Further Analysis

Yi Lin Liu

April 4, 2017

Summary of Further Steps:

- Temperature Data for several days have been collected under `temperature- Copy.csv`. To analyze further, several different graphs will be made.
- The same time vs temperature series graph, except this time, the first 24 hour, 12 first and 6 hour past predictions will be plotted separately.
  - It is predicted that the error bars (prediction errors) will be smaller for the 1-6h predictions compared to the 1-12 hour predictions and also the 1-24h predictions. This is because the closer the prediction to the actual time, the closer the predicted temperature should be to the actual temperature.
- A plot of the differences between the predicted temperatures (error bars) and the current temperature will be plotted.
- A histogram of when does the worst prediction occur latest, as in the worst forecast that is nearest in time to the actual measured temperature.

Organizing Raw Data

- This section is more or less a replica of the steps taken to lag the columns and take only the completed rows from the *Weather_mini_project*.
- The organized data is shown under `24hForecast.csv`
  - *max_Diff:* the closest worst prediction hour.
  - *For1h,For2h...for24h:* the forecasted temperature 1 hour,2 hour, ... 24 hours ago.

```
rm(list = ls())

raw_File = "temperature - Copy.csv"
raw_Data = read.table(file=raw_File,header=TRUE,
                      sep=",",stringsAsFactors = FALSE)

length=dim(raw_Data)[1]
raw_Data = raw_Data[seq(from=1,to=length,by=2),]



## Lagging each column, starting from 3rd column
## made into a quick function.
LagData = function(data,start_Col=3, lag=1, lag_Increment=1){

    for( i in start_Col: length(names(data))){
        data[,start_Col] = data.table::shift(
```

```
            x= data[,start_Col], n=lag
        )
        start_Col = start_Col+1
        lag =lag+lag_Increment
    }

    return(data)
}

lagged = LagData(data=raw_Data, lag_Increment = 1)


##  Selecting only the complete data
library(dplyr)
data = lagged %>%
    filter(!is.na(forcast.temperature.in.24h..C.))

##  Changing the column names:
col_Names = c("Date","current")
for(i in 3:length(names(data))){
    col_Names = append(col_Names, paste("For",i-2,"h",sep=""))
}

colnames(data) <- col_Names
```

Plotting Time vs Temp Series:

1: Clearing Memory:

```
rm(list = setdiff(ls(),"data"))
```

2: Getting All Data:

- End goal is to calculate all the required data.
  - max, min predicated values
  - error bar values
  - hour indicating worst predication closest to actual temperature in time.

```
##  include: the list of cols(names/index) to include for
##  data frame
##  Returns new data frame containing
##  date, current temp,closest worse perdication hours wanted,max and min error bars
Organize = function(data, include){

    new_Data =data[include]
    ##  Max and Min predictions
    max = apply(new_Data,1,function(x){ max(x)} )
    min = apply(new_Data,1,function(x){ min(x)} )


    ##  Processing the index(hour) of maximum difference
    max_Location = vector()
    for(i in 1: length(max)){
```

```
        row = new_Data[i,]
        ##  Which number to look for
        if(abs(max[i] - data$current[i])>
           abs(min[i] - data$current[i]))
            locate = max[i]
        else
            locate = min[i]

        ##  Finding index(forecast hour)
        if(max[i]==data$current[i] &
           min[i]==data$current[i])
            max_Location[i]=0
        else
            max_Location[i]= match(locate, row)


    }

    ##  attaching to a data frame.

    new_Data = cbind(date = data$Date,
                     current = data$current,
                    new_Data,
                    maxPredict=max,
                    minPredict=min,
                    max_Diff = max_Location)

    ##Adding the erMax and erMin from current Temp

    new_Data = new_Data %>%
        mutate(erMax=maxPredict-current)%>%
        mutate(erMin=minPredict-current)
    return(new_Data)
}
```

- Making data frames for 24 hour forecast, 12 hours forecast and 6 hour forecasts

