fishing

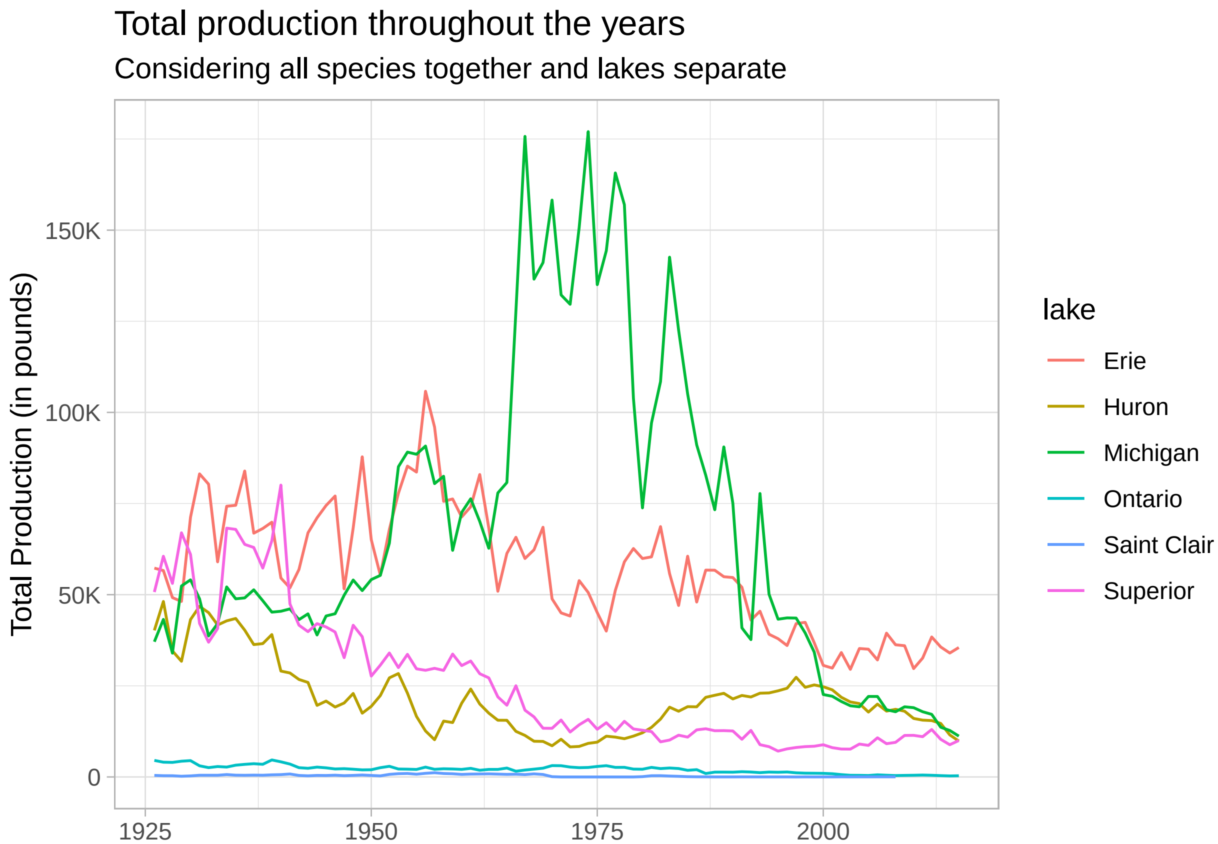
```
library(tidyverse)
library(tidymodels)
library(gdata)
library(skimr)

theme_set(theme_light())

set.seed(123)
```

First, let's load the data

```
fishing <- readr::read_csv(paste0("https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/

fishing <- fishing %>% filter(year > 1925) # Some data before 1955 have weird behavior
```

Exploratory data analysis

```
fishing %>%
  group_by(year, lake) %>%
  summarise(year_production = sum(values, na.rm = T)) %>%
  ggplot(aes(year, year_production)) +
  geom_line(aes(color = lake)) +
  scale_y_continuous(labels = label_number_si()) +
  labs(x = NULL,
       y = "Total Production (in pounds)",
       title = "Total production throughout the years",
       subtitle = "Considering all species together and lakes separate")
```



It is not feasible to analyze the behaviour of each species on each lake.

