Homework 10: Predictive Modeling

Gloria Grace

Getting Started

Load data

```
daily_rides <- read_csv("data/day-hw10.csv", col_types = cols_only(
  date = col_date(),
  year = col_integer(),
  workingday = col_character(),
  temp = col_double(),
  atemp = col_double(),
  casual = col_double(),
  registered = col_double()
)) %>% mutate(across(c(workingday, year), as_factor))
```

Exploratory Analytics

```
daily_rides %>%
  ggplot(aes(x = date, y = casual, color = workingday))+
  geom_point(size = 1)
```

Train-test split

```r
train <- daily_rides %>%
  filter(year == '2011')

test <-daily_rides %>%
  filter(year == '2012')
```

In the test set there are 366 set days and 365

Linear Regression using Temperature

```r
model1 <- linear_reg() %>%
  fit(casual ~ temp, data = train)
```

Look inside the model

```r
model1 %>%
  tidy() %>%
  select(term, estimate)
```

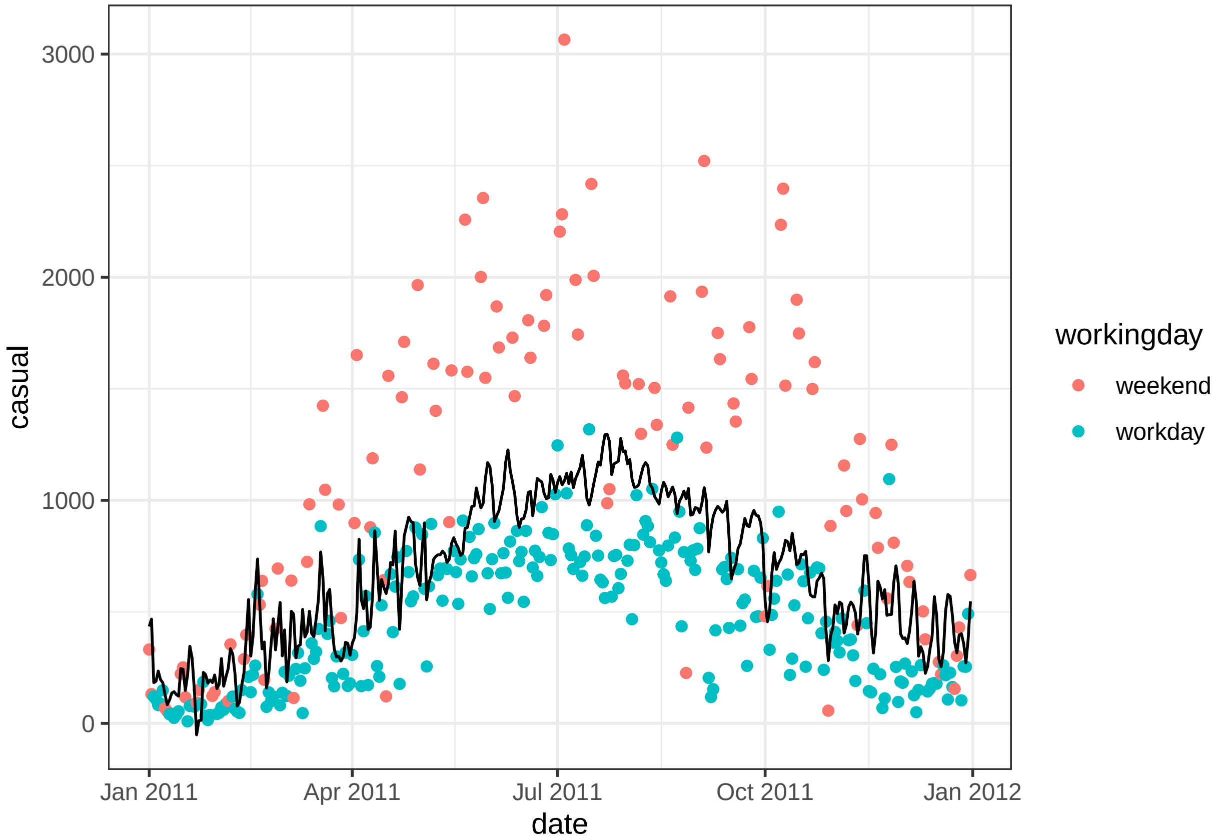
```
## # A tibble: 2 x 2
##   term        estimate
```

```
##   <chr>          <dbl>
## 1 (Intercept)    138.
## 2 temp            36.3
```

Fore every additional degree C, model1 predicts 138 additional riders.

