
Leveraging Data Science to Facilitate Insightful, Reproducible, and Trustworthy I-O

SIOP 2019
Thursday, April 4th
1:30 PM to 2:50 PM
Room Maryland 4-6

Today's Tutorial

- Purpose:
 - Demonstrate how data science techniques can benefit IO Psychology:
 - Improved reproducibility of research
 - Better insights into human behavior at work
 - Increase credibility and impact of research
- Intended audience:
 - Already have working knowledge of R
 - Interested in using data science techniques
- Available materials: **<http://bit.ly/SIOPDataScienceTutorial>**

Presenters

- **Rachel Callan, PhD**
 - People Scientist at Humu
- **Andrew Collmus, MS**
 - People Data Scientist at Flex
 - Graduate Student at Old Dominion University
- **Elena Auer, MS**
 - Graduate Student at University of Minnesota
- **Sebastian Marin**
 - Graduate Student at University of Minnesota
- **Richard Landers, PhD**
 - Associate Professor, John P. Campbell Distinguished Professor of Industrial and Organizational Psychology at University of Minnesota

What to expect in this session

What we will cover:

- Accessible demonstration for later implementation:
 - Open source statistical tools, version control, and code repositories
 - Data sourcing and analytical techniques
- How these methods can improve IO research

What we won't cover:

- Installation of R/Py
- Basic coding in R
- Deep dive into specific data science techniques

How to follow along

- Live demonstration:
 - Connect to Twitter API to source data
 - Process and clean text data
 - Perform some analysis on that data
- Feel free to follow along, but it's not necessary
 - Ask questions as we go along; resources are also available
- After the tutorial:
 - Demonstrations can be reproduced with materials on GitHub
 - Deeper discussion can be found in book chapter
 - Additional resources provided



**Learning Objective #1: Increase the
reproducibility of your current and
future research projects**

Open-source Technologies

- Free (compare to **Many Pricey Languages Used for SEM** that cost 100s of \$\$\$)
- Customizable
- Excellent Documentation
- Active Community Support
- Reproducible Workflow

Open-source Technologies - **Free**

Free - compared to
Some Profitable
Statistical Software
that costs 100s of \$\$\$)



Accessible - to
other people and
your future self



Open-source Technologies - Customizable

- Write your own functions
- Modify someone else's functions



$$d_M(x, y) = \sqrt{(x - y)^T S^{-1} (x - y)}$$

$$= \sqrt{[x_1 - y_1 \quad x_2 - y_2] \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 \\ 0 & \frac{1}{\sigma_2^2} \end{bmatrix} [x_1 - y_1 \quad x_2 - y_2]}$$

$$= \sqrt{\left[\frac{x_1 - y_1}{\sigma_1^2} \quad \frac{x_2 - y_2}{\sigma_2^2} \right] [x_1 - y_1 \quad x_2 - y_2]}$$

$$= \sqrt{\frac{(x_1 - y_1)^2}{\sigma_1^2} + \frac{(x_2 - y_2)^2}{\sigma_2^2}}$$

Open-source Technologies - Excellent Documentation

Important Statsy Info

Hyperlinked Citation

The screenshot shows the scikit-learn documentation for the Naive Bayes section. The top navigation bar includes links for Home, Installation, Documentation, Examples, Google Custom Search, and a search icon. The main content area is titled "1.9. Naive Bayes". It explains that Naive Bayes methods are based on Bayes' theorem with the "naive" assumption of conditional independence between features. The formula for the joint probability is given as:

$$P(y | x_1, \dots, x_n) = \frac{P(y) P(x_1, \dots, x_n | y)}{P(x_1, \dots, x_n)}$$

Using the naive conditional independence assumption that

$$P(x_i | y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i | y),$$

for all i , this relationship is simplified to

$$P(y | x_1, \dots, x_n) = \frac{P(y) \prod_{i=1}^n P(x_i | y)}{P(x_1, \dots, x_n)}$$

Since $P(x_1, \dots, x_n)$ is constant given the input, we can use the following classification rule:

$$P(y | x_1, \dots, x_n) \propto P(y) \prod_{i=1}^n P(x_i | y)$$
$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^n P(x_i | y),$$

and we can use Maximum A Posteriori (MAP) estimation to estimate $P(y)$ and $P(x_i | y)$: the former is then the relative frequency of class y in the training set.

The different naive Bayes classifiers differ mainly by the assumptions they make regarding the distribution of $P(x_i | y)$.

In spite of their apparently over-simplified assumptions, naive Bayes classifiers have worked quite well in many real-world situations, famously document classification and spam filtering. They require a small amount of training data to estimate the necessary parameters. (For theoretical reasons why naive Bayes works well, and on which types of data it does, see the references below.)

Naive Bayes learners and classifiers can be extremely fast compared to more sophisticated methods. The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one dimensional distribution. This in turn helps to alleviate problems stemming from the curse of dimensionality.

On the flip side, although naive Bayes is known as a decent classifier, it is known to be a bad estimator, so the probability outputs from `predict_proba` are not to be taken too seriously.

References:

- H. Zhang (2004). The optimality of Naive Bayes. Proc. FLAIRS.

Open-source Technologies - Community Support

Question I asked

https://stackoverflow.com/questions/51950944/propagate-conditional-column-value-in-pandas

stack overflow Search...

Home PUBLIC Stack Overflow Tags Users Jobs Teams Q&A for work Learn More

propagate conditional column value in pandas Ask Question

I want to create an indicator variable that will propagate to all rows with the same customer-period value pair as the indicator. Specifically, if `baz` is `yes`, I want all rows of that same customer and period email to show my indicator.

6 df

Customer	Period	Question	Score
A	1	foo	2
A	1	bar	3
A	1	baz	yes
A	1	biz	1
B	1	bar	2
B	1	baz	no
B	1	qux	3
A	2	foo	5
A	2	baz	yes
B	2	baz	yes
B	2	biz	2

I've tried

```
df['Indicator'] = np.where(
    (df.Question.str.contains('baz') & (df.Score == 'yes')),
    1, 0)
```

which returns

Customer	Period	Question	Score	Indicator
A	1	foo	2	0
A	1	bar	3	0
A	1	baz	yes	1
A	1	biz	1	0
B	1	bar	2	0
B	1	baz	no	0
B	1	qux	3	0
A	2	foo	5	0
A	2	baz	yes	1
B	2	baz	yes	1
B	2	biz	2	0

But this is the desired output:

Customer	Period	Question	Score	Indicator
A	1	foo	2	1
A	1	bar	3	1
A	1	baz	yes	1
A	1	biz	1	1
B	1	bar	2	0
B	1	baz	no	0
B	1	qux	3	0
A	2	foo	5	1
A	2	baz	yes	1

674 3 21

asked 7 months ago viewed 131 times active 7 months ago

BLOG The Next CEO of Stack Overflow

FEATURED ON META Updating the Hot Network Questions List - now with a bit more network and a... Should we burninate the [like] tag? 2019 Community Moderator Election Results The Ask Question Wizard is Live!

Love this site?

Get the weekly newsletter! In it, you'll get:

- The week's top questions and answers
- Important community announcements
- Questions that need answers

Sign up for the newsletter see an example newsletter

Related

5065 Does Python have a ternary conditional operator?

3326 How do I sort a dictionary by value?

706 Selecting multiple columns in a pandas dataframe

1323 Renaming columns in pandas

743 Adding new column to existing DataFrame in Python pandas

940 Delete column from pandas DataFrame by

Open-source Technologies - Community Support

Answer from Community

The screenshot shows a Stack Overflow question page with the URL <https://stackoverflow.com/questions/51950944/propagate-conditional-column-value-in-pandas>. The question is about propagating conditional column values in pandas. The accepted answer, which is the focus of the screenshot, uses the following code:

```
In [954]: df['Indicator'] = (df.assign(eq=df.Question.eq('baz') & df.Score.eq('yes'))
                           .groupby(['Customer', 'Period'])['eq']
                           .transform('any').astype(int))
```

The code creates a new column 'Indicator' based on the 'Question' and 'Score' columns, grouped by 'Customer' and 'Period'. It checks if both 'Question' and 'Score' are 'baz' and 'yes' respectively, and then applies the 'any' function across the groups to create a binary indicator.

Below the code, a table shows the resulting DataFrame:

	Customer	Period	Question	Score	Indicator
0	A	1	foo	2	1
1	A	1	bar	3	1
2	A	1	baz	yes	1
3	A	1	biz	1	1
4	B	1	bar	2	0
5	B	1	baz	no	0
6	B	1	qux	3	0
7	A	2	foo	5	1
8	A	2	baz	yes	1
9	B	2	baz	yes	1
10	B	2	biz	2	1

At the bottom of the page, there is a section titled "Details" with the following code:

```
In [956]: df.Question.eq('baz') & df.Score.eq('yes')
Out[956]:
0    False
1    False
2     True
3    False
4    False
5    False
6    False
7    False
```

Open-source Technologies - **Reproducible** Workflow

Can you make this?



