MA20277 - Coursework 2

23720

Loading Packages:

```
library( dplyr )
library( ggplot2 )
library( patchwork )
library( tidyr )
library( tidytext )
library( wordcloud )
library( stringr )
library( widyr )
library( topicmodels )
library(RColorBrewer)
library(gstat)
library(sp)
library( sf )
library( mapview )
library( ggspatial )
library( prettymapr )
```

Question 1

The novel "Frankenstein" by Mary Shelley tells the story of a young scientist who creates a creature via an experiment and is subsequently horrified by what he has made. Perform an analysis that addresses the following questions:

```
Frankenstein_raw<- read.csv("Frankenstein.csv")
data("stop_words")
AFINN <- read.csv("AFINN Sentiment Lexicon.csv")
```

a) Which five words, apart from those on the stop list considered in the lectures, appear the most often, and what is their term frequency?

I will start by extracting the individual words, removing any underscores from the extracted words, and removing stop words.

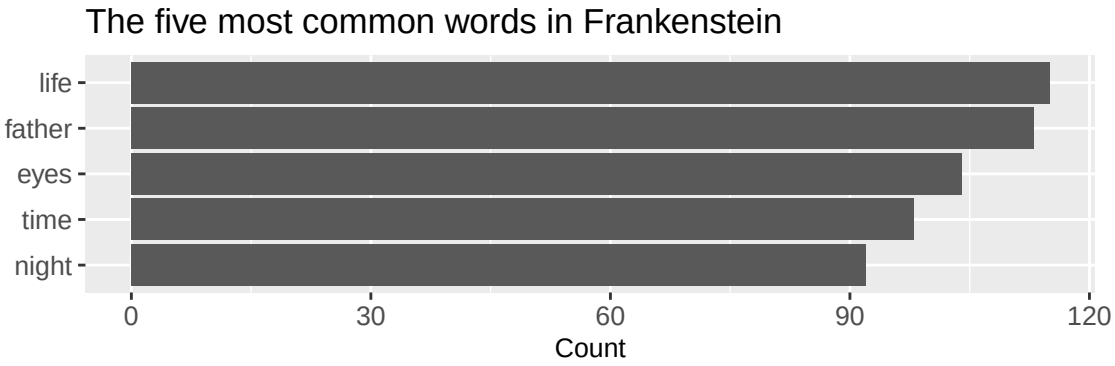
```
Frankenstein <- Frankenstein_raw %>%
  unnest_tokens( word, text ) %>%
  mutate( word = gsub( "\\_", "", word ) )%>%
  anti_join(stop_words)
```

Now counting the absolute values i.e how often they appear in the text; as well as calculating term frequency

```
Frankenstein_count <- Frankenstein%>%
  count(word, sort = TRUE)%>%
  mutate( 'term frequency' = n / sum(n), rank = row_number() )
```

Visualising the word count:

```
Frankenstein_count%>%
  slice_head(n=5)%>%
  mutate( word = reorder(word,n) ) %>%
  ggplot( aes( x=n, y=word) ) + geom_col() +
    labs( x="Count", y="",title = "The five most common words in Frankenstein")+
    theme( axis.text=element_text(size=10), axis.title=element_text(size=10) )
```



And the term frequency table:

```
Frankenstein_count%>%
  slice_head(n=5)%>%
  mutate( word = reorder(word,`term frequency`) )%>%
  print()
```

```
##     word   n term frequency rank
## 1   life 115    0.004211837    1
## 2 father 113    0.004138588    2
## 3   eyes 104    0.003808966    3
## 4   time  98    0.003589218    4
## 5  night  92    0.003369470    5
```

We see that the actual count of the top five words, reflect the proportion that they are in the text. There is no reshuffling in the order in which they appear. This is because calculating term frequency is more useful when comparing texts. When were only looking at one, it doesn't provide us with anymore information than the word count does.

b) Which three words are the most specific to Chapter 14?

We want to find the words that have the highest Term Frequency-Inverse Document Frequency (tf-idf) in Chapter 14 as these will be the most specific to that individual chapter.

First we split the book into its chapters:

```
Frankenstein_chapters<- Frankenstein_raw%>%
  mutate( chapter = cumsum( str_detect(text, regex("^chapter ",
                                                   ignore_case = TRUE) ) ) ) %>%
  unnest_tokens( word, text ) %>%
  mutate( word = gsub( "\\_", "", word ) )%>%
  anti_join(stop_words)
```

Now calculating the tf-idf:

```
Frankenstein_tf.idf <- Frankenstein_chapters%>%
  count(chapter, word, sort = TRUE)%>%
  bind_tf_idf( word, chapter, n ) %>%
  arrange( desc(tf_idf) ) %>%
  filter(chapter == 14)%>%
  slice_head(n=3)%>%
  print()
```

```
##   chapter  word  n         tf      idf     tf_idf
## 1      14 felix 18 0.02469136 1.427116 0.03523744
## 2      14  turk  9 0.01234568 2.525729 0.03118184
## 3      14 safie 14 0.01920439 1.609438 0.03090827
```

We see that "Felix", "Turk", and "Safie" are the three most spectific words to Chapter 14 as they have the highest tf-idf. Having a high tf-idf value means that these words appears a lot in Chapter 14 but not so much outside of Chapter 14. These are character names, so this makes sense, they were probably introduced in Chapter 14.

c) How does the emotional intent evolve throughout the book?

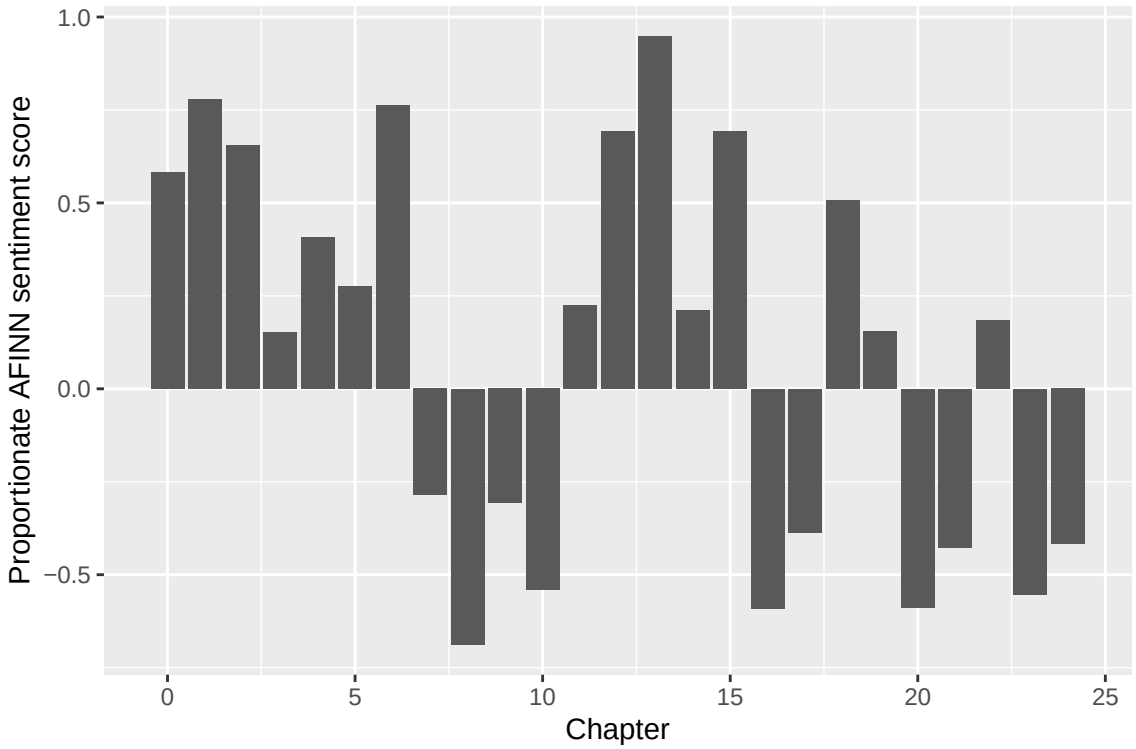
To determine the emotional intent throughout the book, we will conduct sentiment analysis, looking at the sentiment by chapter, then line-by-line:

Firstly, I want the AFINN sentiment score to be proportionate to the number of words in each chapter. This way, longer chapters like Chapter 24, are not disproportionately pulled in one direction. I've defined the table Frankenstein_AFINN as the number of AFINN words in each chapter, and then transformed it into a vector and called it AFFIN_count. This time we also want to include stop words as were looking at sentient, we want our analysis to really consider all words in the text. So let's re-define:

```
Frankenstein_chapters_sw<- Frankenstein_raw%>%
  mutate( chapter = cumsum( str_detect(text, regex("^chapter ",
                                                   ignore_case = TRUE) ) ) ) %>%
  unnest_tokens( word, text ) %>%
  mutate( word = gsub( "\\_", "", word ) )
```

```
Frankenstein_AFINN <- Frankenstein_chapters_sw%>%
  inner_join(AFINN)%>%
  count(chapter)
AFINN_count <- c(Frankenstein_AFINN$n)

Frankenstein_chapters_sw %>%
  inner_join( AFINN ) %>%
  group_by( chapter ) %>%
```

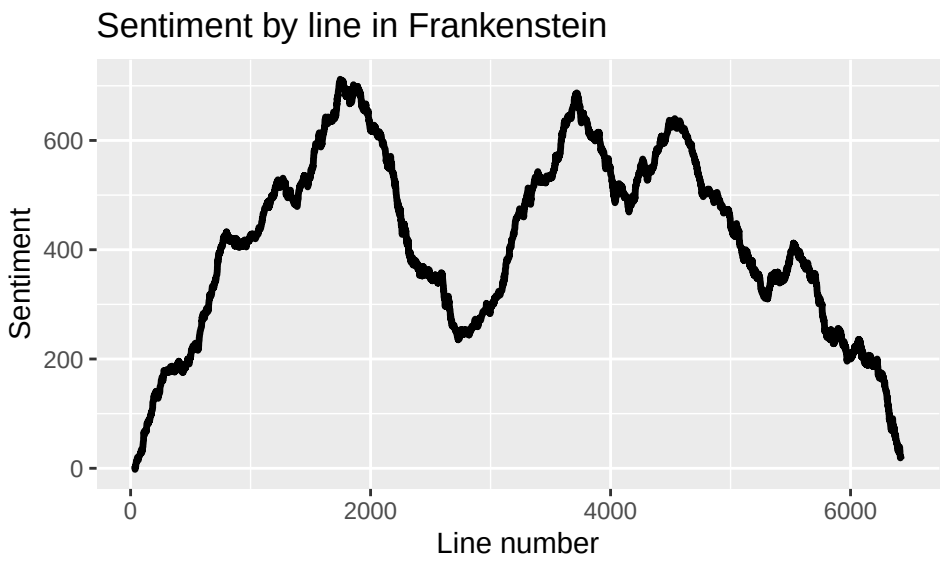
```
  summarise( sentiment = sum(value)) %>%
  mutate(sentiment = sentiment/AFINN_count)%>%
  ggplot( aes( x=chapter, y=sentiment ) ) +geom_col() +
  labs( x="Chapter", y="Proportionate AFINN sentiment score" )
```



We see that the novel starts quite positively, especially in the first three chapters, we then get a dip at Chapter 7 to 10 followed by a peak of sentiment at Chapter 13. However after this the emotion of the novel almost oscillates, but having lower lows than high highs. It ends on a negative note.

Let's confirm this by looking at each line of the book and calculating its sentiment:

```
Frankenstein_line <- Frankenstein_raw%>%
  mutate( line = row_number() ) %>%
  unnest_tokens( word, text ) %>%
  mutate( word = gsub( "_", "", word ) )

Frankenstein_line%>%
  inner_join(AFINN)%>%
  filter( word != "miss" ) %>%
  mutate( sentiment = cumsum( value ) ) %>%
  ggplot( aes( x=line, y=sentiment ) ) + geom_line( size=1.2 ) +
  labs( x="Line number", y="Sentiment", title = "Sentiment by line in Frankenstein" )
```



Here we can make a similar conclusion: we have have high sentiment scores until about a third of a way through the novel (equivalent to Chapters 7-10) then we get a dip in emotion. We see an increase again afterwards, but we see this general decrease towards the end of the novel. But we can also see what I described as oscillations in the last 1500 lines as the decline isn't consistent, going through these ups and downs while overall decreasing.

Question 2

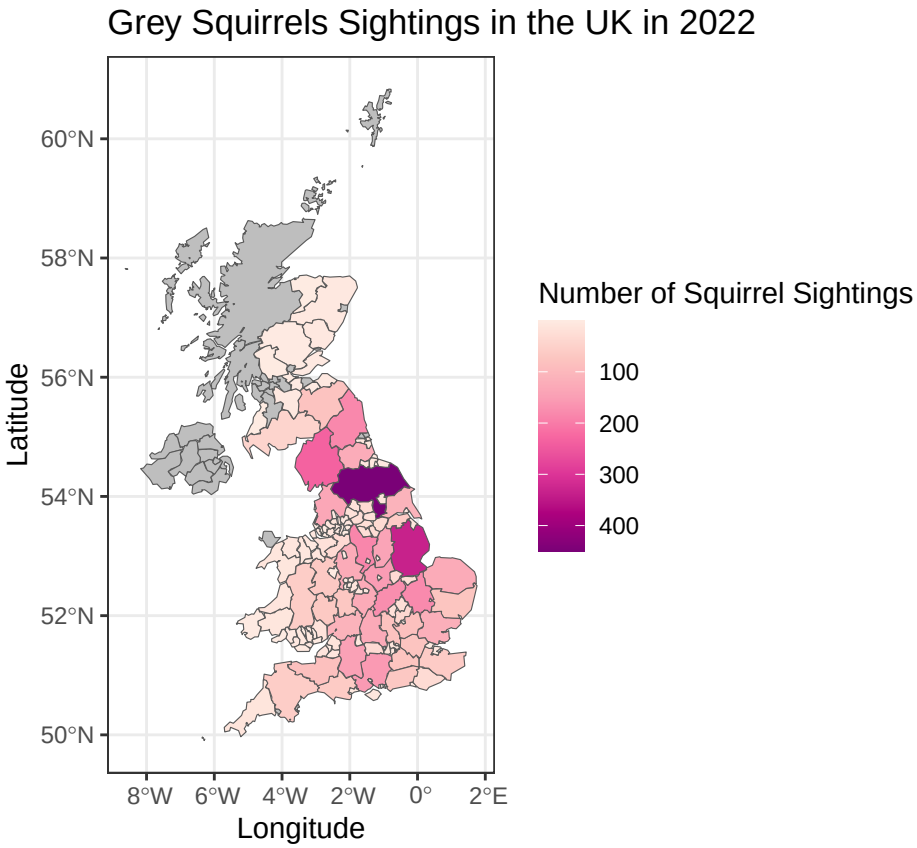
The grey squirrel is classified as an invasive species in the UK, and it has displaced the native red squirrel across large parts of the UK. A wildlife conservation charity has collected data on reported sightings of grey squirrels for 2020-2022. The charity assured us that the data is representative of the spatial distribution of squirrels across the UK for all years. They ask you to use the data to investigate the following aspects:

```
Squirrels<- read.csv("GreySquirrels.csv")

UK_admin<- read_sf("UK Shapefile", layer = "UK_admin")
UK_admin_simple <- st_simplify( UK_admin, dTolerance = 2000, preserveTopology=TRUE )

UK <- read_sf("UK Shapefile", layer = "UK")
UK_simple <- st_simplify( UK, dTolerance = 2000, preserveTopology=TRUE )
```

a) What can we say about the spatial distribution of grey squirrels across Great Britain in 2022?

```
Squirrels_2022 <-Squirrels%>%
  filter(Year == 2022)

Squirrel_count_2022<- Squirrels_2022%>%
  group_by(County)%>%
  count(County)%>%
  rename("Frequency2022" = n)

UK_squirrels<- full_join(x = UK_admin_simple, y = Squirrel_count_2022,
                         by =c("NAME_2"= "County"))
```

```
ggplot( data=UK_squirrels, aes(fill = Frequency2022) ) + theme_bw() + geom_sf() +
  scale_fill_distiller(palette = "RdPu", trans = "reverse", na.value = "grey")+
  labs(x= "Longitude", y = "Latitude", fill = "Number of Squirrel Sightings",
       title = "Grey Squirrels Sightings in the UK in 2022")
```



