Rishi Sharma

8/6/2023

Disaster Relief Project: Part 2

Introduction

During the Haitian Earthquake Crisis of 2010, a key method by which to identify survivors was by locating manmade shelters. Based on the problem statement, many of these shelters used Blue Tarps as covers. Using images taken and classified by the Rochester Institute of Technology, we will attempt to fit optimal models for classifying pixels in the images as either "Tarp" or "Non-Tarp". In this case, "Tarp" indicates the presence of a Blue Tarp and likely, displaced peoples, while "Non-Tarp" indicates image pixels of other classes (Rooftop, Soil, Vegetation, Various Non-Tarp). The 7 model types are Logistic Regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis, K-Nearest Neighbors, Penalized Logistic Regression, Random Forest, and Support Vector Machine. Decisions on parameter tuning, threshold selection, model results and comparisons, and data transformations will all be explained in the report. The end goal is to have a model which very accurately identifies displaced people.

Exploratory Data Analysis (Training)

The data file presented is intended to be used as a training set for the models. It is a CSV ("HaitiPixels.csv") that contains pixel color data for some images that the RIT took. The data was loaded into R with the headers already present. An RGB color model is used for each pixel (row) in the data, with colors Red, Green, and Blue each having their own column (see here a link to Wikipedia that helps describe the "0 to 255" RGB color model). There are a little over 63,000 rows in the training data set. Below is an overview of how many rows equate to each class in the file.

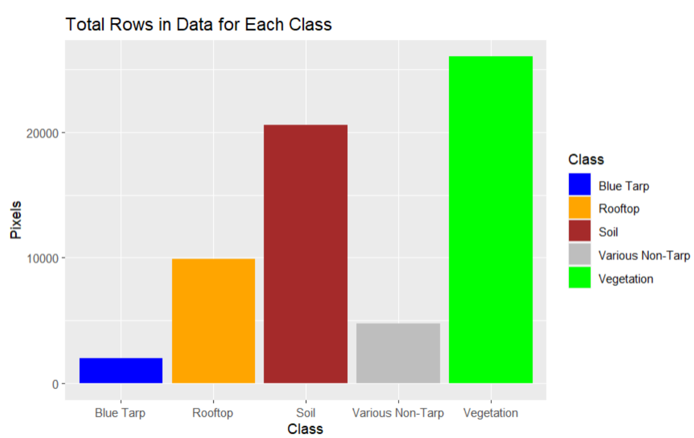


Figure 1. Rows of Data for Each Class

As we can see in Figure 1, Blue Tarps only make up a very small percent of total rows in the data. There are still around 2000 rows that are classified “Blue Tarp”, so enough to work with in the models. Overall, Vegetation and Soil seem to compose most rows in the data, making Blue Tarps, hopefully, stand out more.

Given that, to identify displaced people, we only care whether a Blue Tarp is present or not, a new column was added to the data set named “Displaced”, which simply marks all Blue Tarp rows as “Tarp”, and all Non-Blue Tarp Rows as “Non-Tarp”. From this, there is now a binary variable for model predictions on identifying displaced survivors of the disaster.

It is important to explore what we expect to see in the RGB color model for Tarps vs. Non-Tarps. See below the total intensity by color (sum up columns values per class) for each of the 2 classes.

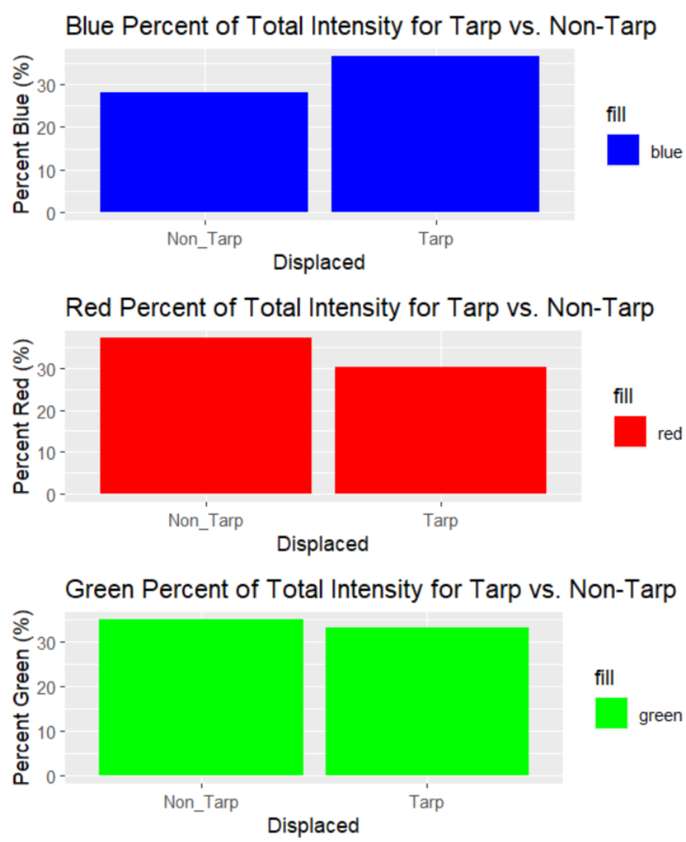


Figure 2. Tarp vs. Non-Tarp Color Intensity

As expected, we see a much higher blue intensity present in Tarps when compared to other classes. The same cannot be said for other colors, which Tarps are lacking.

Another key insight of EDA was how closely correlated each of the color values are in an RGB model. All pairings of the 3 colors share a correlation of 0.90 or higher.

The reason for high correlation is because, in an RGB color model, the magnitude of values correlates to vibrancy and shading. This means the proportion of values between Red, Green, and Blue are not all that matters to determine the exact color present in then pixel. It will be important to capture interactions between colors, and explore transformations that more easily distinguish the vibrancy and shading elements of a pixel. For example, Figure 3 gives a good demonstration of how strongly correlated green and blue intensities are, while also showing how very high intensities of Blue largely belong to Tarp classified pixels.

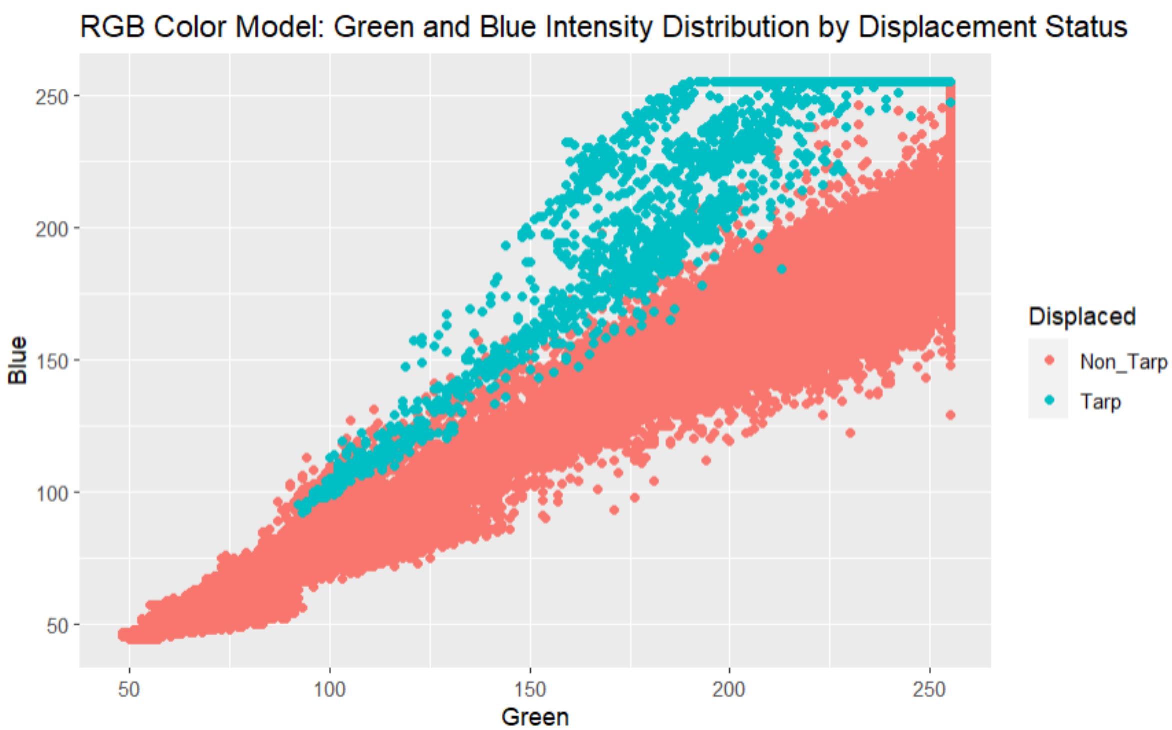


Figure 3. Green and Blue Correlation for Tarps and Non-Tarps

The figure shows that intensity, the scaling of color values, does not have much impact in differentiating the classes, but color (the hue) does. Once it became clear that intensity would not be as significant, it was also clear that another color model would need to be explored in order to separate an intensity parameter from that of hue and brightness.

Because of this, an HSV transformation was explored and tested with different models in the process. Converting from RGB to HSV color model is fairly straight forward, and can be done in R with existing packages and functions. HSV means Hue, Saturation, and Value (see [here](#) documentation for the rgb2hsv() function in R). When the RGB data is converted to HSV format, the correlation between predictors drops out. No pairing of hue, saturation, or value has an absolute correlation higher than 0.2. Figure 4 shows how saturation and hue are now separated in determining the classes.

