ISE 5103 Intelligent Data Analytics

Data Understanding

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

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

Outline

1. Data Understanding
2. Data Quality
   - Accuracy
   - Completeness
   - Timeliness
3. Visualizations and Data Exploration
4. ggplot2
5. Correlation Analysis
6. Outliers
7. Missing Values

data understanding

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

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?

questions in data understanding

Goal: Gain insight into your data in general and with respect to *your project goals*

Find answers to the questions

- What kind of attributes do we have?
- How is the data quality?
- What is the data granularity?
- Does visualization help?
- Are attributes correlated?
- What about outliers? missing data?

questions in data understanding

Goal: Gain insight into your data in general and with respect to *your project goals*

Find answers to the questions

- What kind of attributes do we have?
- How is the data quality?
- What is the data granularity?
- Does visualization help?
- Are attributes correlated?
- What about outliers? missing data?

questions in data understanding

Goal: Gain insight into your data in general and with respect to *your project goals*

Find answers to the questions

- What kind of attributes do we have?
- How is the data quality?
- What is the data granularity?
- Does visualization help?
- Are attributes correlated?
- What about outliers? missing data?

questions in data understanding

Goal: Gain insight into your data in general and with respect to *your project goals*

Find answers to the questions

- What kind of attributes do we have?
- How is the data quality?
- What is the data granularity?
- Does visualization help?
- Are attributes correlated?
- What about outliers? missing data?

questions in data understanding

Goal: Gain insight into your data in general and with respect to *your project goals*

Find answers to the questions

- What kind of attributes do we have?
- How is the data quality?
- What is the data granularity?
- Does visualization help?
- Are attributes correlated?
- What about outliers? missing data?

questions in data understanding

Goal: Gain insight into your data in general and with respect to *your project goals*

Find answers to the questions

- What kind of attributes do we have?
- How is the data quality?
- What is the data granularity?
- Does visualization help?
- Are attributes correlated?
- What about outliers? missing data?

terminology

We (often) assume that the data set is provided in the form of a simple table.

	attribute$_1$	...	attribute$_m$
record$_1$			
$\vdots$			
record$_n$			

terminology

We (often) assume that the data set is provided in the form of a simple table.

	attribute$_1$	...	attribute$_m$
record$_1$			
$\vdots$			
record$_n$			

- The information in the rows of the table are called **records**, **instances**, **observations**, or **data objects**.
- The columns of the table are alternatively referred to as **attributes**, **features** or **variables**.

terminology

We (often) assume that the data set is provided in the form of a simple table.

	attribute$_1$	...	attribute$_m$
record$_1$			
$\vdots$			
record$_n$			

- The information in the rows of the table are called **records**, **instances**, **observations**, or **data objects**.
- The columns of the table are alternatively referred to as **attributes**, **features** or **variables**.

terminology

An attribute has a **domain**: the set of possible values for the attribute.

terminology

An attribute has a **domain**: the set of possible values for the attribute.

- Scale: *nominal* (or *categorical*), *ordinal*, *numeric*
- Types: e.g., character, number, date, LOB
- Semantic values: meaning of the attribute
- Granularity: detail level of the attribute

terminology

An attribute has a **domain**: the set of possible values for the attribute.

- Scale: *nominal* (or *categorical*), *ordinal*, *numeric*
- Types: e.g., character, number, date, LOB
- Semantic values: meaning of the attribute
- Granularity: detail level of the attribute

terminology

An attribute has a **domain**: the set of possible values for the attribute.

- Scale: *nominal* (or *categorical*), *ordinal*, *numeric*
- Types: e.g., character, number, date, LOB
- Semantic values: meaning of the attribute
- Granularity: detail level of the attribute

terminology

An attribute has a **domain**: the set of possible values for the attribute.

- Scale: *nominal* (or *categorical*), *ordinal*, *numeric*
- Types: e.g., character, number, date, LOB
- Semantic values: meaning of the attribute
- Granularity: detail level of the attribute

Outline

1. Data Understanding
2. **Data Quality**
   - Accuracy
   - Completeness
   - Timeliness
3. Visualizations and Data Exploration
4. ggplot2
5. Correlation Analysis
6. Outliers
7. Missing Values

data quality

Low data quality makes it impossible to trust analysis results: **“Garbage in, garbage out”**

data quality

Low data quality makes it impossible to trust analysis results: **“Garbage in, garbage out”**

data quality

Low data quality makes it impossible to trust analysis results: **"Garbage in, garbage out"**

- Accuracy
- Completeness
- Timeliness

accuracy

Accuracy: Closeness between the recorded and true value.

Sources of inaccuracy:

- noisy measurements
- limited precision
- erroneous measurements (including typos, etc.)
- systematic issues (tying data together from multiple sources)
- rule changes over time

accuracy

Accuracy: Closeness between the recorded and true value.

Sources of inaccuracy:

- noisy measurements
- limited precision
- erroneous measurements (including typos, etc.)
- systematic issues (tying data together from multiple sources)
- rule changes over time

accuracy

Accuracy: Closeness between the recorded and true value.

Sources of inaccuracy:

- noisy measurements
- limited precision
- erroneous measurements (including typos, etc.)
- systematic issues (tying data together from multiple sources)
- rule changes over time

accuracy

Accuracy: Closeness between the recorded and true value.

Sources of inaccuracy:

- noisy measurements
- limited precision
- erroneous measurements (including typos, etc.)
- systematic issues (tying data together from multiple sources)
- rule changes over time

accuracy

Accuracy: Closeness between the recorded and true value.

Sources of inaccuracy:

- noisy measurements
- limited precision
- erroneous measurements (including typos, etc.)
- systematic issues (tying data together from multiple sources)
- rule changes over time

syntactic and semantic accuracy

Syntactic inaccuracy

Entry is not in the domain.

- e.g.: “fmale” in gender field, text in numerical attribute, etc.
- e.g.: negative value in an age field

syntactic and semantic accuracy

Syntactic inaccuracy

Entry is not in the domain.

- e.g.: “fmale” in gender field, text in numerical attribute, etc.
- e.g.: negative value in an age field

Semantic inaccuracy

Entry is in domain, but not correct.

- e.g.: John Smith is female

syntactic and semantic accuracy

Syntactic inaccuracy

Entry is not in the domain.

- e.g.: “fmale” in gender field, text in numerical attribute, etc.
- e.g.: negative value in an age field

Semantic inaccuracy

Entry is in domain, but not correct.

- e.g.: John Smith is female
- Needs more information to validate...

completeness

Types of incompleteness:

completeness

Types of incompleteness:

1. missing records

completeness

Types of incompleteness:

1. missing records
2. missing attribute values

completeness

Types of incompleteness:

1. missing records
2. missing attribute values

completeness

Types of incompleteness:

1. missing records
2. missing attribute values

Two special cases:

completeness

Types of incompleteness:

1. missing records
2. missing attribute values

Two special cases:

- censored data

completeness

Types of incompleteness:

1. missing records
2. missing attribute values

Two special cases:

- censored data
- unbalanced data

completeness: missing records

There may be systematic reasons for why certain records are not represented in the data, e.g.

- historical data may have been purged
- loan repayment data only exists for customers with approved loans (i.e. inherently censored)

completeness: missing records

There may be systematic reasons for why certain records are not represented in the data, e.g.

- historical data may have been purged
- loan repayment data only exists for customers with approved loans (i.e. inherently censored)

completeness: missing values

- Why is the data missing?
- How is the missing data represented?

completeness: missing values

- Why is the data missing?
- How is the missing data represented?

completeness: missing values

- Why is the data missing?
- How is the missing data represented?

completeness: missing values

- Why is the data missing?
- How is the missing data represented?

Note: We will talk a lot about this topic in another lecture!

completeness

Unbalanced data

- The data set might be biased to one type of records.

completeness

Unbalanced data

- The data set might be biased to one type of records.
- e.g.: defective goods are very small fraction of total.
- e.g.: fraudulent transactions small percent of business.

completeness

Censored data

- Censored: A condition in which an observation is only partially known.
- e.g.: in reliability, you might collect times until failure for system components under a certain level of stress; the failure time for components that do not fail are said to be *right censored*

timeliness

- Recency: Is the available data up to date?
- Obsolescence: Does the data reflect the current nature of the problem domain?

timeliness

- Recency: Is the available data up to date?
- Obsolescence: Does the data reflect the current nature of the problem domain?

In some situations, historical data is all-important. In others, it is misleading.

Outline

1. Data Understanding
2. Data Quality
   - Accuracy
   - Completeness
   - Timeliness
3. **Visualizations and Data Exploration**
4. ggplot2
5. Correlation Analysis
6. Outliers
7. Missing Values

Two primary motivations for visualization:

1. Exploring
   - ...you're not exactly sure what the data has to tell you and you're trying to get a sense of the relationships / patterns
   - can be imprecise, noisy
   - should be iterated quickly and experimented on
2. Explaining
   - ...you understand what the data is telling you, and you want to communicate that to someone else
   - ...the ability to pare down information to its simplest form is essential
   - **Short attention spans...**

Two primary motivations for visualization:

1. Exploring
   - ...you're not exactly sure what the data has to tell you and you're trying to get a sense of the relationships / patterns
   - can be imprecise, noisy
   - should be iterated quickly and experimented on
2. Explaining
   - ...you understand what the data is telling you, and you want to communicate that to someone else
   - ...the ability to pare down information to its simplest form is essential
   - **Short attention spans...**

Two primary motivations for visualization:

1. Exploring
   - ...you're not exactly sure what the data has to tell you and you're trying to get a sense of the relationships / patterns
   - can be imprecise, noisy
   - should be iterated quickly and experimented on
2. Explaining
   - ...you understand what the data is telling you, and you want to communicate that to someone else
   - ...the ability to pare down information to its simplest form is essential
   - **Short attention spans...**

Two primary motivations for visualization:

1. Exploring
   - ...you're not exactly sure what the data has to tell you and you're trying to get a sense of the relationships / patterns
   - can be imprecise, noisy
   - should be iterated quickly and experimented on
2. Explaining
   - ...you understand what the data is telling you, and you want to communicate that to someone else
   - ...the ability to pare down information to its simplest form is essential
   - **Short attention spans...**

Two primary motivations for visualization:

1. Exploring
   - ...you're not exactly sure what the data has to tell you and you're trying to get a sense of the relationships / patterns
   - can be imprecise, noisy
   - should be iterated quickly and experimented on
2. Explaining
   - ...you understand what the data is telling you, and you want to communicate that to someone else
   - ...the ability to pare down information to its simplest form is essential
   - **Short attention spans...**

Two primary motivations for visualization:

1. Exploring
   - ...you're not exactly sure what the data has to tell you and you're trying to get a sense of the relationships / patterns
   - can be imprecise, noisy
   - should be iterated quickly and experimented on
2. Explaining
   - ...you understand what the data is telling you, and you want to communicate that to someone else
   - ...the ability to pare down information to its simplest form is essential
   - **Short attention spans...**

Two primary motivations for visualization:

1. Exploring
   - ...you're not exactly sure what the data has to tell you and you're trying to get a sense of the relationships / patterns
   - can be imprecise, noisy
   - should be iterated quickly and experimented on
2. Explaining
   - ...you understand what the data is telling you, and you want to communicate that to someone else
   - ...the ability to pare down information to its simplest form is essential
   - **Short attention spans...**

exploration

What are we looking for?

- Outliers, highly skewed distributions
- Correlations among variables
- Truncated values; inexplicable values
- Potential relationships and patterns

What are we looking for?

- Outliers, highly skewed distributions
- Correlations among variables
- Truncated values; inexplicable values
- Potential relationships and patterns

What are we looking for?

- Outliers, highly skewed distributions
- Correlations among variables
- Truncated values; inexplicable values
- Potential relationships and patterns

What are we looking for?

- Outliers, highly skewed distributions
- Correlations among variables
- Truncated values; inexplicable values
- Potential relationships and patterns

What are we looking for?

- Outliers, highly skewed distributions
- Correlations among variables
- Truncated values; inexplicable values
- Potential relationships and patterns

What are we looking for?

- Outliers, highly skewed distributions
- Correlations among variables
- Truncated values; inexplicable values
- Potential relationships and patterns

The more you play with the data, the more "domain expert" you become; and the more ideas you will have during future analysis steps.

How do we explore?

- Univariate analyses
- Bivariate analyses
- Multivariate analyses

How do we explore?

- Univariate analyses
  - descriptive statistics, frequency tables, histograms and densities, box plots
- Bivariate analyses
- Multivariate analyses

exploration

How do we explore?

- Univariate analyses

  descriptive statistics, frequency tables, histograms and densities, box plots
- Bivariate analyses

  correlations and heatmaps, scatterplots, trends, cross tabulations
- Multivariate analyses

How do we explore?

- Univariate analyses
  - descriptive statistics, frequency tables, histograms and densities, box plots
- Bivariate analyses
  - correlations and heatmaps, scatterplots, trends, cross tabulations
- Multivariate analyses
  - parallel plots, mosaic plots, regression, PCA, MDS, variable clustering

note on visualizations

- When visualizations reveal patterns or exceptions, then there is “something” in the data set.
- When visualizations do not indicate anything specific, there might still be patterns or structures in the data that cannot be revealed by the corresponding (simple) visualization techniques.

data visualisation

Tukey: **There is no excuse for failing to plot and look.**

hidden missing values

The zero values might come from a broken or blocked sensor and might be consider as missing values.

example data set: iris data

iris setosa

iris versicolor

iris virginica

- collected by E. Anderson in 1935
- contains measurements of four real-valued variables: sepal length, sepal widths, petal lengths and petal width
- 150 iris flowers of types Iris Setosa, Iris Versicolor, Iris Virginica (50 each)
- The fifth attribute is the name of the flower type.

cited in many papers...

