

# Intuition Learning



Anush Sankaran, Mayank Vatsa and Richa Singh

**Abstract** “By reading only the title and abstract, do you think this research will be accepted in an AI conference?” A common impromptu reply would be “I don’t *know* but I have an *intuition* that this research might get accepted”. Intuition is often employed by humans to solve challenging problems without explicit efforts. Intuition is not trained but is learned from one’s own experience and observation. The aim of this research is to provide *intuition* to an algorithm, apart from what they are trained to *know* in a supervised manner. We present a novel intuition learning framework that learns to perform a task completely from unlabeled data. The proposed framework uses a continuous state reinforcement learning mechanism to learn a feature representation and a data-label mapping function using unlabeled data. The mapping functions and feature representation are succinct and can be used to supplement any supervised or semi-supervised algorithm. The experiments on the CIFAR-10 database show challenging cases where intuition learning improves the performance of a given classifier.

**Keywords** Reinforcement learning · Unsupervised classification · Intuition

## 1 Introduction

*Intuition* refers to knowledge acquired without inference and/or the use of reason [25]. Philosophically, there are several definitions for intuition and the most popularly used one is “Thoughts that are reached with little apparent effort, and typically without conscious awareness” [11] and is considered as the opposite of a rational process.

---

A. Sankaran  
IBM Research, Bangalore, India  
e-mail: [anussank@in.ibm.com](mailto:anussank@in.ibm.com)

M. Vatsa · R. Singh (✉)  
Department of Computer Science and Engineering, IIT Delhi, New Delhi, India  
e-mail: [rsingh@iiitd.ac.in](mailto:rsingh@iiitd.ac.in)

M. Vatsa  
e-mail: [mayank@iiitd.ac.in](mailto:mayank@iiitd.ac.in)

From a machine learning perspective, training a supervised classifier is a rational process where it is trained with labeled data allowing it to learn a decision boundary. Also, traditional unsupervised learning methods do not map the learnt patterns to their corresponding class labels. Semi-supervised approaches bridge this gap by leveraging unlabeled data to better perform supervised learning tasks. However, the final task (say, classification) is performed only by a supervised classifier using labeled data with some additional knowledge from unsupervised learning. The notion of intuition would mean that the system performs tasks using only unlabeled data without any supervised (rational) learning. In other words, intuition is a context-dependent *guesswork* that can be incorrect at certain times. In a typical *learning* pipeline, the concept of intuition can be used for a variety of purposes starting from training data selection up to and including decision-making. Heuristics are the simplest form of intuition that bypass or is used in conjunction with rational decisions to obtain quick approximate results. For example, heuristics can be used in (1) choosing the new data points in an online active learning scenario [6], (2) for feature representation [7], (3) feature selection [10], or (4) choice of classifier and its parameters [4].

Table 1 shows the comparison of existing popular machine learning paradigms. Supervised learning attempts to learn an input–output mapping function on a feature

**Table 1** Comparison of existing popular machine learning paradigms along with the proposed intuition learning paradigm

Paradigm	Input data	Learnt function	Comments
Supervised [3]	<data, label>	data-label mapping	
Unsupervised [3]	<data>	data clusters	
Semi-supervised [5]	<data, label>, unlabeled data	data-label mapping	unlabeled data follow the same distribution
Reinforcement [14]	reward function (or value)	state, action policy	need a teacher to provide reward
Active [24]	<data, label>	data-label mapping, new data selection	need human annotator (Oracle) or expert algorithm to provide labels for new data
Transfer [22]	<sourceData, sourceLabel>, <targetData, targetLabel>	targetData—targetLabel mapping	transfer can be data instances, classification parameters, or features
Imitation [18]	sourceData, sourceData-sourceLabel mapping	targetData—targetLabel mapping	need a teacher to provide reward
Self taught [23]	<data, label>, unlabeled data	data-label mapping	unlabeled data need not follow the same distribution and label as data
Deep learning [2]	<data, label>, unlabeled data	data-label mapping	complex architecture to learn robust data representations
Intuition	data, unlabeled data, reward function (or value)	data-label mapping	unlabeled data need not follow the same distribution, need a reward function

