

To: Cambridge Square Capital LP

From: Kedar Gangopadhyay, NYU Courant

Date: Monday, March 18th 2019

Re: **Currency pairs**

Summary and Recommendation: (5, 20) Moving Average Rule

The recommendation is based on a simple time series moving average rule to exploit the momentum of each currency pair: go long the JPY if the 5-day USD/JPY trailing rate is higher than its 20-day average, else go short. The same would apply to MXN.

The rule assumes active management of the pairs on a monthly basis. This is important considering recent volatility in JPY and idiosyncratic trading in MXN due to U.S. executive branch statements on immigration and trade which directly effects Mexico.

This simple trading strategy has been historically profitable through the sample period *without* considering interest rate differentials and transaction costs. The details behind the rules' performance on 2016-2018 should be read in conjunction with the submitted Python code:

```
+-----+
| Results for (5, 20) USD/JPY momentum strategy |
+-----+-----+-----+-----+
| Year | Avg. Long | Avg. Short | Avg. Return |
+-----+-----+-----+-----+
| 2016 | 110.009   | 105.330   | 4.44        |
| 2017 | 112.531   | 111.270   | 1.13        |
| 2018 | 111.618   | 109.291   | 2.13        |
+-----+-----+-----+-----+
+-----+
| Results for (5, 20) USD/MXN momentum strategy |
+-----+-----+-----+-----+
| Year | Avg. Long | Avg. Short | Avg. Return |
+-----+-----+-----+-----+
| 2016 | 19.093    | 17.763    | 7.49        |
| 2017 | 18.944    | 18.784    | 0.85        |
| 2018 | 19.541    | 18.908    | 3.35        |
+-----+-----+-----+-----+
+-----+
| Aggregate returns |
+-----+-----+
| Year | Avg. Total Return |
+-----+-----+
| 2016 | 11.93             |
| 2017 | 1.98              |
| 2018 | 5.48              |
+-----+-----+
+-----+
```

Rationale

The moving average rule compares a short-run moving average to a long-run moving average (5-day *vs.* 20-day), producing a long signal when the short-run moving average cuts the long-run. This simple strategy is compelling for the following reasons:

- Evidence of profits before transaction costs through the sample period
- Simplicity lends itself to further optimization of the (5, 20) window and heuristics (see Next Steps)
- JPY and MXN are expected to move in opposite directions in 2019 (long and short, respectively)
- Liquidity:
 - JPY is one of the more popular cross currencies and it is traded against *all* of the other majors (see Catalysts)
 - MXN crosses with USD attract fewer participants, but still offers the most liquid access to Latin America
- Transaction costs continue to decline for JPY (EBS) and MXN (Reuters) on competing platforms

Catalysts

The moving average rule can exploit two near-term catalysts:

- JPY will continue to be a haven currency as GBP and EUR face headwinds from Brexit
- MXN will continue to experience pressure due to falling oil prices undermining Mexico's growth while oil production in the country continues to decline

Next Steps

The advantage of starting with a *simple* rule is that the strategy lends itself to optimization. For example, we can try other combinations of the (5, 20). Better yet, it is possible to add other rules and recombine them to find a new class of rules. For example, we might consider combining the (5,20) moving average rule with a filter-based rule which gives the signal to go long (short) if a defined upwards (downwards) exchange rate change has occurred; there can be many such rules.

As a next step the following genetic programming trial can be implemented in Python using the DEAP library:

1. Create an initial “generation” of random rules
2. Measure the fitness of each rule over a training period (i.e., 2000-2015) and rank according to fitness
3. Select the top-ranked rule and calculate its fitness over the selection period. Save it as the initial best rule
4. Randomly select two rules, using weights attaching higher probability to more highly ranked rules
5. Recombine to create a new rule which then replaces an old rule. Repeat this procedure 500 times to create a new generation of rules
6. Measure the fitness of each rule in the new generation over the training period. Take the best rule in the training period and measure its fitness over the selection period. If it outperforms the previous best rule, save it as the new best rule
7. Stop if no new best rule appears (i.e., no more incremental returns to Cambridge are generated), else return to step 4

Technology

This project was completed in Python 2.7. To install the required **PrettyTable** package:

```
conda install -c synthicity prettytable or sudo pip install PTable
```