Assignment 1

2023-02-07

Import Data

```
NL_data = read.csv("data_month_NL.csv")
GDP_data = read_dta("Region by year GDP panel.dta")
GT_NL_data = read_rds("data_NL_GT.rds")
view(NL_data)
view(GDP_data)
view(GT_NL_data)
```

New dataset

```
all_data1 = left_join(NL_data, GDP_data, by = c("year"="year", "reg"="reg"))
all_data = left_join(all_data1, GT_NL_data, by = c("year"="year", "reg"="reg", "name"="name"))
all_data=all_data %>%
  mutate(NLI=nl_sum_4/(area_sq_km - 0.141*nl_nodata_4),NLI2 = nl_mean_4/nl_std_4, Date = as.yearmon(pas
NL_data=NL_data %>%
  mutate(day= "01", Date= as.Date(with(NL_data, paste(year, month, day,sep="-")), "%Y-%m-%d"))
```

In this new dataset, there will be repeated values that do not necessarily match up to the month. Any variable besides the NL information or the tw does not correspond to the given month.

This project we will be focusing on the following regions: Odeska, Sevastopilska, Krym, Mykolaivska, Khersonska

Area of Region

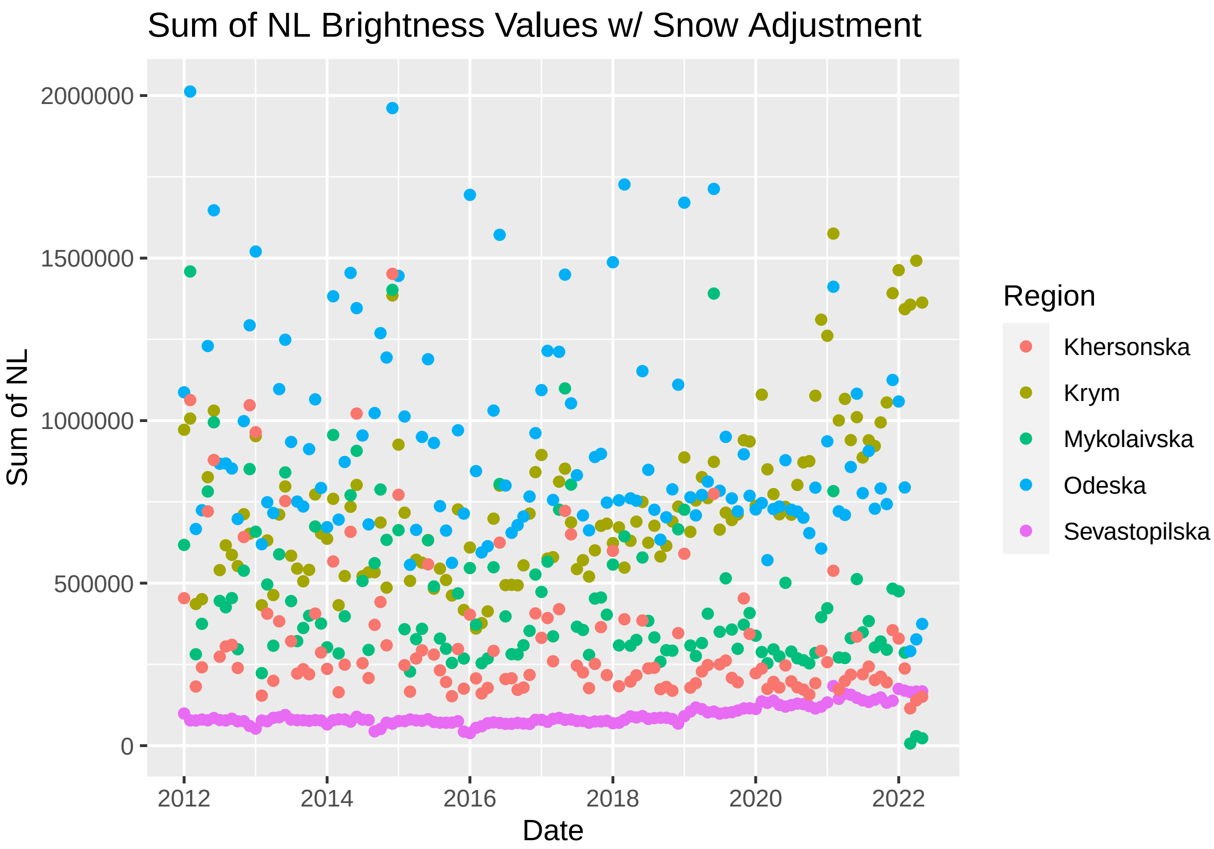
An interesting aspect of each region to note before beginning analyses is the size. In order from largest region to smallest:

1. Odeska (33,347.50 sq km)
2. Khersonska (26,656.97 sq km)
3. Krym (25,558.45 sq km)
4. Mykolaivska (24,015.76 sq km)
5. Sevastopilska (57.44 sq km)

Sum of NL

```
region_subset = subset(all_data, reg == "UA_51"|reg == "UA_40"|reg == "UA_43"|reg == "UA_48"|reg == "UA_
```

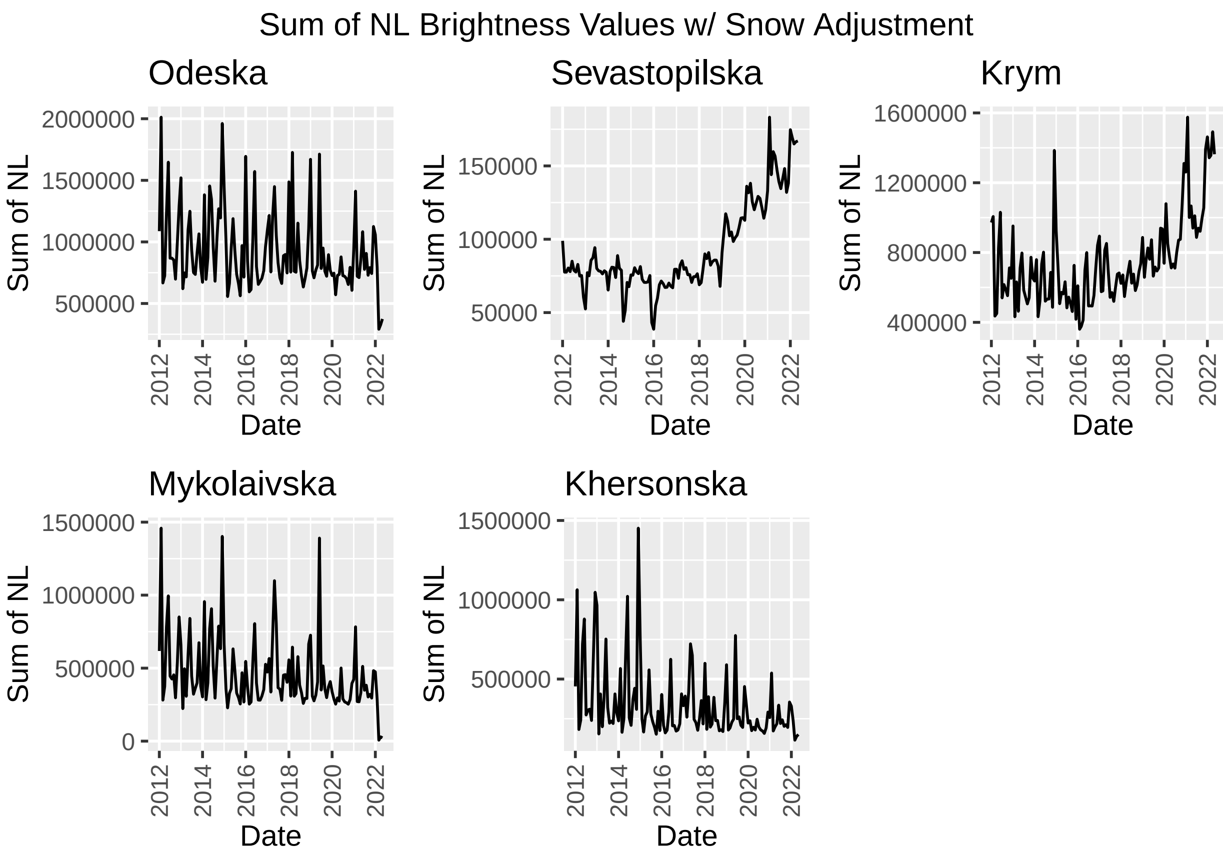
```
graph1_data = subset(NL_data, reg == "UA_51"|reg == "UA_40"|reg == "UA_43"|reg == "UA_48"|reg == "UA_65
graph1_data = drop_na(graph1_data)
lum_overtime = ggplot(graph1_data, aes(x = Date, y= nl_sum_4, color=name))+ geom_point()+labs(title= "Su

lum_overtime
```



Above shows the sum of night lights after adjusted for snow for each reason. There are clear regional differences with Sevastopilska having the smallest on average sum and Odeska having the largest. These numbers tend to make sense for the most part in terms if size of the region with Odeska being the largest region and Sevastopilska being the smallest. To get a better look at these sums, we will look at each region individually.

```
Odeska_sum = ggplot( data = subset(graph1_data, reg == "UA_51"), aes(x = Date, y= nl_sum_4))+ geom_line

Sevastopilska_sum = ggplot( data = subset(graph1_data, reg == "UA_40"), aes(x = Date, y= nl_sum_4))+ geo

Krym_sum = ggplot( data = subset(graph1_data, reg == "UA_43"), aes(x = Date, y= nl_sum_4))+ geom_line()+

Mykolaivska_sum = ggplot( data = subset(graph1_data, reg == "UA_48"), aes(x = Date, y= nl_sum_4))+ geom_

Khersonska_sum = ggplot( data = subset(graph1_data, reg == "UA_65"), aes(x = Date, y= nl_sum_4))+ geom_l
```

```
grid.arrange(Odeska_sum,Sevastopilska_sum,Krym_sum,Mykolaivska_sum,Khersonska_sum, ncol=3, top = "Sum of
```

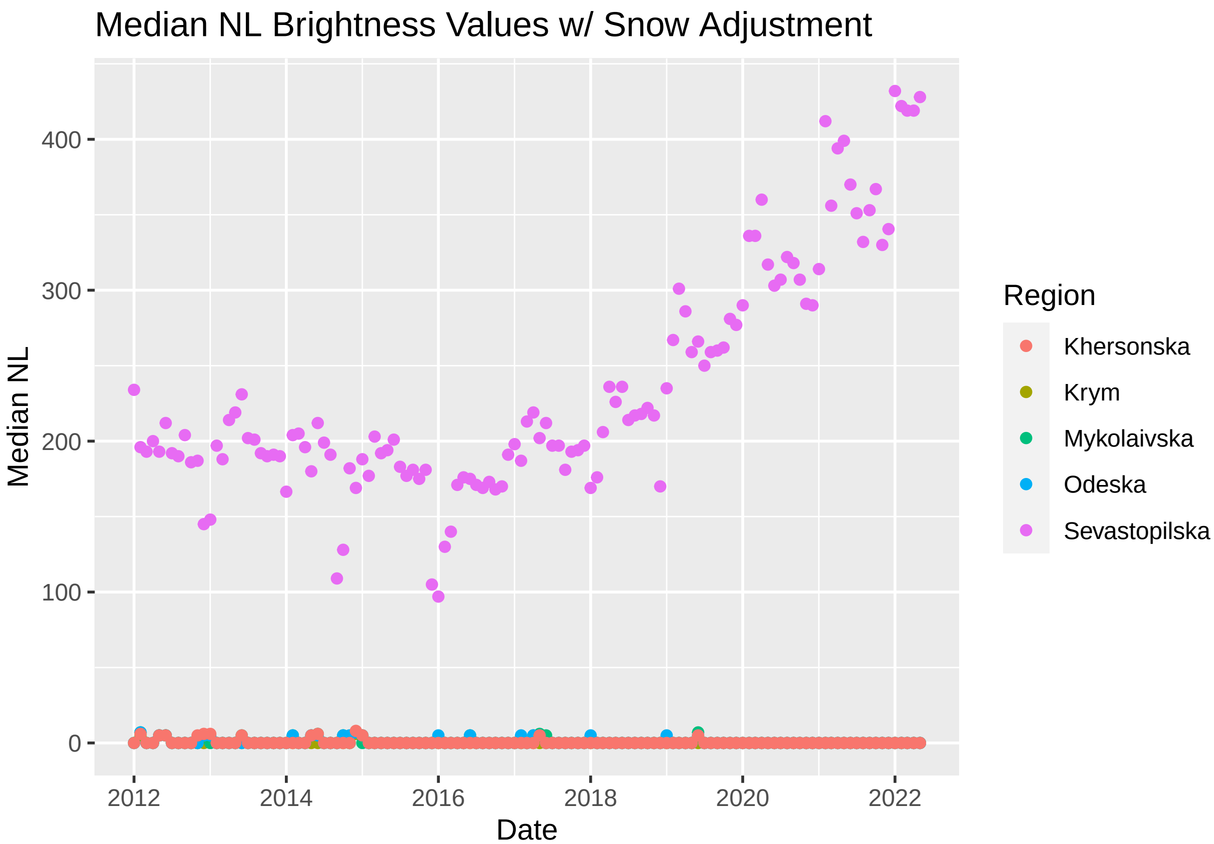


This visual provides a clearer view of trends over time by region. Odeska, Mykolaivska, and Khersonska have similar trends over the course of the ten years. Both Sevastopilska and Krym gradually increase over the ten years.