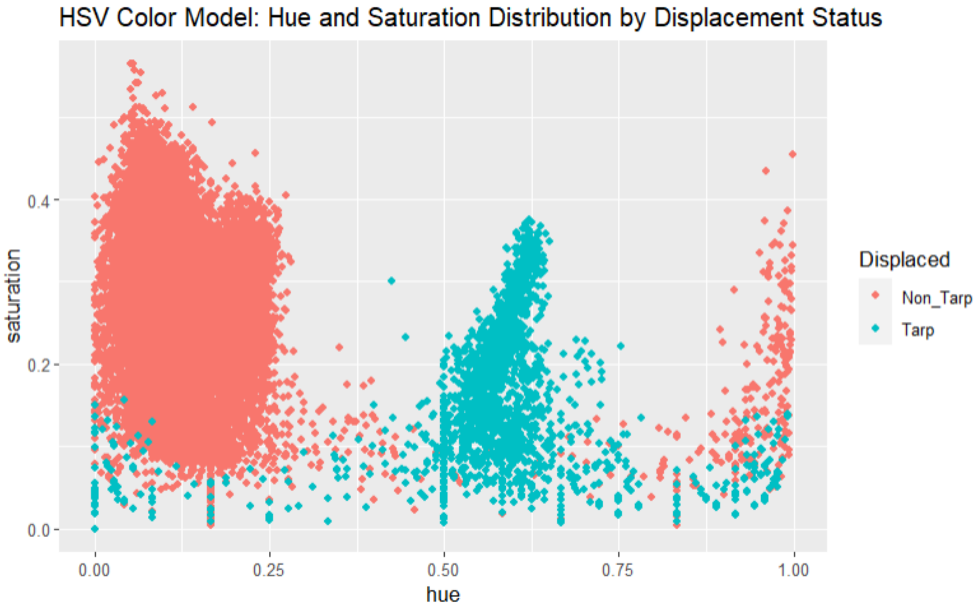


Figure 4. Intensity and color impact on classification

Given that hue is the “color” parameter, it makes sense how it best distinguishes Blue Tarps from Non-Tarp landscapes. Saturation (intensity), on average, seems lower for Tarps than Non-Tarps , but varies enough that it doesn’t seem as important as how distinctly hue separates the classes. An interesting observation is the appearance of Non-Tarps near a hue of 1.0. This is dues to the cylindrical nature of the HSV color model. In a future study, shifting the HSV values by a constant to push the Non-Tarps into one grouping would better distinguish the training data set.

Model Fitting, Selection, and Validation

At a high level, model fitting will be done through 10-fold Cross Validation on the training data, and model selection will be complete by comparing the effect of interaction terms on accuracy, as well as the potential impact from an HSV transformation. By default, cross validation uses Accuracy to determine the final model in the “**caret**” package of R. This makes sense given the problem is of the classification type. From here, threshold selection on probability predictions will be done based on Accuracy, TPR, FPR, and FNR. The trade-off between FPR and FNR will be very important here. A low FNR is the most ideal, as it means we missed less displaced survivors (our main objective is to flag them all as accurately as possible and not miss them). However, doing so at a high FPR means wasted resources/time on checking incorrect locations. Time is crucial for search and rescue, so wasting it on incorrect checks is not ideal.

Logistic Regression

Initially, the logistic regression was built using the RGB color model with no interactions present as a base. Then, using ANOVA and Accuracy, the base model is compared to a similar logistic model with interactions present. See below in Figure 6 the results of ANOVA Chi-Sq comparison between the models.

```
Analysis of Deviance Table

Model 1: Displaced ~ Red + Green + Blue
Model 2: Displaced ~ (Red + Green + Blue)^2
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1     63237     1769.5
2     63234     1327.7  3   441.82 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 5. ANOVA Logistic Regression for No Interactions vs. Interactions Models

Based on this result, it seems clear that interactions in the Logistic Regression model are significant, so they were left in for the optimal Log-Reg RGB model selection. Furthermore, the AIC of the interactions model was 1341.7, much lower than that of the no-interactions model (1777.5). Figure 6 is a summary of the final RGB model for Logistic Regression.

```
Call:
glm(formula = Displaced ~ (Red + Green + Blue)^2, family = "binomial",
    data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.3660  -0.0271  -0.0071   0.0000   2.9738

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.152e+01  8.856e-01 -13.004  < 2e-16 ***
Red          5.639e-02  3.676e-02   1.534 0.125018
Green       -1.903e-01  5.025e-02  -3.787 0.000152 ***
Blue         2.911e-01  4.595e-02   6.335 2.38e-10 ***
Red:Green   -1.114e-03  3.577e-04  -3.114 0.001845 **
Red:Blue    -6.199e-04  3.338e-04  -1.857 0.063287 .
Green:Blue   1.254e-03  2.334e-04   5.373 7.75e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 17901.6  on 63240  degrees of freedom
Residual deviance:  1327.7  on 63234  degrees of freedom
AIC: 1341.7
```

Figure 6. Logistic Regression Final RGB Model Summary

As expected, the biggest positive impact on the odds of there being a Tarp comes from the “Blue” predictor. It is important to note that the Red parameter, and interactions between Red:Blue were both flagged as insignificant. Nonetheless, we leave all parameters and interactions in for this model. From here, it was important to obtain the ROC curve and the results of predictions using the model.

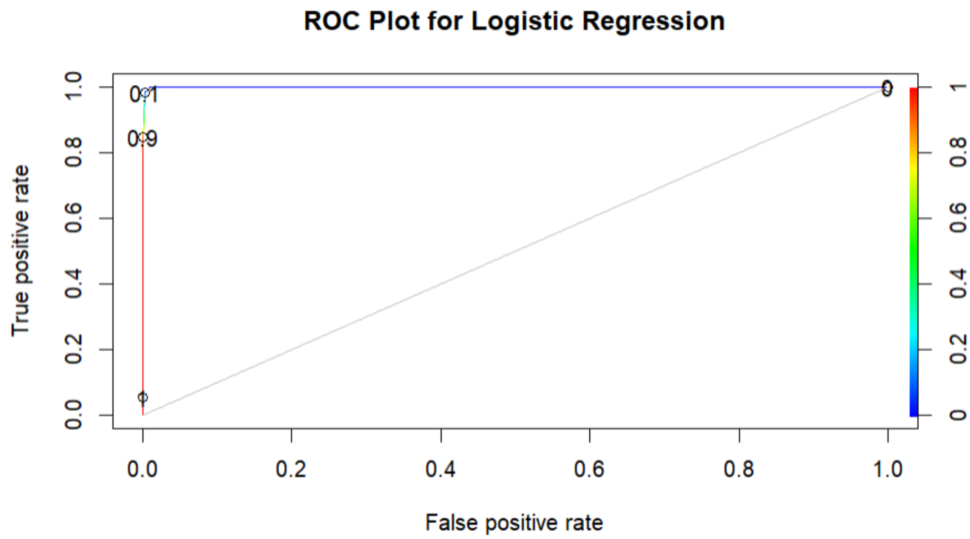


Figure 7. ROC Curve for RGB Logistic Regression

prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.994	1.000	0.170	0.000	0.170	0.997
0.10	0.995	1.000	0.155	0.000	0.155	0.997
0.15	0.995	0.999	0.145	0.001	0.145	0.997
0.20	0.995	0.999	0.137	0.001	0.137	0.997
0.25	0.995	0.999	0.130	0.001	0.130	0.997
0.30	0.995	0.999	0.125	0.001	0.125	0.997
0.35	0.995	0.999	0.117	0.001	0.117	0.997
0.40	0.995	0.999	0.113	0.001	0.113	0.997
0.45	0.995	0.998	0.105	0.002	0.105	0.997
0.50	0.995	0.998	0.094	0.002	0.094	0.998
0.55	0.996	0.998	0.084	0.002	0.084	0.998
0.60	0.996	0.998	0.072	0.002	0.072	0.998
0.65	0.996	0.998	0.065	0.002	0.065	0.998
0.70	0.996	0.998	0.055	0.002	0.055	0.998
0.75	0.996	0.997	0.044	0.003	0.044	0.998
0.80	0.996	0.997	0.036	0.003	0.036	0.998
0.85	0.996	0.997	0.027	0.003	0.027	0.998
0.90	0.996	0.996	0.018	0.004	0.018	0.998
0.95	0.991	0.990	0.002	0.010	0.010	0.995

Figure 8. Thresholds and Model Accuracy Results for Logistic RGB Regression

Figures 7 and 8 are the ROC curve and Threshold Accuracy Table, respectively. The ROC curve indicates that we can select a threshold that yields a high TPR while maintaining a low FPR. This means we can identify displaced people correctly at a high rate (high Recall), while incorrectly flagging displaced people at a low rate. More importantly, our criteria for selecting a threshold must include an evaluation of FNR. The reason we must emphasize FNR is because of our primary objective, identify displaced survivors. FNR indicates those survivors who we incorrectly marked as “Non-Tarp”, meaning they could be missed by a rescue effort. Because of this a threshold of 0.7 was ideal for this model, as Figure 9 shows a low FNR of 0.002 and a relatively low FPR (0.055) compared to what we will see in the later models. Overall, this model is a very strong start for classifying the training pixels.

We repeat the same procedure, but with an HSV transformation on the data set instead. Now that there are 3 predictors that aren’t closely correlated, a logistic model is created from them.

```
Call:  NULL

Coefficients:
(Intercept)          hue   saturation        value
     -8.721       13.380       -9.810        4.620

Degrees of Freedom: 63240 Total (i.e. Null);  63237 Residual
Null Deviance:      17900
Residual Deviance: 5464         AIC: 5472
```

Figure 9. Logistic Regression Final HSV Model Summary

Note the higher AIC and Residual Deviance in Figure 10 when compared to Figure 7. This is an early indication that The RGB model will be favored for Logistic Regression. Proceeding with the same approach, the ROC curve and prediction threshold tables were created for the HSV Logistic Model.

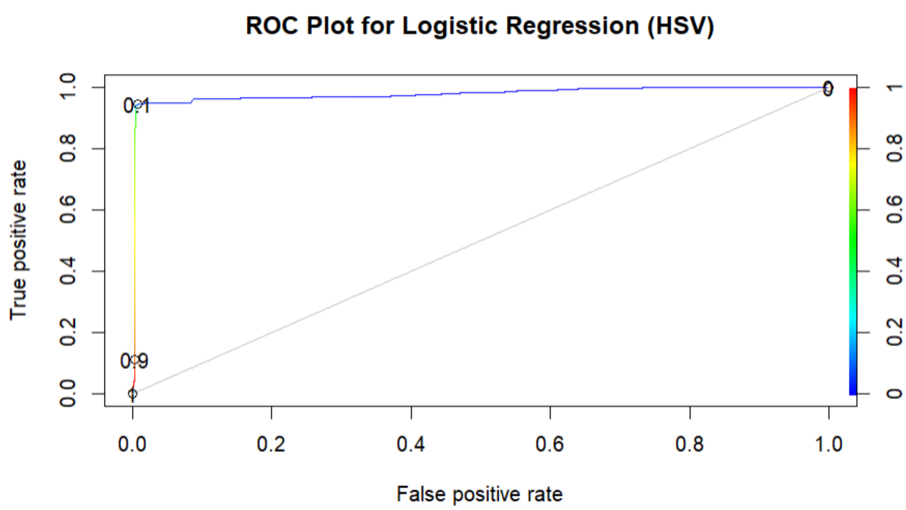


Figure 10. ROC Curve for HSV Logistic Regression

prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.967	0.996	0.934	0.004	0.934	0.983
0.10	0.968	0.996	0.886	0.004	0.886	0.984
0.15	0.971	0.996	0.788	0.004	0.788	0.985
0.20	0.975	0.996	0.650	0.004	0.650	0.987
0.25	0.981	0.996	0.476	0.004	0.476	0.990
0.30	0.985	0.996	0.346	0.004	0.346	0.992
0.35	0.989	0.996	0.232	0.004	0.232	0.994
0.40	0.991	0.996	0.159	0.004	0.159	0.995
0.45	0.992	0.996	0.113	0.004	0.113	0.996
0.50	0.993	0.996	0.091	0.004	0.091	0.996
0.55	0.993	0.996	0.080	0.004	0.080	0.997
0.60	0.994	0.996	0.075	0.004	0.075	0.997
0.65	0.994	0.996	0.070	0.004	0.070	0.997
0.70	0.994	0.996	0.068	0.004	0.068	0.997
0.75	0.994	0.996	0.067	0.004	0.067	0.997
0.80	0.994	0.996	0.065	0.004	0.065	0.997
0.85	0.994	0.995	0.059	0.005	0.060	0.997
0.90	0.991	0.993	0.057	0.007	0.057	0.995
0.95	0.982	0.983	0.053	0.017	0.056	0.991

Figure 11. Thresholds and Model Accuracy Results for Logistic HSV Regression

Figures 10 and 11 show the slightly worse ROC and Accuracy results associated with the HSV model when compared to RGB w/ interactions. At the ideal threshold of 0.80, all Accuracy metrics appear slightly worse for the HSV model, so the final Logistic Model chosen will be RGB Color model with interactions included. Table 1 provides a summary of the two model results.

Table 1. Logistic Regression Models Comparison

Model	Accuracy	TPR	FPR	FNR	AUC	Prob-Threshold
RGB-Interactions	0.996	0.998	0.055	0.002	0.9996	0.70
HSV	0.994	0.996	0.065	0.004	0.9768	0.80

Linear Discriminant Analysis (LDA)

The LDA model will take a similar approach to Logistic Regression, with similar factors determining the threshold choices. However, for LDA, the choice to completely exclude interaction was made due to the very high correlation between the RGB predictors. In general, it is not common to include interactions when a linear assumption is made on the predictor-response relationship. So, the only models to compare are RGB and HSV, with no interactions.

For the final RGB model, as expected, Blue is the only predictor with a positive correlation to the response when in the presence of the other predictors (Figure 12).

