Fast and Fiery-ous : An Examination Into the Likelihood of Forest Fires.

Final Group Project Report

Group 40

Tanisha Kadia - Shiyu Jiang - Sohbat Sandhu - Valentina Ramirez

In [3]:

```
### We load this libaries because they provide us with the resources (ie. functions) we need to do our analysis

#tidyverse make the code more readable by. setting each variable a column and each observation a row, each cell a single measurement
library(tidyverse)

#library(repr) is used to create readable viewable image representations of data
library(repr)

#library(rvest) helps us scrape data from web pages
library(rvest)

# tidymodel is a list of packages for modeling and statistical analysis that share the same structure with tidyverse.
library(tidymodels)

#stringr provides cohesive set of functions. Argument names and functions are consistent designed to be more efficiency in using strings
library(stringr)

options(repr.matrix.max.rows = 10)
options(repr.plot.width = 10, repr.plot.height = 10, repr.plot.res = 100)
```

```
── Attaching packages ──────────────────────────────────────── tidyverse 1.3.0 ──

✓ ggplot2 3.3.2     ✓ purrr   0.3.4
✓ tibble  3.0.3     ✓ dplyr   1.0.2
✓ tidyr   1.1.2     ✓ stringr 1.4.0
✓ readr   1.3.1     ✓ forcats 0.5.0

Warning message:
"package 'ggplot2' was built under R version 4.0.1"
Warning message:
"package 'tibble' was built under R version 4.0.2"
Warning message:
"package 'tidyr' was built under R version 4.0.2"
Warning message:
"package 'dplyr' was built under R version 4.0.2"
── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()

Warning message:
"package 'rvest' was built under R version 4.0.2"
Loading required package: xml2
```

```
Attaching package: ‘rvest’


The following object is masked from ‘package:purrr’:

    pluck


The following object is masked from ‘package:readr’:

    guess_encoding


Warning message:
“package ‘tidymodels’ was built under R version 4.0.2”
── Attaching packages ─────────────────────────────────────── tidymodels 0.1.1 ──

✓ broom     0.7.0      ✓ recipes   0.1.13
✓ dials     0.0.9      ✓ rsample   0.0.7
✓ infer     0.5.4      ✓ tune      0.1.1
✓ modeldata 0.0.2      ✓ workflows 0.2.0
✓ parsnip   0.1.3      ✓ yardstick 0.0.7

Warning message:
“package ‘broom’ was built under R version 4.0.2”
Warning message:
“package ‘dials’ was built under R version 4.0.2”
Warning message:
“package ‘infer’ was built under R version 4.0.3”
Warning message:
“package ‘modeldata’ was built under R version 4.0.1”
Warning message:
“package ‘parsnip’ was built under R version 4.0.2”
Warning message:
“package ‘recipes’ was built under R version 4.0.1”
Warning message:
“package ‘tune’ was built under R version 4.0.2”
Warning message:
“package ‘workflows’ was built under R version 4.0.2”
Warning message:
“package ‘yardstick’ was built under R version 4.0.2”
── Conflicts ──────────────────────────────────────── tidymodels_conflicts() ──
x scales::discard() masks purrr::discard()
x dplyr::filter()   masks stats::filter()
x recipes::fixed()  masks stringr::fixed()
x dplyr::lag()      masks stats::lag()
x rvest::pluck()    masks purrr::pluck()
x yardstick::spec() masks readr::spec()
x recipes::step()   masks stats::step()
```

Introduction

Revelant Background Information:

Forest Fires are massive, unplanned, uncontrolled natural disasters that burn combustible vegetation. Forest Fires can kill many organisms and burn soil so that an environment becomes debilitated. Many geological factors impact the likelihood of a devastating wildfire taking place. These variables can be used to answer the predictive classification question of if a fire will occur or not based on the weather conditions.

