Project Part E: Deployment

In [37]:
```
analyst = "Khoa Nguyen" # Replace this with your name
```

In [38]:
```
f = "setup.R"; for (i in 1:10) { if (file.exists(f)) break else f = paste0("../", f) }; source(f)
options(repr.matrix.max.rows=674)
```

1 Introduction

1.1 Decision

Recommend a portfolio of 12 company investments that will maximize 12-month return of an overall $1,000,000 investment.

1.2 Approach

Retrieve a dataset ready for predictive model construction and use it reproduce a selected model.

Retrieve an investment opportunities dataset, comprising fundamentals for some set of public companies over some one-year period. Transform the representation of the investment opportunities to match the representation expected by the model, leveraging previous analysis.

Use the model to make predictions about the investment opportunities and accordingly recommend a portfolio of 12 company investments.

2 Business Model & Business Parameters

The business model is ...

$$\text{profit} = \left(\sum_{i \in \text{portfolio}} (1 + \text{growth}_i) \times \text{allocation}_i \right) - \text{budget}$$

$$\text{profit rate} = \text{profit} \div \text{budget}$$

$$\text{budget} = \sum_{i \in \text{portfolio}} \text{allocation}_i$$

Business parameters include ...

- budget is total investment to allocate across the companies in the portfolio
- portfolio size is number of companies in the portfolio
- allocation is vector of amounts to allocate to specific companies in the portfolio, must sum to budget
- threshold is growth that qualifies as lowest attractive growth

In [39]:
```
# Set the business parameters.

budget = 1000000
portfolio_size = 12
allocation = rep(budget/portfolio_size, portfolio_size)

fmtsx(fmt(budget), fmt(portfolio_size), fmt(allocation))
```

budget	portfolio_size	allocation
1,000,000	12	83,333
		83,333
		83,333
		83,333
		83,333
		83,333
		83,333
		83,333
		83,333
		83,333
		83,333
		83,333

Portfolio to be filled with companies predicted to have the highest growths.

3 Model

3.1 Retrieve Model Training Data

In [40]:

```
# Retrieve model training data.
# How many observations and variables?
# Present the first few observations.

data = read.csv("My Data.csv", header=TRUE, na.strings=c("NA",""), stringsAsFactors=FALSE)
data$big_growth = factor(data$big_growth, levels=c("YES","NO"))

fmtx(size(data))
fmtx(data[1:6,], FFO)
```

size(data)

observations	variables
4,305	9

data (first few observations)

big_growth	growth	prccq	gvkey	tic	conm	PC1	PC2	PC3
NO	0.0507	43.69	1,004	AIR	AAR CORP	1.4098	0.2125	-0.1874
NO	-0.3829	32.11	1,045	AAL	AMERICAN AIRLINES GROUP INC	-2.8093	0.2246	1.4366
YES	0.3158	6.75	1,050	CECE	CECO ENVIRONMENTAL CORP	1.5247	0.4396	-0.1679
NO	-0.2165	8.66	1,062	ASA	ASA GOLD AND PRECIOUS METALS	1.5737	0.6384	0.0123
NO	-0.1185	15.25	1,072	AVX	AVX CORP	1.2813	0.4529	0.0929
NO	0.0002	85.20	1,075	PNW	PINNACLE WEST CAPITAL CORP	0.3698	-0.4861	-0.0128

3.2 Build Model

In [41]:
```
# Construct a linear regression model to predict growth given PC1, PC2 and PC3, based on the model training da
# Present a brief summary of the model parameters.
model = lm(growth ~ PC1 + PC2 + PC3, data)
model
```

```

Call:
lm(formula = growth ~ PC1 + PC2 + PC3, data = data)

Coefficients:
(Intercept)          PC1          PC2          PC3  
   -0.11859      0.00109     -0.00169     -0.00179  
```

4 Investment Opportunities

4.1 Retrieve Investment Data

In [42]:

```
# Retrieve investment data.
# How many observations and variables?
# Present the first few observations.

data.raw = read.csv("Investment Opportunities.csv", header=TRUE, na.strings=c("NA", ""), stringsAsFactors=FALS

fmtx(size(data.raw))
fmtx(data.raw[1:3,], FFO)
```

```
      size(data.raw)

observations  variables
-----------------------
         918        680
```

gvkey	datadate	fyearq	fqtr	fyr	indfmt	consol	popsrc	datafmt	tic	cusip	conm	acctchgq	acctstdq	adrrq	ajexq	ajpq	b
1,004	02/28/2018	2,017	3	5	INDL	C	D	STD	AIR	000361105	AAR CORP	NA	DS	NA	1	1	
1,004	05/31/2018	2,017	4	5	INDL	C	D	STD	AIR	000361105	AAR CORP	NA	DS	NA	1	1	

gvkey	datadate	fyearq	fqtr	fyr	indfmt	consol	popsrc	datafmt	tic	cusip	conm	acctchgq	acctstdq	adrrq	ajexq	ajpq	b
1,004	08/31/2018	2,018	1	5	INDL	C	D	STD	AIR	000361105	AAR CORP	ASU14-09	DS	NA	1	1	

