ECO395M/exercises01.md at master · jgscott/ECO395M

ECO 395M: Exercises 1

1) Data visualization: gas prices

This problem is about making simple plots and telling stories using those plots.

For this exercise, you'll need to download the `GasPrices.csv` data set from the class website, which contains data from 101 gas stations in the Austin area collected in 2016. There are lots of variables in this data set, but for our purposes here, the important ones are as follows:

- ID: Order in which gas stations were visited
- Name: Name of gas station
- Price: Price of regular unleaded gasoline, gathered on Sunday, April 3rd, 2016
- Highway: Is the gas station accessible from either a highway or a highway access road?
- Stoplight: Is there a stoplight in front of the gas station?
- Competitors: Are there any other gas stations in sight?
- Zipcode: Zip code in which gas station is located
- Income: Median Household Income of the ZIP code where the gas station is located based on 2014 data from the U.S. Census Bureau
- Brand: ExxonMobil, ChevronTexaco, Shell, or Other.

The theories

People have a lot of pet theories about what explains the variation in prices between gas stations. Here are several such theories below. **Which of these theories seem plausible, and which are unsupported by data? Take each theory one by one and assess the evidence for or against the theory using the suggested plot in parentheses.**

A) Gas stations charge more if they lack direct competition in sight (boxplot).

B) The richer the area, the higher the gas price (scatter plot).

C) Shell charges more than other brands (bar plot).

D) Gas stations at stoplights charge more (faceted histogram).

E) Gas stations with direct highway access charge more (your choice of plot).

Include an annotation below each plot. (This is especially easy to do in RMarkdown: just write a paragraph below the plot.) Your annotation of each figure should include two main elements:

- Claim: a statement of the theory itself.
- Conclusion: how the figure informs your conclusion about whether the theory is supported or unsupported by the data.

Note: you don't have to run any statistical tests, fit models, or compute confidence intervals here. Just make pictures and describe what you're seeing.

2) Data visualization: a bike share network

This problem continues the review of data exploration and visualization using `ggplot2` and the `tidyverse` packages, but now asking you to bring in multiple variables into a single plot.

Bike-sharing systems are a new generation of traditional bike rentals where the whole process from rental to return is automatic. There are thousands of municipal bike-sharing systems around the world (e.g. Citi bikes in NYC or "Boris bikes" in London), and they have attracted a great deal of interest because of their important role in traffic, environmental, and health issues---especially in the wake of the COVID-19 pandemic, when ridership levels on public-transit systems have plummeted.

These bike-sharing systems also generate a tremendous amount of data, with time of travel, departure, and arrival position recorded for every trip. This feature turns bike sharing system into a virtual sensor network that can be used for sensing mobility patterns across a city.

Bike-sharing rental demand is highly correlated to environmental and seasonal variables like weather conditions, day of week, time of year, hour of the day, and so on. In this problem, you'll look at some of these demand-driving factors using the `bikeshare.csv` data. This data set contains a two-year historical log (2011 and 2012) from the Capital Bikeshare system in Washington D.C. The raw data is publicly available at http://capitalbikeshare.com/system-data. These data have been aggregated on an hourly and daily basis and then merged with weather and seasonal data.

The variables in this data set are as follows:

- instant: unique record identifier for each row
- dteday: date
- season: season (1:spring, 2:summer, 3:fall, 4:winter)
- yr: year (0: 2011, 1:2012)
- mnth: month (1 to 12)
- hr: hour (0 to 23)
- holiday: whether the day is holiday or not
- weekday: day of the week (1 = Sunday)
- workingday: if day is neither weekend nor holiday is 1, otherwise is 0.
- weathersit: a weather situation code with the following values
    - 1: Clear, Few clouds, Partly cloudy, Partly cloudy
    - 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
    - 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
    - 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- temp: Normalized temperature in Celsius. The actual values are divided by 41 (max)
- total: count of total bike rentals that hour, including both casual and registered users

Your task in this problem is to prepare three figures.

- Plot A: a line graph showing average bike rentals (`total`) versus hour of the day (`hr`).

- Plot B: a faceted line graph showing average bike rentals versus hour of the day, faceted according to whether it is a working day (`workingday`).

