INDR 450/550

Spring 2022

Lecture 13: Classification Problems

March 28, 2022

Fikri Karaesmen

Announcements

- Class Exercise at the end of lecture today. If you are participating online, please upload your document under Course Contents/Class Exercises
- HW 2 available with a deadline of April 4 (Labs 3 and 4).
- Exam scheduled.
- The first four labs were uploaded. Please follow them.
  - Current HW based on lab2 and lab3

Regression for Time Series: Classification Problems- Logistic Regression

- We'll then try to predict the probability of an event $P(Y=1|X)$. We use the shortcut:

$$p(X) = P(Y = 1|X)$$

and we use a logistic function to express this probability:

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

- After some algebraic manipulation, we can write:

$$\frac{p(X)}{1 - p(X)} = e^{\beta_0+\beta_1 X}$$

Regression for Time Series: Classification Problems- Logistic Regression

- Taking logs on both sides we get:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \underset{\checkmark}{\beta_0} + \underset{\checkmark}{\beta_1} X$$

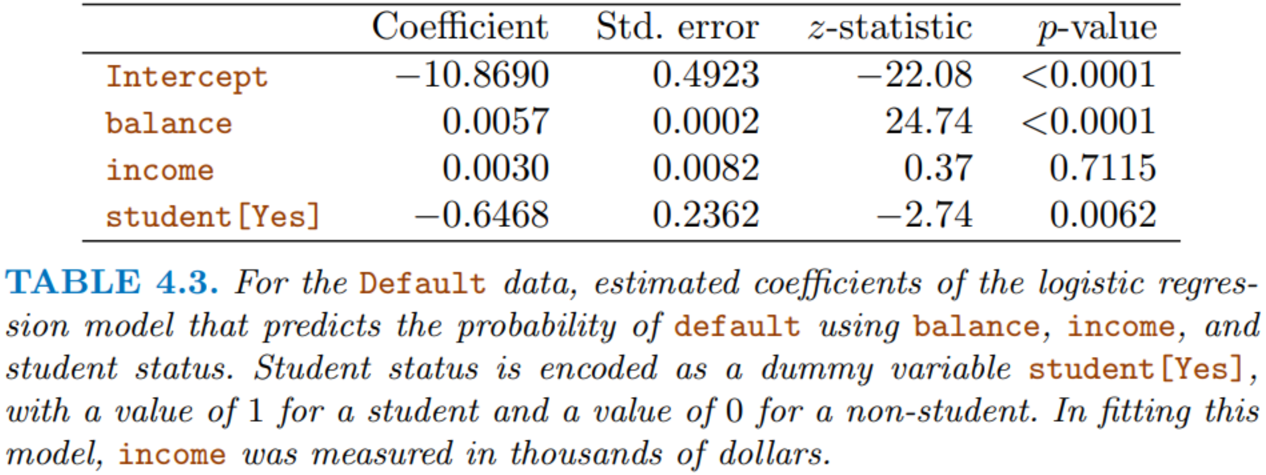
- The left hand side is known as the log-odds ratio also known as logit. The logistic regression model has a logit that is linear in the predictor X.

Regression for Time Series: Logistic Regression

- We can extend the model directly to p predictors X_1, X_2, ,,,, X_p. Let $\mathbf{X} = (X_1, X_2, ..., X_n)$:

$$p(\mathbf{X}) = \frac{e^{\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_p X_p}}$$

- And use maximum likelihood estimation to find $\hat{\beta}_0$ and $\hat{\beta}_1$, $\ldots\hat{\beta}_p$.



	Coefficient	Std. error	z-statistic	p-value
Intercept	−10.8690	0.4923	−22.08	<0.0001
balance	0.0057	0.0002	24.74	<0.0001
income	0.0030	0.0082	0.37	0.7115
student[Yes]	−0.6468	0.2362	−2.74	0.0062

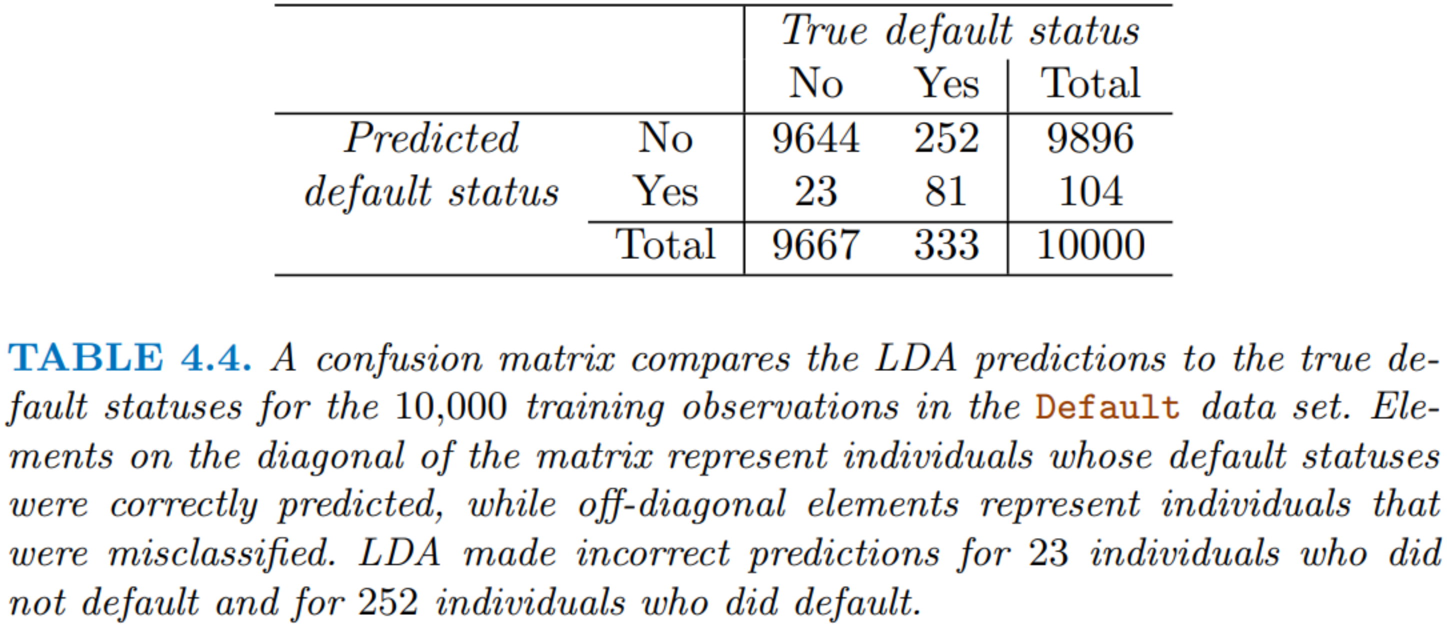
TABLE 4.3. *For the* Default *data, estimated coefficients of the logistic regression model that predicts the probability of* default *using* balance, income, *and student status. Student status is encoded as a dummy variable* student[Yes], *with a value of 1 for a student and a value of 0 for a non-student. In fitting this model,* income *was measured in thousands of dollars.*

