Question 1
Splitting the data into 2 datasets has been included in the text file: **Absent.txt (Contains All Codes)**

Question 2
We use the training dataset for exploring. On exploring we do observe outliers, correlations between variables, and no missing values in this dataset. For data collection, it has been observed that the data has only 6 days per month for every year and we had to add a year column to justify the date as we couldn't work on just a month or day. We do have to make an additional column of year in the excel sheet to better forecast and visualize the graphs.

Elaborating further, the **read.csv** function is used to reading the CSV file located in the working directory. The file must reside in the working directory for proper reading and mapping. We save it in a new vector called dataframe(data). We then check for missing values using the **colSum(is.na(data))** function. As we can see in this dataset, we don't have any missing values.

```
> #To find missing values
> colSums(is.na(data))
      month         day      season    transexp    distance    servtime         age     childen         bmi
          0           0           0           0           0           0           0           0           0
absenttime        year
          0           0
```

We have month, day and year in numeric so we use **data$Date<as.Date(with(data,paste(year,month,day,sep="-")),"%Y-%m-%d")** command to merge the segregated month ,year and day column into one date. The **as.Date** is a function which is used to convert the date into a date format which we can check via **str(data)** command. We then remove month, day and year as we have a new column that fits 3 columns in one using this command **data <- data[,-c(1,2,11)].**

The **ggcorr()** function from the GGally library is used for the correlation matrix which is given via correlation plot as shown below. It works only on numeric columns. From the correlation plot, we see that only **servtime** and **age** have a moderate positive correlation that falls in the range of 0.4 to 0.7.
Anything less than 0.4 range falls for low positive or no correlation. The correlation range is -1 to 1 as you can see. So anything below 0 will be negatively correlated.

Figure 1: Correlation lot between variables.

We use the describe function from the psych package gives us the summary statistics of all continuous variables as shown which includes mean, median, min, max, range etc. We observe that The **transexp** variable has the highest mean with high standard error and standard deviation.

```
--------------------------------------------------------------------------------------------------
transexp
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
     600        0       24    0.984    224.6    75.37      118      118      179      225      260      291
     .95
     361

lowest : 118 155 157 179 184, highest: 330 361 369 378 388
--------------------------------------------------------------------------------------------------
```

We then use the **cov(data_training[,-9])** function to give us the covariance of all variables, and how they are related to each other. A positive value indicates a positive linear relationship between two variables. The **transexp** variable has the highest covariance of 4500.915.

```
> cov(data_training[,-9])
                season     transexp     distance     servtime           age      childen          bmi  absenttime
season      1.39926544     2.692454   -0.9433389   -0.1215359   -0.05432387   0.05652755   0.01649416   0.1849416
transexp    2.69245409  4500.915189  250.8060295  -98.4530885  -93.76201169  28.83681970 -43.02608792 -22.8008653
distance   -0.94333890   250.806029  219.0733194   10.3469115  -11.82217863   1.77565109   6.19797440 -12.6039594
servtime   -0.12153589   -98.453088   10.3469115   18.4207012   18.19899833  -0.44373957   9.03105175  -0.1742905
age        -0.05432387   -93.762012  -11.8221786   18.1989983   38.93052588   0.03386477  12.95038397   0.3152003
childen     0.05652755    28.836820    1.7756511   -0.4437396    0.03386477   1.30414858  -0.85395659   0.7303923
bmi         0.01649416   -43.026088    6.1979744    9.0310518   12.95038397  -0.85395659  17.82833612  -2.1340456
absenttime  0.18494157   -22.800865  -12.6039594   -0.1742905    0.31520033   0.73039232  -2.13404563 141.2691124
```

We can know the distribution of variables with the histogram plots. The histogram graph in **Fig. 2(left)** shows the histogram of **absenttime** which is rightly skewed meaning there are larger values at the start and tend to decrease as time increases.
If skewness has a positive value the mean value is more than the median value which justifies that it is rightly skewed. We have used **hist(data_training$absenttime)** for the same **Fig. 2(right)**. We have also plotted all variables that are continuous using the using **hist.data.frame(data_training[,-9])** command and Hmisc.

