General Topic:

Automated classification of data into relevant categories: A machine learning approach

Logbook 1: [identification of hypothesis]:

Build a classifier that classify hand written digits into 10 (0,1, 2,..,9) different classes.

Logbook2: [statistical analysis]:

Build a classifier that classifies patient is diabetic or not.

Logbook 3: [optimization technique]:

Compare different optimization technique to choose which one is best for classification problem.

Logbook 4 [simulation technique or data analysis]:

Build a classifier that classifies human organs into 5 different classes (neck, lungs, liver, pelvis, legs).

[**NOTE**: for logbook 4 focus should be on simulation technique or data analysis. I did data analysis as to build any kind of classifier, we do not need to use any kind of simulation technique.]

Research Logbook #1

Research Topic: Comparisons on KNN, SVM, BP and the CNN for Handwritten Digit Recognition

Logbook Focus

Identification of hypothesis and parameters of data analysis

Research Paper:

W. Liu, J. Wei and Q. Meng, "Comparisions on KNN, SVM, BP and the CNN for Handwritten Digit Recognition," 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications(AEECA), Dalian, China, 2020, pp. 587-590, doi: 10.1109/AEECA49918.2020.9213482.

Problem Statement

- Do computers have the same ability to recognize handwritten digits as humans do?
- Humans can new thing through summarizing and categorizing things. since AI have ability to recognize pattern, can it recognize handwritten digits precisely as humans?
- Even though we have to classify handwritten digits into 10 different classes, not everyone uses the same way to write any digit. [1, pg.587]
- There are many algorithms we can use to identify handwritten digits but we need to choose which one is the best to solve this handwritten digit recognition problem.

General Points

- This whole classification problem consist 5 parts[1, p.587]
    - Collect sample
    - Data cleaning or preprocessing
    - Feature selection
    - Build classifier based on features we choose
    - Decision making based on classifier
- This paper takes the MNIST dataset which contains thousands of handwritten digits. In this paper, they use around 70% data for training and remaining for the testing. [1, pg.590]

- To conclude CNN algorithm has highest accuracy while testing the data [1, pg.590]

- .

Algorithm Analysis

KNN:

- KNN basically works on geometric measurement principle.[1]
- This algorithm predicts the class for test data by calculating distance between test data and training data point. Suppose we need to predict class for any data X . Then to check in which class that data belongs, in the training data distribution, we need to check the nearest class for data X, and assign the nearest data class to that data X.
- Requires a lot of memory and calculation.[1, pg.588]

SVM:

- Binary classification [1, pg.588]
- Find the best feature space separation hyperplane. [1, pg.588]
- Dimension transform (convert lower dimension data into higher dimension) to tackle nonlinear data problems. [[1, pg.588]]

BP neural network:

- To minimize error, use steepest descent optimization method to track feedback values in backward propagation. [[1, pg.588]]
- Use a combination of input layer, output layer and hidden layers to classify data. [1, pg.588]

CNN

- Use convolutional layer, pooling layer, fully connected and Softmax layer to build a network which classify handwritten digits into their respective classes. [1, pg.589]
- Use shared weight to reduce error in backward propagation and tackle the problem of overfitting. [1, pg.589]

Performance Criteria:

- Maximize accuracy of testing data
- Choose the best algorithm
- Best recognition rate

Definitions

CNN – convolution neural network

KNN - K-Nearest Neighbors Algorithm

SVM - Support Vector Machine

Keywords

MNIST, handwritten digit data, KNN, CNN, SVM, BP neural network. [1, pg.587]

References

[1] W. Liu, J. Wei and Q. Meng, "Comparisions on KNN, SVM, BP and the CNN for Handwritten Digit Recognition," 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications(AEECA), Dalian, China, 2020, pp. 587-590, doi: 10.1109/AEECA49918.2020.9213482.

https://ieeexplore.ieee.org/document/9213482

Research Logbook #2

Research Topic: Supervised Machine Learning Algorithms: Classification and Comparison

Logbook Focus: statistical analysis

Research paper:

Akinsola, J E T. (2017). Supervised Machine Learning Algorithms: Classification and Comparison. International Journal of Computer Trends and Technology (IJCTT). 48. 128 - 138. 10.14445/22312803/IJCTT-V48P126.

Null hypothesis:

Accuracy of classifying patient is diabatic or not is same regardless of type of machine learning algorithms is used.

Alternative hypothesis:

Accuracy of classifying patient is diabatic or not is not same for all the machine learning algorithms.

