DSC 10, Spring 2018
Lecture 28

Review

sites.google.com/eng.ucsd.edu/dsc-10-spring-2018

Credit: Anindita Adhikari and John DeNero

Announcements

- My office hours are Saturday 3-5pm
    - TA office hours for finals week are different
    - See course calendar
- Project 10 due Saturday, 11:59pm
    - Please re-click download link to get updated tests
    - See Piazza to fix known issue with questions 3.1, 3.2
- Please fill out CAPE
    - This is a new major and your feedback matters!
- Final exam is Friday, June 15 from 3-6pm

Final Exam

- Practice materials
  - Blank assignments - see Piazza for link
  - Practice final exams on course website
  - Review Session
    - Tuesday 6-6:50pm in Center 216
- Reference sheet will be provided, see it on course website
- Bring student ID and pen/pencil
  - No computers, calculators, phones, etc.
- Assigned seats will be posted before the exam
  - Split among two rooms

Cause and Effect

A technology start-up is looking to hire new employees so they interview a bunch of recently graduated UCSD students.

A follow-up survey asks the students about their interview experiences, and it is determined that there is a negative association between time spent preparing for the interview and successful performance in the interview.

What could explain this?

Python

Conditionals

```
a = True
b = False
c = True

if (a and b):
    print("First")
elif ((b or c) and not(a)):
    print("Second")
if (not c or not a):
    print ("Third")
```

What will be printed?

A. First
B. Second
C. Third
D. First Second
E. Nothing

Loops

```
for i in make_array(1,2,3,4,5):
    i=i+3
    print(i*2)
```

What would the loop above display?

A: 2, 4, 6, 8, 10

B: 4, 6, 8, 10, 12

C: 8, 10, 12, 14, 16

D: 6, 8, 10, 12 14

E: 8, 16

```
for i in make_array(1,2,3,4,5):
    i=i+3
    print(“Hey!”)
```

What would the loop above display?

A: “Hey!” (1 time)

B: “Hey!” (2 times)

C: “Hey!” (3 times)

D: “Hey!” (5 times)

E: Nothing

Functions

- Functions encapsulate code so that you (or a higher-order function like apply) can use it over and over without rewriting it.

- Structure of a function definition:

```
def <function name>(<zero or more arguments>):
    <body>
    return <some value> # optional
```

Try writing a function

Write a function called `hungry_adder` that does all of these things:

- takes two numbers as arguments
- adds them together and returns the value
- prints the string “Thanks for feeding me those tasty numbers!”

Arrays

- All entries have the same type.

- Review the various array functions (np.sum, np.mean, etc.)

- Arithmetic is component-wise.
  - Arrays must be the same length.
  - `2 * make_array([1, 2, 3]) = array([2, 4, 6])`
  - `make_array([2, 3, 5]) + make_array([7, 8, 9]) = array([9, 11, 14])`

Table Manipulations

Join

students

StudentID	Class	GPA
12345	CSE12	3.3
67890	CSE12	2.3
67890	CSE21	3.5

prof

Course	Professor
CSE12	Alvarado
CSE21	Jones

What is the output?

```
students.join('Class', prof, 'Course')
```

Join

students

StudentID	Class	GPA
12345	CSE12	3.3
67890	CSE12	2.3
67890	CSE21	3.5

prof

Course	Professor
CSE12	Alvarado
CSE21	Jones

Does order matter?

- `students.join('Class', prof, 'Course')`
- `prof.join('Course', students, 'Class')`

Group

table.group(“ColName”, function)

- Group table by categories
    - 1st argument: column we want to group by
        - split column into unique values
    - 2nd argument: (optional) aggregate function
        - If no 2nd argument, then just count # of entries for each category
        - sum, mean, min, max, count, list, abs… or make your own function and pass it in!

Group

students

StudentID	Class	GPA
12345	CSE12	3.3
67890	CSE12	2.3
67890	CSE21	3.5

prof

Course	Professor
CSE12	Alvarado
CSE21	Jones

What is the resulting table?

```
students.group(“Class”, max)
```

Practice Problem 1

We have a table **`exports`** containing the export amounts (in millions of dollars) of various agricultural products from California in the year 2017.

- The first column is labeled “Product.”
- The second column is labeled “Amount”.

Write a line of code that evaluates to

a) the average of the amount column.

b) a table with only the products that exported at an above average amount.

c) True if any exports are *less* than 1 million dollars.

d) the proportion of exports that are *between* 50 and 100 million dollars.

Solutions

a) the average of the amount column.

```
np.mean(exports.column(‘amount’))
```

b) a table with only the products that are exported at an above average amount.

```
exports.where(‘amount’, are.above(np.mean(exports.column(‘amount)))
```

c) True if any exports are *less* than 1 million dollars.

