Influence of hyperparameters on aggregating predictions of infinite number of experts

Sergey Kunin-Bogoiavlenskii

Moscow Institute of Physics and Technology

Course: My first scientific paper (Strijov's practice)/Group 125

Expert: R. D. Zukhba

Consultant: A. V. Zukhba

2024

Goal of research

> Prediction is very difficult,
> especially if it's about the future.
>
> *Niels Bohr*

Goal

Examining the influence of hyperparameters on the accuracy of the aggregation algorithm with an infinite number of experts

Targets

1. Time series generator implementation
2. Aggregating algorithm implementation
3. Experiments with various hyperparameters

Literature

- V. V'yugin, V. Trunov. 2023. Prediction of Locally Stationary Data Using Prediction with Expert Advice.
  http://www.jip.ru/2023/470-487-2023.pdf
- N. Cesa-Bianchi, G. Lugosi. 2006.
  Prediction, Learning, and Games.
  https://ii.uni.wroc.pl/~lukstafi/pmwiki/uploads/AGT/Prediction_Learning_and_Games.pdf.
- Hyndman, R. J. & Athanasopoulos, G., 2nd edition. 2018.
  Forecasting: Principles and Practice.
  https://otexts.com/fpp2/

Problem statement

> There are two kinds of forecasters: those who don't know, and those who don't know they don't know.
>
> *John Kenneth Galbraith*

Data

It is assumed that there are multiple generators, whose structure is unknown to the predictors. These generators switch, producing a time series that is subdivided into a sequence of segments - areas of stationarity, which can be studied using machine learning methods.

Gerators implemented:

- Linear
- ARMA

Problem statement

Terms

- X — signals space
- Y — responses space
- $\mathcal{N}$ — set of experts, indexed by natural numbers
- D — desicion space, to which predictions belong
- $\lambda : \mathrm{D} \times \mathrm{Y} \to \mathbb{R}_+$ — nonnegative loss function
- $L^i_T = \sum\limits_{t=1}^{T} l^i_t$ — cumulative loss of expert i during the first T steps
- $H_T = \sum\limits_{t=1}^{T} h_t$ — master's cumulative loss during the first T steps
- $R^i_T = H_T - L^i_T$ — master's regret relative to the expert i

Problem statement

Algorithm

FOR $t = 1, 2, \ldots :$

1. Expert f^t initialization
2. Experts' predictions $f_t^i = f_t^i(x_t),\ 1 \le i \le t$
3. Master's prediction evaluation $\gamma_t = \mathrm{Subst}(\mathbf{f_t}, \widehat{\mathbf{w}}_\mathbf{t})$
4. Computation of master's loss $h_t = \lambda(p_t, y_t)$ and experts' losses l_t^i
5. **Loss Update** weights modification
6. **Mixing Update** weights modification

ENDFOR

Experiments

Initialization weights

Default weights: $w_1^i = \frac{1}{(i+1)\ln^2(i+1)}$

Experimental: $\frac{1}{i^\alpha}$, $\frac{1}{(i+1)\ln(i+1)\ln^2\ln(i+c)}$, $\frac{1}{e^i}$, etc.

Window size

As the algorithm does not know the locations of generator switches, finding an optimal training window is also a challenge.

Mixing update coefficients

Default coefficient: $\alpha_t = \frac{1}{t+1}$

Experimental: $\frac{1}{(t+1)^\beta}$, $\frac{1}{t+c}$, etc.

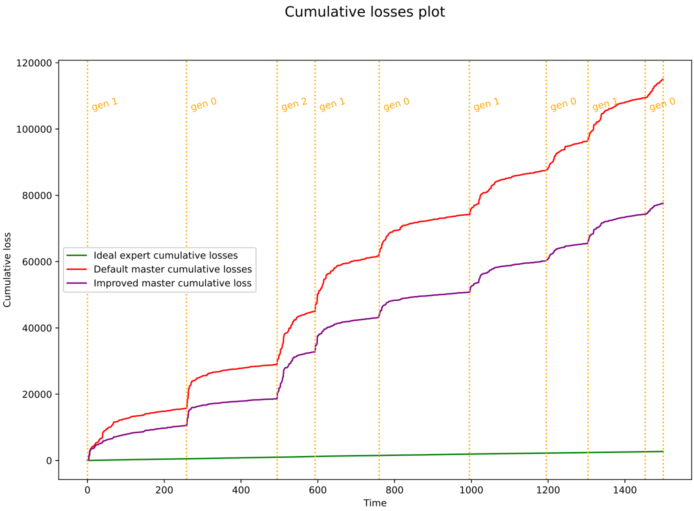
Metric - master's regret relative to the ideal expert (best partition)

Table

train_window	weight_function	regret
5	$1/((i+1) * ln^2(i+1))$	224307.80
5	$1/(i^{1.01})$	175677.06
10	$1/((i+1) * ln^2(i+1))$	142566.48
10	$1/(i^{1.01})$	98022.98
20	$1/((i+1) * ln^2(i+1))$	135308.02
20	$1/(i^{1.01})$	93522.27

Table: Regret for different weight functions

Losses plot


Cumulative losses plot
Cumulative loss
Time
gen 1
gen 0
gen 2
gen 1
gen 0
gen 1
gen 0
gen 1
gen 0
Ideal expert cumulative losses
Default master cumulative losses
Improved master cumulative loss
120000
100000
80000
60000
40000
20000
0
0
200
400
600
800
1000
1200
1400

Conclusion

Summary

- Generators and algorithm implemented
- Correctness of the algorithm veryfied
- A series of experiments conducted
- Enhanced weight function achieved

Further plans

- Run experiments on real data
- Theoretically prove that the obtained function is the best.

The End