금융 통계학 기본(4)

상관 분석, 회귀분석

이승준 fb.com/plusjune

상관분석 Correlation Analysis

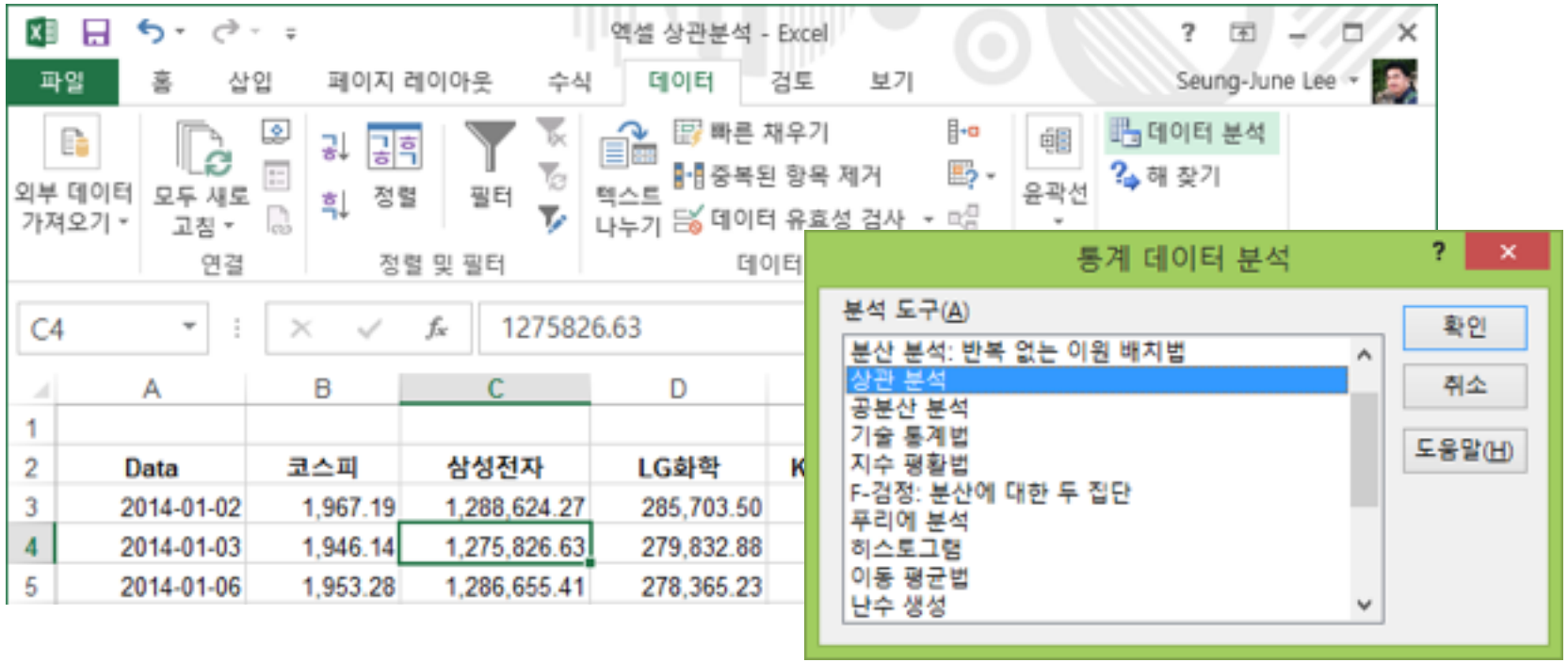
두 변수간에 어떤 선형적 관계를 갖고 있는 지를 분석하는 방법

- 공분산 covariance : (음수, 0, 양수)
- 상관계수 correlation coefficient : (0 ~ 1 사이의 값)
- 베타beta: (몇 배인지 표현)

(데이터 집합 A와 B에 대해)

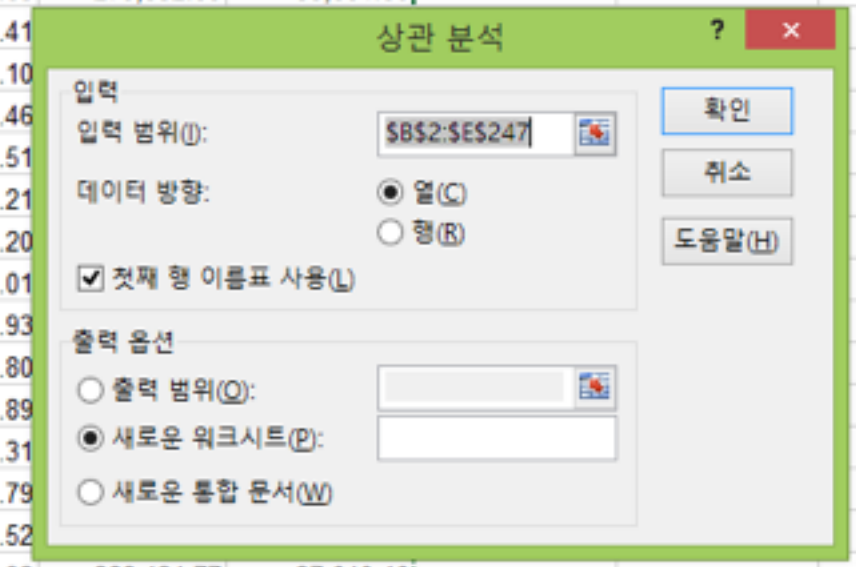
- 공분산: (음수, 0, 양수)
    - 양수면 A가 커짐에 따라 B도 커진다는 것을 의미
    - (음수, 0, 양수인지만 중요, 값의 크기는 중요하지 않음)
- 상관계수: (-1 ~ 1 사이의 값)
    - 양수면 양의 상관관계, 음수면 음의 상관관계
- 베타: (몇 배인지 표현)
    - A값이 움직일 때, B값이 얼마나(몇 배) 움직이는지를 표현
    - 시장베타: 종목의 가격이 시장 KOSPI 200 의 변화에 얼마나 영향을 받는지

엑셀 상관분석



https://goo.gl/ZpPxg3 엑셀 상관분석.xlsx

	A	B	C	D	E
2	Data	코스피	삼성전자	LG화학	KB금융그룹
3	2014-01-02	1,967.19	1,288,624.27	285,703.50	41,253.66
4	2014-01-03	1,946.14	1,275,826.63	279,832.88	39,881.80
5	2014-01-06	1,953.28	1,286,655.41		
6	2014-01-07	1,959.44	1,283,702.10		
7	2014-01-08	1,958.96	1,270,904.46		
8	2014-01-09	1,946.11	1,255,153.51		
9	2014-01-10	1,938.54	1,252,200.21		
10	2014-01-13	1,948.92	1,274,842.20		
11	2014-01-14	1,946.07	1,292,562.01		
12	2014-01-15	1,953.28	1,278,779.93		
13	2014-01-16	1,957.32	1,280,748.80		
14	2014-01-17	1,944.48	1,271,888.89		
15	2014-01-20	1,953.78	1,295,515.31		
16	2014-01-21	1,963.89	1,303,390.79		
17	2014-01-22	1,970.42	1,307,328.52		
18	2014-01-23	1,947.59	1,278,779.93	266,134.77	37,040.10
19	2014-01-24	1,940.56	1,286,655.41	262,710.24	37,040.10
20	2014-01-27	1,910.34	1,271,888.89	256,350.40	36,109.20
21	2014-01-28	1,916.93	1,263,028.99	254,393.53	35,178.30
22	2014-01-29	1,941.15	1,260,075.68	254,393.53	36,452.16
23	2014-02-03	1,919.96	1,252,200.21	248,522.91	35,423.27
24	2014-02-04	1,886.85	1,229,558.23	244,119.94	35,325.28
25	2014-02-05	1,891.32	1,217,745.02	244,609.16	35,570.26



	코스피	삼성전자	LG화학	KB금융그룹
코스피	1			
삼성전자	0.183037839	1		
LG화학	0.595013318	0.39341706	1	
KB금융그룹	0.192568512	-0.54769734	-0.30281811	1

코스피 시가총액 상위

- 한국거래소 홈페이지 데이터
- https://goo.gl/Hu8iN7 (2015-06-30) 기준

순위	종목코드	종목명
1	005930	삼성전자
2	000660	SK하이닉스
3	005380	현대차
4	015760	한국전력
5	090430	아모레퍼시픽
6	028260	제일모직
7	005935	삼성전자우
8	032830	삼성생명
9	035420	NAVER
10	012330	현대모비스

```
syms = [
    '^KS11', # 코스피 지수
    '005930.KS', # 삼성전자
    '000660.KS', # SK하이닉스
    '005380.KS', # 현대차
    '015760.KS', # 한국전력
    '090430.KS', # 아모레퍼시픽
    '028260.KS', # 제일모직
    '005935.KS', # 삼성전자우
    '032830.KS', # 삼성생명
    '035420.KS', # NAVER
    '012330.KS', # 현대모비스
]
```

필요 라이브러리

```
import pandas as pd
import requests
import matplotlib.pyplot as plt
from datetime import datetime

from pandas_datareader import data
```

읽어오기

```
start = datetime(2014, 1 ,1)
end = datetime(2014, 12, 31)

p = data.get_data_yahoo(syms, start=start, end=end)
mask = p['Volume']['^KS11'] > 0
df = p['Adj Close'].ix[mask]
```

패널

다수의 심볼을 지정할 수 있다

종목별 가격 데이터 읽기

```
start = datetime(2014, 1 ,1)
end = datetime(2014, 12, 31)


p = data.get_data_yahoo(syms, start=start, end=end)
```

```
<class 'pandas.core.panel.Panel'>
Dimensions: 6 (items) x 261 (major_axis) x 11 (minor_axis)
Items axis: Open to Adj Close
Major_axis axis: 2014-01-01 00:00:00 to 2014-12-31 00:00:00
Minor_axis axis: 000660.KS to ^KS11
```

