Statistical Computing: Coursework A

James Wright (S1604483)

Preamble

```
#Importing datasets and required libraries for plotting
TMINallobs <- read.csv(file = "~\\TMINallobs.csv",
                       header = TRUE,
                       stringsAsFactors = FALSE)
TMINalltest <- read.csv(file = "~\\TMINalltest.csv",
                        header = TRUE,
                        stringsAsFactors = FALSE)
TMINoneobs <- read.csv(file = "~\\TMINoneobs.csv",
                       header = TRUE,
                       stringsAsFactors = FALSE)
TMINonetest <- read.csv(file = "~\\TMINonetest.csv",
                        header = TRUE,
                        stringsAsFactors = FALSE)

library(ggplot2) #Plotting library
library(xtable) #LaTeX table library
#Map libraries
library(ggmap)
library(maps)
library(mapdata)
```

I am defining the function that computes the Brier scores required in question 4.2 early so it can be stored in the data frames during other questions, for efficiency.

```
# score_brier: Compute predictions from model for given data
# Input:
#   pred : data.frame of means, prediction standard deviations,
#          and prediction probabilities
#   y : vector
# Output:
#   a vector of Brier scores

score_brier <- function(pred, y){
  indicator <- as.numeric(I(y <= 0))
  brier_score <- (indicator - pred$prob)^2
  return (brier_score)}
```

We also source an external file that contains the SE, DS and mean scoring functions.

```
#External file containing scoring functions
source("C:\\Users\\wrigh\\OneDrive\\Desktop\\StatCompCWA\\CWA2019code.R")
```

Question 1

1.1

Below we estimate a naive non-seasonal model for minimum temperature 'jointly for all five stations' using the data set of observations for all stations. We consider an intercept, longitude, latitude, elevation and (re-centered) temporal trend, and assume linearity and continuity for these covariates.

```
#Estimating naive model
model_naive <- lm(Value ~ Longitude + Latitude + Elevation + I(DecYear - 2000),
                  data = TMINallobs)
```

```
#Displaying coefficient summary for naive model as table
coefficients_naive <- summary(model_naive)$coefficients
print(xtable(coefficients_naive[, c(1, 4)], digits = c(0, 4, -3),
             caption = "Coefficients for Naive Model",
             label = "ncoef"), type = "latex", comment = FALSE)
```

	Estimate	Pr(>|t|)
(Intercept)	37.0084	2.324E-44
Longitude	-0.1453	1.541E-10
Latitude	-0.5672	2.844E-33
Elevation	-0.0077	0.000E+00
I(DecYear - 2000)	0.0181	1.145E-77

Table 1: Coefficients for Naive Model

Throughout I will generally discuss the significance of the p-values within the typical significant-level 'star' categories of the "summary()" function, even though many are clearly significant far beyond a 0.1% *level.*

Consider Table **1**. We make note that all of these coefficients are significant at the 0.1% level, as all the p-values are less than 0.001. This means that there is sufficient evidence to suggest that each of the covariates are related enough to minimum temperature to improve the accuracy of our estimates of the response variable, and consequently we can assume all of these terms are worth keeping in the model. We also note an r-squared value of 0.04 implying that his model is a poor fit for our observed data training set.

```
#Printing r-squared value
summary(model_naive)$r.squared
```

```
## [1] 0.04021611
```

An important feature to note about this model is that the intercept value is located at 0 elevation above mean sea level, 0 longitude and 0 latitude, and taken to be the beginning of 1960; that is, this intercept is taken to be that it was 37°C at Null Island at the beginning of 1960. As the data is exclusively from Scotland it is most likely an erroneous estimate as it is unlikely Scottish weather patterns extend perfectly linearly to elsewhere in the world. However, for our purposes this observation is somewhat redundant as we are exclusively interested in predicting the minimum temperatures for the Scottish stations - which are local to this data set. For this experiment we will assume Scotland is a sufficiently small region of the planet for this model to act as a suitable linear approximation to a more global model (which may or may not be linear).

With the above said, we can interpret the coefficients of this model in the following way:

1. The intercept estimate tells us that at mean sea-level, 0 degrees longitude, and 0 degrees latitude you would expect the minimum temperature to have been approximately +37°C at the beginning of 1960.
2. The longitude coefficient tells us that a +1 degree change in longitude you would expect an approximate −0.145°C change in the minimum temperature.
3. The latitude coefficient tells us that a +1 degree change in latitude you would expect an approximate −0.567°C change in the minimum temperature.

4. The elevation coefficient tells us that every +100 metre change in elevation you would expect an approximate $-0.77°$C change in the minimum temperature.
5. The temporal trend coefficient "I(DecYear-2000)" tells us that every 10 years you would expect an approximate $+0.18°$C rise in the minimum temperature globally.

1.2

We now construct a prediction function that produces a data frame of the point estimates "mu", the prediction standard deviations "sigma" and prediction probabilities for $\{Y_i \leq 0\}$ "prob" for a given linear model and data set. We make the following observations and assumptions:

- Our point estimates will lie along the model line and correspond to the parameters of each data point in the test data set.
- By assuming independence and the resulting linearity of variance, we derive that the data-level prediction standard deviations are given by

$$\text{PSD} = \sqrt{(\text{PPS})^2 + (\text{ESDR})^2}.$$

  Where:

  - PSD is the data level prediction standard deviations.
  - PPS is the prediction standard deviations for the point predictions.
  - ESDR is the estimated standard deviations of the residuals.

