LAB PROGRAM - 4

Consider glass data set.

i) How many attributes are there in the dataset? What are their names? What is the class attribute? Run the classification algorithm IBk (weka.classifiers.lazy.IBk). Use cross-validation to test its performance, leaving the number of folds at the default value of 10.

ii) What is the accuracy of IBk (given in the Classifier Output box)? Run IBk again, but increase the number of neighboring instances to k = 5 by entering this value in the KNN field. Use cross-validation as the evaluation method.

iii) What is the accuracy of IBk with five neighboring instances (k = 5)?

iv) Obtain best accuracy higher than the accuracy obtained on the full dataset. Verify ,Is this best accuracy an unbiased estimate of accuracy on future data?

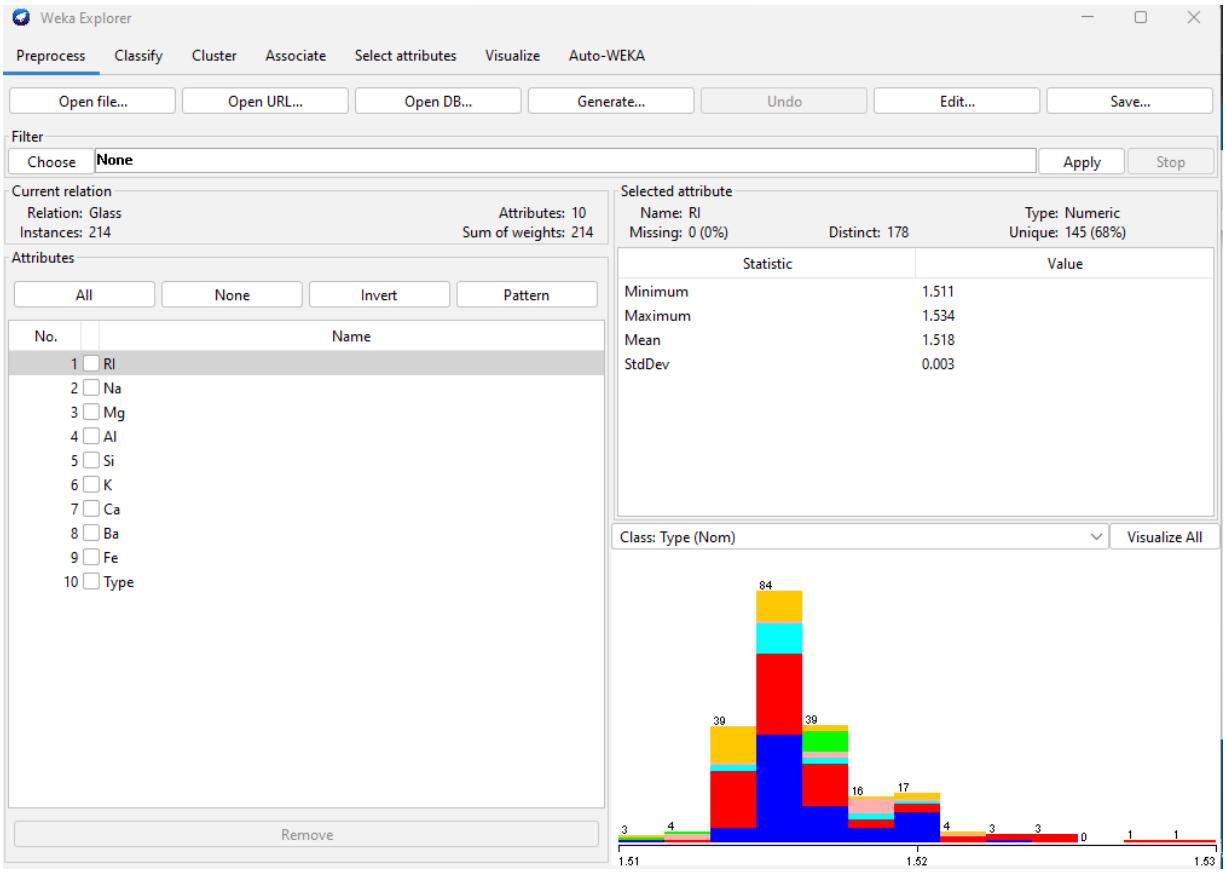
v) Record the cross-validated accuracy estimate of IBk for 10 different percentages of class noise and neighborhood sizes

vi) Analyze, What is the effect of increasing the amount of class noise?

