initial plan

Jinghan Cui

11/14/2017

Topic

Will the star of given business related to the reviewers' review count and average star?

To investigate which factor is most important in rating, I will use the data set of "category", "review" and "user" from the yelp dataset. The "business" data have 156639 rows and contains 12 variables. The "category" data have 590290 rows and 2 variables. The "user" data have 1183352 rows and 20 variables.

Clean Data

To clean the data, I will merge category and business to one data frame by business id and match the review with business by business id as well. Then merge user with review by user id. In cleaned data set, there will be variables of "category", "stars", "state", "city", "review_count", "user_id", "user_review_count", "user_average_star".

EDA

To initially explore the data, my plan is to do a plot of user_average_star with user_review_count, a plot of state and stars. Thus, we can check whether there are within-group relations and is the state significant enough to be a group indicator.

Model

A linear model will be used first, to check whether our hypothesis can be captured by simple linear model. If yes, fit a random effect model see if predictions are improved. If not, fit a multilevel model with category and location of the business as group-level predictor and user as individual-level predictor.

```
mydb = dbConnect(MySQL(), user='mssp', password='mssp2017', dbname='yelp_db', host='45.63.90.29')
dbListTables(mydb)
```

```
##  [1] "attribute"    "business"     "category"     "checkin"
##  [5] "elite_years"  "friend"       "hours"        "photo"
##  [9] "review"       "tip"          "user"         "user_id_1000"
```

```
dbListFields(mydb, 'review')
```

```
## [1] "id"          "stars"       "date"        "text"        "useful"
## [6] "funny"       "cool"        "business_id" "user_id"
```

```
review_100.sql = dbSendQuery(mydb, "select * from review limit 100")
review_100 = fetch(review_100.sql, n = -1)

dbListFields(mydb, 'business')
```

```
##  [1] "id"           "name"         "neighborhood" "address"
##  [5] "city"         "state"        "postal_code"  "latitude"
##  [9] "longitude"    "stars"        "review_count" "is_open"
```

```
business.sql = dbSendQuery(mydb, "select * from business")
business = fetch(business.sql, n = -1)

dbListFields(mydb, 'category')
```

```
## [1] "business_id" "category"
```

```
category.sql = dbSendQuery(mydb, "select * from category")
category = fetch(category.sql, n = -1)

dbListFields(mydb, 'user')
```

```
##  [1] "id"                "name"              "review_count"
##  [4] "yelping_since"     "useful"            "funny"
##  [7] "cool"              "fans"              "average_stars"
## [10] "compliment_hot"    "compliment_more"   "compliment_profile"
## [13] "compliment_cute"   "compliment_list"   "compliment_note"
## [16] "compliment_plain"  "compliment_cool"   "compliment_funny"
## [19] "compliment_writer" "compliment_photos"
```

```
user.sql = dbSendQuery(mydb, "select * from user")
user = fetch(user.sql, n = -1)
```