Data Collection:

- The research data was obtained from National Institute of Diabetes and Digestive and Kidney Diseases which was made available online at University of California. [1.pg 132]
- To check accuracy of 7 algorithms, in this paper, two datasets has been discussed. one is large dataset with more attribute and another is small dataset with less attribute. [1.pg 132]

Statistical review:

Performance on large dataset:

- The Kappa statistic for each algorithm is higher than the MAE (mean absolute error) statistic.[1.pg 133]
- SVM performs the best with the highest precision and accuracy while decision tree has lower accuracy. [1.pg 133]

Performance on small dataset:

- Neural Networks (Perceptron), JRip, Decision Tree (J48) shows lower Kappa statistic compared to MAE(mean absolute error). [1.pg 134]
- SVM classified 72.91% data correctly while Neural Networks (Perceptron) run with only 59% accuracy. [1.pg 134]

Ranking of precision of positive and negative diabetes:

- SVM has highest precision for both positive and negative prediction for both large and small data. [1.pg 135]
- For small and large dataset, Neural Networks and decision tree shows lowest precision rate respectively. [1.pg 135]

Correctly and incorrectly classified:

- For both large and small dataset, SVM is advisable as it classifies data with highest accuracy. [1.pg 135]

- While for small dataset Neural Networks only classified 59% data correctly, same algorithm work with 75.13% accuracy for large dataset. [1.pg 135]

Descriptive Analysis of various Dataset attributes:

- For a small dataset, the attributes with the highest mean (plasma glucose concentration) and lowest mean (diabetes pedigree function) had the most influence. [1.pg 136]

Table to find which algorithm take less and more time to build model:

For large dataset, SVM has fastest model building time while for small dataset, Naïve Bayes and JRIP are fastest. On the other hand Neural Networks (Perceptron) has slowest model building time for large and small dataset. [1.pg 136]

Conclusion:

- Every algorithm give different outcome under different data size.
- There isn't any algorithm universally better than other algorithm. Which algorithm to choose is based on size of the dataset and type of data.
- Accuracy is not equated with how long it takes to develop a model.
- The algorithm won't display precision and accuracy if Kappa Statistic is smaller than Mean Absolute Error (MAE).
- In order to build good classifier , ML algorithms must have precision, accuracy, and minimal error.
- According to this research, a distributed processing environment should be taken into consideration for massive data collections.

Keywords

ML algorithms, MAE, kappa Statistic, learning algorithms, Classifiers

References:

[1] *Researchgate.net*. [Online]. Available: https://www.researchgate.net/publication/318338750_Supervised_Machine_Learning_Algorithms_Classification_and_Comparison. [Accessed: 01-Oct-2023].

Carleton UNIVERSITY

Research Logbook #3

Research Topic: **A Survey of Optimization Methods from a Machine Learning Perspective**

Logbook Focus:

Optimization techniques

Research Paper

S. Sun, Z. Cao, H. Zhu and J. Zhao, "A Survey of Optimization Methods From a Machine Learning Perspective," in IEEE Transactions on Cybernetics, vol. 50, no. 8, pp. 3668-3681, Aug. 2020, doi: 10.1109/TCYB.2019.2950779.

Problem statement:

- Does choose of optimization method effect the accuracy of model?
- Is any one optimizer universally batter than others?
- Does optimization method effect the accuracy of classification?

General points:

Null hypothesis:

There is no difference in optimizer function value while using different optimization technique**.**

Alternative hypothesis:

There is significant difference in optimizer function value while using different optimization technique.

Role of optimizer:

With the specified objective function, the selected numerical or analytical optimisation approaches are normally used to solve the optimisation problem.

Optimization problem in different type of learning algorithms:

In supervised learning:

Minimize loos of optimizer function while training to reduce error while testing.

In semi-supervised learning:

Here we have combined data. some data have label while some don't have. Labeled data use for training, unlabel use in testing. To use unlabel data, additional constraint will added with original data

In unsupervised learning

In this type of learning method we have unlabeled data so, we Use k-mean clustering to assume class for unlabeled data.

In reinforcement learning

the fundamental concept of policy optimization in reinforcement learning, emphasizing the goal of maximizing cumulative returns by guiding an agent's actions through a policy function.

Fundamental Optimization Method:

Gradient descent:

One of the earliest and most used optimisation techniques is gradient descent.

The step size in each iteration is determined by the learning rate, which also affects the number of iterations needed to obtain the ideal value.

The gradient descent approach is sometimes known as batch gradient descent because in linear regression all of the training data are used in each iteration step.

L(θ) = 1/2N X N i=1 (y i − fθ(x i))2

stochastic gradient descent (SGD) :

it is solution for large-scale data.

using one sample randomly to update the gradient per iteration, instead of directly calculating the exact value of the gradient.

When the number of samples N is high, SGD updates significantly more quickly than batch gradient descent does for each iteration.

One issue with SGD is that the gradient direction oscillates as a result of extra noise that random selection introduces.

A learning rate that is too low will slow convergence, while one that is too high will prevent convergence and cause the loss function to change as little as possible.

θ ′ = θ + η(y i − fθ(x i))x i .

Nesterov Accelerated Gradient Descent (NAG):

NAG is a step up from conventional momentum-based optimisation techniques.

In NAG, the parameter vector is combined with the momentum term (v_old * mtm) to generate e, and updates are made using the gradient of e.

It facilitates convergence, especially in non-stochastic optimisation situations, from O(1/k) to O(1/k2).

The update formulae for parameters θ are as follows:

θe = θ + v_old * mtm

v = v_old * mtm + η * (-∂L(θe) / ∂θ)

θ' = θ + v

Adaptive learning rate method:

AdaGrad adjusts the learning rate dynamically based on the historical gradient in some previous iterations.