First of all, we can see that there are some areas of the country where we have really low number of grey squirrel sightings like in the Scottish highlands, this is due to Red Squirrels being more commonly found in these regions. We also do not have any data for the most northern parts of Scotland as Squirrels do not tend to live on these cold coasts. We also have missing data in big cities like Newcastle and Edinburgh, this just may be due to lack of habitat (its more built up). On the the other hand, we see the largest number of sightings in North Yorkshire and Lincolnshire. Another thing to note is the low number of sightings on the Isle of Wight, despite Hampshire having a good amount of Grey Squirrels, counting 158 in 2022. As it's separated from the main island, Grey squirrels have not been able to really infiltrate the Isle, there only being 10 total seen in 2022, it is more common to see a red squirrel here.

b) Are there any areas of Great Britain that saw a notable change in the number of grey squirrels when we compare the data for 2020 and 2022?

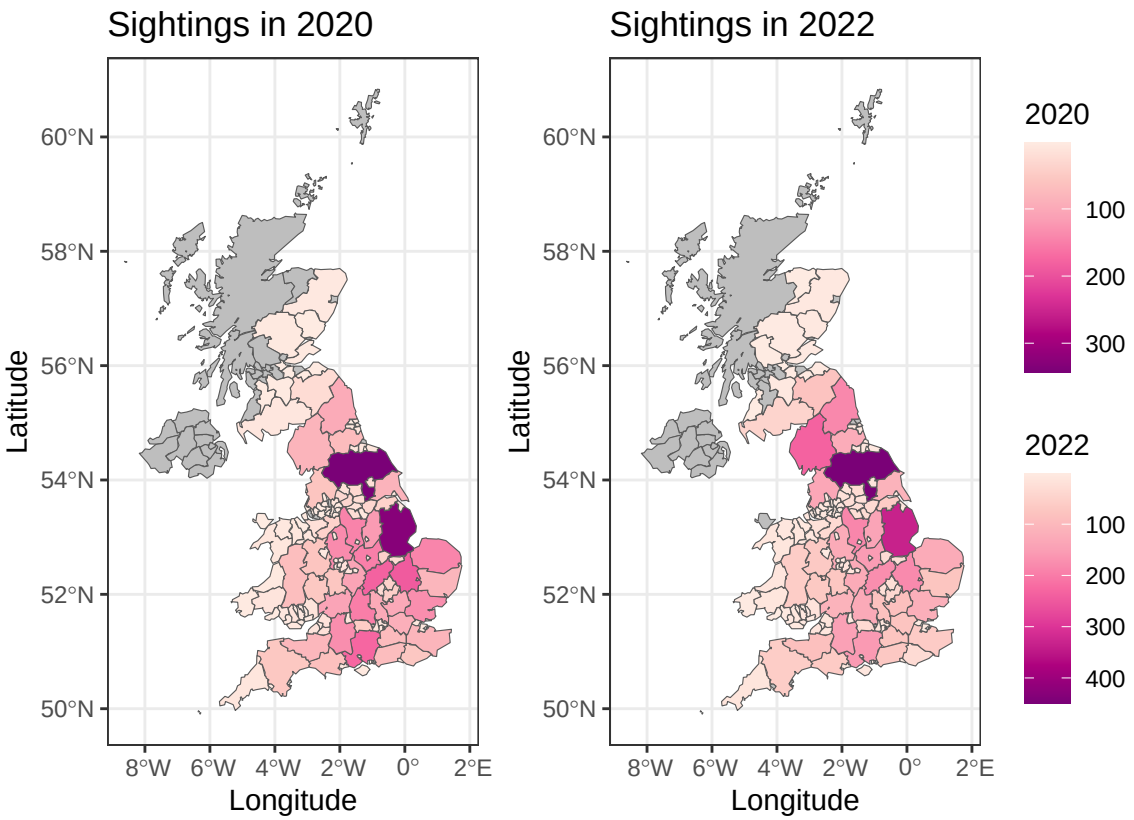
Lets create two separate plots for squirrel sightings in both 2020 and 2022. We already have our map for 2022, so sightings in 2020:

```
Squirrels_2020<- Squirrels%>%
  filter(Year == 2020)
```

```
Squirrel_count_2020<- Squirrels_2020%>%
  group_by(County)%>%
  count(County)%>%
  rename("Frequency2020" = n)

UK_squirrels_2020<- full_join(x = UK_admin_simple, y = Squirrel_count_2020,
                              by =c("NAME_2"= "County"))
```

Now plotting them next to each other:

```
Sightings_2020 <- ggplot( data=UK_squirrels_2020, aes(fill = Frequency2020) ) +
  theme_bw() + geom_sf() + scale_fill_distiller(palette = "RdPu", trans = "reverse",
                                                na.value = "grey")+
  labs(x= "Longitude", y = "Latitude", fill = "2020", title = "Sightings in 2020 ")
Sightings_2022 <- ggplot( data=UK_squirrels, aes(fill = Frequency2022) ) +
  theme_bw() + geom_sf() + scale_fill_distiller(palette = "RdPu", trans = "reverse",
                                                na.value = "grey")+
  labs(x= "Longitude", y = "Latitude", fill = "2022", title = "Sightings in 2022")

Sightings_2020+ Sightings_2022+ plot_layout( guides = "collect", )
```



The hotspots of Grey Squirrels have been unchanged, we still see the highest numbers in North Yorkshire and Lincolnshire. But we do see counties like Cumbria and Northumberland in the northern parts of England have had an increase in the number of Grey Squirrels seen over the two years. This suggests that the Grey

Squirrel invasion is making its way further north. It's possible that there will be a further increase in Grey Squirrels in Scottish counties, and consequently a smaller Red Squirrel population.

Question 3

The Utopian company Amaurot Cookies changed their cookie recipes at the beginning of 2021 due to customer reviews they received. The company would like to understand whether the new recipes have improved customer satisfaction. They thus extracted reviews that contained the word "cookie" for the years 2020-2022, and indicated whether reviews refer to their products or their competitors'. The company would like you to perform a data analysis that uses a sentiment lexicon to address the following two aspects:

```
CookieReviews_raw<- read.csv("CookieReviews.csv")
Bing<- read.csv("Bing Sentiment Lexicon.csv")
```

a) Are their new cookie recipes more positively received than their previous ones by customers?

Classifying the recipies into the old one and the new one. Also numbering the reviews

```
CookieReviews <- CookieReviews_raw%>%
  mutate(Recipe = case_when(Year == "2020" ~ "Old",
                            Year == "2021"~"New",
                            Year == "2022"~ "New"))

CookieReviews <- mutate(CookieReviews, Index=1:nrow(CookieReviews) )
```

I'm going to use the Bing Sentiment lexicon which I've already loaded. I'd like to exclude "fudge" from the list. Its in the name of the recipe but the Bing Lexicon gives it a negative label. This is going to distort our sentiment proportion.

