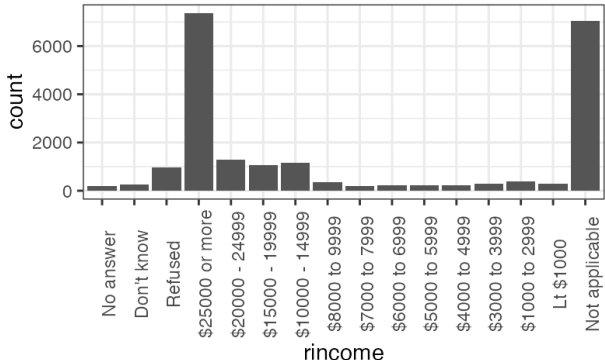
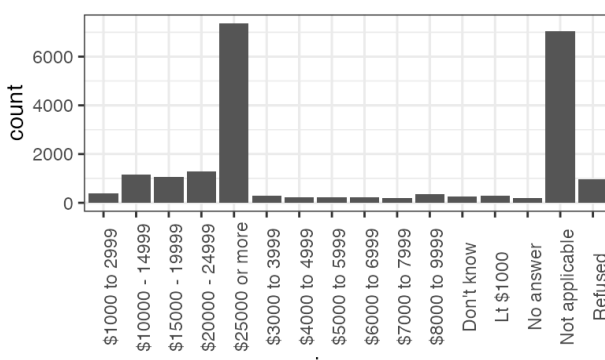


