PL + HCI:

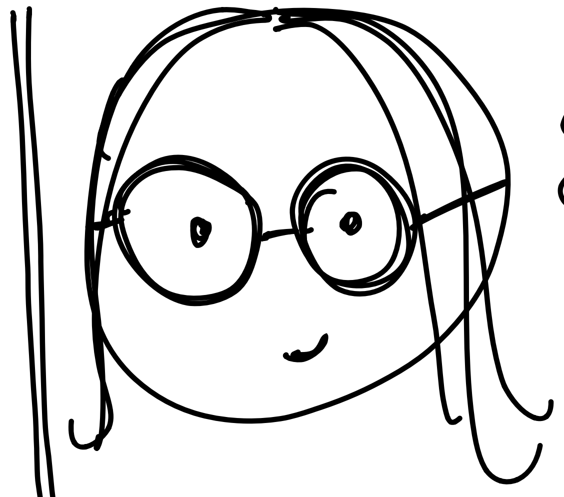
Analysis authoring tools for statistical non-experts

Eunice Jun

February 24, 2022

Hi, I'm **Eunice Jun**

rhymes w/ "sun"



PhD in CS @ U. Washington

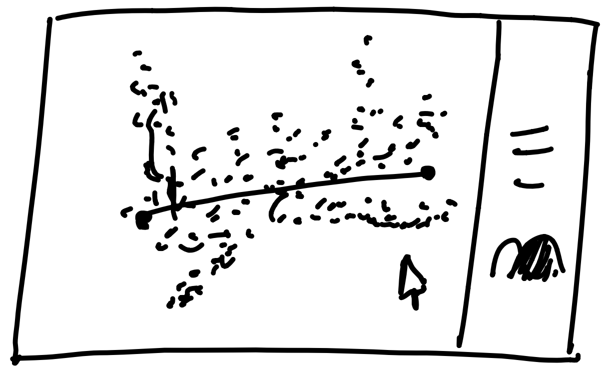
w/ Jeff Heer, René Just

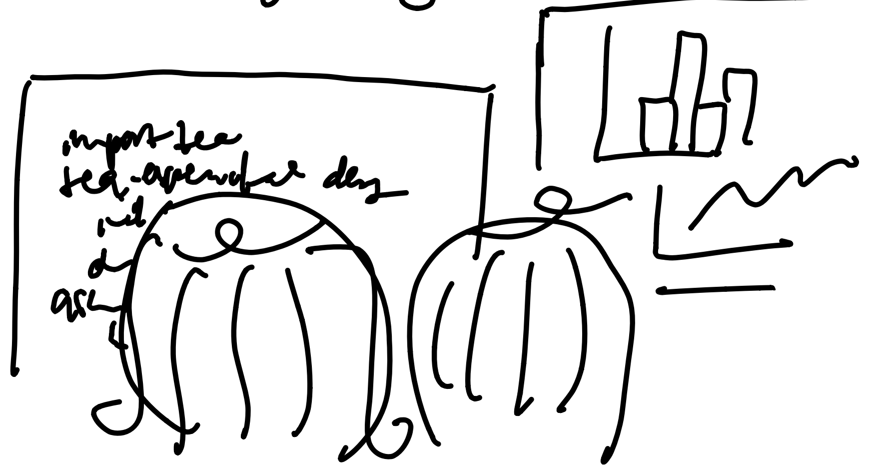
HCI, Viz / PL, SE

I develop new **languages** & **interfaces** for analyzing data.

```
exper_design: {
    indep_var: 'col1',
    dep_var: 'col2'
}
assumptions: {
    'normal': 'col1'
}
```

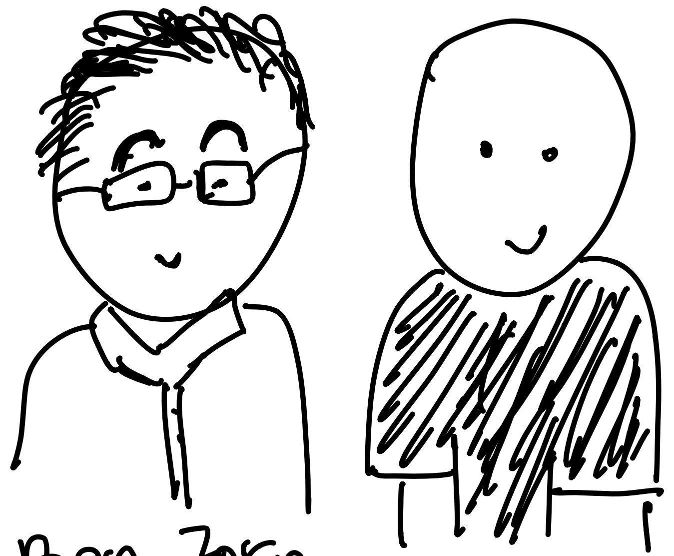
tea-lang.org





@MSR 2019:

RiSE

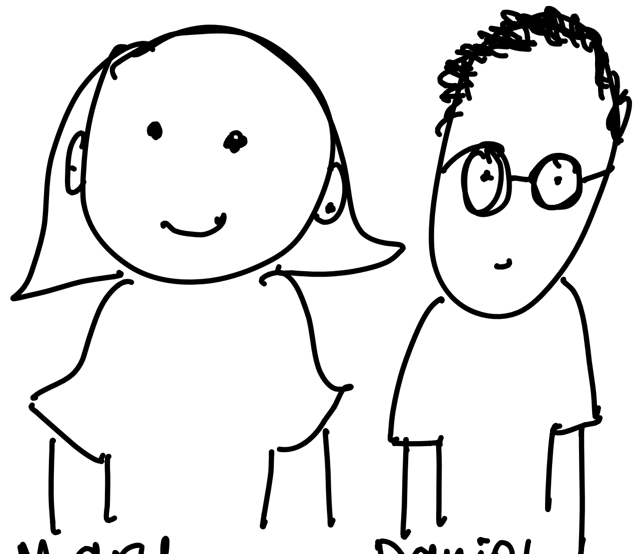


Ben Zorn

Emery Berger

@MSR 2018:

HUE



Mary Czerwinski

Daniel McDuff

I ♥ DATA.

I hope you will, too...*

It's nice to meet you.

*I can help!

Two lenses:

#1.
Programs are UIs.
Programming is HCI.

Software professionals

CSEd teachers

CSEd students

End-users, “non-traditional” coders

Programmers

Two lenses:

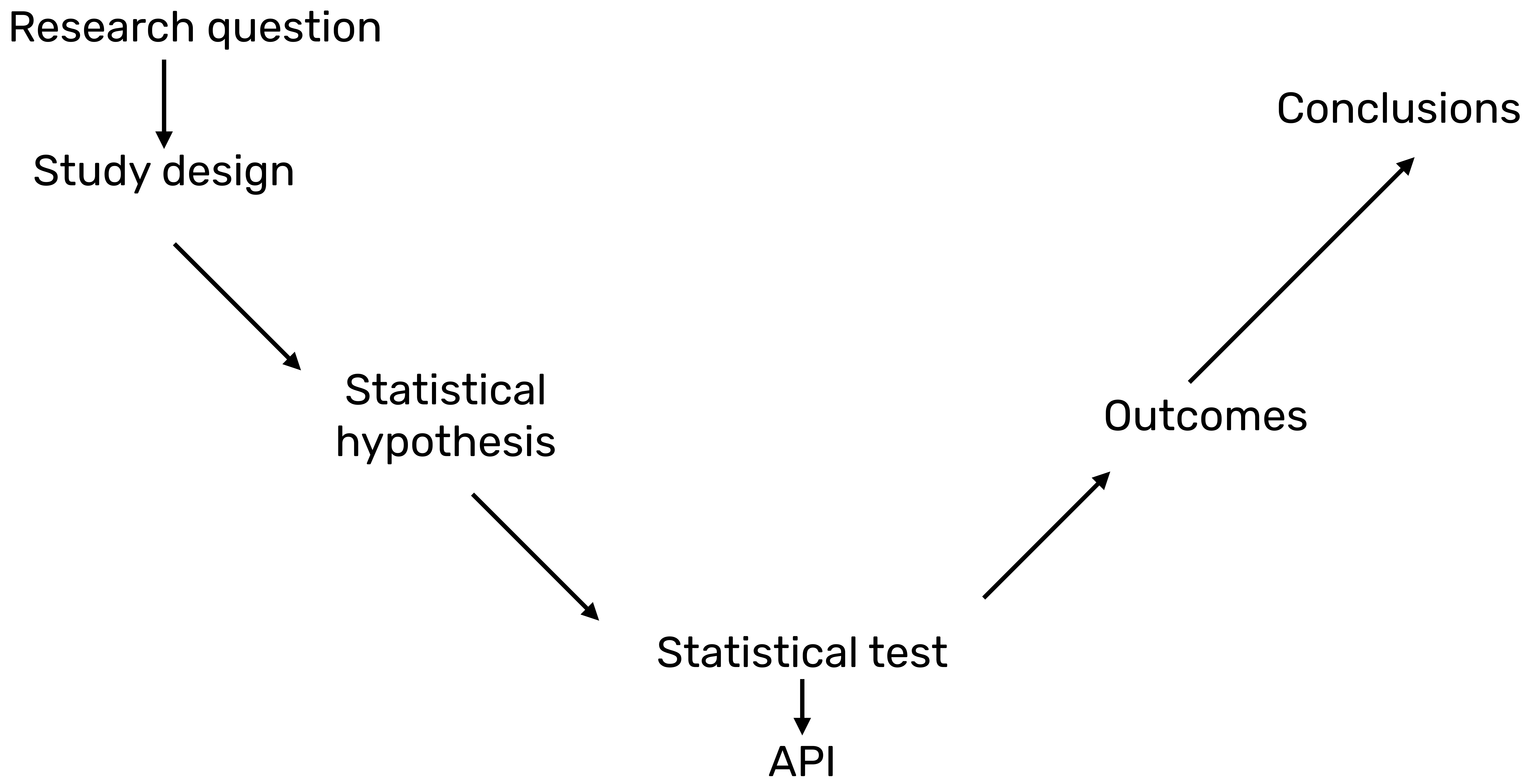
#1.
Programs are UIs.
Programming is HCI.

#2.
PL = Representation
HCI = Interaction

Outline

- **Initial needfinding**
- **Hypothesis formalization** (empirical work + theory building)
- **Tea** (system)
- ***Tisane** (system)
- **Discussion**

Needfinding: Story time!


Research question
Study design
Statistical hypothesis
Statistical test
API
Outcomes
Conclusions


high-level
Research question
Study design
Conclusions
Statistical hypothesis
Outcomes
Statistical test
API
sas
e.g.) t.test(x, y=NULL, alternative = c("two.sided", "less", “greater"), mu = 0, paired = FALSE, var.equal = FALSE, …)
low-level


high-level
Research question
Study design
Statistical hypothesis
Statistical test
API
Outcomes
Conclusions
some support
e.g.) t.test(x, y=NULL, alternative = c("two.sided", "less", “greater"), mu = 0, paired = FALSE, var.equal = FALSE, ...)
low-level


high-level
Research question
Study design
Conclusions
Statistical
hypothesis
Outcomes
up to the user
some support
Incorrect test,
wrong conclusion
Statistical test
API
e.g.) t.test(x, y=NULL, alternative =
c("two.sided", "less", “greater"), mu = 0,
paired = FALSE, var.equal = FALSE, ...)
low-level

Hypothesis Formalization:
Empirical Findings, Software Limitations, and Design Implications

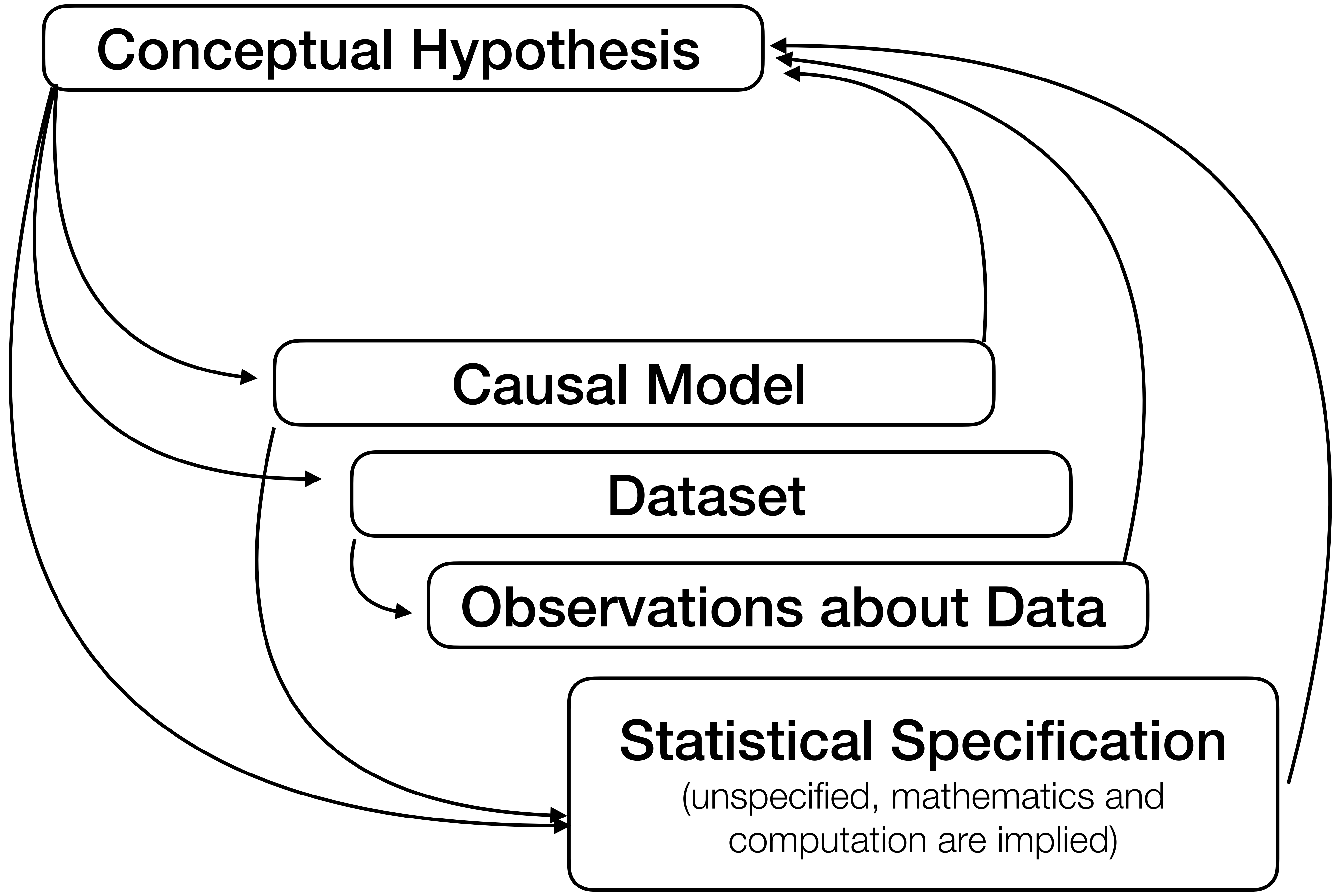
E.Jun, Melissa Birchfield, Nicole de Moura, Jeffrey Heer, René Just. ACM TOCHI 2022. To be presented at CHI 2022.

Research questions

- RQ1: What is the range of **steps** an analyst might consider when formalizing a hypothesis? How do these steps compare to ones that we might expect based on prior work?
- RQ2: How do analysts **think about and perform** the steps?
- RQ3: How might current **software tools** influence hypothesis formalization?

RQ1: Steps to formalize hypotheses

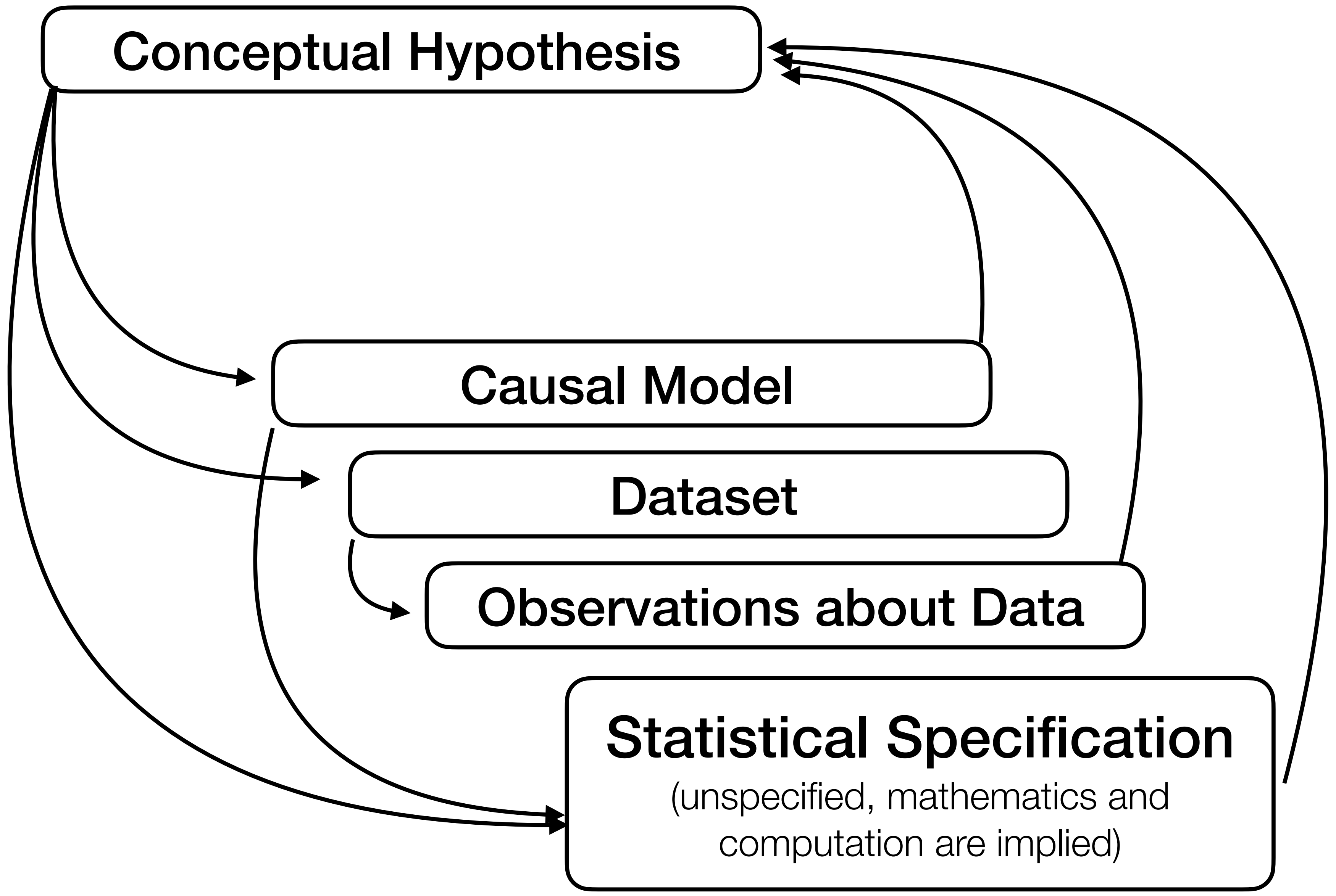
Prior work



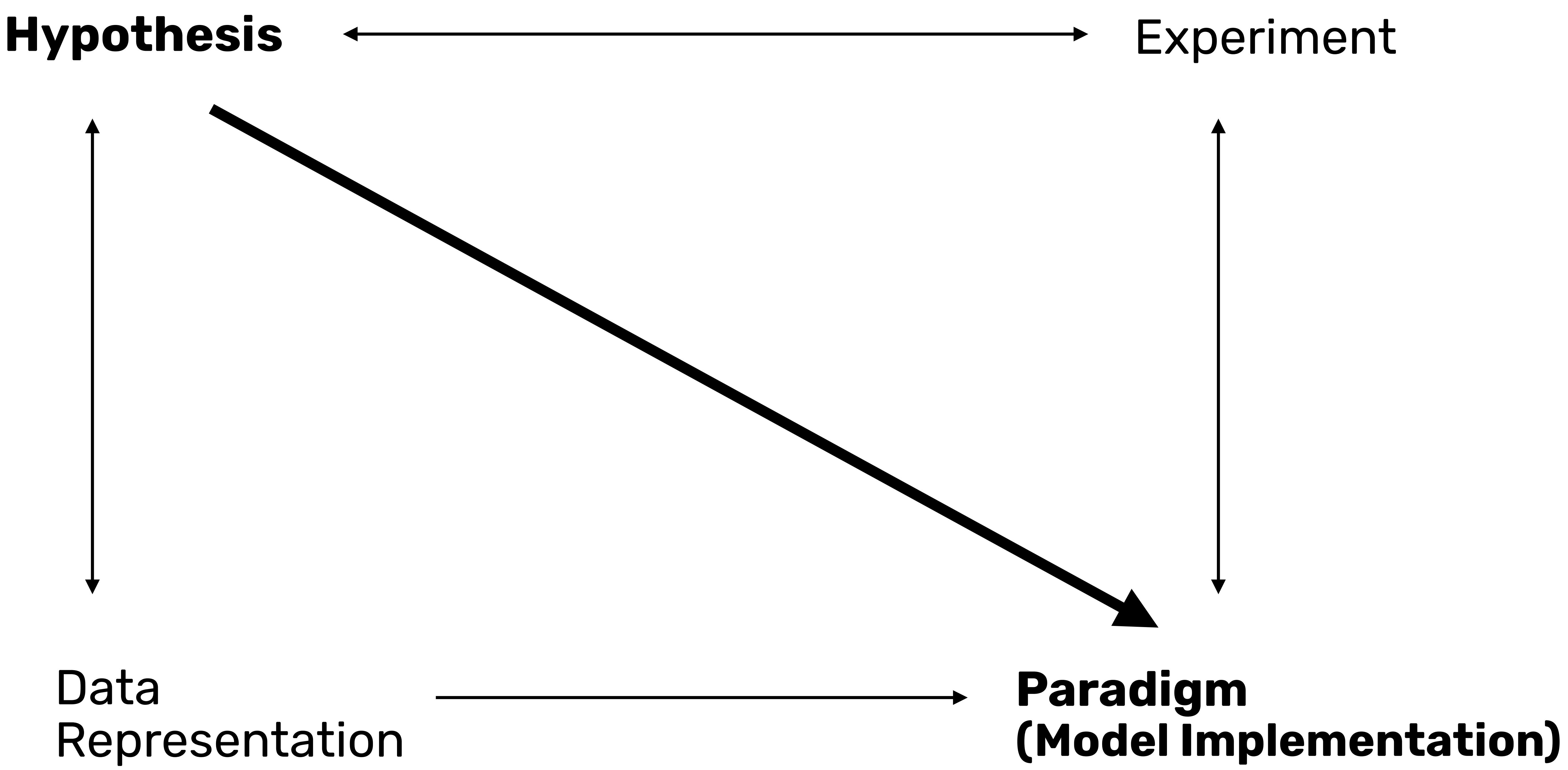
Prior work on data analysis theory + practice

