$$\frac{\partial L}{\partial \boldsymbol{b}} = \sum_{i=1}^{N}[y_i - \Lambda(\boldsymbol{b}'\boldsymbol{x_i})]\boldsymbol{x_i} \text{ (Eq. VIII.53)}$$

A numerical routine for optimization like the Newton method is able to solve the previous equation with respect to $\boldsymbol{b}$. In order to apply this algorithm, we also need to estimate the second derivative, which is obtained as:

$$\frac{\partial^2 L}{\partial \boldsymbol{b}\partial \boldsymbol{b}'} = -\sum_{i=1}^{N}\Lambda(\boldsymbol{b}'\boldsymbol{x_i})[\mathbf{1} - \Lambda(\boldsymbol{b}'\boldsymbol{x_i})]\boldsymbol{x_i}\boldsymbol{x_i}' \text{ (Eq. VIII.54)}$$

The function logit(Dataset,constant,stats) allows to estimate the $\boldsymbol{b}$ in accordance with the previous formulas. To do that, the function requires the following input arguments:

- Dataset is a numpy matrix which can be imported in the Python environment calling the importdata(Filepath) function, where Filepath is the place in which the csv file has been stored. In the first column, the Default indicator for the following year is stored, $\boldsymbol{y}$, and in the other five columns, the factors ($\boldsymbol{x}$) are stored.

- constant is a logic value that assumes the value True if the model includes a constant, otherwise it is False.

- stats is a logic value. If it is True, the routine returns the coefficients $\boldsymbol{b}$ as well as the statistics associated to the model, otherwise only $\boldsymbol{b}$.

Substantially, the core of the code lies in the while loop that allows to estimate the coefficients of the model using the ML principle.

The variable Lambda is the prediction, dlnl is the gradient, hesse is the Hessian and lnl is the log likelihood.

Once the gradient and the Hessian have been computed, the Newton rule can be applied. We take the inverse of the Hessian, hinv and multiply it with the gradient hinvg.

$$b_1 = b_0 - \left[\frac{\partial^2 \ln L}{\partial b_0 \partial b_0'}\right]^{-1}\frac{\partial \ln L}{\partial b_0} = b_0 - H(b_0)^{-1}\nabla(b_0) \text{ (Eq. VIII.55)}$$

The logit model has the convenient feature that the log-likelihood function is globally concave, as a result a gradient-based numerical routine is enough for being sure to reach the global optimum. The while loop ends to update the coefficient vector $\boldsymbol{b}$ when the change in the likelihood is sufficiently small or when the maximum number of iterations is reached, then the function returns the coefficients.

The output of the logit routine can be shaped based on the value assumed by stats. If it is False, the function returns a numeric array in which the coefficients of the regression ($\boldsymbol{b}$) are stored, otherwise a numeric matrix having the form reported in the Table VIII.18.

Statistics are fundamental for understanding the goodness of the model. To assess whether a variable helps to explain the default event or not, we can examine a t-ratio for the hypothesis that the variable's coefficient is zero. For the j-th coefficient, such a t-ratio is constructed so that:

$$t_j = \frac{b_j}{SE(b_j)} \text{ (Eq. VIII.56)}$$

Where SE is the estimated standard error of the coefficient.

b_1	b_2	...	b_K
$SE(b_1)$	$SE(b_2)$	...	$SE(b_K)$
$t_1 = b_1/SE(b_1)$	$t_2 = b_2/SE(b_2)$	...	$t_K = b_K/SE(b_K)$
$p - \text{value}(t_1)$	$p - \text{value}(t_2)$	...	$p - \text{value}(t_K)$
$\text{Pseudo} - R^2$	# iterations	0	0
$LR - \text{test}$	$p - \text{value}(LR)$	0	0
$\text{Log} - \text{likelihood(model)}$	$\text{Log} - \text{likelihood(restricted)}$	0	0

Table VIII.18 Output of the logit function

We take b from the last iteration of the Newton scheme and the standard errors of estimated parameters are derived from the Hessian matrix. Specifically, the variance of the parameter vector is the main diagonal of the negative inverse of the Hessian at the last iteration step. In the logit function, we have already computed the Hessian hinv for the Newton iteration, so we can quickly calculate the standard errors. We set the standard error of the j -th coefficient to sqrt(-hinv[j,j]). The t-ratios are then computed using the previous formula. In the logit model, the t-ratio does not follow a t-distribution as in the classical linear regression. Rather, it is compared to a standard normal distribution. Then, to obtain the p-value of a two-sided test, we exploit the symmetry of the normal distribution:

$$p - \text{value} = 2 \cdot \left(1 - \phi(|t|)\right) \quad \textit{(Eq. VIII.57)}$$

Where ϕ is the cumulative standard normal distribution.