4.2 Partition Investment Data by Calendar Quarter

Partition the dataset by calendar quarter in which information is reported. Filter in observations to include only those with non-missing `prccq` . (Note: it is okay if some observations have prccq $\geq$ 3.) Then remove any observations about companies that reported more than once per quarter. Then change all the variable names (except for the `gvkey` , `tic` , and `conm` variables) by suffixing them with quarter information - e.g., in the Quarter 1 dataset, `prccq` becomes `prccq.q1` , etc.

```
In [43]: # Partition the dataset as described.
         # Present the sizes of the data partitions
         q = quarter(mdy(data.raw$datadate))

         data.current.q1 = data.raw[(q==1) & !is.na(data.raw$prccq),]
         data.current.q2 = data.raw[(q==2) & !is.na(data.raw$prccq),]
         data.current.q3 = data.raw[(q==3) & !is.na(data.raw$prccq),]
         data.current.q4 = data.raw[(q==4) & !is.na(data.raw$prccq),]

         data.current.q1 = data.current.q1[!duplicated(data.current.q1$gvkey),]
         data.current.q2 = data.current.q2[!duplicated(data.current.q2$gvkey),]
         data.current.q3 = data.current.q3[!duplicated(data.current.q3$gvkey),]
         data.current.q4 = data.current.q4[!duplicated(data.current.q4$gvkey),]

         data.current.q1 = rename_with(data.current.q1, ~ifelse(. %in% c("gvkey","tic","conm"), ., paste0(.,".q1")))
         data.current.q2 = rename_with(data.current.q2, ~ifelse(. %in% c("gvkey","tic","conm"), ., paste0(.,".q2")))
         data.current.q3 = rename_with(data.current.q3, ~ifelse(. %in% c("gvkey","tic","conm"), ., paste0(.,".q3")))
         data.current.q4 = rename_with(data.current.q4, ~ifelse(. %in% c("gvkey","tic","conm"), ., paste0(.,".q4")))


         fmtsx(fmt(size(data.current.q1)),
               fmt(size(data.current.q2)),
               fmt(size(data.current.q3)),
               fmt(size(data.current.q4)))
```

```
  size(data.current.q1)        size(data.current.q2)        size(data.current.q3)        size(data.current.q4)

observations   variables     observations   variables     observations   variables     observations   variables
         209         680              221         680              227         680              230         680
```

4.3 Consolidate Investment Data by Company

Consolidate the four quarter datasets into one dataset, with one observation per company that includes variables for all four quarters. Remove any observations with missing `prccq.q4` values.

In [44]:
```
# Consolidate the partitions as described.
# How many observations and variables in the resulting dataset?
data.current = merge(data.current.q1, data.current.q2,by=c("gvkey","tic","conm"), all=TRUE, sort=TRUE)
data.current = merge(data.current, data.current.q3,by=c("gvkey","tic","conm"), all=TRUE, sort=TRUE)
data.current = merge(data.current, data.current.q4,by=c("gvkey","tic","conm"), all=TRUE, sort=TRUE)
data.current = data.current[!is.na(data.current$prccq.q4), ]

fmtx(size(data.current))
```

size(data.current)

observations	variables
230	2,711

4.4 Transform Investment Data

In [45]:
```
# Filter the investment data to include only those variables with at least 95% non-missing
# values in the model training data (from previous analyis).
# How many observations and variables in the resulting dataset?
#
# You can use readRDS("My Filter.rds")

cn = readRDS("My Filter.rds")

data.ps = data.current[, cn]
fmtx(size(data.ps), "investment data after filtration")
```

investment data after filtration

observations	variables
230	200

In [58]:

```
# Impute the investment data using the same imputation values used for the
# model training data (from previous analysis).
# How many observations and variables in the resulting dataset?
#
# You can use readRDS(...)
# You can use put_impute(...)
ml = readRDS("My Imputation.rds")
data.ps = put_impute(data.ps,ml)
fmtx(size(data.ps), "investment data after imputation")
```

investment data after imputation

observations	variables
230	200

In [62]:

```
# Transform the investment data to principal component representation (use centroids and
# weight matrix information from the previous analysis).
# How many observations and variables in the resulting dataset?
# Show the first few observations in the resulting dataset.
#
# You can use the readRDS(...)
# You can use predict(...)
pc = readRDS("My PC.rds")
data.pc = predict(pc, data.ps)
fmtx(size(data.pc))
fmtx(data.pc[1:6,],FFO)
```

size(data.pc)