Primary

- → Fisher,R. "The use of multiple measurements in taxonomic problems" Annual Eugenics, 7, Part II, 179-188 (1936); also in "Contributions to Mathematical Statistics" (John Wiley, NY, 1950).
- → Duda, R., & Hart,P.E. (1973) Pattern Classification and Scene Analysis. (Q327.D83) John Wiley & Sons. ISBN 0-471-22361-1. See page 218.
- → Dasarathy, B. (1980) "Nosing Around the Neighborhood: A New System Structure and Classification Rule for Recognition in Partially Exposed Environments". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-2, No. 1, 67-71.
- → Gates, G. (1972) "The Reduced Nearest Neighbor Rule". IEEE Transactions on Information Theory, May 1972, 431-433.

There are many other works...

S. Kotsiantis and P. Pintelas. Logitboost of Simple Bayesian Classifier. Informatica. 2005.

P. Zhong and M. Fukushima. A Regularized Nonsmooth Newton Method for Multi-class Support Vector Machines. 2005.

I. Fischer and J. Poland. Amplifying the Block Matrix Structure for Spectral Clustering. Telecommunications Lab. 2005.

R. Bouckaert and E. Frank. Evaluating the Replicability of Significance Tests for Comparing Learning Algorithms. PAKDD. 2004.

M. Bilenko, S. Basu, R. Mooney. Integrating constraints and metric learning in semi-supervised clustering. ICML. 2004.

Y. Jiang and Z-H Zhou. Editing Training Data for kNN Classifiers with Neural Network Ensemble. ISNN (1). 2004.

S. Basu. Semi-Supervised Clustering with Limited Background Knowledge. AAAI. 2004.

J. Dy and C. Brodley. Feature Selection for Unsupervised Learning. Journal of Machine Learning Research, 5. 2004.

J. Kubica and A. Moore. Probabilistic Noise Identification and Data Cleaning. ICDM. 2003.

J. Greensmith. New Frontiers For An Artificial Immune System. Digital Media Systems Laboratory HP Laboratories Bristol. 2003.

M. Dash, H. Liu, and P. Scheuermann and Kian-Lee Tan. Fast hierarchical clustering and its validation. Data Knowl. Eng, 44. 2003.

B. Ricks and D. Ventura. Training a Quantum Neural Network. NIPS. 2003.

example data set: iris data

example data set: iris data

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.1	3.5	1.4	0.2	Iris-setosa
...				
...				
5.0	3.3	1.4	0.2	Iris-setosa
7.0	3.2	4.7	1.4	Iris-versicolor
...				
...				
5.1	2.5	3.0	1.1	Iris-versicolor
5.7	2.8	4.1	1.3	Iris-versicolor
...				
...				
5.9	3.0	5.1	1.8	Iris-virginica

boxplots

Iris Sepal Length by Species
Sepal Length (cm)
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
setosa
versicolor
virginica
Species

histograms

Sepal Length
Frequency
0
5
10
15
20
25
30
4
5
6
7
8
Sepal Length
Sepal Width
Frequency
0
5
10
20
30
2.0
2.5
3.0
3.5
4.0
Sepal Width
Petal Length
Frequency
0
10
20
30
1
2
3
4
5
6
7
Petal Length
Petal Width
Frequency
0
5
10
15
20
25
30
35
0.0
0.5
1.0
1.5
2.0
2.5
Petal Width

histograms

densities

Densities for Petal Length of Iris Species
density
0
1
2
2
4
6
Petal Length
Species
setosa
versicolor
virginica

more advanced boxplots

scatter plots

Scatter plots visualize two variables in a two-dimensional plot.

Each axes corresponds to one variable.

Pairs plots depict multiple scatter plots.

scatter plots

Scatter plots can be enriched with additional information.

Color or different symbols can incorporate a third attribute.

scatter plots

The two attributes *petal length* and *width* provide a better **separation** of the species versicolor and virginica than the sepal length and width.

scatter plots

Data objects with the same values cannot be distinguished in a scatter plot. To mitigate this, jitter and alpha-levels can be used.

combinations of visualizations

Sometimes combining several visualizations can convey useful information simply due to the fact that everything is in one place.

This could also be a distraction, so overuse of this is not advised.

However, if used well, they can provide a powerful visual overview of the data.

setosa
versicolor
virginica
Sepal.Length
Sepal.Width
Petal.Length
Petal.Width
Species
Corr:
−0.118
Corr:
0.872
Corr:
0.818
Corr:
−0.428
Corr:
−0.366
Corr:
0.963

parallel coordinates

Parallel coordinates draw the coordinate axes parallel to each other, so that there is no limitation for the the number of axes to be displayed.

For a data object, a polyline is drawn connecting the values of the data object for the attributes on the corresponding axes.

parallel coordinates: iris data

parallel coordinates: iris data

radar plots

Radar plots

- based on a similar idea as parallel coordinates
- coordinate axes are drawn in a star-like fashion intersecting at one point
- not very good for large datasets

Radar plot for the Iris data set

I could not figure out how to draw very good radar plots in R... My best attempt is in the accompanying R script code.
Challenge: beat my radar plot attempt in R!

star plots

Star plots

- same as radar plots, except each data object is drawn separately
- only useful for very small datasets

Star plot for the Iris data set

I could not figure out how to draw very good star plots in R... My best attempt is in the accompanying R script code.
Challenge: beat my radar plot attempt in R!

methods for higher-dimensional data

Issue:

- Plots incorporate only two axes (attributes).
- Color, shape, size, 3D axis can be used for more attributes, but can become confusing.
- Often datasets include many more attributes of interest.

methods for higher-dimensional data

Issue:

- Plots incorporate only two axes (attributes).
- Color, shape, size, 3D axis can be used for more attributes, but can become confusing.
- Often datasets include many more attributes of interest.

methods for higher-dimensional data

Issue:

- Plots incorporate only two axes (attributes).
- Color, shape, size, 3D axis can be used for more attributes, but can become confusing.
- Often datasets include many more attributes of interest.

methods for higher-dimensional data

Issue:

- Plots incorporate only two axes (attributes).
- Color, shape, size, 3D axis can be used for more attributes, but can become confusing.
- Often datasets include many more attributes of interest.

methods for higher-dimensional data

Issue:

- Plots incorporate only two axes (attributes).
- Color, shape, size, 3D axis can be used for more attributes, but can become confusing.
- Often datasets include many more attributes of interest.

One possibility is to reduce the dimensions of the data.

It some cases, the dimensions can be reduced without losing much information. This depends in part on attribute correlations.

Outline

1. Data Understanding
2. Data Quality
   - Accuracy
   - Completeness
   - Timeliness
3. Visualizations and Data Exploration
4. ggplot2
5. Correlation Analysis
6. Outliers
7. Missing Values

Much of this material is adapted from *ggplot: Elegant Graphics for Data Analyis* by Hadley Wickham; the Institute for Digital Research and Education at UCLA; and *R for Data Science* by Garrett Grolemund and Hadley Wickham.

Grammar of Graphics

- Wilkinson (2005,2009) created the layered **grammar of graphics** that provides the theoretical foundation of advanced visualizations in R and other statistical tools.
- A grammar of graphics defines the rules of structuring mathematic and aesthetic elements into a meaningful graph.

Grammar of Graphics

- Wilkinson (2005,2009) created the layered **grammar of graphics** that provides the theoretical foundation of advanced visualizations in R and other statistical tools.
- A grammar of graphics defines the rules of structuring mathematic and aesthetic elements into a meaningful graph.

Elements of the grammar of graphics

- **Data**: data that you want to visualize are mapped to aesthetic attributes
- **Geometric objects** (geoms): represent what you actually see on the plot: points, lines, polygons, etc.
- **Statistical transformations** (stats): summarize the data (e.g., mean, confidence intervals, counts) Optional, but very useful.
- **Scales**: map values in the data space to values in an aesthetic space, whether it be colour, or size, or shape. Legends and axes display these mappings.
- **Coordinate system** (coord): the plane on which the data are mapped (usually Cartesian, but can be others)
- **Faceting**: splits the data into subsets to create multiple variations of the same graph.

Elements of the grammar of graphics

- **Data**: data that you want to visualize are mapped to aesthetic attributes
- **Geometric objects** (geoms): represent what you actually see on the plot: points, lines, polygons, etc.
- **Statistical transformations** (stats): summarize the data (e.g., mean, confidence intervals, counts) Optional, but very useful.
- **Scales**: map values in the data space to values in an aesthetic space, whether it be colour, or size, or shape. Legends and axes display these mappings.
- **Coordinate system** (coord): the plane on which the data are mapped (usually Cartesian, but can be others)
- **Faceting**: splits the data into subsets to create multiple variations of the same graph.

Elements of the grammar of graphics

- **Data**: data that you want to visualize are mapped to aesthetic attributes
- **Geometric objects** (geoms): represent what you actually see on the plot: points, lines, polygons, etc.
- **Statistical transformations** (stats): summarize the data (e.g., mean, confidence intervals, counts) Optional, but very useful.
- **Scales**: map values in the data space to values in an aesthetic space, whether it be colour, or size, or shape. Legends and axes display these mappings.
- **Coordinate system** (coord): the plane on which the data are mapped (usually Cartesian, but can be others)
- **Faceting**: splits the data into subsets to create multiple variations of the same graph.

Elements of the grammar of graphics

- **Data**: data that you want to visualize are mapped to aesthetic attributes
- **Geometric objects** (geoms): represent what you actually see on the plot: points, lines, polygons, etc.
- **Statistical transformations** (stats): summarize the data (e.g., mean, confidence intervals, counts) Optional, but very useful.
- **Scales**: map values in the data space to values in an aesthetic space, whether it be colour, or size, or shape. Legends and axes display these mappings.
- **Coordinate system** (coord): the plane on which the data are mapped (usually Cartesian, but can be others)
- **Faceting**: splits the data into subsets to create multiple variations of the same graph.

Elements of the grammar of graphics

- **Data**: data that you want to visualize are mapped to aesthetic attributes
- **Geometric objects** (geoms): represent what you actually see on the plot: points, lines, polygons, etc.
- **Statistical transformations** (stats): summarize the data (e.g., mean, confidence intervals, counts) Optional, but very useful.
- **Scales**: map values in the data space to values in an aesthetic space, whether it be colour, or size, or shape. Legends and axes display these mappings.
- **Coordinate system** (coord): the plane on which the data are mapped (usually Cartesian, but can be others)
- **Faceting**: splits the data into subsets to create multiple variations of the same graph.

Elements of the grammar of graphics

- **Data**: data that you want to visualize are mapped to aesthetic attributes
- **Geometric objects** (geoms): represent what you actually see on the plot: points, lines, polygons, etc.
- **Statistical transformations** (stats): summarize the data (e.g., mean, confidence intervals, counts) Optional, but very useful.
- **Scales**: map values in the data space to values in an aesthetic space, whether it be colour, or size, or shape. Legends and axes display these mappings.
- **Coordinate system** (coord): the plane on which the data are mapped (usually Cartesian, but can be others)
- **Faceting**: splits the data into subsets to create multiple variations of the same graph.

