Lecture 7: Classifier Evaluation

Statistical Methods for Data Science

Yinan Yu

Department of Computer Science and Engineering

November 20, 2023

Today

Learning outcome

- Be able to calculate and interpret TP, TN, FP, FN, accuracy, precision, recall, specificity, F1 score
- Understand basic concepts of performance evaluation and comparison of different classifiers

Today

1. Training, validation and test
2. Evaluating classifiers
3. Summary

Problem solving in data science

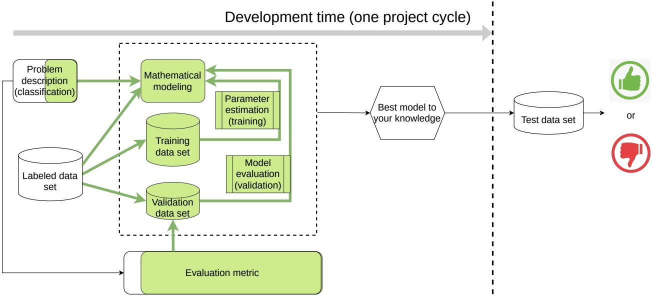

Figure: The (green) shadowed boxes are the actions performed by the data scientist.

Development and testing

During development: split the available dataset into a **training dataset** and a **validation dataset**; these two datasets are used to optimize the performance of the model.

- **Training dataset**: estimate the parameters
- **Validation dataset**: evaluate the performance of one or more classifiers
  What are being evaluated:
  - g (the selection of mathematical models)
    Philosophies:
    - **Occam's razor**: if multiple models are showing similar performances, the simplest model is preferred
    - **All models are wrong but some are useful**
  - $\hat{\theta}$ (parameter estimation method)
  - h (hyperparameter tuning)

After development:

- **Test dataset**: this is the final evaluation. You DO NOT have access to it during the development phase.

How to split the data

Given a labeled dataset $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), (x_6, y_6)\}$, split the dataset into a **training dataset** and a **validation dataset**

- Training-validation split
  - $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)\}$, $\{(x_5, y_5), (x_6, y_6)\}$
- K-fold cross validation, e.g. 3-fold
  - $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)\}$, $\{(x_5, y_5), (x_6, y_6)\}$
  - $\{(x_1, y_1), (x_2, y_2), (x_5, y_5), (x_6, y_6)\}$, $\{(x_3, y_3), (x_4, y_4)\}$
  - $\{(x_3, y_3), (x_4, y_4), (x_5, y_5), (x_6, y_6)\}$, $\{(x_1, y_1), (x_2, y_2)\}$
- Leave-one-out cross validation
  - $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5)\}$, $\{(x_6, y_6)\}$
  - $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_6, y_6)\}$, $\{(x_5, y_5)\}$
  - $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_5, y_5), (x_6, y_6)\}$, $\{(x_4, y_4)\}$
  - $\{(x_1, y_1), (x_2, y_2), (x_4, y_4), (x_5, y_5), (x_6, y_6)\}$, $\{(x_3, y_3)\}$
  - $\{(x_1, y_1), (x_3, y_3), (x_4, y_4), (x_5, y_5), (x_6, y_6)\}$, $\{(x_2, y_2)\}$
  - $\{(x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), (x_6, y_6)\}$, $\{(x_1, y_1)\}$

Today

Validation: four potential outcomes

- Given a **binary classification** problem, a trained classifier $g(x; \hat{\theta} \mid h)$ and a validation dataset containing pairs (x, y), the validation step is to evaluate how "good" the classifier $g(x; \hat{\theta} \mid h)$ is.
- Positive: $y = 1$; negative: $y = 0$
- Compute $\hat{y} = g(x; \hat{\theta} \mid h)$ given x from the validation dataset - there are four potential outcomes:
  - **True Positive (TP):** count(ground truth $y = 1$, classifier output $\hat{y} = 1$)
  - **False Positive (FP):** count(ground truth $y = 0$; classifier output $\hat{y} = 1$)
  - **True Negative (TN):** count(ground truth $y = 0$; classifier output $\hat{y} = 0$)
  - **False Negative (FN):** count(ground truth $y = 1$; classifier output $\hat{y} = 0$)

  Confusion matrix (contingency table)

	$y = 1$	$y = 0$
$\hat{y} = 1$	**TP**	**FP** (Type I error)
$\hat{y} = 0$	**FN** (Type II error)	**TN**

  - What is count$(y = 1)$? (15 sec) **TP**+**FN**
  - What is count$(y = 0)$? (15 sec) **TN**+**FP**
  - What is size of the entire dataset, i.e. count$(y = 1)$+count$(y = 0)$? (15 sec) **TP**+**FN**+**TN**+**FP**

Evaluation metric

- Accuracy:

$$accuracy = \frac{\mathsf{TP} + \mathsf{TN}}{count(y = 1) + count(y = 0)} = \frac{\mathsf{TP} + \mathsf{TN}}{\mathsf{TP} + \mathsf{TN} + \mathsf{FP} + \mathsf{FN}}$$

- True Positive Rate (recall, sensitivity):

$$TPR = \frac{\mathsf{TP}}{count(y = 1)} = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FN}}$$

  Example: There are 10 ducks in total, of which 6 are chonkers ($y = 1$). You correctly identified 5 of these chonkers. Therefore, your recall for spotting chonker ducks is 5 out of 6 (5/6).

