Take Home Points

Hwk #3: Data Visualization & More
Biostat 202: Opportunities and Challenges of "Big Data"
Due Wednesday, August 16th by Midnight

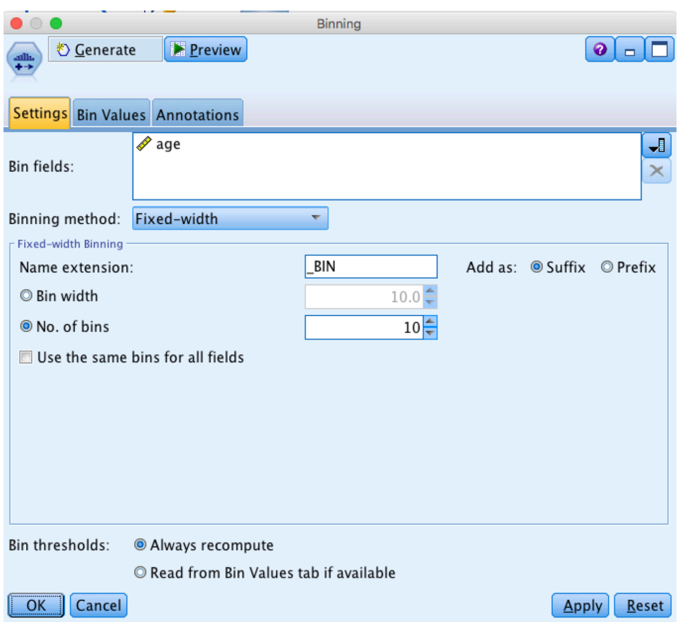
For this homework we will also use the data from the National Electronic Injury Surveillance System (NEISS) that you used for Hwk #1 (http://www.cpsc.gov/en/Research--Statistics/NEISS-Injury-Data/). We will be interested in creating plots of the data and examining how to evaluate models graphically.

Open your stream from Hwk #2. I have also included the data as an Excel file in Hwk #2 if you prefer to use this instead of the .csv file. You should have the recoded variables for admit, Head, Toe and Age already in your Type node.

1. (3 points) Using the codes in the table below from the NEISS codebook, choose two more 'Body Part Affected' codes and create new variables as you did in Hwk #2 for Head & Toe. **Q1. Which areas of the body did you choose? Do you think that injuries to these new areas are likely to involve hospitalization?**

2. (3 points) Using the codes in the table below from the NEISS codebook, choose three diagnoses from the 'Diagnosis' table and create variables. **Q2. Which three diagnoses did you choose? Do you think any of these are related to the body parts chosen above?**

3. (5 points) Attach a Binning node from Field Ops and create Age groups to use in analyses that will not accept continuous data. Create 10 bins (or groups)

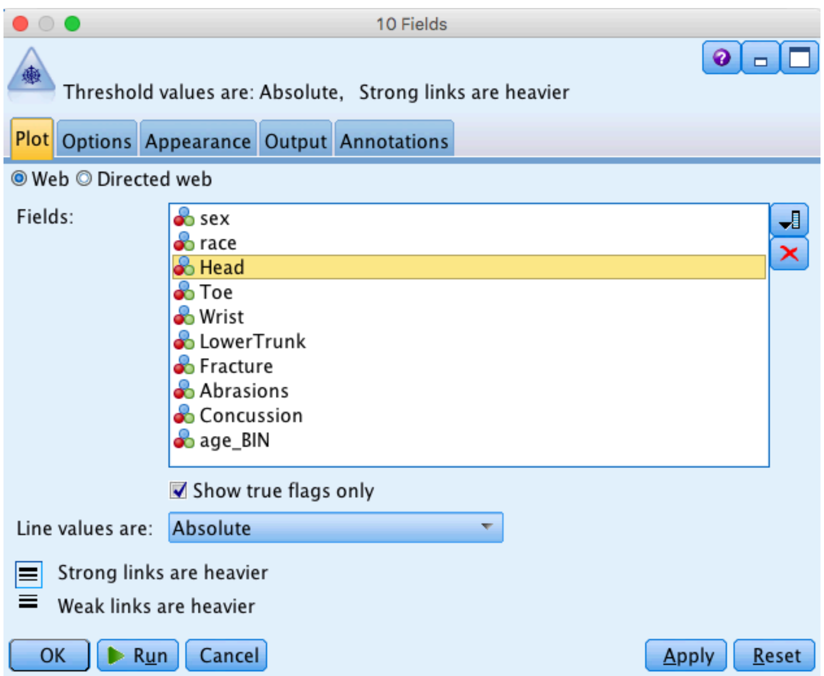


Attach a Type node following all your reclassification nodes and binning node. For the plots and models, we will use the following variables: a continuous Age field ranging from 1 to 107, an admit variable (0/1), 4 body part variables (0/1), 3 diagnosis variables (0/1), a gender variable (0, 1, 2), and an Age_bin for our age groups. Run an Audit node and then get a distribution of your new age_BIN variable by double clicking on the bar chart for the variable. **Q3. By binning by number of bins rather than the actual distribution of Age, what is this distribution telling you about the population?**