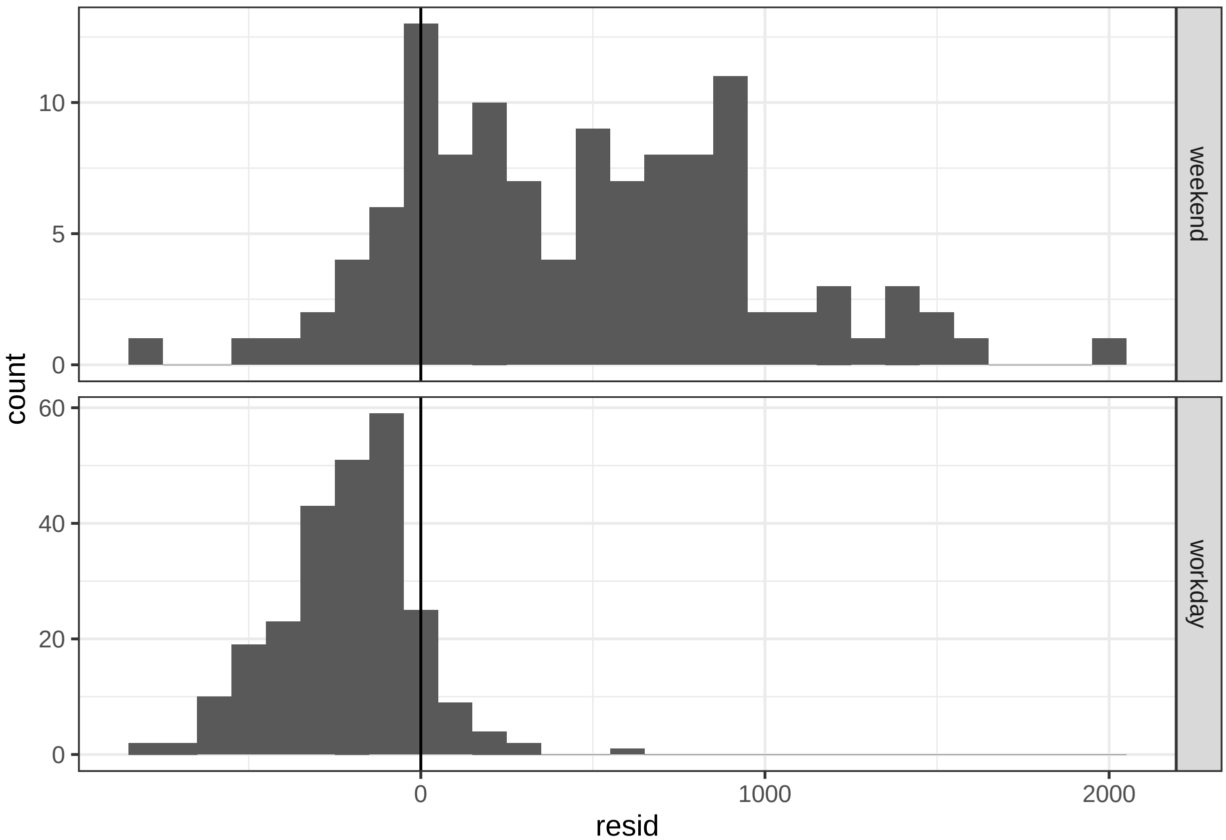
Predictions

```
train %>%
  add_predictions(model1) %>%
  ggplot(aes(x = date)) +
  geom_point(aes(y = casual, color = workingday)) +
  geom_line(aes(y = .pred))
```



