Name: Laavanya Ganesh
UIN: 654324917

SAS IDS 462 Final Project

Part I

1. From the file annualreports.sas7bdat: Convert the fiscal year variable to date only. Eliminate financial reports after fiscal year 2013.

SOLUTION:

```
libname stocks "/folders/myfolders/";

proc contents data=stocks.annualreports varnum;
run;

proc freq data=stocks.annualreports;
table IndFinancialYearEnd;
run;

data work.annualreports;
set stocks.annualreports;
FiscalYearDate=datepart(IndFinancialYearEnd);
FiscalYear=Year(FiscalYearDate);
run;

proc freq data=work.annualreports;
table FiscalYear;
run;

data work.No2014;
set work.annualreports;
if FiscalYearDate<"01Jan2014"d;
run;

proc freq data=work.No2014;
tables FiscalYear;
run;
```

LOG & FINAL RESULTS:

```
1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
61
62         libname stocks "/folders/myfolders/";
NOTE: Libref STOCKS was successfully assigned as follows:
      Engine:        V9
      Physical Name: /folders/myfolders
63
64         proc contents data=stocks.annualreports varnum;
NOTE: Data file STOCKS.ANNUALREPORTS.DATA is in a format that is native to another
host, or the file encoding does not match the
      session encoding. Cross Environment Data Access will be used, which might
require additional CPU resources and might reduce
      performance.
65         run;

NOTE: PROCEDURE CONTENTS used (Total process time):
      real time           0.42 seconds
      cpu time            0.41 seconds


66
67         proc freq data=stocks.annualreports;
NOTE: Data file STOCKS.ANNUALREPORTS.DATA is in a format that is native to another
host, or the file encoding does not match the
      session encoding. Cross Environment Data Access will be used, which might
require additional CPU resources and might reduce
      performance.
68         table IndFinancialYearEnd;
69         run;

NOTE: There were 51527 observations read from the data set STOCKS.ANNUALREPORTS.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.81 seconds
      cpu time            0.54 seconds


70
71         data work.annualreports;
72         set stocks.annualreports;
NOTE: Data file STOCKS.ANNUALREPORTS.DATA is in a format that is native to another
host, or the file encoding does not match the
      session encoding. Cross Environment Data Access will be used, which might
require additional CPU resources and might reduce
      performance.
73         FiscalYearDate=datepart(IndFinancialYearEnd);
74         FiscalYear=Year(FiscalYearDate);
75         run;

NOTE: There were 51527 observations read from the data set STOCKS.ANNUALREPORTS.
NOTE: The data set WORK.ANNUALREPORTS has 51527 observations and 291 variables.
NOTE: DATA statement used (Total process time):
      real time           1.89 seconds
      cpu time            1.37 seconds


76
```

```
77          proc freq data=work.annualreports;
78          table FiscalYear;
79          run;

NOTE: There were 51527 observations read from the data set WORK.ANNUALREPORTS.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.07 seconds
      cpu time            0.06 seconds


80
81          data work.No2014;
82          set work.annualreports;
83          if FiscalYearDate<"01Jan2014"d;
84          run;

NOTE: There were 51527 observations read from the data set WORK.ANNUALREPORTS.
NOTE: The data set WORK.NO2014 has 50942 observations and 291 variables.
NOTE: DATA statement used (Total process time):
      real time           0.33 seconds
      cpu time            0.30 seconds


85
86          proc freq data=work.No2014;
87          tables FiscalYear;
88          run;

NOTE: There were 50942 observations read from the data set WORK.NO2014.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.07 seconds
      cpu time            0.07 seconds


89
90          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
103
```

The FREQ Procedure

FiscalYear	Frequency	Percent	Cumulative Frequency	Cumulative Percent
1991	1	0.00	1	0.00
1992	3	0.01	4	0.01
1993	1003	1.97	1007	1.98
1994	1269	2.49	2276	4.47
1995	1396	2.74	3672	7.21
1996	1528	3.00	5200	10.21
1997	1658	3.25	6858	13.46
1998	1797	3.53	8655	16.99
1999	2010	3.95	10665	20.94
2000	2114	4.15	12779	25.09
2001	2234	4.39	15013	29.47
2002	2355	4.62	17368	34.09
2003	2481	4.87	19849	38.96
2004	2699	5.30	22548	44.26
2005	2825	5.55	25373	49.81
2006	2870	5.63	28243	55.44
2007	2912	5.72	31155	61.16
2008	3034	5.96	34189	67.11
2009	3147	6.18	37336	73.29
2010	3283	6.44	40619	79.74
2011	3362	6.60	43981	86.34
2012	3456	6.78	47437	93.12
2013	3505	6.88	50942	100.00

2. Find your Sector and Industry in the spreadsheet list on Blackboard.

SOLUTION:

Sector: Consumer Servic
Industry: Other Consumer Servic
Metric for ANOVA: PriceSalesToIndustry
Options: Earliest Expiration Date: 3/1/14
Options: Latest Expiration Date: 11/30/14
Beginning Year for Dividend Yield: 2012
Beginning Year for Splits: 1988
Cut-Off Year for Stock Pick: 2009
Stock Picking Method: Little Book That Beats The Market

3. Using the file AnnualReports, create a file that contains only the records in your Sector and Industry. The remainder of the assignment refers to the companies in your sector and industry.

SOLUTION:

```
proc freq data=No2014;
tables sector*industry/list missing missprint;
run;

data MyCompanies;
set work.No2014;
if Sector="Consumer Servic" and Industry="Other Consumer Servic";
run;

proc freq data=MyCompanies order=freq;
title "Number of Annual Report records by Name";
tables name;
run;
title;
```

LOG & RESULTS:

```
1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
61
62         proc freq data=No2014;
63         tables sector*industry/list missing missprint;
64         run;

NOTE: There were 50942 observations read from the data set WORK.NO2014.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.34 seconds
      cpu time            0.34 seconds
```

```
65
66          data MyCompanies;
67          set work.No2014;
68          if Sector="Consumer Servic" and Industry="Other Consumer Servic";
69          run;

NOTE: There were 50942 observations read from the data set WORK.NO2014.
NOTE: The data set WORK.MYCOMPANIES has 651 observations and 291 variables.
NOTE: DATA statement used (Total process time):
      real time           0.04 seconds
      cpu time            0.04 seconds


70
71          proc freq data=MyCompanies order=freq;
72          title "Number of Annual Report records by Name";
73          tables name;
74          run;

NOTE: There were 651 observations read from the data set WORK.MYCOMPANIES.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.08 seconds
      cpu time            0.08 seconds


75          title;
76
77          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
90
```

Number of Annual Report records by Name

The FREQ Procedure

Name	Frequency	Percent	Cumulative Frequency	Cumulative Percent
DeVry Education Group Inc.	21	3.23	21	3.23
Franklin Covey Company	21	3.23	42	6.45
G&K Services, Inc.	21	3.23	63	9.68
GP Strategies Corporation	21	3.23	84	12.90
Gartner, Inc.	21	3.23	105	16.13
H&R Block, Inc.	21	3.23	126	19.35
Regis Corporation	21	3.23	147	22.58
Service Corporation Internat	21	3.23	168	25.81
Team, Inc.	21	3.23	189	29.03
Tyco International, Ltd. (Sw	21	3.23	210	32.26
Unifirst Corporation	21	3.23	231	35.48
ServiceMaster Global Holding	20	3.07	251	38.56
Apollo Education Group, Inc.	19	2.92	270	41.47
ITT Educational Services, In	19	2.92	289	44.39
Learning Tree International,	19	2.92	308	47.31
Carriage Services, Inc.	18	2.76	326	50.08
Helios and Matheson Analytic	18	2.76	344	52.84
Steiner Leisure Limited	18	2.76	362	55.61
Strayer Education, Inc.	18	2.76	380	58.37
Career Education Corporation	17	2.61	397	60.98
Corinthian Colleges, Inc.	16	2.46	413	63.44
XD Group, Inc.	16	2.46	429	65.90
EnviroStarm, Inc.	15	2.30	444	68.20
Ambassadors Group, Inc.	13	2.00	457	70.20
ManTech International Corpor	13	2.00	470	72.20
Weight Watchers Internationa	13	2.00	483	74.19
Lincoln Educational Services	12	1.84	495	76.04
The Advisory Board Company	12	1.84	507	77.88
Capella Education Company	11	1.69	518	79.57
SmartPros Ltd.	11	1.69	529	81.26
StoneMor Partners L.P.	11	1.69	540	82.95
Universal Technical Institut	11	1.69	551	84.64
American Public Education, I	10	1.54	561	86.18
China Distance Education Hol	9	1.38	570	87.56
Grand Canyon Education, Inc.	9	1.38	579	88.94
K12 Inc	9	1.38	588	90.32
ATA Inc.	8	1.23	596	91.55
Bridgepoint Education, Inc.	8	1.23	604	92.78
New Oriental Education & Tec	8	1.23	612	94.01
Ascent Capital Group, Inc.	6	0.92	618	94.93
Education Management Corpora	6	0.92	624	95.85
FTD Companies, Inc.	6	0.92	630	96.77
National American University	5	0.77	635	97.54
Xueda Education Group	5	0.77	640	98.31
TAL Education Group	4	0.61	644	98.92
Liberty Tax, Inc.	3	0.46	647	99.39
Performant Financial Corpora	2	0.31	649	99.69
Yelp Inc.	2	0.31	651	100.00

4. Remove duplicates of Name within the same fiscal year using proc sort nodupkey.

SOLUTION:

```
proc freq data=MyCompanies;
tables Symbol*Name/list missing missprint;
title "Counts of Symbol by Name—Detect Duplicates";
run;
title;

proc sort nodupkey data=MyCompanies;
by name FiscalYear;
run;

proc freq data=MyCompanies;
tables Symbol*Name/list missing missprint;
title "Counts of Symbol by Name—Detect Duplicates";
run;
title;
```

LOG & RESULTS:

```
1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
61
62         proc freq data=MyCompanies;
63         tables Symbol*Name/list missing missprint;
64         title "Counts of Symbol by Name—Detect Duplicates";
65         run;

NOTE: There were 651 observations read from the data set WORK.MYCOMPANIES.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.09 seconds
      cpu time            0.10 seconds


66         title;
67
68         proc sort nodupkey data=MyCompanies;
69         by name FiscalYear;
70         run;

NOTE: There were 651 observations read from the data set WORK.MYCOMPANIES.
NOTE: 0 observations with duplicate key values were deleted.
NOTE: The data set WORK.MYCOMPANIES has 651 observations and 291 variables.
NOTE: PROCEDURE SORT used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds


71
72         proc freq data=MyCompanies;
73         tables Symbol*Name/list missing missprint;
74         title "Counts of Symbol by Name—Detect Duplicates";
75         run;
```

```
NOTE: There were 651 observations read from the data set WORK.MYCOMPANIES.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.07 seconds
      cpu time            0.08 seconds


76          title;
77
78          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
91
```

Counts of Symbol by Name—Detect Duplicates

The FREQ Procedure

Symbol	Name	Frequency	Percent	Cumulative Frequency	Cumulative Percent
ABCO	The Advisory Board Company	12	1.84	12	1.84
APEI	American Public Education, I	10	1.54	22	3.38
APOL	Apollo Education Group, Inc.	19	2.92	41	6.30
ASCMA	Ascent Capital Group, Inc.	6	0.92	47	7.22
ATAI	ATA Inc.	8	1.23	55	8.45
BPI	Bridgepoint Education, Inc.	8	1.23	63	9.68
CECO	Career Education Corporation	17	2.61	80	12.29
COCO	Corinthian Colleges, Inc.	16	2.46	96	14.75
CPLA	Capella Education Company	11	1.69	107	16.44
CSV	Carriage Services, Inc.	18	2.76	125	19.20
DL	China Distance Education Hol	9	1.38	134	20.58
DV	DeVry Education Group Inc.	21	3.23	155	23.81
EDMC	Education Management Corpora	6	0.92	161	24.73
EDU	New Oriental Education & Tec	8	1.23	169	25.96
EPAX	Ambassadors Group, Inc.	13	2.00	182	27.96
ESI	ITT Educational Services, In	19	2.92	201	30.88
EVI	EnviroStarm, Inc.	15	2.30	216	33.18
FC	Franklin Covey Company	21	3.23	237	36.41
FTD	FTD Companies, Inc.	6	0.92	243	37.33
GK	G&K Services, Inc.	21	3.23	264	40.55
GPX	GP Strategies Corporation	21	3.23	285	43.78
HMNY	Helios and Matheson Analytic	18	2.76	303	46.54
HRB	H&R Block, Inc.	21	3.23	324	49.77
IT	Gartner, Inc.	21	3.23	345	53.00
LINC	Lincoln Educational Services	12	1.84	357	54.84
LOPE	Grand Canyon Education, Inc.	9	1.38	366	56.22

5. Use proc freq to determine the four companies in your industry with the largest number of rows in the file from step 3.

