminal in VS Code

source <(curl -s https://raw.githubusercontent.com/radiant-rstats/docker/master/postgres/postgres-createdb.sh)

Run the script and press “y” when it is complete to delete the raw data files

Copy the below in the Scheme tab in pgweb to confirm access to the Northwind database:

postgresql://jovyan:postgres@127.0.0.1:8765/Northwind

Copy the below in the Scheme tab in pgweb to confirm access to the WestCoastImporters database:

postgresql://jovyan:postgres@127.0.0.1:8765/WestCoastImporters

9.5 Adding Connections to the new databases in VS Code

- Click on on the icon in the left navigation bar. If you don't see it, click on "…" in the left navigation bar and then on "PostgreSQL Explorer"

- Click on the + in top left of your VS Code window to Add Database Connection

- Use:
  - 127.0.0.1 as the hostname
  - "jovyan" as the PostgreSQL user
  - "postgres" as the password
  - 8765 as the port number
  - Standard connection
  - "Northwind" as the database
  - "Northwind" as the display name

9.5 Adding Connections to the new databases in VS Code

- Click on on the icon in the left navigation bar. If you don't see it, click on "…" in the left navigation bar and then on "PostgreSQL Explorer"

- Click on the + in top left of your VS Code window to Add Database Connection

- Use:
  - 127.0.0.1 as the hostname
  - "jovyan" as the PostgreSQL user
  - "postgres" as the password
  - 8765 as the port number
  - Standard connection
  - "WestCoastImporters" as the database
  - "WestCoastImporters" as the display name

9.5 Accessing the new databases from VS Code

Open rsm-ict-2023/rsm-ict-init-notebook/postgres-connect-vscode.pgsql

Open PostgreSQL Explorer

If you don't see the Elephant icon, click on the "more" icon "…"

F5 to run query (or right-click)

10. Git and GitHub for version control

10.1 Create a new account on GitHub

- Go to https://github.com

- Sign up using “rsm-” + the first part of your @ucsd.edu email address as the ***Username***
  - *e.g., rsm-xyz123*

Take a picture of the part of your student ID *with* the barcode on it

10.2 Sign up for access to Copilot, Copilot Chat, and Copilot Labs

When you sign up for GitHub as a student, you will get free access to Copilot.
Please also signup for the Copilot Chat and Labs waitlists at the links below:

Copilot: https://github.com/github-copilot/signup

Copilot Chat: https://github.com/github-copilot/chat_waitlist_signup/join

Copilot Labs: https://githubnext.com/projects/copilot-labs/

10.3 Why use Git and GitHub?
Business Analytics and Data Science require version control

LinkedIn

1. Cloud and Distributed Computing
2. Statistical Analysis and Data Mining
3. Middleware and Integration Software
4. Web Architecture and Development Framework
5. User Interface Design
6. Software Revision Control Systems
7. Data Presentation
8. SEO/SEM Marketing
9. Mobile Development
10. Network and Information Security

What is Git?

- A version control system
- Keep track of changes over time and across users
- Avoid emailing different versions of files!
- Think Word track-changes for code / analytics
- “Manage the evolution of files in a project”

What is GitHub (https://github.com)?

- Hosting service for git projects
- Free private repos for assignments and projects

How do your files “evolve” over time?

```
<!DOCTYPE html>
<html>
<body>

<svg height="100" width="100">
  <circle cx="50" cy="50" r="40" stroke="black"
stroke-width="3" fill="red" />
  Sorry, your browser does not support inline SVG.
</svg>

</body>
</html>
```

https://www.youtube.com/watch?v=eWxxfttcMts

What does git help you with?

https://speakerdeck.com/alicebartlett/git-for-humans

By making commits

Alice Bartlett
12:43pm May 8th 2016

Fix Orange

The orange we used fails AAA accessibility contrast tests so beef it up to contrast properly

What does git help you with?

- Git lets you tell the “story” of how your (team’s) project developed
- Use Git to take **snapshots** of files in a folder at a particular point in time
- The folder is called a **repository** or **repo**
- When you take a **snapshot** of a file or files you create a **commit**
- When you **commit** a file, additional information is stored:
  who, when, what, why
- Time-travel
- Backup work to a **remote** server (e.g., GitHub)

https://speakerdeck.com/alicebartlett/git-for-humans

GitHub project (repo) page https://github.com/scikit-learn/scikit-learn

GitHub project (repo) page https://github.com/scikit-learn/scikit-learn

README.rst

Azure Pipelines failed | Cirrus CI passing | codecov 97% | circleci passing | Wheel builder passing

code style black | python 3.8 | 3.9 | 3.10 | pypi v1.3.0 | DOI 10.5281/zenodo.8098905 | Benchmarked by asv

scikit-learn is a Python module for machine learning built on top of SciPy and is distributed under the 3-Clause BSD license.

