GlobalWarmAnalysis

June 11, 2021

0.0.1 Time Series Analysis of Minimum Tempreratures in Melbourne, Australia

In this data, we have daily readings of minimum temperatures from Melbourne, Australia for 10 years (1981-1990). I was curious to see if any global warming trends would be visible on the 10-year scale. I was also curious to see which of the models commonly used in time series forecasting would perform the best.

[309]:
```
temp <- read.csv("https://raw.githubusercontent.com/jbrownlee/Datasets/master/
↪daily-min-temperatures.csv")
```

[349]:
```
head(temp,2)
```

A data.frame: 2 × 2

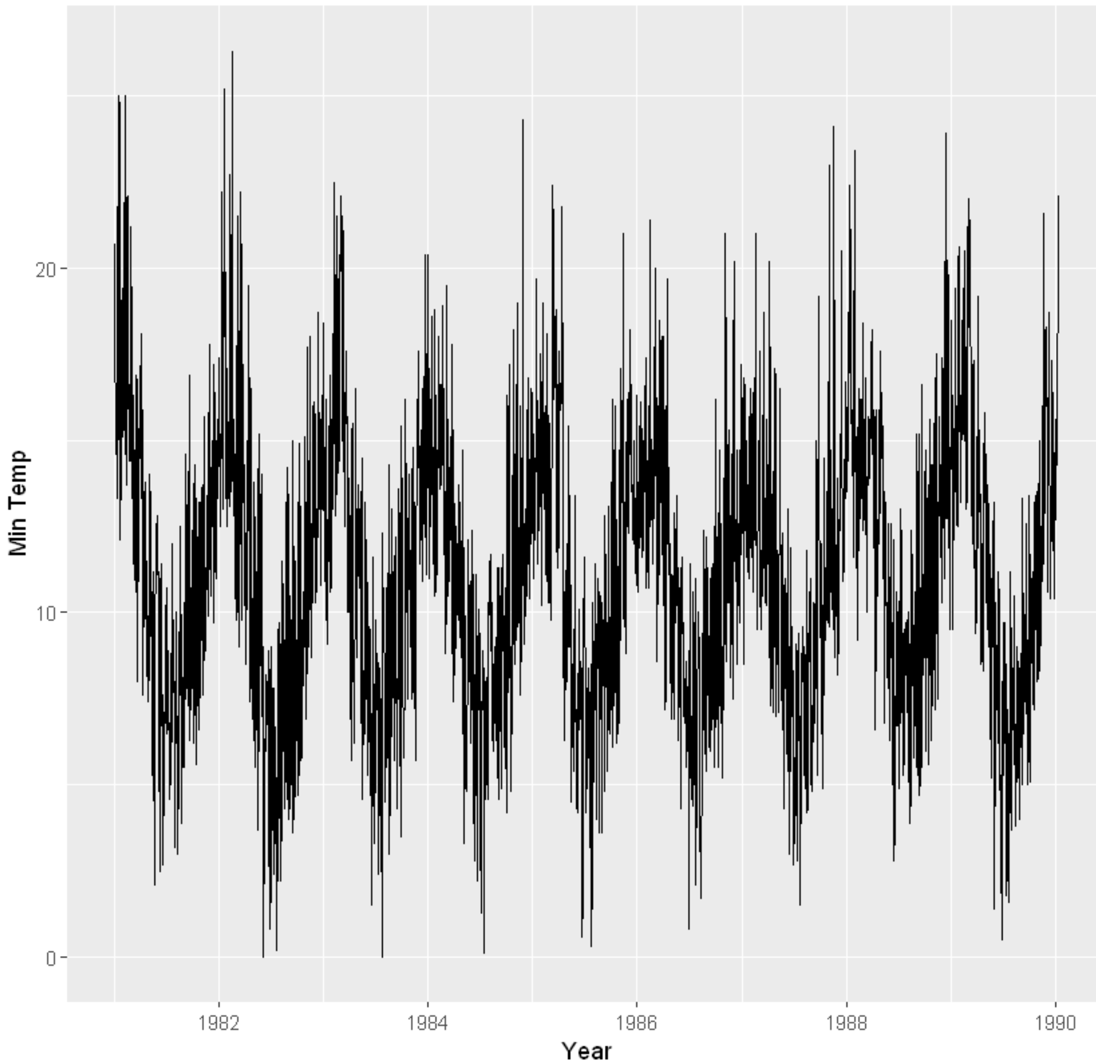
	Date	Temp
	<dttm>	<dbl>
1	1981-01-01	20.7
2	1981-01-02	17.9

[311]:
```
library(forecast)
library(lubridate)
library(dplyr)
```

create time series plot to observe the overall behavior of the data

[312]:
```
temp.ts <-ts(temp$Temp, start = c(1981,1), end = c(1990,12), frequency = 365)
```

[313]:
```
autoplot(temp.ts, xlab = "Year", ylab = "Min Temp")
```



From the plot, we can observe seasonality from the cyclical behavior along with level (average) of about 15 degrees celcius and absence of trend (there is no clear upward or downward trend). To get a better picture of seasonality, it might be better to take the average monthly readings of the temperature and plot them (which I will present below).

