Weather forecast analysis

Mladen Cucak

18 November, 2019

source(here::here("scr","lib", "pkg.R"))

## [1] "All packages were successfully loaded."

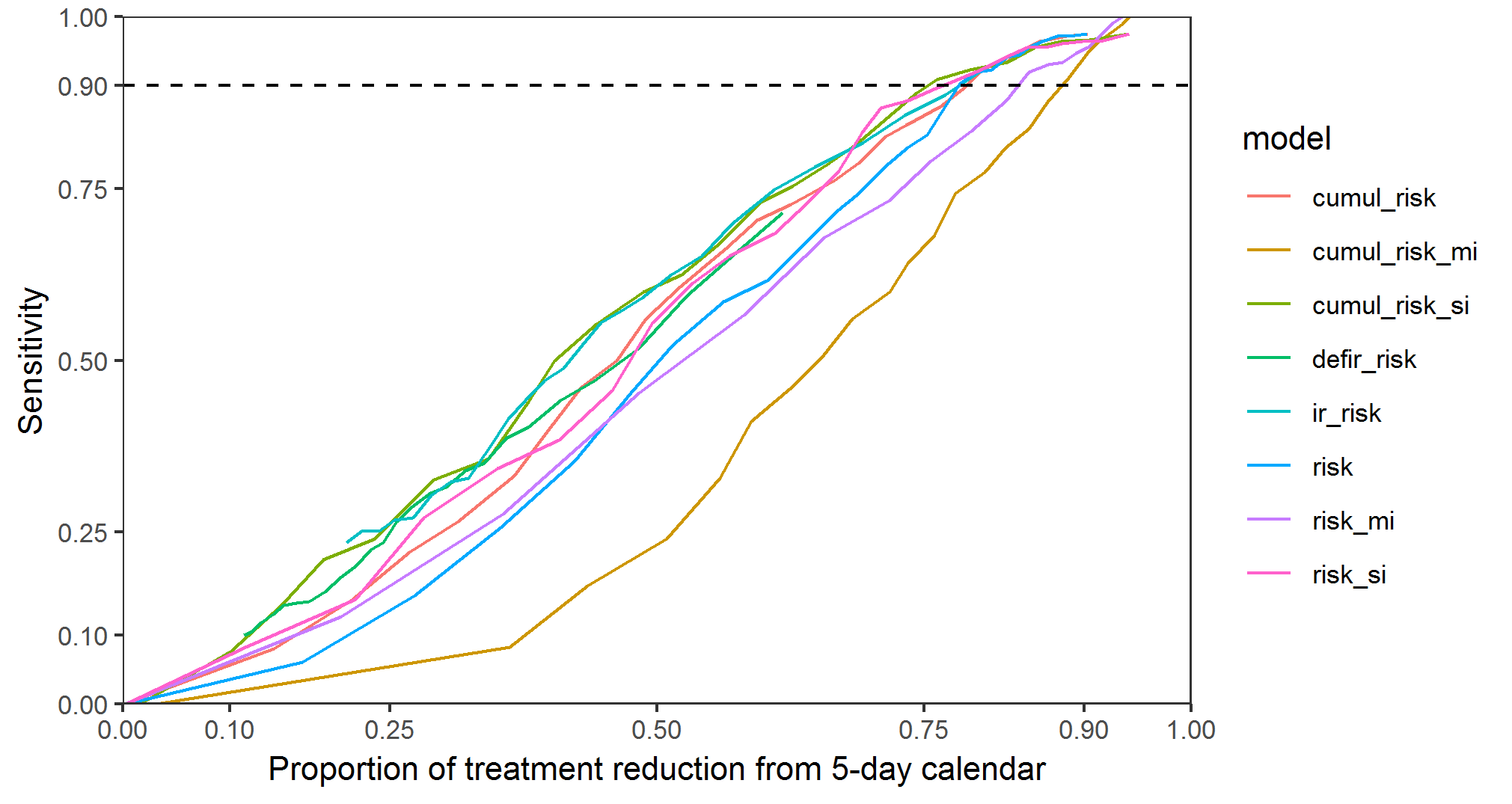
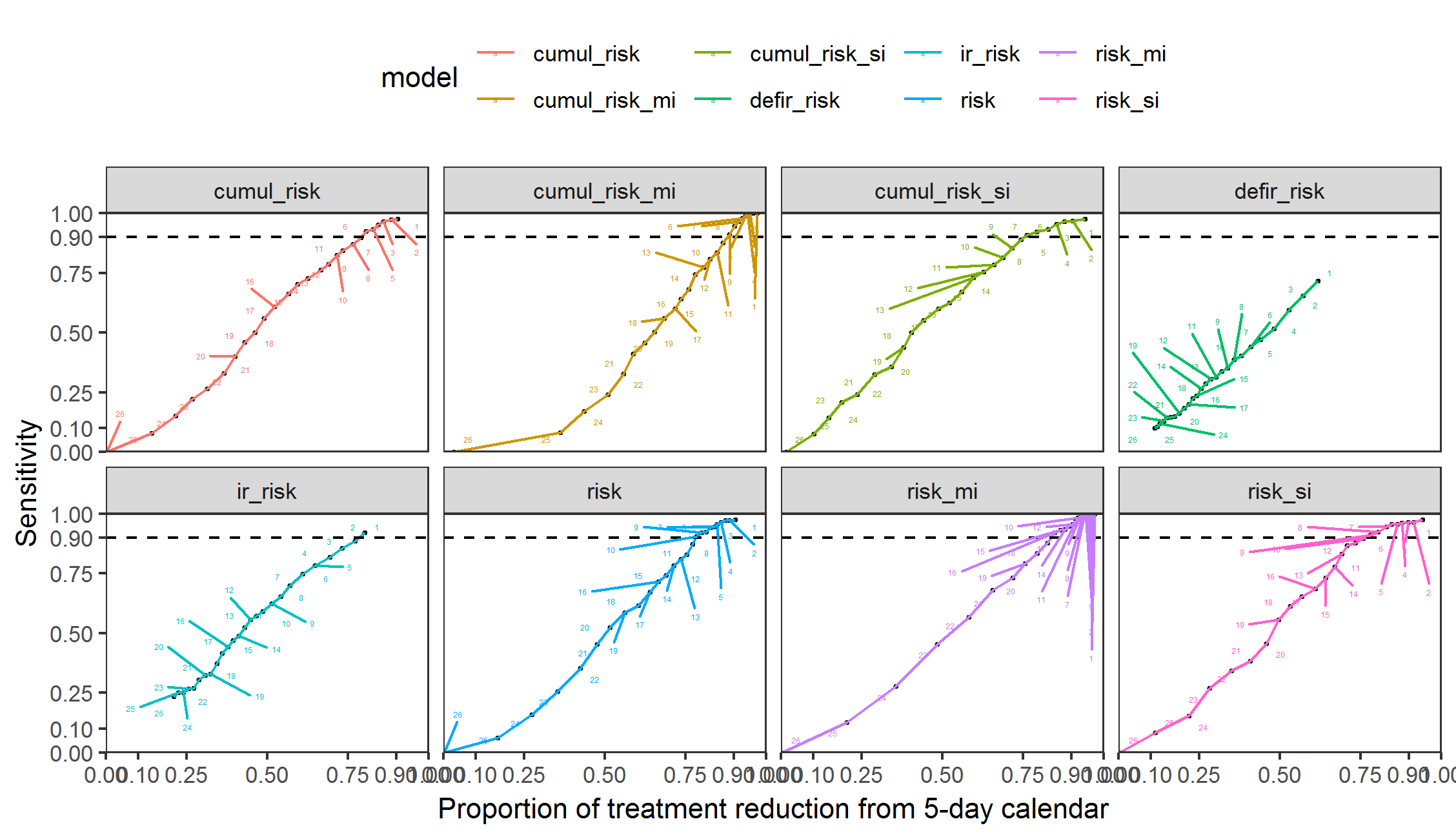
source(here::here("scr", "lib", "funs.R"))

