Classification, Halfspaces, the Perceptron algorithm

Chris Cornwell

Feb 25, 2025

Outline

Outline

Example of a classification task

Use a model to predict if an image of a handwritten digit is 0, 1, 2, 3, 4, 5, 6, 7, 8, or 9.

If $\mathbf{x}$ is the image (converted to a vector in some way), then your model's output $\hat{y}(\mathbf{x})$, is the predicted digit. In the data you work with, you have an "observation" y for which digit was, in fact, being written.

Example of a classification task

Use a model to predict if an image of a handwritten digit is 0, 1, 2, 3, 4, 5, 6, 7, 8, or 9.

If $\mathbf{x}$ is the image (converted to a vector in some way), then your model's output $\hat{y}(\mathbf{x})$, is the predicted digit. In the data you work with, you have an "observation" y for which digit was, in fact, being written.

While y and $\hat{y}$ are numbers, they are more like labels than something on the number line. Getting $\hat{y} = 4$, when $y = 5$, is not any better than getting $\hat{y} = 0$.

Close only counts in ...Regression

When performing linear regression, on independent variables $x_0, x_1, \ldots, x_{d-1}$, had (affine) linear function $\hat{y} = p_0 x_0 + p_1 x_1 + \ldots + p_{d-1} x_{d-1} + p_d$;
values of function $\leftrightarrow$ prediction $\hat{y}$; error term ε, so that $y = \hat{y} + \varepsilon$.

[1] Should not think of this as (deterministic) function; rather, $Y_{\mathbf{x}}$ is random variable, e.g., simple linear regression: $Y_{x_0} = p_0 x_0 + p_1 + \varepsilon$.

Close only counts in …Regression

When performing linear regression, on independent variables $x_0, x_1, \ldots, x_{d-1}$, had (affine) linear function $\hat{y} = p_0 x_0 + p_1 x_1 + \ldots + p_{d-1} x_{d-1} + p_d$;
values of function $\leftrightarrow$ prediction $\hat{y}$; error term ε, so that $y = \hat{y} + \varepsilon$.
In other words, observation $Y_{\mathbf{x}}$ for each data point $\mathbf{x} = (x_0, x_1, \ldots, x_{d-1}) \in \mathbb{R}^d$. Have a linear "model" $\mathbf{x} \mapsto \hat{y}$ that approximates $\mathbf{x} \mapsto Y_{\mathbf{x}}$.[1]

[1] Should not think of this as (deterministic) function; rather, $Y_{\mathbf{x}}$ is random variable, e.g., simple linear regression: $Y_{x_0} = p_0 x_0 + p_1 + \varepsilon$.

Close only counts in ...Regression

When performing linear regression, on independent variables $x_0, x_1, \ldots, x_{d-1}$, had (affine) linear function $\hat{y} = p_0 x_0 + p_1 x_1 + \ldots + p_{d-1} x_{d-1} + p_d$;
values of function $\leftrightarrow$ prediction $\hat{y}$; error term ε, so that $y = \hat{y} + \varepsilon$.
In other words, observation $Y_{\mathbf{x}}$ for each data point $\mathbf{x} = (x_0, x_1, \ldots, x_{d-1}) \in \mathbb{R}^d$. Have a linear "model" $\mathbf{x} \mapsto \hat{y}$ that approximates $\mathbf{x} \mapsto Y_{\mathbf{x}}$.[1]
Would expect $|Y_{\mathbf{x}} - \hat{y}|$ to almost never be exactly 0; good model: one where $|Y_{\mathbf{x}} - \hat{y}|$ is small (but positive), on average.

"Regression"

[1] Should not think of this as (deterministic) function; rather, $Y_{\mathbf{x}}$ is random variable, e.g., simple linear regression: $Y_{x_0} = p_0 x_0 + p_1 + \varepsilon$.

Close only counts in ...Regression

When performing linear regression, on independent variables $x_0, x_1, \ldots, x_{d-1}$, had (affine) linear function $\hat{y} = p_0 x_0 + p_1 x_1 + \ldots + p_{d-1} x_{d-1} + p_d$;
values of function $\leftrightarrow$ prediction $\hat{y}$; error term ε, so that $y = \hat{y} + \varepsilon$.
In other words, observation $Y_{\mathbf{x}}$ for each data point $\mathbf{x} = (x_0, x_1, \ldots, x_{d-1}) \in \mathbb{R}^d$. Have a linear "model" $\mathbf{x} \mapsto \hat{y}$ that approximates $\mathbf{x} \mapsto Y_{\mathbf{x}}$.[1]
Would expect $|Y_{\mathbf{x}} - \hat{y}|$ to almost never be exactly 0; good model: one where $|Y_{\mathbf{x}} - \hat{y}|$ is small (but positive), on average.

"*Regression*"

In a "Classification" task, the value $Y_{\mathbf{x}}$ is more like a *label*. It might not even be a number and, if so, a $\hat{y}$ is just wrong or not; close doesn't count.
That is, you want
$\hat{y}$ to be the same as $Y_{\mathbf{x}}$, as much as possible with your model.

[1]Should not think of this as (deterministic) function; rather, $Y_{\mathbf{x}}$ is random variable, e.g., simple linear regression: $Y_{x_0} = p_0 x_0 + p_1 + \varepsilon$.

Half-space model

Assume data is from $\mathbb{R}^d$ for some $d > 0$ and we only have two labels (e.g., this is Spam ($\mathtt{S}$) or it is Not spam ($\mathtt{N}$)).

Half-space model

Assume data is from $\mathbb{R}^d$ for some $d > 0$ and we only have two labels (e.g., this is Spam ($\mathtt{S}$) or it is Not spam ($\mathtt{N}$)).

A *hyperplane* in $\mathbb{R}^d$ is an (affine) linear subspace that separates $\mathbb{R}^d$ in two. Perhaps we get lucky and can find a hyperplane H so that data points with label $\mathtt{S}$ are on one side of H and data with label $\mathtt{N}$ are on the other side.

Half-space model

Assume data is from $\mathbb{R}^d$ for some $d > 0$ and we only have two labels (e.g., this is Spam ($\mathbb{S}$) or it is Not spam ($\mathbb{N}$)).
A *hyperplane* in $\mathbb{R}^d$ is an (affine) linear subspace that separates $\mathbb{R}^d$ in two. Perhaps we get lucky and can find a hyperplane H so that data points with label $\mathbb{S}$ are on one side of H and data with label $\mathbb{N}$ are on the other side. Using coordinates $(x_1, x_2, \ldots, x_d)$ in $\mathbb{R}^d$, a hyperplane H may be determined from $d + 1$ numbers $w_1, w_2, \ldots, w_d$, and b. It consists of solutions to

$$w_1 x_1 + w_2 x_2 \ldots + w_d x_d + b = 0.$$

- Rewriting in vector form: $\mathbf{w} = (w_1, w_2, \ldots, w_d)$, look for solutions $\mathbf{x} \in \mathbb{R}^d$ to the equation $\mathbf{w} \cdot \mathbf{x} + b = 0$.
- $\mathbf{w}$ is a vector that is orthogonal to a $(d - 1)$-dimensional subspace of $\mathbb{R}^d$; $|b|$ corresponds to a translation away from the origin.

Half-space model, continued

Using the notation from last slide:
a half-space model in $\mathbb{R}^d$ is determined by $d + 1$ parameters $w_1, w_2, \ldots, w_d, b$; the first d parameters grouped into a vector: $\mathbf{w} = (w_1, w_2, \ldots, w_d)$.

Half-space model, continued

Using the notation from last slide:
a half-space model in $\mathbb{R}^d$ is determined by $d+1$ parameters $w_1, w_2, \ldots, w_d, b$; the first d parameters grouped into a vector: $\mathbf{w} = (w_1, w_2, \ldots, w_d)$.
Given $\mathbf{x} \in \mathbb{R}^d$, the side of the hyperplane it is on is determined by the sign of $\mathbf{w} \cdot \mathbf{x} + b$.

- (Positive side) Say that $h(\mathbf{x}) = 1$ if $\mathbf{w} \cdot \mathbf{x} + b > 0$.
- (Negative side) Say that $h(\mathbf{x}) = -1$ if $\mathbf{x} \cdot \mathbf{x} + b < 0$.

Half-space model, continued

Using the notation from last slide:
a half-space model in $\mathbb{R}^d$ is determined by $d + 1$ parameters $w_1, w_2, \ldots, w_d, b$; the first d parameters grouped into a vector: $\mathbf{w} = (w_1, w_2, \ldots, w_d)$.
Given $\mathbf{x} \in \mathbb{R}^d$, the side of the hyperplane it is on is determined by the sign of $\mathbf{w} \cdot \mathbf{x} + b$.

- (Positive side) Say that $h(\mathbf{x}) = 1$ if $\mathbf{w} \cdot \mathbf{x} + b > 0$.
- (Negative side) Say that $h(\mathbf{x}) = -1$ if $\mathbf{x} \cdot \mathbf{x} + b < 0$.

If there exists a hyperplane, given by some $\mathbf{w}$, b, so that $\mathbf{x}$ has one of the labels if and only if it is on the positive side, the labeled data are called **linearly separable**.

Perceptron algorithm

Assuming that $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ is linearly separable, the Perceptron algorithm is a procedure that is guaranteed to find a hyperplane that separates the data.

Perceptron algorithm

Assuming that $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ is linearly separable, the Perceptron algorithm is a procedure that is guaranteed to find a hyperplane that separates the data.