R notebook

The following is your first chunk to start with. Remember, you can add chunks using the menu above (Insert -> R) or using the keyboard shortcut Ctrl+Alt+I. A good practice is to use different code chunks to answer different questions. You can delete this comment if you like.

Other useful keyboard shortcuts include Alt- for the assignment operator, and Ctrl+Shift+M for the pipe operator. You can delete these reminders if you don't want them in your report.

```r
#setwd("...")

library("tidyverse")
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.5
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

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

```r
library("tidymodels")
```

```
## -- Attaching packages -------------------------------------- tidymodels 0.1.0 --
```

```
## v broom     0.5.5     v rsample   0.0.5
## v dials     0.0.4     v tune      0.0.1
## v infer     0.5.1     v workflows 0.1.1
## v parsnip   0.0.5     v yardstick 0.0.4
## v recipes   0.1.9
```

```
## -- Conflicts ----------------------------------------- tidymodels_conflicts() --
## x scales::discard()   masks purrr::discard()
## x dplyr::filter()     masks stats::filter()
## x recipes::fixed()    masks stringr::fixed()
## x dplyr::lag()        masks stats::lag()
## x dials::margin()     masks ggplot2::margin()
## x yardstick::spec()   masks readr::spec()
## x recipes::step()     masks stats::step()
## x recipes::yj_trans() masks scales::yj_trans()
```

```r
library("plotly")
```

```
## 
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
## 
##     last_plot
```

```
## The following object is masked from 'package:stats':
```

```
## 
##     filter

## The following object is masked from 'package:graphics':
## 
##     layout
```

```r
library("skimr")
library("lubridate")
```

```
## 
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
## 
##     date
```

```r
library("tsibble")
```

```
## 
## Attaching package: 'tsibble'

## The following objects are masked from 'package:lubridate':
## 
##     interval, new_interval

## The following object is masked from 'package:dplyr':
## 
##     id
```

```r
library("fpp3")
```

```
## -- Attaching packages ---------------------------------------------- fpp3 0.2 --

## v tsibbledata 0.1.0     v fable       0.1.2
## v feasts      0.1.3

## -- Conflicts ------------------------------------------------- fpp3_conflicts --
## x fabletools::accuracy()   masks yardstick::accuracy()
## x lubridate::date()        masks base::date()
## x scales::discard()        masks purrr::discard()
## x plotly::filter()         masks dplyr::filter(), stats::filter()
## x fabletools::generate()   masks infer::generate()
## x tsibble::id()            masks dplyr::id()
## x tsibble::interval()      masks lubridate::interval()
## x dplyr::lag()             masks stats::lag()
## x dials::margin()          masks ggplot2::margin()
## x tsibble::new_interval()  masks lubridate::new_interval()
## x fabletools::null_model() masks parsnip::null_model()
```

```r
library("fable")
library("tibbletime")
```

```
## 
## Attaching package: 'tibbletime'

## The following object is masked from 'package:stats':
## 
##     filter
```

```
library("forecast")
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

##
## Attaching package: 'forecast'

## The following objects are masked from 'package:fabletools':
##
##     GeomForecast, StatForecast

## The following object is masked from 'package:yardstick':
##
##     accuracy
```

Load the Lending Club dataset

Q1a

```
tsLCOrg <-
  read_csv("lendingClub.csv")
```

```
## Parsed with column specification:
## cols(
##   date = col_date(format = ""),
##   state = col_character(),
##   avgLoans = col_double(),
##   totalLoans = col_double(),
##   avgTerm = col_double(),
##   avgIntRate = col_double(),
##   avgGrade = col_double(),
##   avgEmpLength = col_double(),
##   avgAnnualInc = col_double(),
##   avgVerifStatus = col_double(),
##   avgHomeOwner = col_double(),
##   avgOpenAcc = col_double(),
##   avgRevolBal = col_double(),
##   avgRevolUtil = col_double(),
##   avgTotalAcc = col_double(),
##   countOfLoans = col_double()
## )
```

```
tsLCOrg
```

```
## # A tibble: 4,943 x 16
##    date       state avgLoans totalLoans avgTerm avgIntRate avgGrade avgEmpLength
##    <date>     <chr>    <dbl>      <dbl>   <dbl>      <dbl>    <dbl>        <dbl>
##  1 2008-01-01 AK        5600       5600      36       18.0      7            5
##  2 2008-03-01 AK       11700      23400      36       11.8      3            3.5
##  3 2008-06-01 AK        7500       7500      36       13.9      4            3
##  4 2008-12-01 AK       25000      25000      36       15.2      5            1
##  5 2009-01-01 AK       15000      30000      36       12.5      2.5          7
##  6 2009-03-01 AK       14662.     29325      36       13        3            7
##  7 2009-04-01 AK       20000      20000      36       11.9      2            5
##  8 2009-05-01 AK       16000      16000      36       12.2      2            2
##  9 2009-07-01 AK       1000        1000      36      11.9       2           10
## 10 2009-11-01 AK      11000       11000      36       8.94      1            7
## # ... with 4,933 more rows, and 8 more variables: avgAnnualInc <dbl>,
## #   avgVerifStatus <dbl>, avgHomeOwner <dbl>, avgOpenAcc <dbl>,
## #   avgRevolBal <dbl>, avgRevolUtil <dbl>, avgTotalAcc <dbl>,
## #   countOfLoans <dbl>
```

```
unique(tsLCOrg$state)
```

```
##  [1] "AK" "AL" "AR" "AZ" "CA" "CO" "CT" "DC" "DE" "FL" "GA" "HI" "IA" "ID" "IL"
## [16] "IN" "KS" "KY" "LA" "MA" "MD" "ME" "MI" "MN" "MO" "MS" "MT" "NC" "ND" "NE"
## [31] "NH" "NJ" "NM" "NV" "NY" "OH" "OK" "OR" "PA" "RI" "SC" "SD" "TN" "TX" "UT"
## [46] "VA" "VT" "WA" "WI" "WV" "WY"
```

Convert tsLCOrg to tsibble using state as key and monthly_date and index

Q1b

```
tsLCOrg <-
  tsLCOrg %>%
  mutate(monthly_date = yearmonth(date))%>%
  as_tsibble(key = state, index = monthly_date)
```

```
tsLCOrg
```

```
## # A tsibble: 4,943 x 17 [1M]
## # Key:       state [51]
##    date       state avgLoans totalLoans avgTerm avgIntRate avgGrade avgEmpLength
##    <date>     <chr>    <dbl>      <dbl>   <dbl>      <dbl>    <dbl>        <dbl>
##  1 2008-01-01 AK       5600        5600      36      18.0       7            5
##  2 2008-03-01 AK      11700       23400      36      11.8       3            3.5
##  3 2008-06-01 AK       7500        7500      36      13.9       4            3
##  4 2008-12-01 AK      25000       25000      36      15.2       5            1
##  5 2009-01-01 AK      15000       30000      36      12.5       2.5          7
##  6 2009-03-01 AK      14662.      29325      36      13         3            7
##  7 2009-04-01 AK      20000       20000      36      11.9       2            5
##  8 2009-05-01 AK      16000       16000      36      12.2       2            2
##  9 2009-07-01 AK       1000        1000      36      11.9       2           10
## 10 2009-11-01 AK      11000       11000      36       8.94      1            7
## # ... with 4,933 more rows, and 9 more variables: avgAnnualInc <dbl>,
## #   avgVerifStatus <dbl>, avgHomeOwner <dbl>, avgOpenAcc <dbl>,
## #   avgRevolBal <dbl>, avgRevolUtil <dbl>, avgTotalAcc <dbl>,
## #   countOfLoans <dbl>, monthly_date <mth>
```

Q1c

```
summary(tsLCOrg)
```

```
##       date               state             avgLoans       totalLoans
##  Min.   :2007-06-01   Length:4943        Min.   :  500   Min.   :      500
##  1st Qu.:2010-06-01   Class :character   1st Qu.:10583   1st Qu.:   115688
##  Median :2012-11-01   Mode  :character   Median :13704   Median :   927825
##  Mean   :2012-09-13                      Mean   :12607   Mean   :  4234850
##  3rd Qu.:2015-02-01                      3rd Qu.:14954   3rd Qu.:  4303438
##  Max.   :2017-03-01                      Max.   :29975   Max.   :126477500
##
##     avgTerm        avgIntRate       avgGrade      avgEmpLength
##  Min.   :36.00   Min.   : 6.03   Min.   :1.000   Min.   : 1.000  
##  1st Qu.:36.00   1st Qu.:12.17   1st Qu.:2.571   1st Qu.: 5.026  
##  Median :42.13   Median :12.96   Median :2.750   Median : 5.989  
##  Mean   :41.38   Mean   :12.92   Mean   :2.769   Mean   : 5.569  
##  3rd Qu.:43.90   3rd Qu.:13.89   3rd Qu.:2.923   3rd Qu.: 6.362  
##  Max.   :60.00   Max.   :23.63   Max.   :7.000   Max.   :10.000  
##                                                  NA's   :2       
##   avgAnnualInc    avgVerifStatus    avgHomeOwner       avgOpenAcc    
##  Min.   :  2000   Min.   :0.0000   Min.   :0.00000   Min.   : 1.000  
##  1st Qu.: 62276   1st Qu.:0.5333   1st Qu.:0.04000   1st Qu.: 9.446  
##  Median : 69840   Median :0.6667   Median :0.08898   Median :10.910  
##  Mean   : 69476   Mean   :0.5768   Mean   :0.09383   Mean   :10.505  
##  3rd Qu.: 76669   3rd Qu.:0.7244   3rd Qu.:0.12500   3rd Qu.:11.715  
##  Max.   :556880   Max.   :1.0000   Max.   :1.00000   Max.   :25.000  
##                                                      NA's   :9       
##   avgRevolBal      avgRevolUtil    avgTotalAcc     countOfLoans  
##  Min.   :     0   Min.   : 0.00   Min.   : 1.00   Min.   :   1  
##  1st Qu.: 12984   1st Qu.:49.06   1st Qu.:22.20   1st Qu.:  11  
##  Median : 15465   Median :53.97   Median :24.61   Median :  67  
##  Mean   : 15796   Mean   :52.59   Mean   :23.89   Mean   : 287  
##  3rd Qu.: 17677   3rd Qu.:57.94   3rd Qu.:26.29   3rd Qu.: 290  
##  Max.   :404868   Max.   :99.40   Max.   :61.00   Max.   :8081  
##                   NA's   :12      NA's   :9                      
##   monthly_date   
##  Min.   :2007 Jun  
##  1st Qu.:2010 Jun  
##  Median :2012 Nov  
##  Mean   :2012 Sep  
##  3rd Qu.:2015 Feb  
##  Max.   :2017 Mar  
## 
```

Load the ny_econ dataset

Q1d

```
ny_econ <- read.csv("nyEcon.csv")
ny_econ
```

```
##         date state   NYCPI NYUnemployment NYCondoPriceIdx NYSnapBenefits
## 1    6/1/07    NY 659.861            4.5          228.29        1801707
## 2    7/1/07    NY 660.931            4.6          228.16        1792916
## 3    8/1/07    NY 660.060            4.7          227.16        1816805
## 4    9/1/07    NY 660.006            4.7          226.14        1823494
## 5   10/1/07    NY 660.713            4.8          225.96        1825759
## 6   11/1/07    NY 663.464            4.8          226.88        1830858
## 7   12/1/07    NY 663.150            4.8          226.76        1849851
## 8    1/1/08    NY 664.520            4.8          227.19        1932022
## 9    2/1/08    NY 667.848            4.9          229.21        1927903
## 10   3/1/08    NY 673.924            4.9          231.26        1950582
## 11   4/1/08    NY 675.948            5.0          228.59        1968193
## 12   5/1/08    NY 682.680            5.1          226.87        1986156
## 13   6/1/08    NY 689.702            5.2          225.55        2004511
## 14   7/1/08    NY 694.595            5.4          224.47        2030668
## 15   8/1/08    NY 695.396            5.5          223.54        2051611
## 16   9/1/08    NY 694.064            5.7          221.65        2077774
## 17  10/1/08    NY 689.190            6.0          219.96        2114221
## 18  11/1/08    NY 677.900            6.3          218.30        2137106
## 19  12/1/08    NY 673.604            6.7          215.51        2174325
## 20   1/1/09    NY 674.732            7.1          213.60        2211935
## 21   2/1/09    NY 678.378            7.5          211.56        2246664
## 22   3/1/09    NY 679.546            7.8          208.41        2295103
## 23   4/1/09    NY 681.035            8.1          204.02        2339118
## 24   5/1/09    NY 682.171            8.3          201.63        2384027
## 25   6/1/09    NY 685.631            8.4          199.02        2427841
## 26   7/1/09    NY 686.869            8.5          197.13        2478604
## 27   8/1/09    NY 688.841            8.7          196.38        2508884
## 28   9/1/09    NY 689.668            8.8          196.04        2555081
## 29  10/1/09    NY 689.123            8.8          197.12        2599938
## 30  11/1/09    NY 690.272            8.9          197.42        2623264
## 31  12/1/09    NY 689.261            8.9          198.99        2673143
## 32   1/1/10    NY 690.828            8.9          199.47        2699586
## 33   2/1/10    NY 690.517            8.8          198.91        2712437
## 34   3/1/10    NY 694.099            8.8          199.69        2754632
## 35   4/1/10    NY 695.337            8.7          200.61        2775875
## 36   5/1/10    NY 696.916            8.6          202.38        2799734
## 37   6/1/10    NY 696.168            8.5          199.68        2824845
## 38   7/1/10    NY 697.123            8.5          198.98        2860394
## 39   8/1/10    NY 698.342            8.5          198.00        2874189
## 40   9/1/10    NY 698.099            8.5          199.70        2895995
## 41  10/1/10    NY 699.532            8.5          198.58        2918849
## 42  11/1/10    NY 699.473            8.4          199.31        2934493
## 43  12/1/10    NY 699.225            8.4          195.94        2969868
## 44   1/1/11    NY 701.436            8.3          194.84        2971876
## 45   2/1/11    NY 704.884            8.2          195.52        2975444
## 46   3/1/11    NY 710.044            8.1          195.90        3013945
## 47   4/1/11    NY 712.565            8.1          197.43        3017404
## 48   5/1/11    NY 717.146            8.1          195.10        3019981
## 49   6/1/11    NY 718.394            8.2          197.07        3035825
## 50   7/1/11    NY 720.299            8.3          197.17        3043751
## 51   8/1/11    NY 722.882            8.3          198.61        3040684
## 52   9/1/11    NY 724.331            8.4          197.92        3057767
## 53  10/1/11    NY 722.862            8.5          197.09        3060107
## 54  11/1/11    NY 720.740            8.5          195.88        3046972
## 55  12/1/11    NY 717.820            8.6          194.64        3068575
## 56   1/1/12    NY 720.754            8.6          193.72        3059120
## 57   2/1/12    NY 723.540            8.6          192.37        3059262
## 58   3/1/12    NY 728.171            8.7          194.06        3081831
## 59   4/1/12    NY 729.507            8.7          194.18        3063238
## 60   5/1/12    NY 730.381            8.7          198.36        3082995
## 61   6/1/12    NY 729.670            8.7          201.56        3095534
## 62   7/1/12    NY 728.545            8.6          203.58        3094677
## 63   8/1/12    NY 732.751            8.5          204.55        3109436
## 64   9/1/12    NY 735.879            8.4          203.99        3101190
## 65  10/1/12    NY 735.080            8.3          205.85        3110070
## 66  11/1/12    NY 735.102            8.2          207.49        3152122
## 67  12/1/12    NY 732.992            8.2          209.80        3186236
## 68   1/1/13    NY 736.613            8.1          210.80        3158541
## 69   2/1/13    NY 740.736            8.1          212.47        3153979
## 70   3/1/13    NY 741.764            8.0          213.11        3182976
## 71   4/1/13    NY 739.965            7.9          214.64        3181218
## 72   5/1/13    NY 740.840            7.8          215.84        3183287
## 73   6/1/13    NY 742.694            7.8          215.44        3186788
## 74   7/1/13    NY 743.894            7.7          216.61        3194470
## 75   8/1/13    NY 744.855            7.7          218.43        3186530
## 76   9/1/13    NY 747.300            7.6          224.95        3169363
## 77  10/1/13    NY 743.150            7.4          225.88        3170323
## 78  11/1/13    NY 744.042            7.3          229.03        3156551
## 79  12/1/13    NY 743.771            7.1          226.34        3158376
## 80   1/1/14    NY 750.456            7.0          230.80        3148532
## 81   2/1/14    NY 748.789            6.8          232.78        3114414
## 82   3/1/14    NY 751.541            6.7          234.22        3109524
## 83   4/1/14    NY 751.579            6.6          234.49        3103477
## 84   5/1/14    NY 755.164            6.5          235.72        3101888
## 85   6/1/14    NY 755.527            6.4          236.64        3094747
## 86   7/1/14    NY 755.953            6.2          238.52        3088298
## 87   8/1/14    NY 754.731            6.1          238.52        3075125
## 88   9/1/14    NY 754.728            6.0          239.56        3066686
## 89  10/1/14    NY 753.070            6.0          238.73        3068825
## 90  11/1/14    NY 749.838            5.9          240.16        3057644
## 91  12/1/14    NY 746.075            5.8          241.62        3075720
## 92   1/1/15    NY 746.929            5.7          244.12        3055942
## 93   2/1/15    NY 749.427            5.7          245.12        3045194
## 94   3/1/15    NY 750.602            5.6          245.93        3050058
## 95   4/1/15    NY 751.506            5.5          248.14        3039251
## 96   5/1/15    NY 754.705            5.4          250.48        3027230
## 97   6/1/15    NY 755.996            5.3          252.93        3028373
## 98   7/1/15    NY 755.091            5.2          254.15        3017604
## 99   8/1/15    NY 755.517            5.0          255.54        3001608
## 100  9/1/15    NY 757.080            5.0          256.57        3001849
## 101 10/1/15    NY 756.003            4.9          256.58        2996649
## 102 11/1/15    NY 754.540            4.9          257.21        2982398
## 103 12/1/15    NY 751.453            4.9          256.84        2990471
## 104  1/1/16    NY 752.612            4.9          258.53        2975036
## 105  2/1/16    NY 754.153            4.8          259.80        2972012
## 106  3/1/16    NY 755.983            4.8          260.62        2972806
## 107  4/1/16    NY 759.194            4.8          257.23        2961955
## 108  5/1/16    NY 761.198            4.8          259.21        2965167
## 109  6/1/16    NY 762.832            4.9          260.65        2953595
## 110  7/1/16    NY 762.383            4.9          261.88        2941315
## 111  8/1/16    NY 763.651            4.9          262.28        2957116
## 112  9/1/16    NY 764.929            5.0          261.12        2950208
## 113 10/1/16    NY 765.320            4.9          262.01        2938258
## 114 11/1/16    NY 766.664            4.9          262.45        2940107
## 115 12/1/16    NY 767.295            4.8          264.57        2949168
## 116  1/1/17    NY 771.621            4.7          265.52        2944348
## 117  2/1/17    NY 773.774            4.7          266.55        2922436
## 118  3/1/17    NY 773.542            4.7          267.15        2927021
```

Convert class of date from factor to Date