## Intro

Weather data Weather observations were collated for ????5 Met Éireann synoptic weather stations in Ireland for the period from 2017 to 2019???. The historical weather data consisted of hourly observations air temperature (℃) and relative humidity (%) at 2m, the total hourly precipitation (mm) and solar radiation (J/cm2). ????Solar radiation was converted to MJ/cm2 to allow the model run. 10 day weather forecast (???ECMWF model??) consisted of corresponding variables, with time series ending on 20 September to allow for ten day lag at the end of the month. This period was selected as it is representative for the duration blight season. ???missing dates The weather forecast data fot 3, 5 and 10 following days was deliverd in 1, 3 and 6 hour temporal resolution, respectively.

*Different models/select single model or compare them*  
*Cube idea* information content:  
\* Decision threshold (reducing causes higher risk, more savings and more uncertanty in forecast(?))  
\* Lead time  
\* Observed/forecasted variables (perhaps need more data)  
\* Positive negative forecast

## Intro

Range of decision thresholds  

The data is consisted of model runs with observed and forecasted weather data.

## # A tibble: 6 x 19  
## doy set id for\_date stna short\_date spor spor\_cond inf  
## <dbl> <chr> <chr> <date> <chr> <date> <dbl> <chr> <dbl>  
## 1 121 fore 2017~ 2017-05-01 Duns~ 2017-05-01 0.0385 yes 0.622   
## 2 122 fore 2017~ 2017-05-01 Duns~ 2017-05-02 0.00436 no 0   
## 3 123 fore 2017~ 2017-05-01 Duns~ 2017-05-03 0.00507 yes 0.0248  
## 4 124 fore 2017~ 2017-05-01 Duns~ 2017-05-04 0 no 0   
## 5 125 fore 2017~ 2017-05-01 Duns~ 2017-05-05 0 no 0   
## 6 126 fore 2017~ 2017-05-01 Duns~ 2017-05-06 0 no 0   
## # ... with 10 more variables: surv\_prob <dbl>, risk\_si <dbl>,  
## # risk\_mi <dbl>, risk <dbl>, cumul\_risk\_si <dbl>, cumul\_risk\_mi <dbl>,  
## # cumul\_risk <dbl>, ir\_risk <dbl>, defir\_risk <dbl>, day\_step <dbl>

## # A tibble: 6 x 19  
## doy set id for\_date stna short\_date spor spor\_cond inf  
## <dbl> <chr> <chr> <date> <chr> <date> <dbl> <chr> <dbl>  
## 1 121 fore 2017~ 2017-05-01 Duns~ 2017-05-01 0.0385 yes 0.622   
## 2 122 fore 2017~ 2017-05-01 Duns~ 2017-05-02 0.00436 no 0   
## 3 123 fore 2017~ 2017-05-01 Duns~ 2017-05-03 0.00507 yes 0.0248  
## 4 124 fore 2017~ 2017-05-01 Duns~ 2017-05-04 0 no 0   
## 5 125 fore 2017~ 2017-05-01 Duns~ 2017-05-05 0 no 0   
## 6 126 fore 2017~ 2017-05-01 Duns~ 2017-05-06 0 no 0   
## # ... with 10 more variables: surv\_prob <dbl>, risk\_si <dbl>,  
## # risk\_mi <dbl>, risk <dbl>, cumul\_risk\_si <dbl>, cumul\_risk\_mi <dbl>,  
## # cumul\_risk <dbl>, ir\_risk <dbl>, defir\_risk <dbl>, day\_step <dbl>

