CSC 458 Data Mining and Predictive Analytics I, Fall 2019

Dr. Dale E. Parson, Assignment 3, Using Weka numeric-predicting models to correlate several meteorological and temporal attributes with numeric Hawk Mountain BW migration counts for fall 2017 & fall 2018. Due by 11:59 PM on Wednesday November 13 via make turnitin.

Perform the following steps to set up for this semester's projects and to get my handout. Start out in your login directory on csit (a.k.a. acad).

```
cd  $HOME
mkdir  DataMine  # This should already be there from assignment 1.
cp  ~parson/DataMine/csc458fall2019assn3.problem.zip DataMine/csc458fall2019assn3.problem.zip
cd   ./DataMine
unzip  csc458fall2019assn3.problem.zip
cd  ./csc458fall2019assn3
```

EDIT THE SUPPLIED FILE README.txt when the following questions starting at Q1 below. Keep with the supplied format, and do not turn in a Word or PDF or other file format. I will deduct 20% for other file formats, because with this many varying assignments being turned in, I need a way to grade these in reasonable time, which for me is a batch edit run on the **vim** editor.

STEP1: From the assignment directory, load file **FilteredCSC458assn3.arff** into Weka. This file is identical to the FilteredCSC458assn2.arff file that we saved in Assignment 2 with two exceptions. First, **HawkYear** is now numeric rather than a string to be converted to a nominal value. Even though discrete years such as HawkYear are typically converted to nominal sets such as {2017, 2018} because they are not continuous numeric ranges, Assignment 3 uses them as numeric data because of an AddExpression filtering step that requires numeric data. Second, **SunYear** is deleted because it is 100% redundant with HawkYear, contributing no new information, but potentially adding complexity to the learning process for some models. There are remaining attributes that are partially redundant with each other. We will inspect the effects of that redundancy.[1] **Q1** through **Q15** are worth 6.66% each.

Q1. Run the **ZeroR** classifier under functions after loading this ARFF file and paste the following results. What is the basis for the "ZeroR predicts class value"?

```
ZeroR predicts class value: n.n
…
Correlation coefficient            n.n
Mean absolute error                n.n
Root mean squared error            n.n
Relative absolute error            n      %
Root relative squared error        n      %
Total Number of Instances            2255
```

What is the basis for the "ZeroR predicts class value"?

[1] https://faculty.kutztown.edu/parson/fall2019/csc458fall2019legend.html has the attribute legend for all attributes being studied in this semester's projects.

Q2: Run the **LinearRegression** classifier under **functions** and paste the following results. Are the error measures[2] for LinearRegression better or worse than those for ZeroR? What do you think accounts for the changes in degree of error going from **ZeroR** to **LinearRegression**? Remember that LinearRegression attempts to fit the relationships of non-target attributes → target attribute to a multidimensional line.

```
Correlation coefficient           n.n
Mean absolute error               n.n
Root mean squared error           n.n
Relative absolute error           n     %
Root relative squared error       n     %
Total Number of Instances          2255
```

Q3: Apply the **filter unsupervised -> attribute -> Normalize** after setting **ignoreClass to true**. Make sure that BW gets normalized. This step recalibrates each attribute on the scale (value – minValue)/(maxValue – minValue) as a percentage, i.e., [0, 100]% of its range. We are doing this in order to interpret weights (coefficients) in the LinearRegression formula on a normalized scale. Run **LinearRegression** on this filtered dataset and paste both the LinearRegression formula and its results. Have the error measures gotten better, worse, or stayed the same? Ignore **Mean absolute error** and **Root mean squared erro**r because they are no longer in application units. Instead, compare **Correlation coefficient**, **Relative absolute error**, and **Root relative squared error** to Q2 results, since those measures are always normalized.

```
Linear Regression Model
BW =
     PASTE THE FULL FORMULA HERE.

Correlation coefficient           n.n
Mean absolute error               n.n
Root mean squared error           n.n
Relative absolute error           n     %
Root relative squared error       n     %
Total Number of Instances          2255
```

UNDO the **Normalize** filter changes after completing Q3, and verify that attributes including BW have returned to their unnormalized ranges.

