GENERAL ASSEMBLY

INTRO TO DATA SCIENCE

LECTURE 6: LOGISTIC REGRESSION

LAST TIME:

- INVERSE MATRICES
- BUILDING GOOD LINEAR MODELS
- LINEAR REGRESSION

QUESTIONS?

I. POLYNOMIAL REGRESSION
II. REGULARIZATION
III. LOGISTIC REGRESSION
IV. INTERPRETING RESULTS

EXERCISES:
LASSO AND LOGISTIC REGRESSION WITH SCIKIT LEARN

I: POLYNOMIAL REGRESSION

Consider the following **polynomial regression** model:

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \varepsilon$$

Consider the following **polynomial regression** model:

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \varepsilon$$

Q: This represents a **nonlinear relationship**. Is it still a linear model?

Consider the following **polynomial regression** model:

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \varepsilon$$

Q: This represents a **nonlinear relationship**. Is it still a linear model?

A: Yes, because it's linear in the β's!

"Although polynomial regression fits a nonlinear model to the data, as a statistical estimation problem it is linear, in the sense that the regression function E(y|x) is linear in the unknown parameters that are estimated from the data. For this reason, polynomial regression is considered to be a special case of multiple linear regression." -- Wikipedia

Polynomial regression allows us to fit very complex curves to data.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

Polynomial regression allows us to fit very complex curves to data.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

But there is one problem with the model we've written down so far.

Polynomial regression allows us to fit very complex curves to data.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

But there is one problem with the model.

Q: Does anyone know what it is?

Polynomial regression allows us to fit very complex curves to data.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

But there is one problem with the model.

Q: Does anyone know what it is?

A: This model violates one of the assumptions of linear regression!

POLYNOMIAL REGRESSION

This model displays **multicollinearity**, which means the predictor variables are highly correlated with each other.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

```
> x <- seq(1, 10, 0.1)
> cor(x^9, x^10)
[1] 0.9987608
```

This model displays **multicollinearity**, which means the predictor variables are highly correlated with each other.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

Multicollinearity causes the linear regression model to break down, because it can't tell the predictor variables apart.

This model displays **multicollinearity**, which means the predictor variables are highly correlated with each other.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

NOTE

This results in a singularity. We will see an example of this in just a minute!

Multicollinearity causes the linear regression mo break down, because it can't tell the predictor variables apart.

Q: What can we do about this?

Q: What can we do about this?

A: Replace the correlated predictors with uncorrelated predictors.

Q: What can we do about this?

A: Replace the correlated predictors with uncorrelated predictors.

$$y = \alpha + \beta_1 f_1(x) + \beta_2 f_2(x^2) + \ldots + \beta_n f_n(x^n) + \varepsilon$$

Q: What can we do about this?

A: Replace the correlated predictors with uncorrelated predictors.

$$y = \alpha + \beta_1 f_1(x) + \beta_2 f_2(x^2) + \ldots + \beta_n f_n(x^n) + \varepsilon$$

OPTIONAL NOTE

These polynomial functions form an orthogonal basis of the function space.

So far, we've seen how polynomial regression allows us to fit complex nonlinear relationships, and even to avoid multicollinearity (by using basis functions).

So far, we've seen how polynomial regression allows us to fit complex nonlinear relationships, and even to avoid multicollinearity (by using basis functions).

Q: Can a regression model be too complex?

II: REGULARIZATION

Q: What's **overfitting**?

Q: What's **overfitting**?

Overfitting occurs when a model matches the signal instead of the noise.

Q: What's **overfitting**?

Overfitting occurs when a model matches the signal instead of the noise.

