Exploration of the dataset:

The dataset consists of 785 attributes out of which 784 attributes denote the pixel intensities of an 28x28pi image and, one attribute named ‘class’, which denotes the label of the digit, the image falls under.

The below histogram(fig-1) shows the number of records(images) according to their labels.

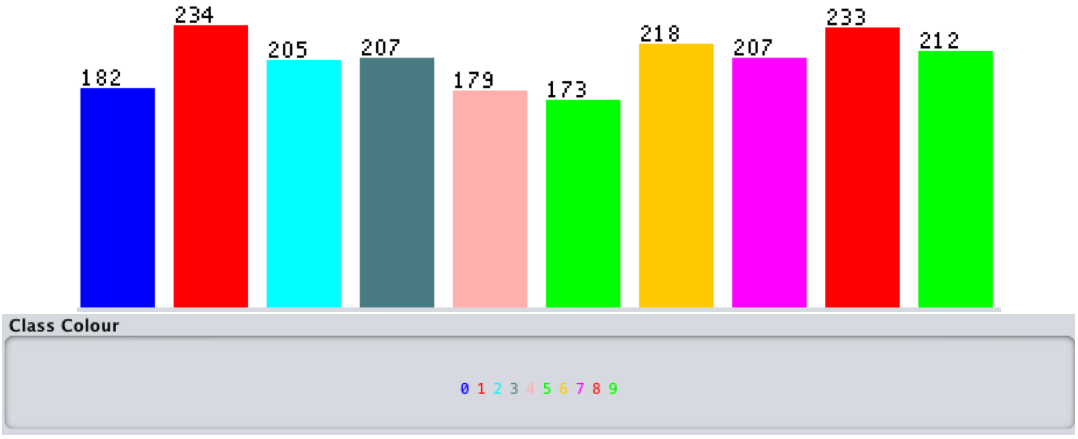

fig-1: Histogram depicting the number of labeled images.

The range of values the pixels take range from 0-255.

The intensity of the pixel must be higher only at the places where the white ink has been used. Whereas, when some scatter plots between class vs pixels is observed, some points tend to stand out. These points are highlighted in gray(fig-2).

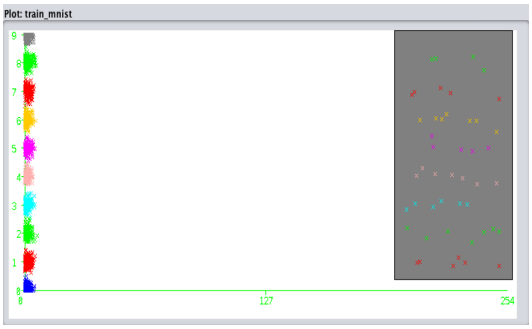

fig-2: plot depicting noisy images.

When the pixel intensities at these points are observed they tend to have a lot of noise. i.e the pixels have higher intensities in the whole image. This can be observed in fig-3.

This was the main observation made during the exploration and visualisations.

```
Plot : Master Plot
Instance: 979
  pixel_1_1 : 248.0
  pixel_1_2 : 214.0
  pixel_1_3 : 208.0
  pixel_1_4 : 204.0
  pixel_1_5 : 229.0
  pixel_1_6 : 225.0
  pixel_1_7 : 208.0
  pixel_1_8 : 235.0
  pixel_1_9 : 236.0
 pixel_1_10 : 216.0
 pixel_1_11 : 216.0
 pixel_1_12 : 241.0
 pixel_1_13 : 216.0
 pixel_1_14 : 239.0
 pixel_1_15 : 246.0
 pixel_1_16 : 226.0
```

fig-3: Pixel intensities for noisy images

Results on Initial dataset:

Naive Bayes:

