

# Exercise III - R

## Part 2 – EB Method

---

CEE412 / CET522

TRANSPORTATION DATA MANAGEMENT AND VISUALIZATION

WINTER 2020



# Introduction

---

- There are a total of **seven questions** in this Exercise that you are expected to answer and submit as **Assignment 4**.
- Your assignment solution should include your answers, as well as all of the codes you have written for each question, both R and SQL.
- The assignment questions are labeled with question numbers and highlighted in **red**.

# Getting Started

- We are going to use a new dataset, so assuming you already have R running and a database connection in place, query all data from the following table and load into R as a dataframe:

`[CEE412_CET522_W20].[dbo].[A4_AccidentCount]`

- Take a look at the available data by printing the top 10 rows

|    | Link_ID | ST_MP | Length | NLane | LaneWidth | LShoulderWidth | RShoulderWidth | AADT  | AccCount | RouteNo |
|----|---------|-------|--------|-------|-----------|----------------|----------------|-------|----------|---------|
| 1  | 1       | 0.00  | 0.07   | 1     | 12        | 10             | 4              | 6225  | 10       | 5       |
| 2  | 2       | 0.24  | 0.19   | 1     | 12        | 10             | 4              | 6225  | 2        | 5       |
| 3  | 3       | 0.43  | 0.04   | 1     | 12        | 10             | 4              | 6225  | 0        | 5       |
| 4  | 4       | 0.50  | 0.07   | 1     | 12        | 10             | 4              | 6225  | 2        | 5       |
| 5  | 5       | 0.04  | 0.35   | 1     | 12        | 10             | 4              | 9147  | 2        | 5       |
| 6  | 6       | 0.00  | 0.18   | 1     | 12        | 10             | 4              | 7202  | 3        | 5       |
| 7  | 7       | 0.00  | 0.03   | 1     | 13        | 4              | 4              | 24044 | 2        | 5       |
| 8  | 8       | 0.28  | 0.01   | 1     | 14        | 6              | 0              | 24044 | 0        | 5       |
| 9  | 9       | 0.92  | 0.01   | 2     | 13        | 6              | 2              | 27763 | 0        | 5       |
| 10 | 10      | 1.46  | 0.15   | 4     | 12        | 5              | 5              | 37503 | 15       | 5       |

# Getting Started

---

- We are going to model **AccCount** as the response variable, but we want to use **log(Length)** and **log(AADT)** in the predictor set instead of the given values.
- I have named my data `AccCnt_Data`, the following commands can do this conversion (do it for both AADT and Length)

```
AccCnt_Data$log_AADT = log(AccCnt_Data$AADT)
AccCnt_Data$log_Length = log(AccCnt_Data$Length)
```

- Notes:
  1. Two new columns are created in the dataframe to store the result.
  2. AADT = Annual Average Daily Traffic
  3. Sometimes numeric data will be imported as factors, there are a couple ways to deal with this. One way is to use the `as.numeric()` function to convert factors to numeric.

# Modeling Accident Data

---

- Create a negative binomial model with `AccCount` as the response and the full predictor set (i.e., `log(Length)`, `NLane`, `LaneWidth`, `LShoulderWidth`, `RShoulderWidth`, and `log(AADT)`).
- Make sure you include `log(Length)` but not `Length` in the model (and do the same for `log(AADT)`).
- Summarize the model, and take a look at the result.

**Question 1:** What variables are significant at  $\alpha = 0.05$ ?

**Question 2:** What is the dispersion parameter and AIC?

- Example shown here for obtaining the dispersion parameter from a negative binomial model named “model.nb”:

```
k = model.nb$theta
```

# Modeling Accident Data

---

- Create a Poisson model for the same data, and compare the resulting AIC.