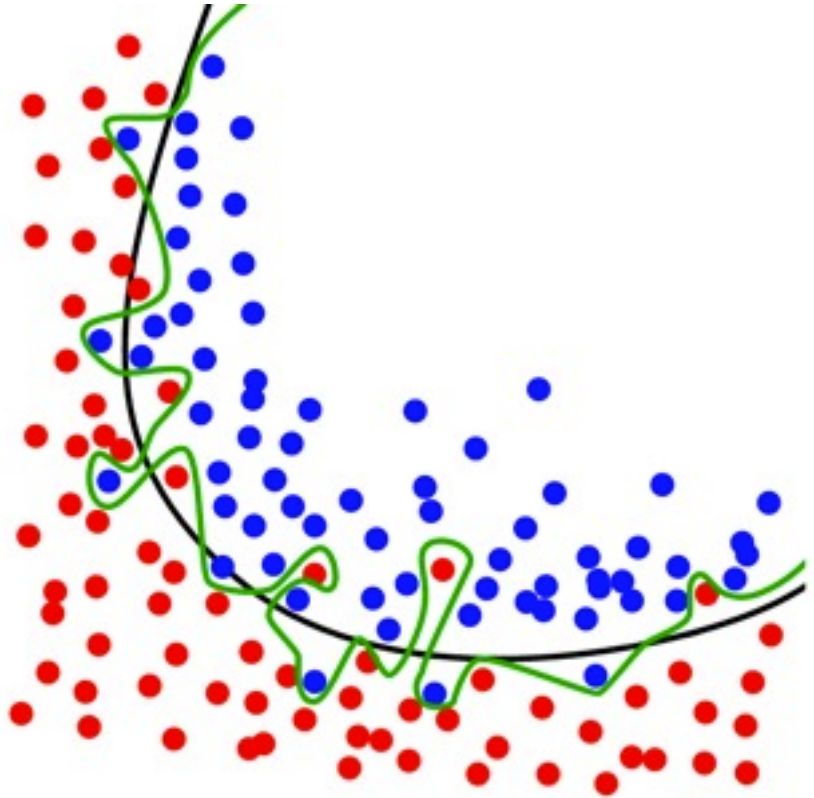
Noise: Extra "cruft" that doesn't contribute to a readable prediction.
Signal: Clean, elegant interpretation of the data

Q: What's **overfitting**?

Overfitting occurs when a model matches the signal instead of the noise.

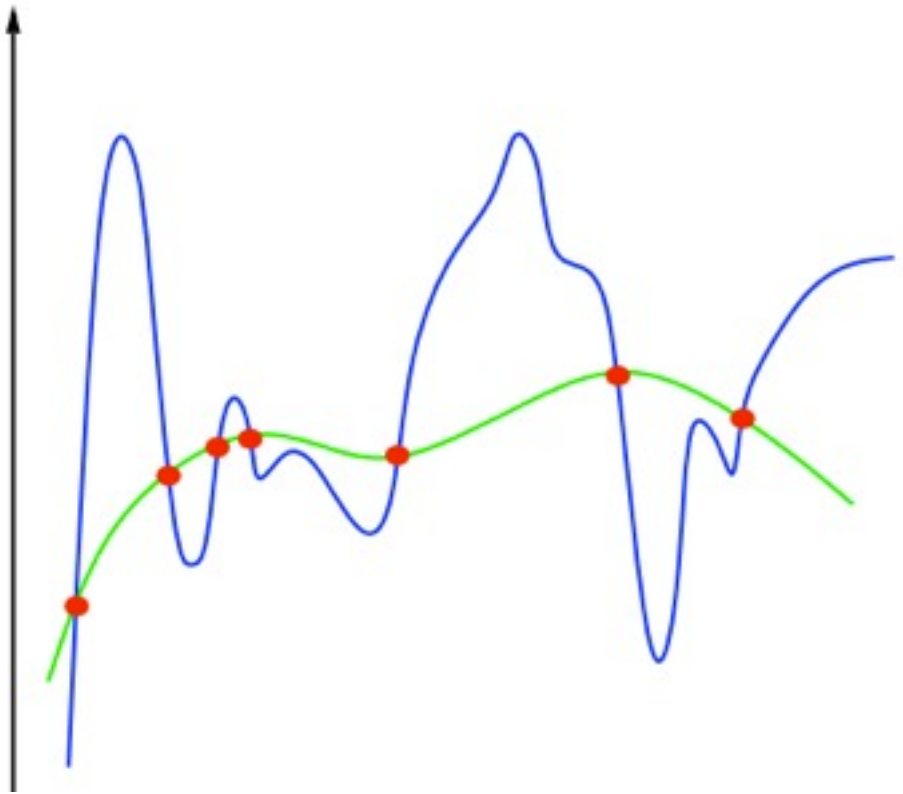
This happens when our model is too complex!

OVERFITTING EXAMPLE (CLASSIFICATION)



source: http://upload.wikimedia.org/wikipedia/commons/1/19/Overfitting.svg

OVERFITTING EXAMPLE (REGRESSION)



source: http://www.mit.edu/~9.520/spring12/slides/class02/class02.pdf

Q: How do we define the **complexity** of a regression model?