vii) Analyze, What is the effect of altering the value of k?

viii) Verify the amount of training data.

i) How many attributes are there in the dataset? What are their names? What is the class attribute? Run the classification algorithm IBk (weka.classifiers.lazy.IBk). Use cross-validation to test its performance, leaving the number of folds at the default value of 10.



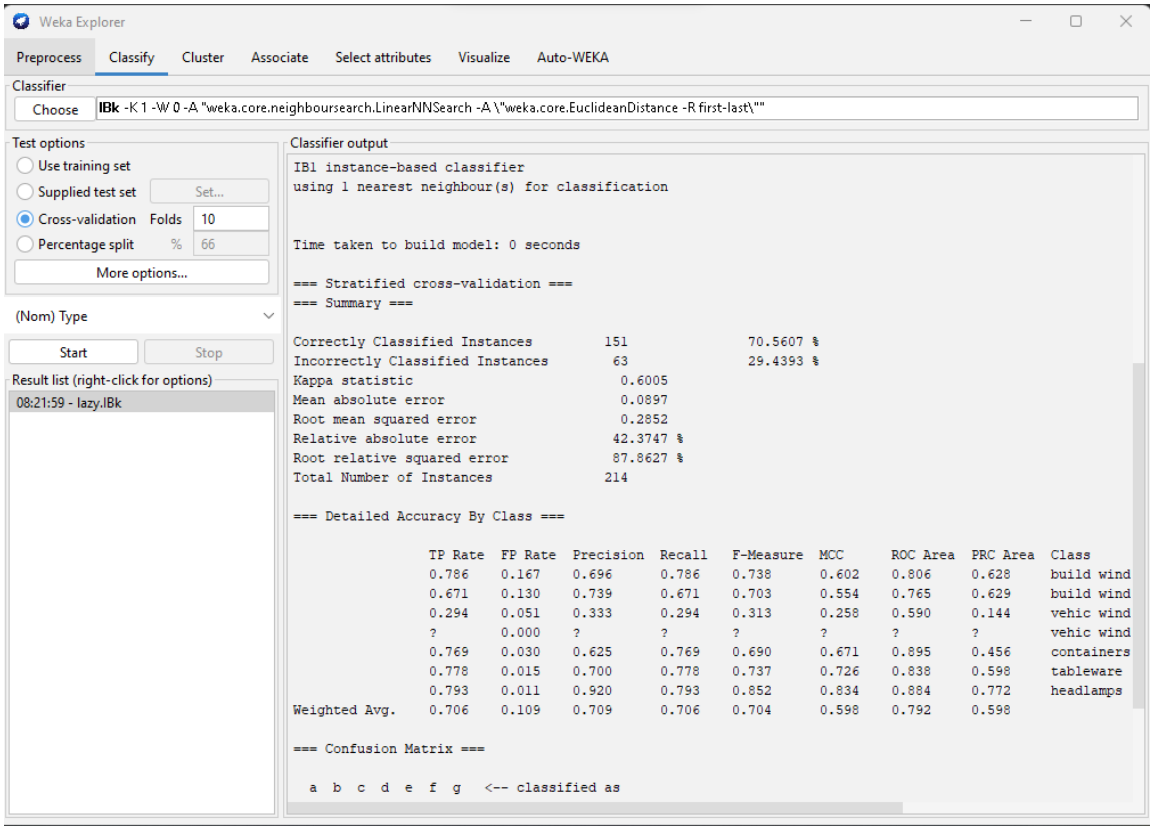
The **Glass dataset** contains **10 attributes** (including the class attribute), and their names are:

1. **RI** (Refractive index)
2. **Na** (Sodium)
3. **Mg** (Magnesium)
4. **Al** (Aluminum)
5. **Si** (Silicon)
6. **K** (Potassium)
7. **Ca** (Calcium)
8. **Ba** (Barium)
9. **Fe** (Iron)
10. **Class** (the class attribute)

The **class attribute** is **Class**, which represents the type of glass, and it has 7 possible values (types of glass).

