SciencesPo
DEPARTMENT OF ECONOMICS

ScPoEconometrics

Sampling

Florian Oswald, Gustave Kenedi and Pierre Villedieu
SciencesPo Paris
2020-03-31

Recap from last week

- *Multiple Linear Regression Model*: $y_i = b_0 + b_1 x_{1,i} + \cdots + b_k x_{k,i} + e_i$
- Interpretation: effect holding all other independent variables constant
- Important extensions: *standardized regression*, *log models*, *interaction terms*

Recap from last week

- *Multiple Linear Regression Model*: $y_i = b_0 + b_1 x_{1,i} + \cdots + b_k x_{k,i} + e_i$
- Interpretation: effect holding all other independent variables constant
- Important extensions: *standardized regression*, *log models*, *interaction terms*

Today[1]

- Fun activity to discover sampling, sampling variation and sampling distributions.
- Sampling terminology: population, sample, population parameter, point estimate or sample statistic, etc.
- Definition of an ***unbiased estimator***.
- Fundamental statistical theorem for inference: ***Central Limit Theorem***.

[1]: This lecture is very heavily based on the wonderful sampling chapter of ModernDive

What's the proportion of green pasta?

What's the proportion of green pasta?

We could count every green pasta but that would be tedious! 😩 What else could we do?

Sampling

- Let's take a sample of 20 pasta.
- We made sure to select them at **random**.
- Here is what we found.

Color	Count	Proportion
Green	14	0.70
Red	5	0.25
Yellow	1	0.05

- 0.70 can be thought of as our guess of the proportion of green pasta in the entire bowl.

Sampling Variation

- What would happen if we took a *new* sample (putting the 20 previous pasta back in the bowl)? Would we also get 14 *greens* as before?

Sampling Variation

- What would happen if we took a *new* sample (putting the 20 previous pasta back in the bowl)? Would we also get 14 *greens* as before?
- What if we repeated this activity multiple times?

Sampling Variation

- What would happen if we took a *new* sample (putting the 20 previous pasta back in the bowl)? Would we also get 14 *greens* as before?
- What if we repeated this activity multiple times?
- Probably not. The samples will vary from draw to draw.

Sampling Variation

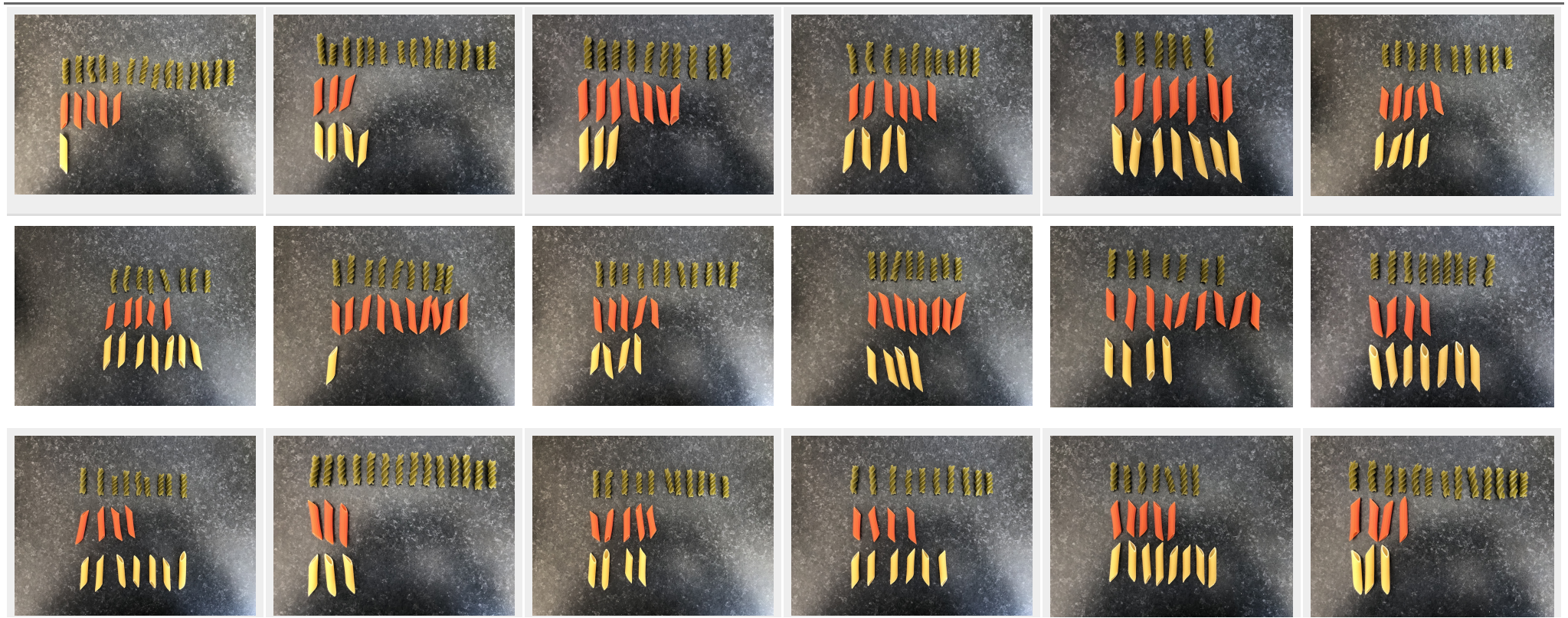
- What would happen if we took a *new* sample (putting the 20 previous pasta back in the bowl)? Would we also get 14 *greens* as before?
- What if we repeated this activity multiple times?
- Probably not. The samples will vary from draw to draw.
- Key to this observation: these are *randomly* drawn samples.

Taking 18 Samples (One per Student)

- Because we can't do this activity in class, we drew 18 samples of 20 pasta (with replacement).

Taking 18 Samples (One per Student)

- Because we can't do this activity in class, we drew 18 samples of 20 pasta (with replacement).
- This is what each looked like:



Taking 18 Samples (One per Student)

- Because we can't do this activity in class, we drew 18 samples of 20 pasta (with replacement) at home.
- For each sample, we computed the share of green pasta.

