Analysis & Prediction of Road Crash for Allegheny County

Tzupin Kuo, Samantha Lipsky

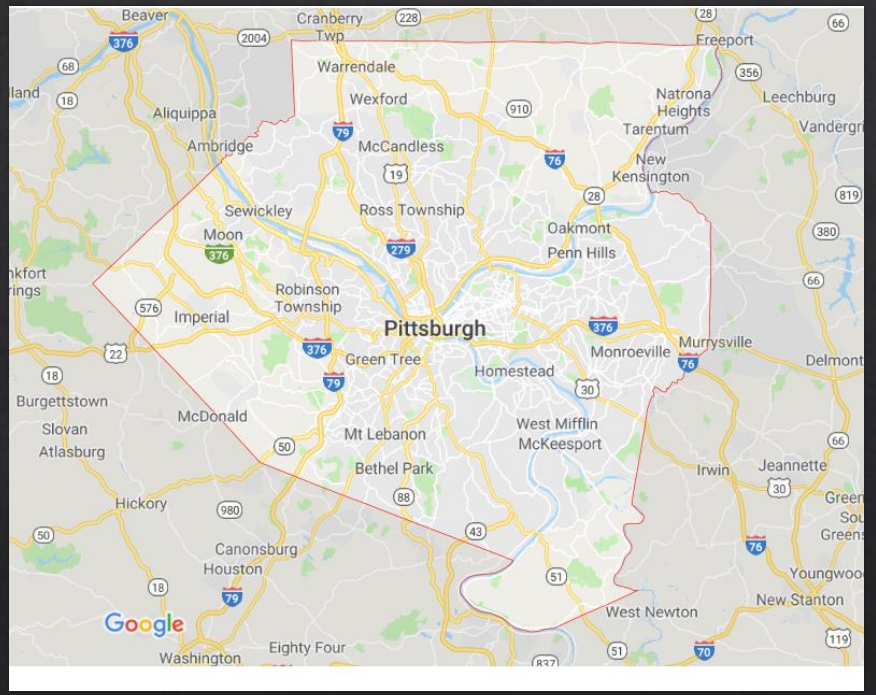
Data Mining Goal

1. Predict the injury severity into two groups:
   (No injury and Injury)
2. Compare the performance between 20 models.

The purpose of this study is to develop a model to predict the injury severity outcomes for vulnerable road users involving car crashes using different classifier algorithms. The study also attempts to identify factors that are important in making an injury severity difference and to explore the impact of such explanatory variables.

Dataset Description

- Fatalities resulting from vehicle crashes is one of the main causes of death in the United States.
- Dataset from Western Pennsylvania Regional Data Center contains locations and information about every crash incident reported to the police in Allegheny County from 2004 to 2018.
- Fields include injury severity, fatalities, information about the vehicles involved, location information, and factors that may have contributed to the crash.



Dataset Basics

Source: Western Pennsylvania Regional Data Center [Link]

Data is provided by Pennsylvania Department of Transportation (PennDOT).

The dataset consists of 69 categorical attributes.

The dataset consists of 11,994 tuples.

The "MAX_SEVERITY_LEVEL" attribute is used as the class label.

Original Dataset

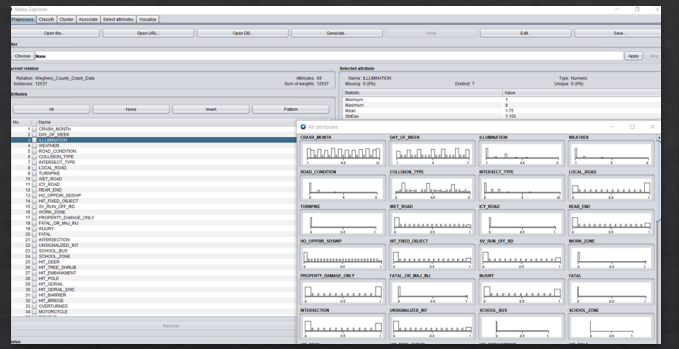
Range of Categories

		3 - Tuesday 4 - Wednesday 5 - Thursday 6 - Friday 7 - Saturday	
ILLUMINATION	Categorical	1 – Daylight 2 – Dark – no street lights 3 – Dark – street lights 4 – Dusk 5 – Dawn 6 – Dark – unknown roadway lighting 8 – Other 9 – Unknown (expired)	Code that defines lighting at c scene
WEATHER		1 – No adverse conditions 2 – Rain 3 – Sleet (hail)	Code for the weather type at t crash

Binary

SCHOOL_ZONE	Categorical	0 = No, 1 = Yes	School Zone Indicator
HIT_DEER	Categorical	0 = No, 1 = Yes	Hit Deer Indicator
HIT_TREE_SHRUB	Categorical	0 = No, 1 = Yes	Hit Tree or Shrub Indicator
HIT_EMBANKMENT	Categorical	0 = No, 1 = Yes	Hit Embankment Indicator
HIT_POLE	Categorical	0 = No, 1 = Yes	Hit Pole Indicator
HIT_GDRAIL	Categorical	0 = No, 1 = Yes	Hit Guide Rail Indicator
HIT_GDRAIL_END	Categorical	0 = No, 1 = Yes	Hit Guide Rail End Indicator
HIT_BARRIER	Categorical	0 = No, 1 = Yes	Hit Barrier Indicator
HIT_BRIDGE	Categorical	0 = No, 1 = Yes	Hit Bridge Indicator
OVERTURNED	Categorical	0 = No, 1 = Yes	Overturned Vehicle Indicator
MOTORCYCLE	Categorical	0 = No, 1 = Yes	Motorcycle Indicator
BICYCLE	Categorical	0 = No, 1 = Yes	Bicycle Indicator
HVY_TRUCK_RELATED	Categorical	0 = No, 1 = Yes	Heavy Truck Related Indicato
VEHICLE_FAILURE	Categorical	0 = No, 1 = Yes	Vehicle Failure Indicator
TRAIN_TROLLEY	Categorical	0 = No, 1 = Yes	Train or Trolley Indicator
PHANTOM_VEHICLE	Categorical	0 = No, 1 = Yes	Phantom Vehicle Indicator
ALCOHOL_RELATED	Categorical	0 = No, 1 = Yes	Alcohol Related Indicator
DRINKING_DRIVER	Categorical	0 = No, 1 = Yes	Drinking Driver Indicator
UNDERAGE_DRNK_DRV	Categorical	0 = No, 1 = Yes	Under Age drinking driver Ind

