

# Data Strategies

+ Data Tactics

Thomas Stoeger

[thomas.stoeger@northwestern.edu](mailto:thomas.stoeger@northwestern.edu)

Northwestern | INFORMATION TECHNOLOGY  
RESEARCH COMPUTING SERVICES

**NUPF** — Northwestern University  
Postdoctoral Forum

**SPIE**   
Student Chapter  
Northwestern University

Northwestern | Northwestern Institute on Complex Systems  
DATA SCIENCE

The  
 BDS STUDENT GROUP



NORTHWESTERN'S POSTDOCTORAL FORUM PRESENTS:

# CURRENT RESEARCH & FUTURE CAREERS 2018

NORTHWESTERN'S LARGEST FULLY INTERDISCIPLINARY SYMPOSIUM

AUGUST 30, 2018, WIEBOLDT HALL, CHICAGO

CURRENT RESEARCH &  
FUTURE CAREERS 2018

[Home](#)

[Program](#)

[Registration](#)

[Abstract Submission](#)

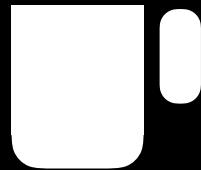
[Contact](#)

[www.current-research-future-careers-2018.org](http://www.current-research-future-careers-2018.org)



# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs



# Environment (very short)

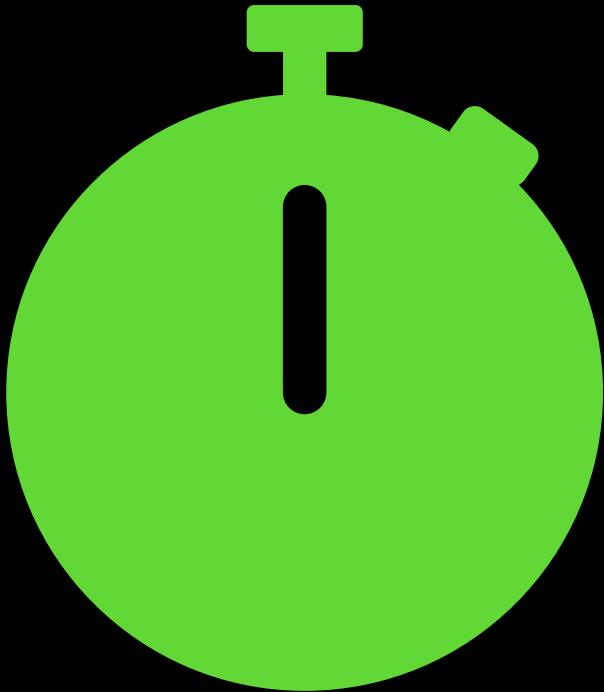
- Language
- Computer
- Space
- Constant learning

# Organization

- Data
- Code
- Computation

# Summary What did we miss?

**After lecture (extra): getting  
your environment ready**



## Two minute exercise.

**Find a neighbor, or  
neighbors, whom  
you do not know.**

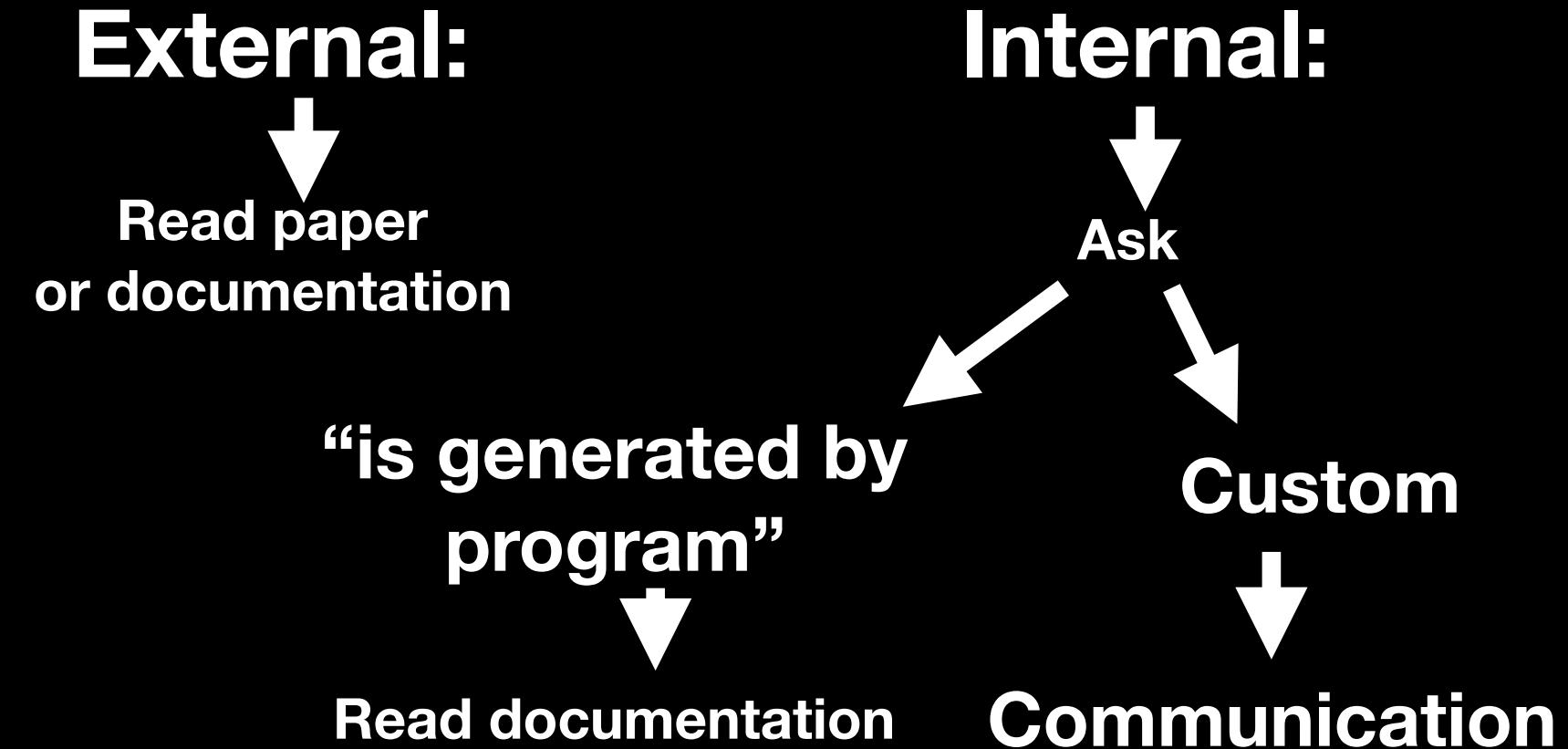
**Then talk about the worst  
data you have encountered.**

# How can we anticipate problems?

**“just”, “easy”, “simple”**

Can not share data or code.

# Understand the past history of the data.



**Write consistency checks!  
(programmatically)**

**Expect to spend  
days or weeks**

**Tips for getting all information:**

- be decent person
- remove fear that you will scoop

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

← **Here**

# Organization

- Data
- Code
- Computation



# Environment (very short)

- Language
- Computer
- Space
- Constant learning

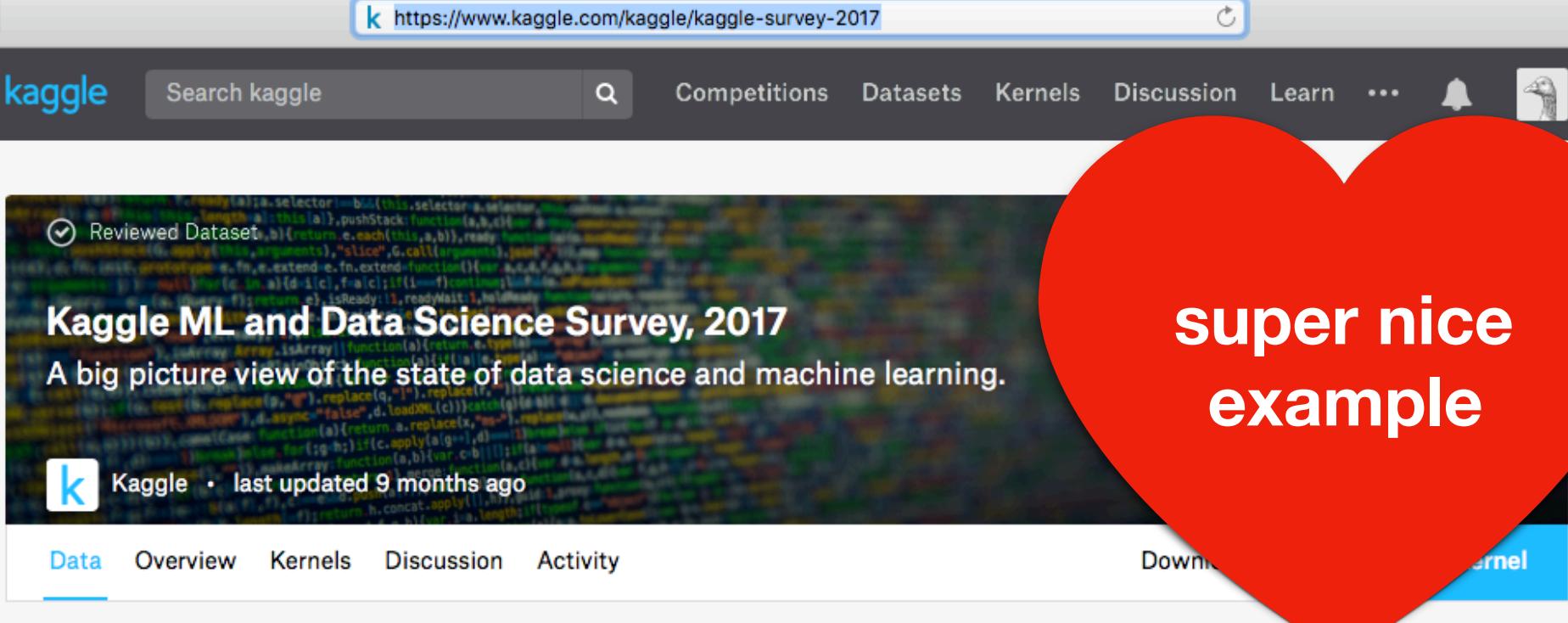
# Summary

**What did we miss?**

**After lecture (extra): getting  
your environment ready**

# Introducing Kaggle ML and Data Science Survey, 2017

## An example data set, which we will be using



A screenshot of a web browser displaying the Kaggle website at <https://www.kaggle.com/kaggle/kaggle-survey-2017>. The page title is "Kaggle ML and Data Science Survey, 2017" and the subtitle is "A big picture view of the state of data science and machine learning." A large red heart graphic with the text "super nice example" is overlaid on the bottom right of the screenshot.

The screenshot shows the following interface elements:

- Header: kaggle, Search kaggle, Q, Competitions, Datasets, Kernels, Discussion, Learn, ..., Bell icon, Profile icon.
- Page Title: Kaggle ML and Data Science Survey, 2017
- Page Subtitle: A big picture view of the state of data science and machine learning.
- Dataset Summary:
  - Reviewed Dataset
  - Kaggle · last updated 9 months ago
- Navigation: Data (selected), Overview, Kernels, Discussion, Activity, Download, Kernel

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

← **Here**

**Will already learn  
some lessons**



# Organization

- Data
- Code
- Computation

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

# Summary

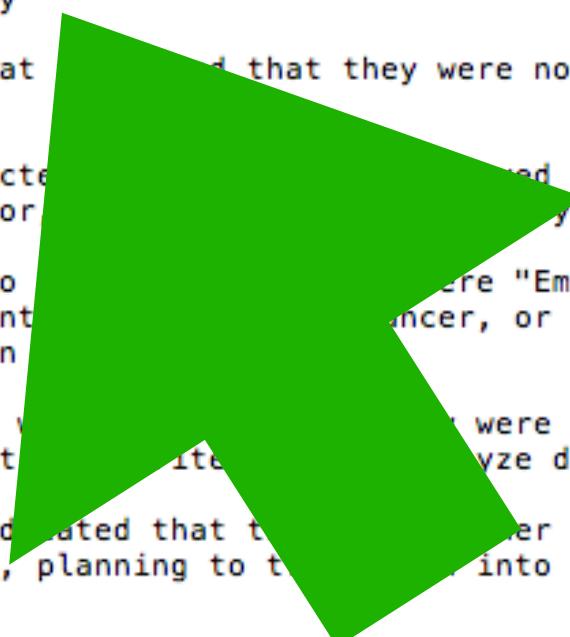
**What did we miss?**

**After lecture (extra): getting  
your environment ready**



## Name

-  conversionRates.csv
-  freeformResponses.csv
-  multipleChoiceResponses.csv
-  RespondentTypeREADME.txt
-  schema.csv



```
All: Every respondent was shown this question

Non-worker: Respondents who indicated that they were "Not employed, and not looking for work" or "I prefer not to say"

Non-switcher: Respondents that indicated that they were not actively looking to switch careers to data science.

Worker: Respondents who indicated that they were "Employed full-time", "Employed part-time", "Independent contractor", "Self-employed", or "Retired"

CodingWorker: Respondents who indicated that they were "Employed full-time", "Employed part-time", or an "Independent contractor". AND that they write code to analyze data in their current job.

CodingWorker-NC: Respondents who indicated that they were "Employed full-time" or "Employed part-time" AND that they did not write code to analyze data in their current job.

Learners: Respondents who indicated that they were "Not employed", "Student", "Under student", "Former student", "Planning to enter", "Entered", or "Not interested" into data science, or not employed but looking for work
```

**Curators might follow specific assumptions.  
e.g.: “I prefer not to say” at employment status  
is essentially “not employed”.**

**Understand terminology!**

schema

Possible Data Loss Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format.