The level of ecological devastation a Forest Fire causes depends on how deep it burns; the deeper into the soil table the fire burns, the longer it will take for the ecosystem to recover from the natural disaster. The top layer of soil consists of fine fuels (biomass litter) such as fallen leaves or grass, detritus, and other biological waste. The layer of moderate depth is the duff layer. Duff is the biological material that accumulates and is buried below the fine fuel top layer of soil. This layer of biomass does not accumulate when an ecosystem experiences frequent forest fires. A buildup of the duff layer is especially problematic because if there is a forest fire, this layer smolders for a long time even after the massive fire is put out. This prolonged smoldering transfers heat to trees and soil in the environment, kills much of the life, and damages the soil's nutritional properties beyond repair. If a fire reaches the duff soil level, that ecosystem will be devastated with a massive loss of life and soil, and likely have to undergo primary succession (completely restart ecological growth because all the life was killed) after the forest fire natural disaster. The moisture and flammability of the duff layer is extremely important in indicating how a fire will occur and how devastating it will be. After the duff layer is a deeper layer of soil where buried and decomposed biomass is found.

Many weather factors also decrease the likelihood of a forest fire taking place. If the biomass has more moisture in it (water) then it will be harder for it to be ignited.

Therefore, factors that increase moisture like rain and humidity mitigate the risk of a wildfire.

Using The Duff Moisture Code and Relative Humidity to predict whether a wildfire will happen or not allows an application of data analysis and classification to answer real-world problems. The Duff Moisture Code indicates how flammable the essential duff soil layer is, and the humidity indicates how much moisture there is in the atmosphere — both providing insight on the likelihood of a wildfire happening.

What are we trying to ask?

Given the Duff Moisture Code and Relative Humidity, will there be a forest fire in the nearby regions of Bejaia and Sidi Bel-abbes in Northwest Algeria?

Identifying and describing the Dataset:

The data set used is the Algerian Forest Fires Data Set. It contains 244 observations (122 from each nearby region studied) of environmental conditions over 4 months (June to September) and whether those conditions correlated with a forest fire that day or not. The data set contained the variables such as the date, weather conditions (temperature, humidity, wind speed, and rain), and Fire Weather Index components representing numeric ratings for fire intensity. The Fire Weather Index variables include: The Fine Fuel Moisture Code (FFMC): which numerically represents the moisture content of biological litter and other fine fuels in a forest and indicates how flammable they are.

The Duff Moisture Code (DMC): numerically represents the moisture content of loosely packed organic layers of moderate depth in the soil and how flammable they are. This code represents the fuel consumption in moderate duff layers and medium-size woody material. Drought Code (DC): numerically represents the average moisture content of deep, compact organic layers of biomass in forest soil. Indicates how flammable deep layers of soil are and the likelihood of drought.

Initial Spread Index (ISI): numerically represents the expected rate of fire spread. It is based on wind speed and FFMC described above. Buildup Index (BUI): numerically represents the total amount of biomass (fuel) available for combustion. It is based on the DMC and the DC.

The Fire Weather Index (FWI): numerically represents fire intensity. It is based on the ISI and the BUI, and it indicates the overall danger level of a forest fire. The data set also includes a Classes variable of "not fire" and "fire" to convey whether each observation resulted in a forest fire or not.

Methods and Results

In [4]:
```r
# Let's read the data on the Algerian Forest Fire using read_csv.

# The data consists of two tables containing observations in regard to the Bejaia and Sidi-Bel Abbes regions.
# Each row is an observation that was taken everyday during 4 months: June through September in 2012.


# As there are two tables, we want to combine both of them by erasing the metadata.
# Hence, as a first step, we are skipping the first line due to the fact that the title of the dataset (metadata) is not needed.
# by using "skip=1" to delete the first row.

fire_md_w <- read_csv("https://archive.ics.uci.edu/ml/machine-learning-databases/00547/Algerian_forest_fires_dataset_UPDATE.csv", skip = 1)

fire_md_w
```

```
Parsed with column specification:
cols(
  day = col_character(),
  month = col_character(),
  year = col_character(),
  Temperature = col_character(),
  RH = col_character(),
  Ws = col_character(),
  Rain = col_character(),
  FFMC = col_character(),
  DMC = col_character(),
  DC = col_character(),
  ISI = col_character(),
  BUI = col_character(),
  FWI = col_character(),
  Classes = col_character()
)

Warning message:
“2 parsing failures.
row col   expected     actual                                                                                          file
123  -- 14 columns 1 columns  'https://archive.ics.uci.edu/ml/machine-learning-databases/00547/Algerian_forest_fires_dataset_UPDATE.csv'
168  -- 14 columns 13 columns 'https://archive.ics.uci.edu/ml/machine-learning-databases/00547/Algerian_forest_fires_dataset_UPDATE.csv'
”
```

