

# Non-linear models

Brooke Anderson

March 10, 2016

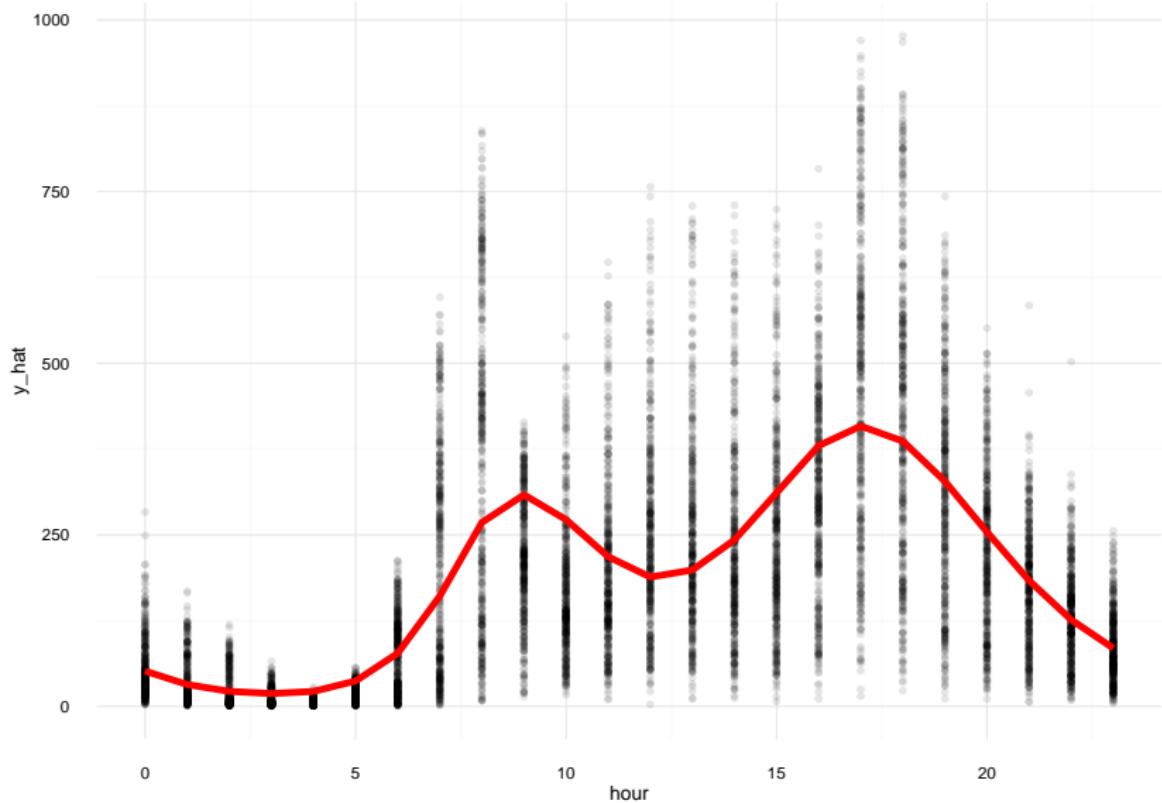
## Spline of hour

```
ex_1 <- glm(count ~ ns(hour, 6), data = train,
             family = quasipoisson)
newdata <- data.frame(hour = 0:23)
newdata$y_hat <- predict(ex_1, newdata = newdata,
                         type = "response")

ggplot(newdata, aes(x = hour, y = y_hat)) +
  geom_point(data = train, aes(x = hour, y = count),
             alpha = 0.1) +
  geom_line(color = "red", size = 2) +
  theme_minimal()
```

