Is your machine sick? An interpretable data-driven approach for condition monitoring

Kriterion brownbag sessions No. 3

Ryan Balshaw

Kriterion
Pretoria, South Africa

27 June 2025

Overview

1. Introduction and general overview
2. Part 1: Formulation and overview
3. Part 2: Interpretation
4. Application
5. Conclusion

Introduction

Context of problem: I wanted my third brown-bag to conform to the following criteria:

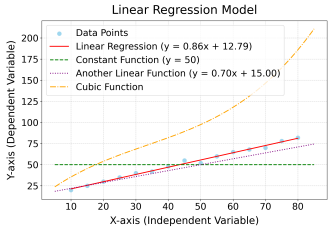
- Introduce a fundamental concept that is ML/AI related.
- Cover a concept that is not too far removed from everyone's understanding and have it be something we all might find interesting.
- Drive intuition and understanding in the concept without over-burdening our minds.

What did I settle on: Logistic regression.

Presentation structure: Fundamentals $\longrightarrow$ Interpretation $\longrightarrow$ Application

Prelude: Linear regression

We are all familiar with the idea of linear regression, which is fitting a linear function through data.



The functional form is simple:

$$y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T\mathbf{x}, \tag{1}$$

where $\mathbf{w} \in \mathbb{R}^D$ and $\mathbf{x} \in \mathbb{R}^D$ are the model parameters and input features, respectively. Importantly, this is a linear function with respect to the *model parameters*, **not** with respect to the data.

The main event: Logistic regression

Logistic regression defines a linear discriminative classification model, i.e., a model of $p(y|\mathbf{x})$ with $y \in [0, 1], \mathbf{x} \in \mathbb{R}^D$. To define this model, a linear mapping of data $\mathbf{x}$ to a single variable z. This linear mapping is given by

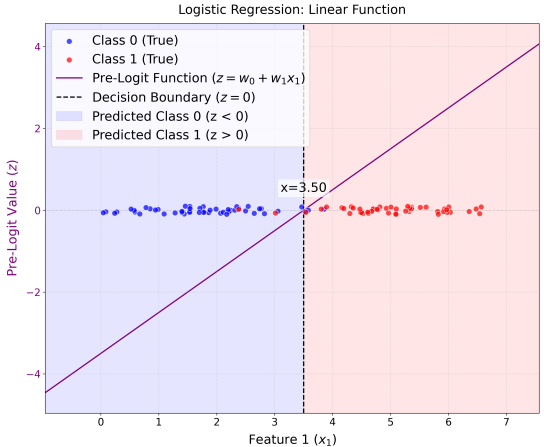
$$z(\mathbf{x}, \boldsymbol{\zeta}) = \mathbf{w}^T\mathbf{x} + b, \tag{2}$$

where $\boldsymbol{\zeta} \in \mathbb{R}^{d+1} = [\mathbf{w}^T, b]^T$ is a vector containing $\mathbf{w} \in \mathbb{R}^d$ (a learnable weight vector) and b, a learnable scalar offset parameter, respectively.

Question: What does this remind us of?

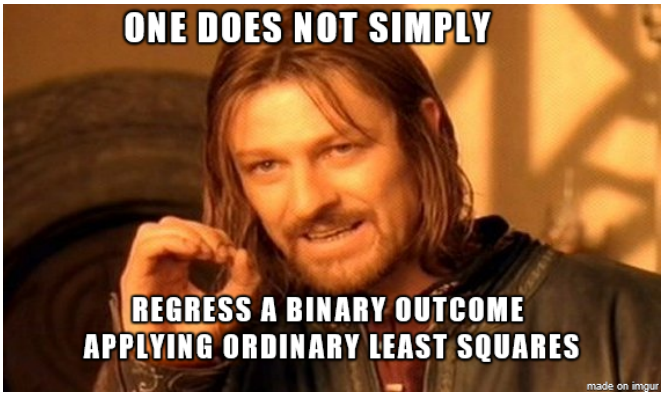
The main event: Logistic regression

So what does this look like?