Open-source Technologies - Reproducible Workflow

Not without this

Ingredients

4 ounces German sweet chocolate, chopped	1-1/2 cups sugar
1/2 cup water	1-1/2 cups evaporated milk
1 cup butter, softened	3/4 cup butter
2 cups sugar	5 large egg yolks, beaten
4 large eggs, separated	2 cups sweetened shredded coconut
1 teaspoon vanilla extract	1-1/2 cups chopped pecans
2-1/2 cups cake flour	1-1/2 teaspoons vanilla extract
1 teaspoon baking soda	ICING:
1/2 teaspoon salt	1 teaspoon shortening
1 cup buttermilk	2 ounces semisweet chocolate

FROSTING:

Directions

- 1 Line three greased 9-in. round baking pans with waxed paper. Grease waxed paper and set aside. In small saucepan, melt chocolate with water over low heat; cool.
- 2 Preheat oven to 350°. In a large bowl, cream butter and sugar until light and fluffy. Beat in 4 egg yolks, one at a time, beating well after each addition. Blend in melted chocolate and vanilla. Combine flour, baking soda and salt; add to the creamed mixture alternately with buttermilk, beating well after each addition.
- 3 In a small bowl and with clean beaters, beat the 4 egg whites until stiff peaks form. Fold a fourth of the egg whites into creamed mixture; fold in remaining whites.
- 4 Pour batter into prepared pans. Bake 24-28 minutes or until a toothpick inserted in center comes out clean. Cool 10 minutes before removing from pans to wire racks to cool completely.
- 5 For frosting, in a small saucepan, heat sugar, milk, butter and egg yolks over medium-low heat until mixture is thickened and golden brown, stirring constantly. Remove from heat. Stir in

<https://www.tasteofhome.com/recipes/german-chocolate-cake>

Syntax, Pipelines, and Versioning

- Syntax creates a self-documenting reproducible workflow
- Pipelines document the journey of the data
- Versioning keeps your directory clean and documents all changes

Syntax, Pipelines, and Versioning

- Syntax creates a self-documenting reproducible workflow
 - Every action is recorded, and replicable

Python

```
In [286]: df = pd.read_csv("ds_ch.csv", sep=',')
....: X = df.iloc[:, :-1] # get features, which make up first n columns of dataframe
....: y = df.iloc[:, -1] # get criterion, which is the last column of the dataframe
....: X_sm = sm.add_constant(X) # fits an intercept
....: sm_model = sm.OLS(y, X_sm).fit()
....: sm_model.summary()
```

R

```
> df <- read.csv('ds_ch.csv')
> names(df) <- c(paste0('X', 1:10), 'y')
>
> Rmodel <- lm(y~., data=df)
> summary(Rmodel)
```

Syntax, Pipelines, and Versioning

Python

```
Out[286]:  
<class 'statsmodels.iolib.summary.Summary'>  
"""  
    OLS Regression Results  
=====-----  
Dep. Variable:          0.1   R-squared:           0.518  
Model:                 OLS   Adj. R-squared:      0.507  
Method:                Least Squares   F-statistic:        46.27  
Date:      Wed, 21 Mar 2018   Prob (F-statistic): 3.83e-62  
Time:      14:04:38   Log-Likelihood:     -2386.0  
No. Observations:      442   AIC:                  4794.  
Df Residuals:          431   BIC:                  4839.  
Df Model:                   10  
Covariance Type:       nonrobust  
=====-----  
            coef    std err        t      P>|t|      [0.025      0.975]  
-----  
const    152.1335    2.576    59.061      0.000    147.071    157.196  
0       -10.0122    59.749    -0.168      0.867    -127.448    107.424  
1       -239.8191    61.222    -3.917      0.000    -360.151    -119.488  
2       519.8398    66.534     7.813      0.000     389.069    650.610  
3       324.3904    65.422     4.958      0.000     195.805    452.976  
4       -792.1842   416.684    -1.901      0.058    -1611.169     26.801  
5       476.7458    339.035     1.406      0.160    -189.621    1143.113  
6       101.0446    212.533     0.475      0.635    -316.685     518.774  
7       177.0642    161.476     1.097      0.273    -140.313    494.442  
8       751.2793    171.902     4.370      0.000     413.409    1089.150  
9       67.6254    65.984     1.025      0.306    -62.065     197.316  
=====-----
```

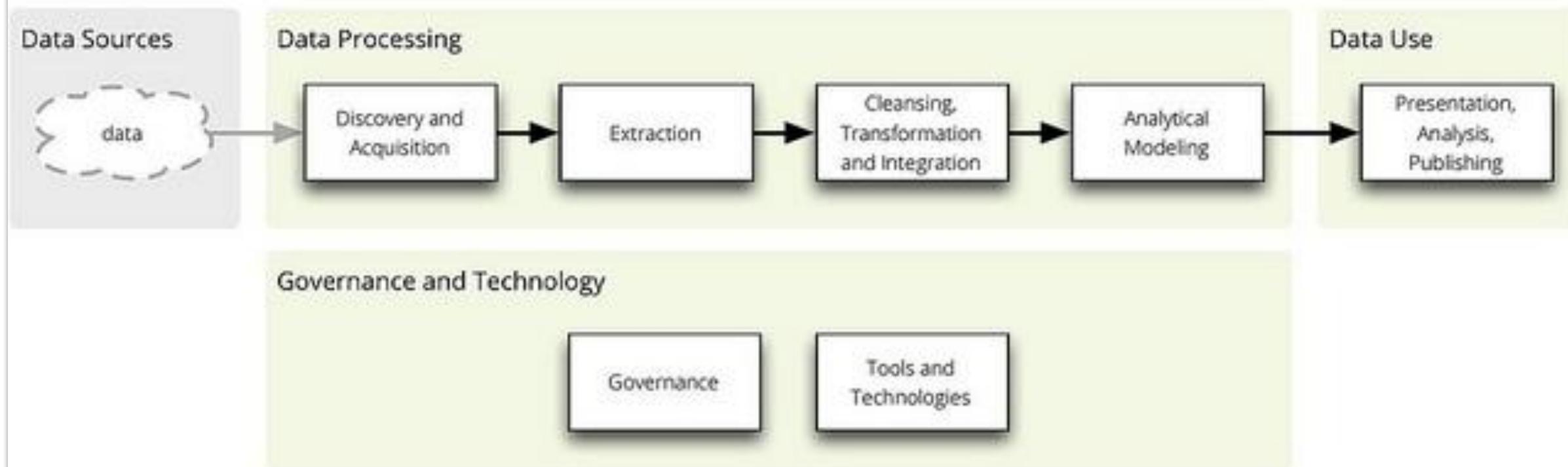
R

```
call:  
lm(formula = y ~ ., data = df)  
  
Residuals:  
    Min      1Q  Median      3Q     Max  
-155.829  -38.534  -0.227  37.806  151.355  
  
Coefficients:  
              Estimate Std. Error t value Pr(>|t|)      
(Intercept) 152.133     2.576  59.061 < 2e-16 ***  
X1          -10.012     59.749  -0.168  0.867000  
X2         -239.819     61.222  -3.917 0.000104 ***  
X3          519.840     66.534   7.813 4.30e-14 ***  
X4          324.390     65.422   4.958 1.02e-06 ***  
X5         -792.184     416.684  -1.901 0.057947 .  
X6          476.746     339.035   1.406 0.160389  
X7          101.045     212.533   0.475 0.634721  
X8          177.064     161.476   1.097 0.273456  
X9          751.279     171.902   4.370 1.56e-05 ***  
X10         67.625      65.984   1.025 0.305998  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 54.15 on 431 degrees of freedom  
Multiple R-squared:  0.5177,    Adjusted R-squared:  0.5066  
F-statistic: 46.27 on 10 and 431 DF,  p-value: < 2.2e-16
```

Syntax, Pipelines, and Versioning

Data Processing Pipeline

School of Data Skill Set



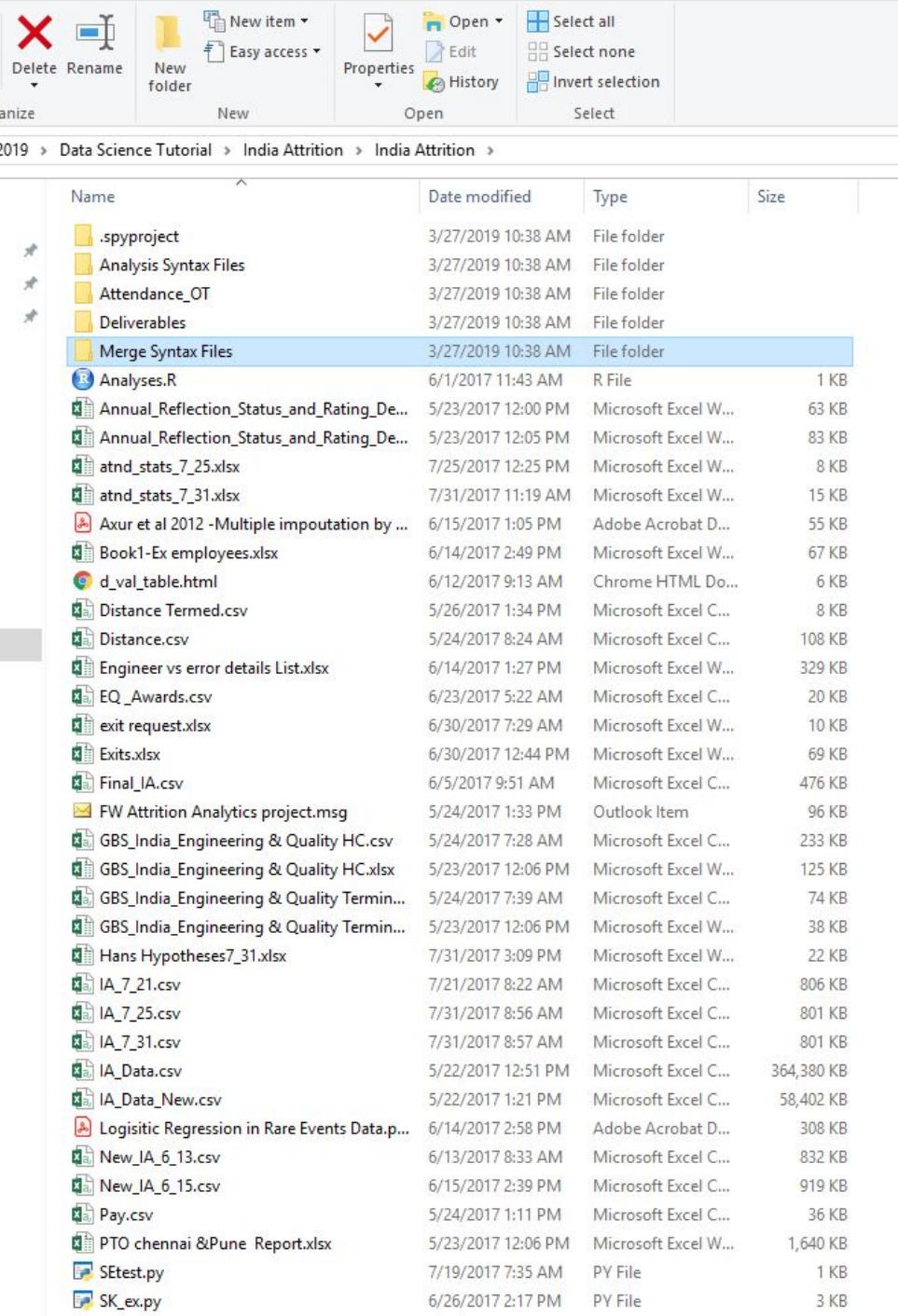
Syntax, Pipelines, and **Versioning**

Versioning keeps your directory clean and documents all changes

Why is this important?