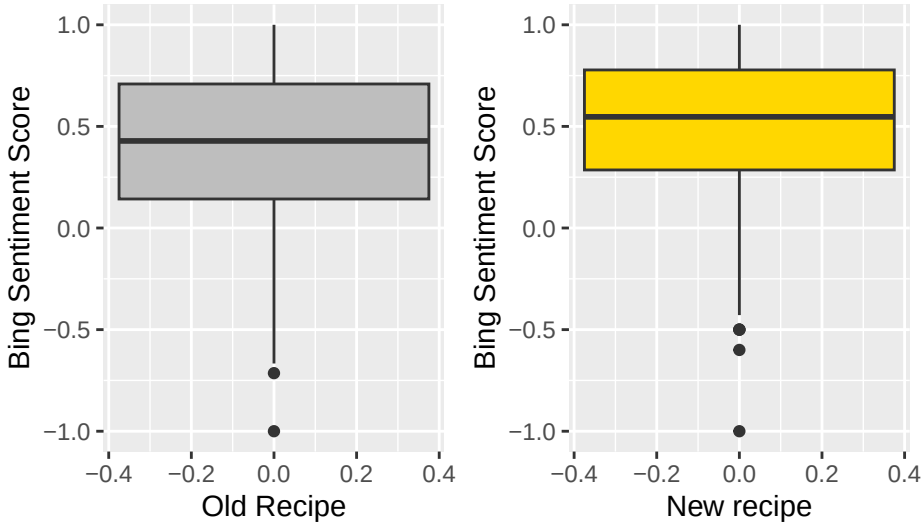
```
Intra_reviews <- CookieReviews%>%
  filter(Company == "Amaurot Cookies")%>%
  select("Index", "Recipe", "Text")%>%
  unnest_tokens( word, Text ) %>%
  mutate( word = gsub( "_", "", word ) ) %>%
  inner_join( Bing )%>%
  mutate(sentiment= case_when(sentiment == "positive"~as.numeric(1),
                              sentiment == "negative"~ as.numeric(-1)))%>%
  filter(word!= "fudge")
```

Now because I'm calculating the proportion of the sentiment score in each review, I need to find the number of suitable words in each review so which we can then calculate the sentiment.

```
Bing_count_old<-Intra_reviews%>%
  filter(Recipe == "Old")%>%
  count(Index)
Bing_count_old <- c(Bing_count_old$n)

Bing_count_new <- Intra_reviews%>%
  filter(Recipe == "New")%>%
  count(Index)
Bing_count_new <- c(Bing_count_new$n)
```

Now plotting:

```
Old<- Intra_reviews%>%
  filter(Recipe == "Old")%>%
  group_by(Index)%>%
  summarise(sentiment = sum(sentiment))%>%
  mutate(sentiment_prop = sentiment/Bing_count_old)%>%
  ggplot(aes(y = sentiment_prop)) + geom_boxplot(fill = "grey") +
    labs(x = "Old Recipe", y = "Bing Sentiment Score")

New<- Intra_reviews%>%
  filter(Recipe == "New")%>%
  group_by(Index)%>%
  summarise(sentiment = sum(sentiment))%>%
  mutate(sentiment_prop = sentiment/Bing_count_new)%>%
  ggplot(aes(y = sentiment_prop)) + geom_boxplot(fill = "gold") +
    labs(x = "New recipe", y = "Bing Sentiment Score")
Old+New
```



We see that the newer recipe is more positively recieved by customers than the older recipe. For the newer recipe we have a higher proportion of words in reviews have a positive sentiment. Meaning on average, customers are more postive about the newer recipe. Changing the recipe has been a success!

b) How different are their reviews for 2021-2022 compared to their competitors' in terms of the frequency of words usually attributed with a positive/negative sentiment?

Now we are comparing between Amaurot's new recipe and their competitors.

```
Inter_Reviews<- CookieReviews%>%
  filter(Year != "2020")%>%
  select("Index", "Company", "Text")%>%
  unnest_tokens( word, Text ) %>%
  mutate( word = gsub( "_", "", word ) ) %>%
```

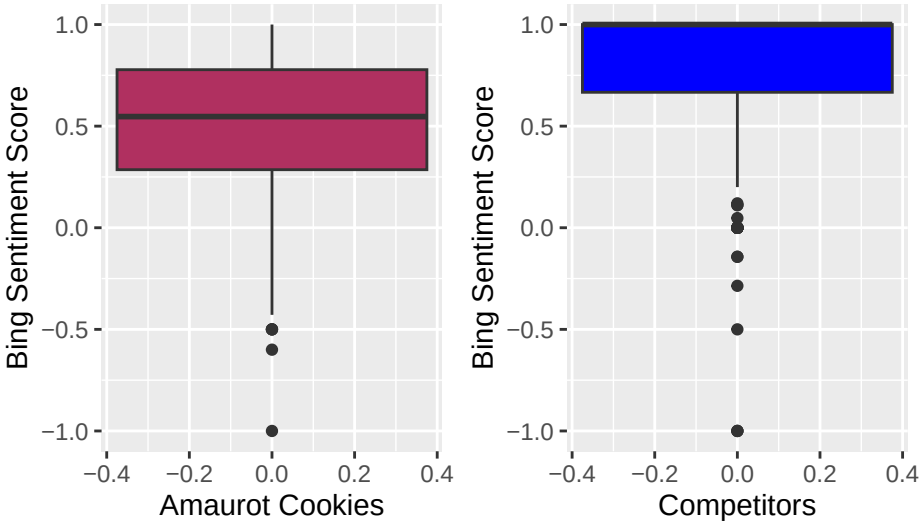
```
  inner_join( Bing )%>%
  mutate(sentiment= case_when(sentiment == "positive"~as.numeric(1),
                              sentiment == "negative"~ as.numeric(-1)))%>%
  filter(word!= "fudge")
```

Once again, lets calculate the number of suitable words per review:

```
Bing_count_Ama<-Inter_Reviews%>%
  filter(Company == "Amaurot Cookies")%>%
  count(Index)
Bing_count_Ama <- c(Bing_count_Ama$n)

Bing_count_Comp <- Inter_Reviews%>%
  filter(Company == "Competitors")%>%
  count(Index)
Bing_count_Comp <- c(Bing_count_Comp$n)
```

And finally plotting both the Sentiment scores for both the Amaurot Cookies' new recipe and its competitors:

```
Amaurot_Reviews <- Inter_Reviews%>%
  filter(Company == "Amaurot Cookies")%>%
  group_by(Index)%>%
  summarise(sentiment = sum(sentiment))%>%
  mutate(sentiment_prop = sentiment/Bing_count_Ama)%>%
  ggplot(aes(y = sentiment_prop)) + geom_boxplot(fill = "maroon") +
  labs(x = "Amaurot Cookies", y = "Bing Sentiment Score")

Comp_reviews<- Inter_Reviews%>%
  filter(Company == "Competitors")%>%
  group_by(Index)%>%
  summarise(sentiment = sum(sentiment))%>%
  mutate(sentiment_prop = sentiment/Bing_count_Comp)%>%
  ggplot(aes(y = sentiment_prop)) + geom_boxplot(fill = "blue") +
  labs(x = "Competitors", y = "Bing Sentiment Score")
Amaurot_Reviews +Comp_reviews
```



We see that reviews for competitors products are overwhelmingly more positive than for Amaurot Cookies. Competitor reviews rarely have sentiment scores below 0, meaning they rarely contain words with negative sentiments. They in fact have an approximate mean of 1, which means the average review of a competitors cookie, has completely positive language.

Question 4

The local authorities in the Utopian capital city Amaurot have seen an alarming increase in the number of people being diagnosed with kidney damage. Diagnostic tests revealed that many patients had high lead concentration levels in their blood. The authorities fear that contamination in the tap water may be the source for these increased levels. They thus measured the level of lead concentration in the tap water of each patient who reported with abdominal pain (another symptom associated with too high lead concentration levels) and ran diagnostics to determine the cause for the patient's pain.

