Supervised Learning requires 3 ingredients:

1. Training data -
$$\mathcal{D} := \Big\{\langle \vec{x}_1, y_1\rangle, \langle \vec{x}_2, y_2\rangle, \ldots, \langle \vec{x}_n, y_n\rangle\Big\}$$
There are historical input-output examples. $\vec{x}_1$ may be Bill's characteristics where $y_1 = 1$ (he paid his loan); $\vec{x}_2$ may be Jill's characteristics where $y_1 = 1$, etc. There are n examples. Note: $\mathcal{D}$ is denoted using vector and matrix notation.
$$\mathcal{D} = \langle X, \vec{Y}\rangle$$
where $X \in \mathcal{X}^n$ if $\dim n \times p$ and $\vec{y} \in \mathcal{Y}^n = \Big\{0, 1\Big\}^n$.

2. $\mathcal{H} = \Big\{$all candidate functions h that can approximate $f\Big\}$ This is needed because f may be a very complicated function that can never be learned. So pick a large set of candidate functions that can approximate it.

3. $\mathcal{A}$- an algorithm that takes in $\mathcal{D}, \mathcal{H}$ and selects one best candidate function g
$$g = \mathcal{A}(\mathcal{D}, \mathcal{H})$$

Review:
$$y = t(z_1, \ldots, z_t)$$
where y is the phenomenon you wish to explain so you can predict in the future, t is the true relationship that produces the phenomenon and $z_1, \ldots, z_t$ is the causal attributes about the object that will produce the phenomenon.
$z_1, \ldots, z_t$ are unobservable but $x_1, \ldots, x_p$ are observable.
$$t = f(x_1, \ldots, x_p) + \delta$$
where
$$\delta = t(\vec{z}) - f(\vec{x})$$
f is the best possible relationship given attributes you can measure and δ is the difference between the true relationship and the "best we can do" (error due to ignorance).

Goal: Estimate f.
You happen to have historical data $\mathcal{D}$ consisting of a prior examples. You have a finite space $\mathcal{H}$ of functions to approximate f and an algorithm $\mathcal{A}$. You now use $\mathcal{A}$ to produce g.
If $f \in \mathcal{H}$,
$$y = g(x_1, x_2, \ldots, x_p) + \underbrace{(t(\vec{z}) - f(\vec{x}))}_{\text{error due to ignorance}} + \underbrace{(f(\vec{x}) - g(\vec{x}))}_{\text{parameter estimator error}}$$
The usual case: $f \notin \mathcal{H}$; this means there is a "closest" function $h* \in \mathcal{H}$ but due to random chance, we pick g instead.
$$y = g(\vec{x}) + \underbrace{(t(\vec{z}) - f(\vec{x}))}_{\text{error due to ignorance}} + \underbrace{(f(\vec{x} - h^*(\vec{x})}_{\substack{\text{model misprediction} \\ f \notin \mathcal{H}}} + \underbrace{(h^*(\vec{x}) - g(\vec{x}))}_{\substack{\text{parameter estimator error} \\ g \neq h^*}}$$
There are 3 sources of error.