StatComp Project 2: Scottish weather

Your Name (s2081957, wattsjack11)

Looking at the data in question

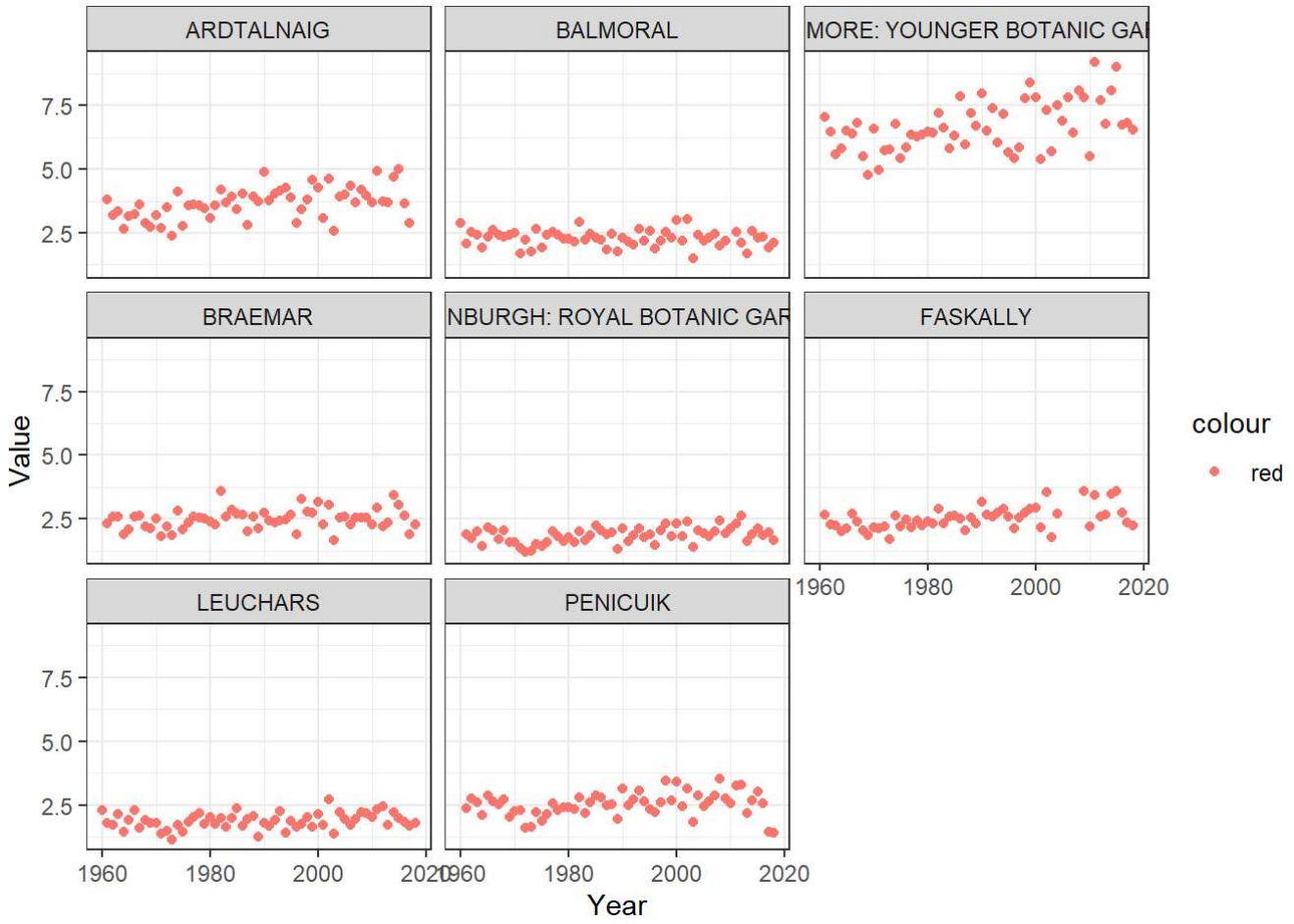
It is important that before we begin our investigation we note that we do not have the same number of results for each category. In the description of the project we were told that some values are missing due to dodgy equipment etc. however we can ignore those as we have been instructed to do. However, we still have a large number of pieces of data for maximum temperature and precipitation. The exact number of values of each can be seen in the table down below. Also for the purposes of this investigation, note that as defined in the project documentation we have Winter to be the months October to March (inclusive), and Summer the remaining time.