Q: How do we define the **complexity** of a regression model?

A: One method is to define **complexity** as a function of the size of the coefficients.

Q: How do we define the **complexity** of a regression model?

A: One method is to define **complexity** as a function of the size of the coefficients.

Ex 1: $\Sigma |\beta_i|$

Ex 2: $\Sigma \beta_i^2$

Q: How do we define the **complexity** of a regression model?

A: One method is to define **complexity** as a function of the size of the coefficients.

Ex 1: $\Sigma |\beta_i|$ this is called the **L1-norm**

Ex 2: $\Sigma \beta_i^2$ this is called the **L2-norm**

These measures of complexity lead to the following regularization techniques:

These measures of complexity lead to the following regularization techniques:

L1 regularization: $y = \Sigma \beta_i x_i + \varepsilon \quad \text{st.} \quad \Sigma \left|\beta_i\right| < s$

These measures of complexity lead to the following regularization techniques:

L1 regularization: $y = \Sigma \beta_i x_i + \varepsilon \quad \text{st.} \quad \Sigma \left|\beta_i\right| < s$

L2 regularization: $y = \Sigma \beta_i x_i + \varepsilon \quad \text{st.} \quad \Sigma \beta_i^2 < s$

These measures of complexity lead to the following regularization techniques:

L1 regularization: $y = \Sigma \beta_i x_i + \varepsilon \quad st. \quad \Sigma |\beta_i| < s$

L2 regularization: $y = \Sigma \beta_i x_i + \varepsilon \quad st. \quad \Sigma \beta_i^2 < s$

Regularization refers to the method of preventing overfitting by explicitly controlling model complexity.

These regularization problems can also be expressed as:

L1 regularization: $\min(\|y - x\beta\|^2 + \lambda\|x\|)$

L2 regularization: $\min(\|y - x\beta\|^2 + \lambda\|x\|^2)$

These regularization problems can also be expressed as:

L1 regularization: $\min(\|y - x\beta\|^2 + \lambda\|x\|)$

L2 regularization: $\min(\|y - x\beta\|^2 + \lambda\|x\|^2)$

but more importantly, we can think about the use cases of these two more clearly this way:

L1 regularization: $\min(\|y - x\beta\|^2 + \lambda\|x\|)$

Used when we have small data but many features.

L2 regularization: $\min(\|y - x\beta\|^2 + \lambda\|x\|^2)$

Used in just about all other cases.

These regularization problems can also be expressed as:

L1 regularization: $\min(\|y - x\beta\|^2 + \lambda\|x\|)$

L2 regularization: $\min(\|y - x\beta\|^2 + \lambda\|x\|^2)$

This (Lagrangian) formulation reflects the fact that there is a cost associated with regularization.

Q: Any ideas?

Q: What are bias and variance?

Q: What are bias and variance?

A: **Bias** refers to predictions that are systematically inaccurate.

Q: What are bias and variance?

A: **Bias** refers to predictions that are systematically inaccurate.

Variance refers to predictions that are generally inaccurate.

Q: What are bias and variance?

A: **Bias** refers to predictions that are systematically inaccurate.

Variance refers to predictions that are generally inaccurate.