RQ1: Steps to formalize hypotheses

Prior work



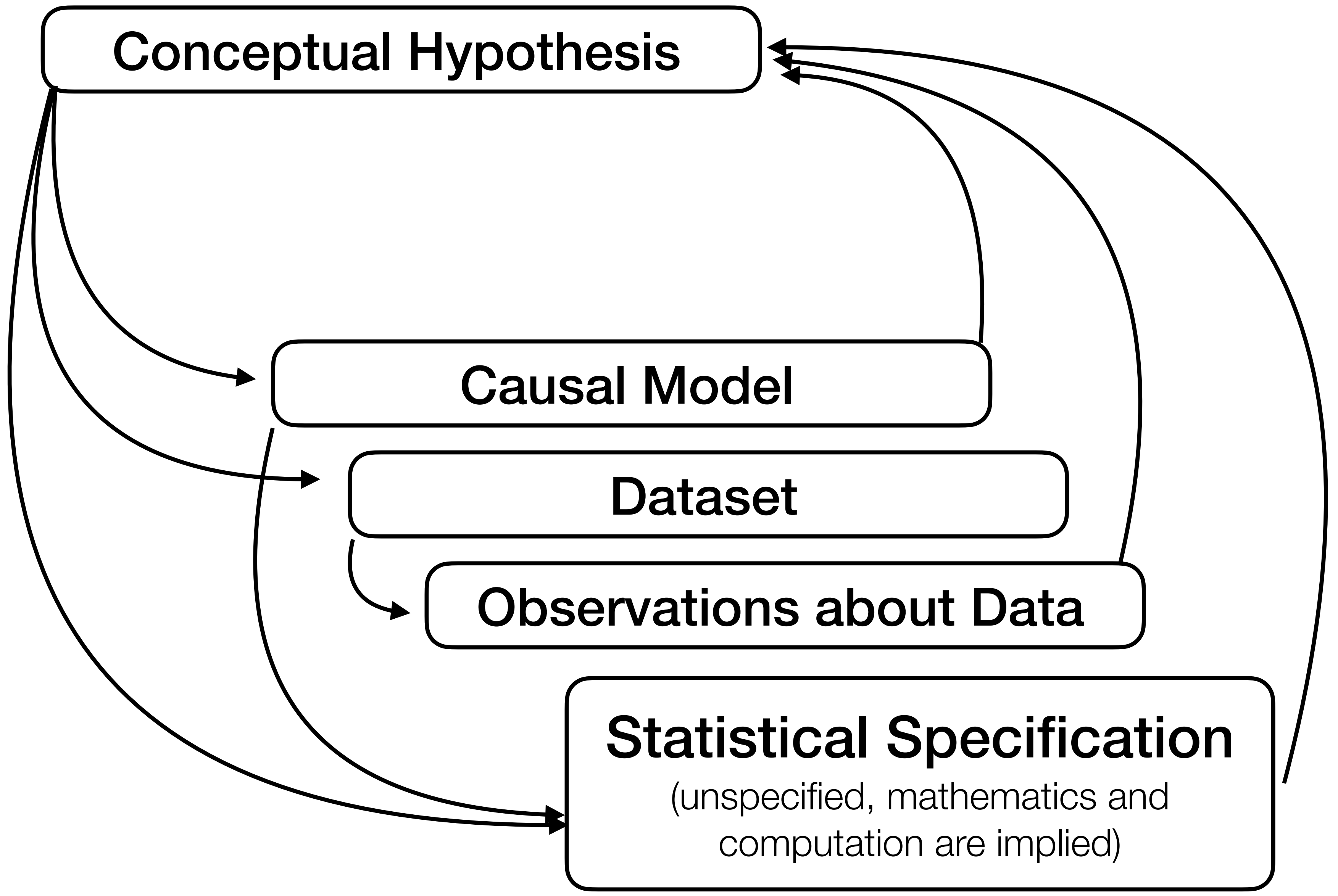
Prior work on data analysis theory + practice



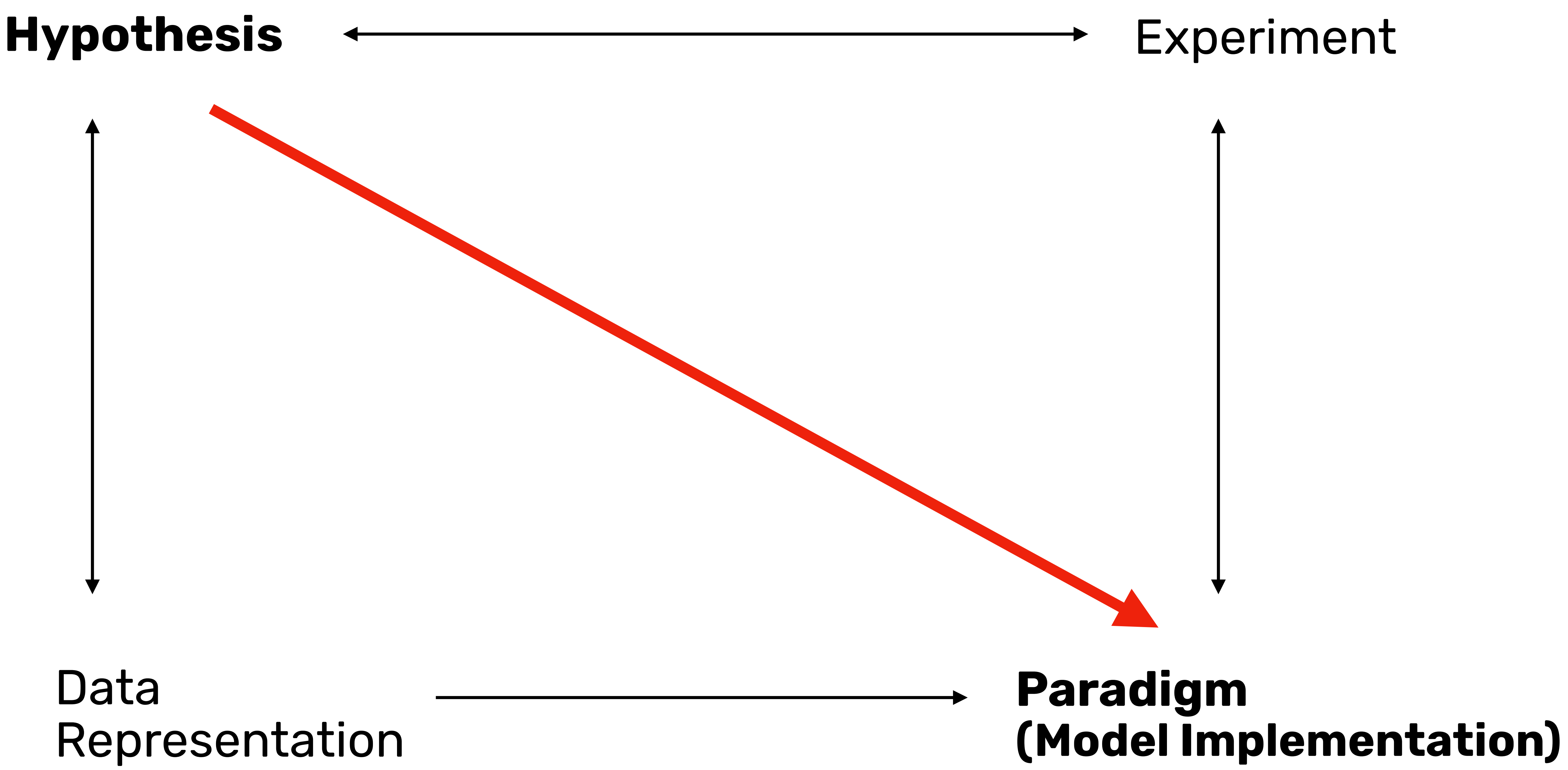
Schunn & Klahr 4-space model of scientific discovery

RQ1: Steps to formalize hypotheses

Prior work



Prior work on data analysis theory + practice



Schunn & Klahr 4-space model of scientific discovery

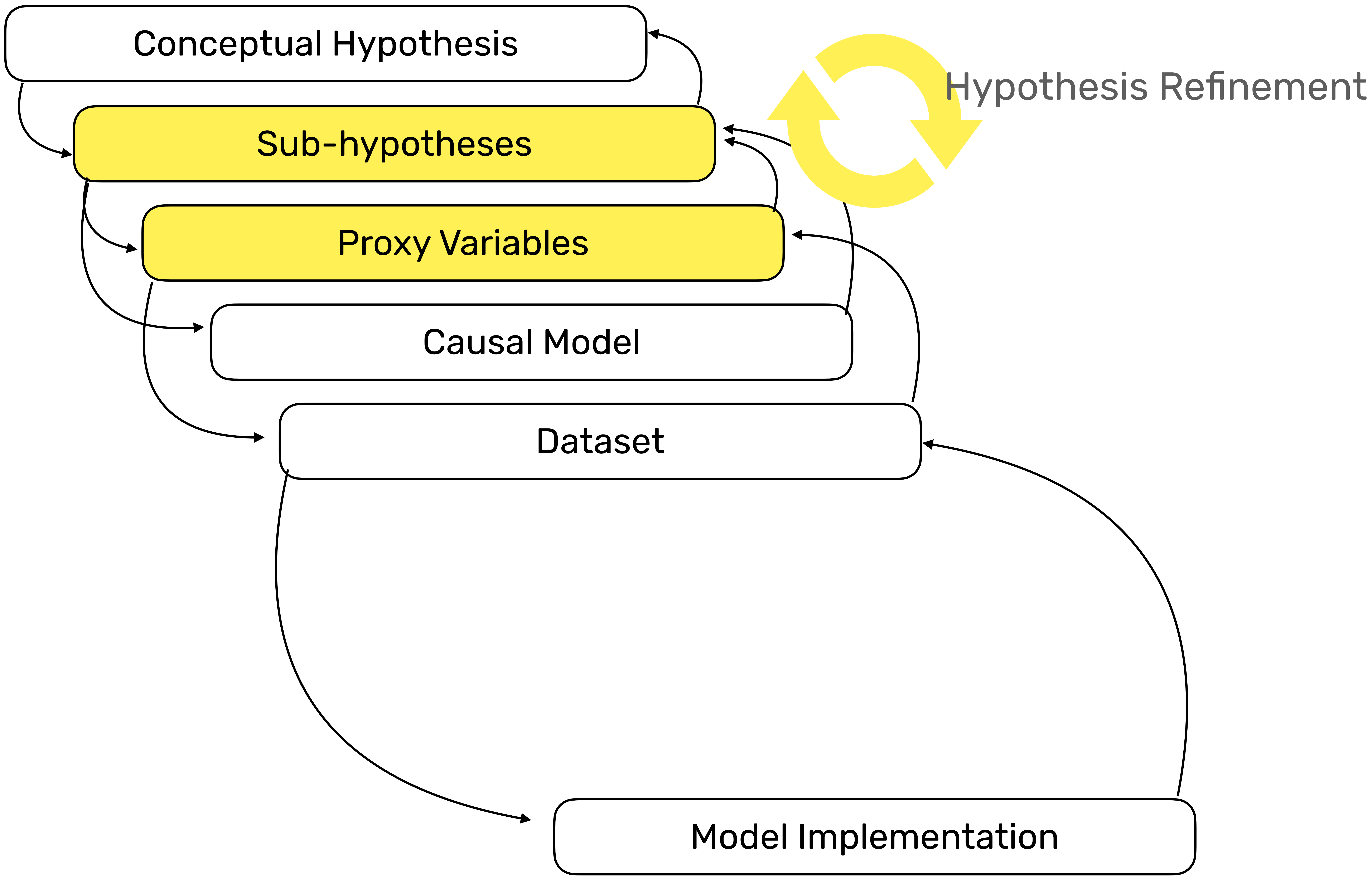
Research questions

- RQ1: What is the range of **steps** an analyst might consider when formalizing a hypothesis? How do these steps compare to ones that we might expect based on prior work?
- RQ2: How do analysts **think about and perform** the steps?
- RQ3: How might current **software tools** influence hypothesis formalization?

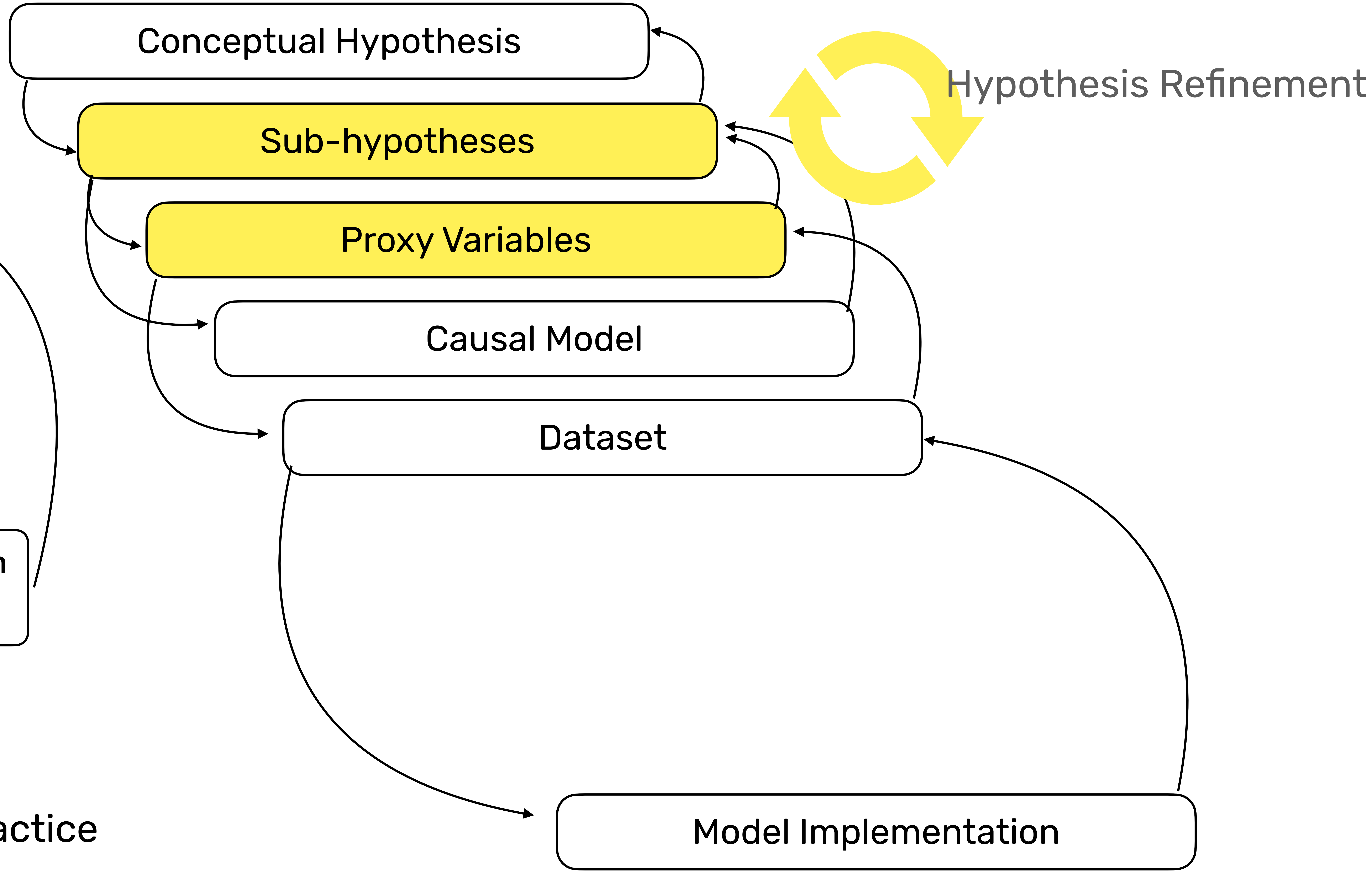
Content Analysis

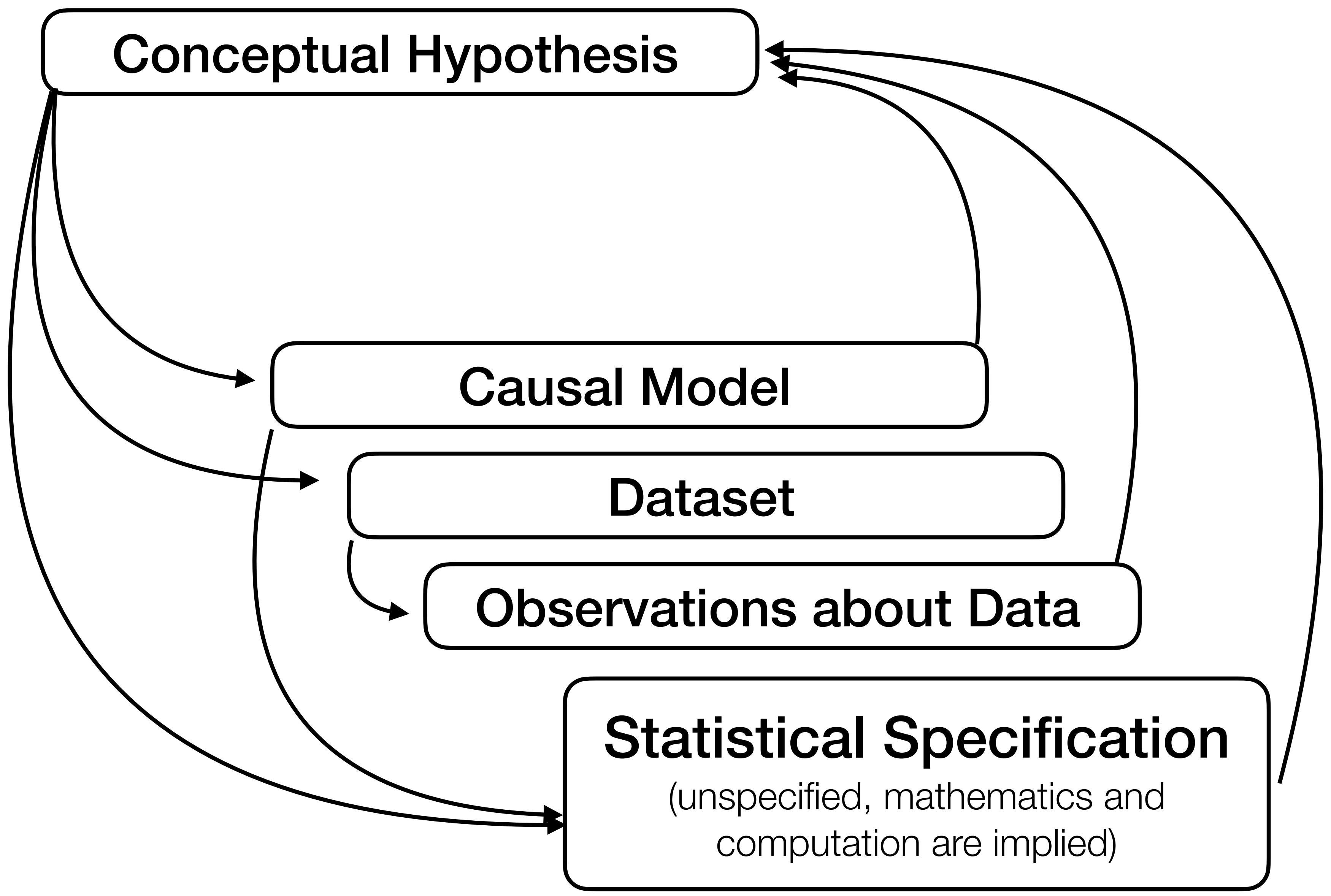
	Paragraph starts with...	[AB	Epis	Sev	Alth	A lir	The	[EX	A tc	[ME	[Prc	Imm	The	[VE	Adu	[ES	The	[RE	As f	[RE	Inte	To a	[EX	Sex	Aga	We	Tog	[GE	The	In a	It is	Las	In c
1	Question or Statement of Unknown	X																		X												X	
2	General Predicted Outcomes					X																											
3	Specific Statistical Expectations																X																
4	Specific objectives																			x													
5	Examining for associations																					x	x										
6	Study Design and Protocol						x			x	x	x	x										x										
7	Initial Data Sourcing	x							x														x										
8	Data Filtering Decisions								x																								
9	Details about data used for analysis																																
10	Proxies						x							x	x																		
11	Equation																																
12	Statistical Specification	x						x								x	x	x		x		x											
13	Statistical results																	x	x	x	x	x		x	x	x	x						
14	Interpreted results	x		x			x											x		x	x	x		x			x	x	x	x	x	x	x
15	Causal model																																
16	Limitations																																
17	Results from other methods																																
18	Other outcomes																															x	
19	Software																x																
20	Computational Details																																

Content Analysis Findings



Content Analysis Findings





Prior work on data analysis theory + practice

Limitation: Scientific narrative bias

Research questions

- RQ1: What is the range of **steps** an analyst might consider when formalizing a hypothesis? How do these steps compare to ones that we might expect based on prior work?
- RQ2: How do analysts **think about and perform** the steps?
- RQ3: How might current **software tools** influence hypothesis formalization?