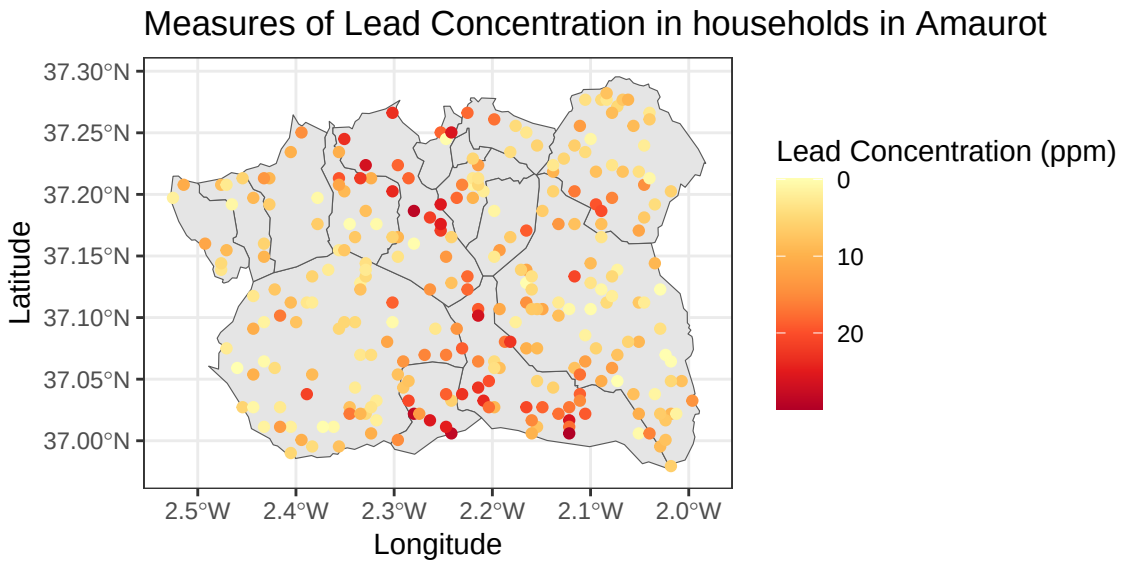
The local authorities have now approached you to help them tackle the health emergency. They provided you with the patient data, and a shapefile and grid for Amaurot. To hide Amaurot's location, constants have been added to the the latitude and longitude coordinates, but the shapes they define are correct. The local authorities ask you to perform the following analysis:

```
Patients<- read.csv("Amaurot Patients.csv")
Grid <- read.csv("AmaurotGrid.csv" )
Amaurot<- read_sf("AmaurotShapefile")
```

a) Explore the spatial distribution (including the spatial dependence) of the measured lead levels in the city's drinking water.

First lets see where the measurements of lead concentration were taken on a mp of Amaurot, and if there's any pattern.

```
LeadsConc<- ggplot(data = Amaurot) + theme_bw() + geom_sf()+
  geom_point(data = Patients,aes(x = Lon, y = Lat, colour = Lead)) +
  scale_color_distiller( palette="YlOrRd", trans="reverse" )+
  labs(x = "Longitude", y = "Latitude", colour = "Lead Concentration (ppm)",
       title = "Measures of Lead Concentration in households in Amaurot")
LeadsConc
```



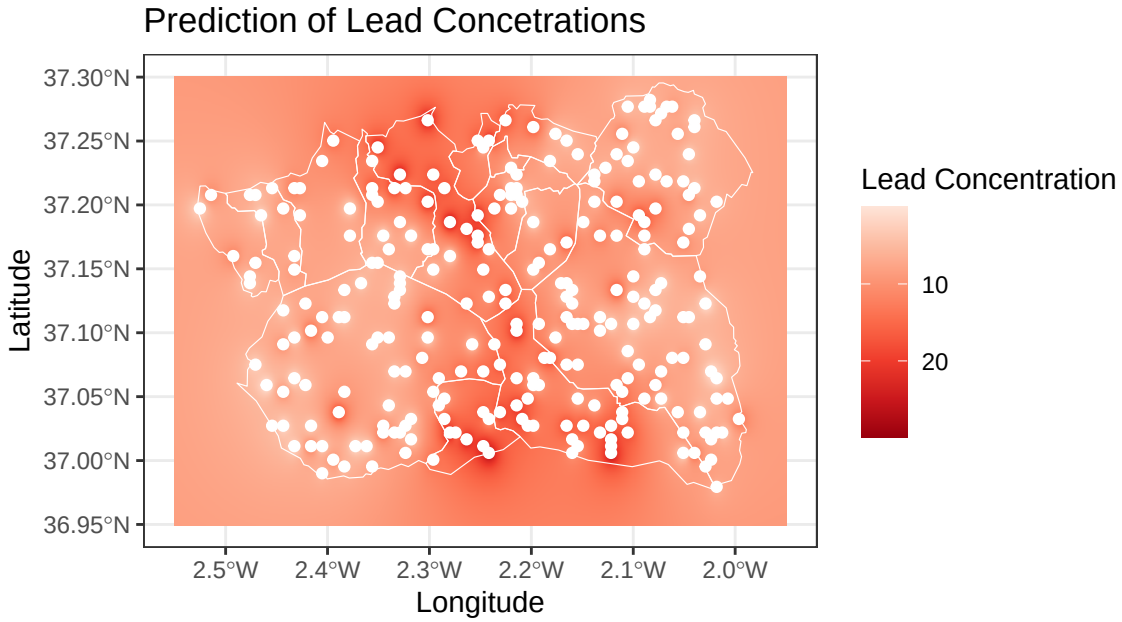
We see that there is a line of larger measurements of lead concentration vertically through the centre of Amaurot. We also see a higher concentration on the southern border.

Lets look at the some predictions. This is point-referenced data as we have exact locations of each measurement, given by the Longitude and Latitude co-ordinates, so we will use inverse-distance weighting.

```
IDW <- function( X, S, s_star, p){
  d <- sqrt( (S[,1]-s_star[1])^2 + (S[,2]-s_star[2])^2 )
  w <- d^(-p)
  if( min(d) > 0 )
    return( sum( X * w ) / sum( w ) )
  else
    return( X[d==0] )
}

lon <- seq( -2.55, -1.95, by=0.001 )
lat <- seq( 36.95, 37.30, by=0.001 )
pixels <- as.matrix( expand.grid( lon, lat ) )
Predict <- c()
coord <- cbind( Patients$Lon, Patients$Lat )

for( j in 1:length(pixels[,1]) )
  Predict[j] <- IDW(  Patients$Lead, coord, pixels[j,], p=2 )

IDW_predict <- data.frame( "Lon"=pixels[,1], "Lat"=pixels[,2],
                           "Pred"=Predict )
```

Now plotting our predictions:

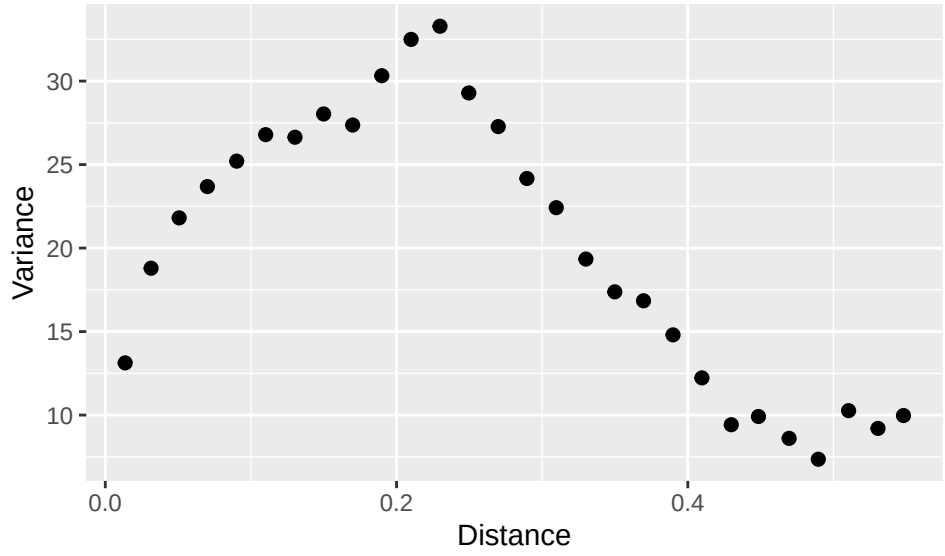
```
ggplot() + theme_bw() +
  geom_raster( data=IDW_predict, aes(x=Lon, y=Lat, fill=Pred) ) +
  scale_fill_distiller( palette="Reds", trans="reverse" ) +
  geom_sf( data=Amaurot, alpha=0.0, color="white" ) +
  geom_point( data=Patients, aes(x=Lon,y=Lat), color="white" ) +
  labs( x="Longitude", y="Latitude", fill="Lead Concentration",
        title = "Prediction of Lead Concetrations")
```



This really highlights the run of high levels of lead in drinking water through the centre of Amaurot. We have some single hotspots in other areas of Amaurot, but the bulk are in these middle districts.

We can look at spatial dependence by looking at the semi-variogram:

```
Patients_coord <- Patients
coordinates( Patients_coord ) <- ~Lon + Lat
gamma_hat <- variogram( data = Patients_coord, Patients_coord$Lead~1, width=0.02, cutoff=50 )
Semi_variogram <- ggplot( gamma_hat, aes( x=dist, y=gamma/2 ) ) + geom_point( size=2 ) +
  labs( x="Distance", y="Variance" )

Semi_variogram
```

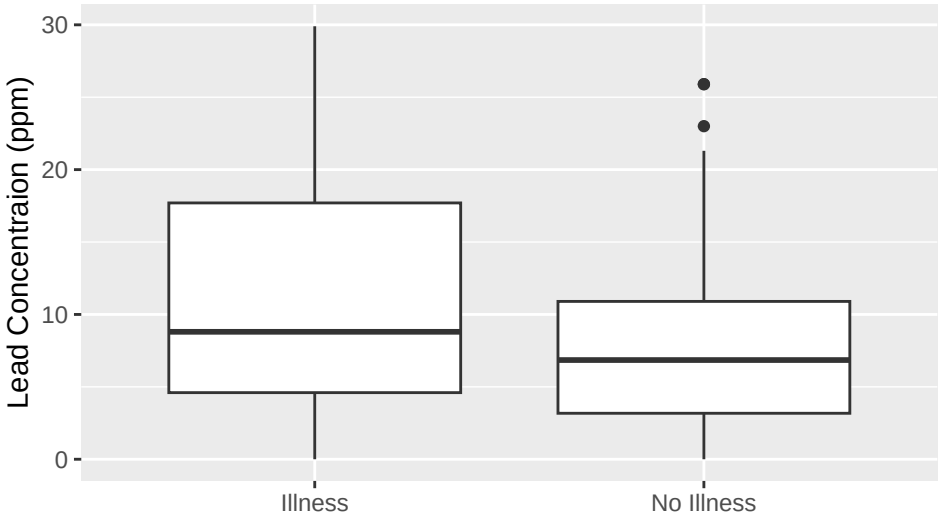


Overall we can say that spatial dependence decreases with increasing spatial distance and then we see a flip, and spatial dependence increases with decreasing spatial distance. We can interpret the semi-variogram by first looking at the smallest distance value, which is where we find medium-sized values for variance. Here we can't really conclude a strong dependence between distance and where we find similar levels of lead concentration. The curve peaks at around 0.2 and here we can assume independence, and we see some levelling off at the furthest spatial distance, this exhibits the strongest independence of the region.

b) Identify a threshold for lead concentration in drinking water beyond which the risk of a person developing kidney damage increases.

Initially, lets prove our assumption that an increased lead concentration in drinking water does have an affect on the amount of people falling ill. I am going to separate the patients into either being ill or not ill.

```
Illness <- Patients%>%
  mutate(Illness = case_when(Disease == "No treatment required"~ "No Illness",
                             Disease!="No treatment required"~ "Illness"))

Illness%>%
  group_by(Illness)%>%
  ggplot(aes( x= Illness, y = Lead)) +geom_boxplot() + labs(x = "",
                                                            y = "Lead Concentraion (ppm)")
```

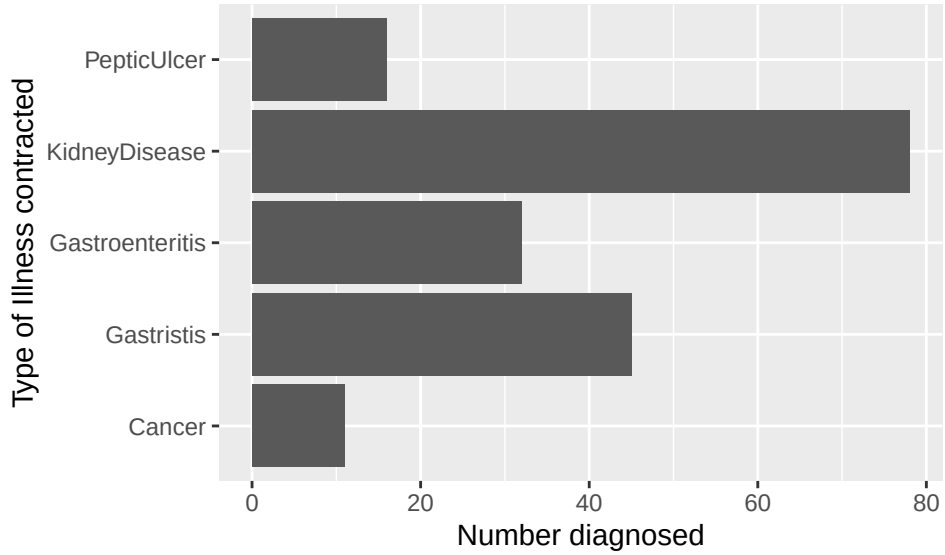


We see that our assumption is true, those diagnosed with an illness, do have a higher level of lead in their drinking water, there is definitely a link here.

Now focusing on Kidney damage:

```
Cancer <- c("Cancer", "Cancer; Gastritis", "Cancer; Gastroenteritis",
            "Cancer; Gastroenteritis; Peptic Ulcer",
            "Kidney Disease; Cancer; Gastritis",
            "Kidney Disease; Cancer; Gastroenteritis")
Gastritis <- c("Cancer; Gastritis", "Gastritis", "Gastritis; Gastroenteritis ",
               "Gastritis; Gastroenteritis; Peptic Ulcer",
               "Kidney Disease; Cancer; Gastritis",
               "Kidney Disease; Gastritis", "Kidney Disease; Gastritis; Gastroenteritis")
Gastroenteritis <- c("Cancer; Gastroenteritis", "Cancer; Gastroenteritis; Peptic Ulcer",
                     "Gastritis; Gastroenteritis",
                     "Gastritis; Gastroenteritis; Peptic Ulcer",
                     "Gastroenteritis", "Kidney Disease; Cancer; Gastroenteritis",
                     "Kidney Disease; Gastritis; Gastroenteritis",
                     "Kidney Disease; Gastroenteritis",
                     "Kidney Disease; Gastroenteritis; Peptic Ulcer")
KidneyDisease <- c("Kidney Disease", "Kidney Disease; Cancer; Gastritis",
                   "Kidney Disease; Cancer; Gastroenteritis",
                   "Kidney Disease; Gastritis",
                   "Kidney Disease; Gastritis; Gastroenteritis",
                   "Kidney Disease; Gastroenteritis",
                   "Kidney Disease; Gastroenteritis; Peptic Ulcer",
                   "Kidney Disease; Peptic Ulcer")
PepticUlcer <- c("Cancer; Gastroenteritis; Peptic Ulcer",
                 "Gastritis; Gastroenteritis; Peptic Ulcer",
                 "Kidney Disease; Gastroenteritis; Peptic Ulcer",
                 "Kidney Disease; Peptic Ulcer", "Peptic Ulcer" )
Suffered <- Illness%>%
  mutate( Cancer = case_when(Disease %in% Cancer~1, .default = 0),
          Gastritis = case_when(Disease %in% Gastritis~1, .default = 0),
          Gastroenteritis = case_when(Disease %in% Gastroenteritis~1, .default = 0),
```

