R_{50} Estimation and Comparison: Final Report

Choosing a Model for Estimating R_{50}

When first looking at this problem and the data I was given I knew that a logistic regression could be used to estimate R_{50}. I decided that a good estimate of R_{50} is the range where there is a 50% probability a target will be detected since R_{50} is the range where 50% of the targets will be detected. However, a prediction using a logistic model can't estimate R_{50} directly. Logistics output a probability given a continuous variable, whereas what I needed was the opposite. I looked to see if there was a way to create a model that would be able to do this "out-of-the-box" but I found nothing and decided to stick with the logistic. I chose to create a logistic regression using the R package `parsnip` since prediction output is in a dataframe and I could obtain confidence intervals easily from it as well.

Solving for R_{50} and Confidence Intervals

Since my model couldn't estimate R_{50} directly, I created a function `solve_R50` that could use the model to solve for R_{50} and the confidence intervals under each condition.

In the function below the argument `R50_mod` is the `parsnip` logistic model and `turbines` is a binary vector indicating the condition to predict under. The argument `type` is one of `"pred"`, `"lower"` or `"upper"`. `"pred"` gives the predicted R_{50} value. `"lower"` or `"upper"` give the corresponding side of the confidence interval.

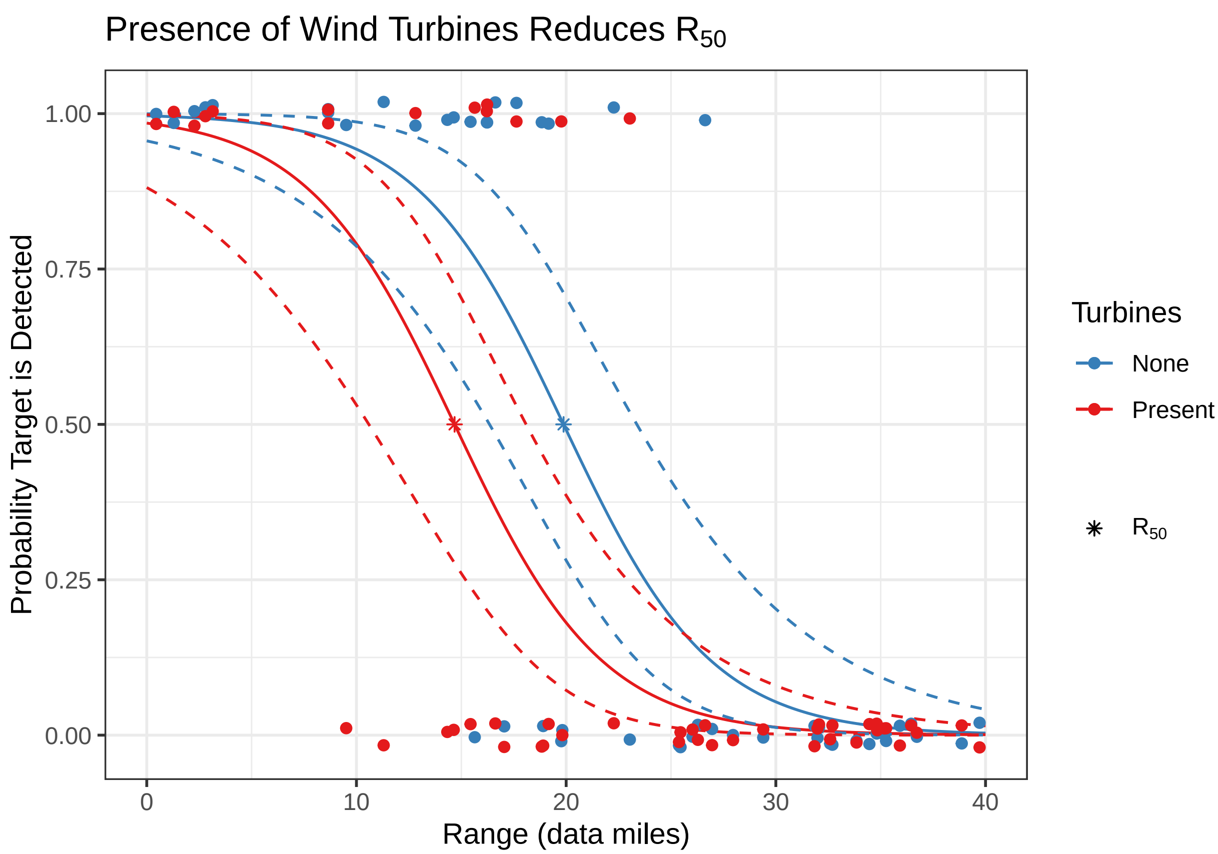
```
solve_R50 <- function(R50_mod, turbines, type) {

  # ... skipped logic for `type` argument

  # create a function that we will find the root of using `uniroot`
  R50_root <- function(range, turbines) {
    predict.model_fit(R50_mod,
                      tibble(range = range, turbines = turbines),
                      # pred_type variable determines if we are solving
                      # for R50 or a confidence interval
                      type = pred_type
    ) %>%
      # pick the column we need from predict.model_fit
      # subtracting .5 creates a root where we want it
      pull(pred_col) - .5
  }

  # solve for each `turbines` condition
  map_dbl(turbines,
          ~ uniroot(R50_root,
                    interval = range(R50_mod$fit$data$range),
                    turbines = .x)$root)
}
```

This is how I used the solver to obtain R_{50} and the confidence intervals for each turbine condition. Below is the information presented in table form and graphically.

```
R50_dat <-
  tibble(turbines = 0:1) %>%
  mutate(R50 = solve_R50(R50_mod, turbines, "pred"),
         lower = solve_R50(R50_mod, turbines, "lower"),
         upper = solve_R50(R50_mod, turbines, "upper"))
```

R_{50} Estimation and Confidence Intervals

Turbines	R_{50}	Lower	Upper
None	19.88	16.35	23.37
Present	14.68	10.59	18.09



Identify the Impact of the Presence of Wind Turbines on R_{50}

We can determine whether the presence of wind turbines affects R_{50} by looking at the impact the `turbines` term has on the model. Below are the estimates of the model's terms. Since the `turbines` term is significant and it represents a horizontal shift in the two curves as shown above, the estimated R_{50} will be significantly different under the two conditions.

R_{50} Model Terms

term	estimate	std.error	statistic	p.value
(Intercept)	5.64	1.29	4.38	0.000012
range	-0.28	0.06	-4.70	0.000003
turbines	-1.47	0.69	-2.12	0.033855

Estimation of power

In order to calculate power I needed to estimate the standard deviation of R_{50}. Unfortunately, the confidence intervals were not symmetric after translating them to be in terms of range. To be conservative in the power calculations I choose the largest margin to determine the standard deviation.

R_{50} Confidence Margins

Turbines	R_{50}	Lower	Upper	Lower Margin	Upper Margin
None	19.88	16.35	23.37	-3.53	3.50
Present	14.68	10.59	18.09	-4.09	3.41

If we assume there are two standard deviations from R_{50} to the interval bounds, then the largest the standard deviation could be is about 2. Since the difference in R_{50} is about 5 data miles less under conditions when there are wind turbines verses no turbines, then the R_{50s} are about 2.5 standard deviation from each other. In other words, the effect size is about 2.5 data miles. Using this information, I created the plot below.