

Winter/Summer - stats testing

evidence of meeting PDA outcome 2.6.

Statistical analyses to identify patterns, trends and relationships in the dataset

evidence of meeting PDA outcome 2.7.

Interpreting the results of statistical analysis

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr  0.3.5
## v tibble  3.1.8      v dplyr  1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(infer)
library(lubridate)
```

```
## Loading required package: timechange
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##   date, intersect, setdiff, union
```

define functions

```
source(here::here("r_scripts_and_notebooks/fun_smoother.R"))
source(here::here("r_scripts_and_notebooks/fun_stats_test.R"))
```

some not so successful attempts at setting up code for use in dashboard making it so it will run with any dataset and not just this one note that data must just be one timeries. date must be date format either weekly or monthly.

load data here

```
admissions_spec <- read_csv(here::here("clean_data/weekly_admissions_spec_clean.csv"))
delayed_discharge <- read_csv(here::here("clean_data/delayed_discharge_clean.csv"))
```

#define filters

```
input <- list(dataset = "delayed_discharge", age_group = "All (18plus)", healthboard = "All Scotland")
```

```

#define parameters to plot
selection <- 1
lookup_dataset <- list("delayed_discharge","admissions_spec")
lookup_param <- list("number_of_delayed_bed_days","number_admissions")
lookup_date <- list("wdate","mdate")
lookup_indicator <- list("weekly","monthly")

# how to evaluate when passing in variable names?
eval(str2expression(str_c("seldata <- ",input$dataset)))
eval(str2expression(str_c("choice1=unique(seldata$",lookup_param[1],")")))

```

start here

this is an example where we need to remove long term trend if we are to reveal any seasonal differences

```

seldata <- delayed_discharge %>%
  filter(hb_name == "All Scotland") %>%
  filter(age_group == "All (18plus)") %>%
  #using this filter - dont expect a significant result
  filter(reason_for_delay == "All Delay Reasons") %>%
  # using this filter expect a significant result
  #filter(reason_for_delay == "Code 9 Non-AWI") %>%
  # here we are selecting just post pandemic data
  filter(mdate > as.Date("2020-04-01")) %>%
  # by just having param it will make app easier
  rename(param = number_of_delayed_bed_days) %>%
  select(mdate, param, iswinter)

data <- seldata[2]
date <- seldata[1]

```

call function to smooth data

this will make it easier to embed in shiny app

```
smoothed_data <- data_smoother(date,seldata)
```

call function to run test and calc p-value

used code from lines below to create this function - we should get similar pvalue. note that because we are using bootstrap pvalues could be slightly different each time we run.

```
p_value2 <- stats_test(smoothed_data)
p_value2
```

```
## # A tibble: 1 x 1
##   p_value
##   <dbl>
## 1    0.388
```

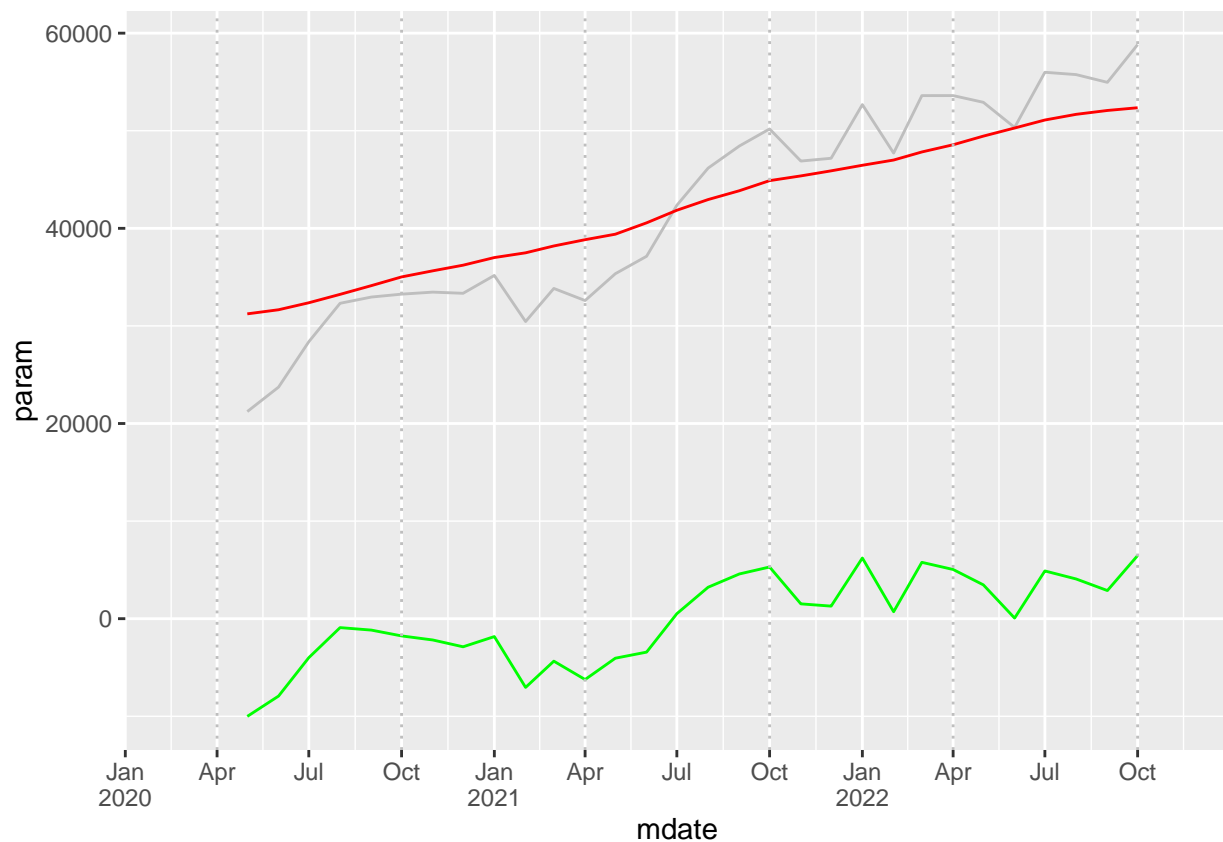
demo that function works