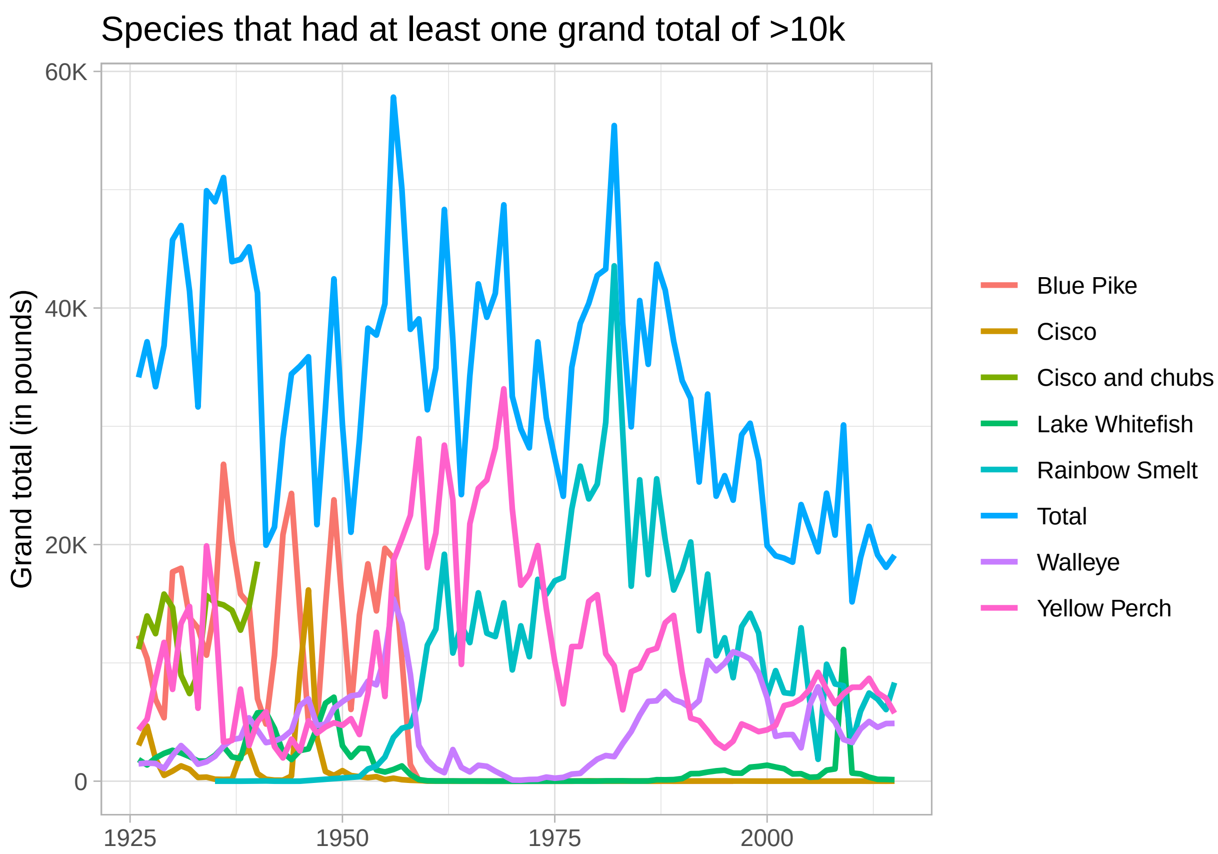
```
fishing %>%
  group_by(year, lake, species) %>%
  summarise(year_production = sum(values, na.rm = T)) %>%
  nrow()
```

```
## `summarise()` has grouped output by 'year', 'lake'. You can override using the `.groups` argument.

## [1] 8682
```

```
grand_total <- fishing %>%
  group_by(species, year) %>%
  slice_head(n = 1) %>%
  select(year, species, grand_total) %>%
  drop_na() %>%
  ungroup() %>%
  group_by(species) %>%
  mutate(species_max = max(grand_total)) %>%
         #species = ifelse(species_max <= 10000, "Other", species)) %>% View()
  filter(species_max > 10000) %>%
  select(year, species, grand_total)

year_total <- grand_total %>%
  ungroup() %>%
  group_by(year) %>%
  summarise(grand_total = sum(grand_total)) %>%
```

```
  mutate(species = "Total")

grand_total <- bind_rows(grand_total, year_total)
```

```
grand_total %>%
  ggplot(aes(year, grand_total)) +
  geom_line(aes(color = species), size = 1) +
  scale_y_continuous(labels = label_number_si()) +
  labs(x = NULL,
       y = "Grand total (in pounds)",
       color = NULL,
       title = "Species that had at least one grand total of >10k")
```