Median NL

We will now look at the median luminosity of NL in a similar manner as the sum. Medians can be informative as they are not impacted by large outliers and can give a reasonable estimate for luminosity levels in the region.

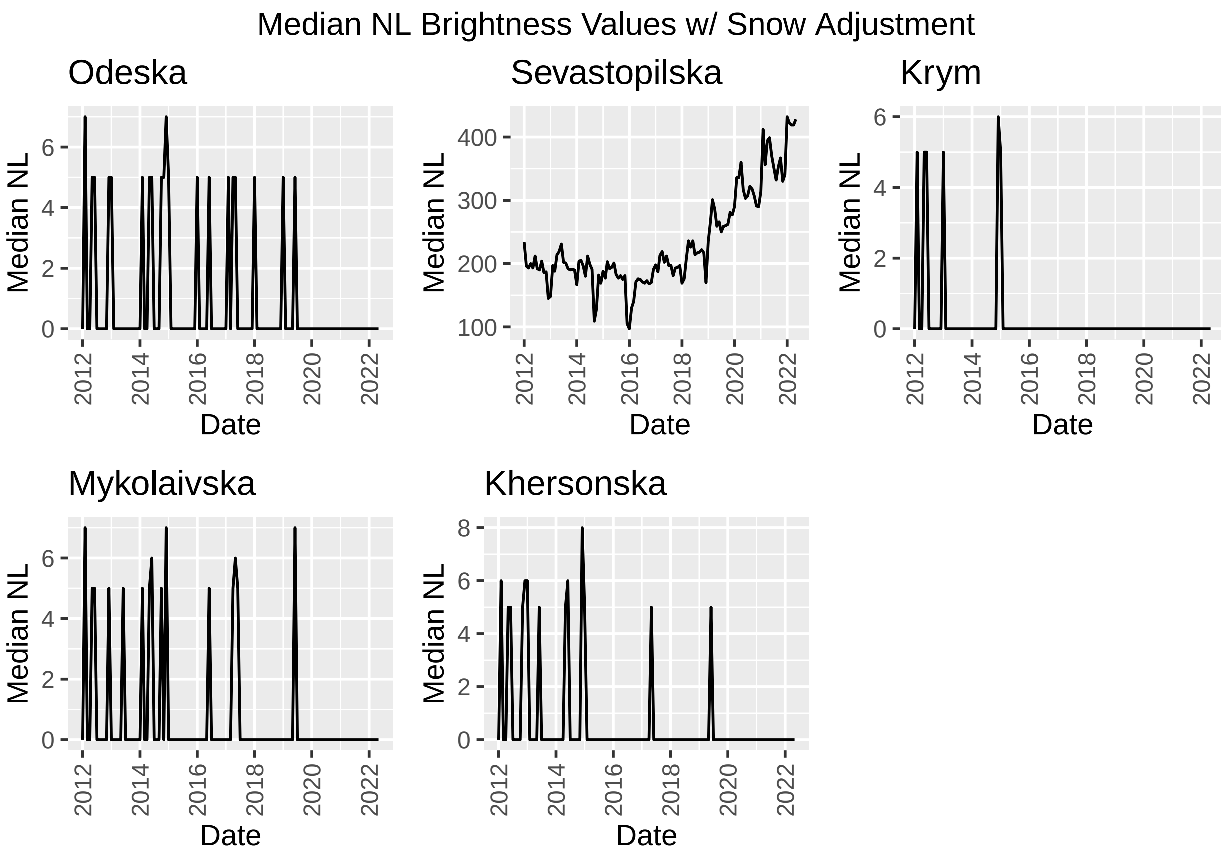
```
med_lum_overtime = ggplot(graph1_data, aes(x = Date, y= nl_median_4, color=name))+ geom_point()+labs(ti

med_lum_overtime
```




```
#mean_lum_overtime = ggplot(graph1_data, aes(x = Date, y= nl_mean_4, color=name))+ geom_point()+labs(ti

#mean_lum_overtime
```

This graph provides an interesting insight as Sevastopilska showed the smallest sum of night lights in the previous visual, yet has the highest median NL values here. This trend is consistent with mean NL data. We will observe these trends on a regional basis below to better understand variations.

```
Odeska_sum = ggplot( data = subset(graph1_data, reg == "UA_51"), aes(x = Date, y= nl_median_4))+ geom_li

Sevastopilska_sum = ggplot( data = subset(graph1_data, reg == "UA_40"), aes(x = Date, y= nl_median_4))+

Krym_sum = ggplot( data = subset(graph1_data, reg == "UA_43"), aes(x = Date, y= nl_median_4))+ geom_line

Mykolaivska_sum = ggplot( data = subset(graph1_data, reg == "UA_48"), aes(x = Date, y= nl_median_4))+ ge

Khersonska_sum = ggplot( data = subset(graph1_data, reg == "UA_65"), aes(x = Date, y= nl_median_4))+ geo

grid.arrange(Odeska_sum,Sevastopilska_sum,Krym_sum,Mykolaivska_sum,Khersonska_sum, ncol=3, top = "Median
```



Odeska, Mykolaivska, Sevastopilska, Khersonska, and Krym all show similar trends with many occurences of median of zero. Sevastopilska has a unique trend of increasing over time and barely having any occurrences of zero.

Look at Median Sum NL per year compared to Labor Index

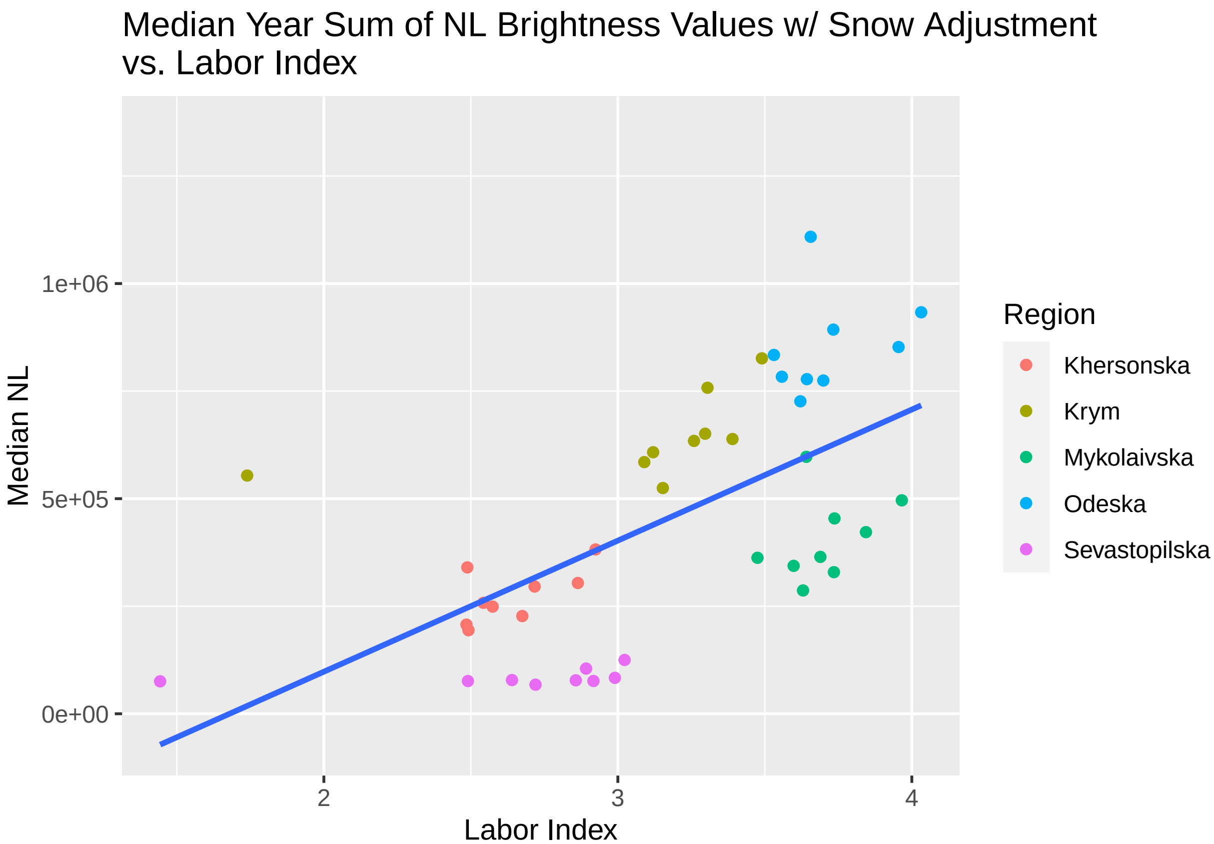
```
agreggated_data = aggregate(cbind(nl_count_0,nl_count_4, nl_min_0, nl_min_4, nl_max_4, nl_mean_4, nl_sum
join_aggregate = left_join(agreggated_data, GT_NL_data, by = c("year"="year", "reg"="reg", "name"="name
join_aggregate = left_join(join_aggregate, GDP_data, by = c("year"="year", "reg"="reg"))
subset_join_agg = subset(join_aggregate, reg == "UA_51"|reg == "UA_40"|reg == "UA_43"|reg == "UA_48"|reg
# here I am combining the data by condensing the Night Light data by taking the median of each value by
```

```
labor_comp_NL = ggplot(subset_join_agg, aes(x = Labor_index, y= nl_sum_4))+ geom_point(aes(color=name))+
vs. Labor Index", y="Median NL", x= "Labor Index") +guides(color = guide_legend(title = "Region"))

labor_comp_NL= labor_comp_NL + geom_smooth(method = "lm", se = FALSE)
labor_comp_NL
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 10 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 10 rows containing missing values (`geom_point()`).
```



