FROM NEURONS TO NETWORKS – GUIDE TO HANDS-ON ACTIVITIES

BY:
KATRIONA GUTHRIE-HONEA
AND ARNI

Artificial and Natural Intelligence

Introduction:

How do brains and computers make sense of the world? Every moment, your brain is taking in a flood of sights, sounds, and sensations, yet somehow it can quickly decide what you're looking at, where you're going, or what to do next. In recent years, artificial intelligence (AI) systems have been built to do something similar: detect patterns in data, combine them, and learn to make decisions. While the details are different, both biological and artificial "neural networks" rely on simple building blocks working together to create surprisingly complex abilities.

This booklet introduces two hands-on activities—**Picture This** and **Layering it On**—that invite kids to step into the role of both neurons and computer scientists. In Picture This, students play the part of AI "detectives," piecing together clues to identify mystery animals, much like a neural network refines its guesses through feedback and training. In Layering it On, they use origami-style puzzles to discover how adding "layers" helps simple decision-making units solve more complicated problems, just as brains and AI models use multiple processing stages to tackle tough tasks.

Why is this important? Neuroscience and AI are rapidly transforming the future of science, technology, and everyday life. By giving kids playful, hands-on experiences with these ideas, we help them see that complex systems can be broken down into understandable parts—and that they, too, can explore and ask questions, test ideas, and discover how intelligence works. These activities aim to spark curiosity, build confidence, and open doors for the next generation of scientists, engineers, and critical thinkers.

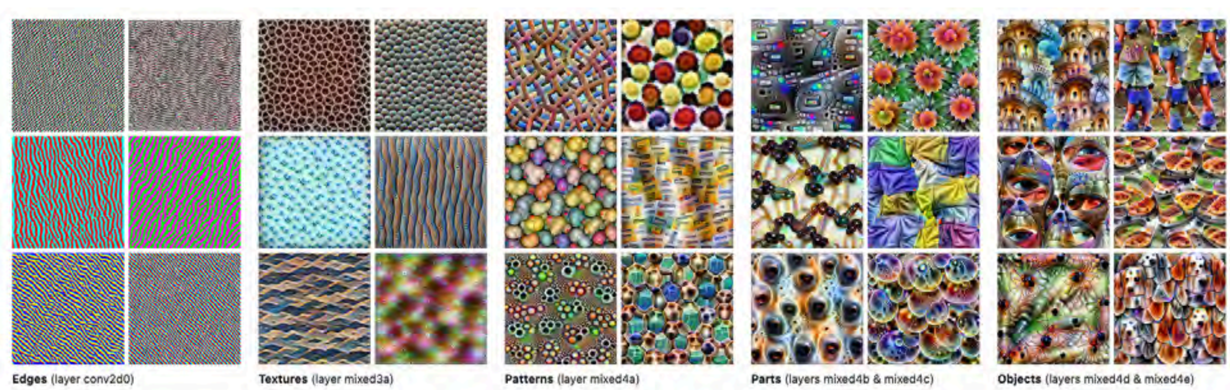
INSTRUCTOR GUIDE

<u>ACTIVITY TITLE</u>: Picture This!

<u>Theme</u>:	Help students intuitively understand how AI image recognition builds up from simple visual information to full object identification, and that guesses become more accurate as more features are combined.
<u>Goal:</u>	Students should learn that computer scientists can design systems that "guess" based on information we give them. These systems use features—simple traits like color, size, or number of legs—to make a decision. They also need to understand that not all features are equally helpful, and that improving guesses often means refining which features we use.

Image classifiers detect a whole variety of patterns to do their job, from simple to very complex

*(from "*Feature Visualization*," linked in* **Resources** *below)*

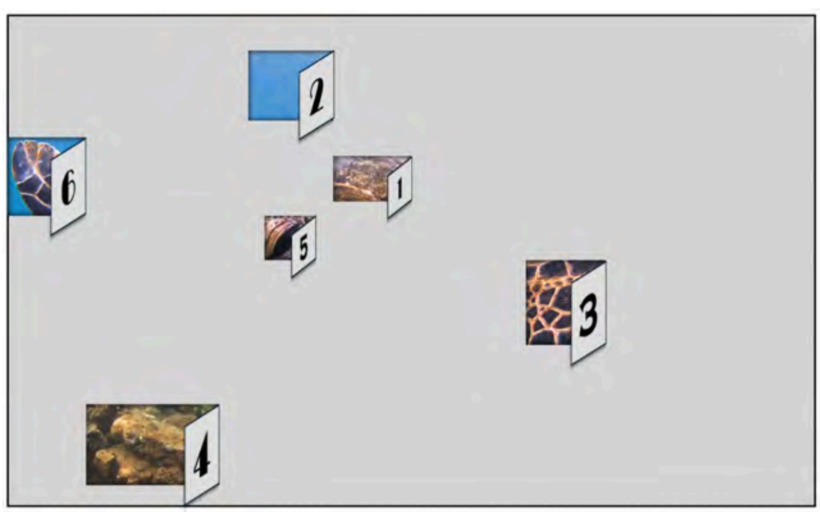


<u>LESSON OUTLINE</u>:

1. Introduction: *Plan a script of what you will say to start.* *- What will this be about? Why's it interesting? (Hook)*	When you look around the room, you typically have no issue identifying the people and objects that you see. But, though that process *feels* easy, it took a long time for computer scientists to teach computers to do the same! Today we'll learn how artificial intelligence uses concepts discovered in our very own brains to recognize objects in all sorts of pictures.

2. Building Background: *List questions you can use to immediately engage your audience and prepare their thinking for your activity.*	**To start, why might we want to teach computers to "see" anyway?** *May get various answers from students, but some include: facial recognition, reading handwriting automatically (like sorting letters at the post office!), self-driving cars, etc.* Who do you think is better at "seeing", humans or computers? **With that in mind, why would we want to learn how sight works in**

-What prior knowledge might they have about/related to your topic? *-What prior knowledge (background) do they need for your activity?*	**humans?** It can be funny to think about, but really the only way we know the world outside of us exists at all is by sensing it. But those senses only mean something when they reach the rest of our brains — as far as our eyes know, we are walking around among blobs of light in all sorts of colors! Today, we'll think more about how detecting patterns in that light can lead to recognizing more complex objects and the concepts that match them, like "dog," "cat," or even "human." The more we know about this process, the more we can build it into computers to help them see too. **What's happening inside a computer when it tries to recognize a picture?** That's where neural networks come in. These systems—loosely inspired by the brain—look at a picture and try to identify useful features. These features could be anything from "fuzzy texture" to "four legs" to "bright orange color." But at first, the computer doesn't know which ones are useful. It has to learn by trying, messing up, and adjusting—just like we do.

3. Lesson & Activity - Version 1 (age 4+): *Outline the key components of your lesson.* **<u>Plan/Note</u>**: - key ideas/ vocabulary - scaffolds - images/media - extension questions <u>*Consider how to best deliver your content!</u> <u>*Plan interactive components that encourage active thinking in your students.</u>	<u>Explanation:</u> **In this activity, you'll play an AI detective. We have an animal hidden here and we'll reveal one small part of the image at a time. Each flap you open gives your brain a new clue to reveal simple patterns - called features- which help refine your guess.** ***Set Up:*** *Before the event, prepare several images with a paper covering them and pre cut-out flaps that will expose certain areas of the image. See example with a sea turtle image below.* 

Step 1: Feature selection

Open the flaps one by one (in numbered order so you're slowly revealing more information as you guy). Ask the participant if they can guess what animal it is.

Questions to ask as they turn over flaps:

- Can you guess what animal this is? Even if they guess correctly, ask what else it could be.
- What do you see now that you didn't before?"
- What other clues do you need to guess if it's a [previous guess]?

Remind them that computers work the same way, that the more clues, or **features** a computer has, the better it will be at guessing

Note: For older kids, you can ask for each flap, does this show an edge, a texture, pattern, parts, or object?

<table>
<tr>
<td>

3. Lesson & Activity - Version 2 (age 6+)
Outline the key components of your lesson.

Plan/Note:
- key ideas/ vocabulary
- scaffolds
- images/media
- extension questions

*Consider how to best deliver your content!
*Plan interactive components that encourage active thinking in your students.

</td>
<td>

Explanation:
Today you'll play the role of computer scientist training an AI system.

One of you will be the computer scientist, and the other will be the computer. The scientist will look at a mystery image and give clues—like how many legs or what color the animal is—to help the computer make a guess. But not all clues are equally helpful, so we'll have to figure out which ones matter most. That's how real AI models learn, too!

Note: For younger participants you can ignore weights
For older participants, you can incorporate explanations of how computers start by recognizing edges, then texture, then patterns, and so on. You can also explain how computers weight texture more heavily in image recognition than humans do, and humans weight shape more heavily than texture.

