Machine learning for **artists**

Gabriel Vigliensoni-Martin
gabriel@vigliensoni.com
Distributed Digital Music Archives & Libraries Lab
Schulich School of Music
McGill University

Learning machines
for **artists**

Gabriel Vigliensoni-Martin
gabriel@vigliensoni.com
Distributed Digital Music Archives & Libraries Lab
Schulich School of Music
McGill University

Machine learning

- What is learning?
  - Memorizing vs learning
- Machine learning
  - Science of getting computers to act without being explicitly programmed
  - Study of algorithms that learn from examples and experiences instead of hard-coded rules
- Machine learning tasks
  - Classification: output is discrete categories (e.g., classification of a handwritten character into any of 26 different letters it represents)
  - Regression: output is a real-value number (e.g., prediction of future values based on past values)

Machine learning

http://www.gestaltu.com/2015/10/apples-and-oranges-a-random-portfolio-case-study.html/

Machine learning

- How to write code to tell the difference between an orange and an apple?
  - Take an input image, do some analysis, and outputs the types of fruit
  - Writing lots of manual rules
    - Count how many green pixels there are in the image and compare that to the number of red pixels? The ratio should give a hint about the type of fruit.
    - Works fine for simple images, but the world is messy and rules start to break. What about red apples? Or other fruits?
    - Colours, edges, shapes, guessing textures, …
  - New problem may imply a new set of rules!

Machine learning

http://www.gestaltu.com/2015/10/apples-and-oranges-a-random-portfolio-case-study.html/

Machine learning

- We need an algorithm that can figure out the rules for us!
- For this, we train a classifier (a function that takes some data and assign a label to it as output)
- One of the biggest distinctions in machine learning: supervised learning versus unsupervised learning

Machine learning

Supervised learning

- Supervised learning approaches use examples of the problem you want to solve
- These methods create a classifier by finding patterns in these examples

Machine learning

Supervised learning

https://previews.123rf.com/images/lestertair/lestertair1506/lestertair150600264/41403024-background-fresh-fruit-apples-oranges-lemons-Stock-Photo.jpg

Machine learning
Supervised learning

- Supervised learning steps for prediction based on features
    - collect training data (provide examples of the problem we want to solve)
    - take some measurements. In the context of machine learning these measurements are called features (e.g., go to an orchard and measure the weight, shape, and texture of fruits, and then label them)
    - put all training data into a table, the more training data the better the classifier (model)
    - train the classifier with this data. There are many different types, but the inputs and outputs are always the same

Machine learning

Supervised learning

- A classifier can be seen as a box of rules, and so the training algorithm is the procedure to learn those rules
- It does so by finding patterns in the training data
- For example, it may notice that oranges tend to weigh more, and so it creates a rule saying the more heavy the fruit is the more likely it is an orange

Machine learning

Supervised learning

- This procedure is the same for a new problem. By changing the training data we can create a new classifier. There is no need of writing new rules
- Questions
  - How much training data is needed?
  - How is the optimization achieved?
  - What makes a good feature?
- "all models are wrong, but some are useful". (George E. P. Box)
- The goal of ML is never to make "perfect" guesses, because ML deals in domains where there is no such thing
- The goal of ML is to make guesses that are good enough to be useful

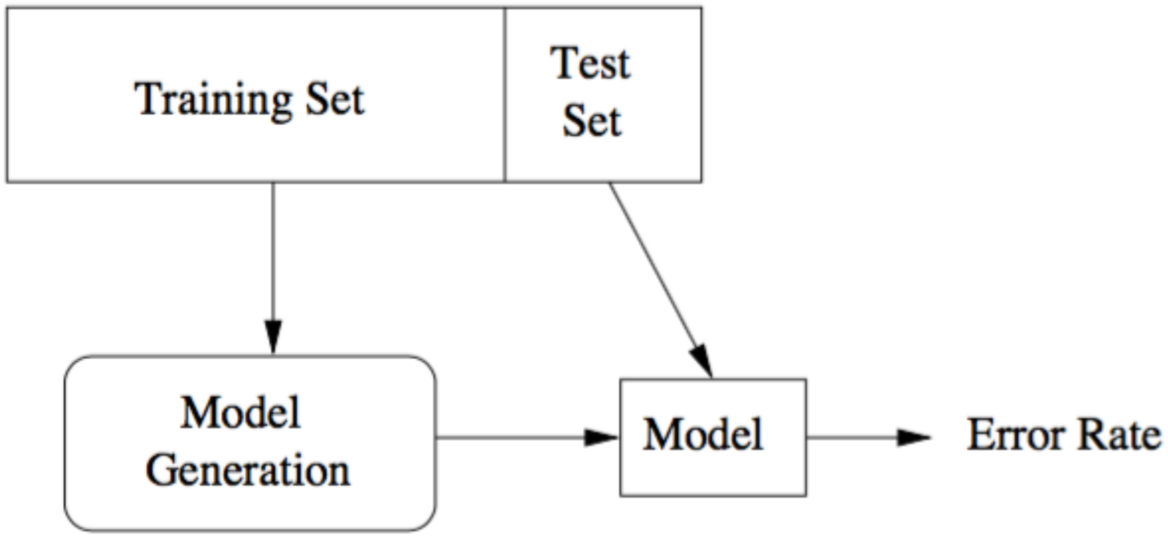
Machine learning

Unsupervised learning

- In unsupervised learning there is no such thing such a class label
- We provide a set of training examples that we believe contains internal patterns
- We leave it to the system to discover those patterns on its own

Machine learning

Train and test architecture



Leskovec, Jurij, Anand Rajaraman, and Jeffrey D. Ullman. 2014. *Mining of Massive Datasets*.
2nd ed. Cambrige, United Kingdom: Cambridge University Press.