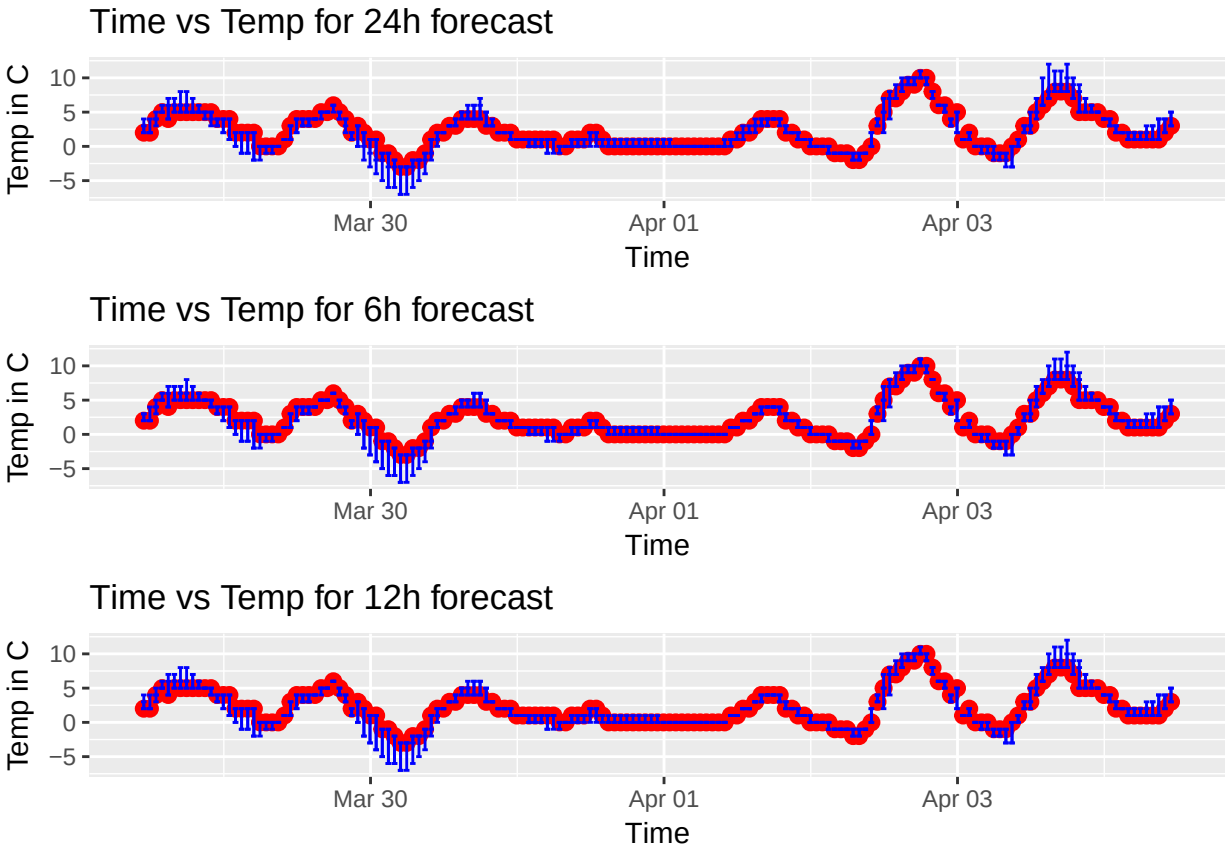
```
names = colnames(data)

include24h = grep("For1h",names):
    grep("For24h",names)
data24h = Organize(data,include24h )

include12h = grep("For1h",names):
    grep("For12h",names)
data12h = Organize(data, include12h)


include6h = grep("For1h",names):
    grep("For6h",names)
data6h = Organize(data, include6h)
```