I suppose the only way that the lack of values could affect our investigation is if the missing data only occurred at a certain time of year.

Name	PRCP	TMAX	TMIN
ARDTALNAIG	20575	20861	20856
BALMORAL	20668	21514	21523
BENMORE: YOUNGER BOTANIC GARDE	21184	21015	21527
BRAEMAR	20301	21399	21277
EDINBURGH: ROYAL BOTANIC GARDE	21184	21483	21540
FASKALLY	19449	19451	19547
LEUCHARS	21550	21548	21548
PENICUIK	20684	21087	21130

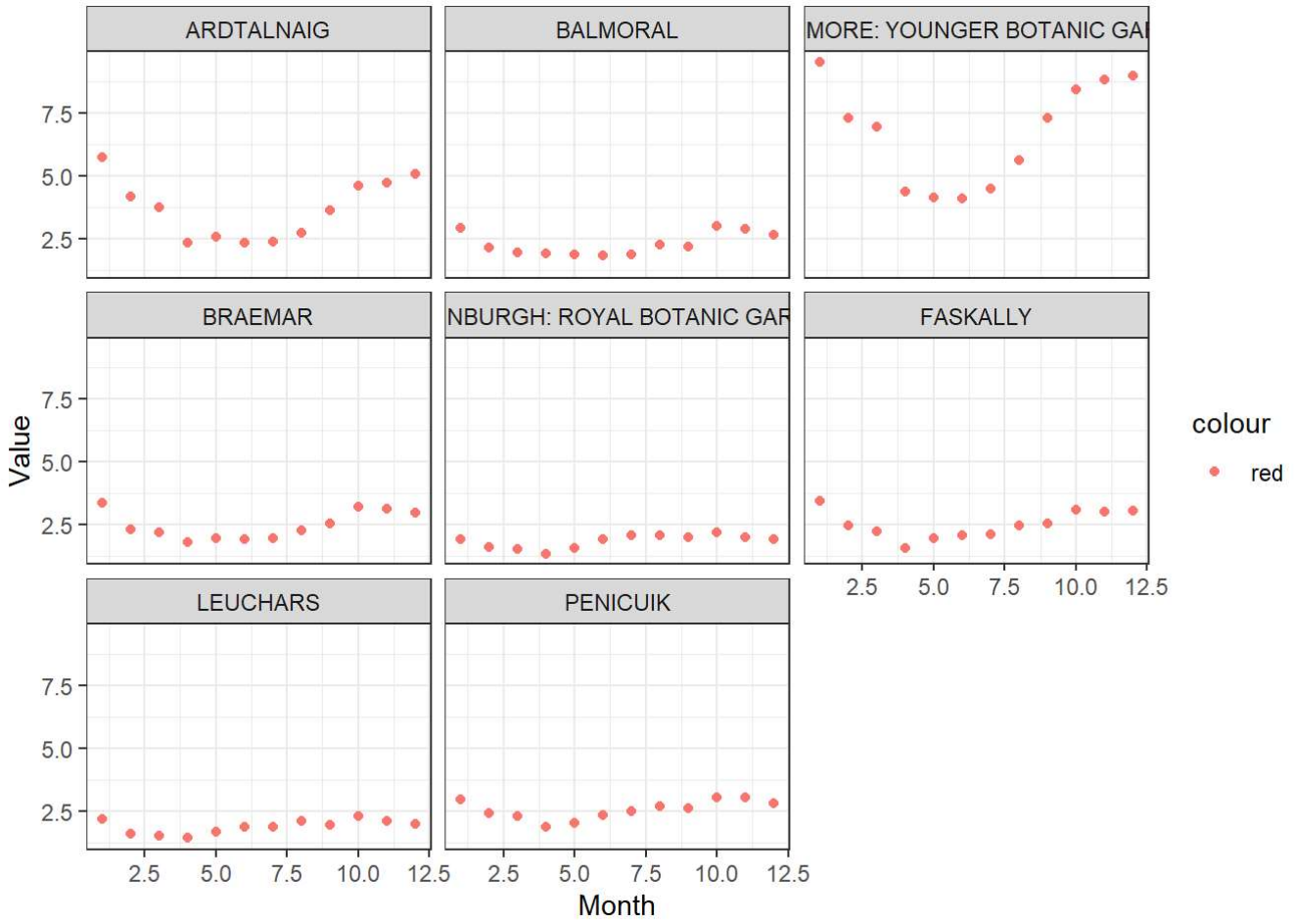
Even for Faskally, which has the least number of results by far, we still have over 19000 pieces of data for precipitation levels which is certainly enough to still make statistical judgements on.

