# Uzbekistan Health Reforms: Within Uzbekistan Analysis

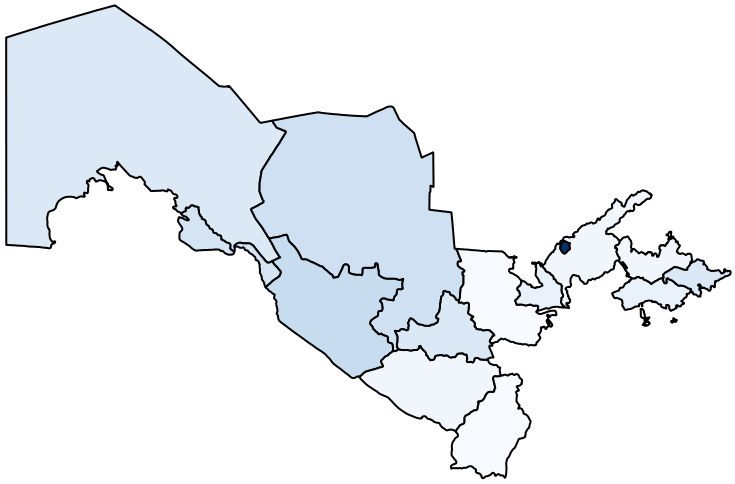
Michelle Schultze

2023-01-22

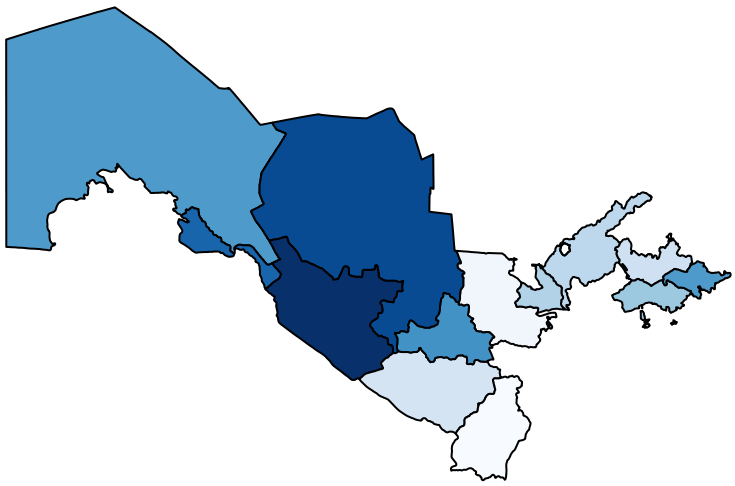
**Doctors Per Capita**

```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(doctors_per_capita == as.numeric(doctors_per_capita)) %>%
  select(doctors_per_capita)

map_1@data$doctors_per_capita <- as.numeric(unlist(test))

library(RColorBrewer)
my_colors <- brewer.pal(9, "Blues")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(test$doctors_per_capita, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Doctors per Capita, 2019")
```

## Doctors per Capita, 2019

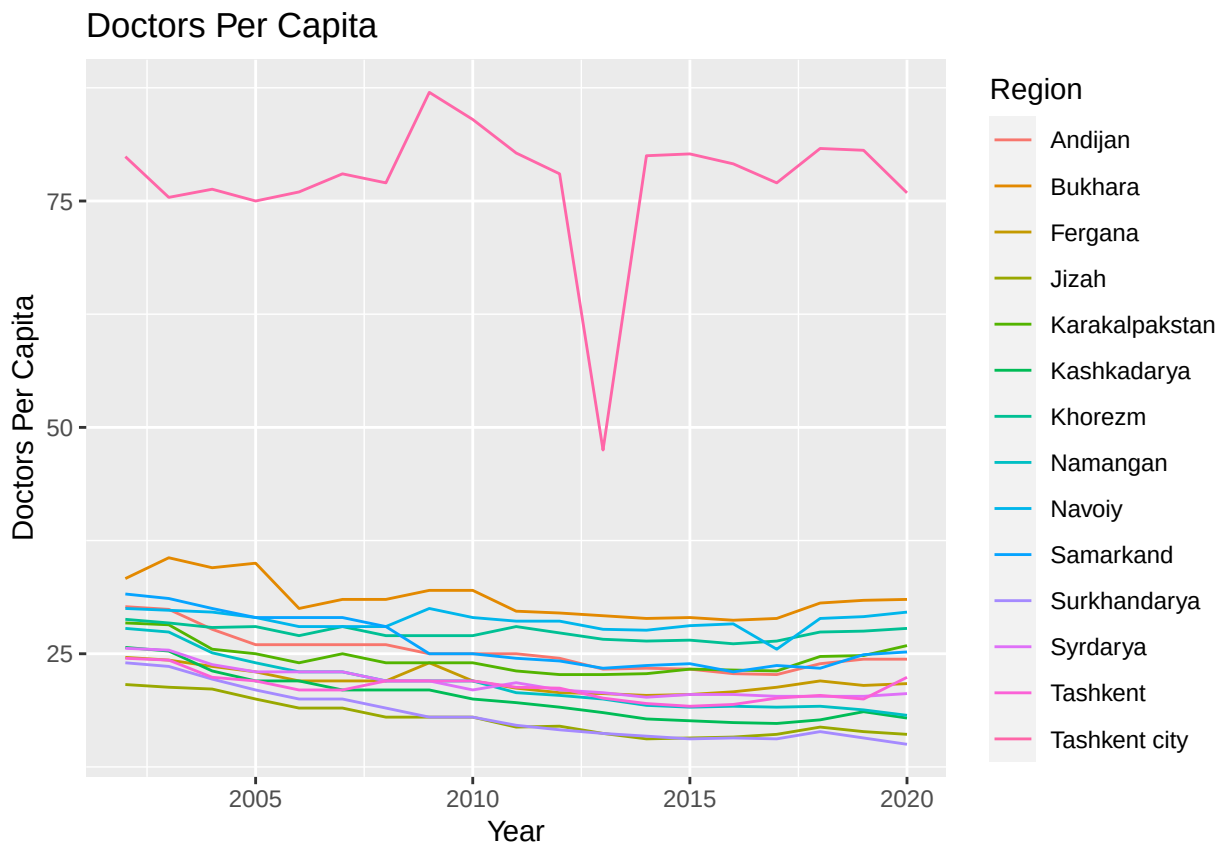


```
## excluding the outlier, tashkent city

test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(doctors_per_capita == as.numeric(doctors_per_capita)) %>%
  mutate(doctors_per_capita = ifelse(Region == "Tashkent city", NA, doctors_per_capita)) %>%
  select(doctors_per_capita)

map_1@data$doctors_per_capita <- as.numeric(unlist(test))

library(RColorBrewer)
my_colors <- brewer.pal(9, "Blues")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(test$doctors_per_capita, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Doctors per Capita, 2019 (excl. Tashkent city)")
```

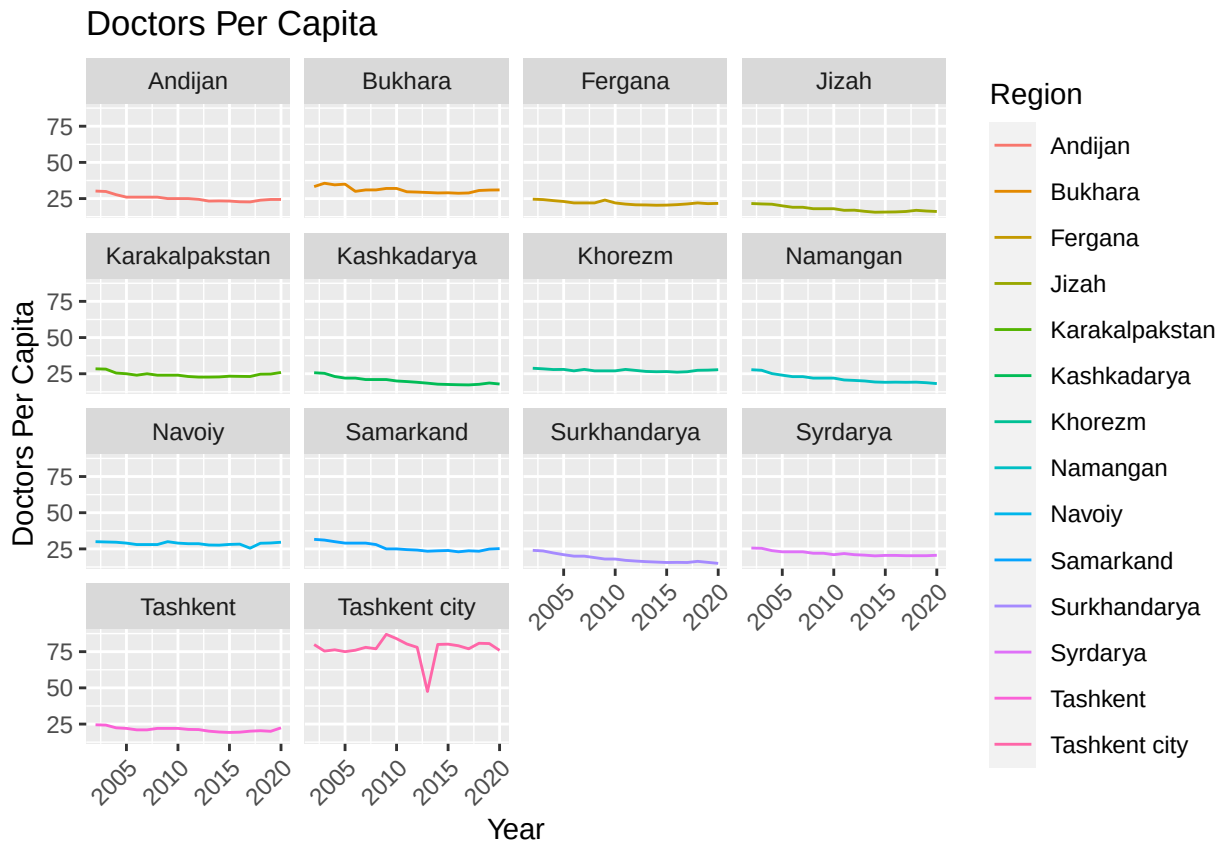



### Charts

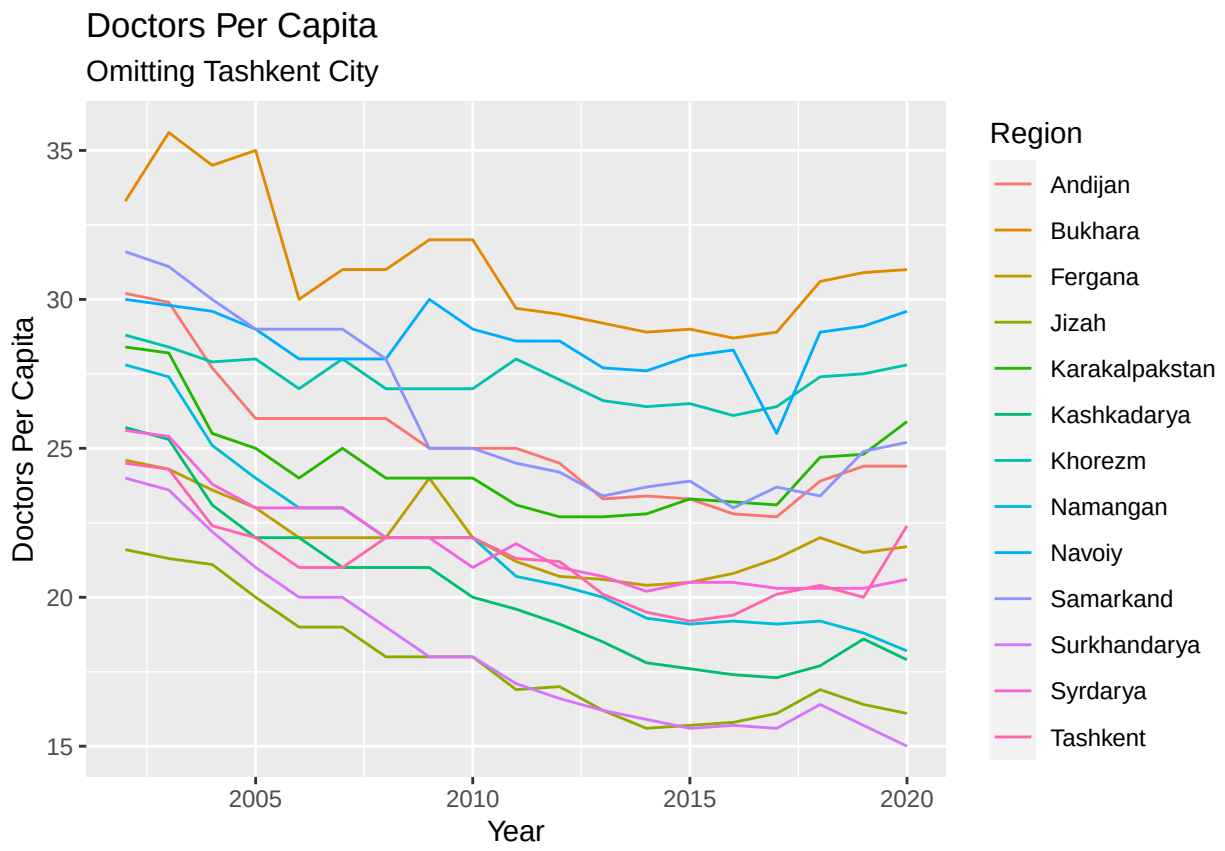
```
yrs <- uz_data %>%
  filter(doctors_per_capita != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- yrs$year[2]



## year range: 2002-2020 automatically stored for future use

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = doctors_per_capita, color = Region)) +
  geom_line() +
  labs(title = "Doctors Per Capita",
       x = "Year", y = "Doctors Per Capita")
```



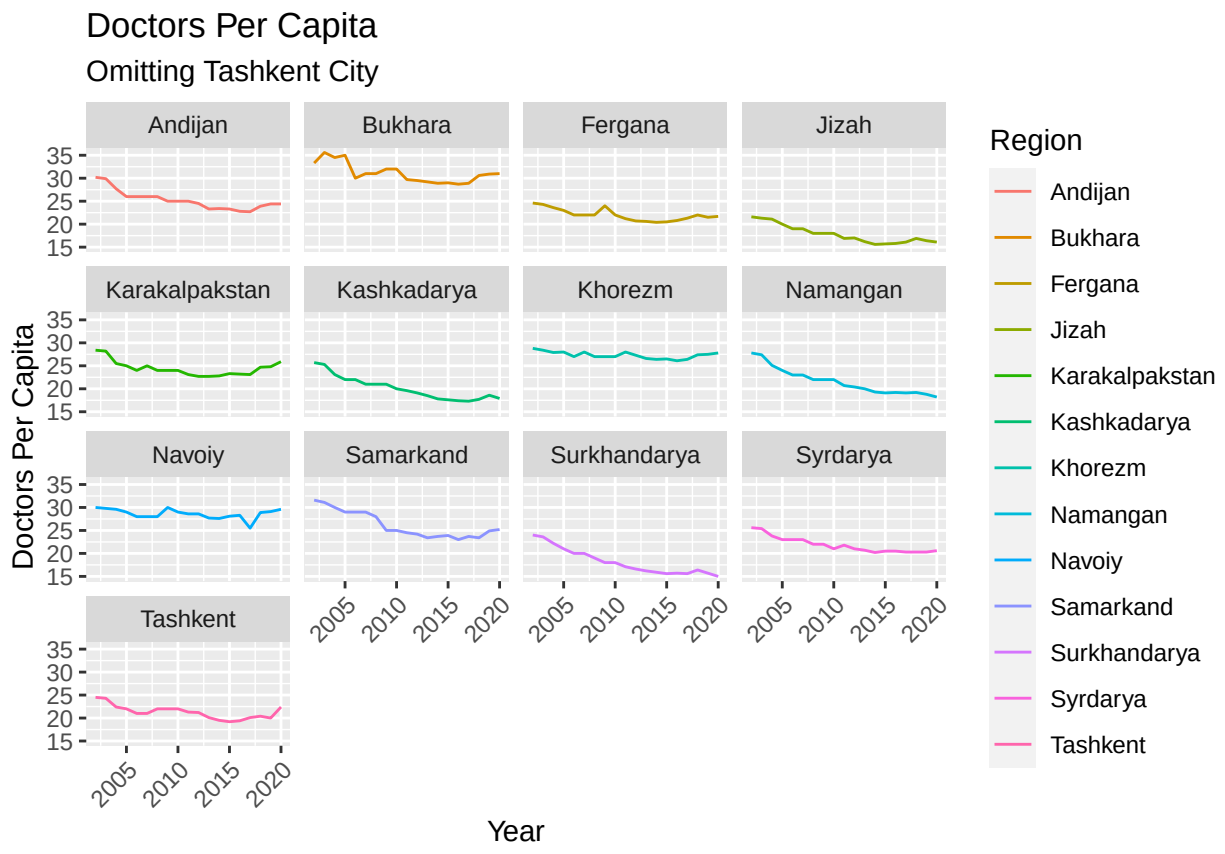

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = doctors_per_capita, color = Region)) +
  geom_line() +
  facet_wrap(~ Region) +
  labs(title = "Doctors Per Capita",
     x = "Year", y = "Doctors Per Capita") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```




```
# Exclude the outlier, Tashkent City
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region != "Tashkent city") %>%
  ggplot(aes(x = year, y = doctors_per_capita, color = Region)) +
  geom_line() +
  labs(title = "Doctors Per Capita",
       x = "Year", y = "Doctors Per Capita",
       subtitle = "Omitting Tashkent City")
```

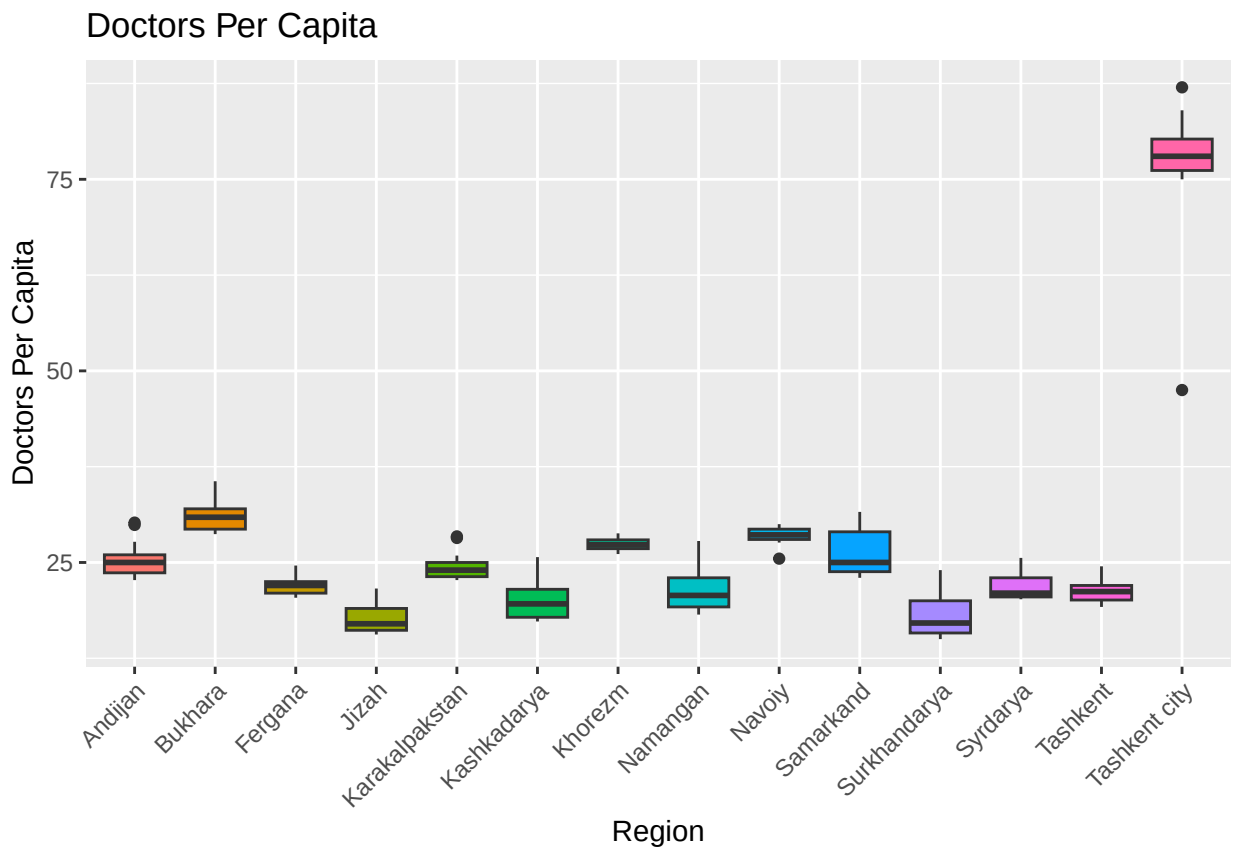



```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region != "Tashkent city") %>%
  ggplot(aes(x = year, y = doctors_per_capita, color = Region)) +
  geom_line() +
  facet_wrap(~ Region) +
  labs(title = "Doctors Per Capita",
       x = "Year", y = "Doctors Per Capita",
       subtitle = "Omitting Tashkent City") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```



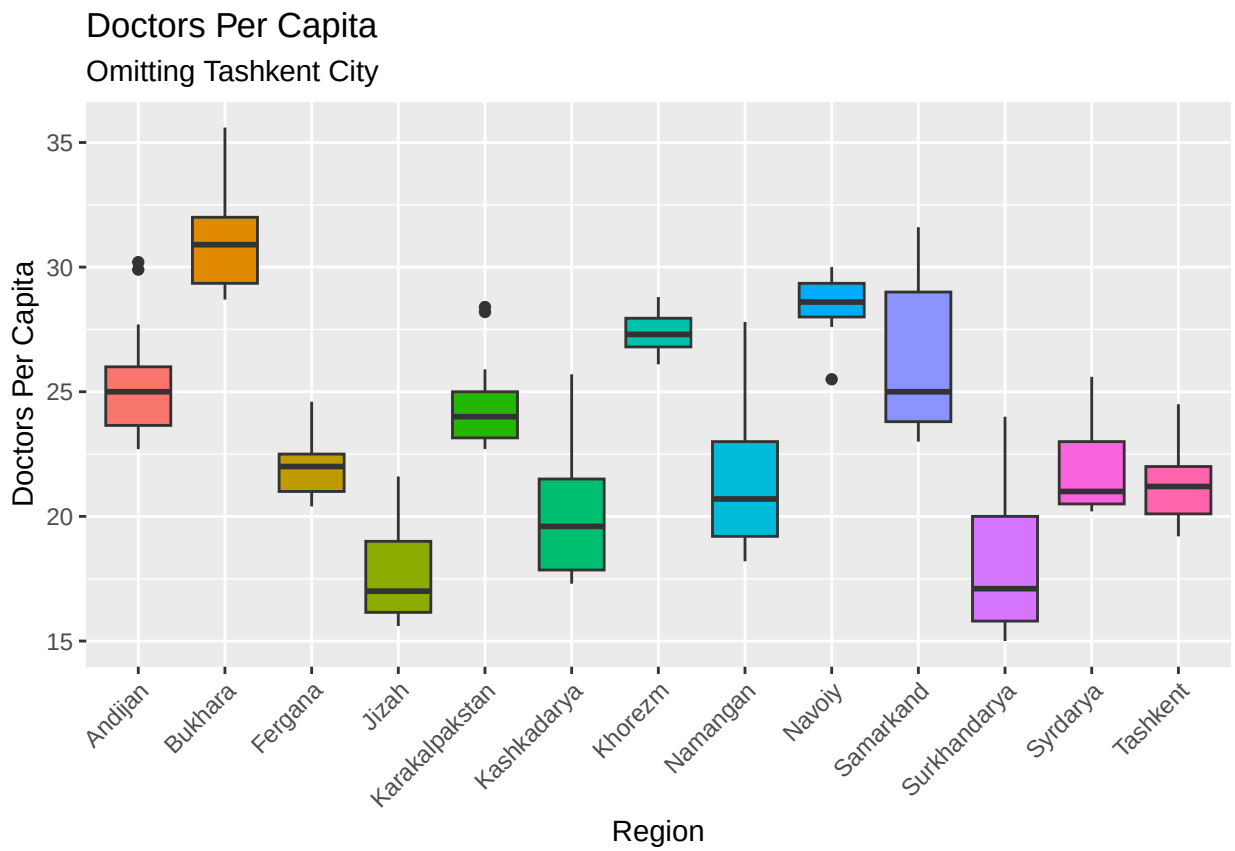

```
# By Region

uz_data %>%
  ggplot(aes(x = Region, y = doctors_per_capita)) +
  geom_boxplot(aes(fill = Region), show.legend = FALSE) +
  labs(title = "Doctors Per Capita",
     x = "Region", y = "Doctors Per Capita") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

```
## Warning: Removed 196 rows containing non-finite values (‘stat_boxplot()‘).
```

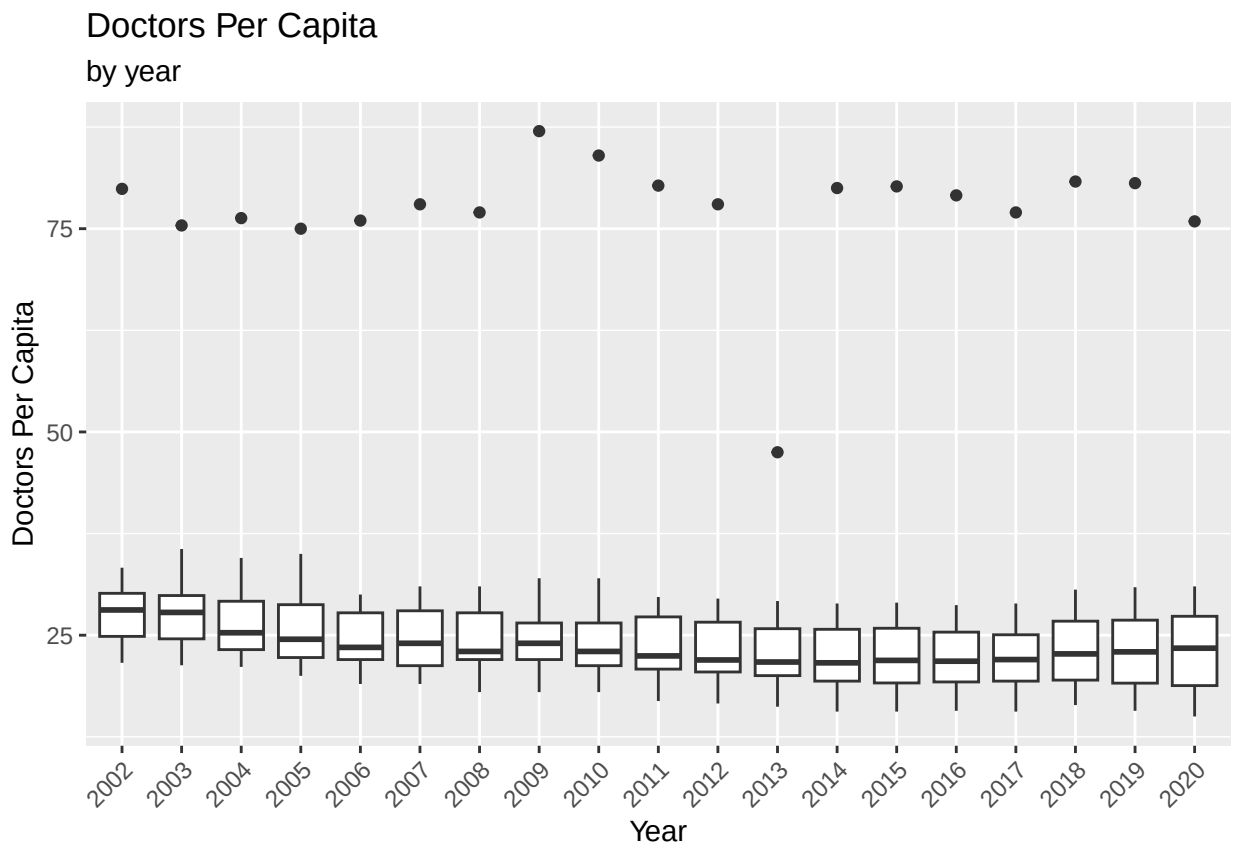



```
uz_data %>%
  filter(Region != "Tashkent city") %>%
  ggplot(aes(x = Region, y = doctors_per_capita)) +
  geom_boxplot(aes(fill = Region), show.legend = FALSE) +
  labs(title = "Doctors Per Capita", subtitle = "Omitting Tashkent City",
     x = "Region", y = "Doctors Per Capita") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

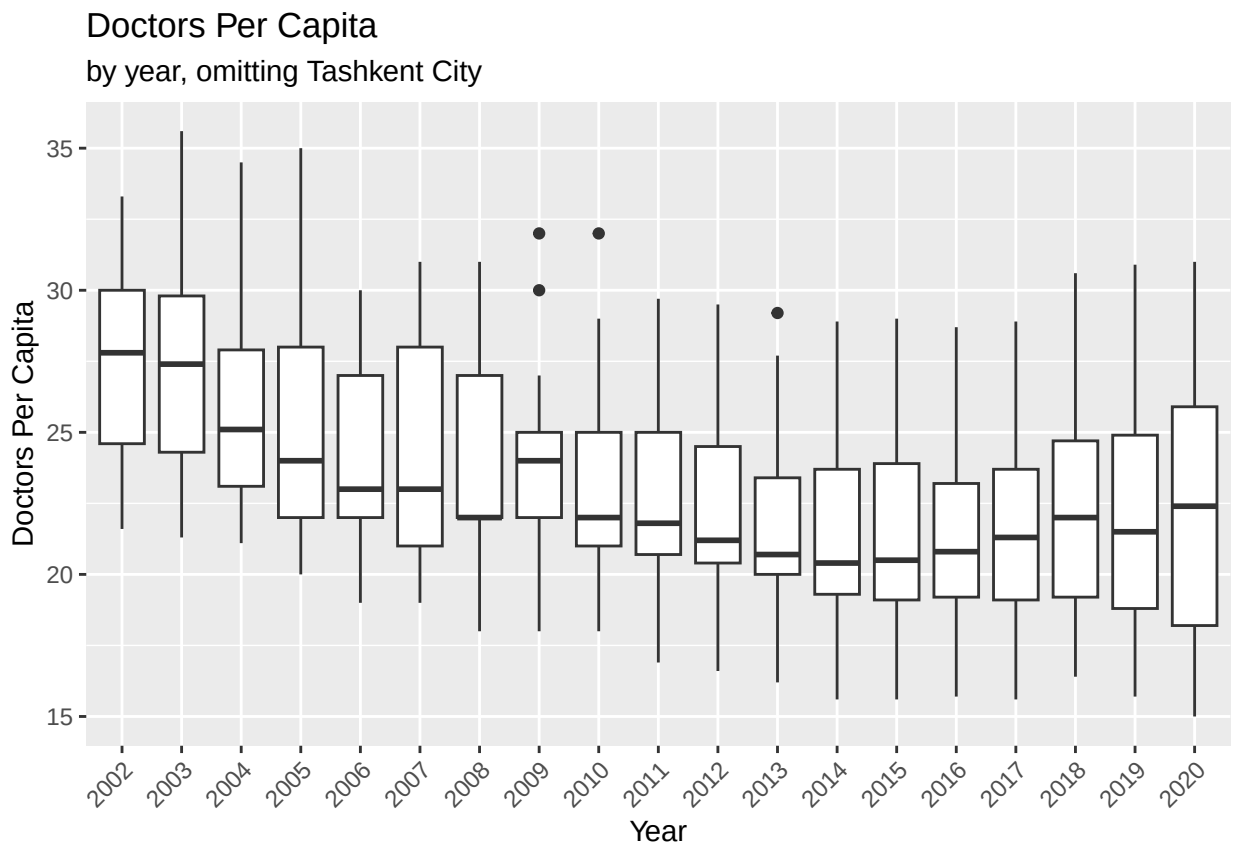
```
## Warning: Removed 182 rows containing non-finite values (‘stat_boxplot()‘).
```



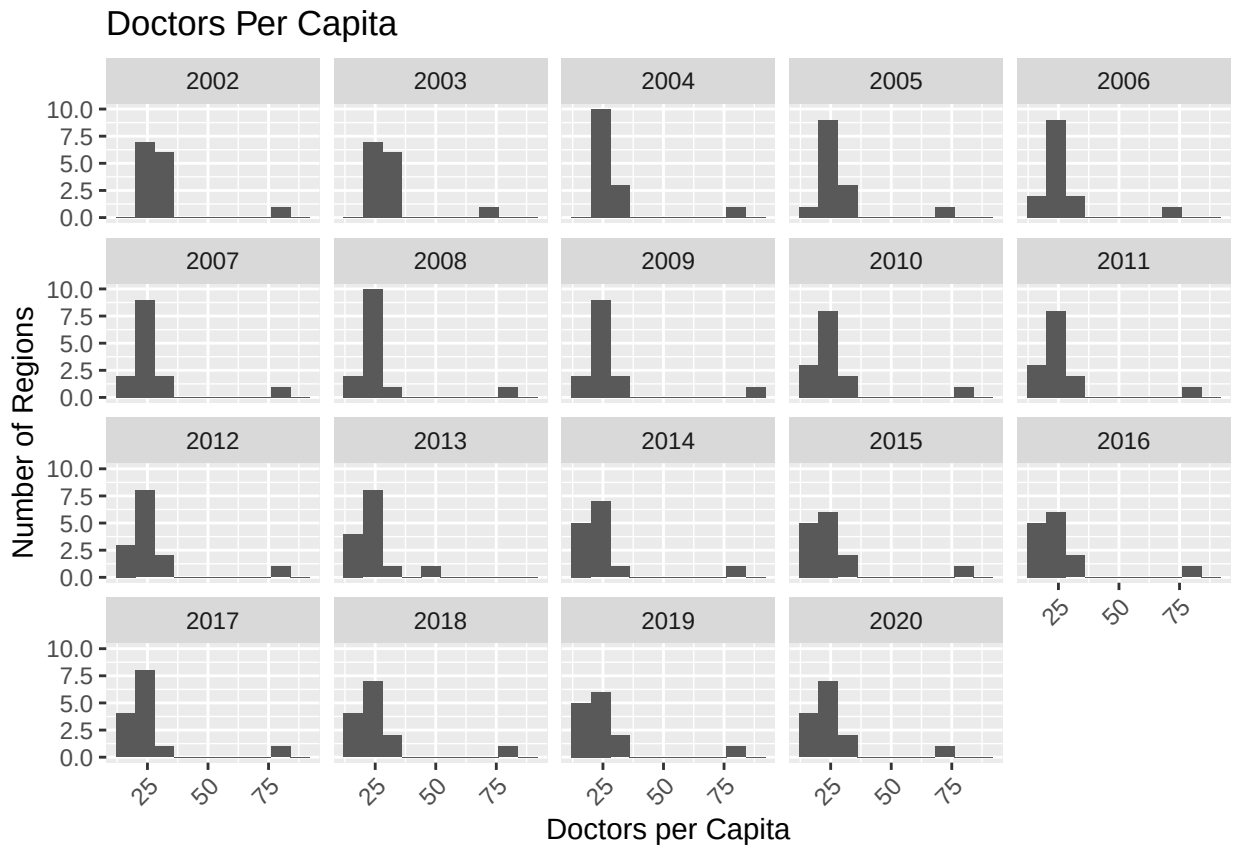

```
# By year

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = as.character(year), y = doctors_per_capita)) +
  geom_boxplot() +
  labs(title = "Doctors Per Capita", subtitle = "by year",
     x = "Year", y = "Doctors Per Capita") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```



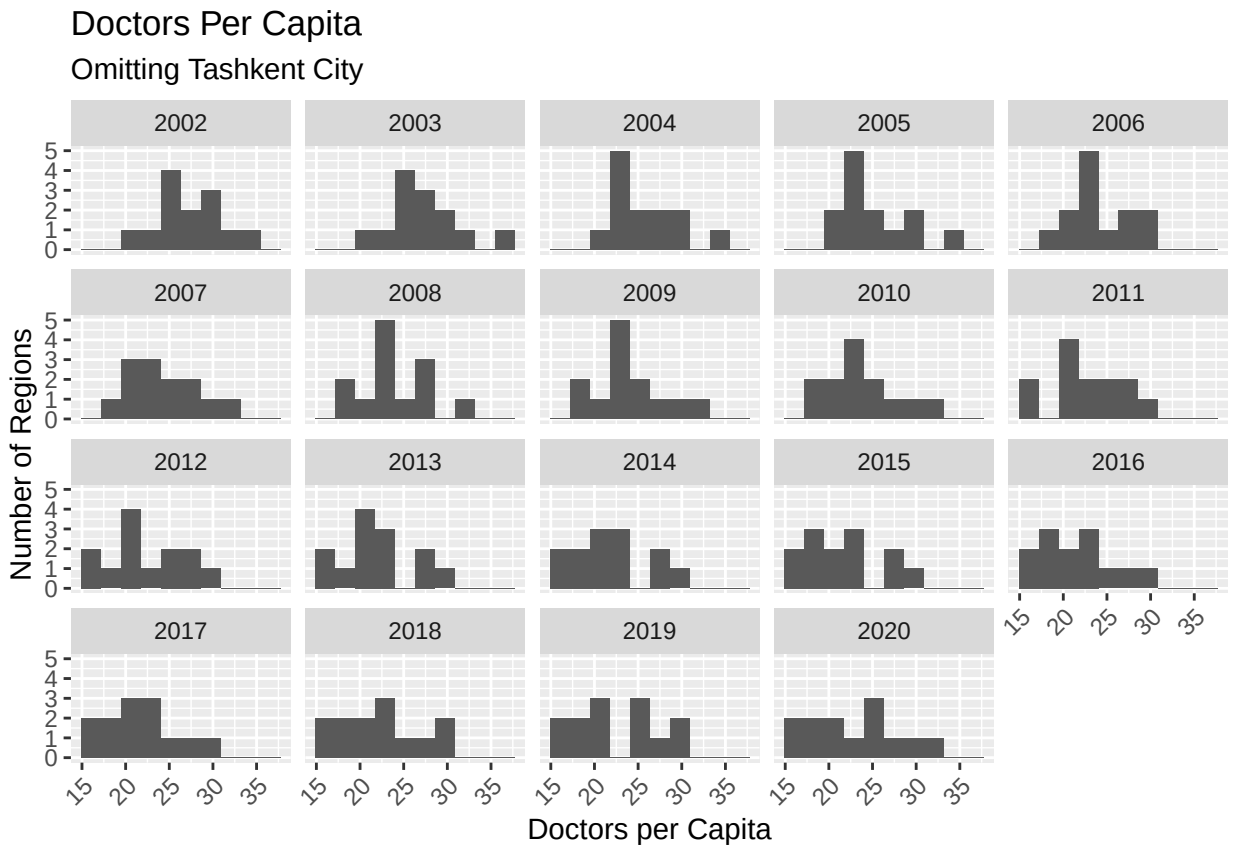

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region != "Tashkent city") %>%
  ggplot(aes(x = as.character(year), y = doctors_per_capita)) +
  geom_boxplot() +
  labs(title = "Doctors Per Capita", subtitle = "by year, omitting Tashkent City",
     x = "Year", y = "Doctors Per Capita") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```




```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = doctors_per_capita)) +
  geom_histogram(bins = 10) +
  facet_wrap(~ year) +
  labs(title = "Doctors Per Capita", x = "Doctors per Capita",
       y = "Number of Regions") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

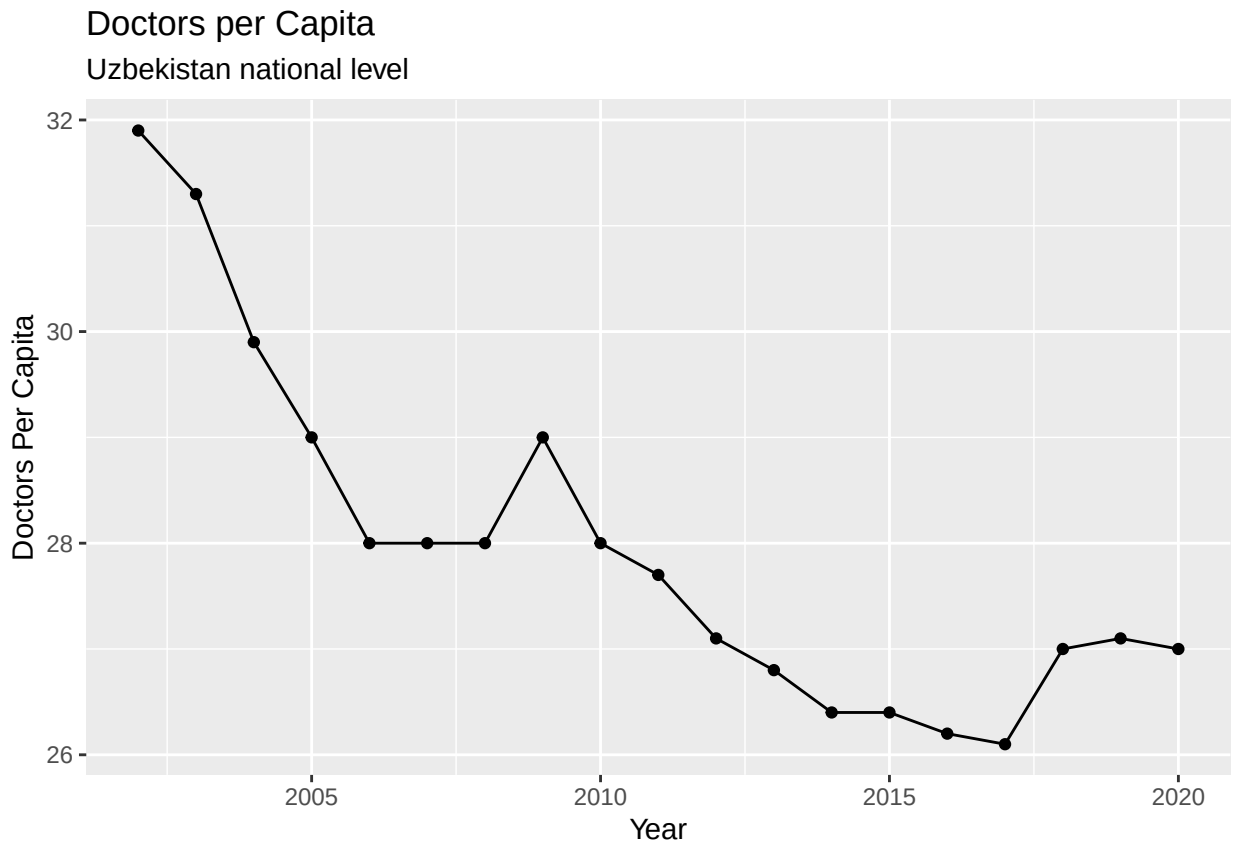



```
# Exclude the outlier, Tashkent City

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region != "Tashkent city") %>%
  ggplot(aes(x = doctors_per_capita)) +
  geom_histogram(bins = 10) +
  facet_wrap(~ year) +
  labs(title = "Doctors Per Capita", x = "Doctors per Capita",
       y = "Number of Regions", subtitle = "Omitting Tashkent City") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```



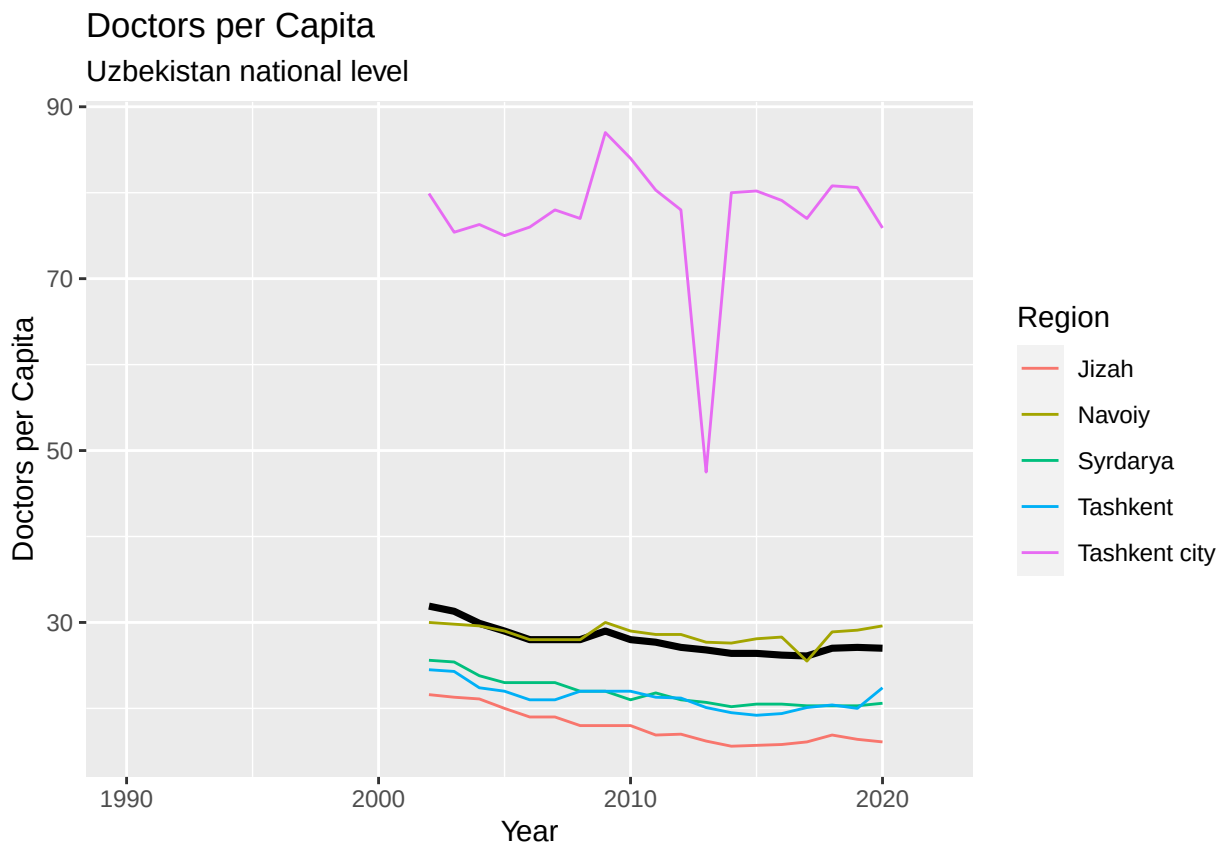

**Looking at the national level**

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = doctors_per_capita)) +
  geom_line() + geom_point() +
  labs(title = "Doctors per Capita", subtitle = "Uzbekistan national level",
       x = "Year", y = "Doctors Per Capita")
```




```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = doctors_per_capita)) +
  geom_line(linewidth=1.25) +
  geom_line(data = filter(uz_data, Region == "Tashkent city" | Region == "Navoiy" | Region == "Jizah" |
  labs(title = "Doctors per Capita", subtitle = "Uzbekistan national level", x = "Year", y = "Doctors p
```

```
## Warning: Removed 70 rows containing missing values (`geom_line()`).
```




**Descriptives**

Jizah and Surkhandarya were the only regions to have less than or equal to 1.7 doctors per 1000 people (global average) in 2019. source: https://data.worldbank.org/indicator/SH.MED.PHYS.ZS?end=2019&start=1963

```
uz_data %>%
  filter(doctors_per_capita <= 17,
         year == 2019) %>%
  select(Region, year, doctors_per_capita)
```

```
## # A tibble: 2 x 3
##   Region        year doctors_per_capita
##   <chr>        <dbl>              <dbl>
## 1 Jizah         2019               16.4
## 2 Surkhandarya  2019               15.7
```

Doctors per capita in Uzbekistan in 2002 was 31.9 doctors per 10000 people. Two regions exhibited more than the national average for doctors per capita in 2002.