tidyverse

- `tidyverse` is a collection of R packages designed for data science
  - ggplot2: creating graphics based on the Grammar of Graphics
  - dplyr: data management (e.g. filtering, selecting, sorting, processing by group)
  - tidyr: tidying/restructuring data
  - readr: fast and friendly way to read data
  - tibble: modern version of the data frame
  - stringr: string variable processing
  - and others (e.g., haven, readxl, forcats, lubridate, magrittr, etc...)
- https://www.tidyverse.org/

tidyverse

- `tidyverse` is a collection of R packages designed for data science
  - ggplot2: creating graphics based on the Grammar of Graphics
  - dplyr: data management (e.g. filtering, selecting, sorting, processing by group)
  - tidyr: tidying/restructuring data
  - readr: fast and friendly way to read data
  - tibble: modern version of the data frame
  - stringr: string variable processing
  - and others (e.g., haven, readxl, forcats, lubridate, magrittr, etc...)
- https://www.tidyverse.org/

tidyverse

- `tidyverse` is a collection of R packages designed for data science
  - ggplot2: creating graphics based on the Grammar of Graphics
  - dplyr: data management (e.g. filtering, selecting, sorting, processing by group)
  - tidyr: tidying/restructuring data
  - readr: fast and friendly way to read data
  - tibble: modern version of the data frame
  - stringr: string variable processing
  - and others (e.g., haven, readxl, forcats, lubridate, magrittr, etc...)
- https://www.tidyverse.org/

tidyverse

- `tidyverse` is a collection of R packages designed for data science
  - ggplot2: creating graphics based on the Grammar of Graphics
  - dplyr: data management (e.g. filtering, selecting, sorting, processing by group)
  - tidyr: tidying/restructuring data
  - readr: fast and friendly way to read data
  - tibble: modern version of the data frame
  - stringr: string variable processing
  - and others (e.g., haven, readxl, forcats, lubridate, magrittr, etc...)
- https://www.tidyverse.org/

tidyverse

- `tidyverse` is a collection of R packages designed for data science
  - ggplot2: creating graphics based on the Grammar of Graphics
  - dplyr: data management (e.g. filtering, selecting, sorting, processing by group)
  - tidyr: tidying/restructuring data
  - readr: fast and friendly way to read data
  - tibble: modern version of the data frame
  - stringr: string variable processing
  - and others (e.g., haven, readxl, forcats, lubridate, magrittr, etc...)
- https://www.tidyverse.org/

tidyverse

- `tidyverse` is a collection of R packages designed for data science
  - ggplot2: creating graphics based on the Grammar of Graphics
  - dplyr: data management (e.g. filtering, selecting, sorting, processing by group)
  - tidyr: tidying/restructuring data
  - readr: fast and friendly way to read data
  - tibble: modern version of the data frame
  - stringr: string variable processing
  - and others (e.g., haven, readxl, forcats, lubridate, magrittr, etc...)
- https://www.tidyverse.org/

tidyverse

- `tidyverse` is a collection of R packages designed for data science
  - ggplot2: creating graphics based on the Grammar of Graphics
  - dplyr: data management (e.g. filtering, selecting, sorting, processing by group)
  - tidyr: tidying/restructuring data
  - readr: fast and friendly way to read data
  - tibble: modern version of the data frame
  - stringr: string variable processing
  - and others (e.g., haven, readxl, forcats, lubridate, magrittr, etc...)
- https://www.tidyverse.org/

tidyverse

- `tidyverse` is a collection of R packages designed for data science
  - ggplot2: creating graphics based on the Grammar of Graphics
  - dplyr: data management (e.g. filtering, selecting, sorting, processing by group)
  - tidyr: tidying/restructuring data
  - readr: fast and friendly way to read data
  - tibble: modern version of the data frame
  - stringr: string variable processing
  - and others (e.g., haven, readxl, forcats, lubridate, magrittr, etc...)
- https://www.tidyverse.org/

tidyverse

- `tidyverse` is a collection of R packages designed for data science
  - ggplot2: creating graphics based on the Grammar of Graphics
  - dplyr: data management (e.g. filtering, selecting, sorting, processing by group)
  - tidyr: tidying/restructuring data
  - readr: fast and friendly way to read data
  - tibble: modern version of the data frame
  - stringr: string variable processing
  - and others (e.g., haven, readxl, forcats, lubridate, magrittr, etc...)
- https://www.tidyverse.org/

ggplot2

- All graphics begin with specifying the `ggplot()` function
- First need the "default" data
  - must be a data frame
  - may use multiple data frames (in other layers)
- The first layer is the aesthetics layer
- Subsequent layers are added producing the actual graphical elements of the plot

ggplot2

- All graphics begin with specifying the `ggplot()` function
- First need the "default" data
  - must be a data frame
  - may use multiple data frames (in other layers)
- The first layer is the aesthetics layer
- Subsequent layers are added producing the actual graphical elements of the plot

ggplot2

- All graphics begin with specifying the `ggplot()` function
- First need the "default" data
  - must be a data frame
  - may use multiple data frames (in other layers)
- The first layer is the aesthetics layer
- Subsequent layers are added producing the actual graphical elements of the plot

ggplot2

- All graphics begin with specifying the `ggplot()` function
- First need the "default" data
  - must be a data frame
  - may use multiple data frames (in other layers)
- The first layer is the aesthetics layer
- Subsequent layers are added producing the actual graphical elements of the plot

ggplot2

- All graphics begin with specifying the `ggplot()` function
- First need the "default" data
  - must be a data frame
  - may use multiple data frames (in other layers)
- The first layer is the aesthetics layer
- Subsequent layers are added producing the actual graphical elements of the plot

ggplot2

- All graphics begin with specifying the `ggplot()` function
- First need the "default" data
  - must be a data frame
  - may use multiple data frames (in other layers)
- The first layer is the aesthetics layer
- Subsequent layers are added producing the actual graphical elements of the plot

ggplot2

- All graphics begin with specifying the `ggplot()` function
- First need the "default" data
  - must be a data frame
  - may use multiple data frames (in other layers)
- The first layer is the aesthetics layer
- Subsequent layers are added producing the actual graphical elements of the plot

ggplot2

- All graphics begin with specifying the `ggplot()` function
- First need the "default" data
  - must be a data frame
  - may use multiple data frames (in other layers)
- The first layer is the aesthetics layer
- Subsequent layers are added producing the actual graphical elements of the plot

```
ggplot(data = <DATA>) +
    <GEOM_FUNCTION>(
      mapping = aes(<MAPPINGS>),
      stat = <STAT>,
      position = <POSITION>) +
    <COORDINATE_FUNCTION> +
    <FACET_FUNCTION>
```

example

Look at the `Milk` dataset from the `nlme` package: `data(Milk, package="nlme")`

example

Look at the `Milk` dataset from the `nlme` package: `data(Milk, package="nlme")`
Contains 1337 rows of the following 4 columns:

- protein: numeric, protein content of milk
- Time: numeric, time since calving
- Cow: ordered factor, cow id
- Diet: factor, diet of cow, 3 levels = barley, barley+lupins, lupins

example

Look at the `Milk` dataset from the `nlme` package: `data(Milk, package="nlme")`
Contains 1337 rows of the following 4 columns:

- protein: numeric, protein content of milk
- Time: numeric, time since calving
- Cow: ordered factor, cow id
- Diet: factor, diet of cow, 3 levels = barley, barley+lupins, lupins

	protein	Time	Cow	Diet
1	3.63	1	B01	barley
2	3.57	2	B01	barley
3	3.47	3	B01	barley
4	3.65	4	B01	barley
5	3.89	5	B01	barley

```
ggplot(data = Milk, aes(x=Time, y=protein))
```

- data set is `Milk`
- inside of the function `aes()`: `Time` mapped to x-axis; `protein` mapped to y-axis
- but no shapes yet

`ggplot(data = Milk, aes(x=Time, y=protein))`

- data set is `Milk`
- inside of the function `aes()`: `Time` mapped to x-axis; `protein` mapped to y-axis
- but no shapes yet

`ggplot(data = Milk, aes(x=Time, y=protein))`

- data set is `Milk`
- inside of the function `aes()`: `Time` mapped to x-axis; `protein` mapped to y-axis
- but no shapes yet

Mappings specified in top line `aes()` function serve as defaults – they are inherited by subsequent layers and do not need to respecified.

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
      geom_point()
```

Mappings specified in top line `aes()` function serve as defaults – they are inherited by subsequent layers and do not need to respecified.

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
      geom_point()
```

Additional `aes()` specifications in other layers override default aesthetics for *that layer only*.

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
      geom_point() +
      geom_boxplot(aes(color=Diet))
```

Additional `aes()` specifications in other layers override default aesthetics for *that layer only*.

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
    geom_point() +
    geom_boxplot(aes(color=Diet))
```

Example aesthetics

- x: positioning along x-axis
- y: positioning along y-axis
- color: color of objects
- fill: fill color of objects
- alpha: transparency of objects (value between 0, transparent, and 1, opaque)
- linetype: how lines should be drawn (solid, dashed, dotted, etc.)
- shape: shape of markers in scatter plots
- size: how large objects appear

Example aesthetics

- x: positioning along x-axis
- y: positioning along y-axis
- color: color of objects
- fill: fill color of objects
- alpha: transparency of objects (value between 0, transparent, and 1, opaque)
- linetype: how lines should be drawn (solid, dashed, dotted, etc.)
- shape: shape of markers in scatter plots
- size: how large objects appear

Example aesthetics

- x: positioning along x-axis
- y: positioning along y-axis
- color: color of objects
- fill: fill color of objects
- alpha: transparency of objects (value between 0, transparent, and 1, opaque)
- linetype: how lines should be drawn (solid, dashed, dotted, etc.)
- shape: shape of markers in scatter plots
- size: how large objects appear

Example aesthetics

- x: positioning along x-axis
- y: positioning along y-axis
- color: color of objects
- fill: fill color of objects
- alpha: transparency of objects (value between 0, transparent, and 1, opaque)
- linetype: how lines should be drawn (solid, dashed, dotted, etc.)
- shape: shape of markers in scatter plots
- size: how large objects appear

Example aesthetics

- x: positioning along x-axis
- y: positioning along y-axis
- color: color of objects
- fill: fill color of objects
- alpha: transparency of objects (value between 0, transparent, and 1, opaque)
- linetype: how lines should be drawn (solid, dashed, dotted, etc.)
- shape: shape of markers in scatter plots
- size: how large objects appear

Example aesthetics

- x: positioning along x-axis
- y: positioning along y-axis
- color: color of objects
- fill: fill color of objects
- alpha: transparency of objects (value between 0, transparent, and 1, opaque)
- linetype: how lines should be drawn (solid, dashed, dotted, etc.)
- shape: shape of markers in scatter plots
- size: how large objects appear

Example aesthetics

- x: positioning along x-axis
- y: positioning along y-axis
- color: color of objects
- fill: fill color of objects
- alpha: transparency of objects (value between 0, transparent, and 1, opaque)
- linetype: how lines should be drawn (solid, dashed, dotted, etc.)
- shape: shape of markers in scatter plots
- size: how large objects appear

Example aesthetics

- x: positioning along x-axis
- y: positioning along y-axis
- color: color of objects
- fill: fill color of objects
- alpha: transparency of objects (value between 0, transparent, and 1, opaque)
- linetype: how lines should be drawn (solid, dashed, dotted, etc.)
- shape: shape of markers in scatter plots
- size: how large objects appear

Map aesthetics to **variables** *inside* the `aes()` function. When mapped to a variable, the aesthetic will vary as the variable varies.

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
      geom_point(aes(color=Diet))
```

Map aesthetics to **variables** *inside* the `aes()` function. When mapped to a variable, the aesthetic will vary as the variable varies.

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
    geom_point(aes(color=Diet))
```

Set aesthetics to a **constant** *outside* the `aes()` function.

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
      geom_point(color="blue")
```

Set aesthetics to a **constant** *outside* the `aes()` function.

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
    geom_point(color="blue")
```

Layers

- Graphs in ggplot2 are built layer-by-layer
- More layers are added with the character "+"
- Layers consists of graphics produced by either a geom or stat element
- Layers inherit the aesthetics from the `ggplot()` function, but they can be respecified for each layer

Layers

- Graphs in ggplot2 are built layer-by-layer
- More layers are added with the character "+"
- Layers consists of graphics produced by either a geom or stat element
- Layers inherit the aesthetics from the `ggplot()` function, but they can be respecified for each layer

Layers

- Graphs in ggplot2 are built layer-by-layer
- More layers are added with the character "+"
- Layers consists of graphics produced by either a geom or stat element
- Layers inherit the aesthetics from the `ggplot()` function, but they can be respecified for each layer

Layers

- Graphs in ggplot2 are built layer-by-layer
- More layers are added with the character "+"
- Layers consists of graphics produced by either a geom or stat element
- Layers inherit the aesthetics from the `ggplot()` function, but they can be respecified for each layer

Layers example

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
        geom_point()
```

Layers example

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
    geom_point() +
    geom_line(aes(group=Cow))
```

Layers example

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
        geom_point() +
        geom_line(aes(group=Cow)) +
        geom_smooth(color="white",size=2)
```

Layers example

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
    geom_point() +
    geom_line(aes(group=Cow)) +
    geom_smooth(color="white",size=2) +
  geom_smooth(aes(color=Diet),method="lm",
        se=FALSE)
```

Layers example

```
p <- ggplot(data = Milk, aes(x=Time, y=protein)) +
        geom_point(alpha=0.1)

p + geom_boxplot(aes(group=Time),alpha=0.1,fill="green")
```

Example geoms

There are many additional geoms to choose from, e.g.,

- `geom_bar()`: bars with bases on the x-axis
- `geom_boxplot()`: boxes-and-whiskers
- `geom_errorbar()`: T-shaped error bars
- `geom_histogram()`: histogram
- `geom_density()`: smoothed density plot
- `geom_bin2d()`: heatmap of 2d bin counts
- `geom_jitter()`: jittered points
- `geom_ribbon()`: bands spanning y-values across a range of x-values
- `geom_smooth()`: smoothed conditional means (e.g. loess smooth)

Example geoms

There are many additional geoms to choose from, e.g.,

- `geom_bar()`: bars with bases on the x-axis
- `geom_boxplot()`: boxes-and-whiskers
- `geom_errorbar()`: T-shaped error bars
- `geom_histogram()`: histogram
- `geom_density()`: smoothed density plot
- `geom_bin2d()`: heatmap of 2d bin counts
- `geom_jitter()`: jittered points
- `geom_ribbon()`: bands spanning y-values across a range of x-values
- `geom_smooth()`: smoothed conditional means (e.g. loess smooth)

Example geoms

There are many additional geoms to choose from, e.g.,

- `geom_bar()`: bars with bases on the x-axis
- `geom_boxplot()`: boxes-and-whiskers
- `geom_errorbar()`: T-shaped error bars
- `geom_histogram()`: histogram
- `geom_density()`: smoothed density plot
- `geom_bin2d()`: heatmap of 2d bin counts
- `geom_jitter()`: jittered points
- `geom_ribbon()`: bands spanning y-values across a range of x-values
- `geom_smooth()`: smoothed conditional means (e.g. loess smooth)