	A	B	C	D
1	Column	Question	Asked	
2	GenderSelect	Select your gender identity. - Selected Choice	All	
3	GenderFreeForm	Select your gender identity. - A different identity - Text	All	
4	Country	Select the country you currently live in.	All	
5	Age	What's your age?	All	
6	EmploymentStatus	What's your current employment status?	All	
7	StudentStatus	Are you currently enrolled as a student at a degree granting school?	Non-worker	
8	LearningDataScience	Are you currently focused on learning data science skills either formally or informally?	Non-worker	
9	KaggleMotivationFreeForm	What's your motivation for being a Kaggle user?	Non-switcher	
10	CodeWriter	Do you write code to analyze data in your current job, freelance contracts, or most recent job if retired?	Worker1	
11	CareerSwitcher	Are you actively looking to switch careers to data science?	Worker1	
12	CurrentJobTitleSelect	Select the option that's most similar to your current job/professional title (or most recent title if retired). - Selected Choice	Worker1	
13	CurrentJobTitleFreeForm	Select the option that's most similar to your current job/professional title (or most recent title if retired). - Other - Text	Worker1	
14	TitleFit	How adequately do you feel your title describes what you do (or what you did if retired)?	Worker1	
15	CurrentEmployerType	Which of the following describe your current employer (or most recent employer if retired)? (Select all that apply)	Worker1	
16	MLToolNextYearSelect	Which tool or technology are you most excited about learning in the next year? (Select one option) - Selected Choice	All	
17	MLToolNextYearFreeForm	Which tool or technology are you most excited about learning in the next year? (Select one option) - Other - Text	All	
18	MLMethodNextYearSelect	Which ML/DS method are you most excited about learning in the next year? (Select one option) - Selected Choice	All	
19	MLMethodNextYearFreeForm	Which ML/DS method are you most excited about learning in the next year? (Select one option) - Other - Text	All	
20	LanguageRecommendationSelect	What programming language would you recommend a new data scientist learn first? (Select one option) - Selected Choice	All	
21	LanguageRecommendationFreeForm	What programming language would you recommend a new data scientist learn first? (Select one option) - Other - Text	All	
22	PublicDatasetsSelect	Where do you find public datasets to practice data science skills? (Select all that apply) - Selected Choice	All	
23	PublicDatasetsFreeForm	Where do you find public datasets to practice data science skills? (Select all that apply) - Other - Text	All	
24	PersonalProjectsChallengeFreeForm	What is your biggest challenge with the public datasets you find for personal projects?	All	
25	LearningPlatformSelect	What platforms & resources have you used to continue learning data science skills? (Select all that apply) - Selected Choice	All	
26	LearningPlatformCommunityFreeForm	What platforms & resources have you used to continue learning data science skills? (Select all that apply) - Non-Kaggle online communities - Text	All	
27	LearningPlatformFreeForm1	What platforms & resources have you used to continue learning data science skills? (Select all that apply) - Other - Text1	All	
28	LearningPlatformFreeForm2	What platforms & resources have you used to continue learning data science skills? (Select all that apply) - Other - Text2	All	
29	LearningPlatformFreeForm3	What platforms & resources have you used to continue learning data science skills? (Select all that apply) - Other - Text3	All	
30	LearningPlatformUsefulnessArxiv	How useful did you find these platforms & resources for learning data science skills? - Arxiv	All	
31	LearningPlatformUsefulnessBlogs	How useful did you find these platforms & resources for learning data science skills? - Blogs	All	
32	LearningPlatformUsefulnessCollege	How useful did you find these platforms & resources for learning data science skills? - College/University	All	
33	LearningPlatformUsefulnessCompany	How useful did you find these platforms & resources for learning data science skills? - Company internal community	All	
34	LearningPlatformUsefulnessConferences	How useful did you find these platforms & resources for learning data science skills? - Conferences	All	
35	LearningPlatformUsefulnessFriends	How useful did you find these platforms & resources for learning data science skills? - Friends network	All	
36	LearningPlatformUsefulnessKaggle	How useful did you find these platforms & resources for learning data science skills? - Kaggle	All	
37	LearningPlatformUsefulnessNewsletters	How useful did you find these platforms & resources for learning data science skills? - Newsletters	All	
38	LearningPlatformUsefulnessCommunities	How useful did you find these platforms & resources for learning data science skills? - Non-Kaggle online communities	All	

# Understand possibilities!

Paste

Format

**B***I*U

Merge &amp; Center

\$

**Possible Data Loss** Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve them, save as PDF or XLSX.

A1

fx

Column

A

1	Column	Question
2	GenderSelect	Select your gender identity. - Selected Choice
3	GenderFreeForm	Select your gender identity. - A different identity - Text
4	Country	Select the country you currently live in.
5	Age	What's your age?
6	EmploymentStatus	What's your current employment status?
7	StudentStatus	Are you currently enrolled as a student at a degree granting school?
8	LearningDataScience	Are you currently focused on learning data science skills either formally or informally?
9	KaggleMotivationFreeForm	What's your motivation for being a Kaggle user?
10	CodeWriter	Do you write code to analyze data in your current job, freelance contracts, or most recently?
11	CareerSwitcher	Are you actively looking to switch careers to data science?
12	CurrentJobTitleSelect	Select the option that's most similar to your current job/professional title (or most recent)
13	CurrentJobTitleFreeForm	Select the option that's most similar to your current job/professional title (or most recent)
14	TitleFit	How adequately do you feel your title describes what you do (or what you did if retired)?
15	CurrentEmployerType	Which of the following describe your current employer (or most recent employer if retired)?
16	MLToolNextYearSelect	Which tool or technology are you most excited about learning in the next year? (Select all that apply)
17	MLToolNextYearFreeForm	Which tool or technology are you most excited about learning in the next year? (Select all that apply)
18	MLMethodNextYearSelect	Which ML/DS method are you most excited about learning in the next year? (Select all that apply)
19	MLMethodNextYearFreeForm	Which ML/DS method are you most excited about learning in the next year? (Select all that apply)
20	LanguageRecommendationSelect	What programming language would you recommend a new data scientist learn first?
21	LanguageRecommendationFreeForm	What programming language would you recommend a new data scientist learn first?
22	PublicDatasetsSelect	Where do you find public datasets to practice data science skills? (Select all that apply)
23	PublicDatasetsFreeForm	Where do you find public datasets to practice data science skills? (Select all that apply)
24	PersonalProjectsChallengeFreeForm	What is your biggest challenge with the public datasets you find for personal projects?
25	LearningPlatformSelect	What platforms & resources have you used to continue learning data science skills?
26	LearningPlatformCommunityFreeForm	What platforms & resources have you used to continue learning data science skills?
27	LearningPlatformFreeForm1	What platforms & resources have you used to continue learning data science skills?

er	\$	%		.00	.00	Conditional Formatting	Format as Table	Cell Styles	Insert	Delete	Format	 Clear	Sort & Filter
----	----	---	--	-----	-----	---------------------------	--------------------	----------------	--------	--------	--------	--	------------------

o preserve these features, save it in an Excel file format.

Save As

B	C	D
	Asked	
	All	
ly or informally?	Non-worker	
cts, or most recent job if retired?	Non-worker	
title (or most recent title if retired). - Selected Choice	Non-switcher	
title (or most recent title if retired). - Other - Text	Worker1	
you did if retired)?	Worker1	
st employer if retired)? (Select all that apply)	Worker1	
ext year? (Select one option) - Selected Choice	Worker1	
ext year? (Select one option) - Other - Text	All	
year? (Select one option) - Selected Choice	All	
year? (Select one option) - Other - Text	All	
ntist learn first? (Select one option) - Selected Choice	All	
ntist learn first? (Select one option) - Other - Text	All	
ect all that apply) - Selected Choice	All	
ect all that apply) - Other - Text	All	
personal projects?	All	
science skills? (Select all that apply) - Selected Choice	All	
science skills? (Select all that apply) - Non-Kaggle online communities - Text	All	

multipleChoiceResponses

Possible Data Loss Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format.

	A1	GenderSelected	Country	Age	Employment	StudentStatus	LearningData	CodeWriter	CareerSwitch	CurrentJobTitle	TitleFit	CurrentEmployment	MLToolNext	MLMethod	N	Language	Re	PublicDataset	LearningPlatform	LearningPlatform	LearningPlatform	LearningPlatform	Learn		
1	GenderSelected	Country	Age	Employment	StudentStatus	LearningData	CodeWriter	CareerSwitch	CurrentJobTitle	TitleFit	CurrentEmployment	MLToolNext	MLMethod	N	Language	Re	PublicDataset	LearningPlatform	LearningPlatform	LearningPlatform	LearningPlatform	Learn			
2	Non-binary, genderqueer, NA			Employed full-time		Yes			DBA/Database	Fine	Employed by SAS	Base	Random For F#				Dataset	agg	College/University,Conferences,Podcasts,Trade book				Very		
3	Female	United States	30	Not employed, but looking for work							Python	Random For Python						Dataset	agg	Kaggle					
4	Male	Canada	28	Not employed, but looking for work						Amazon Web Services	Deep learning	R	TensorFlow	Neural Nets	Python	I collect my own data	Blogs,College/University,Collaborations,Conferences,Podcasts,Trade book	Very useful	Arxiv,College/University,Conferences,Podcasts,Trade book	Very useful	Arxiv,College/University,Conferences,Podcasts,Trade book	Very useful	Very		
5	Male	United States	56	Independent contractor, freelancer, or self-employed	Yes				Operations	Poorly	Self-employed	TensorFlow	Neural Nets	Python	I collect my own data	Blogs,College/University,Collaborations,Conferences,Podcasts,Trade book	Very useful	Arxiv,College/University,Conferences,Podcasts,Trade book	Very useful	Arxiv,College/University,Conferences,Podcasts,Trade book	Very useful	Very			
6	Male	Taiwan	38	Employed full-time		Yes			Computer Science	Fine	Employed by TensorFlow	Text Mining	Python	GitHub	Arxiv,Conferences,Podcasts,Trade book	Very useful	Dataset	agg	Very useful	Very useful	Very useful	Very useful	Somewhat useful		
7	Male	Brazil	46	Employed full-time		Yes			Data Scientist	Fine	Employed by TensorFlow	Genetic & Evolutionary	Python	Dataset	agg	Kaggle,Online courses,Stack Overflow Q&A,Textbook	Very useful	Dataset	agg	Arxiv,Blogs,Kaggle,Online courses,Stack Overflow Q&A,Textbook	Very useful	Very useful	Very		
8	Male	United States	35	Employed full-time		Yes			Computer Science	Fine	Employed by TensorFlow	Text Mining	R	Dataset	agg	Arxiv,Blogs,Kaggle,Online courses,Stack Overflow Q&A,Textbook	Very useful	Dataset	agg	Arxiv,Blogs,Kaggle,Online courses,Stack Overflow Q&A,Textbook	Very useful	Very useful	Very		
9	Female	India	22	Employed full-time		No	Yes		Software Developer	Fine	Employed by Google Cloud	Deep learning	SQL	Dataset	agg	College/University,Kaggle,Online courses,Stack Overflow Q&A,Textbook	Very useful	Dataset	agg	College/University,Kaggle,Online courses,Stack Overflow Q&A,Textbook	Very useful	Very useful	Very		
10	Female	Australia	43	Employed full-time		Yes			Business Analyst	Fine	Employed by Microsoft Excel	Link Analysis	Python	University/N	Blogs,Company internal,Conferences,Podcasts,Trade book	Very useful	Dataset	agg	Very useful	Very useful	Very useful	Very useful	Very		
11	Male	Russia	33	Employed full-time		Yes			Software Developer	Fine	Employed by C/C++	Deep learning	Python	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Very useful	Somewhat useful		
12	Female	Russia	20	Not employed	Yes	Yes, I'm focused on learning mostly data science skills				Python	Neural Nets	Python	Dataset	agg	Kaggle,Online courses	Very useful	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Very useful	Very useful	Very		
13	Male	India	27	Employed full-time		Yes			Data Scientist	Fine	Employed by Other	Deep learning	Python	Dataset	agg	Kaggle,Non-Kaggle online communities,Personal Projects,YouTube Videos	Very useful	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Very useful	Very		
14	Male	Brazil	26	Employed full-time		No	Yes		Engineer	Fine	Employed by DataRobot	Deep learning	R	Dataset	agg	College/University,Conferences,Kaggle,Personal Projects,Podcasts,Trade book	Very useful	Dataset	agg	College/University,Conferences,Kaggle,Personal Projects,Podcasts,Trade book	Very useful	Somewhat useful	Somewhat useful		
15	Male	Netherlands	54	Employed full-time		No	No		Software Developer	Fine	Employed by TensorFlow	Deep learning	Python	Dataset	agg	Blogs,Conferences,Kaggle,Personal Projects,Podcasts,Trade book	Very useful	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Very useful	Very		
16	Male	Taiwan	26	Employed full-time		Yes			DBA/Database	Poorly	Employed by Python	Rule Induction	R	Dataset	agg	Blogs,Conferences,Kaggle,Personal Projects,Podcasts,Trade book	Very useful	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Very useful	Very		
17	Male	United States	58	Independent contractor, freelancer, or self-employed	Yes				Software Developer	Fine	Employed by TensorFlow	Deep learning	Python	Dataset	agg	Kaggle,Personal Projects,Podcasts,Stack Overflow Q&A,Trade book	Very useful	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Very useful	Very		
18	Male	Italy	58	Employed full-time		No	No		DBA/Database	Poorly	Employed by Python	Rule Induction	R												
19	Male	United Kingdom	24	Employed full-time		No	No																		
20	Male	United States	26	Not employed, but looking for work							TensorFlow	Regression	Python	GitHub	Textbook										
21	Male	Brazil	39	Not employed, but looking for work							Python	Python	Python	University/N	College/University,Textbook,Tutoring	Very useful	Dataset	agg	Very useful	Very useful	Very useful	Very useful	Very		
22	Male	United States	49	Independent contractor, freelancer, or self-employed	No	Yes			Scientist/Researcher	Fine	Self-employed	Amazon Mac	Proprietary	A Java	Google Search	Online courses,Podcasts		Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Very useful	Very useful	Very	
23	Male	United States	25	Employed part-time		Yes			Researcher	Fine	Employed by Amazon Mac	Deep learning	Python	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Very useful	Very		
24	Male	United States	33	Employed full-time		Yes			Scientist/Researcher	Perfectly	Employed by R	Deep learning	Matlab	I collect my own data	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Very useful	Very useful	Very		
25	Male	Czech Republic	21	Employed part-time		Yes			Other	Fine	Employed by R	Deep learning	Python	I collect my own data	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Very useful	Very useful	Very		
26	Male	United States	NA	Employed full-time		Yes			Software Developer	Fine	Employed by Spark / MLlib	Deep learning	Matlab	Dataset	agg	Online courses,Personal Projects,Textbook	Very useful	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Very useful	Very		
27	Male	Russia	22	Employed full-time		Yes			Data Analyst	Fine	Employed by TensorFlow	Genetic & Evolutionary	Python	Dataset	agg	Arxiv,College,Very useful	Very useful	Dataset	agg	Arxiv,College,Very useful	Very useful	Very useful	Very		
28	Male	Netherlands	51	Employed full-time		Yes			Engineer	Poorly	Employed by I don't plan	I don't plan	C R	I collect my own data	Blogs,Tutoring/mentoring	Very useful	Dataset	agg	Arxiv,College,Very useful	Very useful	Very useful	Very useful	Very		
29	Male	Colombia	34	Employed full-time		Yes			Data Scientist	Fine	Employed by Spark / MLlib	Ensemble	M Python	Google Search	Online courses,Personal Projects,Stack Overflow Q&A	Very useful	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Very useful	Very useful	Very		
30	Male	Germany	41	Independent contractor, freelancer, or self-employed	Yes				Data Scientist	Fine	Self-employed	I don't plan	C Factor	Analy Python	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Somewhat useful	Somewhat useful	
31	Female	Canada	32	Not employed	Yes	Yes, I'm focused on learning mostly data science skills							Amazon Web Services	Genetic & Evolutionary	C/C++/C#	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Very useful	Very
32	Male	Denmark	53	Employed full-time		Yes			Business Analyst	Fine	Employed by professional	Proprietary	A Python	Dataset	agg	Blogs,Friends network,Personal Projects,Textbook	Very useful	Dataset	agg	Blogs,Friends network,Personal Projects,Textbook	Very useful	Very useful	Very		
33	Male	Poland	29	Employed full-time		Yes			Software Developer	Fine	Employed by TensorFlow	Deep learning	Python	Dataset	agg	Kaggle,Online courses,Personal Projects,Stack Overflow Q&A,Textbook	Very useful	Dataset	agg	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Very useful	Very		
34	Male	United Kingdom	36	Employed full-time		Yes			Data Scientist	Poorly	Employed by Microsoft Azure	Proprietary	A Python	University/N	Arxiv,Blogs,Collaborations,Conferences,Podcasts,Trade book	Very useful	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Very useful	Very useful	Very		
35	Male	Russia	34	Employed full-time		Yes			Machine Learning	Perfectly	Employed by Python	Deep learning	Python	Google Search	Arxiv,College,Somewhat useful	Very useful	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Very useful	Very useful	Very		
36	Male	United States	35	Employed full-time		Yes			Engineer	Fine	Employed by Spark / MLlib	Deep learning	Python	Dataset	agg	Kaggle,Online courses,Personal Projects,Stack Overflow Q&A,Textbook	Very useful	Dataset	agg	Arxiv,College,Somewhat useful	Very useful	Very useful	Very		

Some answers are well defined (multiple choice).