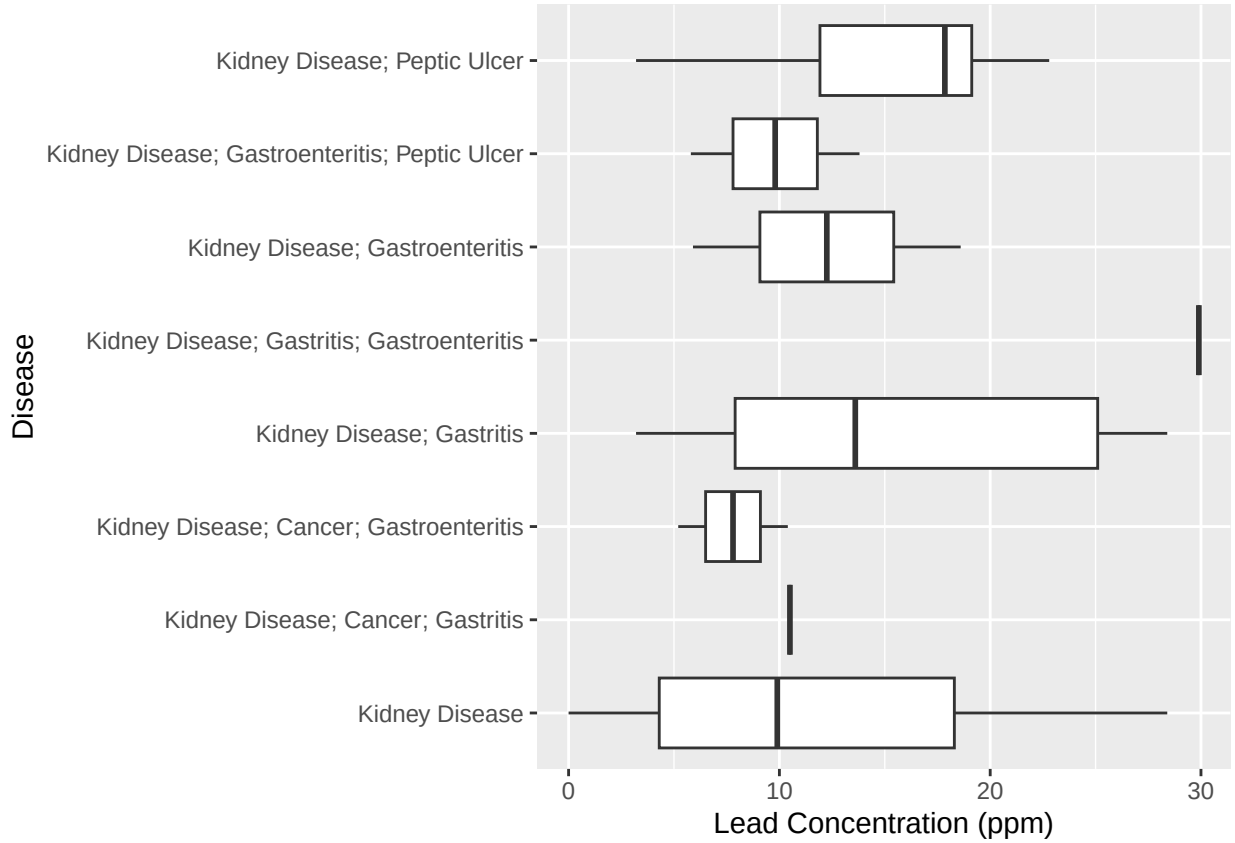
```
          KidneyDisease = case_when(Disease %in% KidneyDisease~1, .default = 0),
          PepticUlcer= case_when(Disease %in% PepticUlcer~1, .default = 0))
Mostcom<- Suffered%>%
  summarise(Cancer = sum(Cancer), Gastristis = sum(Gastritis),
            Gastroenteritis = sum(Gastroenteritis), KidneyDisease = sum(KidneyDisease),
            PepticUlcer = sum(PepticUlcer))%>%
  pivot_longer(1:5)

ggplot(data = Mostcom, aes(x =name, y = value )) + geom_col() + coord_flip() +
  labs(x = "Type of Illness contracted", y = "Number diagnosed")
```



We can see that kidney disease is the most common illness to be contracted from the houses where lead concentration was measured.

Now lets identify a threshold:

```
Patients_Kidney <- Suffered%>%
  filter(KidneyDisease ==1)%>%
  select("Lead", "Disease", "KidneyDisease")
ggplot(data = Patients_Kidney, aes(x =Disease ,y = Lead)) + geom_boxplot() +
  coord_flip() + labs(y = "Lead Concentration (ppm)")
```



For a baseline, I would choose having a lead concentration of 4.5ppm in drinking water as it is the lower quartile for developing just Kidney Disease. However, we do see that it does have the lowest concentration of lead, when diagnosed. And it makes sense that Kidney Disease can still be contracted when lead concentration is zero or close to zero; other factors are at play. It seems that the higher the concentration of lead in drinking water can lead to further complications and becoming more ill.

c) The local authorities wish to deploy a team to one of the districts to reduce lead concentration to a safe level. Which district would benefit the most from such an action?

I want to use our threshold of 4.5ppm to classify the drinking water as unsafe.

```
PatientsDistrict <- inner_join(x = Grid, y = Patients)
```

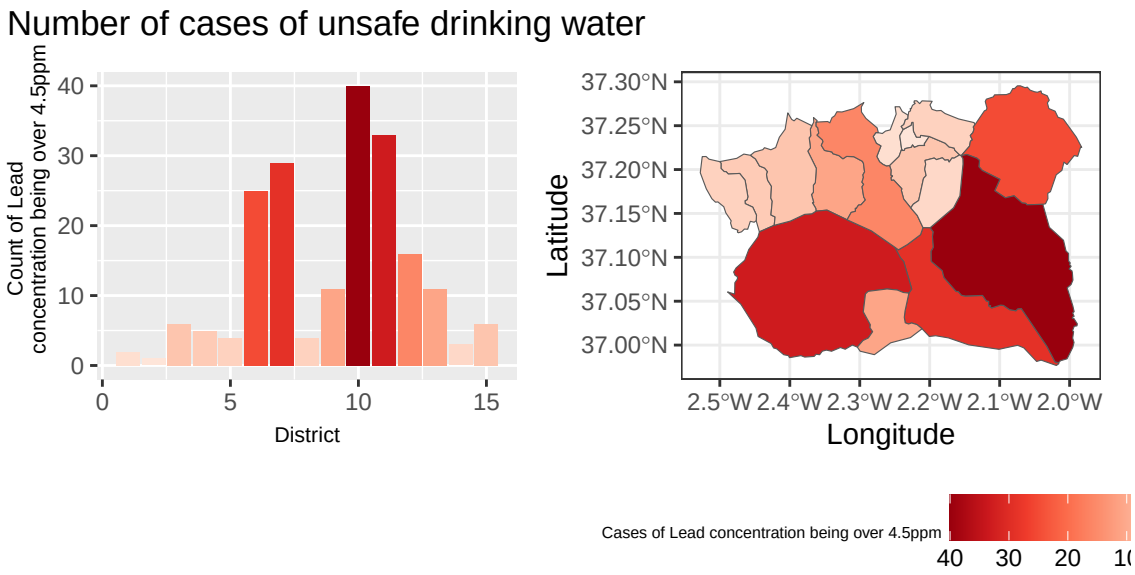
Now all the data points in the Patients dataset have their District shown.