Example geoms

There are many additional geoms to choose from, e.g.,

- `geom_bar()`: bars with bases on the x-axis
- `geom_boxplot()`: boxes-and-whiskers
- `geom_errorbar()`: T-shaped error bars
- `geom_histogram()`: histogram
- `geom_density()`: smoothed density plot
- `geom_bin2d()`: heatmap of 2d bin counts
- `geom_jitter()`: jittered points
- `geom_ribbon()`: bands spanning y-values across a range of x-values
- `geom_smooth()`: smoothed conditional means (e.g. loess smooth)

Example geoms

There are many additional geoms to choose from, e.g.,

- `geom_bar()`: bars with bases on the x-axis
- `geom_boxplot()`: boxes-and-whiskers
- `geom_errorbar()`: T-shaped error bars
- `geom_histogram()`: histogram
- `geom_density()`: smoothed density plot
- `geom_bin2d()`: heatmap of 2d bin counts
- `geom_jitter()`: jittered points
- `geom_ribbon()`: bands spanning y-values across a range of x-values
- `geom_smooth()`: smoothed conditional means (e.g. loess smooth)

Example geoms

There are many additional geoms to choose from, e.g.,

- `geom_bar()`: bars with bases on the x-axis
- `geom_boxplot()`: boxes-and-whiskers
- `geom_errorbar()`: T-shaped error bars
- `geom_histogram()`: histogram
- `geom_density()`: smoothed density plot
- `geom_bin2d()`: heatmap of 2d bin counts
- `geom_jitter()`: jittered points
- `geom_ribbon()`: bands spanning y-values across a range of x-values
- `geom_smooth()`: smoothed conditional means (e.g. loess smooth)

Example geoms

There are many additional geoms to choose from, e.g.,

- `geom_bar()`: bars with bases on the x-axis
- `geom_boxplot()`: boxes-and-whiskers
- `geom_errorbar()`: T-shaped error bars
- `geom_histogram()`: histogram
- `geom_density()`: smoothed density plot
- `geom_bin2d()`: heatmap of 2d bin counts
- `geom_jitter()`: jittered points
- `geom_ribbon()`: bands spanning y-values across a range of x-values
- `geom_smooth()`: smoothed conditional means (e.g. loess smooth)

Example geoms

There are many additional geoms to choose from, e.g.,

- `geom_bar()`: bars with bases on the x-axis
- `geom_boxplot()`: boxes-and-whiskers
- `geom_errorbar()`: T-shaped error bars
- `geom_histogram()`: histogram
- `geom_density()`: smoothed density plot
- `geom_bin2d()`: heatmap of 2d bin counts
- `geom_jitter()`: jittered points
- `geom_ribbon()`: bands spanning y-values across a range of x-values
- `geom_smooth()`: smoothed conditional means (e.g. loess smooth)

Example geoms

There are many additional geoms to choose from, e.g.,

- `geom_bar()`: bars with bases on the x-axis
- `geom_boxplot()`: boxes-and-whiskers
- `geom_errorbar()`: T-shaped error bars
- `geom_histogram()`: histogram
- `geom_density()`: smoothed density plot
- `geom_bin2d()`: heatmap of 2d bin counts
- `geom_jitter()`: jittered points
- `geom_ribbon()`: bands spanning y-values across a range of x-values
- `geom_smooth()`: smoothed conditional means (e.g. loess smooth)

geoms

- Each geom is defined by aesthetics required for it to be rendered. For example, `geom_point()` requires both *x* and *y*, the minimal specification for a scatterplot.
- Geoms differ in which aesthetics they accept as arguments. For example, `geom_point()` understands and accepts the aesthetic shape, which defines the shapes of points on the graph. In contrast, `geom_bar()` does not accept shape.
- Check the geom function help files for required and understood aesthetics: https://ggplot2.tidyverse.org/reference/#section-layer-geoms

geoms

- Each geom is defined by aesthetics required for it to be rendered. For example, `geom_point()` requires both *x* and *y*, the minimal specification for a scatterplot.
- Geoms differ in which aesthetics they accept as arguments. For example, `geom_point()` understands and accepts the aesthetic shape, which defines the shapes of points on the graph. In contrast, `geom_bar()` does not accept shape.
- Check the geom function help files for required and understood aesthetics: https://ggplot2.tidyverse.org/reference/#section-layer-geoms

geoms

- Each geom is defined by aesthetics required for it to be rendered. For example, `geom_point()` requires both x and y, the minimal specification for a scatterplot.
- Geoms differ in which aesthetics they accept as arguments. For example, `geom_point()` understands and accepts the aesthetic shape, which defines the shapes of points on the graph. In contrast, `geom_bar()` does not accept shape.
- Check the geom function help files for required and understood aesthetics: https://ggplot2.tidyverse.org/reference/#section-layer-geoms

Statistics layer

- The stat functions statistically transform data, usually as some form of summary, e.g.,
  - frequencies of values of a variable (histogram, bar graphs)
  - mean, standard error
  - confidence limit
- Each stat function is associated with a default geom, so no geom is required for shapes to be rendered.
- The geom can often be re-specfied in the stat for different shapes

Statistics layer

- The stat functions statistically transform data, usually as some form of summary, e.g.,
  - frequencies of values of a variable (histogram, bar graphs)
  - mean, standard error
  - confidence limit
- Each stat function is associated with a default geom, so no geom is required for shapes to be rendered.
- The geom can often be re-specfied in the stat for different shapes

Statistics layer

- The stat functions statistically transform data, usually as some form of summary, e.g.,
  - frequencies of values of a variable (histogram, bar graphs)
  - mean, standard error
  - confidence limit
- Each stat function is associated with a default geom, so no geom is required for shapes to be rendered.
- The geom can often be re-specfied in the stat for different shapes

Statistics layer

- The stat functions statistically transform data, usually as some form of summary, e.g.,
  - frequencies of values of a variable (histogram, bar graphs)
  - mean, standard error
  - confidence limit
- Each stat function is associated with a default geom, so no geom is required for shapes to be rendered.
- The geom can often be re-specfied in the stat for different shapes

Statistics layer

- The stat functions statistically transform data, usually as some form of summary, e.g.,
  - frequencies of values of a variable (histogram, bar graphs)
  - mean, standard error
  - confidence limit
- Each stat function is associated with a default geom, so no geom is required for shapes to be rendered.
- The geom can often be re-specfied in the stat for different shapes

Statistics layer

- The stat functions statistically transform data, usually as some form of summary, e.g.,
  - frequencies of values of a variable (histogram, bar graphs)
  - mean, standard error
  - confidence limit
- Each stat function is associated with a default geom, so no geom is required for shapes to be rendered.
- The geom can often be re-specfied in the stat for different shapes

`stat_bin()` transforms a continuous variable mapped to x into bins (intervals) of counts. Its default geom is bar, producing a histogram:

```
ggplot(Milk, aes(x=protein)) +
      stat_bin()
```

`stat_summary()` applies a summary function to *y* at each value or interval of *x*.

```
ggplot(Milk, aes(x=Time, y=protein)) +
    stat_summary()
```

- default function is `mean_se`, which returns the mean ± standard error
- default geom is `pointrange`: dot at mean of *y* (for that *x*) and lines to mean ± s.e.
- other options including user-defined

`stat_summary()` applies a summary function to *y* at each value or interval of *x*.

```
ggplot(Milk, aes(x=Time, y=protein)) +
    stat_summary()
```

- default function is `mean_se`, which returns the mean ± standard error
- default geom is `pointrange`: dot at mean of *y* (for that *x*) and lines to mean ± s.e.
- other options including user-defined

`stat_summary()` applies a summary function to *y* at each value or interval of *x*.

```
ggplot(Milk, aes(x=Time, y=protein)) +
    stat_summary()
```

- default function is `mean_se`, which returns the mean ± standard error
- default geom is `pointrange`: dot at mean of *y* (for that *x*) and lines to mean ± s.e.
- other options including user-defined

```
iqr <- function(x, ...) {
      qs <- quantile(as.numeric(x), c(0.25, 0.75), na.rm = T)
      names(qs) <- c("ymin", "ymax")
      qs
}

ggplot(data = Milk, aes(x=Time, y=protein)) +
       stat_summary(fun.data = "iqr", geom="ribbon",alpha=0.1) +
        geom_smooth()
```

Scales

- Scales define which aesthetic values are mapped to data values, e.g.,
  - what colors are mapped to 1,2,3?
  - what shapes are mapped to 0,1?

Scales

- Scales define which aesthetic values are mapped to data values, e.g.,
  - what colors are mapped to 1,2,3?
  - what shapes are mapped to 0,1?
- Needed for every aesthetic used on a plot

Scales

- Scales define which aesthetic values are mapped to data values, e.g.,
  - what colors are mapped to 1,2,3?
  - what shapes are mapped to 0,1?
- Needed for every aesthetic used on a plot
- Each scale operates across all the data in the plot, ensuring a consistent mapping from data to aesthetics

Scales

- Scales define which aesthetic values are mapped to data values, e.g.,
  - what colors are mapped to 1,2,3?
  - what shapes are mapped to 0,1?
- Needed for every aesthetic used on a plot
- Each scale operates across all the data in the plot, ensuring a consistent mapping from data to aesthetics
- Scale functions like `scale_{AES}_manual()` specify which visual elements will appear on the graph

Scales

- Scales define which aesthetic values are mapped to data values, e.g.,
  - what colors are mapped to 1,2,3?
  - what shapes are mapped to 0,1?
- Needed for every aesthetic used on a plot
- Each scale operates across all the data in the plot, ensuring a consistent mapping from data to aesthetics
- Scale functions like `scale_{AES}_manual()` specify which visual elements will appear on the graph

Example scale_shape_manual()

```
ggplot(Milk, aes(x=Time, y=protein, shape=Diet)) +
    geom_point() +
    scale_shape_manual(values=c(5, 3, 8))
```

Colors

ggplot2 provides many scale functions designed to make controlling color scales simpler

- `scale_fill_hue()`: evenly spaced hues, default used for factor variables
- `scale_fill_brewer()`: sequential, diverging or qualitative color schemes, originally intended to display factor levels on a map.

Colors

ggplot2 provides many scale functions designed to make controlling color scales simpler

- `scale_fill_hue()`: evenly spaced hues, default used for factor variables
- `scale_fill_brewer()`: sequential, diverging or qualitative color schemes, originally intended to display factor levels on a map.

Colors

ggplot2 provides many scale functions designed to make controlling color scales simpler

- `scale_fill_hue()`: evenly spaced hues, default used for factor variables
- `scale_fill_brewer()`: sequential, diverging or qualitative color schemes, originally intended to display factor levels on a map.

Colors

ggplot2 provides many scale functions designed to make controlling color scales simpler

- `scale_fill_hue()`: evenly spaced hues, default used for factor variables
- `scale_fill_brewer()`: sequential, diverging or qualitative color schemes, originally intended to display factor levels on a map.

Example scale_color_brewer()

```
ggplot(data = Milk, aes(x=Time, y=protein,color=Diet)) +
    geom_point() +
    scale_color_brewer(type="qual")
```

Common scale functions

Some common changes to scales are handled by convenience functions:

- `lims`, `xlim`, `ylim`: set axis limits
- `xlab`, `ylab`, `ggtitle`, `labs`: give labels (titles) to x-axis, y-axis, or graph; labs can set labels for all aesthetics and title

Common scale functions

Some common changes to scales are handled by convenience functions:

- `lims`, `xlim`, `ylim`: set axis limits
- `xlab`, `ylab`, `ggtitle`, `labs`: give labels (titles) to x-axis, y-axis, or graph; labs can set labels for all aesthetics and title

Coords

Default coordinate system is the Cartesian, but there are others (e.g., polar, maps). Can use `coords` to flip the coordinate system, e.g.:

```
ggplot(data = Milk, aes(x=Time, y=protein)) +
        geom_boxplot(aes(group=Time)) +
        coord_flip()
```

Faceting

Faceting subsets data to create multiples of a plot

- `facet_wrap()`: wraps a ribbon of plots into multirow panel of plots
- `facet_grid()`: allows direct specification of which variables are used to split plots along rows and columns

facet_wrap()

- for single variables
- can specify number of rows
- first parameter is an R formula

```
ggplot(Milk, aes(x=protein, color=Diet)) +
    geom_density() +
    facet_wrap(~Time)
```

facet_grid()

- for one or two variables
- row splitting before ~; column after
- . specifies no faceting along a dimension

```
ggplot(Milk, aes(x=protein)) +
    geom_density() +
    facet_grid(Diet~Time)
```

Themes

Themes control elements of the graph not related to the data, e.g.,

- background color
- size of fonts
- gridlines
- color of labels

These data-independent graph elements are known as theme elements.
https://ggplot2.tidyverse.org/reference/theme.html