AutoSave OFF

multipleC

Home Insert Page Layout Formulas Data Review View

Cut Calibri (Body) 12 A A = = = Wrap Text

Copy Paste Format B I U = = = Merge & Center

Possible Data Loss Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To pre

A1 Formula Bar GenderSelect

	A	B	C	D	E	F	G	H	I	J
1	GenderSelect	Country	Age	Employment	StudentStatus	LearningData	CodeWriter	CareerSwitch	CurrentJobTitle	TitleFit
2	Non-binary, genderqueer,	NA		Employed full-time			Yes		DBA/Database	Fine
3	Female	United States	30	Not employed, but looking for work						
4	Male	Canada	28	Not employed, but looking for work						
5	Male	United States	56	Independent contractor, freelancer, or self-employed	Yes				Operations	Poorly
6	Male	Taiwan	38	Employed full-time		Yes			Computer Sc	Fine
7	Male	Brazil	46	Employed full-time		Yes			Data Scientis	Fine
8	Male	United States	35	Employed full-time		Yes			Computer Sc	Fine
9	Female	India	22	Employed full-time		No	Yes		Software Dev	Fine
10	Female	Australia	43	Employed full-time		Yes			Business Ana	Fine
11	Male	Russia	33	Employed full-time		Yes			Software Dev	Fine
12	Female	Russia	20	Not employed	Yes	Yes, I'm focused on learning mostly data science skills				
13	Male	India	27	Employed full-time		Yes			Data Scientis	Fine
14	Male	Brazil	26	Employed full-time		No	Yes	Engineer		Fine
15	Male	Netherlands	54	Employed full-time		No	No			
16	Male	Taiwan	26	Employed full-time		Yes			Software Dev	Fine
17	Male	United States	58	Independent contractor, freelancer, or self-employed	Yes			DBA/Database	Poorly	
18	Male	Italy	58	Employed full-time		No	No			
19	Male	United Kingdom	24	Employed full-time		No	No			



## multipleChoiceResponses



Wrap Text



Merge & Center ▾

General

\$

▼

%

,

← .0  
.00

.00  
→ .0

-delimited (.csv) format. To preserve these features, save it in an E

I

J

K

L

M

switch CurrentJobTi TitleFit

CurrentEmpl MLToolNext\ MLMETHODN

freeformResponses

This screenshot shows a Microsoft Excel spreadsheet titled "freeformResponses". The data is organized into columns A through T and rows 1 through 35. The first few rows contain general headers and some placeholder text. Rows 2 through 10 are grouped under the heading "teacher". Rows 11 through 19 are grouped under "PyTorch". Rows 20 through 24 are grouped under "Curious". Rows 25 through 27 are grouped under "Promote our". Rows 28 through 34 are grouped under "Analytics Vidya, DataCamp, Machinelearning, and Machinelearning". Row 35 contains a single entry: "half man - half dog". The data is mostly composed of "NA" entries, indicating missing or undefined responses.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	
1	GenderFree	KaggleMotiv	CurrentJobTi	MLToolNext\	MLMethodN	LanguageRe	PublicDatabase	PersonalProj	LearningPlat	LearningPlat	LearningPlat	LearningPlat	LearningPlat	LearningPlat	LearningPlat	LearningPlat	LearningPlat	LearningPlat	BlogsPodcas	Jobs
2									Data manipulation		NA	NA	NA	NA	NA				NA	None
3									I can't find time to practice consistently		NA	NA	NA	NA	NA				NA	
4									Meetups		NA	NA	NA	NA	NA				NA	
5									Connectivity/data fusion		NA	NA	NA	NA	NA				NA	
6											NA	NA	NA	NA	NA				NA	
7									kdnuggets	Prepping data		NA	NA	NA	NA				NA	
8											NA	NA	NA	NA	NA				NA	
9									Stanford SNAP		NA	NA	NA	NA	NA				NA	
10											NA	NA	NA	NA	NA				NA	
11									PyTorch		NA	NA	NA	NA	NA				NA	
12											NA	NA	NA	NA	NA				NA	
13											NA	NA	NA	NA	NA				NA	
14											NA	NA	NA	NA	NA				NA	
15											NA	NA	NA	NA	NA				NA	
16											NA	NA	NA	NA	NA				NA	
17											NA	NA	NA	NA	NA				NA	
18											NA	NA	NA	NA	NA				NA	
19											NA	NA	NA	NA	NA				NA	
20									Curious		NA	NA	NA	NA	NA				NA	
21									Hydrographic Surveyor		NA	NA	NA	NA	NA				NA	
22											NA	NA	NA	NA	NA				NA	
23										Poor data quality / lack of documentation		NA	NA	NA	NA				NA	
24											NA	NA	NA	NA	NA				NA	
25									mechanical engineer	don't know		NA	NA	NA	NA				NA	
26									Promote our	Technical support engineer		NA	NA	NA	NA				NA	
27											NA	NA	NA	NA	NA				NA	
28											NA	NA	NA	NA	NA				NA	
29										Crawling Airbnb		NA	NA	NA	NA				NA	
30											NA	NA	NA	NA	NA				NA	
31										Amount and quality of data		NA	NA	NA	NA				NA	machinelearning
32											NA	NA	NA	NA	NA				NA	
33											NA	NA	NA	NA	NA				NA	Analytics Vidya, DataCamp
34									half man - half dog	None in specific.		NA	NA	NA	NA				NA	Machinelearning
35											NA	NA	NA	NA	NA				NA	

Some answers are not well defined (free form).

24

25 mechanical engineer

26 Promote our Technical support engineer

27

28

29

30

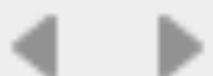
31

32 Gender (free form).

33

34 half man - half dog

35



freeformResponses



Ready

```
In [1]: %matplotlib inline
```

```
In [2]: cd '/Users/tstoeger/Dropbox/Work/kaggle_survey'
```

```
/Users/tstoeger/Dropbox/Work/kaggle_survey
```

```
In [3]: import pandas as pd  
import seaborn as sns
```

```
In [1]: %matplotlib inline
```

```
In [2]: cd '/Users/tstoeger/Dropbox/Work/kaggle_survey'  
/Users/tstoeger/Dropbox/Work/kaggle_survey
```

```
In [3]: import pandas as pd  
import seaborn as sns
```

```
In [4]: df_multiple_choice = pd.read_csv(  
    './multipleChoiceResponses.csv',  
    encoding='latin-1', # files have some special characters  
    low_memory=False) # some columns have mixed type
```

```
In [1]: %matplotlib inline
```

```
In [2]: cd '/Users/tstoeger/Dropbox/Work/kaggle_survey'  
/Users/tstoeger/Dropbox/Work/kaggle_survey
```

```
In [3]: import pandas as pd  
import seaborn as sns
```

```
In [4]: df_multiple_choice = pd.read_csv(  
    './multipleChoiceResponses.csv',  
    encoding='latin-1', # files have some special characters  
    low_memory=False) # some columns have mixed type
```

```
In [5]: df_multiple_choice.shape # get idea of number of datasets
```

```
Out[5]: (16716, 228)
```

```
In [1]: %matplotlib inline
```

```
In [2]: cd '/Users/tstoeger/Dropbox/Work/kaggle_survey'  
/Users/tstoeger/Dropbox/Work/kaggle_survey
```

```
In [3]: import pandas as pd  
import seaborn as sns
```

```
In [4]: df_multiple_choice = pd.read_csv(  
    './multipleChoiceResponses.csv',  
    encoding='latin-1', # files have some special characters  
    low_memory=False) # some columns have mixed type
```

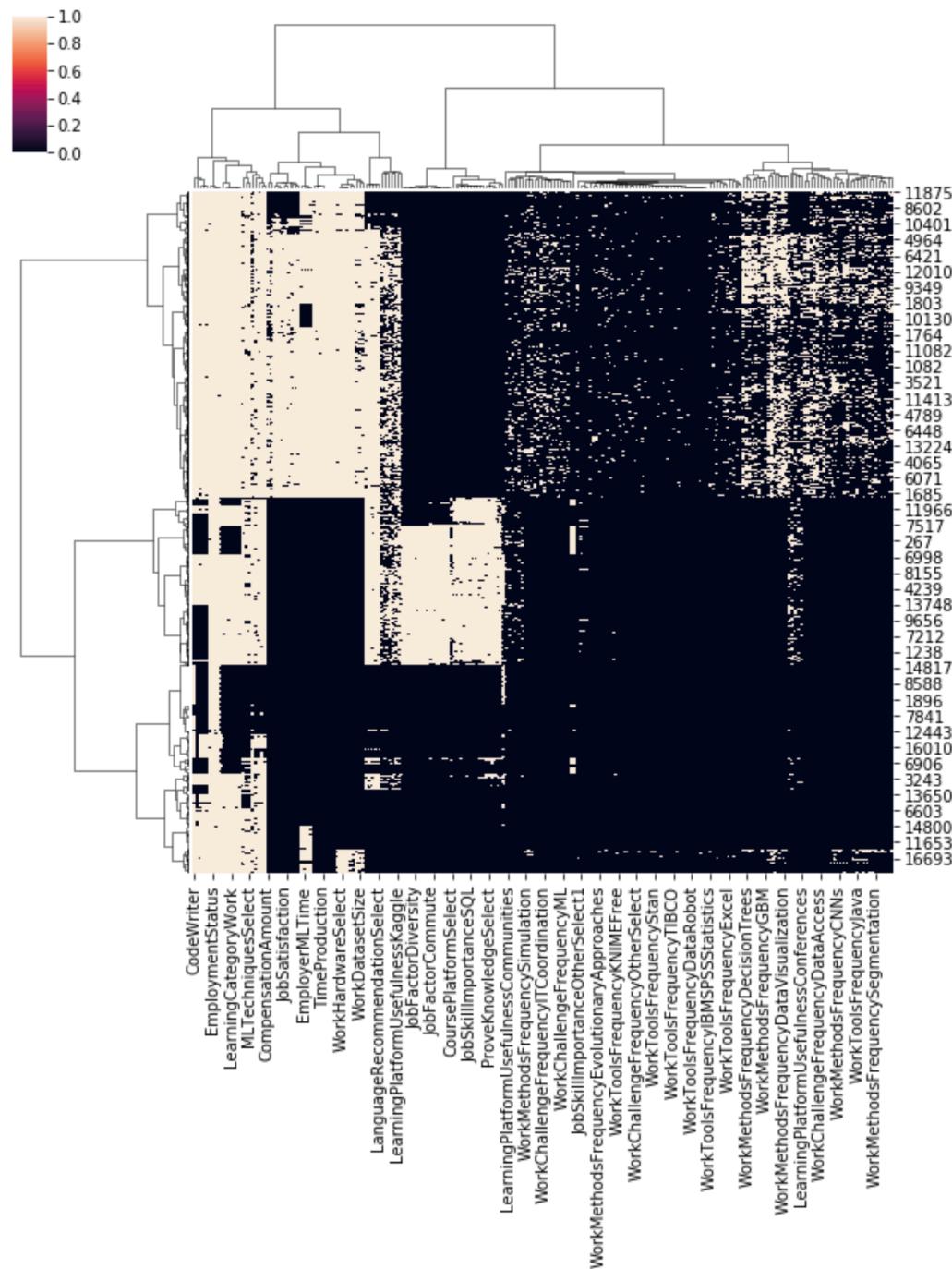
```
In [5]: df_multiple_choice.shape # get idea of number of datasets
```

```
Out[5]: (16716, 228)
```

```
In [6]: sns.clustermap(# visualize presence of data  
    df_multiple_choice.notnull(),  
    method='ward')
```

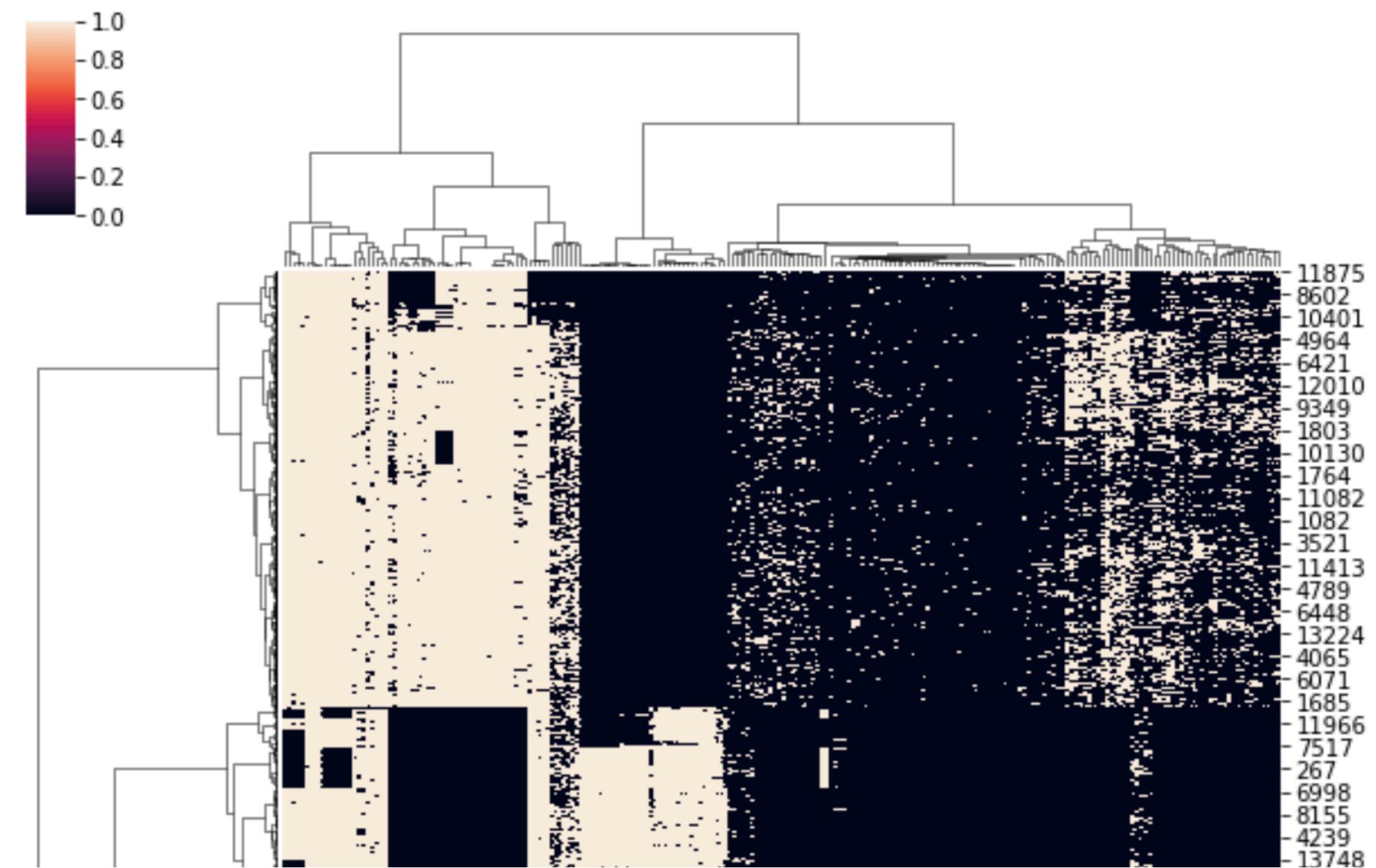
```
In [6]: sns.clustermap(           # visualize presence of data
                      df_multiple_choice.notnull(),
                      method='ward')
```

