% of people in x category

-0.16

% of people in y category

1.2

<https://tidyr.tidyverse.org/reference/gather.html>

data= read.csv("C:\\00\\covid\_cases.csv")

#Gather (in Excel similar to Pivot)

library(tidyverse)

ncols = ncol(data)

data1 = gather(data, "year", "cases", 3:ncols)

|  |  |
| --- | --- |
| Group | Done |
| 1 | Done! |
| 2 | Done |
| 3 | Done |
| 4 | Done |
| 5 | Done |
| 6 | Done |
| 7 | Done |

#mutate

#%>% (pipe operator -> works like a . in java or c#)

#data.gather...

data1 = data %>% gather("year", "cases", 3:ncols)

data2 = data1 %>% mutate(year = str\_replace(year, "X", ""))

data3 = data2 %>% mutate(year = as.Date(year, "%m.%d.%Y"))

#data.gather...

data1 = data %>% gather("year", "cases", 3:ncols)

data2 = data1 %>% mutate(year = str\_replace(year, "X", ""))

data3 = data2 %>% mutate(year = as.Date(year, "%m.%d.%Y"))

data4 = data %>% gather("year", "cases", 3:ncols) %>%

mutate(year = year %>% str\_replace("X", "")) %>%

mutate(year = year %>% as.Date("%m.%d.%Y"))

#SQL Where

wa\_data = data3 %>% filter(State == "WA")

king\_data = wa\_data %>% filter(ï..CountyName == "King County")

#SQL Select

king\_data1 = king\_data %>% select(-State)

king\_data1 = king\_data %>% select(ï..CountyName, year, cases)

king\_data2 = king\_data1 %>% mutate(CountyName = ï..CountyName) %>%

mutate(Date = year) %>%

mutate(Cases = cases) %>%

select(-year, -cases, -ï..CountyName)

#mutate creates columns (potentially overwriting an existing column)

#select picks or removes columns

#filter filters rows by a condition

**Problem**

1- Take covid data

2- Gather it

3- Keep only Florida

4- Keep cases from the first available date till June 1st 2020  
5- Plot the trend of cases over time

|  |  |
| --- | --- |
| Group | Chart |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |

florida\_data = data %>% gather("year", "cases", 3:c) %>% mutate(

year = as.Date(str\_replace(year, "X", ""), "%m.%d.%y"))

florida\_data = florida\_data %>% filter(State == "FL") %>%

filter(year <= "2020-6-1")

plot(florida\_data$cases)

#group\_by / summarize (groups and aggregates)

#any summarization functions, special one n() = count

florida\_total = florida\_data %>%

group\_by(State, year) %>%

summarise(Total = sum(cases))

plot(florida\_total$Total)

x = 1:nrow(florida\_total)

totals\_data = data.frame(x = x, y = florida\_total$Total)

totals\_data = totals\_data %>% filter(x > 65)

plot(totals\_data$x, totals\_data$y)

model = lm(y ~ x, totals\_data)

summary(model)

plot(model$residuals)

florida\_total

newdate = data.frame(x = 133:150)

p = predict(model, newdata = newdate)

**Break - Back 7:45PM**

c = ncol(data)

florida\_data = data %>% gather("year", "cases", 3:c) %>% mutate(

year = as.Date(str\_replace(year, "X", ""), "%m.%d.%y"))

florida\_data = florida\_data %>% filter(State == "FL") %>%

filter(year <= "2020-6-1" )

plot(florida\_data$cases)

#group\_by / summarize (groups and aggregates)

#any summarization functions, special one n() = count

florida\_total = florida\_data %>%

group\_by(State, year) %>%

summarise(Total = sum(cases))

plot(florida\_total$Total)

florida\_total$x = 1:nrow(florida\_total)

florida\_total = florida\_total %>% filter(x > 65)

plot(florida\_total$x, florida\_total$Total)

model = lm(Total ~ x, florida\_total)

summary(model)

plot(model$residuals)

florida\_total$original = 1

x\_predict = data.frame(x = 133:150)

p = predict(model, newdata = x\_predict)

predicted = data.frame(x = x\_predict, Total = p, year=as.Date("1/1/1900"), State="FL", original=0)