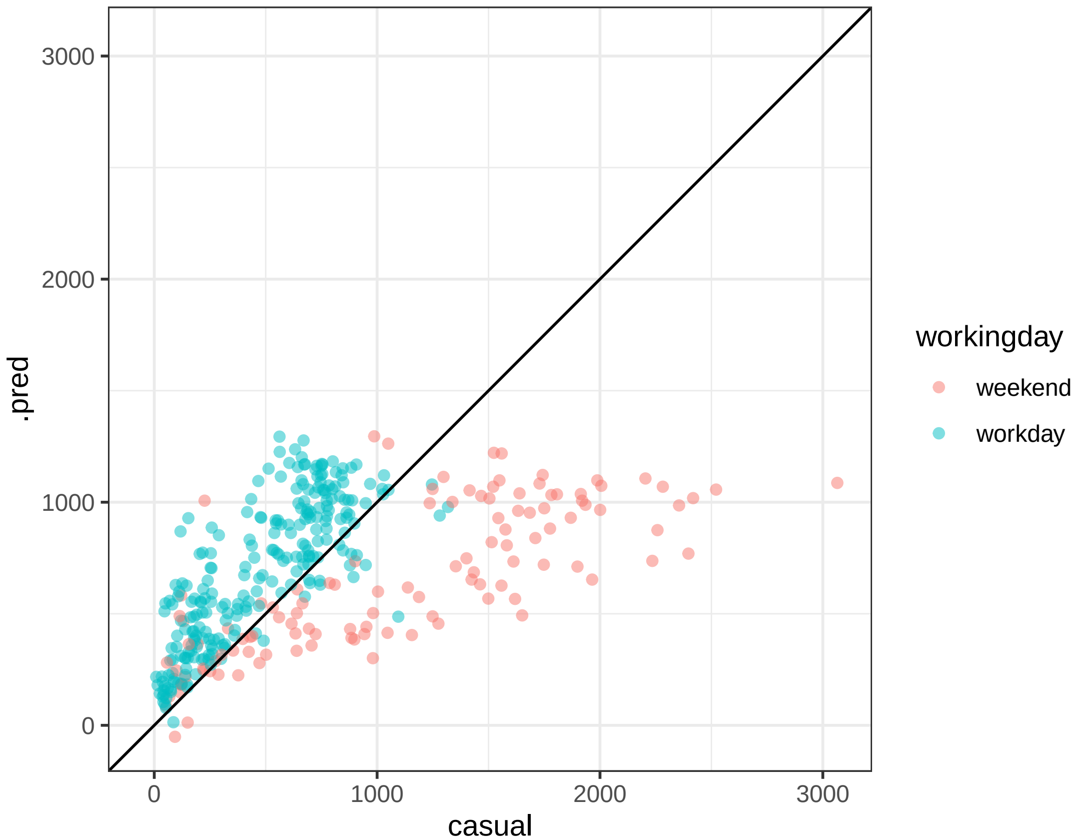
Residuals Histogram

```
train %>%
  add_predictions(model1) %>%
  mutate(resid = casual - .pred) %>%
  ggplot(aes(x = resid)) +
  geom_histogram(binwidth = 100) +
  facet_grid(vars(cols = workingday), scales = "free_y")+
  geom_vline(xintercept = 0)
```



