Project 1
Forecasting Earnings

SAS

Luca Katsavos

ACO5170

TP3 2022

Data Preparation

The data in this project concern the financial positions of different companies at various times over the last two decades. The data set contains items you would normally see in a company financial report, and are commonly used by analysts in forecasting figures like company profit and share price. The original data set contains some 280,000 rows of data across 21 variables. Through our data preparation process, we will be selecting a sample size smaller than this. One thing to note about the data is the dollar values are clearly abbreviated for simplicity in the data set, but we can assume these values are in millions. For instance, the figure on the right shows an example of a Net Income value. This can be read as $18,531,000. While it does not affect the statistical workings of our analysis, it helps to understand the true value of these numbers when conducting analyses. This report involves creating, estimating, selecting, and backtesting a range of models in SAS. We will start by selecting the bulk of our sample through some specified criteria, followed by creating our new variables.

ni
18.5310

1.0 - $18,531,000

```
DATA EARNINGS1; SET ASS1.earnings_project1;
FORMAT FIRST 1.;
FIRST = SUBSTR(GVKEY, 1, 1);
IF FIRST LE 4;
DROP FIRST
```

1.1 - Criteria for Sample Selection

Figure 1.1 shows the beginning of our initial DATA step. We create a temporary variable, FIRST, which is simply the first character of GVKEY (our company identifier). We only allow entries through to our sample data if FIRST has a value of less than or equal to 4.

We now must construct our needed variables. This process involves calculations using variables from the original data. Not all will be shown, however below is an example of 2 calculations, namely: Earnings per Share (EPS) and Net Operating Assets (NOA). Note the SUM function, which treats any missing data points as having 0 in value.

```
EPS = NI/CSHO;
NOA = SUM(DLTT, DLC, CEQ, PSTK, MIB, -CHE, -IVST);
```

1.2 - Example Variable Creation

The next two steps are necessary for cleaning our data. We should first delete observations where there is missing data for our created variables, as these entries are essentially of no use to us. This is achieved using an array and a DO loop, shown in Figure 1.3 (right). Moreover, we must mitigate the impact of extreme values like outliers within our sample, so that our statistical outcomes aren't swayed as much. A way to do this without losing observations is through a process called winsorizing. We will winsorize at 1 percent for each year of data, meaning we will convert all values below the 1st percentile to the value of the 1st percentile itself for a particular year, and all values above the 99th percentile to the value of the 99th. The SAS procedures that action this can be found in the figure below.

```
ARRAY V(9) EPS--SGX;
DO I = 1 TO 9;
IF V(I) = . THEN DELETE;
END;
```

1.3 - Removing Missing Data

```
PROC SORT DATA = EARNINGS1; BY FYEAR; RUN;

PROC MEANS DATA = EARNINGS1 NOPRINT; BY FYEAR;
VAR EPS--SGX;
OUTPUT OUT = WINS (DROP=_TYPE_ _FREQ_) P1 = L1-L9 P99 = H1-H9;
RUN;

DATA ASS1.EARNINGS; MERGE EARNINGS1 WINS; BY FYEAR;
ARRAY V(*) EPS--SGX;
ARRAY H(*) H1-H9;
ARRAY L(*) L1-L9;
DO I = 1 TO 9;
IF V(I) < L(I) THEN V(I) = L(I);
IF V(I) > H(I) THEN V(I) = H(I);
END;
KEEP GVKEY FYEAR EPS--SGX;
RUN;
```

1.4 - Winsorizing (1%)

Our data is first sorted on FYEAR. The 1st and 99th percentile values for each of our continuous variables are exported to a data set named WINS through a MEANS procedure. WINS is then merged with our original data set, where a DO loop checks if the value of any variable is outside the respective 1st and 99th percentiles, amending the value to match the closest percentile if this is the case. This winsorizing process should ultimately lessen the influence of extreme values on our statistical results, whilst also keeping observations.

The last line of code in Figure 1.4 says to only keep certain variables. We are finally left with GVKEY and FYEAR as our characterising variables, EPS to be predicted through simple models, and RNOA—SGX to be used in our regression models later on.

Descriptive Statistics

This section will provide some basic descriptive statistics of our variables. A PROC CONTENTS procedure in SAS foremost provides us with an overview of our whole sample set. Below are two snippets from the SAS output.