Sample #	Count	Proportion
1	14	0.70
2	14	0.70
3	10	0.50
4	10	0.50
5	6	0.30
6	10	0.50
7	8	0.40
8	9	0.45
9	11	0.55

Sample #	Count	Proportion
10	8	0.40
11	7	0.35
12	9	0.45
13	9	0.45
14	14	0.70
15	11	0.55
16	10	0.50
17	7	0.35
18	13	0.65

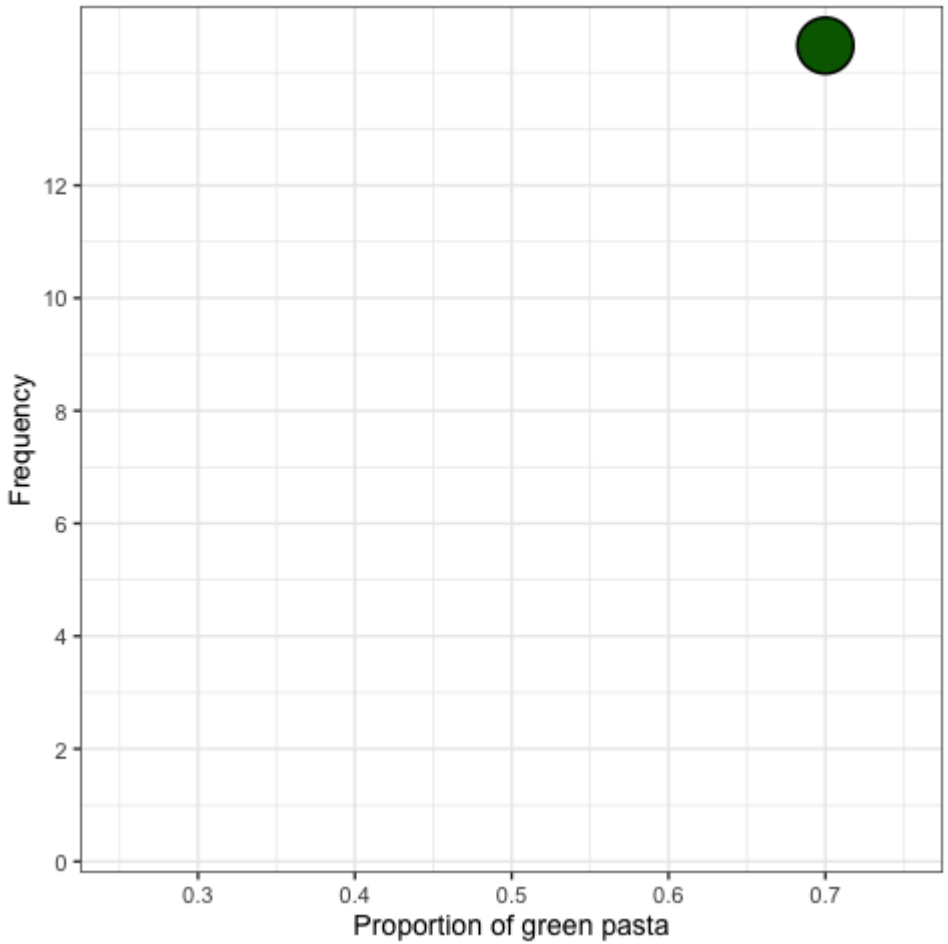
Task 1 (10 minutes)

1. Create a data.frame containing the proportions of green pasta from the previous slide. Name it `pasta` and name the variable containing the proportions `prop_green`. (Hint: to create a data.frame you need to use the `data.frame()` function.)

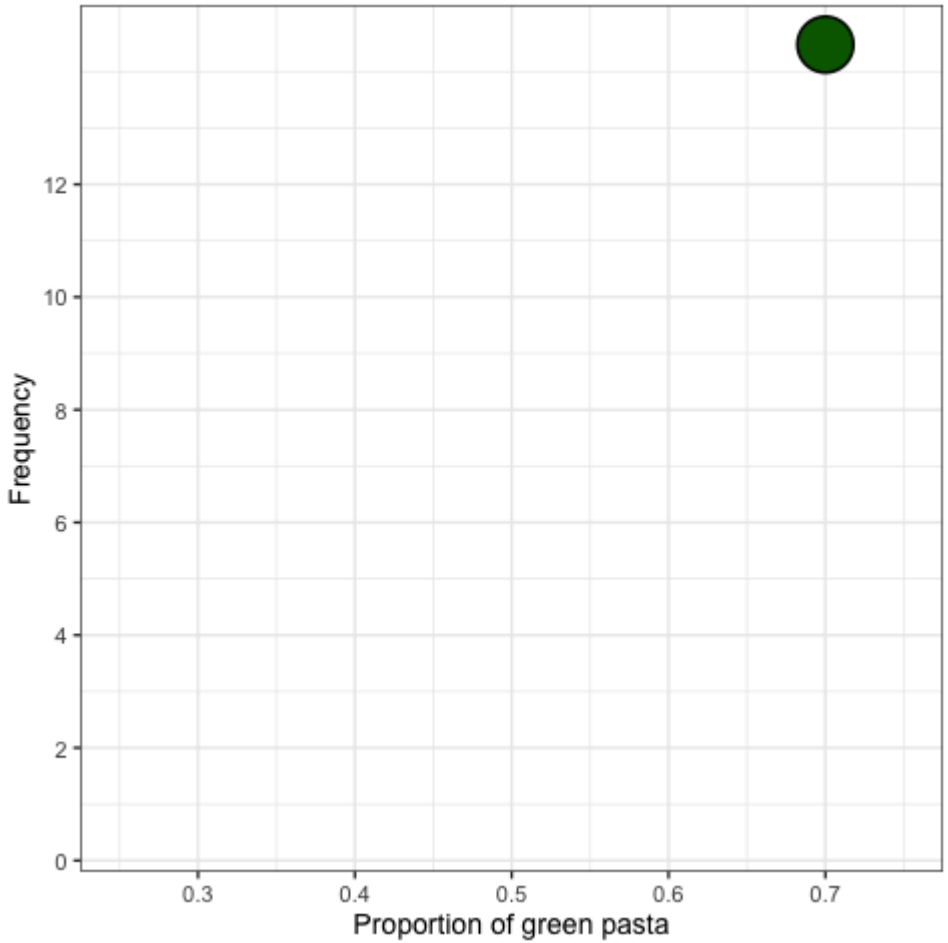
2. Create a histogram of these proportions using `ggplot2`. Use these parameters in `geom_histogram()`: `boundary = 0.325, binwidth = 0.05`. Hint: you need to