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| doy | id | short\_date | day\_step | fore | fore\_rhum | fore\_sol\_rad | fore\_temp | obs | obs\_rhum | obs\_sol\_rad | obs\_temp |
| 121 | 2017-05-01\_Dunsany | 2017-05-01 | 1 | 0.023929 | 1.278152 | 0.023929 | 0.020802 | 0.583015 | 0.020802 | 0.583015 | 1.37243 |
| 121 | 2017-05-01\_Gurteen | 2017-05-01 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.00000 |
| 121 | 2017-05-01\_Johnstown | 2017-05-01 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.00000 |
| 121 | 2017-05-01\_Moorepark | 2017-05-01 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.00000 |
| 121 | 2017-05-01\_Oakpark | 2017-05-01 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.00000 |
| 121 | 2018-05-01\_Dunsany | 2018-05-01 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.00000 |
| 121 | 2018-05-01\_Gurteen | 2018-05-01 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.00000 |
| 121 | 2018-05-01\_Johnstown | 2018-05-01 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.00000 |
| 121 | 2018-05-01\_Moorepark | 2018-05-01 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.00000 |
| 121 | 2018-05-01\_Oakpark | 2018-05-01 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.00000 |

## Cases&controls

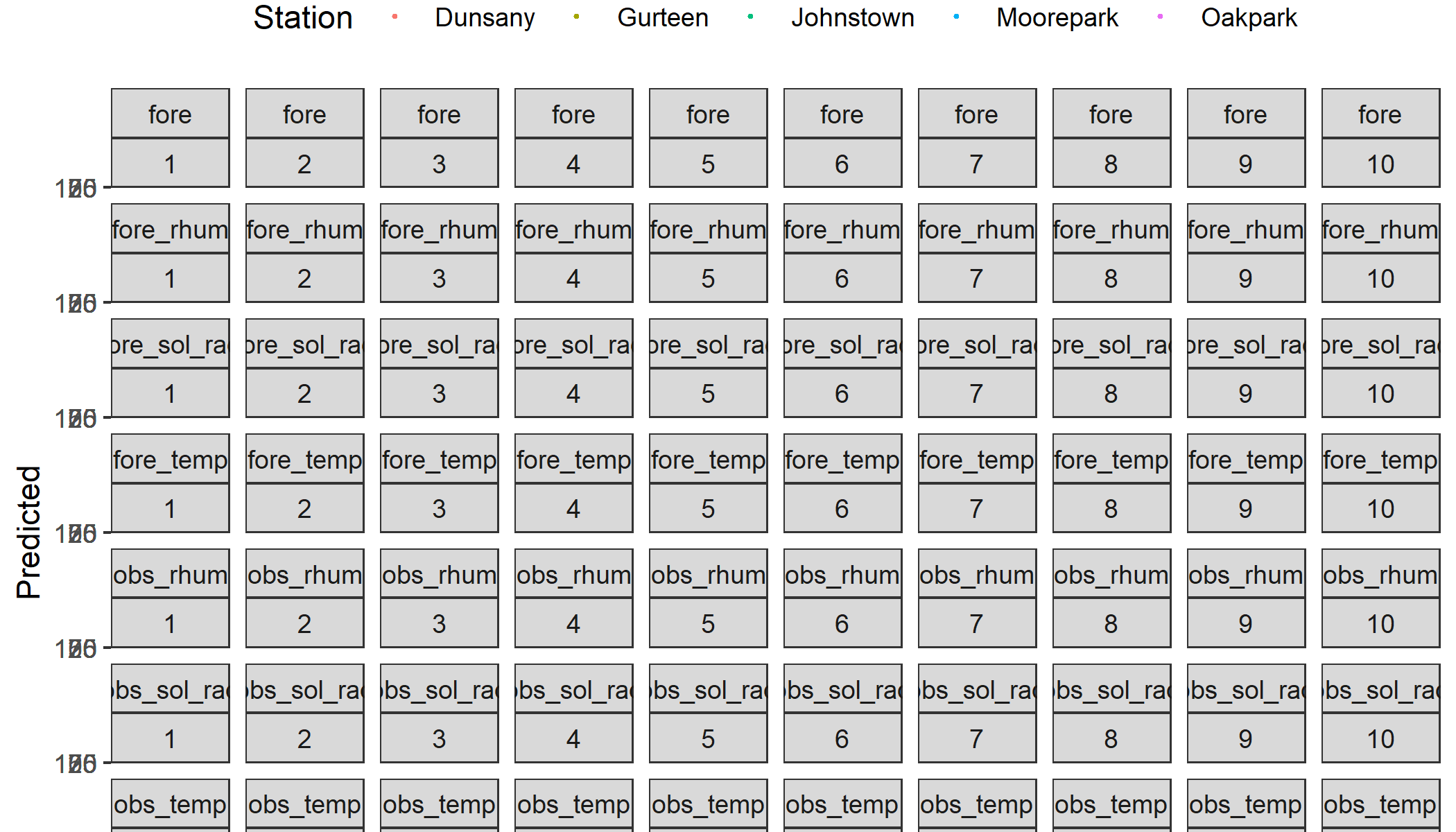
If the analysis is run for a specific decision threshold.  
The corresponding risk to desired threshold is found using linear interpolation between the two nearest thresholds.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | warn\_thresh | risk\_si | risk\_mi | risk | cumul\_risk\_si | cumul\_risk\_mi | cumul\_risk | ir\_risk | defir\_risk |