Lab study

- 24 participants
- 3 part study
  - *“What aspects of an individual’s background and demographics are associated with income after they have graduated from high school?”*
    - Hypotheses
    - Conceptual models
    - Statistical model specification
  - Implement
  - Reflect

Key findings

- Consider proxies and data collection while articulating hypotheses.
- Consider **implementation and tools** when specifying statistical models.

Focus on implementation and tools

Create new variables:

Adj_annual_income - take the midpoint of the ranges in the Annual Income column as a numeric value. (numeric)

State_avg_income - find the average income of individuals in each state from established benchmarks. (numeric)

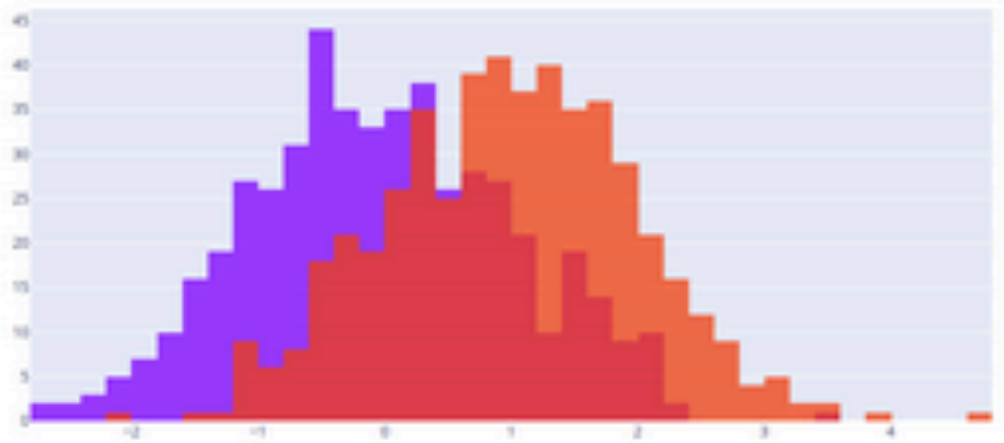
Income_over_avg - take the difference between each individual's income with the average for their state.

Testing Major vs income: take all rows with a college degree (2 year associate and up) & major. Omit rows with no info on income.

For each major, calculate the average *Adj_annual_income*.

Also, calculate the average Adj_annual_income for all the college rows from above.

Create a set of histograms (one for each major) showing the spread of Adj_annual_income for the people in that group. The histograms should share the same x axis. The bins will be normalized to sum to 100% for each major group.



Arrange the data like so

Major	Avg Income (within major)	Avg income (sample population)
Bio	####	####
Stats	####	####
etc.	####	####

Chi-squared test.

H_0: for each major group, the average income is equal to the entire sample population's average income. That is, no single group has a significant difference in avg income from the sample population.

H_A: at least one of the major groups has an average income that's significantly different from the sample population.

Test for a p-value <= 0.05

One caveat of our selected test is even if we are able to reject H_0, we can't make conclusions about which major group is the one making the different. It's possible that just one group is; it's possible that *every* group is significantly different from the population writ large.

Key findings

- Consider proxies and data collection while articulating hypotheses.
- Consider **implementation and tools** when specifying statistical models.
- Fit analyses to previous projects and **familiar approaches.**

Fit to familiar approaches

*“I usually tend to jump…to look at data and **match** [the analysis problem] with **similar patterns** I have seen in the past and start implementing that or do some rough diagrams [for thinking about parameters, data type, and implementation] on paper…and **start implementing** it.”*

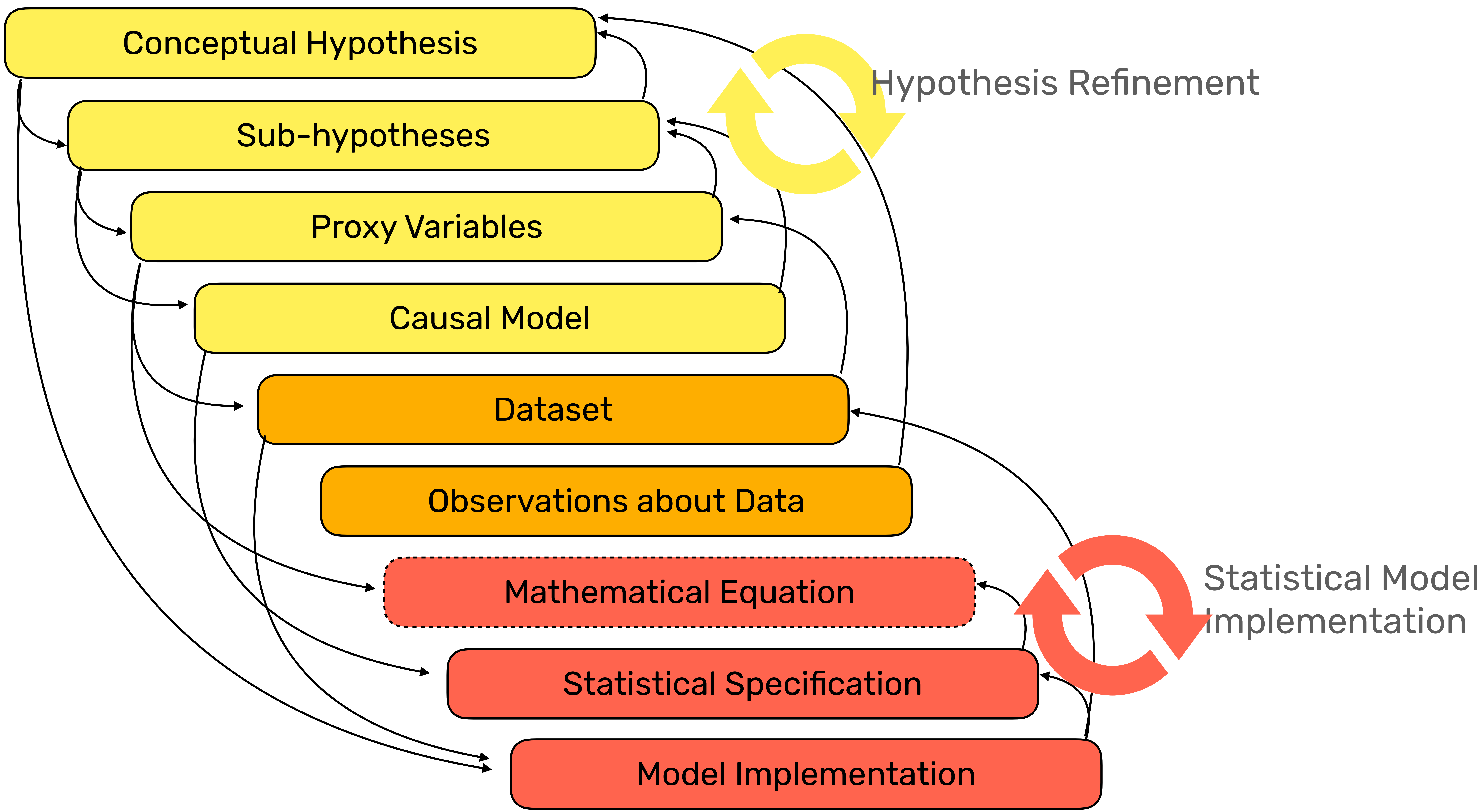
*“I feel like having non normal data is something that’s like hard for us to deal with. Like it just kind of **messes everything up** like….we tend to **try really hard** to get our variables to be normally distributed. So, you know, we might like transform it or, you know, kind of clean it like clean outliers, maybe transform if needed…"*

Key findings

- Consider proxies and data collection while articulating hypotheses.
- Consider **implementation and tools** when specifying statistical models.
- Fit analyses to previous projects and **familiar approaches.**
- Try to minimize their biases by focusing on data.

Key findings

- Consider proxies and data collection while articulating hypotheses.
- Consider **implementation and tools** when specifying statistical models.
- Fit analyses to previous projects and **familiar approaches.**
- Try to minimize their biases by focusing on data.
- Face challenges obtaining and **integrating conceptual and statistical information.**


Conceptual Hypothesis
Sub-hypotheses
Proxy Variables
Causal Model
Dataset
Observations about Data
Mathematical Equation
Statistical Specification
Model Implementation
Hypothesis Refinement
Statistical Model Implementation

Research questions

- RQ1: What is the range of **steps** an analyst might consider when formalizing a hypothesis? How do these steps compare to ones that we might expect based on prior work?

- RQ2: How do analysts **think about and perform** the steps?

- RQ3: How might current **software tools** influence hypothesis formalization?

Tools analysis

- 20 tools
- Focus on
  - Specialization and Scope
  - Model Expression
  - Computationl Control
  - Statistical Taxonomies

ID	Tool name	Specialized Scope	Mathematical Notation	Computational Control
R Packages				
T1	MASS	—	✓	✓
T2	brms	✓	✓	✓
T3	edgeR	✓	✓	✓
T4	glmmTMB	✓	✓	✓
T5	glmnet	✓	—	✓(additional)
T6	lme4	✓	✓	✓
T7	MCMCglmm	✓	✓	✓
T8	nlme	✓	✓	✓
T9	RandomForest	✓	✓	✓(minimal)
T10	stats (core library)	—	✓	✓
Python Packages				
T11	Keras	✓	—	✓(minimal)
T12	Scikit-learn	✓	—	✓
T13	Scipy (scipy.stats)	—	—	✓(additional)
T14	Statsmodels	—	✓	—
Suites, with DSLs for programming				
T15	Matlab (Statistics and ML Toolbox)	—	—	✓
T16	SPSS	—	✓	✓
T17	Stata	—	✓	—
Suites, without programming				
T18	GraphPrism	—	✓*	✓
T19	JASP	—	✓*	—
T20	JMP	—	✓*	—

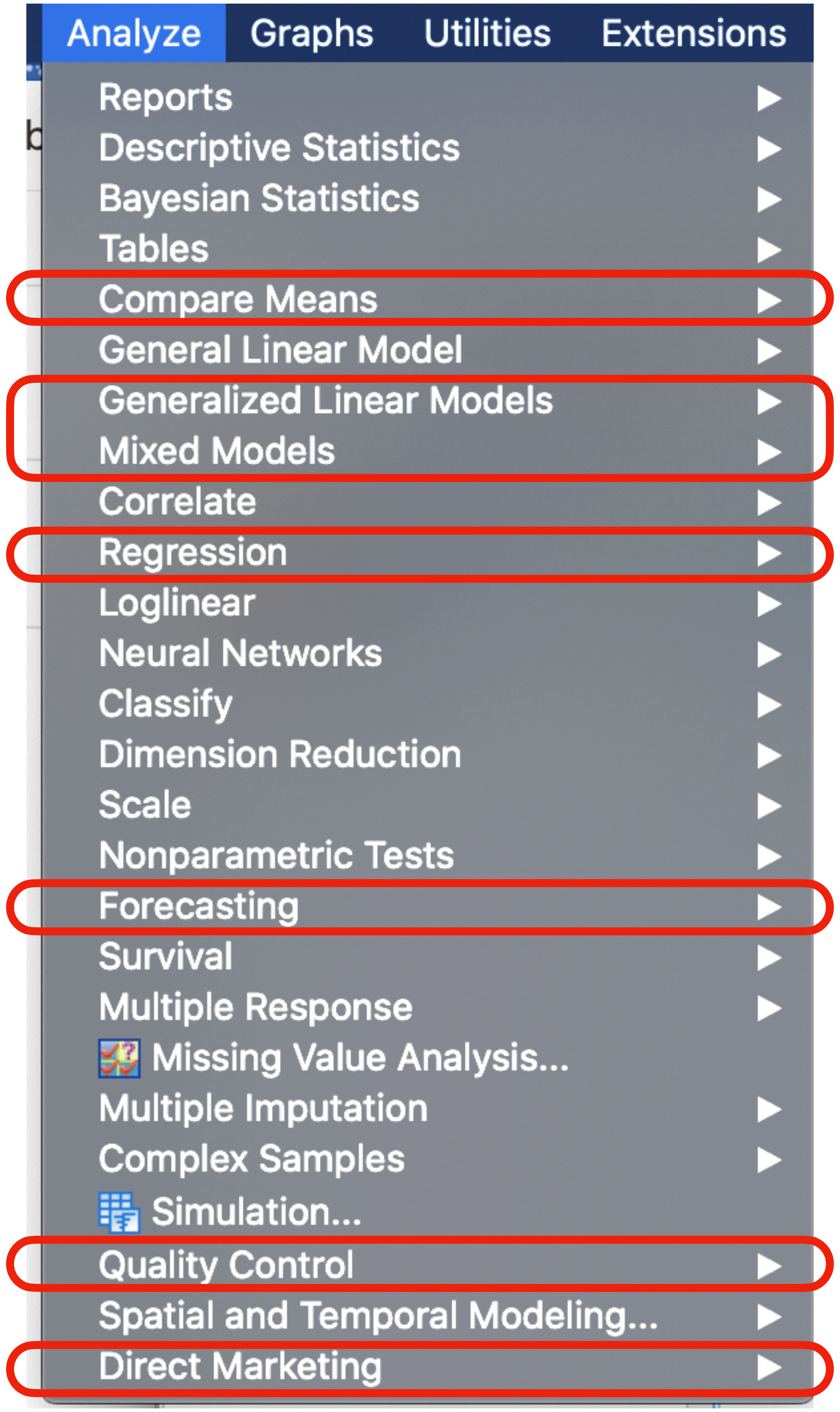
Key findings

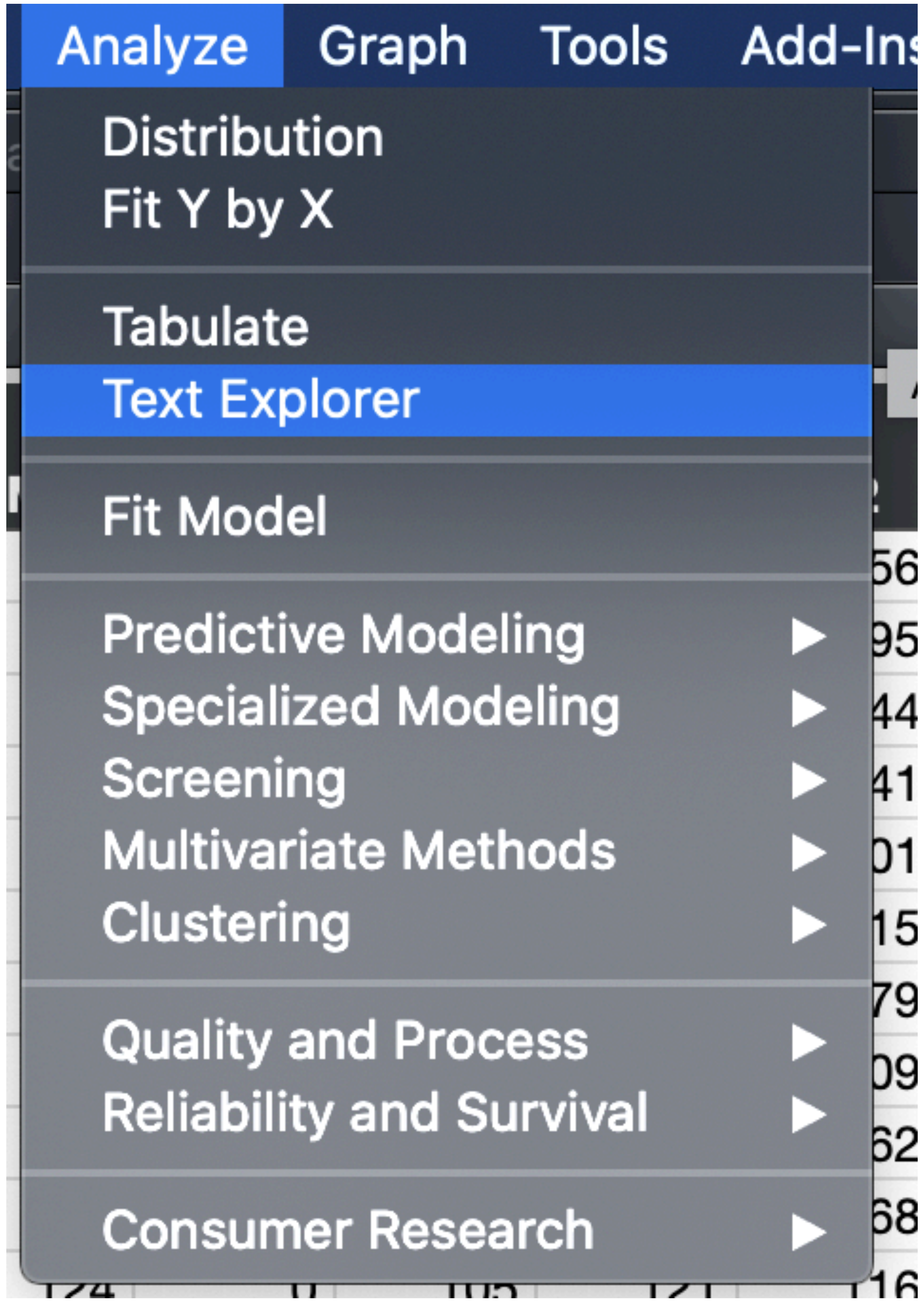
- Specialized tools require analysts to **consider computational settings while picking a statistical tool** and, possibly, even while mathematically relating their variables.