[314]:
```
head(temp,2)
```

A data.frame: 2 × 2

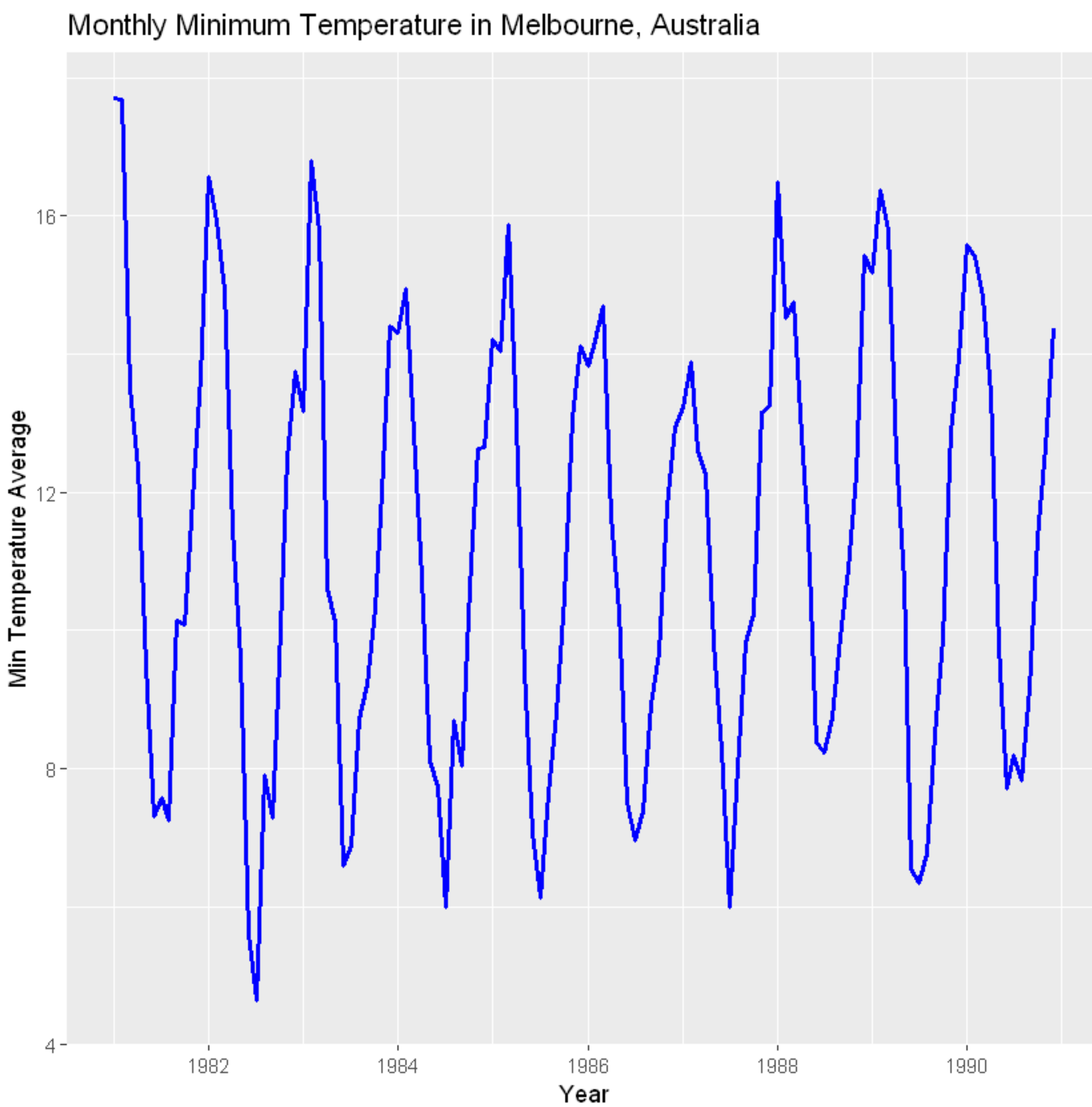
	Date	Temp
	<chr>	<dbl>
1	1981-01-01	20.7
2	1981-01-02	17.9

[315]:
```
temp$Date <- strptime(temp$Date, "%Y-%m-%d")
```

[316]:
```
avg_mon <- aggregate(Temp ~ month(Date) + year(Date), temp, FUN = mean)
```

[317]:
```
monthly_temp.ts <-ts(avg_mon$Temp, start = c(1981,1), end = c(1990,12),␣
↪frequency = 12)
```

[318]:
```
autoplot(monthly_temp.ts, xlab = "Year", ylab = "Min Temperature Average", main␣
↪= "Monthly Minimum Temperature in Melbourne, Australia",
        color = "blue", size = 1)
```

