Challenge 1: Fire mapping

Team Wildcats: Margaret Hughes Amissah-Mensah, Jiyeon Park, Lee Park

Dr. Bing Zhang Department of Statistics at the University of Kentucky

There are five test linescan images of wild forests in Australia and this project is to map the fire in the image based on their linescan values (infra-red color amount). We took different approaches based on some characteristics of each test linescan image. When linescan images of the same geographical region in the training image set are available, we utilized those to predict area on fire. Note that, as though the coordinates of the training linescan images are close enough to be used, they are not considered if the time difference between the test linescan image and the training linescan image are noticeably large. We will answer the first five questions separately with respect to the approaches.

I. Approach 1: When geographically and chronologically relevant training image(s) is available

1. The key elements of your approach.

We tried to overlap images from closest time and location. We call this *overlap-and-mask* strategy. One key element of *overlap-and-mask* is to find the picture taken from the closest time and location. Since the fire light vanishes after the fire burns the bushes, it gets difficult to capture the fire in a later stage. However, using this method, we could get clearer picture of the fire since the fire is clear in the earlier stage.

2. The training algorithm & feature engineering (Figure 1)

In short, we overlapped the training and test linescan images and singled out the pixels whose linescan values are greater than 100. Then using the contraction and dilation method, we made this mask smoother. Unless exhaustive effort to extinguish the fire is not provided, the fire region is likely to remain in the area. Otherwise, we apply the second approach.