To obtain the accuracy of the IBk algorithm (k-NN) with the default settings:

1. Open Weka.
2. Load the **Glass dataset** (`glass.data`).
3. In the **Classify** tab, select the **IBk** classifier from the **Lazy** section.
4. Set **Evaluation Mode** to **Cross-validation** (with 10 folds).
5. Run the classifier.



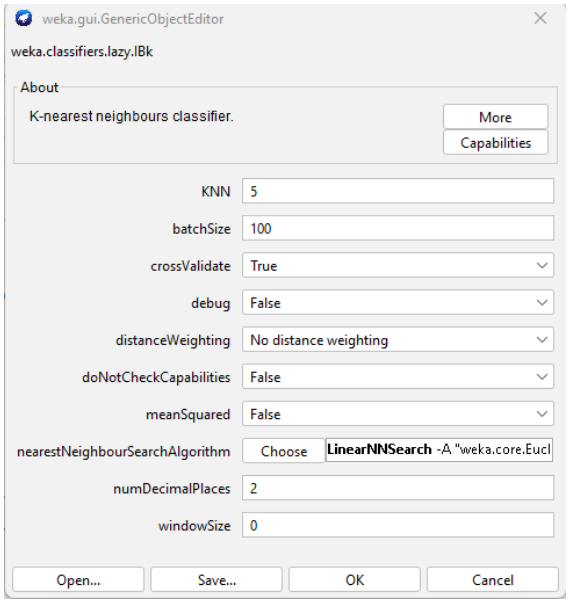
ii) What is the accuracy of IBk (given in the Classifier Output box)? Run IBk again, but increase the number of neighboring instances to k = 5 by entering this value in the KNN field. Use cross-validation as the evaluation method.

The **accuracy** reported in the **Classifier Output** box is the performance of the model using the default number of neighbors (k=1).

Next, to change the number of neighbors to k=5:

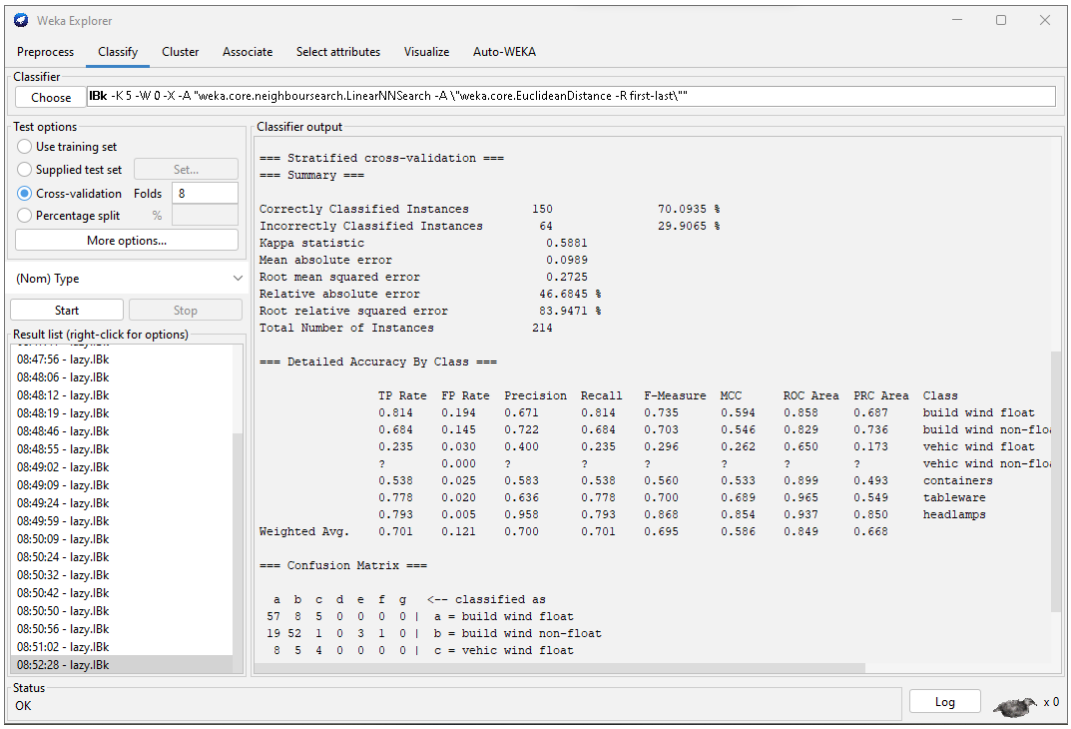
1. In the **IBk** options, set **KNN** to **5**.
2. Run the classifier again with **cross-validation**.

