Chapter 19 - Exercise 1: Monthly milk production

- Cho dữ liệu monthly-milk-production-pounds.csv.
- Áp dụng mô hình HoltWinters để dự báo lượng sản phẩm sữa cho 6 tháng đầu năm 1976.

In [0]:
```
# from google.colab import drive
# drive.mount("/content/gdrive", force_remount=True)
# %cd '/content/gdrive/My Drive/LDS6_MachineLearning/practice/Chapter19_HoltWinte
```

In [0]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.holtwinters import ExponentialSmoothing
```

Đọc dữ liệu, kiểm tra/định dạng thời gian

In [0]:
```
df = pd.read_csv('monthly-milk-production-pounds.csv',
                 parse_dates=['Month'],
                 index_col='Month')
```

In [0]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 168 entries, 1962-01-01 to 1975-12-01
Data columns (total 1 columns):
milk_production_pounds_per_ cow    168 non-null int64
dtypes: int64(1)
memory usage: 2.6 KB
```

In [0]:
```
# https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html
# freq='H', 'D', 'W', 'M', 'MS': Hour, Day, Week, Month, Calendar month begin
df.index.freq = 'MS'
```

In [0]:
```
df.head()
```

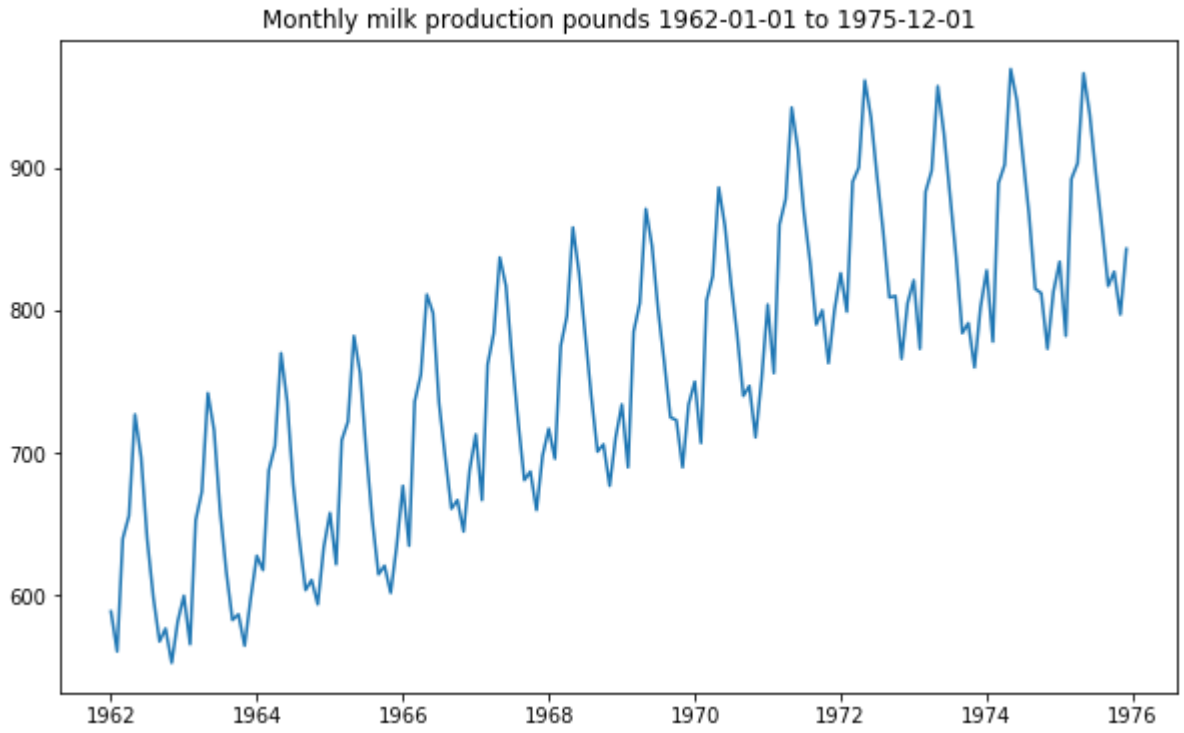
Out[5]:

Month	milk_production_pounds_per_ cow
1962-01-01	589
1962-02-01	561
1962-03-01	640
1962-04-01	656
1962-05-01	727

In [0]:
```
plt.figure(figsize=(10,6))
plt.plot(df)
plt.title("Monthly milk production pounds 1962-01-01 to 1975-12-01")
plt.show()
```

```
c:\program files\python36\lib\site-packages\pandas\plotting\_matplotlib\convert
er.py:103: FutureWarning: Using an implicitly registered datetime converter for
a matplotlib plotting method. The converter was registered by pandas on import.
Future versions of pandas will require you to explicitly register matplotlib co
nverters.

To register the converters:
        >>> from pandas.plotting import register_matplotlib_converters
        >>> register_matplotlib_converters()
  warnings.warn(msg, FutureWarning)
```

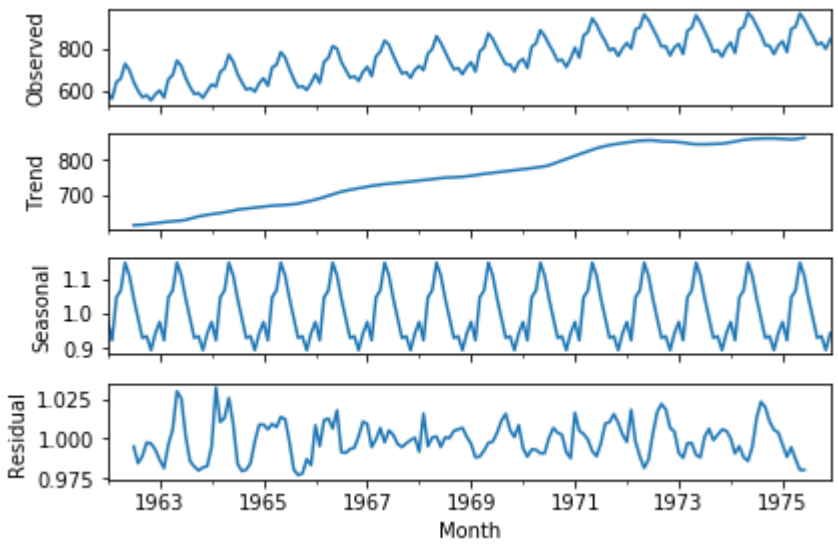


Decomposition

In [0]:
```
from statsmodels.tsa.seasonal import seasonal_decompose
result = seasonal_decompose(df, model='multiplicative')
result
```

Out[7]:
```
<statsmodels.tsa.seasonal.DecomposeResult at 0x27716306be0>
```

In [0]:
```
result.plot()
plt.show()
```



Chia dữ liệu train/test => Áp dụng mô hình

In [0]:
```
train, test = df.iloc[:130, 0], df.iloc[130:, 0]
```

In [0]:
```
train[0:5]
```

Out[10]:
```
Month
1962-01-01    589
1962-02-01    561
1962-03-01    640
1962-04-01    656
1962-05-01    727
Freq: MS, Name: milk_production_pounds_per_ cow, dtype: int64
```