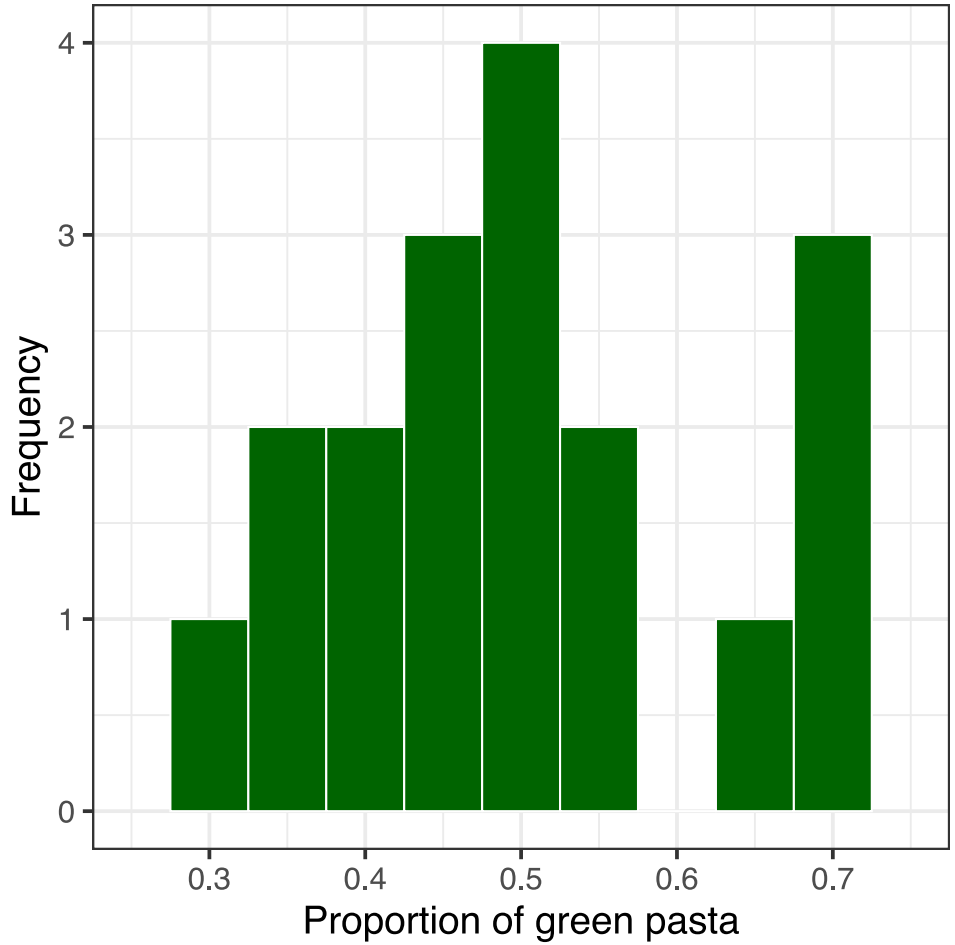
3. What do you observe?

Sample Distribution: Histogram



Sample Distribution: Histogram





What Did We Just Do?

What Did We Just Do?

- Demonstrated the statistical concept of ***sampling***.

What Did We Just Do?

- Demonstrated the statistical concept of ***sampling***.
- *Objective*: know the proportion of green pasta

What Did We Just Do?

- Demonstrated the statistical concept of ***sampling***.
- *Objective*: know the proportion of green pasta
- *Methods*:
    1. **Census**: time-consuming (and in many cases very costly);

What Did We Just Do?

- Demonstrated the statistical concept of ***sampling***.
- *Objective*: know the proportion of green pasta
- *Methods*:

  1. **Census**: time-consuming (and in many cases very costly);
  2. **Sampling**: extract a *sample* of 20 pasta from the bowl to obtain an ***estimate***.
     Out first ***estimate*** of the proportion of green pasta was 0.70, but it was actually larger than most other ***estimates***.

What Did We Just Do?

- Demonstrated the statistical concept of ***sampling***.
- *Objective*: know the proportion of green pasta
- *Methods*:

  1. **Census**: time-consuming (and in many cases very costly);

  2. **Sampling**: extract a *sample* of 20 pasta from the bowl to obtain an ***estimate***.
     Out first ***estimate*** of the proportion of green pasta was 0.70, but it was actually larger than most other ***estimates***.

- *Important*: each *sample* was drawn ***randomly*** $\rightarrow$ samples are different from each other!
  $\rightarrow$ different proportions 👉 ***sampling variation***

Taking Virtual (not Real) Samples

- We counted the exact number of green, red and yellow pasta in the bowl 🧐 `#confinement`
- All the pasta in the bowl are stored in a csv file here.

```
bowl <- read.csv("https://www.dropbox.com/s/qpjsk0rfg

head(bowl)
```

```
##   pasta_ID  color
## 1        1 yellow
## 2        2    red
## 3        3  green
## 4        4 yellow
## 5        5    red
## 6        6  green
```

Taking Virtual (not Real) Samples

- We counted the exact number of green, red and yellow pasta in the bowl 🧐 `#confinement`
- All the pasta in the bowl are stored in a csv file here.

```
bowl <- read.csv("https://www.dropbox.com/s/qpjsk0rfg

head(bowl)
```

```
##   pasta_ID  color
## 1        1 yellow
## 2        2    red
## 3        3  green
## 4        4 yellow
## 5        5    red
## 6        6  green
```

- `pasta_ID`: ball identifier
- `color`: ball color

```
nrow(bowl)
```

```
## [1] 713
```

- Instead of selecting pasta with our hands, we'll take *virtual* draws from the bowl.
- We'll use the *virtual shovel* to take a sample of 50 pasta from our virtual bowl.

Using A Virtual Shovel Once

- We will take a first sample of size 50, using the `moderndive` function `rep_sample_n`.

Using A Virtual Shovel Once

- We will take a first sample of size 50, using the `moderndive` function `rep_sample_n`.

```
#load moderndive package
library(moderndive)

virtual_shovel <- bowl %>% # notice that moderndive functions can be "pipped"
  rep_sample_n(size = 50) # take a sample of 50 balls
```

```
# display the sample's first 6 rows
head(virtual_shovel)
```

```
## # A tibble: 6 x 3
## # Groups:   replicate [1]
##   replicate pasta_ID color
##       <int>    <int> <fct>
## 1         1      176 red
## 2         1      198 green
## 3         1      705 green
## 4         1      410 green
## 5         1      215 red
## 6         1      398 green
```

- Column `replicate` tells us the ID of the sample. Here: `1`.

Using A Virtual Shovel Once

- We will take a first sample of size 50, using the `moderndive` function `rep_sample_n`.

```
#load moderndive package
library(moderndive)

virtual_shovel <- bowl %>% # notice that moderndive functions can be "pipped"
  rep_sample_n(size = 50) # take a sample of 50 balls
```

```
# display the sample's first 6 rows
head(virtual_shovel)
```

```
## # A tibble: 6 x 3
## # Groups:   replicate [1]
##   replicate pasta_ID color
##       <int>    <int> <fct>
## 1         1      176 red
## 2         1      198 green
## 3         1      705 green
## 4         1      410 green
## 5         1      215 red
## 6         1      398 green
```

```
# number of observations in sample
nrow(virtual_shovel)
```

```
## [1] 50
```

- Column `replicate` tells us the ID of the sample. Here: `1`.

Proportion of Green Pasta

```
sample_1 <- virtual_shovel %>%
  summarize(
    # number of green pasta in sample
    num_green = sum(color == "green"),
    # number of observations in sample
    sample_n = n()) %>%
  mutate(
    # proportion of green pasta in sample
    prop_green = num_green / sample_n)
sample_1
```

```
## # A tibble: 1 x 4
##   replicate num_green sample_n prop_green
##       <int>     <int>    <int>      <dbl>
## 1         1        23       50       0.46
```

1. Compute:
    - sum of green pasta in sample,
    - number of observations in sample (i.e. 50 in this case)
2. Compute proportion of green pasta

👉 0.46 are green! This is an ***estimate*** of the proportion of green pasta in the bowl. What if we try again?

What if we try many times, like, 33 times?