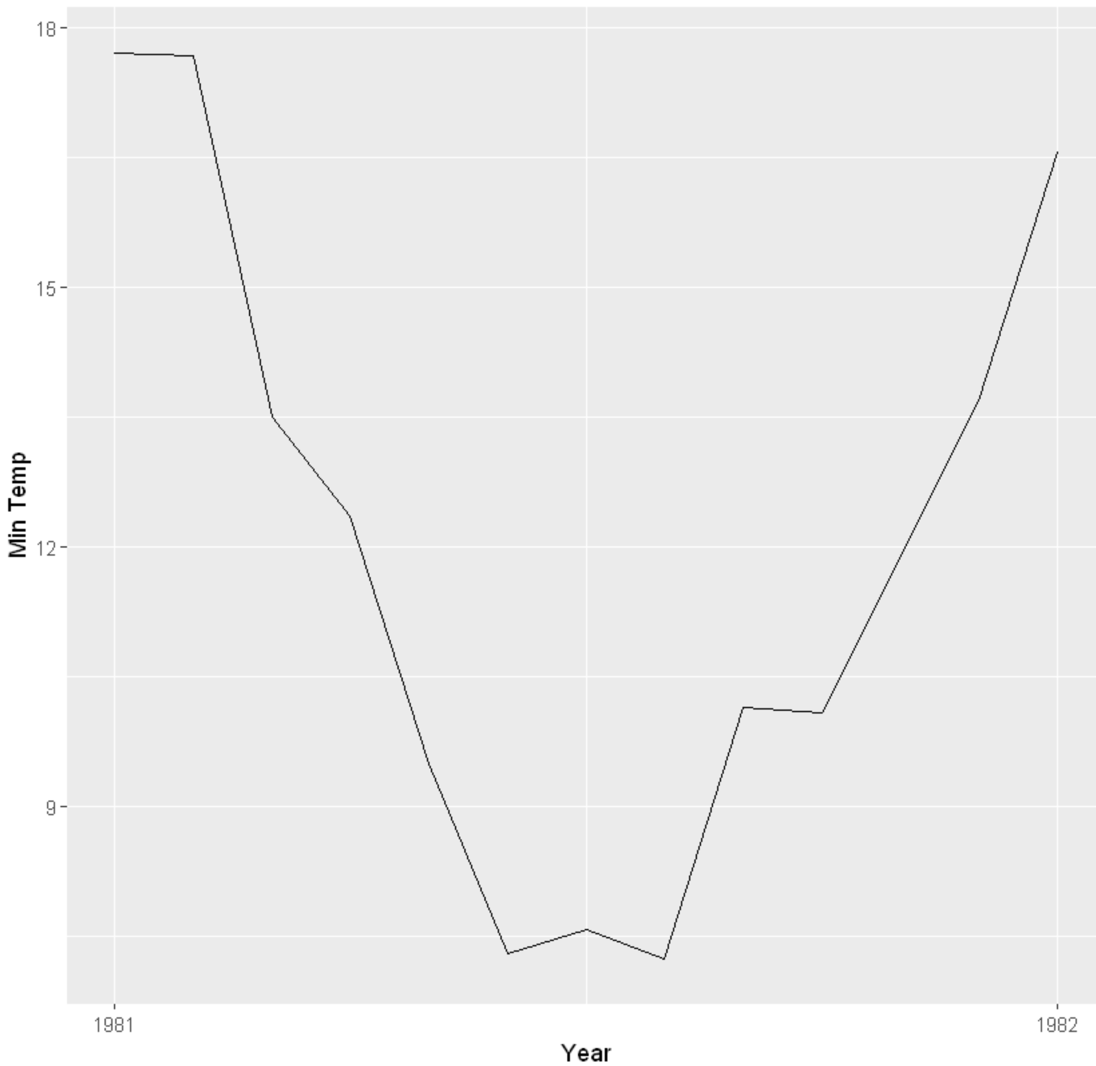


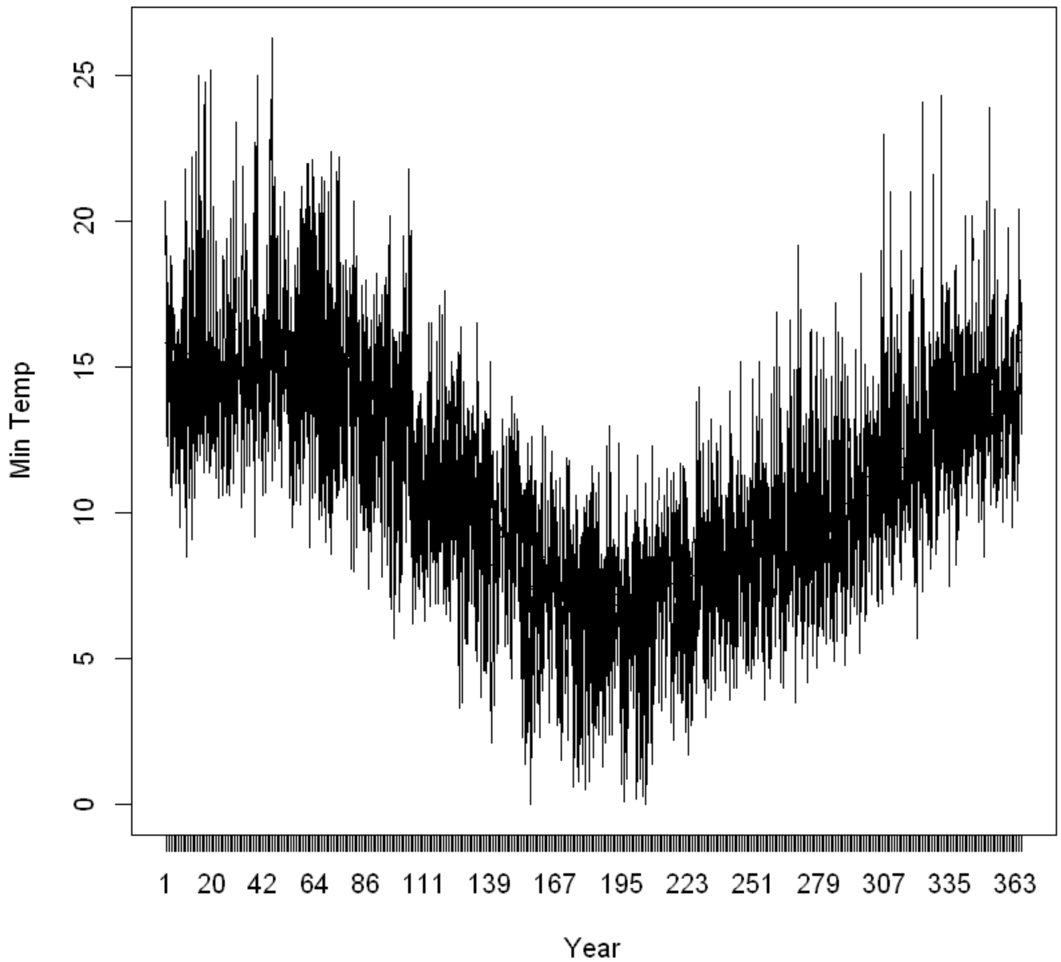
From the plot above, it is much easier to notice seasonality in the data. To better inspect the temperatire cycle, let's zoom in into 1 year of temperature reading

[328]:
```
yr_temp <- window(monthly_temp.ts, start = c(1981,1), end = c(1982,1)) # values
