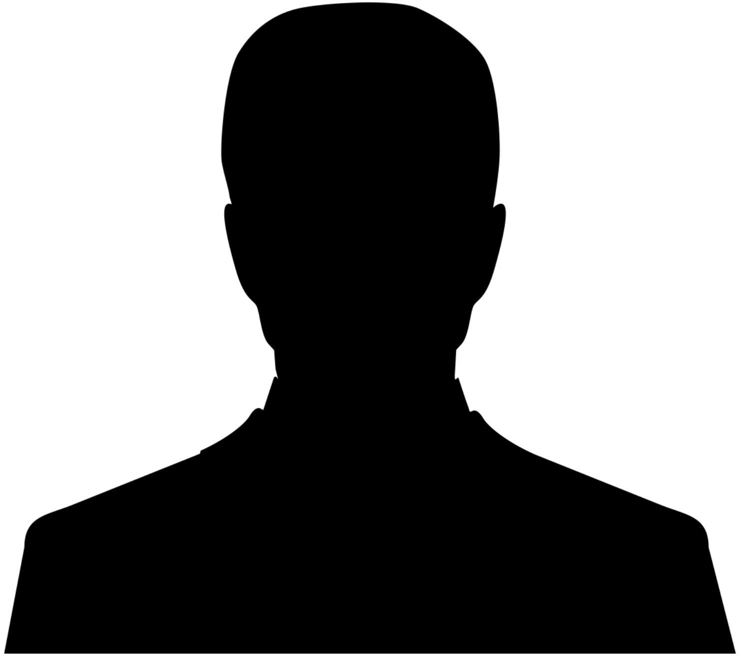
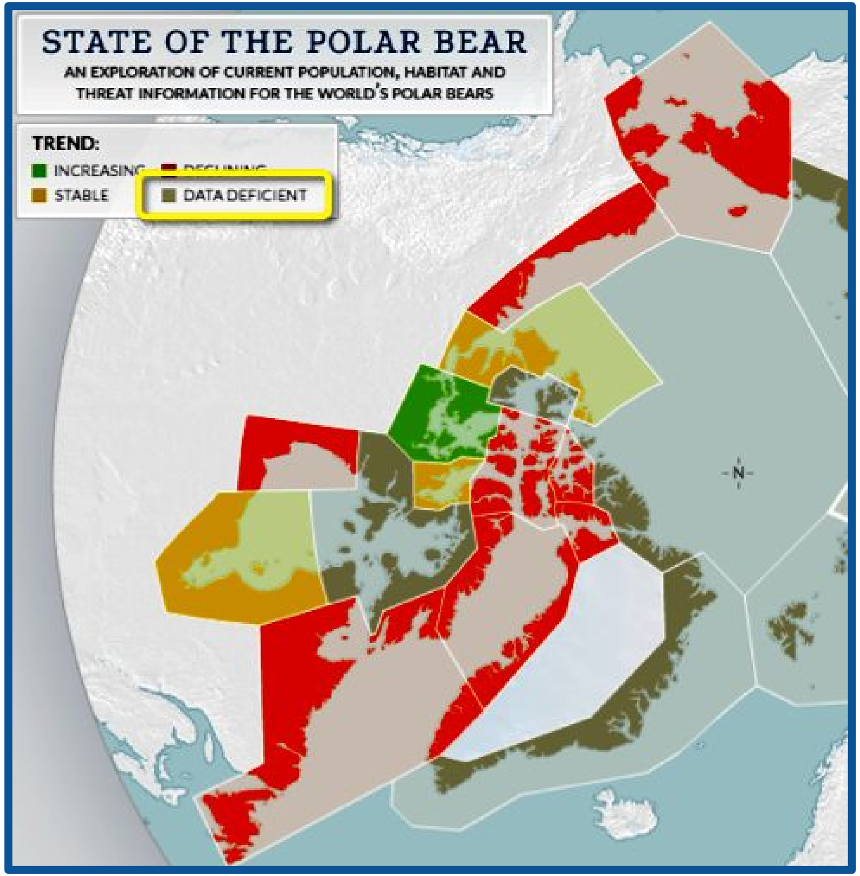
Preprocessing Data
Lesson Preview

- Learn how to handle **missing data**
- Learn how to handle **outliers**
- Learn when and how to **transform data**
- Learn **standard data manipulations techniques**





Polar Bear Quiz

Select the reason **why there are areas with no information** on the polar bear population.

- [ ] There are no polar bears in these areas
- [ ] There are too few polar bears to achieve a reliable count
- [x] These areas are controlled by a country that did not allow the collection of data

Missing Data

Data may be missing for a **variety of reasons:**

corrupted during its transfer or storage

some instances in the data collection process were skipped due to difficulty or price associated with obtaining the data

Missing Data

Sample #	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
1	3.5	1.2		2.2	
2	3.4	2.1	3.2	2.3	
3	3.45	2.2	3.25	2.4	

Missing Data Examples

Recommendation systems: users don't rate every item

Longitudinal studies: subjects may drop out of the study

Sensor Data: sensor failure

User Surveys: users may have privacy concerns

Missing Completely at Random (MCAR)

MCAR = probability of an observation being missing does not depend on observed or unobserved measurements.

Missing Completely at Random (MCAR)

User	Casablanca	The Godfather	The Wizard of Oz	Throne of Blood	Spies
1	5 stars	3 stars	5 stars	2.2	
2	3 stars		5 stars	2.3	
3	4 stars	4.5 stars		4 stars	1 star

Missing at Random (MCAR)

MAR = given the observed data, the probability that data is missing does not depend on the unobserved data.

Missing at Random (MAR)

Response	Gender	Race	Income
1	M	Asian	$$$$$
2	M	Pacific Islander	**MAR**
3	F	Asian	

Missing at Random (MAR)

Response	Gender	Race	Income
1	M	Asian	$$$$$
2	M	Pacific Islander	**MAR**
3	F	Asian	**NOT MAR**

Profession

Age

MAR Quiz

Select the data that is MAR but not MCAR:

- [x] In the study of quality of life the psychologist finds that elderly patients and patients with less education have a higher probability to refuse the QL questionnaire.
- [x] Missing blood pressure measurement may be lower than measured blood pressure because younger people may be more likely to have missing blood pressure measurements.
- [ ] Blood pressure measurement is missing because of a breakdown of an automatic sphygmomanometer.
- [ ] The study is not effective for reducing the blood pressure, and there may be a chance subjects will drop out of the study.

Handling Missing Data

Response	Gender	Race	Income
1	M	Asian	$$$$$
2	M	Pacific Islander	
3	F	Asian	

Handling Missing Data

Remove all data instances (for example dataframe rows) containing missing values.

Response	Gender	Race	Income
1	M	Asian	$$$$$

Handling Missing Data

Replace all missing entries with a substitute value, for example the mean of the observed instances of the missing variable.

Response	Gender	Race	Income
1	M	Asian	$$$$$
2	M	Pacific Islander	**mean of income**
3	F	Asian	**mean of income**

Handling Missing Data

Estimate a probability model for the missing variable and **replace the missing value with one or more samples from that probability model.**

Response	Gender	Race	Income
1	M	Asian	$$$$$
2	M	Pacific Islander	**Probability Model Sample 1**
3	F	Asian	**Probability Model Sample 2**

Handling Missing Data

MCAR:
the three techniques are reasonable, though some are better

MAR or Non-MAR:
the techniques may introduce systematic bias into the data analysis process.

Missing Data and R

Response	Gender	Race	Income
1	M	Asian	$$$$$
2	M	Pacific Islander	**NA**
3	F	Asian	**NA**

R represents missing data using the NA keyword.

Missing Data and R

is.na	• Returns TRUE for missing data • Returns FALSE otherwise
complete.cases()	• Returns a vector whose components are FALSE for all samples • Returns TRUE otherwise
na.omit()	• Returns a new dataframe omitting all samples containing missing values
na.rm	• If set TRUE changes the function behavior so that it proceeds to operate on the supplied data after removing all dataframe rows with missing values

NA Quiz

Fill in the blanks with the purpose of the command:

mean(movies$length) | # average length |

mean(movies$budget) | # average budget |

mean(movies$budget, na.rm = | true |) mean avg budget, remove missing values

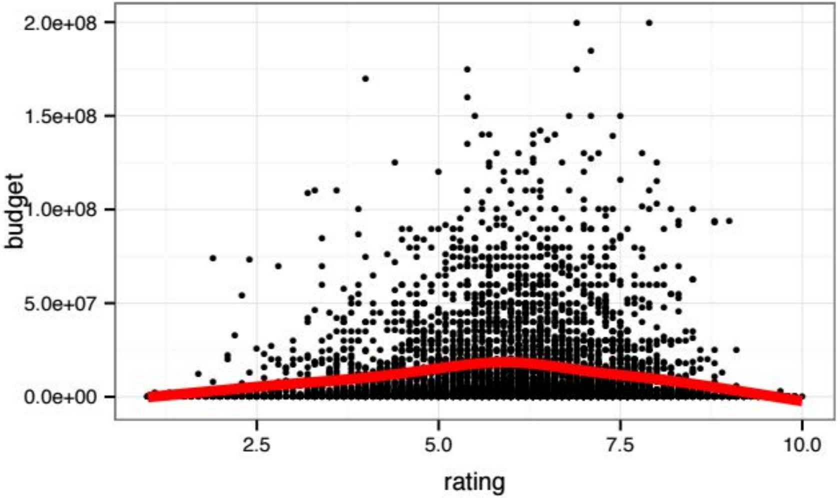
mean(is.na(movies$budget)) | # frequency of missing budget |

moviesNoNA = na.omit(| movies |) # returns a dataset with all missing data removed.