```
## plot data
#calculate difference to create mvar
```

```
smoothed_data <- smoothed_data %>%
  mutate(mvar = param - moving_avg)
```

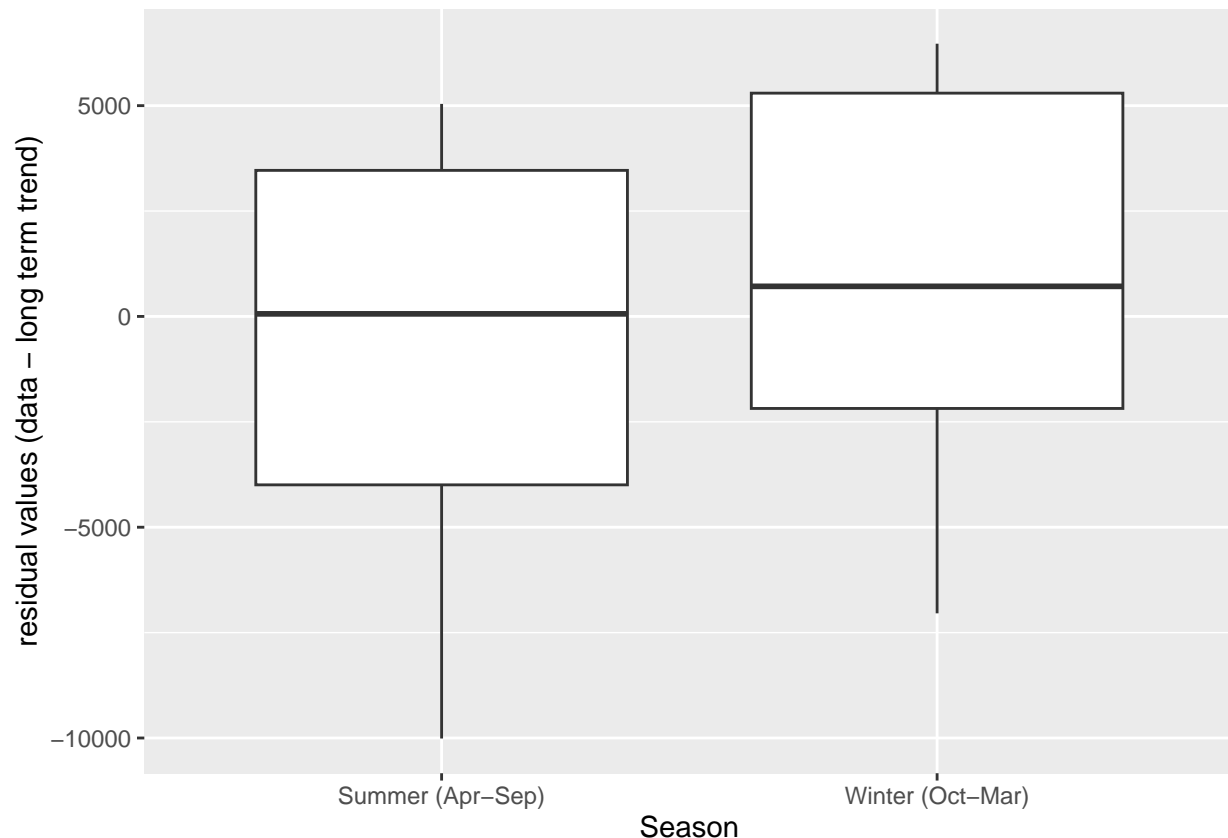
plot - check the smoother makes sense and tweak values if they don't line plot can be added to the app - show pre pandemic and post pandemic in different plots in this plot seasonal var much greater than long term var