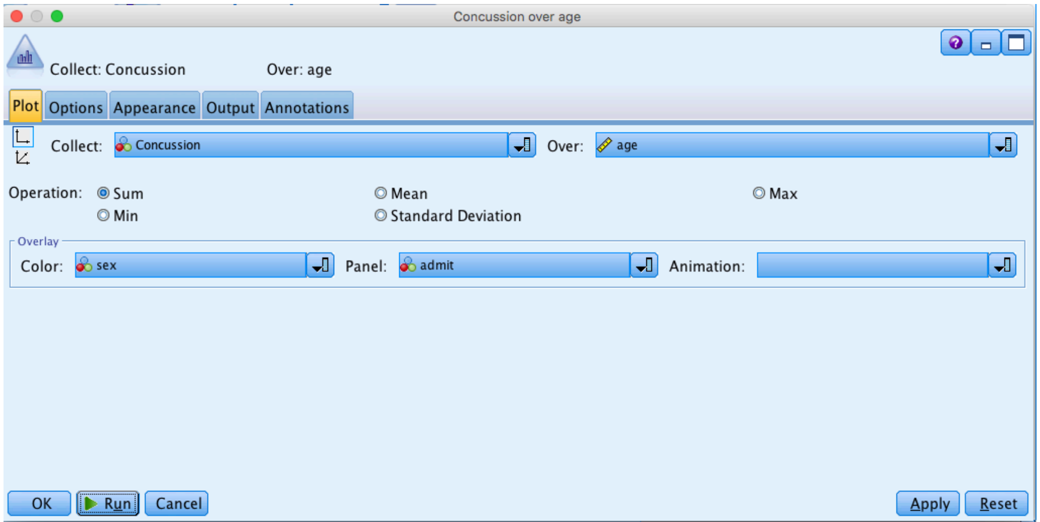
In this question, the purpose is to reformat a variable from continuous to categorical. What this allows you to do is to directly compare within the different categories (or age groups), something that is more difficult to do when you have a continuous variable. For example, you can now more easily see that although there were more ED visits in the younger age categories (ED visits are skewed towards the younger population), the percentage of admissions is much higher in the older age categories. By changing the number of bins, you can change the granularity of the graph to be able to make a better interpretation depending upon your research question.

4. (4 points) Attach a web graph to your Type node and include the variables above (except Age). Use the bar below the web to examine how the variables are related. **Q4. Which variables show relationships? Are they variables that occur together or that do not occur together? Why would links between diagnoses and body parts that are not related be stronger than those that are related?**

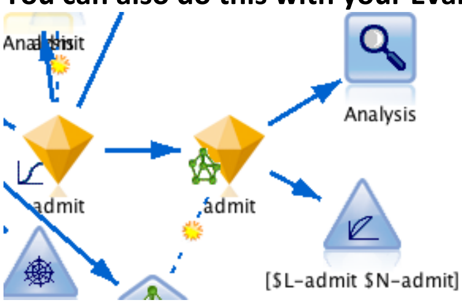
One thing to keep in mind when looking at the web graphs is that the strength of the lines is dependent upon the number of observations that are linking the two. Hence, you will notice that diagnoses and body parts that are not related are stronger compared to those that are (just because of the sheer number of observations that are unrelated). One of the take home points is that for infrequent observations, web graphs may not be as useful compared to datasets with more frequent observations with fewer variables to deal with (e.g. the titanic dataset where the web graph was more interpretable).



5. (5 points) Create a Collection graph (from the Graph tab): Collect Concussion Over Age and Overlay Sex (Color) and Panel admit. **Q5. What does this tell you about concussions by age and admission to the hospital?**



6. (4 points) Finally, fit a Logistic model and a Neural net with admit as the Target and using your new diagnosis and body_part variables as well as Age and Sex. Attach Analysis nodes and Evaluation graphs to each model. **Q6. Are there differences in the accuracy of the models? Do the Evaluation plots differ? Are these models better than those fit for last week's homework? Attach the Neural model gem to the Logistic model gem and run an Analysis node. This will tell you how much your 2 models agree. Do they agree? You can also do this with your Evaluation plots.**



You will notice that the output between the evaluation plot and the analysis node may be incongruent. The accuracy of the model based upon the analysis node is roughly 90% while the line in the evaluation node is very close to the red (chance) line suggesting that the model is actually not that accurate! The reason for this discordance is because of the difference in methods by which SPSS uses to evaluate the model. Evaluation plots (or ROC plots) utilize several different cut-off points to determine the best sensitivity and specificity of a model. On the other hand, the analysis node only uses one cut-off point (typically the best one) and provides the accuracy of the model based upon that one cut-off value. Hence, the ROC/Evaluation plot is thought to be the better method to assess "accuracy" of a model compared to the analysis plot. Depending upon which variables you use for your model, you may get different outputs with respect to your evaluation plot and/or the accuracy plot although between neural network and logistic regression, the models should perform very similarly.