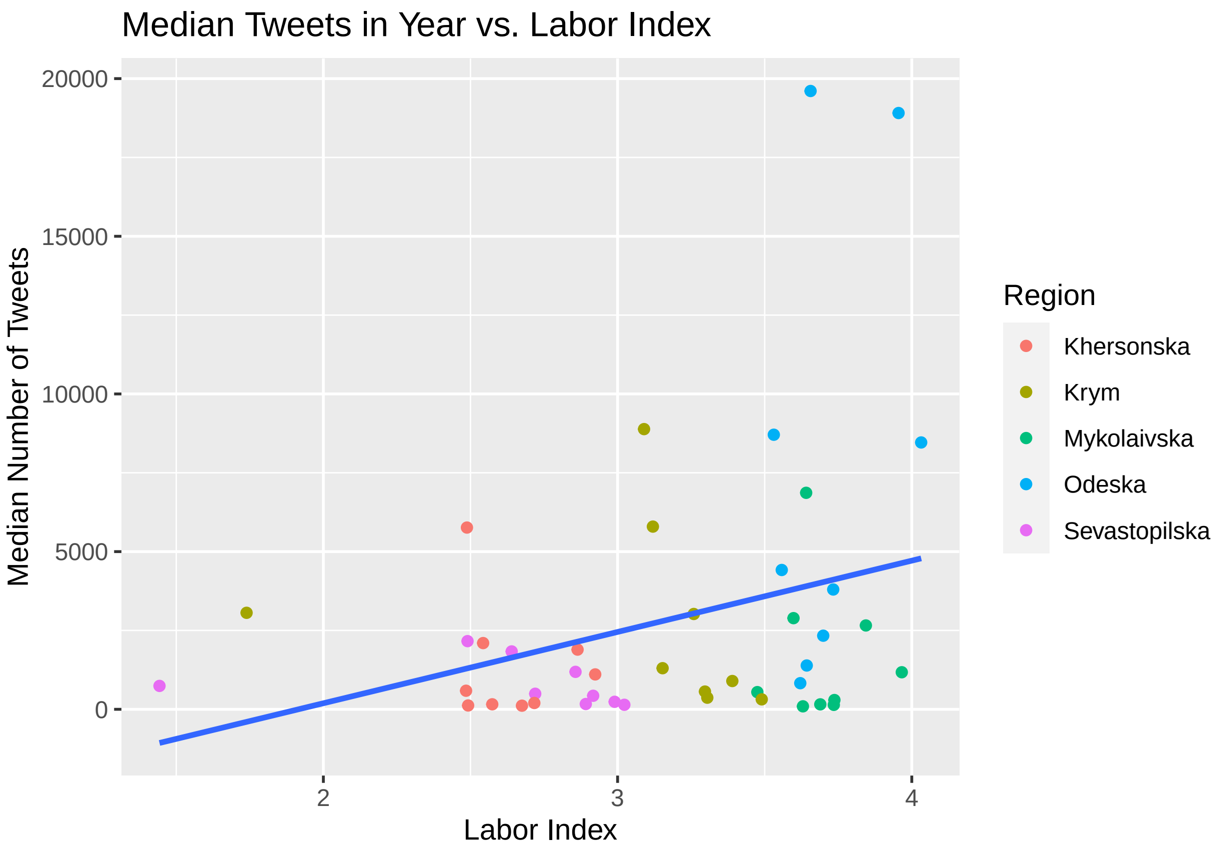
Median Number of Tweets vs. Labor Index

```
labor_comp_tweets = ggplot(subset_join_agg, aes(x = Labor_index, y= tw_count))+ geom_point(aes(color=na

labor_comp_tweets= labor_comp_tweets + geom_smooth(method = "lm", se = FALSE)
labor_comp_tweets
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 10 rows containing non-finite values (`stat_smooth()`).
```

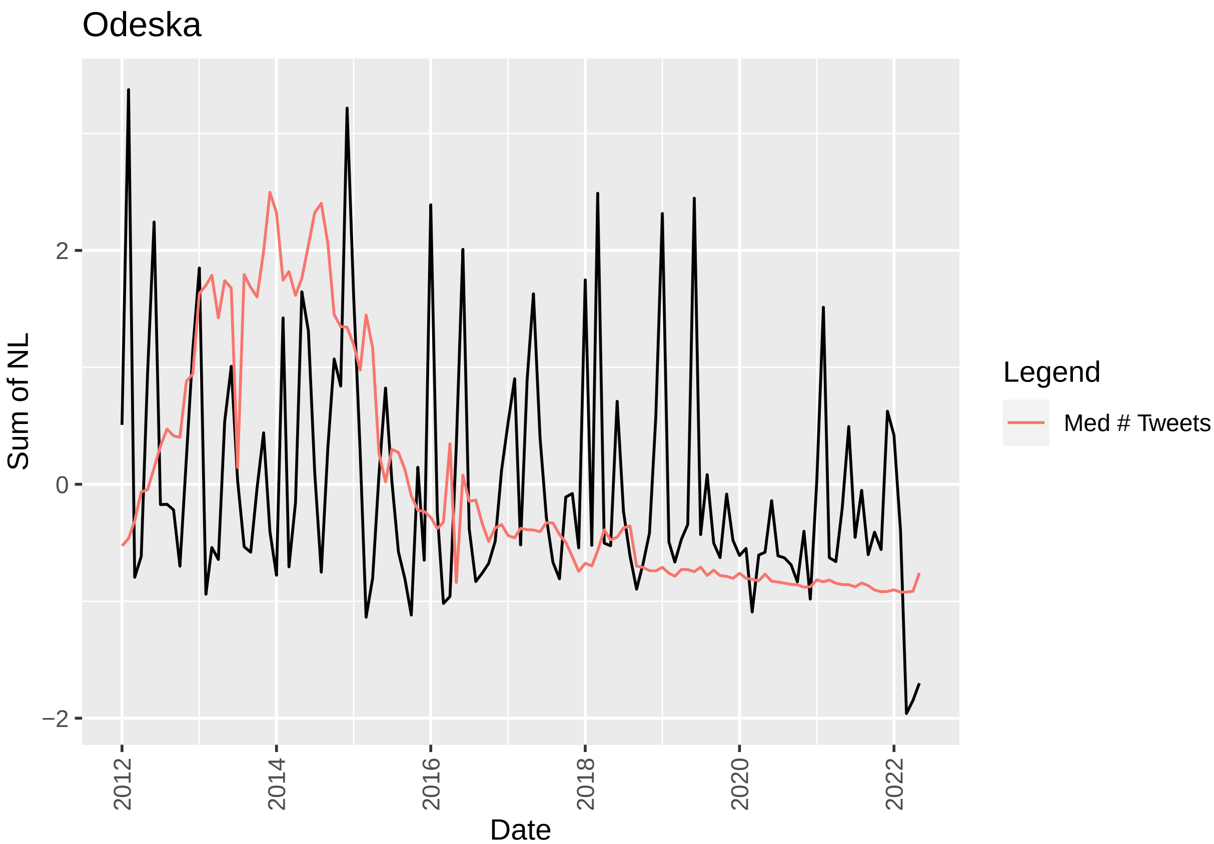
```
## Warning: Removed 10 rows containing missing values (`geom_point()`).
```



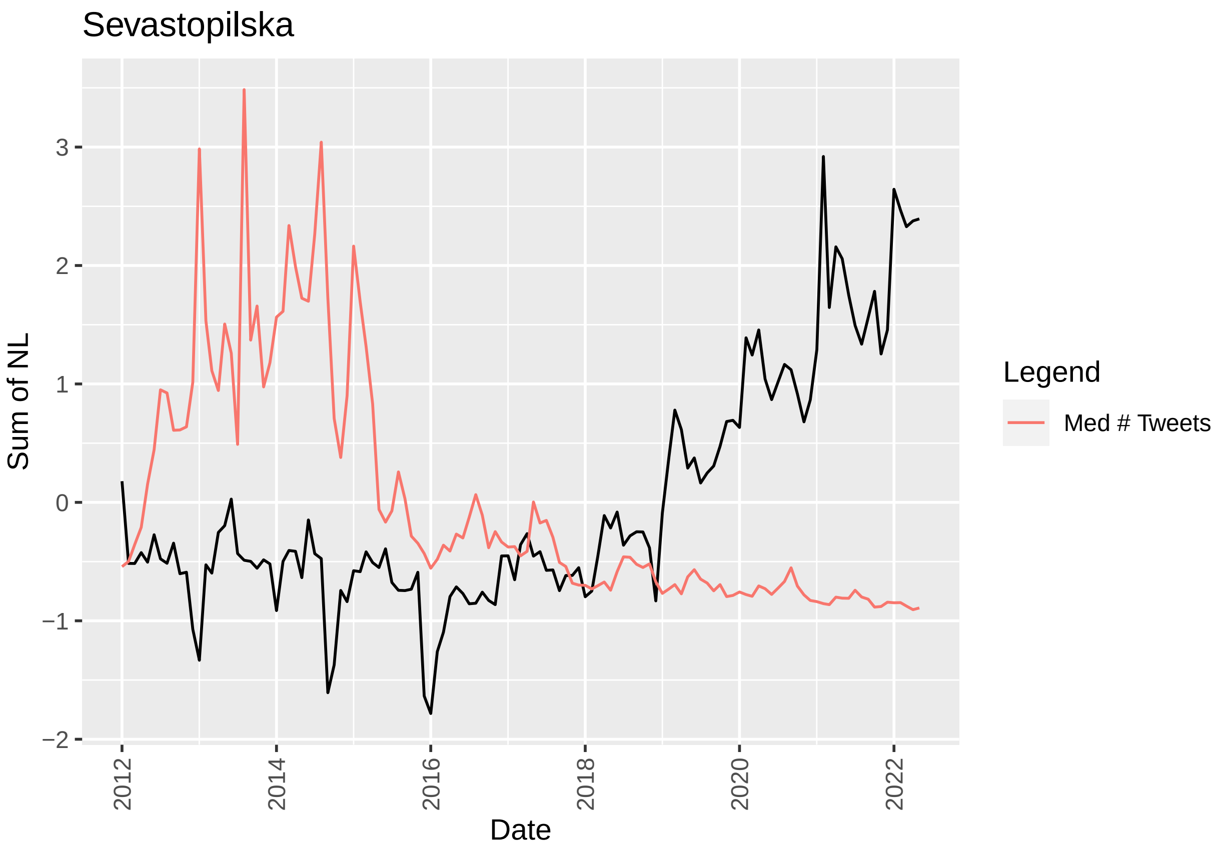
Here we have a visual of the relaionship between the median number of tweets in a year and the labor index. Overall, there appears to be a positive correlation. Clearly there are regional differences, however.

Log median tweets and median sum of night lights in a year

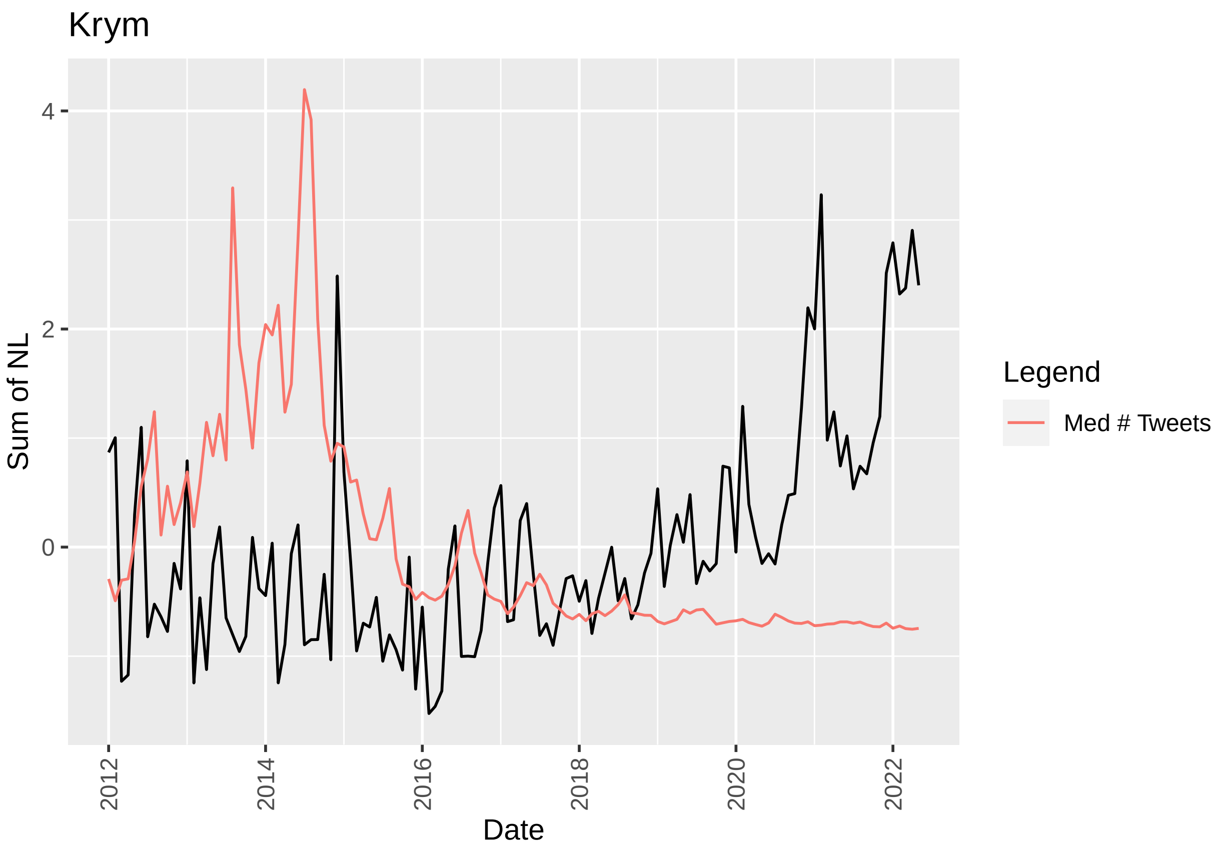
```
Odeska_tweets= ggplot( data = subset(graph1_data, reg == "UA_51"))+ geom_line(aes(x = Date, y= scale(nl_

Sevastopilska_tweets = ggplot( data = subset(graph1_data, reg == "UA_40"), aes(x = Date, y= nl_sum_4))+

Krym_tweets = ggplot( data = subset(graph1_data, reg == "UA_43"), aes(x = Date, y= nl_sum_4))+ geom_line

Mykolaivska_tweets = ggplot( data = subset(graph1_data, reg == "UA_48"), aes(x = Date, y= nl_sum_4))+ ge

Khersonska_tweets = ggplot( data = subset(graph1_data, reg == "UA_65"), aes(x = Date, y= nl_sum_4))+ ge

#grid.arrange(Odeska_tweets,Sevastopilska_tweets,Krym_tweets,Mykolaivska_tweets,Khersonska_tweets, ncol
Odeska_tweets
```



