UC Wages

William Lovejoy

2/9/2022

The data for this project was requested from the University of California System via the FOIA. It contains the names, position, and gross yearly pay for all UC employees. The available data stretched from 2015 to 2020, and came in multiple separate csv and excel files. The only ones necessary for this project were CSV, so they were renamed to make organization easier before being added to R.

```
library(dplyr)

## Warning: package 'dplyr' was built under R version 4.0.5

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(scales)
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.0.5

df_2015 <- read.csv("C:/Users/William
Lovejoy/Documents/Codes/R/DataScience/UCWageCSV/2015.csv")
df_2016 <- read.csv("C:/Users/William
Lovejoy/Documents/Codes/R/DataScience/UCWageCSV/2016.csv")
df_2017 <- read.csv("C:/Users/William
Lovejoy/Documents/Codes/R/DataScience/UCWageCSV/2017.csv")
df_2018 <- read.csv("C:/Users/William
Lovejoy/Documents/Codes/R/DataScience/UCWageCSV/2018.csv")
df_2019 <- read.csv("C:/Users/William
Lovejoy/Documents/Codes/R/DataScience/UCWageCSV/2019.CSV")
df_2020 <- read.csv("C:/Users/William
Lovejoy/Documents/Codes/R/DataScience/UCWageCSV/2020.csv")
```

As always, check the structure of all your dataframes.

```r
str(df_2015)
str(df_2016)
str(df_2017)
str(df_2018)
str(df_2019)
str(df_2020)
```

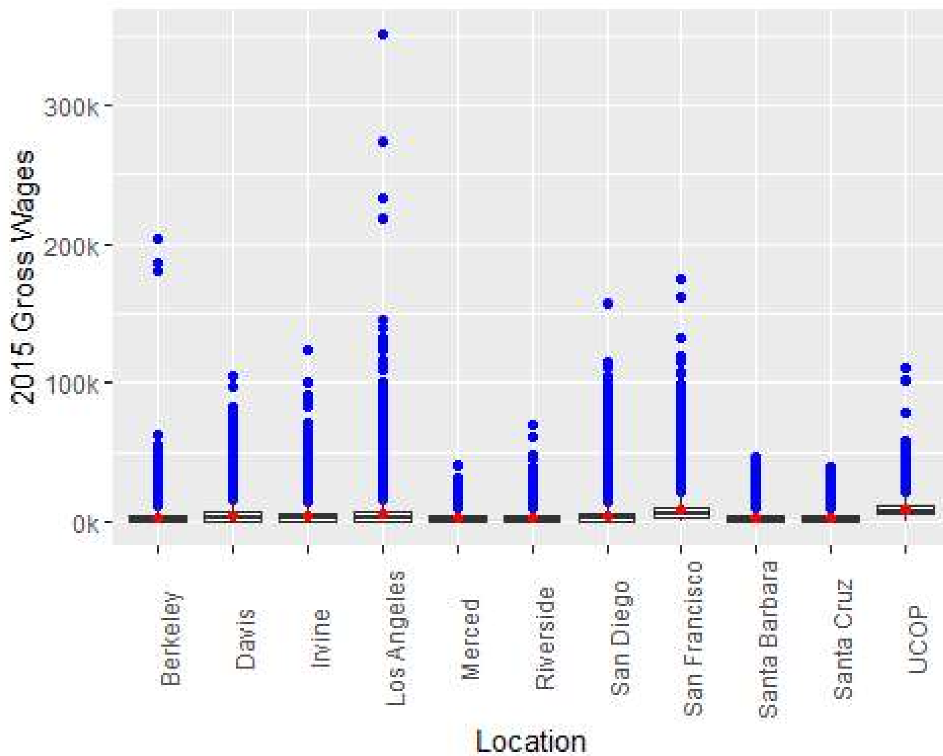
One of these data frames actually has a problem. One of the column names in df_2018 has a typo in it, which is something we'll need to fix in order to make sure it doesn't cause problems later.

```r
str(df_2018)

## 'data.frame':    310295 obs. of  8 variables:
##  $ Year              : int  2018 2018 2018 2018 2018 2018 2018 2018 2018 2018 ...
##  $ Location          : chr  "Berkeley" "Berkeley" "Hastings College Of Law" "Hastings College Of Law" ...
##  $ Lastt             : chr  "CARDOZA" "SIKORSKY" "COHEN" "SPECTER" ...
##  $ First             : chr  " ENA" " CHARLES" "MARSHA N." "SHANIN" ...
##  $ Title             : chr  "FOOD SVC WORKER SR" "TEACHER-UNEX" "ADJ. PROF" "LECTURER" ...
##  $ X2018.Gross       : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ X2018.Base        : num  0 0 1 1 0 0 0 0 0 0 ...
##  $ X2018.Overtime.Pay: num  1 0 0 0 0 0 0 0 0 0 ...

df_2018 <- rename(df_2018, Last = Lastt)
str(df_2018)

## 'data.frame':    310295 obs. of  8 variables:
##  $ Year              : int  2018 2018 2018 2018 2018 2018 2018 2018 2018 2018 ...
##  $ Location          : chr  "Berkeley" "Berkeley" "Hastings College Of Law" "Hastings College Of Law" ...
##  $ Last              : chr  "CARDOZA" "SIKORSKY" "COHEN" "SPECTER" ...
##  $ First             : chr  " ENA" " CHARLES" "MARSHA N." "SHANIN" ...
##  $ Title             : chr  "FOOD SVC WORKER SR" "TEACHER-UNEX" "ADJ. PROF" "LECTURER" ...
##  $ X2018.Gross       : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ X2018.Base        : num  0 0 1 1 0 0 0 0 0 0 ...
##  $ X2018.Overtime.Pay: num  1 0 0 0 0 0 0 0 0 0 ...
```

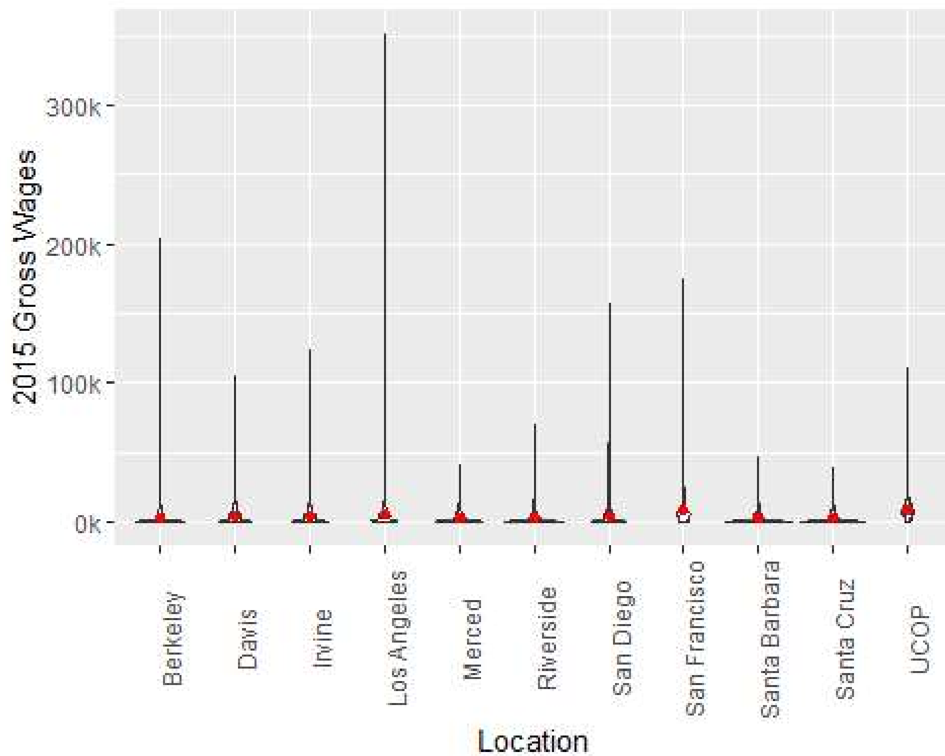
Next up, we'll combine the names columns to get a full name we can search through later if we want to, and then we can make some exploratory graphs.

```r
df_2015$Name <- paste(df_2015$First, df_2015$Last, sep = " ")
df_2016$Name <- paste(df_2016$First, df_2016$Last, sep = " ")
df_2017$Name <- paste(df_2017$First, df_2017$Last, sep = " ")
df_2018$Name <- paste(df_2018$First, df_2018$Last, sep = " ")
```