Using The Virtual Shovel 33 Times

33 samples (*replicates*) of size 50.

```
virtual_samples <- bowl %>%
  # get 33 samples of size 50
  rep_sample_n(size = 50, reps = 33)
virtual_samples
```

```
## # A tibble: 1,650 x 3
## # Groups:   replicate [33]
##    replicate pasta_ID color
##        <int>    <int> <fct>
##  1         1      270 green
##  2         1      377 red
##  3         1      697 red
##  4         1      189 green
##  5         1      166 yellow
##  6         1      313 yellow
##  7         1      293 yellow
##  8         1      108 green
##  9         1      154 red
## 10         1      111 red
## # … with 1,640 more rows
```

Using The Virtual Shovel 33 Times

33 samples (*replicates*) of size 50.

```
virtual_samples <- bowl %>%
  # get 33 samples of size 50
  rep_sample_n(size = 50, reps = 33)
virtual_samples
```

```
## # A tibble: 1,650 x 3
## # Groups:   replicate [33]
##    replicate pasta_ID color
##        <int>    <int> <fct>
##  1         1      270 green
##  2         1      377 red
##  3         1      697 red
##  4         1      189 green
##  5         1      166 yellow
##  6         1      313 yellow
##  7         1      293 yellow
##  8         1      108 green
##  9         1      154 red
## 10         1      111 red
## # … with 1,640 more rows
```

Compute the proportion of green pasta in each sample.

```
virtual_prop_green <- virtual_samples %>%
  group_by(replicate) %>% # calculate stat by sample
  summarize(
    num_green = sum(color == "green"),
    sample_n = n()) %>%
  mutate(prop_green = num_green / sample_n)
virtual_prop_green
```

```
## # A tibble: 33 x 4
##    replicate num_green sample_n prop_green
##        <int>     <int>    <int>      <dbl>
##  1         1        21       50      0.42
##  2         2        27       50      0.54
##  3         3        25       50      0.5
##  4         4        26       50      0.52
##  5         5        20       50      0.4
##  6         6        26       50      0.52
##  7         7        27       50      0.54
##  8         8        29       50      0.580
##  9         9        22       50      0.44
## 10        10        21       50      0.42
## # … with 23 more rows
```

(Virtual!) Sampling Variation

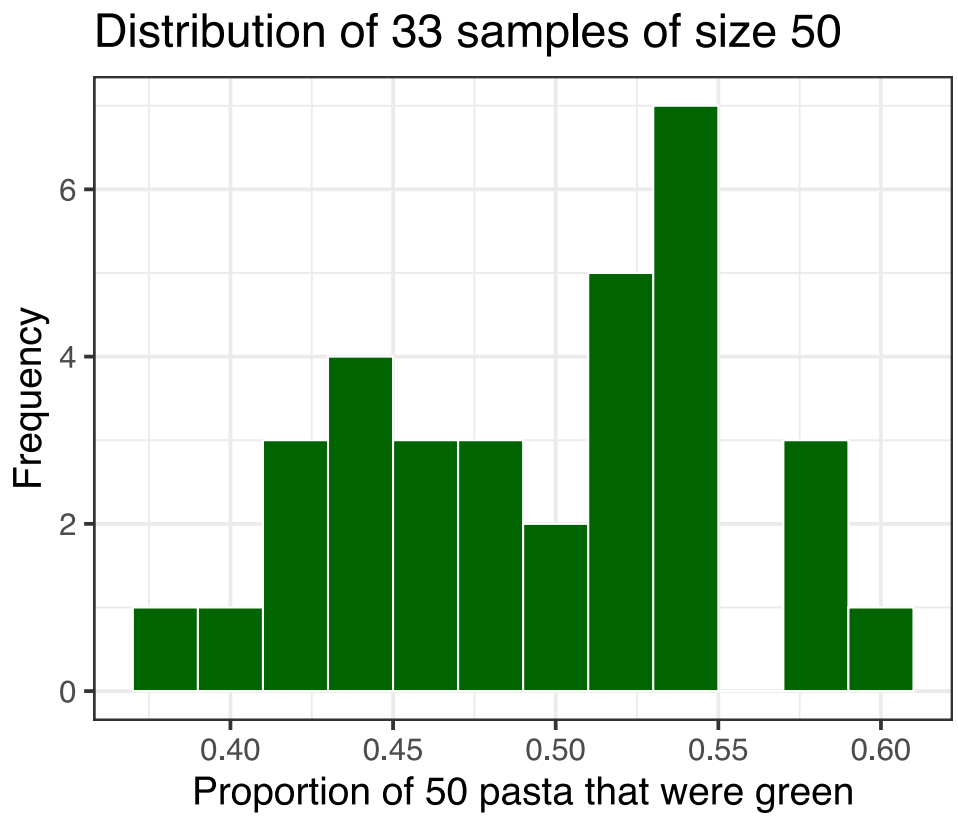
- Just as when we did it, the virtual sampler *also* creates random samples.
- The `prop_green` column in the `virtual_prop_green` data.frame differs across samples.
- And again, we can visualize the ***sampling distribution***:

```
ggplot(virtual_prop_green, aes(x = prop_green)) +
  geom_histogram(binwidth = 0.02,
                 boundary = 0.51,
                 color = "white",
                 fill = "darkgreen") +
  scale_y_continuous(breaks = seq(0, 12, by = 2)) +
  labs(x = "Proportion of 50 pasta that were green",
       y = "Frequency",
       title = "Distribution of 33 samples of size 50'
  theme_bw(base_size = 20)
```

(Virtual!) Sampling Variation

- Just as when we did it, the virtual sampler *also* creates random samples.
- The `prop_green` column in the `virtual_prop_green` data.frame differs across samples.
- And again, we can visualize the ***sampling distribution***:

```
ggplot(virtual_prop_green, aes(x = prop_green)) +
  geom_histogram(binwidth = 0.02,
                 boundary = 0.51,
                 color = "white",
                 fill = "darkgreen") +
  scale_y_continuous(breaks = seq(0, 12, by = 2)) +
  labs(x = "Proportion of 50 pasta that were green",
       y = "Frequency",
       title = "Distribution of 33 samples of size 50"
  theme_bw(base_size = 20)
```