*Keep the set of **mystery images** hidden from students at all times, until ready to reveal.*

Set Up: Split students into "computer scientist" and "AI" groups. Have index cards for writing clues ready

Step 1: Feature Selection

1. Choose a mystery image and carefully show it to the computer scientist group
2. Ask the computer scientist to choose and write down **3-4 features** of the image. For example, what color is it? How tall is it? How many legs does it have? Etc. This will be up to their imagination! *(**NOTE:** ideally, these features will all be visual, in keeping with the theme of image classification (i.e., not "what they eat"). However, this is flexible per the age/level of students.)*

Step 2 (Optional): Weighting

3. Ask the computer scientist to put the clues in the order that they think they're most important.

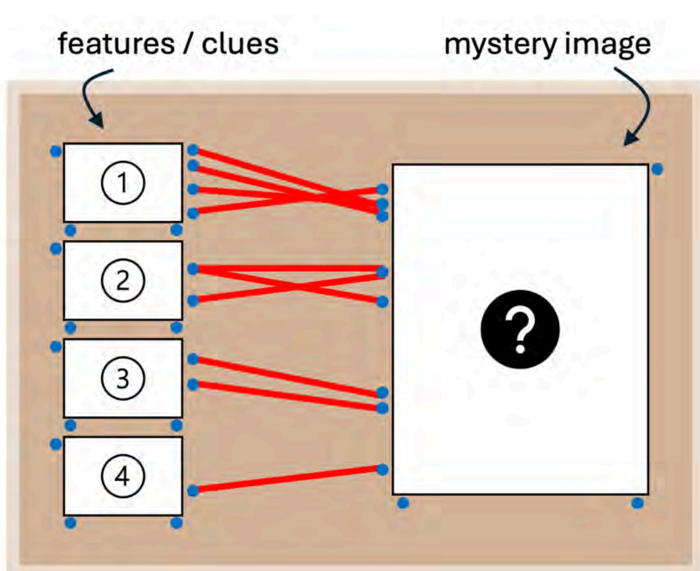
Step 3: Guess

4. Have the computer group guess the image. If they get it right, switch groups, if not, go to retraining

Step 4: Feature refinement (training)

5. "This is exactly what happens when training a network. The network will guess incorrectly and that will tell the computer scientist that they have to change the clues. That's where **training the network** comes in. We can do this in two ways: first,

</td>
</tr>
</table>

	by detecting **new features**, and second, [optional] by **altering the weights** of the features. 6. Have the computer scientists think of **one more feature** (or, if they are stuck, help them think of one) to replace/add the lowest-weighted (least useful) feature. Then, if needed, **rearrange features** in weighted order, in case the new feature is very useful for identification. 7. Repeat the "clue reveal" process by adding the value of the new feature to the board, and ask the detective to guess once more. 8. If incorrect, and if time allows, this refinement process can happen once more, or until the animal is guessed. Otherwise, just reveal the animal — "so close!"
3. Lesson & Activity - Version 3 (age 12+): *Outline the key components of your lesson.* **<u>Plan/Note</u>**: - key ideas/ vocabulary - scaffolds - images/media - extension questions <u>*Consider how to best deliver your content!</u> <u>*Plan interactive components that encourage active thinking in your students.</u>	<u>Explanation:</u> **Your task will be to play the role of an AI "detective," where you'll use simple patterns — called *features* — to try to guess the identity of a mystery animal.** ***Set Up:*** *Keep the set of **mystery images** hidden from students at all times, until ready to reveal. Each activity leader will hold a **corkboard**, where 3-4 "slots" made from pinned thumbtacks will be able to hold one index card each. Those slots are each connected by varying amounts of red string (the "detective" theme) to one last slot, which will hold the mystery animal image when revealed. Place the **index cards** and **writing utensils** (markers, crayons, pencils, etc.) in a place where both students and activity leaders can write on them.* 

	Step 1: Feature selection 1. Ask the student (now, "detective") to pick a number, 1-8. Shuffle the **mystery images** and use that number. Place it on the slot of the corkboard, but facing inward, hidden from view. 2. Ask the detective to write **3-4 features** of the image, each on one index card, that might be useful to help identify it. For example, what color is it? How tall is it? How many legs does it have? Etc. This will be up to their imagination! (Offer to write for students too young to do so.) *(**NOTE:** ideally, these features will all be visual, in keeping with the theme of image classification (i.e., not "what they eat"). However, this is flexible per the age/level of students.)* 3. Collect the cards, and based on your discretion, place them in slots according to **how useful** each feature might be. Here, the slots with more red string connecting them to the mystery image slot are more useful — they have higher **weights** in the network. *(Example: "color" may be less useful, as this can vary for many animals, than something like "number of legs.")* **Step 2: Feature detection & guess** 4. Next, write on new index cards the actual value of each of the features (*4 legs, orange color, etc.*), and place each in the corresponding slot. These are the clues the detective now uses. 5. Remind the detective that the amount of red string tells them how important each feature is, and have them make a guess! If correct, play again. If not, move to Step 3 (more likely). **Step 3: Feature refinement (training)** 6. There are lots of animals out there, so students likely won't guess correctly on the first go. That's where **training the network** comes in. We can do this in two ways: first, by detecting **new features**, and second, by **altering the weights** of the features. 7. Tell the detective that they'll get one more chance to guess, after two changes are made. Have them think of **one more feature** (or, if they are stuck, help them think of one) to replace the lowest-weighted (least useful) feature. Then, if needed, **rearrange features** in weighted order, in case the new feature is very useful for identification. *(**NOTE:** Alternatively, could add more red string to increase weight of certain features, though feasibility here depends on logistics & throughput of activity.)* 8. Repeat the "clue reveal" process by adding the value of the new feature to the board, and ask the detective to guess once more. 9. If incorrect, and if time allows, this refinement process can happen once more, or until the animal is guessed. Otherwise, just reveal the animal — "so close!"

4. Wrap Up: *- Review key ideas* *- Share takeaways and final thoughts* *- Discuss connections to other questions and ideas. Extensions.* *- Ask: Who could you teach what you learned here today?* *- Ask/Suggest: What can I do to learn more?*	**Question/review: Why did it matter which feature/clue was assigned which weight?** - In the detective's case, some clues are more helpful than others — number of legs being more useful than color, for example. Similarly, in networks designed and taught to identify images, some features matter more than others. - In a network, weights are the strength of the connection from one unit (a neuron, or node in a computer network) to another. The stronger the weight, the more easily the first unit will pass a message on to the second. **Question/review: What is the importance of training the network?** - A big part of the process of teaching a computer how to "see" is to give it many examples of the types of objects it should identify, and have it practice over and over again. Its first guesses will be quite bad, but over time, it will refine both features and weights to get better and better — it learns *what* to look for, and *how important* each of those things is. **Closing:** Show students the **Example** sheet, which depicts a network at top, and the features detected by different layers of that network at bottom. Point out how, as we move from sensing to identification (example of seeing an eye, and actually identifying "eye"), the features detected by each layer get more and more complex. Near the end, even whole objects start to be detected in these artificial neurons, built out of signals from neurons that detect much simpler patterns. Today, the students played the role of one layer of this very complex process. **Final thought: This process was inspired by the brain! We, too, have cells that first respond to simple inputs (lines, edges) and eventually to more complicated features. This network works in the blink of an eye (ha!) whenever you identify something you see.**

MATERIALS NEEDED: ***(please list all items and quantities necessary for preparation)***

- **Per table / activity leader**
  - Potentially a cork board to pin clues in order to

- **Per student**
  - 8-10x **index cards** (4-5x used by student)
  - **writing utensils** (markers, crayons, pencils, etc.)

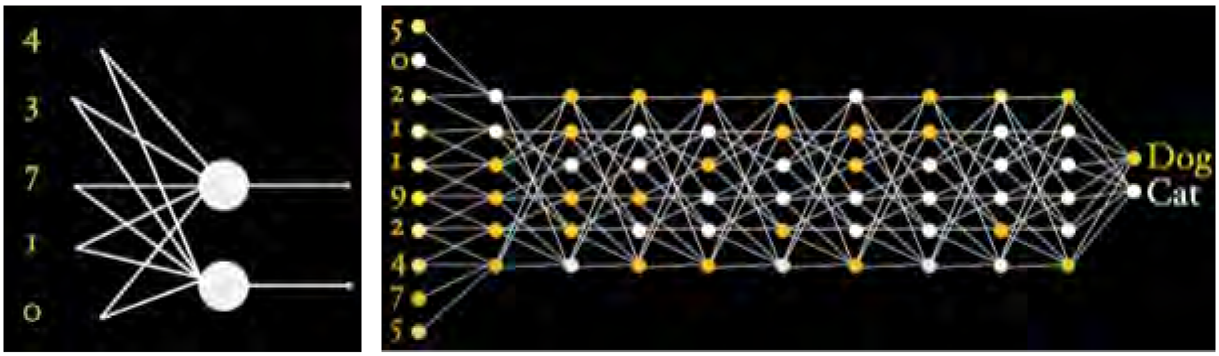
Resources:

- For more information for activity leaders: “Feature Visualization,” https://distill.pub/2017/feature-visualization/
- If using in a classroom setting, with access to a computer / projector, introduce the topic with this example image classifier: https://demo.viktor.ai/public/image-classification. It has been trained to discriminate between a set of several objects/animals; you can find some online, upload them, and show students how the network generates guesses (with respective % confidence levels) for the identification of the image uploaded.