The project was started in 2007 by David Cournapeau as a Google Summer of Code project, and since then many volunteers have contributed. See the About us page for a list of core contributors.

It is currently maintained by a team of volunteers.

Website: https://scikit-learn.org

Installation

Dependencies

scikit-learn requires:

- Python (>= 3.8)
- NumPy (>= 1.17.3)
- SciPy (>= 1.5.0)
- joblib (>= 1.1.1)

Commits and commit messages

“diffs” show what changes were made in a commit (Commit or History)

Commit

FIX Update pairwise distance function argument names (#26351)

main (#26351)

Browse files

Micky774 committed 13 hours ago Verified

1 parent 07f6586 commit 59048f9

Showing 4 changed files with 52 additions and 52 deletions.

Split Unified

26 sklearn/metrics/_pairwise_distances_reduction/_argkmin_classmode.pyx.tp

```diff
@@ -25,8 +25,8 @@ cdef class ArgKminClassMode{{name_suffix}}(ArgKmin{{name_suffix}}):
25 25     {{name_suffix}}bit implementation of ArgKminClassMode.
26 26     """
27 27     cdef:
28    -     const intp_t[:] class_membership,
29    -     const intp_t[:] unique_labels
   28 +     const intp_t[:] Y_labels,
   29 +     const intp_t[:] unique_Y_labels
30 30         float64_t[:, :] class_scores
31 31         cpp_map[intp_t, intp_t] labels_to_index
32 32         WeightingStrategy weight_type
@@ -38,14 +38,14 @@ cdef class ArgKminClassMode{{name_suffix}}(ArgKmin{{name_suffix}}):
38 38         Y,
39 39         intp_t k,
40 40         weights,
41    -     class_membership,
42    -     unique_labels,
   41 +     Y_labels,
   42 +     unique_Y_labels,
43 43         str metric="euclidean",
44 44         chunk_size=None,
45 45         dict metric_kwargs=None,
46 46         str strategy=None,
47 47     ):
48    -     """Compute the argkmin reduction with class_membership.
   48 +     """Compute the argkmin reduction with Y_labels.
```

See who contributed which line of code using “blame”

Micky774 FIX Update pairwise distance function argument names (#26351) ✕ 59048f9 · 13 hours ago History

Code Blame 188 lines (165 loc) · 6.38 KB Raw

Older Newer Contributors 4

```
4 months ago  PERF Implement PairwiseDist...   1  from cython cimport floating, integral
                                               2  from cython.parallel cimport parallel, prange
                                               3  from libcpp.map cimport map as cpp_map, pair as cpp_pair
                                               4  from libc.stdlib cimport free
                                               5
4 months ago  MAINT Consistent cython typ...   6  from ...utils._typedefs cimport intp_t, float64_t
                                               7
4 months ago  PERF Implement PairwiseDist...   8  import numpy as np
                                               9  from scipy.sparse import issparse
                                              10  from sklearn.utils.fixes import threadpool_limits
                                              11
                                              12  cpdef enum WeightingStrategy:
                                              13      uniform = 0
                                              14      # TODO: Implement the following options, most likely in
                                              15      # `weighted_histogram_mode`
                                              16      distance = 1
                                              17      callable = 2
                                              18
4 months ago  MAINT Consistent cython typ...  19  {{for name_suffix in ["32", "64"]}}
4 months ago  PERF Implement PairwiseDist...  20  from ._argkmin cimport ArgKmin{{name_suffix}}
                                              21  from ._datasets_pair cimport DatasetsPair{{name_suffix}}
                                              22
                                              23  cdef class ArgKminClassMode{{name_suffix}}(ArgKmin{{name_suffix}}):
                                              24      """
                                              25      {{name_suffix}}bit implementation of ArgKminClassMode.
                                              26      """
                                              27      cdef:
13 hours ago  FIX Update pairwise distance f... 28          const intp_t[:] Y_labels,
                                              29          const intp_t[:] unique_Y_labels
4 months ago  MAINT Consistent cython typ...  30          float64_t[:, :] class_scores
                                              31          cpp_map[intp_t, intp_t] labels_to_index
4 months ago  PERF Implement PairwiseDist...  32          WeightingStrategy weight_type
                                              33
```

