The Simpler the Better

Valliappa Lakshmanan[1,2*]

Abstract

We built three different AI models to predict the category (frozen, liquid or none) based on polarimetric radar variables. One model was a long-shot model for the purpose of winning the competition (it didn't). The second was a complex black box model to represent the average learnability of the dataset. The third was a simple, human-readable model that would perform similarly to the above two models but possess the additional advantage of being easy to implement and comprehend.

1. Introduction

The number of training (847) and testing (346) patterns is misleadingly large. Because the dataset was concentrated in Oklahoma and collected on a handful of winter events, the true number of independent training and testing instances is actually far less. We hypothesized that there would be no virtual difference in skill between simple and more complex models on such a small dataset. Therefore, any good approach would be middle-of-the-pack. In such a situation, the most easily justifiable approach is to build simple models. If the testing patterns were chosen from the same set of cases, then it was likely that a nearest neighbor approach might even beat out the competition even if it would not be operationally feasible. Hence, we built three AI models:

- a nearest neighbor approach because we felt this afforded a reasonable chance of actually winning
- a neural network
- a simple decision tree

The nearest neighbor approach did not win because the creator of the data set had the foresight to choose the test patterns from a different pool than the training patterns! But as expected, the NN and decision tree were well within the performance bounds of the best submitted entries. Because the decision tree, especially, is much simpler, we suggest that it would be a better candidate for operational implementation than techniques that are harder to comprehend or implement.

If a human-readable data-driven model provides results statistically similar to complex methods, then the non-technical benefits of a human-readable model should cause it to be selected.

*Corresponding author: lakshman@ou.edu [1]The Cooperative Institute of Mesoscale Meteorological Studies (CIMMS), University of Oklahoma, [2]The National Severe Storms Laboratory, Norman, OK

2. Preprocessing

We omitted attributes that should, physically, not matter in the final classification, using only the following attributes:

1. Temperature in Celsius
2. Relative Humidity
3. Speed (computed from the u and v components)
4. Freezing level
5. Height above freezing (computed by subtracting the height from the freezing level)
6. Zdr
7. RhoHV
8. Kdp
9. Z

In addition, we created two new features that form a pre-classification: whether the pattern in question corresponds to "clear air" or to "frozen". To create the clear-air preclassification, we divided up the dataset into two parts: those for which the category was "none" and the rest. To create the frozen classification, we similarly created a binary dataset based on whether or not the category was "frozen".

Whether or not a pattern corresponded to clear-air was learned from the training data set using a decision tree (Quinlan 1993). The clear-air attribute was set to 1 if the following condition was met:

```
RhoHV <= 0.768208 AND \\
Z  <= 10.626065 AND \\
RelH > 81 AND \\
( HtAboveFreezing <= 853 OR \\
  HtAboveFreezing > 1113)
```

Otherwise, it was set to zero. The frozen attribute was set to 0 if the following condition was met:

```
RhoHV <= 0.487228 OR
(TmpC > -2.143738 AND \\
 Z <= 24.4076)
```

Otherwise, it was set to one.

So, the final number of inputs to the training set of routines was 11. We used WEKA (Witten and Frank 2005) to perform the training.

3. Methods

The training of each of the models followed a cross-validation approach. The model was trained on 90% of the data and tested on 10%. This was repeated ten times and the average cross-validated accuracy (percent correct) was used as the measure of the "goodness" of the model.

As explained earlier, the first model we trained was a nearest-neighbor model. A candidate pattern was assigned to a category based on the category of its 10 closest neighbors. The distance measure used was simply the Euclidean distance of the 11 feature attributes. Rather use a simple majority, we weighted the vote of each neighbor by the reciprocal of its Euclidean distance. Because the cross-validation samples (the 10%) were selected randomly from the training dataset, it was likely that at least some of the patterns were very similar. Not surprisingly, the cross validation accuracy of the nearest-neighbor approach was the highest of the three approaches we tried – 72%. If the test patterns were similarly chosen from the same pool, we felt, this rather simplistic approach even had a chance of winning. So, this was our official entry.

The next two approaches were aimed towards being in the middle of the leaders,

not neccessarily win. However, the goal of these two approaches was simplicity.

The neural network was chosen for simplicity of implementation and retraining – an operational implementation can simply read the new weights from a file as is done in Lakshmanan et al. (2007). The neural network was trained by backpropagation with a learning rate of 0.1, momentum of 0.2 with 45% of the training samples (i.e. 45% of the 90% used for training) used for early stopping. There were 7 hidden nodes. The neural network had a cross-validation accuracy of 68%.

The decision tree was chosen for human readability. It is relatively simple to examine; if human operations need to make decisions based on the output of the AI model, it is helpful if they can develop a "feel for the model", a process that is greatly aided if the model is a simple, easy to understand set of rules. The decision tree training was the method of Quinlan (1993) with a pruning threshold of 0.4 and stopping the splitting of nodes if it would result in leaves with less than 20 instances. The tree obtained is quite easy to comprehend – it is shown below. This simple model had a cross-validation accuracy of 68%.

```
clearair <= 0
|   frozen <= 0
|   |   tmpc <= -3.549988: frozen (35.0/15.0)
|   |   tmpc > -3.549988: liquid (168.0/64.0)
|   frozen > 0
|   |   speed <= 1.431182: liquid (47.0/19.0)
|   |   speed > 1.431182: frozen (436.0/83.0)
clearair > 0
|   tmpc <= 1.575012
|   |   Z <= -25.6025: none (31.0/8.0)
|   |   Z > -25.6025
|   |   |   frzl <= 853: none (30.0/9.0)
|   |   |   frzl > 853
|   |   |   |   tmpc <= -7.324982: none (29.0/14.0)
|   |   |   |   tmpc > -7.324982: frozen (51.0/19.0)
|   tmpc > 1.575012: liquid (20.0/9.0)
```

4. Results

The test data set was chosen from a different pool of instances than the training data set (different radar scans). Hence, not surprisingly, the nearest-neighbor approach did not win. Its Pearson Skill Score (PSS) on the test data set was 0.27. However, as expected, the methods all came in the middle of the pack, statistically not different from the other entries. The neural network approach had a PSS of 0.30 and the decision tree approach had a PSS of 0.28.

If the other entries are, as we expect, considerably more complex, we suggest the operation use of the decision tree above. In our favor, we'd like to point out that the complete human-readable description of our rules fits onto half-a-page.

Looking at the resulting decision tree, it is also quite clear that more training cases are needed when the air is truly clear ("no weather") and the temperature is above freezing – right now, all such cases are classified as "liquid" which is clearly erroneous.

Acknowledgments

Funding for this research was provided under NOAA-OU Cooperative Agreement NA17RJ1227. The statements, findings, conclusions, and recommendations are those of the authors and do not necessarily reflect the views of the National Severe Storms Laboratory (NSSL) or the U.S. Department of Commerce.

References

Lakshmanan, V., A. Fritz, T. Smith, K. Hondl, and G. J. Stumpf, 2007: An automated technique to quality control radar reflectivity data. *J. Applied Meteorology*, **46**, 288–305.

Quinlan, J. R., 1993: *C4.5: Programs for Machine Learning*. Morgan, Kaufmann, Los Altos.

Witten, I. and E. Frank, 2005: *Data Mining*. Elsevier, 524 pp.