- We take the distribution of our predictions to be $Y_i \sim N(\mu_i, \sigma_i^2)$.

```r
# prediction: Compute predictions from model for given data
# Input:
#   fit : linear model ~ lm()
#   newdata : data.frame
# Output:
#   a data.frame of point estimates, standard deviations, prediction probabilities

prediction <- function(fit, newdata){
  p <- predict.lm(fit, newdata, interval = "prediction", se.fit = TRUE) #Predicting from model
  mu <- p$fit[, "fit"] #Point estimates
  sigma <- (sigma(fit)^2 + p$se.fit^2)^0.5 #Prediction standard deviations
  prob <- pnorm(q = 0, mean = mu, sd = sigma) #Prediction probabilites for {Y_i <= 0}
  data_frame <- data.frame(mu, sigma, prob, row.names = NULL)
  return (data_frame)
}
```

1.3

To test our naive model, we use the all stations test data set to make predictions using the prediction function created above at both an overall level and station-by-station. From this, we compute the scores using the functions sourced and defined in the preamble. These are the Squared Error Scores, Dawid-Sebastiani scores (and for later use Brier Scores) - from hereon referred to as SE and DS (and Brier) respectively. We also make note that these scores are all negatively orientated so lower scores are better.

```
#Predicting from Naive model using TMINalltest data
pred_naive <- prediction(model_naive, TMINalltest)

#Storing scores by locaility in data frame for naive model and calculating their means
naivescore_local <- data.frame(ID = TMINalltest$ID, #By ID
                               SE = score_se(pred_naive, TMINalltest$Value),
                               DS = score_ds(pred_naive, TMINalltest$Value),
                               Brier = score_brier(pred_naive, TMINalltest$Value))

naivemean_local <- mean_score(naivescore_local, by.ID = TRUE)

#Storing scores in data frame for naive model and calculating their means
naivescore_overall <- data.frame(SE = score_se(pred_naive, TMINalltest$Value),
                                 DS = score_ds(pred_naive, TMINalltest$Value),
                                 Brier = score_brier(pred_naive, TMINalltest$Value))

naivemean_overall <- mean_score(naivescore_overall, by.ID = FALSE)
```

We obtain the following overall average SE and DS scores for the naive model:

```
#Displaying naive overall scores as table
print(xtable(naivemean_overall[, 1:2],
             caption = "Average Overall Scores for the Naive Model",
             label = "nosc"), type = "latex", comment = FALSE)
```

	SE	DS
1	22.92	4.13

Table 2: Average Overall Scores for the Naive Model

We also obtain the following average SE and DS scores from each station for the naive model:

```
#Displaying naive local scores as table
print(xtable(naivemean_local[, 1:3],
             caption = "Average Scores for the Naive Model by Station",
             label = "nlsc"), type = "latex", comment = FALSE)
```

	ID	SE	DS
1	UKE00105875	26.31	4.27
2	UKE00105884	21.06	4.06
3	UKE00105886	22.40	4.11
4	UKE00105887	22.78	4.13
5	UKE00105888	23.10	4.14
6	UKE00105930	21.87	4.09

Table 3: Average Scores for the Naive Model by Station

Before further discussion of these values in question 2.2, an immediate observation that can be made is that the SE are lower than DS scores both station-by-station and overall. Recall that the SE score only considers the predictive expectation, while the DS score also takes into account uncertainty. By comparing the SE scores in Table **3** with the average in Table **2**, we can deduce that the naive model is noticeably better than average at making point predictions for the stations UKE00105884 and UKE00105930, and likewise noticeably worse at making point predictions for the station UKE00105875. Similarly, by comparing the station DS scores with the overall average, we can see that the naive model has similar certainty about the accuracy of its predictions for all stations - with the exception of UKE00105875 for which it is noticeably less certain.

We have the following range for minimum temperatures for our all stations test data:

```
range(TMINalltest$Value)
```

```
## [1] -17.7  17.8
```

Observe that the square root of the overall average squared error value is approximately 4.8°C - in the context of temperature prediction for Scotland this is a severe level of error. We see that the mean minimum temperature in our test data set is 4.63°C - an error size in our predictions exceeding the average of the data values in many ways renders these temperature predictions useless as the error is large in proportion to our possible range of values.

Question 2

2.1

Below we extend our previous model to account for seasonality by using a truncated Fourier series of order 2. This is because the sine and cosine functions should estimate the fluctuating periodic nature of the seasons that you would expect in real world weather patterns. In doing so, we have implicitly assumed that seasonal fluctuations will be similar enough across Scotland for this to make suitable predictions. I chose to use an order 2, as if we chose Fourier series of order 1 the frequencies of the terms would not be sufficient to match that of the seasons as they can each only take positive and negative values over each half of a cycle; the more frequent terms $\cos(4\pi x)$ and $\sin(4\pi x)$ should allow us to account for any of the more subtle variations within the seasons themselves. We do not choose a higher order as this risks over-fitting the model. This model is again assuming linearity of the covariates and is being estimated using the observed data for all stations.

```
#Estimating seasonal model
model_f <- lm(Value ~ Longitude + Latitude + Elevation + I(DecYear - 2000) #Previous model
              + I(cos(2*pi*DecYear)) + I(sin(2*pi*DecYear)) #Fourier series
              + I(cos(4*pi*DecYear)) + I(sin(4*pi*DecYear)), #Fourier series
              data =  TMINallobs)
```

We obtain the following estimated coefficients for our model:

```
#Displaying seasonal model coefficient summary data as table
coefficients_f <- summary(model_f)$coefficients
print(xtable(coefficients_f[, c(1, 4)], digits = c(0, 4, -4),
             caption = "Coefficients for Seasonal Model",
             label = "fcoef"), type = "latex", comment = FALSE)
```

	Estimate	Pr(>|t|)
(Intercept)	36.8905	1.5946E-104
Longitude	-0.1453	1.7759E-23
Latitude	-0.5672	2.3603E-78
Elevation	-0.0077	0.0000E+00
I(DecYear - 2000)	0.0181	7.2278E-186
I(cos(2 * pi * DecYear))	-4.7959	0.0000E+00
I(sin(2 * pi * DecYear))	-2.2989	0.0000E+00
I(cos(4 * pi * DecYear))	0.1547	1.2238E-26
I(sin(4 * pi * DecYear))	0.6036	0.0000E+00

Table 4: Coefficients for Seasonal Model

We again make a note that all of these coefficients are significant at the 0.1% level, as all p-values are less than 0.001. It follows that we have sufficient evidence to believe that all of these covariates contribute to the accuracy of our response variable predictions. Notice further that the coefficients corresponding to the covariates from the original naive model see no significant change, and also note that this model faces the same extrapolation drawbacks for the intercept as mentioned for the naive model in 1.1. However, we have now also discovered that in our model the sinusoidal covariates are a highly significant predictor of minimum temperature, and so should improve our model. We also note an r-squared value of 0.606 implying that his model is a much better fit for our observed data training set than the naive model was.

```
#Printing r-squared value
summary(model_f)$r.squared
```

```
## [1] 0.6056635
```

Obviously, the coefficients of the covariates from the naive model have the same interpretations as before in regards to their effect on the response variable - as they retain the same values. We can interpret our Fourier

coefficients of the same frequency together as the sum will simply be some corresponding scalar multiple and phase shift resulting from the form of the trigonometric identity $\cos(x) + \sin(x) = \sqrt{2}\sin(\pi/4 + x)$.

