

Sample Title^{a)}

A. Author,^{1, b)} B. Author,^{1, c)} and C. Author^{2, d)}

¹⁾ *Authors' institution and/or address*

²⁾ *Second institution and/or address*

(Dated: 15 February 2018)

An article usually includes an abstract, a concise summary of the work covered at length in the main body of the article. It is used for secondary publications and for information retrieval purposes. Valid PACS numbers may be entered using the `\pacs{#1}` command.

I. INTRODUCTION

The the last few decades have seen a drastic increase in the amount of data available to astrophysicists. As manual methods of evaluation are often inefficient, expensive and, in many cases, highly impractical, effective data mining techniques have often been detrimental to many of the recent discoveries in this field.

. Give examples

In this report, we consider the classification task of radio galaxies with active nuclei within the Fanaroff-Riley scheme. Under this scheme, galaxies that decrease in luminosity as the distance from the galactic nuclei increases are designated as FR-I, and sources that increase in luminosity towards the outer parts of the nuclei as FR-II.

. why is this problem important?

The morphological differences between the two classes in radio images have allowed experts to classify candidates visually with a reasonable degree of accuracy. However, due to the large pool of potential candidates, this has resulted in the discovery of only (insert number). The citizen science project Radio Galaxy Zoo has allowed for crowd sourcing the classification task, with (what were the results?). While this has increased the number of examples available to researchers, the sample size is still small enough to make it difficult apply the state of the art image classification methods such as deep learning.

. List attempts and results

More recently, Aniyani and Thorat showed moderate success with convolutional neural networks, achieving a classification accuracy of AA and AA for FR-I and FR-II galaxies, respectively.

A core issue with classification tasks in this domain is that it is difficult to evaluate the success of classifiers due to the lack of past studies to compare new results to. Up until recently, there was insufficient data to attempt most machine learning techniques. In essence, this means that over twenty years of machine learning methods remain untested on the task, leaving a potential wealth of information undiscovered. In addition, the success of many

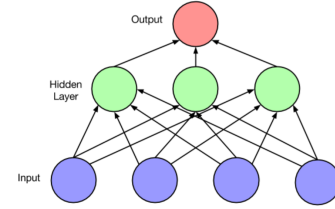


FIG. 1. A Basic Feed forward Neural Network.

machine learning algorithms on image classification tasks is based upon performance on optical images - radio imagery (different in what way).

A. Machine Learning

Stuff

1. Logistic Regression

Other more mathy stuff

2. Neural Networks

. Update Heavily

The ability of the human mind to recognize patterns in complex systems is extremely hard to match algorithmically. Artificial Neural Networks (ANN) are a computational approach of matching the functions of the human brain by mimicking synaptic connections between neurons by series of interconnected nodes organized in layers. ANNs have been successfully applied to a wide range of tasks, including character recognition, face detection, intrusion detection, speech recognition, autonomous driving, stock market prediction, and astronomy.

The most fundamental unit of a Neural Network is a neuron. A basic feed forward neural network consists of sets of neurons, commonly referred to as nodes, arranged into stacked layers⁷. For a layer consisting of n individual nodes that is preceded by a layer of m nodes, the number of connections that exist between the two layers

^{a)}Footnote to title of article.

^{b)}Also at Physics Department, XYZ University.

^{c)}Electronic mail: Second.Author@institution.edu.

^{d)}<http://www.Second.institution.edu/~Charlie.Author>.

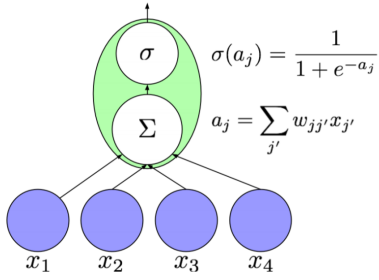


FIG. 2. A node with a sigmoid activation function

is nm . Each link between a node j' and a node j , are assigned a set of trainable values, a weight $w_{jj'}$ and a bias. Associated with each node is an activation function $l_j(\cdot)$. The output value for node j is determined by passing the weighted sum of the inputs values through the activation function, such that:

$$v_j = l_j\left(\sum_{j'} w_{jj'} \cdot v_{j'}\right) \quad (1)$$

This is represented on figure 2, where the sigmoid activation function is denoted by σ .

To train the network, input data that is labeled with a corresponding value or set of values that indicate the desired output is passed through the network. Initially, the network's weights and bias are random, thereby resulting in a random output. In order to train the network, we must find the optimal values for weights and biases that minimize the loss function, or error. This is accomplished through an optimization function. Back propagation is a highly successful gradient based optimization algorithm that operates by calculating the derivative of the loss function with respect to all the weights in the network⁷.

The network learns by iteratively feeding training data through the network and updating the weights according to the loss function. Once the loss between the predicted value is close enough approximation to the correct output, when fed unseen data, the network can accurately approximate the correct output.

B. Histogram of Orientated Gradients (HOG)

HOG description

II. DATA AND SOFTWARE

For the empirical application, we use
(What do we use again?)