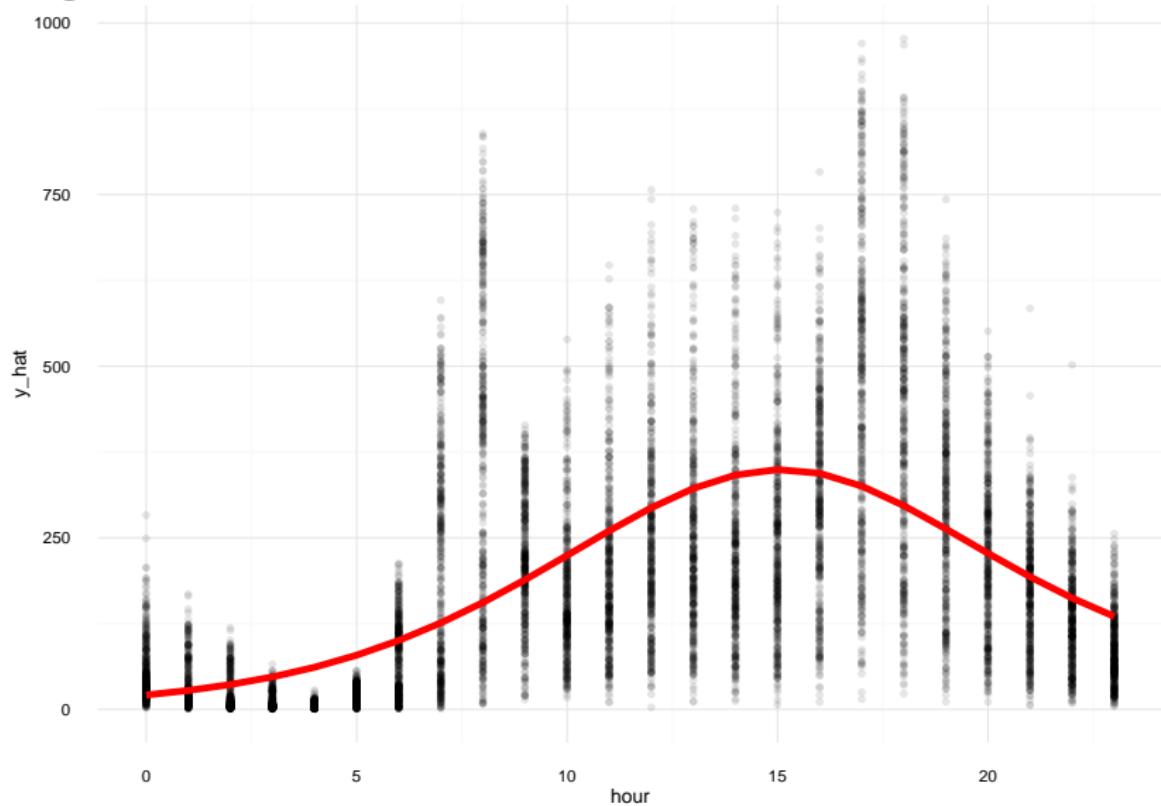
# Spline of hour

Degrees of freedom: 6



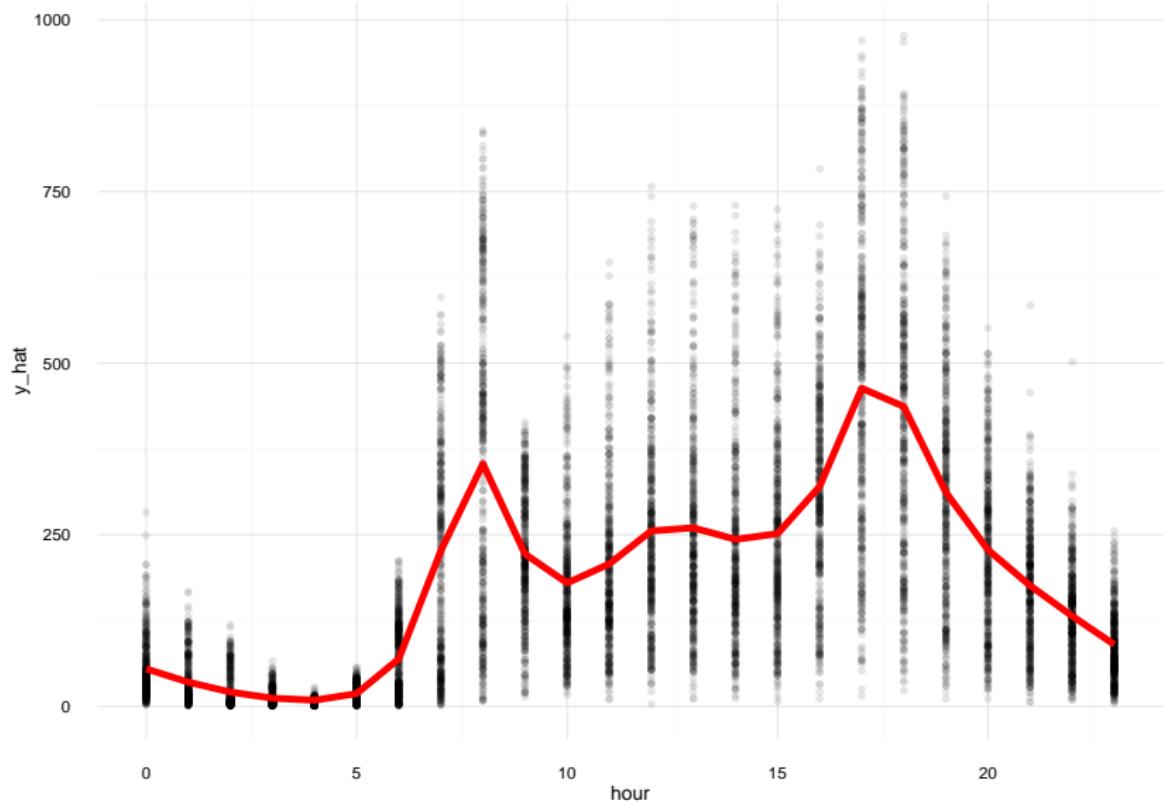
# Spline of hour

Degrees of freedom: 3



# Spline of hour

Degrees of freedom: 15



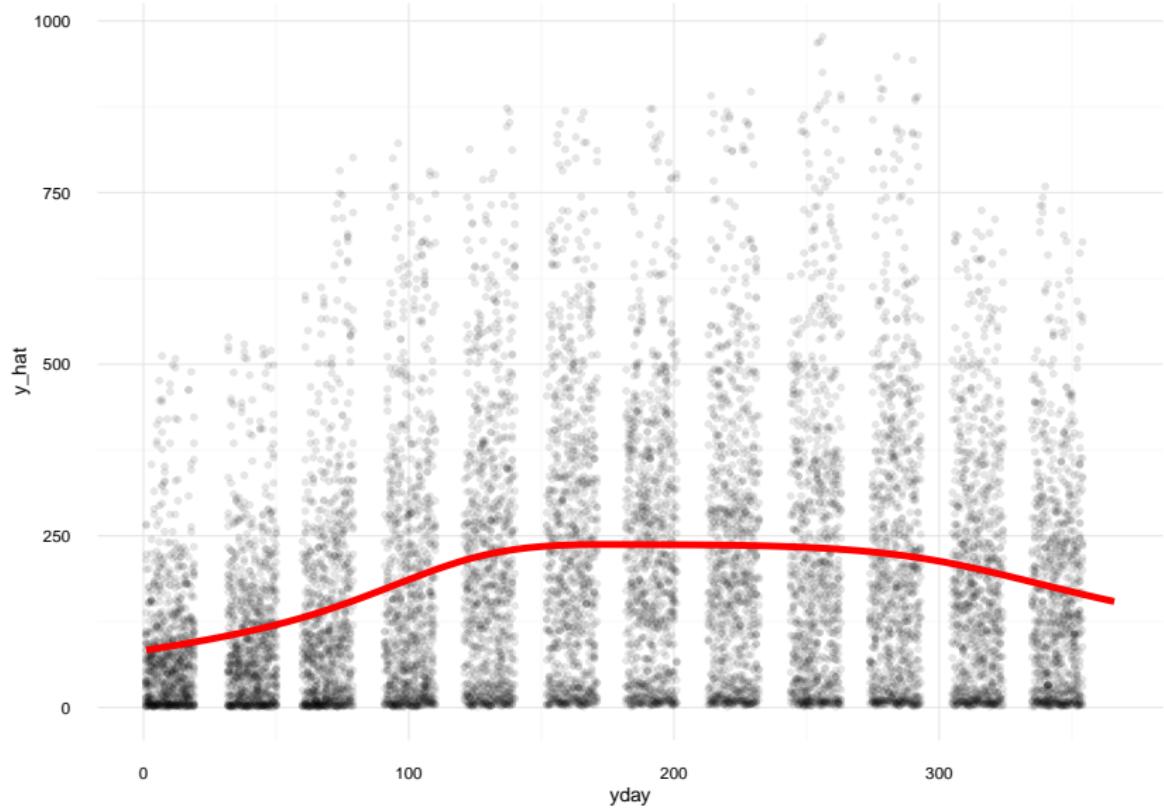
## Spline of day of year

```
ex_1 <- glm(count ~ ns(yday, 6), data = train,
             family = quasipoisson)
newdata <- data.frame(yday = 1:366)
newdata$y_hat <- predict(ex_1, newdata = newdata,
                         type = "response")

ggplot(newdata, aes(x = yday, y = y_hat)) +
  geom_point(data = train, aes(x = yday, y = count),
             alpha = 0.1) +
  geom_line(color = "red", size = 2) +
  theme_minimal()
```

