Time Series Model for the Philippine Stock Exchange Index (PSEI)

Using a range versus point forecast to manage investor expectations

Introduction

While it is a common notion that stock prices cannot be predicted, it is often still a necessary exercise for asset management. Targets are often based on the use of a long-run multiple (P/E) combined with earnings growth expectations. However, even with that methodology, downwards revisions are often done, and no one gets it exactly right consistently.

The output of the resulting time series model, as well as from other methods, can be used to create a range of predictions to better manage expectations of clients.

Data Inputs and Preparation

We extract the index data from Yahoo Finance using the quantmod and xts libraries and getSymbols() function in R. The result initially has several columns:

```
> head(PSEI.PS)
           PSEI.PS.Open PSEI.PS.High PSEI.PS.Low PSEI.PS.Close
1987-01-01       439.53       524.58      439.53        521.64
1987-02-01       521.64       536.51      496.53        513.97
1987-03-01       500.33       511.22      447.03        511.22
1987-04-01       505.37       572.59      505.37        566.22
1987-05-01       566.22       609.86      551.97        609.86
1987-06-01       614.08       944.83      614.08        918.41
           PSEI.PS.Volume PSEI.PS.Adjusted
1987-01-01              0           521.64
1987-02-01              0           513.97
1987-03-01              0           511.22
1987-04-01              0           566.22
1987-05-01              0           609.86
1987-06-01              0           918.41
```

We extract the closing prices, omit N/As, and ensure we have sufficient data points. We cut the data to May 2019 (389 months' worth) and reserve the rest (15 months' worth) to compare with our forecasts later.[1]

```
> head(psei)
           PSEI.PS.Close
1987-01-01        521.64
1987-02-01        513.97
1987-03-01        511.22
1987-04-01        566.22
1987-05-01        609.86
1987-06-01        918.41
> length(psei)
[1] 403
> length(psei_train)
[1] 389
```

[1] I initially used daily data to create the time series model. However, none of the models passed the diagnostics. In particular, the residuals would still exhibit correlation (specifically, at lag 8). I hypothesized it may be because there is too much noise in daily data and proceeded to model monthly data. This is also consistent with industry practice that index level targets are normally submitted as annual predictions and at most monthly-level predictions.

Methodology

To model a time series using ARIMA, the time series first must be stationary, i.e. constant mean and variance over time. We initially perform visual inspection and if we see a trend, then it is likely not stationary. Statistically, we can use the Dickey-Fuller test whose null hypothesis is that a unit root is present.

The Dickey-Fuller Test starts out with the regression form:

$$y_t = \phi y_{t-1} + \epsilon_t \ (1)$$

A unit root exists if $|\phi| = 1$. This regression is transformed to the first difference:

$$y_t - y_{t-1} = \phi y_{t-1} + \epsilon_t - y_{t-1}$$

$$\Delta y_t = (\phi - 1) y_{t-1} + \epsilon_t$$

$$\Delta y_t = \beta y_{t-1} + \epsilon_t \ (2)$$

The test statistic is then the t-statistic of regression equation (2), i.e. $\frac{\beta}{SE(\beta)}$.

$H_0: \beta = 0, i.e. \phi = 1, unit\ root\ exists, non-stationary$
$H_1: \beta < 0, i.e. \phi < 1, stationary$

As such, we expect to see high negative values for the Dickey Fuller test statistic in order to reject the null hypothesis that a unit root exists.

To correct for non-stationarity, we use differencing. Specifically, we effectively get the return of our stock prices using diff(log(series)). The final differencing level is the "d" in the ARIMA (p,d,q) model. This is consistent with the suggestion: "for data with a curved upward trend accompanied by increasing variance, you should consider transforming the series with either a logarithm or a square root."[2]

Once the series is adjusted for non-stationarity, we can use the ACF (autocorrelation function) and PACF (partial autocorrelation function) to dictate the MA and AR levels.