SOLUTION:

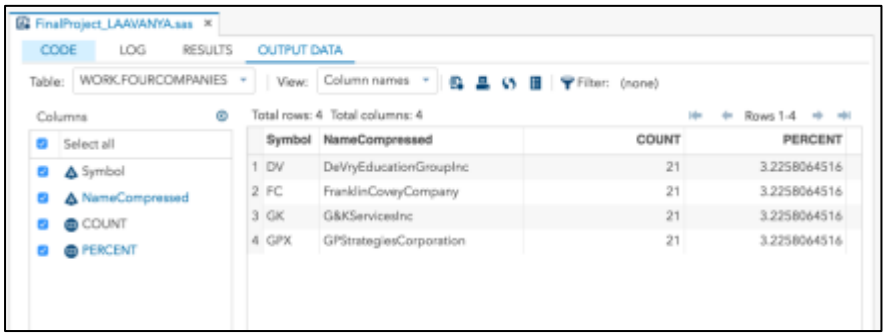
```
data MyCompanies;
set MyCompanies;
NameCompressed=compress(Name," .(),");
run;

proc freq data=MyCompanies order=freq;
tables Symbol*NameCompressed/list out=CompanyCounts;
title "Counts of Symbol by Name—Detect Duplicates";
run;
title;

proc sort data=CompanyCounts;
by descending Count Symbol;
run;
```

```
data FourCompanies;
set CompanyCounts(obs=4);
run;
```

LOG & RESULTS:

```
1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
61
62         data MyCompanies;
63         set MyCcompanies;
64         NameCompressed=compress(Name," .(),");
65         run;

NOTE: There were 651 observations read from the data set WORK.MYCOMPANIES.
NOTE: The data set WORK.MYCOMPANIES has 651 observations and 292 variables.
NOTE: DATA statement used (Total process time):
      real time           0.00 seconds
      cpu time            0.01 seconds


66
67         proc freq data=MyCompanies order=freq;
68         tables Symbol*NameCompressed/list out=CompanyCounts;
69         title "Counts of Symbol by Name—Detect Duplicates";
70         run;

NOTE: There were 651 observations read from the data set WORK.MYCOMPANIES.
NOTE: The data set WORK.COMPANYCOUNTS has 48 observations and 4 variables.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.10 seconds
      cpu time            0.10 seconds


71         title;
72
73         proc sort data=CompanyCounts;
74         by descending Count Symbol;
75         run;

NOTE: There were 48 observations read from the data set WORK.COMPANYCOUNTS.
NOTE: The data set WORK.COMPANYCOUNTS has 48 observations and 4 variables.
NOTE: PROCEDURE SORT used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds


76
77         data FourCompanies;
78         set CompanyCounts(obs=4);
79         run;

NOTE: There were 4 observations read from the data set WORK.COMPANYCOUNTS.
NOTE: The data set WORK.FOURCOMPANIES has 4 observations and 4 variables.
NOTE: DATA statement used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds
```

```
80
81          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
94
```



6. Write programming to create flag variables to indicate whether the record is a member of each of the top four companies in your sector & industry (do not use %CatToBinWithDrop). Name each flag variable as the name of the company. Execute proc freq as demonstrated in the instruction video to show that this programming worked correctly.
    a. Use compress() to remove any spaces or additional nuisance characters (parentheses, periods, commas, etc.) in the company name in order to create a workable SAS variable. Do not use underscores or any other character to differentiate the words in the industry name. So, for example, if one of your companies is "Daily Journal Corp. (S.C.)" then the flag variable would be named DailyJournalCorpSC. All records with the name Daily Journal Corp. (S.C.) would have a 1 for DailyJournalCorpSC and all other records would have a zero.

SOLUTION:

```
data MyCompanies;
set MyCompanies;
NameCompressed=compress(Name," .(),");
run;

proc freq data=MyCompanies order=freq;
tables Symbol*NameCompressed/list out=CompanyCounts;
title "Counts of Symbol by Name—Detect Duplicates";
run;
title;

data WithBinaries;
set MyCompanies;
if NameCompressed="DeVryEducationGroupInc"
then DeVryEducationGroupInc=1;
else DeVryEducationGroupInc=0;
if NameCompressed="FranklinCoveyCompany"
then FranklinCoveyCompany=1;
```

```
      else FranklinCoveyCompany=0;
      if NameCompressed="G&KServicesInc" then GAndKServicesInc=1;
      else GAndKServicesInc=0;
      if NameCompressed="GPStrategiesCorporation"
      then GPStrategiesCorporation=1;
      else GPStrategiesCorporation=0;
      run;

      proc freq data=WithBinaries order=freq;
      tables
Name*DeVryEducationGroupInc*FranklinCoveyCompany*GAndKServicesInc*GPStrategi
esCorporation/list missing missprint;
      run;
```

LOG & RESULTS:

```
 1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
 61
 62
 63         data MyCompanies;
 64         set MyCcompanies;
 65         NameCompressed=compress(Name," .(),");
 66         run;

 NOTE: There were 651 observations read from the data set WORK.MYCOMPANIES.
 NOTE: The data set WORK.MYCOMPANIES has 651 observations and 292 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.01 seconds
       cpu time            0.02 seconds


 67
 68         proc freq data=MyCompanies order=freq;
 69         tables Symbol*NameCompressed/list out=CompanyCounts;
 70         title "Counts of Symbol by Name—Detect Duplicates";
 71         run;

 NOTE: There were 651 observations read from the data set WORK.MYCOMPANIES.
 NOTE: The data set WORK.COMPANYCOUNTS has 48 observations and 4 variables.
 NOTE: PROCEDURE FREQ used (Total process time):
       real time           0.10 seconds
       cpu time            0.09 seconds


 72         title;
 73
 74         /*proc sort data=CompanyCounts;
 75         by descending Count Symbol;
 76         run;
 77
 78         data FourCompanies;
 79         set CompanyCounts(obs=4);
 80         run; */
 81
 82         data WithBinaries;
 83         set MyCompanies;
```

```
 84          if NameCompressed="DeVryEducationGroupInc" then
DeVryEducationGroupInc=1;
 85          else DeVryEducationGroupInc=0;
 86          if NameCompressed="FranklinCoveyCompany" then FranklinCoveyCompany=1;
 87          else FranklinCoveyCompany=0;
 88          if NameCompressed="G&KServicesInc" then GAndKServicesInc=1;
 WARNING: Apparent symbolic reference KSERVICESINC not resolved.
 89          else GAndKServicesInc=0;
 90          if NameCompressed="GPStrategiesCorporation" then
GPStrategiesCorporation=1;
 91          else GPStrategiesCorporation=0;
 92          run;

 NOTE: There were 651 observations read from the data set WORK.MYCOMPANIES.
 NOTE: The data set WORK.WITHBINARIES has 651 observations and 296 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.01 seconds
       cpu time            0.01 seconds


 93
 94          proc freq data=WithBinaries order=freq;
 95          tables
Name*DeVryEducationGroupInc*FranklinCoveyCompany*GAndKServicesInc*GPStrategiesCorpo
ration/list missing missprint;
 96          run;

 NOTE: There were 651 observations read from the data set WORK.WITHBINARIES.
 NOTE: PROCEDURE FREQ used (Total process time):
       real time           0.12 seconds
       cpu time            0.12 seconds


 97
 98          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
 111
```

The FREQ Procedure

Name	DVEGI	FCC	GKSI	GPSC	Frequency	Percent	Cumulative Frequency	Cumulative Percent
DeVry Education Group Inc.	1	0	0	0	21	3.23	21	3.23
Franklin Covey Company	0	1	0	0	21	3.23	42	6.45
G&K Services, Inc.	0	0	1	0	21	3.23	63	9.68
GP Strategies Corporation	0	0	0	1	21	3.23	84	12.90
Gartner, Inc.	0	0	0	0	21	3.23	105	16.13
H&R Block, Inc.	0	0	0	0	21	3.23	126	19.35
Regis Corporation	0	0	0	0	21	3.23	147	22.58
Service Corporation Internat	0	0	0	0	21	3.23	168	25.81
Team, Inc.	0	0	0	0	21	3.23	189	29.03
Tyco International, Ltd. (Be	0	0	0	0	21	3.23	210	32.26
Unifirst Corporation	0	0	0	0	21	3.23	231	35.48
ServiceMaster Global Holding	0	0	0	0	20	3.07	251	38.56
Apollo Education Group, Inc.	0	0	0	0	19	2.92	270	41.47
ITT Educational Services, In	0	0	0	0	19	2.92	289	44.39
Learning Tree International,	0	0	0	0	19	2.92	308	47.31
Carriage Services, Inc.	0	0	0	0	18	2.76	326	50.08
Helios and Matheson Analytic	0	0	0	0	18	2.76	344	52.84
Steiner Leisure Limited	0	0	0	0	18	2.76	362	55.61
Strayer Education, Inc.	0	0	0	0	18	2.76	380	58.37
Career Education Corporation	0	0	0	0	17	2.61	397	60.98
Corinthian Colleges, Inc.	0	0	0	0	16	2.46	413	63.44
XO Group, Inc.	0	0	0	0	16	2.46	429	65.90
EnviroStarm, Inc.	0	0	0	0	15	2.30	444	68.20
Ambassadors Group, Inc.	0	0	0	0	13	2.00	457	70.20
ManTech International Corpor	0	0	0	0	13	2.00	470	72.20
Weight Watchers Internationa	0	0	0	0	13	2.00	483	74.19
Lincoln Educational Services	0	0	0	0	12	1.84	495	76.04

The Advisory Board Company	0	0	0	0	12	1.84	507	77.88
Capella Education Company	0	0	0	0	11	1.69	518	79.57
SmartPros Ltd.	0	0	0	0	11	1.69	529	81.26
Stonelitor Partners L.P.	0	0	0	0	11	1.69	540	82.95
Universal Technical Institut	0	0	0	0	11	1.69	551	84.64
American Public Education, I	0	0	0	0	10	1.54	561	86.18
China Distance Education Hol	0	0	0	0	9	1.38	570	87.56
Grand Canyon Education, Inc.	0	0	0	0	9	1.38	579	88.94
K12 Inc	0	0	0	0	9	1.38	588	90.32
ATA Inc.	0	0	0	0	8	1.23	596	91.55
Bridgepoint Education, Inc.	0	0	0	0	8	1.23	604	92.78
New Oriental Education & Tec	0	0	0	0	8	1.23	612	94.01
Ascent Capital Group, Inc.	0	0	0	0	6	0.92	618	94.93
Education Management Corpora	0	0	0	0	6	0.92	624	95.85
FTD Companies, Inc.	0	0	0	0	6	0.92	630	96.77
National American University	0	0	0	0	5	0.77	635	97.54
Xueda Education Group	0	0	0	0	5	0.77	640	98.31
TAL Education Group	0	0	0	0	4	0.61	644	98.92
Liberty Tax, Inc.	0	0	0	0	3	0.46	647	99.39
Performant Financial Corpora	0	0	0	0	2	0.31	649	99.69
Yelp Inc.	0	0	0	0	2	0.31	651	100.00

7. Use the four companies with the largest number of records in your sector and industry to execute an analysis of variance for the metric listed in the assignment spreadsheet. If there is a tie for the companies with the largest number of records then break the tie based on alphabetical order. Use the ticker symbol for the company as the independent variable. Execute a difference of means test (as was done in the homework) to see which means (if any) differ from the others.

SOLUTION:

```
data ForANOVA;
set WithBinaries;
if DeVryEducationGroupInc=1 or FranklinCoveyCompany=1 or
GAndKServicesInc=1 or GPStrategiesCorporation=1;
      ;
run;

data ConvertMetric;
set ForANOVA;
PriceSToInd=input(PriceSalesToIndustry,8.);
run;

proc means data=ConvertMetric;
class symbol;
var PriceSToInd;
run;

proc anova data=ConvertMetric;
class symbol;
model PriceSToInd=symbol;
means symbol/snk;
run;
quit;
```

LOG & RESULTS:

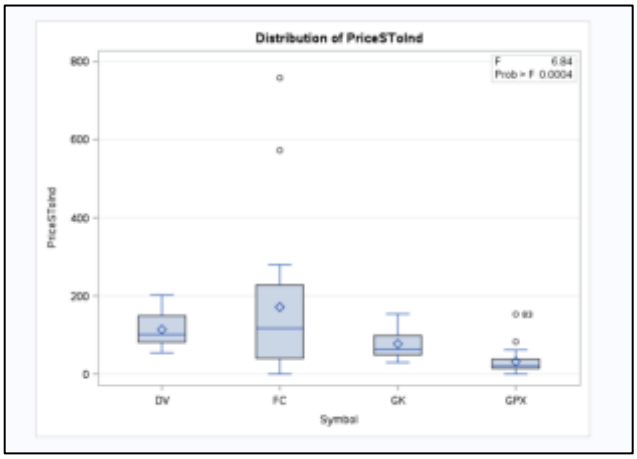
```
1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
61
62         proc anova data=ConvertMetric;
63         class symbol;
64         model PriceSToInd=symbol;
65         means symbol/snk;
66         run;

67         quit;

NOTE: PROCEDURE ANOVA used (Total process time):
      real time           4.31 seconds
      cpu time            0.34 seconds


68
```

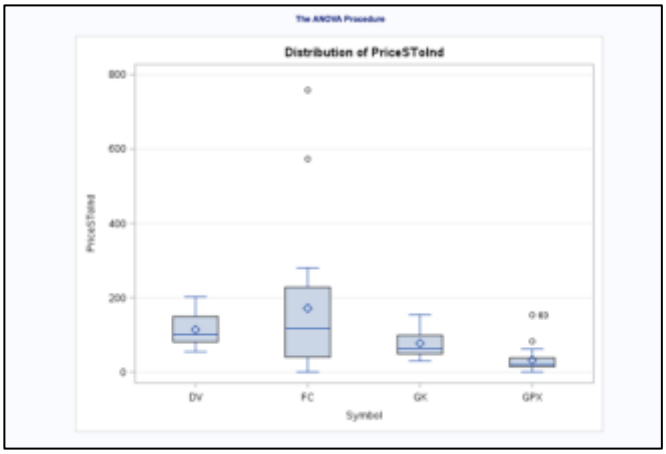
```
69          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
82
```