Example: sleepstudy dataset visual analysis

- data(sleepstudy,package="lme4")
- study on sleep deprivation: subjects were restricted to 3 hours of sleep per night, their reaction times on a series of tests were recorded each subsequent day
- consists of 180 observations of 18 Subjects (repeated measures)
- three variables in sleepstudy:
  - Reaction: reaction time (ms)
  - Days: number of days of sleep deprivation, 0 to 9
  - Subject: subject id

```
> str(sleepstudy)
'data.frame': 180 obs. of  3 variables:
$ Reaction: num  250 259 251 321 357 ...
$ Days    : num  0 1 2 3 4 5 6 7 8 9 ...
$ Subject : Factor w/ 18 levels "308","309","310",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Example: sleepstudy dataset visual analysis

- data(sleepstudy,package="lme4")
- study on sleep deprivation: subjects were restricted to 3 hours of sleep per night, their reaction times on a series of tests were recorded each subsequent day
- consists of 180 observations of 18 Subjects (repeated measures)
- three variables in sleepstudy:
  - Reaction: reaction time (ms)
  - Days: number of days of sleep deprivation, 0 to 9
  - Subject: subject id

```
> str(sleepstudy)
'data.frame': 180 obs. of  3 variables:
$ Reaction: num  250 259 251 321 357 ...
$ Days    : num  0 1 2 3 4 5 6 7 8 9 ...
$ Subject : Factor w/ 18 levels "308","309","310",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Example: sleepstudy dataset visual analysis

- data(sleepstudy,package="lme4")
- study on sleep deprivation: subjects were restricted to 3 hours of sleep per night, their reaction times on a series of tests were recorded each subsequent day
- consists of 180 observations of 18 Subjects (repeated measures)
- three variables in sleepstudy:
  - Reaction: reaction time (ms)
  - Days: number of days of sleep deprivation, 0 to 9
  - Subject: subject id

```
> str(sleepstudy)
'data.frame': 180 obs. of  3 variables:
$ Reaction: num  250 259 251 321 357 ...
$ Days    : num  0 1 2 3 4 5 6 7 8 9 ...
$ Subject : Factor w/ 18 levels "308","309","310",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Example: sleepstudy dataset visual analysis

- data(sleepstudy,package="lme4")
- study on sleep deprivation: subjects were restricted to 3 hours of sleep per night, their reaction times on a series of tests were recorded each subsequent day
- consists of 180 observations of 18 Subjects (repeated measures)
- three variables in sleepstudy:
  - Reaction: reaction time (ms)
  - Days: number of days of sleep deprivation, 0 to 9
  - Subject: subject id

```
> str(sleepstudy)
'data.frame': 180 obs. of  3 variables:
$ Reaction: num  250 259 251 321 357 ...
$ Days    : num  0 1 2 3 4 5 6 7 8 9 ...
$ Subject : Factor w/ 18 levels "308","309","310",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Example: sleepstudy dataset visual analysis

- data(sleepstudy,package="lme4")
- study on sleep deprivation: subjects were restricted to 3 hours of sleep per night, their reaction times on a series of tests were recorded each subsequent day
- consists of 180 observations of 18 Subjects (repeated measures)
- three variables in sleepstudy:
  - Reaction: reaction time (ms)
  - Days: number of days of sleep deprivation, 0 to 9
  - Subject: subject id

```
> str(sleepstudy)
'data.frame': 180 obs. of  3 variables:
$ Reaction: num  250 259 251 321 357 ...
$ Days    : num  0 1 2 3 4 5 6 7 8 9 ...
$ Subject : Factor w/ 18 levels "308","309","310",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Example: sleepstudy dataset visual analysis

- data(sleepstudy,package="lme4")
- study on sleep deprivation: subjects were restricted to 3 hours of sleep per night, their reaction times on a series of tests were recorded each subsequent day
- consists of 180 observations of 18 Subjects (repeated measures)
- three variables in sleepstudy:
  - Reaction: reaction time (ms)
  - Days: number of days of sleep deprivation, 0 to 9
  - Subject: subject id

```
> str(sleepstudy)
'data.frame': 180 obs. of  3 variables:
$ Reaction: num  250 259 251 321 357 ...
$ Days    : num  0 1 2 3 4 5 6 7 8 9 ...
$ Subject : Factor w/ 18 levels "308","309","310",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Example: sleepstudy dataset visual analysis

- data(sleepstudy,package="lme4")
- study on sleep deprivation: subjects were restricted to 3 hours of sleep per night, their reaction times on a series of tests were recorded each subsequent day
- consists of 180 observations of 18 Subjects (repeated measures)
- three variables in sleepstudy:
  - Reaction: reaction time (ms)
  - Days: number of days of sleep deprivation, 0 to 9
  - Subject: subject id

```
> str(sleepstudy)
'data.frame': 180 obs. of  3 variables:
$ Reaction: num  250 259 251 321 357 ...
$ Days    : num  0 1 2 3 4 5 6 7 8 9 ...
$ Subject : Factor w/ 18 levels "308","309","310",..: 1 1 1 1 1 1 1 1 1 1 ...
```

visual exploration of relationship

1. Create a scatter plot of reaction times vs. days.
2. Add a local regression smoothing layer.

```
ggplot(sleepstudy, aes(x=Days, y=Reaction)) +
      geom_point() +
      geom_smooth()
```

- relationship appears linear
- let's use "lm" (linear model) to run a regression model

- relationship appears linear
- let's use "lm" (linear model) to run a regression model

```
> linModel <- lm(data=sleepstudy, Reaction~Days)
> summary(linModel)
```

- relationship appears linear
- let's use "lm" (linear model) to run a regression model

```
> linModel <- lm(data=sleepstudy, Reaction~Days)
> summary(linModel)

Residuals:
Min        1Q    Median        3Q       Max
-110.848  -27.483     1.546    26.142  139.953

Coefficients:
Estimate     Std. Error    t value    Pr(>|t|)
(Intercept)  251.405      6.610      38.033     < 2e-16 ***
Days          10.467      1.238       8.454     9.89e-15 ***

Residual standard error: 47.71 on 178 degrees of freedom
Multiple R-squared:  0.2865,Adjusted R-squared:  0.2825
F-statistic: 71.46 on 1 and 178 DF,  p-value: 9.894e-15
```

visualize model evaluation

1. How do we add the residuals of the model to `sleepstudy`?
2. How do we add the predicted values from the model to `sleepstudy`?
3. And then how to plot residuals vs. fitted values with a loess smoother?

visualize model evaluation

1. How do we add the residuals of the model to `sleepstudy`?
2. How do we add the predicted values from the model to `sleepstudy`?
3. And then how to plot residuals vs. fitted values with a loess smoother?

```
sleepstudy$res <- residuals(linModel)
sleepstudy$fit <- predict(linModel)
```

visual model evaluation

```
ggplot(sleepstudy, aes(x=fit, y=res)) +
        geom_point() +
        geom_smooth()
```

residuals vs. fitted shows linearity seems plausible, though there may be some evidence of heteroscedasticity

Challenge

In addition to linearity and homoscedasticity, linear regression models also assume independence of the observations. Data collected from a repeated measurements design almost certainly violate this assumption. A plot of residuals by `Subject`, with means and standard errors, will assess this assumption.

1. Plot residuals by subject
2. Add a statistical layer to plot means and standard errors
3. Change the color of the "pointrange" geom to red and increase size to 1

visual exploration of relationship

```
ggplot(sleepstudy, aes(x=Subject, y=res)) +
        geom_point() +
        stat_summary(color="red", size=1)
```

Heterogeneity among means is evidence of clustering by Subject, suggesting that the assumption of independence has been violated.

Challenge

Faced with evidence that Subjects have different baseline levels of Reaction time, we further suspect that not all subjects react to sleep deprivation in the same way. In other words, we believe there is heterogeneity in the relationship between days of deprivation and reaction time (i.e., the slope) among subjects.

- Visualize the relationships between `Reaction` and `Days` for different subjects.

```
p <- ggplot(sleepstudy, aes(x=Days, y=Reaction, color=Subject)) +
        geom_point() +
        geom_smooth(se=F,span=1.5)
```

```
p + facet_wrap(~Subject,nrow=3)
```

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Aggregate
  - e.g., average of all reaction times from a given subject
  - the aggregated data would then be independent
  - but this basically throws away really important data and reduces us to merely 18 data points
  - as my teenage daughter says, “ewwww...” *rolls eyes*

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Aggregate
  - e.g., average of all reaction times from a given subject
  - the aggregated data would then be independent
  - but this basically throws away really important data and reduces us to merely 18 data points
  - as my teenage daughter says, “ewwww...” *rolls eyes*

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Aggregate
  - e.g., average of all reaction times from a given subject
  - the aggregated data would then be independent
  - but this basically throws away really important data and reduces us to merely 18 data points
  - as my teenage daughter says, “ewwww...” *rolls eyes*

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Aggregate
  - e.g., average of all reaction times from a given subject
  - the aggregated data would then be independent
  - but this basically throws away really important data and reduces us to merely 18 data points
  - as my teenage daughter says, “ewwww...” *rolls eyes*

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Aggregate
  - e.g., average of all reaction times from a given subject
  - the aggregated data would then be independent
  - but this basically throws away really important data and reduces us to merely 18 data points
  - as my teenage daughter says, “ewwww...” *rolls eyes*

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Individual regression models
  - run 18 separate linear regressions – one for each subject
  - each model is entirely independent and does not take advantage of the information in data from other subject
  - can make the results “noisy” in that the separate estimates are not based on very much data
  - this works, but we can do better

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Individual regression models
  - run 18 separate linear regressions – one for each subject
  - each model is entirely independent and does not take advantage of the information in data from other subject
  - can make the results “noisy” in that the separate estimates are not based on very much data
  - this works, but we can do better

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Individual regression models
  - run 18 separate linear regressions – one for each subject
  - each model is entirely independent and does not take advantage of the information in data from other subject
  - can make the results “noisy” in that the separate estimates are not based on very much data
  - this works, but we can do better

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Individual regression models
  - run 18 separate linear regressions – one for each subject
  - each model is entirely independent and does not take advantage of the information in data from other subject
  - can make the results “noisy” in that the separate estimates are not based on very much data
  - this works, but we can do better

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

- Individual regression models
  - run 18 separate linear regressions – one for each subject
  - each model is entirely independent and does not take advantage of the information in data from other subject
  - can make the results “noisy” in that the separate estimates are not based on very much data
  - this works, but we can do better

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

Mixed models can be thought of as a trade off between these two: individual regressions have many estimates and lots of data, but is noisy. The aggregate is less noisy, but loses important differences by averaging. Mixed models are somewhere in between.

- extension of simple linear models to allow both fixed and random effects
- useful when there is non-independence in the data (e.g., correlated data, repeated measures, hierarchical data)
- essentially allows each subject to have its own intercept and slope

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

Mixed models can be thought of as a trade off between these two: individual regressions have many estimates and lots of data, but is noisy. The aggregate is less noisy, but loses important differences by averaging. Mixed models are somewhere in between.

- extension of simple linear models to allow both fixed and random effects
- useful when there is non-independence in the data (e.g., correlated data, repeated measures, hierarchical data)
- essentially allows each subject to have its own intercept and slope

mixed model

We see evidence of heterogeneity in the effect of days of deprivation across subjects. There are multiple ways to deal with such data.

Mixed models can be thought of as a trade off between these two: individual regressions have many estimates and lots of data, but is noisy. The aggregate is less noisy, but loses important differences by averaging. Mixed models are somewhere in between.

- extension of simple linear models to allow both fixed and random effects
- useful when there is non-independence in the data (e.g., correlated data, repeated measures, hierarchical data)
- essentially allows each subject to have its own intercept and slope

mixed model (also called multi-level models)

mixed models in R

We estimate a mixed model using `lmer()` from the `lme4` package, with a fixed effect of `Days`, and random intercepts and coefficients for `Days` by `Subject`:

```
mixed <- lmer(Reaction ~ Days + (1+Days|Subject), data=sleepstudy)
```

mixed models in R

We estimate a mixed model using `lmer()` from the `lme4` package, with a fixed effect of `Days`, and random intercepts and coefficients for `Days` by `Subject`:

```
mixed <- lmer(Reaction ~ Days + (1+Days|Subject), data=sleepstudy)
```

You can get the new residuals by subject:

```
sleepstudy$res_mix <- residuals(mixed)   #residuals mixed model
```

Challenge

1. Plot new residuals by subject with means and standard errors (in red, size=1)
2. *And...* add to the same plot the previous residuals, means, and standard errors (in blue, size 1)

This should allow for quick comparison to see if we have improved the model or not.

```
ggplot(sleepstudy, aes(x=Subject, y=res_mix)) +
      geom_point() +
      stat_summary(color="red", size=1) +
      stat_summary(aes(y=res), color="blue", size=1)
```

Publication challenge

Now let's make these results really clear and pretty for a wider audience. Specifically, we want to communicate that the relationship between days of sleep deprivation and reaction time is strong but quite different per person.

