AI Strategy and Digital transformation
1. Introduction

Piotr Wójcik
University of Warsaw (Poland)
pwojcik@wne.uw.edu.pl

January 2025

Section 1

Introduction

Who I am?

- associate professor at the Faculty of Economic Sciences of the University of Warsaw
- the head of the Department of Data Science
- teaching courses: Machine Learning, Statistics, Econometrics, Advanced R programming, Quantitative trading strategies on high-frequency data
- since 2017, head and lecturer of the postgraduate studies "Data Science in business applications"
- 15+ years of professional experience as a quantitative analyst and head of the analytical teams in the financial, telecommunications and marketing research industries
- head of Data Science teams in R&D projects on the edge of academia and industry

Who are you?

Aims of my part of the course

- bring machine learning closer to researchers
- present briefly selected algorithms including the idea of bagging and boosting
- present selected tools of so-called eXplainable Artificial Intelligence (XAI)
- show practical application of **best practices** in ML application in python on real data
- discuss the **most common mistakes** made not only by beginners in data science and show how to avoid them

The most common mistakes

- data transformations (standardization, scaling, imputation, rebalancing) **BEFORE** train/test split or **BEFORE** cross-validation
- data transformations (standardization, scaling, imputation, rebalancing) **AFTER** train/test split BUT independently in the train and TEST data based on their distributions
- using test data to select the best model
- cross-validation of alternative models using **different** random division into folds (in python – different random state)
- applying data rebalancing also on **test dataset**
- using ML algorithms only with **default** values of hyperparameters – assuming same variant is best for all kind of data and problems

Plan of my part of the course

1. Introduction, data preparation
2. Cross-validation and KNN
3. Regularization
4. Support Vector Machine and Regression
5. Tree based models
6. Data rebalancing methods
7. Introduction to eXplainable AI
8. AutoML – PyCaret workshop

Motivation: Increasing availability = wider audience

- recent progress in machine learning comes from **increasing availability of powerful and relatively user-friendly software**
- such software generated interest in the field from non-statisticians, eager to use modern statistical tools to analyze their data
- highly technical nature of statistics **restricted its practical use to experts** in statistics, computer science, and related fields
- in recent years, new and improved software **have significantly eased the implementation** burden for many statistical learning methods
- at the same time, there has been **growing recognition across a number of industries** that data modelling is a powerful tool with important practical applications
- as a result, the field has moved from one of primarily academic interest to a **mainstream discipline**, with an enormous potential audience

The purpose of my course

- I will **NOT** discuss all technical details behind machine learning methods, such as optimization algorithms and theoretical properties
- most users do not need a deep understanding of these aspects to become **informed users** of the various methodologies
- the aim is to focus on **intuitions**, and **strengths and weaknesses** of the various methods
- and present methods which are **most widely** used in **practical applications**
- describe **basic assumptions** and **intuition** together with **trade-offs** behind each of the approaches
- assumption: student **is comfortable with basic mathematical concepts**
- examples will show applications of machine learning methods on **real data**

Suggested literature (interactive links)

- Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani (2017), “Introduction to statistical learning. With Applications in pyhthon”, Springer-Verlag
- Hastie Trevor, Robert Tibshirani and Jerome Friedman (2009), “Elements of statistical learning”, Springer-Verlag

What is machine learning?

- The term **machine learning** is often used interchangeably with **predictive modelling**, **statistical learning**, **pattern recognition** and refers to a vast set of tools for understanding data
- these tools are usually used to build a model whose **main objective** is to provide accurate forecasts on **test data**
- the tools can be classified as **supervised** or **unsupervised**
- **supervised learning** involves building a statistical model for predicting, or estimating, an output based on one or more inputs.
- With **unsupervised statistical** learning, there are inputs but **no supervising output** – grouping variables, clustering observation, customer segmentation

Regression versus classification

- variables can be either **quantitative** or **qualitative**
- **quantitative** (numeric, continuous) variables take on numerical values which have direct interpretation (and are measured in some units, e.g. *m*, *kg*, *hours*, *euros*, *persons*, etc.)
- **qualitative** (categorical) variables take on values in one of K different classes, or categories – can be **nominal** or **ordered**
- we tend to refer to problems with a **quantitative response** as **regression problems**, while those involving a **qualitative response** are often referred to as **classification problems**

Regression versus classification – cont'd

- the distinction is not always sharp
- for example **logistic regression** is used with a qualitative (two-class, or binary) response, so it is used to solve **classification** problems
- most of the modern ML algorithms can be used in **both cases** – for either quantitative or qualitative responses
- it is less important whether the **predictors** are qualitative or quantitative
- most of the models can be applied **regardless of the type of predictors**, provided that **qualitative predictors are properly (re)coded** into numeric variables

Accuracy is a good measure... for balanced data

- **accuracy** is one of the simplest and most common metrics to assess the performance of a **classification** model
- it is a good measure when the data are **balanced** – the frequency of both levels of the binary outcome is comparable
- if these proportions are disturbed this measure will not be correct – in the example below the accuracy is 0.901, recall for 0s (specificity) 0.990, but recall for 1s (sensitivity) only 0.011, precision for 1s only 0.100, and precision for 0s 0.909

		real	
		0	1
predicted	0	900 (TN)	90 (FN)
	1	9 (FP)	1 (TP)

