Project 2

Vinay Srinivasan

May 29, 2018

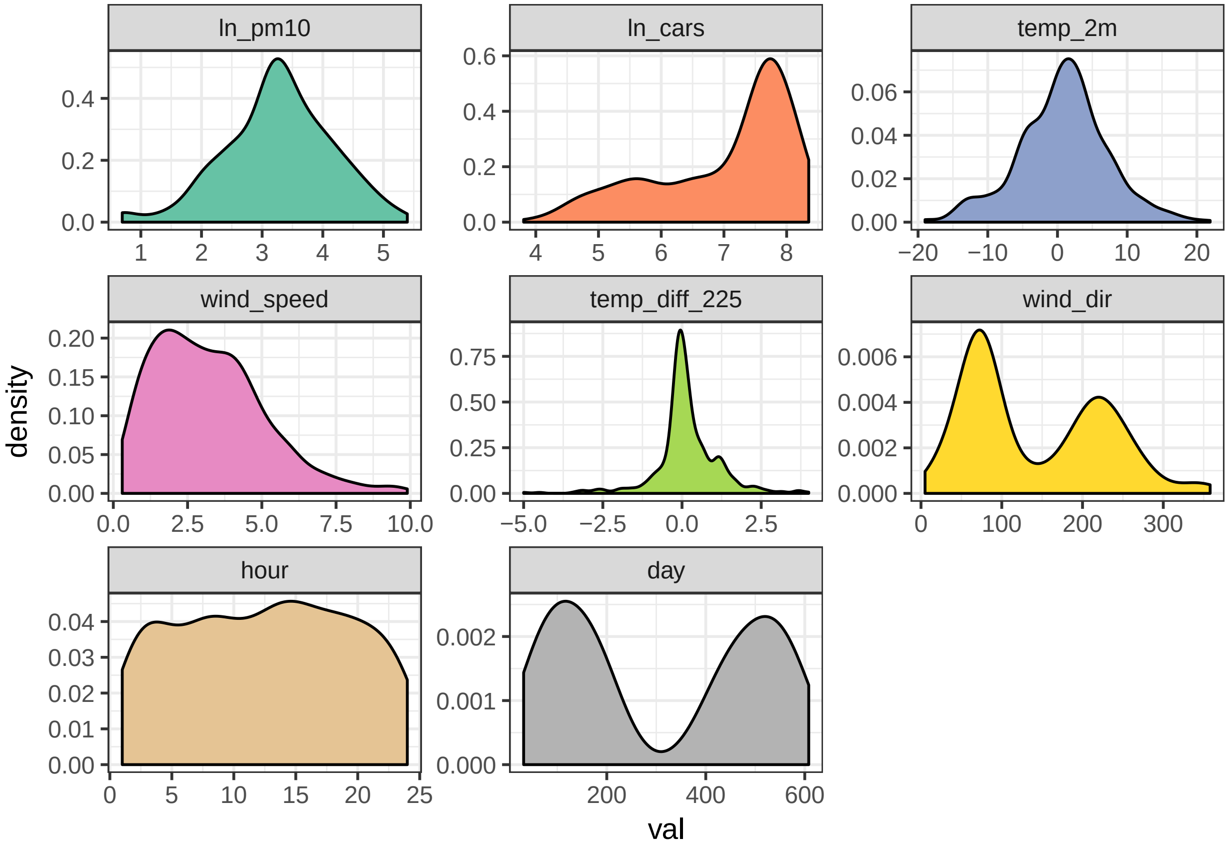
Exploring Data

```
# Loading data
datadf <- fread("../PM10.txt")
setnames(datadf, 1:8, c("ln_pm10", "ln_cars", "temp_2m", "wind_speed",
    "temp_diff_225", "wind_dir", "hour", "day"))

# Creating continuous time variable
datadf <- datadf[order(day, hour)]
datadf[, `:=`(t, 24 * day + hour)]

# Creating season variable (assuming start day october first)
datadf[, `:=`(season, (as.Date("2001-10-01") + day) %>% month %>%
    as.character())]
datadf[season %in% c(12, 1, 2), `:=`(season, "winter")][season %in%
    3:5, `:=`(season, "spring")][season %in% 6:8, `:=`(season,
    "summer")][season %in% 9:11, `:=`(season, "fall")]

# Creating binned wind direction variable
datadf[, `:=`(wind_dir_bin, round_any(wind_dir, 180))][, `:=`(wind_dir_bin,
    ifelse(wind_dir_bin == 180, 0, 1))]
```