**Question 3:** Based on the AIC, which model would you choose?

# Modeling Accident Data

---

- Drop the LaneWidth variable from the NB model
- You can recreate the model with reduced predictor set, or drop variables from the original model using the update method as shown:

```
new_model = update(original_model, .~.-LaneWidth)
```



This means the new model will include all predictors and response in the original model except the predictor named “LaneWidth”

**Question 4:** Is the new NB model better or worse than the original NB model (in terms of AIC)?

# EB Calculations

---

- The rest of this exercise will use the negative binomial model created on the last slide, not the Poisson model.
- Get predicted values for the model, the result should be a vector with length equal to the number of rows in your data frame
- You can use the `predict(model, type="response")` function. Alternatively, you can obtain the model fitted values from the model result as shown:

```
predictions = model$fitted.values
```



# EB Calculations

---

- Compute the weight values ( $\alpha$ ) for all road segments, equation given below:

$$\alpha_i = \frac{1}{1 + SPF_i / (kL_i^\gamma)}$$

where,

$\alpha_i$  = weight value for segment  $i$

$SPF_i$  = the predicted crash count for segment  $i$

$k$  = the dispersion parameter from the NB model (theta in R)

- Note: In R, dividing a vector by a constant or another vector results in element-wise division.

# EB Calculations

---

- Compute the EB estimated safety for each segment, equation given below:

$$\pi_i = \alpha_i SPF_i + (1 - \alpha_i) K_i$$

where,

$\pi_i$  = the EB estimated safety for segment  $i$

$K_i$  = actual accident count for segment  $i$

$\alpha_i$  = weight parameter for segment  $i$  from the previous page

# EB Calculations

---

- Compute the Accident Reduction Potential for each segment:

$$ARP_i = (1 - \alpha_i)(K_i - SPF_i)$$

where,

$ARP_i$  = Accident Reduction Potential for segment  $i$

# EB Calculations

---

- Create a data frame of your final results. Assuming you have the following vectors representing various stages of your analysis:
  1. SegID = road segment id number, Link\_ID from the original dataset
  2. RouteNo = Route number, RouteNo from the original dataset
  3. BegMP = road segment starting milepost, ST\_MP from the original dataset
  4. AccCount = Accident count, AccCount from the original dataset
  5. SPF = the predicted accident count from the NB model
  6. EB\_SAFE = the EB estimated safety ( $\pi$ )
  7. ARP = Accident Reduction Potential, from previous step
- Bind them together using the data.frame() method:

```
Result <- data.frame(SegID, RouteNo, BegMP, AccCount, SPF, EB_SAFE, ARP)
```

# EB Calculations

---

- Order the result and show the top 15 road segments in terms of ARP
- You can use the `order()` method to order a dataframe as shown (similar to the `order by` in SQL, use the negative sign to order descending):

```
Result.ordered <- Result[with(Result,order(-ARP)),]
```

**Question 5:** Show the top 15 road segments in terms of ARP.

**Question 6:** Would a road segment which is predicted (by NB model) to have a very high accident count be likely to be identified as a high safety treatment priority using this method? Why or why not?

# EB Calculations

---

- Save the results of your analysis in a SQL database.
- Note: to save the results into SQL, you must have write privileges in the default database for your DSN. But this is not the case if you are using the same DSN created in the R Exercise Part 1 (as the default database is the class database where you only have data reader permissions).
- You can create another DSN for which the default database is your personal database, and then save your result using a similar code as shown:

```
sqlSave(conn2, Result, "AccCnt_EB")
```

- Where conn2 is the connection object linked to the new DSN; Result is the dataframe I created containing the EB results; and AccCnt\_EB is the name I am giving to the SQL table this command will create.

# Rank Results in SQL

- Open SQL Server Management Studio and take a look at the result of your work.
- Write a SQL query to return the entire dataset, with a column indicating the rank in terms of ARP for each segment, partitioning on route number.
- You need to make use of a window function, and this is the basic form that will appear in the **SELECT** list of your SQL query:

Attribute(s) that will be the basis for the order of the ranking function (ARP)

```
DENSE_RANK() OVER (PARTITION BY Attributes1, ORDER BY Attributes2)
```

The ranking function, there are several to choose from

Attribute(s) that will be the basis for subdividing the table (route number)

- You can find more information of window functions in Lecture Slide 7.

# Rank Results in SQL

---

**Question 7:** Show the top 3 road segments for each route in terms of ARP based on the EB results you saved in your personal database.