BIAS AND VARIANCE

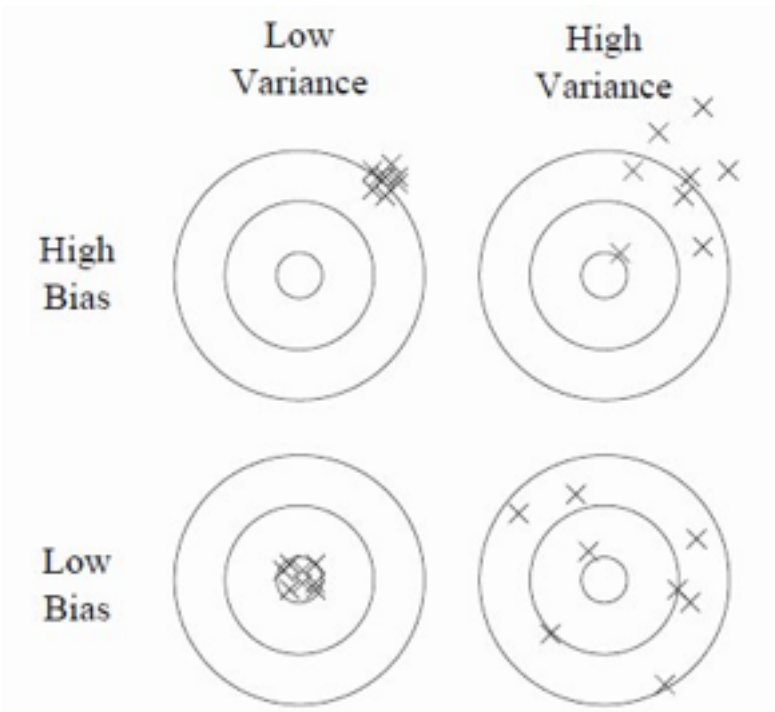


Figure 1: Bias and variance in dart-throwing.

source: http://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf

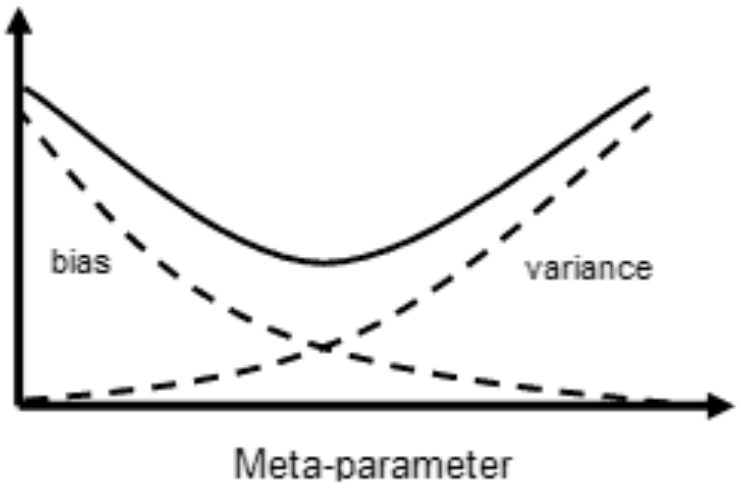
Q: What are bias and variance?

A: **Bias** refers to predictions that are systematically inaccurate.

Variance refers to predictions that are generally inaccurate.

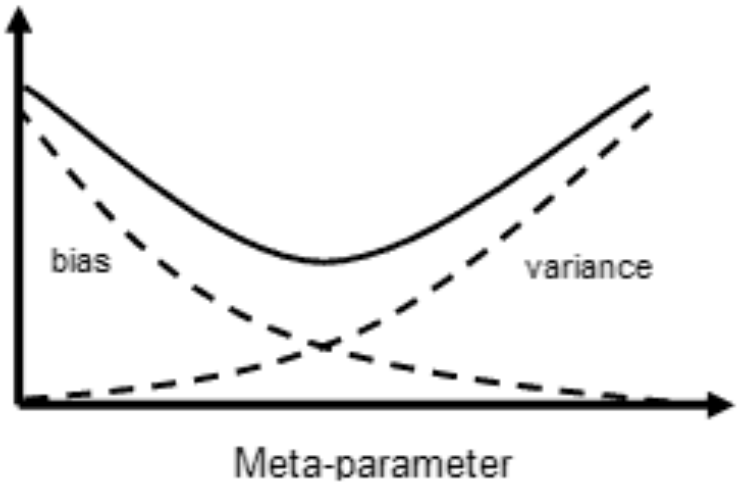
It turns out (after some math) that the generalization error in our model can be decomposed into a bias component and variance component.

This is another example of the **bias-variance tradeoff.**



source: http://www.isu.edu/chem/images/kalivasmeta.gif

This is another example of the **bias-variance tradeoff.**



NOTE

The "meta-parameter" here is the lambda we saw above.

A more typical term is "hyperparameter".

source: http://www.isu.edu/chem/images/kalivasmeta.gif

This tradeoff is regulated by a **hyperparameter** λ, which we've already seen:

L1 regularization: $y = \Sigma \beta_i x_i + \varepsilon \quad \text{st.} \quad \Sigma |\beta_i| < \lambda$

L2 regularization: $y = \Sigma \beta_i x_i + \varepsilon \quad \text{st.} \quad \Sigma \beta_i^2 < \lambda$

We should take advantage of generalization to trade off variance in our data for bias in our fit, which will overall produce a clearer and better overall fit to our data!