```
ny_econ$date<-mdy(ny_econ$date)
```

```
class(ny_econ$date)
```

```
## [1] "Date"
```

Convert ny_econ to tsibble using state as key and monthly_date and index

```
ts_ny_econ <- ny_econ %>%
  mutate(monthly_date = yearmonth(date))%>%
  as_tsibble(key = state, index = monthly_date)
ts_ny_econ
```

```
## # A tsibble: 118 x 7 [1M]
## # Key:       state [1]
##    date       state NYCPI NYUnemployment NYCondoPriceIdx NYSnapBenefits
##    <date>     <fct> <dbl>          <dbl>           <dbl>          <int>
##  1 2007-06-01 NY     660.            4.5            228.        1801707
##  2 2007-07-01 NY     661.            4.6            228.        1792916
##  3 2007-08-01 NY     660.            4.7            227.        1816805
##  4 2007-09-01 NY     660.            4.7            226.        1823494
##  5 2007-10-01 NY     661.            4.8            226.        1825759
##  6 2007-11-01 NY     663.            4.8            227.        1830858
##  7 2007-12-01 NY     663.            4.8            227.        1849851
##  8 2008-01-01 NY     665.            4.8            227.        1932022
##  9 2008-02-01 NY     668.            4.9            229.        1927903
## 10 2008-03-01 NY     674.            4.9            231.        1950582
## # ... with 108 more rows, and 1 more variable: monthly_date <mth>
```

Load the population dataset

Q1e

```
pop_data <- read_csv("DECENNIALSF12010.P1_data_with_overlays_2020-04-16T204138.csv")
```

```
## Parsed with column specification:
## cols(
##   GEO_ID = col_character(),
##   NAME = col_character(),
##   P001001 = col_character()
## )
```

```
pop_data
```

```
## # A tibble: 53 x 3
##    GEO_ID      NAME                 P001001
##    <chr>       <chr>                <chr>
##  1 id          Geographic Area Name Total
##  2 0400000US01 Alabama              4779736
##  3 0400000US02 Alaska               710231
##  4 0400000US04 Arizona              6392017
##  5 0400000US05 Arkansas             2915918
##  6 0400000US06 California           37253956
##  7 0400000US22 Louisiana            4533372
##  8 0400000US21 Kentucky             4339367
##  9 0400000US08 Colorado             5029196
## 10 0400000US09 Connecticut          3574097
## # ... with 43 more rows
```

Put State abbreviation as state

```
clean_pop <- pop_data %>%
  select(NAME,P001001)%>%
  mutate(state = state.abb[match(pop_data$NAME,state.name)])%>%
  mutate(population_1 = P001001)
clean_pop
```

```
## # A tibble: 53 x 4
##    NAME                 P001001  state population_1
##    <chr>                <chr>    <chr> <chr>
##  1 Geographic Area Name Total    <NA>  Total
##  2 Alabama              4779736  AL    4779736
##  3 Alaska               710231   AK    710231
##  4 Arizona              6392017  AZ    6392017
##  5 Arkansas             2915918  AR    2915918
##  6 California           37253956 CA    37253956
##  7 Louisiana            4533372  LA    4533372
##  8 Kentucky             4339367  KY    4339367
##  9 Colorado             5029196  CO    5029196
## 10 Connecticut          3574097  CT    3574097
## # ... with 43 more rows
```

Convert class type of population from character to numeric

```
clean_pop <- clean_pop %>%
  mutate(population = as.numeric(population_1))
```

```
## Warning: NAs introduced by coercion
```

```
  #select(state, population)
clean_pop
```

```
## # A tibble: 53 x 5
##    NAME                 P001001  state population_1 population
##    <chr>                <chr>    <chr> <chr>             <dbl>
##  1 Geographic Area Name Total    <NA>  Total                NA
##  2 Alabama              4779736  AL    4779736         4779736
##  3 Alaska               710231   AK    710231           710231
##  4 Arizona              6392017  AZ    6392017         6392017
##  5 Arkansas             2915918  AR    2915918         2915918
##  6 California           37253956 CA    37253956       37253956
##  7 Louisiana            4533372  LA    4533372         4533372
##  8 Kentucky             4339367  KY    4339367         4339367
##  9 Colorado             5029196  CO    5029196         5029196
## 10 Connecticut          3574097  CT    3574097         3574097
## # ... with 43 more rows
```

select two columns for clean_pop

```
clean_pop <- clean_pop %>%
  select(state, population)
clean_pop
```

```
## # A tibble: 53 x 2
##    state population
##    <chr>      <dbl>
##  1 <NA>          NA
##  2 AL       4779736
##  3 AK        710231
##  4 AZ       6392017
##  5 AR       2915918
##  6 CA      37253956
##  7 LA       4533372
##  8 KY       4339367
##  9 CO       5029196
## 10 CT       3574097
## # ... with 43 more rows
```

Join tsLCOrg with Population Data

```
tsLCOrg <- tsLCOrg %>% left_join(clean_pop)
```

```
## Joining, by = "state"
```

```
tsLCOrg
```

```
## # A tsibble: 4,943 x 18 [1M]
## # Key:       state [51]
##    date       state avgLoans totalLoans avgTerm avgIntRate avgGrade avgEmpLength
##    <date>     <chr>    <dbl>      <dbl>   <dbl>      <dbl>    <dbl>        <dbl>
##  1 2008-01-01 AK       5600        5600      36      18.0       7            5
##  2 2008-03-01 AK      11700       23400      36      11.8       3            3.5
##  3 2008-06-01 AK       7500        7500      36      13.9       4            3
##  4 2008-12-01 AK      25000       25000      36      15.2       5            1
##  5 2009-01-01 AK      15000       30000      36      12.5       2.5          7
##  6 2009-03-01 AK      14662.      29325      36      13         3            7
##  7 2009-04-01 AK      20000       20000      36      11.9       2            5
##  8 2009-05-01 AK      16000       16000      36      12.2       2            2
##  9 2009-07-01 AK       1000        1000      36      11.9       2           10
## 10 2009-11-01 AK      11000       11000      36       8.94      1            7
## # ... with 4,933 more rows, and 10 more variables: avgAnnualInc <dbl>,
## #   avgVerifStatus <dbl>, avgHomeOwner <dbl>, avgOpenAcc <dbl>,
## #   avgRevolBal <dbl>, avgRevolUtil <dbl>, avgTotalAcc <dbl>,
## #   countOfLoans <dbl>, monthly_date <mth>, population <dbl>
```

Calculate Loans Per Capita

```
tsLCOrg <- tsLCOrg %>%
  mutate(loansPerCapita = totalLoans/population)
tsLCOrg
```

```
## # A tsibble: 4,943 x 19 [1M]
## # Key:       state [51]
##    date       state avgLoans totalLoans avgTerm avgIntRate avgGrade avgEmpLength
##    <date>     <chr>    <dbl>      <dbl>   <dbl>      <dbl>    <dbl>        <dbl>
##  1 2008-01-01 AK       5600        5600      36      18.0       7            5
##  2 2008-03-01 AK      11700       23400      36      11.8       3            3.5
##  3 2008-06-01 AK       7500        7500      36      13.9       4            3
##  4 2008-12-01 AK      25000       25000      36      15.2       5            1
##  5 2009-01-01 AK      15000       30000      36      12.5       2.5           7
##  6 2009-03-01 AK      14662.      29325      36      13         3             7
##  7 2009-04-01 AK      20000       20000      36      11.9       2             5
##  8 2009-05-01 AK      16000       16000      36      12.2       2             2
##  9 2009-07-01 AK       1000        1000      36      11.9       2            10
## 10 2009-11-01 AK      11000       11000      36       8.94      1             7
## # ... with 4,933 more rows, and 11 more variables: avgAnnualInc <dbl>,
## #   avgVerifStatus <dbl>, avgHomeOwner <dbl>, avgOpenAcc <dbl>,
## #   avgRevolBal <dbl>, avgRevolUtil <dbl>, avgTotalAcc <dbl>,
## #   countOfLoans <dbl>, monthly_date <mth>, population <dbl>,
## #   loansPerCapita <dbl>
```

Coverting ny_econ state from factor to character

```r
ny_econ <- ny_econ %>%
  mutate(state = as.character(state))
ny_econ
```

```
##           date state   NYCPI NYUnemployment NYCondoPriceIdx NYSnapBenefits
## 1   2007-06-01    NY 659.861            4.5          228.29        1801707
## 2   2007-07-01    NY 660.931            4.6          228.16        1792916
## 3   2007-08-01    NY 660.060            4.7          227.16        1816805
## 4   2007-09-01    NY 660.006            4.7          226.14        1823494
## 5   2007-10-01    NY 660.713            4.8          225.96        1825759
## 6   2007-11-01    NY 663.464            4.8          226.88        1830858
## 7   2007-12-01    NY 663.150            4.8          226.76        1849851
## 8   2008-01-01    NY 664.520            4.8          227.19        1932022
## 9   2008-02-01    NY 667.848            4.9          229.21        1927903
## 10  2008-03-01    NY 673.924            4.9          231.26        1950582
## 11  2008-04-01    NY 675.948            5.0          228.59        1968193
## 12  2008-05-01    NY 682.680            5.1          226.87        1986156
## 13  2008-06-01    NY 689.702            5.2          225.55        2004511
## 14  2008-07-01    NY 694.595            5.4          224.47        2030668
## 15  2008-08-01    NY 695.396            5.5          223.54        2051611
## 16  2008-09-01    NY 694.064            5.7          221.65        2077774
## 17  2008-10-01    NY 689.190            6.0          219.96        2114221
## 18  2008-11-01    NY 677.900            6.3          218.30        2137106
## 19  2008-12-01    NY 673.604            6.7          215.51        2174325
## 20  2009-01-01    NY 674.732            7.1          213.60        2211935
## 21  2009-02-01    NY 678.378            7.5          211.56        2246664
## 22  2009-03-01    NY 679.546            7.8          208.41        2295103
## 23  2009-04-01    NY 681.035            8.1          204.02        2339118
## 24  2009-05-01    NY 682.171            8.3          201.63        2384027
## 25  2009-06-01    NY 685.631            8.4          199.02        2427841
## 26  2009-07-01    NY 686.869            8.5          197.13        2478604
## 27  2009-08-01    NY 688.841            8.7          196.38        2508884
## 28  2009-09-01    NY 689.668            8.8          196.04        2555081
## 29  2009-10-01    NY 689.123            8.8          197.12        2599938
## 30  2009-11-01    NY 690.272            8.9          197.42        2623264
## 31  2009-12-01    NY 689.261            8.9          198.99        2673143
## 32  2010-01-01    NY 690.828            8.9          199.47        2699586
## 33  2010-02-01    NY 690.517            8.8          198.91        2712437
## 34  2010-03-01    NY 694.099            8.8          199.69        2754632
## 35  2010-04-01    NY 695.337            8.7          200.61        2775875
## 36  2010-05-01    NY 696.916            8.6          202.38        2799734
## 37  2010-06-01    NY 696.168            8.5          199.68        2824845
## 38  2010-07-01    NY 697.123            8.5          198.98        2860394
## 39  2010-08-01    NY 698.342            8.5          198.00        2874189
## 40  2010-09-01    NY 698.099            8.5          199.70        2895995
## 41  2010-10-01    NY 699.532            8.5          198.58        2918849
## 42  2010-11-01    NY 699.473            8.4          199.31        2934493
## 43  2010-12-01    NY 699.225            8.4          195.94        2969868
## 44  2011-01-01    NY 701.436            8.3          194.84        2971876
## 45  2011-02-01    NY 704.884            8.2          195.52        2975444
## 46  2011-03-01    NY 710.044            8.1          195.90        3013945
## 47  2011-04-01    NY 712.565            8.1          197.43        3017404
## 48  2011-05-01    NY 717.146            8.1          195.10        3019981
## 49  2011-06-01    NY 718.394            8.2          197.07        3035825
## 50  2011-07-01    NY 720.299            8.3          197.17        3043751
## 51  2011-08-01    NY 722.882            8.3          198.61        3040684
## 52  2011-09-01    NY 724.331            8.4          197.92        3057767
## 53  2011-10-01    NY 722.862            8.5          197.09        3060107
## 54  2011-11-01    NY 720.740            8.5          195.88        3046972
## 55  2011-12-01    NY 717.820            8.6          194.64        3068575
## 56  2012-01-01    NY 720.754            8.6          193.72        3059120
## 57  2012-02-01    NY 723.540            8.6          192.37        3059262
## 58  2012-03-01    NY 728.171            8.7          194.06        3081831
## 59  2012-04-01    NY 729.507            8.7          194.18        3063238
## 60  2012-05-01    NY 730.381            8.7          198.36        3082995
## 61  2012-06-01    NY 729.670            8.7          201.56        3095534
## 62  2012-07-01    NY 728.545            8.6          203.58        3094677
## 63  2012-08-01    NY 732.751            8.5          204.55        3109436
## 64  2012-09-01    NY 735.879            8.4          203.99        3101190
## 65  2012-10-01    NY 735.080            8.3          205.85        3110070
## 66  2012-11-01    NY 735.102            8.2          207.49        3152122
## 67  2012-12-01    NY 732.992            8.2          209.80        3186236
## 68  2013-01-01    NY 736.613            8.1          210.80        3158541
## 69  2013-02-01    NY 740.736            8.1          212.47        3153979
## 70  2013-03-01    NY 741.764            8.0          213.11        3182976
## 71  2013-04-01    NY 739.965            7.9          214.64        3181218
## 72  2013-05-01    NY 740.840            7.8          215.84        3183287
## 73  2013-06-01    NY 742.694            7.8          215.44        3186788
## 74  2013-07-01    NY 743.894            7.7          216.61        3194470
## 75  2013-08-01    NY 744.855            7.7          218.43        3186530
## 76  2013-09-01    NY 747.300            7.6          224.95        3169363
## 77  2013-10-01    NY 743.150            7.4          225.88        3170323
## 78  2013-11-01    NY 744.042            7.3          229.03        3156551
## 79  2013-12-01    NY 743.771            7.1          226.34        3158376
## 80  2014-01-01    NY 750.456            7.0          230.80        3148532
## 81  2014-02-01    NY 748.789            6.8          232.78        3114414
## 82  2014-03-01    NY 751.541            6.7          234.22        3109524
## 83  2014-04-01    NY 751.579            6.6          234.49        3103477
## 84  2014-05-01    NY 755.164            6.5          235.72        3101888
## 85  2014-06-01    NY 755.527            6.4          236.64        3094747
## 86  2014-07-01    NY 755.953            6.2          238.52        3088298
## 87  2014-08-01    NY 754.731            6.1          238.52        3075125
## 88  2014-09-01    NY 754.728            6.0          239.56        3066686
## 89  2014-10-01    NY 753.070            6.0          238.73        3068825
## 90  2014-11-01    NY 749.838            5.9          240.16        3057644
## 91  2014-12-01    NY 746.075            5.8          241.62        3075720
## 92  2015-01-01    NY 746.929            5.7          244.12        3055942
## 93  2015-02-01    NY 749.427            5.7          245.12        3045194
## 94  2015-03-01    NY 750.602            5.6          245.93        3050058
## 95  2015-04-01    NY 751.506            5.5          248.14        3039251
## 96  2015-05-01    NY 754.705            5.4          250.48        3027230
## 97  2015-06-01    NY 755.996            5.3          252.93        3028373
## 98  2015-07-01    NY 755.091            5.2          254.15        3017604
## 99  2015-08-01    NY 755.517            5.0          255.54        3001608
## 100 2015-09-01    NY 757.080            5.0          256.57        3001849
## 101 2015-10-01    NY 756.003            4.9          256.58        2996649
## 102 2015-11-01    NY 754.540            4.9          257.21        2982398
## 103 2015-12-01    NY 751.453            4.9          256.84        2990471
## 104 2016-01-01    NY 752.612            4.9          258.53        2975036
## 105 2016-02-01    NY 754.153            4.8          259.80        2972012
## 106 2016-03-01    NY 755.983            4.8          260.62        2972806
## 107 2016-04-01    NY 759.194            4.8          257.23        2961955
## 108 2016-05-01    NY 761.198            4.8          259.21        2965167
## 109 2016-06-01    NY 762.832            4.9          260.65        2953595
## 110 2016-07-01    NY 762.383            4.9          261.88        2941315
## 111 2016-08-01    NY 763.651            4.9          262.28        2957116
## 112 2016-09-01    NY 764.929            5.0          261.12        2950208
## 113 2016-10-01    NY 765.320            4.9          262.01        2938258
## 114 2016-11-01    NY 766.664            4.9          262.45        2940107
## 115 2016-12-01    NY 767.295            4.8          264.57        2949168
## 116 2017-01-01    NY 771.621            4.7          265.52        2944348
## 117 2017-02-01    NY 773.774            4.7          266.55        2922436
## 118 2017-03-01    NY 773.542            4.7          267.15        2927021
```