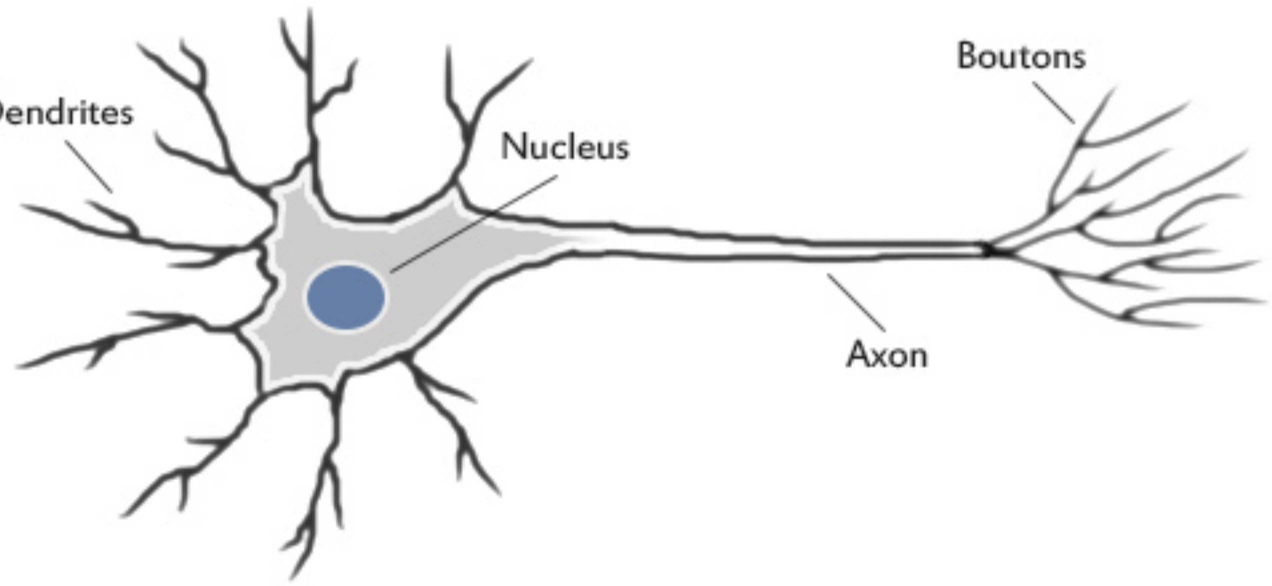
Wrongness

The wrongness measure is known as the cost function (a.k.a., loss function or error)

Machine learning

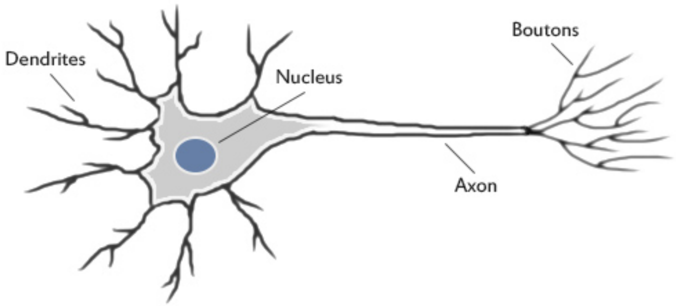
https://previews.123rf.com/images/lestertair/lestertair1506/lestertair150600264/41403024-background-fresh-fruit-apples-oranges-lemons-Stock-Photo.jpg