Overall



Needed to translate into nominal values

Preprocessing

Data Cleaning

- Converting numeric values to nominal values
- We used R to convert all the numeric values to nominal values. For example, we convert 0 to “NO”, 1 to “YES”
- Missing values
  - Removed Tuples with Illumination = other or unknown (values 8,9)
  - Removed Tuples with Road Conditions = Other
  - Removed Tuples with weather codes = 6,7,8,9 (no information on these codes)
  - Removed Tuples with Collision = (8,9) Other/Unknown
  - Removed Tuples with Intersection = 10 (Other)

Class Attribute

- Class Attribute from the original dataset : Max Severity Level

  0 - Not injured
  1 - Killed
  2 - Major injury
  3 - Moderate injury
  4 - Minor injury
  8 - Injury/ Unknown Severity
  9 – Unknown

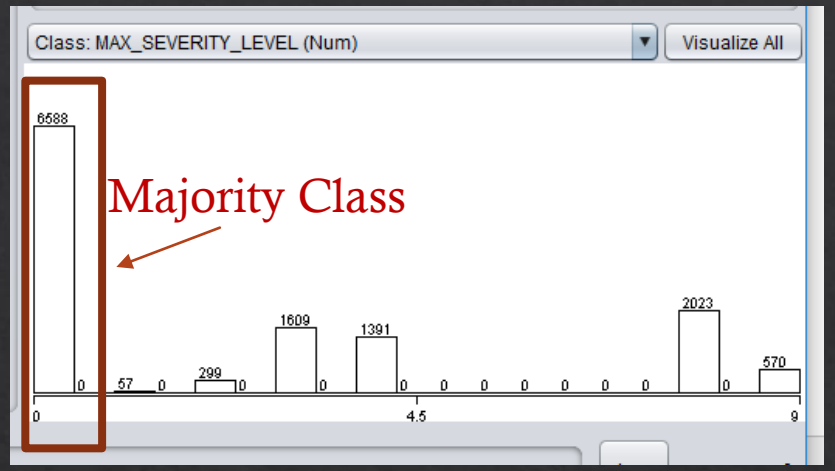
  - Deleted unknown classes (8&9) and simplified to binary classes.
- Class Attrib from modified dataset : Max Severity Level

  No injury
  Injury : {Minor injury, Moderate injury, Major injury, Killed}

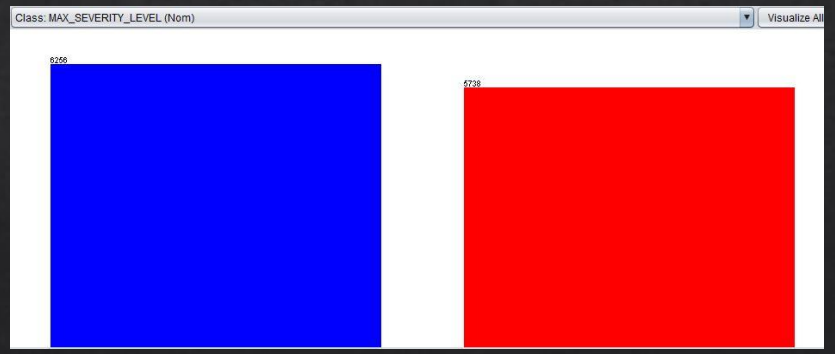
Class Attribute Distribution

- After removing Class 8 & 9, still distribution of classes was still skewed left
- To avoid bias towards a majority class (Class 1 = No Injury), we made two classes
  - Class 1 = No Injury (6,256 tuples)
  - Class 0 = Injury (5,738 tuples)