↪in the 1940's
```

[335]:
```
par(mfrow=c(1,2))
```

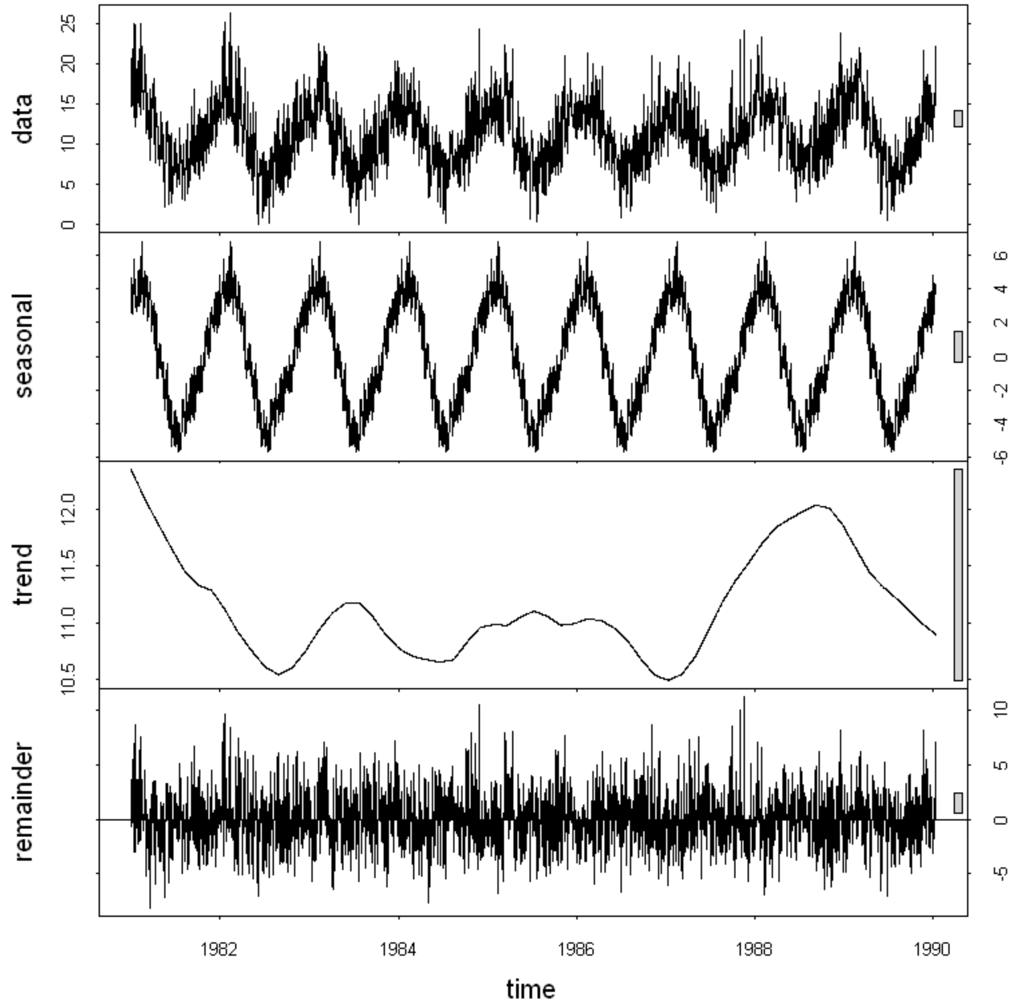
[336]:
```
autoplot(yr_temp, xlab = "Year", ylab = "Min Temp")
monthplot(temp.ts, xlab = "Year", ylab = "Min Temp") # similar result with
↪different code (plot is in days)
```