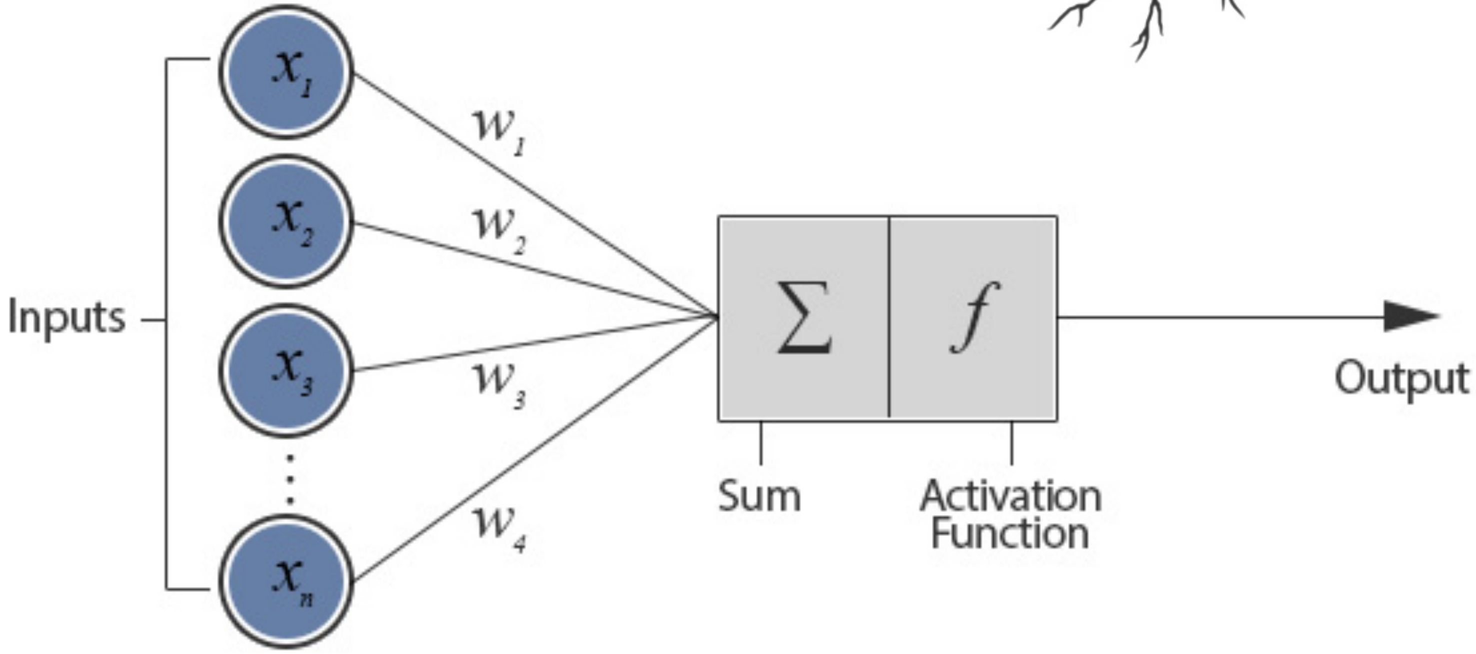
Neuron



http://www.theprojectspot.com/tutorial-post/introduction-to-artificial-neural-networks-part-1/7

Artificial Neuron





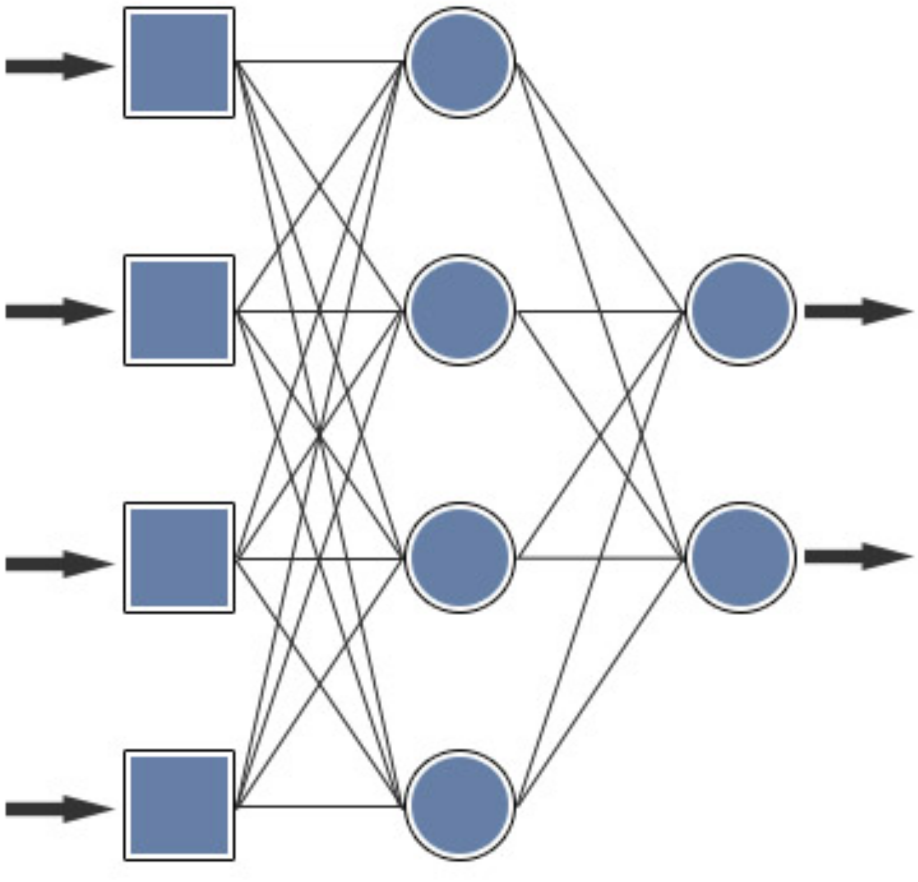
weight vector

$$y = \sum_i w_i x_i = \mathbf{w}^{\mathbf{T}}\mathbf{x}$$

input vector

http://www.theprojectspot.com/tutorial-post/introduction-to-artificial-neural-networks-part-1/7

