Predictive Modeling

Classification and Logistic Regression

Mirko Birbaumer

HSLU T&A

Examples of Classification Problems

- An e-mail client (such as MS Outlook or Mozilla Thunderbird) receives an e-mail. Is it a `spam` or proper (`ham`) mail?
- Based on process data such as temperatures, pressures etc., a manufacturer wants to predict the state (`okay` vs. `defect`) of an engine in the near future (*predictive maintenance*).
- A doctor has to attribute the symptoms of a patient to three possible medical conditions, e.g. `stroke`, `drug overdose`, and `epilleptic seizure`.

Predictive Model: Classification versus Regression

- A **predictive model** is a functional relation between a (dependent) *response variable* Y and (independent) *predictor variables* $X_1, \ldots, X_p$

  $$Y = f(X_1, \ldots, X_p)$$

  The function f is usually unknown and has to be *estimated* from data

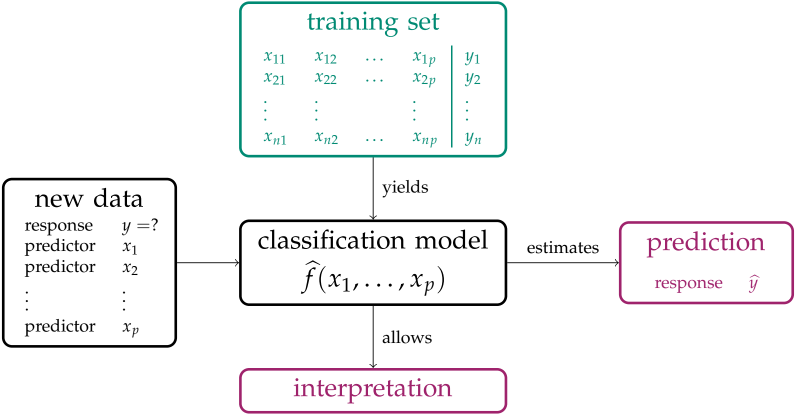
- **Multiple linear regression** is a methodology of prediction by means of a linear function f that is estimated via least squares optimization.

- Use of multiple linear regression, however, is limited to *quantitative* response variables Y, i.e., where Y is a numeric scalar.

- **Classification** deals with *qualitative* (or *categorical*) response variables, i.e., if Y takes on values in a finite number of *classes* or *categories*.

Examples of Categorical Variables

- A person's gender: `male` or `female`
- A brand name: brands `A`, `B`, or `C`
- Tumor class: `EWS`, `RMS`, `NB`, or `BL`
- A person's eye color: `green`, `blue`, or `brown`
- Whether a student passes an exam: `yes` or `no`



training set
x_{11} x_{12} $\dots$ x_{1p} | y_1
x_{21} x_{22} $\dots$ x_{2p} | y_2
$\vdots$ $\vdots$ $\vdots$ | $\vdots$
x_{n1} x_{n2} $\dots$ x_{np} | y_n
yields
new data
response $y = ?$
predictor x_1
predictor x_2
$\vdots$ $\vdots$
predictor x_p
classification model
$\widehat{f}(x_1, \dots, x_p)$
estimates
prediction
response $\widehat{y}$
allows
interpretation

Classification: Example

We want to predict whether a person will default on his or her debt (i.e., is not able to pay his or her due), based on the annual income and the monthly credit card bill. The data set consists of the following variables:

- `default`: Binary response variable (`Yes` or `No`), whether or not the person defaults.
- `income`: (first numeric predictor) annual income of the person.
- `balance`: (second numeric predictor) monthly credit card balance.

Please check example `0.1` in the chapter `Logistic Regression`

Logistic Regression: `Default` Example

- We aim at modeling the *probability* that `default` equals `Yes` depending on the value of `balance` (numeric predictor X)

- We are looking for a model that predicts the *conditional probability*

$$P(\texttt{default=Yes} \mid \texttt{balance})$$

  which we abbreviate $p(\texttt{balance})$

- For any given new observation of `balance` our model then predicts a probability for the response `default` being `Yes`

Logistic Regression: `Default` Example

- We aim at modeling the conditional probability

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

- We consider only *binary* response variables Y, where we use the numeric encoding 1 and 0 (so $Y = 1$ would correspond to `default = Yes`)

- Naïve approach:

$$p(X) = \beta_0 + \beta_1 X$$

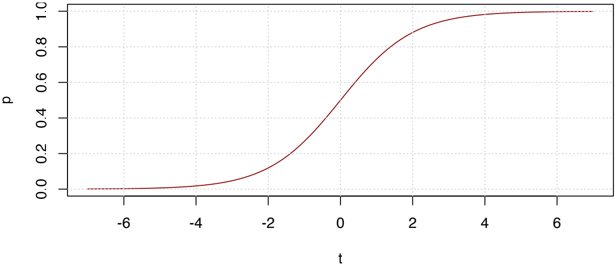
- Please check example `1.1` of the `Logistic Regression` chapter