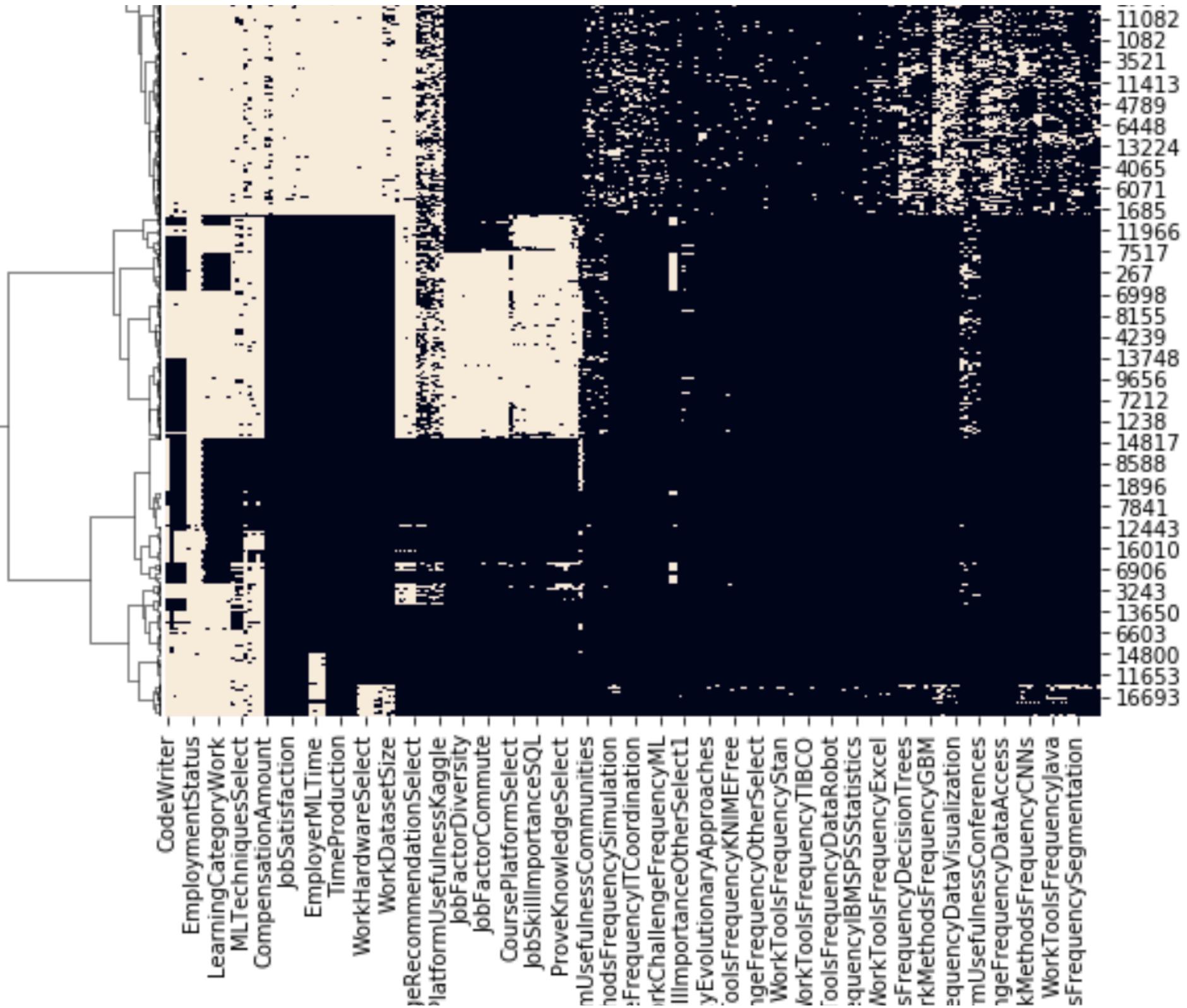
```
Out[6]: <seaborn.matrix.ClusterGrid at 0x10cd02a58>
```



```
sns.clustermap( # visualize presence of data
    df_multiple_choice.notnull(),
    method='ward' )
```

```
<seaborn.matrix.ClusterGrid at 0x10cd02a58>
```





**Attention. There will be a small test in three slides.**

# Always look at data!

In [7]: df\_multiple\_choice

Out[7]:

	GenderSelect	Country	Age	EmploymentStatus	StudentStatus	LearningDataScience
0	Non-binary, genderqueer, or gender non- conforming		NaN	NaN	Employed full-time	NaN
1	Female	United States	30.0			NaN
2	Male	Canada	29.0			NaN
3	Male	United States	56.0	free		NaN
4	Male	Taiwan	38.0	Employed full-time		NaN
5	Female	India	22.0	Employed full-time		NaN
6	Female	Australia	43.0	Employed full-time		NaN
7	Female	India	22.0	Employed full-time		NaN
8	Female	Australia	43.0	Employed full-time		NaN
9	Male	Russia	33.0	Employed full-time		NaN

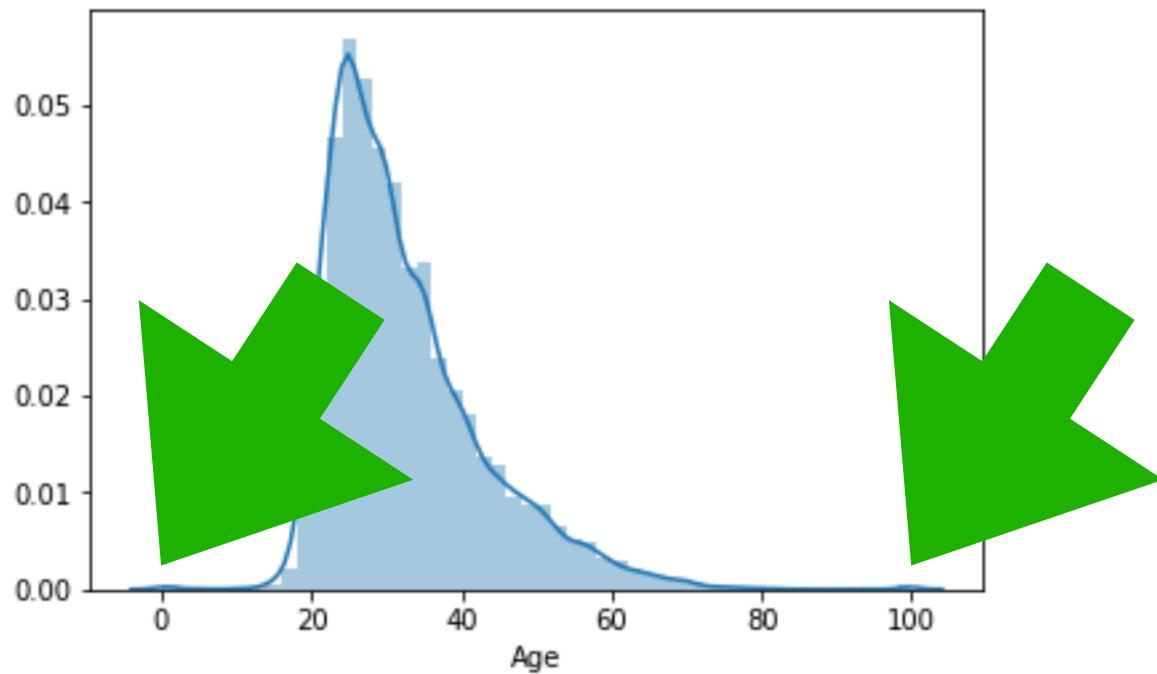
In a nicely organized data set, columns are organized in a meaningful way

```
In [8]: sns.distplot(  
    df_multiple_choice[ 'Age' ].dropna(),  
    )
```

Sanity-check numbers, and  
test for unusual values.

```
In [8]: sns.distplot(  
    df_multiple_choice[ 'Age' ].dropna(),  
    )
```

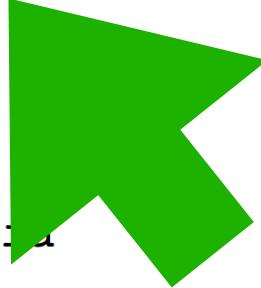
```
Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x1a124307f0>
```



# Sanity-check numbers, and test for unusual values.

```
In [11]: df_multiple_choice['Country'].value_counts()
```

```
Out[11]: United States          4197  
India                  2704  
Other                  1023  
Russia                 578  
United Kingdom          535  
People's Republic of China 471  
Brazil                 465  
Germany                460  
France                 442  
Canada                 440  
Australia              421  
Spain                  320  
...
```



**Always get aware of spelling conventions!**

**Sanity-check occurrences of categorial data.**

```
In [10]: df_multiple_choice['GenderSelect'].value_counts()
```

```
Out[10]: Male           13610  
Female          2778  
A different identity    159  
Non-binary, genderqueer, or gender non-conforming    74  
Name: GenderSelect, dtype: int64
```

**Test: Given the observed gender frequency,  
why do we have to be suspicious about some  
of the data seen in one of the recent slides?**

**Sanity-check occurrences  
of categorial data.**

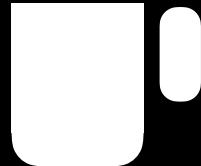
# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

← **Here**

# Organization

- Data
- Code
- Computation



# Environment (very short)

- Language
- Computer
- Space
- Constant learning

# Summary

**What did we miss?**

**After lecture (extra): getting  
your environment ready**

*Data science is 90% data cleaning and  
10% complaining about data cleaning.*

# PersonalProjectsChallenge

this workbook in the comma-delimited (.csv) format. To preserve these features

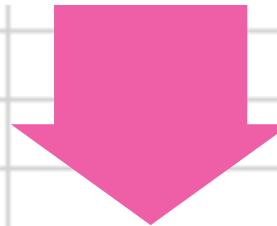


F	G	I	J	K	L
languageRev	PublicDataset	PersonalProj	LearningPlat	LearningPlat	LearningPlat
		Data manipulation			
		I can't find time to practice consistently			
			Meetups		
		Connectivity/data fusion			
kdnuggets	Prepping data				
	Stanford SNAP				

You are not alone.

# Personal Projects Challenge

You are not alone.



Poor data quality / lack of documentation

Don't know

Crawling Airbnb

Amount and quality of data

None in specific.



# Main tactics for data cleaning.

- Regular expressions are your best friend to only clean what you want to clean.
- Use errors (codified sanity checks) to your advantage.
- Fuzzy matching and entity matching help you to combine related data sources, which do not have a shared identifier.

# *“Let us explain the salary of data scientists!”*



```
In [13]: # For the sake of this example, let us only consider  
# salaries that have been converted to USD  
f = df_multiple_choice['CompensationCurrency'] == 'USD'  
df_multiple_choice = df_multiple_choice[f]
```

```
In [13]: # For the sake of this example, let us only consider  
# salaries that have been converted to USD  
f = df_multiple_choice['CompensationCurrency'] == 'USD'  
df_multiple_choice = df_multiple_choice[f]
```

```
In [14]: # For the sake of this example, let us only consider  
# records, where CompensationAmount is defined  
# (we still keep records, if something else, e.g.  
# StudentStatus, would be not defined)  
df_multiple_choice = df_multiple_choice.dropna(  
    subset=['CompensationAmount'])  
)
```

```
In [13]: # For the sake of this example, let us only consider  
# salaries that have been converted to USD  
f = df_multiple_choice['CompensationCurrency'] == 'USD'  
df_multiple_choice = df_multiple_choice[f]
```

```
In [14]: # For the sake of this example, let us only consider  
# records, where CompensationAmount is defined  
# (we still keep records, if something else, e.g.  
# StudentStatus, would be not defined)  
df_multiple_choice = df_multiple_choice.dropna(  
    subset=['CompensationAmount'])  
)
```

```
In [15]: # Manually inspect the data in CompensationAmount  
df_multiple_choice['CompensationAmount']
```

```
Out[15]: 3      250,000  
21      20000  
22      100000  
34      133000  
37      80000  
61      15000  
75      215000  
...      ...
```

```
# Manually inspect the data  
df_multiple_choice['Compensa
```

3	250,000
21	200
22	10000
34	13300
37	80000
61	15000
75	215000
86	83500

```
In [16]: # Let us convert these values to a number  
df_multiple_choice[ 'CompensationAmount' ] = df_multiple_choice[  
    'CompensationAmount' ].astype( float )
```

# Error

```
# Let us convert these values to a number
df_multiple_choice['CompensationAmount'] = df_multiple_choice[
    'CompensationAmount'].astype(float)

-----
ValueError                                Traceback (most recent call last)
<ipython-input-16-f163d2412d49> in <module>()
      1 df_multiple_choice['CompensationAmount'] = df_multiple_choice[
----> 2     'CompensationAmount'].astype(float)

~/anaconda3/lib/python3.6/site-packages/pandas/util/_decorators.py in wrapper(*args, **kwargs)
   176         else:
   177             kwargs[new_arg_name] = new_arg_value
--> 178         return func(*args, **kwargs)
   179     return wrapper
   180 return _deprecate_kwarg

~/anaconda3/lib/python3.6/site-packages/pandas/core/construction.py in astype(self, dtype, copy, errors, **kwargs)
  4995         # else, only a single dtype is given
  4996         new_data = self._data.astype(dtype=dtype, copy=copy, errors=errors,
--> 4997                         **kwargs)
  4998         return self._constructor(new_data).__finalize__(self)
  4999

~/anaconda3/lib/python3.6/site-packages/pandas/core/frame.py in astype(self, dtype, copy, errors, **kwargs)
  3712     def astype(self, dtype, **kwargs):
--> 3714         return self.apply('astype', dtype=dtype, **kwargs)
  3715
  3716     def convert(self, **kwargs):

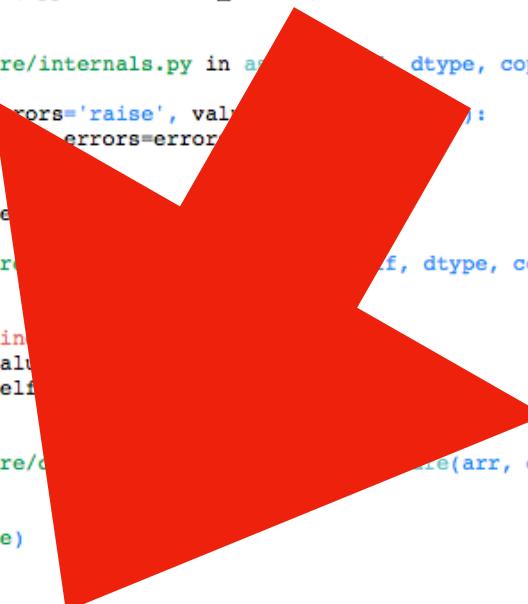
~/anaconda3/lib/python3.6/site-packages/pandas/core/internal.py in apply(self, f, axis, skipna, convert_dtype, errors, **kwargs)
  3579         kwargs['mgr'] = self
--> 3581         applied = getattr(b, f)(**kwargs)
  3582         result_blocks = _extend_blocks(applied, result_blocks)
  3583

~/anaconda3/lib/python3.6/site-packages/pandas/core/internals.py in astype(self, dtype, copy, errors, values, **kwargs)
  573     def astype(self, dtype, copy=False, errors='raise', values=None):
  574         return self._astype(dtype, copy=copy, errors=errors, values=values):
--> 575         **kwargs)
  576
  577     def _astype(self, dtype, copy=False, errors='raise', values=None):

~/anaconda3/lib/python3.6/site-packages/pandas/core/internal.py in _astype(self, dtype, copy, errors, values, klass, mgr, **kwargs)
  662
  663         # _astype_nansafe works fine here
--> 664         values = astype_nansafe(values, dtype)
  665         values = values.reshape(self.shape)
  666

~/anaconda3/lib/python3.6/site-packages/pandas/core/dtypes/creation.py in astype(arr, dtype, copy)
  728
  729     if copy:
--> 730         return arr.astype(dtype, copy=True)
  731     return arr.view(dtype)
  732

ValueError: could not convert string to float: '85,000'
```