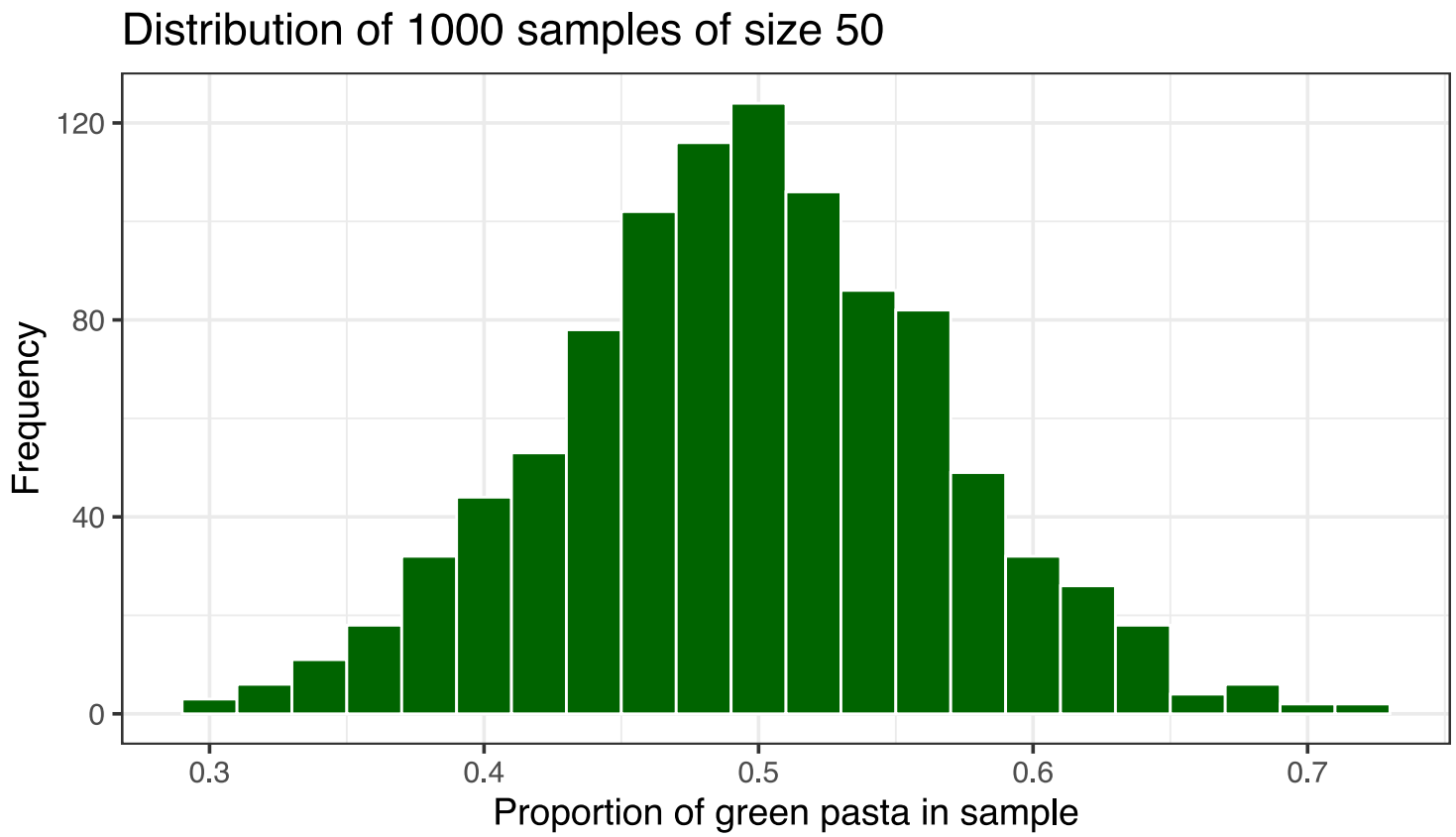


Task 2 (10 minutes)

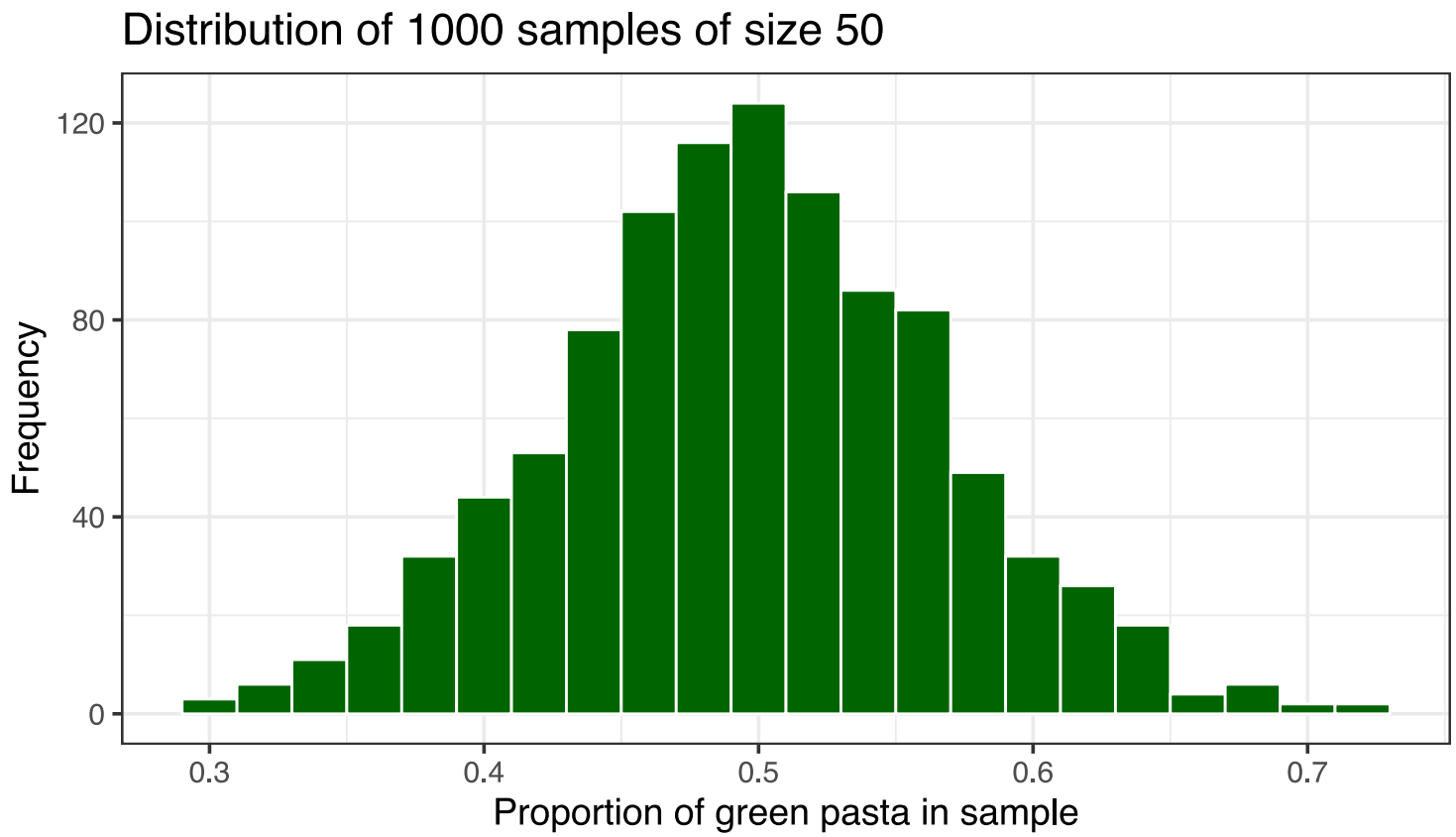
Instead of taking only 33 samples, let's take ***1000***!

