Project 2
Predicting Accounting Fraud

SAS

Luca Katsavos

ACO5170

TP3 2022

Intro & Data Preparation

The purpose of this report is to predict whether a company is undertaking fraudulent activities by estimating logistic regression models, and to then assess the accuracy of these estimates. The components of our models are according to the 2011 article by Dechow et al. The first step is constructing these components. We will then clean the data by removing missing values, winsorizing our continuous variables, and finally randomly selecting our smaller sample for analysis. Before we make our model components, we must tidy our data a bit, as, through inspection of the data, we see many companies have duplicate entries for the same year.

```
DATA FRAUD; SET FRAUD;
MISS = NMISS(OF ACT--PRCC_F);
RUN;

PROC SORT DATA = FRAUD; BY GVKEY FYEAR MISS; RUN;

DATA FRAUD; SET FRAUD;
BY GVKEY FYEAR MISS;
IF FIRST.FYEAR;
```

1.0 - Tidying Data

The above code groups entries that have the same company key and entry year, then only permits the entry in each group with the lowest count of missing values. Now our data is a little tidier, we can move to constructing our model components.

Three supporting variables are first made; a lagged company key (GVKEY1), a lagged entry year (FYEAR1), and average total assets (ATA). These variables will aid in the making of our model components. For each of our components, we will replicate the formula given in Dechow at al. (2011). Figure 1.1, for instance, shows the code that builds Change in Receivables (CH_REC). We see not only the formula for CH_REC, but a safeguard so that 'change' values are only calculated if the data is consecutive, i.e., there is sequential entry years and the company of entry matches that of the previous. This will retain the authenticity of our data, and is applied to every variable that involves previous year data.

```
*Change in Receivables;
RECT1 = LAG(RECT);
CH_RECT = RECT-RECT1;
IF GVKEY = GVKEY1 AND FYEAR = FYEAR1+1 THEN DO;
CH_REC = CH_RECT/ATA;
END;
```

1.1 - CH_REC

Let's now remove those observations that have a missing value for any of our model components. The SAS code that performs this is shown below. Arrays are used for efficiency.

```
ARRAY W(6) RSST_ACC CH_REC CH_INV SOFT_ASSETS CH_CS CH_ROA;
DO I = 1 TO 6;
IF W(I) = . THEN DELETE;
END;
RUN;
```

1.2 - Missing Values

We should also winsorize our continuous variables. This process involves setting a cut off point at which any value above or below the respective corresponding percentile is changed to that of the closest percentile. In our case, cut off will be 1%, so all values below the 1st percentile and above the 99th will be changed to the value of that percentile, respectively. Specifically, values are amended based on the 1st and 99th percentile value of their year of entry. Rather than just removing these values, winsorizing should leave us with data much less influenced by outliers, whilst also keeping the same number of observations.

Choosing our final sample has a few steps. The first is splitting our data by fraudulent and non-fraudulent cases. Of those non-fraudulent cases, 20,000 are chosen at random. These cases are then merged with all fraudulent cases, so our final data set has 20,000 + N observations, where N is equal to the number of fraudulent cases. N is further added as a variable to our sample set for use in calculations in later steps. These steps are shown in Figure 1.3 (left).

```
DATA FRAUD0 FRAUD1; SET FRAUD;
IF FRAUD = 0 THEN OUTPUT FRAUD0;
ELSE OUTPUT FRAUD1;
RUN;

PROC SQL;
CREATE TABLE FRAUD_N AS
    SELECT COUNT(*) AS FRAUD_N
    FROM FRAUD1;
QUIT;

PROC SURVEYSELECT DATA = FRAUD0 OUT = FRAUD0
METHOD = SRS
SAMPSIZE = 20000
SEED = 555
NOPRINT;
RUN;

DATA ASS2.FRAUD; SET FRAUD0 FRAUD1;
IF _N_ = 1 THEN DO;
SET FRAUD_N;
END;
```

1.3 - Winsorizing

After a final sortation by company key and entry year, our variables and sample data are ready. The next section will provide a basic understanding of each variable.

Descriptive Statistics

A PROC CONTENTS provides us with an overview of our variables. They are as follows: GVKEY and FYEAR are for characterising, FRAUD is for estimating, FRAUD_N is relevant in calculations later on, and the remaining are model components. A brief description of each is given in the 'Label' column.