```
plotlim <- as.Date(c("2020-01-01", "2022-12-31"))
smoothed_data %>%
  ggplot() +
    #geom_line(aes(x = mdate, y = c(param, moving_avg, mvar)))
    #above is an attempt to merge them - doesnt work!
    geom_line(aes(x = mdate, y = param), colour = "gray") +
    geom_line(aes(x = mdate, y = moving_avg), colour = "red") +
    geom_line(aes(x = mdate, y = mvar), colour = "green") +
    scale_x_date(limits=plotlim, date_breaks="3 month",
      labels = scales::label_date_short(), expand = c(0,0)) +
    # add dashed lines to show the boundaries
    geom_vline(xintercept=ymd(20201001), color="gray", linetype="dotted", linewidth = 0.5) +
    geom_vline(xintercept=ymd(20211001), color="gray", linetype="dotted", linewidth = 0.5) +
    geom_vline(xintercept=ymd(20221001), color="gray", linetype="dotted", linewidth = 0.5) +
    geom_vline(xintercept=ymd(20200401), color="gray", linetype="dotted", linewidth = 0.5) +
    geom_vline(xintercept=ymd(20210401), color="gray", linetype="dotted", linewidth = 0.5) +
    geom_vline(xintercept=ymd(20220401), color="gray", linetype="dotted", linewidth = 0.5)
```



#now create a boxplot this box plot will be added to the app

```
smoothed_data %>%
  ggplot() +
  aes(x=iswinter, y=mvar) +
  geom_boxplot() +
  ylab("residual values (data - long term trend)") +
  xlab("Season") +
  scale_x_discrete(
    labels=c("FALSE" = "Summer (Apr-Sep)", "TRUE" = "Winter (Oct-Mar)"),
    limits = c("FALSE", "TRUE"))
```



This box plot shows that for this dataset, the median values in summer were lower than the median values in winter, so there is a suggestion that there could be a seasonal variability here. However it is necessary to do a statistical test to determine if these differences are statistically significant.

Hypothesis Test

Is summer is significantly different to winter?

The null hypothesis is that the average value for summer is the same as the average value for winter. The alternative hypothesis is that winter was significantly different

$H_0 : \text{mean winter} - \text{mean summer} = 0$ $H_A : \text{mean winter} - \text{mean summer} \neq 0$

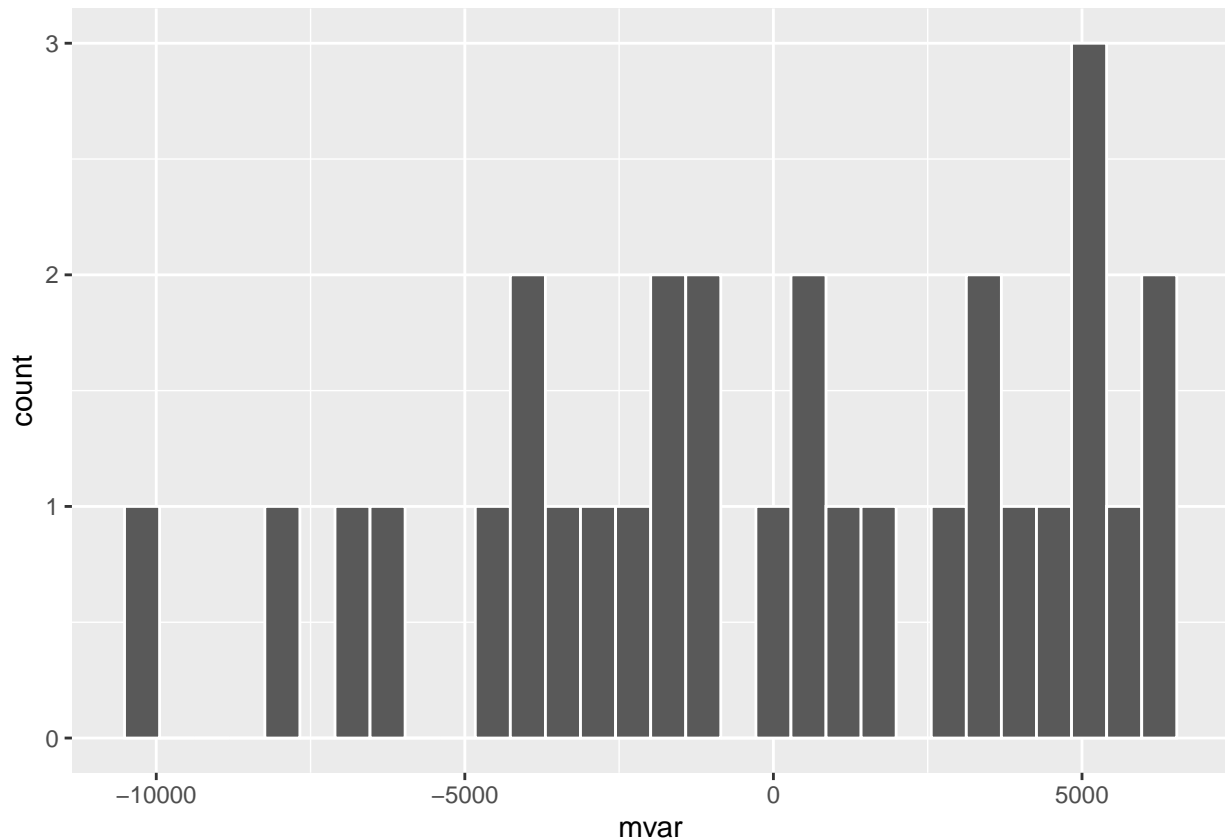
we are setting alpha for this test to be 0.05. This is the level that we need our p-value to be below in order to consider that our test is significant. alpha is the false positive rate, it shows us how unusual the result is.