INSTRUCTOR GUIDE

ACTIVITY TITLE: Layering It On

Theme:	What does it mean for a network in a brain or computer to have “layers”? Fold and snip your way through this origami-like challenge to learn more!



Examples of a 1-layer vs. many-layer neural network — what good are all those extra layers?
*(from “*Why Neural Networks Need Layers,*” linked in* **Resources** *below)*

LESSON OUTLINE:

1. Introduction: *Plan a script of what you will say to start.* *- What will this be about? Why’s it interesting? (Hook)*	Computers and nervous systems alike are made up of networks: tiny units that talk to other tiny units, that talk to other tiny units, and so on and so on to eventually get something done. Today, we’ll learn more about what these networks are actually made of, how their individual pieces “talk” to one another, and how we get complicated outcomes from very simple processes. We’ll use some fun origami-like brain teasers to really drive this point home!

2. Building Background: *List questions you can use to immediately engage your audience and prepare their thinking for your activity.* *-What prior knowledge might they have about/related to your topic?* *-What prior knowledge (background) do they need for your activity?*	**To start, can anyone tell me what a neuron is?** Neurons are the main type of cell in the brain and nervous system. Different kinds of neurons can store information, sense input from the outside world, and make our bodies move, but they do all of this by talking to one another. **What about a “network”? What does it mean for something to be in a network?** When neuroscientists talk about networks, we are referring to big webs of neurons that all talk to each other, so information can move through the *network* of those neurons in order to sense something or do something. For example, there’s a network of neurons that connects from your eyes to several different parts of your brain to eventually recognize the objects and people in your field of view. Messages start off in neurons in your eyes, detecting light, and they send those messages deeper and deeper into the brain. Meanwhile, if you want to grab

	something sitting in front of you, another network can send a message from your brain to your hand in order to do just that. **When a neuron "fires," what does that mean?** In a network of neurons, each cell is generally "on" or "off" — either it sends a signal, called an **action potential**, or it doesn't. In turn, the next neuron in the network either turns on or off (fires or doesn't) in response to that signal, and so on. Very simple! In computers, a similar thing occurs: tiny switches called **transistors** are either "on" (1) or "off" (0), which is why computer code at its simplest level is a bunch of 1s and 0s. But out of that, clever designers have built everything that computers can do! **Keep this in mind: brains and computers both do very complex things with very simple "on-off" code. Today we're exploring one of the tricks they both use to make that happen: organizing their components in <u>layers</u>.**

3. Lesson & Activity: *Outline the key components of your lesson.* **<u>Plan/Note</u>**: - key ideas/ vocabulary - scaffolds - images/media - extension questions <u>*Consider how to best deliver your content!</u> <u>*Plan interactive components that encourage active thinking in your students.</u>	***Set Up:** Have **Demo** and **Keys** available to be shown to students as needed. Have a stack of handouts to provide **Handouts 1, 2, & 3** to each student. Make **safety scissors** and **recycling bin** accessible at table. (If students are too young to use scissors, offer to use them instead, or demonstrate the activity rather than having them attempt it themselves — or ask their guardian to give it a shot!)* ______________________ **Part 1: Demo + Handout 1** 1. Ask students to look at the sheet. What do they see? *(Red cherries; green apples)* 2. Tell them to imagine a neuron that detects cherries. It turns ON if it detects cherries, but OFF if it detects anything else, like apples. Ask: can you imagine cutting **one straight line** across the sheet to do this? *(yes, easy: diagonal line from bottom left to top right)* 3. Great! Exactly. Above/left of this line, the neuron would be ON, detecting the cherries, and below/right of this line, it would be OFF. If we cut there, we'd cleanly separate out the cherries. 4. Let's try again! For **Handout 1**, now imagine a neuron that detects things in a garden. And this time, we really get to cut! Can you cut **one straight line** that separates the garden objects from everything else? *(see **Key**; straight line across, near bottom)* 5. You've got the hang of it! But let's make things more complicated. ***NOTE:** Here, for younger students, you may jump to **Handout 3** and use it as a demonstration (sort of like a magic trick), showing that a few*

	*simple folds, and one cut, can detect the dog, or animals that live underwater, or animals with wings. Then, return to **Handout 2** to see if they can solve that simpler puzzle on their own, now knowing that it takes a fold to do so. For older students, continue in the lesson order listed here.* **Part 2: Handout 2** 6. Ask: this time, let's imagine the neuron is detecting things that have wheels. On **Handout 2,** which of these have wheels? *(bicycle, car, truck, train, & bus)* 7. Ask: can you think of a way to use **one straight line** (our **one** neuron having **one** output, ON or OFF) to separate the wheeled group? *(without folding, no!)* 8. Ah! Here is where a network trick comes up. What if we change the "input" to our neuron. What if we **fold** the paper? Ask: can you **fold** the sheet and then make **one straight line cut** to detect the wheeled vehicles? *(see **Key**; fold in half, then cut across)* 9. What we did there was to add a **<u>layer</u> to our network** — rather than just showing our detector neuron the flat piece of paper, we changed that input in an in-between step, called a **layer**, so we could detect all wheels with a simple ON/OFF split. Nice work! **Part 3: Handout 3** 10. Lastly, here's the real challenge. We have lots of animals to detect, and only one neuron to do it! Let's get to **folding (layering)** to complete these tasks. ● ***NOTE:*** *These challenges increase in order of difficulty. All 3 can be completed on the same handout (cuts won't intersect), but students will likely need to make several attempts at any/all of these. Several spare copies of each handout per student are highly recommended.* ● **Challenge A:** Detect the **dog**. *(**Key**; fold to form a cross through dog, then snip off corner)* ● **Challenge B:** Detect the animals who **live underwater**. *(see **Key**; fold in half to align fish & dolphins; fold across to align corners; cut off corners)* ● **Challenge C:** Detect animals that **have wings**. *(see **Key**; fold diagonally across to overlap penguin & chicken, then fold twice to form cross as in **A**, then snip off corner)*

4. Wrap Up: *- Review key ideas* *- Share takeaways and final thoughts* *- Discuss connections to other questions and ideas.*	**Question/Review: Why is folding (layering) useful?** - It is a more efficient way of making complex computations with simple ON/OFF units. Even though vehicles with wheels or animals with wings were mixed up with other vehicles or animals in Handouts 2 & 3, we combined our folds — each of which itself

Extensions. *- <u>Ask</u>: Who could you teach what you learned here today?* *- <u>Ask/Suggest</u>: What can I do to learn more?*	is an ON/OFF decision! — in clever ways to nonetheless detect them with one simple cut. - We *could* make a bunch of individual cuts to carefully outline each set, but if we think of each cut as the decision of an individual neuron, that's a lot more neurons needed! **Final note: What does a layer look like in the brain, or a computer?** - A layer is just one extra step from input to output (see **Example** sheet). It's all about combining simple units in clever ways to produce a complex outcome!

MATERIALS NEEDED: ***(please list all items and quantities necessary for preparation)***

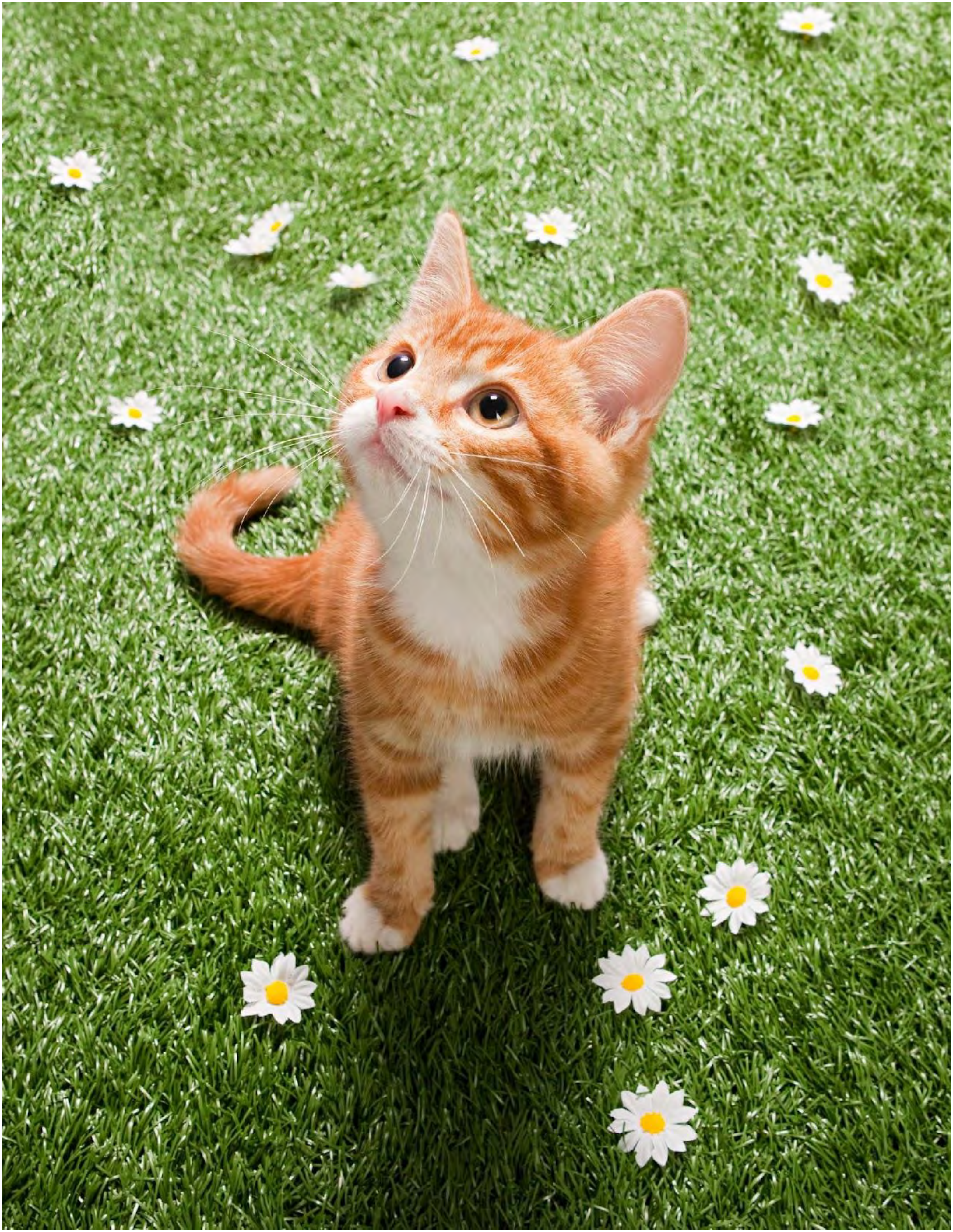
- **Per table / activity leader**
  - 1-2x **Demo** sheet
  - 2-3x **Key** sheets (to show students as needed)
  - 1-2x **Example** sheets
  - **Encouraged: Recycling bin** nearby for paper scraps (there will be a lot!)