From the plots above, it seems that in June the temperature reaches its average minimum whereas its highest points are in January. This makes sense because typically winter in Austalia coincides with North America's summer time and vice versa. To fither explore the data, we are going to decompose it into 4 main parameters, which will help us figure out which models to choose from.

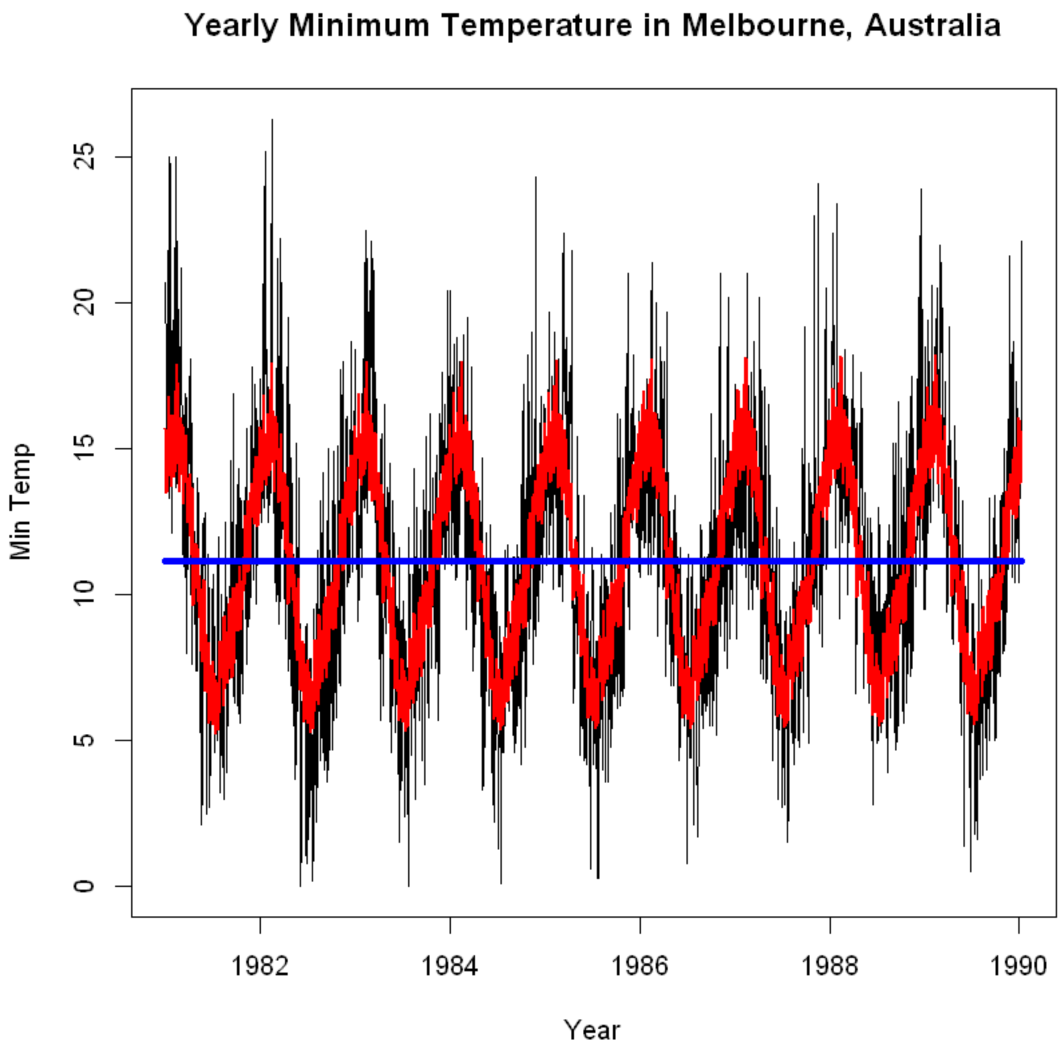
[343]:
```
# Seasonal decomposition
fit <- stl(temp.ts, s.window="period")
plot(fit)
```



Again, data shows strong seasonality and lack of trend. I will use several models to find out which one performs the best