컬럼 이름 바꾸기

```
df = df.rename(columns=syms_names)
df.head()
```

	SK hynix	HyundaiMtr	Samsung Electronics	Samsung Electronics (Preferred)	Mobis	KEPCO	SAMSUNG C&T	SAMSUNG LIFE	NAVER	AMOREPACIFIC	KOSPI Composite Index
Date											
2014-01-02	35323.08	219006.02	1288624.27	969026.31	275505.22	34147.79	NaN	99958.23	724229.68	96663.03	1967.189941
2014-01-03	36068.30	218518.26	1275826.63	948450.43	272542.80	33851.28	NaN	98973.42	698257.30	93495.32	1946.140015
2014-01-06	37409.68	222908.13	1286655.41	967066.70	275998.96	34839.64	NaN	99958.23	698257.30	94071.27	1953.280029
2014-01-07	37558.72	223395.90	1283702.10	955309.05	274024.01	35185.56	NaN	99958.23	699256.24	93303.34	1959.439941
2014-01-08	38751.06	223883.66	1270904.46	955309.05	272542.80	34987.89	NaN	99465.82	699256.24	94551.23	1958.959961

```
df = p['Adj Close']
df.head()
```

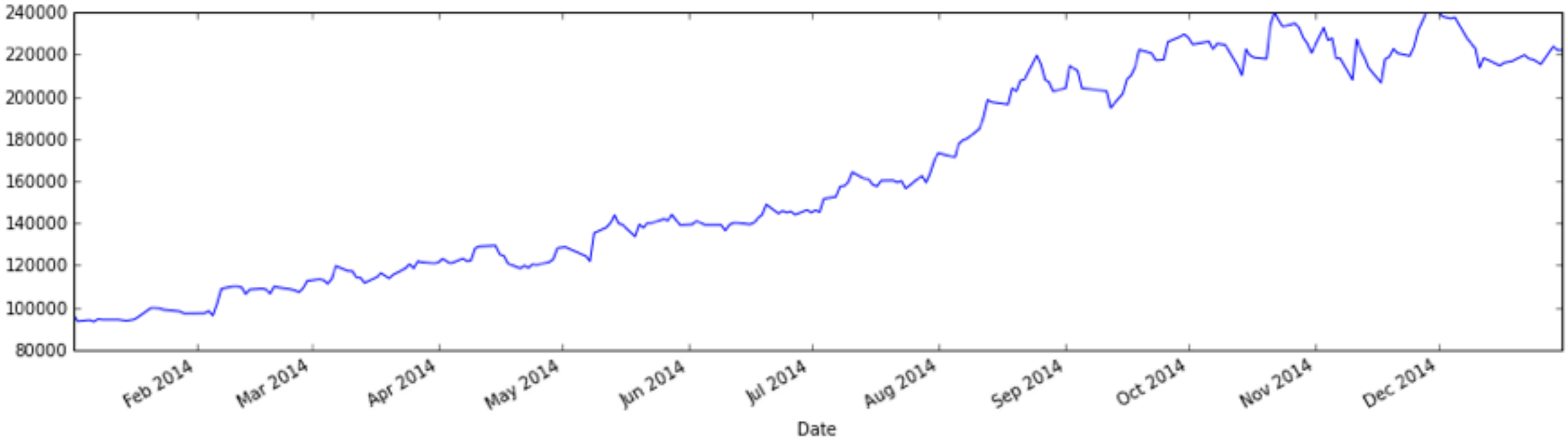
	000660.KS	005380.KS	005930.KS	005935.KS	012330.KS	015760.KS	028260.KS	032830.KS	035420.KS	090430.KS	^KS11
Date											
2014-01-01	36565.11	230712.36	1350643.62	NaN	289823.59	34345.46	NaN	102420.25	723230.74	95991.09	NaN
2014-01-02	35323.08	219006.02	1288624.27	969026.31	275505.22	34147.79	NaN	99958.23	724229.68	96663.03	1967.189941
2014-01-03	36068.30	218518.26	1275826.63	948450.43	272542.80	33851.28	NaN	98973.42	698257.30	93495.32	1946.140015
2014-01-06	37409.68	222908.13	1286655.41	967066.70	275998.96	34839.64	NaN	99958.23	698257.30	94071.27	1953.280029
2014-01-07	37558.72	223395.90	1283702.10	955309.05	274024.01	35185.56	NaN	99958.23	699256.24	93303.34	1959.43

심볼+이름

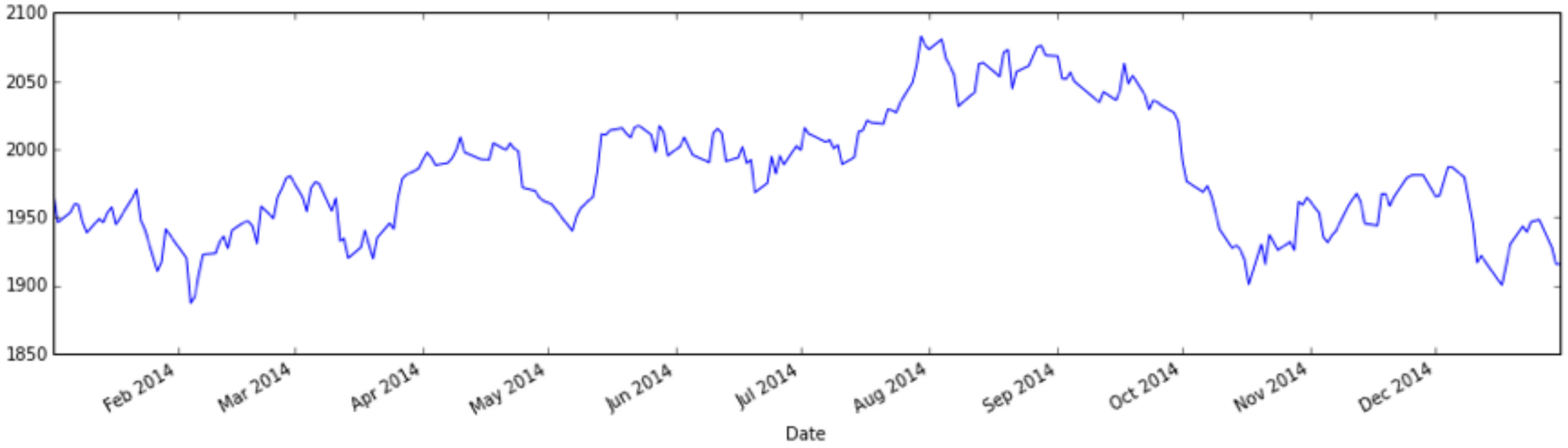
```
syms_names = {}
url_tmp = 'http://finance.yahoo.com/d/quotes.csv?s=%s&f=%s'
for sym in syms:
    url = url_tmp % (sym, 'n')
    r = requests.get(url)
    name = r.text.replace('"', '').strip()
    print (name)
    syms_names[sym] = name

print (syms_names)
```

```
KOSPI Composite Index
Samsung Electronics
SK hynix
HyundaiMtr
KEPCO
AMOREPACIFIC
SAMSUNG C&T
Samsung Electronics (Preferred)
SAMSUNG LIFE
NAVER
Mobis
{'000660.KS': 'SK hynix', '028260.KS': 'SAMSUNG C&T', '005380.KS':
'HyundaiMtr', '032830.KS': 'SAMSUNG LIFE', '005935.KS': 'Samsung
Electronics (Preferred)', '015760.KS': 'KEPCO', '090430.KS':
'AMOREPACIFIC', '005930.KS': 'Samsung Electronics', '012330.KS':
'Mobis', '^KS11': 'KOSPI Composite Index', '035420.KS': 'NAVER'}
```

```
df['AMOREPACIFIC'].plot(figsize=(16,4))
```



```
df['KOSPI Composite Index'].plot(figsize=(16,4))
```

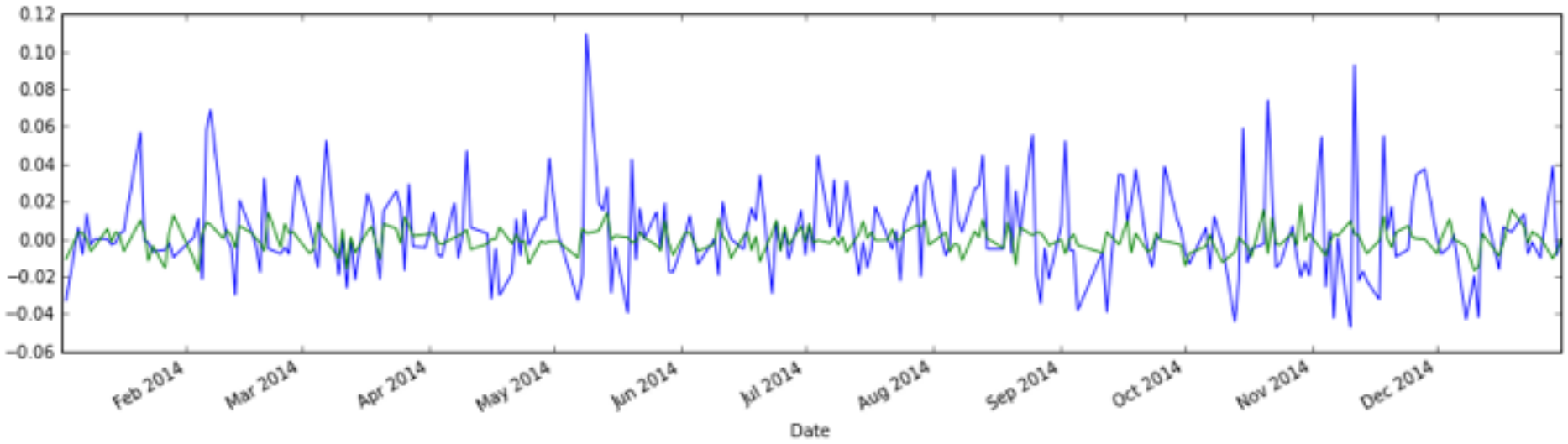


등락률 (전일 대비)