Balanced accuracy and F1

- in this case one should rather use a measure called **balanced accuracy** (called **avg macro recall** in python), which is the arithmetic mean of recall for 1 and 0 (sensitivity and specificity) – for the example above it would be 0.501
- if the data is not balanced (usually it is NOT) and it is equally expensive to incorrectly predict positives as negatives (FN) and negatives as positives (FP), one can also use the measure known as **F1**
- F1 is calculated as the **harmonic** mean of recall and precision:

$$F1 = (\frac{recall^{-1} + precision^{-1}}{2})^{-1} = 2 * \frac{precision * recall}{precision + recall}$$

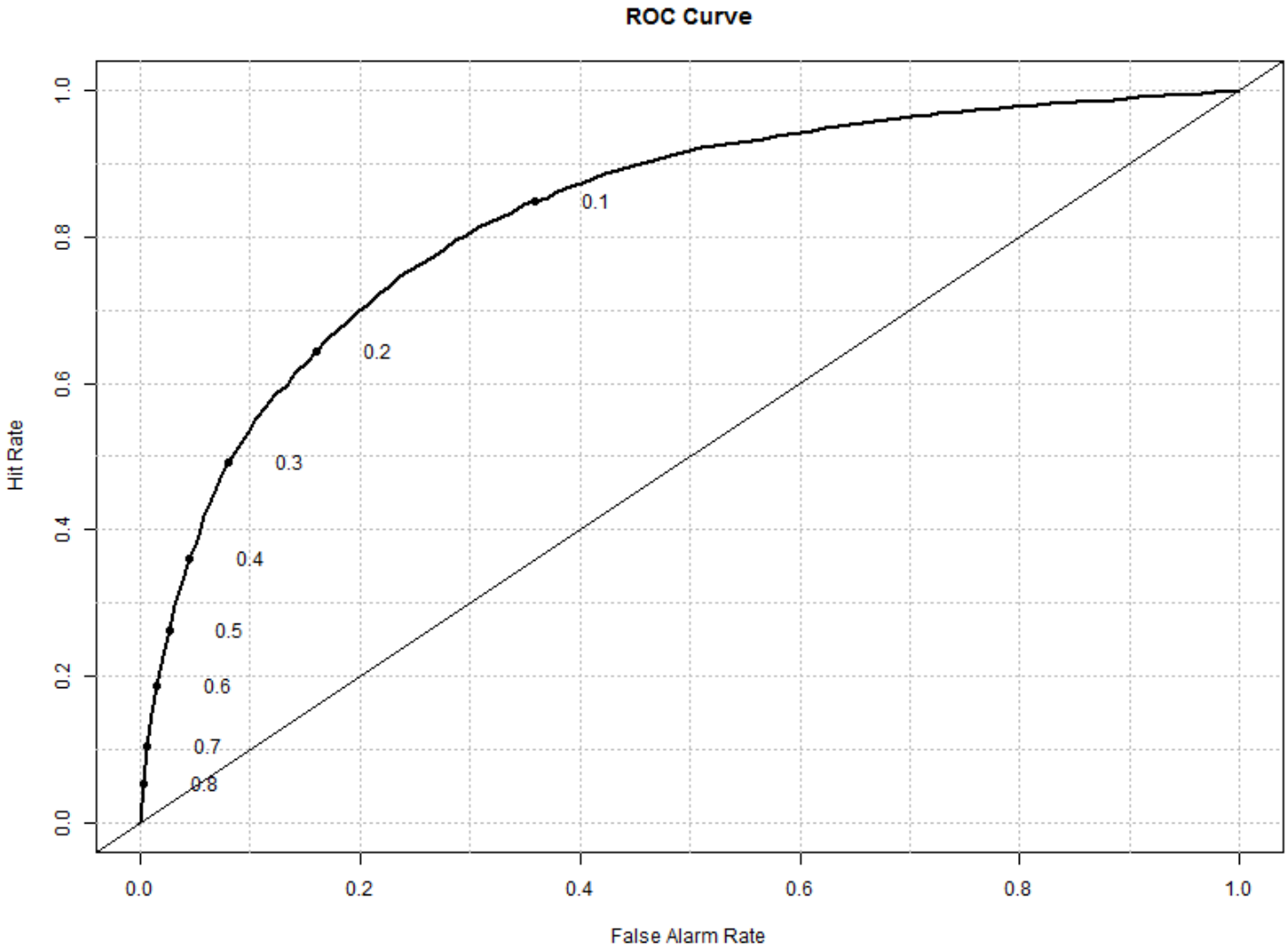
Why F1 is a hamonic mean?

- harmonic mean is used to calculate F1, because it imposes the **strongest punishment** when one of the components of this measure (recall or precision) is low
- suppose we have a model in which precision is 0.95, but recall is only 0.5
- **arithmetic mean** of these values is 0.725
- **geometric mean** of these values is 0.689
- **harmonic mean** of these values is 0.655

ROC curve useful also for imbalanced data

- another way to compare the quality of the classification model, also for non-balanced data is **ROC** curve (**receiver operating curve**) borrowed from the theory of signals
- it compares **sensitivity** and **1 - specificity** for different values of the cut-off point on the graph (each point of the curve is a separate cut-off point)
- the **more concave** the curve (bulged towards the upper left corner), the **better the classifier**
- the measure used for the overall assessment of the model may therefore be the **area under the ROC curve** (**AUC**)

ROC curve – sample plot



Area under ROC curve

- area under the ROC curve is the **averaged measure of the predictive power** for all possible cut-off points
- AUC/AUROC expresses **the probability that based on the model and data a randomly selected real 0 has a lower predicted probability of being 1 than a randomly selected real 1**
- The AUC/AUROC for the random model with no predictive power is 0.5, while for the ideal model it is 1
- it is usually assumed that values above 0.8-0.9 indicate a good model

Introduction – practical exercises in python

Thank you for your attention