

TFT-Copy1

July 20, 2023

```
[19]: library(zoo)
library(tidyverse)
library(tsibble)
library(remotes)
library(dplyr)
library(caret)
library(tidymodels)
library(tft)
library(lubridate)
library(timetk)
library(tibbletime)
library(luz)
library(torch)
set.seed(565)

[20]: options(torch.threshold_call_gc = 6000)

[21]: #read comodity prices and convert to xts object
com_prices_data = read.csv("commodity-prices-2016.csv")

com_prices_data = select(com_prices_data, Date, #Crude.Oil.
  ↪petroleum,Aluminum,Bananas,Barley,Beef,Coal,Cocoa.beans,Coffee.Other.Mild.
  ↪Arabicas,Coffee.Robusta,Rapeseed.oil,Copper,Cotton,Fishmeal,Groundnuts.
  ↪peanuts,Hides,Lamb,Lead,Soft.Logs,Hard.Logs,Maize.corn,Olive.
  ↪Oil,Oranges,Palm.oil,Poultry.chicken,Rice,Rubber,Fish.salmon,Hard.
  ↪Sawnwood,Soft.Sawnwood,Shrimp,Soybean.Meal,Soybean.Oil,Soybeans,Sunflower.
  ↪oil,Tea,Tin,Uranium,
  Wheat)

com_prices_data$Date = as.Date(com_prices_data$Date, format = "%Y-%m-%d")

[22]: require(curl) # To load from url for data.table's fread
require(data.table) # High performance data frame
require(zoo) # High performance data frame
com_prices_data2 = as.data.table(com_prices_data)
com_prices_data2[, Day := as.Date(Date)]
setkey(com_prices_data2, Day)
```

```

start = com_prices_data2$Date[1]
end = tail(com_prices_data2$Date, 1)

date.indexes = seq(
  from = start,
  to = end,
  by = "days"
)

com_prices_data_daily = com_prices_data2[J(date.indexes), roll = 31]

```

```

[23]: com_prices_data3 <- com_prices_data_daily %>%
  mutate(Week = lubridate::floor_date(Day, unit = "week")) %>%
  group_by(Week) %>%
  summarise(across(everything(), .fns = ~mean(.x, na.rm = TRUE)), .groups =
    ↪ "drop") %>%
  pivot_longer(c(
    #Crude.Oil.petroleum,Aluminum,Bananas,Barley,Beef,Coal,Cocoa.beans,Coffee.Other.
    ↪Mild.Arabicas,Coffee.Robusta,Rapeseed.oil,Copper,Cotton,Fishmeal,Groundnuts.
    ↪peanuts,Hides,Lamb,Lead,Soft.Logs,Hard.Logs,Maize.corn,Olive.
    ↪Oil,Oranges,Palm.oil,Poultry.chicken,Rice,Rubber,Fish.salmon,Hard.
    ↪Sawnwood,Soft.Sawnwood,Shrimp,Soybean.Meal,Soybean.Oil,Soybeans,Sunflower.
    ↪oil,Tea,Tin,Uranium,
    Wheat)) %>%
  rename("Price" = "value", "Commodity" = "name") %>%
  group_by(Week, Commodity) %>%
  summarise(across(everything(), .fns = ~mean(.x, na.rm = TRUE)), .groups =
    ↪ "drop")
tt_p = com_prices_data3

```

```

[24]: tt = tt_p[,-(3:4)]
colnames(tt)[1] = "date"

```

```

[25]: last_date <- max(tt$date)
train <- tt %>% filter(date <= (last_date - lubridate::weeks(48)))
valid <- tt %>% filter(date > (last_date - lubridate::weeks(48)),
  date <= (last_date - lubridate::weeks(12)))
test <- tt %>% filter(date > (last_date - lubridate::weeks(12)))

rec <- recipe(Price ~ ., data = train) %>%
  step_mutate(
    date_time_since_begining = as.numeric(difftime(
      time1 = date,
      time2 = lubridate::ymd(min(tt$date)),
      units = "weeks"
    )),
    date_week = as.factor(lubridate::week(date)),

```

```

    date_month = as.factor(lubridate::month(date)),
    date_wday = as.factor(lubridate::wday(date))
  ) %>%
  step_normalize(all_numeric_predictors())

```

```

[26]: spec <- tft_dataset_spec(rec, train) %>%
  spec_covariate_index(date) %>%
  spec_covariate_key(Commodity) %>%
  spec_covariate_known(starts_with("date_")) %>%
  spec_time_splits(lookback = 5*35, horizon = 12)

```

```

[27]: spec <- prep(spec)
spec

```

A `<prepared_tft_dataset_spec>` with:

lookback = 175 and horizon = 12.
 The number of possible slices is 1,650

Covariates:
``index``: date
``keys``: Commodity
``static``:
``known``: date_time_since_begining, date_week, date_month, and date_wday
``unknown``:
 Variables that are not specified in other types are considered ``unknown``.