Regression for Time Series: Logistic Regression – Classification Errors

- At the end of the day, we are usually asked to classify the data (label it 0 or 1). The probabilities must be converted to 0’s and 1’s.
- This does not appear to be hard to do. If $p(X)>1/2$, we convert it to 1 and otherwise we convert it to 0.
    - But we don’t have take a threshold of 1/2. Depending on the decision we might prefer 9/10 or ¼ or anything else.
- There are two types of classification errors
    - An individual who defaults is incorrectly classified as ‘non-default’.
    - An individual who does not default is incorrectyl classified as ‘default’
- For decision making purposes, it’s important to determine the rates of both errors.

Regression for Time Series: Logistic Regression – Classification Errors

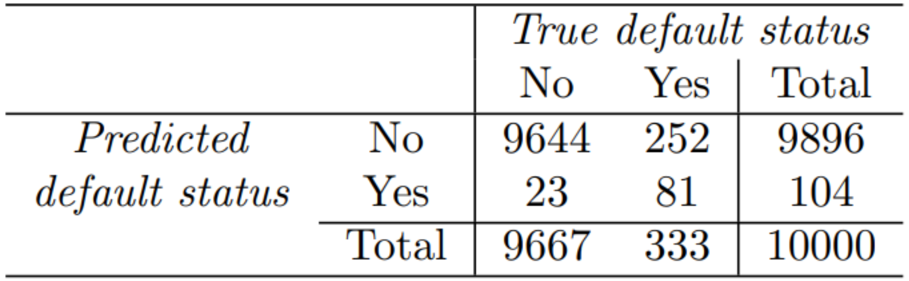
- The errors can be specified in a ‘confusion’ matrix.



		True default status		
		No	Yes	Total
Predicted default status	No	9644	252	9896
	Yes	23	81	104
	Total	9667	333	10000

TABLE 4.4. *A confusion matrix compares the LDA predictions to the true default statuses for the* 10,000 *training observations in the* `Default` *data set. Elements on the diagonal of the matrix represent individuals whose default statuses were correctly predicted, while off-diagonal elements represent individuals that were misclassified. LDA made incorrect predictions for* 23 *individuals who did not default and for* 252 *individuals who did default.*

Regression for Time Series: Classification Errors



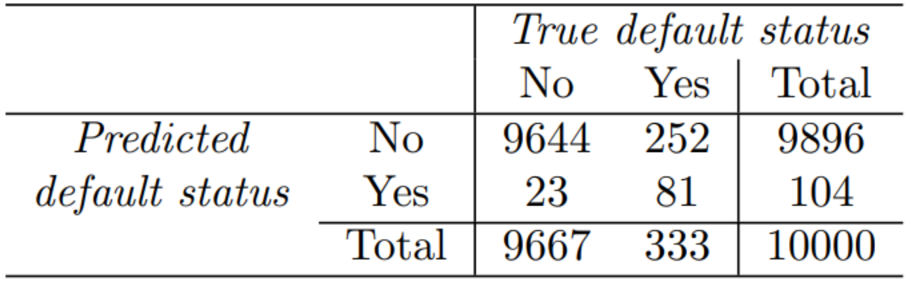
		True default status		
		No	Yes	Total
Predicted default status	No	9644	252	9896
	Yes	23	81	104
	Total	9667	333	10000

- Sensitivity: Percentage of True Defaulters who are identified = 81/333
- Specificity: Percentage of non-defaulters correctly identified = 9644/9667

Regression for Time Series: Classification Errors

- Recall the simple assignment rule: If $p(X)>1/2$, we classify it as 1 and otherwise we classify it to 0.
- By reducing the assignment threshold from ½, we would be able to 'catch' more defaulters (i.e improve sensitivity) but this would be at the expense of specificity.
- This is therefore a tradeoff without a simple 'optimal' level and optimization depends on the context.
- If the governor of Istanbul is trying to predict a snow storm, it makes more sense to take a low threshold.
- If a medical test is for a serious issue, it also makes sense to choose a low threshold.

