Portfolio Performance Attribution Analysis

In this example, we show how simple and powerful it is to use Matlab (especailly table operation) to perform Performance Attribution analysis.

Single Period Brinson Model

Brinson model: *value added return* can be segmented into the impact of assigning the asset weight of the portfolio to various economic sectors and the impact of selecting securities within those sectors.

$$R_{\mathrm{ValueAdded}} = \sum_{i=1}^{N_s} R_{\mathrm{Pi}} W_{\mathrm{Pi}} - \sum_{i=1}^{N_s} R_{\mathrm{Bi}} W_{\mathrm{Bi}}$$

$$= \sum_{i=1}^{N_s} R_{\mathrm{Bi}} (W_{\mathrm{Pi}} - W_{\mathrm{Bi}})$$ Pure sector allocation, portfolio managers' responsibility

$$+ \sum_{i=1}^{N_s} (R_{\mathrm{Pi}} - R_{\mathrm{Bi}})\, W_{\mathrm{Bi}}$$ Within-sector selection, security analysts' responsibility

$$+ \sum_{i=1}^{N_s} (R_{\mathrm{Pi}} - R_{\mathrm{Bi}})(W_{\mathrm{Pi}} - W_{\mathrm{Bi}})$$ Allocation/selection interaction, joint effect

where

N_s is the total number of sectors

R_{Pi} is the return of sector i in the portfolio

W_{Pi} is the allocated weight for sector i in the portfolio

R_{Bi} is the return of sector i in the benchmark

W_{Bi} is the allocated weight for sector i in the benchmark

R_B is the overall benchmark return.

R_{Pi} , W_{Pi}, R_{Bi}, W_{Bi} are computed by aggregating all asset weights and returns in the given portfolio and benchmark as follows:

$$W_{\mathrm{Pi}} = \sum_j w_{\mathrm{pj}},\ \ R_{\mathrm{Pi}} = \frac{\sum_j r_{\mathrm{Pj}} w_{\mathrm{Pj}}}{\sum_j w_{\mathrm{pj}}},\ \ W_{\mathrm{Bi}} = \sum_j w_{\mathrm{Bj}},\ \ R_{\mathrm{Bi}} = \frac{\sum_j r_{\mathrm{Bj}} w_{\mathrm{Bj}}}{\sum_j w_{\mathrm{Bj}}}$$

where

r_{Pj}, w_{Pj} are the return and weight of asset j that belongs to sector i in the portfolio

r_{Bj}, w_{Bj} are the return and weight of asset j that belongs to sector i in the benchmark.

Factor-based Performance Attribution

Factor models decompose *asset returns* into a systematic component that is explained by factors and a residual component that is not. One big advantage of this approach is that it allows one to define their own attribution model by easily incorporating multiple variables (factors).

$r_i = \sum_k X_{\text{ik}} f_k + u_i$,

where X_{ik} is the exposure of stock i to factor k (or factor loadings), f_k is the return for factor k, u_i is the residual return of the stock. *It is assumed that the stock exposures are known at the start of the evaluation period, and the factor returns are estimated by cross-sectional regression at the end of the evaluation period.* Active return can be attributed by aggregating asset returns to the factor level.

$$R_{\text{ValueAdded}} = \sum_k X_k^A f_k + \sum_i w_i^A u_i \quad ,$$

$$X_k^A = \sum_i w_i^A X_{\text{ik}} = \sum_i (w_{\text{Pi}} - w_{\text{Bi}}) X_{\text{ik}}$$

Where w_i^A is the active weight of asset i, X_k^A is the active exposure to factor k for all assets in the portfolio.

Load data

The dataset is simulated and used for demonstration purposes in this example.

```
load('demo_data.mat');
portData(1:7, :)
```

ans = 7×5 table

	AssetTicker	AssetWgt	AssetRetn	Sector	Security
1	A	0.0100	-0.0031	Health Care	'Agilent Te...
2	ALGN	0.0100	0.0911	Health Care	'Align Tech...
3	AON	0.0100	-0.0667	Financials	'Aon plc'
4	APC	0.0100	-0.4817	Energy	'Anadarko P...
5	AXP	0.0100	-0.1115	Financials	'American E...
6	BK	0.0100	-0.0476	Financials	'The Bank o...
7	BSX	0.0100	0.0313	Health Care	'Boston Sci...

```
benchmarkData(1:7, :)
```

ans = 7×5 table

	AssetTicker	AssetWgt	AssetRetn	Sector	Security
1	A	0.0022	-0.0031	Health Care	'Agilent Te...
2	AAP	0.0022	0.0348	Consumer Di...	'Advance Au...

