Data Science 1

STAT/CS 287

Jim Bagrow, UVM Dept of Math and Statistics

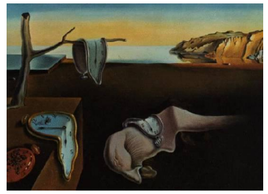
LECTURE 07

More on "tabular" data

We spoke previously about storing tabular data, but at a base level, dealing with file formats, row and column delimiters, and other details.

Let's expand the scope somewhat.

Tidy data



Journal of Statistical Software

August 2014, Volume 59, Issue 10. *http://www.jstatsoft.org/*

Tidy Data

Hadley Wickham
RStudio

H. Wickham, 2014

Tidy data — Motivation

Journal of Statistical Software
August 2014, Volume 59, Issue 10. *http://www.jstatsoft.org/*

Tidy Data

Hadley Wickham
RStudio

Abstract (excerpt)

A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible.

H. Wickham, 2014

Tidy data — Motivation

Journal of Statistical Software
August 2014, Volume 59, Issue 10. *http://www.jstatsoft.org/*

Tidy Data

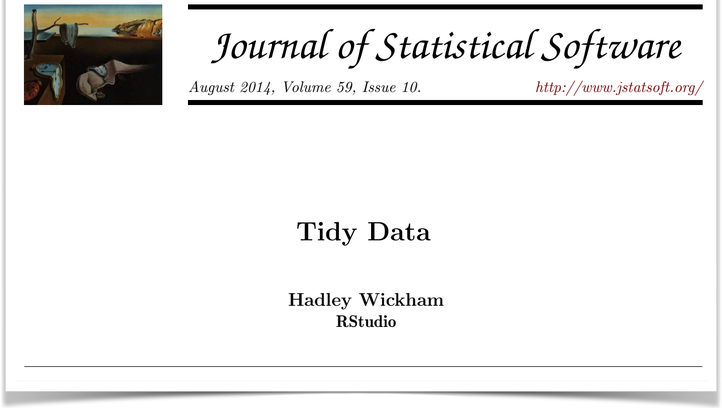
Hadley Wickham
RStudio

Abstract (excerpt)

A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible.

H. Wickham, 2014

Tidy data — Motivation



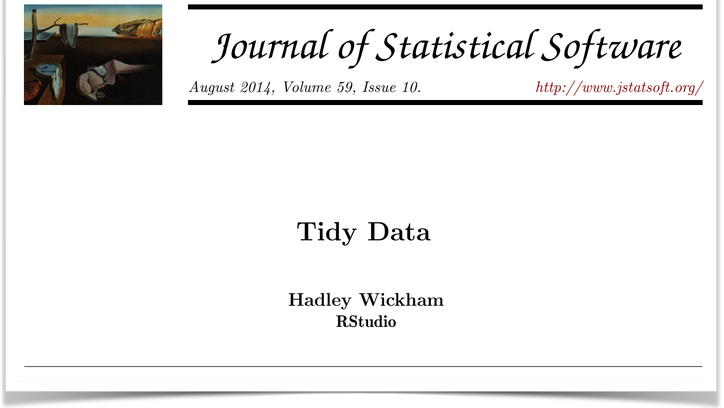
Abstract (excerpt)

A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible.

This paper tackles a small, but important, component of data cleaning: data tidying. Tidy datasets are easy to manipulate, model and visualize, and have a specific structure: each variable is a column, each observation is a row, and each type of observational unit is a table.

H. Wickham, 2014

Tidy data — Motivation



Abstract (excerpt)

A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible.

This paper tackles a small, but important, component of data cleaning: data tidying. Tidy datasets are easy to manipulate, model and visualize, and have a specific structure: each variable is a column, each observation is a row, and each type of observational unit is a table.

H. Wickham, 2014

Tidy data — Motivation

Journal of Statistical Software
August 2014, Volume 59, Issue 10. *http://www.jstatsoft.org/*

Tidy Data

Hadley Wickham
RStudio

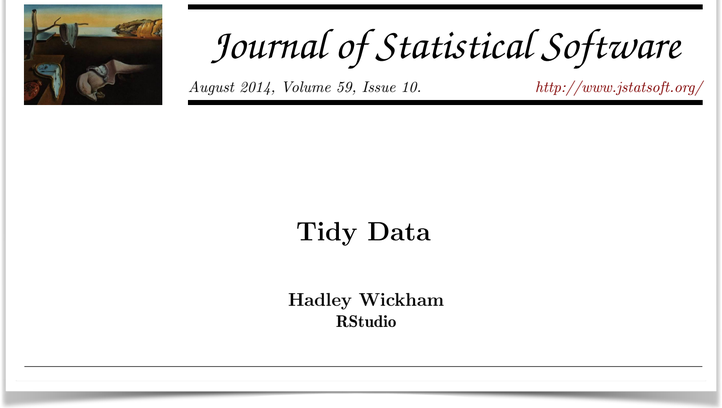
Abstract (excerpt)

A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible.

This paper tackles a small, but important, component of data cleaning: data tidying. Tidy datasets are easy to manipulate, model and visualize, and have a specific structure: each variable is a column, each observation is a row, and each type of observational unit is a table.