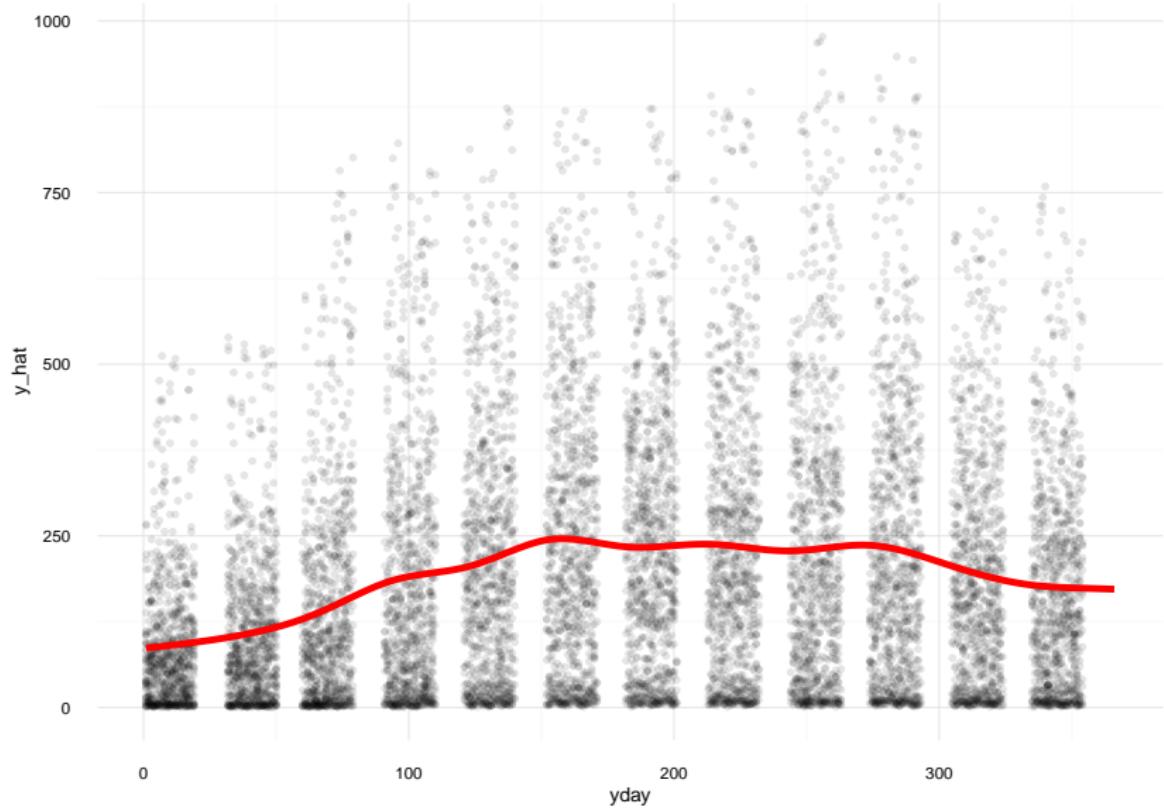
# Spline of day of year

Degrees of freedom: 6



# Spline of day of year

Degrees of freedom: 12



# Model

On Kaggle: 0.77821.

```
mod_1 <- glm(count ~ ns(hour, 12) + ns(yday, 6),  
              data = train, family = quasipoisson)
```

```
train_preds <- predict(mod_1, type = "response")  
actual_preds <- train$count  
rmsle(train_preds, actual_preds)
```

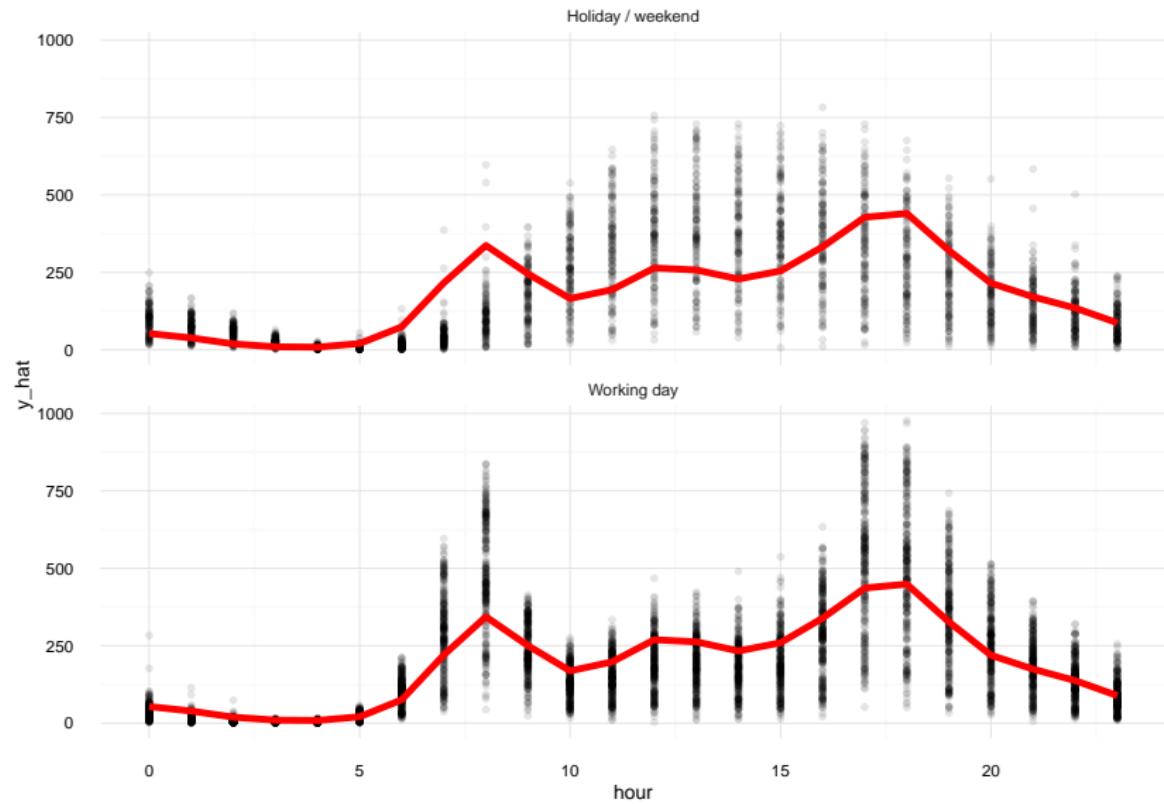
```
## [1] 0.7103122
```

```
test_preds <- predict(mod_1, newdata = test,  
                      type = "response")  
write_test_preds(test_preds, mod_name = "gam_1")
```

## Spline of hour

```
ex_2 <- glm(count ~ workingday + ns(hour, 12),  
             data = train,  
             family = quasipoisson)  
newdata <- data.frame(hour = rep(0:23, 2),  
                       workingday = rep(unique(train$workingday),  
                                         each = 24))  
newdata$y_hat <- predict(ex_2, newdata = newdata,  
                         type = "response")  
  
ggplot(newdata, aes(x = hour, y = y_hat)) +  
  geom_point(data = train, aes(x = hour, y = count),  
              alpha = 0.1) +  
  geom_line(color = "red", size = 2) +  
  facet_wrap(~ workingday, ncol = 1) +  
  theme_minimal()
```