Versioning

The Enemy



A screenshot of a Windows File Explorer window. The title bar shows the path: 2019 > Data Science Tutorial > India Attrition > India Attrition. The ribbon menu at the top includes options like Delete, Rename, New folder, New, Open, Select all, Select none, and Invert selection. The main area displays a list of files and folders. One file, 'Merge Syntax Files', is selected and highlighted with a blue border. The columns in the list view are Name, Date modified, Type, and Size.

Name	Date modified	Type	Size
.spyproject	3/27/2019 10:38 AM	File folder	
Analysis Syntax Files	3/27/2019 10:38 AM	File folder	
Attendance_OT	3/27/2019 10:38 AM	File folder	
Deliverables	3/27/2019 10:38 AM	File folder	
Merge Syntax Files	3/27/2019 10:38 AM	File folder	
Analyses.R	6/1/2017 11:43 AM	R File	1 KB
Annual_Reflection_Status_and_Rating_De...	5/23/2017 12:00 PM	Microsoft Excel W...	63 KB
Annual_Reflection_Status_and_Rating_De...	5/23/2017 12:05 PM	Microsoft Excel W...	83 KB
atnd_stats_7_25.xlsx	7/25/2017 12:25 PM	Microsoft Excel W...	8 KB
atnd_stats_7_31.xlsx	7/31/2017 11:19 AM	Microsoft Excel W...	15 KB
Axur et al 2012 -Multiple imputation by ...	6/15/2017 1:05 PM	Adobe Acrobat D...	55 KB
Book1-Ex employees.xlsx	6/14/2017 2:49 PM	Microsoft Excel W...	67 KB
d_val_table.html	6/12/2017 9:13 AM	Chrome HTML Do...	6 KB
Distance Termed.csv	5/26/2017 1:34 PM	Microsoft Excel C...	8 KB
Distance.csv	5/24/2017 8:24 AM	Microsoft Excel C...	108 KB
Engineer vs error details List.xlsx	6/14/2017 1:27 PM	Microsoft Excel W...	329 KB
EQ_Awards.csv	6/23/2017 5:22 AM	Microsoft Excel C...	20 KB
exit request.xlsx	6/30/2017 7:29 AM	Microsoft Excel W...	10 KB
Exits.xlsx	6/30/2017 12:44 PM	Microsoft Excel W...	69 KB
Final_IA.csv	6/5/2017 9:51 AM	Microsoft Excel C...	476 KB
FW Attrition Analytics project.msg	5/24/2017 1:33 PM	Outlook Item	96 KB
GBS_India_Engineering & Quality HC.csv	5/24/2017 7:28 AM	Microsoft Excel C...	233 KB
GBS_India_Engineering & Quality HC.xlsx	5/23/2017 12:06 PM	Microsoft Excel W...	125 KB
GBS_India_Engineering & Quality Termin...	5/24/2017 7:39 AM	Microsoft Excel C...	74 KB
GBS_India_Engineering & Quality Termin...	5/23/2017 12:06 PM	Microsoft Excel W...	38 KB
Hans Hypotheses7_31.xlsx	7/31/2017 3:09 PM	Microsoft Excel W...	22 KB
IA_7_21.csv	7/21/2017 8:22 AM	Microsoft Excel C...	806 KB
IA_7_25.csv	7/31/2017 8:56 AM	Microsoft Excel C...	801 KB
IA_7_31.csv	7/31/2017 8:57 AM	Microsoft Excel C...	801 KB
IA_Data.csv	5/22/2017 12:51 PM	Microsoft Excel C...	364,380 KB
IA_Data_New.csv	5/22/2017 1:21 PM	Microsoft Excel C...	58,402 KB
Logisitic Regression in Rare Events Data.p...	6/14/2017 2:58 PM	Adobe Acrobat D...	308 KB
New_IA_6_13.csv	6/13/2017 8:33 AM	Microsoft Excel C...	832 KB
New_IA_6_15.csv	6/15/2017 2:39 PM	Microsoft Excel C...	919 KB
Pay.csv	5/24/2017 1:11 PM	Microsoft Excel C...	36 KB
PTO chennai &Pune Report.xlsx	5/23/2017 12:06 PM	Microsoft Excel W...	1,640 KB
SEtest.py	7/19/2017 7:35 AM	PY File	1 KB
SK_ex.py	6/26/2017 2:17 PM	PY File	3 KB

Versioning

This one does not spark joy

Name	Date modified	Type	Size
IA_Merge	5/26/2017 3:09 PM	File	7 KB
IA_Merge_6_6	6/6/2017 2:01 PM	File	11 KB
IA_Merge_MA_Version	6/7/2017 2:04 PM	File	11 KB
IA_Merge_MA_Version.py	6/27/2017 12:25 PM	PY File	11 KB
IA_Merge6_1	6/6/2017 2:02 PM	File	7 KB
IA_Merge6_1.1	6/5/2017 8:19 AM	File	7 KB
IA_Merge6_5.5	6/5/2017 2:39 PM	File	11 KB
IA_Slice	5/23/2017 11:26 AM	File	2 KB
Impute_LRG_VOICE.py	6/19/2017 12:40 PM	PY File	1 KB
Merge_6_17	6/16/2017 7:26 AM	File	16 KB
Merge_6_17.py	6/19/2017 12:51 PM	PY File	16 KB
Merge_6_17_messUP.py	6/19/2017 2:18 PM	PY File	15 KB
Merge_7_17.py	7/18/2017 9:48 AM	PY File	20 KB
Merge_7_19.py	7/20/2017 10:55 AM	PY File	23 KB
Merge_7_20.py	7/20/2017 1:39 PM	PY File	26 KB
Merge_7_21.py	7/21/2017 8:22 AM	PY File	28 KB
Merge_7_25.py	7/25/2017 1:15 PM	PY File	29 KB
Merge_7_31.py	7/31/2017 3:10 PM	PY File	26 KB
Merge_8_9.py	8/9/2017 2:32 PM	PY File	26 KB
Merge_Lvoice_16Voice_errors	6/15/2017 3:08 PM	File	16 KB
MergeNewVoice6_13	6/13/2017 11:33 AM	File	11 KB
Voice2016	6/13/2017 3:48 PM	File	3 KB

