Machine Learning
for
Detection of Pediatric Otitis

Corrado Lanera

corrado.lanera@unipd.it

University of Padova
Unit of Biostatistics, Epidemiology, and Public Health
Department of Cardiac, Thoracic, Vascular Sciences, and Public Health.

2019/06/20 - HLP Lab @UPenn

My research at UBEPH

MLT in clinical environments

Structured data (AKA *phenomapping*)

- Statistical analyses for clinical research

- Clustering patterns for genetic counting data

Unstructured data

- Patient kinetic data from wearable devices

- Free-Text mining from electronic medical records

Outline

A brief introduction to **PEDIANET**

Experiences at UBEPH on health-related free-text analyses

1. Case detection of Varicella Zoster Virus infections in Italian children
2. Extend systematic review searches from literature to registries
3. Classification of pediatric emergency department discharging notes

The otitis project

1. Data
2. Task
3. Challenges

Machine learning strategy proposal

1. Pre-processing
2. Weights and features definition/selection
3. Class imbalance and learning algorithms
4. Training flow
5. Test
6. Performance evaluation

PEDIANET

A pediatric general practice research database

- A network of more than 450 family pediatricians distributed throughout Italy (120-130 providing data)

- Data about 100,000 children

The system will be designed to create a database that can execute queries so that we can address the following queries:
- the frequency and reasons for accessing healthcare
- the frequency of prescriptions (pharmaceutical, specialist appointments, diagnostic investigations, hospital admissions) and pharmacovigilance outcomes

For each child seen, the participating pediatricians send information electronically in an anonymous format including:
-reason for accesing healthcare
-personal details
-diagnosis and clinical details
-prescriptions (pharmaceutical, specialist appointments, diagnostic procedures, hospital admissions)
- growth parameters
- outcome

http://pedianet.it/en/general-design

Experiences at UBEPH

#1 Varicella Zoster (*chickenpox*) detection
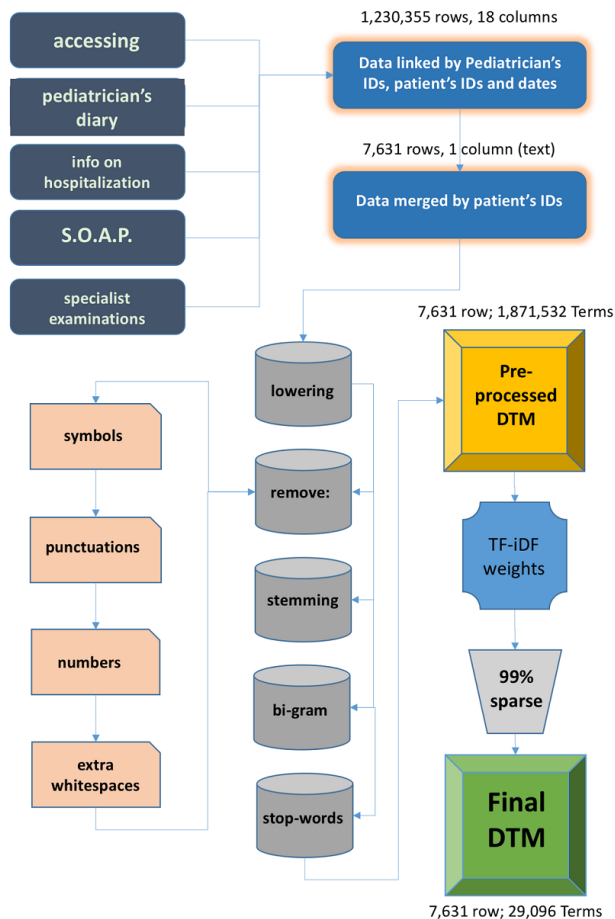
- **Train and validation**
  - region: Veneto (IT)
  - patients: 7,631
  - records: 1,230,355

- **Test**
  - region: Sicilia (IT)
  - patients: 2,347
  - records: 569,926

- **Learners**[1]
  - LogitBoost (F: 68.5 [59.3–77.7])
  - GLMNet (F: 36.5 [32.2–40.8])
  - Maxent (F: 19.1 [17.2–20.9])

[1] Performance on the test set reported.