H. Wickham, 2014

Tidy data — Motivation



Journal of Statistical Software

August 2014, Volume 59, Issue 10. *http://www.jstatsoft.org/*

Tidy Data

Hadley Wickham
RStudio

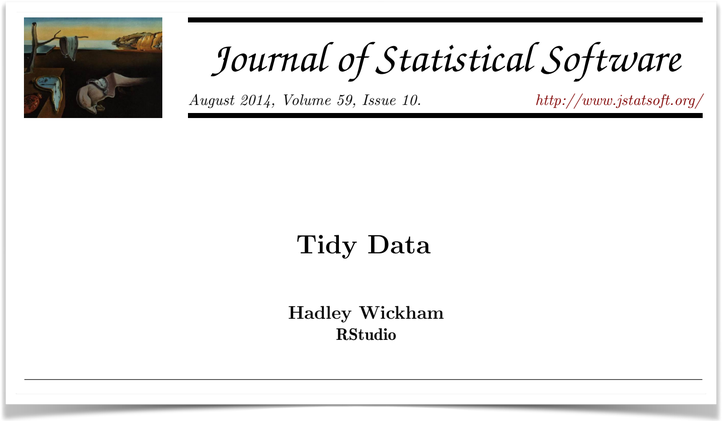
Abstract (excerpt)

A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible.

This paper tackles a small, but important, component of data cleaning: data tidying. Tidy datasets are easy to manipulate, model and visualize, and have a specific structure: each variable is a column, each observation is a row, and each type of observational unit is a table.

H. Wickham, 2014

Tidy data — Motivation



Journal of Statistical Software
August 2014, Volume 59, Issue 10. *http://www.jstatsoft.org/*

Tidy Data

Hadley Wickham
RStudio

"*Tidy*" is now a term of art!

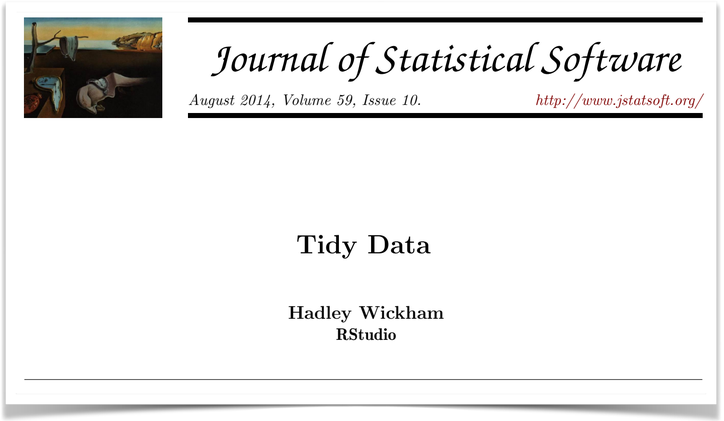
Abstract (excerpt)

A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible.

This paper tackles a small, but important, component of data cleaning: data tidying. Tidy datasets are easy to manipulate, model and visualize, and have a specific structure: each variable is a column, each observation is a row, and each type of observational unit is a table.

H. Wickham, 2014

Tidy data — Motivation



"*Tidy*" is now a term of art!

Abstract (excerpt)

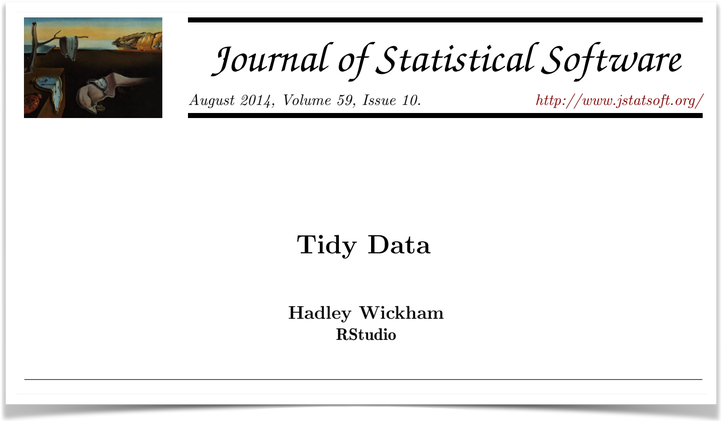
A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible.

This paper tackles a small, but important, component of data cleaning: data tidying. Tidy datasets are easy to manipulate, model and visualize, and have a specific structure: each variable is a column, each observation is a row, and each type of observational unit is a table.

This framework makes it easy to tidy messy datasets because only a small set of tools are needed to deal with a wide range of un-tidy datasets. This structure also makes it easier to develop tidy tools for data analysis, tools that both input and output tidy datasets.

H. Wickham, 2014

Tidy data — Motivation



"*Tidy*" is now a term of art!

Abstract (excerpt)

A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible.

This paper tackles a small, but important, component of data cleaning: data tidying. Tidy datasets are easy to manipulate, model and visualize, and have a specific structure: each variable is a column, each observation is a row, and each type of observational unit is a table.

This framework makes it easy to tidy messy datasets because only a small set of tools are needed to deal with a wide range of un-tidy datasets. This structure also makes it easier to develop tidy tools for data analysis, tools that both input and output tidy datasets.

H. Wickham, 2014

Datasets — Terminology

Data Structure

Most datasets organized* as ***tables*** of ***rows*** and ***columns***

- Columns often *labeled*
- Rows sometimes labeled

The same data can be organized in different ways.
For example:

* or, can be organized

H. Wickham, 2014

Datasets — Terminology

Data Structure

Most datasets organized* as *tables* of *rows* and *columns*

- Columns often *labeled*
- Rows sometimes labeled

The same data can be organized in different ways.
For example:

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

* or, can be organized

H. Wickham, 2014

Datasets — Terminology

Data Structure

Most datasets organized* as *tables* of *rows* and *columns*

- Columns often *labeled*
- Rows sometimes labeled

The same data can be organized in different ways.
For example:

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

Terminology

A dataset is a collection of ***values***

- *numbers* (quant.) or *strings* (qual.)+

* or, can be organized

+ However: Stevens' levels of measurement!

H. Wickham, 2014

Datasets — Terminology

Data Structure

Most datasets organized* as *tables* of *rows* and *columns*

- Columns often *labeled*
- Rows sometimes labeled

The same data can be organized in different ways.
For example:

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

* or, can be organized

Terminology

A dataset is a collection of ***values***

- *numbers* (quant.) or *strings* (qual.)+

Values are organized into ***variables*** and ***observations***

+ However: Stevens' levels of measurement!

