# Kenyan Agricultural Exports: Data Science and Predictive Modeling

This report provides a detailed workflow for data processing, cleaning, visualization, and predictive modeling to support decision-making in Kenya's agricultural export sector. The project leverages R programming for mapping crops, handling data, and developing machine learning models.

## 1. Data Processing and Mapping

Kenya's cash crops were mapped based on environmental conditions suitable for export crops. The R code below demonstrates the process of mapping, data replacement, and exporting the modified dataset.

```r  
# Importing and exploring the data  
crop <- read.csv("../data/Crop\_recommendation.csv")  
print(head(crop))  
str(crop)  
  
# Mapping crops based on conditions  
kenya\_export\_crops <- c("Tea", "Coffee", "Avocado", "Macadamia Nuts", ...)  
crop\_mapping <- setNames(kenya\_export\_crops, unique(crop$label))  
crop$label <- crop\_mapping[crop$label]  
  
# Export the processed data  
write.csv(crop, file = "../data/kenyan\_cash\_crops\_conditions.csv", row.names = FALSE)  
```

## 2. Data Cleaning

```r  
# Load the processed data  
kenyancrops <- read.csv("../data/kenyan\_cash\_crops\_conditions.csv")  
  
# Check for missing and duplicate values  
missing\_vals <- sapply(kenyancrops, function(x) sum(is.na(x)))  
print(missing\_vals)  
  
duplicate\_rows <- kenyancrops[duplicated(kenyancrops), ]  
num\_duplicates <- nrow(duplicate\_rows)  
print(num\_duplicates)  
```

## 3. Data Visualization

```r  
# Bar plot of crop counts  
ggplot(data = as.data.frame(label\_counts), aes(x = Var1, y = Freq)) +  
 geom\_bar(stat = "identity", fill = "steelblue") +  
 labs(title = "Counts of Unique Crops", x = "Crop", y = "Count") +  
 theme\_minimal()  
  
# Scatter plot for rainfall vs. humidity  
ggplot(kenyancrops, aes(x = rainfall, y = humidity, color = label)) +  
 geom\_point() +  
 labs(title = "Rainfall vs Humidity", x = "Rainfall", y = "Humidity") +  
 theme\_minimal()  
```

## 4. Predictive Modeling

```r  
# Train-test split and feature scaling  
trainIndex <- createDataPartition(kenyancrops$crop\_num, p = 0.8, list = FALSE)  
x\_train\_scaled <- predict(preProcess\_scale, x\_train)  
  
# Train machine learning models  
models <- list(  
 rf = train(x\_train\_scaled, y\_train, method = "rf", trControl = control)  
)  
  
# Evaluate and select the best model  
model\_accuracies <- sapply(models, function(model) {  
 y\_pred <- predict(model, x\_test\_scaled)  
 confusionMatrix(y\_pred, y\_test)$overall['Accuracy']  
})  
  
print(model\_accuracies)  
```