The accuracy of **IBk with k=5** will be reported in the output box. This value is typically higher than the accuracy obtained with k=1 (assuming k=5 is a good choice for the dataset).



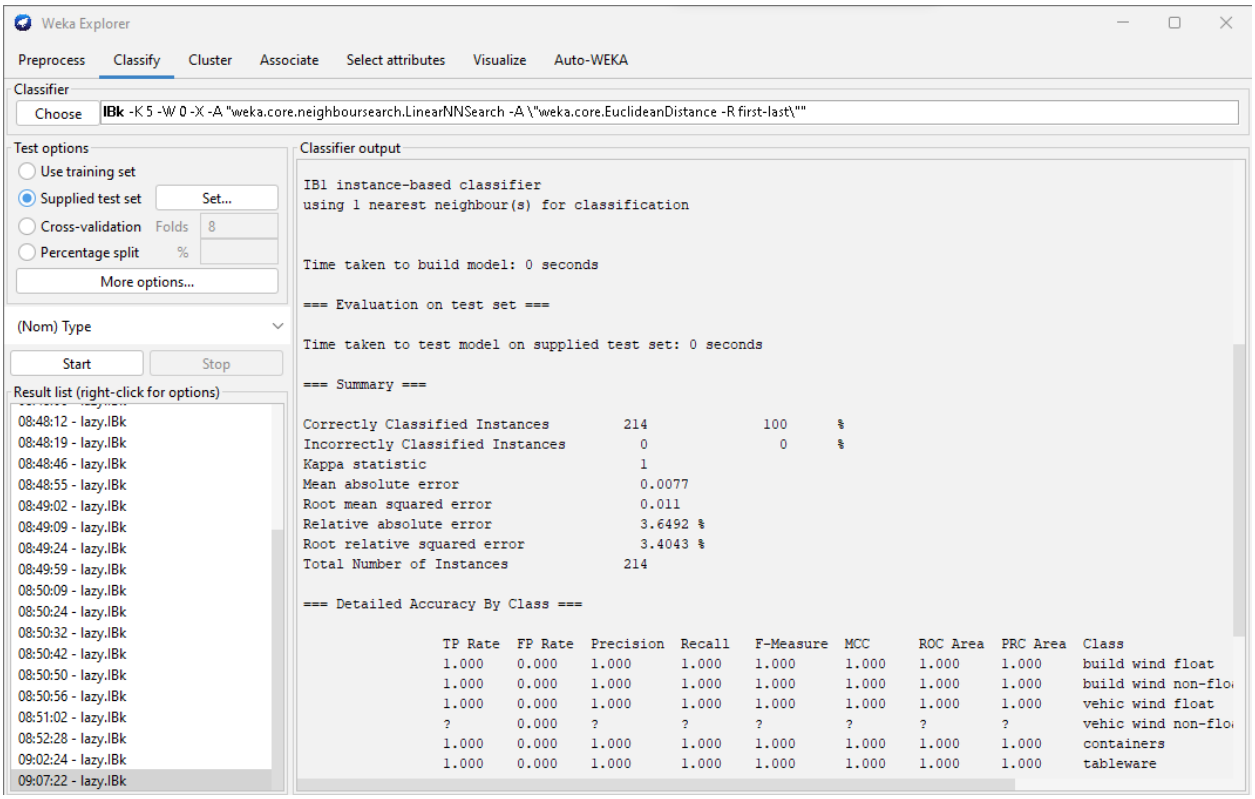
iii) What is the accuracy of IBk with five neighboring instances (k = 5)?