```
alpha_test=0.05
```

the sample size here is quite low

```
smoothed_data %>%
  ggplot() +
  aes(x = mvar) +
  geom_histogram(col = "white")
```

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



```
seasonal_avg <- smoothed_data %>%
  group_by(iswinter) %>%
  summarise(mean = mean(mvar, na.rm=TRUE))
```

```
seasonal_avg
```

```
## # A tibble: 2 x 2
##   iswinter mean
##   <lgl>     <dbl>
## 1 FALSE   -530.
## 2 TRUE    555.
```

```
observed_stat = seasonal_avg$mean[seasonal_avg$iswinter ==TRUE]
               -seasonal_avg$mean[seasonal_avg$iswinter ==FALSE]
```

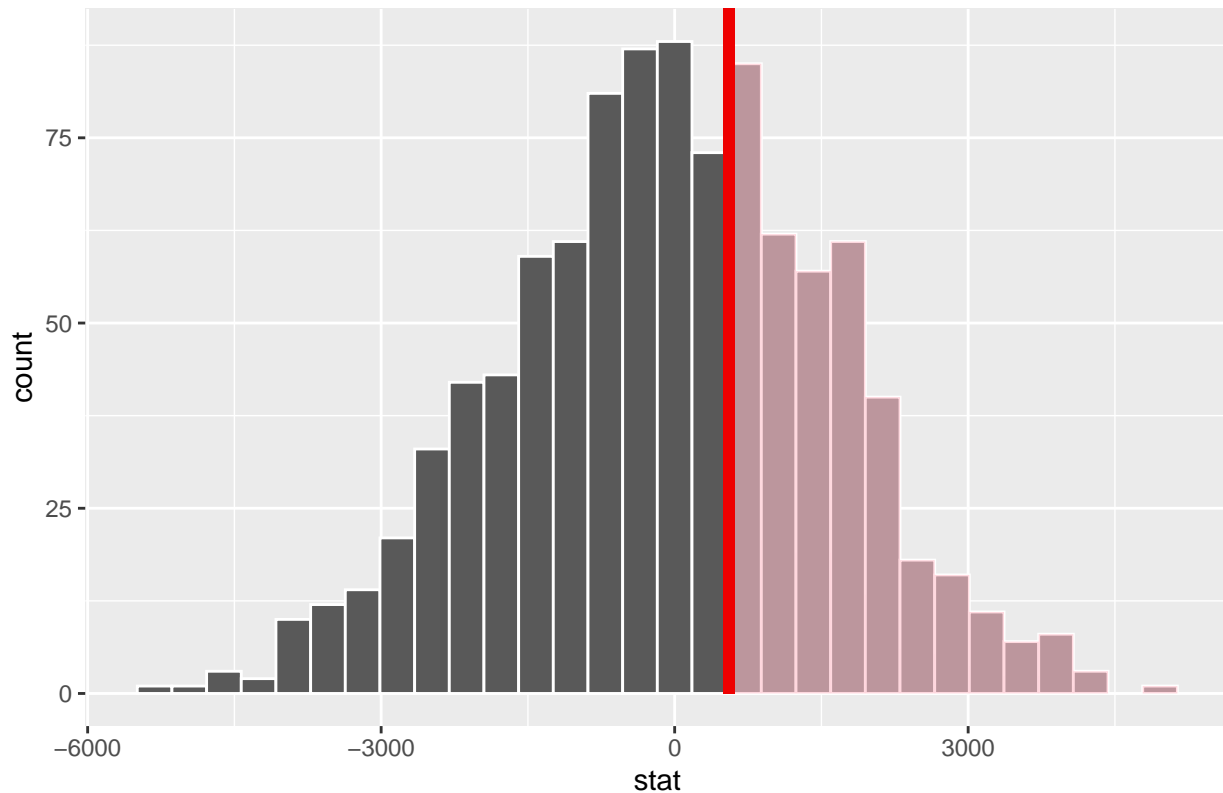
```
## [1] 529.8216
```

```
null_distribution <- smoothed_data %>%
  specify(response = mvar, explanatory = iswinter) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
```

```
calculate(stat = "diff in means", order = c(TRUE,FALSE)) #winter - summer

null_distribution %>%
  visualise(bins = 30) +
  shade_p_value(obs_stat = observed_stat
               , direction = "greater")
```

Simulation-Based Null Distribution



```
p_value <- null_distribution %>%
  get_p_value(obs_stat = observed_stat
             , direction = "greater") %>%
  pull()
```

This plot shows the null distribution, the red line is the observed statistic (difference in means of ~529). The red line is close to the edge of the null distribution.