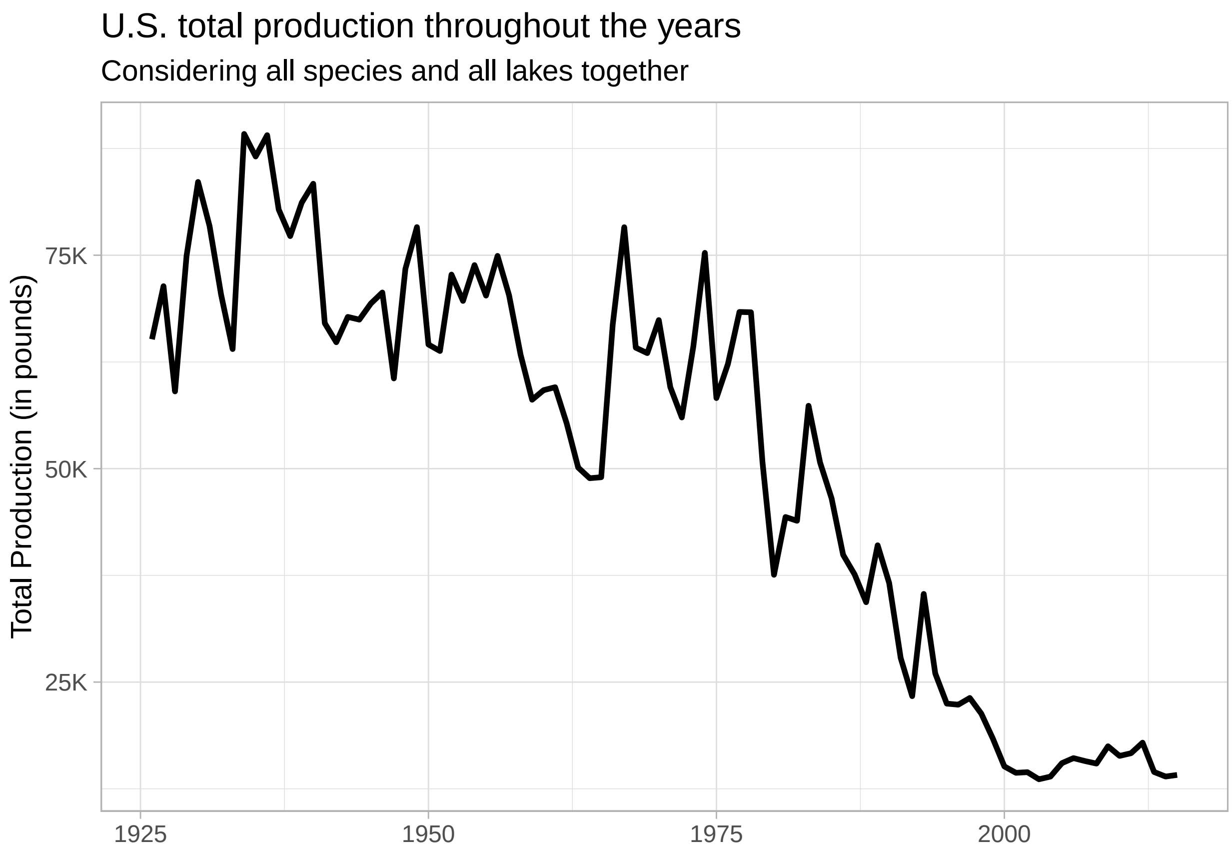
Modeling

Let's try to predict the U.S. total production based on the production of Ohio only.

The next two plots represent the data used on the model.

```
us_total_production <- fishing %>%
  filter(region == "U.S. Total") %>%
  group_by(year) %>%
  summarise(us_total_production = sum(values, na.rm = T))

us_total_production %>%
```