```
Call:
lda(x, grouping = y)

Prior probabilities of groups:
  Non_Tarp       Tarp 
0.96802707 0.03197293 

Group means:
              Red    Green     Blue
Non_Tarp 162.7604 152.5808 122.4993
Tarp     169.6627 186.4149 205.0371

Coefficients of linear discriminants:
              LD1
Red   -0.02896984
Green -0.02328544
Blue   0.06923974
```

Figure 12. LDA RGB Model Summary

Looking at the ROC curve for the RGB LDA Model, there seems to be a significant TPR and FPR tradeoff happening at early higher TPR values, unlike what happened with Logistic Regression.

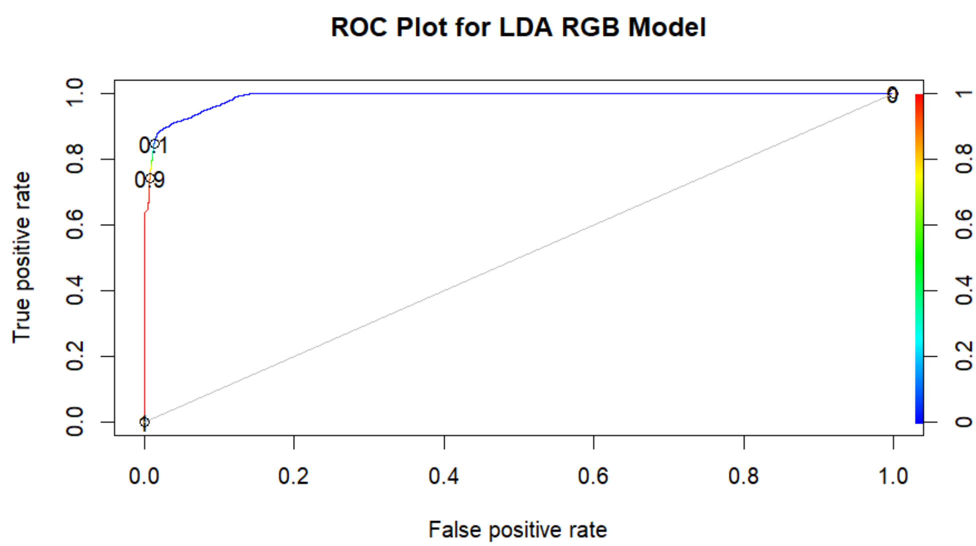


Figure 13. ROC Curve for LDA RGB Model

prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.985	0.994	0.289	0.006	0.289	0.992
0.10	0.985	0.993	0.259	0.007	0.259	0.992
0.15	0.985	0.992	0.242	0.008	0.242	0.992
0.20	0.985	0.992	0.236	0.008	0.236	0.992
0.25	0.984	0.992	0.228	0.008	0.228	0.992
0.30	0.984	0.991	0.223	0.009	0.223	0.992
0.35	0.984	0.991	0.213	0.009	0.213	0.992
0.40	0.984	0.990	0.209	0.010	0.209	0.992
0.45	0.984	0.990	0.204	0.010	0.204	0.992
0.50	0.984	0.990	0.198	0.010	0.199	0.992
0.55	0.984	0.990	0.194	0.010	0.194	0.992
0.60	0.983	0.989	0.190	0.011	0.191	0.991
0.65	0.984	0.989	0.185	0.011	0.186	0.992
0.70	0.983	0.989	0.180	0.011	0.180	0.991
0.75	0.983	0.988	0.176	0.012	0.176	0.991
0.80	0.983	0.988	0.165	0.012	0.166	0.991
0.85	0.983	0.987	0.159	0.013	0.160	0.991
0.90	0.982	0.987	0.152	0.013	0.153	0.991
0.95	0.981	0.985	0.140	0.015	0.141	0.990

Figure 14. Thresholds and Model Accuracy Results for LDA RGB Regression

Generally, the Accuracy of LDA is lower than that of Logistic Regression. Once again, the threshold will be selected with the idea that missing any displaced persons is very bad, but we need to stay in a reasonable range of FPR. For relatively high accuracy in this model, 0.65 appears to be a sweet spot. The FNR is much higher than that of Logistic Regression, and the same goes for FPR, at over triple what we saw in the best Log-Regression model. Overall, this model should be avoided. Next, we check if an HSV transformation can help.

As with the RGB model, no interaction terms are tested. It appears hue and value seem to play positive roles in determining whether there is a Blue Tarp in LDA (Figure 15). This makes sense, in that hue refers to wavelength of the color, for which Blue is larger than Green and Red. Likewise, the Tarps tend to be brighter and more vibrant in the image compared to the surrounding soil and vegetation. Therefore, we expect a higher "value" predictor to appear for Tarps.

```
Call:
lda(x, grouping = y)

Prior probabilities of groups:
  Non_Tarp       Tarp
0.96802707 0.03197293

Group means:
               hue saturation     value
Non_Tarp 0.1385302  0.2455782 0.6415835
Tarp     0.5685617  0.1674437 0.8069510

Coefficients of linear discriminants:
                  LD1
hue        14.1204183
saturation -1.4435343
value       0.9983322
```

Figure 15. LDA HSV Model Summary

The ROC curve for the HSV color model is much better than that of RGB for LDA. We see less of a significant tradeoff between TPR and FPR at early high TPR points on the curve. The very end of the curve has a performance that is equivalent to random guessing. Referring to Figure 4, this could be dues to the cylindrical nature of HSV models, and how the highest hue values, if shifted, would become the lowest to match with the rest of Non-Tarp classes.

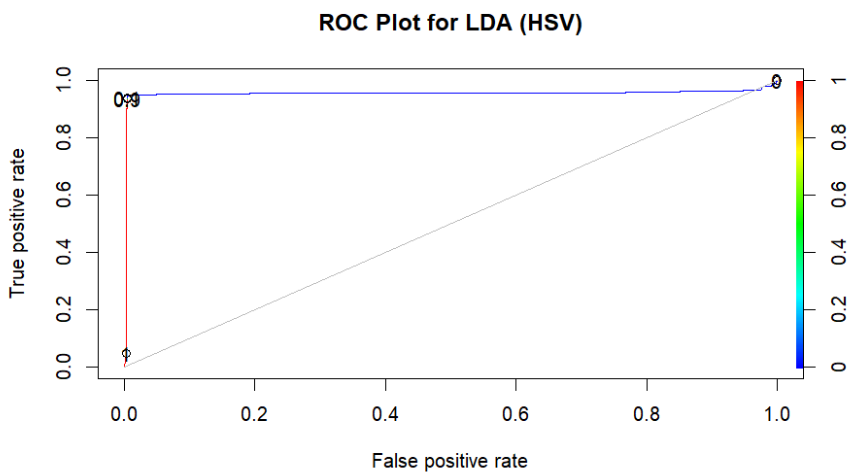


Figure 16. ROC Curve for LDA HSV Model

prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.994	0.996	0.067	0.004	0.067	0.997
0.10	0.994	0.996	0.066	0.004	0.066	0.997
0.15	0.994	0.996	0.066	0.004	0.066	0.997
0.20	0.994	0.996	0.065	0.004	0.065	0.997
0.25	0.994	0.996	0.065	0.004	0.065	0.997
0.30	0.994	0.996	0.064	0.004	0.064	0.997
0.35	0.994	0.996	0.064	0.004	0.064	0.997
0.40	0.994	0.996	0.064	0.004	0.064	0.997
0.45	0.994	0.996	0.064	0.004	0.064	0.997
0.50	0.994	0.996	0.064	0.004	0.064	0.997
0.55	0.994	0.996	0.064	0.004	0.064	0.997
0.60	0.994	0.996	0.063	0.004	0.063	0.997
0.65	0.994	0.995	0.063	0.005	0.063	0.997
0.70	0.994	0.995	0.063	0.005	0.063	0.997
0.75	0.994	0.995	0.062	0.005	0.062	0.997
0.80	0.994	0.995	0.062	0.005	0.062	0.997
0.85	0.994	0.995	0.062	0.005	0.062	0.997
0.90	0.994	0.995	0.061	0.005	0.062	0.997
0.95	0.994	0.995	0.060	0.005	0.061	0.997

Figure 17. Thresholds and Model Accuracy Results for LDA HSV Regression

LDA sees a substantial boost from an HSV transformation in all Accuracy metrics (other than AUC). FPR is nearly 1/3 of that in the RGB model, while the FNR is also less than half. Overall, even the best LDA model is not as good as the optimal Logistic Model, but the improvement from HSV transformation is noteworthy. Table 2 has the comparison between the two LDA models.

Table 2. LDA Models Comparison

Model	Accuracy	TPR	FPR	FNR	AUC	Prob-Threshold
RGB	0.984	0.989	0.185	0.011	0.9889	0.65
HSV	0.994	0.995	0.063	0.004	0.9537	0.60

Quadratic Discriminant Analysis (QDA)

For QDA, we will once again check interaction variables for Accuracy improvement. Without conducting a full analysis, we can check at a high level the accuracy between the two models.