```
p_value
```

```
## [1] 0.363
```

If pvalue is less than alpha we can reject null hypothesis If pvalue is greater than alpha we cannot reject null hypothesis

in this case the p_value of 0.369 is greater than 0.05 and this means we cannot reject the null hypothesis. We cannot say that the two samples means are significantly different to each other. So for this dataset there is no significant difference between summer and winter values.

```
if(p_value > alpha_test){
  print("We cannot reject the null hypothesis. The average winter values in
        this data are the same as the average summer values")
}
```

```

}else{
  print("We can reject the null hypothesis in favour of the alternative
        hypothesis. The average winter values in this data are
        significantly different to the average winter values")}]

```

```

## [1] "We cannot reject the null hypothesis. The average winter values in \n      this data are the s
#Notes smoother not perfect, hard to remove long term trend in a short dataset maybe need to smooth less?
this will do for now though ...

```

repeat of code with diff dataset

The case below is an an example where we definitely expect winter summer to be different

```

seldata <- delayed_discharge %>%
  filter(hb_name == "All Scotland") %>%
  filter(age_group == "All (18plus)") %>%
  #using this filter - dont expect a significant result
  #filter(reason_for_delay == "All Delay Reasons") %>%
  # using this filter expect a significant result
  filter(reason_for_delay == "Code 9 Non-AWI") %>%
  # here we are selecting just post pandemic data
  filter(mdate > as.Date("2020-04-01")) %>%
  # by just having param it will make app easier
  rename(param = number_of_delayed_bed_days) %>%
  select(mdate, param, iswinter)

data <- seldata[2]
date <- seldata[1]

```

call function to smooth data

this will make it easier to embed in shiny app

```

smoothed_data <- data_smoother(date,seldata)

```

call function to run test and calc p-value

used code from lines below to create this function - we should get similar pvalue. note that because we are using bootstrap pvalues could be slightly different each time we run.

```

p_value2 <- stats_test(smoothed_data)
p_value2

```

```

## # A tibble: 1 x 1
##   p_value
##   <dbl>
## 1     0.03

```

demo that function works

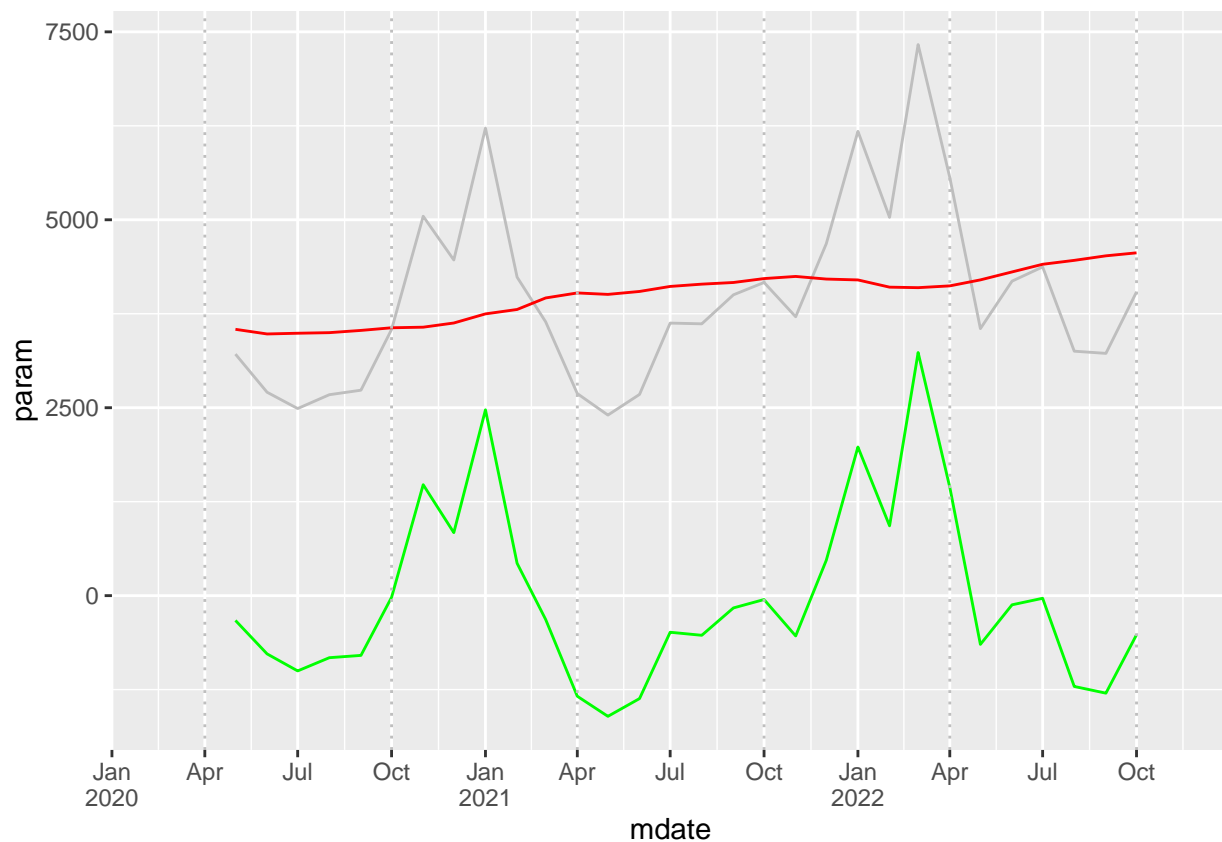
```

