CORE S119 FA24 Prog 3

Grusha Prasad

2024-09-17

```
titanic = read.csv("titanic.csv")
```

Tidyverse

In the tidyverse package, you are taking a dataframe and performing different computations on it. You use the "%>%" notation to "pipe" the dataframe through different processes. At the end of the entire piping process a new dataframe is generated. You need to either assign the new dataframe to a new variable, or re-assign it to the old variable.

Modifying dataframes: mutate and rename

Earlier, we used `$` to add columns one at a time. In tidyverse, we can use mutate function to add columns. For example, the following code adds a column called `age_in_decade` using the original Age column. And then adds another column called `decade_floor` which rounds down `age_in_decade` to the nearest whole number. Note how within `mutate()` you do not need to use the `$` sign.

```
titanic_new = titanic %>%
  mutate(age_in_decade = Age/10,
         decade_floor = floor(age_in_decade))
```

Written Q1 If you look at the titanic dataframe, it does not have the columns `age_in_decade` or `decade_floor`. Why?

Answer

Written Q2 The following code, which is very similar to the code above results in an error. Why?

Answer

```
titanic_new = titanic %>%
  mutate(decade_floor = floor(age_in_decade),
         age_in_decade = Age/10)
## Error in `mutate()`:
## i In argument: `decade_floor = floor(age_in_decade)`.
## Caused by error:
## ! object 'age_in_decade' not found
```

Coding Q1 Write code that will add `age_in_decade` or `decade_floor` to the original `titanic` dataset. Make sure to run your code!

```
## WRITE YOUR CODE BELOW THIS LINE
```

Coding Q2 Write code that will add a column called `age_sqrt` to the original `titanic` dataset, which will hold hold the square root of the original age. *Hint: Google how to compute square root in R*

```
## WRITE YOUR CODE BELOW THIS LINE
```

Coding Q3 Write code that will add a column called `age_category` to the original `titanic` dataset, which will hold the string value "young" if age is less than 25, "middle age" if age is between 25 and 55, and "old" if age is greater than 55.

```
## WRITE YOUR CODE BELOW THIS LINE
```

You can also rename existing columns using the `rename` function. For example, renames the `Age` column to be `age`.

```
titanic = titanic %>%
  rename(age = Age)

#
#   mutate(age_in_decade = Age/10,
#          decade_floor = floor(age_in_decade)) %
```

Written Q2 The following code, tries to rename the column Fare to fare. Why does it throw an error?

Answer

```
titanic = titanic %>%
  rename(Fare = fare)
## Error in `rename()`:
## ! Can't rename columns that don't exist.
## x Column `fare` doesn't exist.
```

Written Q3 Describe what the code below does.

Answer

```
titanic_new = read.csv("titanic.csv") %>%
  rename(age = Age) %>%
  mutate(age_in_decade = age/10,
         age_decade_floor = floor(age_in_decade),
         old = ifelse(age_decade_floor >= 6, TRUE, FALSE))
```

Select and Filter

You can use the select function to pick only specific columns. For example, the code below keeps only the `Survived`, `age` and `Sex` columns and assigns it to a dataframe called `titanic2`.

```
titanic2 <- titanic %>%
  select(Survived, age, Sex)
```

You can also choose to include all columns except a few. The code below excludes just the columns `X` and `Cabin` from `titanic`

```
titanic3 <- titanic %>%
  select(-X, -Cabin)
```

Written Q4 Explain why the following code results in an error

Answer

```
titanic4 = titanic3 %>%
  select(Fare, Cabin, Survived, Embarked)
## Error in `select()`:
## ! Can't subset columns that don't exist.
## x Column `Cabin` doesn't exist.
```

Coding Q4 Write code that takes the `titanic` dataset, selects all columns except Parch, SibSp and Cabin. Then renames then columns so that they are all starting with a lower case. Assign the resulting dataset to a variable called `titanic5`. Print the column names of `titanic5` to make sure that your code worked correctly.

```
## WRITE YOUR CODE BELOW THIS LINE
```

You can also get subsets of rows using `filter`. For example the following code creates a subset of the women who survived and assigns it to a new variable.

```
women_survived = titanic %>%
  filter(Sex == 'female',
         Survived == 1)
```

Coding Q5 Write code that gets the subset of the data of men above 50 who survived and assign it to a new variable called `old_men_survived`

```
## WRITE YOUR CODE BELOW THIS LINE
```

Group by and summarize

You can compute different summaries for different groups using group_by summarize. This will result in a dataframe with as many rows as there are groups. The following code computes the mean and standard deviation of age, and the total number and proportion of people who survived separately for men and women.

```
summ1 = titanic %>%
  group_by(Sex) %>%
  summarise(mean_age = mean(age, na.rm=TRUE), #ignores NA values
            sd_age = sd(age, na.rm=TRUE),
            total_survied = sum(Survived),
            n = n(), # number in each group
            prop_survived = sum(Survived)/n, .groups="drop")
```

Coding Q6 Write code to compute the mean fare and age for the people who survived vs. the people who did not. Assign the resulting dataframe to a variable called `summ2`.

```
## WRITE YOUR CODE BELOW THIS LINE
```

You can also group by more than one group. For example, the following code finds the mean and standard error of fare for people who survived vs. did not, in each of the PClass. Standard error is the standard deviation divided by the square root of n.

```
summ3 = titanic %>%
  group_by(Survived, Pclass) %>%
```

```
  summarise(mean_fare = mean(Fare, na.rm=TRUE), #ignores NA values
            se_fare = sd(Fare, na.rm=TRUE)/sqrt(n()), .groups="drop")
```

Written Q5 What do you notice about the mean fares in the six categories?

Answer

Coding Q7 Write code to compute the mean age and standard error for the men and women who survived vs. the men and women who did not. Assign the resulting dataframe to a variable called `summ4`.

```
## WRITE YOUR CODE BELOW THIS LINE
```

Written Q6 What do you notice about the mean ages in the four categories?

Answer

Putting it together (Homework)

```
# Read about dataset here: https://vincentarelbundock.github.io/Rdatasets/doc/AER/HousePrices.html
houses = read.csv("HousePrices.csv")
```

1. Create a column in the `houses` dataframe called `size_category` which holds the value "small" if it is less than 3000, "medium" if it is between 3000 and 7000 and "large" if it is greater than 7000.
2. There are six columns which have values "yes" and "no". For each of these columns, change "yes" to 1 and "no" to 0. Note, by the end of this step you should not have more columns than before.
3. Consider the houses without a full basement and without a driveway. Compute the mean lot size and the mean number of bedrooms for these houses for houses in preferred and unpreferred locations.
4. Consider the houses with three or fewer bedrooms that don't have a garage. For these houses, create a summary dataframe that holds the mean price, the median number of bathrooms and the proportion of houses with air conditioning for each combination of house category and number of stories.
5. Come up with a question you can answer from the dataset. Write the code to answer the question.