```
Quadratic Discriminant Analysis

63241 samples
    3 predictor
    2 classes: 'Non_Tarp', 'Tarp'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 56917, 56917, 56917, 56917, 56918, 56917, ...
Resampling results:

  Accuracy   Kappa
  0.9895637  0.8523061
```

```
Quadratic Discriminant Analysis

63241 samples
    3 predictor
    2 classes: 'Non_Tarp', 'Tarp'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 56918, 56917, 56917, 56917, 56916, 56917, ...
Resampling results:

  Accuracy   Kappa
  0.9945605  0.9049272
```

Figure 18. RGB QDA Accuracy with Interactions (left) vs. without interactions (right)

Based on Figure 18, it seems better to use a no-interactions approach for QDA on the RGB color model. Accuracy takes a one percent hit when using interactions. This is significant, especially when lives are involved in any error rate. It is best to proceed with the no interactions model, as it has a relatively high accuracy of 99.46%.

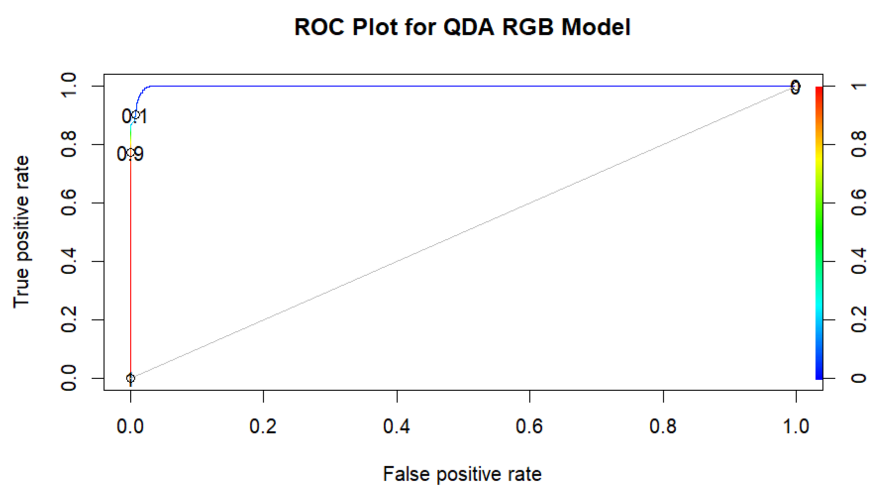


Figure 19. ROC Curve for QDA RGB Model

In the ROC curve, there is a significant bump in FPR in the early high TPR points. This can be explained further using Figure 20, in which an FNR of zero can be achieved at a high threshold of 0.60. The tradeoff is that more pixels are misclassified as Tarps (high FPR). So, no Tarps will be missed, but resources and time will be wasted in searching areas that do not have displaced people.

prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.992	1.000	0.255	0.000	0.255	0.996
0.10	0.993	1.000	0.226	0.000	0.226	0.996
0.15	0.993	1.000	0.210	0.000	0.210	0.997
0.20	0.994	1.000	0.201	0.000	0.201	0.997
0.25	0.994	1.000	0.192	0.000	0.192	0.997
0.30	0.994	1.000	0.185	0.000	0.185	0.997
0.35	0.994	1.000	0.180	0.000	0.180	0.997
0.40	0.994	1.000	0.172	0.000	0.172	0.997
0.45	0.994	1.000	0.167	0.000	0.167	0.997
0.50	0.995	1.000	0.161	0.000	0.161	0.997
0.55	0.995	1.000	0.155	0.000	0.155	0.997
0.60	0.995	1.000	0.152	0.000	0.152	0.997
0.65	0.995	0.999	0.148	0.001	0.148	0.997
0.70	0.995	0.999	0.144	0.001	0.144	0.997
0.75	0.995	0.999	0.138	0.001	0.138	0.997
0.80	0.994	0.998	0.132	0.002	0.132	0.997
0.85	0.990	0.994	0.114	0.006	0.114	0.995
0.90	0.990	0.993	0.099	0.007	0.099	0.995
0.95	0.988	0.990	0.060	0.010	0.061	0.994

Figure 20. Thresholds and Model Accuracy Results for QDA RGB Regression

As the main goal is to have a model that efficiently finds all displaced people, having a very high FPR of 0.152 will lead to an inefficient search, and therefore, an ineffective rescue strategy. This model serves as a good example of the FPR-FNR tradeoff dilemma present with this problem set.

As an alternative approach, an HSV transformation was fitted with a QDA model. Below are the results of the fit in Figure 21.

```
Call:
qda(x, grouping = y)

Prior probabilities of groups:
  Non_Tarp       Tarp
0.96802707 0.03197293

Group means:
               hue saturation     value
Non_Tarp 0.1385302  0.2455782 0.6415835
Tarp     0.5685617  0.1674437 0.8069510
```

Figure 21. QDA HSV Model Summary

As explained in LDA, the higher hue and value predictor coefficients for Tarps when compared to Non-Tarps is expected given that Tarps are bluer and brighter than their surroundings.

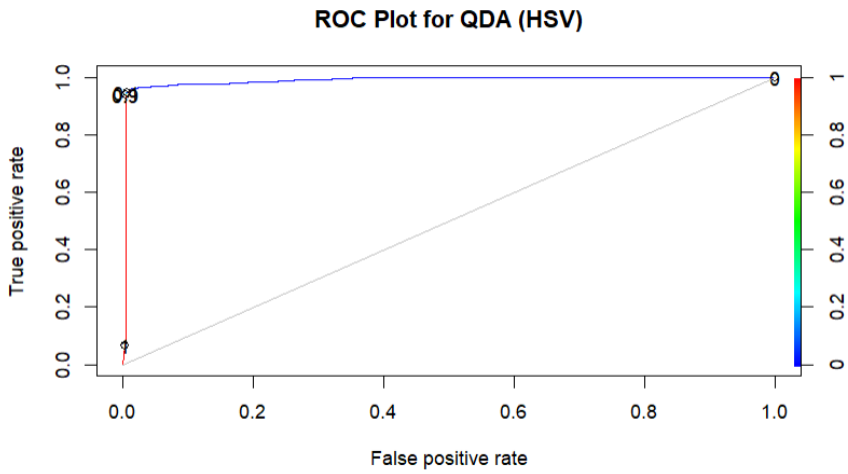


Figure 22. ROC Curve for QDA HSV Model

prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.994	0.995	0.062	0.005	0.062	0.997
0.10	0.994	0.995	0.060	0.005	0.061	0.997
0.15	0.994	0.995	0.059	0.005	0.059	0.997
0.20	0.994	0.995	0.058	0.005	0.058	0.997
0.25	0.994	0.995	0.057	0.005	0.058	0.997
0.30	0.994	0.995	0.057	0.005	0.058	0.997
0.35	0.994	0.995	0.057	0.005	0.057	0.997
0.40	0.994	0.995	0.057	0.005	0.057	0.997
0.45	0.994	0.995	0.056	0.005	0.057	0.997
0.50	0.994	0.995	0.056	0.005	0.057	0.997
0.55	0.993	0.995	0.056	0.005	0.057	0.997
0.60	0.993	0.995	0.056	0.005	0.056	0.997
0.65	0.993	0.995	0.056	0.005	0.056	0.997
0.70	0.993	0.995	0.055	0.005	0.056	0.997
0.75	0.993	0.995	0.055	0.005	0.056	0.997
0.80	0.993	0.995	0.054	0.005	0.054	0.997
0.85	0.993	0.995	0.053	0.005	0.054	0.996
0.90	0.993	0.995	0.053	0.005	0.053	0.996
0.95	0.993	0.994	0.052	0.006	0.052	0.996

Figure 23. Thresholds and Model Accuracy Results for QDA HSV Regression

Threshold selection is once again dependent on the FNR to FPR tradeoff present, although with the HSV transformation, there is a much more balanced set options available. Almost all FNRs are at .005, with FPRs also only slightly differing. A fair choice on the high end would be a 0.80 threshold. While the FNR is slightly higher than the RGB model, The FPR is less than half of the RGB approach. This once again brings up the dilemma of the desired solution. We want the lowest FNR possible, but within reasonable means of search. A low FPR allows for more efficient search, as there are less incorrect locations to check. The HSV transformation for the optimal QDA for now. Either way, the Logistic RGB color model with interactions is still the best option so far.

Table 3. QDA Models Comparison

Model	Accuracy	TPR	FPR	FNR	AUC	Prob-Threshold
RGB	0.995	1.000	0.152	0.000	0.9982	0.60
HSV	0.993	0.995	0.054	0.005	0.9892	0.80

K-Nearest Neighbors (KNN)

For KNN, we will need to test 2 different color models, RGB and HSV. We will avoid using interactions in the interest of computation time. We should also test many K-values. Let's check a range of 3 to 30 (a K of 1 or 2 will likely overtrain the data).

```
k-Nearest Neighbors

63241 samples
    3 predictor
    2 classes: 'Non_Tarp', 'Tarp'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 56918, 56917, 56917, 56917, 56916, 56917, ...
Resampling results across tuning parameters:

  k   Accuracy   Kappa
   3  0.9972486  0.9553989
   4  0.9971063  0.9532202
   5  0.9972803  0.9560524
   6  0.9972803  0.9561944
   7  0.9971063  0.9534581
   8  0.9970747  0.9528582
   9  0.9972328  0.9554017
  10  0.9969166  0.9502435
  11  0.9971063  0.9533755
  12  0.9971379  0.9538568
  13  0.9970747  0.9528270
  14  0.9969324  0.9505529
  15  0.9970114  0.9517747
  16  0.9970431  0.9523055
  17  0.9969324  0.9505107
  18  0.9970114  0.9518131
  19  0.9969956  0.9515545
  20  0.9970115  0.9518800
  21  0.9970115  0.9517633
  22  0.9970589  0.9525015
  23  0.9970114  0.9517829
  24  0.9970114  0.9516838
  25  0.9970431  0.9522086
  26  0.9970431  0.9522365
  27  0.9969324  0.9503953
  28  0.9969482  0.9506515
  29  0.9969956  0.9514285
  30  0.9969008  0.9499147

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was k = 5.
```

```
k-Nearest Neighbors

63241 samples
    3 predictor
    2 classes: 'Non_Tarp', 'Tarp'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 56918, 56917, 56917, 56917, 56916, 56917, ...
Resampling results across tuning parameters:

  k   Accuracy   Kappa
   3  0.9972170  0.9549669
   4  0.9972803  0.9560003
   5  0.9971538  0.9541315
   6  0.9971854  0.9545484
   7  0.9971696  0.9542434
   8  0.9971696  0.9542785
   9  0.9971854  0.9545982
  10  0.9972170  0.9549700
  11  0.9973119  0.9565521
  12  0.9971380  0.9536915
  13  0.9972012  0.9547957
  14  0.9973119  0.9565196
  15  0.9972328  0.9552157
  16  0.9972645  0.9557288
  17  0.9972328  0.9552050
  18  0.9972486  0.9554409
  19  0.9972803  0.9559117
  20  0.9972328  0.9551383
  21  0.9971854  0.9543680
  22  0.9971063  0.9530233
  23  0.9971221  0.9532773
  24  0.9971063  0.9530722
  25  0.9971063  0.9530549
  26  0.9971380  0.9535891
  27  0.9971379  0.9535568
  28  0.9971538  0.9538450
  29  0.9971221  0.9532829
  30  0.9971221  0.9532517

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was k = 14.
```

Figure 24. KNN Results using RGB Color Model (left) vs. HSV Color Model (right)

The RGB Color Model for KNN had an optimal K-value of 5 with an Accuracy of 99.725%. This is a very high Accuracy. A point of concern, however, is how the optimal k-value is very low. K=5 is prone for overtraining, so the Accuracy may need to be sacrificed to take a higher K-value. The HSV color model has an optimal K of k=14. This is a safe k-value, and the Accuracy is marginally better at 99.731%. Proceed with HSV color model.

KNN is a very flexible model, so the following results should be taken with a grain of salt, as the low bias will have a tradeoff of higher variance. The results will later be tested on a holdout set to determine the impact of flexibility.

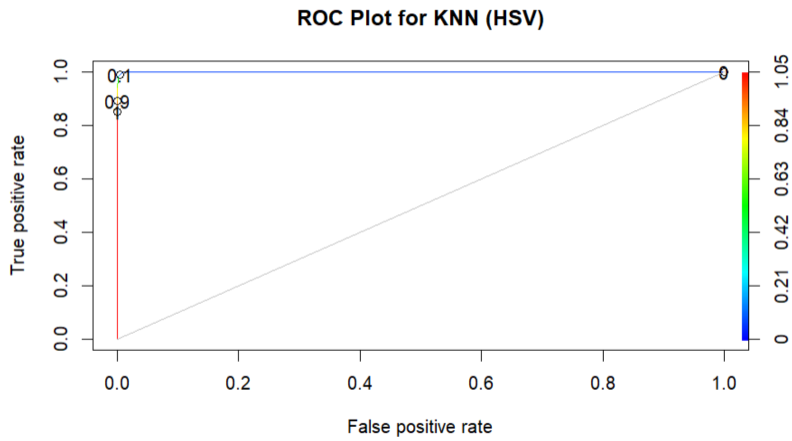


Figure 25. ROC Curve for KNN HSV Model

prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.994	1.000	0.180	0.000	0.180	0.997
0.10	0.996	1.000	0.135	0.000	0.135	0.998
0.15	0.996	1.000	0.107	0.000	0.107	0.998
0.20	0.996	1.000	0.107	0.000	0.107	0.998
0.25	0.997	1.000	0.087	0.000	0.087	0.998
0.30	0.997	0.999	0.073	0.001	0.073	0.998
0.35	0.997	0.999	0.073	0.001	0.073	0.998
0.40	0.997	0.999	0.060	0.001	0.060	0.999
0.45	0.997	0.999	0.050	0.001	0.050	0.999
0.50	0.997	0.998	0.039	0.002	0.039	0.999
0.55	0.997	0.998	0.039	0.002	0.039	0.999
0.60	0.997	0.998	0.029	0.002	0.029	0.999
0.65	0.997	0.998	0.024	0.002	0.024	0.998
0.70	0.997	0.998	0.024	0.002	0.024	0.998
0.75	0.997	0.997	0.018	0.003	0.018	0.998
0.80	0.996	0.997	0.016	0.003	0.016	0.998
0.85	0.996	0.997	0.016	0.003	0.016	0.998
0.90	0.996	0.996	0.015	0.004	0.016	0.998
0.95	0.991	0.990	0.001	0.010	0.010	0.995

Figure 26. Thresholds and Model Accuracy Results for KNN HSV Regression

The ROC curve in Figure 25 indicates that a very high TPR can be achieved at a very low FPR. The threshold table of Figure 26 confirms this, as an ideal probability of 0.70 results in very Accuracy. The performance of KNN on HSV color model is the best so far. It has equivalent FNR and Accuracy to the optimal Logistic Regression model, but with half the FPR. At k=14, this model has the best fit so far on training data. The only downside is the real possibility of overtraining that comes with this flexible of a model type.

Table 4. KNN Model Overview

Model	Accuracy	TPR	FPR	FNR	AUC	Prob-Threshold
HSV	0.997	0.998	0.024	0.002	0.9998	0.70

Penalized Logistic Regression (Elastic-Net)

Using the “glmnet” functions with an alpha of 0.5 fits an elastic regression (between lasso and ridge) to the predictors. The first case of this will be on the RGB color data. When doing so with the interaction terms included (as this was the optimal Logistic Regression approach), a rather strange result can be seen in Figure 27.

```
glmnet

63241 samples
    3 predictor
    2 classes: 'Non_Tarp', 'Tarp'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 56917, 56918, 56917, 56917, 56916, 56916, ...
Resampling results across tuning parameters:

  lambda        Accuracy   Kappa
  1.000000e-05  0.9950981  0.9182047
  2.335721e-05  0.9949242  0.9144809
  5.455595e-05  0.9946712  0.9088951
  1.274275e-04  0.9944973  0.9044202
  2.976351e-04  0.9937857  0.8898826
  6.951928e-04  0.9919672  0.8521276
  1.623777e-03  0.9882197  0.7680591
  3.792690e-03  0.9829857  0.6294913
  8.858668e-03  0.9722806  0.2285025
  2.069138e-02  0.9680271  0.0000000
  4.832930e-02  0.9680271  0.0000000
  1.128838e-01  0.9680271  0.0000000
  2.636651e-01  0.9680271  0.0000000
  6.158482e-01  0.9680271  0.0000000
  1.438450e+00  0.9680271  0.0000000
  3.359818e+00  0.9680271  0.0000000
  7.847600e+00  0.9680271  0.0000000
  1.832981e+01  0.9680271  0.0000000
  4.281332e+01  0.9680271  0.0000000
  1.000000e+02  0.9680271  0.0000000

Tuning parameter 'alpha' was held constant at a value of 0.5
Accuracy was used to select the optimal model using the largest value.
The final values used for the model were alpha = 0.5 and lambda = 1e-05.
```

```
glmnet

63241 samples
    3 predictor
    2 classes: 'Non_Tarp', 'Tarp'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 56917, 56917, 56917, 56917, 56917, 56917, ...
Resampling results across tuning parameters:

  lambda        Accuracy   Kappa
  1.000000e-05  0.9932322  0.89251773
  2.335721e-05  0.9932322  0.89251773
  5.455595e-05  0.9932322  0.89251773
  1.274275e-04  0.9932322  0.89251773
  2.976351e-04  0.9932638  0.89307230
  6.951928e-04  0.9933271  0.89422467
  1.623777e-03  0.9933903  0.89532540
  3.792690e-03  0.9933271  0.89410509
  8.858668e-03  0.9927578  0.88397498
  2.069138e-02  0.9854525  0.73737795
  4.832930e-02  0.9682484  0.18824200
  1.128838e-01  0.9662086  0.05889365
  2.636651e-01  0.9680271  0.00000000
  6.158482e-01  0.9680271  0.00000000
  1.438450e+00  0.9680271  0.00000000
  3.359818e+00  0.9680271  0.00000000
  7.847600e+00  0.9680271  0.00000000
  1.832981e+01  0.9680271  0.00000000
  4.281332e+01  0.9680271  0.00000000
  1.000000e+02  0.9680271  0.00000000