H. Wickham, 2014

Datasets — Terminology

Data Structure

Most datasets organized* as *tables* of *rows* and *columns*

- Columns often *labeled*
- Rows sometimes labeled

The same data can be organized in different ways.
For example:

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

Terminology

A dataset is a collection of ***values***

- *numbers* (quant.) or *strings* (qual.)+

Values are organized into ***variables*** and ***observations***

A **variable** contains all the values measuring the same underlying quantity (height, temperature, etc.) across observations

An **observation** contains all the values associated with the same **unit** (participant, day, demographic group, etc.)

* or, can be organized

+ However: Stevens' levels of measurement!

H. Wickham, 2014

Datasets — Terminology

Data Structure

Most datasets organized* as *tables* of *rows* and *columns*

- Columns often *labeled*
- Rows sometimes labeled

The same data can be organized in different ways.
For example:

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

* or, can be organized

Terminology

A dataset is a collection of ***values***

- *numbers* (quant.) or *strings* (qual.)+

Values are organized into ***variables*** and ***observations***

A **variable** contains all the values measuring the same underlying quantity (height, temperature, etc.) across observations

An **observation** contains all the values associated with the same **unit** (participant, day, demographic group, etc.)

Each **type** of observational unit is grouped into a **table**

+ However: Stevens' levels of measurement!

H. Wickham, 2014

Datasets — Terminology

Terminology

→ **Four components** to a dataset:
- Values
- Variables
- Observations
- Table(s)

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

H. Wickham, 2014

Datasets — Terminology

Terminology

→ **Four components** to a dataset:
- Values
- Variables
- Observations
- Table(s)

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

H. Wickham, 2014

Datasets — Terminology

Terminology

→ **Four components** to a dataset:
- Values
- Variables
- Observations
- Table(s)

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

When *confronted* with a dataset, always *consider* what these components are

H. Wickham, 2014

Datasets — Terminology

Terminology

→ **Four components** to a dataset:
- Values
- Variables
- Observations
- Table(s)

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

When *confronted* with a dataset, always *consider* what these components are

Variable	**Values**
person	John Smith, Jane Doe, Mary Johnson
treatment	a, b
result	16, 3, 2, 11, 1 (and possibly —)

H. Wickham, 2014

Datasets — Terminology

Terminology

→ **Four components** to a dataset:
- Values
- Variables
- Observations
- Table(s)

When *confronted* with a dataset, always *consider* what these components are

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

Variable	**Values**
person	John Smith, Jane Doe, Mary Johnson
treatment	a, b
result	16, 3, 2, 11, 1 (and possibly —)

Observations?

People?

Treatments?

Multiple options

H. Wickham, 2014

Messy vs. Tidy datasets

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

Reorganize into a **standard form** ("**tidy**"):

Variable	**Values**
person	John Smith, Jane Doe, Mary Johnson
treatment	a, b
result	16, 3, 2, 11, 1 (and possibly —)

Observations?

People? ***Multiple options***

Treatments?

H. Wickham, 2014

Messy vs. Tidy datasets

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

Variable	**Values**
person	John Smith, Jane Doe, Mary Johnson
treatment	a, b
result	16, 3, 2, 11, 1 (and possibly —)

Observations?

People? ***Multiple options***

Treatments?

Reorganize into a **standard form** ("**tidy**"):

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Make values, variables and observations *more clear*:

Dataset contains **one table** with **18 values** across **three variables** and **six observations**

H. Wickham, 2014

Messy vs. Tidy datasets

Reorganize into a **standard form** ("**tidy**"):

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Make values, variables and observations *more clear*:

Dataset contains **one table** with **18 values** across **three variables** and **six observations**

Tidy data

1. Each **variable** forms a column.
2. Each **observation** forms a row.
3. Each **type of observational unit** forms a table.

"Tidy data is a standard way of mapping the meaning of a dataset to its structure. A dataset is **messy or tidy** depending on how rows, columns and tables are matched up with observations, variables and types"

H. Wickham, 2014

Messy vs. Tidy datasets

Reorganize into a **standard form** ("**tidy**"):

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Make values, variables and observations *more clear*:

Dataset contains **one table** with **18 values** across **three variables** and **six observations**

Tidy data

1. Each **variable** forms a column.
2. Each **observation** forms a row.
3. Each **type of observational unit** forms a table.

Messy (non-tidy):

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

H. Wickham, 2014

Messy vs. Tidy datasets

Reorganize into a **standard form** ("**tidy**"):

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Make values, variables and observations *more clear*:

Dataset contains **one table** with **18 values** across **three variables** and **six observations**

Tidy data

1. Each **variable** forms a column.
2. Each **observation** forms a row.
3. Each **type of observational unit** forms a table.

Messy (non-tidy):

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

H. Wickham, 2014

Messy vs. Tidy datasets

Reorganize into a **standard form** ("**tidy**"):