*L., Berchialla, Baldi, et.al, *JMIR* [under review]

#1 insights

- Single occurrence in a lifetime (time-independent)
- Always test on unseen data in the training set
- Impact of the pre-processing (most impact from lowering and 2-gram)
- Value of bootstrap-based learner

#2 Extend Systematic Review to registries

Assumption: The validity of Systematic Reviews depends on the ability to fully capture the complete body of evidence through searches of many heterogeneous data sources.

Baudard et al. (2017) **"Impact of Searching Clinical Trial Registries in Systematic Reviews of Pharmaceutical Treatments: Methodological Systematic Review and Reanalysis of Meta-Analyses."** *BMJ 356*

- Increase in the number of patients: **from 10% to 50%**
- Change in summary statistics: **from 0% to 29%**

Our aim: to replace complex interfaces for researchers with Text Mining of available textual fields in clinical registries

Our results

- sensitivity: 100% in all SRs except one (87.5%)
- specificity: ranged from 97.2% to 99.9%
- ROC: from 93.4% to 99.9%.

Main issue

MLTs are generally biased towards the majority class samples

#2 Extend Systematic Review to registries

Data[1]

- 14 Systematic Reviews
- **Train** (from PubMed, overall)
  - 72,000 negative
  - 185 positive
- **Test**:
  - 233,609 (from ct.gov)

[1] Baudard et al. (2017)

Learners

- GLMNet
- Support-Vector Machine (SVM)
- Random Forests (RF)
- k-Nearest Neighbor (k-NN)
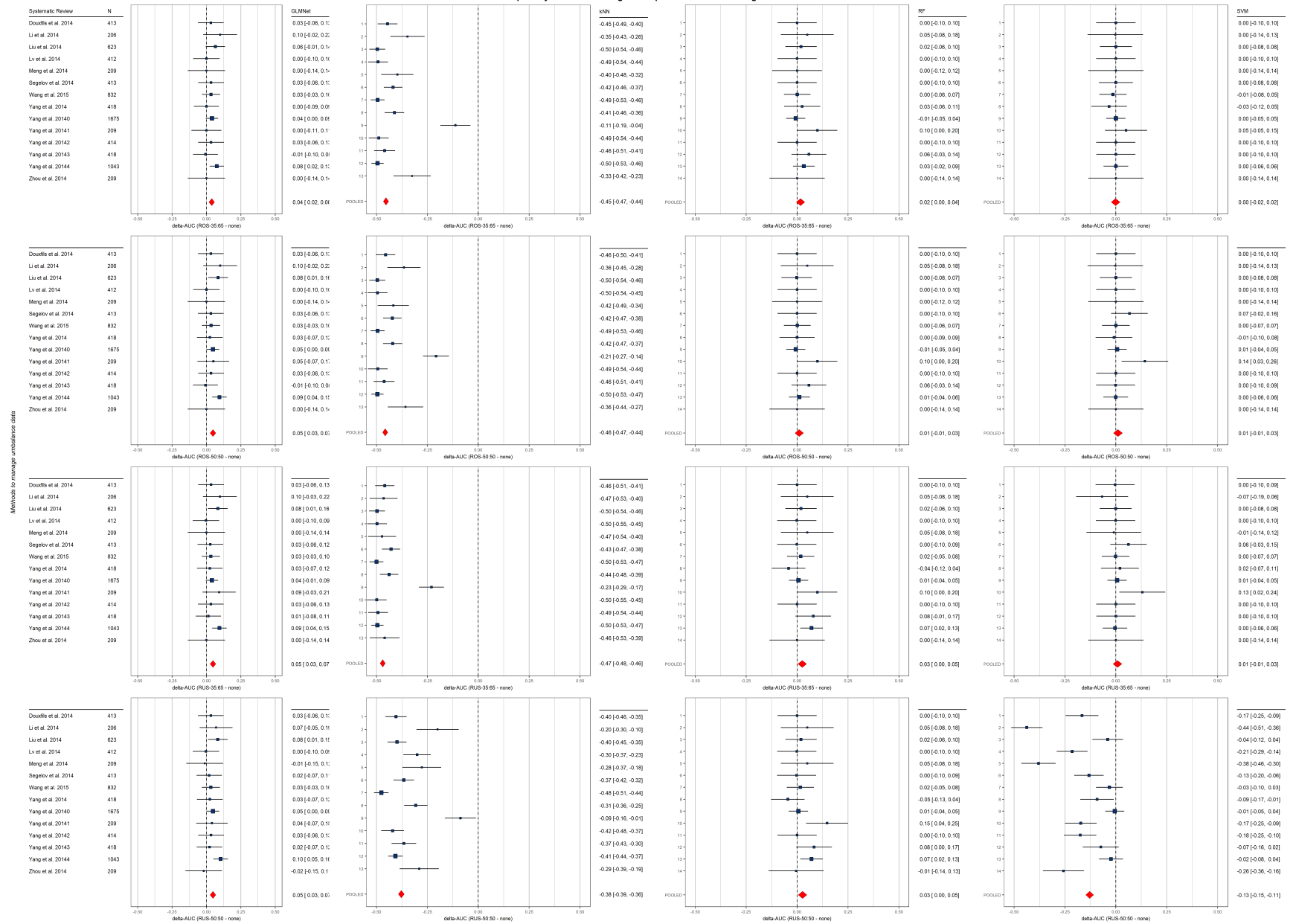
Strategies[2]

