

Pandas and Stock Price Statistics

Advanced Python

Prof. Groner

Pandas and Stock Price Statistics – Topics

- ▶ Daily returns
 - Calculating returns from prices
 - Charting: histogram
 - Statistics: testing for normal distribution
- ▶ Comparing a stock to the market – Beta
 - Charting: scatter chart
 - Statistics: regression
- ▶ Portfolio Allocation
 - Statistics: covariance matrix

Daily Returns

- ▶ Given a price time series, returns are given by:
 - ▶ Return = $(\text{Price}_t - \text{Price}_{t-1}) / \text{Price}_{t-1}$
- ▶ In Pandas, if x is a column, the returns are given by:
 - ▶ $xret = x.\text{pct_change}()[1:]$
 - ▶ Note: 1st entry is omitted, since returns begin at second date

Returns – Histogram Charting

- ▶ A histogram gives a visual review of a series' empirical distribution
- ▶ Using matplotlib, a histogram plot can be created using:
 - ▶ `plt.hist(data, bin)`
 - ▶ where bin is the number of bins (granularity) for the plot

Daily Returns – Testing for Normal Distribution

- ▶ Statistics for a series can be calculated by methods, including:
 - `mean()` – mean of the values
 - `std()` – standard deviation
 - `skew()` – skewness (leaning left or right)
 - `kurt()` – kurtosis (fat or thin tails)
- ▶ Normal distributions of 250 observations should have, at 95% confidence:
 - skewness within $(-.3, +.3)$
 - kurtosis with $(-.6, +.6)$

Stock Returns – Beta

- ▶ An individual stock returns are generally influenced by the overall market
- ▶ An individual stock will also be influenced by its own company's news
- ▶ The sensitivity of the stock to the market is Beta
- ▶ Beta can be estimated by a regression between a broad index's returns (e.g., S&P500) and the individual stock's returns; Beta is the line's slope

Stock Returns – Beta

- ▶ To perform a regression, Python can use a statistics package, for example statsmodels
- ▶ For example:
 - ▶ `import statsmodels.api as sm`
 - ▶ `# (in this fragment, omitting lines to get/setup data)`
 - ▶ `# ...`
 - ▶ `# OLS = ordinary least-squares regression`
 - ▶ `model = sm.OLS(yret, xret).fit()`
 - ▶ `print(model.summary())`

Asset Allocation – Covariance

- ▶ One approach to asset allocation (mean–variance optimization) is based on:
 - ▶ historical returns: mean, standard deviation
 - ▶ correlations of returns among asset choices
- ▶ pandas can be used to calculate the correlation matrix related to all series in a DataFrame
 - ▶ `corrmat = frame.corr()`
- ▶ pandas methods `mean()` and `std()` can be used on a column, or on the set of columns in a DataFrame

Pandas and Stock Price Statistics – Recap

- ▶ Daily returns
 - Calculating returns from prices
 - Charting: histogram
 - Statistics: testing for normal distribution
- ▶ Comparing a stock to the market – Beta
 - Charting: scatter chart
 - Statistics: regression
- ▶ Portfolio Allocation
 - Statistics: covariance matrix