Versioning

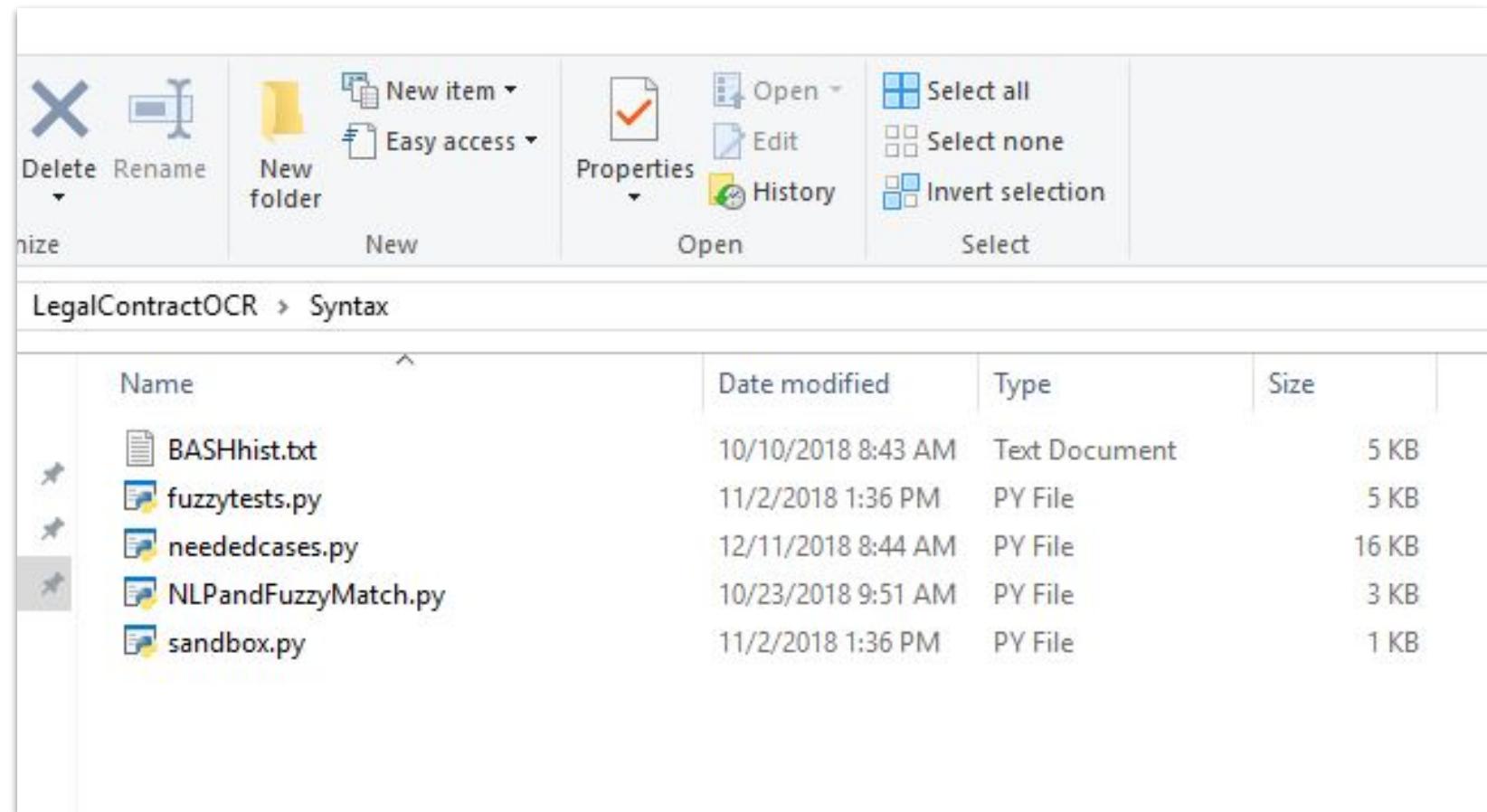
The Hero

The screenshot shows a Windows-style file explorer window. The title bar indicates the path: 'LegalContractOCR'. The ribbon menu at the top has tabs for 'Analyze', 'New', 'Open', and 'Select'. The 'Open' tab is active, showing a list of items. The list includes several folders (.git, .spyproject, Data, Docs, Output, Syntax, Viz) and one text document (.gitignore). The 'Syntax' folder is currently selected, highlighted with a blue background. The columns in the list view are 'Name', 'Date modified', 'Type', and 'Size'.

Name	Date modified	Type	Size
.git	12/11/2018 9:39 AM	File folder	
.spyproject	10/12/2018 10:02 ...	File folder	
Data	11/5/2018 7:44 AM	File folder	
Docs	11/29/2018 3:46 PM	File folder	
Output	2/25/2019 3:36 PM	File folder	
Syntax	10/30/2018 11:47 ...	File folder	
Viz	10/12/2018 10:02 ...	File folder	
.gitignore	10/30/2018 1:01 PM	Text Document	1 KB

Versioning

This one sparks joy



A screenshot of a Windows File Explorer window. The title bar shows 'LegalContractOCR > Syntax'. The ribbon menu is visible at the top, with tabs for 'File', 'Edit', 'View', 'Insert', 'Format', 'Tools', and 'Help'. The 'File' tab is selected. The 'New' section of the ribbon has 'New folder' highlighted. The 'Open' section has 'Properties' highlighted. The 'Select' section has 'Select all' checked. The main area displays a list of files:

	Name	Date modified	Type	Size
1	BASHhist.txt	10/10/2018 8:43 AM	Text Document	5 KB
2	fuzzytests.py	11/2/2018 1:36 PM	PY File	5 KB
3	neededcases.py	12/11/2018 8:44 AM	PY File	16 KB
4	NLPandFuzzyMatch.py	10/23/2018 9:51 AM	PY File	3 KB
5	sandbox.py	11/2/2018 1:36 PM	PY File	1 KB

Versioning

Git - Language used for versioning (keeps track of changes) **Two Main Functions**

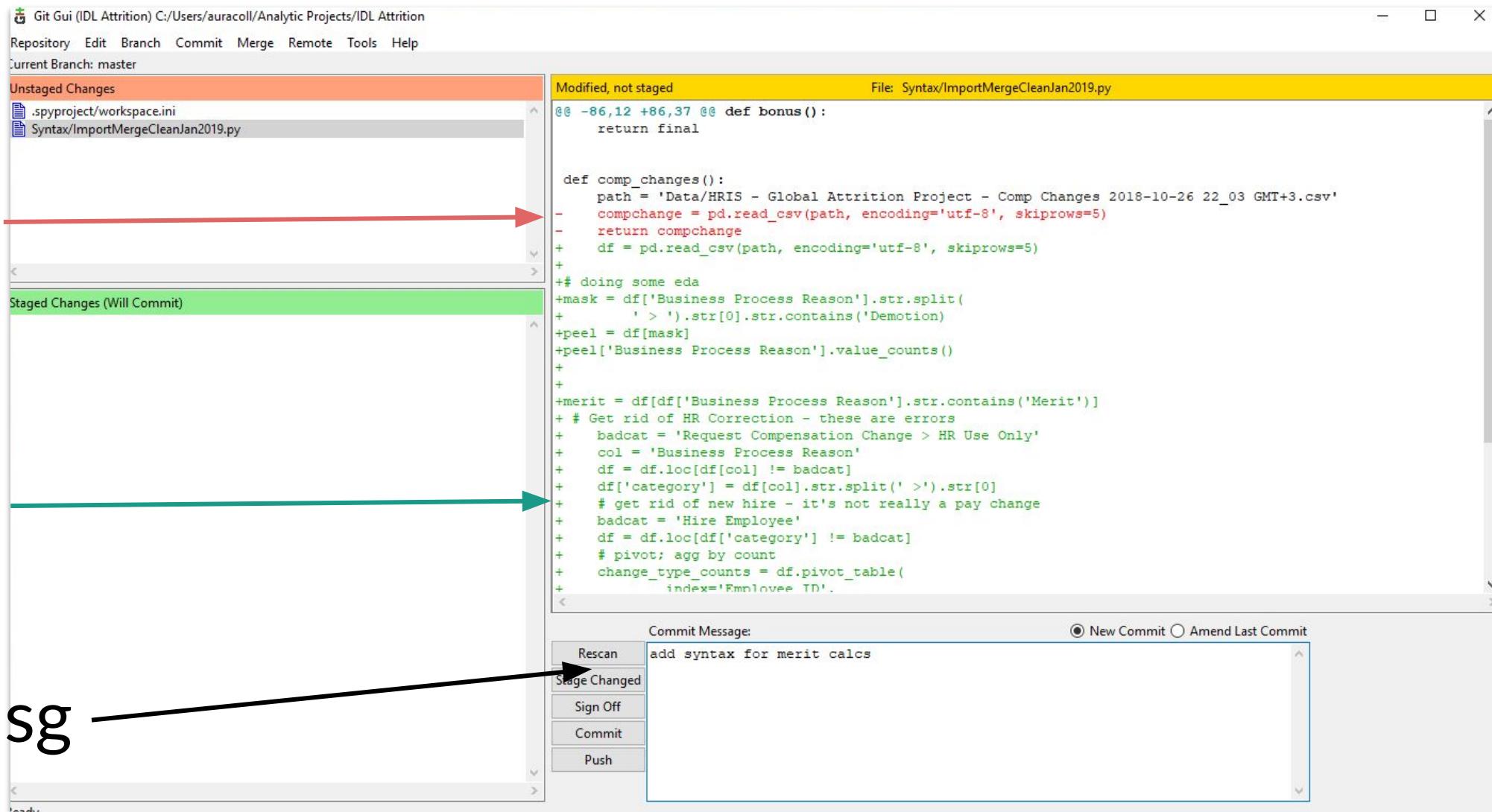
1. Stage - make changes to a file
2. Commit - publish group of changes, accompanied by
 - a. Commit Message - Brief present-tense description
 - e.g. “*create pie chart*” “*resolve duplicate IDs*”