- Random Under Sampling (RUS)
- Random Over Sampling (ROS)

- 35:65 minority:majority ratio
- 50:50 minority:majority ratio

[2] Compared with the straight use of full data-set

Results[3]

- Improve: RUS-35:65, ROS-50:50
- Worsen: k-NN (all strategies)
- Almost neutral: SVM
  - (RUS-50:50 worsened it)

[3] Based on ΔAUC respect to the application on the full data-set w/out dealing with class imbalance

Forests' plots by Machine Learning Technique and Method to manage umbalance data

Insights #2

- SVM could be the first choice for a fast and performing MLT
- RUS 35:65 can be useful to reduce the time (and the space) gaining AUC
- Class imbalance strategies drastically worsen k-NN models

#3 Diagnoses classification for children

Data

- 1789 ED visits with reported discharge diagnoses (Free-Text) from 9 Nicaraguan hospitals
- Diagnoses were manually revised and classified by an independent peer-review group of expert pediatricians

Learning strategy

- Bootstrap samples: distribution of results
- Repeated CV: over-fitting control
- Out-Of-Bag (OOB) performance: full data used
- RF: 500 trees each forest (to convergence)

Results

- Overall accuracy: 78.3% [77.9%-79.6%]

Insights #3

- Lemmata extraction (instead of stemming) improve results and interpretation
- Value of a bootstrap superstructure
  - strong internal validation
  - data-driven, non-parametric confidence intervals
  - complete use of all the labeled data at the disposal

Detection and Classification of Otitis from PEDIANET free-text medical notes

#1 Data

During 2018, PEDIANET team search the DB for AOM treatments.

They searched only from the "primary diagnosis" field:

- ICD-9 codes
- search string

The primary limitation of the study was the impossibility of manually validate the cases, possibly including False Positive or detecting cases reported only in the "diary" field.

For the current project, they have provided us an extraction of the DB after the same search string filtering, on all diagnoses, sign-and-symptoms, and diaries free-text fields.

> Whatever excluded is not considered a otitis case (label = 0).

#1 Data

- **Textual** variables[1]

  - diagnosis 1-3
  - sign-and-symptoms 1-3
  - diary 1-4
  - prescription 1-8
  - visit descriptions 1-8
  - visit results 1-8

- **Patients**: 4,475

- **Records**

  - 297,373 overall (2004-2017)
  - 4,928 train (2004-2007)
  - 723 dev (2008-2017)
  - 880 test (2008-2017)

- **Structured** variables
  - patient's id (by pediatrician)
  - patient's gender
  - patient's date of birth
  - pediatrician id
  - date and time of the visit

[1] Italian language.