	AssetTicker	AssetWgt	AssetRetn	Sector	Security
3	AAPL	0.0022	-0.1821	Information...	'Apple Inc.'
4	ABBV	0.0022	-0.0841	Health Care	'AbbVie Inc.'
5	ABC	0.0022	-0.1001	Health Care	'Amerisourc...
6	ABT	0.0022	-0.0445	Health Care	'Abbott Lab...
7	ACN	0.0022	0.1065	Information...	'Accenture ...

Some analysis and visualization

Use the above data which is on the asset level, and group them to the sector level, and conduct the performance analysis.

```
unique(portData.Sector)
```

```
ans = 11×1 categorical array
    Communication Services
    Consumer Discretionary
    Consumer Staples
    Energy
    Financials
    Health Care
    Industrials
    Information Technology
    Materials
    Real Estate
    Utilities
```

```
format compact
summary(portData)
```

```
Variables:
    AssetTicker: 70×1 categorical
        Properties:
            Description:  Original column heading: 'Ticker symbol'
        Values:
            A             1
            ALGN          1
            AON           1
            APC           1
            AXP           1
            BK            1
            BSX           1
            CA            1
            CBOE          1
            CI            1
            CLX           1
            CMA           1
            COF           1
            COST          1
            CVS           1
            DIS           1
            EMR           1
            EQT           1
            FL            1
            FOXA          1
```

```
        GD            1
        GM            1
        HAS           1
        HCP           1
        IDXX          1
        IPG           1
        IR            1
        JCI           1
        K             1
        KLAC          1
        LEN           1
        LNT           1
        LRCX          1
        LUV           1
        LYB           1
        M             1
        MA            1
        MDT           1
        MMC           1
        NEE           1
        NFLX          1
        NI            1
        NOV           1
        NWS           1
        PGR           1
        PKI           1
        PNW           1
        PRGO          1
        PX            1
        RJF           1
        ROST          1
        SBAC          1
        SEE           1
        SJM           1
        SLG           1
        SPG           1
        STT           1
        SWK           1
        T             1
        TDG           1
        TMO           1
        TRIP          1
        TRV           1
        TSCO          1
        UAA           1
        VAR           1
        VTR           1
        XLNX          1
        XRX           1
        XYL           1
AssetWgt: 70×1 double
    Values:
        Min          0.01
        Median       0.01
        Max         0.157
AssetRetn: 70×1 double
    Values:
        Min          -0.4817
        Median    -0.0032875
        Max           0.4429
Sector: 70×1 categorical
    Properties:
        Description:  Original column heading: 'GICS Sector'
    Values:
        Communication Services        7
```

```
        Consumer Discretionary        8
        Consumer Staples              4
        Energy                        3
        Financials                   11
        Health Care                  11
        Industrials                   8
        Information Technology        6
        Materials                     3
        Real Estate                   5
        Utilities                     4
    Security: 70×1 cell array of character vectors
```

```
summary(benchmarkData)
```

```
Variables:
    AssetTicker: 460×1 categorical
        Properties:
            Description:  Original column heading: 'Ticker symbol'
        Values:
            A              1
            AAP            1
            AAPL           1
            ABBV           1
            ABC            1
            ABT            1
            ACN            1
            ADBE           1
            ADI            1
            ADM            1
            ADP            1
            ADS            1
            ADSK           1
            AEE            1
            AEP            1
            AES            1
            AET            1
            AFL            1
            AGN            1
            AIG            1
            AIV            1
            AIZ            1
            AJG            1
            AKAM           1
            ALB            1
            ALGN           1
            ALK            1
            ALL            1
            ALXN           1
            AMAT           1
            AME            1
            AMG            1
            AMGN           1
            AMP            1
            AMT            1
            AMZN           1
            ANSS           1
            ANTM           1
            AON            1
            AOS            1
            APA            1
            APC            1
            APD            1
            APH            1
            APTV           1
```

