**Applied Machine Learning**

**Final project**

Ortal Parpara 311127005

Yiftah Szoke 204646913

**The article:**

"Deep Incremental Boosting"

**Stage 1- The algorithm:**

1. **A description of the algorithm:**

The algorithm presented in the article is called Deep Incremental Boosting(DIB), which combines the power of AdaBoost, Deep Neural Networks and Transfer of Learning principles, in a Boosting variant which can improve generalization.

1. **The advantages:**

* The algorithm reduces the required training time.
* The algorithm improves generalization.

1. **The disadvantages:**

* The network could be more complexity.

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

**Stage 2- Suggesting an improvement:**

To improve the algorithm performance there are few ways we can do that:

* Data- get more data, features selection, diversity, etc.
* Algorithm- change the algorithm, combination of algorithms, etc.
* Tuning- **network topology**, weights, learning rate, early stopping, etc.
* Ensembles- combine models.

we chose to improve the performance by using network topology- network structure.

the difficult with this way to improve is to choose the right network and layers which could improve the performance.

To choose a good network for our problem, we needed to discover a good configuration, to do that, we need to experiment few options:

* Try one hidden layer with a lot of neurons (wide).
* Try a deep network with few neurons per layer (deep).
* Try combinations of the above.
* Try architectures from recent papers on problems like yours.
* Try topology patterns (fan out then in) and rules of thumb from books and papers (see links below).

after considering the above options, we build a network.

**The network:**

Attached to the project: 'densenet.model'.

**Stage 3- A well-known algorithm for comparison:**

Algorithm for comparison: **Adaboost**.

Adaboost (Adaptive Boosting)- It can be used in conjunction with many other types of learning algorithms to improve performance. The output of the other learning algorithms ('weak learners') is combined into a weighted sum that represents the final output of the boosted classifier. AdaBoost is adaptive in the sense that subsequent weak learners are tweaked in favor of those instances misclassified by previous classifiers.

**Stage 4- Evaluating the algorithms:**

1. **Versions:**

We chose the Adaboost Algorithm to evaluate and compare to our improvement.

1. **data:**

Classification task.

**Stages:**

* read 20 datasets (classification).
* data preparation- fill missing data, split the data.

1. **evaluation+** **Hyperparameter Optimization:**

External 10-fold Cross Validation and Internal 3-fold Cross Validation (for optimization).

**Stages:**

* Define the models.
* Create the models using external and internal cross-validation.
* Choose the best parameters for the models.

1. **Performance metrics for evaluation:**

In attached excel.

**Stage 5- Statistical significance testing of the results:**

Choose one of the performance metrics (AUC for example) and use the Friedman test as was presented in the lecture to determine whether the differences are statistically significant. If the results are statistically significant (i.e., the null hypothesis is rejected), do a Post-Hoc test to test the differences between the algorithms. Report and analyze your results and conclusions, <https://www.statology.org/friedman-test-python/>

**Friedman test** used to detect differences in treatments across multiple test attempts. The procedure involves [ranking](https://en.wikipedia.org/wiki/Ranking) each row together, then considering the values of ranks by columns.

**Hypothesis**: the several treatments have the same distributions.

**Friedman test** **- 5 stages:**

1. Name the number of treatments- k and blocks- n.
2. Rank the data within each block (rank the treatment outcomes for each patient).
3. Add the ranks for each treatment separately: name the sums T1, T2,…, Tk.
4. Calculate the Friedman Fr statistic, which is distributed as chi-square, by:
5. Obtain the p-value (as if it were α) from Table III (χ2right tail) for k –1 df.

We chose to use the AUC performance metric to use the Friedman test.

**The results were:**

Friedman test

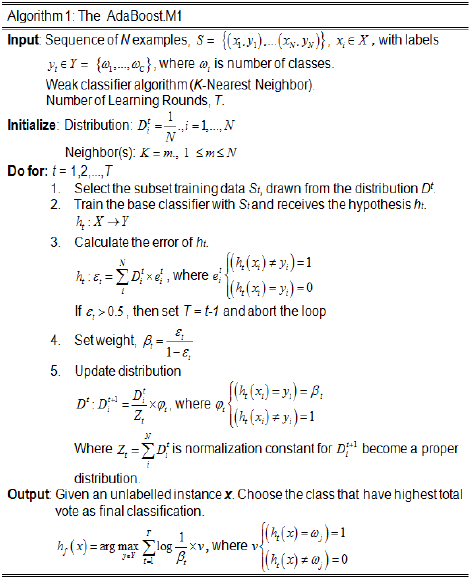
Statistics=16.800, p=0.00022

Different distributions (reject H0)

**Stage 6- Conclusions:**

Present the three algorithms you are evaluating (stages 1-3). Include the pseudo code, results and conclusions.

**Adaboost:**



**DIB:**

pseudo code:תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

model:

**Improved DIB:**

pseudo code:

model:

**Results:**

**Conclusion:**