Versioning Git

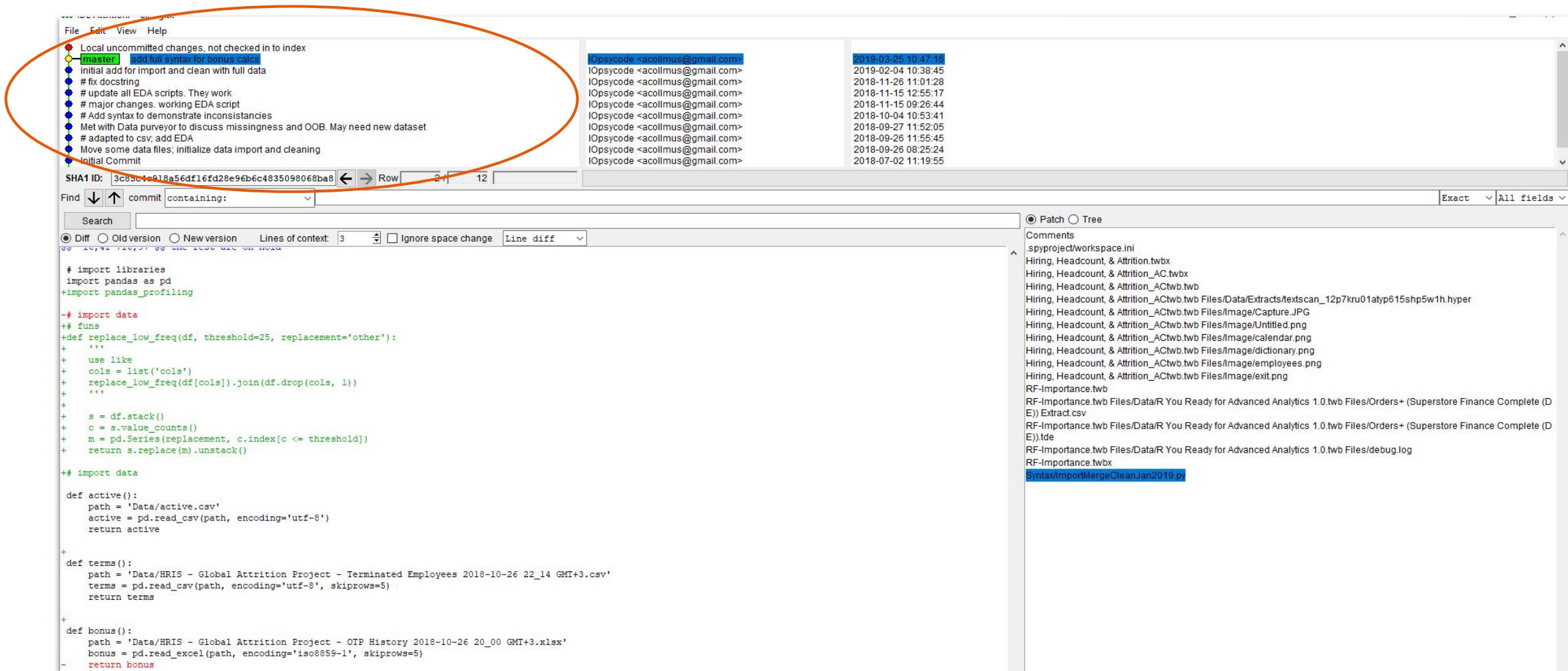
Deletions

Insertions

Commit Msg



Versioning Git Tree



The screenshot shows a Git commit history interface with several panels:

- Top Left Panel:** Shows a list of local uncommitted changes under the "master" branch. One change is highlighted with a red background and white text: "add full syntax for bonus calc".
 - SHA1 ID: 3c85c4e918a56df16fd28e96b6c4835098068ba8
 - Find commit containing: (Search bar)
 - Diff, Old version, New version, Lines of context (3), Ignore space change, Line diff buttons.
- Top Right Panel:** A list of commits from the master branch, ordered by date.

Date	Author
2019-03-25 10:47:16	IOpsycode <acollmus@gmail.com>
2019-02-04 10:38:45	IOpsycode <acollmus@gmail.com>
2018-11-26 11:01:28	IOpsycode <acollmus@gmail.com>
2018-11-15 12:55:17	IOpsycode <acollmus@gmail.com>
2018-11-15 09:26:44	IOpsycode <acollmus@gmail.com>
2018-10-04 10:53:41	IOpsycode <acollmus@gmail.com>
2018-09-27 11:52:05	IOpsycode <acollmus@gmail.com>
2018-09-26 11:55:45	IOpsycode <acollmus@gmail.com>
2018-09-26 08:25:24	IOpsycode <acollmus@gmail.com>
2018-07-02 11:19:55	IOpsycode <acollmus@gmail.com>
- Bottom Right Panel:** A tree view of project files and folders, showing a large number of files related to "Hiring, Headcount, & Attrition" and "RF-Importance".
 - Comments
 - .spyproject/workspace.ini
 - Hiring, Headcount, & Attrition.twbx
 - Hiring, Headcount, & Attrition_AC.twbx
 - Hiring, Headcount, & Attrition_AActwb.twb
 - Hiring, Headcount, & Attrition_AActwb.twb Files/Data/Extracts/textscan_12p7kru01atyp615shp5w1.hyper
 - Hiring, Headcount, & Attrition_AActwb.twb Files/Image/Capture.JPG
 - Hiring, Headcount, & Attrition_AActwb.twb Files/Image/Untitled.png
 - Hiring, Headcount, & Attrition_AActwb.twb Files/Image/calendar.png
 - Hiring, Headcount, & Attrition_AActwb.twb Files/Image/dictionary.png
 - Hiring, Headcount, & Attrition_AActwb.twb Files/Image/employees.png
 - Hiring, Headcount, & Attrition_AActwb.twb Files/Image/exit.png
 - RF-Importance.twb
 - RF-Importance.twb Files/Data/R You Ready for Advanced Analytics 1.0.twb Files/Orders+ (Superstore Finance Complete (D E)) Extract.csv
 - RF-Importance.twb Files/Data/R You Ready for Advanced Analytics 1.0.twb Files/Orders+ (Superstore Finance Complete (D E)).tde
 - RF-Importance.twb Files/Data/R You Ready for Advanced Analytics 1.0.twb Files/debug.log
 - RF-Importance.twbx
 - Syntax/ImportMergeCleanJan2019.py

Versioning Git Tree

The screenshot shows a gitk window titled "IDL Attrition: --all - gitk". The menu bar includes File, Edit, View, and Help. A toolbar at the bottom has buttons for Diff, Old version, New version, Lines of context (set to 3), Ignore space change, Line diff, and a search field.

The main area displays a commit history for the "master" branch:

- Local uncommitted changes, not checked in to index
- master add full syntax for bonus calcs
- initial add for import and clean with full data
- # fix docstring
- # update all EDA scripts. They work
- # major changes. working EDA script
- # Add syntax to demonstrate inconsistencies
- Met with Data purveyor to discuss missingness and OOB. May need new dataset
- # adapted to csv; add EDA
- Move some data files; initialize data import and cleaning
- Initial Commit

Below the commits, the SHA1 ID is listed as 3c85c4c918a56df16fd28e96b6c4835098068ba8. Navigation buttons include back, forward, and row selection (2 / 12). A search bar at the bottom allows finding commits containing specific text, and a search button is present.

Versioning Git Tree

The screenshot shows a Git commit history interface with a red circle highlighting the commit message area. The commit message is:

```
master add full syntax for bonus calc
initial add for import and clean with full data
# fix docstring
# update all EDA scripts. They work
# major changes. working EDA script
# Add syntax to demonstrate inconsistencies
Met with Data purveyor to discuss missingness and OOB. May need new dataset
# adapted to csv, add EDA
Move some data files, initialize data import and cleaning
Initial Commit
```

The commit message is highlighted with a red circle.

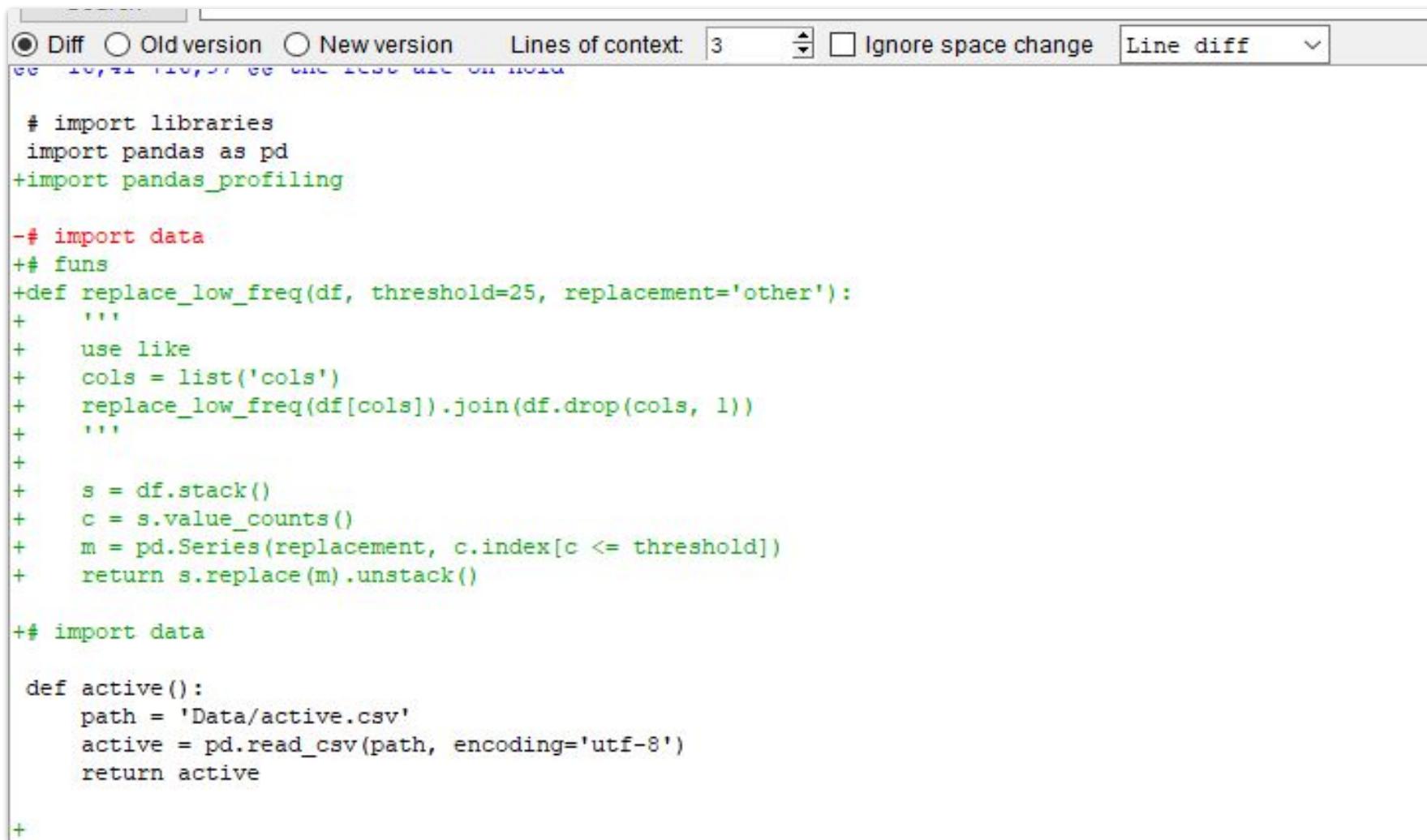
Below the commit message, the SHA1 ID is shown as 3c85c4c918a56df16fd28e96b6c483509806ba8. The interface includes a search bar for 'commit containing:' and various filtering options like 'Diff', 'Old version', 'New version', 'Lines of context: 3', 'Ignore space change', and 'Line diff'.