observations	variables
230	151

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11	PC12	PC13	PC14	PC15	PC16
1.4196	0.0580	-0.2577	-1.6804	-0.2984	-6.983	0.3955	-0.0481	0.3551	-1.4680	0.3802	0.3845	-0.6218	-0.7727	-0.4989	0.0576
1.0563	0.0729	-0.1602	-0.3743	-0.1003	-2.147	0.3451	-0.4175	0.6624	-1.1558	0.4755	0.6401	-1.3768	-1.0224	-1.5236	0.3794
1.6304	0.3224	-0.1279	0.0009	-1.1960	-6.883	-3.0653	-1.9660	0.3024	-0.9527	0.2032	0.0334	-0.1049	-0.6736	0.3222	-0.1936
0.8877	0.1452	-0.6410	-1.8609	-0.3006	-7.567	0.4622	-0.2079	0.5868	-1.3304	0.3044	0.4553	-0.1186	-0.8623	-0.2302	-0.0617
-1.6234	-0.4854	-0.9771	-0.3111	-0.3102	-2.015	-0.7721	-0.6594	-0.5673	-1.1827	-0.0115	-1.1051	0.5062	-0.3665	2.3484	-1.1518
1.4219	-0.1529	-0.3698	-2.2095	-0.3606	-7.891	0.6239	0.3987	0.3382	-0.5269	-0.0219	-0.0532	-0.5036	-0.4108	0.5741	-0.2236

In [84]:
```
# Restore identifier variables and keep only predictor variables stored from previous analysis.
# How many observations and variables?
# Present the few few observations of the resulting dataset.
#
# You can use readRDS(...)
prevars = readRDS("My Predictors.rds")
data.real = cbind(data.current[, 1:6], data.pc)
data.real = data.real[, prevars]
fmtx(size(data.real))
fmtx(data.real[1:6,],FFO)
```

size(data.real)

observations	variables
230	6

data.real (first few observations)

gvkey	tic	conm	PC1	PC2	PC3
1,004	AIR	AAR CORP	1.4196	0.0580	-0.2577
1,410	ABM	ABM INDUSTRIES INC	1.0563	0.0729	-0.1602
1,562	AMSWA	AMERICAN SOFTWARE -CL A	1.6304	0.3224	-0.1279
1,618	AXR	AMREP CORP	0.8877	0.1452	-0.6410
1,632	ADI	ANALOG DEVICES	-1.6234	-0.4854	-0.9771
1,686	APOG	APOGEE ENTERPRISES INC	1.4219	-0.1529	-0.3698

5 Apply Model

5.1 Predict & Recommend Portfolio

In [90]:

```
# Use the model to predict growths of each investment opportunity.
# Recommend a portfolio of allocations to 12 investment opportunities: gvkey, tic, conm, allocation
growth.predicted = predict(model,data.real)
portfolio = data.real
portfolio$growth.predicted = growth.predicted
portfolio = portfolio[order(-portfolio$growth.predicted),]
portfolio = portfolio[1:12, c("gvkey","tic","conm")]
portfolio$allocation = allocation
fmtx(portfolio)
```

portfolio

gvkey	tic	conm	allocation
23,809	AZO	AUTOZONE INC	83,333
180,711	AVGO	BROADCOM INC	83,333
29,692	WEBC	WEBCO INDUSTRIES INC	83,333
3,570	CBRL	CRACKER BARREL OLD CTRY STOR	83,333
178,704	ULTA	ULTA BEAUTY INC	83,333
65,430	PLCE	CHILDRENS PLACE INC	83,333
63,172	FDS	FACTSET RESEARCH SYSTEMS INC	83,333
8,551	PVH	PVH CORP	83,333
1,864	REX	REX AMERICAN RESOURCES CORP	83,333
3,504	COO	COOPER COS INC (THE)	83,333
3,062	CTAS	CINTAS CORP	83,333
7,921	NDSN	NORDSON CORP	83,333

5.2 Store Portfolio Recommendation

In [91]:

```
# Store portfolio recommendation

write.csv(portfolio, paste0(analyst, ".csv"), row.names=FALSE)
```

5.3 Confirm That Format Is Correct

In [92]:

```
portfolio.retrieved = read.csv(paste0(analyst, ".csv"), header=TRUE)
opportunities = unique(read.csv("Investment Opportunities.csv", header=TRUE)$gvkey)

columns = all(colnames(portfolio.retrieved) == c("gvkey", "tic", "conm", "allocation"))
companies = all(portfolio.retrieved$gvkey %in% opportunities)
allocations = round(sum(portfolio.retrieved$allocation)) == budget

check = data.frame(analyst, columns, companies, allocations)
fmtx(check, "Portfolio Recommendation | Format Check")
```

Portfolio Recommendation | Format Check

analyst	columns	companies	allocations
Khoa Nguyen	TRUE	TRUE	TRUE

Copyright (c) Huntsinger Associates, LLC

Document revised May 6, 2023