Linear Regression on PA Population Data

Alexander Vaillant

9/7/2021

Environment Setup

Import Necessary Libraries

```
#Load all necessary packages
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.1
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1
```

```
## Warning: package 'tibble' was built under R version 4.1.1
```

```
## Warning: package 'tidyr' was built under R version 4.1.1
```

```
## Warning: package 'readr' was built under R version 4.1.1
```

```
## Warning: package 'purrr' was built under R version 4.1.1
```

```
## Warning: package 'forcats' was built under R version 4.1.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.1.1
```

```
library(dplyr)
library(tidyr)
library(modelr)
```

```
## Warning: package 'modelr' was built under R version 4.1.1
```

```
library(ggplot2)
```

Data Gathering

Load Dataset into Dataframe using read_excel()

```
# Import the raw dataset using readxl::read_excel
url <- "C:/Users/tedda/Desktop/Data Science Portfolio/Machine Learning/Supervised Learning/Regression/Li
raw_data <- read_excel(url)
```

Data Preparation

```
# Cleanse the data for Pennsylvania 2010 - 2019 using dplyr::select & dplyr::filter
data <- raw_data %>% select(NAME, "2010":"2019")
data <- filter(data, NAME == "Pennsylvania")

# Remove the NAME column using dplyr::select
data2 <- data %>% select("2010":"2019")

# Create a dataframe from data2 using data.frame
# t(data2) is used to transpose the data from horizontal to vertical
# 2010:2019 creates a column of Years
df2<-data.frame(x=2010:2019, y=t(data2))
rownames(df2) <- NULL #This resets the rownames

# Export cleansed dataset
write.csv(df2, "C:/Users/tedda/Desktop/Data Science Portfolio/Machine Learning/Supervised Learning/Regr
```

Model Building

```
# Create your linear regression model using modelr::lm()
df2_mod<- lm(y ~ x, data = df2)
coef(df2_mod)
```

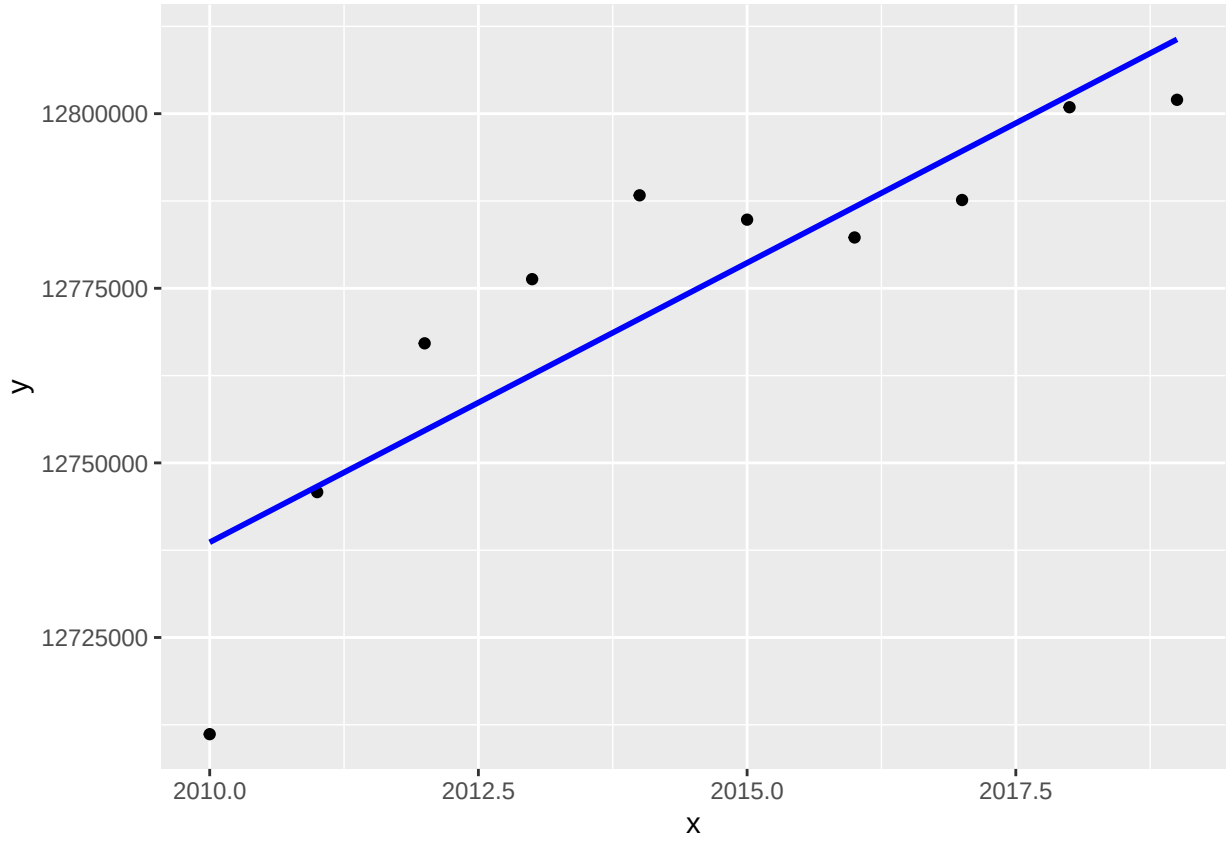
```
## (Intercept)           x
## -3344244.55     8001.43
```