Observed by Predicted

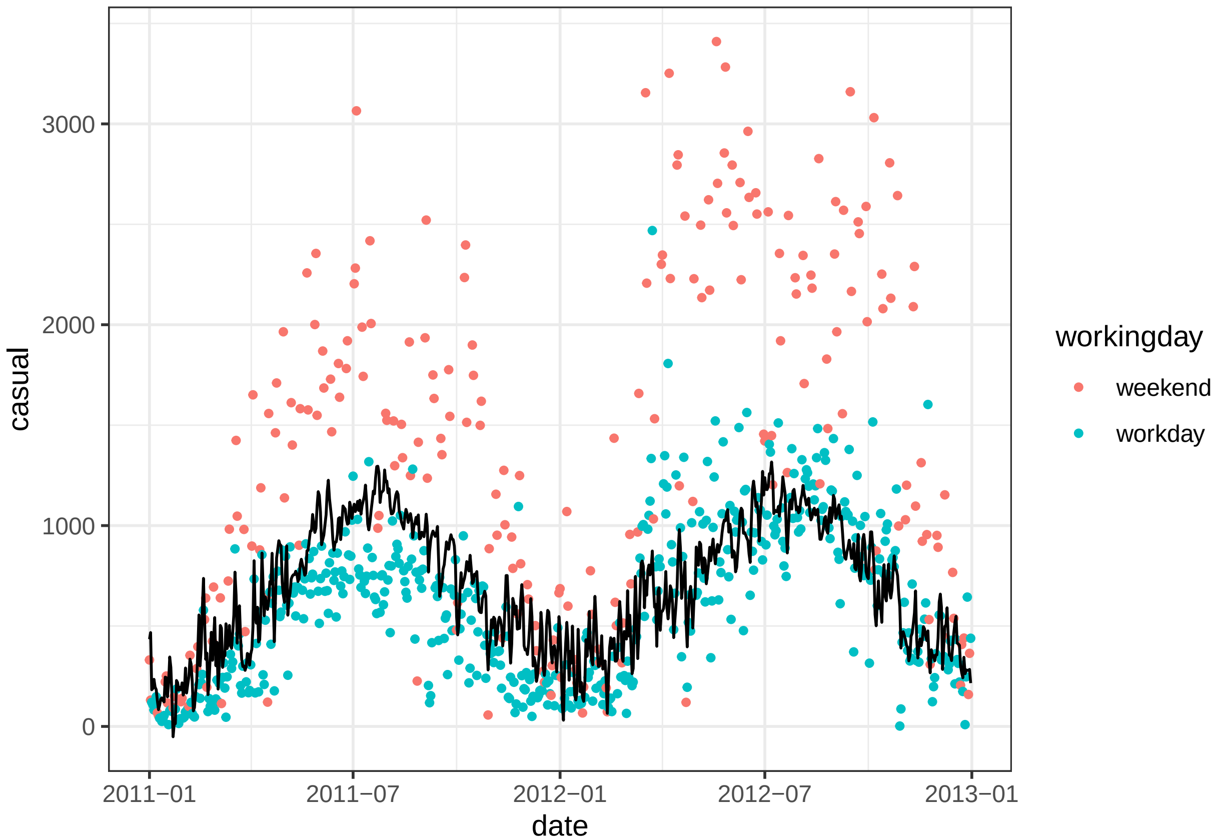
```
train %>%
  add_predictions(model1) %>%
  ggplot(aes(x = casual, y = .pred, color = workingday))+
  geom_point(alpha = 0.5)+
  coord_obs_pred()+
  geom_abline()
```



Under the weekend circumstances, it seems like the model predict the workday too high and the weekend too low. In the causal part, there is high number of rides on the weekend and in the predictions, there is no 2000 < predictions of rides on the weekend.

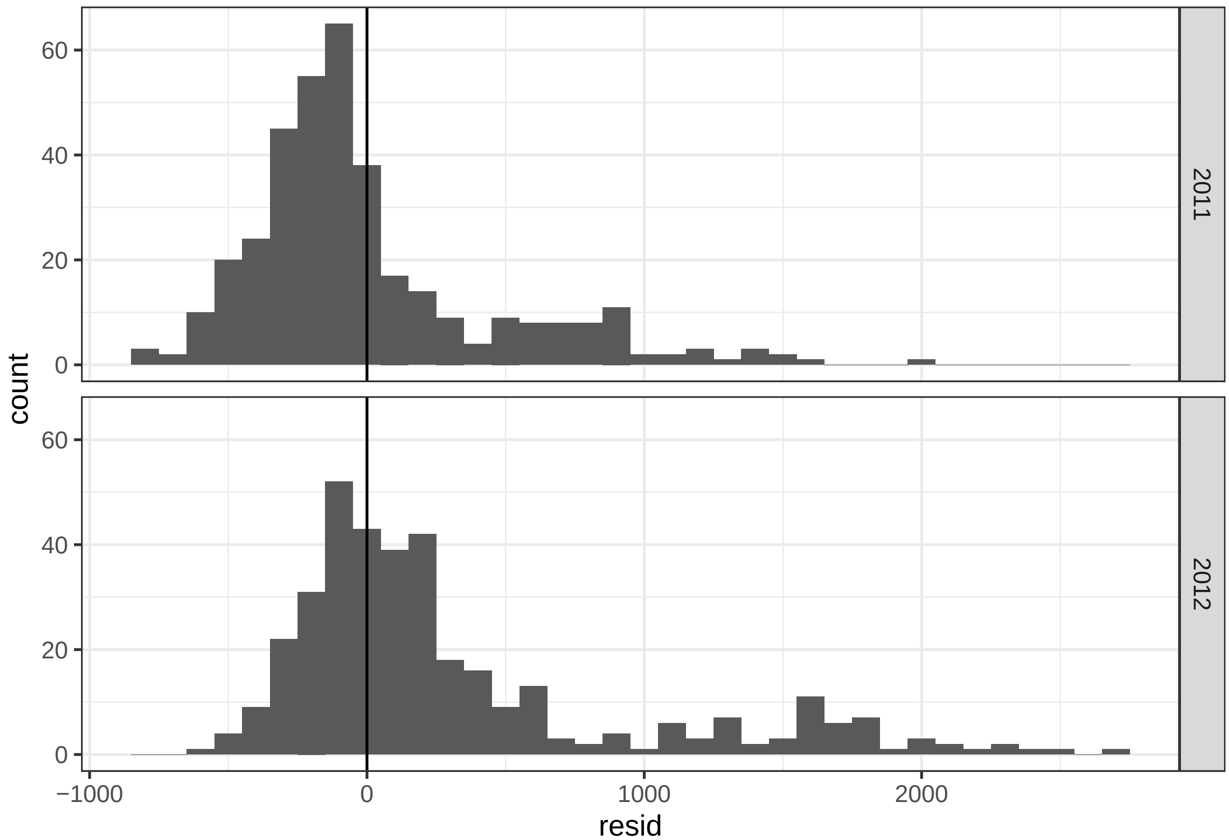
Validate the model on the test set

```
daily_rides %>%
  add_predictions(model1) %>%
  ggplot(aes(x = date)) +
  geom_point(aes(y = casual, color = workingday), size = 1) +
  geom_line(aes(y = .pred))
```