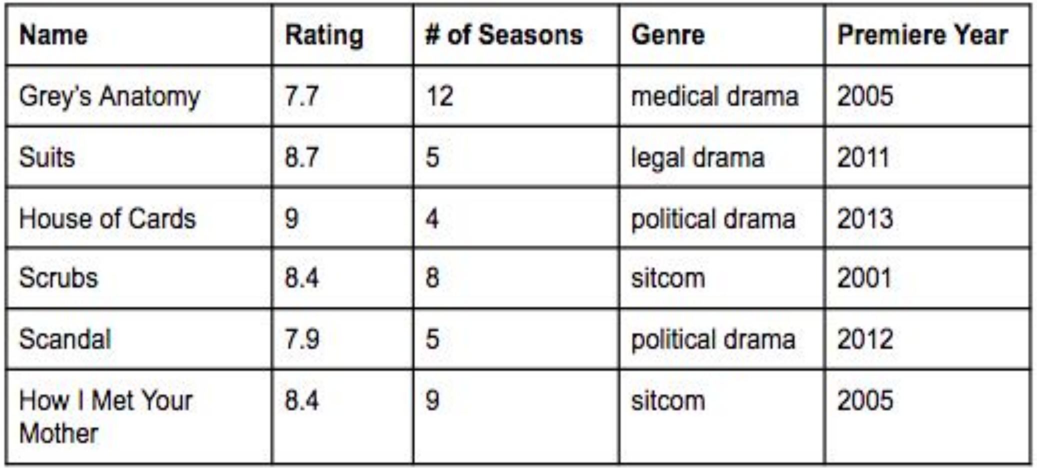
```
exports.sort(‘amount’).column(‘amount’).item(0) < 1
```

d) the proportion of exports that are *between* 50 and 100 million dollars.

```
exports.where(‘amount’, are.between(50,100)).num_rows / exports.num_rows
```

TV Options

Alice wants to pick a new TV show to watch, and decides to use some data she found online about various TV shows to make a decision.



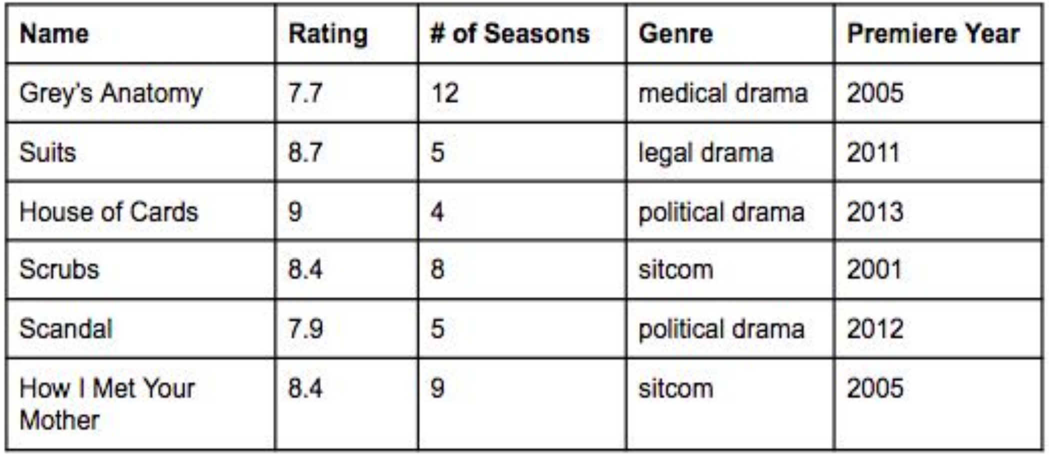
Name	Rating	# of Seasons	Genre	Premiere Year
Grey's Anatomy	7.7	12	medical drama	2005
Suits	8.7	5	legal drama	2011
House of Cards	9	4	political drama	2013
Scrubs	8.4	8	sitcom	2001
Scandal	7.9	5	political drama	2012
How I Met Your Mother	8.4	9	sitcom	2005

The table of TV shows she is using is called `tv_shows`, and each row corresponds to a unique show. The first 6 rows of the table are shown above.

Practice Problem 2

Alice decides to filter the table to only include shows that satisfy both of the following conditions:

- At least an 8.0 rating.
- At least 6 seasons.



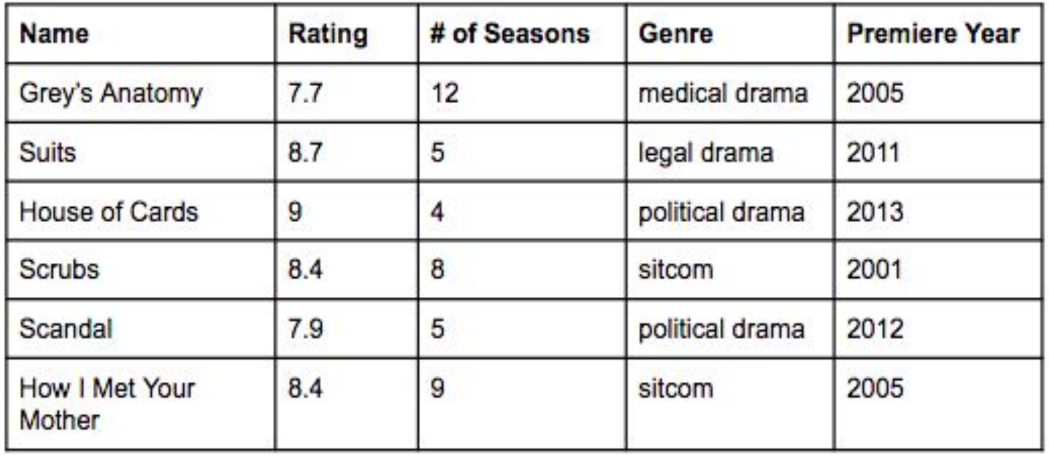
Name	Rating	# of Seasons	Genre	Premiere Year
Grey's Anatomy	7.7	12	medical drama	2005
Suits	8.7	5	legal drama	2011
House of Cards	9	4	political drama	2013
Scrubs	8.4	8	sitcom	2001
Scandal	7.9	5	political drama	2012
How I Met Your Mother	8.4	9	sitcom	2005