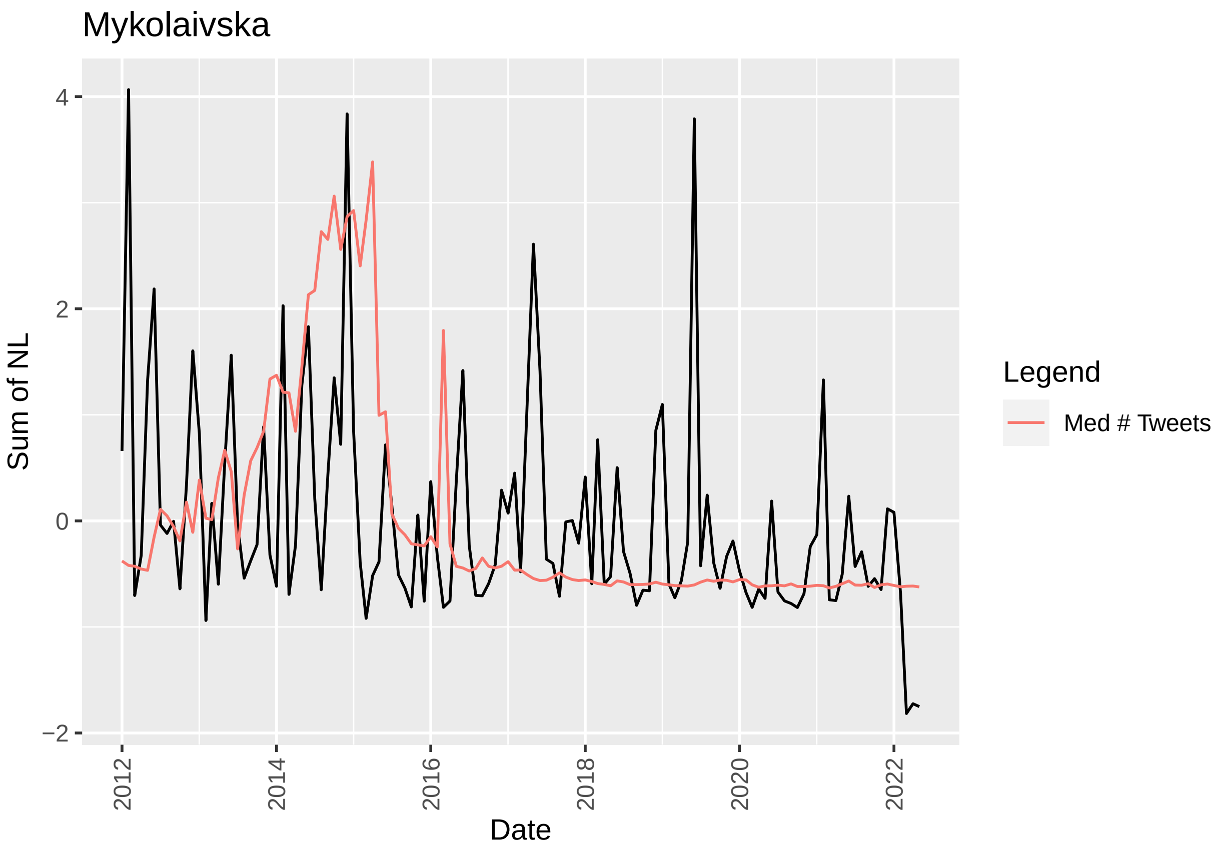

```
Sevastopilska_tweets
```


Sevastopilska
Sum of NL
3
2
1
0
−1
−2
2012
2014
2016
2018
2020
2022
Date
Legend
Med # Tweets

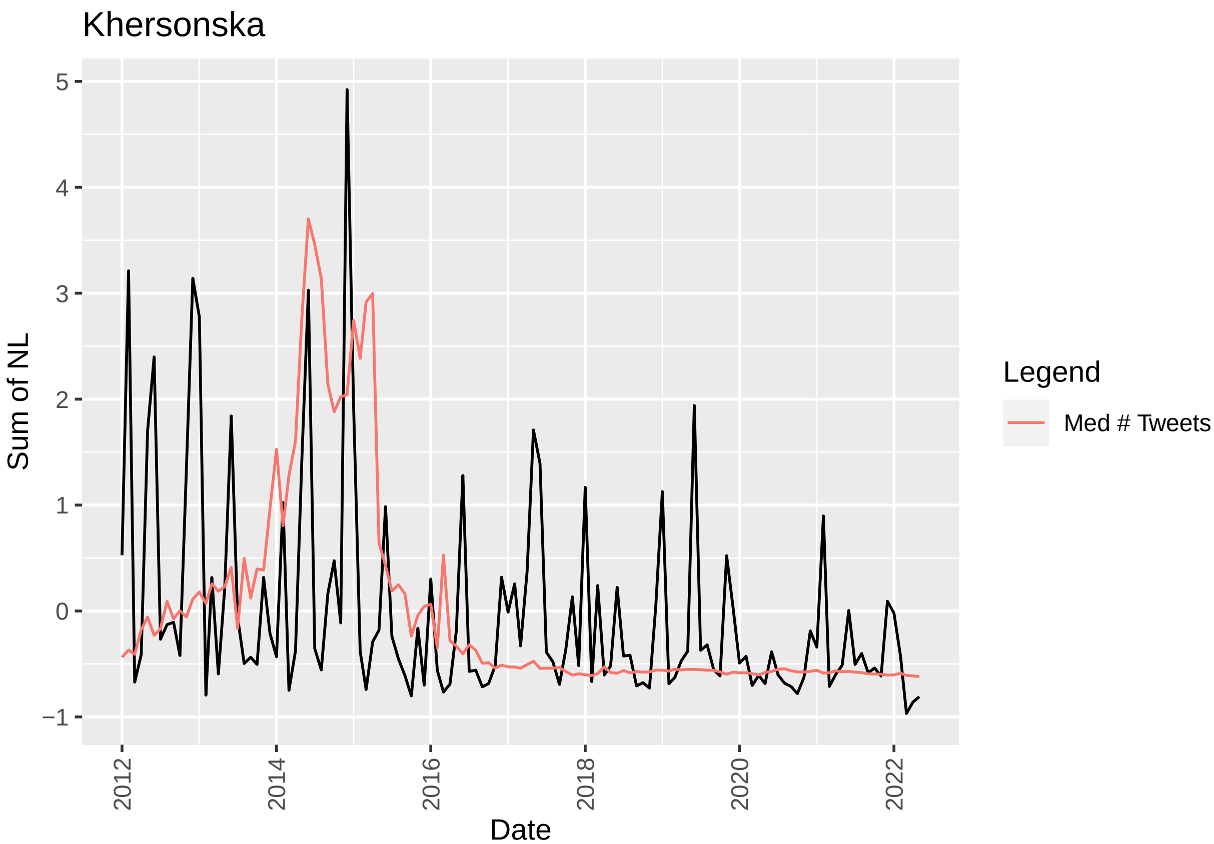
```
Krym_tweets
```




```
Mykolaivska_tweets
```




```
Khersonska_tweets
```



For these visualizations, I have scaled two values(median tweets and median sum of night lights in a year) in order to better understand the relationship between the two. For Regions like Odeska, Krym, and Sevastopilska, there appears to be an inverse relationship with median tweets and median sum of night lights in a year. For Mykolaivska and Khersonska, there is not as clear of relationship.

