QMB 6358: Software Tools for Business Analytics
College of Business
University of Central Florida
Fall 2023

Final Examination

Due Thursday, December 7, 2023 at 12:59 PM
in your GitHub repo.

Instructions:

Complete this assignment within the space on your GitHub repo in a folder called `final_exam`. You may organize your files any way you like but leave your answers to all questions in this folder.

All of your responses can be completed using the language of your choice, as long as your solutions meet the specifications in each question. Store any printed output by writing or pasting into a document of your choice or typing comments in your code. This output can also be automated by redirecting ouput from a script in Question 6.

When you are finished, submit your code and any other documents by pushing your changes to your GitHub repo, following the instructions in Question 7. Complete these exercises individually and `git push` your own work.

Part A: Data Handling and Regression Modelling

Estimate the best regression model you can by solving as many of Questions 1 to 4 as you can. You do not necessarily have to solve them in order.

Question 1:

The folder `final_exam_2023` contains four `.csv` files: `renewals.csv`, `season_passes.csv`, `visits.csv`, and `demographic.csv`. The first dataset `renewals.csv` contains the following variables.

`customer_num`:	a unique identifier for each customer who bought an annual pass for Colossal Theme Parks
`renewed`:	a binary variable to indicate whether a customer who held an annual pass in 2021 renewed their membership to obtain an annual pass in 2022
`zip_code`:	the zip code in which the customer resides
`purchase_month`:	a categorical variable comprising the three-letter string that indicates the month of the year in which the customers bought their 2021 annual passes

Use this dataset to estimate a regression model to predict whether or not each customer `renewed` their annual pass.

a) Read in the `renewals.csv` dataset and store it in a data frame called `renewals` in your workspace.

b) Calculate and store the printed output from either a `summary` of the data or `describe` the data, according to your choice of software. Use this to get familiar with the contents of the dataset.

c) Estimate a regression model to predict `renewed` as a function of the other variables in the dataset. Ignore the variables `customer_num` and `zip_code`, which are keys for databases. Store the printed estimation output with the `print` and/or `summary` command, as appropriate.

Question 2:

Next re-create a modified version of the dataset `renewals.csv` from the original source, a table of the sales of season passes over the years 2021 and 2022. The dataset `season_passes.csv` contains the following variables.

`pass_num`:	a unique identifier for each annual pass
`customer_num`:	a unique identifier for each customer who bought an annual pass
`zip_code`:	the zip code in which the customer resides
`purchase_year`:	an integer representing the year in which the customer bought an annual pass
`purchase_month`:	a categorical variable comprising the three-letter string that indicates the month of the year in which the customer bought an annual pass
`purchase_day`:	an integer representing the day of the month in which the customer bought an annual pass
`pass_level`:	a categorical variable denoting one of the three tiers of annual pass membership: gold, silver, or bronze

Use the variables from this dataset to estimate a better regression model to predict renewals of annual passes.

a) Perform any pre-processing that needs to be done to the application data in `season_passes.csv` before re-creating `renewals.csv`: clean it, `sort` it or `read` it, according to your strategy of choice.

b) Split the data into two datasets, two tables, or two subqueries, by `purchase_year`, in order to join the renewals to previous annual pass holders using `customer_num` as a key.

c) Form a dataset `renewals_w_levels.csv`, a data frame, or a table `renewals_w_levels` by `paste`ing, `join`ing, or `merge`ing the pair of datasets or tables by `customer_num`, as needed. Be sure to keep the records of customers without annual passes in 2022 to keep track of the customers who did not renew. Drop all the observations for customers who only bought an annual pass in 2022, since we cannot observe a renewal for these customers.

d) Create an indicator variable `renewed` that equals one for the observations with customers who held annual passes in both 2021 and 2022. The remaining observations of customers who only held annual passes in 2021 should be assigned zero.

e) If not already done in the above, `read` the new dataset and store it in a data frame called `renewals_w_levels` in your workspace.

f) Calculate and store the printed output from either a `summary` of the data or `describe` the data, according to your choice of software. Use this to get familiar with the contents of the dataset.

g) Estimate a regression model to predict `renewed` as a function of the other variables in the dataset from the dataset of annual passes purchased in 2021. In other words, do not use any information that would only be known once the passes were renewed. Ignore the variables `pass_num`, `customer_num`, and `zip_code`, which are keys for databases. Store the printed estimation output with the `print` and/or `summary` command, as appropriate.

Question 3:

Now join your table or data frame from either Questions 1 or 2 above to the file `demographic.csv` in the folder `final_exam_2023`. The dataset `demographic.csv` contains the following variables.