7. (5 points) The data collection from the Health eHeart study (co-led by Mark Pletcher) generated a Fitbit dataset that recorded the number of steps taken each minute for a number of participants across a number of days. Participants contributed between 10 and 2000 days each.

The data available are the date of the measurement, listed in the format ("29mar2013"), which minute during the day the steps were recorded (from 1 to 1400 = 60minutes*24hours) and the number of steps for that date and that minute. Because the data is massive (an individual with 2000 days would contribute almost 3 million rows of data), minutes with no activity are left out of the dataset.

Suppose we wanted to see if activity patterns differed between individuals with different levels of severity of multiple sclerosis. Summarizing the data simply as average number of steps per day (a single variable) may too coarsely summarize the information. It may be that different levels of disease are associated with different patterns during the day or during the week. Suppose we want to manipulate the data to reduce it to a more manageable size, recording (for each person) the average number of steps for each minute during a week. That is, we want to reduce the data to (7 days)*(1440 minutes) = 10,080 values per person by averaging across the same time during the day and day during the week. Said another way, suppose we want to have average step counts for Monday at 12:01 am, Monday at 12:02 am, …, Sunday at 11:59pm, Sunday at 12:00 midnight. **Q7. Describe the steps you would need to go through to compute those averages. No programming required!**

There are many ways to do this…but the basic steps are

1) **Find some convenient way to store the information (e.g. in a matrix)**
2) **Find some way to save the step counts into the matrix**
3) **Once you have the matrix of step counts per minute, then you can easily do statistics/operations on the matrix**

Here is one method using pseudocode:

Assuming that the data are essentially counts with a timestamp, e.g. step 1 – Monday, April 21, 2016 @ 12:00:03.45, we can go through the following steps to find the average number of steps for each minute during the week.

```
Create variable minute_count[0,1440] = 0 (creates a minute variable to cycle through)
Create variable day_of_the_week[0,6] = 0 (creates a day variable to cycle through)
Create steps_matrix[0:6;0:1440] = 0 (creates a matrix to store all the steps for the week = 10,080 values)
N, j, k = 0 (variables for stepping through the days of the week and minutes of the day)
For each day of the week day_of_the_week = n,6 (step though each day of the week)
        For each minute of the day minute_count = j,1440
                k = sum of counts of steps within each minute (obtained from raw data)
                update step_matrix[n,j] = k (stores the steps in each minute in the matrix)
                j+1 (go to the next minute and continue loop until minute 1440)
        j = 0 (resets the minute of the day in preparation for the next day)
        n+1 (go to the next day and continue loop until day 6)
Calculate average steps in steps_matrix
```

For #1

Body Part Affected	**Code**
Arm, lower (not including elbow or wrist)	33
Arm, upper	80
Ankle	37
Ear	94
Elbow	32
Eyeball	77
Face (including eyelid, eye area and nose)	76
Finger	92
Foot	83
Hand	82
Head	75
Internal (use with aspiration and ingestion)	00
Knee	35
Leg, lower (not including knee or ankle)	36
Leg, upper	81
Mouth (including lips, tongue and teeth)	88
Neck	89
Pubic region	38
Shoulder (including clavicle, collarbone)	30
Toe	93
Trunk, lower	79
Trunk, upper (not including shoulders)	31
Wrist	34
25-50% of body	84
All parts of body (more than 50% of body)	85
Not recorded	87

Diagnosis	Diagnosis Code	Body Part Code
Anoxia	65	85
Electric shock	67	85
Poisoning	68	85
Submersion (incl. drowning)	69	85

For #2

Diagnosis	Code
Amputation	50
Anoxia	65
Aspirated foreign object	42
Avulsion	72
Burns, scald (from hot liquids or steam)	48
Burns, thermal (from flames or hot surface)	51
Burns, chemical (caustics, etc.)	49
Burns, radiation (includes all cell damage by ultraviolet, x-rays, microwaves, laser beam, radioactive materials, etc.)	73
Burns, electrical	46
Burns, not specified	47
Concussions	52
Contusions, Abrasions	53
Crushing	54
Dental injury	60
Dermatitis, Conjunctivitis	74
Dislocation	55
Electric shock	67
Foreign body	56
Fracture	57
Hematoma	58
Hemorrhage	66
Ingested foreign object	41
Internal organ injury	62
Laceration	59
Nerve damage	61
Poisoning	68
Puncture	63
Strain or Sprain	64
Submersion (including Drowning)	69
Other/Not Stated	71