Q4. What are the top six attributes in the Q3 LinearRegression formula, in terms of coefficient weights? Ignore numeric signs; compare on basis of numeric magnitude. Paste both the coefficient multiplier and its attribute from the formula, one per line as in the Weka output. Are any of these attributes redundant with each other, i.e., they have almost identical meaning or very close correlation with each other because they measure essentially the same thing? Explain.

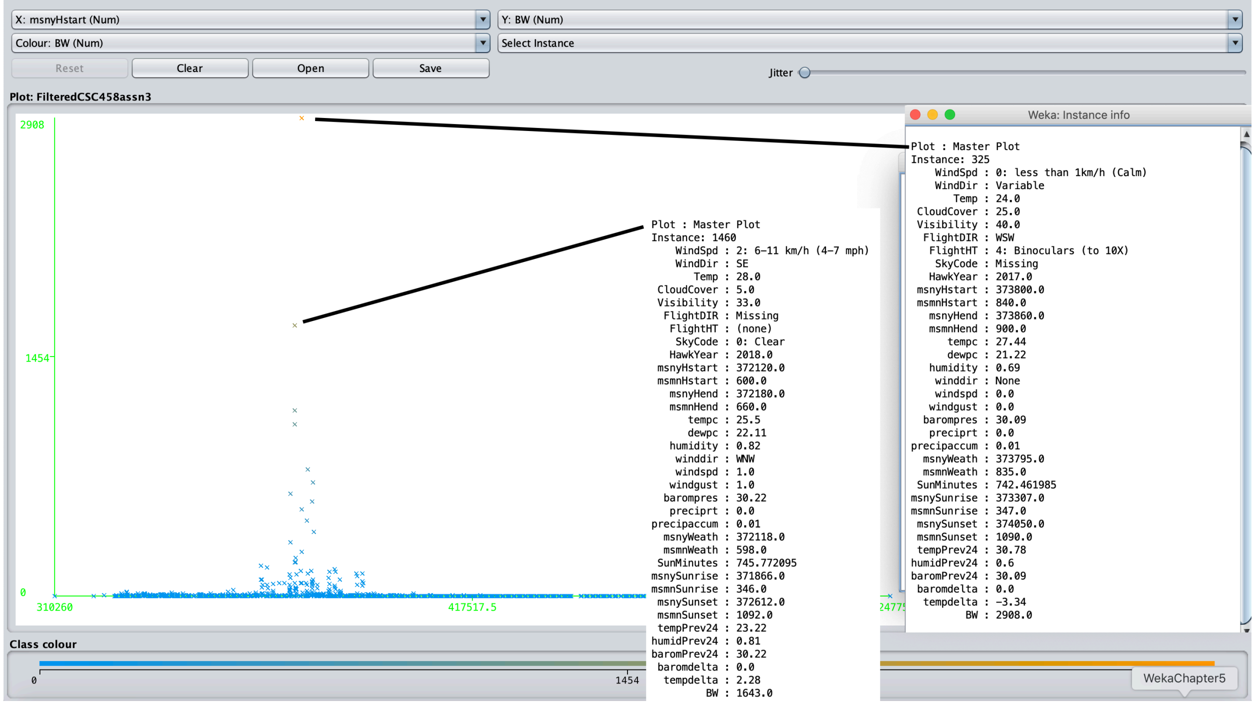
Note that when a pair of redundant attributes have opposite signs – one positive, the other negative – then the one of smaller magnitude is used to make an adjustment on the weight of the other. When they both have the same sign, they reinforce each other.