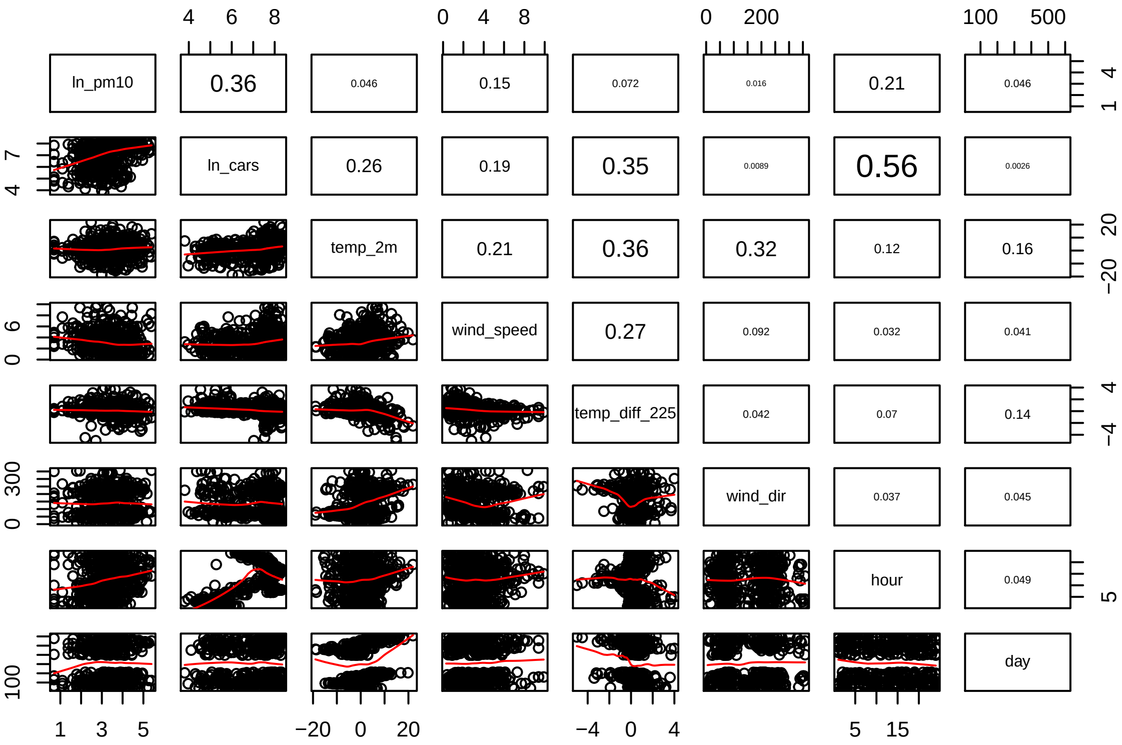
```
# Examing distributions of varibles in dataset
distdata <- melt(datadf[, !c("t", "wind_dir_bin", "season")],
    variable.name = "var", value.name = "val", measure.vars = grep("^t$|wind_dir_bin|season",
        names(datadf), invert = T, value = T))
```

```
## Warning in melt.data.table(datadf[, !c("t", "wind_dir_bin", "season")], :
## 'measure.vars' [ln_pm10, ln_cars, temp_2m, wind_speed, ...] are not all
## of the same type. By order of hierarchy, the molten data value column will
## be of type 'double'. All measure variables not of type 'double' will be
## coerced to. Check DETAILS in ?melt.data.table for more on coercion.
```

```
ggplot(data = distdata) + geom_density(aes(x = val, fill = var)) +
    scale_fill_brewer(palette = "Set2") + facet_wrap(~var, scales = "free") +
    guides(fill = F) + theme_bw()
```



```
# Examine bivariate relationships

panel.cor <- function(x, y, digits = 2, prefix = "", cex.cor,
    ...) {
    usr <- par("usr")
    on.exit(par(usr))
    par(usr = c(0, 1, 0, 1))
    r <- abs(cor(x, y))
    txt <- format(c(r, 0.123456789), digits = digits)[1]
    txt <- paste(prefix, txt, sep = "")
    if (missing(cex.cor))
        cex.cor <- 0.8/strwidth(txt)
    text(0.5, 0.5, txt, cex = 0.4 + cex.cor * r)
}

pairs(datadf[, !c("t", "wind_dir_bin", "season")], lower.panel = panel.smooth,
    upper.panel = panel.cor)
```