3: Plotting the Time vs Temp Series:

```
plotTimeTempSeries = function(data, title){
    library(ggplot2)
    plot = ggplot(data=data, aes(x=strptime(data$date,format="%m/%d/%Y %H:%M"),
            y=data$current) ) + geom_point(color="red",size=2.5)+
        scale_x_datetime() +
        ##  titles and things
        xlab("Time")+ylab("Temp in C")+
        ggtitle(title)+
        ##  Error Bar
        geom_errorbar(aes(ymin=data$current,
                          ymax = data$maxPredict),
                      color = "blue")+
            geom_errorbar(aes(ymin=data$minPredict,
                          ymax = data$current),
                          color="blue")


}
```

- Formatting Notes

```
library(ggplot2)
library(gridExtra)

p24h = plotTimeTempSeries(data24h,"Time vs Temp for 24h forecast")
p12h = plotTimeTempSeries(data12h,"Time vs Temp for 12h forecast")
p6h = plotTimeTempSeries(data6h,"Time vs Temp for 6h forecast")


grid.arrange(p24h,p6h,p12h)
```




```
##  Save the graphs for better viewing
ggsave("./Pictures/p24h.png", p24h, device="png")
ggsave("./Pictures/p12h.png", p12h, device="png")
ggsave("./Pictures/p6h.png", p6h, device="png")
```

- As predicted, the sooner the prediction. The more accurate the predictions.The effect is not immediately clear. This is likely because the error bars only express max and min values but neglates frequency of measurements.
- An interesting note is that the error bars tend to swing in the same direction as the slope. If temperature is rising, the predictions tend to be higher than the actual temperature and vice-versa.
- **The temperature on the website only reports to the nearest degree.**

Plotting Difference Graph Between Error Bars:

1: Make a function that plots the graph of error bars

```
plotEr = function(data, title){
   library(ggplot2)
   library(tidyr)
   library(dplyr)

   ## joining the erUp and erDown
   plot_Data = select(data, c(date,erMax,erMin))
   plot_Data = gather(plot_Data, key="Type",
                      value = "value",erMax,erMin)
```

```
    plot = ggplot(data=plot_Data, aes(x=strptime(plot_Data$date,format="%m/%d/%Y %H:%M"),
            y=value, color=Type) ) + geom_line()+
        scale_x_datetime() +
        ##  titles and things
        xlab("Time")+ylab("Magnitude of Error")+
        ggtitle(title)+facet_wrap(~Type)

    return(plot)
}
```

2: Plotting the Error:

```
p24hEr = plotEr(data24h,"Er vs Time")
p24hEr
```


```
ggsave(file="./Pictures/p24hEr.png",p24hEr,device="png")
```

```
## Saving 6.5 x 4.5 in image
```

Worst Perdication Time:

- The idea is to locate the least accurate predication for every temperature measurement and produce a histogram, where the bins are the hours of forecast that showed the worst predictions closest.