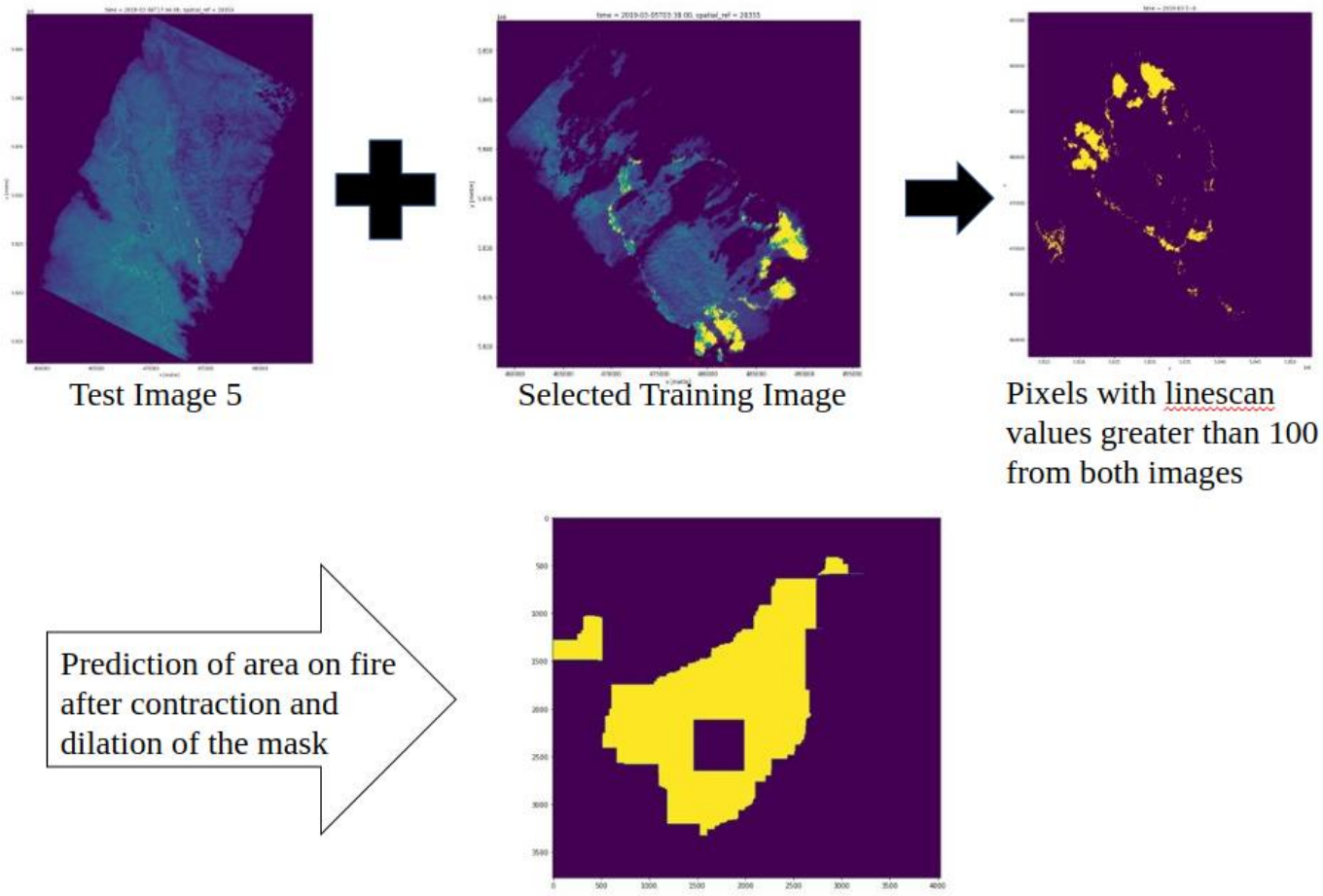

Figure 1 : Illustration of the overlap-and-mask strategy

3. Approach to model validation

We investigated how well this algorithm works with the training and validation datasets. The table 1 below shows their performance. In addition, through another validation process, we found the parameters for the kernelOpen and kernelClose that make the prediction stable while attaining high F1 score. When choosing the parameters for kernelOpen and kernelClose, we tested different sets of numbers for them.

	Train/Validation	Linescan id number	Time	F1 Score
Area 1	Train	74	3/9 05:59	0.9244
	Validation	92	3/9 14:29	
Area 2	Train	97	3/15 3:12	0.939
	Validation	124	3/16 4:58	
Area 3	Train	27	1/27 4:00	0.876
	Validation	29	1/27 8:01	

Table 1 : Performance of the overlap-and-mask algorithm

4. **Highest-performing features**

The features that were used to select the training image are coordinates and time of the image taken; furthermore, when predicting the fire, the only features were the linescan values of the training and the test linescan image. Therefore, this discussion is not germane to this algorithm.

II. Approach 2: When such a relevant training image is not available

1. **The key elements of your approach.**

When the first approach is not available, the second approach trains models with pixel-wise data from training images sharing some characteristics. Since the linescan images are taken from an airplane flying below the cloud, we conjectured that the missing values would be from the amount of smoke. Therefore, amount of smoke was a critical factor when choosing the small number of training linescan images to train candidate models.

2. **The training methods you used.**

We split both the training and test datasets into the one for clear area where linescan values are greater than 0 and smoky area where linescan values are 0. Then, we tested various models both parametric and non-parametric; they are, including but not limited to, logistic regression, linear support vector machine, random forest, and the K-nearest means.