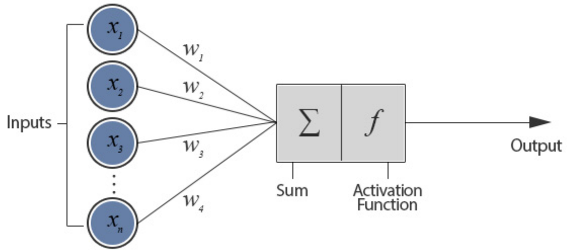
Neural Networks



Input layer

Hidden layer

Output layer

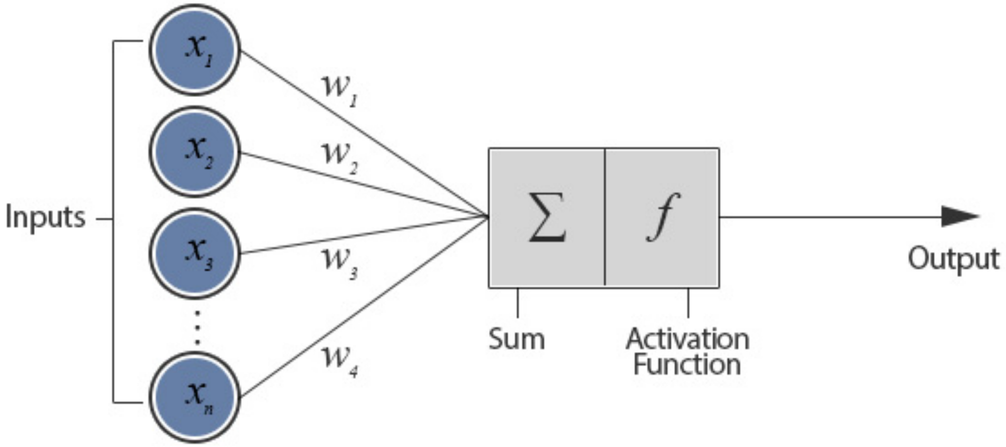


http://www.theprojectspot.com/tutorial-post/introduction-to-artificial-neural-networks-part-1/7

Iterative optimization

- How to learn the weights of a neuron?
- By using a learning algorithm
    - learning turns to be a numerical optimization of weight parameters
    - The actual output values get closer to the target values in each iteration
    - Many quite different sets of weights may work well

Learning as iterative optimization



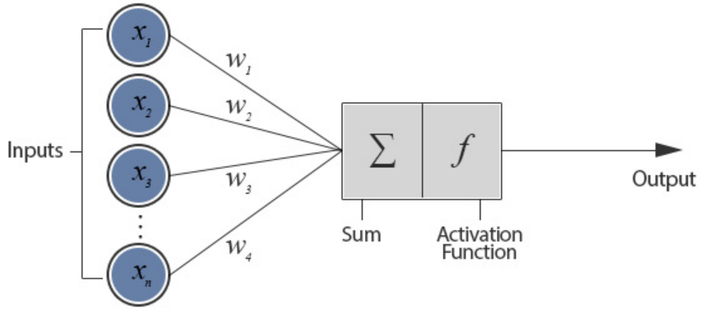
$$y = \sum_i w_i x_i = \overset{\text{weight vector}}{\mathbf{w}}^{\mathbf{T}} \underset{\text{input vector}}{\mathbf{x}}$$

- Simplest example: linear neuron with error measure
  - A linear neuron has a real-valued output which is a weighted sum of its inputs
- The aim of the learning is to minimize the error summed over all training cases

Learning as iterative optimization

- Toy example to illustrate the iterative method:
  - Each day you get lunch at the cafeteria
    - Diet consists of protein, salad, and carbohydrate
    - You get several portions of each
  - The cashier only tells you the total price of the meal
  - After a few days, you ought to be able to figure out what the price is for each portion of each kind of food from all previous examples

Learning as iterative optimization



$$price = x_{protein} w_{protein} + x_{carbo} w_{carbo} + x_{salad} w_{salad}$$

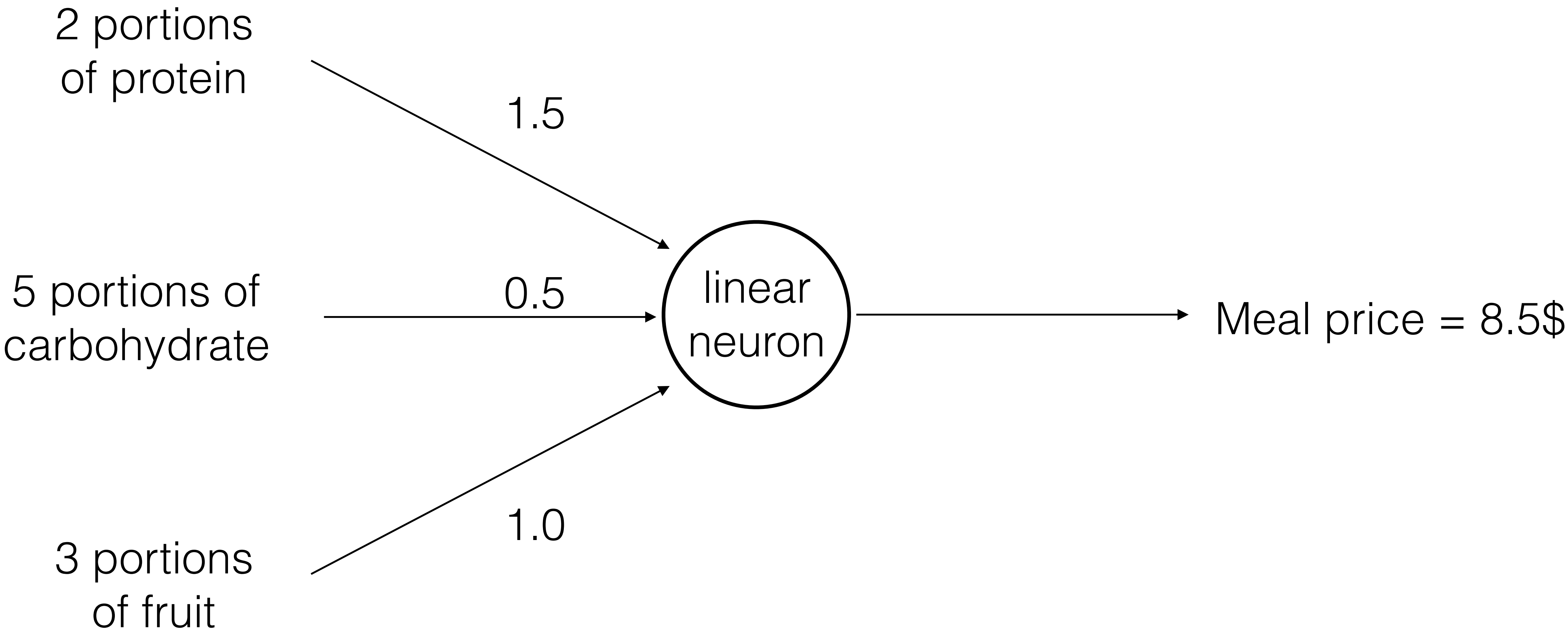
- The prices of the portions are like the weights of a linear neuron