The MEANS Procedure

Analysis Variable : PriceSToInd

Symbol	N Obs	N	Mean	Std Dev	Minimum	Maximum
DV	21	20	113.7500000	41.7348208	54.2000000	202.2000000
FC	21	20	171.3300000	191.4262620	0	758.3000000
GK	21	20	77.0800000	35.5003721	29.4000000	153.8000000
GPX	21	20	31.5450000	34.6427706	0	153.1000000



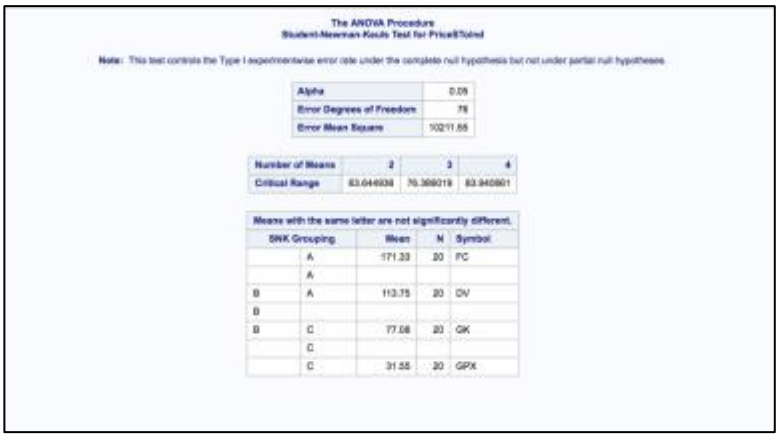


ANOVA INTERPRETATIONS

Ho: $\mu_{(FC)} = \mu_{(DV)} = \mu_{(GK)} = \mu_{(GPX)}$
Ha: At least one of them is different.

The p-value for the ANOVA test is 0.0004.
Conclusion: Reject Ho for Ha (any standard alpha 1%, 5%, or 10%).
Inference in Plain English: We therefore conclude that a statistically significant difference does exist for the 4 different symbols.

Results of the Student-Newman-Keuls Test



The ANOVA Procedure
Student-Newman-Keuls Test for Price$Tolnd

Note: This test controls the Type I experimentwise error rate under the complete null hypothesis but not under partial null hypotheses.

Alpha	0.05
Error Degrees of Freedom	76
Error Mean Square	10211.55

Number of Means	2	3	4
Critical Range	83.044938	70.389019	83.940801

Means with the same letter are not significantly different.

SNK Grouping	Mean	N	Symbol
A	171.33	20	FC
A			
B A	113.75	20	DV
B			
B C	77.08	20	GK
C			
C	31.55	20	GPX

The average for symbol FC is 171.33 and for symbol DV is 113.75. The statistical test does not detect a statistically significant difference between these two averages, as indicated by the 'A' connecting their two averages in the results table above.
The average for symbol DV is 113.55 and for symbol GK is 77.08. The statistical test does not detect a statistically significant difference between these two averages, as indicated by the 'B' connecting their two averages in the results table above.
The average for symbol GK is 77.08 and for symbol GPX is 31.55. The statistical test does not detect a statistically significant difference between these two averages, as indicated by the 'C' connecting their two averages in the results table above.

Symbol FC had a much higher average of 171.33 as compared to the other symbols, while Symbol GPX had a much lower average of 31.55 as compared to the other symbols. These two symbols were statistically different as compared to the others which is indicated by their different SNK grouping letter: 'A' and 'C'.

8. Match the companies in your sector and industry (NOT just your top 4) to the options file using the ticker symbol. For the time period range listed for you in the spreadsheet for the expiration date of the options, how many different options traded for your companies? Use proc freq to produce a count report for the number of options for each of the companies. What was the average strike price of all the Puts, and separately all the Calls for your companies for these options. If a company did not have options then report its number of puts and calls as zero and the average price of each as blank.

SOLUTION:

Method1

```
proc sort nodupkey data=MyCompanies;
by Symbol;
run;

data work.OptionsFile;
set stocks.optionsfile(rename=(underlying=Symbol));
if "01Mar2014"d<=expdate<="30Nov2014"d;
run;

proc sort data=work.OptionsFile;
by Symbol expdate strike;
run;

data MyOptions;
merge MyCompanies(in=OnCompanies keep=symbol)
           work.OptionsFile(in=OnOptions)
           ;
by Symbol;
if OnCompanies and OnOptions;
run;
```

Method2

```
proc sort data=stocks.optionsfile;
by underlying expdate strike;
run;

data MyOptions;
merge MyCompanies(in=OnCompanies keep=symbol)
           stocks.OptionsFile(in=OnOptions rename=(underlying=symbol))
           ;
by Symbol;
if OnCompanies and OnOptions and
        "01Mar2014"d<=expdate<="30Nov2014"d;
```

```
run;

proc freq data=MyOptions;
table Symbol;
run;

proc means data=MyOptions;
class Symbol type;
var strike;
run;

proc summary data=MyOptions nway;
class Symbol type;
var strike;
Output out=OptionStrikes mean=;
run;

proc export data=OptionStrikes
outfile="/folders/myfolders/OptionStrikes.csv "
dbms=csv
replace;
run;
```

LOG & RESULTS:

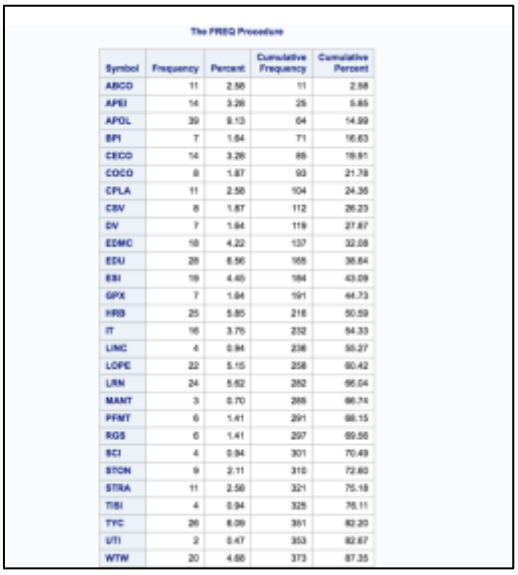
```
1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
61
62
63         proc sort nodupkey data=MyCompanies;
64         by Symbol;
65         run;

NOTE: Input data set is already sorted, no sorting done.
NOTE: PROCEDURE SORT used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds


66
67         data work.OptionsFile;
68         set stocks.optionsfile(rename=(underlying=Symbol));
NOTE: Data file STOCKS.OPTIONSFILE.DATA is in a format that is native to another
host, or the file encoding does not match the
      session encoding. Cross Environment Data Access will be used, which might
require additional CPU resources and might reduce
      performance.
69         if "01Mar2014"d<=expdate<="30Nov2014"d;
70         run;

NOTE: There were 294934 observations read from the data set STOCKS.OPTIONSFILE.
NOTE: The data set WORK.OPTIONSFILE has 67987 observations and 11 variables.
NOTE: DATA statement used (Total process time):
      real time           0.43 seconds
```

```
      cpu time            0.34 seconds


71
72         proc sort data=work.OptionsFile;
73         by Symbol expdate strike;
74         run;

NOTE: There were 67987 observations read from the data set WORK.OPTIONSFILE.
NOTE: The data set WORK.OPTIONSFILE has 67987 observations and 11 variables.
NOTE: PROCEDURE SORT used (Total process time):
      real time           0.04 seconds
      cpu time            0.03 seconds


75
76         data MyOptions;
77         merge MyCompanies(in=OnCompanies keep=symbol)
78            work.OptionsFile(in=OnOptions)
79            ;
80         by Symbol;
81         if OnCompanies and OnOptions;
82         run;

WARNING: Multiple lengths were specified for the BY variable Symbol by input data
sets. This might cause unexpected results.
NOTE: There were 48 observations read from the data set WORK.MYCOMPANIES.
NOTE: There were 67987 observations read from the data set WORK.OPTIONSFILE.
NOTE: The data set WORK.MYOPTIONS has 427 observations and 11 variables.
NOTE: DATA statement used (Total process time):
      real time           0.01 seconds
      cpu time            0.02 seconds


83
84         proc sort data=stocks.optionsfile;
NOTE: Data file STOCKS.OPTIONSFILE.DATA is in a format that is native to another
host, or the file encoding does not match the
      session encoding. Cross Environment Data Access will be used, which might
require additional CPU resources and might reduce
      performance.
85         by underlying expdate strike;
86         run;

NOTE: Input data set is already sorted, no sorting done.
NOTE: PROCEDURE SORT used (Total process time):
      real time           0.00 seconds
      cpu time            0.01 seconds


87
88         data MyOptions;
89         merge MyCompanies(in=OnCompanies keep=symbol)
90            stocks.OptionsFile(in=OnOptions rename=(underlying=symbol))
91            ;
NOTE: Data file STOCKS.OPTIONSFILE.DATA is in a format that is native to another
host, or the file encoding does not match the
      session encoding. Cross Environment Data Access will be used, which might
require additional CPU resources and might reduce
      performance.
```

```
 92          by Symbol;
 93          if OnCcompanies and OnOptions and
 94             "01Mar2014"d<=expdate<="30Nov2014"d;
 95          run;

 WARNING: Multiple lengths were specified for the BY variable Symbol by input data
sets. This might cause unexpected results.
 NOTE: There were 48 observations read from the data set WORK.MYCOMPANIES.
 NOTE: There were 294934 observations read from the data set STOCKS.OPTIONSFILE.
 NOTE: The data set WORK.MYOPTIONS has 427 observations and 11 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.47 seconds
       cpu time            0.37 seconds


 96
 97          proc freq data=MyOptions;
 98          table Symbol;
 99          run;

 NOTE: There were 427 observations read from the data set WORK.MYOPTIONS.
 NOTE: PROCEDURE FREQ used (Total process time):
       real time           0.06 seconds
       cpu time            0.06 seconds


100
101         proc means data=MyOptions;
102         class Symbol type;
103         var strike;
104         run;

 NOTE: There were 427 observations read from the data set WORK.MYOPTIONS.
 NOTE: PROCEDURE MEANS used (Total process time):
       real time           0.12 seconds
       cpu time            0.11 seconds


105
106         proc summary data=MyOptions nway;
107         class Symbol type;
108         var strike;
109         Output out=OptionStrikes mean=;
110         run;

 NOTE: There were 427 observations read from the data set WORK.MYOPTIONS.
 NOTE: The data set WORK.OPTIONSTRIKES has 63 observations and 5 variables.
 NOTE: PROCEDURE SUMMARY used (Total process time):
       real time           0.00 seconds
       cpu time            0.01 seconds


111
112         proc export data=OptionStrikes
113         outfile="/folders/myfolders/OptionStrikes.csv "
114         dbms=csv
115         replace;
116         run;
```

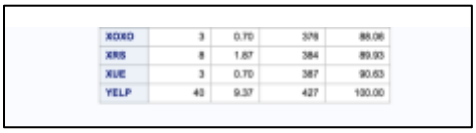
```
 NOTE: Unable to open parameter catalog: SASUSER.PARMS.PARMS.SLIST in update mode.
Temporary parameter values will be saved to
 WORK.PARMS.PARMS.SLIST.
 117
/*************************************************************************
 118         *   PRODUCT:   SAS
 119         *   VERSION:   9.4
 120         *   CREATOR:   External File Interface
 121         *   DATE:      07MAY17
 122         *   DESC:      Generated SAS Datastep Code
 123         *   TEMPLATE SOURCE:  (None Specified.)
 124
*************************************************************************/
 125            data _null_;
 126            %let _EFIERR_ = 0; /* set the ERROR detection macro variable */
 127            %let _EFIREC_ = 0;     /* clear export record count macro variable
*/
 128            file '/folders/myfolders/OptionStrikes.csv' delimiter=',' DSD
DROPOVER lrecl=32767;
 129            if _n_ = 1 then        /* write column names or labels */
 130             do;
 131               put
 132                  "Symbol"
 133               ','
 134                  "type"
 135               ','
 136                  "_TYPE_"
 137               ','
 138                  "_FREQ_"
 139               ','
 140                  "strike"
 141               ;
 142             end;
 143           set  OPTIONSTRIKES   end=EFIEOD;
 144              format Symbol $5. ;
 145              format type $8. ;
 146              format _TYPE_ best12. ;
 147              format _FREQ_ best12. ;
 148              format strike best12. ;
 149             do;
 150               EFIOUT + 1;
 151               put Symbol $ @;
 152               put type $ @;
 153               put _TYPE_ @;
 154               put _FREQ_ @;
 155               put strike ;
 156               ;
 157             end;
 158            if _ERROR_ then call symputx('_EFIERR_',1);  /* set ERROR detection
macro variable */
 159            if EFIEOD then call symputx('_EFIREC_',EFIOUT);
 160            run;

 NOTE: The file '/folders/myfolders/OptionStrikes.csv' is:
       Filename=/folders/myfolders/OptionStrikes.csv,
       Owner Name=root,Group Name=vboxsf,
       Access Permission=-rwxrwx---,
       Last Modified=08May2017:02:18:36

 NOTE: 64 records were written to the file '/folders/myfolders/OptionStrikes.csv'.
```

```
      The minimum record length was 11.
      The maximum record length was 32.
NOTE: There were 63 observations read from the data set WORK.OPTIONSTRIKES.
NOTE: DATA statement used (Total process time):
      real time           0.00 seconds
      cpu time            0.01 seconds


63 records created in /folders/myfolders/OptionStrikes.csv from OPTIONSTRIKES.


NOTE: "/folders/myfolders/OptionStrikes.csv" file was successfully created.
NOTE: PROCEDURE EXPORT used (Total process time):
      real time           0.04 seconds
      cpu time            0.03 seconds


161
162         OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
175
```