1. Why do we not take 1000 samples "by hand"?
2. Load the data into an object `pasta`.
3. Obtain 1000 samples of size 50 using the `rep_sample_n()` function from the `moderndive` package.
4. Calculate the proportion of green pasta in each sample.
5. Plot a histogram of the obtained proportion of green pasta in each sample.
6. What do you observe? Which proportions occur most frequently? How does the shape of the histogram compare to when we took only 33 samples?
7. How likely is it that we sample 50 pasta of which less than 20% are green?

Sampling Distribution of 1000 Samples



Sampling Distribution of 1000 Samples



Looks remarkably close to a ***normal distribution*** $\rightarrow$ the more samples we take, the more their ***sampling distribution*** will resemble a ***normal distribution***.

Role of Sample Size

Imagine you could change the size of your samples and had the option of the following sizes: 25, 50 and 100.

If your goal is still to estimate the proportion of the bowl's pasta that are green, which shovel would you choose?

Role of Sample Size

- Let's repeat what we did previously but for different sample sizes.
- Let's take 1000 samples each for $n = 25, n = 50, n = 100$.

Role of Sample Size

- Let's repeat what we did previously but for different sample sizes.
- Let's take 1000 samples each for $n = 25, n = 50, n = 100$.
- We will use `rep_sample_n()` again.

Role of Sample Size

- Let's repeat what we did previously but for different sample sizes.
- Let's take 1000 samples each for $n = 25, n = 50, n = 100$.
- We will use `rep_sample_n()` again.

Generate all samples of different sizes:

```
# Sample size: 25
virtual_samples_25 <- bowl %>%
  rep_sample_n(size = 25, reps = 1000)

# Sample size: 50
virtual_samples_50 <- bowl %>%
  rep_sample_n(size = 50, reps = 1000)

# Sample size: 100
virtual_samples_100 <- bowl %>%
  rep_sample_n(size = 100, reps = 1000)
```

Role of Sample Size

- Let's repeat what we did previously but for different sample sizes.
- Let's take 1000 samples each for $n = 25, n = 50, n = 100$.
- We will use `rep_sample_n()` again.

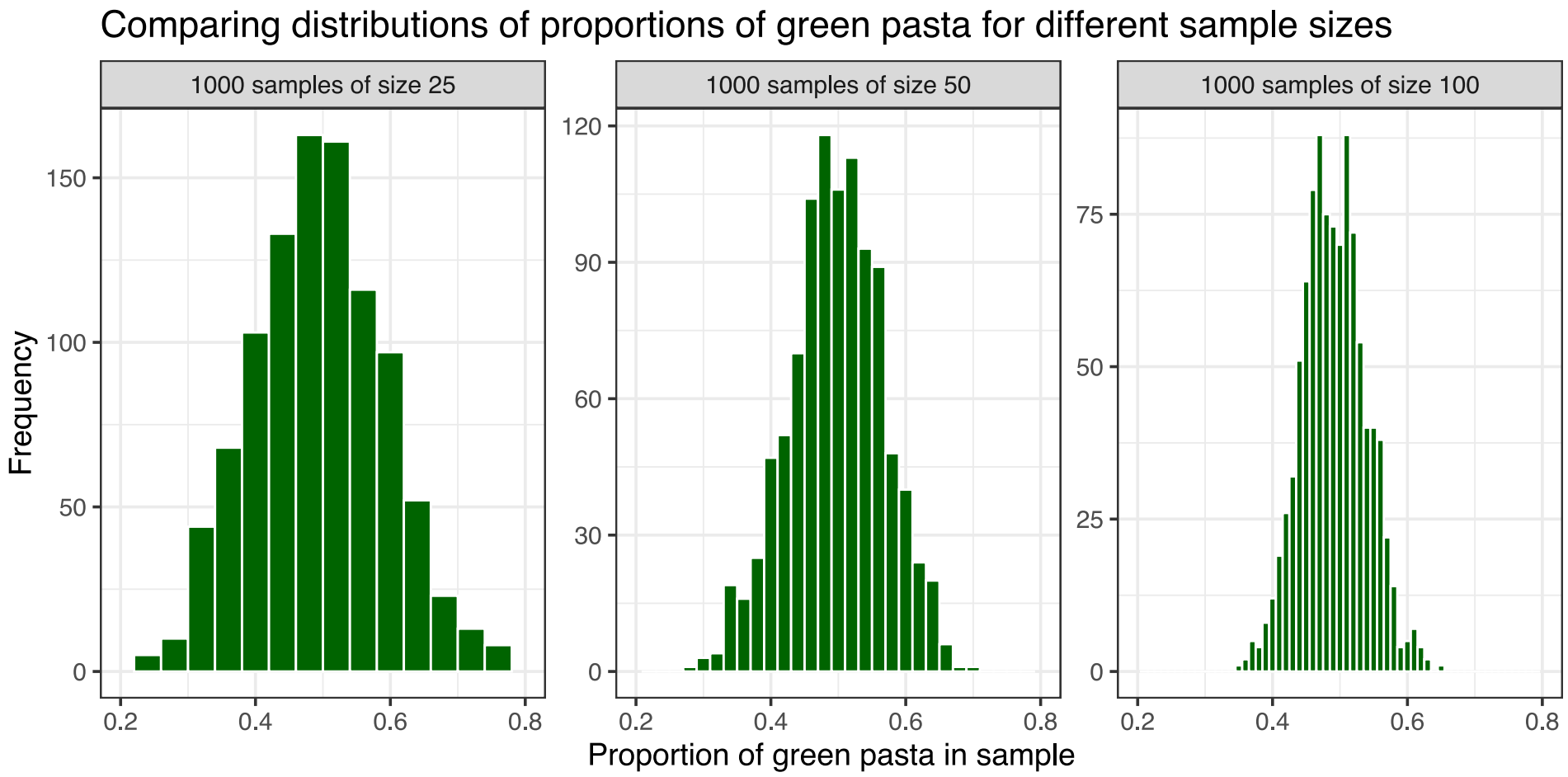
Generate all samples of different sizes:

```
# Sample size: 25
virtual_samples_25 <- bowl %>%
  rep_sample_n(size = 25, reps = 1000)

# Sample size: 50
virtual_samples_50 <- bowl %>%
  rep_sample_n(size = 50, reps = 1000)

# Sample size: 100
virtual_samples_100 <- bowl %>%
  rep_sample_n(size = 100, reps = 1000)
```

Compute proportion of green pasta:

```
# Sample size: 25
# The same code is used for the other sample sizes
virtual_prop_green_25 <- virtual_samples_25 %>%
  group_by(replicate) %>%
  summarize(
    num_green = sum(color == "green"),
    sample_n = n()) %>%
  mutate(prop_green = num_green / sample_n)
```

Role of Sample Size



Sample Size and Sampling Distributions

- The larger the sample size, the *narrower* the resulting ***sampling distribution***.