Given this, we deduce the following from Table **4**:

1. The lower frequency - "I(cos(2 * pi * DecYear))" and "I(sin(2 * pi * DecYear))" - coefficients are relatively large compared to the other covariates, with values of -4.7959 and -2.2989 respectively. These imply that the less frequent aspect of seasonality has a large amplitude - that is, the difference in temperature in each period is more pronounced and should have a large impact on our expectation of the minimum temperature. This matches our intuition reasonably well - you would expect the summer and winter average minimum temperatures to each deviate from the year average somewhere around 5°C and -5°C respectively.

2. The higher frequency - "I(cos(4 * pi * DecYear))" and "I(sin(4 * pi * DecYear))" - coefficients are $+0.1514$ and $+0.6036$. These suggest the more frequently oscillating aspect of seasonality are of smaller amplitude - meaning the "intra-season" periodic behaviour is less pronounced - changing the minimum average temperature by around ± 0.6°C within each half-cycle.

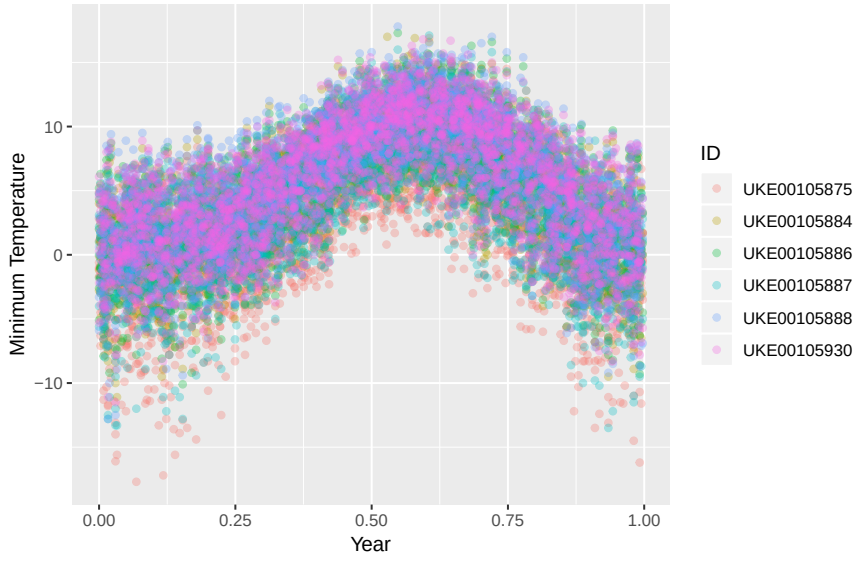
2.2

Again, to test this seasonal model we make predictions using the prediction function both at an overall level and by station, using the all stations test data set. We compute the scores using the functions sourced / defined in the preamble.

```
#Predicting from seasonal model using TMINalltest data
pred_f <- prediction(model_f, TMINalltest)

#Storing scores by locality in data frame for seasonal model and calculating their means
fourierscore_1ocal <- data.frame(ID = TMINalltest$ID, #By ID
                                 SE = score_se(pred_f, TMINalltest$Value),
                                 DS = score_ds(pred_f, TMINalltest$Value),
                                 Brier = score_brier(pred_f, TMINalltest$Value))

fouriermean_local <- mean_score(fourierscore_1ocal, by.ID = TRUE)

#Storing scores in data frame for seasonal model and calculating their means
fourierscore_overall <- data.frame(SE = score_se(pred_f, TMINalltest$Value),
                                   DS = score_ds(pred_f, TMINalltest$Value),
                                   Brier = score_brier(pred_naive, TMINalltest$Value))

fouriermean_overall <- mean_score(fourierscore_overall, by.ID = FALSE)
```

```
#Plotting minimum temperature test data against season
ggplot(data=TMINalltest) +
  geom_point(aes(x = DecYear - floor(DecYear), y = Value, col = ID), alpha = 0.3) +
  ggtitle("Figure 1: Plot of Minimum Temperatures Against Season") +
  xlab("Year") + ylab("Minimum Temperature") + theme(plot.title = element_text(hjust = 0.5))
```

Figure 1: Plot of Minimum Temperatures Against Season



We have the following overall average SE and DS scores from the seasonal model:

```
#Displaying overall scores as table for seasonal model
print(xtable(fouriermean_overall[, 1:2],
             caption = "Average Overall Scores for the Seasonal Model",
             label = "fosc"), type = "latex", comment = FALSE)
```

	SE	DS
1	9.51	3.25

Table 5: Average Overall Scores for the Seasonal Model

Recalling the average overall score results from Table **2** in 1.3, we see that the average squared error is smaller for the seasonal model than the naive model, with $9.51 < 22.92$. We also have that the average Dawid-Sebastiani score is smaller for the seasonal model than the naive model, with $3.25 < 4.13$. As these are negatively orientated, we can conclude that the seasonal model outperforms the naive model for the overall data according to both scoring methods - while noting that SE only cares about the predictive expectation while DS also takes into account the uncertainty of these predictions. Therefore, on average the seasonal model is better overall both from a point prediction perspective and from an uncertainty of prediction perspective. Looking at Figure 1, this is to be expected as the complete test data set (ignoring colour) clearly shows some sinusoidal behaviour. As alluded to in 2.1, all of the coefficients for all other covariates are almost the same, but the naive linear model does not account for this periodicity. Meanwhile, the model with the Fourier series is able to correct for this feature leading to smaller distances between predictions and true values for many data points - hence smaller scores.