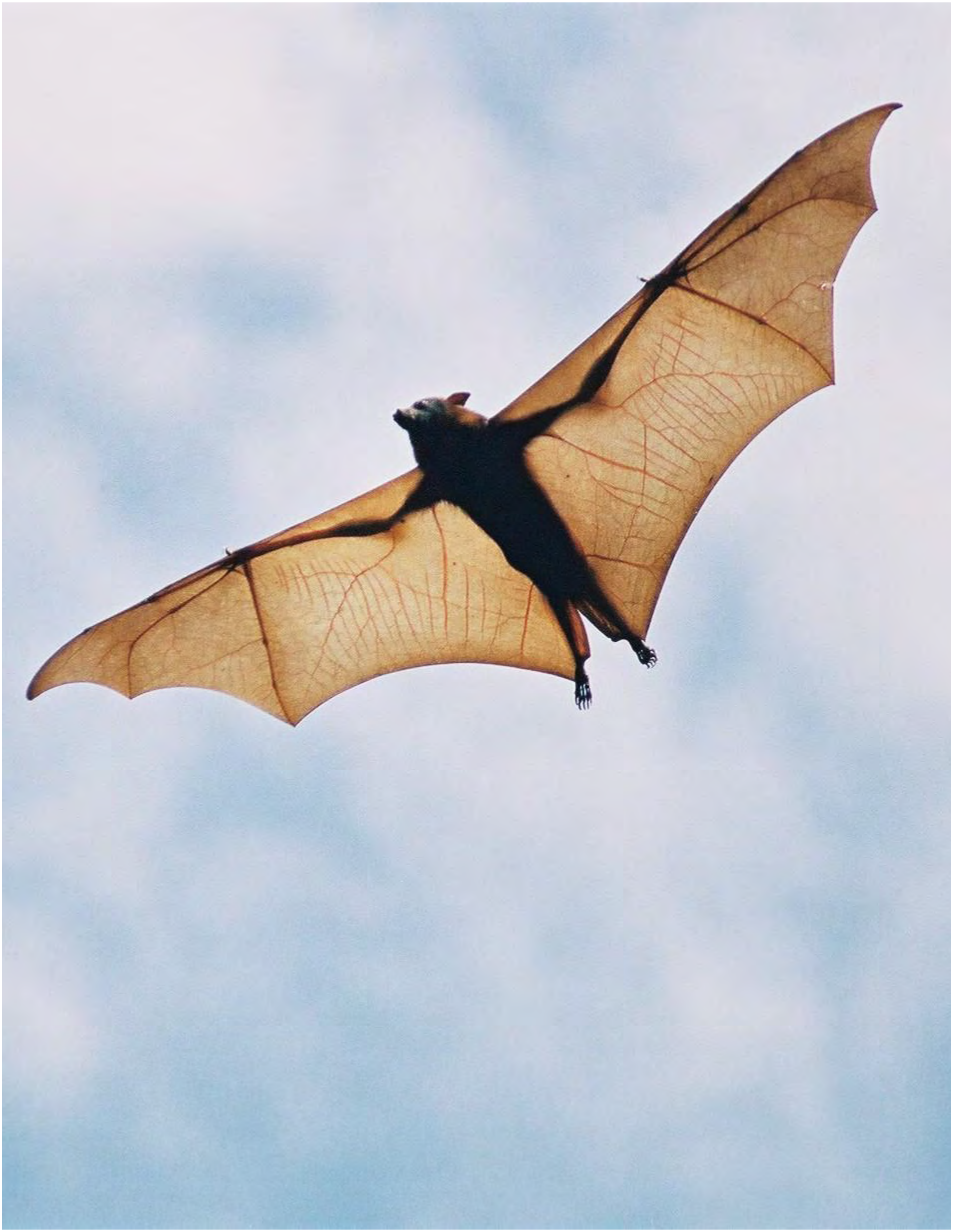
- **Per student**
  - 1x **Handout 1**
  - 1x **Handout 2**
  - 2-3x **Handout 3** (will likely take multiple attempts to solve)
  - 1x pair of **safety scissors**

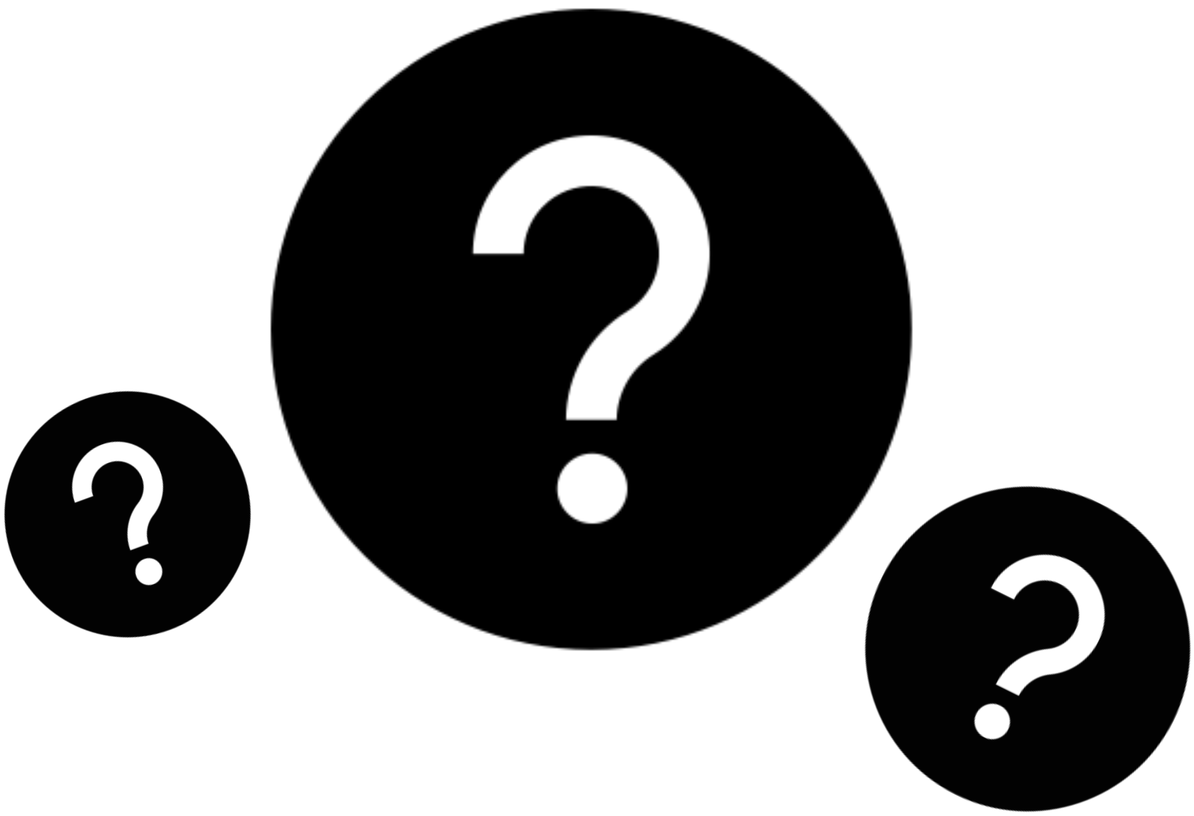
Resources:

- For more information for activity leaders: "**Why Neural Network Size Matters**," https://www.youtube.com/watch?v=e5xKayCBOeU

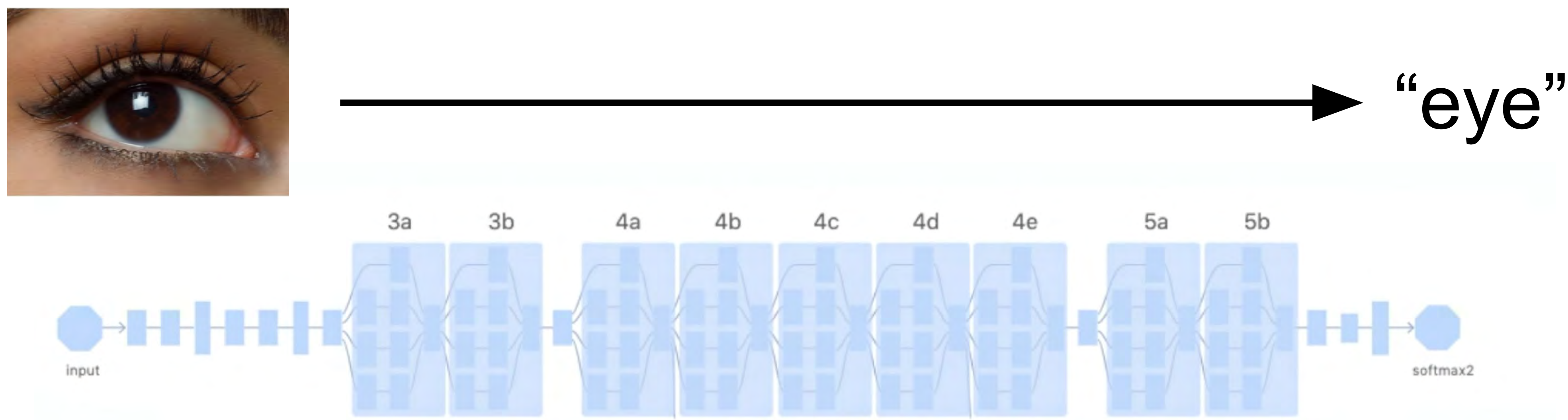
MYSTERY IMAGES



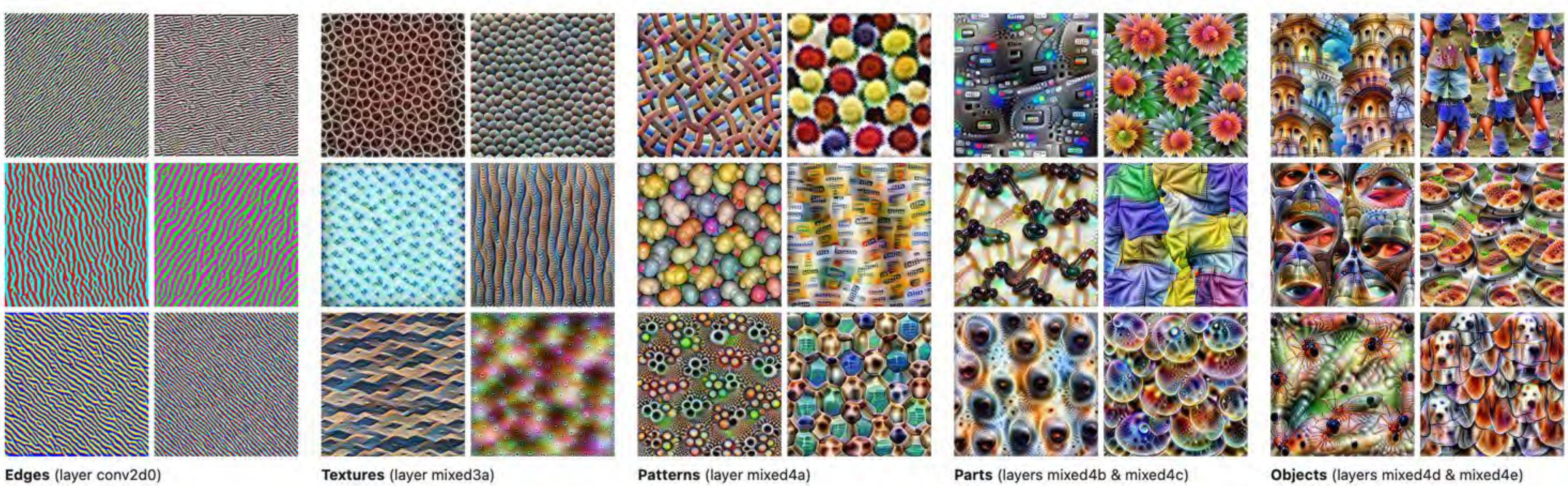




EXAMPLE

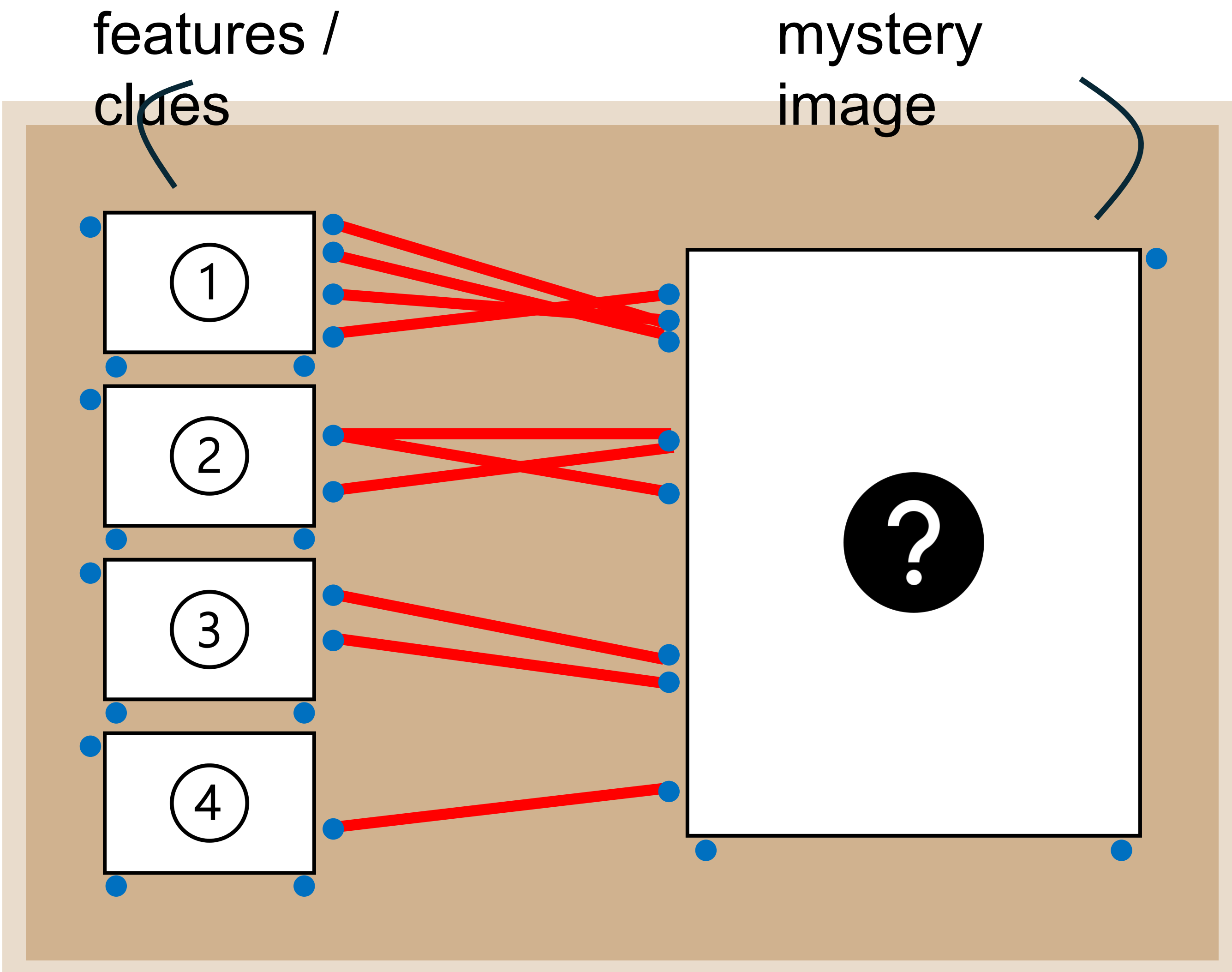


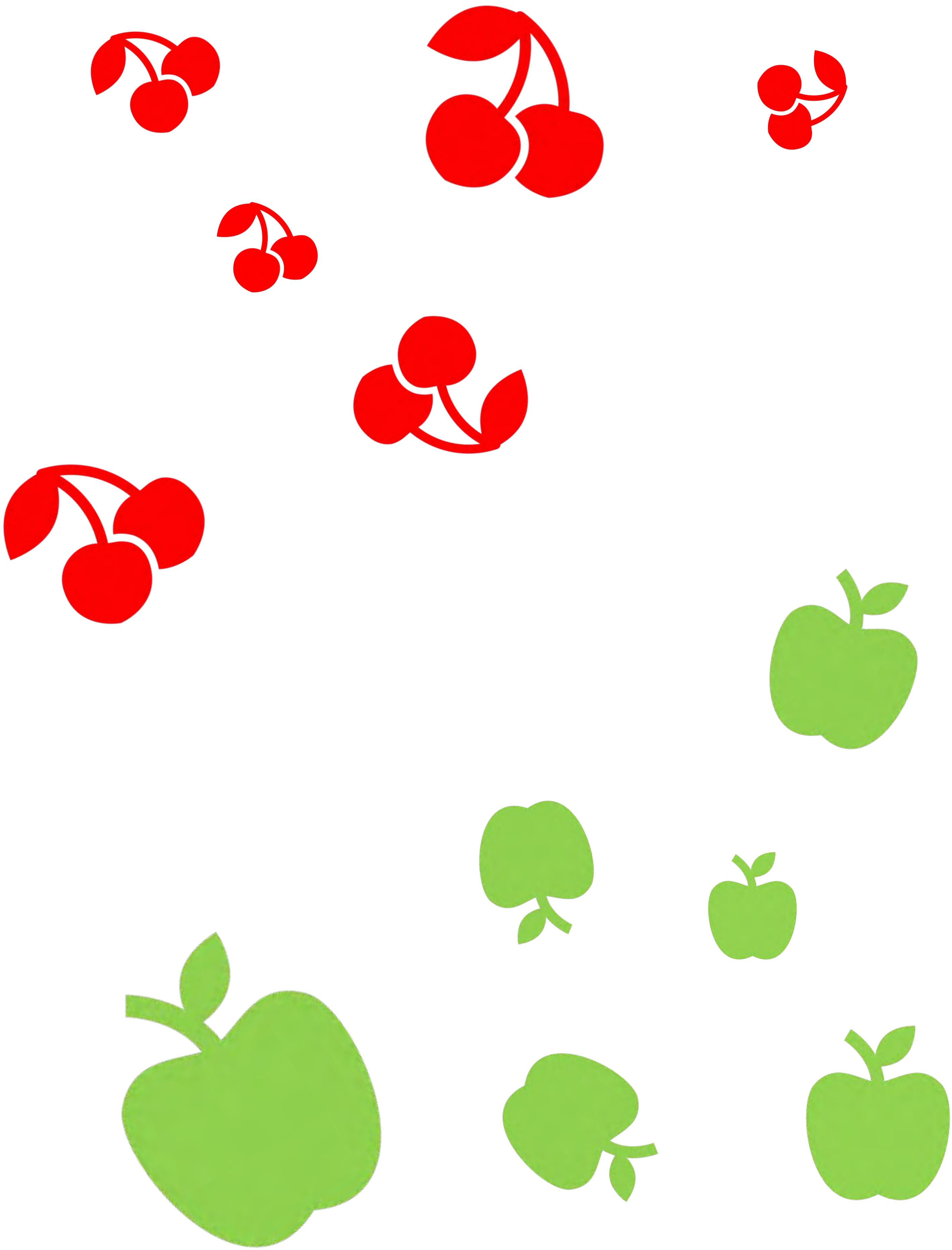
Source: "Why Neural Networks Need Layers" *Art of the Problem (YouTube)*



Source: "Feature Visualization" *(Distill.pub)*

(corkboard layout)





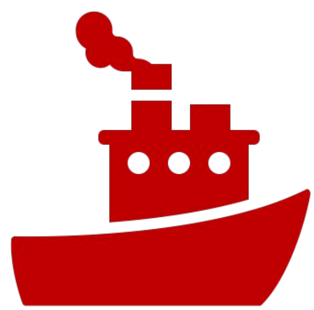
HANDOUT 1

Print 2-sided with **short edge flipping**

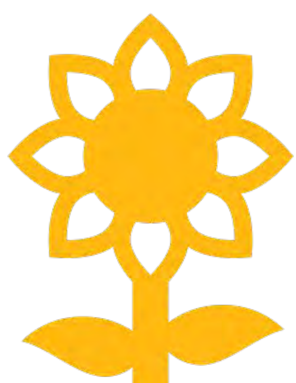


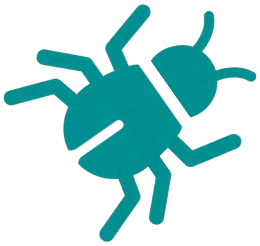
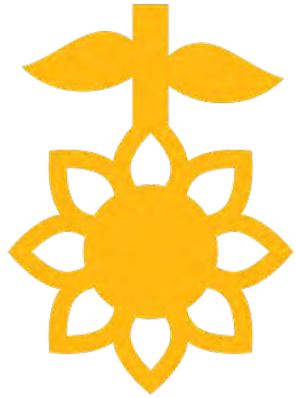






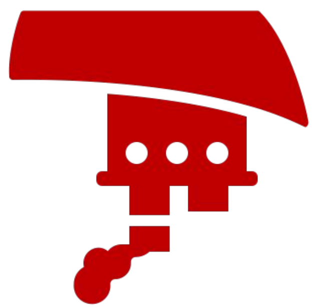
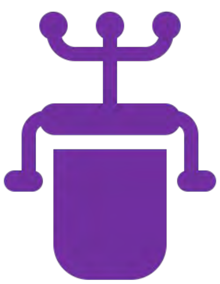
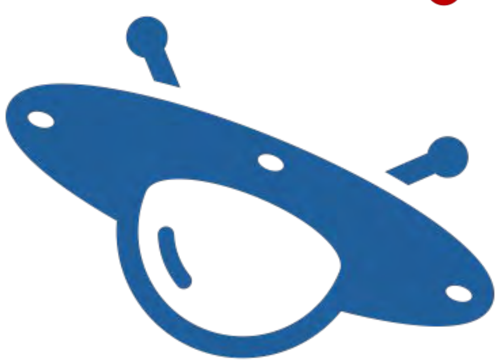