Call ``transform()`` to apply this spec to a different dataset.

```

[28]: model <- temporal_fusion_transformer(
  spec,
  hidden_state_size = 8,
  learn_rate = 1e-3,
  #dropout = 0.5,
  num_attention_heads = 1,
  num_lstm_layers = 1
)

```

```

[29]: fitted <- model %>%
  fit(
    transform(spec),
    valid_data = transform(spec, new_data = valid),
    epochs = 3,
    callbacks = list(
      luz::luz_callback_keep_best_model(monitor = "valid_loss"),
      luz::luz_callback_early_stopping(
        monitor = "valid_loss",

```

```

        patience = 5,
        min_delta = 0.001
    )
),
verbose = TRUE,
dataloader_options = list(batch_size = 32, num_workers = 4)
)

```

Epoch 1/3

Train metrics: Loss: 0.364 - q10: 0.1986 - q50: 0.4283 - q90: 0.465

Valid metrics: Loss: 0.1327 - q10: 0.1142 - q50: 0.1897 - q90: 0.0941

Epoch 2/3

Train metrics: Loss: 0.2342 - q10: 0.1067 - q50: 0.313 - q90: 0.2827

Valid metrics: Loss: 0.0861 - q10: 0.0747 - q50: 0.1279 - q90: 0.0558

Epoch 3/3

Train metrics: Loss: 0.1911 - q10: 0.0956 - q50: 0.2603 - q90: 0.2175

Valid metrics: Loss: 0.1558 - q10: 0.0431 - q50: 0.3349 - q90: 0.0893

```

[30]: fitted %>%
  luz::evaluate(
    transform(spec, new_data = test, past_data = bind_rows(train, valid)),
    metrics = list(luz_metric_rmse(), luz_metric_mae())
  )

```

A `luz_module_evaluation`

Results

loss: 0.1523

rmse: 1.2303

mae: 1.9638

```

[31]: fitted

```

A `luz_module_fitted`

Time

- Total time: 10m 26.9s
- Avg time per training epoch: 3m 11.3s

Results

Metrics observed in the last epoch.

Training:

loss: 0.1911

q10: 0.0956

q50: 0.2603

q90: 0.2175

Model

An ``nn_module`` containing 9,852 parameters.

Modules

- preprocessing: `<preprocessing>` #1,112 parameters
- context: `<static_context>` #1,709 parameters
- temporal_selection: `<selection>` #4,693 parameters
- locality_enhancement: `<locality_enhancement_layer>` #1,472 parameters
- temporal_attn: `<nn_module>` #310 parameters
- position_wise: `<nn_module>` #528 parameters
- output_layer: `<nn_module>` #27 parameters

Parameters

- .check: Float [1:1]

```
[32]: forecasts <- generics::forecast(fitted, past_data = bind_rows(train, valid))
      as.data.frame(forecasts)
```

A data.frame: 12 × 5

	date <date>	Commodity <chr>	.pred_lower <dbl>	.pred <dbl>	.pred_upper <dbl>
	2015-11-15	Wheat	144.7985	204.6912	215.4223
	2015-11-22	Wheat	146.1455	199.8800	217.9047
	2015-11-29	Wheat	142.8414	196.7768	218.2290
	2015-12-06	Wheat	143.5570	201.6434	217.0840
	2015-12-13	Wheat	146.8060	202.9227	216.1608
	2015-12-20	Wheat	143.9311	201.7064	216.6562
	2015-12-27	Wheat	143.5825	197.1212	218.7276
	2016-01-03	Wheat	143.4986	196.3618	218.8069
	2016-01-10	Wheat	147.6533	197.7174	218.2329
	2016-01-17	Wheat	142.8465	195.0666	219.6921
	2016-01-24	Wheat	144.1247	200.9608	217.1580
	2016-01-31	Wheat	144.0747	194.7415	219.4793

```
[33]: options(repr.plot.width = 10, repr.plot.height = 10)
      tt %>%
        filter(date > lubridate::ymd("2015-01-01")) %>%
        full_join(forecasts) %>%
        #filter(Commodity == "Copper") %>%
        ggplot(aes(x = date, y = Price)) +
        geom_line() +
        geom_line(aes(y = .pred), color = "green") +
        geom_ribbon(aes(ymin = .pred_lower, ymax = .pred_upper), alpha = 0.3)
        #facet_wrap(~Commodity)
        #theme(text = element_text(size = 10), element_line(size = 1))
```

Joining with ``by = join_by(date, Commodity)``

Warning message:

"Removed 45 rows containing missing values (``geom_line()``)."