The logit function returns standard errors, t-ratios and p-values in lines two to four of the output if the logical value stats is set to True. In a linear regression, an R^2 is usually reported as a measure of the overall goodness of fit. In nonlinear models estimated with maximum likelihood, the Pseudo- R^2 suggested by McFadden (1974) is typically reported. It is calculated as 1 minus the ratio of the likelihood of the estimated model ($\ln L$) and the one of a restricted model that only has a constant ($\ln L_0$):

$$\text{Pseudo} - R^2 = 1 - \frac{\ln L}{\ln L_0} \quad \textit{(Eq. VIII.58)}$$

Like the standard R^2, this measure is bounded by zero and one. Higher values indicate a better fit. The log-likelihood $\ln L$ is given by the log-likelihood function of the last iteration of the Newton procedure,and is thus already available. The loglikelihood of the restricted model is then left to be determined. With a constant only, the likelihood is maximized if the predicted default probability is equal to the mean default rate, $\bar{y}$. This can be achieved by setting the constant equal to the logit of the default, that is $b_1 = \ln\left[\frac{\hat{y}}{1-\hat{y}}\right]$.

For the restricted log-likelihood, we then obtain:

$$\ln L_0 = \sum_{i=1}^{N} y_i \ln[\Lambda(\boldsymbol{b}'\boldsymbol{x_i})] + (1 - y_i) \ln[1 - \Lambda(\boldsymbol{b}'\boldsymbol{x_i})] =$$

$$= \sum_{i=1}^{N} y_i \ln(\bar{y}) + (1 - y_i) \ln(1 - \bar{y}) = \textit{(Eq. VIII.59)}$$

$$= N \cdot [\bar{y} \ln(\bar{y}) + (1 - y_i) \ln(1 - \bar{y})]$$

The two likelihoods used for the Pseudo-R^2 can also be used to conduct a statistical test of the entire model, that is, test the null hypothesis that all coefficients except for the constant are zero. The test is structured as a likelihood ratio test: $LR = 2(\ln L - \ln L_0)$.

The more likelihood is lost by imposing the restriction, the larger the LR-statistic will be. The test statistic is distributed asymptotically χ^2 with the degrees of freedom equal to the number of restrictions imposed.

When testing the significance of the entire regression, the number of restrictions equals the number of variables K minus one. The chi2.sf(2*(lnL[Iter]-lnL0),K-1) gives the p-value of the LR test. Both LR and its p-value have been returned by the full output structure. The likelihoods $\ln L$ and $\ln L_0$ are also reported as well as the number of iterations needed to achieve the convergence. Running the function logit(Dataset, constant=True, stats=True), we obtain the results reported in the next figure. The arrangement of the numbers follows exactly the output template shown previously.

Regarding the overall fitting model statistics, we can look at the LR test (160.148) and its p-value (10^{-33}): the logistic regression is highly significant. The hypotheses "the five ratios add nothing to the prediction" can be rejected with high confidence and the regression model helps to explain the default events. Knowing that the model does predict defaults, we would like to know how well it does so.

An analyst usually turns to the R^2 for answering this question, but as in linear regression, setting up general quality standards in terms of a Pseudo- R^2 is difficult.

	0	1	2	3	4	5
0	-2.54348	0.414394	-1.45402	-7.99906	-1.59359	0.619721
1	0.266029	0.572478	0.229486	2.7024	0.323405	0.349199
2	-9.56089	0.723861	-6.33598	-2.95998	-4.92754	1.77469
3	0	0.469151	2.35832e-10	0.0030766	8.32703e-07	0.0759483
4	0.222058	12	0	0	0	0
5	160.148	9.20493e-33	0	0	0	0
6	-280.526	-360.6	0	0	0	0

Figure VIII.14 Full output of the logit function applied on the original credit dataset using all five ratios

A simple but often effective way of assessing this measure is to compare it with ones from other models estimated on similar data sets. From the literature, we know that scoring for listed US companies can achieve a

Pseudo- R^2 of about 35%-40%. This unfortunately indicates that the way we have set up the model may not be ideal given that our Pseudo- R^2 is equal to 22.21%: after this statistical analysis we focus on how to improve the performance of the model.