- Tools require analysts to match their conceptual hypotheses with the tools' taxonomies, which may **misalign with their personal taxonomies.**

Misaligned taxonomies

SPSS





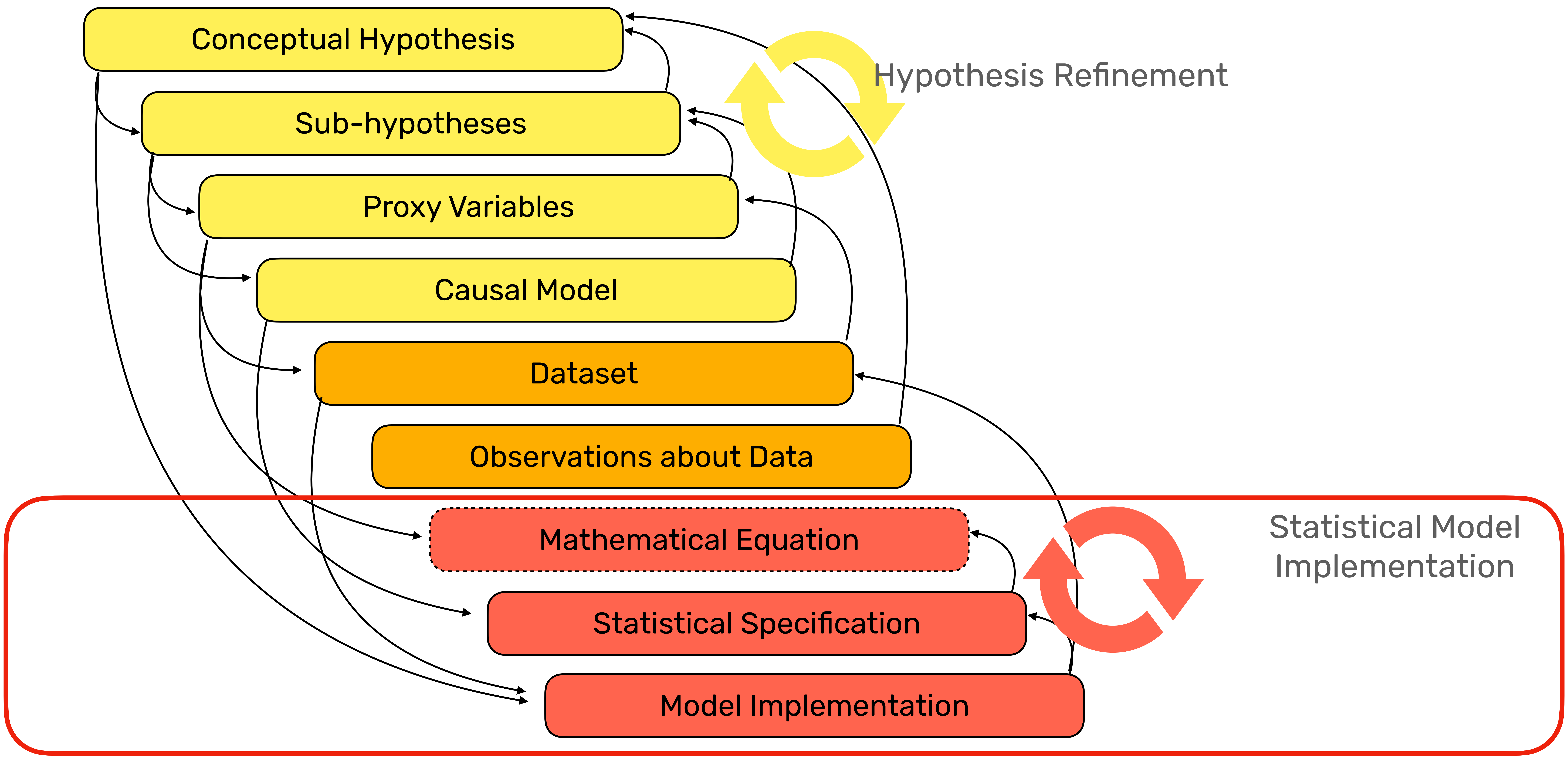
JMP

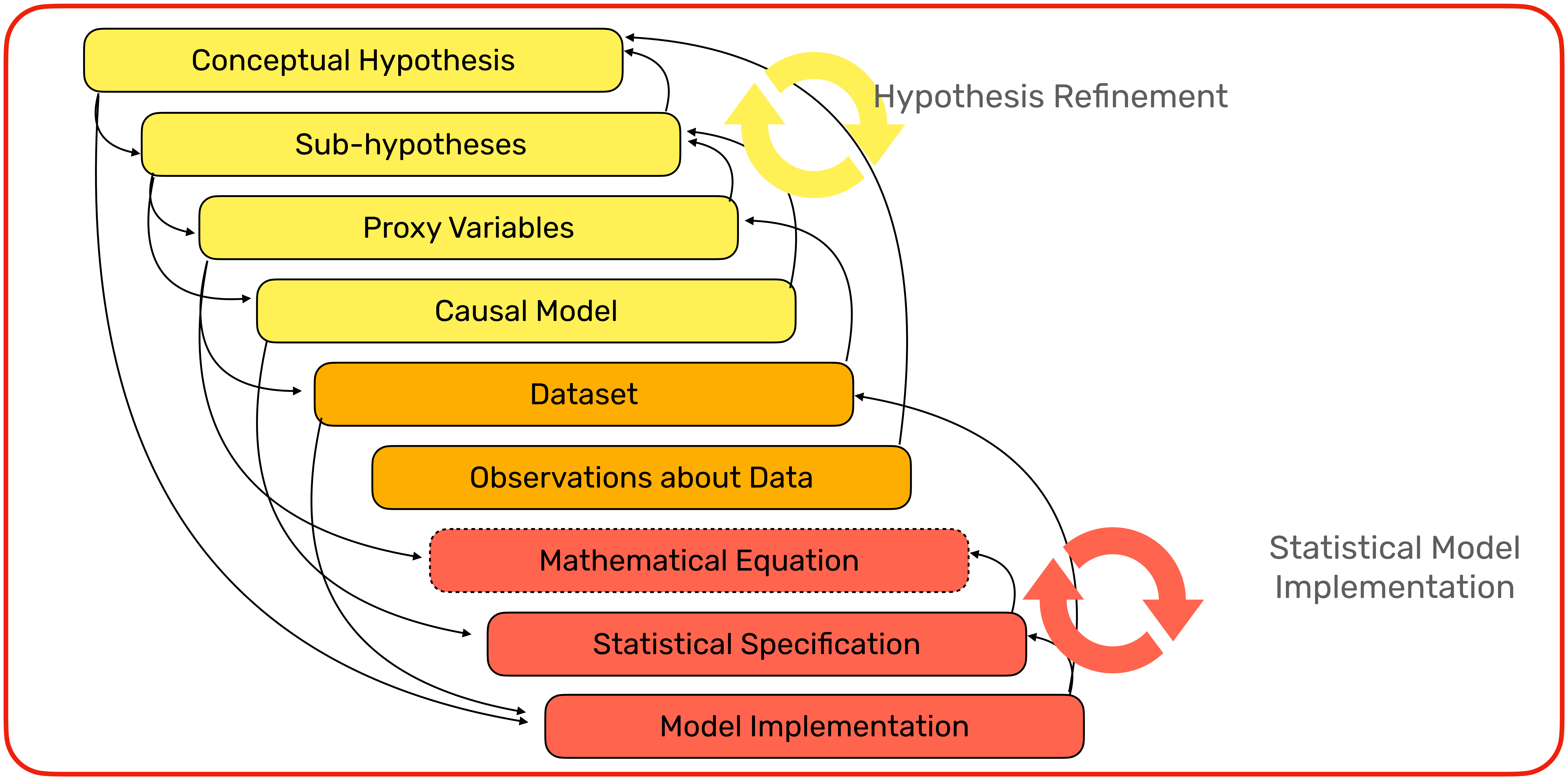
Key findings

- Specialized tools require analysts to **consider computational settings while picking a statistical tool** and, possibly, even while mathematically relating their variables.
- Tools require analysts to match their conceptual hypotheses with the tools' taxonomies, which may **misalign with their personal taxonomies.**
- **Syntactic and semantic mismatches** can create a rift between model implementations and conceptual hypotheses.
- Low-level control could help but introduce a **gulf of evaluation.**

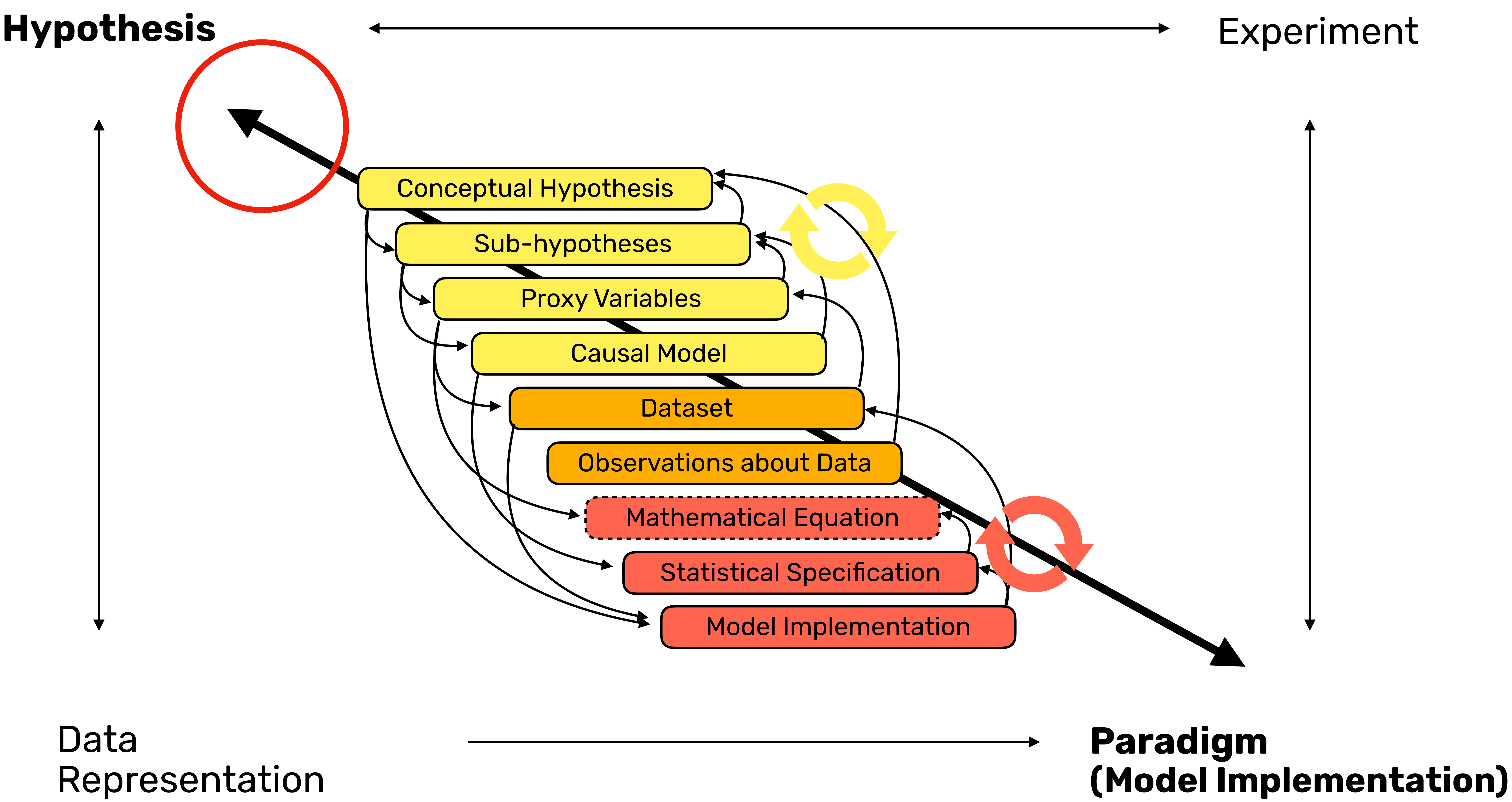
Implications

- High-level abstractions
- Co-authoring conceptual and statistical models


Conceptual Hypothesis
Sub-hypotheses
Proxy Variables
Causal Model
Dataset
Observations about Data
Mathematical Equation
Statistical Specification
Model Implementation
Hypothesis Refinement
Statistical Model
Implementation


Conceptual Hypothesis
Sub-hypotheses
Proxy Variables
Causal Model
Dataset
Observations about Data
Mathematical Equation
Statistical Specification
Model Implementation
Hypothesis Refinement
Statistical Model
Implementation

Theoretical Implications



Schunn & Klahr 4-space model of scientific discovery

high-level

Research question

↓

Study design

↘

Statistical hypothesis

↘

Statistical test

↓

API

e.g.) t.test(x, y=NULL, alternative = c("two.sided", "less", “greater"), mu = 0, paired = FALSE, var.equal = FALSE, …)

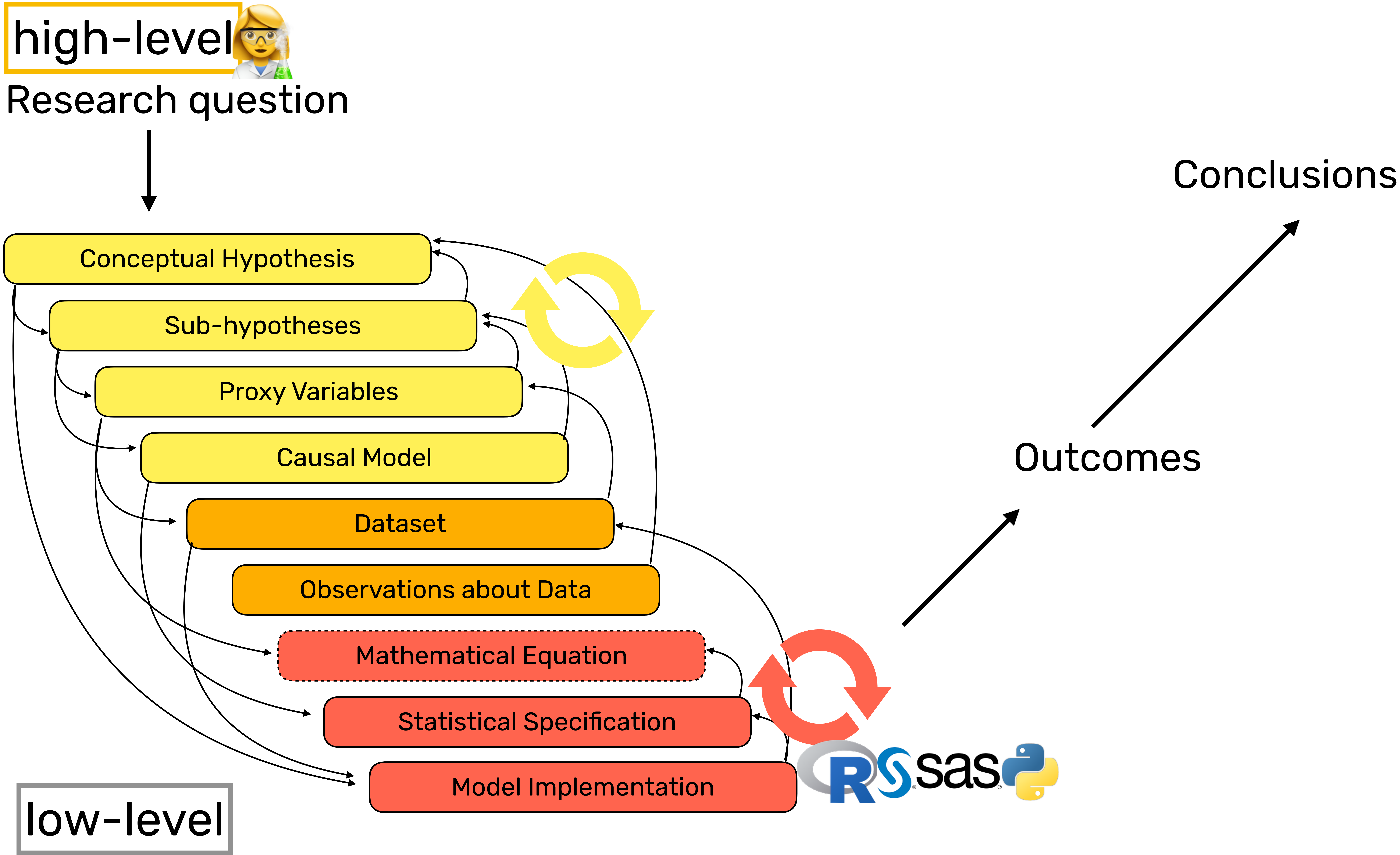
↗

Outcomes

↗

Conclusions

low-level


high-level
Research question
Conceptual Hypothesis
Sub-hypotheses
Proxy Variables
Causal Model
Dataset
Observations about Data
Mathematical Equation
Statistical Specification
Model Implementation
low-level
Outcomes
Conclusions
R
sas

Tea:
A High-level Language and Runtime System for Statistical Analysis

E. Jun, Maureen Daum, Jared Roesch, Sarah Chasins, Emery Berger, René Just, Katharina Reinecke. ACM UIST 2019.

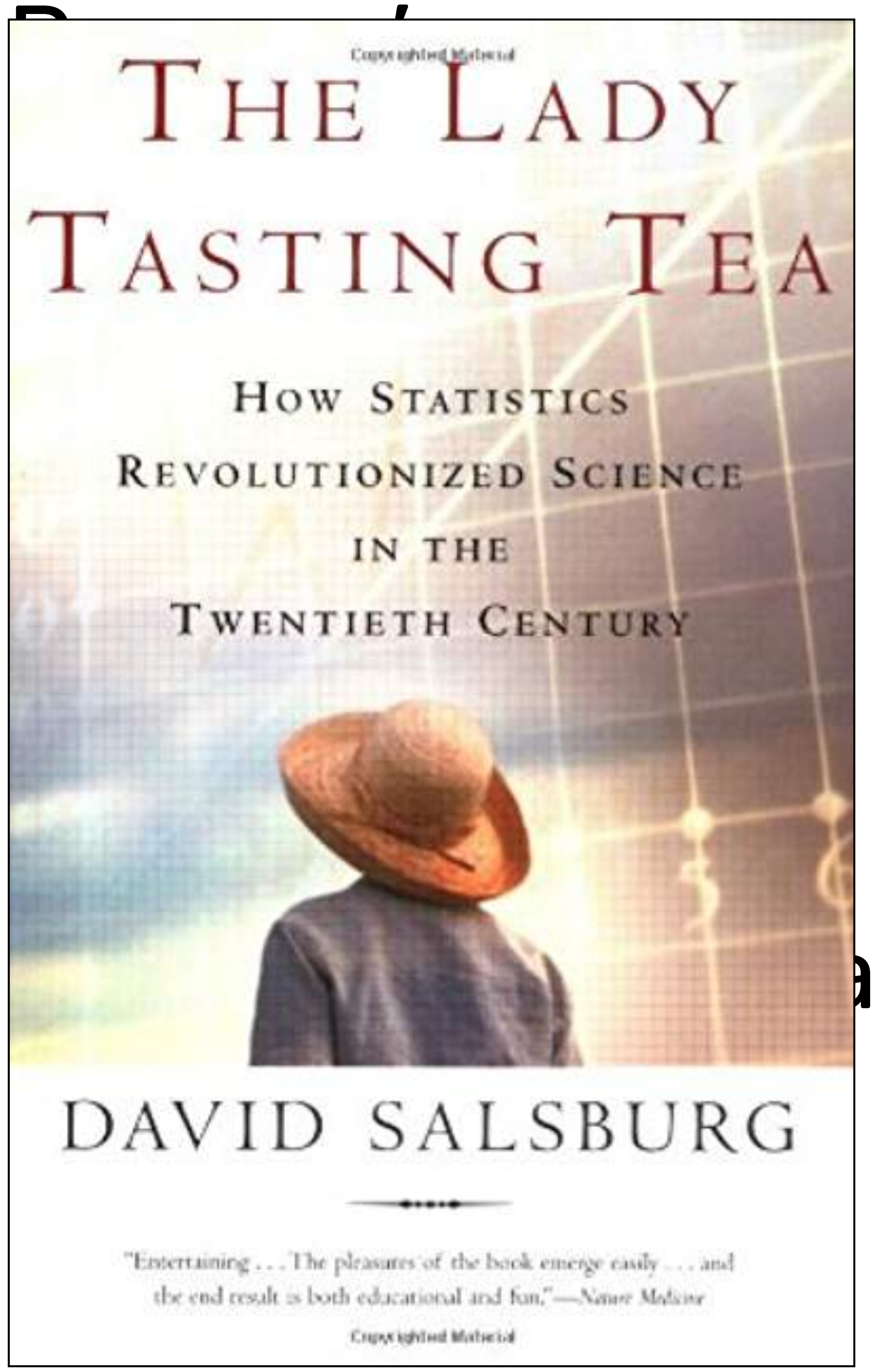
Does caffeine consumption affect question asking?

Group A

Group B

Stats needed!

Does tea taste different with milk added before vs. after tea?



ank

Welch's
F-test
Repeated measures
one-way ANOVA
Factorial ANOVA
Two-way ANOVA
Kruskal Wallis
Friedman

Fisher's Exact
Linear regression
Logistic regression
MANOVA
ANCOVA
MANCOVA
McNemar
Chi Square

Which statistical test?

Fisher's Exact Test!