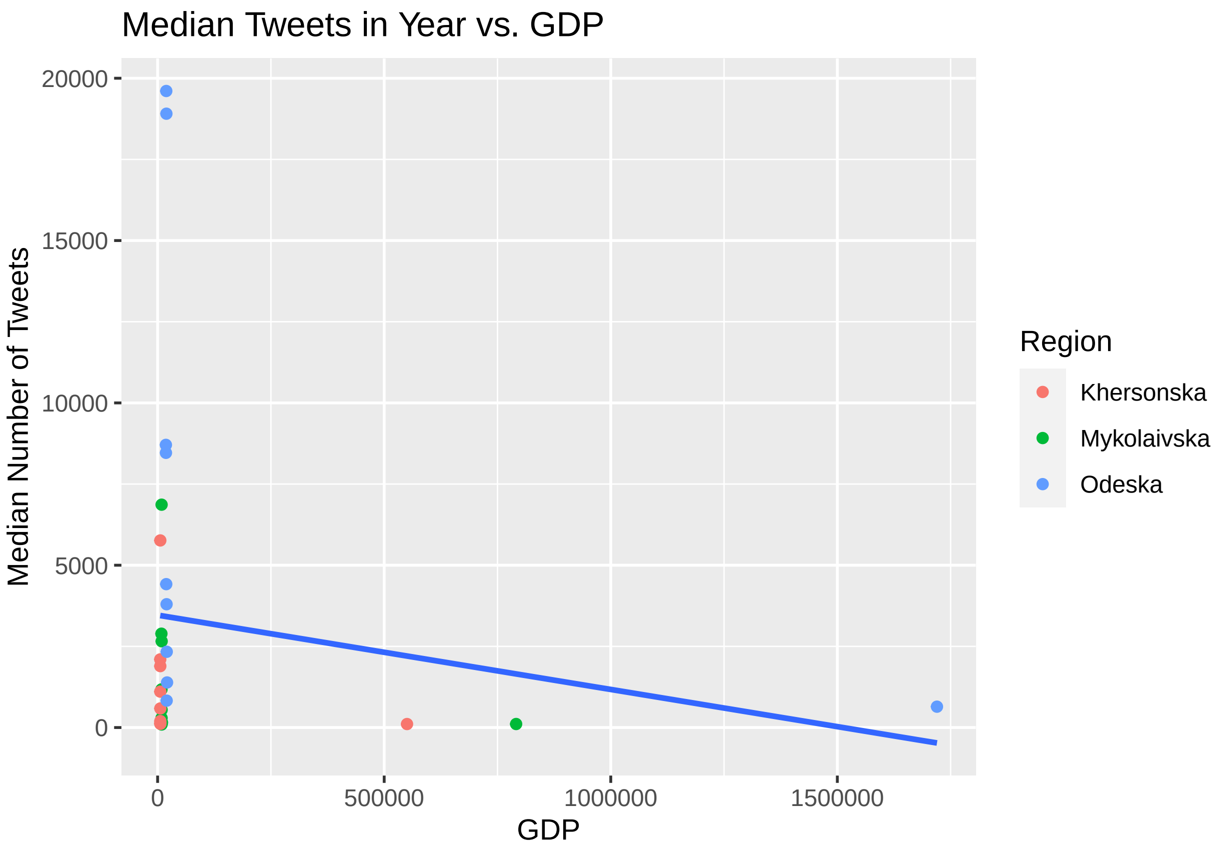
Twitter vs. GDP

```
graph_exc_K_S = subset(subset_join_agg, reg != "UA_40" & reg !="UA_43", !is.na(GDP))
GDP_comp_tweets = ggplot(graph_exc_K_S, aes(x = GDP, y= tw_count))+ geom_point(aes(color=name))+labs(ti

GDP_comp_tweets= GDP_comp_tweets + geom_smooth(method = "lm", se = FALSE)
GDP_comp_tweets
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

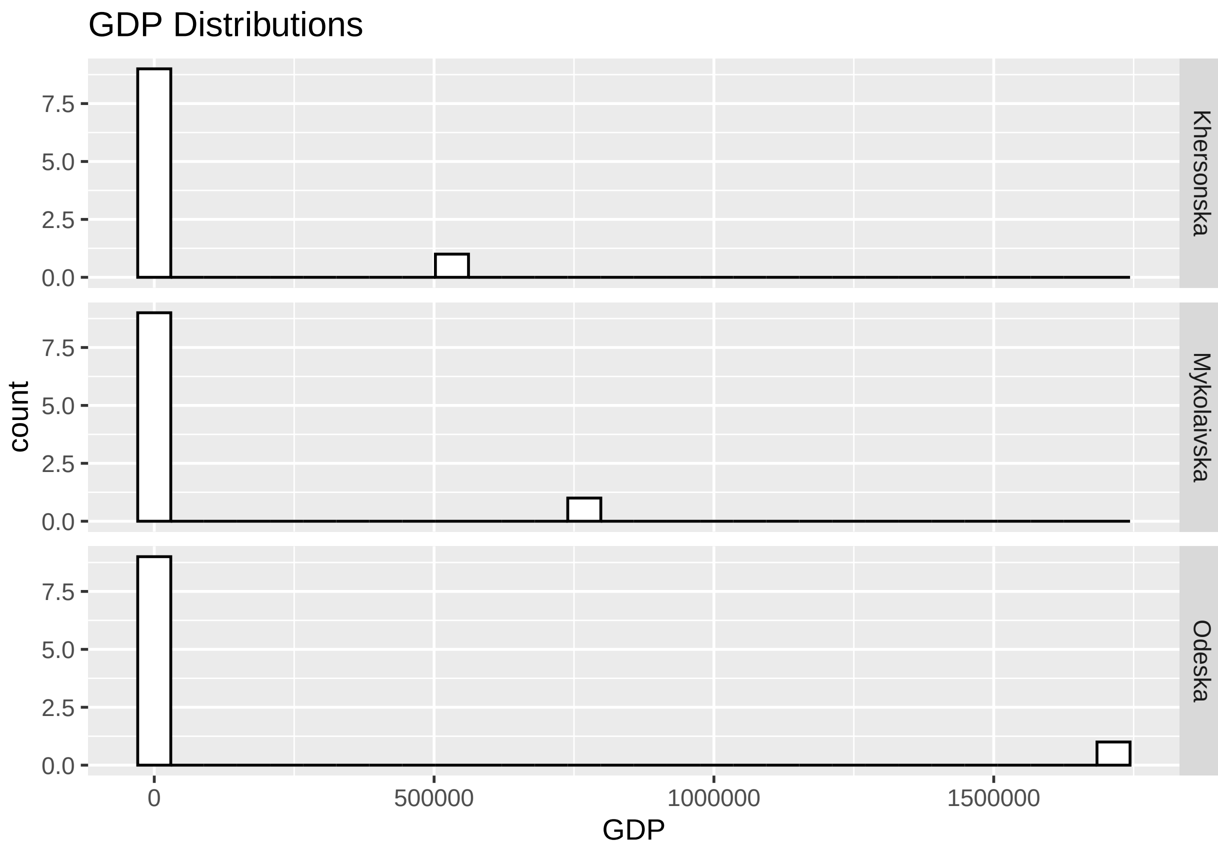
```
## Warning: Removed 3 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 3 rows containing missing values (`geom_point()`).
```




```
GDP_histo = ggplot(graph_exc_K_S, aes(x = GDP)) +
  geom_histogram(fill = "white", colour = "black") +
  facet_grid(name ~ ., scales = "free")+labs(title="GDP Distributions")
GDP_histo
```

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

## Warning: Removed 3 rows containing non-finite values (`stat_bin()`).
```

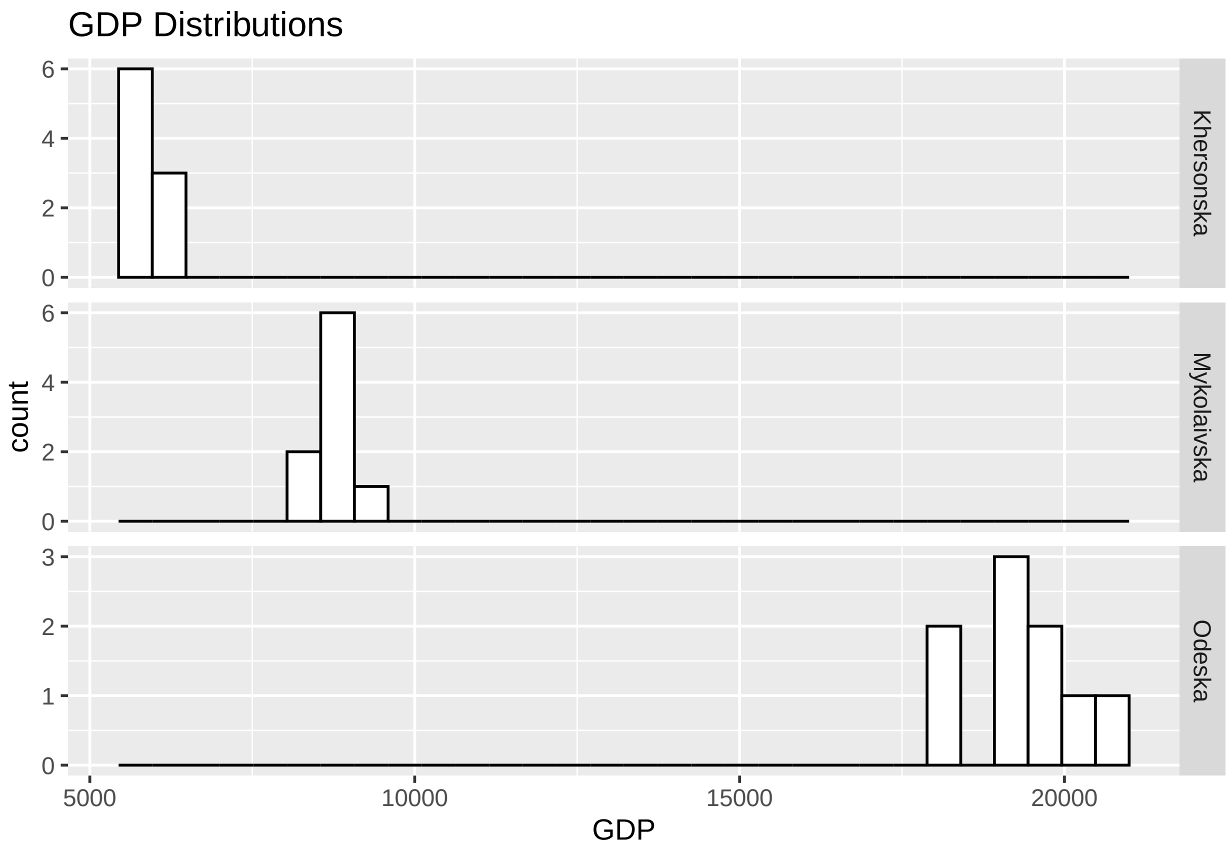



```
look_outlier = subset(subset_join_agg, GDP >250000)
```

These outliers were all collected in 2021. Now we will look at the distribution without the outliers.

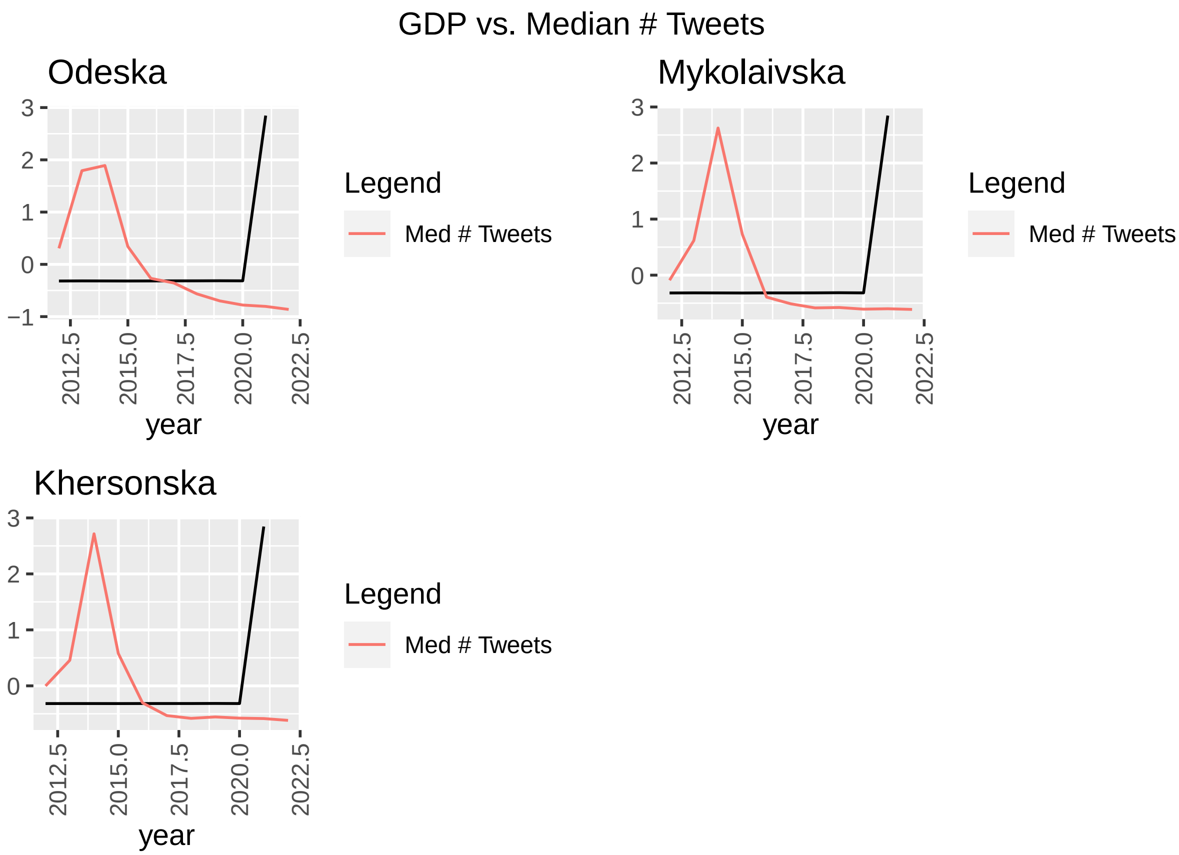
```
GDP_histo1 = ggplot(subset(graph_exc_K_S, GDP<250000), aes(x = GDP)) +
  geom_histogram(fill = "white", colour = "black") +
  facet_grid(name ~ ., scales = "free")+labs(title="GDP Distributions")
GDP_histo1
```

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




```
Odeska_GDP_tweets= ggplot( data = subset(subset_join_agg, reg == "UA_51"))+ geom_line(aes(x = year, y= 

Mykolaivska_GDP_tweets = ggplot( data = subset(subset_join_agg, reg == "UA_48"), aes(x = year, y= GDP))

Khersonska_GDP_tweets = ggplot( data = subset(subset_join_agg, reg == "UA_65"), aes(x = year, y= GDP))+

grid.arrange(Odeska_GDP_tweets, Mykolaivska_GDP_tweets, Khersonska_GDP_tweets, ncol=2, top = "GDP vs. M
```

```
## Warning: Removed 1 row containing missing values (‘geom_line()‘).
## Removed 1 row containing missing values (‘geom_line()‘).
## Removed 1 row containing missing values (‘geom_line()‘).
```

