Truncated model: Training and Testing Hazard Parent Nodes: Wind and Rain

```
# Environment: Clear working environment
rm(list = ls())

# load libraries
library(rpart)
library(here)
```

```
## here() starts at /Users/masinde/Projects/causal_fairness_Ph_IbF
```

```
library(rpart.plot)
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

Reusable functions

```
rmse <- function(actual, predicted) {
  sqrt(mean((actual - predicted)^2))
}
```

Inputs

```
# Recipe inputs
trunc_train <- read.csv(here("data",  "truncated_train2.csv"))
trunc_test <- read.csv(here("data", "truncated_test.csv"))

nrow(trunc_train)
```

```
## [1] 396
```

Wind Model Training & Testing

Decision trees

```
trunc_wind_model <- rpart(wind_max ~ track_min_dist + island_groups,
                          data = trunc_train,
                          method = "anova")
```

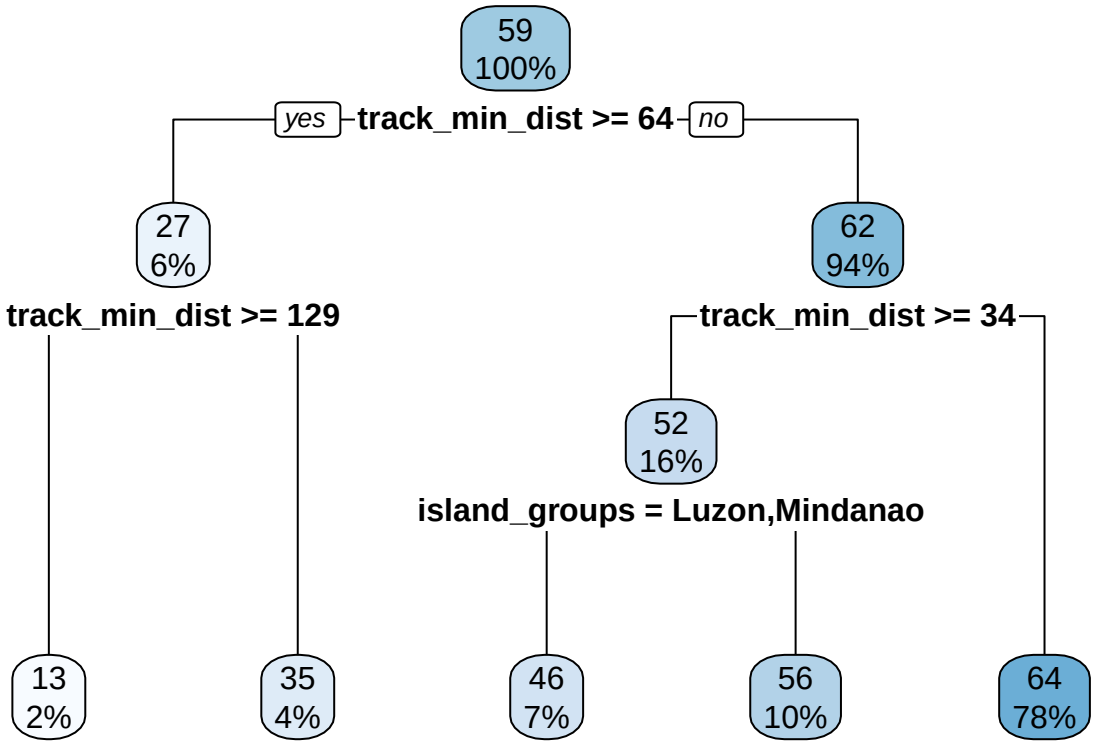
```
wind_pred <- predict(trunc_wind_model, trunc_test)

rmse_wind_pred <- rmse(actual = trunc_test$wind_max,
                               predicted = wind_pred)

cat("rmse of decision tree of wind model", rmse_wind_pred, sep = " ")
```

```
## rmse of decision tree of wind model 11.46146
```

```
rpart.plot(trunc_wind_model)
```



Optimizing For Best parameters

```
set.seed(1234)
# Define training control
control <- trainControl(method = "cv", number = 8)