Firstly we will look at the levels of rain fall by the year that they occurred.



We note that the average level of precipitation by year is fairly similar in most of the locations. However when you look at Benmore you notice that there is a huge variance in the average precipitation. The same can be said for Ardtalnaig too, but perhaps to a slightly lesser extent. We must bare these results in mind for our analysis in the project. We also note Faskally, there is a substantial gap in datapoints from c2004 - c2009. This perhaps can explain the fact that this station has not reported as many results as the others. Crucually, the missing results are not all in the same month. They are several years of missing results that affect each seson and month the same amount and so will not have as large of an affect. We still have roughly the same amount of datapoints for each month at Faskally

Below we can see the average levels of precipitation by month at each of the stations we have data from:



All of the graphs show a fairly similar shape i.e with an uptick of precipitation towards the beginning and the end of the year. With the excption of Ardtalnaig and Benmore the other graphs all have a fairly similar values for precipitation. Upon further investigation, the Ardtalnaig station is located around 6 miles away from Loch Tay, the sixth largest loch in Scotland. Due to the process of how rain is formed this perhaps explains why the value is so high. There will be much higher levels of evaporation occuring here. Likewise Benmore Botanic Garden is also located within Loch Lomond's National Park and so we can perhpas think that its values will be higher as a result.

Seasonal Variability

We shall now test the following two hypothesis

$$H_0: \text{The rainfall distribution is the same in Winter as it is in Summer.}$$
$$H_1: \text{The Winter and Summer distributions have different expected values.}$$

and also

$$H_0: \text{The daily probability of rainfall is the same in Winter as in Summer.}$$
$$H_1: \text{The daily probability of rainfall is different in Winter and Summer.}$$

Is precipitation seasonally varying?

We shall first discuss the first hypothesis test. Within our function MCT, we shall test our hypothesis tests.The function verifies whether we need to randomise the data and hence the data for ech station is randomised. We then use the test statistic $T = |(\text{Average over Winter}) - (\text{Average over Summer}|)$. We then compare this value to non-randomised data set's T-value and see if we have significance.

We will run this test 10000 times (hence the long run time) and each shall be compared to the genuine value.

For the occurances where the randomised data's test statistic is larger than the genuine value, then we shall find the p-value estimator.

We have two cases, i.e when our estimator is equal to zero or it is not. For the case where it is not equal to zero we can construct a confidence interval of the form:

$$\text{Confidence\_Interval}_p = (0, 1 - 0.025^{1/N})$$

The results which we have ran in the analysis file are below.

```
data <- readRDS(file = "data/final.rds")
knitr::kable(data)
```

Name	Test_1	Test_2	Test_1_CI_L	Test_1_CI_U	Test_2_CI_L	Test_2_CI_U
ARDTALNAIG	0.0000	0	0.0000000	0.9996312	0	0.9996312
BALMORAL	0.0000	0	0.0000000	0.9996312	0	0.9996312
BENMORE: YOUNGER BOTANIC GARDE	0.0000	0	0.0000000	0.9996312	0	0.9996312
BRAEMAR	0.0000	0	0.0000000	0.9996312	0	0.9996312
EDINBURGH: ROYAL BOTANIC GARDE	0.6506	0	0.6410644	0.6601356	0	0.9996312
FASKALLY	0.0000	0	0.0000000	0.9996312	0	0.9996312
LEUCHARS	0.0316	0	0.0281013	0.0350987	0	0.9996312
PENICUIK	0.0000	0	0.0000000	0.9996312	0	0.9996312