First I'm going to just consider the number of cases of lead concentration in drinking water being unsafe. I expect to observe that large districts will have higher counts.

```
Lead_Count_bar <- PatientsDistrict%>%
  filter(Lead>=4.5)%>%
  group_by(District)%>%
  count(District)%>%
  arrange(desc(n))%>%
  ggplot(aes(x = District, y = n, fill = n)) + geom_col() +
    labs(y = "Count of Lead
         concentration being over 4.5ppm") +
```

```
          scale_fill_distiller(palette = "Reds", trans = "reverse") +
  theme(legend.position = "none", axis.title=element_text(size=8))
```

```
Lead_Count<- PatientsDistrict%>%
  filter(Lead>= 4.5)%>%
  group_by(District)%>%
  count(District)%>%
  rename(LeadCount = n)

Amaurot<- Amaurot%>%
  mutate( Index=1:nrow(Amaurot) )
Amaurot <-inner_join(x = Amaurot, y = Lead_Count, by = c("Index"="District" ))


Lead_Count_map <- ggplot(data = Amaurot, aes(fill = LeadCount)) + theme_bw() + geom_sf() +
  scale_fill_distiller(palette = "Reds", trans = "reverse") +
  labs( x = "Longitude", y = "Latitude",
        fill = "Cases of Lead concentration being over 4.5ppm") +
  theme(legend.position = 'bottom', legend.title=element_text(size=6))
Lead_Count_bar + Lead_Count_map +
  plot_annotation(title = "Number of cases of unsafe drinking water")
```

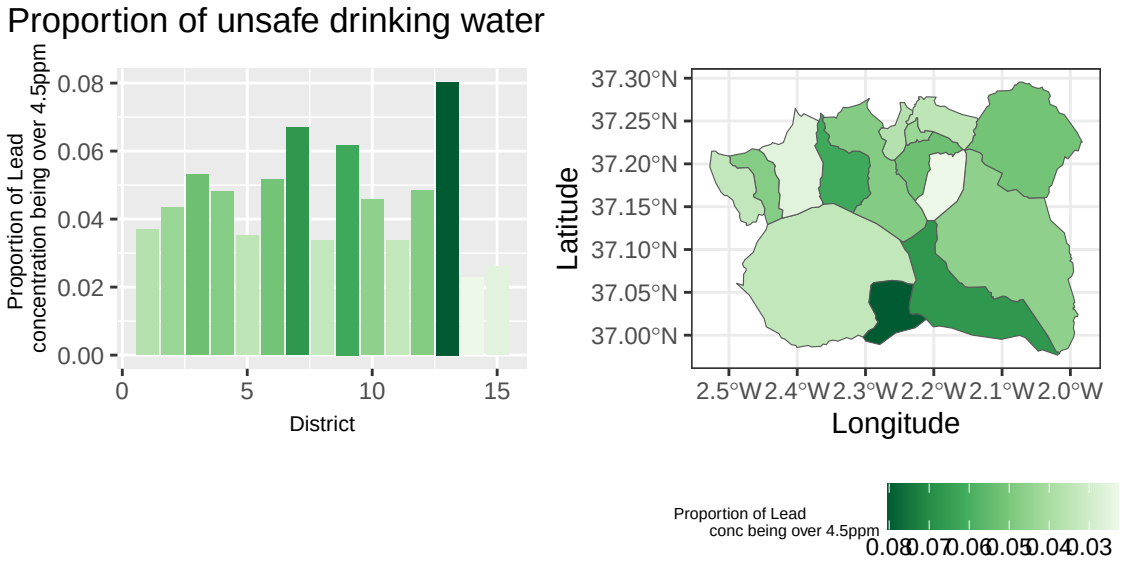


We see that District 10 has the most cases of the lead concentration in drinking water being over the level where we start to see an increased level of Kidney disease. I.e it has the most cases of unsafe drinking water. Therefore, I could advise that district 10 receives the special action on reducing the lead concentration in the drinking water. Looking at this count of unsafe drinking water on a map of Amaurot, we can see where each district lies and its count of unsafe drinking water. We observe that district 10 does appear to be the biggest, which would explain why it has the highest number of cases.

I now want to consider the size of each district, so I'll use the number of points associated to each district given in the Grid data.

```
Area <- Grid%>%
  group_by(District)%>%
  count(District)%>%
  rename("Area" = n)
Count <- PatientsDistrict%>%
  filter(Lead>=4.5)%>%
  group_by(District)%>%
  count(District)
Prop <- inner_join(x = Count, y = Area)%>%
  mutate( Proportion = n/Area)
```

```
Lead_Prop_bar <- ggplot(data = Prop, aes(x = District, y = Proportion,
                                         fill = Proportion)) + geom_col() +
  scale_fill_distiller(palette = "Greens", trans = "reverse")+
  labs(y = "Proportion of Lead
       concentration being over 4.5ppm")+
  theme(legend.position = "none", axis.title=element_text(size=8))

Amaurot<- Amaurot%>%
  mutate( Index=1:nrow(Amaurot) )
Amaurot <-inner_join(x = Amaurot, y = Prop, by = c("Index"="District" ))

Lead_Prop_map <- ggplot(data = Amaurot, aes(fill = Proportion)) + theme_bw() + geom_sf() +
  scale_fill_distiller(palette = "Greens", trans = "reverse") +
  labs( x = "Longitude", y = "Latitude",
        fill = "Proportion of Lead
        conc being over 4.5ppm")+
  theme(legend.position = 'bottom', legend.title=element_text(size=6))

Lead_Prop_bar + Lead_Prop_map +
  plot_annotation(title = "Proportion of unsafe drinking water")
```



This gives us a different answer, we see that District 13 has the highest proportion of unsafe drinking water relative to it's size. This suggests that choosing District 10 as to receive help, will be a larger job, and will reduce the most amount of unsafe drinking water cases. But choosing District 13 will be more cost and time effective as it has the highest concentration of unsafe drinking water. I would advise District 13 to receive the help.

d) Write a non-scientific summary of your analysis for parts (a)-(c) that can be understood by someone with A-level Mathematics knowledge. State possible recommendations that may be of interest to the authorities of Amaurot.

When exploring the differencing in the measured lead concentration across all of Amaurot, we find that there's an obvious increased level through the middle of the country, as well as along the south coast. As for the rest of the country, we see relatively low levels of lead in drinking water on the east and west coasts. The districts that have the most cases of unsafe drinking water are the largest ones- District 10 and 11. So District 10 could be argued at the place to deploy a team to reduce the lead concentration to a safe level. However, further analysis was conducted, taking into account the size of each district and it was found that District 13 had the highest concentration of high lead concentration in drinking water. So it would be both more time and cost effective to choose District 13. Hence my recommendation would be to deploy the team to District 13.

Another insight is the illnesses contracted due to the lead contamination, and overall, higher lead concentration caused a greater number of illnesses diagnosed. The most common was found to be Kidney Disease, almost 80 Amaurot citizens suffering from this condition. It is reasonable to make the conclusion that higher levels in lead concentration also lead to further diseases such as Cancer, Gastritis, Gastroenteritis and Peptic Ulcers being diagnosed. I found 4.5 ppm to be the threshold for lead concentration in drinking water, beyond which drinking water becomes unsafe and causes illness as 75% of those that contracted Kidney Disease had over this level in their drinking water.