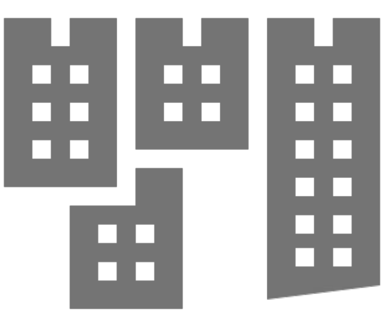







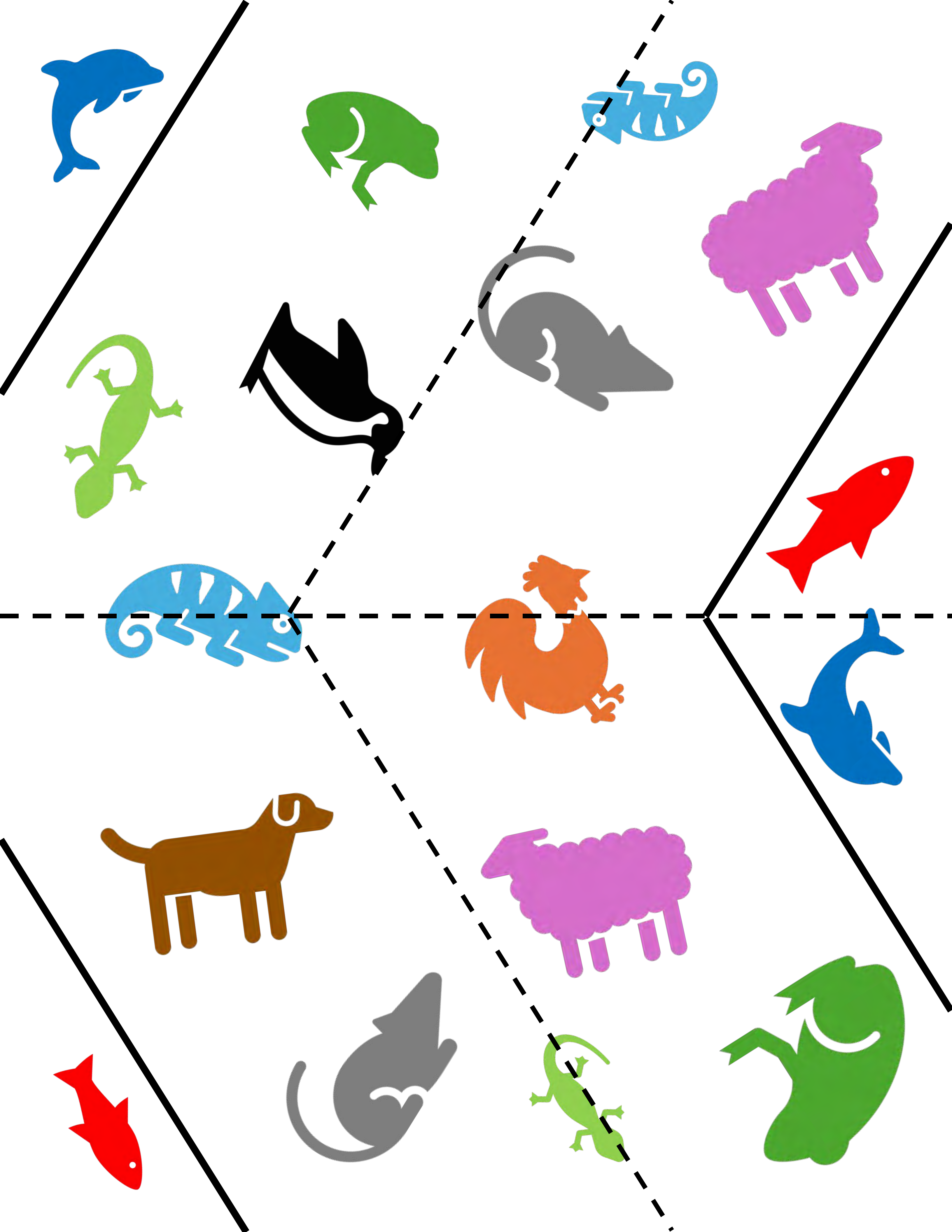


HANDOUT 2

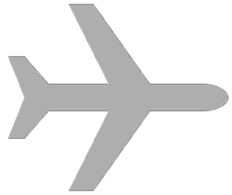
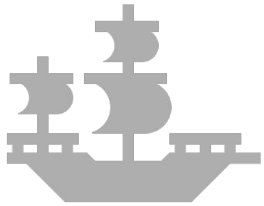
Print 2-sided with **short edge flipping**

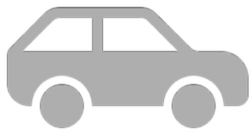
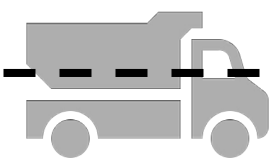
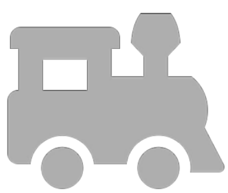
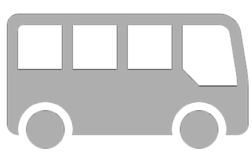
HANDOUT 3

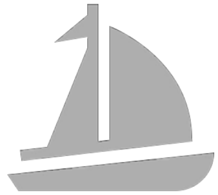
Print 2-sided with **short edge flipping**

