DSA/ISE 5103 Intelligent Data Analytics

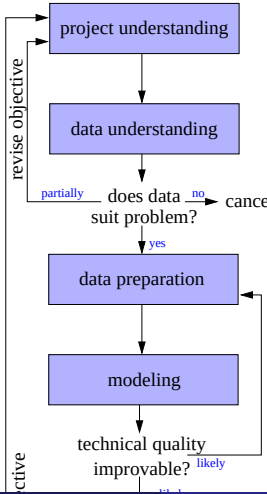
Data Preparation

Charles Nicholson, Ph.D.
cnicholson@ou.edu

University of Oklahoma
Gallogly College of Engineering
School of Industrial and Systems Engineering

Outline

data preparation



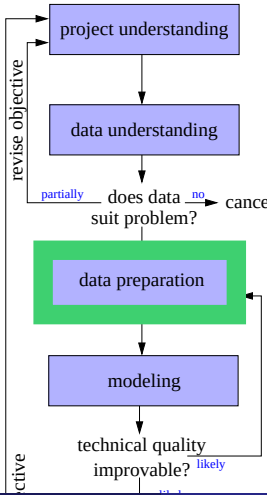
What exactly is the problem, the expected benefit?
How would a solution look like?
What is known about the domain?

What data do we have available?
Is the data relevant to the problem?
Is it valid? Does it reflect our expectations?
Is the data quality, quantity, recency sufficient?

Which data should we concentrate on?
How is the data best transformed for modeling?
How may we increase the data quality?

What kind of model architecture suits the problem best?
What is the best technique/method to get the model?
How good does the model perform technically?

data preparation



What exactly is the problem, the expected benefit?
How would a solution look like?
What is known about the domain?

What data do we have available?
Is the data relevant to the problem?
Is it valid? Does it reflect our expectations?
Is the data quality, quantity, recency sufficient?

Which data should we concentrate on?
How is the data best transformed for modeling?
How may we increase the data quality?

What kind of model architecture suits the problem best?
What is the best technique/method to get the model?
How good does the model perform technically?

data understanding vs preparation

Data understanding provides visualizations and metadata:

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality
- treat outliers

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality
- treat outliers
- resolve missing values

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality
- treat outliers
- resolve missing values
- transform data

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality
- treat outliers
- resolve missing values
- transform data
- feature engineering

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality
- treat outliers
- resolve missing values
- transform data
- feature engineering
  - feature extraction

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality
- treat outliers
- resolve missing values
- transform data
- feature engineering
  - feature extraction
  - feature construction

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality
- treat outliers
- resolve missing values
- transform data
- feature engineering
  - feature extraction
  - feature construction
  - feature selection

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality
- treat outliers
- resolve missing values
- transform data
- feature engineering
  - feature extraction
  - feature construction
  - feature selection

data understanding vs preparation

Data understanding provides visualizations and metadata:

- level of quality
- character of attributes (distributions, etc.)
- dependencies between attributes
- the existence and nature of missing values
- outliers

$\longrightarrow$ for **data preparation**:

- improve data quality
- treat outliers
- resolve missing values
- transform data
- feature engineering
  - feature extraction
  - feature construction
  - feature selection

which may alter **data understanding**

data scrubbing

Data cleansing or data scrubbing refers to detecting, correcting and/or removing

- inaccurate
- incorrect
- incomplete

records from a data set.

improve basic data quality

Some examples:

- Turn all characters into capital letters
- Remove spaces and non-printing characters
- Split fields that carry mixed information into two separate attributes, e.g. *"Chocolate, 100g"* into *"Chocolate"* and *"100.0"*. This is known as field overloading.
- Use spell-checker to normalize spelling in free text entries.
- Replace abbreviations by their long form
- Normalize the writing of addresses and names

improve basic data quality

Some examples:

- Turn all characters into capital letters
- Remove spaces and non-printing characters
- Split fields that carry mixed information into two separate attributes, e.g. *"Chocolate, 100g"* into *"Chocolate"* and *"100.0"*. This is known as field overloading.
- Use spell-checker to normalize spelling in free text entries.
- Replace abbreviations by their long form
- Normalize the writing of addresses and names

improve basic data quality

Some examples:

- Turn all characters into capital letters
- Remove spaces and non-printing characters
- Split fields that carry mixed information into two separate attributes, e.g. *"Chocolate, 100g"* into *"Chocolate"* and *"100.0"*. This is known as field overloading.
- Use spell-checker to normalize spelling in free text entries.
- Replace abbreviations by their long form
- Normalize the writing of addresses and names

improve basic data quality

Some examples:

- Turn all characters into capital letters
- Remove spaces and non-printing characters
- Split fields that carry mixed information into two separate attributes, e.g. *"Chocolate, 100g"* into *"Chocolate"* and *"100.0"*. This is known as field overloading.
- Use spell-checker to normalize spelling in free text entries.
- Replace abbreviations by their long form
- Normalize the writing of addresses and names

improve basic data quality

Some examples:

- Turn all characters into capital letters
- Remove spaces and non-printing characters
- Split fields that carry mixed information into two separate attributes, e.g. *"Chocolate, 100g"* into *"Chocolate"* and *"100.0"*. This is known as field overloading.
- Use spell-checker to normalize spelling in free text entries.
- Replace abbreviations by their long form
- Normalize the writing of addresses and names

improve basic data quality

Some examples:

- Turn all characters into capital letters
- Remove spaces and non-printing characters
- Split fields that carry mixed information into two separate attributes, e.g. *"Chocolate, 100g"* into *"Chocolate"* and *"100.0"*. This is known as field overloading.
- Use spell-checker to normalize spelling in free text entries.
- Replace abbreviations by their long form
- Normalize the writing of addresses and names

improve data quality

Ensure the computer is storing the values in *appropriate* system data types:

- e.g., computers store “dates” in several ways
  - as text
  - number of days since Jan, 1, 1960
  - number of seconds since Jan 1, 1960

improve data quality

- e.g., are your “numeric” fields really numeric?
  - in the `Soybean` data (from the `mlbench` package), the value for `hail` is not numeric

Soybean

683 observations of 36 variables

	Class	date	plant.stand	precip	temp	hail	crop.hist	area.dam	sever	seed.tmt	germ	p
1	diaporthe-stem-canker	6	0	2	1	0	1	1	1	0	0	1
2	diaporthe-stem-canker	4	0	2	1	0	2	0	2	1	1	1
3	diaporthe-stem-canker	3	0	2	1	0	1	0	2	1	2	1
4	diaporthe-stem-canker	3	0	2	1	0	1	0	2	0	1	1
5	diaporthe-stem-canker	6	0	2	1	0	2	0	1	0	2	1
6	diaporthe-stem-canker	5	0	2	1	0	3	0	1	0	1	1
7	diaporthe-stem-canker	5	0	2	1	0	2	0	1	1	0	1

```
> mean(Soybean$hail)
[1] NA

Warning message:
In mean.default(Soybean$hail) :
argument is not numeric or logical: returning NA
```

FYI – the `str` command is a useful way to look at the structure of an R object.

None of the "numeric" looking data in the `Soybean` data is actually stored as numeric values!

```
> str(Soybean)
'data.frame': 683 obs. of  36 variables:
 $ Class       : Factor w/ 19 levels "2-4-d-injury",..: 11 11 11 11 11 11 11 11 11 11 ...
 $ date        : Factor w/ 7 levels "0","1","2","3",..: 7 5 4 4 7 6 6 5 7 5 ...
 $ plant.stand : Ord.factor w/ 2 levels "0"<"1": 1 1 1 1 1 1 1 1 1 1 ...
 $ precip      : Ord.factor w/ 3 levels "0"<"1"<"2": 3 3 3 3 3 3 3 3 3 3 ...
 $ temp        : Ord.factor w/ 3 levels "0"<"1"<"2": 2 2 2 2 2 2 2 2 2 2 ...
 $ hail        : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 2 1 1 ...
 $ crop.hist   : Factor w/ 4 levels "0","1","2","3": 2 3 2 2 3 4 3 2 4 3 ...
 $ area.dam    : Factor w/ 4 levels "0","1","2","3": 2 1 1 1 1 1 1 1 1 1 ...}
```

digression: converting data types in R

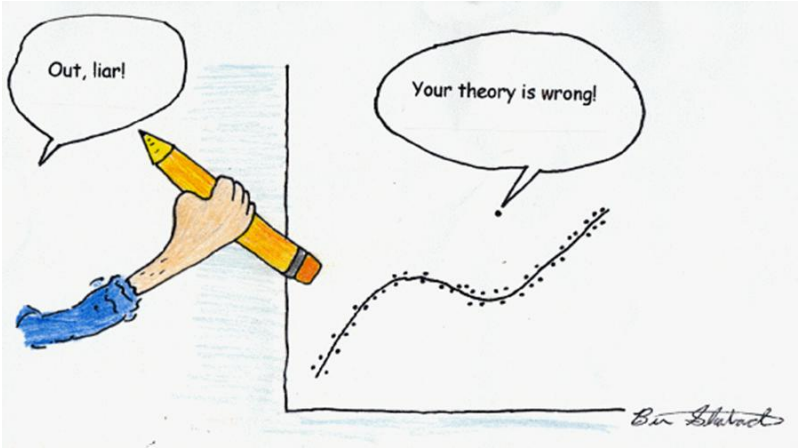
Data type conversion in R is relatively simple (assuming the values and types are compatible)

e.g., `Soybean$hail <- as.numeric(Soybean$hail)`

There are many resources online that explain type conversion in R.

e.g., `www.statmethods.net/management/typeconversion.html`

treating outliers



http://davidmlane.com/ben/cartoons.html

treating outliers

treating outliers

Knowing why an observation is an outlier is important.

- Are they mistakes? If so, fix.
- Unusual circumstances that differ from the study objectives? If so, delete them - but explain the reason clearly.
- Are they just unusual? If so, discuss why they are unusual. Does it suggest other variables that might be included in the model?
- In many analyses the outliers are the most interesting things.

treating outliers

Knowing why an observation is an outlier is important.

- Are they mistakes? If so, fix.
- Unusual circumstances that differ from the study objectives? If so, delete them - but explain the reason clearly.
- Are they just unusual? If so, discuss why they are unusual. Does it suggest other variables that might be included in the model?
- In many analyses the outliers are the most interesting things.

treating outliers

Knowing why an observation is an outlier is important.

- Are they mistakes? If so, fix.
- Unusual circumstances that differ from the study objectives? If so, delete them - but explain the reason clearly.
- Are they just unusual? If so, discuss why they are unusual. Does it suggest other variables that might be included in the model?
- In many analyses the outliers are the most interesting things.

treating outliers

Knowing why an observation is an outlier is important.

- Are they mistakes? If so, fix.
- Unusual circumstances that differ from the study objectives? If so, delete them - but explain the reason clearly.
- Are they just unusual? If so, discuss why they are unusual. Does it suggest other variables that might be included in the model?
- In many analyses the outliers are the most interesting things.

treating outliers

- It is *usually* not permissible to delete outliers automatically, without disclosure.

  NSF defined 3 types of research misconduct: **fabrication**, **falsification**, and **plagiarism**.

[1] www.nsf.gov/od/ogc/regulation.jsp

treating outliers

- It is *usually* not permissible to delete outliers automatically, without disclosure.

  NSF defined 3 types of research misconduct: **fabrication**, **falsification**, and **plagiarism**.

[1] www.nsf.gov/od/ogc/regulation.jsp

treating outliers

- It is *usually* not permissible to delete outliers automatically, without disclosure.

  NSF defined 3 types of research misconduct: **fabrication**, **falsification**, and **plagiarism**.

[1] www.nsf.gov/od/ogc/regulation.jsp

treating outliers

- It is *usually* not permissible to delete outliers automatically, without disclosure.

  NSF defined 3 types of research misconduct: **fabrication**, **falsification**, and **plagiarism**.

*Falsification is manipulating research materials, equipment, or processes or changing or **omitting data or results** such that the research is not accurately represented in the research record.* [1]

[1] www.nsf.gov/od/ogc/regulation.jsp

treating outliers

- Even if you disclose outlier deletion,
  - *en masse* deletion is usually not a good idea
  - that said, in ML, *en masse* outlier deletion might be appropriate
- Can the outliers be diminished by transformations?

treating outliers

- Even if you disclose outlier deletion,
  - *en masse* deletion is usually not a good idea
  - that said, in ML, *en masse* outlier deletion might be appropriate
- Can the outliers be diminished by transformations?

treating outliers

- Even if you disclose outlier deletion,
  - *en masse* deletion is usually not a good idea
  - that said, in ML, *en masse* outlier deletion might be appropriate
- Can the outliers be diminished by transformations?

treating outliers

- Even if you disclose outlier deletion,
  - *en masse* deletion is usually not a good idea
  - that said, in ML, *en masse* outlier deletion might be appropriate
- Can the outliers be diminished by transformations?

treating outliers

- Even if you disclose outlier deletion,
  - *en masse* deletion is usually not a good idea
  - that said, in ML, *en masse* outlier deletion might be appropriate
- Can the outliers be diminished by transformations?

treating outliers

- Even if you disclose outlier deletion,
  - *en masse* deletion is usually not a good idea
  - that said, in ML, *en masse* outlier deletion might be appropriate
- Can the outliers be diminished by transformations?