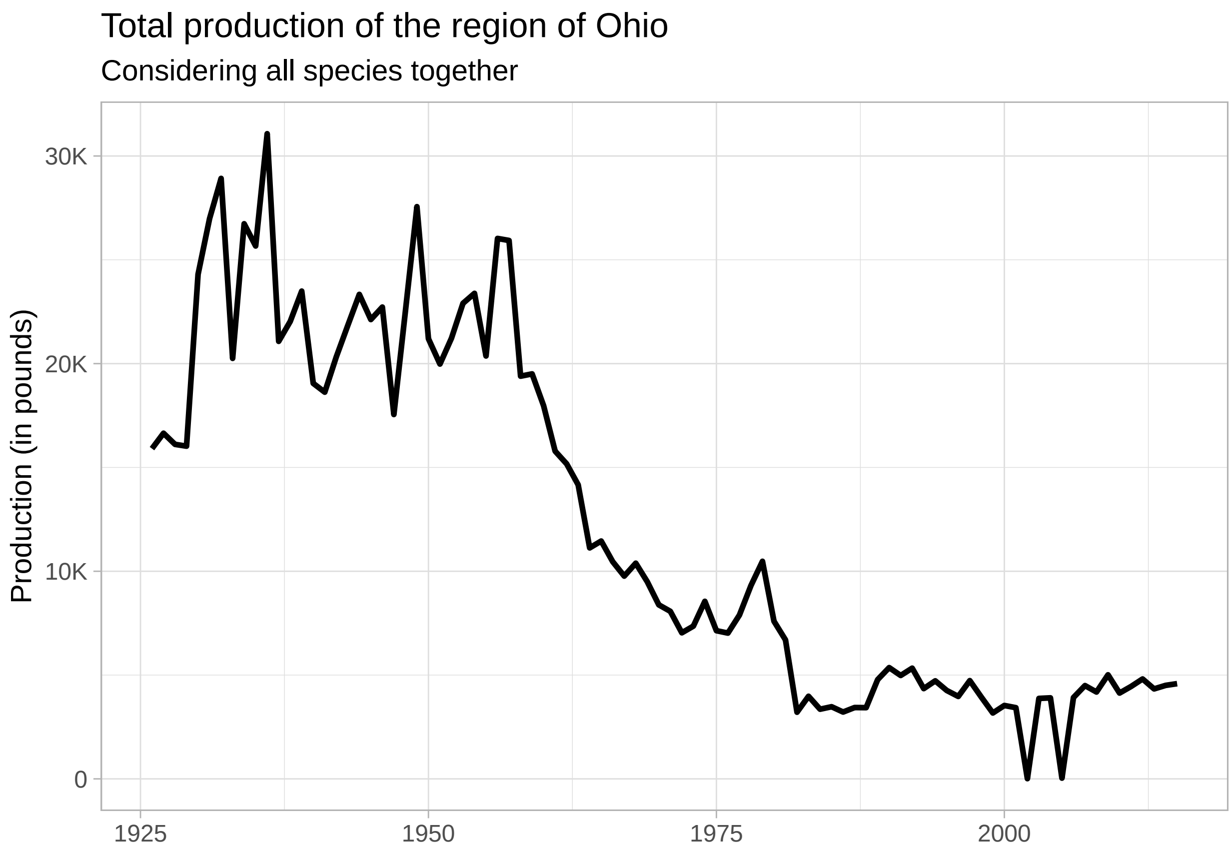
```r
  ggplot(aes(year, us_total_production)) +
  geom_line(size = 1) +
  scale_y_continuous(labels = label_number_si()) +
  labs(x = NULL,
       y = "Total Production (in pounds)",
       title = "U.S. total production throughout the years",
       subtitle = "Considering all species and all lakes together")
```




```r
region_production <- fishing %>%
  group_by(year, region) %>%
  summarise(region_production = sum(values, na.rm = T)) %>%
  ungroup() %>%
  group_by(region) %>%
  mutate(region_max_production = max(region_production),
         region_min_production = min(region_production)) %>%
  # This prevents to keep data from regions that did not start fishing activities
  # by the year of 1925.
  filter(region_max_production > 10000, region_min_production > 0) %>%
  select(-region_min_production, -region_max_production) %>%
  pivot_wider(names_from = region, values_from = region_production)
# These are the regions that present at least one production of >10k,
#as shown on a previous plot.
```