Code Sleuth

Levels

Inference

Code example	Output																																		
<pre>glimpse(gss_cat)</pre>	Rows: 21,483 Columns: 9 \$ year <int> 2000, 2000, 2000, 2000, 2000, 2000, 200... \$ marital <fct> Never married, Divorced, Widowed, Never... \$ age <int> 26, 48, 67, 39, 25, 25, 36, 44, 44, 47,... \$ race <fct> White, White, White, White, White, Whit... \$ rincome <fct> \$8000 to 9999, \$8000 to 9999, Not appli... \$ partyid <fct> "Ind,near rep", "Not str republican", "... \$ relig <fct> Protestant, Protestant, Protestant, Ort... \$ denom <fct> "Southern baptist", "Baptist-dk which",... \$ tvhours <int> 12, NA, 2, 4, 1, NA, 3, NA, 0, 3, 2, NA...																																		
<pre>levels(gss_cat\$rincome)</pre>	[1] "No answer" "Don't know" "Refused" [4] "\$25000 or more" "\$20000 - 24999" "\$15000 - 19999" [7] "\$10000 - 14999" "\$8000 to 9999" "\$7000 to 7999" [10] "\$6000 to 6999" "\$5000 to 5999" "\$4000 to 4999" [13] "\$3000 to 3999" "\$1000 to 2999" "Lt \$1000" [16] "Not applicable"																																		
<pre>ggplot(data = gss_cat, mapping = aes(x = rincome)) + geom_bar() + theme_bw() + theme(axis.text.x = element_text(angle = 90))</pre>	 <table border="1"><thead><tr><th>Income Level</th><th>Count</th></tr></thead><tbody><tr><td>No answer</td><td>~100</td></tr><tr><td>Don't know</td><td>~100</td></tr><tr><td>Refused</td><td>~1000</td></tr><tr><td>\$25000 or more</td><td>~7000</td></tr><tr><td>\$20000 - 24999</td><td>~1500</td></tr><tr><td>\$15000 - 19999</td><td>~1200</td></tr><tr><td>\$10000 - 14999</td><td>~1200</td></tr><tr><td>\$8000 to 9999</td><td>~500</td></tr><tr><td>\$7000 to 7999</td><td>~200</td></tr><tr><td>\$6000 to 6999</td><td>~200</td></tr><tr><td>\$5000 to 5999</td><td>~200</td></tr><tr><td>\$4000 to 4999</td><td>~200</td></tr><tr><td>\$3000 to 3999</td><td>~200</td></tr><tr><td>\$1000 to 2999</td><td>~200</td></tr><tr><td>Lt \$1000</td><td>~200</td></tr><tr><td>Not applicable</td><td>~7000</td></tr></tbody></table>	Income Level	Count	No answer	~100	Don't know	~100	Refused	~1000	\$25000 or more	~7000	\$20000 - 24999	~1500	\$15000 - 19999	~1200	\$10000 - 14999	~1200	\$8000 to 9999	~500	\$7000 to 7999	~200	\$6000 to 6999	~200	\$5000 to 5999	~200	\$4000 to 4999	~200	\$3000 to 3999	~200	\$1000 to 2999	~200	Lt \$1000	~200	Not applicable	~7000
Income Level	Count																																		
No answer	~100																																		
Don't know	~100																																		
Refused	~1000																																		
\$25000 or more	~7000																																		
\$20000 - 24999	~1500																																		
\$15000 - 19999	~1200																																		
\$10000 - 14999	~1200																																		
\$8000 to 9999	~500																																		
\$7000 to 7999	~200																																		
\$6000 to 6999	~200																																		
\$5000 to 5999	~200																																		
\$4000 to 4999	~200																																		
\$3000 to 3999	~200																																		
\$1000 to 2999	~200																																		
Lt \$1000	~200																																		
Not applicable	~7000																																		
<pre>gss_cat > mutate(rincome = as.character(rincome)) > ggplot(mapping = aes(x = rincome)) + geom_bar() + theme_bw() + theme(axis.text.x = element_text(angle = 90))</pre>	 <table border="1"><thead><tr><th>Income Level</th><th>Count</th></tr></thead><tbody><tr><td>\$1000 to 2999</td><td>~200</td></tr><tr><td>\$10000 - 14999</td><td>~1200</td></tr><tr><td>\$15000 - 19999</td><td>~1200</td></tr><tr><td>\$20000 - 24999</td><td>~1500</td></tr><tr><td>\$25000 or more</td><td>~7000</td></tr><tr><td>\$3000 to 3999</td><td>~200</td></tr><tr><td>\$4000 to 4999</td><td>~200</td></tr><tr><td>\$5000 to 5999</td><td>~200</td></tr><tr><td>\$6000 to 6999</td><td>~200</td></tr><tr><td>\$7000 to 7999</td><td>~200</td></tr><tr><td>\$8000 to 9999</td><td>~200</td></tr><tr><td>Don't know</td><td>~200</td></tr><tr><td>Lt \$1000</td><td>~200</td></tr><tr><td>No answer</td><td>~200</td></tr><tr><td>Not applicable</td><td>~7000</td></tr><tr><td>Refused</td><td>~1000</td></tr></tbody></table>	Income Level	Count	\$1000 to 2999	~200	\$10000 - 14999	~1200	\$15000 - 19999	~1200	\$20000 - 24999	~1500	\$25000 or more	~7000	\$3000 to 3999	~200	\$4000 to 4999	~200	\$5000 to 5999	~200	\$6000 to 6999	~200	\$7000 to 7999	~200	\$8000 to 9999	~200	Don't know	~200	Lt \$1000	~200	No answer	~200	Not applicable	~7000	Refused	~1000
Income Level	Count																																		
\$1000 to 2999	~200																																		
\$10000 - 14999	~1200																																		
\$15000 - 19999	~1200																																		
\$20000 - 24999	~1500																																		
\$25000 or more	~7000																																		
\$3000 to 3999	~200																																		
\$4000 to 4999	~200																																		
\$5000 to 5999	~200																																		
\$6000 to 6999	~200																																		
\$7000 to 7999	~200																																		
\$8000 to 9999	~200																																		
Don't know	~200																																		
Lt \$1000	~200																																		
No answer	~200																																		
Not applicable	~7000																																		
Refused	~1000																																		

Topic: Factors

What does “glimpse()” say is the type of column “rincome”?