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

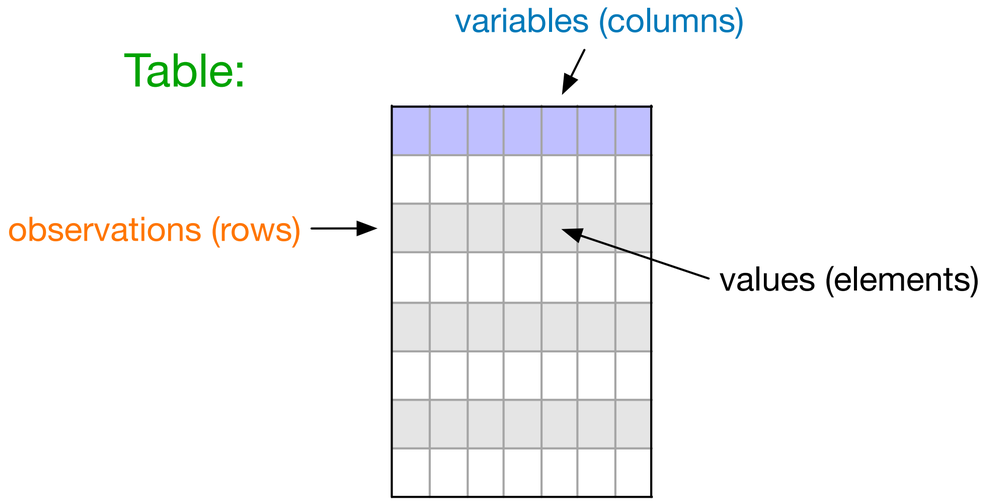
Make values, variables and observations *more clear*:
Dataset contains **one table** with **18 values** across **three variables** and **six observations**

Tidy data

1. Each **variable** forms a column.
2. Each **observation** forms a row.
3. Each **type of observational unit** forms a table.

Tidy:

Table:



(data may not be *stored* in this format)

H. Wickham, 2014

Advantages of Tidy data

Tidy data

1. Each **variable** forms a column.
2. Each **observation** forms a row.
3. Each **type of observational unit** forms a table.

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

By ensuring that every variable is in its own column, it becomes easy and consistent to refer to all the values of a variable. In messy data, different variables may need to be selected in different ways, making code less consistent and more error-prone

Ability to work row-wise: each observation self-contained
Ability to work column-wise: each variable self-contained
→Makes vectorized subroutines straightforward to use

Can explicitly label every variable

Adding new variables (feature engineering) is just appending columns. Code previously written for the dataset is more likely to work without modification with new columns added

Statistical modeling is often more straightforward

Advantages of Tidy data

By ensuring that every variable is in its own column, it becomes easy and consistent to refer to all the values of a variable. In messy data, different variables may need to be selected in different ways, making code less consistent and more error-prone

Ability to work row-wise: each observation self-contained
Ability to work column-wise: each variable self-contained
→Makes vectorized subroutines straightforward to use

Can explicitly label every variable

Adding new variables (feature engineering) is just appending columns. Code previously written for the dataset is more likely to work without modification with new columns added

Statistical modeling is often more straightforward

H. Wickham, 2014

Advantages of Tidy data

By ensuring that every variable is in its own column, it becomes easy and consistent to refer to all the values of a variable. In messy data, different variables may need to be selected in different ways, making code less consistent and more error-prone

Ability to work row-wise: each observation self-contained
Ability to work column-wise: each variable self-contained
→Makes vectorized subroutines straightforward to use

Can explicitly label every variable

Adding new variables (feature engineering) is just appending columns. Code previously written for the dataset is more likely to work without modification with new columns added

Statistical modeling is often more straightforward

Data manipulation operations are made easier when there is a consistent way to refer to variables. Tidy data provides this because each variable resides in its own column

The four fundamental verbs of data manipulation:

- Filter: subsetting or removing observations based on some condition.
- Transform: adding or modifying variables. These modifications can involve either a single variable (e.g., log-transform), or multiple variables (e.g., computing density from weight and volume).
- Aggregate: collapsing multiple values into a single value (e.g., by summing or taking means).
- Sort: changing the order of observations

H. Wickham, 2014

Advantages of Tidy data

By ensuring that every variable is in its own column, it becomes easy and consistent to refer to all the values of a variable. In messy data, different variables may need to be selected in different ways, making code less consistent and more error-prone

Ability to work row-wise: each observation self-contained
Ability to work column-wise: each variable self-contained
→Makes vectorized subroutines straightforward to use

Can explicitly label every variable

Adding new variables (feature engineering) is just appending columns. Code previously written for the dataset is more likely to work without modification with new columns added

Statistical modeling is often more straightforward

Messy

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

Tidy

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Advantages of Tidy data

By ensuring that every variable is in its own column, it becomes easy and consistent to refer to all the values of a variable. In messy data, different variables may need to be selected in different ways, making code less consistent and more error-prone

Ability to work row-wise: each observation self-contained
Ability to work column-wise: each variable self-contained
→Makes vectorized subroutines straightforward to use

Can explicitly label every variable

Adding new variables (feature engineering) is just appending columns. Code previously written for the dataset is more likely to work without modification with new columns added

Statistical modeling is often more straightforward

Messy

	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

???

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

Tidy

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Disadvantages of Tidy data

Primary **disadvantage** is **space efficiency**

Files are generally **bigger** with more **redundancy**

Tidy

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Messy

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

→ Larger table makes **data entry** more time-consuming and error prone

→ Tidy tables are less suitable for **presentation**

Disadvantages of Tidy data

Primary **disadvantage** is **space efficiency**

Files are generally **bigger** with more **redundancy**

Tidy

person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

Messy

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

→ Larger table makes data entry more time-consuming and error prone

→ Tidy tables are less suitable for presentation

However

Disk space is cheap, writing code is not

Tidy data are intended for analysis and archiving, not presentation

- "De-tidying" a dataset, for example to make a more compact table for presentation, is often more straightforward than tidying the original messy table.