Tuning parameter 'alpha' was held constant at a value of 0.5
Accuracy was used to select the optimal model using the largest value.
The final values used for the model were alpha = 0.5 and lambda = 0.001623777.
```

Figure 27. Elastic Net Logistic Regression Results using RGB Color Model (left) vs. HSV Color Model (right)

Accuracy increases as we lower lambda to almost zero. This seems to indicate that a penalized regression will not work out better than normal logistic regression on RGB color model. Removing interactions also seems to favor an extremely small lambda of nearly zero. Repeatedly decreasing the lower bound always results in the lowest value for lambda being picked as optimal Accuracy. As a result, we should look to use HSV data to see if a penalized approach will work out best or not.

It appears a small lambda of 0.0016 is the ideal value for Accuracy in the HSV color model. This too is a very small value. Overall, a penalized approach doesn't seem useful, but we will proceed for results, nonetheless.

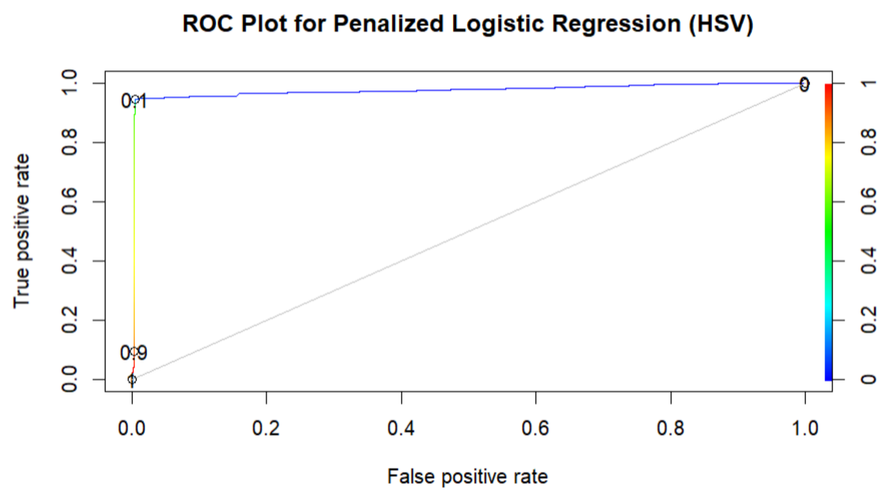


Figure 28. ROC Curve for Penalized Logistic Regression HSV Model

prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.967	0.996	0.937	0.004	0.937	0.983
0.10	0.967	0.996	0.906	0.004	0.906	0.983
0.15	0.969	0.996	0.858	0.004	0.858	0.984
0.20	0.972	0.996	0.760	0.004	0.760	0.986
0.25	0.977	0.996	0.606	0.004	0.606	0.988
0.30	0.984	0.996	0.396	0.004	0.396	0.992
0.35	0.988	0.996	0.243	0.004	0.243	0.994
0.40	0.992	0.996	0.140	0.004	0.140	0.996
0.45	0.993	0.996	0.101	0.004	0.101	0.996
0.50	0.993	0.996	0.083	0.004	0.083	0.997
0.55	0.994	0.996	0.077	0.004	0.077	0.997
0.60	0.994	0.996	0.072	0.004	0.072	0.997
0.65	0.994	0.996	0.069	0.004	0.069	0.997
0.70	0.994	0.996	0.068	0.004	0.068	0.997
0.75	0.994	0.996	0.066	0.004	0.066	0.997
0.80	0.994	0.996	0.064	0.004	0.064	0.997
0.85	0.994	0.995	0.059	0.005	0.060	0.997
0.90	0.993	0.995	0.055	0.005	0.056	0.997
0.95	0.986	0.987	0.052	0.013	0.054	0.993

Figure 29. Thresholds and Model Accuracy Results for Penalized Logistic Regression (HSV)

Penalized models for this data set seem to favor very small lambda values, or just no penalty at all. A small lambda penalty of .0016 seems to improve performance of an HSV logistic model by a negligible amount. An RGB model would result in a lambda of nearly zero, which is effectively a normal Logistic Regression. Overall, it appears best to avoid a penalized model, as the optimal Logistic Regression model outperforms it in every Accuracy metric.

Table 5. Penalized Logistic Regression Model Overview

Model	**Accuracy**	**TPR**	**FPR**	**FNR**	**AUC**	**Prob-Threshold**
HSV	0.994	0.996	0.064	0.004	0.9741	0.80

Random Forest Model

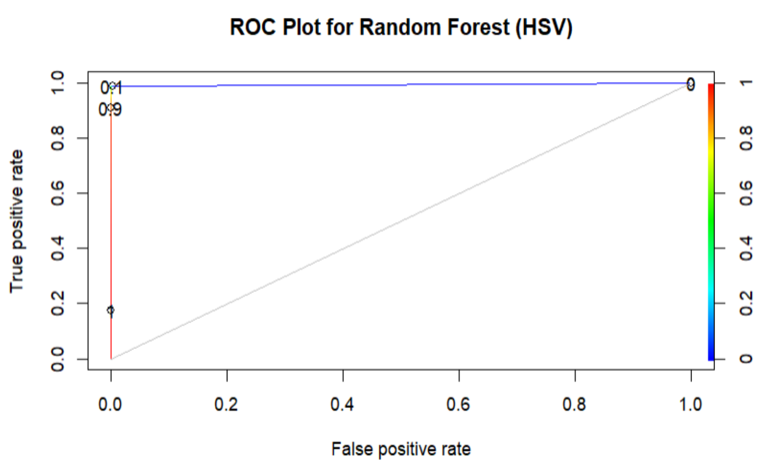
For Random Forest models, the two key parameters are the number of predictors to test at each split (mtry) and the number of total trees to have in the model (ntree). The ntree parameter, by default in R, is 500, and is typically less important than the mtry parameter. In order to efficiently train models, the default tree count was kept, while the number of predictors at each split was varied between 1 and 3 to determine the best fit on the training data.

```
Call:
 randomForest(x = x, y = y, mtry = param$mtry)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 1

        OOB estimate of  error rate: 0.29%
Confusion matrix:
         Non_Tarp Tarp class.error
Non_Tarp    61145   74 0.001208775
Tarp          110 1912 0.054401583
```

```
Call:
 randomForest(x = x, y = y, mtry = param$mtry)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 1