The commit history table has three columns: Author, Date, and Commit Message. The commits are listed in reverse chronological order, starting from 2019-03-25 at 10:47:16.

Author	Date	Commit Message
IOpsycode <acollmus@gmail.com>	2019-03-25 10:47:16	master add full syntax for bonus calc
IOpsycode <acollmus@gmail.com>	2019-02-04 10:38:45	initial add for import and clean with full data
IOpsycode <acollmus@gmail.com>	2018-11-26 11:01:28	# fix docstring
IOpsycode <acollmus@gmail.com>	2018-11-15 12:55:17	# update all EDA scripts. They work
IOpsycode <acollmus@gmail.com>	2018-11-15 09:26:44	# major changes. working EDA script
IOpsycode <acollmus@gmail.com>	2018-10-04 10:53:41	# Add syntax to demonstrate inconsistencies
IOpsycode <acollmus@gmail.com>	2018-09-27 11:52:05	Met with Data purveyor to discuss missingness and OOB. May need new dataset
IOpsycode <acollmus@gmail.com>	2018-09-26 11:55:45	# adapted to csv, add EDA
IOpsycode <acollmus@gmail.com>	2018-09-26 08:25:24	Move some data files, initialize data import and cleaning
IOpsycode <acollmus@gmail.com>	2018-07-02 11:19:55	Initial Commit

The right panel shows a tree view of the repository structure, including files like .spyproject/workspace.ini, Hiring, Headcount, & Attrition.twbx, and RF-Importance.twbx.

Versioning Git Tree



The screenshot shows a software interface for viewing file differences, likely from a Git commit history. The top bar includes options for 'Diff' (selected), 'Old version', 'New version', 'Lines of context: 3', 'Ignore space change', and 'Line diff'. The main area displays the following Python code with color-coded highlights for changes:

```
# import libraries
import pandas as pd
+import pandas_profiling

-# import data
+# funs
+def replace_low_freq(df, threshold=25, replacement='other'):
+    """
+        use like
+        cols = list('cols')
+        replace_low_freq(df[cols]).join(df.drop(cols, 1))
+    """
+
+    s = df.stack()
+    c = s.value_counts()
+    m = pd.Series(replacement, c.index[c <= threshold])
+    return s.replace(m).unstack()

+# import data

def active():
    path = 'Data/active.csv'
    active = pd.read_csv(path, encoding='utf-8')
    return active
```

A small green '+' character is visible at the bottom left of the code area.

Versioning GitHub

The screenshot shows a GitHub repository page for 'TNT-Lab / 2019-SIOP-ML-Competition'. The repository is private, has 0 issues, 0 pull requests, 1 project, and no wiki. The 'Code' tab is selected. The branch is 'master'. The commit history shows two main entries:

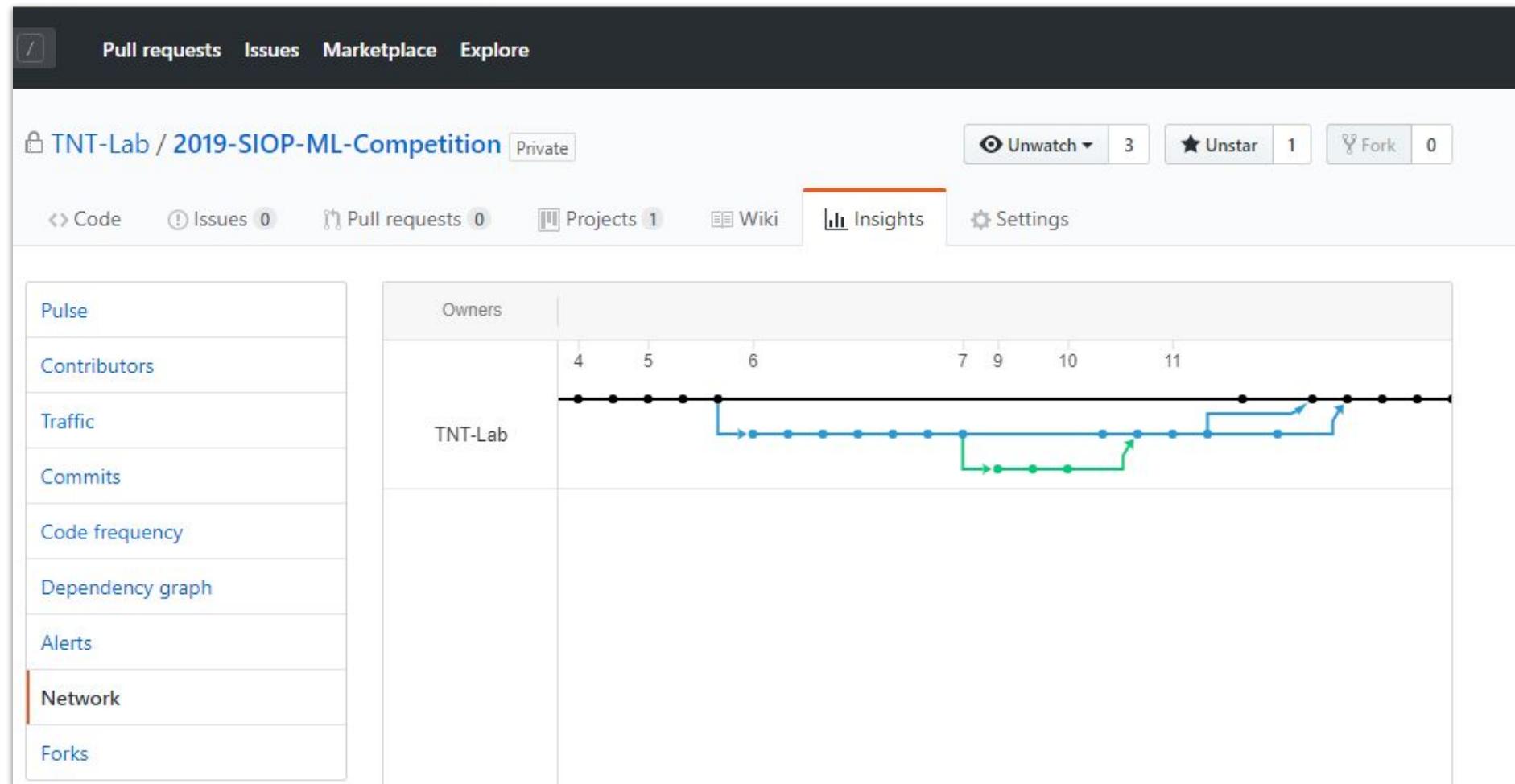
- Commits on Mar 19, 2019:**
 - some attempts to work in R
IOpsycode committed 9 days ago
- Commits on Mar 14, 2019:**
 - Add files via upload
nbjornberg committed 14 days ago
 - add R syntax for someone else to use please
IOpsycode committed 14 days ago
 - add hyperparams and backup syntax cuz computer crash imminent
IOpsycode committed 14 days ago
 - add R file for Andy; add flair w no preproc (which is bad)
IOpsycode committed 14 days ago

Versioning

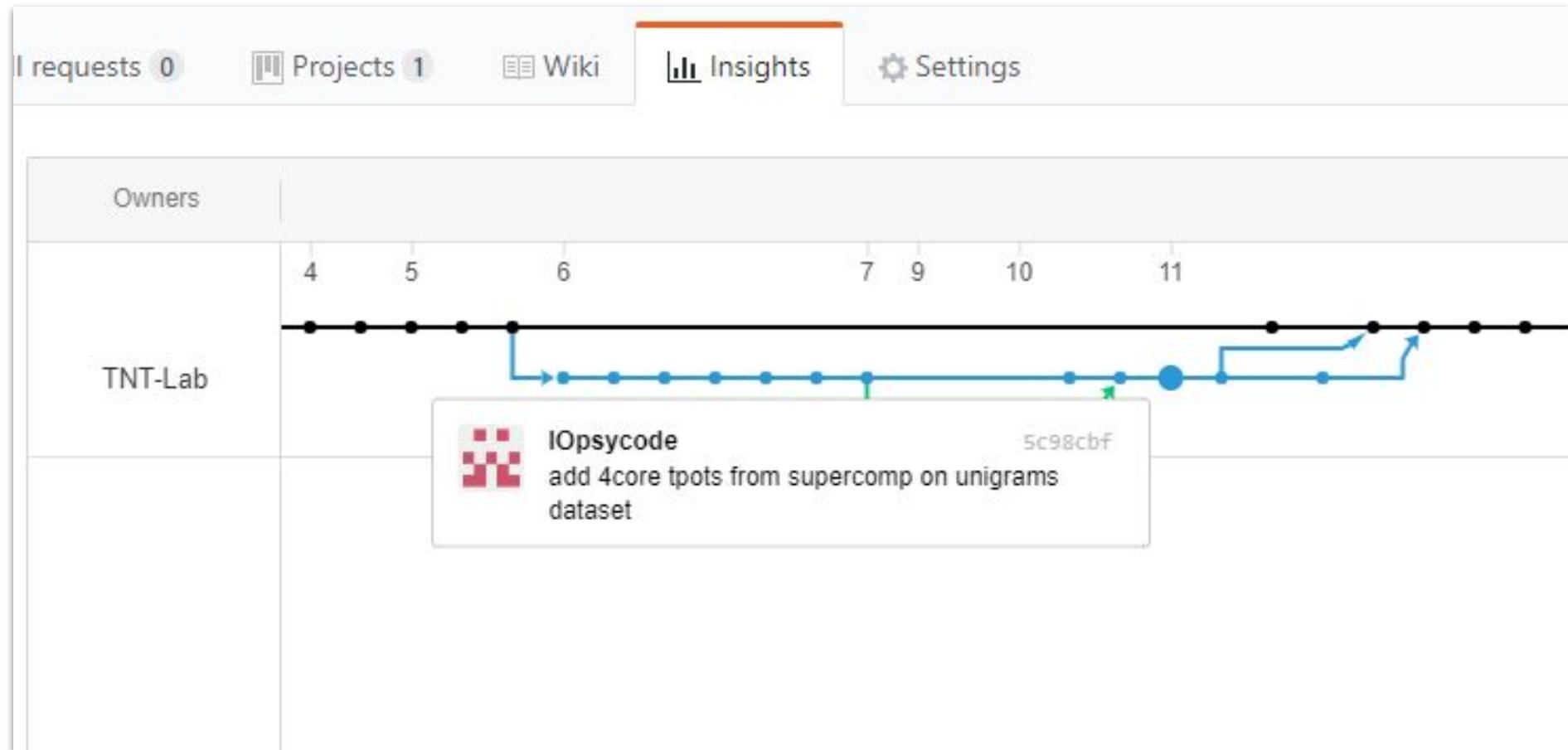
Github - Online Repo built on Git Language **Two New Functions**