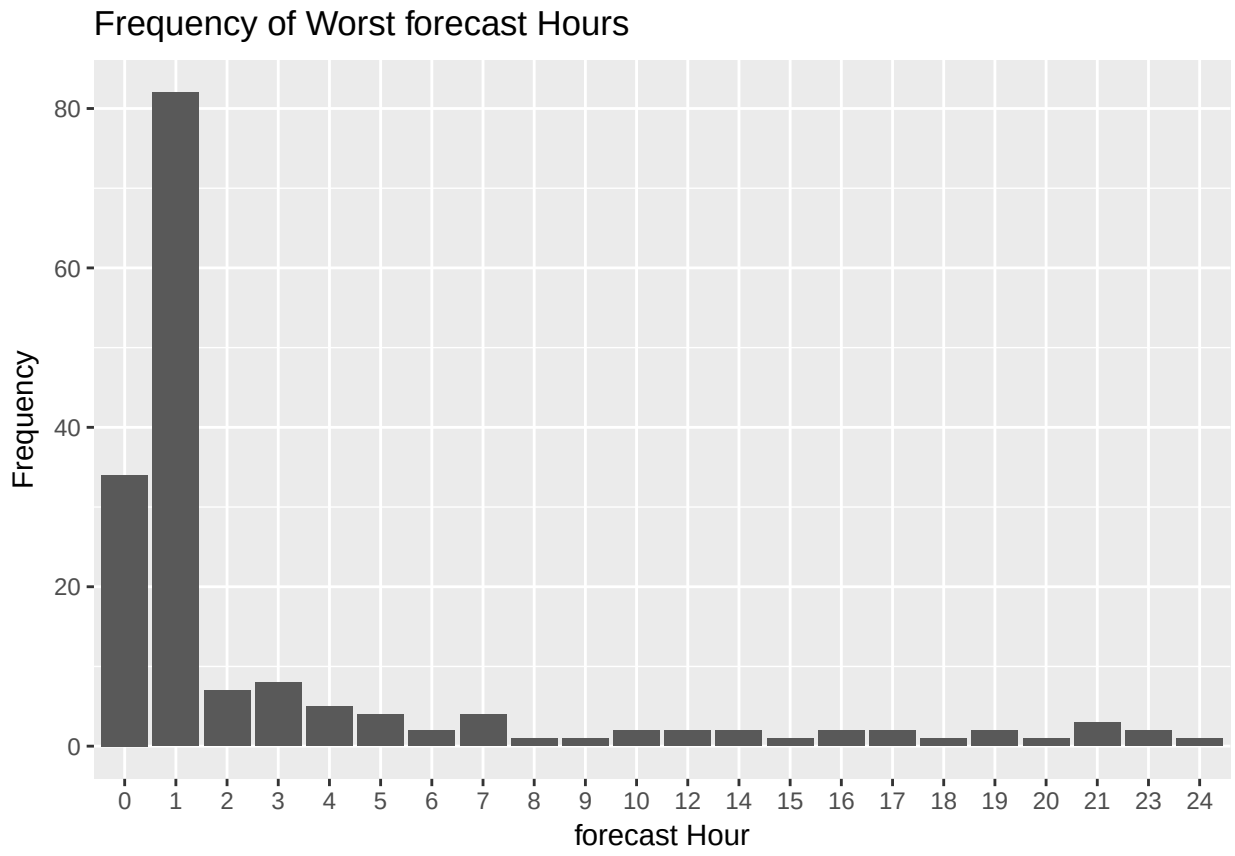
1: Plotting Earliest Worst Prediction Vs Time

```
 plotWorstAsTime = function(data, title){
    library(ggplot2)

    plot = ggplot(data=data, aes(x=strptime(data$date,format="%m/%d/%Y %H:%M"),
            y=max_Diff) ) + geom_point()+
        scale_x_datetime() +
        ##  titles and things
        xlab("Time")+ylab("Earliest Worst forecast Hour")+
        ggtitle(title)

    return(plot)
 }
```

```
forWT = plotWorstAsTime(data24h,"Earliest Worst Prediction")

##  Histograms
forWTH = ggplot(data = as.data.frame(
    table(data24h$max_Diff)),
    aes(x=Var1,y=Freq))+
    geom_bar(stat="identity")+
    xlab("forecast Hour")+ ylab("Frequency")+
    ggtitle("Frequency of Worst forecast Hours")

ggsave("./Pictures/ClosestWorstPredication.png", forWTH, device="png")
```

```
## Saving 6.5 x 4.5 in image
```

```
write.table(data24h,file="24hForecast.csv",sep=",",
            col.names = TRUE, row.names=FALSE)

forWTH
```



- The majority of the worst predictions occur at the closest prediction (1 hour forecast). This means that all forecasts (1-24 hours) predicted the same temperature. I personally believe that this likely results from rounding. It is highly unlikely that the actual predictions are all the same temperature. It might also be because the model used to predict the future temperatures does not change much based on newly collected data. Also the temperature is only reported to the nearest degree, which is not precise. The actual predicted temperatures could be changing slightly, but just not reported.
- Another note is that the majority of closest worst prediction time is at early forecast hours (even when disregarding the 1hour and 0h forecasts). This again supports the hypothesis that the model for prediction does not vary much based on new data.