[2] Definitions for error measures appear in Chapter 5 slides, slides 60 "Evaluating numeric prediction" through 64 ""Which measure?. Recall from my lecture that I usually consult **Mean absolute error** and **Root mean squared error** because they are in application units, BW counts for this dataset. The % measures are simply those values normalized across the range [0, maxErrorValue]. **Mean absolute error** emphasizes the average error, and **Root mean squared error** emphasizes the outliers, so it is much bigger **when there are significant outliers**.

Q5: In Weka's Select Attributes tab hit Start with the default configuration parameters and paste the Selected attributes below, including the list of actual attributes. Also, click CfsSubsetEval to inspect this Attribute Evaluators documentation paragraph, and paste that below. Which attributes from your top six in Q4 appear in Q5's results. What accounts for the difference in important attributes?

Selected attributes: Indices of selected non-target attributes
Actual list of most important non-target attributes, one per line, from Weka output
CfsSubsetEval :

Evaluates … (complete this sentence using paste).



X: msnyHstart (Num)
Y: BW (Num)
Colour: BW (Num)
Select Instance

Plot: FilteredCSC458assn3

```
Plot : Master Plot
Instance: 1460
     WindSpd : 2: 6–11 km/h (4–7 mph)
     WindDir : SE
        Temp : 28.0
  CloudCover : 5.0
  Visibility : 33.0
   FlightDIR : Missing
    FlightHT : (none)
     SkyCode : 0: Clear
    HawkYear : 2018.0
  msnyHstart : 372120.0
  msmnHstart : 600.0
    msnyHend : 372180.0
    msmnHend : 660.0
       tempc : 25.5
       dewpc : 22.11
    humidity : 0.82
     winddir : WNW
     windspd : 1.0
    windgust : 1.0
   barompres : 30.22
    preciprt : 0.0
 precipaccum : 0.01
   msnyWeath : 372118.0
   msmnWeath : 598.0
  SunMinutes : 745.772095
 msnySunrise : 371866.0
 msmnSunrise : 346.0
  msnySunset : 372612.0
  msmnSunset : 1092.0
  tempPrev24 : 23.22
 humidPrev24 : 0.81
 baromPrev24 : 30.22
  baromdelta : 0.0
   tempdelta : 2.28
          BW : 1643.0
```

```
Plot : Master Plot
Instance: 325
    WindSpd : 0: less than 1km/h (Calm)
    WindDir : Variable
       Temp : 24.0
 CloudCover : 25.0
 Visibility : 40.0
  FlightDIR : WSW
   FlightHT : 4: Binoculars (to 10X)
    SkyCode : Missing
   HawkYear : 2017.0
 msnyHstart : 373800.0
 msmnHstart : 840.0
   msnyHend : 373860.0
   msmnHend : 900.0
      tempc : 27.44
      dewpc : 21.22
   humidity : 0.69
    winddir : None
    windspd : 0.0
   windgust : 0.0
  barompres : 30.09
   preciprt : 0.0
precipaccum : 0.01
  msnyWeath : 373795.0
  msmnWeath : 835.0
 SunMinutes : 742.461985
msnySunrise : 373307.0
msmnSunrise : 347.0
 msnySunset : 374050.0
 msmnSunset : 1090.0
 tempPrev24 : 30.78
humidPrev24 : 0.6
baromPrev24 : 30.09
 baromdelta : 0.0
  tempdelta : -3.34
         BW : 2908.0
```

Figure 1: BW (Y axis) count as a function of msnyHstart (X axis) with max hour for 2017 & 2018 Consult HawkYear in the pasted Instance Info panels to determine which is which.

Q6: Based on Figure 1, why might LinearRegression have a hard time modeling this data?

Q7. Run the **M5P** tree classifier and paste the following results. How do its error measures compare with ZeroR in Q1 and LinearRegression in Q2?