```
uz_doc_2002 <- nat_uz_data %>%
  filter(year == 2002) %>%
  select(doctors_per_capita)
uz_doc_2002 <- uz_doc_2002$doctors_per_capita
uz_doc_2002
```

```
## [1] 31.9
```

```
uz_data %>%
  filter(doctors_per_capita > uz_doc_2002,
         year == 2002) %>%
  select(Region, year, doctors_per_capita)
```

```
## # A tibble: 2 x 3
##   Region         year doctors_per_capita
##   <chr>         <dbl>              <dbl>
## 1 Bukhara        2002               33.3
## 2 Tashkent city  2002               79.9
```

Doctors per capita in Uzbekistan in 2020 was 27 doctors per 10000 people. Four regions had over than the national average for doctors per capita in 2020.

```
uz_doc_2020 <- nat_uz_data %>%
  filter(year == 2020) %>%
  select(doctors_per_capita)
uz_doc_2020 <- uz_doc_2020$doctors_per_capita
uz_doc_2020
```

```
## [1] 27
```

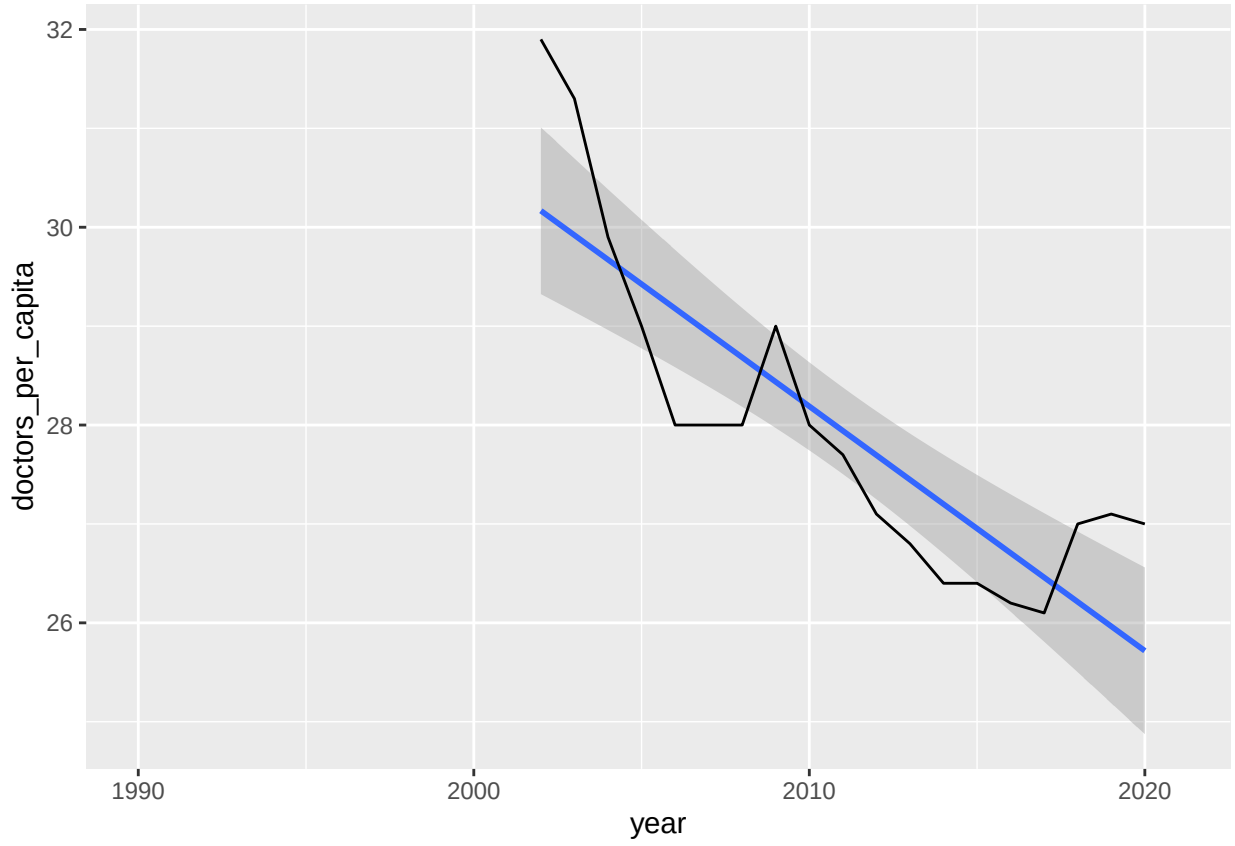
```
uz_data %>%
  filter(doctors_per_capita > uz_doc_2020,
         year == 2020) %>%
  select(Region, year, doctors_per_capita)
```

```
## # A tibble: 4 x 3
##   Region         year doctors_per_capita
##   <chr>         <dbl>              <dbl>
## 1 Bukhara        2020               31
## 2 Khorezm        2020               27.8
## 3 Navoiy         2020               29.6
## 4 Tashkent city  2020               75.9
```

Simple regressions

```
nat_uz_data %>%
  ggplot(aes(x=year,y=doctors_per_capita)) +
  geom_smooth(method = "lm") +
  geom_line()
```

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

## Warning: Removed 13 rows containing non-finite values (`stat_smooth()`).

## Warning: Removed 13 rows containing missing values (`geom_line()`).
```




```
uz_doctorsmodel <- linear_reg() %>%
  set_engine("lm") %>%
  fit(doctors_per_capita ~ year, data = nat_uz_data)
tidy(uz_doctorsmodel)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic    p.value
##   <chr>          <dbl>     <dbl>     <dbl>      <dbl>
## 1 (Intercept)  525.      76.3         6.88 0.00000265
## 2 year          -0.247    0.0379     -6.52 0.00000526
```

```
glance(uz_doctorsmodel)$p.value < 0.01
```

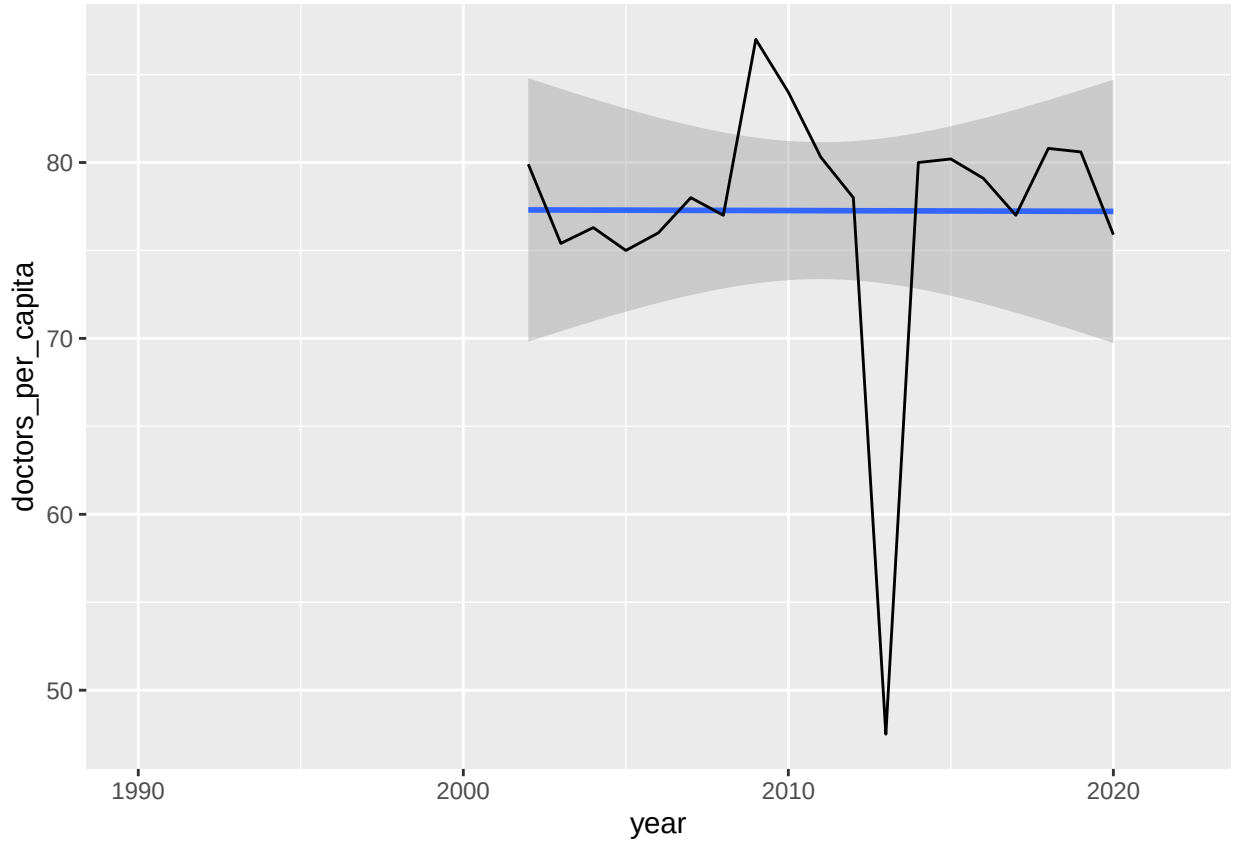
```
## value
##  TRUE
```

```
uz_data %>%
  filter(Region == "Tashkent city") %>%
  ggplot(aes(x=year,y=doctors_per_capita)) +
  geom_smooth(method = "lm") +
  geom_line()
```

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

```
## Warning: Removed 14 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 14 rows containing missing values (`geom_line()`).
```




```
doctorsmodel <- linear_reg() %>%
  set_engine("lm") %>%
  fit(doctors_per_capita ~ year, data = filter(uz_data, Region == "Tashkent city"))
tidy(doctorsmodel)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept) 86.4       677.       0.128    0.900
## 2 year        -0.00456     0.337   -0.0135   0.989
```

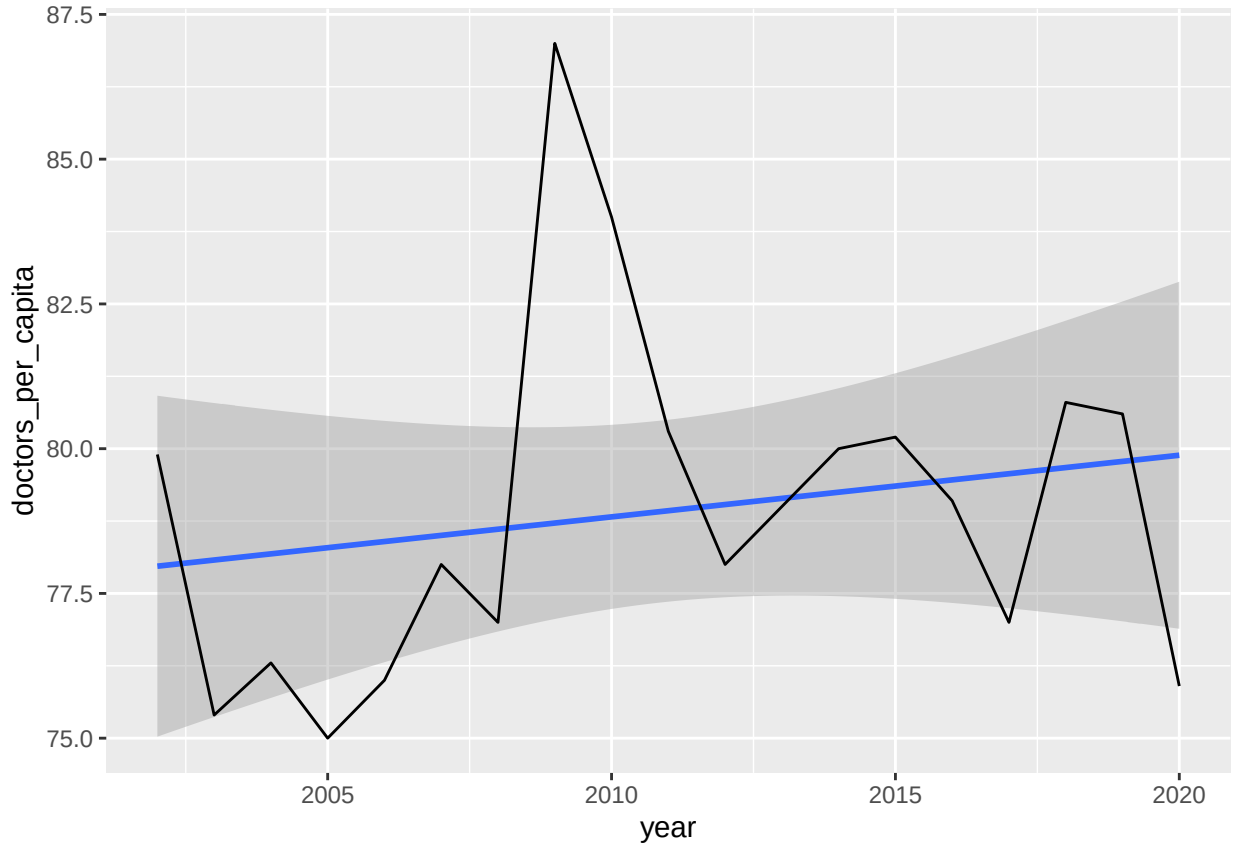
```
glance(doctorsmodel)$p.value < 0.01
```

```
## value
## FALSE
```

```
#remove likely error

uz_data %>%
```

```
  filter(Region == "Tashkent city", doctors_per_capita >= 60) %>%
  ggplot(aes(x=year,y=doctors_per_capita)) +
  geom_smooth(method = "lm") +
  geom_line()
```

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




```
doctorsmodel <- linear_reg() %>%
  set_engine("lm") %>%
  fit(doctors_per_capita ~ year, data = filter(uz_data, Region == "Tashkent city"))
tidy(doctorsmodel)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept) 86.4       677.       0.128    0.900
## 2 year        -0.00456     0.337   -0.0135   0.989
```

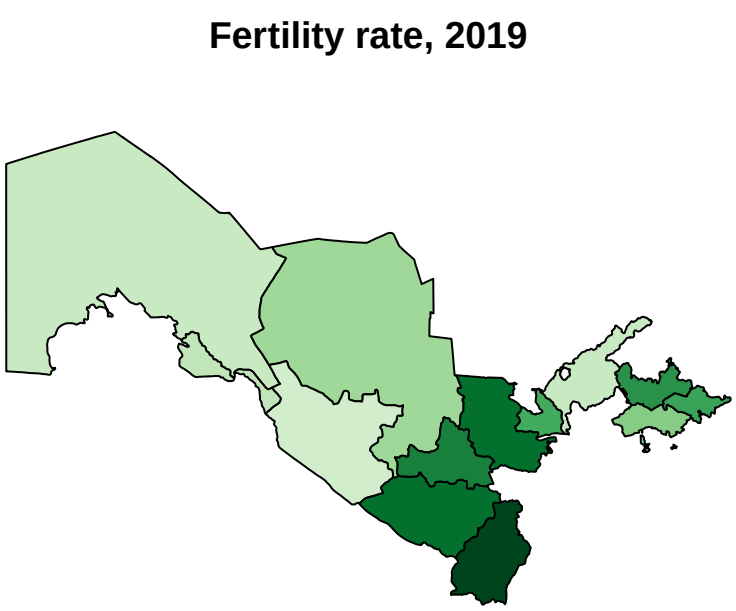
```
glance(doctorsmodel)$p.value < 0.01
```

```
## value
## FALSE
```

We have evidence to support that doctors per capita have decreased by approximately 0.24 doctors per 10000 population each year in Uzbekistan. Even once removing the likely error in our data, we do not have evidence to support a linear relationship between doctors per capita and year during this time period in Tashkent City.

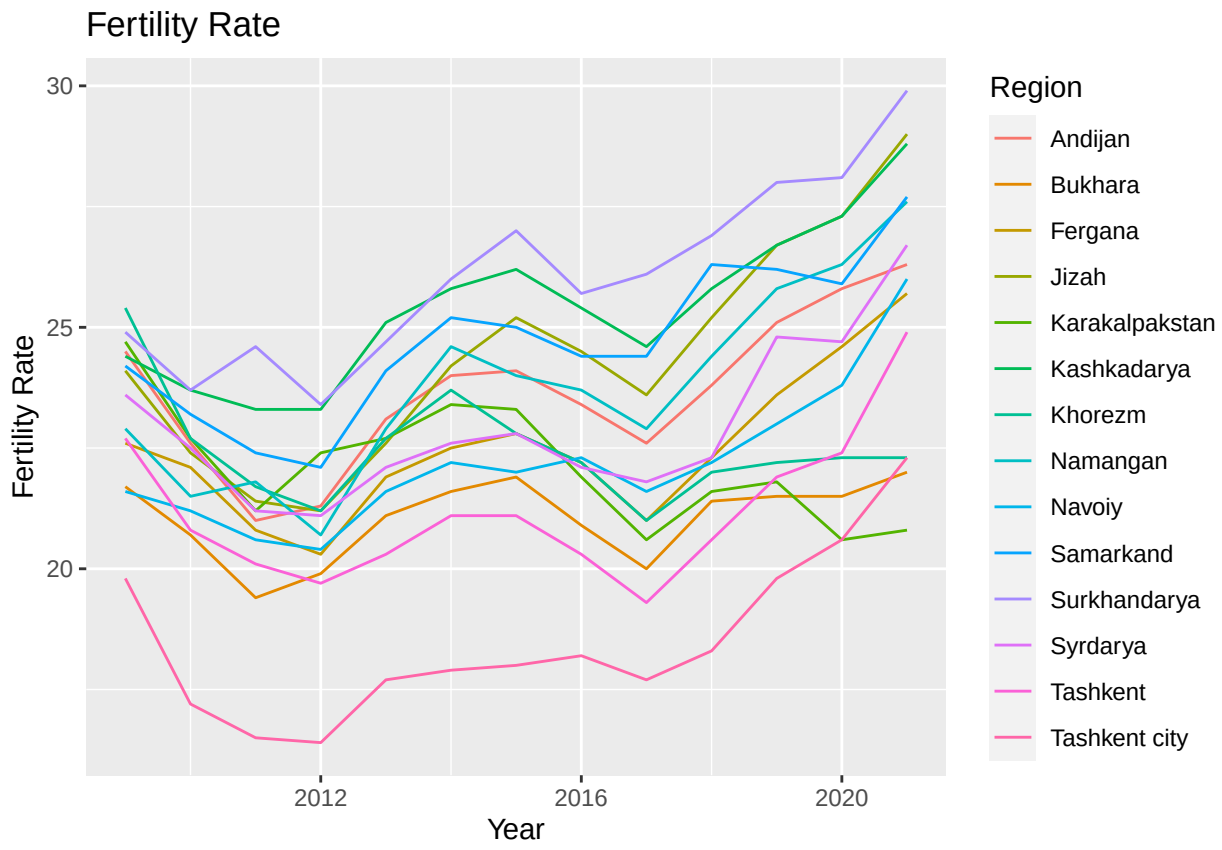
**Fertility Rates**

```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(fert_rate == as.numeric(fert_rate)) %>%
  select(fert_rate)

map_1@data$fert_rate <- as.numeric(unlist(test))

library(RColorBrewer)
my_colors <- brewer.pal(9, "Greens")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(test$fert_rate, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Fertility rate, 2019")
```

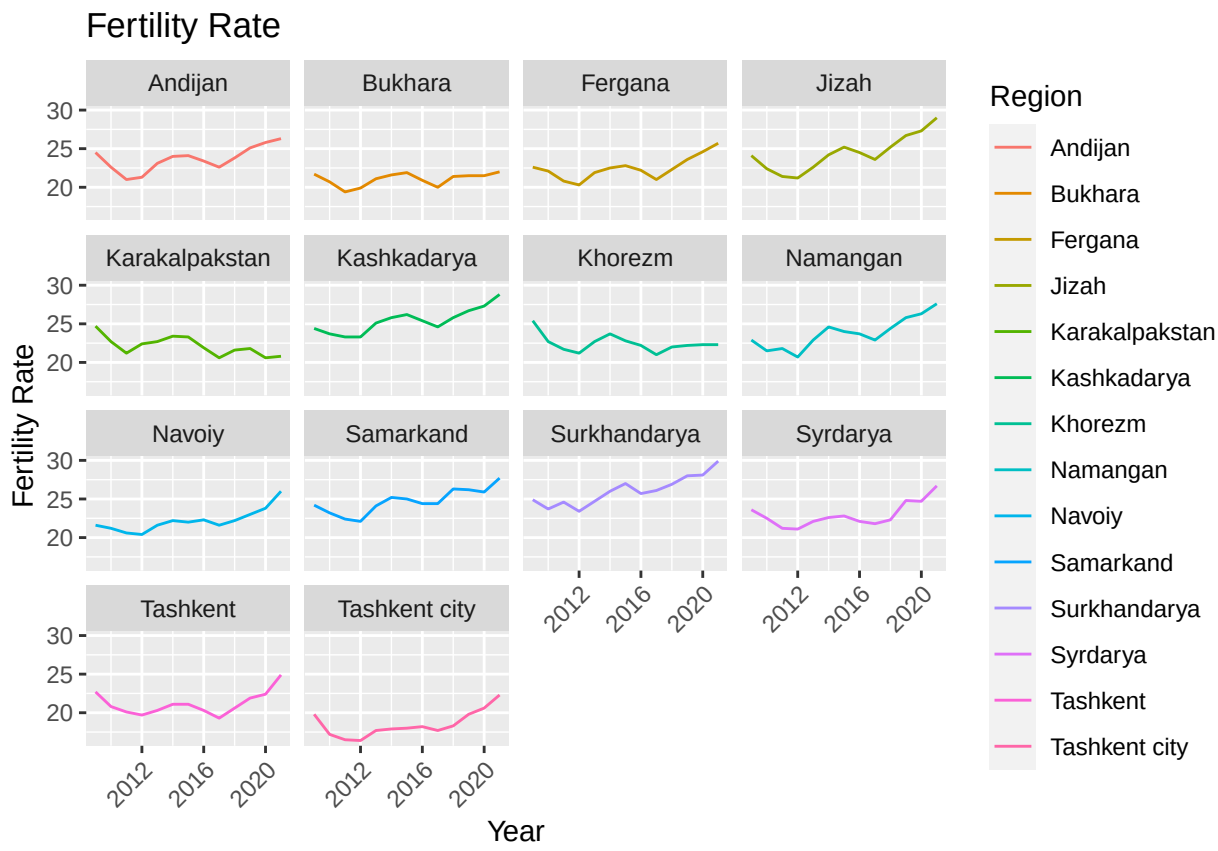



**Charts**

```r
yrs <- uz_data %>%
  filter(fert_rate != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- yrs$year[2]

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
ggplot(aes(x = year, y = fert_rate, color = Region)) +
  geom_line() +
  labs(title = "Fertility Rate",
       x = "Year", y = "Fertility Rate")
```

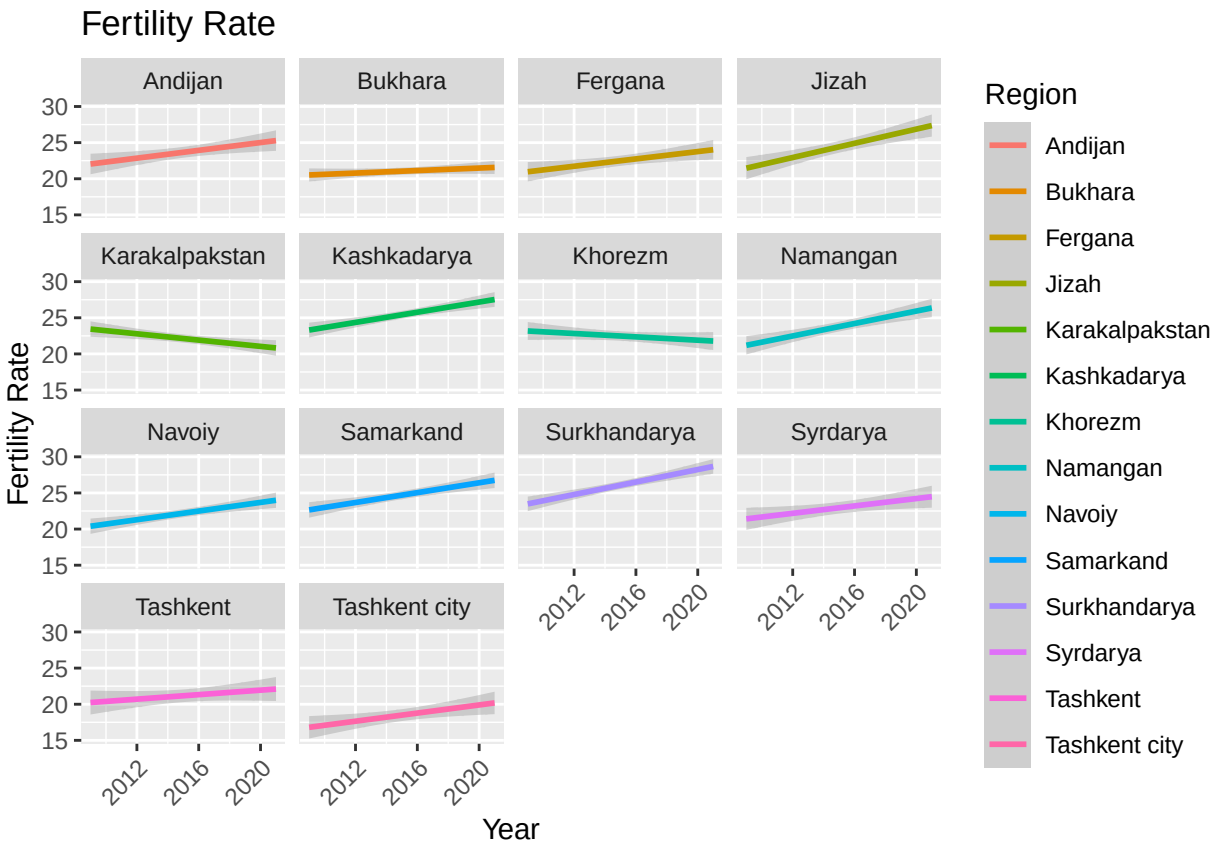



```r
## Note the trend for Tashkent city (I can highlight this if necessary)

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
ggplot(aes(x = year, y = fert_rate, color = Region)) +
  geom_line() +
  facet_wrap(~ Region) +
  labs(title = "Fertility Rate",
     x = "Year", y = "Fertility Rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```




```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
ggplot(aes(x = year, y = fert_rate, color = Region)) +
  geom_smooth(method = "lm") +
  facet_wrap(~ Region) +
  labs(title = "Fertility Rate",
     x = "Year", y = "Fertility Rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

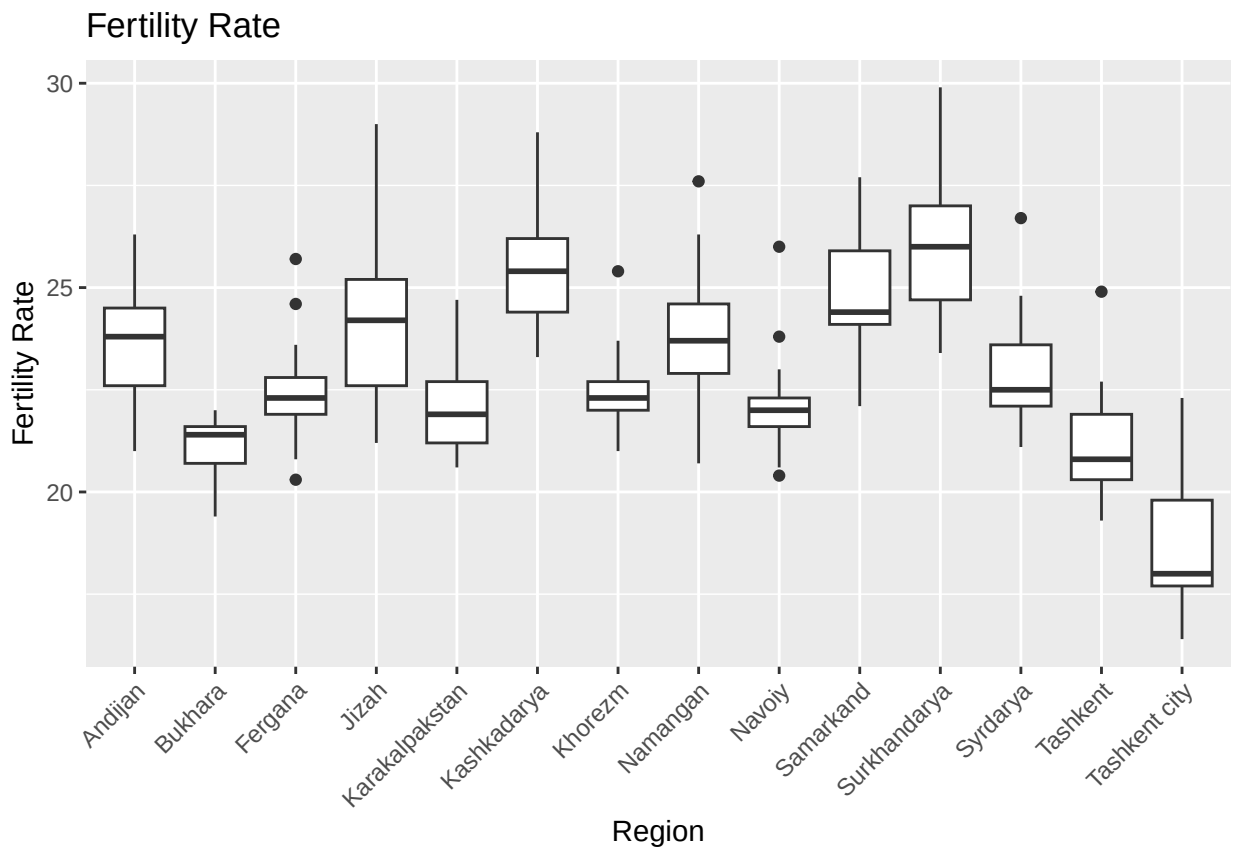
```
## `geom_smooth()` using formula = 'y ~ x'
```



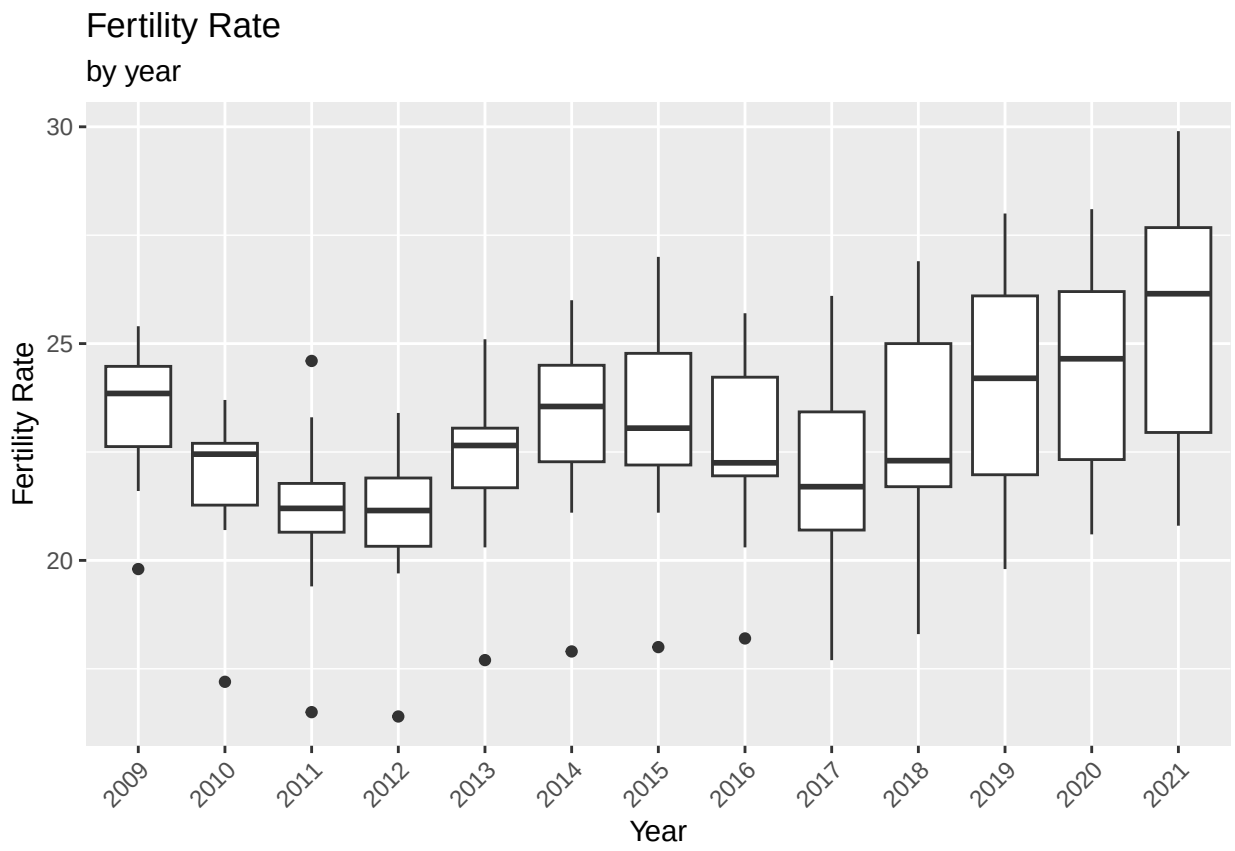

```
## very small standard error for each
## which regions tend to have downward trends? Karakalpakstan, Khorezm (West/SW regions) + Bukhara with
```

```
# By Region

uz_data %>%
  ggplot(aes(x = Region, y = fert_rate)) +
  geom_boxplot() +
  labs(title = "Fertility Rate",
     x = "Region", y = "Fertility Rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

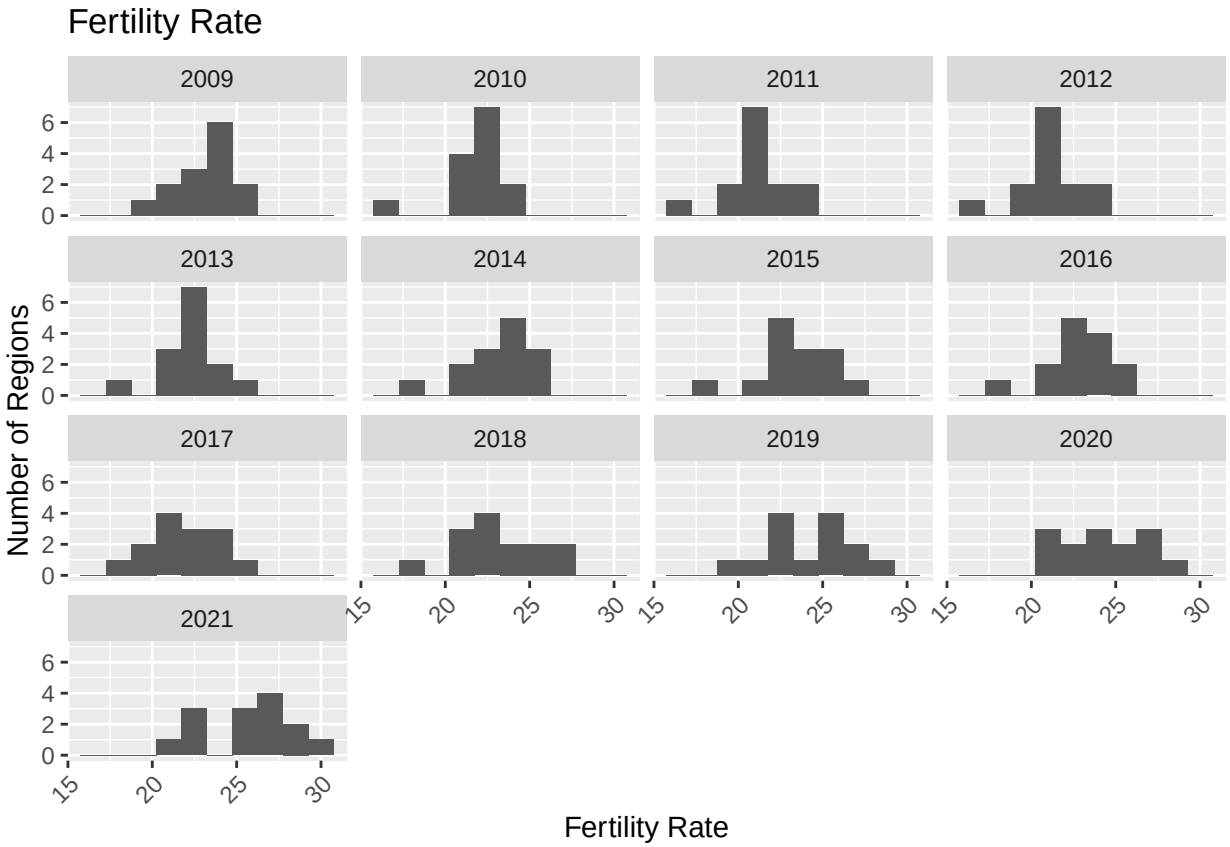
```
## Warning: Removed 280 rows containing non-finite values (‘stat_boxplot()‘).
```




```
# By year

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = as.character(year), y = fert_rate)) +
  geom_boxplot() +
  labs(title = "Fertility Rate", subtitle = "by year",
     x = "Year", y = "Fertility Rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```



