There is a volcano on that Venus’ surface image?

Samuel Sanches

Abstract

Using the dataset https://www.kaggle.com/fmena14/volcanoesvenus, try to predict if there is a volcano on the image.

For that, I will use the machine learning tool from scikit-learn: Decision Tree Classifier, Gaussian NB and Random Forest Classifier, to see what is the best for the given dataset.

The findings is: the best tool was Decision Tree, with 70 nodes, getting an accuracy of 90%.

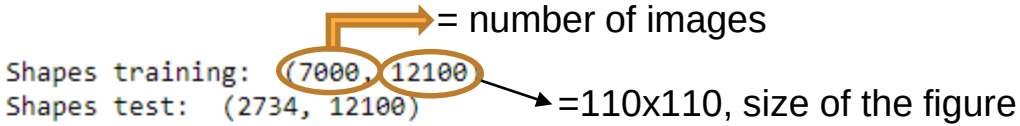
Motivation

Using machine learn, I want to create a model to predict if the image has or not a volcano on the Venus’ surface.

Dataset

The dataset is from kaggle: https://www.kaggle.com/fmena14/volcanoesvenus

Divided in 7000 images on the training and 2734 on the test csv files. Consisting of a 110x110 pixels image, as the image below shows:



	0	1	2	3	4	5	6	7	8	9	...	12090	12091	12092	12093	12094	12095	12096	12097	12098	12099
0	95	101	99	103	95	86	96	89	70	104	...	111	107	92	89	103	99	117	116	118	96
1	91	92	91	89	92	93	96	101	107	104	...	103	92	93	95	98	105	104	100	90	81
2	87	70	72	74	84	78	93	104	106	106	...	84	71	95	102	94	80	91	80	84	90
3	0	0	0	0	0	0	0	0	0	0	...	94	81	89	84	80	90	92	80	88	96
4	114	118	124	119	95	118	105	116	123	112	...	116	113	102	93	109	104	106	117	111	115

Dataset

The labels are divided as the image below:

```
Shapes labels training:  (7000, 4)
Shapes labels test:  (2734, 4)
```

	Volcano?	Type	Radius	Number Volcanoes
0	1	3.0	17.46	1.0
1	0	NaN	NaN	NaN
2	0	NaN	NaN	NaN
3	0	NaN	NaN	NaN
4	0	NaN	NaN	NaN

Data Preparation and Cleaning

The csv file was on a strange form, so I have to make they back to a matrix form:

‘train_reshape = df_train.values.reshape((df_train.shape[0],1,110,110))’

Then making back to 0 to 255 to get the RGB like for image:

‘train_reshape_graunded_to_rgb = train_reshape/255.0’

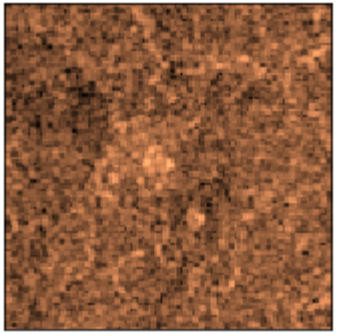
And transpose to get the 110x100 image:

‘train_reshape_graunded_to_rgb_transpose = train_reshape_graunded_to_rgb.transpose([0, 2, 3, 1])’

Data Preparation and Cleaning

With that, I could see the image, like those:

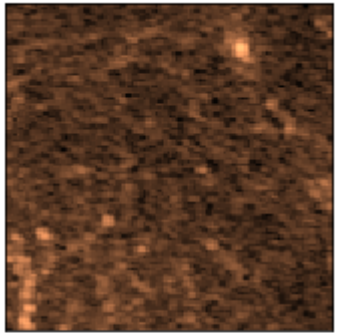
Volcano?: Yes



Volcano?	1.00
Type	3.00
Radius	17.46
Number Volcanoes	1.00

Name: 0, dtype: float64

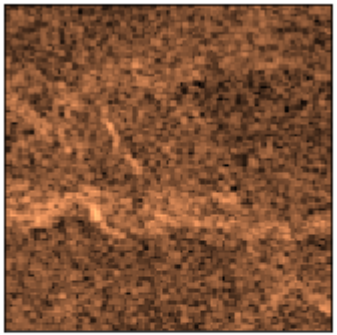
Volcano?: No



Volcano?	0.0
Type	NaN
Radius	NaN
Number Volcanoes	NaN

Name: 20, dtype: float64

Volcano?: No



Volcano?	0.0
Type	NaN
Radius	NaN
Number Volcanoes	NaN

Name: 6500, dtype: float64

Research Question

With the dataset, I want to try to create a model to predict if the image has or not a volcano.

Methods

To make the predictions I use:

- Decision Tree Classifier
- Gaussian Naïve Bayes
- Random Forest Classifier

Findings

The methods predictions accuracy was:

- Decision Tree: 90%
- Gaussian NB: 33%
- Random Forest: 84%

Limitations

Making simple machine learn techniques I could get an acceptable accuracy, with the simple Decision Tree.

That just say if the image has or not a volcano, but like the dataset shows, we have some images with more than just one volcano.

Conclusions

I could get a model to predict if a image has a volcano or not.

References

Some websites were very helpfull:

https://www.kaggle.com/fmena14/exploratory-analysis

https://gogul09.github.io/software/image-classification-python

https://www.digitalocean.com/community/tutorials/how-to-build-a-machine-learning-classifier-in-python-with-scikit-learn

https://stackoverflow.com/questions/34165731/a-column-vector-y-was-passed-when-a-1d-array-was-expected

http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html