The ACF is the correlation between a time series and different lags of itself (e.g. correlation of x(t) and x(t-1), x(t-2)).[3] The PACF is the partial correlation coefficients between the series and lags of itself. The partial correlation is the amount of correlation between the series and "a lag of itself that is not explained by correlations at all lower-order lags."[4]

The PACF plot is used to identify the order of the AR(p) model.[5]

$$x_t = \beta_0 + \beta_1 x_{t-1} + \beta_2 x_{t-2} + \cdots + \beta_2 x_{t-p} + \epsilon_t$$

[2] https://online.stat.psu.edu/stat510/lesson/3/3.1
[3] https://people.duke.edu/~rnau/411arim3.htm
[4] Ibid.
[5] https://online.stat.psu.edu/stat501/lesson/14/14.1

Where p is determined as the last lag point at which the PACF is not significantly different from zero, or where it begins to taper off to zero. The AR(q) model indicates a time series value is dependent on its previous value/s, i.e. there is still significant correlation at order q not explained by lower-order lags.

The ACF plot is used to identify the order of the MA(q) model. [6]

$$x_t = \mu + w_t + \theta_1 w_{t-1} + \theta_2 w_{t-2} + \cdots + \theta_2 w_{t-q} + \epsilon_t$$

Where q is determined as the lag point at which the ACF is not significantly different from zero, or where it begins to taper off to zero, and $w_t \sim N(0, \sigma_w^2)$. The MA(q) model indicates a time series value is dependent on previous error term/s.

Combining these into one model ARIMA(p,d,q):,

$$x_t = \beta_0 + \beta_1 x_{t-1} + \beta_2 x_{t-2} + \cdots + \beta_2 x_{t-p} + w_t + \theta_1 w_{t-1} + \theta_2 w_{t-2} + \cdots + \theta_2 w_{t-q} + \epsilon_t$$

From the possible models, the best model can be determined using the AIC (Akaike Information Criterion), where the higher it is (more negative), the better the model.

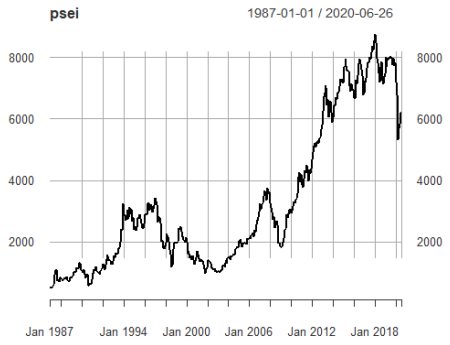
$$AIC = -2\log(L) + 2k$$

Where L is the likelihood and k is the number of parameters. The AIC is only used to compare with other models, not to check how good the model itself is. As such, we follow this up by checking the actual selected "best" model using a coefficient test. After which, we check the residuals.

Apart from visual inspection of the ACF plot of the residuals, where the ideal case is that it is zero for all non-zero lags, the Ljung-Box test is used to check if the residuals are white noise. The statistic makes use of the accumulated autocorrelations up to the specified lag. As such, a high statistic (low p-value) indicates something is wrong with the model and we must specify another one.[7] Once the final model is decided on, we make our n-ahead forecasts and compare this with actual data.

Results

The initial plot shows there is a clear trend and the ADF test confirms that this series of PSEI closing prices is not stationary.



[6] https://online.stat.psu.edu/stat510/lesson/2/2.1

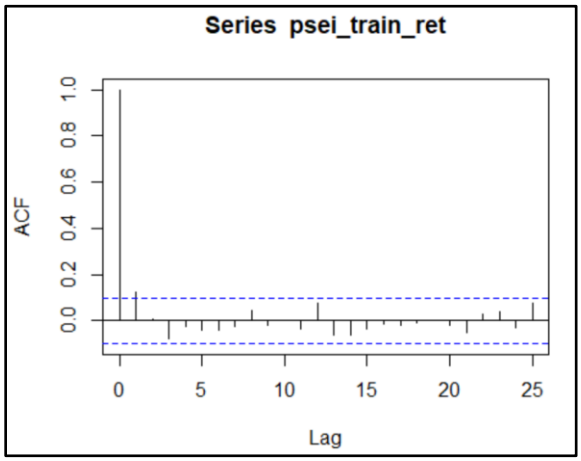
[7] https://online.stat.psu.edu/stat510/lesson/3/3.2