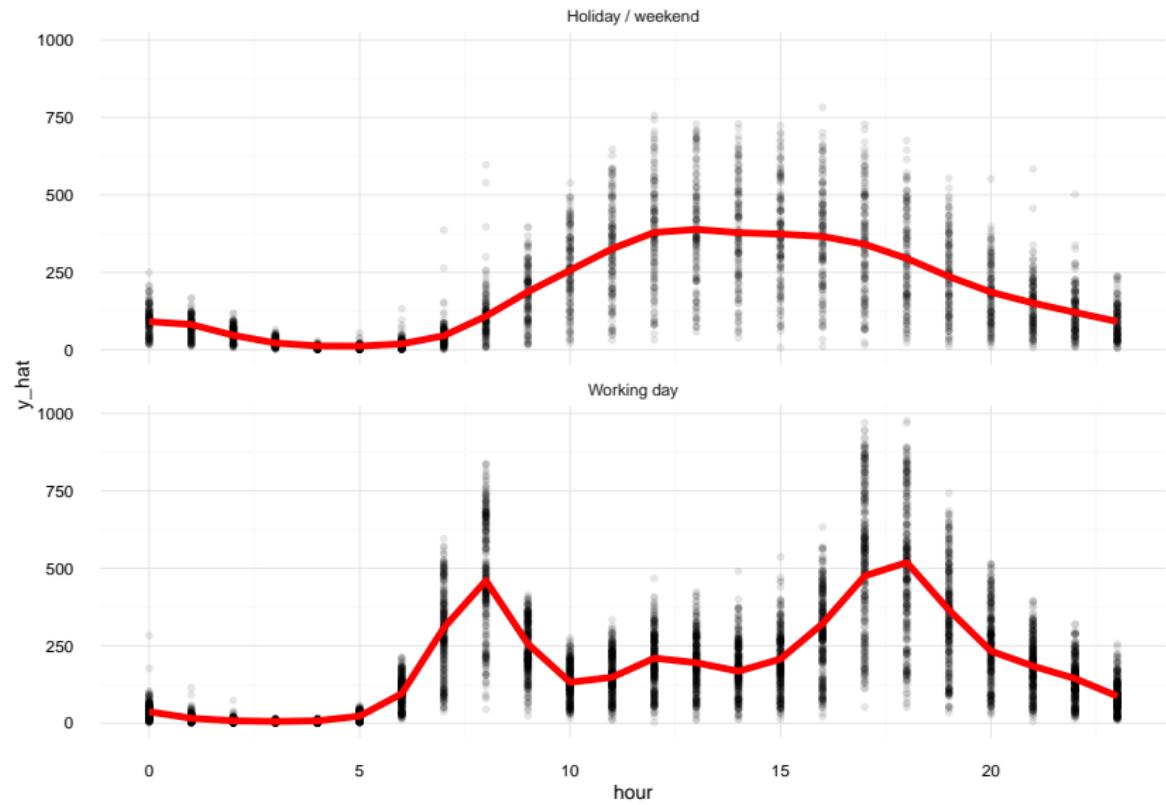
# Spline of hour



## Spline of hour

```
ex_2 <- glm(count ~ workingday*ns(hour, 12) ,  
             data = train,  
             family = quasipoisson)  
newdata <- data.frame(hour = rep(0:23, 2) ,  
                       workingday = rep(unique(train$workingday) ,  
                                         each = 24))  
newdata$y_hat <- predict(ex_2, newdata = newdata,  
                         type = "response")  
  
ggplot(newdata, aes(x = hour, y = y_hat)) +  
  geom_point(data = train, aes(x = hour, y = count) ,  
              alpha = 0.1) +  
  geom_line(color = "red", size = 2) +  
  facet_wrap(~ workingday, ncol = 1) +  
  theme_minimal()
```

# Spline of hour



# Model

On Kaggle: 0.60629.

```
mod_2 <- glm(count ~ workingday * ns(hour, 12) +  
              ns(yday, 6),  
              data = train, family = quasipoisson)
```

```
train_preds <- predict(mod_2, type = "response")  
actual_preds <- train$count  
rmsle(train_preds, actual_preds)
```

```
## [1] 0.509331
```

```
test_preds <- predict(mod_2, newdata = test,  
                      type = "response")  
write_test_preds(test_preds, mod_name = "gam_2")
```

## Spline of hour

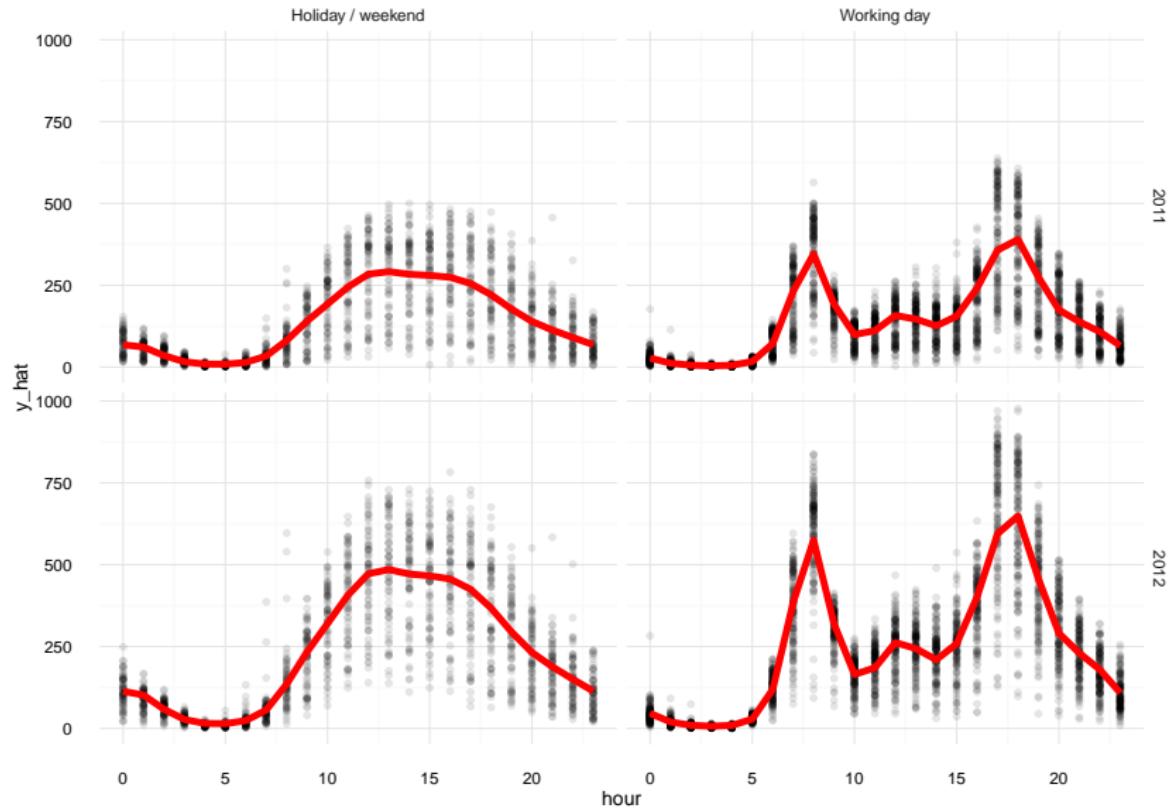
```
ex_3 <- glm(count ~ year + workingday*ns(hour, 12) ,  
             data = train,  
             family = quasipoisson)  
  
new_hour <- 0:23  
new_workingday <- levels(train$workingday)  
new_year <- unique(train$year)  
  
newdata <- expand.grid(new_hour, new_workingday, new_year)  
colnames(newdata) <- c("hour", "workingday", "year")  
head(newdata, 3)
```

```
##      hour      workingday year  
## 1      0 Holiday / weekend 2011  
## 2      1 Holiday / weekend 2011  
## 3      2 Holiday / weekend 2011
```