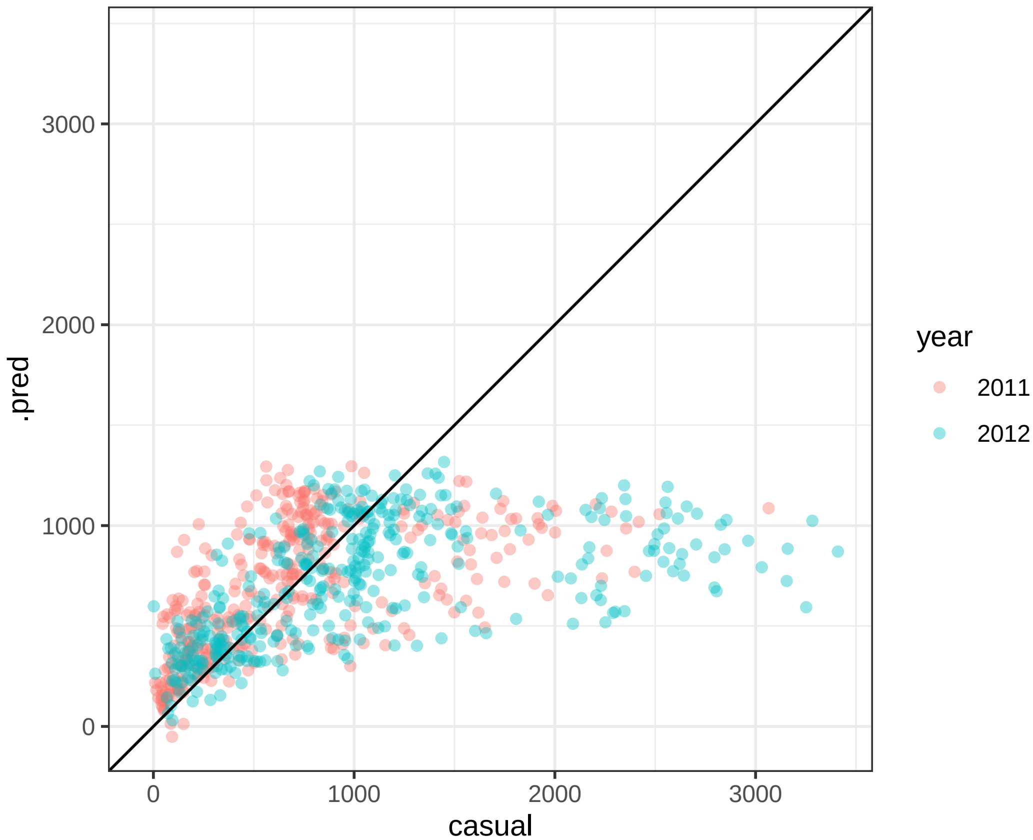
Predictions

```
daily_rides %>%
  add_predictions(model1) %>%
  mutate(resid = casual - .pred) %>%
  ggplot(aes(x = resid)) +
  geom_histogram(binwidth = 100) +
  facet_grid(vars(year))+
  geom_vline(xintercept = 0)+
  theme(legend.position = "top")
```