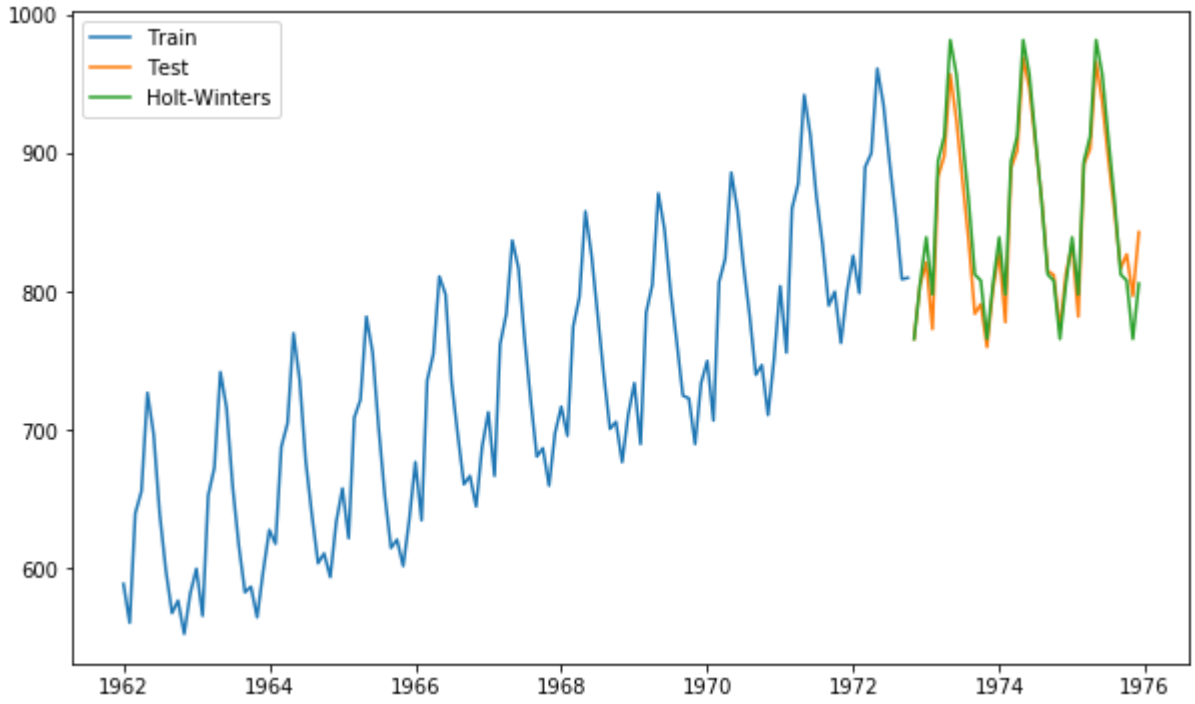
In [0]:
```
test[0:5]
```

Out[11]:
```
Month
1972-11-01    766
1972-12-01    805
1973-01-01    821
1973-02-01    773
1973-03-01    883
Freq: MS, Name: milk_production_pounds_per_ cow, dtype: int64
```

In [0]:
```
model = ExponentialSmoothing(train, seasonal='mul',
                             seasonal_periods=12).fit()
pred = model.predict(start=test.index[0],
                     end=test.index[-1])
```

In [0]:
```
plt.figure(figsize=(10,6))
plt.plot(train.index, train, label='Train')
plt.plot(test.index, test, label='Test')
plt.plot(pred.index, pred, label='Holt-Winters')
plt.legend(loc='best')
```

Out[13]:
```
<matplotlib.legend.Legend at 0x27719764278>
```



Dự đoán

In [0]:
```
df.tail()
```

Out[14]:

Month	milk_production_pounds_per_ cow
1975-08-01	858
1975-09-01	817
1975-10-01	827
1975-11-01	797
1975-12-01	843

In [0]:
```
import datetime
s = datetime.datetime(1976, 1, 1)
e = datetime.datetime(1976, 6, 1)
pred_next_6_month = model.predict(start= s, end=e)
pred_next_6_month
```

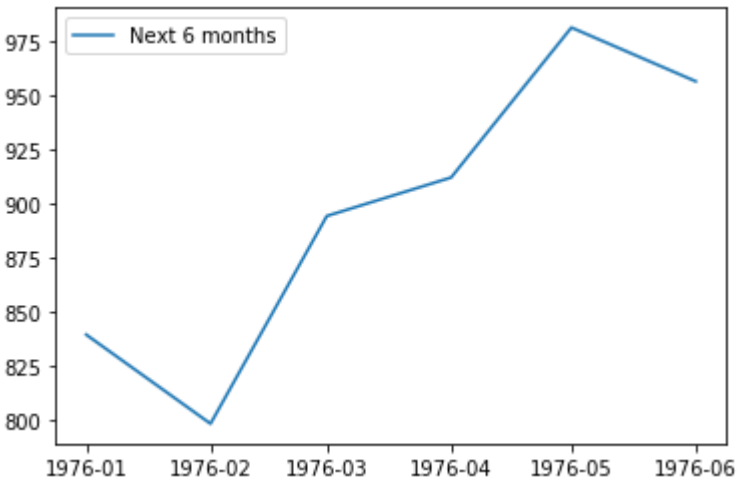
Out[15]:
```
1976-01-01    839.255997
1976-02-01    797.959544
1976-03-01    894.299974
1976-04-01    912.154098
1976-05-01    981.713246
1976-06-01    956.770929
Freq: MS, dtype: float64
```

