Table of Contents

Data Archive Book

Welcome to the Data Archive Book.

==Logistic regression models the log-odds of the probability as a linear function of the input features.==

It models the probability of an input belonging to a particular class using a logistic (sigmoid) function.

The model establishes a decision boundary (threshold) in the feature space.

Logistic regression is best suited for cases where the decision boundary is approximately linear in the feature space.

Logistic [[Regression]] can be used for [[Binary Classification]] tasks.

Related Notes:

First Header	Second Header
Content Cell	Content Cell
Content Cell	Content Cell

- [[Logistic Regression Statsmodel Summary table]]
- [[Logistic Regression does not predict probabilities]]
- [[Interpreting logistic regression model parameters]]
- [[Model Evaluation]]
- To get [[Model Parameters]] use [[Maximum Likelihood Estimation]]

In [[ML_Tools]], see:

- [[Regression_Logistic_Metrics.ipynb]]

Key Concepts of Logistic Regression

Logistic Function (Sigmoid Function)

```
def greet():
    print("Hello, world!")
```

Logistic σ regression models the probability that an input belongs to a particular class using the logistic (sigmoid) function. This function maps any real-valued input into the range (0,1), representing the probability of belonging to the positive class (usually class 1).

The sigmoid function is defined as:

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

where

$z = \mathbf{w} \cdot \mathbf{x} + b$

Thus, the logistic regression model is given by:

$P(y = 1 \mid \mathbf{x}) = \sigma(z)$

Log odds: Transforming from continuous to 0-1

Logistic regression is based on the ==log-odds== (logit) transformation, which expresses probability in terms of odds:

$\text{Odds} = \frac{P(y=1|\mathbf{x})}{1-P(y=1|\mathbf{x})}$

Taking the natural logarithm of both sides gives the logit function:

$\log\left(\frac{P(y=1|\mathbf{x})}{1-P(y=1|\mathbf{x})}\right) = \mathbf{w} \cdot \mathbf{x} + b$

This equation shows that ==logistic regression models the log-odds of the probability as a linear function of the input features.==

Decision Boundary

- Similar to [[Support Vector Machines]], logistic regression defines a decision boundary that separates the two classes.
- The logistic function determines the probability of a data point belonging to a specific class. If this probability exceeds a given ==threshold== (typically 0.5), the model assigns the point to the positive class; otherwise, it is classified as negative.

[[Binary Classification]]

- Logistic regression is primarily used for binary classification tasks, where the target variable has only two possible values (e.g., "0" and "1").
- It can handle multiple independent variables (features) and assigns probabilities to the target classes based on the feature values.
- Examples include:
  - Predicting whether a tumor is malignant or benign (Breast Cancer dataset).
  - Determining whether a passenger survived the Titanic disaster (Titanic dataset).

No Residuals

- Unlike [[Linear Regression]], logistic regression does not compute standard residuals.
- Instead, [[Model Evaluation]] is performed by comparing predicted probabilities with actual class labels using metrics such as accuracy, precision, recall, and the [[Confusion Matrix]].

Also see:

Related terms:

- Cost function for logistic regression

- Gradient computation in logistic regression
- Regularized logistic regression
- Cost function for regularized logistic regression

Logistic regression can be extended to handle non-linear decision boundaries through:

- Polynomial features to capture more complex relationships.
- Regularization techniques to improve generalization.

Explaining logistic regression

Statsmodel has this summary table unlike [[Sklearn]]

Explanation of summary

The dependent variable is 'duration'. The model used is a Logit regression (logistic in common lingo), while the method

- Maximum Likelihood Estimation ([[MLE]]). It has clearly converged after classifying 518 observations.
- The Pseudo R-squared is 0.21 which is within the 'acceptable region'.
- The duration variable is significant and its coefficient is 0.0051.
- The constant is also significant and equals: -1.70 (p value close to 0)
- High p value, suggests to remove from model, drop one by one, ie [[Feature Selection]].

