



Logistic Regression

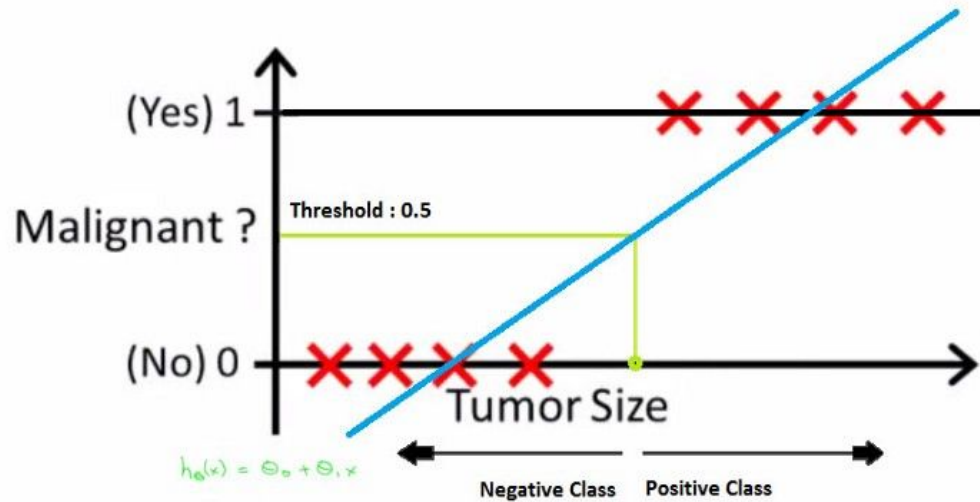
Galuh Kirana Miftahul Jannah - 16/398507/PA/17468

