LDAPS Regression Analysis

Will Rauen

2024-02-21

Setup

For this Data Analysis we will be using the tidyverse and ggplot2 packages in R.

```
require(tidyverse)
```

```
## Loading required package: tidyverse
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.4     v purrr   1.0.2
## v tibble  3.2.1     v dplyr   1.1.4
## v tidyr   1.3.1     v stringr 1.5.1
## 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()
```

```
require(ggplot2)
```

Data

```
climate_data <- read.csv("/Users/williamrauen/Desktop/DS 3100/Assignments/Assignment 2/Data/LDAPS.csv")

dim(climate_data)
```

```
## [1] 7752   25
```

The data used in this dataset comes from the UC Irvine Machine Learning Repository and can be accessed with the link:

https://archive.ics.uci.edu/dataset/514/bias+correction+of+numerical+prediction+model+temperature+forecast

This specific dataset takes data from 25 weather stations in the country of South Korea each of which recording fourteen numerical weather predictions for future forecast data. The dataset containss from the years 2013 - 2016 and only recorded data points during the summer. The input data is largely comprised of the LDAPS model's next-day forecast data, which is the current model operated by the Korea Meteorological Administration over Seoul, South Korea. Thus, the data is organized by independent variables columns such as the different LDAPS predictors, daily maximum and minimum temperatures, and geographic auxiliary variables, and two dependent columns being the next days maximum and minimum temperatures (Celsius).

Research Question

In this assignment, I aim to see if solely the previous day's maximum and minimum temperature is the best predictor of the next day's maximum and minimum temperature. If not, does Korea's LDAPS model predict the next day's maximum and minimum temperatures more accurately?

Variables of Interest

Our two response variables in the dataset are labeled with the column names "Next_Tmax" and "Next_Tmin" representing the next day's maximum and minimum temepratures respectively. Important independent variables of interest are labeled as "Present_Tmax" and "Present_Tmin" which are the present day's maximum and minimum temperatures, and "LDAPS_Tmax_lapse" and "LDAPS_Tmin_lapse" which are the LDAPS model forecast of next-day air temperature. Perhaps other variables concerning LDAPS prediction of next day humidity, cloud coverage, wind, and daily solar radiation could be of interest.

Data Wrangling

The first thing to check for in the dataset is missing data. Upon inspection, there were found to be some missing values of data across various columns, so I decided to omit those rows of data entirely. This means my new cleaned data has 174 less rows than before.

```
climate_data %>% #Count missingness in data
  count(NaN)
```

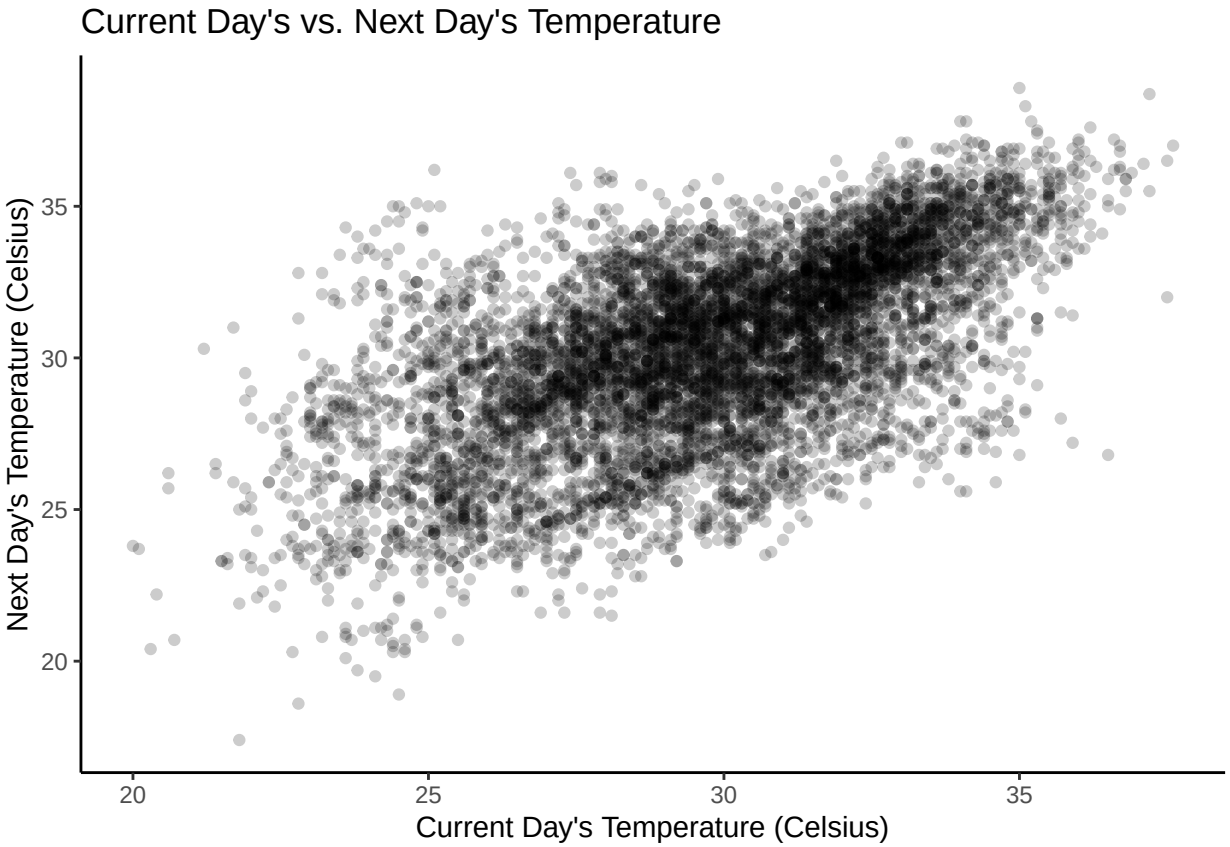
```
##   NaN    n
## 1 NaN 7752
```

```
climate_data_new <- climate_data %>% #Creates new datafram with completed rows
  drop_na()

dim(climate_data_new) #dimensions of new data set
```

```
## [1] 7588   25
```

The next step is to check if the data can be modeled linearly by making a simple graph of our parameters of interest.

```
climate_data_new %>%
  ggplot(aes(x = Present_Tmax, y = Next_Tmax)) +
  geom_point(alpha = 0.2) +
  labs(title = "Current Day's vs. Next Day's Temperature",
       x = "Current Day's Temperature (Celsius)",
       y = "Next Day's Temperature (Celsius)") +
  theme_classic()
```



Since the data visualization shows no direct abnormalities that would indicate another model should be used, we can safely proceed with using a linear regression model for this data set.

Lastly, since we have a large population with over 7500 rows of data, it might benefit us to subset a sample of the data. Since the population is relatively large (> 1000 samples) we will used the general rule of subsetting around 10% of the data. In this case I've chosen to the subset the data into a sample of 800.

```
set.seed(123) #seed for randomness
# Obtain indices for 100 players at random
sample_climate <- sample(1:nrow(climate_data_new), 800)
# Obtain sub-sample
climate_sample <- climate_data_new[sample_climate,]
```