Correctly Classified Instances(accuracy)	945	46.0976 %
Incorrectly Classified Instances	1105	53.9024 %

```
=== Confusion Matrix ===

   a   b   c   d   e   f   g   h   i   j   <-- classified as
 179   0   2   0   0   1   0   0   0   0 |   a = 0
  85 135   4   0   2   0   1   0   4   3 |   b = 1
  69   1  73   4   8   3  12   1  31   3 |   c = 2
  95   1   2  75   5   0   3   3  18   5 |   d = 3
  85   0   4   0  23   1   1   4  22  39 |   e = 4
  91   0   3   2   4  13   1   0  53   6 |   f = 5
  61   3   2   0   5   2 135   0  10   0 |   g = 6
  39   0   0   0   5   0   0  79   7  77 |   h = 7
 108   4   3   3   1   3   0   2  97  12 |   i = 8
  43   0   2   0   6   0   0  13  12 136 |   j = 9
```

From the above confusion matrix, we can observe that a lot of instances have been predicted to be 0's. This could possibly be because of the noise that exists in the images. This noise might confuse the model from learning correctly. This can also be proved by taking a look at the confusion matrix. The model failed to classify the digits '0', '8' and '9' in most cases. This might be because they are very similar when hand written.

J48 Decision tree:

Correctly Classified Instances(accuracy)	1454	70.9268 %
Incorrectly Classified Instances	596	29.0732 %

```
=== Confusion Matrix ===

   a   b   c   d   e   f   g   h   i   j   <-- classified as
 158   1   5   2   1   6   2   2   4   1 |   a = 0
   0 202   7   2   3   3   4   6   6   1 |   b = 1
  11  10 131  12   9   6  12   2  10   2 |   c = 2
   5   9   7 130   8  13   2   6  19   8 |   d = 3
   7   5   9  10 113   6   3   8   7  11 |   e = 4
   6   4   9  34   1  89   5   4  15   6 |   f = 5
   5   4  13   2   9   4 172   2   5   2 |   g = 6
   3   1   3   6   7   3   2 164   4  14 |   h = 7
   6  11  25  13  12  14   4   2 140   6 |   i = 8
   1   5   5   8  16   6   0   7   9 155 |   j = 9
```

From the above confusion matrix, we can observe that the J48 model has performed far better than that of the Naive Bayes. This is probably because the decision trees must have formed the constraints(conditions) really good and must have succeeded in eliminating the noise to an extent.

Luckily, all most all the noisy images have high intensity in pixel number 1. No digit, no matter how oddly it is written, does utilise pixel_1. Hence images with higher intensities at pixel_1 are removed. This has been done by using the

'SubsetByExpression' filter in weka. The expression used was 'ATT1<100'. This has subsetted the data with pixel_1 intensity less than 100. 'Remove with values' has been used to remove this subsetted data from the original data. The remaining data now has 2000 instances.

Retraining and evaluating the model with the cleaned dataset:

Naive Bayes:

Correctly Classified Instances(accuracy)	1417	70.85 %
Incorrectly Classified Instances	583	29.15 %

```
=== Confusion Matrix ===

   a   b   c   d   e   f   g   h   i   j   <-- classified as
 144   0   7   1   2   2   3   1  20   2 |   a = 0
   0 207   2   1   1   0   5   1  10   1 |   b = 1
   2   0 136   6   3   3  22   0  24   2 |   c = 2
   1   3  14 119   4   2   6   4  34  14 |   d = 3
   2   0   8   1  63   4   6  11  20  57 |   e = 4
   3   2   8   9   2  63   6   5  64   6 |   f = 5
   0   3  13   0   0   2 190   0   3   0 |   g = 6
   1   1   0   1   9   2   0 170   2  16 |   h = 7
   4  16   6   5   3  13   1   5 156  20 |   i = 8
   2   1   0   0   8   1   0  22   6 169 |   j = 9
```

As can be observed from the above output, the accuracy of Naive Bayes has turned out to be 70.85% which is a vast difference when compared to its performance on the uncleaned dataset. Removal of these noisy images has hence been a very important step in our process of analysis. Also, if we look at the confusion matrix, the model had problems classifying the digit '8' properly.