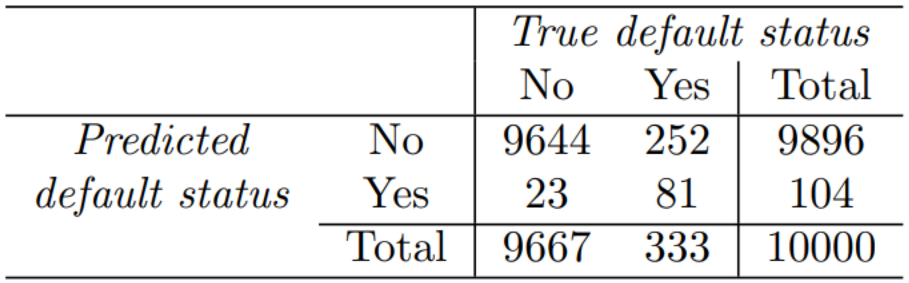
Regression for Time Series: Classification Errors



		True default status		
		No	Yes	Total
Predicted default status	No	9644	252	9896
	Yes	23	81	104
	Total	9667	333	10000

- Note that one reason making predictions with the above data is that non-defaulters are relatively rare (333/10000).
- We can already make trivial predictions by labeling all customers as ‘non-defaulters’
  - Sometimes called the null model

Regression for Time Series: Classification Errors



		True default status		
		No	Yes	Total
Predicted	No	9644	252	9896
default status	Yes	23	81	104
	Total	9667	333	10000

Label everyone as a non-defaulter

	No	**Yes**	**Total**
No	9667	333	10000
Yes	0	0	0
Total	9667	333	1000

- If everyone is labeled ‘non-defaulter’ (i.e. we pick a probability threshold of 1)
- We have 100% specificity (all non-defaulters are automatically classified correctly)
- But 0% sensitivity (all defaulters are classified as non-defaulters)
- The total error rate is 333/1000, only 3%!

Regression for Time Series: Classification Errors

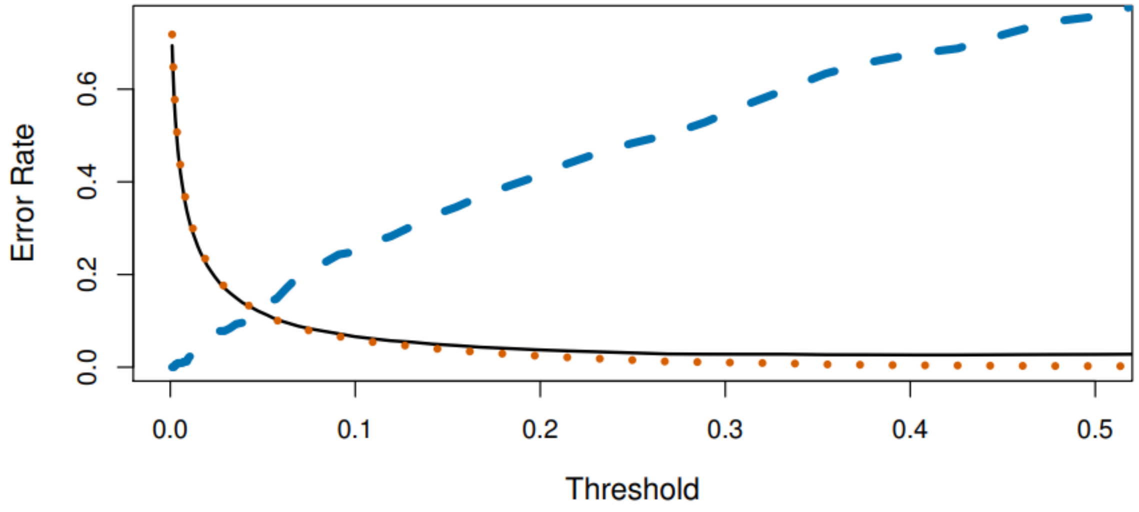
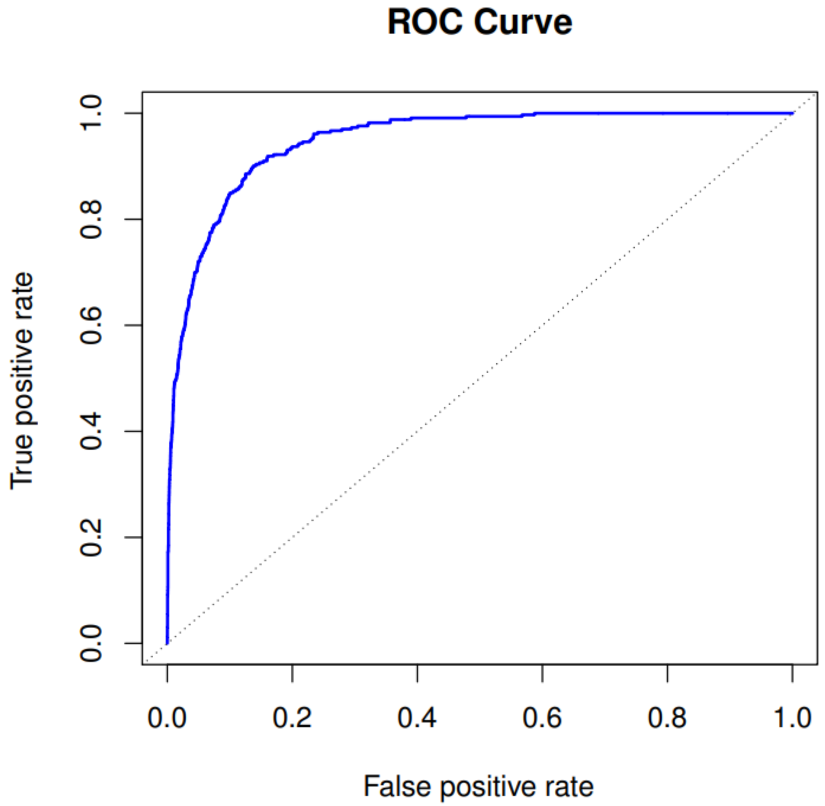


FIGURE 4.7. *For the* `Default` *data set, error rates are shown as a function of the threshold value for the posterior probability that is used to perform the assignment. The black solid line displays the overall error rate. The blue dashed line represents the fraction of defaulting customers that are incorrectly classified, and the orange dotted line indicates the fraction of errors among the non-defaulting customers.*

- The total error rate is decreasing in the threshold in the given range
- But the misclassification of default customers is increasing.