3. Feature engineering and features selection.

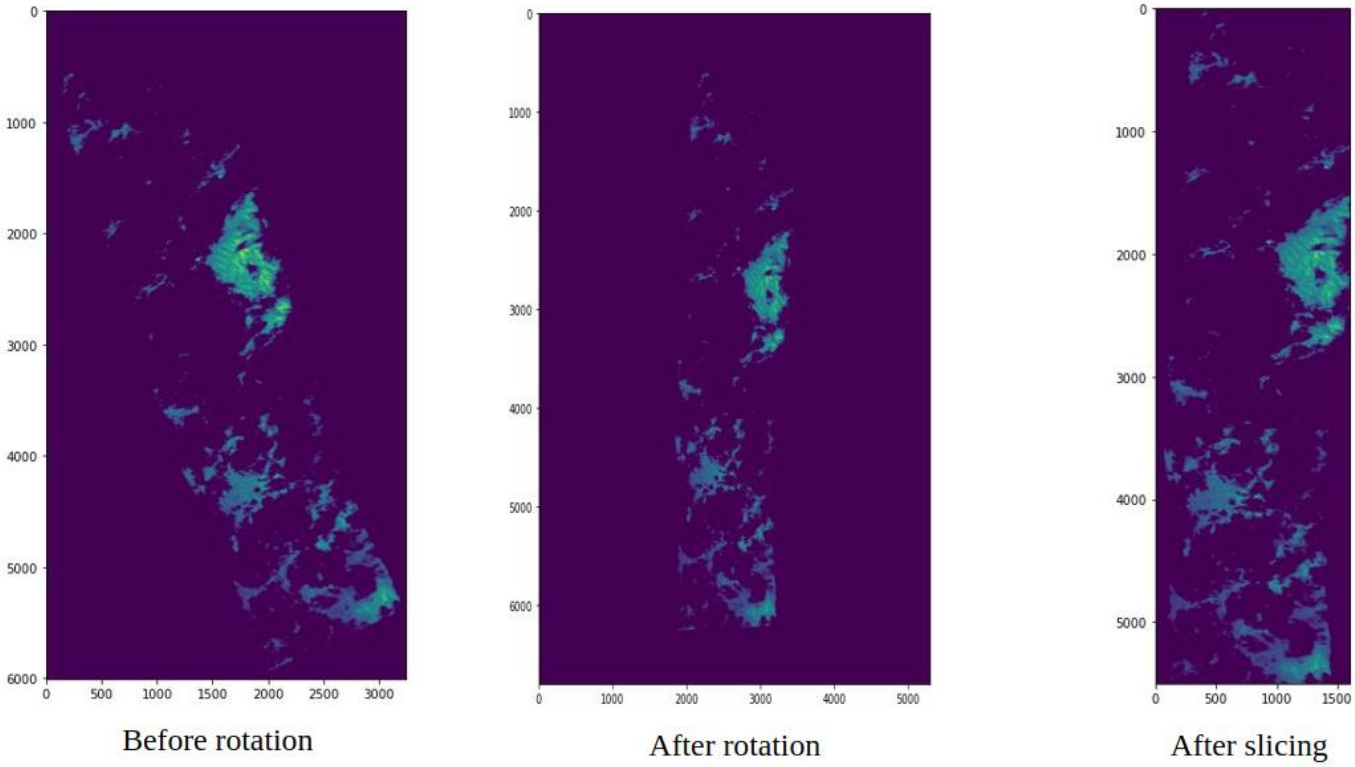

Figure 2: Rotating and erasing pixels outside the linescan image

There are 9 numeric features, before introducing quadratic and interaction terms, and 25 binary categorical features for each pixel. Obviously, the first numeric feature is the linescan value; the next six numeric features are historical satellite images from the landset7 and the landset8. We used the NBART because of the wild terrain shapes of Australian wild forests. The colors selected are red, green, blue, nir, swir_1 and swir_2; using some of those colors, we also computed the NDVI and the MNDWI, which indicates the amount of vegetations and aquatic substances in each pixel. The last two numeric features, ‘distance’ and ‘distance_mask’, indicate distances to certain points in the linescan image. The feature 'distance’ is the distance of each pixel to the closest pixel with linescan value greater than 200. The other, ‘distance_mask’, is the distance to the closest pixel within an inflated mask with the threshold linescan value equal to 100. The contraction (kernelOpen) and dilation (kernelClose) parameters are 1 and 36, respectively. This mask consistently and stably predicted the fire within the clear area. The 25 categorical features indicate whether a given pixel is inside a mask engineered by a combination of parameters for contraction and dilation of the mask. These 25 categorical features were used when we fit the models for the clear area.

In the previous section, the smoky area was defined as area with pixels whose linescan values are 0. However, the image on the left in the figure 2 suggests that not all pixels with linescan values equal to 0 are smoky areas. To be classified as smoky areas, they must be within

the linescan frame; in the figure 2, the linescan frame is tilted in about 30°. There were two challenges in this process. We wanted to find the perfect angle to straighten up the image; in addition, when rotating back the image, they retain the correct coordinate information. The detail of this procedure can be found in the github repository.

When all the features are collected, we transformed the numeric features using Yeo-Johnson transformation and created quadratic and interaction features. Then, we standardized the numeric features. When it comes to the feature selection for our model, instead of dropping individual variables, we used the L2 regularization to reduce the dimension of the feature set.

4. Model validation

Creating a training dataset needs a careful approach. When the fire is small compared to the size of the linescan image, every model is geared toward predicting no-fire if we fit the model with every pixel in the image. Therefore, stratified sampling from a given image was necessary. In the final training dataset, 50% of the cases were for fire and the other half were for the no-fire.

Cross-validation was used not only to select a model but also to find the parameters that maximize performance of each model. The scikit-learn's GridsSrachCV method was used.

5. Highest-performing features

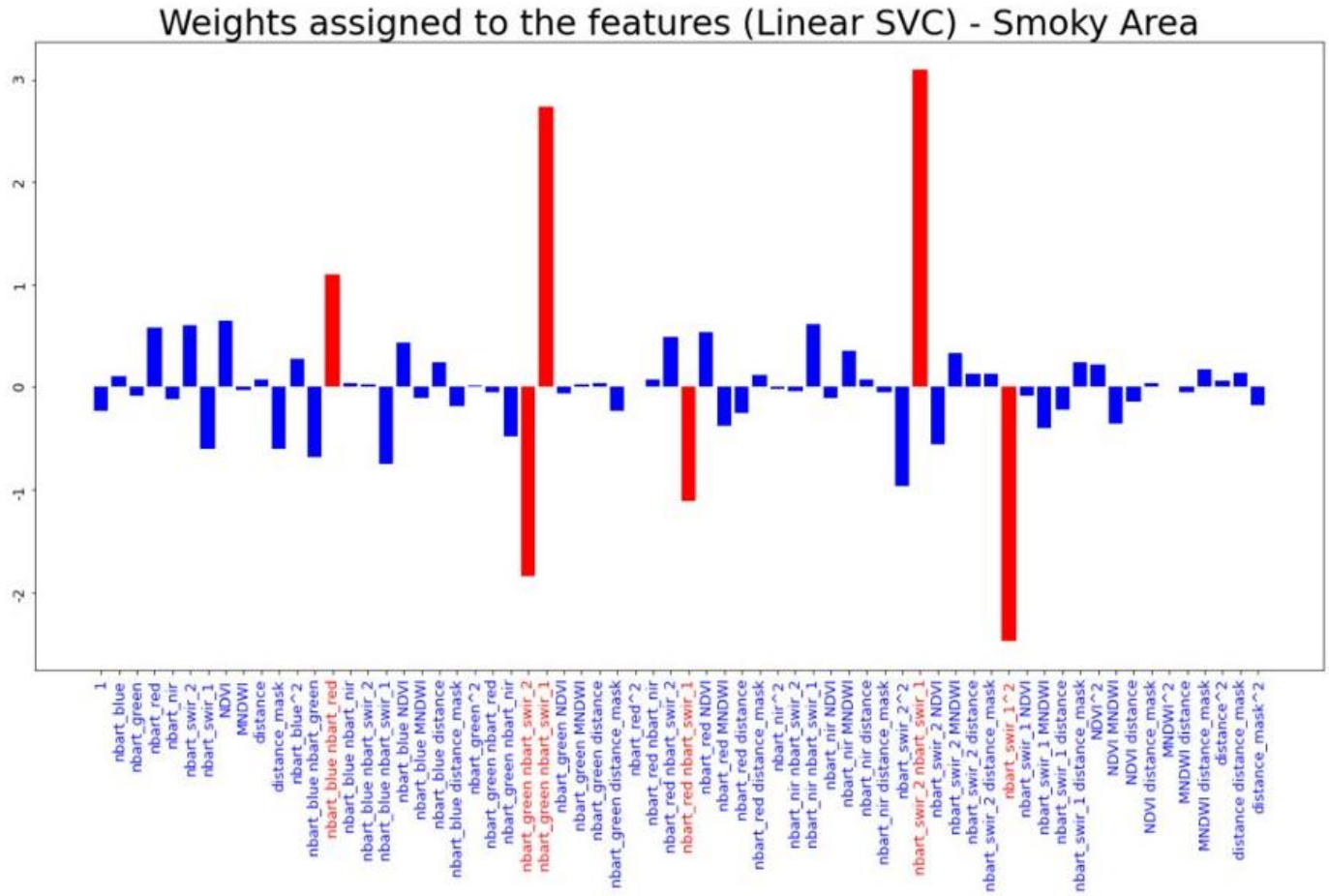

Figure 3. Weights assigned to the features for the linear SVC model

The figure 3 shows that there are six features that are assigned of absolute weights greater than 1. They are all interaction or quadratic features: blue*red, green*swir_2, green*swir_1, red*swir_1, swir_2*swir_1, and swir_1^2. However, it does not mean that swir_1 and swir_2 are important features; as you see the signs of weights for features including swir_1 and swir_2 are contracting, we rather experienced the multicollinearity issue. Among the individual features, we found that ‘distance_mask’ had a large weight relative to other individual features while the weight for the distance feature was noticeably smaller.