ARE	1
ATVI	1
AVB	1
AVGO	1
AVY	1
AWK	1
AXP	1
AZO	1
BA	1
BAC	1
BAX	1
BBT	1
BBY	1
BDX	1
BEN	1
BHGE	1
BIIB	1
BK	1
BKNG	1
BLK	1
BLL	1
BMY	1
BSX	1
BWA	1
BXP	1
C	1
CA	1
CAG	1
CAH	1
CAT	1
CB	1
CBOE	1
CBS	1
CCI	1
CCL	1
CDNS	1
CELG	1
CERN	1
CF	1
CHD	1
CHRW	1
CHTR	1
CI	1
CINF	1
CL	1
CLX	1
CMA	1
CMCSA	1
CME	1
CMG	1
CMI	1
CMS	1
CNC	1
CNP	1
COF	1
COG	1
COL	1
COO	1
COP	1
COST	1
COTY	1
CPB	1
CRM	1
CSCO	1
CTAS	1

CTL	1
CTSH	1
CTXS	1
CVS	1
CVX	1
CXO	1
D	1
DAL	1
DE	1
DFS	1
DG	1
DGX	1
DHI	1
DHR	1
DIS	1
DISCA	1
DISCK	1
DISH	1
DLR	1
DLTR	1
DOV	1
DRE	1
DRI	1
DTE	1
DUK	1
DVA	1
DVN	1
DWDP	1
EA	1
EBAY	1
ECL	1
ED	1
EFX	1
EIX	1
EL	1
EMN	1
EMR	1
EOG	1
EQIX	1
EQR	1
EQT	1
ES	1
ESRX	1
ESS	1
ETFC	1
ETN	1
ETR	1
EW	1
EXC	1
EXPD	1
EXPE	1
EXR	1
F	1
FAST	1
FB	1
FBHS	1
FCX	1
FDX	1
FE	1
FFIV	1
FIS	1
FISV	1
FITB	1
FL	1
FLIR	1

FLR	1
FLS	1
FMC	1
FOX	1
FOXA	1
FRT	1
FTI	1
GD	1
GE	1
GILD	1
GIS	1
GLW	1
GM	1
GOOGL	1
GPC	1
GPN	1
GPS	1
GRMN	1
GS	1
GT	1
GWW	1
HAL	1
HAS	1
HBAN	1
HBI	1
HCA	1
HCP	1
HD	1
HES	1
HIG	1
HII	1
HOG	1
HOLX	1
HON	1
HP	1
HPQ	1
HRB	1
HRL	1
HRS	1
HSIC	1
HST	1
HSY	1
HUM	1
IBM	1
ICE	1
IDXX	1
IFF	1
ILMN	1
INCY	1
INTC	1
INTU	1
IP	1
IPG	1
IQV	1
IR	1
IRM	1
ISRG	1
IT	1
ITW	1
IVZ	1
JBHT	1
JCI	1
JEC	1
JNJ	1
JNPR	1

JPM	1
JWN	1
K	1
KEY	1
KIM	1
KLAC	1
KMB	1
KMI	1
KMX	1
KO	1
KORS	1
KR	1
KSS	1
KSU	1
L	1
LB	1
LEG	1
LEN	1
LH	1
LKQ	1
LLL	1
LLY	1
LMT	1
LNC	1
LNT	1
LOW	1
LRCX	1
LUV	1
LYB	1
M	1
MA	1
MAA	1
MAC	1
MAR	1
MAS	1
MAT	1
MCD	1
MCHP	1
MCK	1
MCO	1
MDLZ	1
MDT	1
MET	1
MGM	1
MHK	1
MKC	1
MLM	1
MMC	1
MMM	1
MNST	1
MO	1
MOS	1
MPC	1
MRK	1
MRO	1
MS	1
MSFT	1
MSI	1
MTB	1
MTD	1
MU	1
MYL	1
NBL	1
NDAQ	1
NEE	1

NEM	1
NFLX	1
NFX	1
NI	1
NKE	1
NLSN	1
NOC	1
NOV	1
NRG	1
NSC	1
NTAP	1
NTRS	1
NUE	1
NVDA	1
NWL	1
NWS	1
NWSA	1
O	1
OKE	1
OMC	1
ORCL	1
ORLY	1
OXY	1
PAYX	1
PBCT	1
PCAR	1
PCG	1
PEG	1
PFE	1
PG	1
PGR	1
PH	1
PHM	1
PKG	1
PKI	1
PLD	1
PM	1
PNC	1
PNR	1
PNW	1
PPG	1
PPL	1
PRGO	1
PRU	1
PSA	1
PSX	1
PVH	1
PWR	1
PX	1
PXD	1
QCOM	1
RCL	1
RE	1
REG	1
REGN	1
RF	1
RHI	1
RHT	1
RJF	1
RL	1
RMD	1
ROK	1
ROP	1
ROST	1
RSG	1