# Only fix what you want to fix. (85,000 and similar)

```
ValueError: could not convert string to float: '85,000'
```

```
In [17]: f = df_multiple_choice['CompensationAmount'].str.contains(  
    '[0-9]*,[0-9]{3}$$') # create a highly specific regular expression
```

```
In [18]: df_multiple_choice.loc[f, 'CompensationAmount'] = df_multiple_choice.loc[  
    f, 'CompensationAmount'].str.replace(',', '')
```

## Can someone translate this?

[0-9]\*,[0-9]{3}\$\$

## Trick question: Could this be shorter to solve our problem?

```
In [19]: df_multiple_choice['CompensationAmount'] = df_multiple_choice['CompensationAmount'].astype(float)
```

# Cleaning data is an interactive process

```
In [19]: df_multiple_choice['CompensationAmount'] = df_multiple_choice['CompensationAmount'].astype(float)

-----
ValueError                                Traceback (most recent call last)
<ipython-input-19-f163d2412d49> in <module>()
      1 df_multiple_choice['CompensationAmount'] = df_multiple_choice[
--> 2     'CompensationAmount'].astype(float)

~/anaconda3/lib/python3.6/site-packages/pandas/util/_decorators.py in wrapper(*args, **kwargs)
    176         else:
    177             kwargs[new_arg_name] = new_arg_value
--> 178         return func(*args, **kwargs)
    179     return wrapper
    180 return _deprecate_kwarg

~/anaconda3/lib/python3.6/site-packages/pandas/core/generic.py in astype(self, dtype, copy, errors, **kwargs)
    4995         # else, only a single dtype is given
    4996         new_data = self._data.astype(dtype=dtype, copy=copy, errors=errors,
--> 4997                         **kwargs)
    4998         return self._constructor(new_data).__finalize__(self)
    4999

~/anaconda3/lib/python3.6/site-packages/pandas/core/internals.py in astype(self, dtype, **kwargs)
    3712
    3713     def astype(self, dtype, **kwargs):
--> 3714         return self.apply('astype', dtype=dtype, **kwargs)
    3715
    3716     def convert(self, **kwargs):

~/anaconda3/lib/python3.6/site-packages/pandas/core/internals.py in apply(self, f, axes, filter, do_integrity_check,
consolidate, **kwargs)
    3579
    3580         kwargs['mgr'] = self
--> 3581         applied = getattr(b, f)(**kwargs)
    3582         result_blocks = _extend_blocks(applied, result_blocks)
    3583

~/anaconda3/lib/python3.6/site-packages/pandas/core/internals.py in astype(self, dtype, copy, errors, values, **kwargs)
    573     def astype(self, dtype, copy=False, errors='raise', values=None, **kwargs):
    574         return self._astype(dtype, copy=copy, errors=errors, values=values,
--> 575                         **kwargs)
    576
    577     def _astype(self, dtype, copy=False, errors='raise', values=None,
```

# Main tactics for data cleaning.

- Regular expressions are your best friend to only clean what you want to clean.
- Use errors (codified sanity checks) to your advantage.
- Fuzzy matching and entity matching help you to combine related data sources, which do not have a shared identifier.

e.g: match Thomas Stoeger to Thomas Stoger



# Missing data can be tricky.

Advance Access publication August 22, 2016

*Political Analysis* (2016) 24:414–433  
doi:10.1093/pan/mpw020

## How Multiple Imputation Makes a Difference

Ranjit Lall

*Department of Government, Harvard University, 1737 Cambridge Street, Cambridge, MA 02138*  
*e-mail:* ranjitlall@fas.harvard.edu (corresponding author)

Edited by R. Michael Alvarez

Political scientists increasingly recognize that multiple imputation represents a superior strategy for analyzing missing data to the widely used method of listwise deletion. However, there has been little systematic investigation of how multiple imputation affects existing empirical knowledge in the discipline. This article presents the first large-scale examination of the empirical effects of substituting multiple imputation for listwise deletion in political science. The examination focuses on research in the major subfield of comparative and international political economy (CIPE) as an illustrative example. Specifically, I use multiple imputation to reanalyze the results of almost every quantitative CIPE study published during a recent five-year period in *International Organization* and *World Politics*, two of the leading subfield journals in CIPE. The outcome is striking: in almost half of the studies, key results “disappear” (by conventional statistical standards) when reanalyzed.

# A rough guide to missing data:

**Random**

Discard records

Impute based on  
similar records

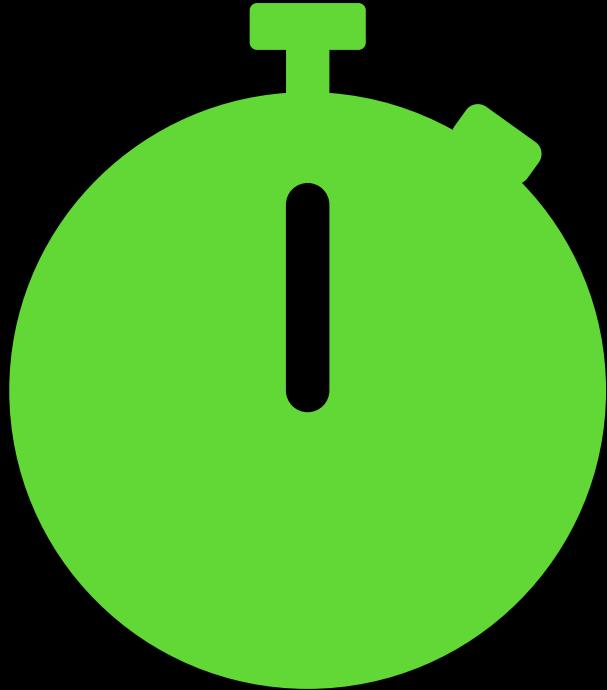
**Non-random**

Use as  
information

↓  
Consider as  
“below measurement  
limit”

Advanced (often  
domain-specific)  
imputation

# **Three minute exercise.**



**Talk to your neighbor.**

**Thinking of your worst  
encounter with data.**

**What would have been, or  
was, a good strategy to  
clean the data or avoid  
unwanted surprises.**

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

Here  
is the  
fun



# Organization

- Data
- Code
- Computation

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

# Summary

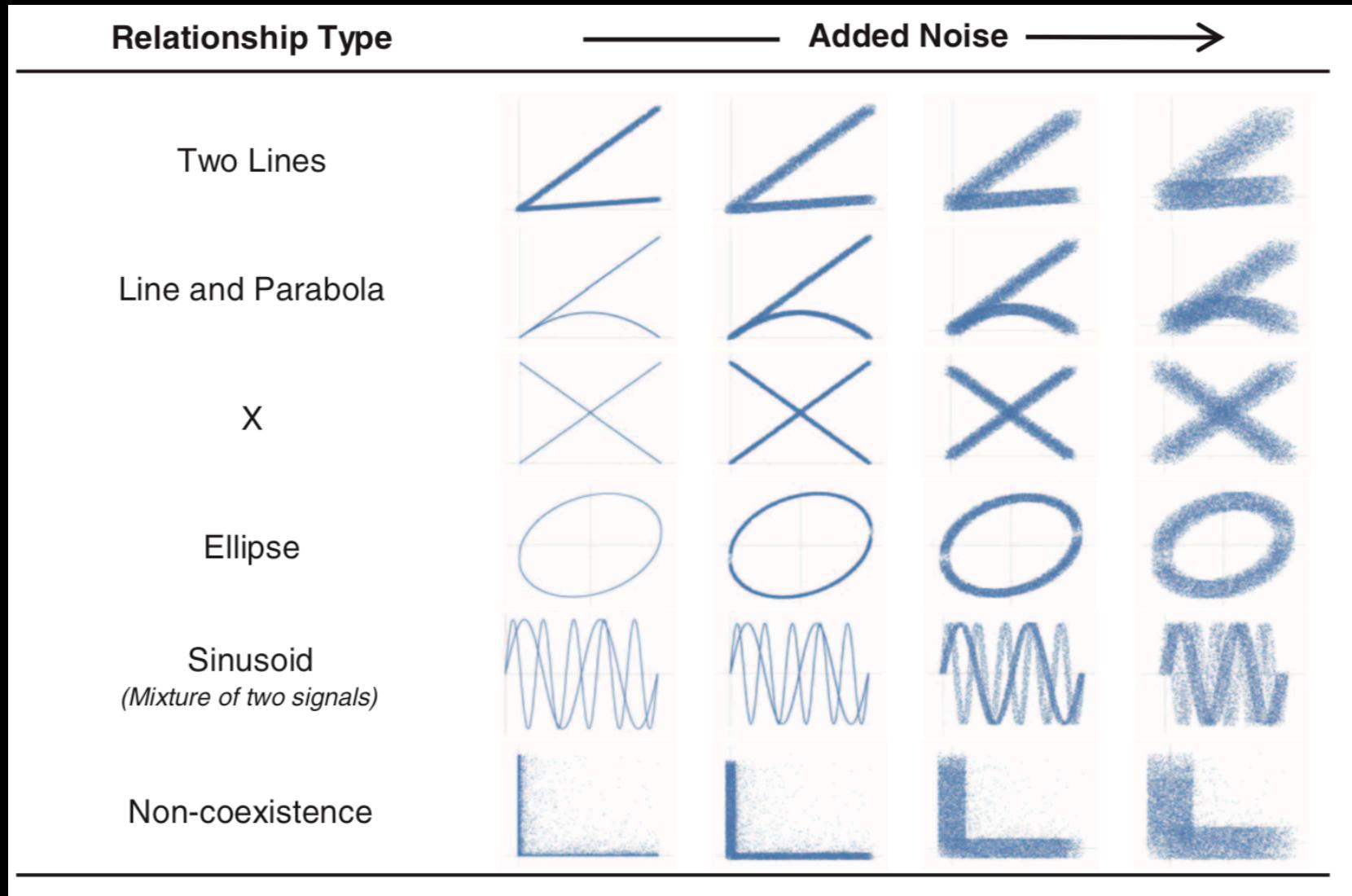
What did we miss?

After lecture (extra): getting  
your environment ready

# Main tactics for finding connections.

- Correlations
- Exploratory visualization
- Machine learning

# Distinct types of correlations can embody distinct assumptions on the data.



Reshef et al. 2011

# Distinct types of correlations can embody distinct assumptions on the data.

Relationship Type	MIC	Pearson	Spearman	Mutual Information (KDE)	Mutual Information (Kraskov)	CorGC (Principal Curve-Based)	Maximal Correlation
Random	0.18	-0.02	-0.02	0.01	0.03	0.19	0.01
Linear	1.00	1.00	1.00	5.03	3.89	1.00	1.00
Cubic	1.00	0.61	0.69	3.09	3.12	0.98	1.00
Exponential	1.00	0.70	1.00	2.09	3.62	0.94	1.00
Sinusoidal (Fourier frequency)	1.00	-0.09	-0.09	0.01	-0.11	0.36	0.64
Categorical	1.00	0.53	0.49	2.22	1.65	1.00	1.00
Periodic/Linear	1.00	0.33	0.31	0.69	0.45	0.49	0.91
Parabolic	1.00	-0.01	-0.01	3.33	3.15	1.00	1.00
Sinusoidal (non-Fourier frequency)	1.00	0.00	0.00	0.01	0.20	0.40	0.80
Sinusoidal (varying frequency)	1.00	-0.11	-0.11	0.02	0.06	0.38	0.76

Reshef et al. 2011

# Exploratory visualization can give an idea of magnitudes.

```
In [108]: f = (
    df_multiple_choice['GenderSelect'].isin(['Male', 'Female'])) & (
    df_multiple_choice['Country'].isin(['United States'])) & (
    df_multiple_choice['EmploymentStatus'].isin(['Employed full-time'])) & (
    df_multiple_choice['Age'].isin(range(20, 35))) & (
    df_multiple_choice['CurrentJobTitleSelect'].isin(['Data Scientist']))
)
```

# Exploratory visualization can give an idea of magnitudes.

```
In [108]: f = (
    df_multiple_choice['GenderSelect'].isin(['Male', 'Female'])) & (
    df_multiple_choice['Country'].isin(['United States'])) & (
    df_multiple_choice['EmploymentStatus'].isin(['Employed full-time'])) & (
    df_multiple_choice['Age'].isin(range(20, 35))) & (
    df_multiple_choice['CurrentJobTitleSelect'].isin(['Data Scientist']))
)
```

```
In [109]: import numpy as np
```

```
In [110]: sns.pointplot(
    x='Age',
    y='CompensationAmount',
    data=df_multiple_choice[f],
    estimator=np.median,
    hue='GenderSelect',
    dodge=True
)
```