The main event: Logistic regression

If only it was just a least-squares solution (sad)



The main event: Logistic regression

In a classification setting, a discriminative conditional distribution for a specific label y, i.e., $p(y = 1|\mathbf{x}, \boldsymbol{\zeta})$, is required *(Read: We need probabilities!!)*. This representation is easily given by

$$p(y = 1|\mathbf{x}, \boldsymbol{\zeta}) = \sigma(z_{\boldsymbol{\zeta}}(\mathbf{x})) = p, \tag{3}$$

where $\sigma(z)$ represents the sigmoid function and is given by

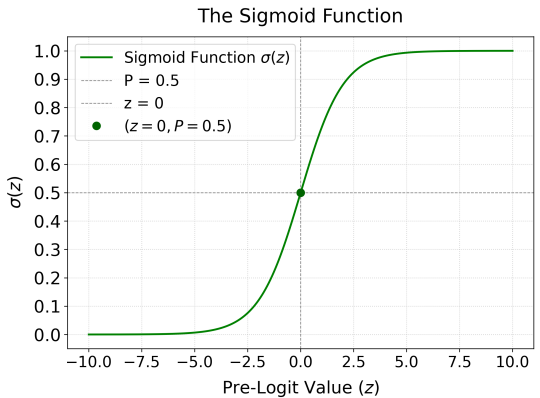
$$\sigma(z) = \frac{1}{1 + e^{-z}}. \tag{4}$$

We can also get the probability of the other label, given that $\sum_j p_j = 1$

$$p(y = 0|\mathbf{x}, \boldsymbol{\zeta}) = 1 - p. \tag{5}$$

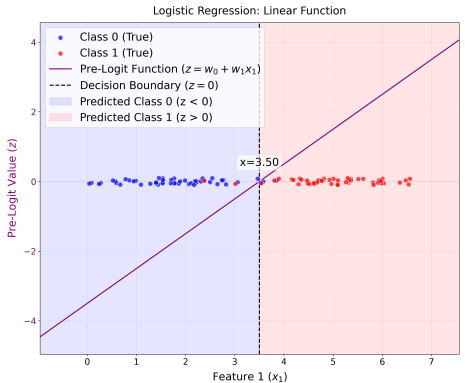
The main event: Logistic regression

The sigmoid function looks as follows:

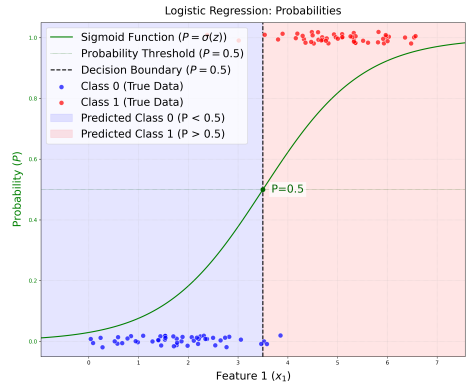


The main event: Logistic regression

So what are we doing? I think this figure summarises it well:



(a) Pre-sigmoid.



(b) Post-sigmoid.

The main event: Logistic regression

A Bernoulli distribution can be used to define an estimator (*read: objective function*) for the parameters $\widehat{\zeta}$ via conditional maximum likelihood estimation (*read: job security*). This is given by

$$p(y|\mathbf{x}, \boldsymbol{\zeta}) = \text{Bernoulli}(y|\sigma(z_{\boldsymbol{\zeta}}(\mathbf{x}))). \tag{6}$$

Using this distribution, the probability mass function for a single observation $\mathbf{x}_i$ becomes

$$p(y|\mathbf{x}_i, \boldsymbol{\zeta}) = \sigma(z_{\boldsymbol{\zeta}}(\mathbf{x}_i))^y \cdot (1 - \sigma(z_{\boldsymbol{\zeta}}(\mathbf{x}_i)))^{1-y}. \tag{7}$$

Given a dataset $\mathcal{D}$ consisting of independent and identically distributed (i.i.d.) samples from some true data distribution $p_{data}\left(\mathbf{x}, y\right)$ the likelihood function (*read: objective function*) is given by

$$l(\boldsymbol{\zeta}, \mathbf{X}, \mathbf{y}) = \prod_{i=1}^{N} p_i^{y_i} \cdot (1 - p_i)^{1-y_i}. \tag{8}$$

where $\mathbf{X} \in \mathbb{R}^{N \times d} = [\mathbf{x}_1, \cdots, \mathbf{x}_N]^T$ and $p_i = p\left(y = 1|\mathbf{x}_i, \boldsymbol{\zeta}\right)$.