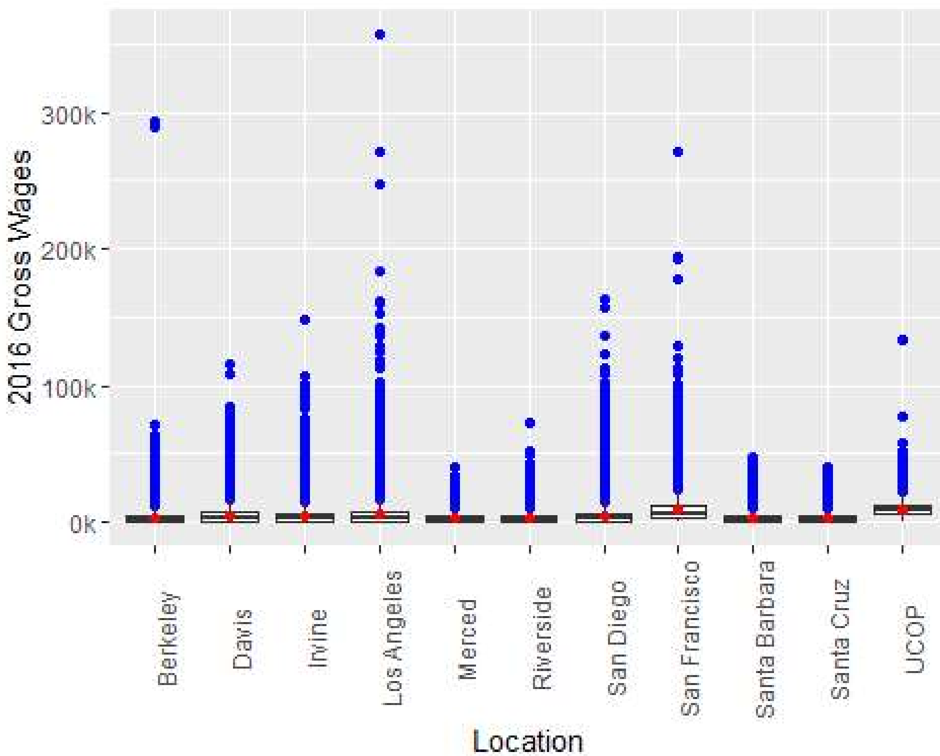
```
df_2019$Name <- paste(df_2019$First, df_2019$Last, sep = " ")
df_2020$Name <- paste(df_2020$First, df_2020$Last, sep = " ")

g <- ggplot(data = df_2015, mapping = aes(x = Location, y = X2015.Gross))
g + geom_boxplot(outlier.color = "blue") + scale_y_continuous(name = "2015
Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



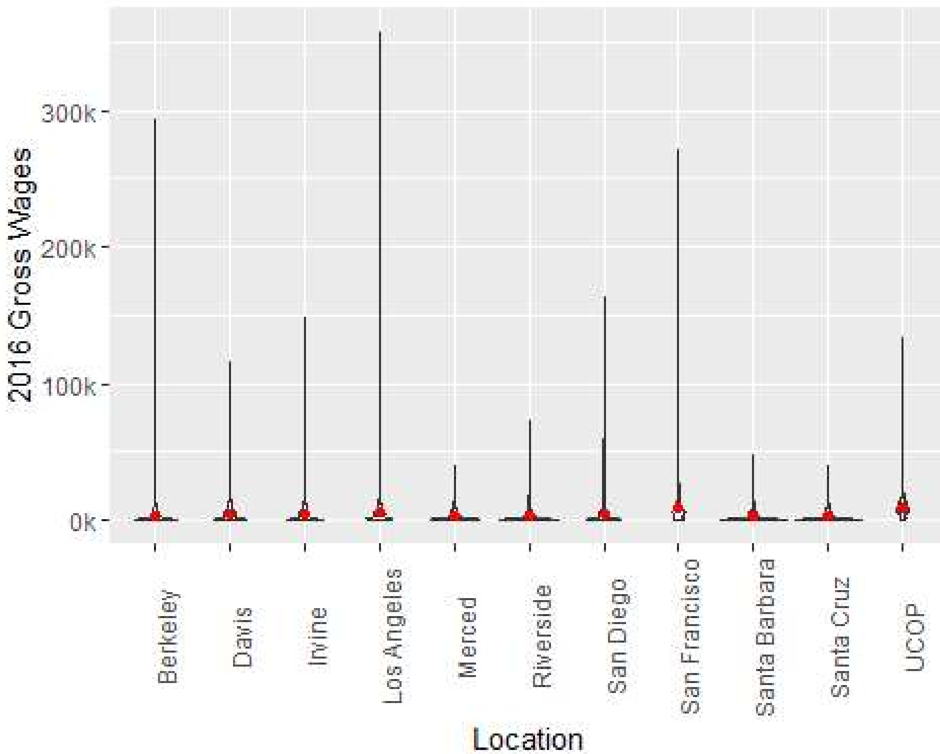

```
g + geom_violin() + scale_y_continuous(name = "2015 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



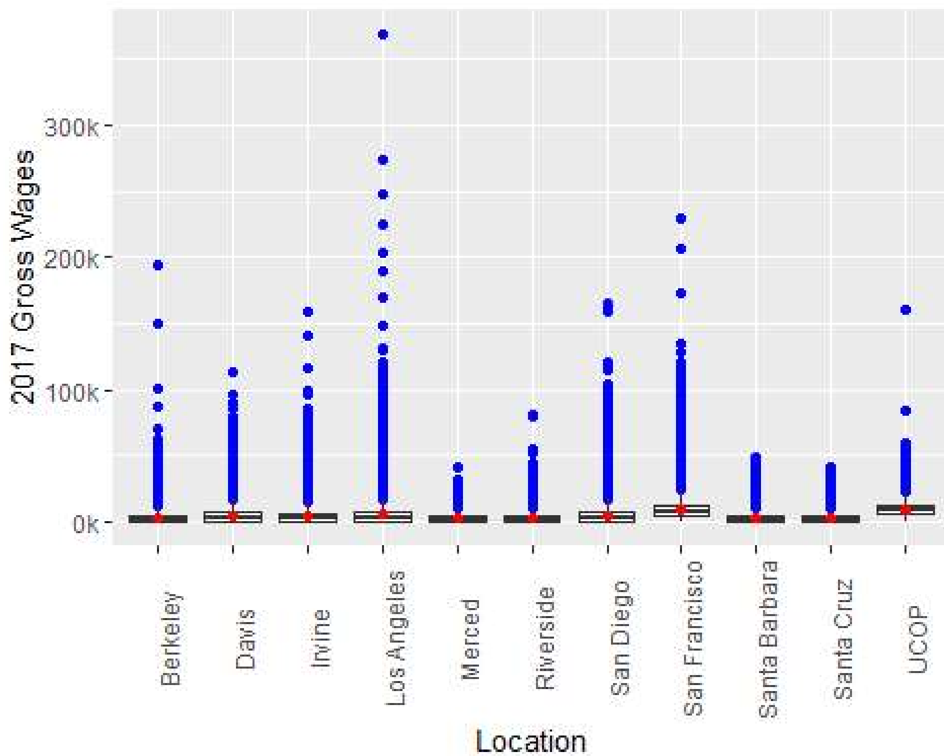

```r
a <- ggplot(data = df_2016, mapping = aes(x = Location, y = X2016.Gross))
a + geom_boxplot(outlier.color = "blue") + scale_y_continuous(name = "2016
Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



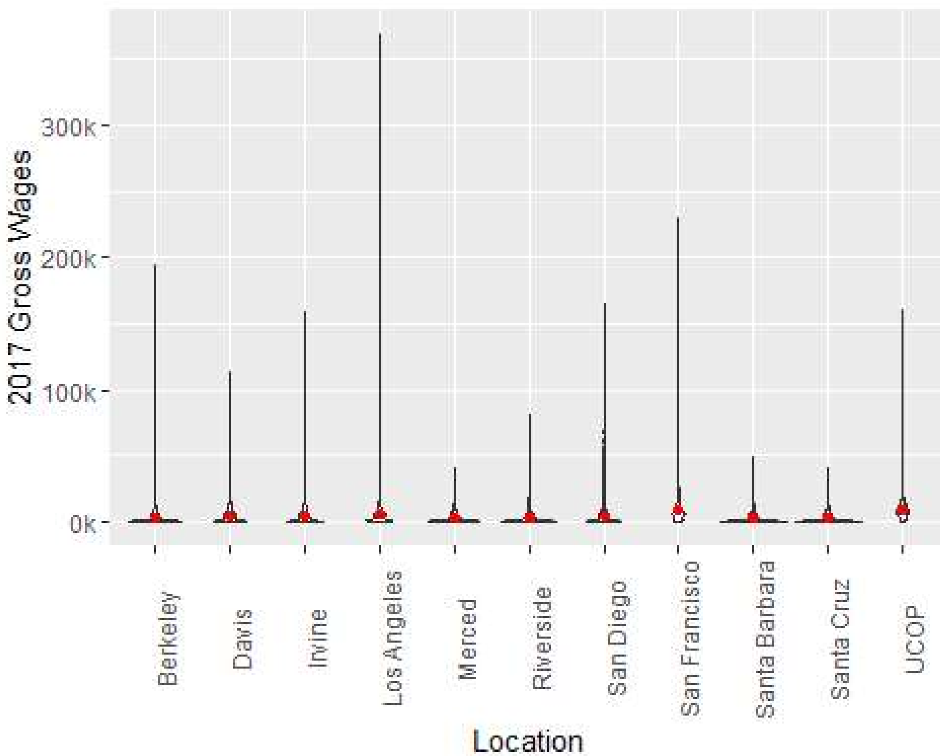

```
a + geom_violin() + scale_y_continuous(name = "2016 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