Save and Load Model

```
# Save and Load Model
model_url <- "C:/Users/tedda/Desktop/Data Science Portfolio/Machine Learning/Supervised Learning/Regress
saveRDS(df2_mod, model_url)
df2_mod <- readRDS(model_url)
```

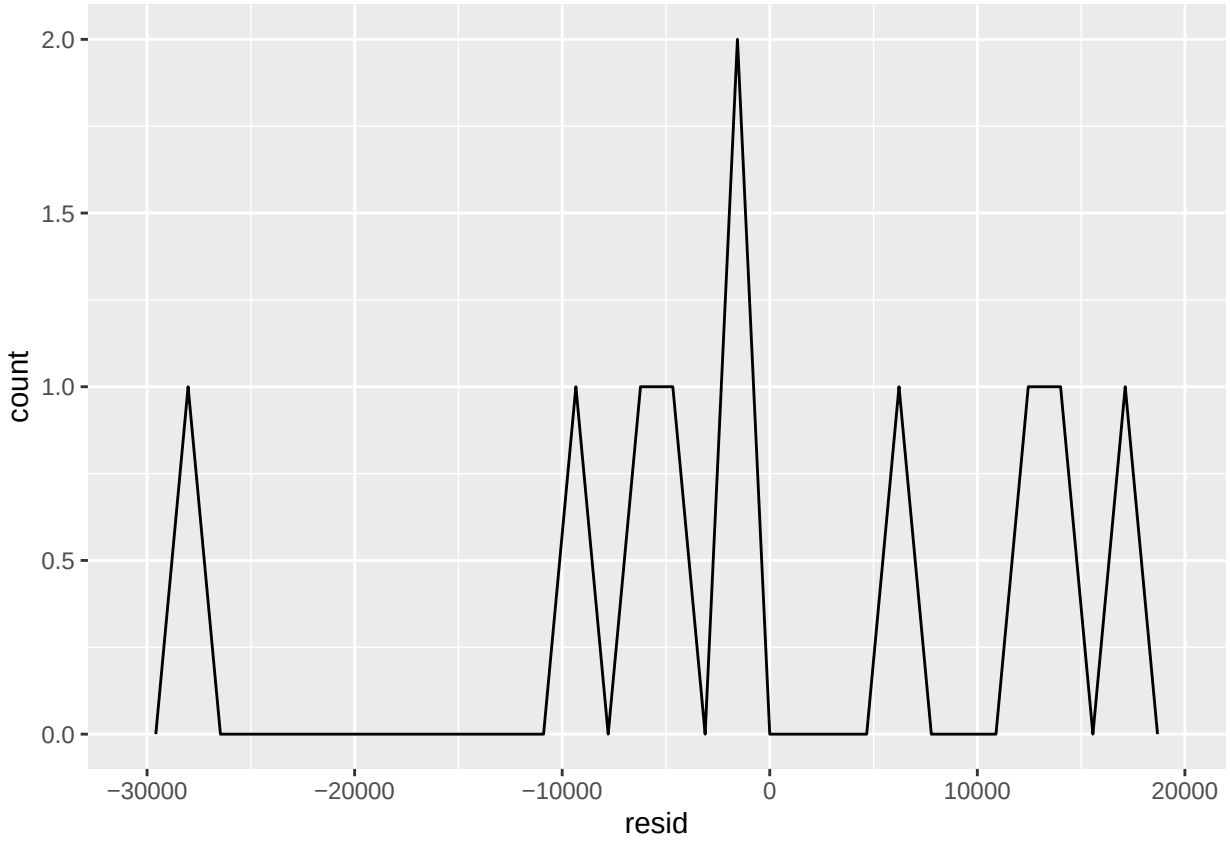
Model Evaluation & Prediction

```
# Create a grid to add your predictions for plotting using add_predictions()
grid <- df2 %>% data_grid(x)
grid <- grid %>% add_predictions(df2_mod)

# Plot the grid using ggplot2::ggplot()
ggplot(df2, aes(x)) +
  geom_point(aes(y = y)) +
  geom_line(aes(y = pred), data = grid, colour = "blue", size = 1)
```




```
# Add the residuals to your dataframe using add_residuals()
df2 <- df2 %>% add_residuals(df2_mod)

# Plot the residuals to view the frequency using geom_freqpoly()
ggplot(df2, aes(resid)) +
  geom_freqpoly()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# Create a summary of your data using summary()
summary(df2)
```

```
##        x               y                resid
##  Min.   :2010   Min.   :12711160   Min.   :-27470
##  1st Qu.:2012   1st Qu.:12769416   1st Qu.: -6341
##  Median :2014   Median :12783550   Median : -1268
##  Mean   :2014   Mean   :12774637   Mean   :     0
##  3rd Qu.:2017   3rd Qu.:12788145   3rd Qu.: 10911
##  Max.   :2019   Max.   :12801989   Max.   : 17677
```

```
# Predict the population for the next five years using predict()
five_year_pred <- data.frame (x=2020:2025)
rownames(five_year_pred) <- c(2020:2025)
predict(df2_mod,newdata = five_year_pred)
```

```
##     2020     2021     2022     2023     2024     2025
## 12818645 12826646 12834648 12842649 12850650 12858652
```