Simple Logistic Regression

- Idea underlying *logistic regression* is to modify a linear function by composing it with another function which *shrinks* all of $\mathbb{R}$ to $[0, 1]$

- In logistic regression, we choose the *logistic* function

$$p(t) = \frac{e^t}{1 + e^t} \quad \text{with} \quad t \in \mathbb{R}$$



Please check example 1.2 of the Logistic Regression chapter

Simple Logistic Regression

Simple logistic regression

Given a binary response variable Y and a quantitative predictor X, the *simple logistic regression model* is defined as

$$P(Y = 1|X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} \tag{1}$$

The parameters β_0 and β_1 are called *regression coefficients* and are estimated from the training set.

- In order to estimate β_0 and β_1, maximum likelihood method is applied (see lecture notes)
- Please check example `2.2` of the chapter `Logistic Regression`

Odds

- Using some basic rearrangements, we find from (1)

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

- Please solve `Problem 1` of the exercise sheet

- The quantity $p(X)/(1 - p(X))$ is called **odds**

- **odds** is an equivalent way of expressing the probability of an event and is used for instance in betting agencies

- Odds values close to 0 and ∞ indicate small and large probabilities, respectively

Example: odds

- If one out of five individuals default on their debts, then $p(X) = 0.2$. For the odds we find

$$\frac{0.2}{1-0.2} = \frac{0.2}{0.8} = \frac{1}{4}$$

- If, however, 9 out of 10 persons default, so $p(X) = 0.9$, and the odds are

$$\frac{0.9}{1-0.9} = \frac{0.9}{0.1} = 9$$

Example: odds

- In a betting office, the odds of $4:1$ for soccer team A winning against soccer team B means that

$$\frac{p}{1-p} = \frac{4}{1} \quad \Leftrightarrow \quad p = \frac{4}{5}$$

- Team A is expected to win 4 out of 5 matches against B.

Logit

- Taking the natural logarithm on both sides of

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

  we obtain

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

- Left side of this equation is called the *log-odds* or **logit**
- Interpretation of the coefficients β_0 and β_1 in terms of the logit: a change of the predictor X by one unit amounts to an average change by β_1 of the logit of the response being true.
- Please solve `Problem 2` of the exercise sheet

Model Prediction: Example `Default`

- In the `Default` example, we find the estimates

$$\hat{\beta}_0 = -10.6513 \qquad \text{and} \qquad \hat{\beta}_1 = 0.0055$$

- Thus the estimated logistic regression model set is

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X}} = \frac{e^{-10.6513 + 0.0055X}}{1 + e^{-10.6513 + 0.0055X}}$$

- Thus, if an individual has `balance = 1000`, then the model yields

$$\hat{p}(1000) = \frac{e^{-10.65 + 0.0055 \cdot 1000}}{1 + e^{-10.65 + 0.0055 \cdot 1000}} \approx 0.00577$$

- An individual with `balance = 2000` has default probability of $\approx 59\%$
- Please check example `3.1` of the `Logistic Regression` chapter

Model Prediction: Example `Default`

- Obviously, values of $p(\texttt{balance})$ are between 0 and 1
- For any given new observation of `balance` our model then predicts a probability for the response `default` being `Yes`
- Predicting the proper class then is carried out by thresholding, say, at 0.5: i.e., if for an individual we find $p(\texttt{balance}) > 0.5$, then the prediction would be `default=Yes`
- Please check example `3.2` of the `Logistic Regression` chapter

- For a given value x of X, the corresponding response value $\hat{y}$ of Y is predicted as

$$\hat{y} = \hat{f}(x) := \begin{cases} 1 & \text{if } \hat{p}(x) > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

- The choice of the threshold (here 0.5) is crucial and by no means trivial

- For example, consider a situation where you want to avoid false accusations (*false positives*), say in court. Then you should use a higher threshold

- At this point, a natural question arises: How well does this classification scheme *predict* the binary response variable Y?

Model Assessment

A first approach for model assessment is the

Classification error

Let $(x_1, y_1), \ldots, (x_n, y_n)$ be observations of (X, Y) and $\hat{y}_i = \hat{f}(x_i)$ the corresponding predictions. The *classification error* is given by

$$\mathrm{Err} = \frac{1}{n} \sum_{i=1}^{n} I(y_i \neq \hat{y}_i) \tag{2}$$

Here, I denotes the *indicator function* which takes on value 1 if its argument is true and 0 otherwise. In other words, the classification error is the proportion of cases with a wrong prediction (either false positive or false negative).