#	Variable	Type	Len	Format	Informat	Label
8	CH_CS	Num	8			% Change in Cash Sales
6	CH_INV	Num	8			Change in Inventory
5	CH_REC	Num	8			Change in Receivables
9	CH_ROA	Num	8			Change in Return on Assets
3	FRAUD	Num	8	BEST12.	BEST32.	Fraud (Fraudulent = 1)
11	FRAUD_N	Num	8			Total No. of Fraudulent Cases
10	ISSUE	Num	8			Issuance (Issued Securities = 1)
4	RSST_ACC	Num	8			RSST Accruals
7	SOFT_ASSETS	Num	8			% Soft Assets
2	fyear	Num	8	BEST12.	BEST32.	Fiscal Year of Entry
1	gvkey	Num	8	BEST12.	BEST32.	Company Identifier

2.0 - Variables

Variable	Label	Median	Mean	Range	Std Dev
RSST_ACC	RSST Accruals	0.013	-0.026	7.034	0.511
CH_REC	Change in Receivables	0.002	0.006	0.783	0.071
CH_INV	Change in Inventory	0.000	0.003	0.513	0.042
SOFT_ASSETS	% Soft Assets	0.516	0.496	0.985	0.278
CH_CS	% Change in Cash Sales	103.813	109.678	3049.745	161.742
CH_ROA	Change in Return on Assets	-0.001	-0.046	12.632	0.702

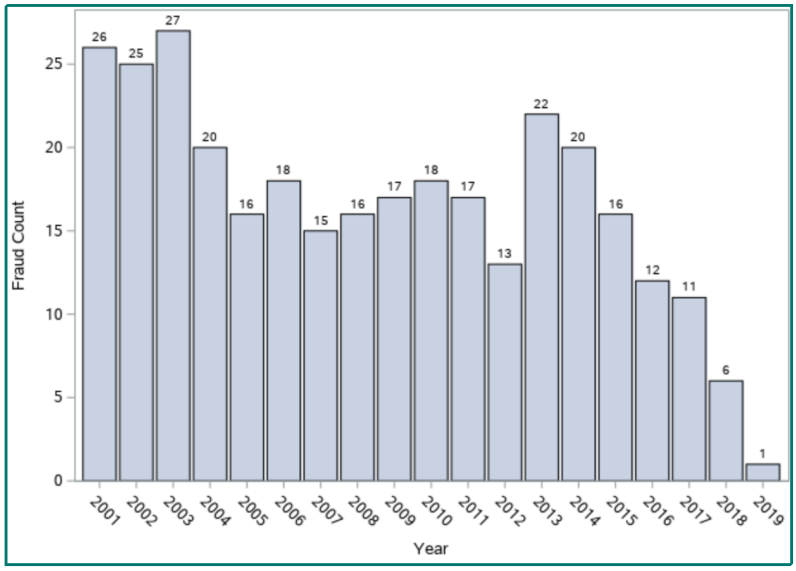
2.1 - Continuous Variables

Figure 2.1 reports basic statistics for our continuous variables. Note that CH_CS reports much larger numbers as it is a percentage. For CH_REC, CH_INV, SOFT_ASSETS and CH_CS, we see medians and means not too far apart. Their reported range, however, is rather high when considering their standard deviations. We suspect their data are rather central, but also somewhat spread out. RSST_ACC and CH_ROA appear to have spreads most affected by outliers. We see both means are much lower than their middle values, indicating negative outliers to some degree. Again, range figures much higher than standard deviations suggests spread out data. We suspect their data are positively skewed, and rather spread out around their means.

Fraud (Fraudulent = 1)	COUNT
0	20000
1	316

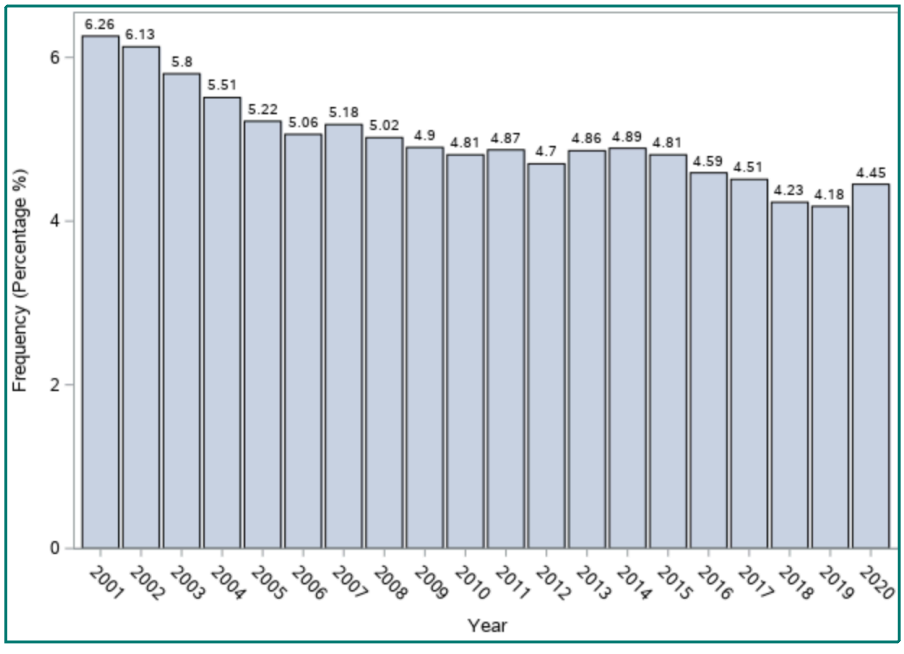
2.2 - Fraud Split

We look to the distribution of fraudulent activities in our sample. Figure 2.2 (left) reports our sample size is 20,316, of which 20,000 are non-fraudulent entries, and 316 are fraudulent.