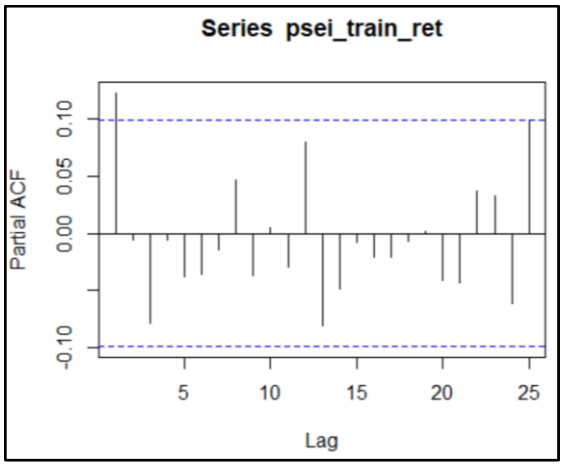
```
> adf.test(psei_train,k=0,alternative="stationary")

        Augmented Dickey-Fuller Test

data:  psei_train
Dickey-Fuller = -1.3438, Lag order = 0, p-value = 0.8543
alternative hypothesis: stationary
```

After de-trending and cutting the data up to May 2019 only (reserve as testing set to compare forecasts later), the plot shows no trend and ADF test confirms the series is stationary. The transformation performed was diff(log()), or effectively $x_{new} = \ln\left(\frac{x_t}{x_{t-1}}\right) \cong \frac{x_t}{x_{t-1}} - 1$ is the return series.

```
> adf.test(psei_train_ret,k=0,alternative="stationary")

        Augmented Dickey-Fuller Test

data:  psei_train_ret
Dickey-Fuller = -17.32, Lag order = 0, p-value = 0.01
alternative hypothesis: stationary
```

The ACF plot indicates possible MA level of differenced data at 1, while the PACF indicates possible AR level of differenced data at 1.





We loop for several combinations to find the best possible model. We extract the AIC of fitted models for AR lag orders 0 to 1 and MA lag orders 0 to 1. The matrix below has the AR lag-orders for the rows (minus 1) and MA lag-orders for the columns (minus 1).

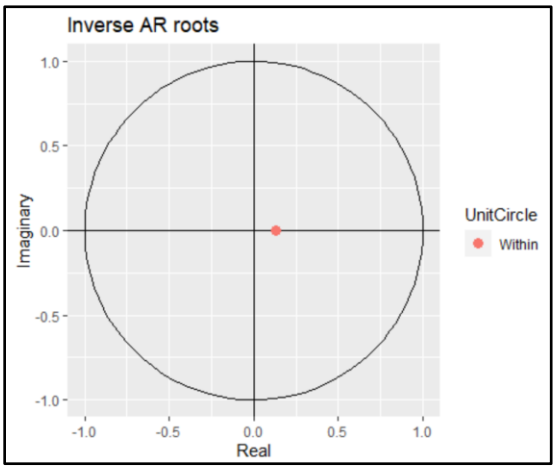
```
> aic
           [,1]      [,2]
[1,] -844.2590 -848.6294
[2,] -848.8475 -846.8475
```

The lowest AIC level is -848.8475 or ARIMA(1,1,0)