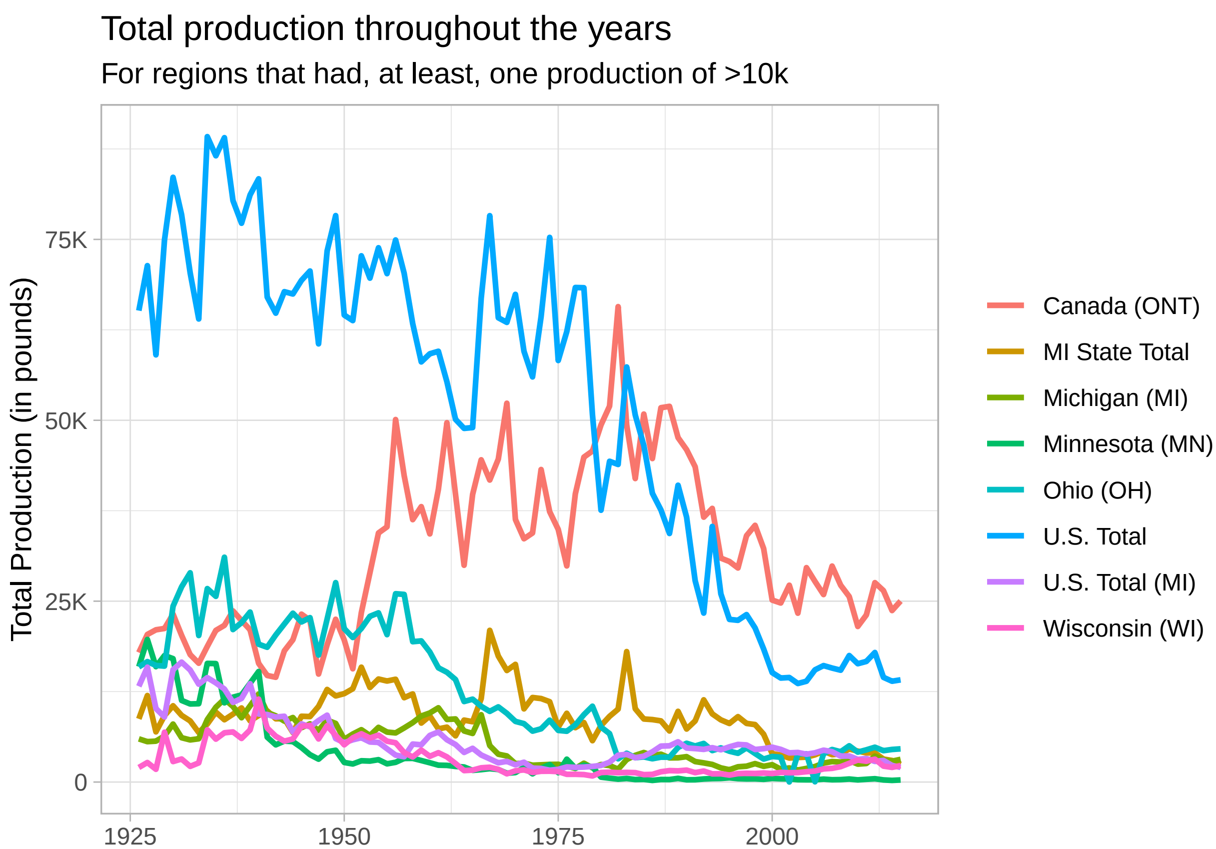
```r
region_production %>%
  ggplot(aes(year, `Ohio (OH)`)) +
  geom_line(size = 1) +
```

```
  scale_y_continuous(labels = label_number_si()) +
  labs(x = NULL, y = "Production (in pounds)",
       title = "Total production of the region of Ohio",
       subtitle = "Considering all species together")
```



We can also see how other regions total production are distributed.

```
region_production %>%
  pivot_longer(cols = !starts_with("year"),
               names_to = "region", values_to = "year_production") %>%
  ggplot(aes(year, year_production)) +
  geom_line(aes(color = region), size = 1) +
  scale_y_continuous(labels = label_number_si()) +
  labs(x = NULL,
       y = "Total Production (in pounds)",
       title = "Total production throughout the years",
       subtitle = "For regions that had, at least, one production of >10k",
       color = NULL)
```



By now, let's try to predict the total U.S. production by using Ohio production as a predictor. At first, let's split the data on training and testing sets.

```
data <- initial_split(region_production)

train_production <- training(data)
test_production <- testing(data)
```

It is possible to fit the model and make the predictions right away.

