HW 8

Marion Geary

3/15/2022

```r
library(tidyverse)
library(tidymodels)
library(mlbench)
data("Ozone")

mycores <- parallel::detectCores(logical = FALSE)
library(doMC)
registerDoMC(cores = mycores)

load("HW_8.Rdata")
```

Exercise 1

```r
ozone <- Ozone %>% rename("month" = V1, "day_of_month" = V2, "day_of_week" = V3, "reading" = V4, "pressu
ozone <- ozone %>% mutate(month = as.numeric(month), day_of_month = as.numeric(day_of_month), day_of_wee
ozone <- ozone %>% filter(!is.na(reading))
set.seed(13)
ozone_split <- initial_split(ozone, prop = .8)
ozone_train <- training(ozone_split)
ozone_test <- testing(ozone_split)

ozone_recipe <- recipe(reading ~ ., data = ozone) %>%
  step_normalize(all_numeric_predictors()) %>%
  step_impute_knn(all_predictors(), neighbors = 5)  %>%
  step_corr(all_numeric_predictors())

ozone_folds <- vfold_cv(ozone_train, v = 10)

elastic_net_mod <- linear_reg(penalty = tune(), mixture = tune()) %>%
    set_engine("glmnet")

elastic_net_wkflow <- workflow() %>%
    add_recipe(ozone_recipe) %>%
    add_model(elastic_net_mod)

penalty_grid <- expand_grid(grid_regular(penalty(), levels = 20), mixture = seq(from = 0, to = 1, by = 0

elastic_net_res <- elastic_net_wkflow %>% tune_grid(resamples = ozone_folds, grid = penalty_grid)
elastic_metrics <- collect_metrics(elastic_net_res)
```

```
my_metrics <- metric_set(rmse, rsq)
ozone_pred <- control_resamples(save_pred = TRUE)

final_workflow <- elastic_net_wkflow %>% finalize_workflow(select_best(elastic_net_res, metric = "rmse")

final_fit <- final_workflow %>% fit(data = ozone_train)

elastic_net_final <- augment(final_fit, new_data = ozone_test)
my_metrics(elastic_net_final, truth = reading, estimate = .pred)
```

```
## # A tibble: 2 x 3
##    .metric .estimator .estimate
##    <chr>   <chr>          <dbl>
## 1 rmse     standard        4.62
## 2 rsq      standard        0.568
```