```r
class(ny_econ$state)
```

```
## [1] "character"
```

Joining tsLCOrg with ny_econ

```r
tsLC <- left_join(tsLCOrg, ny_econ, by = c("date", "state"))
tsLC
```

```
## # A tsibble: 4,943 x 23 [1M]
## # Key:       state [51]
##    date       state avgLoans totalLoans avgTerm avgIntRate avgGrade avgEmpLength
##    <date>     <chr>    <dbl>      <dbl>   <dbl>      <dbl>    <dbl>        <dbl>
##  1 2008-01-01 AK       5600        5600      36      18.0       7             5
##  2 2008-03-01 AK      11700       23400      36      11.8       3             3.5
##  3 2008-06-01 AK       7500        7500      36      13.9       4             3
##  4 2008-12-01 AK      25000       25000      36      15.2       5             1
##  5 2009-01-01 AK      15000       30000      36      12.5       2.5           7
##  6 2009-03-01 AK      14662.      29325      36      13         3             7
##  7 2009-04-01 AK      20000       20000      36      11.9       2             5
##  8 2009-05-01 AK      16000       16000      36      12.2       2             2
##  9 2009-07-01 AK       1000        1000      36      11.9       2            10
## 10 2009-11-01 AK      11000       11000      36       8.94      1             7
## # ... with 4,933 more rows, and 15 more variables: avgAnnualInc <dbl>,
## #   avgVerifStatus <dbl>, avgHomeOwner <dbl>, avgOpenAcc <dbl>,
## #   avgRevolBal <dbl>, avgRevolUtil <dbl>, avgTotalAcc <dbl>,
## #   countOfLoans <dbl>, monthly_date <mth>, population <dbl>,
## #   loansPerCapita <dbl>, NYCPI <dbl>, NYUnemployment <dbl>,
## #   NYCondoPriceIdx <dbl>, NYSnapBenefits <int>
```

Converting to tsibble

```
tsLC <-
  tsLC %>%
  mutate(monthly_date = yearmonth(date))%>%
  as_tsibble(key = state, index = monthly_date)
```

```
tsLC
```

```
## # A tsibble: 4,943 x 23 [1M]
## # Key:       state [51]
##    date       state avgLoans totalLoans avgTerm avgIntRate avgGrade avgEmpLength
##    <date>     <chr>    <dbl>      <dbl>   <dbl>      <dbl>    <dbl>        <dbl>
##  1 2008-01-01 AK       5600       5600      36      18.0       7            5
##  2 2008-03-01 AK      11700      23400      36      11.8       3            3.5
##  3 2008-06-01 AK       7500       7500      36      13.9       4            3
##  4 2008-12-01 AK      25000      25000      36      15.2       5            1
##  5 2009-01-01 AK      15000      30000      36      12.5       2.5          7
##  6 2009-03-01 AK      14662.     29325      36      13         3            7
##  7 2009-04-01 AK      20000      20000      36      11.9       2            5
##  8 2009-05-01 AK      16000      16000      36      12.2       2            2
##  9 2009-07-01 AK       1000       1000      36      11.9       2           10
## 10 2009-11-01 AK      11000      11000      36       8.94      1            7
## # ... with 4,933 more rows, and 15 more variables: avgAnnualInc <dbl>,
## #   avgVerifStatus <dbl>, avgHomeOwner <dbl>, avgOpenAcc <dbl>,
## #   avgRevolBal <dbl>, avgRevolUtil <dbl>, avgTotalAcc <dbl>,
## #   countOfLoans <dbl>, monthly_date <mth>, population <dbl>,
## #   loansPerCapita <dbl>, NYCPI <dbl>, NYUnemployment <dbl>,
## #   NYCondoPriceIdx <dbl>, NYSnapBenefits <int>
```

```
class(tsLC)
```

```
## [1] "tbl_ts"     "tbl_df"     "tbl"        "data.frame"
```

Q2a

```
temp <- tsLC %>%
    as_data_frame() %>%
    group_by(state)%>%
    summarize(mean_pop = mean(population), mean_lpc = mean(loansPerCapita))
```

```
## Warning: `as_data_frame()` is deprecated, use `as_tibble()` (but mind the new semantics).
## This warning is displayed once per session.
```

```
temp
```

```
## # A tibble: 51 x 3
##    state mean_pop mean_lpc
##    <chr>    <dbl>    <dbl>
##  1 AK      710231    0.857
##  2 AL     4779736    0.495
##  3 AR      2915918    0.489
##  4 AZ      6392017    0.635
##  5 CA     37253956    0.703
##  6 CO      5029196    0.750
##  7 CT      3574097    0.813
##  8 DC           NA   NA
##  9 DE       897934    0.614
## 10 FL     18801310    0.625
## # ... with 41 more rows
```

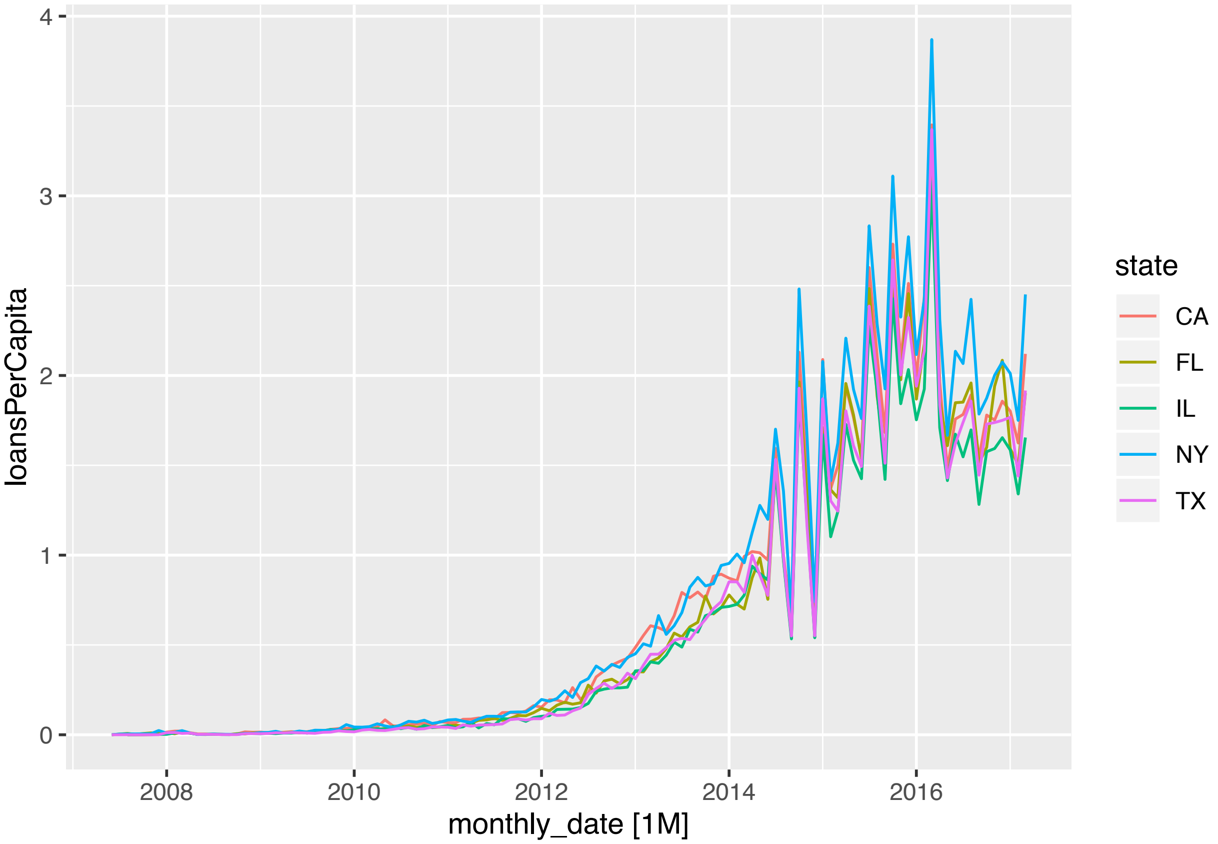
```
bottom_10<-temp %>%
  filter(temp$mean_pop < quantile(temp$mean_pop, 0.1, na.rm = TRUE))
bottom_10
```

```
## # A tibble: 5 x 3
##   state mean_pop mean_lpc
##   <chr>    <dbl>    <dbl>
## 1 AK      710231    0.857
## 2 ND      672591    1.88
## 3 SD      814180    0.531
## 4 VT      625741    0.754
## 5 WY      563626    0.894
```

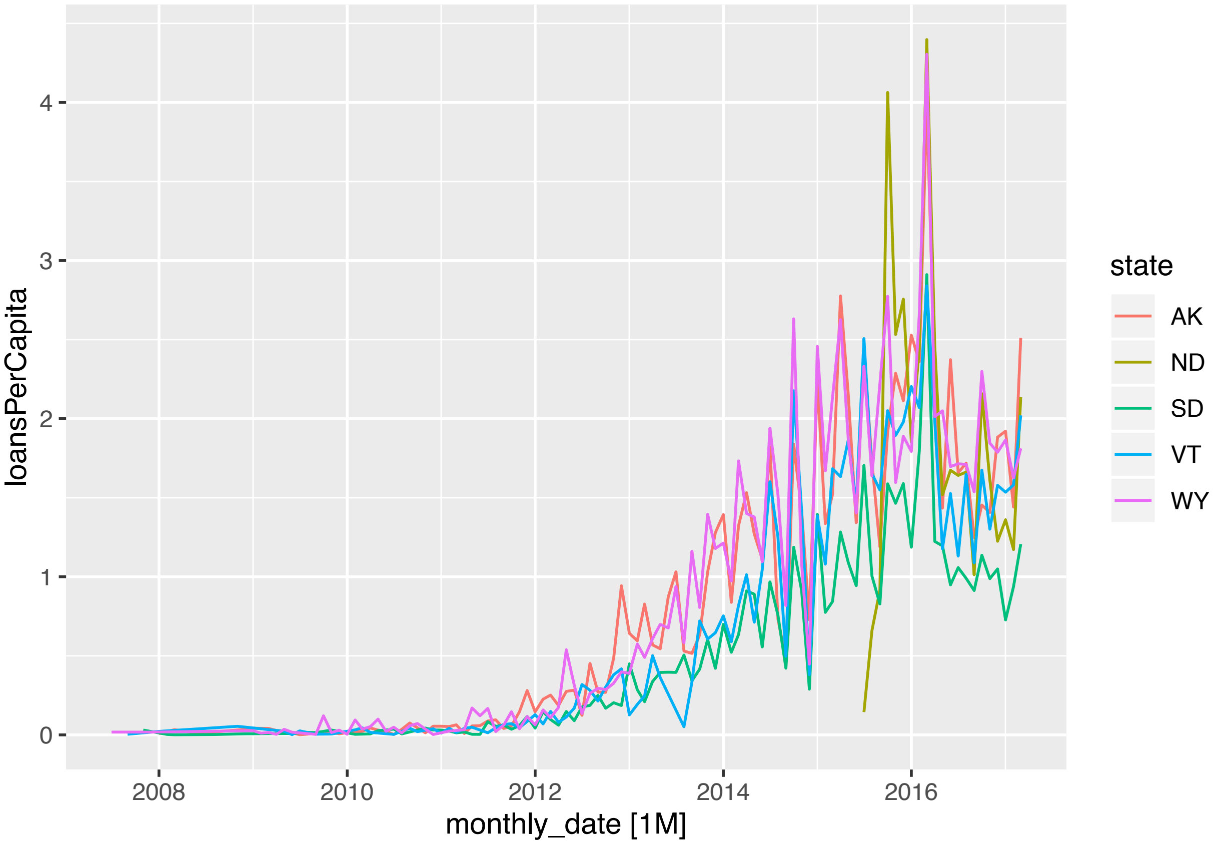
```
top_10<-temp %>%
  filter(temp$mean_pop > quantile(temp$mean_pop, 0.9, na.rm = TRUE))
top_10
```

```
## # A tibble: 5 x 3
##   state mean_pop mean_lpc
##   <chr>    <dbl>    <dbl>
## 1 CA    37253956    0.703
## 2 FL    18801310    0.625
## 3 IL    12830632    0.597
## 4 NY    19378102    0.750
## 5 TX    25145561    0.625
```

```
plot_top10 <- tsLC %>%
  filter(state %in% c("CA","FL","IL","NY","TX"))
plot_top10 %>%
  autoplot(loansPerCapita)
```



```
plot_bottom10 <- tsLC %>%
  filter(state %in% c("AK", "ND","SD","VT","WY"))
plot_bottom10 %>%
  autoplot(loansPerCapita)
```



###Q2b