Turning to the regression coefficients, we can summarize that three out of five ratios have b that are significant on the 1% level or better because their p-value is below 0.01. If we reject the hypothesis that one of these coefficients is zero, we can expect to err with a probability of less than 1%. Each of the three variables (RE/TA, EBIT/TA and ME/TL) has a negative coefficient, meaning that increasing values of the variables reduce default probability. This is coherent, by economic reasoning, as retained earnings, EBIT and market value of equity over liabilities should be inversely related to default probabilities. The constant is also highly significant (p-value ~ 0) and the coefficient on working capital over total assets (WC/TA) and sales over total assets (S/TA), by contrast, exhibit a significance of only 46.92% and 7.59%, respectively. By conventional standards of statistical significance (5% traditionally is the most common choice) we would conclude that these two variables are not or only marginally significant and we would probably consider not using them for prediction. If on the other hand we simultaneously remove two or more variables based on their t-ratios, we should be aware of the possibility that variables might jointly explain defaults even though they are insignificant individually. To statistically test this possibility, we can perform a second regression in which we exclude variables that were insignificant in the first run and then conduct a likelihood ratio test. Running logit(CreditDataset[:,[0,2,3,4]],True,True), we reach the results shown in Figure VIII.15.

The likelihood test for the hypothesis $b_{WC/TA} = b_{S/TA} = 0$ is based on the comparison of the log-likelihoods $\ln L$ of the two models. It is constructed as:

$$LR = 2 \cdot [\ln L\,(\text{model 1}) - \ln L\,(\text{model 2})] \quad \textit{(Eq. VIII.60)}$$

In this case it is referred to a χ^2 distribution with two degrees of freedom because we impose two restrictions. chi2.sf(2*(-280.526-(-282.219)),2) gives 0.1840. This means that if we add the two variables WC/TA and S/TA to model 2, there is a probability of 18.40% that we do not add any explanatory power.

	0	1	2	3
0	-2.31833	-1.41974	-7.17936	-1.61562
1	0.235635	0.228767	2.72537	0.324763
2	-9.83866	-6.20607	-2.63427	-4.97476
3	0	5.43268e-10	0.00843192	6.53297e-07
4	0.217361	12	0	0
5	156.761	9.16409e-34	0	0
6	-282.219	-360.6	0	0

Figure VIII.15 Full output of the logit function applied on the original credit dataset using the most significant financial ratios

The LR test thus confirms the results of the individual tests: both individually and jointly, the two variables would be considered only marginally significant. However, it is important to highlight that a common best practice is to also perform out-of-sample tests of predictive performance before dropping variables from the model. Having specified a scoring model, we want to use it for predicting probabilities of default. In order to do so, we calculate the score and then translate it into a default probability in accordance with:

$$\text{Prob}(\text{Default}_i) = \Lambda(\text{Score}_i) = \frac{\exp(\boldsymbol{b'x_i})}{1+\exp(\boldsymbol{b'x_i})} = \frac{1}{1+\exp(-\boldsymbol{b'x_i})} \quad \textit{(Eq. VIII.61)}$$

The getPDfromLogit(CreditDataset,constant) is able to perform this task for the N observations in the database.

Even the sensitivity to the change of a factor is considered as very important to properly manage the logit model. The getPDsensitivityfromLogit(CreditDataset,constant,bump) absolves this necessity. It takes a further compulsory input argument, that is the bump to be applied to the variable to get the sensitivity of the model compared to the analyzed variable. The sensitivity of the model has been estimated like a Greek for all the available observations, using a two-sided finite difference: $\frac{f(x_i+h)-f(x_i-h)}{2h}$, where h is the bump applied to the factors and $i = 1, \dots, N$.

The output of the function is a matrix made of K columns. In the first column, the sensitivity of the model to a bump applied to all the K factors simultaneously has been reported, the other columns contain the partial sensitivity of the logit model obtained by individually applying the shock to each factor. Hence the second column contains the sensitivity of the model to the first factor (i.e. WC/TA) leaving the other variables unchanged; the third column contains the sensitivity of the model to the second factor (i.e. RE/TA) leaving the other $K-1$ variables unchanged and so on. The final output is a matrix of a dimension equal to $N \times K + 1$.

	0	1	2	3	4	5	6
0	0.00210391	-0.00239384	0.000927692	0.00119954	-0.000261078	-0.000499787	0
1	0.000604251	-0.00218426	0.000812542	0.0011971	0.000305775	-0.000418571	0
2	0.00688436	-0.00307632	0.000970709	0.00105519	-0.00180328	-0.000518927	0
3	0.0438242	-0.00532576	0.00210638	0.00301124	-0.0196708	-0.0013346	0
4	0.00681597	-0.00303925	0.000933568	0.00127319	-0.00192584	-0.000580051	0

Figure VIII.16 Model sensitivity

Having set the scoring models, the Probability of Default and their sensitivities respect to factors, it is worth wondering if the global performance of the regression can be improved in terms of Pseudo- R^2.