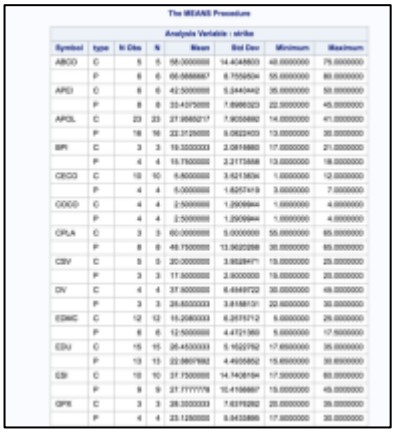


The FREQ Procedure

Symbol	Frequency	Percent	Cumulative Frequency	Cumulative Percent
ABCO	11	2.56	11	2.56
APEI	14	3.26	25	5.85
APOL	39	9.13	64	14.99
BPI	7	1.64	71	16.63
CECO	14	3.28	85	19.91
COCO	8	1.87	93	21.78
CPLA	11	2.58	104	24.36
CSV	8	1.87	112	26.23
DV	7	1.64	119	27.87
EDMC	18	4.22	137	32.08
EDU	28	6.56	165	38.64
ESI	19	4.45	184	43.09
GPX	7	1.64	191	44.73
HRB	25	5.85	216	50.59
IT	16	3.75	232	54.33
LINC	4	0.94	236	55.27
LOPE	22	5.15	258	60.42
LRN	24	5.62	282	66.04
MANT	3	0.70	285	66.74
PFMT	6	1.41	291	68.15
RGS	6	1.41	297	69.56
SCI	4	0.94	301	70.49
STON	9	2.11	310	72.60
STRA	11	2.58	321	75.18
TISI	4	0.94	325	76.11
TYC	26	6.09	351	82.20
UTI	2	0.47	353	82.67
WTW	20	4.68	373	87.35



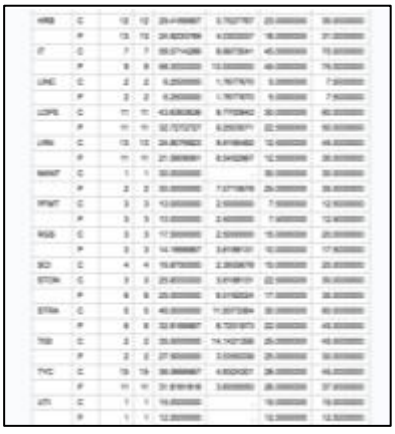
XOXO	3	0.70	378	88.06
XRS	8	1.87	384	89.93
XUE	3	0.70	387	90.63
YELP	40	9.37	427	100.00



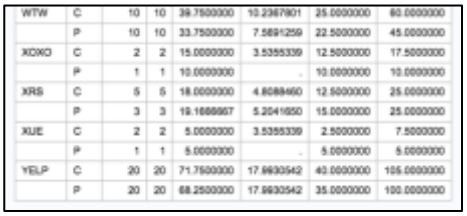
The MEANS Procedure

Analysis Variable : strike

Symbol	type	N Obs	N	Mean	Std Dev	Minimum	Maximum
ABCO	C	5	5	55.0000000	14.4048603	40.0000000	75.0000000
	P	6	6	66.6666667	8.7559504	55.0000000	80.0000000
APEI	C	6	6	42.5000000	5.3440442	35.0000000	50.0000000
	P	8	8	33.4375000	7.8986023	22.5000000	45.0000000
APOL	C	23	23	27.9565217	7.9056892	14.0000000	41.0000000
	P	16	16	33.3125000	9.0832403	13.0000000	50.0000000
BPI	C	3	3	19.3333333	2.0816660	17.0000000	21.0000000
	P	4	4	15.7500000	2.2173558	13.0000000	18.0000000
CECO	C	10	10	5.8000000	3.6213636	1.0000000	12.0000000
	P	4	4	5.0000000	1.8257419	3.0000000	7.0000000
COCO	C	4	4	2.5000000	1.2909944	1.0000000	4.0000000
	P	4	4	2.5000000	1.2909944	1.0000000	4.0000000
CPLA	C	3	3	60.0000000	5.0000000	55.0000000	65.0000000
	P	8	8	48.7500000	13.9620259	30.0000000	65.0000000
CSV	C	5	5	30.0000000	3.9528471	25.0000000	35.0000000
	P	3	3	17.5000000	2.5000000	15.0000000	20.0000000
DV	C	4	4	37.5000000	6.4549722	30.0000000	45.0000000
	P	3	3	26.6666667	3.8188131	22.5000000	30.0000000
EDMC	C	12	12	15.2083333	6.2675172	5.0000000	25.0000000
	P	6	6	12.5000000	4.4721360	5.0000000	17.5000000
EDU	C	15	15	26.4333333	5.1522752	17.5000000	35.0000000
	P	13	13	22.5961538	4.4935852	15.0000000	30.0000000
ESI	C	10	10	37.7500000	14.7408164	17.5000000	60.0000000
	P	9	9	37.7777778	10.4166667	15.0000000	45.0000000
GPX	C	3	3	28.3333333	7.6376262	20.0000000	35.0000000
	P	4	4	23.1250000	3.9433866	17.5000000	30.0000000



HRB	C	12	12	[illegible]	[illegible]	[illegible]	[illegible]
	P	13	13	[illegible]	[illegible]	[illegible]	[illegible]
IT	C	7	7	[illegible]	[illegible]	[illegible]	[illegible]
	P	9	9	[illegible]	[illegible]	[illegible]	[illegible]
LINC	C	2	2	[illegible]	[illegible]	[illegible]	[illegible]
	P	2	2	[illegible]	[illegible]	[illegible]	[illegible]
LOPE	C	11	11	[illegible]	[illegible]	[illegible]	[illegible]
	P	11	11	[illegible]	[illegible]	[illegible]	[illegible]
LRN	C	13	13	[illegible]	[illegible]	[illegible]	[illegible]
	P	11	11	[illegible]	[illegible]	[illegible]	[illegible]
MANT	C	1	1	[illegible]		[illegible]	[illegible]
	P	2	2	[illegible]	[illegible]	[illegible]	[illegible]
PFMT	C	3	3	[illegible]	[illegible]	[illegible]	[illegible]
	P	3	3	[illegible]	[illegible]	[illegible]	[illegible]
RGS	C	3	3	[illegible]	[illegible]	[illegible]	[illegible]
	P	3	3	[illegible]	[illegible]	[illegible]	[illegible]
SCI	C	4	4	[illegible]	[illegible]	[illegible]	[illegible]
STON	C	3	3	[illegible]	[illegible]	[illegible]	[illegible]
	P	6	6	[illegible]	[illegible]	[illegible]	[illegible]
STRA	C	5	5	[illegible]	[illegible]	[illegible]	[illegible]
	P	6	6	[illegible]	[illegible]	[illegible]	[illegible]
TISI	C	2	2	[illegible]	[illegible]	[illegible]	[illegible]
	P	2	2	[illegible]	[illegible]	[illegible]	[illegible]
TYC	C	15	15	[illegible]	[illegible]	[illegible]	[illegible]
	P	11	11	[illegible]	[illegible]	[illegible]	[illegible]
UTI	C	1	1	[illegible]		[illegible]	[illegible]
	P	1	1	[illegible]		[illegible]	[illegible]



WTW	C	10	10	39.7500000	10.2367801	25.0000000	60.0000000
	P	10	10	33.7500000	7.5891259	22.5000000	45.0000000
XOXO	C	2	2	15.0000000	3.5355339	12.5000000	17.5000000
	P	1	1	10.0000000	.	10.0000000	10.0000000
XRS	C	5	5	18.0000000	4.8989460	12.5000000	25.0000000
	P	3	3	19.1666667	5.2041650	15.0000000	25.0000000
XUE	C	2	2	5.0000000	3.5355339	2.5000000	7.5000000
	P	1	1	5.0000000	.	5.0000000	5.0000000
YELP	C	20	20	71.7500000	17.9930542	40.0000000	105.0000000
	P	20	20	68.2500000	17.9930542	35.0000000	100.0000000

Symbol	type	_TYPE_	_FREQ_	strike
ABCO	C	3	5	58
ABCO	P	3	6	66.6666667
APEI	C	3	6	42.5
APEI	P	3	8	33.4375
APOL	C	3	23	27.9565217
APOL	P	3	16	22.3125
BPI	C	3	3	19.3333333
BPI	P	3	4	15.75
CECO	C	3	10	5.8
CECO	P	3	4	5
COCO	C	3	4	2.5
COCO	P	3	4	2.5
CPLA	C	3	3	60
CPLA	P	3	8	48.75
CSV	C	3	5	20
CSV	P	3	3	17.5
DV	C	3	4	37.5
DV	P	3	3	25.8333333
EDMC	C	3	12	15.2083333
EDMC	P	3	6	12.5
EDU	C	3	15	26.4533333
EDU	P	3	13	22.8807692
ESI	C	3	10	37.75
ESI	P	3	9	27.7777778
GPX	C	3	3	28.3333333
GPX	P	3	4	23.125
HRB	C	3	12	29.4166667
HRB	P	3	13	24.9230769
IT	C	3	7	58.5714286
IT	P	3	9	58.3333333
LINC	C	3	2	6.25
LINC	P	3	2	6.25
LOPE	C	3	11	43.6363636
LOPE	P	3	11	32.7272727
LRN	C	3	13	24.8076923
LRN	P	3	11	21.5909091
MANT	C	3	1	30
MANT	P	3	2	30
PFMT	C	3	3	10
PFMT	P	3	3	10
RGS	C	3	3	17.5
RGS	P	3	3	14.1666667
SCI	C	3	4	16.875
STON	C	3	3	25.8333333
STON	P	3	6	25
STRA	C	3	5	46
STRA	P	3	6	32.9166667
TISI	C	3	2	35
TISI	P	3	2	27.5
TYC	C	3	15	36.0666667
TYC	P	3	11	31.8181818
UTI	C	3	1	15
UTI	P	3	1	12.5
WTW	C	3	10	39.75
WTW	P	3	10	33.75
XOXO	C	3	2	15
XOXO	P	3	1	10
XRS	C	3	5	18
XRS	P	3	3	19.1666667
XUE	C	3	2	5
XUE	P	3	1	5
YELP	C	3	20	71.75
YELP	P	3	20	68.25

9. Determine the dividend yield for your companies as of the date listed for you in the assignment spreadsheet. Pretend that you bought shares in each of these companies on the first trading date of the year listed for you in the spreadsheet. Determine how much was paid in dividends per share for the next number of years as listed for you in the assignment spreadsheet. Total up the amount of the dividends and divide by the share price for each company and calculate the dividend yield (total dividends paid divided by share price). If a company did not pay dividends, then their dividend amounts and yield will be zero.