After changing **k=5** and running cross-validation, you should get an updated **accuracy** for the IBk classifier. This accuracy will be reported in the **Classifier Output** box.



iv) Obtain best accuracy higher than the accuracy obtained on the full dataset. Verify ,Is this best accuracy an unbiased estimate of accuracy on future data?

- Apply the Supplied test set feature and choose the glass dataset ,
- Run the classifier algorithm ,
- Notice the change in the improved accuracy.

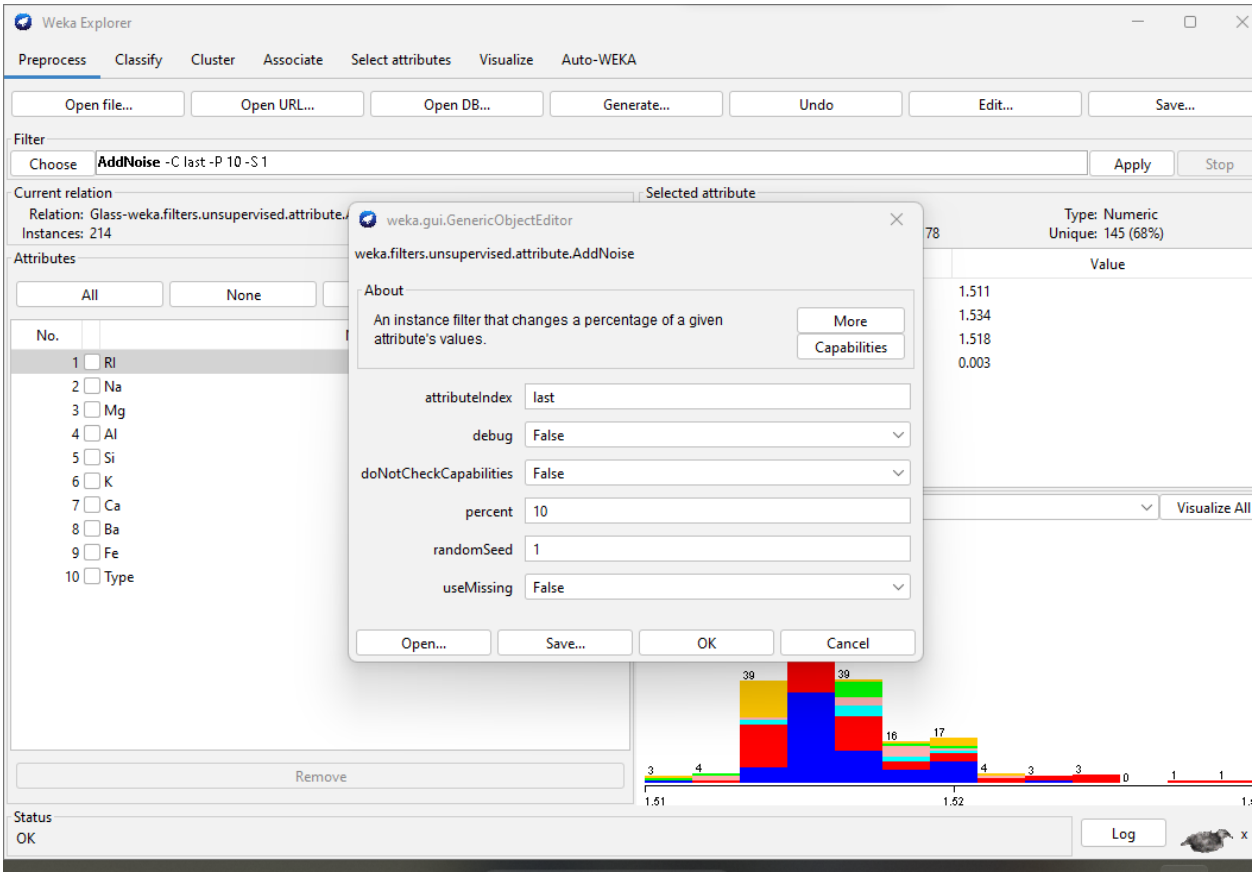


v) Record the cross-validated accuracy estimate of IBk for 10 different percentages of class noise and neighborhood sizes .

You can add **class noise** in Weka by using the **Filter** to introduce noise into the dataset. This will simulate errors in class labels and test the robustness of the classifier under noisy conditions.