| 0% | 1 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 1 | 1 |
| 4% | 2 | 0.07 | 0.00 | 0.00 | 0.25 | 0.00 | 0.00 | 2 | 2 |
| 8% | 3 | 0.25 | 0.00 | 0.00 | 0.76 | 0.01 | 0.00 | 3 | 3 |
| 12% | 4 | 0.51 | 0.00 | 0.00 | 1.65 | 0.02 | 0.00 | 4 | 4 |
| 16% | 5 | 0.80 | 0.00 | 0.00 | 2.86 | 0.04 | 0.01 | 5 | 5 |
| 20% | 6 | 1.23 | 0.00 | 0.00 | 4.76 | 0.08 | 0.02 | 6 | 6 |
| 24% | 7 | 1.72 | 0.00 | 0.00 | 6.98 | 0.14 | 0.04 | 7 | 7 |
| 28% | 8 | 2.46 | 0.01 | 0.01 | 8.59 | 0.21 | 0.07 | 8 | 8 |
| 32% | 9 | 3.26 | 0.01 | 0.01 | 11.04 | 0.35 | 0.12 | 9 | 9 |
| 36% | 10 | 4.06 | 0.01 | 0.02 | 14.34 | 0.54 | 0.17 | 10 | 10 |
| 40% | 11 | 5.01 | 0.02 | 0.02 | 17.87 | 0.71 | 0.24 | 11 | 11 |
| 44% | 12 | 6.18 | 0.02 | 0.03 | 21.89 | 0.97 | 0.36 | 12 | 12 |
| 48% | 13 | 7.25 | 0.03 | 0.05 | 26.13 | 1.25 | 0.57 | 13 | 13 |
| 52% | 14 | 8.41 | 0.04 | 0.07 | 31.33 | 1.62 | 0.82 | 14 | 14 |
| 56% | 15 | 9.87 | 0.06 | 0.10 | 37.78 | 1.91 | 1.13 | 15 | 15 |
| 60% | 16 | 11.88 | 0.08 | 0.13 | 45.49 | 2.34 | 1.58 | 16 | 16 |
| 64% | 17 | 14.92 | 0.11 | 0.18 | 55.65 | 2.67 | 2.07 | 17 | 17 |
| 68% | 18 | 17.40 | 0.16 | 0.27 | 65.62 | 3.25 | 2.60 | 18 | 18 |
| 72% | 19 | 19.89 | 0.22 | 0.41 | 72.55 | 3.93 | 3.23 | 19 | 19 |
| 76% | 20 | 22.60 | 0.30 | 0.61 | 83.95 | 4.73 | 4.06 | 20 | 20 |
| 80% | 21 | 26.50 | 0.43 | 0.82 | 102.15 | 5.67 | 4.98 | 21 | 21 |
| 84% | 22 | 31.69 | 0.65 | 1.20 | 130.52 | 6.62 | 7.26 | 22 | 22 |
| 88% | 23 | 39.16 | 1.01 | 1.92 | 162.22 | 8.14 | 9.43 | 23 | 23 |
| 92% | 24 | 47.41 | 1.69 | 3.26 | 195.59 | 10.89 | 13.51 | 24 | 24 |
| 96% | 25 | 65.91 | 3.15 | 7.07 | 258.56 | 14.49 | 22.84 | 25 | 25 |
| 100% | 26 | 159.39 | 16.90 | 74.22 | 625.16 | 70.49 | 193.32 | 26 | 26 |