Outlier analysis and treatment requires context.

types of missing values

MCAR: it can be assumed that the missing values follow the same distribution as the observed values.

types of missing values

MCAR: it can be assumed that the missing values follow the same distribution as the observed values.

- incomplete cases can be deleted

types of missing values

MCAR: it can be assumed that the missing values follow the same distribution as the observed values.

- incomplete cases can be deleted
- or values can be *imputed*.

types of missing values

MCAR: it can be assumed that the missing values follow the same distribution as the observed values.

- incomplete cases can be deleted
- or values can be *imputed*.

MAR: the missing values might not follow the distribution of observed values.

types of missing values

MCAR: it can be assumed that the missing values follow the same distribution as the observed values.

- incomplete cases can be deleted
- or values can be *imputed*.

MAR: the missing values might not follow the distribution of observed values.

By taking the other attributes into account, it is possible to derive reasonable imputations for the missing values. (This is better than deletion)

types of missing values

MCAR: it can be assumed that the missing values follow the same distribution as the observed values.

- incomplete cases can be deleted
- or values can be *imputed*.

MAR: the missing values might not follow the distribution of observed values.

By taking the other attributes into account, it is possible to derive reasonable imputations for the missing values. (This is better than deletion)

Non-ignorable: It is difficult to provide sensible estimations for the missing values.

types of missing values

MCAR: it can be assumed that the missing values follow the same distribution as the observed values.

- incomplete cases can be deleted
- or values can be *imputed*.

MAR: the missing values might not follow the distribution of observed values.

By taking the other attributes into account, it is possible to derive reasonable imputations for the missing values. (This is better than deletion)

Non-ignorable: It is difficult to provide sensible estimations for the missing values.

- It can be difficult to distinguish MCAR, MAR, and MNAR.

types of missing values

MCAR: it can be assumed that the missing values follow the same distribution as the observed values.

- incomplete cases can be deleted
- or values can be *imputed*.

MAR: the missing values might not follow the distribution of observed values.

By taking the other attributes into account, it is possible to derive reasonable imputations for the missing values. (This is better than deletion)

Non-ignorable: It is difficult to provide sensible estimations for the missing values.

- It can be difficult to distinguish MCAR, MAR, and MNAR.
- MNAR is the most likely missing value mechanism.

missing data examples in R - quiz!

```
x<-rexp(1000)    # with exponential distribution
y<-rnorm(1000)   # with normal distribution
z<-runif(1000)   # with uniform distribution
```

missing data examples in R - quiz!

```
x<-rexp(1000)    # with exponential distribution
y<-rnorm(1000)   # with normal distribution
z<-runif(1000)   # with uniform distribution


    df<-data.frame(x,y)
    df[df$x>2.1, "y"]<-NA
```

missing data examples in R - quiz!

```
x<-rexp(1000)    # with exponential distribution
y<-rnorm(1000)   # with normal distribution
z<-runif(1000)   # with uniform distribution
```

```
df<-data.frame(x,y)
df[df$x>2.1, "y"]<-NA
```

```
df<-data.frame(x,y)
df[df$y>1.10, "y"]<-NA
```

missing data examples in R - quiz!

```
x<-rexp(1000)    # with exponential distribution
y<-rnorm(1000)   # with normal distribution
z<-runif(1000)   # with uniform distribution
```

```
df<-data.frame(x,y)
df[df$x>2.1, "y"]<-NA
```

```
df<-data.frame(x,y)
df[df$y>1.10, "y"]<-NA
```

```
df<-data.frame(x,y)
df[z>0.9, "y"]<-NA
```

missing data examples in R - quiz!

```
x<-rexp(1000)    # with exponential distribution
y<-rnorm(1000)   # with normal distribution
z<-runif(1000)   # with uniform distribution
```

```
df<-data.frame(x,y)
df[df$x>2.1, "y"]<-NA
```

```
df<-data.frame(x,y)
df[df$y>1.10, "y"]<-NA
```

```
df<-data.frame(x,y)
df[z>0.9, "y"]<-NA
```

missing data examples in R - quiz!

```
x<-rexp(1000)    # with exponential distribution
y<-rnorm(1000)   # with normal distribution
z<-runif(1000)   # with uniform distribution
```

```
df<-data.frame(x,y)
df[df$x>2.1, "y"]<-NA
```
⟵ MAR

```
df<-data.frame(x,y)
df[df$y>1.10, "y"]<-NA
```
⟵ MNAR

```
df<-data.frame(x,y)
df[z>0.9, "y"]<-NA
```
⟵ MCAR

dealing with missing values

- Deletion
- Indicators
- Imputation

dealing with missing values

- Deletion
  - Complete Case Analysis
  - Available Case Analysis
- Indicators
- Imputation

dealing with missing values

- Deletion
  - Complete Case Analysis
  - Available Case Analysis
- Indicators
- Imputation

deletion

Complete case analysis a.k.a. *listwise deletion*: analyze only cases without missing values (delete the record if any value missing)

- Advantage: simple
- Disadvantages:
  - throws away information
  - reduces statistical power (reduces sample sizes)

age	income	gender
54	93500	F
~~44~~	~~72000~~	
27	90000	F
~~26~~		~~M~~
~~47~~	~~75000~~	
31	135000	F
25	40000	M
~~39~~		~~F~~

deletion

Complete case analysis a.k.a. *listwise deletion*: analyze only cases without missing values (delete the record if any value missing)

- Advantage: simple
- Disadvantages:
  - throws away information
  - reduces statistical power (reduces sample sizes)

age	income	gender
54	93500	F
~~44~~	~~72000~~	
27	90000	F
~~26~~		~~M~~
~~47~~	~~75000~~	
31	135000	F
25	40000	M
~~39~~		~~F~~

deletion

Complete case analysis a.k.a. *listwise deletion*: analyze only cases without missing values (delete the record if any value missing)

- Advantage: simple
- Disadvantages:
  - throws away information
  - reduces statistical power (reduces sample sizes)

age	income	gender
54	93500	F
~~44~~	~~72000~~	
27	90000	F
~~26~~		~~M~~
~~47~~	~~75000~~	
31	135000	F
25	40000	M
~~39~~		~~F~~

deletion

Complete case analysis a.k.a. *listwise deletion*: analyze only cases without missing values (delete the record if any value missing)

- Advantage: simple
- Disadvantages:
  - throws away information
  - reduces statistical power (reduces sample sizes)

age	income	gender
54	93500	F
~~44~~	~~72000~~	
27	90000	F
~~26~~		~~M~~
~~47~~	~~75000~~	
31	135000	F
25	40000	M
~~39~~		~~F~~

deletion

Available Case Analysis a.k.a. *pairwise deletion*: use all cases if they have non-missing values

- Advantage: keeps as much information as possible
- Disadvantages:
  - different sample sizes
  - different analysis are based on different subsets of data

age	income	gender
54	93500	F
44	72000	
27	90000	F
26		M
47	75000	
31	135000	F
25	40000	M
39		F

deletion

Available Case Analysis a.k.a. *pairwise deletion*: use all cases if they have non-missing values

- Advantage: keeps as much information as possible
- Disadvantages:
  - different sample sizes
  - different analysis are based on different subsets of data

age	income	gender
54	93500	F
44	72000	
27	90000	F
26		M
47	75000	
31	135000	F
25	40000	M
39		F

deletion

Available Case Analysis a.k.a. *pairwise deletion*: use all cases if they have non-missing values

- Advantage: keeps as much information as possible
- Disadvantages:
  - different sample sizes
  - different analysis are based on different subsets of data

age	income	gender
54	93500	F
44	72000	
27	90000	F
26		M
47	75000	
31	135000	F
25	40000	M
39		F

deletion

Available Case Analysis a.k.a. *pairwise deletion*: use all cases if they have non-missing values

- Advantage: keeps as much information as possible
- Disadvantages:
  - different sample sizes
  - different analysis are based on different subsets of data

age	income	gender
54	93500	F
44	72000	
27	90000	F
26		M
47	75000	
31	135000	F
25	40000	M
39		F

excluding missing values from analyses in R

Arithmetic on missing values yield missing values

```
x <- c(1,2,NA,3)
mean(x) # returns NA
mean(x, na.rm=TRUE) # returns 2
```

excluding missing values from analyses in R

Arithmetic on missing values yield missing values

```
x <- c(1,2,NA,3)
mean(x) # returns NA
mean(x, na.rm=TRUE) # returns 2
```

Most modeling functions in R offer options for dealing with missing values, e.g.

Usage

```
median(x, na.rm = FALSE)
```

Arguments

`x`	an object for which a method has been defined, or a numeric vector containing the values whose median is to be computed.
`na.rm`	a logical value indicating whether `NA` values should be stripped before the computation proceeds.

excluding missing values from analyses in R

Extract only complete cases

`complete.cases()` returns a logical vector indicating which cases are complete

```
# list rows of data that have missing values
mydata[!complete.cases(mydata),]
```

`na.omit()` directly returns complete cases in data frame

```
# create new data frame without missing data
newdata <- na.omit(mydata)
```

dealing with missing values

- Deletion
- Indicators
- Imputation

indicators

The “missingness” of the data itself might be an important factor

- New binary attribute can be created to indicate whether or not a variable has/had missing data.

age	income	incomeMissing	gender	genderMissing
54	93500	0	F	0
44	72000	0		**1**
27	90000	0	F	0
26		**1**	M	0
47	75000	0		**1**
31	135000	0	F	0
25	40000	0	M	0
39		**1**	F	0

dealing with missing values

- Deletion
- Indicators
- Imputation
  - single imputation
  - multiple imputation (MI)
  - maximum likelihood (ML)

single imputation

- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to “predict” the missing value based on non-missing values

single imputation

- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to “predict” the missing value based on non-missing values

single imputation

- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to “predict” the missing value based on non-missing values

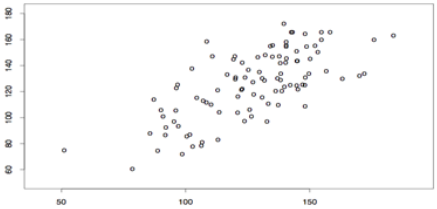
single imputation

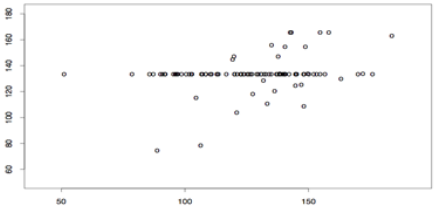
- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to “predict” the missing value based on non-missing values

single imputation

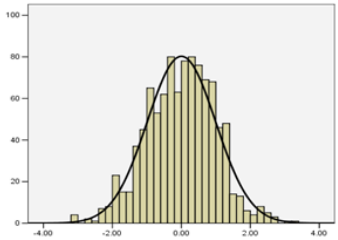
- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to “predict” the missing value based on non-missing values

example: mean imputation

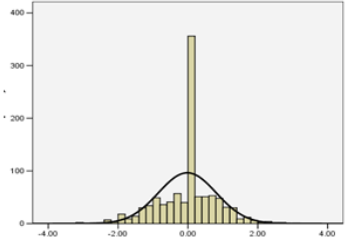




We'd like to obtain…



But instead we get…



single imputation

- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to “predict” the missing value based on non-missing values

single imputation

- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to "predict" the missing value based on non-missing values

single imputation

- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to "predict" the missing value based on non-missing values

single imputation

- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to “predict” the missing value based on non-missing values

single imputation

- single value imputation
  - missing values are replaced by an estimate (e.g. mean, median)
  - ...changes the distribution of that variable
  - ...the variance is underestimated
  - generally considered **BAD!**
- stratified imputation
  - replace missing values with conditional mean or median
- model-based imputation
  - use model to “predict” the missing value based on non-missing values

single imputation

- single value and stratified imputation
  - hotdeck (not exactly a “single value” or ”stratified” imputation method)
  - mean imputation; conditional mean imputation
- model-based imputation
  - regression imputation
  - regression with error imputation
  - predictive mean matching
  - *k*-nearest neighbor imputation

examples of imputation effects

1,000 records without missing values:

$x \sim \text{Exp}(1)$
$y \sim \text{N}(0, 0.5) + 0.5x$

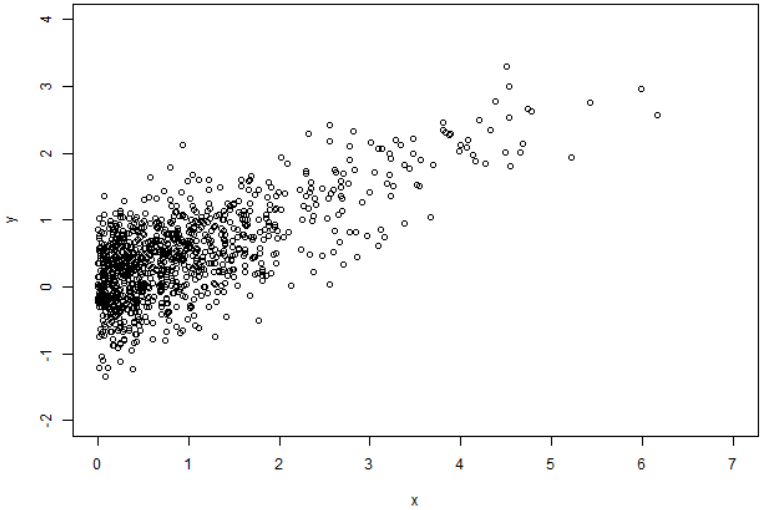
Missing Value Mechanism:

for y: MCAR, MAR, *and* MNAR

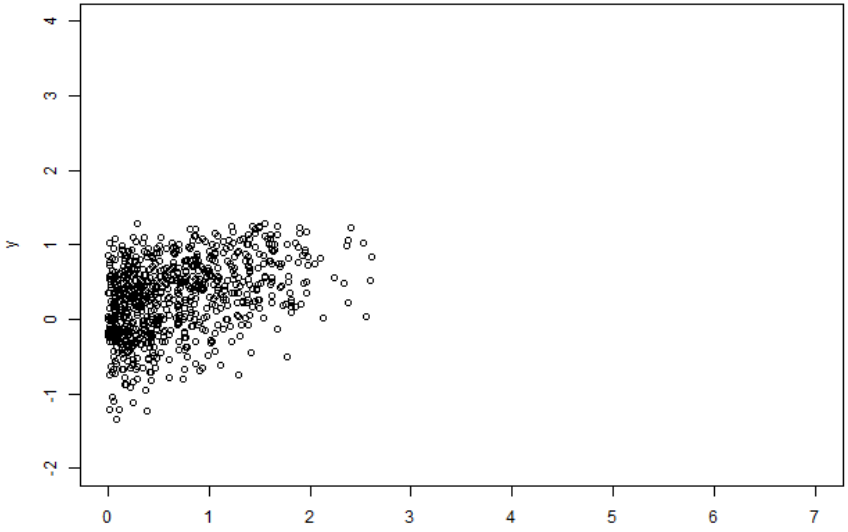
R code: data creation for example

```
#CREATE SOME FAKE DATA  (with y = f(x))
x<-rexp(1000)
y<-0.5*rnorm(1000) + 0.5*x
df<-data.frame(x,y)

# now lets create some missing values....
dfMiss <- df

beta<-runif(1000)  # not included in dataframe

dfMiss[df$y>1.30,"y"]<-NA          #MNAR
dfMiss[beta>0.90,"y"]<-NA          #MCAR
dfMiss[df$x>2.65,"y"]<-NA          #MAR
```

scatter plot with no missing data

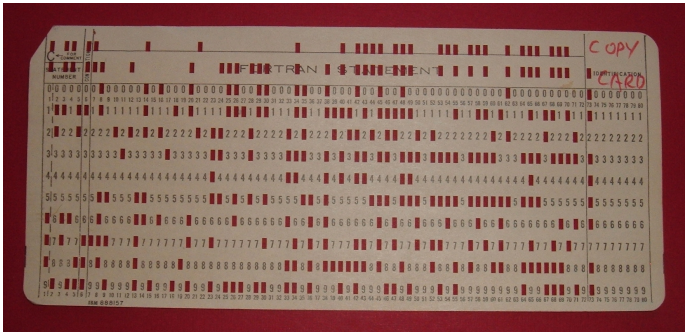


scatter plot (missing data)



hotdeck

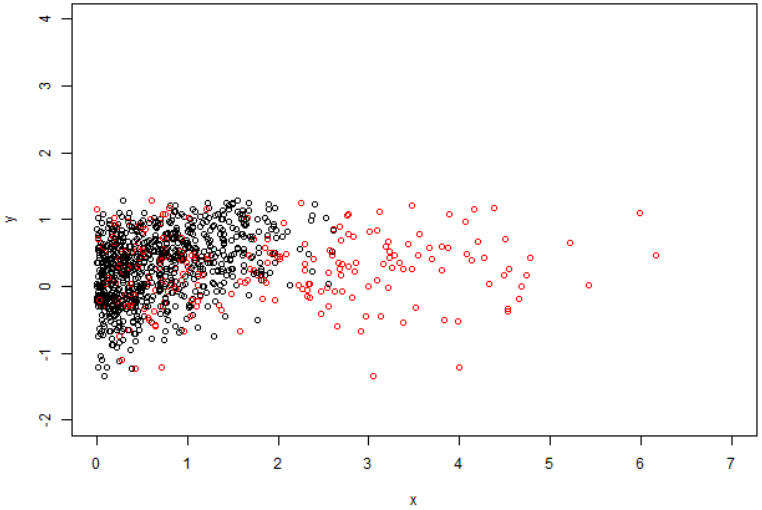
- “Hotdeck” imputation from 1940’s – 1950’s by census bureau
- uses Hollerith cards from “similar group” for missing data



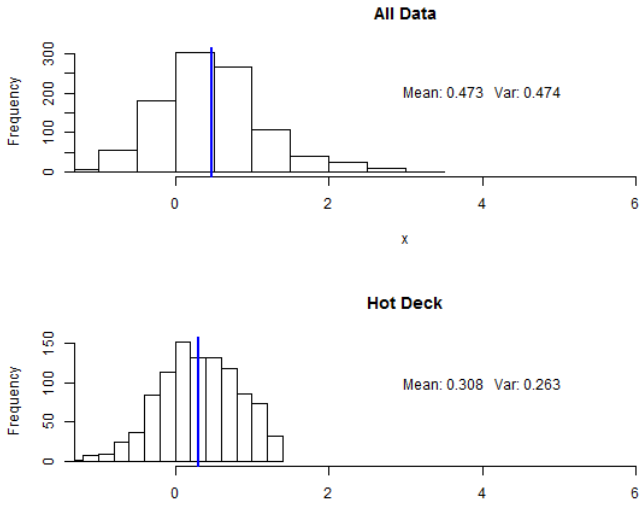
hotdeck (R code)

```
dfHD.imp <- dfMiss   #copy of data with missings

#create vector identifying missings
missing <- is.na(dfHD.imp$y)

#create sample pool from non-missing data
hotdeck <- dfHD.imp[!missing,"y"]  #sample pool

n <- length(hotdeck)  #size of sample pool
m <- sum(missing)     #how many samples do I need?

#sample m values (with replacement) from pool
hotdeck <- hotdeck[sample(n,m,replace=TRUE)]
```

scatter plot: hotdeck imputation



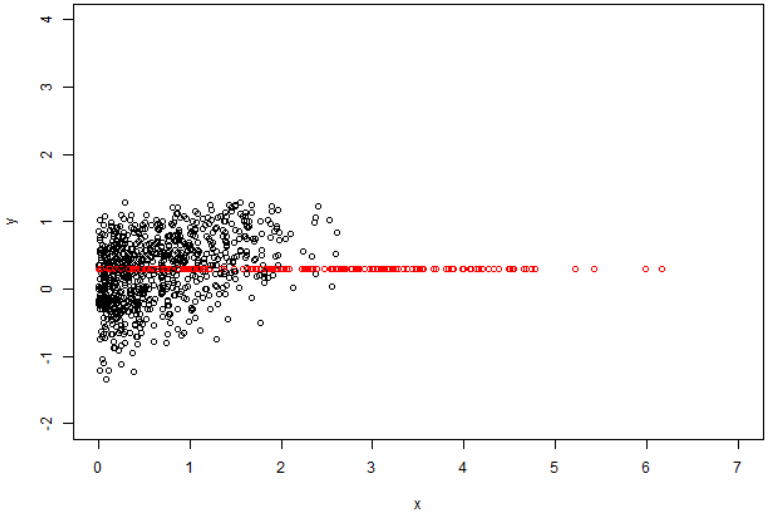
histogram: hotdeck



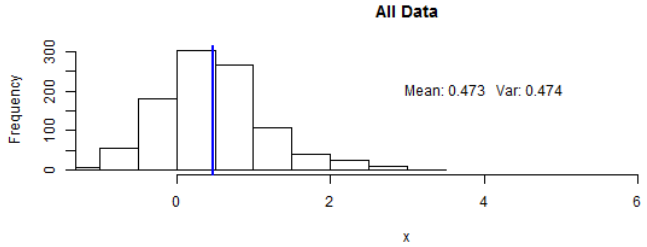
mean imputation (R code)

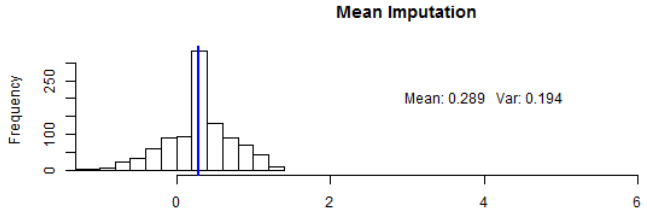
```
#imputation by mean

dfMean.imp<-dfMiss

dfMean.imp[missing,"y"]<-mean(dfMean.imp$y,na.rm=T)
```

scatter plot: mean imputation



histogram: mean imputation





regression imputation

Build regression model on complete cases. Use the model to generate predicted values for the missing ones.

`lm(dfMiss$y ~ dfMiss$x)`

		Estimate	Std. Error	t value	Pr(>|t|)	
Coefficients:	(Intercept)	0.00490	0.0217	0.236	0.821	
	x	0.49430	0.0159	30.99	< 2.2e-16	***

Residual std error: 0.4918 on 998 degrees of freedom

Multiple R-squared: 0.49, Adjusted R-squared: 0.49

F-statistic: 960.5 on 1 and 998 DF, p-value: < 2.2e-16

regression imputation

Build regression model on complete cases. Use the model to generate predicted values for the missing ones.

`lm(dfMiss$y ~ dfMiss$x)`

		Estimate	Std. Error	t value	Pr(>|t|)	
Coefficients:	(Intercept)	0.00490	0.0217	0.236	0.821	
	x	0.49430	0.0159	30.99	< 2.2e-16	***

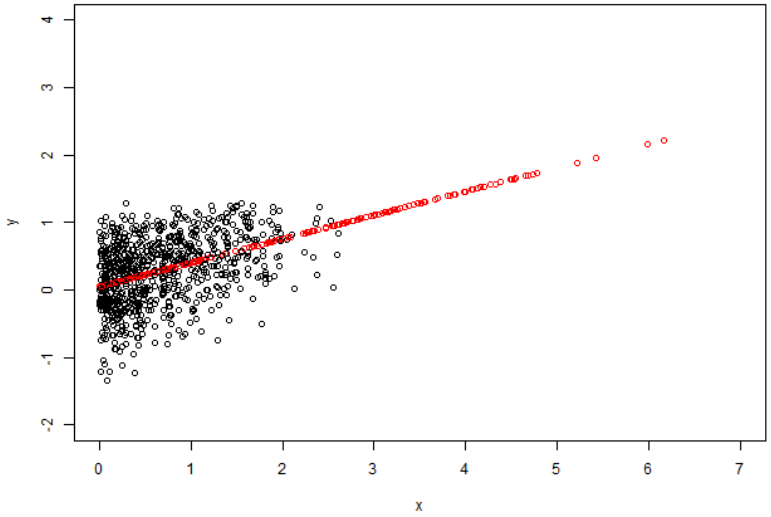
Residual std error: 0.4918 on 998 degrees of freedom

Multiple R-squared: 0.49, Adjusted R-squared: 0.49

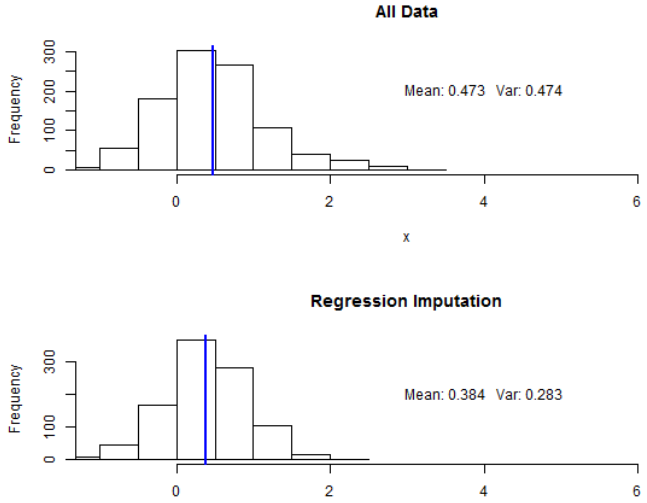
F-statistic: 960.5 on 1 and 998 DF, p-value: < 2.2e-16

$$\hat{y} = 0.00490 + 0.49430x$$

scatter plot: regression imputation



histogram: regression imputation



regression imputation with error

		Estimate	Std. Error	t value	Pr($>\|t\|$)	
Coefficients:	(Intercept)	0.00490	0.0217	0.236	0.821	
	x	0.49430	0.0159	30.99	$<$ 2.2e-16	***