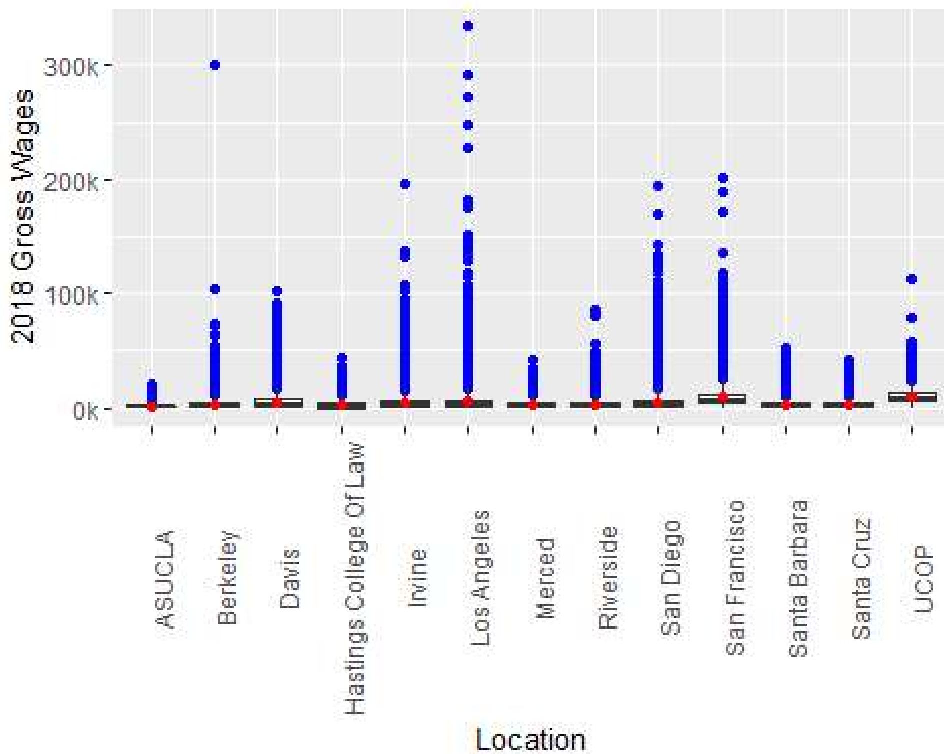

```
b <- ggplot(data = df_2017, mapping = aes(x = Location, y = X2017.Gross))
b + geom_boxplot(outlier.color = "blue") +  scale_y_continuous(name = "2017
Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



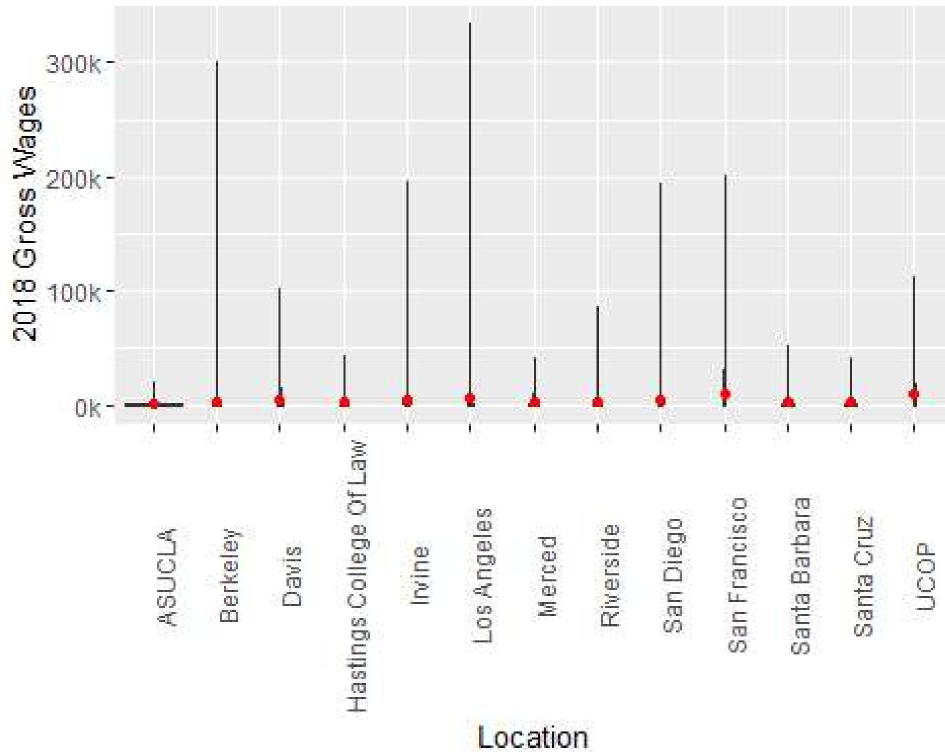

```r
b + geom_violin() +  scale_y_continuous(name = "2017 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



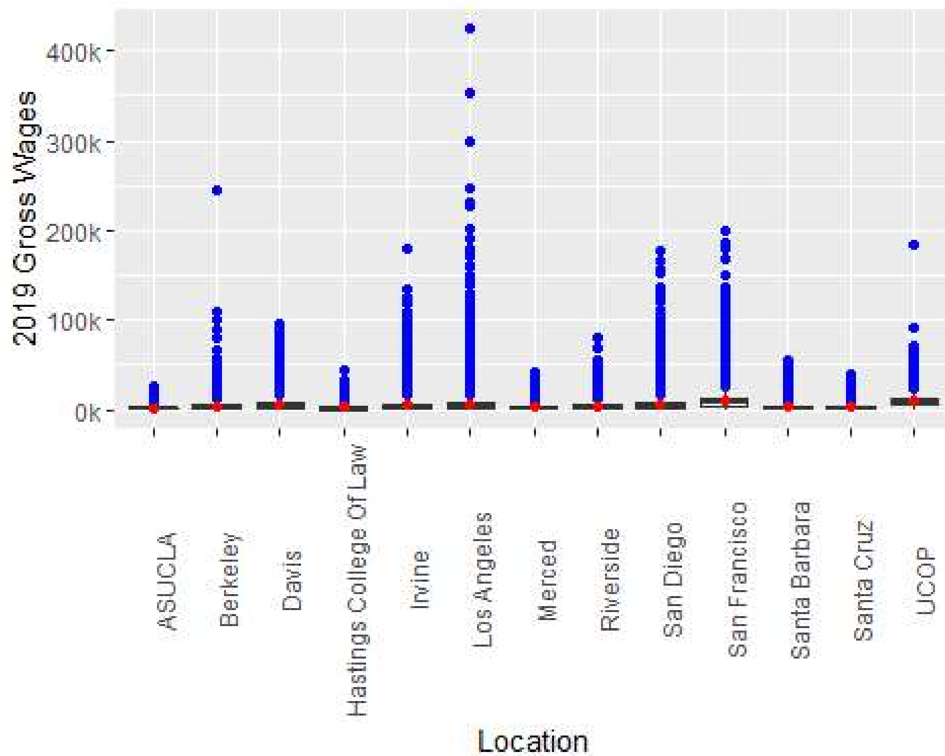

```
c <- ggplot(data = df_2018, mapping = aes(x = Location, y = X2018.Gross))
c + geom_boxplot(outlier.color = "blue") +  scale_y_continuous(name = "2018
Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



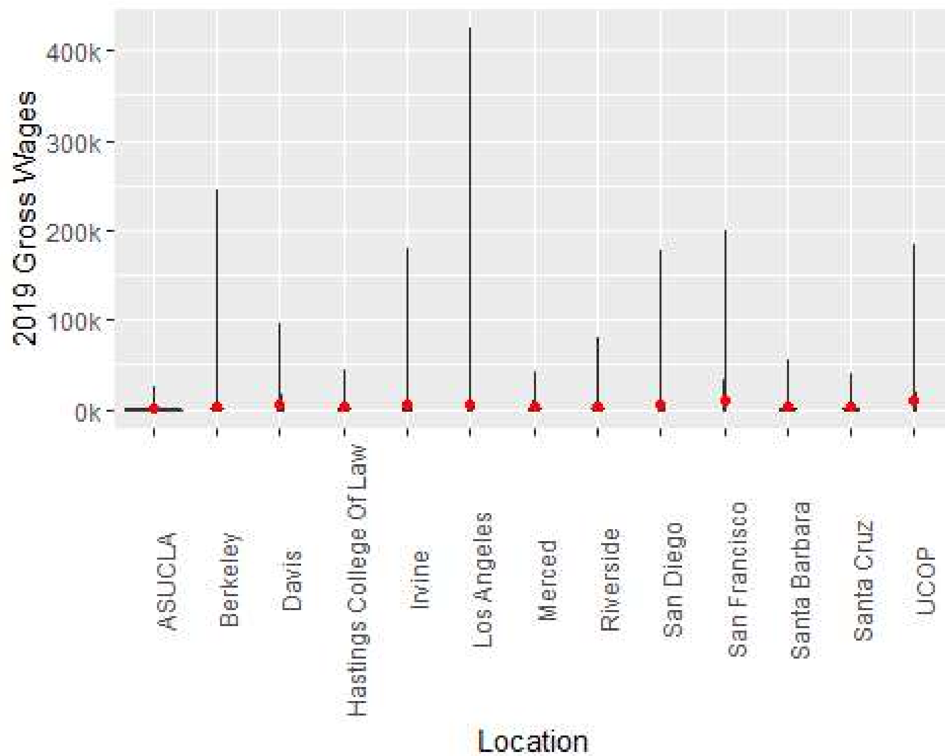

```
c + geom_violin() +  scale_y_continuous(name = "2018 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



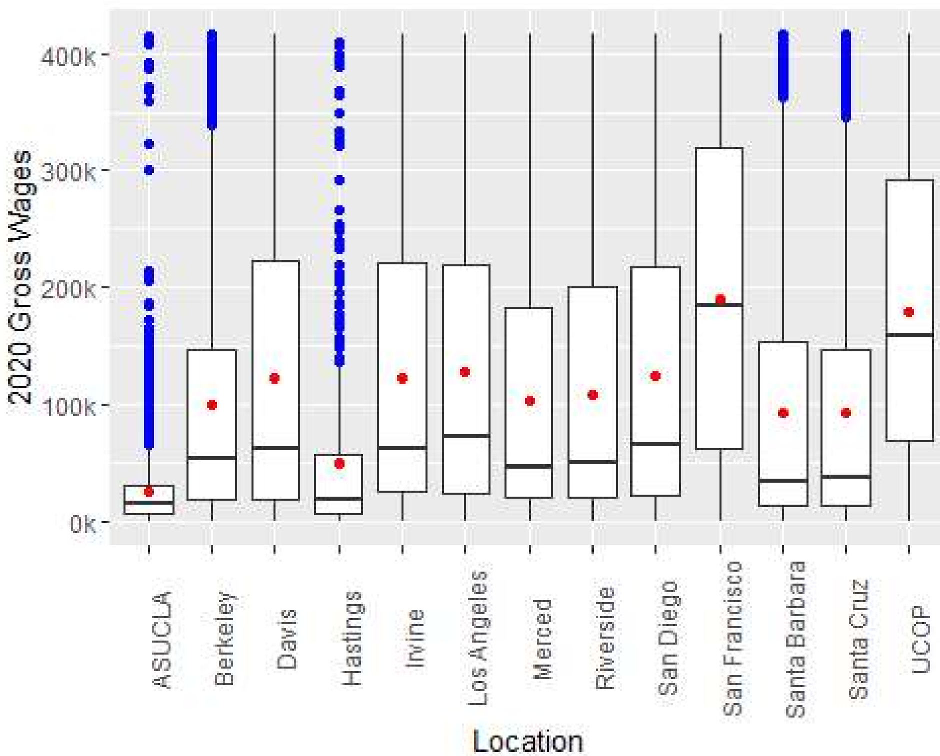

```
d <- ggplot(data = df_2019, mapping = aes(x = Location, y = X2019.Gross))
d + geom_boxplot(outlier.color = "blue") +  scale_y_continuous(name = "2019
Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



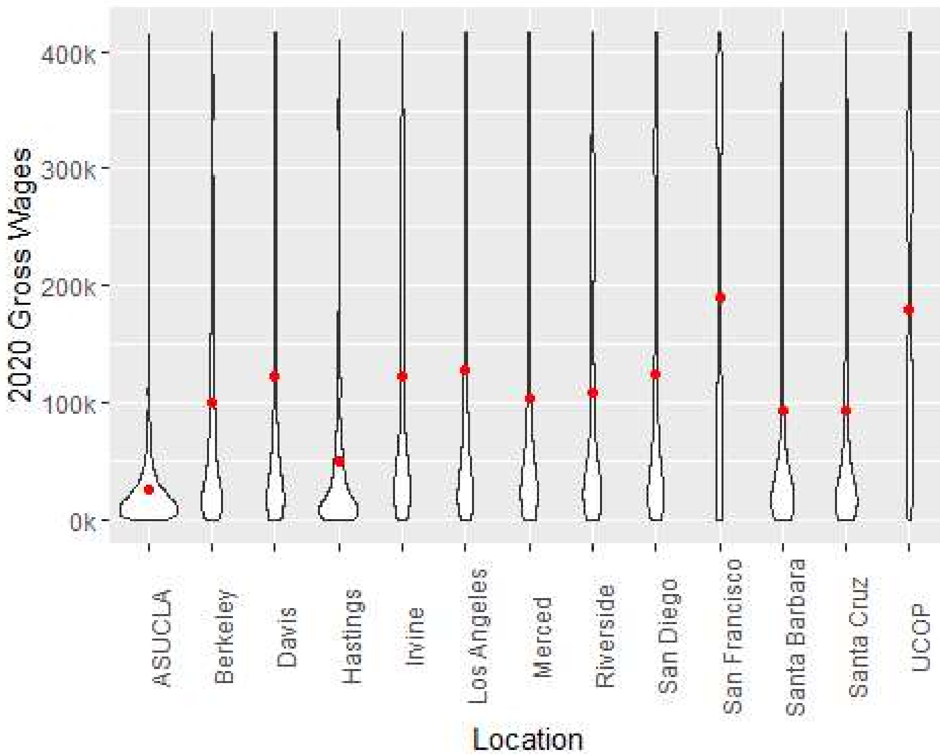