RTN	1
SBAC	1
SBUX	1
SCG	1
SCHW	1
SEE	1
SHW	1
SJM	1
SLB	1
SLG	1
SNA	1
SNPS	1
SO	1
SPG	1
SPGI	1
SRCL	1
SRE	1
STI	1
STT	1
STX	1
STZ	1
SWK	1
SWKS	1
SYK	1
SYMC	1
SYY	1
T	1
TAP	1
TDG	1
TEL	1
TGT	1
TIF	1
TJX	1
TMK	1
TMO	1
TPR	1
TRIP	1
TROW	1
TRV	1
TSCO	1
TSN	1
TSS	1
TXN	1
TXT	1
UAA	1
UAL	1
UDR	1
UHS	1
ULTA	1
UNH	1
UNM	1
UNP	1
UPS	1
URI	1
USB	1
UTX	1
V	1
VAR	1
VFC	1
VIAB	1
VLO	1
VMC	1
VNO	1
VRSK	1
VRSN	1

```
        VRTX           1
        VTR            1
        VZ             1
        WAT            1
        WDC            1
        WEC            1
        WELL           1
        WFC            1
        WHR            1
        WM             1
        WMB            1
        WMT            1
        WU             1
        WY             1
        WYNN           1
        XEC            1
        XLNX           1
        XOM            1
        XRAY           1
        XRX            1
        XYL            1
        YUM            1
        ZBH            1
        ZION           1
        ZTS            1
AssetWgt: 460×1 double
    Values:
        Min       0.0021739
        Median    0.0021739
        Max       0.0021739
AssetRetn: 460×1 double
    Values:
        Min        -0.71056
        Median     -0.02248
        Max         0.66925
Sector: 460×1 categorical
    Properties:
        Description:  Original column heading: 'GICS Sector'
    Values:
        Communication Services       23
        Consumer Discretionary       62
        Consumer Staples             28
        Energy                       29
        Financials                   58
        Health Care                  60
        Industrials                  63
        Information Technology       56
        Materials                    23
        Real Estate                  31
        Utilities                    27
Security: 460×1 cell array of character vectors
```

Very easy to query from the table, e.g. find me the assets with Sector in 'Engery', or sort by 'AssetRetn':

```
rows = portData.Sector=='Energy';
portData(rows, :)
```

ans = 3×5 table

	AssetTicker	AssetWgt	AssetRetn	Sector	Security
1	APC	0.0100	-0.4817	Energy	'Anadarko P...
2	EQT	0.0100	-0.4188	Energy	'EQT Corpor...

	AssetTicker	AssetWgt	AssetRetn	Sector	Security
3	NOV	0.0100	-0.3882	Energy	'National O...

```
sortrows(portData, 'Sector')
```

ans = 70×5 table

	AssetTicker	AssetWgt	AssetRetn	Sector	Security
1	DIS	0.0100	-0.0536	Communicati...	'The Walt D...
2	FOXA	0.0100	-0.2121	Communicati...	'Twenty-Fir...
3	IPG	0.0100	0.1049	Communicati...	'Interpubli...
4	NFLX	0.0100	0.4429	Communicati...	'Netflix Inc.'
5	NWS	0.0100	-0.1313	Communicati...	'News Corp....
6	T	0.0449	-0.0035	Communicati...	'AT&T Inc.'
7	TRIP	0.0100	0.0668	Communicati...	'TripAdvisor'
8	FL	0.0100	0.0744	Consumer Di...	'Foot Locke...
9	GM	0.0100	-0.0393	Consumer Di...	'General Mo...
10	HAS	0.0100	-0.0539	Consumer Di...	'Hasbro Inc.'
11	LEN	0.0100	0.0543	Consumer Di...	'Lennar Corp.'
12	M	0.0100	-0.4642	Consumer Di...	'Macy's Inc.'
13	ROST	0.0100	0.0626	Consumer Di...	'Ross Stores'
14	TSCO	0.0100	-0.0210	Consumer Di...	'Tractor Su...
15	UAA	0.0100	0.0454	Consumer Di...	'Under Armo...
16	CLX	0.1281	0.1908	Consumer St...	'The Clorox...
17	COST	0.0100	0.1081	Consumer St...	'Costco Who...
18	K	0.1570	0.1269	Consumer St...	'Kellogg Co.'
19	SJM	0.0100	0.0420	Consumer St...	'JM Smucker'
20	APC	0.0100	-0.4817	Energy	'Anadarko P...
21	EQT	0.0100	-0.4188	Energy	'EQT Corpor...
22	NOV	0.0100	-0.3882	Energy	'National O...
23	AON	0.0100	-0.0667	Financials	'Aon plc'
24	AXP	0.0100	-0.1115	Financials	'American E...
25	BK	0.0100	-0.0476	Financials	'The Bank o...
26	CBOE	0.0100	0.1687	Financials	'Cboe Globa...
27	CMA	0.0100	-0.1332	Financials	'Comerica I...
28	COF	0.0100	-0.1225	Financials	'Capital On...
29	MMC	0.0100	-0.0280	Financials	'Marsh & Mc...