## plot data
#calculate difference to create mvar
smoothed_data <- smoothed_data %>%
  mutate(mvar = param - moving_avg)

```

plot - check the smoother makes sense and tweak values if they dont line plot can be added to the app - show pre pandemic and post pandemic in different plots in this plot seasonal var much greater than long term var

```
plotlim <- as.Date(c("2020-01-01","2022-12-31"))
smoothed_data %>%
ggplot() +
  #geom_line(aes(x = mdate, y = c(param, moving_avg, mvar)))
  #above is an attempt to merge them - doesnt work!
  geom_line(aes(x = mdate, y = param), colour = "gray") +
  geom_line(aes(x = mdate, y = moving_avg), colour = "red") +
  geom_line(aes(x = mdate, y = mvar), colour = "green") +
  scale_x_date(limits=plotlim, date_breaks="3 month",
               labels = scales::label_date_short(), expand = c(0,0)) +
  # add dashed lines to show the boundaries
  geom_vline(xintercept=ymd(20201001),color="gray",linetype="dotted", linewidth = 0.5) +
  geom_vline(xintercept=ymd(20211001),color="gray",linetype="dotted", linewidth = 0.5) +
  geom_vline(xintercept=ymd(20221001),color="gray",linetype="dotted", linewidth = 0.5) +
  geom_vline(xintercept=ymd(20200401),color="gray",linetype="dotted", linewidth = 0.5) +
  geom_vline(xintercept=ymd(20210401),color="gray",linetype="dotted", linewidth = 0.5) +
  geom_vline(xintercept=ymd(20220401),color="gray",linetype="dotted", linewidth = 0.5)
```

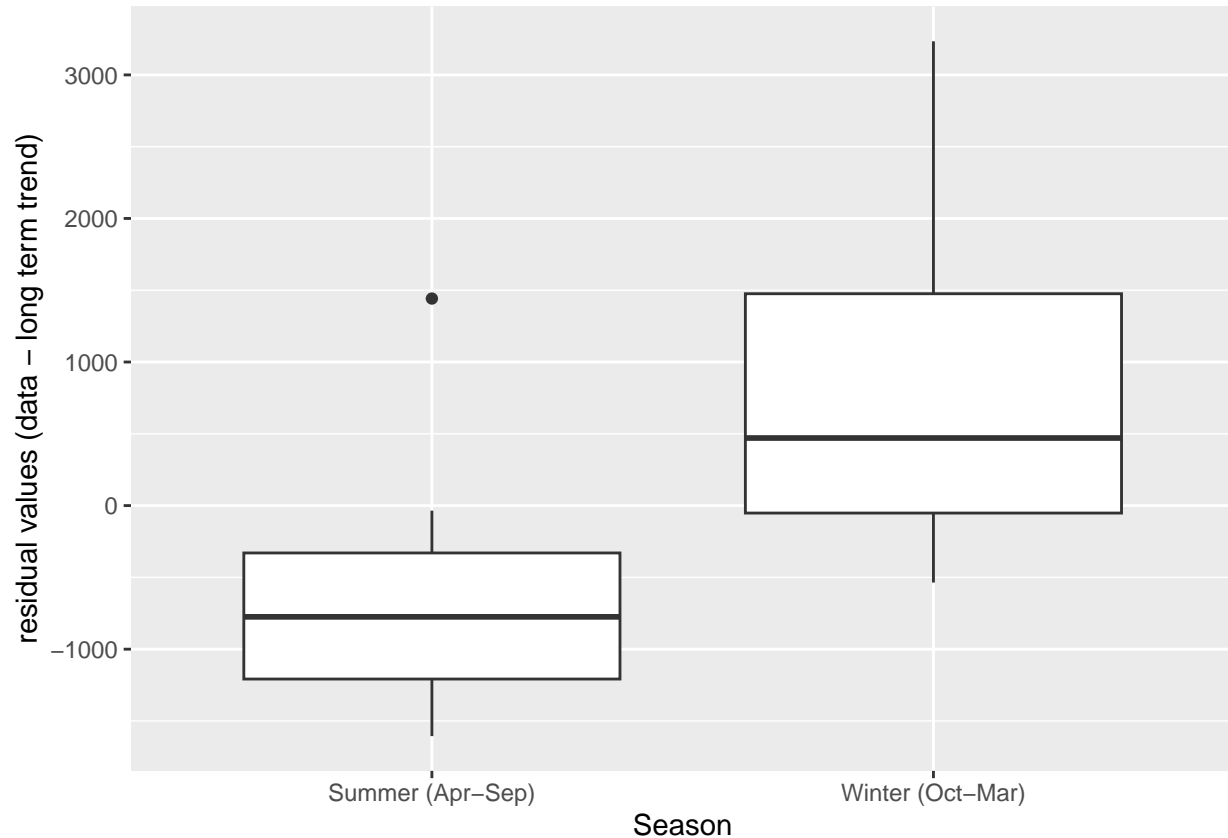


#now create a boxplot this box plot will be added to the app