# Set tuning grid
```

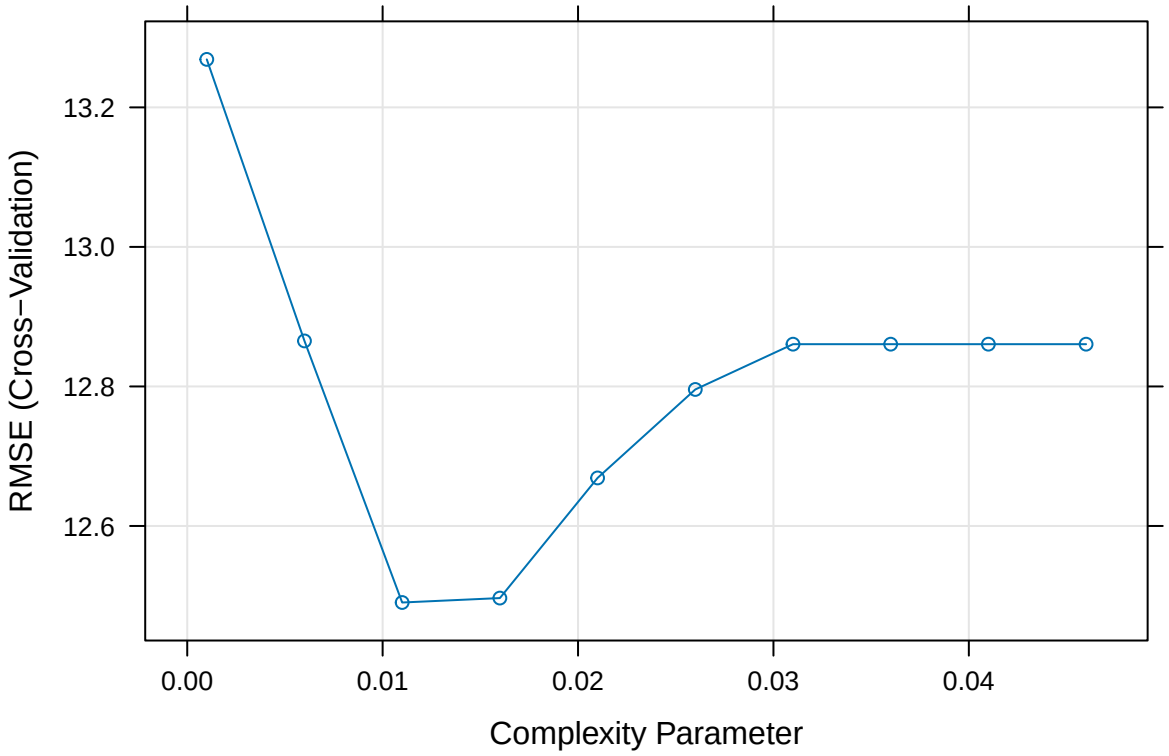
```r
grid <- expand.grid(
  cp = seq(0.001, 0.05, by = 0.005)  # try several cp values
)

# Train model
trunc_wind_model_tuned <- train(
  wind_max ~ track_min_dist + island_groups, data = trunc_train,
  method = "rpart",
  trControl = control,
  tuneGrid = grid
)

print(trunc_wind_model_tuned)
```

```
## CART
##
## 396 samples
##   2 predictor
##
## No pre-processing
## Resampling: Cross-Validated (8 fold)
## Summary of sample sizes: 348, 347, 345, 346, 346, 346, ...
## Resampling results across tuning parameters:
##
##   cp     RMSE      Rsquared   MAE
##   0.001  13.26881  0.2978749  10.57566
##   0.006  12.86523  0.3192651  10.34596
##   0.011  12.49031  0.3426368  10.12511
##   0.016  12.49664  0.3393380  10.09432
##   0.021  12.66884  0.3220873  10.16561
##   0.026  12.79567  0.3155626  10.33153
##   0.031  12.86057  0.3153265  10.39235
##   0.036  12.86057  0.3153265  10.39235
##   0.041  12.86057  0.3153265  10.39235
##   0.046  12.86057  0.3153265  10.39235
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was cp = 0.011.
```