LAB. RIDGE REGRESSION

III. LOGISTIC REGRESSION

	continuous	categorical
supervised	???	???
unsupervised	???	???

	continuous	categorical
supervised	regression	classification
unsupervised	dimension reduction	clustering

Q: What is **logistic regression**?

Q: What is **logistic regression**?

A: **A generalization of the linear regression model to classification problems.**

Q: What is **logistic regression**?

A: **A generalization of the linear regression model to classification problems.**

Q: (Review) What does regularization do for our model?

Q: What is **logistic regression**?

A: **A discriminative classification algorithm.**

Discriminative: result does not depend completely the data set.

In linear regression, we used a set of covariates to predict the value of a (continuous) outcome variable.

In linear regression, we used a set of covariates to predict the value of a (continuous) outcome variable.

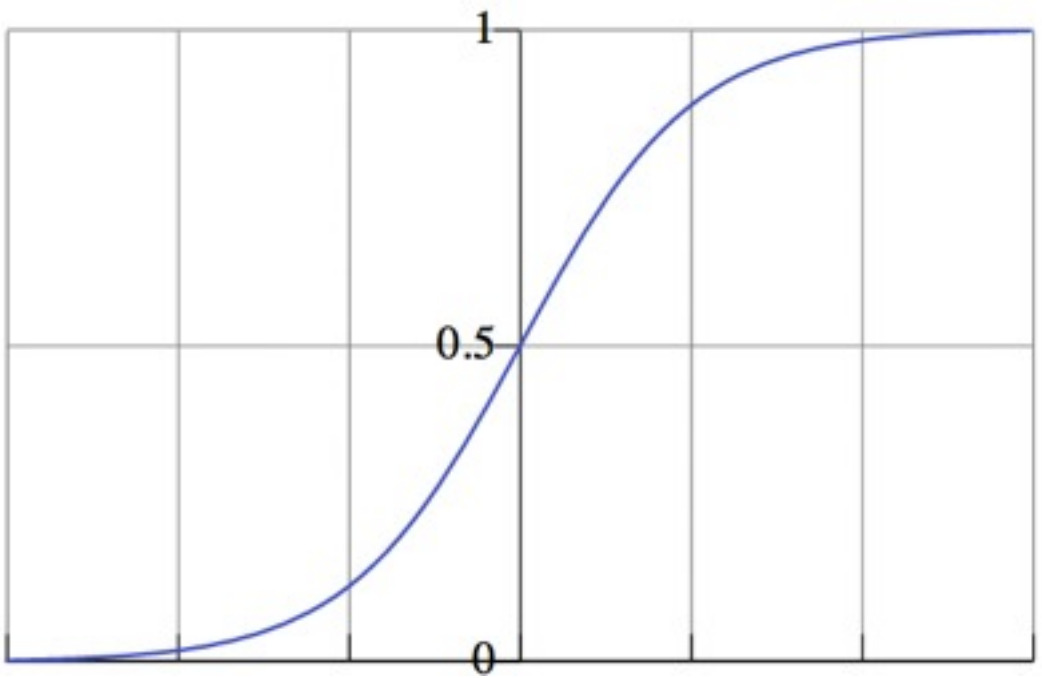
In logistic regression, we use a set of covariates to predict probabilities of (binary) class membership.

In linear regression, we used a set of covariates to predict the value of a (continuous) outcome variable.

In logistic regression, we use a set of covariates to predict probabilities of (binary) class membership.

These probabilities are then mapped to class labels, thus solving the classification problem.