Jota Fauzan - 16/398515/PA/17476

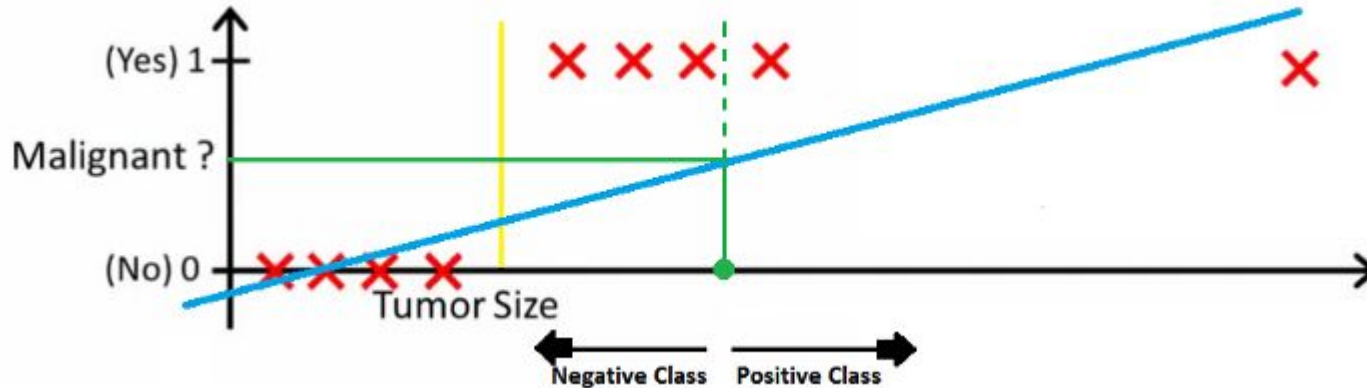
Emerald M - 16/398501/PA/17462

Vincent Michael Sutanto - 16/398531/PA/17492

Masalah pada Linear Regression

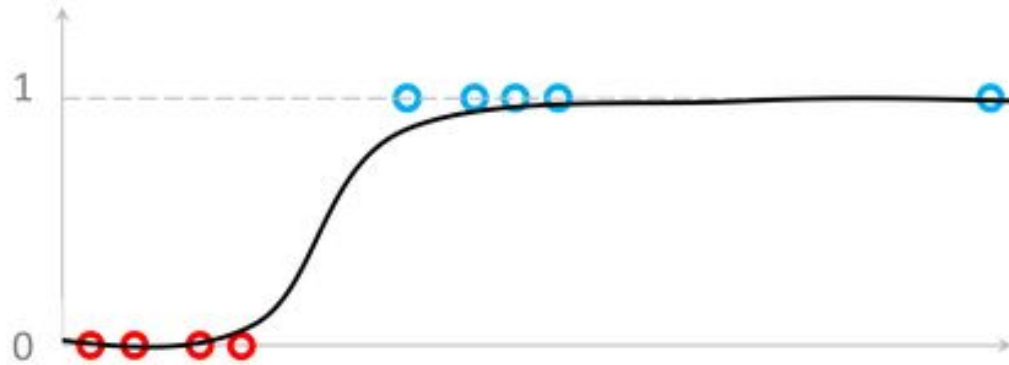


Masalah pada Linear Regression



Outlier Data akan mengganggu
prediksi dari Linear Regression

Solusi: Logistic Regression





Apa itu Logistic Regression?

Logistic Regression adalah sebuah **algoritma klasifikasi** untuk mencari hubungan antara **fitur-fitur diskrit / kontinu** dengan probabilitas hasil **output diskrit** tertentu.

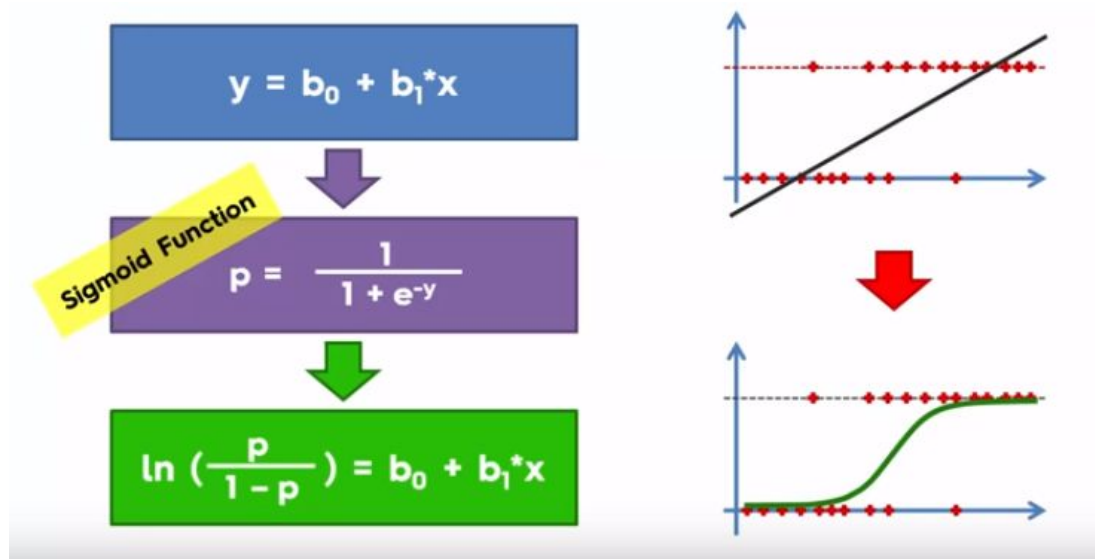


Tipe-tipe Logistic Regression

- 1. Binary Logistic Regression**
hanya memiliki 2 output
- 2. Multinomial Logistic Regression**
output lebih dari 2 tanpa memperhatikan urutan
- 3. Ordinal Logistic Regression**
output lebih dari 2 dengan memperhatikan urutan

Logit Function

Logit Function



Penentuan Koefisien Logit Function

Maximum Likelihood

—



Maximum Likelihood

Adalah cara yang digunakan untuk menentukan posisi Sigmoid yang merepresentasikan klasifikasi data paling baik

Learning Resource:

<https://www.youtube.com/watch?v=yIYKR4sgzI8>

<https://www.youtube.com/watch?v=XepXtl9YKwc>



Maximum Likelihood

$$\ln \left(\frac{p}{1-p} \right) = b_0 + b_1 \cdot x$$

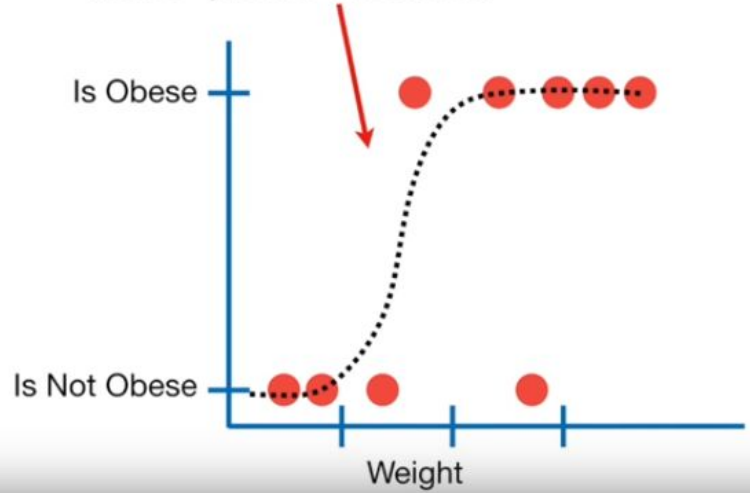
Nilai b_0 dan b_1 akan diupdate secara terus menerus untuk mendapatkan hasil Maximum Likelihood

Update nilai b_0 mempengaruhi posisi sigmoid pada sumbu x

Update nilai b_1 mempengaruhi posisi kelandaian sigmoid

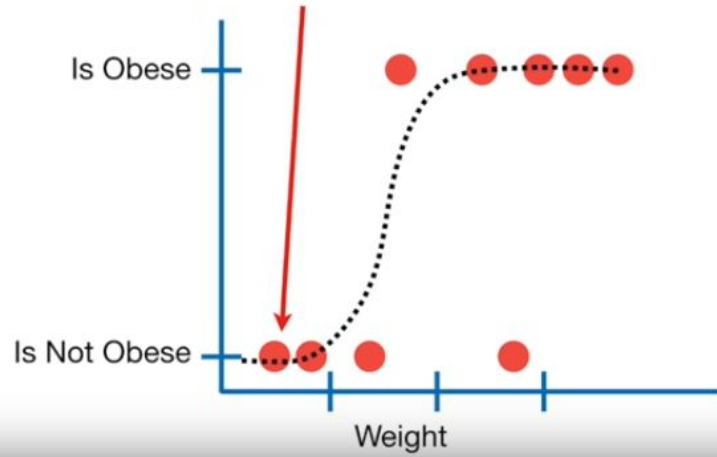
Maximum Likelihood

You pick a probability, scaled by weight, of observing an obese mouse - just like this curve...



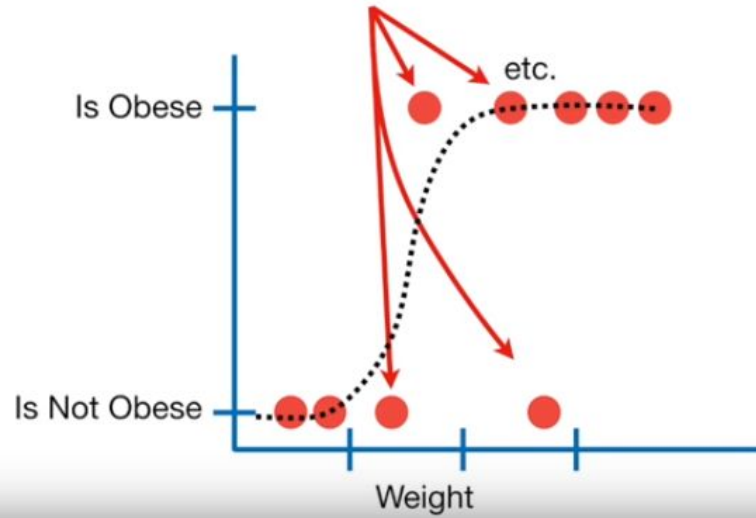
Maximum Likelihood

...and you use that to calculate the
likelihood of observing a non-
obese mouse that weighs this
much...



