Volcanic Eruption Prediction

Matt Barrett

Problem

The goal of this competition was to predict how long it would be until a volcano erupted. Our submissions were evaluated on the mean absolute error(MAE) between our predictions and the actual time until eruption

Data

The data given was 10 minutes of readings taken every 1/100th of a second from 10 sensors around the volcano, for a total of 60,000 data points per volcano. We were provided with training data from 4,300 volcanoes including the time until the volcano erupted. We were also given 4,500 volcanoes for which we were to predict the time until eruption.

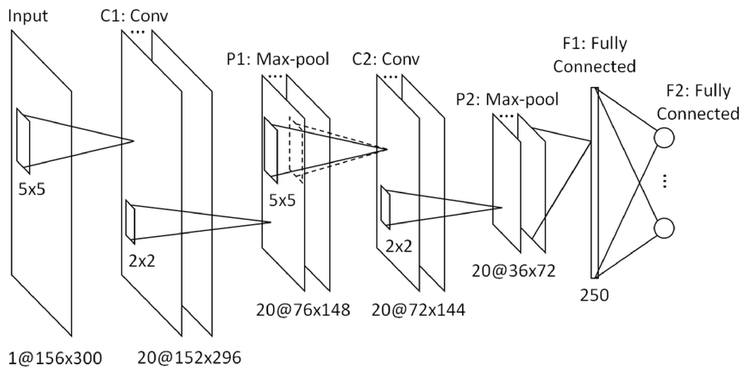
Approach

I took several approaches to the problem. I tried extracting features from the data two different ways and I used both neural networks and random forests on each set of extracted features. The most successful feature extraction was to get the mean, standard deviation, and every 10th percentile, from 0 to 100, from each sensor for a total of 130 features per volcano.

	sensor_1	sensor_2	sensor_3	...	sensor_8	sensor_9	sensor_10
mean	-1.363644	-0.069682	-4.534174	...	1.383094	0.564724	-1.542924
std	248.904124	380.558987	216.675726	...	261.692704	243.421896	548.082090
min	-2004.000000	-3583.000000	-1891.000000	...	-1291.000000	-2070.000000	-5808.000000
10%	-285.000000	-394.000000	-244.000000	...	-323.000000	-261.000000	-589.000000
20%	-181.000000	-247.000000	-154.000000	...	-212.000000	-168.000000	-370.000000
30%	-110.000000	-152.000000	-94.000000	...	-133.000000	-103.000000	-229.000000
40%	-52.000000	-73.000000	-44.000000	...	-64.000000	-50.000000	-109.000000
50%	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000
60%	51.000000	72.000000	43.000000	...	62.000000	48.000000	108.000000
70%	110.000000	152.000000	89.000000	...	133.000000	101.000000	225.000000
80%	176.000000	248.000000	145.000000	...	216.000000	169.000000	373.000000
90%	274.000000	393.000000	224.000000	...	330.000000	263.000000	588.000000
max	1934.000000	4636.000000	1799.000000	...	1446.000000	2604.000000	6104.000000

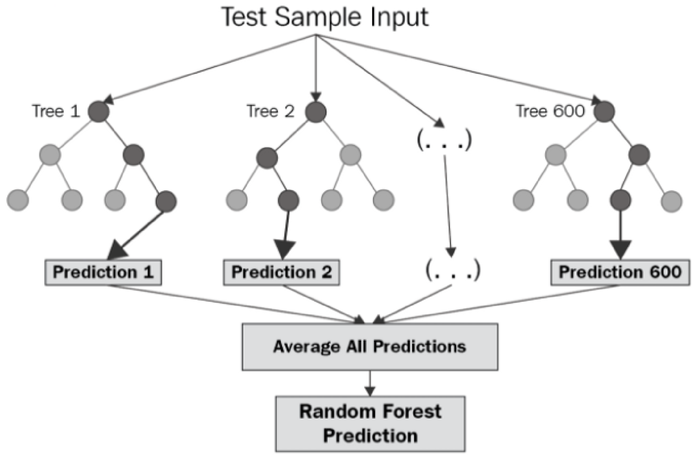
CNNs

I tried a variety of fully connected and convolutional neural networks. The best results came from a neural network made up of two pairs of convolutional/pooling layers followed by four fully connected layers



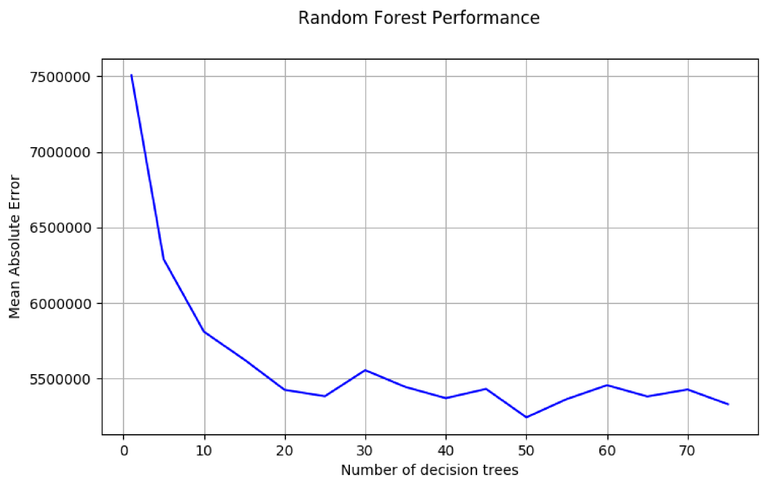
Random Forests

I tried random forests made up of increasing numbers of decision trees to find where accuracy leveled off, which tends to happen with random forests. In this case, the accuracy began to converge after around 25 decision trees.



Results

My random forests significantly out performed my neural networks. On my validation set, the best random forest had an MAE of 5.2 million, while my best neural network had an MAE of 10.6 million. The random forests were also considerably quicker to train and predict with.



Conclusions

I believe that random forests work better on this problem than neural networks because I am already extracting relevant features from the data. This is where random forests excel. Neural networks are best when you do not know which features are important, e.g. image classification. My current best score on the Kaggle competition is 264 out of 358 with an MAE of 7.7 million

264	**Matthew Barrett**	7755572

Source Code

https://github.com/mattb1888/cs5665_project

Image Citations

https://miro.medium.com/max/700/1*ZFuMI_HrI3jt2Wlay73IUQ.png

https://www.researchgate.net/profile/Shun_Miao/publication/280538045/figure/fig2/AS:354647291252736@1461565911272/Structure-of-the-multi-task-learning-convolutional-neural-network.png