Regression for Time Series: Classification Errors

- This information is better summarized by a Receiver Operating Characteristics (ROC curve).



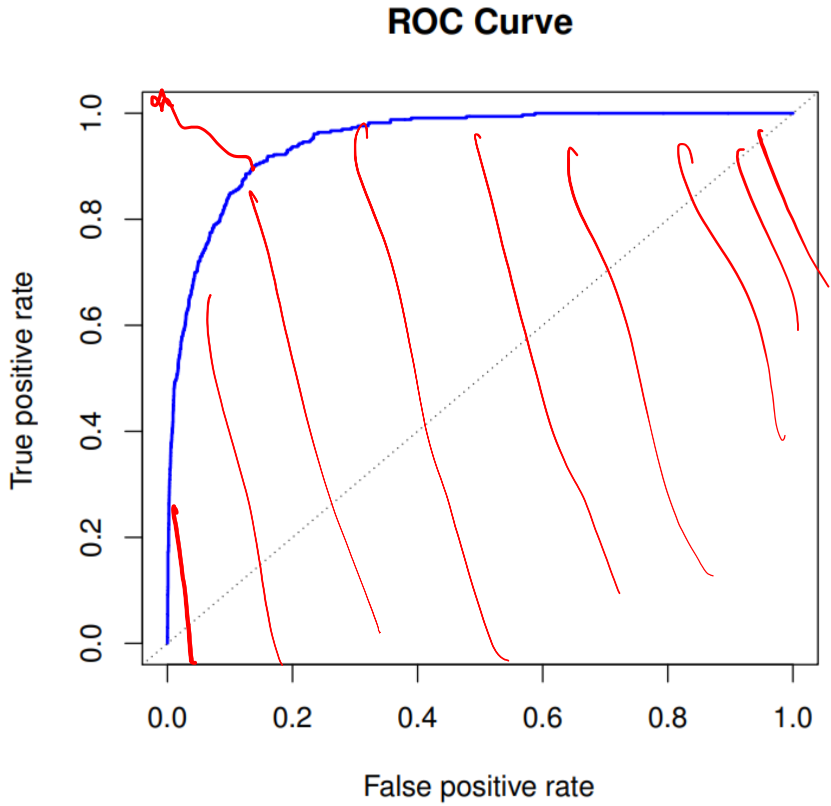
(sensitivity)

(1-specificity)

We change the probability threshold and plot a curve of false positive Rates vs. True positive rates for each threshold.

Regression for Time Series: Classification Errors

- The overall performance of a classifier can be measured by the Area Under the ROC curve (AUC).



(sensitivity)

(1-specificity)

Note that a perfect classifier would have the blue curve passing through the upper left corner and would have AUC=1.
All imperfect classifiers have AUC’s less than 1
But the bigger AUC, the better the overall performance of the classifier.

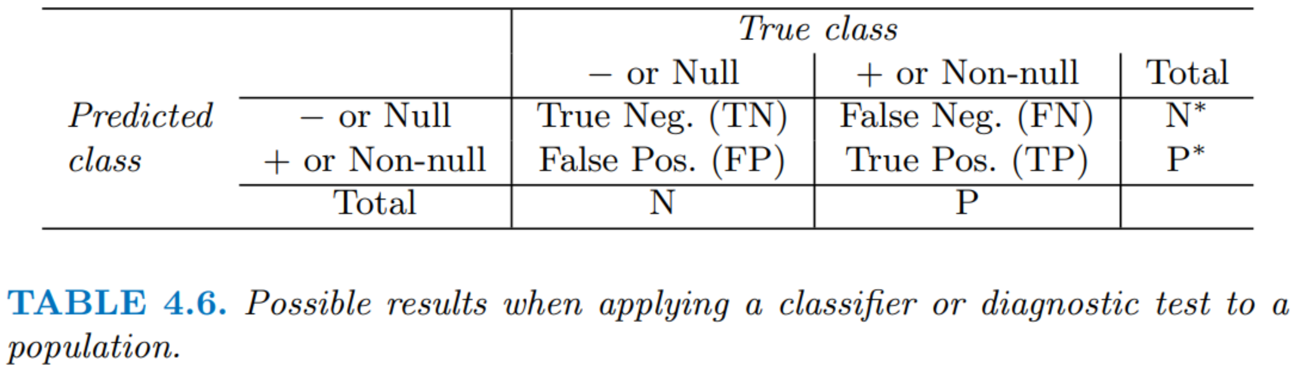
The AUC for the credit card default data is 0.95 (very good).

Regression for Time Series: Classification Errors

- The AUC is a useful criterion to compare different classifier models.
- A random classifier (toss a coin to decide the classification) is expected to have an AUC lower than ½.
- These give us useful benchmarks (1/2 at the low end and 1 at the high end).