1. Plot reaction times per days in a **faceted** scatter plot with 6 subjects per row
2. Add the predicted mixed model values as a **line** to each of the facets
3. Add the overall mean intercept and slope to the plot for comparison
4. Use color, shape, linetype, etc. as you please to make it all look good

Hint: to get the overall mean intercept and slope from the mixed model use this code:

```
mean_int <- fixef(mixed)[1]      #mean intercept for the mixed model
mean_slope <- fixef(mixed)[2]    #mean slope for the mixed model
```

Hint: To plot a line with specific slope and intercept, check out `geom_abline()`

```
mean_int <- fixef(mixed)[1]      #mean intercept for the mixed model
mean_slope <- fixef(mixed)[2]    #mean slope for the mixed model

sleepstudy$fit_mix <- predict(mixed)  #fitted values from the mixed model

ggplot(sleepstudy, aes(x=Days, y=Reaction)) +
        geom_point() +
        facet_wrap(~Subject, nrow=3) +
        geom_line(aes(y=fit_mix), size=.75) +
        geom_abline(intercept=mean_int, slope=mean_slope, color="red", size=.75)
```

308
309
310
330
331
332
333
334
335
337
349
350
351
352
369
370
371
372
Reaction
Days
200
300
400
0.0
2.5
5.0
7.5

Publication challenge

Not bad, not bad at all. But, we can still do a few things to clean it up more, e.g.,

- Sort the subjects in order of their slopes and facet on that
- The `Days` axis should be integers
- Improve the axis labeling
- Use different fonts and/or theme

Publication challenge

Getting the subject specific slopes; adding it to the overall mean slope; sorting by result

```
re <- ranef(mixed)$Subject
re$Subject <- factor(rownames(re))
colnames(re)[2] <- "rand_slope"

sleepstudy <- merge(sleepstudy, re, by="Subject")

mean_slope <- fixef(mixed)[2]
sleepstudy$myslope <- sleepstudy$rand_slope + mean_slope
sleepstudy$myslope <- format(sleepstudy$myslope, digits=2)
sleepstudy <- arrange(sleepstudy, myslope, Days)
```

Publication challenge

```
ggplot(sleepstudy, aes(x=Days, y=Reaction)) +
        geom_point(color="white") +
        facet_wrap(~myslope, nrow=3) +
        geom_abline(intercept=mean_int, slope=mean_slope,
        color="black", size=.75) +
        geom_line(aes(y=fit_mix), size=.75, color="white") +
        scale_x_continuous(breaks=c(0,3,6,9)) +
        labs(title="increase in reaction time (ms) per day of deprivation",
                x="days of deprivation",
                y="reaction time (ms)") +
        theme_dark(base_family="serif") +
        theme(plot.title=element_text(size=10),
             axis.title=element_text(size=10),
             strip.text.x=element_text(size=8),
             axis.text=element_text(size=8))
```

increase in reaction time (ms)
per day of deprivation
−0.28
1.85
5.02
5.65
7.40
7.45
9.48
10.20
10.24
11.34
11.54
11.64
11.75
14.00
15.29
17.08
19.10
19.67
reaction time (ms)
200
300
400
0
3
6
9
days of deprivation

Outline

1. Data Understanding
2. Data Quality
   - Accuracy
   - Completeness
   - Timeliness
3. Visualizations and Data Exploration
4. ggplot2
5. **Correlation Analysis**
6. Outliers
7. Missing Values

correlation analysis

How can similiar behaviour of two attributes be measured?

- Pearson's correlation coefficient
- Spearman rank correlation coefficient
- Kendall rank correlation coefficient

Pearson's correlation coefficient

Pearson's correlation coefficient is a measure for a **linear relationship** between two **numerical** attributes X and Y

$$r_{xy} = \frac{\sum_i^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i^n (x_i - \bar{x})^2 \sum_i^n (y_i - \bar{y})^2}}$$

- $-1 \leq r_{xy} \leq 1$
- Larger values of $|r_{xy}| \rightarrow$ stronger linear relationship between the attributes.
- Positive (negative) correlation indicates a line with positive (negative) slope.

Pearson's correlation coefficient

Pearson's correlation coefficient is a measure for a **linear relationship** between two **numerical** attributes X and Y

$$r_{xy} = \frac{\text{cov}(X, Y)}{s_x s_y}$$

- $-1 \leq r_{xy} \leq 1$
- Larger values of $|r_{xy}| \rightarrow$ stronger linear relationship between the attributes.
- Positive (negative) correlation indicates a line with positive (negative) slope.

correlation strenths

Rule of thumb for strength of relationship

.90 to 1.00 (-.90 to -1.00)	Very high positive (negative) correlation
.70 to .90 (-.70 to -.90)	High positive (negative) correlation
.50 to .70 (-.50 to -.70)	Moderate positive (negative) correlation
.30 to .50 (-.30 to -.50)	Low positive (negative) correlation
.00 to .30 (.00 to -.30)	Little if any correlation

Pearson's correlation coefficient

Example: Iris data set

	sepal length	sepal width	petal length	petal width
sepal length	1.000	−0.118	0.872	0.818
sepal width	−0.118	1.000	−0.428	−0.366
petal length	0.872	−0.428	1.000	0.963
petal width	0.818	−0.366	0.963	1.000

Pearson correlation coefficient

- Pearson's correlation coefficient measures *linear* correlation.

Pearson correlation coefficient

- Pearson's correlation coefficient measures *linear* correlation.
- Very commonly used.

Pearson correlation coefficient

- Pearson's correlation coefficient measures *linear* correlation.
- Very commonly used.
- Closely related to linear regression.

Pearson correlation coefficient

- Pearson's correlation coefficient measures *linear* correlation.
- Very commonly used.
- Closely related to linear regression.
- Forms basis for other techniques: PCA

Pearson correlation coefficient

- Pearson's correlation coefficient measures *linear* correlation.
- Very commonly used.
- Closely related to linear regression.
- Forms basis for other techniques: PCA
- Issues:

Pearson correlation coefficient

- Pearson's correlation coefficient measures *linear* correlation.
- Very commonly used.
- Closely related to linear regression.
- Forms basis for other techniques: PCA
- Issues:
  - Sensitive to outlying groups or instances.

Pearson correlation coefficient

- Pearson's correlation coefficient measures *linear* correlation.
- Very commonly used.
- Closely related to linear regression.
- Forms basis for other techniques: PCA
- Issues:
  - Sensitive to outlying groups or instances.
  - Does not work well with non-linear relationships even if they are monotone

Pearson: sensitive to extreme value(s)

Assume: $X \sim N(0,1)$; $Y \sim N(0,1)$

Pearson: sensitive to extreme value(s)

Assume: $X \sim N(0,1)$; $Y \sim N(0,1)$

There should be no correlation for X and Y.

The correlation value should be near 0.

$r = -0.05380358$

Pearson: sensitive to extreme value(s)

Now, add one more point which is extreme in both X and Y...

Pearson: sensitive to extreme value(s)

Now, add one more point which is extreme in both X and Y...

Pearson correlation:

$r = 0.9669555$

Pearson: sensitive to extreme value(s)

And somewhat more likely, and problematic...

Pearson: sensitive to extreme value(s)

And somewhat more likely, and problematic...

Pearson correlation:

$r = 0.984445$

Pearson: inaccurate with non-linear relationships

Y is a monotonically increasing function of X

Pearson: inaccurate with non-linear relationships

Y is a monotonically increasing function of X The correlation value should ideally be 1.

$r = 0.8051005$

rank correlation coefficients

Rank correlation coefficients mitigate these problems by ignoring the exact numerical values of the attributes and considering only the ordering of the values, that is, the *rank* of the values.

Spearman's rank correlation coefficient

Commonly known as "Spearman's rho"

Spearman's rank correlation coefficient:

$$\rho = 1 - 6\frac{\sum_{i=1}^{n}(r(x_i) - r(y_i))^2}{n(n^2 - 1)},$$

where $r(x_i)$ is the rank of value x_i when we sort the list $(x_1, \ldots, x_n)$ in increasing order. $r(y_i)$ is defined analogously.

- When the rankings of the x- and y-values are exactly in the same order, Spearman's rho will yield the value 1.

Spearman's rank correlation coefficient

Commonly known as "Spearman's rho"

Spearman's rank correlation coefficient:

$$\rho = 1 - 6\frac{\sum_{i=1}^{n}(r(x_i) - r(y_i))^2}{n(n^2 - 1)},$$

where $r(x_i)$ is the rank of value x_i when we sort the list $(x_1, \ldots, x_n)$ in increasing order. $r(y_i)$ is defined analogously.

- When the rankings of the x- and y-values are exactly in the same order, Spearman's rho will yield the value 1.
- If they are in reverse order, we will obtain the value -1.

rank examples

$r = 0.9669555$

$\rho = 0.1906465$

rank examples

$r = 0.8051005$

$\rho = 1$

Spearman and Pearson correlation coefficients

Example: Iris data set

Pearson

	sepal length	sepal width	petal length	petal width
sepal length	1.000	−0.118	0.872	0.818
sepal width	−0.118	1.000	−0.428	−0.366
petal length	0.872	−0.428	1.000	0.963
petal width	0.818	−0.366	0.963	1.000

Spearman

	sepal length	sepal width	petal length	petal width
sepal length	1.000	−0.167	0.882	0.834
sepal width	−0.167	1.000	−0.289	−0.289
petal length	0.882	−0.289	1.000	0.938
petal width	0.834	−0.289	0.938	1.000

Sepal.Length
Sepal.Width
Petal.Length
Petal.Width
2.0
3.0
4.0
0.5
1.5
2.5
4.5
6.5
3.5
2.0
7
5
3
1
2.0
0.5
4.5
5.5
6.5
7.5
1
2
3
4
5
6
7

Kendall's rank correlation coefficient

Commonly known as Kendall's tau

Coefficient represents degree of *concordance* between two sets of ranked data.

Kendall's tau is:

$$\tau_a = \frac{C - D}{C + D}$$

where C and D denote the numbers of concordant and discordant pairs, respectively.

concordant and discordant pairs

Concordant pairs: the number of observed ranks below a particular rank which are larger than that particular rank for both variables.

Discordant pairs: the number of observed ranks below a particular rank which are larger than that particular rank for one variable, but lower for the other.

$$\begin{aligned} C &= |\{(i,j) \mid x_i < x_j \text{ and } y_i < y_j\}| \\ D &= |\{(i,j) \mid x_i < x_j \text{ and } y_i > y_j\}| \end{aligned}$$

concordant and discordant pairs

Example

	Alex	Daniela
Chipotle	1	
Fuzzy's		
Coriander Cafe		
Taco Bell		

concordant and discordant pairs

Example

	Alex	Daniela
Chipotle	1	
Fuzzy's	2	
Coriander Cafe		
Taco Bell		

concordant and discordant pairs

Example

	Alex	Daniela
Chipotle	1	
Fuzzy's	2	
Coriander Cafe	3	
Taco Bell		

concordant and discordant pairs

Example

	Alex	Daniela
Chipotle	1	
Fuzzy's	2	
Coriander Cafe	3	
Taco Bell	4	

concordant and discordant pairs

Example

	Alex	Daniela
Chipotle	1	
Fuzzy's	2	
Coriander Cafe	3	1
Taco Bell	4	

concordant and discordant pairs

Example

	Alex	Daniela
Chipotle	1	2
Fuzzy's	2	
Coriander Cafe	3	1
Taco Bell	4	

concordant and discordant pairs

Example

	Alex	Daniela
Chipotle	1	2
Fuzzy's	2	3
Coriander Cafe	3	1
Taco Bell	4	

concordant and discordant pairs

Example

	Alex	Daniela
Chipotle	1	2
Fuzzy's	2	3
Coriander Cafe	3	1
Taco Bell	4	4

concordant and discordant pairs

Example

	Alex	Daniela	Concordant
Chipotle	1	2	
Fuzzy's	2	3	
Coriander Cafe	3	1	
Taco Bell	4	4	

concordant and discordant pairs

Example

	Alex	Daniela	Concordant
Chipotle	1	2	2
Fuzzy's	2	3	
Coriander Cafe	3	1	
Taco Bell	4	4	

concordant and discordant pairs

Example

	Alex	Daniela	Concordant
Chipotle	1	2	2
Fuzzy's	2	3	1
Coriander Cafe	3	1	
Taco Bell	4	4	

concordant and discordant pairs

Example

	Alex	Daniela	Concordant
Chipotle	1	2	2
Fuzzy's	2	3	1
Coriander Cafe	3	1	1
Taco Bell	4	4	

concordant and discordant pairs

Example

	Alex	Daniela	Concordant
Chipotle	1	2	2
Fuzzy's	2	3	1
Coriander Cafe	3	1	1
Taco Bell	4	4	
			C = 4

concordant and discordant pairs

Example

	Alex	Daniela	Concordant	Discordant
Chipotle	1	2	2	
Fuzzy's	2	3	1	
Coriander Cafe	3	1	1	
Taco Bell	4	4		
			C = 4	

concordant and discordant pairs

Example

	Alex	Daniela	Concordant	Discordant
Chipotle	1	2	2	1
Fuzzy's	2	3	1	
Coriander Cafe	3	1	1	
Taco Bell	4	4		

C = 4

concordant and discordant pairs

Example

	Alex	Daniela	Concordant	Discordant
Chipotle	1	2	2	1
Fuzzy's	2	3	1	1
Coriander Cafe	3	1	1	
Taco Bell	4	4		

$C = 4$

concordant and discordant pairs

Example

	Alex	Daniela	Concordant	Discordant
Chipotle	1	2	2	1
Fuzzy's	2	3	1	1
Coriander Cafe	3	1	1	
Taco Bell	4	4		
			C = 4	D = 2

Spearman and Kendall correlation coefficients

Example: Iris data set

Spearman

	sepal length	sepal width	petal length	petal width
sepal length	1.000	−0.167	0.882	0.834
sepal width	−0.167	1.000	−0.289	−0.289
petal length	0.882	−0.289	1.000	0.938
petal width	0.834	−0.289	0.938	1.000