```
d + geom_violin() +  scale_y_continuous(name = "2019 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```




```
e <- ggplot(data = df_2020, mapping = aes(x = Location, y = X2020.Gross))
e + geom_boxplot(outlier.color = "blue") +  scale_y_continuous(name = "2020
Gross Wages",labels = label_number(suffix = "k", scale = 1e-2)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")

## Warning: Removed 1 rows containing non-finite values (stat_boxplot).

## Warning: Removed 1 rows containing non-finite values (stat_summary).
```

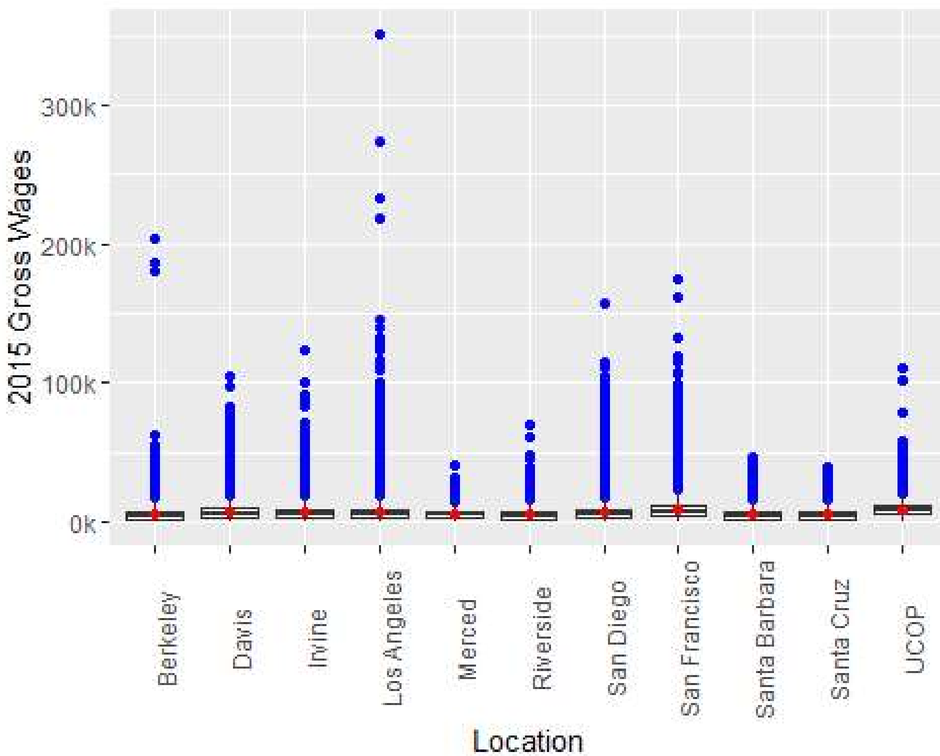



```
e + geom_violin() +  scale_y_continuous(name = "2020 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-2)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")

## Warning: Removed 1 rows containing non-finite values (stat_ydensity).

## Warning: Removed 1 rows containing non-finite values (stat_summary).
```

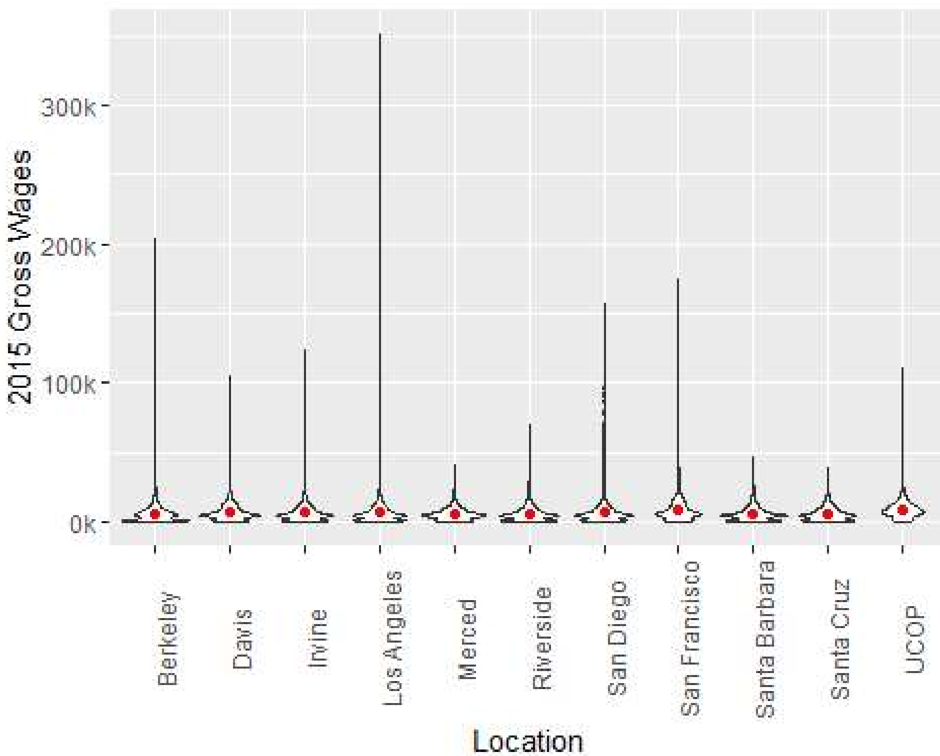


At first glance, these are not very evenly distributed graphs. The outliers take up so much space, and are so frequent, that they compress the rest of the data. However these charts also include the data from student workers, which typically won't be making as much money as someone working as a regular employee. So we're going to remove the student data, and remake the graphs.

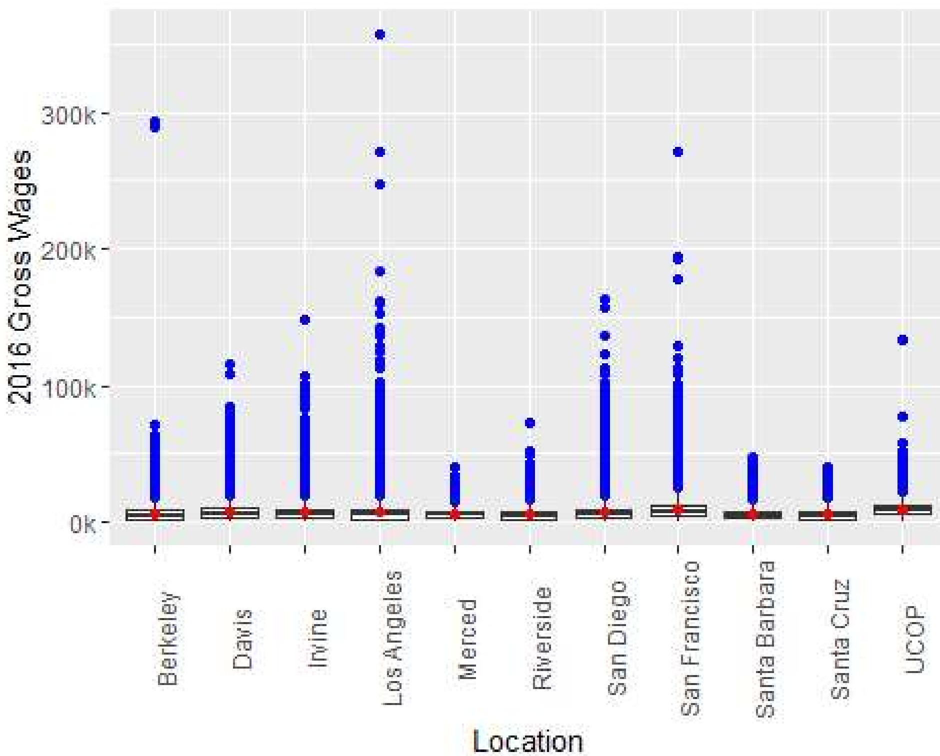
```
df_2015 <- subset(df_2015, First != "*****")
df_2016 <- subset(df_2016, First != "*****")
df_2017 <- subset(df_2017, First != "*****")
df_2018 <- subset(df_2018, First != "*****")
df_2019 <- subset(df_2019, First != "*****")
df_2020 <- subset(df_2020, First != "*****")

h <- ggplot(data = df_2015, mapping = aes(x = Location, y = X2015.Gross))
h + geom_boxplot(outlier.color = "blue") + scale_y_continuous(name = "2015 Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean, geom = "point", color = "red")
```



