ECON 577 Programming Assignment 2

Stanley Hong

Due October 16, 2023

Problem: Part 1. We want to test if US equity returns are mean reverting (homoscedastic and serially uncorrelated) at various return horizons. Compute the variance ratio to test for mean-reversion of 1-year returns over various horizons. The variance ratio is defined as:

$$VR(q) = \frac{\sigma_q^2}{q\sigma_1^2},$$

where VR is the variance ratio, q is the return horizon, and σ_q^2 is the variance of q-horizon return. Thus, σ_1^2 is the variance of 1-year returns.

The tests should be conducted on log excess returns. Thus, you'll need to compute the log returns for various return horizons, q, and subtract the realized log 3M T-bill rate over the same horizon. Use horizons of $1, 2, 3, 5, 7, 10$ years for your tests, computed using monthly returns. Run your tests on data after 1947. Show the variance ratio for each return horizon.

Sol. We separate the question to minor steps following the problem guide. First we obtain the log return every month. This allows for addition across months. For example, the log return over a year would simply be the sum of log return over each of the 12 months. Then we transform the annualized returns of 3M T-bills to monthly returns, then taking log so that they are also summable. Then the log excess return each month is obtained from subtraction. This could be done as we wish to compute the difference between log returns and log realized risk-free returns for a 12-month period for example, we essentially do

$$\log(R_{12}) - \log(R_{12,f}) = \log(R_1) + \cdots + \log(R_{12}) - (\log R_{1,f} + \cdots + \log R_{12,f}) = \sum_{i=1}^{12} (\log R_i - R_{i,f}).$$

Lastly we cut the data frame so that it only includes data from year 1948.

```
df['log_return'] = np.log(df['SPX TR'] / df['SPX TR'].shift(1))
df['log_rf'] = np.log(np.power((1 + df['3M Govt']), 1/12))
df['log_excess'] = df['log_return'] - df['log_rf']

df = df[df['Date'] >= '1948-01-01']
```

Then, we further setup for the horizons `[12,24,36,60,84,120]` and the variance ratios column for final display. Additionally, we also calculate the one-year return variance calling `rolling` with 12 months. This will be the standard basis to be divided by each of the horizon variances.

```
horizons_months = [12, 24, 36, 60, 84, 120]
variance_ratios = []

```

```
df['sum_one_year'] = df['log_excess'].rolling(12).sum()
var_one_year = df['sum_one_year'].var()
```

Lastly, we do the calculation. For q taking different months, we first form the rows of rolling sums over horizon of excess returns by summing the previous q months of monthly $\log$ excess return. Then for each of the rows, we take their variance through the `.var()` call, and then we apply the equation

$$VR(q) = \frac{\sigma_q^2}{q\sigma_1^2}$$

to obtain the variance ratios. The code is as follows.

```
for q in horizons_months: # take q = 12, 24, 36, 60, 84, 120
    # rolling sums for q months
    df[f'sum_{q}'] = df['log_excess'].rolling(q).sum()
    # calculate variance for the column
    variance = df[f'sum_{q}'].var()
    # calculate variance ratio
    variance_ratio = variance / (q * var_one_year / 12)
    variance_ratios.append(variance_ratio)

print(variance_ratios)
```

The variance ratios are as follows:

```
[1.0, 1.851109157041867, 2.6243569222963443,
4.013796433054526, 5.189812438052744, 6.291407809248365]
```

Problem: Part 2. We want to test whether or not the variance ratio results in part 1 are statistically different from 1. The test statistic is defined as

$$J_r(q) = \frac{\sigma_q^2}{q\sigma_1^2} - 1 = VR(q) - 1,$$

where VR is the variance ratio, q is the return horizon, and σ_q^2 is the variance of q-horizon return. The null hypothesis is $VR(q) - 1 = 0$ and the alternative hypothesis is $VR(q) \neq 1$. The asymptotic distribution of $\sqrt{Tq}J_r(q) \sim N(0, 2(q-1))$, where T is the length of the data. Compute the test statistic for each variance ratio in part 1 to determine whether to reject the null hypothesis that $\log$ excess returns are serially uncorrelated and homoscedastic. Use a significance level of $z = \pm 2$.

Sol. We directly calculate the z-scores as follows:

```
z_list = []
for i in range(0, len(horizons_months)):
    z_score = (variance_ratios[i] - 1) * np.sqrt(len(df) * horizons_months[i] / 12) / np.sqrt(2 *
        (horizons_months[i] / 12 - 1))
    z_list.append(z_score)

print(z_list)
```

The z-scores are as follows:

```
[nan, 25.190530648393516, 41.63553488349727,
70.51897258604218, 94.71200527893714, 116.7312226748709]
```

The z-scores are very high, implying the null hypothesis should be rejected and there exists mean reversion.