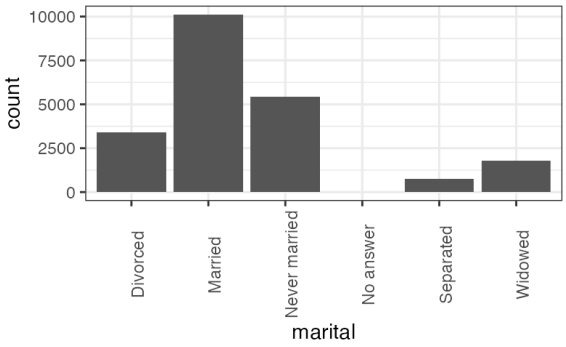
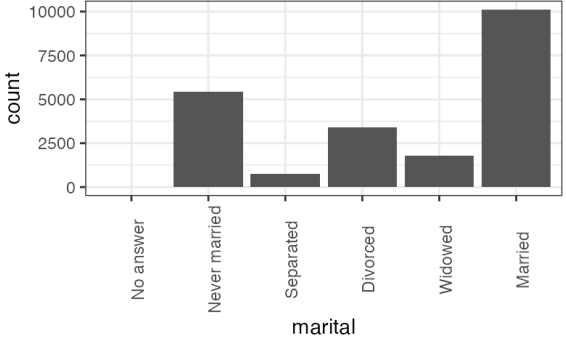
What do the levels of a factor represent?

How do levels affect the appearance of figures?

Prediction

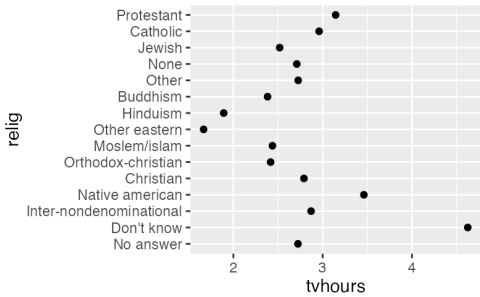
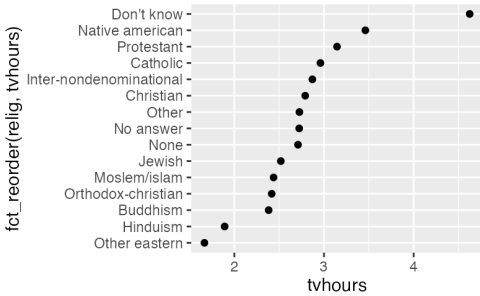
Which code example goes with which figure?

```
> levels(gss_cat$marital)
[1] "No answer"      "Never married" "Separated"      "Divorced"      "Widowed"      "Married"
```

Code example	Output														
<pre>ggplot(data = gss_cat, mapping = aes(x = marital)) + geom_bar() + theme_bw() + theme(axis.text.x = element_text(angle = 90))</pre>	 <table border="1"><thead><tr><th>marital</th><th>count</th></tr></thead><tbody><tr><td>Divorced</td><td>3500</td></tr><tr><td>Married</td><td>10000</td></tr><tr><td>Never married</td><td>5500</td></tr><tr><td>No answer</td><td>0</td></tr><tr><td>Separated</td><td>500</td></tr><tr><td>Widowed</td><td>2000</td></tr></tbody></table>	marital	count	Divorced	3500	Married	10000	Never married	5500	No answer	0	Separated	500	Widowed	2000
marital	count														
Divorced	3500														
Married	10000														
Never married	5500														
No answer	0														
Separated	500														
Widowed	2000														
<pre>gss_cat > mutate(marital = as.character(marital)) > ggplot(mapping = aes(x = marital)) + geom_bar() + theme_bw() + theme(axis.text.x = element_text(angle = 90))</pre>	 <table border="1"><thead><tr><th>marital</th><th>count</th></tr></thead><tbody><tr><td>No answer</td><td>0</td></tr><tr><td>Never married</td><td>5500</td></tr><tr><td>Separated</td><td>500</td></tr><tr><td>Divorced</td><td>3500</td></tr><tr><td>Widowed</td><td>2000</td></tr><tr><td>Married</td><td>10000</td></tr></tbody></table>	marital	count	No answer	0	Never married	5500	Separated	500	Divorced	3500	Widowed	2000	Married	10000
marital	count														
No answer	0														
Never married	5500														
Separated	500														
Divorced	3500														
Widowed	2000														
Married	10000														