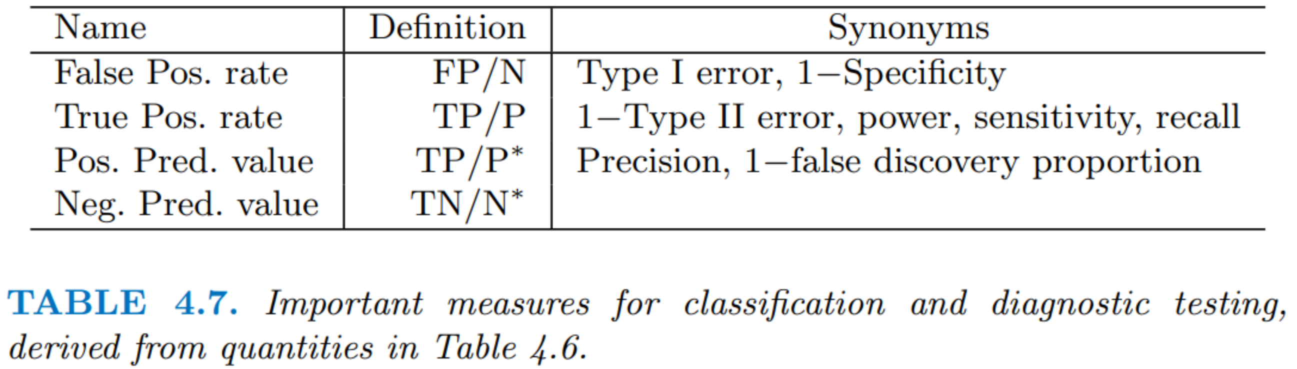
Regression for Time Series: Classification Errors

- Unfortunately, different disciplines use different names for classification errors. This is also true for software.



		True class		
		− or Null	+ or Non-null	Total
Predicted class	− or Null	True Neg. (TN)	False Neg. (FN)	N^*
	+ or Non-null	False Pos. (FP)	True Pos. (TP)	P^*
	Total	N	P	

TABLE 4.6. *Possible results when applying a classifier or diagnostic test to a population.*



Name	Definition	Synonyms
False Pos. rate	FP/N	Type I error, 1−Specificity
True Pos. rate	TP/P	1−Type II error, power, sensitivity, recall
Pos. Pred. value	TP/P^*	Precision, 1−false discovery proportion
Neg. Pred. value	TN/N^*	

TABLE 4.7. *Important measures for classification and diagnostic testing, derived from quantities in Table 4.6.*

From ISL (James et al.)

Classification Example:

- Inspired by the stock market prediction example in ‘Introduction to Statistical Learning’, let us try to predict whether the sales for BMW cars in Turkey increase or decrease with respect to previous month.

Month	Sales	Up
1	82	
2	152	1
3	377	1
4	366	0
5	281	0
6	239	0
7	352	1
8	279	0
9	295	1
10	384	1

Classification Example:

- Let us use the lagged monthly differences as predictors:
- $\Delta_{t-1}=y_{t-1}-y_{t-2}$, $\Delta_{t-2}=y_{t-2}-y_{t-3}$, $\Delta_{t-3}=y_{t-3}-y_{t-4}$ and $D_{t-1} = y_{t-1}-y_{t-13}$

Month	Sales	Up
1	82	
2	152	1
3	377	1
4	366	0
5	281	0
6	239	0
7	352	1
8	279	0
9	295	1
10	384	1

Classification Example:

- Let us use the lagged monthly differences and a seasonal difference as predictors:
- $\Delta_{t\text{-}1}=y_{t\text{-}1}\text{-}y_{t\text{-}2}$, $\Delta_{t\text{-}2}=y_{t\text{-}2}\text{-}y_{t\text{-}3}$, $\Delta_{t\text{-}3}=y_{t\text{-}3}\text{-}y_{t\text{-}4}$ and $D_{t\text{-}1} = y_{t\text{-}1}\text{-}y_{t\text{-}13}$
- We then fit:

$$p(\mathbf{X}) = \frac{e^{\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_p X_p}}$$

- where $X_1 = \Delta_{t\text{-}1}$, $X_2 = \Delta_{t\text{-}2}$, $X_3 = \Delta_{t\text{-}3}$

Classification Example:

- Let us use the lagged monthly differences and a seasonal difference as predictors:
- $\Delta_{t\text{-}1} = y_{t\text{-}1}\text{-}y_{t\text{-}2}$, $\Delta_{t\text{-}2} = y_{t\text{-}2}\text{-}y_{t\text{-}3}$, $\Delta_{t\text{-}3} = y_{t\text{-}3}\text{-}y_{t\text{-}4}$ and $D_{t\text{-}1} = y_{t\text{-}1}\text{-}y_{t\text{-}13}$
- We then fit:

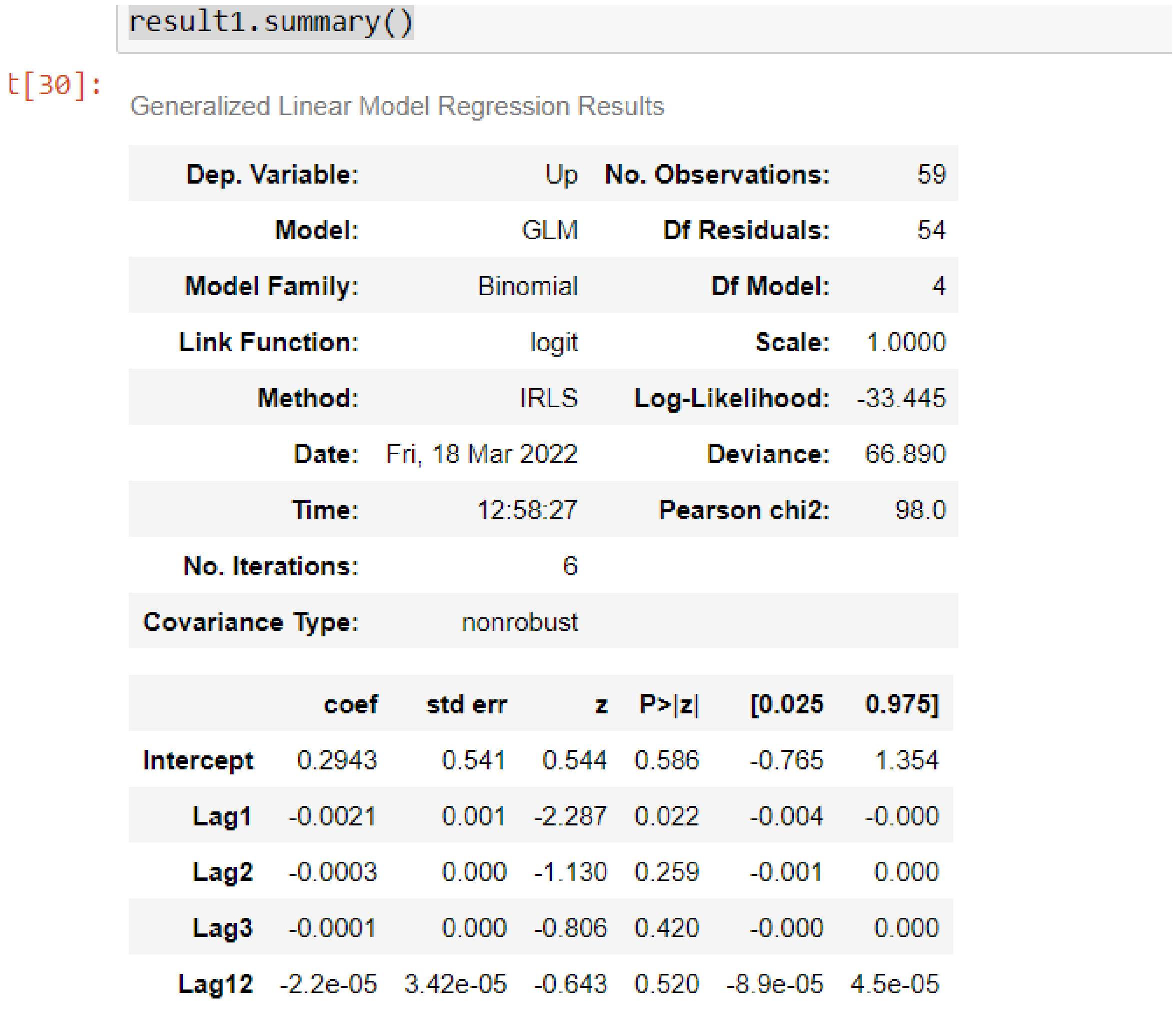
$$p(\mathbf{X}) = \frac{e^{\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_p X_p}}$$

- where $X_1 = \Delta_{t\text{-}1}$, $X_2 = \Delta_{t\text{-}2}$, $X_3 = \Delta_{t\text{-}3,}$ $X_4 = D_{t\text{-}1.}$

Classification Example: data

Month	Sales	Up	Lag1	Lag2	Lag3	Lag12
1	82					
2	152	1				
3	377	1	70			
4	366	0	225	70		
5	281	0	-11	155	70	
6	239	0	-85	-236	85	
7	352	1	-42	-74	-391	
8	279	0	113	43	162	
9	295	1	-73	155	117	
10	384	1	16	-186	112	
11	439	1	89	89	-341	
12	1912	1	55	73	275	
13	105	0	1473	-34	-16	
14	131	1	-1807	1418	-107	-82
15	159	1	26	-3280	1452	-234

Classification Example: results



```
result1.summary()
```

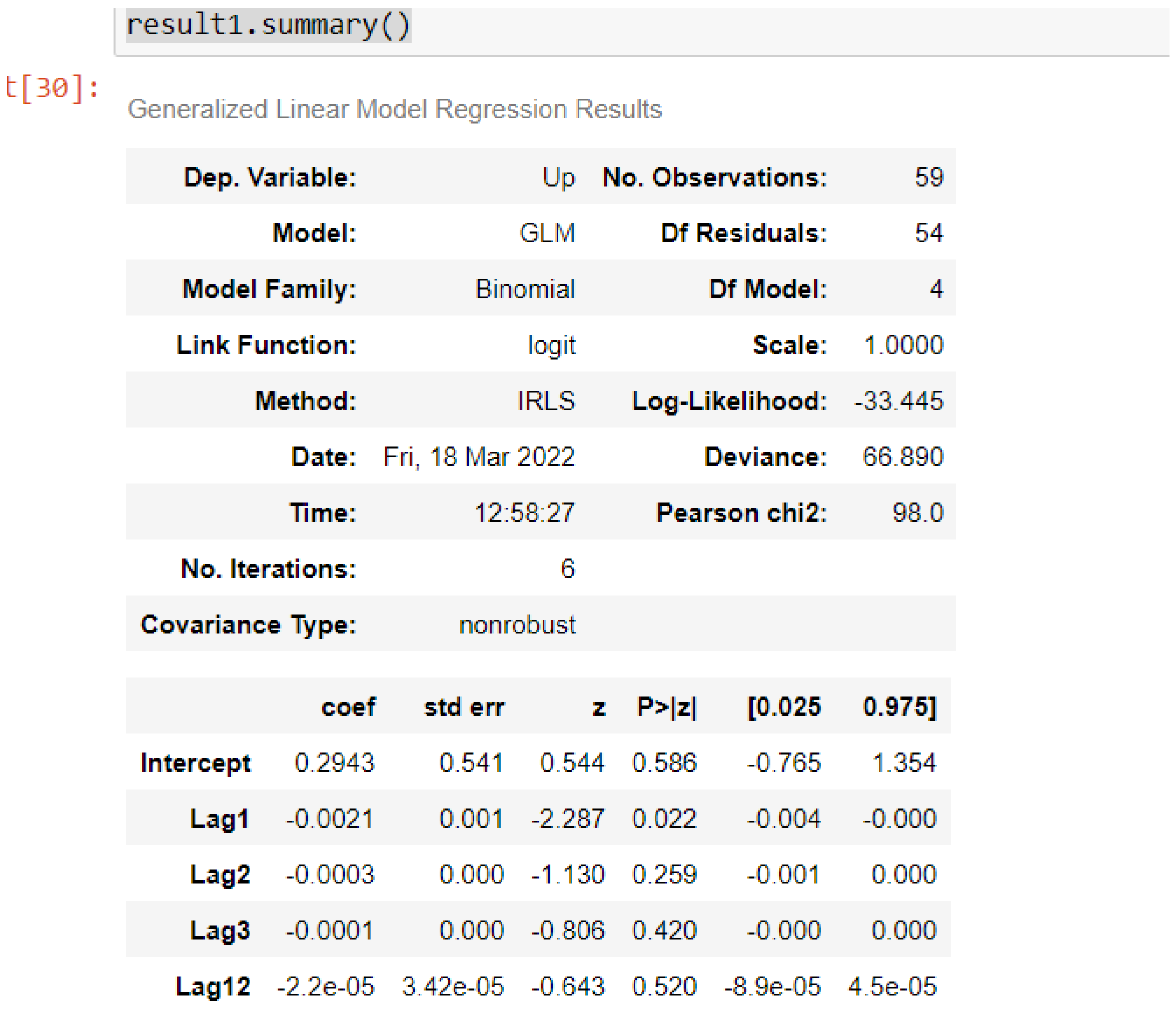
t[30]:

Generalized Linear Model Regression Results

Dep. Variable:	Up	**No. Observations:**	59
Model:	GLM	**Df Residuals:**	54
Model Family:	Binomial	**Df Model:**	4
Link Function:	logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-33.445
Date:	Fri, 18 Mar 2022	**Deviance:**	66.890
Time:	12:58:27	**Pearson chi2:**	98.0
No. Iterations:	6		
Covariance Type:	nonrobust		

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
Intercept	0.2943	0.541	0.544	0.586	-0.765	1.354
Lag1	-0.0021	0.001	-2.287	0.022	-0.004	-0.000
Lag2	-0.0003	0.000	-1.130	0.259	-0.001	0.000
Lag3	-0.0001	0.000	-0.806	0.420	-0.000	0.000
Lag12	-2.2e-05	3.42e-05	-0.643	0.520	-8.9e-05	4.5e-05

Classification Example: results



```
result1.summary()
```

t[30]:

Generalized Linear Model Regression Results

Dep. Variable:	Up	**No. Observations:**	59
Model:	GLM	**Df Residuals:**	54
Model Family:	Binomial	**Df Model:**	4
Link Function:	logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-33.445
Date:	Fri, 18 Mar 2022	**Deviance:**	66.890
Time:	12:58:27	**Pearson chi2:**	98.0
No. Iterations:	6		
Covariance Type:	nonrobust		

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
Intercept	0.2943	0.541	0.544	0.586	-0.765	1.354
Lag1	-0.0021	0.001	-2.287	0.022	-0.004	-0.000
Lag2	-0.0003	0.000	-1.130	0.259	-0.001	0.000
Lag3	-0.0001	0.000	-0.806	0.420	-0.000	0.000
Lag12	-2.2e-05	3.42e-05	-0.643	0.520	-8.9e-05	4.5e-05

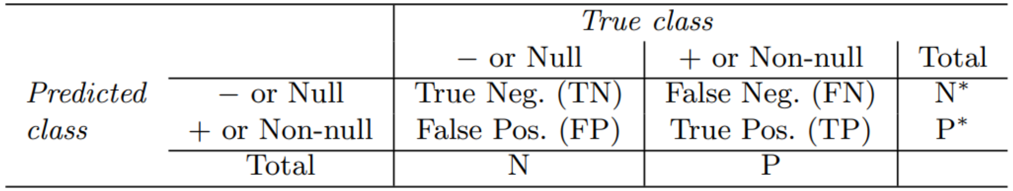
Let's try a probability threshold for ½.
The confusion matrix is then:

Sklearn: ISL book:

$$\begin{bmatrix}11 & 12\\ 4 & 32\end{bmatrix} \longrightarrow \begin{bmatrix}11 & 4\\ 12 & 32\end{bmatrix}$$

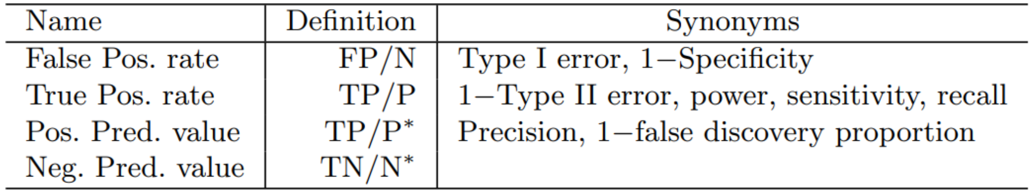
An overall error rate of 16/59.
Sensitivity of 32/36
Specificity of 12/23

Classification Example: results



		True class		
		− or Null	+ or Non-null	Total
Predicted class	− or Null	True Neg. (TN)	False Neg. (FN)	N^*
	+ or Non-null	False Pos. (FP)	True Pos. (TP)	P^*
	Total	N	P	

TABLE 4.6. *Possible results when applying a classifier or diagnostic test to a population.*



Name	Definition	Synonyms
False Pos. rate	FP/N	Type I error, 1−Specificity
True Pos. rate	TP/P	1−Type II error, power, sensitivity, recall
Pos. Pred. value	TP/P^*	Precision, 1−false discovery proportion
Neg. Pred. value	TN/N^*	

TABLE 4.7. *Important measures for classification and diagnostic testing, derived from quantities in Table 4.6.*