SOLUTION:

```
data work.prices;
set stocks.prices;
year=year(date);
run;
```

```
proc means data=work.prices n nmiss min;
class year;
var date;
run;

proc summary data=work.prices nway;
class year;
var date;
output out=FirstTradingDayPerYear min=;
run;

proc print data=FirstTradingDayPerYear;
run;

data MyFirstTradingDay;
set stocks.prices;
if date="03Jan2012"d;
run;

proc sort data=MyFirstTradingDay;
by tic;
run;

data MyPriceFirstTradingDay;
merge MyCompanies(in=OnCompanies keep=Symbol)
MyFirstTradingDay(in=OnPrices rename=(tic=Symbol));
By Symbol;
If OnCcompanies and OnPrices;
run;

data work.DivFile;
set stocks.DivFile;
where Date ge "01Jan2012"d;
rename tic=Symbol;
run;

data MyDividends;
merge MyPriceFirstTradingDay(in=OnPrice)
           DivFile(in=OnDiv)
            ;
by symbol;
if OnPrice and OnDiv;
run;

proc summary data=MyDividends nway;
class symbol adjclose;
var DivAmount;
```

Obs	year	_TYPE_	_FREQ_	date
1	1990	1	236427	1990-01-02
2	1991	1	265352	1991-01-02
3	1992	1	324379	1992-01-02
4	1993	1	368578	1993-01-04
5	1994	1	413628	1994-01-03
6	1995	1	461929	1995-01-03
7	1996	1	521295	1996-01-02
8	1997	1	567305	1997-01-02
9	1998	1	607120	1998-01-02
10	1999	1	658318	1999-01-04
11	2000	1	709691	2000-01-03
12	2001	1	729227	2001-01-01
13	2002	1	770016	2002-01-02
14	2003	1	807331	2003-01-02
15	2004	1	858335	2004-01-02
16	2005	1	911405	2005-01-03
17	2006	1	963640	2006-01-03
18	2007	1	1030282	2007-01-03
19	2008	1	1094823	2008-01-02
20	2009	1	1113056	2009-01-02
21	2010	1	1160974	2010-01-04
22	2011	1	1223019	2011-01-03
23	2012	1	1274769	2012-01-02
24	2013	1	1368596	2013-01-01
25	2014	1	1170214	2014-01-01

```
Output out=DivSum sum=;
run;

data DivCalc;
format DivYield percent8.1;
set DivSum;
DivYield=DivAMount/adjclose;
run;

proc export data=DivCalc
outfile="/folders/myfolders/DivCalc.csv "
dbms=csv
replace;
run;
```

LOG & RESULTS:

```
1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
61
62
63         data work.prices;
64         set stocks.prices;
NOTE: Data file STOCKS.PRICES.DATA is in a format that is native to another host,
or the file encoding does not match the session
      encoding. Cross Environment Data Access will be used, which might require
additional CPU resources and might reduce
      performance.
65         year=year(date);
66         run;

NOTE: There were 19609709 observations read from the data set STOCKS.PRICES.
NOTE: The data set WORK.PRICES has 19609709 observations and 9 variables.
NOTE: DATA statement used (Total process time):
      real time           26.36 seconds
      cpu time            21.16 seconds


67
68         proc means data=work.prices n nmiss min;
69         class year;
70         var date;
71         run;

NOTE: There were 19609709 observations read from the data set WORK.PRICES.
NOTE: PROCEDURE MEANS used (Total process time):
      real time           3.09 seconds
      cpu time            2.81 seconds


72
73         proc summary data=work.prices nway;
74         class year;
75         var date;
76         output out=FirstTradingDayPerYear min=;
```

```
 77         run;

 NOTE: There were 19609709 observations read from the data set WORK.PRICES.
 NOTE: The data set WORK.FIRSTTRADINGDAYPERYEAR has 25 observations and 4
variables.
 NOTE: PROCEDURE SUMMARY used (Total process time):
       real time           3.26 seconds
       cpu time            3.03 seconds


 78
 79         proc print data=FirstTradingDayPerYear;
 80         run;

 NOTE: There were 25 observations read from the data set
WORK.FIRSTTRADINGDAYPERYEAR.
 NOTE: PROCEDURE PRINT used (Total process time):
       real time           0.05 seconds
       cpu time            0.04 seconds


 81
 82         data MyFirstTradingDay;
 83         set stocks.prices;
 NOTE: Data file STOCKS.PRICES.DATA is in a format that is native to another host,
or the file encoding does not match the session
       encoding. Cross Environment Data Access will be used, which might require
additional CPU resources and might reduce
       performance.
 84         if date="03Jan2012"d;
 85         run;

 NOTE: There were 19609709 observations read from the data set STOCKS.PRICES.
 NOTE: The data set WORK.MYFIRSTTRADINGDAY has 4952 observations and 8 variables.
 NOTE: DATA statement used (Total process time):
       real time           20.87 seconds
       cpu time            16.82 seconds


 86
 87         proc sort data=MyFirstTradingDay;
 88         by tic;
 89         run;

 NOTE: There were 4952 observations read from the data set WORK.MYFIRSTTRADINGDAY.
 NOTE: The data set WORK.MYFIRSTTRADINGDAY has 4952 observations and 8 variables.
 NOTE: PROCEDURE SORT used (Total process time):
       real time           0.00 seconds
       cpu time            0.01 seconds


 90
 91         data MyPriceFirstTradingDay;
 92         merge MyCompanies(in=OnCompanies keep=Symbol)
 93            MyFirstTradingDay(in=OnPrices rename=(tic=Symbol));
 94         By Symbol;
 95         If OnCompanies and OnPrices;
 96         run;

 NOTE: There were 48 observations read from the data set WORK.MYCOMPANIES.
```

```
 NOTE: There were 4952 observations read from the data set WORK.MYFIRSTTRADINGDAY.
 NOTE: The data set WORK.MYPRICEFIRSTTRADINGDAY has 43 observations and 8
variables.
 NOTE: DATA statement used (Total process time):
       real time           0.00 seconds
       cpu time            0.00 seconds


 97
 98          data work.DivFile;
 99          set stocks.DivFile;
 NOTE: Data file STOCKS.DIVFILE.DATA is in a format that is native to another host,
or the file encoding does not match the session
       encoding. Cross Environment Data Access will be used, which might require
additional CPU resources and might reduce
       performance.
 100         where Date ge "01Jan2012"d;
 101         rename tic=Symbol;
 102         run;

 NOTE: There were 41061 observations read from the data set STOCKS.DIVFILE.
       WHERE Date>='01JAN2012'D;
 NOTE: The data set WORK.DIVFILE has 41061 observations and 4 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.37 seconds
       cpu time            0.26 seconds


 103
 104         data MyDividends;
 105         merge MyPriceFirstTradingDay(in=OnPrice)
 106            DivFile(in=OnDiv)
 107               ;
 108         by symbol;
 109         if OnPrice and OnDiv;
 110         run;

 WARNING: Multiple lengths were specified for the BY variable Symbol by input data
sets. This might cause unexpected results.
 NOTE: There were 43 observations read from the data set
WORK.MYPRICEFIRSTTRADINGDAY.
 NOTE: There were 41061 observations read from the data set WORK.DIVFILE.
 NOTE: The data set WORK.MYDIVIDENDS has 193 observations and 10 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.02 seconds
       cpu time            0.01 seconds


 111
 112         proc summary data=MyDividends nway;
 113         class symbol adjclose;
 114         var DivAmount;
 115         Output out=DivSum sum=;
 116         run;

 NOTE: There were 193 observations read from the data set WORK.MYDIVIDENDS.
 NOTE: The data set WORK.DIVSUM has 27 observations and 5 variables.
 NOTE: PROCEDURE SUMMARY used (Total process time):
       real time           0.01 seconds
       cpu time            0.00 seconds
```

```
 117
 118         data DivCalc;
 119         format DivYield percent8.1;
 120         set DivSum;
 121         DivYield=DivAMount/adjclose;
 122         run;

 NOTE: There were 27 observations read from the data set WORK.DIVSUM.
 NOTE: The data set WORK.DIVCALC has 27 observations and 6 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.00 seconds
       cpu time            0.00 seconds


 123
 124         proc export data=DivCalc
 125         outfile="/folders/myfolders/DivCalc.csv "
 126         dbms=csv
 127         replace;
 128         run;

 NOTE: Unable to open parameter catalog: SASUSER.PARMS.PARMS.SLIST in update mode.
Temporary parameter values will be saved to
 WORK.PARMS.PARMS.SLIST.
 129
/***********************************************************************
 130          *   PRODUCT:   SAS
 131          *   VERSION:   9.4
 132          *   CREATOR:   External File Interface
 133          *   DATE:      07MAY17
 134          *   DESC:      Generated SAS Datastep Code
 135          *   TEMPLATE SOURCE:  (None Specified.)
 136
***********************************************************************/
 137             data _null_;
 138             %let _EFIERR_ = 0; /* set the ERROR detection macro variable */
 139             %let _EFIREC_ = 0;     /* clear export record count macro variable
*/
 140             file '/folders/myfolders/DivCalc.csv' delimiter=',' DSD DROPOVER
lrecl=32767;
 141             if _n_ = 1 then        /* write column names or labels */
 142              do;
 143                put
 144                   "DivYield"
 145                ','
 146                   "Symbol"
 147                ','
 148                   "AdjClose"
 149                ','
 150                   "_TYPE_"
 151                ','
 152                   "_FREQ_"
 153                ','
 154                   "DivAmount"
 155                ;
 156              end;
 157            set  DIVCALC   end=EFIEOD;
 158                format DivYield percent8.1 ;
```

```
 159               format Symbol $5. ;
 160               format AdjClose best12. ;
 161               format _TYPE_ best12. ;
 162               format _FREQ_ best12. ;
 163               format DivAmount best12. ;
 164             do;
 165               EFIOUT + 1;
 166               put DivYield @;
 167               put Symbol $ @;
 168               put AdjClose @;
 169               put _TYPE_ @;
 170               put _FREQ_ @;
 171               put DivAmount ;
 172               ;
 173             end;
 174            if _ERROR_ then call symputx('_EFIERR_',1);  /* set ERROR detection
macro variable */
 175            if EFIEOD then call symputx('_EFIREC_',EFIOUT);
 176            run;

 NOTE: The file '/folders/myfolders/DivCalc.csv' is:
       Filename=/folders/myfolders/DivCalc.csv,
       Owner Name=root,Group Name=vboxsf,
       Access Permission=-rwxrwx---,
       Last Modified=08May2017:03:50:48

 NOTE: 28 records were written to the file '/folders/myfolders/DivCalc.csv'.
       The minimum record length was 20.
       The maximum record length was 48.
 NOTE: There were 27 observations read from the data set WORK.DIVCALC.
 NOTE: DATA statement used (Total process time):
       real time           0.01 seconds
       cpu time            0.01 seconds


 27 records created in /folders/myfolders/DivCalc.csv from DIVCALC.


 NOTE: "/folders/myfolders/DivCalc.csv" file
was successfully created.
 NOTE: PROCEDURE EXPORT used (Total process
time):
       real time           0.12 seconds
       cpu time            0.04 seconds


 177
 178
 179        OPTIONS NONOTES NOSTIMER NOSOURCE
NOSYNTAXCHECK;
 192
```

DivYield	Symbol	AdjClose	TYPE_	FREQ_	DivAmount
0.00%	APOL	53.13	3	5	0
9.00%	ATAI	6.49	3	2	0.584
3.90%	CPLA	35.72	3	4	1.4
5.10%	CSV	5.39	3	11	0.275
58.10%	DL	1.86	3	2	1.08
2.20%	DV	37.75	3	5	0.83
2.80%	EDU	22.86	3	2	0.65
8.90%	EPAX	4.03	3	6	0.36
0.00%	ESI	54.55	3	5	0
114.90%	EVI	0.87	3	2	1
70.20%	GK	20.92	3	13	14.69
8.80%	HMNY	1.93	3	2	0.17
14.90%	HRB	14.74	3	11	2.2
10.60%	LINC	6.81	3	11	0.72
7.70%	MANT	30.02	3	11	2.31
6.70%	NAUH	6.74	3	11	0.451
3.00%	RGS	16.13	3	8	0.48
7.90%	SCI	10.17	3	12	0.8
9.50%	SPRO	1.66	3	11	0.157
36.90%	STON	17.72	3	11	6.54
4.50%	STRA	89.3	3	4	4
7.40%	TYC	22.6	3	12	1.67284
0.70%	UNF	57.65	3	11	0.415
9.10%	UTI	12.03	3	11	1.1
2.30%	WTW	52.51	3	7	1.225
5.30%	XRS	9.41	3	1	0.5
23.10%	XUE	2.86	3	2	0.66

10. Determine the minimum and maximum split amounts for your companies and the total number of splits. Match to the splits file by ticker for your companies. For the time period listed for you in the assignment spreadsheet determine the maximum and minimum split amount for your companies. If a company did not have a split in the time period, then report a split amount of blank and a number of splits of zero.

SOLUTION:

```
data work.Splits(drop=date rename=(splitdate=date));
set stocks.splits;
SplitDate = input(date, YYMMDD10.);
format SplitDate YYMMDD10.;
rename tic=Symbol;
run;

data MySplits;
merge MyCompanies(in=OnCompanies keep=Symbol)
             Splits(in=OnSplits)
                  ;
by Symbol;
if OnCompanies and OnSplits
          and date ge "01Jan1998"d;
run;

proc means data=MySplits max min;
class Symbol;
var split;
run;

proc summary data=MySplits nway;
class Symbol;
var split;
Output out=SplitMinMax(drop=_type_) min=SplitMin max=SplitMax;
run;
```

LOG & RESULTS:

```
 1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
 61
 62         data work.Splits(drop=date rename=(splitdate=date));
 63         set stocks.splits;
 NOTE: Data file STOCKS.SPLITS.DATA is in a format that is native to another host,
or the file encoding does not match the session
       encoding. Cross Environment Data Access will be used, which might require
additional CPU resources and might reduce
       performance.
 64         SplitDate = input(date, YYMMDD10.);
 65         format SplitDate YYMMDD10.;
 66         rename tic=Symbol;
 67         run;

 NOTE: There were 6015 observations read from the data set STOCKS.SPLITS.
 NOTE: The data set WORK.SPLITS has 6015 observations and 3 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.01 seconds
       cpu time            0.01 seconds


 68
 69         data MySplits;
 70         merge MyCompanies(in=OnCompanies keep=Symbol)
 71              Splits(in=OnSplits)
 72                      ;
 73         by Symbol;
 74         if OnCompanies and OnSplits
 75             and date ge "01Jan1998"d;
 76         run;

 WARNING: Multiple lengths were specified for the BY variable Symbol by input data
sets. This might cause unexpected results.
 NOTE: There were 48 observations read from the data set WORK.MYCOMPANIES.
 NOTE: There were 6015 observations read from the data set WORK.SPLITS.
 NOTE: The data set WORK.MYSPLITS has 24 observations and 3 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.00 seconds
       cpu time            0.00 seconds


 77
 78         proc means data=MySplits max min;
 79         class Symbol;
 80         var split;
 81         run;

 NOTE: There were 24 observations read from the data set WORK.MYSPLITS.
 NOTE: PROCEDURE MEANS used (Total process time):
       real time           0.05 seconds
       cpu time            0.05 seconds


 82
 83         proc summary data=MySplits nway;
 84         class Symbol;
```

```
85          var split;
86          Output out=SplitMinMax(drop=_type_) min=SplitMin max=SplitMax;
87          run;

NOTE: There were 24 observations read from the data set WORK.MYSPLITS.
NOTE: The data set WORK.SPLITMINMAX has 14 observations and 4 variables.
NOTE: PROCEDURE SUMMARY used (Total process time):
      real time           0.01 seconds
      cpu time            0.00 seconds


88
89          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
102
```