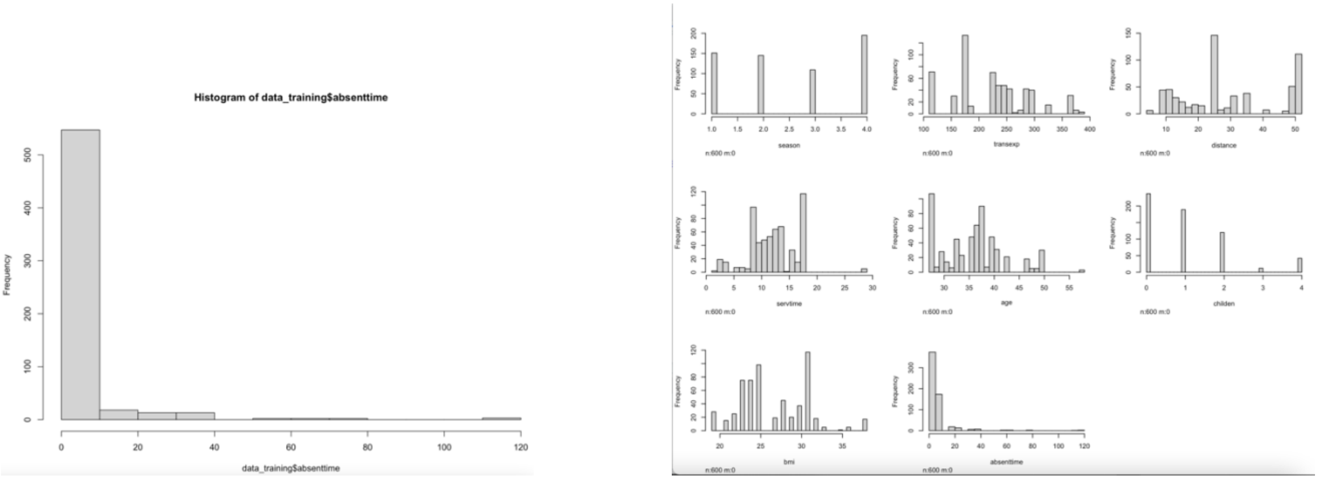


Figure 2: Absenttime (left) & All Variables(right)

We use the box plot in **Fig. 3** to check outliers in the figure below for the target variable **absenttime** . The dark line in the box is the median. The open circles are the outliers detected. The edges of the box are the interquartile range (IQR).

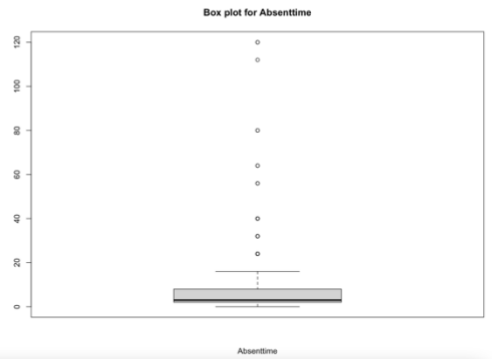


Figure 3: Box-Plot to see Outliers for AbsentTime

The stationarity is tested via the Augmented Dickey-Fuller test (ref. code in absent.txt file) using the **adf.test(data_training_ts)** function and we observe that the absent data is stationary with a p-value of 0.01 which is less than 0.05 proving the data is stationary. The p-value from the Augmented Dickey-Fuller test suggests that the data is very unlikely to give the null hypothesis, so we would rather "believe in" the alternative hypothesis.

Question 3 (a)

The best fit model we have chosen is the 2nd Arima Model with p,d,q values of (0,1,1) by seeing the RMSE, AIC, BIC value which is lower for this model. We have also checked the Neural Network Model(**Table 3**) and AutoArima(**Table 2**) models along with Arima Model(**Table 1**) to order to find the best fit model but they all have values much higher than the best fit Arima model 2 which is highlighted in Table 1.

Arima Models	(p,d,q)	AIC	BIC	AICc	RMSE
Arima 1	(1, 1, 0)	1013.3	1020.961	1013.564	58.22022
Arima 2	**(0, 1, 1)**	**980.5441**	**988.2057**	**980.8078**	**52.07339**
Arima 3	(0, 0, 1)	1398.791	1410.726	1399.075	52.08662

Table 1: Comparison 3 Arima models with its values

Auto-Arima	AIC	BIC	AICc	RMSE
(1,0,0)(0,1,0)	986.2018	991.3305	986.3308	62.90368

Table 2: Auto-Arima model with its values

Neural Network	RMSE
(1,1,2)	62.90368

Table 3: Neural Network models with its values

Question 3 (b)

The model has no trend or seasonality. The residuals mean is close to 0 and we see there is no significant correlation in the residual series from the plot. The time plot of the residuals graph shows that the variation of the residuals stays much the same across the historical data, apart from the one outlier we can see below. As a result, the residual variance can be considered constant. We can observe the same thing in the histogram residuals graph. It is suggested in the histogram as we can see that the residuals might not be normal as the right tail seems a little too long, even when we ignore the outliers. Therefore, the forecasts from this method are significantly good, but prediction intervals that are computed assuming a normal distribution may sometime be inaccurate.