Then, we can make some conclusions. For the results that we have at Leuchars. As we have that our p-value and whole confidence interval for Leuchars is below the 0.05 value we can state that this result is significant. We can therefore definately say that for that station we can reject the null hypothesis and accept the alternate. For the otherstations, we have confinece intervals that are far too large to make an accurate and credible judgement on and so we have insufficient evidence to reject H_0.

How often does it rain?

Within the same MCT function we test the second hypothesis and for that our test statistic shall be $T = |(\text{empirical nonzero proportion for Summer}) - \text{empirical nonzero proportion for Winter}|$.

Again, in a similar fashion to the first hypothesis test we have insufficient evidence to reject H_0 due to the sheer size of the confidence intervals that have been generated.

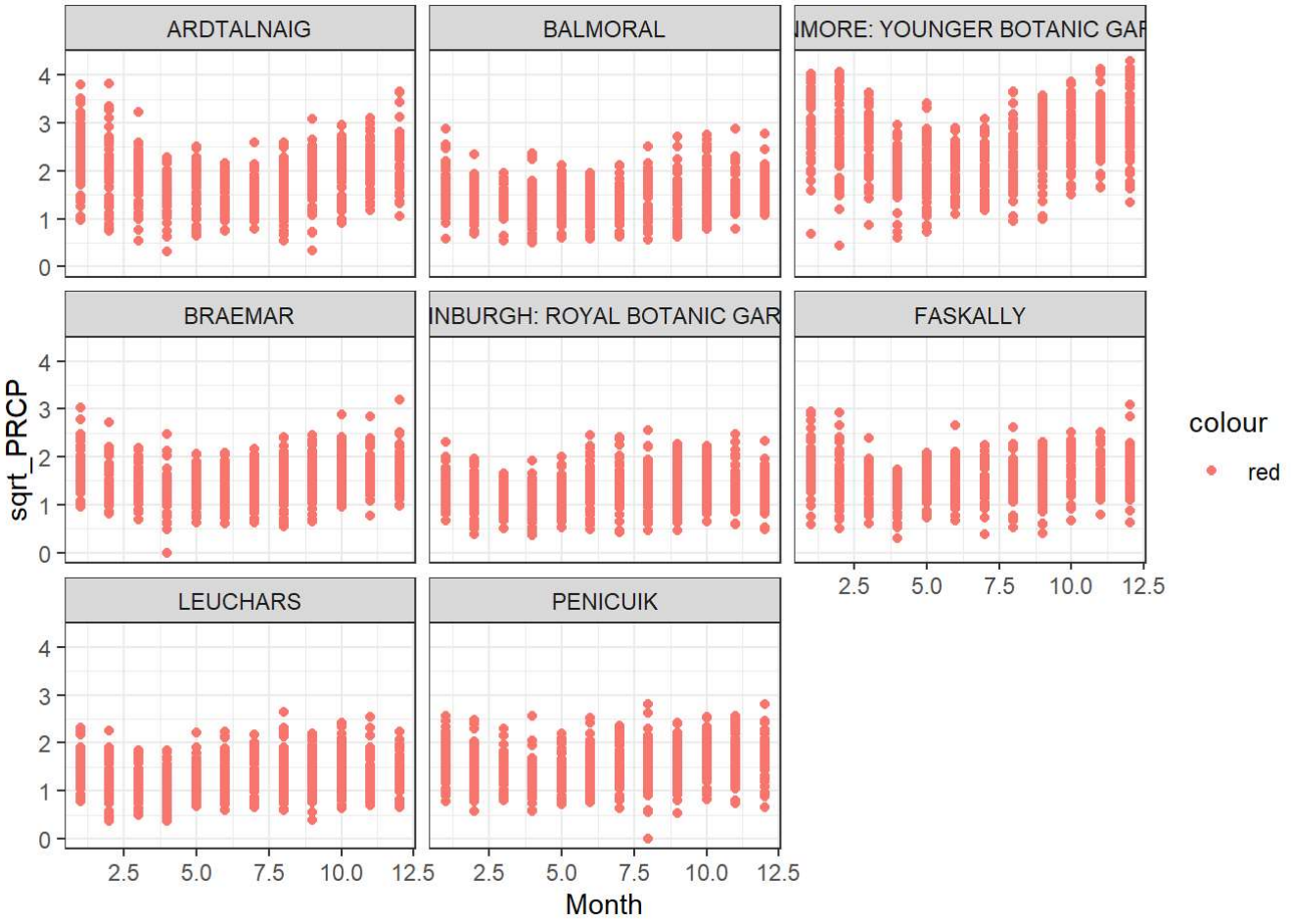
Spatial Weather Prediction

Now we shall attempt to build a model (using sin and cos) to estimate precipitation by month.

We shall now plot all the values of the monthly average amount of precipitation for each weather station below.

```
monthly_average <- ghcnd %>%
  filter(Element %in% c("PRCP")) %>%
  pivot_wider(names_from = Element, values_from = Value) %>%
  group_by(ID, Name, Year, Month, Latitude, Longitude, Elevation) %>%
  filter(!is.na(PRCP)) %>%
  summarise(Mean = mean(PRCP), .groups = "drop") %>%
  mutate(sqrt_PRCP = sqrt(Mean))

monthly_average %>%
  ggplot(aes(Month, sqrt_PRCP)) +
  geom_point(aes(colour = 'red')) +
  facet_wrap(~ Name)
```



We can see that the graphs deomstrate a distinctive wave-esque shape and so it would not be remiss to suggest that the average monthly precipitation levels are of periodic nature. Therefore uit would seem sensible to include periodic functions (like sin and cos) in our model.

Estimation and Prediction

We shall now attempt to model precipitation using periodic functions. After playing around on DESMOS I found that the curve including $\sin((2 * \pi * Month/7) - 2.5)$ and $cos((2 * \pi * Month/9) - 2.5)$ was somewhat accurate.

Using these functions, a stratified cross validation test has been conducted. Within this, values for the Dawid-Sebastiani scores and the squared errors as required.