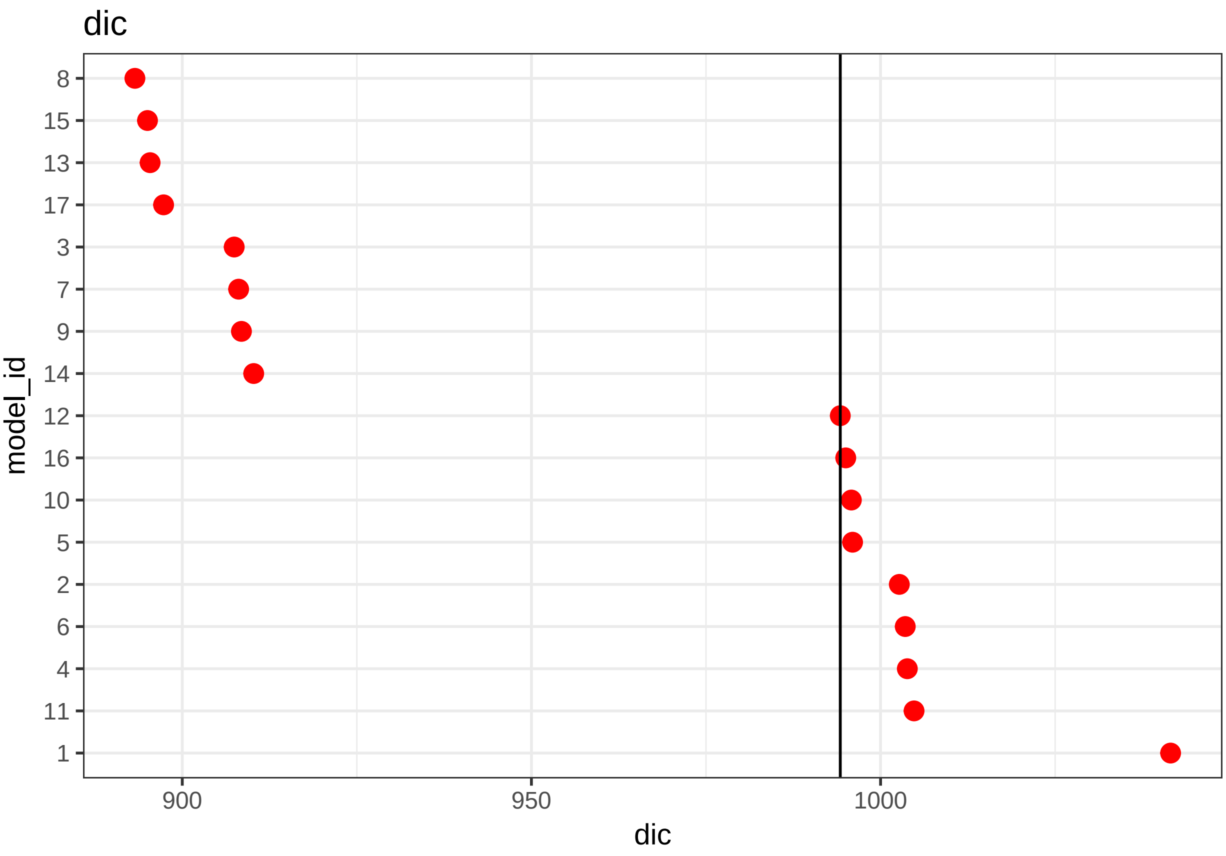
```
## Creating all combinations of formulas
form_bases <- c(as.formula("ln_pm10~1 + f(hour, model = 'rw1') + f(as.factor(season), model = 'iid')"),
    as.formula("ln_pm10~1 + f(t, model ='rw1')"), as.formula("ln_pm10~1 + f(day, model = 'iid') + f(hou

form_base <- form_bases[2]

listcombo <- unlist(sapply(0:4, function(x) combn(4, x, simplify = FALSE)),
    recursive = FALSE)
predterms <- lapply(listcombo, function(x) paste(c(form_base,
    c("ln_cars", "temp_2m", "wind_speed", "temp_diff_225", "as.factor(wind_dir_bin)")[x]),
    collapse = " + ")) %>% unlist

predterms <- c("ln_pm10~1", predterms)

# Model
resultsdf <- {
}
set.seed(98109)
seeds <- sample(10000, 5)
print(seeds)
```