```
library(anomalize)
```

```
## == Use anomalize to improve your Forecasts by 50%! =============================
## Business Science offers a 1-hour course - Lab #18: Time Series Anomaly Detection!
## </> Learn more at: https://university.business-science.io/p/learning-labs-pro </>
```

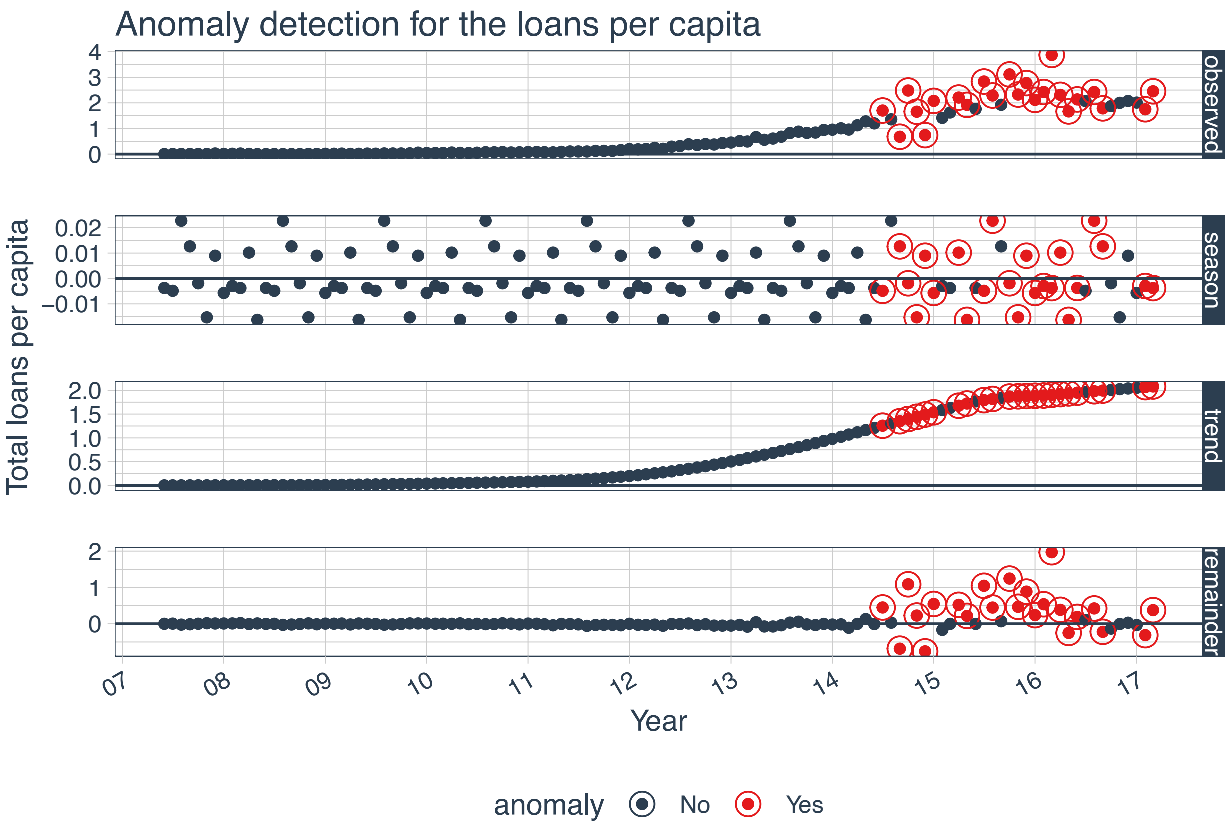
```
library(tibbletime)
library(tsibbledata)
```

```
ny_anomaly <-
  tsLCOrg %>%
  filter(state == 'NY') %>%
  as_tbl_time(index = date) %>% as_period("month") %>%
  time_decompose(loansPerCapita, method = "stl") %>%
  anomalize(remainder, method = "iqr") %>%
  #plot_anomalies() +
  plot_anomaly_decomposition()+
  labs(title = "Anomaly detection for the loans per capita") +
  xlab("Year") + ylab("Total loans per capita ") +
  scale_x_date(date_breaks = "years" , date_labels = "%y")
```

```
## frequency = 12 months
```

```
## trend = 31 months
```

```
ny_anomaly
```

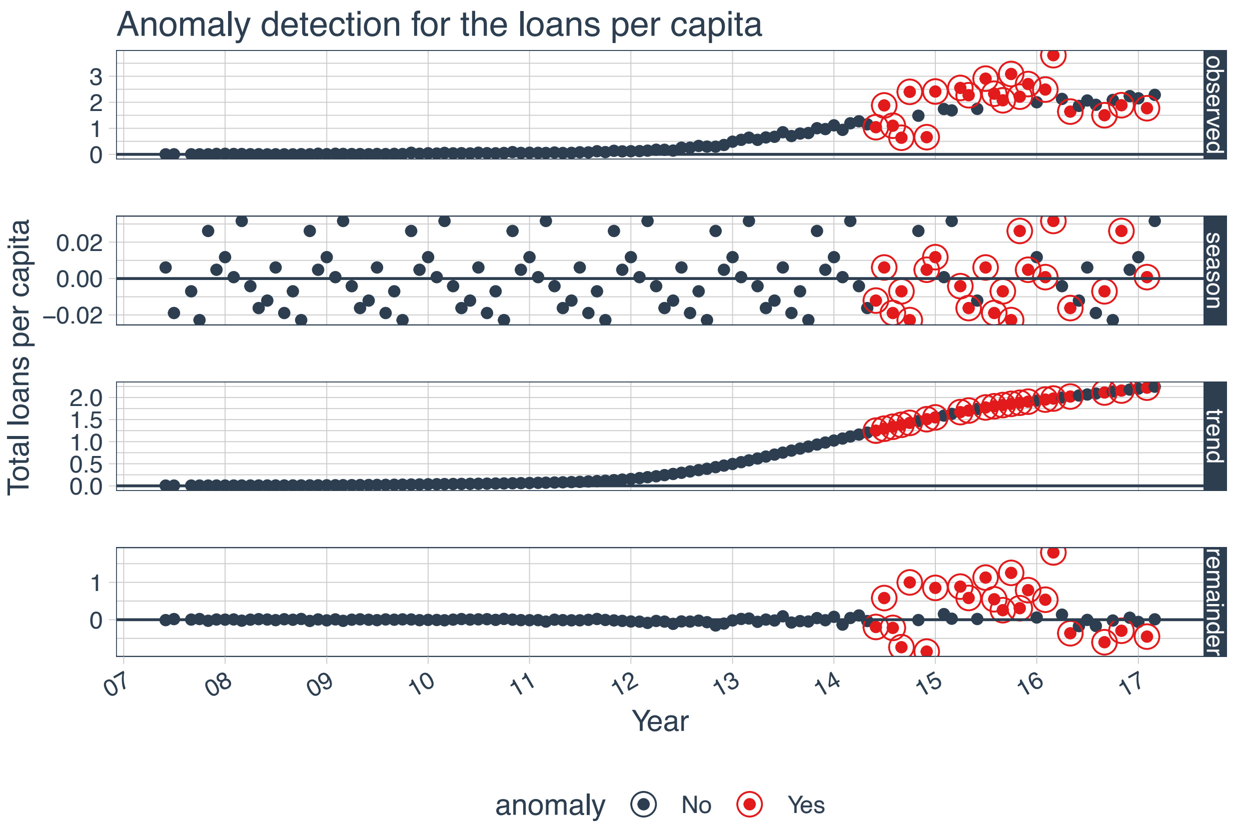


```
co_anomaly <-
  tsLCOrg %>%
  filter(state == 'CO') %>%
  as_tbl_time(index = date) %>% as_period("month") %>%
  time_decompose(loansPerCapita, method = "stl") %>%
  anomalize(remainder, method = "iqr") %>%
  plot_anomaly_decomposition() +
  labs(title = "Anomaly detection for the loans per capita") +
  xlab("Year") + ylab("Total loans per capita ") +
  scale_x_date(date_breaks = "years" , date_labels = "%y")
```

```
## frequency = 12 months
```

```
## trend = 30 months
```

```
co_anomaly
```

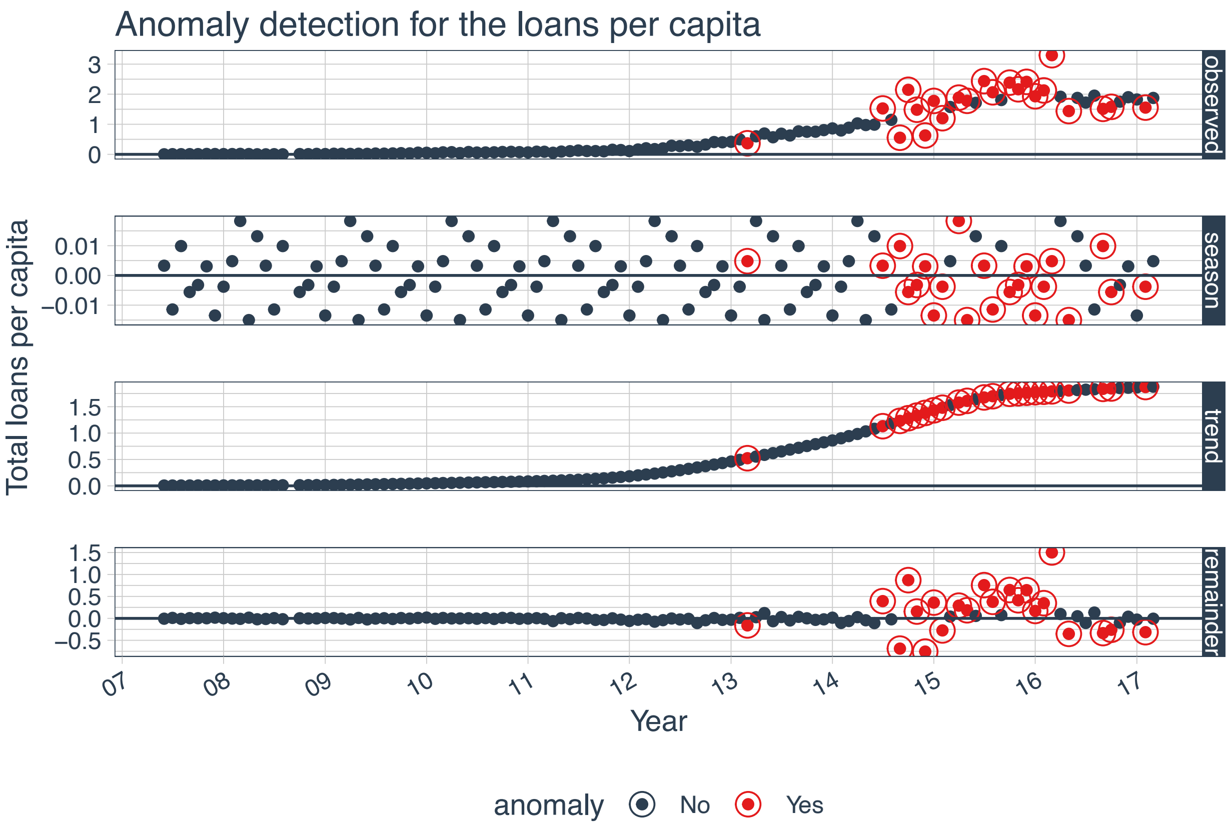


```
ma_anomaly <-
  tsLCOrg %>%
  filter(state == 'MA') %>%
  as_tbl_time(index = date) %>% as_period("month") %>%
  time_decompose(loansPerCapita, method = "stl") %>%
  anomalize(remainder, method = "iqr") %>%
  #plot_anomalies() +
  plot_anomaly_decomposition() +
  labs(title = "Anomaly detection for the loans per capita") +
  xlab("Year") + ylab("Total loans per capita ") +
  scale_x_date(date_breaks = "years" , date_labels = "%y")
```

```
## frequency = 12 months
```

```
## trend = 30 months
```

```
ma_anomaly
```

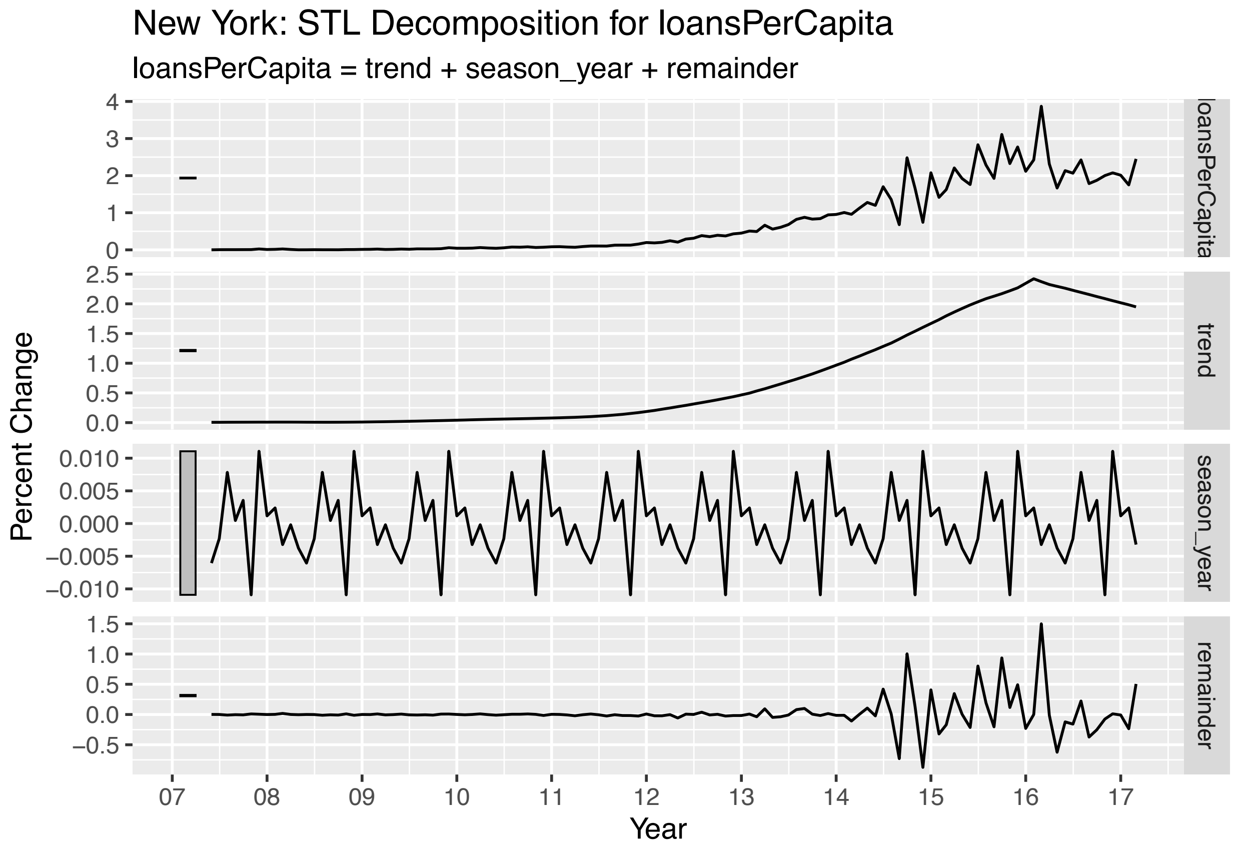


Q2c

```
ny_decomp <-
  tsLC %>%
  filter(state == "NY") %>%
  mutate(date = yearmonth(date)) %>%
  select(date, loansPerCapita) %>%
  model(STL(loansPerCapita ~ trend() + season(window = "periodic"), robust = TRUE)) %>%
  components() %>%
  autoplot() +
  xlab("Year") + ylab("Percent Change") +
  ggtitle("New York: STL Decomposition for loansPerCapita") +
  scale_x_date(date_breaks = "years" , date_labels = "%y")
```

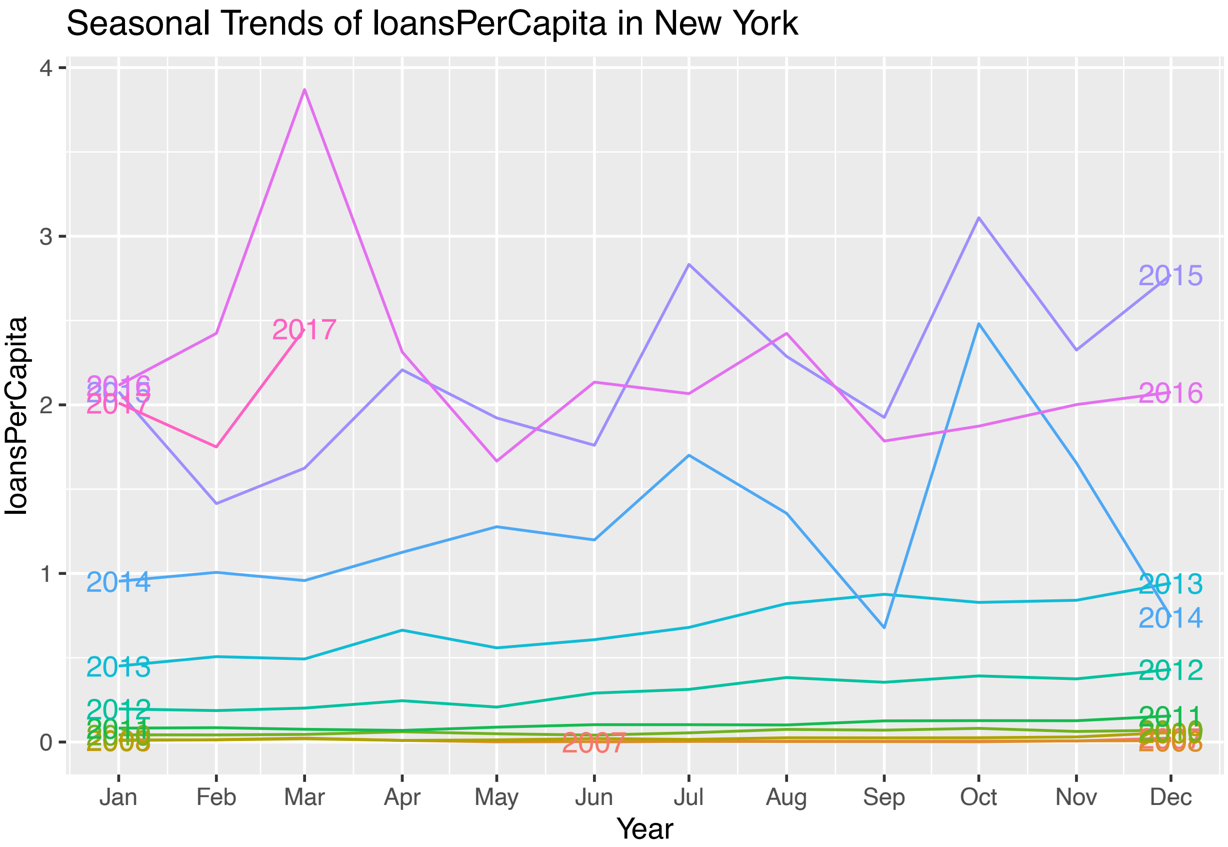
```
## Selecting index: "monthly_date"
```