Original Distribution

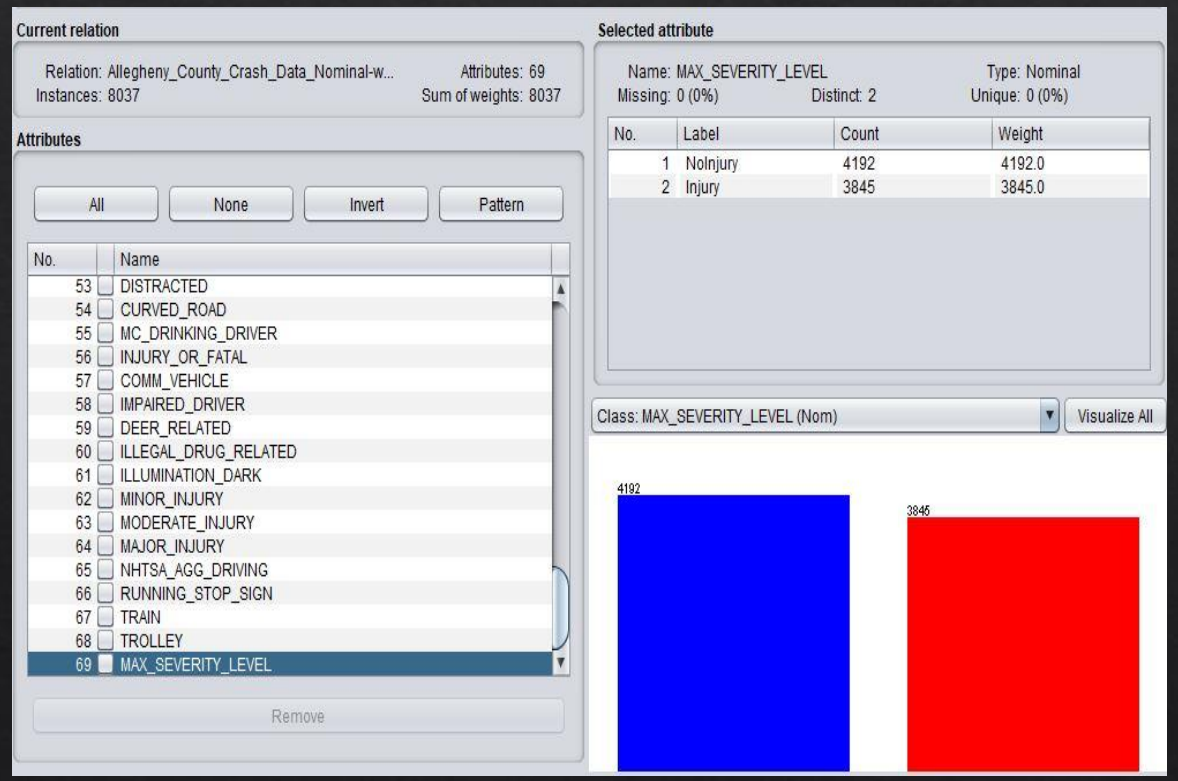


New Distribution

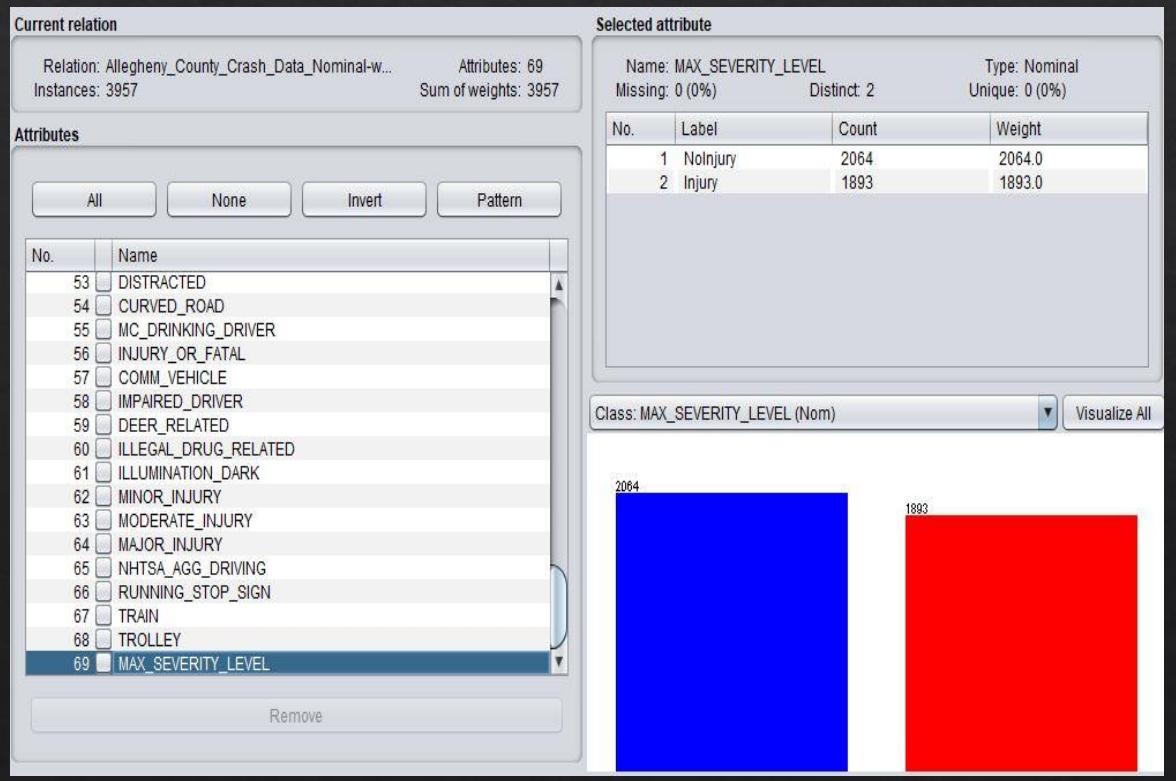


Splitting of Training/Testing Data

- Image from Weka on Train



- Image from Weka on Test

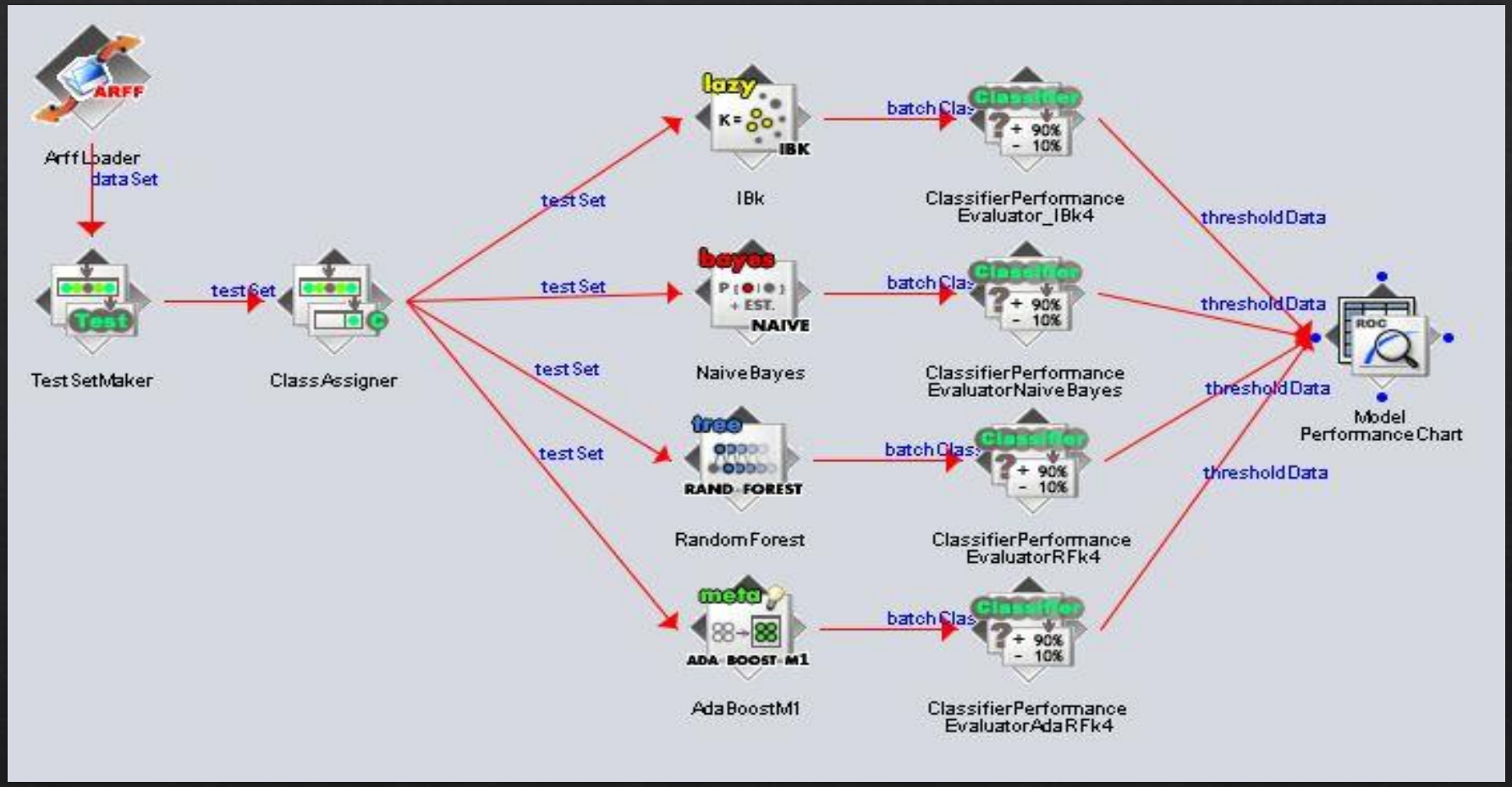


Data Mining Algorithms

- Tools used: Weka and R

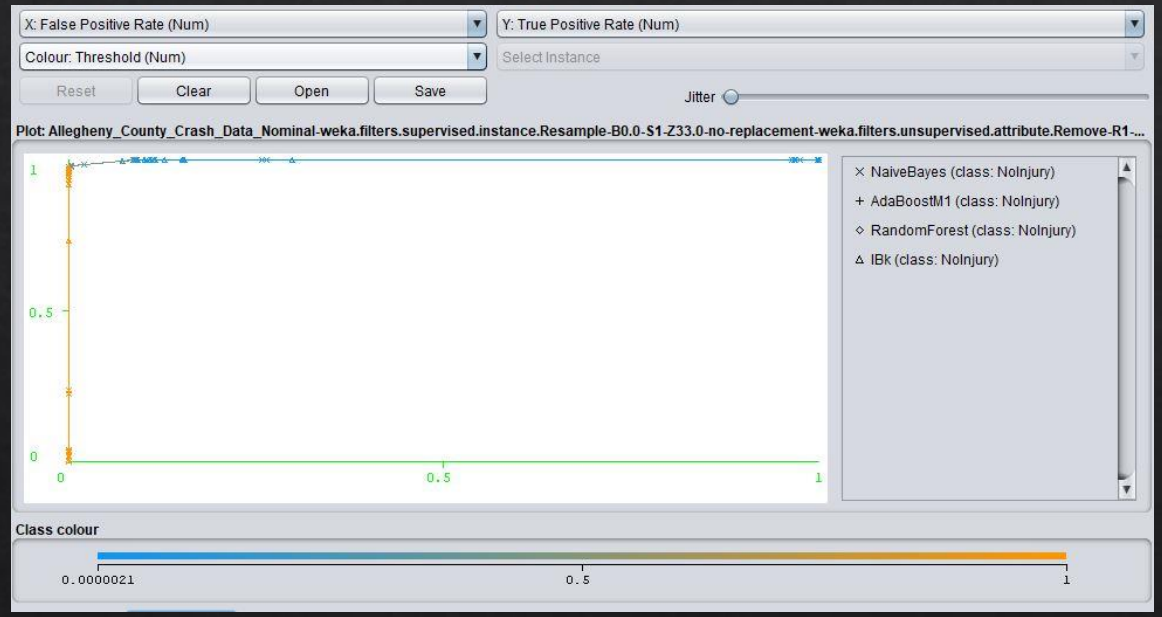
Feature selection / Classifier	CfsSubset	CorrAttr	InfoGainAttr	GainRatioAttr	ClassifierAttrJ48
IBk	Model 1	Model 2	Model 3	Model 4	Model 5
Naïve Bayes	Model 6	Model 7	Model 8	Model 9	Model 10
Random Forest	Model 11	Model 12	Model 13	Model 14	Model 15
AdaBoostM1+ Random Forest	Model 16	Model 17	Model 18	Model 19	Model 20