Residuals

```
daily_rides %>%
  add_predictions(model1) %>%
  ggplot(aes(x = casual, y = .pred, color = year))+
  geom_point(alpha = 0.4)+
  coord_obs_pred()+
  geom_abline()
```



Observed by Predicted

Quantify errors

```
daily_rides %>%
  add_predictions(model1) %>%
  group_by(year) %>%
  mae(truth = casual, estimate = .pred) %>%
  select(year, mae=.estimate)
```

```
## # A tibble: 2 x 2
##   year    mae
##   <fct> <dbl>
## 1 2011   331.
## 2 2012   446.
```

Summarize This model on the training set perform not quite accurate.

When comparing with the test set performance with the training set, the residual number in the 2012 has less difference than the training set.

I can easily observe the predictions and actual number and accuracy easily from the plots but not the table.

I can easily observe the information to make the predictions like temp from the table but not the plots.

Linear Regression using Temperature and Working Day

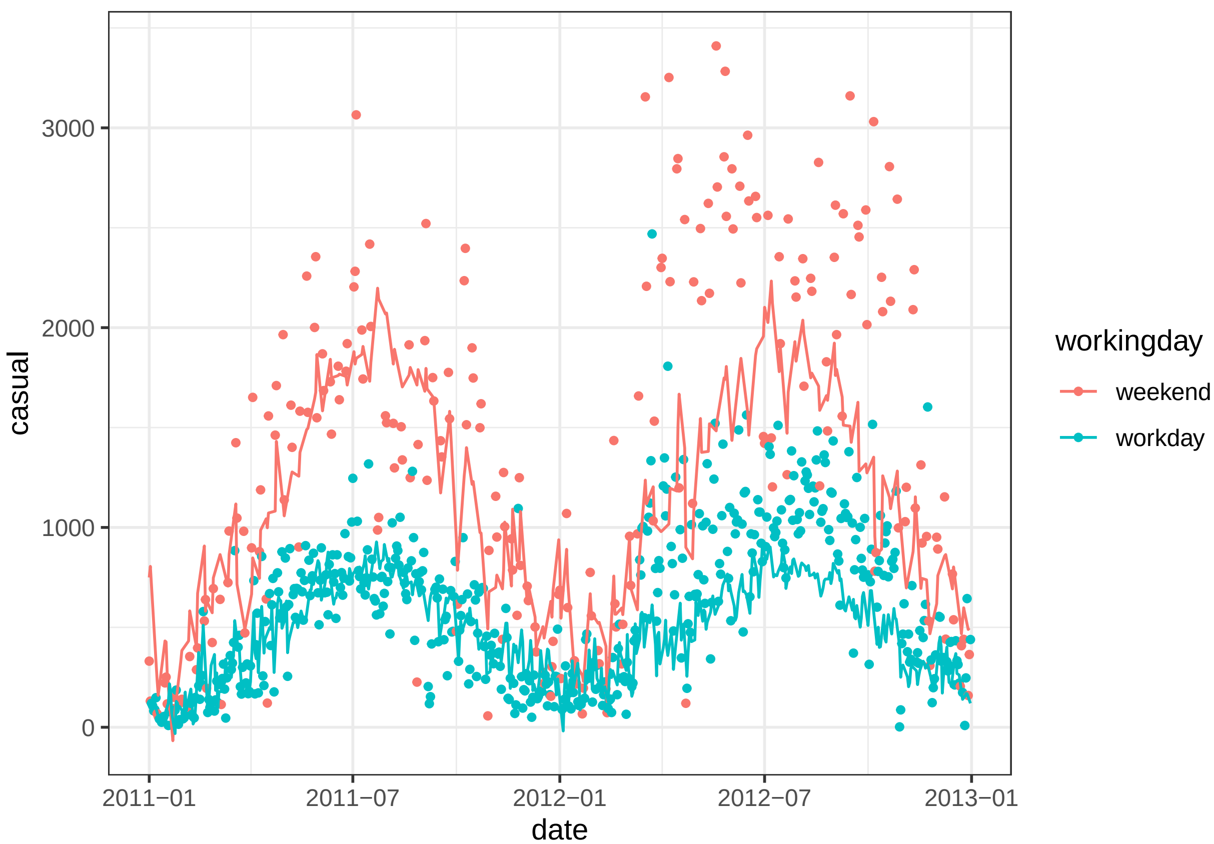
```
recipe2 <-
  recipe(casual ~ temp + workingday, data = train) %>%
  step_dummy(workingday) %>%
  step_interact(~ temp:starts_with("workingday"))

model2 <- workflow() %>%
  add_recipe(recipe2) %>%
  add_model(linear_reg()) %>%
  fit(train)
```

```
model2 %>% tidy() %>% select(term, estimate)
```

```
## # A tibble: 4 x 2
##   term                        estimate
##   <chr>                          <dbl>
## 1 (Intercept)                    251.
## 2 temp                            61.0
## 3 workingday_workday            -190.
## 4 temp_x_workingday_workday      -33.8
```