Observations	129005
Variables	11

2.0 - Dimensions

#	Variable	Type	Len	Format	Informat	Label
9	AR	Num	8			Receivables Ratio
6	ATO	Num	8			Asset Turnover
3	EPS	Num	8			Earnings per Share
10	ETR	Num	8			Effective Tax Rate
8	INV	Num	8			Inventory Ratio
5	PM	Num	8			Profit Margin
4	RNOA	Num	8			Return on Net Operating Assets
11	SGX	Num	8			Gen Exp to Sales Ratio
7	TACC	Num	8			Total Accruals
2	fyear	Num	8	F6.	6.	Data Year - Fiscal
1	gvkey	Char	6	$6.	$6.	Global Company Key

2.1 - Variables and Attributes

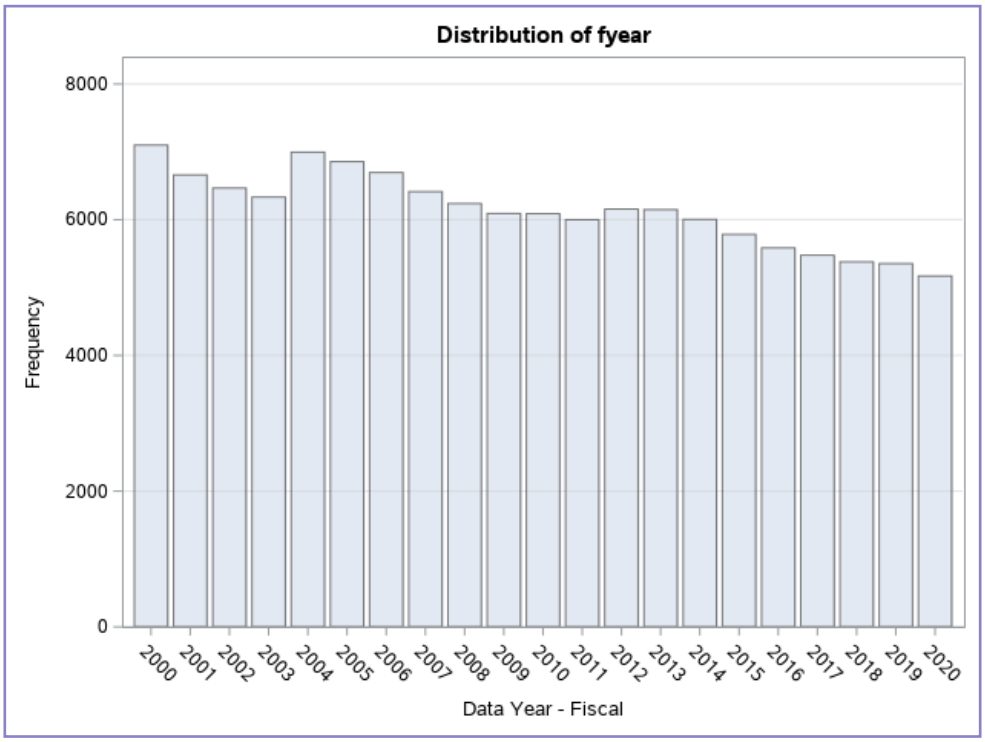
The left figure shows we have 129,005 rows of sample data across 11 variables. The right provides a list of these variables, where the 'Label' column briefly informs us of their economic meanings.

COUNT	Frequency	Percent	Cumulative Frequency	Cumulative Percent
2	1895	11.88	1895	11.88
1	1606	10.07	3501	21.95
3	1455	9.12	4956	31.08
21	1403	8.80	6359	39.88
5	1204	7.55	7563	47.43
4	1169	7.33	8732	54.76

2.2 - Frequency of GVKEY Incidences

We are interested in knowing how many years of data a company usually holds in our sample. It was previously mentioned that GVKEY acts as our company identifier, in that it tells us the unique company of the entry. Figure 2.2 (left) illustrates the frequency distribution of GVKEY incidences. It tells us that most companies have only 2 years of financial data, making up 11.88% of our sample. We see that only 8.8% of companies have a full set of data (2000-2020), and can determine that around 46% of entries are from companies with 5 or less entries. If we were to construct models for those companies with fewer years of data separate to those with more, those with less would presumably yield not as accurate forecasts due to the irregularity of values within the alternating company

entries, as companies ultimately possess financial information that is independent of one another.