R Plot Quiz

Fill in the blanks to create the given plot:

```
moviesNoNA = na.omit(movies)
qplot(rating, budget, data = moviesNoNA , size = I(1.2 )) +
    stat_smooth(color = "red", size = I(2 ), se = F )
```

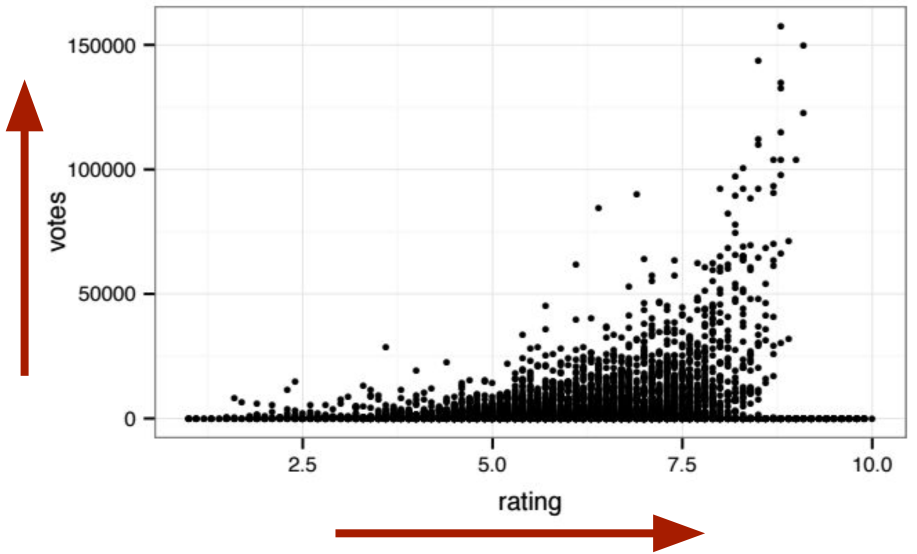


Movie Data Analysis Quiz

Select **which of the following statements can be derived** from the plot:

- [x] Number of votes tend to increase as the average ratings increase.
- [x] Spread in the number of votes increases with the average rating.
- [x] Movies featuring the highest average ratings have a very small number of votes.
- [x] Observed ratings will tend to be higher than ratings gathered after showing users random movies

```
moviesNoNA = na.omit(movies)
qplot(rating, votes, data = moviesNoNA, size = I(1.2))
```

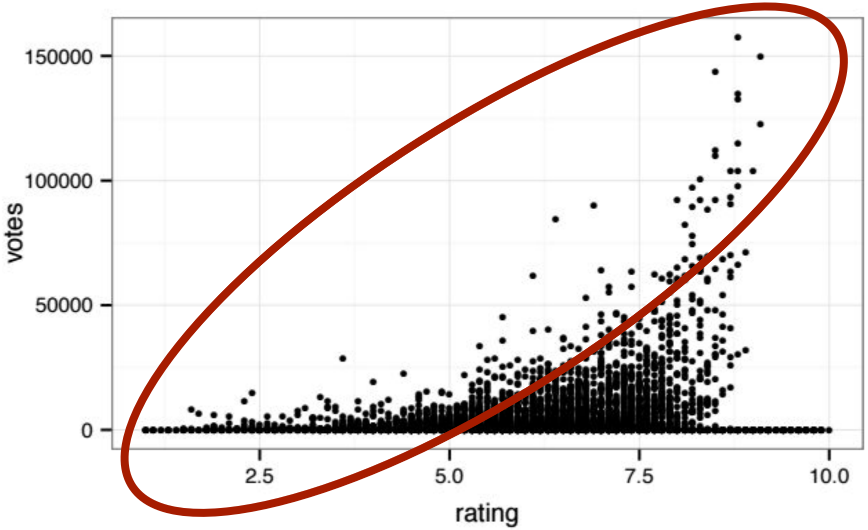


Movie Data Analysis Quiz

Select **which of the following statements can be derived** from the plot:

- [x] Number of votes tend to increase as the average ratings increase.
- [x] Spread in the number of votes increases with the average rating.
- [x] Movies featuring the highest average ratings have a very small number of votes.
- [x] Observed ratings will tend to be higher than ratings gathered after showing users random movies

```
moviesNoNA = na.omit(movies)
qplot(rating, votes, data = moviesNoNA, size = I(1.2))
```

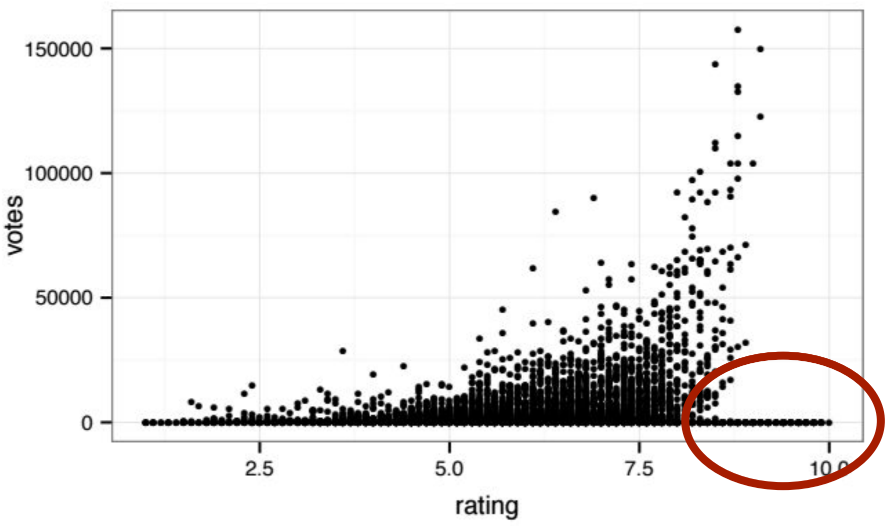


Movie Data Analysis Quiz

Select **which of the following statements can be derived** from the plot:

- [x] Number of votes tend to increase as the average ratings increase.
- [x] Spread in the number of votes increases with the average rating.
- [x] Movies featuring the highest average ratings have a very small number of votes.
- [x] Observed ratings will tend to be higher than ratings gathered after showing users random movies

```
moviesNoNA = na.omit(movies)
qplot(rating, votes, data = moviesNoNA, size = I(1.2))
```