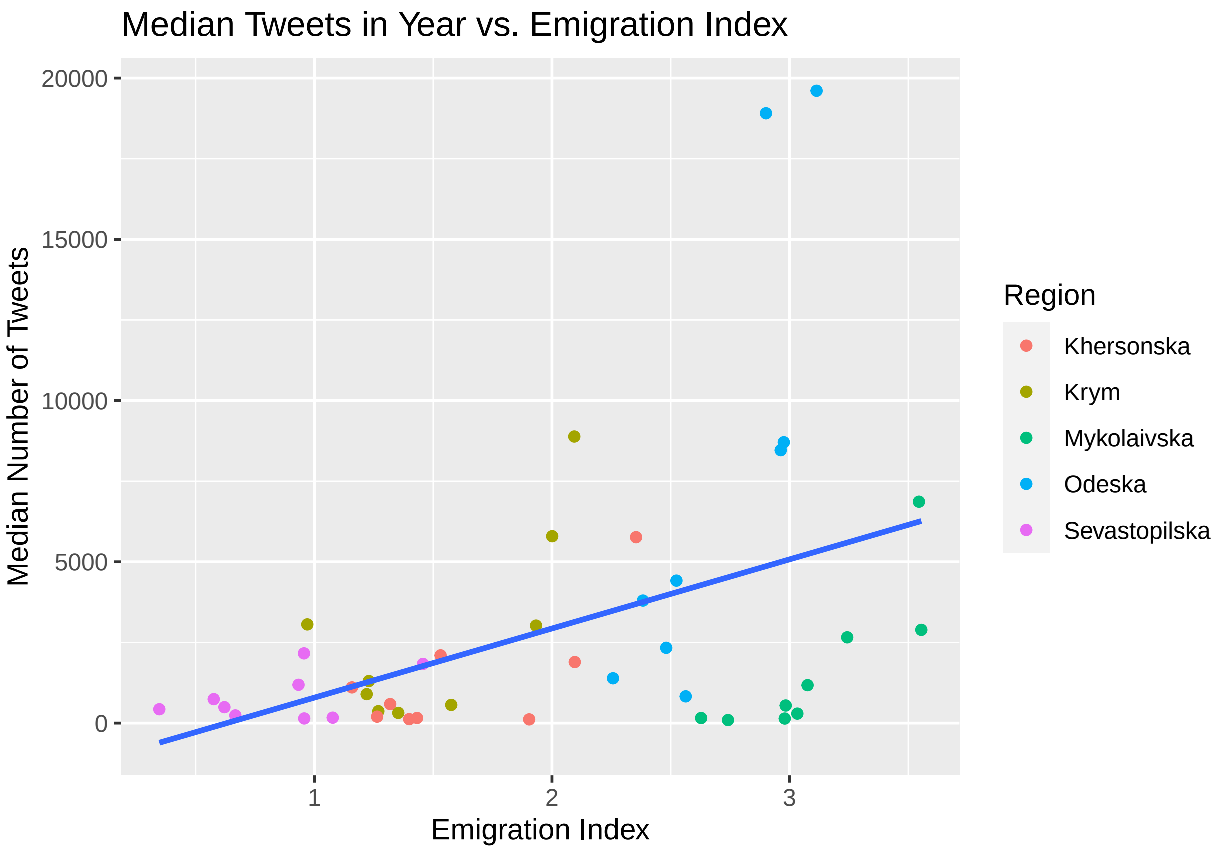


There is a lot of missingness in data for GDP for Sevastopilska and Krym, so these regions were not included in this visualization. Above is the scaled values of Median number of tweets in a year and GDP of the year. A scaled value is used in order to better visualize this relationship.

Tweets vs. Emigration Index

```
emig_comp_tweets = ggplot(subset_join_agg, aes(x = Emigration_index, y= tw_count))+ geom_point(aes(colo

emig_comp_tweets= emig_comp_tweets + geom_smooth(method = "lm", se = FALSE)
emig_comp_tweets
```

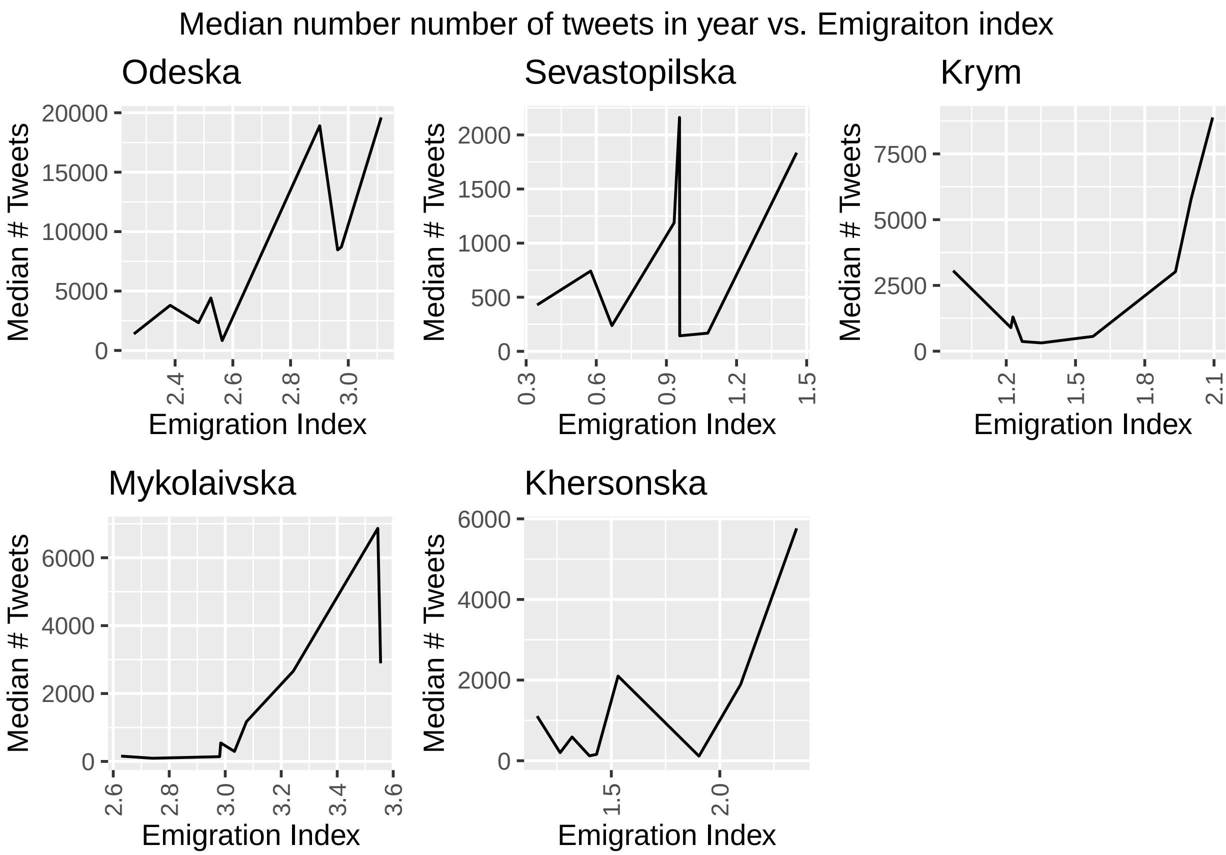
```
## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 10 rows containing non-finite values (`stat_smooth()`).

## Warning: Removed 10 rows containing missing values (`geom_point()`).
```



There is positive relationship with the emigration index and number of tweets and clear trend differences by region.

```
Odeska_Em_tweet = ggplot( data = subset(subset_join_agg, reg == "UA_51"), aes(x = Emigration_index, y= t

Sevastopilska_Em_tweet = ggplot( data = subset(subset_join_agg, reg == "UA_40"), aes(x = Emigration_inde

Krym_Em_tweet = ggplot( data = subset(subset_join_agg, reg == "UA_43"), aes(x = Emigration_index, y= tw

Mykolaivska_Em_tweet = ggplot( data = subset(subset_join_agg, reg == "UA_48"), aes(x = Emigration_index

Khersonska_Em_tweet = ggplot( data = subset(subset_join_agg, reg == "UA_65"), aes(x = Emigration_index,

grid.arrange(Odeska_Em_tweet,Sevastopilska_Em_tweet,Krym_Em_tweet,Mykolaivska_Em_tweet,Khersonska_Em_twe
```

```
## Warning: Removed 2 rows containing missing values (`geom_line()`).
## Removed 2 rows containing missing values (`geom_line()`).
## Removed 2 rows containing missing values (`geom_line()`).
## Removed 2 rows containing missing values (`geom_line()`).
## Removed 2 rows containing missing values (`geom_line()`).
```



The correlation between median number of tweets varies according to region. For many of the regions, there appears to be a positive trend.

Fixed Effect Models

The fixed effect model is a special version of a linear regression model that can capture variation due to endogenous sources. From the EDA, it is clear that year and region differences or associated with variation not explained by the given data. For this reason, I will create a two-way fixed effect model that controls for oblast and time.

```
subset_join_agg = subset_join_agg %>%
  mutate(Log_GDP = log(GDP))
```

```
### Unit FE Model
NL_fe_mod <- plm(Log_GDP ~ NLI+nl_sum_4+nl_median_4+Labor_index+tw_count+MBA_degree_index+Unemployment_
                 data = subset_join_agg,
                 index = c("name", "year"),
                 model = "within",
                 effect="twoways")
summary(NL_fe_mod)
```

```
## Twoways effects Within Model
##
## Call:
## plm(formula = Log_GDP ~ NLI + nl_sum_4 + nl_median_4 + Labor_index +
```

```
##     tw_count + MBA_degree_index + Unemployment_benefits_index +
##     Emigration_index + Mercedes_index + Washing_machine_index,
##     data = subset_join_agg, effect = "twoways", model = "within",
##     index = c("name", "year"))
##
## Unbalanced Panel: n = 5, T = 2-9, N = 31
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -0.0188891 -0.0057785 -0.0017532  0.0053370  0.0188891
##
## Coefficients:
##                               Estimate  Std. Error t-value Pr(>|t|)
## NLI                          8.5915e-02  7.6000e-02  1.1305   0.2910
## nl_sum_4                     8.7567e-08  1.8306e-07  0.4783   0.6452
## nl_median_4                  3.2191e-03  1.0958e-02  0.2938   0.7764
## Labor_index                 -6.8788e-03  1.0015e-01 -0.0687   0.9469
## tw_count                    -9.5136e-07  2.1139e-06 -0.4501   0.6646
## MBA_degree_index             1.4506e-02  1.3176e-02  1.1009   0.3030
## Unemployment_benefits_index -3.5860e-03  1.1875e-02 -0.3020   0.7704
## Emigration_index            -4.0894e-02  1.8772e-02 -2.1785   0.0610 .
## Mercedes_index              -9.7683e-02  8.3345e-02 -1.1720   0.2749
## Washing_machine_index       -2.8090e-02  5.6721e-02 -0.4952   0.6338
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    0.0044912
## Residual Sum of Squares: 0.0017652
## R-Squared:      0.60697
## Adj. R-Squared: -0.47388
## F-statistic: 1.23545 on 10 and 8 DF, p-value: 0.38978
```

```r
tidy(NL_fe_mod)
```

```
## # A tibble: 10 x 5
##    term                           estimate   std.error statistic p.value
##    <chr>                             <dbl>       <dbl>     <dbl>   <dbl>
##  1 NLI                          0.0859      0.0760         1.13    0.291
##  2 nl_sum_4                     0.0000000876 0.000000183    0.478   0.645
##  3 nl_median_4                  0.00322     0.0110         0.294   0.776
##  4 Labor_index                 -0.00688     0.100         -0.0687  0.947
##  5 tw_count                    -0.000000951 0.00000211    -0.450   0.665
##  6 MBA_degree_index             0.0145      0.0132         1.10    0.303
##  7 Unemployment_benefits_index -0.00359     0.0119        -0.302   0.770
##  8 Emigration_index            -0.0409      0.0188        -2.18    0.0610
##  9 Mercedes_index              -0.0977      0.0833        -1.17    0.275
## 10 Washing_machine_index       -0.0281      0.0567        -0.495   0.634
```

```r
coeftest(NL_fe_mod, vcov = vcovHC, type = "HC1")
```

```
##
## t test of coefficients:
##
```

```
##                                Estimate  Std. Error t value  Pr(>|t|)
## NLI                          8.5915e-02  4.9412e-02  1.7388 0.1202700
## nl_sum_4                     8.7567e-08  1.1627e-07  0.7531 0.4729595
## nl_median_4                  3.2191e-03  8.9992e-03  0.3577 0.7298130
## Labor_index                 -6.8788e-03  2.5731e-02 -0.2673 0.7959723
## tw_count                    -9.5136e-07  1.1544e-06 -0.8241 0.4337532
## MBA_degree_index             1.4506e-02  6.0045e-03  2.4158 0.0421234 *
## Unemployment_benefits_index -3.5860e-03  3.8520e-03 -0.9309 0.3791266
## Emigration_index            -4.0894e-02  6.2794e-03 -6.5124 0.0001857 ***
## Mercedes_index              -9.7683e-02  5.6288e-02 -1.7354 0.1208857
## Washing_machine_index       -2.8090e-02  2.6446e-02 -1.0621 0.3191824
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Unit FE == OLS with dummies for Units
NL_lm_mod <- lm(Log_GDP ~ as.factor(year)+name+NLI+nl_sum_4+nl_median_4+Labor_index+tw_count+MBA_degree_
tidy(NL_lm_mod)
```

```
## # A tibble: 23 x 5
##    term                estimate std.error statistic      p.value
##    <chr>                  <dbl>     <dbl>     <dbl>        <dbl>
##  1 (Intercept)          8.93       0.475     18.8   0.0000000667
##  2 as.factor(year)2013  0.0718     0.0320     2.24  0.0551
##  3 as.factor(year)2014  0.0123     0.0727     0.169 0.870
##  4 as.factor(year)2015  0.00908    0.0717     0.127 0.902
##  5 as.factor(year)2016  0.0476     0.0764     0.623 0.551
##  6 as.factor(year)2017  0.112      0.0660     1.70  0.128
##  7 as.factor(year)2018  0.148      0.0838     1.77  0.116
##  8 as.factor(year)2019  0.180      0.0884     2.04  0.0760
##  9 as.factor(year)2020  0.174      0.114      1.53  0.164
## 10 nameKrym             0.662      0.0908     7.29  0.0000846
## # ... with 13 more rows
```