See who contributed and as what level over time

Standard workflow for a new repo (project) that you own

- Create a repo (project) on GitHub
- Clone the new repo to the ~/**git** directory on your computer
- **Tip 1: DO NOT nest repos**
- **Tip 2: DO NOT put repos on Dropbox , Google Drive, OneDrive, etc.**
- Write, edit, and save code in the project directory (i.e., repo)
- Stage and commit changes locally
- Push changes to GitHub

Standard workflow for “repos” you want to contribute to

1. Fork the repo
2. Clone (Pull / Sync)
3. Write, edit, and save
4. Review and commit changes locally
5. Push changes to GitHub
6. Create a “pull request” (merge request)

10.4 Introduce yourself to Git -- Update Git Gadget to 0.8.0

1. Open a terminal in VS Code using CTRL + ` (or
2. Copy-and-paste the code from the HTML links file in section 10.4

10.4 Git Gadget and Introduce yourself to git

- Open Git Gadget from the launch menu (4 + Enter)
- Provide username (e.g., “rsm-xyz123” based on xyz123@uscd.edu)
- Provide your UCSD email address
- Provide the following as the “Server API:

  https://api.github.com/

- Click on the “Create” button to go to GitHub and create a **Personal Access Token** (see next slide) to paste it in the input box in GitGadget

10.4 Create a Personal Access Token

- Choose RSM-MSBA as the Name for the token
- Set "Expires at" to "No expiration"
- "Scopes" should be set already
- Click "Generate token" button at the bottom of the screen and copy the token to Git Gadget
- Leave the window open!
- Can retrieve the token by running either of the below from a terminal in Jypter or (5 + Enter):
  cat ~/.Rprofile
  cat ~/.rsm-msba/zsh/.zshrc

10.4 Set up an SSH key to conveniently access GitHub

- Click “SSH key” in GitGadget on the ***Introduce*** tab
- Paste the full SSH key printed on screen into the “Key” input box on the GitHub page that will be opened and press “Add new SSH key”
- Use your computer (nick)name as the “title”

10.4 Using GitGadget to Introduce yourself to git

- User type: Student
- Base directory “~/git”
- Click “Introduce” button
- Click “Done”
- Type “**cat ~/.Renviron**” in a (Jupyter) terminal to see if all settings were stored
- Stop the docker container with **q + Enter**, start it again, and go back to GitGadget to see is all settings are listed

10.4 Git commands

- After re-opening GitGadget you should see the info, you provided in the input boxes

- Could also have used a terminal in Jupyter to introduce yourself to git
  - *git config --global --replace-all user.name 'rsm-vnijs'*
  - *git config --global --replace-all user.email 'vnijs@ucsd.edu'*
  - *git config --global init.defaultBranch main*
  - *git config --global pull.rebase false*
  - *git config --global –list*

- Hover over buttons shown in GitGadget to see comments and the code used

10.4 Create a new repository on GitHub by clicking on “+”

10.4 Create a new repository on GitHub

10.4 Create a new repo on GitHub

10.4 Clone the “test1” repo to your computer using terminal in VS Code

Start by creating a directory where you will store ALL of your GitHub repos

Using terminal:

```
mkdir ~/git; cd ~/git;

git clone git@github.com:rsm-xyz123/test1.git;

ls -l test1/;

rm -rf ~/git/test1;
```

10.4 Clone the “test1” repo to your computer using VS Code

10.4 Clone the “test1” repo to your computer using VS Code

GITGADGET (0.7.5)

10.4 Create a new repo from an existing directory

- Open a terminal in VS Code to create a new directory **~/git/test2**
- Copy an existing file to **~/git/test2** (see commands below)

cp ../test1/README.md test2/;

- Use ***VS Code*** to create a repo connected to GitHub using this directory
- In the Command Palette type ”init” to initialize a git repo
- Commit changes to README.md and
- **WARNING**: Never “Create” a repo using ***~/git*** as the “Local directory”! Now your test1 and test2 repos would be nested in the ***~/git*** repo.