Recall the results from Table **3**, and observe we have the following average SE and DS scores by station from the seasonal model:

```
#Displaying local scores as table for seasonal model
print(xtable(fouriermean_local[, 1:3],
             caption = "Average Scores for the Seasonal Model by Station",
             label = "flsc"), type = "latex", comment = FALSE)
```

	ID	SE	DS
1	UKE00105875	12.47	3.55
2	UKE00105884	8.05	3.11
3	UKE00105886	8.56	3.16
4	UKE00105887	9.91	3.29
5	UKE00105888	8.85	3.19
6	UKE00105930	9.19	3.22

Table 6: Average Scores for the Seasonal Model by Station

We see that station-by-station the average squared error scores are smaller for the seasonal model than their counterpart in the naive model. This is also the case with the Dawid-Sebastiani scores. As these scores are negatively orientated, we can conclude that the seasonal model's forecasts outperform those of the naive model for each station according to both scoring methods - that is, the seasonal model is both better from a point prediction standpoint and when taking into consideration of the uncertainty of the predictions station-by-station. As before, by considering Figure 1 we see that the sinusoidal nature appears to occur across all stations (considering colour) - which the naive model fails to account for while the seasonal model does; explaining the difference in the size of the scores.

2.3

Recall that for both for the naive model (Table **1**) and seasonal model (Table **4**) we had the following coefficients for the covariates elevation and temporal trend, and that for both models their covariates were significant at the 0.01% level as they had a p-value of less than 0.001, so there was sufficient evidence to suggest that they contribute to the performance of the estimate of the response variable:

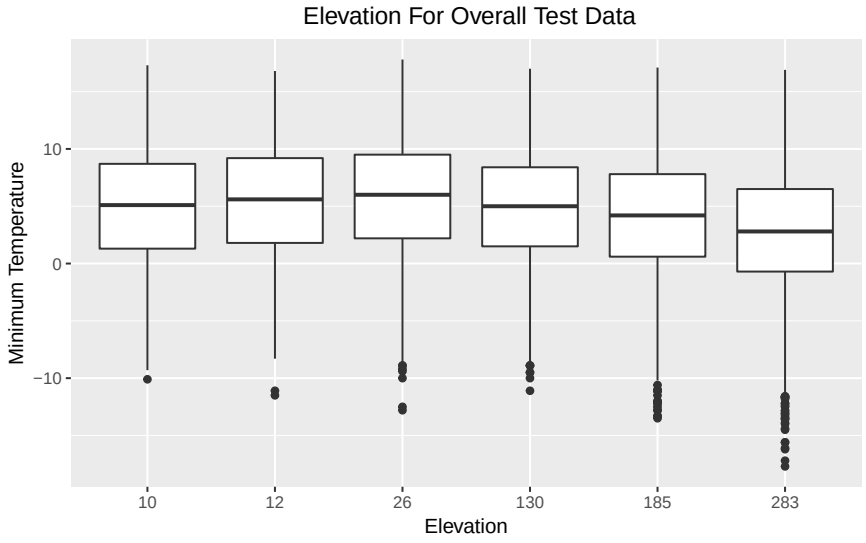
```
#Displaying coefficients for elevation and temporal trend (either model)
print(xtable(coefficients_naive[, c(1, 4)], digits = c(0, 4, 4),
             caption = "Coefficients for the Naive and Seasonal Model")[4:5, ],
             type = "latex", comment = FALSE)
```

	Estimate	Pr(>|t|)
Elevation	-0.0077	0.0000
I(DecYear - 2000)	0.0181	0.0000

Table 7: Coefficients for the Naive and Seasonal Model

```
Elev <- as.factor(TMINalltest$Elevation) #Converting elevations to factor for plotting

#Box plotting temperature data against elevation
ggplot(data = NULL) +
  geom_boxplot(aes(x = Elev, y = TMINalltest$Value)) +
  ggtitle("Figure 2: Plot of Average Minimum Temperatures Against
          \n Elevation For Overall Test Data") +
  xlab("Elevation") + ylab("Minimum Temperature") +
  theme(plot.title = element_text(hjust = 0.5))
```

Figure 2: Plot of Average Minimum Temperatures Against



As stated in 1.3, we see that within both models the coefficient for elevation is approximately -0.0077. It tells us that a $+100$ metre increase above mean sea level leads to an expected $-0.77°$C change in the minimum temperature.

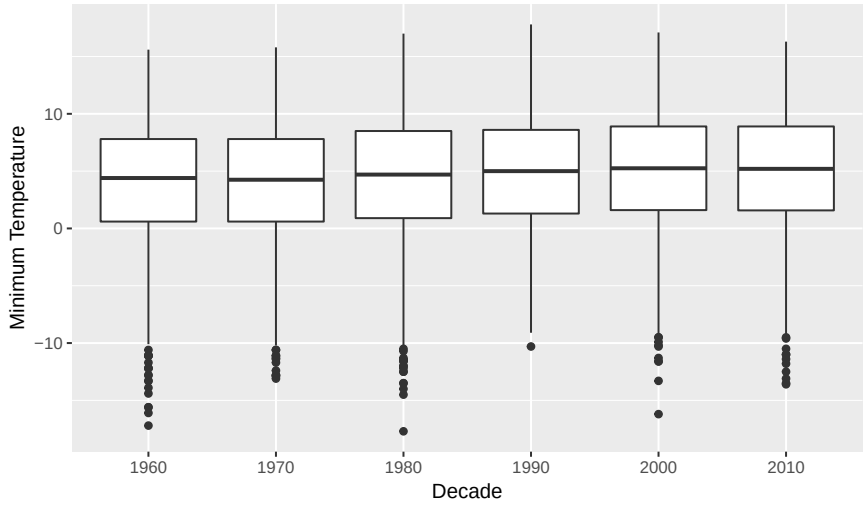
Looking at the means of the box plots in Figure 2 above, we see by the average minimum temperature means that a $-0.77°$C change in minimum temperature for a 100 metre change in elevation is a reasonable estimate at higher elevations. However, for the stations at low elevations the average minimum temperatures seem to be rising with elevation - this is possibly due to the small differences in their elevations, in conjunction