All data preparation, configuration, and management, is handled entirely within python 3.5, relying on the packages Numpy for array manipulation, Scipy for for HOG,

and Skimage for it's implementations of legacy machine learning methods, such as Nave Bayes, Random Forrest Trees, and Support Vector Machines. Neural Netowrks and Logistic Regressions were constructed using Keras, a high-level deep learning library for large scale machine learning, with Google's Tensorflow as a backend

III. METHODOLOGY

IV. METHODOLOGY

A core problem for quantitatively classify images is that in almost all cases there is no single feature that can be used to infer the class of the image. Human classification is a complicated and dynamic process build up through millions of years of evolution. First, we are easily able to identify if the object of interest is indeed within our field of vision. By observing an extremely minimal number of examples, we are able to intuitively infer features of importance to the object's class, subsequently allowing us to determine the class of an object by observing the degree to which each relevant feature is present, and taking a mentally weighted sum to determine which class it is most likely to belong to. It is extremely difficult to algorithmically mimic this approach.

Take the FR-I and FR-II classification task. Visually, the radio object's class is often clearly evident. Giving the FR-I galaxies frequently possess a few key features: the brightest points are always at the center of the source, the brightness usually changes more gradually, and brightness is usually inversely proportional to the distance from the sources. FR-II radio galaxies, on the other hand, often comprise of two points that are equidistant from the source, form a strong linear correlation with the source, are often mirrored across the source, and are significantly brighter than the actual source itself. We could, in principle, take a single one of these features and build a classifier. Take the gradient for example. If the change in brightness for an FR-I source is, as a general rule, more gradual than that of an FR-II source, we can use this to determine a source's class. To illustrate: by taking the mean magnitude of the gradient for each pixel in a class for half the data, and then classifying the other half of the data based upon which value the mean gradient magnitude for each image is closest to gives an accuracy of 52.1%. If we are to achieve an accuracy that is remotely within the range of human performance, we need to take into about all relevant features. While the degree to which these features are present in a sample can drastically vary, it is generally possible to classify a sample based upon a dynamic combination features.

If we were to take this approach algorithmically, that is quantitatively determining the degree to which each of the relevant features exists and taking the weighted sum to determine the probability of the sample belong to either class, we would be feature engineering. This is not desirable. While theoretically capable of achiev-

ing high accuracies, many of the induvial features are extremely difficult to quantify, and are often areas of on-going research in their own right. In addition, feature engineering is difficult to implement and is highly inflexible, making it hard to adapt to similar tasks. Due to these reasons, we instead take a machine learning based approach.

Machine Learning techniques have been staggering successful in recent years for images classification. However, many of these algorithms, particularly state of the art methods such as convolutional neural networks, rely on vast amounts of images in order to extract useful features for classification. This makes them unsuited for tasks such as the FR-I/FR-II classification problem, where only 125 and 234 (or something) samples are known for the FR-I and FR-II classes respectively. We employ a number of methods to minimize this issue, primarily, by artificially increasing the number of samples. Using the ImageDataGenerator function in Keras, we applied a combination of random rotations, horizontal and vertical flips, horizontal and vertical translations, and zooms, allowing us to expand the dataset by a factor of one hundred. While genuine samples are preferable, this technique has been proven to be effective in the past (find sources). Another method we employ is to convert the images into a format that allows less computationally exhaustive feature extraction. For this, we use HOG.

A. Legacy Machine Learning Algorithms

To establish a baseline for classification performance, we implement a Logistic Regression, Support Vector Machines (SVM), Nave Bayes, and Random Forrest Trees (RFT).

V. RESULTS

Results - lots of tables and graphs

VI. DISCUSSION

VII. MATH AND EQUATIONS

Inline math may be typeset using the `$` delimiters. Bold math symbols may be achieved using the `\bm` package and the `\bm{#1}` command it supplies. For instance, a bold α can be typeset as `\bm{\alpha}` giving α . Fraktur and Blackboard (or open face or double struck) characters should be typeset using the `\mathfrak{#1}` and `\mathbb{#1}` commands respectively. Both are supplied by the `amssymb` package. For example, `\mathbb{R}` gives \mathbb{R} and `\mathfrak{G}` gives \mathfrak{G} .

In \LaTeX there are many different ways to display equations, and a few preferred ways are noted below. Dis-

played math will center by default. Use the class option `\fleqn` to flush equations left.

Below we have numbered single-line equations, the most common kind:

$$\chi_+(p) \lesssim [2|\mathbf{p}|(|\mathbf{p}| + p_z)]^{-1/2} \left(\frac{|\mathbf{p}| + p_z}{px + ip_y} \right), \quad (2)$$

$$\left\{ 1234567890abc123\alpha\beta\gamma\delta123456\alpha\beta\frac{1\sum_b^a}{A^2} \right\}. \quad (3)$$

Note the open one in Eq. (3).