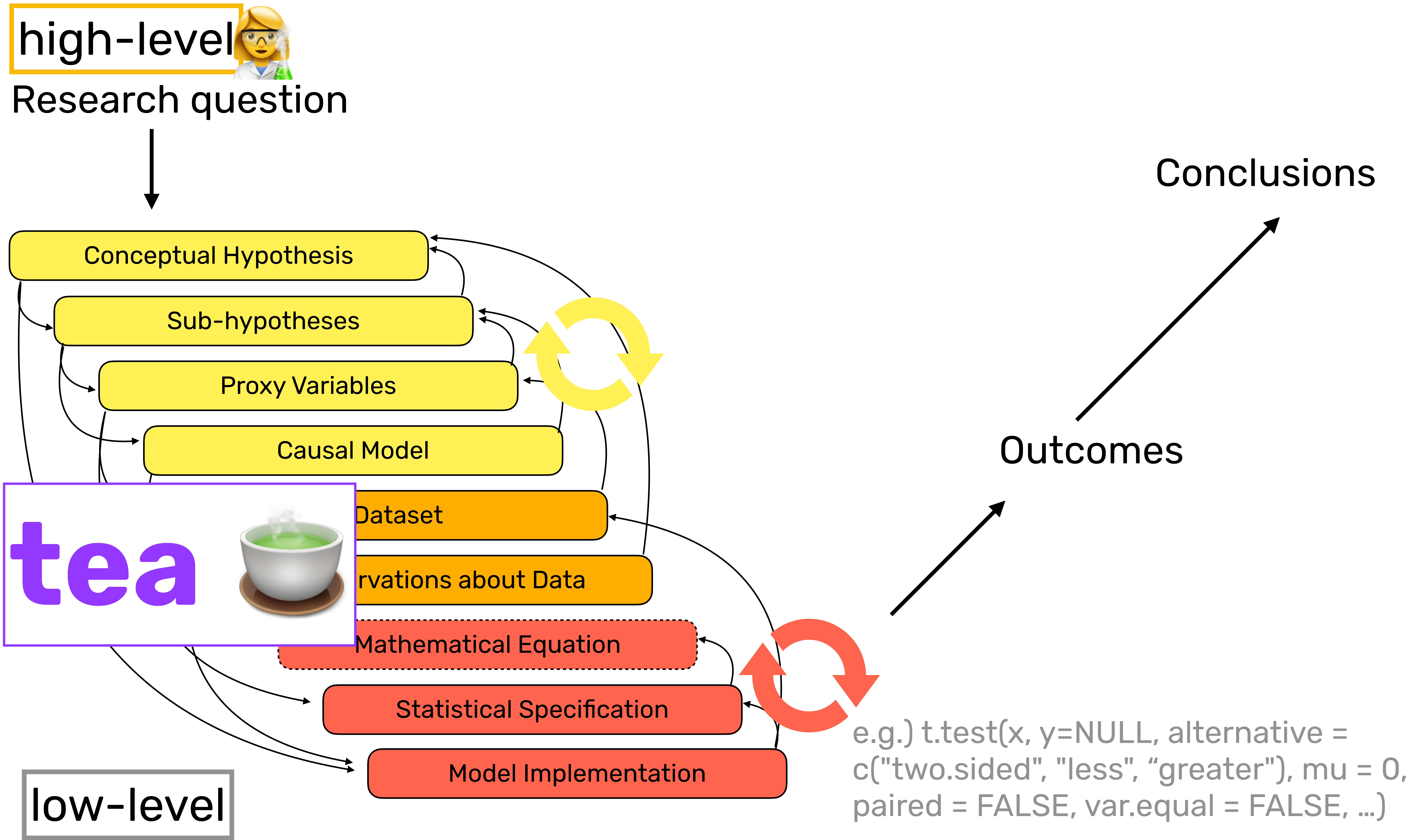
tea

EASY

- Does caffeine consumption affect question asking?
- Does tea taste different with milk added before vs. after tea?

HARD

Pearson's r	Welch's	Fisher's Exact
Pointbiserial	F-test	Linear regression
Kendall's T	Repeated measures	Logistic regression
Spearman's p	one-way ANOVA	MANOVA
Student's t-test	Factorial ANOVA	ANCOVA
Paired t-test	Two-way ANOVA	MANCOVA
Mann-Whitney U	Kruskal Wallis	McNemar
Wilcoxon signed rank	Friedman	Chi Square


high-level
Research question
Conceptual Hypothesis
Sub-hypotheses
Proxy Variables
Causal Model
Dataset
rvations about Data
Mathematical Equation
Statistical Specification
Model Implementation
tea
low-level
Outcomes
Conclusions
e.g.) t.test(x, y=NULL, alternative =
c("two.sided", "less", “greater"), mu = 0,
paired = FALSE, var.equal = FALSE, …)


high-level
Research question
tea
abstracts away
Conclusions
Conceptual Hypothesis
Sub-hypotheses
Proxy Variables
Causal Model
Dataset
Observations about Data
Mathematical Equation
Statistical Specification
Model Implementation
Outcomes
e.g.) t.test(x, y=NULL, alternative = c("two.sided", "less", “greater"), mu = 0, paired = FALSE, var.equal = FALSE, …)
low-level

Overview of Tea

tea

What:

Tea is high-level.

Tea infers statistical tests.

Tea provides precise output.

Tea improves upon expert choices, prevents common mistakes.

Who:

Domain experts (not in stats!)

Comfortable with study design

Minimal programming

Tea helps domain experts conduct valid, replicable statistical analyses.

Replicable: Different team, same experimental setup; Same results

Tea:

How to use it

How it works

How it performs

Tea:

How to use it

How it works

How it performs

tea

pip install tealang
import tea

jupyter

data

variables

study design

assumptions

hypothesis

```
Test: students_t
***Test assumptions:
Exactly two variables involved in analysis: So Prob
Exac
Exac
Inde
Vari
Vari
Cont
Equal variance: So Prob
Groups are normally distributed: So Prob:
NormalTest(W=0.8997463583946228 p_value=0.07962072640657425)

***Test results:
name = Student's T Test
test_statistic = 4.20213
adjusted p value = 0.00006
alph
dof
Effe
Cohe
A12
Null                                                    tween So =
0 an
Inte                                                     the null
hypothesis at alpha = 0.05. The mean of Prob for So = 1
(M=0.06371 SD=0.02251) is significantly greater than the mean
for So = 0 (M=0.03851 SD=0.01778). The effect size is Cohen's
d = 1.24262 A12 = 0.83669. The effect size is the magnitude of
the difference which gives a holistic view of the results [1].
[1] Sullivan G. M. & Feinn R. (2012). Using effect size—or why
the P value is not enough. Journal of graduate medical
education 4(3) 279-282.
```

Explain rationale for test selection.

Contextualize results for accurate interpretation.

tea

pip install tealang
import tea

jupyter

Pearson's r
Pointbiserial,
Kendall's T,
Spearman's p,
Student's t-test,
Paired t-test,
Mann-Whitney U,
Wilcoxon signed rank,
Welch's,
F-test,
Repeated measures one-way ANOVA,
Factorial ANOVA,
Two-way ANOVA,
Kruskal Wallis,
Friedman,
Chi Square,
Fisher's Exact,
Bootstrapping

data

variables

study design

assumptions

hypothesis

```
Test: students_t
***Test assumptions:
Exactly two variables involved in analysis: So Prob
Exac
Exac
Inde
Vari
Vari
Cont
Equal variance: So Prob
Groups are normally distributed: So Prob:
NormalTest(W=0.8997463583946228 p_value=0.07962072640657425)

***Test results:
name = Student's T Test
test_statistic = 4.20213
adjusted p value = 0.00006
alph
dof
Effe
Cohe
A12
Null                                                    tween So =
0 an
Inte                                                    the null
hypothesis at alpha = 0.05. The mean of Prob for So = 1
(M=0.06371 SD=0.02251) is significantly greater than the mean
for So = 0 (M=0.03851 SD=0.01778). The effect size is Cohen's
d = 1.24262 A12 = 0.83669. The effect size is the magnitude of
the difference which gives a holistic view of the results [1].
[1] Sullivan G. M. & Feinn R. (2012). Using effect size—or why
the P value is not enough. Journal of graduate medical
education 4(3) 279-282.
```

Explain rationale for test selection.

Contextualize results for accurate interpretation.

tea

```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
    }
]
tea.defi
                   observational study',
        contributor variables': 'Southern',
        'outcome variables': 'Probability',
       }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                  [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

data

variables

study design

assumptions

hypothesis

* * NO STATISTICAL TEST * *

tea

data

```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
    }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
        }
tea.define_study_design(study_design)
assumptions = {
```

tea

variables

```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
    }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
        }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
```

tea

options:
Nominal
Ordinal
Interval
Ratio

```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
    }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
        }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
```

variables

tea

study design

```
            'data type' : 'ratio',
        }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
          }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                    [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

```
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
           }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                     [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

assumptions

tea

hypothesis

```
                    [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
   }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
           }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                    [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

data

variables

study design

assumptions

hypothesis

Tea:

How to use it

How it works

How it performs

```python
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
   }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
        }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

✓completeness
✓syntax
✓well-formed hypotheses

Nominal,Ordinal:

Northern > Western

Low SES < High SES

Ordinal,Ratio,Interval:

SES ~ Income

Age ~ - Income

Test selection as constraint satisfaction!

What are constraints?

???

Pearson's r
Pointbiserial,
Kendall's T,
Spearman's p,
Student's t-test,
Paired t-test,
Mann-Whitney U,
Wilcoxon signed rank,
Welch's,
F-test,
Repeated measures one-way ANOVA,
Factorial ANOVA,
Two-way ANOVA,
Kruskal Wallis,
Friedman,
Chi Square,
Fisher's Exact,
Bootstrapping

```
Test: students_t
***Test assumptions:
Exactly two variables involved in analysis: So, Prob
Exactly one explanatory variable: So
Exactly one explained variable: Prob
Independent (not paired) observations: So
Variable is categorical: So
Variable has two categories: So
Continuous (not categorical) data: Prob
Equal variance: So, Prob
Groups are normally distributed: So, Prob

***Test results:
name = Student's T Test
test_statistic = 4.202130736875173
p_value = 0.00012364897266532775
adjusted_p_value = 6.182448633266387e-05
alpha = 0.05
dof = 45
Effect size:
Cohen's d = 1.2426167296374897
A12 = 0.8366935483870968
Null hypothesis = There is no difference in means between 0 and 1 on Prob.
Interpretation = t(45) = 4.202130736875173, 6.182448633266387e-05. Reject the null
hypothesis at alpha = 0.05. The mean of Prob for So = 1 is significantly greater than
the mean for So = 0. The effect size is {"Cohen's d": 1.2426167296374897, 'A12':
0.8366935483870968}. The effect size is the magnitude of the difference, which gives
a holistic view of the results [1].
[1] Sullivan, G. M., & Feinn, R. (2012). Using effect size—or why the P value is not
enough. Journal of Graduate Medical Education, 4(3), 279-282.
```

Statistical test selection as constraint satisfaction

tea

```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
    }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
           }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
               [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

constraints

.
.
.
.
.
.
.
.
.
.
.
.
.
.
.

Pearson's r
Pointbiserial,
Kendall's T,
Spearman's p,
Student's t-test,
Paired t-test,
Mann-Whitney U,
Wilcoxon signed rank,
Welch's,
F-test,
Repeated measures
one-way ANOVA,
Factorial ANOVA,
Two-way ANOVA,
Kruskal Wallis,
Friedman,
Chi Square,
Fisher's Exact,
Bootstrapping

Statistical test selection as constraint satisfaction

tea

```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
   }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
          }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                  [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

Student's t-test ✅ ❌

Exactly 2 groups ✅ ✅

✅ ✅

✅ ❌

✅ ✅

✅ ✅

Groups are normally distributed ✅ ✅

✅ ✅

✅ ✅

Statistical test selection as constraint satisfaction

tea

```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
   }
]
tea.define_variables(variables)
study_design = {
             'study type': 'observational study',
             'contributor variables': 'Southern',
             'outcome variables': 'Probability',
            }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                  [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

Student's t-test ✅ Test = constraints

Exactly 2 groups ✅ $\wedge$

⋮ ✅

✅

✅

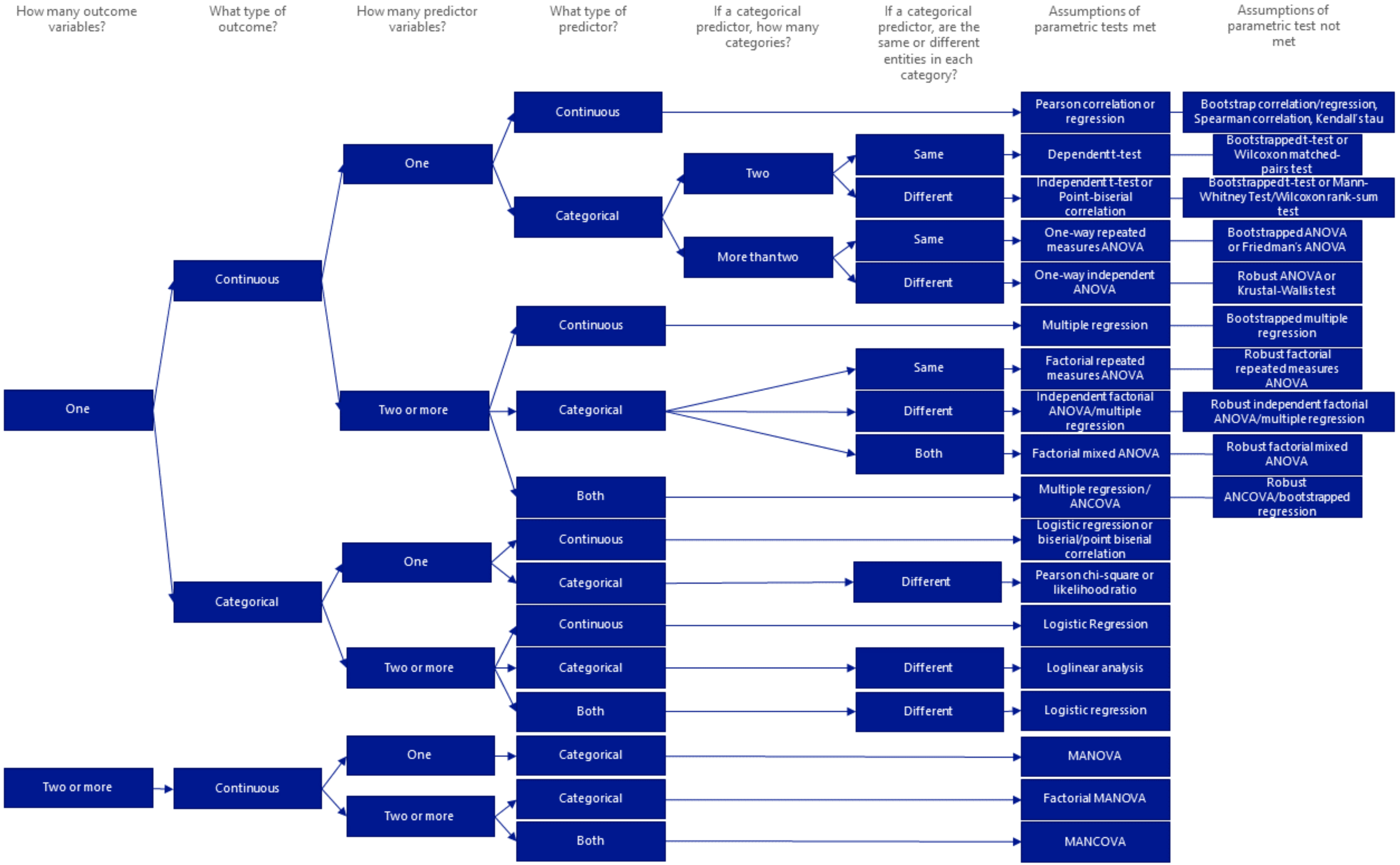
✅

Groups are normally distributed ✅

✅ $\wedge$

⋮ ✅

Why constraints?



Source: Field, Andy, Jeremy Miles and Zoe Field. *Discovering Statistics Using R*. Thousand Oaks: SAGE Publications, Inc., 2012.

Benefits of Tea's Implementation

tea

Extensibility

Support new statistical tests

New test ↔
```
bivariate(x,y)
one_x_variable(x,y)
one_y_variable(x,y)
independent_obs(x,y)
categorical(x)
```

* Tea supports more tests than Statsplorer [Wacharamanotham et al. 2015]

Flexibility

Evolve with statistical best practices

N < 200		N >= 200	
w = .7	`normal_distribution(x)`	w = .4	`normal_distribution(x)`
w = .3	`equal_variance(x,y)`	w = .6	`equal_variance(x,y)`

Tea:

How to use it

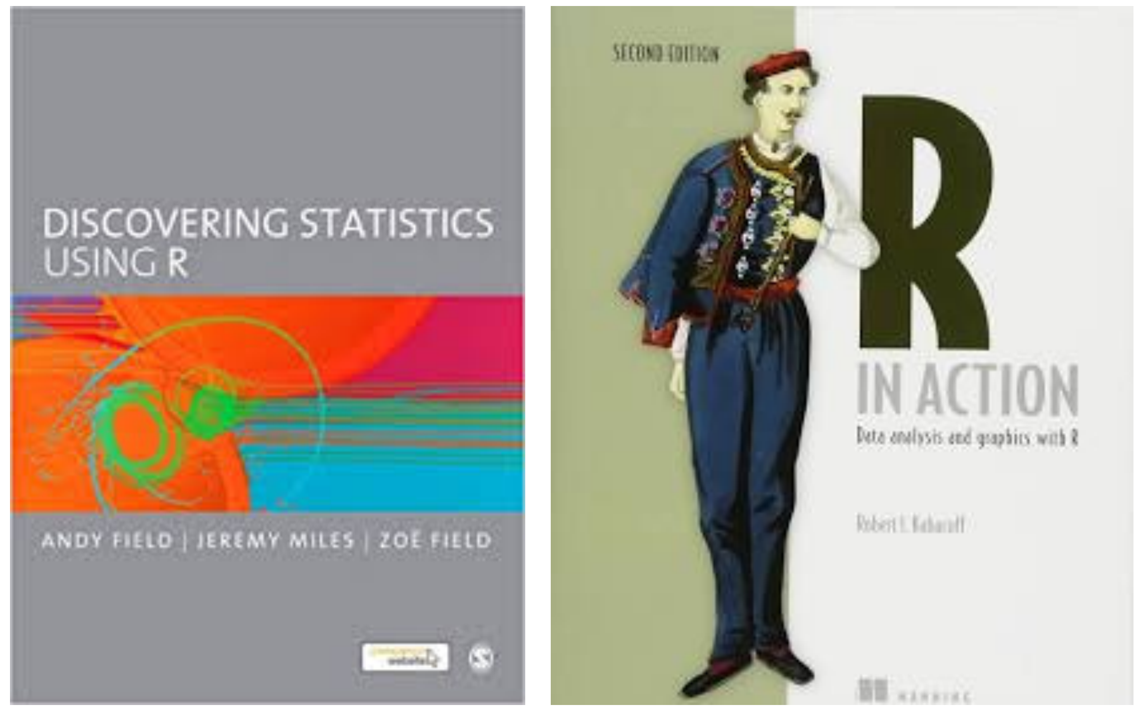
How it works

How it performs

Initial Evaluation

How does Tea compare to experts?

12 tutorials
code snippets + text




```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
    }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
        }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

```
Test: students_t
***Test assumptions:
Exactly two variables involved in analysis: So, Prob
Exactly one explanatory variable: So
Exactly one explained variable: Prob
Independent (not paired) observations: So
Variable is categorical: So
Variable has two categories: So
Continuous (not categorical) data: Prob
Equal variance: So, Prob
Groups are normally distributed: So, Prob

***Test results:
name = Student's T Test
test_statistic = 4.202130736875173
p_value = 0.000123648972665327 75
adjusted_p_value = 6.182448633266387e-05
alpha = 0.05
dof = 45
Effect size:
Cohen's d = 1.2426167296374897
A12 = 0.8366935483870968
Null hypothesis = There is no difference in means between 0 and 1 on Prob.
Interpretation = t(45) = 4.202130736875173, 6.182448633266387e-05. Reject the null hypothesis at alpha = 0.05. The mean of Prob for So = 1 is significantly greater than the mean for So = 0. The effect size is {"Cohen's d": 1.2426167296374897, 'A12': 0.8366935483870968}. The effect size is the magnitude of the difference, which gives a holistic view of the results [1].
[1] Sullivan, G. M., & Feinn, R. (2012). Using effect size—or why the P value is not enough. Journal of Graduate Medical Education, 4(3), 279-282.
```

Replicate 9

Improve 3

How does Tea compare to novices?

data


```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
    }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
        }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```


Avoid common mistakes and false conclusions

Vision: Democratize data science

Lower the barrier to statistical analysis

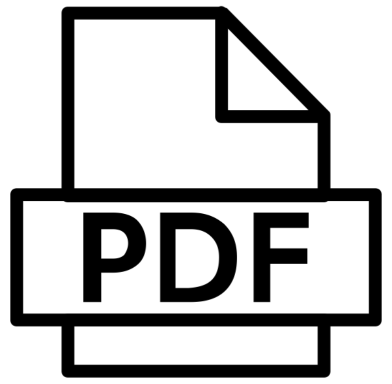
Eiselmayer et al. 2019, Hwang et al. 2016, Wacharamanotham et al. 2015, Guimbretière et al. 2007