The MEANS Procedure

Analysis Variable : Split

Symbol	N Obs	Maximum	Minimum
ABCO	1	0.5000000	0.5000000
APOL	3	0.6666667	0.6666667
CECO	3	0.5000000	0.5000000
COCO	3	0.5000000	0.5000000
DV	1	0.5000000	0.5000000
EDU	1	0.2500000	0.2500000
EPAX	1	0.5000000	0.5000000
ESI	1	0.5000000	0.5000000
HMNY	2	4.0000000	2.5000000
HRB	2	0.5000000	0.5000000
RGS	1	0.6666667	0.6666667
STNR	1	0.6666667	0.6666667
TISI	1	0.5000000	0.5000000
TYC	3	6.6875000	0.5000000

Total rows: 14 Total columns: 4

	Symbol	_FREQ_	SplitMin	SplitMax
1	ABCO	1	0.5	0.5
2	APOL	3	0.6666666667	0.6666666667
3	CECO	3	0.5	0.5
4	COCO	3	0.5	0.5
5	DV	1	0.5	0.5
6	EDU	1	0.25	0.25
7	EPAX	1	0.5	0.5
8	ESI	1	0.5	0.5
9	HMNY	2	2.5	4
10	HRB	2	0.5	0.5
11	RGS	1	0.6666666667	0.6666666667
12	STNR	1	0.6666666667	0.6666666667
13	TISI	1	0.5	0.5
14	TYC	3	0.5	6.6875

11. Build a summary of one record per symbol summarizing the options, split and dividend information as detailed in the previous instructions and demonstrated in the web video.

SOLUTION:

```
data OnePerSymbolStart;
merge MyCompanies(in=OnBase keep=symbol)
          SplitMinMax(in=OnSplits)
        DivCalc(in=OnDiv)
   ;
by Symbol;
if OnBase;
run;

options nolabel;
proc freq data=MyOptions;
table Symbol/out=OptionsCount(drop=Percent rename=(count=OptionsCount));
run;
options label;
```

```
proc transpose data=OptionStrikes(drop=_type_ _freq_)
         out=OptionsTransposed Prefix=StrikePrice_;
by symbol; id type; var strike;
run;

options nolabel;
data OnePerSymbolRound2;
merge MyCompanies(in=OnBase keep=symbol)
   SplitMinMax(in=OnSplits rename=(_freq_=SplitCount))
   work.DivCalc (in=OnDiv drop=_type_ _freq_ adjclose)
   OptionsCount(in=OnOptions)
   OptionsTransposed(in=OptionsPrices drop=_NAME_)
   ;
by Symbol;
if OnBase;
run;
options label;


data OnePerSymbolNoBlanks;
set OnePerSymbolRound2;
format StrikePrice_C StrikePrice_P 8.2;
array numbervars _numeric_;
do over numbervars;
if numbervars=. Then numbervars=0;
end;
run;

data OnePerSymbolNoBlanks;
set OnePerSymbolRound2;
format StrikePrice_C StrikePrice_P 8.2;
array BlankToZero SplitCount DivYield DivAmount OptionsCount;
do over BlankToZero;
if BlankToZero =. Then BlankToZero =0;
end;
run;

proc export data=OnePerSymbolNoBlanks
outfile="/folders/myfolders/OnePerSymbolNoBlanks.csv "
dbms=csv
replace;
run;
```

LOG & RESULTS:

```
1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
61
62
63         data OnePerSymbolStart;
64         merge MyCompanies(in=OnBase keep=symbol)
65                     SplitMinMax(in=OnSplits)
66               DivCalc(in=OnDiv)
67         ;
68         by Symbol;
69         if OnBase;
70         run;

NOTE: There were 48 observations read from the data set WORK.MYCOMPANIES.
NOTE: There were 14 observations read from the data set WORK.SPLITMINMAX.
NOTE: There were 27 observations read from the data set WORK.DIVCALC.
NOTE: The data set WORK.ONEPERSYMBOLSTART has 48 observations and 8 variables.
NOTE: DATA statement used (Total process time):
      real time           0.00 seconds
      cpu time            0.01 seconds


71
72         options nolabel;
73         proc freq data=MyOptions;
74         table Symbol/out=OptionsCount(drop=Percent
rename=(count=OptionsCount));
75         run;

NOTE: There were 427 observations read from the data set WORK.MYOPTIONS.
NOTE: The data set WORK.OPTIONSCOUNT has 32 observations and 2 variables.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.06 seconds
      cpu time            0.07 seconds


76         options label;
77
78         proc transpose data=OptionStrikes(drop=_type_ _freq_)
79          out=OptionsTransposed Prefix=StrikePrice_;
80         by symbol; id type; var strike;
81         run;

NOTE: There were 63 observations read from the data set WORK.OPTIONSTRIKES.
NOTE: The data set WORK.OPTIONSTRANSPOSED has 32 observations and 4 variables.
NOTE: PROCEDURE TRANSPOSE used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds


82
83         options nolabel;
84         data OnePerSymbolRound2;
85         merge MyCompanies(in=OnBase keep=symbol)
86               SplitMinMax(in=OnSplits rename=(_freq_=SplitCount))
87               work.DivCalc (in=OnDiv drop=_type_ _freq_ adjclose)
88               OptionsCount(in=OnOptions)
```

```
 89                OptionsTransposed(in=OptionsPrices drop=_NAME_)
 90                ;
 91          by Symbol;
 92          if OnBase;
 93          run;

 NOTE: There were 48 observations read from the data set WORK.MYCOMPANIES.
 NOTE: There were 14 observations read from the data set WORK.SPLITMINMAX.
 NOTE: There were 27 observations read from the data set WORK.DIVCALC.
 NOTE: There were 32 observations read from the data set WORK.OPTIONSCOUNT.
 NOTE: There were 32 observations read from the data set WORK.OPTIONSTRANSPOSED.
 NOTE: The data set WORK.ONEPERSYMBOLROUND2 has 48 observations and 9 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.00 seconds
       cpu time            0.00 seconds


 94          options label;
 95
 96
 97          data OnePerSymbolNoBlanks;
 98          set OnePerSymbolRound2;
 99          format StrikePrice_C StrikePrice_P 8.2;
 100         array numbervars _numeric_;
 101         do over numbervars;
 102            if numbervars=. Then numbervars=0;
 103         end;
 104         run;
 NOTE: There were 48 observations read from the data set WORK.ONEPERSYMBOLROUND2.
 NOTE: The data set WORK.ONEPERSYMBOLNOBLANKS has 48 observations and 9 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.00 seconds
       cpu time            0.00 seconds
 105
 106         data OnePerSymbolNoBlanks;
 107         set OnePerSymbolRound2;
 108         format StrikePrice_C StrikePrice_P 8.2;
 109         array BlankToZero SplitCount DivYield DivAmount OptionsCount;
 110         do over BlankToZero;
 111            if BlankToZero =. Then BlankToZero =0;
 112         end;
 113         run;
 NOTE: There were 48 observations read from the data set WORK.ONEPERSYMBOLROUND2.
 NOTE: The data set WORK.ONEPERSYMBOLNOBLANKS has 48 observations and 9 variables.
 NOTE: DATA statement used (Total process time):
       real time           0.00 seconds
       cpu time            0.00 seconds

 114
 115         proc export data=OnePerSymbolNoBlanks
 116         outfile="/folders/myfolders/OnePerSymbolNoBlanks.csv "
 117         dbms=csv
 118         replace;
 119         run;

 NOTE: Unable to open parameter catalog: SASUSER.PARMS.PARMS.SLIST in update mode.
Temporary parameter values will be saved to
 WORK.PARMS.PARMS.SLIST.
 120
/*************************************************************
```

```
 121         *   PRODUCT:   SAS
 122         *   VERSION:   9.4
 123         *   CREATOR:   External File Interface
 124         *   DATE:      07MAY17
 125         *   DESC:      Generated SAS Datastep Code
 126         *   TEMPLATE SOURCE:  (None Specified.)
 127
***********************************************************************/
 128            data _null_;
 129            %let _EFIERR_ = 0; /* set the ERROR detection macro variable */
 130            %let _EFIREC_ = 0;     /* clear export record count macro variable
*/
 131            file '/folders/myfolders/OnePerSymbolNoBlanks.csv' delimiter=','
DSD DROPOVER lrecl=32767;
 132            if _n_ = 1 then        /* write column names or labels */
 133             do;
 134               put
 135                  "Symbol"
 136               ','
 137                  "SplitCount"
 138               ','
 139                  "SplitMin"
 140               ','
 141                  "SplitMax"
 142               ','
 143                  "DivYield"
 144               ','
 145                  "DivAmount"
 146               ','
 147                  "OptionsCount"
 148               ','
 149                  "StrikePrice_C"
 150               ','
 151                  "StrikePrice_P"
 152               ;
 153             end;
 154           set  ONEPERSYMBOLNOBLANKS   end=EFIEOD;
 155              format Symbol $5. ;
 156              format SplitCount best12. ;
 157              format SplitMin best12. ;
 158              format SplitMax best12. ;
 159              format DivYield percent8.1 ;
 160              format DivAmount best12. ;
 161              format OptionsCount best12. ;
 162              format StrikePrice_C 8.2 ;
 163              format StrikePrice_P 8.2 ;
 164             do;
 165               EFIOUT + 1;
 166               put Symbol $ @;
 167               put SplitCount @;
 168               put SplitMin @;
 169               put SplitMax @;
 170               put DivYield @;
 171               put DivAmount @;
 172               put OptionsCount @;
 173               put StrikePrice_C @;
 174               put StrikePrice_P ;
 175               ;
 176             end;
```

```
 177            if _ERROR_ then call symputx('_EFIERR_',1);  /* set ERROR detection
macro variable */
 178            if EFIEOD then call symputx('_EFIREC_',EFIOUT);
 179            run;

 NOTE: The file '/folders/myfolders/OnePerSymbolNoBlanks.csv' is:
       Filename=/folders/myfolders/OnePerSymbolNoBlanks.csv,
       Owner Name=root,Group Name=vboxsf,
       Access Permission=-rwxrwx---,
       Last Modified=08May2017:04:31:24

 NOTE: 49 records were written to the file
'/folders/myfolders/OnePerSymbolNoBlanks.csv'.
       The minimum record length was 17.
       The maximum record length was 95.
 NOTE: There were 48 observations read from the data set WORK.ONEPERSYMBOLNOBLANKS.
 NOTE: DATA statement used (Total process time):
       real time           0.00 seconds
       cpu time            0.00 seconds

 48 records created in /folders/myfolders/OnePerSymbolNoBlanks.csv from
ONEPERSYMBOLNOBLANKS.

 NOTE: "/folders/myfolders/OnePerSymbolNoBlanks.csv" file was successfully created.
 NOTE: PROCEDURE EXPORT used (Total process time):
       real time           0.04 seconds
       cpu time            0.03
seconds


 180
 181        OPTIONS NONOTES
NOSTIMER NOSOURCE
NOSYNTAXCHECK;
 194
```

Symbol	SplitCount	SplitMin	SplitMax	DivYield	DivAmount	OptionsCount	StrikePrice_C	StrikePrice_P
ABCO	1	0.5	0.5	0.00%	0	11	58	66.67
APEI	0			0.00%	0	14	42.5	33.44
APOL	3	0.66666667	0.66666667	0.00%	0	39	27.96	22.31
ASCMA	0			0.00%	0	0		
ATAI	0			9.00%	0.584	0		
BPI	0			0.00%	0	7	19.33	15.75
CECO	3	0.5	0.5	0.00%	0	14	5.8	5
COCO	3	0.5	0.5	0.00%	0	8	2.5	2.5
CPLA	0			3.90%	1.4	11	60	48.75
CSV	0			5.10%	0.275	8	20	17.5
DL	0			58.10%	1.08	0		
DV	1	0.5	0.5	2.20%	0.83	7	37.5	25.83
EDMC	0			0.00%	0	18	15.21	12.5
EDU	1	0.25	0.25	2.80%	0.65	28	26.45	22.88
EPAX	1	0.5	0.5	8.90%	0.36	0		
ESI	1	0.5	0.5	0.00%	0	19	37.75	27.78
EVI	0			114.90%	1	0		
FC	0			0.00%	0	0		
FTD	0			0.00%	0	0		
GK	0			70.20%	14.69	0		
GPX	0			0.00%	0	7	28.33	23.13
HMNY	2	2.5	4	8.80%	0.17	0		
HRB	2	0.5	0.5	14.90%	2.2	25	29.42	24.92
IT	0			0.00%	0	16	58.57	58.33
LINC	0			10.60%	0.72	4	6.25	6.25
LOPE	0			0.00%	0	22	43.64	32.73
LRN	0			0.00%	0	24	24.81	21.59
LTRE	0			0.00%	0	0		
MANT	0			7.70%	2.31	3	30	30
NAUH	0			6.70%	0.451	0		
PFMT	0			0.00%	0	6	10	10
RGS	1	0.66666667	0.66666667	3.00%	0.48	6	17.5	14.17
SCI	0			7.90%	0.8	4	16.88	
SERV	0			0.00%	0	0		
SPRO	0			9.50%	0.157	0		
STNR	1	0.66666667	0.66666667	0.00%	0	0		
STON	0			36.90%	6.54	9	25.83	25
STRA	0			4.50%	4	11	46	32.92
TAX	0			0.00%	0	0		
TISI	1	0.5	0.5	0.00%	0	4	35	27.5
TYC	3	0.5	6.6875	7.40%	1.67284	26	36.07	31.82
UNF	0			0.70%	0.415	0		
UTI	0			9.10%	1.1	2	15	12.5
WTW	0			2.30%	1.225	20	39.75	33.75
XOXO	0			0.00%	0	3	15	10
XRS	0			5.30%	0.5	8	18	19.17
XUE	0			23.10%	0.66	3	5	5
YELP	0			0.00%	0	40	71.75	68.25

Part II

Evaluate the stocks in your sector and industry using the cut-off year and method specified in the assignment spreadsheet and as discussed in the videos providing background and demonstration of the application of that technique on Blackboard. Retain all graphs and regression output specified in the demonstration video. Write a brief synopsis as to whether the evaluation method was predictive of the performance of the stocks in your industry and sector and why. A few paragraphs is all that is required for the writing portion of this assignment. You will also be required to turn in some combination of your code and log, so please save it.

