Time Series Forecast

Uche Kalu

4/15/2022

The Goal of this Analysis is to predict the future as accurately as possible based on current information available .

When forecasting time series data, the aim is to estimate how the sequence of observation will continue into the future

```
library(modeltime)
library(tidymodels)
library(tidyverse)
library(timetk)
library(lubridate)
library(glmnet)
```

DATA

The Dataset we will be using is **bike_sharing_daily**. It is a built-in dataset from timetk package.

We will be using only two variables from the dataset - dteday and cnt columns. cnt variable is the response variable

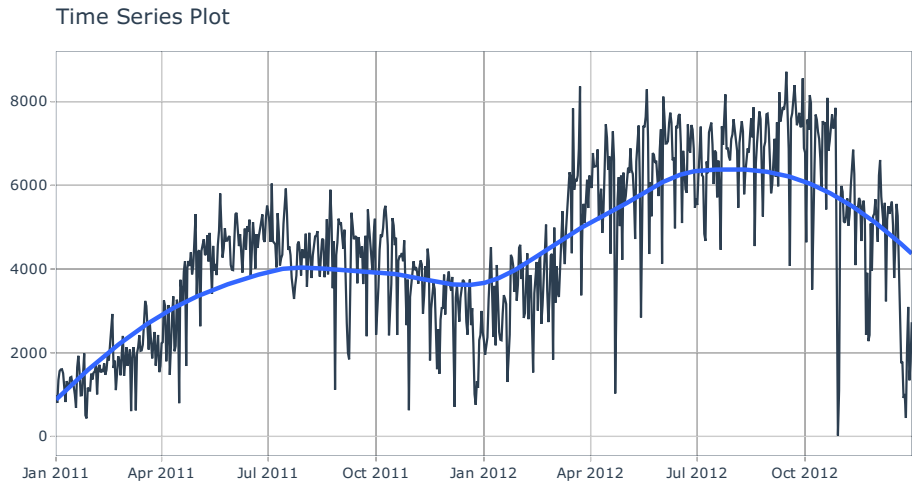
```
View(bike_sharing_daily)

mydata <- bike_sharing_daily %>%
  select(dteday, cnt)

mydata
```

This displays an interactive time series chart using the stated variables

```
mydata %>% plot_time_series(dteday, cnt)
```



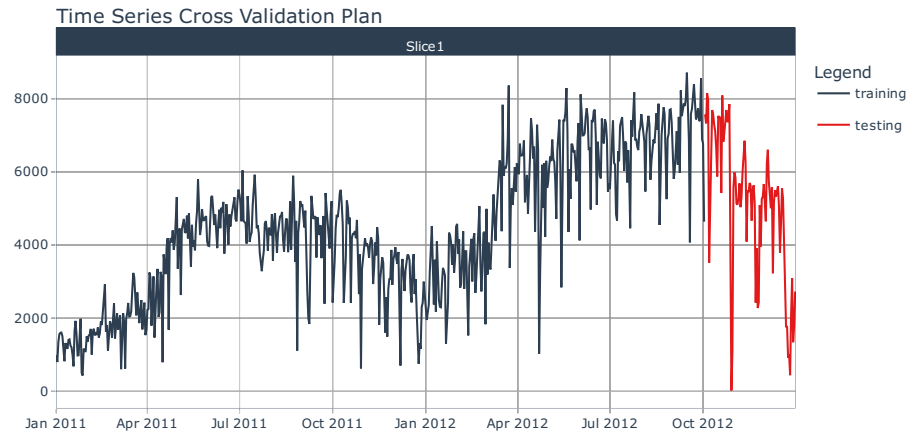
TRAIN / TEST SPLITS

We will split the dataset into training and testing data by using the last 3 months in the data as testing set

```
splits <- time_series_split(mydata, assess = "3 months",
                            cumulative = TRUE)
```

Visualizing the training and testing data

```
splits %>%
  tk_time_series_cv_plan() %>%
  plot_time_series_cv_plan(dteday, cnt)
```



Creating FORECAST using Auto-Arima model, Prophet model and GLM machine learning algorithm

```
mymodel <- arima_reg() %>%
  set_engine("auto_arima") %>%
  fit(cnt ~ dteday, training(splits))
```

AUTO ARIMA model (this is an auto-regressive forecasting algorithm)

```
ourmodel <- prophet_reg(seasonality_yearly = TRUE) %>%
  set_engine("prophet") %>%
  fit(cnt ~ dteday, training(splits))
```

Facebook Prophet model

```
yrmodel <- linear_reg(penalty = 0.01) %>%
  set_engine("glmnet") %>%
  fit(cnt ~ wday(dteday, label = TRUE) +
        month(dteday, label = TRUE) +
        as.numeric(dteday), training(splits))
```

GLM Machine Learning algorithm

Using MODELTIME package to organize and compare our models

Create a Modeltime Table (this helps to organize our models)

```
hismodel <- modeltime_table(mymodel, ourmodel, yrmodel)
hismodel
```

```
## # Modeltime Table
## # A tibble: 3 x 3
##   .model_id .model   .model_desc
##       <int> <list>   <chr>
## 1         1 <fit[+]> ARIMA(0,1,3) WITH DRIFT
## 2         2 <fit[+]> PROPHET
## 3         3 <fit[+]> GLMNET
```

Calibration.