	AssetTicker	AssetWgt	AssetRetn	Sector	Security
30	PGR	0.0100	0.1717	Financials	'Progressiv...
31	RJF	0.0100	0.0061	Financials	'Raymond Ja...
32	STT	0.0100	-0.1572	Financials	'State Stre...
33	TRV	0.0100	0.1046	Financials	'The Travel...
34	A	0.0100	-0.0031	Health Care	'Agilent Te...
35	ALGN	0.0100	0.0911	Health Care	'Align Tech...
36	BSX	0.0100	0.0313	Health Care	'Boston Sci...
37	CI	0.0100	0.1547	Health Care	'CIGNA Corp.'
38	CVS	0.0100	-0.0239	Health Care	'CVS Health'
39	IDXX	0.0100	0.1174	Health Care	'IDEXX Labo...
40	MDT	0.0100	0.0134	Health Care	'Medtronic ...
41	PKI	0.0100	0.0290	Health Care	'PerkinElmer'
42	PRGO	0.0100	-0.2301	Health Care	'Perrigo'
43	TMO	0.0100	0.1101	Health Care	'Thermo Fis...
44	VAR	0.0100	-0.0859	Health Care	'Varian Med...
45	EMR	0.0100	-0.1960	Industrials	'Emerson El...
46	GD	0.0100	-0.0230	Industrials	'General Dy...
47	IR	0.0100	-0.1734	Industrials	'Ingersoll-...
48	JCI	0.0100	-0.2001	Industrials	'Johnson Co...
49	LUV	0.0100	0.0294	Industrials	'Southwest ...
50	SWK	0.0100	0.0615	Industrials	'Stanley Bl...
51	TDG	0.0100	0.0371	Industrials	'TransDigm ...
52	XYL	0.0100	-0.0108	Industrials	'Xylem Inc.'
53	CA	0.0100	-0.1019	Information...	'CA, Inc.'
54	KLAC	0.0100	0.1504	Information...	'KLA-Tencor...
55	LRCX	0.0100	0.0329	Information...	'Lam Research'
56	MA	0.0100	0.0678	Information...	'Mastercard...
57	XLNX	0.0100	0.0613	Information...	'Xilinx'
58	XRX	0.0100	-0.0969	Information...	'Xerox'
59	LYB	0.0100	-0.1634	Materials	'LyondellBa...
60	PX	0.0100	-0.1625	Materials	'Praxair Inc.'
61	SEE	0.0100	-0.0595	Materials	'Sealed Air'
62	HCP	0.0100	-0.0616	Real Estate	'HCP Inc.'
63	SBAC	0.0100	-0.0968	Real Estate	'SBA Commun...

	AssetTicker	AssetWgt	AssetRetn	Sector	Security
64	SLG	0.0100	-0.0863	Real Estate	'SL Green R...
65	SPG	0.0100	0.0604	Real Estate	'Simon Prop...
66	VTR	0.0100	-0.0632	Real Estate	'Ventas Inc'
67	LNT	0.0100	0.0094	Utilities	'Alliant En...
68	NEE	0.0100	0.0110	Utilities	'NextEra En...
69	NI	0.0100	0.1248	Utilities	'NiSource I...
70	PNW	0.0100	0.0557	Utilities	'Pinnacle W...

Brinson attribution analysis

Let's analyze the performance by sector: need to group assets by sector for both the given portfolio and benchmark, and compute the sector weghts and returns. This can be easily done using *findgroups* for table operations. 'pa_brinson' function implements the single period brinson model.

```
results = pa_brinson(portData, benchmarkData)
```

```
results = 12×9 table
```

...