2.3 - Fraud cases over time

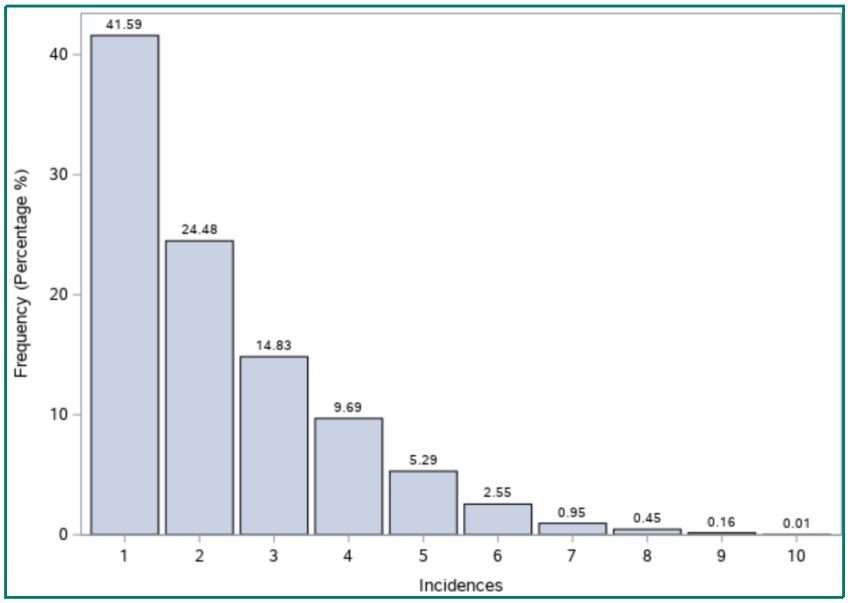
To the right we see a decreasing trend in fraud cases over time. This is an expected result, and most likely due to the issue becoming more prominent around the globe, with auditors constantly improving their detection methods. It could also, however, be due to the simple fact our sample contains more data in previous years, as seen in Figure 2.4 below. We can expect that, if we were to estimate our models on earlier data in our sample, it would produce more accurate results due to the fact there is not only more data present, but more fraudulent cases to estimate.



2.4 - FYEAR Frequency

UNIQUE_GVKEYS
8937

2.5 - Unique GVKEYs



2.6 - GVKEY Incidence

Figures 2.5 and 2.6 report statistics on our company identifier, GVKEY. We see there are 8,937 unique companies in our sample. Considering our sample size of 20,316, companies must, on average, have 1 or 2 entries each. Figure 2.6 confirms this. Over 40% of our sample is made up of single entry companies, and over 80% is made up of companies with fewer than 4 entries. Our models may produce inaccurate results due to the heavily unrelated data.

Frequency Col Pct	Table of ISSUE by FRAUD		
	FRAUD(Fraud (Fraudulent = 1))		
ISSUE(Issuance (Issued Securities = 1))	0	1	Total
0	2969 14.85	30 9.49	2999
1	17031 85.16	286 90.51	17317
Total	20000	316	20316

2.7 - ISSUE by FRAUD

Figure 2.7 illustrates the distribution of ISSUE in our sample. Looking at column percentage, we learn it is more common for a company to issue securities when they are fraudulent. This goes against what we know about fraud. If managers were under pressure to commit fraud and aimed to maintain or increase share price, issuing shares would dilute the market and cause share price to go the opposite way. The statistic could represent, however, managers trying to not act suspicious as a last minute resort. On the other hand, the difference is not that great, and could simply be due to our sample selection.