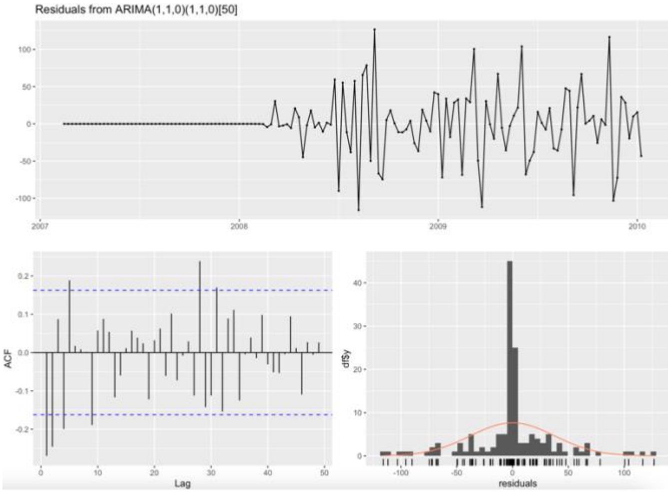
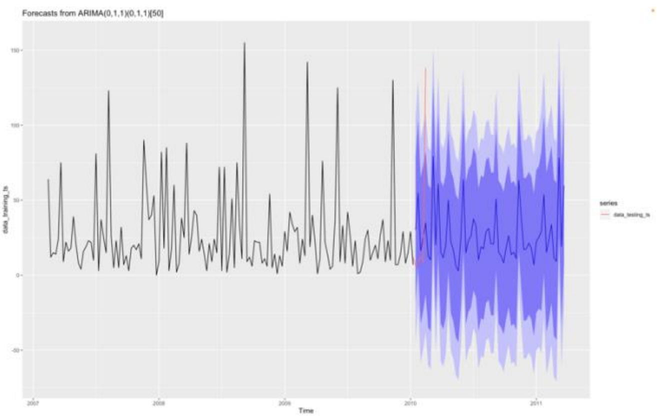


Figure 4: Residual check for ARIMA(0,1,1)

Question 3 (c)

The forecast of the **absenttime** from 2010 is shown using the Arima model with order(0,1,1). The forecast model is chosen by using the RMSE value among all the models forecasted from the data below. We can say that Arima Model has the lowest RMSE value(52.0733) and found the best fit model and has less trend and no seasonality. Hence this is the best-fitted model.




```
> #RMSE values for different models
> accuracy(forecast(nn_forecast, h = 60), data_testing_ts)
                     ME    RMSE      MAE       MPE     MAPE      MASE        ACF1 Theil's U
Training set 0.01855551 27.4798 17.84274 -188.8268 213.3204 0.6621193 -0.01481859        NA
Test set     5.60639253 54.2851 39.79101 -190.7104 223.6071 1.4765895 -0.04336097 0.9370102
> accuracy(forecast(autoarima_1, h = 60), data_testing_ts)
                     ME     RMSE      MAE       MPE     MAPE      MASE        ACF1 Theil's U
Training set -0.7423293 32.67785 17.25653       Inf      Inf 0.6403659  0.01677819        NA
Test set      5.4262535 62.90368 46.56933 -242.8161 280.4941 1.7281237 -0.04664728  1.062529
> accuracy(forecast(arima1, h = 60), data_testing_ts)
                      ME     RMSE      MAE       MPE     MAPE      MASE        ACF1 Theil's U
Training set -0.09089384 37.58988 21.99418      -Inf      Inf 0.8161736 -0.26956841        NA
Test set     19.27331463 58.22022 34.98684 -74.66341 142.7699 1.2983132  0.02659837  1.011899
> accuracy(forecast(arima2, h = 60), data_testing_ts) #bestfit
                    ME     RMSE      MAE       MPE     MAPE      MASE         ACF1 Theil's U
Training set -1.322723 29.21352 15.75904      -Inf      Inf 0.5847964 -0.219863559        NA
Test set      2.699065 52.07339 38.54374 -208.6543 238.5404 1.4303050 -0.004612419 0.9633478
> accuracy(forecast(arima3, h = 60), data_testing_ts)
                     ME     RMSE      MAE       MPE     MAPE      MASE          ACF1 Theil's U
Training set 0.05642614 28.32370 19.55964      -Inf      Inf 0.7258311 -0.0005963964        NA
Test set     7.20310476 52.08662 36.88697 -166.7746 198.7239 1.3688246 -0.0531125858 0.8917782
```

Figure 5: Forecasting ARIMA(0,1,1)

Question 3 (d)

In consideration of all the observations the weakness of the analysis is the time series of data is fully dependent on the past data and the forecast data can change based on different values, parameters and circumstances. Many a time with the dataset the forecast part tends to give a straight line so we put seasonal parameters to get a better forecasting graph and visualization which was also observed.