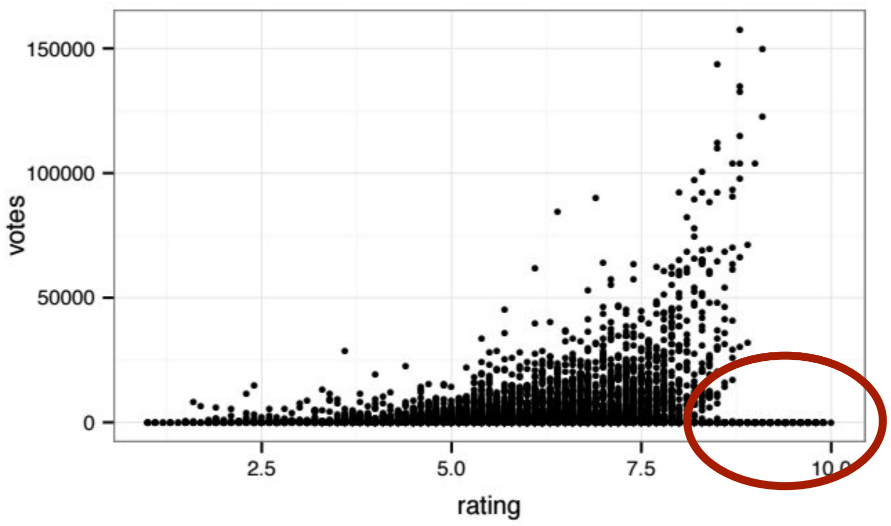


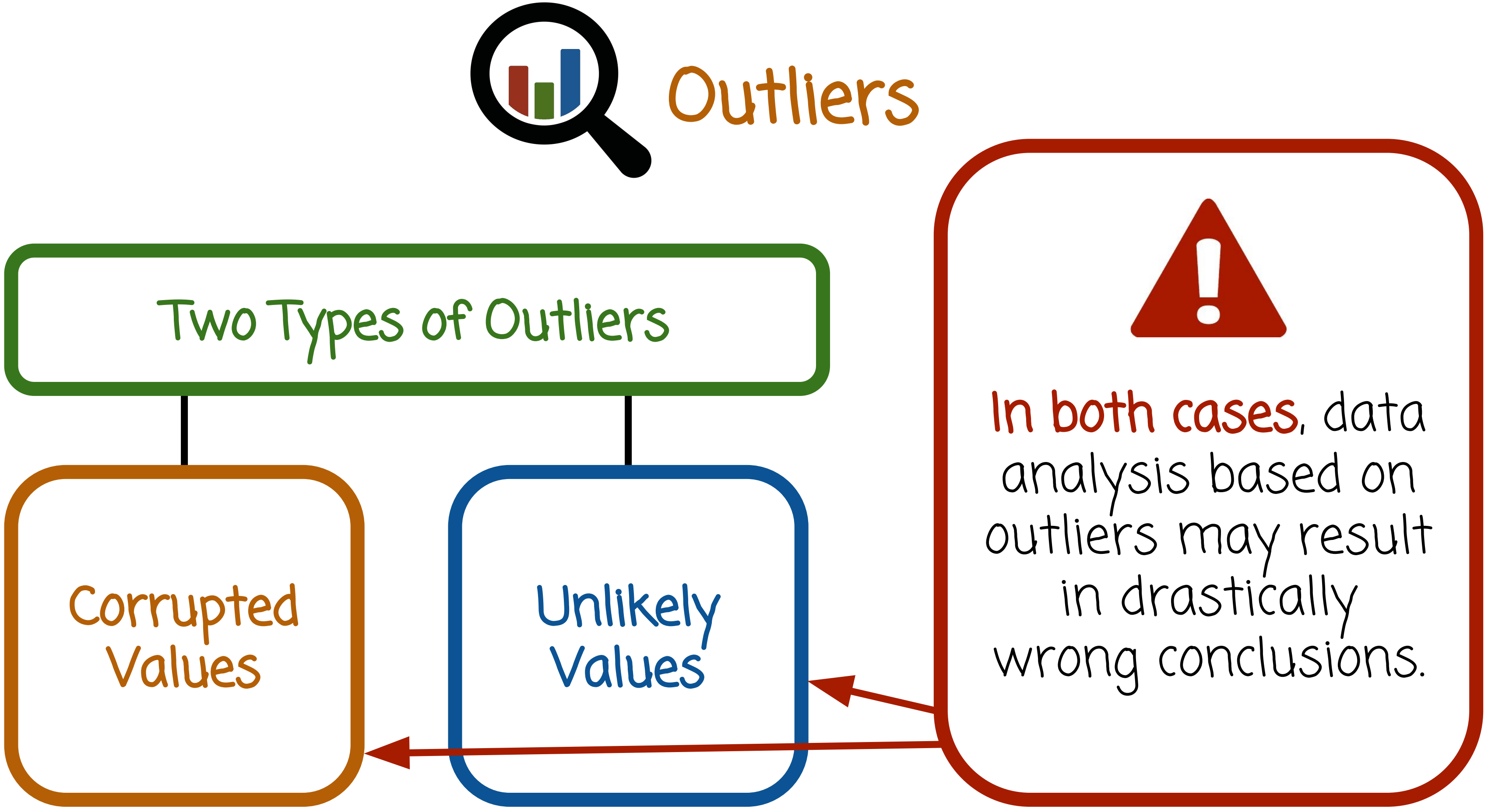
Movie Data Analysis Quiz

Select **which of the following statements can be derived** from the plot:

- [x] Number of votes tend to increase as the average ratings increase.
- [x] Spread in the number of votes increases with the average rating.
- [x] Movies featuring the highest average ratings have a very small number of votes.
- [x] Observed ratings will tend to be higher than ratings gathered after showing users random movies

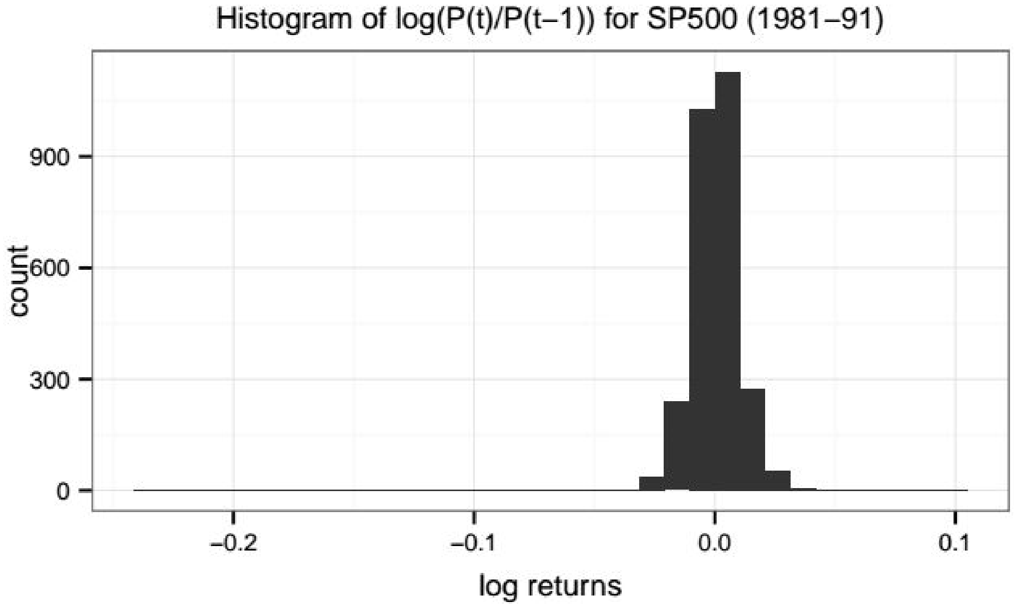
```
moviesNoNA = na.omit(movies)
qplot(rating, votes, data = moviesNoNA, size = I(1.2))
```




Outliers
Two Types of Outliers
Corrupted Values
Unlikely Values
In both cases, data analysis based on outliers may result in drastically wrong conclusions.

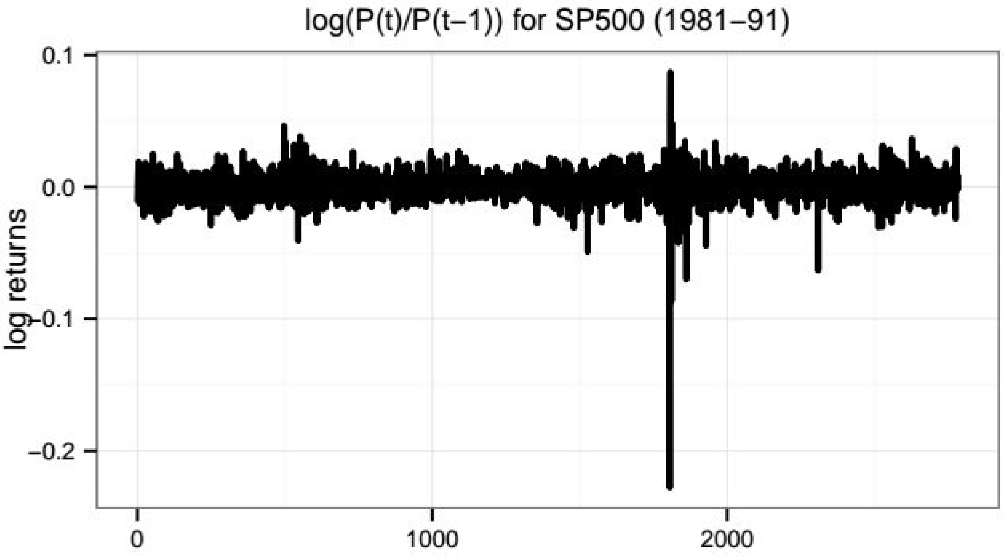
Outliers

```
library(Ecdat)
data(SP500, package = 'Ecdat')
qplot(r500,
  main = "Histogram of log(P(t)/P(t-1)) for SP500 (1981-91)",
  xlab = "log returns",
  data = SP500)
```



Outliers

```
qplot(seq(along = r500),
  r500,
  data = SP500,
  geom = "line",
  xlab = "trading days since January 1981",
  ylab = "log returns",
  main = "log(P(t)/P(t-1)) for SP500 (1981-91)")
```



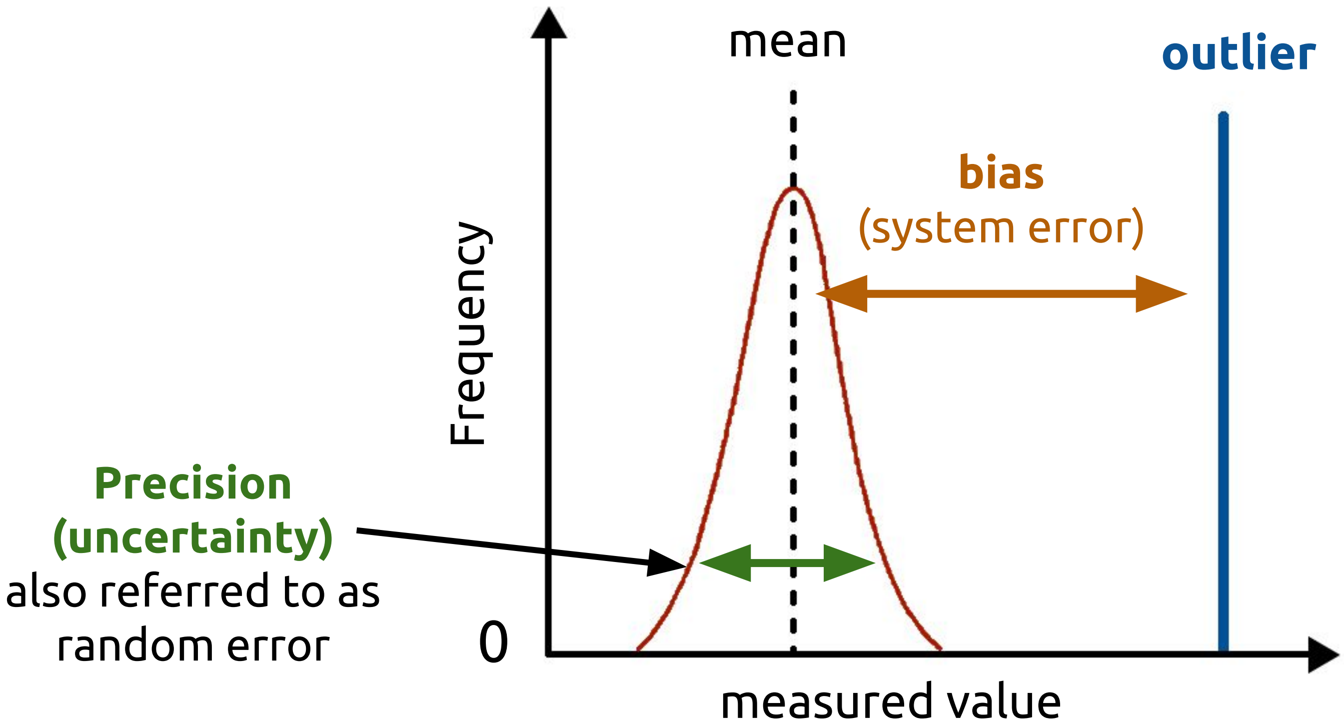
Robustness Quiz

Robustness describes a lack of sensitivity of data analysis procedures to outliers.