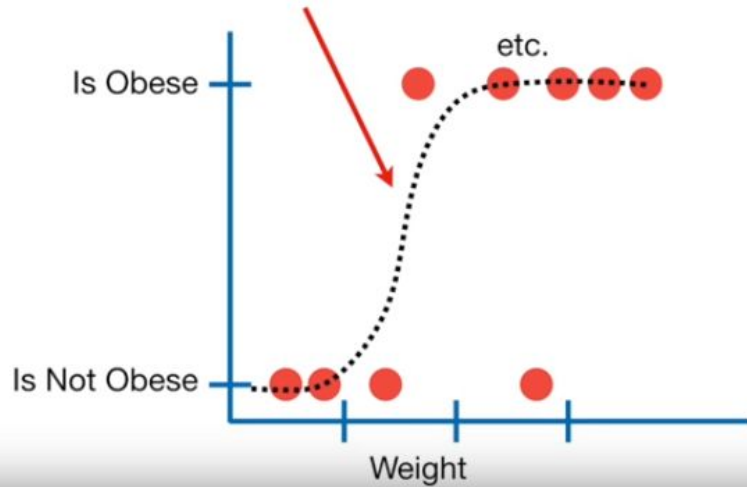
Maximum Likelihood

...and you do that for all of the mice...



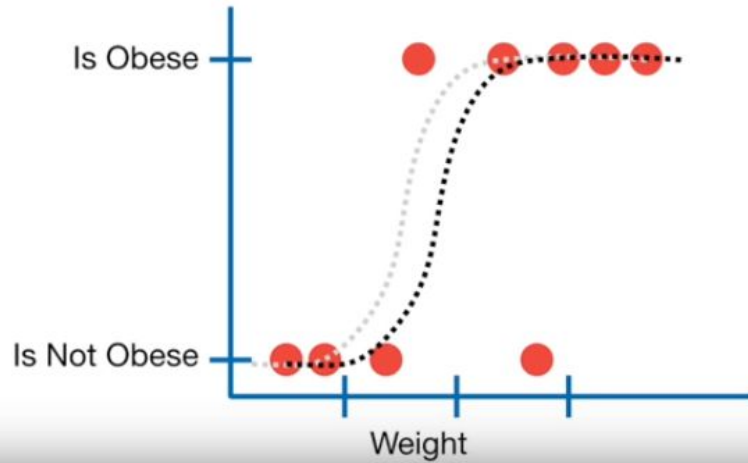
Maximum Likelihood

...and lastly you multiply all of those likelihoods together. That's the likelihood of the data given this line.



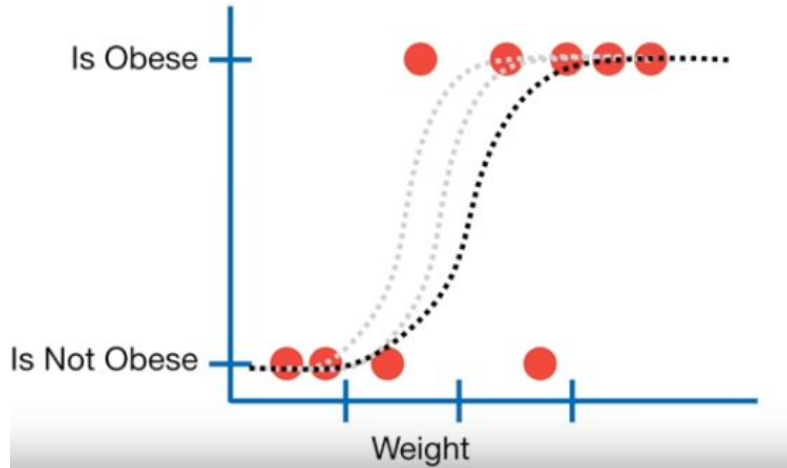
Maximum Likelihood

Then you shift the line and calculate a new likelihood of the data...

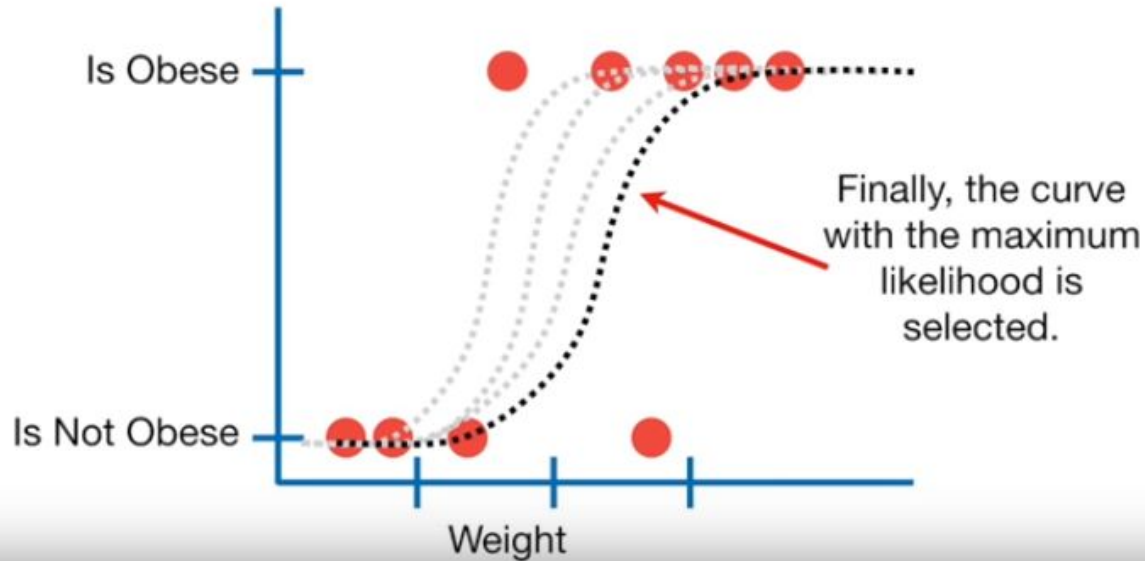


Maximum Likelihood

...then shift the line and calculate the likelihood again...



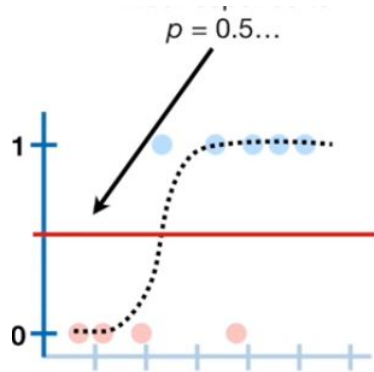
Maximum Likelihood



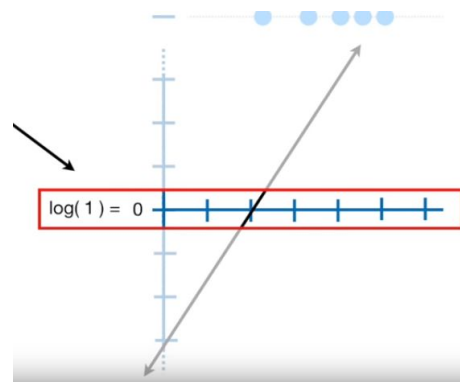
Penentuan Koefisien Logit Function

Penentuan Koefisien untuk Logit Function

Sumbu Y dari logit function kita ubah kedalam bentuk linear kembali dengan fungsi $\log(\text{odds})$ (dari 0 sampai 1 menjadi $-\infty$ sampai $+\infty$)