Write an expression to produce a table with only shows that satisfy both of these conditions.

Solution

Alice decides to filter the table to only include shows that satisfy both of the following conditions:

- At least an 8.0 rating.
- At least 6 seasons.



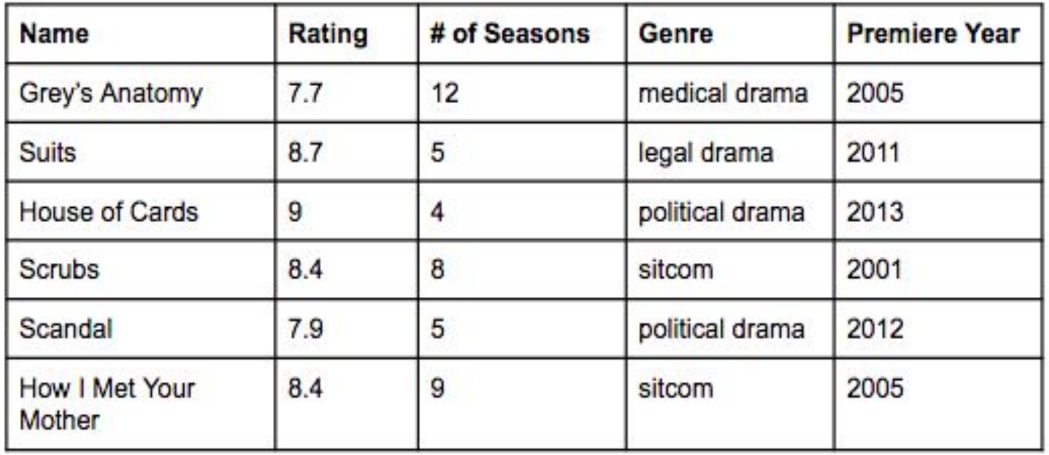
Name	Rating	# of Seasons	Genre	Premiere Year
Grey's Anatomy	7.7	12	medical drama	2005
Suits	8.7	5	legal drama	2011
House of Cards	9	4	political drama	2013
Scrubs	8.4	8	sitcom	2001
Scandal	7.9	5	political drama	2012
How I Met Your Mother	8.4	9	sitcom	2005

Write an expression to produce a table with only shows that satisfy both of these conditions.

```
tv_shows.where(‘Rating’, are.above_or_equal_to(8)).where('# of Seasons', are.above_or_equal_to(6))
```

Practice Problem 3

Alice realizes that she doesn't even know what type of show she wants to watch, so she decides to look at what genres people seem to like best.

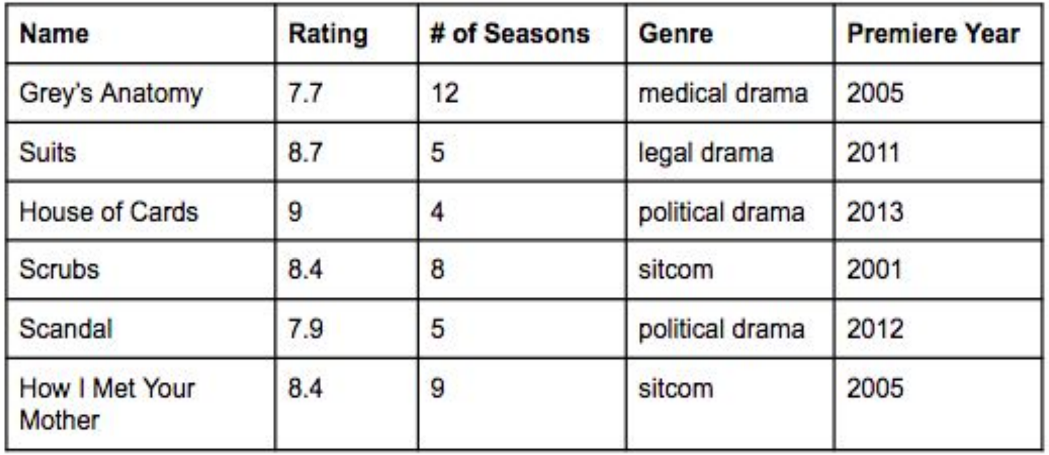


Name	Rating	# of Seasons	Genre	Premiere Year
Grey's Anatomy	7.7	12	medical drama	2005
Suits	8.7	5	legal drama	2011
House of Cards	9	4	political drama	2013
Scrubs	8.4	8	sitcom	2001
Scandal	7.9	5	political drama	2012
How I Met Your Mother	8.4	9	sitcom	2005

Find the genre with the highest average rating and assign it to the variable `best_genre`.