Data Analysis

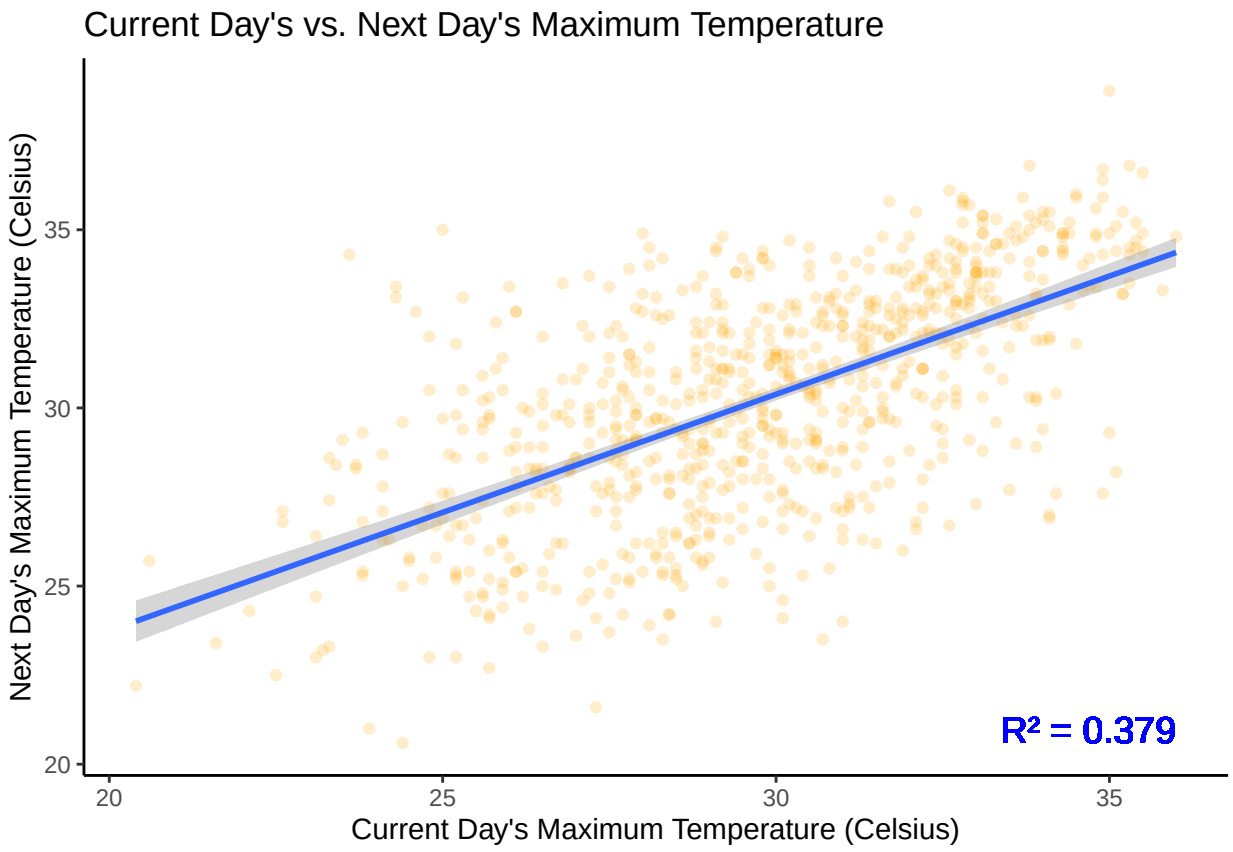
Investigation 1

The first investigation will see if daily maximum/minimum temperature is a good indicator of the next day's maximum/minimum temperature.

```
lm_max1 <- lm(Next_Tmax ~ Present_Tmax, data = climate_sample)


lm_min1 <- lm(Next_Tmin ~ Present_Tmin, data = climate_sample)

# Graph for Predicting Max Temperature
climate_sample %>%
  ggplot(aes(x = Present_Tmax, y = Next_Tmax)) +
```

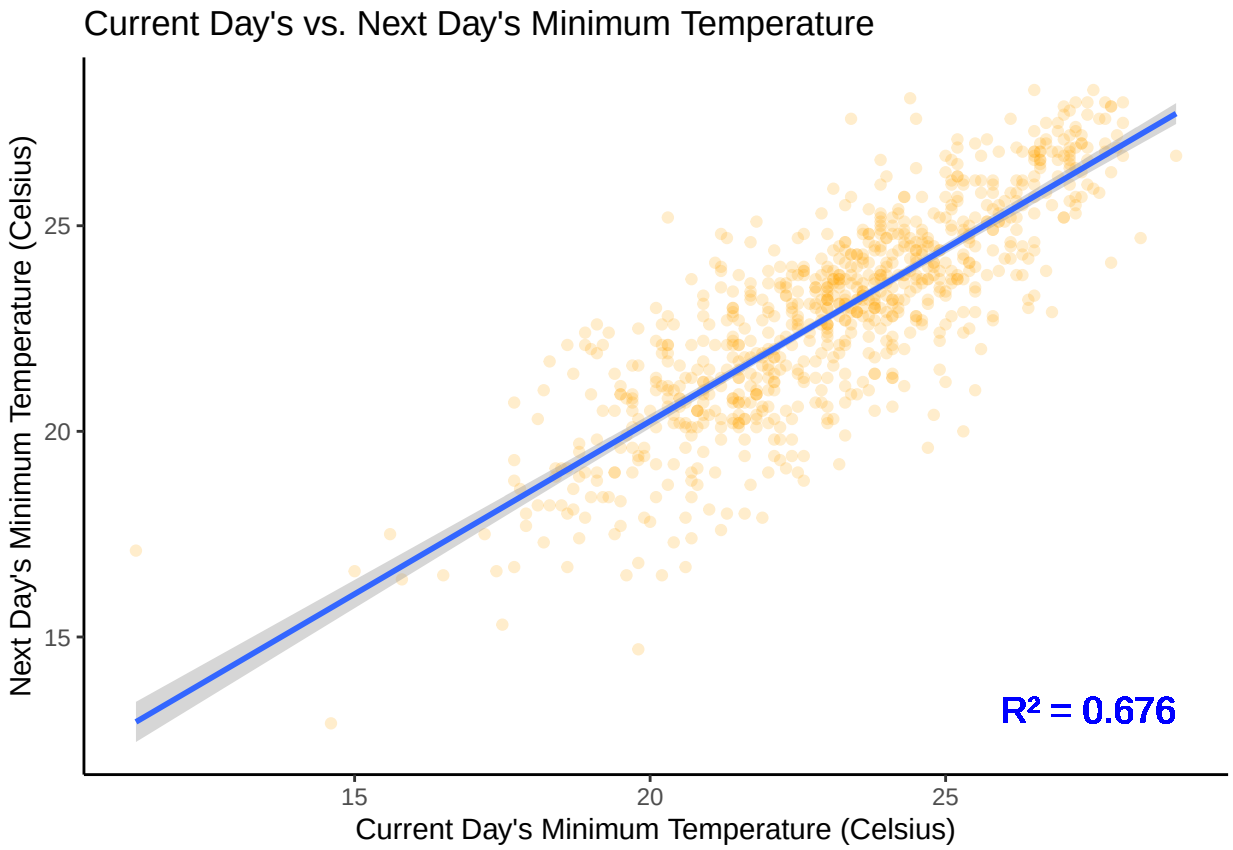
```
  geom_point(alpha = 0.2, color = "orange") +
  geom_smooth(method = lm) +
  geom_text(aes(x = max(Present_Tmax), y = min(Next_Tmax),
                label = paste("R² =", round(summary(lm_max1)$r.squared, 3))),
            hjust = 1, vjust = 0, color = "blue", size = 5) +
  labs(title = "Current Day's vs. Next Day's Maximum Temperature",
       x = "Current Day's Maximum Temperature (Celsius)",
       y = "Next Day's Maximum Temperature (Celsius)") +
  theme_classic()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
#Graph for Predicting Min Temperature
climate_sample %>%
  ggplot(aes(x = Present_Tmin, y = Next_Tmin)) +
  geom_point(alpha = 0.2, color = "orange") +
  geom_smooth(method = lm) +
  geom_text(aes(x = max(Present_Tmin), y = min(Next_Tmin),
                label = paste("R² =", round(summary(lm_min1)$r.squared, 3))),
            hjust = 1, vjust = 0, color = "blue", size = 5) +
  labs(title = "Current Day's vs. Next Day's Minimum Temperature",
       x = "Current Day's Minimum Temperature (Celsius)",
       y = "Next Day's Minimum Temperature (Celsius)") +
  theme_classic()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Interestingly, the graphs reveal that there is a pretty significant difference in the ability for current temperature to predict the next day's temperature. While predicting the next day's maximum temperature R-squared value is 0.38, the corresponding R-squared value for minimum temperature prediction is 0.68. This would imply that there is a much stronger linear relationship for minimum temperature prediction compared to maximum temperature prediction.

Residual Analysis and Homoscedasticity