## Spline of hour

```
newdata$y_hat <- predict(ex_3, newdata = newdata,  
                         type = "response")  
  
ggplot(newdata, aes(x = hour, y = y_hat)) +  
  geom_point(data = train, aes(x = hour, y = count),  
             alpha = 0.1) +  
  geom_line(color = "red", size = 2) +  
  facet_grid(year ~ workingday) +  
  theme_minimal()
```

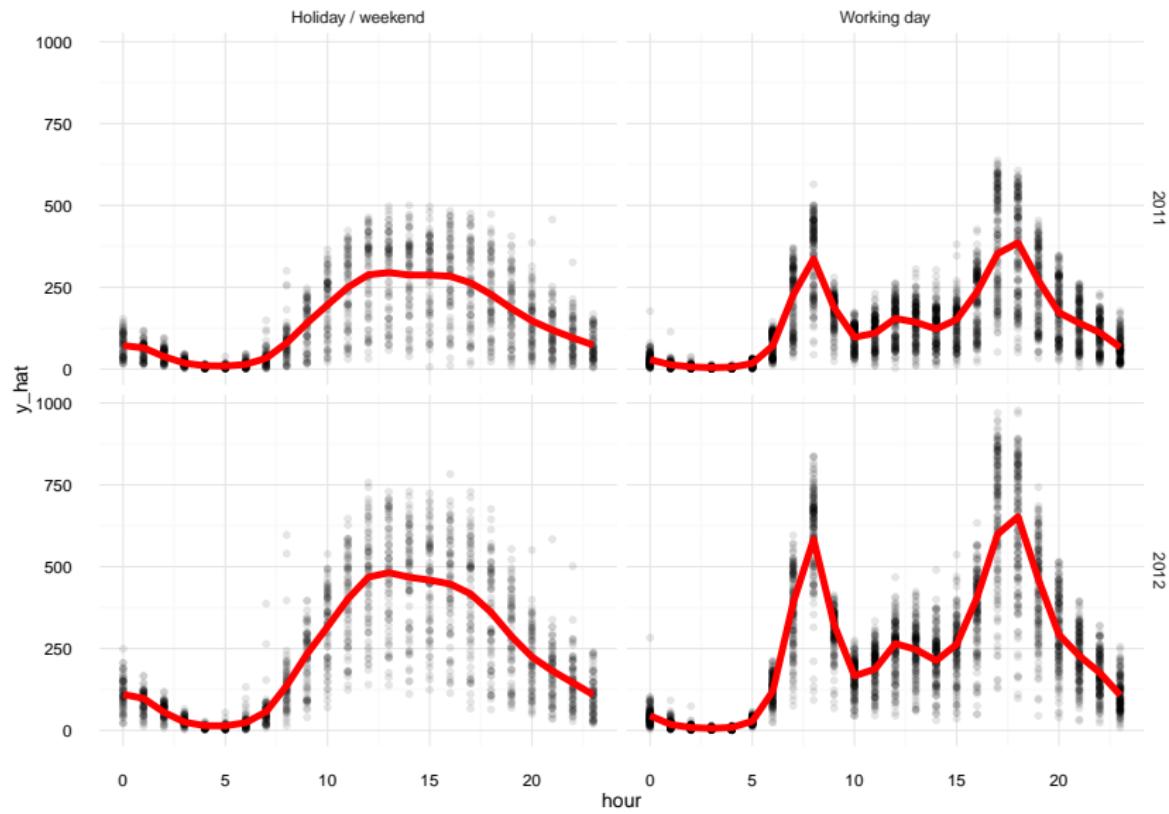
# Spline of hour



## Spline of hour

```
ex_4 <- glm(count ~ year*workingday*ns(hour, 12),  
             data = train,  
             family = quasipoisson)  
  
new_hour <- 0:23  
new_workingday <- levels(train$workingday)  
new_year <- unique(train$year)  
  
newdata <- expand.grid(new_hour, new_workingday, new_year)  
colnames(newdata) <- c("hour", "workingday", "year")  
  
newdata$y_hat <- predict(ex_4, newdata = newdata,  
                         type = "response")
```

# Spline of hour



# Model

On Kaggle: 0.54972.

```
mod_3 <- glm(count ~ workingday * ns(hour, 12) +
               ns(yday, 6) + year,
               data = train, family = quasipoisson)

train_preds <- predict(mod_3, type = "response")
actual_preds <- train$count
rmsle(train_preds, actual_preds)

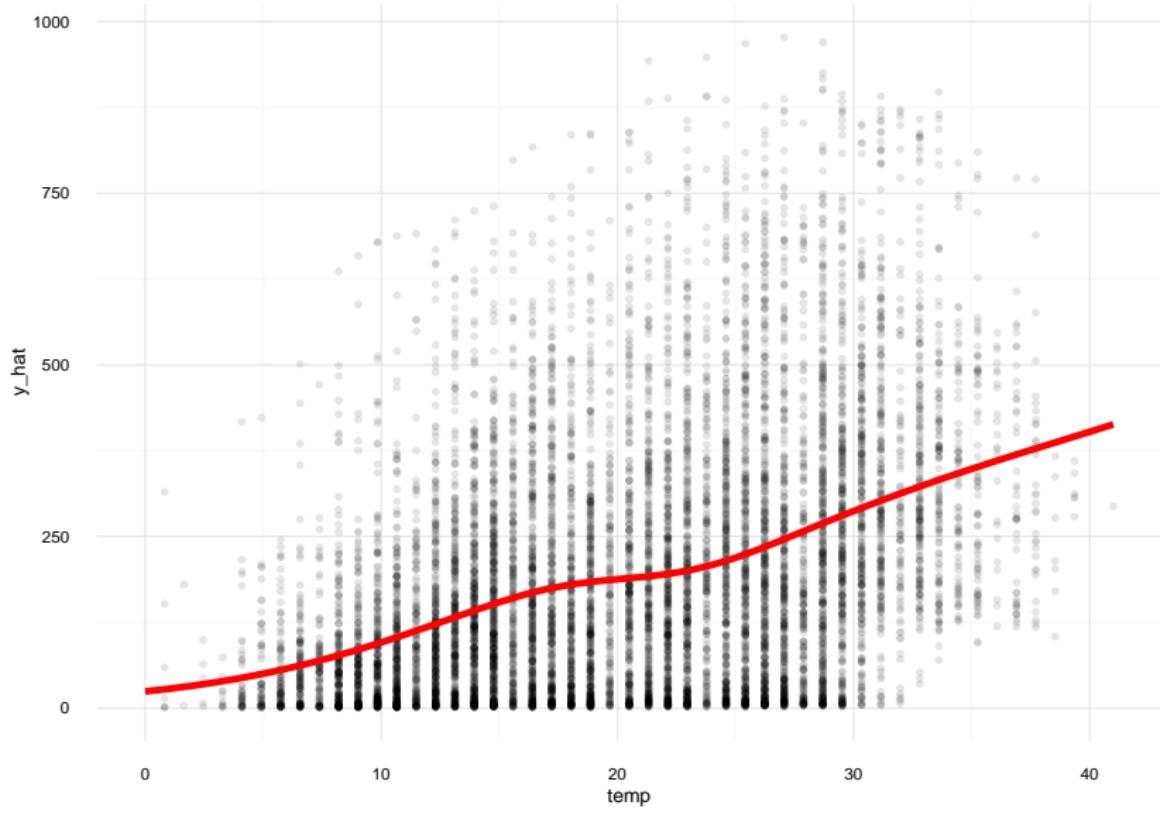
## [1] 0.4290578
```

```
test_preds <- predict(mod_3, newdata = test,
                       type = "response")
write_test_preds(test_preds, mod_name = "gam_3")
```