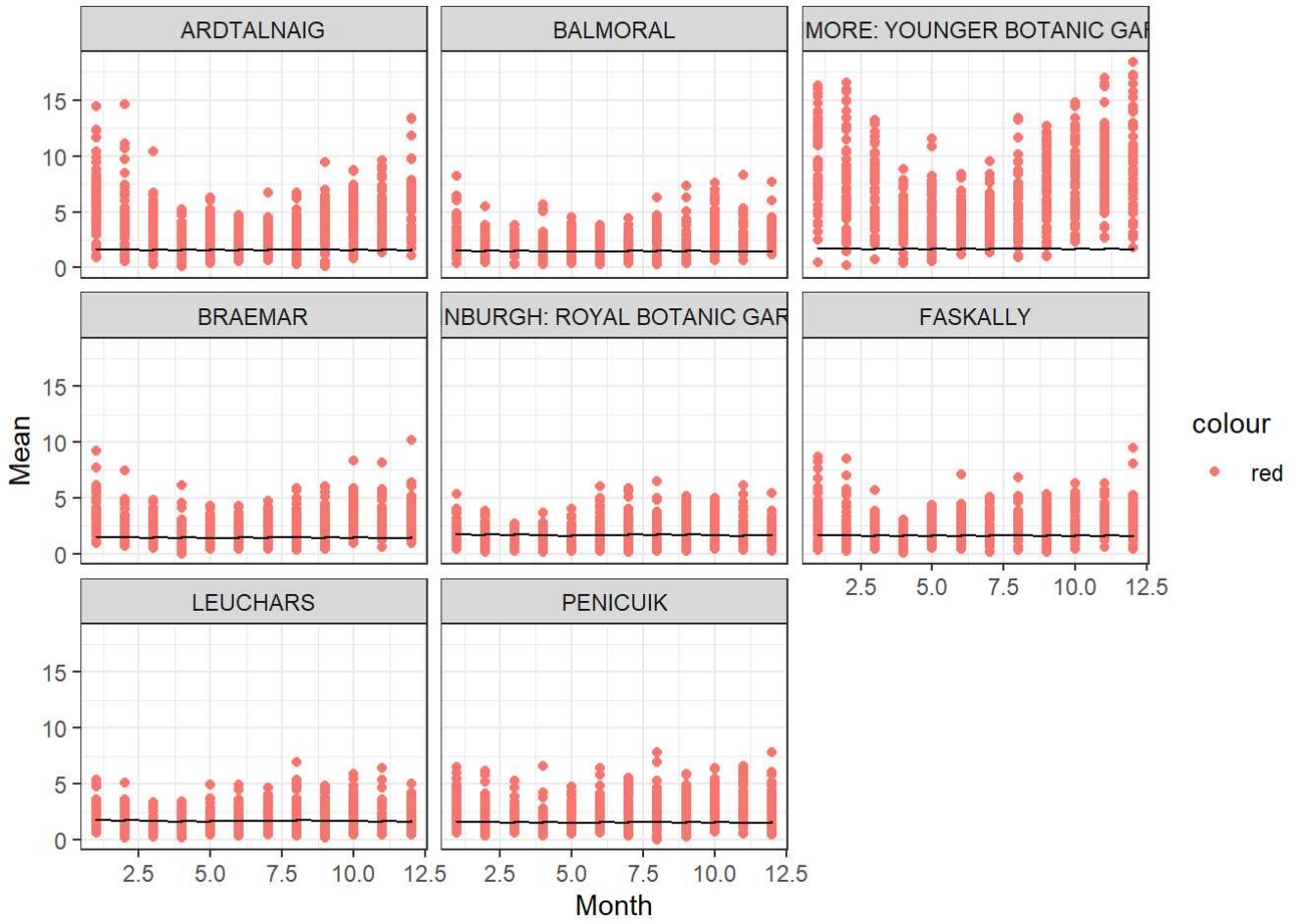
```
model_fit <- lm(sqrt_PRCP ~
  I(Latitude + Elevation + Longitude)^2 + Year + sin((2*pi * Month / 7) -2.5) + cos((2*pi * M
onth / 9)-2.5),
                monthly_average)

summary(model_fit)
```

```
##
## Call:
## lm(formula = sqrt_PRCP ~ I(Latitude + Elevation + Longitude)^2 +
##     Year + sin((2 * pi * Month/7) - 2.5) + cos((2 * pi * Month/9) -
##     2.5), data = monthly_average)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.68347 -0.39982 -0.07341  0.31058  2.58516
##
## Coefficients:
##                                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        -3.142e+00  9.599e-01  -3.273  0.00107 **
## I(Latitude + Elevation + Longitude) -7.321e-04  6.839e-05 -10.704  < 2e-16 ***
## Year                                2.462e-03  4.825e-04   5.103 3.45e-07 ***
## sin((2 * pi * Month/7) - 2.5)      -4.332e-02  1.676e-02  -2.585  0.00977 **
## cos((2 * pi * Month/9) - 2.5)       1.326e-02  1.831e-02   0.725  0.46876
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5962 on 5442 degrees of freedom
## Multiple R-squared:  0.02721,    Adjusted R-squared:  0.0265
## F-statistic: 38.06 on 4 and 5442 DF,  p-value: < 2.2e-16
```

Now we should plot the model

```
ggplot(monthly_average) + geom_point(aes(Month, Mean, colour = 'red')) + geom_line(aes(Month,
fitted(model_fit))) + facet_wrap("Name")
```