Solution

Alice realizes that she doesn't even know what type of show she wants to watch, so she decides to look at what genres people seem to like best.



Name	Rating	# of Seasons	Genre	Premiere Year
Grey's Anatomy	7.7	12	medical drama	2005
Suits	8.7	5	legal drama	2011
House of Cards	9	4	political drama	2013
Scrubs	8.4	8	sitcom	2001
Scandal	7.9	5	political drama	2012
How I Met Your Mother	8.4	9	sitcom	2005

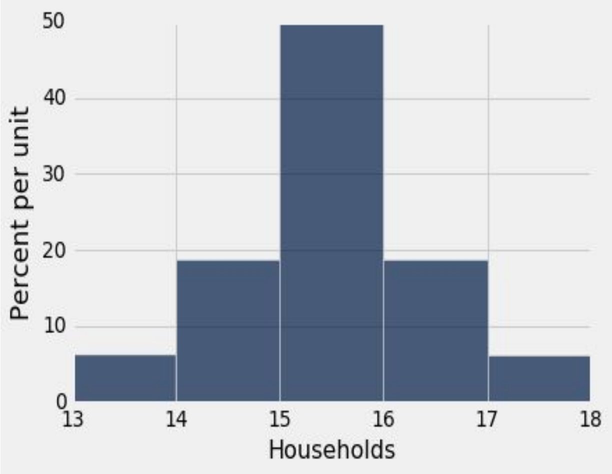
Find the genre with the highest average rating and assign it to the variable `best_genre`.

```
best_genre =  tv_shows.group('Genre', np.mean)
                      .sort('Rating mean', descending=True)
                      .column('Genre').item(0)
```