## Spline of temperature

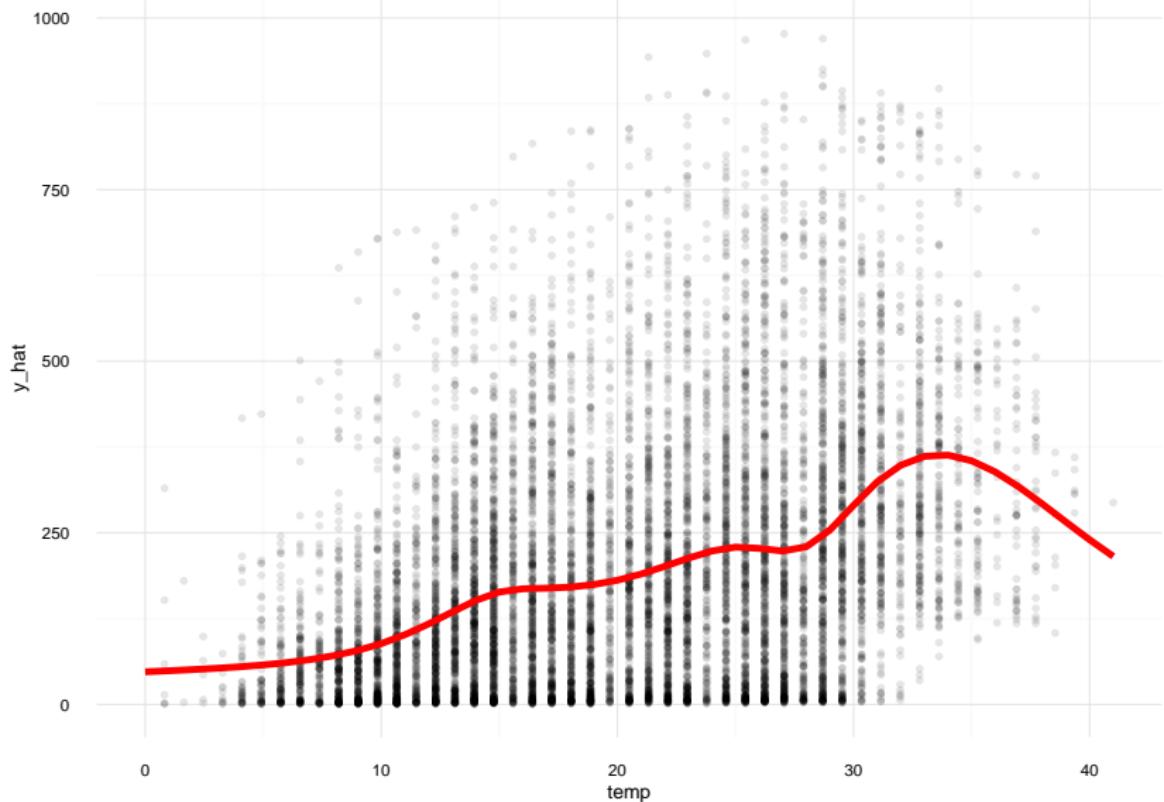
```
ex_5 <- glm(count ~ ns(temp, 5),  
             data = train,  
             family = quasipoisson)  
  
newdata <- data.frame(temp = seq(from =  
                                 floor(min(train$temp)),  
                                 to = max(train$temp)))  
  
newdata$y_hat <- predict(ex_5, newdata = newdata,  
                         type = "response")
```

# Spline of temperature



# Spline of temperature

Degrees of freedom: 10.



# Model

On Kaggle: 0.51936.

```
mod_4 <- glm(count ~ workingday * ns(hour, 12) +  
              ns(yday, 6) + year +  
              ns(temp, 10),  
              data = train, family = quasipoisson)
```

```
train_preds <- predict(mod_4, type = "response")  
actual_preds <- train$count  
rmsle(train_preds, actual_preds)
```

```
## [1] 0.4137786
```

```
test_preds <- predict(mod_4, newdata = test,  
                      type = "response")  
write_test_preds(test_preds, mod_name = "gam_4")
```

## Model

On Kaggle: 0.46740.

```
mod_5 <- glm(count ~ season * workingday * ns(hour, 12) +
               ns(yday, 6) + year +
               ns(temp, 10) + weather + wday +
               ns(humidity, 5),
               data = train, family = quasipoisson)
```

```
train_preds <- predict(mod_5, type = "response")
actual_preds <- train$count
rmsle(train_preds, actual_preds)
```

```
## [1] 0.3580469
```

```
test_preds <- predict(mod_5, newdata = test,
                      type = "response")
write_test_preds(test_preds, mod_name = "gam_5")
```

## Spline of hour

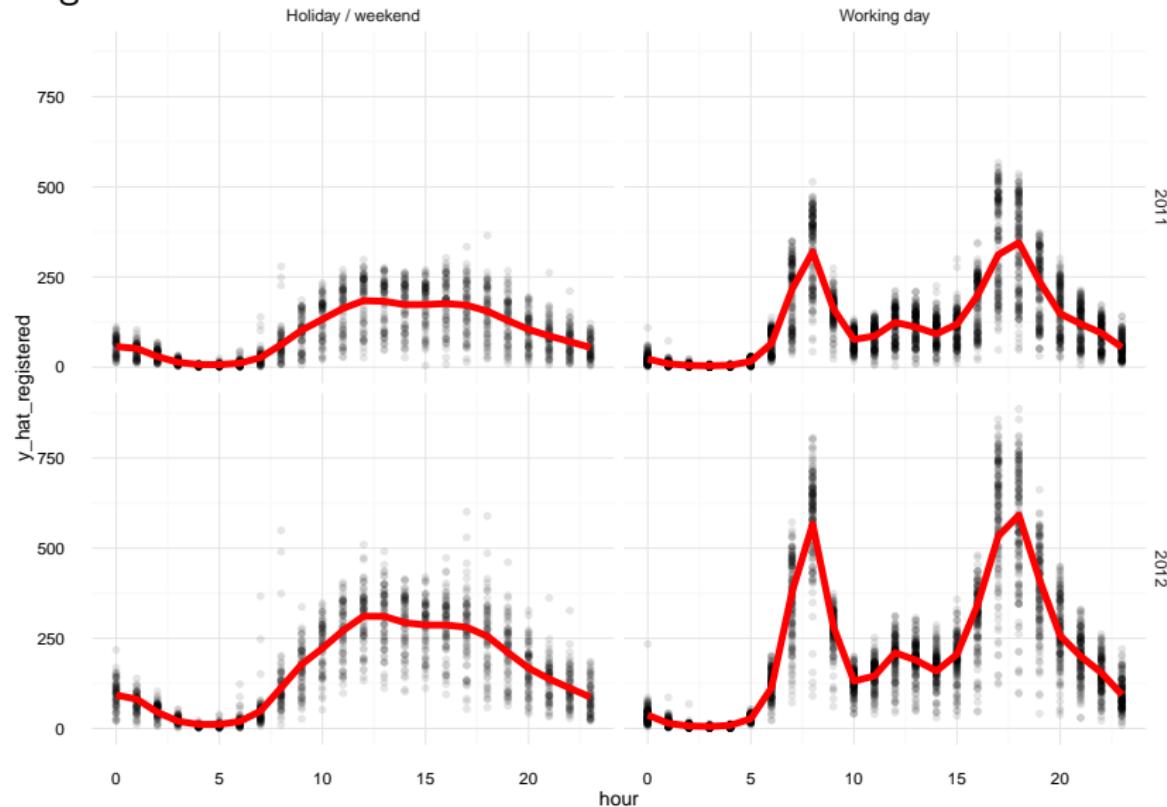
```
ex_5a <- glm(registered ~ year*workingday*ns(hour, 12) ,  
               data = train,  
               family = quasipoisson)  
ex_5b <- glm(casual ~ year*workingday*ns(hour, 12) ,  
               data = train,  
               family = quasipoisson)  
  
new_hour <- 0:23  
new_workingday <- levels(train$workingday)  
new_year <- unique(train$year)
```