We want to examine if our explanatory variables, or model components, hold statistically different means between fraudulent and non-fraudulent cases. Student T-Tests will enable us to do this. We will perform a two-tailed T-Test for each explanatory variable, using FRAUD to create our test subsamples. After having run the tests, we can export the results and manipulate them in a way so that it's easy to comprehend and draw conclusions. Figure 2.8 is the resulting table:

Obs	Variable	Method	Variances	Probt	ProbF
1	SOFT_ASSETS	Satterthwaite	Unequal	<.0001	<.0001
2	RSST_ACC	Satterthwaite	Unequal	<.0001	<.0001
3	ISSUE	Satterthwaite	Unequal	0.0015	<.0001
4	CH_CS	Satterthwaite	Unequal	0.0811	<.0001
5	CH_REC	Satterthwaite	Unequal	0.1649	0.0015
6	CH_ROA	Satterthwaite	Unequal	0.3179	<.0001
7	CH_INV	Pooled	Equal	0.0985	0.4078

2.8 - T-Tests

We chose to evaluate our F-values at the conventional significance level of 5%. Therefore, tests that produced an F-value of less than 0.05 were granted the Satterthwaite method for unequal variances, and those equal to or above 0.05 were granted the Pooled method for equal variances. CH_INV is the only variable reporting an equal variance for its test. We therefore cannot reject the null hypothesis for this test, that is, that fraudulent and non-fraudulent cases have equal means of CH_INV. Of the remaining six tests, we look to the 'Probt' column to determine their ability to reject their null hypotheses. Only three hold a p-value significant at a 5% level, those being: SOFT_ASSETS, RSST_ACC, and ISSUE. We can say with confidence that, between fraudulent and non-fraudulent cases, the means of these explanatory variables are statistically and significantly different. In other words, we can reject the null hypothesis that the means of SOFT_ASSETS, RSST_ACC, and ISSUE are equal across companies practicing and not practicing fraud. We can suspect these three variables will provide the lowest P-values in our logistic regressions for the next section.

Logistic Regressions

This section involves: estimating a model using our previously created model components, calculating F-score, choosing an appropriate cut-off point using classification tables, and discussing the predictive power of our model. Below we see our initial model which will be estimated using the LOGISTIC procedure. Some initial results are also shown.

```
MODEL FRAUD (EVENT = "1") = RSST_ACC CH_REC CH_INV SOFT_ASSETS CH_CS CH_ROA ISSUE;
```

3.0 - The Model

Criterion	Intercept Only	Intercept and Covariates
AIC	3260.342	3138.301
SC	3268.261	3201.655
-2 Log L	3258.342	3122.301

3.1 - Model Fit

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	136.0405	7	<.0001
Score	125.4764	7	<.0001
Wald	118.6494	7	<.0001

3.2 - Testing Global H0

Figure 3.1 reports the model fit statistics. For each measure, we see that including the intercept and covariates in the model provides us with a lower statistic, hence, our intercept and variables, overall, have a positive impact on our ability to predict fraud. The right figure reports the global test of the null hypothesis that the estimated coefficients are all 0. We see extremely low p-values, leading us to reject that hypothesis, and say that our estimated coefficients are, as a whole, statistically different from 0. Below we see the individual estimated coefficients for the intercept and our explanatory variables.

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-6.0391	0.2484	591.1729	<.0001
RSST_ACC	1	0.3699	0.1380	7.1802	0.0074
CH_REC	1	-0.3250	0.7793	0.1739	0.6766
CH_INV	1	0.5401	1.2263	0.1940	0.6596
SOFT_ASSETS	1	2.4372	0.2401	103.0364	<.0001
CH_CS	1	0.000230	0.000366	0.3946	0.5299
CH_ROA	1	-0.0123	0.0888	0.0191	0.8901
ISSUE	1	0.5050	0.1942	6.7645	0.0093

3.3 - Parameter Estimates

As predicted, RSST_ACC, SOFT_ASSETS, and ISSUE are the only variables reporting a significant parameter estimate at the 5% confidence level. The intercept

is also significant in this way. We can conclude that only these 4 parameters have an influence in the models ability to predict fraud. We can compare our results to the estimates of Dechow et al. The table below highlights the key results.