A spec_tbl_df: 246 × 14

day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Classes
<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>
01	06	2012	29	57	18	0	65.7	3.4	7.6	1.3	3.4	0.5	not fire
02	06	2012	29	61	13	1.3	64.4	4.1	7.6	1	3.9	0.4	not fire
03	06	2012	26	82	22	13.1	47.1	2.5	7.1	0.3	2.7	0.1	not fire
04	06	2012	25	89	13	2.5	28.6	1.3	6.9	0	1.7	0	not fire
05	06	2012	27	77	16	0	64.8	3	14.2	1.2	3.9	0.5	not fire
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
26	09	2012	30	65	14	0	85.4	16	44.5	4.5	16.9	6.5	fire
27	09	2012	28	87	15	4.4	41.1	6.5	8	0.1	6.2	0	not fire

day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Classes
<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>
28	09	2012	27	87	29	0.5	45.9	3.5	7.9	0.4	3.4	0.2	not fire
29	09	2012	24	54	18	0.1	79.7	4.3	15.2	1.7	5.1	0.7	not fire
30	09	2012	24	64	15	0.2	67.3	3.8	16.5	1.2	4.8	0.5	not fire

Table 1: Raw Data of the Forest Fires in Algeria

In [5]:

```
# Second, we need to tidy the data before performing any analysis because we need to make sure that our data is transparent, reproducable and organized.

# Step 1 for Tidy Data: In our original dataset, there is metadata in between the observations as you can see in:
slice(fire_md_w, 123, 124)

# Hence, we delete rows 123 and 124 :
fire_w <- fire_md_w[-c(123, 124 ) ,]
```

A spec_tbl_df: 2 × 14

	day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Classes
	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>
	Sidi-Bel Abbes Region Dataset	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Classes

Table 2: Metadata we Have to Remove

In [6]:

```
# Step 2 in Tidy Data: One observation in our dataset was inputted incorrectly in row 171 of the original csv file from the data frame
slice(fire_w, 166)

# So, we are removing observation 165 after accounting for all the previous removed rows
fire_data <- fire_w[-c(166),]
```

A tibble: 1 × 14

day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Classes
<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>
14	07	2012	37	37	18	0.2	88.9	12.9	14.6 9	12.5	10.4	fire	NA

Table 3: Observation that was Inputted Incorrectly This is done in order to tidy data

In [7]:

```
# Hence, our original dataset after deleting the metadata is:
fire_data
```

A tibble: 243 × 14

day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Classes
<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>
01	06	2012	29	57	18	0	65.7	3.4	7.6	1.3	3.4	0.5	not fire

day	month	year	Temperature	RH	Ws	Rain	FFMC	DMC	DC	ISI	BUI	FWI	Classes
<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>
02	06	2012	29	61	13	1.3	64.4	4.1	7.6	1	3.9	0.4	not fire
03	06	2012	26	82	22	13.1	47.1	2.5	7.1	0.3	2.7	0.1	not fire
04	06	2012	25	89	13	2.5	28.6	1.3	6.9	0	1.7	0	not fire
05	06	2012	27	77	16	0	64.8	3	14.2	1.2	3.9	0.5	not fire
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
26	09	2012	30	65	14	0	85.4	16	44.5	4.5	16.9	6.5	fire
27	09	2012	28	87	15	4.4	41.1	6.5	8	0.1	6.2	0	not fire
28	09	2012	27	87	29	0.5	45.9	3.5	7.9	0.4	3.4	0.2	not fire
29	09	2012	24	54	18	0.1	79.7	4.3	15.2	1.7	5.1	0.7	not fire
30	09	2012	24	64	15	0.2	67.3	3.8	16.5	1.2	4.8	0.5	not fire