Not all numbered equations will fit within a narrow column this way. The equation number will move down automatically if it cannot fit on the same line with a one-line equation:

$$\left\{ ab12345678abc123456abcdef\alpha\beta\gamma\delta123456\alpha\beta\frac{1\sum_b^a}{A^2} \right\}. \quad (4)$$

When the `\label{#1}` command is used [cf. input for Eq. (3)], the equation can be referred to in text without knowing the equation number that \TeX will assign to it. Just use `\ref{#1}`, where `#1` is the same name that used in the `\label{#1}` command.

Unnumbered single-line equations can be typeset using the `\[, \]` format:

$$g^+g^+ \rightarrow g^+g^+g^+g^+ \dots, \quad q^+q^+ \rightarrow q^+g^+g^+ \dots$$

A. Multiline equations

Multiline equations are obtained by using the `eqnarray` environment. Use the `\nonumber` command at the end of each line to avoid assigning a number:

$$\begin{aligned} \mathcal{M} = & ig_Z^2 (4E_1 E_2)^{1/2} (l_i^2)^{-1} \delta_{\sigma_1, -\sigma_2} (g_{\sigma_2}^e)^2 \chi_{-\sigma_2}(p_2) \\ & \times [\epsilon_j l_i \epsilon_i]_{\sigma_1} \chi_{\sigma_1}(p_1), \end{aligned} \quad (5)$$

$$\begin{aligned} \sum |M_g^{\text{viol}}|^2 = & g_S^{2n-4} (Q^2) N^{n-2} (N^2 - 1) \\ & \times \left(\sum_{i < j} \right) \sum_{\text{perm}} \frac{1}{S_{12}} \frac{1}{S_{12}} \sum_{\tau} c_{\tau}^f. \end{aligned} \quad (6)$$

Note: Do not use `\label{#1}` on a line of a multiline equation if `\nonumber` is also used on that line. Incorrect cross-referencing will result. Notice the use `\text{#1}` for using a Roman font within a math environment.

To set a multiline equation without *any* equation numbers, use the `\begin{eqnarray*}`, `\end{eqnarray*}` format:

$$\begin{aligned} \sum |M_g^{\text{viol}}|^2 = & g_S^{2n-4} (Q^2) N^{n-2} (N^2 - 1) \\ & \times \left(\sum_{i < j} \right) \left(\sum_{\text{perm}} \frac{1}{S_{12} S_{23} S_{n1}} \right) \frac{1}{S_{12}}. \end{aligned}$$

To obtain numbers not normally produced by the automatic numbering, use the `\tag{#1}` command, where `#1`

is the desired equation number. For example, to get an equation number of (2.6'),

$$g^+g^+ \rightarrow g^+g^+g^+g^+ \dots, \quad q^+q^+ \rightarrow q^+g^+g^+ \dots. \quad (2.6')$$

A few notes on `\tag{#1}`. `\tag{#1}` requires `amsmath`. The `\tag{#1}` must come before the `\label{#1}`, if any. The numbering set with `\tag{#1}` is *transparent* to the automatic numbering in REVTeX; therefore, the number must be known ahead of time, and it must be manually adjusted if other equations are added. `\tag{#1}` works with both single-line and multiline equations. `\tag{#1}` should only be used in exceptional case - do not use it to number all equations in a paper.

Enclosing single-line and multiline equations in `\begin{subequations}` and `\end{subequations}` will produce a set of equations that are “numbered” with let-

ters, as shown in Eqs. (7a) and (7b) below:

$$\left\{ abc123456abcdef\alpha\beta\gamma\delta1234556\alpha\beta\frac{1\sum_b^a}{A^2} \right\}, \quad (7a)$$

$$\mathcal{M} = ig_Z^2 (4E_1 E_2)^{1/2} (l_i^2)^{-1} (g_{\sigma_2}^e)^2 \chi_{-\sigma_2}(p_2) \times [\epsilon_i]_{\sigma_1} \chi_{\sigma_1}(p_1). \quad (7b)$$

Putting a `\label{#1}` command right after the `\begin{subequations}`, allows one to reference all the equations in a subequations environment. For example, the equations in the preceding subequations environment were Eqs. (7).

1. Wide equations

The equation that follows is set in a wide format, i.e., it spans across the full page. The wide format is reserved for long equations that cannot be easily broken into four lines or less:

$$\mathcal{R}^{(d)} = g_{\sigma_2}^e \left(\frac{[\Gamma^Z(3, 21)]_{\sigma_1}}{Q_{12}^2 - M_W^2} + \frac{[\Gamma^Z(13, 2)]_{\sigma_1}}{Q_{13}^2 - M_W^2} \right) + x_W Q_e \left(\frac{[\Gamma^\gamma(3, 21)]_{\sigma_1}}{Q_{12}^2 - M_W^2} + \frac{[\Gamma^\gamma(13, 2)]_{\sigma_1}}{Q_{13}^2 - M_W^2} \right). \quad (8)$$

This is typed to show the output is in wide format. (Since

there is no input line between `\equation` and this paragraph, there is no paragraph indent for this paragraph.)