Big Idea #3: Learning

Key Insights	Explanation
Machine learning allows a computer to acquire behaviors without people explicitly programming those behaviors.	Definition of "machine learning"
Learning new behaviors results from changes the learning algorithm makes to the internal representations of a reasoning model, such as a decision tree or a neural network.	How machine learning algorithms work
Large amounts of training data are required to narrow down the learning algorithm's choices when the reasoning model is capable of a great variety of behaviors.	The role of training data
The reasoning model constructed by the machine learning algorithm can be applied to new data to solve problems or make decisions.	Learning phase vs. application phase

Big Idea #3: Learning	*Computers can learn from data.*	LO = Learning Objective: What students should be able to do.	EU = Enduring Understanding: What students should know.	Unpacked descriptions are included when necessary to illustrate the LO or EU
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Nature of Learning *(Humans vs. machines)* **3-A-i**	**LO:** Describe and provide examples of how people learn and how computers learn. **EU:** Computers learn differently than people. **Unpacked:** People learn by observation, by being told, by asking questions, by experimentation, by practice, and by making connections to past experience. Computers learn by finding patterns in data, or by trial and error. **Activities:** Describe a time when you learned something by being told, by watching another person, or by asking questions. A demo such as Teachable Machine can be used to illustrate a computer learning something from positive and negative examples.	**LO:** Differentiate between how people learn and how computers learn. **EU:** Both people and computers can learn by finding patterns in data, or by trial and error. But people are flexible learners who can adapt to unfamiliar situations and learn in other ways, such as by observing others, by asking questions, or by making connections to prior learning. **Unpacked:** People are natural learners, while computers have to be programmed to learn. Presently there are two ways that computers can be programmed to learn: they can learn by finding patterns in human-supplied examples, or they can learn by trial and error.	**LO:** Contrast the unique characteristics of human learning with the ways machine learning systems operate. **EU:** People learn by observation, by being told, by asking questions, by experimentation, by practice, and by making connections to past experience. Computers learn by applying specialized algorithms to large amounts of training data, or by thousands or even millions of trial and error experiences, to solve narrowly defined problems. **Unpacked:** People are flexible learners who employ multiple strategies. Computers use specialized algorithms that require large amounts of data or many trials, and only solve narrowly defined problems. While humans can construct reasoners by explicitly programming them, for complex problems it is often more convenient to let the machine learning algorithm do the work.	**LO:** Define supervised, unsupervised, and reinforcement learning algorithms, and give examples of human learning that are similar to each algorithm. **EU:** Both supervised and unsupervised learning algorithms find patterns in data. Supervised learning uses features to predict the class label supplied by a teacher; unsupervised learning groups similar instances together, creating its own classes. Reinforcement learning uses trial and error to find a policy for choosing actions that maximizes the reinforcement signal. **Unpacked:** Supervised learning is like being corrected by a coach. Unsupervised learning is like noticing that your store has three kinds of customers based on their distinctive purchasing patterns. Reinforcement learning is like trying different moves in a video game and seeing which yields the most points (greatest reward).
Nature of Learning (Finding patterns in data) **3-A-ii**	**LO:** Identify patterns in labeled data and determine the features that predict labels. **EU:** Classes can be defined in terms of feature values. The relevant features can be inferred by examining labeled examples. **Unpacked:** To give students a feel for the problem of learning to classify we must ask them to learn a class that's not intuitively obvious, e.g., learn "poisonous fish" by examining cartoon fish images labeled "poisonous" or "not poisonous". They can then be asked to describe which features indicate a fish is poisonous, e.g., red fish with square heads. Using images as input simplifies the task because the features are intuitive, even though the classification rule should not be.	**LO:** Model how supervised learning identifies patterns in labeled data. **EU:** When learning to classify labeled data, the patterns (or rules) that are discovered can be expressed as weights in a neural network or nodes in a decision tree. **Unpacked:** This extends the K-2 version by having students draw a decision tree instead of merely verbalizing their proposed rule. In addition, the task can be made richer in 3-5 by increasing the number of classes or by making the class definitions more complex. For example, a fish could be poisonous if it is either red with a square head or blue with a round head or purple with pointy spines and any shape head. Each node of the decision tree can test one feature value, e.g., color, so complex features require deeper trees.	**LO:** Model how unsupervised learning finds patterns in unlabeled data. **EU:** Unsupervised learning is useful when we don't know in advance what classes exist. It discovers patterns (or classes) in data by grouping nearby points into clusters. Once a set of clusters has been found, new points can be classified based on distance from the cluster boundaries. **Unpacked:** This can be done graphically using points in the plane and visually constructing cluster boundaries by outlining (e.g., drawing an ellipse around) each cluster.	**LO:** Model how machine learning constructs a reasoner for classifcation or prediction by adjusting the reasoner's parameters (its internal representations). **EU:** Supervised learning adjusts the parameters of a mathematical model (selected in advance by a human) to generate correct classifications or predictions. This model could be a simple linear equation, a high-degree polynomial, or an even more complex nonlinear equation such as a deep neural network. The internal representations that encode the relationship between inputs and outputs express the "patterns" found in the data. **Unpacked:** In regression, we pick a mathematical model such as a linear equation y=mx+b and then adjust its parameters to fit a set of data points as best we can. The model can then be used to predict a y value for any x value. Linear regression can be done with a ruler by eyeballing the distance between the line and the points. Students can model polynomial or logistic regression by giving them a graphical display with sliders to control the parameter values. They can manually adjust the sliders to reach what they perceive as a best fit to the data. More advanced students can be shown how quality of fit can be measured mathematically using mean squared error. For classification problems the Y value is either 1 for "in class" or 0 for "not in class" and the decision boundary is the line or surface y=0.5.

Big Idea #3: Learning	*Computers can learn from data.*	LO = Learning Objective: What students should be able to do.	EU = Enduring Understanding: What students should know.	Unpacked descriptions are included when necessary to illustrate the LO or EU
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Nature of Learning (Training a model) **3-A-iii**	**LO:** Demonstrate how to train a computer to recognize something. **EU:** Computers can learn from examples. **Unpacked:** With instructor assistance, Teachable Machine could be used to recognize hand gestures or sounds.	**LO:** Train a classification model using machine learning, and then examine the accuracy of the model on new inputs. **EU:** Computers can learn to classify instances or predict values by being shown labeled examples. If the results on new inputs are unsatisfactory, additional training may be required to improve the accuracy. **Activity:** Using Teachable Machine or Machine Learning for Kids, training examples can be supplied by webcam input or collected from an image search on the web, and the model can be trained on a task such as recognizing pictures of cats.	**LO:** Train and evaluate a classification or prediction model using machine learning on a tabular dataset. **EU:** Computers can learn to classify instances or predict values by examining feature values. If the results on new inputs are unsatisfactory, additional training may be required to improve the accuracy. **Unpacked:** Within a tabular dataset, each training example is a row in the table and is described by a set of feature values; the features are the columns of the table. Classification assigns each example to one of a discrete set of classes (e.g., cat or dog); prediction outputs a continuous value, such as predicting a person's height from their age. The learning algorithm is likely to be a decision tree learner rather than a neural network. **Activity:** Sites like MachineLearningForKids and eCraft2Learn include decision tree learning. The learning algorithm figures out which are the relevant features and what values they should have for each class.	**LO:** Use either a supervised or unsupervised learning algorithm to train a model on real world data, then evaluate the results. **EU:** In supervised learning the model is trained on a training set to produce the correct labels for labeled data. We evaluate the results by measuring the percent of items in a test set that are labeled correctly. In unsupervised learning, the model is trained to assign each input to a cluster of similar inputs. The clusters are determined by the learning algorithm since there are no labels attached to the training data. We evaluate the results by examining the clusters to see if they capture useful distinctions in the dataset. **Unpacked:** Both supervised and unsupervised learning algorithms find patterns in data. In supervised learning, the "pattern" is the relationship between feature values and class labels. In unsupervised learning the pattern is the way that data is grouped into clusters. Real world data sets are now widely available on the web. In earlier grade bands students might test their trained models on a few new data points, but in this grade band students are asked to quantitatively measure the performance of a trained model on a nontrivial test set.
Nature of Learning *(Constructing vs. using a reasoner)* **3-A-iv**	**N/A**	**LO:** Demonstrate how training data are labeled when using a machine learning tool. **EU:** In preparation for training a model, training data can be labeled by first defining the classes (the labels) and then adding examples for each class separately. After training, new data can be presented to the model and it will predict the class, but the data are unlabeled so the model receives no feedback about the correctness of its class predictions. **Unpacked:** Teachable Machine provides three classes by default and has a separate "Hold to Record" button for each class, so training examples are implicitly labeled based on which class they are recorded for. After training, the model is classifying webcam input in real time but receives no feedback.	**LO:** Explain the difference between training and using a reasoning model. **EU:** Machine learning algorithms use labeled training data to construct reasoning models that do classification or prediction. During training, the reasoning model runs on the training inputs and the learning algorithm adjusts the model to make its outputs more closely match the labels. Once training is complete, the reasoning model can be applied to new data to solve problems or make decisions. Using a trained reasoning model this way does not change it; only the learning algorithm can change the model.	**LO:** Illustrate what happens during each of the steps required when using machine learning to construct a classifier or predictor. **EU:** The steps are: deciding what problem you want to solve, figuring out where you will obtain the training data, choosing a feature set, figuring out how to label the data, running the learning algorithm, use of a cross-validation set to decide when training should stop, and using a test set to measure performance. **Unpacked:** The cross-validation set is used to avoid overfitting. The test set consists of examples that were not used during training or for cross-validation, so it provides an unbiased prediction of the reasoner's performance on new inputs.

Big Idea #3: Learning	*Computers can learn from data.*	LO = Learning Objective: What students should be able to do.	EU = Enduring Understanding: What students should know.	Unpacked descriptions are included when necessary to illustrate the LO or EU
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Nature of Learning (Adjusting internal representations) **3-A-v**	**N/A**	**LO:** Analyze a game where one constructs a decision tree, describing the organization of the tree and the learning algorithm used to add nodes. **EU:** In a decision tree learning game, the tree's branch nodes are questions and the leaf nodes are classes. The learning algorithm moves through the tree by asking the questions at the branch nodes (testing features of the input) until it arrives at a leaf node. If that leaf node's class is incorrect, the node is replaced by a branch node with a new question, and the leaf node is reattached at that branch. **Activity:** the "guess the animal" game, troubleshooting problems, and the Pasta Land activity are good choices for demonstrating decision tree learning.	**LO:** Compare how a decision tree learning algorithm works vs. how a neural network learning algorithm works. **EU:** In decision tree learning, each step adds a new node, which tests a single feature value. In neural network learning, each step makes a small change to every weight in the network. **Unpacked:** A decision tree's internal representations are the nodes, the feature each node examines, and the value the feature is compared against. A neural network's internal representations are the weights. Decision tree learning algorithms try to find, for each new node they create, the most informative feature to examine. Changing the training data can result in a different choice of feature to examine next. With a neural network, changing the training data will lead to different weight adjustments as the algorithm tries to reduce the error signal.	**LO:** Describe how various types of machine learning algorithms learn by adjusting their internal representations. **EU:** Decision tree learners build decision trees by adding nodes one at a time. Neural net learning algorithms adjust weights. Regression algorithms adjust equation parameters. Reinforcement learners update value predictions or policies.
Nature of Learning (Learning from experience) **3-A-vi**	**N/A**	**LO:** Explain how reinforcement learning allows a computer to learn from experience (i.e., trial and error). **EU:** Computers can learn from experience if there is a "reinforcement" signal indicating whether the computer's actions are leading to good or bad outcomes. **Unpacked:** Computers can learn to play games using a reinforcement signal that indicates whether the computer won or lost the most recent game, or how many points it scored. The computer may have to play hundreds of thousands of games to become an expert player. **Demonstration:** Reinforcement learning can be illustrated using an agent navigating through a grid world with obstacles and hazards; the task is to learn the best path to a goal location. At each grid square, the allowable actions are to move N/S/E/W. Over repeated trials, the agent learns the best move to make in each square.	**LO:** Explain the differences between supervised learning and reinforcement learning. **EU:** Supervised learning tells the agent what output it should produce for each input; reinforcement learning only tells the agent how well it's doing as it chooses actions to take. **Unpacked:** In supervised learning, the teacher indicates the correct output for each training example, so the learning algorithm can see what it's doing wrong. In reinforcement learning, the reinforcement signal indicates how well the model is performing, but does not tell the learning algorithm what actions the model should have chosen to do better. This must be discovered by trial and error, so it may take hundreds of thousands of trials to reach expert level performance. For example, when playing a video game, the reinforcement signal could be the number of points scored. Because the computer learns from its own experience, reinforcement learning can find solutions to problems where there is no teacher who could tell it the best action to take.	**LO:** Select the appropriate type of machine learning algorithm (supervised, unsupervised, or reinforcement learning) to solve a reasoning problem. **EU:** Major types of learning algorithms and the kinds of reasoning problems they are used to solve are: supervised learning, used for classification and prediction; unsupervised learning, used for clustering; and reinforcement learning, used for sequential decision making. **Unpacked:** Both supervised and unsupervised learning algorithms find patterns in data. Supervised algorithms use labeled training data and adjust the reasoning model's parameters to try to produce the correct labels. They are used for classification or prediction problems. Unsupervised learning algorithms. which use unlabeled data, try to group similar data points together. They are used to discover classes in the data. Reinforcement learning algorithms are used for sequential decision problems. They learn policies for choosing actions that maximize the reinforcement the model will receive. Reinforcement learning can be slow because learning must proceed by trial and error; there is no teacher telling the algorithm the best action at each step. But having a computer learn from its own experience has the advantage that it can discover solutions to problems where it's not known in advance which is the best action.

Big Idea #3: Learning	*Computers can learn from data.*	LO = Learning Objective: What students should be able to do.	EU = Enduring Understanding: What students should know.	Unpacked descriptions are included when necessary to illustrate the LO or EU
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Neural Networks *(Structure of a neural network)* **3-B-i**	**N/A**	**LO:** Illustrate how a neural network of 1 to 3 neurons is a function that computes an output. **EU:** A neural network uses one or more neurons working together to form a function. Each neuron takes a set of numbers as input and produces a single number as its output. **Unpacked:** A neural network is a collection of neurons that are connected to each other. Every neuron has a set of input connections, each with an attached weight. Each input connection carries a value. The neuron multiplies each input value by the connection weight to produce a weighted input. The sum of all the weighted inputs is compared to the neuron's threshold value. If the sum is above the threshold value, the neuron outputs a 1; otherwise it outputs a 0. The output value can be used as an input for other neurons. **Activity:** Calculate the output of a single neuron with multiple inputs, or a network of two multi-input "hidden" neurons feeding a single output neuron. Such networks can compute simple functions such as "AND", "OR", or "at least 2 out of 3". For a quick tutorial on neural nets for grades 3-5, see https://docs.google.com/document/d/1bYs0tTiL44sQhsMADgU2bDmVjWaVKVl2pT_SYZTzEwI/edit#heading=h.g640mybwbie6	**LO:** Illustrate the structure of a neural network and describe how its parts form a set of functions that compute an output. **EU:** Neural networks are organized as layers of units (input, hidden, and output layers), with weighted connections between units in successive layers. Each unit computes the sum of its weighted inputs. It passes that sum through a transfer function to produce a numeric output. **Unpacked:** A neural network maps input patterns to output patterns in a complex way. Each neuron computes a function, and the network as a whole computes a complex function that can be considered a very wiggly mathematical function.	**LO:** Describe the following neural network architectures and their uses: feed-forward network, 2D convolutional network, recurrent network, generative adversarial network. **EU:** Feed-forward networks can learn arbitrary functions and are used for both classification an regression. 2D convolutional networks learn small "kernels" that are convolved with the input, and max-pooling layers to reduce image resolution; they are used for image analysis. Recurrent networks have feedback connections and are used for language processing. Generative adversarial networks have generator and discriminator modules and are used to create deepfakes.
Neural Networks *(Weight adjustment)* **3-B-ii**	**N/A**	**LO:** Demonstrate how weights are assigned in a neural network to produce a desired input/output behavior. **EU:**The behavior of a neural network can be altered by adjusting its weights.	**LO:** Demonstrate how a learning rule can be used to adjust the weights in a one-layer neural network. **EU**: During training, weights are adjusted in response to errors in the network's output, so that an error will be less likely when the input is seen again. **Unpacked:** Training can be done using binary units and a simple learning rule for adjusting the weights (such as the perceptron learning rule in the "Will this dog bite me?" exercise).	**LO:** Train a multilayer neural network using the backpropagation learning algorithm and describe how the weights of the neurons and the outputs of the hidden units change as a result of learning. **EU:** A neuron's weights start out as small random values and evolve to a more precise pattern through learning. The changes in the neuron's weights are computed by a learning rule driven by a back-propagated error signal. The neuron's weight pattern determines the features that the neuron detects. **Unpacked:** Students are not expected to know the details of the backpropagation learning algorithm, only that error is propagated backward from later layers to earlier ones. **Activity:** An online demo such as TensorFlow Playground can be used to visualize the changes in weights during learning.

Big Idea #3: Learning	*Computers can learn from data.*	LO = Learning Objective: What students should be able to do.	EU = Enduring Understanding: What students should know.	Unpacked descriptions are included when necessary to illustrate the LO or EU
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Datasets *(Feature sets)* **3-C-i**	**LO:** Create a labeled dataset with explicit features to illustrate how computers can learn to classify things like foods, movies, or toys. **EU:** We can get a computer to classify things by describing them in terms of feature values the computer can reason about. People choose the features. **Unpacked:** Classify food as healthy/unhealthy, or classify toys as safe or unsafe for babies. Optionally, they could build a decision tree using these features, but most important to understand is that they could feed their data to a machine learning algorithm to create the decision tree for them.	**LO:** Create a labeled dataset with explicit features of several types and use a machine learning tool to train a classifier on this data. **EU:** Feature types include discrete values (e.g., "New York", "Pennsylvania", "Iowa"), binary values (yes/no), and continuous values (age, height). **Unpacked:** Sites such as MachineLearningForKids will train decision tree classifiers based on data of this type.	**LO:** Create a dataset for training a decision tree classifier or predictor and explore the impact that different feature encodings have on the decision tree. **EU:** The choice of features to include, and the best encoding to use for these features, depends on the particular reasoning problem we are trying to solve. **Unpacked:** At each node of the decision tree, the learning algorithm tries to pick a feature that will be most helpful in separating the remaining instances into different classes. Features that don't correlate strongly with any class will not be chosen. **Resource:** MachineLearningForKids will draw the decision tree so students can inspect it and see which feature is referenced at each node.	**LO:** Compare two real world datasets in terms of the features they comprise and how those features are encoded. **EU:** Humans decide which features to include in a dataset and how to encode them. This can have consequences for machine learning algorithms trained on these datasets. **Unpacked:** age can be encoded in months (for pediatric datasets), years (for adults), or age ranges (infant, child, teenager, adult, senior, extreme elderly). Encoding a continuous variable using discrete values can guide the learning algorithm to make distinctions that conform to humans' understanding of the domain.
Datasets *(Large datasets)* **3-C-ii**	**N/A**	**LO**: Illustrate how training a classifier for a broad concept such as "dog" requires a large amount of data to capture the diversity of the domain. **EU:** Machine learning requires large amounts of data to be effective. To recognize dogs in images one must have not only many types of dogs, but also many different viewing angles and contexts. **Unpacked:** One way to help students visualize the diversity required would be to browse some of the standard datasets used for object recognition, such as ImageNet or Coco.	**LO:** Illustrate how objects in an image can be segmented and labeled to construct a training set for object recognition. **EU:** Machine learning requires large amounts of data to be effective. Human expertise is usually required to label the data, which could be labor intensive. **Unpacked:** Students can be given a set of images and asked to draw a bounding box around every person, dog, or traffic sign in the image, and label the object appropriately. As a follow-up, students could be asked to estimate the time it would take to construct a labeled dataset with several thousand examples.	**LO:** Evaluate a dataset used to train a real AI system by considering the size of the dataset, the way that the data were acquired and labeled, the storage required, and the estimated time to produce the dataset. **EU:** A large dataset is typically required to capture the diversity of a complex domain and narrow down the range of possible reasoner behaviors. There are multiple ways to construct, clean, and verify a dataset. There can be large costs associated with creating the dataset and processing the data. Labeling training data is labor-intensive and may require specialized expertise (e.g., spotting disease in x-rays.) Bias can be introduced during each step of dataset creation. **Unpacked:** Datasets for real-world problems may involve many features, and the defining characteristics of a class may involve compex relationships among these features. In order to narrow down the class to be learned and distinguish it from millions of other possible classes, the learning algorithm must see many examples. **Activity:** An activity can be done with common machine learning datasets found in repositories such as Kaggle, or publicly available demographic, economic, or environmental datasets.

Big Idea #3: Learning	*Computers can learn from data.*	LO = Learning Objective: What students should be able to do.	EU = Enduring Understanding: What students should know.	Unpacked descriptions are included when necessary to illustrate the LO or EU
Concept	**K-2**	**3-5**	**6-8**	**9-12**
Datasets *(Bias)* **3-C-iii**	**LO:** Examine a labeled dataset and identify problems in the data that could lead a computer to make incorrect predictions. **EU:** How well a computer learns to classify depends on the data used to train it. **Unpacked:** If examples of healthy foods are broccoli, green beans, peas, and spinach (all green), and unhealthy foods are donuts, cake, and candy bars, what will the computer conclude about green gummy bears?	**LO:** Examine features and labels of training data to detect potential sources of bias. **EU**: Machine learning algorthims require a representative collection of data in order to build an accurate model. Training datasets drawn from historical data may reflect pre-exisiting human and societal biases. **Unpacked:** Amazon's resume sorter learned a bias against female applicants because it was trained to mimic the statistics of past hiring history.	**LO:** Explain how the choice of training data shapes the behavior of the classifier, and how bias can be introduced if the training set is not properly balanced. **EU:** Bias can result if the model is asked to classify inputs that don't resemble the training data, or if the training data contains irrelevant correlations we don't want the classifier to rely on. **Unpacked:** A classifier trained on only Caucasian faces will do poorly on Black or Asian faces. A classifier trained on a loan application dataset where most of the rejected applicants lived in Pleasntville might decide to never make a loan to anyone who lives in Pleasantville.	**LO:** Investigate imbalances in training data in terms of gender, age, ethnicity, or other demographic variables that could result in a biased model, by using a data visiualization tool. **EU:** Machine learning algorithms will take advantage of any imbalances or correlations in the training set that help lower the error rate. If the dataset is not representative, those correlations can be misleading **Unpacked:** Data exploration to help students uncover imbalances or correlations can be done using histograms in Excel, or using any number of data visualization tools such as Pandas (for Python).