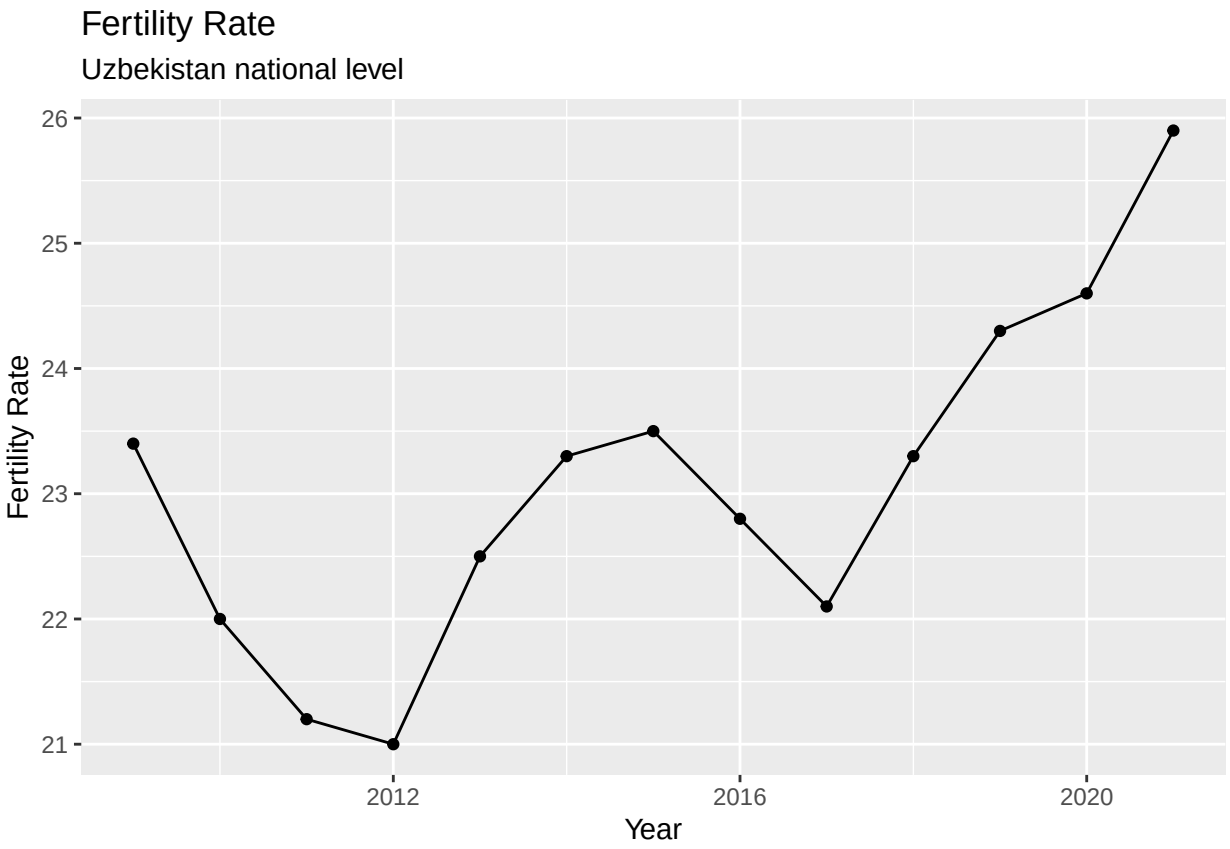

```
## Note the greater spread in fertility rate as the years go on

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = fert_rate)) +
  geom_histogram(bins = 10) +
  facet_wrap(~ year) +
  labs(title = "Fertility Rate", x = "Fertility Rate",
       y = "Number of Regions") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```




```
## more of showing the increased spread
```

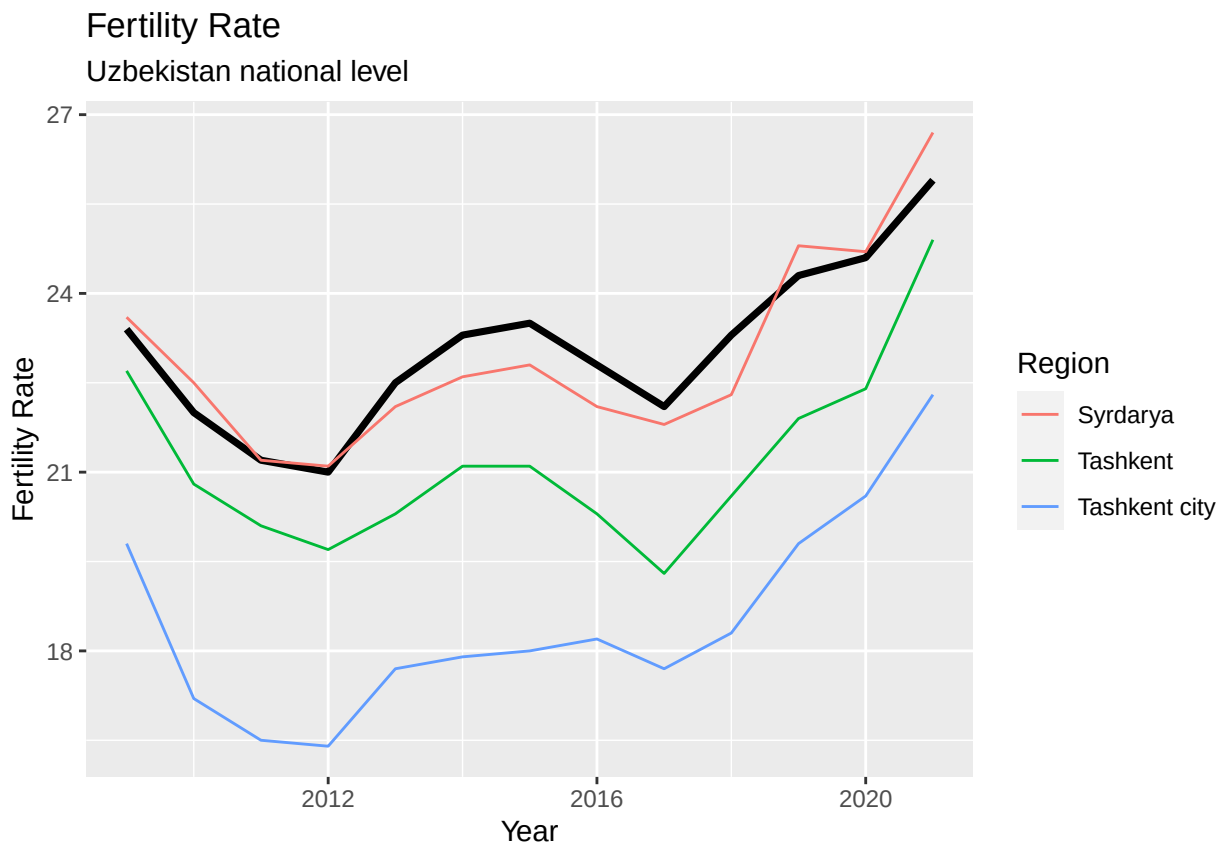
**Looking at the national level**

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = fert_rate)) +
  geom_point() +
  geom_line() +
  labs(title = "Fertility Rate", subtitle = "Uzbekistan national level", x = "Year", y = "Fertility Rat
```




```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
ggplot(aes(x = year, y = fert_rate)) +
  geom_line(size=1.25) +
  geom_line(data = filter(uz_data, Region == "Tashkent city" | Region == "Navoi" | Region == "Jizzakh"
  labs(title = "Fertility Rate", subtitle = "Uzbekistan national level", x = "Year", y = "Fertility Rat
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
```




### Descriptives

The fertility rate in Uzbekistan in 2009 was 23.4. Eight regions exhibited more than the national average for the fertility rate in 2009.

```
uz_fert_2009 <- nat_uz_data %>%
  filter(year == 2009) %>%
  select(fert_rate)
uz_fert_2009 <- uz_fert_2009$fert_rate
uz_fert_2009
```

```
## [1] 23.4
```

```
uz_data %>%
  filter(fert_rate > uz_fert_2009,
         year == 2009) %>%
  select(Region, year, fert_rate)
```

```
## # A tibble: 8 x 3
##   Region          year fert_rate
##   <chr>          <dbl>     <dbl>
## 1 Andijan         2009      24.5
## 2 Jizah           2009      24.1
## 3 Karakalpakstan  2009      24.7
## 4 Kashkadarya     2009      24.4
## 5 Khorezm         2009      25.4
```

```
## 6 Samarkand       2009      24.2
## 7 Surkhandarya    2009      24.9
## 8 Syrdarya        2009      23.6
```

The fertility rate in Uzbekistan in 2021 was 25.9. Eight regions had over than the national average for the fertility rate in 2021, with Khorezm/Karakalpakstan trading places with Navoi/Namagan.

```
uz_fert_2021 <- nat_uz_data %>%
  filter(year == 2021) %>%
  select(fert_rate)
uz_fert_2021 <- uz_fert_2021$fert_rate
uz_fert_2021
```

```
## [1] 25.9
```

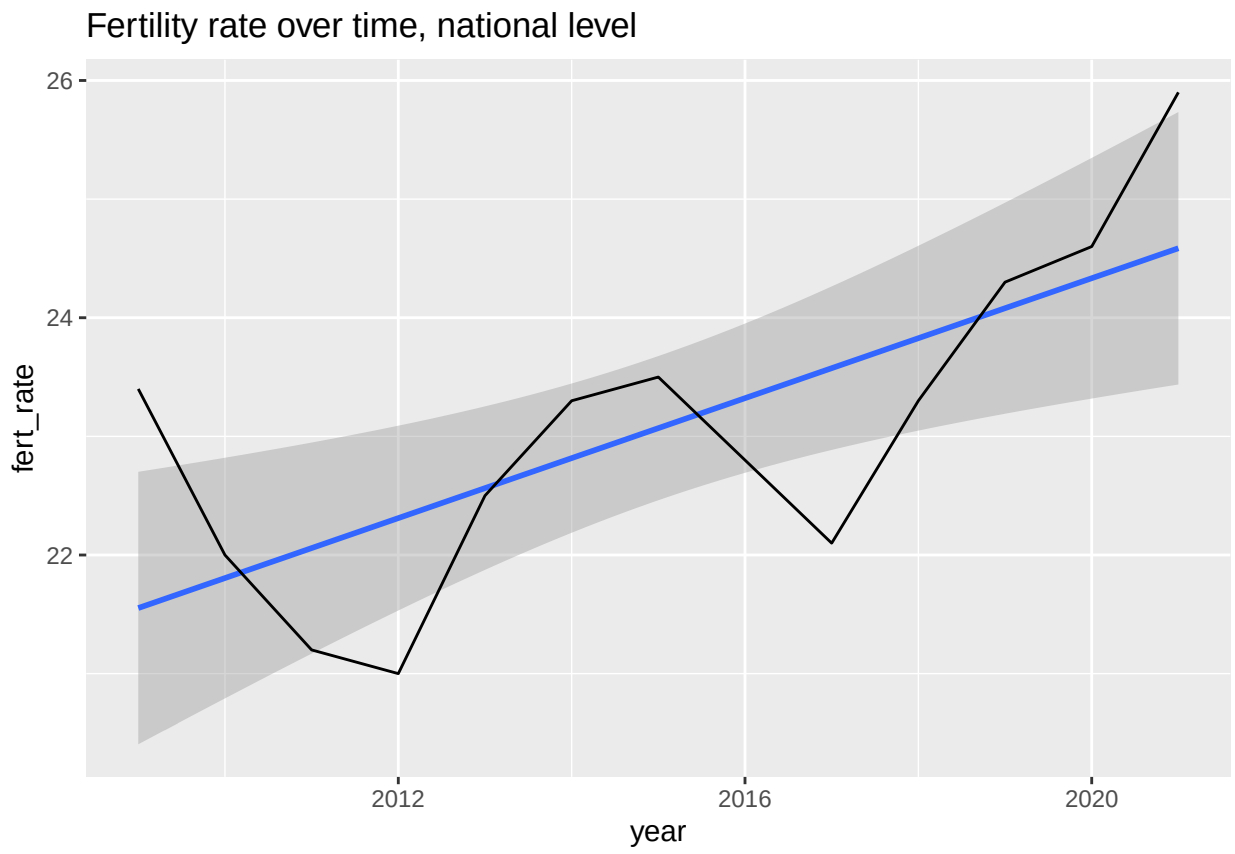
```
uz_data %>%
  filter(fert_rate > uz_fert_2021,
         year == 2021) %>%
  select(Region, year, fert_rate)
```

```
## # A tibble: 8 x 3
##   Region        year fert_rate
##   <chr>        <dbl>     <dbl>
## 1 Andijan       2021      26.3
## 2 Jizah         2021      29
## 3 Kashkadarya   2021      28.8
## 4 Namangan      2021      27.6
## 5 Navoiy        2021      26
## 6 Samarkand     2021      27.7
## 7 Surkhandarya  2021      29.9
## 8 Syrdarya      2021      26.7
```

### Simple regressions

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x=year,y=fert_rate)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "Fertility rate over time, national level")
```

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




```
uz_fert_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(fert_rate ~ year, data = nat_uz_data)
tidy(uz_fert_model)
```

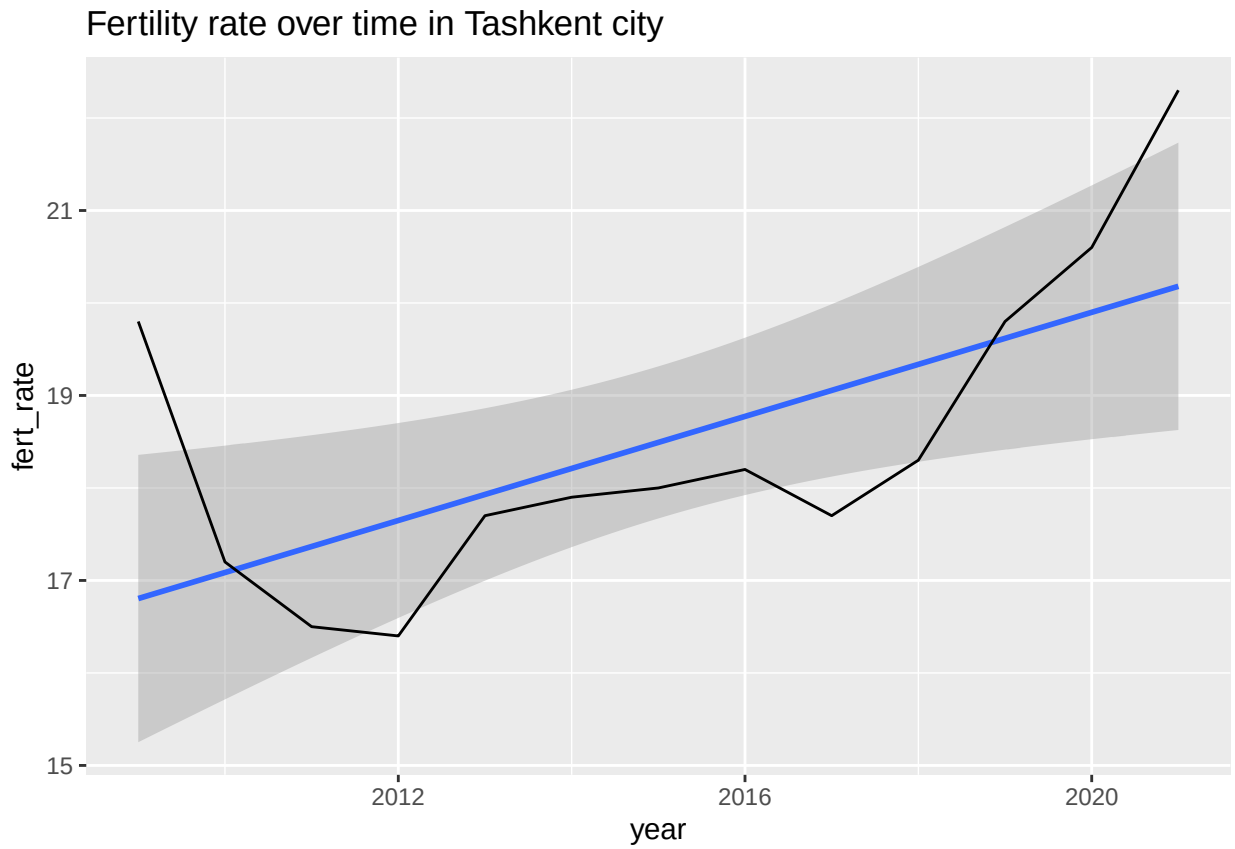
```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept) -486.     149.         -3.27 0.00747
## 2 year           0.253    0.0738      3.43 0.00567
```

```
glance(uz_fert_model)$p.value < 0.01
```

```
## value
##  TRUE
```

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region == "Tashkent city") %>%
  ggplot(aes(x=year,y=fert_rate)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "Fertility rate over time in Tashkent city")
```

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




```
fert_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(fert_rate ~ year, data = filter(uz_data, Region == "Tashkent city"))
tidy(fert_model)
```

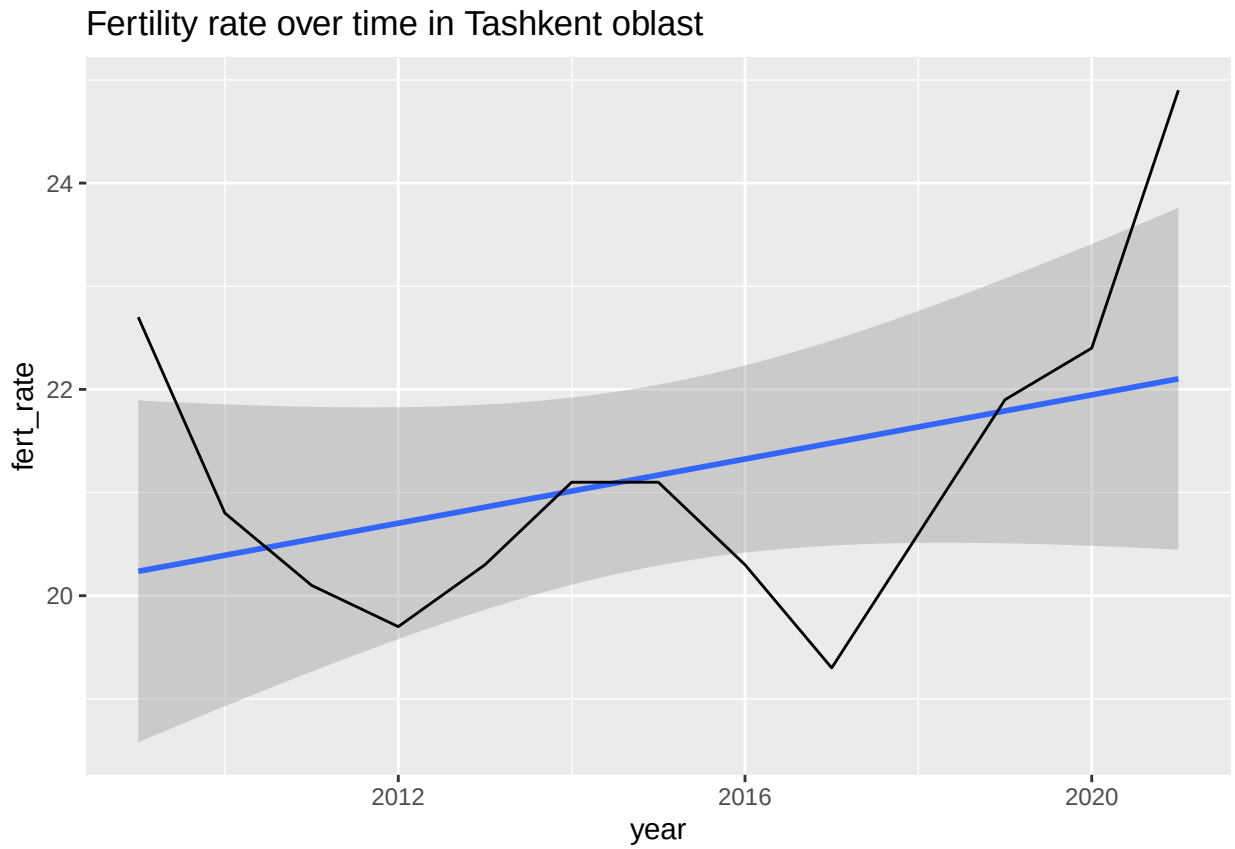
```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept) -548.     201.         -2.73  0.0197
## 2 year           0.281    0.0998      2.82  0.0167
```

```
glance(fert_model)$p.value ##almost significant
```

```
##      value
## 0.01671483
```

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region == "Tashkent") %>%
  ggplot(aes(x=year,y=fert_rate)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "Fertility rate over time in Tashkent oblast")
```

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




```
fert_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(fert_rate ~ year, data = filter(uz_data, Region == "Tashkent"))
tidy(fert_model)
```

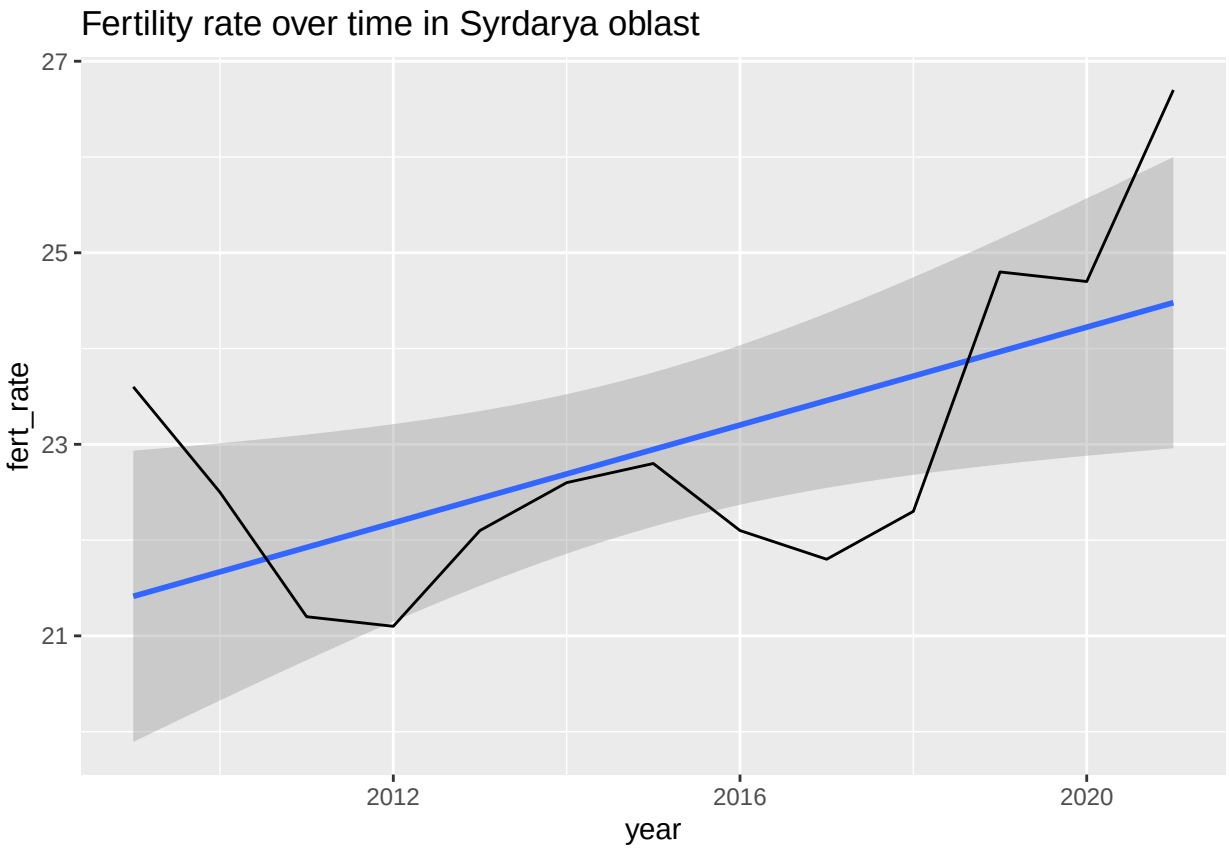
```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept) -292.      215.        -1.36   0.200
## 2 year           0.155     0.106      1.46   0.172
```

```
glance(fert_model)$p.value < 0.01
```

```
## value
## FALSE
```

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region == "Syrdarya") %>%
  ggplot(aes(x=year,y=fert_rate)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "Fertility rate over time in Syrdarya oblast")
```

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




```
fert_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(fert_rate ~ year, data = filter(uz_data, Region == "Syrdarya"))
tidy(fert_model)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept) -492.     197.         -2.50  0.0296
## 2 year           0.255    0.0977      2.61  0.0240
```

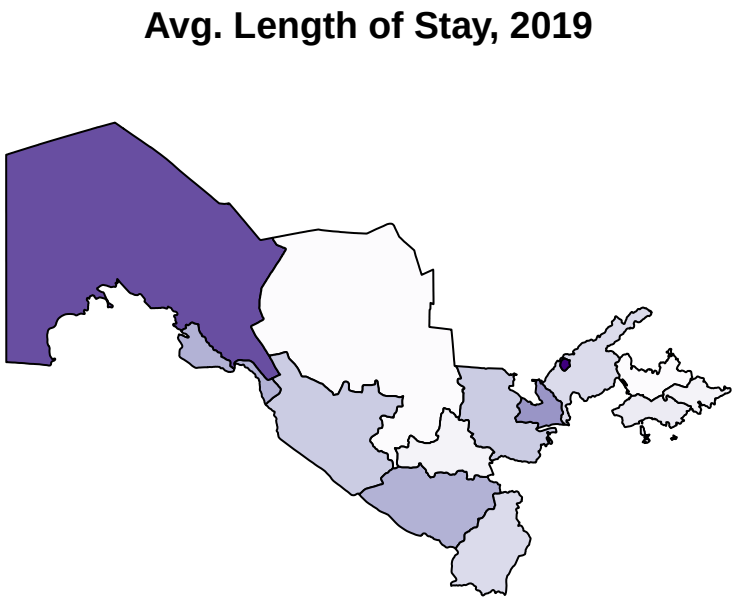
```
glance(fert_model)$p.value < 0.01
```

```
## value
## FALSE
```

We have evidence to support that the fertility rate in Uzbekistan has increased by approximately 0.25 units each year 2009-2021. We also have moderately strong evidence to support a linear relationship between fertility rate and year during this time period in Tashkent City. However, we don't have evidence to support such a relationship in Tashkent ot Syrdarya oblasts.

**Average Length of Stay**

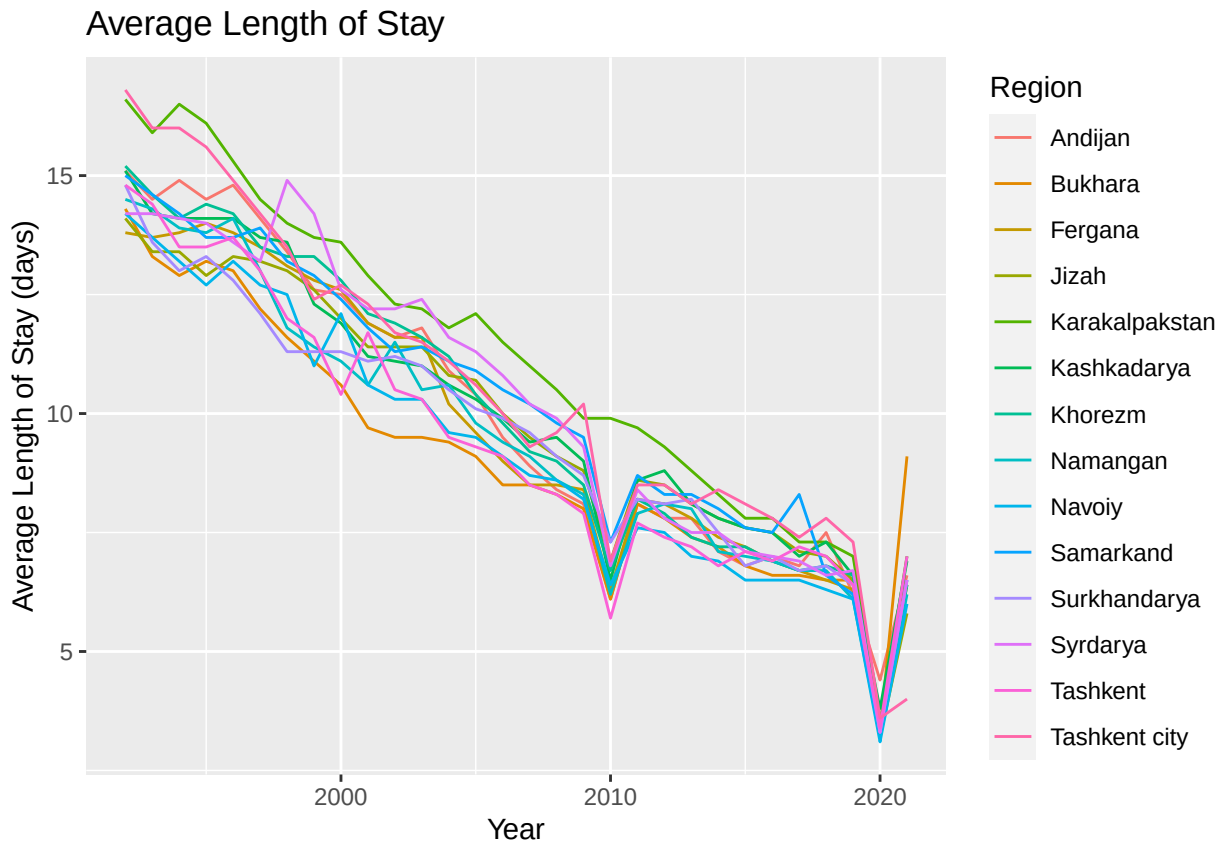
```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(LOS == as.numeric(LOS)) %>%
  select(LOS)

map_1@data$LOS <- as.numeric(unlist(test))

library(RColorBrewer)
my_colors <- brewer.pal(9, "Purples")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(test$LOS, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Avg. Length of Stay, 2019")
```



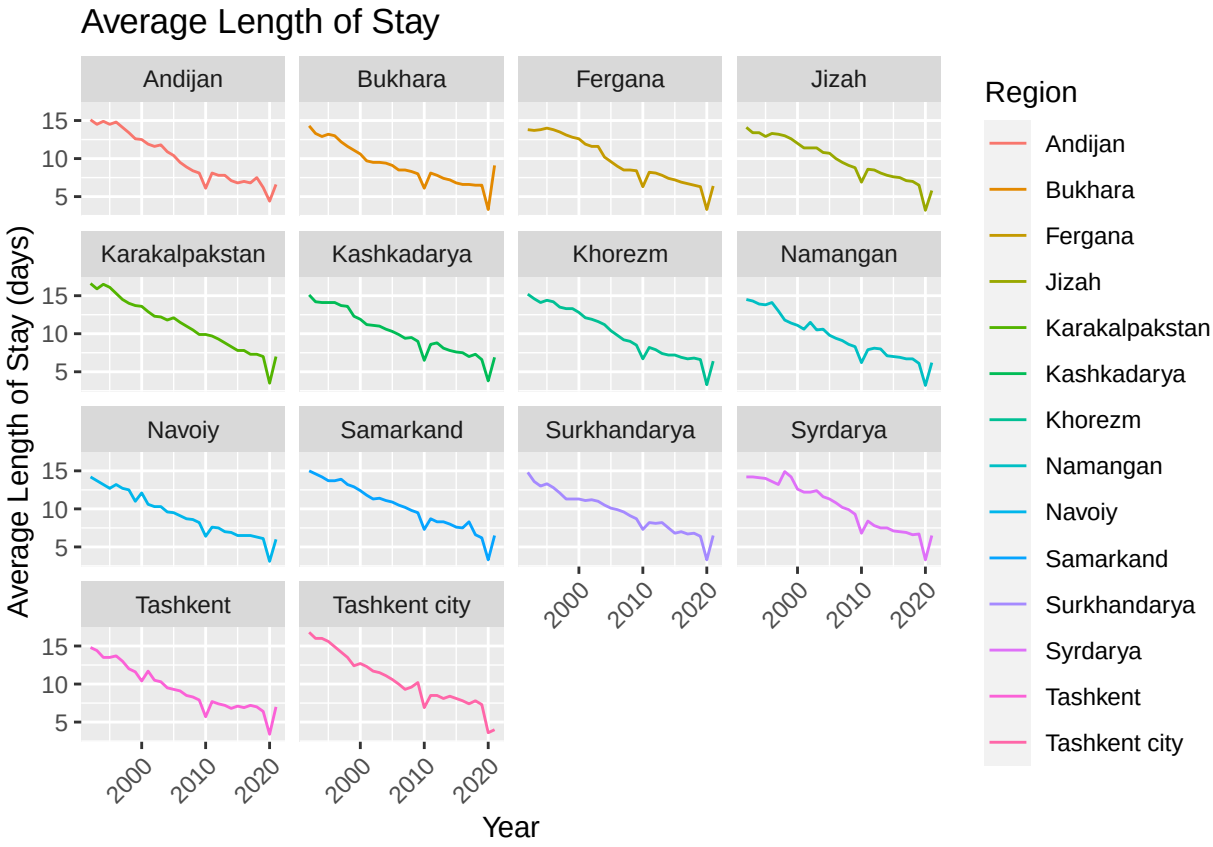

Charts

```
yrs <- uz_data %>%
  filter(LOS != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- yrs$year[2]
```

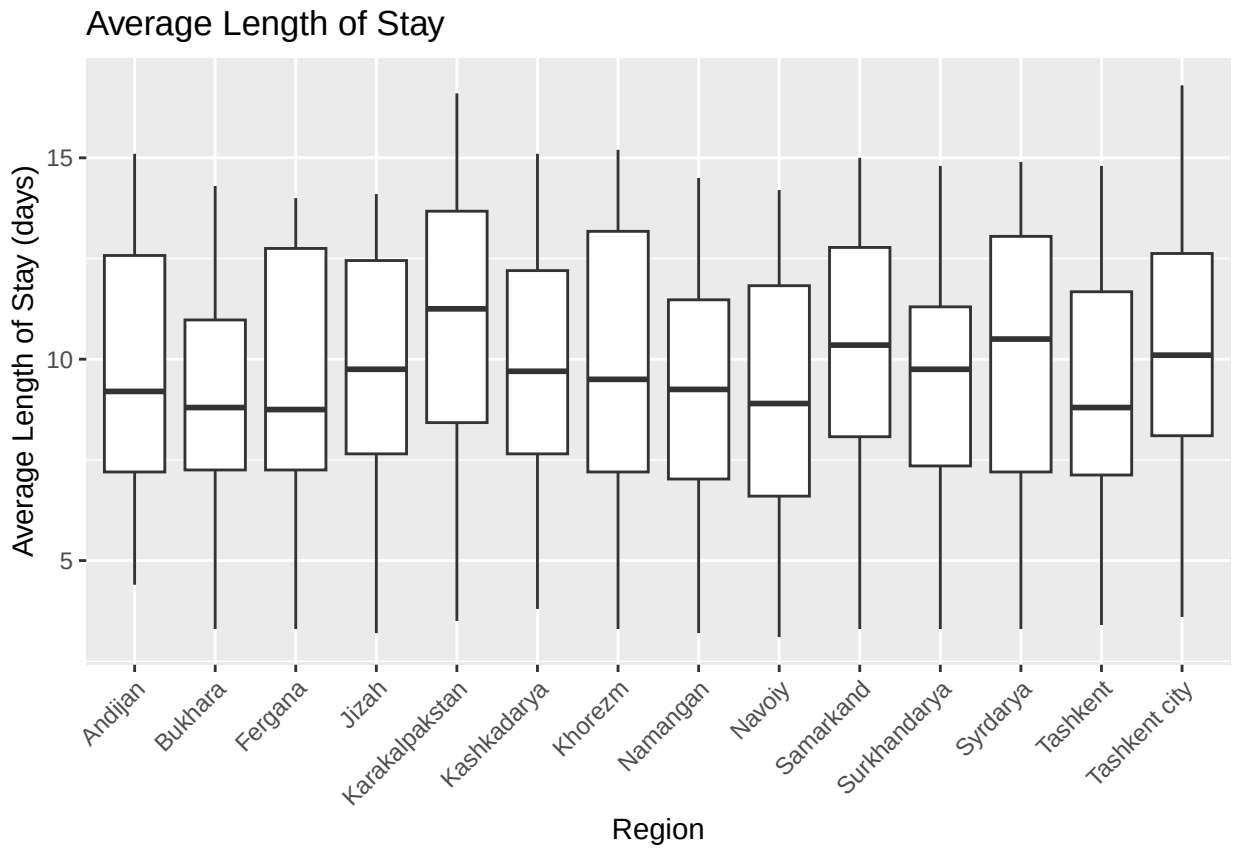
```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
ggplot(aes(x = year, y = LOS, color = Region)) +
  geom_line() +
  labs(title = "Average Length of Stay",
       x = "Year", y = "Average Length of Stay (days)")
```



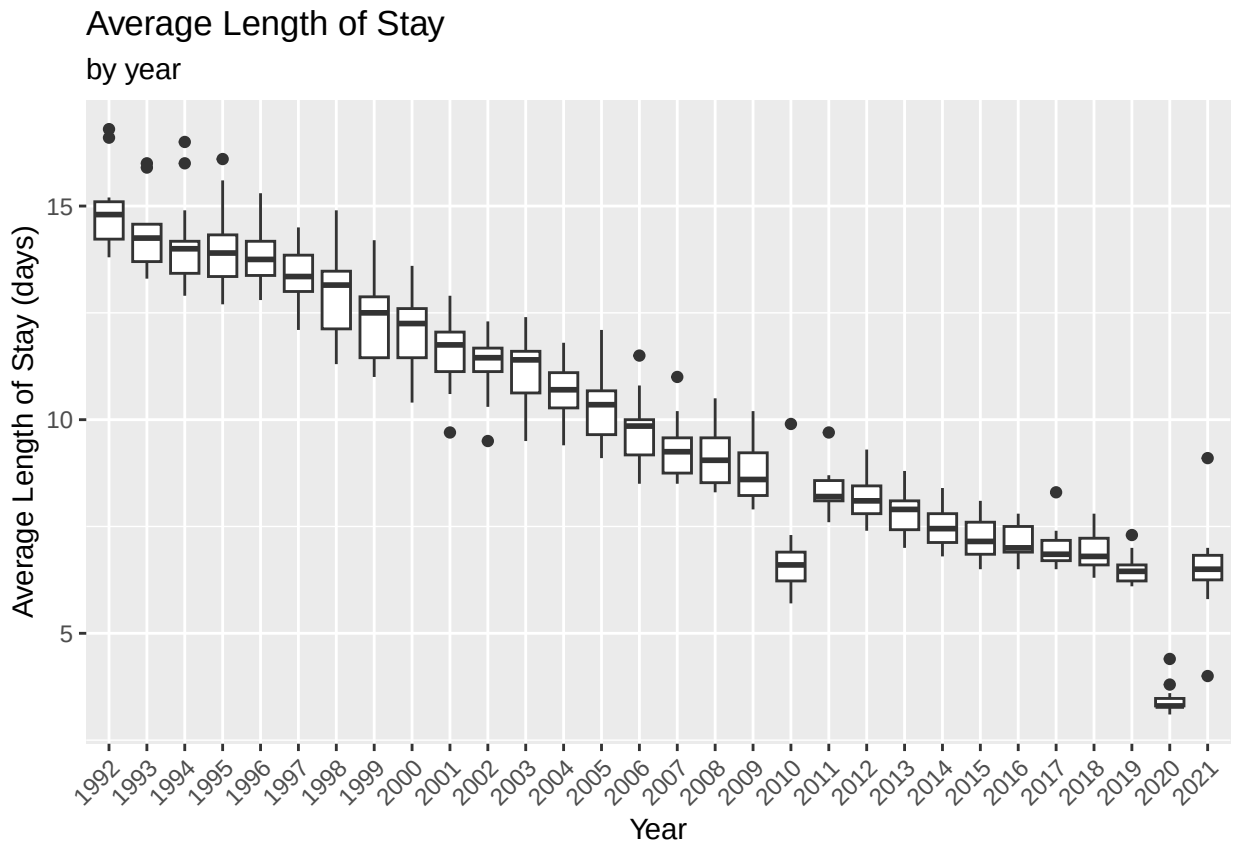

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
ggplot(aes(x = year, y = LOS, color = Region)) +
  geom_line() +
  facet_wrap(~ Region) +
  labs(title = "Average Length of Stay",
     x = "Year", y = "Average Length of Stay (days)") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```



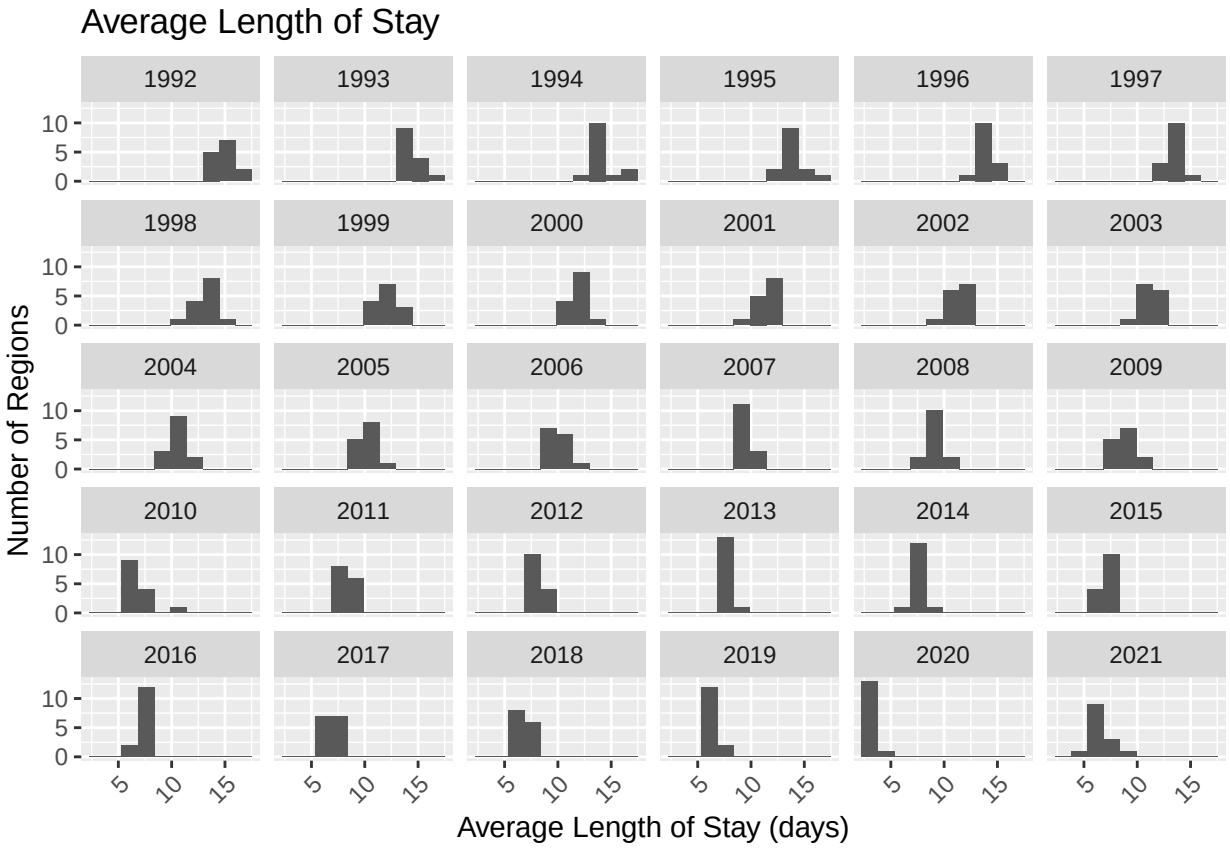

```
# By Region

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = Region, y = LOS)) +
  geom_boxplot() +
  labs(title = "Average Length of Stay",
     x = "Region", y = "Average Length of Stay (days)") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```




```
# By year

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = as.character(year), y = LOS)) +
  geom_boxplot() +
  labs(title = "Average Length of Stay", subtitle = "by year",
     x = "Year", y = "Average Length of Stay (days)") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```