Problem: Part 3. The purpose of this exercise is to determine whether or not the dividend-to-price ratio forecasts returns or dividend growth. Variation in the dividend-to-price ratio by definition must predict changes in expected returns, changes in dividend growth, or some combination thereof. To determine this, you will estimate two different regression models of the form:

$$R_{t,t+k} = \hat{a} + \hat{b}\left(\frac{D}{P}\right)_t + \varepsilon_{1,t,t+k},$$

$$d^g_{t,t+k} = \hat{c} + \hat{d}\left(\frac{D}{P}\right)_t + \varepsilon_{2,t,t+k},$$

where $R_{t,t+k}$ is the realized excess return from time t to time k, $(D/P)_t$ is the dividend yield at time t, $d^g_{t,t+k}$ is the real growth rate of dividends between time t and k, and $\varepsilon_{1,t,t+k}$ and $\varepsilon_{2,t,t+k}$ are two zero-mean residuals. Run each regression at $1, 2, 3, 4, 7, 10$ year return horizons. For the first analysis, regress the return of each of the excess return series from part 1 on the beginning-of=period dividend yield. Be sure there is no overlap between the dividend yield and the return. For the second analysis regress forward log real dividend growth on the dividend-to-price ratio for the same horizons.
Show the intercept, β, R^2, and t-statistic of each regression. What do your results tell you about the relationship between the dividend yield and equity predictability? What about the relationship between dividend yield and dividend growth?

Sol. We use the statmodel package in addition to the previous numpy and pandas package.

```
import statsmodels.api as sm
```

Regarding the first analysis, we could directly use the Div Yield column along with the previously created column to do linear regression. Again, as we have to regress against each time intervals.

```
for q in horizons_months: # take q = 12, 24, 36, 48, 60, 84, 120
   temp_df = df[[f'sum_{q}', 'Div Yield']].dropna()
   model = sm.OLS(temp_df[f'sum_{q}'], sm.add_constant(temp_df['Div Yield']))
   result = model.fit()
   print(result.summary())
```

The results are as follows:

- 12 months: $R^2 = 0.069$, $\beta = -13.9778$, $t = -7.976$.
- 24 months: $R^2 = 0.087$, $\beta = -31.1094$, $t = -8.982$.
- 36 months: $R^2 = 0.106$, $\beta = -51.8783$, $t = -9.971$.
- 60 months: $R^2 = 0.176$, $\beta = -116.7647$, $t = -13.195$.
- 84 months: $R^2 = 0.236$, $\beta = -193.6526$, $t = -15.630$.
- 120 months: $R^2 = 0.0246$, $\beta = -259.0740$, $t = -15.707$.

We see that R^2 is increasing but β decreasing - implying a decreasing relationship between realized excess return and dividend yield, increasing with respect to the increasing time period.

For the second analysis, we conduct similar analysis but before that we create the column of log realized dividend growth using similar methods to part 1.

```
df['log_div_growth'] = np.log(df['Real Dividends'] / df['Real Dividends'].shift(1))
for q in horizons_months: # take q = 12, 24, 36, 60, 84, 120
    # rolling sums for q months
    df[f'div_sum_{q}'] = df['log_div_growth'].rolling(q).sum()
    temp_df = df[[f'div_sum_{q}', 'Div Yield']].dropna()
    model = sm.OLS(temp_df[f'div_sum_{q}'], sm.add_constant(temp_df['Div Yield']))
    result = model.fit()
    print(result.summary())
```

The results are as follows:

- 12 months: $R^2 = 0.001$, $\beta = 0.1082$, $t = 0.691$.
- 24 months: $R^2 = 0.001$, $\beta = 0.3080$, $t = 1.066$.
- 36 months: $R^2 = 0.008$, $\beta = 0.1006$, $t = 0.261$.
- 60 months: $R^2 = 0.024$, $\beta = -2.2591$, $t = -4.459$.
- 84 months: $R^2 = 0.106$, $\beta = -5.7273$, $t = -9.667$.
- 120 months: $R^2 = 0.252$, $\beta = -9.4707$, $t = -15.929$.

Again, R^2 is increasing and β showing a decreasing trend. In both regression sequences, an increasing interval period will increase the predicting power R^2 while maintaining a (vaguely) negatively proportional relationship with the interval. Again we also see t being more "sharp", also implying stronger predictive power.

Concluding comments. Although the results are conclusive, they are contrary to the textbook results. Re-checking with the methodology, the primary error source could be from the data processing process rather than calculation. Noticeably, even with removing the n/a columns in part 3, the problem with the oversensitivity of data still exists. Additionally, accounting for the realized 3M bill by computing the "monthlied" rate based on the annualized rate is highly theoretical and also could see a change. As a result of part 1, the calculation of z-scores should also be flawed, with z-scores way higher than expected (also due to the highly sensitive nature of the model.)