Residual std error: 0.4918 on 998 degrees of freedom

Multiple R-squared: 0.49, Adjusted R-squared: 0.49

F-statistic: 960.5 on 1 and 998 DF, p-value: $<$ 2.2e-16

regression imputation with error

		Estimate	Std. Error	t value	Pr(> |t|)	
Coefficients:	(Intercept)	0.00490	0.0217	0.236	0.821	
	x	0.49430	0.0159	30.99	< 2.2e-16	***

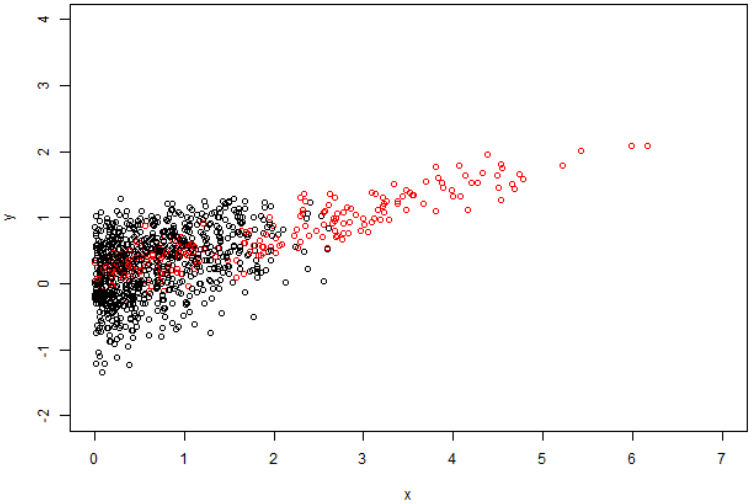
Residual std error: 0.4918 on 998 degrees of freedom

Multiple R-squared: 0.49, Adjusted R-squared: 0.49

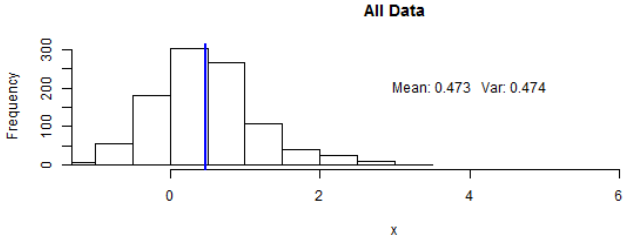
F-statistic: 960.5 on 1 and 998 DF, p-value: < 2.2e-16

$$\hat{y} = 0.00490 + 0.49430x + N(0, \text{SE}^2)$$

scatter plot: regression with error



histogram: regression with error



predictive mean matching

To mitigate problems with “global models” creating outrageous estimates, predictive mean matching is a hybrid approach that incorporates a model with the observed value range.

Predictive mean matching (PMM) is essentially a sophisticated “hot-deck” method which produces values that are more like real values.

- If the original variable is skewed, the imputed values will be skewed; if the original variable is bounded, the imputed values will be bounded
- PMM returns only observed values

predictive mean matching

Assume **x** has missing values and variables **y** and **z** do not.

1. Perform a linear regression $\mathbf{x} \sim \mathbf{y} + \mathbf{z}$ and estimate β (and σ)
2. Draw β^* from the "posterior predictive distribution" of β, (e.g., a multivariate normal distribution with mean = β and related σ)
3. Compute predicted values $\hat{\mathbf{x}}$ for *all* **x** using β^*: $\hat{\mathbf{x}}_{\text{miss}}$ and $\hat{\mathbf{x}}_{\text{obs}}$
4. For each case of missing **x**, identify non-missing cases whose $\hat{\mathbf{x}}_{\text{obs}}$ are closest $\hat{\mathbf{x}}_{\text{miss}}$
5. Randomly sample from these cases and use the observed value of **x** as the imputed value

predictive mean matching

Assume **x** has missing values and variables **y** and **z** do not.

1. Perform a linear regression $\mathbf{x} \sim \mathbf{y} + \mathbf{z}$ and estimate β (and σ)
2. Draw β^* from the "posterior predictive distribution" of β, (e.g., a multivariate normal distribution with mean = β and related σ)
3. Compute predicted values $\hat{\mathbf{x}}$ for *all* **x** using β^*: $\hat{\mathbf{x}}_{\text{miss}}$ and $\hat{\mathbf{x}}_{\text{obs}}$
4. For each case of missing **x**, identify non-missing cases whose $\hat{\mathbf{x}}_{\text{obs}}$ are closest $\hat{\mathbf{x}}_{\text{miss}}$
5. Randomly sample from these cases and use the observed value of **x** as the imputed value

predictive mean matching

Assume **x** has missing values and variables **y** and **z** do not.

1. Perform a linear regression $\mathbf{x} \sim \mathbf{y} + \mathbf{z}$ and estimate β (and σ)
2. Draw β^* from the "posterior predictive distribution" of β, (e.g., a multivariate normal distribution with mean = β and related σ)
3. Compute predicted values $\hat{\mathbf{x}}$ for *all* **x** using β^*: $\hat{\mathbf{x}}_{\text{miss}}$ and $\hat{\mathbf{x}}_{\text{obs}}$
4. For each case of missing **x**, identify non-missing cases whose $\hat{\mathbf{x}}_{\text{obs}}$ are closest $\hat{\mathbf{x}}_{\text{miss}}$
5. Randomly sample from these cases and use the observed value of **x** as the imputed value

predictive mean matching

Assume **x** has missing values and variables **y** and **z** do not.

1. Perform a linear regression $\mathbf{x} \sim \mathbf{y} + \mathbf{z}$ and estimate β (and σ)
2. Draw β^* from the "posterior predictive distribution" of β, (e.g., a multivariate normal distribution with mean = β and related σ)
3. Compute predicted values $\hat{\mathbf{x}}$ for *all* **x** using β^*: $\hat{\mathbf{x}}_{\text{miss}}$ and $\hat{\mathbf{x}}_{\text{obs}}$
4. For each case of missing **x**, identify non-missing cases whose $\hat{\mathbf{x}}_{\text{obs}}$ are closest $\hat{\mathbf{x}}_{\text{miss}}$
5. Randomly sample from these cases and use the observed value of **x** as the imputed value

predictive mean matching

Assume **x** has missing values and variables **y** and **z** do not.

1. Perform a linear regression $\mathbf{x} \sim \mathbf{y} + \mathbf{z}$ and estimate β (and σ)
2. Draw β^* from the "posterior predictive distribution" of β, (e.g., a multivariate normal distribution with mean = β and related σ)
3. Compute predicted values $\hat{\mathbf{x}}$ for *all* **x** using β^*: $\hat{\mathbf{x}}_{\text{miss}}$ and $\hat{\mathbf{x}}_{\text{obs}}$
4. For each case of missing **x**, identify non-missing cases whose $\hat{\mathbf{x}}_{\text{obs}}$ are closest $\hat{\mathbf{x}}_{\text{miss}}$
5. Randomly sample from these cases and use the observed value of **x** as the imputed value

predictive mean matching

Note: in PMM, the purpose of the linear regression is *not* to generate imputed values, but to construct a metric for matching.

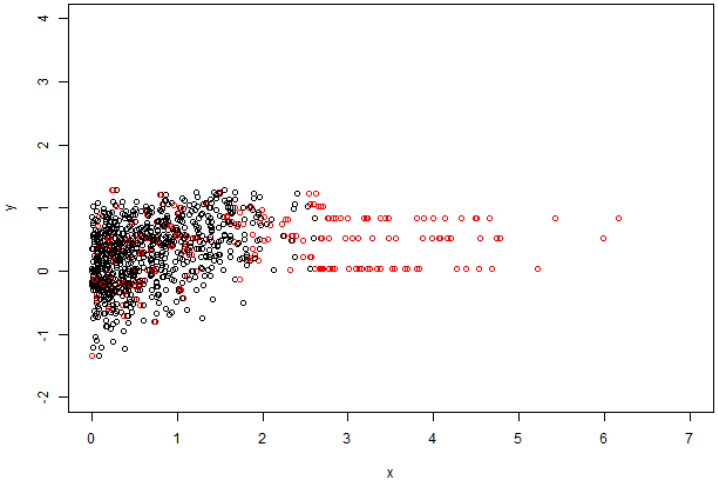
predictive mean matching

Note: in PMM, the purpose of the linear regression is *not* to generate imputed values, but to construct a metric for matching.

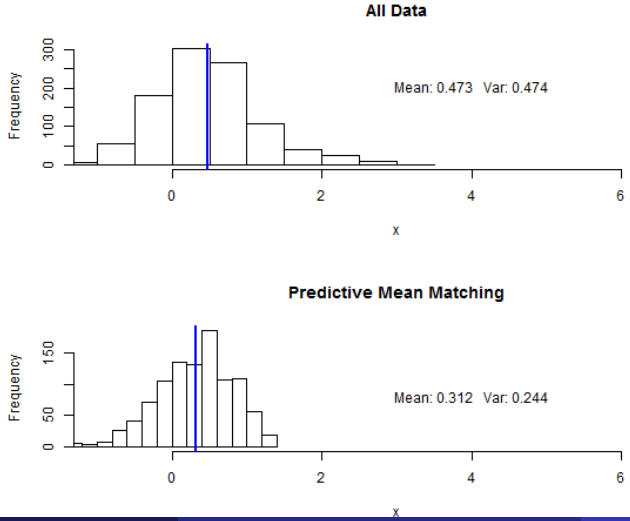
```
library(mice)
dfPMM.imp <- dfMiss

dfPMM.imp[missing,"y"] <-
            mice.impute.pmm(dfPMM.imp$y,
                        !dfPMM.imp$missing,
                        dfPMM.imp$x)
```

The default set of "donor pool" in the `mice` package is 5.

scatter plot: predictive mean matching



histogram: predictive mean matching



k-nearest neighbor

k-nearest neighbors (kNN) is a simple modeling technique that has applications beyond missing value imputation.

k-nearest neighbor

k-nearest neighbors (kNN) is a simple modeling technique that has applications beyond missing value imputation.

To that end, let's digress a bit and introduce the important concept of a *neighbor*.

what is a neighbor?

What is a *neighbor* of an observation?
How do you determine which cases are it's *nearest* neighbors?

what is a neighbor?

What is a *neighbor* of an observation?
How do you determine which cases are it's *nearest* neighbors?

Remember, usually the data is mixed-type,
e.g. what are neighbors of cases:

x	y	z	Gender	Vote	Education
0.15	NA	.94	M	Dem	3
0.47	0.33	NA	F	NA	2
0.39	NA	NA	F	Rep	3
0.07	0.10	NA	M	Ind	1

distance

To evaluate *nearness*, we need a measure for distance, e.g.:

- Euclidean distance:

$$d(i,j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \ldots + (x_{ip} - x_{jp})^2}$$

- Manhattan distance:

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}|$$

- Minkowski distance:

$$d(i,j) = \left(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \ldots + |x_{ip} - x_{jp}|^q\right)^{\frac{1}{q}}$$

mixed data: Gower's distance

Use distance measure between 0 and 1 for each variable: $d_{ij}^{(f)}$

mixed data: Gower's distance

Use distance measure between 0 and 1 for each variable: $d_{ij}^{(f)}$

Aggregate:

$$d(i,j) = \frac{1}{p}\sum_{f=1}^{p} \delta_{ij}^{f} d_{ij}^{(f)}$$

where δ_{ij}^{f} is a weighting factor

$$\delta_{ij}^{f} = \begin{cases} 0 & \text{if } x_{if} \text{ or } x_{jf} \text{ is missing} \\ 0 & \text{if } x_{if} \text{ and } x_{jf} \text{ are false (with binary data)} \\ 1 & \text{otherwise} \end{cases}$$

Gower's distance

For interval or ordinal data....

Uses range scaling:

$$d_{ij}^{(f)} = \frac{\left| x_{if} - x_{jf} \right|}{R_f}$$

x_{if}, x_{jf} are values for object i and j in variable f
and R_f is the range of variable f

Ordinal: Use normalized ranks then use interval-scaled method

Gower's distance

For nominal data...