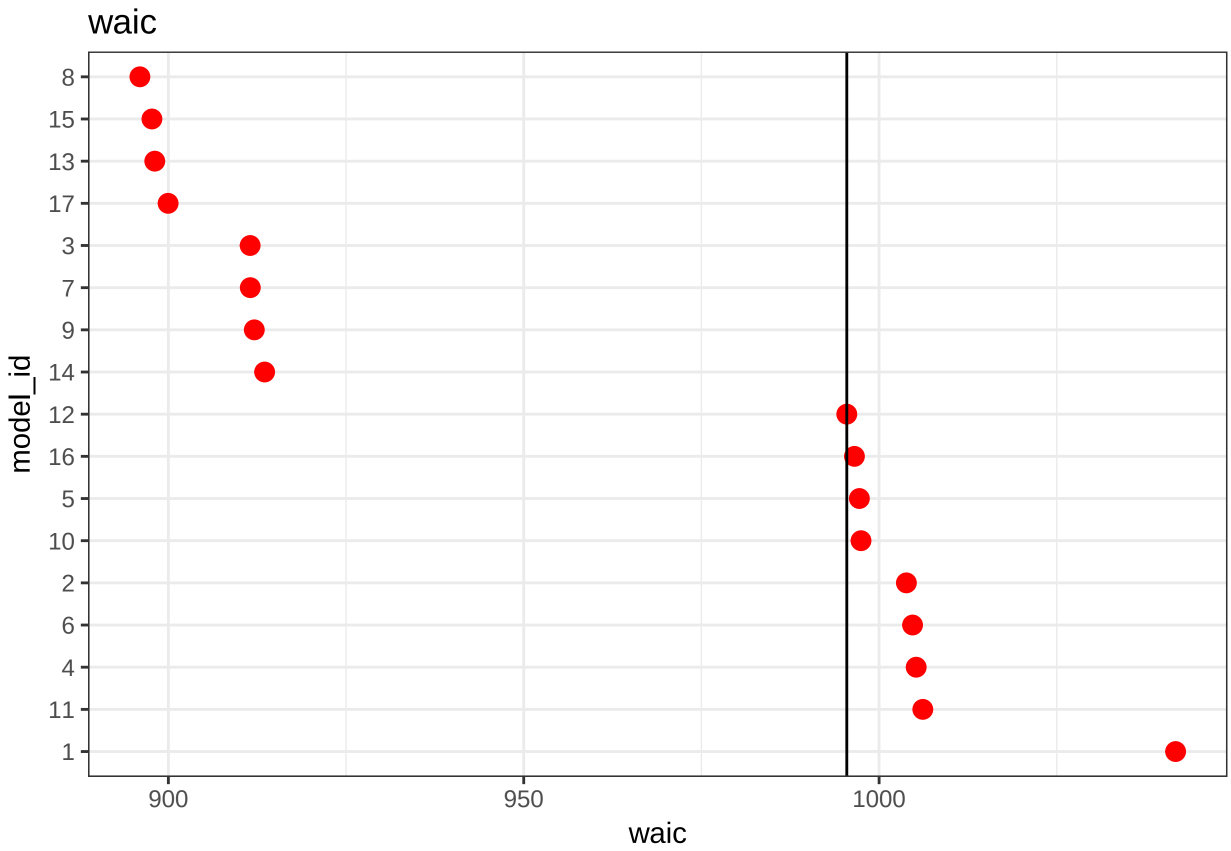
```
## [1] 6680 9834 9534 4349 1142
```

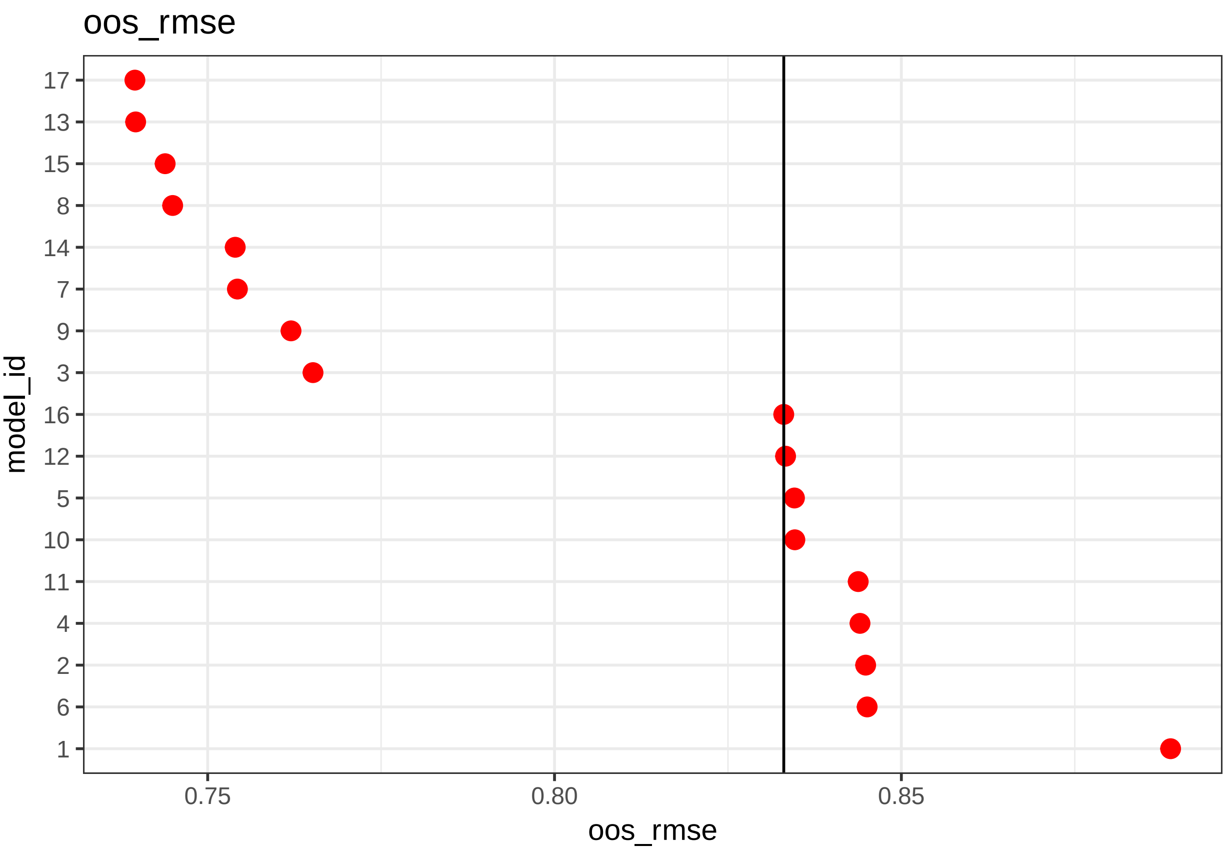
```
for (s in seeds) {

    message(paste0("Setting seed to ", s))

    ## Splitting data into train and test sets
    set.seed(s)
    datadf[, `:=`(samp, sample(.N, replace = F))]
    datadf <- datadf[order(samp)]
    designdf <- copy(datadf)[samp > 400, `:=`(ln_pm10, NA)]

    for (i in 1:length(predterms)) {

        message(paste0("FITTING ", predterms[i]))
        mod <- inla(formula = as.formula(predterms[i]), data = designdf,
            control.predictor = list(compute = T), control.compute = list(dic = T,
                waic = T), family = "gaussian")

        rmse <- (mod$summary.fitted.values[401:500, 1] - datadf[samp %in%
            401:500, ln_pm10])^2 %>% mean %>% sqrt

        results <- data.table(seed = s, model_id = i, model_form = predterms[i],
            waic = mod$waic$waic, dic = mod$dic$dic, oos_rmse = rmse)

        resultsdf <- rbind(resultsdf, results, use.names = T,
            fill = T)

    }

}
```

```
## Setting seed to 6680

## FITTING ln_pm10~1

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1")

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_diff_225

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_diff_225

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + temp_diff_225

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed + temp_diff_225

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + temp_diff_225

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed + temp_diff_225

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed + temp_diff_225

## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed + temp_diff_225

