## Objectives

Logistic regression is one of the most studied and widely used classification algorithms, probably due to its popularity in regulated industries and financial settings. Although more modern classifiers might likely output models with higher accuracy, logistic regressions are great baseline models due to their high interpretability and parametric nature. This module will walk you through extending a linear regression example into a logistic regression, as well as the most common error metrics that you might want to use to compare several classifiers and select that best suits your business problem.

* Identify, use, and interpret error metrics to evaluate classification models.
* Identify common supervised machine learning algorithms.
* Build logistic regression classification models with sklearn.
* Use logistic regression models for classification.

## About this course

This course introduces you to one of the main types of modelling families of supervised Machine Learning: Classification. You will learn how to train predictive models to classify categorical outcomes and how to use error metrics to compare across different models. The hands-on section of this course focuses on using best practices for classification, including train and test splits, and handling data sets with unbalanced classes. By the end of this course you should be able to:

1. Differentiate uses and applications of classification and classification ensembles
2. Describe and use logistic regression models
3. Describe and use decision tree and tree-ensemble models
4. Describe and use other ensemble methods for classification
5. Use a variety of error metrics to compare and select the classification model that best suits your data
6. Use oversampling and undersampling as techniques to handle unbalanced classes in a data set

**Who should take this course?**

This course targets aspiring data scientists interested in acquiring hands-on experience with Supervised Machine Learning Classification techniques in a business setting.

**What skills should you have?**

To make the most out of this course, you should have familiarity with programming on a Python development environment, as well as fundamental understanding of Data Cleaning, Exploratory Data Analysis, Calculus, Linear Algebra, Probability, and Statistics.

## Optional: Introduction to IBM Watson Studio

The coding sections of this course leverage Jupyter notebooks. You may use the Python Interactive Development Environment (IDE) of your preference to open those files. At IBM, we prefer using IBM Watson Studio as it runs on the cloud, leverages collaboration, and makes it easy to interact with multiple AI tools and APIs. If you don't have an IBM Cloud account already, you can sign up for a free account and access Watson Studio using this link: <https://www.ibm.com/cloud/watson-studio>.

If you are not familiar with IBM Watson Studio, the resources below will help you get started.

## **IBM Watson Studio in a Nutshell**

**IBM Watson Studio is based on open source tools such as Python, R, Apache Spark, and Jupyter notebooks.**

You will not have the problems of having to learn an entirely new framework, or of ‘’vendor lock-in’’ caused by creating code and data that only runs on one system.

Watson Studio gives you a web-based tool to run all of your open source data science tools. The main advantage to using Watson Studio is that you do not have to spend a lot of time setting up environments on your own computer to run your tools. Another advantage of Watson Studio is that it makes collaboration much easier by allowing you to share code and data via Watson Knowledge Catalog.

Most of the case studies in this course may be run in Watson Studio or your preferred Python IDE. However if you are serious about learning how data science applies in a large enterprise setting, use Watson Studio. IBM clients from across all industries and sizes rely on Watson Studio as a tool that facilitates collaboration, innovation, and easy deployment.

## **Running the course notebooks in Watson Studio**

**All of the notebooks in these courses are written to run locally on your computer running a Jupyter notebook server. If you wish to run the notebooks in Watson Studio in the IBM Cloud, you will need to add some modifications to each notebook.**

## Optional: Overview of IBM Watson Studio

IBM Watson Studio is a cloud-based platform for machine learning and data science. **For data scientists, one of Watson Studio’s strengths is that its development environment is fully integrated with the most relevant Data Science and Machine Learning open source tools, including Python, R, Scala, Jupyter, and Apache Spark.**

With Watson Studio, you can keep all of the assets associated with your work organized into projects. Watson Studio also allows you to easily collaborate with others and share assets across an enterprise. If you are unfamiliar with IBM Watson Studio, you should take some time to review the materials below.