See example `3.2` of the `Logistic Regression` chapter

Confusion Matrix

observed value y

		1	**0**	**total**
predicited value $\widehat{y}$	**1**	True positive	False positive	P'
	0	False negative	True negative	N'
	total	P	N	

Here,

1. *True positive* refers to the number of cases that are correctly classified as 1 ($y_i = \widehat{y}_i = 1$).
2. *False positive* is the number of cases that are classified as 1 but which are truly 0 ($\widehat{y}_i = 1$ and $y_i = 0$)
3. *False negative* is the number of cases that are classified as 0 but which are truly 1 ($\widehat{y}_i = 0$ and $y_i = 1$)
4. *True negative* is the number of cases that are classified as 0 and which are truly 0 ($\widehat{y}_i = y_i = 0$)

See examples `3.3` of the `Logistic Regression` chapter

Accuracy

Accuracy

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observations to the total observations:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

- Example:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} = \frac{100 + 9625}{100 + 42 + 233 + 9625} = 0.9725$$

- For our model, we have got 0.97 which means our model is approx. 97 % accurate.

Precision

Precision

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- Example `Default`: The question that this metric answers is: among all people that were predicted to default, i.e., `default=Yes`, how many actually defaulted?

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{100}{100 + 42} = 0.70$$

- Low precision relates to high false positive rate. We have got a precision of 0.70 which is not any more so convincing

Recall (Sensitivity)

Recall (Sensitivity)

Recall is the ratio of correctly predicted positive observations to all positive observations.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- Example `Default`: The question recall answers is: among all people that truly defaulted, how many did we predict?

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{100}{100 + 233} = 0.30$$

- We have got a recall of 0.30 which is very bad for this model.

F1 score

F1 score

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account.

$$\text{F1 Score} = \frac{2 \cdot (\text{Recall} \cdot \text{Precision})}{(\text{Recall} + \text{Precision})}$$

- Example `Default`:

$$\text{F1 Score} = \frac{2 \cdot (\text{Recall} \cdot \text{Precision})}{(\text{Recall} + \text{Precision})} = \frac{2 \cdot 0.30 \cdot 0.70}{0.30 + 0.70} = 0.42$$

- For the logistic regression model on the `Default` data set, the F1 score thus is 0.42
- See example `3.8` of the `Logistic Regression` chapter

Imbalance of Classes: Example `Default`

- Confusion matrix shows that the present classification scheme is by **no** means useful, in particular, if you want to predict the case of `default=Yes`

- Reason for this bad result: **imbalance** of the two classes; training data only contains 333 out of 10 000 cases with `default=Yes`

- Therefore, the likelihood function is dominated by the factors corresponding to `default=No`, so the parameters are chosen as to match mainly those cases

- Note also that the trivial classifier predicting all observations x to $\hat{f}(x) = 0$ has a classification error of $333/10000 = 0.0333$ which is not much worse than that of our logistic model

Imbalance of Classes: Example `Default`

- There are several approaches for coping with the problem of **imbalanced classes**
- One of the simplest is **down-sampling** of the major class
- See example `3.10` of the `Logistic Regression` chapter

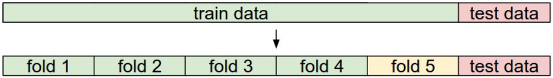
Cross-Validation

- Validating the predictive accuracy of a statistical model on the same data the model was built from: by no means a good idea!

- Alternative: collect new data with known labels and validate the model by computing the classification error on this new set - known as **validation set** or **test data** $\longrightarrow$ expansive!

- Split data into **test data** and **training data**, then split training data into k **folds**

- 5-fold Cross-Validation:



- In general: $k = 5, 10$; for $k = n$: *leave-one-out cross-validation*

Cross-Validation

- Estimated classification error with classification error of ith fold: Err_i

$$\mathrm{CV}_{(k)} = \frac{1}{k}\sum_{i=1}^{k}\mathrm{Err}_i$$

- See example `4.1` of the `Logistic Regression` chapter

- Please solve `Problem 5` of exercise sheet

Multiple Logistic Regression

If we replace the linear function $\beta_0 + \beta_1 X$ in the simple logistic regression approach by a multivariate linear function

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p$$

Multiple Logistic regression

Given a binary response variable Y and predictors $X_1, \ldots, X_p$, the logistic regression model is defined by

$$P(Y|X_1, \ldots, X_p) = \frac{\mathrm{e}^{\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p}}{1 + \mathrm{e}^{\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p}}$$

The parameters $\beta_0, \beta_1, \ldots, \beta_p$ are called *regression coefficients* and are estimated from the training set.

Multiple Logistic Regression: `Default` Example

- The coefficients $\beta_0, \ldots, \beta_p$ are estimated using the maximum likelihood method
- See example **5.1** of the **Logistic Regression** chapter
- Please solve **Problem 4** of exercise sheet