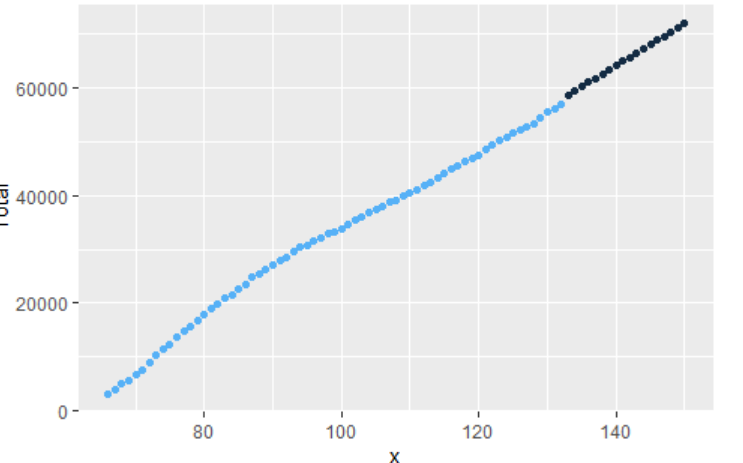
together = rbind(florida\_total, predicted)

plot(together$x, together$Total)

together %>% ggplot(aes(x, Total, color=original)) + geom\_point()

**Problem**

Use the data from Washington in June to predict the cases in the first five days of July using a linear model. Produce a chart like this:



|  |  |
| --- | --- |
| Group | Chart |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |

c = ncol(data)

wa\_data = data %>% gather("year", "cases", 3:c) %>% mutate(

year = as.Date(str\_replace(year, "X", ""), "%m.%d.%y"))

wa\_data = wa\_data %>% filter(State == "WA") %>%

filter(year >= "2020-6-1" & year <= "2020-6-30")

View(wa\_data)

plot(wa\_data$cases)

#group\_by / summarize (groups and aggregates)

#any summarization functions, special one n() = count

wa\_total = wa\_data %>%

group\_by(State, year) %>%

summarise(Total = sum(cases))

View(wa\_total)

plot(wa\_total$Total)

wa\_total$x = 1:nrow(wa\_total)

View(wa\_total)

plot(wa\_total$x, wa\_total$Total)

model = lm(Total ~ x, wa\_total)

summary(model)

plot(model$residuals)

wa\_total$original = 1

x\_predict = data.frame(x = 31:35)

p = predict(model, newdata = x\_predict)

predicted = data.frame(x = x\_predict, Total = p, year=as.Date("2020-7-1"), State="WA", original=0)

View(predicted)

together = rbind(wa\_total, predicted)

plot(together$x, together$Total)

together %>% ggplot(aes(x, Total, color=original)) + geom\_point()

*(connected extrapolation version, differences in* ***bold****)*

*wa\_data = data %>% filter(State == "WA") %>%*

*filter(Date >= "2020-06-01") %>% filter(Date <= "2020-06-30")*

*wa\_total = wa\_data %>% group\_by(State, Date) %>%*

*summarise(Total = sum(Cases))*

*wa\_total$x = 1:nrow(wa\_total)*

***totals\_data = data.frame(x = x , total = wa\_total$Total)***

***totals\_data = totals\_data%>% filter(x > 21)***

***plot (totals\_data$x, totals\_data$total)***

***model = lm(total ~ x, totals\_data)***

***summary(model)***

*plot(model$residuals)*

*wa\_total$original = 1*

*x\_predict = data\_frame(x = 31:35)*

*p = predict(model, newdata = x\_predict)*

*predicted = data.frame(x = x\_predict, Total = p, year = as.Date("1/1/1900"), State = "WA", original = 0)*

*together = rbind(wa\_total, predicted)*

*plot(together$x, together$Total)*

*together %>% ggplot(aes(x, Total, color = original)) + geom\_point()*

Week 4 - Data Preparation (gather, select, mutate, filter, group\_by)

Week 4 - Assignment pure week 4 material (use one or more of these operations)

Week 5 - Midterm → E2E problem from reading data, manipulating it, producing a prediction based on a model. One about using distributions on one variable. One using linear models.

Midterm - Take home, open book, open internet, no direct copy-paste and no collaboration with other students.

Given data set data about human height, calculate the chance that human female is between 64 and 65 inches.

No Session next week -> Spend the time on the Midterm

**Only output -> Number**