#2 Task

Classification of patients' records into six hierarchical classes

- 0 = not an otitis case (i.e., false-positive on the search-string filter)
- 1 = otitis case, not media
  - 2 = otitis media (OM), not acute
    - 3 = acute otitis media (AOM), not recurrent nor with perforation
      - 4 = AOM with perforation
      - 5 = AOM recurrent

Notes:

- Label 4 and label 5 can coexist
- Label 5 is time-dependent: three OMA events within six months or 4+ OMA events in 12 months
- Label 5: if the pediatrician explicitly reports that OMA is recurrent, the record has to be marked as well despite the timing

#3 Challenges

- Patients cannot be identified across pediatricians

- Multiple text-style (i.e., different pediatricians): tags?

- Hierarchical models: sequential multi-stage?

- Possible multi-label classification: parallel independent stages?

- Time-dependent classification: post-process layer?

- All stages are possibly affected by data imbalance

Machine learning strategy proposal

#1 Pre-processing

- **Removing**

  - extra white spaces
  - non-words
  - stop-words

- **Merging**

  - lowering
  - lemmata extraction

#2 Weights and features def./sel.

- **Features augmenting**

  - n-gram $(n \in \{1, 2\})$

- **Features enrichment** (tags)[*]

  - POS
  - pediatrician id

- **Weighting strategy**

  - TF-iDF

- **Feature selection**

  - based on the TF-iDF rank
  - 80%-20% Pareto principle

- **Diagnoses attribution** (labels 0/1)
  - could be it discovered by augmentation and enrichment only?

[*] Disclaimer: I have **no** experience about tagging.

#3 Class imbalance and learning algorithms

- **Type**

    - RUS 35:65 (first, because it's cheaper)

    - none

- **Learners**

    - GLMNet (first, *benchmark*)

    - SVM

    - RF[*]

---

[*]The `ntree` into forests will be always checked to guarantee the predictions are stable.

#4 Training flow

One model (and subset of data) per stage, trained independently

1. **Training phase: identification**

   - 500 bootstrap superstructure
     - test on OOBs
   - parameter selection
     - 10 x each hyper-par[1]
     - 5 rep of 10-fold cross-validation (CV)
   - best model selection
   - further evaluations for 100 incremental train subsets[2]
     - learning curve examination

   [1]Randomly chosen.

   [2]Best model and hyper-par set only. CV excluded.

2. **Dev phase: tailoring[2]**

   - bootstrap superstructure on the dev set
     - test on OOBs
   - parameter selection
     - 10 x each hyper-par[3]
     - 5 x 10-fold dev-CV[4]
   - best hyper-pars
   - further evaluation for 100 incremental dev subsets[2]
     - learning curve examination

   [3]Nested around the phase-one best selection.

   [4]Added to the full training set.

5# Test

3. **Test phase: final evaluation**:

- 100 models trained on incremental train+dev subset[*]

- 500 bootstrap estimation on the test set each model

- learning curve examination

[*]Best algorithm and hyper-par for the dev set.

#6 Performance evaluation

- Learning curves analyses

  - more data Vs. more flexibility

- Stage-chain performance analyses

  - for each stage from the second to the fourth, compare the performance obtained by the model on *its own gold test set*, i.e., the one consisting of the records corresponding to 100% correct previous stage prediction

  - the Δ from *own-gold-* to *chain-* performances will be examined and reported for future improvement

Limitation

Train, dev and test sets distributions are biased from the full **PEDIANET** 's DB by the search string matching

Direct application of the model to the full **PEDIANET** can behave unexpectedly on the search-string-complementary set

Thank you
for your attention

Slides created via the R package **xaringan**
powered by remark.js, **knitr**, and R Markdown.