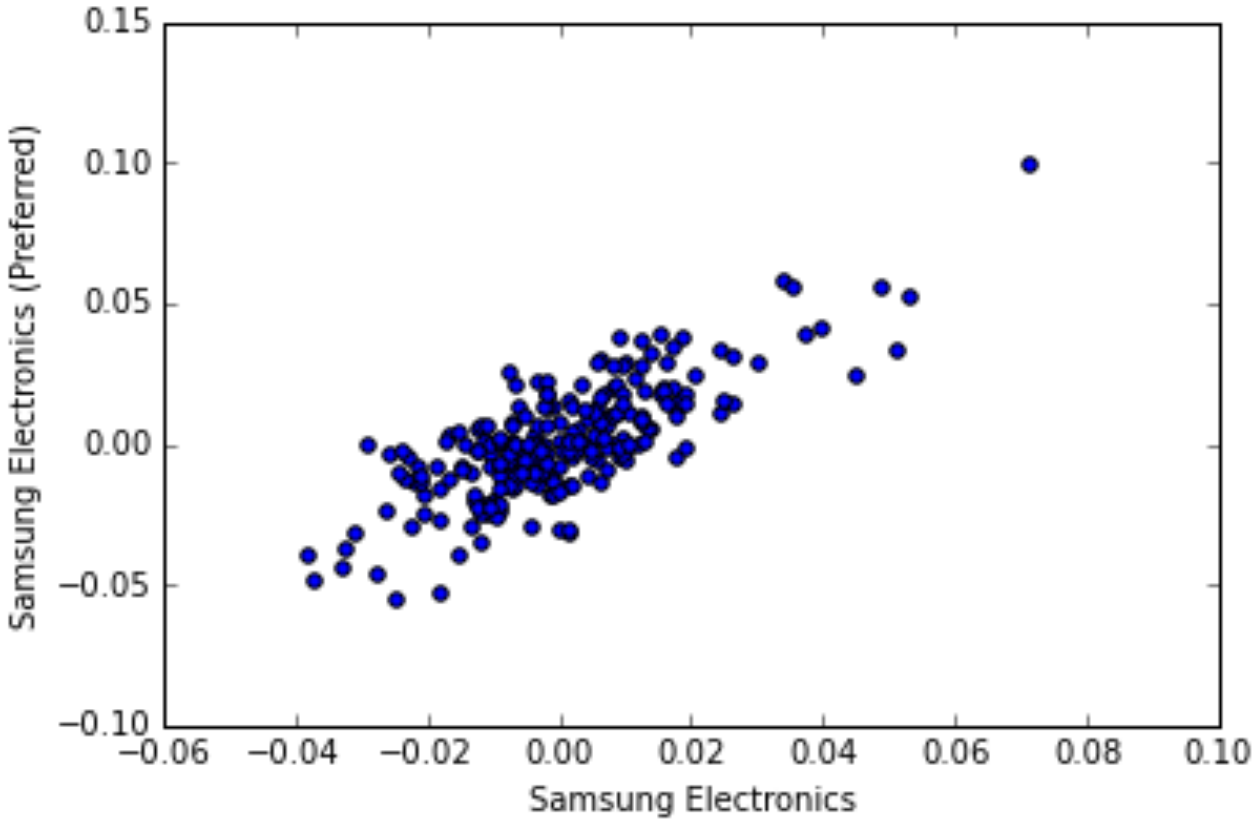
```
changes = df.pct_change()
changes.head()
```

	SK hynix	Hyundai Mtr	Samsung Electronics	Samsung Electronics (Preferred)	Mobis	KEPCO	SAMSUNG C&T	SAMSUNG LIFE	NAVER	AMORE PACIFIC	KOSPI Composite Index
Date											
2014-01-02	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2014-01-03	0.021097	-0.002227	-0.009931	-0.021234	-0.010753	-0.008683	NaN	-0.009852	-0.035862	-0.032771	-0.010701
2014-01-06	0.037190	0.020089	0.008488	0.019628	0.012681	0.029197	NaN	0.009950	0.000000	0.006160	0.003669
2014-01-07	0.003984	0.002188	-0.002295	-0.012158	-0.007156	0.009929	NaN	0.000000	0.001431	-0.008163	0.003154
2014-01-08	0.031746	0.002183	-0.009969	0.000000	-0.005405	-0.005618	NaN	-0.004926	0.000000	0.013375	-0.000245

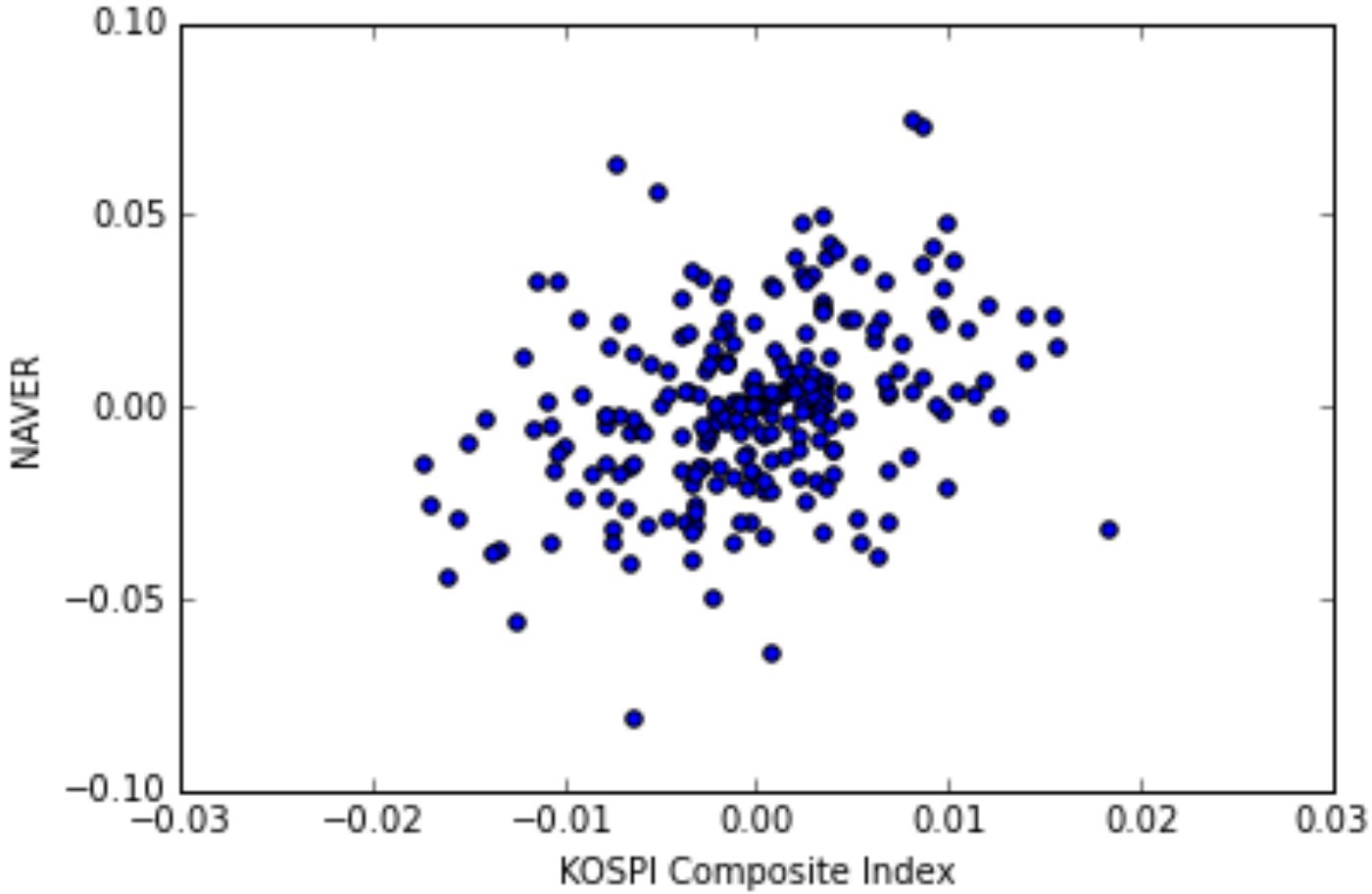
```
changes = df.pct_change()
changes['AMOREPACIFIC'].plot(figsize=(16,4))
changes['KOSPI Composite Index'].plot(figsize=(16,4))
```