1. **Pull** - get up-to-date version from master repository
2. **Stage** -
3. **Commit** -
 - a. **Commit Message** -
4. **Push** - send your version to master repository

Versioning GitHub



Versioning GitHub



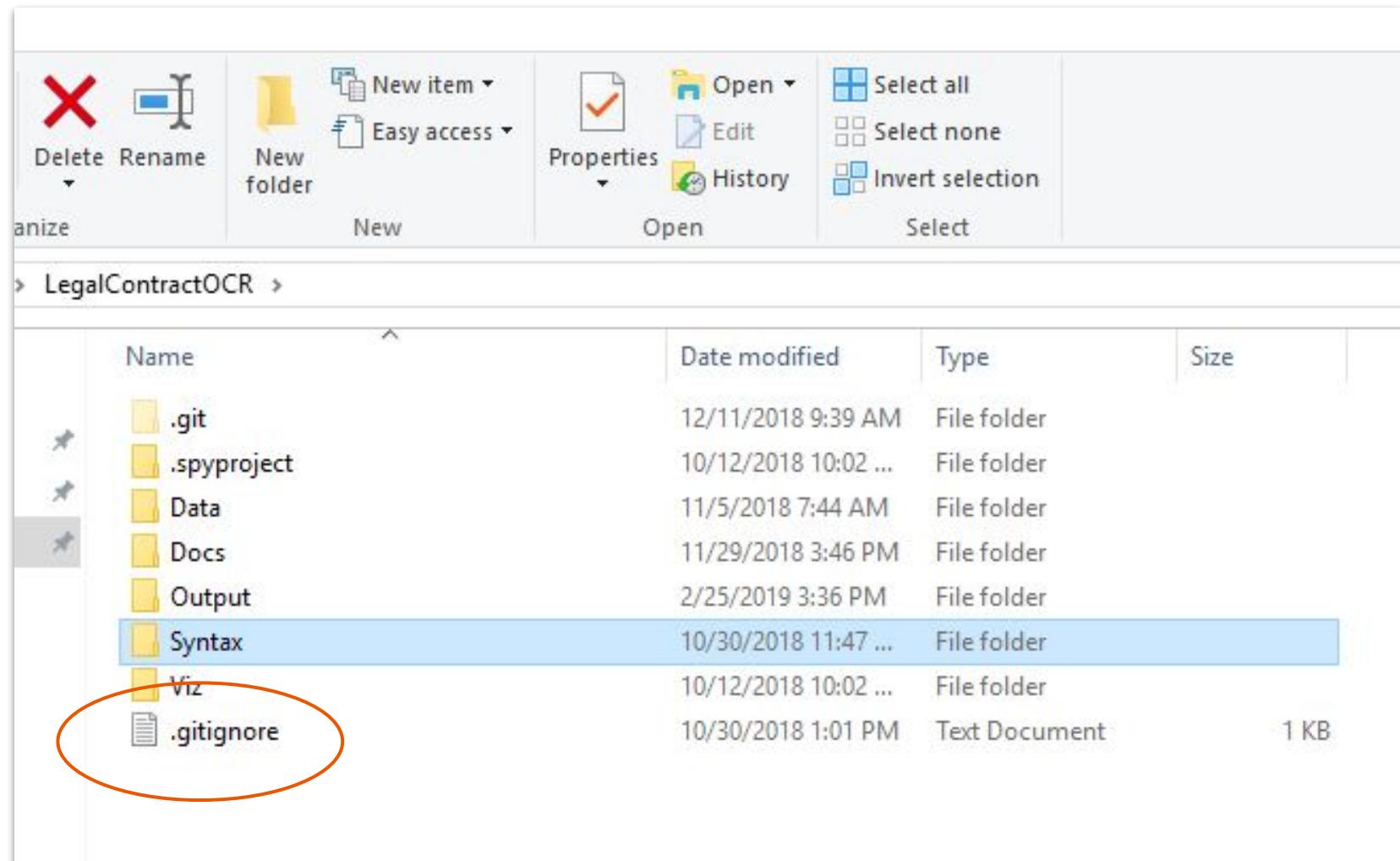
Versioning

Git Ignore

- txt file that lists what you want to ignore
 - i.e., don't commit, don't push, don't pull
- Use to protect sensitive data
- Use to avoid workspace conflicts
- Accepts wildcards *

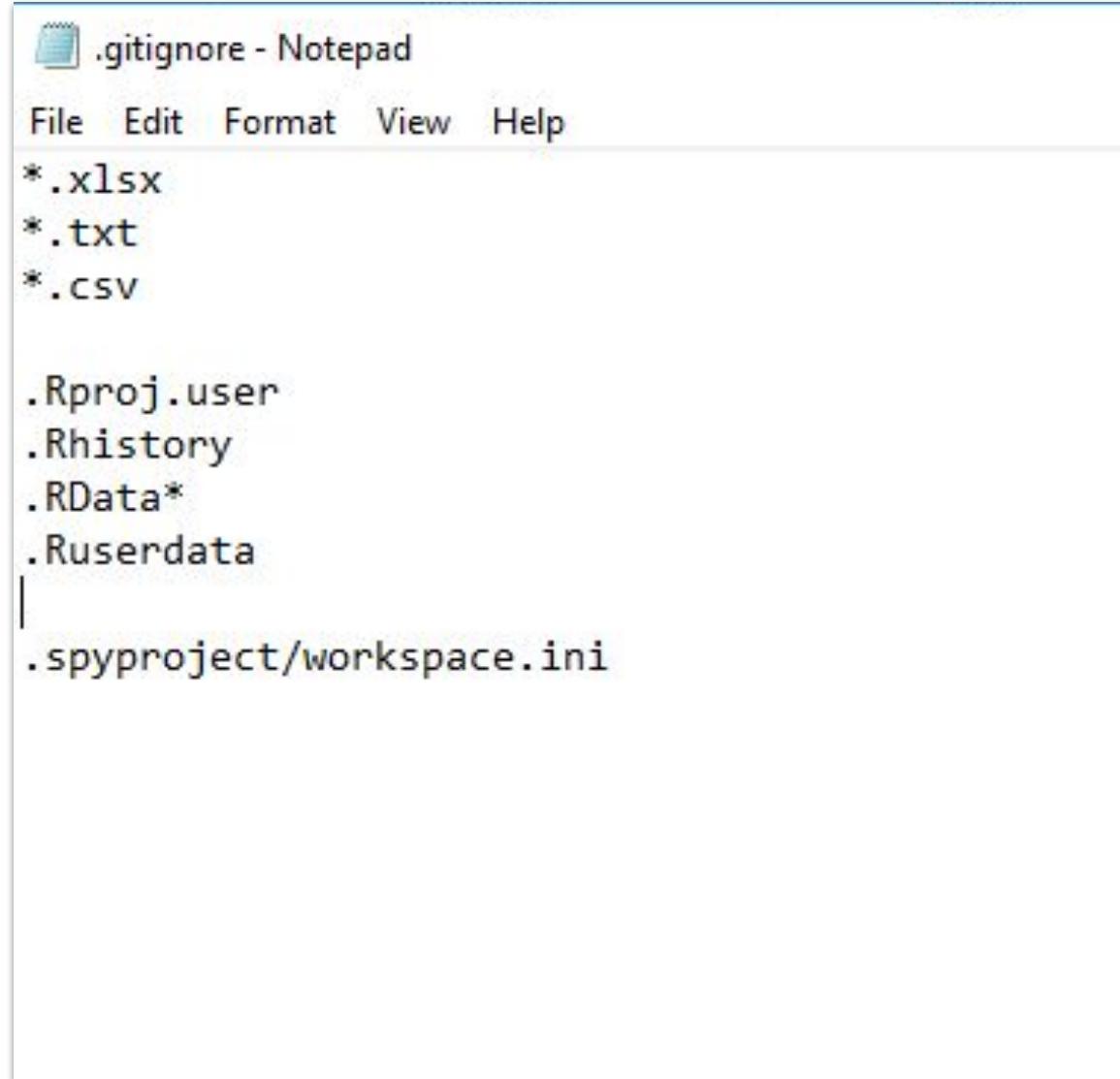
Versioning Git Ignore

Remember
our Hero?