Now that the exploratory part is over, we are going to first fit 2 linear time series models and see how they behave

[338]:
```
plot(temp.ts, xlab = "Year", ylab = "Min Temp", main = "Yearly Minimum␣
 ↪Temperature in Melbourne, Australia")
temp.season.tslm <-tslm(temp.ts ~ trend + season)
temp.tslm <- tslm(temp.ts ~ trend)
lines(temp.season.tslm$fitted, lwd = 2, col = "red") # model with season
lines(temp.tslm$fitted, lwd = 4, col = "blue") # model without season
```

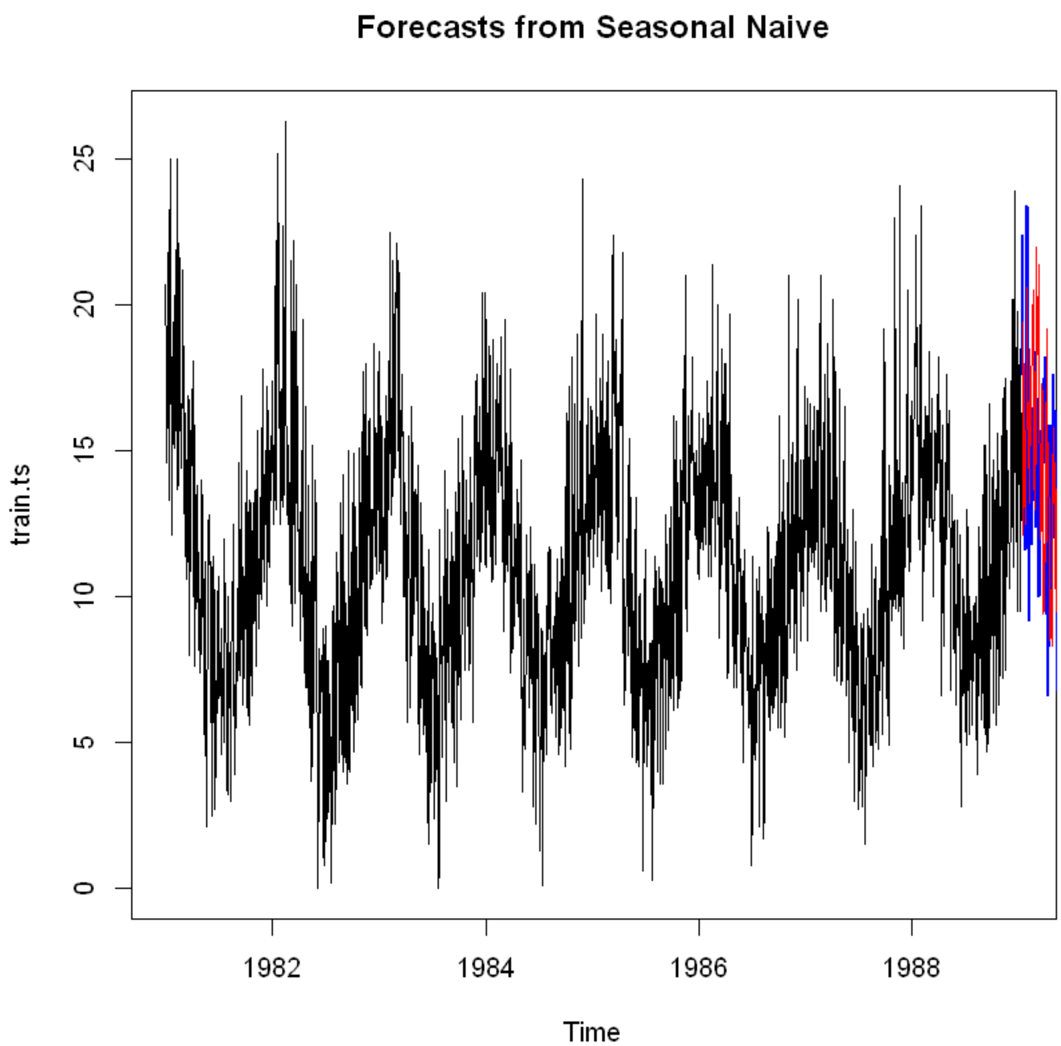


While the linear model without seasonality fails to capture the data trend, the linear model with trend and seasonality does a better job. In order to evaluate model's accuracy, we should split the data into test and train sets. I am going to build and compare 3 models: seasonal naive, Hold Winter's, and Linear Time Series

[341]:
```
# Model Building - Seasonal Naive

nValid <- 365
nTrain <- length(temp.ts) - nValid

# create training and validation partitions for temperature
temp.train.ts <- window(temp.ts, start = c(1981, 1), end = c(1981, nTrain))
```

```
temp.valid.ts <- window(temp.ts, start = c(1981, nTrain + 1), end = c(1990,␣
 ↪nTrain + nValid))

train.ts <- temp.train.ts
valid.ts <- temp.valid.ts

snaive.pred <- snaive(train.ts, h = nValid)

plot(train.ts, main = "Forecasts from Seasonal Naive")
lines(snaive.pred$mean, lwd = 2, col = "blue", lty = 1) # prediction
lines(valid.ts, col="red") # validation
```

```
Warning message in window.default(x, …):
"'end' value not changed"
```