## Setting seed to 9834
## FITTING ln_pm10~1
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1")
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed + temp_diff_225
## Setting seed to 9534
## FITTING ln_pm10~1
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1")
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed + temp_diff_225
## Setting seed to 4349
## FITTING ln_pm10~1
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1")
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed + temp_diff_225
## Setting seed to 1142
## FITTING ln_pm10~1
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1")
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + temp_2m + wind_speed + temp_diff_225
## FITTING ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + temp_2m + wind_speed + temp_diff_225
```

```
# Summarize
summdf <- resultsdf[, lapply(.SD, function(x) as.numeric(mean(x))),
    by = .(model_id, model_form), .SDcols = c("dic", "waic",
        "oos_rmse")]

# Evaluate
plotlist <- {
}

for (plotvar in c("dic", "waic", "oos_rmse")) {

    p <- ggplot(data = summdf) + geom_point(aes(x = get(plotvar),
        y = factor(model_id, levels = summdf[order(-get(plotvar))]$model_id)),
        color = "red", size = 3) + geom_vline(xintercept = median(summdf[,
        get(plotvar)])) + theme_bw() + labs(title = plotvar,
        x = plotvar, y = "model_id")

    print(p)

}
```




waic
model_id
8
15
13
17
3
7
9
14
12
16
5
10
2
6
4
11
1
900
950
1000
waic



```r
# summdf[as.numeric(oos_rmse) == min(as.numeric(oos_rmse)),
# oos_rmse := paste0('*', oos_rmse)] summdf[as.numeric(waic)
# == min(as.numeric(waic)), waic := paste0('*',waic)]

kable(summdf[order(oos_rmse), !c("dic", "waic", "oos_rmse")],
    digits = 4, caption = "In-and Out of Sample Performance for Tested Models")
```

