DSCI351-351m-451: Class 01a, (CWRU, Pitt, UCF, UTRGV)

Profs: R. H. French, L. S. Bruckman, P. Leu, K. Davis, S. Cirlos

TAs: W. Oltjen, K. Hernandez, M. Li, M. Li, D. Colvin

19 October, 2022

Contents

1.1.1.1 Class Readings, Assignments, Syllabus Topics

1.1.1.1.1 Reading, Lab Exercises, SemProjects

- Readings:
  - For today: None
  - For next class: R4DS17-21
- Laboratory Exercises:
  - LE4 due tonight!
  - LE5 due 11/10
- Office Hours: (Class Canvas Calendar for Zoom Link)
  - Wednesday @ 4:00 PM to 5:00 PM, Will Oltjen
  - Saturday @ 3:00 PM to 4:00 PM, Kristen Hernandez
  - **Office Hours are on Zoom, and recorded**
- Semester Projects
  - DSCI 451 Students Biweekly Update 1 Due
  - DSCI 451 Students
    - Next **Update 4 is due Friday**
- Exams
  - Final: Monday December 19, 2022, 12:00PM - 3:00PM, Nord 356 or remote

1.1.1.2 Textbooks

- Peng: R Programming for Data Science
- Peng: Exploratory Data Analysis with R
- Open Intro Stats, v4
- Wickham: R for Data Science
- Hastie: Intro to Statistical Learning with R, 2nd Ed.

Introduction to R and Data Science

- For R, Coding, Inferential Statistics
  - Peng: R Programming for Data Science
  - Peng: Exploratory Data Analysis with R

Textbooks for this class

- OIS = Diez, Barr, Çetinkaya-Runde: Open Intro Stat v4
- R4DS = Wickham, Grolemund: R for Data Science

Textbooks for DSCI353/353M/453, And in your Repo now

- ISLR2 = James, Witten, Hastie, Tibshirani: Intro to Statistical Learning with R 2nd Ed.
- ESL = Trevor Hastie, Tibshirani, Friedman: Elements of Statistical Learning
- DLwR = Chollet, Allaire: Deep Learning with R

Magazine Articles about Deep Learning

- DL1 to DL13 are "Deep Learning" articles in 3-readings/2-articles/

1.1.1.3 Syllabus

1.1.1.4 For the DSCI 451 students they have an EDA SemProj to do

- SemProjects:
  - SemProjects have a bi-weekly progress update
    - due Friday's at 11:59 pm (6 updates)
  - Each update should be made in the report template
    - found in the Repo with each update filled out
  - SemProj Report Out #1 Class W5, (recorded 10 min presentation)
    - Peer Grading by All DSCI 351/351m/451 students due a week later
  - SemProj Report Out #2 in Class W9 (recorded 10 min presentation)
    - Peer Grading by All DSCI 351/351m/451 students due a week later
  - SemProj Report Out #3 in Class W13 (recorded 10 min presentation)
    - Peer Grading by All DSCI 351/351m/451 students due a week later
  - SemProj Report is full comprehensive written project
    - (report template updated from each report)
    - **due Friday 12-11-2021**
- Assistance on SemProjects is done with DSCI352-352m-452 Class
  - SemProj's are taught by Prof. Laura Bruckman
  - SemProject office hours 9-10 am on Tuesdays

1.1.1.4.1 Care should be taken when choosing SemProj datasets.

- Report Out 1 focuses on
  - Explaining the 'why' of your research project
  - Describing your dataset
  - Presenting an analysis plan
  - Cleaning your data
- Report Out 2 focuses on:

Day:Date	Foundation	Practicum	Reading	Due
w01a:Tu:8/30/22	ODS Tool Chain	R, Rstudio, Git		
w01b:Th:9/1/22	Setup ODS Tool Chain	Bash, Git, Slack, Agile	PRP4-33	LE1
w02a:Tu:9/6/22	Bash-Git-Knuth-Lit.Prog.	RIntroR	PRP35-64	
w02b:Th:9/8/22	What is Data Science	OIS:Intro2R	OIS1,2	
w02Pr:Fr:9/9/22			PRP65-93	**451 Update1**
w03a:Tu:9/13/22	Data Intro	Data Analytic Style	PRP94-116	LE2 **LE1 Due**
w03b:Th:9/15/22	Rand. Var. Normal Dist.	Git, Rmds, Loops	OIS4	
w04a:Tu:9/20/22	Tidy Check Explore	Tidy GapMinder	EDA1-31	
w04b:Th:9/22/22	Inference, DSCI Process	Other Distrib. 7 ways	R4DS1-3	LE3 **LE2 Due**
w04Pr:Fr:9/23/22			EDA32-58	**451 Update2**
w05a:Tu:9/27/22	OIS4 Rand. Var.	EDA of PET Degr.	OIS5	
w05b:Th:9/29/22	OIS5 Found. of Infer.	Multivar Corr. Plot	R4DS4-6	
w05Pr:Fr:9/30/22				**451 RepOut1**
w06a:Tu:10/4/22	Pred., Algorithm, Model		R4DS7-8	
w06b:Th:10/6/22	Summ. Stats & Vis.	Anscombe's Quartets	R4DS9-16	LE4 **LE3 Due**
w06Pr:Fr:10/7/22				**451 Update3**
w07a:Tu:10/11/22	Midterm Rev. Tidy Data	Correl Plots Summ Stats	OIS6.1-2	**PeerRv1 Due**
w07b:Th:10/13/22	HypoTest, Infer. Recap	Penguin EDA, Sampling		
w08a:Tu:10/18/22	**MIDTERM**	**EXAM**		
w08b:Th:10/20/22	Programming & Coding	Code Packaging		**LE4 Due**
w08Pr:Fr:10/21/22				**451 Update4**
Tu:10/24,25	**CWRU**	**FALL BREAK**	R4DS17-21	
w09b:Th:10/27/22	Cat. Inf. 1 & 2 propor.	Indep. Test,2-way tables	OIS6.3-4	LE5
w09Pr:Fr:10/28/22				**451 RepOut2**
w10a:Tu:11/1/22	Goodness of Fit, χ^2 test	t-tests 1&2 means	OIS7.1-4	
w10b:Th:11/3/22	Num. Infer, Cont. Tables	Stat. Power		
w10Pr:Fr:11/4/22				**451 Update5**
w11a:Tu:11/8/22	Sample & Effect Size	Stat. Power GGmap	OIS8	**PeerRv2 Due**
w11b:Th:11/10/22	Inf. 4 Regr, Test & Train	Curse of Dimen.	ISLR1,2.1,2	LE6 **LE5 Due**
w12a:Tu:11/15/22	Lin. Regr. Part 1	Residuals	OIS9	
w12b:Th:11/17/22	Lin. Regr. Part 2	Regr. Diagnostics		
w12Pr:Fr:11/18/22				**451 Update6**
w13a:Tu:11/22/22	Mult. Lin. Regr.	Var. & Mod. Selec.,	ISLR3.1	LE7 **LE6 due**
w13b:Th:11/24/22	Log. Regr.	GIS Trends	ISLR3.2	
w13Pr:Fr:11/25/22				**451 RepOut3**
w14a:Tu:11/23/22	Classificat., Sup. Lrning	Caret, Broom 4 modeling	ISLR4.1-3	
Th,Fr:11/24,25	**THANKSGIVIING**	**Vacation**		
w15a:Tu:11/29/22		Clustering		**PeerRv3 Due**
w15b:Th:12/1/22	Big Data Analytics	Dist. Comp., Hadoop		
w15SPr:Fr:12/2/22		Read Article by	Mirletz,2015	
w16a:Tu:12/6/22	Final Exam Review			
w15b:Th:12/8/22				**LE7 due**
Friday 12/12	**SemProj**	**Final Report**		**SemProj4 due**
Monday 12/19	**FINAL EXAM**	**12:00-3:00pm**	Nord 356	or remote

Figure 1: DSCI351-351M-451 Syllabus

  - EDA of your data
  - Visualizing your data
  - Further cleaning of your data
  - Reevaluation of your data analysis plan (Do you need more data?)
- Report Out 3:
  - More data visualization
  - Initial modeling
  - Conclusions about your data
  - Were you able to answer your why question?
  - What else would you need to do to get to understanding your data better?