```
smoothed_data %>%
ggplot() +
  aes(x=iswinter, y=mvar) +
  geom_boxplot() +
  ylab("residual values (data - long term trend)") +
```



```
xlab("Season") +
scale_x_discrete(
  labels=c("FALSE" = "Summer (Apr-Sep)", "TRUE" = "Winter (Oct-Mar)"),
  limits = c("FALSE", "TRUE"))
```



The boxplot here shows that the median value on winter is higher than the median value for summer months. The distribution of samples is also different with much less overlap than in the previous examples. It is likely that these samples are statistically different but we need to run the test in order to be sure.

Hypothesis Test

Is summer is significantly different to winter?

The null hypothesis is that the average value for summer is the same as the average value for winter. The alternative hypothesis is that winter was significantly higher

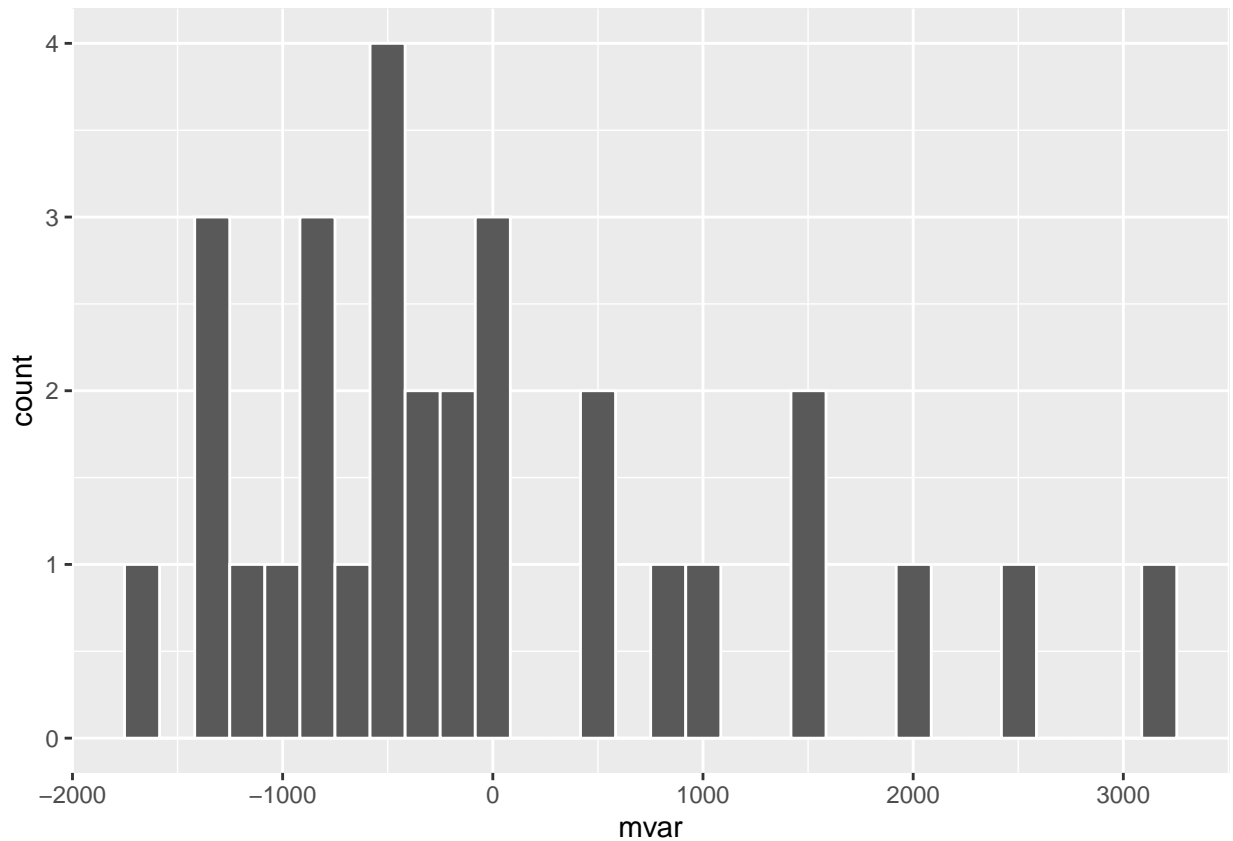
H_0 : mean winter - mean summer ≤ 0 H_A : mean winter - mean summer > 0

we are setting alpha for this test to be 0.05

```
alpha_test=0.05
```

```
smoothed_data %>%
  ggplot() +
  aes(x = mvar) +
  geom_histogram(col = "white")
```

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



```
seasonal_avg <- smoothed_data %>%
  group_by(iswinter) %>%
  summarise(mean= mean(mvar, na.rm=TRUE))
```