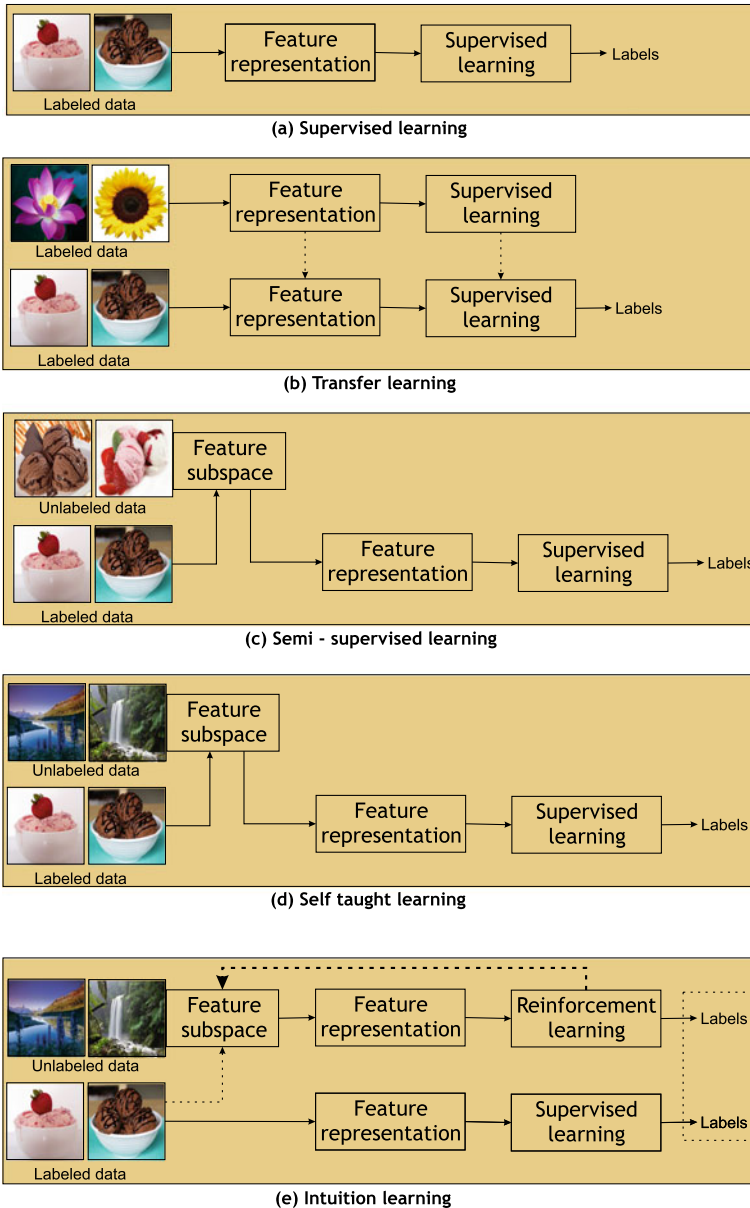
space using a set of labeled training data. Transfer learning aims to improve the target learning function using the knowledge in source (related) domain and source learning tasks [22]. Many types of knowledge transfer such as classification parameters [17], feature representations [9], and training instances [12] have been tested to improve the performance of supervised learning tasks. Semi-supervised learning utilizes additional knowledge from unlabeled data, drawn from the same distribution and having the same task labels as the labeled data. Many of the research works have focused on unsupervised feature learning, i.e., to create a feature subspace using the unlabeled data, to which the labeled data can be projected to obtain a new feature representation [5]. In 2007, Raina et al. [23] proposed a framework termed as “Self-taught learning” to create the generic feature subspace using sparse autoencoders irrespective of the task labels. Self-taught learning dismisses the same class label assumption of semi-supervised learning and forms a generic high-level feature subspace from the unlabeled data, where the labeled data can be projected.

As shown in Fig. 1, we postulate a framework of supplementing *intuition decisions* at the decision level to a supervised or semi-supervised classifier. The decisions drawn by the reinforcement learning block in Fig. 1 are called *intuition* because they are learnt only using the unlabeled data with an indirect reward from a teacher. Existing algorithms, broadly, require training labels for building a classifier or borrows the classifier parameters from an already trained classifier. Direct or indirect training is not always possible as obtaining data labels are very costly. To address this challenge, we propose a novel paradigm for unsupervised task performance mechanism learnt from cumulative experience. Intuition is modeled as a learning framework, which provides the ability to learn a task completely from unlabeled data. By using continuous state reinforcement learning as a classifier, the framework learns to perform the classification task without the need for explicit labeled data. Reinforcement learning helps in adapting a randomly initialized feature space to the specific task at hand, where a parallel supervised classifier is used as a teacher. As the proposed framework is able to learn a mapping function from the input data to the output class labels, without the requirement for explicit training, it functions similar to human intuition and we term this approach as *Intuition Learning*.

## 1.1 Research Contributions

This research proposes a novel intuition learning framework to enable algorithms learn a specific classification or regression task completely from unlabeled data. The major contributions of this research are as follows:

- A continuous state reinforcement learning-based classification framework is proposed to map input data to output class label, without the explicit use of training.
- A residual Q-learning-based function approximation method for learning the feature representation of task-specific data. A novel reward function which does not



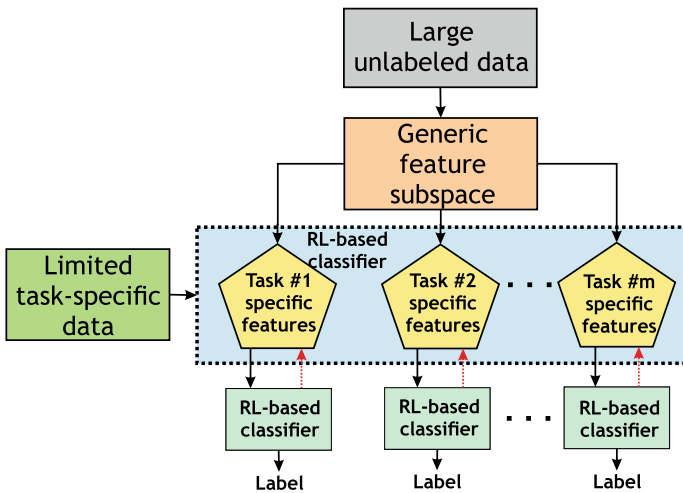
**Fig. 1** Comparing the different learning paradigms such as supervised, semi-supervised, and transfer learning with the proposed intuition learning paradigm. Intuition learning transfer the knowledge to perform classification from unlabeled data using reinforcement learning

- require class labels is designed to provide feedback to the reinforcement-based classification system.
- A context-dependent addition framework is proposed, where the result of the intuition framework can be supplemented based on the confidence of the trained supervised or semi-supervised mapping function.