The main event: Logistic regression

Finally, the log-likelihood (LL) function $L(\boldsymbol{\zeta}, \mathbf{X}, \mathbf{y})$ then becomes

$$L(\boldsymbol{\zeta}, \mathbf{X}, \mathbf{y}) = \sum_{i=1}^{N} y_i \ln p_i + (1 - y_i) \ln(1 - p_i), \tag{9}$$

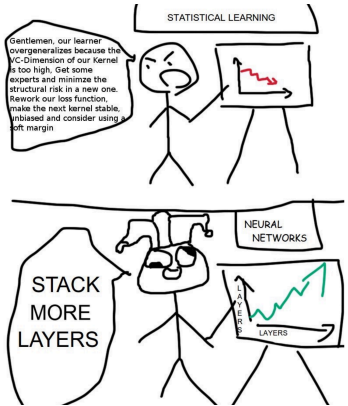
The LL function can be used to define the conditional maximum likelihood estimator (*read: objective function*) for the estimate $\widehat{\boldsymbol{\zeta}}$ in numerical format as

$$\widehat{\boldsymbol{\zeta}} = \max_{\boldsymbol{\zeta}} L(\boldsymbol{\zeta}, \mathbf{X}, \mathbf{y}). \tag{10}$$

What about the negative log likelihood? It is commonly known as the binary cross-entropy loss. If you want more information on how to optimise this function, refer to Sjmelck!

The main event: Logistic regression

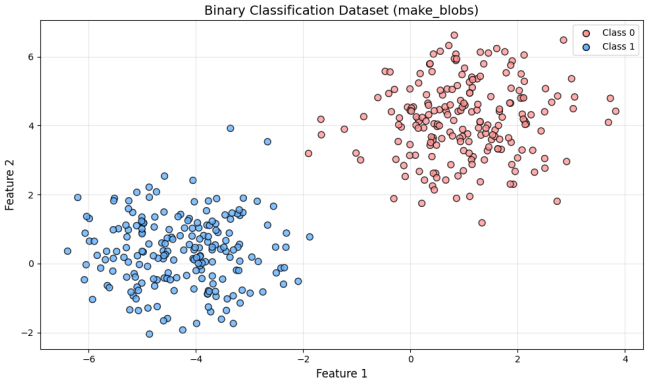
Just a reminder on why this is a fundamental approach:



Logistic regression = one layer neural network with no activation!

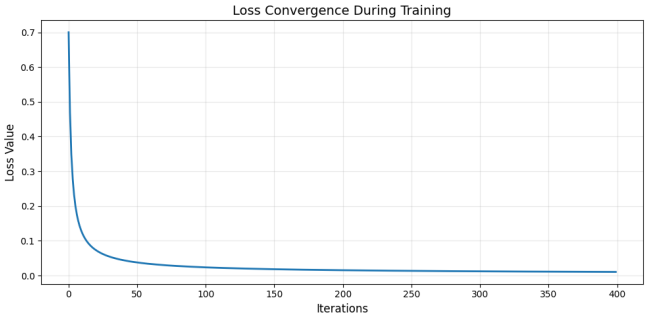
Testing the model

Consider a 2D problem where we have data sampled from two Gaussians:



Testing the model

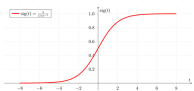
Training a model on this loss produces the following loss function



Testing the model

To interpret the logistic regression model, the linear function and distribution for class one, i.e., $p(y = 1|\mathbf{x}, \boldsymbol{\zeta})$ can be inspected. These two functions are given by (external visualisation incoming!)