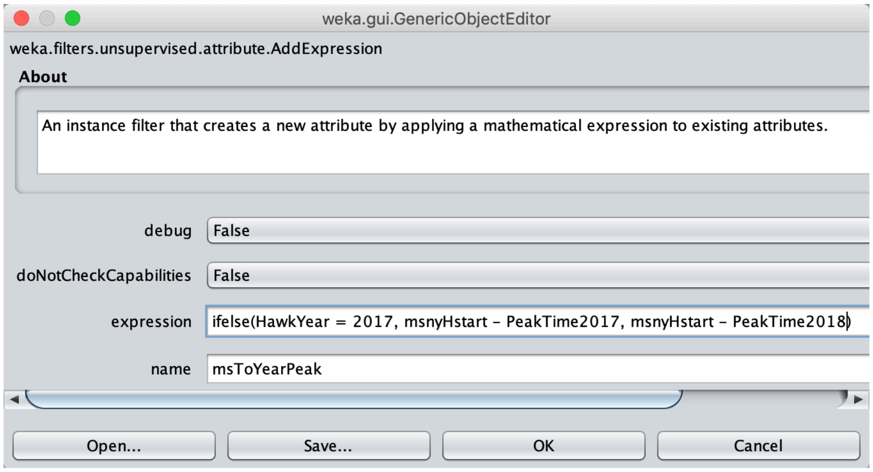
Number of Rules : N
Correlation coefficient n.n
Mean absolute error n.n
Root mean squared error n.n
Relative absolute error n %
Root relative squared error n %
Total Number of Instances 2255

After our field trip to Hawk Mountain, a photo and summary appeared in *The Daily Brief*, the KU email-based newsletter for faculty and staff. Our Provost and a former biology professor, Dr. Anne Zayaitz, wrote me to congratulate. We got into a discussion about this dataset, and she wrote, "It's not really surprising that the data re migration are non linear—what's probably a bit more interesting to a biologist are those birds that are widely trailing the main group or are coming much earlier than most. We like 'average' but it's the outliers that peak curiosity and questions!" We will take this approach shortly, after a few preliminary steps.

Q8: Use Weka's EDIT window to sort instances on BW in descending order. Note the msnyHstart time of year for the maximum-BW 2017 observation hour, and separately for the maximum-BW 2018 observation hour. What are those msnyHstart values and their corresponding BW counts by year? While you are in the Edit window, scroll down and look at HawkYear-msnyHstart-BW values for the top six or seven lines. Note how the years interleave. Close the Edit window after you answer this question.

2017 msnyHstart N BW N
2018 msnyHstart N BW N

STEP1.5: In experimenting with Weka, I discovered this week that the **AddExpression** filter that we used last assignment supports the ifelse(), conditional expression construct, meaning that we can avoid using the mutating MathExpression filter. Apply AddExpression as it appears in Figure 2 to create derived attribute **msToYearPeak**, substituting the appropriate aN attribute number (like **a100** for the 100th attribute) for HawkYear and msnyHstart, and the peak BW msnyHstart values from Q8 for PeakTime2017 and PeakTime2018. After applying, re-open the Edit window, sort in descending order on BW, and make sure that the peak 2017 and 2018 BW instances have the new **msToYearPeak** derived attribute set to 0.


weka.gui.GenericObjectEditor

weka.filters.unsupervised.attribute.AddExpression

About

An instance filter that creates a new attribute by applying a mathematical expression to existing attributes.

debug: False

doNotCheckCapabilities: False

expression: ifelse(HawkYear = 2017, msnyHstart - PeakTime2017, msnyHstart - PeakTime2018)

name: msToYearPeak

Figure 2: AddExpression for determining minutes from current msnyHstart, to msnyHstart for peak BW, by year