Assuming a symmetric distribution of samples around 0, **which data analysis procedure is more robust**?

- [ ] Mean
- [x] Median

Robustness Quiz



Dealing with Outliers

Truncating: Remove all values deemed as outliers.

Winsorization: Shrink outliers to border of main part of data

Robustness: Analyze the data using a robust procedure

Detecting Outliers

To **remove outliers** we need to first **detect them**:

values below the alpha percentile or above the 100-alpha percentile

values more than c times standard deviation away from the mean

Detecting Outliers

Compute the mean and standard deviation after removing the most extreme values

Percentiles (that are more robust) can be used.

Programming Quiz

Write code using 'R' to first create samples from a normal distribution and an outlier, print it, and then winsorize it

```
library(robustHD)
originalData = c(1000, rnorm(10))
print(originalData[1:5])

## [1] 1000.0000   -0.6265    0.1836   -0.8356    1.5953

print(winsorize(originalData[1:5]))

## [1]   3.2060 -0.6265  0.1836 -0.8356  1.5953
```

Programming Quiz

Write code using 'R' to remove data that is 5 std less than the mean and 5 std greater than the mean, where the std and mean are computed without extreme measurements

```
original_data = rnorm(20)
original_data[1] = 1000
sorted_data = sort(original_data)
filtered_data = original_data[3:18]
lower_limit = mean(filtered_data) - 5 * sd(filtered_data)
upper_limit = mean(filtered_data) + 5 * sd(filtered_data)
not_outlier_ind = (lower_limit < original_data) &
  (original_data < upper_limit)
print(not_outlier_ind)
data_w_no_outliers = original_data[not_outlier_ind]
```

Data Transformations: Skewness and Power Transformations

Data is drawn from a highly-skewed distribution

A simple transformation may map the data to a form that is well described by common distributions

A suitable model can then be fitted to the transformed data

Data Transformations: Skewness and Power Transformations

Power Transformation Family: replace non-negative data *x* by...

$$f_\lambda(x) = \begin{cases} (x^\lambda - 1)/\lambda & \lambda > 0 \\ \log x & \lambda = 0 \\ -(x^\lambda - 1)/\lambda & \lambda < 0 \end{cases} \qquad x > 0, \quad \lambda \in \mathbb{R}.$$

Data Transformations: Skewness and Power Transformations

- The power transform maps x to x^λ up to multiplication by a constant and addition of a constant.
- Subtracting 1 and dividing by λ makes $f_\lambda(x)$ continuous in λ as well as in x

$\lambda > 1$	$\lambda < 1$
mapping is convex	mapping is concave
removes left skewness	removes right skewness

Data Transformations: Skewness and Power Transformations

To select λ :

Try different values, graph the histograms, and select one of them.

Use a method based on maximum likelihood.

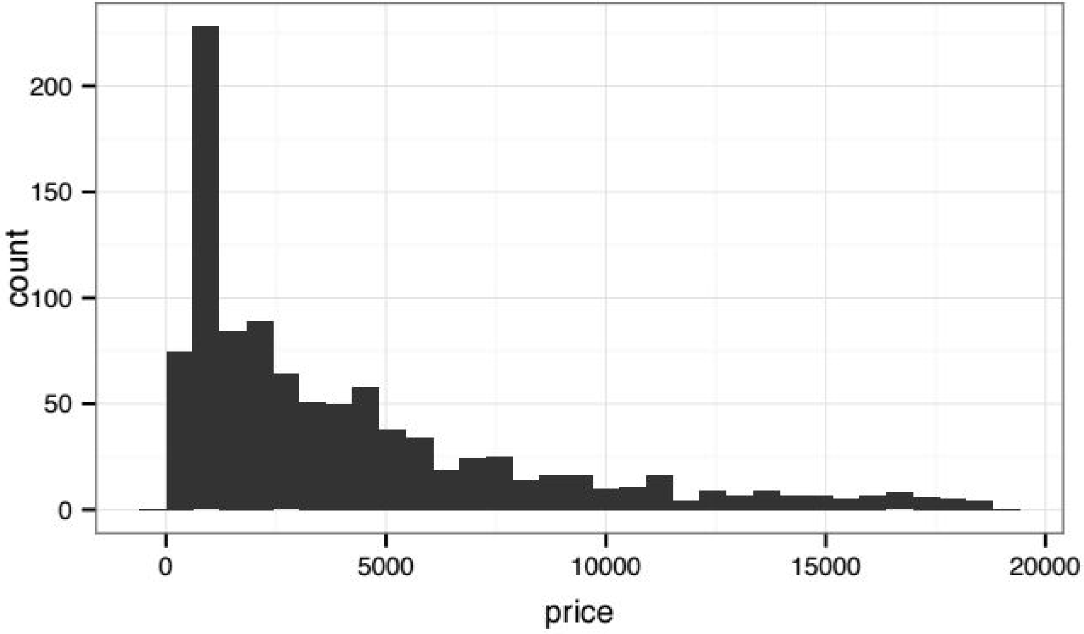
Diamonds Example

```
print(diamonds[1:10,1:8])
```

```
##     carat       cut color clarity depth table price    x
## 1    0.23     Ideal     E     SI2  61.5    55   326 3.95
## 2    0.21   Premium     E     SI1  59.8    61   326 3.89
## 3    0.23      Good     E     VS1  56.9    65   327 4.05
## 4    0.29   Premium     I     VS2  62.4    58   334 4.20
## 5    0.31      Good     J     SI2  63.3    58   335 4.34
## 6    0.24 Very Good     J    VVS2  62.8    57   336 3.94
## 7    0.24 Very Good     I    VVS1  62.3    57   336 3.95
## 8    0.26 Very Good     H     SI1  61.9    55   337 4.07
## 9    0.22      Fair     E     VS2  65.1    61   337 3.87
## 10   0.23 Very Good     H     VS1  59.4    61   338 4.00
```

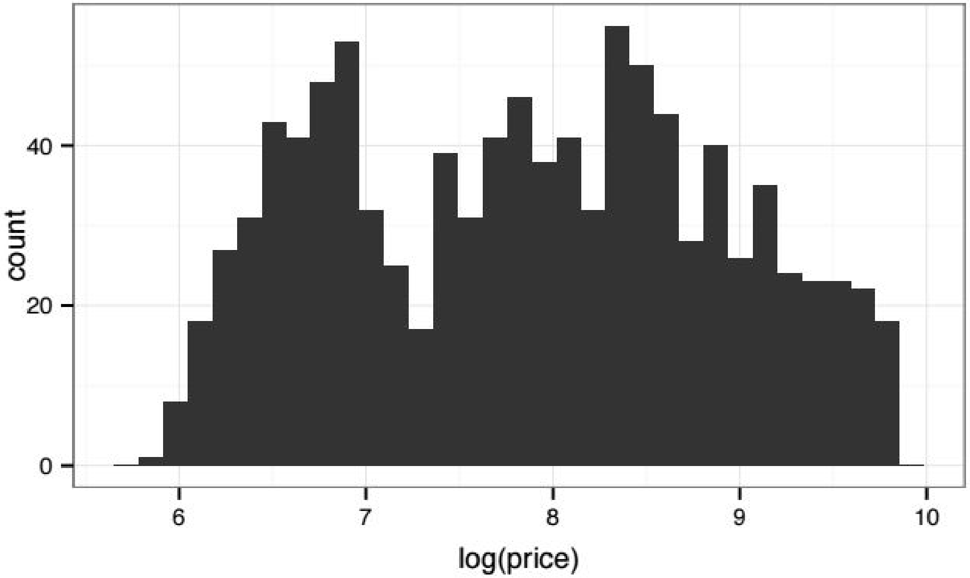
Diamonds Example

```
diamondsSubset = diamonds[sample(dim(diamonds)[1], 1000),]
qplot(price, data = diamondsSubset)
```



Diamond Quiz

```
qplot(log(price), size = I(1), data = diamondsSubset)
```



Based on the given graph, what conclusions can we draw about the count-price relationship?