Table 4: Tidy Data With All the Predictors

In [8]:

```
fire_tidy <- fire_data %>%
    mutate(Classes = as_factor(Classes)) %>% # Turn "Classes" from character into factor as it is categorical data
    mutate_if(is.character,as.numeric) %>%   # And the rest of our columns into numeric as it is quantitative data
    select(DMC, RH, Classes)                 # Select only the variable that are needed in our study

fire_tidy
```

A tibble: 243 × 3

DMC	RH	Classes
<dbl>	<dbl>	<fct>
3.4	57	not fire
4.1	61	not fire
2.5	82	not fire
1.3	89	not fire
3.0	77	not fire
⋮	⋮	⋮
16.0	65	fire
6.5	87	not fire
3.5	87	not fire
4.3	54	not fire
3.8	64	not fire

Table 5: Tidy Data with Selected Variables These will be used to form model

In [9]:
```
set.seed(9876) # We need to set the seed to a random number to maintain the same seed in our study!

# We split the data into 75% in Training and 25% in Testing AND we need to ensure we have the same proportion  of "Classes" in both of them
fire_tidy_split <- initial_split(fire_tidy , prop = 0.75, strata = Classes)

training_fire <- training(fire_tidy_split)  # We assign the training data amd testing data
testing_fire <- testing(fire_tidy_split)

training_fire
```

A tibble: 183 × 3

DMC	RH	Classes
<dbl>	<dbl>	<fct>
4.1	61	not fire
2.5	82	not fire
1.3	89	not fire
9.9	54	fire
7.9	88	not fire
⋮	⋮	⋮
16.0	65	fire
6.5	87	not fire
3.5	87	not fire
4.3	54	not fire
3.8	64	not fire

Table 6: Training Data of Algerian Forest Fire

In [10]:
```
# Before we start making our model, we will do a summary table in order to know the proportions of each class (preliminary data analysis).
# We need to check if the training data needs to upsamled if the data is too unbalanced.

summary_table_fire <- training_fire %>%
    group_by(Classes) %>%
    summarise (n= n())

summary_table_fire
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

A tibble: 2 × 2

Classes	n
<fct>	<int>
not fire	80

Classes	n
<fct>	<int>
fire	103

Table 7: Summary Table in regard to Training Data (Preliminary Data Analysis) This summary table indicates that the observations which say "fire" and "not fire" are not imbalanced, so upsampling is not necessary to counter class imbalance.

We are not upsampling as the number of observations that are classified as "fire" and the number of observations that are classified as "not fire" are close on value (only deviate from each other by 23 points) and thus are not imbalanced.

In [11]:
```
#Visualising the traing data observations
train_plot <- training_fire %>%
    ggplot(aes(x= RH,  y= DMC, color = Classes)) +
    geom_point() +
    labs(x = "Relative Humidity (%)", y = "Duff Moisture Code", color = "Fire?") +
    ggtitle("Duff Moisture vs Relative Humidity from Training Data") +
    scale_color_manual(labels = c("Not Fire", "Fire"),
                       values = c("green4", "firebrick1")) +
    theme(text = element_text(size = 18), plot.title = element_text(hjust = 0.5))