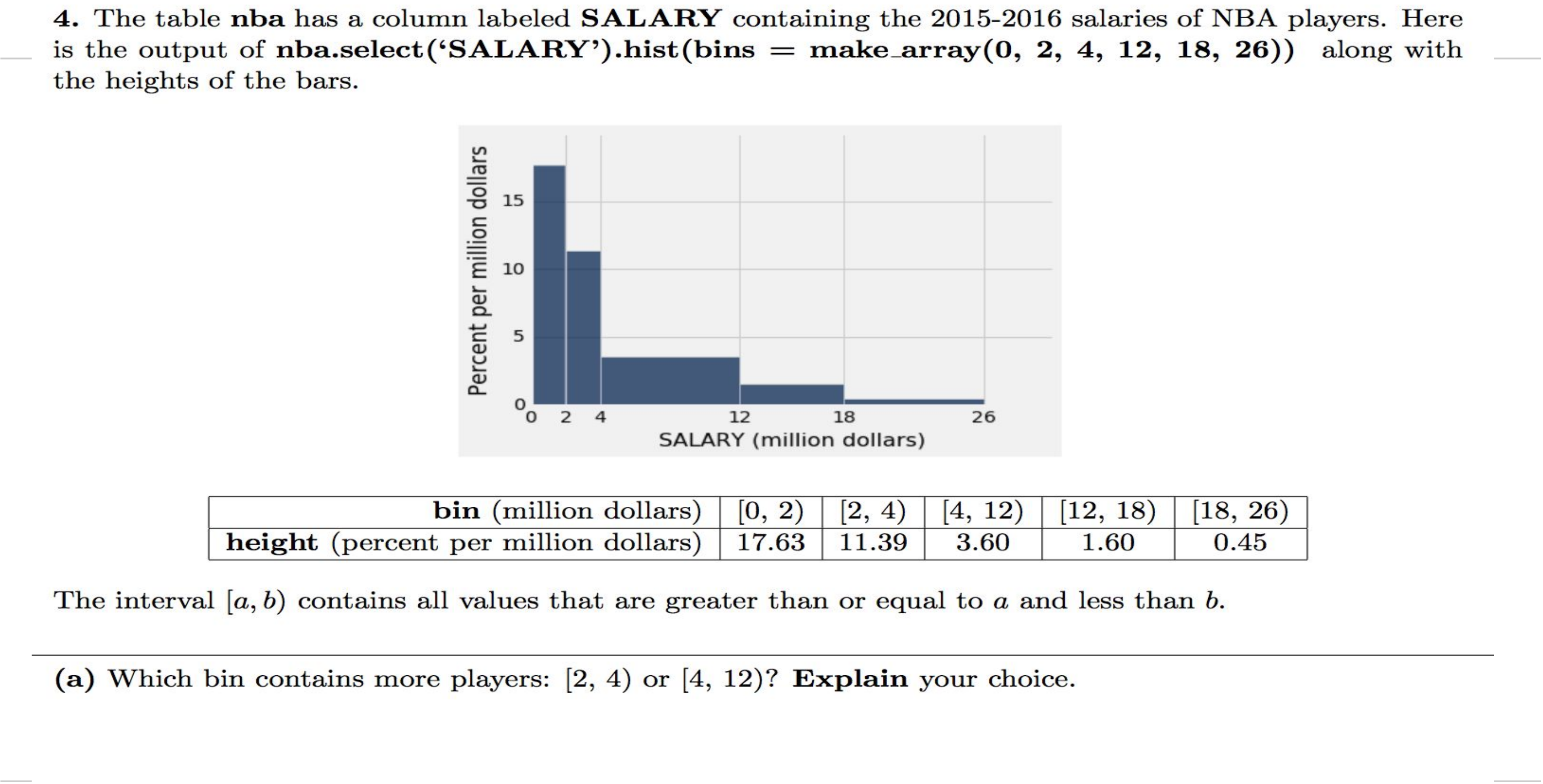
Histograms

Histograms

- Total area of the bars = 1 (100%)
- Area represents proportion
- Area of bar = height of bar * width of bar



Practice Problem 4



4. The table **nba** has a column labeled **SALARY** containing the 2015-2016 salaries of NBA players. Here is the output of **nba.select('SALARY').hist(bins = make_array(0, 2, 4, 12, 18, 26))** along with the heights of the bars.

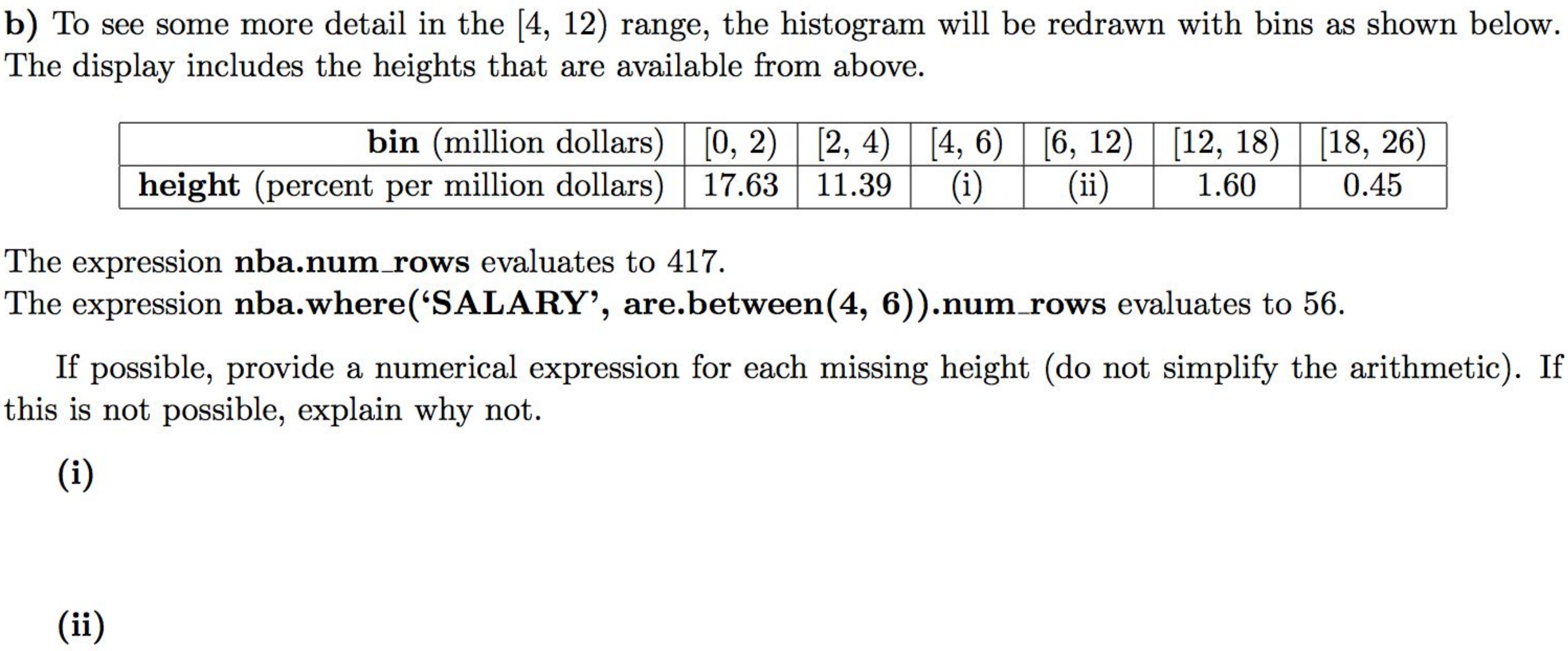
Percent per million dollars
0
5
10
15
0
2
4
12
18
26
SALARY (million dollars)

bin (million dollars)	[0, 2)	[2, 4)	[4, 12)	[12, 18)	[18, 26)
height (percent per million dollars)	17.63	11.39	3.60	1.60	0.45

The interval $[a, b)$ contains all values that are greater than or equal to a and less than b.