	Sector	PortRetn	BenchRetn	PortWgt	BenchWgt	PureSectorA...
1	Communicati...	1.9253	-1.1249	10.4938	5.0000	0.1503
2	Consumer Di...	-4.2695	-2.2786	8.0000	13.4783	-0.0867
3	Consumer St...	15.0331	9.2173	30.5062	6.0870	3.1936
4	Energy	-42.9559	-30.5781	3.0000	6.3043	0.8828
5	Financials	-1.9611	-3.0001	11.0000	12.6087	-0.0138
6	Health Care	1.8558	0.2137	11.0000	13.0435	-0.0833
7	Industrials	-5.9437	-5.3316	8.0000	13.6957	0.0838
8	Information...	1.8943	-2.3853	6.0000	12.1739	-0.0911
9	Materials	-12.8452	-15.0496	3.0000	5.0000	0.2238
10	Real Estate	-4.9497	3.4787	5.0000	6.7391	-0.1276
11	Utilities	5.0211	-4.1217	4.0000	5.8696	0.0049
12	Summary	2.3524	-3.8610	100.0000	100.0000	4.1366

The last row shows the total portfolio return and benchmark return, also the total effects of sector allocation and stock selection.

```
results(end, :)
```

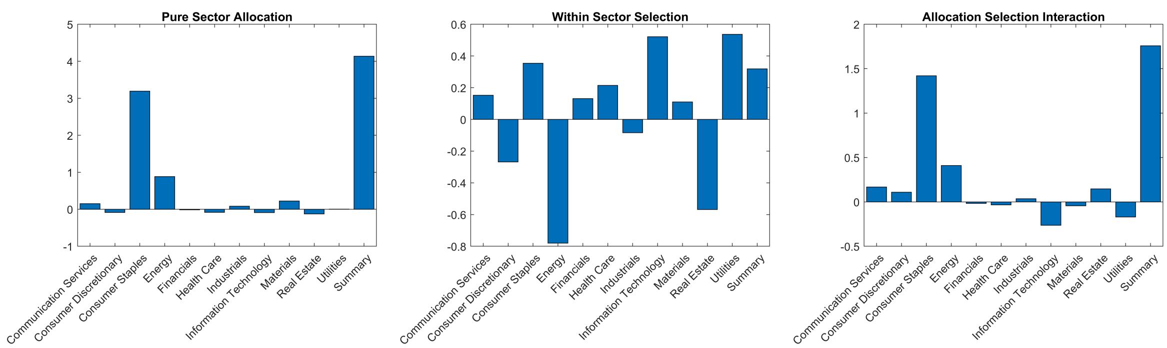
```
ans = 1×9 table
```

...

	Sector	PortRetn	BenchRetn	PortWgt	BenchWgt	PureSectorA...
1	Summary	2.3524	-3.8610	100.0000	100.0000	4.1366

We can also plot the active returns resulting from allocation/selection/interaction effects for each sector.

```
f=figure;
f.Position(3)= 3*f.Position(3);
subplot(1,3,1);
bar(results.Sector, results.PureSectorAllocation);
title('Pure Sector Allocation');
subplot(1,3,2);
bar(results.Sector, results.WithinSectorSelection);
title('Within Sector Selection');
subplot(1,3,3);
bar(results.Sector, results.AllocSelecInteraction);
title('Allocation Selection Interaction');
```



Factor-based Analysis

We can acutally think of the Brinson sector-based attribution analysis as a special case of factor-based analysis. This is done by considering the different sectors as different factors driving the active portfolio returns. Our assumption is that the factor loadings X_{ik} are known at the beginning; here in our case X_{ik} is 1, if stock i belongs to sector k. So, we need to run regression to get the factor returns.

To replicate the results from the Brinson model, we note that the factor returns are different between the portfolio and benchmark, due to the difference in the underlying strategies.

```
% formulate active weight for each asset from the given dataset.
assetTable = outerjoin(portData, benchmarkData, 'MergeKeys',true,'type', 'right', ...
    'LeftKey', 'AssetTicker', 'RightKey', 'AssetTicker', ...
    'LeftVariables', {'AssetTicker', 'AssetWgt'}, 'RightVariables', {'AssetWgt', 'AssetRetn',
assetTable = fillmissing(assetTable, 'constant', 0, 'DataVariables',{'AssetWgt_portData'});
assetTable.ActiveWgt = assetTable.AssetWgt_portData - assetTable.AssetWgt_benchmarkData;
sectors = unique(assetTable.Sector);
assetTable(1:7,:)
```

```
ans = 7×6 table
```