Using these functions, a stratified cross validation test has been conducted. Within this, values for the Dawid-Sebastiani scores and the squared errors. The first table is them relating to the respective weather stations that we have data from.

```
scores <- readRDS(file = 'data/scores.rds')

scores %>%
  group_by(ID, Name) %>%
  summarise(Mean_SE = mean(se), Mean_DE = mean(ds), .groups = 'drop') %>%
  knitr::kable()
```

ID	Name	Mean_SE	Mean_DE
UKE00105874	BRAEMAR	0.1867722	-0.4754506
UKE00105875	BALMORAL	0.1763091	-0.5010973
UKE00105884	ARDTALNAIG	0.3858227	0.0518155
UKE00105885	FASKALLY	0.2168829	-0.4029952
UKE00105886	LEUCHARS	0.4561127	0.2507630
UKE00105887	PENICUIK	0.1797502	-0.4931624
UKE00105888	EDINBURGH: ROYAL BOTANIC GARDE	0.4054719	0.1055307
UKE00105930	BENMORE: YOUNGER BOTANIC GARDE	1.6090199	5.9279669

From this data we can clearly see that the accuracy of our predictions was actually fairly good for many of the data stations, with particular note to Ardtalnaig and Penicuik. However as discussed earlier in the project, for the stations were we saw the greatest variance in results. Benmore was by far the worst but again this was where we saw the greatest difference in results.

Also these have been grouped together and put in a table relating to months as can be found below.

```
scores %>%
  group_by(Month) %>%
  summarise(Mean_se = mean(se), Mean_ds = mean(ds), .groups = 'drop') %>%
  knitr::kable()
```

Month	Mean_se	Mean_ds
1	0.6689347	1.4684696
2	0.5701779	0.9718271
3	0.4529694	0.5891118
4	0.3618364	0.0830861
5	0.2988207	-0.0867427
6	0.2623938	-0.2322054
7	0.2757219	-0.1740683
8	0.3759687	0.2099759
9	0.4715588	0.6536372
10	0.5546880	1.0205252
11	0.5743953	1.1362754
12	0.6293916	1.3466219

When we group the data corresponding to the months we note that for the warmer months of the year (Summer time) we have better prediction accuracy. This is when the results are a lot more stable and in the graphs that we saw at he beginning of the project, the results are fairly constant from around April to August. Though for the Winter months this is when are results are not as good, and perhaps we see less predictable weather and the most varied amount of precipitation too. This is where our model is not as effective.

Code appendix

Function definitions

```
# Your Name (s2081957, wattsjack11)

# Place your function definitions that may be needed in analysis.R and report.Rmd
# in this file, including documentation.
# You can also include any needed library() calls here
MCT <- function(rand = TRUE) {
  if (isTRUE(rand)) {
    test_data <- prcp_season %>%
      group_by(Name) %>%
      mutate(Season = sample(Season)) %>%
      group_by(Name, Season)%>%
      summarise(PRCP = mean(PRCP), Rain = (sum(Rain)/n())) %>%
      summarise(PRCP = abs(diff(PRCP)), Rain = abs(diff(Rain)))

  } else {
    test_data <- prcp_season %>%
      group_by(Name, Season) %>%
      summarise(PRCP = mean(PRCP), Rain = (sum(Rain)/n())) %>%
      summarise(PRCP = abs(diff(PRCP)), Rain = abs(diff(Rain)))

  }
}
```