        OOB estimate of  error rate: 0.3%
Confusion matrix:
         Non_Tarp Tarp class.error
Non_Tarp    61135   84 0.001372123
Tarp          105 1917 0.051928783
```

Figure 30. RGB RF Model (left) vs. HSV RF Model (right)

Figure 30 above indicates that the models do not differ by much (0.29% vs. 0.30 % Out of Bag error rate). Both models favor using only one predictor at each split (mtry=1). The accuracy achieved by the models was 99.69% for RGB and 99.71% for HSV. The difference between the models is marginal, so we will choose the HSV model due to the predictors being less correlated.



prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.994	1.000	0.188	0.000	0.188	0.997
0.10	0.995	1.000	0.141	0.000	0.141	0.998
0.15	0.996	1.000	0.119	0.000	0.119	0.998
0.20	0.996	1.000	0.101	0.000	0.101	0.998
0.25	0.997	0.999	0.091	0.001	0.092	0.998
0.30	0.997	0.999	0.084	0.001	0.084	0.998
0.35	0.997	0.999	0.074	0.001	0.074	0.998
0.40	0.997	0.999	0.064	0.001	0.064	0.998
0.45	0.997	0.999	0.057	0.001	0.057	0.998
0.50	0.997	0.999	0.050	0.001	0.050	0.998
0.55	0.997	0.998	0.047	0.002	0.047	0.998
0.60	0.997	0.998	0.041	0.002	0.041	0.998
0.65	0.997	0.998	0.032	0.002	0.032	0.998
0.70	0.997	0.998	0.029	0.002	0.029	0.998
0.75	0.997	0.998	0.025	0.002	0.025	0.998
0.80	0.997	0.997	0.021	0.003	0.022	0.998
0.85	0.996	0.997	0.019	0.003	0.019	0.998
0.90	0.996	0.996	0.015	0.004	0.016	0.998
0.95	0.994	0.994	0.014	0.006	0.016	0.997

Figure 31. ROC Curve and Thresholds/ Model Accuracy Results for Random Forest Model (HSV)

Figure 31 indicates that the model performs very strongly on the training data, with better FPR (0.025) and matching FNR (0.002) to the best Log Regression model. At an optimal threshold of 0.75, the FPR is very close, but slightly under that of KNN. This model rivals KNN for the best overall performance, depending on how they each perform on the holdout set.

Table 6. Random Forest Model Overview

Model	Accuracy	TPR	FPR	FNR	AUC	Prob-Threshold
HSV	0.997	0.998	0.025	0.002	0.9945	0.75

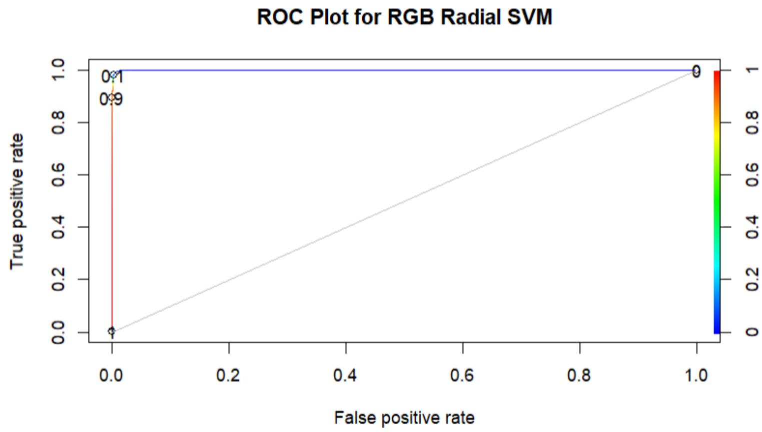
Support Vector Machine

The "**tune**" function in R allows us to perform 10-fold cross validation on a range of parameters, dependent on the choice of kernel. For this study, 3 different SVM model types were tested: linear, radial, and polynomial. All 3 use a "cost" parameter, which sets the margins under which observations qualify as support vectors in the model. Radial and polynomial kernels have an additional parameter (gamma and degree, respectively) that help determine the shape/size of the decision boundary. SVM, especially for radial kernels, is very computationally taxing. It takes hours to run 10-cross validation on multiple cost and gamma/degree values, limiting the testing windows for the parameters. See below a table of the initial results of the 3 kernels on an RGB color model.

Table 7. SVM Kernel Optimal Result Comparison

Kernel	Optimal Parameters	Support Vectors Used	Error
Linear	cost = 1000	717	0.0046
Radial	cost = 100, gamma = 5	404	0.0026
Polynomial	cost = 1000, degree = 3	652	0.0042

The linear and polynomial kernels use more support vectors than the radial kernel, yet they both have a higher cost than the radial method. A higher cost equates to narrower margins, meaning more observations fall close to the decision boundary of the linear and poly kernels. In terms of error, the radial model outshines the other two as the preferred model for further analysis. As a side note, an HSV model was attempted on the radial kernel as well, although it proved too computationally taxing and would not finish running after many hours.



prob_threshold	Accuracy	TPR	FPR	FNR	Dist	F1
0.05	0.994	1.000	0.192	0.000	0.192	0.997
0.10	0.996	1.000	0.100	0.000	0.100	0.998
0.15	0.997	0.999	0.082	0.001	0.082	0.998
0.20	0.997	0.999	0.069	0.001	0.069	0.998
0.25	0.997	0.999	0.063	0.001	0.063	0.999
0.30	0.997	0.999	0.056	0.001	0.056	0.999
0.35	0.997	0.999	0.051	0.001	0.051	0.999
0.40	0.997	0.999	0.049	0.001	0.049	0.999
0.45	0.997	0.999	0.045	0.001	0.046	0.999
0.50	0.997	0.999	0.042	0.001	0.042	0.999
0.55	0.997	0.999	0.039	0.001	0.039	0.999
0.60	0.997	0.998	0.036	0.002	0.036	0.999
0.65	0.997	0.998	0.032	0.002	0.032	0.999
0.70	0.997	0.998	0.029	0.002	0.029	0.999
0.75	0.997	0.998	0.029	0.002	0.029	0.999
0.80	0.997	0.998	0.024	0.002	0.024	0.999
0.85	0.997	0.998	0.023	0.002	0.023	0.999
0.90	0.997	0.998	0.020	0.002	0.020	0.998
0.95	0.992	0.992	0.006	0.008	0.012	0.996

Figure 32. ROC Curve and Thresholds/ Model Accuracy Results for Radial SVM on RGB color model

The ROC curve for the radial SVM is very promising, being almost perfect on the early high TPR values showing very little change in FPR. At the ideal threshold of 0.80, the FPR to FNR tradeoff is very strong for this model. The 0.002 FNR matches that of the best methods tested so far, while the 0.024 FPR ties KNN for the lowest.

Optimal Performance for Each Method (Training)

Given that the **caret** package from R was used, the default metric for cross-validation is going to be Accuracy for any classification type problem. The problem here is a binary classification of identifying displaced individuals, so the default was used in all models. Model accuracy, specifically how it relates to the FNR and FPR tradeoff, was the key factor in determining the optimal model for each method.

Table 8. Comparison of 7 Models: Training Results

Model	Key Parameters	Accuracy	TPR	FPR	FNR	AUC	Prob-Threshold
Logistic Regression: RGB w/ Interactions	Interactions between Colors Included	0.996	0.998	0.055	0.002	0.9996	0.70
LDA: HSV	HSV Color Model Transformation	0.994	0.995	0.063	0.004	0.9537	0.60

QDA: HSV	HSV Color Model Transformation	0.993	0.995	0.054	0.005	0.9892	0.80
KNN: HSV	K = 14, HSV Color Model Transformation	0.997	0.998	0.024	0.002	0.9998	0.70
Penalized Logistic Regression: HSV	λ = 0.0016, HSV Color Model Transformation	0.994	0.996	0.064	0.004	0.9741	0.80
Random Forest: HSV	mtry = 1, ntree=500, HSV Color Model Transformation	0.997	0.998	0.025	0.002	0.9945	0.75
Support Vector Machine: Radial RGB	cost = 100, gamma = 5, Radial kernel	0.997	0.998	0.024	0.002	0.9997	0.80

Exploratory Data Analysis (Holdout)

As mentioned, numerous times throughout the report, a holdout set will be needed to test the best trained models. The holdout set for this report is multiple files containing pixel data for either Blue Tarp image points or Non-Blue-Tarp image points. There were many columns in the files, referring to different metrics, such as geographical location, pixel id, and RGB metrics. Of course, for the purposes of our analysis, we only care for the three RGB color columns. These columns were labelled B1, B2, and B3. To determine which referred to Red, Blue, and Green respectively, some intuition was used. Taking one of the Blue Tarp data files (orthovnir069_ROI_Blue_Tarps.txt), we get the average value of columns B1, B2, and B3. The values were 122.6, 138.4, and 168.5, respectively. Our prior knowledge of Blue Tarp data indicates that the order of prominence for the colors is first Blue, then Green, then Red. Based on this, we conclude that columns B1, B2, and B3 are Red, Green, and Blue, respectively. After combining the seven Non-Tarp and Tarp data sets, there were 2,004,177 total rows in the final holdout set, which means predictive fits may take some extra time to complete.

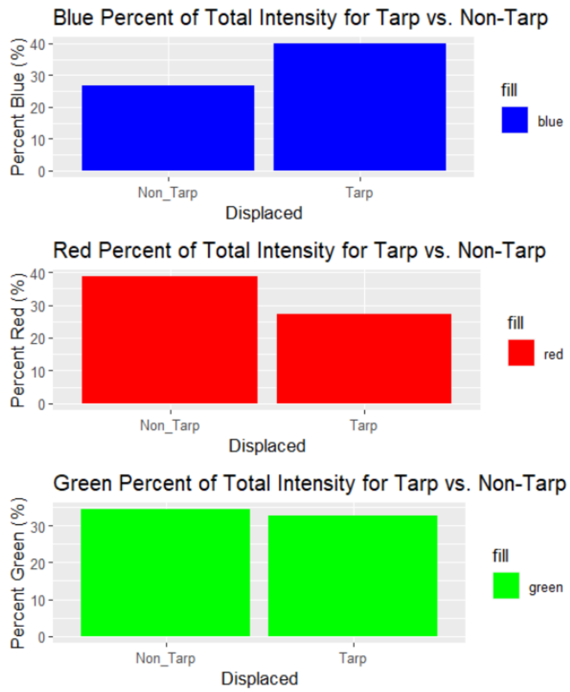


Figure 33. Tarp vs. Non-Tarp Color Intensity in Holdout Set

Figure 33 shows that the holdout set has a similar percentage distribution of color intensity between Non-Tarps and Tarps as the training data (Figure 2). The most notable difference between the two data sets is that the holdout Non-Tarps seem to have a higher percent of their total intensity in Reds than the training data did. Essentially, Red is more prevalent in the holdout Non-Tarps than the training Non-Tarps.

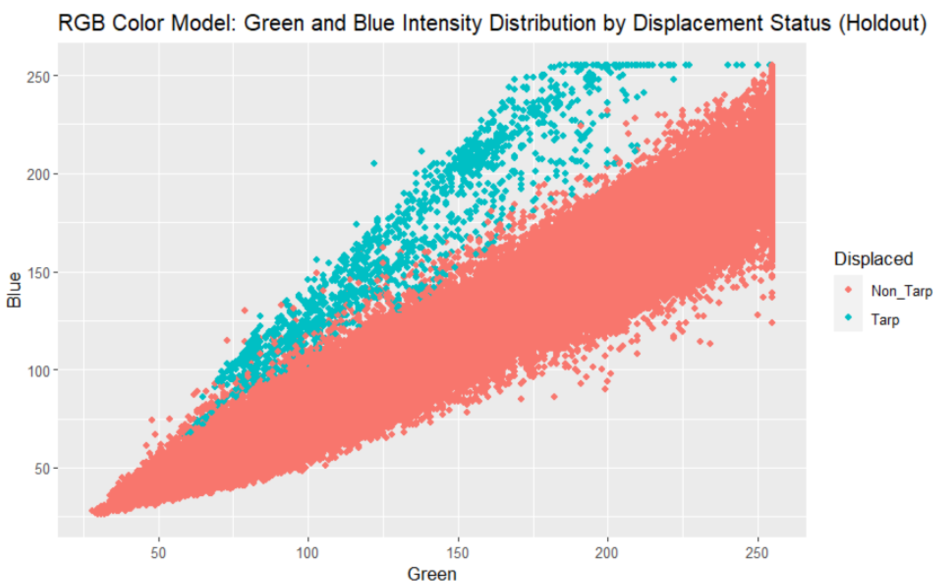


Figure 34. Green and Blue Correlation for Tarps and Non-Tarps in Holdout Set

Figures 3 and 34 share a similar finding in that we don't expect intensity to play as large a role as color difference in the holdout set, much like the training set. Let's confirm by looking at the graph in Figure 4 applied to the holdout set.

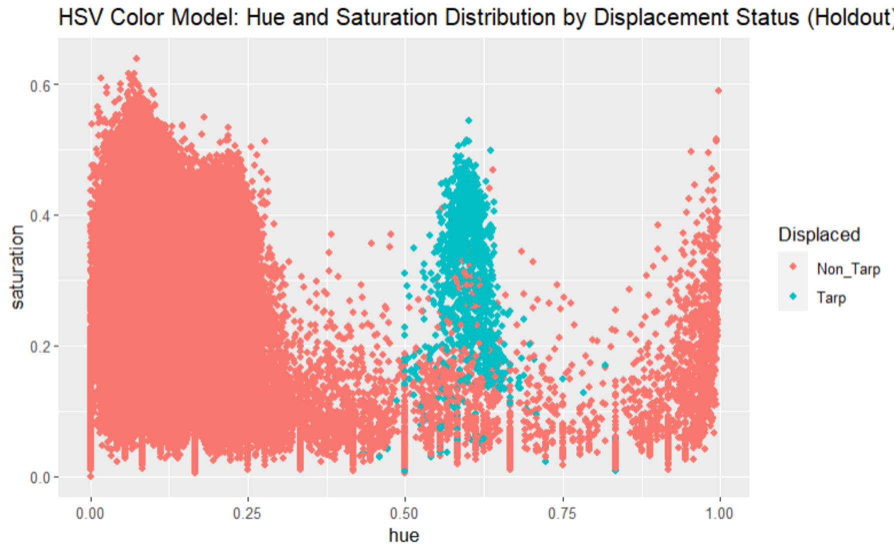


Figure 35. Intensity and color impact on classification in Holdout Set

For the training set, we determined that saturation would not play much of a role in classification. Here, in Figure 35, we see that saturation may be important given how many Non-Tarps are showing hues similar to Tarps. Blue hues with high saturation are the most identifiable Tarps.

Optimal Performance for Each Method (Holdout)

Much like on the training set, the tradeoff between FPR and FNR will be the key factor in determining which model performs best on the holdout test set. Parameters were held fixed to how they were optimized in the model training phase.

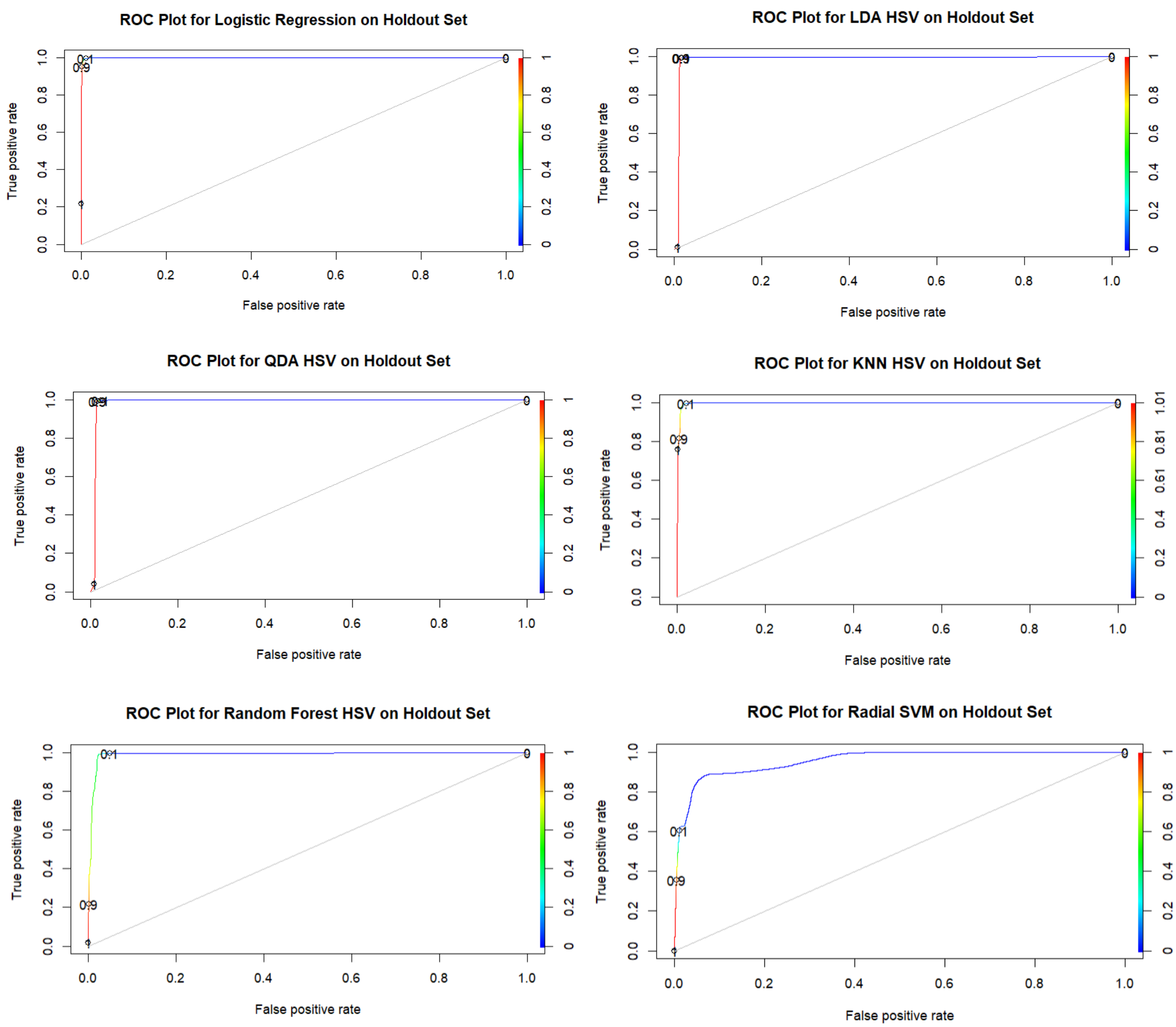


Figure 36. ROC curves for 6 Final Models. Tested on Holdout Set. Models are RGB Logistic Regression with interactions (top left), HSV LDA (top right), HSV QDA (middle left), HSV KNN (middle right), HSV Random Forest (bottom left), and RGB Radial SVM (bottom left)

The Logistic Regression, LDA and QDA ROC curves appear to hold well on the holdout set. Logistic Regression in particular looks almost identical to how it did on training data (Figure 7). KNN appears to have a higher FPR early on than it did during training. This is early evidence of possible overtraining from the KNN model. The Random Forest and SVM learning models have very strange ROC curves. While they outperform random guessing, the Tradeoff between TPR and FPR is heavy in the early stages. SVM reaches its plateau at a relatively high FPR.

