# Fuzzy – Interview Challenge – Data Analyst

## Introduction

There are four data sets provided with these questions. For the most part, the column names in each file are self-explanatory, but for "InterviewData_Activity.csv", here is a quick description of the variables:

- **userid** (our unique identifier of a member in the database)
- **date** the member last placed an order
- **age** (an integer value)
- **gender** ("M" or "F")
- **metropolitan_area** (sample metro areas)
- **device_type** used (Desktop, Tablet, or Mobile)
- **active** (the response variable and a binary outcome, "0" meaning they were not an active member and "1" meaning they were active at the time this illustrative data was generated)

For the following, please use either R or Python (if you know both, please pick the one you're more comfortable with), and you are free to use any packages you want to help answer these questions.

## Required Questions

*Questions 1, 2, and 3 deal with "InterviewData_Cost.csv" and "InterviewData_Rev.csv".*

(1) Join these two data sets by "date" and "source_id", returning all rows from both regardless of whether there is a match between the two data sets.

(2) Join these two data sets by "date" and "source_id", returning only the rows from the "Cost" file that have no corresponding date in the "Revenue" file.

(3) Using your result from #1:
   a. What are the Top 4 sources ("source_id" values) in terms of total revenue generation across this data set?
   b. How would you visualize the monthly revenue for those Top 4 sources?

*Questions 4 and 5 deal with "InterviewData_Activity.csv".*

(4) In either R or Python, read the data into an object called `activity_data`, and then build a vanilla logistic regression model to predict activity. That is, predict activity using just age, gender, metropolitan area, and device type. Depending on your choice of language and package to build the regression, you may need to convert some of the categorical variables such as gender. Given this vanilla logistic regression model, assess the prediction accuracy.

(5) Split the data into training and test samples by separating the first 4000 rows into training data, and build a model over the new training data.

Assess the training data model's accuracy on the test data. How and why does the accuracy differ compared to (4)?

*Question 6 deals with "InterviewData_Parsing.csv".*

(6) This data comes from a subset of userdata JSON blobs stored in our database. Parse out the values (stored in the "data_to_parse" column) into four separate columns. So for example, the four additional columns for the first entry would have values of "N", "U", "A7", and "W". You can use any R functions/ packages you want for this.

## Additional Questions – Pick One

**Pick only one** of these questions and answer it in less than 400 words.

A) Within our web and mobile apps, users sign up for vet care through a funnel with various questions before their account is created. Let's say that we decide to test two different series of questions (two funnels). The Product team asks for your help in setting up the test and calling the results. How would you help them: (i) figure out how long we should run this test; (ii) decide what metric to measure; (iii) and then evaluate the test?

B) One of the ways we attract new members is through digital marketing campaigns (e.g., on Facebook). Assume that we know a little bit about potential users who see an ad for Fuzzy on Facebook – things like name and general metropolitan area, and then can measure the impressions on the ad, clicks to our landing page, and then conversions on our landing page. Our goal then is to drive more conversions on the landing page. What are some ways you might look at the already collected data (or some ways to enrich the existing data set) to try and make recommendations to the Marketing team for how to optimize their campaigns?

C) When we introduce new products or features, we generally prefer to implement them as a test at first – sometimes though, we don't have that option. Imagine that on the operations side, which deals primarily with veterinarians, we decide to revamp the way in which potential chats(from a member) are offered to a vet (they can either ignore the notification, decline it, or accept it). We've got 60 days of data under the old system and 60 days after the revamp was implemented, we're hoping to figure out whether it led to an increase in offer acceptance rate (proportion of accepts out of all the response options). How might you go about trying to quantify the change due to the revamp?