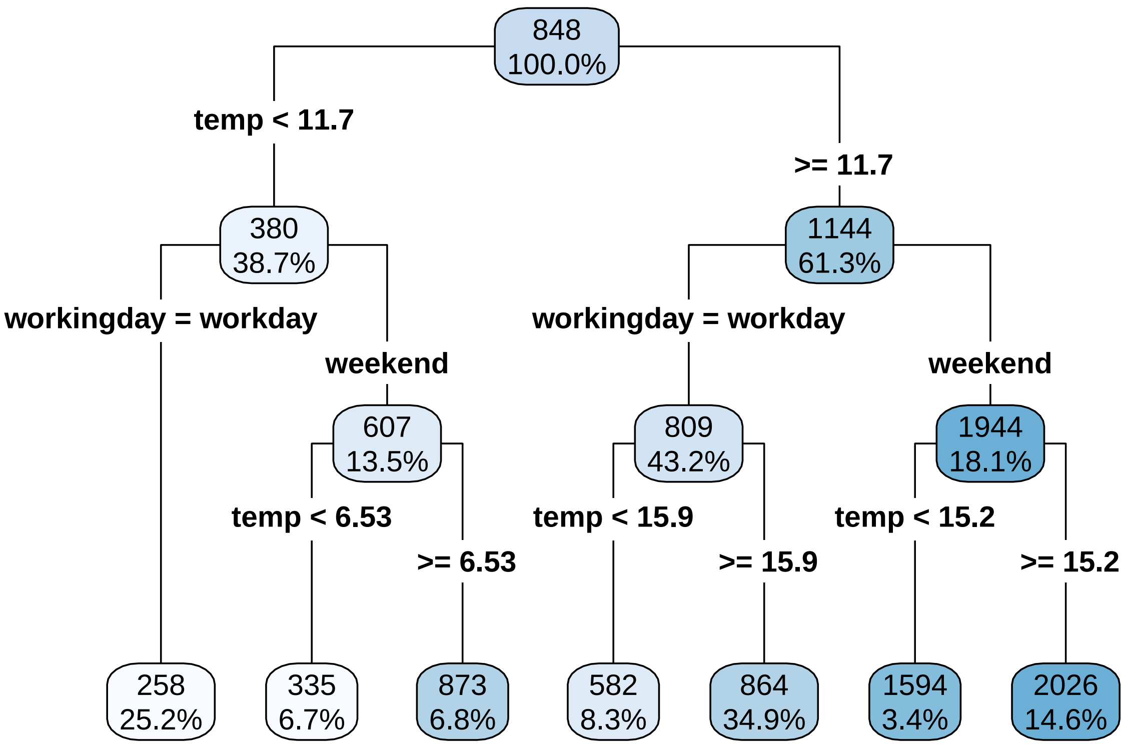
```
daily_rides %>%
  add_predictions(model2) %>%
  ggplot(aes(x = date, y = casual, color = workingday))+
  geom_point(size = 1) +
  geom_line(aes(y = .pred))
```