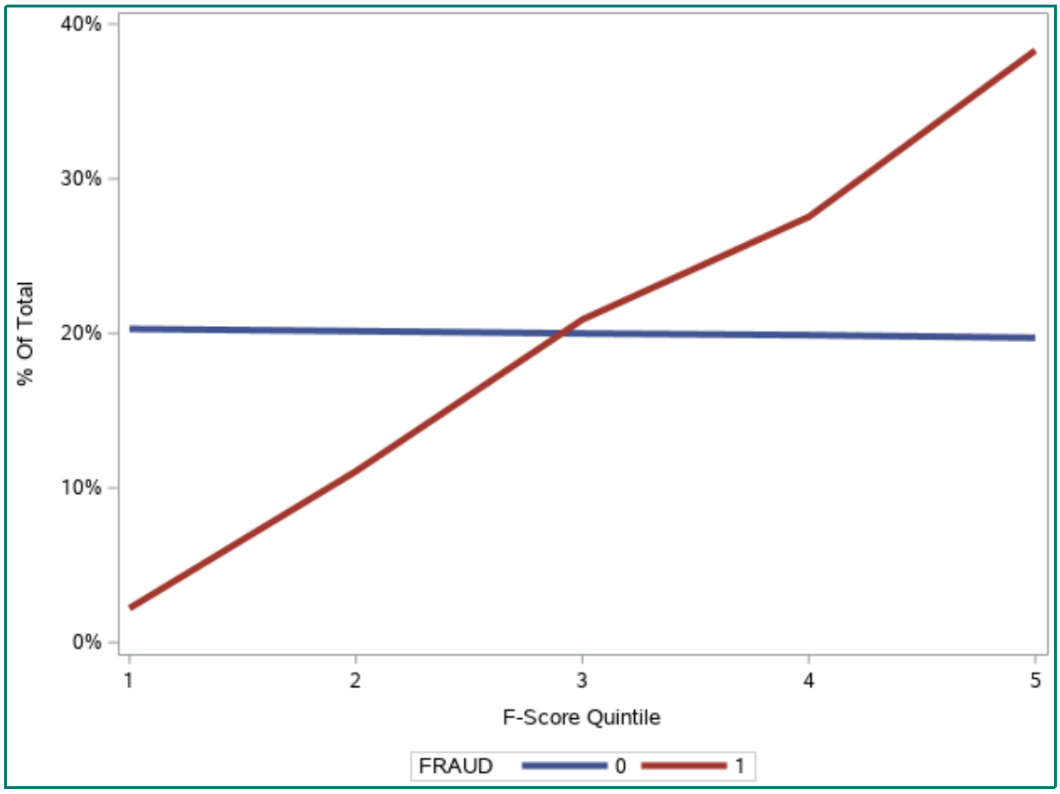
Obs	VARIABLE	ESTIMATE	WALDCHISQ	PROBCHISQ
1	Intercept	-6.03910	591.17	0.00000
2	Intercept*	-7.89300	1180.00	0.00100
3	RSST_ACC	0.36986	7.18	0.00737
4	RSST_ACC*	0.79000	24.10	0.00100
5	CH_REC	-0.32503	0.17	0.67663
6	CH_REC*	2.51800	28.50	0.00100
7	CH_INV	0.54005	0.19	0.65965
8	CH_INV*	1.19100	4.40	0.01900
9	SOFT_ASSETS	2.43724	103.04	0.00000
10	SOFT_ASSETS*	1.97900	86.30	0.00100
11	CH_CS	0.00023	0.39	0.52989
12	CH_CS*	0.17100	17.00	0.00100
13	CH_ROA	-0.01228	0.02	0.89006
14	CH_ROA*	-0.93200	19.90	0.00100
15	ISSUE	0.50502	6.76	0.00930
16	ISSUE*	1.02900	28.50	0.00100

3.4 - Comparing Estimate Results

We see for each explanatory variable: a parameter estimate, a Wald Chi-square, and a p-value. Those belonging to Dechow et al. have an asterisk (*). We see for the most part, the two sets of estimates are similar. The p-values, however, are all below a 5% level in the new set of estimates, so each variable is statistically significant to the model. We notice a large difference in the reported estimates of CH_REC, where one relationship with fraud is negative, and one is largely positive. All other estimate pairs seem to be in the same nature of association with fraud, in that they are either both positive or both negative. We see for all variables except SOFT_ASSETS, the new parameter estimates are higher, indicating they have a stronger relationship with fraud, and hence a larger influence in predicting a case of it. Wald Chi-square values follow a similar pattern, where they are larger for the new set of estimates in every pair of variables, except for SOFT_ASSETS, indicating a larger significance in each, compared to our model.

We can further calculate F-score for our sample. F-score captures the probability of fraud relative to the sample average. Once having calculated F-score for our sample, we will divide our sample into quintiles, or five groups, based on their individual scores.

Figure 3.5 first shows us the changes in percentage of data across the quintiles. Know that the 1st quintile holds observations with lower F-scores, and the 5th holds higher F-scores.