```
ny_decomp
```



#Q2d

```
plotNySeason <-
  tsLC %>% filter(state=="NY") %>%  mutate(date = yearmonth(date)) %>%
  gg_season(loansPerCapita, labels = "both") +
  xlab("Year") + ylab("loansPerCapita") +
  ggtitle("Seasonal Trends of loansPerCapita in New York")
(plotNySeason)
```

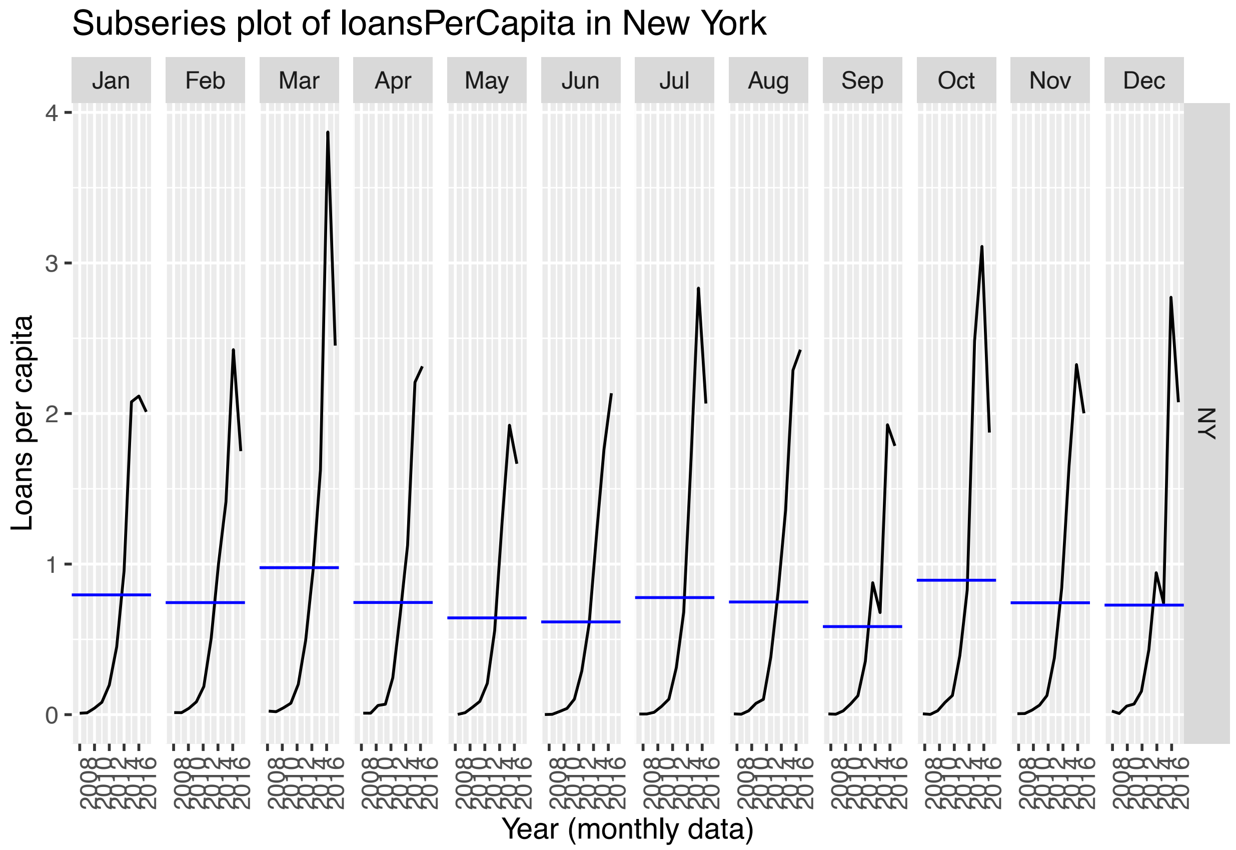


```
ny_s<-tsLC %>%
  filter(state=="NY") %>%
  mutate(monthly_date = yearmonth(date)) %>%
  as_tbl_time(index = monthly_date)
```

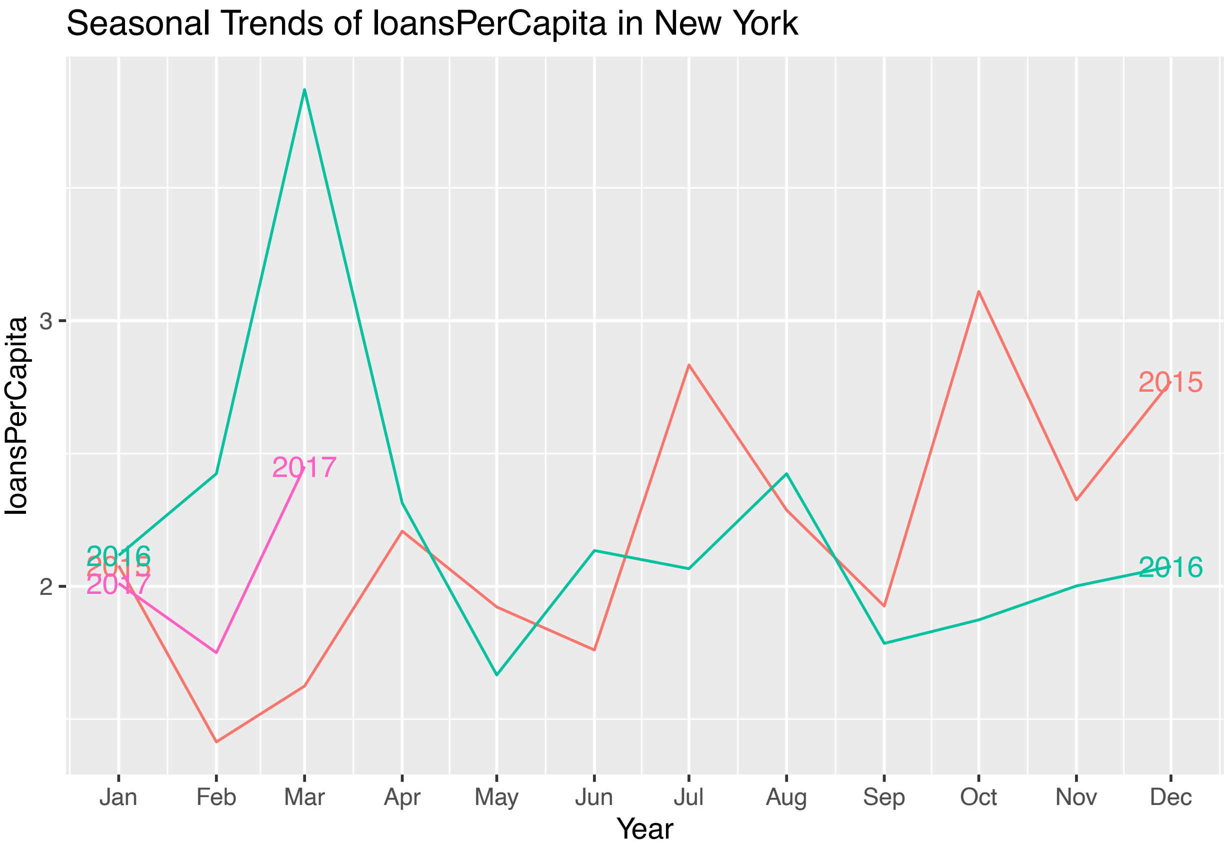
```
class(ny_s)
```

```
## [1] "tbl_time"   "tbl_df"     "tbl"        "data.frame"
```

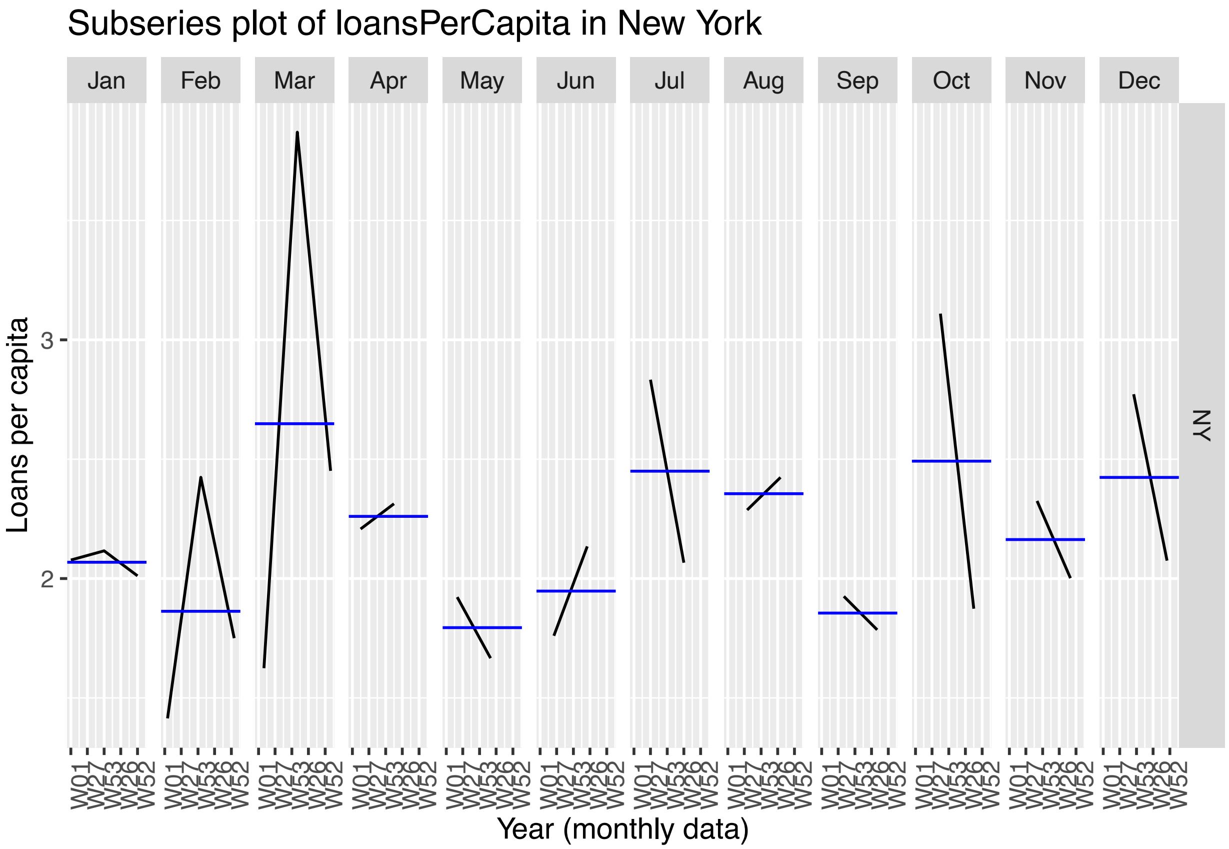
```
plotNySubSeries <-
 tsLC %>% filter(state=="NY") %>%  mutate(date = yearmonth(date)) %>%
  gg_subseries(loansPerCapita) +
  ylab("Loans per capita") +
  xlab("Year (monthly data)") +
  ggtitle("Subseries plot of loansPerCapita in New York")
plotNySubSeries
```



```r
plotNySeason3 <- tsLC %>%
  filter(year(date) > 2014)%>%
  filter(state=="NY") %>%
  mutate(date = yearmonth(date)) %>%
  gg_season(loansPerCapita, labels = "both") +
  xlab("Year") + ylab("loansPerCapita") +
  ggtitle("Seasonal Trends of loansPerCapita in New York")
(plotNySeason3)
```

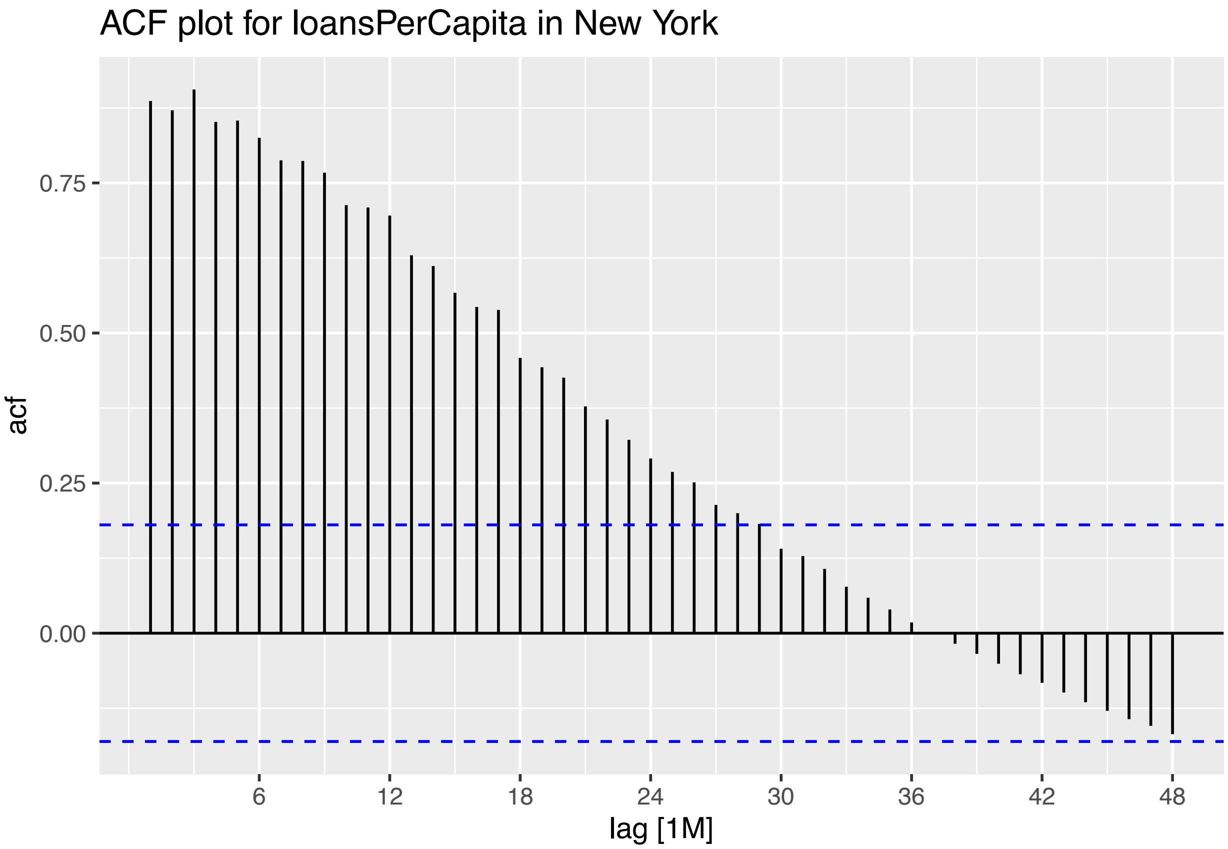