with a small coefficient value, enabling the resulting change in the response variable to be easily cancelled out by differences in some of the other covariates. Ceteris paribus, this implies that when considering small changes in elevation this covariate is potentially less useful for temperature predictions as an individual estimator. Additionally, with rising elevations there appears to be an increasing number of outliers for colder temperatures which could suggest a non-linear relationship between elevation and temperature change - which calls into question our earlier assumption of linearity for this covariate.

```
#Turning decade into factor for plotting
decades <- floor(TMINalltest$Year/10)*10 #Removing 'ones' digit
decades <- as.factor(decades)

#Box plotting temperature data against decade
ggplot(data = NULL) +
  geom_boxplot(aes(x = decades, y = TMINalltest$Value)) +
  ggtitle("Figure 3: Plot of Temperatures Against
          \n Average Minimum Temperatures Per Year") +
  xlab("Decade") + ylab("Minimum Temperature") +
  theme(plot.title = element_text(hjust = 0.5))
```

Figure 3: Plot of Temperatures Against

Average Minimum Temperatures Per Year



We also saw that within both models the coefficient for temporal trend was 0.0181 The temporal trend coefficient implies that the average change in minimum temperature in Scotland per decade is approximately $+0.18°C$. Looking at the averages of the box plot in Figure 3 we see this is a reasonable estimate as the average temperature does seem to be rising approximately at this rate.

Question 3

3.1

We now estimate a model for the UKE00105875 station, using an intercept, temporal trend, seasonal oscillations via a truncated Fourier series, and then add the remaining stations as covariates. Let us keep the order of the Fourier series as 2 to aid our discussions - if the seasons are stable across Scotland then a model for a specific station should not see much difference in regards to these coefficients than the joint seasonal model - and that fact would in turn help support a case for the jointly seasonal model. We train this model using the one station observations data set.

```
#Estimating station model
model_st <- lm(UKE00105875 ~ I(DecYear - 2000) #Temporal trend
               + I(cos(2*pi*DecYear)) + I(sin(2*pi*DecYear)) #Fourier series
               + I(cos(4*pi*DecYear)) + I(sin(4*pi*DecYear)) #Fourier series
               + UKE00105884 + UKE00105886 + UKE00105887 + UKE00105888 + UKE00105930, #Stations
               data = TMINoneobs)
```

We obtain the following coefficients for our station model:

```
#Displaying coefficient summary data for station model as table
coefficients_st <- summary(model_st)$coefficients[, c(1, 4)]
print(xtable(coefficients_st, digits = c(0, 4, -4),
             caption = "Coefficients for UKE00105874 Station Model of Order 2",
             label = "stcoef"), type = "latex", comment = FALSE)
```

	Estimate	Pr(>|t|)
(Intercept)	-2.6031	0.0000E+00
I(DecYear - 2000)	0.0038	2.3043E-05
I(cos(2 * pi * DecYear))	0.4646	6.0897E-45
I(sin(2 * pi * DecYear))	0.2974	1.3990E-35
I(cos(4 * pi * DecYear))	-0.0836	5.5800E-05
I(sin(4 * pi * DecYear))	-0.0631	2.4200E-03
UKE00105884	0.5901	0.0000E+00
UKE00105886	0.4193	8.4417E-292
UKE00105887	0.2165	1.5470E-70
UKE00105888	-0.0456	1.7198E-03
UKE00105930	-0.0552	3.4537E-08

Table 8: Coefficients for UKE00105874 Station Model of Order 2

An interesting observation here is that the sinusoidal terms of order 2 have relatively large p-values compared to those in the seasonal model model and in comparison to the terms of order 1 in the model. More specifically, the $\sin(4\pi x)$ term is 'only' significant at the 1% level so higher order terms are less associated with the response variable than the other covariates. This also suggests our assumption in 2.1 - that the behaviour of seasonality is similar enough across Scotland to estimate all models using the same amplitudes - is in fact incorrect. Additionally, we observe that one more covariate, the station UKE00105888, is also 'only' significant at the 1% level, so is again a less significant estimator of the minimum temperature at this station. We also observe that this model has an r-squared value of 0.88 making it a good fit for our observed data training set.

```
#Printing r-squared value
summary(model_st)$r.squared
```

```
## [1] 0.8812376
```

A final noticeable change in this model is that the intercept value is approximately $-2.60°C$. This is as a consequence of the fact that we are not considering this model as a linear approximation of a more global

model (considering elevation, longitude and latitude), but are implicitly only defining our model within the vicinity of where the data is recorded (Scotland); we have not had to extrapolate the temperatures to the equator in order to estimate our intercept.

As before we can interpret our Fourier coefficients of the same frequency together as the sum will simply be some corresponding scalar multiple and phase shift resulting from the form of the trigonometric identity $\cos(x) + \sin(x) = \sqrt{(2)}\sin(\pi/4 + x)$. With the above said, we can interpret the coefficients of this model in the following way with reference to Table **8**:

1. The intercept estimate tells us that you would expect the minimum temperature of the station UKE00105875 to have been approximately $-2.6031°$C at the beginning of 1960, if all the covariate stations were 0°C.

2. The UKE00105884 coefficient tells us that a $+1°$C change in the minimum temperature at the station UKE00105884 you would expect an approximate $+0.5901°$C change in the average minimum temperature and the UKE00105886 coefficient tells us that a $+1°$C change in the minimum temperature at the station UKE00105886 you would expect an approximate $+0.4193°$C change in the average minimum temperature. These coefficients tell us that you would expect the minimum temperatures at these stations to be highly related with what temperatures you'd expect at UKE00105875.

3. The UKE00105887 coefficient tells us that every $+1°$C change in the minimum temperature at the station UKE00105887 you would expect an approximate $+0.2165°$C change in the average minimum temperature. This tells us that the temperature of this station is somewhat less important in regards to our expected minimum temperature at UKE00105875, but still contributes to the efficacy of our prediction.