The difference between AdaGrad and gradient descent is that during the parameter update process, the learning rate is no longer fixed, but is computed using all the historical gradients accumulated up to this iteration.

One main benefit of AdaGrad is that it eliminates the need to tune the learning rate manually.

Variance Reduction Methods:

Use d to update parameter at every iteration. So,the updated item d is calculated by replacing the old gradient G in d with the new gradient G (θt−1) in iteration t.

Each update calculate the gradient of one sample, not gradient of all sample.

Alternating Direction Method of Multipliers(ADMM):

It separate larger problem into smaller parts. And after solving those smaller problem it combine solutions.

Frank-Wolfe Method:
It simplifies complex optimization problems by approximating them with linear functions, finding feasible descending directions through linear programming, and then searching for the optimal solution in that direction within the feasible domain.

Keywords

machine learning, optimization method, supervised learning, semi supervised learning, unsupervised learning, learning rate, reinforcement learning (RL).

References

[1] S. Sun, Z. Cao, H. Zhu and J. Zhao, "A Survey of Optimization Methods From a Machine Learning Perspective," in IEEE Transactions on Cybernetics, vol. 50, no. 8, pp. 3668-3681, Aug. 2020, doi: 10.1109/TCYB.2019.2950779.

Carleton UNIVERSITY

Research Logbook #4

Research Topic: **ANATOMY-SPECIFIC CLASSIFICATION OF MEDICAL IMAGES**

Logbook Focus

Data analysis

Research Paper

H. R. Roth et al., "Anatomy-specific classification of medical images using deep convolutional nets," 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), Brooklyn, NY, USA, 2015, pp. 101-104, doi: 10.1109/ISBI.2015.7163826.

Problem Statement:

- Automated classification of human organs is an important part of many computer-based diagnosis system.
- It is challenging to classify the human body due to its spatial complexity and anatomical diversity.
- Since we cannot afford any kind of misclassified data, after these challenges is it possible to classify human bodies, parts accurately using deep learning?
- Aim of the research is classify medical images into 5 different classes(neck, lungs, liver, pelvis, legs)

General Points:

- When it comes to automatically classifying images, deep learning performs remarkably well, with high accuracy and a low error rate.
- Using 4,298 different axial 2D key-images, we train a ConvNet(convolutional networks) to distinguish 5 distinct anatomical classes in order to create a good classifier.
- Research use 5 cascaded layers of convolutional filters with maxpooling, softmax activation function with dropout layer for prevent overfitting.

Data Analysis:

- We need image data of human body parts before build any kind of model.
- 1675 patients participate in the hospital PACS help to produce data. Through a CT scan of those patients, we captured medical images of various bodily parts.
- Research collect and use 2D version of medical images (32 * 32 pixels).
- Some anatomy images are initially manually categorised with their DICOM image tags (name of bodily component) to establish ground confidence, and later after building model, this test set use as a reference to check data is classified properly or not.
- Here we have CT images of 1675 patients. There is a risk that two hypothetical lunges from different people will differ from one another (lunges with cancer looks little bit different than normal healthy person lungs). Therefore, study article introduces data augmentation to

enrich the data. Also, another benefit of data augmentation is our model perform well as model has more data to learn.

- By applying some data augmentation technique, we expand our data collection.
- A thin-plate spline (TPS) is used to model non-rigid deformations. Control points are randomly changed at the 2D slice level and utilised to generate deformations in a 2D grid. Deformation generation uses the radial basis function (r).
- The augmentation technique considerably broadens the diversity and sample space of training data by include variations in translation, rotation, and non-rigid deformations. So, now total size of data is:
  $N_{TOTAL} = N_{ORIGINAL} * N_{TRANSLATION} * N_{ROTATION} * N_{DEFORMATION}$
- Data augmentation also prevent model by over fitting. If we have less data then while training model, model memorize training data, so with new data, it does not perform well but, because of data augmentation, we have large dataset so, we can eliminate that problem.
- We divide data to 80-20 for training and testing.
- Initially, research suggest to take $N_{TRANSLATION} = 2$ and $N_{ROTATION} = 2$, and then adjust other parameter accordingly.
- Data augmentation help to reduce error by 3.7. ACU is also increased after augmentation which implies that the model is doing a great job at classifying and sorting cases. We can say that model is better at producing precise predictions. Moreover, it is not possible to achieve perfect 1 ACU values so, 0.998 ACU value indicate our classifier is very next to perfect classifier.
- If we examine confusion matrix we can see that, in pelvis we had 2 misclassifies (2 pelvis labelled as liver) but after data augmentation, we do not have any mislabel data for pelvis. Likely for liver, lungs, after augmentation wrong classification decrease by 28 ,5 respectively.

Keywords:

Image Classification, Convolutional Networks, Deep Learning, AUC, Confusion Matrix

References

H. R. Roth et al., "Anatomy-specific classification of medical images using deep convolutional nets," 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), Brooklyn, NY, USA, 2015, pp. 101-104, doi: 10.1109/ISBI.2015.7163826.