Analysis code

```
# Your Name (s2081957, wattsjack11)
# Place analysis code that may take too long to run every time the report.Rmd
# document is run.
# Run the code in this file with
# source("analysis.R")
# in a fresh R session to ensure consistency of the results.
# Load function definitions
source("functions.R")
#### Delete this example
# Example object that could have taken a long time to compute
random_numbers <- rnorm(1000)
saveRDS(random_numbers, file = "data/random_numbers.rds")
####
suppressPackageStartupMessages(library(tidyverse))
theme_set(theme_bw())
suppressPackageStartupMessages(library(StatCompLab))
options(dplyr.summarise.inform = FALSE)
data(ghcnd_stations, package = "StatCompLab")
data(ghcnd_values, package = "StatCompLab")
ghcnd <- left_join(ghcnd_values, ghcnd_stations, by = "ID")

prcp_season = ghcnd %>%
  pivot_wider(names_from = Element, values_from = Value) %>%
  filter(!is.na(PRCP)) %>%
  mutate(Season =
           if_else(is.element(Month, c(1, 2, 3, 10, 11, 12)) == TRUE, FALSE, TRUE)
  ) %>%
  mutate(Rain =
           if_else(PRCP > 0, TRUE, FALSE)
  )
non_rand <- MCT(FALSE)
counter <- non_rand %>% mutate(PRCP = 0,
                               Rain = 0)
for (iter in c(1:10000)) {
  random <- MCT()
  random <- random %>% mutate(PRCP = if_else(
    (non_rand$PRCP < PRCP), 1, 0),
    Rain = if_else(
      (non_rand$Rain < Rain), 1, 0))
  counter <- counter %>% mutate(PRCP = PRCP + random$PRCP,
                                Rain = Rain + random$Rain)
}
p_estimate <- counter %>% mutate(PRCP = PRCP/10000,
                                 Rain = Rain/10000)

a = 0.025^(1/10000)
final <- data.frame(p_estimate$Name, p_estimate$PRCP, p_estimate$Rain)
names(final)[1] <- 'Name'
names(final)[2] <- "Test_1"
names(final)[3] <- 'Test_2'
final <- final %>%
  mutate(Test_1_CI_L =
           if_else(Test_1 > 0,  Test_1 - 2*sqrt((Test_1*(1 - Test_1))/(10000)), 0),
```

```
         Test_1_CI_U =
           if_else(Test_1 > 0, Test_1 + 2*sqrt((Test_1*(1 - Test_1))/(10000)), a),
         Test_2_CI_L = 0,
         Test_2_CI_U = a)
final
saveRDS(final, file = "data/final.rds")


monthly_average <- ghcnd %>%
  filter(Element %in% c("PRCP")) %>%
  pivot_wider(names_from = Element, values_from = Value) %>%
  group_by(ID, Name, Year, Month, Latitude, Longitude, Elevation) %>%
  filter(!is.na(PRCP)) %>%
  summarise(Mean = mean(PRCP), .groups = "drop") %>%
  mutate(sqrt_PRCP = sqrt(Mean))

data_places <- unique(monthly_average$ID)
arranged_values <- monthly_average %>% arrange(ID)
values_for_std_dev <- c()
values_for_mean <- c()

for (i in c(1:8)) {
  model <- lm(sqrt_PRCP ~
                I(Latitude + Elevation + Longitude)^2
              + Year + sin((2*pi * Month / 7) -2.5)
              + cos((2*pi * Month / 9)-2.5),
              data = arranged_values %>%
                filter(ID != data_places[i]))
  model_prediction <- predict(model, newdata = arranged_values %>%
                                filter(ID == data_places[i]),
  se.fit = TRUE, interval = "prediction")
  values_for_mean <- c(values_for_mean, model_prediction$fit[, "fit"])
  values_for_std_dev <- c(values_for_std_dev, sqrt(model_prediction$se.fit^2 + model_predicti
on$residual.scale^2))
}

arranged_values$mean <- values_for_mean
arranged_values$sd <- values_for_std_dev

scores <- arranged_values %>% mutate(se = proper_score('se', sqrt_PRCP, mean = mean),
                                     ds = proper_score('ds', sqrt_PRCP, mean = mean, sd = s
d))

saveRDS(scores, file = 'data/scores.rds')

# A code=readlines() code chunk in the report appendix will include the code
# in the report, without running it.
#
# You can place long-running analysis code in this file,
# and save results using
# saveRDS(object, file = "data/object.rds")
# When the results are needed in the report.Rmd file, use
# object <- readRDS(file = "data/object.rds")
# Make sure to use different filenames for each object, such as the object
# name itself.
#
```

```
# Remember to rerun this code to save new results when you change the code.
#
# The .gitignore file has been setup so that it ignores .rds files in the data/
# folder, so that you don't accidentally make git handle large binary data files.
```