Simple matching coefficient:

$$d^{(f)}(i,j) = \frac{m}{p}$$

where

m: number of variables in which object i and j *mismatch*

p: number of variables

Gower's distance

Binary data:

Jaccard distance:

$$d^{(f)}(i,j) = 1 - \frac{b_{ij}}{b_{ij} + x_{ij}}$$

where,
- b : both values are 1
- x : only one value is 1
- n : both values are 0

Gower's distance

Binary data:

Jaccard distance:

$$d^{(f)}(i,j) = 1 - \frac{b_{ij}}{b_{ij} + x_{ij}}$$

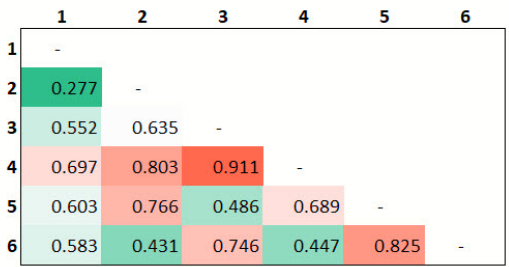
where,
- b : both values are 1
- x : only one value is 1
- n : both values are 0

i	j	b	n	x	d_{ij}
101000	111000	2	3	1	$0.3\bar{3}$

Obs	male	citizen	own car	marital status	State	age	income
1	TRUE	TRUE	TRUE	M	TX	41	53
2	FALSE	TRUE	TRUE	M	TX	39	72
3	TRUE	TRUE	FALSE	S	CA	43	70
4	FALSE	FALSE	TRUE	D	OK	48	47
5	TRUE	TRUE	FALSE	D	NY	50	63
6	FALSE	FALSE	TRUE	M	OK	45	71

dist	male	citizen	own car	marital status	State	age	income
1 & 2	$1-\frac{2}{3}$	$1-\frac{2}{3}$	$1-\frac{2}{3}$	$\frac{0}{2}$	$\frac{0}{2}$	$\frac{2}{11}$	$\frac{19}{25}$
2 & 6	$1-\frac{1}{2}$	$1-\frac{1}{2}$	$1-\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{6}{11}$	$\frac{1}{25}$

Obs	male	citizen	own car	marital status	State	age	income
1	TRUE	TRUE	TRUE	M	TX	41	53
2	FALSE	TRUE	TRUE	M	TX	39	72
3	TRUE	TRUE	FALSE	S	CA	43	70
4	FALSE	FALSE	TRUE	D	OK	48	47
5	TRUE	TRUE	FALSE	D	NY	50	63
6	FALSE	FALSE	TRUE	M	OK	45	71



	1	2	3	4	5	6
1	-					
2	0.277	-				
3	0.552	0.635	-			
4	0.697	0.803	0.911	-		
5	0.603	0.766	0.486	0.689	-	
6	0.583	0.431	0.746	0.447	0.825	-

distance metrics using R

For distance calculations in R, see the following functions:

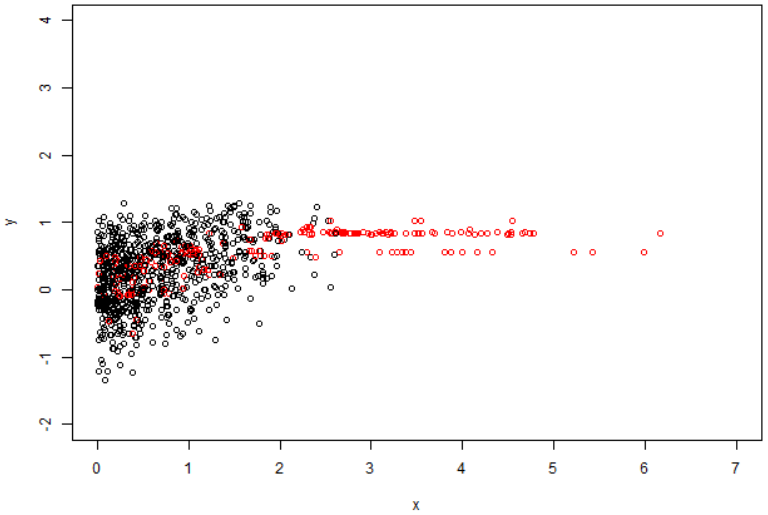
- `dist` in base distribution (includes Minkowski)
- `daisy` in package `cluster`
- `gower.dist` in package `StatMatch`

k-nearest neighbor

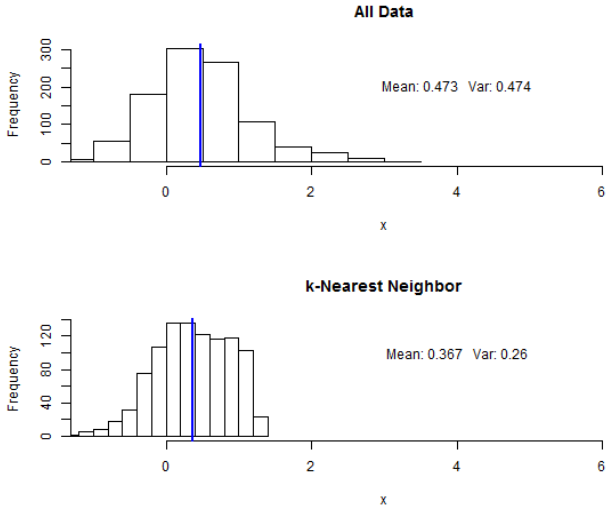
kNN uses the values of the *k*-nearest neighbors of the observation to impute the missing value

```
library(VIM)

#using x,y, and z data for kNN computation
dfKNN.imp <- kNN(dfMiss[,1:3],k=5)
```

scatter plot: kNN with 5 neighbors



histogram: kNN

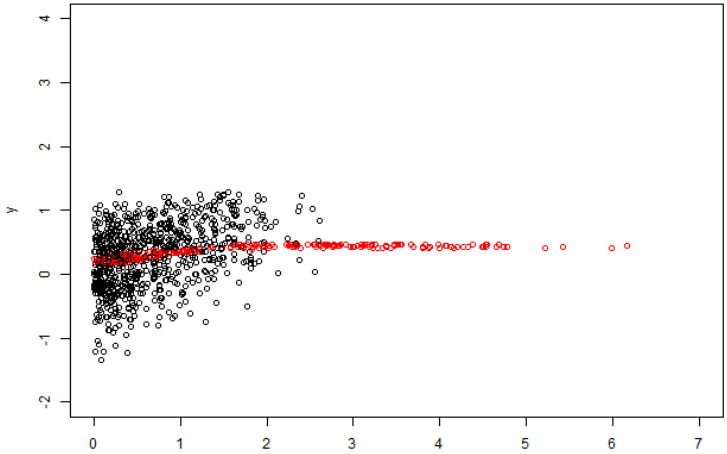


scatter plot: kNN with 400 neighbors

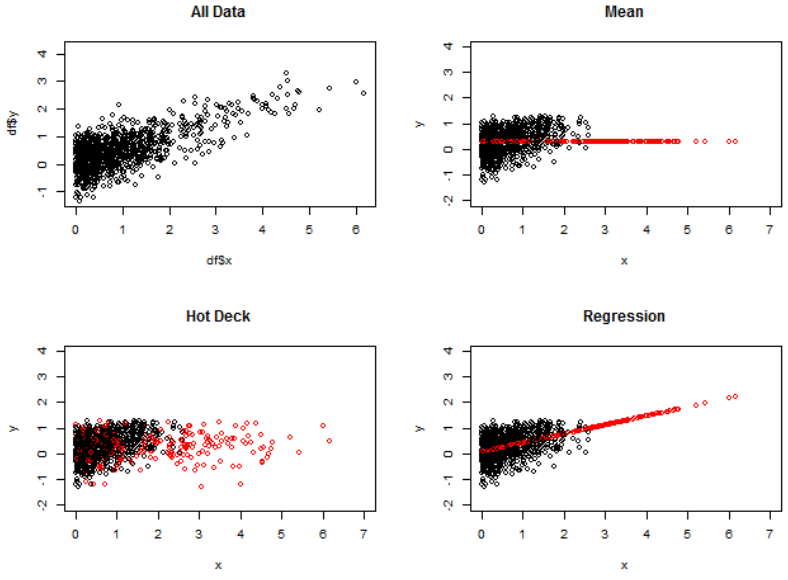
What would happen if we use 400 of the nearest neighbors?

scatter plot: kNN with 400 neighbors

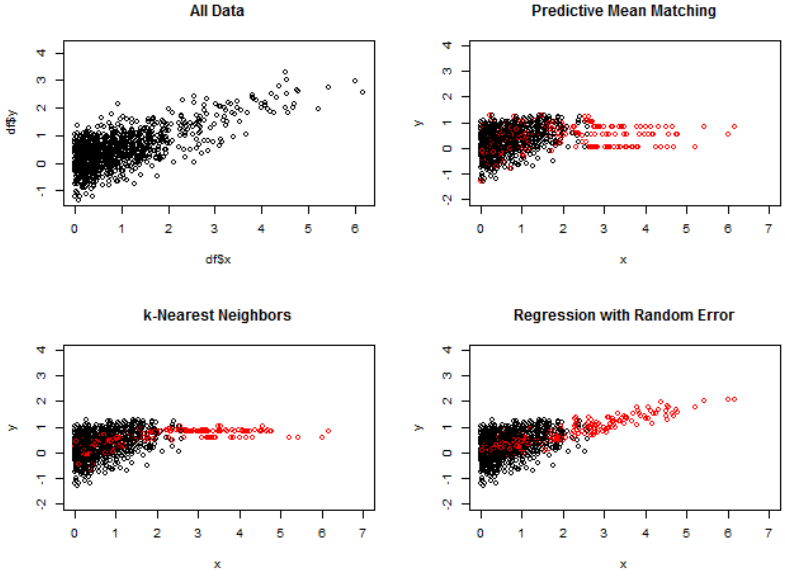
What would happen if we use 400 of the nearest neighbors?



single imputation summary



single imputation summary



single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

single imputation and missing data

- Missing data is bad, yet almost certain to occur
- The three missing value mechanism types may all be present
  - most resolution approaches assume MCAR or MAR
- Imputation methods do not create new information...
  - they are not magical
- ...and can hurt estimates
  - e.g. by artificially decreasing variance
- The goals of imputation:
  - Preserve the essential characteristics of the data (distributions, relationships among the variables)
  - Maintain the representativeness of the analyzed data

multiple imputation

Multiple imputation (MI) is a modern and increasingly popular Monte Carlo technique for analyzing data with missing values (Rubin 1987).

multiple imputation

Multiple imputation (MI) is a modern and increasingly popular Monte Carlo technique for analyzing data with missing values (Rubin 1987).

MI is comprised of four steps.

multiple imputation

Multiple imputation (MI) is a modern and increasingly popular Monte Carlo technique for analyzing data with missing values (Rubin 1987).

MI is comprised of four steps.

1. Replication

Rubin, D. 1987. *Multiple Imputation for Nonresponse in Surveys*. J. Wiley & Sons, New York.

multiple imputation

Multiple imputation (MI) is a modern and increasingly popular Monte Carlo technique for analyzing data with missing values (Rubin 1987).

MI is comprised of four steps.

1. Replication
2. Imputation

Rubin, D. 1987. *Multiple Imputation for Nonresponse in Surveys*. J. Wiley & Sons, New York.

multiple imputation

Multiple imputation (MI) is a modern and increasingly popular Monte Carlo technique for analyzing data with missing values (Rubin 1987).

MI is comprised of four steps.

1. Replication
2. Imputation
3. Analysis

Rubin, D. 1987. *Multiple Imputation for Nonresponse in Surveys*. J. Wiley & Sons, New York.

multiple imputation

Multiple imputation (MI) is a modern and increasingly popular Monte Carlo technique for analyzing data with missing values (Rubin 1987).

MI is comprised of four steps.

1. Replication
2. Imputation
3. Analysis
4. Recombination

Rubin, D. 1987. *Multiple Imputation for Nonresponse in Surveys*. J. Wiley & Sons, New York.

multiple imputation

MI produces estimates that have nearly optimal statistical properties:

- approximately unbiased in large samples
- stable estimates

MI requires:

- missing data be “ignorable”
- good imputation model

replication step

Create $m > 1$ copies of the original data sets.

replication step

Create $m > 1$ copies of the original data sets.

- the number of sufficient copies for imputations, m, is typically small, i.e. 3-10 (depends on percentage of missing data)
- managing multiple copies of the data set can be complex
  - e.g., with “big data” and large volume, maybe very difficult
  - practically, keeping up with multiple copies could be a headache...
  - however, modern software packages (including R) take care of the latter problem automatically

imputation step

In each copy, perform *single imputation* with some random element, e.g. regression with error.

imputation step

In each copy, perform *single imputation* with some random element, e.g. regression with error.

The result is *m* complete, but slightly different data sets.

imputation step

imputation step

- the imputation step in MI is the most complicated

imputation step

- the imputation step in MI is the most complicated
- many possible strategies, e.g.,

imputation step

- the imputation step in MI is the most complicated
- many possible strategies, e.g.,
  - regression with error

imputation step

- the imputation step in MI is the most complicated
- many possible strategies, e.g.,
  - regression with error
  - predictive mean matching

imputation step

- the imputation step in MI is the most complicated
- many possible strategies, e.g.,
  - regression with error
  - predictive mean matching
  - random forests

imputation step

- the imputation step in MI is the most complicated
- many possible strategies, e.g.,
  - regression with error
  - predictive mean matching
  - random forests
  - random sampling

imputation step

- the imputation step in MI is the most complicated
- many possible strategies, e.g.,
  - regression with error
  - predictive mean matching
  - random forests
  - random sampling
- imputation model needs to be compatible with the analysis model

imputation step: several missing variables

A brief digression...

It is common to have missing data in *several* variables in an analysis.

imputation step: several missing variables

A brief digression...

It is common to have missing data in *several* variables in an analysis.

In such a case you cannot simply impute based on a model for a single partially observed variable y given a set of fully observed X variables.

imputation step: several missing variables

A brief digression...

It is common to have missing data in *several* variables in an analysis.

In such a case you cannot simply impute based on a model for a single partially observed variable y given a set of fully observed X variables.

One method to address this is to perform iterative imputation.

iterative imputation

Iterative imputation uses the complete cases for a simple imputation of all missing values.

iterative imputation

Iterative imputation uses the complete cases for a simple imputation of all missing values.