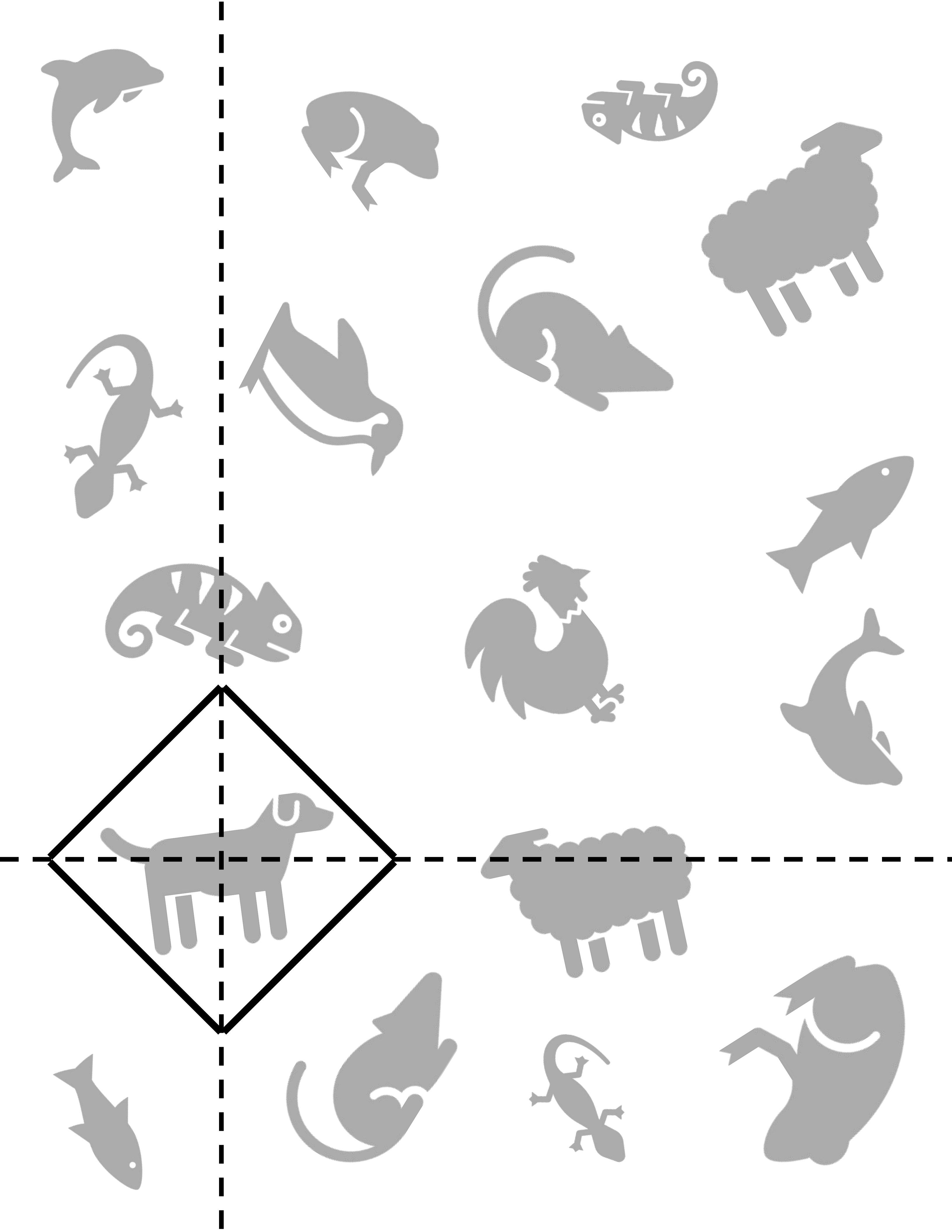


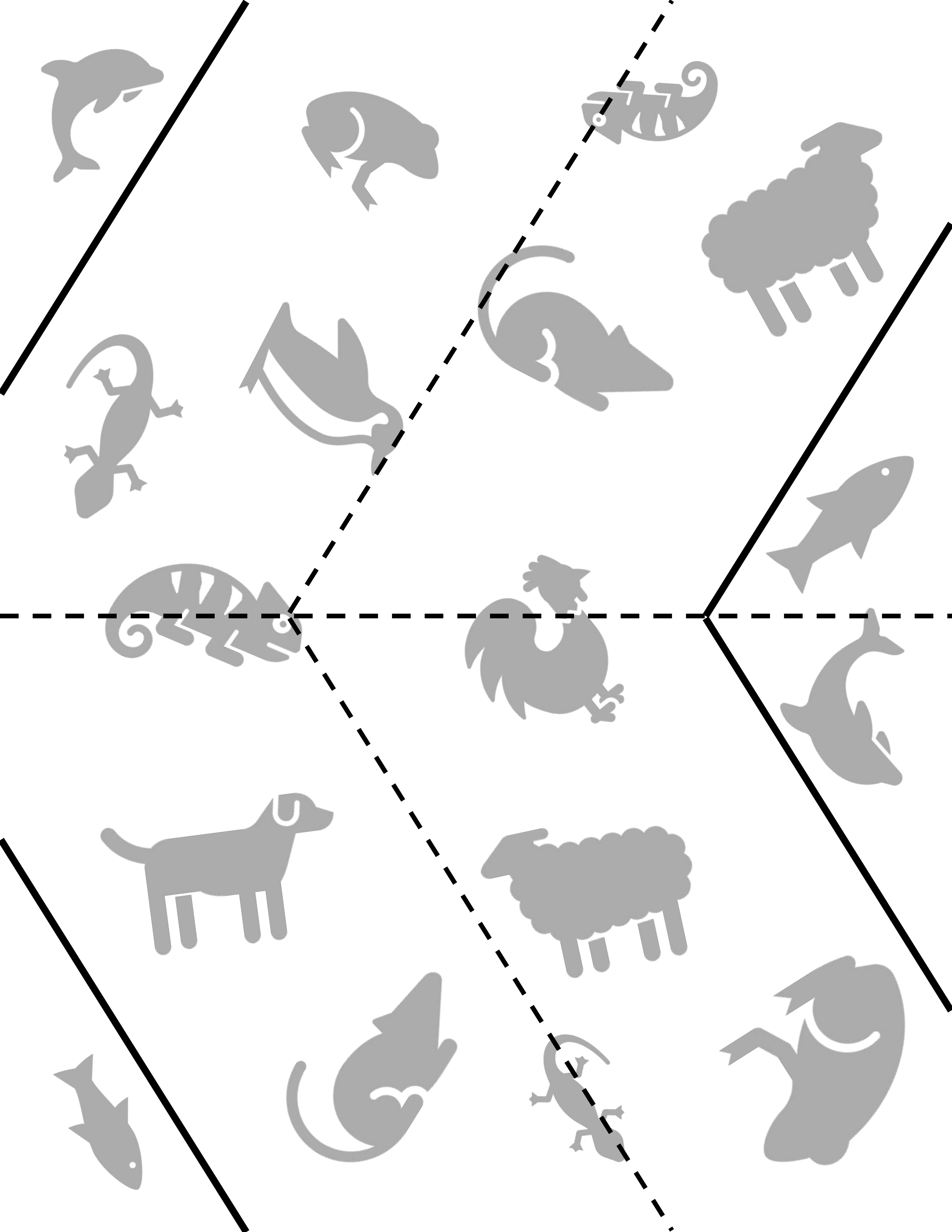
KEY

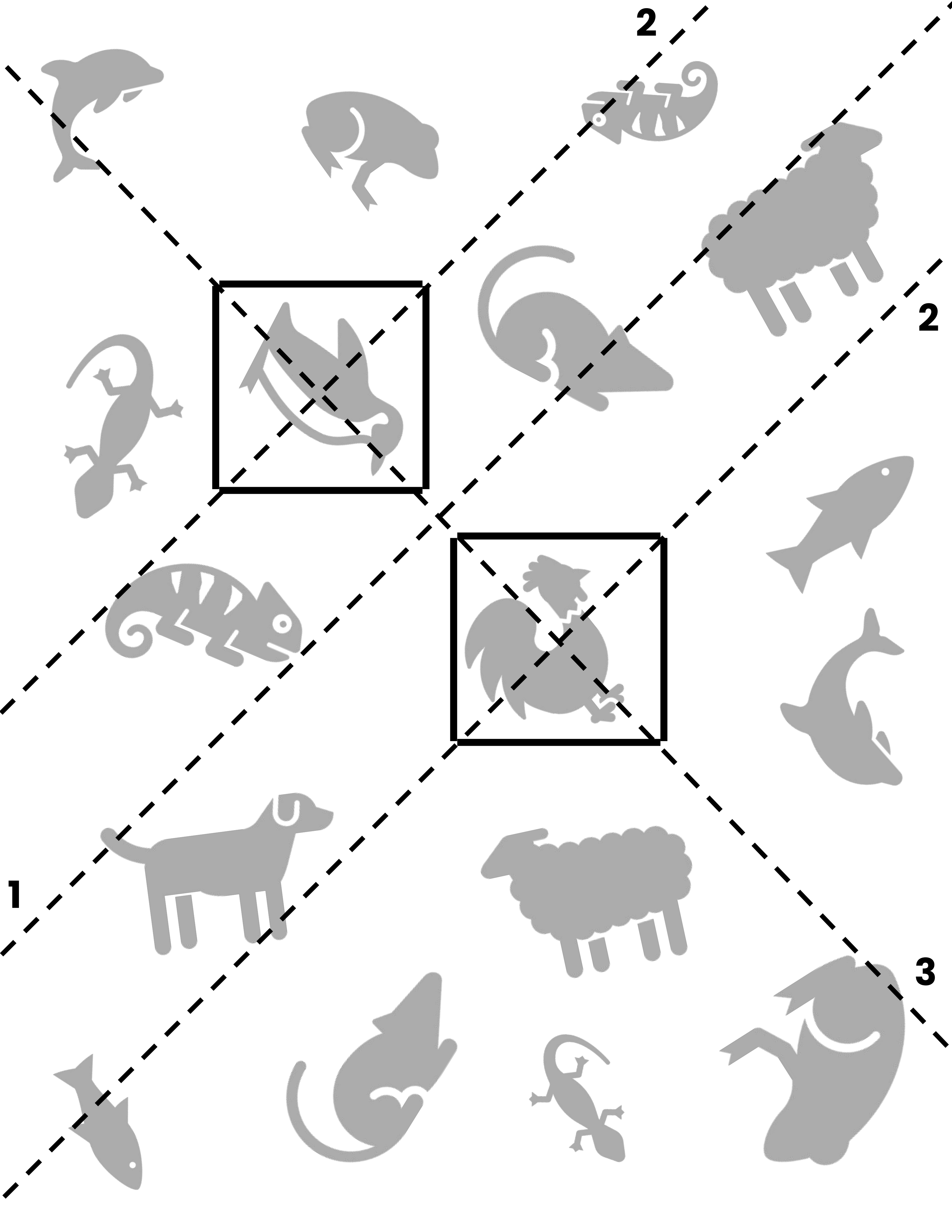





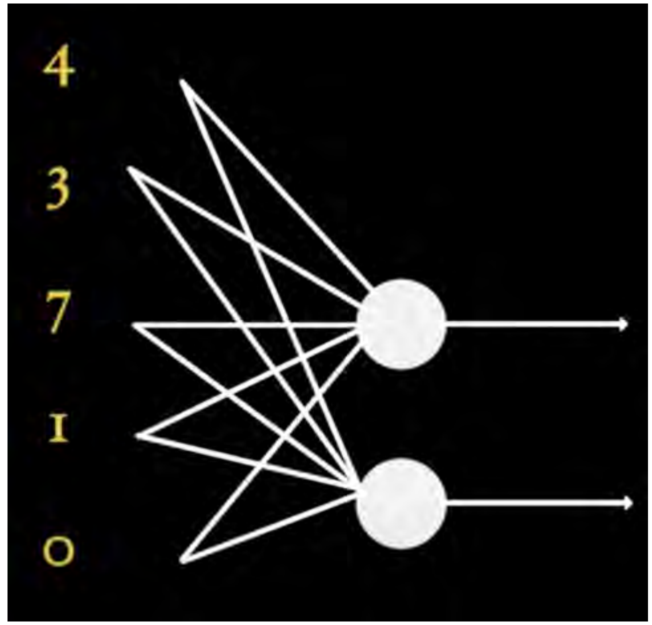