In [0]:
```
x = pd.Series(pred_next_6_month)
type(x)
```

Out[16]:
```
pandas.core.series.Series
```

In [0]:
```
plt.plot(x.index, x.values, label='Next 6 months')
plt.legend(loc='best')
```

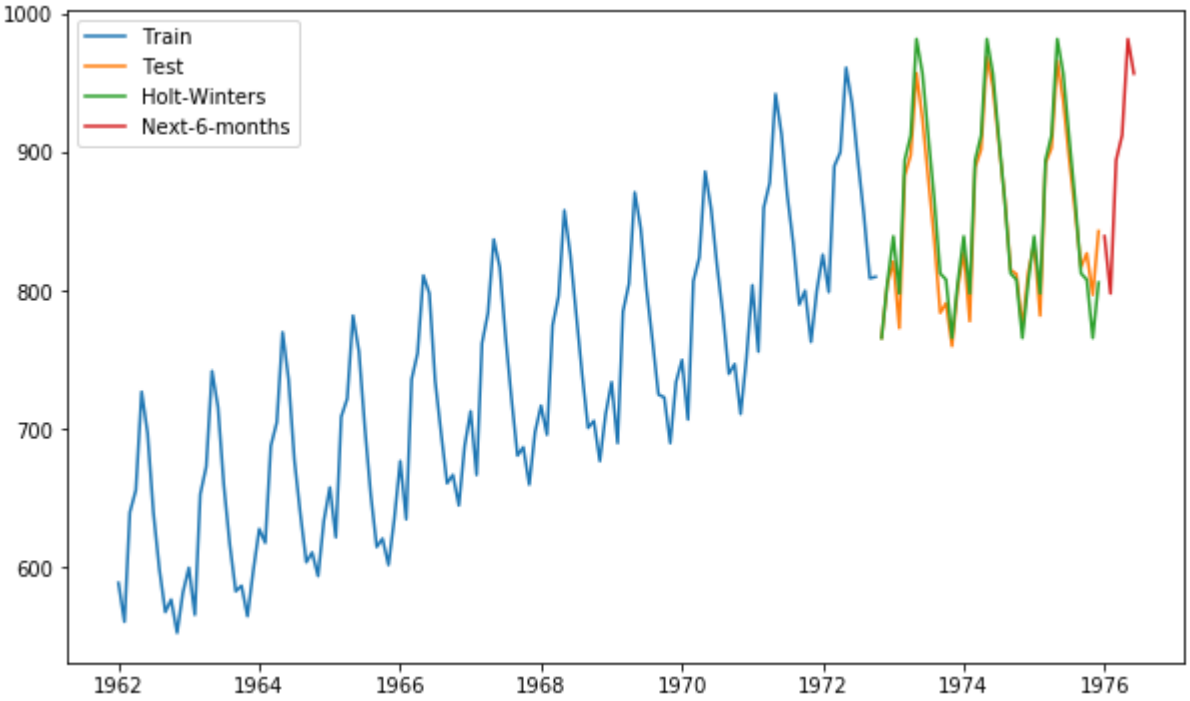
Out[19]:
```
<matplotlib.legend.Legend at 0x2771985feb8>
```



Trực quan hóa dữ liệu

In [0]:
```
plt.figure(figsize=(10,6))
plt.plot(train.index, train, label='Train')
plt.plot(test.index, test, label='Test')
plt.plot(pred.index, pred, label='Holt-Winters')
plt.plot(x.index, x.values, label='Next-6-months')
plt.legend(loc='best')
```

Out[20]:
```
<matplotlib.legend.Legend at 0x2771994fb38>
```



In [0]:
```

```