In fact, many of the design choices that lead to messy data are due to formatting data for presentation

Messy data intended for presentation

Example: Pew Survey on income and religion

Messy

religion	<$10k	$10–20k	$20–30k	$30–40k	$40–50k	$50–75k
Agnostic	27	34	60	81	76	137
Atheist	12	27	37	52	35	70
Buddhist	27	21	30	34	33	58
Catholic	418	617	732	670	638	1116
Don't know/refused	15	14	15	11	10	35
Evangelical Prot	575	869	1064	982	881	1486
Hindu	1	9	7	9	11	34
Historically Black Prot	228	244	236	238	197	223
Jehovah's Witness	20	27	24	24	21	30
Jewish	19	19	25	25	30	95

Tidy (first few rows)

religion	income	freq
Agnostic	<$10k	27
Agnostic	$10–20k	34
Agnostic	$20–30k	60
Agnostic	$30–40k	81
Agnostic	$40–50k	76
Agnostic	$50–75k	137
Agnostic	$75–100k	122
Agnostic	$100–150k	109
Agnostic	>150k	84
Agnostic	Don't know/refused	96

Presentation:
easy to read across rows to compare within a single group (religion)
easy to read across columns to compare within a single group (income)

Can explicitly label every variable

H. Wickham, 2014

Messy data intended for presentation

Example: Pew Survey on income and religion

Messy

religion	<$10k	$10–20k	$20–30k	$30–40k	$40–50k	$50–75k
Agnostic	27	34	60	81	76	137
Atheist	12	27	37	52	35	70
Buddhist	27	21	30	34	33	58
Catholic	418	617	732	670	638	1116
Don't know/refused	15	14	15	11	10	35
Evangelical Prot	575	869	1064	982	881	1486
Hindu	1	9	7	9	11	34
Historically Black Prot	228	244	236	238	197	223
Jehovah's Witness	20	27	24	24	21	30
Jewish	19	19	25	25	30	95

Tidy (first few rows)

religion	income	freq
Agnostic	<$10k	27
Agnostic	$10–20k	34
Agnostic	$20–30k	60
Agnostic	$30–40k	81
Agnostic	$40–50k	76
Agnostic	$50–75k	137
Agnostic	$75–100k	122
Agnostic	$100–150k	109
Agnostic	>150k	84
Agnostic	Don't know/refused	96

Presentation:
easy to read across rows to compare within a single group (religion)
easy to read across columns to compare within a single group (income)

Non-tidy:
a variable (income) is across columns

Can explicitly label every variable

H. Wickham, 2014

Tidying messy datasets

In *tidy data*:

1. Each variable forms a column.
2. Each observation forms a row.
3. Each type of observational unit forms a table.

Most common "problems" with messy datasets

- Column headers are values, not variable names.
- Multiple variables are stored in one column.
- Variables are stored in both rows and columns.
- Multiple types of observational units are stored in the same table.
- A single observational unit is stored in multiple tables.

H. Wickham, 2014

Tidying messy datasets

Example: Billboard song charts
measurements over time

- Column headers are values, not variable names

Messy (first few rows)

75 columns

year	artist	track	time	date.entered	wk1	wk2	wk3
2000	2 Pac	Baby Don't Cry	4:22	2000-02-26	87	82	72
2000	2Ge+her	The Hardest Part Of ...	3:15	2000-09-02	91	87	92
2000	3 Doors Down	Kryptonite	3:53	2000-04-08	81	70	68
2000	98^0	Give Me Just One Nig...	3:24	2000-08-19	51	39	34
2000	A*Teens	Dancing Queen	3:44	2000-07-08	97	97	96
2000	Aaliyah	I Don't Wanna	4:15	2000-01-29	84	62	51
2000	Aaliyah	Try Again	4:03	2000-03-18	59	53	38
2000	Adams, Yolanda	Open My Heart	5:30	2000-08-26	76	76	74

H. Wickham, 2014

Tidying messy datasets

Example: Billboard song charts
measurements over time

- Column headers are values, not variable names

Messy (first few rows)

75 columns

year	artist	track	time	date.entered	wk1	wk2	wk3
2000	2 Pac	Baby Don't Cry	4:22	2000-02-26	87	82	72
2000	2Ge+her	The Hardest Part Of ...	3:15	2000-09-02	91	87	92
2000	3 Doors Down	Kryptonite	3:53	2000-04-08	81	70	68
2000	98^0	Give Me Just One Nig...	3:24	2000-08-19	51	39	34
2000	A*Teens	Dancing Queen	3:44	2000-07-08	97	97	96
2000	Aaliyah	I Don't Wanna	4:15	2000-01-29	84	62	51
2000	Aaliyah	Try Again	4:03	2000-03-18	59	53	38
2000	Adams, Yolanda	Open My Heart	5:30	2000-08-26	76	76	74

week variable created from **wk1**, **wk2**, …
date computed from **data.entered** & **week**

Tidy (first few rows)

year	artist	time	track	date	week	rank
2000	2 Pac	4:22	Baby Don't Cry	2000-02-26	1	87
2000	2 Pac	4:22	Baby Don't Cry	2000-03-04	2	82
2000	2 Pac	4:22	Baby Don't Cry	2000-03-11	3	72
2000	2 Pac	4:22	Baby Don't Cry	2000-03-18	4	77
2000	2 Pac	4:22	Baby Don't Cry	2000-03-25	5	87
2000	2 Pac	4:22	Baby Don't Cry	2000-04-01	6	94
2000	2 Pac	4:22	Baby Don't Cry	2000-04-08	7	99
2000	2Ge+her	3:15	The Hardest Part Of ...	2000-09-02	1	91
2000	2Ge+her	3:15	The Hardest Part Of ...	2000-09-09	2	87
2000	2Ge+her	3:15	The Hardest Part Of ...	2000-09-16	3	92
2000	3 Doors Down	3:53	Kryptonite	2000-04-08	1	81
2000	3 Doors Down	3:53	Kryptonite	2000-04-15	2	70
2000	3 Doors Down	3:53	Kryptonite	2000-04-22	3	68
2000	3 Doors Down	3:53	Kryptonite	2000-04-29	4	67
2000	3 Doors Down	3:53	Kryptonite	2000-05-06	5	66

Can explicitly label every variable

H. Wickham, 2014

Tidying messy datasets

Example: Tuberculosis cases

- Column headers are values, not variable names.
- Multiple variables are stored in one column.