Reimagine the ecosystem of tools

Tosch et al. 2019, Bakshy et al. 2014

End-to-end support for iterative data analysis

Tea programs for pre-registration



- Idiosyncratic
- Manual checking

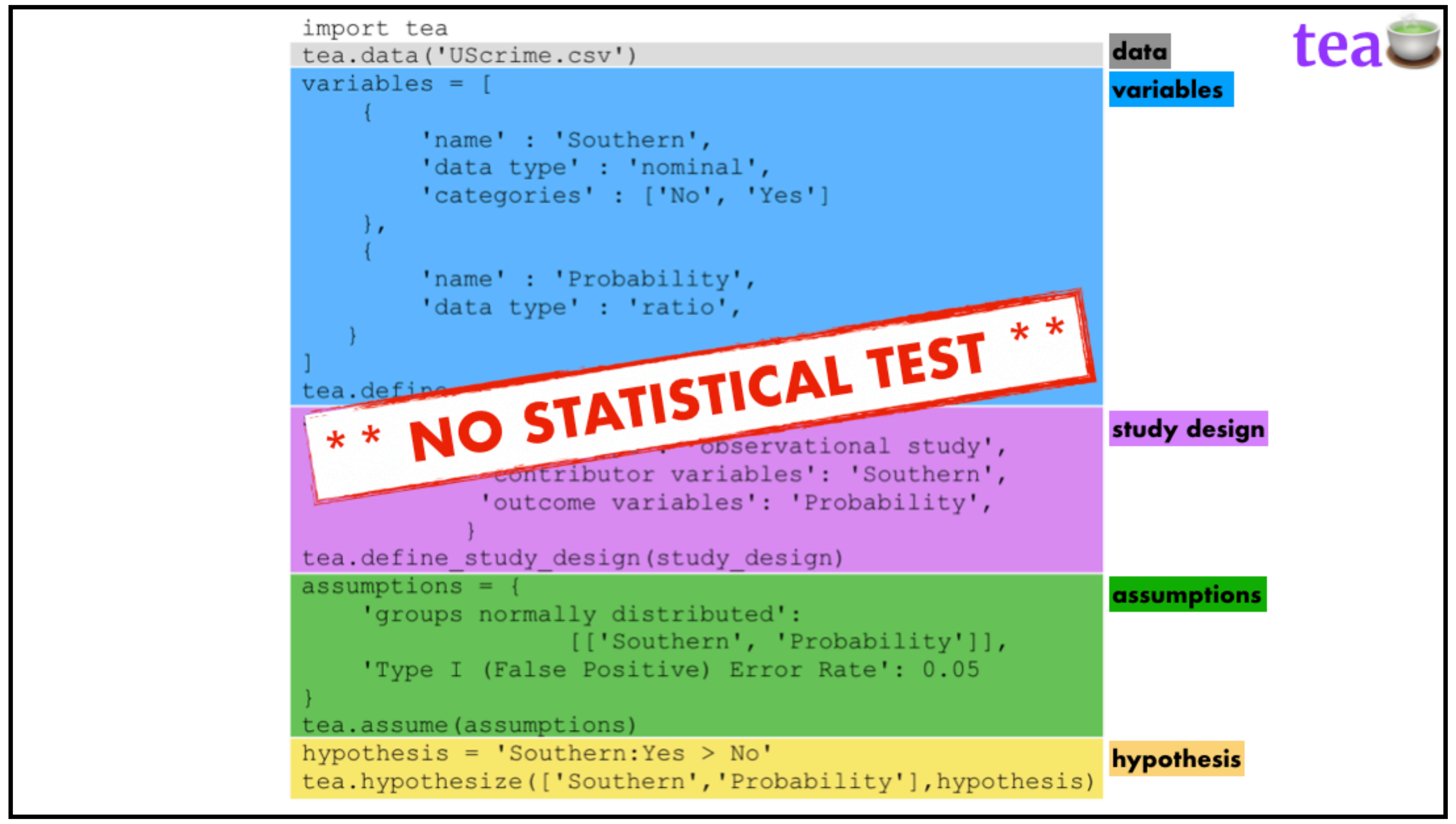
```
import tea
tea.data('UScrime.csv')
variables = [
    {
        'name' : 'Southern',
        'data type' : 'nominal',
        'categories' : ['No', 'Yes']
    },
    {
        'name' : 'Probability',
        'data type' : 'ratio',
   }
]
tea.define_variables(variables)
study_design = {
            'study type': 'observational study',
            'contributor variables': 'Southern',
            'outcome variables': 'Probability',
          }
tea.define_study_design(study_design)
assumptions = {
    'groups normally distributed':
                  [['Southern', 'Probability']],
    'Type I (False Positive) Error Rate': 0.05
}
tea.assume(assumptions)
hypothesis = 'Southern:Yes > No'
tea.hypothesize(['Southern','Probability'],hypothesis)
```

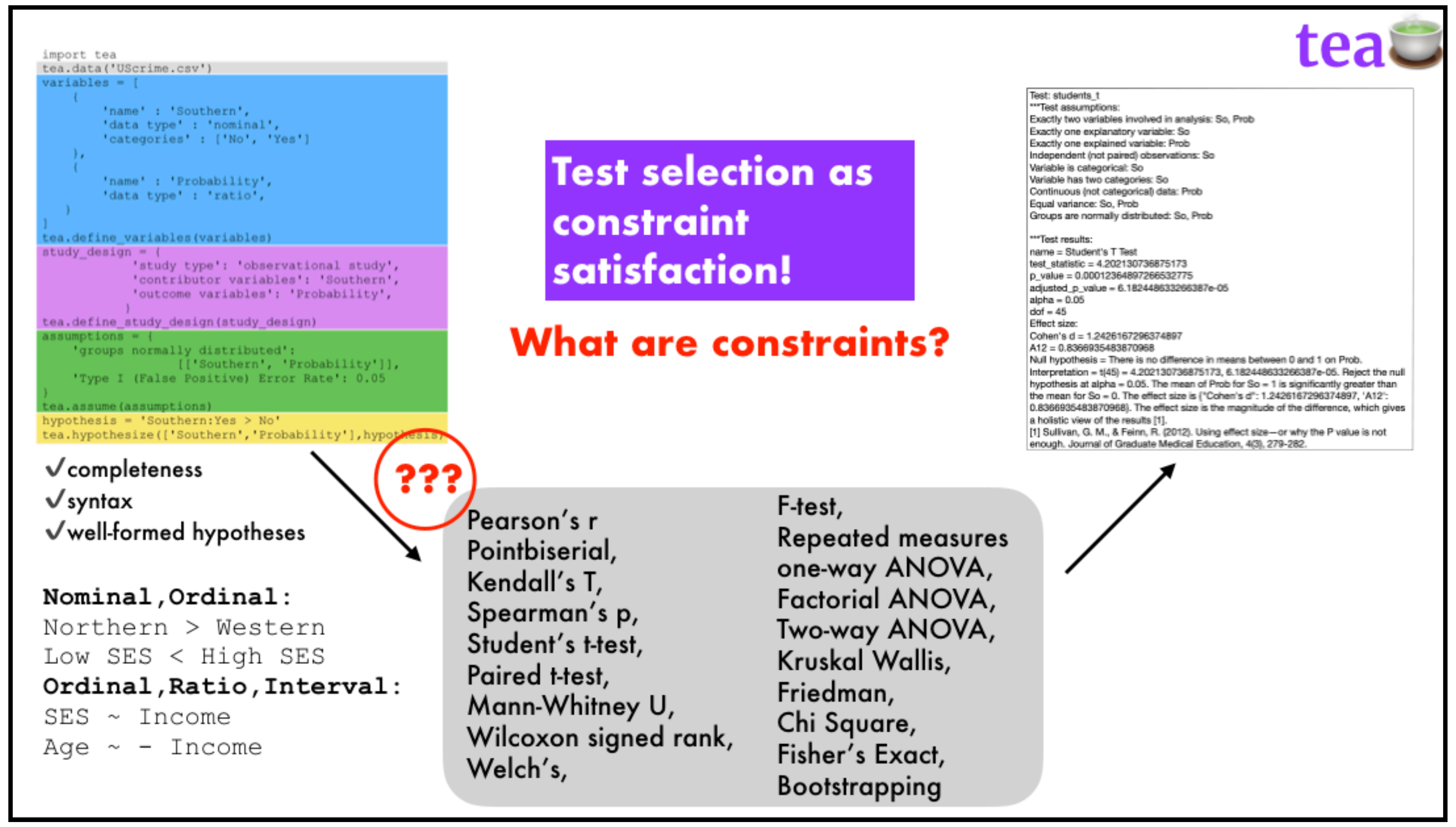
+ Consistent
+ Verifiable
+ Executable

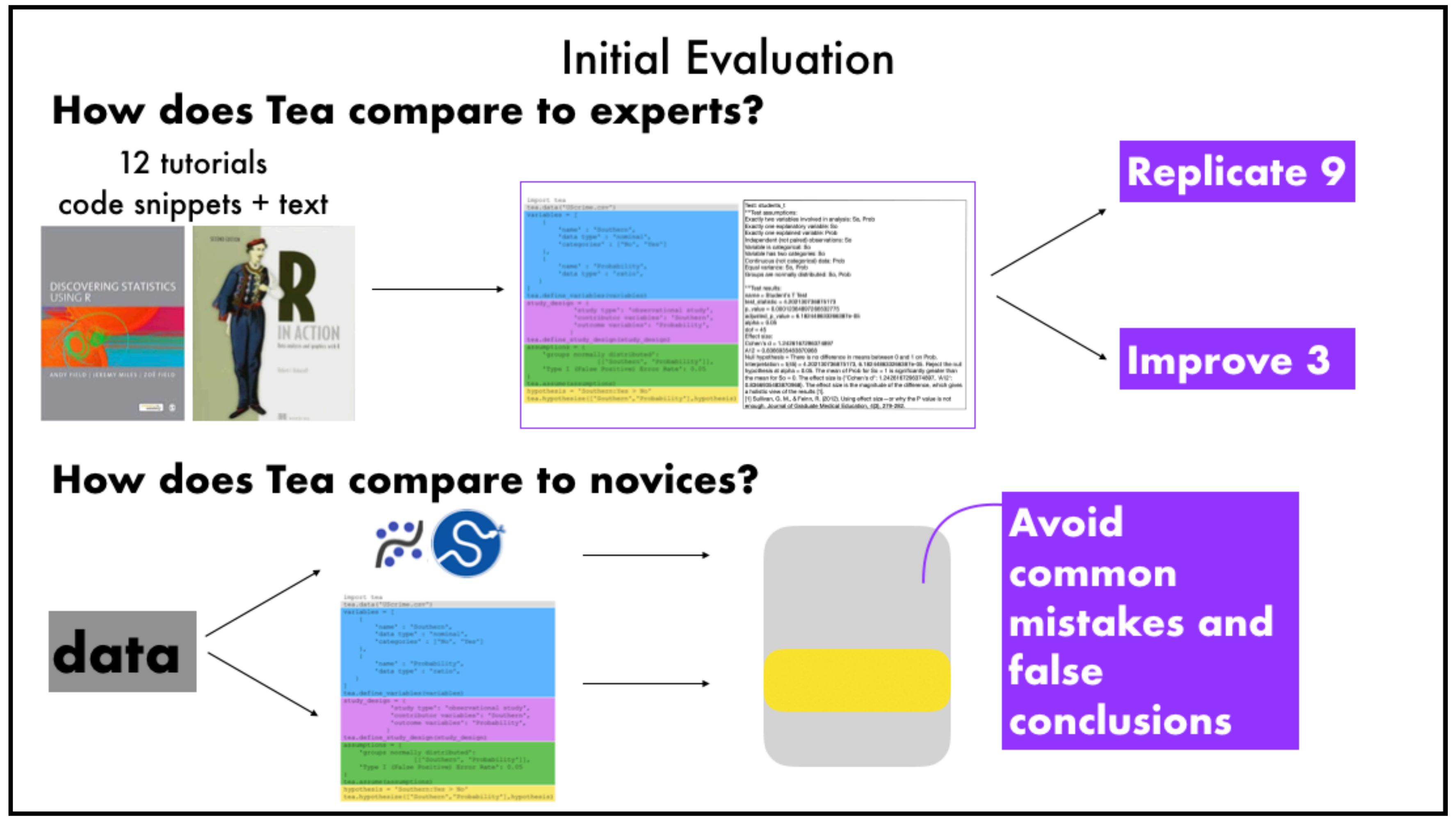
tea

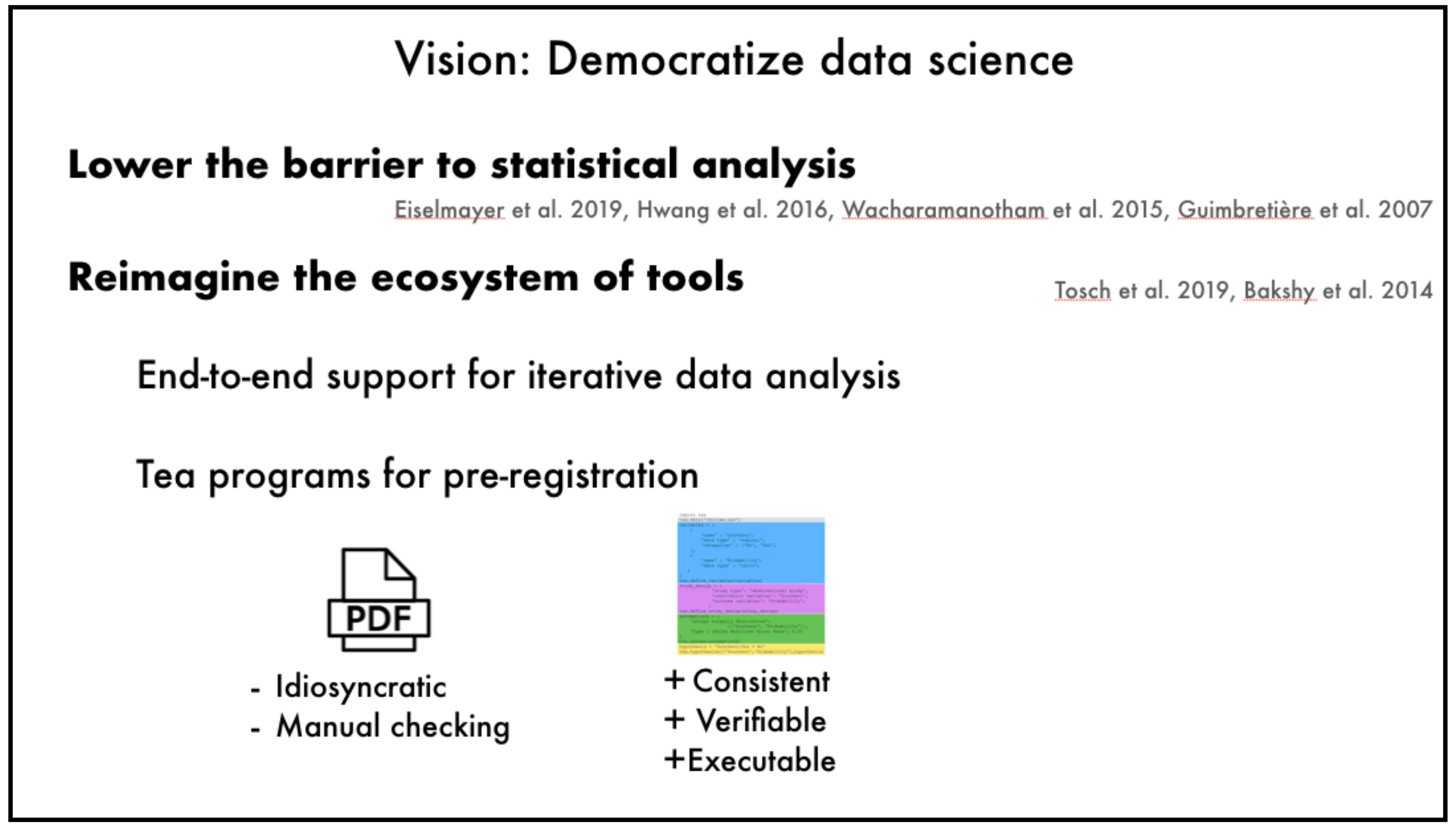
www.tea-lang.org

`pip install tealang`









Eunice Jun @eunicemjun
Maureen Daum
Jared Roesch
Sarah Chasins
Emery Berger
Rene Just
Katharina Reinecke

PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING
UMass Amherst
Berkeley UNIVERSITY OF CALIFORNIA

Limitations with Tea

- Language design
- Implicit conceptual model
- More complex hypotheses
- More complex statistical analyses required

PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING

Tisane:
Authoring Statistical Models via Formal Reasoning from Conceptual and Data Relationships

E. Jun, Audrey Seo, Jeffrey Heer, René Just. ACM CHI 2022.

Tisane: Authoring Statistical Models via Formal Reasoning from Conceptual and Data Relationships

Eunice M. Jun, Audrey Seo, Jeffrey Heer, and René Just | @eunicemjun, emjun@cs.washington.edu

Domain

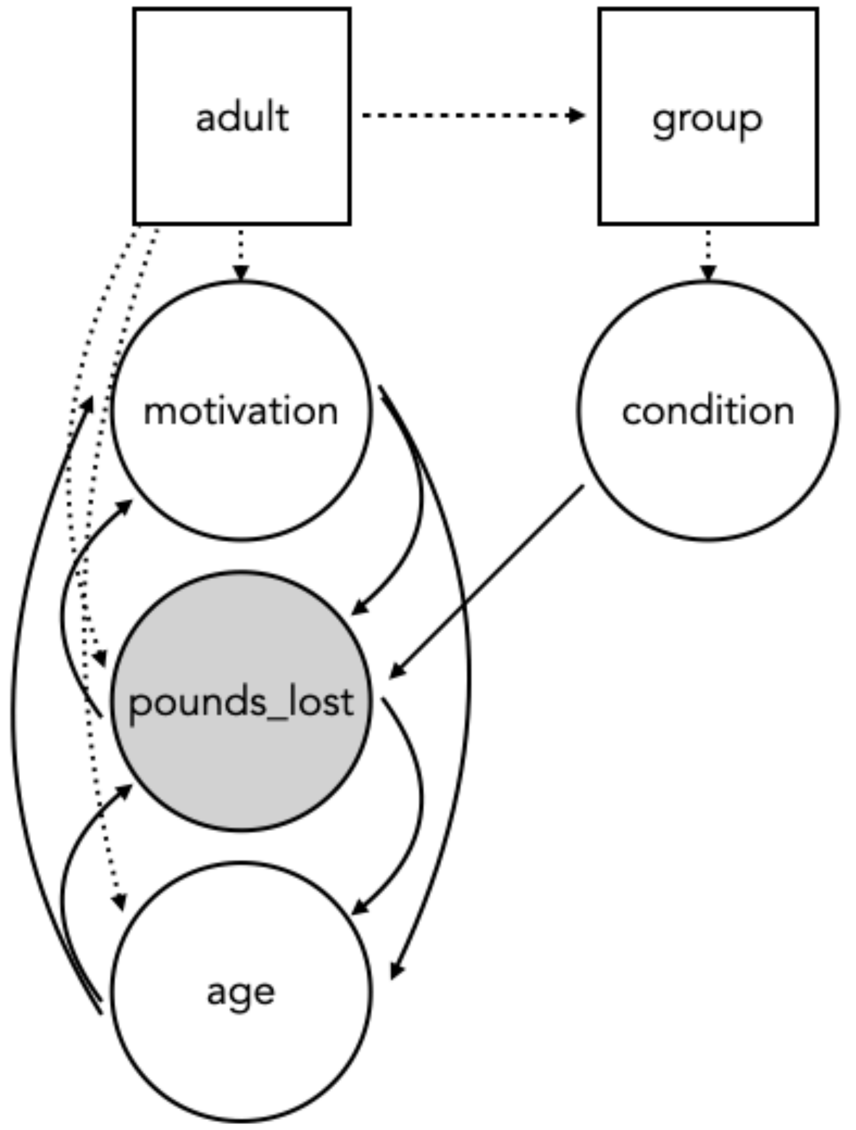
Data

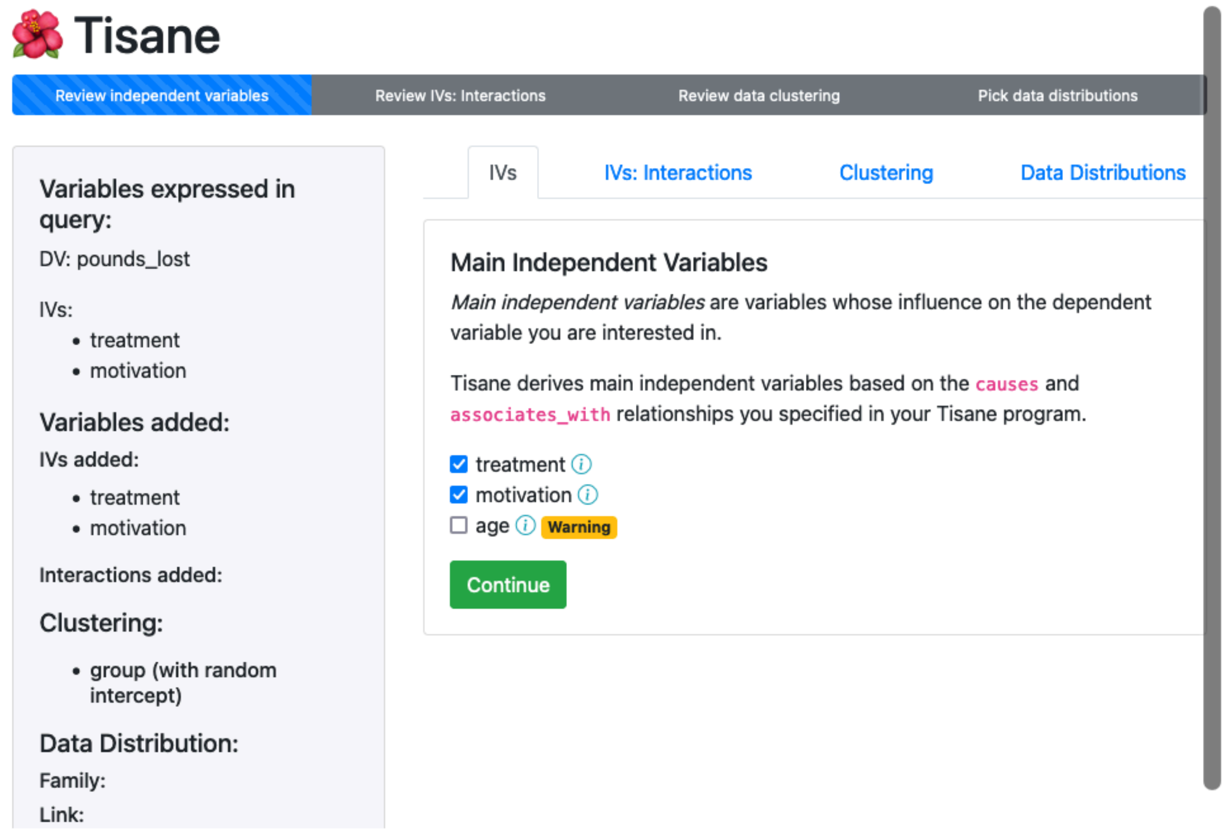
Statistics

```
glm(y ~ x1 + x2,
family=gaussian())
```

Interactive compilation

```
import tisane as ts

adult = ts.Unit("adult", cardinality=386)
motivation = adult.numeric("motivation")
pounds_lost = adult.numeric("pounds_lost")
age = adult.numeric("age")
group = ts.Unit("group", cardinality=40)
condition = group.nominal("treatment", cardinality=2)
adult.nests_within(group)
condition.causes(pounds_lost)
motivation.associates_with(pounds_lost)
age.associates_with(pounds_lost)
age.associates_with(motivation)
```





Python

```
pip install tisane
github.com/emjun/tisane
```

R

```
install.packages("tisaner")
github.com/emjun/tisaner
```

Come to my generals talk on
Monday, March 14 at 2pm PT!