2.3 - Distribution of FYEAR

FYEAR is our other discrete variable. Its distribution is shown above. As time moves on, there is a clear downwards trend in the quantity of data. While this could be for a multitude of reasons, it is important to recognise this pattern. We can guess that models built on earlier data will be more accurate because there is simply more data to estimate from. Nonetheless, more recent data will always have the upper hand in predicting the future due to the fact it is newer and more relevant.

To better understand our remaining, continuous variables, a PROC MEANS is performed. The output is shown in the figure below.

Variable	Label	Mean	Median	Std Dev	Minimum	Maximum	Skewness
EPS	Earnings per Share	0.676	0.188	5.421	-57.275	93.557	4.694
RNOA	Return on Net Operating Assets	0.063	0.092	1.959	-17.720	15.711	-0.436
PM	Profit Margin	-0.789	0.058	4.694	-65.491	0.630	-8.135
ATO	Asset Turnover	1.664	1.155	5.810	-33.907	41.938	0.805
TACC	Total Accruals	-186.409	-10.719	742.722	-12972.000	828.000	-7.648
INV	Inventory Ratio	0.181	0.053	0.441	-2.039	2.927	1.370
AR	Receivables Ratio	0.596	0.198	2.712	-19.126	24.544	2.117
ETR	Effective Tax Rate	0.143	0.169	0.457	-4.578	3.565	-1.987
SGX	Gen Exp to Sales Ratio	1.009	0.278	3.929	0.013	58.520	8.188

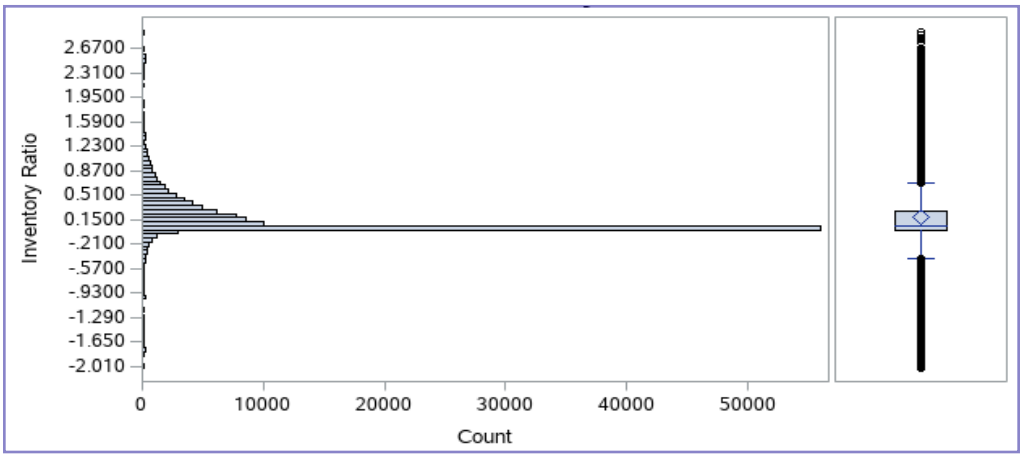
2.4 - PROC MEANS

We first notice that all the reported statistics involve fairly low values, except for those of TACC. This is because TACC is not a ratio. We see TACC's mean appears to be heavily influenced by negative outliers, where it also has a skewness of -7.648. The majority of our companies must earn more in cash than they do in accruals. For some companies, this is drastically the case.

When we consider those variables that *are* ratios, we see that EPS, PM, ATO, and SGX all appear to have the highest standard deviations. Their values are more spread out. When we look at how these four variables were made, we see they all happen to be comprised of some figure involving the *sales* of a company; either Net Income, or Sales itself. We therefore suspect a large variation in the sales figures of different companies.

RNOA and ETR appear to have very little difference between their means and medians. Their data is not heavily influenced by outliers, and must be somewhat symmetrical. We can suspect Effective Tax Rate (ETR), to have fairly centred data, as the rate at which companies pay tax is ultimately restricted.

We see that INV has the smallest standard deviation out of all our variables, as well as very little difference between its mean and median. The figure below shows its distribution, where we can observe the data is, predictably, quite symmetrical. It is apparent that within our data there is not a huge variation in what portion a company's operating assets are made up of inventory.



2.5 - Distribution of INV

Basic descriptive statistics on our variables taught us that a 1% winsorize may not have been enough to reduce the impact of outliers on some of our data. All variables appear to have a skew of some sort, however for some, this is small and the data is relatively normally distributed.

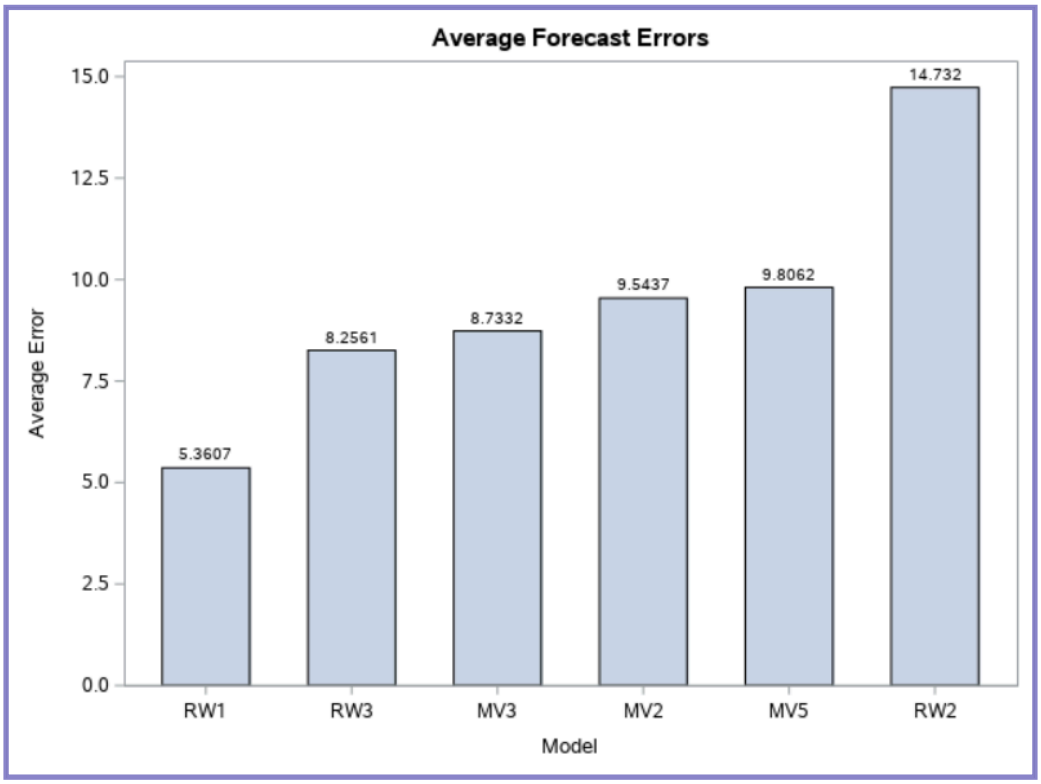
Simple Models

In this section, we will construct and discuss our forecasts for Earnings per Share (EPS). We must first create lagged variables of GVKEY, FYEAR, and EPS for 1, 2, 3 and 5 years, so that we can implement them into the appropriate models to forecast our earnings values. Note that for the purpose of this discussion, RW refers to random walk, and MV refers to moving average.

```
IF GVKEY = GVKEY1 = GVKEY2 = GVKEY3 AND
FYEAR = FYEAR1+1 = FYEAR2+2 = FYEAR3+3 THEN DO;
    RW3 = EPS3; ERROR_RW3 = ABS((EPS-EPS3)/EPS);
    MV3 = SUM(OF EPS1--EPS3)/3; ERROR_MV3 = ABS((EPS-MV3)/EPS);
END;
```

3.0 - 3 Year Models

Figure 3.0 shows our 3 year forecasting. In this example, if the current company ID matches the past 3 company IDs while there are also 4 continuous and uninterrupted years of data, a random walk and moving average value and their respective errors are calculated. These error values measure how correct each estimated EPS value is, and so, making use of some basic statistical measures, we are able to determine which forecast is most accurate.