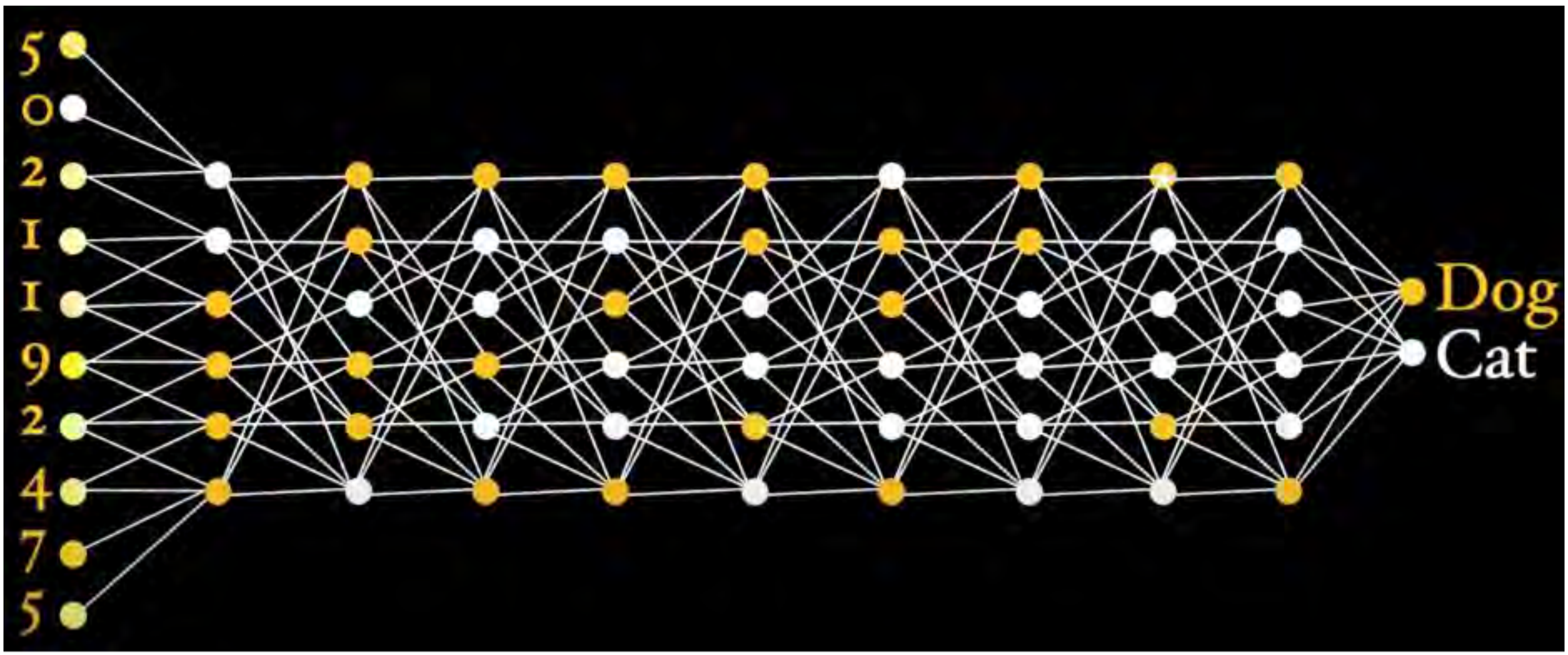




2
2
1
3

EXAMPLE





Source: “Why Neural Networks Need Layers” *Art of the Problem (YouTube)*