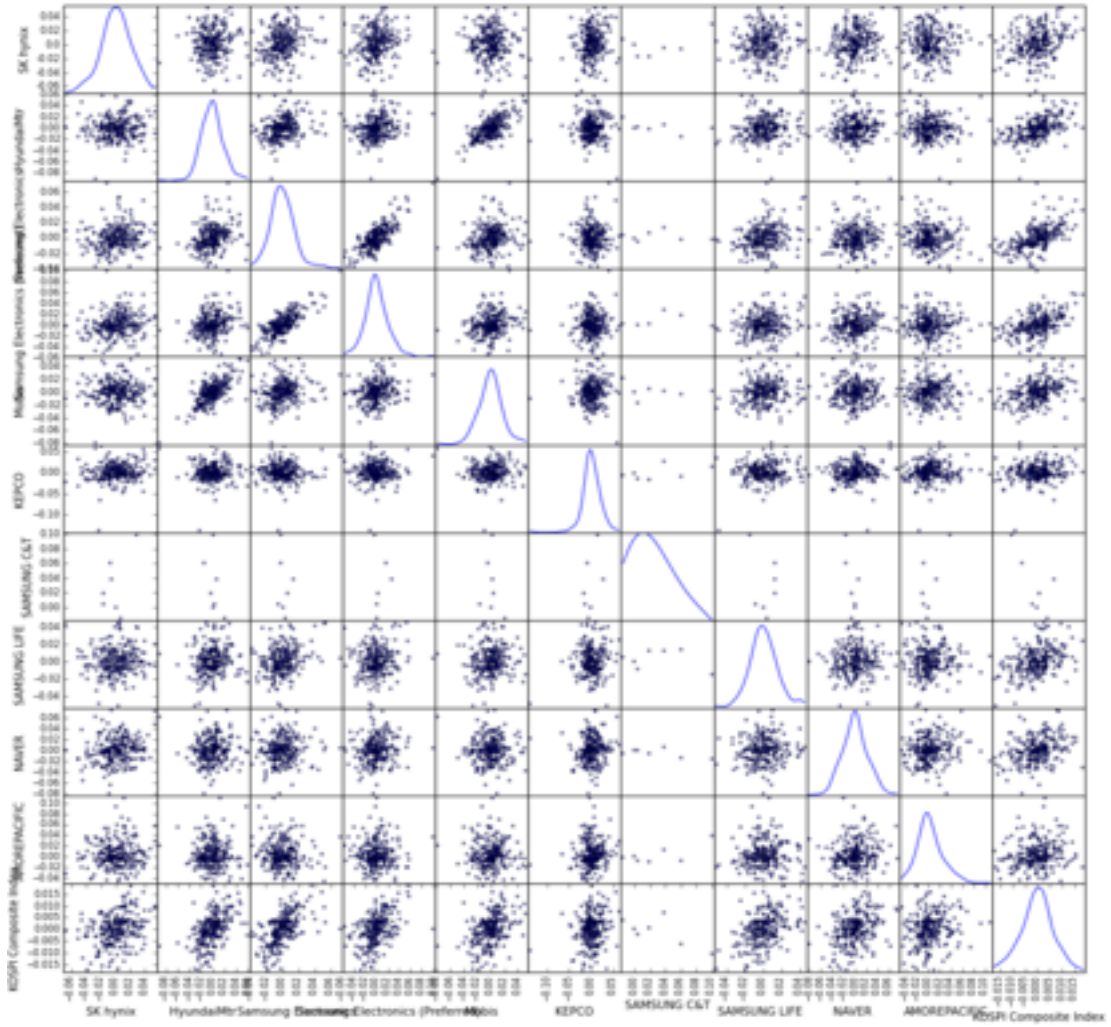
```
plt.scatter(changes['Samsung Electronics'], changes['Samsung Electronics (Preferred)'])
```



```
plt.scatter(changes['KOSPI Composite Index'], changes['NAVER'])
plt.xlabel('KOSPI Composite Index')
plt.ylabel('NAVER')
```



```
changes = df.pct_change()
pd.scatter_matrix(changes, diagonal='kde', figsize=(16, 16));
```



상위 10개 종목 상관관계 차트

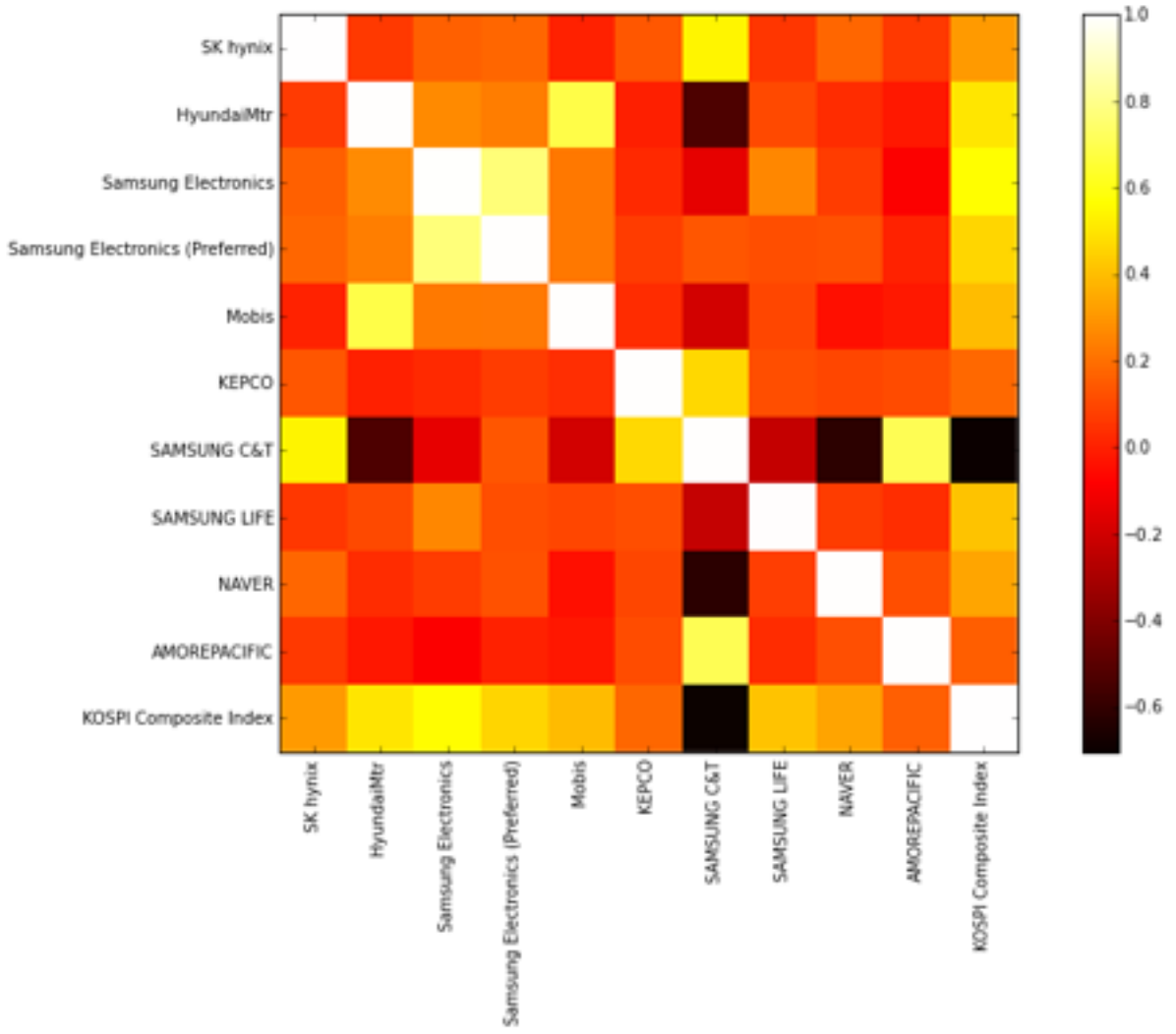
corr = changes.corr()

corr

	SK Hynix	HyundaiMtr	Samsung Electronics	Samsung Electronics (Preferred)	Mobis	KEPCO	SAMSUNG C&T	SAMSUNG LIFE	NAVER	AMORE PACIFIC	KOSPI Composite Index
SK hynix	1.000000	0.066778	0.166342	0.178147	0.001798	0.141248	0.545476	0.058417	0.178661	0.062172	0.308035
HyundaiMtr	0.066778	1.000000	0.268184	0.237778	0.692799	-0.000775	-0.535312	0.105561	0.030537	-0.018473	0.505858
Samsung Electronics	0.166342	0.268184	1.000000	0.771678	0.223979	0.026865	-0.140085	0.267070	0.069761	-0.090591	0.565236
Samsung Electronics (Preferred)	0.178147	0.237778	0.771678	1.000000	0.225650	0.070241	0.136124	0.119014	0.124076	0.004499	0.462604
Mobis	0.001798	0.692799	0.223979	0.225650	1.000000	0.032031	-0.191717	0.095242	-0.044201	-0.023647	0.398527
KEPCO	0.141248	-0.000775	0.026865	0.070241	0.032031	1.000000	0.472297	0.119383	0.100210	0.110954	0.185996
SAMSUNG C&T	0.545476	-0.535312	-0.140085	0.136124	-0.191717	0.472297	1.000000	-0.227417	-0.623445	0.708167	-0.703830
SAMSUNG LIFE	0.058417	0.105561	0.267070	0.119014	0.095242	0.119383	-0.227417	1.000000	0.071500	0.033258	0.414999
NAVER	0.178661	0.030537	0.069761	0.124076	-0.044201	0.100210	-0.623445	0.071500	1.000000	0.118209	0.335176
AMOREPACIFIC	0.062172	-0.018473	-0.090591	0.004499	-0.023647	0.110954	0.708167	0.033258	0.118209	1.000000	0.159926
KOSPI Composite Index	0.308035	0.505858	0.565236	0.462604	0.398527	0.185996	-0.703830	0.414999	0.335176	0.159926	1.000000

상위 10개 종목 상관관계 히트맵

```
corr = changes.corr()


plt.figure(figsize=(14,8))
plt.imshow(corr, cmap='hot', interpolation='
plt.colorbar()
plt.xticks(range(len(corr)), corr.columns, ro
plt.yticks(range(len(corr)), corr.columns)
plt.show()
```



