DataSci 400

lesson 9: performance measures

Seth Mottaghinejad

today's agenda

- performance measures for **classification**
  - problems performance measures in classification
    - **hard vs soft** classifiers
    - dealing with **class imbalance**
  - **binary** classification
  - **multi-class** classification
- performance measures for **regression**

hard vs. soft predictions

- **hard classifiers** only return the prediction for the class
- **soft classifiers** return the "probability" or confidence that prediction is in each class
  - example: 3 classes - cat, dog, squirrel
  - **hard prediction**: dog, **soft prediction**: [0.12, 0.84, 0.04]
- most classifiers return **soft predictions**
- we can later set a **threshold** and obtain **hard predictions**
- most performance metrics depend on the **choice** of the threshold

class imbalance

- also sometimes referred to as **rate events** scenaro
- class imbalance is very common in many use-cases:
  - fraud detection (binary classification)
  - medical diagnosis (binary classification)
- class imbalance usually implies that not all errors (misclassification) should have the same importance
  - looking at accuracy (percent misclassifications) can be optimistic
  - instead we look at other metrics, like precision, recall, or AUC

the confusion matrix (two-class)

- it's really not that confusing!

	predicted positive	predicted negative
actually positve	true positive `TP`	false negative `FN`
actually negative	false positive `FP`	true negative `TN`

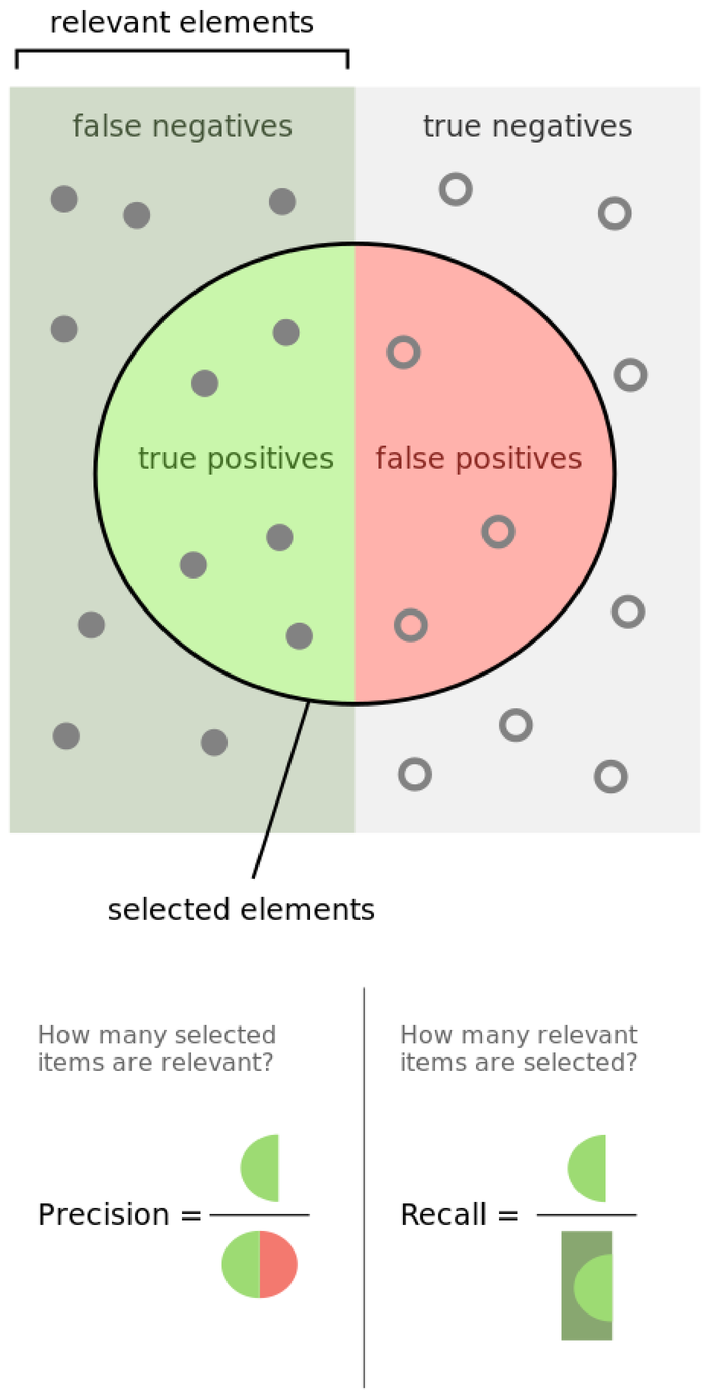
- for `TP` / `FP` / `TN` / `FN`
  - the second letter indicates what the prediction was, and
  - the first letter indicates if the prediction was right or not