`zip`:	the zip code to indicate each geographic region
`avg_income`:	the average income in each zip code
`density`:	the population density in each zip code
`avg_num_children`:	the average number of children per household in each zip code

Use the variables from these datasets to estimate an even better regression model to predict renewals of annual passes.

a) Perform any pre-processing that needs to be done to the file `demographic.csv` before joining it to the others: clean, `sort` or `read`, according to your strategy of choice.

b) Form a dataset `renewals_w_demo.csv` by `paste`ing, `join`ing, or `merge`ing the datasets, as needed.

c) If not already done in the above, `read` the new dataset and store it in a data frame called `renewals_w_demo` in your workspace.

d) Calculate and store the printed output from either a `summary` of the new variables or `describe` the new variables, according to your choice of software. Use this to get familiar with the contents of the dataset.

e) Estimate a regression model to predict `renewed` as a function of the other variables in the dataset, using only the variables corresponding to the annual passes purchased in 2021. Ignore the variables `pass_num`, `customer_num`, `zip`, and `zip_code`, which are keys for databases. Store the printed estimation output with the `print` and/or `summary` command, as appropriate.

Question 4:

Now calculate new variables to estimate a model for annual pass renewals using a dataset `visits.csv` in the folder `final_exam_2023`. Join any new variables you create to the dataset for your best model from Questions 1 through 3. The dataset `visits.csv` contains the following variables.

`pass_num`:	a unique identifier for each annual pass
`visit_year`:	an integer representing the year in which the customer visited one of the parks
`visit_month`:	a categorical variable comprising the three-letter string that indicates the month of the year in which the customer visited one of the parks
`visit_day`:	an integer representing the day of the month in which the customer visited one of the parks
`park_name`:	a categorical variable denoting one of the three theme parks: Colossal Kingdom, Colossal Zoo, or Colossal Studios

Create a new dataset, data frame, or table called `visits_agg` to join to one of your datasets.

a) First, determine which of the observations in `visits.csv` correspond to visits made by customers who purchased an annual pass in 2021 *before* renewing their pass. You will notice that these customers obtain a new `pass_num` upon renewal in 2022.

b) Select only the relevant subset of `visits.csv` according to your determination in part (a). For these observations, create as many of the following variables as you can. These variables should be aggregated by `pass_num` before being joined to the dataset for your best model from Questions 1 through 3.

   i) The count of the number of visits to any of the three parks in a new variable called `num_visits`.

   ii) The count of the number of visits to each of the three parks, in new variables called `num_visits_kingdom`, `num_visits_zoo`, and `num_visits_studios`.

   iii) An indicator variable `has_visited` for having visited any of the three parks. It equals one if a customer with this `pass_num` is in the database.

   iv) Indicator variables to indicate that a customer has visited any of the three parks: `has_visited_kingdom`, `has_visited_zoo`, and `has_visited_studios`.

c) Join these variables to one of your previous datasets. When joining the above variables to the other data, make sure that observations with no value for the above variables are recorded as zeros.

d) Calculate and store the printed output from either a `summary` of the new variables or `describe` the new variables, according to your choice of software. Use this to get familiar with the contents of the dataset.

e) Estimate a regression model to predict `renewed` as a function of the other variables in the dataset, using only the variables corresponding to the annual passes purchased in 2021. Ignore the variables `pass_num`, `customer_num`, `zip`, and `zip_code`, which are keys for databases. Store the printed estimation output with the `print` and/or `summary` command, as appropriate.

Part B: Automatic Document Generation

Question 5:

In this exercise, you will create a simple LaTeX document that highlights your best example among the models estimated above in Part A. To begin this exercise, you will need to have saved your chosen dataset in the `Data` folder.

a) In an R script, read in a dataset that contains all variables required to estimate the best model from Part A.

b) Estimate your model including all the variables from your best model from Part A.

c) Create a LaTeX table of the regression output from the same model, using the `texreg` package.

d) Print the LaTeX code for the table to a text file and save it in the `Tables` folder.

e) Input this text file within a simple LaTeX script that `input`s only the selected table, so that running your LaTeX script will create a `pdf` file with the single table.

f) Make sure your shell script `final_exam.sh` includes code to run your R script and the commands to build your LaTeX table.

Part C: Software Management and Version Control

Question 6:

Create a UNIX shell script called `final_exam.sh` that runs all the software to answer Questions 1 to 5 in Parts A and B.

a) Use commands such as `Rscript`, `python3`, or `sqlite3` to run your software.

b) Redirect the output of each script to appropriately-named `.txt` or `.out` files, using the "`>`" operator, to save your output.

c) You can test your script by running `./final_exam.sh`.

Question 7:

Push your completed files to your GitHub repository following these steps. See the `README.md` and the `GitHub_Quick_Reference.md` in the folder `demo_02_version_control` in the QMB6358F23 course repository for more instructions.

1. Open GitBash and navigate to the folder inside your local copy of your git repo containing your assignments. Any easy way to do this is to right-click and open GitBash within the folder in Explorer. A better way is to navigate with UNIX commands.

2. Enter `git add .` to stage all of your files to commit to your repo. You can enter `git add my_filename.ext` to add files one at a time, such as `my_filename.ext`. in this example.

3. Enter `git commit -m "Describe your changes here"`, with an appropriate description, to commit the changes. This packages all the `add`ed changes into a single unit and stages them to `push` to your online repo.

4. Enter `git push origin main` to push the changes to the online repository. After this step, the changes should be visible on a browser, after refreshing the page.