```
plotNySubSeries3 <- tsLC %>%
  filter(year(date) > 2014)%>%
  filter(state=="NY") %>%
  mutate(date = yearmonth(date)) %>%
  gg_subseries(loansPerCapita) +
  ylab("Loans per capita") +
  xlab("Year (monthly data)") +
  ggtitle("Subseries plot of loansPerCapita in New York")
plotNySubSeries3
```

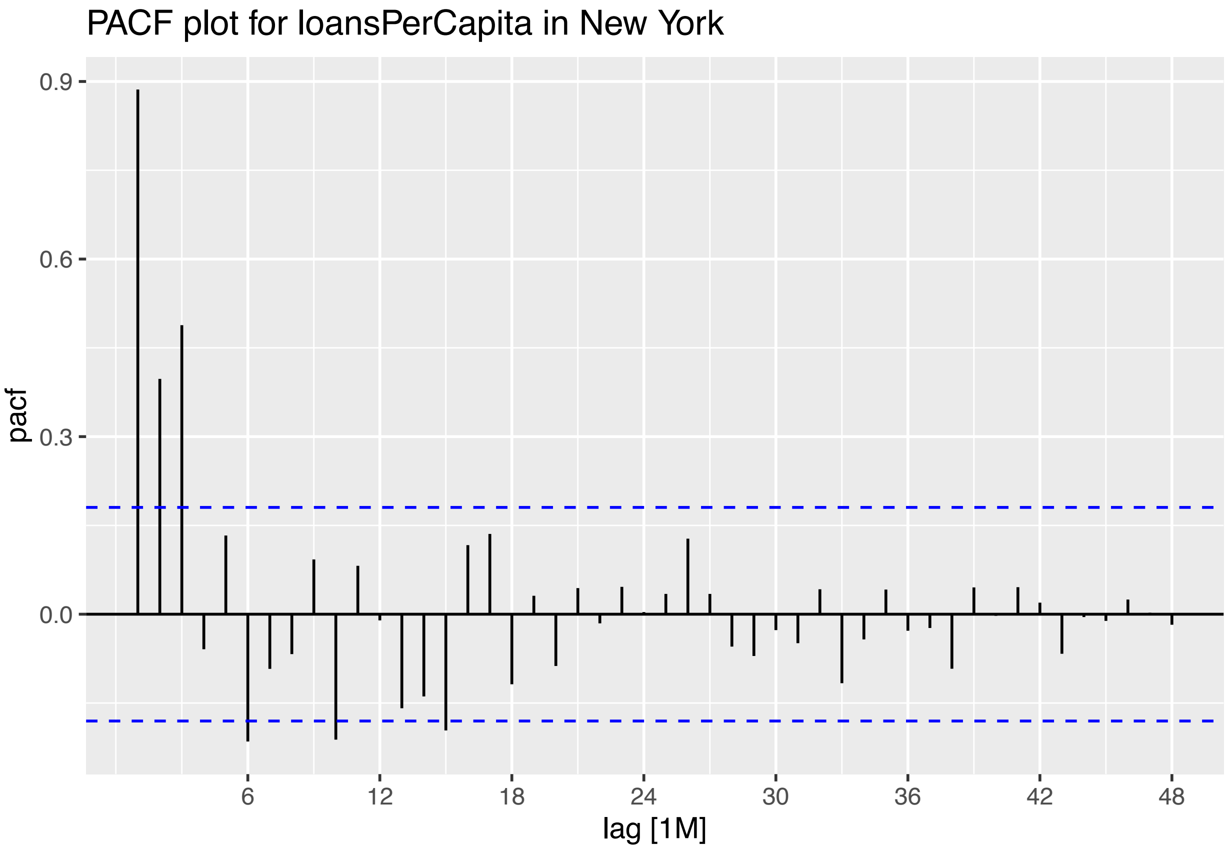


#Q2e

```
plotNyACF <- tsLC %>%
  filter(state=="NY") %>%
  mutate(date = yearmonth(date)) %>%
  ACF(loansPerCapita, lag_max = 48) %>%
  autoplot() +  ggtitle("ACF plot for loansPerCapita in New York")
plotNyACF
```



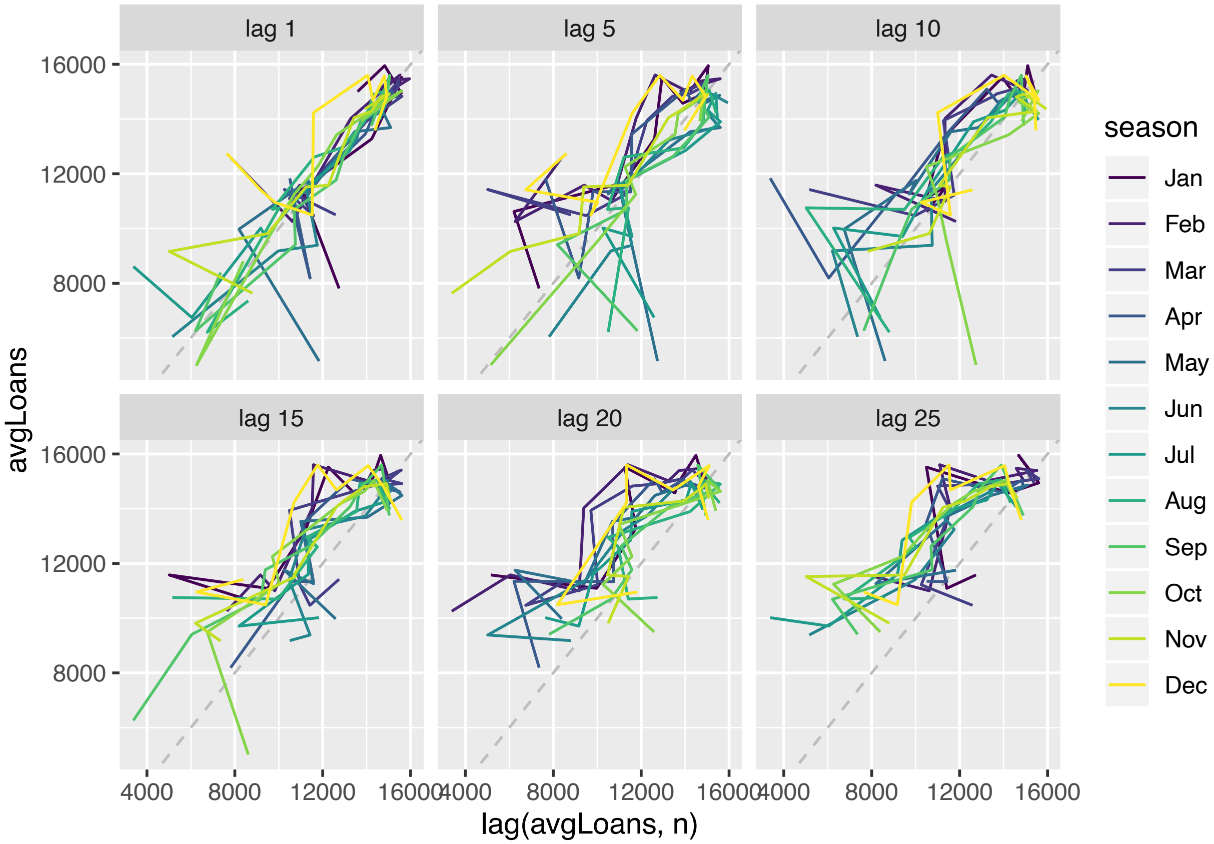
```
plotNyPACF <- tsLC %>%
  filter(state=="NY") %>%
  mutate(date = yearmonth(date)) %>%
  PACF(loansPerCapita, lag_max = 48) %>%
  autoplot() +  ggtitle("PACF plot for loansPerCapita in New York")
plotNyPACF
```



#Q2f

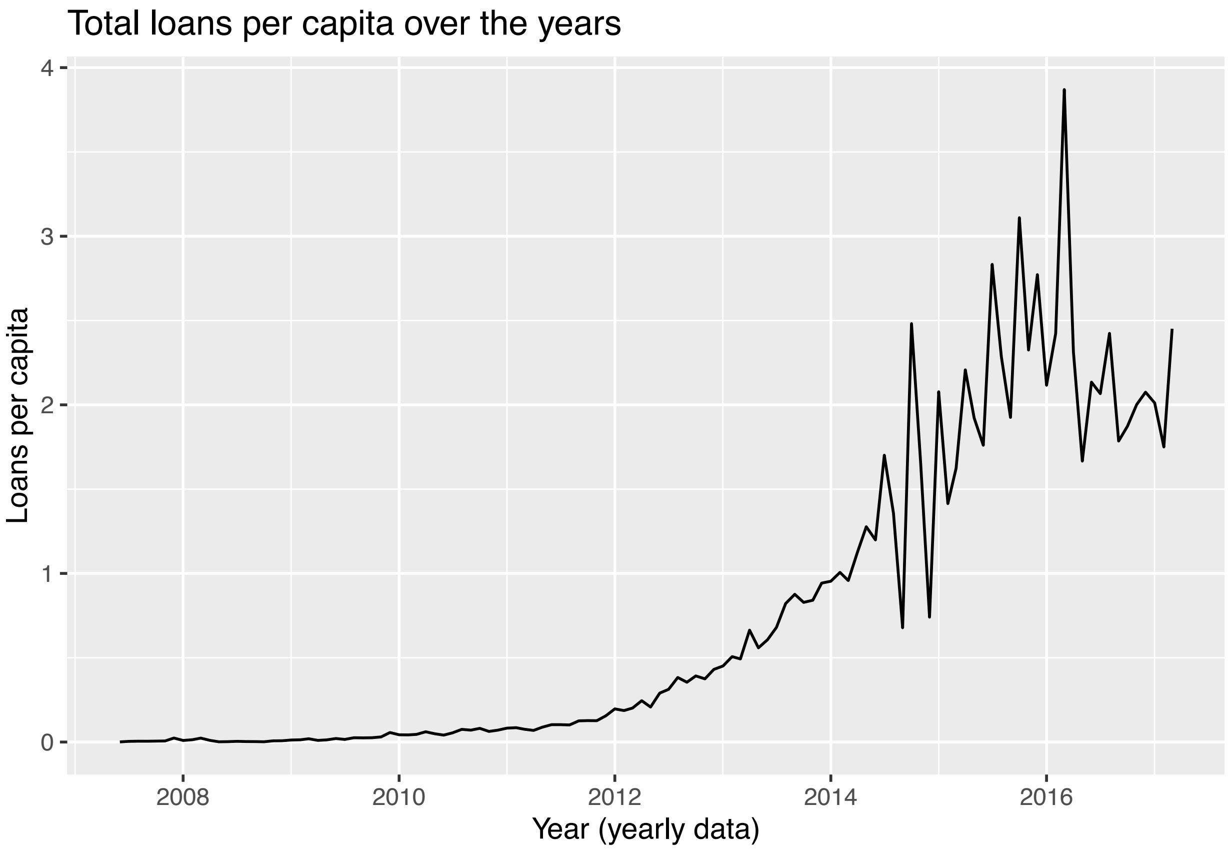
```
plotLag <- tsLC %>%
  filter(state == "NY") %>%
  mutate(date = yearmonth(date))%>%
  as_tsibble(key = state, index = date)

gg_lag(plotLag,lags = c(1,5,10,15,20,25))
```

```
## Plot variable not specified, automatically selected `y = avgLoans`
```



#Q2g

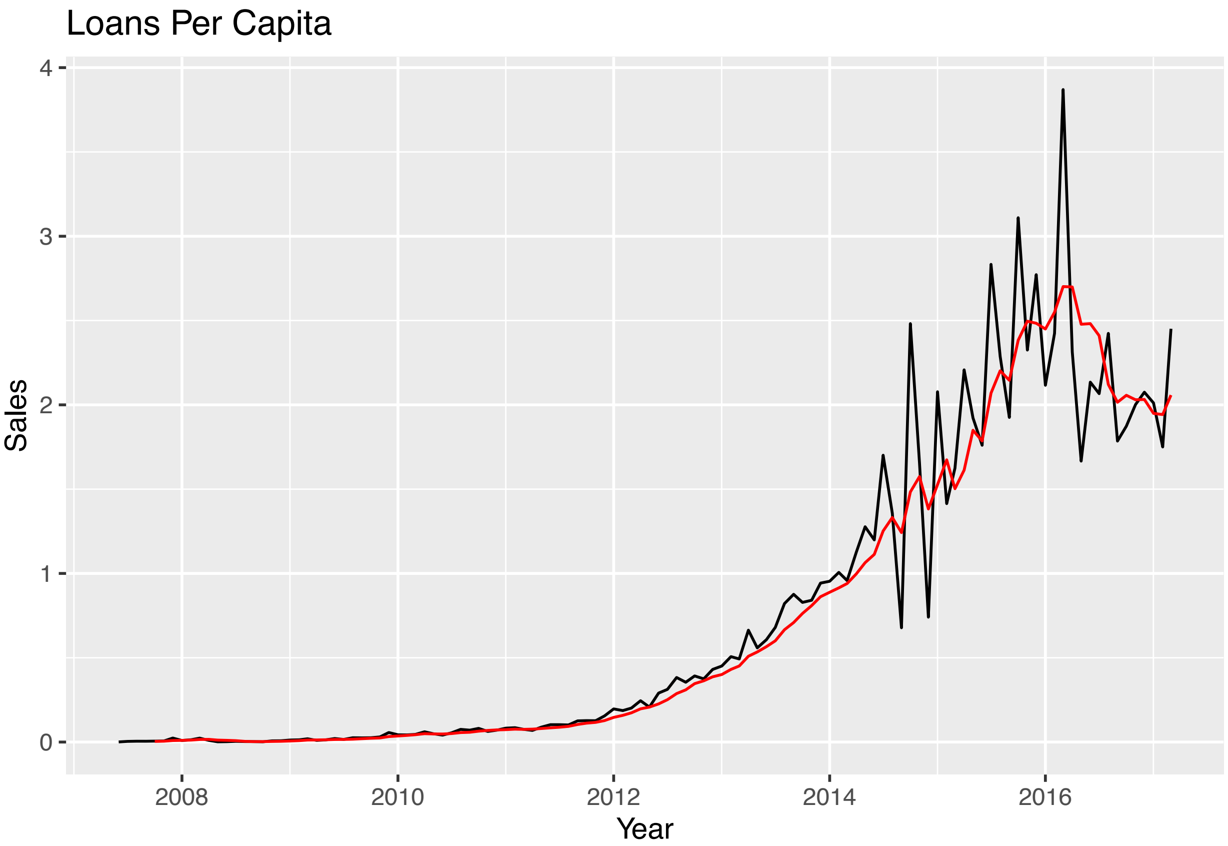
```
lpc_ny <- tsLC %>%
  filter(state=="NY") %>%
  autoplot(loansPerCapita) +
  xlab("Year (yearly data)") + ylab("Loans per capita") +
  ggtitle("Total loans per capita over the years")
lpc_ny
```



```
tsny_5ma <- tsLC %>%
  filter(state == "NY")%>%
  mutate(`5-MA` = slide_dbl(loansPerCapita, mean, .size = 5))
```

