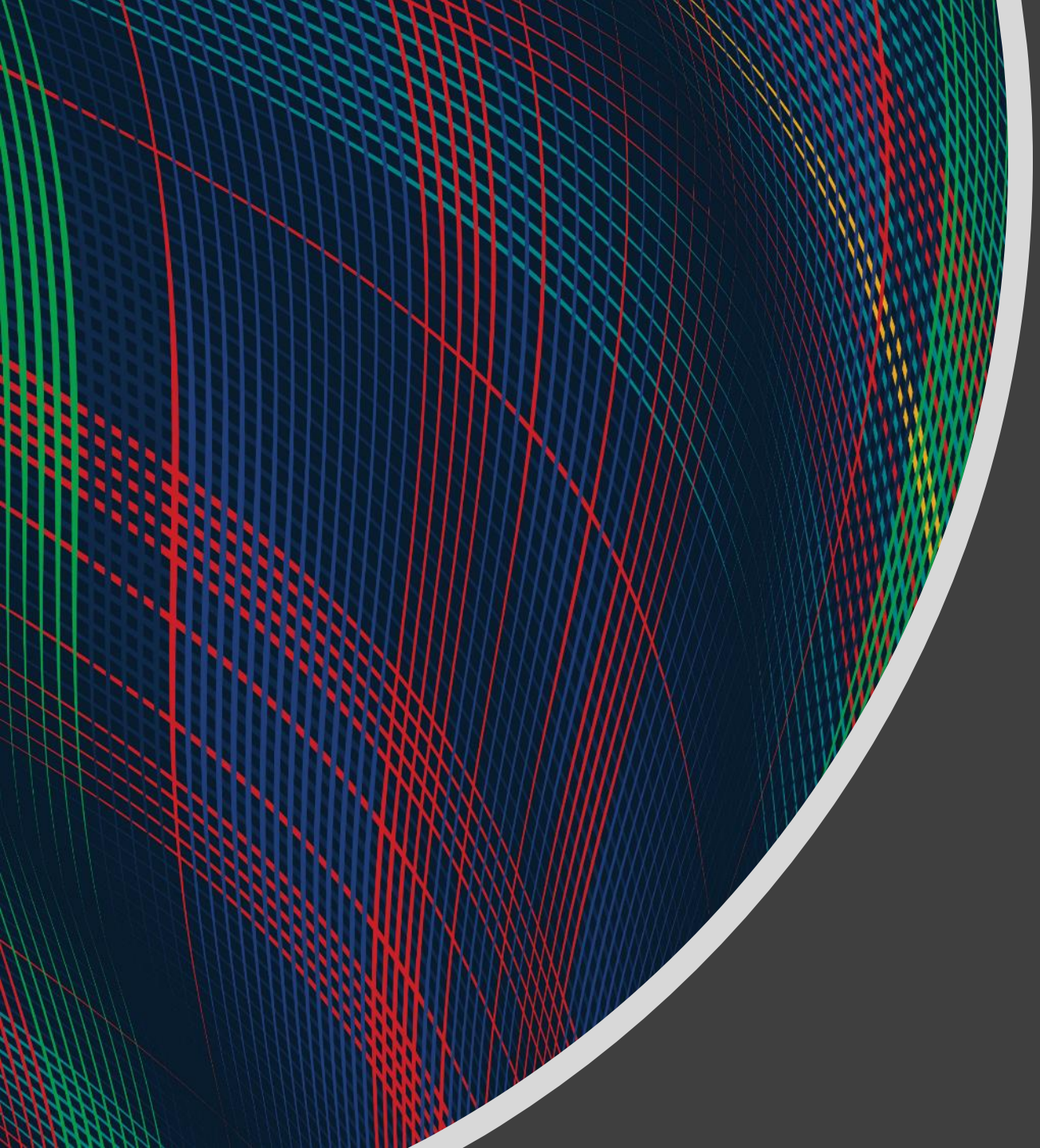


Demystifying AI

Neural Networks

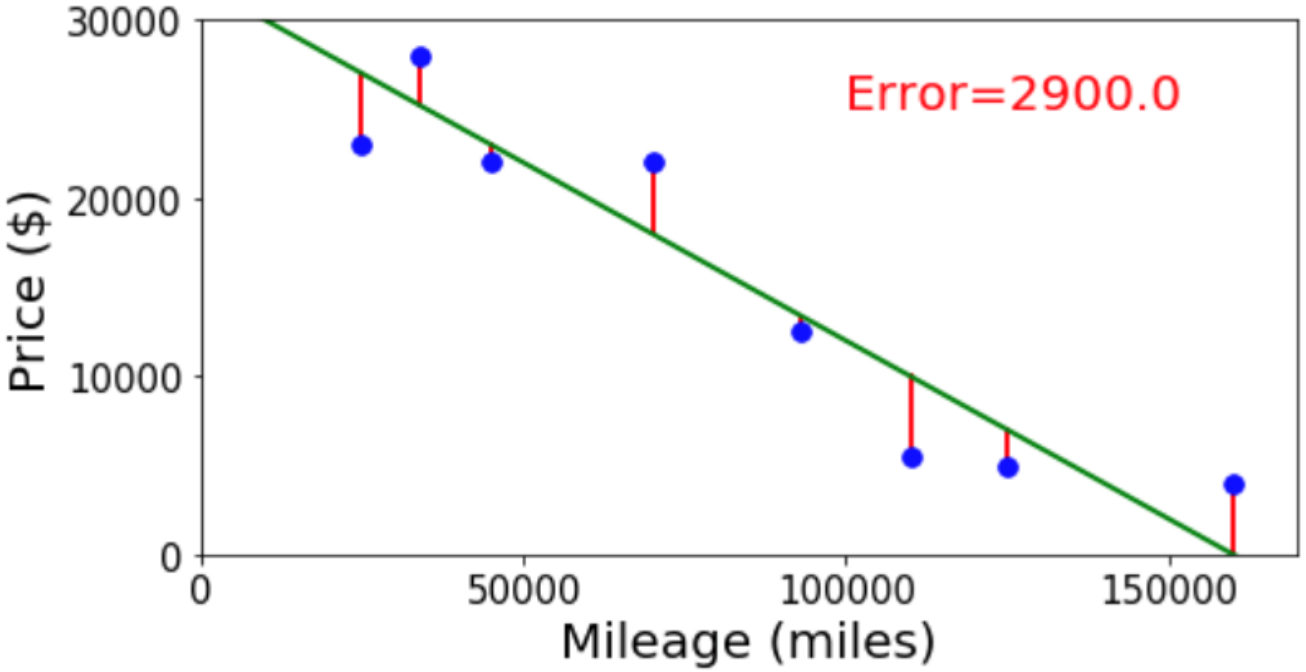
Instructor: Pat Virtue

Warm-up Exercise: Plotting Functions

Paper handout

Linear Regression

Selling my car example

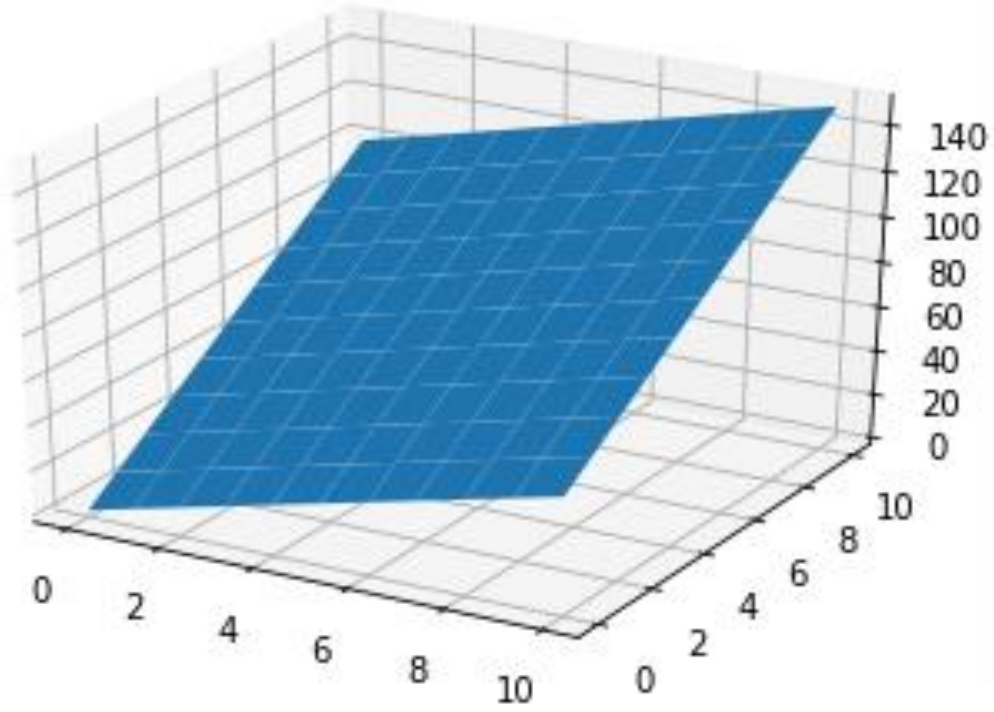


Regression for Non-linear data

Paper handout

Linear Regression

With N-D inputs



Linear Regression

With N-D inputs

1-D linear function

$$y = w_1 x_1 + b$$

2-D linear function

$$y = w_1 x_1 + w_2 x_2 + b$$

3-D linear function