$$\mathbf{w} = (w_{protein}, w_{carbo}, w_{salad})$$

- Let's suppose that the true weights that the cashier using to figure out the price, are 1.5$ for a portion of protein, 0.5$ for portion of carbohydrate, and a 1.0$ for a portion of salad.
- We start with guesses for these prices and then we adjust the guesses slightly, so that we agree better with what the cashier says

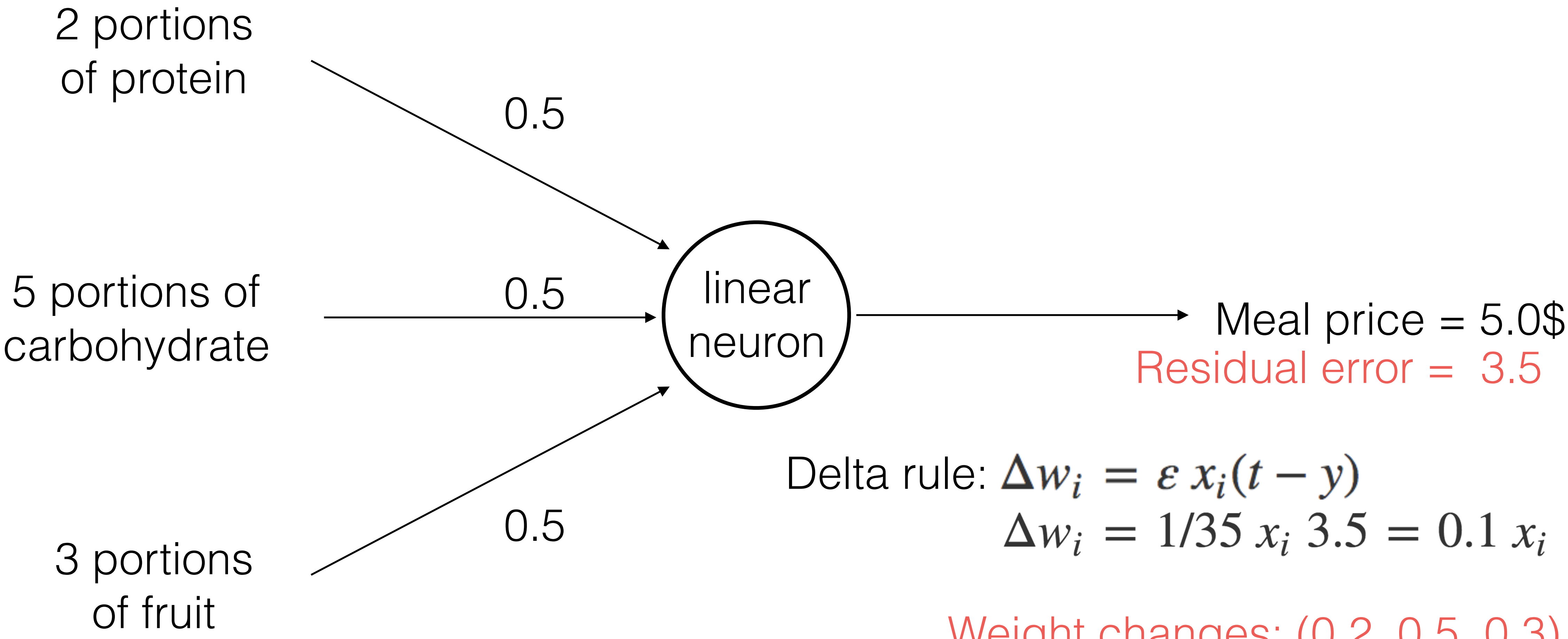
Learning as iterative optimization

True weights used by the cashier



Learning as iterative optimization

Starting with guesses



Delta rule: $\Delta w_i = \varepsilon\, x_i (t - y)$

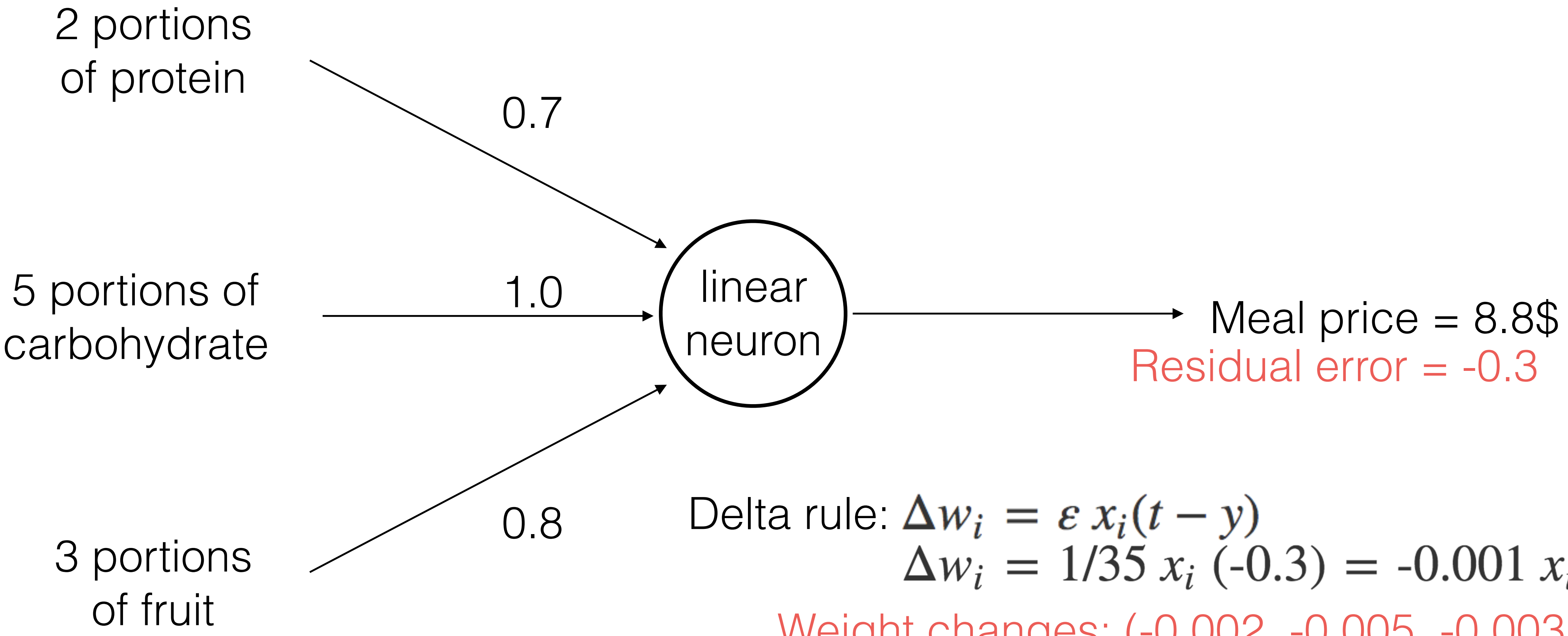
$$\Delta w_i = 1/35\ x_i\ 3.5 = 0.1\ x_i$$

Weight changes: (0.2, 0.5, 0.3)

New weights: (0.7, 1.0, 0.8)