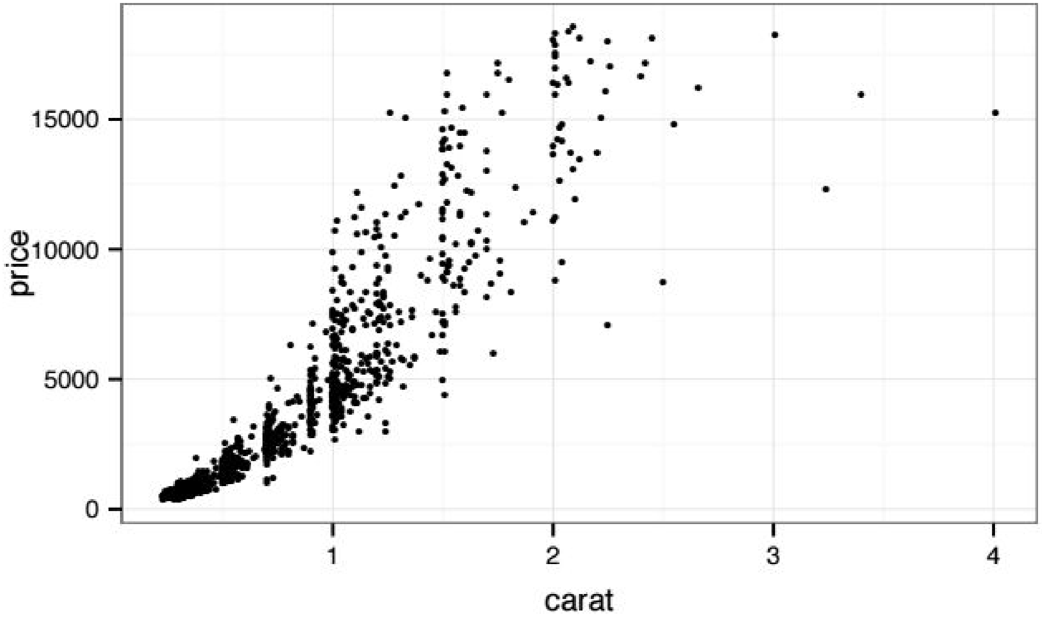
we see a bi-modal relationship here that was not visible on the original scale

Power Transformation

Power Transformations can be used to examine the relationship between two or more data types.

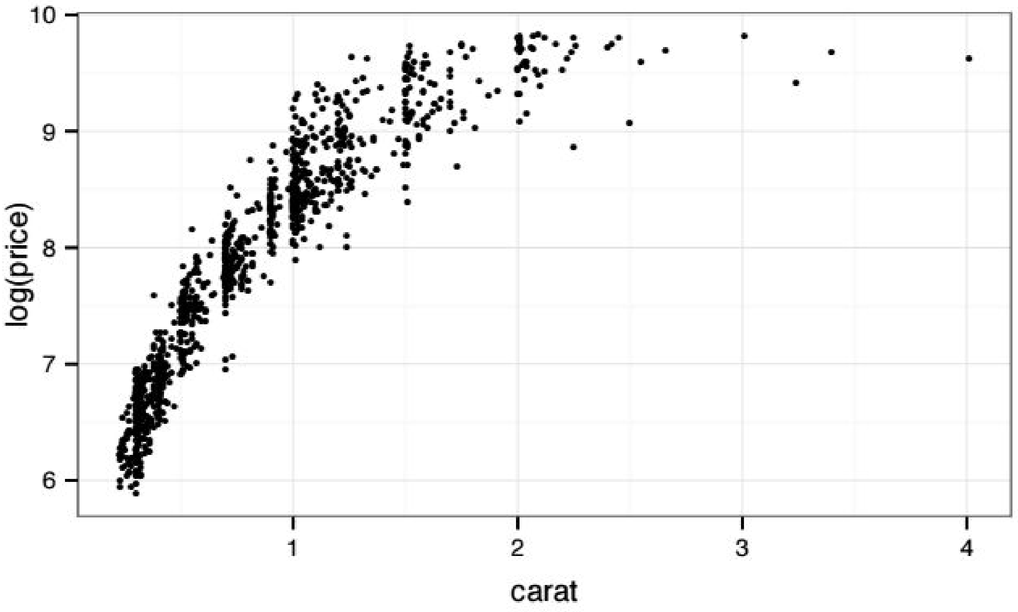
Power Transformation

```
qplot(carat,
      price,
      size = I(1),
      data = diamondsSubset)
```



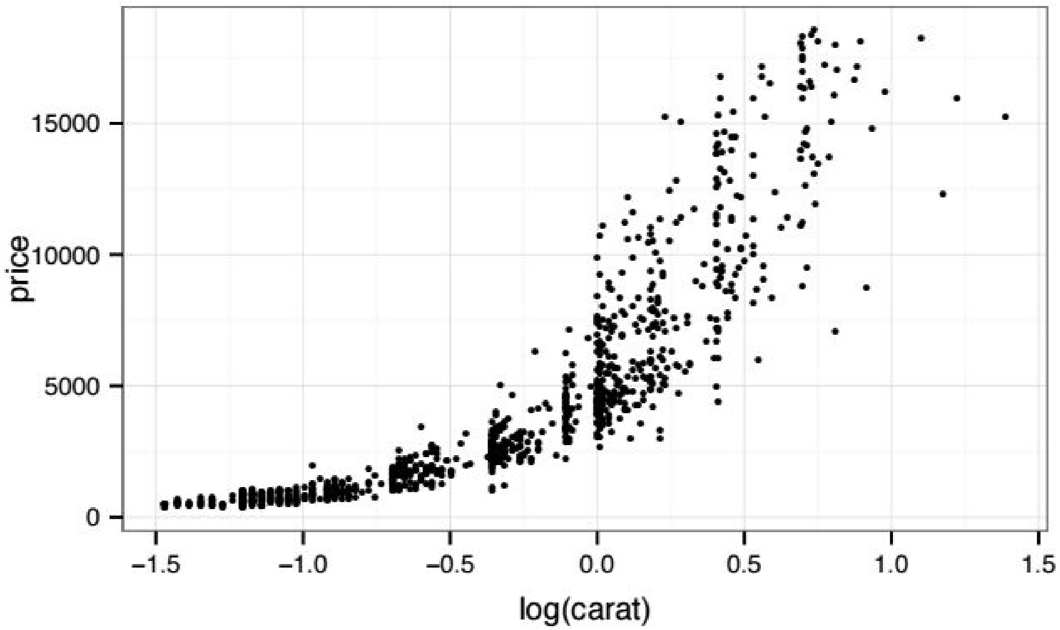
Power Transformation

```
qplot(carat,
      log(price),
      size = I(1),
      data = diamondsSubset)
```



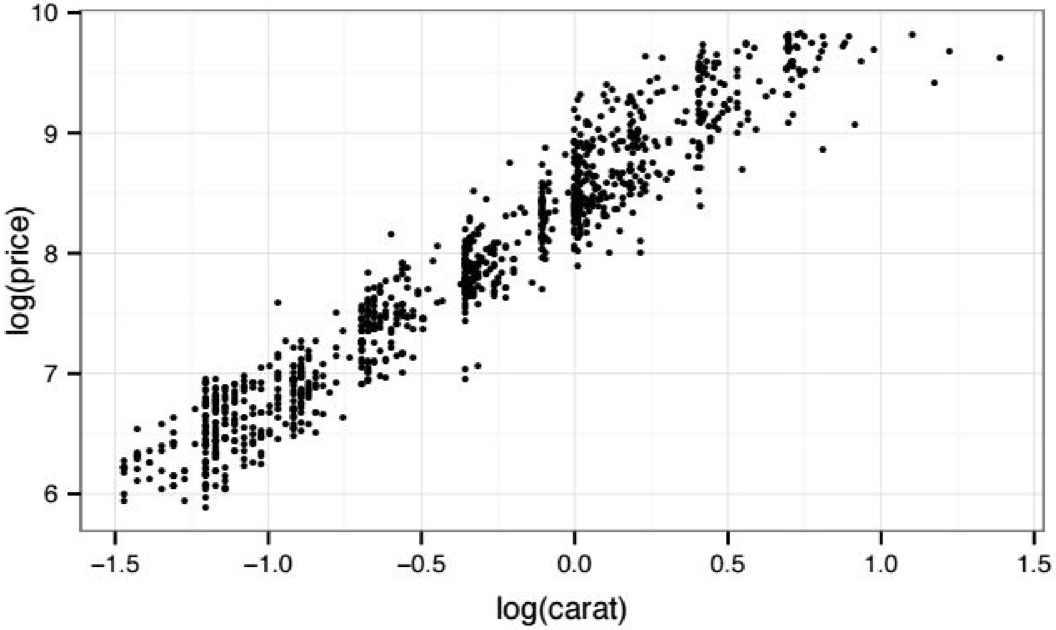
Power Transformation

```
qplot(log(carat),
      price,
      size = I(1),
      data = diamondsSubset)
```



Power Transformation

```
qplot(log(carat),
      log(price),
      size = I(1),
      data = diamondsSubset)
```



Power Quiz

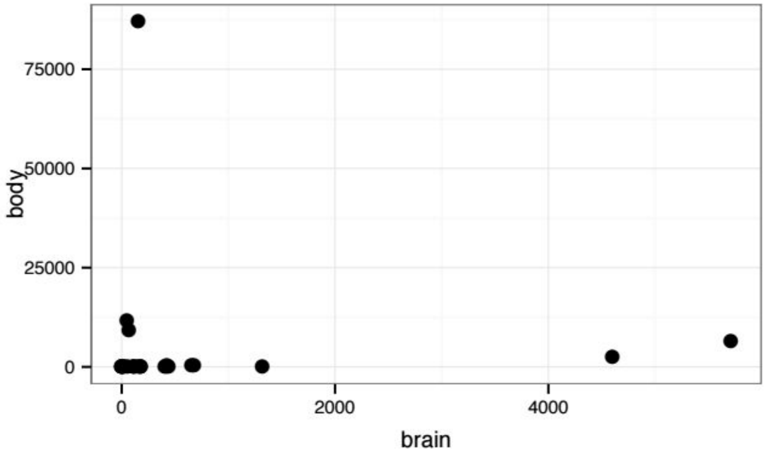
Given the following data and its plot, **change the qplot command** to show the relationship between the body and brain mass on a log log scale

```
qplot(brain, body, log = "xy", data = Animals)
```

```r
library(MASS)
print(Animals[1:12,])
```

```
##                     body  brain
## Mountain beaver     1.35    8.1
## Cow               465.00  423.0
## Grey wolf          36.33  119.5
## Goat               27.66  115.0
## Guinea pig          1.04    5.5
## Dipliodocus     11700.00   50.0
## Asian elephant   2547.00 4603.0
## Donkey            187.10  419.0
## Horse             521.00  655.0
## Potar monkey       10.00  115.0
## Cat                 3.30   25.6
## Giraffe           529.00  680.0
```

```r
qplot(brain, body, data = Animals)
```