3.1 - Average Forecast Errors

The figure above illustrates the means of our forecast errors. We see that our 1-year random walk model produces, on average, the most accurate forecasts of EPS with an average error of 5.36. Simply by changing the random walk model from 1-year to 2-year, however, we almost triple the average error to 14.73. Backtracking 1 more year, our 3-year random walk has an average of 8.25.

For our MV models, we see they all produce an average error of around 9. We notice how our MV forecasts are more similar in their averages than our RW forecasts, most likely because our MV models estimate from shared values, whereas our RW models are designed to estimate from one unique value at a time. Looking at their error averages only, it seems as though a company's earnings from 1 year previous are more of a predicting factor of their current year's earnings than from any other year.

Variable	Mean	Lower Quartile	Median	Upper Quartile	90th Pctl	95th Pctl	99th Pctl	Maximum
ERROR_RW1	5.361	0.218	0.590	1.411	4.139	8.889	49.397	89542.507
ERROR_RW2	14.732	0.308	0.755	1.754	5.325	11.810	68.384	727798.235
ERROR_MV2	9.544	0.249	0.632	1.487	4.410	9.468	52.245	363903.681
ERROR_RW3	8.256	0.359	0.833	1.899	5.809	12.950	73.083	33816.422
ERROR_MV3	8.733	0.267	0.647	1.489	4.339	9.307	51.803	253874.595
ERROR_MV5	9.806	0.500	1.103	2.747	8.008	16.921	93.571	16771.571

3.2 - Forecast Error Statistics

A PROC MEANS produces a summary of our errors, seen in the figure above. Observe the rows for our RW1, MV2 and MV3 errors, highlighted in blue. Although we previously claimed RW1 produced the most accurate forecasts, we now learn that up until the 99th percentile, so for 99% of our observations, these three models held relatively the same forecast accuracy. It is only the top 1% of values that sway the average errors of these forecasts so much.

In the same manner, the accuracy of our RW2 forecast, the one with the highest average error of 14.73, was obscured by extreme outliers. Looking at our new statistics, we learn that for 99% of our observations, RW2 forecast errors were in fact not as high as those of RW3 and MV5. RW2's top 1% of values is to blame for its unappealing mean.

Through this short analysis, we learned that outliers are very much affecting our results, where the top 1% of errors impacted the average of errors greatly. We can still safely conclude, however, that earnings forecasted using a 1-year random walk model were the most accurate.

Complex Models

Here we will perform our first set of regressions to predict Return on Net Operating Assets for one year forward (RNOA1). Our independent variables, or those that are trying to explain the variation in RNOA1, will be the same continuous variables as previously discussed, except for Earnings per Share. We must first curate a data set to be used in our regression estimates. (In this part of the report, we will be using our full sample set to estimate the model).

```
DATA NEARNINGS; SET ASS1.EARNINGS;
N = _N_;
RUN;

DATA FWD; SET ASS1.EARNINGS (KEEP = GVKEY FYEAR RNOA FIRSTOBS = 2);
N = _N_;
RENAME GVKEY = GVKEY1 FYEAR = FYEAR1 RNOA = RNOA1;
RUN;

DATA ASS1.REGMODEL; MERGE NEARNINGS FWD; BY N;
IF GVKEY = GVKEY1 AND FYEAR+1 = FYEAR1 THEN OUTPUT;
RUN;
```

4.0 - Preparing Data for Regression

There are three variables we first need to extract from our 'Earnings' data set: GVKEY, FYEAR and RNOA. By setting FIRSTOBS to 2 (see Figure 4.0), SAS begins reading the data in these columns from the second row. Essentially, this moves all observations up one row. After merging with our original data, we check if the current GVKEY and FYEAR values matches those of the next row, and if so, the forward RNOA value is kept. This criteria forbids the tainting of RNOA data with non-matching companies or non-consecutive observation years. We applied similar ideas to our EPS models.