Learning as iterative optimization

First iteration



Delta rule: $\Delta w_i = \varepsilon \, x_i (t - y)$

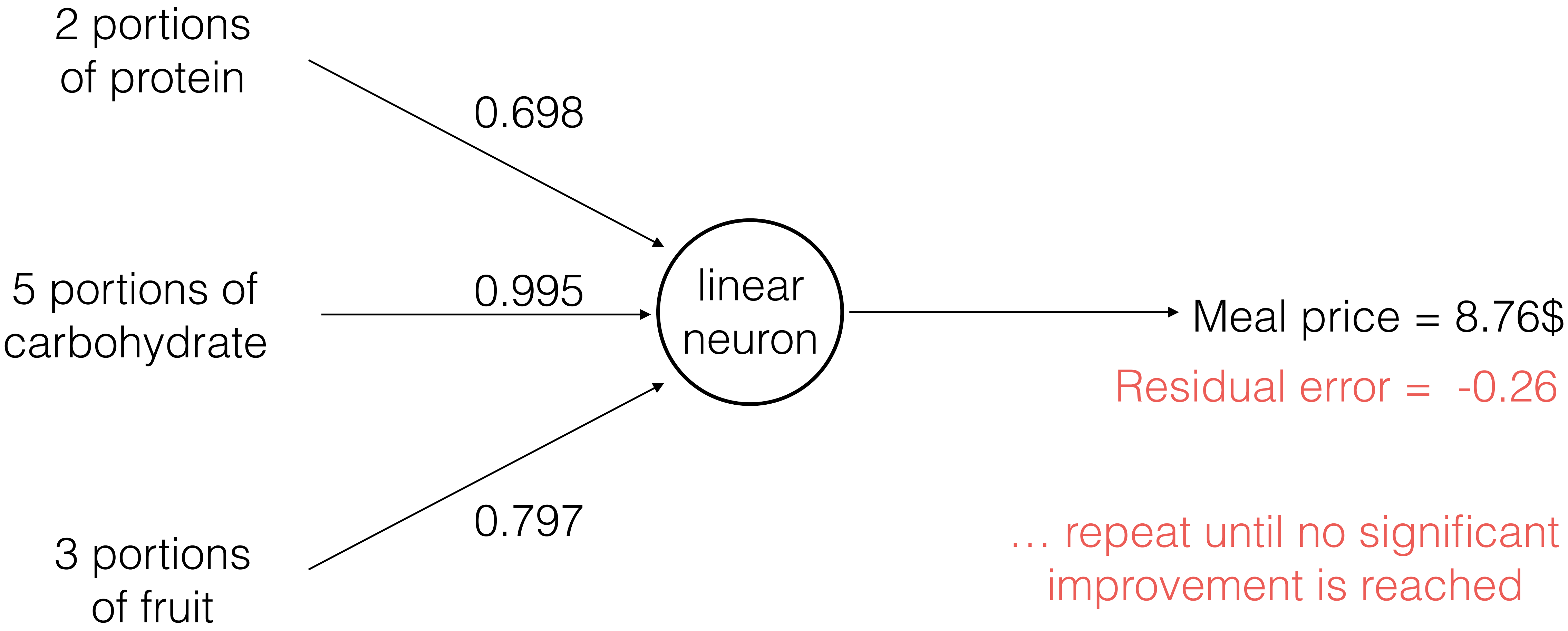
$$\Delta w_i = 1/35 \; x_i \; (-0.3) = -0.001 \; x_i$$

Weight changes: (-0.002, -0.005, -0.003)

New weights: (0.698, 0.995, 0.797)