```
# Looking at Homoscedasticity of Maximum Temperature

summary(lm_max1)
```

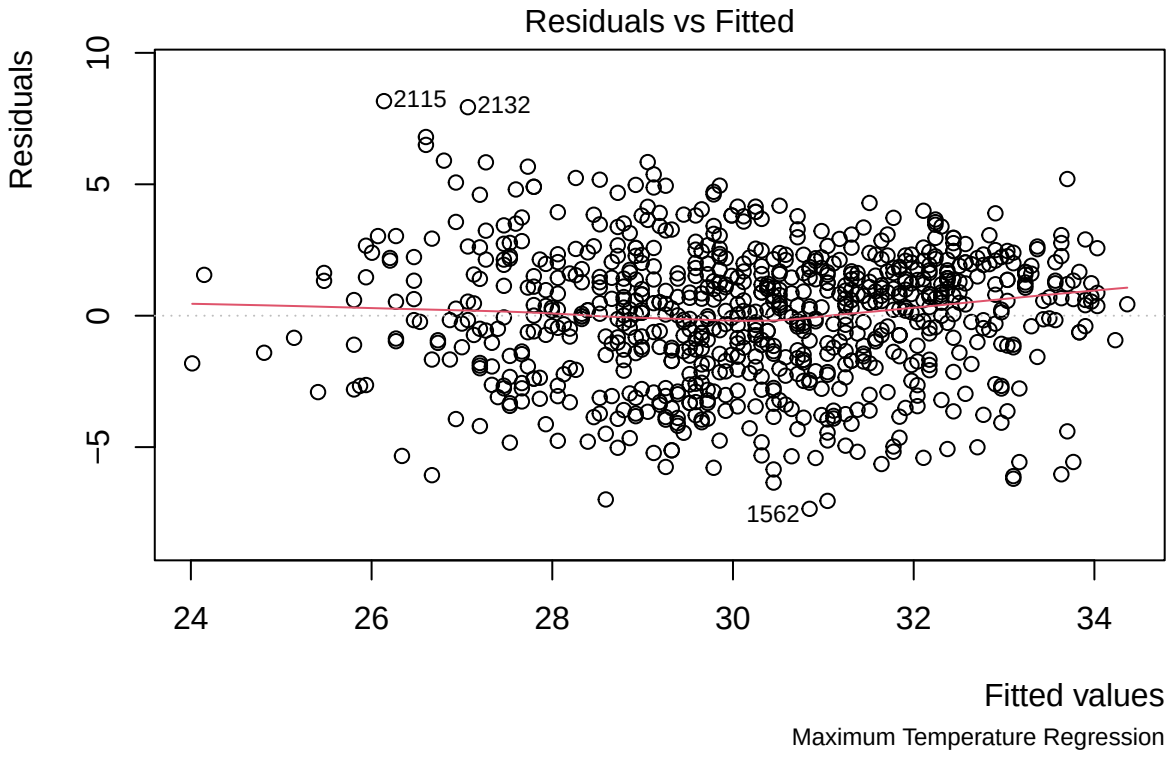
```
##
## Call:
## lm(formula = Next_Tmax ~ Present_Tmax, data = climate_sample)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.3477 -1.6697  0.2724  1.6826  8.1627
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.47999    0.89926   11.65   <2e-16 ***
## Present_Tmax  0.66344    0.03008   22.05   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.515 on 798 degrees of freedom
```

```
## Multiple R-squared:  0.3787, Adjusted R-squared:  0.3779
## F-statistic: 486.4 on 1 and 798 DF,  p-value: < 2.2e-16
```

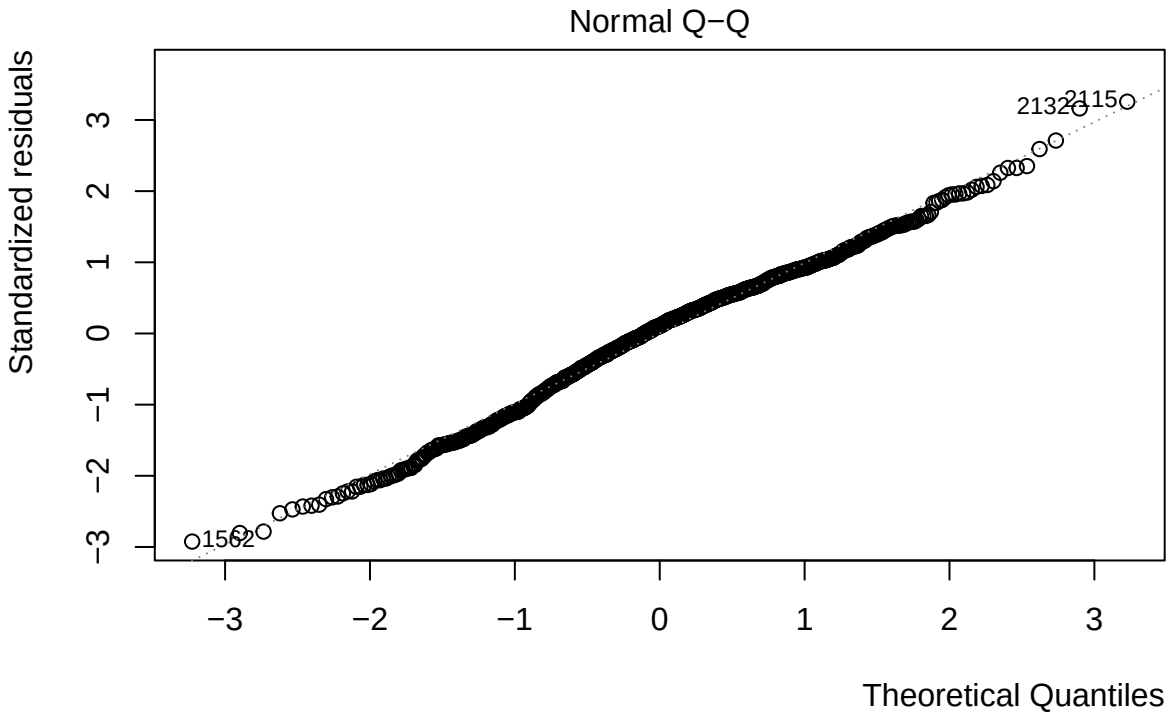
```
shapiro.test(residuals(lm_max1)) #Shapiro test for maximum temp. regression
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(lm_max1)
## W = 0.99181, p-value = 0.0002073
```

```
## Residual vs Fitted Plot for maximum temp regression
plot(lm_max1, which = 1,
     sub = "Maximum Temperature Regression",
     cex.sub = 0.75,
     adj = 1)
```




```
## QQ Plot for maximum temp. regression
plot(lm_max1, which = 2,
     sub = "Maximum Temperature Regression",
     cex.sub = 0.75,
     adj = 1)
```



Maximum Temperature Regression

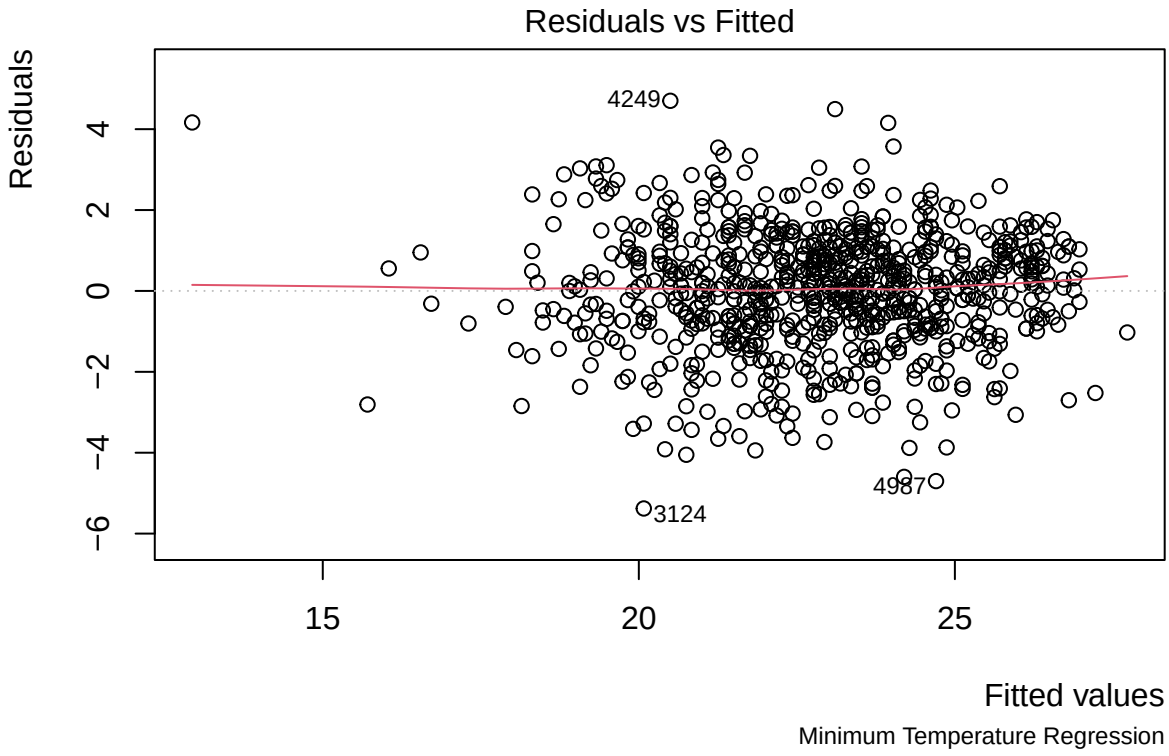
```
# Looking at Homoscedasticity of Minimum Temperature
summary(lm_min1)
```