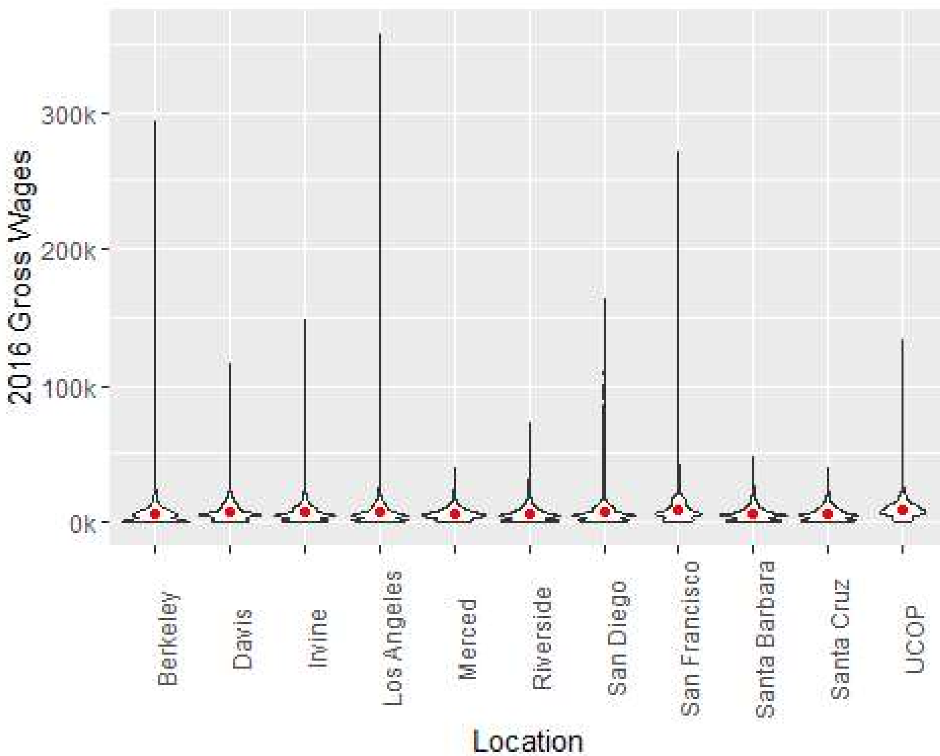

```r
h + geom_violin() + scale_y_continuous(name = "2015 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



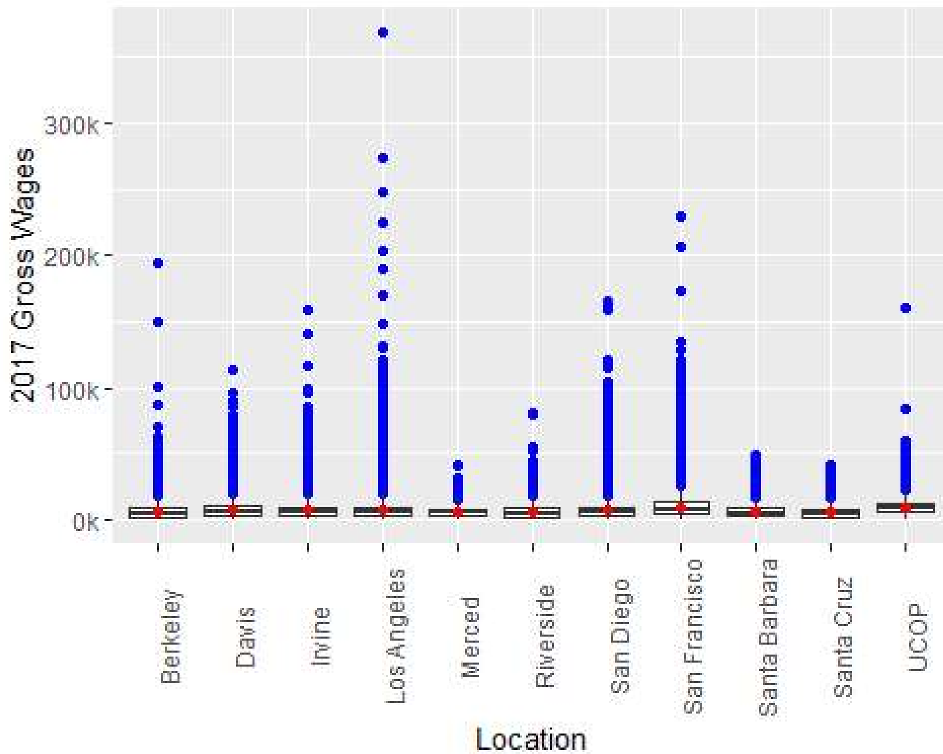

```r
i <- ggplot(data = df_2016, mapping = aes(x = Location, y = X2016.Gross))
i + geom_boxplot(outlier.color = "blue") + scale_y_continuous(name = "2016
Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



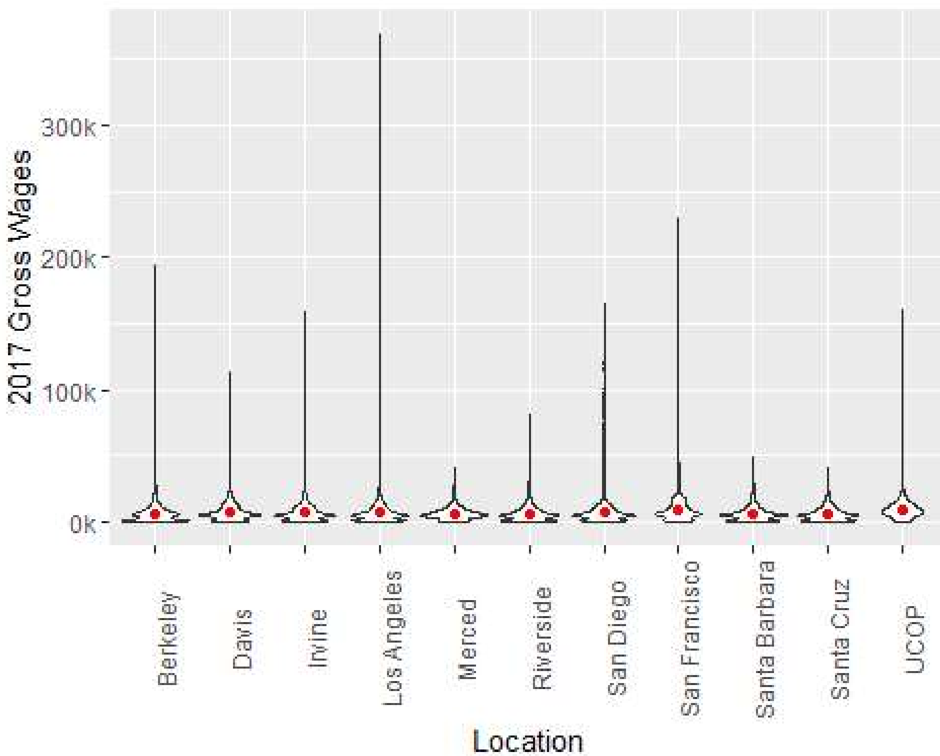

```
i + geom_violin() + scale_y_continuous(name = "2016 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



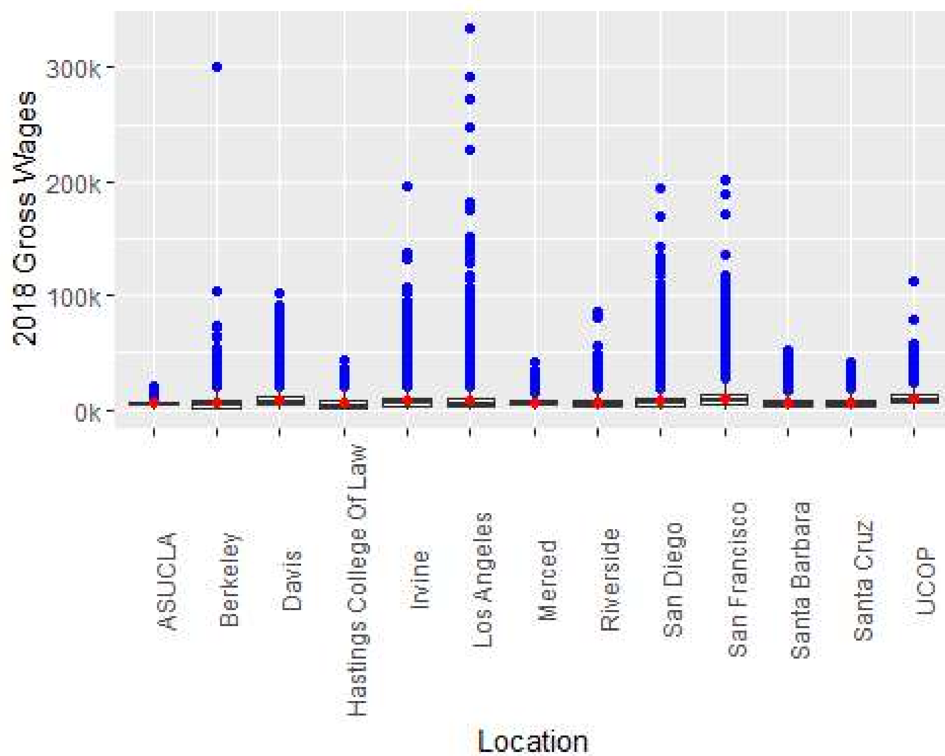

```r
j <- ggplot(data = df_2017, mapping = aes(x = Location, y = X2017.Gross))
j + geom_boxplot(outlier.color = "blue") +  scale_y_continuous(name = "2017
Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