상관계수 순위 만들기

```
idx = []; vals = []
for ix, i in enumerate(corr.columns.values):
    for j in corr.columns.values[ix + 1:]:
        idx.append((i, j))
        vals.append(corr[i][j])


ser = pd.Series(data=vals, index=idx)
ser_ord = ser.sort_values(ascending=False)


ser_ord[:10]
```

```
(SK hynix, HyundaiMtr)                          0.066778
(SK hynix, Samsung Electronics)                 0.166342
(SK hynix, Samsung Electronics (Preferred))     0.178147
(SK hynix, Mobis)                               0.001798
(SK hynix, KEPCO)                               0.141248
(SK hynix, SAMSUNG C&T)                         0.545476
(SK hynix, SAMSUNG LIFE)                        0.058417
(SK hynix, NAVER)                               0.178661
(SK hynix, AMOREPACIFIC)                        0.062172
(SK hynix, KOSPI Composite Index)               0.308035
dtype: float64
```

종목간 상관계수

```
code_names = {
    '^KS11':'KOSPI', '005930.KS':'삼성전자', '005380.KS':'현대차',
    '035420.KS':'네이버', '006280.KS':'녹십자', '012750.KS':'에스원',
    '003490.KS':'대한항공‘'000210.KS':'대림산업', '004990.KS':'롯데제과',
    '192400.KS':'쿠쿠전자', '007310.KS':'오뚜기' }


start=datetime(2014, 1, 1)
end=datetime(2014, 12, 31)


df = data.get_data_yahoo(list(code_names.keys()), start=start,
end=end)
df = df['Adj Close']
df = df.rename(columns=code_names)
chg = df.pct_change()
chg_corr = chg.corr()
```

	대림산업	대한항공	롯데제과	현대차	삼성전자	녹십자	오뚜기	에스원	네이버	쿠쿠전자	KOSPI
대림산업	1.000000	0.051987	0.096130	0.094469	0.016846	0.000303	0.003993	0.160489	0.073830	-0.063781	0.318489
대한항공	0.051987	1.000000	0.123438	0.207103	0.100588	0.072983	0.069704	0.082295	0.150868	-0.021284	0.257539
롯데제과	0.096130	0.123438	1.000000	0.070146	-0.016590	0.178293	0.170773	0.145363	0.118259	0.049030	0.290734
현대차	0.094469	0.207103	0.070146	1.000000	0.285957	-0.037805	-0.013826	0.037299	0.024776	-0.015548	0.485877
삼성전자	0.016846	0.100588	-0.016590	0.285957	1.000000	-0.049646	-0.063424	-0.040898	0.078287	-0.010870	0.575700
녹십자	0.000303	0.072983	0.178293	-0.037805	-0.049646	1.000000	0.289841	0.228975	0.219104	0.013372	0.183375
오뚜기	0.003993	0.069704	0.170773	-0.013826	-0.063424	0.289841	1.000000	0.184417	0.225953	0.025435	0.219538
에스원	0.160489	0.082295	0.145363	0.037299	-0.040898	0.228975	0.184417	1.000000	0.147795	0.035987	0.244487
네이버	0.073830	0.150868	0.118259	0.024776	0.078287	0.219104	0.225953	0.147795	1.000000	0.087253	0.348849
쿠쿠전자	-0.063781	-0.021284	0.049030	-0.015548	-0.010870	0.013372	0.025435	0.035987	0.087253	1.000000	0.145964
KOSPI	0.318489	0.257539	0.290734	0.485877	0.575700	0.183375	0.219538	0.244487	0.348849	0.145964	1.000000

종목간 상관계수 소트

```
ser = chg_corr['KOSPI']
ser_ord = ser.sort_values(ascending=False)
ser_ord[1:]
```

```
삼성전자    0.575700
현대차     0.485877
네이버     0.348849
대림산업    0.318489
롯데제과    0.290734
대한항공    0.257539
에스원     0.244487
오뚜기     0.219538
녹십자     0.183375
쿠쿠전자    0.145964
Name: KOSPI, dtype: float64
```

(Kia Motor, Hyundai Motor)	0.793340
(Samsung Elec, Samsung Elec(Prep))	0.702924
(Hyundai Motor, Hyundai Mobis)	0.576877
(Kia Motor, Hyundai Mobis)	0.533587
(Samsung Elec, Shinhan)	0.388561
(POSCO, Shinhan)	0.384025
(POSCO, Samsung Elec)	0.357618
(SK Hynix, Samsung Elec)	0.342223
(SK Hynix, Shinhan)	0.329163
(Samsung Elec, Hyundai Mobis)	0.321344

종목간 상관계수

```
code_names = { '^KS11':'KOSPI',
               '005930.KS':'Samsung Elec', '005380.KS':'Hyundai Motor',
               '012330.KS':'Hyundai Mobis', '005490.KS':'POSCO',
               '000660.KS':'SK Hynix','035420.KS':'Naver',
               '005935.KS':'Samsung Elec(Prep)', '000270.KS':'Kia Motor',
               '055550.KS':'Shinhan', '015760.KS':'Korea Elc Pwr' }

  df = DataReader(code_names.keys(), 'yahoo', start='2013-01-01', end='2013-12-31')
  df = df['Adj Close']
  df = df.rename(columns=code_names)

  changes = df.pct_change()
  chg_corr = changes.corr()
  chg_corr
```

	Kia Motor	SK Hynix	Hyundai Motor	POSCO	Samsung Elec	Samsung Elec(Prep)	Hyundai Mobis	Korea Elc Pwr	Naver	Shinhan	KOSPI
Kia Motor	1.000	0.194	0.793	0.177	0.267	0.158	0.534	0.068	-0.022	0.172	0.446
SK Hynix	0.194	1.000	0.192	0.188	0.342	0.292	0.123	0.109	0.102	0.329	0.459
Hyundai Motor	0.793	0.192	1.000	0.167	0.320	0.239	0.577	0.017	0.057	0.284	0.525
POSCO	0.177	0.188	0.167	1.000	0.358	0.314	0.190	0.169	0.093	0.384	0.544
Samsung Elec	0.267	0.342	0.320	0.358	1.000	0.703	0.321	0.135	0.225	0.389	0.770
Samsung Elec(Prep)	0.158	0.292	0.239	0.314	0.703	1.000	0.243	0.089	0.200	0.248	0.569
Hyundai Mobis	0.534	0.123	0.577	0.190	0.321	0.243	1.000	0.043	0.074	0.271	0.471
Korea Elc Pwr	0.068	0.109	0.017	0.169	0.135	0.089	0.043	1.000	0.101	0.135	0.233
Naver	-0.022	0.102	0.057	0.093	0.225	0.200	0.074	0.101	1.000	0.242	0.305
Shinhan	0.172	0.329	0.284	0.384	0.389	0.248	0.271	0.135	0.242	1.000	0.659
KOSPI	0.446	0.459	0.525	0.544	0.770	0.569	0.471	0.233	0.305	0.659	1.000

KOSPI 와 다른 종목간 상관계수

```
ser = chg_corr['KOSPI']
ser_ord = ser.order(ascending=False)
ser_ord[1:]
```

```
Samsung Elec          0.770216
Shinhan               0.659464
Samsung Elec(Prep)    0.569390
POSCO                 0.543671
Hyundai Motor         0.524798
Hyundai Mobis         0.470791
SK Hynix              0.459246
Kia Motor             0.446092
Naver                 0.304662
Korea Elc Pwr         0.232799
Name: KOSPI, dtype: float64
```

수익과 위험

- 수익 **returns**: 수익률 평균 **mean**
- 위험 **risk**: 표준편차 **std**, 값이 클수록 변동성이 크므로 위험이 크다

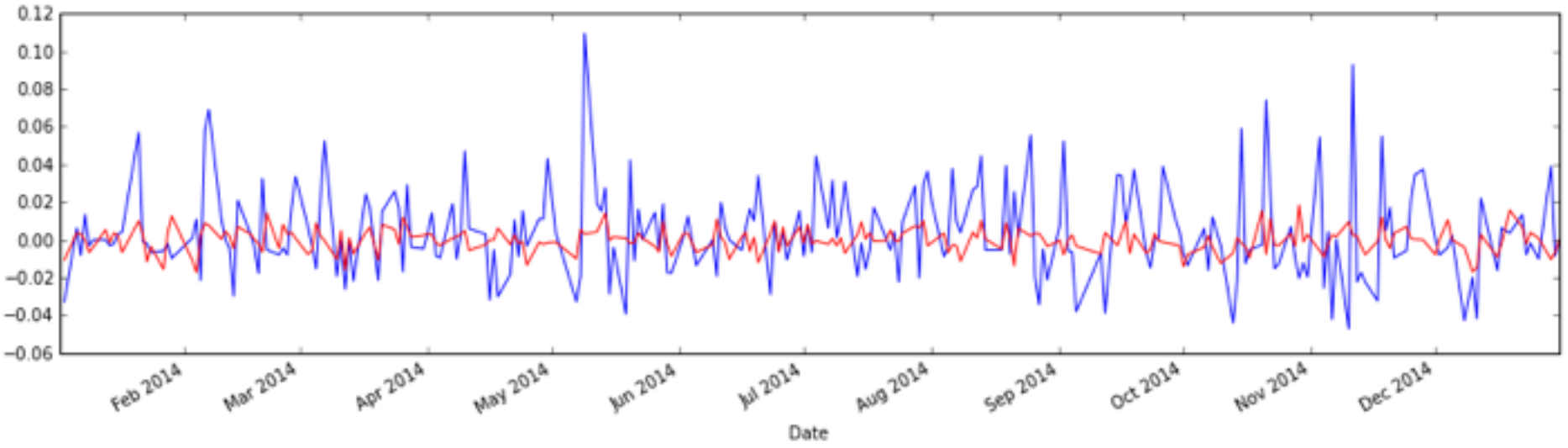
```
syms = [
    '^KS11', # 코스피 지수
    '005930.KS', # 삼성전자
    '000660.KS', # SK하이닉스
    '005380.KS', # 현대차
    '015760.KS', # 한국전력
    '090430.KS', # 아모레퍼시픽
    '032830.KS', # 삼성생명
    '035420.KS', # NAVER
    '012330.KS', # 현대모비스
    '006280.KS', # 녹십자
    '012750.KS', # 에스원
    '003490.KS', # 대한항공
    '000210.KS', # 대림산업
    '004990.KS', # 롯데제과
    '192400.KS', # 쿠쿠전자
    '007310.KS', # 오뚜기
]
```

```
start = datetime(2014, 1 ,1)
end = datetime(2014, 12, 31)

p = data.get_data_yahoo(syms, start=start, end=end)
mask = p['Volume']['^KS11'] > 0
df = p['Adj Close'].ix[mask]
syms_names = {}

url_tmp = 'http://finance.yahoo.com/d/quotes.csv?s=%s&f=%s'
for sym in syms:
    url = url_tmp % (sym, 'n')
    r = requests.get(url)
    name = r.text.replace("", '').strip()
    print (name)
    syms_names[sym] = name
```