```
##
## Call:
## lm(formula = Next_Tmin ~ Present_Tmin, data = climate_sample)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.3786 -0.8337  0.0890  0.9608  4.7011
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.43626    0.47885   7.176 1.64e-12 ***
## Present_Tmin  0.84052    0.02059  40.828  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.448 on 798 degrees of freedom
## Multiple R-squared:  0.6763, Adjusted R-squared:  0.6759
## F-statistic:  1667 on 1 and 798 DF,  p-value: < 2.2e-16
```

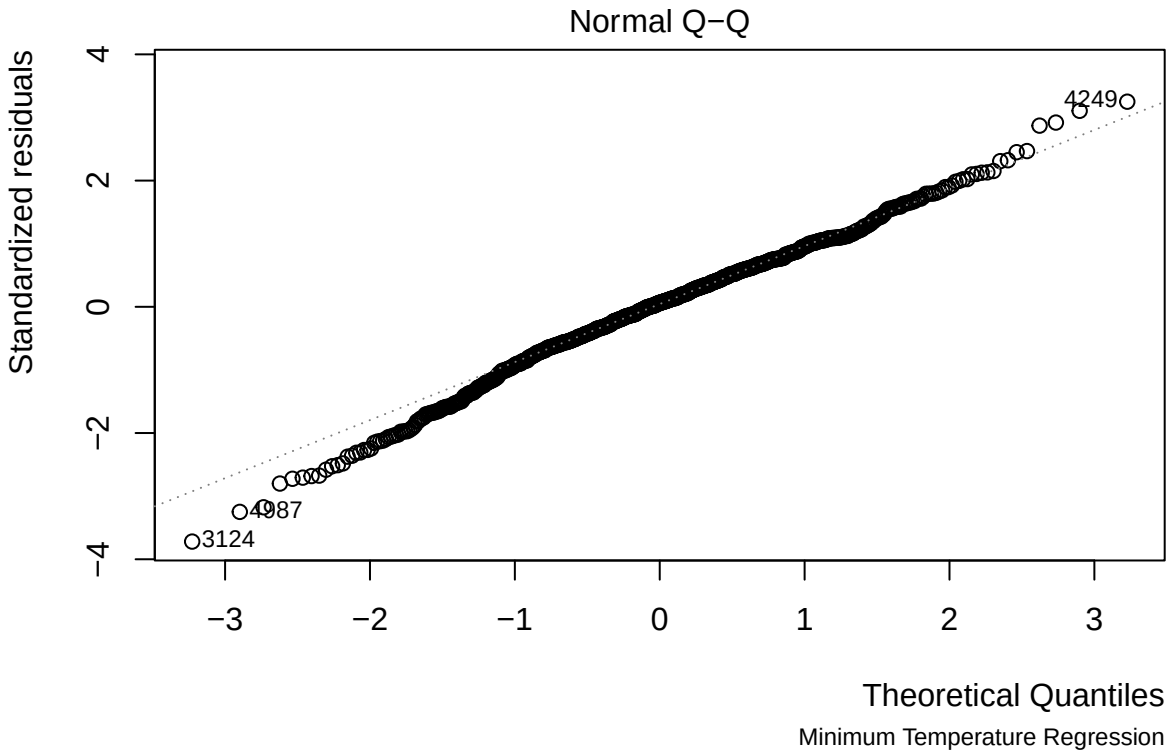
```
shapiro.test(residuals(lm_min1)) #Shapiro test for minimum temp. regression
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(lm_min1)
## W = 0.9919, p-value = 0.0002289
```

```
#Residual vs Fitted Plot for minimum temp. regression
plot(lm_min1, which = 1,
     sub = "Minimum Temperature Regression",
     cex.sub = 0.75,
     adj = 1)
```




```
#QQ Plot for minimum temp. regression
plot(lm_min1, which = 2,
     sub = "Minimum Temperature Regression",
     cex.sub = 0.75,
     adj = 1)
```



Minimum Temperature Regression

The residual values for both the maximum and minimum temperature prediction appear to show that a linear model fits the data as the QQ Plots for each show predictions that are close to the line and the Residual vs Fitted plots show seemingly random predictions. However, for the residual vs fitted plot for our maximum temperature prediction, those residuals appear to be spread out more randomly. For both plots, we also find that the Shapiro-Wilk test for normality is not rejected implying that our residuals do not follow a normal distribution. It's important to know though that this test is highly sensitive, and considering the sample size is greater than 50 (the appropriate amount for the test), we should take the impact of this test with a grain of salt for our discussion.

Thus, I think there is sufficient evidence to state that there exists a linear relationship between the current day's temperature in predicting the next day's temperature especially in the case of predicting minimum temperatures. Although the Shapiro-Wilk test might have been violated, other measures would indicate homoscedasticity of the data.

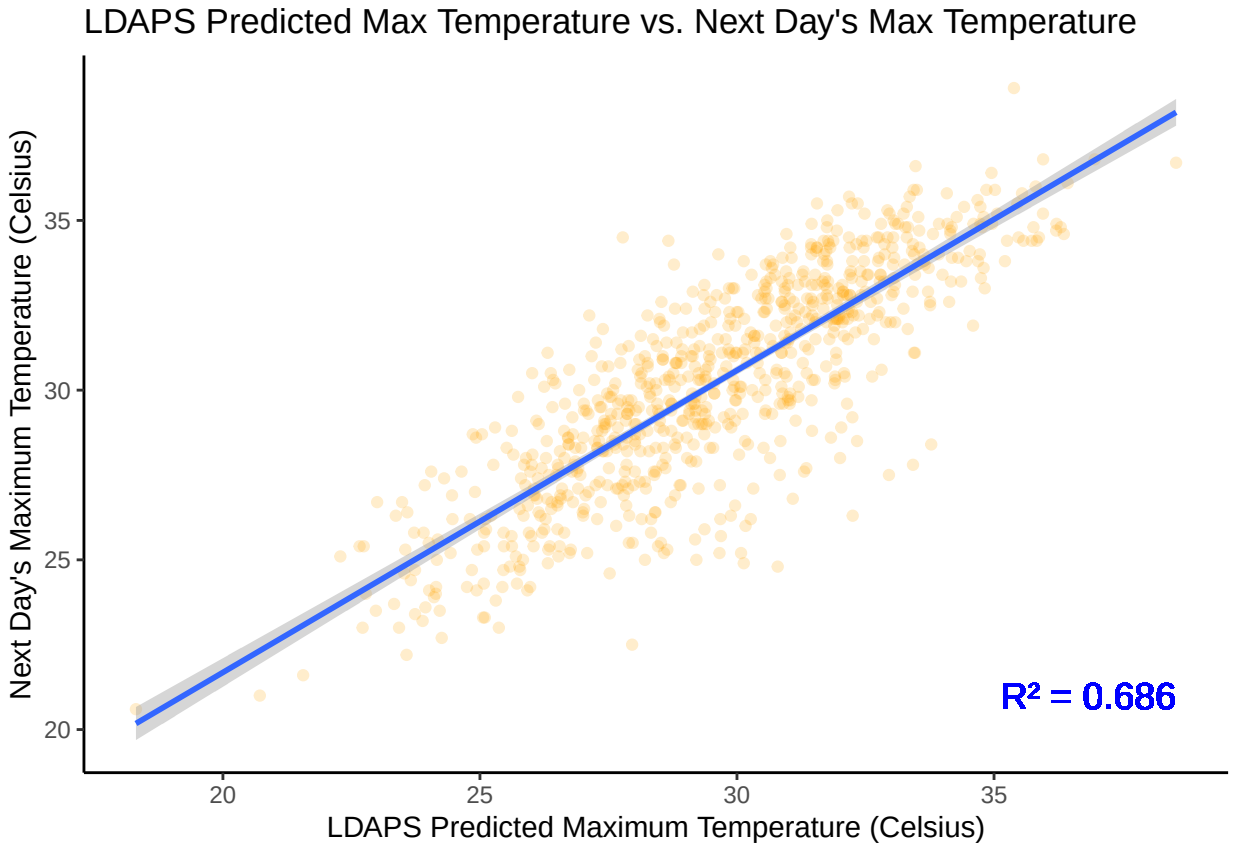
Investigation 2

The second investigation will check to see if Korea's LDAPS model does a better job predicting the next day's maximum and minimum air temperature compared to our previous model. First we will graph the LDAPS model to see if there is linear relationship with predicted temperature.