Table 1: In-and Out of Sample Performance for Tested Models

model_id	model_form
17	ln_pm10 ~ 1 + f(t, model = “rw1”) + ln_cars + temp_2m + wind_speed + temp_diff_225
13	ln_pm10 ~ 1 + f(t, model = “rw1”) + ln_cars + temp_2m + wind_speed
15	ln_pm10 ~ 1 + f(t, model = “rw1”) + ln_cars + wind_speed + temp_diff_225
8	ln_pm10 ~ 1 + f(t, model = “rw1”) + ln_cars + wind_speed
14	ln_pm10 ~ 1 + f(t, model = “rw1”) + ln_cars + temp_2m + temp_diff_225
7	ln_pm10 ~ 1 + f(t, model = “rw1”) + ln_cars + temp_2m
9	ln_pm10 ~ 1 + f(t, model = “rw1”) + ln_cars + temp_diff_225
3	ln_pm10 ~ 1 + f(t, model = “rw1”) + ln_cars
16	ln_pm10 ~ 1 + f(t, model = “rw1”) + temp_2m + wind_speed + temp_diff_225
12	ln_pm10 ~ 1 + f(t, model = “rw1”) + wind_speed + temp_diff_225
5	ln_pm10 ~ 1 + f(t, model = “rw1”) + wind_speed
10	ln_pm10 ~ 1 + f(t, model = “rw1”) + temp_2m + wind_speed
11	ln_pm10 ~ 1 + f(t, model = “rw1”) + temp_2m + temp_diff_225
4	ln_pm10 ~ 1 + f(t, model = “rw1”) + temp_2m
2	ln_pm10 ~ 1 + f(t, model = “rw1”)
6	ln_pm10 ~ 1 + f(t, model = “rw1”) + temp_diff_225

model_id	model_form
1	ln_pm10~1

The best model based on in-sample fit, out-of-sample performance, and parsimony seems to be ln_pm10 ~ 1 + f(t, model = "rw1") + ln_cars + wind_speed.