probability of belonging to class

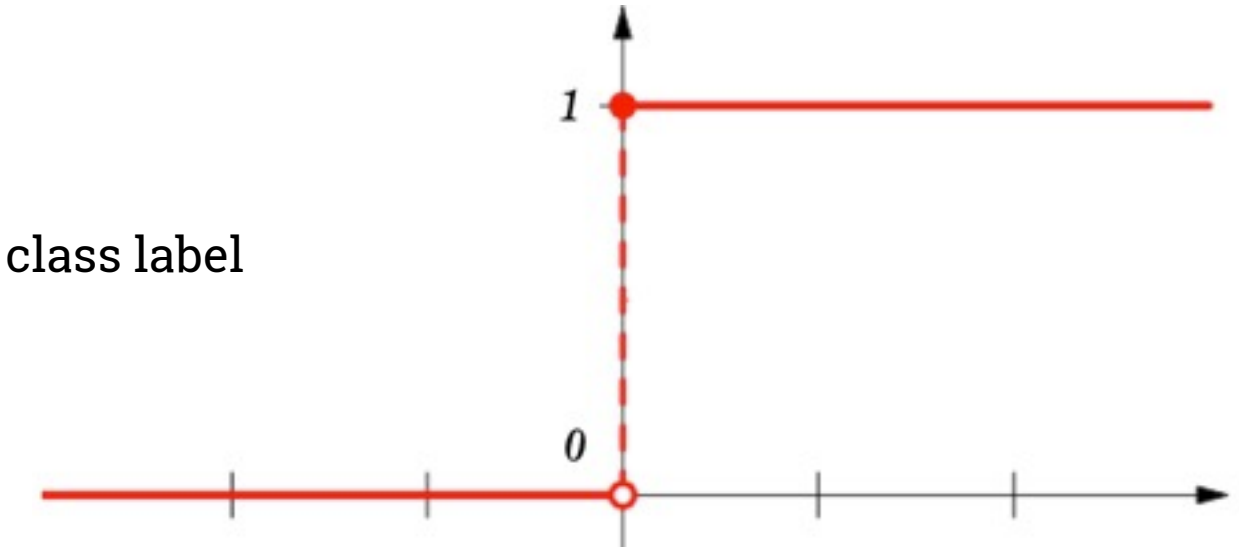


value of independent variable

NOTE

Probability predictions look like this.

CLASS LABELS



NOTE

Probabilities are "snapped" to class labels (eg by threshholding at 50%).

Note that Logistic Regression is primarily used to solve a **binary classification problem**.

Note that Logistic Regression is primarily used to solve a **binary classification problem**.

Examples:
Was this credit transaction fraudulent? (Y/N)
User a boy or a girl?
Do I have x disease?
Should this stock be bought or sold?

The logistic regression model is an extension of the linear regression model, with a couple of important differences.

The logistic regression model is an extension of the linear regression model, with a couple of important differences.

Difference 1: Outcome Variables

The logistic regression model is an extension of the linear regression model, with a couple of important differences:

Difference 1: Outcome Variables

Difference 2: Error Terms

IV. OUTCOME VARIABLES

The key variable in any regression problem is the **conditional mean** of the outcome variable y given the value of the covariate x: $E(y|x)$

The key variable in any regression problem is the **conditional mean** of the outcome variable y given the value of the covariate x: $E(y|x)$

In linear regression, we assume that this conditional mean is a linear function taking values in $(-\infty, +\infty)$:

$$E(y|x) = \alpha + \beta x$$

Q: How is this different from just using a linear regression to solve a classification problem?

Q: How is this different from just using a linear regression to solve a classification problem?

A: If the original values are not scaled correctly, then the results for data once classified **0** may be reclassified as **1.**

Q: How is this different from just using a linear regression to solve a classification problem?

A: If the original values are not scaled correctly, then the results for data once classified **0** may be reclassified as **1.**

We don't want that to happen!

In logistic regression, we've seen that the conditional mean of the outcome variable takes values only in the unit interval [0, 1].

In logistic regression, we've seen that the conditional mean of the outcome variable takes values only in the unit interval [0, 1].

0 = negative class, 1 = positive class

In logistic regression, we've seen that the conditional mean of the outcome variable takes values only in the unit interval [0, 1].

0 = negative class, 1 = positive class

The first step in extending the linear regression model to logistic regression is to map the outcome variable E(y|x) into the unit interval.