```
tsny_5ma %>%
  autoplot(loansPerCapita) +
  autolayer(tsny_5ma, `5-MA`, color='red') +
  xlab("Year") + ylab("Sales") +
  ggtitle("Loans Per Capita") +
  guides(colour=guide_legend(title="series"))
```

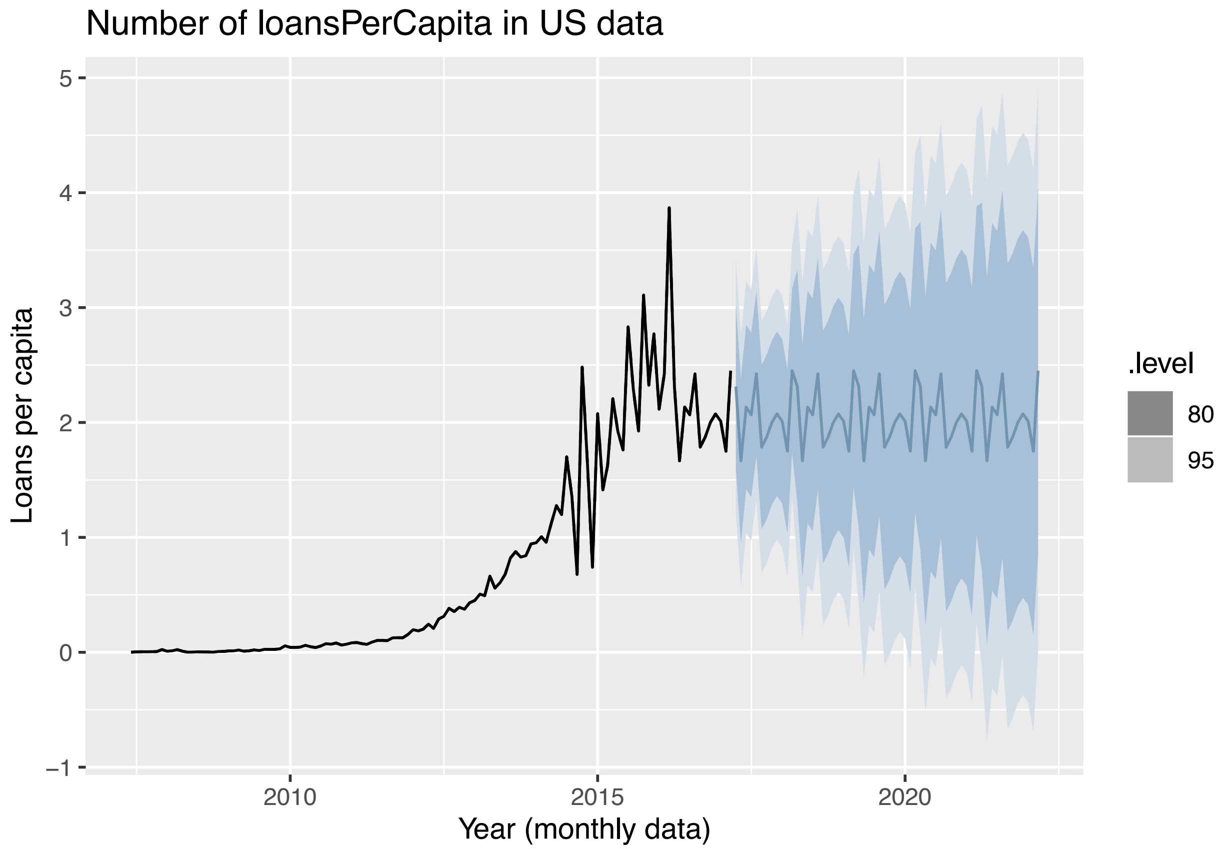
```
## Warning: Removed 4 rows containing missing values (geom_path).
```



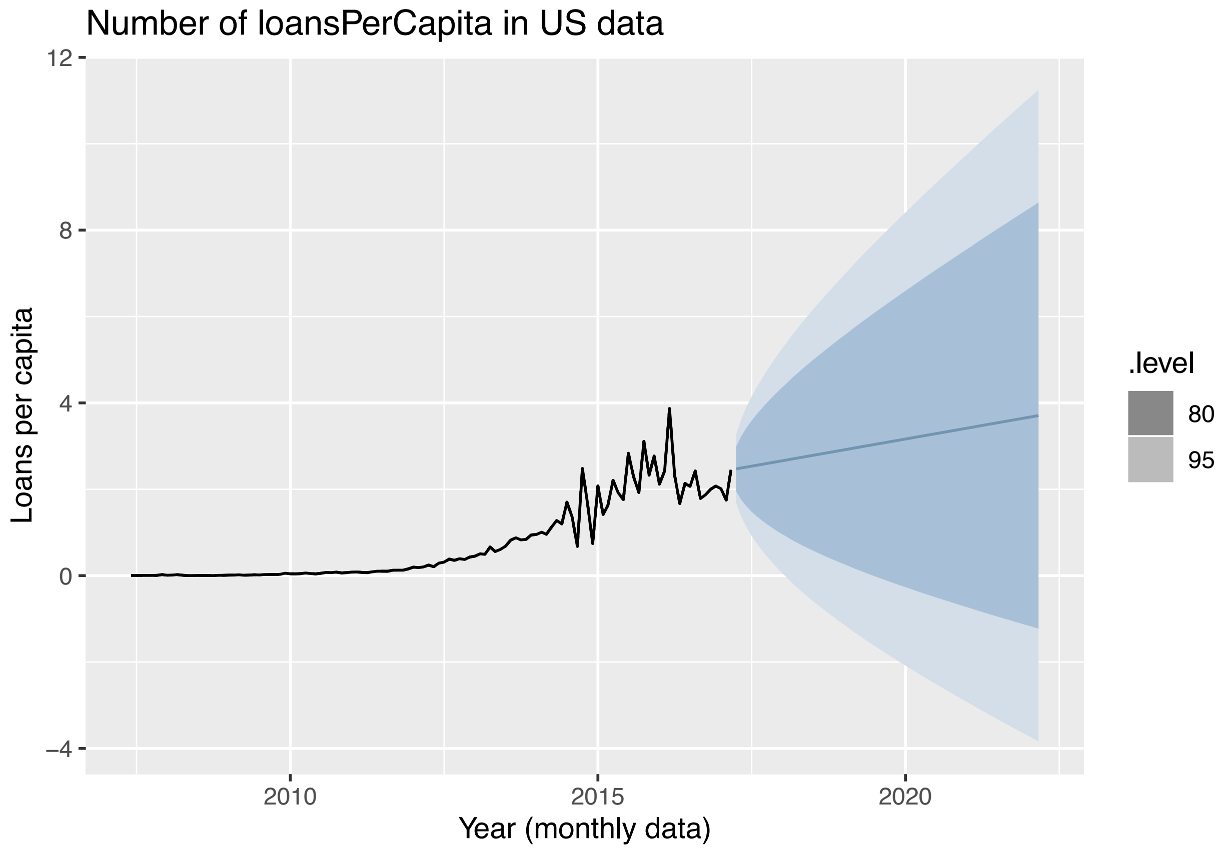
Q3a

```
tsNY <- tsLC %>% filter(state=="NY") %>% mutate(date=yearmonth(date)) %>% as_tsibble(key = state,index=
```

```
fc_NYNaive <-
  tsNY %>%
  model(SNAIVE(loansPerCapita)) %>%
  forecast(h = "5 years") %>%
  autoplot(tsNY, colour = "#769ECB") +
  geom_line(linetype = 'dashed', colour = '#000000') +
  xlab("Year (monthly data)") + ylab("Loans per capita") +
  ggtitle("Number of loansPerCapita in US data")
fc_NYNaive
```



```
fc_NYDrift <-
  tsNY %>%
  model(RW(loansPerCapita ~ drift())) %>%
  forecast(h = "5 years") %>%
  autoplot(tsNY, colour = "#769ECB") +
  geom_line(linetype = 'dashed', colour = '#000000') +
  xlab("Year (monthly data)") + ylab("Loans per capita") +
  ggtitle("Number of loansPerCapita in US data")
fc_NYDrift
```



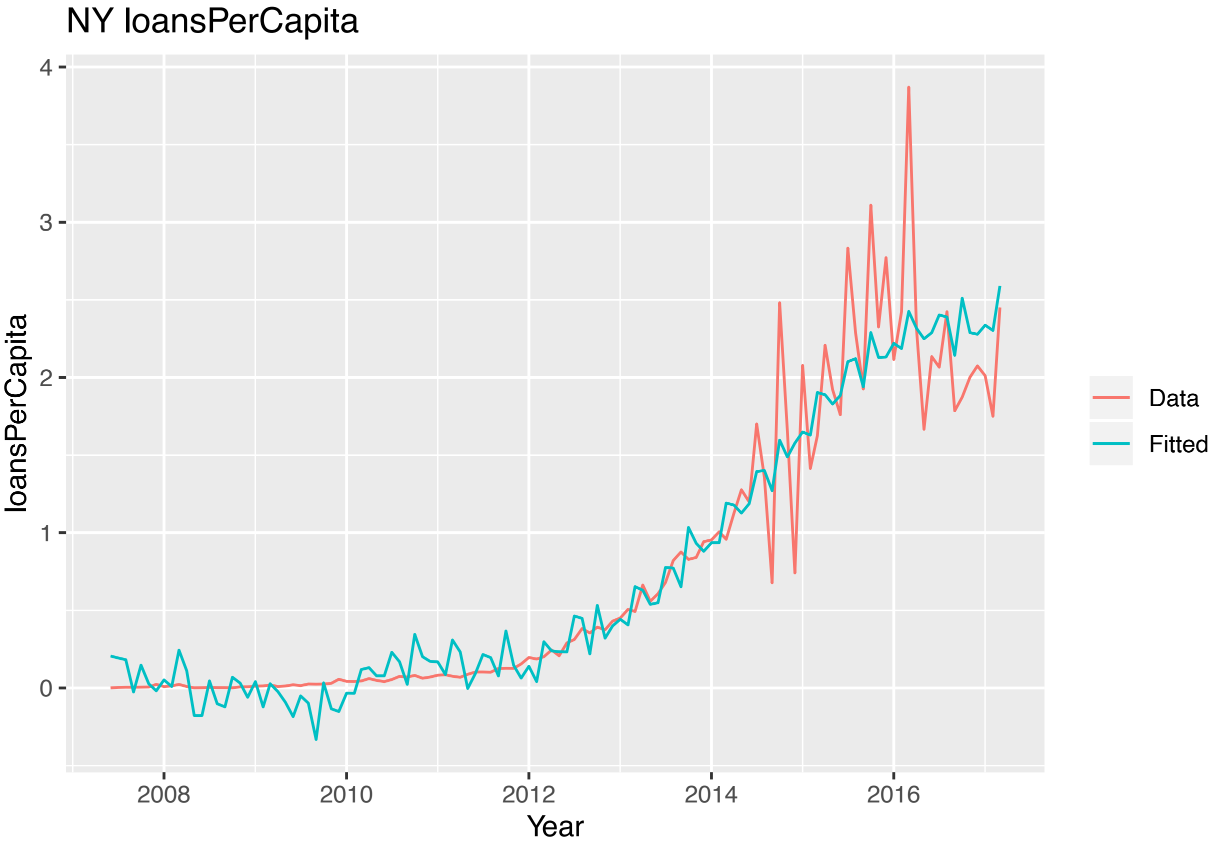
Q3b

```
fit_tsother <-
  tsNY %>%
  model(TSLM(loansPerCapita ~ trend() + season()+NYCPI+NYCondoPriceIdx+avgIntRate))
report(fit_tsother)
```

```
## Series: loansPerCapita
## Model: TSLM
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.836552 -0.135070 -0.005781  0.091424  1.444271
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       5.859105   3.423590   1.711   0.0900 .
## trend()           0.029817   0.005130   5.812 7.08e-08 ***
## season()year2    -0.063744   0.133640  -0.477   0.6344
## season()year3     0.177539   0.134228   1.323   0.1889
## season()year4     0.120125   0.138055   0.870   0.3863
## season()year5     0.034024   0.139149   0.245   0.8073
## season()year6     0.040913   0.136302   0.300   0.7647
## season()year7     0.188790   0.136000   1.388   0.1681
## season()year8     0.159958   0.135956   1.177   0.2421
## season()year9    -0.054590   0.136311  -0.400   0.6896
## season()year10    0.223708   0.134324   1.665   0.0989 .
## season()year11   -0.005926   0.133668  -0.044   0.9647
## season()year12  -0.045744   0.134090  -0.341   0.7337
## NYCPI           -0.013199   0.005182  -2.547   0.0123 *
## NYCondoPriceIdx  0.016023   0.001767   9.070 9.28e-15 ***
## avgIntRate      -0.076552   0.033088  -2.314   0.0227 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2984 on 102 degrees of freedom
## Multiple R-squared: 0.9086,  Adjusted R-squared: 0.8952
## F-statistic:  67.6 on 15 and 102 DF, p-value: < 2.22e-16
```

#Q3c

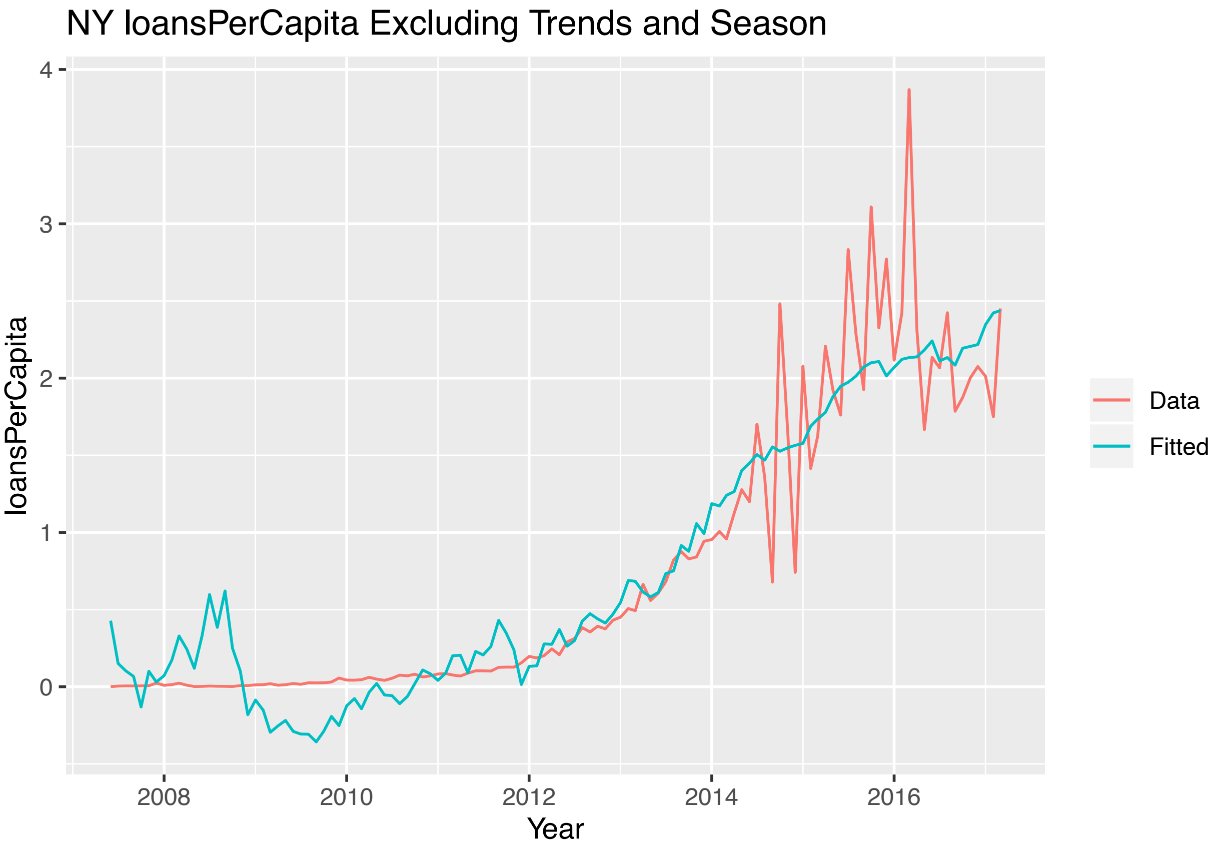
```
fit_tsother_plot <-
  augment(fit_tsother) %>%
  ggplot(aes(x = yearmonth(date))) +
  geom_line(aes(y = loansPerCapita, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  xlab("Year") + ylab("loansPerCapita") +
  ggtitle("NY loansPerCapita") +
  guides(colour=guide_legend(title=NULL))
(fit_tsother_plot)
```



```
fit_other <-
  tsNY %>%
  model(TSLM(loansPerCapita ~ NYCPI+NYCondoPriceIdx+avgIntRate))
report(fit_other)
```

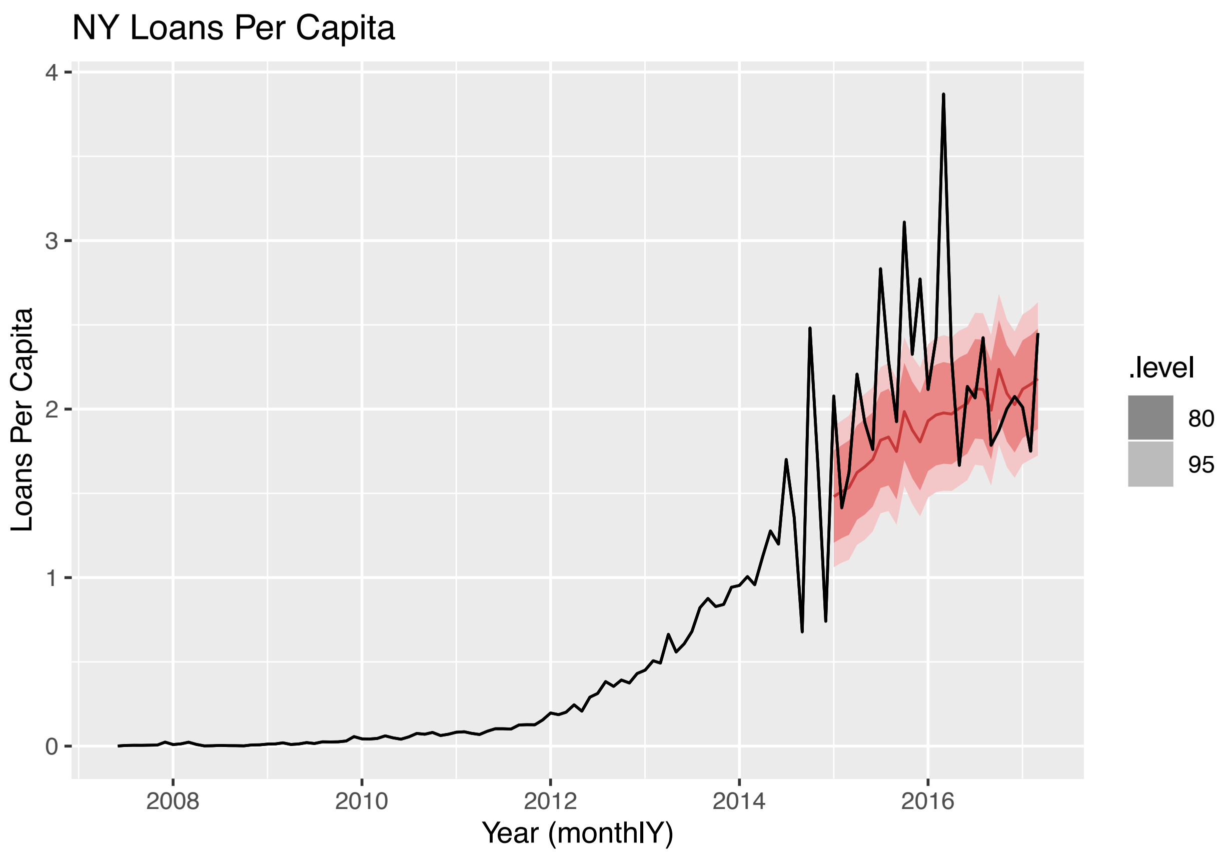
```
## Series: loansPerCapita
## Model: TSLM
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.87651 -0.15528 -0.04117  0.13591  1.73697
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -13.702184   0.703755 -19.470  < 2e-16 ***
## NYCPI             0.015725   0.001489  10.558  < 2e-16 ***
## NYCondoPriceIdx   0.021374   0.001729  12.362  < 2e-16 ***
## avgIntRate       -0.128267   0.035454  -3.618 0.000444 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3398 on 114 degrees of freedom
## Multiple R-squared: 0.8675,  Adjusted R-squared: 0.864
## F-statistic: 248.8 on 3 and 114 DF, p-value: < 2.22e-16
```