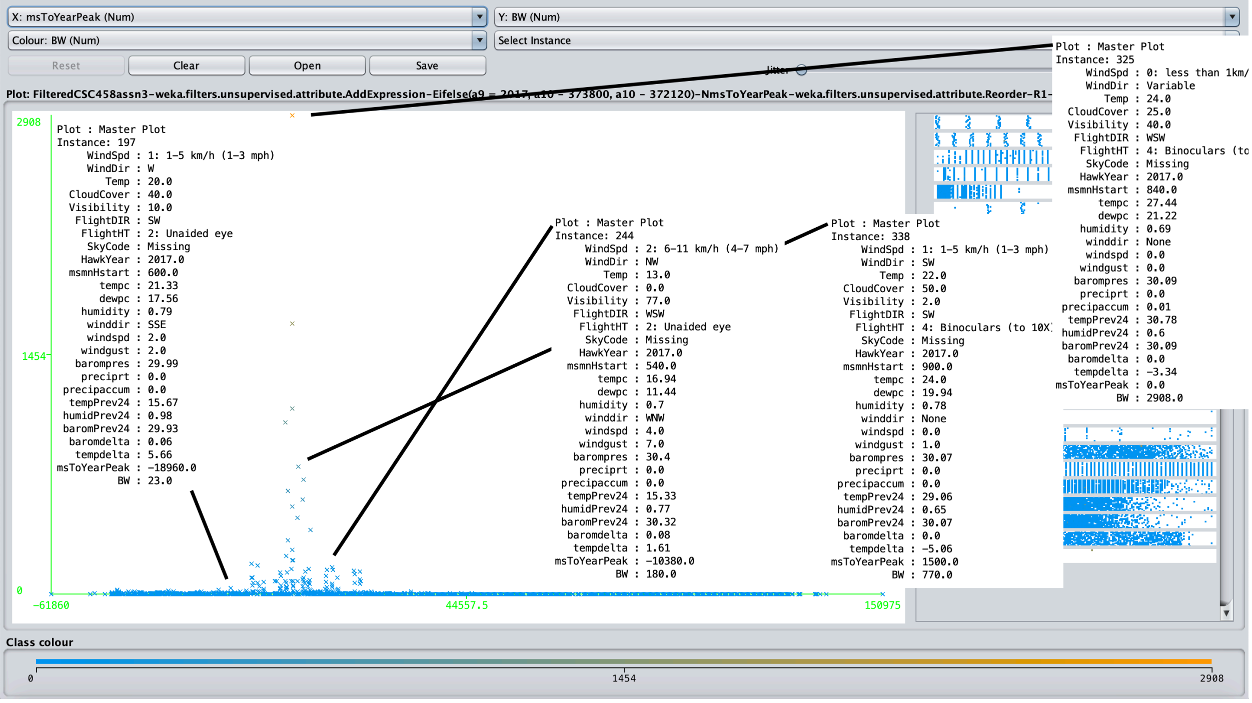
STEP2. Apply the **unsupervised -> attribute -> Reorder** filter to place BW last without disturbing the relative order of the other attributes. Remove **ALL** of the msny* prefixed attributes because they are redundant with **msToYearPeak**. Our day-of-year time base **msToYearPeak** measures the observation time's minutes relative to the peak BW hour. That helps to normalize time across years. KEEP msmnHstart, but remove ALL OF THE OTHER msmn* prefixed attributes, because they are redundant to some degree with msmnHstart. Keep **msToYearPeak**, of course. Remove SunMinutes because the length-of-daylight correlates with the day-of-year. There would be aliasing (roughly equal SunMinutes) between days before the start of summer, which is the longest day of the year, and corresponding days after start of summer, but since these counts begin in August, SunMinutes correlates exactly with the date, so we Remove SunMinutes. There should be 26 attributes and 2255 instances at this point, with BW last and msToYearPeak just before it. Temporarily Remove attribute HawkYear all by itself (do not Remove it with any other attribute with it – you need to UNDO this removal shortly).

