

AR model estimation to any real data

I used "ISTANBUL STOCK EXCHANGE Data Set" in UCI Machine Learning Repository. "The data is collected from imkb.gov.tr and finance.yahoo.com. Data is organized with regard to working days in Istanbul Stock Exchange." I apply AR(2) model to "NIKKEI" of the data set. The training set is the first 70% of the whole data. The test set is the last 30%. The following are my python code , the graph of real data, and the prediction graph.

In [41]:

```
# import library
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split

# import the data and the data matrix
df = pd.read_excel("/Users/daigofujiwara/Documents/授業資料/パターン認識特論/data_akbilgic.xlsx", header=0)
y = df["NIKKEI"].values[2:]
length = y.shape[0]
x = df["NIKKEI"].values[1:-1]
pre_x = df["NIKKEI"].values[:-2]
x = x.reshape([length, 1])
pre_x = pre_x.reshape([length, 1])
X = np.hstack((x, pre_x))

# split the real data to train data and test data
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                    test_size=0.1, shuffle=False)

# estimate the coefficients of AR model and predict in test set
lr = LinearRegression()
lr.fit(X_train, y_train)
y_pred = lr.predict(X_test)
n_train = len(X_train)
n_test = len(X_test)

# estimate the variance
sigma = np.std(y_test - y_pred)

lr.coef_
```

Out[41]:

```
array([-0.07373912,  0.02506034])
```

In [36]:

```
# plot the whole real data
fig = plt.figure(figsize=(13, 5))
ax = fig.add_subplot(111)
ax.plot(df["date"], df["NIKKEI"].values, label="real")
ax.legend()
ax.set_xlabel("date")
ax.set_ylabel("NIKKEI")
plt.show()
```

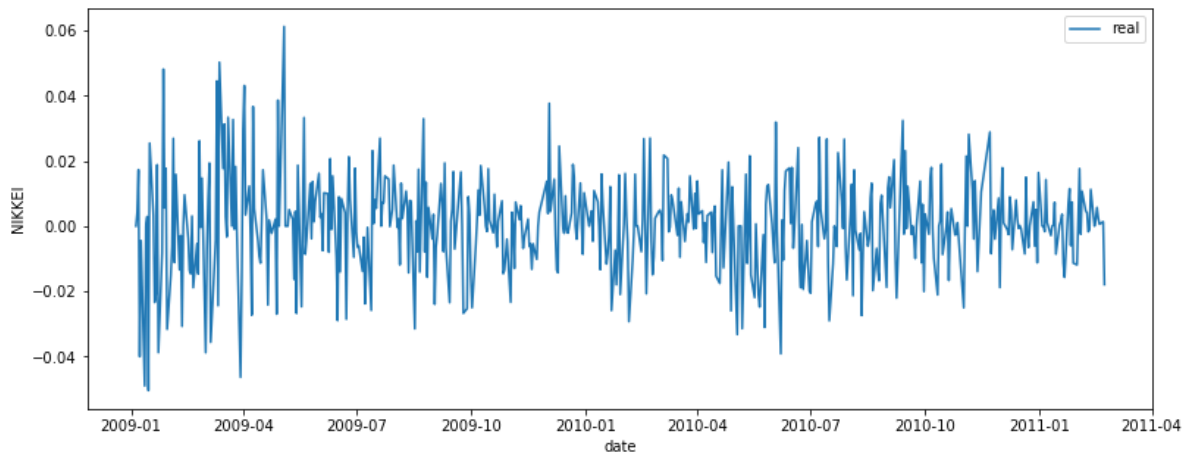


Figure 1: The whole real data. The red line is real data.

In [40]:

```
# plot the prediction
fig = plt.figure(figsize=(13, 3))
ax = fig.add_subplot(111)
ax.plot(df["date"][n_train+2:], y_test, label="real")
ax.plot(df["date"][n_train+2:], y_pred+np.random.normal(0, sigma, n_test), label="AR(2) model")
ax.legend()
ax.set_xlabel("date")
ax.set_ylabel("NIKKEI")
plt.show()
```

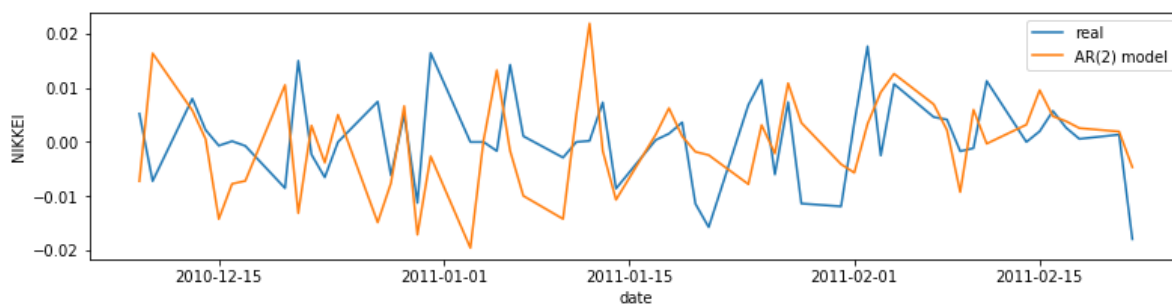


Figure 2: Prediction in test set. The blue line is the prediction. The red line is real data.