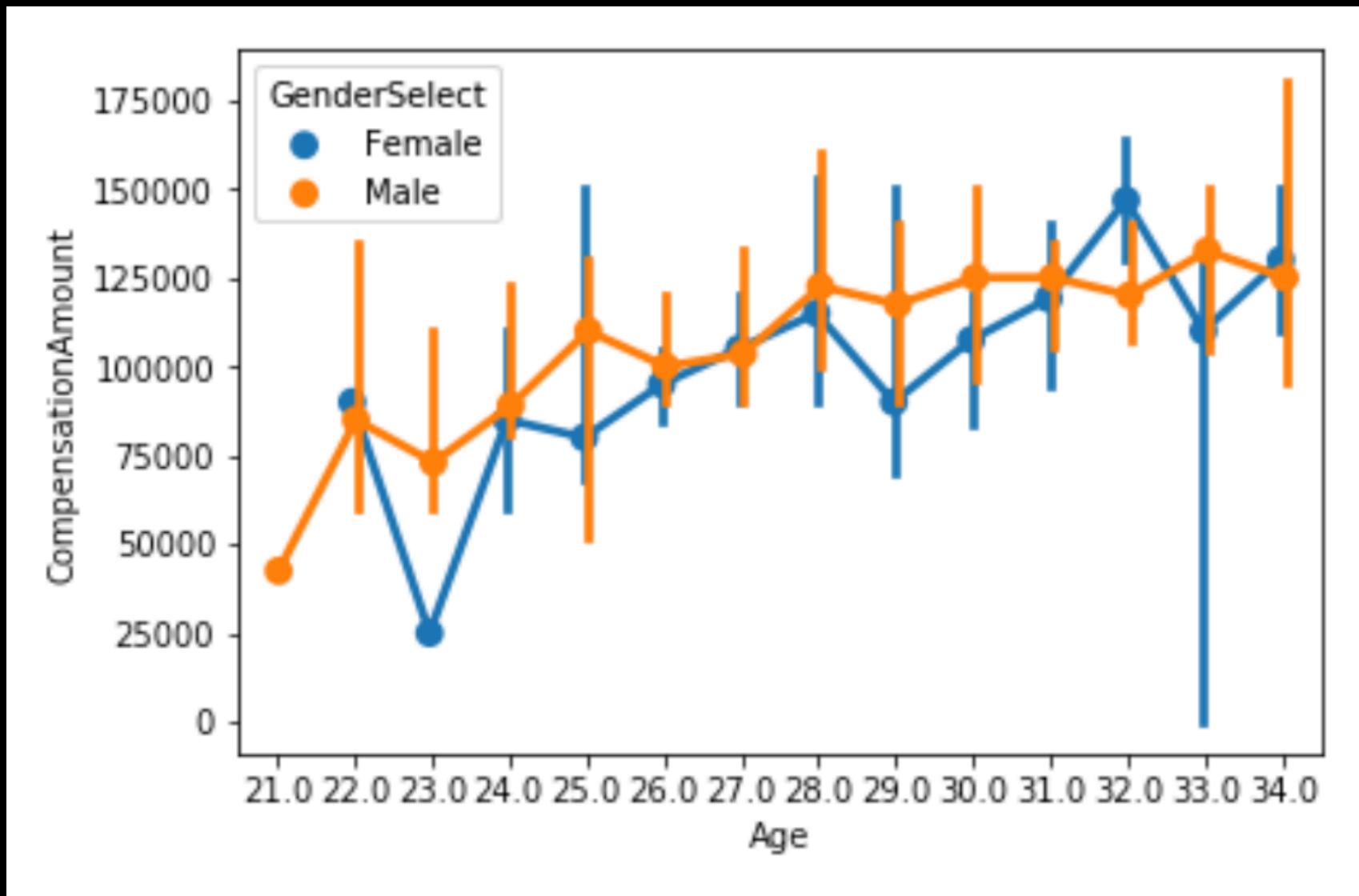
# Exploratory visualization can give an idea of magnitudes.

```
In [108]: f = (
    df_multiple_choice['GenderSelect'].isin(['Male', 'Female'])) & (
    df_multiple_choice['Country'].isin(['United States'])) & (
    df_multiple_choice['EmploymentStatus'].isin(['Employed full-time'])) & (
    df_multiple_choice['Age'].isin(range(20, 35))) & (
    df_multiple_choice['CurrentJobTitleSelect'].isin(['Data Scientist']))
)
```

```
In [109]: import numpy as np
```

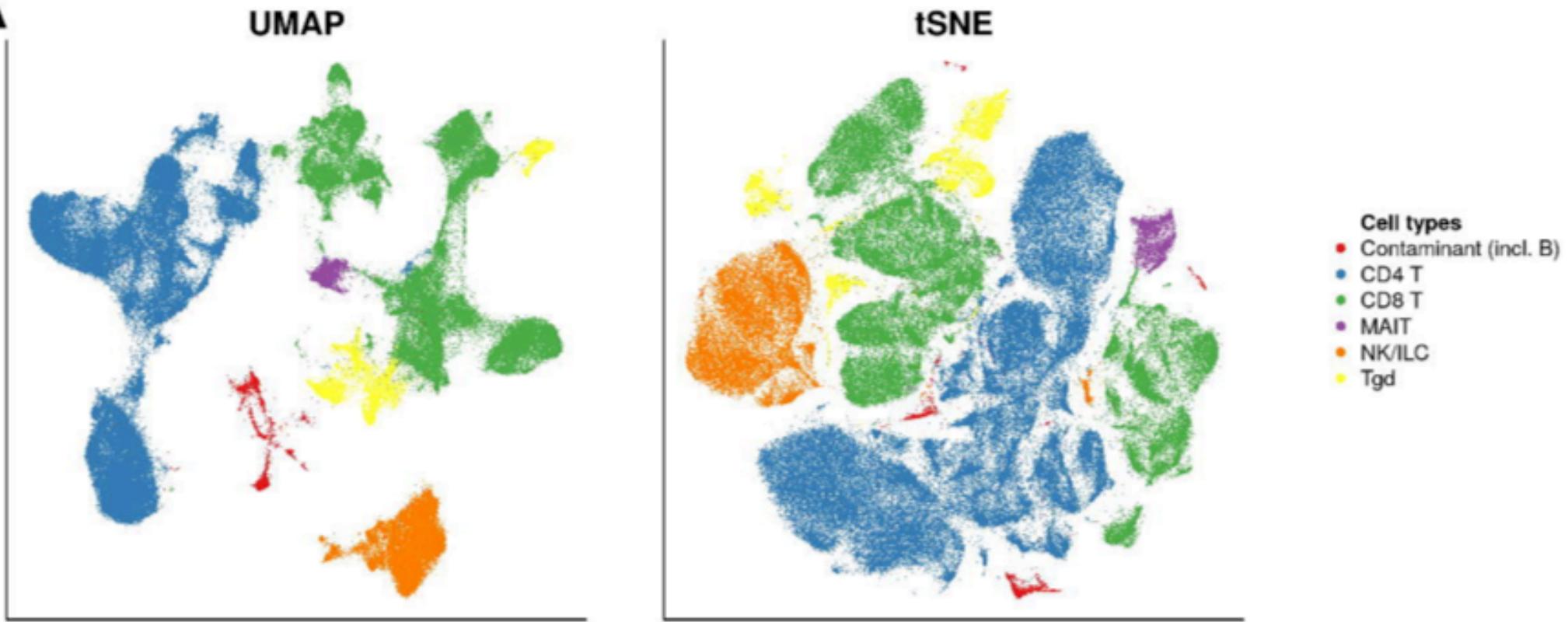
```
In [110]: sns.pointplot(
    x='Age',
    y='CompensationAmount',
    data=df_multiple_choice[f],
    estimator=np.median,
    hue='GenderSelect',
    dodge=True
)
```

# Median salary of data scientists at university ages.



# Extra tactic: visualize upon dimensionality reduction.

A



Dimensionality reduction  
Fast

Only visualization  
Slow

Becht et al., bioRxiv 2018

# Main tactics for finding connections.

- Correlations
- Exploratory visualization
- Machine learning

# Reminder.

```
In [113]: df_multiple_choice['MLMethodNextYearSelect'].value_counts()
```

```
Out[113]: Deep learning                               535
          Neural Nets                                207
          Bayesian Methods                            101
          Time Series Analysis                      90
          Genetic & Evolutionary Algorithms        66
          Anomaly Detection                           61
          Text Mining                                 52
          Social Network Analysis                   48
          Other                                      46
          Ensemble Methods (e.g. boosting, bagging) 42
          I don't plan on learning a new ML/DS method 33
          Cluster Analysis                            31
          Monte Carlo Methods                         28
          Support Vector Machines (SVM)              26
          Proprietary Algorithms                      21
          Random Forests                             18
          Survival Analysis                          18
          Regression                                 15
          Rule Induction                            8
          Link Analysis                             8
          Decision Trees                            8
          Uplift Modeling                           6
          MARS                                     5
          Factor Analysis                           5
          Association Rules                        3
Name: MLMethodNextYearSelect, dtype: int64
```

# Any question?

# **How to learn about relations through predictions?**

- 1) Through contribution to model. E.g.: linear regression, or random forest, or neural nets.**

# **How to learn about relations through predictions?**

- 1) Through contribution to model. E.g.: linear regression, or random forest, or neural nets.**
- 2) Through ability to predict.**

# How to find best modeling strategy?

Generalist

Domain-specific

# Finding strategies through generalist approaches can be automated.

## auto-sklearn

The screenshot shows the GitHub repository page for 'automl / auto-sklearn'. The repository has 1,948 commits and 9 branches. A prominent pull request by 'mfeurer' titled 'Merge pull request #495 from automl/development' is listed at the top. The commit message is 'Prepare new release'. Below this, there is a list of other pull requests and commits, each with a brief description and timestamp. The commits include updates to documentation, CI scripts, examples, and Dockerfiles.

Commit Message	Time Ago
mfeurer Merge pull request #495 from automl/development	2 months ago
ci_scripts MAINT update circle install	3 months ago
doc Prepare new release	2 months ago
examples MAINT maintenance updates	3 months ago
scripts FIX bugs	9 months ago
test Fix CI (#494)	2 months ago
.gitignore Remove unnecessary files	3 years ago
.landscape.yaml ADD landscape	3 years ago
.travis.yml CI automatically test examples	7 months ago
COPYING Add license file	3 years ago
Dockerfile Add Dockerfile	a year ago
LICENSE.txt Add documentation	2 years ago
MANIFEST.in FIX uri in setup, MAINT cleanup MANIFEST.in	2 years ago
Makefile The test directory is called "test", not tests	2 years ago
README.md ADD codecov badge	a year ago
circle.vml FIX nin conda version to older conda due to conda/conda#6030	10 months ago

Beats >95% of data scientists

Two lines of code.

Combines all standard and advanced machine learning approaches.

Uses machine learning to find nearly optimal data normalization and machine learning strategy.

# Some science-specific approaches have very interesting properties.

TIME SERIES ANALYSIS

## Information leverage in interconnected ecosystems: Overcoming the curse of dimensionality

Turn high dimensionality from a curse to a blessing.

Hao Ye and George Sugihara\*

In ecological analysis, complexity has been regarded as an obstacle to overcome. Here we present a straightforward approach for addressing complexity in dynamic interconnected systems. We show that complexity, in the form of multiple interacting components, can actually be an asset for studying natural systems from temporal data. The central idea is that multidimensional time series enable system dynamics to be reconstructed from multiple viewpoints, and these viewpoints can be combined into a single model. We show how our approach, multiview embedding (MVE), can improve forecasts for simulated ecosystems and a mesocosm experiment. By leveraging complexity, MVE is particularly effective for overcoming the limitations of short and noisy time series and should be highly relevant for many areas of science.

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

# Organization

- Data
- Code
- Computation

# Summary

**What did we miss?**

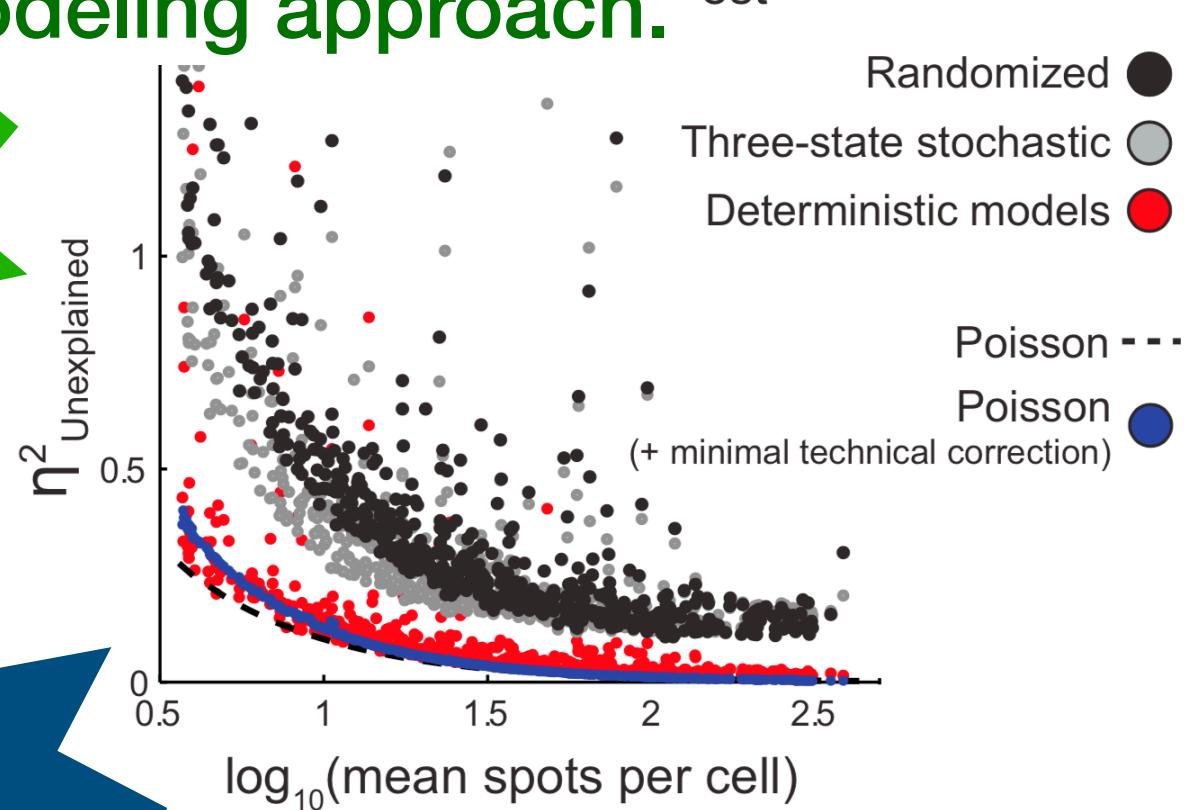
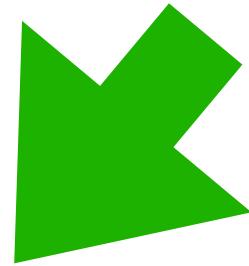
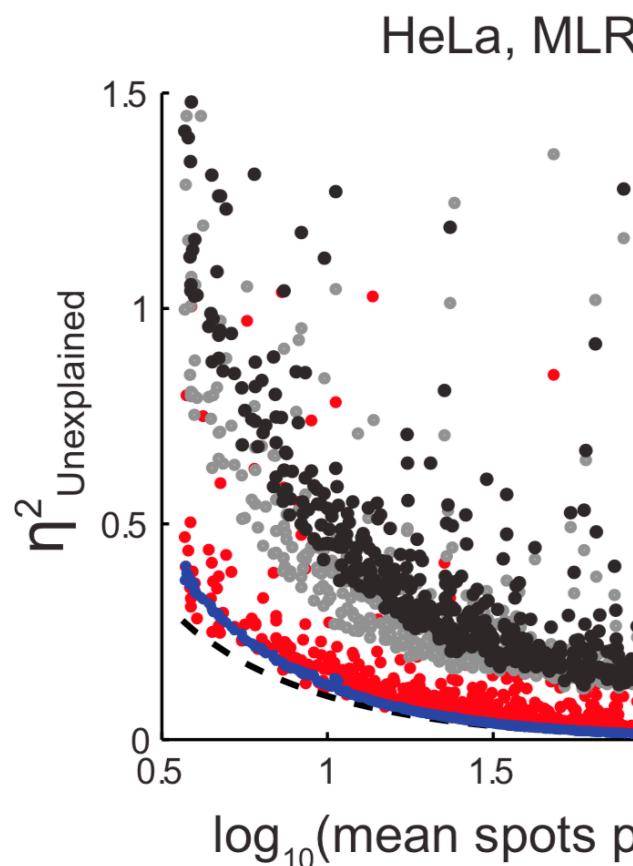
**After lecture (extra): getting  
your environment ready**



**Pass reviewers.**

We wanted to say that the red dots are as low as possible (little remained unexplained by our models).