Q9: Run LinearRegression and M5P on this reduced attribute, and copy & paste these values below. Also paste M5P's entire decision tree, and its **LM num: 1** linear regression formula; do not paste the other LM formulas. How do the error measures compare with LinearRegression from Q2 and M5P from Q7? How do LinearRegression versus M5P error measures within Q9 compare for dealing with average BW counts versus outliers? We will discuss the M5P decision tree when I go over the assignment's solution.

```
LinearRegression for Q9:
Correlation coefficient             n.n
Mean absolute error                 n.n
Root mean squared error              n.n
Relative absolute error              n      %
Root relative squared error          n      %
Total Number of Instances             2255

M5P for Q9:
Number of Rules : N
Correlation coefficient             n.n
Mean absolute error                 n.n
Root mean squared error              n.n
Relative absolute error              n      %
Root relative squared error          n      %
Total Number of Instances             2255

M5 pruned model tree:
(using smoothed linear models)
PASTE THE ENTIRE TREE

LM num: 1
BW =  PASTE FORMULA LM1
```

Q10: Execute UNDO once to restore only the **HawkYear** attribute, undoing its Removal. After consulting Figure 3, write an AddExpression using nested ifelse() and value comparisons on BW to create 7 distinct value ranges for BW as follows. Name this derived attribute **BWbins**. **SAVE** this ARFF 27-attribute data as **HawkData20172018.arff**. TURN this file into me by placing it in the handout directory before running **make turnitin** after you have completed the project. This is the only ARFF file you need to turn in. **Q11**. Also, paste your AddExpression for **BWbins** into README.txt. You should get a BWbins distribution that looks like Figure 4. You should also inspect the BWbins (Y) to msToYearPeak(X) visualization to see that this custom discretization looks correct.

BW range	BWbins value
0	0
1	1
2	2
<30	3
<200	4
<1000	5
Else (>= 1000)	6


X: msToYearPeak (Num)

Y: BW (Num)

Colour: BW (Num)

Select Instance

Plot: FilteredCSC458assn3-weka.filters.unsupervised.attribute.AddExpression-Eifelse(a9 = 2017, a10 - 373800, a10 - 372120)-NmsToYearPeak-weka.filters.unsupervised.attribute.Reorder-R1-

```
Plot : Master Plot
Instance: 197
     WindSpd : 1: 1-5 km/h (1-3 mph)
     WindDir : W
        Temp : 20.0
  CloudCover : 40.0
  Visibility : 10.0
   FlightDIR : SW
    FlightHT : 2: Unaided eye
     SkyCode : Missing
    HawkYear : 2017.0
  msmnHstart : 600.0
       tempc : 21.33
       dewpc : 17.56
    humidity : 0.79
     winddir : SSE
     windspd : 2.0
    windgust : 2.0
   barompres : 29.99
    preciprt : 0.0
 precipaccum : 0.0
  tempPrev24 : 15.67
 humidPrev24 : 0.98
 baromPrev24 : 29.93
  baromdelta : 0.06
   tempdelta : 5.66
msToYearPeak : -18960.0
          BW : 23.0
```

```
Plot : Master Plot
Instance: 244
     WindSpd : 2: 6-11 km/h (4-7 mph)
     WindDir : NW
        Temp : 13.0
  CloudCover : 0.0
  Visibility : 77.0
   FlightDIR : WSW
    FlightHT : 2: Unaided eye
     SkyCode : Missing
    HawkYear : 2017.0
  msmnHstart : 540.0
       tempc : 16.94
       dewpc : 11.44
    humidity : 0.7
     winddir : WNW
     windspd : 4.0
    windgust : 7.0
   barompres : 30.4
    preciprt : 0.0
 precipaccum : 0.0
  tempPrev24 : 15.33
 humidPrev24 : 0.77
 baromPrev24 : 30.32
  baromdelta : 0.08
   tempdelta : 1.61
msToYearPeak : -10380.0
          BW : 180.0
```

```
Plot : Master Plot
Instance: 338
     WindSpd : 1: 1-5 km/h (1-3 mph)
     WindDir : SW
        Temp : 22.0
  CloudCover : 50.0
  Visibility : 2.0
   FlightDIR : SW
    FlightHT : 4: Binoculars (to 10X
     SkyCode : Missing
    HawkYear : 2017.0
  msmnHstart : 900.0
       tempc : 24.0
       dewpc : 19.94
    humidity : 0.78
     winddir : None
     windspd : 0.0
    windgust : 1.0
   barompres : 30.07
    preciprt : 0.0
 precipaccum : 0.0
  tempPrev24 : 29.06
 humidPrev24 : 0.65
 baromPrev24 : 30.07
  baromdelta : 0.0
   tempdelta : -5.06
msToYearPeak : 1500.0
          BW : 770.0
```

```
Plot : Master Plot
Instance: 325
     WindSpd : 0: less than 1km/
     WindDir : Variable
        Temp : 24.0
  CloudCover : 25.0
  Visibility : 40.0
   FlightDIR : WSW
    FlightHT : 4: Binoculars (to
     SkyCode : Missing
    HawkYear : 2017.0
  msmnHstart : 840.0
       tempc : 27.44
       dewpc : 21.22
    humidity : 0.69
     winddir : None
     windspd : 0.0
    windgust : 0.0
   barompres : 30.09
    preciprt : 0.0
 precipaccum : 0.01
  tempPrev24 : 30.78
 humidPrev24 : 0.6
 baromPrev24 : 30.09
  baromdelta : 0.0
   tempdelta : -3.34
msToYearPeak : 0.0
          BW : 2908.0
```