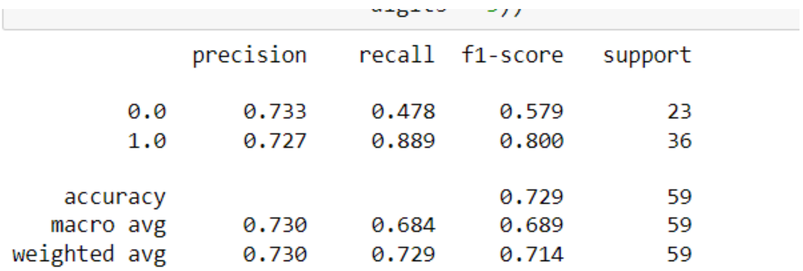
Sklearn:

$$\begin{bmatrix} 11 & 12 \\ 4 & 32 \end{bmatrix}$$

ISL book:

$$\begin{bmatrix} 11 & 4 \\ 12 & 32 \end{bmatrix}$$

An overall error rate of 16/59.
Sensitivity of 32/36 (Type 2 Error: 4/36)
Specificity of 11/23 (Type 1 Error =12/23)
Precision of 32/44



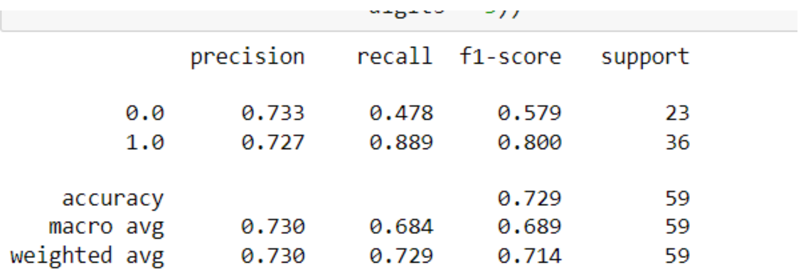
```
                      digits=3))

              precision    recall  f1-score   support

         0.0      0.733     0.478     0.579        23
         1.0      0.727     0.889     0.800        36

    accuracy                          0.729        59
   macro avg      0.730     0.684     0.689        59
weighted avg      0.730     0.729     0.714        59
```

Classification Example: results

For a probability threshold of ½.
The confusion matrix is:

$$\begin{bmatrix} 11 & 12 \\ 4 & 32 \end{bmatrix} \longrightarrow \begin{bmatrix} 11 & 4 \\ 12 & 32 \end{bmatrix}$$

An overall error rate of 16/59.
Sensitivity of 32/36 (Type 2 Error: 4/36)
Specificity of 11/23 (Type 1 Error =12/23)
Precision of 32/44



```
                digits=3))

              precision    recall  f1-score   support

         0.0      0.733     0.478     0.579        23
         1.0      0.727     0.889     0.800        36

    accuracy                          0.729        59
   macro avg      0.730     0.684     0.689        59
weighted avg      0.730     0.729     0.714        59
```

Let's try a probability threshold of 0.4.
The confusion matrix is:

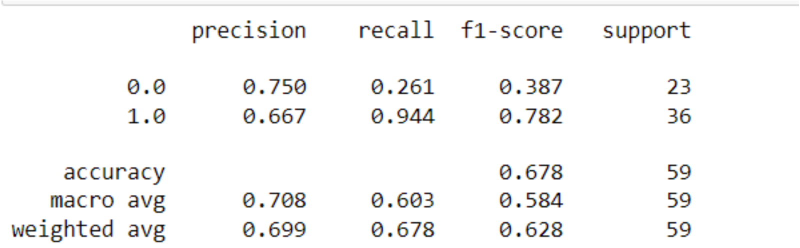
Sklearn:

$$\begin{bmatrix} 6 & 17 \\ 2 & 34 \end{bmatrix}$$

ISL book:

$$\begin{bmatrix} 6 & 2 \\ 17 & 34 \end{bmatrix}$$

An overall error rate of 19/59.
Sensitivity of 34/36 (type 2 error = 2/36)
Specificity of 6/23 (type 1 error = 17/23)
Precision of 34/36



```
              precision    recall  f1-score   support

         0.0      0.750     0.261     0.387        23
         1.0      0.667     0.944     0.782        36

    accuracy                          0.678        59
   macro avg      0.708     0.603     0.584        59
weighted avg      0.699     0.678     0.628        59
```

Classification Example: results

Let’s also look at the case where all months are classified as up (1)
The confusion matrix is:

$$\begin{bmatrix} 0 & 0 \\ 15 & 44 \end{bmatrix}$$

An overall error rate of 15/59.
Sensitivity of 44/44 (type 2 error = 0)
Specificity of 0/15 (type 1 error = 1)
Precision of 44/59

Classification Example: results

We now have three points on the ROC curve.

We need some more!

Sensitivity of 32/36
Type 1 Error =12/23

Sensitivity of 34/36
type 1 error = 17/23

Sensitivity of 44/44
type 1 error = 1

Classification: KNN Classifier

- Same simple principle, find the most similar points in the training data.
- Let’s make a prediction for month 101 using the first three lag differences (-23, 975, 7087) (using Euclidean distances)

95	2401	1	-296	356	395
96	3291	1	986	-429	337
97	334	0	890	1282	-785
98	617	1	-2957	-96	1711
99	1875	1	283	-3847	-1378
100	1852	0	1258	3240	-3751
101	1938	1	-23	975	7087

Classification: KNN Classifier

- If K = 1, the nearest neighbour is month 77

101	1938	1	-23	975	7087

77	1403	1	-130	422	5967

- Our prediction for month 101 is 1 (‘up’)

Classification: KNN Classifier

- If K = 5, the nearest neighbours are months 77, 29, 17, 65, 89

101	1	-23	975	7087

77	1	-130	422	5967
29	1	-185	331	5090
17	1	209	2	5113
65	0	292	146	9467
89	1	-263	-411	4037

- We can now have richer predictions, in 4 of the 5 similar months, the market went up 4 times. We can then write P(Up=4/5).
- If we assign weights to the distances we can compute a weighted estimate for the probability:

$$(P(\mathrm{up}) = 1\ (w_1 + w_2 + w_3 + w_5) + 0(w_4)),$$

where w_i is the weight for the i’th nearest month.