KF

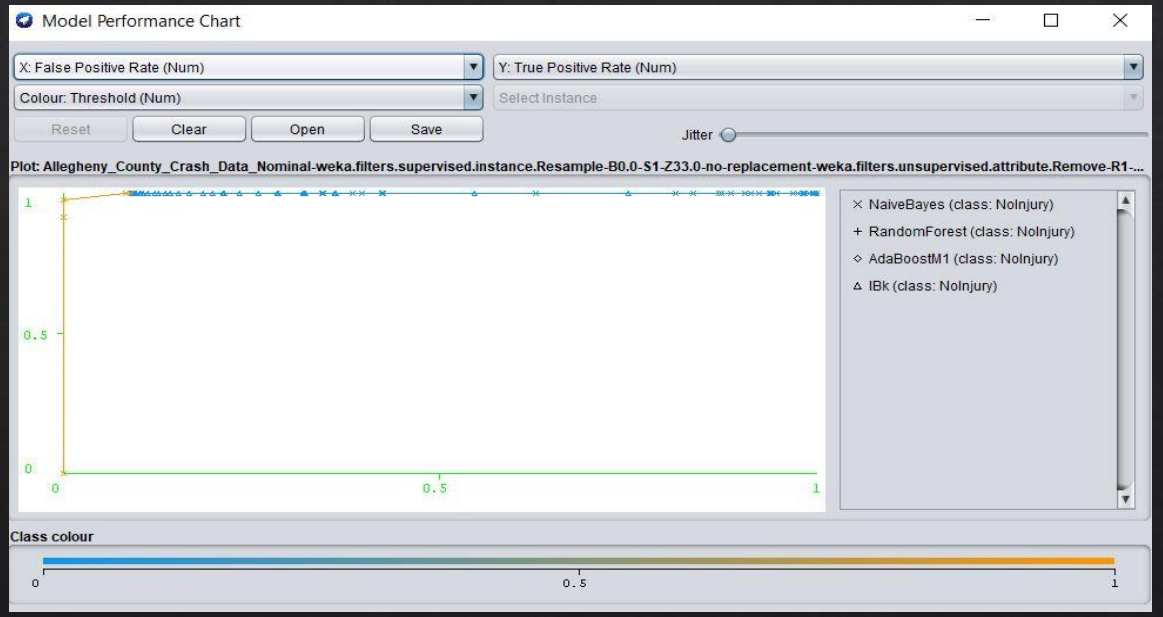

ARFF
Arff Loader
data Set
Test
Test SetMaker
test Set
C
ClassAssigner
test Set
lazy
K=
IBK
IBk
test Set
bayes
P (|)
+ EST.
NAIVE
Naive Bayes
test Set
free
RAND FOREST
Random Forest
test Set
meta
ADA BOOST M1
AdaBoostM1
batch Clas
Classifier
+ 90%
- 10%
ClassifierPerformance Evaluator_IBk4
batch Clas
Classifier
+ 90%
- 10%
ClassifierPerformance EvaluatorNaive Bayes
batch Clas
Classifier
+ 90%
- 10%
ClassifierPerformance EvaluatorRFk4
batch Clas
Classifier
+ 90%
- 10%
ClassifierPerformance EvaluatorAdaRFk4
threshold Data
threshold Data
threshold Data
threshold Data
ROC
Model Performance Chart