train_plot
```

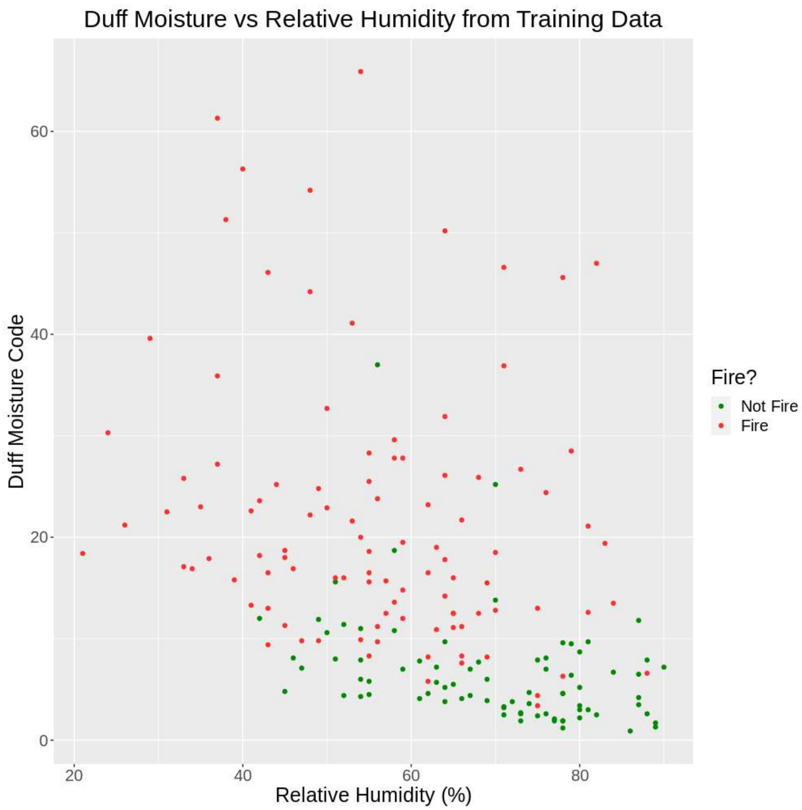

Figure 1: Scatterplot of Duff Moisture and Relative Humity using the Training Data. This visualization represents a weak, negatively associated relationship since the data points are not too close to each other, and the points decrease in the graph as we move to the right. The data points with a true class of "Not Fire" have greater Relative Humidity values and lower Duff Moisture codes. The points which have a true class of "Fire" have low to medium Relative Humidity values and greater Duff Moisture codes.

In [12]:

```
# Creating the standardized recipe with appropriate predictors
fire_recipe <- recipe(Classes ~ ., data = training_fire) %>%
    step_scale(all_predictors()) %>%
    step_center(all_predictors())
fire_recipe

# Create a model specification and tune for optimal k
knn_spec <- nearest_neighbor(weight_func = "rectangular", neighbors = tune()) %>%
    set_engine("kknn") %>%
    set_mode("classification")
knn_spec

# Create a 10-fold cross validation object
fire_vfold <- vfold_cv(training_fire, v = 10, strata = Classes)

# Creating a neighbors dataframe
gridvals <- tibble(neighbors = seq(from = 1, to = 15, by = 2))

# Fit the knn model
fire_workflow_metrics <- workflow() %>%
    add_recipe(fire_recipe) %>%
    add_model(knn_spec) %>%
    tune_grid(resample = fire_vfold, grid = gridvals) %>%
    collect_metrics()
```

```
Data Recipe

Inputs:

      role #variables
   outcome          1
 predictor          2

Operations:

Scaling for all_predictors()
Centering for all_predictors()
K-Nearest Neighbor Model Specification (classification)

Main Arguments:
  neighbors = tune()
  weight_func = rectangular

Computational engine: kknn
```

In [13]:

```
# Get predictions on the validation data and computing the accuracy
fire_accuracy <- fire_workflow_metrics %>%
    filter(.metric == "accuracy") %>%
    arrange(neighbors)
fire_accuracy
```

A tibble: 8 × 7

neighbors	.metric	.estimator	mean	n	std_err	.config
<dbl>	<chr>	<chr>	<dbl>	<int>	<dbl>	<chr>
1	accuracy	binary	0.7538012	10	0.02743557	Model1
3	accuracy	binary	0.8359649	10	0.02179164	Model2
5	accuracy	binary	0.8307018	10	0.02787064	Model3
7	accuracy	binary	0.8634503	10	0.03308691	Model4
9	accuracy	binary	0.8634503	10	0.03410763	Model5
11	accuracy	binary	0.8578947	10	0.03307872	Model6
13	accuracy	binary	0.8467836	10	0.03852357	Model7
15	accuracy	binary	0.8412281	10	0.03346658	Model8

Table 8: Accuracy of Different K-Values The optimal k vlaue output by this analysis is 7 with an accuracy of 86% and a standard error of 3% meaning the range of accuracy will be from 83-89%

In [14]:

```
# Create a accuracy estimate vs number of neighbors plot
cross_val_plot <- ggplot(fire_accuracy, aes(x = neighbors, y = mean)) +
    geom_point() +
    geom_line() +
    labs(x = "Neighbors", y = "Accuracy Estimates") +
    theme(text = element_text(size = 18), plot.title = element_text(hjust = 0.5)) +
    ggtitle ("Accuracy Estimates vs Number of Neighbors")

cross_val_plot
```

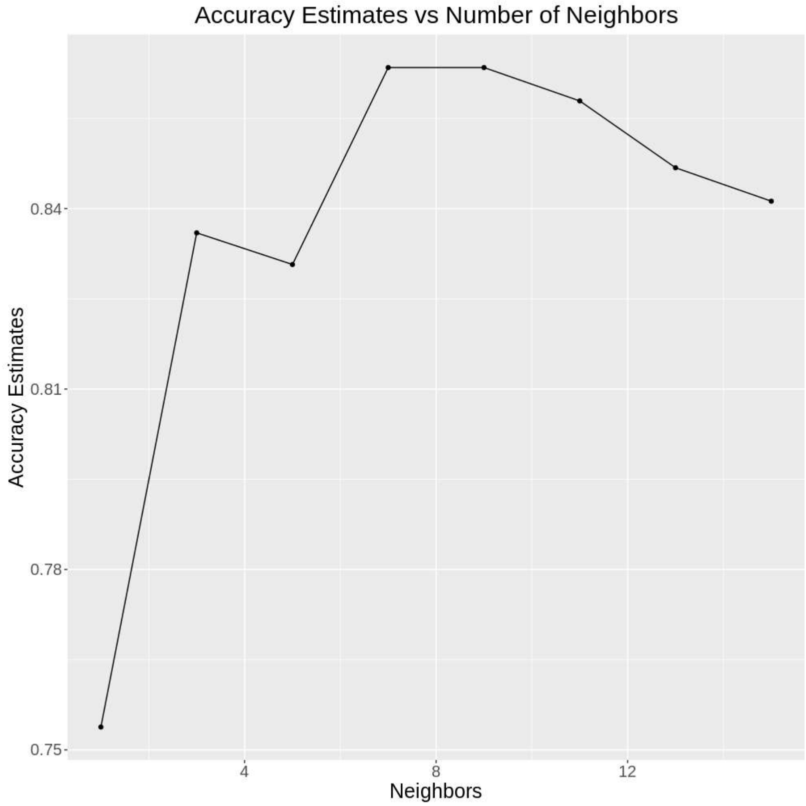

Figure 2: Accuracy Estimates of Different Neighbors Visualizes k value accuracies as a line plot

In [15]:

```
# Get the best k value
best_k <- fire_accuracy %>%
    arrange(desc(mean)) %>%
    slice(1) %>%
    pull(neighbors)
best_k
```

7

In [16]:

```
# Recreate the model specification using the acquired best_k value
best_knn_spec <- nearest_neighbor(weight_func = "rectangular", neighbors = best_k) %>%
    set_engine("kknn") %>%
    set_mode("classification")
best_knn_spec

# Refit the knn model using the previous recipe and the new model specifications
fire_fit <- workflow() %>%
    add_model(best_knn_spec) %>%
    add_recipe(fire_recipe) %>%
    fit(data = training_fire)
fire_fit
```

```
K-Nearest Neighbor Model Specification (classification)

Main Arguments:
  neighbors = best_k
  weight_func = rectangular

Computational engine: kknn
══ Workflow [trained] ══════════════════════════════════════════════════════════
Preprocessor: Recipe
Model: nearest_neighbor()