Messy

country	year	m014	m1524	m2534	m3544	m4554	m5564	m65	mu	f014
AD	2000	0	0	1	0	0	0	0	—	—
AE	2000	2	4	4	6	5	12	10	—	3
AF	2000	52	228	183	149	129	94	80	—	93
AG	2000	0	0	0	0	0	0	1	—	1
AL	2000	2	19	21	14	24	19	16	—	3
AM	2000	2	152	130	131	63	26	21	—	1
AN	2000	0	0	1	2	0	0	0	—	0
AO	2000	186	999	1003	912	482	312	194	—	247
AR	2000	97	278	594	402	419	368	330	—	121
AS	2000	—	—	—	—	1	1	—	—	—

H. Wickham, 2014

Tidying messy datasets

Example: Tuberculosis cases

- Column headers are values, not variable names.
- Multiple variables are stored in one column.

Messy

country	year	m014	m1524	m2534	m3544	m4554	m5564	m65	mu	f014
AD	2000	0	0	1	0	0	0	0	—	—
AE	2000	2	4	4	6	5	12	10	—	3
AF	2000	52	228	183	149	129	94	80	—	93
AG	2000	0	0	0	0	0	0	1	—	1
AL	2000	2	19	21	14	24	19	16	—	3
AM	2000	2	152	130	131	63	26	21	—	1
AN	2000	0	0	1	2	0	0	0	—	0
AO	2000	186	999	1003	912	482	312	194	—	247
AR	2000	97	278	594	402	419	368	330	—	121
AS	2000	—	—	—	—	1	1	—	—	—

m014: males, age 0-14, **f3544**: females, age 35-44, etc.
(many columns omitted)

H. Wickham, 2014

Tidying messy datasets

Example: Tuberculosis cases

- Column headers are values, not variable names.
- Multiple variables are stored in one column.

Messy

country	year	m014	m1524	m2534	m3544	m4554	m5564	m65	mu	f014
AD	2000	0	0	1	0	0	0	0	—	—
AE	2000	2	4	4	6	5	12	10	—	3
AF	2000	52	228	183	149	129	94	80	—	93
AG	2000	0	0	0	0	0	0	1	—	1
AL	2000	2	19	21	14	24	19	16	—	3
AM	2000	2	152	130	131	63	26	21	—	1
AN	2000	0	0	1	2	0	0	0	—	0
AO	2000	186	999	1003	912	482	312	194	—	247
AR	2000	97	278	594	402	419	368	330	—	121
AS	2000	—	—	—	—	1	1	—	—	—

m014: males, age 0-14, **f3544**: females, age 35-44, etc.
(many columns omitted)

Partially tidy

country	year	column	cases
AD	2000	m014	0
AD	2000	m1524	0
AD	2000	m2534	1
AD	2000	m3544	0
AD	2000	m4554	0
AD	2000	m5564	0
AD	2000	m65	0
AE	2000	m014	2
AE	2000	m1524	4
AE	2000	m2534	4
AE	2000	m3544	6
AE	2000	m4554	5
AE	2000	m5564	12
AE	2000	m65	10
AE	2000	f014	3

(a) Molten data

Tidy

country	year	sex	age	cases
AD	2000	m	0–14	0
AD	2000	m	15–24	0
AD	2000	m	25–34	1
AD	2000	m	35–44	0
AD	2000	m	45–54	0
AD	2000	m	55–64	0
AD	2000	m	65+	0
AE	2000	m	0–14	2
AE	2000	m	15–24	4
AE	2000	m	25–34	4
AE	2000	m	35–44	6
AE	2000	m	45–54	5
AE	2000	m	55–64	12
AE	2000	m	65+	10
AE	2000	f	0-14	3

(b) Tidy data

H. Wickham, 2014

Tidying messy datasets

Example: Tuberculosis cases

Tidying these data **fixes another problem**

H. Wickham, 2014

Tidying messy datasets

Example: Tuberculosis cases

Tidying these data **fixes another problem**

Messy

country	year	m014	m1524	m2534	m3544	m4554	m5564	m65	mu	f014
AD	2000	0	0	1	0	0	0	0	—	—
AE	2000	2	4	4	6	5	12	10	—	3
AF	2000	52	228	183	149	129	94	80	—	93
AG	2000	0	0	0	0	0	0	1	—	1
AL	2000	2	19	21	14	24	19	16	—	3
AM	2000	2	152	130	131	63	26	21	—	1
AN	2000	0	0	1	2	0	0	0	—	0
AO	2000	186	999	1003	912	482	312	194	—	247
AR	2000	97	278	594	402	419	368	330	—	121
AS	2000	—	—	—	—	1	1	—	—	—

(Number of cases)

H. Wickham, 2014

Tidying messy datasets

Example: Tuberculosis cases

We really want to know the **rates of TB**, not number of cases. This requires knowing the **population** for each group

Tidying these data **fixes another problem**

Messy

country	year	m014	m1524	m2534	m3544	m4554	m5564	m65	mu	f014
AD	2000	0	0	1	0	0	0	0	—	—
AE	2000	2	4	4	6	5	12	10	—	3
AF	2000	52	228	183	149	129	94	80	—	93
AG	2000	0	0	0	0	0	0	1	—	1
AL	2000	2	19	21	14	24	19	16	—	3
AM	2000	2	152	130	131	63	26	21	—	1
AN	2000	0	0	1	2	0	0	0	—	0
AO	2000	186	999	1003	912	482	312	194	—	247
AR	2000	97	278	594	402	419	368	330	—	121
AS	2000	—	—	—	—	1	1	—	—	—

(Number of cases)

H. Wickham, 2014

Tidying messy datasets

Example: Tuberculosis cases

We really want to know the **rates of TB**, not number of cases. This requires knowing the **population** for each group