$$y = w_1 x_1 + w_2 x_2 + w_2 x_2 + b$$

N-D linear function

$$y = \sum_{i=1}^{N} w_i x_i + b$$

Network Diagrams

Linear functions

1-D linear function

$$y = w_1 x_1 + b$$

2-D linear function

$$y = w_1 x_1 + w_2 x_2 + b$$

3-D linear function

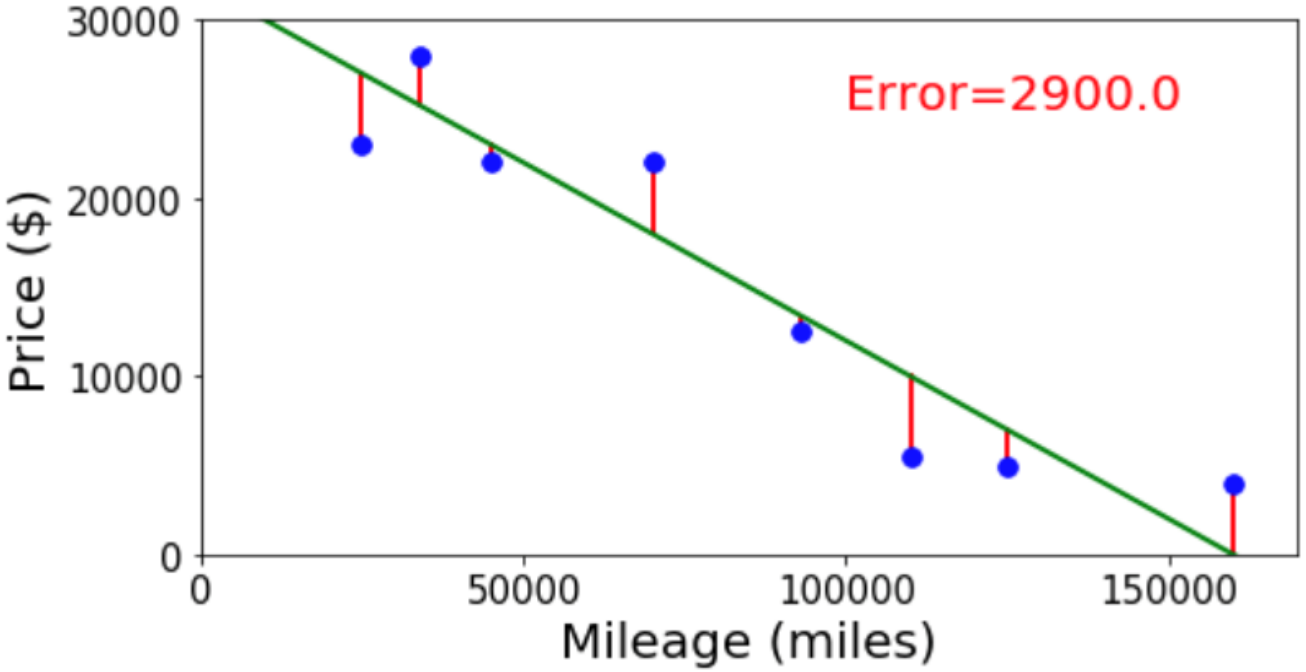
$$y = w_1 x_1 + w_2 x_2 + w_2 x_2 + b$$

N-D linear function

$$y = \sum_{i=1}^{N} w_i x_i + b$$

Linear Regression

Selling my car example



Linear plus ReLU

Rectified linear unit

Plotting Functions

Connecting functions

Plot the function $f_3(x)$ vs x

$f_1(x) = \max(0, 1x + 2)$

$f_2(x) = \max(0, 1x - 2)$

$f_3(x) = f_1(x) - f_2(x)$

Neural Networks

https://www.cs.cmu.edu/~15181/tfp

Setup:

- Switch to the regression dataset that looks like the letter M
- Set the learning rate to 0.003

Steps:

- Set up your architecture: add as many hidden layers and neurons per layer as you like
- Click the play button
- Observe the resulting fit and loss (mean squared error)
- Repeat to try to use as few neurons a possible and still get a good fit