(a) Which bin contains more players: [2, 4) or [4, 12)? **Explain** your choice.

Practice Problem 5



b) To see some more detail in the [4, 12) range, the histogram will be redrawn with bins as shown below. The display includes the heights that are available from above.

bin (million dollars)	[0, 2)	[2, 4)	[4, 6)	[6, 12)	[12, 18)	[18, 26)
height (percent per million dollars)	17.63	11.39	(i)	(ii)	1.60	0.45

The expression **nba.num_rows** evaluates to 417.
The expression **nba.where('SALARY', are.between(4, 6)).num_rows** evaluates to 56.

If possible, provide a numerical expression for each missing height (do not simplify the arithmetic). If this is not possible, explain why not.

(i)

(ii)

Estimation

Sampling: Parameters and Statistics

- Parameter: numerical quantity associated with population
- Statistic: number computed from data in a sample
- We use a sample statistic to estimate a population parameter.
- Repeatedly sampling gives us a sense of the variability of our estimate.

Bootstrapping, Confidence Intervals

- When impractical to sample repeatedly, bootstrap:
    - Sample and then resample from the original sample
    - Each resample is the same size as original, with replacement
- Confidence intervals estimate the true value of a population parameter to be within some interval.
- This estimate is based on many values of a statistic generated by repeated sampling.

Central Limit Theorem

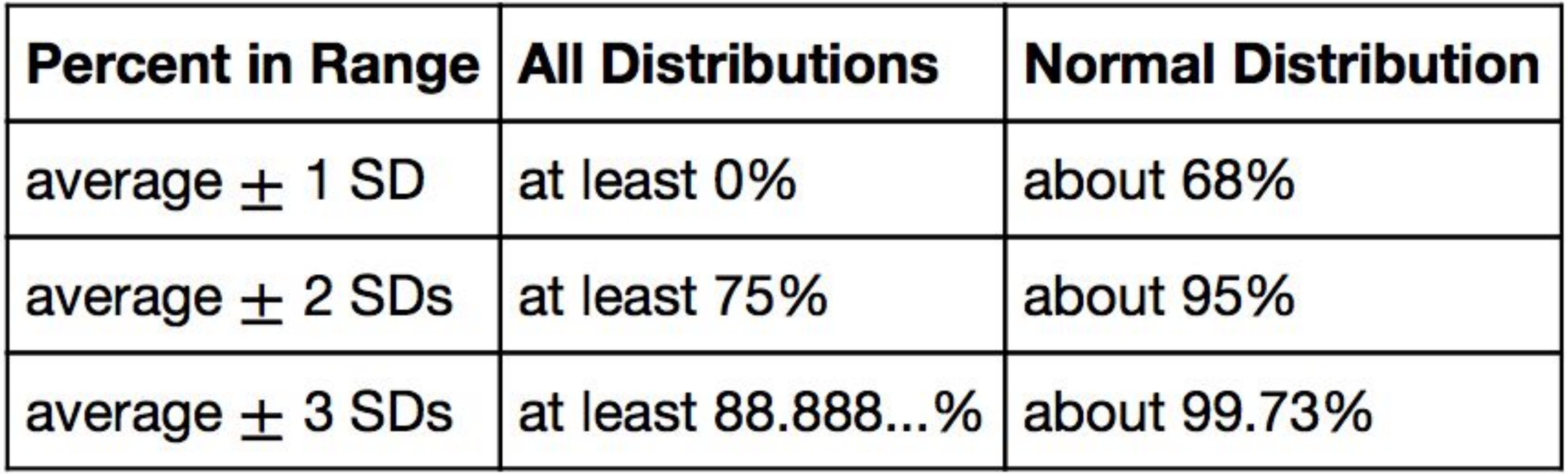
- The distribution of the sample mean (or sum or proportion) will be roughly normal.

$$\text{sample mean's average} = \text{population average}$$

$$\text{sample mean's SD} = (\text{population SD}) \,/\, \sqrt{\text{sample size}}$$