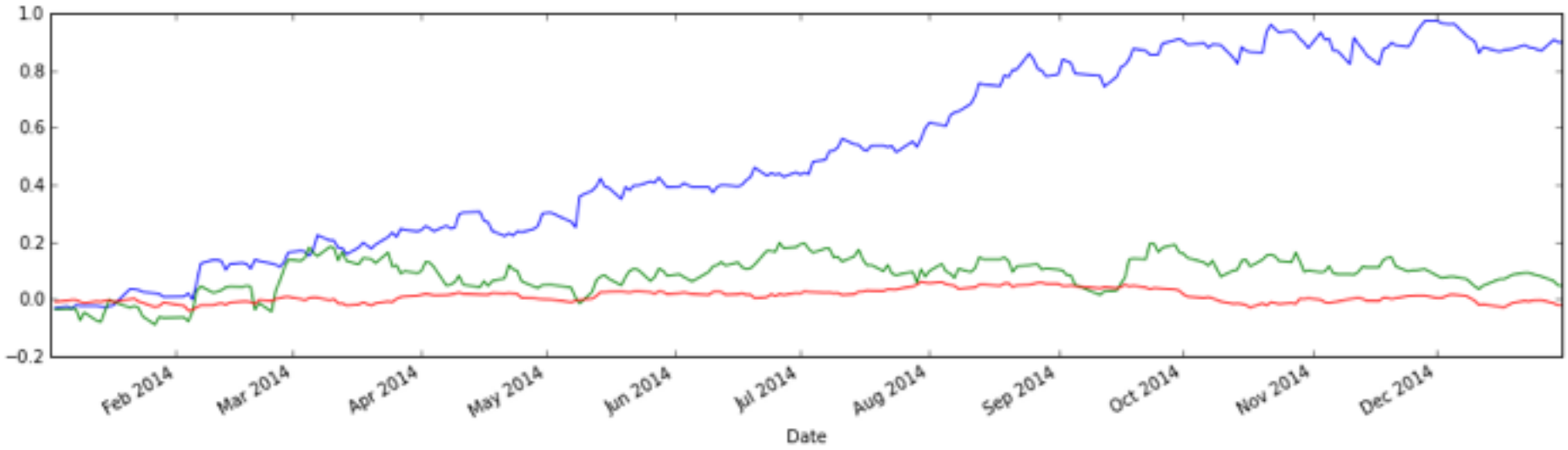
```
changes = df.pct_change()
changes.head()
```

	DaelimInd	SK hynix	KAL	LotteConf	Hyundai Mtr	Samsung Electronics	GC Corp	Ottogi	Mobis	S-1	KEPCO	SAMSUNG LIFE	NAVER	AMORE PACIFIC	CUC KOO	KOSPI Composite Index
Date																
2014-01-02	92600	35323.08	31200	1863630.47	219006.02	1288624.27	120922.26	407500	275505.22	73928.16	34147.79	99958.23	724229.68	96663.03	NaN	1967.189941
2014-01-03	89500	36068.30	30800	1793831.57	218518.26	1275826.63	120922.26	386500	272542.80	72156.24	33851.28	98973.42	698257.30	93495.32	NaN	1946.140015
2014-01-06	81500	37409.68	31300	1868616.10	222908.13	1286655.41	124886.92	382500	275998.96	72156.24	34839.64	99958.23	698257.30	94071.27	NaN	1953.280029
2014-01-07	81200	37558.72	31200	1828731.02	223395.90	1283702.10	125382.51	384500	274024.01	71467.17	35185.56	99958.23	699256.24	93303.34	NaN	1959.439941
2014-01-08	81200	38751.06	32000	1898529.91	223883.66	1270904.46	124391.34	386500	272542.80	72648.44	34987.89	99465.82	699256.24	94551.23	NaN	1958.959961

```
changes['AMOREPACIFIC'].plot(figsize=(16,4))
changes['KOSPI Composite Index'].plot(figsize=(16,4), color='r')
```

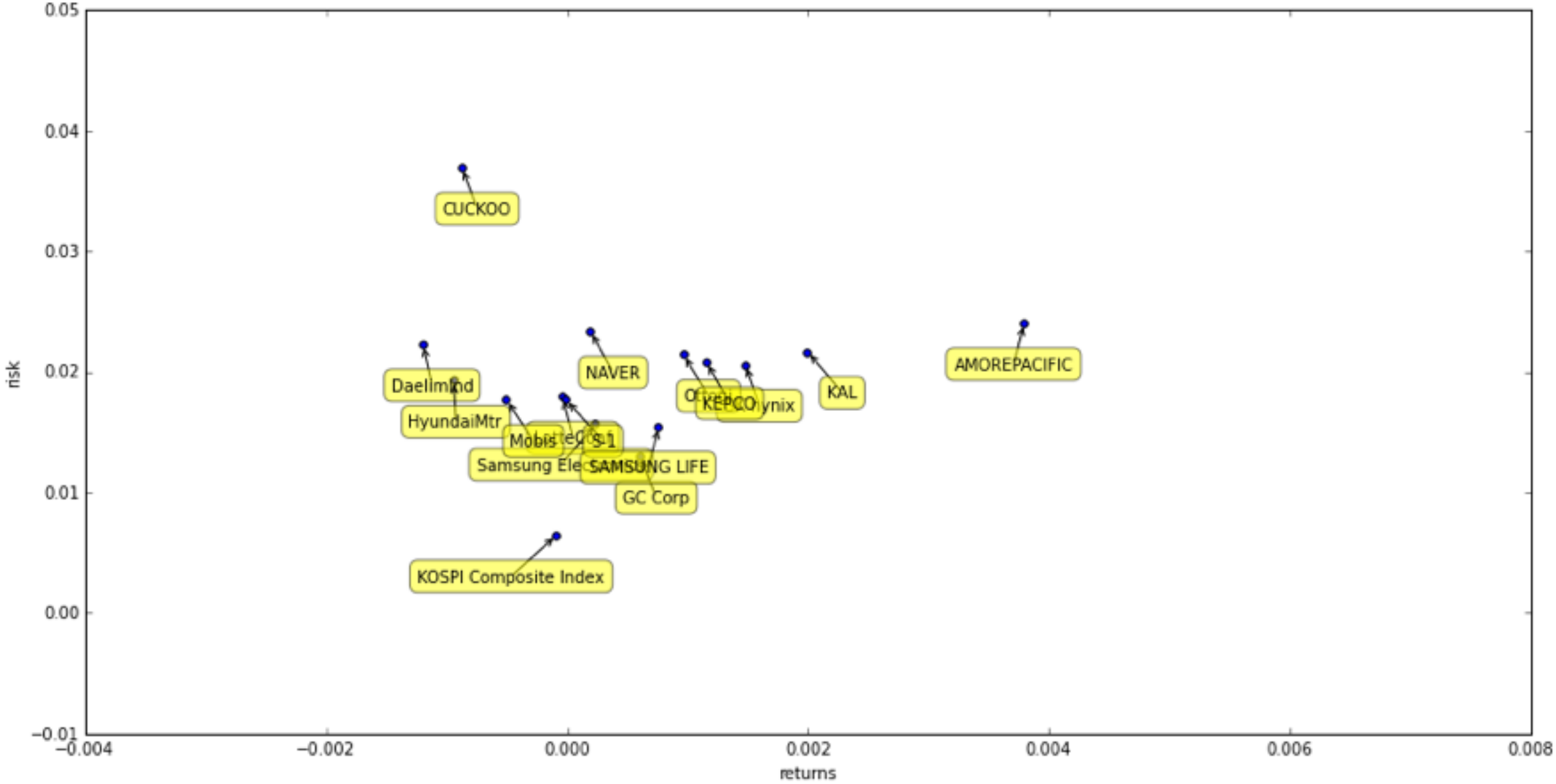


```
changes['AMOREPACIFIC'].cumsum().plot(figsize=(16,4), color='b')
changes['NAVER'].cumsum().plot(figsize=(16,4), color='g')
changes['KOSPI Composite Index'].cumsum().plot(figsize=(16,4), color='r')
```



