localhost

linear in parameters:

$\sum w_i \cdot \boxed{}$

can't involve any ws! function of non-linear $\vec{x}$ only

Question 🤔 (Answer at practicaldsc.org/q)

Which hypothesis function is **not** linear in the parameters?

- A. $H(\vec{x}) = w_1 \boxed{\left(x^{(1)}x^{(2)}\right)} + \frac{w_2}{x^{(1)}} \sin\left(x^{(2)}\right)$
- B. $H(\vec{x}) = 2^{w_1} x^{(1)}$ w_1 is being exponentiated!
- C. $H(\vec{x}) = \vec{w} \cdot \text{Aug}(\vec{x})$
- D. $H(\vec{x}) = w_1 \boxed{\cos(x^{(1)})} + w_2 \boxed{2^{x^{(2)}\log x^{(3)}}}$
- E. More than one of the above.

$\vec{w} \cdot \text{Aug}(\vec{x}) = w_0 + w_1 x^{(1)} + w_2 x^{(2)} + \dots$

as linear as it gets!

$$\text{Aug}(\vec{x}) = \begin{bmatrix} 1 \\ x^{(1)} \\ x^{(2)} \\ \vdots \\ x^{(d)} \end{bmatrix}$$

?

localhost

$$H(x) = w_0 e^{w_1 x}$$ not currently linear!

Idea: take log of both sides

$$\log H(x) = \log\left(w_0 e^{w_1 x}\right)$$

$$\underbrace{\log H(x)}_{T(x)} = \underbrace{\log(w_0)}_{b_0} + \underbrace{w_1}_{b_1} x$$

use the relationships:

$w_1^* = b_1^*$

$w_0^* = e^{b_0^*}$

find the b^*'s from normal eq'ns

$\Rightarrow T(x) = b_0 + b_1 x \Longrightarrow$ is linear in its parameters!!!

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_3 \end{bmatrix}_{n\times 2} \qquad \vec{b} = \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} \qquad \vec{t} = \begin{bmatrix} \log y_1 \\ \log y_2 \\ \vdots \\ \log y_n \end{bmatrix}$$

solve normal equations for b_0^*, b_1^*

17.1

- Suppose we have the following fitted model:

For illustration, assume `'weekend'` was originally a categorical feature with two possible values, `'Yes'` or `'No'`.

$$H(x) = 1 + 2 \cdot (\text{weekend==Yes}) - 2 \cdot (\text{weekend==No})$$

- This is equivalent to:

$$H(x) = 10 - 7 \cdot (\text{weekend==Yes}) - 11 \cdot (\text{weekend==No})$$

- Note that for a particular row in the dataset, weekend==Yes + weekend==No is always equal to 1.

$$X = \begin{array}{c} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ \vdots & \vdots & \vdots \\ 1 & 1 & 0 \end{bmatrix} \\ \begin{array}{ccc} & \text{Yes} & \text{No} \\ 1 & 2 & 3 \end{array} \end{array}$$

col 2 + col 3 = col 1

$\Rightarrow$ X's columns are not linearly independent!

$\Rightarrow$ X is not full rank

$\Rightarrow$ X^TX is not invertible!

$\Rightarrow$ infinitely many solutions to $\boxed{X^TX\vec{w} = X^T\vec{y}}$

localhost

```
In [65]: stdscaler.var_

Out[65]: array([    3.89, 35191.58,    13.72,    25.44,    23.08])
```

- If needed, the `fit_transform` method will fit the transformer and then transform the data in one go.

Why are these values different?

```
In [66]: new_scaler = StandardScaler()

In [68]: stdscaler.transform(sales.iloc[:, 1:].tail(5))

Out[68]: array([[-1.13, -1.31, -1.35, -1.6 ,  0.89],
                [ 0.14,  0.39,  0.4 ,  0.32, -0.36],
                [-0.09, -0.03,  0.46,  0.36, -0.57],
                [ 0.9 ,  1.08,  1.05,  1.19, -1.61],
                [ 2.67,  0.69, -0.3 ,  0.46,  0.05]])
```

fit on all rows

```
In [69]: new_scaler.fit_transform(sales.iloc[:, 1:].tail(5))

Out[69]: array([[-1.33, -1.79, -1.71, -1.88,  1.48],
                [-0.32,  0.28,  0.43,  0.19, -0.05],
                [-0.36, -0.24,  0.49,  0.23, -0.31],
                [ 0.29,  1.11,  1.22,  1.13, -1.58],
                [ 1.71,  0.64, -0.43,  0.34,  0.46]])
```

fit on just the last 5 rows

42.1