In [13]:

y_pred

Out[13]:

```
array([ 9.79485158e-04, 1.16795912e-03, 9.02215686e-04, -9.09063691e-04,
       -3.47081305e-04, -1.60278633e-04, 1.00964541e-03, -2.35347807e-03,
        9.81468422e-04, 7.01954961e-04, 6.00024908e-05, -2.07370834e-03,
        2.29300328e-03, 1.98529928e-03, -2.55732290e-04, -3.85277242e-04,
        3.35935402e-04, 5.48239302e-04, -1.97177010e-03, 5.62370540e-04,
        5.59291390e-04, -2.05003690e-03, 1.85399754e-03, 1.21573419e-03,
       -8.95175701e-04, -7.36804887e-04, 2.47235169e-03, -2.23407043e-03,
        1.03938047e-03, 6.79037800e-04, -5.39937258e-06, 2.26919209e-03,
       -4.12927825e-04, -6.12823293e-04, 8.51003024e-04, 1.72434990e-04,
       -7.25099232e-04, -5.24264400e-04, 2.35838679e-03, -2.08697308e-04,
        9.20818497e-04, 8.91107528e-04, -1.18659797e-03, -3.03293162e-04,
        2.10554026e-03, -1.66021233e-03, -5.44384636e-04, 3.85063626e-04,
       -1.24399047e-03, 1.74503498e-03, 1.56431166e-03, -3.50304757e-03,
        1.83799396e-03, -1.84024496e-03, 1.27241340e-03, -8.77091259e-04,
        9.11546582e-04, 3.17695615e-05, 4.50114859e-04, 7.82129388e-04,
       -3.11863113e-04, -9.72726150e-04, 1.69599507e-03, -9.14217834e-04,
        2.08562271e-03, -1.08122341e-03, 5.18593389e-04, -1.15043456e-03,
       -6.41017248e-04, 1.03163834e-03, 9.23784170e-04, 1.40157870e-03,
       -8.26695519e-04, 1.83004608e-05, -1.32009615e-03, 1.73018833e-03,
       -2.40648862e-04, -2.11188575e-04, 1.70046665e-03, -5.78547712e-04,
       -3.12267391e-04, 6.13721313e-04, 2.27324515e-04, -5.35374538e-05,
        2.22950666e-03, -1.05582617e-03, -1.57046139e-03, 1.13730210e-03,
       -2.13955256e-03, 5.48280227e-04, 9.77761624e-04, -1.16735414e-03,
        5.34410113e-04, 1.42507483e-03, -1.35006833e-03, -1.71185963e-03,
        2.16383445e-03, -2.42820472e-04, -2.54205190e-04, 6.99173268e-04,
       -7.33708493e-04, 2.07207449e-03, -1.13793909e-03, -1.07133443e-03,
        8.94673031e-04, 2.79808072e-04, 3.05537582e-04, -6.99238443e-04,
        1.36041151e-04, 9.73474940e-04, -8.35005116e-04, 3.35186974e-04,
        3.04297111e-04, 1.03671220e-04, 2.12366899e-04, 8.05441551e-04,
       -1.46226353e-03, 1.02610660e-03, 5.69654954e-04, -1.54167189e-04,
       -4.57712815e-04, 9.88635353e-04, -5.44850498e-04, 1.29176136e-03,
       -1.70395054e-03, 9.03564536e-04, 1.46831404e-04, 2.82303497e-04,
       -1.08748745e-03, 7.17260251e-04, 4.30476810e-04, 1.37752493e-05,
        1.30451785e-04, -4.36251215e-04, 1.18206429e-03, -2.79607016e-04,
        3.90559215e-05, -7.62102596e-05, 1.24052217e-03, 8.97106820e-04,
       -1.13441641e-03, -4.68622078e-04, 1.16249659e-03, -7.26733385e-04,
        1.40827176e-03, 5.88366002e-04, -6.91173309e-04, -1.12136915e-03,
        1.16279951e-03, -8.36597614e-04, 2.67300385e-04, 2.24119538e-05,
        4.76515897e-04, 1.61262281e-04, -8.19063800e-04, 6.65384116e-04,
       -1.30124779e-05, -2.26731017e-04, 1.97833256e-04, 2.21519747e-04,
        6.27887365e-05])
```

In []: