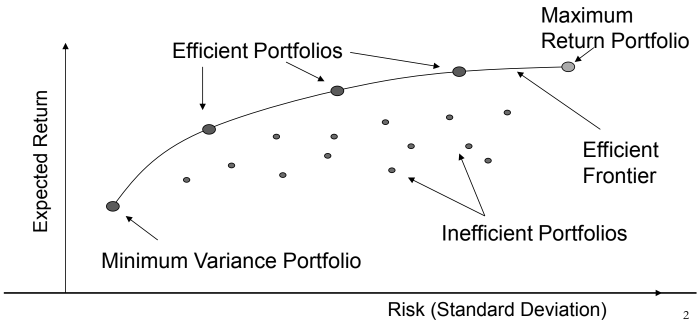
Stat 107: Introduction to Business and Financial Statistics
Class 10: Portfolios, Part III

An Efficient Frontier



Review: Using Excel to find Eff fr

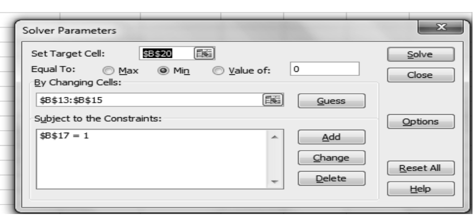
- We will use file mike_3_try.xls to play with.
- We will work with monthly returns of AAPL, RIMM and JNJ.
- We will allow sort sales (negative weights), though this can be changed via the SOLVER constraints.

The Spreadsheet

	A	B	C	D	E
1		Univariate Statistics			
2		aapl	rmm	jnj	
3	Average	0.03469	-0.01273	0.00071	
4	Standard Deviation	0.10605	0.19759	0.04465	
5	Variance	0.01125	0.03904	0.00199	
6					
7	Covraiance matrix		aapl	rimm	jnj
8		aapl	0.01125	0.01096	0.00161
9		rimm	0.01096	0.03904	0.00204
10		jnj	0.00161	0.00204	0.00199
11					
12					
13	w1	0.3000			
14	w2	0.4000			
15	w3	0.3000			
16					
17	constraint	1			
18					
19	port mean	0.005528			
20	port variance	0.0108478			
21	port sd	0.104152772			

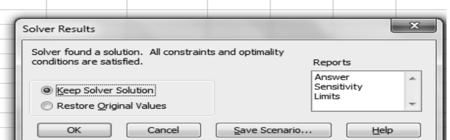
Find the Global MinimumVariance

12		
13	w1	0.3000
14	w2	0.4000
15	w3	0.3000
16		
17	constraint	1
18		
19	port mean	0.0055
20	port variance	0.0108
21	port sd	0.1042
22		
23		
24		



The Solver Output

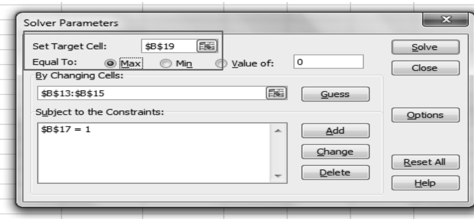
12		
13	w1	0.0511
14	w2	-0.0142
15	w3	0.9631
16		
17	constraint	1
18		
19	port mean	0.0026
20	port variance	0.0020
21	port sd	0.0444
22		
23		
24		



The minimum variance portfolio has mean 0.0026 and standard deviation 0.0444

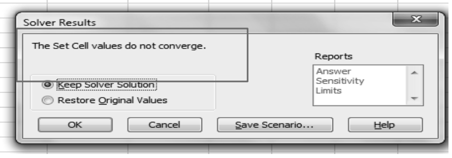
Find the Maximum Return Portfolio

12		
13	w1	0.0511
14	w2	-0.0142
15	w3	0.9631
16		
17	constraint	1
18		
19	port mean	0.0026
20	port variance	0.0020
21	port sd	0.0444
22		
23		
24		
25		



Find the Maximum Return Portfolio

12		
13	w1	############
14	w2	############
15	w3	############
16		
17	constraint	0.999999985
18		
19	port mean	############
20	port variance	############
21	port sd	############
22		
23		



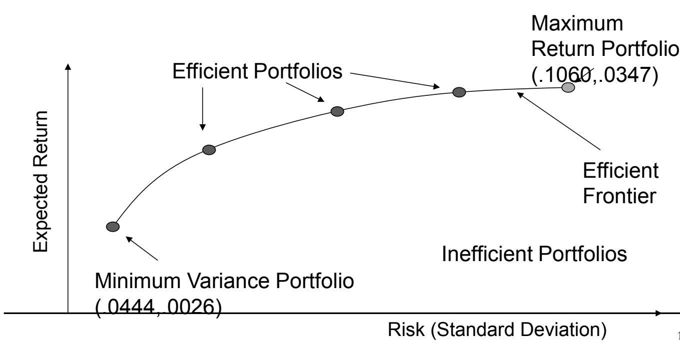
Why no maximum return portfolio?

What to use for Maximum Return?

- It is customary and an easy solution to use (when the weights can be negative) the maximum of the individual security returns as the maximum return portfolio.

	Univariate Statistics			
1				
2		aapl	rmm	jnj
3	Average	0.03469	-0.01273	0.00071
4	Standard Deviation	0.10605	0.19759	0.04465
5	Variance	0.01125	0.03904	0.00199

What we know so far



How do we find other points?

- We know the returns have to range between 0.0026 and 0.0347.
- So we make a table of returns in this range, and use solver to find the portfolio standard deviation that has a specified return.

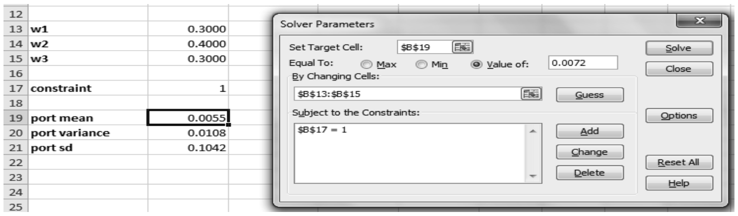
That is, we want to fill in this table

risk	return	
0.0444	0.0026	
	0.0072	=0.0026+(1*(0.0347-0.0026)/7)
	0.0118	
	0.0164	
	0.0209	
	0.0255	=0.0026+(6*(0.0347-0.0026)/7)
	0.0301	
0.1060	0.0347	

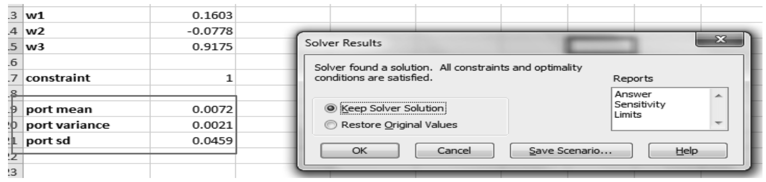
Arbitrary grid of 7 values

Finding the table values

- Need to use Solver like this



The Solution



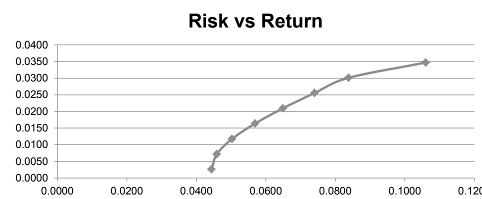
So the point (0.0459,0.0072) is on the efficient frontier.

Do this for all 7 grid points

- We obtain the following filled in table:

risk	return
0.0444	0.0026
0.0459	0.0072
0.0503	0.0118
0.0569	0.0164
0.0649	0.0209
0.0740	0.0255
0.0838	0.0301
0.1060	0.0347

Can then plot this in Excel



Although this method of computing the efficient frontier (hopefully!) makes it clear what is going on, it is a complete pain to do!

Some R functions

- portfolio.txt code on course web site
- Will compute
  - Efficient portfolio for a target return
  - The efficient frontier
  - Global Minimum Portfolio
  - Tangency Portfolio (discussed today)

Example: Minimum Portfolio

- R Code

```
gmin.port <- globalMin.portfolio(er, covmat)
attributes(gmin.port)
print(gmin.port)
plot(gmin.port)
```

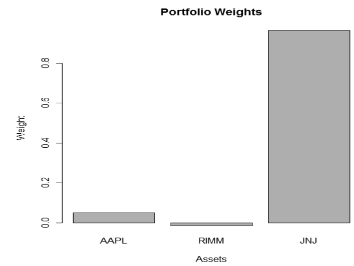
R output for GMV

```
> er
    AAPL     RIMM      JNJ
 0.03469 -0.01273  0.00071
> covmat
        AAPL    RIMM     JNJ
AAPL 0.01125 0.01096 0.00161
RIMM 0.01096 0.03904 0.00204
JNJ  0.00161 0.00204 0.00199
> gmin.port=globalMin.portfolio(er,covmat)
> print(gmin.port)
Call:
globalMin.portfolio(er = er, cov.mat = covmat)

Portfolio expected return:     0.002638269
Portfolio standard deviation:  0.04438313
Portfolio weights:
   AAPL    RIMM     JNJ
 0.0511 -0.0142  0.9631
```


R output for GMV

plot(gmin.port)

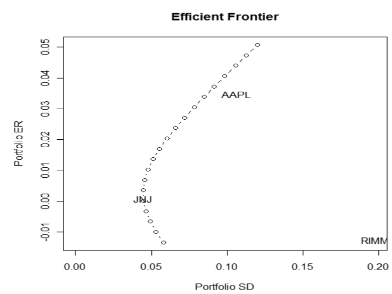


The efficient frontier in R

- The R code

```
> ef=efficient.frontier(er,covmat)
> plot(ef,plot.assets=T)
```


The Output



Can do this with many stocks

- Loaded in the first 20 stocks of the Nasdaq 100

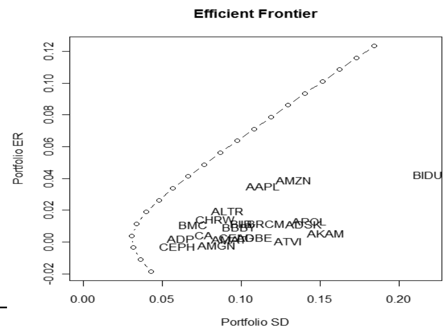
```
 [1] "ATVI" "ADBE" "AKAM" "ALTR" "AMZN" "AMGN" "APOL"
"AAPL" "AMAT" "ADSK"
[11] "ADP"  "BIDU" "BBBY" "BIIB" "BMC"  "BRCM" "CHRW"
"CA"   "CELG" "CEPH"
```


The GMV

- Easy for R to compute

```
Portfolio expected return:     0.001796427
Portfolio standard deviation:  0.03050987
Portfolio weights:
   ATVI    ADBE    AKAM    ALTR    AMZN    AMGN    APOL    AAPL    AMAT    ADSK
 0.0238  0.0485  0.0784  0.1209  0.0063  0.0647  0.0593 -0.0272  0.1280 -0.1455
    ADP    BIDU    BBBY    BIIB     BMC    BRCM    CHRW      CA    CELG    CEPH
 0.3335 -0.0498 -0.0788 -0.0585  0.2273 -0.0514  0.1792 -0.2517  0.1484  0.2446
```

The Efficient Fronteir

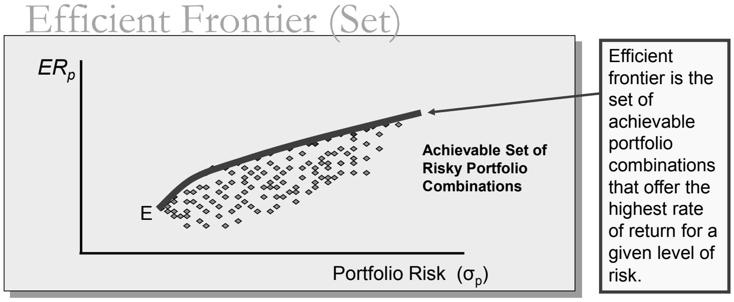


What now?

- The efficient frontier displays all the feasible portfolios possible, given any number of assets.
- However, one can do even better by introducing a risk-free asset into the mix.
- We will now describe the new, or super, efficient frontier.

Recall: Achievable Portfolio Combinations

Efficient Frontier (Set)



A Risk-Free Asset

- Now consider a risk-free asset
- Such an asset is generally consider a T-bill or Treasury Bill, or money in the bank.
- Let r_f denote the return of the risk-free asset
- Then $Var(r_f) = 0$ (its risk-free!).

Risk-free Investing

- When we introduce the presence of a risk-free investment, a whole new set of portfolio combinations becomes possible.
- We can estimate the return on a portfolio made up of RF asset (r_f) and a risky asset A letting the weight (*w*) invested in the risky asset and the weight invested in RF as (1-*w*)
- Our portfolio return: $R_p = (1-w)(r_f) + wR_A$

The Risk-Free Asset

- Note that by definition
  - $E[r_f] = r_f$
  - $Var[r_f] = 0$
  - $Cov(A, r_f) = 0$ for any other portfolio A

The New Efficient Frontier

Risk-Free Investing

- Expected return and risk on a two asset portfolio made up of risky asset *A* and *RF*:

$$E[R_p] = (1\text{-}w)(r_f) + wE[R_A]$$

$$\sigma_P^2 = w^2 \ \sigma_A^2$$

$$\sigma_P = w\sigma_A$$

Equation of a line

- From

$$E[R_p] = (1\text{-}w)(r_f) + wE[R_A]$$

$$\sigma_P = w\sigma_A$$

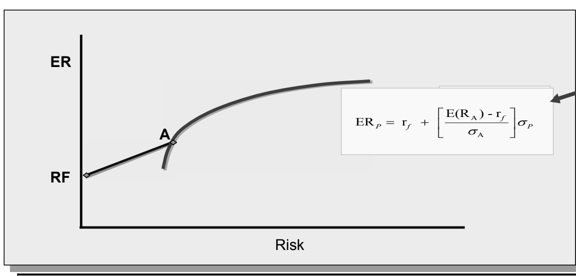
- We obtain

$$E[R_p] = r_f + w(E[R_A]\text{-}r_f)$$

$$= r_f + \sigma_p \left(\frac{E[R_A]\text{-}r_f}{\sigma_A} \right)$$

The New Efficient Frontier

Attainable Portfolios Using *RF* and *A*

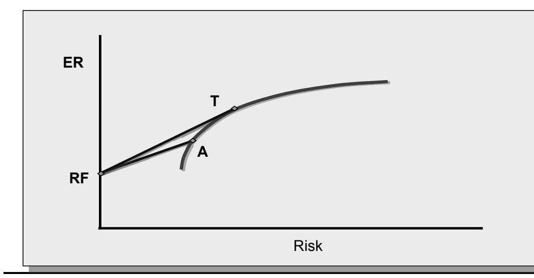


$$ER_P = r_f + \left[\frac{E(R_A) - r_f}{\sigma_A} \right] \sigma_P$$

This means you can achieve any portfolio combination along the blue coloured line simply by changing the relative weight of *RF* and *A* in the two asset portfolio.

The New Efficient Frontier

Attainable Portfolios using the *RF* and *A*, and *RF* and *T*



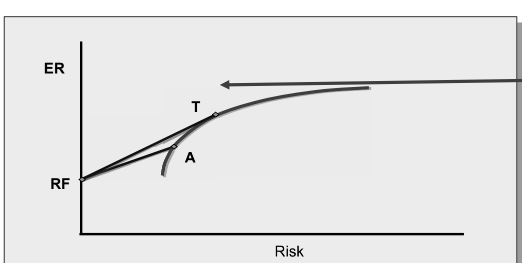
Which risky portfolio would a rational risk-averse investor choose in the presence of a *RF* investment?

Portfolio *A*?

Tangent Portfolio *T*?

The New Efficient Frontier

Efficient Portfolios using the Tangent Portfolio *T*

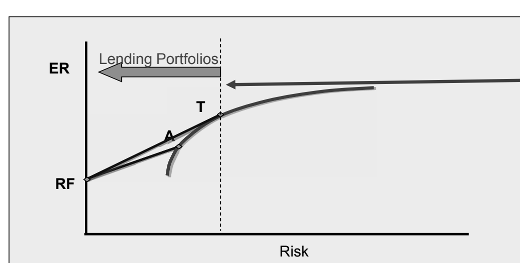


Clearly *RF* with *T* (the tangent portfolio) offers a series of portfolio combinations that dominate those produced by *RF* and *A*.

Further, they dominate all but one portfolio on the efficient frontier!

The New Efficient Frontier

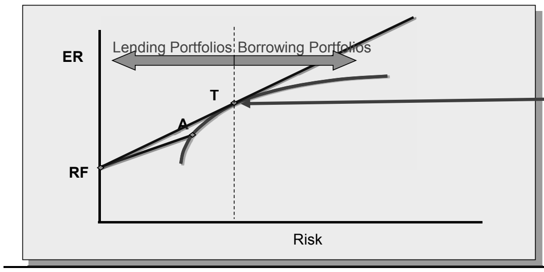
Lending Portfolios



Portfolios between *RF* and *T* are 'lending' portfolios, because they are achieved by investing in the Tangent Portfolio and lending funds to the government (purchasing a T-bill, the *RF*).

The New Efficient Frontier

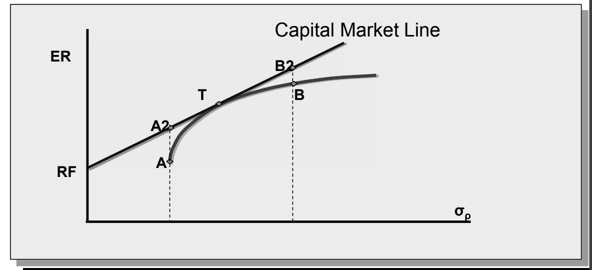
Borrowing Portfolios



The line can be extended to risk levels beyond 'T' by borrowing at RF and investing it in T. This is a levered investment that increases both risk and expected return of the portfolio.

The New Efficient Frontier

The New (Super) Efficient Frontier



This is now called the new (or super) efficient frontier of risky portfolios.

Investors can achieve any one of these portfolio combinations by borrowing or investing in RF in combination with the market portfolio.

The Market Portfolio

- Portfolio T, called the tangent portfolio, is also called the Market Portfolio
- The Capital Market Line is tangent to the efficient frontier, so every portfolio on the efficient frontier is below this line.
- ALL the best investment portfolios are on this line.

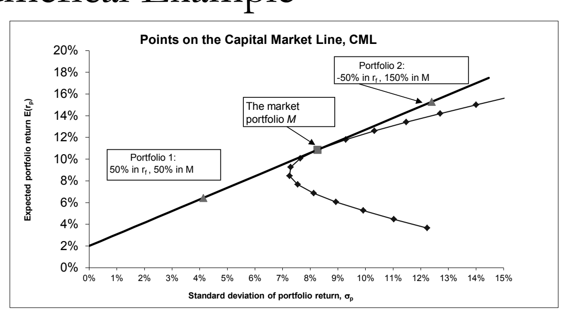
The Capital Market Line

- Is the set of optimal portfolio investments
- Each point on the line is
  - A combination of some percentage invested in the risk free asset
  - Another percentage invested in the market portfolio M.

About the CML

- All the portfolios on the CML incorporate this choice: Each CML portfolio is a combination of an investment in the risk-free asset Rf and the market portfolio M.
- Any portfolio on the CML is optimal in the sense that it could possibly be a rational investor's choice of his best investment portfolio.

Numerical Example



Examples of CML Portfolios

$E(r_M) = 10.85\%$, $r_f = 2\%$, and $\sigma_M = 8.26\%$.

Portfolio Proportions and Investment Returns on the Capital Market Line (CML)

Percentage invested in market portfolio M	$E(r_p) = \%\ in\ risk\text{-}free * r_f + \%\ in\ market * E(r_M)$	$\sigma_p = \%\ in\ market * \sigma_M$
0% (invest all your wealth in risk-free asset r_f)	$E(r_p) = 100\% * r_f = 2\%$	$\sigma_p = 0\% * \sigma_M = 0$
50% (invest 50% of your wealth in market portfolio M and 50% in risk-free asset)	$E(r_p) = 50\% * r_f + 50\% * E(r_M)$ $= 50\% * 2\% + 50\% * 10.85\%$ $= 6.43\%$	$\sigma_p = 50\% * \sigma_M$ $= 50\% * 8.26\% = 4.13\%$
100% (invest all your wealth in market portfolio M)	$E(r_p) = 0\% * r_f + 100\% * E(r_M)$ $= 100\% * 10.85\%$ $= 10.85\%$	$\sigma_p = 100\% * \sigma_M$ $= 100\% * 8.26\% = 8.26\%$
125% (borrow 25% of your wealth to increase investment in risky assets M)	$E(r_p) = -25\% * r_f + 125\% * E(r_M)$ $= -25\% * 2\% + 125\% * 10.85\%$ $= -0.5\% + 13.57\% = 13.06\%$	$\sigma_p = 125\% * \sigma_M$ $= 125\% * 8.26\% = 10.33\%$
150% (borrow 50% of your wealth to increase investment in risky assets M)	$E(r_p) = -50\% * r_f + 150\% * E(r_M)$ $= -50\% * 1\% + 150\% * 10.85\%$ $= -1\% + 16.28\% = 15.28\%$	$\sigma_p = 150\% * \sigma_M$ $= 150\% * 8.26\% = 12.39\%$
200% (borrow 100% of your wealth to increase investment in risky assets M)	$E(r_p) = -100\% * r_f + 200\% * E(r_M)$ $= -100\% * 2\% + 200\% * 10.85\%$ $= -2\% + 21.70\% = 19.70\%$	$\sigma_p = 200\% * \sigma_M$ $= 200\% * 8.26\% = 16.52\%$

The Separation Property: task 1

- Introduced by James Tobin, the 1983 Nobel Laureate for Economics.
- It implies that portfolio choice can be separated into two independent tasks.
- **Task 1**: determining the optimal risky portfolio M (the tangent portfolio)
- Given the particular input data, the best risky portfolio is the same for all clients regardless of risk aversion.

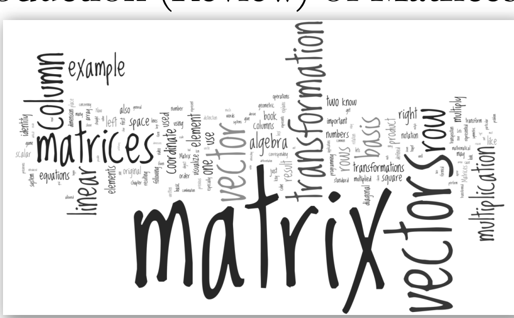
The Separation Property: task 2

- The second task, construction of the complete portfolio from a risk free asset (tbills, say) and portfolio M, however, depends on personal preferences.
- Here the client is the decision maker.
- If the optimal portfolio is the same for all clients, management is more efficient and less costly-the real competition among money managers is their choice of securities.

Practical Implications (summary)

- The analyst or planner should identify what they believe will be the best performing well diversified portfolio, call it P.
- P may include funds, stocks, bonds, international and other alternative investments.
- This portfolio will serve as the starting point for all their clients.
- The planner will then change the asset allocation between the risky portfolio and "near cash" investments according to risk tolerance of client.
- The risky portfolio P may have to be adjusted for individual clients for tax and liquidity concerns if relevant and for the client's opinions.

Introduction (Review) of Matrices



What is it?

- Matrix algebra is a means of making calculations upon arrays of numbers (or data).
- Most data sets are matrix-type
- We've already been using matrices in R to store our data

Why use it?

- Matrix algebra makes *mathematical expression and computation* easier.
- It allows you to get rid of cumbersome notation, concentrate on the concepts involved and understand where your results come from.

Important in Finance

Advanced Portfolio Theory:
Why Understanding The Math Matters

Tom Arnold
Louisiana State University

The goal of this paper is to motivate the use of efficient set mathematics for portfolio analysis [as seen in Roll, 1977] in the classroom. Many treatments stop at the two asset portfolio case (avoiding the use of matrix algebra) and an alarming number of treatments rely on illustration and templates to provide a

50

Definitions - scalar

- a scalar is a number
  - (denoted with regular type: 1 or 22)

Definitions - vector

- Vector: a single row or column of numbers
  - denoted with **bold small letters**
  - row vector $\begin{bmatrix} 1 & 2 & 3 & 4 & 5 \end{bmatrix}$
  - column vector $\begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}$

Definitions - Matrix

- A matrix is an array of numbers

A = $\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}$

- Denoted with a **bold Capital letter**
- All matrices have an order (or dimension): that is, the number of rows × the number of columns. So, **A** is 2 by 3 or (2 × 3).

Definitions

- A square matrix is a matrix that has the same number of rows and columns (n × n)

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

This is actually both a **square** and **symmetric** matrix

Matrix Equality

- Two matrices are equal if and only if
  - they both have the same number of rows and the same number of columns
  - their corresponding elements are equal

Matrix Operations

- Transposition
- Addition and Subtraction
- Multiplication
- Inversion

The Transpose of a Matrix: **A'**

- The transpose of a matrix is a new matrix that is formed by interchanging the rows and columns.
- The transpose of **A** is denoted by **A'** or ($\mathbf{A}^T$)

Example of a transpose

- Thus,

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} \qquad A' = \begin{bmatrix} a_{11} & a_{21} & a_{31} \\ a_{12} & a_{22} & a_{32} \end{bmatrix}$$

- If **A** = **A'**, then **A** is symmetric

Example in R-the `t()` command

```
> mat=matrix(nrow=3,ncol=3,c(1,2,3,4,5,6,7,8,9))
> mat
     [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
> t(mat)
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
>
```

Addition and Subtraction

- Two matrices may be added (or subtracted) iff they are the same order.
- Simply add (or subtract) the corresponding elements. So, **A + B = C** yields

Addition and Subtraction (cont.)

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ c_{31} & c_{32} \end{bmatrix}$$

- Where

$$a_{11} + b_{11} = c_{11}$$
$$a_{12} + b_{12} = c_{12}$$
$$a_{21} + b_{21} = c_{21}$$
$$a_{22} + b_{22} = c_{22}$$
$$a_{31} + b_{31} = c_{31}$$
$$a_{32} + b_{32} = c_{32}$$

Example

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 6 & 8 \\ 10 & 12 \end{bmatrix}$$

```
> mat1=matrix(ncol=2,nrow=2,c(1,2,3,4),byrow=TRUE)
> mat2=matrix(ncol=2,nrow=2,c(5,6,7,8),byrow=TRUE)
> mat1+mat2
     [,1] [,2]
[1,]    6    8
[2,]   10   12
```


Matrix Multiplication

- To multiply a scalar times a matrix, simply multiply each element of the matrix by the scalar quantity

$$k\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} ka_{11} & ka_{12} \\ ka_{21} & ka_{22} \end{bmatrix}$$

Matrix Multiplication (cont.)

- To multiply a matrix times a matrix, we write
  - **AB** (**A** times **B**)
- This is pre-multiplying B by A, or post-multiplying A by B.

Matrix Multiplication (cont.)

- In order to multiply matrices, they must be **CONFORMABLE**
- that is, the number of columns in A must equal the number of rows in B
- So,

A × **B** = **C**

(m × **n**) × (**n** × p) = (m × p)

Matrix Multiplication (cont.)

- (m × n) × (p × n) = cannot be done
- (1 × n) × (n × 1) = a scalar (1x1)

Matrix Multiplication (cont.)

- Thus

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} x \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ c_{31} & c_{32} \end{bmatrix}$$

- where

$$\begin{aligned} c_{11} &= a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31} \\ c_{12} &= a_{11}b_{12} + a_{12}b_{22} + a_{13}b_{32} \\ c_{21} &= a_{21}b_{11} + a_{22}b_{21} + a_{23}b_{31} \\ c_{22} &= a_{21}b_{12} + a_{22}b_{22} + a_{23}b_{32} \\ c_{31} &= a_{31}b_{11} + a_{32}b_{21} + a_{33}b_{31} \\ c_{32} &= a_{31}b_{12} + a_{32}b_{22} + a_{33}b_{32} \end{aligned}$$

Matrix Multiplication- an example %*%

- Thus

$$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix} x \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ c_{31} & c_{32} \end{bmatrix} = \begin{bmatrix} 30 & 66 \\ 36 & 81 \\ 42 & 96 \end{bmatrix}$$

```
> mat1=matrix(nrow=3,ncol=3,c(1,4,7,2,5,8,3,6,9),byrow=TRUE)
> mat2=matrix(nrow=3,ncol=2,c(1,4,2,5,3,6),byrow=TRUE)
> mat1%*%mat2
     [,1] [,2]
[1,]   30   66
[2,]   36   81
[3,]   42   96
```

Properties

- AB does not necessarily equal BA
- (BA may even be an impossible operation)
- For example,

$$\begin{array}{ccccc} \mathbf{A} & \times & \mathbf{B} & = & \mathbf{C} \\ (2 \times 3) & \times & (3 \times 2) & = & (2 \times 2) \\ \mathbf{B} & \times & \mathbf{A} & = & \mathbf{D} \\ (3 \times 2) & \times & (2 \times 3) & = & (3 \times 3) \end{array}$$

Properties

- Matrix multiplication is Associative

  A(BC) = **(AB)C**

- Multiplication and transposition

  (AB)' = B'A'

A popular matrix: **X'X (regression)**

$$\boldsymbol{X} = \begin{bmatrix} 1 & x_{11} \\ 1 & x_{12} \\ \vdots & \vdots \\ 1 & x_{1n} \end{bmatrix}$$

$$X'X = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_{11} & x_{12} & \cdots & x_{1n} \end{bmatrix} \times \begin{bmatrix} 1 & x_{11} \\ 1 & x_{12} \\ \vdots & \vdots \\ 1 & x_{1n} \end{bmatrix} = \begin{bmatrix} n & \sum_{i=1}^{n} x_{1i} \\ \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{1i}^2 \end{bmatrix}$$

Another popular matrix: **e'e**

$$\boldsymbol{e} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

$$\boldsymbol{e'e} = \begin{bmatrix} e_1 & e_2 & \cdots & e_n \end{bmatrix} \times \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix} = \sum_{i=1}^{n} e_i^2$$

Special matrices

- There are a number of special matrices
  - Diagonal
  - Null
  - Identity

Diagonal Matrices

- A diagonal matrix is a square matrix that has values on the diagonal with all off-diagonal entities being zero.

$$\begin{bmatrix} a_{11} & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & a_{33} & 0 \\ 0 & 0 & 0 & a_{44} \end{bmatrix}$$

Identity Matrix

- An identity matrix is a diagonal matrix where the diagonal elements all equal one.

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{A} \times \mathbf{I} = \mathbf{A}$$

Null Matrix

- A square matrix where all elements equal zero.

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The Inverse of a Matrix (A^{-1})

- For an $n \times n$ matrix **A**, there may be a **B** such that **AB** = **I** = **BA**.
- The inverse is analogous to a reciprocal
- A matrix which has an inverse is nonsingular.
- A matrix which does not have an inverse is singular.
- An inverse exists only if $|A| \neq 0$

Properties of inverse matrices

- $(\boldsymbol{AB})^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1}$
- $(\boldsymbol{A}')^{-1} = (\boldsymbol{A}^{-1})'$
- $(\boldsymbol{A}^{-1})^{-1} = \boldsymbol{A}$
-

How to find inverse matrixes?

- Use R! The command is **`solve(mat)`**

$$A = \begin{bmatrix} 2 & 1 & 1 \\ 3 & 2 & 1 \\ 2 & 1 & 2 \end{bmatrix} \quad A^{-1} = ?$$

```
>mat=matrix(nrow=3,ncol=3,c(2,1,1,3,2,1,2,
1,2),byrow=TRUE)
> invmat=solve(mat)
> mat%*%invmat
             [,1] [,2] [,3]
[1,] 1.000000e+00    0    0
[2,] 0.000000e+00    1    0
[3,] 8.881784e-16    0    1
```