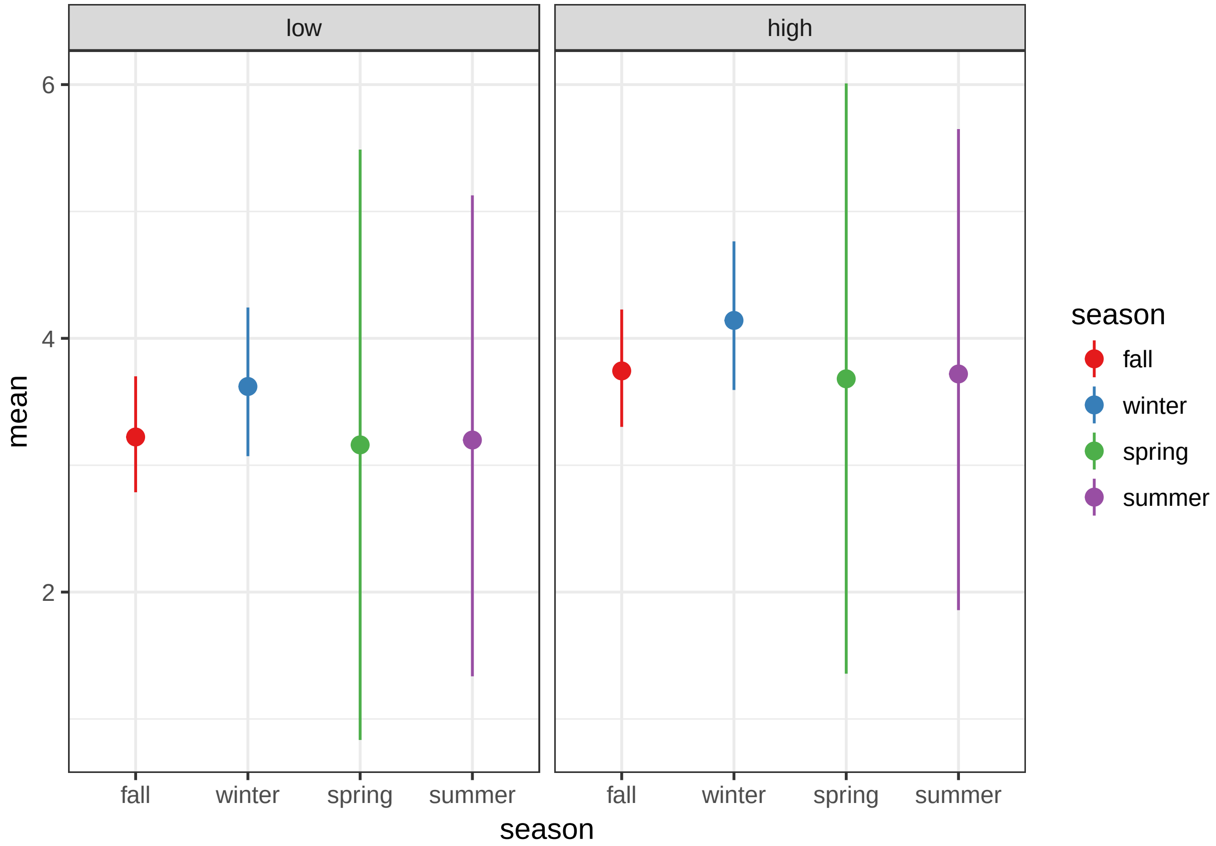
```
best_form <- summdf[model_id == 8, model_form]

# Pick reference dates in each of four seasons
ref_dates <- c("2001-01-01", "2001-04-01", "2001-07-01", "2001-10-01") %>%
    as.Date
ref_dates <- ref_dates - as.Date("2000-10-01")

# Convert to hours since day 1
ref_dates <- 24 * ref_dates %>% as.numeric
ref_hours <- sort(rep(ref_dates, 24)) + rep(1:24, length(ref_dates))

# Pick reference number of cars as quantiles
ref_ln_cars <- quantile(datadf$ln_cars, probs = c(0.25, 0.5,
    0.75))

# Pick reference wind speed as average wind speed
ref_wind_speed <- mean(datadf$wind_speed)

# Combine into prediction template
preddf <- data.table(expand.grid(ln_pm10 = NA_real_, ln_cars = ref_ln_cars,
    t = ref_hours, wind_speed = ref_wind_speed))

preddf <- preddf[order(t)]

preddf[, `:=`(ref_dates, sort(rep(ref_dates, 72)))]

best_data <- rbind(datadf[, .(ln_pm10, ln_cars, t, wind_speed)],
    preddf, use.names = T, fill = T)

best_mod <- inla(formula = as.formula(best_form), data = best_data,
    control.predictor = list(compute = T), control.compute = list(dic = T,
        waic = T), family = "gaussian")