Here we calculate predictions and residuals(error) for the test data

```
mycalib <- hismodel %>%
  modeltime_calibrate(testing(splits))
mycalib
```

```
## # Modeltime Table
## # A tibble: 3 x 5
##   .model_id .model   .model_desc             .type .calibration_data
##       <int> <list>   <chr>                   <chr> <list>
## 1         1 <fit[+]> ARIMA(0,1,3) WITH DRIFT Test  <tibble [90 x 4]>
## 2         2 <fit[+]> PROPHET                 Test  <tibble [90 x 4]>
## 3         3 <fit[+]> GLMNET                  Test  <tibble [90 x 4]>
```

Viewing Accuracy.

Lets see the accuracy from our testing dataset predictions

ae - Mean Absolute Error (is the average error aggregated for across d prediction. d smaller d value, d better); we see that ARIMA model is d worst

rsq - R-Squared value (Its value is from 0 to 1. the higher d value, d better): we see PROPHET model is d best

```
mycalib %>% modeltime_accuracy()
```

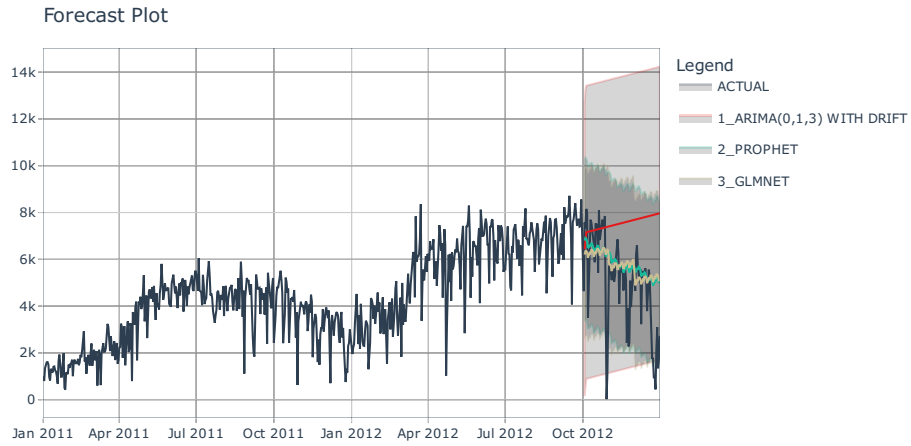
```
## # A tibble: 3 x 9
##   .model_id .model_desc             .type   mae  mape  mase smape  rmse   rsq
##       <int> <chr>                   <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1         1 ARIMA(0,1,3) WITH DRIFT Test  2540.  475.  2.74  46.0 3188. 0.390
## 2         2 PROPHET                 Test  1220.  365.  1.32  28.7 1763. 0.437
## 3         3 GLMNET                  Test  1289.  373.  1.39  29.7 1835. 0.247
```

Visualization of the Testing data

Displays an interactive time series chart using the three models on training and testing data

We see that ARIMA model is not following d time series trend unlike the other models so you can disable ARIMA in d legend bar to remove it from the chart

```
mycalib %>%
  modeltime_forecast(new_data = testing(splits),
                     actual_data = mydata) %>%
  plot_modeltime_forecast()
```



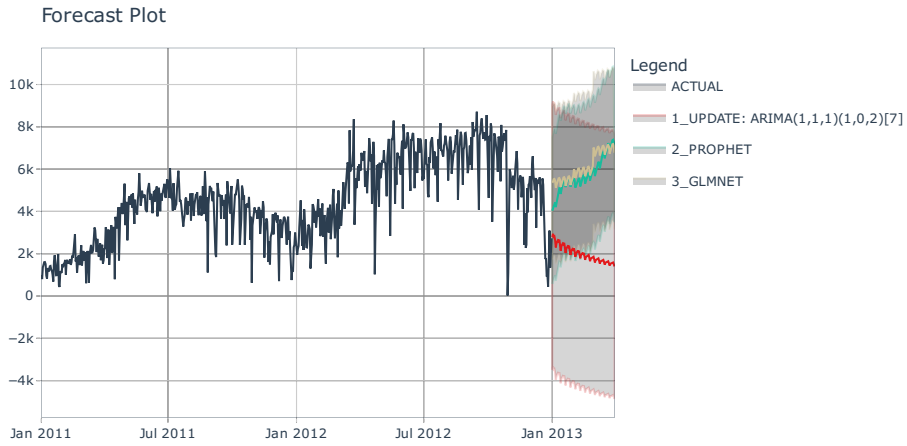
Forecast The Future

We will forecast for the next 3 months

We re-train the three models on the full dataset (not the training data) so we get the most accurate predictions in the future

```
myforecast <- mycalib %>%
  modeltime_refit(mydata) %>%
  modeltime_forecast(h = "3 months", actual_data = mydata)
```

```
myforecast %>%
  plot_modeltime_forecast()
```


Forecast Plot
10k
8k
6k
4k
2k
0
−2k
−4k
Jan 2011
Jul 2011
Jan 2012
Jul 2012
Jan 2013
Legend
ACTUAL
1_UPDATE: ARIMA(1,1,1)(1,0,2)[7]
2_PROPHET
3_GLMNET