Kendall

	sepal length	sepal width	petal length	petal width
sepal length	1.000	−0.077	0.719	0.655
sepal width	−0.077	1.000	−0.186	−0.157
petal length	0.719	−0.186	1.000	0.807
petal width	0.655	−0.157	0.807	1.000

Kendall and Spearman examples

(adapted from how2stats.com)

Master	Student
1	2
2	1
3	4
4	3
5	6
6	5
7	8
8	7
9	10
10	9
11	12
12	11

		Kendall		Spearman	
Master	Student	C	D	$r(x_i) - r(y_i)$	$(r(x_i) - r(y_i))^2$
1	2				
2	1				
3	4				
4	3				
5	6				
6	5				
7	8				
8	7				
9	10				
10	9				
11	12				
12	11				

		Kendall		Spearman	
Master	Student	C	D	$\lvert r(x_i) - r(y_i) \rvert$	$(r(x_i) - r(y_i))^2$
1	2	10	1	1	1
2	1	10	0	1	1
3	4	8	1	1	1
4	3				
5	6				
6	5				
7	8				
8	7				
9	10				
10	9				
11	12				
12	11				

		Kendall		Spearman	
Master	Student	C	D	$\lvert r(x_i) - r(y_i)\rvert$	$(r(x_i) - r(y_i))^2$
1	2	10	1	1	1
2	1	10	0	1	1
3	4	8	1	1	1
4	3	8	0	1	1
5	6	6	1	1	1
6	5	6	0	1	1
7	8	4	1	1	1
8	7	4	0	1	1
9	10	2	1	1	1
10	9	2	0	1	1
11	12	0	1	1	1
12	11			1	1
		60	6		12

Kendall and Spearman examples

Kendall's tau:

$$\tau_a = \frac{C - D}{C + D} = \frac{60 - 6}{60 + 6} = 0.818$$

Spearman's rho:

$$\rho = 1 - \frac{6 \times 12}{12(12^2 - 1)} = \frac{72}{1716} = 0.958$$

		Kendall		Spearman	
Master	Student	C	D	$\lvert r(x_i) - r(y_i)\rvert$	$(r(x_i) - r(y_i))^2$
1	12	0	11	11	121
2	2	9	1	0	0
3	3	8	1	0	0
4	4	7	1	0	0
5	5	6	1	0	0
6	6	5	1	0	0
7	7	4	1	0	0
8	8	3	1	0	0
9	9	2	1	0	0
10	10	1	1	0	0
11	11	0	1	0	0
12	1			11	121
		45	21		242

Kendall and Spearman examples

Kendall's tau:

$$\tau_a = \frac{C - D}{C + D} = \frac{45 - 21}{45 + 21} = 0.364$$

Spearman's rho:

$$\rho = 1 - \frac{6 \times 242}{12(12^2 - 1)} = 1 - \frac{1452}{1716} = 0.154$$

Kendall's tau and Spearman's rho

Kendall's tau and Spearman's rho

- Represent different effects.

Kendall's tau and Spearman's rho

- Represent different effects.
  - Kendall's τ: proportion of concordant to discordant pairs

Kendall's tau and Spearman's rho

- Represent different effects.
  - Kendall's τ: proportion of concordant to discordant pairs
  - Spearman's ρ: Pearson's correlation applied to ranks

Kendall's tau and Spearman's rho

- Represent different effects.
  - Kendall's τ: proportion of concordant to discordant pairs
  - Spearman's ρ: Pearson's correlation applied to ranks
- Spearman usually (but not always) larger values

Kendall's tau and Spearman's rho

- Represent different effects.
  - Kendall's τ: proportion of concordant to discordant pairs
  - Spearman's ρ: Pearson's correlation applied to ranks
- Spearman usually (but not always) larger values
- Spearman is more sensitive to a few large discrepancies (maybe too sensitive)

R code for correlations

Let `x` and `y` denote numeric R vectors of the same length.

`cor(x,y)`	Pearson (by default)
`cor(x,y,method="spearman")`	Spearman's ρ
`cor(x,y,method="kendall")`	Kendall's τ

Let `data` denote a numeric R data frame or matrix.

`cor(data)`	Correlation matrix (using Pearson)
`heatmap(cor(data))`	Produces heatmap

correlation matrix

The correlation matrix from the R data set `mtcars`:

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
mpg	1	-0.85216	-0.84755	-0.77617	0.681172	-0.86766	0.418684	0.664039	0.599832	0.480285	-0.55093
cyl	-0.85216	1	0.902033	0.832448	-0.69994	0.782496	-0.59124	-0.81081	-0.52261	-0.49269	0.526988
disp	-0.84755	0.902033	1	0.790949	-0.71021	0.88798	-0.4337	-0.71042	-0.59123	-0.55557	0.394977
hp	-0.77617	0.832448	0.790949	1	-0.44876	0.658748	-0.70822	-0.7231	-0.2432	-0.1257	0.749812
drat	0.681172	-0.69994	-0.71021	-0.44876	1	-0.71244	0.091205	0.440279	0.712711	0.69961	-0.09079
wt	-0.86766	0.782496	0.88798	0.658748	-0.71244	1	-0.17472	-0.55492	-0.6925	-0.58329	0.427606
qsec	0.418684	-0.59124	-0.4337	-0.70822	0.091205	-0.17472	1	0.744535	-0.22986	-0.21268	-0.65625
vs	0.664039	-0.81081	-0.71042	-0.7231	0.440278	-0.55492	0.744535	1	0.168345	0.206023	-0.56961
am	0.599832	-0.52261	-0.59123	-0.2432	0.712711	-0.6925	-0.22986	0.168345	1	0.794059	0.057534
gear	0.480285	-0.49269	-0.55557	-0.1257	0.69961	-0.58329	-0.21268	0.206023	0.794059	1	0.274073
carb	-0.55093	0.526988	0.394977	0.749813	-0.09079	0.427606	-0.65625	-0.56961	0.057534	0.274073	1

correlation matrix

The heatmap from `mtcars` correlation matrix:

why do we care about correlations?

- simple method to begin evaluating possibly more complex relationships
- if variables are highly correlated, then in some sense they are redundant
- form the basis of other techniques (dimension reduction, clustering)
- correlation $\neq$ causation, but it is often interesting anyways!
- correlation matrices and *heatmaps* provide a unique overview of the data

why do we care about correlations?

- simple method to begin evaluating possibly more complex relationships
- if variables are highly correlated, then in some sense they are redundant
- form the basis of other techniques (dimension reduction, clustering)
- correlation $\neq$ causation, but it is often interesting anyways!
- correlation matrices and *heatmaps* provide a unique overview of the data

why do we care about correlations?

- simple method to begin evaluating possibly more complex relationships
- if variables are highly correlated, then in some sense they are redundant
- form the basis of other techniques (dimension reduction, clustering)
- correlation $\neq$ causation, but it is often interesting anyways!
- correlation matrices and *heatmaps* provide a unique overview of the data

why do we care about correlations?

- simple method to begin evaluating possibly more complex relationships
- if variables are highly correlated, then in some sense they are redundant
- form the basis of other techniques (dimension reduction, clustering)
- correlation $\neq$ causation, but it is often interesting anyways!
- correlation matrices and *heatmaps* provide a unique overview of the data

why do we care about correlations?

- simple method to begin evaluating possibly more complex relationships
- if variables are highly correlated, then in some sense they are redundant
- form the basis of other techniques (dimension reduction, clustering)
- correlation $\neq$ causation, but it is often interesting anyways!
- correlation matrices and *heatmaps* provide a unique overview of the data

Outline

1. Data Understanding
2. Data Quality
   - Accuracy
   - Completeness
   - Timeliness
3. Visualizations and Data Exploration
4. ggplot2
5. Correlation Analysis
6. **Outliers**
7. Missing Values

Outlier

An observation that appears to deviate markedly from other observations in the sample.

Outlier

An observation that appears to deviate markedly from other observations in the sample.

Detecting outliers is important for the following reasons:

- An outlier may indicate *bad data*.
- Outliers may be due to random variation, or they may indicate something *exceptional* or scientifically interesting.
- Sometimes outliers are the subject of the investigation.

Engineering Statistics Handbook, 'Detection of Outliers': www.itl.nist.gov/div898/handbook/eda/section3/eda35h.htm

Outlier

An observation that appears to deviate markedly from other observations in the sample.

Detecting outliers is important for the following reasons:

- An outlier may indicate *bad data*.
- Outliers may be due to random variation, or they may indicate something *exceptional* or scientifically interesting.
- Sometimes outliers are the subject of the investigation.

Engineering Statistics Handbook, 'Detection of Outliers': www.itl.nist.gov/div898/handbook/eda/section3/eda35h.htm

Outlier

An observation that appears to deviate markedly from other observations in the sample.

Detecting outliers is important for the following reasons:

- An outlier may indicate *bad data*.
- Outliers may be due to random variation, or they may indicate something *exceptional* or scientifically interesting.
- Sometimes outliers are the subject of the investigation.

Engineering Statistics Handbook, ‘Detection of Outliers’: www.itl.nist.gov/div898/handbook/eda/section3/eda35h.htm

three areas relating to outliers

- outlier labeling
- outlier identification
- outlier accommodation

Boris Iglewicz and David Hoaglin (1993), “Volume 16: How to Detect and Handle Outliers”, The ASQC Basic References in Quality Control: Statistical Techniques, Edward F. Mykytka, Ph.D., Editor.

outlier detection

- Outlier detection may depend on distributional assumptions, e.g. normally distributed data.
- Tools to check for normality:
  - Visualizations, e.g. histograms, QQ plots
  - Statistical tests, e.g. Kolmogorov-Smirnov, Shapiro-Wilk, Anderson-Darling test, chi-square goodness of fit test
- If the data is not normally distributed, then:
  - Use caution in the formal/informal tests
  - Consider transforming the data

example: QQ plots

- Quantile-Quantile plots are one visualization technique for “goodness-of-fit”
- The closer the data is to the diagonal line, the better the fit to the theoretical distribution
- In R:

`qqnorm`	QQ plot for normal distribution
`qqline`	plots diagonal line
`qqplot`	QQ plotting for other distributions

outlier labeling

Informal test to flag potential outliers for further investigation.

- standard deviation method; z-score
- MAD_e method
- modified z-score
- boxplot; adjusted boxplot

standard deviation (SD) method

standard deviation (SD) method

- Very simple.

standard deviation (SD) method

- Very simple.
- Assumed to follow an approximately normal distribution.

standard deviation (SD) method

- Very simple.
- Assumed to follow an approximately normal distribution.

  2 SD method: $\bar{x} \pm 2s$
  3 SD method: $\bar{x} \pm 3s$

z-scores

The z-score has an identical result as the SD method. The only difference is that the data set is transformed:

$$z_i = \frac{x_i - \bar{x}}{s}$$

The transformed data is given in units of how many standard deviations it is from the mean, e.g., if $|z_i| > 3 \rightarrow$ potential outlier.

z-score

Chebyshev's theorem

$$P\left[|X-\mu| \geq k\sigma\right] \leq \frac{1}{k^2}$$

Chebyshev's theorem applies regardless of the distribution of the data.

MAD_e method

- MAD_e method is a more robust version of the SD method
- based on median absolute deviation (MAD)

$$\text{MAD} = \text{median}(|x_i - \tilde{x}|)$$

($\tilde{x}$ denotes the median of X)

MAD_e method

- MAD_e method is a more robust version of the SD method
- based on median absolute deviation (MAD)

$$\text{MAD} = \text{median}(|x_i - \tilde{x}|)$$

($\tilde{x}$ denotes the median of X)

MAD_e method

- MAD_e method is a more robust version of the SD method
- based on median absolute deviation (MAD)

$$\text{MAD} = \text{median}(|x_i - \tilde{x}|)$$

($\tilde{x}$ denotes the median of X)

MAD_e method

- MAD_e method is a more robust version of the SD method
- based on median absolute deviation (MAD)

$$\text{MAD} = \text{median}(|x_i - \tilde{x}|)$$

($\tilde{x}$ denotes the median of X)

2 MAD_e method: median $\pm 2\ \text{MAD}_\text{e}$

3 MAD_e method: median $\pm 3\ \text{MAD}_\text{e}$

where $\text{MAD}_\text{e} = 1.483 \times \text{MAD}$

modified z-score

modified z-score

- The z-score method relies on sample mean and standard deviation, both sensitive to extreme values.

modified z-score

- The z-score method relies on sample mean and standard deviation, both sensitive to extreme values.
- The modified z-score uses the more robust median and MAD which as estimators:

$$m_i = \frac{0.6745\,(x_i - \tilde{x})}{MAD}$$

modified z-score

- The z-score method relies on sample mean and standard deviation, both sensitive to extreme values.
- The modified z-score uses the more robust median and MAD which as estimators:

$$m_i = \frac{0.6745\,(x_i - \tilde{x})}{MAD}$$

- Rule of thumb: observations are labeled outliers when $|m_i| > 3.5$

z-score and modified z-score

z-score with delimiters at $|z| \geq 3$

modified z-score with delimiters at $|m| \geq 3.5$

boxplot and adjusted boxplot

The boxplot method makes no distributional assumptions.

Potential outliers are outside the “inner fences”

- Lower inner fence: Q1 - 1.5 $\times$ IQR
- Upper inner fence: Q3 + 1.5 $\times$ IQR

With highly skewed data it may identify (too) many outliers.

boxplot and adjusted boxplot

The adjusted boxplot allows for more skewed distributions by incorporating a measure of skewness in the computation.