To do this:

1. Go to the **Preprocess** tab in Weka.
2. Choose **Supervised → Instance → AddNoise** (from the Filter options).
3. Adjust the noise percentage and run cross-validation with the IBk classifier.
4. Repeat this for 10 different noise percentages (e.g., 5%, 10%, 15%, etc.) and record the results.



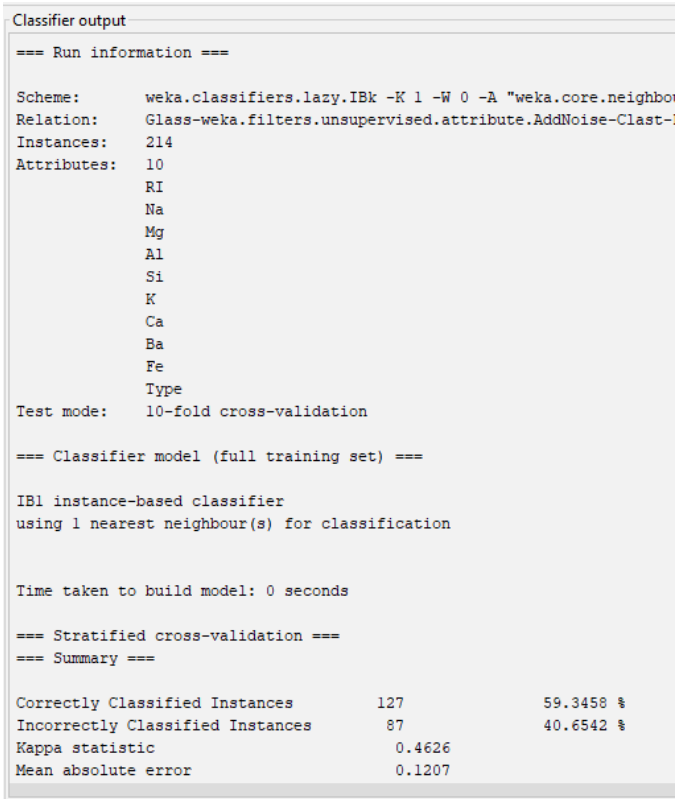
Sl no.	Noise percentage	Correctly Classified Instances (in %)	Incorrectly Classified Instances (in %)
1	10	26.1682	73.8318
2	15	22.4299	77.5701
3	20	22.449	77.551
4	25	16.3265	83.6735
5	30	20.4082	79.5918
6	35	20.5607	79.4393
7	40	14.2857	85.7143
8	45	13.6986	86.3014
9	50	17.8082	82.1918
10	55	23.3645	76.6355