```r
plot(trunc_wind_model_tuned)
```




```
png(here("adjusted SCM/outputs", "dec_wind_trunc_model.png"),
       width = 6,
       units = "in",
       height = 4,
    res = 300)
rpart.plot(trunc_wind_model_tuned$finalModel)
dev.off()
```

```
## pdf
##   2
```

```
wind_pred_tuned <- predict(trunc_wind_model_tuned, trunc_test)

rmse_wind_pred_tuned <- rmse(actual = trunc_test$wind_max,
                             predicted = wind_pred_tuned)

cat("rmse of tuned decision tree of wind model", rmse_wind_pred_tuned, sep = " ")
```

```
## rmse of tuned decision tree of wind model 11.46146
```

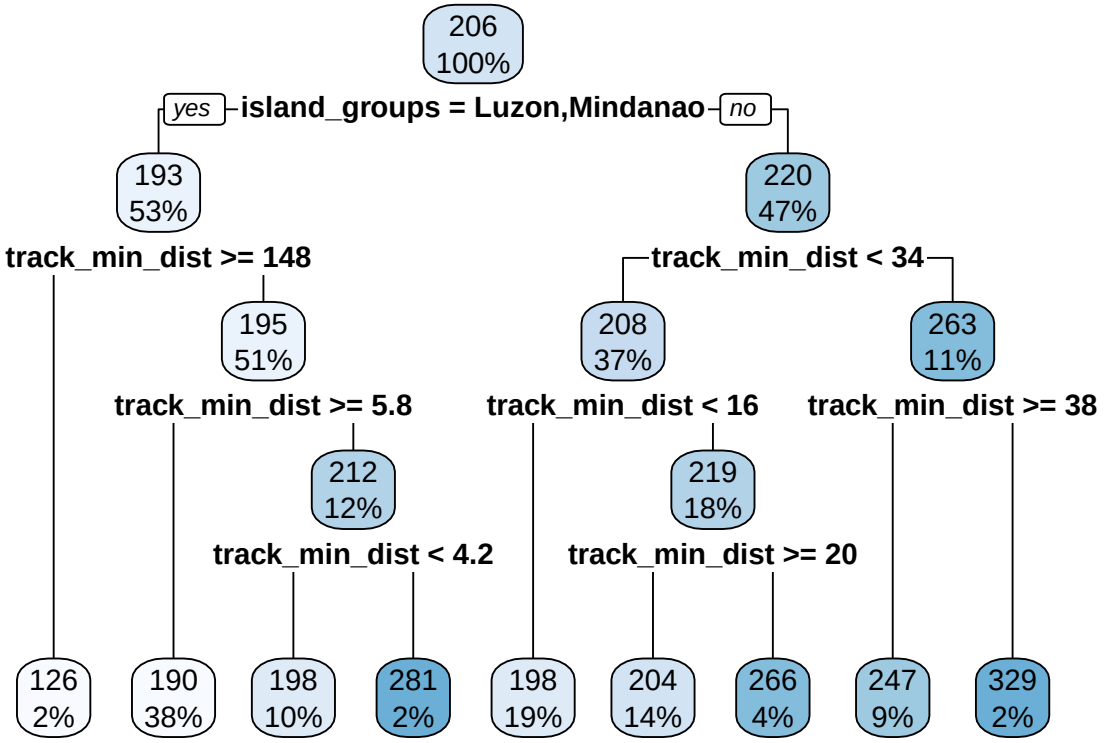
Output

We output the caret tuned decision tree model

```
full_path <- here("adjusted SCM/new trunc models")
saveRDS(trunc_wind_model_tuned,
        file = file.path(full_path, paste0("trunc_wind_model_tuned", ".rds")))
```

Rain Model Training & Testing

```
rpart.plot(trunc_rain_model)
```




```
full_path <- here("adjusted SCM/new trunc models")
saveRDS(trunc_rain_model,
        file = file.path(full_path, paste0("dec_trunc_rain_model_tuned", ".rds")))
```

Optimizing For Best parameters

THIS DID NOT WORK OUT WELL!

```
#rpart.plot(trunc_rain_model_tuned$finalModel)
```

Saving Tuned Rain_Total Decision Tree Model