	AssetTicker	AssetWgt_po...	AssetWgt_be...	AssetRetn	Sector	ActiveWgt
1	A	0.0100	0.0022	-0.0031	Health Care	0.0078
2	ALGN	0.0100	0.0022	0.0911	Health Care	0.0078
3	AON	0.0100	0.0022	-0.0667	Financials	0.0078
4	APC	0.0100	0.0022	-0.4817	Energy	0.0078
5	AXP	0.0100	0.0022	-0.1115	Financials	0.0078
6	BK	0.0100	0.0022	-0.0476	Financials	0.0078
7	BSX	0.0100	0.0022	0.0313	Health Care	0.0078

Sector Allocation effect: we can use factor-based analysis to compute the allocation effect in the Brinson model. In this case, we would like to use weighted regression to find benchmark factor return and active exposure for each factor.

1) Formulate asset exposure to factors, should usually be predefined. We see that the entry in Xik is 1 if asset i belongs to sector k.

```
Xik_B = (benchmarkData.Sector==sectors');
Xik_B(1:5,:);
```

```
    0    0    0    0    0    1    0    0    0    0    0
    0    1    0    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    1    0    0    0
    0    0    0    0    0    1    0    0    0    0    0
    0    0    0    0    0    1    0    0    0    0    0
```

2) Run weighted regression to get the benchmark factor return fk: r_i = Xik * f_k.

```
warning('off');
mdl = fitlm(Xik_B, benchmarkData.AssetRetn, 'Intercept', false, 'Weights', benchmarkData.AssetW
```

```
mdl = 
Linear regression model:
    y ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 + x10 + x11

Estimated Coefficients:
           Estimate        SE         tStat        pValue  
           _________    ________    ________    __________

    x1     -0.011249    0.034321    -0.32774       0.74326
    x2     -0.022786    0.020904       -1.09       0.27629
    x3      0.092173    0.031106      2.9632     0.0032068
    x4      -0.30578    0.030565     -10.004    2.0962e-21
    x5     -0.030001    0.021613     -1.3881       0.16579
    x6     0.0021366     0.02125     0.10055       0.91995
    x7     -0.053316    0.020738      -2.571      0.010462
    x8     -0.023853    0.021996     -1.0844       0.27875
    x9       -0.1505    0.034321     -4.3849    1.4469e-05
    x10     0.034787    0.029563      1.1767       0.23993
    x11    -0.041217    0.031677     -1.3011       0.19388

Number of observations: 460, Error degrees of freedom: 449
Root Mean Squared Error: 0.00767
```

The factor returns are the coefficents from the trained model. We can see that they exactly match the benchmark sector returns obtained from Brinson model.

```
fk_B = mdl.Coefficients.Estimate(1:end);
[fk_B*100, results.BenchRetn(1:end-1)]
```

```
ans = 11×2
   -1.1249   -1.1249
   -2.2786   -2.2786
    9.2173    9.2173
  -30.5781  -30.5781
   -3.0001   -3.0001
    0.2137    0.2137
   -5.3316   -5.3316
   -2.3853   -2.3853
  -15.0496  -15.0496
    3.4787    3.4787
       ⋮
```

3) Compute the factor contribution for allocation effects:

$$R_{\text{Allocation}} = \sum_k X_k^A f_k$$

```
X_act_exposure = assetTable.ActiveWgt'*(assetTable.Sector==sectors');
R_allc = X_act_exposure*fk_B;
```

We can see that R_allc matches the total 'pure sector allocation effect' from Brinson model, by comparing the values below:

```
R_allc*100
```

```
ans = 4.1366
```

```
results.PureSectorAllocation(end)
```

```
ans = 4.1366
```

Within-Sector Selection effect: We can use factor-based regression analysis to replicate the within-sector selection effects in Brinson model. In this case, both portfolio factor returns and benchmark factor returns are needed.

```
Xik_P = (portData.Sector==sectors');
mdl_P = fitlm(Xik_P, portData.AssetRetn, 'Intercept', false, 'Weights', portData.AssetWgt);
fk_P = mdl_P.Coefficients.Estimate;
[fk_P*100 results.PortRetn(1:end-1)]
```

```
ans = 11×2
    1.9253    1.9253
   -4.2695   -4.2695
   15.0331   15.0331
  -42.9559  -42.9559
   -1.9611   -1.9611
```

```
     1.8558    1.8558
    -5.9437   -5.9437
     1.8943    1.8943
   -12.8452  -12.8452
    -4.9497   -4.9497
       ⋮
```