J48 Decision tree:

Correctly Classified Instances (accuracy)	1459	72.95 %
Incorrectly Classified Instances	541	27.05 %

```
=== Confusion Matrix ===

   a   b   c   d   e   f   g   h   i   j   <-- classified as
 151   1  11   1   1   4   6   0   5   2 |   a = 0
   1 199   3   5   4   1   3   1   8   3 |   b = 1
   6   1 137  10   5  10   5   3  18   3 |   c = 2
   2   3   9 129   5  24   2   7  13   7 |   d = 3
   2   2   7   6 107   8   0  11   8  21 |   e = 4
   9   3   7  21   9  98   2   2  11   6 |   f = 5
   8   5   9   1   4   3 173   1   6   1 |   g = 6
   3   2   6   3   4   3   0 159   4  18 |   h = 7
   5  11  11  13   5  20   5   3 150   6 |   i = 8
   3   1   6   7  13   2   1  13   7 156 |   j = 9
```

As can be observed from the above accuracy metrics, the performance of J48 decision tree has slightly increased with the cleaned dataset. It has increased by 2.03% to be exact. Therefore, the removal of these noise images has evidently improved the performance of the J48 Decision tree too.

About 39 attributes have been checked and no difference has been observed in the histograms(fig-4). This is because, these attributes doesn't take any pixel intensities. This could be because the attributes are located in the corners of the picture. Whereas, the images in the dataset have been centred. So, we can remove this attributes. The 'RemoveUseless' filter has been used to perform this operation.

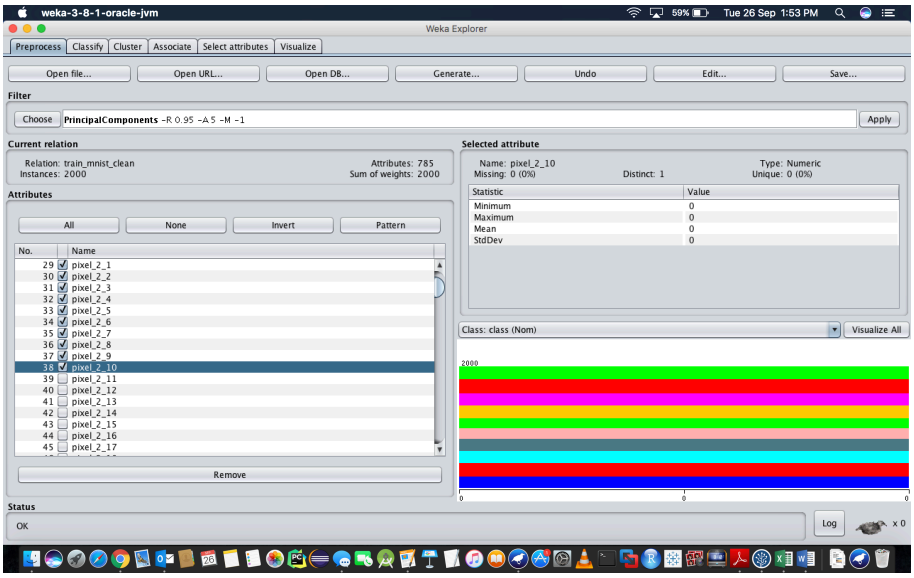

Fig-4: Histogram of useless attributes

After the attributes are removed, the dataset is left with 2000 instances and 640 attributes(144 attributes have been removed). Now, lets train and see the performance of the models on the reduced dataset.

Naive Bayes:

Correctly Classified Instances	1417	70.85 %
Incorrectly Classified Instances	583	29.15 %

J48 Decision Tree:

Correctly Classified Instances	1459	72.95 %
Incorrectly Classified Instances	541	27.05 %

As can be observed from the above values that, removing these attributes had zero impact on the model. They have the exact same accuracy as before. This proves that these attributes doesn't provide any information to our model and hence can be removed to decrease the complexity of the model classifier.

