Finance of Emerging Markets, Tutorial Week 4: Currency Risk

Question 1: **Optimal hedge ratio**

The variance of a hedged portfolio is:

$\sigma^2_{hedged} = \sigma^2_{unhedged} + h^2\sigma^2_{future} - 2h\rho\sigma_{unhedged}\sigma_{future}$,

where σ^2_{hedged} is the variance of the hedged portfolio, $\sigma^2_{unhedged}$ the variance of the unhedged portfolio, σ^2_{future} is the variance of the hedging portfolio of futures, and h is the hedge ratio.

Question 1.a. Derive the optimal minimum variance hedge ratio.

First Order Condition: $2h\sigma^2_{future} - 2\rho\sigma_{unhedged}\sigma_{future} = 0$

Divide both sides by $2\sigma^2_{future}$

$h = \rho\sigma_{unhedged}/\sigma_{future}$

Question 1.b. Assume that $\sigma^2_{unhedged}$ = 0.37, σ^2_{future} = 0.23 and ρ = 0.30

What is the optimal hedge ratio?

h = 0.30 * √0.37 / √0.23 = 0.380503

Question 1.c. Plot the variance of the portfolio after hedging as a function of the h ratio in Python. Use the commands from the "Loan Pricing" Python of the last tutorials.

```
import matplotlib.pyplot as plt

import numpy as np


#Inputs

correlation = input('Enter the correlation between R and F \n')

correlation = float(correlation)

variance_r = input('Enter the variance of R \n')

variance_r = float(variance_r)

variance_f = input('Enter the variance of F \n')

variance_f = float(variance_f)


# 10 linearly spaced numbers

x = np.linspace(0,1)

# the function, which is y = x^2 here

y = variance_r + x**2 * variance_f - 2 * x * correlation * variance_r**0.5 * variance_f**0.5

# setting the axes at the centre

fig = plt.figure()

# plot the function

plt.plot(x,y)

#Label axis

plt.xlabel("Hedge Ratio")

plt.ylabel("Variance Portfolio (Hedged)")

# show the plot
```

Question 1.d. Explain why a hedge ratio of 1 is not always optimal.

> Exchange rates and local return returns are potentially correlated, and therefore lower hedge ratio's than 1 minimize the variance of the portfolio after hedging.

Question 1.e. Give an economic explanation why exchange rates and local return returns are potentially correlated (hint: think how the exchange rate affect exports).

> Consider an exporting firm (e.g. Ferrari in Italy). An increase in the exchange rate E(foreign currency / domestic currency) makes Ferrari cars more expensive abroad, and therefore exports will drop. The drop in exports negatively affect the local returns of Ferrari. This example shows that exchange rates and returns are correlated.

Question 2: **Currency momentum trading strategy**

Question 2.a Explain when a trader who follows the momentum trading strategy buys (sells) a currency.

The trader buys (sells) currencies that realized positive (negative) returns in the past x days or months.

Question 2.b Read the introduction of Moskowitz et al. (JFE2012), Time Series Momentum. What could explain positive time series momentum according to the authors?

Positive time series momentum that partially reverse over the long-term is according to the authors consistent with initial under-reaction and delayed over-reaction of prices. Sentiment theories can produce these return patterns.

Now we are going to implement this trading strategy in Python. To start, install the Yahoo Finance Package (type "pip3 install yfinance" in Anaconda Prompt). This package allows to import financial time series from Yahoo Finance.

Next, open Jupyter notebook and import the following packages:

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import yfinance as yf
```

Now load the GBP / USD exchange rate series from Yahoo Finance using the following code:

```
ticker = 'GBPUSD=X'

yfObj = yf.Ticker(ticker)

data = yfObj.history(start='2000-01-01', end='2021-12-31')
```

Question 2.c Print the data in Python and plot the time series. Plot the closing exchange rate using the following command.

```
data['Close'].plot(title="GBP/USD Exchange Rate")
```

Did the USD appreciate or depreciate against the GBP over the sample period?

The USD depreciated

Question 2.d Calculate the daily log returns using the following command:

```
returns = np.log(data['Close'] / data['Close'].shift(1)).dropna()
```

Explain all elements of the function: np, log(), data['Close'], shift(1), dropna()

np = use the numphy package log() = takes the natural logarithm of the function in brackets. data['Close'] = uses the "Close" column in the data frame "data" shift(1) = take the one period (1-day) lag (t-1) of the data "data['Close']" dropna = Removes missing values

In Python you could define a function, such as $f(x) = x^2 + 1$ in the following way:

```
def function(x):
        return x**2 + 1
```

You could type `function(4)` and you will see that Python returns the answer (17). We will now define a 1-day Time Series Momentum Trading Strategy (TSMStrategy) which buys the currency if the returns of the previous day are positive. The function `TSMStrategy` uses the *`returns`* as input, and the performance of the strategy as output.

```
def TSMStrategy(returns, period=1):

    position = returns.rolling(period).mean().map(

        lambda x: 0 if x <= 0 else 1)

    performance = position.shift(1) * returns

    return performance
```

You could calculate the performance of the momentum trading strategy over the entire trading period using the following code:

```
performance = TSMStrategy(returns, period=1).dropna()

print(performance)
```

Question 2.e What is the performance of the trading strategy if the currency return of the previous day is negative?

0 The trader does not take a position in the currency if the returns are negative.

In the tutorial we will evaluate the cumulative performance of this trading strategy (returns and Sharpe ratio). We will also compare the momentum strategy with a simple Buy-and-Hold Strategy.[1] We will also consider the cumulative returns of 3, 5, 15, 30, and 90 day momentum strategies.

[1] https://www.investopedia.com/terms/b/buyandhold.asp