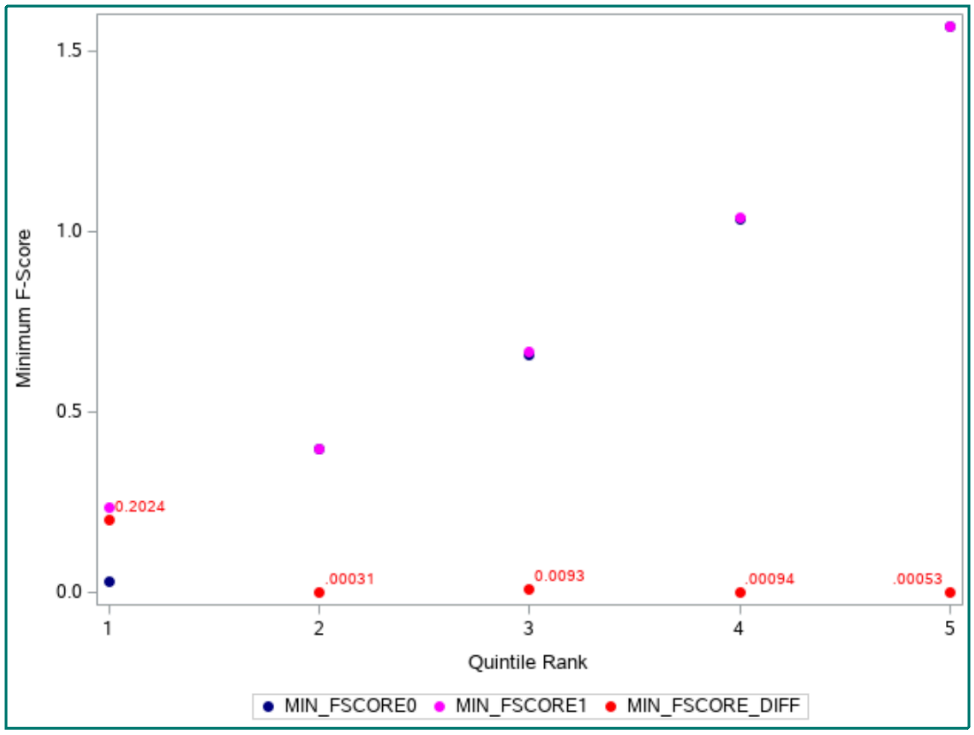


3.5 - % of Total per Quintile

For non-fraudulent cases, there remains roughly 20% of observations in each quintile. Their data is evenly spread. For fraudulent cases, we see a constant increase in cases across the quintiles, so as F-Score increases, so does the frequency of companies practicing fraud. This tells us F-score is somewhat indicative of fraud in our sample. Figure 3.6 below shows the data behind the above plot, with the addition of a minimum F-score for each quintile. It would be easier to note the differences in minimum F-score, however, if we were to create a scatter plot. The next figure (3.7) does that for us.

Obs	RANK	FRAUD	N	MIN_FSCORE	PCT_OF_TOTAL
1	1	0	4056	0.03215	20.28%
2	1	1	7	0.23451	2.22%
3	2	0	4028	0.39760	20.14%
4	2	1	35	0.39790	11.08%
5	3	0	3998	0.65815	19.99%
6	3	1	66	0.66740	20.89%
7	4	0	3976	1.03693	19.88%
8	4	1	87	1.03786	27.53%
9	5	0	3942	1.56864	19.71%
10	5	1	121	1.56917	38.29%

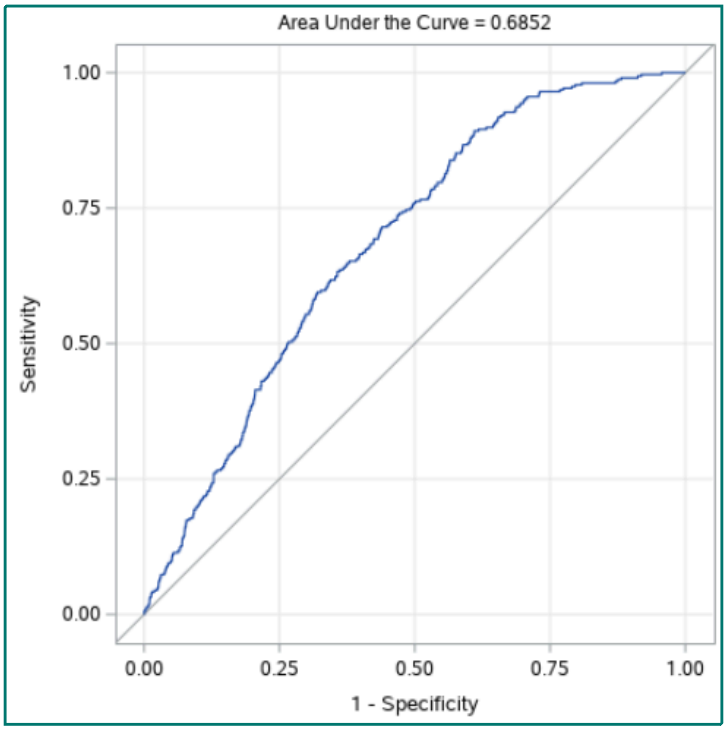
3.6 - F-score Quintiles



3.7 - Minimum F-Score across Quintiles

The navy and magenta points illustrate the minimum F-score of fraudulent companies and non-fraudulent companies in our sample, where both follow almost an identical increasing trend. The circle points illustrate the difference in minimum F-score. We see the largest difference in minimum F-score is reported as 0.2024 in the first quintile, or where F-scores are at their lowest. For the next 4 quintiles, minimum F-scores are essentially the same across both groups, although the difference still remains positive, in that the minimum F-score is still higher in fraudulent cases for the entirety of our sample, only by not much. It is clear that there is not a huge difference in minimum F-score between these two groups.