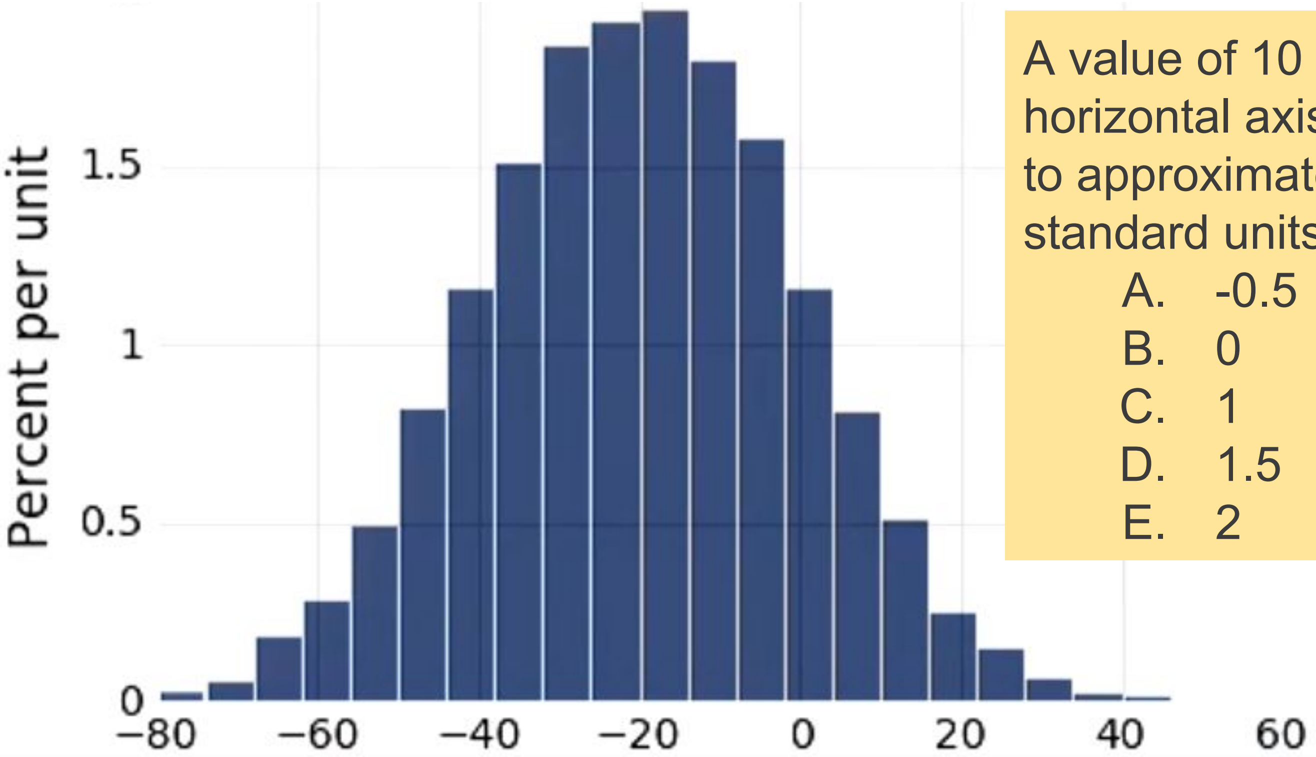
- Can solve for the sample size needed to keep sample mean's SD sufficiently small, which keeps confidence interval narrow

Bounds and Normal Approximations



Percent in Range	All Distributions	Normal Distribution
average ± 1 SD	at least 0%	about 68%
average ± 2 SDs	at least 75%	about 95%
average ± 3 SDs	at least 88.888...%	about 99.73%

Practice Problem 6



A value of 10 on the horizontal axis corresponds to approximately how many standard units?

A. -0.5
B. 0
C. 1
D. 1.5
E. 2

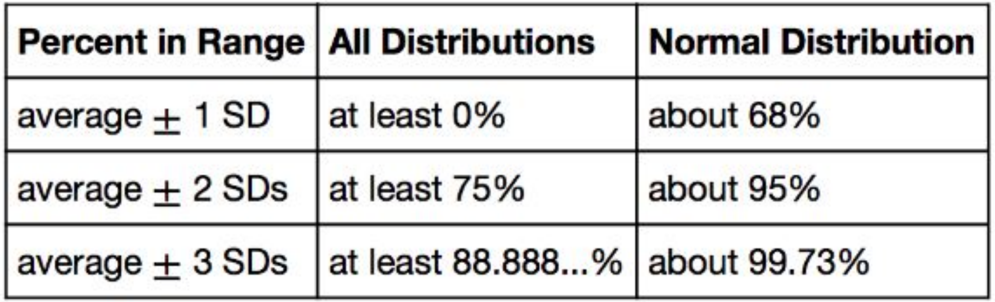
Practice Problem 7

What will happen to our confidence interval if we decrease the confidence level, while keeping the sample size the same?

A. Will become wider
B. Will become narrower
C. Will stay the same
D. Cannot be determined from the given information

sample mean's average = population average

sample mean's SD = (population SD) / $\sqrt{\text{sample size}}$



Percent in Range	All Distributions	Normal Distribution
average ± 1 SD	at least 0%	about 68%
average ± 2 SDs	at least 75%	about 95%
average ± 3 SDs	at least 88.888...%	about 99.73%

Practice Problem 8

What will happen to our confidence interval if we increase both the confidence level and the sample size?