Discussion

#1. Cross-disciplinary teams

#2. Mixed, not staged, process

#3. Qual + Systems + Quant

#4. Highly iterative!

#5. Do people really care?

Outline

- **Initial inspiration**
- **Hypothesis formalization** (empirical work + theory building)
- **Tea** (system)
- **Tisane** (system)
- **Discussion**

Two lenses:

#1.
Programs are UIs.
Programming is HCI.

#2.
PL = Representation
HCI = Interaction

PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING

Tisane:
Authoring Statistical Models via Formal Reasoning from Conceptual and Data Relationships

E. Jun, Audrey Seo, Jeffrey Heer, René Just. ACM CHI 2022.

Scenario: How does exercise affect weight loss?

Adapted from Cohen et al. 2013

Scenario: How does exercise affect weight loss?

386 females 🏋️ 🏋️ 🏋️ 🏋️

🏋️ = approx. 100 females

Adapted from Cohen et al. 2013

Scenario: How does exercise affect weight loss?

386 females 🏋️ 🏋️ 🏋️ 🏋️

40 groups | Group 1 | … | … | Group 40 |

🏋️ = approx. 100 females

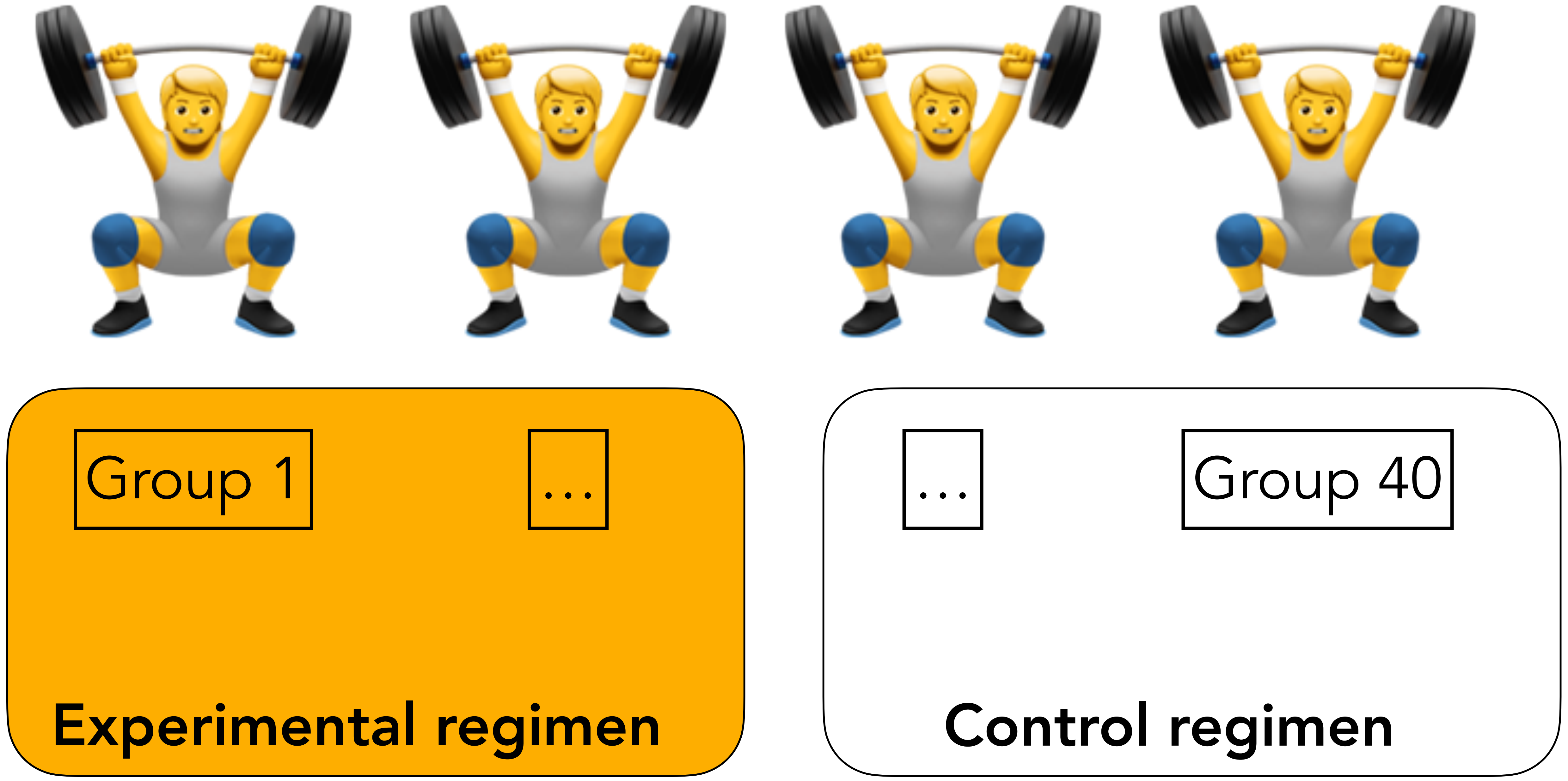
Adapted from Cohen et al. 2013

Scenario: How does exercise affect weight loss?

386 females

40 groups

2 conditions



Adapted from Cohen et al. 2013

Scenario: How does exercise affect weight loss?

386 females

40 groups

Group 1 | ... | ... | Group 40

2 conditions

Experimental regimen | **Control regimen**

 = approx. 100 females

+ motivation scores
+ pounds lost
+ age

Adapted from Cohen et al. 2013

Scenario: How to analyze the data?

386 females

40 groups

Group 1 | … | … | Group 40

2 conditions

Experimental regimen | Control regimen

+ motivation scores
+ pounds lost
+ age

Scenario: How to analyze the data?

Which independent variables should we include?

`Condition`   `Motivation`   `Condition+Motivation`   `Condition+Group`   `???`

Do we include interaction effects?

`Condition*Motivation`   `Condition*Age`   `Condition*Motivation*Group`   `???`

How do we account for grouping?

`Fixed effect?`   `Random effect?`   `Does it matter???`

What type of linear model should we use?

`Linear regression`   `Logistic regression`   `Mixed-effects model`   `???`

Domain

Data

Statistics

Domain

Data

Statistics

```
glm(y ~ x1 + x2, family=gaussian())
```

Tisane enables users to
(i) **express + leverage existing knowledge** and
(ii) **ensures alignment** of considerations.

Domain

Data

Statistics

```
glm(y ~ x1 + x2, family=gaussian())
```

Tisane

Study design specification language { Domain, Data

Model generation + Disambiguation { Statistics

Final model output {

```
glm(y ~ x1 + x2,
family=gaussian())
```

Tisane

Interactive compilation

Study design specification language

Domain

Data

Model generation + Disambiguation

Statistics

Final model output

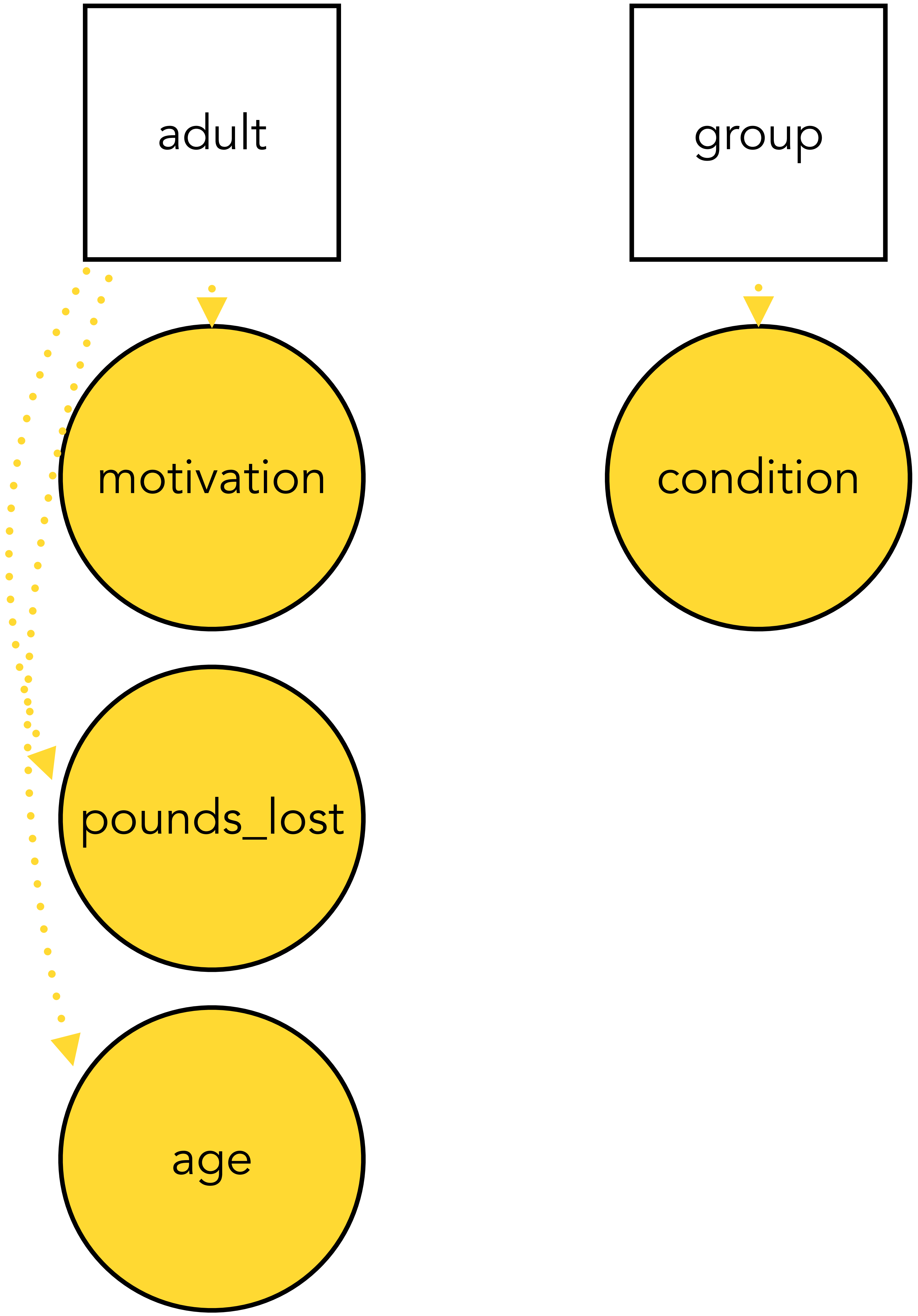
```
glm(y ~ x1 + x2,
family=gaussian())
```

Brew a Tisane program

```
import tisane as ts

adult = ts.Unit(“adult", cardinality=386)

group = ts.Unit(“group", cardinality=40)
```

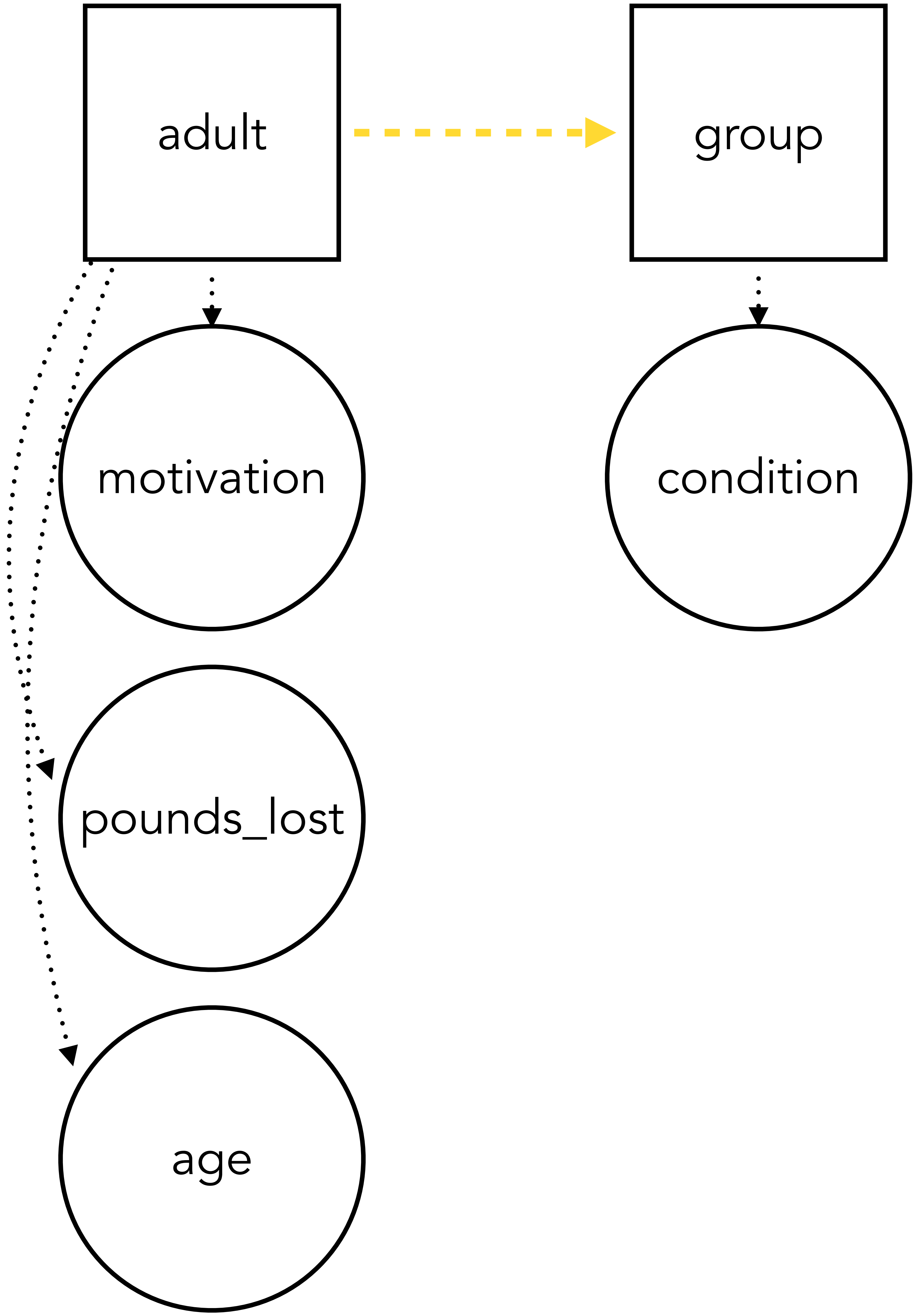
Brew a Tisane program

```
import tisane as ts

adult = ts.Unit(“adult", cardinality=386)
motivation = adult.numeric(“motivation")
pounds_lost = adult.numeric(“pounds_lost")
age = adult.numeric(“age”)
group = ts.Unit(“group", cardinality=40)
condition = group.nominal("treatment", cardinality=2)
```



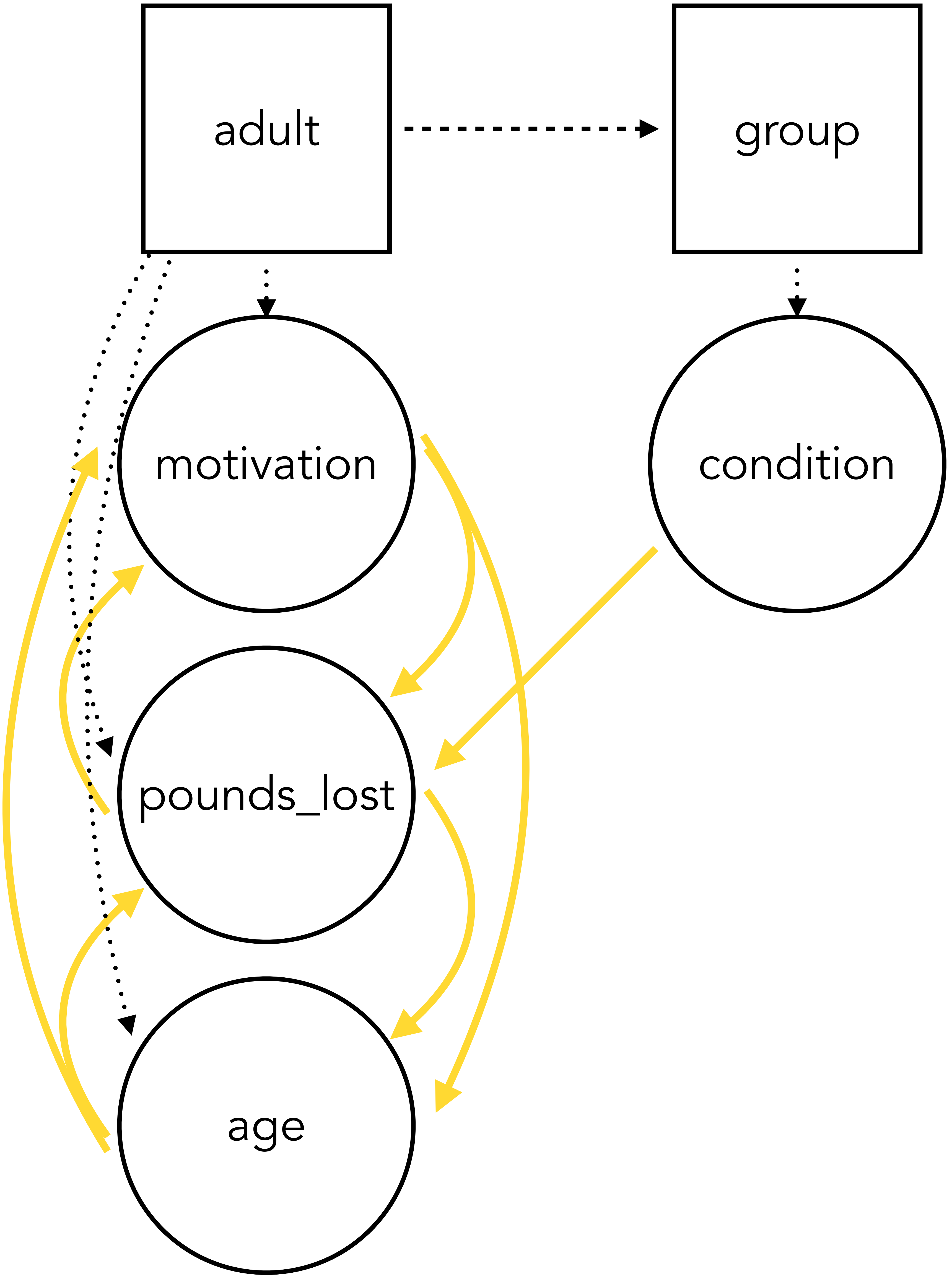
Brew a Tisane program

```
import tisane as ts

adult = ts.Unit("adult", cardinality=386)
motivation = adult.numeric("motivation")
pounds_lost = adult.numeric("pounds_lost")
age = adult.numeric("age")
group = ts.Unit("group", cardinality=40)
condition = group.nominal("treatment", cardinality=2)
adult.nests_within(group)
```