## 2 An Intuition Learning Algorithm

The basic idea of the proposed intuition learning framework is presented in Fig. 2. Given a large set of unlabelled data, different kinds of feature representations are extracted to describe the data, irrespective of the task in hand. To further leverage the knowledge interpretation from unlabeled data, a continuous state reinforcement learning mechanism is used to perform the given classification task. As reinforcement is a continuous learning process, using a reward-based feedback mechanism, the classification task improves with time. The reinforcement learning, on one hand acts as a classifier, while on the other hand continuously adapts the feature representation with respect to the given task. Thus, given multiple tasks, the proposed intuition learning framework can adapt the generic feature space to be consistent with the corresponding task.

Let  $\{(I_l^{(1)}, y^{(1)}), (I_l^{(2)}, y^{(2)}), \dots, (I_l^{(m)}, y^{(m)})\}$  be the set of  $m$  labeled training data drawn i.i.d. from a distribution  $D$ . The labeled data are represented as  $\{(x_l^{(1)}, y^{(1)}), (x_l^{(2)}, y^{(2)}), \dots, (x_l^{(m)}, y^{(m)})\}$ , where  $x_l^{(i)} \in R^n$  is the feature representation of the



**Fig. 2** A block diagram outlining on how a feature space can be adapted using reinforcement learning algorithm with feedback from a supervised classifier trained on limited task-specific data

data  $I_l^{(i)}$  and  $y^{(i)} \in [1, 2, \dots, C]$  denotes the class label corresponding to  $x_l^{(i)}$ . Let the set of unlabeled data be  $\{I_u^{(1)}, I_u^{(2)}, \dots, I_u^{(p)}\}$ , where the subscript  $u$  represents that they are unlabeled data. Contrary to self-taught learning [23], we do not assume that the labeled and unlabeled data should be drawn from the same distribution  $D$  or have the same class labels, however, they should be derived from the same modality. Given a set of labeled and large unlabeled data, the aim of intuition learning is to learn a hypothesis  $h' : (X \rightarrow R) \in [1, 2, \dots, C]$  that predicts the labels for a given input representation of data drawn. However, the hypothesis  $h'$  is learnt without the direct use of labels  $y^{(i)}$  and is used as a supplement for the hypothesis  $h$  learnt using  $(x_l^{(1)}, y^{(1)})$  in a supervised (or semi-supervised) manner.

## 2.1 Adapting Feature Representation

From a large set of unlabeled data, many different kind of feature representations are extracted. Each representation may correspond to a different property of the data that we try to capture. For image data, the features could be color, texture, and shape while for text data, the features could be n-grams, bag-of-words, and word embeddings. The features can also be a set of different color features or set of hierarchical n-grams. If the large set of unlabeled data is seen as the world (or the universal set), the features are the different observations made by the algorithm from the world. Similar to human intuition, the set of feature representations extracted are task-independent, and later depending on the learning task a subset of these features could be dominantly used. This task-independent feature space is similar to the human intuition learnt by observing the environment.

Figure 3 provides a detailed description of the proposed intuition learning framework. From the set of unlabeled data  $I_u$ , we extract  $r$  different kinds of feature representations,  $\{X_{u_1}, X_{u_2}, \dots, X_{u_r}\}$ , where  $X_{u_i} = \{x_{u_i}^{(1)}, x_{u_i}^{(2)}, \dots, x_{u_i}^{(p)}\}$ , where  $x_{u_i}^{(j)} \in R^{n_i}$ . For every feature representation  $q \in [1, 2, \dots, r]$ , we cluster the representation  $[x_{u_q}^{(1)}, x_{u_q}^{(2)}, \dots, x_{u_q}^{(p)}]$  into  $C$  clusters<sup>1</sup> using k-means clustering. The centroid of each cluster for the  $i$ th feature representation is given as  $[z_{u(i)}^1, z_{u(i)}^2, \dots, z_{u(i)}^C]$ . This feature collection of  $[z_{u(q)}^1, z_{u(q)}^2, \dots, z_{u(q)}^C]$ , for  $q = [1, 2, \dots, r]$  is called as *Intuition-based Feature Subspace (IFS)*, as it clusters the entire set of unlabeled data into groups, based on every observation (feature).

---

<sup>1</sup>The best adaption results are obtained when we fix  $C$  to be the number of classes we have in the learning task.

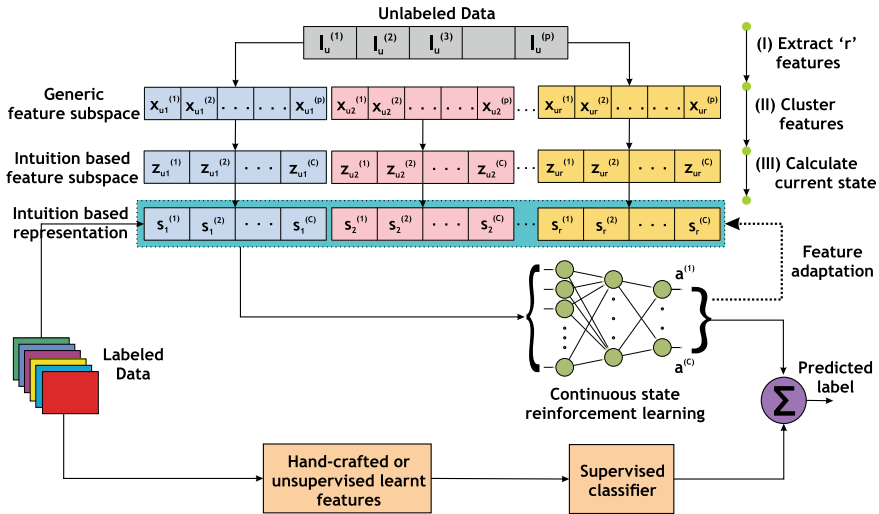


Fig. 3 Overall scheme of the proposed intuition learning algorithm that aids a supervised classifier