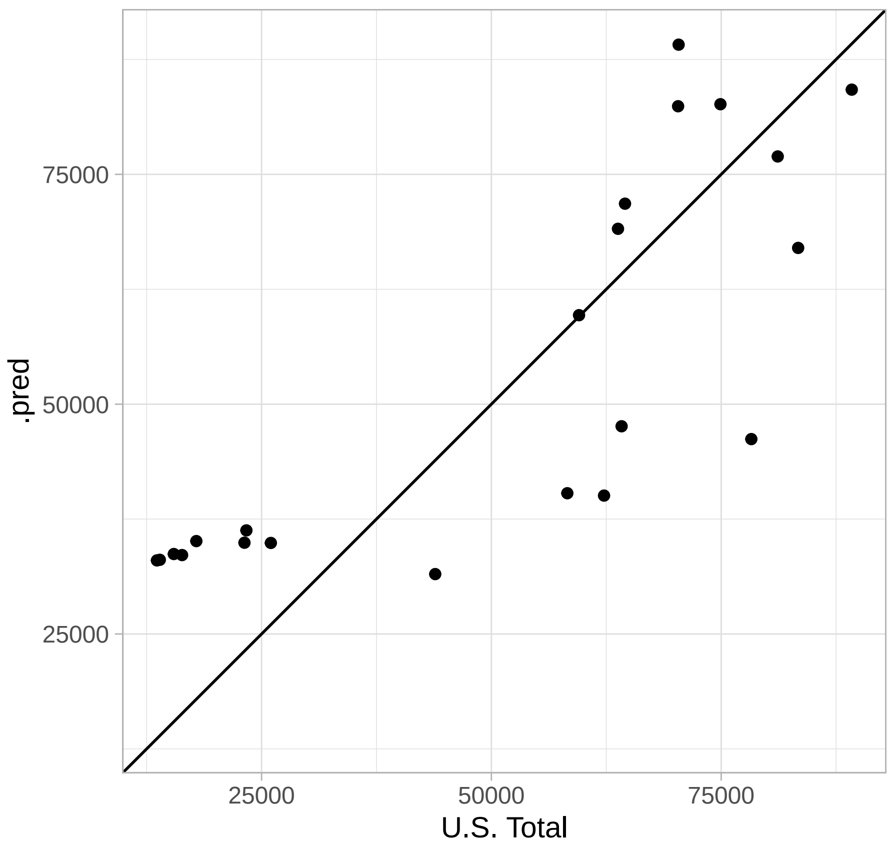
```
lm_model <- linear_reg() %>% set_engine("lm")

lm_workflow <-
  workflow() %>%
  add_model(lm_model) %>%
  add_formula(`U.S. Total` ~ `Ohio (OH)`)


model_fit <-
  fit(lm_workflow, data = train_production)


prediction <- predict(model_fit, new_data = test_production)
```

We can apply some metrics to judge model effectiveness.

```
#Let's bind the real values and the predictions make on the test set.

prediction <- bind_cols(test_production, prediction) %>%
  ungroup() %>%
  select(-year)

pred_metrics <- metric_set(rmse, mae)

prediction %>%
  ungroup() %>%
  pred_metrics(truth = `U.S. Total`, estimate = .pred)
```

```
## # A tibble: 2 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard      15482.
## 2 mae     standard      13764.
```

```
prediction %>%
  ggplot(aes(`U.S. Total`, .pred)) +
  geom_abline() +
  geom_point() +
  coord_obs_pred()
```



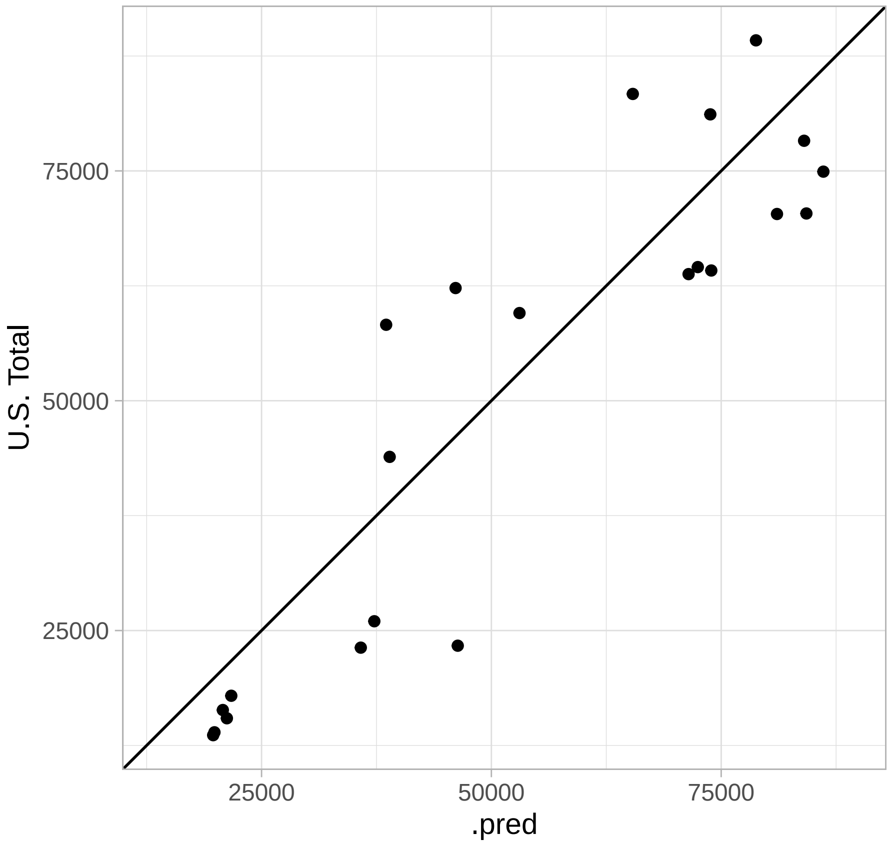
This model produced a mean absolute error of 13k. Let's try adding more regions to the prediction and fit the model once again.

```
lm_workflow <- lm_workflow %>%
  update_formula(`U.S. Total` ~ `Ohio (OH)` + `Minnesota (MN)` + `Wisconsin (WI)` +
        `Michigan (MI)` + `MI State Total`)


model_fit <- fit(lm_workflow, data = train_production)


prediction <- predict(model_fit, new_data = test_production)
```

```
#Let's bind the real values and the predictions make on the test set.

prediction <- bind_cols(test_production, prediction)

prediction %>%
  ungroup() %>%
  pred_metrics(truth = `U.S. Total`, estimate = .pred)
```

```
## # A tibble: 2 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard      11365.
## 2 mae     standard      10134.
```

```
prediction %>%
  ggplot(aes(.pred, `U.S. Total`)) +
  geom_abline() +
  geom_point() +
  coord_obs_pred()
```