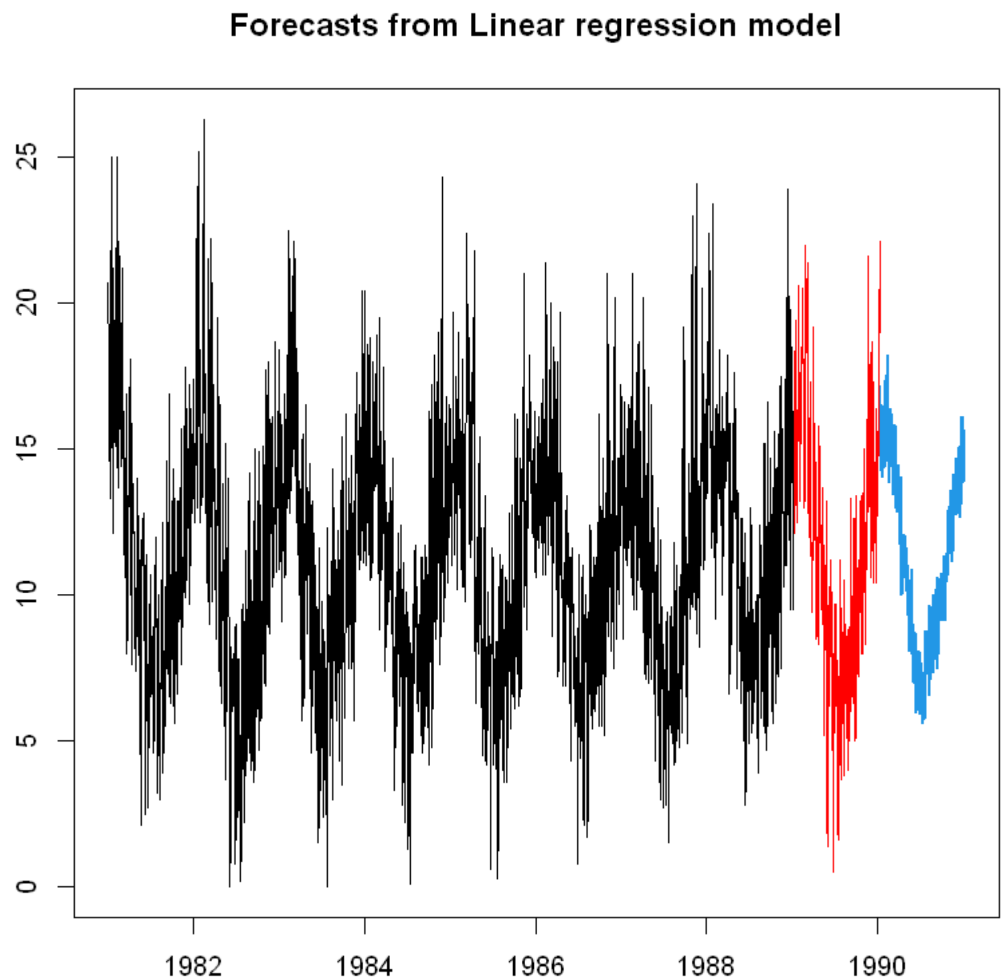
[342]:

```
a1 = accuracy(snaive.pred, valid.ts)
```

[257]:

```
# Model Building - Linear Time Series

nValid <- 365
nTrain <- length(temp.ts) - nValid

# create training and validation partitions for temperature
temp.train.ts <- window(temp.ts, start = c(1981, 1), end = c(1981, nTrain))
temp.valid.ts <- window(temp.ts, start = c(1981, nTrain + 1),
                        end = c(1990, nTrain + nValid))

train.ts <- temp.train.ts
valid.ts <- temp.valid.ts


train.lm.season <- tslm(temp.ts ~ trend + season)
train.lm.season.pred <- forecast(train.lm.season, h = nValid, level=0) # takes␣
↪the average forecast readings

plot(train.lm.season.pred)
lines(valid.ts, col="red")
```

```
Warning message in window.default(x, …):
"'end' value not changed"
```