4. The UKE00105888 coefficient tells us that every $+1°$C change in the minimum temperature at the station UKE00105888 you would expect an approximate $-0.0456°$C change in the average minimum temperature and the UKE00105930 coefficient tells us that every $+1°$C change in the minimum temperature at the station UKE00105930 you would expect an approximate $-0.0552°$C change in the average minimum temperature. The small values of these coefficients imply that the minimum temperatures of these stations should have very little influence on what we would predict as the minimum temperatures for UKE00105875.

5. The temporal trend coefficient "I(DecYear-2000)" tells us that every 10 years you would expect an approximate $+0.038°$C rise in the average minimum temperature at the station UKE00105875. That is, for this station in particular the temperature seems to be rising at a lower rate per decade than elsewhere, by comparing with the corresponding coefficient of the seasonal model.

6. The lower frequency [I(cos(2 * pi * DecYear)) and I(sin(2 * pi * DecYear))] coefficients are relatively large compared to the other covariates, with values of 0.4646 and 0.2974 respectively. These imply that the less frequent aspect of seasonality has a big impact on our predictions for the minimum temperature for this station - although it is smaller than that of the more general 'joint' model discussed earlier. In other words, the difference in temperature in each period is less than the other model and its impact on our expectation of the minimum temperature in different seasons is less severe, but still highly significant, with an amplitude of around 0.5°C. This suggests that at this station the seasonal differences are not as severe if we compare with the corresponding coefficients in the 'joint' seasonal model.

7. The higher frequency [I(cos(4 * pi * DecYear)) and I(sin(4 * pi * DecYear))] coefficients are -0.0836 and -0.0631. These suggest the more frequently oscillating aspect of seasonality should have very little impact on our predictions of the minimum temperature at this station with an amplitude of almost zero - i.e. there is very little variability 'within' the broader seasons for UKE00105875.

3.2

As in previous questions, we now wish to obtain predictions for the stations model using the one station test data set using our previously constructed prediction function. We then evaluate the model using the defined scoring functions.

```
#Predicting from stations model using TMINonetest data
pred_st <- prediction(model_st, TMINonetest)

#Storing scores in data frame for stations model and calculating their mean
SVscore_station <- data.frame(SE = score_se(pred_st, TMINonetest$UKE00105875),
                              DS = score_ds(pred_st, TMINonetest$UKE00105875),
                              Brier = score_brier(pred_st, TMINonetest$UKE00105875))

SVmean_station <- mean_score(SVscore_station, by.ID = FALSE)
```

We obtain the following average SE and DS scores for this model of the station:

```
#Displaying scores as table for station model
print(xtable(SVmean_station[, 1:2],
             caption = "Average Scores of the UK00105875 Station Specfic Model",
             label = "stsc"),
            type = "latex", comment = FALSE)
```

	SE	DS
1	3.28	2.19

Table 9: Average Scores of the UK00105875 Station Specfic Model

I also present the SE and DS score data from the naive and seasonal models for this station to aid comparison:

```
#Displaying scores for UKE00105875 as table for naive model
nm_l <- filter(naivemean_local, ID == "UKE00105875")[, 1:3]
print(xtable(nm_l,
             caption = "Average Scores Under the Naive Model for UK00105875 Station"),
             type = "latex", comment = FALSE)
```

	ID	SE	DS
1	UKE00105875	26.31	4.27

Table 10: Average Scores Under the Naive Model for UK00105875 Station

```
#Displaying scores for UKE00105875 as table for seasonal model
fm_l <- filter(fouriermean_local, ID == "UKE00105875")[, 1:3]
print(xtable(fm_l,
             caption = "Average Scores Under the Seasonal Model for UK00105875 Station"),
             type = "latex", comment = FALSE)
```

	ID	SE	DS
1	UKE00105875	12.47	3.55

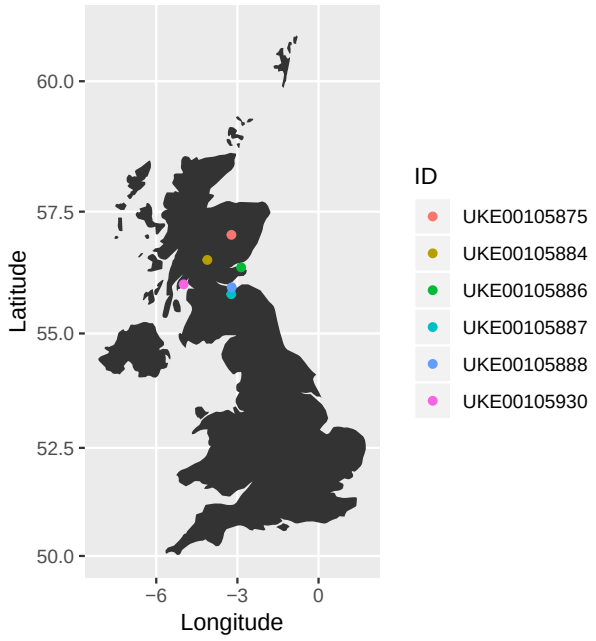
Table 11: Average Scores Under the Seasonal Model for UK00105875 Station

We see that the average SE score for the station model is lower than those of both the naive and seasonal model, with a value of 3.28 compared to 26.31 and 12.41 respectively. We also see that the DS score for the station model is lower than those of both the naive and seasonal model, with a value of 2.19 compared to 4.27 and 3.55 respectively. By negative orientation, it follows that this model outperforms both the naive and seasonal model based on these scores. In particular, the significantly lower SE score tells us that this station specific model is far more accurate at making point predictions than the 'joint' prediction models described earlier. Also, the far lower DS score also implies that this model is much more certain about the predictions it is making than the other models as there appears to be no obvious linear relationship.