And then uses the full information in iterative steps to improve on the previous imputation values.

iterative imputation

Iterative imputation uses the complete cases for a simple imputation of all missing values.

And then uses the full information in iterative steps to improve on the previous imputation values.

The process continues until a certain level of convergence is reached.

iterative regression imputation

Assume the variables with missingness are a $n \times k$ matrix Y with columns $y_1, \ldots, y_k$ and the fully observed predictors are an $n \times (p - k)$ matrix X.

iterative regression imputation

Assume the variables with missingness are a $n \times k$ matrix Y with columns $y_1, \ldots, y_k$ and the fully observed predictors are an $n \times (p - k)$ matrix X.

1. impute all missing Y values using a crude approach (e.g., impute by randomly selecting from observed outcomes of variable)
2. re-impute y_1 based on $y_2, \ldots, y_k$ and X
3. re-impute y_2 based on $y_1, y_3, \ldots, y_k$ and X (using the newly imputed values for y_1)
4. and so on, randomly imputing each variable and looping through until approximate convergence.

iterative regression imputation

Assume the variables with missingness are a $n \times k$ matrix Y with columns $y_1, \ldots, y_k$ and the fully observed predictors are an $n \times (p - k)$ matrix X.

1. impute all missing Y values using a crude approach (e.g., impute by randomly selecting from observed outcomes of variable)
2. re-impute y_1 based on $y_2, \ldots, y_k$ and X
3. re-impute y_2 based on $y_1, y_3, \ldots, y_k$ and X (using the newly imputed values for y_1)
4. and so on, randomly imputing each variable and looping through until approximate convergence.

iterative regression imputation

Assume the variables with missingness are a $n \times k$ matrix Y with columns $y_1, \ldots, y_k$ and the fully observed predictors are an $n \times (p - k)$ matrix X.

1. impute all missing Y values using a crude approach (e.g., impute by randomly selecting from observed outcomes of variable)
2. re-impute y_1 based on $y_2, \ldots, y_k$ and X
3. re-impute y_2 based on $y_1, y_3, \ldots, y_k$ and X (using the newly imputed values for y_1)
4. and so on, randomly imputing each variable and looping through until approximate convergence.

iterative regression imputation

Assume the variables with missingness are a $n \times k$ matrix Y with columns $y_1, \ldots, y_k$ and the fully observed predictors are an $n \times (p - k)$ matrix X.

1. impute all missing Y values using a crude approach (e.g., impute by randomly selecting from observed outcomes of variable)
2. re-impute y_1 based on $y_2, \ldots, y_k$ and X
3. re-impute y_2 based on $y_1, y_3, \ldots, y_k$ and X (using the newly imputed values for y_1)
4. and so on, randomly imputing each variable and looping through until approximate convergence.

analysis step

Analyze each of the m complete data sets independently.

analysis step

Analyze each of the *m* complete data sets independently.

A standard analysis is run on each of the datasets.

Calculate and save the estimates and standard errors from each analysis.

The inferences will later be combined across all *m* datasets.

recombination step

Combine analyses for some quantity of interest q by computing average and overall variance estimate across all m datasets.

e.g., the quantity of interest might be a regression coefficient, β_1

Let $\hat{q}_j$ denote the estimate obtained from dataset $j = 1, \ldots, m$.
Let s_j^2 denote the corresponding variance associated with $\hat{q}_j$.

The overall estimate is the average; the overall variance has two parts: "within" and "between" imputation variance.

overall estimate: $\bar{q} = \frac{1}{m} \sum_{j=1}^{m} \hat{q}_j$

"within" : $W = \frac{1}{m} \sum_{j=1}^{m} s_j^2$

"between" : $B = \frac{1}{m-1} \sum_{j=1}^{m} (\hat{q}_j - \bar{q})^2$

overall variance : $\bar{s}_q^2 = W + \left(1 + \frac{1}{m}\right) B$

Confidence intervals are obtained by:

$$\bar{q} \pm t_{\alpha,df} \bar{s}_q$$

where the degrees of freedom are given by,

$$df = (m-1)\left(1 + \frac{mW}{(m+1)B}\right)^2$$

A significance test of the null hypothesis $q = 0$ is performed by comparing the ratio to the same t-distribution.

maximum likelihood

Maximum likelihood (ML) methods represent another kind of modern approach for handling missing values.

maximum likelihood

Maximum likelihood (ML) methods represent another kind of modern approach for handling missing values.

- ML not dependent on simulation

P. Allison. 2012. Paper 312-2012: Handling Missing Data by Maximum Likelihood.
statisticalhorizons.com/resources/unpublished-papers Statistical Horizons, Haverford, PA, USA

maximum likelihood

Maximum likelihood (ML) methods represent another kind of modern approach for handling missing values.

- ML not dependent on simulation
- Therefore, unlike MI, ML always produces the same result

P. Allison. 2012. Paper 312-2012: Handling Missing Data by Maximum Likelihood.
statisticalhorizons.com/resources/unpublished-papers Statistical Horizons, Haverford, PA, USA

maximum likelihood

Maximum likelihood (ML) methods represent another kind of modern approach for handling missing values.

- ML not dependent on simulation
- Therefore, unlike MI, ML always produces the same result
- In ML no conflict between the imputation and analysis models

P. Allison. 2012. Paper 312-2012: Handling Missing Data by Maximum Likelihood.
statisticalhorizons.com/resources/unpublished-papers Statistical Horizons, Haverford, PA, USA

maximum likelihood

Maximum likelihood (ML) methods represent another kind of modern approach for handling missing values.

- ML not dependent on simulation
- Therefore, unlike MI, ML always produces the same result
- In ML no conflict between the imputation and analysis models
- In ML, you do need to consider the distributions of all your variables

P. Allison. 2012. Paper 312-2012: Handling Missing Data by Maximum Likelihood.
statisticalhorizons.com/resources/unpublished-papers Statistical Horizons, Haverford, PA, USA

maximum likelihood

Suppose we have n observations and p variables.

First step in ML is to construct the likelihood function:

$$L = \prod_{i=1}^{n} f_i(x_{i1}, \ldots, x_{ip}; \Theta)$$

where $f_i(.)$ is the joint probability function for observations $i = 1, \ldots, n$ and Θ is a set of parameters to be estimated. The ML estimates for Θ are the ones that maximize L.

maximum likelihood

If we have missing data, e.g., if for observation i, variables x_1 and x_2 are missing, the joint probability i is:

$$f_i^*(x_{i3}, \ldots, x_{ip}; \Theta) = \int_{x_1} \int_{x_2} f_i(x_{i1}, \ldots, x_{ip}) dx_1 dx_2$$

If there are m observations with complete data and $n - m$ observations with data missing on x_1 and x_2, the overall likelihood function for the full data set becomes:

$$L = \prod_{i=1}^{m} f_i(x_{i1}, \ldots, x_{ip}; \Theta) \prod_{i=m+1}^{n} f_i^*(x_{i3}, \ldots, x_{ip}; \Theta)$$

maximum likelihood

Major issue...

- With MI, the *m* datasets are complete and can be analyzed by any tool/software that you would normally use; ML requires specialized software which limits your analysis options.
- This will probably improve over time.

maximum likelihood

Major issue...

- With MI, the *m* datasets are complete and can be analyzed by any tool/software that you would normally use; ML requires specialized software which limits your analysis options.
- This will probably improve over time.

In many applications, at least for now, approximate solutions with good properties (i.e., MI) may be preferable to those with potentially theoretically better properites, (i.e., ML).

multivariate MI example (using `mice`)

Example:

multivariate MI example (using `mice`)

Example:

- data set with 100 observations and 5 variables: y, x_1, x_2, x_3, x_4

multivariate MI example (using `mice`)

Example:

- data set with 100 observations and 5 variables: y, x_1, x_2, x_3, x_4
- four variables have missing values
  (11% for y; 71% for x_1; 35% for x_2; 40% for x_3)

multivariate MI example (using `mice`)

Example:

- data set with 100 observations and 5 variables: y, x_1, x_2, x_3, x_4
- four variables have missing values
  (11% for y; 71% for x_1; 35% for x_2; 40% for x_3)
- Missing value mechanism: MAR

multivariate MI example (using `mice`)

Example:

- data set with 100 observations and 5 variables: y, x_1, x_2, x_3, x_4
- four variables have missing values
  (11% for y; 71% for x_1; 35% for x_2; 40% for x_3)
- Missing value mechanism: MAR
- Imputation model: iterative regression model with error

multivariate MI example (using `mice`)

Example:

- data set with 100 observations and 5 variables: y, x_1, x_2, x_3, x_4
- four variables have missing values
  (11% for y; 71% for x_1; 35% for x_2; 40% for x_3)
- Missing value mechanism: MAR
- Imputation model: iterative regression model with error
- Analysis model: $y \sim x1 + x2 + x3 + x4$

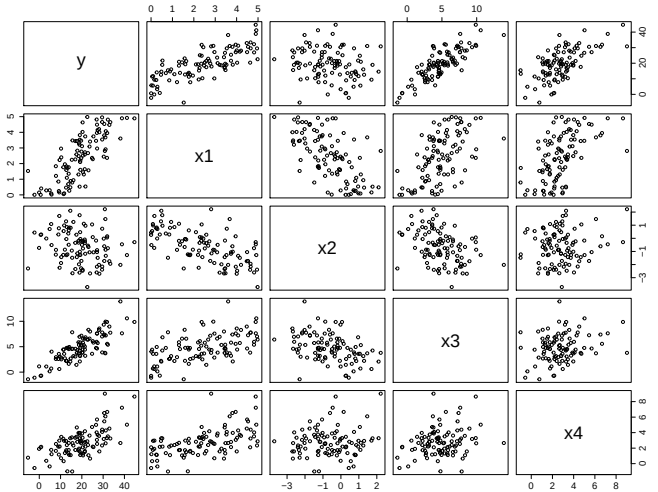
multivariate MI example (using `mice`)

Example:

- data set with 100 observations and 5 variables: y, x_1, x_2, x_3, x_4
- four variables have missing values
  (11% for y; 71% for x_1; 35% for x_2; 40% for x_3)
- Missing value mechanism: MAR
- Imputation model: iterative regression model with error
- Analysis model: $y \sim x1 + x2 + x3 + x4$
- Goal: Estimate coefficients for analysis model

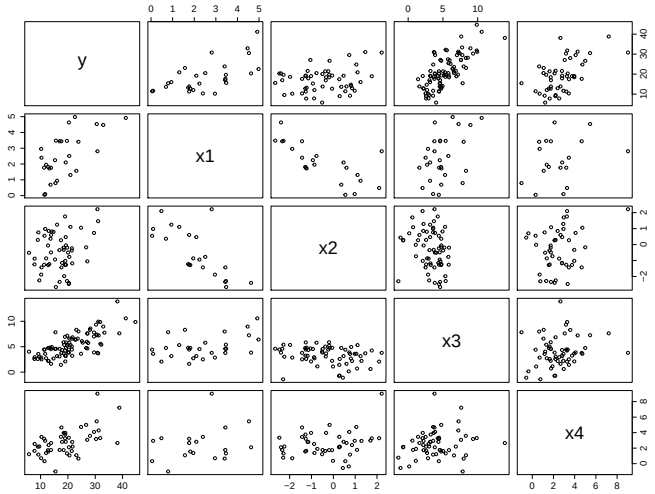
multivariate MI example (using `mice`)

Full data pairs plot



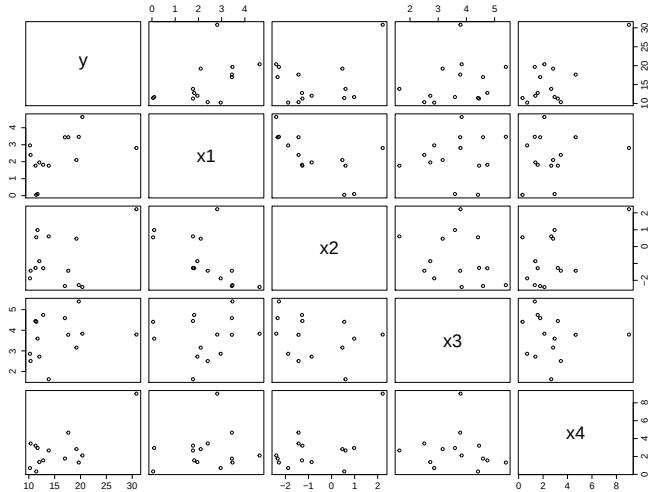
multivariate MI example (using `mice`)

Available cases pairs plot



multivariate MI example (using `mice`)

Complete cases pairs plot



multivariate MI example (using `mice`)