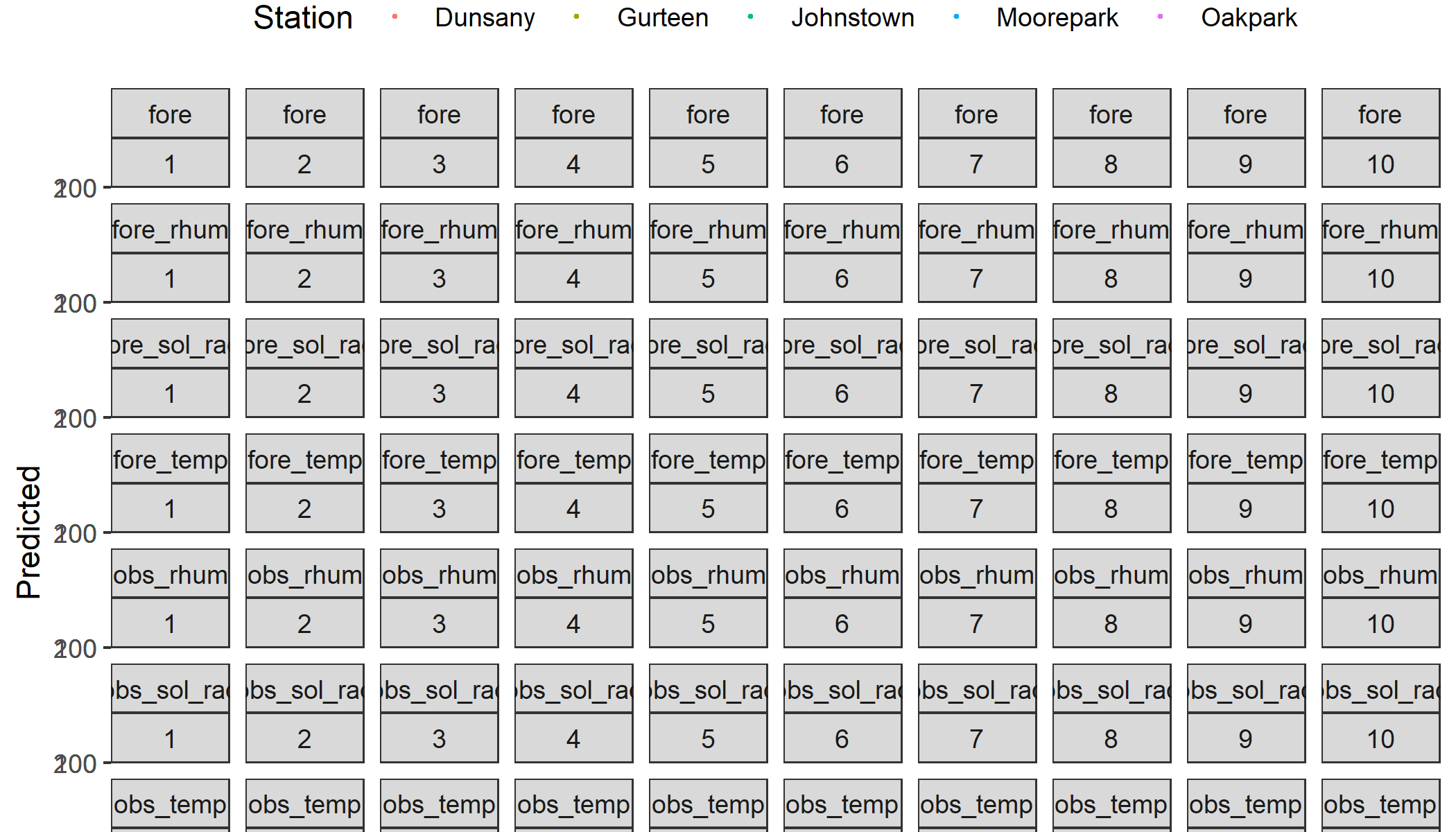
A single decision threshold is used throughout the analysis,