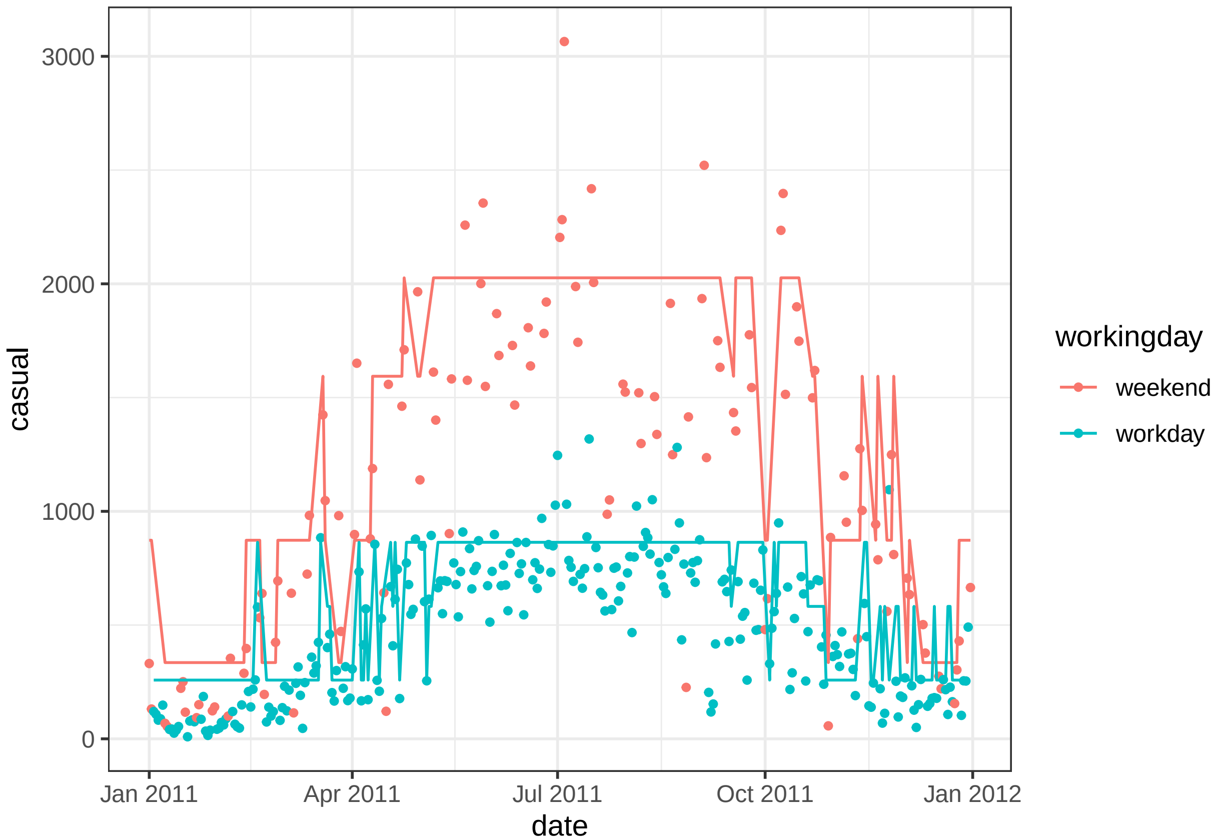
Decision Tree Regression

```
model3 <-
  decision_tree(mode = "regression", tree_depth = 3) %>%
  fit(casual ~ temp + workingday, data = daily_rides)
```

```
model3 %>%
  extract_fit_engine() %>%
  rpart.plot::rpart.plot(roundint = FALSE, digits = 3, type = 4)
```




```
train %>%
  add_predictions(model3) %>%
  ggplot(aes(x = date, y = casual, color = workingday))+
  geom_point(size = 1)+
  geom_line(aes(y = .pred))
```



Wrap-up

With the model 2 and 3, on both year the mean absolute error has slightly different numbers.

These models on 2011 data are more accurate and has a lower mae than the 2012.

Maybe there are other features that affected both of the differences.