SOLUTION:

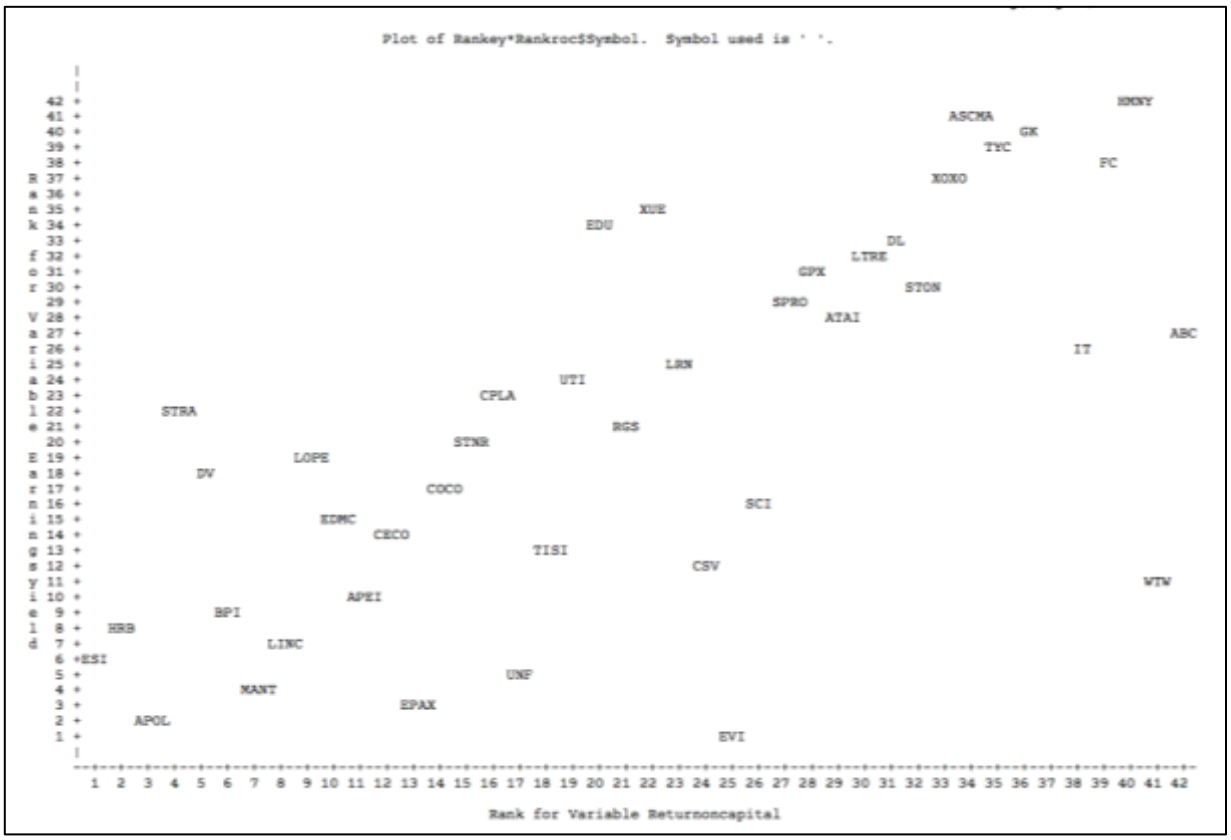
```
libname stocks "/folders/myfolders/";

Data Mycompany;
Set Stocks.Annualreports;
Format Infoavaildate Yymmdd10.;
Where Sector = "Consumer Servic" And Industry = "Other Consumer
Servic";
Fiscalyeardate = Datepart(Indfinancialyearend);
Fiscalyear = Year(Fiscalyeardate);
Infoavaildate = Input(Inddateprelimloaded, Yymmdd10.);
Run;

Proc Sort Data = Mycompany Nodupkey;
By Symbol Indfinancialyearend;
Run;

Data Report2009;
Set Mycompany (Keep = Fiscalyear Ebit Bstotalcurrentliabilities
Bsltdebt Bsminorintliab Bsprefstockeq Bscash Bsnetfixedass Bswc Symbol
Infoavaildate Bssharesoutcommon);
Where Fiscalyear = 2009;
Returnoncapital = Ebit/(Bsnetfixedass + Bswc);
Run;

Proc Rank Data = Report2009 Out = Report2009roc Descending;
Var Returnoncapital;
Ranks Rankroc;
Run;

Data Getprices;
Merge Report2009roc (In = Onbase)
Stocks.Prices (In = Onprices Rename = (Tic = Symbol) Keep = Tic Date
Close Adjclose);
By Symbol;
If Onbase And Date = Infoavaildate;
Run;
```

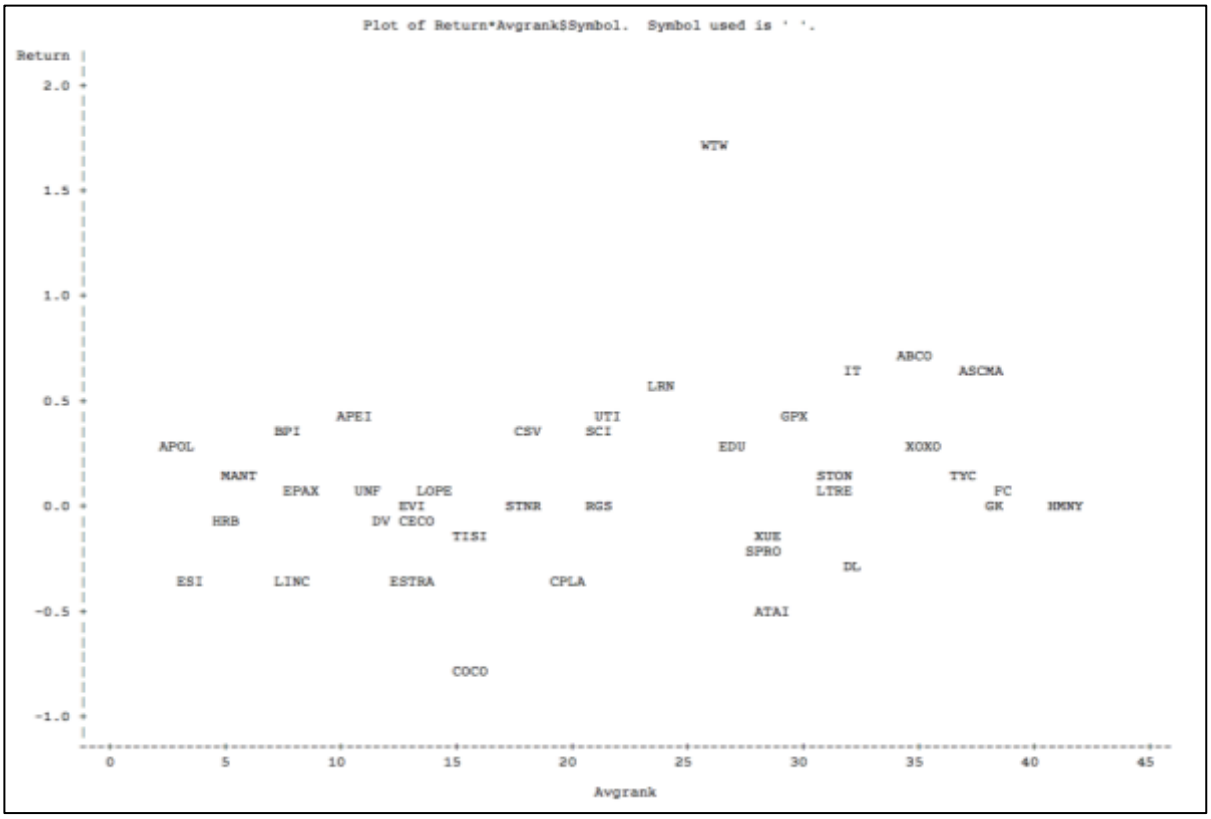
```
Proc Freq Data = Getprices;
Tables Symbol;
Title "GetPrices";
Run;
Title;

Data Getprices2;
Merge Report2009roc (In = Onbase)
Stocks.Prices (In = Onprices Rename = (Tic = Symbol) Keep = Tic Date
Close Adjclose);
By Symbol;
If Onbase And Infoavaildate <= Date <= Infoavaildate + 5;
Run;

Proc Freq Data = Getprices2;
Tables Symbol;
Title "GetPrices2";
Run;
Title;

Data Getpricesfirst;
Set Getprices2;
By Symbol Date;
If First.Symbol;
Run;

Data Earningsyield;
Set Getpricesfirst;
Marketcap = Close * Bssharesoutcommon;
Earningsyield = Ebit / (Marketcap + Bstotalcurrentliabilities +
Bsltdebt + Bsminorintliab + Bsprefstockeq - Bscash);
Run;

Proc Rank Data = Earningsyield Out = Eyandrocrank Descending;
Var Earningsyield;
Ranks Rankey;
Run;

Proc Plot Data = Eyandrocrank;
Plot Rankey * Rankroc =' '$Symbol;
Run;
Quit;

Data Avgrank;
Set Eyandrocrank;
Avgrank = (Rankey + Rankroc)/2;
Run;
```