$$\log\left(\frac{p}{1-p}\right)$$



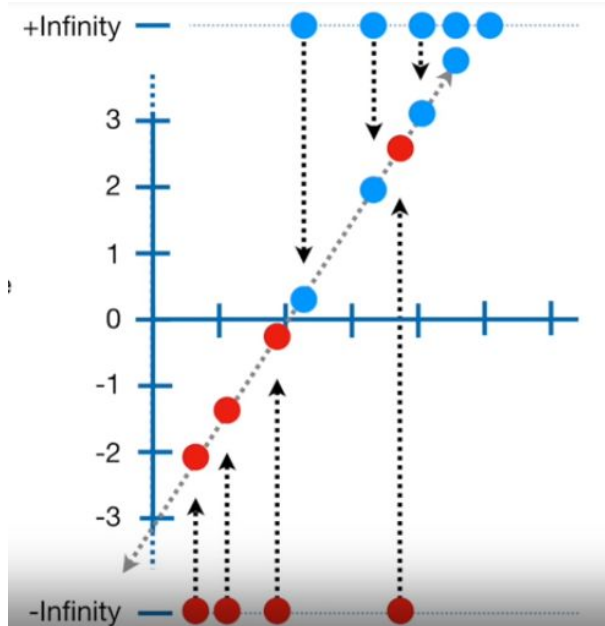


Penentuan Koefisien untuk Logit Function

Kenapa diubah kedalam garis kembali?

Koefisien Logit Function dipengaruhi oleh fungsi garis lurus, sehingga mengubah posisi / kemiringan garis lurus akan mempengaruhi bentuk dari Logit Function

Penentuan Koefisien untuk Logit Function



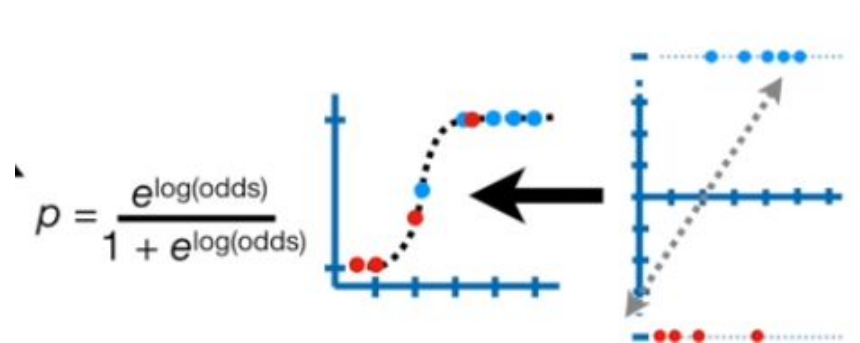
Proyeksikan setiap data ke dalam garis sementara.

Setiap data sekarang memiliki nilai $\log(\text{odds})$

(contoh data merah paling kiri memiliki nilai $\log(\text{odds}) = -2.1$)

Penentuan Koefisien untuk Logit Function

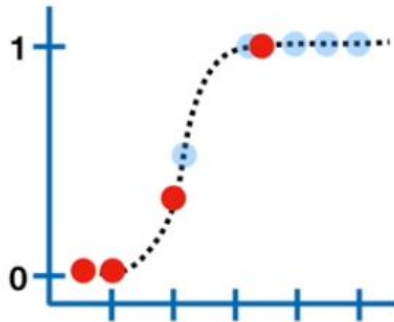
nilai **log(odds)** setiap data
kita **kembalikan ke dalam**
grafik sigmoid dengan
formula berikut:



Penentuan Koefisien untuk Logit Function

Hitung dan catat Likelihood dari garis yang dibentuk.

Likelihood = $\text{Log}(P(\text{data1}) + P(\text{data2}) + \dots + P(\text{data n}))$

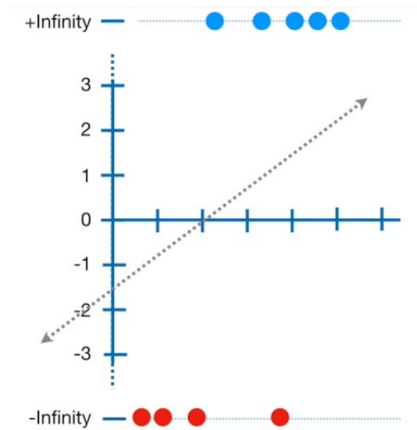
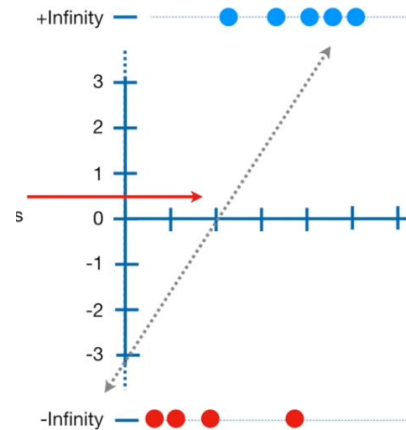