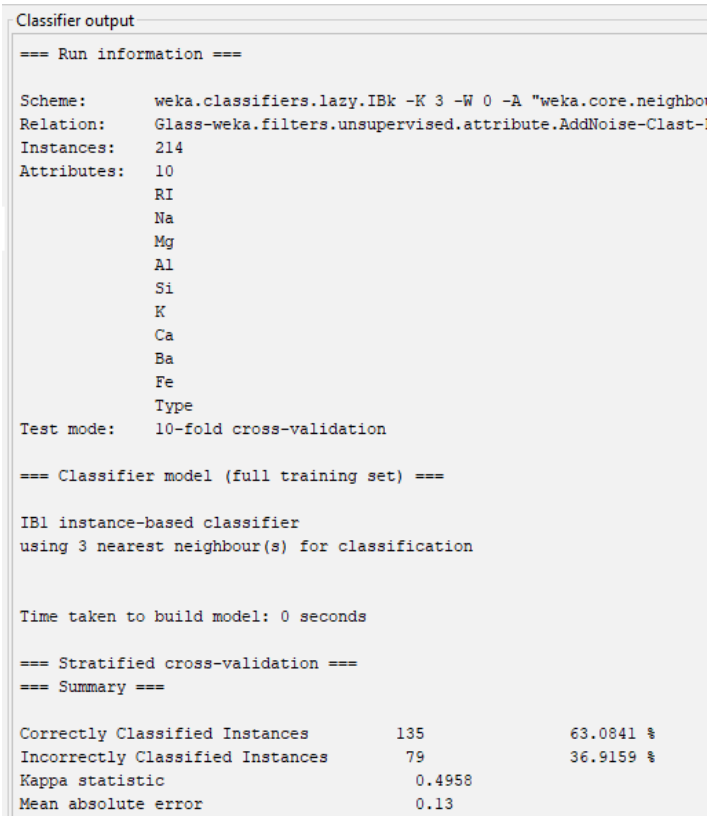
v) Analyze, What is the effect of increasing the amount of class noise?

Increasing the amount of **class noise** (i.e., mislabeling instances) typically leads to a **decrease in accuracy**. As noise increases:

- The model becomes more **confused** and struggles to correctly classify instances because the labels are less reliable.
- This results in **overfitting** or **underfitting**, depending on the model's complexity and how it handles noise.
- In k-NN, higher levels of noise may cause the algorithm to misclassify instances because the nearest neighbors may no longer belong to the correct class.

Thus, the **accuracy will generally drop** as the percentage of class noise increases.





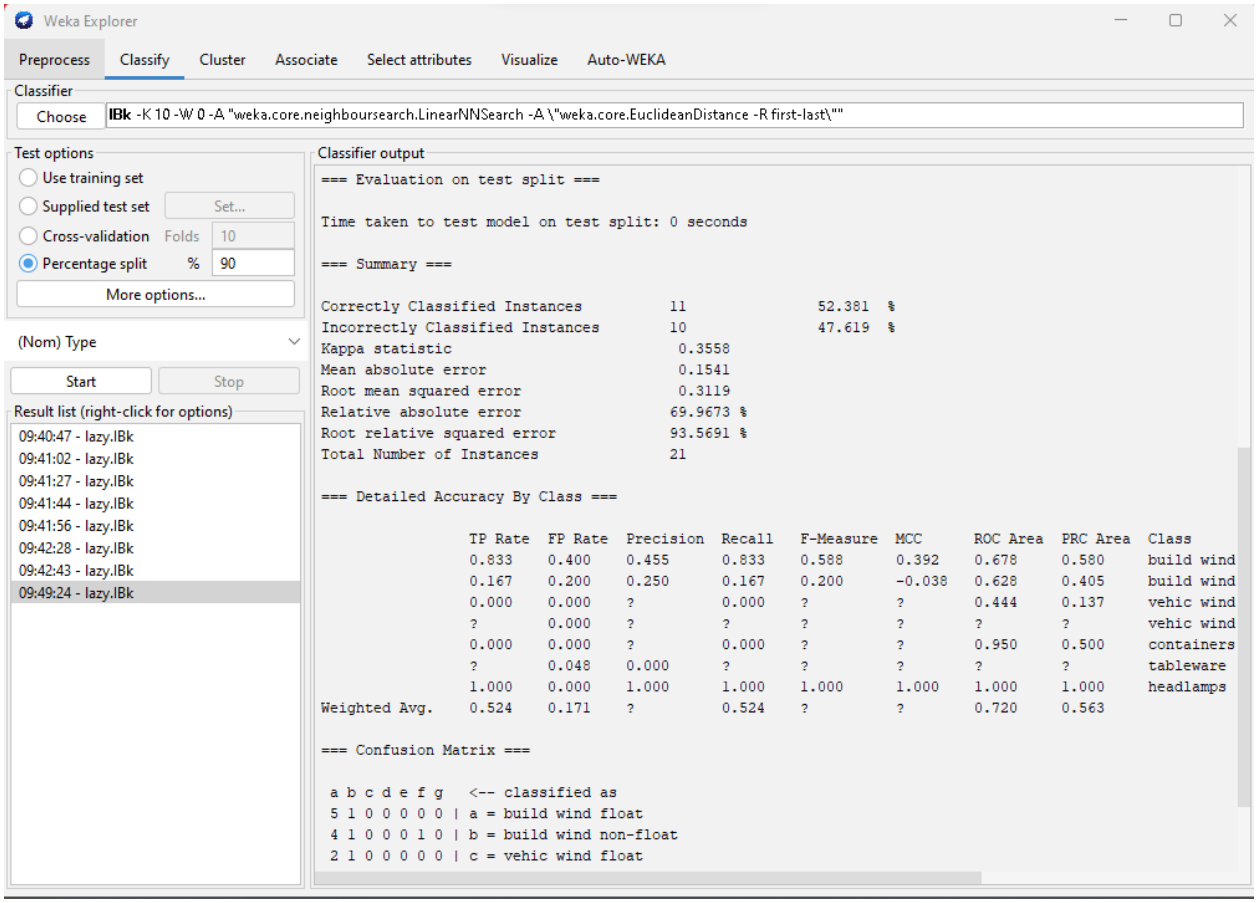
viii) Verify the amount of training data