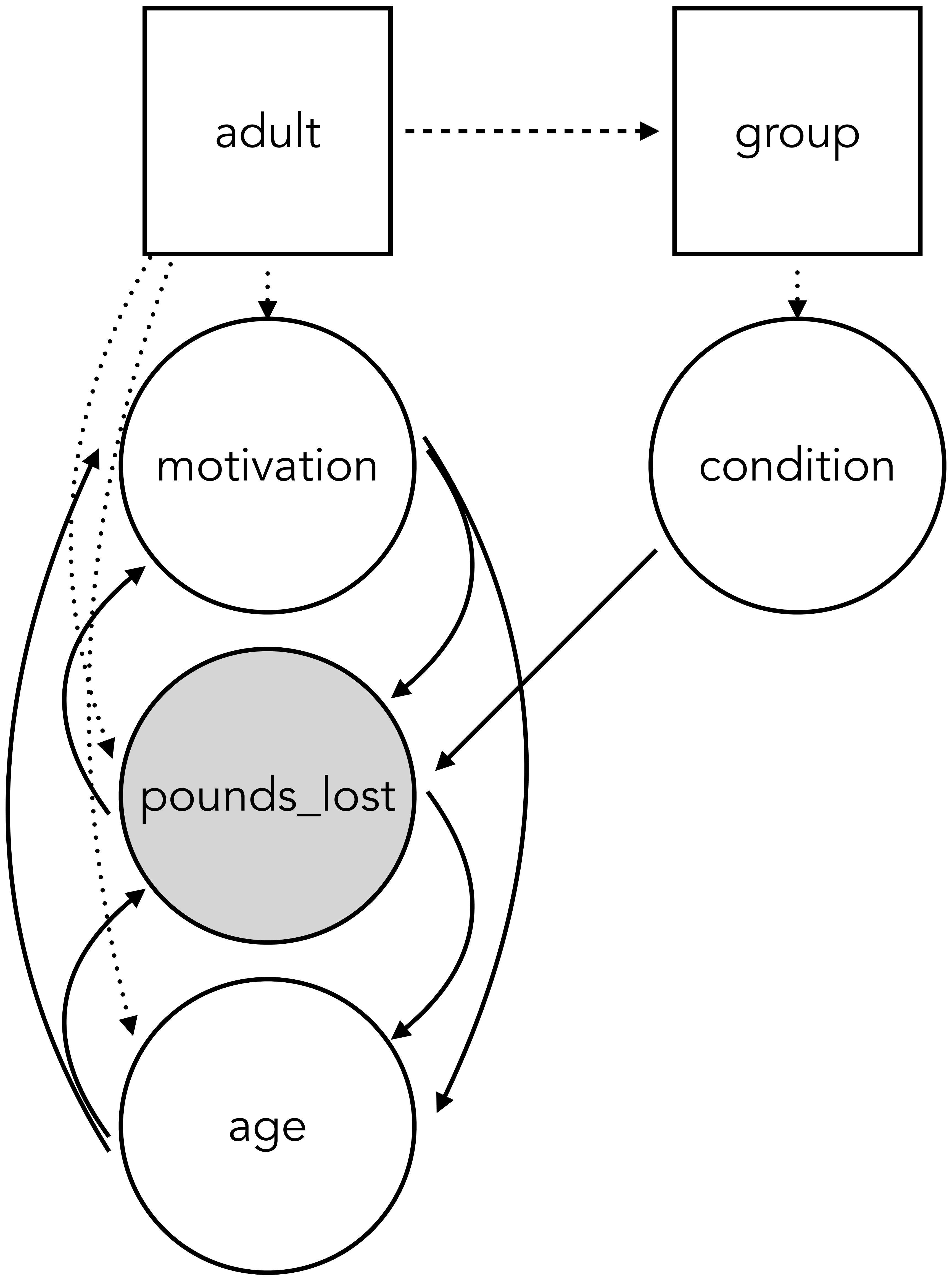
Brew a Tisane program

```
import tisane as ts

adult = ts.Unit(“adult", cardinality=386)
motivation = adult.numeric(“motivation")
pounds_lost = adult.numeric(“pounds_lost")
age = adult.numeric(“age”)
group = ts.Unit(“group", cardinality=40)
condition = group.nominal("treatment", cardinality=2)
adult.nests_within(group)
condition.causes(pounds_lost)
motivation.associates_with(pounds_lost)
age.associates_with(pounds_lost)
age.associates_with(motivation)
```

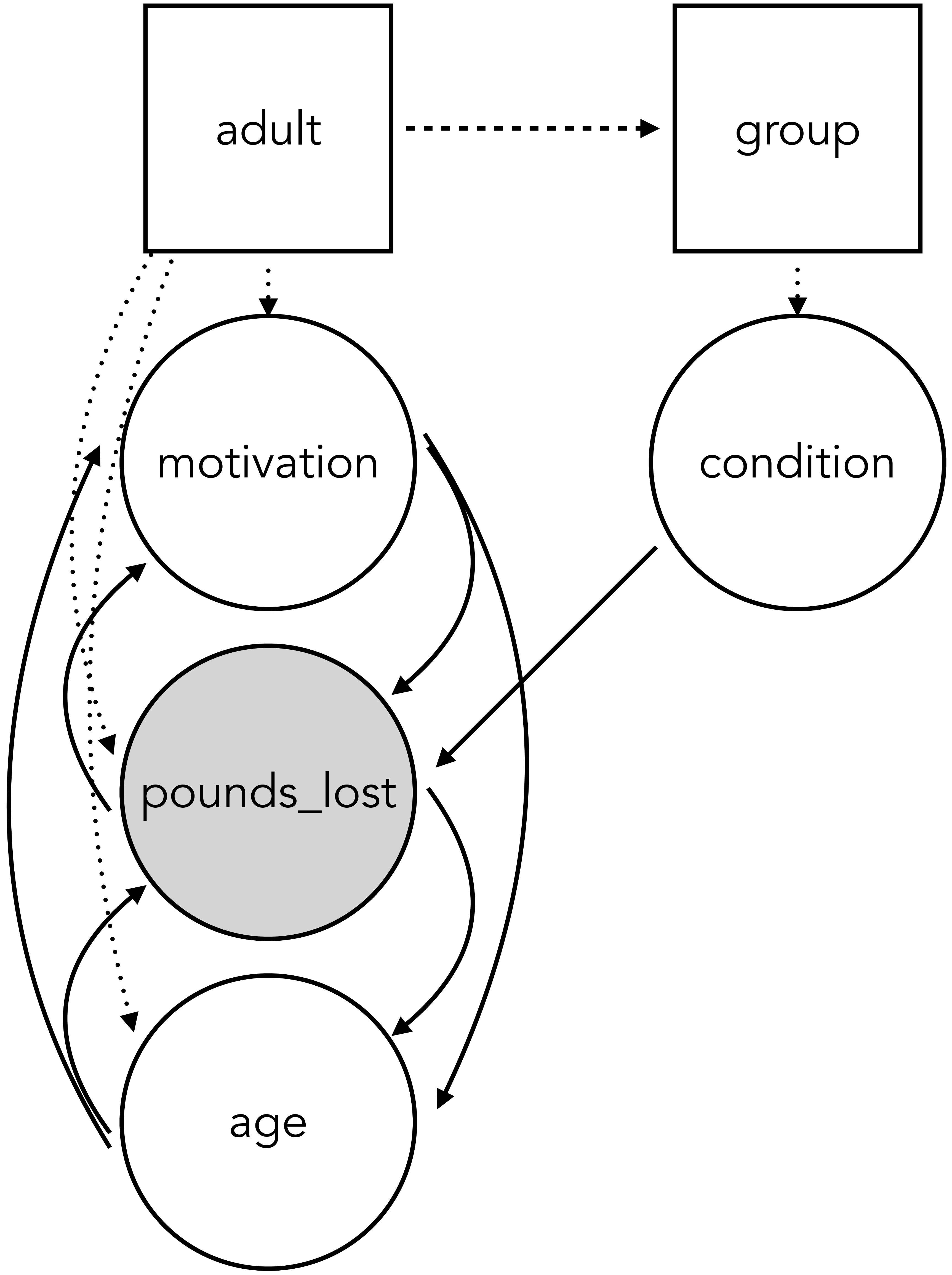


Brew a Tisane program

```
import tisane as ts

adult = ts.Unit(“adult", cardinality=386)
motivation = adult.numeric(“motivation")
pounds_lost = adult.numeric(“pounds_lost")
age = adult.numeric(“age”)
group = ts.Unit(“group", cardinality=40)
condition = group.nominal("treatment", cardinality=2)

adult.nests_within(group)
condition.causes(pounds_lost)
motivation.associates_with(pounds_lost)
age.associates_with(pounds_lost)
age.associates_with(motivation)
design = ts.Design(dv=pounds_lost,
                   ivs=[condition, motivation])
               .assign_data(“data.csv”)

ts.infer_model(design=design)
```



Need user input



Which independent variables should we include?

Check, infer based on graph.
Is age part of the user's research question?

Do we include interaction effects?

Look for moderating relationships.

How do we account for grouping?

Infer maximal random effects to maximize generalizability.
Correlated slope and intercept?

What type of linear model should we use?

Infer possible residual distributions from variable data types.
What will the data look like?

File Edit View Insert Cell Kernel Widgets Help Trusted chi-demo

Run Code

In [ ]:
```python
import tisane as ts

import pandas as pd
import numpy as np
import os
```

Load data

In [ ]:
```python
df = pd.read_csv("exercise_group_age_added.csv")
```

Specify variables

In [ ]:
```python
import tisane as ts

adult = ts.Unit("member", cardinality=386)
motivation = adult.numeric("motivation")
pounds_lost = adult.numeric("pounds_lost")
age = adult.numeric("age")

group = ts.Unit("group", cardinality=40)
condition = group.nominal("treatment",cardinality=2)
```

Specify relationships

In [ ]:
```python
adult.nests_within(group)

condition.causes(pounds_lost)
motivation.associates_with(pounds_lost)
age.associates_with(motivation)
```

*Jupyter notebook not required, also runs outside!

Final model: Avoid common mistakes.

```
pounds_lost ~ motivation + treatment + (1|group)
```

Conceptually founded, maximal random effects

```
pounds_lost~motivation+treatment
```

Overlook groups, inflate statistical power

```
pounds_lost~motivation+treatment + group
```

"Ecological fallacy," inflate statistical power

```
pounds_lost~group_motivation+group_treatment
```

Average across groups, deflate statistical power

Tisane

Interactive compilation

Study design specification language — Domain, Data

Model generation + Disambiguation — Statistics

Final model output

```
glm(y ~ x1 + x2,
family=gaussian())
```

Case studies:

🧠
Psychology

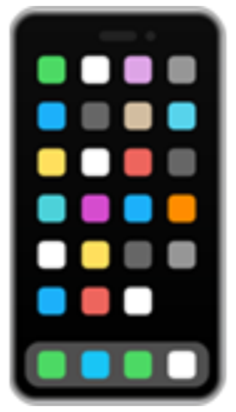
📱
HCI

🌎
Health policy

Case studies: Impact on workflows

Psychology

"…in terms of I don't know [what] I was exactly picking, because there's like, what is it like 'poisson regression' or whatever, right. And like, you have to pick these things in SPSS. And like, I honestly, ***admittedly did not really look into which I should have been picking****, but I just had one of his previous students [who] was like, 'This is what I did. So you should just do that.'…these are like,* ***major gaps****….****[Tisane] fills in a lot of gaps*** *in that, in that sense, in the sense of like, I think* ***one of the biggest issues*** *for psychologists is like what tests to run? And I don't think anyone ever has a very good answer."*



HCI

"I think that like, like, so close to a deadline, it's a little bit unnerving to be like, 'Oh, f*ck what I just wrote about could be incorrect.' And then also, it's like, but also, **if it's incorrect, I should know before I submit**. So I feel like a little bit of that tension with it….And now I like **know, of some stuff I didn't know about before**."

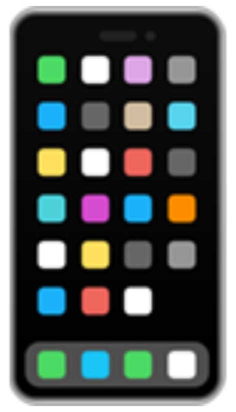
Health policy

"But what I think I could use…to help ***fill that gap in my knowledge****, and some of the places where I'm not sure about how to set things up….if we're interested in in linear models with mixed effects, then this seems like it would do it."*

Case studies: Cognitive fixation

Psychology

*“Yeah, **I keep [study design] in my head, which I probably shouldn't.** And that when I, I guess, run tests, I just, I only plop in the variables I'm looking at at that moment.”*



HCI

*“Okay, so I think that in this case, what I want to add is that each of the independent variables causes dissociation. **I'm actually not sure. Is it possible?** Or is that just correlated…**I don't feel comfortable.** We can just say it's associated.”*

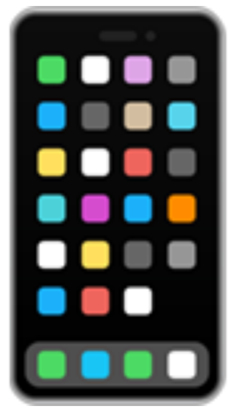
Health policy

*“[Tisane] would be interesting in any of those cases, because it would **help you explore your relationships** pretty easily would help you, you know, **fit a really simple model, but in the best way you can.** So if I say, ‘Hey, like here, I want these things in there,’ [Tisane] would be like, ‘Well, you know, I guess you know, here's probably a good way to set that up.’ And then you could kind of easily get some plots that you don't need to write code for.”*

Case studies: Future possibilities

Psychology

“ But is there yet anywhere that you might be able to specify, like, ***I want to control for this*** and not have a factor into really like this relationship? Or I guess I want to factor in but ***insofar as it's acts as a control and not as like a real variable.***"



HCI

“…the only thing that feels like a little difficult is, like, ***knowing the number of instances.*** I don't know why … a ***DSM [Diary Study Me***… o like, can vary so much bet…

Streamline specification for simpler models,
Guide prototyping for more complex models

Health policy

“…make the app more ***able to be run without like the mouse***….you could run this 2000 times in the parallel session….[T]he benefit of this isn't just that it spits out the best model for you. It's also that it's ***exploratory***, you know, what I mean? So, it could be useful in an exploratory way, just for… like, you know, I can ***look at one model and kind of infer that the others are similar*** and do some ***spot checking*** as well. Definitely seems like a ***good first place to go.***"

Tisane: Authoring Statistical Models via Formal Reasoning from Conceptual and Data Relationships

Eunice M. Jun, Audrey Seo, Jeffrey Heer, and René Just | @eunicemjun, emjun@cs.washington.edu

Domain

Data

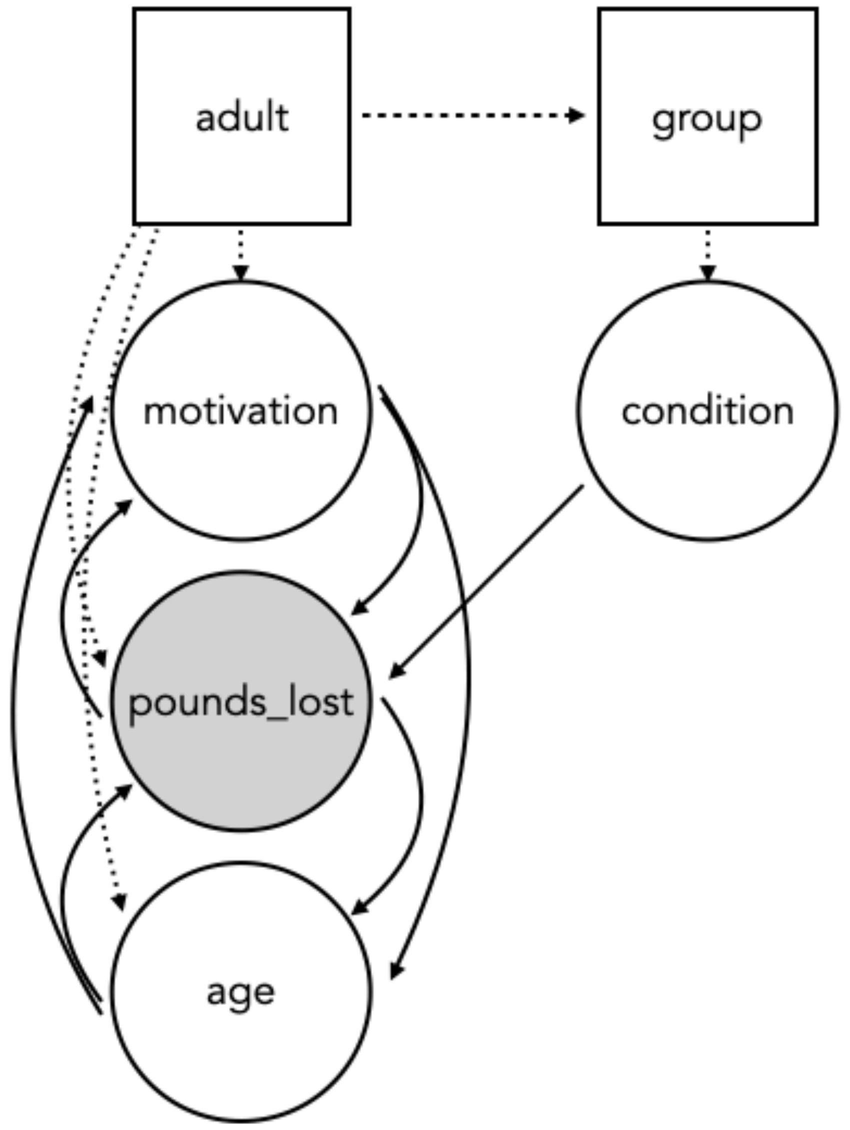
Statistics

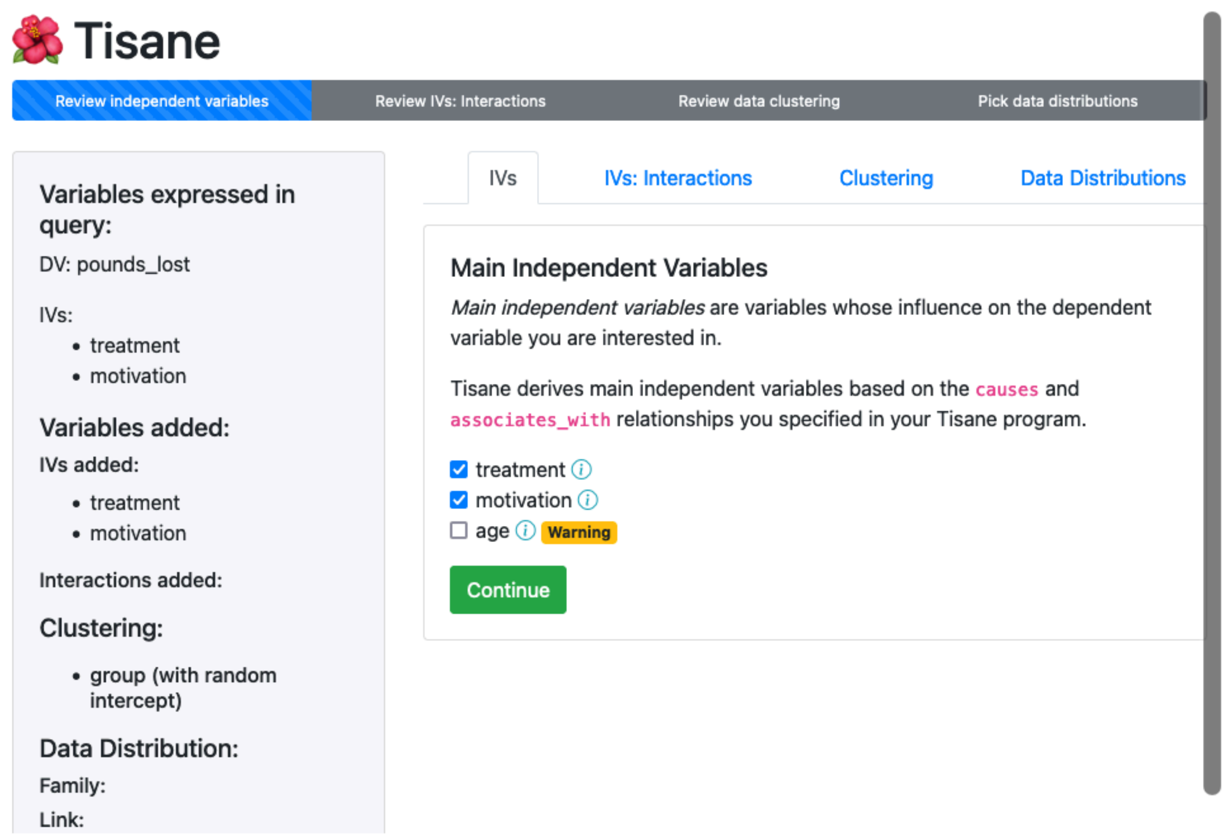
↓

```
glm(y ~ x1 + x2,
family=gaussian())
```

Interactive compilation

```
import tisane as ts

adult = ts.Unit("adult", cardinality=386)
motivation = adult.numeric("motivation")
pounds_lost = adult.numeric("pounds_lost")
age = adult.numeric("age")
group = ts.Unit("group", cardinality=40)
condition = group.nominal("treatment", cardinality=2)
adult.nests_within(group)
condition.causes(pounds_lost)
motivation.associates_with(pounds_lost)
age.associates_with(pounds_lost)
age.associates_with(motivation)
```





Python

```
pip install tisane
github.com/emjun/tisane
```

R

```
install.packages("tisaner")
github.com/emjun/tisaner
```