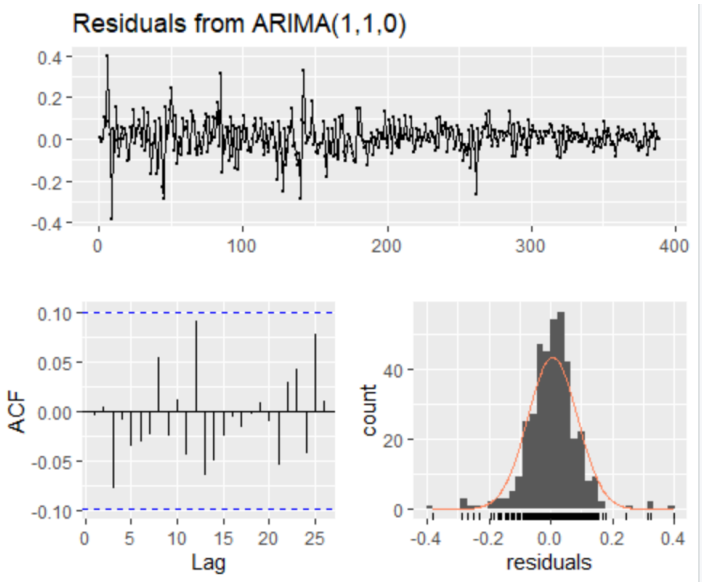
Diagnostics

The coefficient test indicates that the coefficient for AR1 is indeed significant and below 1.

```
> fit <- arima(log(psei_train), order=c(1, 1, 0))
> coeftest(fit)

z test of coefficients:

    Estimate Std. Error z value Pr(>|z|)
ar1 0.129596   0.050266  2.5782 0.009931 **
---
```



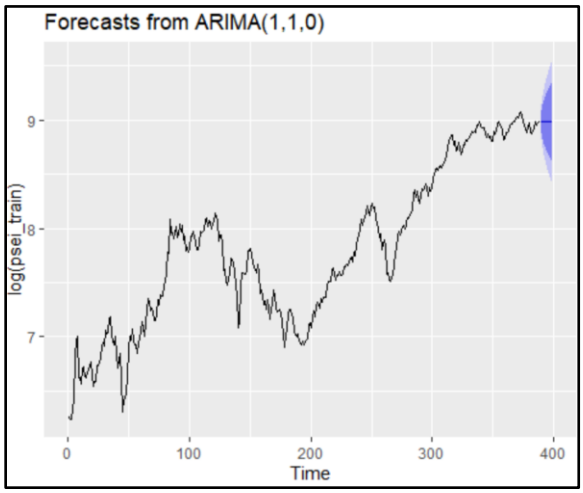
The ACF plot and the Ljung-Box test indicates the residuals are white noise. We also don't observe time-varying variance in the plots, so we do not have to model ARCH.



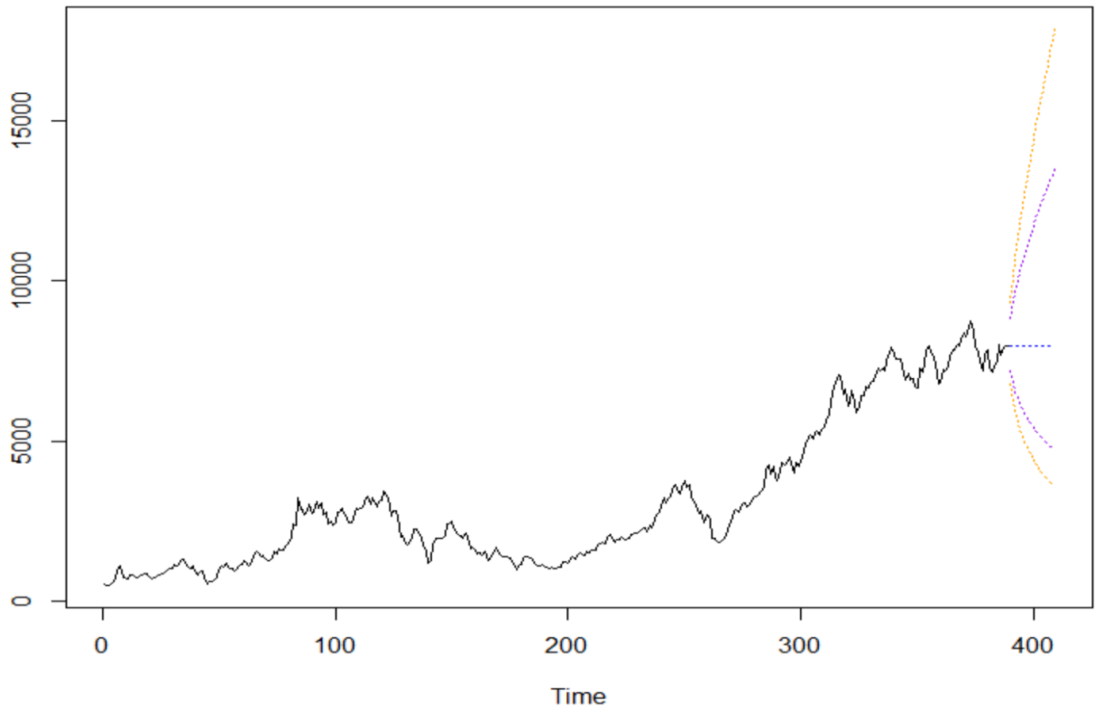
```
> Box.test(fit$residuals,lag=20)

        Box-Pierce test

data:  fit$residuals
X-squared = 11.929, df = 20, p-value = 0.9185
```