- Plot C: a faceted bar plot showing average ridership **during the 8 AM hour** by weather situation code (`weathersit`), faceted according to whether it is a working day or not. Note: remember you can focus on a specific subset of rows of a data set using `filter` , e.g.

```
bikeshare %>%
  filter(hr==8)
```

Your write-up should include each plot, together with an informative annotation/caption below the plot. Your caption should clearly explain the plot itself (e.g. what the axes are and what the panels show). The caption should also contain a one-sentence *take-home lesson* of what we have learned about ridership patterns from the plot.

3) Data visualization: flights at ABIA

This problem continues the themes of data exploration and data visualization using `ggplot2` and the `tidyverse`, but is entirely open-ended.

Consider the data in ABIA.csv, which contains information on every commercial flight in 2008 that either departed from or landed at Austin-Bergstrom Interational Airport. The variable codebook is as follows:

- Year all 2008
- Month 1-12
- DayofMonth 1-31
- DayOfWeek 1 (Monday) - 7 (Sunday)
- DepTime actual departure time (local, hhmm)
- CRSDepTime scheduled departure time (local, hhmm)
- ArrTime actual arrival time (local, hhmm)
- CRSArrTime scheduled arrival time (local, hhmm)
- UniqueCarrier unique carrier code
- FlightNum flight number
- TailNum plane tail number
- ActualElapsedTime in minutes
- CRSElapsedTime in minutes
- AirTime in minutes
- ArrDelay arrival delay, in minutes
- DepDelay departure delay, in minutes
- Origin origin IATA airport code
- Dest destination IATA airport code
- Distance in miles
- TaxiIn taxi in time, in minutes
- TaxiOut taxi out time in minutes
- Cancelled was the flight cancelled?
- CancellationCode reason for cancellation (A = carrier, B = weather, C = NAS, D = security)
- Diverted 1 = yes, 0 = no
- CarrierDelay in minutes
- WeatherDelay in minutes
- NASDelay in minutes
- SecurityDelay in minutes
- LateAircraftDelay in minutes

Your task is to create a figure, or set of related figures, that tell an interesting story about flights into and out of Austin. You should annotate your figure(s), of course, but strive to make them as easy to understand as possible at a quick glance. (A single figure shouldn't need many, many paragraphs to convey its meaning.) For example, you might consider one of the following questions:

- What is the best time of day to fly to minimize delays, and does this change by airline?
- What is the best time of year to fly to minimize delays, and does this change by destination? (You'd probably want to focus on a handful of popular destinations.)
- How do patterns of flights to different destinations or parts of the country change over the course of the year?
- What are the bad airports to fly to and does this change by time of year or day?

But anything interesting will fly :-). If you want to try your hand at mapping or looking at geography, you can cross-reference the airport codes here: https://github.com/datasets/airport-codes. Combine this with a mapping package like ggmap, and you should have lots of possibilities!

4) K-nearest neighbors

The data in sclass.csv contains data on over 29,000 Mercedes S Class vehicles---essentially every such car in this class that was advertised on the secondary automobile market during 2014. For websites like Cars.com or Truecar that aim to provide market-based pricing information to consumers, the Mercedes S class is a notoriously difficult case. There is a huge range of sub-models that are all labeled "S Class,"" from large luxury sedans to high-performance sports cars; one sub-category of S class even serves as the official pace car in Formula 1 Races. Moreover, individual submodels involve cars with many different features. This extreme diversity---unusual for a single model of car---makes it difficult to provide accurate pricing predictions to consumers.

We'll revisit this data set later in the semester when we've got a larger toolkit for building predictive models. For now, let's focus on three variables in particular:

- trim: categorical variable for car's trim level, e.g. 350, 63 AMG, etc. The trim is like a sub-model designation.
- mileage: mileage on the car
- price: the sales price in dollars of the car

Your goal is to use K-nearest neighbors to build a predictive model for price, given mileage, separately for each of two trim levels: 350 and 65 AMG. (There are lots of other trim levels that you'll be ignoring for this question.) That is, you'll be treating the 350's and the 65 AMG's as two separate data sets. (Recall the `filter` command.)

For each of these two trim levels:

1. Split the data into a training and a testing set.
2. Run K-nearest-neighbors, for many different values of K, starting at K=2 and going as high as you need to. For each value of K, fit the model to the training set and make predictions on your test set.
3. Calculate the out-of-sample root mean-squared error (RMSE) for each value of K.

For each trim, make a plot of RMSE versus K, so that we can see where it bottoms out. Then for the optimal value of K, show a plot of the fitted model, i.e. predictions vs. x. (Again, separately for each of the two trim levels.)

Which trim yields a larger optimal value of K? Why do you think this is?

▶ Details