$$\begin{aligned} &= \log(0.49) + \log(0.9) + \log(0.91) + \log(0.91) + \\ &\quad \log(0.92) + \log(1 - 0.9) + \log(1 - 0.3) + \\ &\quad \log(1 - 0.01) + \log(1 - 0.01) \end{aligned}$$

Penentuan Koefisien untuk Logit Function

Selanjutnya kita **rotasi garis bayangan**, dan **hitung kembali likelihood**nya.

Bandingkan dengan likelihood sebelumnya, jika nilai lebih tinggi maka model yang dihasilkan **lebih baik**



R -squared



R-squared

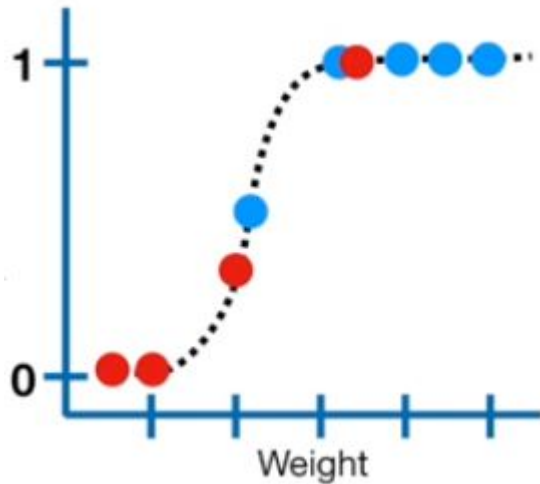
Adalah cara yang digunakan untuk mengetahui apakah nilai Maximum Likelihood yang didapat merepresentasikan model dengan baik (R-squared = 1) atau tidak (R-squared = 0)

Learning Resource:

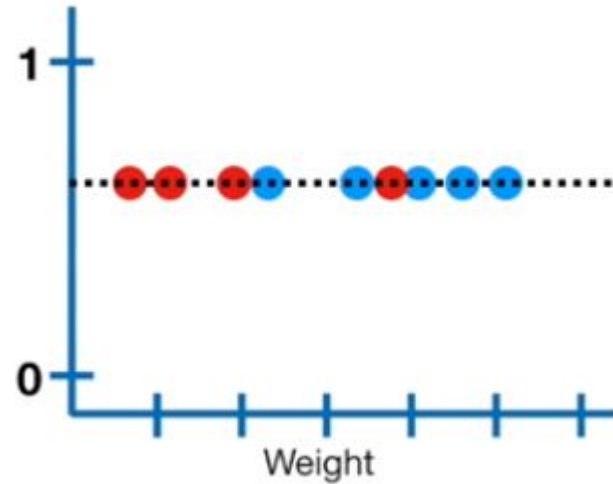
<https://www.youtube.com/watch?v=xxFYro8QuXA>

Parameter R-squared

Maximum Likelihood



Bad-fit Likelihood





Parameter R-squared: bad-fit

Garis lurus pada Bad-fit didapat dengan langkah:

$$Y = \text{number of Class-1} / \text{number of data}$$

Sehingga Bad-fit Likelihood-nya:

$$\text{Log}(Y) + \text{Log}(Y) + \dots + \text{Log}(1-Y) + \text{Log}(1-Y)$$



R-squared

Dari **Maximum Likelihood** dan **Bad-fit Likelihood** dapat dibentuk formula R-squared sebagai berikut:

$$\text{R-squared} = \frac{\text{Bad-fit Likelihood} - \text{Maximum Likelihood}}{\text{Bad-fit Likelihood}}$$

P-value



P-Value

$2(\text{Maximum Likelihood} - \text{Bad-fit Likelihood}) = \text{Chi-squared Value}$

Dengan Chi-squared Value = 1

P-value digunakan untuk mengetahui fitur-fitur yang memiliki pengaruh banyak untuk model.