Learning as iterative optimization

Second iteration

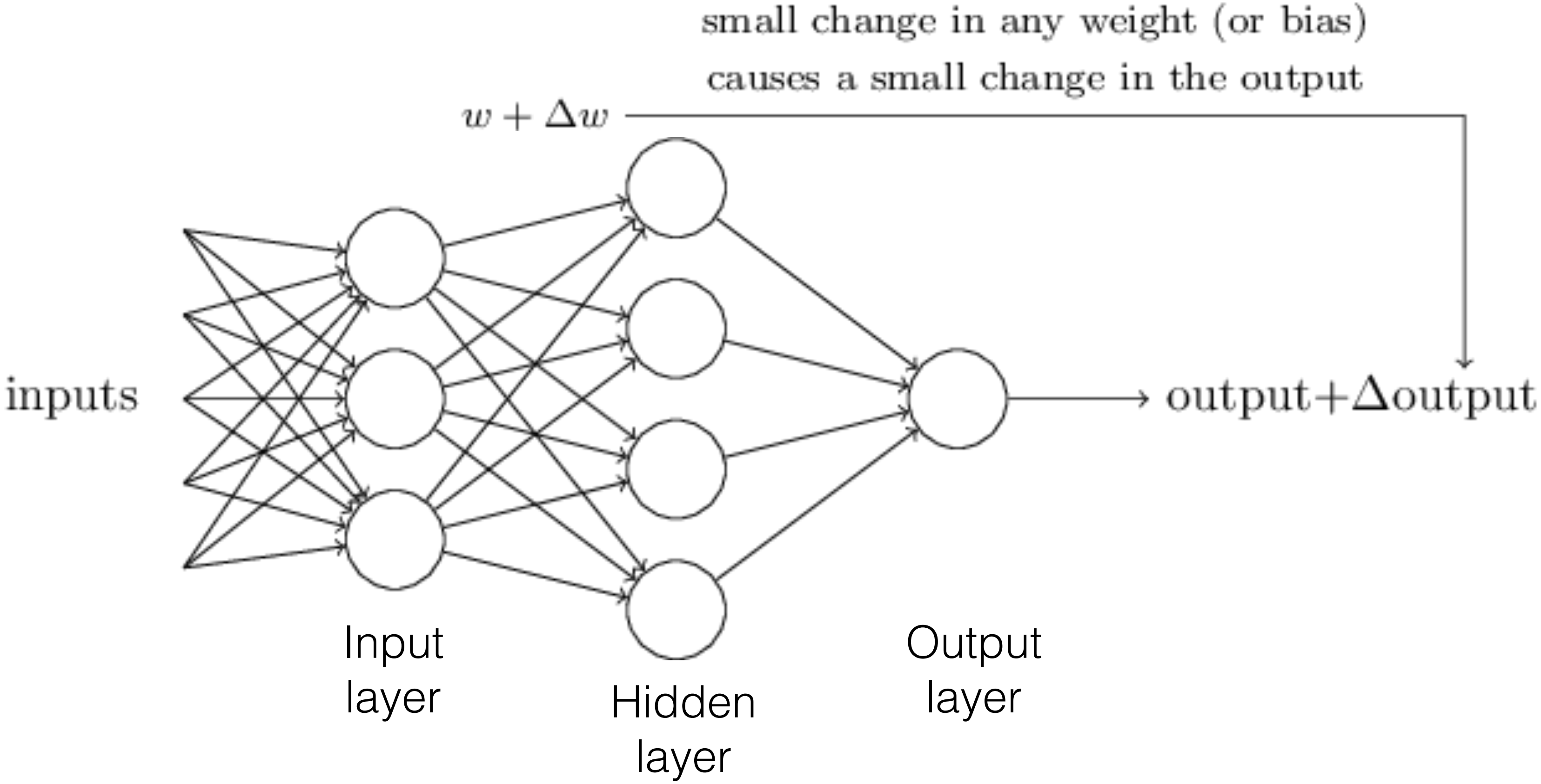


Learning as iterative optimization

- But the delta rule is applied to all training cases, not just one, and so the weights are learnt from all training cases