The next task is to use the attribute evaluator InfoGainAttributeEval to find the usefulness of the attributes in the dataset. This evaluator calculates the worth of the attributes by measuring the information gain with respect to the class.
The information gain is calculated by,

$$InfoGain(Class,Attribute) = H(Class) - H(Class \mid Attribute)$$

Where H represents the entropy of the element.

Lets look at the plots of Highest information gain attribute (vs) Lowest information gain attribute against class.

fig-5: Attribute with highest information gain

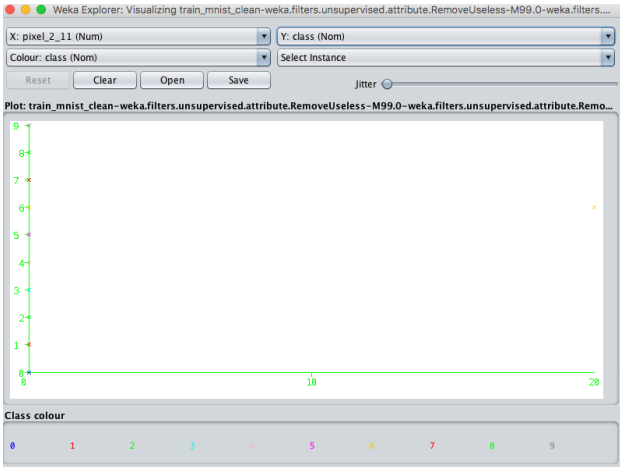

fig-6: Attribute with lowest information gain

The above scatter plots evidently show how the low information gain attribute carries almost zero information. Is it safe to remove these attributes could be subjective from one dataset to other dataset. But, in our case it should be very safe to perform these operation. It decreases the complexity on the model on could possibly increase the accuracy of classification. The attributes with this zero information gain have been removed using the 'Remove' filter.

Classifier's performance on the reduced dataset:

Naive Bayes:

Correctly Classified Instances(accuracy)	1431	71.55 %
Incorrectly Classified Instances	569	28.45 %

```
=== Confusion Matrix ===

   a   b   c   d   e   f   g   h   i   j   <-- classified as
 143   0   7   1   2   3   3   0  21   2 |   a = 0
   0 208   2   1   1   0   5   1   9   1 |   b = 1
   1   0 138   6   3   3  21   0  23   3 |   c = 2
   1   3  12 123   5   2   6   3  31  15 |   d = 3
   1   0   7   1  66   4   6  10  20  57 |   e = 4
   3   2   8   9   3  63   6   4  65   5 |   f = 5
   0   3  11   0   0   2 192   0   3   0 |   g = 6
   1   1   0   2   9   1   0 170   3  15 |   h = 7
   3  16   6   5   3  12   1   4 158  21 |   i = 8
   1   1   1   0   7   2   0  21   6 170 |   j = 9
```

J48 Decision Tree:

Correctly Classified Instances(accuracy)	1465	73.25 %

Incorrectly Classified Instances 535 26.75 %

```
=== Confusion Matrix ===

   a   b   c   d   e   f   g   h   i   j   <-- classified as
 151   1  10   3   1   3   4   0   7   2 |   a = 0
   1 197   4   5   2   2   5   1   8   3 |   b = 1
   7   5 137  11   5   8   7   4  12   2 |   c = 2
   3   4  10 130   5  24   2   7  10   6 |   d = 3
   3   1  10   3 108   7   1  10   3  26 |   e = 4
   7   3   7  17   9 103   2   1  12   7 |   f = 5
   6   4  12   1   3   5 173   1   5   1 |   g = 6
   3   2   5   6   4   3   0 158   3  18 |   h = 7
   4  11   8  15   5  21   2   3 155   5 |   i = 8
   4   1   4   7  17   2   1  12   8 153 |   j = 9
```

From the above results we can observe that the accuracy has significantly increased. This means the performance of the models has increased. This proves our point that removing the attributes with zero information gain has a positive effect.