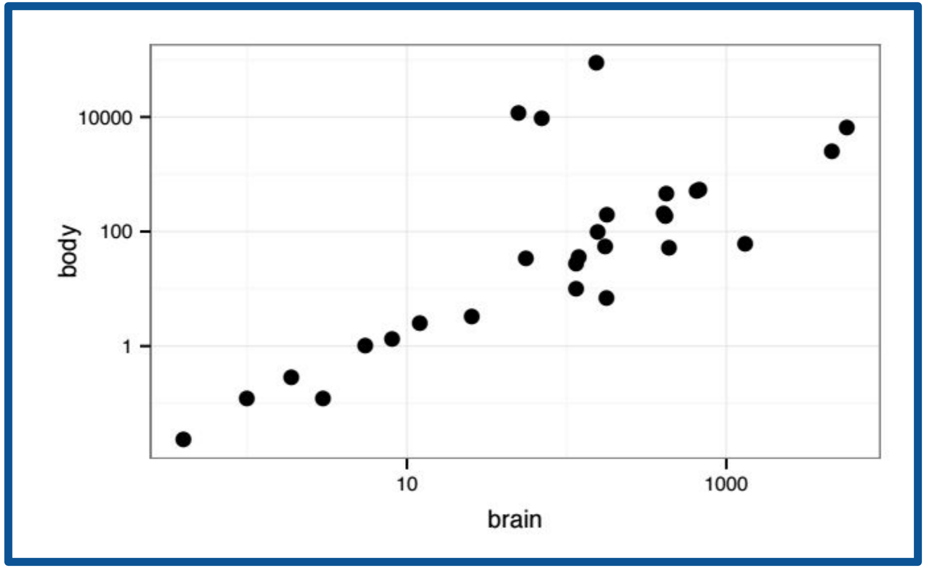
Power Quiz

Given the following data and its plot, **change the qplot command** to show the relationship between the body and brain mass on a log log scale

```
library(MASS)
print(Animals[1:12,])
```

```
##                     body  brain
## Mountain beaver     1.35    8.1
## Cow               465.00  423.0
## Grey wolf          36.33  119.5
## Goat               27.66  115.0
## Guinea pig          1.04    5.5
## Dipliodocus     11700.00   50.0
## Asian elephant   2547.00 4603.0
## Donkey            187.10  419.0
## Horse             521.00  655.0
## Potar monkey       10.00  115.0
## Cat                 3.30   25.6
## Giraffe           529.00  680.0
```

qplot(brain, body, log = "xy", data = Animals)



Data Transformations: Binning

Definitions:

Numeric variable: represents real valued measurements whose values are ordered in a manner consistent with the natural ordering of the real line.

Ordinal variable: represents measurements in a certain range R for which we have a well defined order relation.

Categorical variable: represents measurements that do not satisfy the ordinal or numeric assumption.

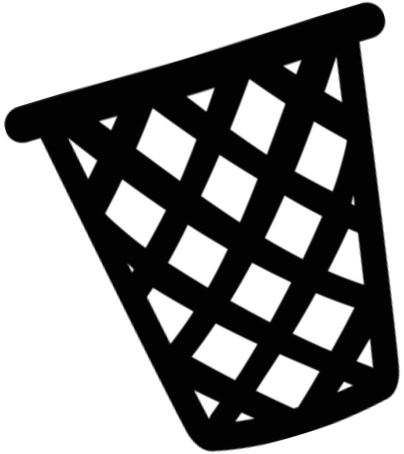
Data Transformations: Binning

Binning (also known as discretization): taking a numeric variable x 2 R (typically a real value, though it may be an integer), dividing its range into several bins, and replacing it with a number representing the corresponding bin.

Binarization: a special case (replaces a variable with either 0 or 1 depending on whether the variable is greater or smaller than a certain threshold).

Discretization in R can be done via the function cut.

Data Transformations: Binning



Binning Example:
Suppose x represent the tenure of an employee (in years) and ranges from 0 to 50.

Binning would divide the range into (0,10], (11,20], ...,(41, 50] and pick a representative number for each bin (for example middle point)

Data Transformations: Indicator Variables

Replace a variable x (numeric, ordinal, or categorical) taking k values with a binary k-dimensional vector v, such that v[i] (or v_i in mathematical notation) is one if and only if x takes on the i-value in its range.

Replace variable by vector that is all zeros, except for one component that equals one.

Data Transformations: Indicator Variables

Often, **indicator variables are used in conjunction with binning:** bin the variable into k bins and then create a k dimensional indicator variable.

High dimensional indicator vectors may be easily handled in computations by taking advantage of its extreme sparsity.

Data Transformations: Indicator Variables

Models for numeric or binary data cannot directly model ordinal or categorical data.

Transform the data using several non-linear transformations bin the transformed data, and create indicator vectors.

It is often **much easier to compute with indicator functions since they are binary**, and thus replacing numeric variables with indicator vectors may improve scalability.

Indicator Variables Quiz

A study on students and standardized test scores collected the following information: female, vocational, asian. Translate the variables to indicator variables.

Variable Sex:

male = 0

female = 1

Variable Program:

general = 0

vocational = 1

academic = 0

Variable Race:

hispanic = 0

asian = 1

african-american = 0

white= 0

Data Manipulations: Shuffling

A common operation in data analysis is to **select a random subset of the rows of a dataframe**, with or without replacement.

Data Manipulations: Shuffling

A common operation in data analysis is to **select a random subset of the rows of a dataframe**, with or without replacement.

sample() accepts a vector of values from which to sample (typically a vector of row indices), the number of samples, whether the sampling is done with or without replacement, and the probability of sampling different values.

sample(k,k) generates a random permutation of order k.

```
D = array(data = seq(1, 20, length.out = 20), dim = c(4, 5))
D_shuffled = D[sample(4, 4),]
```

Data Manipulations: Partitioning

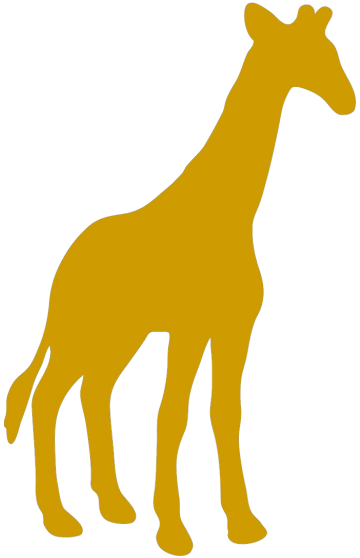
In some cases, **we need to partition the dataset's rows** into two or more collection of rows.

Generate a random permutation of k objects (using sample(k,k)), where k is the number of rows in the data, and then divide the permutation vector into two or more parts based on the prescribed sizes, and new dataframes whose rows correspond to the divided permutation vector.

Data Manipulations: Partitioning

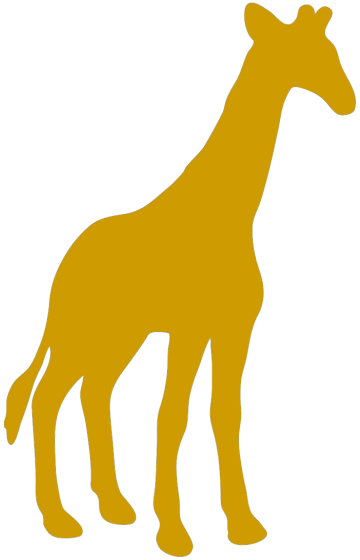
```
D = array(data = seq(1, 20, length.out = 20), dim = c(4, 5))
rand_perm = sample(4,4)
first_set_of_indices = rand_perm[1:floor(4*0.75)]
second_set_of_indices = rand_perm[(floor(4*0.75)+1):4]
D1 = D[first_set_of_indices,]
D2 = D[second_set_of_indices,]
```

Tall Data



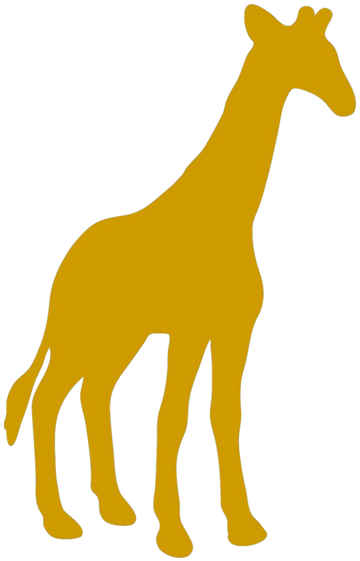
Data in tall format is an array or data frame containing multiple columns where one or more columns act as a unique identifier and an additional column represents value.

