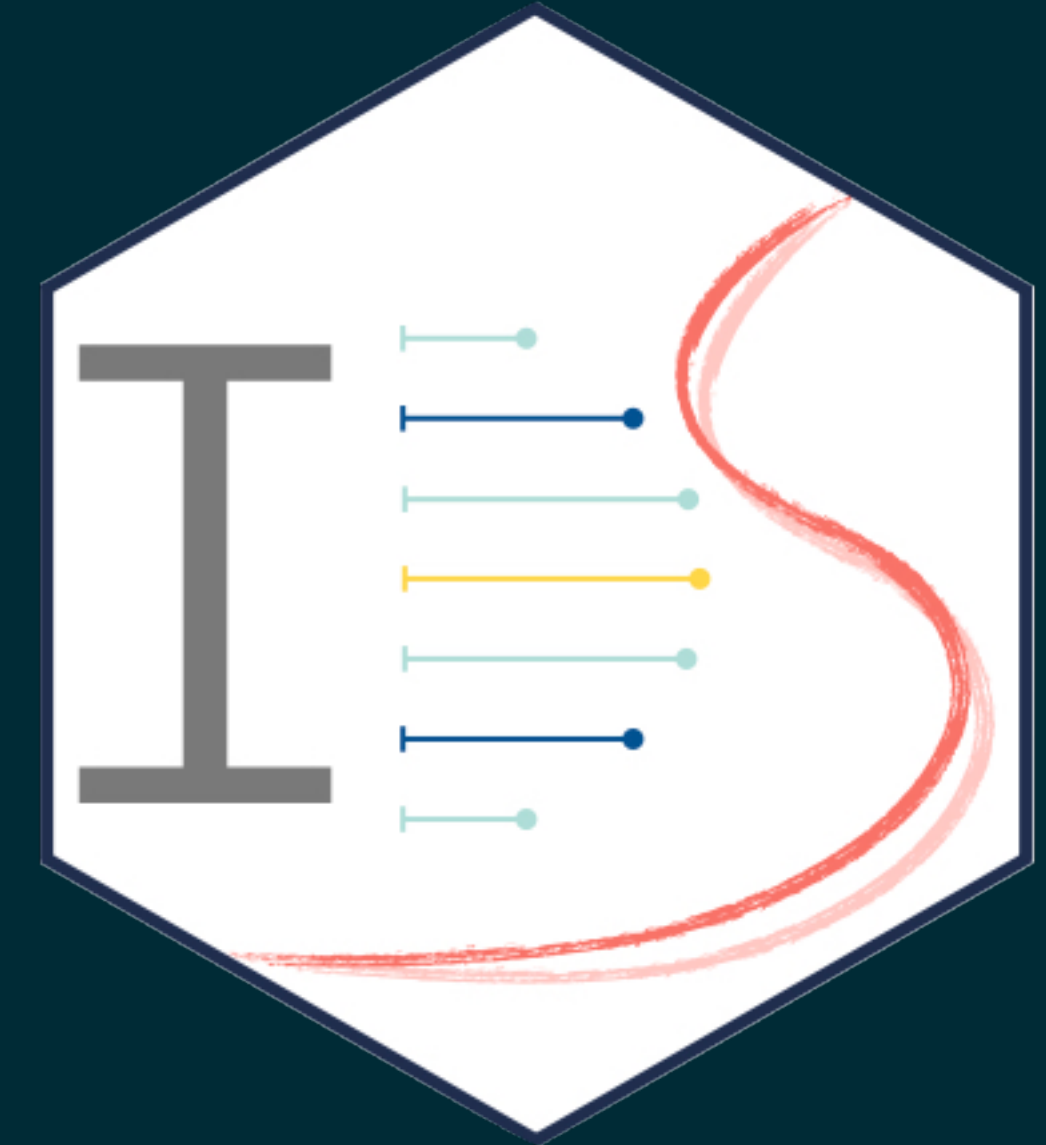





















keeping up with

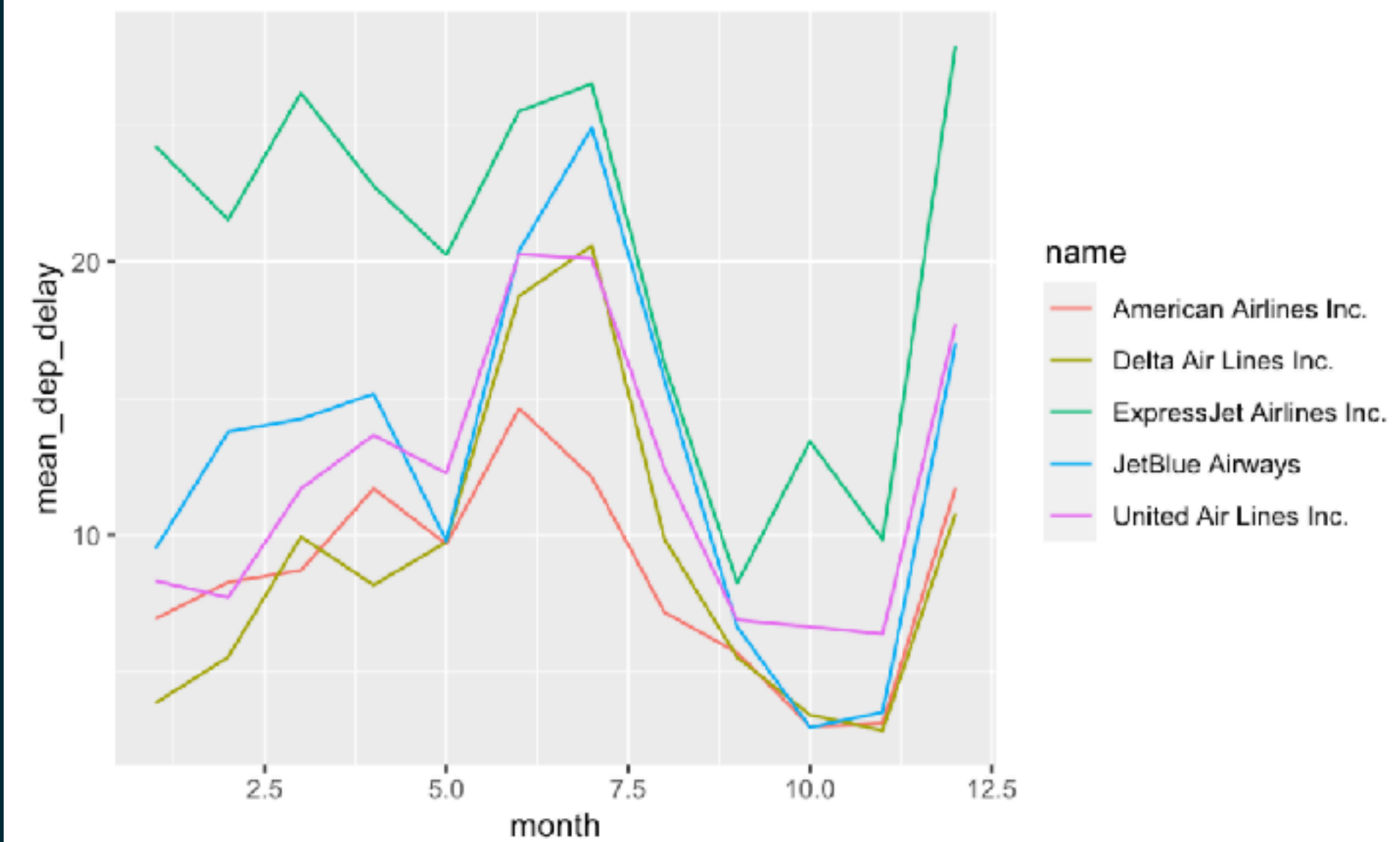


week 5

Week 4 - Data importing & recoding

No.	Title	YouTube	MediaHopper	Slides	Length
01	Keeping up with IDS: Week 4				19:58
02	Data types				30:29
03	Data classes				19:28
04	Importing data				32:08
05	Recoding data				20:22
06	AE: Hotels + Data types				17:00
07	AE: Nobels + Sales + Data import				19:28