This link will take you a useful Getting Started tutorial on setting up IBM Watson Studio: [IBM Watson Studio: Getting Started](https://dataplatform.cloud.ibm.com/docs/content/wsj/getting-started/get-started-wdp.html)

If you want to learn more about IBM Watson Studio, please check out our [learning resources for Watson Studio](http://ibm.biz/watson-studio-learning).

## Introduction: What is Classification?

Regression – outcome is continuouis (numerical) answers how much

Prediction examples:

* House prrices
* Box office revenues
* Event attendance
* Network load
* Portfolio losses

Classification – outcome is a category

Prediction examples:

* Detecting fraudulent transactions
* Customer churn
* Event attendance
* Network load
* Loan default

What is classification:

What is it needed for

Model data with:

* Features that can be quantified
* Labels that are known

Method to measure similarity

Modeling classification:

Examples of models used for supervised learning: classification:

* Logistic regression
* k-nearest neighbors
* Support vector machines
* Neural networks
* Decision tree
* Random forests
* Boosting
* Ensemble models

Each of these models can be used for both regression and classification

Introduction to logistic regression

Introducing the sigmoid function

Linear regression for classification.

Logistic vs linear regression

Decision boundary

Classification with logistic regression

Multiple classes: one vs all

Multiclass decision boundary – assign most probable class to each region

#import the class containing the classification method

From sklearn.linear\_model import LogisticRegression

# create an instance of the class

LR = LogisticRegression(penalty=’12’, c=10.0)

# fit the instance of the data and then predict the expected value

LR = LR.fit(X\_train, y\_train)

Y\_predict = LR.predict(X\_test)

# can now view the output fitted coefficients

LR.coef\_

#tune regularization parameters wsith cross validation

LogisticRegressionCV

Classification with logistic regression

Applications for logistic regression:

* Customer spending: how likely is a customer is to be a top 5% spender, using previous purchasaed data
* Customer engagement which customers are most likely to engage in the next 6 months
* E commerce which transactions are fraudulent, using customer characteristics, location, ip address, etc.
* Finance/risk predicting whther a loan will default

Interpretation vs prediction:

* In addition to prediction, we may want to evaluate the importance of each factor in influencing outcomes

Confusion matrix, accuracy, specificity, precision, and recall

Choosing the right error measurement

You are asked to build a classifier to predict whether individuals have leukemia

Training data:

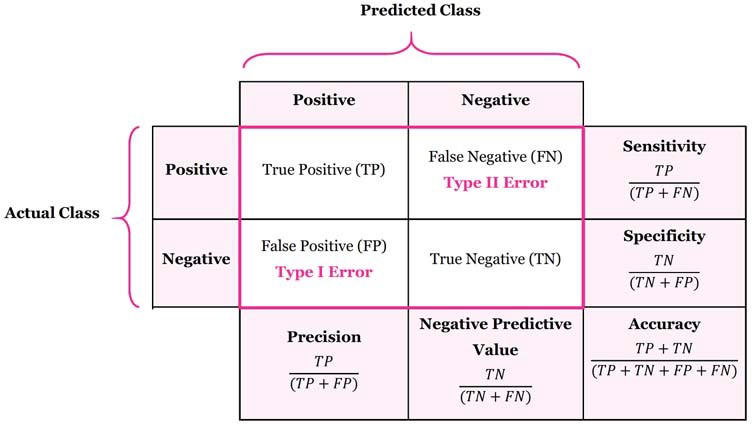
* 1% patients have leukemia, 99% are healther

Measure accuracy:

* Total % of predictions that are correct

Builod a simple model that always predicts “healthy”.