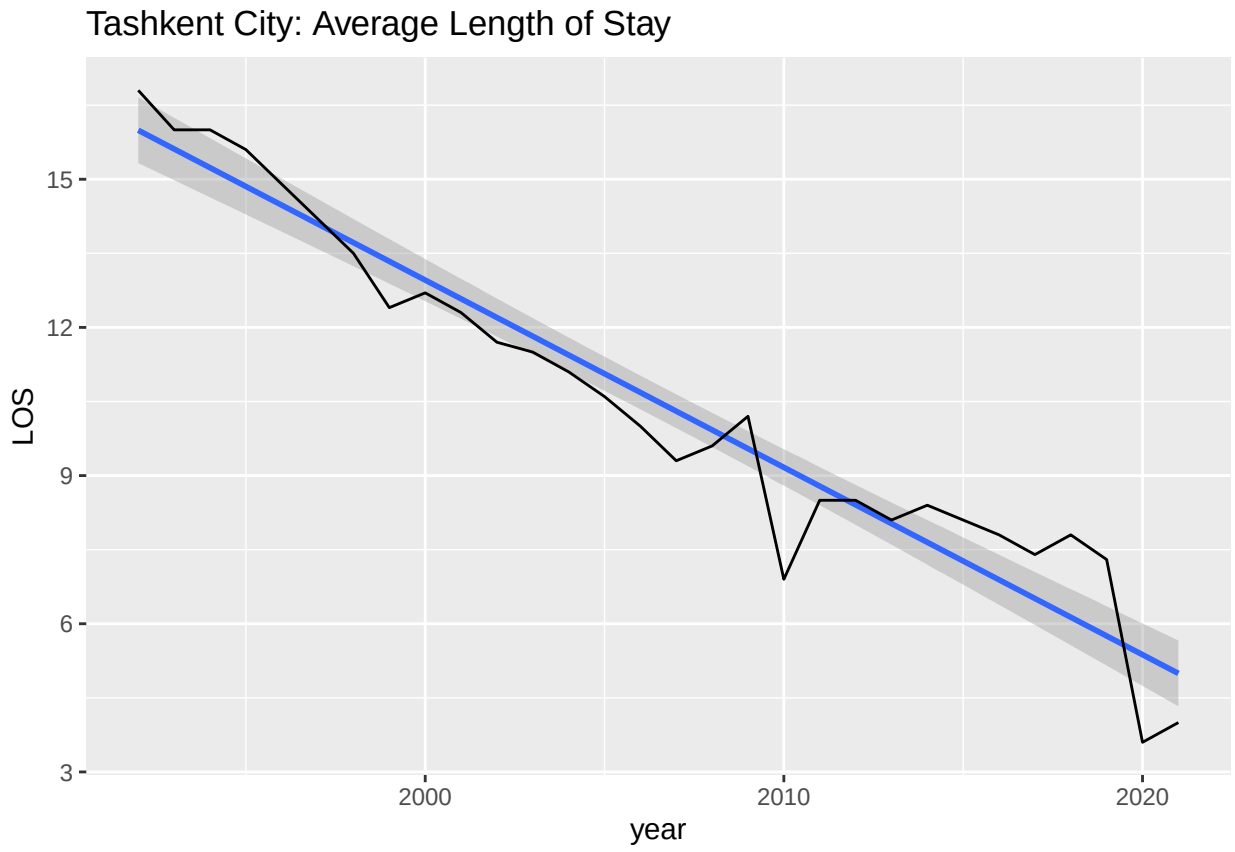

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = LOS)) +
  geom_histogram(bins = 10) +
  facet_wrap(~ year) +
  labs(title = "Average Length of Stay", x = "Average Length of Stay (days)",
       y = "Number of Regions") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```




Simple regressions

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region == "Tashkent city") %>%
  ggplot(aes(x=year,y=LOS)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "Tashkent City: Average Length of Stay")
```

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




```
LOS_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(LOS ~ year, data = filter(uz_data, Region == "Tashkent city"))
tidy(LOS_model)
```

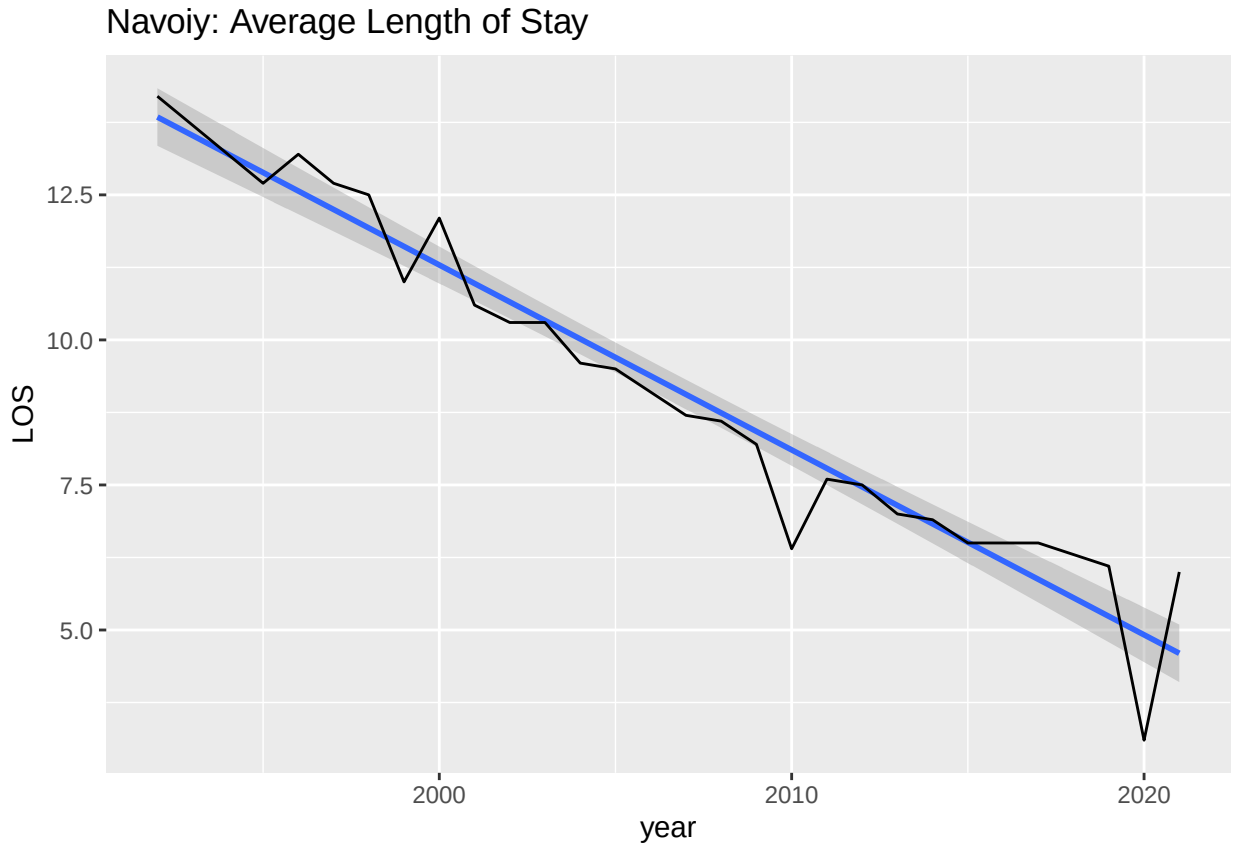
```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  771.      38.6         20.0 4.19e-18
## 2 year          -0.379    0.0192     -19.7 5.99e-18
```

```
glance(LOS_model)$p.value < 0.01
```

```
## value
##  TRUE
```

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region == "Navoiy") %>%
  ggplot(aes(x=year,y=LOS)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "Navoiy: Average Length of Stay")
```

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




```
LOS_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(LOS ~ year, data = filter(uz_data, Region == "Navoiy"))
tidy(LOS_model)
```

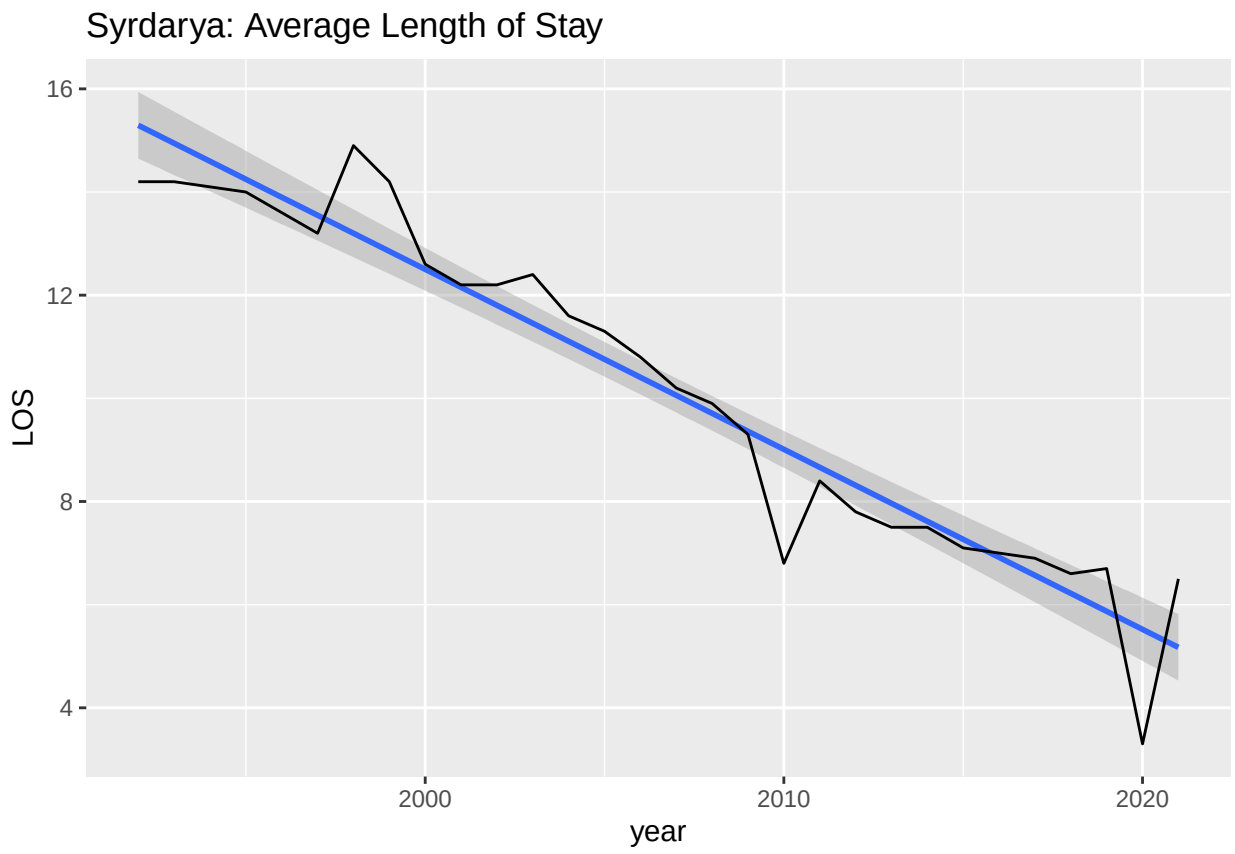
```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  649.      28.8         22.6 1.69e-19
## 2 year          -0.319    0.0143     -22.2 2.48e-19
```

```
glance(LOS_model)$p.value < 0.01
```

```
## value
##  TRUE
```

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  filter(Region == "Syrdarya") %>%
  ggplot(aes(x=year,y=LOS)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "Syrdarya: Average Length of Stay")
```

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




```
LOS_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(LOS ~ year, data = filter(uz_data, Region == "Syrdarya"))
tidy(LOS_model)
```

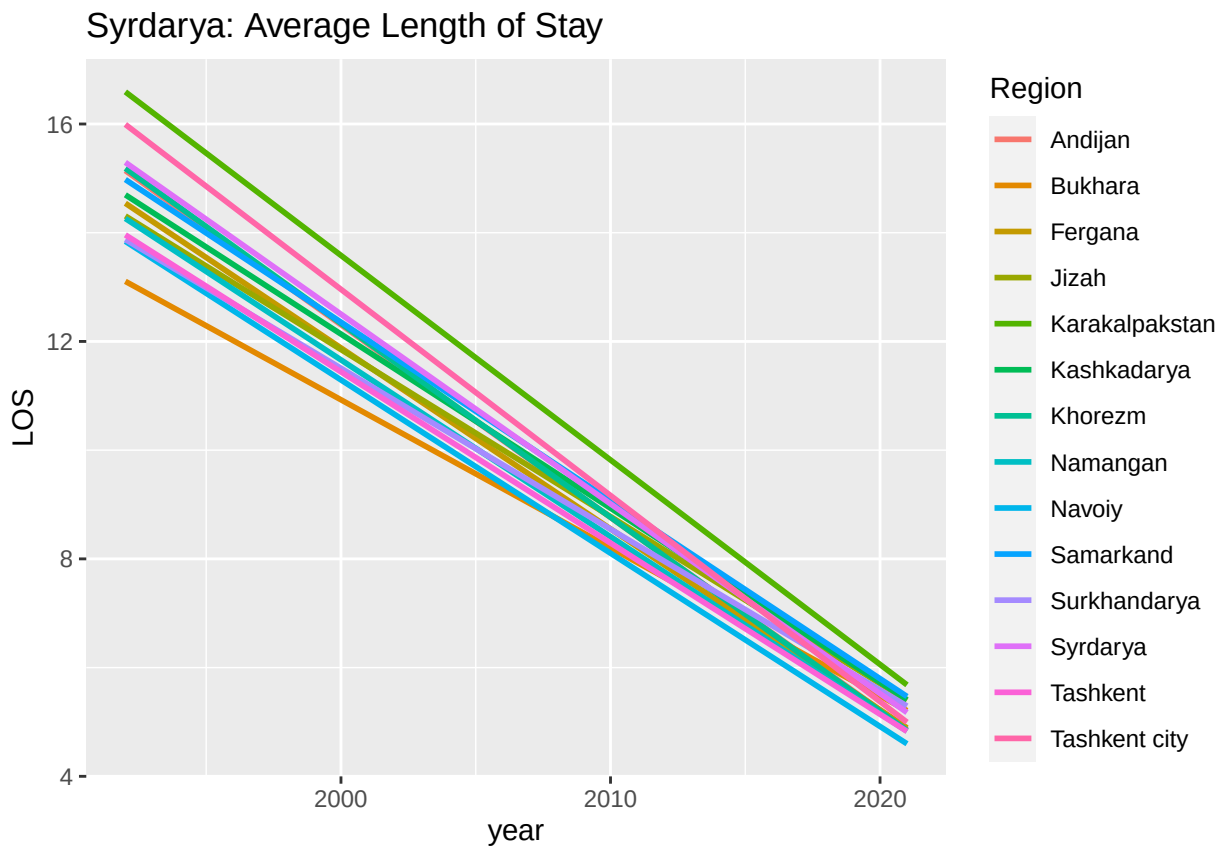
```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  711.      37.5         19.0 1.61e-17
## 2 year          -0.349    0.0187     -18.7 2.35e-17
```

```
glance(LOS_model)$p.value < 0.01
```

```
## value
##  TRUE
```

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x=year,y=LOS)) +
  geom_smooth(method = "lm", aes(color = Region), se = FALSE) +
  labs(title = "Syrdarya: Average Length of Stay")
```

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




```
LOS_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(LOS ~ year, data = uz_data)
tidy(LOS_model)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic   p.value
##   <chr>          <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)  674.     11.0          61.4 2.47e-211
## 2 year          -0.331   0.00547     -60.5 6.43e-209
```

```
glance(LOS_model)$p.value < 0.01
```

```
## value
##  TRUE
```

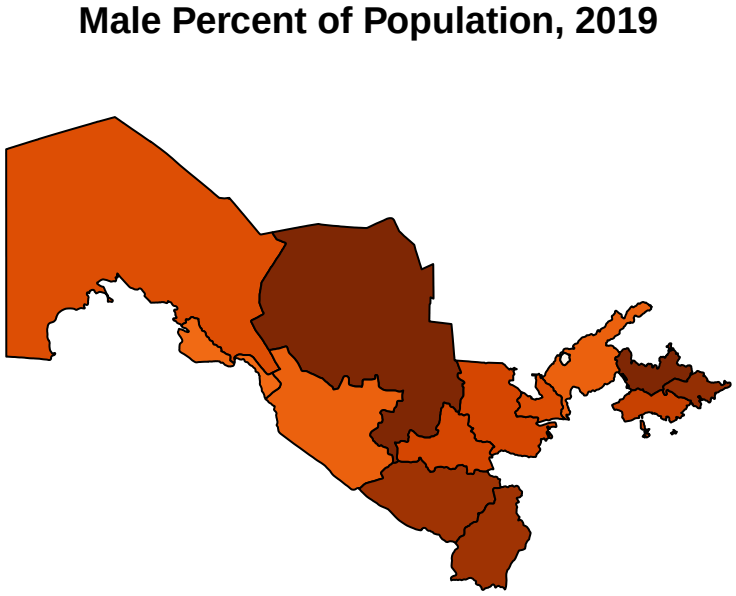
We have evidence to support that average length of hospital stay per capita decreases by approximately -0.38 days per year in Tashkent City from 2000-2021, -0.32 days per year in Navoiy, -0.35 days per year in Syrdarya, and so on.

**Population analysis**

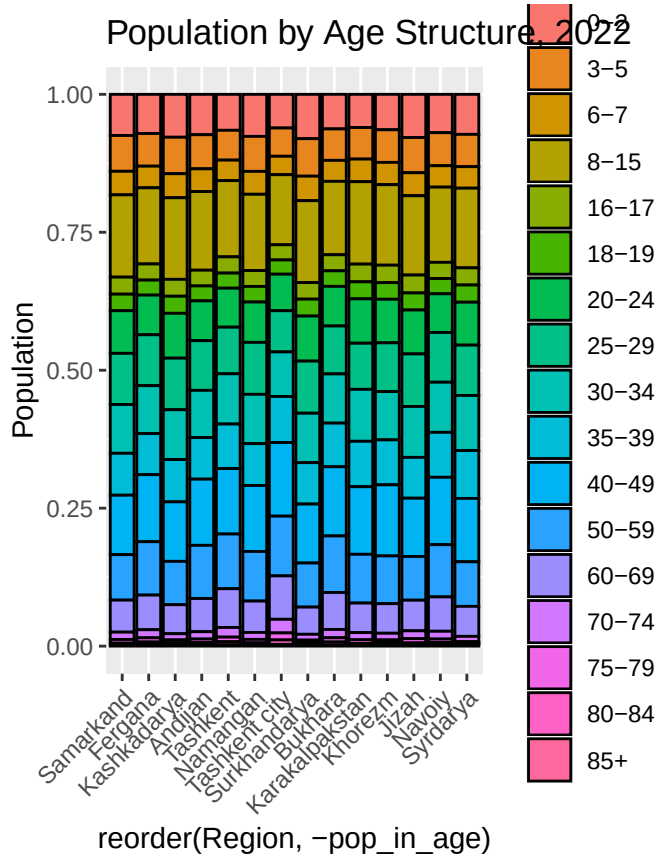
```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pop == as.numeric(pop)) %>%
  select(pop)

map_1@data$pop <- as.numeric(unlist(test))

library(RColorBrewer)
my_colors <- brewer.pal(9, "Oranges")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(test$pop, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Population, 2019")
```


```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(male_percent == as.numeric(male_percent)) %>%
  select(male_percent)
```

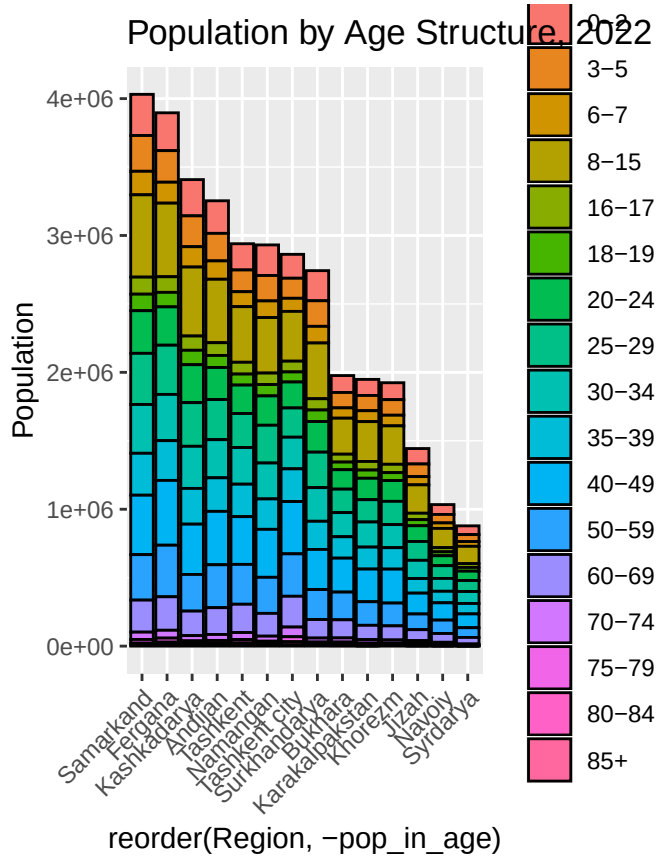
```
map_1@data$male_percent <- as.numeric(unlist(test))

library(RColorBrewer)
my_colors <- brewer.pal(9, "Oranges")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(test$male_percent, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Male Percent of Population, 2019")
```



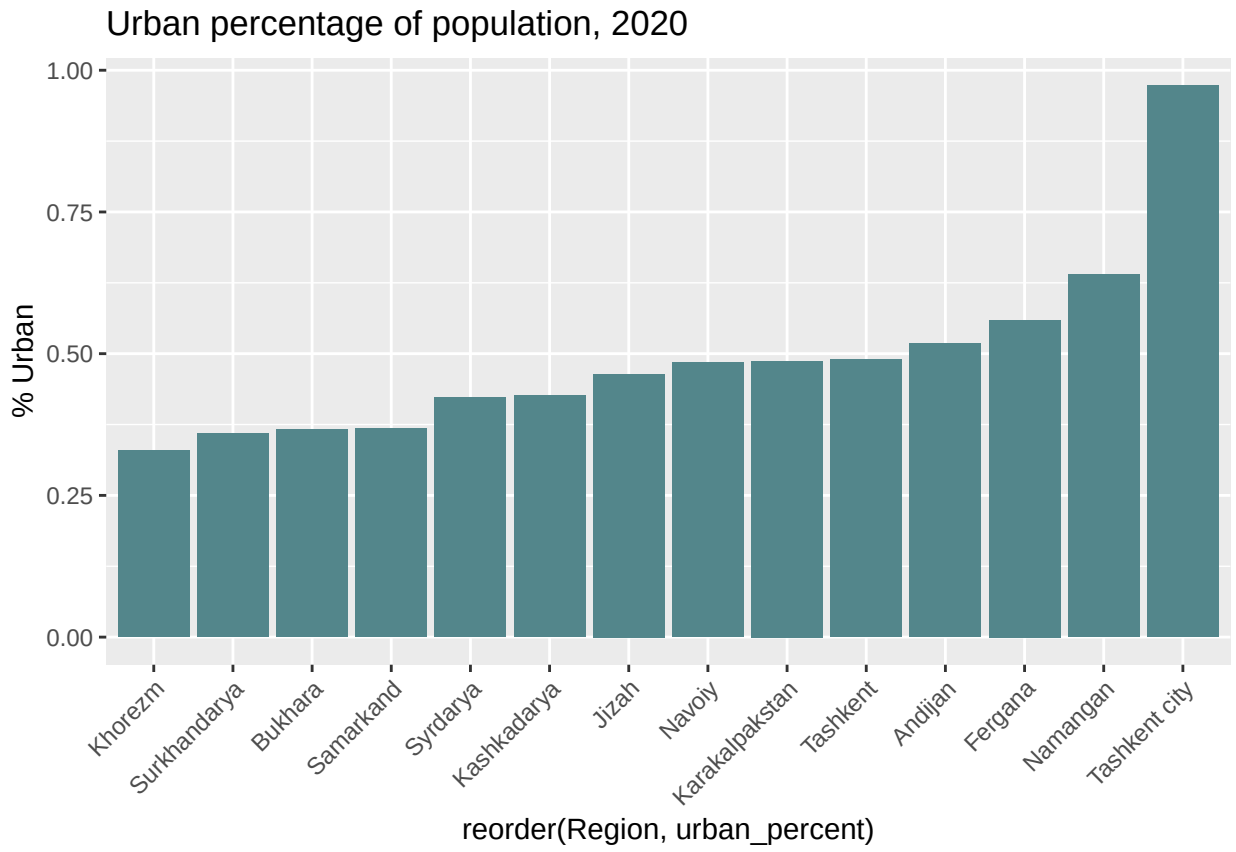

```
age_structure %>%
  filter(age_range != "Population") %>%
ggplot(mapping = aes(x = reorder(Region, -pop_in_age),
                     y = pop_in_age,
                     fill = fct_relevel(age_range,
                              c("0-2", "3-5", "6-7","8-15")))) +
        geom_col(position = "fill", color = "black") +
  labs(title = "Population by Age Structure, 2022", y = "Population") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```



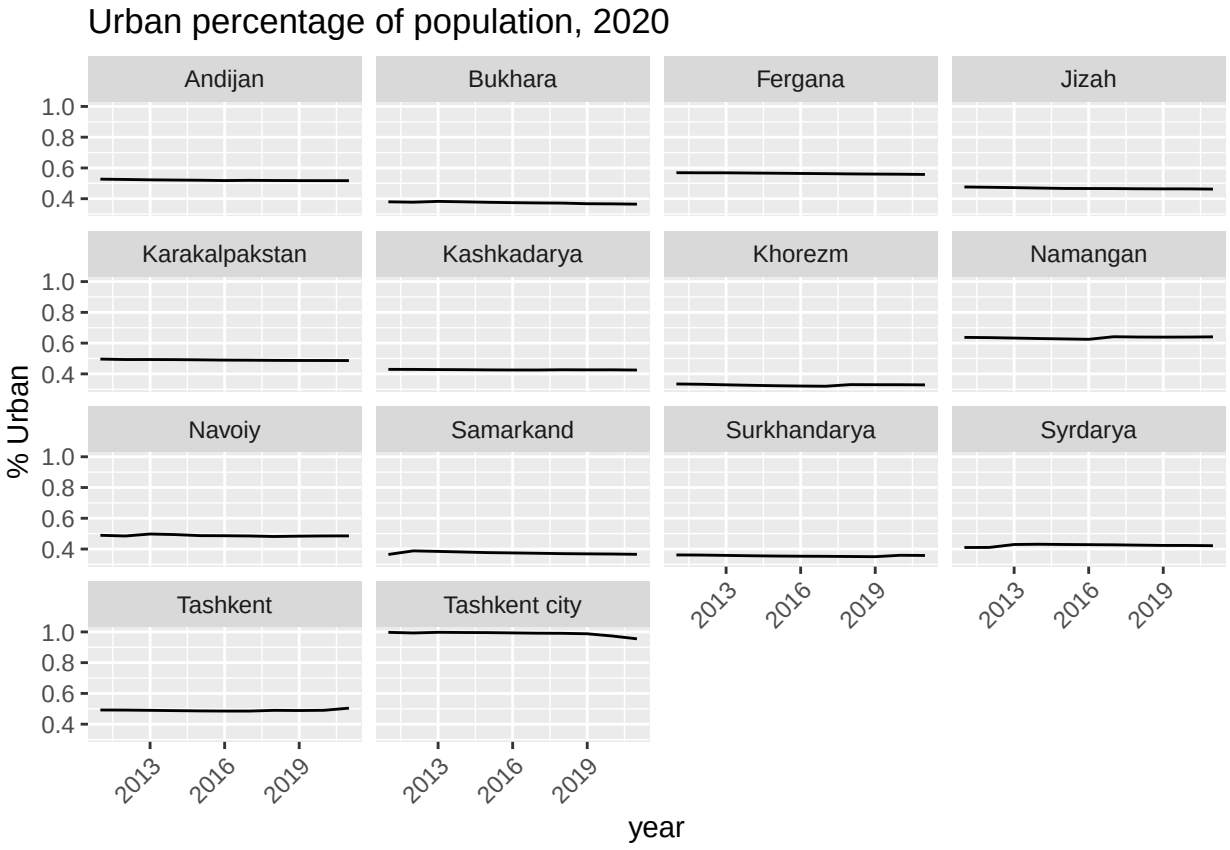

```
ggplot(data = filter(age_structure,
                     age_range != "Population"),
       mapping = aes(x = reorder(Region, -pop_in_age),
                     y = pop_in_age,
                     fill = fct_relevel(age_range,
                              c("0-2", "3-5", "6-7","8-15")))) +
         geom_col(position = "stack", color = "black") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Population by Age Structure, 2022", y = "Population")
```



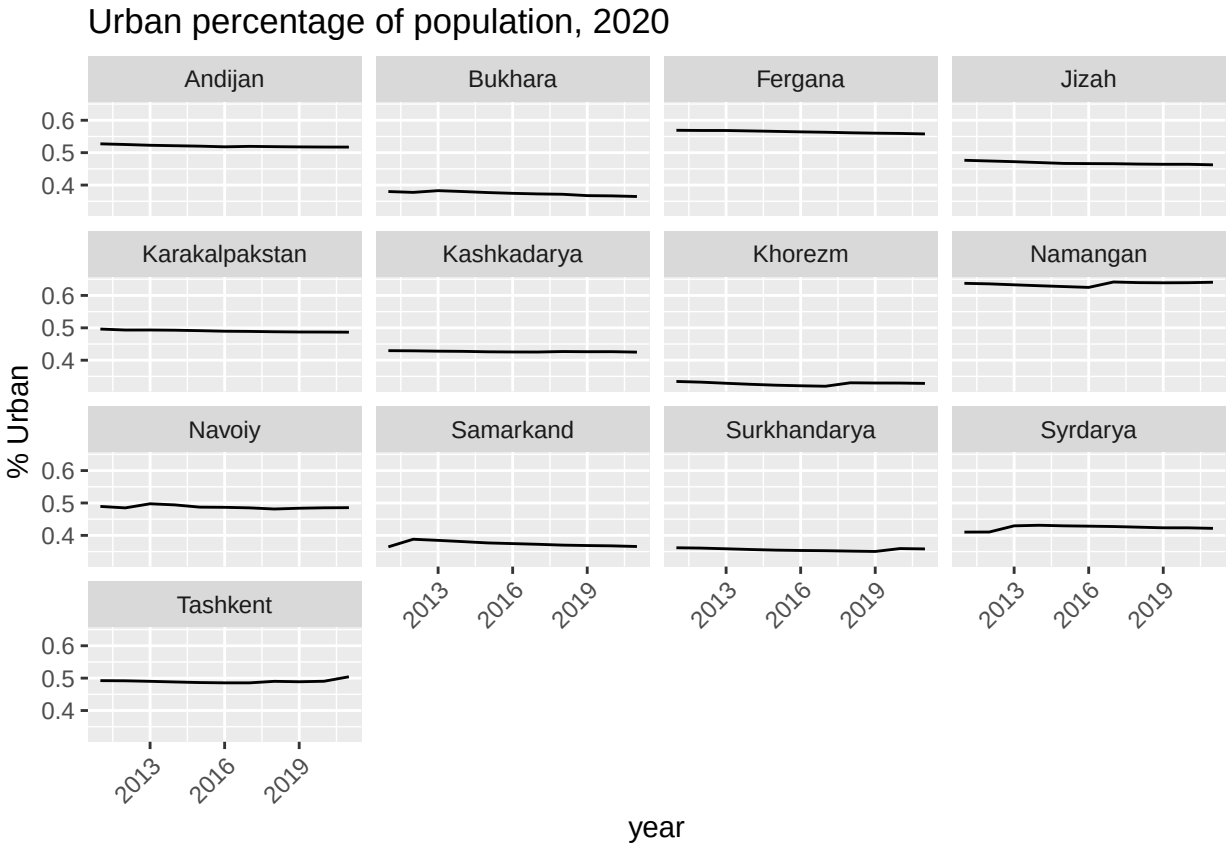

```
yrs <- uz_data %>%
  filter(pop_urban != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- yrs$year[2]


uz_data %>%
  filter(year == 2020) %>%
  group_by(Region) %>%
  ggplot(mapping = aes(x=reorder(Region, urban_percent), y=urban_percent)) +
  geom_col(fill = "cadetblue4") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Urban percentage of population, 2020", y = "% Urban")
```



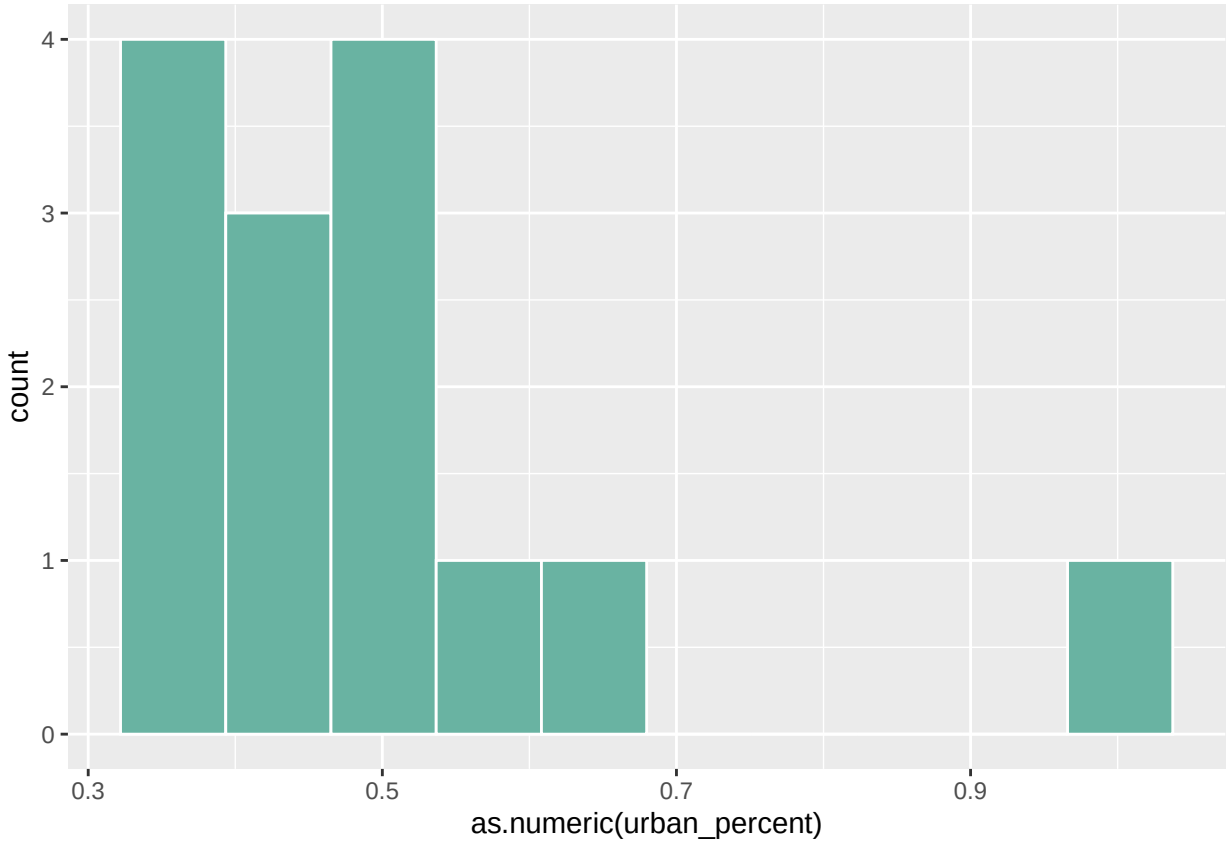

```
## filtering out likely data entry error
uz_data %>%
  filter(year >= yr_start, year <= yr_end, urban_percent <= 1) %>%
  ggplot(mapping = aes(x=year, y=urban_percent)) +
  geom_line() +
  facet_wrap(Region ~ .) +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Urban percentage of population, 2020", y = "% Urban")
```



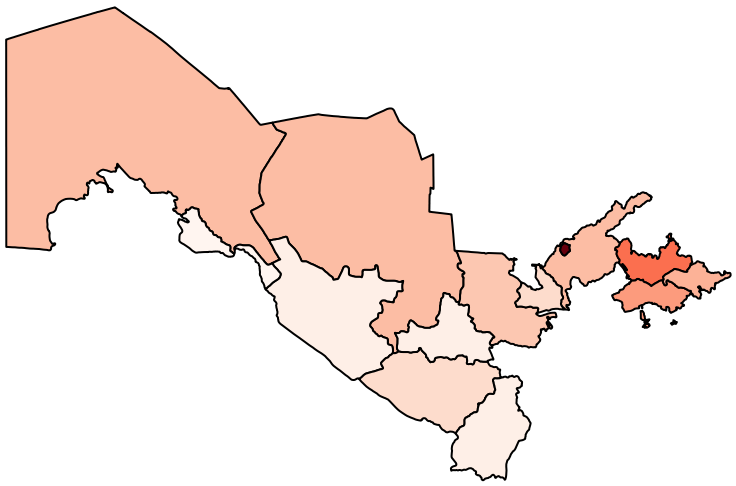

```
## excluding outlier, tashkent city
uz_data %>%
  filter(year >= yr_start, year <= yr_end, urban_percent <= 1,
         Region != "Tashkent city") %>%
  ggplot(mapping = aes(x=year, y=urban_percent)) +
  geom_line() +
  facet_wrap(Region ~ .) +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Urban percentage of population, 2020", y = "% Urban")
```




```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2020), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(urban_percent == as.numeric(urban_percent)) %>%
  select(urban_percent)

map_1@data$urban_percent <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(urban_percent))) +
    geom_histogram(bins=10, fill='#69b3a2', color='white')
```

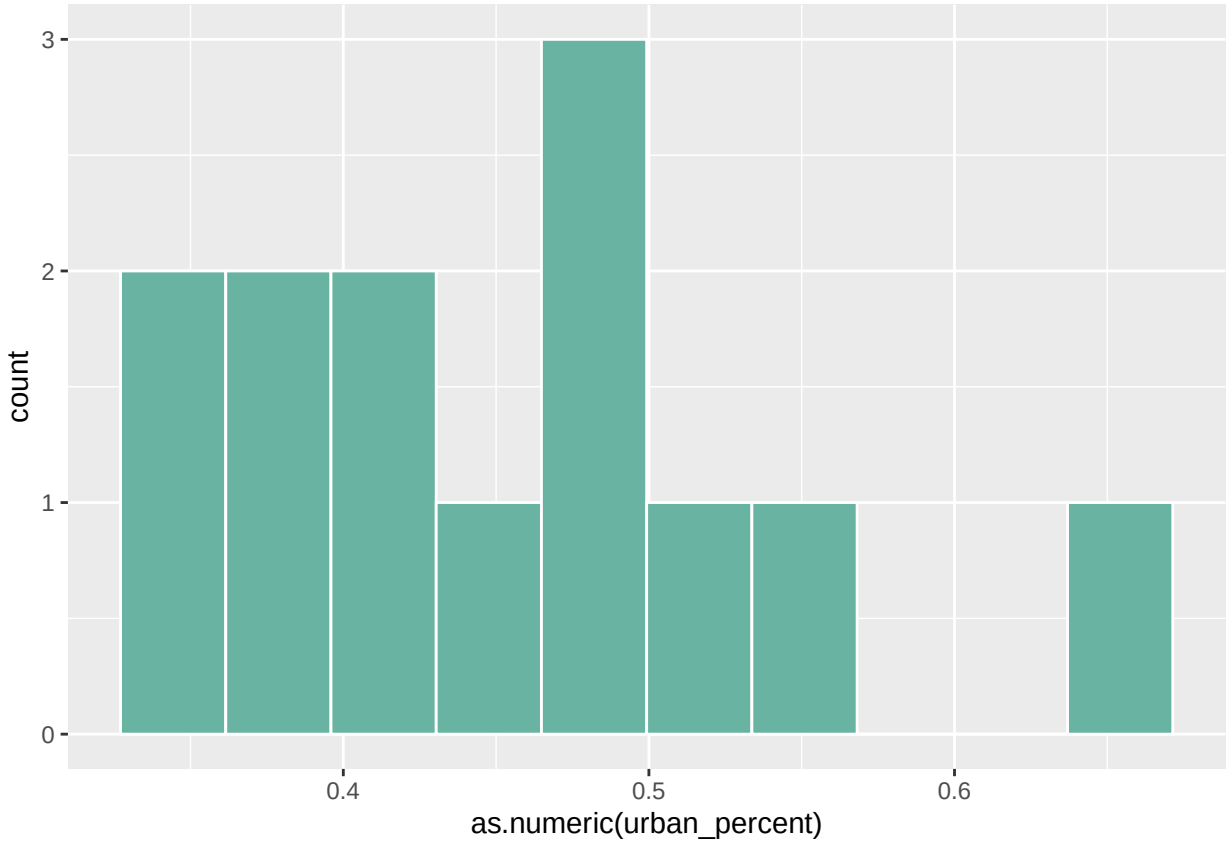



```
library(RColorBrewer)
my_colors <- brewer.pal(9, "Reds")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$urban_percent, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Urban population as percent of total")
```

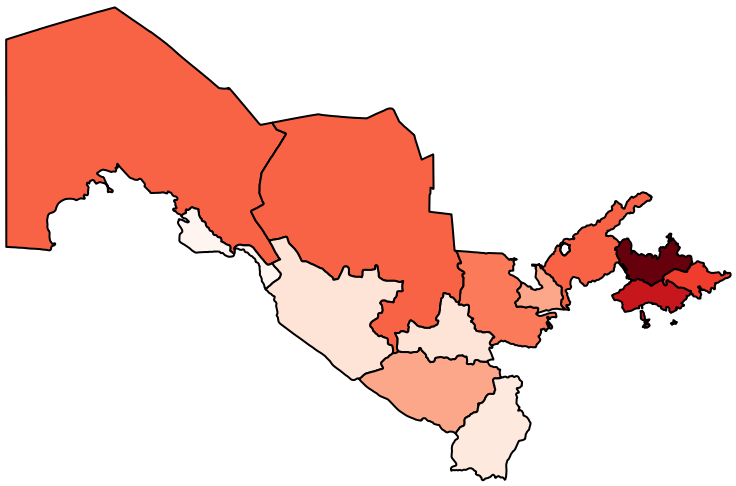



```
## removing the outlier, Tashkent city

test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2020), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(urban_percent == as.numeric(urban_percent)) %>%
  mutate(urban_percent = ifelse(Region == "Tashkent city", NA, urban_percent)) %>%
  select(urban_percent)

map_1@data$urban_percent <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(urban_percent))) +
    geom_histogram(bins=10, fill='#69b3a2', color='white')
```

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

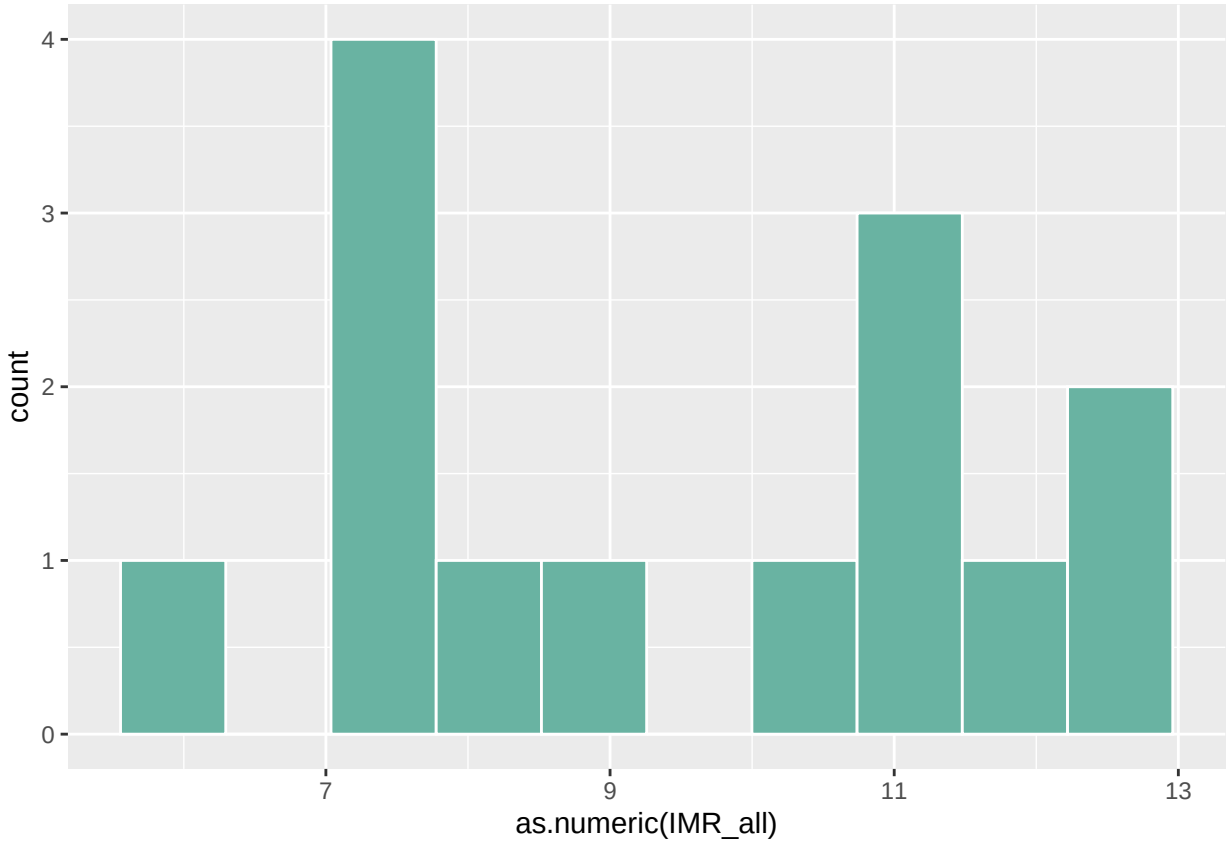