Difference-difference

For the difference-difference analysis, we will do the total change in GDP from the previous year.

```
subset_join_agg_1  = subset(subset_join_agg, reg == "UA_43")
subset_join_agg_1 = subset_join_agg_1 %>%
  arrange(year)
subset_join_agg_1 = subset_join_agg_1 %>%
  mutate(Diff_year = year - lag(year, n=1),  # Difference in time (just in case there are gaps)
         Diff_growth = GDP - lag(GDP,default=first(GDP)), # Difference in route between years
         Rate_percent = (Diff_growth /Diff_year)/ lag(GDP)) # growth rate
```

```
#U_40 Sevastopilska
subset_join_agg_2  = subset(subset_join_agg, reg == "UA_40")
GDP_diff = c(diff(subset_join_agg_2$GDP))
numbers = as.data.frame(GDP_diff)
numbers = numbers %>%
  add_column(year = c("2013", "2014", "2015", "2016", "2017", "2018", "2019", "2020","2021", "2022"))
```

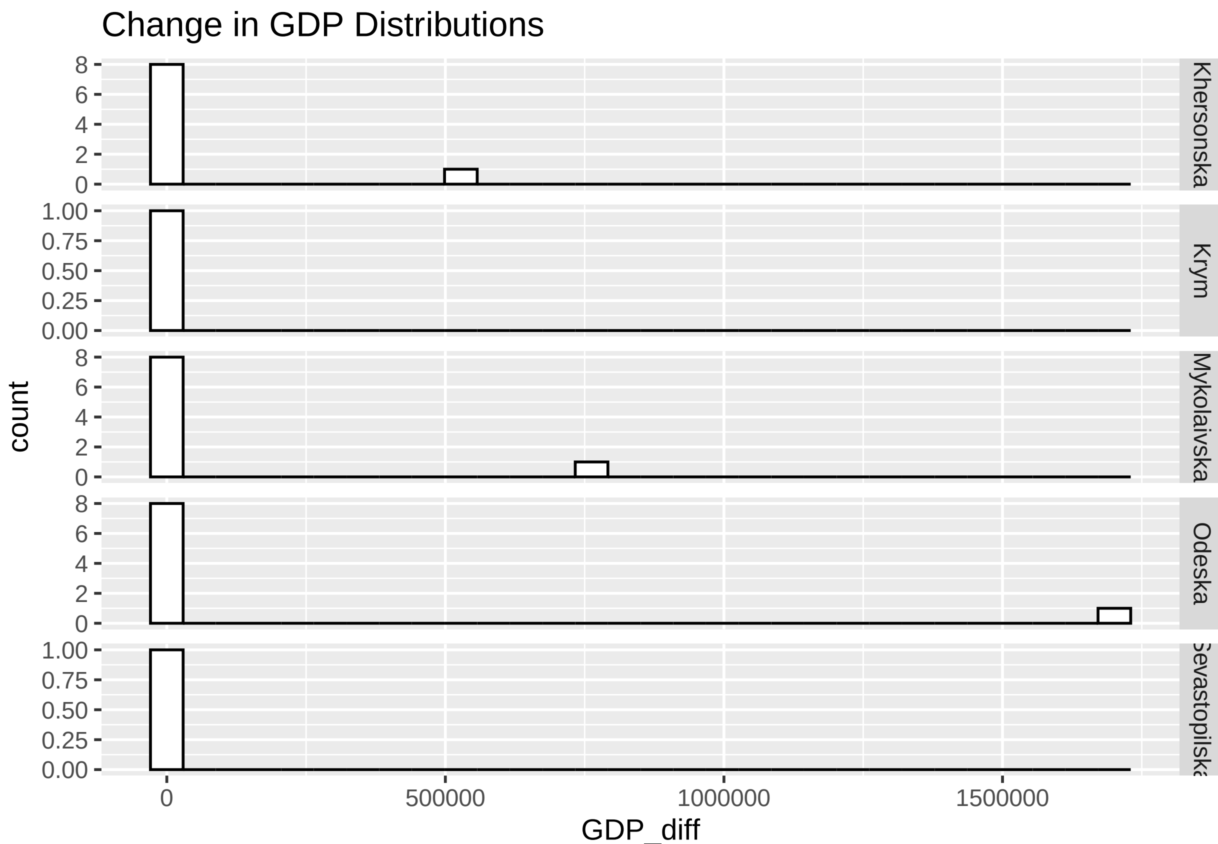
```
numbers$year = as.numeric(numbers$year)
try2 = merge(numbers, subset_join_agg_2, by.x = "year", by.y = "year")


#U_43 Krym
subset_join_agg_1  = subset(subset_join_agg, reg == "UA_43")
GDP_diff = c(diff(subset_join_agg_1$GDP))
numbers = as.data.frame(GDP_diff)
numbers = numbers %>%
  add_column(year = c("2013", "2014", "2015", "2016", "2017", "2018", "2019", "2020","2021", "2022"))
numbers$year = as.numeric(numbers$year)
try1 = merge(numbers, subset_join_agg_1, by.x = "year", by.y = "year")


#U_48 Mykolaivska
subset_join_agg_3  = subset(subset_join_agg, reg == "UA_48")
GDP_diff = c(diff(subset_join_agg_3$GDP))
numbers = as.data.frame(GDP_diff)
numbers = numbers %>%
  add_column(year = c("2013", "2014", "2015", "2016", "2017", "2018", "2019", "2020","2021", "2022"))
numbers$year = as.numeric(numbers$year)
try3 = merge(numbers, subset_join_agg_3, by.x = "year", by.y = "year")

#U_51 Odeska
subset_join_agg_4  = subset(subset_join_agg, reg == "UA_51")
GDP_diff = c(diff(subset_join_agg_4$GDP))
numbers = as.data.frame(GDP_diff)
numbers = numbers %>%
  add_column(year = c("2013", "2014", "2015", "2016", "2017", "2018", "2019", "2020","2021", "2022"))
numbers$year = as.numeric(numbers$year)
try4 = merge(numbers, subset_join_agg_4, by.x = "year", by.y = "year")


#U_65 Khersonska
subset_join_agg_5  = subset(subset_join_agg, reg == "UA_65")
GDP_diff = c(diff(subset_join_agg_5$GDP))
numbers = as.data.frame(GDP_diff)
numbers = numbers %>%
  add_column(year = c("2013", "2014", "2015", "2016", "2017", "2018", "2019", "2020","2021", "2022"))
numbers$year = as.numeric(numbers$year)
try5 = merge(numbers, subset_join_agg_5, by.x = "year", by.y = "year")
```

```
change_data=rbind(try1, try2, try3, try4, try5)
```

```
GDP_diff_histo = ggplot(change_data, aes(x = GDP_diff)) +
  geom_histogram(fill = "white", colour = "black") +
  facet_grid(name ~ ., scales = "free")+labs(title="Change in GDP Distributions")
GDP_diff_histo
```