Tall Data



Date	Item	Quantity
2015/01/01	apples	200
2015/01/01	oranges	150
2015/01/02	apples	220
2015/01/02	oranges	130

Tall Data Quiz

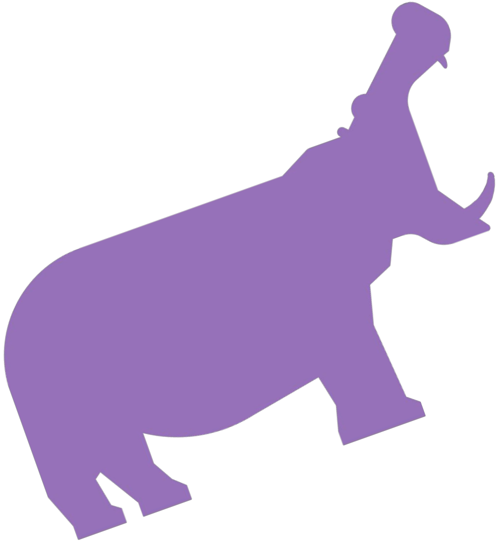


Check all the **true statements**:

- [ ] Tall data is not convenient for adding new records incrementally and for removing old records.

- [ ] The tall data format makes it easy to conduct analysis or summarizing.

Wide Data



Represents in multiple columns the information that tall data holds in multiple rows

Simpler to analyze

Harder to add/remove entries

Wide Data

2015/01/01	apples	200
2015/01/01	oranges	150
2015/01/02	apples	220
2015/01/02	oranges	130

Date	apples	oranges
2015/01/01	200	150
2015/01/02	220	130

- When converting tall data to wide data, we need to specify ID variables that define the row and column structure (date and item in the example above).

Reshaping Data

```
print(smiths)

##        subject time age weight height
## 1 John Smith    1  33     90   1.87
## 2 Mary Smith    1  NA     NA   1.54
```

```
smiths_tall = melt(smiths, id = 1)
print(smiths_tall[1:4,])

##        subject variable value
## 1 John Smith     time     1
## 2 Mary Smith     time     1
## 3 John Smith      age    33
## 4 Mary Smith      age    NA
```

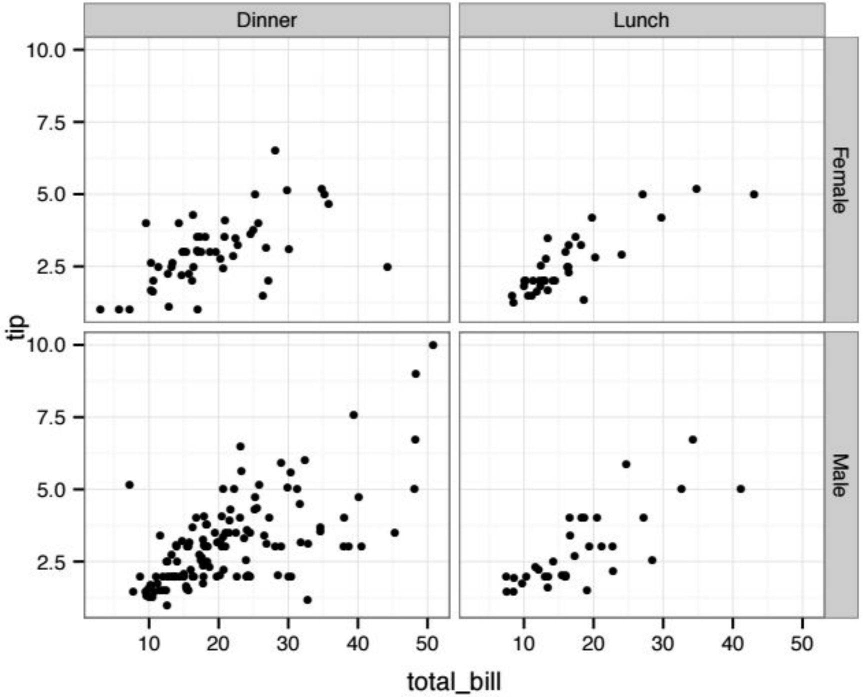
Reshaping Data

acast/dcast is the **inverse of melt**

The arguments are a dataframe in wide form, **a formula a ~ b ~ c … where each of a, b,… represents a sum of variables** whose values will be displayed along the dimensions of the returned array or data frame (a for rows, b for columns, etc.), and a function fun.aggregate that aggregates multiple values into a single value.

Reshaping Data

```
qplot(total_bill,
      tip,
      facets = sex~time,
      size = I(1.5),
      data = tips)
```



Smoker-Tip Example

The Variables:

Sex of the customer: Male, Female

Smoker: True, False

Day: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday

Time: Lunch, Dinner

Size: Number of people in party

Smoker-Tip Example

```
tipsm = melt(tips, id = c("sex","smoker","day","time","size"))
dcast(tipsm,  # Mean of measurement variables broken by sex
      sex~variable,
      fun.aggregate = mean)
```

```
##      sex total_bill   tip
## 1 Female      18.06 2.833
## 2   Male      20.74 3.090
```

Smoker-Tip Example

```
# Number of occurrences for measurement variables broken by sex
dcast(tipsm,
      sex~variable,
      fun.aggregate = length)
```

```
##      sex total_bill tip
## 1 Female         87  87
## 2   Male        157 157
```

Smoker-Tip Example

```r
# Average total bill and tip for different times
dcast(tipsm,
      time~variable,
      fun.aggregate = mean)
```

```
##      time total_bill   tip
## 1 Dinner      20.80 3.103
## 2  Lunch      17.17 2.728
```

Smoker-Tip Example

```
# Similar to above with breakdown for sex and time:
dcast(tipsm,
      sex+time~variable,
      fun.aggregate = length)
```

```
##      sex   time total_bill tip
## 1 Female Dinner         52  52
## 2 Female  Lunch         35  35
## 3   Male Dinner        124 124
## 4   Male  Lunch         33  33
```

Smoker-Tip Example

```
# Similar to above, but with mean and added margins
dcast(tipsm,
      sex+time~variable,
      fun.aggregate = mean,
      margins = TRUE)
```

```
##      sex   time total_bill   tip  (all)
## 1 Female Dinner      19.21 3.002 11.108
## 2 Female  Lunch      16.34 2.583  9.461
## 3 Female  (all)      18.06 2.833 10.445
## 4   Male Dinner      21.46 3.145 12.303
## 5   Male  Lunch      18.05 2.882 10.465
## 6   Male  (all)      20.74 3.090 11.917
## 7  (all)  (all)      19.79 2.998 11.392
```

Smoker-Tip Quiz

Based on the output from the Smoker-Tip Example that we just discussed, **which of the following statements are true**?

- [x] On average, males pay higher total bill and tip than females.
- [ ] Females pay more frequently than males.
- [x] Dinner bills and tips are generally higher than lunch bills and tips.
- [x] Males pay disproportionately more times for dinner than they do for lunch (this holds much less for females).
- [ ] Accounting for the above statement. By conditioning on paying for lunch or dinner, males do not pay higher total bills and tips than females.

Split-Apply-Combine

Many data analysis operations on dataframes can be decomposed to three stages:

1. splitting the dataframe along some dimensions to form smaller arrays or dataframes,

2. applying some operation to each of the smaller arrays or dataframes, and

3. combining the results of the application stage into a single meaningful array or dataframe.

Split-Apply-Combine
The Plyr Package

output / input	array	dataframe	list	discarded
array	aaply	adply	alply	a_ply
dataframe	daply	ddply	dlply	d_ply
list	laply	ldply	llply	l_ply

- **Arguments:** data, dimensions/columns used to to split the data, function to execute in the apply stage.

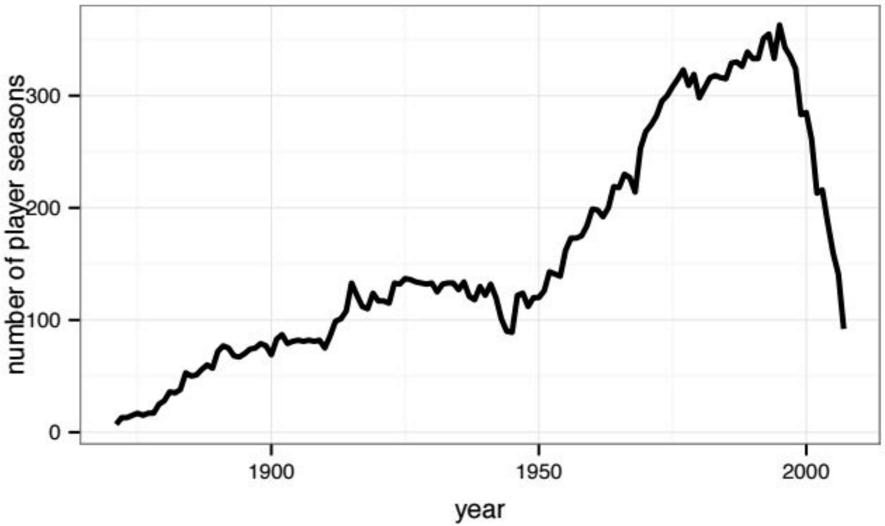
Split-Apply-Combine

The Plyr Package

```
library(plyr)
names(baseball)

##  [1] "id"    "year"  "stint" "team"  "lg"    "g"     "ab"
##  [9] "h"     "X2b"   "X3b"   "hr"    "rbi"   "sb"    "cs"
## [17] "so"    "ibb"   "hbp"   "sh"    "sf"    "gidp"

# count number of players recorded for each year
bbPerYear = ddply(baseball, "year", "nrow")
head(bbPerYear)

##   year nrow
## 1 1871    7
## 2 1872   13
## 3 1873   13
## 4 1874   15
## 5 1875   17
## 6 1876   15
```