10.4 Cleanup (Local and GitHub > Settings > Delete this Repo)

- remove the **test1** and **test2** directories from ***~/git*** on your computer by running the commands below

rm -rf ~/git/test1;

rm -rf ~/git/test2;

- remove the **test1** and **test2** repos from GitHub
- Settings > Delete this Repo (scroll to bottom of page)

10.5 Fork and Clone a project folder for ICT

- Go to *https://github.com/rady-msba/rsm-ict-2023* (**see links document**)
- Click on the **Fork** icon and select the "**your-id**" namespace

- After you forked the repo, click on the **Clone** button and copy the git URL for the **your-id/rsm-ict-2023** repo
- Use VS Code to **Clone** the repo to your laptop

10.6 Submit GitHub account information

GitHub User Information

Provide information about your GitHub account

vnijs@ucsd.edu Switch account

* Indicates required question

Email *

☑ Record **vnijs@ucsd.edu** as the email to be included with my response

Confirm your new GitHub user name. This should be "rsm-" + the first part of your @ucsd.edu email (e.g., rsm-xyz123) *

rsm-vnijs

GitHub Personal Access Token from this URL: *
https://github.com/settings/tokens/new?scopes=repo,workflow&description=RSM-MSBA
The token would look something like the following "ghp_123abc123ABC".

ghp_123abc123ABCASDLKFJALDKJF

A copy of your responses will be emailed to vnijs@ucsd.edu.

Submit

Clear form

10.7 Navigate to the rsm-merge-conflict-practice repo on GibHub

10.7 What is a ”merge conflict” ?

10.7 Edit, Save, Stage, Commit and Push in rsm-group-assignment

- Edit the **merge-conflict.md** file **on github.com**
- Enter ONLY the date, select “Commit to main branch” then enter a Commit message (“GitHub Commit”), and press Commit

10.7 Edit, Save, Stage, Commit and Push in rsm-merge-conflict-practice

10.7 If you “failed to push” you probably forgot to pull before you started

10.7 After a “pull” you now have what is called a “merge conflict”

10.7 Edit the merge-conflict.Rmd file to fix the merge conflict

10.6 Learning more about Git

Resources:

- http://happygitwithr.com
- http://r-pkgs.had.co.nz/git.html
- http://stackoverflow.com/questions/tagged/git or just a google search
- https://try.github.io
- https://www.manning.com/books/git-in-practice
- https://github.com/GitInPractice/GitInPractice#readme

11. ChatGPT+ with Code Interpreter

On the importance of ChatGPT

On the importance of AI Tools

Sign up for ChatGPT+ at https://chat.openai.com

- Got to the listed URL (https://chat.openai.com) and click on “Sign up”
- Then choose “Continue with Google”
- Choose a Workspace account (i.e., owned by UCSD.EDU)

Sign up for ChatGPT+

- Enter your @ucsd.edu email
- Provide your Single Sign-on credentials
- Select “Send me a Push” and confirm authentication through your phone or watch if you have the Duo app installed

Sign up for ChatGPT+

- You should now be logged in to ChatGPT
- On the bottom left of your screen you should see the option to "Upgrade to Plus"
- Provide payment information in the form shown on the right to Subscribe

Must use your @ucsd.edu email address to qualify for fellowship

Sign up for ChatGPT+

- Click on your Name on the bottom left of the screen and select “My plan”
- Then select “Manage my subscription”
- Create a screenshot like the one on the right that provides proof-of-purchase and post that to …

Sign up for ChatGPT+ and get a $100 fellowship

Fall 2023 MSBA ChatGPT+ Fellowship

Submit this form to request a one-time $100 fellowship for the cost associated with the ChatGPT+ subscription. **Deadline August 1, 2023**

vnijs@ucsd.edu Switch account

The name and photo associated with your Google account will be recorded when you upload files and submit this form. Your email is not part of your response.

* Indicates required question

First (given) Name: *

Your answer

Last (family) Name: *

Your answer

Student PID *(A12345678)*: *

Your answer

MSBA cohort *

- ◯ Full Time MSBA 2024 (Incoming Summer 2023)
- ◯ Flex MSBA 2024 (Incoming Summer 2023)
- ◯ Flex MSBA 2023 (Incoming Summer 2022)

As a recipient of the fellowship, I understand that I am expected to maintain a ChatGPT+ subscription through December 2023 at $20/month for a total of $100 USD. I understand that the Rady School of Management is supporting the subscription to enhance my academic and professional development throughout the duration of the fellowship. *

- ☐ I agree to these terms and accept the ChatGPT+ Fellowship
- ☐ I decline to these terms and do not accept the ChatGPT+ Fellowship

Attach proof of purchase of ChatGPT+ in PDF format (Last name, First Name - ChatGPT payment) *

Add file

Submit

Clear form

ChatGPT+ Site License?