It is worth recalling that for most of the stations in both 'joint' models their average SE and DS scores are reasonably close to or below the respective overall average score, but UKE00105875 is an exception to this - as commented on in 1.3 for the naive case. Looking at the seasonal model, it has an overall average SE score 9.51 and an overall average DS score of 3.25, which are significantly smaller than the respective local averages for UKE00105875 of 12.47 and 3.55 respectively. These comments suggest that although the seasonal model is much better than the naive model, these 'joint' prediction models faired worse at predicting this particular station.

```
#Extracting station position data
places <- data.frame(ID = unique(TMINallobs$ID),
                     Longitude = unique(TMINallobs$Longitude),
                     Latitude = unique(TMINallobs$Latitude),
                     Elevation = unique(TMINallobs$Elevation))

#Plotting positions of stations on map
UK <- map_data(map = "world", region = "UK")
ggplot(data = UK) +
  geom_polygon(aes(x = long, y = lat, group = group)) +
  coord_map()+
  geom_point(data = places, aes(x = Longitude, y = Latitude, col = ID)) +
  ggtitle("Figure 4: Station Locations") +
  xlab("Longitude") + ylab("Latitude") +
  theme(plot.title = element_text(hjust = 0.5))
```

Figure 4: Station Locations



```
#Displaying locations as table
print(xtable(places,
             caption = "Geographic Locations of Stations",
             label = "places"),
             type = "latex", comment = FALSE)
```

	ID	Longitude	Latitude	Elevation
1	UKE00105875	-3.22	57.04	283
2	UKE00105884	-4.11	56.53	130
3	UKE00105886	-2.86	56.38	10
4	UKE00105887	-3.23	55.82	185
5	UKE00105888	-3.21	55.97	26
6	UKE00105930	-4.99	56.03	12

Table 12: Geographic Locations of Stations

Looking at the significance of the coefficients for the station model in Table **8** we see that the stations UKE00105884, UKE00105886 have much smaller p-values compared to the other three stations. By plotting all of the positions of these stations on a map [2], we see that these two stations are in fact the closest to UK00105875. However, the station with the third-most significant p-value, UKE00105887, is the furthest station from UK00105875. Also, while UKE00105884 and UKE00105887 are on high elevations, we see that UKE00105886 is on the lowest of all elevations. Based on this discussion, the elevations, longitudes and latitudes are possibly not as apt predictors of the minimum temperature at this specific station as assumed by the other models.

It is also now worth recalling from Figure 2 that the sinusoidal behaviour has different amplitudes between different stations, with UKE00105875 clearly lying below the general shape of the rest of the stations, which is reflected by its larger SE and DS scores in Tables **3** and **6** under the joint model. As discussed with the Fourier coefficients in 3.1 above, the assumption that the variability of seasons is identical across Scotland is also clearly wrong.

It is clear that this model performs better at predictions as during estimation it has the freedom to weight its predictions towards stations that are better predictors for UKE00106875 in order to maximise the fit for that *particular* station. On the other hand, the seasonal and naive models are being designed to fit a more general model that instead aims to most accurately predict *across* Scotland - and so the coefficients are being chosen in order to maximise the accuracy for all stations - probably at the expense of accuracy at particular stations. An example of this is that the station specific model was able to choose coefficients for seasonality that matched its own temperature data, rather than having to accommodate the irrelevant seasonal patterns in other areas of the country such as the west coast which is affected by the Gulf stream. Additionally, this model may contain more features that are important characteristics for predictions at this particular station implicitly within what causes the minimum temperatures at these - while the less significant stations have different characteristics - such as air pressure, distance from water and so on.

Question 4

4.1

We make note of the following theorems:

$$[\mathbb{I}(y \leq 0)]^2 = \mathbb{I}(y \leq 0) = \begin{cases} 1 & y \leq 0. \\ 0 & \text{Otherwise.} \end{cases} \tag{1}$$

and by the proper scoring rules notes we have [1]

$$E[\mathbb{I}(y \leq 0)] = \mathbb{P}(y \leq 0) = G(0). \tag{2}$$

Note that the Brier score is negatively orientated, so proper if and only if it fulfills the condition

$$S(F, G) \geq S(G, G).$$

Proof.

We have by the definition of expectation of scores and the definition of a Brier score:

$$\begin{aligned}
S(F,G) &= E_{y\sim G}[S(F,y)] \\
&= E_{y\sim G}\left[(\mathbb{I}(y \leq 0) + F(0))^2\right] \\
&= E_{y\sim G}\left[[\mathbb{I}(y \leq 0)]^2 - 2F(0)\mathbb{I}(y \leq 0) + F(0)^2\right] \\
&= E_{y\sim G}\left[\mathbb{I}(y \leq 0) - 2F(0)\mathbb{I}(y \leq 0) + F(0)^2\right] && \text{By (1)} \\
&= E_{y\sim G}\left[\mathbb{I}(y \leq 0)\right] - 2E_{y\sim G}\left[F(0)\mathbb{I}(y \leq 0)\right] + E_{y\sim G}\left[F(0)^2\right] && \text{By Linearity of Expectation}
\end{aligned}$$

Then using the fact that F(0) is just a constant value under a cumulative distribution:

$$\begin{aligned}
S(F,G) &= E_{y\sim G}\left[\mathbb{I}(y \leq 0)\right] - 2F(0)E_{y\sim G}\left[\mathbb{I}(y \leq 0)\right] + F(0)^2 \\
&= G(0) - 2F(0)G(0) + F(0)^2 && \text{By (2)} \\
&= G(0) + G(0)^2 - 2F(0)G(0) + F(0)^2 - G(0)^2 \\
&= G(0) + [G(0) - F(0)]^2 - G(0)^2 \\
&\geq G(0) - G(0)^2 \text{ as this is minimised where G(0) = F(0).}
\end{aligned}$$

Now we observe:

$$\begin{aligned}
S(G,G) &= E_{y\sim G}\left[(\mathbb{I}(y \leq 0) - G(0))^2\right] \\
&= E_{y\sim G}\left[[\mathbb{I}(y \leq 0)]^2 - 2\mathbb{I}(y \leq 0)G(0) + G(0)^2\right] \\
&= E_{y\sim G}\left[[\mathbb{I}(y \leq 0)] - 2\mathbb{I}(y \leq 0)G(0) + G(0)^2\right] && \text{By (1)} \\
&= E_{y\sim G}\left[[\mathbb{I}(y \leq 0)] - 2G(0)E_{y\sim G}\left[\mathbb{I}(y \leq 0)\right] + G(0)^2\right. && \text{By Linearity of Expectation} \\
&= G(0) - 2G(0)G(0)\left[\mathbb{I}(y \leq 0)\right] + G(0)^2 && \text{By (2)} \\
&= G(0) - G(0)^2
\end{aligned}$$