- True Negative Rate (specificity):

$$TNR = \frac{\mathsf{TN}}{count(y = 0)} = \frac{\mathsf{TN}}{\mathsf{TN} + \mathsf{FP}}$$

- Precision:

$$precision = \frac{\mathsf{TP}}{count(\hat{y} = 1)} = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FP}}$$

  Example: There are 10 ducks in total. You identified 7 as chonker ducks. But only 3 of them are actual chonkers. Your precision for spotting chonker ducks is 3 out of 7 (3/7).

- F1 score:

$$F = 2 \times \frac{precision \times recall}{precision + recall}$$

- More from scikit-learn:
  `www.scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics`

Evaluation metric (cont.)

- **Use these metrics with caution**, particularly for **imbalanced** datasets, e.g. data from medical measurements typically contain more negative instances (90% healthy volunteers) than positive ones (10% patients)
  - Terrible metric: accuracy
  - Okay metric:
    - Precision vs recall
    - Sensitivity vs specificity
    - F1 score

  Reference: read the data science design manual, section 7.4.1
- In this lecture, we only consider **binary classification** $c \in \{0, 1\}$
- In the multi-class case $c \in \{1, \cdots, C\}$:
  - **Macro**: the metrics are computed for each class c and then the average is calculated
  - **Micro**: the metrics are computed globally for all classes

Evaluate one classifier

Given a classifier $g(x; \hat{\theta} \mid h)$,

- **Goal:** to evaluate the performance of the classifier $g(x; \hat{\theta} \mid h)$
- **Steps:**
  - Step 1: compute $\hat{y}_i = g(x_i; \hat{\theta} \mid h)$ given x_i in the validation dataset $\{(x_1, y_1), \cdots, (x_n, y_n)\}$
  - Step 2: compute an evaluation metric (e.g. page 11), denoted by s
- **Interpretation:**
  - Case 1: one validation dataset (e.g. training-validation split, leave-one-out cross validation) - only one s, e.g. $s = 0.92$
  - Case 2: multiple validation datasets (e.g. K-fold cross validation) - a set of s, e.g. for 3-fold cross validation, we have 3 validation datasets, which gives us a set of 3 scores $\{s_1, s_2, s_3\}$. We can then compute sample statistics (e.g. sample mean, sample standard deviation) from this set and use the sample statistics to evaluate the classifier.

Compare two classifiers

Given two classifiers $g_1(x; \hat{\theta} \mid h)$ and $g_2(x; \hat{\xi} \mid t)$ (note: we use different symbols $\hat{\theta}$ and $\hat{\xi}$ to indicate that they have different estimated parameters; likewise, h and t indicates the respective hyperparameters)

- **Goal:** find out which classifier is better, $g_1(x; \hat{\theta} \mid h)$ or $g_2(x; \hat{\xi} \mid t)$
- **Steps:**
  - Step 1: compute $\hat{y}_i^1 = g_1(x_i; \hat{\theta} \mid h)$ and $\hat{y}_i^2 = g_2(x_i; \hat{\xi} \mid t)$ given x_i in the validation dataset $\{(x_1, y_1), \cdots, (x_n, y_n)\}$
  - Step 2: compute an evaluation metric (e.g. page 11) for each classifier. Let's call them s^1 and s^2.
  - Step 3: compare s^1 and s^2 to determine which classifier is better, $g_1(x; \hat{\theta} \mid h)$ or $g_2(x; \hat{\xi} \mid t)$.
    Example: say we choose the F1 score as the evaluation metric
    - Question: if $s^1 = 0.92$ and $s^2 = 0.52$, which classifier is better, $g_1(x; \hat{\theta} \mid h)$ or $g_2(x; \hat{\xi} \mid t)$?
    - Answer: probably $g_1(x; \hat{\theta} \mid h)$ since $s^1 \gg s^2$
    - Question: if $s^1 = 0.92$ and $s^2 = 0.91$, now which classifier is better?
    - Still $g_1(x; \hat{\theta} \mid h)$ with $s^1 \approx s^2$?

    We can use **hypothesis testing** to quantify the comparison between two classifiers.
    - Case 1: one validation dataset (e.g. training-validation split, leave-one-out cross validation) - **McNemar's test**
    - Case 2: multiple validation datasets (e.g. K-fold cross validation) - **paired t-test**

Today

1. Training, validation and test
2. Evaluating classifiers
3. Summary

Summary

So far:

- Data types and data containers
- Descriptive data analysis: descriptive statistics, visualization
- Probability distributions, events, random variables, PMF, PDF, parameters
- CDF, Q-Q plot, how to compare two distributions (data vs theoretical, data vs data)
- Modeling
- Parameter estimation: maximum likelihood estimation (MLE) and maximum a posteriori estimation (MAP)
- Classification, multinomial naive Bayes classifier, Gaussian naive Bayes classifier

Next:

- Another application of probablistic models in machine learning: Gaussian Mixture Models (GMM)

Before next lecture:

- Gaussian distribution, Bayes rule, likelihood function