```
#Regression fits for our models
lm_max2 <- lm(Next_Tmax ~ LDAPS_Tmax_lapse, data = climate_sample)


lm_min2 <- lm(Next_Tmin ~ LDAPS_Tmin_lapse, data = climate_sample)

# Graph for Predicted temperature high's using LDAPS
climate_sample %>%
  ggplot(aes(x = LDAPS_Tmax_lapse, y = Next_Tmax)) +
  geom_point(alpha = 0.2, color = "orange") +
  geom_smooth(method = lm) +
```

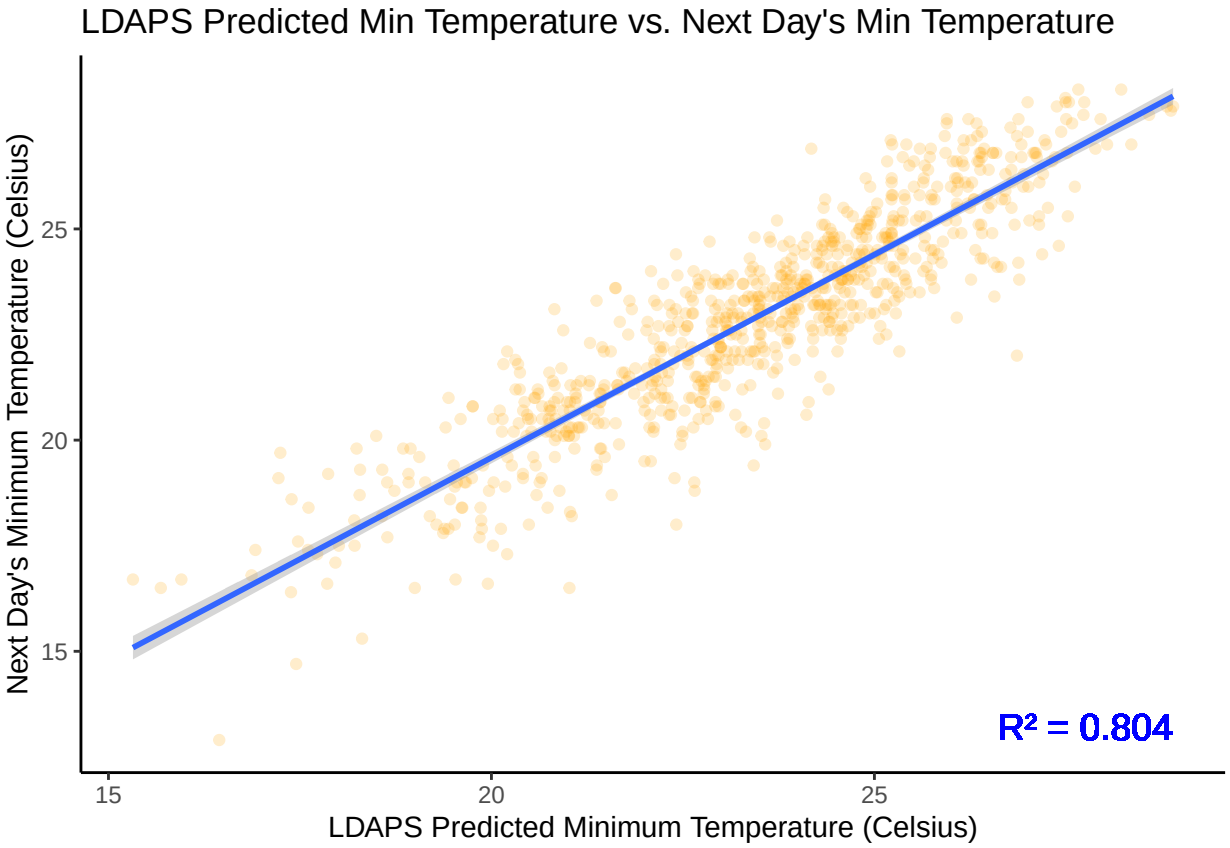
```
  geom_text(aes(x = max(LDAPS_Tmax_lapse), y = min(Next_Tmax),
                label = paste("R² =", round(summary(lm_max2)$r.squared, 3))),
            hjust = 1, vjust = 0, color = "blue", size = 5) +
  labs(title = "LDAPS Predicted Max Temperature vs. Next Day's Max Temperature",
       x = "LDAPS Predicted Maximum Temperature (Celsius)",
       y = "Next Day's Maximum Temperature (Celsius)") +
  theme_classic()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
climate_sample %>%
  ggplot(aes(x = LDAPS_Tmin_lapse, y = Next_Tmin)) +
  geom_point(alpha = 0.2, color = "orange") +
  geom_smooth(method = lm) +
  geom_text(aes(x = max(LDAPS_Tmin_lapse), y = min(Next_Tmin),
                label = paste("R² =", round(summary(lm_min2)$r.squared, 3))),
            hjust = 1, vjust = 0, color = "blue", size = 5) +
  labs(title = "LDAPS Predicted Min Temperature vs. Next Day's Min Temperature",
       x = "LDAPS Predicted Minimum Temperature (Celsius)",
       y = "Next Day's Minimum Temperature (Celsius)") +
  theme_classic()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



There appears to be a very strong linear relationship with the LDAPS future forecast model predicting the next day's temperature. It is stronger than our previous investigation as there is a R-squared value of 0.686 for the LDAPS prediction of the next day high and a R-squared value of 0.804 for the LDAPS prediction. This indicates that more variability in the next day's temperature can be explained by the regression prediction of the LDAPS model.

Residual Analysis and Homoscedasticity (Investigation 2)

```
# Looking at Homoscedasticity of LDAPS Maximum Temperature

summary(lm_max2)
```

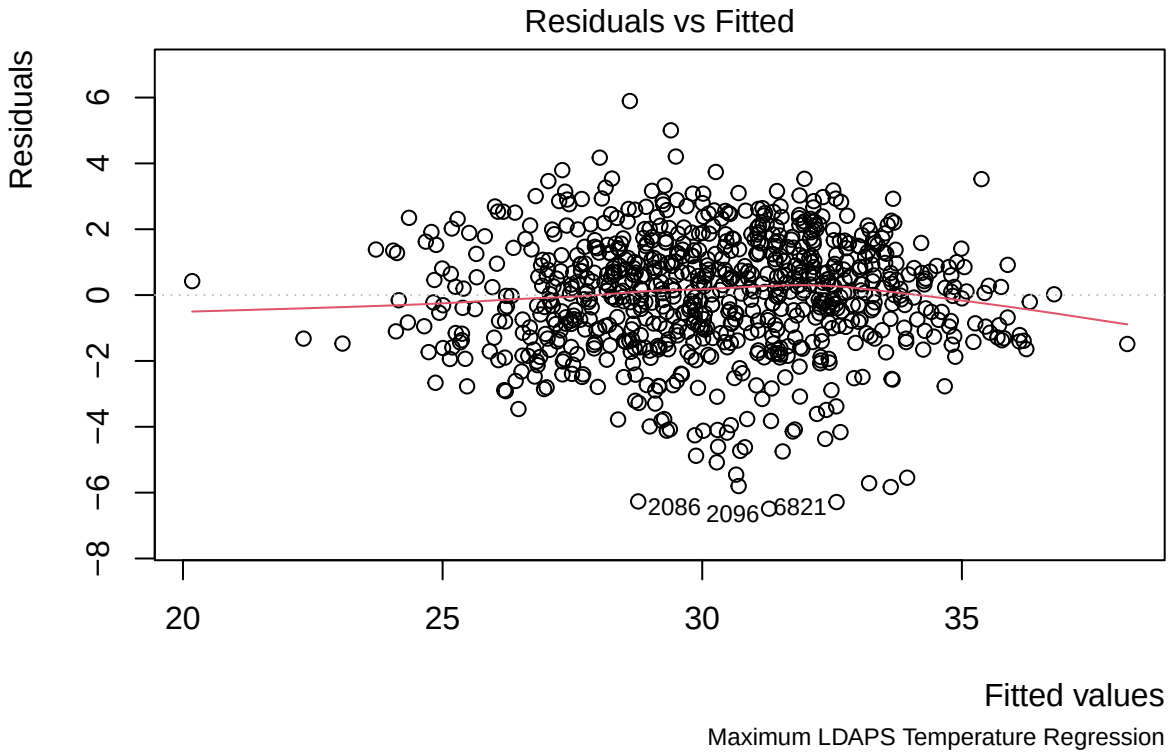
```
##
## Call:
## lm(formula = Next_Tmax ~ LDAPS_Tmax_lapse, data = climate_sample)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.4879 -1.1075  0.1403  1.2466  5.8942
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.88259    0.63311   6.133 1.36e-09 ***
## LDAPS_Tmax_lapse  0.89000    0.02129  41.800  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 1.786 on 798 degrees of freedom
## Multiple R-squared:  0.6865, Adjusted R-squared:  0.6861
## F-statistic:  1747 on 1 and 798 DF,  p-value: < 2.2e-16
```