lab time

- we saw there a binary classification model can make two kinds of errors: `FP` and `FN`
- for the following scenarios, say what kind of error is more costly (use common sense)
  - **credit card fraud detection**: someone impersonates you to use your credit card
  - **medical diagnosis**: finding out who has a disease
  - **information retrieval**: finding relevant web pages based on a search query

precision and recall

- **accuracy** is just the misclassification rate
- **precision** is the percentage of positive predictions that were actually positive
- **recall** is the percentage of positive cases that we correctly predicted as such
- image source: [wikipedia]



accuracy, precision and recall (two-class)

- accuracy $= \dfrac{TP + TN}{TP + FP + TN + FN}$
- precision $= \dfrac{TP}{TP + FP}$
- recall $= \dfrac{TP}{TP + FN}$
- for rare events usually `TN` far exceeds `TP`, `FN`, or `FP`, **inflating accuracy**, but precision and recall don't have `TN` in it

lab time

here's an analogy that to why we should evaluate a classification model's accuracy using **both** precision and recall:

- when you stand witness in a court of law, you are asked to tell the truth:
  - **the whole truth**: no lie of omission
  - **nothing but the truth**: no lies
- relate the above two statements to precision and recall

break time

the confusion matrix (multi-class)

we use 3-classes here **as an example**, but applies to any class size C

actual \ predicted	pred class 1	pred class 2	pred class 3	actual total
actual **class 1**	TP_1	FP_{12}	FP_{13}	A_1
actual **class 2**	FP_{21}	TP_2	FP_{23}	A_2
actual **class 3**	FP_{31}	FP_{32}	TP_3	A_3
predicted **total**	P_1	P_2	P_3	N

micro averages (multi-class)

- **accuracy** $= \frac{\sum_i TP_i}{N}$
- micro average **precision** $= \frac{\sum_i TP_i}{\sum_i P_i}$ is the **same as accuracy** unless predictions are not mutually exclusive
- micro average **recall** $= \frac{\sum_i TP_i}{\sum_i A_i}$ is the **same as accuracy** unless predictions are not collectively exhaustive
- micro average precision and recall are only relevant in **multi-label** classification

macro averages (multi-class)

- **accuracy** $= \frac{\sum_i TP_i}{N}$
- **macro averages** use **one-vs-all** to find precision and recall for each class and then averages them
- macro precision $= \frac{1}{C} \sum_i \frac{TP_i}{TP_i + FP_i}$
- macro recall $= \frac{1}{C} \sum_i \frac{TP_i}{TP_i + FN_i}$
- C is the class size
- we can use a **weighted average** if we want to emphasize certain classes

break time

performance measures for regression

- measuring performance in **regression** is more straight-forward
- **errors** or **residuals** are the difference between the model's predicted value and actual value (ground truth) for each row
- we use the **hat notation**: $\hat{Y}_i$ is the **prediction** at ith row and Y_i is the **actual**, so $Y_i - \hat{Y}_i$ is the **error** or the **residual**
- a good model should **minimize error** on the test data, and the error should look like **random noise** (nothing left to predict)

performance measures for regression

- **root mean squared error:** $\text{RMSE} = \sqrt{\frac{1}{N}\sum_i (Y_i - \hat{Y}_i)^2}$
- **mean absolute error:** $\text{MAE} = \frac{1}{N}\sum_i |Y_i - \hat{Y}_i|$
- **coefficient of determination:** R^2
  - is 1 minus the ratio of $\sum_i (Y_i - \hat{Y}_i)^2$ over $\sum_i (Y_i - \bar{Y})^2$ where $\bar{Y}$ is the mean of Y
  - represents the proportion of variation **explained away** by model
- **adjusted** R^2: lowers R^2 in proportion of the number of features (discouraging us to just keep adding useless features)

the end