Figure 3: Regions of BW custom discretization via AddExpression: 0, 1, 2, <30, <200, <1000, >= 1000

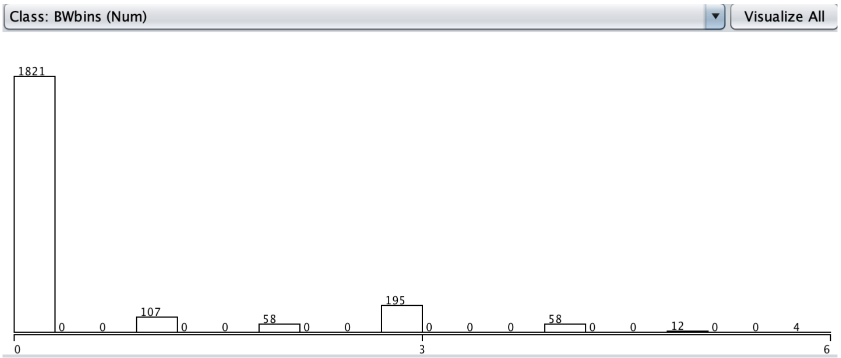


Figure 4: BWbins distribution

Q12: Remove **BW** by itself, since **BWbins** is now the target attribute, and they are non-linearly redundant. Then, temporarily Remove attribute **HawkYear** all by itself (do not Remove it with any other attribute with it – you need to UNDO this removal shortly). Run both LinearRegression and M5P against this dataset with BWbins as the target attribute, and paste indicated results below. How do you account for the changes in Correlation Coefficient and the error measures going from Q9 to Q12? (Note that **Mean absolute error** and **Root mean squared error** are in units of our 0-6 bin scale, not in BW counts, so the **Relative absolute error** and **Root relative squared error** measures are probably more useful.) Discuss the changes from Q9 in **Root relative squared error** brought about by the compression of the outlying peaks in bin 6.

LinearRegression for Q12:
Correlation coefficient n.n
~~Mean absolute error n.n~~
~~Root mean squared error n.n~~
Relative absolute error n %
Root relative squared error n %
Total Number of Instances 2255

M5P for Q12:
Number of Rules : N
Correlation coefficient n.n
~~Mean absolute error n.n~~
~~Root mean squared error n.n~~
Relative absolute error n %
Root relative squared error n %
Total Number of Instances 2255

STEP3: The second-last stage of this assignment is to use these BWbins data from 2017 as training data, and the 2018 data as test data. UNDO once to restore the **HawkYear** attribute. Take a note in the lower right of the Preprocess window how many 2017 instances there are, and how many 2018 instances there are. Apply **unsupervised -> instance -> RemoveWithValues** to remove all 2018 instances. Check to make sure that there remain the correct number of values, then save this data as TRAIN2017.arff. Execute UNDO once to restore the 2018 instances, then apply **unsupervised -> instance -> RemoveWithValues** to remove all 2017 instances. Check to make sure that there remain the correct number of values, then save this data as TEST2018.arff. Re-load (Open file) TRAIN2017.arff. Remove attribute **HawkYear** again. (Do not turn in any TRAIN*.arff or TEST*.arff files.)