We are also working on a new ChatGPT Business subscription for professionals who need more control over their data as well as enterprises seeking to manage their end users. ChatGPT Business will follow our API's data usage policies, which means that end users' data won't be used to train our models by default. We plan to make ChatGPT Business available in the coming months.

https://openai.com/blog/new-ways-to-manage-your-data-in-chatgpt

Add Custom Instructions

Key step: Activate Code Interpreter

Add a bookmark to:

https://chat.openai.com/?model=gpt-4-code-interpreter

12. Rstudio

12.1 Learning to use Rstudio and Rstudio Projects

- Rstudio essentials
  - https://www.rstudio.com/resources/webinars/rstudio-essentials-webinar-series-part-1/
  - Writing Code in Rstudio
    - Rstudio panes
      - Console
      - Environment, History
      - Editor, View
      - Files, Plots, Packages, Help, …
    - Cheatsheets!
    - Tab completion & Help
- Rstudio projects
  - https://www.rstudio.com/resources/webinars/rstudio-essentials-webinar-series-managing-change-part-1/

13 Installing Python and R packages

12.3 Install Python packages “locally” using pip (py-packages.ipynb)

13.1 Create a new (local) python environment using Conda

"Conda is **a cross platform package and environment manager that installs and manages conda packages from the Anaconda repository as well as from the Anaconda Cloud**. Conda packages are binaries."

- The docker container is built on top of the Jupyter "pyspark-notebook" notebook (offers environment that is compiled for M1 and M2 macs)
- Conda was used to set up a "base" environment with the python packages that you will likely need for the Rady MSBA
- What if there is a package that that is not already available? Can be added to the RSM-JUPYTER image if it is used by many (most) Rady MSBA students
- Convenience functions are available to make creating and managing your own environments (kernels) a bit easier

ANACONDA.

13.1 Create a new (local) python environment using Conda

ccenv ("conda create", cat /usr/local/bin/cc) to create new environment (or add pkgs)

ccenv myenv package_a package_b ***ccenv myenv other_package_c***

cl ("conda list", cat /usr/local/bin/cl) to list available environments

cr ("conda remove", cat /usr/local/bin/cr) to remove an environment

cr myenv

ce ("conda export", cat /usr/local/bin/ce) to export an environment

ce myenv myenv.yaml

ci ("conda import", cat /usr/local/bin/ci) to import an environment

ci myenv.yaml myenv

ANACONDA.

13.1 Add local python packages using pip

Tip: Use *pip* to install any additional packages

Tip: If you need a more restrictive / controlled environment for a specific project. Use Conda to create it

13.2 Installing R packages “locally”

13.3 Installing “in” the container and committing changes

```
conda activate base
conda install package_a package_b
pip install package_c
sudo R install.packages("fortunes")
```

- These changes will be gone if you restart the container so you can “play-around” as much as you want without “breaking” anything

- To keep (i.e., commit) changes press c (and Enter) and follow the prompts
- To share your custom version with others, use Docker Hub (dockerhub.com)

13.4 Removing packages that you installed

```
~/Desktop/launch-rsm-msba-arm.command
--------------------------------------------------------------------------------
Starting the rsm-msba-arm computing environment on macOS (ARM64)
Version   : 2.7.0
Build date: 2023-06-29
Base dir. : /Users/vnijs
Cont. name: rsm-msba-arm
--------------------------------------------------------------------------------
Press (1) to show Jupyter Lab, followed by [ENTER]:
Press (2) to show Rstudio, followed by [ENTER]:
Press (3) to show Radiant, followed by [ENTER]:
Press (4) to show GitGadget, followed by [ENTER]:
Press (5) to show a (ZSH) terminal, followed by [ENTER]:
Press (6) to update the rsm-msba-arm container, followed by [ENTER]:
Press (7) to update the launch script, followed by [ENTER]:
Press (8) to clear Rstudio sessions and packages, followed by [ENTER]:
Press (9) to clear local Python packages, followed by [ENTER]:
Press (10) to start a Selenium container, followed by [ENTER]:
Press (h) to show help in the terminal and browser, followed by [ENTER]:
Press (c) to commit changes, followed by [ENTER]:
Press (q) to stop the docker process, followed by [ENTER]:
--------------------------------------------------------------------------------
Note: To start, e.g., Jupyter on a different port type 1 8991 [ENTER]
Note: To start a specific container version type, e.g., 6 2.7.0 [ENTER]
Note: To commit changes to the container type, e.g., c myversion [ENTER]
--------------------------------------------------------------------------------
```

- Press 8 to cleanup Rstudio settings and remove R-packages
- Press 9 to remove python packages
- Or from a terminal in the docker container, use “clean”
- Use “**cr myenv**” to remove a conda environment you added