#decide on what is minimum accepted proportion of true predictions  
prop\_tpp <- 0.9  
  
#pick the model  
model <- colnames(fun\_df[, grepl("risk" , names(fun\_df))])[1]  
  
# find two nearest warning thresholds to the accepted decision threshold  
closest\_high <- sum(default\_eval\_lss[[1]][model]>prop\_tpp)  
closest\_low <-which(default\_eval\_lss[[1]][model]<prop\_tpp)[1]  
  
tpp <-   
 default\_eval\_lss[[1]][model][closest\_high:closest\_low,] %>% unlist()  
# Find the risk estimate for these two thresholds  
risk <-   
 warn\_t\_df[default\_eval\_lss[[1]]["warning\_thres"][closest\_high:closest\_low,] %>% unlist(),model]   
#estimate the risk for point prop\_tpp  
risk\_thresh <-   
 predict(lm(risk ~ tpp), data.frame(tpp = prop\_tpp))

#List of outputs per model  
models <- colnames(out\_df[, grepl("risk" , names(out\_df))])  
  
# Plot obs vs pred scatterplts for each model faceted weather and day step  
model <- models[7]  
out\_df %>%   
 select(c(set,for\_date,stna, day\_step, model)) %>%   
 spread(set, model) %>%   
 reshape2::melt( .,  
 id.vars = c("stna", "for\_date", "day\_step", "obs"),  
 measure.vars= c("fore", "fore\_rhum", "fore\_sol\_rad", "fore\_temp","obs", "obs\_rhum", "obs\_sol\_rad", "obs\_temp"),  
 variable.name = "set",  
 value.name = "pred",  
 factorsAsStrings = FALSE  
 ) %>%   
 filter(set != "obs") %>%   
 ggplot(aes(obs,pred, colour =stna))+  
 geom\_point(size= 0.5)+  
 facet\_wrap(set~day\_step, ncol = 10)+  
 theme\_bw()+  
 theme(legend.position = "top")+  
 labs(colour= "Station",   
 title = "Combinations of outputs using forecasted versus observed weather data over lead time",  
 subtitle = paste("Outputs for model", model),  
 x = "Observed", y = "Predicted")+  
 ggsave(filename = here::here("out", "fore", "fig", paste(model, "risk\_si model out for all forecast versions vs lead time .png")), width = 13, height = 14)



#List of outputs per model  
models <- colnames(out\_df[, grepl("risk" , names(out\_df))])  
  
# Plot obs vs pred scatterplts for each model faceted weather and day step  
model <- models[1]  
out\_df %>%   
 select(c(set,for\_date,stna, day\_step, model)) %>%   
 spread(set, model) %>%   
 reshape2::melt( .,  
 id.vars = c("stna", "for\_date", "day\_step", "obs"),  
 measure.vars= c("fore", "fore\_rhum", "fore\_sol\_rad", "fore\_temp","obs", "obs\_rhum", "obs\_sol\_rad", "obs\_temp"),  
 variable.name = "set",  
 value.name = "pred",  
 factorsAsStrings = FALSE  
 ) %>%   
 filter(set != "obs") %>%   
 ggplot(aes(obs,pred, colour =stna))+  
 geom\_point(size= 0.5)+  
 facet\_wrap(set~day\_step, ncol = 10)+  
 theme\_bw()+  
 theme(legend.position = "top")+  
 labs(colour= "Station",   
 title = "Combinations of outputs using forecasted versus observed weather data over lead time",  
 subtitle = paste("Outputs for model", model),  
 x = "Observed", y = "Predicted")+  
 ggsave(filename = here::here("out", "fore", "fig", paste(model, "risk\_si model out for all forecast versions vs lead time .png")), width = 13, height = 14)