Potential outliers are outside the “fences”

Adjusted boxplots are available in the `robustbase` package in R through the `adjbox` function.

E. Vanderviere and M. Huber (2008), An adjusted boxplot for skewed distributions. Computational Statistics and Data Analysis **52**, 5186-5201

Exponential distribution

Lognormal distribution

outlier identification

Outlier identification is comprised of formal tests for detecting outliers.

- normality is assumed in most tests
- single outlier vs multiple outlier tests
- issues of *masking* and *swamping*
- Grubb's test;
  generalized extreme studentized deviates (ESD)

outlier identification

Outlier identification is comprised of formal tests for detecting outliers.

- normality is assumed in most tests
- single outlier vs multiple outlier tests
- issues of *masking* and *swamping*
- Grubb's test;
  generalized extreme studentized deviates (ESD)

outlier identification

Outlier identification is comprised of formal tests for detecting outliers.

- normality is assumed in most tests
- single outlier vs multiple outlier tests
- issues of *masking* and *swamping*
- Grubb's test;
  generalized extreme studentized deviates (ESD)

outlier identification

Outlier identification is comprised of formal tests for detecting outliers.

- normality is assumed in most tests
- single outlier vs multiple outlier tests
- issues of *masking* and *swamping*
- Grubb's test;
  generalized extreme studentized deviates (ESD)

masking and swamping

Masking

One outlier masks another if the second outlier can be considered an outlier only by itself, but not in the presence of the first.

masking and swamping

Masking

One outlier masks another if the second outlier can be considered an outlier only by itself, but not in the presence of the first.

Swamping

One outlier swamps a second observation if the latter can be considered as an outlier only under the presence of the first one.

masking and swamping

Hawkins, Bradu and Kass, 1984

- 75 obs, 3 predictors
- 10 outliers (cases 1 - 10)
- cases 11,12,13,14 possibly swamped

masking and swamping

Hawkins, Bradu and Kass, 1984

- 75 obs, 3 predictors
- 10 outliers (cases 1 - 10)
- cases 11,12,13,14 possibly swamped

masking and swamping

Hadi and Simonoff, 1993

- 25 obs, 2 predictors
- 3 significant outliers: cases 1-3
- cases 6,11,13,17,24 possibly swamped

masking and swamping

Hadi and Simonoff, 1993

- 25 obs, 2 predictors
- 3 significant outliers: cases 1-3
- cases 6,11,13,17,24 possibly swamped

Grubb's test

H_0: no outliers in the data set
H_a: *exactly one* outlier in the data set

test statistic:

$$G = \frac{\max\{|x_i - \bar{x}|\}}{s}$$

For a given α, the null hypothesis is rejected if

$$G > \frac{n-1}{\sqrt{n}} \sqrt{\frac{t^2_{\alpha/(2n),n-2}}{n-2+t^2_{\alpha/(2n),n-2}}}$$

where $t_{\alpha/(2n),n-2}$ denotes the critical value of the t-distribution with $(n-2)$ degrees of freedom and a significance level of $\alpha/2n$.

generalized extreme studentized deviates

- used to detect *one or more* outliers
- requires an upper bound on outliers to be specified
- given an upper bound, r, generalized ESD test essentially performs r separate tests: a test for one outlier, a test for two outliers, and so on up to r outliers.

generalized extreme studentized deviates

H_0: no outliers in the data set
H_a: up to r outliers in the data set

test statistic:

$$R_j = \frac{\max\{|x_i - \bar{x}|\}}{s}$$

generalized extreme studentized deviates

H_0: no outliers in the data set
H_a: up to r outliers in the data set

test statistic:

$$R_j = \frac{\max\{|x_i - \bar{x}|\}}{s}$$

Remove the observation that maximizes $|x_i - \bar{x}|$ and recompute R with $n-1$ observations. Repeat until r observations have been removed. Produces r test statistics: $R_1, R_2, \ldots, R_r$.

generalized extreme studentized deviates

H_0: no outliers in the data set
H_a: up to r outliers in the data set

test statistic:

$$R_j = \frac{\max\{|x_i - \bar{x}|\}}{s}$$

Remove the observation that maximizes $|x_i - \bar{x}|$ and recompute R with $n-1$ observations. Repeat until r observations have been removed. Produces r test statistics: $R_1, R_2, \ldots, R_r$.

The r critical values:

$$\lambda_j = \frac{(n-j)\, t_{1-\alpha/(2(n-j+1)),n-j+1}}{\sqrt{\left(n-j-1+t^2_{1-\alpha/(2(n-j+1)),n-j+1}\right)(n-j+1)}} \quad j = 1, \ldots, r$$

example: Grubbs' vs. generalized ESD

Consider this data from Rosner (1983):

-0.25	0.68	0.94	1.15	1.20	1.26	1.26	1.34	1.38
1.43	1.49	1.49	1.55	1.56	1.58	1.65	1.69	1.70
1.76	1.77	1.81	1.91	1.94	1.96	1.99	2.06	2.09
2.10	2.14	2.15	2.23	2.24	2.26	2.35	2.37	2.40
2.47	2.54	2.62	2.64	2.90	2.92	2.92	2.93	3.21
3.26	3.30	3.59	3.68	4.30	4.64	5.34	5.42	6.01

B. Rosner (1983). Percentage Points for a Generalized ESD Many-Outlier Procedure,Technometrics, **25**, 165-172.

example: Grubbs' vs. generalized ESD

Test for a single outlier with $\alpha = 0.05$.

```
> grubbs.test(rosner)

Grubbs test for one outlier

data:  rosner
G = 3.1189, U = 0.8130, p-value = 0.02949
alternative hypothesis: highest value 6.01 is an outlier
```

example: Grubbs' vs. generalized ESD

example: Grubbs' vs. generalized ESD

Inappropriate sequential application of a single outlier test.

```
> grubbs.test(rosner[rosner<6.01])

Grubbs test for one outlier

data:  rosner[rosner < 6.01]
G = 2.9430, U = 0.8302, p-value = 0.05759
alternative hypothesis: highest value 5.42 is an outlier
```

example: Grubbs' vs. generalized ESD

example: Grubbs' vs. generalized ESD

generalized ESD test with $\alpha = 0.05$ and $r = 10$

```
> removeoutliers(rosner,10,0.05)

$numOutliers
[1] 3

$outliers
[1] 5.34 5.42 6.01
```

example: Grubbs' vs. generalized ESD

Number of outliers, j	Test statistic, R_j	Critical value, λ_j
1	3.118	3.158
2	2.942	3.151
3	3.179	3.143
4	2.810	3.136
5	2.815	3.128
6	2.848	3.120
7	2.279	3.111
8	2.310	3.103
9	2.101	3.094
10	2.067	3.085

example: Grubbs' vs. generalized ESD

detecting outliers: R

`outliers` package in R includes:

- `cochran.test` useful to check if largest variance in several groups of data is “outlying”. Alternatively, if one group has very small variance, can test for "inlying" variance.
- `dixon.test` performs several variants of Dixon test (Dixon 1950, 1950) for detecting outlier in data sample.
- `grubbs.test` performs three versions of Grubb’s test

User defined code for computing Generalized ESD is available in course website: `generalizedESD.R`

outlier detection for multidimensional data

- Scatter plots for (visually detecting) outliers w.r.t. two attributes.
- PCA plots for (visually detecting) outliers.
- Cluster analysis techniques: Outliers are those points which cannot be assigned to any cluster.

Outline

1. Data Understanding
2. Data Quality
   - Accuracy
   - Completeness
   - Timeliness
3. Visualizations and Data Exploration
4. ggplot2
5. Correlation Analysis
6. Outliers
7. **Missing Values**

missing values

Missing data are common.

- e.g. Wood et al. 2004 reviewed 71 published articles in top tier medical journals.

missing values

Missing data are common.

- e.g. Wood et al. 2004 reviewed 71 published articles in top tier medical journals.
- 89% had partly missing outcome data.

missing values

Missing data are common.

- e.g. Wood et al. 2004 reviewed 71 published articles in top tier medical journals.
- 89% had partly missing outcome data.

missing values

Missing data are common.

- e.g. Wood et al. 2004 reviewed 71 published articles in top tier medical journals.
- 89% had partly missing outcome data.

> *...missing outcome data are a **common problem** in randomized controlled trials, and are **often inadequately handled** in the statistical analysis...*

A. Wood, I. White, and S. Thompson. (2004) Are missing outcome data adequately handled? A review of published randomized controlled trials in major medical journals. Clinical Trials.

missing values

Example causes for missing values:

- broken sensors
- refusal to answer a question
- irrelevant attribute for the corresponding object (pregnant (yes/no) for men)

missing values

Example causes for missing values:

- broken sensors
- refusal to answer a question
- irrelevant attribute for the corresponding object (pregnant (yes/no) for men)

Note: Missing value might not necessarily be indicated as missing $\longrightarrow$ this can be problematic.

digression: missing values in R

- *missing values* are represented by **NA** (not available)
- *impossible values* (e.g., dividing by zero) are represented by **NaN** (not a number).

digression: missing values in R

- *missing values* are represented by **NA** (not available)
- *impossible values* (e.g., dividing by zero) are represented by **NaN** (not a number).

digression: missing values in R

Checking for Missing Values in R

```
is.na(x) # returns TRUE if x is missing
y <- c(1,2,3,NA)
is.na(y) # returns a vector (F F F T)
```

digression: missing values in R

Checking for Missing Values in R

```
is.na(x) # returns TRUE if x is missing
y <- c(1,2,3,NA)
is.na(y) # returns a vector (F F F T)
```

Recoding Values to Missing in R

```
# recode 99 to missing for variable v1
mydata$v1[mydata$v1==99] <- NA
```

some notation

Let $\mathbf{X}$ denote the data we have, partitioned as follows:

$$\mathbf{X} = \{\mathbf{X}_o, \mathbf{X}_m\}$$

where $\mathbf{X}_o$ is observed and $\mathbf{X}_m$ is missing.

Corresponding to every observation of $\mathbf{X}$, there is a missing value indicator $\mathbf{R}$ defined as:

$$\mathbf{R} = \begin{cases} 1 \text{ if } \mathbf{X} \text{ observed} \\ 0 \text{ if } \mathbf{X} \text{ missing} \end{cases}$$

missing value mechanism

The missing value mechanism is the probability that a set of values are missing given the values of the observed and missing observations:

$$P(\mathbf{R} = r | \mathbf{x}_o, \mathbf{x}_m)$$

missing value mechanism: examples

- Probability of nonresponse to questions about income usually depend on the person's income.
- Someone may not be at home for an interview because they are at work.
- The chance of a subject leaving a clinical trial may depend on their response to treatment.

missing value mechanism: examples

- Probability of nonresponse to questions about income usually depend on the person's income.
- Someone may not be at home for an interview because they are at work.
- The chance of a subject leaving a clinical trial may depend on their response to treatment.

missing value mechanism: examples

- Probability of nonresponse to questions about income usually depend on the person's income.
- Someone may not be at home for an interview because they are at work.
- The chance of a subject leaving a clinical trial may depend on their response to treatment.

missing value mechanism: MCAR

Suppose the probability of an observation being missing does not depend on observed or unobserved measurements.

That is, if

$$P(\mathbf{r}|\mathbf{x}_o, \mathbf{x}_m) = P(\mathbf{r})$$

then the observation is Missing Completely At Random (MCAR)

missing value mechanism: MCAR

Suppose the probability of an observation being missing does not depend on observed or unobserved measurements.

That is, if

$$P(\mathbf{r}|\mathbf{x}_o, \mathbf{x}_m) = P(\mathbf{r})$$

then the observation is Missing Completely At Random (MCAR)

Example: a laboratory sample is dropped, so the resulting observation is missing.

missing value mechanism: MCAR

Suppose the probability of an observation being missing does not depend on observed or unobserved measurements.

That is, if

$$P(\mathbf{r}|\mathbf{x}_o, \mathbf{x}_m) = P(\mathbf{r})$$

then the observation is Missing Completely At Random (MCAR)

Example: a laboratory sample is dropped, so the resulting observation is missing.

MCAR is also called Observed At Random (OAR).

missing value mechanism: MAR

Missing At Random (MAR) is when the missingness mechanism does not depend on the unobserved data.

That is, if

$$P(\mathbf{r}|\mathbf{x}_o, \mathbf{x}_m) = P(\mathbf{r}|\mathbf{x}_o)$$

missing value mechanism: MAR

Missing At Random (MAR) is when the missingness mechanism does not depend on the unobserved data.

That is, if

$$P(\mathbf{r}|\mathbf{x}_o, \mathbf{x}_m) = P(\mathbf{r}|\mathbf{x}_o)$$

Example: Income is MAR if the probability of missing data on income depends on an observed variable like marital status (assuming marital status is observed.) Within each category of marital status (single, married, divorced, etc.), the probability of missing income is unrelated to the value of income.

missing value mechanism: MNAR

Missing Not At Random (MNAR): The probability that a value is missing depends on the true value of the missing variable.

Also called non-ignorable.

Example: Income is MNAR if the probability of missing data on income depends on the actual income.

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 (impossible?) 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 (impossible?) 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 (impossible?) 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.