Tidying these data **fixes another problem**

Messy

country	year	m014	m1524	m2534	m3544	m4554	m5564	m65	mu	f014
AD	2000	0	0	1	0	0	0	0	—	—
AE	2000	2	4	4	6	5	12	10	—	3
AF	2000	52	228	183	149	129	94	80	—	93
AG	2000	0	0	0	0	0	0	1	—	1
AL	2000	2	19	21	14	24	19	16	—	3
AM	2000	2	152	130	131	63	26	21	—	1
AN	2000	0	0	1	2	0	0	0	—	0
AO	2000	186	999	1003	912	482	312	194	—	247
AR	2000	97	278	594	402	419	368	330	—	121
AS	2000	—	—	—	—	1	1	—	—	—

(Number of cases)

Here we would need a second table to store population, which makes it harder to correctly match populations and counts to get rates

H. Wickham, 2014

Tidying messy datasets

Example: Tuberculosis cases

We really want to know the **rates of TB**, not number of cases. This requires knowing the **population** for each group

Tidying these data **fixes another problem**

Messy

country	year	m014	m1524	m2534	m3544	m4554	m5564	m65	mu	f014
AD	2000	0	0	1	0	0	0	0	—	—
AE	2000	2	4	4	6	5	12	10	—	3
AF	2000	52	228	183	149	129	94	80	—	93
AG	2000	0	0	0	0	0	0	1	—	1
AL	2000	2	19	21	14	24	19	16	—	3
AM	2000	2	152	130	131	63	26	21	—	1
AN	2000	0	0	1	2	0	0	0	—	0
AO	2000	186	999	1003	912	482	312	194	—	247
AR	2000	97	278	594	402	419	368	330	—	121
AS	2000	—	—	—	—	1	1	—	—	—

(Number of cases)

Here we would need a second table to store population, which makes it harder to correctly match populations and counts to get rates

Tidy

country	year	sex	age	cases	population	rate
AD	2000	m	0–14	0	⋮	⋮
AD	2000	m	15–24	0	⋮	⋮
AD	2000	m	25–34	1	⋮	⋮
AD	2000	m	35–44	0	⋮	⋮
AD	2000	m	45–54	0	⋮	⋮
AD	2000	m	55–64	0	⋮	⋮
AD	2000	m	65+	0	⋮	⋮
AE	2000	m	0–14	2	⋮	⋮
AE	2000	m	15–24	4	⋮	⋮
AE	2000	m	25–34	4	⋮	⋮
AE	2000	m	35–44	6	⋮	⋮
AE	2000	m	45–54	5	⋮	⋮
AE	2000	m	55–64	12	⋮	⋮
AE	2000	m	65+	10	⋮	⋮
AE	2000	f	0-14	3	⋮	⋮

(b) Tidy data

Here we can just append columns

H. Wickham, 2014

Tidying messy datasets

Example: daily weather data

- Variables are stored in both rows and columns

(most complicated form of messy data)

Messy

id	year	month	element	d1	d2	d3	d4	d5	d6	d7	d8
MX17004	2010	1	tmax	—	—	—	—	—	—	—	—
MX17004	2010	1	tmin	—	—	—	—	—	—	—	—
MX17004	2010	2	tmax	—	27.3	24.1	—	—	—	—	—
MX17004	2010	2	tmin	—	14.4	14.4	—	—	—	—	—
MX17004	2010	3	tmax	—	—	—	—	32.1	—	—	—
MX17004	2010	3	tmin	—	—	—	—	14.2	—	—	—
MX17004	2010	4	tmax	—	—	—	—	—	—	—	—
MX17004	2010	4	tmin	—	—	—	—	—	—	—	—
MX17004	2010	5	tmax	—	—	—	—	—	—	—	—
MX17004	2010	5	tmin	—	—	—	—	—	—	—	—

H. Wickham, 2014

Tidying messy datasets

Example: daily weather data

- Variables are stored in both rows and columns

(most complicated form of messy data)

Messy

id	year	month	element	d1	d2	d3	d4	d5	d6	d7	d8
MX17004	2010	1	tmax	—	—	—	—	—	—	—	—
MX17004	2010	1	tmin	—	—	—	—	—	—	—	—
MX17004	2010	2	tmax	—	27.3	24.1	—	—	—	—	—
MX17004	2010	2	tmin	—	14.4	14.4	—	—	—	—	—
MX17004	2010	3	tmax	—	—	—	—	32.1	—	—	—
MX17004	2010	3	tmin	—	—	—	—	14.2	—	—	—
MX17004	2010	4	tmax	—	—	—	—	—	—	—	—
MX17004	2010	4	tmin	—	—	—	—	—	—	—	—
MX17004	2010	5	tmax	—	—	—	—	—	—	—	—
MX17004	2010	5	tmin	—	—	—	—	—	—	—	—

Partially tidy

id	date	element	value
MX17004	2010-01-30	tmax	27.8
MX17004	2010-01-30	tmin	14.5
MX17004	2010-02-02	tmax	27.3
MX17004	2010-02-02	tmin	14.4
MX17004	2010-02-03	tmax	24.1
MX17004	2010-02-03	tmin	14.4
MX17004	2010-02-11	tmax	29.7
MX17004	2010-02-11	tmin	13.4
MX17004	2010-02-23	tmax	29.9
MX17004	2010-02-23	tmin	10.7

(a) Molten data

Tidy

id	date	tmax	tmin
MX17004	2010-01-30	27.8	14.5
MX17004	2010-02-02	27.3	14.4
MX17004	2010-02-03	24.1	14.4
MX17004	2010-02-11	29.7	13.4
MX17004	2010-02-23	29.9	10.7
MX17004	2010-03-05	32.1	14.2
MX17004	2010-03-10	34.5	16.8
MX17004	2010-03-16	31.1	17.6
MX17004	2010-04-27	36.3	16.7
MX17004	2010-05-27	33.2	18.2