```
library(RColorBrewer)
my_colors <- brewer.pal(9, "Reds")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$urban_percent, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Urban percent (excl. Tashkent)")
```



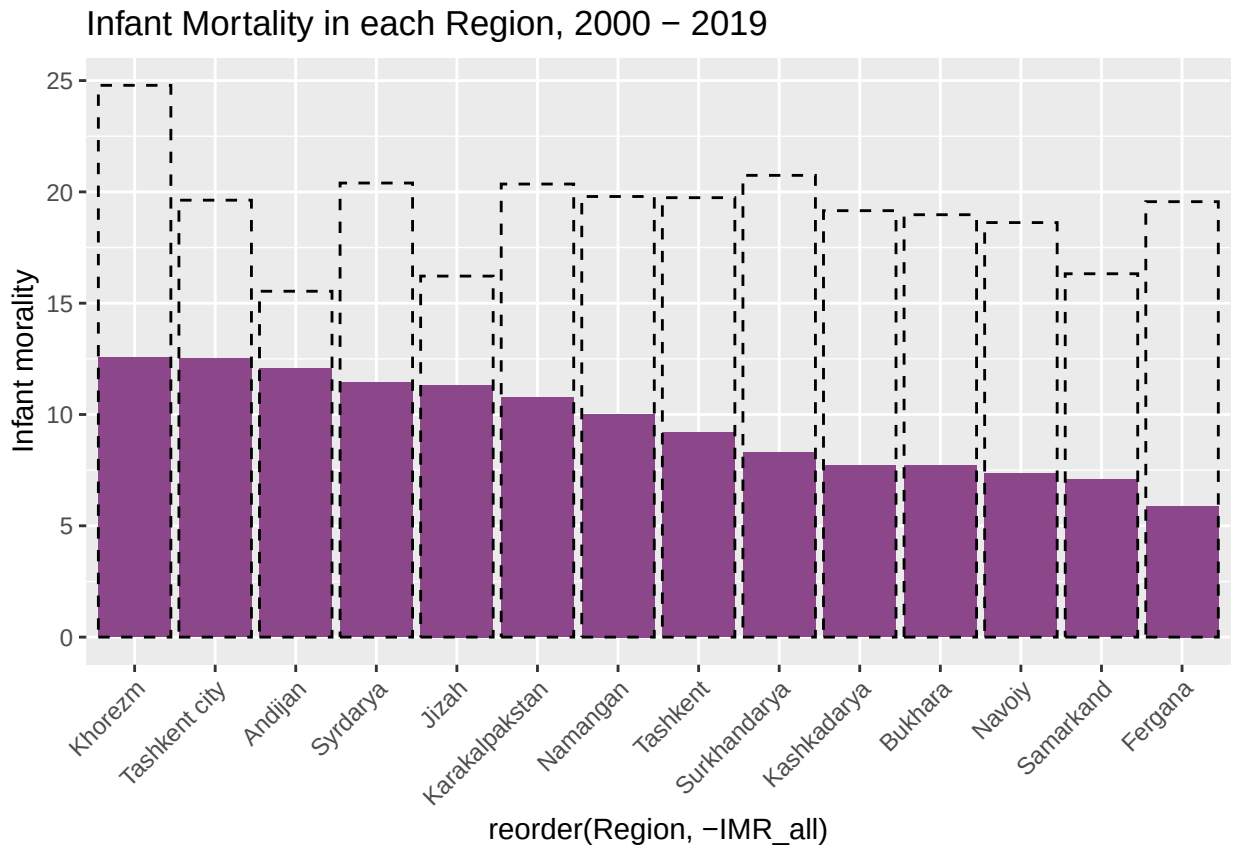

*Infant Mortality Rate*

```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(IMR_all == as.numeric(IMR_all)) %>%
  select(IMR_all)

map_1@data$IMR_all <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(IMR_all))) +
    geom_histogram(bins=10, fill='#69b3a2', color='white')
```

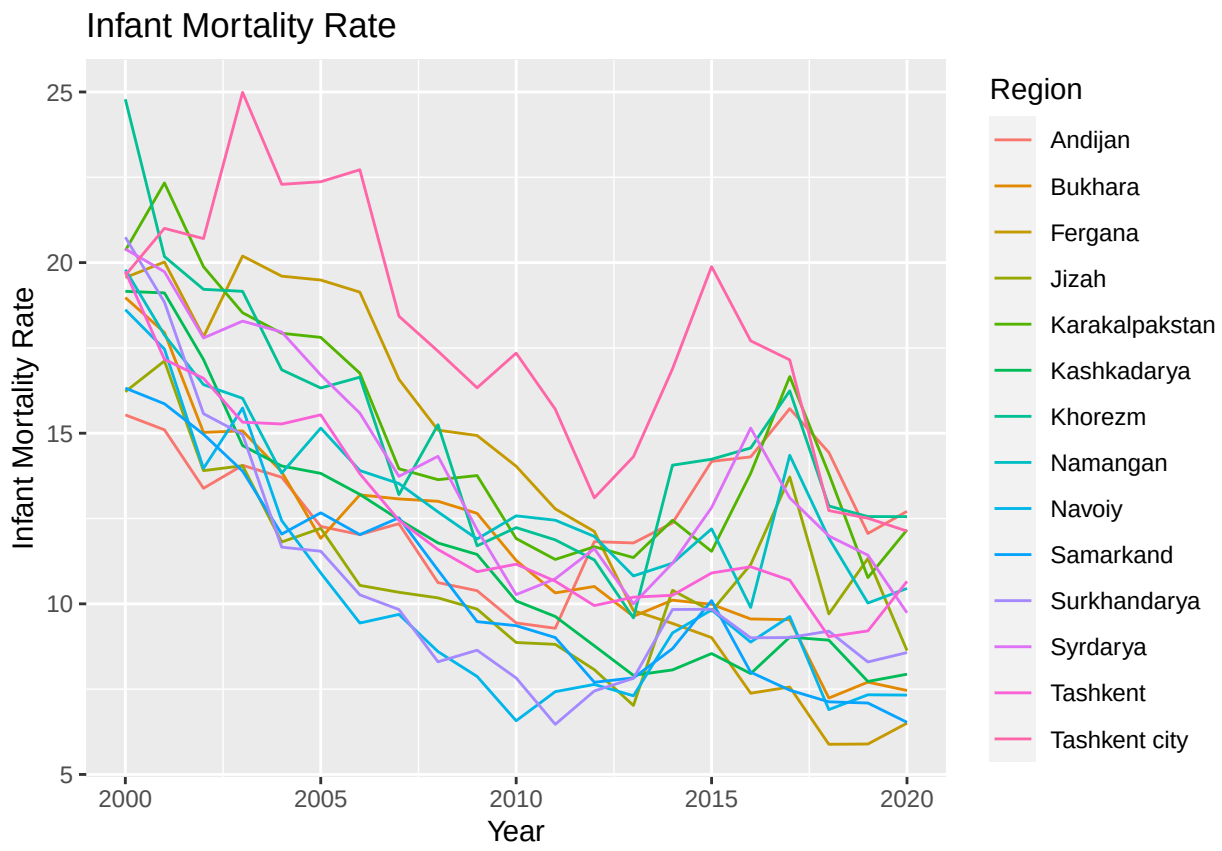



```
library(RColorBrewer)
my_colors <- brewer.pal(7, "Purples")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$IMR_all, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Infant mortality rate")
```


```
yrs <- uz_data %>%
  filter(infant_mortality_count_all != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- yrs$year[2]


uz_data %>%
  na.omit() %>%
  group_by(Region) %>%
  ggplot(mapping = aes(x=reorder(Region, -IMR_all), y=IMR_all)) +
  geom_bar(data = filter(uz_data, year == 2019), stat = "identity", linetype = "blank", fill = "orchid4
  geom_bar(data = filter(uz_data, year == 2000), stat = "identity", linetype = "dashed", fill = NA, col
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Infant Mortality in each Region, 2000 - 2019", y = "Infant morality")
```

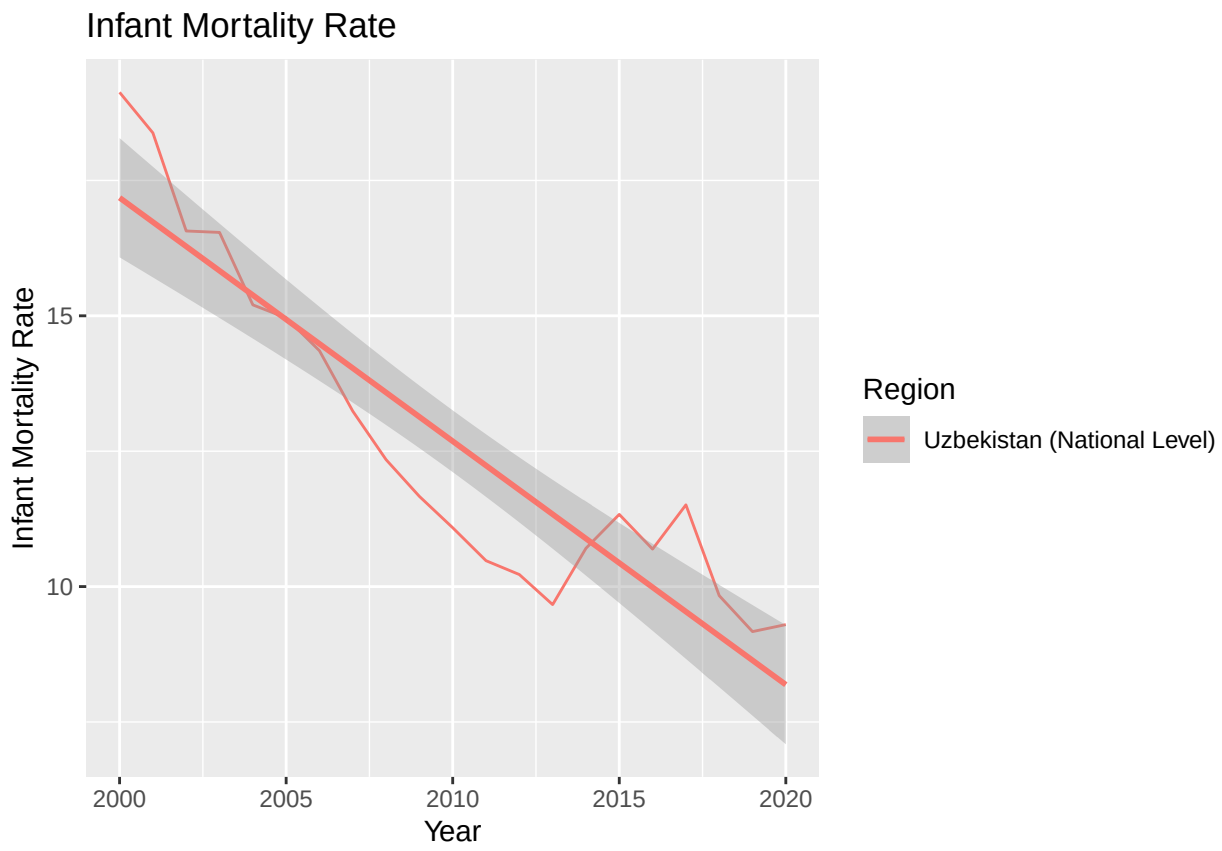



```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = IMR_all, color = Region)) +
  geom_line() +
  labs(title = "Infant Mortality Rate",
       x = "Year", y = "Infant Mortality Rate")
```



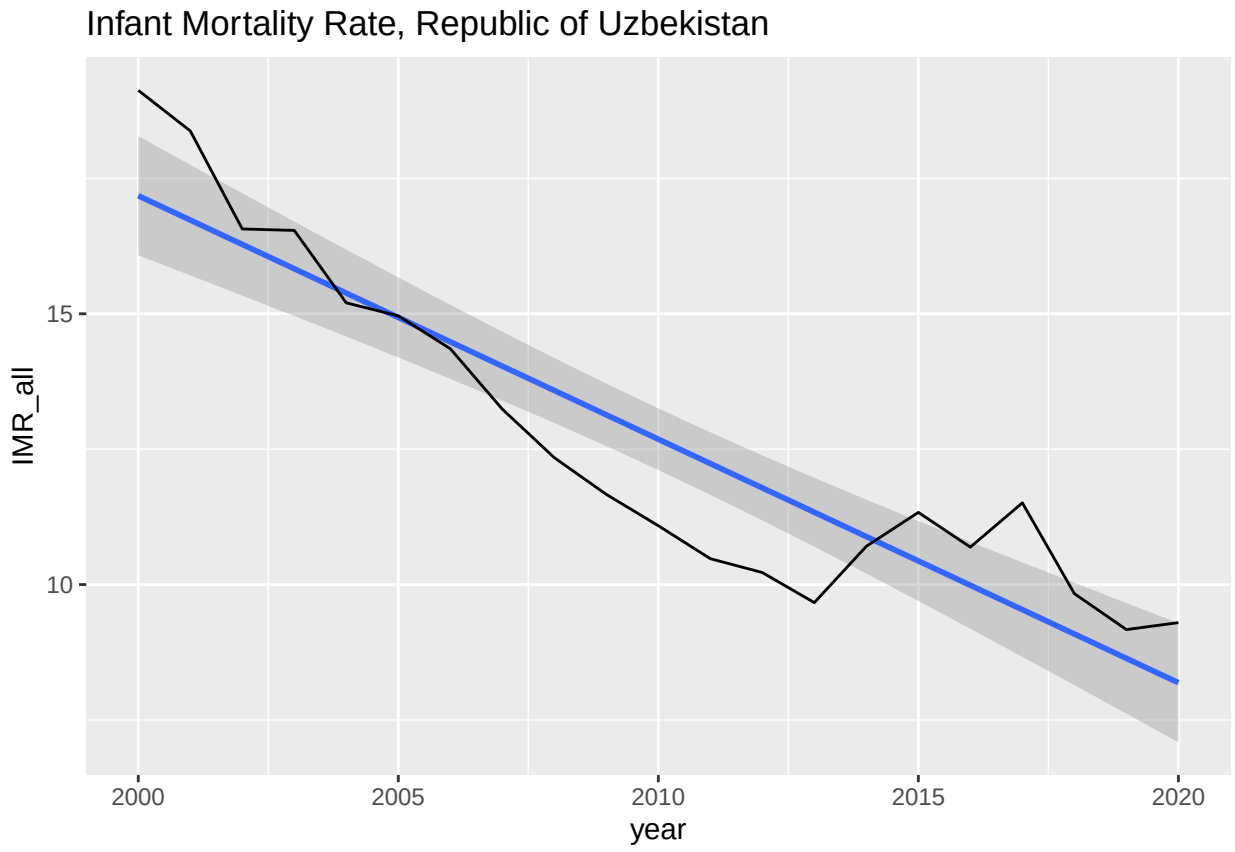

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = IMR_all, color = Region)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(title = "Infant Mortality Rate",
       x = "Year", y = "Infant Mortality Rate") +
  scale_fill_discrete(name = "Infant Mortality Rate", labels = c("", "B", "C"))
```

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




```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x=year,y=IMR_all)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "Infant Mortality Rate, Republic of Uzbekistan")
```

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




```
IMR_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(IMR_all ~ year, data = nat_uz_data)
tidy(IMR_model)
```

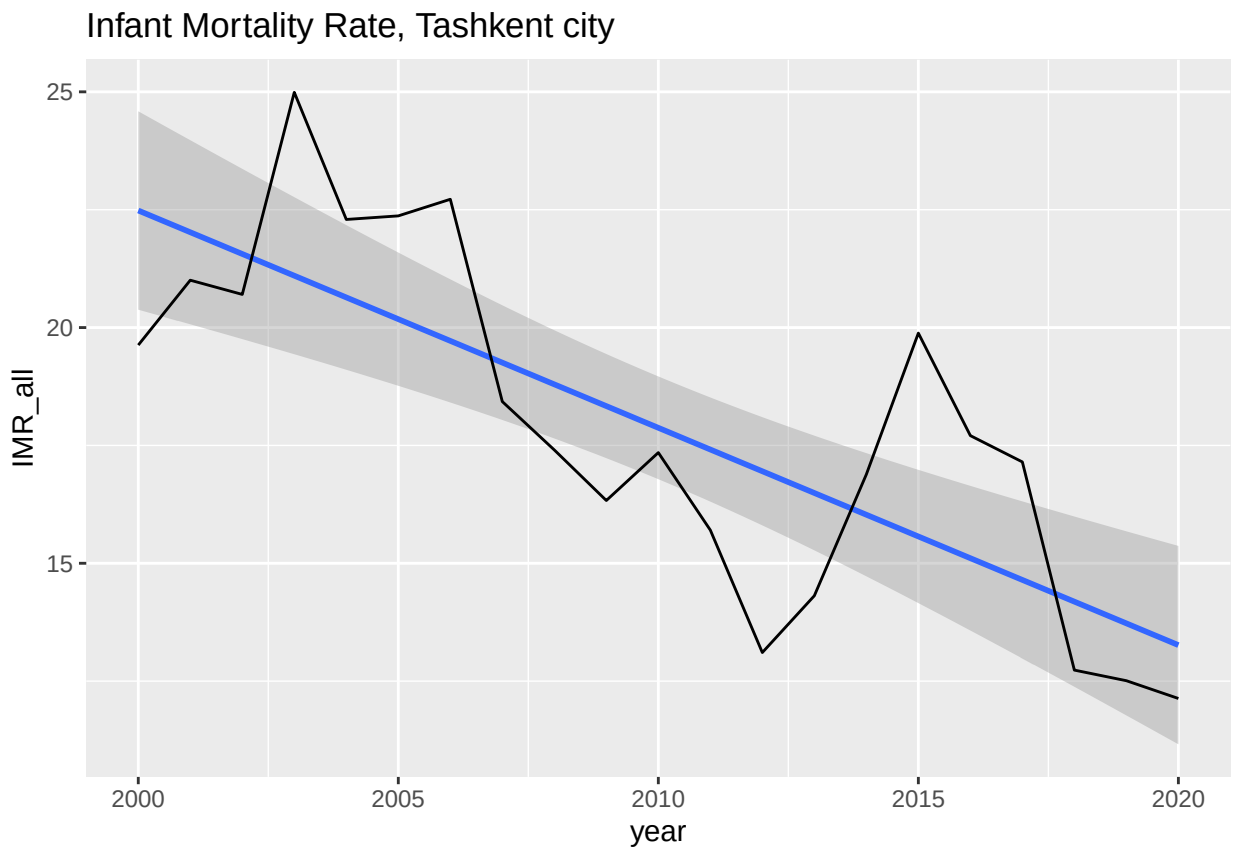
```
## # A tibble: 2 x 5
##   term        estimate std.error statistic       p.value
##   <chr>          <dbl>     <dbl>     <dbl>         <dbl>
## 1 (Intercept)  916.      90.4        10.1  0.00000000425
## 2 year          -0.450    0.0450     -9.99 0.00000000532
```

```
glance(IMR_model)$p.value < 0.01
```

```
## value
##  TRUE
```

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end, Region == "Tashkent city") %>%
  ggplot(aes(x=year,y=IMR_all)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "Infant Mortality Rate, Tashkent city")
```

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




```
IMR_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(IMR_all ~ year, data = filter(uz_data, Region == "Tashkent city"))
tidy(IMR_model)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic   p.value
##   <chr>          <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)  944.     173.          5.46 0.0000286
## 2 year          -0.461    0.0860     -5.36 0.0000358
```

```
glance(IMR_model)$p.value < 0.01
```

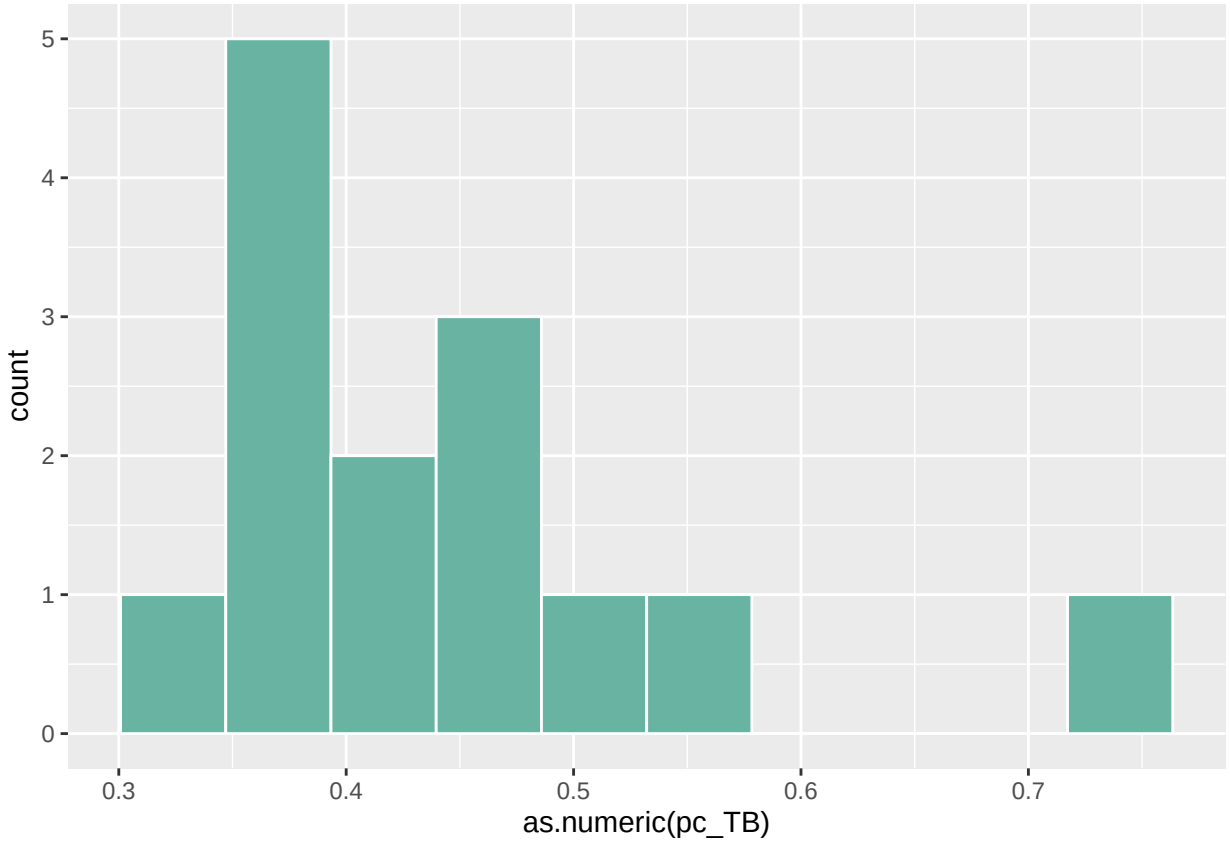
```
## value
##  TRUE
```

We have evidence to support that the infant mortality rate decreases by approximately -0.45 births per year in Uzbekistan (and -0.46 in Tashkent City) from 2000-2020.

*Diseases*

*Tuberculosis*

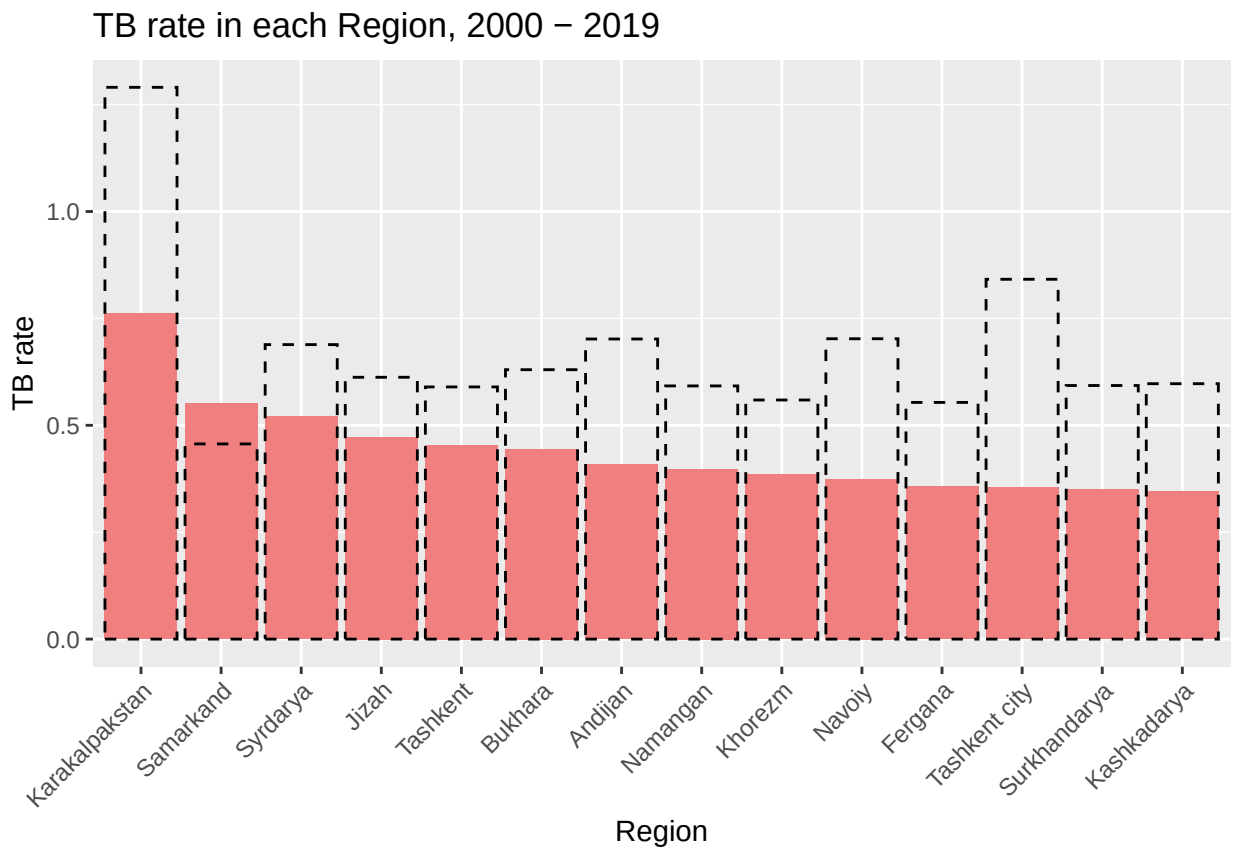
```r
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pc_TB == as.numeric(pc_TB)) %>%
  select(pc_TB)

map_1@data$pc_TB <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(pc_TB))) +
    geom_histogram(bins=10, fill='#69b3a2', color='white')
```



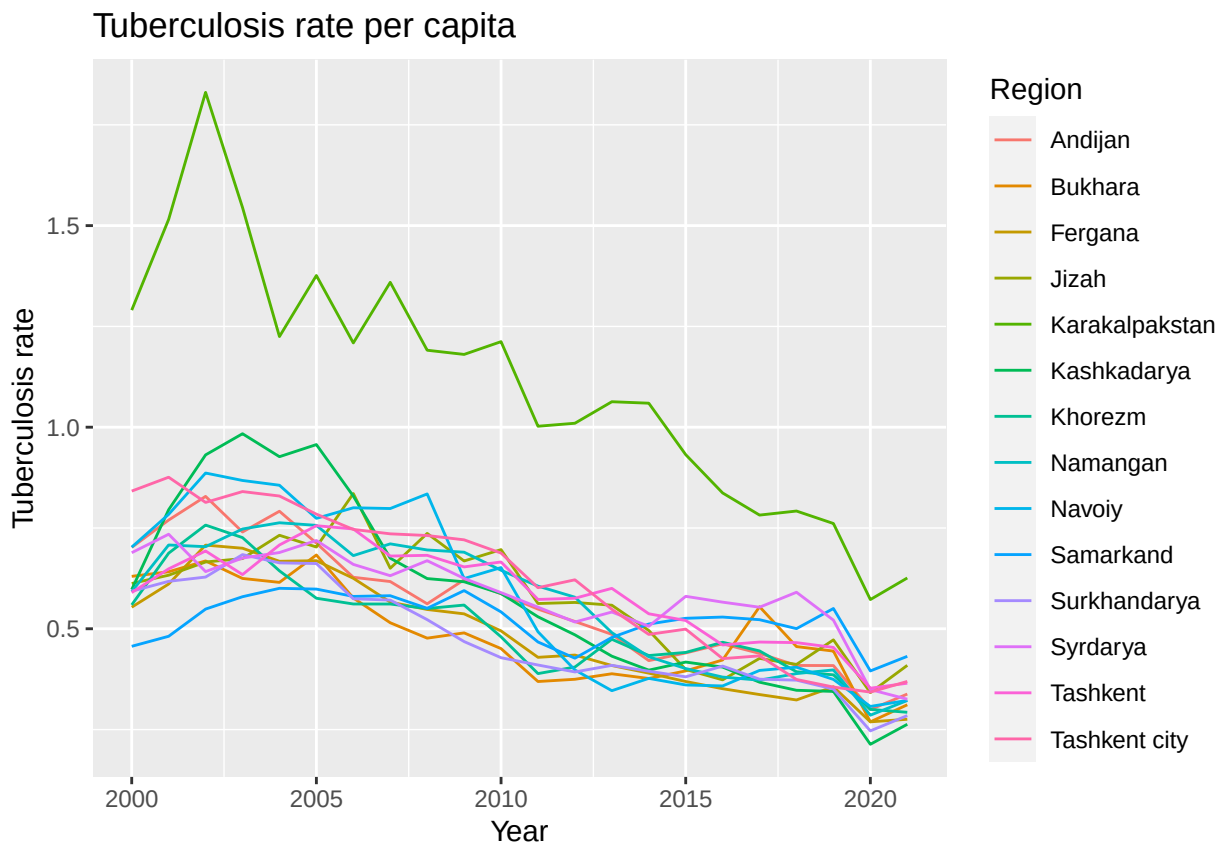

```r
my_colors <- brewer.pal(7, "Oranges")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$pc_TB, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Tuberculosis per capita, 2019")
```


Karakalpakstan seems to have the highest recorded TB rates.

```
yrs <- uz_data %>%
  filter(pc_TB != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- yrs$year[2]


uz_data %>%
  group_by(Region) %>%
  ggplot(mapping = aes(x=reorder(Region, -pc_TB), y=pc_TB)) +
  geom_bar(data = filter(uz_data, year == 2019), stat = "identity", linetype = "blank", fill = "lightco
  geom_bar(data = filter(uz_data, year == 2000), stat = "identity", linetype = "dashed", fill = NA, col
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "TB rate in each Region, 2000 - 2019", y = "TB rate", x = "Region")
```

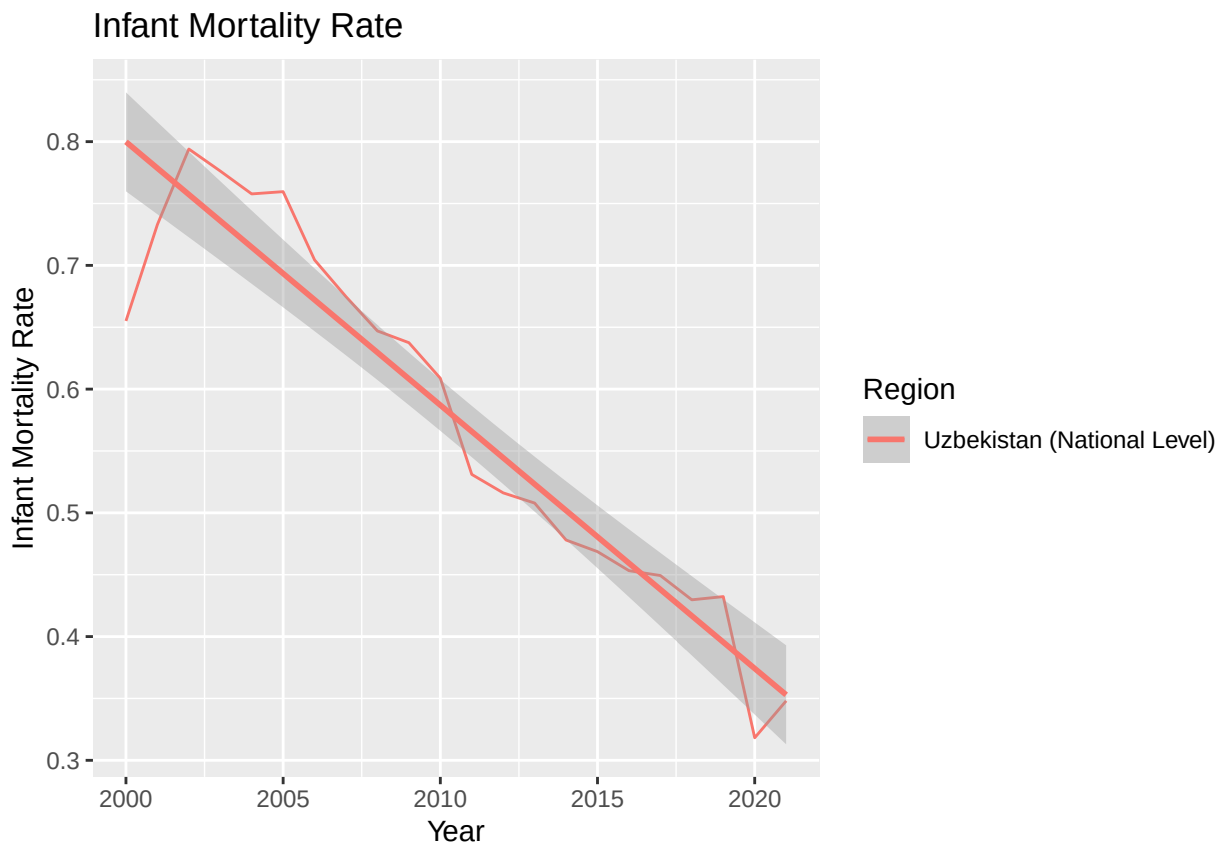



```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_TB, color = Region)) +
  geom_line() +
  labs(title = "Tuberculosis rate per capita",
       x = "Year", y = "Tuberculosis rate")
```



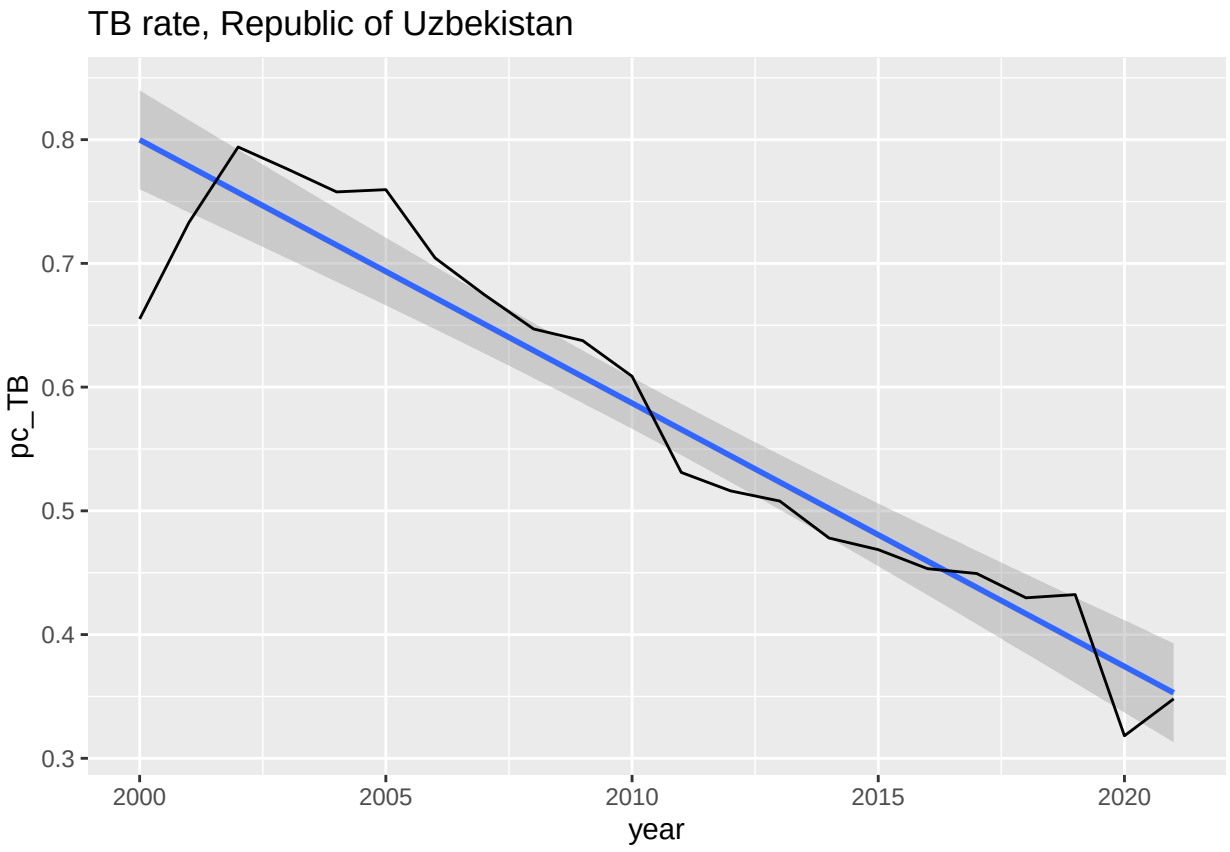

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_TB, color = Region)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(title = "Infant Mortality Rate",
       x = "Year", y = "Infant Mortality Rate")
```

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




```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x=year,y=pc_TB)) +
  geom_smooth(method = "lm") +
  geom_line() +
  labs(title = "TB rate, Republic of Uzbekistan")
```

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




```
TB_model <- linear_reg() %>%
  set_engine("lm") %>%
  fit(pc_TB ~ year, data = nat_uz_data)
tidy(TB_model)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  43.4      3.14         13.8 1.11e-11
## 2 year         -0.0213   0.00156     -13.6 1.42e-11
```

```
glance(TB_model)$p.value < 0.01
```

```
## value
##  TRUE
```

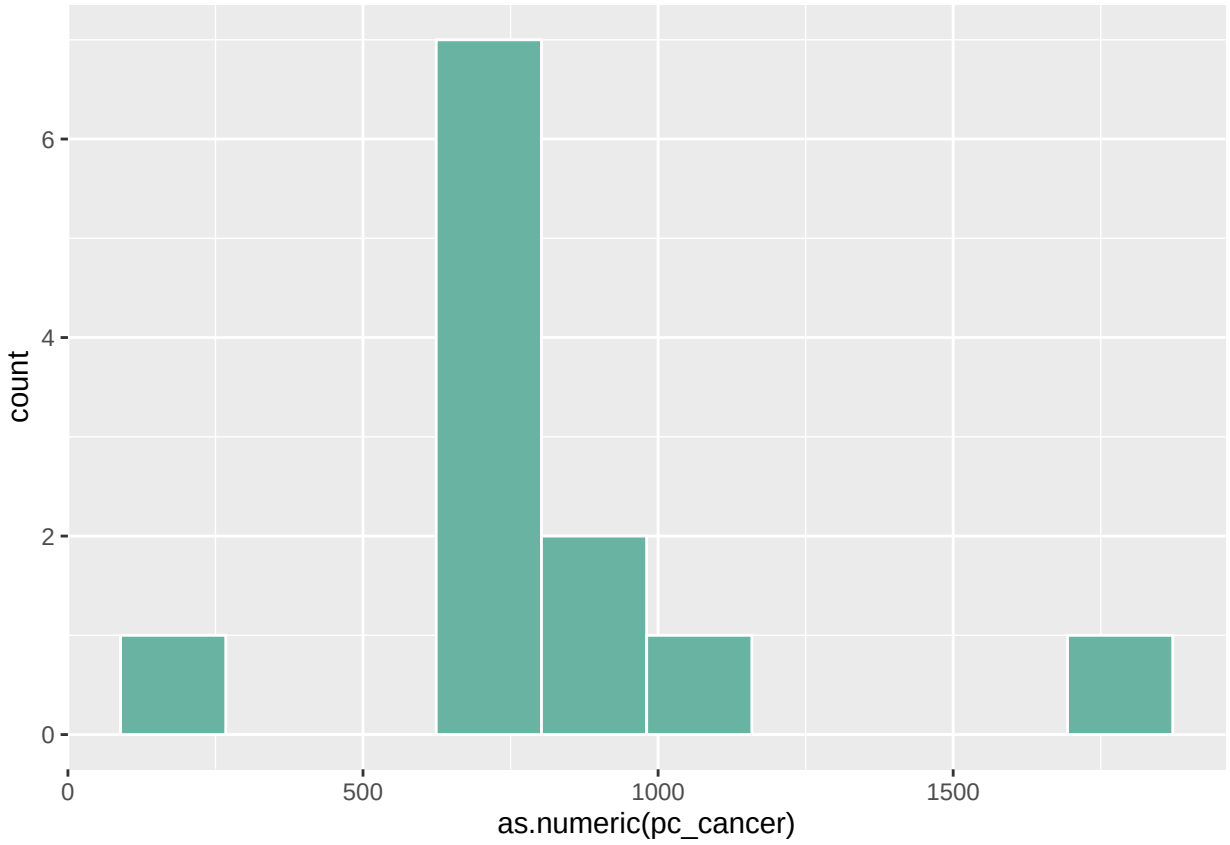
There is evidence to support that the TB rate decreases by 21.3 cases per 1000 people each year between 1991-2021 at the national level. ** these numbers feel wrong, but they are calculated to be per 1000 people