A. Will become wider

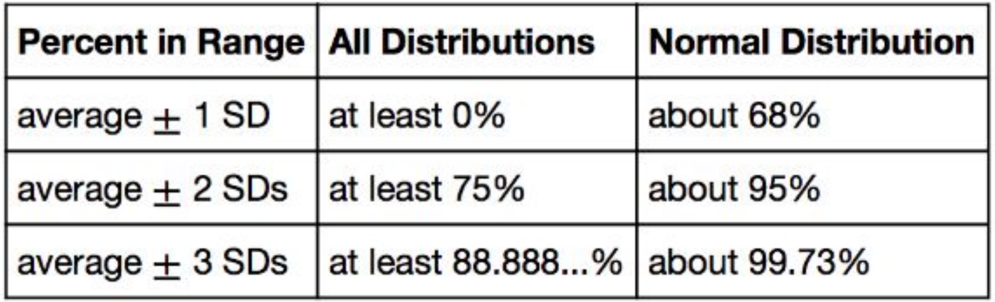
B. Will become narrower

C. Will stay the same

D. Cannot be determined from the given information

sample mean's average = population average

sample mean's SD = (population SD) / $\sqrt{\text{sample size}}$



Percent in Range	All Distributions	Normal Distribution
average ± 1 SD	at least 0%	about 68%
average ± 2 SDs	at least 75%	about 95%
average ± 3 SDs	at least 88.888...%	about 99.73%

Practice Problem 9

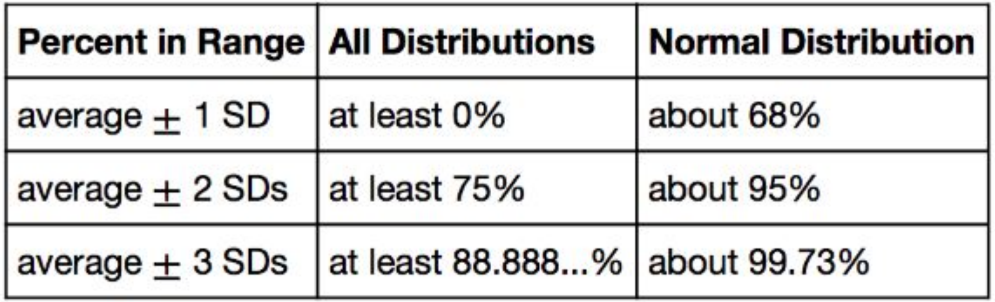
The time it takes to bake a pan of brownies is normally distributed, with a mean of 36 minutes and a standard deviation of 3 minutes.

What percentage of brownies bake in 30 minutes or less?

A. 2.5%
B. 5%
C. 95%
D. 97.5%
E. Cannot be determined

sample mean’s average = population average

sample mean’s SD = (population SD) / $\sqrt{\text{sample size}}$



Percent in Range	All Distributions	Normal Distribution
average ± 1 SD	at least 0%	about 68%
average ± 2 SDs	at least 75%	about 95%
average ± 3 SDs	at least 88.888...%	about 99.73%

Practice Problem 10

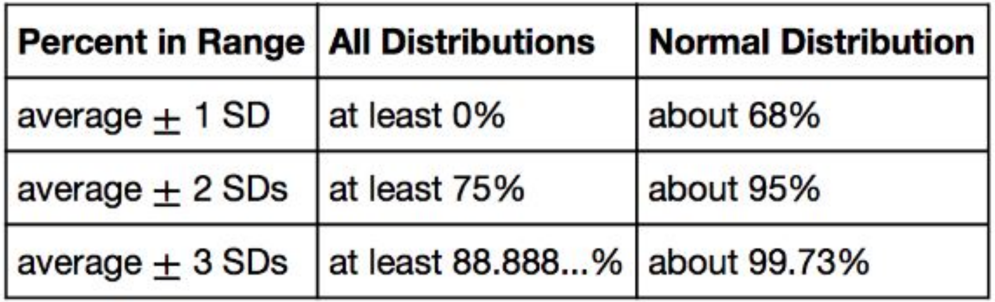
The time it takes to bake a pan of brownies is normally distributed, with a mean of 36 minutes and a standard deviation of 3 minutes.

I bake 36 pans of brownies and calculate the average baking time. What is the probability that this average is over 35 minutes?

A. 2.5%
B. 5%
C. 95%
D. 97.5%
E. Cannot be determined

sample mean's average = population average

sample mean's SD = (population SD) / $\sqrt{\text{sample size}}$



Percent in Range	**All Distributions**	**Normal Distribution**
average ± 1 SD	at least 0%	about 68%
average ± 2 SDs	at least 75%	about 95%
average ± 3 SDs	at least 88.888...%	about 99.73%

Regression

- Know how to use the formula for r, standard units, slope and intercept

- Know the relationship between residuals and predictions

- Know how to calculate (root) mean squared error

- Know how to use minimize, what it takes and what it returns

Regression

The correlation between child height and midparent height is 0.3. The equation of a regression line for estimating a child's height based on midparent height, in original units of inches, has

- Slope 0.6
- Intercept 23

Estimate the height of someone whose midparent height is 70 inches.

Classification

- Binary classification based on attributes
  - k-nearest neighbor classifiers
- Training and test sets
  - Why these are needed
  - How to generate them
- Implementation:
  - Distance between two points
  - Class of the majority of the k nearest neighbors
- Accuracy: Proportion of test set correctly classified

Classification

- Be careful when interpreting distance visually when graph axes are of different scale.
- What happens when k is too large?
  - Why is it a problem?
- What happens when k is even?
  - Why is it a problem?