## Multiple Linear Regression

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.0.5 v dplyr 1.0.3  
## v tidyr 1.1.2 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(tidymodels)

## -- Attaching packages -------------------------------------- tidymodels 0.1.2 --

## v broom 0.7.3 v recipes 0.1.15  
## v dials 0.0.9 v rsample 0.0.8   
## v infer 0.5.4 v tune 0.1.2   
## v modeldata 0.1.0 v workflows 0.2.1   
## v parsnip 0.1.5 v yardstick 0.0.7

## -- Conflicts ----------------------------------------- tidymodels\_conflicts() --  
## x scales::discard() masks purrr::discard()  
## x dplyr::filter() masks stats::filter()  
## x recipes::fixed() masks stringr::fixed()  
## x dplyr::lag() masks stats::lag()  
## x yardstick::spec() masks readr::spec()  
## x recipes::step() masks stats::step()

library(glmnet)

## Loading required package: Matrix

##   
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':  
##   
## expand, pack, unpack

## Loaded glmnet 4.1

library(GGally)

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library(ggcorrplot)  
library(MASS)

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

library(car)

## Loading required package: carData

##   
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':  
##   
## recode

## The following object is masked from 'package:purrr':  
##   
## some

library(lubridate)

##   
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

library(lmtest)

## Loading required package: zoo

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

bike = read\_csv("bike\_cleaned.csv")

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## instant = col\_double(),  
## dteday = col\_character(),  
## season = col\_character(),  
## mnth = col\_character(),  
## hr = col\_double(),  
## holiday = col\_character(),  
## weekday = col\_character(),  
## workingday = col\_character(),  
## weathersit = col\_character(),  
## temp = col\_double(),  
## atemp = col\_double(),  
## hum = col\_double(),  
## windspeed = col\_double(),  
## casual = col\_double(),  
## registered = col\_double(),  
## count = col\_double()  
## )

bike = bike %>% mutate(dteday = mdy(dteday))

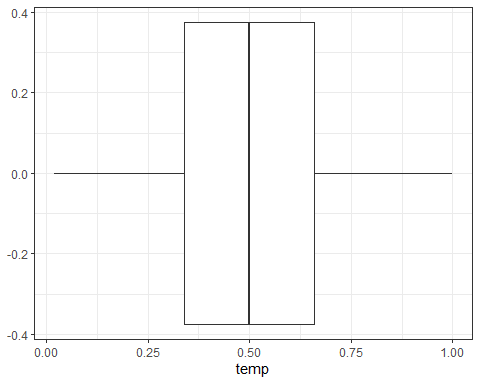
bike1 <- sapply(bike, is.character)  
 bike[bike1] <- lapply(bike[bike1], factor)

bike = bike %>% mutate(hr = as\_factor(hr))

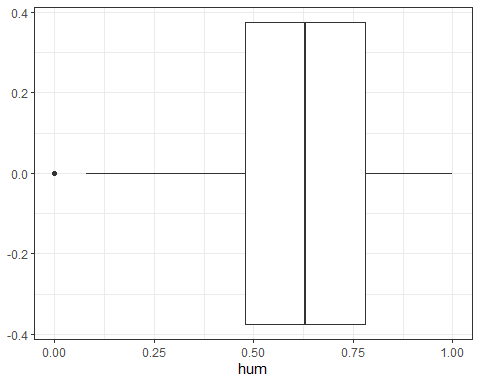
We needed to convert it into a factor so we would be given the full information on hr and make sure there was no data missing.

Temp. appears to be correlated with count.

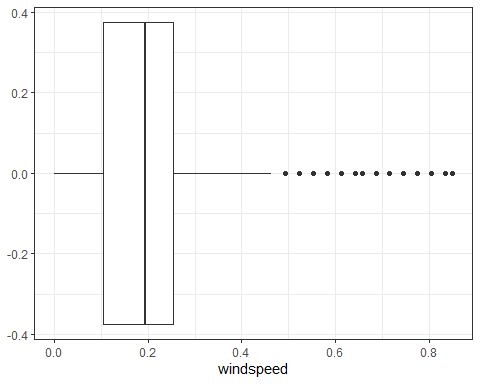
ggplot(bike,aes(x=temp)) + geom\_boxplot() + theme\_bw()

 Compared to hr, temp does appear to affect count.

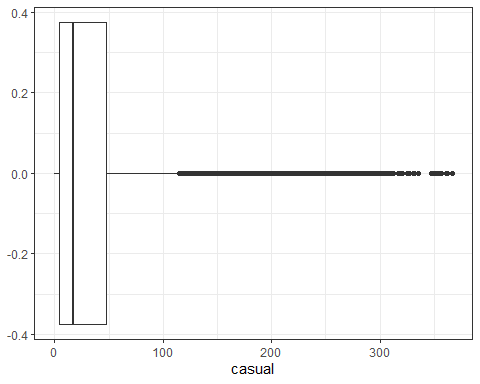
ggplot(bike,aes(x=hum)) + geom\_boxplot() + theme\_bw()

 Humidity could be but not like temp.

ggplot(bike,aes(x=windspeed)) + geom\_boxplot() + theme\_bw()

 Windspeed doesn’t really affect count.

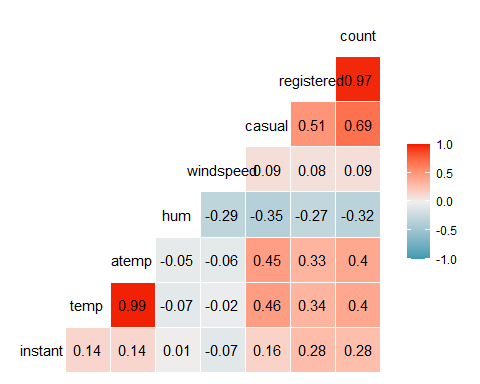
ggplot(bike,aes(x=casual)) + geom\_boxplot() + theme\_bw()



Casual could affect count, but not as much as others.

ggcorr(bike, label="TRUE", label\_round = 2)

## Warning in ggcorr(bike, label = "TRUE", label\_round = 2): data in column(s)  
## 'dteday', 'season', 'mnth', 'hr', 'holiday', 'weekday', 'workingday',  
## 'weathersit' are not numeric and were ignored



This model shows which variables are most likely and least likely to affect count. OVerall it’s a pretty good model.

bike\_recipe = recipe(count ~ temp + atemp+ hum + windspeed, bike) %>%  
 step\_ns(temp, deg\_free = 4) %>%  
 step\_other(hum, threshold = 0.01) %>%  
 step\_dummy(all\_nomial()) %>%  
 step\_center(all\_predictors()) %>%  
 step\_scale(all\_predictors())  
  
  
ridge\_model =  
 linear\_reg(mixture = 0) %>%  
 set\_engine("glmnet")  
  
ridge\_wkflow =   
 workflow() %>%  
 add\_model(ridge\_model) %>%  
 add\_recipe(bike\_recipe)