		Estimate	Std. Error	Pr($>\|t\|$)	Notes
Full data	(Intercept)	-0.4613	0.2376	0.0552	
	x1	5.1622	0.15894	$< 2^{-16}$	
	x2	4.2215	0.15024	$< 2^{-16}$	95 df
	x3	2.0978	0.04478	$< 2^{-16}$	
	x4	-0.0734	0.0986	0.4589	
Listwise deletion	(Intercept)	-1.837	1.0596	0.117	
	x1	5.3471	0.3608	1.25E-07	
	x2	4.7343	0.3832	6.01E-07	9 df
	x3	2.4607	0.2536	4.60E-06	
	x4	-0.2205	0.1977	0.294	
MICE	(Intercept)	-0.4993	1.2335	7.30E-01	imputed
	x1	4.9422	0.3426	1.58E-04	71
	x2	4.2716	0.2767	1.41E-05	35
	x3	2.1312	0.2643	3.96E-02	0
	x4	-0.0504	0.1378	7.23E-01	40

summary

- Most imputation methods expect MCAR or MAR as missing value mechanisms
- Deletion should only be used under MCAR and with relatively few affected cases
- Single imputation could be used when very few cases have missing values; even then, it is not advisable
- MI is not the *only* principled way to deal with missing values, but it is one good way
- ML is an excellent technique, however not as flexible/accessible as MI
- Modern software makes MI easy to use

summary

- Most imputation methods expect MCAR or MAR as missing value mechanisms
- Deletion should only be used under MCAR and with relatively few affected cases
- Single imputation could be used when very few cases have missing values; even then, it is not advisable
- MI is not the *only* principled way to deal with missing values, but it is one good way
- ML is an excellent technique, however not as flexible/accessible as MI
- Modern software makes MI easy to use

summary

- Most imputation methods expect MCAR or MAR as missing value mechanisms
- Deletion should only be used under MCAR and with relatively few affected cases
- Single imputation could be used when very few cases have missing values; even then, it is not advisable
- MI is not the *only* principled way to deal with missing values, but it is one good way
- ML is an excellent technique, however not as flexible/accessible as MI
- Modern software makes MI easy to use

summary

- Most imputation methods expect MCAR or MAR as missing value mechanisms
- Deletion should only be used under MCAR and with relatively few affected cases
- Single imputation could be used when very few cases have missing values; even then, it is not advisable
- MI is not the *only* principled way to deal with missing values, but it is one good way
- ML is an excellent technique, however not as flexible/accessible as MI
- Modern software makes MI easy to use

summary

- Most imputation methods expect MCAR or MAR as missing value mechanisms
- Deletion should only be used under MCAR and with relatively few affected cases
- Single imputation could be used when very few cases have missing values; even then, it is not advisable
- MI is not the *only* principled way to deal with missing values, but it is one good way
- ML is an excellent technique, however not as flexible/accessible as MI
- Modern software makes MI easy to use

summary

- Most imputation methods expect MCAR or MAR as missing value mechanisms
- Deletion should only be used under MCAR and with relatively few affected cases
- Single imputation could be used when very few cases have missing values; even then, it is not advisable
- MI is not the *only* principled way to deal with missing values, but it is one good way
- ML is an excellent technique, however not as flexible/accessible as MI
- Modern software makes MI easy to use

feature engineering

- Feature transformation
- Feature construction
- Feature extraction
- Feature selection

definition of feature

Features

Features are functions of the original measurement variables that are useful for classification, pattern recognition, prediction, or other modeling technique.

transformations

"This seems like a kind of cheating. You don't like how the data are, so you decide to change them."

transformations

"This seems like a kind of cheating. You don't like how the data are, so you decide to change them."

"But how do I know when this trick will work with some other data, or if another trick is needed, or if no transformation is needed?"

transformations

"This seems like a kind of cheating. You don't like how the data are, so you decide to change them."

"But how do I know when this trick will work with some other data, or if another trick is needed, or if no transformation is needed?"

"Transformations are needed because there is no guarantee that the world works on the scales it happens to be measured on."

transformations

"This seems like a kind of cheating. You don't like how the data are, so you decide to change them."

"But how do I know when this trick will work with some other data, or if another trick is needed, or if no transformation is needed?"

"Transformations are needed because there is no guarantee that the world works on the scales it happens to be measured on."

"Transformations are most appropriate when they match a scientific view of how a variable behaves."

transformations

"This seems like a kind of cheating. You don't like how the data are, so you decide to change them."

"But how do I know when this trick will work with some other data, or if another trick is needed, or if no transformation is needed?"

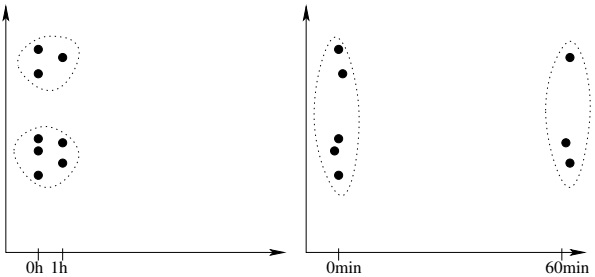
"Transformations are needed because there is no guarantee that the world works on the scales it happens to be measured on."

"Transformations are most appropriate when they match a scientific view of how a variable behaves."

"Transformations impact interpretability"

standardization

Some techniques (e.g. PCA, clustering) are sensitive to scale



To mitigate, some kind of standardization should be applied.

standardization

for a numerical attribute X, value $x_i \in X$:

min-max normalization : $\text{dom}(X) \rightarrow [0, 1]$

$$x_i' \leftarrow \frac{x_i - \min_X}{\max_X - \min_X}$$

standardization

for a numerical attribute X, value $x_i \in X$:

min-max normalization : $\text{dom}(X) \rightarrow [0, 1]$

$$x_i^{'} \leftarrow \frac{x_i - \min_X}{\max_X - \min_X}$$

z-score standardization : $\text{dom}(X) \rightarrow \mathbb{R}$

$$x_i^{'} \leftarrow \frac{x_i - \bar{x}}{s}$$

standardization

for a numerical attribute X, value $x_i \in X$:

min-max normalization : $\text{dom}(X) \rightarrow [0, 1]$

$$x_i^{'} \leftarrow \frac{x_i - \min_X}{\max_X - \min_X}$$

z-score standardization : $\text{dom}(X) \rightarrow \mathbb{R}$

$$x_i^{'} \leftarrow \frac{x_i - \bar{x}}{s}$$

robust z-score standardization : $\text{dom}(X) \rightarrow \mathbb{R}$

$$x_i^{'} \leftarrow \frac{x_i - \tilde{x}}{IQR_x}$$

transforming skewness

Problems with skewed distribution

- Data difficult to examine because most observations are in a small part of the range of the data.
- The mean is not a good summary of the center of a skewed distribution.
- Certain techniques (e.g. outlier identification, some imputation techniques, etc.) expect the data to be normally distributed.

transforming skewness

Problems with skewed distribution

- Data difficult to examine because most observations are in a small part of the range of the data.
- The mean is not a good summary of the center of a skewed distribution.
- Certain techniques (e.g. outlier identification, some imputation techniques, etc.) expect the data to be normally distributed.

transforming skewness

Problems with skewed distribution

- Data difficult to examine because most observations are in a small part of the range of the data.
- The mean is not a good summary of the center of a skewed distribution.
- Certain techniques (e.g. outlier identification, some imputation techniques, etc.) expect the data to be normally distributed.

transforming skewness

Ladder of Powers

transforming skewness

Ladder of Powers

- Positive skew: need to compress large values
  $\rightarrow$ *descend* the ladder of powers

transforming skewness

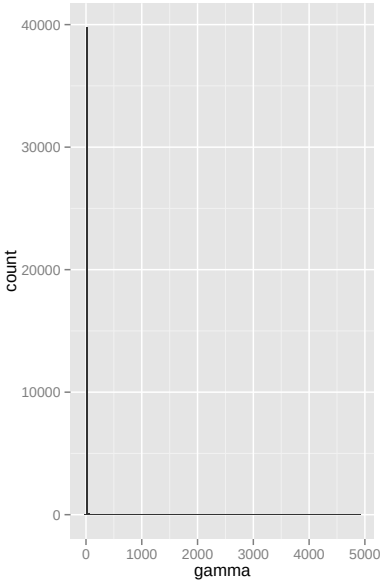
Ladder of Powers

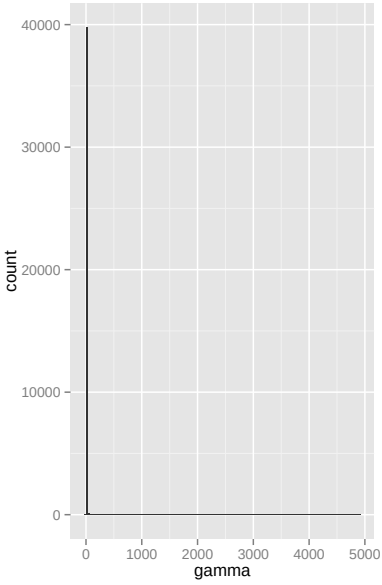
- Positive skew: need to compress large values
  $\rightarrow$ *descend* the ladder of powers
- Negative skew: need to compress small values
  $\rightarrow$ *ascend* the ladder of powers

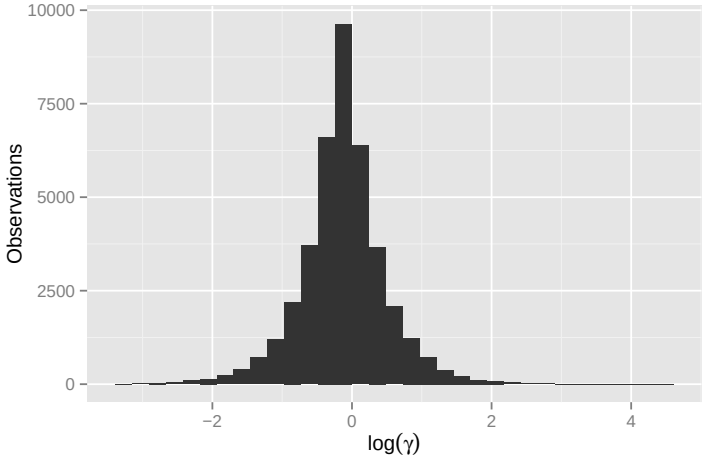
ladder of powers

Transformation is (basically): $x \leftarrow x^{\lambda}$

λ	Transformation $f_{\lambda}(x)$
3	x^3
2	x^2
1	x
$\frac{1}{2}$	$\sqrt{x}$
0	$\log(x)$
-1	$-\frac{1}{x}$
-2	$-\frac{1}{x^2}$
-3	$-\frac{1}{x^3}$


40000
30000
20000
10000
0
count
0
1000
2000
3000
4000
5000
gamma


40000
30000
20000
10000
0
count
0
1000
2000
3000
4000
5000
gamma


10000
7500
5000
2500
0
Observations
−2
0
2
4
log(γ)

Box Cox transformations

Box-Cox transformations try to make the data more normal.

$$x \leftarrow \begin{cases} \frac{x^{\lambda}-1}{\lambda} & \text{if } \lambda \neq 0 \\ \log(x) & \text{if } \lambda = 0 \end{cases}$$

transformation example with R

Example transformations on the `Prestige` data frame in the `car` package; as well as simulated exponential distributions.

Using the `symbox` function from `car` package, and the `boxcox` function from `EnvStats` package.

feature construction

Feature construction is a process that "augments the space of features" by creating new features from the existing ones in order to fill missing information about relationships between the variables.

Liu, H. and H. Motoda. 1998. *Feature Extraction, Construction and Selection: A Data Mining Perspective.* Kluwer Academic Publishers.

feature construction
Example:

Find the best workers in a company

feature construction
Example:

Find the best workers in a company

- Attributes:

feature construction
Example:

Find the best workers in a company

- Attributes:
  - the tasks, a worker has finished within each month,

feature construction
Example:

Find the best workers in a company

- Attributes:
  - the tasks, a worker has finished within each month,
  - the number of hours each has worked each month,

feature construction
Example:

Find the best workers in a company

- Attributes:
  - the tasks, a worker has finished within each month,
  - the number of hours each has worked each month,
  - the number of hours normally needed to finish each task.

feature construction
Example:

Find the best workers in a company

- Attributes:
  - the tasks, a worker has finished within each month,
  - the number of hours each has worked each month,
  - the number of hours normally needed to finish each task.
- these *contain* information about the efficiency of the worker.

feature construction
Example:

Find the best workers in a company

- Attributes:
  - the tasks, a worker has finished within each month,
  - the number of hours each has worked each month,
  - the number of hours normally needed to finish each task.
- these *contain* information about the efficiency of the worker.
- but instead using these “raw” attributes, it might be more useful to define a new attribute *efficiency*.

feature construction
Example:

Find the best workers in a company

- Attributes:
  - the tasks, a worker has finished within each month,
  - the number of hours each has worked each month,
  - the number of hours normally needed to finish each task.
- these *contain* information about the efficiency of the worker.
- but instead using these “raw” attributes, it might be more useful to define a new attribute *efficiency*.
- efficiency $= \frac{\text{hours actually spent to finish the tasks}}{\text{hours normally needed to finish the tasks}}$

feature construction

Example:

y	x_1	x_2	x_3	x_4
2148.54	18.4105	-66.0605	30.3264	-97.7713
1790.37	42.3706	-71.0270	27.7770	-97.4632
2227.08	43.0214	-76.1977	47.6838	-122.3012
360.94	35.7977	-78.6253	33.7930	-84.5041
1239.83	35.5514	-97.4075	36.7464	-119.6397

feature construction

Example:

y	x_1	x_2	x_3	x_4
2148.54	18.4105	-66.0605	30.3264	-97.7713
1790.37	42.3706	-71.0270	27.7770	-97.4632
2227.08	43.0214	-76.1977	47.6838	-122.3012
360.94	35.7977	-78.6253	33.7930	-84.5041
1239.83	35.5514	-97.4075	36.7464	-119.6397

$$a = \sin^2\left(\frac{x_1 - x_3}{2}\frac{\pi}{180}\right) + \cos\left(\frac{x_1\pi}{180}\right)\cos\left(\frac{x_3\pi}{180}\right)\sin^2\left(\frac{x_2 - x_4}{2}\frac{\pi}{180}\right)$$

$$Y = 2R\arctan\left(\sqrt{a}, \sqrt{1-a}\right)$$

feature construction

Example:

Distance	Latitude 1	Longitude 1	Latitude 2	Longitude 2
2148.54	18.4105	-66.0605	30.3264	-97.7713
1790.37	42.3706	-71.0270	27.7770	-97.4632
2227.08	43.0214	-76.1977	47.6838	-122.3012
360.94	35.7977	-78.6253	33.7930	-84.5041
1239.83	35.5514	-97.4075	36.7464	-119.6397

feature construction

Example:

Distance	Latitude 1	Longitude 1	Latitude 2	Longitude 2
2148.54	San Juan, PR		Austin, TX	
1790.37	Boston, MA		Corpus Christi,TX	
2227.08	Syracuse, NY		Seattle, WA	
360.94	Raleigh, NC		Atlanta, GA	
1239.83	Oklahoma City, OK		Fresno, CA	

feature construction

feature construction

- ratios (e.g., crimes per capita)

feature construction

- ratios (e.g., crimes per capita)
- products (e.g., length $\times$ width = area)

feature construction

- ratios (e.g., crimes per capita)
- products (e.g., length $\times$ width = area)
- aggregation (e.g., averaging, rolling averages)

feature construction

- ratios (e.g., crimes per capita)
- products (e.g., length $\times$ width = area)
- aggregation (e.g., averaging, rolling averages)
- mathematical properties (e.g., sine, cosine)

feature construction

- ratios (e.g., crimes per capita)
- products (e.g., length $\times$ width = area)
- aggregation (e.g., averaging, rolling averages)
- mathematical properties (e.g., sine, cosine)
- lead/lag/differencing variables for time series data (year-on-year percent change)

feature construction

- ratios (e.g., crimes per capita)
- products (e.g., length $\times$ width = area)
- aggregation (e.g., averaging, rolling averages)
- mathematical properties (e.g., sine, cosine)
- lead/lag/differencing variables for time series data (year-on-year percent change)
- binning (create binary indicators, or “levels” corresponding to intervals)

feature extraction

Feature extraction is a process that extracts a set of new features from the original features through some functional mapping.

feature extraction

Feature extraction is a process that extracts a set of new features from the original features through some functional mapping.

- PCA, MDS

Liu, H. and H. Motoda. 1998. *Feature Extraction, Construction and Selection: A Data Mining Perspective.* Kluwer Academic Publishers.

feature extraction

Feature extraction is a process that extracts a set of new features from the original features through some functional mapping.

- PCA, MDS
- LDA

Liu, H. and H. Motoda. 1998. *Feature Extraction, Construction and Selection: A Data Mining Perspective.* Kluwer Academic Publishers.

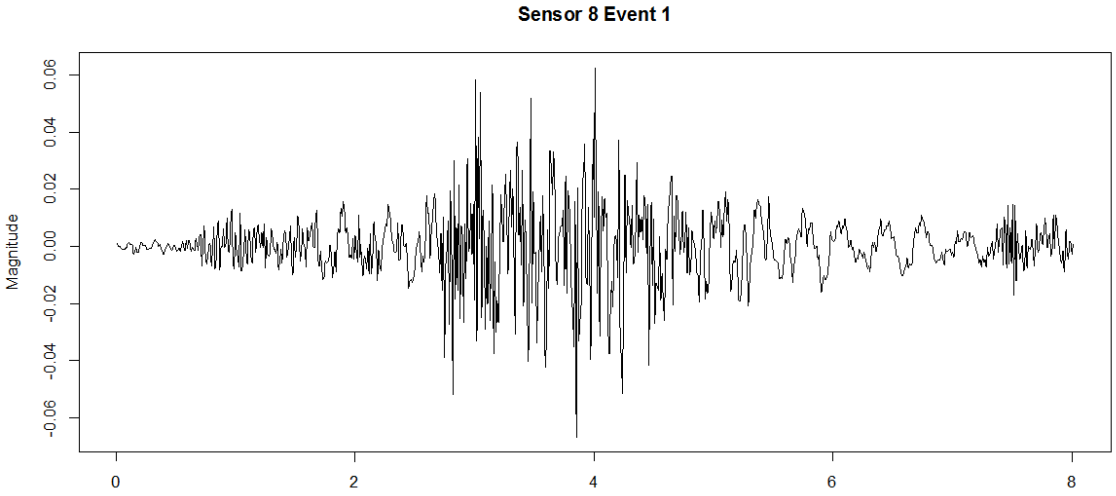
feature extraction

Feature extraction is a process that extracts a set of new features from the original features through some functional mapping.

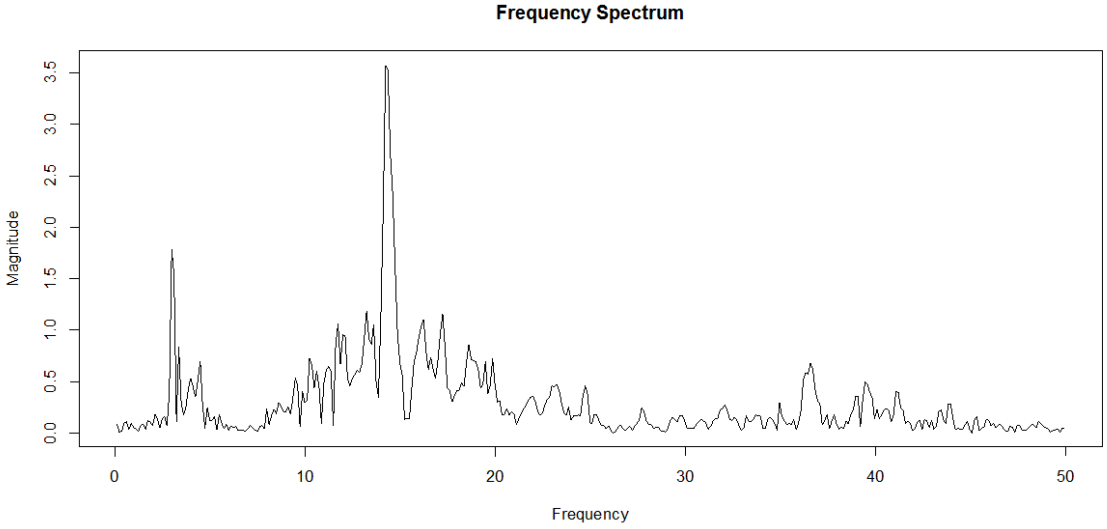
- PCA, MDS
- LDA
- other mappings to new space (e.g., fast fourier transforms)

Liu, H. and H. Motoda. 1998. *Feature Extraction, Construction and Selection: A Data Mining Perspective.* Kluwer Academic Publishers.

mapping to new space



mapping to new space



feature selection

Feature selection refers to techniques to choose a subset of the features (attributes) that is as small as possible and sufficient for the data analysis.

feature selection

Feature selection refers to techniques to choose a subset of the features (attributes) that is as small as possible and sufficient for the data analysis.

- removing (more or less) irrelevant features and
- removing redundant features

feature selection

Feature selection refers to techniques to choose a subset of the features (attributes) that is as small as possible and sufficient for the data analysis.

- removing (more or less) irrelevant features and
- removing redundant features

Basic approach: can be effective; can be disastrous

- find highly correlated variables, eliminate one from analysis
- remove *predictor* attributes NOT related to *target* attribute

feature selection

Feature selection refers to techniques to choose a subset of the features (attributes) that is as small as possible and sufficient for the data analysis.

- removing (more or less) irrelevant features and
- removing redundant features

Basic approach: can be effective; can be disastrous

- find highly correlated variables, eliminate one from analysis
- remove *predictor* attributes NOT related to *target* attribute

Advanced approaches include *forward selection* and *backward elimination*; automatic selection in trees; variable ranking

summary questions

summary questions

1. Do you have domain knowledge?

Guyon, I. and A. Elisseef. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** 1157-1182.

summary questions

1. Do you have domain knowledge?
   - If yes, construct a better set of “ad hoc” features.

Guyon, I. and A. Elisseef. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** 1157-1182.

summary questions

1. Do you have domain knowledge?
   - If yes, construct a better set of “ad hoc” features.
2. Are your features commensurate?

Guyon, I. and A. Elisseef. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** 1157-1182.

summary questions

1. Do you have domain knowledge?
   - If yes, construct a better set of “ad hoc” features.
2. Are your features commensurate?
   - If no, consider normalizing them.

Guyon, I. and A. Elisseef. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** 1157-1182.

summary questions

1. Do you have domain knowledge?
   - If yes, construct a better set of “ad hoc” features.
2. Are your features commensurate?
   - If no, consider normalizing them.
3. Do you suspect interdependence of features?

Guyon, I. and A. Elisseef. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** 1157-1182.

summary questions

1. Do you have domain knowledge?
   - If yes, construct a better set of “ad hoc” features.
2. Are your features commensurate?
   - If no, consider normalizing them.
3. Do you suspect interdependence of features?
   - If yes, expand your feature set by constructing conjunctive features or products of features.

Guyon, I. and A. Elisseef. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** 1157-1182.

summary questions

1. Do you have domain knowledge?
   - If yes, construct a better set of “ad hoc” features.
2. Are your features commensurate?
   - If no, consider normalizing them.
3. Do you suspect interdependence of features?
   - If yes, expand your feature set by constructing conjunctive features or products of features.
4. Are your variables highly skewed?

Guyon, I. and A. Elisseef. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** 1157-1182.

summary questions

1. Do you have domain knowledge?
   - If yes, construct a better set of “ad hoc” features.
2. Are your features commensurate?
   - If no, consider normalizing them.
3. Do you suspect interdependence of features?
   - If yes, expand your feature set by constructing conjunctive features or products of features.
4. Are your variables highly skewed?
   - If yes, consider reducing skew through transformations.

Guyon, I. and A. Elisseef. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** 1157-1182.

4. Do you expect non-linear relationships are important in your analysis?

4. Do you expect non-linear relationships are important in your analysis?
   - If yes, consider using variable binning as well (and products with those bins) to allow for this in regression; or, look at a trees approach to evaluate possible non-linearities.

4. Do you expect non-linear relationships are important in your analysis?
   - If yes, consider using variable binning as well (and products with those bins) to allow for this in regression; or, look at a trees approach to evaluate possible non-linearities.
5. Is the data “compacted” to the right level?

4. Do you expect non-linear relationships are important in your analysis?
   - If yes, consider using variable binning as well (and products with those bins) to allow for this in regression; or, look at a trees approach to evaluate possible non-linearities.
5. Is the data “compacted” to the right level?
   - If no, consider aggregation methods – sums, averages over time, percent change, etc.

4. Do you expect non-linear relationships are important in your analysis?
   - If yes, consider using variable binning as well (and products with those bins) to allow for this in regression; or, look at a trees approach to evaluate possible non-linearities.
5. Is the data “compacted” to the right level?
   - If no, consider aggregation methods – sums, averages over time, percent change, etc.
6. Have other people worked with this problem type?

4. Do you expect non-linear relationships are important in your analysis?
   - If yes, consider using variable binning as well (and products with those bins) to allow for this in regression; or, look at a trees approach to evaluate possible non-linearities.
5. Is the data “compacted” to the right level?
   - If no, consider aggregation methods – sums, averages over time, percent change, etc.
6. Have other people worked with this problem type?
   - If yes, look for standard approaches to pre-processing and transforming data.

4. Do you expect non-linear relationships are important in your analysis?
   - If yes, consider using variable binning as well (and products with those bins) to allow for this in regression; or, look at a trees approach to evaluate possible non-linearities.
5. Is the data “compacted” to the right level?
   - If no, consider aggregation methods – sums, averages over time, percent change, etc.
6. Have other people worked with this problem type?
   - If yes, look for standard approaches to pre-processing and transforming data.
7. Do you have new ideas, time, and computing resources?

4. Do you expect non-linear relationships are important in your analysis?
   - If yes, consider using variable binning as well (and products with those bins) to allow for this in regression; or, look at a trees approach to evaluate possible non-linearities.
5. Is the data “compacted” to the right level?
   - If no, consider aggregation methods – sums, averages over time, percent change, etc.
6. Have other people worked with this problem type?
   - If yes, look for standard approaches to pre-processing and transforming data.
7. Do you have new ideas, time, and computing resources?
   - If yes, try lots of things! The biggest bang for your buck is in feature engineering.