Accuracy will be 99%



Accuracy: predicting correctly

Recall / sensitivity: identifying all positive instances

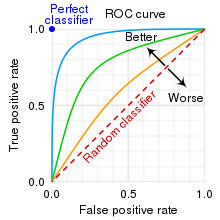
Precision: identify only positive instances

Specificity: avoiding false alarms

F1: precision \* recall / precision + recall

Classification Error metrics: ROC and Precision-recall curves

Receiver operating characteristic(roc)



Precision-recall curve

Measures tradeoff between precison and recall

Choosing the right approach

Which approach works best for choosing a classifier?  
Roc curve:

* Generally better for data with balanced classes

Precision-recall curve:

* Generally better for data with imbalanced classes

The right curve depends on tying results (true positives, true negatives, etc.) to outcomes (relative cost of false positive or false negative)

The curves compare classifiers generally ( across possible decion thresholds), which may be less relevant to bus. Objectives.

Multiple class error metrics

Most multi-class error metrics are similar to binary versions, they just expand elements as a sum.

Classification error metrics: the syntax

#import the desired error function

From sklearn.metrics import accuracy\_score

# calculat ethe error on the test and predicted ddata sets

Accuracy\_value = accuracy\_score(y\_test, y\_pred)

# lost of other error metrics and diagnostic tools:

From sklearn.metrics import precision\_score, recall\_score, f1\_score, roc\_auc\_score, confusion\_matrix, roc\_curve, precision\_recall\_curve

O3a\_lab\_logistic regression\_error metrics LAB

Data.iloc[:-1].min().value\_counts()

We have 561 –1.0 as our minimum values

-1 doesn’t include last column

Data.Activity.value\_counts()

Shows that the data is balanced

From hree we can think of what type of error metric to use with our balanced dataset

LabelEncoder for converting a string to a number

What does `df.columns[:-1]` do

.query will narrow down the entries we are looking for

StratifiedShuffleSplit

Test\_size=0.3 meaning 30%

We call split on our stra shuffle split with our two x values data[feature\_cols] and data.activity

C is the inverse of lambda

Questino 6

Create list of coeff labels and models

Append a panda series to y\_pred

Mod.predict = model predict

Mod.predict\_proba

Question 7

Y\_prob

Question 8

Plot out confusion matrix for each label

Create heat map

## End of module review: Logistic Regression

### **Classification Problems**

The two main types of supervised learning models are:

* Regression models, which predict a continuous outcome
* Classification models, which predict a categorical outcome

The most common models used in supervised learning are:

* Logistic Regression
* K-Nearest Neighbors
* Support Vector Machines
* Decision Tree
* Neural Networks
* Random Forests
* Boosting
* Ensemble Models

With the exception of logistic regression, these models are commonly used for both regression and classification. Logistic regression is most common for dichotomous and nominal dependent variables.

### **Logistic Regression**

Logistic regression is a type of regression that models the probability of a certain class occurring given other independent variables. It uses a logistic or logit function to model a dependent variable. It is a very common predictive model because of its high interpretability.

### **Classification Error Metrics**

A confusion matrix tabulates true positives, false negatives, false positives and true negatives. Remember that the false positive rate is also known as a type I error. The false negatives are also known as a type II error.

Accuracy is defined as the ratio of true positives and true negatives divided by the total number of observations. It is a measure related to predicting correctly positive and negative instances.

Recall or sensitivity identifies the ratio of true positives divided by the total number of actual positives. It quantifies the percentage of positive instances correctly identified.

Precision is the ratio of true positive divided by total of predicted positives. The closer this value is to 1.0, the better job this model does at identifying only positive instances.

Specificity is the ratio of true negatives divided by the total number of actual negatives. The closer this value is to 1.0, the better job this model does at avoiding false alarms.

The receiver operating characteristic (ROC) plots the true positive rate (sensitivity) of a model vs. its false positive rate (1-sensitivity).

The area under the curve of a ROC plot is a very common method of selecting a classification methods.

The precision-recall curve measures the trade-off between precision and recall.

The ROC curve generally works better for data with balanced classes, while the precision-recall curve generally works better for data with unbalanced classes.

### Question 1

Correct

1.00 points out of 1.00

Flag question

#### Question text

(True/False) One of the requirements of logistic regression is that you need a variable with two classes.