Hence,

$$S(F, G) \geq S(G, G) \text{ as required.}$$

□

4.2

For completeness / the marker I have included this function in its appropriate question part, although it is used prior to this in previous questions.

```
# score_brier: Compute predictions from model for given data
# Input:
#   pred : data.frame of means, prediction standard deviations, and prediction probabilities
#   y : data.frame
# Output:
#   a vector of Brier scores
score_brier <- function(pred, y){
   indicator <- I(y <= 0)
   brier_score <- (indicator - pred$prob)^2
   return (brier_score)
}
```

4.3

We now extract the Brier scores from our previously created data frames.

```
#Extracting Brier scores from each data frame
bs_naive <- filter(naivemean_local, ID == "UKE00105875")[, 4]
bs_f <- filter(fouriermean_local, ID == "UKE00105875")[, 4]
bs_st <- SVmean_station[, 3]

brier_output <- data.frame(Naive = bs_naive,
                           Seasonal = bs_f,
                           Station = bs_st)
```

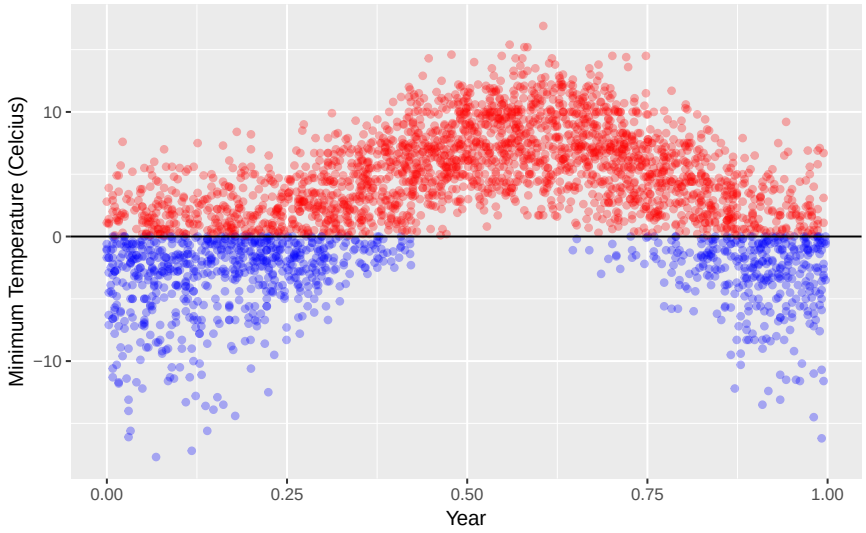
```
#Displaying Brier scores as table
print(xtable(brier_output,
             caption = "Brier Scores for Each Model - Station UKE00105875",
             label = "brier"),
             type = "latex", comment=FALSE)
```

	Naive	Seasonal	Station
1	0.22	0.14	0.07

Table 13: Brier Scores for Each Model - Station UKE00105875

```
#Splitting data into sign of response variable
splitset <- split(TMINonetest, TMINonetest$UKE00105875 > 0)
negative <- splitset$`FALSE`
positive <- splitset$`TRUE`

#Plotting UKE00105875 minimum temperatures against season
ggplot(data = NULL, aes(x = DecYear - floor(DecYear), y = UKE00105875)) +
  geom_point(data = negative, col = "blue", alpha = 0.3) +
  geom_point(data = positive, col = "red", alpha = 0.3) +
  geom_hline(yintercept = 0, col="black") +
  ggtitle("Figure 5: Plot of Minimum Temperatures Against Season \n for UKE00105875") +
  xlab("Year") + ylab("Minimum Temperature (Celcius)") +
  theme(plot.title = element_text(hjust = 0.5))
```

Figure 5: Plot of Minimum Temperatures Against Season
for UKE00105875



Note that the Brier score is negatively orientated, so lower scores are again better, and it is a proper so you know it is closer to the true value as the score gets lower. The Brier score measures the accuracy of a models probabilistic predictions - in this context, a low score indicates better predictions for the chance of sub-zero temperatures.

We first observe that the naive model has a higher average Brier score than the seasonal model. This tells us that the seasonal model is better at predicting the chance of sub-zero temperatures. This is likely to be a consequence of the lack of seasonality in the naive model - the seasonal model accounts for the fact there is a higher chance of freezing temperatures in the winter months, while the naive model assumes that there is a more-or-less equal chance of freezing temperatures throughout the year in a given location.

We observe that the Brier score for the stations model is lower than both of the seasonal and naive models. This tells us the station model outperforms the others at predicting the occurrence of sub-zero temperatures. The main reason for its out performance of the naive model is identical to that of the seasonal model; it allows for seasonal variation to be incorporated into its predictions whether or not the temperatures will be sub-zero.

The station specific model outperforms the seasonal model as the seasonal model assumes that the chance of freezing temperatures varies only by location and season - it does not account for large variations / random unexpected events that deviate from what is typical within seasons. The stations model allows for this as, for instance, if a random cold front occurred at UKE00106875, then it is likely this would have also occurred at the predicting stations. We see in Figure 5 that this is surprisingly important - observing that during typically 'warmer' months in the middle third of the year there have been occurrences of negative minimum temperatures. While the seasonal model's Fourier coefficients are probably not sufficiently large enough to predict this is a possibility at all, the station covariates in the station model probably enabled the prediction to be adjusted for these random events, as intuitively the stations would also have been affected by this random event.

References

[1] Page 12.
Finn Lindgren. *Proper Scoring Rules*.
The University of Edinburgh, 2019.

[2] Stack Overflow.
https://stackoverflow.com/questions/40673461/creating-a-uk-map-with-geom-polygon
Accessed: [24/2/2019]