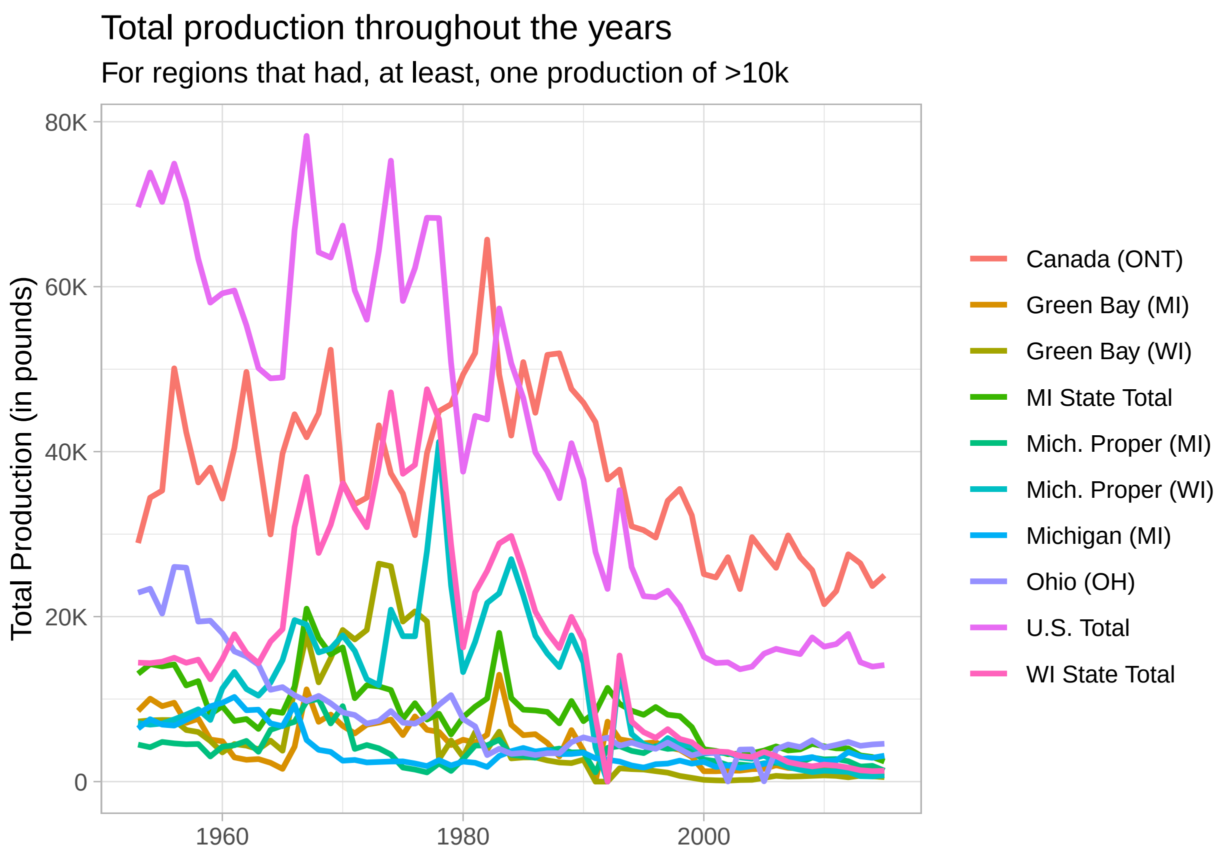
Some regions fit the requirement of having least one production of >10k, but present data starting only at 1953. Let's try using them on our model to see if they have a good impact.

```
region_production <- fishing %>%
  filter(year >= 1953) %>%
  group_by(year, region) %>%
  summarise(region_production = sum(values, na.rm = T)) %>%
  ungroup() %>%
  group_by(region) %>%
  mutate(region_max_production = max(region_production)) %>%
  filter(region_max_production > 10000) %>%
  select(-region_max_production) %>%
  pivot_wider(names_from = region, values_from = region_production)
```