1.1.1.5 Tidyverse Cheatsheets, Functions and Reading Your Code

- Look at the Tidyverse Cheatsheet
  - **Tidyverse For Beginners Cheatsheet**
    - In the Git/20s-dsci353-353m-453-prof/3-readings/3-CheatSheets/ folder
  - **Data Wrangling with dplyr and tidyr Cheatsheet**

  Tidyverse Functions & Conventions

  - The pipe operator `%>%`
  - Use `dplyr::filter()` to subset data row-wise.
  - Use `dplyr::arrange()` to sort the observations in a data frame
  - Use `dplyr::mutate()` to update or create new columns of a data frame
  - Use `dplyr::summarize()` to turn many observations into a single data point
  - Use `dplyr::arrange()` to change the ordering of the rows of a data frame
  - Use `dplyr::select()` to choose variables from a tibble,
    - keeps only variables you mention
  - Use `dplyr::rename()` keeps all the variables and renames variables
    - rename(iris, petal_length = Petal.Length)
  - These can be combined using `dplyr::group_by()`
    - which lets you perform operations "by group".
  - The `%in%` matches conditions provided by a vector using the c() function
  - The **forcats** package has tidyverse functions
    - for factors (categorical variables)
  - The **readr** package has tidyverse functions
    - to read_…, melt_… col_…, parse_… data and objects

Reading Your Code: Whenever you see

- The assignment operator `<-`, think **"gets"**
- The pipe operator, `%>%`, think **"then"**

1.1.1.6 Topic

1.1.1.6.1 Python Packages in R

2 Python in RMarkdown/RStudio

In the debate between Python and R, there moments where either side will excel. - In these cases, it can be fruitful - to use Python code snippets in R scripts or vice versa.

In this .Rmd, we will demonstrate how Python can be used in R. - and how to publish a Python package

2.1 Reticulate Package

"The reticulate package provides a comprehensive set of tools - for interoperability between Python and R." 1

Core functions include:

- Calling Python from R in a variety of ways:
  - R Markdown
  - sourcing Python scripts
  - importing Python modules
  - using Python interactively within an R session. 1
- Translation between R and Python objects
  - (for example, between R and Pandas data frames, or between R matrices and NumPy arrays. 1

```
# import reticulate
library(reticulate)

# set console messages off
options(reticulate.repl.quiet = TRUE)
```

In most environments this will be enough to start using Python.

In our ODS Desktops, we need to set the Python path explicitly - by editing the R profile.

Run the command below and add the following to the file: - RETICULATE_PYTHON="C:/Python/Python-3.10.4/Scripts/python.exe"

```
# edit the r profile file
usethis::edit_r_profile()
```

```
## * Edit '/home/wco3/.Rprofile'
## * Restart R for changes to take effect
```

2.2 Reticulate Basics

Once reticulate is imported, it is as easy at setting the chunk to use python with {python}.

```
# set value of a
a = "Hello" + " World"
print(a)
```

```
## Hello World
```

Note: the variables created in your Python environment will not be contained in your R environment.

```
# check if a exists
exists('a')
```

```
## [1] FALSE
```

To get around this, we can pass the variable from one environment to another.

```
# return value of a by calling py environment
py$a
```

```
## [1] "Hello World"
```

Likewise with R:

```
# set value of b
b <- 5
```

```
# return value of b by calling R env
r.b
```

```
## 5.0
```

2.3 Reticulate Libraries

Basic variable manipulation is not the only Python feature available.