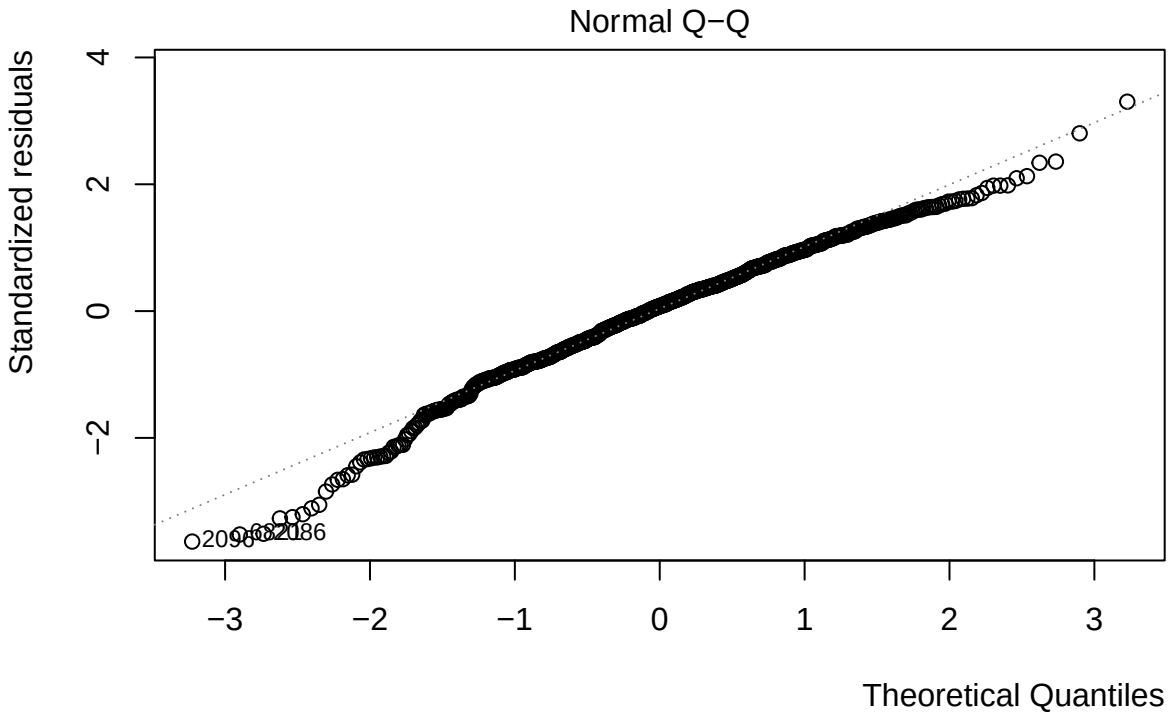
```
shapiro.test(residuals(lm_max2)) #Shapiro test for LDAPS maximum temp. regression
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(lm_max2)
## W = 0.98387, p-value = 1.032e-07
```

```
## Residual vs Fitted Plot for LDAPS max temp. regression
plot(lm_max2, which = 1,
     sub = "Maximum LDAPS Temperature Regression",
     cex.sub = 0.75,
     adj = 1)
```




```
## QQ Plot for LDAPS maximum temp. regression
plot(lm_max2, which = 2,
     sub = "Maximum LDAPS Temperature Regression",
     cex.sub = 0.75,
     adj = 1)
```



Maximum LDAPS Temperature Regression

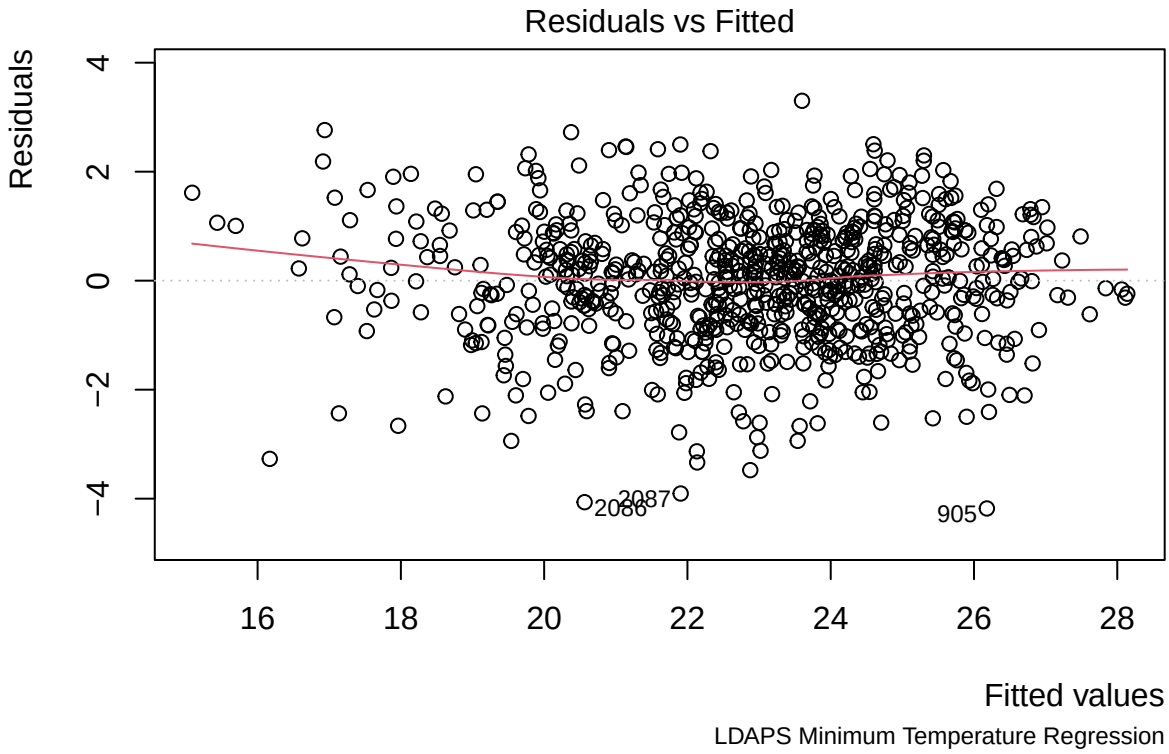
```
# Looking at Homoscedasticity of LDAPS Minimum Temperature
summary(lm_min2)
```

```
##
## Call:
## lm(formula = Next_Tmin ~ LDAPS_Tmin_lapse, data = climate_sample)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1800 -0.6886  0.0586  0.7765  3.3004
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.3627     0.3954   0.917    0.359
## LDAPS_Tmin_lapse   0.9611     0.0168  57.224   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.127 on 798 degrees of freedom
## Multiple R-squared:  0.8041, Adjusted R-squared:  0.8038
## F-statistic:  3275 on 1 and 798 DF,  p-value: < 2.2e-16
```