## 2.2 Classification Using Reinforcement Learning

For a given set of  $m$  labeled training data,  $\{I_l^{(1)}, I_l^{(2)}, \dots, I_l^{(m)}\}$ , the set of  $r$  features (as used for the unlabeled data) are extracted as  $[x_{l_q}^{(1)}, x_{l_q}^{(2)}, \dots, x_{l_q}^{(m)}]$ , where  $q = [1, 2, \dots, r]$ . The extracted features are then projected onto the Intuition-based Feature Subspace (IFS) by calculating the distance of features from the corresponding cluster centroids shown as,

$$s_q^{(i)} = \|x_{l_q}^{(i)} - z_{u_q}^{(j)}\|_2 \quad (1)$$

for  $j = [1, 2, \dots, C]$ ,  $q = [1, 2, \dots, r]$ , and  $i = [1, 2, \dots, m]$ . The representation of the data  $i$  is given by concatenating the distances corresponding to all the features,

$$s^{(i)} = [s_1^{(i)}, s_2^{(i)}, \dots, s_r^{(i)}] \quad (2)$$

The obtained representation is succinct with a fixed length dimension of  $rC \times 1$ , where  $r$  is the number of different feature types extracted and  $C$  is the number of clusters. In essence, every value represents the distance from a cluster centroid. Also, in a typical semi-supervised (or self-taught) learning scheme, the mapping between intuition based representation and the output class labels,  $\{(s^{(1)}, y^{(1)}), (s^{(2)}, y^{(2)}), \dots, (s^{(m)}, y^{(m)})\}$  is learnt in a supervised manner. However, in the proposed intuition learning, we attempt to learn the data-label mapping without using the class labels, using reinforcement learning. The aim of reinforcement learning is to learn an action policy  $\pi : s \rightarrow a$ , where  $s \in S$  is the current state of the system and  $a$  is the action

performed in that state. As the setup involves a continuous state environment, the optimal action policy is learnt using a model free, off-policy Temporal Difference (TD) algorithm called *Q-learning*, where  $Q(s, a)$ -value denotes the effectiveness of a state-action pair. The *TD(0)* Q-learning algorithm is given by,

$$Q(s_t, a) = Q(s_t, a) + \alpha \left[ r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a) \right] \quad (3)$$

where  $r_t \in R^n$  is the immediate reward obtained for performing action  $a$  in state  $s_t$ ,  $\gamma \in [0, 1]$  is the factor with which the future rewards are discounted and  $\alpha \in [0, 1]$  is the learning rate. In our problem, reinforcement learning is formulated as a classification problem, where *IFS* is the current state  $s$  and action  $a$  is the output label to be predicted, the policy  $\pi$  learns the data-label relation for the given data. Due to the large, probabilistic, and continuous definition of the space  $s$ , the Q-values are approximated using a universal function approximation, i.e., a neural network [26].

$$Q(s, a) = \psi(s, a, \theta) = \sum_i \phi_i(s, a) \cdot \theta_i = \phi^T(s, a) \cdot \theta \quad (4)$$

where  $\phi$  is the approximation function. Using residual Q-learning algorithm [21], the free parameters  $\theta$  are updated as follows:

$$\theta_{t+1} = \theta_t + \alpha \cdot \psi \cdot \Delta \psi \quad (5)$$

$$\begin{aligned} \theta_{t+1} = \theta_t + \alpha \left[ r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a) \right] \\ \times \left[ \beta \gamma \frac{\partial}{\partial \theta} \max_{a'} Q(s_{t+1}, a') - \frac{\partial}{\partial \theta} Q(s_t, a) \right] \end{aligned} \quad (6)$$

where  $\beta$  is a weighting factor called the Bellman residual. Baird [1] guaranteed the convergence of the above approximate Q-learning function, the details of which are skipped for the sake of brevity.  $\epsilon$ - exploration strategy is adopted, where, in every state a random action is preferred with a probability of  $\epsilon$ . As observed in [16], “*the crucial factor for a successful approximate algorithm is the choice of the parametric approximation architecture and the choice of the projection (parameter adjustment) method(s)*”. The choice of reward function employed is highly important and directly implies the effectiveness of adaption, which is explained in the next section.

### 2.3 Design of Reward Function

The Intuition-based Feature Subspace (IFS) is defined by the cluster centroid points obtained using unlabeled data for every feature  $q$  as  $[z_{u_q}^{(1)}, z_{u_q}^{(2)}, \dots, z_{u_q}^{(C)}]$ , where



$q = [1, 2, \dots, r]$ . This space provides an organized definition of how the entire set of unlabeled data is observed and inferred. From the various features of the labeled training data  $[(x_{l_q}^{(1)}, y^{(1)}), (x_{l_q}^{(2)}, y^{(2)}), \dots, (x_{l_q}^{(m)}, y^{(m)})]$ , where  $q \in [1, 2, \dots, r]$ , the centroid points for every feature and every class are calculated as,  $[z_{l_q}^{(1)}, z_{l_q}^{(2)}, \dots, z_{l_q}^{(C)}]$ , where  $q = [1, 2, \dots, r]$ . This space, called the *Labeled data Feature Subspace (LFS)*, formed by these centroid points provide us the inference of the particular learning task to be performed. It is to be noted that:

- Apart from unlabeled data, every labeled training data (and even testing data) gets incrementally added to the *IFS*, as the observed data affects the overall understanding of features.
- The aim of incremental learning is to shape the *IFS* as close as possible to *LF* while learning the feature-label mapping using reinforcement learning.