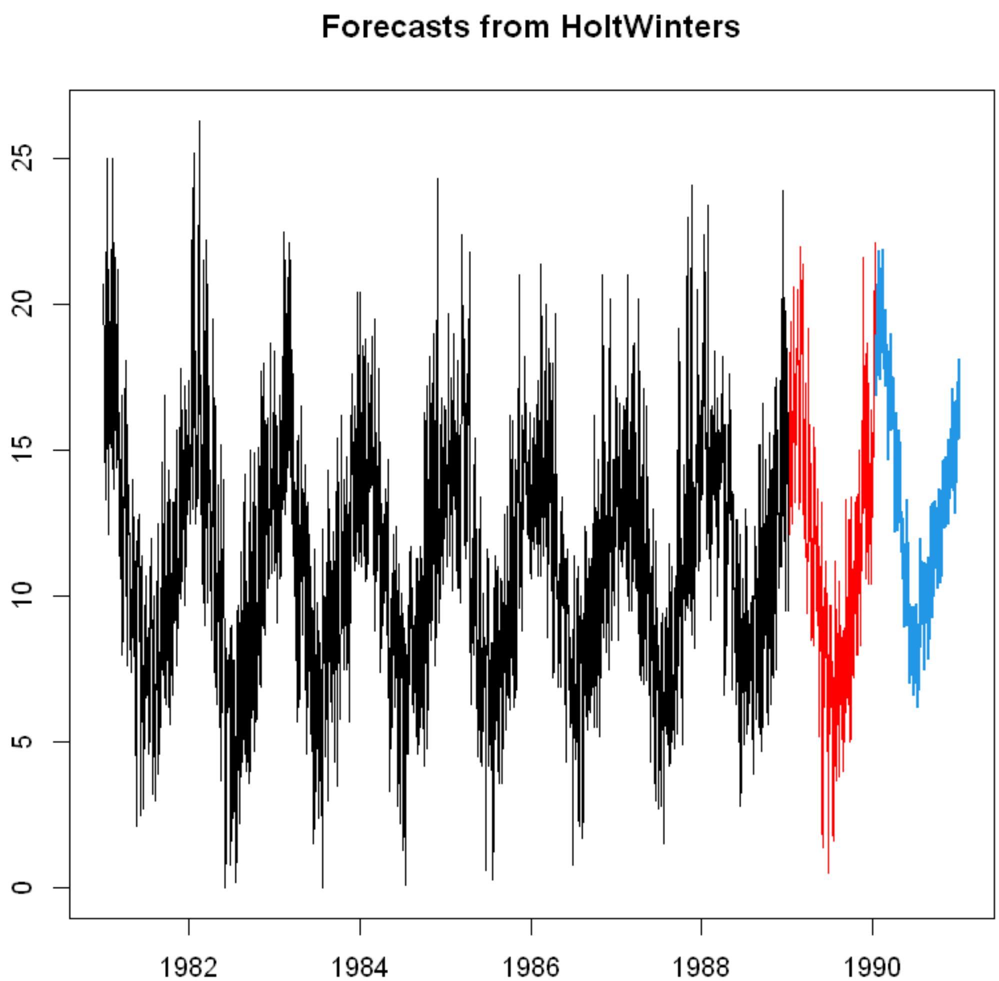
[260]:
```
forecasted_Values <-data.frame(train.lm.season.pred)$Point.Forecast
```

[261]:
```
a2 = accuracy(valid.ts,forecasted_Values)
```

[263]:
```
# Model Building - Holt Winter's

nValid <- 365
nTrain <- length(temp.ts) - nValid

# create training and validation partitions for temperature readings
train.ts <- window(temp.ts, start = c(1981, 1))
valid.ts <- window(temp.ts, start = c(1981, nTrain + 1),
                   end = c(1990, nTrain + nValid))
```

```
train.hw.season <- HoltWinters(train.ts)
train.hw.season.pred <- forecast(train.hw.season, h = nValid, level=0) # takes
↪the average forecast readings

plot(train.hw.season.pred)
lines(valid.ts, col="red")
```

```
Warning message in window.default(x, …):
"'end' value not changed"
```



[264]:
```
forecasted_Values <-data.frame(train.hw.season.pred)$Point.Forecast
a3 = accuracy(valid.ts,forecasted_Values)
```

```
[344]: mat2<-t(as.matrix(a1[2,1:5])) # extract the test set of the seasonal naive␣
       ↪accuracy
```

```
[345]: a.table<-rbind(mat2,a2, a3)
```

```
[346]: row.names(a.table) <-c('Seasonal Naive','Time Series Linear␣
       ↪Model','Holt-Winter')
```

```
[347]: a.table
```

A matrix: 3 × 5 of type dbl

	ME	RMSE	MAE	MPE	MAPE
Seasonal Naive	-0.586849315	3.825430	3.073973	-24.842553	42.578
Time Series Linear Model	-0.006436047	2.359140	1.847088	1.567733	17.734
Holt-Winter	1.915609706	2.911865	2.421405	16.161779	19.984

Conclusion: we normally use RMSE to evaluate model accuracy. Even though I expected for Holt Winter's model to have the best performance, it turns out that a simpler time series linear model has the best performance. Seasonal naive was used as a base model. It is used for highly seasonal data and the forecasting accuracy might've not been as good as time series' because the model uses the most recently observed value.

Even though we learned a lot about the models, it wasn't possible to conclude anything global warming tendencies. There are 2 possible reasons for this: 1). The range of data is not large enough. It might be more reasonable to obtain at least 50 years of minimum temperature readings and take a look at the data trend to make more appropriate conclusions; 2). While only Melbourne, Austria, was considered in the this analysis, it might be more effective to consider at lear 3 various locations to draw out a better conclusion.

```
[ ]:
```