0Q 03 - Question 7

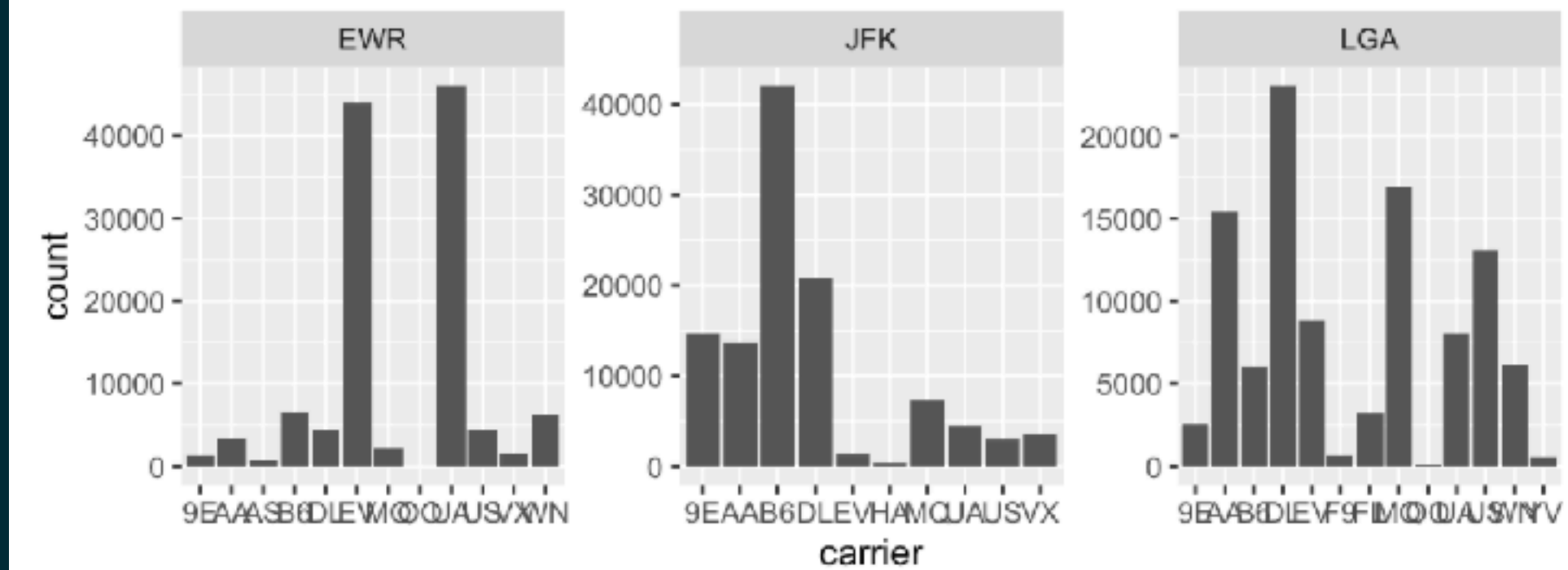


Which of the following would *necessarily* improve this visualization? Check all that apply.

- Replace x-axis labels with month names. ✓
- Flip the coordinates so months are on the y-axis and mean departure delays are on the x-axis. ✗
- Change the colors of the lines. ✗
- Indicate units of mean departure time. ✓
- Change the background color of the plot. ✗
- Place airline names at the end of lines on the plot instead of in the legend. ✓
- Add a title. ✓
- Fix the axis labels to remove underscores and use proper capitalization. ✓
- Use bars instead of lines to represent the data. ✗

Correct!

OQ 03 - Question 8



Which of the following would *necessarily* improve this visualization? Check all that apply.

- Use airline names instead of carrier codes. ✓
- Change the background color of the plot. ✗
- Use a different color for each bar. ✗
- Add a title. ✓
- Place carrier names at an angle so they don't overlap. ✗
- Use a histogram to represent the data. ✗
- Fix the scales of the axes so that the carriers are in the same order for each and the counts are on the same scale. ✓
- Flip the coordinates so carriers are on the y-axis and counts are on the x-axis. ✓

Correct!

join functions

Week 03 videos

No.	Title	YouTube	MediaHopper	Slides	Length
01	Keeping up with IDEs				
02	Working with a single data frame				32:15
05	Working with multiple data frames				17:26
06	Tidying data				
07	AE: Hotel				

Two-table verbs

Source: vignettes/two-table.Rmd

It's rare that a data analysis involves only a single table of data. In practice, you'll normally have many tables that contribute to an analysis, and you need flexible tools to combine them. In dplyr, there are three families of verbs that work with two tables at a time:

- Mutating joins, which add new variables to one table from matching rows in another.
- Filtering joins, which filter observations from one table based on whether or not they match an observation in the other table.
- Set operations, which combine the observations in the data sets as if they were set elements.

(This discussion assumes that you have tidy data, where the rows are observations and the columns are variables. If you're not familiar with that framework, I'd recommend reading up on it first.)

All two-table verbs work similarly. The first two arguments are `x` and `y`, and provide the tables to combine. The output is always a new table with the same type as `x`.

Contents

- Mutating joins
- Filtering joins
- Set operations
- Multiple-table verbs

Week 5 - Communicating data science results effectively

- 👁️ Peer evaluations
- 👁️ Project proposals