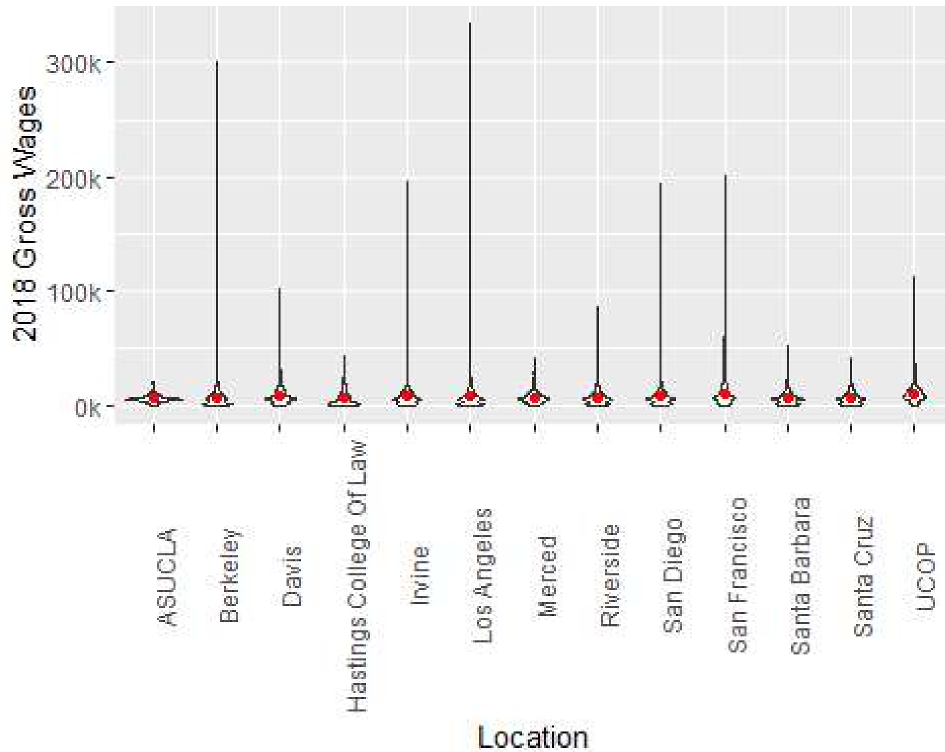

```
j + geom_violin() +  scale_y_continuous(name = "2017 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



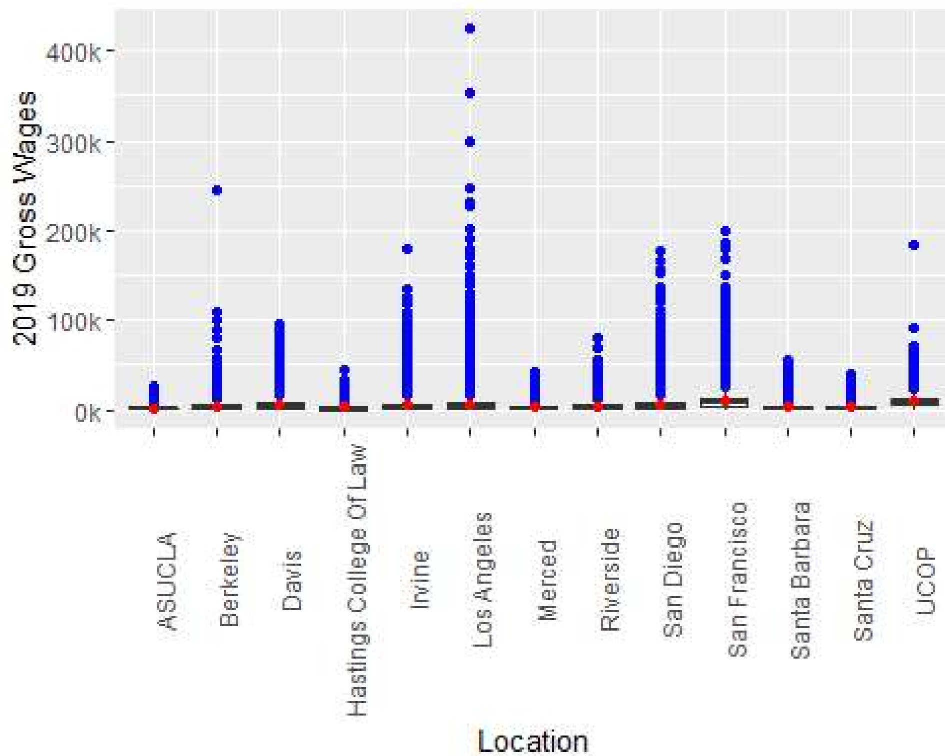

```r
k <- ggplot(data = df_2018, mapping = aes(x = Location, y = X2018.Gross))
k + geom_boxplot(outlier.color = "blue") +  scale_y_continuous(name = "2018
Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



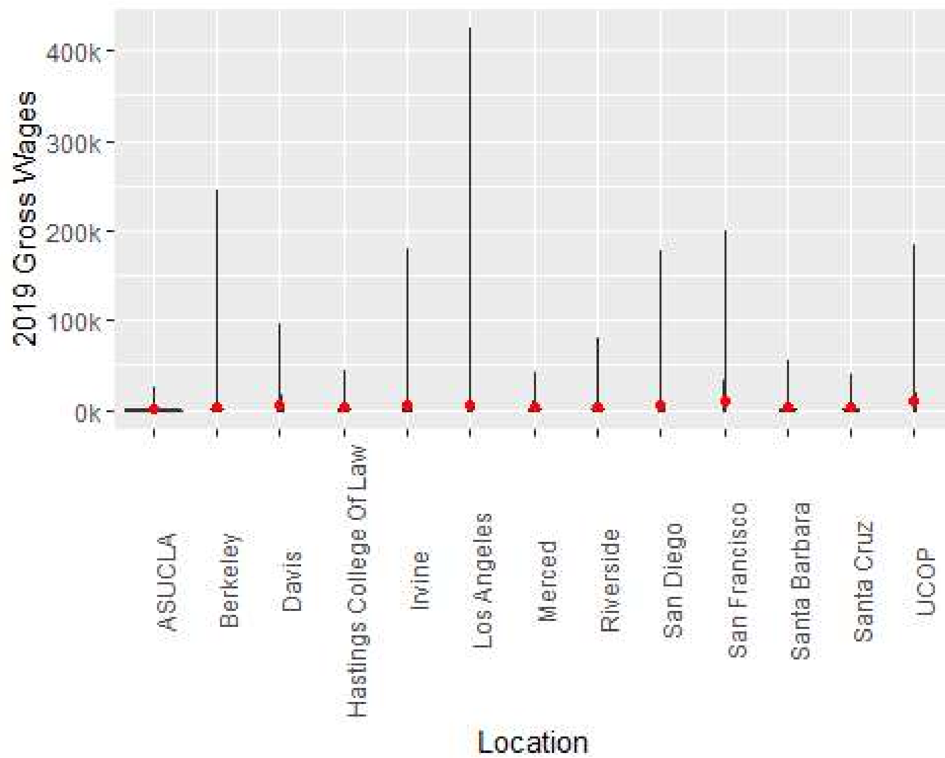

```
k + geom_violin() +  scale_y_continuous(name = "2018 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



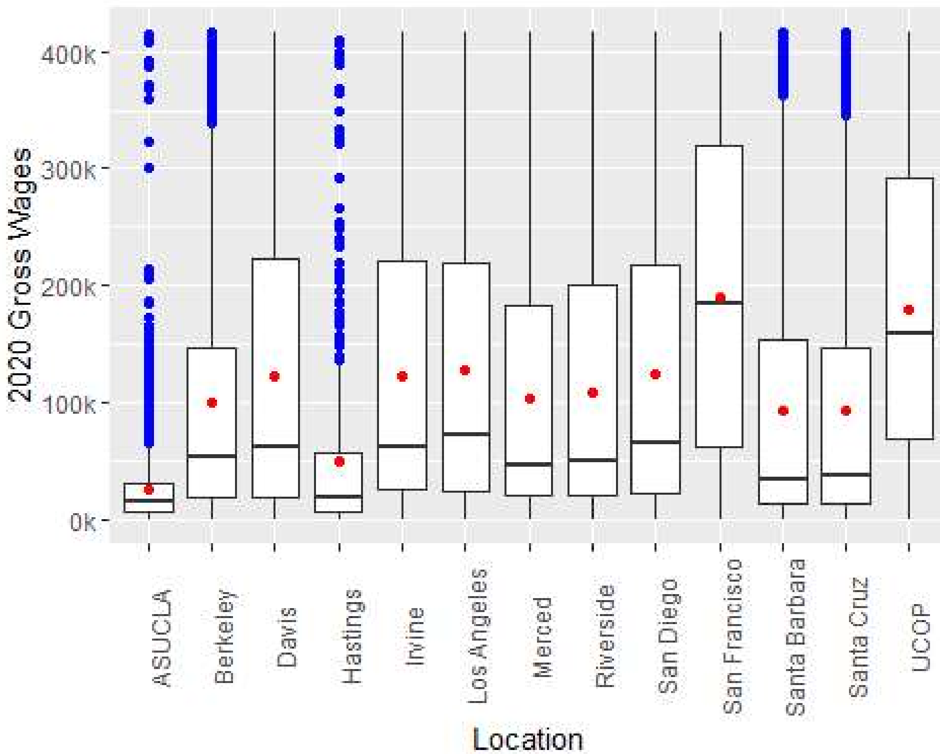

```
l <- ggplot(data = df_2019, mapping = aes(x = Location, y = X2019.Gross))
l + geom_boxplot(outlier.color = "blue") +  scale_y_continuous(name = "2019
Gross Wages",labels = label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```