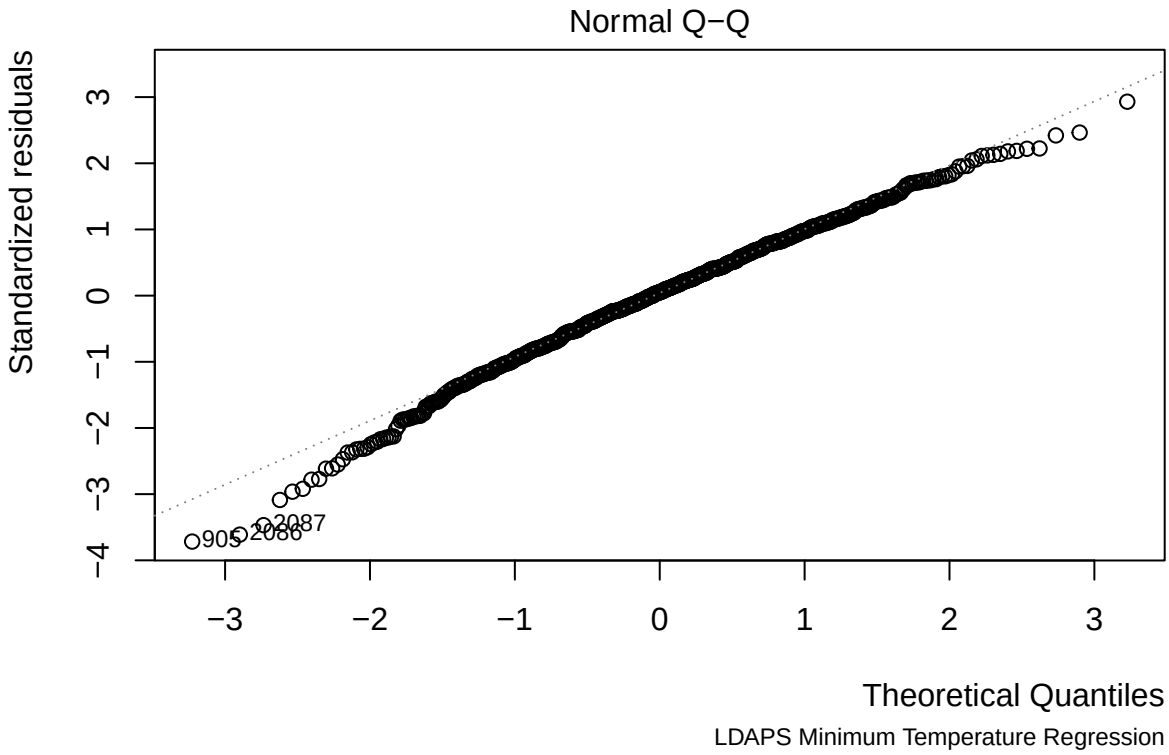
```
shapiro.test(residuals(lm_min2)) #Shapiro test for LDAPS minimum temp. regression
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(lm_min2)
## W = 0.99138, p-value = 0.0001274
```

```r
#Residual vs Fitted Plot for LDAPS minimum temp. regression
plot(lm_min2, which = 1,
     sub = "LDAPS Minimum Temperature Regression",
     cex.sub = 0.75,
     adj = 1)
```




```r
#QQ Plot for minimum temp. LDAPS regression
plot(lm_min2, which = 2,
     sub = "LDAPS Minimum Temperature Regression",
     cex.sub = 0.75,
     adj = 1)
```



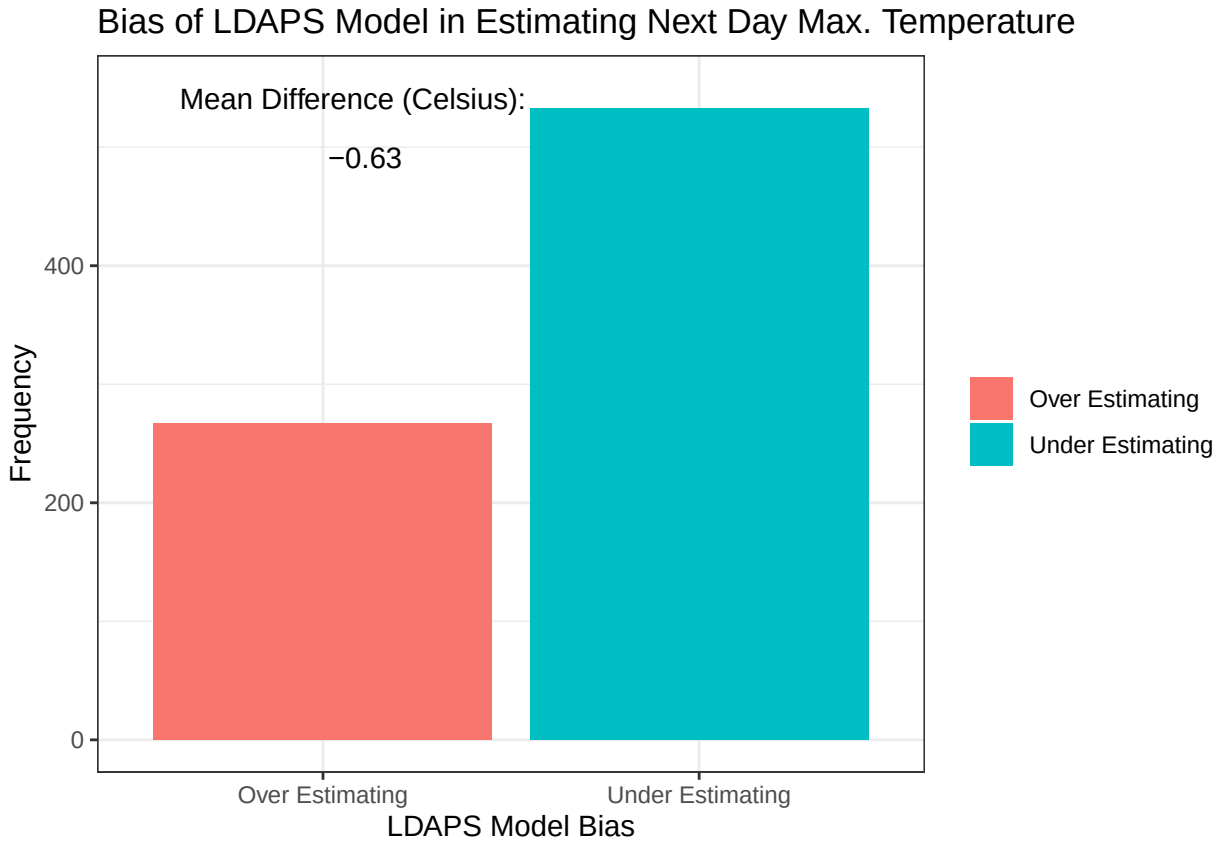
Once again, there does appear to be evidence that a linear model is the correct model for fitting the relationships between the LDAPS model of prediction. The correlating QQ plots show very strong evidence for the LDAPS prediction of next day temperature except at each of the graph's ends. It is also interesting to note that the residual vs fitted plot for minimum temperature prediction seems slightly more randomly distributed compared to its maximum temperature prediction counter part. One might argue that the opposite is true for our investigation one. Lastly, the Shapiro-Wilk test for normality once again was rejected in this investigation, but similar to before, this was likely due to it being a high sensitivity test and the sample containing more than 50 data points. Overall, there is strong evidence that a linear model is the correct form for this function regression considering the high R-squared of the LDAPS model, but this conclusion may be limited by the fact that residual analysis for homoscedasticity showed weaker results than in the previous investigation.

Bias

Another important attribute of the LDAPS model to consider is whether or not the model might be bias. An example of this would be if the LDAPS model consistently overestimated the next day's high temperature high. A simple bias estimator will be used to see if there is any detectable bias in the sample.