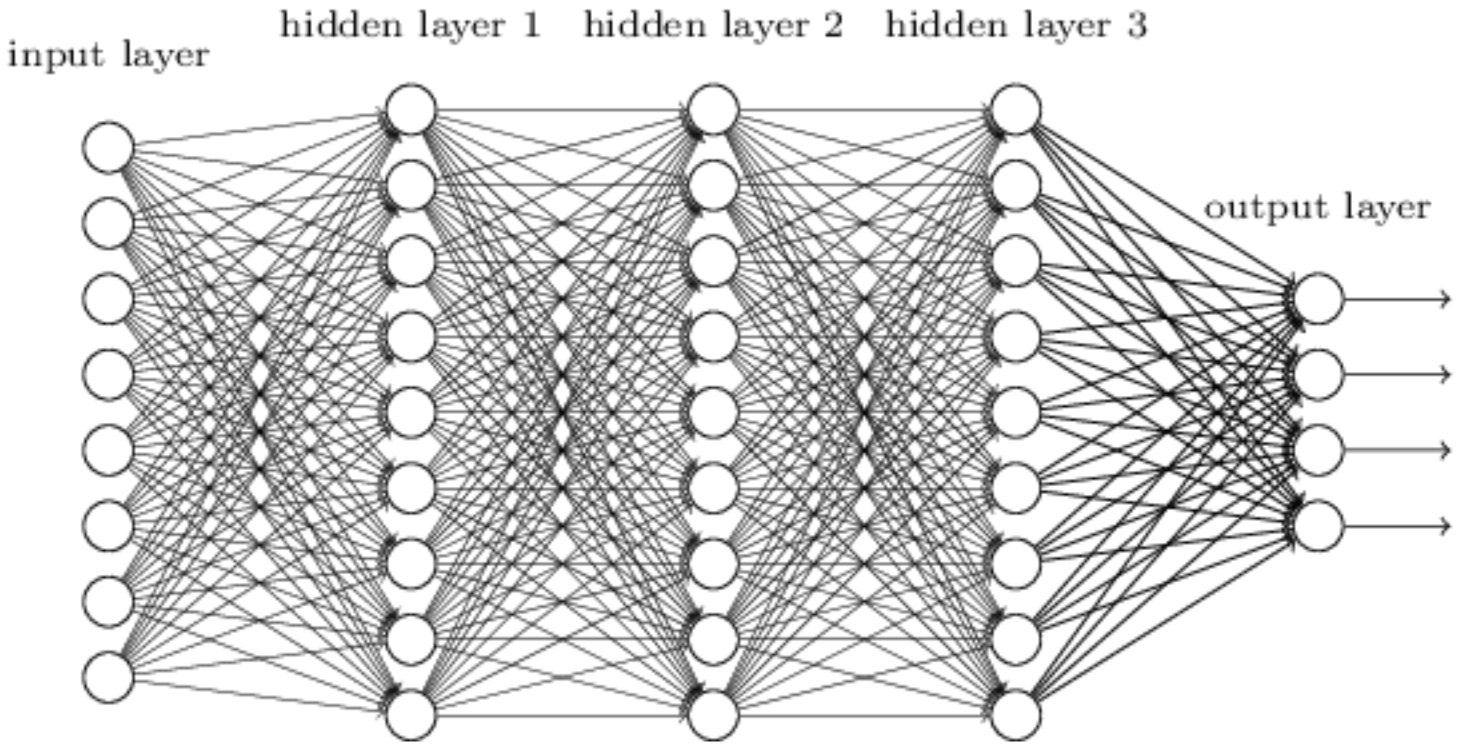
$$\Delta w_i = \sum_n \varepsilon \, x_i^n \, (t^n - y^n)$$

Neural Networks



http://neuralnetworksanddeeplearning.com/chap1.html

Deep Neural Networks



http://stats.stackexchange.com/questions/234891/difference-between-convolution-neural-network-and-deep-learning/235265

Deep Neural Networks
Tensorflow online example

- Tensorflow playground

ML benchmark datasets

- UCI ML Repository
- Deep Learning datasets
  - MNIST online (benchmark)
  - CIFAR10 online (benchmark)

Derivative work

- Google’s Deep dream generator
- Neural network "art"
- SONY CSL Research Flow Machines
- Daddy's car
- Mr. Shadow

Thanks!

Rebecca Fiebrink's "Machine Learning for Artists and Musicians" online course on Kadenze:

https://www.kadenze.com/courses/machine-learning-for-musicians-and-artists/info