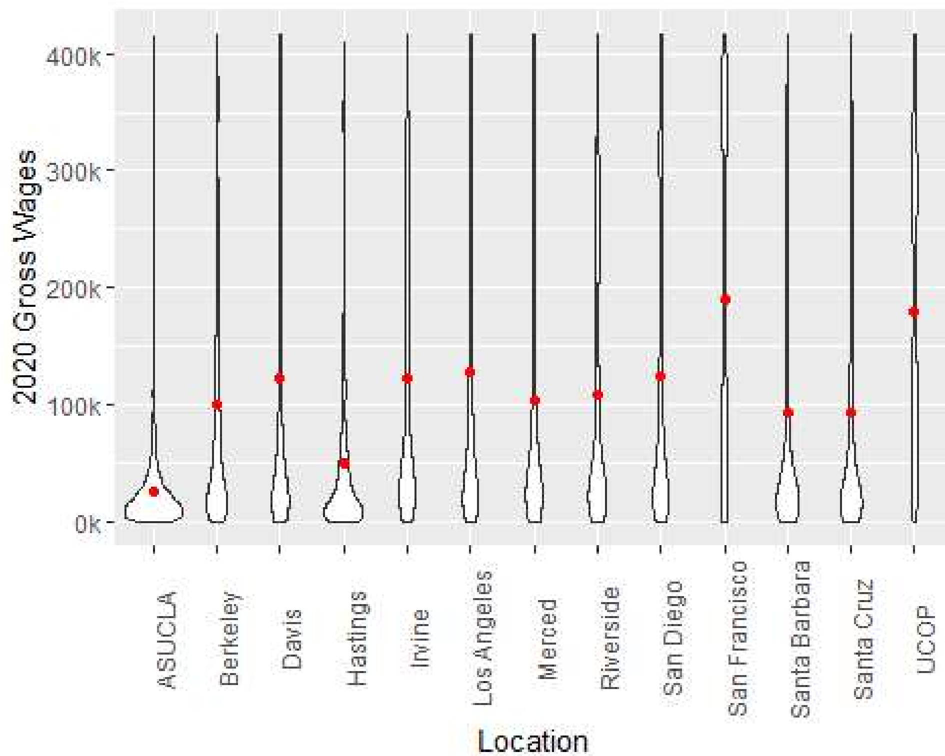

```
l + geom_violin() +  scale_y_continuous(name = "2019 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-4)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")
```




```
m <- ggplot(data = df_2020, mapping = aes(x = Location, y = X2020.Gross))
m + geom_boxplot(outlier.color = "blue") +  scale_y_continuous(name = "2020
Gross Wages",labels = label_number(suffix = "k", scale = 1e-2)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")

## Warning: Removed 1 rows containing non-finite values (stat_boxplot).

## Warning: Removed 1 rows containing non-finite values (stat_summary).
```




```
m + geom_violin() +  scale_y_continuous(name = "2020 Gross Wages",labels =
label_number(suffix = "k", scale = 1e-2)) +
  theme(axis.text.x = element_text(angle = 90)) + stat_summary(fun = mean,
geom = "point", color = "red")

## Warning: Removed 1 rows containing non-finite values (stat_ydensity).

## Warning: Removed 1 rows containing non-finite values (stat_summary).
```



That brought about some changes in the violin plots, although not so much in the boxplots. So let's break down the data by position and location to see if the mean pay per position title can tell us anything. We're also going to use this chance to calculate the average percent of change between each year's wages, across all the years, as well as what each wage should have risen to each year factoring in inflation by year in California. And finally, we'll use 2015 as the base year, and note down what gross pay should have been in 2020 using overall inflation in the state of California.. We'll do the same for all locations, but we won't list that code here. It's largely just copied off this batch below.

```
UCOP15 <- subset(df_2015, Location == "UCOP")
UCOP16 <- subset(df_2016, Location == "UCOP")
UCOP17 <- subset(df_2017, Location == "UCOP")
UCOP18 <- subset(df_2018, Location == "UCOP")
UCOP19 <- subset(df_2019, Location == "UCOP")
UCOP20 <- subset(df_2020, Location == "UCOP")


UCOP_titles_2015 <- UCOP15 %>%
  group_by(Title) %>%
  summarize(count15 = n(), Mean.Gross.15 = mean(X2015.Gross))

UCOP_titles_2016 <- UCOP16 %>%
  group_by(Title) %>%
  summarize(count16 = n(), Mean.Gross.16 =  mean(X2016.Gross))
```

```
UCOP_titles_2017 <- UCOP17 %>%
  group_by(Title) %>%
  summarize(count17 = n(), Mean.Gross.17 =  mean(X2017.Gross))

UCOP_titles_2018 <- UCOP18 %>%
  group_by(Title) %>%
  summarize(count18 = n(), Mean.Gross.18 =  mean(X2018.Gross))

UCOP_titles_2019 <- UCOP19 %>%
  group_by(Title) %>%
  summarize(count19 = n(), Mean.Gross.19 =  mean(X2019.Gross))

UCOP_titles_2020 <- UCOP20 %>%
  group_by(Title) %>%
  summarize(count20 = n(), Mean.Gross.20 =  mean(X2020.Gross))

UCOP1516 <- inner_join(UCOP_titles_2015, UCOP_titles_2016, by = "Title")
UCOP1718 <- inner_join(UCOP_titles_2017, UCOP_titles_2018, by = "Title")
UCOP1920 <- inner_join(UCOP_titles_2019, UCOP_titles_2020, by = "Title")
UCOP5678 <- inner_join(UCOP1516, UCOP1718, by = "Title")
UCOP <- inner_join(UCOP5678, UCOP1920, by = "Title")


UCOP <- mutate(UCOP,
       Difference.15.16 = Mean.Gross.16 - Mean.Gross.15,
       Difference.16.17 = Mean.Gross.17 - Mean.Gross.16,
       Difference.17.18 = Mean.Gross.18 - Mean.Gross.17,
       Difference.18.19 = Mean.Gross.19 - Mean.Gross.18,
       Difference.19.20 = Mean.Gross.20 - Mean.Gross.19,
       Percent.15.16 = Difference.15.16 / Mean.Gross.15 * 100,
       Percent.16.17 = Difference.16.17 / Mean.Gross.16 * 100,
       Percent.17.18 = Difference.17.18 / Mean.Gross.17 * 100,
       Percent.18.19 = Difference.18.19 / Mean.Gross.18 * 100,
       Percent.19.20 = Difference.19.20 / Mean.Gross.19 * 100)

UCOP$Average.Percent.Change <-  rowMeans(subset(UCOP, select =
c(Percent.15.16, Percent.16.17, Percent.17.18, Percent.18.19,
Percent.19.20)))

summary(UCOP$Average.Percent.Change)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -23.337 -14.500  -9.851  -7.501  -6.253 110.675

#To see what inflation suggests the new wages should be, I used the CA dept.
of Finance CPI Table.
UCOP <- mutate(UCOP,
               Inflation.Calculation.15.16 = (Mean.Gross.15 + (Mean.Gross.15
```

```
* 0.023)),
               Inflation.Calculation.16.17 = (Inflation.Calculation.15.16 +
(Inflation.Calculation.15.16 * 0.023)),
               Inflation.Calculation.17.18 = (Inflation.Calculation.16.17 +
(Inflation.Calculation.16.17 * 0.029)),
               Inflation.Calculation.18.19 = (Inflation.Calculation.17.18 +
(Inflation.Calculation.17.18 * 0.037)),
               Inflation.Calculation.19.20 = (Inflation.Calculation.18.19 +
(Inflation.Calculation.18.19 * 0.03)))

UCOP$Inflated.20 <- UCOP$Mean.Gross.15 + (UCOP$Mean.Gross.15 * .1038223)

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  -29.149  -15.003  -10.510    4.129   -3.325 1662.907

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -38.9576 -13.0228  -9.1819  -1.3500  -0.6071 906.6650

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -42.0046 -14.0935  -8.8704   2.7534  -0.9041 878.7642

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -32.475 -12.134  -6.455   2.713   2.283 530.534

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  -26.987  -11.522   -6.272    6.654    2.573 1439.509

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -44.3417 -12.2382  -6.1167  -3.3163   0.6995  82.4726

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -25.068 -10.928  -5.580  10.298   2.035 721.205

##      Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
##  -36.2328  -12.9483   -8.0386    6.3767    0.9781 1978.6220

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -28.482 -11.452  -3.415   7.532   9.649 227.093

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -27.705 -12.619  -5.681  -3.115   1.325  69.350
```

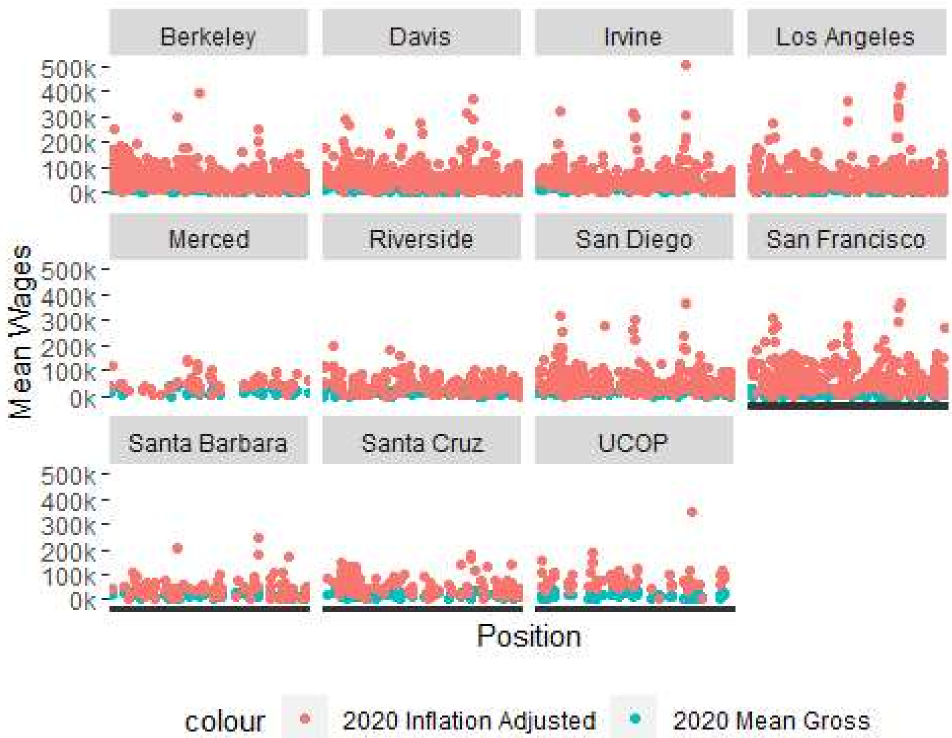
That’s a lot of negative percent changes. So let’s compare total inflation between 2015 and 2020. None of the chunked down data have the location within them, so we’ll assign that, and merge them all into one larger dataframe we can work with. Since we’ll be looking at the difference between 2015 and 2020, we won’t need to use the data from newer UC’s such as Hastings.

```
Berkeley$Location <- "Berkeley"
Davis$Location <- "Davis"
Irvine$Location <- "Irvine"
```

```r
Los_Angeles$Location <- "Los Angeles"
Merced$Location <-  "Merced"
Riverside$Location <- "Riverside"
San_Diego$Location <- "San Diego"
SF$Location <- "San Francisco"
Santa_Barbara$Location <- "Santa Barbara"
Santa_Cruz$Location <- "Santa Cruz"
UCOP$Location <- "UCOP"

UCWages <- do.call("rbind", list(Berkeley, Davis, Irvine, Los_Angeles,
Merced, Riverside, San_Diego, SF, Santa_Barbara, Santa_Cruz, UCOP))
```