Forecast

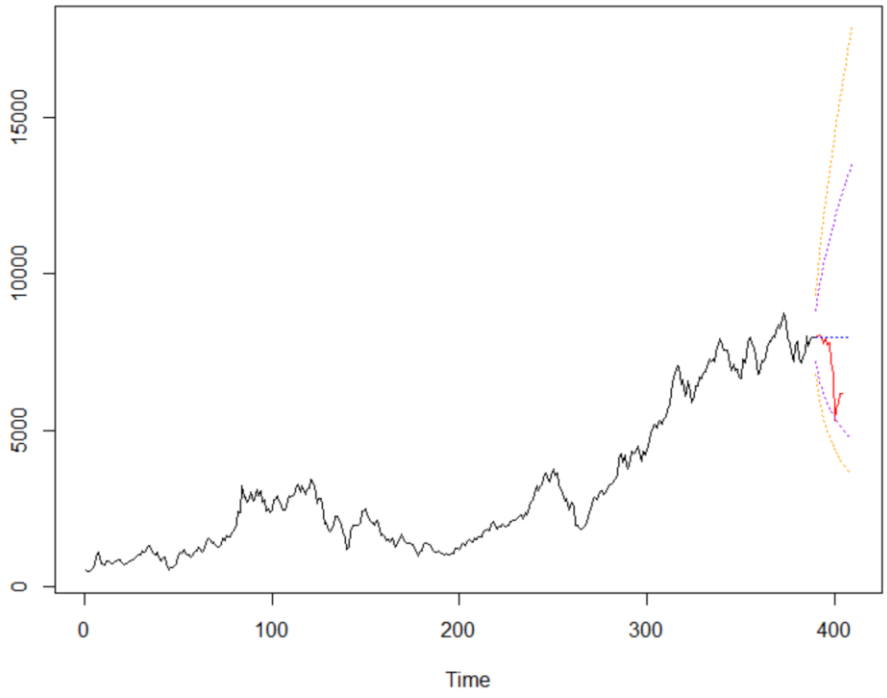
The forecast functions provides the mean forecast and the 80% and 95% CI forecasts. Notice that our forecasts is still on the logged data.



After converting this back to prices, our forecasts are shown below:



The latest prices (red) are in the bounds of the forecasts, specifically it is within the 80% confidence level projection, or Lower Bound 80%.



Conclusion and Discussion

While the time series forecast provides a mean prediction, it is beneficial to present the forecasts within a certain confidence level to manage expectations of clients/investors. Overall, it is good to project a range of outcomes vs. a single point and we can use the results here alongside existing analyst forecasts.

The results here can be used for stress-testing purposes to ready one's finances prior to disasters. This can also be used to project for much longer horizons under a Goals-Based Investing Framework where the output is the "Probability of Meeting Target Amount."

References

Overall process:
https://towardsdatascience.com/the-complete-guide-to-time-series-analysis-and-forecasting-70d476bfe775
https://towardsdatascience.com/advanced-time-series-analysis-with-arma-and-arima-a7d9b589ed6d
https://online.stat.psu.edu/stat510/lesson/3/3.1

ADF test:
https://www.real-statistics.com/time-series-analysis/stochastic-processes/dickey-fuller-test/
https://nwfsc-timeseries.github.io/atsa-labs/sec-boxjenkins-aug-dickey-fuller.html

AR and MA levels:
https://people.duke.edu/~rnau/411arim3.htm
https://online.stat.psu.edu/stat501/lesson/14/14.1
https://online.stat.psu.edu/stat510/lesson/2/2.1

Using R:
https://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/
https://a-little-book-of-r-for-time-series.readthedocs.io/en/latest/src/timeseries.html