We have already found the benchmark factor returns fk_B. It is shown that R_wss matches the within-sector selection effect in the Brinsion model.

$$R_{\text{Selection}} = \sum_k X_k^B \left(f_k^P - f_k^B \right)$$

```
R_wss = (benchmarkData.AssetWgt'*Xik_B)*(fk_P - fk_B);
R_wss*100
```

```
ans = 0.3191
```

```
results.WithinSectorSelection(end)
```

```
ans = 0.3191
```

In summary, We have shown how to conduct Brinson and factor-based analysis for performance attribution. We also showed that Brinson model is a special case of factor-based performance analysis, using sectors as the factors. To perform factor-based attribution analysis, we need to have predefined factor loading Xik and asset return ri, so that we can extract the factor return fk. Together with the portfolio and benchmark specifications, we can then find how the portfolio returns are actively driven by the underlying factors.

Supporting function:

```
function resultsPort = pa_brinson(portData, benchmarkData)
overallBenchRetn = benchmarkData.AssetRetn'*benchmarkData.AssetWgt*100; % a scalar

% aggregate Benchmark asset data up to sector level (group)
[G, resultsBench] = findgroups(benchmarkData(:,'Sector'));
resultsBench.SectorRetn = splitapply(@(x, y) x'*y*100/sum(y), benchmarkData.AssetRetn, benchmar
resultsBench.SectorWgt = splitapply(@(y) sum(y)*100, benchmarkData.AssetWgt, G);

% aggregate Portfolio asset data up to sector level (group)
[G, resultsPort] = findgroups(portData(:,'Sector'));
resultsPort.SectorRetn = splitapply(@(x, y) x'*y*100/sum(y), portData.AssetRetn, portData.Asset
resultsPort.SectorWgt = splitapply(@(y) sum(y)*100, portData.AssetWgt, G);

% fill in 0 for categories not invested in Portfolio
resultsPort = outerjoin(resultsPort, resultsBench, 'key', 'Sector','MergeKeys',true, ...
    'RightVariables', {}, 'LeftVariables', {'Sector', 'SectorRetn', 'SectorWgt'});
resultsPort.SectorRetn(isnan(resultsPort.SectorRetn)) = 0.0;
resultsPort.SectorWgt(isnan(resultsPort.SectorWgt)) = 0.0;

% compute the three terms in value-added return
resultsPort.PureSectorAllocation = (resultsPort.SectorWgt - resultsBench.SectorWgt).*...
```

```
    (resultsBench.SectorRetn - overallBenchRetn)/100;
resultsPort.WithinSectorSelection = (resultsBench.SectorWgt).*...
    (resultsPort.SectorRetn - resultsBench.SectorRetn)/100;
resultsPort.AllocSelecInteraction = (resultsPort.SectorWgt - resultsBench.SectorWgt).*...
    (resultsPort.SectorRetn - resultsBench.SectorRetn)/100;

resultsPort.Properties.VariableNames{'SectorRetn'} = 'PortRetn';
resultsPort.Properties.VariableNames{'SectorWgt'} = 'PortWgt';
resultsBench.Properties.VariableNames{'SectorRetn'} = 'BenchRetn';
resultsBench.Properties.VariableNames{'SectorWgt'} = 'BenchWgt';

resultsPort = [resultsPort(:, 1:2), resultsBench(:, 2), resultsPort(:, 3), resultsBench(:, 3),
    resultsPort(:, end-2: end)];

% adding the conclusion row: total portfolio
resultsPort.TotalValueAdded = resultsPort.PureSectorAllocation + resultsPort.WithinSectorSelect
    resultsPort.AllocSelecInteraction;
totalPortRetn = resultsPort.PortRetn'*resultsPort.PortWgt/100;
totalBenchRetn = resultsPort.BenchRetn'*resultsPort.BenchWgt/100;
total = array2table([totalPortRetn, totalBenchRetn, sum(resultsPort{:,4:end})], ...
    'VariableNames', resultsPort.Properties.VariableNames(2:end));
totalPort = [table({'Summary'}, 'VariableNames', resultsPort.Properties.VariableNames(1)), tota
resultsPort = [resultsPort; totalPort];

end
```

References:

- https://cran.r-project.org/web/packages/pa/vignettes/pa.pdf
- http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.170.7676&rep=rep1&type=pdf