Specifically a graph such as, ![Pasted image 20240124095916.png]

2.

```
Optimization terminated successfully.
         Current function value: 0.546118
         Iterations 7
```

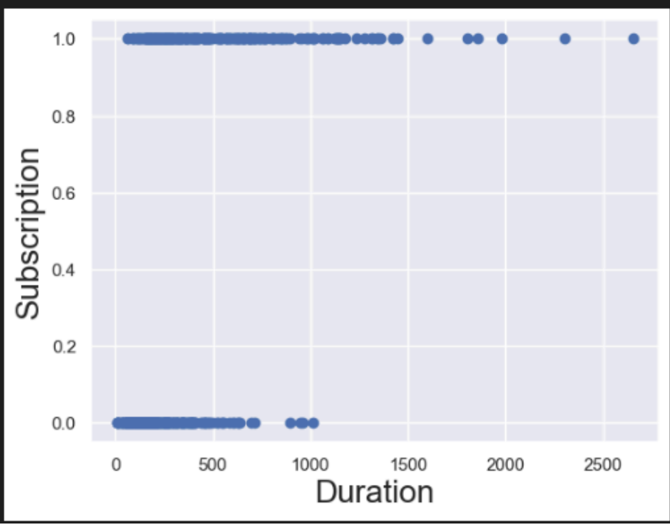
Logit Regression Results

Dep. Variable:	y	No. Observations:	518
Model:	Logit	Df Residuals:	516
Method:	MLE	Df Model:	1
Date:	Wed, 24 Jan 2024	Pseudo R-squ.:	0.2121
Time:	09:55:29	Log-Likelihood:	-282.89
converged:	True	LL-Null:	-359.05
Covariance Type:	nonrobust	LLR p-value:	5.387e-35

	coef	std err	z	P>|z|	[0.025	0.975]
const	-1.7001	0.192	-8.863	0.000	-2.076	-1.324
duration	0.0051	0.001	9.159	0.000	0.004	0.006

```
# Create a scatter plot of x1 (Duration, no constant) and y (Subscribed)
plt.scatter(x1,y,color = 'C0')
# Don't forget to label your axes!
plt.xlabel('Duration', fontsize = 20)
plt.ylabel('Subscription', fontsize = 20)
plt.show()
```

[7] ✓ 0.2s



3.

```
Optimization terminated successfully.
         Current function value: 0.546118
         Iterations 7
```

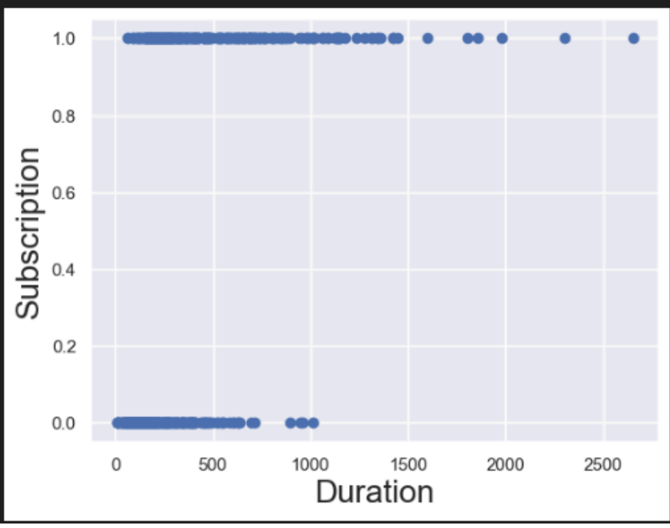
Logit Regression Results

Dep. Variable:	y	No. Observations:	518
Model:	Logit	Df Residuals:	516
Method:	MLE	Df Model:	1
Date:	Wed, 24 Jan 2024	Pseudo R-squ.:	0.2121
Time:	09:55:29	Log-Likelihood:	-282.89
converged:	True	LL-Null:	-359.05
Covariance Type:	nonrobust	LLR p-value:	5.387e-35

	coef	std err	z	P>|z|	[0.025	0.975]
const	-1.7001	0.192	-8.863	0.000	-2.076	-1.324
duration	0.0051	0.001	9.159	0.000	0.004	0.006

```
# Create a scatter plot of x1 (Duration, no constant) and y (Subscribed)
plt.scatter(x1,y,color = 'C0')
# Don't forget to label your axes!
plt.xlabel('Duration', fontsize = 20)
plt.ylabel('Subscription', fontsize = 20)
plt.show()
```

[7] 0.2s



ℕ