ROCs

CfsSubset

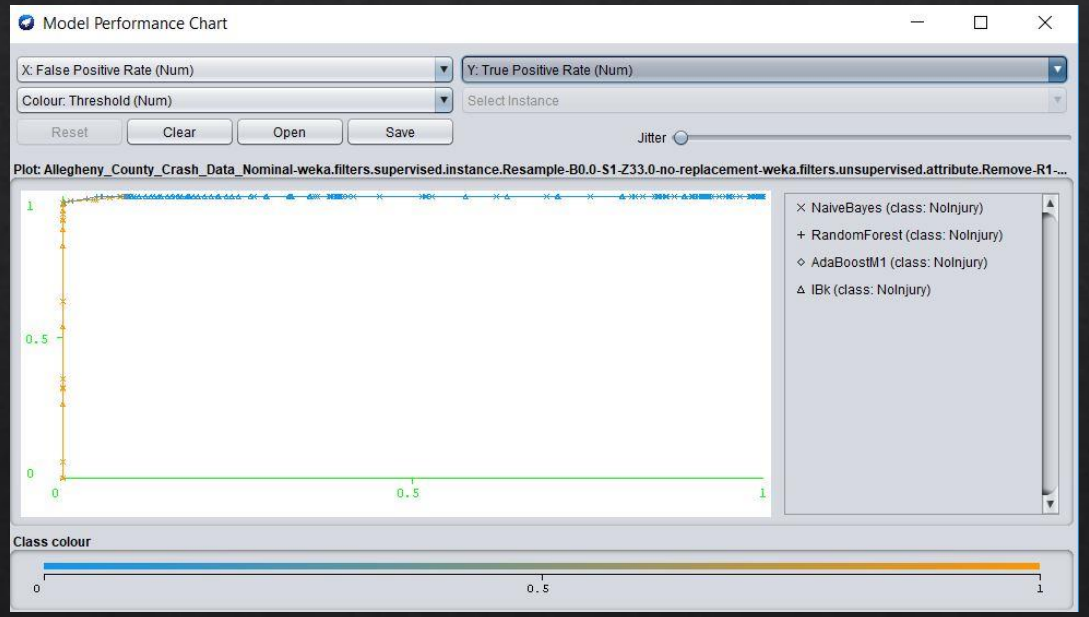


CorrAttribute

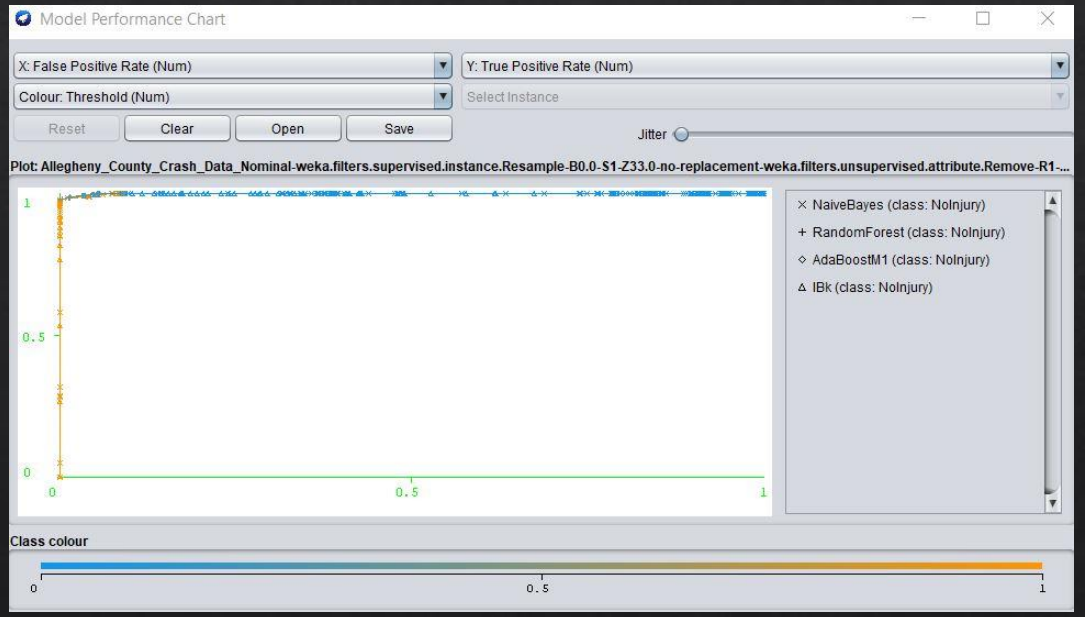


ROCs (Cont.)

InfoGainAttribute

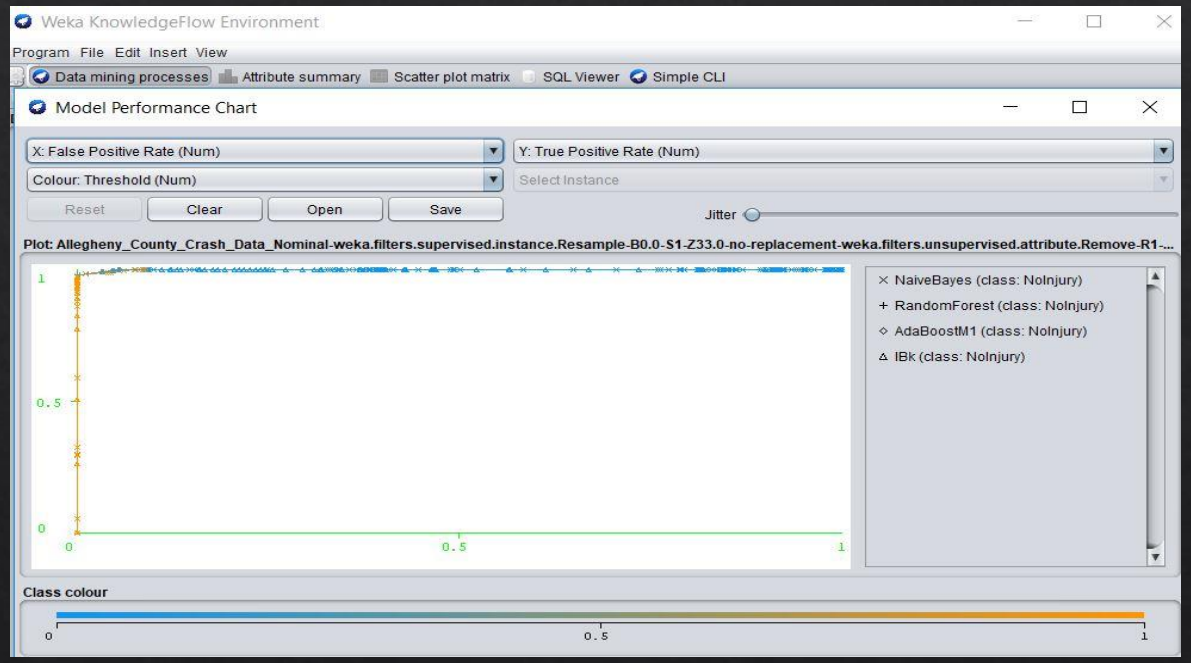


InfoRatioAttribute



ROCs (Cont.)