The incremental update of the *IFS* happens for the  $i$ th training example belonging to  $j$ th class, as shown in the following equation:

$$z_{u_q}^{(j)} = z_{u_q}^{(j)} + \left( \frac{x_{l_q}^{(i)} - z_{u_q}^{(j)}}{n_q^j} \right) \quad (7)$$

for  $q = [1, 2, \dots, r]$ , where  $n_q^j$  is the number of data points in the  $j$ th cluster for  $q$ th feature. Further, to make effective learning from this incremental update, the reward function is defined as a function of the distance between the current *IFS* and *LFS*, as follows:

$$r_t = \left( \|z_{u_q,t}^{(j)} - z_{l_q}^{(j)}\|_2 \right)^{-1} \quad (8)$$

for  $q = [1, 2, \dots, r]$ ,  $j = [1, 2, \dots, C]$  at a given time  $t$ .

## 2.4 Context-Dependent Addition Mechanism

Intuition learning framework acts as a supplement to (and not complementing) supervised learning. The need for intuition arises only when the confidence of supervised learner falls below a particular threshold. Therefore, a context-dependent mechanism is designed to leverage supervised learning using intuition only when required. For given labeled training data  $\{I_l^{(1)}, I_l^{(2)}, \dots, I_l^{(m)}\}$ , some handwritten or unsupervised features are extracted,  $\{(x_l^{(1)}, y^{(1)}), (x_l^{(2)}, y^{(2)}), \dots, (x_l^{(m)}, y^{(m)})\}$  and a supervised model is learnt,  $H_s : (x_l^{(i)} \rightarrow \hat{y}_s)$ . Based on the supervised learning algorithm, the classification confidence is computed for the  $i$ th data point and is given as  $conf_s^{(i)} = [cs_1^{(i)}, cs_2^{(i)}, \dots, cs_C^{(i)}]$ . The mechanism to calculate the classification confidence depends on the supervised learning model used. Similarly, the intuition learning can be represented as  $H_{int} : (s^{(i)} \rightarrow \hat{y}_{int})$  and the classification confidence

is the output of the last layer of the value function approximation neural architecture, given as  $conf_{int}^{(i)} = [cint_1^{(i)}, cint_2^{(i)}, \dots, cint_C^{(i)}]$ . A label switching mechanism is performed to give the final predicted label,  $\hat{y}$ , as follows:

$$\hat{y} = \begin{cases} \hat{y}_s, & \Delta > th \\ \hat{y}_{new}, & \text{otherwise} \end{cases} \quad (9)$$

where  $th$  is the threshold for using intuition and the condition for context  $\Delta$  is calculated as follows:

$$\Delta = \max_j (cs_j^{(i)}) - \max_{l \neq j} (cs_l^{(i)}) \quad (10)$$

In such cases where intuition is used to boost the confidence of supervised classifier the new label is computed as follows:

$$cnew_k^{(i)} = \lambda cs_k^{(i)} + (1 - \lambda) cint_k^{(i)} \quad (11)$$

$$\hat{y}_{new} = \arg \max_j (cnew_j^{(i)}) \quad (12)$$

where  $\lambda$  is the trade-off parameter between intuition and supervised learning. Thus, in simple words, we add the feeling of intuition to an algorithm. The entire approach is summarized as an algorithm in Algorithm 1.

---

### Algorithm 1 Intuition Learning Algorithm

---

1: **Input:** Labeled Data:  $\{(I_l^{(1)}, y^{(1)}), (I_l^{(2)}, y^{(2)}), \dots, (I_l^{(m)}, y^{(m)})\}$ , Unlabeled Data:  $\{I_u^{(1)}, I_u^{(2)}, \dots, I_u^{(p)}\}$ ,  $maxNumberOfEpochs$

2: **repeat**

3:   **for**  $i = 1$  to  $m$  **do** ▷ Extract  $r$  different types of features

4:      $\{[x_{u1}^{(1)}, x_{u1}^{(2)}, \dots, x_{u1}^{(p)}], [x_{u2}^{(1)}, x_{u2}^{(2)}, \dots, x_{u2}^{(p)}], \dots, [x_{ur}^{(1)}, x_{ur}^{(2)}, \dots, x_{ur}^{(p)}]\}$  ←  $\{I_u^{(1)}, I_u^{(2)}, \dots, I_u^{(p)}\}$  ▷ Cluster data into  $C$  groups based on each feature

5:      $[z_{uq}^1, z_{uq}^2, \dots, z_{uq}^C], \forall q = [1, 2, \dots, r]$  ▷ Compute the current state

6:      $s_q^{(i)} = \|x_{I_q}^{(i)} - z_{I_q}^j\|_2, \forall j = [1, 2, \dots, C], \forall q = [1, 2, \dots, r], \forall i = [1, 2, \dots, m]$  ▷ Approximate the current state Q-value

7:      $Q(s, a) = \psi(s, a, \theta) = \sum_i \phi_i(s, a) \cdot \theta_i$  ▷ Update  $\theta$

8:      $\theta_{t+1} = \theta_t + \alpha \cdot \psi \cdot \Delta \psi$  ▷ Compute reward value

9:      $r_t = \left( \|z_{uq,t}^j - z_{I_q}^j\|_2 \right)^{-1} \forall q = [1, 2, \dots, r], \forall j = [1, 2, \dots, C]$

10:   **end for**

11: **until**  $maxNumberOfEpochs$  or  $\Delta \psi < \text{thresh}$  ▷ Test phase

12: For  $I_l^{(i)}$ , calculate  $\beta, \hat{y}_s, \hat{y}_{new}$

13: **if**  $\Delta > \text{thresh}$  **then**

14:    $\hat{y} \leftarrow \hat{y}_s$

15: **else**

16:    $\hat{y} \leftarrow \hat{y}_{new}$ , as shown in Eqs. 11, 12.