*Cancer*

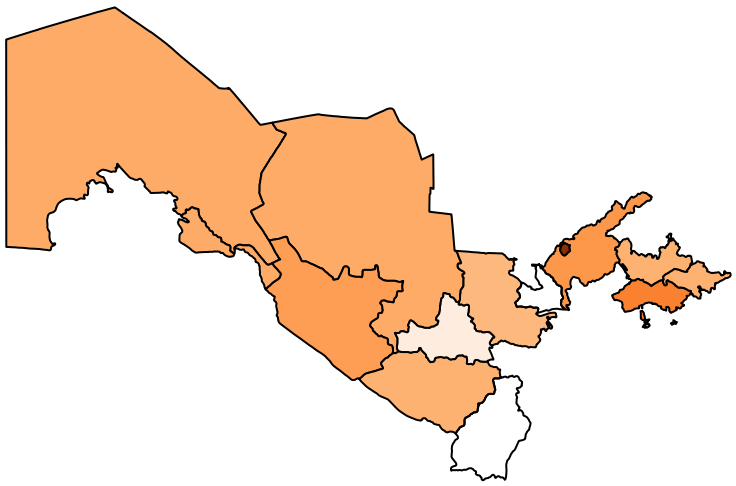
```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
```

```
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pc_cancer == as.numeric(pc_cancer)) %>%
  select(pc_cancer)

map_1@data$pc_cancer <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(pc_cancer))) +
    geom_histogram(bins=10, fill='#69b3a2', color='white')
```

```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```

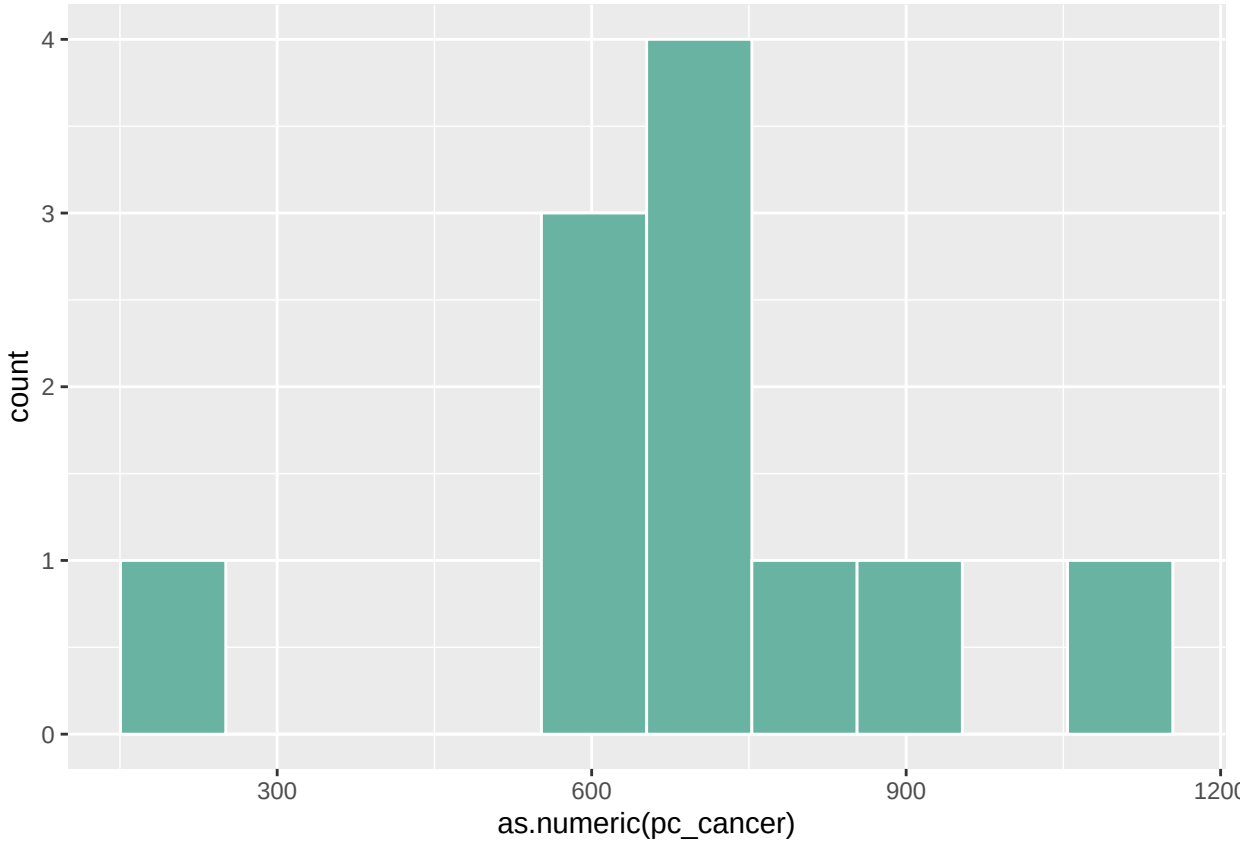



```
library(RColorBrewer)
my_colors <- brewer.pal(7, "Oranges")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$pc_cancer, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Cancer per capita, 2019")
```

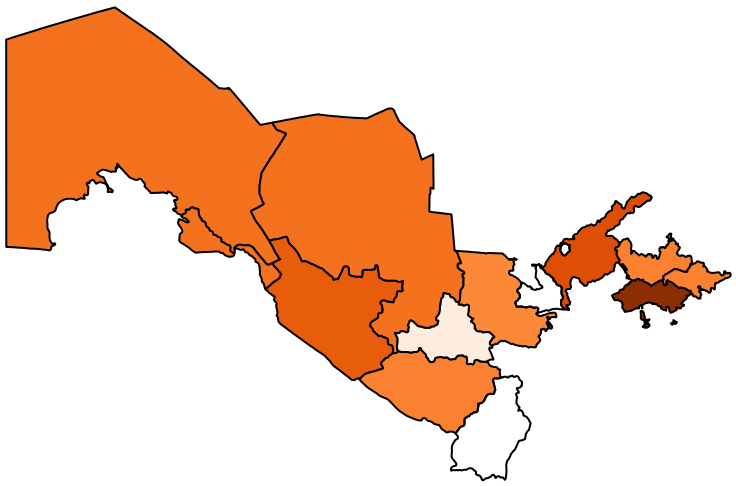



```
## excluding the outlier, Tashkent city

test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pc_cancer == as.numeric(pc_cancer)) %>%
  mutate(pc_cancer = ifelse(Region == "Tashkent city", NA, pc_cancer)) %>%
  select(pc_cancer)

map_1@data$pc_cancer <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(pc_cancer))) +
    geom_histogram(bins=10, fill='#69b3a2', color='white')
```

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

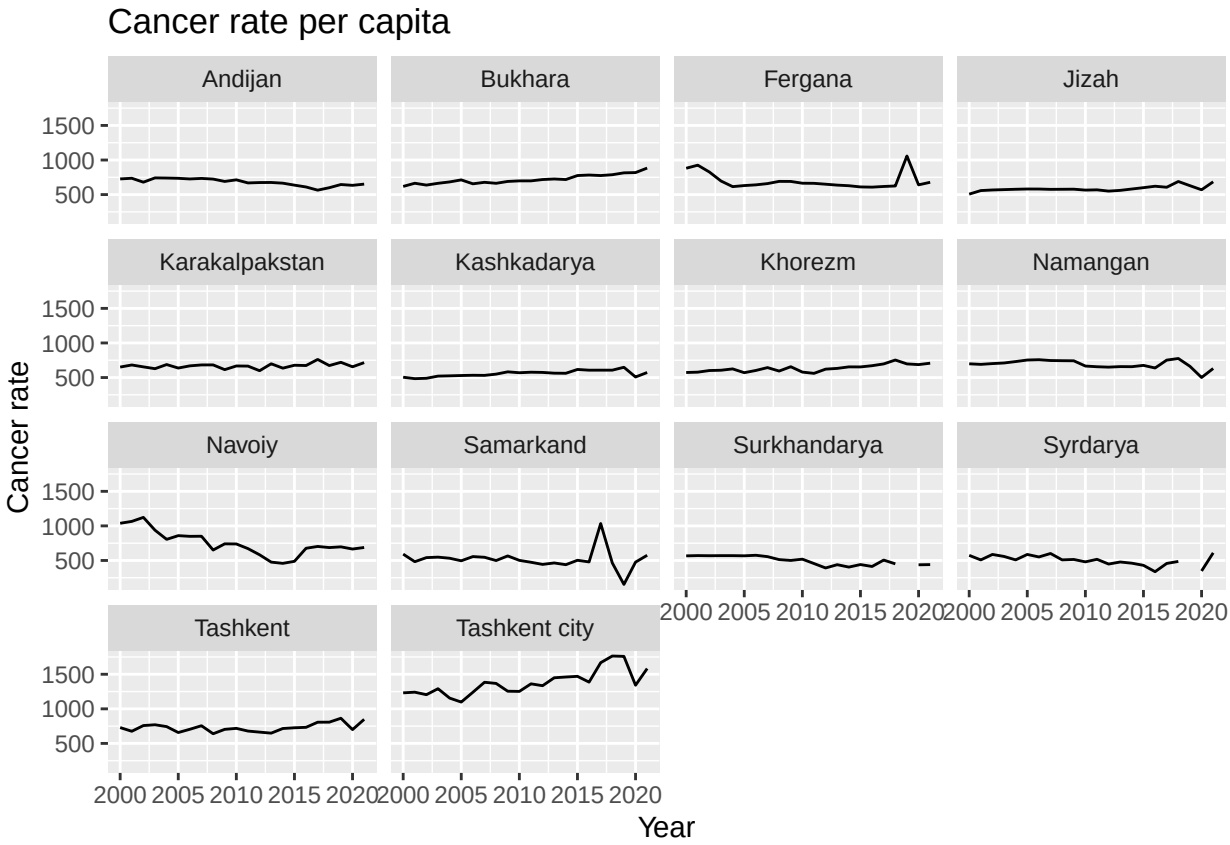



```
library(RColorBrewer)
my_colors <- brewer.pal(7, "Oranges")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$pc_cancer, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Cancer per capita, 2019, (excl. Tashkent city)")
```



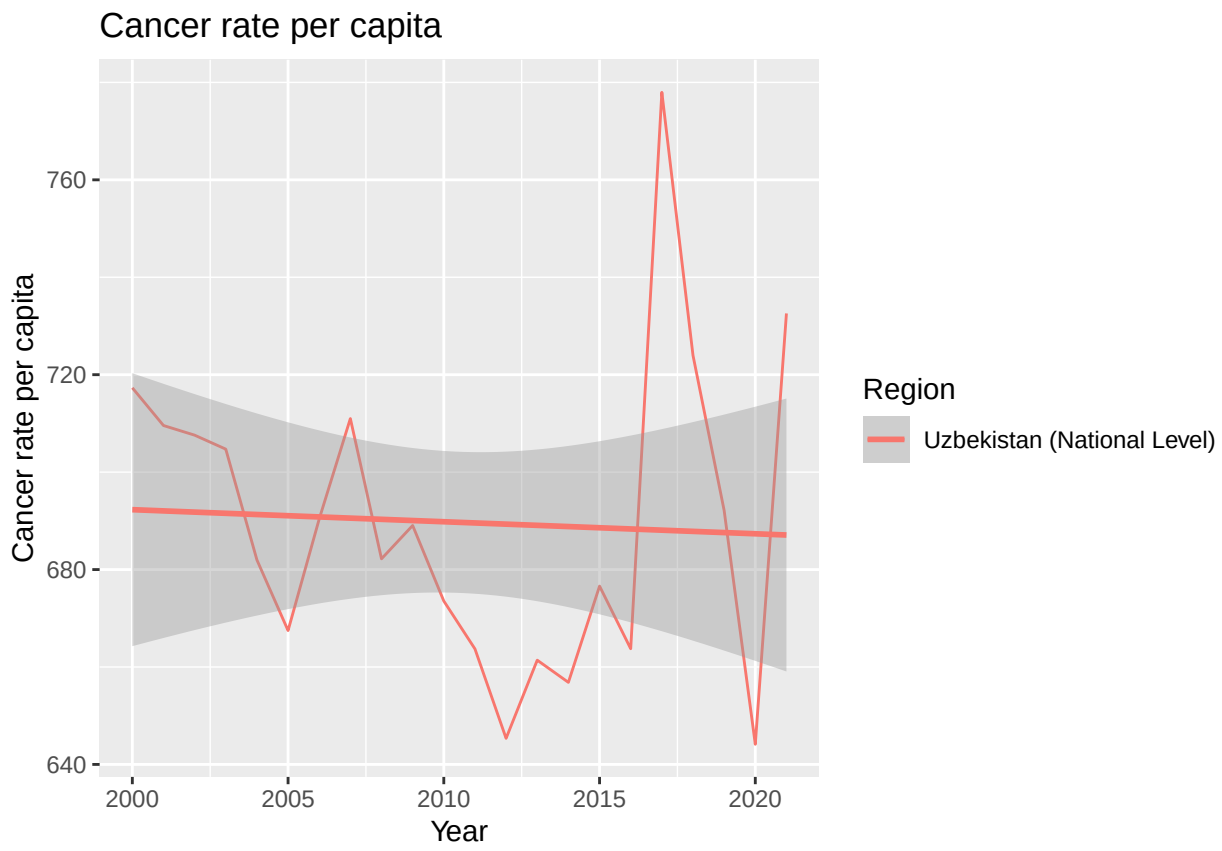

Andijan seems to have the highest cancer rates. The high recorded cancer rate in TB in Tashkent city is likely due to more robust diagnosis and treatment resources available in the city.

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_cancer)) +
  geom_line() +
  facet_wrap(vars(Region)) +
  labs(title = "Cancer rate per capita",
       x = "Year", y = "Cancer rate")
```



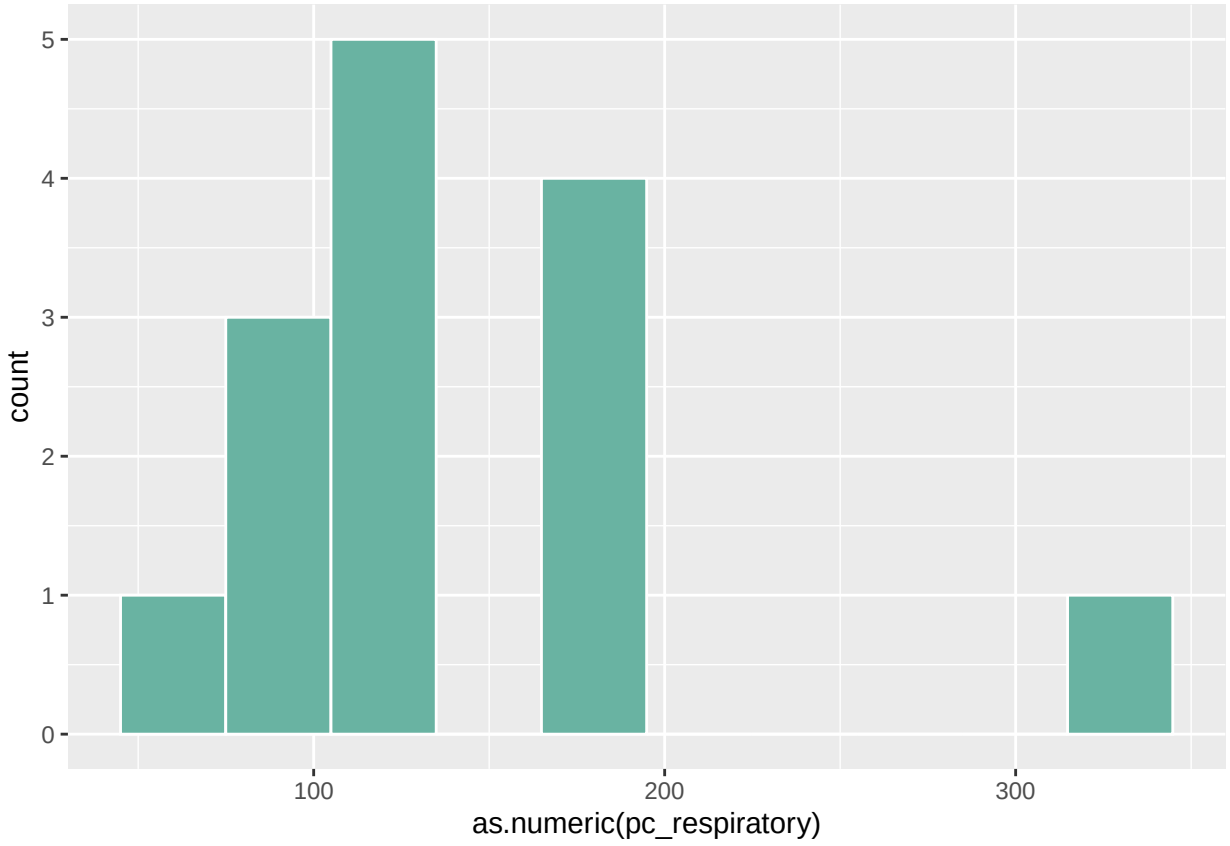

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_cancer, color = Region)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(title = "Cancer rate per capita",
       x = "Year", y = "Cancer rate per capita")
```

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

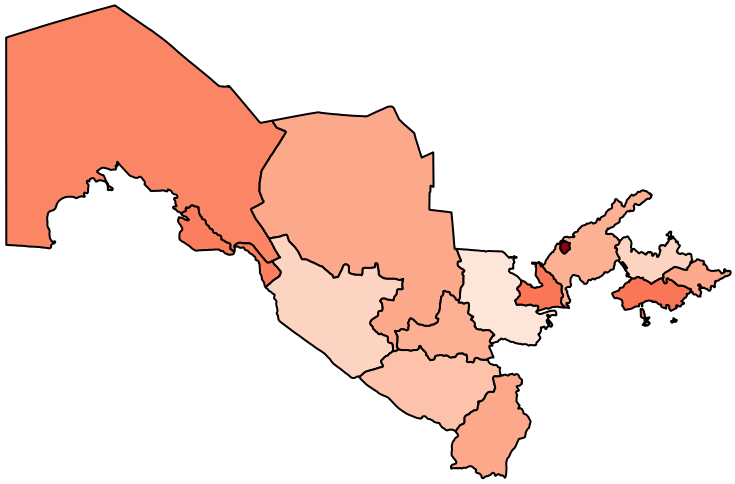



*Respiratory diseases*

```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pc_respiratory == as.numeric(pc_respiratory)) %>%
  select(pc_respiratory)

map_1@data$pc_respiratory <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(pc_respiratory))) +
    geom_histogram(bins=10, fill='#69b3a2', color='white')
```

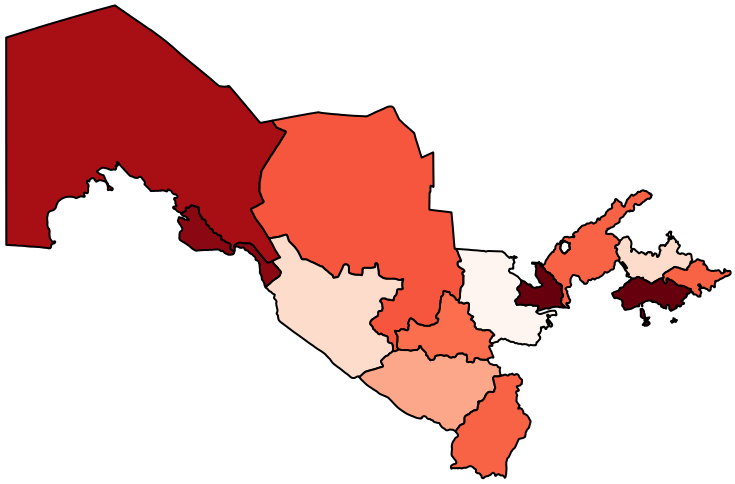



```
library(RColorBrewer)
my_colors <- brewer.pal(7, "Reds")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$pc_respiratory, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Respiratory diseases per capita, 2019")
```

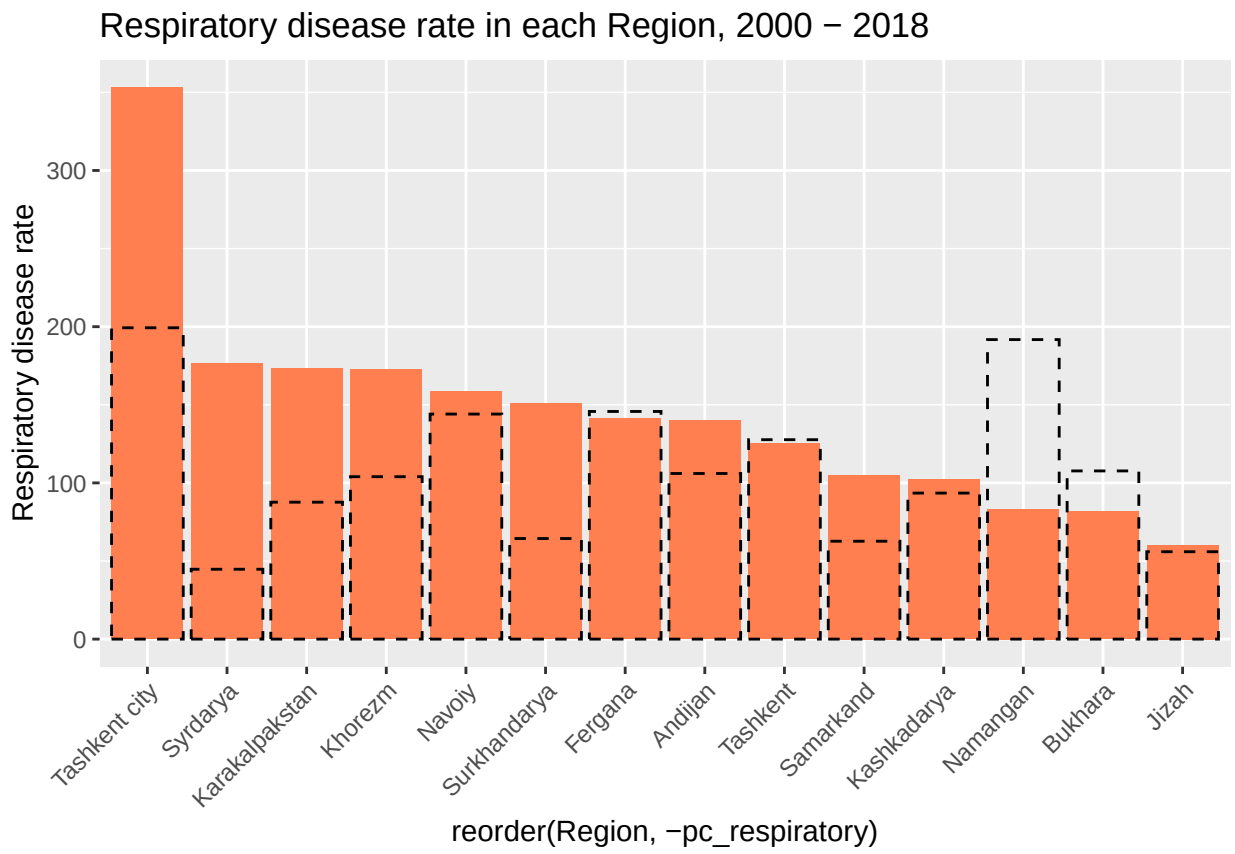



```
## excluding the outlier, tashkent city

test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pc_respiratory == as.numeric(pc_respiratory)) %>%
  mutate(pc_respiratory = ifelse(Region == "Tashkent city", NA, pc_respiratory)) %>%
  select(pc_respiratory)

map_1@data$pc_respiratory <- as.numeric(unlist(test))

library(RColorBrewer)
my_colors <- brewer.pal(9, "Reds")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(test$pc_respiratory, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Respiratory Disease rate, 2019 (excl. Tashkent city)")
```

## Respiratory Disease rate, 2019 (excl. Tashkent city)

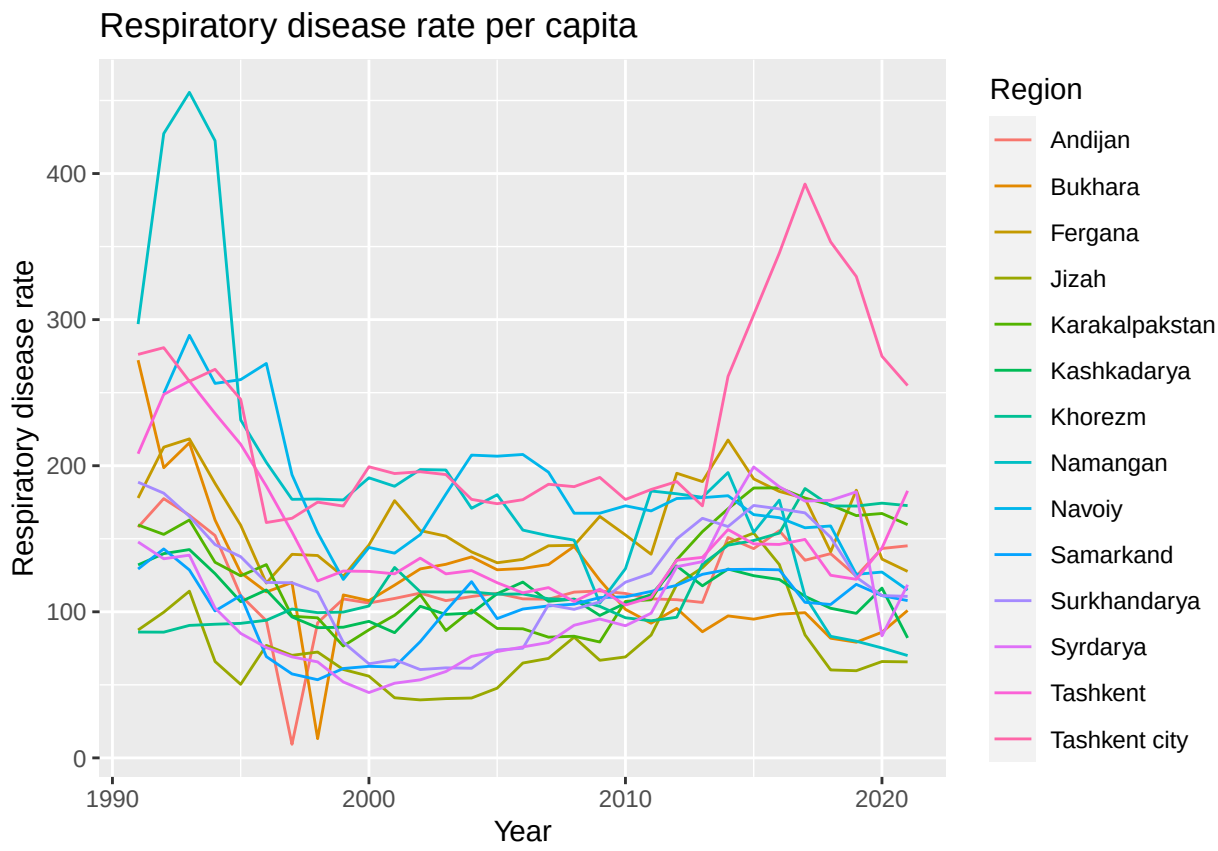


```
yrs <- uz_data %>%
  filter(pc_respiratory != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- yrs$year[2]


uz_data %>%
  na.omit() %>%
  group_by(Region) %>%
  ggplot(mapping = aes(x=reorder(Region, -pc_respiratory), y=pc_respiratory)) +
  geom_bar(data = filter(uz_data, year == 2018), stat = "identity", linetype = "blank", fill = "coral")
  geom_bar(data = filter(uz_data, year == 2000), stat = "identity", linetype = "dashed", fill = NA, col
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Respiratory disease rate in each Region, 2000 - 2018", y = "Respiratory disease rate")
```



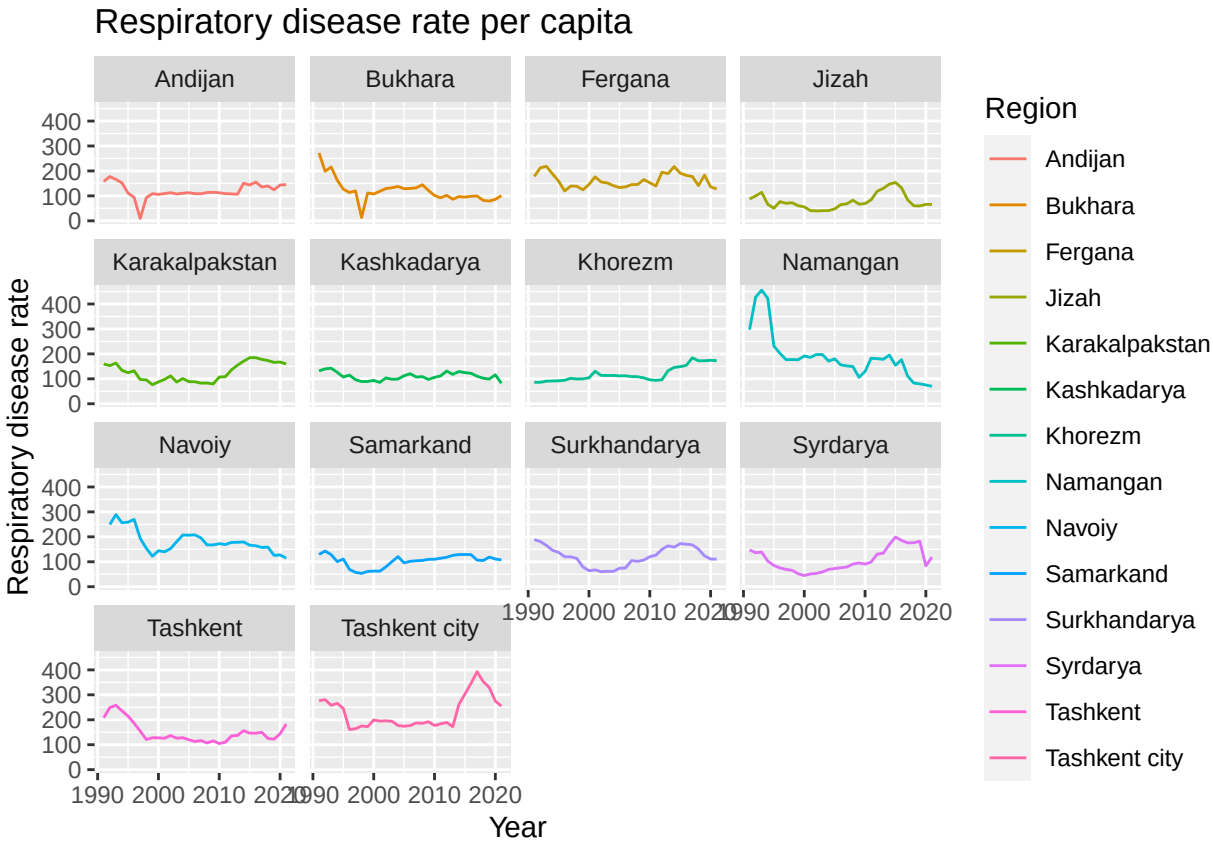

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_respiratory, color = Region)) +
  geom_line() +
  labs(title = "Respiratory disease rate per capita",
       x = "Year", y = "Respiratory disease rate")
```

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



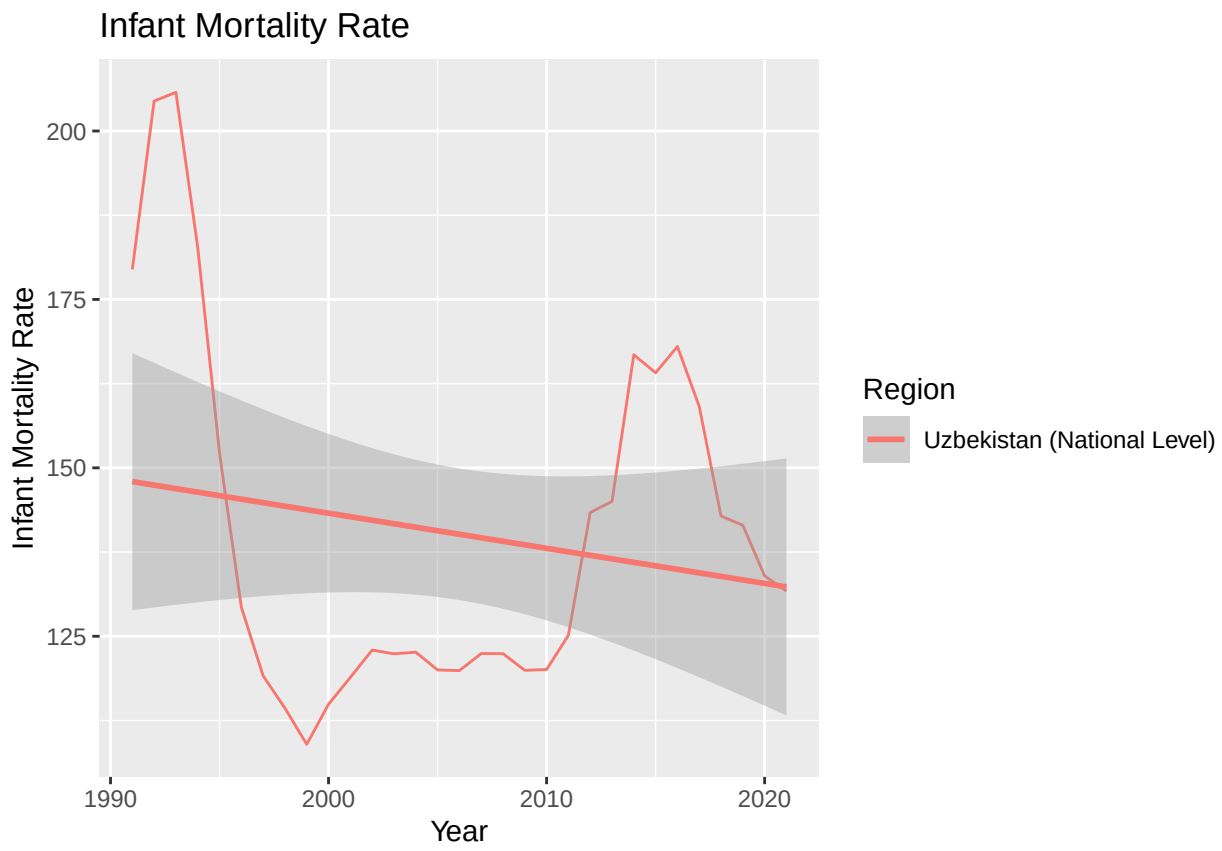

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_respiratory, color = Region)) +
  facet_wrap(Region ~ .) +
  geom_line() +
  labs(title = "Respiratory disease rate per capita",
       x = "Year", y = "Respiratory disease rate")
```

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



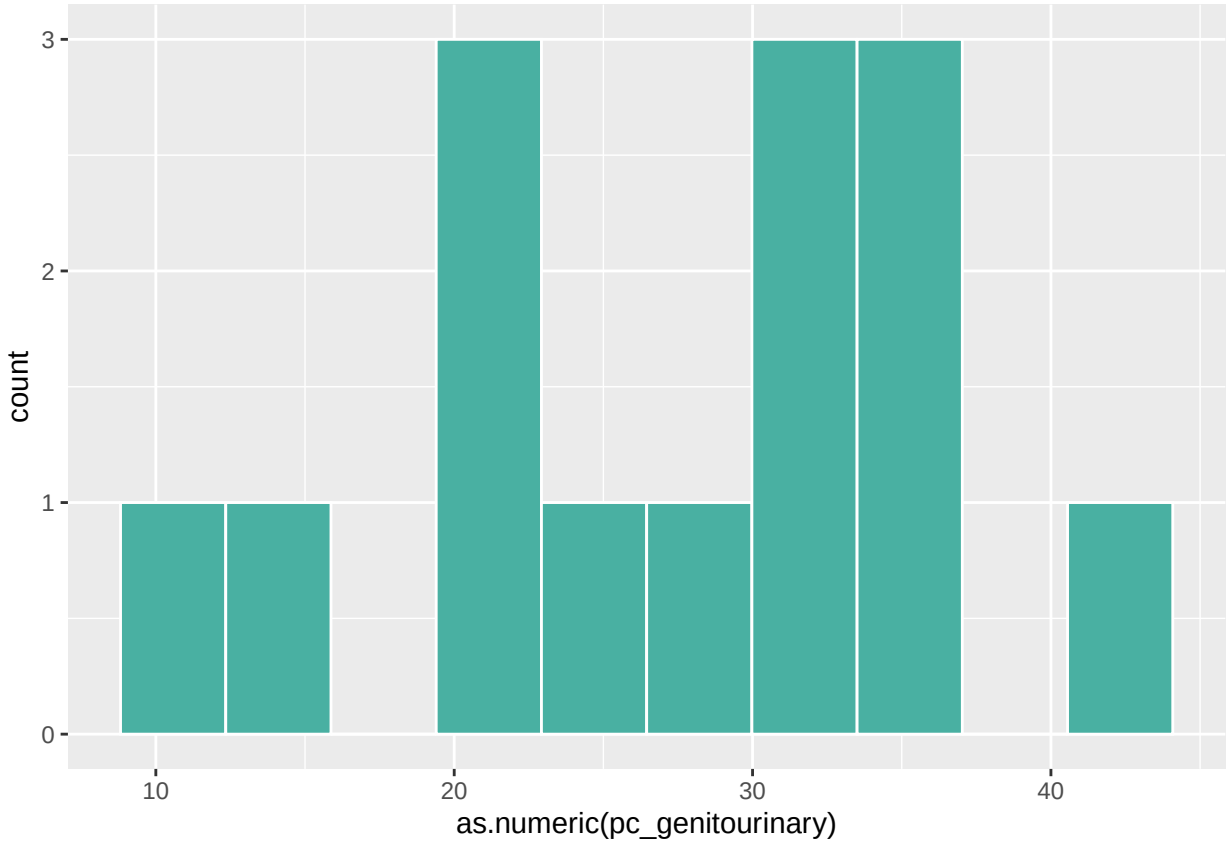

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_respiratory, color = Region)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(title = "Infant Mortality Rate",
       x = "Year", y = "Infant Mortality Rate") +
  scale_fill_discrete(name = "Infant Mortality Rate", labels = c("", "B", "C"))
```

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



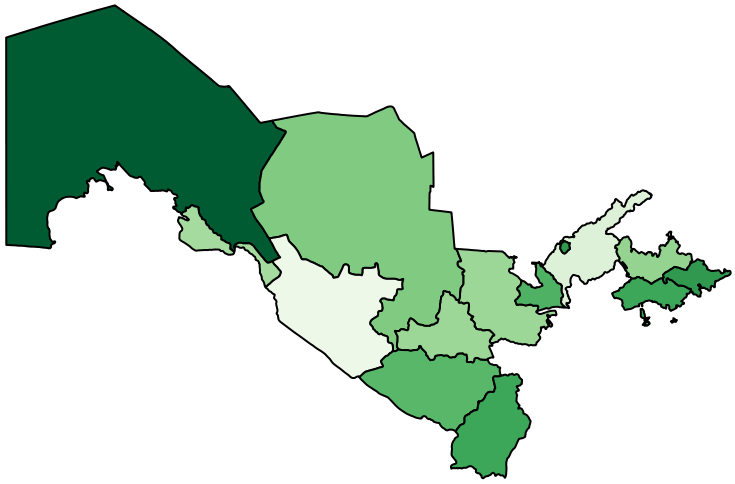

*Genitourinary diseases*

```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pc_genitourinary == as.numeric(pc_genitourinary)) %>%
  select(pc_genitourinary)

map_1@data$pc_genitourinary <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(pc_genitourinary))) +
    geom_histogram(bins=10, fill='#49b0a2', color='white')
```

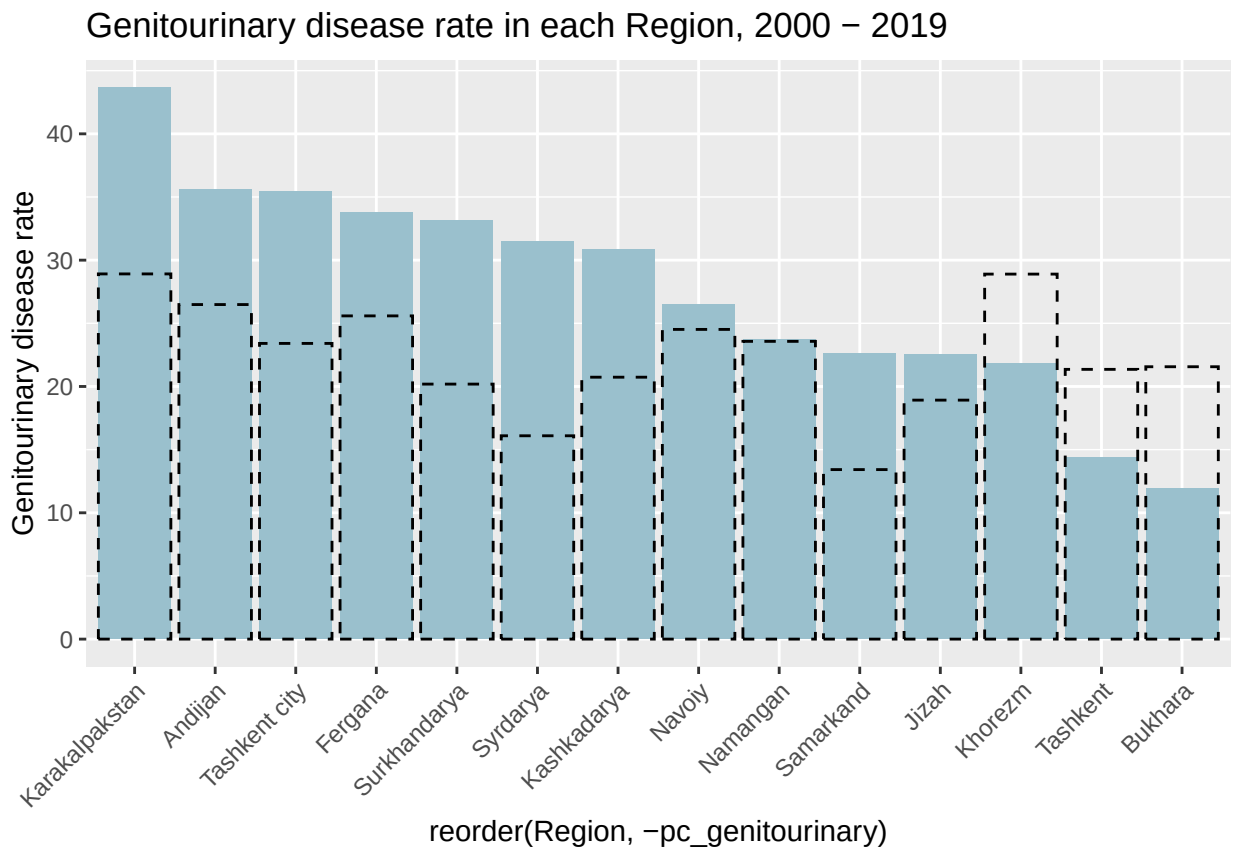



```
library(RColorBrewer)
my_colors <- brewer.pal(7, "Greens")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$pc_genitourinary, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Genitourinary diseases per capita, 2019")
```