ClassifierAttribute



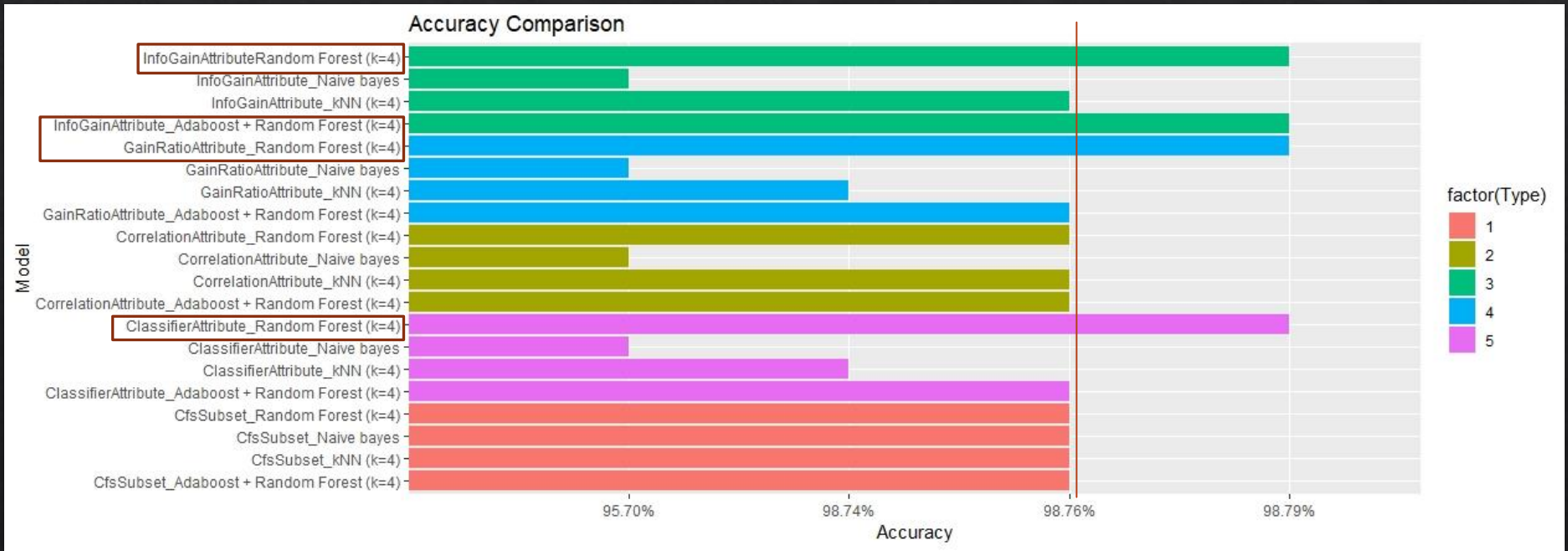
Model Performance Measures

- All measures are essentially the same for Random Forest classifiers, regardless of Attribute Selection.
- Naïve Bayes had a higher False Positive Rate than the other classifiers.

Model	Accuracy	ROC Area	F-Measure	TPR	FPR
CfsSubset_kNN (k=4)	98.76%	0.999	0.988	0.988	0.011
CfsSubset_Naive bayes	98.76%	0.999	0.988	0.988	0.011
CfsSubset_Random Forest (k=4)	98.76%	0.999	0.988	0.988	0.011
CfsSubset_Adaboost + Random Forest (k=4)	98.76%	0.999	0.988	0.988	0.011
CorrelationAttribute_kNN (k=4)	98.76%	0.999	0.988	0.988	0.011
CorrelationAttribute_Naive bayes	95.70%	0.999	0.957	0.957	0.047
CorrelationAttribute_Random Forest (k=4)	98.76%	0.999	0.988	0.988	0.011
CorrelationAttribute_Adaboost + Random Forest (k=4)	98.76%	0.999	0.988	0.988	0.011
InfoGainAttribute_kNN (k=4)	98.76%	0.999	0.988	0.988	0.011
InfoGainAttribute_Naive bayes	95.70%	0.999	0.957	0.957	0.047
InfoGainAttributeRandom Forest (k=4)	98.79%	0.999	0.988	0.988	0.011
InfoGainAttribute_Adaboost + Random Forest (k=4)	98.79%	0.999	0.988	0.988	0.011
GainRatioAttribute_kNN (k=4)	98.74%	0.999	0.987	0.987	0.012
GainRatioAttribute_Naive bayes	95.70%	0.999	0.957	0.957	0.047
GainRatioAttribute_Random Forest (k=4)	98.79%	0.999	0.988	0.988	0.011
GainRatioAttribute_Adaboost + Random Forest (k=4)	98.76%	0.999	0.988	0.988	0.011
ClassifierAttribute_kNN (k=4)	98.74%	0.999	0.987	0.987	0.012
ClassifierAttribute_Naive bayes	95.70%	0.999	0.957	0.957	0.047
ClassifierAttribute_Random Forest (k=4)	98.79%	0.999	0.988	0.988	0.011
ClassifierAttribute_Adaboost + Random Forest (k=4)	98.76%	0.999	0.988	0.988	0.011

Model Accuracy Comparison

- Feature Selection + Random Forest Classifier provides highest accuracy (98.79%)
- Which Feature Selection is best…?



Best Feature Selection Algorithm
InfoGain: Less Redundant (?)

- Varied multiple settings in the Random Forest classifier, none resulted in change in accuracy or F-measure

  k = 0, 2, 3, 4, 16
  smaller batch size (n = 25 versus 100)
  breaks ties randomly
  number of iterations from 100 to 1000

- Evaluated best model based on the information that would be given with the attributes provided in each selection algorithm.

- Classification Attribute showed "UNBELTED" contributing 0.02 to the model; whereas, InfoGain did not have this attribute in its model. One thought is that "UNBELTED" is actually a redundant attribute.
  * Correlation Analysis with null invariant measures.