17: **end if**

---

### 3 Experimental Analysis

#### 3.1 Dataset

The proposed intuition learning algorithm is applied for 10-class classification problem using the CIFAR-10 database [15]. The database contains 60,000 color images labeled, each of size  $32 \times 32$  pertaining to 10 classes (i.e., 6,000 images per class). There are 50,000 training images and 10,000 test images. The data set contains small size images, leading to limited and noisy information content and it provides the most relevant case study to demonstrate the effectiveness of the proposed paradigm. The STL-10 database [7] is used as the unlabeled image data set having one million colored images of size  $96 \times 96$ . As shown in Table 2 six different feature representations are extracted from the images. These features comprehensively comprise the various types of features that could be extracted from image data. For all the experiments, five times random cross-validation is performed and the best model accuracy is reported for all the experiments. Sample images from CIFAR-10 and STL-10 datasets are shown in Fig. 4.

#### 3.2 Interpreting Intuition-Based Feature Subspace

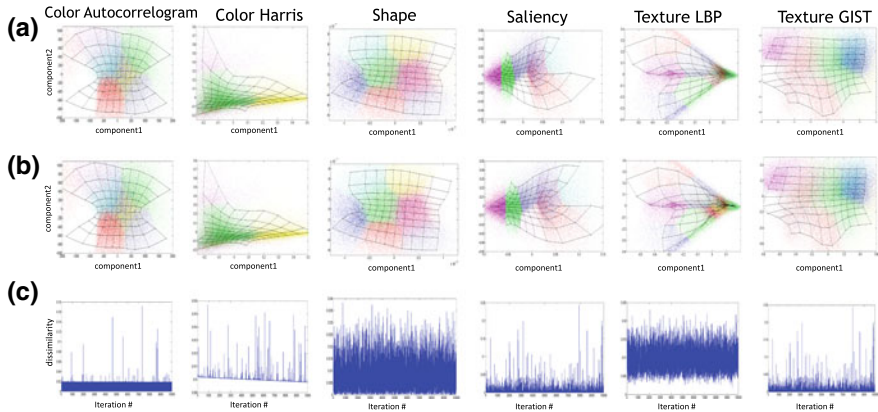
The primary aim of the approach is to construct the feature subspace completely from unlabeled data and to adapt it to a specific learning task. Figure 5 shows the clusters of entire unlabeled data corresponding to every feature extracted. The concatenation of the feature spaces put together in Fig. 5a represents the *IFS*. Figure 5b shows the adapted task-specific feature subspace after performing 300 epochs of learning with the given labeled data. Figure 5c shows the amount of update in the cluster after adding an image, by calculating the dissimilarity between the cluster centroid, before and after the addition of the image. Cluster dissimilarity is calculated for the  $r - th$  feature representation as follows:

**Table 2** Details of different features extracted from the image data

Type	Feature	Dimension	Parameters
Color	Color harris [27]	$10 \times 2$	$\sigma_g = 1.5, \sigma_a = 5$
Color	Color autocorrelograms [13]	$64 \times 1$	Quantization level, $m = 64$
Local texture	Local binary pattern (LBP) [19]	$59 \times 4$	$N = 8, R = 1$
Global texture	GIST [20]	$512 \times 1$	$n_\theta = 8, n_{\text{block}} = 4$
Saliency	Region covariances [8]	$32 \times 32$	$r = 3, \sigma = 1.2, m = 1/10$
Shape	Multilayer autoencoder [28]	$10 \times 1$	size = [10, 110, 110, 340, 340, 1024, 1024]



**Fig. 4** Sample set labeled images from CIFAR-10 database and unlabeled images from STL-10 database



**Fig. 5** Image showing the data clusters for each of the extracted feature and grid depicts the cluster density at local regions. **a** shows the *IFS* of all the unlabeled data, **b** shows the adapted task-specific feature space after 300 epochs of learning, and **c** shows the amount of change happening in the cluster after the addition of an image. Best viewed in color

$$C_{dis} = \sum_{j=1}^C \frac{1}{D_j} \cdot \|z_{u_r}^{(j)}|_{(t+1)} - z_{u_r}^{(j)}|_{(t)}\|_2 \quad (13)$$

where  $D_j$  is the density of the  $j$ th cluster. It can be visually observed from the plot that, shape, gist, and LBP feature spaces are updated (learns) after the addition of each image, indicating that these features contribute more towards the classification task. However, both color Harris and autocorrelogram features are not much updated by the training data.

### 3.3 Performance Analysis

It is to be noted that intuition learning framework is used to supplement any supervised or semi-supervised learning mechanism. In this research, we show the results in the following scenario:

1. Using two supervised learning algorithms (backpropagation neural network and multi-class SVM) with Uniform Circular Local Binary Pattern (UCLBP) [19] as features. Labeled data,  $[(x_{l_q}^{(1)}, y^{(1)}), (x_{l_q}^{(2)}, y^{(2)}), \dots, (x_{l_q}^{(m)}, y^{(m)})]$ , from CIFAR-10 is used to train the supervised algorithms.
2. Using a semi-supervised learning algorithm, with neural network as classifier and UCLBP features trained on CIFAR-10 dataset. The semi-supervised algorithm used for comparison is one approach for self-taught learning [23], with unlabeled data from STL-10 dataset,  $\{(s^{(1)}, y^{(1)}), (s^{(2)}, y^{(2)}), \dots, (s^{(m)}, y^{(m)})\}$ .
3. Using a intuition learning framework only, having the intuition-based task-specific feature representation combined with a continuous state reinforcement learning (Q-learning) in Eq. 4 for classification.
4. Using a supervised intuition framework, where the output of the supervised learning algorithm and the intuition learning framework is combined using the context-dependent addition mechanism in Eq. 12.
5. Using a semi-supervised intuition framework, where the output of the semi-supervised learning algorithm and the intuition learning framework is combined using the context-dependent addition mechanism.

The optimized values of various parameters used in our framework are as follows:  $\alpha = 0.99$ ,  $\gamma = 0.95$ ,  $\beta = 0.2$ ,  $th = 0.9$ ,  $\lambda = 0.5$ , and  $\epsilon = 0.05$ . Preprocessing of features is done using z-score normalization. All the experiments are performed on a Intel Xeon E5 – 2640 0, 2.50GHz, 64GB RAM server.

As already discussed, intuition has a better significance in challenging problems with limited training data. Tables 3, 4, and 5 show the performance of the proposed intuition learning in comparison with other learning methods, by varying the train-

**Table 3** The performance accuracy (%) of supervised intuition learning is compared with supervised (neural network) and semi-supervised (self-taught) learning methods. The significance of intuition is studied by varying the amount of available training data. 5 times random cross-validation is performed and the best model’s performance is reported

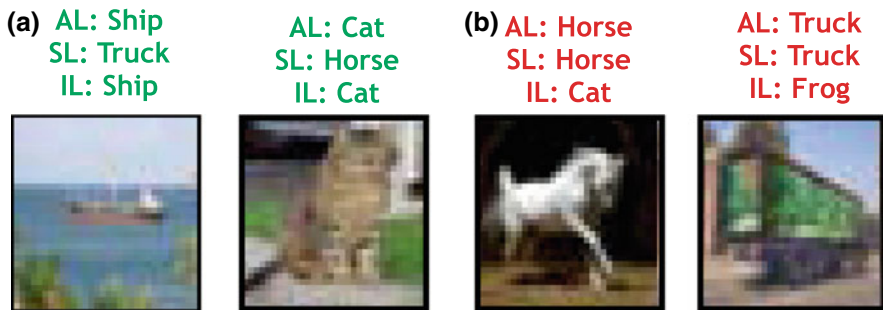
Training size	Supervised	Semi-supervised	Supervised intuition
10000	<b>38.90</b>	29.64	36.19
5000	<b>37.13</b>	24.34	34.78
3000	14.12	17.07	<b>25.65</b>
1000	10.34	16.54	<b>19.61</b>

**Table 4** The influence of supplementing intuition to supervised and semi-supervised algorithm is shown by improvement in the performance accuracy (%)

Training size	Intuition	Intuition supervised	Intuition semi-supervised
10000	12.11	36.19	29.21
5000	10.00	34.78	23.85
3000	10.00	<b>25.65</b>	<b>22.49</b>
1000	08.99	<b>19.61</b>	<b>20.53</b>

**Table 5** The performance accuracy (%) of supervised and supervised intuition framework using SVM classifier is studied

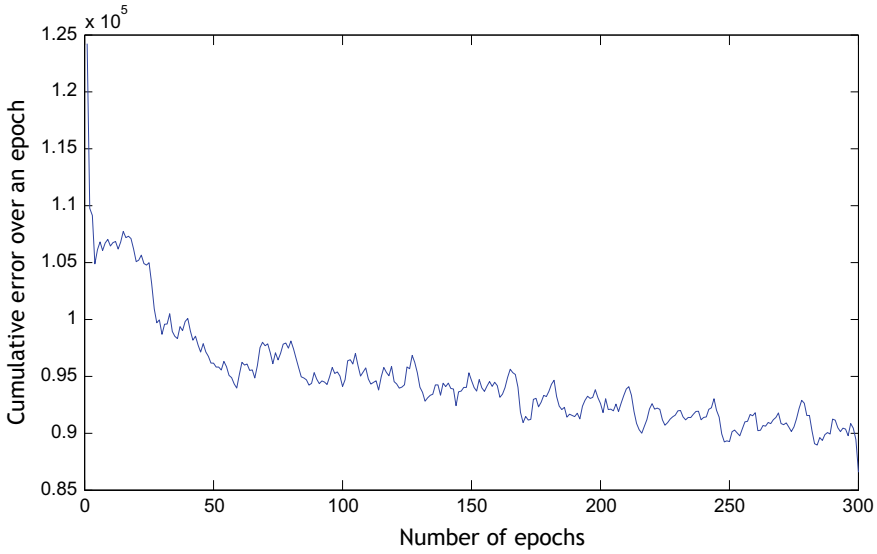
Training size	Supervised	Intuition supervised
10000	<b>44.57</b>	41.83
5000	<b>43.21</b>	40.77
3000	13.56	<b>17.48</b>
1000	06.19	<b>09.78</b>



**Fig. 6** Examples of **a** success and **b** failure cases of the proposed intuition learning. AL = actual ground truth label, SL = label predicted by the supervised neural network learner, and IL = label predicted when intuition is combined with supervised neural network learner

ing size as parameter.<sup>2</sup> It can be observed that with enough training data, supervised algorithms (both neural network and SVM) yield the best classification performance. However, with decrease in the size of training data, the performance of all the three algorithms, supervised, semi-supervised, and intuition learning reduces. The results show that in such a scenario, incorporating intuition with supervised or semi-supervised algorithm yields improved results. This supports our hypothesis that adding intuition would improve the performance from under challenging circumstances such as limited training data. Similarly from a human's perspective, under

<sup>2</sup>For a given training size, the same subset of images is used across all the classifiers to avoid any training bias.



**Fig. 7** A plot between the cumulative errors across each epoch empirically showing the learning effectiveness of the residual Q-learning performed in Eq.6

the presence of all data and information, one may take correct decisions. However, when the background training data information is limited, intuition learning helps. Further, some key analysis are summarized below:

1. To study the effectiveness of residual learning in Eq.6, training error over successive epochs is plotted, as shown in Fig.7, for a training size of 10000. It can be observed that the training error gradually decreases and remains constant after 300 epochs, indicating that maximum training capacity has been achieved, with minimum training error.
2. The computation time required for intuition learning depends on the complexity of  $r$  features that are extracted. However, for one sample, under the assumption that the feature extraction happens off-line, the overall intuition decision and feature space can be generated in 0.082s while the supervised decision can be taken in  $\sim 4$ s on an average. This shows that intuition is much faster requiring little effort than supervised decision-making.
3. In Fig. 6, some success and failure example cases are shown where (a) intuition helps in correctly classifying a data but supervised learning fails and (b) data was incorrectly classified because of intuition. As previously discussed, intuition can go wrong sometimes. Upon analyzing the first horse example in failure case (Fig. 6b), it is observed that horses are clustered more towards brown color in the autocorrelogram color feature space. However, as the horse shown in the images is white in color, it gets clustered along with cat and misclassified by intuition learning.

## 4 Conclusion

Inspired from human capabilities of instinct reasoning, this research presents a intuition learning framework that supplements a classifier for improved performance, especially with limited training data. Intuition is modeled as a continuous state reinforcement learning, that adapts to a particular task using large amount of unlabeled data and limited task-specific data. The performance of intuition is shown in a 10 class image classification problem, in comparison with supervised, semi-supervised, and reinforcement learning. The results indicate that the application of intuition improves the performance of the classifier with limited training data.

## References

1. Baird L (1995) Residual algorithms: reinforcement learning with function approximation. In: ICML, pp 30–37
2. Bengio Y (2009) Learning deep architectures for AI. *Found Trends Mach Learn* 2(1):1–127
3. Bishop CM, Nasrabadi NM (2006) *PRML*, vol 1
4. Cavalin PR, Sabourin R, Suen CY (2012) LoGID: an adaptive framework combining local and global incremental learning for dynamic selection of ensembles of HMMs. *PR* 45(9):3544–3556
5. Chapelle O, Schölkopf B, Zien A et al (2006) *Semi-supervised learning*, vol 2. MIT Press, Cambridge
6. Chu W, Zinkevich M, Li L, Thomas A, Tseng B (2011) Unbiased online active learning in data streams. In: *ACM SIGKDD*, pp 195–203
7. Coates A, Ng AY, Lee H (2011) An analysis of single-layer networks in unsupervised feature learning. In: *ICAI*, pp 215–223
8. Erdem E, Erdem A (2013) Visual saliency estimation by nonlinearly integrating features using region covariances. *J Vis* 13(4)
9. Evgeniou A, Pontil M (2007) Multi-task feature learning. In: *NIPS*, vol 19, p 41
10. Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. *JMLR* 3:1157–1182
11. Hogarth RM (2001) *Educating intuition*. University of Chicago Press, Chicago
12. Huang J, Gretton A, Borgwardt KM, Schölkopf B, Smola AJ (2006) Correcting sample selection bias by unlabeled data. In: *NIPS*, pp 601–608
13. Huang J, Kumar SR, Mitra M, Zhu WJ, Zabih R (1997) Image indexing using color correlograms. In: *CVPR*, pp 762–768
14. Kaelbling LP, Littman ML, Moore AW (1996) Reinforcement learning: a survey. [arXiv:cs/9605103](https://arxiv.org/abs/cs/9605103)
15. Krizhevsky A, Hinton G (2009) Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto
16. Lagoudakis MG, Parr R (2003) Least-squares policy iteration. *JMLR* 4:1107–1149
17. Lawrence ND, Platt JC (2004) Learning to learn with the informative vector machine. In: *ICML*, pp 65–72
18. Natarajan S, Joshi S, Tadepalli P, Kersting K, Shavlik J (2011) Imitation learning in relational domains: a functional-gradient boosting approach. In: *IJCAI*, pp 1414–1420
19. Ojala T, Pietikainen M, Maenpää T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE TPAMI* 24(7):971–987
20. Oliva A, Torralba A (2001) Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV* 42(3):145–175



21. Paletta L, Pinz A (2000) Active object recognition by view integration and reinforcement learning. *Robot Auton Syst* 31(1):71–86
22. Pan SJ, Yang Q (2010) A survey on transfer learning. *IEEE TKDE* 22(10):1345–1359
23. Raina R, Battle A, Lee H, Packer B, Ng AY (2007) Self-taught learning: transfer learning from unlabeled data. In: *ICML*, pp 759–766
24. Settles B (2010) Active learning literature survey. Technical report, Computer Sciences Technical Report 1648, University of Wisconsin, Madison
25. Simpson JA, Weiner ES et al (1989) *The Oxford english dictionary*, vol 2. Clarendon Press, Oxford
26. Sutton RS, Barto AG (1998) *Reinforcement learning: an introduction*. Cambridge University Press, Cambridge
27. Van De Weijer J, Gevers T, Smeulders AW (2006) Robust photometric invariant features from the color tensor. *IEEE TIP* 15(1):118–127
28. Vincent P, Larochelle H, Bengio Y, Manzagol PA (2008) Extracting and composing robust features with denoising autoencoders. In: *ICML*, pp 1096–1103