Choose the “worse” modeling approach.



If some data isn't explained yet, test your performance against a problem-specific theoretical model.

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

**What did we miss?**

JO

# **Environment (very short)**

- Language
- Computer
- Space
- Constant learning

## Language

- Images: Matlab
- Otherwise: Python
- Maybe: R, Excel, Tableau

## Computer

- RAM bottlenecks will have strongest impact on your waiting time (if you have more RAM than needed, you can write inefficient, but possibly more readable, code)
- Consider a fast single core performance as it will mean less waiting time than multiple cores, if your code isn't optimized to multiple cores (and no workstation can compete with a cluster).
- If possible: get a desktop and a small laptop.
- Avoid isolated rooms.

## Space

- Books, scientific literature; e.g.: note that NU has license for huge ebook collection on data science through Safari Online
- Meetups

**Please object. As some of this is scientific field specific.**

Some shameless advertising.

The screenshot shows a website for "DATA SCIENCE NIGHTS @NORTHWESTERN". The header includes the URL "data-science-nights.org" and a navigation bar with links for HOME, ABOUT, GETTING STARTED WITH DATA SCIENCE, DATA SCIENCE AROUND CAMPUS, and ARCHIVES. The main section features a large blue background image of people at a table, overlaid with the text "DATA SCIENCE NIGHTS". Below this, a detailed description of the events is provided, followed by a welcome message at the bottom.

data-science-nights.org

DATA SCIENCE NIGHTS  
@NORTHWESTERN

HOME    ABOUT    GETTING STARTED WITH DATA SCIENCE    DATA SCIENCE AROUND CAMPUS    ARCHIVES

# DATA SCIENCE NIGHTS

MONTHLY HACK NIGHTS ON POPULAR DATA SCIENCE TOPICS, ORGANIZED BY FELLOWS AND SCHOLARS FROM THE NORTHWESTERN DATA SCIENCE INITIATIVE. EACH NIGHT WILL FEATURE REFRESHMENTS, A **TALK ON DATA SCIENCE TECHNIQUES OR APPLICATIONS**, AND A **HACKING NIGHT** WITH DATA SCIENCE PROJECTS OR LEARNING GROUPS OF YOUR CHOICE.

ASPIRING, BEGINNING, AND ADVANCED DATA SCIENTISTS ARE WELCOME!

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

# Organization

- Data
- Code
- Computation



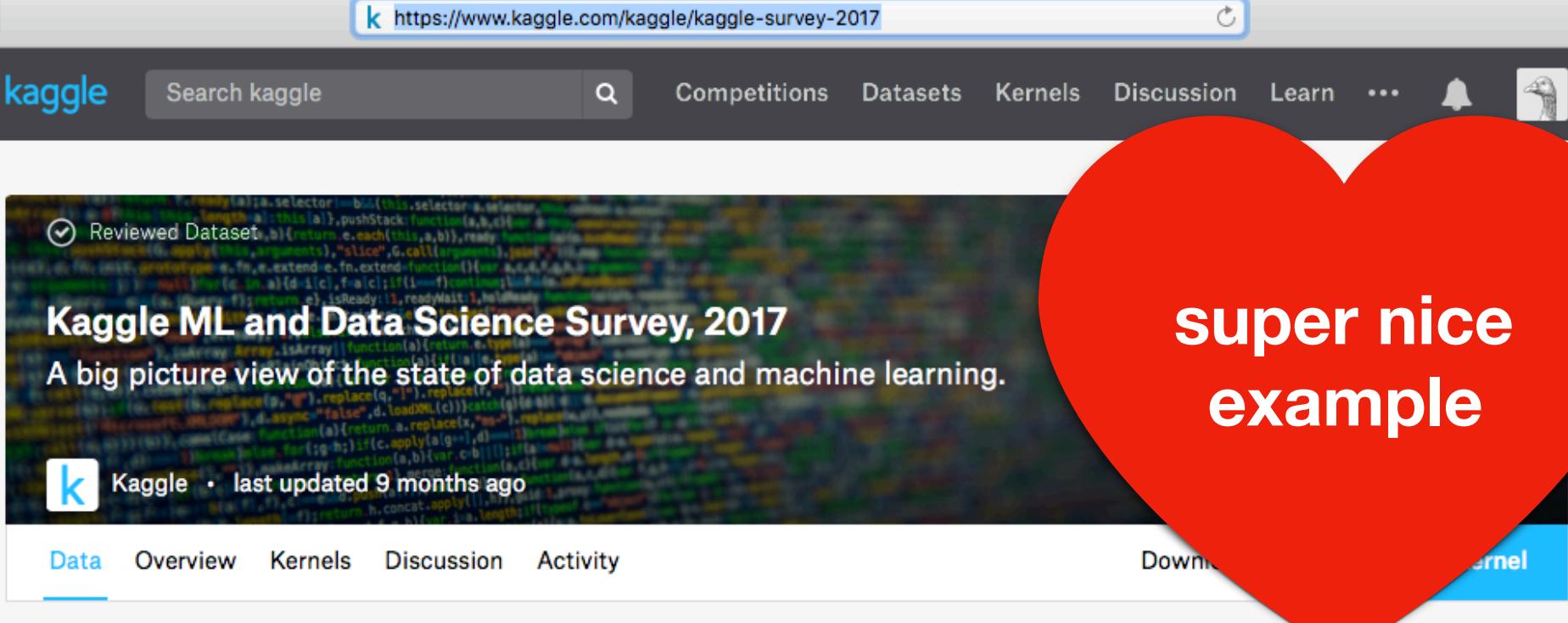
**Here**

What did we miss?

After lecture (extra): getting  
your environment ready

# Introducing Kaggle ML and Data Science Survey, 2017

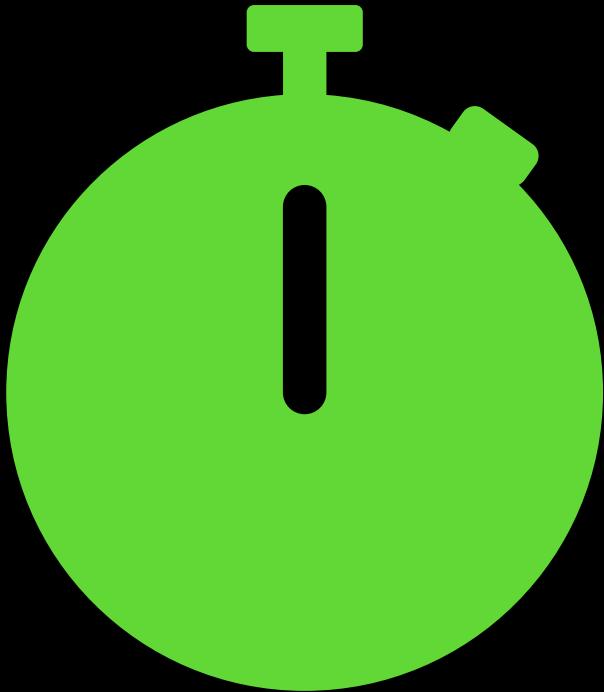
## An example data set, which we will be using



A screenshot of a web browser displaying the Kaggle website at <https://www.kaggle.com/kaggle/kaggle-survey-2017>. The page title is "Kaggle ML and Data Science Survey, 2017" and the subtitle is "A big picture view of the state of data science and machine learning." A large red heart graphic with the text "super nice example" is overlaid on the bottom right of the screenshot.

The screenshot shows the following interface elements:

- Header: kaggle, Search kaggle, Q, Competitions, Datasets, Kernels, Discussion, Learn, ..., Bell icon, Profile icon.
- Page Title: Kaggle ML and Data Science Survey, 2017
- Page Subtitle: A big picture view of the state of data science and machine learning.
- Dataset Overview:
  - Reviewed Dataset (checkbox checked)
  - Kaggle logo, last updated 9 months ago
  - Data, Overview, Kernels, Discussion, Activity tabs (Data is selected)
  - Download, Kernel buttons



## Three minute exercise.

**Find a neighbor, or  
neighbors, whom  
you do not know.**

**Then recall tactics applied  
by Kaggle to create an  
organized data set.**

Logical layout.

# Portability and access.

Know your options.

# **Portability and access.**

**Do I want to work without internet access?**

**With whom do I collaborate?**

**Is data structured?**

# Portability and access.

Central location

Database

Folder with datasets

- Domain-specific online host (e.g.: NIH's Synapse)

Local copies

- Have one single reference

- e.g.: have reference synced over Dropbox across Workstations, and unidirectionally copy to NU's cluster (but never copy data back from cluster)

**Logical layout.**

**Portability and access.**

**Know your options.**

**CSV**  
**Spreadsheets**

**HDF5**  
**Selectively load**  
**Parts of data**  
**(avoid breaking**  
**cluster)**

**MSGPACK**  
**Very fast reading**  
**and writing of**  
**entire tables**

**SQL**  
**Elegant query**  
**language**

```
In [ ]: df_multiple_choice.to_
In [ ]: df_multiple_choice.to_clipboard
In [ ]: df_multiple_choice.to_csv
In [ ]: df_multiple_choice.to_dense
In [ ]: df_multiple_choice.to_dict
In [ ]: df_multiple_choice.to_excel
In [ ]: df_multiple_choice.to_feather
In [ ]: df_multiple_choice.to_gbq
In [ ]: df_multiple_choice.to_hdf
In [ ]: df_multiple_choice.to_html
In [ ]: df_multiple_choice.to_json
```

```
In [ ]: df_multiple_choice.to_
In [ ]: df_multiple_choice.to_latex
In [ ]: df_multiple_choice.to_msgpack
In [ ]: df_multiple_choice.to_panel
In [ ]: df_multiple_choice.to_parquet
In [ ]: df_multiple_choice.to_period
In [ ]: df_multiple_choice.to_pickle
In [ ]: df_multiple_choice.to_records
In [ ]: df_multiple_choice.to_sparse
In [ ]: df_multiple_choice.to_sql
In [ ]: df_multiple_choice.to_stata
```

```
In [172]: example_data = df_multiple_choice[['Country', 'Age', 'GenderSelect']].dropna()
```

```
In [173]: example_data.head()
```

Out[173]:

	Country	Age	GenderSelect
1	United States	30.0	Female
2	Canada	28.0	Male
3	United States	56.0	Male
4	Taiwan	38.0	Male
5	Brazil	46.0	Male

HDF5

```
In [174]: example_data.to_hdf(  
    './example.h5',  
    format='table',  
    key='searchable_data',  
    mode='w',  
    data_columns=True,  
)
```

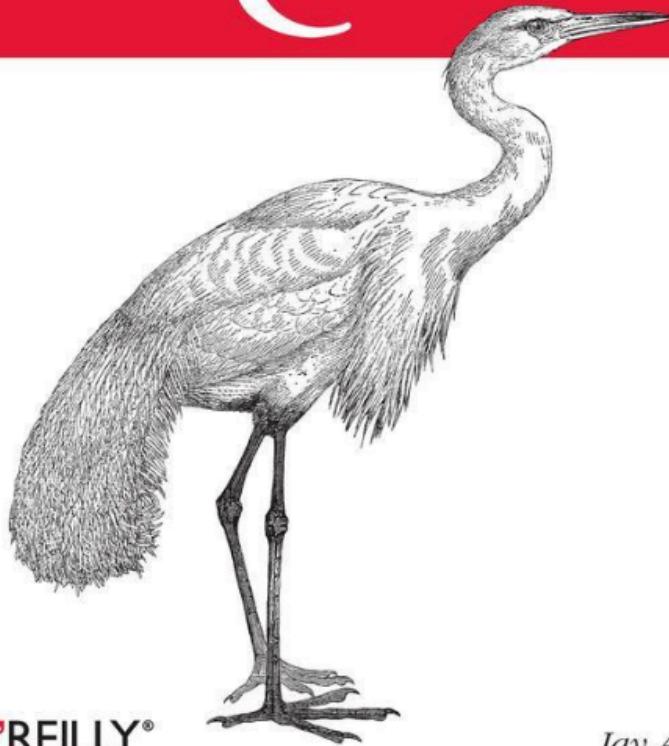
```
In [175]: pd.read_hdf('./example.h5', 'searchable_data', where=['Age=20'])
```

Out[175]:

	Country	Age	GenderSelect
10	Russia	20.0	Female
123	United States	20.0	Female
125	Turkey	20.0	Male
144	United States	20.0	Male
148	India	20.0	Male
...	-	--	--

*Using*

# SQLite



O'REILLY®

*Jay A. Kreibich*

## Remember

You can have a server-less database to store and organize data.

NU provides tons of free ebooks.

# **Some NU specific considerations:**

**Data transfer to Quest: Globus (also has command line option)**

**Box: has some file restrictions and sometimes doesn't copy all files.  
Doesn't work on all (old) Linux, Mac, Windows-> recommendation: Instead  
pay for Dropbox Plus with automated backup (~150 USD per year)**

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

# Organization

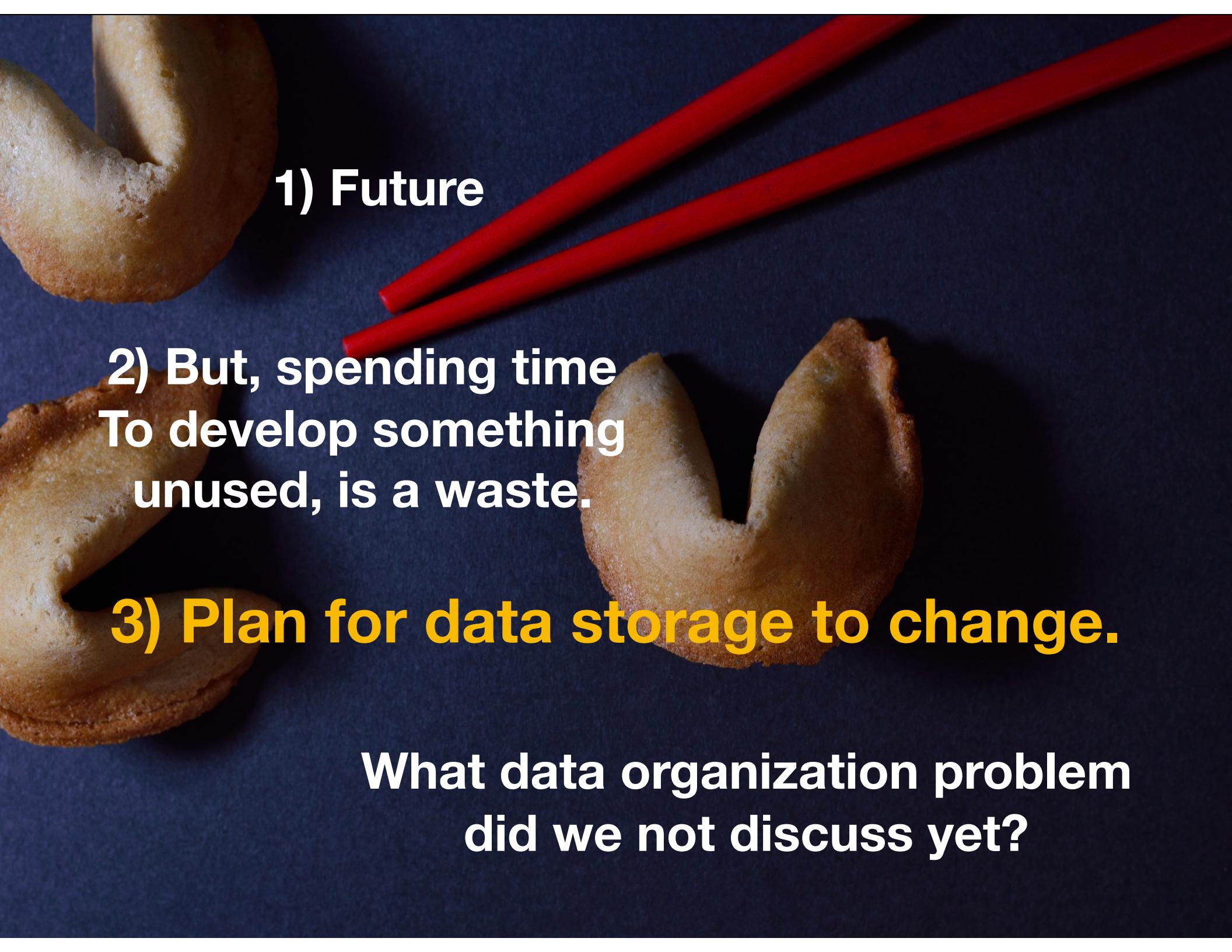
- Data
- Code
- Computation



**Here**

What did we miss?