R-Square	0.0231
Adj R-Sq	0.0230

4.1 - R Squared

Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	0.03419	0.00687	4.98	<.0001
RNOA	Return on Net Operating Assets	1	0.13892	0.00293	47.35	<.0001
PM	Profit Margin	1	0.04805	0.00571	8.41	<.0001
ATO	Asset Turnover	1	0.01099	0.00121	9.09	<.0001
TACC	Total Accruals	1	-0.00001513	0.00000769	-1.97	0.0491
INV	Inventory Ratio	1	-0.12867	0.01508	-8.54	<.0001
AR	Receivables Ratio	1	-0.00074556	0.00223	-0.33	0.7384
ETR	Effective Tax Rate	1	0.06187	0.01167	5.30	<.0001
SGX	Gen Exp to Sales Ratio	1	0.06497	0.00684	9.50	<.0001

4.2 - Parameter Estimates

Above we see the estimates of our regression model `RNOA1 = RNOA--SGX;`, where RNOA—SGX signifies the following variables from our model data set are to be used as regressors: `RNOA` `PM` `ATO` `TACC` `INV` `AR` `ETR` `SGX`.

Our initial observation is the rather low R-Squared, where our model only explains 2.3% of the variability in RNOA1 (Figure 4.1). Looking at Figure 4.2, we see our model is still a fit, regardless of how little explaining it does, where most of our variables are significant at a 1% level; TACC at the 5% level. Our Receivables Ratio is the only variable not statistically different from zero.

A stepwise regression should narrow our model down to include only the significant variables. Adding the option `SELECTION = STEPWISE;` to our existing code gives us the following results:

Step	Variable Entered	Variable Removed	Label	Number Vars In	Partial R-Square	Model R-Square	C(p)	F Value	Pr > F
1	RNOA		Return on Net Operating Assets	1	0.0210	0.0210	228.987	2399.34	<.0001
2	ATO		Asset Turnover	2	0.0002	0.0213	203.343	27.59	<.0001
3	INV		Inventory Ratio	3	0.0007	0.0219	129.071	76.19	<.0001
4	SGX		Gen Exp to Sales Ratio	4	0.0002	0.0222	104.404	26.64	<.0001
5	PM		Profit Margin	5	0.0006	0.0228	35.7760	70.61	<.0001
6	ETR		Effective Tax Rate	6	0.0003	0.0230	9.0059	28.77	<.0001
7	TACC		Total Accruals	7	0.0000	0.0231	7.1115	3.89	0.0484

4.1 - Stepwise Selection

Variable	Parameter Estimate
Intercept	0.03405
RNOA	0.13885
PM	0.04794
ATO	0.01086
TACC	-0.00001517
INV	-0.12894
ETR	0.06178
SGX	0.06485

4.2 - Paramaters

Predictably, all but our Receivables Ratio were added to the final model. Looking at the parameter estimates for our significant variables (left), we notice most have a positive correlation (are positive in value). For these, an increase in their value ultimately has a positive influence in RNOA1's estimation. As expected, our independent RNOA variable seems to have the highest correlation, and hence the most explaining power over RNOA1. Combining its estimate with our previous knowledge of the data, we can conclude that an increase of $138,850 in RNOA1 is expected to occur when RNOA increases by $1,000,000, that is, when holding all other independent variables constant. Moreover, INV appears to have almost the opposite affect, where if it increases by $1,000,000, RNOA1 is expected to decrease by $128,940, again, if all other independents remain unchanged. We must keep in mind this model holds an R-Squared of 0.0231, so these statements are true for only 2.31% of our data.

Model Backtesting

For this final section, we will be performing our second set of regression estimates. We will be splitting our sample into an estimation and test sample, performing the regression on the first and computing forecast errors on the second. This process will be repeated 100 times using a SAS macro.

```
%DO I = 1 %TO &N.;

    DATA REGEST REGTEST; SET REGMODEL;
    RANDOM = RANUNI(0);
    IF RANDOM LE 0.4 THEN OUTPUT REGEST; ELSE OUTPUT REGTEST;
    RUN;

    PROC REG DATA = REGEST NOPRINT OUTEST = COEFF;
    MODEL RNOA1 = RNOA--SGX / SELECTION = STEPWISE;
    RUN;

    DATA COEFF; SET COEFF;
    RENAME
        RNOA = CO_RNOA PM = CO_PM ATO = CO_ATO TACC = CO_TACC
        INV = CO_INV AR = CO_AR ETR = CO_ETR SGX = CO_SGX;
    DROP _MODEL_ _TYPE_ _DEPVAR_ _RMSE_ RNOA1;
    RUN;
```