# Table of posteriors
kable(best_mod$summary.fixed[, c(1:3, 5)], row.names = T, caption = "Posterior Estimates of Fixed Effec
```

Table 2: Posterior Estimates of Fixed Effects, Best Model

	mean	sd	0.025quant	0.975quant
(Intercept)	1.3150738	0.2241708	0.8742536	1.7546436
ln_cars	0.3322827	0.0302955	0.2727131	0.3916933
wind_speed	-0.1025599	0.0193295	-0.1404895	-0.0645947

```
best_pred <- cbind(preddf[, !"ln_pm10"], best_mod$summary.fitted.values[501:788,
    ])
```

```
# Factorize variables
best_pred[, `:=`(season, factor(round_any(t, 1000, floor), levels = seq(2000,
    8000, 2000), labels = c("fall", "winter", "spring", "summer")))]
best_pred[, `:=`(h, t%%24)]
best_pred[, `:=`(traffic, factor(ln_cars, levels = ref_ln_cars,
    labels = c("low", "medium", "high")))]

# Plot
ggplot(data = best_pred[traffic != "medium" & h == 12], aes(x = season,
    y = mean)) + geom_pointrange(aes(color = season, ymin = get("0.025quant"),
    ymax = get("0.975quant"))) + scale_color_brewer(palette = "Set1") +
    theme_bw() + facet_wrap(~traffic)
```


low
high
mean
fall
winter
spring
summer
fall
winter
spring
summer
season
season
fall
winter
spring
summer