## Genitourinary diseases per capita, 2019

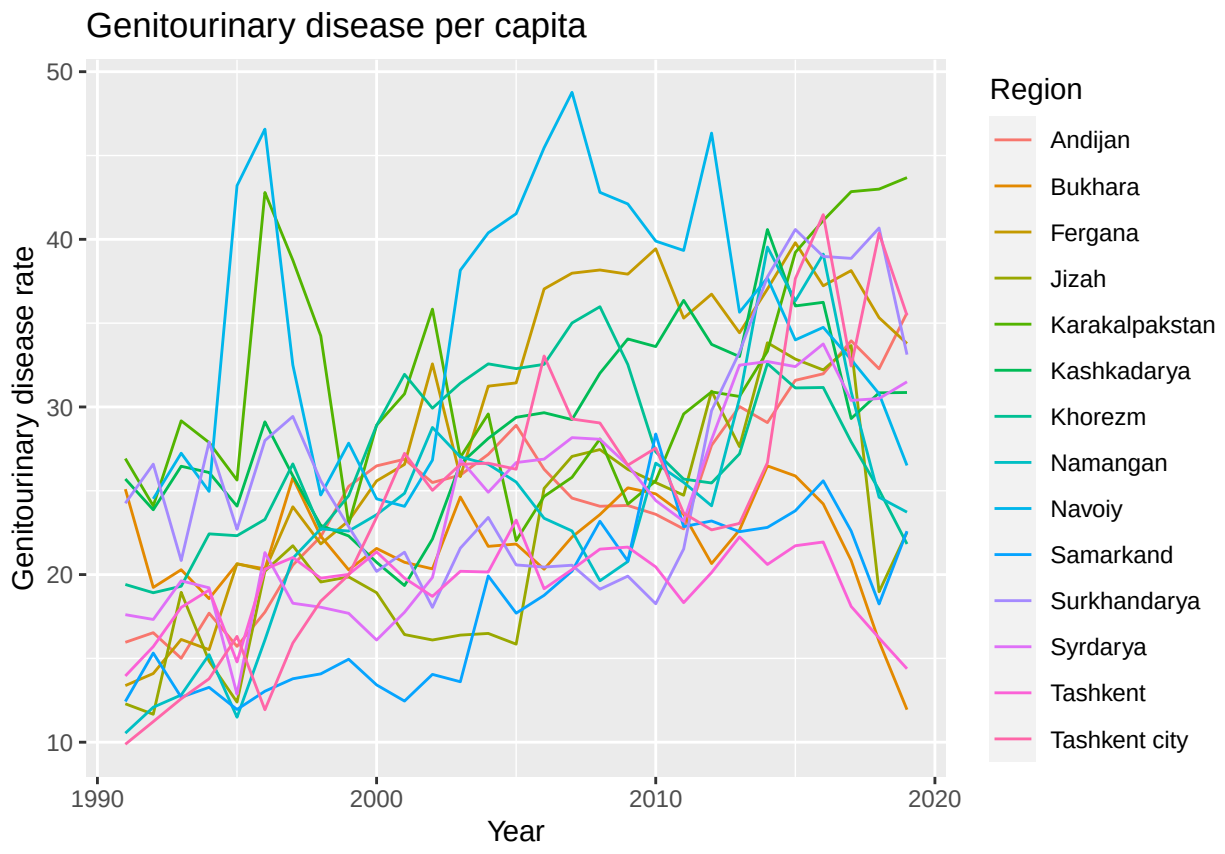


```
yrs <- uz_data %>%
  filter(pc_genitourinary != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- 2019 #weird data


uz_data %>%
  na.omit() %>%
  group_by(Region) %>%
  ggplot(mapping = aes(x=reorder(Region, -pc_genitourinary), y=pc_genitourinary)) +
  geom_bar(data = filter(uz_data, year == 2019), stat = "identity", linetype = "blank", fill = "lightblu
  geom_bar(data = filter(uz_data, year == 2000), stat = "identity", linetype = "dashed", fill = NA, colo
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Genitourinary disease rate in each Region, 2000 - 2019", y = "Genitourinary disease rate
```



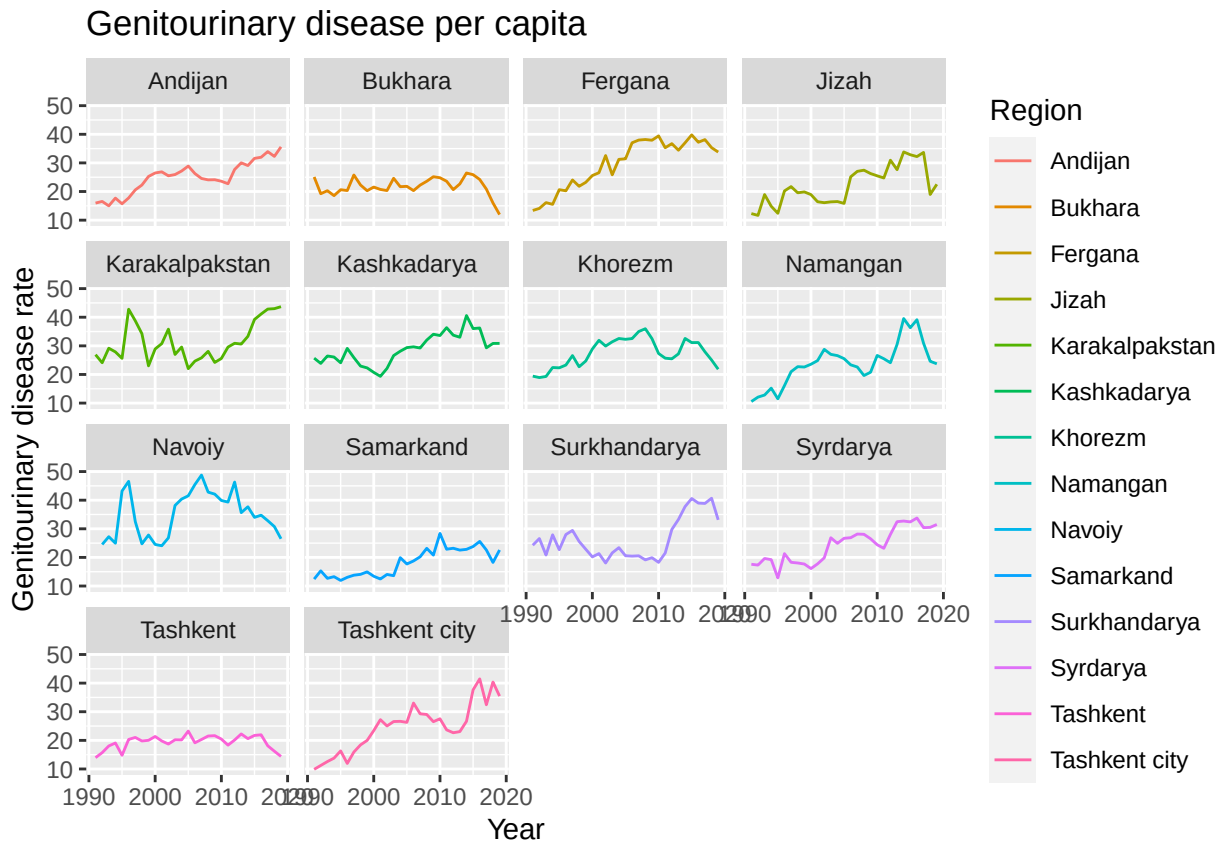

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_genitourinary, color = Region)) +
  geom_line() +
  labs(title = "Genitourinary disease per capita",
       x = "Year", y = "Genitourinary disease rate")
```

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




```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_genitourinary, color = Region)) +
  geom_line() +
  facet_wrap(Region ~ .) +
  labs(title = "Genitourinary disease per capita",
       x = "Year", y = "Genitourinary disease rate")
```

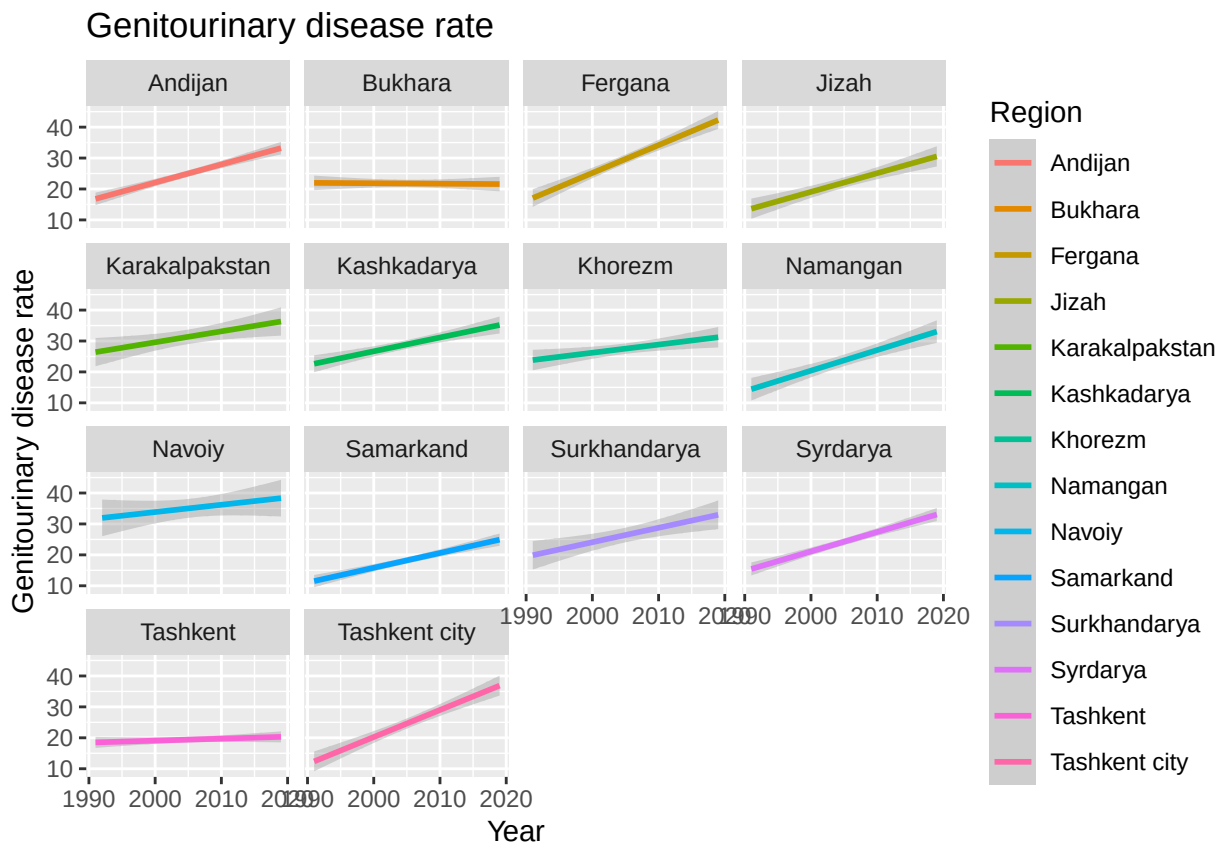
```
## Warning: Removed 1 row containing missing values (‘geom_line()‘).
```




```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_genitourinary, color = Region)) +
  geom_smooth(method = "lm") +
  facet_wrap(Region ~ .) +
  labs(title = "Genitourinary disease rate",
       x = "Year", y = "Genitourinary disease rate")
```

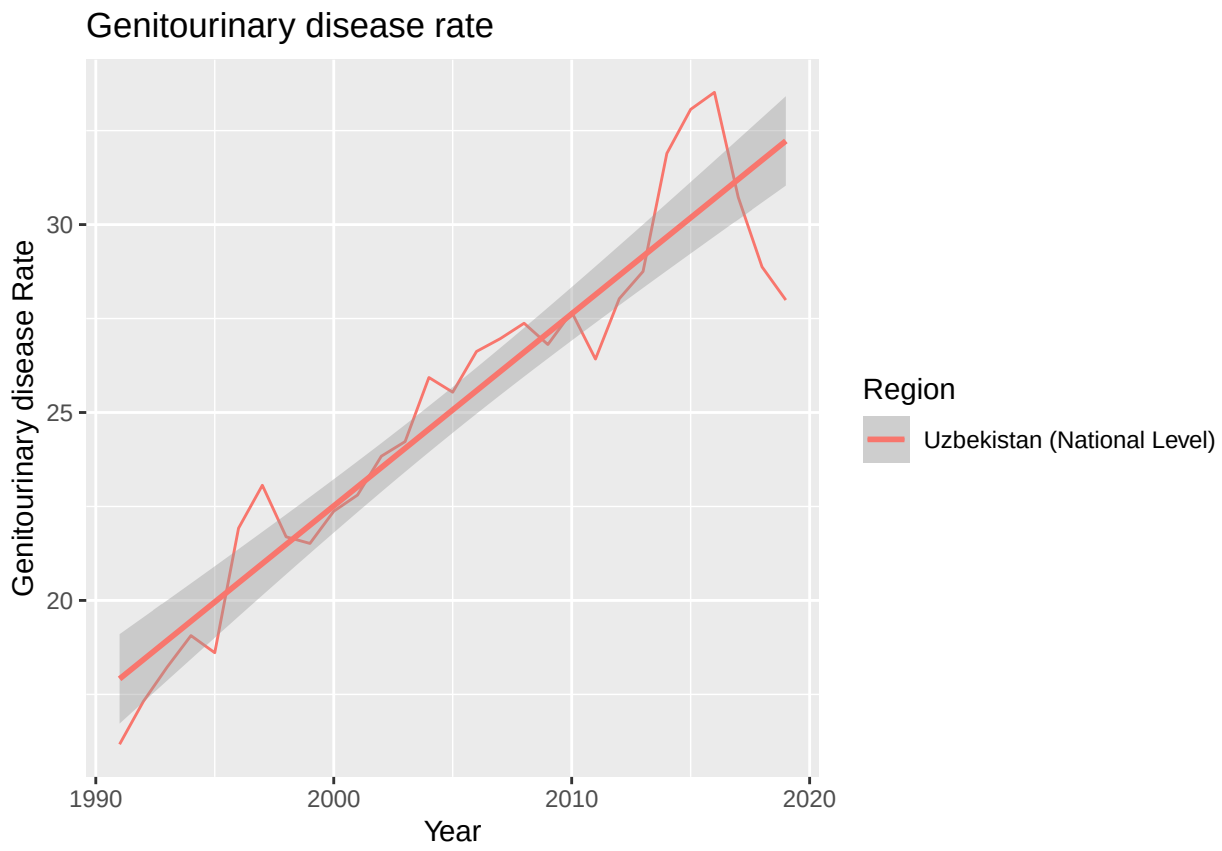
```
## `geom_smooth()` using formula = 'y ~ x'
```

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



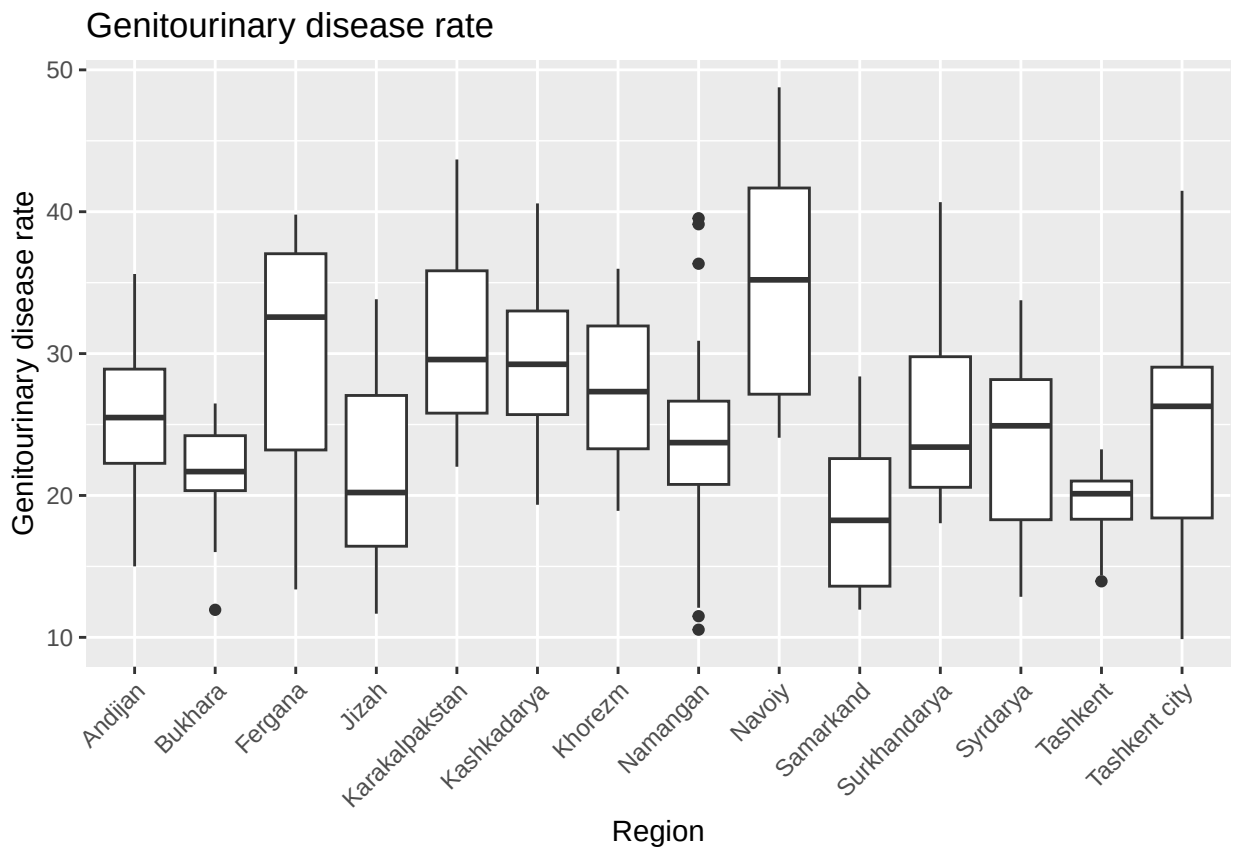

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_genitourinary, color = Region)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(title = "Genitourinary disease rate",
       x = "Year", y = "Genitourinary disease Rate") +
  scale_fill_discrete(name = "Genitourinary disease Rate", labels = c("", "B", "C"))
```

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



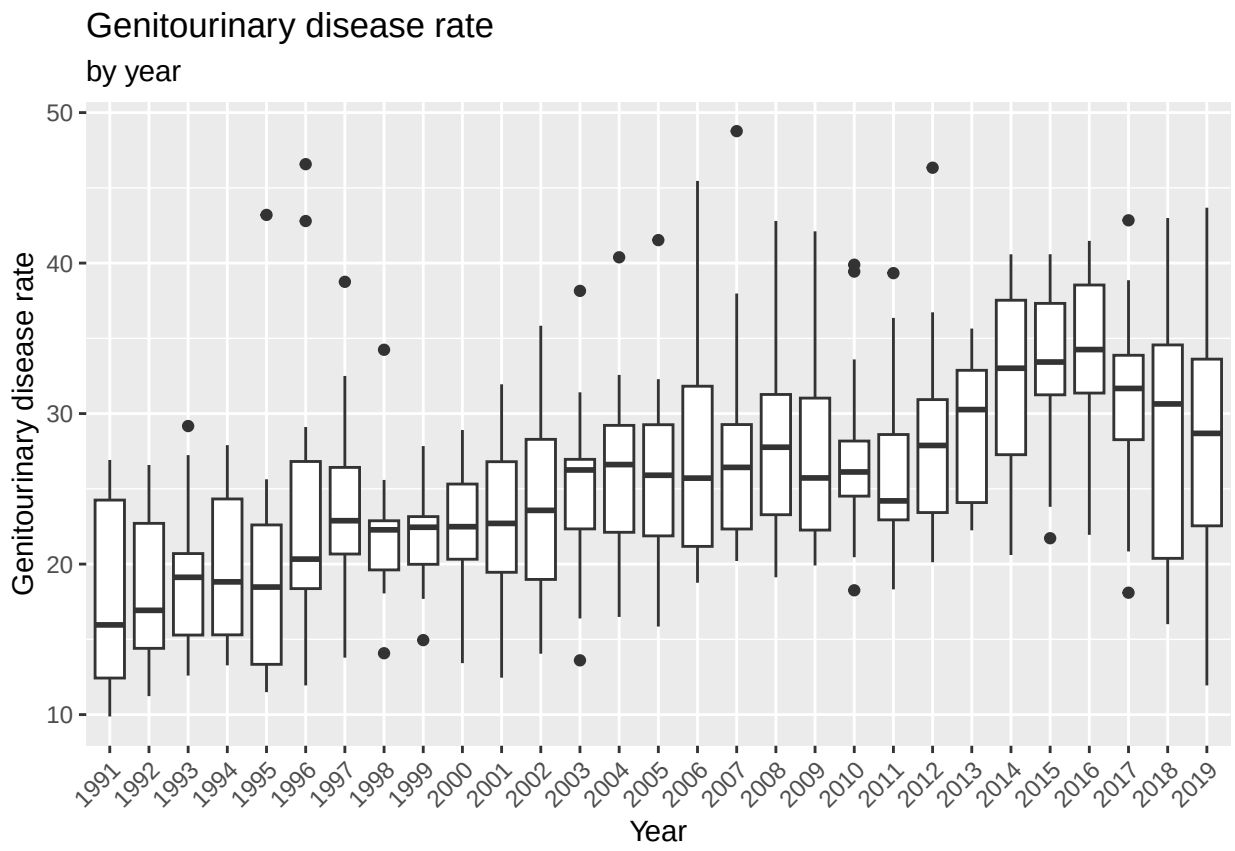

```
# By Region

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = Region, y = pc_genitourinary)) +
  geom_boxplot() +
  labs(title = "Genitourinary disease rate",
     x = "Region", y = "Genitourinary disease rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

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



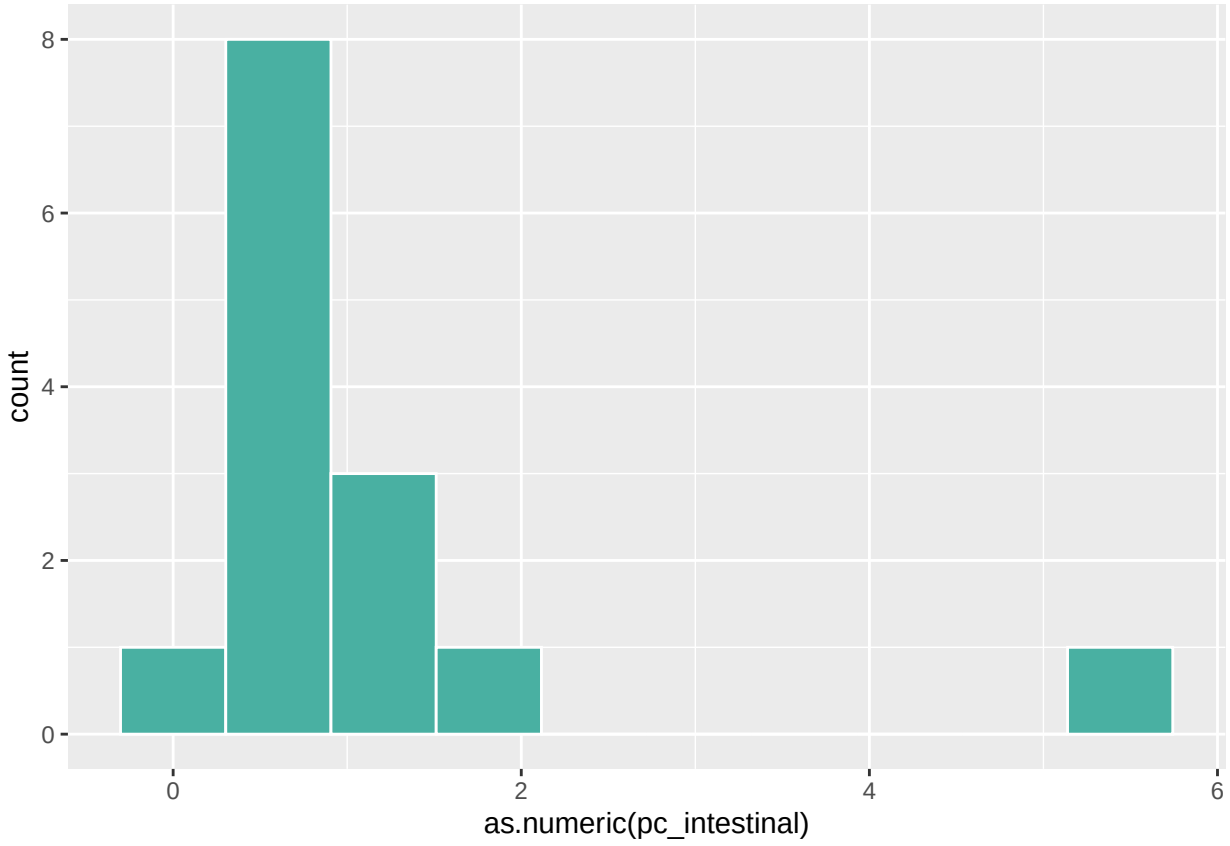

```
# By year

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = as.character(year), y = pc_genitourinary)) +
  geom_boxplot() +
  labs(title = "Genitourinary disease rate", subtitle = "by year",
     x = "Year", y = "Genitourinary disease rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

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




*Intestinal diseases*

```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pc_intestinal == as.numeric(pc_intestinal)) %>%
  select(pc_intestinal)

map_1@data$pc_intestinal <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(pc_intestinal))) +
    geom_histogram(bins=10, fill='#49b0a2', color='white')
```

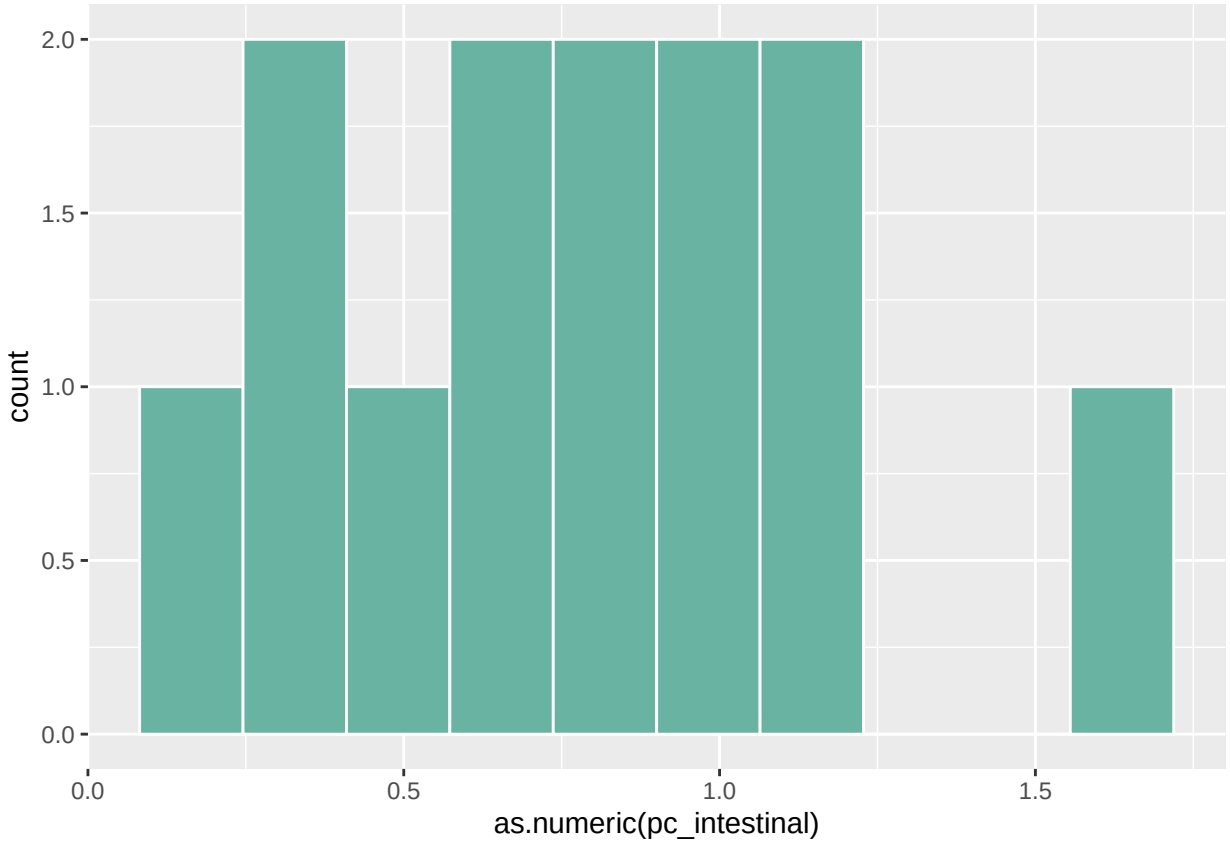



```
library(RColorBrewer)
my_colors <- brewer.pal(7, "Purples")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$pc_intestinal, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Intestinal diseases per capita, 2019")
```

## Intestinal diseases per capita, 2019

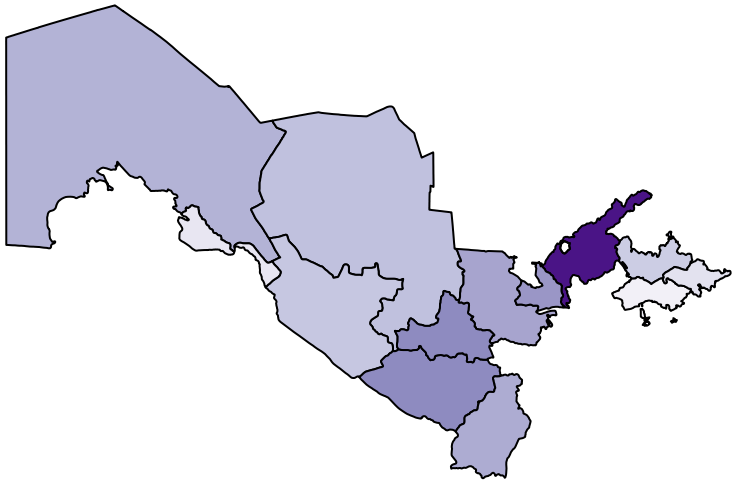
```
## excluding the outlier, Tashkent city

test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pc_intestinal == as.numeric(pc_intestinal)) %>%
  mutate(pc_intestinal = ifelse(Region == "Tashkent city", NA, pc_intestinal)) %>%
  select(pc_intestinal)

map_1@data$pc_intestinal <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(pc_intestinal))) +
    geom_histogram(bins=10, fill='#69b3a2', color='white')
```

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

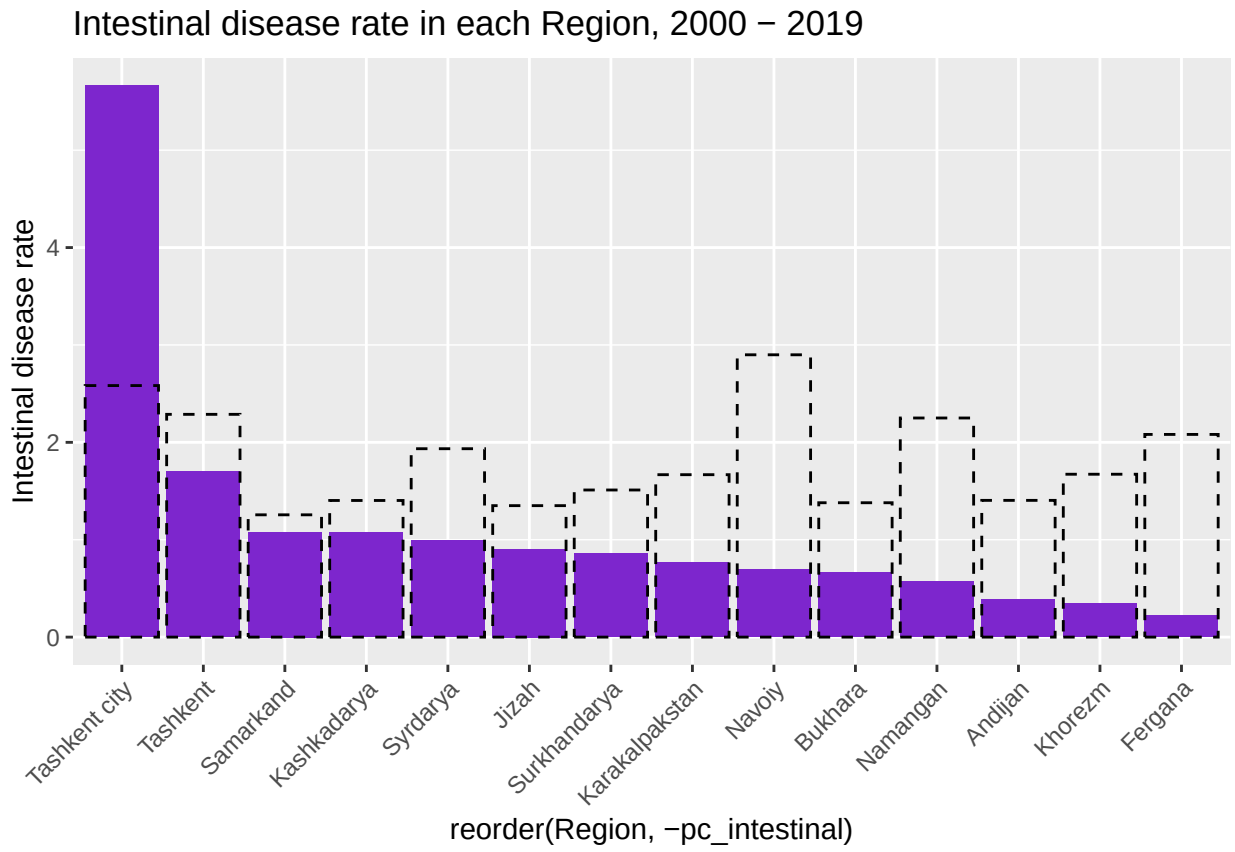



```
library(RColorBrewer)
my_colors <- brewer.pal(7, "Purples")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$pc_intestinal, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Intestinal Diseases per capita, 2019, (excl. Tashkent city)")
```

## Intestinal Diseases per capita, 2019, (excl. Tashkent city)

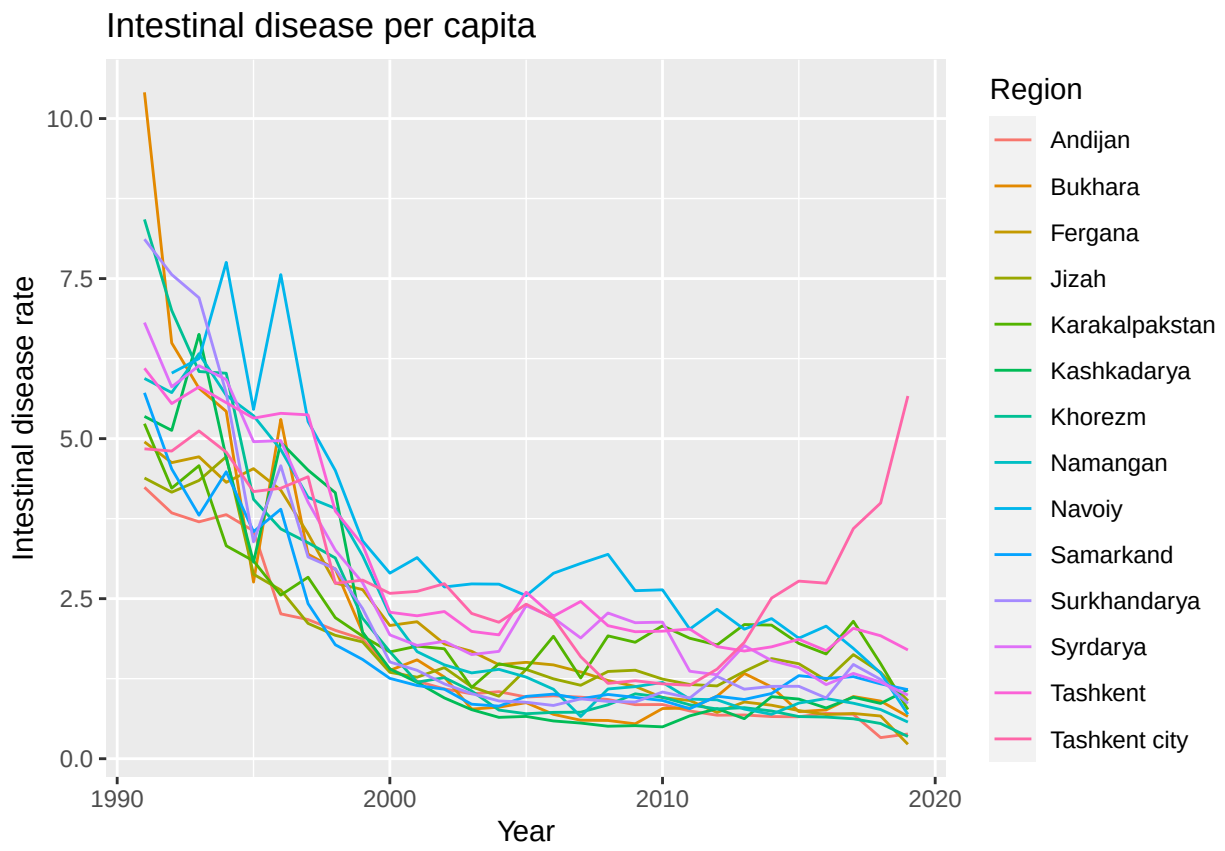


```
yrs <- uz_data %>%
  filter(pc_intestinal != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- 2019 #weird data


uz_data %>%
  na.omit() %>%
  group_by(Region) %>%
  ggplot(mapping = aes(x=reorder(Region, -pc_intestinal), y=pc_intestinal)) +
  geom_bar(data = filter(uz_data, year == 2019), stat = "identity", linetype = "blank", fill = "purple3
  geom_bar(data = filter(uz_data, year == 2000), stat = "identity", linetype = "dashed", fill = NA, col
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Intestinal disease rate in each Region, 2000 - 2019", y = "Intestinal disease rate")
```



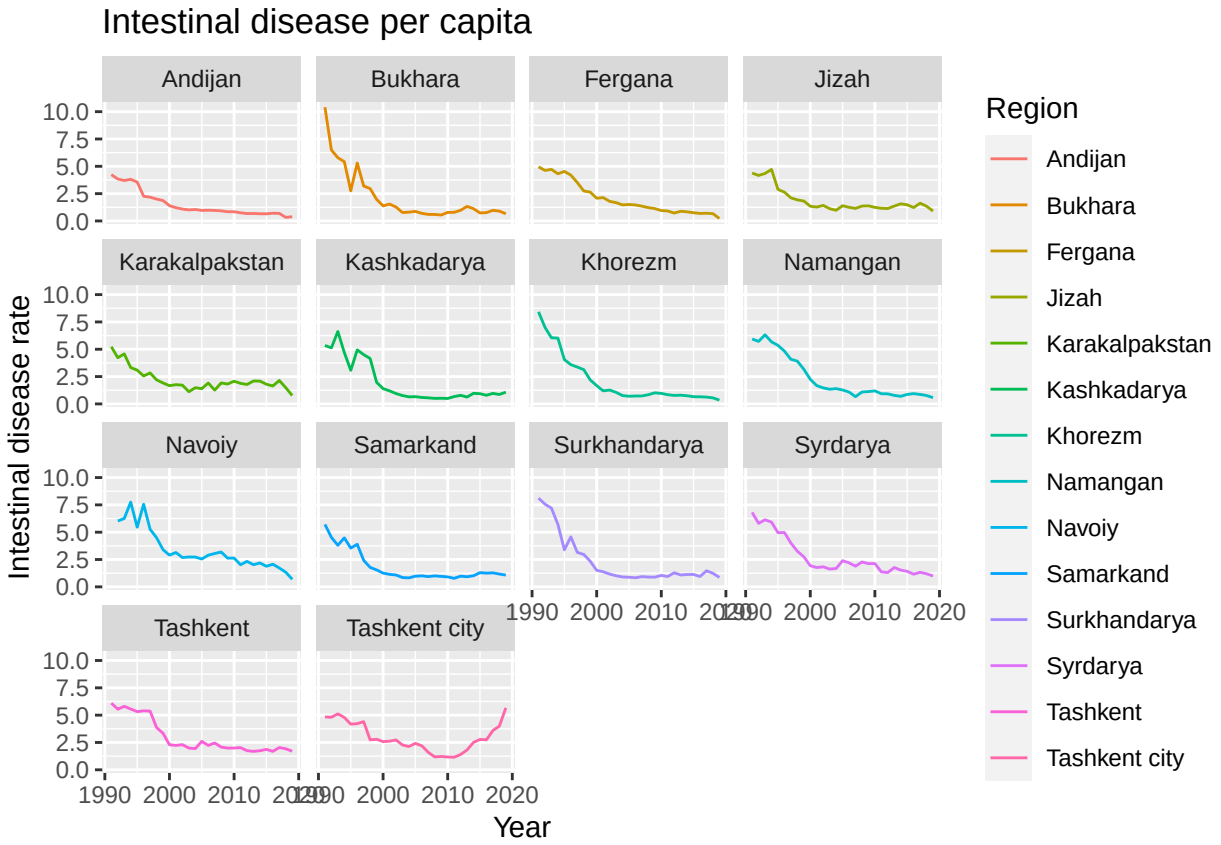

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_intestinal, color = Region)) +
  geom_line() +
  labs(title = "Intestinal disease per capita",
       x = "Year", y = "Intestinal disease rate")
```

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




```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_intestinal, color = Region)) +
  geom_line() +
  facet_wrap(Region ~ .) +
  labs(title = "Intestinal disease per capita",
       x = "Year", y = "Intestinal disease rate")
```

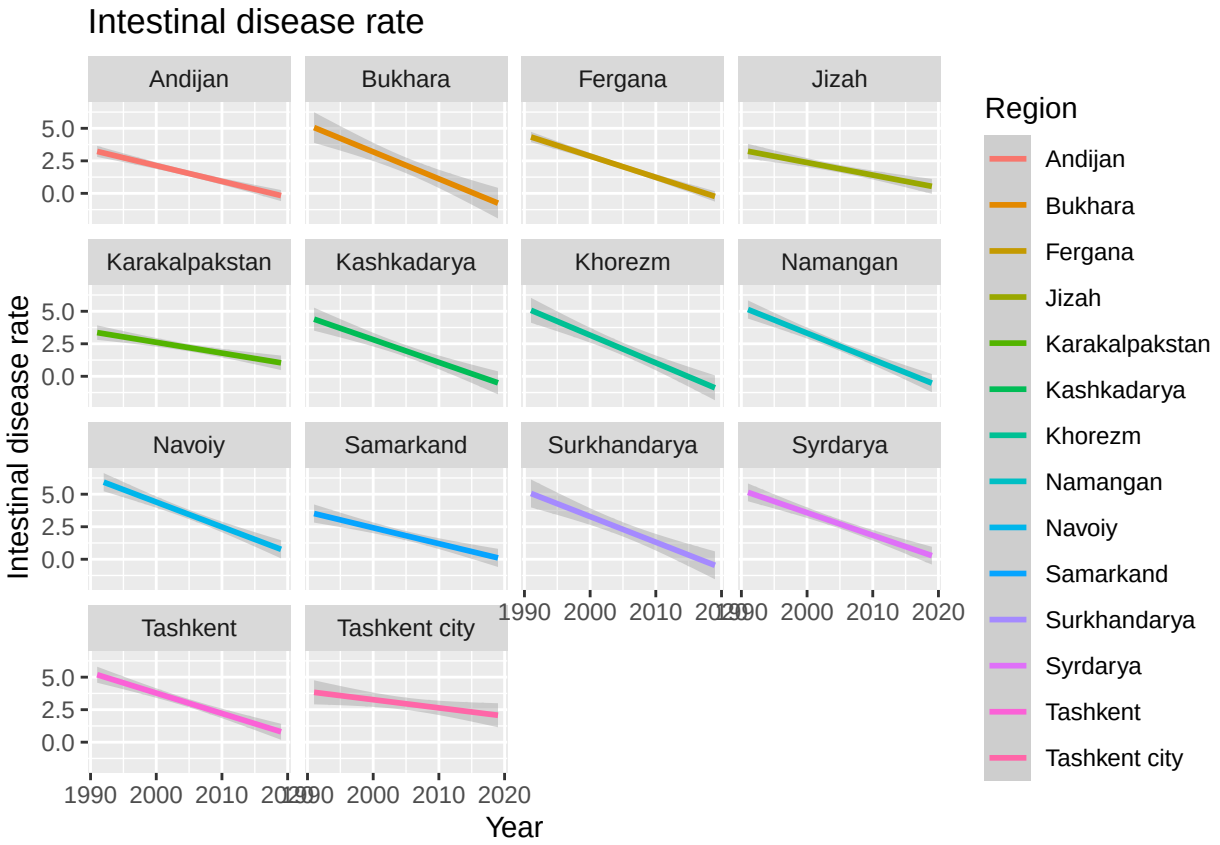
```
## Warning: Removed 1 row containing missing values (‘geom_line()‘).
```