After lecture (extra): getting  
your environment ready

- 
- The background of the slide features a dark, textured surface. In the upper left corner, a single fortune cookie is partially visible. In the upper right corner, two red chopsticks are positioned diagonally. The text is overlaid on this image.
- 1) Future
  - 2) But, spending time  
To develop something  
unused, is a waste.
  - 3) Plan for data storage to change.

What data organization problem  
did we not discuss yet?



# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

# Organization

- Data
  - Code
  - Computation
- ← Here

# Summary

What did we miss?

After lecture (extra): getting  
your environment ready

**Plan for changing data.**

**Very brief coding practices.**

# **Plan for changing data.**

**Very brief coding practices.**

## **Functions:**

- **organize code**
- **Reusability**



## **Adaptor:**

- **Can change data resource  
without worrying in notebook**

# Direct loading is perfectly fine for new datasets.

```
In [1]: %matplotlib inline
```

```
In [2]: cd '/Users/tstoeger/Dropbox/Work/kaggle_survey'  
/Users/tstoeger/Dropbox/Work/kaggle_survey
```

```
In [3]: import pandas as pd  
import seaborn as sns
```

```
In [4]: df_multiple_choice = pd.read_csv(  
    './multipleChoiceResponses.csv',  
    encoding='latin-1', # files have some special characters  
    low_memory=False) # some columns have mixed type
```

```
In [5]: df_multiple_choice.shape # get idea of number of datasets
```

```
Out[5]: (16716, 228)
```

# ... but move your loading into adaptor functions in the long run.

```
In [190]: def load_country_gender_at_given_age(age_of_interest):
    """
        Loads the country and gender of survey takes, which
        are of a given age

    Input:
        age_of_interest int

    Output:
        dataframe

    """
    df = pd.read_csv(
        './multipleChoiceResponses.csv',
        encoding='latin-1', # files have some special characters
        low_memory=False) # some columns have mixed type
    df = df[['Country', 'Age', 'GenderSelect']].dropna()
    df = df[df['Age']==age_of_interest]

    return df
```

Define function  
In human digestible  
Manner

# ... but move your loading into adaptor functions in the long run.

```
In [190]: def load_country_gender_at_given_age(age_of_interest):
    """
        Loads the country and gender of survey takes, which
        are of a given age

        Input:
            age_of_interest    int

        Output:
            dataframe

    """

    df = pd.read_csv(
        './multipleChoiceResponses.csv',
        encoding='latin-1', # files have some special characters
        low_memory=False)   # some columns have mixed type
    df = df[['Country', 'Age', 'GenderSelect']].dropna()
    df = df[df['Age']==age_of_interest]

    return df
```

```
In [191]: d = load_country_gender_at_given_age(20)
```

```
In [192]: d.head()
```

```
Out[192]:
```

	Country	Age	GenderSelect
10	Russia	20.0	Female
123	United States	20.0	Female

```
In [193]: def load_country_gender_at_given_age(age_of_interest):
    """
    Loads the country and gender of survey takes, which
    are of a given age

    Input:
        age_of_interest int

    Output:
        dataframe

    """
    df = pd.read_hdf('./example.h5', 'searchable_data', where=['Age={}'.format(
        int(age_of_interest))])

    return df
```



# Changed to HDF5

```
In [193]: def load_country_gender_at_given_age(age_of_interest):
    """
    Loads the country and gender of survey takes, which
    are of a given age

    Input:
        age_of_interest    int

    Output:
        datafram

    """
    df = pd.read_hdf('./example.h5', 'searchable_data', where=['Age={}'.format(
        int(age_of_interest))])

    return df
```

```
In [194]: d = load_country_gender_at_given_age(20)
```

```
In [195]: d.head()
```

Out[195]:

	Country	Age	GenderSelect
10	Russia	20.0	Female
123	United States	20.0	Female
125	Turkey	20.0	Male
144	United States	20.0	Male
148	India	20.0	Male

**Plan for changing data.**

**Very brief coding practices.**

# **Reduce mistakes.**

**Aim for: Everything important on same screen page.**

**Meaningful active names, unless variable  
used in same paragraph, e.g.: “is\_stored”**

**Avoid duplications.**

**Use GitHub to track code and collaborate (free private repositories through their educational service); every ~20 minutes. Consider GUI (GitHub desktop) rather than command line.**

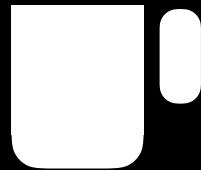
# **Any further recommendation?**

**Plan for changing data.**

**Very brief coding practices.**

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs



# Organization

- Data
- Code
- Computation

← Here

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

# Summary

**What did we miss?**

**After lecture (extra): getting  
your environment ready**

**Save time!**

**Reduce time coding.**

**Reduce time running  
code.**

# **Save time!**

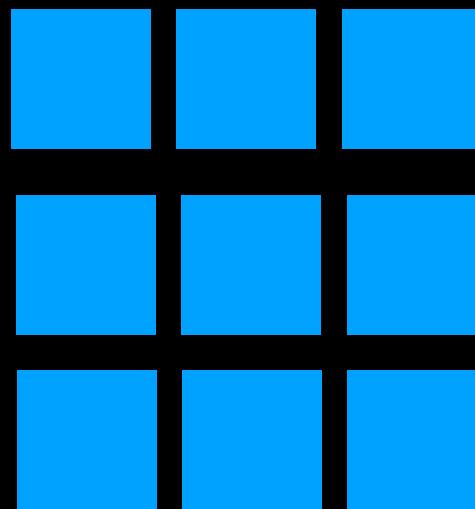
**Important to many.**

**Reduce time coding.**

## **Specialist applications.**

Reduce time running  
code.

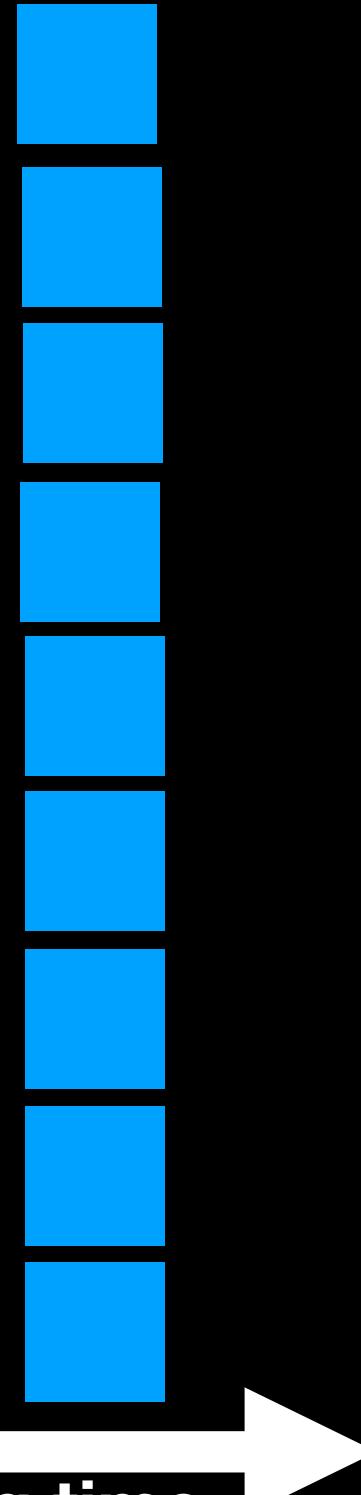
# How to organize many computations?



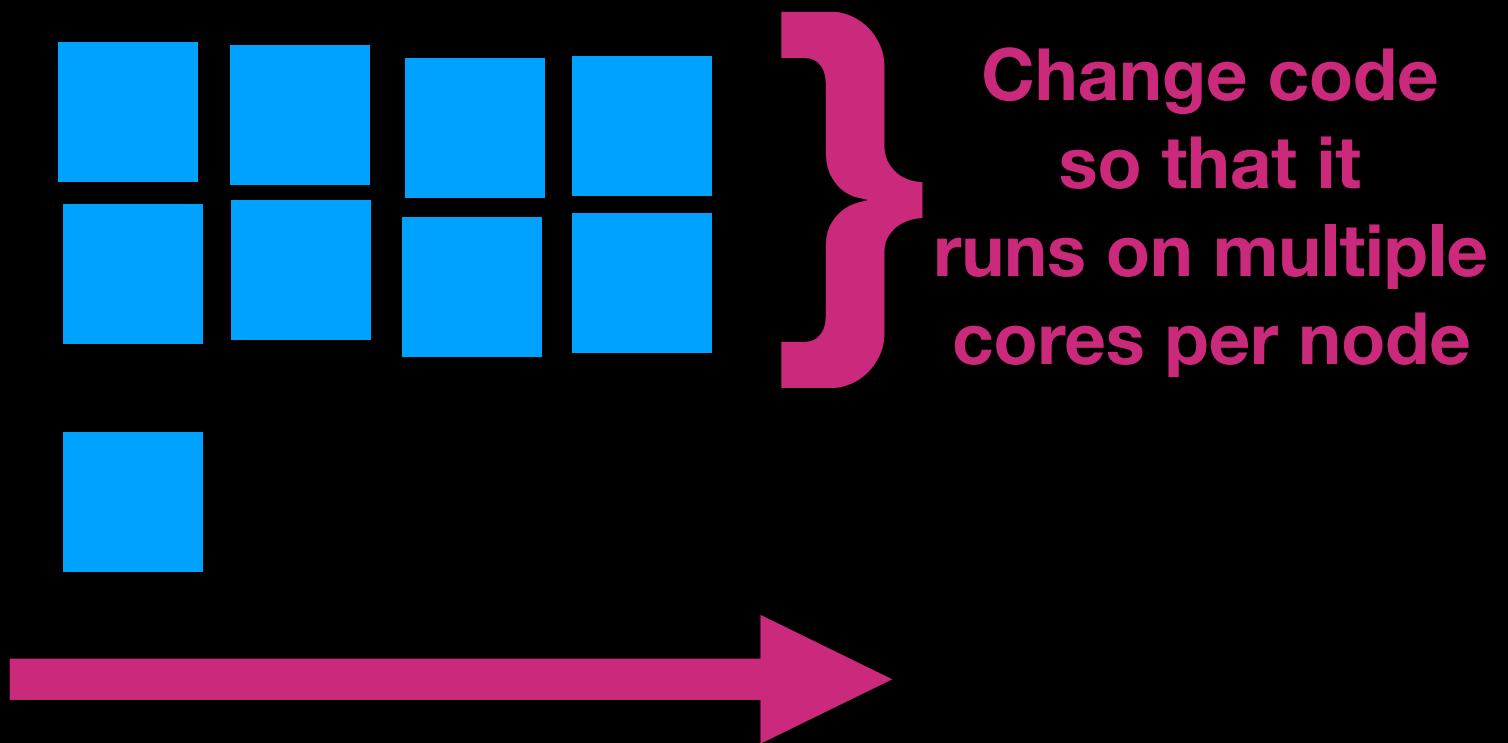
e.g.: iterations of a model

**Ideal computing strategy:**

**Parallelize to different  
nodes, and use one core per node.  
(cluster distributes jobs over single  
cores and nodes)**



# Attention: The prior strategy doesn't work on all clusters.

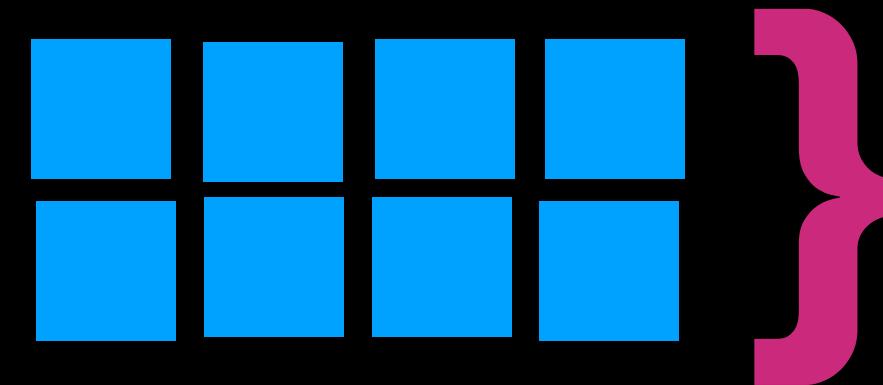


**Chain computations within single jobs**

# Attention: The prior strategy doesn't work on all clusters.

Alternatives seen:

- own clusters
- amazon cloud
- reorganize computations



Change code  
so that it  
runs on multiple  
cores per node

Chain computations within single jobs

**Plan for changing data.**

**Very brief coding practices.**

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

# Organization

- Data
- Code
- Computation

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

Here



Summary

What did we miss?

After lecture (extra): getting  
your environment ready



**One last four minute  
exercise.**

**Discuss in groups of 3.**

**What is applicable to your research?**

**What data science problems of your  
field are not covered?**

**What is common to your data science?**

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

# Organization

- Data
- Code
- Computation

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

**Here**



# Summary

**What did we miss?**

**After lecture (extra): getting  
your environment ready**

# Thank you!

Northwestern | INFORMATION TECHNOLOGY  
RESEARCH COMPUTING SERVICES

**NUPF** — Northwestern University  
Postdoctoral Forum

**SPIE**   
Student Chapter  
Northwestern University

Northwestern | DATA SCIENCE  
Northwestern Institute on Complex Systems

The  
**BDS STUDENT GROUP** 

# Workflow

- Anticipate problems
- Get Overview
- Clean data
- Find connections
- How science differs

# Organization

- Data
- Code
- Computation

# Environment (very short)

- Language
- Computer
- Space
- Constant learning

Here



# Summary

What did we miss?

After lecture (extra): getting  
your environment ready