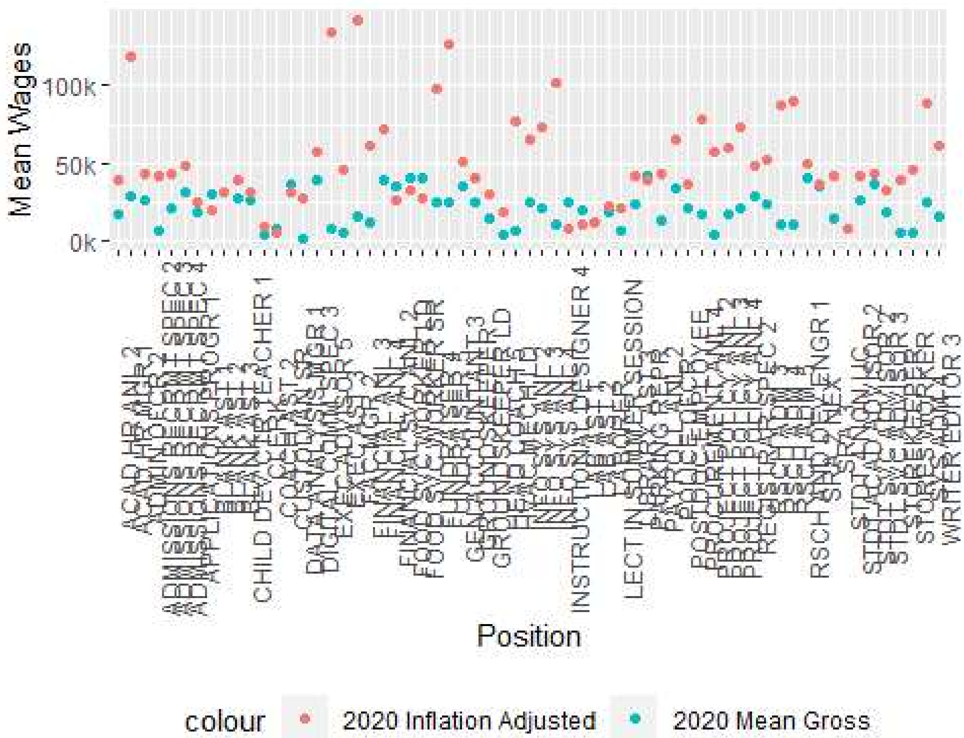
Now that that's done, let's make some fresher graphs with the data. Since there are 12 UC's, and we only want a broad overview to start, we won't use any x-axis labels, and we'll use a facet wrap to see all the data.

```r
colors <- c("2020 Mean Gross" = "red", "2020 Inflation Adjusted" = "green")

t <- ggplot(UCWages, mapping = aes(x = Title))
t + geom_point(mapping = aes(y = Mean.Gross.20, color = "2020 Mean Gross")) +
  geom_point(mapping = aes(y = Inflated.20, color = "2020 Inflation
Adjusted")) +
  facet_wrap(~Location) + scale_y_continuous(name = "Mean Wages",labels =
label_number(suffix = "k", scale = 1e-3)) +
  theme(axis.text.x = element_blank(), legend.position = "bottom") +
xlab("Position")
```



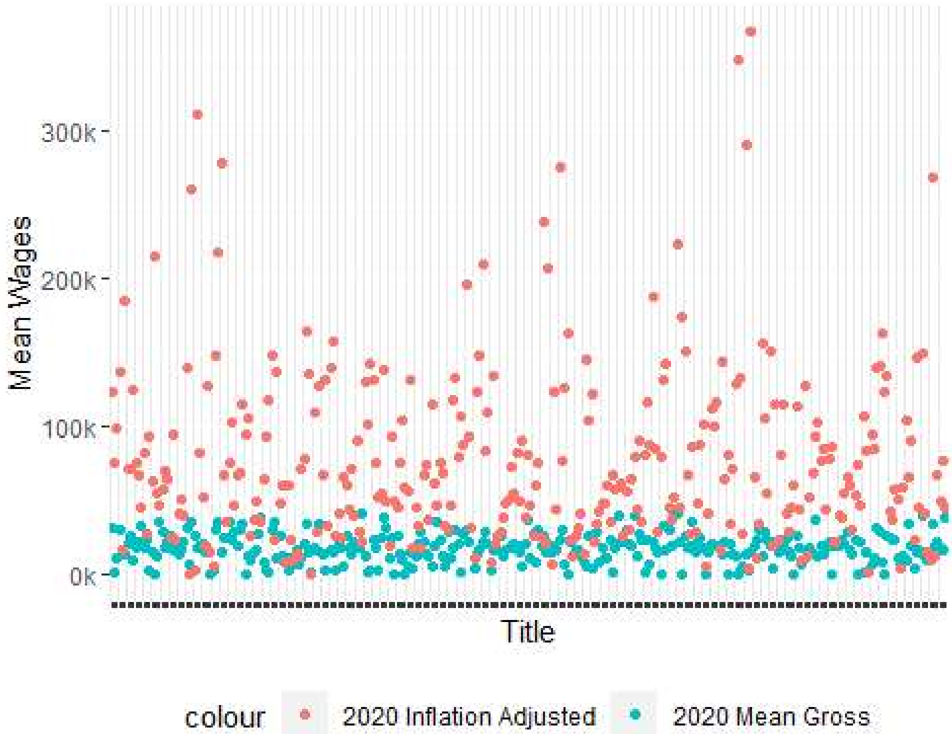
That is not great to see. The difference between mean gross pay and what it should be when adjusted for inflation over just 5 years is a lot more distinct than I would have liked. So let's look at some closer examples. Merced has a fairly small sample size, so we'll look at that graph in more detail.

```
m <- ggplot(Merced, mapping = aes(x = Title))
m + geom_point(mapping = aes(y = Mean.Gross.20, color = "2020 Mean Gross")) +
  geom_point(mapping = aes(y = Inflated.20, color = "2020 Inflation
Adjusted")) +
  scale_y_continuous(name = "Mean Wages", labels = label_number(suffix = "k",
scale = 1e-3)) +
  theme(axis.text.x = element_text(angle = 90), legend.position = "bottom") +
xlab("Position")
```



That's actually not the worst. A couple positions actually have a higher average than the 2020 calculated inflation. Coach Assistant 2, as well as some financial analysts, are doing better than inflation. So now we'll look at UC San Francisco, which is in one of the most expensive

American cities to live in.




```
## # A tibble: 1 x 31
##   Title         count15 Mean.Gross.15 count16 Mean.Gross.16 count17
Mean.Gross.17
##   <chr>           <int>         <dbl>   <int>         <dbl>   <int>
<dbl>
## 1 PROF OF CLI~     271       332653.     283       342140.     297
361759.
## # ... with 24 more variables: count18 <int>, Mean.Gross.18 <dbl>,
## #   count19 <int>, Mean.Gross.19 <dbl>, count20 <int>, Mean.Gross.20
<dbl>,
## #   Difference.15.16 <dbl>, Difference.16.17 <dbl>, Difference.17.18
<dbl>,
## #   Difference.18.19 <dbl>, Difference.19.20 <dbl>, Percent.15.16 <dbl>,
## #   Percent.16.17 <dbl>, Percent.17.18 <dbl>, Percent.18.19 <dbl>,
## #   Percent.19.20 <dbl>, Average.Percent.Change <dbl>,
## #   Inflation.Calculation.15.16 <dbl>, Inflation.Calculation.16.17 <dbl>,
...

## [1] 7.257885

## [1] 14.51577
```

Unfortunately, there are just too many job positions at UCSF to all fit on the x axis. So we stripped off the labels to prevent crowding. But this data is much more disheartening for employees. Some positions seem to be making less than $50k a year, while inflation necessitates, they make almost $400k gross on average. The worst victim of this is the "Professor of Clin-HCOMP" Position. In 2020, the gross pay for this position averaged $15,096.4, which is $374,742.5 less than was made in 2019. Staff was also cut by 323 people for the same position in that time frame. Meaning that along with reduced staff members, they received a severe pay cut. They should be making $367,189.70 a year with that title at UCSF, instead the position saw a 96% pay cut. Assuming that this position is full time (which means roughly 2,080 working hours a year for an average of 40 a week), those individuals would be making roughly $7.26 an hour, and $14.52 an hour if they only worked 20-hour weeks in a part-time position (which typically means they would be without benefits). And according to the MIT Living Wage Calculator, a single adult with no children (working 2,080 hours a year) needs to be making $22.88/hour to live in the San Francisco-Oakland-Hayward area. These people cannot live with these wages. The closest they get to living on this wage (according to MIT) is if there are 2 adults and 0 children both working full time in the household. However, this only brings the living wage down to $16.23/hour, a ~30% decrease.

Based on these graphs and data analysis, the UC System's gross earnings have not kept up with inflation in just the past 5 years.

```
t.test(UCWages$Mean.Gross.20, UCWages$Inflated.20, paired = TRUE)

##
##  Paired t-test
##
## data:  UCWages$Mean.Gross.20 and UCWages$Inflated.20
## t = -43.117, df = 2633, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -44082.67 -40247.50
## sample estimates:
## mean of the differences
##               -42165.09
```

As a final check to see if we've only been observing outliers, we ran a simple paired t-test to compare the values for the Average gross pay in 2020, with what should be paid according to inflation over just 5 years. We calculated a value of $p < 2.2e-16$. In long form, that's 0.00000000000000022. The standard used to say if 2 sets of data are statistically different is $p < 0.05$. This lends very strong evidence to our earlier statement that the UC system in the state of California does not keep up with rising costs of living, and that they are failing their employees.