```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_intestinal, color = Region)) +
  geom_smooth(method = "lm") +
  facet_wrap(Region ~ .) +
  labs(title = "Intestinal disease rate",
       x = "Year", y = "Intestinal disease rate")
```

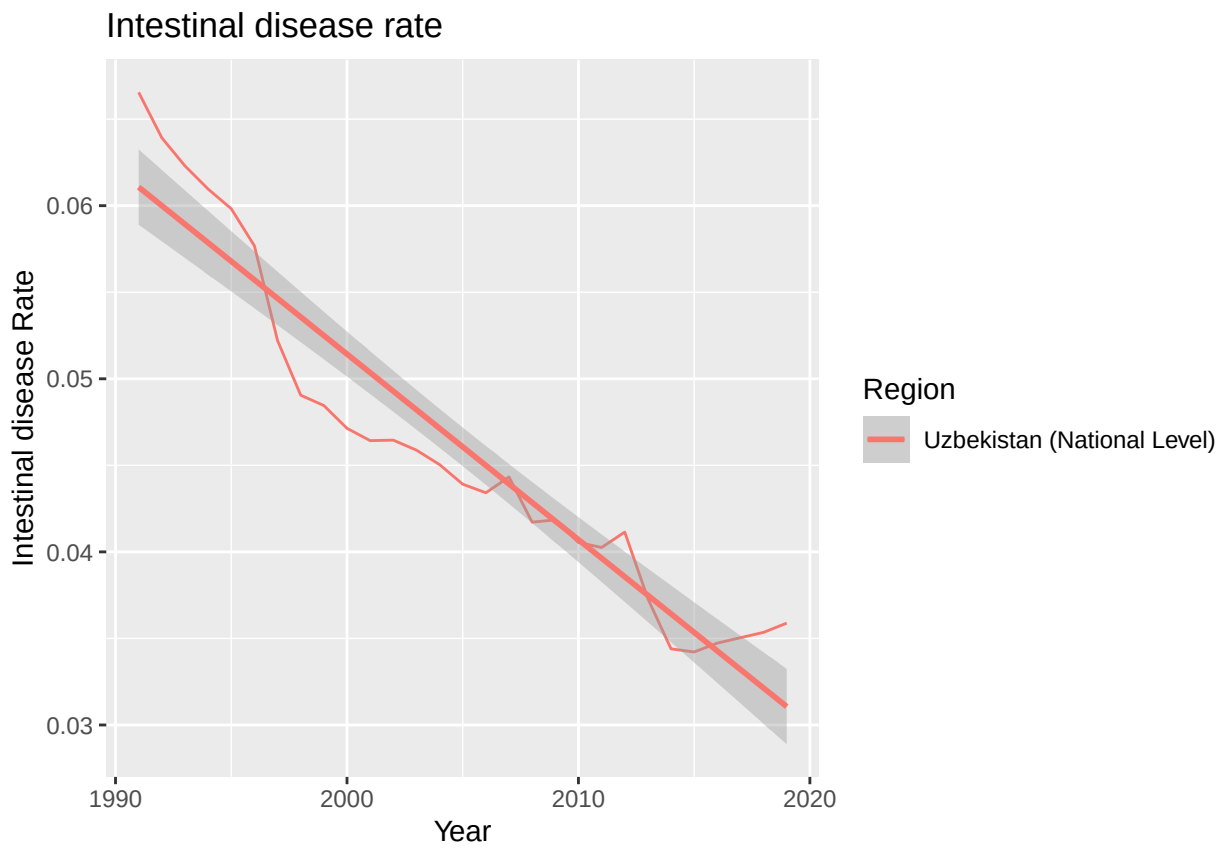
```
## `geom_smooth()` using formula = 'y ~ x'
```

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



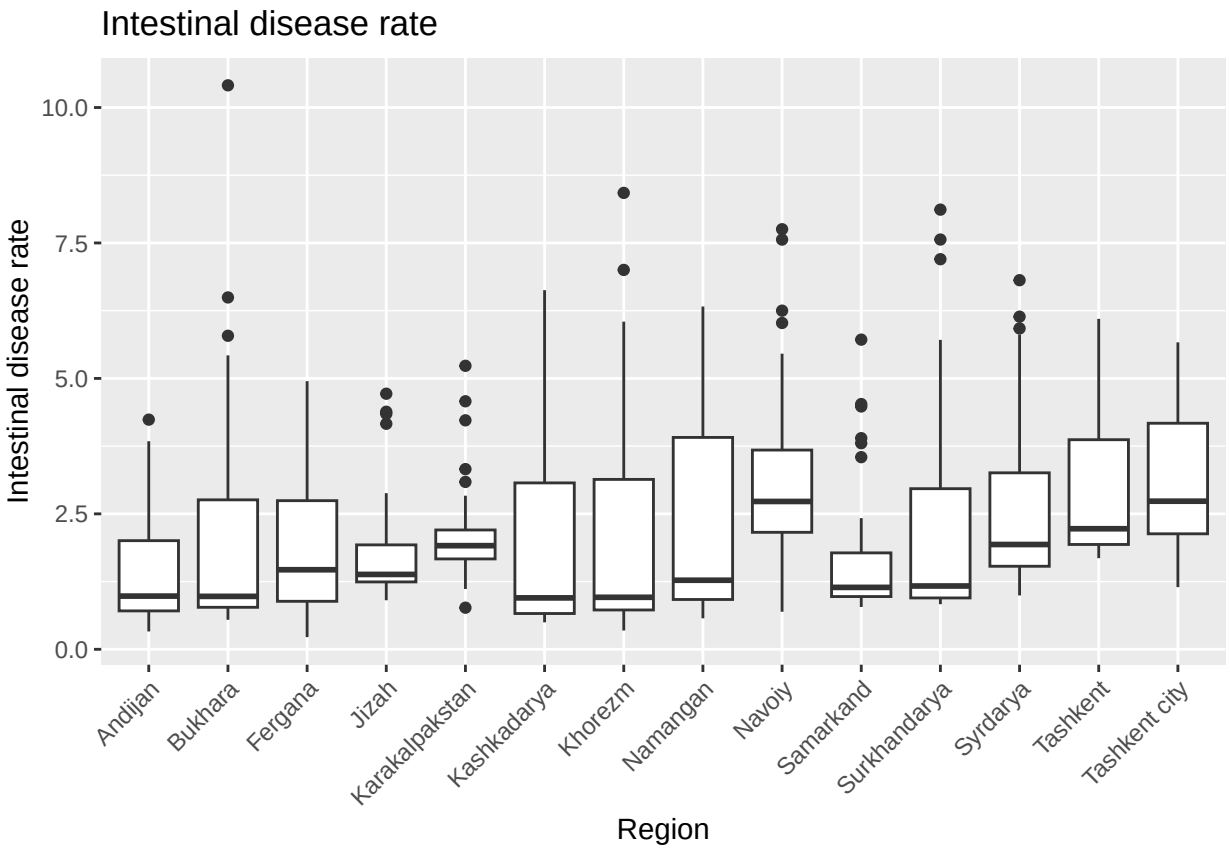

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_intestinal, color = Region)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(title = "Intestinal disease rate",
       x = "Year", y = "Intestinal disease Rate") +
  scale_fill_discrete(name = "Intestinal disease Rate", labels = c("", "B", "C"))
```

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



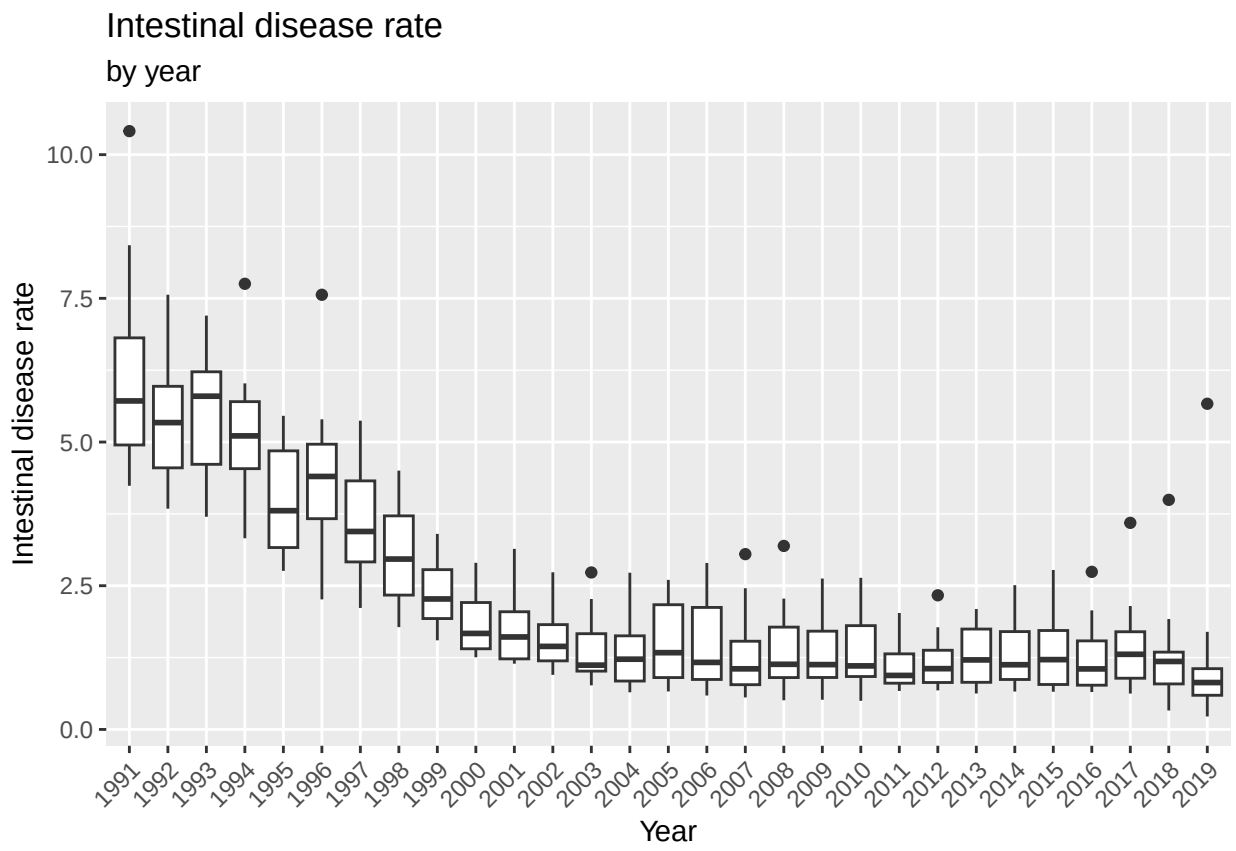

```
# By Region

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = Region, y = pc_intestinal)) +
  geom_boxplot() +
  labs(title = "Intestinal disease rate",
     x = "Region", y = "Intestinal disease rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

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



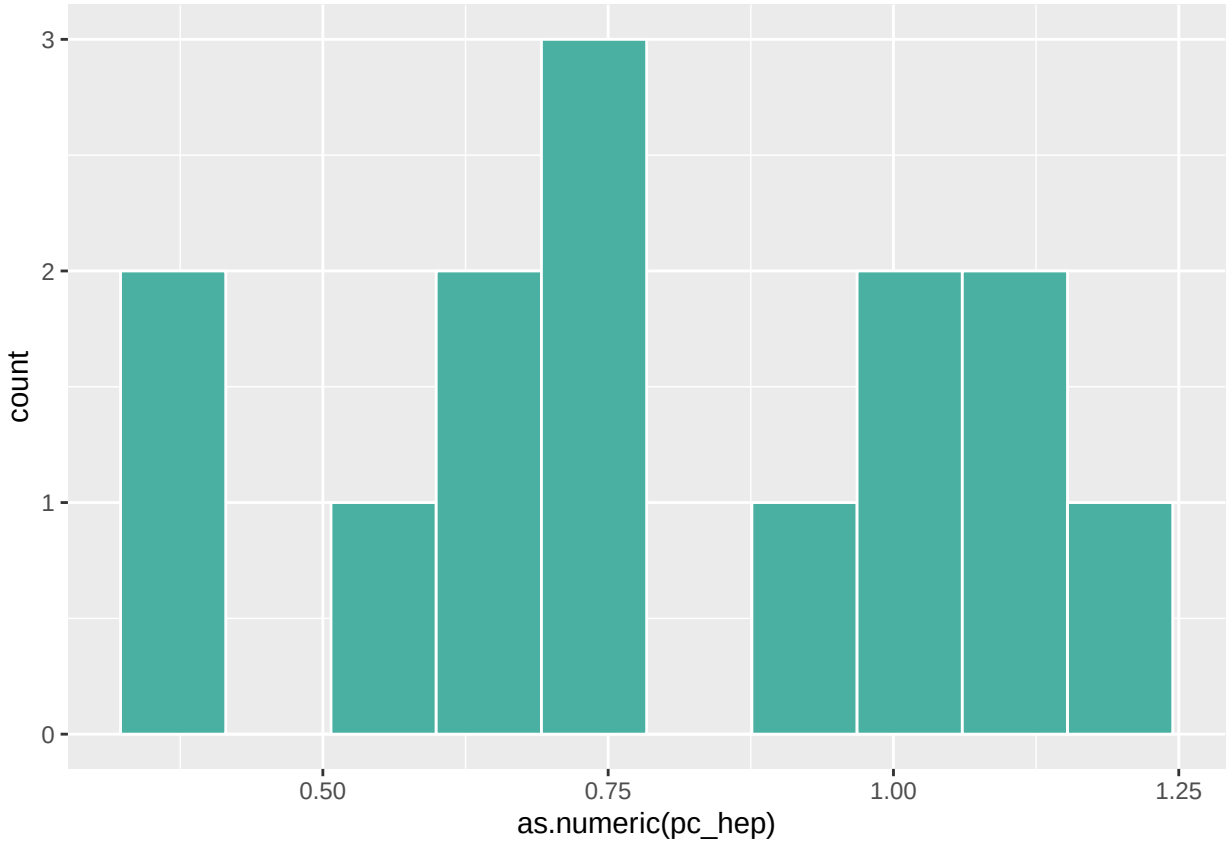

```
# By year

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = as.character(year), y = pc_intestinal)) +
  geom_boxplot() +
  labs(title = "Intestinal disease rate", subtitle = "by year",
     x = "Year", y = "Intestinal disease rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

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

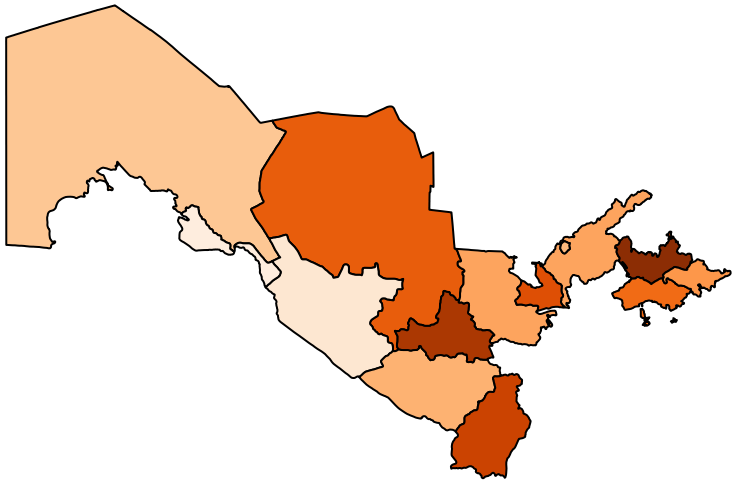



*Viral Hepatitis*

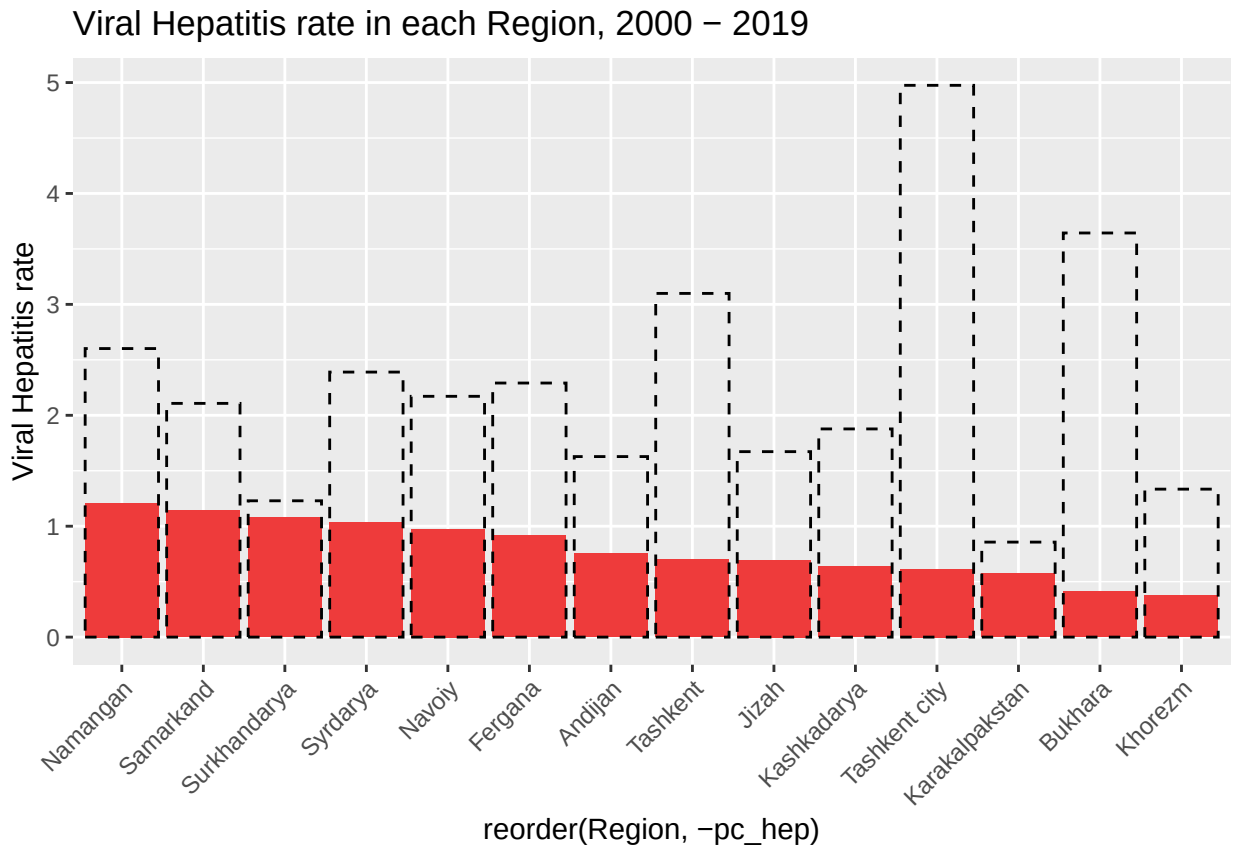
```
test <- data.frame(map_1@data$NAME_1)
test <- test %>%
  mutate(Region = map_1.data.NAME_1) %>%
  merge(filter(uz_data, year == 2019), by = "Region", all = TRUE, sort = FALSE) %>%
  mutate(pc_hep == as.numeric(pc_hep)) %>%
  select(pc_hep)

map_1@data$pc_hep <- as.numeric(unlist(test))

map_1@data %>%
  ggplot( aes(x=as.numeric(pc_hep))) +
    geom_histogram(bins=10, fill='#49b0a2', color='white')
```




```
library(RColorBrewer)
my_colors <- brewer.pal(7, "Oranges")
my_colors <- colorRampPalette(my_colors)(30)

class_of_country <- cut(map_1@data$pc_hep, 30)
my_colors <- my_colors[as.numeric(class_of_country)]

plot(map_1 , col=my_colors, main = "Viral Hepatitis per capita, 2019")
```

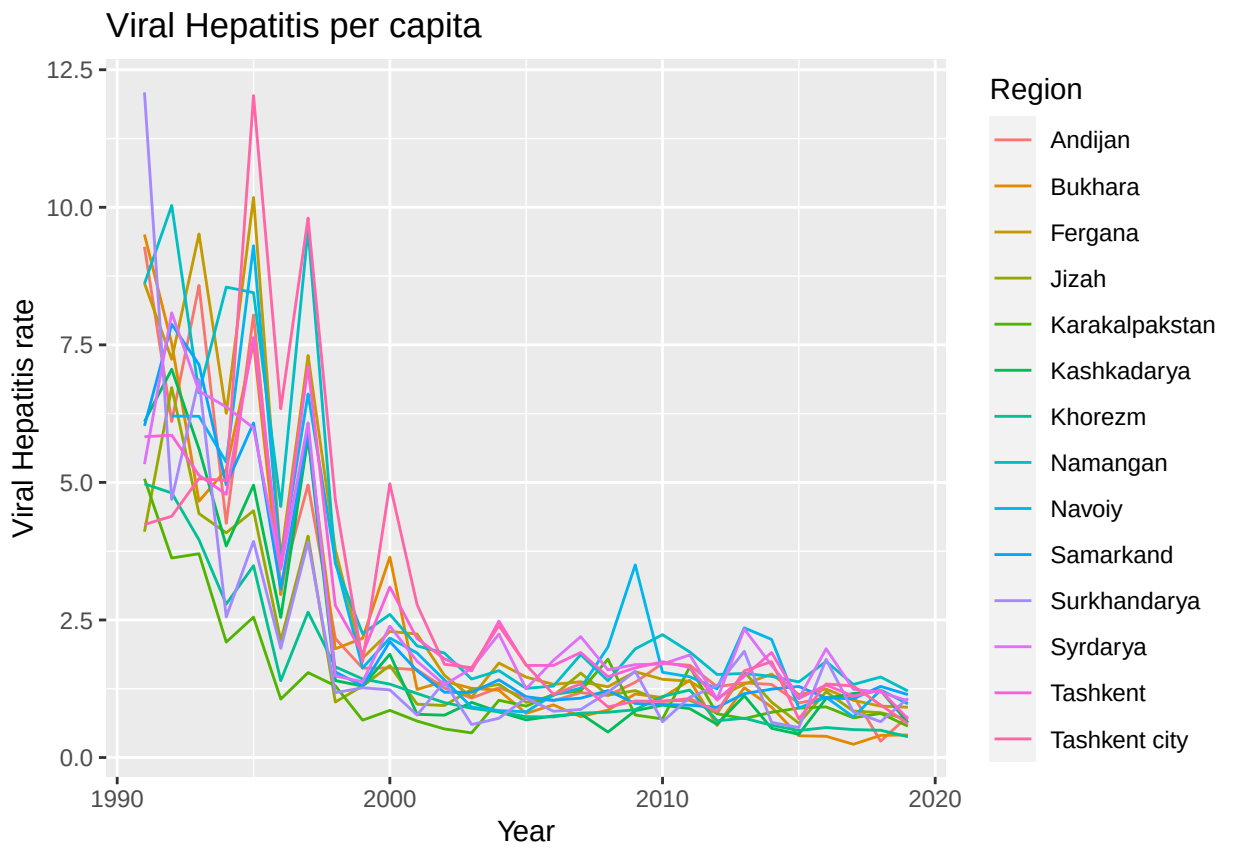



```
yrs <- uz_data %>%
  filter(pc_hep != 0) %>%
  select(year) %>%
  slice(1, n())
yr_start <- yrs$year[1]
yr_end <- 2019 #weird data


uz_data %>%
  na.omit() %>%
  group_by(Region) %>%
  ggplot(mapping = aes(x=reorder(Region, -pc_hep), y=pc_hep)) +
  geom_bar(data = filter(uz_data, year == 2019), stat = "identity", linetype = "blank", fill = "brown2"
  geom_bar(data = filter(uz_data, year == 2000), stat = "identity", linetype = "dashed", fill = NA, col
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Viral Hepatitis rate in each Region, 2000 - 2019", y = "Viral Hepatitis rate")
```



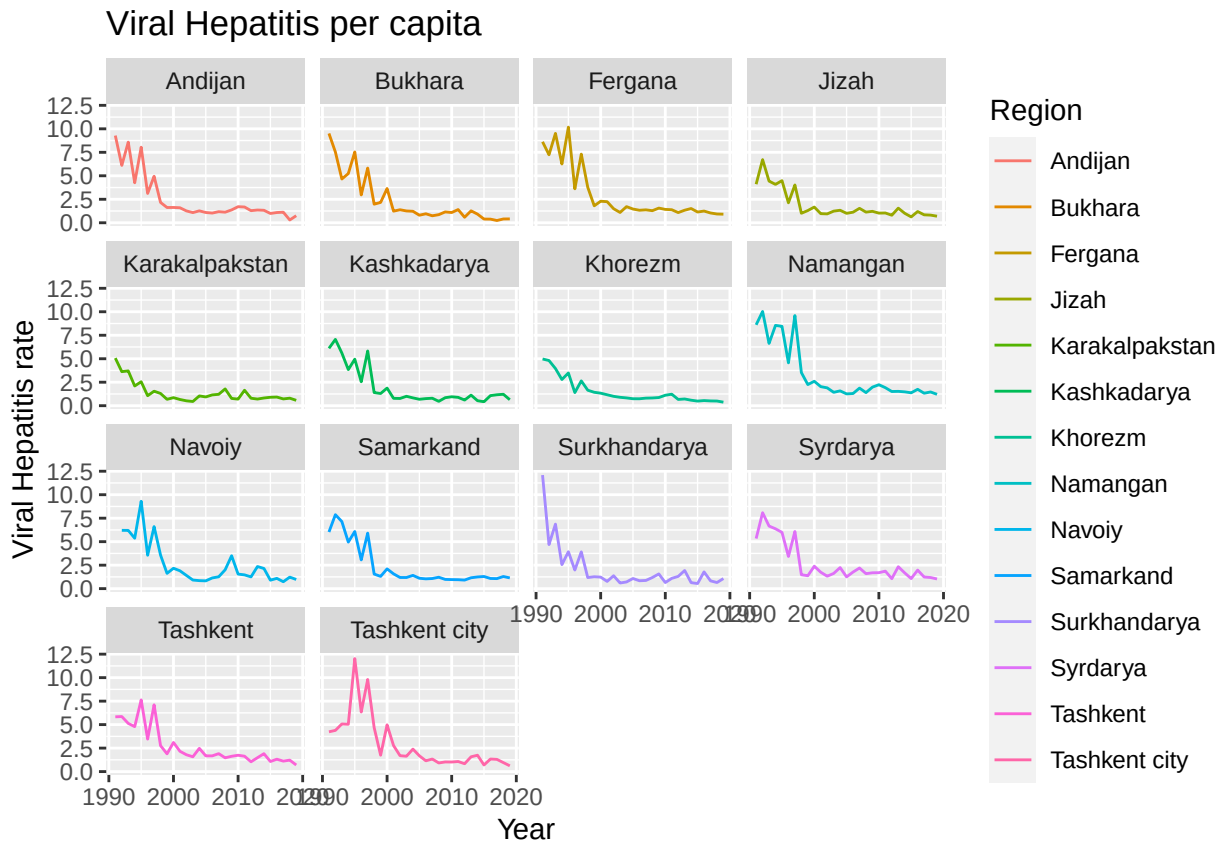

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_hep, color = Region)) +
  geom_line() +
  labs(title = "Viral Hepatitis per capita",
       x = "Year", y = "Viral Hepatitis rate")
```

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



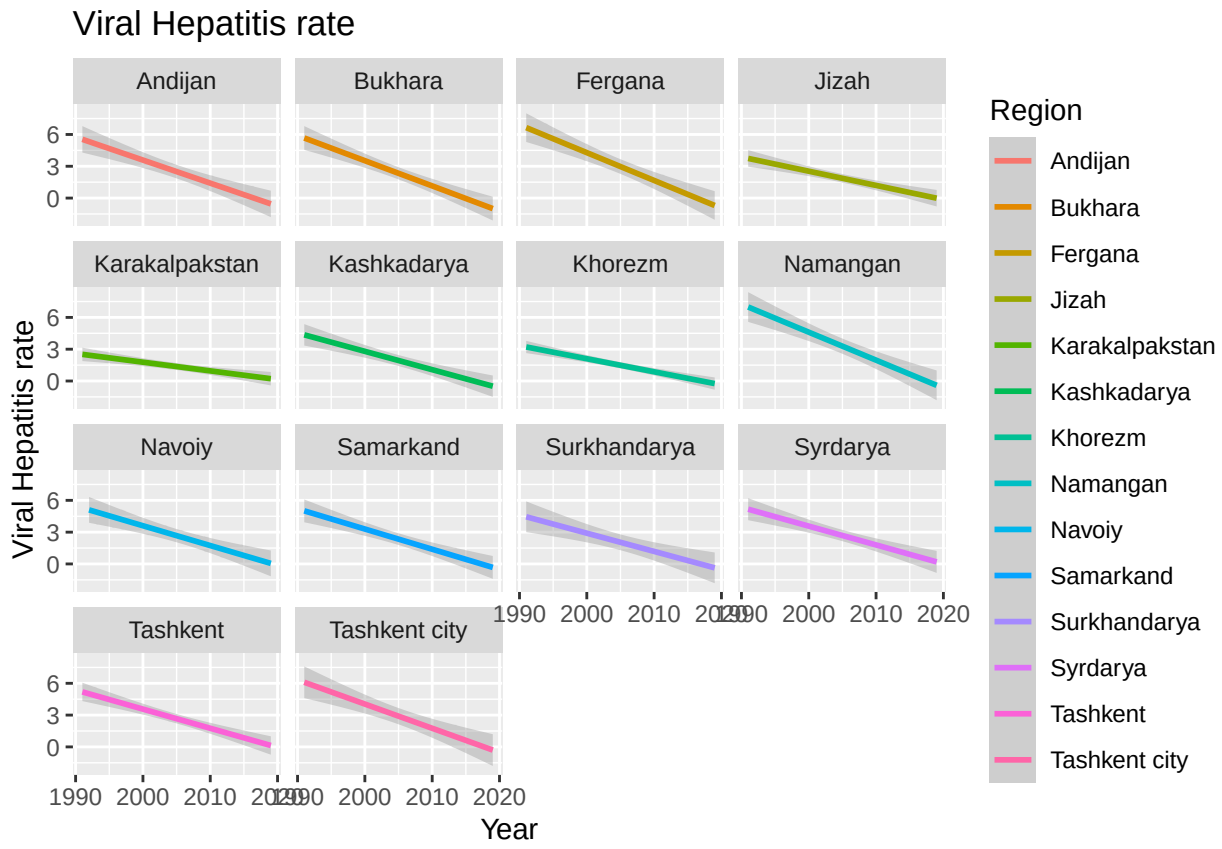

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_hep, color = Region)) +
  geom_line() +
  facet_wrap(Region ~ .) +
  labs(title = "Viral Hepatitis per capita",
       x = "Year", y = "Viral Hepatitis rate")
```

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



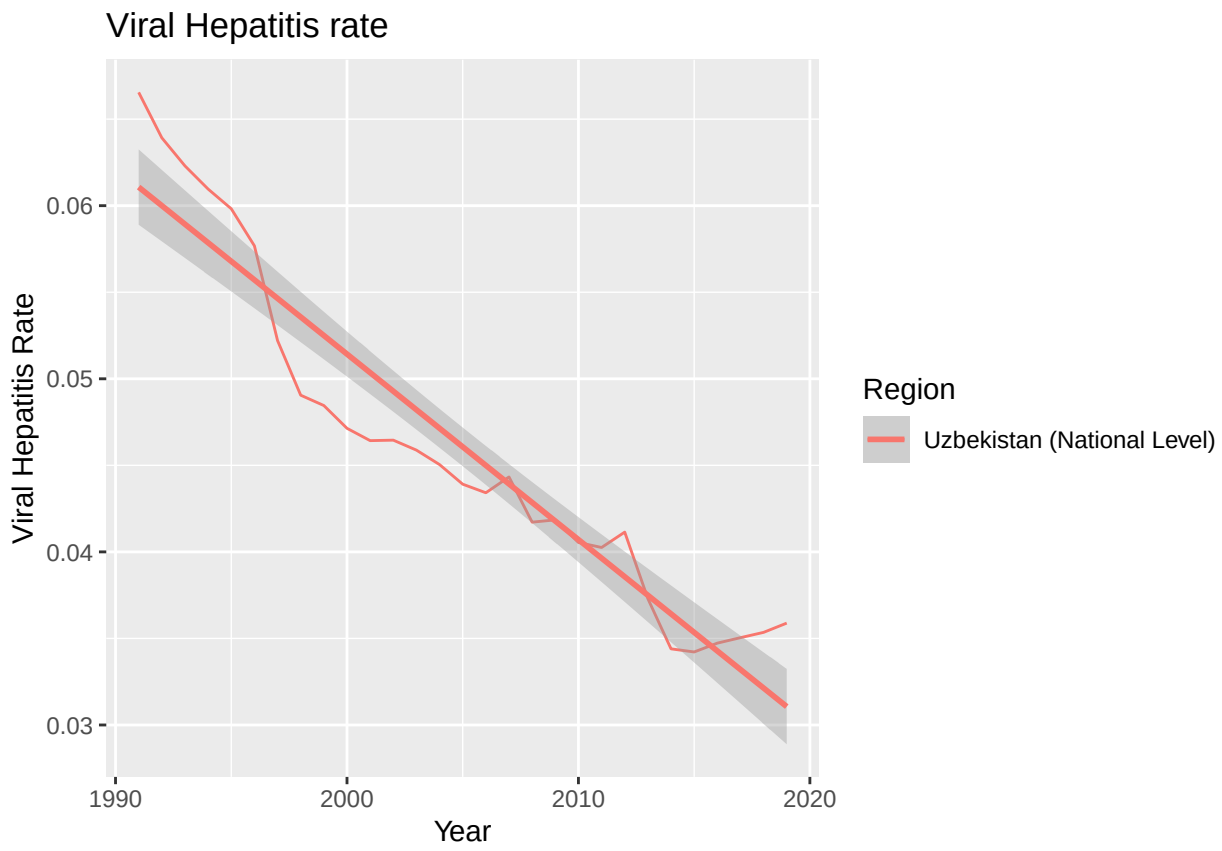

```
uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_hep, color = Region)) +
  geom_smooth(method = "lm") +
  facet_wrap(Region ~ .) +
  labs(title = "Viral Hepatitis rate",
       x = "Year", y = "Viral Hepatitis rate")
```

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

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



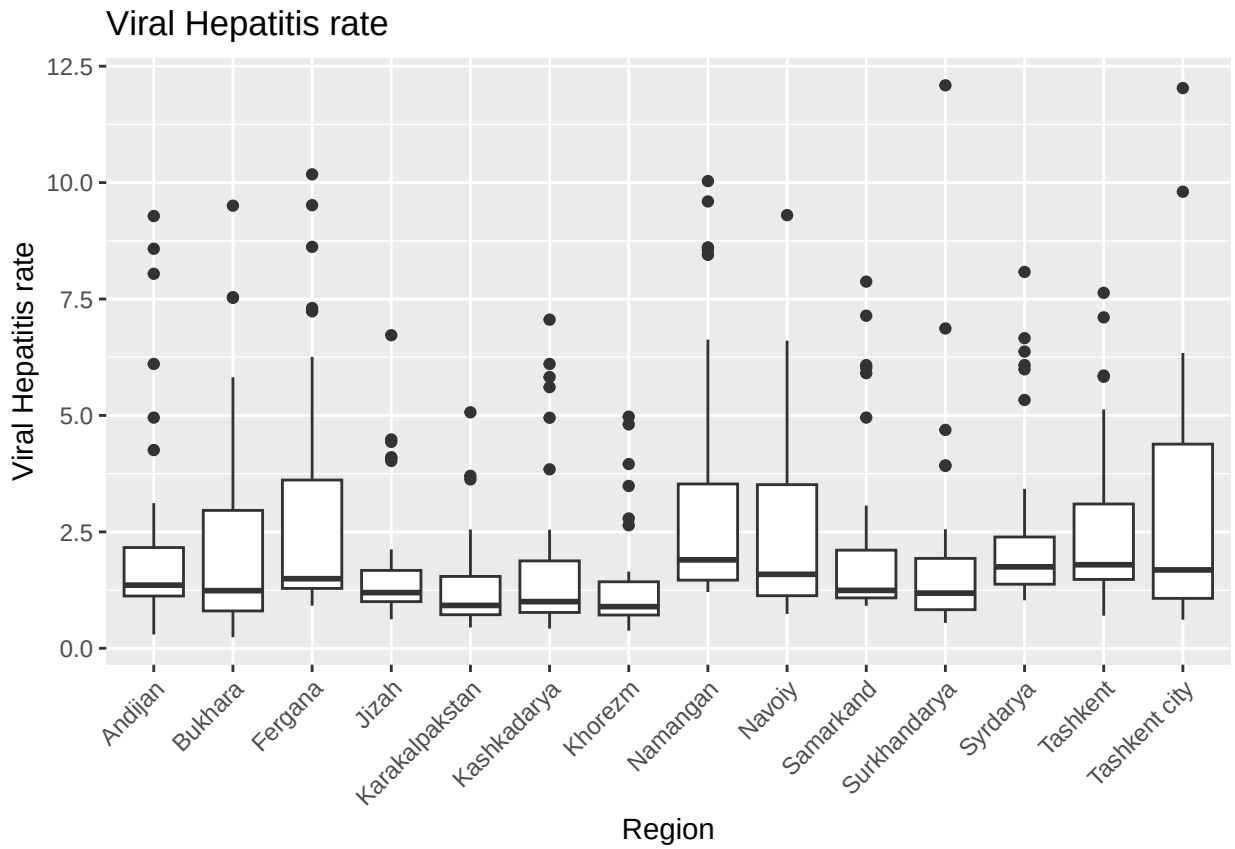

```
nat_uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = year, y = pc_hep, color = Region)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(title = "Viral Hepatitis rate",
       x = "Year", y = "Viral Hepatitis Rate") +
  scale_fill_discrete(name = "Viral Hepatitis Rate", labels = c("", "B", "C"))
```

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



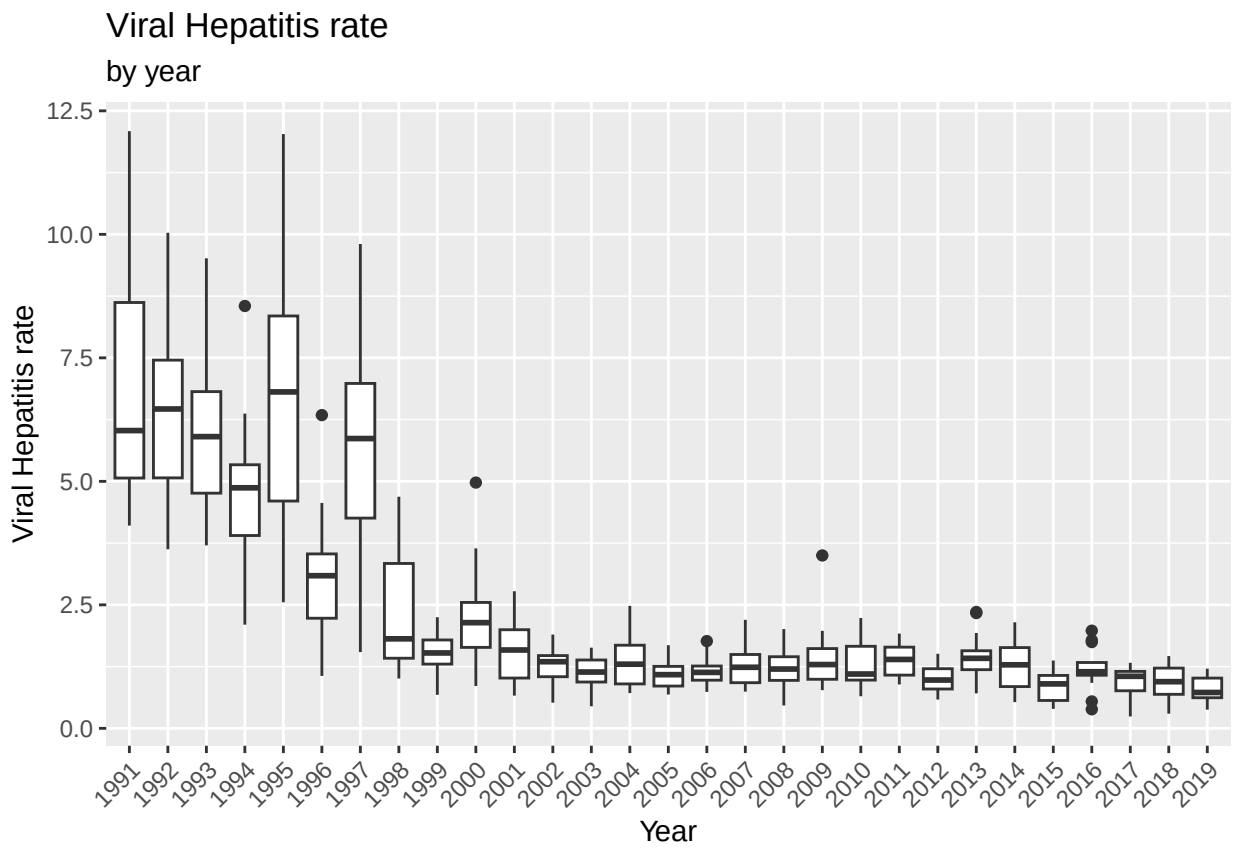

```
# By Region

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = Region, y = pc_hep)) +
  geom_boxplot() +
  labs(title = "Viral Hepatitis rate",
     x = "Region", y = "Viral Hepatitis rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

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




```
# By year

uz_data %>%
  filter(year >= yr_start, year <= yr_end) %>%
  ggplot(aes(x = as.character(year), y = pc_hep)) +
  geom_boxplot() +
  labs(title = "Viral Hepatitis rate", subtitle = "by year",
     x = "Year", y = "Viral Hepatitis rate") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

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




*OUR REGIONS OF INTEREST: BORDER WITH KAZAKHSTAN*

```
map_1@data  <- map_1@data %>%
  mutate(color = if_else(NAME_1 %in% c("Tashkent","Syrdarya","Jizah","Navoiy", "Karakalpakstan"), "ligh
plot(map_1, col = map_1@data$color, main = "Border regions with Kazakhstan")
```

## Border regions with Kazakhstan