We now must decide on an appropriate cut-off point to use for our logistic model, so as to classify a prediction as fraudulent or not. Re-running our model, we output two new figures: an ROC curve, and a classification table. The ROC curve for our model is shown in Figure 3.8 below. With an Area Under the Curve (or AUC) of 0.6852, we just fall short of the conventionally required AUC of 0.7 in order to conclude our model has some sort of predicting power. Nonetheless, we look to our classification table.



3.8 - ROC Curve (AUC = 0.6852)

We notice first that when the probability level hits 7.5%, our model is already at 100% specificity. The models predictive power is clearly poor. For predicting fraud, we want to prioritise sensitivity, meaning we want to ensure we are predicting all fraudulent cases as positive. Predicting a negative case as positive is not as much of a worry. 0.015, or 1.5% is the last probability level at which sensitivity is higher than specificity. For this reason, we will choose 1.5% as our cut-off in classifying fraud. This results in an accuracy of 57.1% for our sample, where 217 cases of fraud and 11,392 cases of non-fraud are correctly predicted.

	Correct		Incorrect		Percentages				
Prob Level	Event	Non-Event	Event	Non-Event	Correct	Sensi-tivity	Speci-ficity	Pos Pred	Neg Pred
0.000	316	0	20000	0	1.6	100.0	0.0	1.6	.
0.005	311	2408	17592	5	13.4	98.4	12.0	1.7	99.8
0.010	275	7894	12106	41	40.2	87.0	39.5	2.2	99.5
0.015	217	11392	8608	99	57.1	68.7	57.0	2.5	99.1
0.020	166	14170	5830	150	70.6	52.5	70.9	2.8	99.0
0.025	103	16286	3714	213	80.7	32.6	81.4	2.7	98.7
0.030	64	17877	2123	252	88.3	20.3	89.4	2.9	98.6
0.035	32	18926	1074	284	93.3	10.1	94.6	2.9	98.5
0.040	13	19564	436	303	96.4	4.1	97.8	2.9	98.5
0.045	5	19847	153	311	97.7	1.6	99.2	3.2	98.5
0.050	3	19912	88	313	98.0	0.9	99.6	3.3	98.5
0.055	2	19942	58	314	98.2	0.6	99.7	3.3	98.4
0.060	1	19956	44	315	98.2	0.3	99.8	2.2	98.4
0.065	1	19967	33	315	98.3	0.3	99.8	2.9	98.4
0.070	0	19983	17	316	98.4	0.0	99.9	0.0	98.4
0.075	0	19992	8	316	98.4	0.0	100.0	0.0	98.4
0.080	0	19994	6	316	98.4	0.0	100.0	0.0	98.4
0.085	0	19997	3	316	98.4	0.0	100.0	0.0	98.4
0.090	0	19999	1	316	98.4	0.0	100.0	0.0	98.4
0.095	0	19999	1	316	98.4	0.0	100.0	0.0	98.4
0.100	0	20000	0	316	98.4	0.0	100.0	.	98.4

3.9 - Classification Table

Model Selection & Backtesting

In this section, stepwise regression will first select the most influential and significant explanatory variables for predicting fraud. The final model will be used in 100 regression estimations, where, each time, our sample will be split into estimation and test subsamples based on random entry years. Using the coefficients estimated from our estimation sample, we will determine F-score on out-of-sample data (our test data).

Throughout backtesting, we are looking for proof to support our hypothesis that, if F-score is greater than 1, fraud is present. Below we see the results of our stepwise regression. As expected, the three variables we previously found to be statistically significant were kept in the final model.

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-6.0172	0.2452	602.3205	<.0001
RSST_ACC	1	0.3723	0.1326	7.8855	0.0050
SOFT_ASSETS	1	2.4353	0.2389	103.8779	<.0001
ISSUE	1	0.5115	0.1936	6.9798	0.0082

4.0 - Stepwise Results

The process for a single estimate and test pair begins with randomly splitting the sample. 10 randomly selected years of data will be estimation data, and the remaining years will be test data. The regression is run on the estimation sample, where the coefficients are outputted to a new data set to then be merged with the test sample. With the test sample holding the coefficients, we can calculate for each observation: a predicted value, a probability value, and an F-score. A MEANS procedure finally produces statistics on the test sample's F-scores, appending them to a data set to complete the backtesting process.