When conducting Penalized Logistic Regression, we determined that the near zero lambda value indicates that a penalty is not necessary. Hence, the ROC curve for this method is omitted from consideration. The results table also omits Penalized Logistic Regression, as a normal logistic regression is preferred. For a high-level summary, the accuracy of the HSV Penalized Logistic

Regression model was 98.39%, with a ridiculous FPR of 96.50% (Specificity is 0.035). See below the results on holdout for all other optimized methods.

Table 9. Comparison of 7 Models: Holdout Results

Model	Key Parameters	Accuracy	TPR	FPR	FNR	AUC	Prob-Threshold
Logistic Regression: RGB w/ Interactions	Interactions between Colors Included	0.997	0.997	0.023	0.003	0.9996	0.70
LDA: HSV	HSV Color Model Transformation	0.983	0.983	0.008	0.017	0.9849	0.60
QDA: HSV	HSV Color Model Transformation	0.983	0.983	0.006	0.018	0.9895	0.80
KNN: HSV	K = 14, HSV Color Model Transformation	0.992	0.993	0.037	0.007	0.9976	0.70
Random Forest: HSV	mtry = 1, ntree=500, HSV Color Model Transformation	0.993	0.998	0.630	0.002	0.9903	0.75
Support Vector Machine: Radial RGB	cost = 100, gamma = 5, Radial kernel	0.991	0.995	0.6006	0.005	0.9572	0.80

Conclusions

Observation 1

The best performing models in cross-validation were KNN on HSV color model, Random Forest on HSV color model, and Radial SVM on RGB color model (no interactions included for each). See Table 8 for the results and optimal parameters for each method. They each had comparable low FNR's of 0.002, while having FPR's on or just below 0.025. Logistic Regression on the RGB color model with interactions included also performed relatively well in terms of TPR and FNR but had over double the FPR (0.055). As discussed, many times in this report, the FPR-FNR tradeoff is key to determine the best model. FNR tells us how many blue tents we missed, while FPR tells us how much search effort is being wasted.

However, the story completely changes when looking at the results for the holdout set. From Table 9, we see that KNN, Random Forest, and Radial SVM all underperformed from their training results. Random Forest and Radial SVM struggled greatly with over-identifying Tarps. Each had an FPR of over 60%, essentially marking many Non-Tarps as Tarps. This level of misclassification is too much wasted effort. KNN didn't underperform to the same extent as RF and SVM, but it still fell far below the best model, Logistic Regression. Logistic Regression (RGB w/ interactions) performed even better on the holdout set than it did in training. A low FNR of 0.003 combined with the lowest observed FPR of 0.023 is the best tradeoff amongst all holdout results, and crown Logistic Regression as the best performing model overall in the study.

Observation 2

The results mentioned in Observation 1 could be explained by the types of models used. KNN, even at k=14, is a relatively flexible model, and is susceptible to overfitting. While it was still the second-best performing model on the holdout sets, it tripled its training FNR. This is evidence of overtraining. Likewise, Random Forest and SVM are learning models that can struggle with overtraining. Furthermore, computational limitations didn't allow for more broad testing of parameters for these models. With more training capacity, these models may have worked better, but the result was very high FPR for these models on the holdout set. This means the models lean towards classifying Tarps at a disproportionate rate.

Logistic Regression, LDA, and QDA are not as flexible. Logistic Regression and LDA, specifically, are high bias models that don't provide much flexibility. While they can't capture more complex relationships in the data, they are less susceptible to overtraining. With this being a classification problem, Logistic Regression had the advantage over LDA. Lack of overtraining mixed with classification resulted in consistent performance from Logistic Regression from training to testing.

Observation 3

Based on the above two observations, the RGB Logistic Regression model with interactions included is the best approach for identifying displaced individuals in the disaster relief search effort. Having very low FNR in both the training and holdout set means that we greatly reduce the chance of missing a Tarp observation. This, combined with the lowest holdout FPR, means that the search effort will likely be the most efficient with this Logistic Model. Less time and resources will be wasted on checking locations that don't have Blue Tarps. As discussed throughout this report, the FNR and FPR tradeoff is key in determining the best model, given the nature of the problem is survival identification. Our wiggle room for missing Tarps and wasting resources cannot be taken lightly.

With more training and computational power, the more taxing models (KNN, Random Forest, SVM) could prove more useful. In our study, however, their flexibility proved to overtrain and perform poorly on the holdout set. Overall, KNN still performed relatively well, having the second lowest FPR. The FNR was double that of the Logistic Regression, so we favor the latter. Being able to test a larger range of parameters for all methods could serve to benefit the models fits, so more computational power and resources would be ideal.

Observation 4

The key metrics in Tables 8 and 9 are Accuracy, TPR, FPR and FNR. TPR and FNR are compliments in getting total Tarps in the data, so only FNR is needed for decision making purposes. In the context of disaster relief, TPR represents the proportion of Tarps that were correctly predicted by the model. For example, in Table 9 we see the Logistic Regression model correctly predicted 99.7% of Tarps. Conversely, FNR represents the Tarps that were predicted as Non-Tarps, meaning they were misclassified and possibly missed survivors. A key goal is to have a low FNR/high TPR, as this means we missed less survivors. This, however, should not come at an extreme cost of FPR. FPR represents the proportion of Non-Tarps that are predicted as Tarps. Essentially, this is the wasted effort during search in the disaster relief effort. While having an FNR of zero would be ideal, it shouldn't be accomplished at the expense of high FPR and wasted effort, as survivors have limited time for rescue.

Accuracy is simply the proportion of total observations classified correctly by the model (Tarps that are truly Tarps, Non-Tarps that truly Non-Tarps). A probability threshold is used to determine the exact FPR and FNR tradeoff we want to implement in the holdout test. This threshold is selected based on the training results, so may differ from what the best choice is for the holdout. This value was not changed based on holdout results because, in practice, we wouldn't know the true classifications in raw image data given for search efforts. That's why the trained model with selected parameters and thresholds should be reliable. AUC tells us, at a high level, how well the model separates the Tarps and Non-Tarps. A higher AUC is preferred, as it means the model distinctly separates Tarps from Non-Tarps. This is yet another reason the Logistic Regression model performance is the best.

Observation 5

An HSV transformation significantly reduces correlation between predictors. One key issue with an RGB color model was the high correlation between the three colors in the data set. This is due to key color metrics, such as vibrancy, brightness, and shade, all being linked to the scale of the three colors, not just the ratio between them. For example, 60-60-60 R-G-B is a very different color from 180-180-180 (the former is light grey, while the latter is nearly black). An HSV transformation separates these color integers into less correlated components that more easily distinguish color, vibrancy, and brightness. Because of this, an HSV transformation plays well with models that would otherwise be blown out by the highly correlated RGB color values.

Another interesting observation is how the cylindrical nature of the HSV color model plays a role in classification. We expect higher hue values to favor Blue Tarps because Blue has a higher hue than Green and Red. In Figure 4, this mostly holds true, however, there are many Non-Tarps at a high hue value that fit almost perfectly next to the lowest hue value Non-Tarps. This is an example of how the cylindrical nature of the HSV model wraps the hue in a circular form, rather than straight 0 to 1 scale. If the hue values were shifted by a constant (say 0.20), and we forced a rule to have any hue over 1 to be reset to 0 plus the remainder added, then the data would show an almost perfect split in Non-Tarp and Tarp hues.

Observation 6

All optimal methods performed well in training, with the main distinguishing factor being the FNR-FPR tradeoff and how it would impact the rescue effort. Clearing an Accuracy of 99% for all models is impressive for an initial 10-cross validation run. The only model that can be considered redundant would be Penalized Logistic Regression, simply because the near-zero lambda values indicate that normal Logistic Regression was performing just fine. Even comparing results between an HSV Logistic Regression and the best HSV Penalized Logistic Regression (Tables 1 and 5, respectively) shows almost identical results in all Accuracy metrics. Overall, a Penalized Logistic Regression appears redundant in the presence of the other methods and was therefore removed from consideration on holdout comparison.

Furthermore, promising learning methods such as Random Forest and Support Vector Machine proved to overclassify Tarps in the holdout data. With more time and power, these models can likely be tuned to produce more selective results, as they usually perform well in real world classification problems. A good example of this is the inability to train a Radial SVM using HSV color data. The model ran for many hours and still never completed. Having more computational power could have led to a Radial SVM that was tuned to produce better test results.