Sample Size and Sampling Distributions

- The larger the sample size, the *narrower* the resulting ***sampling distribution***.
- In other words, there are fewer differences due to ***sampling variation***.

Sample Size and Sampling Distributions

- The larger the sample size, the *narrower* the resulting ***sampling distribution***.
- In other words, there are fewer differences due to ***sampling variation***.
- Holding constant the number of replicates (i.e. 1000 in our case), ***bigger samples*** will yield *normal distributions* with ***smaller standard deviations***.

Sample Size and Sampling Distributions

- The larger the sample size, the *narrower* the resulting ***sampling distribution***.
- In other words, there are fewer differences due to ***sampling variation***.
- Holding constant the number of replicates (i.e. 1000 in our case), ***bigger samples*** will yield *normal distributions* with ***smaller standard deviations***.

Sample Size	Standard Deviation
25	0.10
50	0.07
100	0.05

Sample Size and Sampling Distributions

- The larger the sample size, the *narrower* the resulting ***sampling distribution***.
- In other words, there are fewer differences due to ***sampling variation***.
- Holding constant the number of replicates (i.e. 1000 in our case), ***bigger samples*** will yield *normal distributions* with ***smaller standard deviations***.

Sample Size	Standard Deviation
25	0.10
50	0.07
100	0.05

- Remember that the ***standard deviation*** measures the *spread* of a variable around its mean.

Sample Size and Sampling Distributions

- The larger the sample size, the *narrower* the resulting ***sampling distribution***.
- In other words, there are fewer differences due to ***sampling variation***.
- Holding constant the number of replicates (i.e. 1000 in our case), ***bigger samples*** will yield *normal distributions* with ***smaller standard deviations***.

Sample Size	Standard Deviation
25	0.10
50	0.07
100	0.05

- Remember that the ***standard deviation*** measures the *spread* of a variable around its mean.
- So as the sample size increases, our ***estimates*** of the true proportion of the bowl's green pasta get more *precise*.

Sampling Framework

- We used sampling for the purpose of ***estimation***.

Sampling Framework

- We used sampling for the purpose of ***estimation***.
- We extracted samples in order to ***estimate*** the proportion of the bowl's pasta that are green.

Sampling Framework

- We used sampling for the purpose of ***estimation***.
- We extracted samples in order to ***estimate*** the proportion of the bowl's pasta that are green.
- 2 key concepts relating to sampling for estimation:

    1. The effect of *sampling variation* on our estimates: different samples give different estimates.

    2. The effect of sample size on *sampling variation*: the bigger the size of our sample the closer our estimate should be from the true value.

Sampling Glossary 📖

Population: collection of individuals or observations we are interested in.
$N = 713$ pasta.

Population parameter: numerical summary quantity about the population that is unknown but that we want to know.
Examples: population mean (μ), proportion of green pasta (p).

Census: exhaustive enumeration or counting of all N individuals or observations in the population in order to compute the population parameter's value *exactly*.

Sampling: collecting sample(s) of size n from the population of size N.

Sampling Glossary 📖

Population: collection of individuals or observations we are interested in.
$N = 713$ pasta.

Population parameter: numerical summary quantity about the population that is unknown but that we want to know.
Examples: population mean (μ), proportion of green pasta (p).

Census: exhaustive enumeration or counting of all N individuals or observations in the population in order to compute the population parameter's value *exactly*.

Sampling: collecting sample(s) of size n from the population of size N.

- ***Point estimate*** or ***Sample statistic:*** summary statistic computed from a sample that estimates an unknown population parameter.
  Example: sample proportion of green pasta $(\hat{p})$. The "hat" on top of the p indicates that it is an *estimate* of the population proportion p.
- ***Representative sampling:*** does the sample *look like* the population?
- ***Biased sampling:*** did all pasta have an equal chance of being included in a sample?
- ***Random sampling:*** randomly sampling in an unbiased fashion.

Statistical Definitions

- We have been estimating $\hat{p}$ all along.

Statistical Definitions

- We have been estimating $\hat{p}$ all along.
- We plotted the *sampling distribution* to display the *sampling variation* of the *sample proportion* $\hat{p}$.

Statistical Definitions

- We have been estimating $\hat{p}$ all along.
- We plotted the *sampling distribution* to display the *sampling variation* of the *sample proportion* $\hat{p}$.
- We computed the *standard deviation* of the *sampling distribution* of $\hat{p}$. This standard deviation has a special name: ***standard error*** of the *point estimate* $\hat{p}$.

Statistical Definitions

- We have been estimating $\hat{p}$ all along.
- We plotted the *sampling distribution* to display the *sampling variation* of the *sample proportion* $\hat{p}$.
- We computed the *standard deviation* of the *sampling distribution* of $\hat{p}$. This standard deviation has a special name: ***standard error*** of the *point estimate* $\hat{p}$.
- Let's reproduce the summary table and labelling properly:

Sample Size (n)	**Standard Error of** $\hat{p}$
25	0.10
50	0.07
100	0.05

- Key takeaway: as the *sample size* n goes up, the "typical" error of your *point estimate* will go down, as quantified by the *standard error*.

Putting It All Together

- ***Point estimates*** from ***random samples*** provide a *good guess* of the true unknown ***population parameter***.

Putting It All Together

- ***Point estimates*** from ***random samples*** provide a *good guess* of the true unknown ***population parameter***.
- How good? Sometimes $\hat{p}$ will be far from p, sometimes close. There's ***sampling variation***.

Putting It All Together

- ***Point estimates*** from ***random samples*** provide a *good guess* of the true unknown ***population parameter***.
- How good? Sometimes $\hat{p}$ will be far from p, sometimes close. There's ***sampling variation***.
- ***On average***, our estimates will be correct. This is because of random sampling. We say that:

> $\hat{p}$ is an ***unbiased estimator*** of p, i.e. $\mathbb{E}[\hat{p}] = p$

Putting It All Together

- ***Point estimates*** from ***random samples*** provide a *good guess* of the true unknown ***population parameter***.
- How good? Sometimes $\hat{p}$ will be far from p, sometimes close. There's ***sampling variation***.
- ***On average***, our estimates will be correct. This is because of random sampling. We say that:

> $\hat{p}$ is an ***unbiased estimator*** of p, i.e. $\mathbb{E}[\hat{p}] = p$

- What is the true population proportion p of green pasta in the population of $N = 713$ pasta?