스캐터 차트, 수익과 위험 분석

```
plt.figure(figsize=(16,8))

plt.scatter(changes.mean(), changes.std())
plt.xlabel('returns')
plt.ylabel('risk')

for label, x, y in zip(changes.columns, changes.mean(), changes.std()):
    plt.annotate( label, xy=(x, y), xytext=(30, -30),
      textcoords = 'offset points',
      ha = 'right', va = 'bottom',
      bbox = dict(boxstyle = 'round,pad=0.5', fc = 'yellow', alpha = 0.5),
      arrowprops = dict(arrowstyle = '->', connectionstyle = 'arc3,rad=0'))
```


0.05
0.04
0.03
0.02
0.01
0.00
−0.01
risk
−0.004
−0.002
0.000
0.002
0.004
0.006
0.008
returns
CUCKOO
NAVER
AMOREPACIFIC
KAL
KEPCO
HyundaiMtr
Mobis
S-1
SAMSUNG LIFE
GC Corp
KOSPI Composite Index

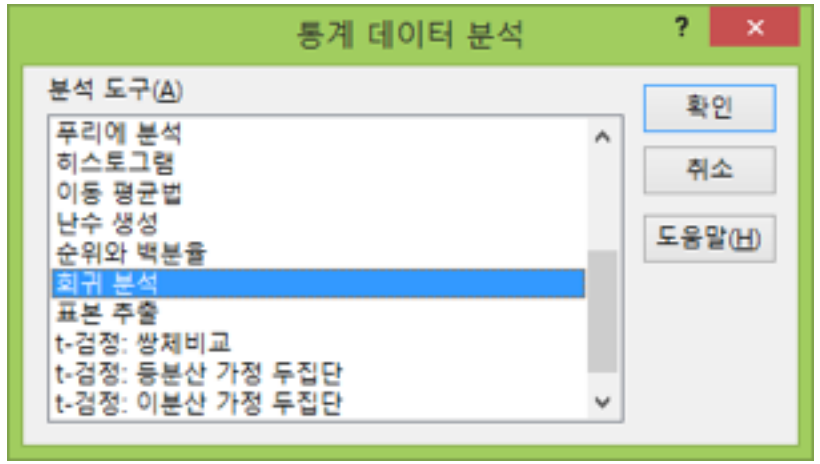
회귀분석 Regression analysis

주어진 데이터를 가장 잘 나타낼 수 있는 수식(간단한 직선 식)을 찾아내는 방법

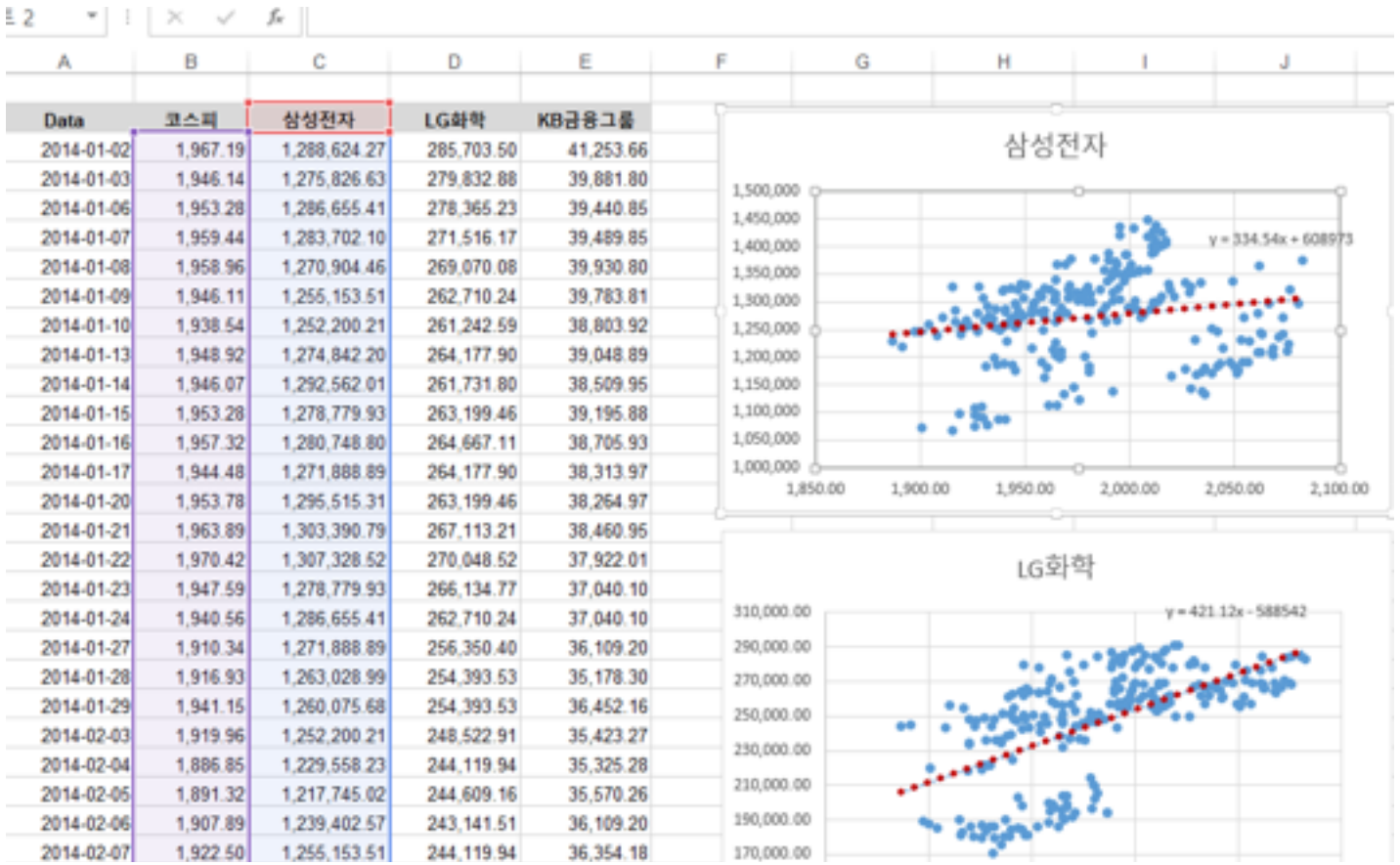
- A가 커짐에 따라 B는 얼마나 커지는가?
- 공분산과 상관계수는 같이 움직일 확률 정도를 알 수 있지, 즉 A가 1만큼 움직일 때, B는 얼마나 움직이는가'는 알 수 없다. (이를 위해 사용하는 것이 회귀분석)
- 값들 회귀 방정식을 구하고 기울기를 구할 수 있으면 X 값이 움직일 때, Y값이 얼마나(몇 배) 움직이는지를 표현하는 것이 베타(β)

엑셀 회귀분석

데이터 분석 도구, '회귀분석'