In general terms, it can be considered a good result to obtain an overall performance of the model which can be compared to the literature benchmarks for the US listed firms. One of the most popular techniques is to treat outliers in input variables. Explanatory variables in scoring models often contain a few extreme values. They can reflect truly exceptional situations of borrowers, but they can also be due to data errors, conceptual problems in defining a variable or accounting discretion. In any case, extreme values can have a large influence on coefficient estimates, which could impair the overall quality of the scoring model.

A first step in approaching the problem is to examine the distribution of the variables. With this aim, the function getDatasetDescriptiveStats(Dataset) allows to compute the main descriptive statistics for the five analyzed ratios. The output is an object of the DescriptiveStatistics class which contains the fields reported in the first column of Table VIII.19, together with the values assumed for each factor x_i.

Field	WC/TA	RE/TA	EBIT/TA	ME/TL	S/TA
Average	0.14	0.21	0.05	1.95	0.30
Median	0.12	0.22	0.05	1.14	0.26
Standard Deviation	0.17	0.33	0.03	2.99	0.21
Skewness	-1.01	-2.55	-4.84	7.75	4.48
Kurtosis	17.68	17.44	86.00	103.13	71.22
Minimum	-2.24	-3.31	-0.59	0.02	0.04
Percentiles[0]=0.50	-0.33	-1.72	-0.05	0.05	0.06
Percentiles[1]=1	-0.17	-0.92	-0.02	0.08	0.07
Percentiles[2]=5	-0.06	-0.25	0.02	0.22	0.10
Percentiles[3]=95	0.44	0.65	0.09	5.60	0.68
Percentiles[4]=99	0.58	0.90	0.12	14.44	1.05
Percentiles[5]=99.5	0.63	0.94	0.13	18.94	1.13
Maximum	0.77	1.64	0.20	60.61	5.01

Table VIII.19 Descriptive Statistics

A common benchmark for assessing an empirical distribution is the normal distribution. The reason is not that there is a priori a reason why the variables should follow a gaussian distribution, but rather that the normal serves as a good reference point because it describes a distribution in which extreme events have been averaged out.

We remind that the relevant theorem from statistics is the central limit theorem, which says that if we sample from any probability distribution with finite mean and finite variance, the sample mean will tend to the normal distribution as we increase the number of observations to infinity. A good indicator for the existence of outliers is the excess kurtosis (that is defined as kurtosis minus 3). The normal distribution has excess kurtosis of zero, but the variables used in the example have very high values ranging from 17.4 to 103.1.

In this context, a positive excess kurtosis indicates that there are relatively more observations far away from the mean, compared to the normal.

The variables are also skewed, meaning that extreme observations are concentrated on the left (if the skewness is negative) or on the right (if skewness is positive) of the distribution. In addition, we can look at the percentiles.

For example, a normal distribution has the property that 99% of all observations are within ± 2.58 standard deviations of the mean. For the variable ME/TL, this would lead to the interval [-5.77, 9.68]. The empirical 99% confidence interval, however, is [0.05, 18.94] that is wider and shifted to the right, confirming the information we acquired by observing the skewness and kurtosis of ME/TL.

Considering WC/TA, we see that 99% of all values are in the interval [-0.33, 0.63], which is roughly in line with what we would expect under a normal distribution, namely [-0.30, 0.58]. In the case of WC/TA, the outlier problem is thus confined to a small subset of observations. This is most evident by looking at the minimum value of WC/TA: it is -2.24, which is very far away from the bulk of the observations, as it is 14 standard deviations away from the mean, and 11.2 standard deviations away from the 0.5% percentile.

Having identified the existence of extreme observations, a rigorous inspection of the data is advisable because it can lead to the discovery of correctable data errors. In many applications, however, this will not lead to a complete elimination of outliers; even data sets that are 100% correct can exhibit bizarre distributions. Accordingly, it is useful to have a procedure that controls the influence of outliers in an automated and objective way. A commonly used technique applied for this purpose is the so-called **winsorization**, which means that extreme values are pulled to less extreme ones.