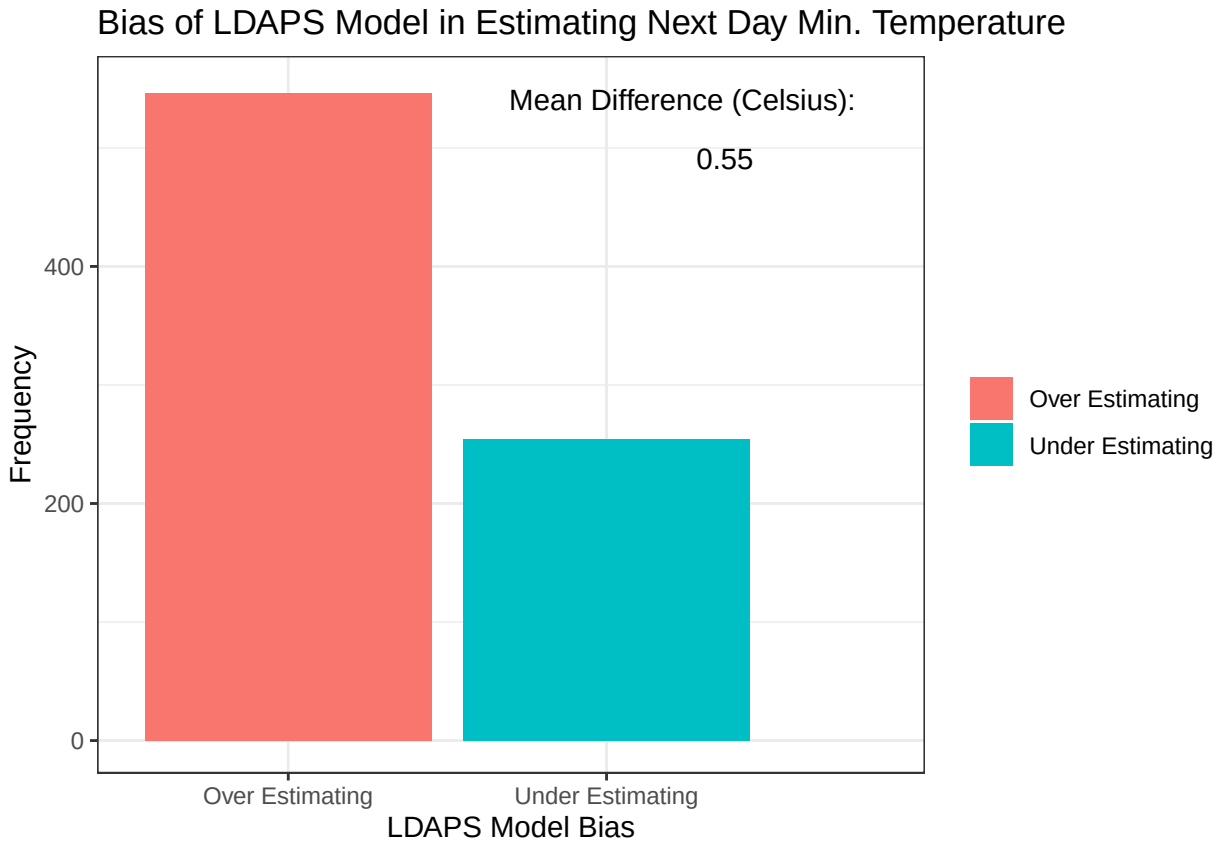
```
# Stats regarding average temperatures and average predicted temperatures

mean_high = mean(climate_sample$Next_Tmax)
mean_low =  mean(climate_sample$Next_Tmin)
mean_LDAPS_high = mean(climate_sample$LDAPS_Tmax_lapse)
mean_LDAPS_low = mean(climate_sample$LDAPS_Tmin_lapse)

LDAPS_bias_max2 = mean_LDAPS_high - mean_high
LDAPS_bias_min2 = mean_LDAPS_low - mean_low


#Graph of LDAPS Bias for Maximum Temperatures
```

```
climate_sample %>%
  mutate(LDAPS_max_bias = ifelse(LDAPS_Tmax_lapse > Next_Tmax,
                                 "Over Estimating",
                                 "Under Estimating")) %>%
  ggplot(aes(x = LDAPS_max_bias, fill = LDAPS_max_bias)) +
  geom_bar(stat = 'count') +
  annotate("text", x = 2, y = 500 ,
           label = round(LDAPS_bias_max2, digits = 2),
           vjust = 1,
           hjust = 5) +
  annotate("text", x = 2, y = 550 ,
           label = "Mean Difference (Celsius):",
           vjust = 1,
           hjust = 1.5) +
  labs(title = "Bias of LDAPS Model in Estimating Next Day Max. Temperature",
       x = "LDAPS Model Bias",
       y = "Frequency",
       fill = NULL) +
  theme_bw()
```




```
#Graph of LDAPS Bias for Minimum Temperatures
climate_sample %>%
  mutate(LDAPS_min_bias = ifelse(LDAPS_Tmin_lapse > Next_Tmin,
                                 "Over Estimating",
                                 "Under Estimating")) %>%
  ggplot(aes(x = LDAPS_min_bias, fill = LDAPS_min_bias)) +
  geom_bar(stat = 'count') +
```

```
  annotate("text", x = 3, y = 500 ,
           label = round(LDAPS_bias_min2, digits = 2),
           vjust = 1,
           hjust = 4) +
  annotate("text", x = 3, y = 550 ,
           label = "Mean Difference (Celsius):",
           vjust = 1,
           hjust = 1.2) +
  labs(title = "Bias of LDAPS Model in Estimating Next Day Min. Temperature",
       x = "LDAPS Model Bias",
       y = "Frequency",
       fill = NULL) +
  theme_bw()
```



The graphs reveal that there is detectable bias in the LDAPS model for predicting the next day's temperature. Interestingly, when predicting the next day maximum temperature, the LDAPS model often underestimates the amount by a mean of 0.55 degrees Celsius, and for predicting the next day minimum temperature, the LDAPS model often overestimates the amount by 0.55 degrees Celsius. Thus, it appears that the LDAPS model for forecasting weather does have some limitations to it. To check this idea further, we can use a t-test with an alpha level of 0.05 to determine significance of bias as we are using sample data with an unknown population variance.

```
#T-test to see if LDAPS model predicted temperatures
#are equal to true mean of temperatures

t.test(x = climate_sample$LDAPS_Tmax_lapse , y = NULL,
       alternative = "two.sided",
       mu = mean_high, paired = FALSE, var.equal = FALSE,
```

```
       conf.level = 0.95)
```

```
##
##  One Sample t-test
##
## data:  climate_sample$LDAPS_Tmax_lapse
## t = -5.9859, df = 799, p-value = 3.249e-09
## alternative hypothesis: true mean is not equal to 30.21487
## 95 percent confidence interval:
##  29.38079 29.79274
## sample estimates:
## mean of x
##  29.58676
```

```
t.test(x = climate_sample$LDAPS_Tmin_lapse , y = NULL,
       alternative = "two.sided",
       mu = mean_low, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95)
```

```
##
##  One Sample t-test
##
## data:  climate_sample$LDAPS_Tmin_lapse
## t = 6.5231, df = 799, p-value = 1.221e-10
## alternative hypothesis: true mean is not equal to 22.87463
## 95 percent confidence interval:
##  23.25729 23.58673
## sample estimates:
## mean of x
##  23.42201
```

In both t-tests, the null hypothesis states that the hypothesized mean (LDAPS model) is equal to the true mean (actual temperature). However, after running a two-sided t-test, the null hypothesis is rejected at the 0.05 alpha level of significance, leading to the conclusion that the predicted mean of the temperature using the LDAPS model is not equal to the true mean of next day temperature. Thus, there is indication of bias in the LDAPS prediction model.

Discussion

In our investigation, we aimed to determine if Korea's LDAPS Model for forecasting maximum and minimum temperature did a better job at predicting the next day's weather compared to solely using temperature as a predictor. The first investigation looked only at the day's temperature as the parameter for predicting the next day's temperature. In that investigation, we found that current temperature did a better job of predicting the minimum temperature than the maximum temperature with R-squared values of 0.676 and 0.379 respectively. While these values indicated some strength in a linear relationship between the two variables, residual analysis aimed to see if there were any blatant dicrepancies in the model. After plotting residuals, using the Shapiro-Wilk test, and using a QQ-plot, I concluded that it was possible for current temperature to have a mediocre linear relationship in predicting the next day's temperatures.

The second investigation looked at answering our research question of weather or not the LDAPS model is a better predictor of temperature. Using the same methods in the previous investigation, we found that there was a stronger relationship between using the LDAPS model prediction for the weather and the actual next day weather. The linear regression found that similar to investigation 1, predicting minimum temperature was more accurate than predicting maximum temperature as the R-squared values for this analysis were 0.804 and 0.686. However, residual analysis for homoscedasticity showed weaker results in this investigation which

may imply that a linear model using the LDAPS model as a predictor for weather might not be entirely the correct form.

The last thing analyzed in this investigation was to see if the LDAPS model had any bias in its prediction. An initial data visualization showed a general trend in overestimating low temperatures and underestimating high temperatures. Using a two-sided t-test that compared the means of the LDAPS forecast and the true forecast for weather found that the null hypothesis could be rejected at the 5% alpha level of significance. Thus, there is strong evidence that the LDAPS model is biased; however, correcting that bias might not be through a simple linear transformation.

A future investigation might try to use other LDAPS measures in the data set to see if there are any significant predictors that could account for this bias. However, not having access to the algorithm behind the LDAPS model means future investigations would have to be careful in dealing with collinearity and confounding variables when working with these predictors.

References

The original data set was taken from the UC Irvine Machine Learning Repository:

https://archive.ics.uci.edu/dataset/514/bias+correction+of+numerical+prediction+model+temperature+forecast