── Preprocessor ────────────────────────────────────────────────────────────────
2 Recipe Steps

● step_scale()
● step_center()

── Model ───────────────────────────────────────────────────────────────────────

Call:
kknn::train.kknn(formula = ..y ~ ., data = data, ks = ~best_k,     kernel = ~"rectangular")

Type of response variable: nominal
Minimal misclassification: 0.1420765
Best kernel: rectangular
Best k: 7
```

In [17]:

```
# Predicted Classes from the classifier
fire_predictions <- predict(fire_fit, testing_fire) %>%
    bind_cols(testing_fire)
fire_predictions
```

A tibble: 60 × 4

.pred_class	DMC	RH	Classes
<fct>	<dbl>	<dbl>	<fct>
not fire	3.4	57	not fire
not fire	3.0	77	not fire
not fire	5.8	67	fire
fire	12.1	73	fire
not fire	13.8	81	fire
⋮	⋮	⋮	⋮
not fire	8.3	72	fire
not fire	11.5	49	fire
fire	24.9	41	fire
fire	23.6	34	fire
fire	29.4	56	fire

Table 9: Fire Predictions of Testing Data .pred_class shows what the model predicted and Classes is the true value of whether the environmental conditions resulted in a fire or not.

In [18]:
```
# Visualizing the predicted and true "Classes" of the testing data

fire_pred_plot <- ggplot(fire_predictions, aes(x= RH, y= DMC, color = Classes, shape = .pred_class)) +
    geom_point(size = 2.5) +        # We increased the size of data points so we can better identified the distictions (shape)
    labs(x = "Relative Humidity (%)", y = "Duff Moisture Code", color = "True Classes", shape = "Predicted Classes") +
    ggtitle("Duff Moisture vs Relative Humidity from Testing Data
            (Predicted & Observed)") +
    scale_color_manual(labels = c("Not Fire", "Fire"),
                       values = c("green4", "firebrick1")) +
    theme(text = element_text(size = 18), plot.title = element_text(hjust = 0.5))

fire_pred_plot
```

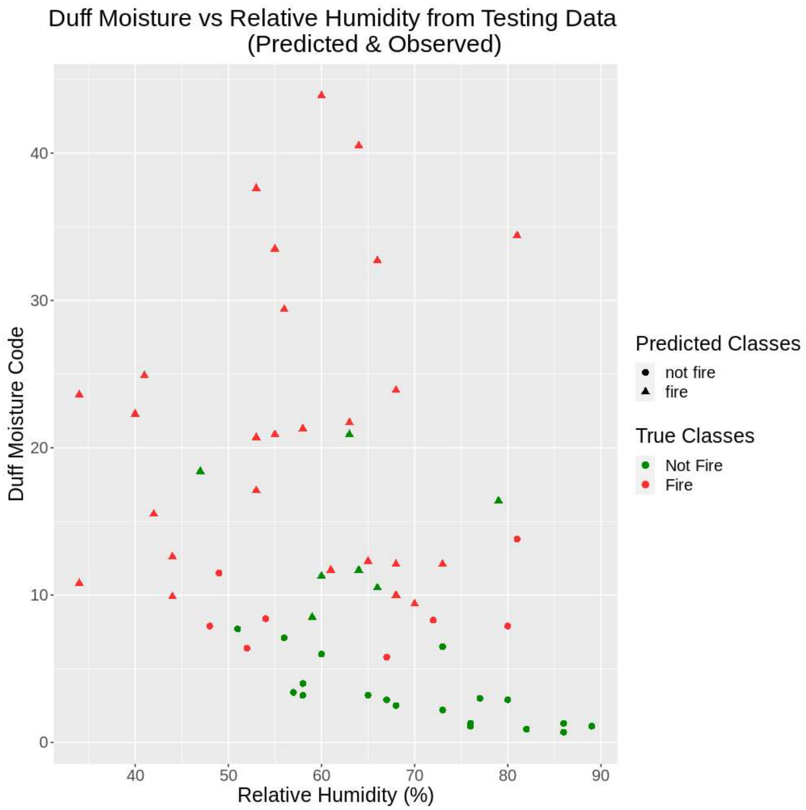

Figure 3: Predicted and Observed Classes from Testing Data The general trend of the visulization is similatr to the trauning data visual. Most data points' predictions match their true class. However, some points' predictions do not match their true classes (for example: a red circle or a green triangle).

In [19]:
```
# Assessing the classifier's accuracy
fire_metrics <- fire_predictions %>%
    metrics(truth = Classes, estimate = .pred_class)
```