Reordering

Inference

Code example	Output
<pre>relig_summary <- gss_cat > group_by(relig) > summarize(tvhours = mean(tvhours, na.rm = TRUE), n = n()) relig_summary</pre>	<pre># A tibble: 15 × 3 relig tvhours n <fct> <dbl> <int> 1 No answer 2.72 93 2 Don't know 4.62 15 3 Inter-nondenominational 2.87 109 4 Native american 3.46 23 5 Christian 2.79 689 # i 10 more rows # i Use `print(n = ...)` to see more rows</pre>
<pre>ggplot(relig_summary, aes(x = tvhours, y = relig)) + geom_point()</pre>	
<pre>ggplot(relig_summary, aes(x = tvhours, y = fct_reorder(relig, tvhours))) + geom_point()</pre>	

How did “fct_reorder()” change the appearance of the figure?

What do you think determines the new order of the factor?

Topic: Factors

Prediction

Which code example goes with which figure?

```

rincome_summary <- gss_cat |>
  group_by(rincome) |>
  summarize(
    age = mean(age, na.rm = TRUE),
    n = n()
  )
rincome_summary
# A tibble: 16 × 3
  rincome      age      n
  <fct>      <dbl> <int>
1 No answer    45.5    183
2 Don't know   45.6    267
3 Refused      47.6    975
4 $25000 or more 44.2   1363
5 $20000 - 24999 41.5   1283
# i 11 more rows
# i Use `print(n = ...) ` to see more rows

```

Code example	Output
<pre> ggplot(rincome_summary, aes(x = age, y = rincome)) + geom_point() </pre>	
<pre> ggplot(rincome_summary, aes(x = age, y = fct_reorder(rincome, age))) + geom_point() </pre>	

Modifying

Inference

Code example	Output
<pre>gss_cat > count(partyid)</pre>	<pre># A tibble: 10 × 2 partyid n <fct> <int> 1 No answer 154 2 Don't know 1 3 Other party 393 4 Strong republican 2314 5 Not str republican 3032 # i 5 more rows</pre>
<pre>gss_cat > mutate(partyid = fct_recode(partyid, "Republican, strong" = "Strong republican", "Republican, weak" = "Not str republican", "Independent, near rep" = "Ind,near rep", "Independent, near dem" = "Ind,near dem", "Democrat, weak" = "Not str democrat", "Democrat, strong" = "Strong democrat")) > count(partyid)</pre>	<pre># A tibble: 10 × 2 partyid n <fct> <int> 1 No answer 154 2 Don't know 1 3 Other party 393 4 Republican, strong 2314 5 Republican, weak 3032 # i 5 more rows</pre>
<pre>gss_cat > mutate(partyid = fct_collapse(partyid, "Republican" = c("Strong republican", "Not str republican"), "Independent" = c("Ind,near rep", "Ind,near dem"), "Democrat" = c("Not str democrat", "Strong democrat"), "Other" = c("No answer", "Don't know", "Other party"))) > count(partyid)</pre>	<pre># A tibble: 4 × 2 partyid n <fct> <int> 1 Other 548 2 Republican 5346 3 Independent 8409 4 Democrat 7180</pre>

What does “fct_recode()” do?

What does “fct_collapse()” do?

Prediction

What will be the factor levels in the output of this code?

```
gss_cat |> count(marital)

gss_cat |>
  mutate(
    marital = fct_collapse(
      marital,
      "Currently married" = "Married",
      "Previously married" = c("Separated", "Divorced", "Widowed"),
      "Never married" = "Never married"
    )
  ) |>
  count(marital)
```