```
fit_other_plot <-
  augment(fit_other) %>%
  ggplot(aes(x = yearmonth(date))) +
  geom_line(aes(y = loansPerCapita, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  xlab("Year") + ylab("loansPerCapita") +
  ggtitle("NY loansPerCapita Excluding Trends and Season")  +
  guides(colour=guide_legend(title=NULL))
(fit_other_plot)
```

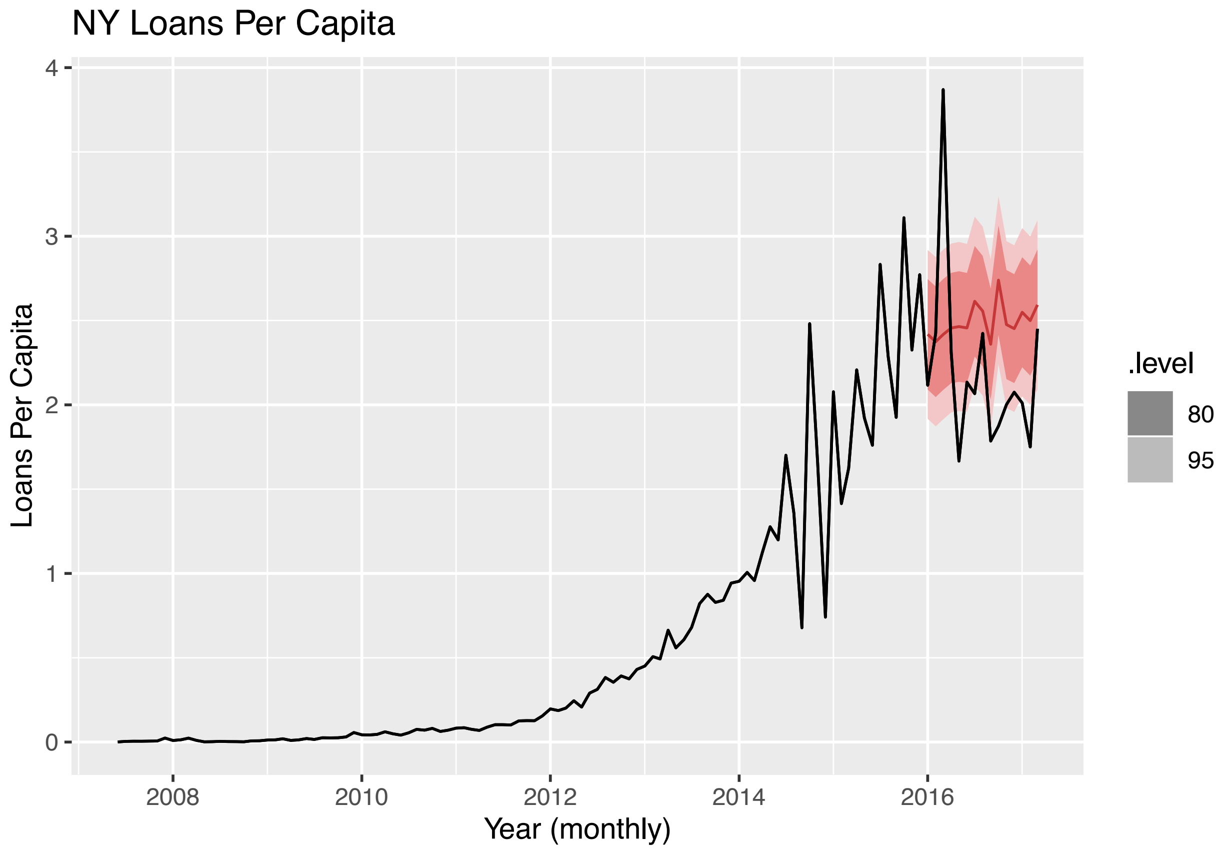


#Q3d

```
plot_14predictive <-
  tsLC %>%filter(state=="NY") %>% filter(date < '2015-01-01')  %>%
  model(TSLM(loansPerCapita ~ trend()+season()+ NYCPI+NYCondoPriceIdx+avgIntRate)) %>%
  forecast(new_data = tsLC %>% filter(date >= '2015-01-01') )%>%
  autoplot(tsLC, colour = "#960A0A") +
  geom_line(colour = '#000000') +
  xlab("Year (monthlY)") + ylab("Loans Per Capita") +
  ggtitle("NY Loans Per Capita")
plot_14predictive
```

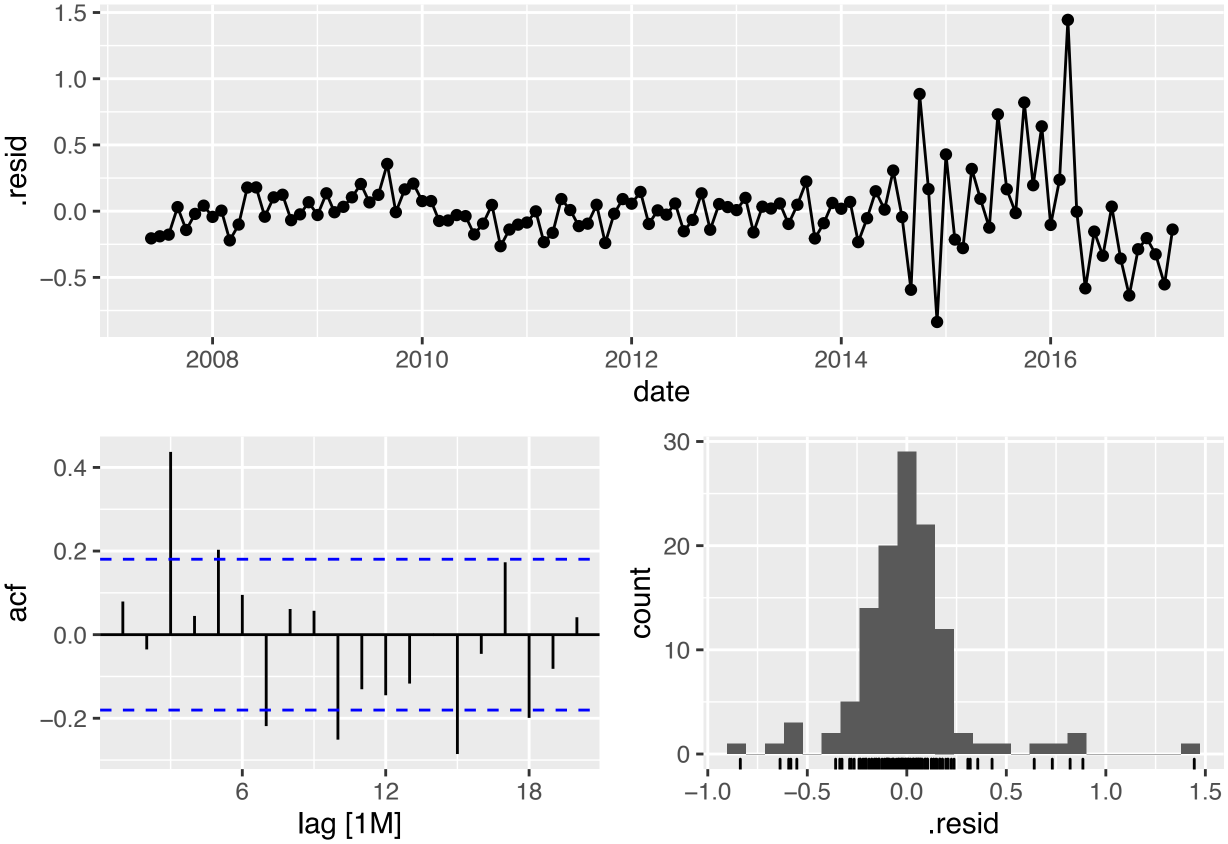


```
plot_15predictive <-
  tsLC %>%filter(state=="NY") %>%
  filter(date < '2016-01-01') %>%
  model(TSLM(loansPerCapita ~ trend()+season()+NYCPI+NYCondoPriceIdx+avgIntRate)) %>%
  forecast(new_data = tsLC %>% filter(date >= ('2016-01-01') ))%>%
  autoplot(tsLC, colour = "#960A0A") +
  geom_line(colour = '#000000') +
  xlab("Year (monthly)") + ylab("Loans Per Capita") +
  ggtitle("NY Loans Per Capita")
plot_15predictive
```



#Q3e

```r
fit_tsother %>% gg_tsresiduals()
```



#Q3f

```
fit_NYARIMA_3f <- tsNY %>%
  model(fitArima = ARIMA(loansPerCapita ~ pdq(1:3,0:1, 0:2) + PDQ(0:2, 0:1, 0:2)+
          NYCPI+NYCondoPriceIdx+avgIntRate,stepwise = FALSE, approximation = FALSE))
report(fit_NYARIMA_3f)
```

```
## Series: loansPerCapita
## Model: LM w/ ARIMA(3,0,2)(0,0,1)[12] errors
##
## Coefficients:
##           ar1     ar2     ar3     ma1      ma2    sma1    NYCPI
##       -0.0080  0.4253  0.5568  0.3544  -0.4387  0.2675  -0.0024
## s.e.   0.0925  0.0791  0.0876  0.0946   0.0843  0.1103   0.0024
##       NYCondoPriceIdx  avgIntRate
##                0.0115     -0.0086
## s.e.           0.0067      0.0292
##
## sigma^2 estimated as 0.06819:  log likelihood=-6.88
## AIC=33.76   AICc=35.81   BIC=61.46
```

#Q3g #KPSS before

```
tsNY %>%
  features(loansPerCapita,unitroot_kpss)
```

```
## # A tibble: 1 x 3
##   state kpss_stat kpss_pvalue
##   <chr>     <dbl>       <dbl>
## 1 NY          2.09        0.01
```

```
tsNY %>%
  features(loansPerCapita, unitroot_nsdiffs)
```

```
## # A tibble: 1 x 2
##   state nsdiffs
##   <chr>   <int>
## 1 NY          0
```

```
tsNY %>%
  features(loansPerCapita, unitroot_ndiffs)
```

```
## # A tibble: 1 x 2
##   state ndiffs
##   <chr>  <int>
## 1 NY         1
```

#KPSS after

```
tsNY %>%
  mutate(Lpc_diff=difference(loansPerCapita)) %>%
  features(Lpc_diff,unitroot_kpss)
```

```
## # A tibble: 1 x 3
##   state kpss_stat kpss_pvalue
##   <chr>     <dbl>       <dbl>
## 1 NY        0.129         0.1
```

```
tsNY %>%
  mutate(Lpc_diff=difference(loansPerCapita)) %>%
  features(Lpc_diff, unitroot_ndiffs)
```

```
## # A tibble: 1 x 2
##   state ndiffs
##   <chr>  <int>
## 1 NY         0
```

#Q3h

```
fit_NYARIMA <- tsNY %>%
  model(fitArima = ARIMA(loansPerCapita ~ pdq(,1,) + PDQ(,0,)+
          NYCPI+NYCondoPriceIdx+avgIntRate,stepwise = FALSE, approximation = FALSE))
report(fit_NYARIMA)
```

```
## Series: loansPerCapita
## Model: LM w/ ARIMA(0,1,4) errors
##
## Coefficients:
##           ma1      ma2     ma3      ma4    NYCPI  NYCondoPriceIdx  avgIntRate
##       -0.8095  -0.2965  0.8164  -0.3097  -0.0034           0.0095     -0.0099
## s.e.   0.0927   0.0899  0.0762   0.0992   0.0044           0.0066      0.0279
##       intercept
##          0.0185
## s.e.     0.0103
##
## sigma^2 estimated as 0.06436:  log likelihood=-2.58
## AIC=23.16   AICc=24.84   BIC=48.02
```

#Q4a

```r
set.seed(333)
ts_NYTrainMarch <- tsLC %>% filter(state=='NY') %>% filter(date < '2016-03-01')
ts_NYTestMarch <- tsLC %>% filter(state=='NY') %>% filter(date >= '2016-03-01')
```

```r
ts_NYFitAll <-
  ts_NYTrainMarch %>%
  model(
    TSts = TSLM(loansPerCapita ~ trend() + season()),
    TSother = TSLM(loansPerCapita ~ trend()+season()+ NYCPI+NYCondoPriceIdx+avgIntRate),
    ARIMA_model = ARIMA(loansPerCapita)
  )

ts_NYPredictMarch <-
  ts_NYFitAll %>%
  forecast(new_data = ts_NYTestMarch)
```

```r
forecast::accuracy(ts_NYPredictMarch, ts_NYTestMarch)
```

```
## # A tibble: 3 x 10
##   .model      state .type     ME  RMSE   MAE   MPE  MAPE  MASE   ACF1
##   <chr>       <chr> <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 ARIMA_model NY    Test  -0.791 0.973 0.907 -41.8  45.0   NaN 0.291
## 2 TSother     NY    Test  -0.306 0.642 0.532 -19.1  24.9   NaN 0.0932
## 3 TSts        NY    Test   0.341 0.703 0.407  11.4  15.1   NaN 0.137
```

```r
# fit_NYARIMA <- tsNY %>%
#   model(fitArima = ARIMA(loansPerCapita ~ pdq(,1,) + PDQ(,0,)+
#           NYCPI+NYCondoPriceIdx+avgIntRate,stepwise = FALSE, approximation = FALSE))

fit_NYPredictMarch <-
  fit_NYARIMA_3f %>%
  forecast(new_data = ts_NYTestMarch)
```

```r
forecast::accuracy(fit_NYPredictMarch, ts_NYTestMarch)
```

```
## # A tibble: 1 x 10
##   .model   state .type    ME  RMSE   MAE   MPE  MAPE  MASE   ACF1
##   <chr>    <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 fitArima NY    Test  0.424 0.656 0.467  16.1  18.7   NaN 0.0860
```

#Q4b

```r
set.seed(333)
ts_NYTrainApril <- tsLC %>% filter(state=='NY') %>% filter(date < '2016-04-01')
ts_NYTestApril <- tsLC %>% filter(state=='NY') %>% filter(date >= '2016-04-01')
```

```r
ts_NYFitAllApril <-
  ts_NYTrainApril %>%
  model(
    TSts = TSLM(loansPerCapita ~ trend() + season()),
    TSother = TSLM(loansPerCapita ~ trend()+season()+ NYCPI+NYCondoPriceIdx+avgIntRate),
    ARIMA_model = ARIMA(loansPerCapita)
  )

ts_NYPredictApril <-
```

```
  ts_NYFitAllApril %>%
  forecast(new_data = ts_NYTestApril)
```

```
forecast::accuracy(ts_NYPredictApril, ts_NYTestApril)
```

```
## # A tibble: 3 x 10
##   .model      state .type      ME  RMSE   MAE    MPE  MAPE  MASE   ACF1
##   <chr>       <chr> <chr>   <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl>  <dbl>
## 1 ARIMA_model NY    Test  -1.53   1.64  1.53  -76.5  76.5    NaN  0.188
## 2 TSother     NY    Test  -0.559  0.604 0.559 -28.9  28.9    NaN -0.380
## 3 TSts        NY    Test   0.0944 0.268 0.208   3.43  9.73   NaN -0.185
```

```
fit_NYPredictApril <-
  fit_NYARIMA_3f %>%
  forecast(new_data = ts_NYTestApril)
```

```
forecast::accuracy(fit_NYPredictApril, ts_NYTestApril)
```

```
## # A tibble: 1 x 10
##   .model   state .type    ME  RMSE   MAE   MPE  MAPE  MASE   ACF1
##   <chr>    <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 fitArima NY    Test  0.282 0.338 0.282  13.0  13.0   NaN -0.315
```