Versioning Git Ignore

Example from
Recent Project



.gitignore - Notepad

File Edit Format View Help

```
*.xlsx
*.txt
*.csv

.Rproj.user
.Rhistory
.RData*
.Ruserdata
|
.spyproject/workspace.ini
```

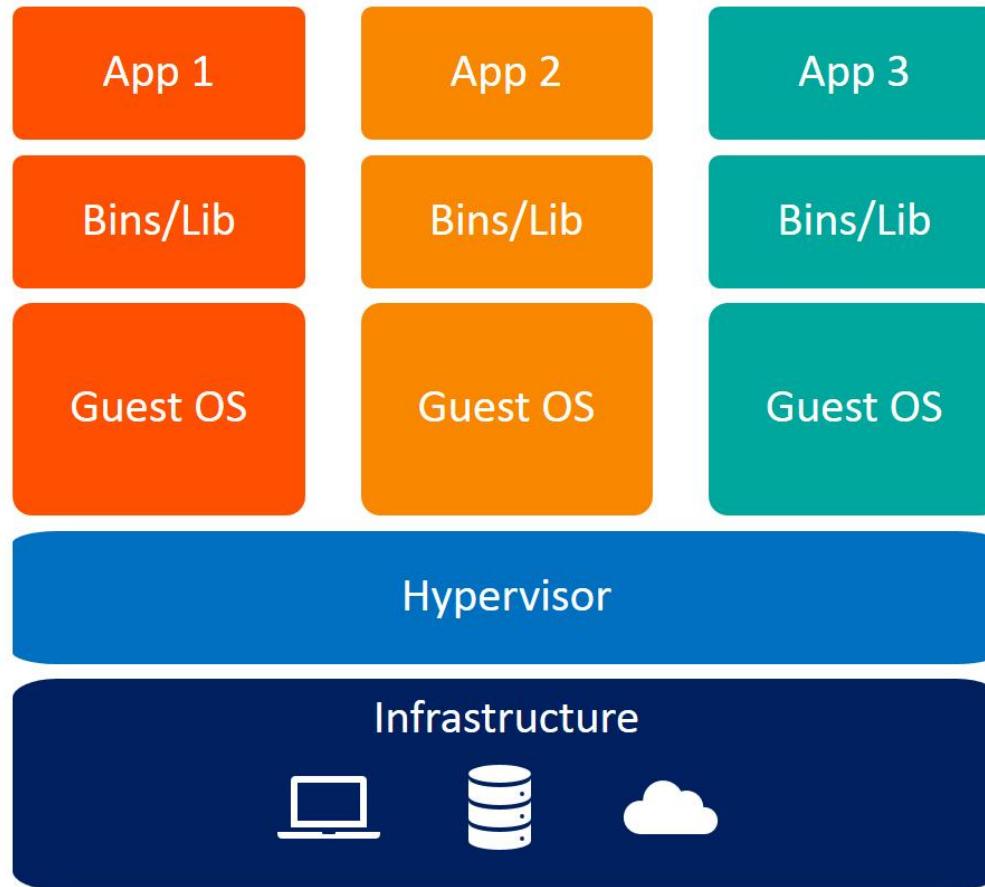
Interactive Notebooks and Containers

- Jupyter
- Docker
- Binder

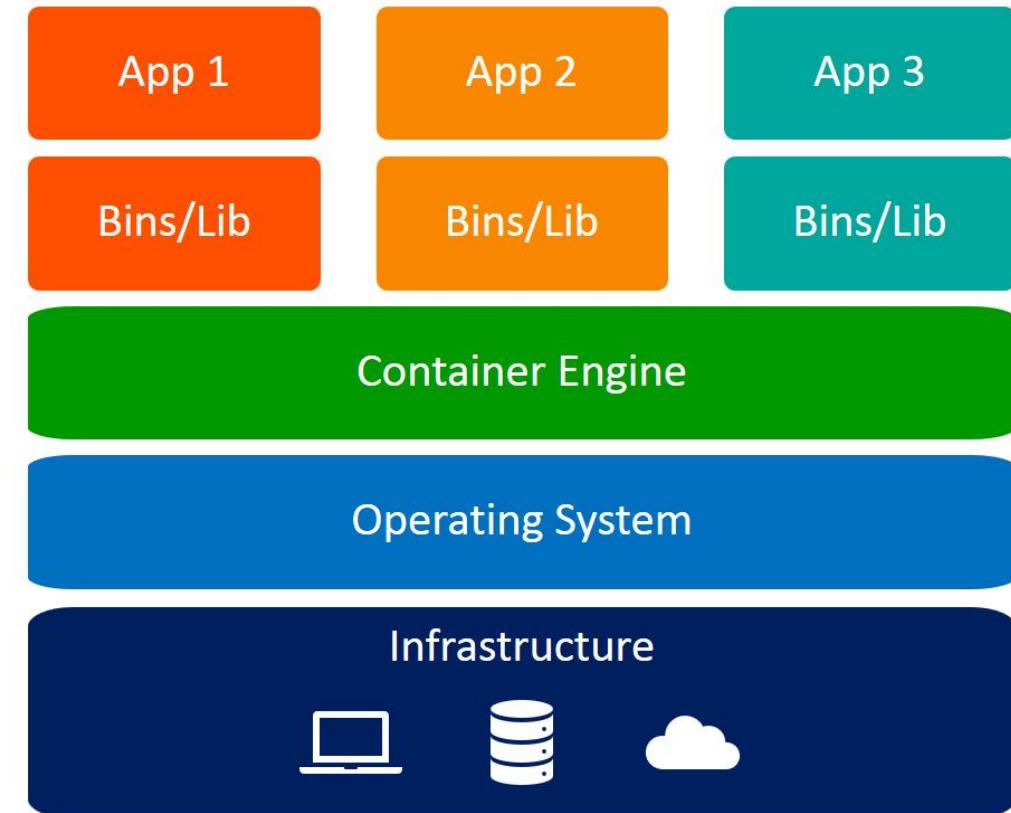
Interactive Notebooks and Containers

- Jupyter - notebook that can run R or Python and markdown in your web browser.
 - Easy to show interactive visualizations
 - Easy to demonstrate concepts to stakeholders
 - You will see example shortly

Interactive Notebooks and Containers



Virtual Machines



Containers

Interactive Notebooks and Containers

Binder - easily turn
github repos into
shareable
notebooks



[eauer22 / SIOP-2019-Master-Tutorial-Creating-Reproducible-and-Interactive-Analyses-with-JupyterLab-and-Binder-](#)

[Watch 0](#) [Star 1](#) [Fork 0](#)

[Code](#) [Issues 0](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Insights](#)

This tutorial demonstrates two data science tools that enable IO psychologists to create interactive, literate code documents, enabling others to replicate analyses with one click on the web.

5 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find File Clone or download

eauer22 update README Latest commit 1cf3bbe 2 days ago

JNandBinderTutorial.ipynb Create sample jupyter notebook 2 days ago

README.md update README 2 days ago

install.R create install.R for binder 2 days ago

runtime.txt create runtime.txt for binder 2 days ago

README.md

SIOP-2019-Master-Tutorial-Creating-Reproducible-and-Interactive-Analyses-with-JupyterLab-and-Binder-

This tutorial demonstrates two data science tools that enable IO psychologists to create interactive, literate code documents, enabling others to replicate analyses with one click on the web.

[launch binder](#)



**Learning Objective #2: Organize and analyze
new sources of data to better address old
research questions and develop new research
questions not previously testable**

Next, we'll be moving to a Jupyter Notebook
To follow along with us:

<http://bit.ly/SIOPDataScienceTutorial>

Topics Discussed:

Data Sourcing Techniques:

- APIs & Webscraping

Data Science Methods:

- Natural Language Processing
- Machine Learning

Demo:

1. Access Tweets from Twitter API
2. Clean Tweet Text
3. Predict Tweet Popularity using Machine Learning



**Learning Objective #3: Increase the credibility
of your research and IO psychology research in
general to outside researchers and the public**

Data Science can be leveraged to make I-O more insightful, reproducible, and trustworthy

Insightful:

- Measurement
- Triangulation

Reproducible:

- Open-source tools
- Documentation

Trustworthy:

- Increase credibility of I-O research
- Help influence public and inform public policy

An **insightful** I-O psychology

- Change in measurement of behavior
 - “new” data (e.g., microbehaviors, text data)
- Data science methods add to triangulation of theory
 - Improved prediction
 - New methods for theory-testing
 - Inductive theory building
- Example: examining dynamic phenomena with wearable bluetooth sensors (Matusik et al., 2018)

A reproducible I-O psychology

- Data science practices make reproducibility *easier*
- Data science practices make reproducibility *more explicit*
- Data science practices make reproducibility *the default*
- Why not make verification immediate and easy for other researchers?

A **trustworthy** I-O psychology

- I-O's have led the way with:
 - reporting effect sizes
 - meta-analysis
- Will we lead the way in integrating data science practices?
- Data science tools can help reduce *p*-hacking and QRPs by increasing transparency

The public and public policy

Empower citizens directly	Influence target stakeholders
<ul style="list-style-type: none">• Publish data, code/syntax for public access<ul style="list-style-type: none">• learners/researchers (and your future self)• encourages inclusion	<ul style="list-style-type: none">• Rapid dissemination of results<ul style="list-style-type: none">• Research synthesis (e.g., metaBUS)• Good PR: better reproducibility & replication, stakeholder buy-in

Additional Resources

Tutorial GitHub page:

<http://bit.ly/SIOPDataScienceTutorial>

Data Science for Social Scientists Online Course:

<http://datascience.tntlab.org/>

New Book Chapter:

Landers, R. N., Auer, E. M., Collmus, A. B., & Marin, S. (2019). Data science as a new foundation for insightful, reproducible, and trustworthy social science. In R. N. Landers (Ed.), *Cambridge Handbook of Technology and Employee Behavior* (pp. 761-789). New York, NY: Cambridge University Press.

Thank you!