## Spline of hour

```
newdata <- expand.grid(new_hour, new_workingday,  
                      new_year)  
colnames(newdata) <- c("hour", "workingday", "year")  
  
newdata$y_hat_registered <- predict(ex_5a,  
                                      newdata = newdata,  
                                      type = "response")  
newdata$y_hat_casual <- predict(ex_5b,  
                                 newdata = newdata,  
                                 type = "response")
```

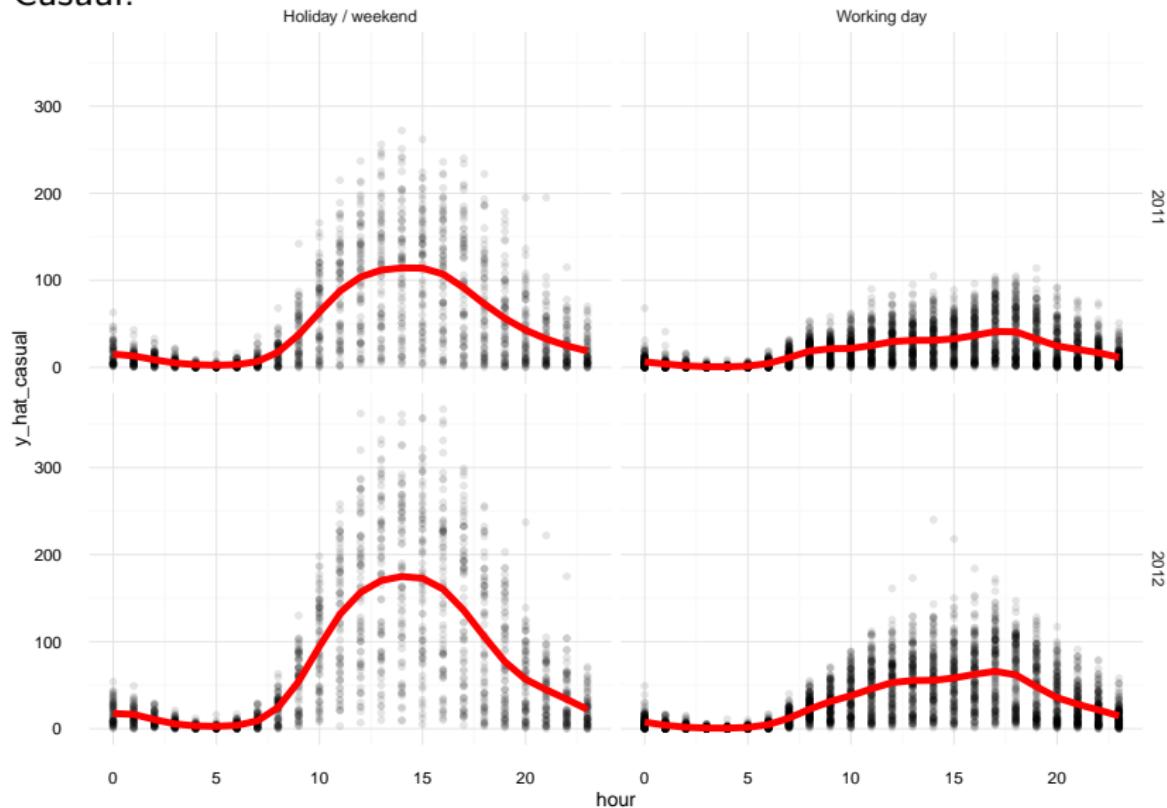
# Spline of hour

Registered:



# Spline of hour

Casual:



## Model

```
mod_6a <- glm(registered ~ season * workingday *  
                 ns(hour, 12) +  
                 ns(yday, 6) + year +  
                 ns(temp, 10) + weather + wday +  
                 ns(humidity, 5),  
                 data = train, family = quasipoisson)  
mod_6b <- glm(casual ~ season * workingday *  
                 ns(hour, 12) +  
                 ns(yday, 6) + year +  
                 ns(temp, 10) + weather + wday +  
                 ns(humidity, 5),  
                 data = train, family = quasipoisson)
```

# Model

```
train_predsa <- predict(mod_6a, type = "response")
train_predsb <- predict(mod_6b, type = "response")
train_preds <- train_predsa + train_predsb
actual_preds <- train$count
rmsle(train_preds, actual_preds)

## [1] 0.3555703
```

# Model

On Kaggle: 0.47690.

```
test_predsa <- predict(mod_6a, newdata = test,
                       type = "response")
test_predsb <- predict(mod_6b, newdata = test,
                       type = "response")
test_preds <- test_predsa + test_predsb
write_test_preds(test_preds, mod_name = "gam_6")
```