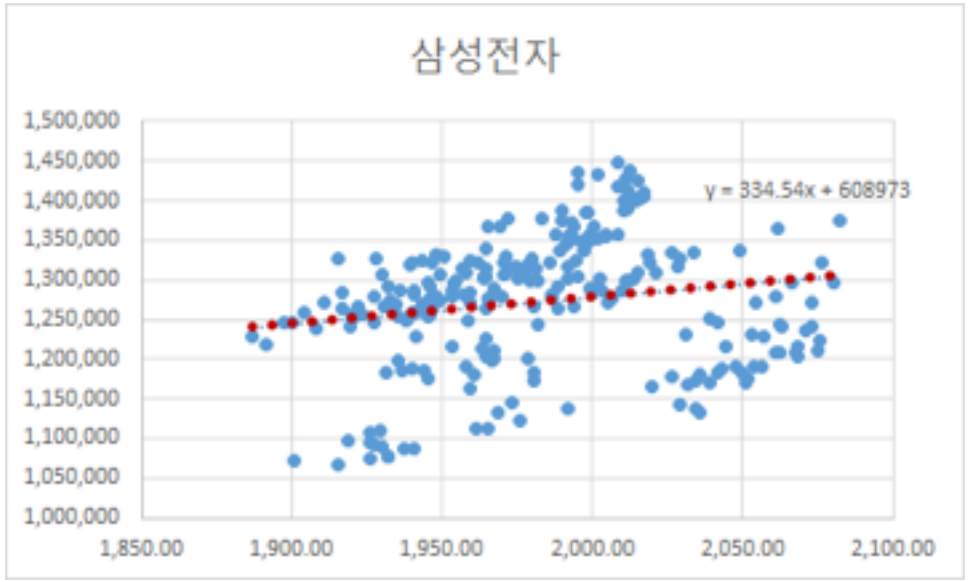


차트의 분산형을 추가

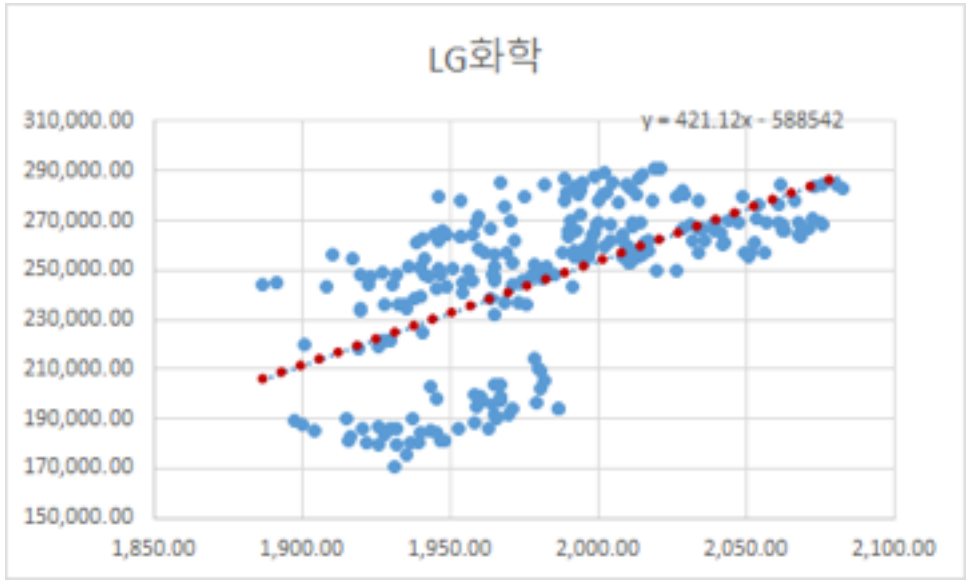


Data	코스피	삼성전자	LG화학	KB금융그룹
2014-01-02	1,967.19	1,288,624.27	285,703.50	41,253.66
2014-01-03	1,946.14	1,275,826.63	279,832.88	39,881.80
2014-01-06	1,953.28	1,286,655.41	278,365.23	39,440.85
2014-01-07	1,959.44	1,283,702.10	271,516.17	39,489.85
2014-01-08	1,958.96	1,270,904.46	269,070.08	39,930.80
2014-01-09	1,946.11	1,255,153.51	262,710.24	39,783.81
2014-01-10	1,938.54	1,252,200.21	261,242.59	38,803.92
2014-01-13	1,948.92	1,274,842.20	264,177.90	39,048.89
2014-01-14	1,946.07	1,292,562.01	261,731.80	38,509.95
2014-01-15	1,953.28	1,278,779.93	263,199.46	39,195.88
2014-01-16	1,957.32	1,280,748.80	264,667.11	38,705.93
2014-01-17	1,944.48	1,271,888.89	264,177.90	38,313.97
2014-01-20	1,953.78	1,295,515.31	263,199.46	38,264.97
2014-01-21	1,963.89	1,303,390.79	267,113.21	38,460.95
2014-01-22	1,970.42	1,307,328.52	270,048.52	37,922.01
2014-01-23	1,947.59	1,278,779.93	266,134.77	37,040.10
2014-01-24	1,940.56	1,286,655.41	262,710.24	37,040.10
2014-01-27	1,910.34	1,271,888.89	256,350.40	36,109.20
2014-01-28	1,916.93	1,263,028.99	254,393.53	35,178.30
2014-01-29	1,941.15	1,260,075.68	254,393.53	36,452.16
2014-02-03	1,919.96	1,252,200.21	248,522.91	35,423.27
2014-02-04	1,886.85	1,229,558.23	244,119.94	35,325.28
2014-02-05	1,891.32	1,217,745.02	244,609.16	35,570.26
2014-02-06	1,907.89	1,239,402.57	243,141.51	36,109.20
2014-02-07	1,922.50	1,255,153.51	244,119.94	36,354.18

추세선 trend line



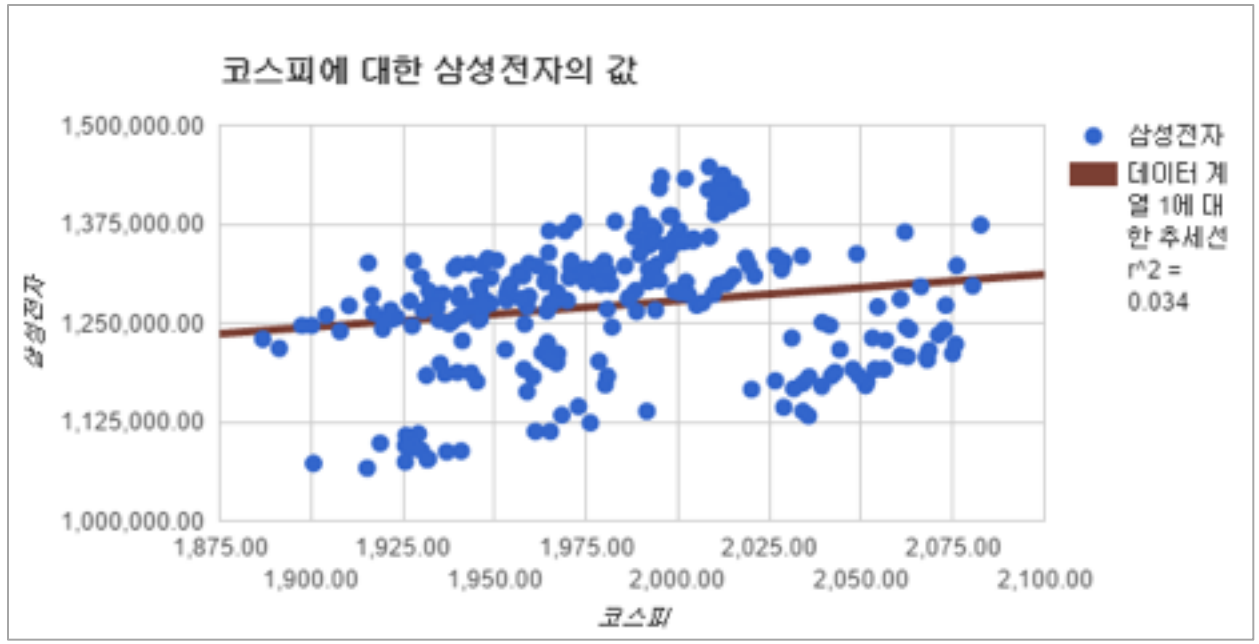
y = 334.54x + 608973



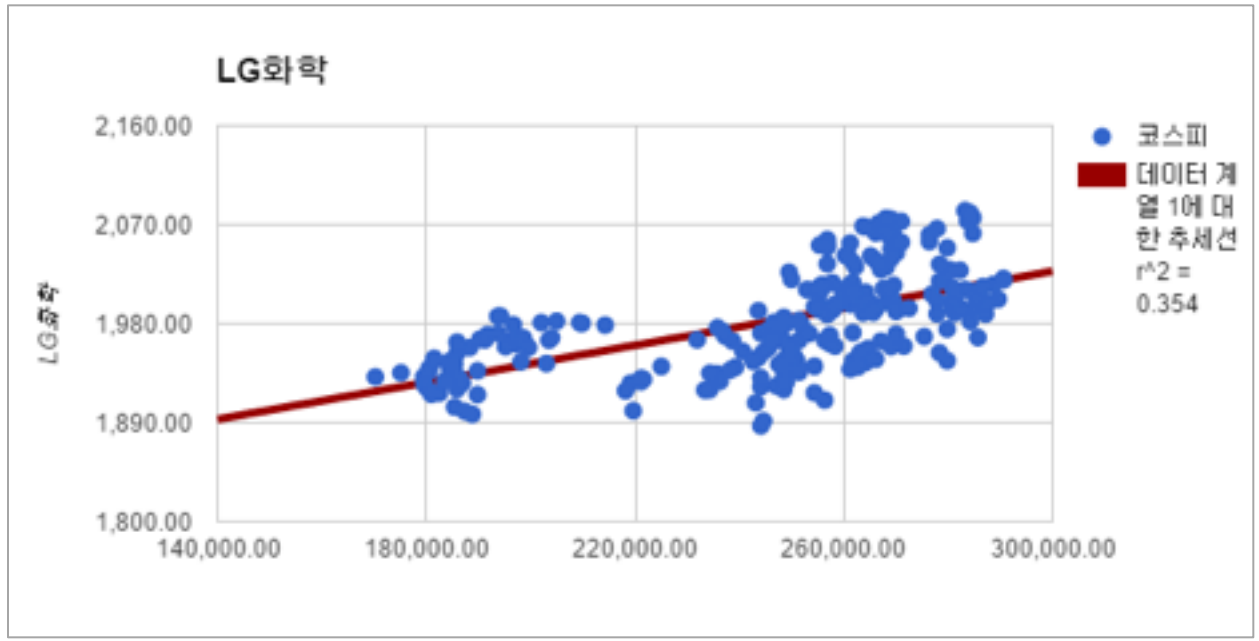
y = 421.12x - 588542

결정계수 coefficient of determination

- 결정계수 R^2, 회귀식의 예측도(얼마나 믿을 만 한지)를 표현
- $0 \leq R^2 \leq 1$



$R^2 = 0.034$



$R^2 = 0.0354$

종목 베타

아래와 같은 회귀식에서 β (베타)가 바로 직선의 기울기

$$y = \alpha + \beta x$$

종목A라는 종목이 있고 시장이라고 하면

- A의 베타 = 종목A와 시장M의 공분산 / 시장분산

스프레드시트에서 종목 베타의 계산

A의 베타 = COVARIANCE.P(시장M, 종목A) / VAR.P(시장M)

=COVARIANCE.P(E3:E124,F3:F124)/VAR.P(E3:E124)

=SLOPE(F3:F124, E3:E124) SLOPE(known_y's, known_x's)

모두 모아보기

```
code_names = {
    '^KS11':'KOSPI',
    '005930.KS':'Samsung',
    '005380.KS':'Hyundai',
}
```

```
start=datetime(2014, 1, 1)
end=datetime(2014, 12, 31)

p = data.get_data_yahoo(list(code_names.keys()),
start=start, end=end)
mask = p['Volume']['^KS11'] > 0
df = p['Adj Close'].ix[mask]
df = df.rename(columns=code_names)
chg = df.pct_change()
chg_corr = chg.corr()
```

파이썬에서 α, β 구하기

- Liner expression

$$y = \alpha + \beta x$$

  - beta = df['KOSPI'].cov(df['Samsung']) / df['KOSPI'].var()
  - alpha = df['Samsung'].mean() - beta * df['KOSPI'].mean()

리뷰

- 상관분석: 공분산, 상관계수, 베타
- 선형회귀 분석
- 종목 베타 (시장과 종목의 비교)
- 종목간 상대적 비교