Three-neuron network

Connecting functions

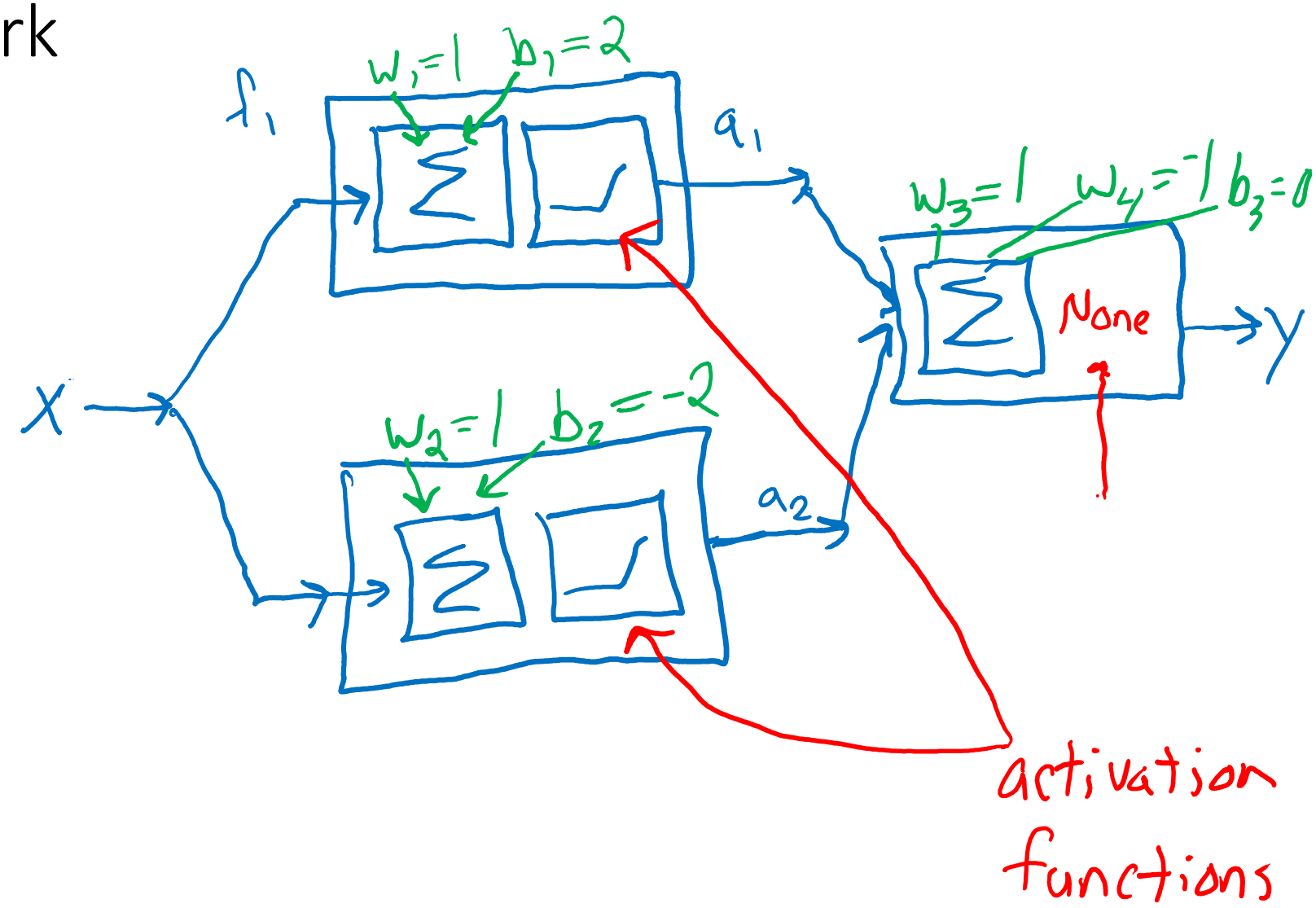
Plot the function $f_3(x)$ vs x

$a_1 = f_1(x) = \max(0, 1x + 2)$

$a_2 = f_2(x) = \max(0, 1x - 2)$

$y = f_3(x) = f_1(x) - f_2(x)$

$1 \cdot a_1 + (-1) a_2 + 0$



Three-neuron network

Connecting functions

Plot the function $f_3(x)$ vs x

$f_1(x) = \max(0, 1x + 2)$

$f_2(x) = \max(0, 1x - 2)$

$f_3(x) = f_1(x) - f_2(x)$