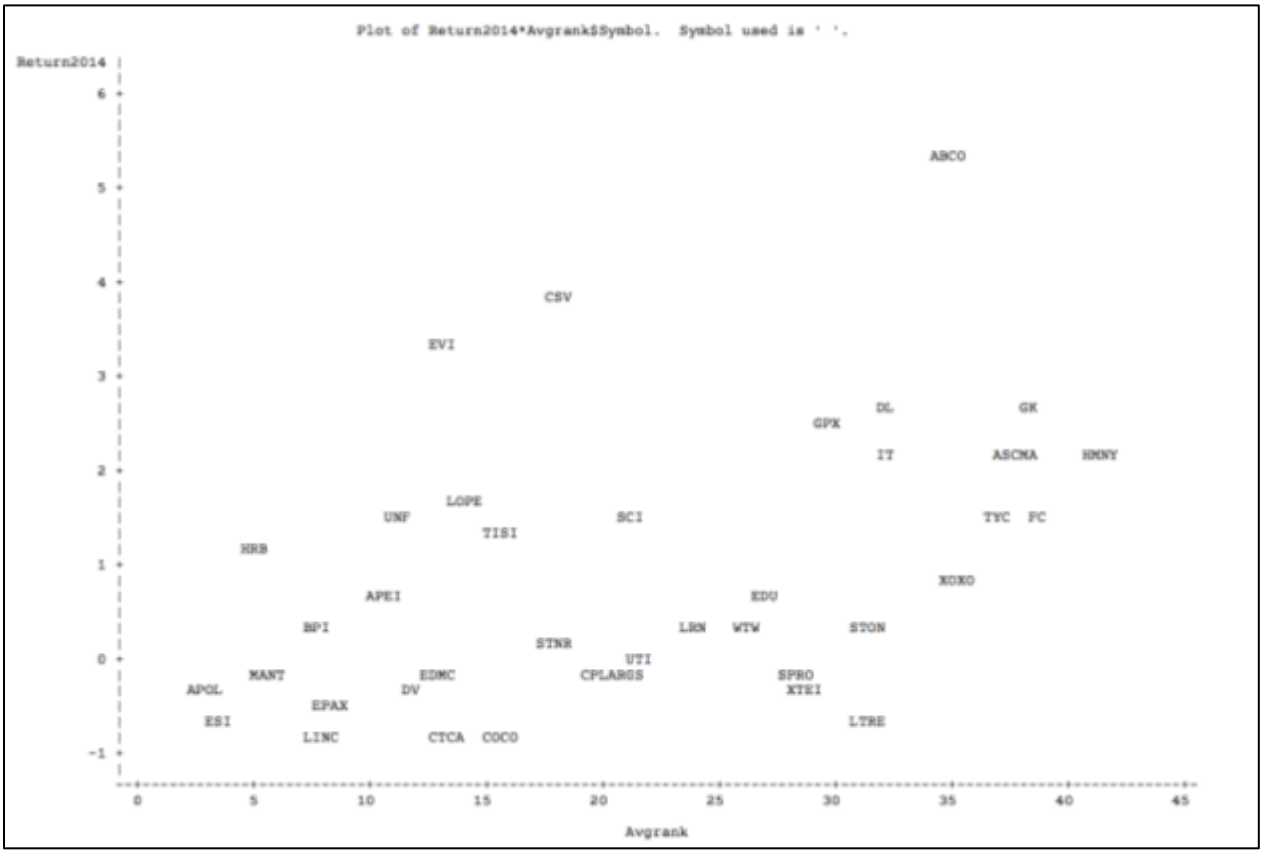
```
Data Mycompaniesoneyearlater (Keep = Symbol Fiscalyear Infoavaildate);
Set Stocks.Annualreports;
Format Infoavaildate Yymmdd10.;
Where Sector = "Consumer Servic" And Industry = "Other Consumer
Servic";
Fiscalyeardate = Datepart(Indfinancialyearend);
Fiscalyear = Year(Fiscalyeardate);
Infoavaildate = Input(Inddateprelimloaded, Yymmdd10.);
If Fiscalyear = 2010;
Run;

Data Oneyearlaterwithprice;
Merge Mycompaniesoneyearlater (In = Oncompanies)
Stocks.Prices (In = Onprices Rename = (Tic=Symbol
Adjclose=Lateradjclose) Keep = Tic Date Close Adjclose);
By Symbol;
If Infoavaildate-5 <= Date <= Infoavaildate-1;
Run;

Data Pricebeforenextreport;
Set Oneyearlaterwithprice;
By Symbol Date;
If Last.Symbol;
Run;

Data Evalbeforenextreport;
Merge Avgrank (In = Onbase)
Pricebeforenextreport (In = Onnext);
By Symbol;
If Onbase;
Return = (Lateradjclose-Adjclose)/Adjclose;
Run;

Proc Plot Data = Evalbeforenextreport;
Plot Return * Avgrank =' ' $Symbol;
Run;
Quit;

Data Muchlaterprice (Keep = Tic Adjclose Rename = (Tic=Symbol
Adjclose=Adjclose2014));
Set Stocks.Prices;
If Date = "02jan2014"d;
Run;

Data Laterreturn;
Merge Evalbeforenextreport (In = Onbase)
Muchlaterprice (In = Onlater);
```

```
By Symbol;
If Onbase;
Return2014 = (Adjclose2014-Adjclose)/Adjclose;
Run;

Proc Plot Data = Laterreturn;
Plot Return2014 * Avgrank =' ' $Symbol;
Run;
Quit;

Proc Reg Data = Laterreturn;
Model Return2014 = Avgrank;
Run;

Proc Reg Data = Laterreturn;
Model Return = Avgrank;
Run;
Quit;
```

INTERPRETATIONS AND RESULTS:




Plot of Return*Avgrank$Symbol. Symbol used is ' '.
Return
2.0
1.5
1.0
0.5
0.0
-0.5
-1.0
WTW
ABCO
IT
ASCMA
LRN
APEI
UTI
GPX
BPI
CSV
SCI
APOL
EDU
XOXO
MANT
STON
TYC
EPAX
UNF
LOPE
LTRE
FC
EVI
STNR
RGS
GK
HMNY
HRB
DV
CECO
TISI
XUE
SPRO
DL
ESI
LINC
ESTRA
CPLA
ATAI
COCO
0
5
10
15
20
25
30
35
40
45
Avgrank


Plot of Return2014*Avgrank$Symbol. Symbol used is ' '.
Return2014
6
5
4
3
2
1
0
-1
ABCO
CSV
EVI
DL
GK
GPX
IT
ASCMA
HMNY
LOPE
UNF
SCI
TYC
FC
TISI
HRB
XOXO
APEI
EDU
BPI
LRN
WTW
STON
STNR
UTI
MANT
EDMC
CPLARGS
SPRO
APOL
DV
XTEI
EPAX
ESI
LTRE
LINC
CTCA
COCO
0
5
10
15
20
25
30
35
40
45
Avgrank

The REG Procedure
Model: MODEL1
Dependent Variable: Return2014

Number of Observations Read	42
Number of Observations Used	41
Number of Observations with Missing Values	1

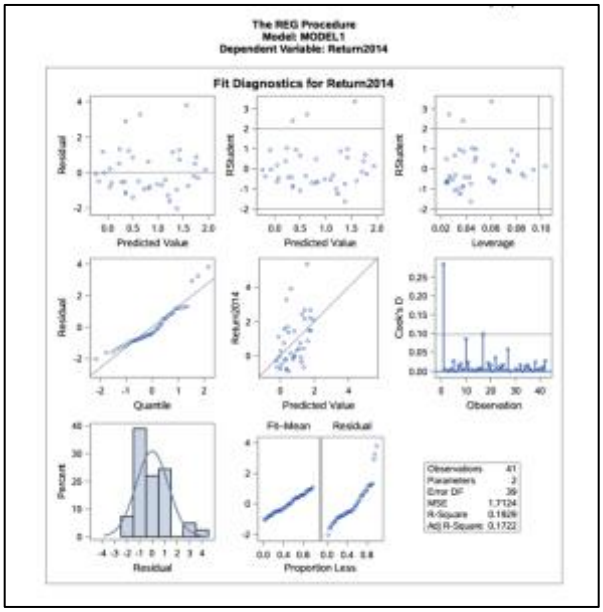
Analysis of Variance

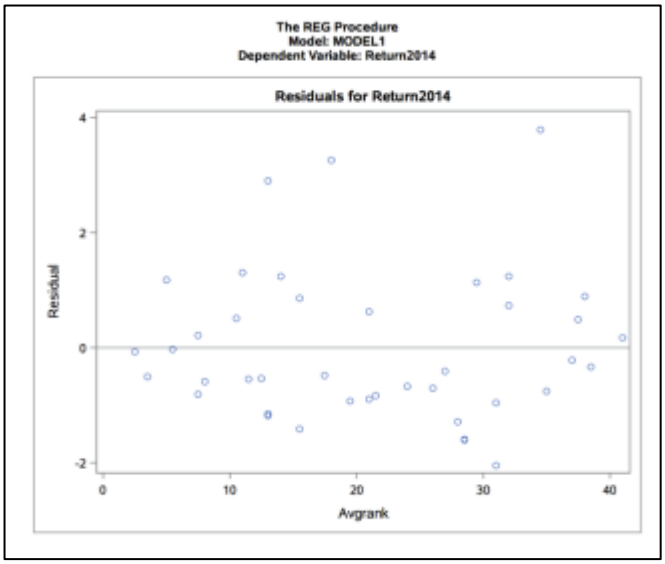
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	15.95964	15.95964	9.32	0.0041
Error	39	66.78554	1.71245		
Corrected Total	40	82.74517			

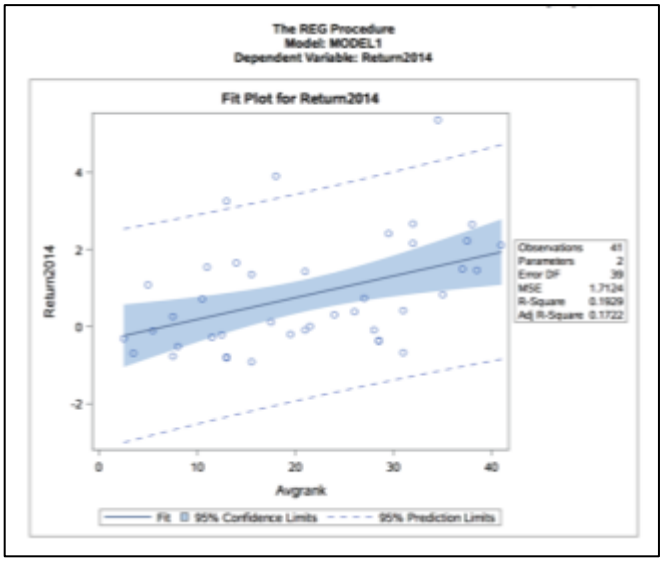
Root MSE	1.30861	R-Square	0.1929
Dependent Mean	0.81421	Adj R-Sq	0.1722
Coeff Var	160.72016		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-0.37707	0.44050	-0.86	0.3972
Avgrank	1	0.05637	0.01846	3.05	0.0041







The REG Procedure
Model: MODEL1
Dependent Variable: Return

Number of Observations Read	42
Number of Observations Used	41
Number of Observations with Missing Values	1

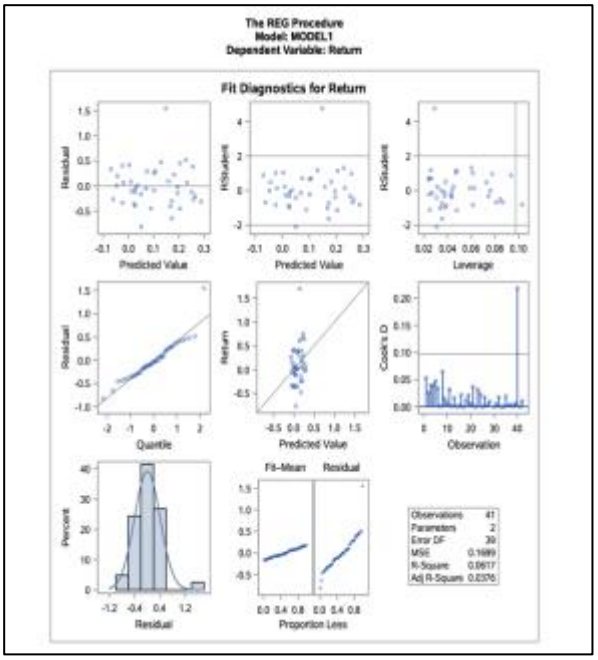
Analysis of Variance

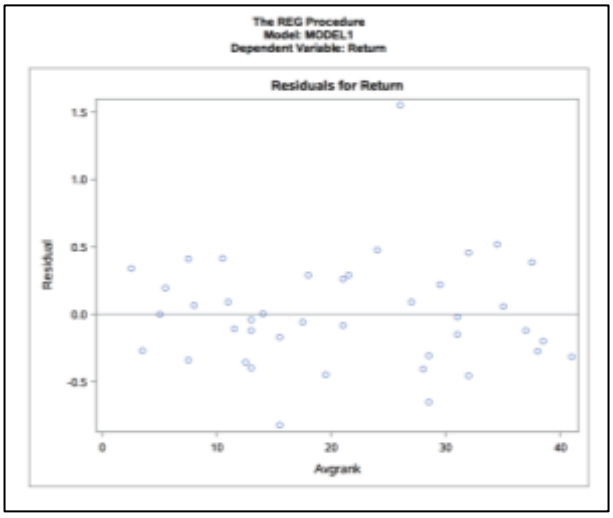
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	0.43546	0.43546	2.56	0.1174
Error	39	6.62527	0.16988		
Corrected Total	40	7.06072			

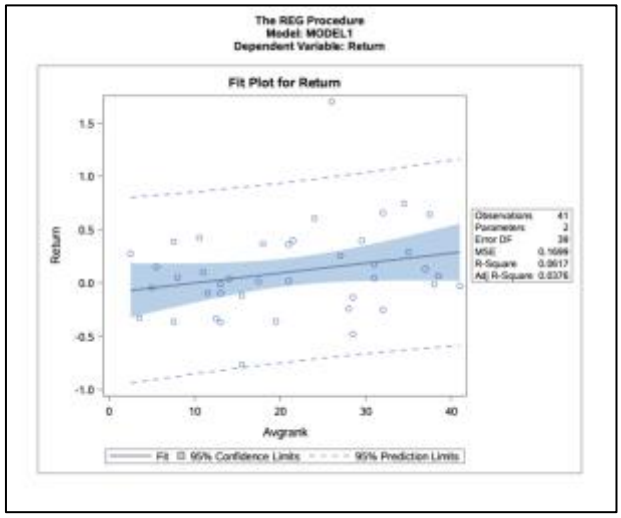
Root MSE	0.41216	R-Square	0.0617
Dependent Mean	0.10290	Adj R-Sq	0.0376
Coeff Var	400.53936		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-0.09388	0.13874	-0.68	0.5026
Avgrank	1	0.00931	0.00582	1.60	0.1174







From the descriptive plot of Rank of Variable Earnings Yield Vs Rank of Variable Return on Capital, the companies at the lower left bottom (APOL, ESI, MANT, HRB, BPI, LINC) are the ones which are the good performers. With the exception of these few good performers, most of the poorly performing companies show a linear trend (i.e a high rank in both earnings yield and return on capital)

From the graphs for AvgRank Vs Return for the years 2010 and 2014, the well performing companies continue to be consistent in their rankings with either an increase or a decrease in their returns. So there is no significant trend that can be seen from the two graphs.

From the regression results for the years 2010 and 2014, we can see that the predicted values for stock prices for the year 2014 are 17.22% (adj.R^2) closer to actual values. The p-value for the year 2014 is 0.004 which is less that the standard alpha values (1%, 5% and 10%). The p-value for the year 2010 is 0.1174 which is higher than the standard alpha values. As a result, we can conclude that there is not much predictive power in terms of the Little Book That Beats the Market method.

NOTE: The log for the PART2 of the final project is attached as a separate file due to larger file size.