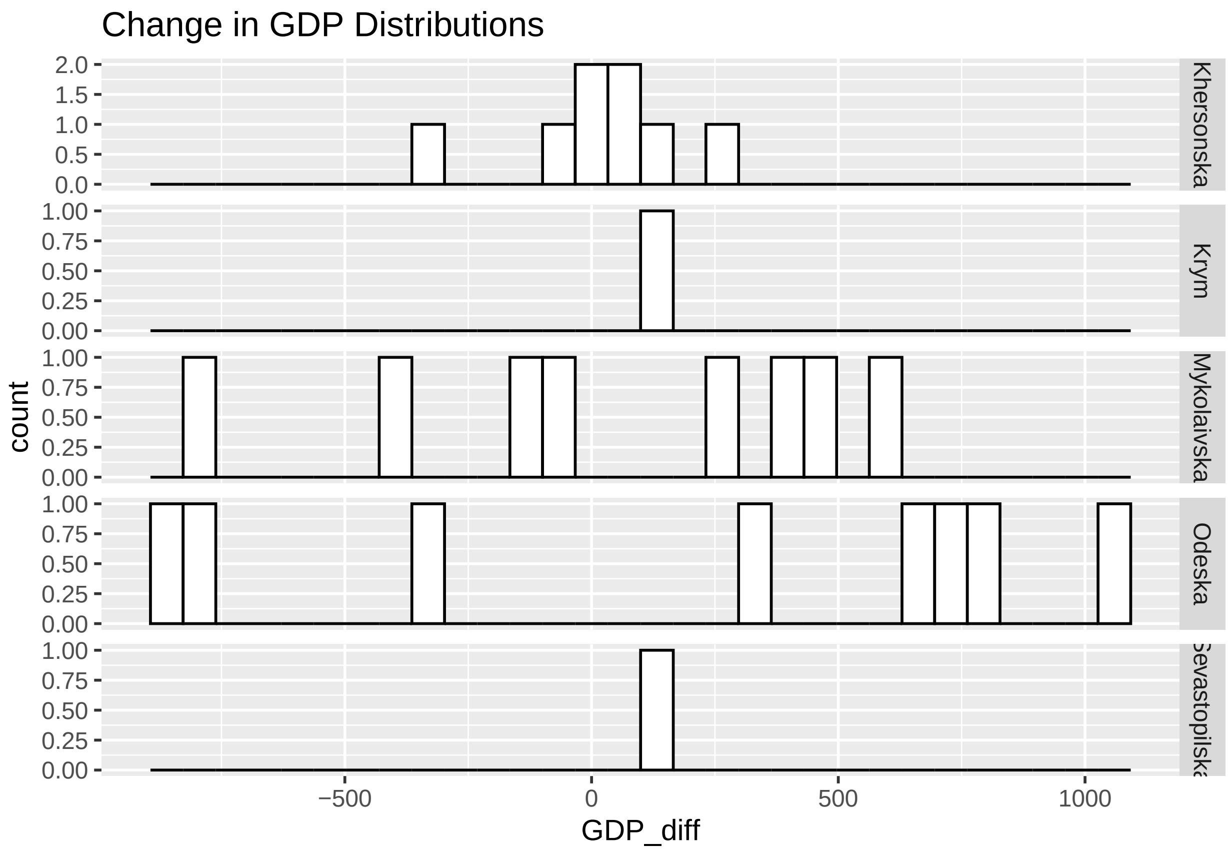
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 21 rows containing non-finite values (`stat_bin()`).
```



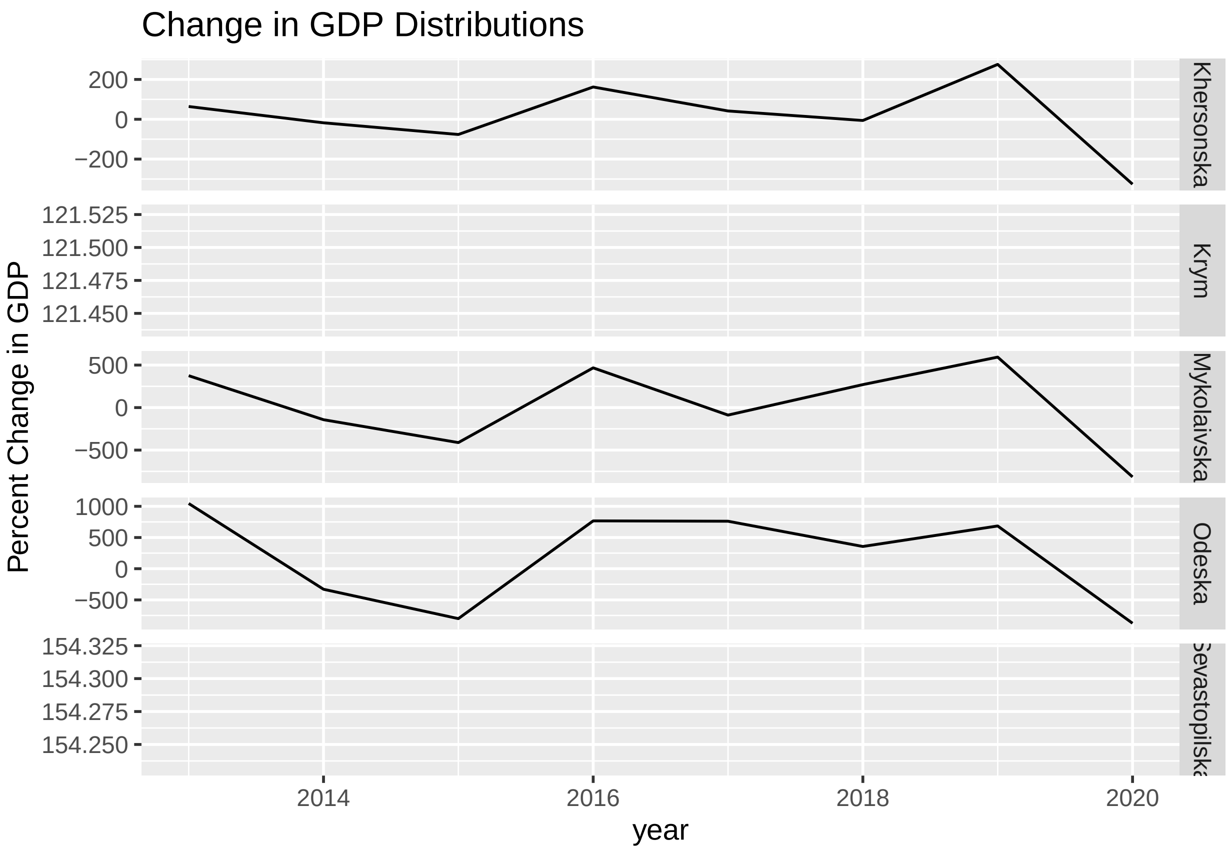

```r
GDP_diff_histo1 = ggplot(subset(change_data, GDP_diff<250000), aes(x = GDP_diff)) +
  geom_histogram(fill = "white", colour = "black") +
  facet_grid(name ~ ., scales = "free")+labs(title="Change in GDP Distributions")
GDP_diff_histo1
```

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




```
GDP_diff_line1 = ggplot(subset(change_data, GDP_diff<250000), aes(y = GDP_diff, x =year)) +
  geom_line() +
  facet_grid(name ~ ., scales = "free")+labs(title="Change in GDP Distributions", y = "Percent Change in
GDP_diff_line1
```

```
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```



Run Model with Total Change in GDP

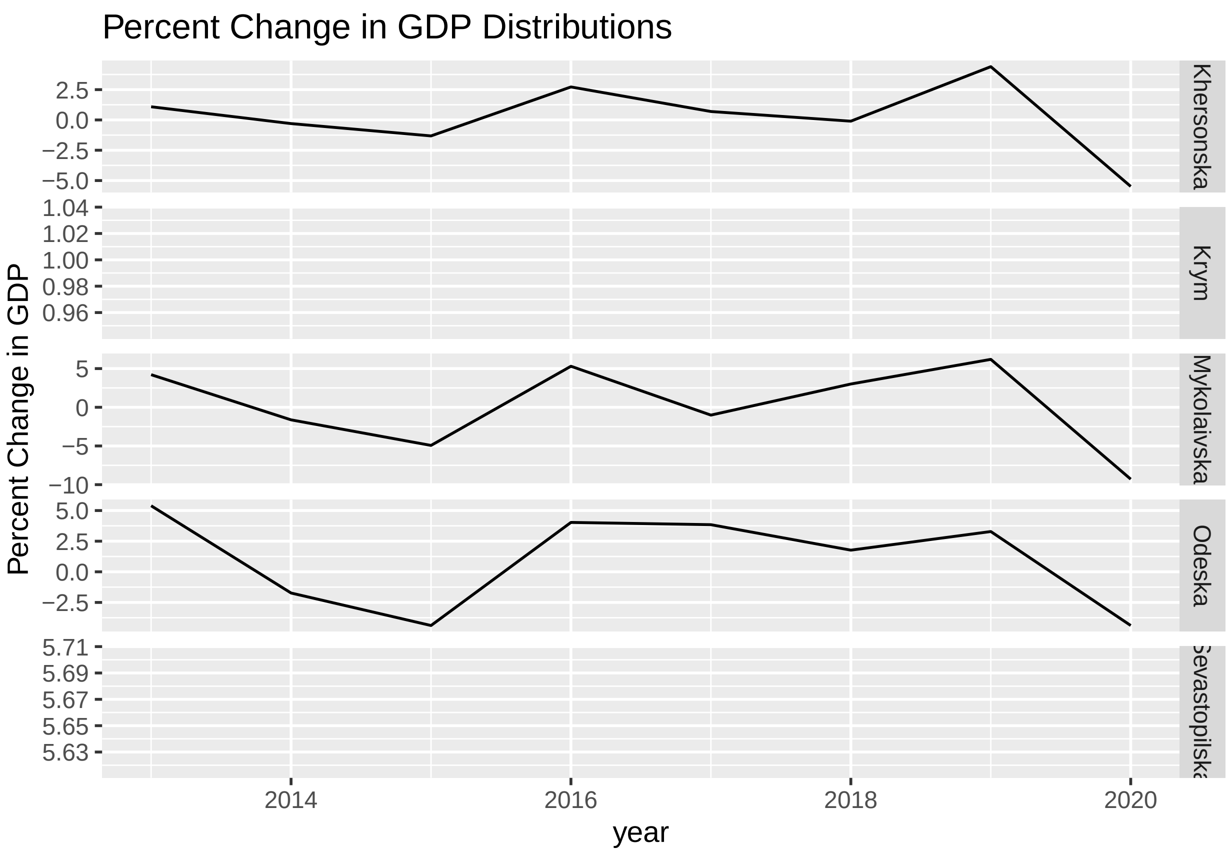
```
# Unit FE == OLS with dummies for Units
change_data = subset(change_data, GDP_diff<250000)
new_model <- lm(GDP_diff ~ as.factor(year)+name+NLI+nl_sum_4+nl_median_4+Labor_index+tw_count+MBA_degree
tidy(new_model)
```

```
## # A tibble: 22 x 5
##    term                estimate std.error statistic p.value
##    <chr>                  <dbl>     <dbl>     <dbl>   <dbl>
##  1 (Intercept)             731.     7649.    0.0956  0.928
##  2 as.factor(year)2014    -720.     1189.   -0.606   0.577
##  3 as.factor(year)2015   -2203.      813.   -2.71    0.0536
##  4 as.factor(year)2016   -2188.     1006.   -2.17    0.0953
##  5 as.factor(year)2017   -3087.     1242.   -2.49    0.0678
##  6 as.factor(year)2018   -3422.     1410.   -2.43    0.0722
##  7 as.factor(year)2019   -4276.     1717.   -2.49    0.0675
##  8 as.factor(year)2020   -4351.     1816.   -2.40    0.0747
##  9 nameKrym               -812.     1760.   -0.461   0.669
## 10 nameMykolaivska        5934.     2864.    2.07    0.107
## # ... with 12 more rows
```

2015, 201620172018, 2019, 2020, Unemployment Benefits Index, Emigration Index # Run Model with Percent Change GDP

```r
change_data = change_data %>%
  mutate(Perc_change = 100*GDP_diff/GDP)

GDP_diff_line = ggplot(change_data, aes(y = Perc_change, x =year)) +
  geom_line() +
  facet_grid(name ~ ., scales = "free")+labs(title="Percent Change in GDP Distributions", y = "Percent C
GDP_diff_line
```

```
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```




```r
# Unit FE == OLS with dummies for Units
new_model2 <- lm(Perc_change ~ as.factor(year)+name+NLI+nl_sum_4+nl_median_4+Labor_index+tw_count+MBA_de
tidy(new_model2)
```

```
## # A tibble: 22 x 5
##    term                estimate std.error statistic p.value
##    <chr>                  <dbl>     <dbl>     <dbl>   <dbl>
##  1 (Intercept)             1.75     49.5     0.0353  0.974
##  2 as.factor(year)2014   -19.1       7.70   -2.49    0.0677
##  3 as.factor(year)2015   -12.0       5.26   -2.28    0.0848
##  4 as.factor(year)2016    -5.76      6.51   -0.885   0.426
##  5 as.factor(year)2017    -9.87      8.04   -1.23    0.287
```

```
##  6 as.factor(year)2018    -6.59      9.13   -0.722   0.510
##  7 as.factor(year)2019    -9.90     11.1    -0.890   0.424
##  8 as.factor(year)2020    -6.26     11.8    -0.533   0.622
##  9 nameKrym              -28.9      11.4    -2.53    0.0644
## 10 nameMykolaivska        16.2      18.5     0.875   0.431
## # ... with 12 more rows
```

Significant in Level-Level Model (alpha = 0.1)

- Intercept
- Krym
- Odeska
- 2013
- 2019
- Emigration Index

Significant in Difference-difference Model Total Change GDP (alpha = 0.1)

- 2015
- 2016
- 2017
- 2018
- 2019
- 2020
- Unemployment Benefits Index
- Emigration Index

Significant in Difference-difference Model Percent Change GDP (alpha = 0.1)

- 2014
- 2015
- Krym
- MBA Degree Index
- Emigration Index

```
#
# subset_join_agg_try = subset_join_agg %>%
#     group_by(reg) %>%
#     arrange(year, .by_group = TRUE) %>%
#     mutate(diff_year = year - lag(year))
#
#
# try = subset_join_agg %>%
#     group_by(reg) %>%
#     mutate(diff = GDP - lag(GDP, order_by = year))
```