GainRatio, Random Forest, k = 4
weka.classifiers.trees.RandomForest -K 4 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities

Attribute importance based on average impurity decrease (and number of nodes using that attribute)

0.46 (438) PROPERTY_DAMAGE_ONLY
0.37 (149) INJURY
0.37 (137) INJURY_OR_FATAL
0.17 (165) COLLISION_TYPE
0.11 (54) MINOR_INJURY
0.08 (50) MODERATE_INJURY
0.04 (291) OVERTURNED
0.04 (40) FATAL
0.03 (68) FATAL_OR_MAJ_INJ
0.03 (150) MOTORCYCLE
0.02 (10) MC_DRINKING_DRIVER
0.02 (454) UNBELTED
0.01 (19) MAJOR_INJURY
0.01 (33) PEDESTRIAN
0.01 (23) BICYCLE
0 (13) TRAIN_TROLLEY
0 (5) TRAIN

ClassifierAtt(J48), Random Forest, k =4
weka.classifiers.trees.RandomForest -K 4 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities

Attribute importance based on average impurity decrease (and number of nodes using that attribute)

0.47 (298) PROPERTY_DAMAGE_ONLY
0.34 (91) INJURY
0.25 (140) INJURY_OR_FATAL
0.13 (132) COLLISION_TYPE
0.09 (23) MINOR_INJURY
0.06 (37) MODERATE_INJURY
0.03 (45) FATAL_OR_MAJ_INJ
0.02 (132) MOTORCYCLE
0.02 (9) PEDESTRIAN
0.02 (413) UNBELTED
0 (2) MAJOR_INJURY

InfoGain, Random Forest, k = 4
weka.classifiers.trees.RandomForest -K 4 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities

Attribute importance based on average impurity decrease (and number of nodes using that attribute)

0.46 (232) PROPERTY_DAMAGE_ONLY
0.33 (83) INJURY
0.3 (118) INJURY_OR_FATAL
0.09 (32) MODERATE_INJURY
0.09 (104) COLLISION_TYPE
0.09 (22) MINOR_INJURY
0.03 (58) FATAL_OR_MAJ_INJ
0.02 (125) MOTORCYCLE
0.01 (13) PEDESTRIAN
0 (1) MAJOR_INJURY

GO DEEPER on **InfoGain** Dataset

Association Analysis – find out the relationship between attributes for practical decision making

We implemented association analysis using Apriori algorithm on ***InfoGainAttributeEvalTrain.arff*** to find out top 10 rules between attributes: (settings: minConfidence = 0.9, delta:0.05, upperBoundMinSupport=1, lowerBoundMinSupport=0.1)

1. FATAL_OR_MAJ_INJ=NO → MAJOR_INJURY=NO	<conf:(1)> lift:(1.03) lev:(0.02) [191] conv:(191.31)
2. FATAL_OR_MAJ_INJ=NO, MOTORCYCLE=NO → MAJOR_INJURY= NO	<conf:(1)> lift:(1.03) lev:(0.02) [187] conv:(187.88)
3. MOTORCYCLE=NO, MAJOR_INJURY=NO → FATAL_OR_MAJ_INJ=NO	<conf:(1)> lift:(1.03) lev:(0.02) [192] conv:(7.17)
4. MAJOR_INJURY=NO → FATAL_OR_MAJ_INJ=NO	<conf:(1)> lift:(1.03) lev:(0.02) [191] conv:(6.29)
5. FATAL_OR_MAJ_INJ=NO 7805 → MOTORCYCLE=NO	<conf:(0.98)> lift:(1) lev:(0) [28] conv:(1.19)
6. FATAL_OR_MAJ_INJ=NO MAJOR_INJURY=NO → MOTORCYCLE=NO	<conf:(0.98)> lift:(1) lev:(0) [28] conv:(1.19)
7. FATAL_OR_MAJ_INJ=NO → MOTORCYCLE=NO, MAJOR_INJURY=NO	<conf:(0.98)> lift:(1.03) lev:(0.02) [192] conv:(2.36)
8. MAJOR_INJURY=NO → MOTORCYCLE=NO	<conf:(0.98)> lift:(1) lev:(0) [23] conv:(1.16)
9. MOTORCYCLE=NO → MAJOR_INJURY=NO	<conf:(0.98)> lift:(1) lev:(0) [23] conv:(1.13)
10. MAJOR_INJURY=NO → FATAL_OR_MAJ_INJ=NO, MOTORCYCLE=NO	<conf:(0.98)> lift:(1.03) lev:(0.02) [187] conv:(2.06)

Discussion

- InfoGain with Random Forest classification (k=4) = best model; high accuracy and F-score for classifying No Injury versus Injury.
- Other models achieved the same results, InfoGain was able to use one less attribute than Classification Attribute (J48).
  - Regression analysis could be done to further analyze the two models, if there is numeric representation of this data.
- CfsSubset + Random Forest used fewer attributes (4), but had had lower accuracy and F-score than InfoGain + Random Forest. Performance alone is not based on the fewest number of attributes, but also the accuracy or error rate (1-accuracy).
- If we used more classes, this dataset would have resulted in an unbalanced distribution of classes (skewed right). The trade-off in having a robust model (No Injury versus Injury) was that we lost learning about the granularity within the types of injuries.
- Future work:
  1. Keep skewed class distribution and look at other performance measures, such as specificity.
  2. Resample the dataset using under-sampling technique to balance three classes: 1. no injury, 2. minor injury/moderate injury, 3. major injury/death. Try to dig in more detailed information on these different levels of injuries.

Is this dataset practically usable?

- We can remove the non-injured class. Focus on minor versus major injuries/death.
- Knowing how to predict major injuries, could result in better preventative actions or local ordinances to reinforce preventative actions

Conclusion

- Overall, we were able to predict non-injuries from road accidents from the Allegheny dataset (2004-2018) with high accuracy using InfoGain Attribute Selection and Random Forest. The FPR (FP/N) was also very low (0.011), thus also leading us to conclude that we have a model that will not predict non-injuries as injuries.