## K-nearest neighbors

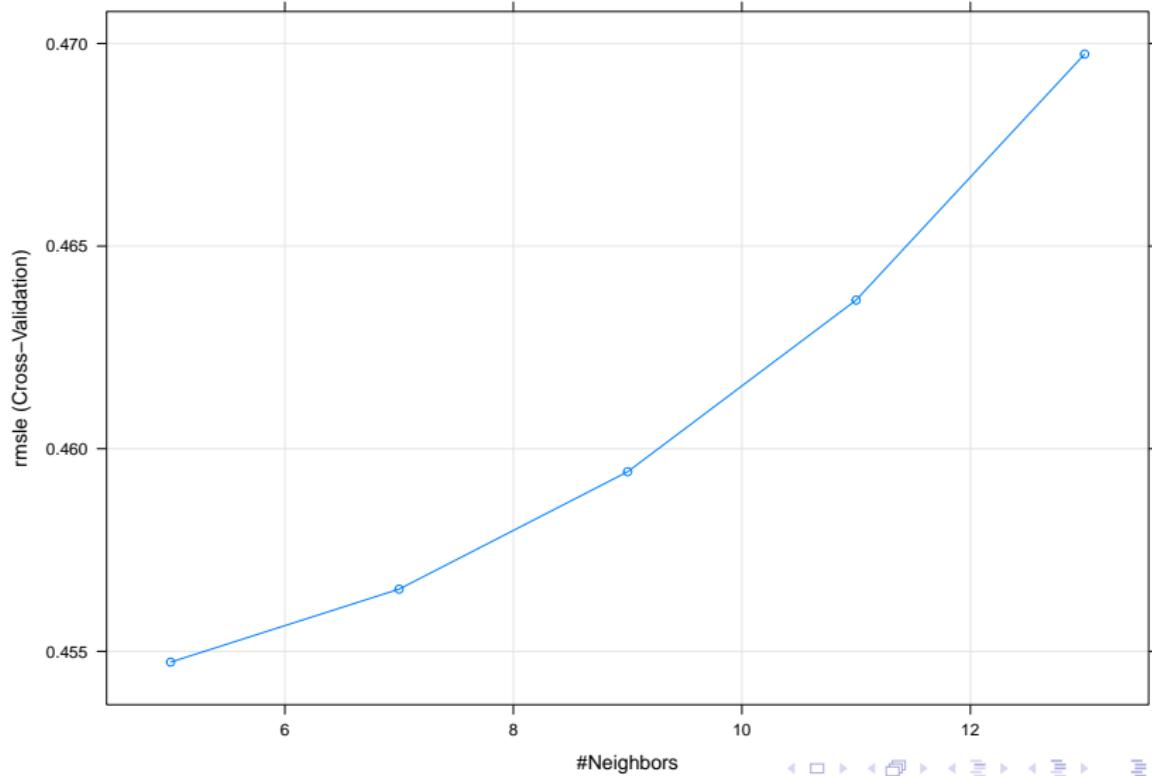
```
set.seed(825)
mod_1 <- train(count ~ temp + hour +
                 workingday + year,
                 data = train,
                 method = "knn",
                 trControl = fitControl,
                 metric = "rmsle",
                 maximize = FALSE,
                 preProcess = c("center", "scale",
                               "spatialSign"),
                 tuneLength = 5)

train_preds <- predict(mod_1,
                       newdata = train)
rmsle(train_preds, train$count)

## [1] 0.3958522
```

# K-nearest neighbors

```
plot(mod_1)
```



## K-nearest neighbors

On Kaggle: 0.49535.

```
test_preds <- predict(mod_1, newdata = test)
write_test_preds(test_preds, mod_name = "knn")
```

## K-nearest neighbors

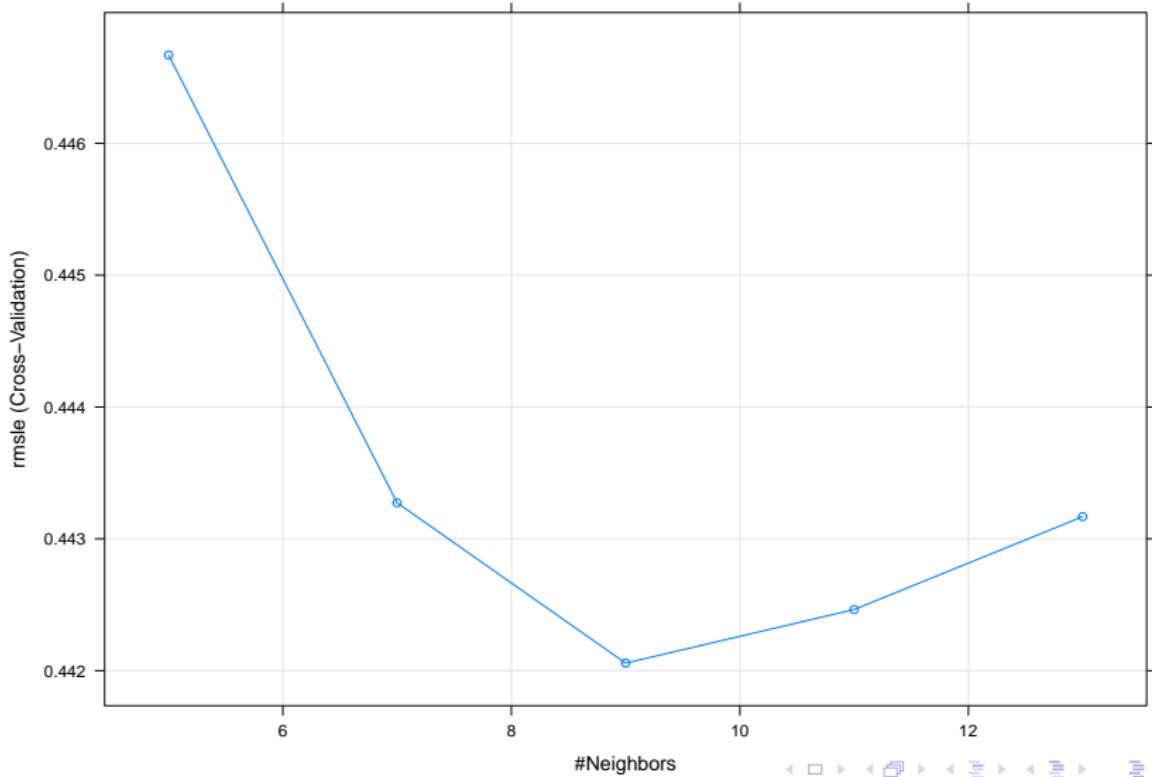
```
set.seed(825)
mod_2 <- train(count ~ temp + factor(hour) + workingday + y
                data = train,
                method = "knn",
                trControl = fitControl,
                metric = "rmsle",
                maximize = FALSE,
                preProcess = c("center", "scale",
                              "spatialSign"),
                tuneLength = 5)

train_preds <- predict(mod_2, newdata = train)
rmsle(train_preds, train$count)

## [1] 0.4080661
```

# K-nearest neighbors

```
plot(mod_2)
```



## K-nearest neighbors

On Kaggle: 0.48618.

```
test_preds <- predict(mod_2, newdata = test)
write_test_preds(test_preds, mod_name = "knn2")
```

# SVM

```
set.seed(825)
mod_3 <- train(log(count) ~ temp + hour +
                 workingday + year,
                 data = train,
                 method = "svmRadial",
                 trControl = fitControl,
                 metric = "rmsle",
                 maximize = FALSE,
                 preProcess = c("center", "scale",
                               "spatialSign"),
                 tuneLength = 5)

train_preds <- exp(predict(mod_3,
                           newdata = train))
rmsle(train_preds, train$count)

## [1] 0.7620723
```

# SVM

On Kaggle: 0.77061.

```
test_preds <- exp(predict(mod_3, newdata = test))
write_test_preds(test_preds, mod_name = "svm")
```