Saying that the analyst specifies a certain winsorization level α means that values below the α -percentile of the variable distribution are set equal to the α -percentile, values above the $1-\alpha$ a percentile are set equal to $1-\alpha$. Common values for α are 0.5%, 1%, 2% or 5%. The winsorization level can be set separately for each variable in accordance with its distributional characteristics, providing a flexible and easy way of dealing with outliers without discarding observations. Running the function for $\alpha = 1\%$ for each x_i we obtain the third and fourth moments shown in Table VIII.20:

Measure	WC/TA	RE/TA	EBIT/TA	ME/TL	S/TA
Skewness	0.63	-0.95	0.14	3.30	1.68
Kurtosis	0.01	3.20	1.10	13.48	3.42

Table VIII.20 Skewness and Kurtosis

Both skewness and kurtosis are now much closer to zero. Let us note that both statistical characteristics are still unusually high for ME/TL. This might explain a higher winsorization level for this factor, but there is a smarter alternative. Given that ME/TL has many extreme values to the right of the distribution, a good idea can be to take the logarithm with the aim of pulling them to the left without blurring the differences between those beyond a certain threshold, as we do applying the Winsor method. The logarithm of ME/TL (after winsorization at the 1% level) has a skewness of -0.11 and a kurtosis of 0.18, suggesting that the logarithmic

transformation works for ME/TL in terms of outliers.

This transformation leads to an important benefit in terms of Pseudo- R^2: running the logit(WinsorDataset, True, True) we obtain a value of 25.48% and launching the same command with the log of ME/TL we reach 33.97%. Figure VIII.17 shows the full output of the logit function applied on the original credit dataset using winsorized variables at 1%:

	0	1	2	3	4	5
0	-2.4745	0.376492	-2.53848	-22.978	-1.16084	1.409
1	0.318889	0.829262	0.335097	5.48897	0.298466	0.567607
2	-7.75977	0.454008	-7.57534	-4.18622	-3.88935	2.48236
3	8.43769e-15	0.649823	3.57492e-14	2.83644e-05	0.000100513	0.0130517
4	0.254793	12	0	0	0	0
5	183.757	8.43311e-38	0	0	0	0
6	-268.721	-360.6	0	0	0	0

Figure VIII.17 Full output of the logit function applied on the original credit dataset using winsorized variables at 1%

Figure VIII.18 shows the full output of the logit function applied on the original credit dataset using winsorized variables at 1% and the $\ln ME/TL$:

	0	1	2	3	4	5
0	-4.70936	0.909294	-1.67887	-17.0034	-1.40481	1.07475
1	0.377638	0.852232	0.379533	5.81077	0.167202	0.595173
2	-12.4706	1.06696	-4.42352	-2.9262	-8.40187	1.80578
3	0	0.285991	9.71075e-06	0.00343135	0	0.0709521
4	0.339705	10	0	0	0	0
5	244.995	6.51378e-51	0	0	0	0
6	-238.102	-360.6	0	0	0	0

Figure VIII.18 Full output of the logit function applied on the original credit dataset using winsorized variables at 1% and the ln ME/TL

The examination of univariate relationships between default rates and explanatory variables can give valuable hints as to which transformation is appropriate. In the case of ME/TL, it supports the logarithmic one, but in many other cases, it may support a polynomial representation like in the case of sales growth. Often, however,

which transformation to choose may not be clear, and an automatic procedure can be very useful especially when there is a huge number of factors. To such end, we can employ the procedure coded in the XTransform function. The main idea is to use the default rate of the range to which they are assigned instead of entering the original values of the variable into logit analysis. In this way we use a data-driven, nonparametric transformation of the input data. Running the function with a number of range equal to 20, we obtain a Pseudo-R^2 very close to that reported in the literature (46.001%). In this example we have described the estimation of a scoring model with logit.

A common alternative is the **probit model**, which replaces the logistic distribution in $\Lambda(\text{Score})$ with the standard normal distribution. Experience and literature suggest that the choice of the distribution is not so crucial in most settings; predicted default probabilities and performance are fairly close.

probit(Dataset, constant, stats) implements this alternative and the Pseudo-R^2 is very close to the previous model (45.1%).

	0	1	2	3	4	5
0	5.15291	0.459452	0.329845	0.187741	0.323704	0.344958
1	2.41525	0.347555	0.175824	0.207495	0.254786	0.268809
2	2.13349	1.32195	1.87599	0.904797	1.27049	1.28328
3	0.0328844	0.186183	0.060656	0.365573	0.203908	0.199393
4	0.450994	48	0	0	0	0
5	325.257	3.70297e-68	0	0	0	0
6	-197.971	-360.6	0	0	0	0

Figure VIII.19 Full output of the probit function

Scoring models have advantages and limitations. Among the first ones we can identify:

- **objectivity**: the choice of variables and the weights to be attributed to each of them depend on a statistical model and on the database used to calibrate the model.

- **strength**: once the model has been defined, the bank is able to implement the investigation by minimizing the time and costs of the procedure and can also implement adequate monitoring procedures.

- **uniformity**: the adoption of a standardized model for the evaluation of the counterparty allows the calculation of average scores by geographical area and branch.