Figure 4.1 (below) reports the mean and median figures of our backtesting results, where we notice mean and median are quite similar for each statistic, indicating little variation in backtesting results. We see that out of our 100 estimation and test pairs, the average mean of F-score for fraudulent cases was 1.38, and the average middle value was 1.31.

FRAUD	N Obs	Variable	Mean	Median
0	100	MEAN	0.9828698	0.9664434
		MIN	0.0435420	0.0373004
		Q1	0.4465689	0.4339012
		MEDIAN	0.8177291	0.8010349
		Q3	1.3862679	1.3694497
		MAX	5.0224728	5.0485896
1	100	MEAN	1.3811920	1.3469815
		MIN	0.2567296	0.2138033
		Q1	0.8562460	0.8085111
		MEDIAN	1.3162600	1.2792480
		Q3	1.7858288	1.7237388
		MAX	4.2024525	4.1657240

4.1 - Backtesting Results

For non-fraudulent cases, the average mean of F-score was only just less than 1 (0.98). We would hope this to be much lower than 1 so as to confidently say an F-score of 1 is indicative of a fraudulent case, however this is not the case. The average median F-score for no fraud was 0.81, significantly lower than fraud's 1.31. The same is the case for minimum values within the results, however not for the maximums. The highest F-scores were in fact reported by non-fraudulent cases. This completely goes against our hypothesis. We would say that this is due to outliers, as for 75% of results, fraudulent cases reported higher F-score, but the average maximum F-score is an average of 100 test results, and so it is most likely due to the poor predictive power of our model.

Concerning our hypothesis, we figure there are contradicting findings in the backtesting results, and for that reason we cannot confidently conclude that an F-score of greater than 1 suggests a company is practicing fraud. We instead say an F-score greater than 1 provides reasonable evidence towards a fraudulent case, but no conclusions can be made.

Big Data Application

For our big data application, we will be focusing on Apple Inc. Apple is constantly collecting data about its consumers; from device analytics, location services, and personal iCloud details, the tech giant has the ever-growing ability to predict a multitude of things. One way in which Apple could use their data to provide new value to their operations as a business is by predicting the physical storage needed for iCloud users to store their data on. Apple has 11 data centres around the globe; 6 in US, 2 in Denmark, and 3 in China (Wikipedia 2022). One way to be prepared for the ever-growing consumer need of cloud storage is to predict whether they must expand their physical storage locations in the next 5 years. The following model is proposed for Apple:

```
NEW = c1 + CTR*c2 + USERS*c3 + STORE*c4 + STORED*c5 + CH_SH*c6 + PCT_VID*c7
```

5.0 - Model for predicting the need of iCloud Storage

The variables are defined as follows:

```
CTR = 'No. of Data Centres'
STORE = 'Average GB Available per User'
STORED = 'Average GB Used per User'
CH_SH = 'Change in Share Price'
CH_USER = 'Change in No. of iCloud Users'
PCT_VID = 'Average % of Video Media'
```

5.1 - Variable Definitions

CTR was chosen as it provides a baseline for the model. The model should know how many data centres are currently in use by Apple. It goes without saying it is a good indicator of whether another storage location is needed. This variable is easy to measure.

STORE was included in the model as it provides the amount of storage iCloud users can use. In other words, STORE is the storage that Apple should have available per user. This variable can easily be measured, given Apple has the total gigabytes available per user. Not many iCloud users occupy their full range of storage however, so the next variable was made to identify the actual use of storage.

STORED measures the storage *actually used* by iCloud users. This offsets the previous variable STORE, and provides the model with an indication of how much storage iCloud users really make use of. The calculation of STORES should be no effort, given Apple has the total gigabytes of data each user utilises.

CH_SH was included in the model as an overall profitability measure of the company. Share price is generally an indicator to the market of how profitable a company is. In Apple's case, they can use their change in share price as an expectation of future sales, and therefore future iCloud user growth or decline. This variable, like all others, is an easy one to measure.

CH_USER was added to our model as a benchmark for iCloud user growth. It will inform the model of the change in iCloud users, and hence the storage capacity needed by Apple. This variable is simple to measure.

The last variable in our model is PCT_VID. As video files will always be the largest files stored in iCloud, the model should grasp the average percentage of each users makeup of video files. Apple will have this data available, and so it is easy to measure.

The way of estimating this model will be logistic regression. The dependent variable is either a yes or no, that is, Apple needs to predict whether or not they need to obtain a new data storage location within the next 5 years.

References

Dechow et al. (2011) Predicting Material Accounting Misstatements, *Contemporary Accounting Research,* Vol. 28 No. 1 pp. 17–82, accessed 14 June 2022.

Wikipedia (2022) *iCloud*, Wikipedia, accessed 19 June 2022. https://en.wikipedia.org/wiki/ICloud