Select one:

True

False

#### Feedback

Correct! You can use a multinomial logistic regression if you have more than two classes. You can review the demo in lesson 2 of this module, in which you did a multinomial logistic to predict a target variable with more than two classes.

The correct answer is 'False'.

### Question 2

Correct

1.00 points out of 1.00

Flag question

#### Question text

(True/False) The shape of ROC curves are the leading indicator of an overfitted logistic regression.

Select one:

True

False

#### Feedback

Correct! Although overfitted models tend to have really high ROC curves with high values of area under the curve, a classification matrix or a measure like accuracy can be more reliable. Please review the lesson Confusion Matrix, Accuracy, Specificity, Precision, and Recall.

The correct answer is 'False'.

### Question 3

Correct

1.00 points out of 1.00

Flag question

#### Question text

You are evaluating a binary classifier. There are 50 positive outcomes in the test data, and 100 observations. Using a 50% threshold, the classifier predicts 40 positive outcomes, of which 10 are incorrect.

What is the classifier’s Precision on the test sample?

Select one:

A.

60%

B.

75%

C.

80%

D.

25%

#### Feedback

Correct! You can find more information in the lesson Confusion Matrix, Accuracy, Specificity, Precision, and Recall.

The correct answer is: 75%

### Question 4

Correct

1.00 points out of 1.00

Flag question

#### Question text

You are evaluating a binary classifier. There are 50 positive outcomes in the test data, and 100 observations. Using a 50% threshold, the classifier predicts 40 positive outcomes, of which 10 are incorrect.

What is the classifier’s Recall on the test sample?

Select one:

A.

60%

B.

25%

C.

75%

D.

80%

#### Feedback

Correct! You can find more information in the lesson Confusion Matrix, Accuracy, Specificity, Precision, and Recall.

The correct answer is: 60%

### Question 5

Correct

1.00 points out of 1.00

Flag question

#### Question text

You are evaluating a binary classifier. There are 50 positive outcomes in the test data, and 100 observations. Using a 50% threshold, the classifier predicts 40 positive outcomes, of which 10 are incorrect.

What is the classifier’s F1 score on the test sample?

Select one:

A.

66.7%

B.

50%

C.

67.5%

D.

70%

#### Feedback

Correct! You can find more information in the lesson Confusion Matrix, Accuracy, Specificity, Precision, and Recall.

The correct answer is: 66.7%

### Question 6

Correct

1.00 points out of 1.00

Flag question

#### Question text

You are evaluating a binary classifier. There are 50 positive outcomes in the test data, and 100 observations. Using a 50% threshold, the classifier predicts 40 positive outcomes, of which 10 are incorrect.

Increasing the threshold to 60% results in 5 additional positive predictions, all of which are correct. Which of the following statements about this new model (compared with the original model that had a 50% threshold) is TRUE?

Select one:

A.

The area under the ROC curve would remain the same.

B.

The area under the ROC curve would decrease.

C.

The F1 score of the classifier would remain the same.

D.

The F1 score of the classifier would decrease.

#### Feedback

Correct! For more information, please review the lesson ROC and Precision-Recall Curves.

The correct answer is: The area under the ROC curve would remain the same.

### Question 7

Correct

1.00 points out of 1.00

Flag question

#### Question text

You are evaluating a binary classifier. There are 50 positive outcomes in the test data, and 100 observations. Using a 50% threshold, the classifier predicts 40 positive outcomes, of which 10 are incorrect.

The threshold is now increased further, to 70%. Which of the following statements is TRUE?

Select one:

A.

The Precision of the classifier would decrease.

B.

The Recall of the classifier would decrease.

C.

The Recall of the classifier would increase or remain the same.

D.

The Precision of the classifier would increase or remain the same.

#### Feedback

Correct! For more information, please review the lesson ROC and Precision-Recall Curves.

The correct answer is: The Recall of the classifier would increase or remain the same.