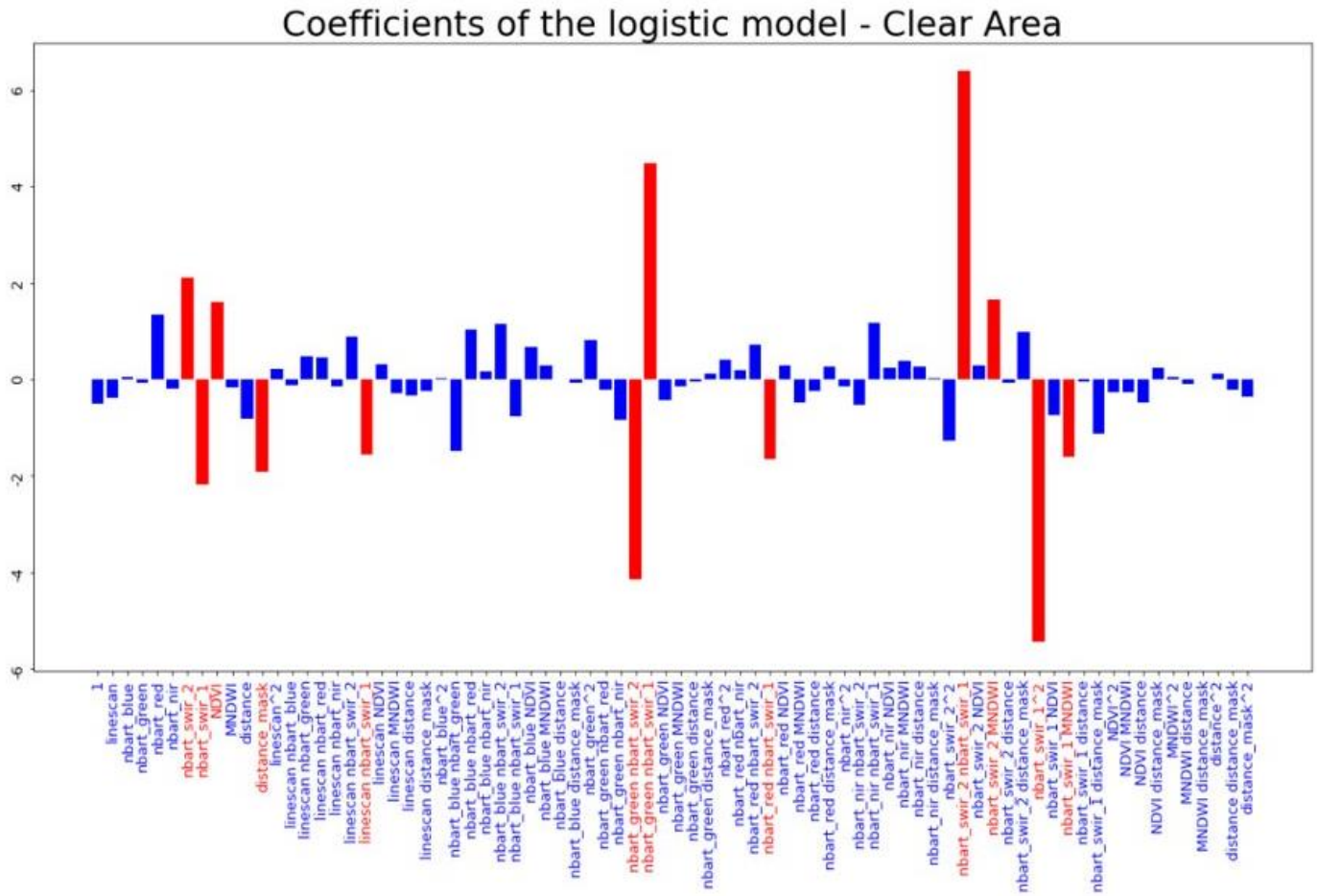

Figure 4. Coefficient of the features of the logistic regression model

The figure 4 suggests that, for the logistic regression model used for the clear area, similar to the linear SVC model, we have the similar issued with swir_1 and swir_2. However, it also suggests that the NDVI and ‘distance_mask’ are important features.

Opposite to our assumption, the feature ‘distance’ was not that important in both models. One reason is that ‘distance’ alone without the information on direction is not enough. The ‘distance_mask’ alleviated this concern but at the same time might also have reduced the importance of ‘distance’.

6. Evolution of the approach and the decisions at each point.

Feature engineering was the key to the second approach. When we simply tried to fit the model with linescan values or other values from the satellite, they were not useful.

7. **What is innovative and unique about your approach?**

Our approach is innovative as we tackled this problem not in a uniform but in the flexible way in that we choose a model based on certain conditions. Also, cutting off the pixels outside of the linescan frame is innovative in that this shrinks the size of the image data and allows the model to be fit only with relevant pixel data. Since the model is fitted with a training dataset containing information of more than 70000 pixels, several outliers won't change the performance of the model. For the first approach, if our assumptions are intact, the algorithm is simple and fast while sustaining a high performance. Therefore, our approaches are robust.

8. **Any limitations bushfire authorities should be aware of**

The first approach requires to check certain conditions that have already been discussed. The second approach requires exhaustive steps of data preprocessing and feature engineering.

9. **Could your model be applied outside Australia?**

Yes.

10. **The most important breakthrough**

In the middle of the project, we changed our mind and began to think that complicated models do not necessarily outperform simpler models. This change helped us to explore simpler and more interpretable models.

11. **The hardest thing about solving this problem.**

Memory management was the challenge when dealing with large-size datasets.