Materinya nyari sendiri ya :(



Cara lain menentukan
koefisien Logit Function:

Stochastic Gradient Descent

Learning source : <https://machinelearningmastery.com/logistic-regression-tutorial-for-machine-learning/>



SGD

Langkah-langkah yang dilakukan dalam menentukan koefisien dengan SGD:

1. Hitung prediksi dengan nilai koefisien sementara
2. Hitung nilai koefisien baru berdasarkan error nilai koefisien sebelumnya
3. Repeat



SGD

Misalkan kita memiliki logit function sebagai berikut:

$$P(\text{data}) = 1 / (1 + e^{-(b_0 + b_1 * x_1 + b_2 * x_2)})$$

Inisialisasi bobot awal untuk b_0 , b_1 , dan b_2 :

$$b_0 = b_1 = b_2 = 0$$



SGD

Substitusi suatu data training:

$$x_1=2.7810836, x_2=2.550537003, P_{\text{target}}(\text{data}) = 0$$

Kedalam Logit function:

$$P(\text{data}) = 1 / (1 + e^{-(0 + 0*2.7810836 + 0*2.550537003)})$$

$$P_{\text{prediksi}}(\text{data}) = 0.5$$



SGD

Hitung nilai koefisien baru:

$$B_{\text{baru}} = B_{\text{lama}} + \alpha * (P_{\text{target}} - P_{\text{prediksi}}) * P_{\text{prediksi}} * (1 - P_{\text{prediksi}}) * x$$

Dan ulangi terus sampai mendapat akurasi yang baik (error kecil)

Data Preparation



Data Preparation

Beberapa hal yang perlu diperhatikan untuk menghasilkan model Logistic Regression yang baik:

1. Binary Output Variable
2. Remove Noise
3. Gaussian Distribution
4. Remove Correlated Input
5. Fail to Converge



Data Preparation: **Binary Output Variable**

Logistic Regression ditujukan untuk mengklasifikasi data kedalam 2 kelas (Representasi dengan nilai 0 dan 1)



Data Preparation: **Remove Noise**

Pertimbangkan untuk menghapus data outlier dan data yang mungkin memiliki label yang salah dari data training



Data Preparation: **Gaussian Distribution**

Transformasi data anda sebelum diolah dengan Logistic Regression untuk mendapatkan hasil model yang lebih baik



Data Preparation: **Remove Correlated Input**

Terlalu banyaknya input yang digunakan dapat menyebabkan model mengalami overfit. Pertimbangkan pilihan input yang sangat berkorelasi.



Data Preparation: **Fail to Converge**

Ada kemungkinan dimana model yang dihasilkan tidak konvergen dengan hasil yang diinginkan. Contoh penyebabnya:

- Terlalu banyak input yang berkorelasi satu dengan yang lainnya
- Data Training yang tidak berimbang (terlalu banyak label 0/1 yang menyebabkan kemiripan / likelihood condong kearah label yang banyak)

Referensi (artikel)



<https://towardsdatascience.com/understanding-logistic-regression-9b02c2aec102>

<https://medium.com/data-science-group-iitr/logistic-regression-simplified-9b4efe801389>

<https://www.statisticssolutions.com/what-is-logistic-regression/>

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

https://www.medcalc.org/manual/logistic_regression.php

<https://machinelearningmastery.com/>

Referensi (video)



<https://www.youtube.com/watch?v=yIYKR4sgzI8>

<https://www.youtube.com/watch?v=XepXtl9YKwc>

<https://www.youtube.com/watch?v=D8alok2P468>

<https://www.youtube.com/watch?v=nz-FrbAa8dY>

<https://www.youtube.com/watch?v=zUxZ95lXTco>

<https://www.youtube.com/watch?v=H6ii7NFdDeg&list=WL&index=30&t=391s>