Logistic Regression
be like:



We can see the linear function is a plane that splits the data into two classes.

Interpreting the model

How can we interpret the model easily? Currently, only $z(\mathbf{x}, \boldsymbol{\zeta})$ contains linearity, while $p(y = 1|\mathbf{x}, \boldsymbol{\zeta})$ is non-linear due to $\sigma(u)$. Well, let's invert $\sigma(u)$ (conveniently is given by the logit function):

$$\sigma^{-1}(p_i) = \ln \frac{p_i}{1 - p_i}.$$

Thus, the logit function can be applied as follows to

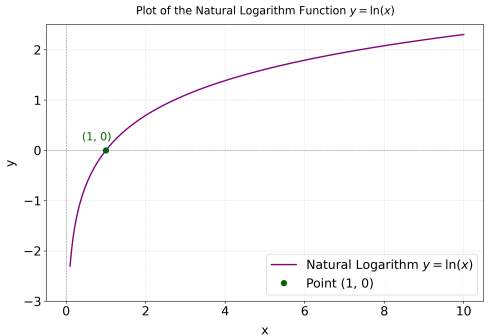
$$\ln \frac{p(y = 1|\mathbf{x}, \boldsymbol{\zeta})}{p(y = 0|\mathbf{x}, \boldsymbol{\zeta})} = \mathbf{w}^T\mathbf{x} + b = z(\boldsymbol{\zeta}, \mathbf{x}).$$

This is primarily useful as it provides an indication of what is being learnt by the model: the mapping z represents the log-odds between the two classes.

Interpreting the model

What is the implication of representing the log-odds? Let's consider these three scenarios:

1. $p(y = 1|\mathbf{x}, \boldsymbol{\zeta}) = p(y = 0|\mathbf{x}, \boldsymbol{\zeta})$
2. $p(y = 1|\mathbf{x}, \boldsymbol{\zeta}) > p(y = 0|\mathbf{x}, \boldsymbol{\zeta})$
3. $p(y = 1|\mathbf{x}, \boldsymbol{\zeta}) < p(y = 0|\mathbf{x}, \boldsymbol{\zeta})$



Interpreting the model

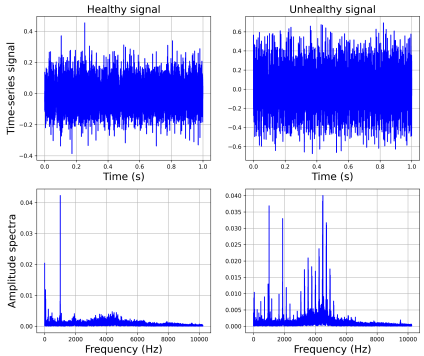
Scenario: All $\mathbf{x}$ values are non-negative

If all input features in $\mathbf{x}$ are non-negative ($x_i \geq 0$), the implication for interpreting the model is primarily related to the sign of the corresponding model parameters (weights, $\mathbf{w}$, and bias, b).

- If all components of the weight vector $\mathbf{w}$ are also non-negative (excluding the bias term b, which can be any real number), then the log-odds $z = \mathbf{w}^T\mathbf{x} + b$ will tend to be larger (more positive) for larger non-negative x values.
- A positive w_i for a non-negative x_i will contribute positively to z. Conversely, a negative w_i would contribute negatively to z.
- Therefore, if your features are strictly non-negative, the magnitude and sign of your learned weights $\mathbf{w}$ directly indicate the directional influence of each feature on the log-odds (and thus on the probability of $y = 1$).
  - A large positive w_i means that an increase in x_i strongly increases the log-odds of $y = 1$.
  - A large negative w_i means that an increase in x_i strongly decreases the log-odds of $y = 1$.

Applying the model

So let's take this into the real world. Consider the scenario where we obtain vibration data from a rotating machine. The vibration data is assumed to contain crucial information related to the instantaneous asset health state.