5.0 - Splitting Data and Regressing

Here is the start of our macro. The DO statement acts as our repeater, it will cycle through the code below it 'N' amount of times. To select a random sample, we randomly assign values between 0 and 1 to our existing regression data set, outputting rows that get 0.4 or below to the estimating data set, and the remaining to the testing one. A seed of 0 is used with RANUNI so the random values are different each time. The model is then estimated on the estimate data, using stepwise selection to choose the most significant model. The coefficients of the regression are sent to a new data set, where we adjust our coefficient names, ready for merging with our test data.

Below we see the merging of our coefficients with the test data. Predicted RNOA1 values are determined, as well as their errors.

```
DATA EST; SET REGTEST;
IF _N_ = 1 THEN SET COEFF;
P_RNOA1 = SUM(INTERCEPT, CO_RNOA*RNOA, CO_PM*PM, CO_ATO*ATO, CO_TACC*TACC,
              CO_INV*INV, CO_AR*AR, CO_ETR*ETR, CO_SGX*SGX);
ERROR = ABS(RNOA1 - P_RNOA1);
RUN;
```

5.1 - Merging

The last part of our macro outputs a range of statistics for the forecast errors, either amending them to an existing data set, or creating a new data set with the statistics, depending on if it is the initial regression estimate.

```
        PROC MEANS DATA = EST NOPRINT;
        VAR ERROR;
        OUTPUT OUT = MEAN MEAN = MEAN MIN = MIN Q1 = Q1 MEDIAN = MEDIAN Q3 = Q3
                     P90 = P90 P95 = P95 P99 = P99 MAX = MAX;
        RUN;

        DATA MEAN; SET MEAN;
        EST = &I.;
        KEEP MEAN MIN Q1 MEDIAN Q3 P90 P95 P99 MAX EST;
        RUN;

        %IF &I. EQ 1 %THEN %DO;
            DATA ERRORSTATS; SET MEAN; RUN;
        %END;
        %ELSE %DO;
            PROC APPEND BASE = ERRORSTATS DATA = MEAN; RUN;
        %END;
    %END;
%MEND BACKTEST;
```

5.2 - Extracting Statistics

After we run the code `%BACKTEST(100);`, we are left with a data set containing 100 rows of statistics on the errors of each RNOA1 forecast. We can run further statistics to grasp the performance of the entirety of our estimates.

Variable	Minimum	Mean	Maximum
MEAN	0.608	0.619	0.632
MIN	0.000	0.000	0.000
Q1	0.041	0.044	0.048
MEDIAN	0.111	0.118	0.124
Q3	0.337	0.345	0.355
P90	1.222	1.262	1.295
P95	2.950	3.050	3.125
P99	9.513	9.656	9.777
MAX	18.030	18.935	19.541

5.3 - Forecast Error Statistics

Figure 5.3 (left) exhibits these statistics. Looking at the top left value of our table, we learn the model that had, on average, the most accurate forecasts, held a mean error of 0.608. In this way, the least accurate model produced a mean error of 0.632. We also learn the maximum error value of our forecasts was, on average, 18.935, and the highest error value achieved by any of our models was 19.541. These high errors cannot be from the mixing of company data, as precautions were taken to remove these RNOA1 values, so we can assume outliers are to blame. Our error values are very high when taking into account the scope of true RNOA1 values (seen on right). We should conclude that, at large, our models are not very accurate in predicting RNOA1, despite little variation in errors across models.

Median	Mean
0.0939003	0.0767374

5.3 - Forecast Error Statistics

Reference List

CSRP (Center for Research in Security Prices) (2022) *Annual Data - Industrial,* CSRP, accessed 1 June 2022. https://www.crsp.org/products/documentation/annual-data-industrial

PUL (Princeton University Library) (2007) *Interpreting Regression Output,* Princeton University, accessed 7 June 2022. https://dss.princeton.edu/online_help/analysis/interpreting_regression.htm