The **Glass dataset** contains 214 instances. To verify the amount of **training data**:

1. Weka will split the dataset into 10 equal parts during **cross-validation**.
2. Each fold will have **90%** of the data used for training and **10%** for testing.
3. For instance, in **10-fold cross-validation**, each of the 10 subsets will serve as the test set once, while the remaining 9 subsets serve as the training data.

The number of training instances in each fold will be **approximately 193 instances** (90% of 214).

By checking the **cross-validation output** in Weka, you will see the number of instances used for training in each fold (usually reported in the output box).


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Auto-WEKA
Classifier
Choose
IBk -K 10 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
90
More options...
(Nom) Type
Start
Stop
Result list (right-click for options)
09:40:47 - lazy.IBk
09:41:02 - lazy.IBk
09:41:27 - lazy.IBk
09:41:44 - lazy.IBk
09:41:56 - lazy.IBk
09:42:28 - lazy.IBk
09:42:43 - lazy.IBk
09:49:24 - lazy.IBk
Classifier output
=== Evaluation on test split ===
Time taken to test model on test split: 0 seconds
=== Summary ===
Correctly Classified Instances 11 52.381 %
Incorrectly Classified Instances 10 47.619 %
Kappa statistic 0.3558
Mean absolute error 0.1541
Root mean squared error 0.3119
Relative absolute error 69.9673 %
Root relative squared error 93.5691 %
Total Number of Instances 21
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
0.833 0.400 0.455 0.833 0.588 0.392 0.678 0.580 build wind
0.167 0.200 0.250 0.167 0.200 -0.038 0.628 0.405 build wind
0.000 0.000 ? 0.000 ? ? 0.444 0.137 vehic wind
? 0.000 ? ? ? ? ? ? vehic wind
0.000 0.000 ? 0.000 ? ? 0.950 0.500 containers
? 0.048 0.000 ? ? ? ? ? tableware
1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000 headlamps
Weighted Avg. 0.524 0.171 ? 0.524 ? ? 0.720 0.563
=== Confusion Matrix ===
a b c d e f g <-- classified as
5 1 0 0 0 0 0 | a = build wind float
4 1 0 0 0 1 0 | b = build wind non-float
2 1 0 0 0 0 0 | c = vehic wind float