```
seasonal_avg
```

```
## # A tibble: 2 x 2
##   iswinter mean
##   <lgl>     <dbl>
## 1 FALSE   -652.
## 2 TRUE     798.
```

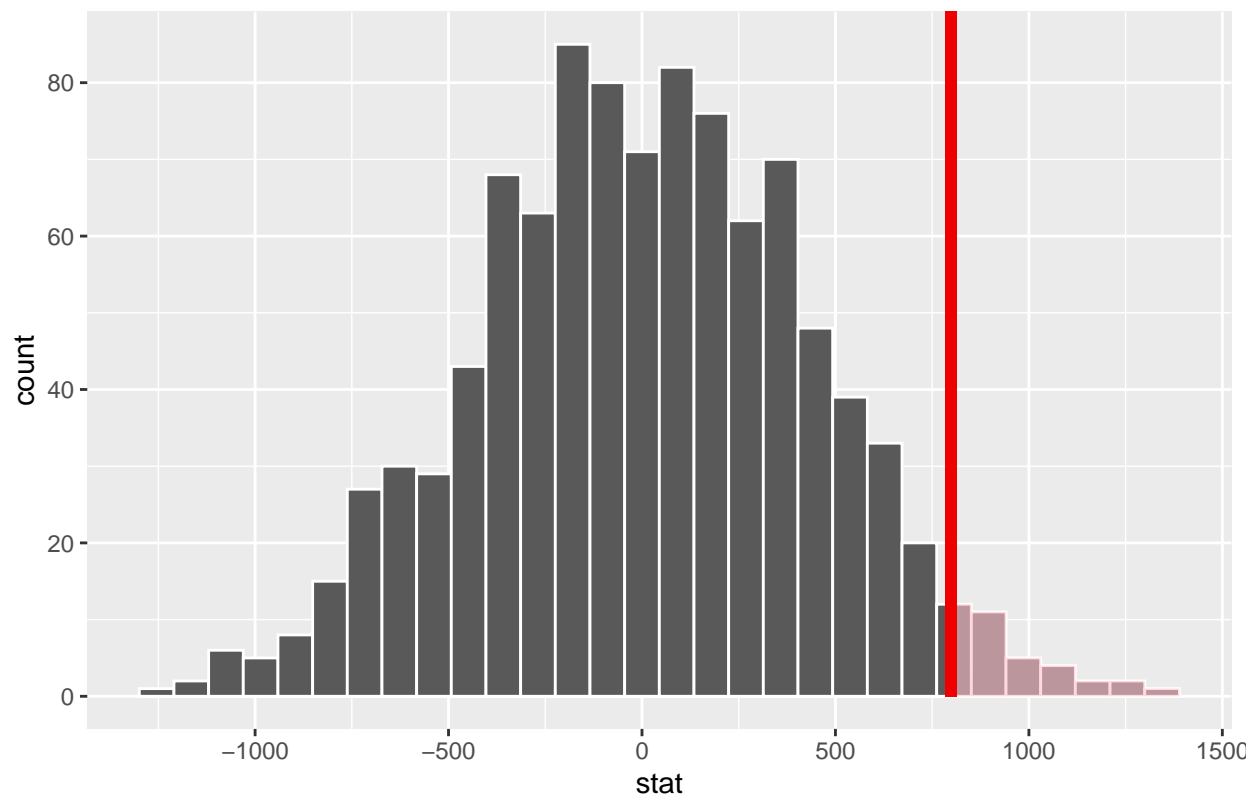
```
observed_stat = seasonal_avg$mean[seasonal_avg$iswinter ==TRUE]
                -seasonal_avg$mean[seasonal_avg$iswinter ==FALSE]
```

```
## [1] 652.2753
```

```
null_distribution <- smoothed_data %>%
  specify(response = mvar, explanatory = iswinter) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "diff in means", order = c(TRUE,FALSE)) #winter - summer
```

```
null_distribution %>%
  visualise(bins = 30) +
  shade_p_value(obs_stat = observed_stat
    , direction = "greater")
```

Simulation-Based Null Distribution



```
p_value <- null_distribution %>%
  get_p_value(obs_stat = observed_stat
             , direction = "greater") %>%
  pull()
```

This plot shows the null distribution, the red line is the observed statistic (difference in means of ~652)

```
p_value
```

```
## [1] 0.029
```

If pvalue is less than alpha we can reject null hypothesis If pvalue is greater than alpha we cannot reject null hypothesis

In this case the `p_value` of 0.039 is less than 0.05. This means that we can reject the null hypothesis in favour of the alternative hypothesis. We can say that the two samples means are significantly different to each other. We can therefore say that for this dataset the winter values are significantly higher than the summer values.

```
if(p_value > alpha_test){
  print("We cannot reject the null hypothesis. The average winter values in
        this data are the same as the average summer values")
}else{
  print("We can reject the null hypothesis in favour of the alternative
        hypothesis. The average winter values in this data are
        significantly different to the average summer values")}
```

```
## [1] "We can reject the null hypothesis in favour of the alternative \n        hypothesis. The average winter values in this data are significantly different to the average summer values"
```