(b) Tidy data

H. Wickham, 2014

Tidying messy datasets

Example: daily weather data

- Variables are stored in both rows and columns

(most complicated form of messy data)

Messy

id	year	month	element	d1	d2	d3	d4	d5	d6	d7	d8
MX17004	2010	1	tmax	—	—	—	—	—	—	—	—
MX17004	2010	1	tmin	—	—	—	—	—	—	—	—
MX17004	2010	2	tmax	—	27.3	24.1	—	—	—	—	—
MX17004	2010	2	tmin	—	14.4	14.4	—	—	—	—	—
MX17004	2010	3	tmax	—	—	—	—	32.1	—	—	—
MX17004	2010	3	tmin	—	—	—	—	14.2	—	—	—
MX17004	2010	4	tmax	—	—	—	—	—	—	—	—
MX17004	2010	4	tmin	—	—	—	—	—	—	—	—
MX17004	2010	5	tmax	—	—	—	—	—	—	—	—
MX17004	2010	5	tmin	—	—	—	—	—	—	—	—

Missing values (—) are explicit

Partially tidy

id	date	element	value
MX17004	2010-01-30	tmax	27.8
MX17004	2010-01-30	tmin	14.5
MX17004	2010-02-02	tmax	27.3
MX17004	2010-02-02	tmin	14.4
MX17004	2010-02-03	tmax	24.1
MX17004	2010-02-03	tmin	14.4
MX17004	2010-02-11	tmax	29.7
MX17004	2010-02-11	tmin	13.4
MX17004	2010-02-23	tmax	29.9
MX17004	2010-02-23	tmin	10.7

(a) Molten data

Tidy

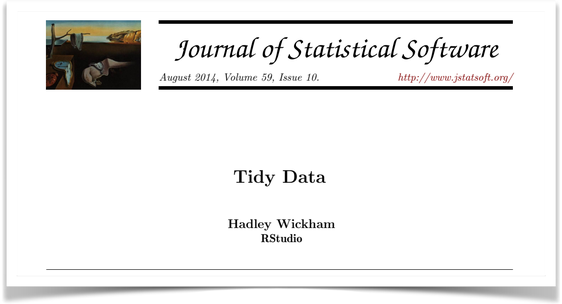
id	date	tmax	tmin
MX17004	2010-01-30	27.8	14.5
MX17004	2010-02-02	27.3	14.4
MX17004	2010-02-03	24.1	14.4
MX17004	2010-02-11	29.7	13.4
MX17004	2010-02-23	29.9	10.7
MX17004	2010-03-05	32.1	14.2
MX17004	2010-03-10	34.5	16.8
MX17004	2010-03-16	31.1	17.6
MX17004	2010-04-27	36.3	16.7
MX17004	2010-05-27	33.2	18.2

(b) Tidy data

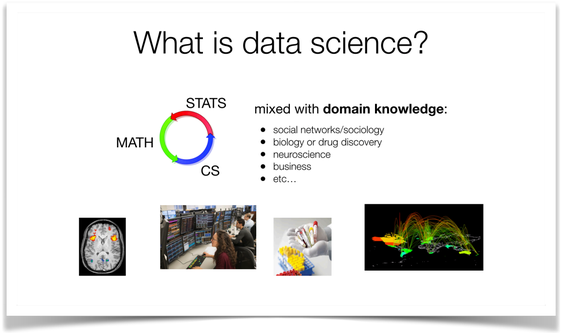
Missing values (—) are implicit
(but can be reconstructed from date variable)

H. Wickham, 2014

Tidy data—Discussion



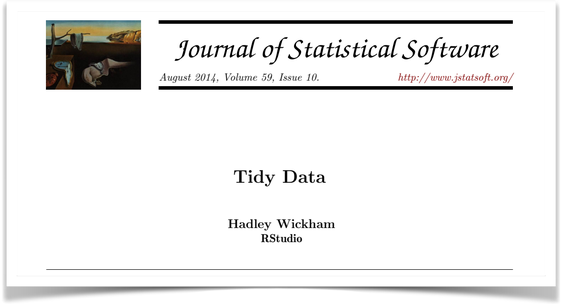
"Data cleaning is an important problem, but it is an uncommon subject of study in statistics."

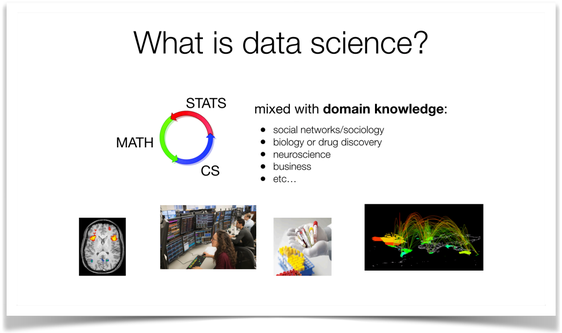


Recall - Lecture 1

–Hadley Wickham (2014) [emphasis added]

Tidy data—Discussion





Recall - Lecture 1

"Data cleaning is an important problem, but it is an uncommon subject of study in statistics."

“Surprisingly, I have found few principles to guide the design of tidy data, which acknowledge both statistical and cognitive factors. To date, my work has been driven by my experience doing data analysis, my knowledge of relational database design, and my own rumination on the tools of data analysis. The human factors, user-centered design, and human-computer interaction communities may be able to add to this conversation, but the design of data and tools to work with it has not been an active research topic in those fields. In the future, I hope to use methodologies from these fields (user-testing, ethnography, talk-aloud protocols) to improve our understanding of the cognitive side of data analysis, and to further improve our ability to design appropriate tools.”

–Hadley Wickham (2014) [emphasis added]