Q13: Run both **LinearRegression** and **M5P** against this dataset, using 10-fold cross validation (default) with BWbins as the target attribute, and paste indicated results below. How do Correlation Coefficient and the error measure compare to those of Q12 for **LinearRegression** and **M5P**?

LinearRegression for Q13:
Correlation coefficient n.n
Mean absolute error n.n
Root mean squared error n.n
Relative absolute error n %
Root relative squared error n %
Total Number of Instances 1141

M5P for Q13:
Number of Rules : N
Correlation coefficient n.n
Mean absolute error n.n
Root mean squared error n.n
Relative absolute error n %

```
Root relative squared error      n    %
Total Number of Instances         1141
```

Q14: Run both **LinearRegression** and **M5P** against this dataset, using this 2017 data with HawkYear still removed as the training dataset, and TEST2018.arff as the external supplied test dataset, and paste indicated results below. How do Correlation Coefficient and the error measure compare to those of Q13 for **LinearRegression** and **M5P**? A degradation of more than 10% (.10) of Correlation Coefficient indicates over-fitting to the 2017 training data. Is there overfitting?

```
LinearRegression for Q14:
Correlation coefficient         n.n
Mean absolute error             n.n
Root mean squared error          n.n
Relative absolute error         n    %
Root relative squared error      n    %
Total Number of Instances         1114
```

```
M5P for Q14:
Number of Rules : N
Correlation coefficient         n.n
Mean absolute error             n.n
Root mean squared error          n.n
Relative absolute error         n    %
Root relative squared error      n    %
Total Number of Instances         1114
```

Q15: Load your saved file **HawkData20172018.arff**, and **Remove HawkYear** and **BW**, retaining BWbins as the target attribute out of 25 attributes total. Run **unsupervised -> instance -> Randomize** one time to shuffle (stratify) the 2017 and 2018 instances together. Apply **unsupervised -> instance -> RemovePercentage** with a default value of 50%, and note the number of instance that remain. SAVE this dataset as **TRAINHALF.arff**. Execute UNDO one time to restore the total 2255 instances, then run **instance -> RemovePercentage AFTER** setting **invertSelection** to **true** while leaving the percentage at 50%. SAVE this dataset as **TESTHALF.arff**. **Load TRAINHALF.arff** as the training set, then test it using **TESTHALF.arff** as the supplied external test dataset (not cross validation). Run **LinearRegression** and **M5P** and compare these Q15 results with Q14. Also, compare these Q15 results with Q12. What accounts for improvements in Q15 over Q14, given the fact that both use external test datasets of about the same size? What change do you see in over-fitting in going from Q14 to Q15, and why has that change occurred?

```
LinearRegression for Q15:
Correlation coefficient         n.n
Mean absolute error             n.n
Root mean squared error          n.n
Relative absolute error         n    %
Root relative squared error      n    %
Total Number of Instances         1128
```

```
M5P for Q15:
Number of Rules : N
Correlation coefficient         n.n
Mean absolute error             n.n
```

```
Root mean squared error              n.n
Relative absolute error             n      %
Root relative squared error          n      %
Total Number of Instances             1128
```

When you have completed all of your work and double-checked the assignment requirements, make sure that both **HawkData20172018.arff** saved in a previous step, and your **README.txt** that answers Q1 through Q15, are sitting in your **csc458fall2019assn3**/ directory, then run **make turnitin** by the due date. Late assignments lose 10% per day late, and I will not accept an assignment after I go over its solution in class.