First, we can make a plot to have a general ideia of their behavior.

```
region_production %>%
  pivot_longer(cols = !starts_with("year"),
               names_to = "region", values_to = "year_production") %>%
  ggplot(aes(year, year_production)) +
  geom_line(aes(color = region), size = 1) +
  scale_y_continuous(labels = label_number_si()) +
  labs(x = NULL,
       y = "Total Production (in pounds)",
       title = "Total production throughout the years",
```

```
      subtitle = "For regions that had, at least, one production of >10k",
      color = NULL)
```



Since the data were reduced, let's do the initial split once again.

```
data <- initial_split(region_production)

train_production <- training(data)
test_production <- testing(data)
```

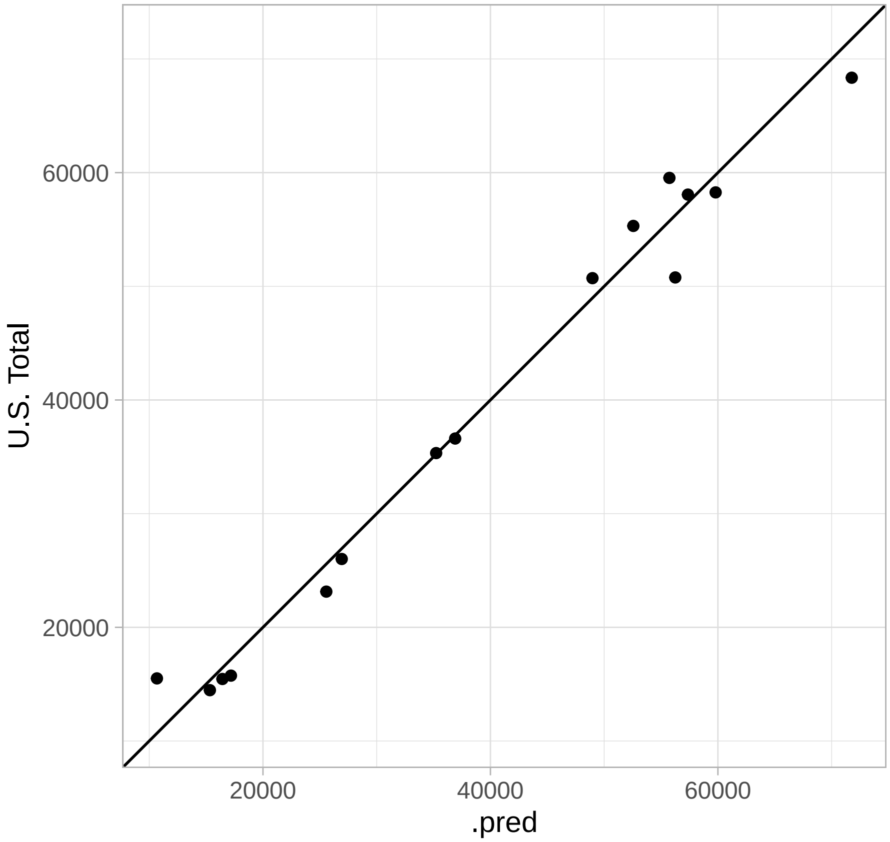
```
lm_workflow <- lm_workflow %>%
  update_formula(`U.S. Total` ~ `Ohio (OH)` + `Mich. Proper (MI)` +
                 `Mich. Proper (WI)` + `Green Bay (MI)` + `Green Bay (WI)`)

model_fit <-
  fit(lm_workflow, data = train_production)

prediction <- predict(model_fit, new_data = test_production)
```

```
prediction <- bind_cols(test_production, prediction)

prediction %>%
  ungroup() %>%
  pred_metrics(truth = `U.S. Total`, estimate = .pred)
```

```
## # A tibble: 2 x 3
```

```
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard       2626.
## 2 mae     standard       2085.
```

```
prediction %>%
  ggplot(aes(.pred, `U.S. Total`)) +
  geom_abline() +
  geom_point() +
  coord_obs_pred()
```



That's it! A mean squared error of 2k. A much better result than the ones obtained by previous models.