```
# Creating the confusion matrix of the classifer
fire_conf_mat <- fire_predictions %>%
    conf_mat(truth = Classes, estimate = .pred_class)
fire_conf_mat
```

```
          Truth
Prediction not fire fire
  not fire       19    8
  fire            7   26
```

Table 10: Confusion Matrix The model produced 8 false positives, 7 false negatives, 26 true positives, and 19 true negatives.

Discussion

Summary:

The general results indicate that the points which have high relative humidity and lower DMC are predicted to have a class of "Not Fire". Conversely, the points with high DMC and low to medium humidity are classified as "Fire." Moreover, the results within the confusion matrix indicate that our model predicted 8 false positives (predicted fire when the true class was not fire) and 7 false negatives (predicted not fire when the true class was fire). The overall accuracy of our model using the optimal k was 86%. This accuracy is adequate for our purposes since the classifier makes mistakes in predicting the probability of fire only 14% of the time.

Discuss whether this aligned with expectations:

Initially, we expected that the points which have a low RH and high DMC are more likely to be classified as a "Fire" class since the dryness in the atmosphere and flammability of the duff soil layer will lead to ideal wildfire conditions. Our model predictions aligned with our expectations. Something slightly deviating from our expectations was how the mid-value RH points were also more likely to be classified as "Fire." This could be due to specific regional climate conditions and other omitted variables impacting the weather, but further analysis would be needed to be certain.

Discuss what impact could such findings have:

Determining which weather conditions will lead to devastating wildfires could prevent a huge loss of life since fires can be anticipated. A model akin to the one above could let National Park Officials input the environmental condition data for that day and determine the risk of a fire. Such a measure of fire probability could save National Parks hundreds of dollars since they would be prepared for fire dispatch more readily if the risk of fire was genuine and avoid spending resources on false fire alarms. Essentially, the impact of determining environmental trends would lead to a conservation of life and resources.

Discuss what future questions could this lead to?

How can the predictors be used to predict fire intensity (how hot it burns or how much area it covers)? Can we apply this model to other countries than are not Algeria (ie. other locations that might have different weather conditions)?

References

Faroudja ABID et al. , â€œPredicting Forest Fire in Algeria using Data Mining Techniques: Case Study of the Decision Tree Algorithmâ€ , International Conference on Advanced Intelligent Systems for Sustainable Development (AI2SD 2019) , 08 - 11 July , 2019, Marrakech, Morocco. "Algerian Forest Fires Dataset Data Set." UCI Machine Learning Repository: Algerian Forest Fires Dataset Data Set, https://archive.ics.uci.edu/ml/datasets/Algerian+Forest+Fires+Dataset++.

Canada, Natural Resources. “Canadian Wildland Fire Information System: Canadian Forest Fire Weather Index (FWI) System.” Canadian Wildland Fire Information System | Canadian Forest Fire Weather Index (FWI) System, Government of Canada, n.d., https://cwfis.cfs.nrcan.gc.ca/background/summary/fwi?wbdisable=true#:~:text=The%20Duff%20Moisture%20Code%20.

Service, BC Wildfire. “Wildfire Glossary.” Province of British Columbia, Province of British Columbia, 15 Mar. 2021, https://www2.gov.bc.ca/gov/content/safety/wildfire-status/about-bcws/glossary.

Penn State. "Forest 'duff' must be considered in controlled burning to avoid damaging trees." ScienceDaily. ScienceDaily, 24 February 2020. https://www.sciencedaily.com/releases/2020/02/200224165303.htm.