Putting It All Together

- ***Point estimates*** from ***random samples*** provide a *good guess* of the true unknown ***population parameter***.
- How good? Sometimes $\hat{p}$ will be far from p, sometimes close. There's ***sampling variation***.
- ***On average***, our estimates will be correct. This is because of random sampling. We say that:

> $\hat{p}$ is an ***unbiased estimator*** of p, i.e. $\mathbb{E}[\hat{p}] = p$

- What is the true population proportion p of green pasta in the population of $N = 713$ pasta?

```
sum(bowl$color == "green")/nrow(bowl)
```

```
## [1] 0.4936886
```

Putting It All Together

- ***Point estimates*** from ***random samples*** provide a *good guess* of the true unknown ***population parameter***.
- How good? Sometimes $\hat{p}$ will be far from p, sometimes close. There's ***sampling variation***.
- ***On average***, our estimates will be correct. This is because of random sampling. We say that:

> $\hat{p}$ is an ***unbiased estimator*** of p, i.e. $\mathbb{E}[\hat{p}] = p$

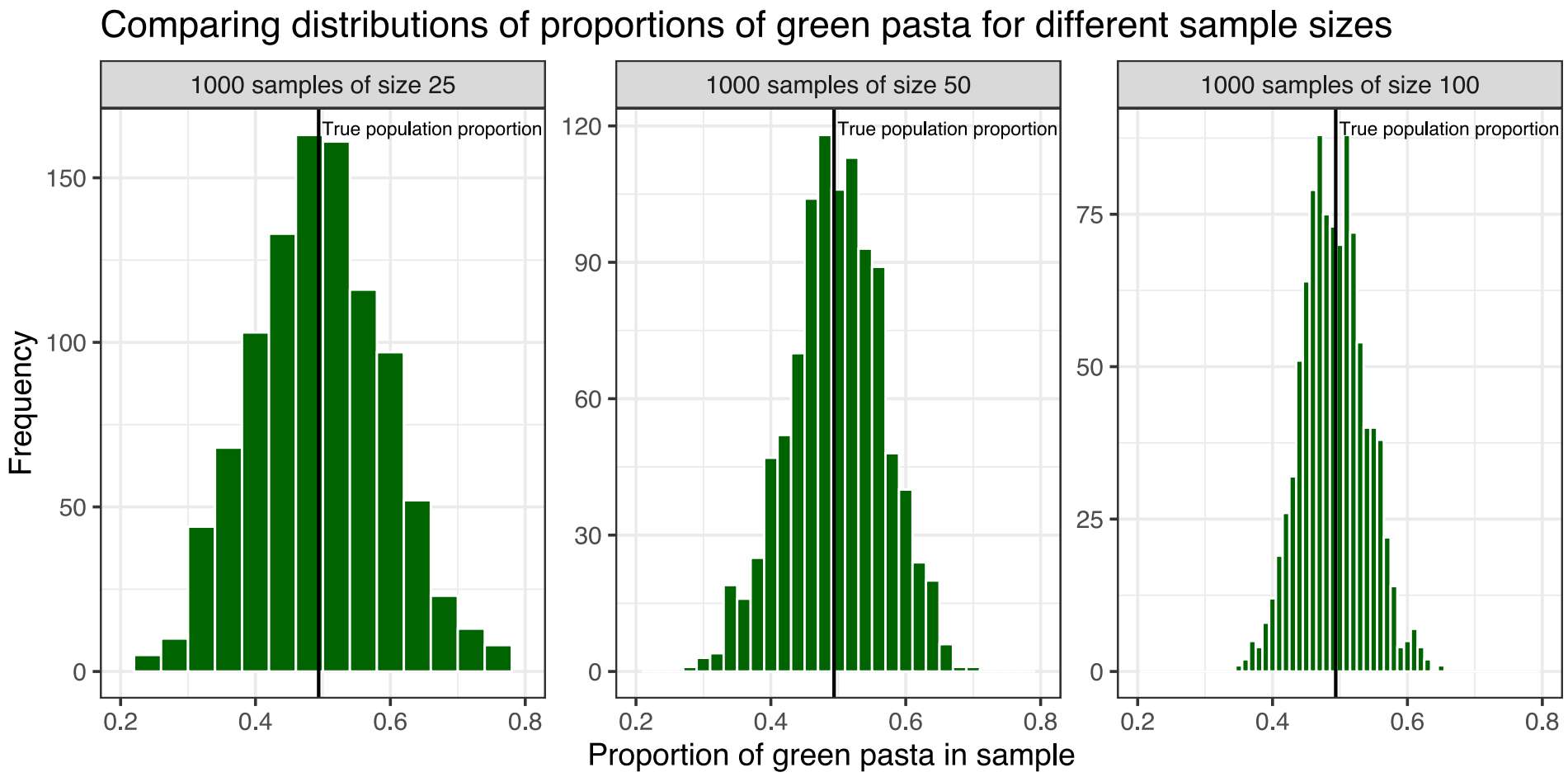
- What is the true population proportion p of green pasta in the population of $N = 713$ pasta?

```
sum(bowl$color == "green")/nrow(bowl)
```

```
## [1] 0.4936886
```

- Let's insert the ***true population proportion*** $p = 0.49$ into our previous plots!

Visualizing Unbiasedness and Sampling Variation



Some Sampling Scenarios

Scenario	Population parameter	Notation	Point estimate	Symbol(s)
1	Population proportion	p	Sample proportion	$\hat{p}$
2	Population mean	μ	Sample mean	$\overline{x}$ or $\widehat{\mu}$
3	Difference in population proportions	$p_1 - p_2$	Difference in sample proportions	$\hat{p}_1 - \hat{p}_2$
4	Difference in population means	$\mu_1 - \mu_2$	Difference in sample means	$\overline{x}_1 - \overline{x}_2$
5	Population regression slope	β_1	Fitted regression slope	b_1 or $\widehat{\beta}_1$
6	Population regression intercept	β_0	Fitted regression intercept	b_0 or $\widehat{\beta}_0$

The Central Limit Theorem (CLT)

- The fact that our sample statistics ***converge*** to a *central limit* is well known in statistics.

The Central Limit Theorem (CLT)

- The fact that our sample statistics ***converge*** to a *central limit* is well known in statistics.
- It's due to a famous result known as the ***central limit theorem***.

The Central Limit Theorem (CLT)

- The fact that our sample statistics ***converge*** to a *central limit* is well known in statistics.
- It's due to a famous result known as the ***central limit theorem***.

> *Central Limit Theorem:* regardless of how the underlying population distribution looks like, **when sample *means* are based on larger and larger sample sizes, the sampling distribution of these sample *means* becomes both more and more normally shaped and more and more narrow**.

The Central Limit Theorem (CLT)

- The fact that our sample statistics ***converge*** to a *central limit* is well known in statistics.
- It's due to a famous result known as the ***central limit theorem***.

> *Central Limit Theorem:* regardless of how the underlying population distribution looks like, **when sample *means* are based on larger and larger sample sizes, the sampling distribution of these sample *means* becomes both more and more normally shaped and more and more narrow**.

- In other words, their sampling distribution increasingly follows a ***normal distribution*** and the *variation of these sampling distributions gets smaller*, as quantified by their ***standard errors***.

Central Limit Theorem - NYTimes video

THANKS

To the amazing moderndive team!

SciencesPo
DEPARTMENT OF ECONOMICS

SEE YOU NEXT WEEK!

	florian.oswald@sciencespo.fr
	Slides
	Book
	@ScPoEcon
	@ScPoEcon