Baseball Example

```
qplot(x = year, y = nrow,
      data = bbPerYear, geom = "line",
      ylab="number of player seasons")
```

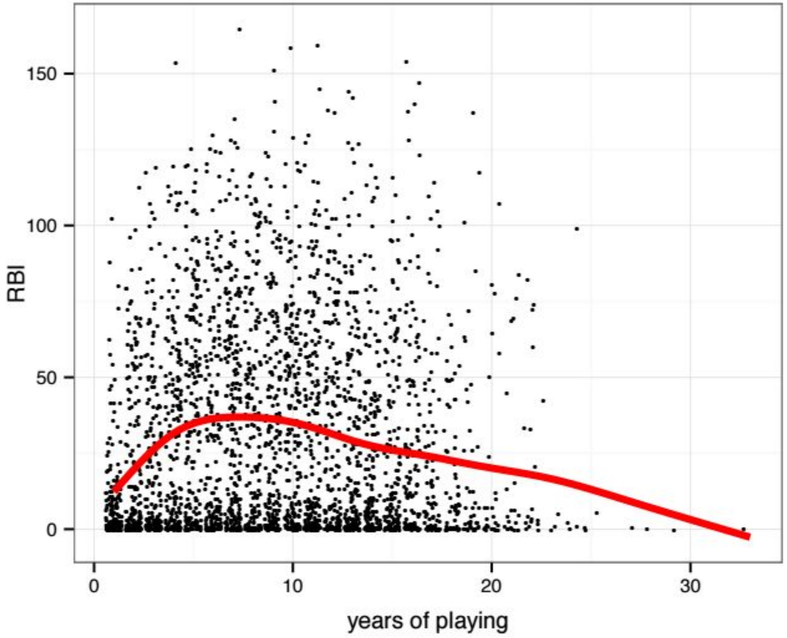


Baseball Example

```r
# compute mean rbi (batting attempt resulting in runs)
# for all years. Summarize is the apply function, which
# takes as argument a function that computes the rbi mean
bbMod=ddply(baseball, "year", summarise,
            mean.rbi = mean(rbi, na.rm = TRUE))
qplot(x = year, y = mean.rbi, data = bbMod,
      geom = "line", ylab = "mean RBI")
```

Baseball Example

```
# add a column career.year which measures the number of years
# passed since each player started batting
bbMod2 = ddply(baseball,
               "id",
               transform,
               career.year = year - min(year) + 1)
# sample a random subset 3000 rows to avoid over-plotting
bbSubset = bbMod2[sample(dim(bbMod2)[1], 3000),]
qplot(career.year,
      rbi, data = bbSubset,
      size = I(0.8),
      geom = "jitter",
      ylab = "RBI",
      xlab = "years of playing") +
  geom_smooth(color = "red", se = F, size = 1.5)
```

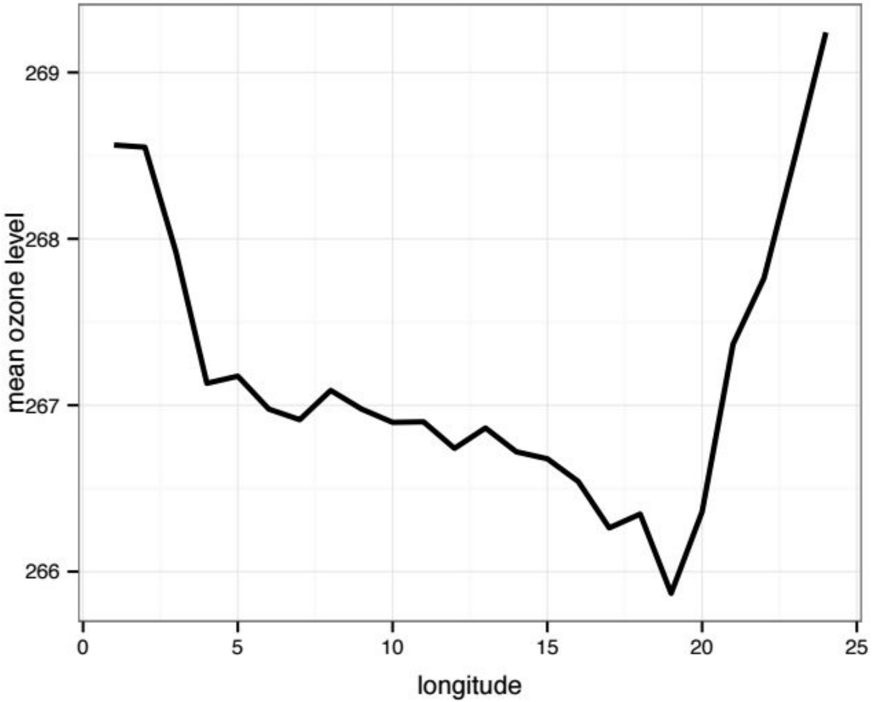


Ozone Example

- The ozone dataset contains a **3-dimensional array of ozone measurements** varying by latitude, longitude, and time.

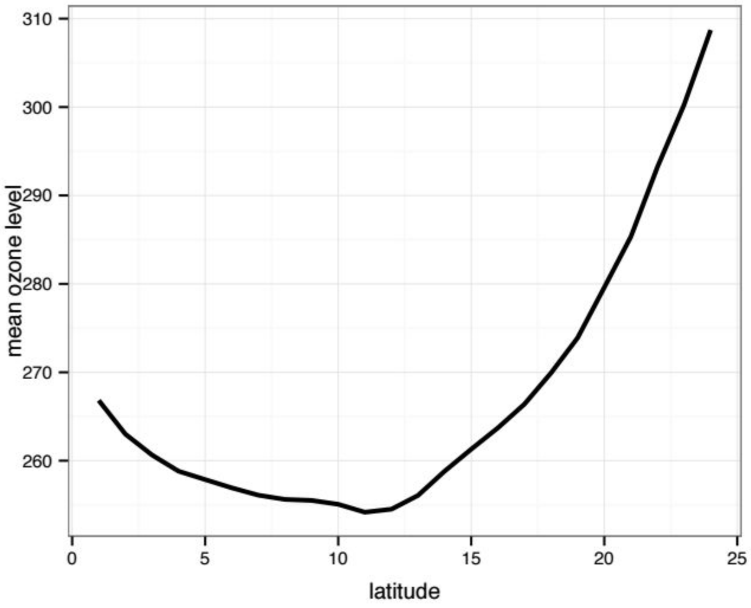
```
library(plyr)
latitude.mean = aaply(ozone, 1, mean)
longitude.mean = aaply(ozone, 2, mean)
time.mean = aaply(ozone, 3, mean)
longitude = seq(along = longitude.mean)
qplot(x = longitude,
      y = longitude.mean,
      ylab = "mean ozone level",
      geom="line")
```

Ozone Example



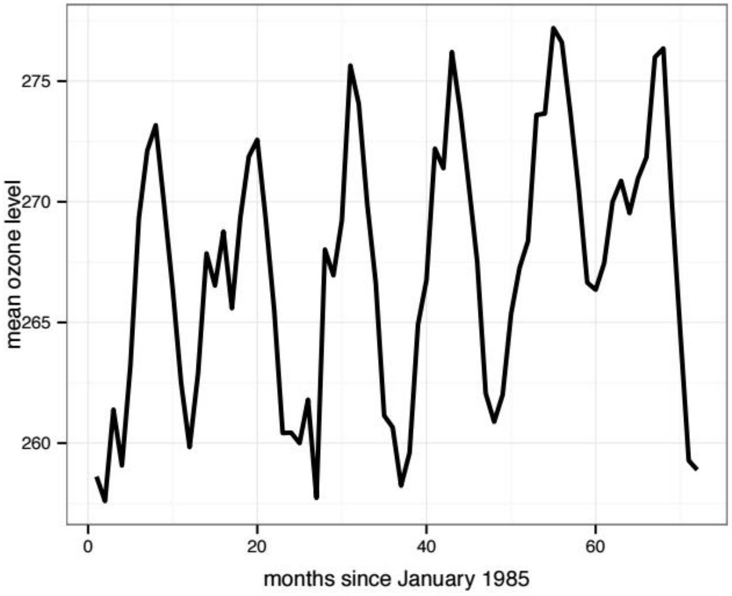
Ozone Example

```
latitude = seq(along = latitude.mean)
qplot(x = latitude,
      y = latitude.mean,
      ylab = "mean ozone level",
      geom = "line")
```



Ozone Example

```
months = seq(along = time.mean)
qplot(x = months,
      y = time.mean,
      geom = "line",
      ylab = "mean ozone level",
      xlab = "months since January 1985")
```



Ozone Quiz

Based on the outputs from the Ozone Example that we just discussed, **which of the following statements are true**?

- [ ] Ozone has a clear minimum mean ozone level at longitude 12 and latitude 19
- [ ] Ozone level has an interesting temporal periodicity superimposed with a slight decreasing trend.
- [x] The periodicity coincides with the annual season cycle (each period is 12 months)
- [x] The functions in the plyr package are very general and simplify the coding of many data analysis tasks.

Preprocessing Data
Lesson Summary

- It is important to know how to handle missing data and outliers
- Transforming the data can reveal insights and improve modeling
- Standard data manipulation techniques can be automated by tools in the R language