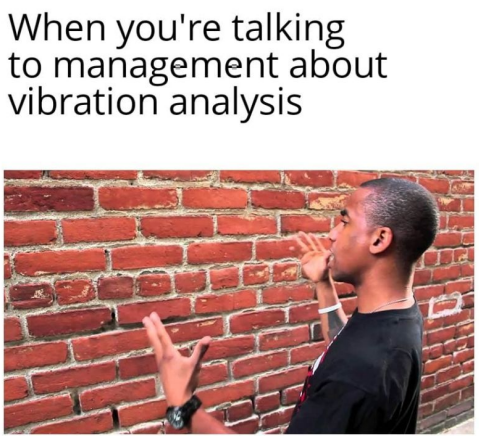


Applying the Model: Discussion Points

1. **Utility of Example:** What makes this a particularly useful example for the application of the model?
   - *Answer:* The model isn't trained on the raw time-series data directly. We develop it in data that is non-negative: The amplitude spectra.
2. **Concept Origin:** How was this approach conceived?
   - *Answer:* The concept is credited to Hou et al. [2], whose work serves as the foundation for this methodology.
3. **Dataset Selection:** Which dataset will be utilised, and why?
   - *Answer:* The IMS (Intelligent Maintenance Systems) bearing fault Diagnosis dataset will be used due to (1) popularity and (2)reference [2] uses it. This allows for direct methodological comparisons.

A self-introspective interlude

Me rn:



Paper process

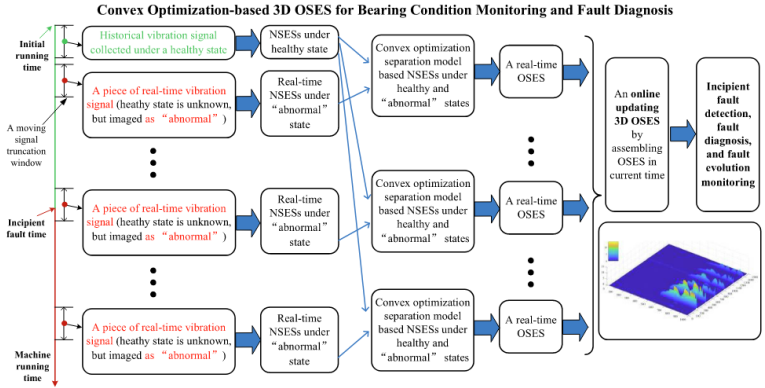
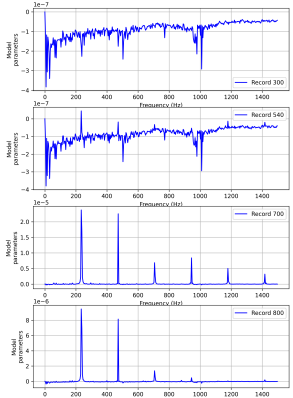


Fig. 2. Graphical explanation of convex optimization-based 3D OSES for bearing condition monitoring and fault diagnosis.

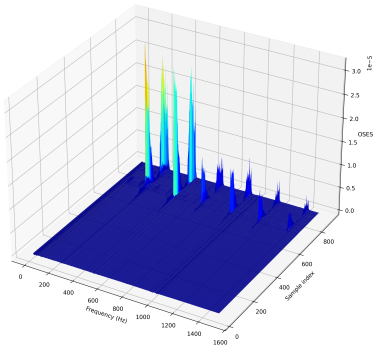
Results

Solution weights for specific vibration signals:



Results

Solution weights for all vibration signals:



Conclusion

I hope that you have gained some insight into:

1. Logistic regression and how it works.
2. How logistic regression can be used for interpretability.
3. How this interpretability can be used for fault diagnostics.

Thank you for listening!

References

[1] Calvin Cordozar Broadus Jr. (n.d.) I Wanna Thank Me.

[2] Hou et al. Interpretable online updated weights: Optimized square envelope spectrum for machine condition monitoring and fault diagnosis, Mechanical Systems and Signal Processing, Volume 169, 15 April 2022, 108779