Exercise 2

```
nn_mod <-
  mlp(hidden_units = tune(), penalty = tune(), epochs = tune()) %>%
  set_engine("nnet") %>%
  set_mode("regression")

nn_wkflow <- workflow() %>%
  add_recipe(ozone_recipe) %>%
  add_model(nn_mod)

nn_tuning_grid <- crossing(
  hidden_units = 1:5,
  penalty = seq(0.001, 0.1, length = 6),
  epochs = c(1000)
)

nn_tuned <- nn_wkflow %>%
  tune_grid(resamples = ozone_folds, grid = nn_tuning_grid)
```
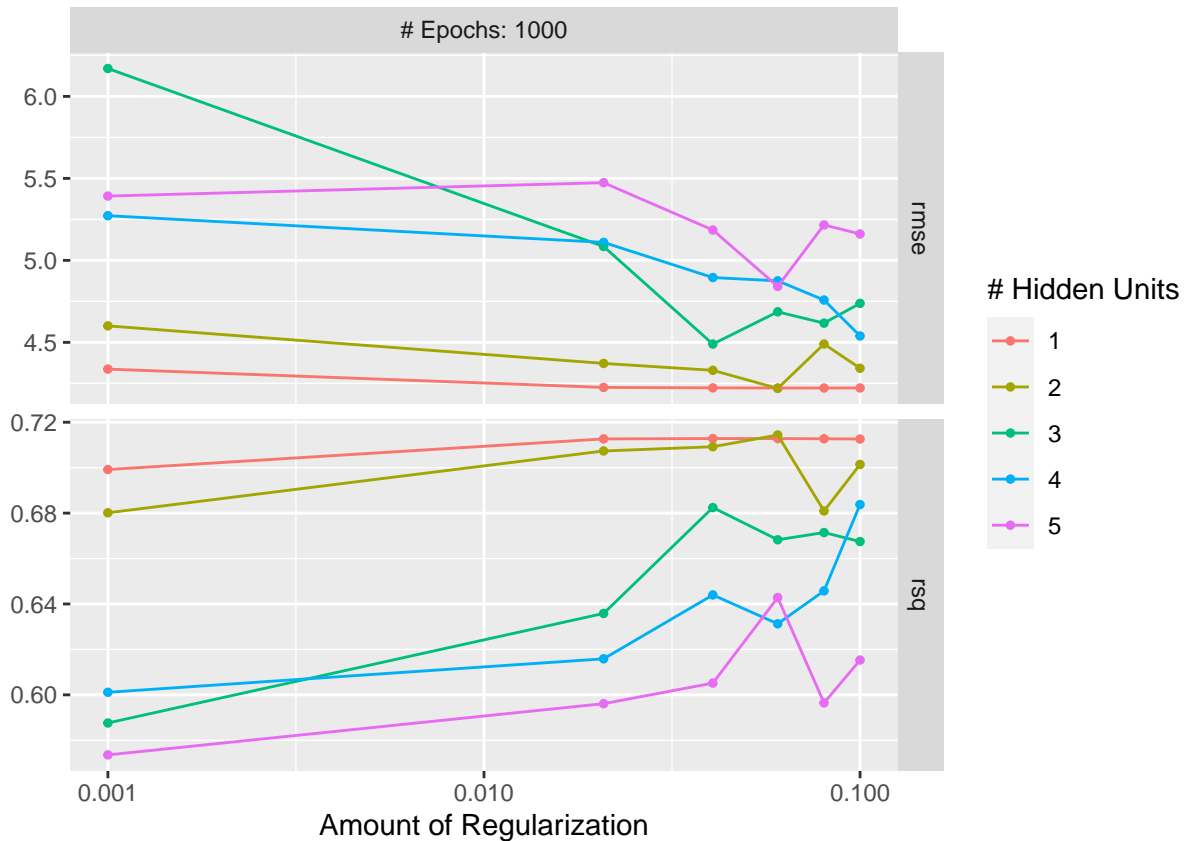
```
autoplot(nn_tuned)
```

From tuning our neural net, we see that 2 hidden nodes is the best for `RMSE`. In fact, this also results in the best `R^2`. As regularization increases, the metrics generally improve for all hidden nodes. The best results seem to appear with slightly less regularization than 0.100. Overall, it seems that less hidden nodes is better for this data, with 1 or 2 hidden nodes having the best metrics for all amounts of regularization. Having 2 hidden nodes is only slightly better than 1.

Exercise 3

```
lowest_rmse <- nn_tuned %>%
  select_best("rmse")

nn_final_fit <- finalize_workflow(
  nn_wkflow, lowest_rmse
) %>%
  fit(data = ozone_train)

nn_final_test <- augment(nn_final_fit, new_data = ozone_test)
my_metrics(nn_final_test, truth = reading, estimate = .pred)
```

```
my_metrics(nn_final_test, truth = reading, estimate = .pred)
```

```
## # A tibble: 2 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
```

```
## 1 rmse    standard      4.56
## 2 rsq     standard      0.567
```

In terms of results, the neural net results in a slightly lower `RMSE` of 4.56 instead of 4.62. The neural net also has a slightly lower `R^2` than that of the elastic net, 0.567 vs 0.568. One reason that the `RMSE` values are so similar is that elastic net is very similar to a neural net when the neural net has few hidden nodes. Using only 2 hidden nodes with the neural net is only slightly different from the elastic net, having done changed the linear regression only slightly. The neural net has some non-linear aspects because of the activation functions while the elastic net does not, but the similar metrics tell us that the relationship between the data is pretty linear. Both models have regularization through the use of penalties. While the neural net results in slightly better metrics, it is not a significant improvement over the elastic net model.