# Reshaping the data to set up columns for observed and forecasted values  
dff <-   
out\_df %>%   
 select(c(set,for\_date,stna, day\_step, models)) %>%   
 reshape2::melt( .,  
 id.vars = c( "set", "stna", "for\_date", "day\_step"),  
 measure.vars= models,  
 variable.name = "model",  
 value.name = "risk",  
 factorsAsStrings = FALSE  
 ) %>%  
 spread(set, risk) %>%   
 reshape2::melt( .,  
 id.vars = c("stna", "for\_date", "day\_step", "model", "obs"),  
 measure.vars= c("fore", "fore\_rhum", "fore\_sol\_rad", "fore\_temp","obs", "obs\_rhum", "obs\_sol\_rad", "obs\_temp"),  
 variable.name = "set",  
 value.name = "pred",  
 factorsAsStrings = FALSE  
 ) %>%   
 filter(set != "obs")  
  
head(dff)

## stna for\_date day\_step model obs set pred  
## 1 Dunsany 2017-05-01 1 risk\_si 0.583015 fore 0.023929  
## 2 Dunsany 2017-05-01 1 risk\_mi 0.000000 fore 0.000000  
## 3 Dunsany 2017-05-01 1 risk 0.000000 fore 0.000000  
## 4 Dunsany 2017-05-01 1 cumul\_risk\_si 0.583015 fore 0.023929  
## 5 Dunsany 2017-05-01 1 cumul\_risk\_mi 0.000000 fore 0.000000  
## 6 Dunsany 2017-05-01 1 cumul\_risk 0.000000 fore 0.000000

The model output data is turned into binary values. Here we take the model output data and convert it to 1/0 values for each corresponding warning threshold (first loop) for each model(2nd loop - the within the 1st loop).

wtls <- list()  
out\_ls <- split(dff, dff$model)   
names(out\_ls) <-  
 sapply(out\_ls, function(x) unique(x$model))  
  
  
for(i in seq\_along(warn\_t\_df$warn\_thresh)){  
 loop\_list <- list()  
 for(y in names(out\_ls)){  
 # Find threshold in (lookup table) with all decision thresholds.   
 threshold <- warn\_t\_df[i, y]  
   
 loop\_list[[y]] <-  
 mutate(out\_ls[[y]], warn\_thresh = warn\_t\_df$warn\_thresh[[i]]) %>% # Add a column for warning threshold  
 mutate(obs = ifelse(obs >= threshold, 1, 0)) %>% #  
 mutate(pred = ifelse(pred >= threshold[1] , 1, 0))  
 rm(threshold)  
 }  
 wtls [[i]] <- loop\_list %>% bind\_rows()  
 rm(loop\_list)  
}  
  
  
dff <-   
wtls %>%   
 bind\_rows() %>%   
 select(set, model, warn\_thresh, day\_step, stna, for\_date, obs, pred)  
  
   
dff %>% head()

## set model warn\_thresh day\_step stna for\_date obs pred  
## 1 fore risk\_si 1 1 Dunsany 2017-05-01 1 1  
## 2 fore risk\_si 1 2 Dunsany 2017-05-01 0 0  
## 3 fore risk\_si 1 3 Dunsany 2017-05-01 1 0  
## 4 fore risk\_si 1 4 Dunsany 2017-05-01 0 0  
## 5 fore risk\_si 1 5 Dunsany 2017-05-01 0 0  
## 6 fore risk\_si 1 6 Dunsany 2017-05-01 0 0

Generate frequency tables. *This might be very slow and probably should not be run on computers with RAM < 8GB*

#tab <-  
# with(tab, table(set, model, warn\_thresh, day\_step, obs, pred))