Risk & Asset Allocation (Spring)
Case for Week 5

John A. Dodson

February 18, 2015

Example of allocation-implied calibration

```
%% load historical price data
try
    price; % do not re-load data if it is already in the workspace
catch
    [price last]=yahoo_prices({'ATVI' 'ADBE' 'AKAM' 'ALXN' 'ALTR' 'AMZN'...
        'AAL' 'AMGN' 'ADI' 'AAPL' 'AMAT' 'ADSK' 'ADP' 'AVGO' 'BIDU' 'BBBY'...
        'BIIB' 'BRCM' 'CA' 'CTRX' 'CELG' 'CERN' 'CHRW' 'CHTR' 'CHKP' 'CSCO'...
        'CTXS' 'CTSH' 'CMCSA' 'CMCSK' 'COST' 'DTV' 'DISCK' 'DISCA' 'DISH'...
        'DLTR' 'EBAY' 'EA' 'EQIX' 'EXPD' 'ESRX' 'FB' 'FAST' 'FISV' 'GRMN'...
        'GILD' 'GOOGL' 'HSIC' 'ILMN' 'INTC' 'INTU' 'ISRG' 'GMCR'...
        'KLAC' 'KRFT' 'LRCX' 'LBTYK' 'LBTYA' 'QVCA' 'LMCA' 'LVNTA'...
        'LLTC' 'MAR' 'MAT' 'MU' 'MSFT' 'MDLZ' 'MNST' 'MYL' 'NTAP' 'NFLX'...
        'NVDA' 'NXPI' 'ORLY' 'PCAR' 'PAYX' 'PCLN' 'QCOM' 'REGN' 'ROST'...
        'SNDK' 'SBAC' 'STX' 'SIAL' 'SIRI' 'SPLS' 'SBUX' 'SRCL' 'SYMC'...
        'TSLA' 'TXN' 'TSCO' 'TRIP' 'FOX' 'FOXA' 'VRSK' 'VRTX' 'VIAB'...
        'VIP' 'VOD' 'WDC' 'WFM' 'WYNN' 'XLNX' 'YHOO'},'13-Feb-2013','13-Feb-2015');
% holding out 'GOOG' and 'LMCK' for limited histories
    tickers=gettimeseriesnames(price);
    price=resample(price,price.Time(~isnan(price.(tickers{1}).Data)));
end
%% extract arrays
y=[];p=[];
for ticker=tickers;ticker=ticker{:};
    ts=price.(ticker); % adjusted prices
    y=[y diff(log(ts.Data))]; % daily log-returns
    p=[p;last.(ticker)]; % latest close price
end
Sigma=cov(y,1); % (unconditional) covariance of daily log-returns
%% fit normal-exponential equilibrium model
alphaCap=[716.265;502.47;178.081;199.292;306.38;462.312;724.567;770.015
        314.629;5859.031;1234.324;229.035;476.801;258.866;355.745;186.646
        235.706;592.946;442.667;208.759;807.919;343.388;146.866;109.338
        185.194;5154.868;160.971;607.754;2588.097;2610.57;441.452;503.322
        692.473;668.766;460.55;206.586;1208.271;313.634;55.761;193.181
        736.6;2839.504;297.103;244.6;191.155;1530.641;1367.387;84.492
        144.049;4877.839;290.275;36.776;159.43;164.769;589.573;159.765
        908.934;881.648;481.954;687.875;142.785;240.432;288.133;339.427
        1097.25;8211.523;1685.069;167.981;377.537;313.377;60.388;545.935
        231.958;101.256;356.813;363.528;52.192;1668.952;101.995;208.662
        222.614;130.017;332.501;119.111;5603.945;640.96;750.39;86.085
        696.694;126.162;1070.291;136.078;142.82;2181.839;2115.89;165.433
        242.138;406.813;1791.944;2662.368;235.5;360.642;101.703;263.229
        949.116]*1e6; % share float
% GOOG combined with GOOGL and LMCK combined with LMCA
```

```
lambda=0.8/sqrt(252); % equilibrium market price of risk (per day)
EP=p; % initital candidate for 'EP'; we need to iterate here a few times
for i=1:4 % ... because 'covP' depends on 'EP' and vice versa
    covP=EP*EP'.*(exp(Sigma)-1);
    zetaCap=sqrt(alphaCap'*covP*alphaCap)/lambda; % eq. utility parameter
    EP=p+covP*alphaCap/zetaCap;
end
zeta=zetaCap/(alphaCap'*p); % absolute risk aversion per dollar (about 17%)
%% translate to log-normal model
mu=log(EP./p)-diag(Sigma)/2; % implied daily drift in log-returns
```

Example of index of satisfaction

```
function ES_3=satisfaction_cf(p,mu,Sigma,alpha,c)
% coherent index of satisfaction with confidence 'c' for log-normal
% market vector 'M=p*(exp(X)-1)' where 'X~N(mu,Sigma)' and
% portfolio 'alpha' using the Cornish-Fisher approximation
% JDodson 20140216
%% calculate the moments of the market vector
EM=p.*(exp(mu+diag(Sigma)/2)-1);
covM=(p+EM)*(p+EM)'.*(exp(Sigma)-1);
%% calculate third central moment for skew
psiM3=0;
for j=1:length(alpha)
    for k=1:length(alpha)
       for l=1:length(alpha)
        psiM3=psiM3+alpha(j)*alpha(k)*alpha(l)...
            *(p(j)+EM(j))*(p(k)+EM(k))*(p(l)+EM(l))*(...
            2+exp(Sigma(j,k)+Sigma(j,l)+Sigma(k,l))...
            -exp(Sigma(j,k))-exp(Sigma(j,l))-exp(Sigma(k,l)));
       end
    end
end
%% calculate value-at-risk and expected shortfall
% central moments of the obejctive
EPsi=EM'*alpha;
sdPsi=sqrt(alpha'*covM*alpha);
skPsi=psiM3/sdPsi^3;
% normal factors for Cornish-Fisher expansion
z1=@(c)sqrt(2)*erfinv(2*c-1);
z2=@(c)z1(c).^2;
z1_ES=@(c)quad(z1,c,1-eps)/(1-c);
z2_ES=@(c)quad(z2,c,1-eps)/(1-c);
% 2nd-order approximations
VaR_2=EPsi-z1(c)*sdPsi;
ES_2=EPsi-z1_ES(c)*sdPsi;
% 3rd-order approximations
VaR_3=VaR_2+(z2(c)-1)/6*skPsi*sdPsi;
ES_3=ES_2+(z2_ES(c)-1)/6*skPsi*sdPsi;
```