More advanced Python can be leveraged with the ability to import Python libraries.

```
# import os library
import os

# get current working directory
os.getcwd()
```

```
## '/mnt/rstor/CSE_MSE_RXF131/cradle-members/sdle/wco3/sdle_repo/22f-dsci351-451-prof/2-class'
```

Reticulate can also be used outside of .Rmd files where you can specify the cell language.

It can be run in R scripts as well.

Here is a sample of how a Python library would be called and used in an R script

```
# import os library
os <- import("os")

# get current working directory
os$getcwd()
```

```
## [1] "/mnt/rstor/CSE_MSE_RXF131/cradle-members/sdle/wco3/sdle_repo/22f-dsci351-451-prof/2-class"
```

These libraries can be leveraged to do classic Python manipulations.

```
# import numpy (specify no automatic Python to R conversion)
np <- import("numpy", convert = FALSE)

# create numpy array of 1-4
a <- np$array(c(1:4))

# apply cumulative sum to array
sum <- a$cumsum()

# convert object to R
py_to_r(sum)
```

```
## [1]  1  3  6 10
```

2.4 Reticulate Functions

One can design and run a function in Python as well.

```
pyFunction <- "def print_message():
                 print('Hello world!')"

py_run_string(pyFunction)

py$print_message()
```

One can even write a Python script into a .py file then run the script using reticulate

```
py_run_file("2208-mds-rely-bootcamp-reticulate-script.py")
```

```
## Error in py_run_file_impl(file, local, convert): Unable to open file '2208-mds-rely-bootcamp-reticula
```

```
library(ggplot2)

df <- read.csv('./data/model_results.csv')
```

```
## Warning in file(file, "rt"): cannot open file './data/model_results.csv': No
## such file or directory
```

```
## Error in file(file, "rt"): cannot open the connection
```

```
step_range <- 1:nrow(df)
```

```
## Error in 1:nrow(df): argument of length 0
```

```
percent_range <- (0:10) / 10

df$X <- step_range
```

```
## Error in eval(expr, envir, enclos): object 'step_range' not found
```

```
colors <- c("training" = "red", "validation" = "blue")

ggplot(df, aes(x=X)) +
  geom_line(aes(y = accuracy), color = "red") +
  geom_line(aes(y = val_accuracy), color="blue", linetype="twodash") +
  labs(color = "Legend") +
  scale_color_manual(values = colors)
```

```
## Error in `ggplot()`:
## !   You're passing a function as global data.
##   Have you misspelled the `data` argument in `ggplot()`
```

```
ggplot(df, aes(x=X)) +
  geom_line(aes(y = loss), color="red") +
  geom_line(aes(y = val_loss), color="blue", linetype="twodash") +
  labs(color = "Legend") +
  scale_color_manual(values = colors)
```

```
## Error in `ggplot()`:
## !   You're passing a function as global data.
##   Have you misspelled the `data` argument in `ggplot()`
```

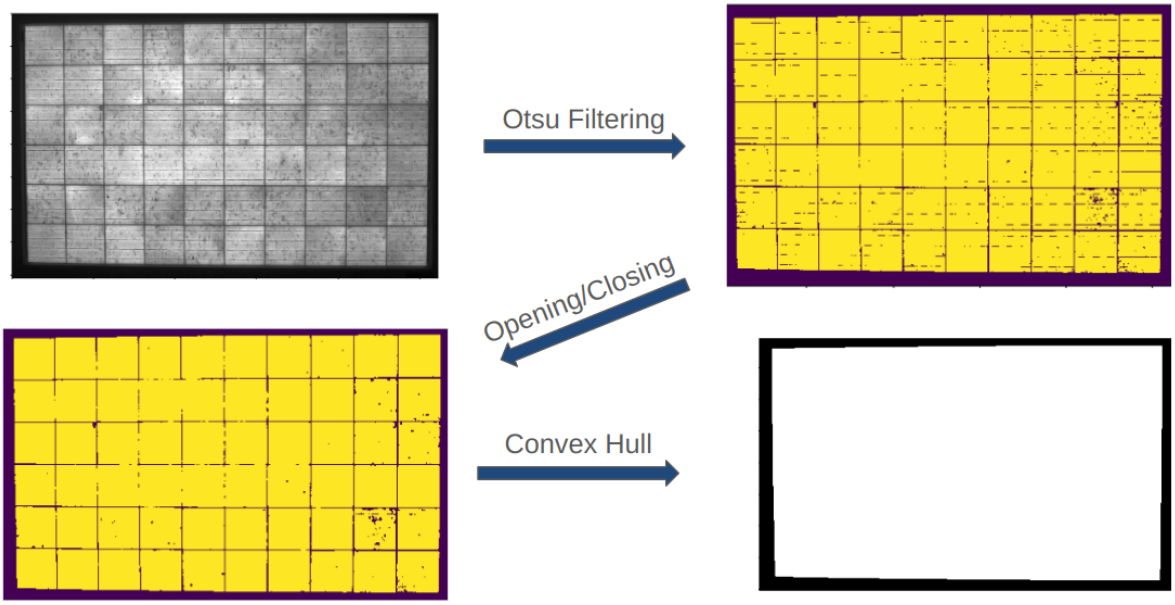
2.4.1 Python Packages

Now that we know how Python and R can work together - we can discuss how to make Python packages

We have a number of published Python packages in our lab - one of which is PVimage

PVimage https://pypi.org/project/pvimage/ - is a package designed for the analysis of images in solar - we can do automated segmentation of images - data cleaning! - then automated analysis of those segmented images

PVImage Image Processing

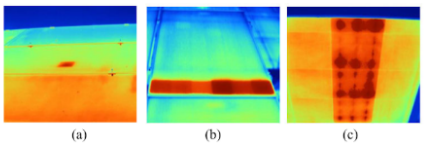


Neural Network Models Applied in PV Reliability Studies

Image characterizations for PV reliability study

Manually checking is time-consuming

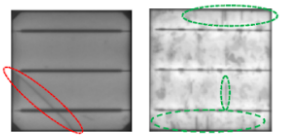
Large datasets -> machine learning models



CNN (convolution neural network) Models

Classification

- Binary: Defective or defect-free
- Multi-class: based on kinds of defects



Regression -> predict change in electrical performance

- Define image features -> polynomial fitting (${}^{n}P_{mp,IV}$ & ${}^{n}R_{s,IV}$)
- CNN to predict electrical features, with help from generative adversarial networks (GAN) to construct a fake defective-free image

Temporal relation in stepwise measurement is not utilized, helpful?

In current published NN models for PV reliability studies

RNN (Recurrent neural network) models

- Power forecast & prediction

Within the package, we've designed pipelines:

https://pvimage-doc.readthedocs.io/en/latest/pipelines.html

- with a number of functions that are called by each pipeline
  - so that the code is streamlined for users
  - each function is used together in a string

We have documentation published online

- to explain how to use our package
- with examples written out in code

We publish our code and documentation to Pypi

- The Python package index
- So that anyone can find and use our packages

In order to publish a Python package

- You need to use the pip package
  - Python package installer

Python packages are published much like R ones

- except it is much easier to get Python packages published
- since CRAN is much more specific about standards
  - this is good for R
    - Python has had issues with malware installed in packages due to poor standards

To publish a Python package

- You need to make a setup.py file in your main package directory
- Fill it with
  - Licence
  - Title
  - Version
  - Author
  - Dependencies
  - Github URL
- And you should be good to go!

Here's a decent guide for how to do this https://towardsdatascience.com/how-to-upload-your-python-package-to-pypi-de1b363a1b3

2.5 Acknowledgements

1 R Interface to Python. Interface to Python • reticulate. (n.d.). Retrieved April 12, 2022, from https://rstudio.github.io/reticulate/

2.5.0.1 Links