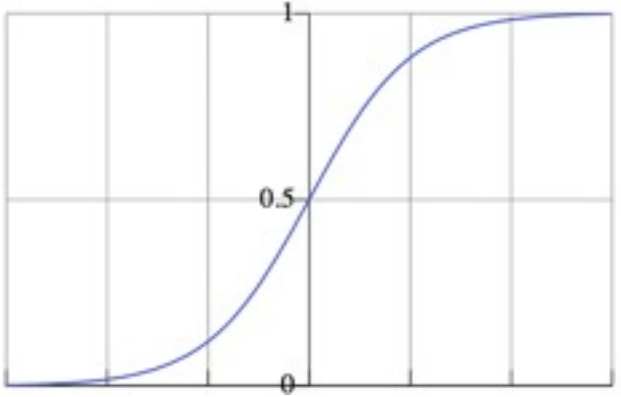
A: By using a transformation called the logistic function:

$$E(y|x) = \pi(x) = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}}$$

A: By using a transformation called the logistic function:

$$E(y|x) = \pi(x) = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}}$$

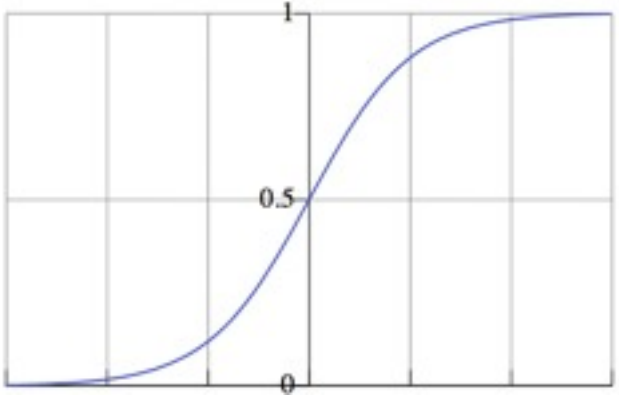
We've already seen what this looks like:



A: By using a transformation called the logistic function:

$$E(y|x) = \pi(x) = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}}$$

We've already seen what this looks like:



NOTE

For any value of x, y is in the interval [0, 1]

This is a nonlinear transformation!

The **logit function** is an important transformation of the logistic function. Notice that it returns the linear model!

$$g(x) = ln(\frac{\pi(x)}{1-\pi(x)}) = \alpha + \beta x$$

The **logit function** is an important transformation of the logistic function. Notice that it returns the linear model!

$$g(x) = ln(\frac{\pi(x)}{1-\pi(x)}) = \alpha + \beta x$$

The logit function is also called the log-odds function.

The **logit function** is an important transformation of the logistic function. Notice that it returns the linear model!

$$g(x) = ln(\frac{\pi(x)}{1-\pi(x)}) = \alpha + \beta x$$

NOTE

This name hints at its usefulness in interpreting our results.

We will see why shortly.

The logit function is also called the log-odds function.

V. ERROR TERMS

The second difference between linear regression and the logistic regression model is in the error term.

The second difference between linear regression and the logistic regression model is in the error term.

One of the key assumptions of linear regression is that the error terms follow independent Gaussian distributions with zero mean and constant variance:

$$\epsilon \sim N(0, \sigma^2)$$

In logistic regression, the outcome variable can take only two values: 0 or 1.

In logistic regression, the outcome variable can take only two values: 0 or 1.

It's easy to show from this that instead of following a Gaussian distribution, the error term in logistic regression follows a Bernoulli distribution:

$$\epsilon \sim B(0, \pi(1-\pi))$$

In logistic regression, the outcome variable can take only two values: 0 or 1.

It's easy to show from this that instead of following a Gaussian distribution, the error term in logistic regression follows a Bernoulli distribution:

$$\epsilon \sim B(0, \pi(1-\pi))$$

NOTE

This is the same distribution followed by a coin toss.

Think about why this makes sense!

These two key differences define the logistic regression model, and they also lead us to a kind of unification of regression techniques called **generalized linear models**.

These two key differences define the logistic regression model, and they also lead us to a kind of unification of regression techniques called **generalized linear models**.

Briefly, **GLMs generalize the distribution of the error term, and allow the conditional mean of the response variable to be related to the linear model by a link function.**

In the present case, the error term follows a Bernoulli distribution, and the logit is the link function that connects us to the linear predictor.

In the present case, the error term follows a Bernoulli distribution, and the logit is the link function that connects us to the linear predictor.

$$g(x) = ln(\frac{\pi(x)}{1-\pi(x)}) = \alpha + \beta x$$

In the present case, the error term follows a Bernoulli distribution, and the logit is the link function that connects us to the linear predictor.

$$g(x) = ln(\frac{\pi(x)}{1-\pi(x)}) = \alpha + \beta x$$

Since the Bernoulli distribution and the logit function share a common parameter π, we say that the logit is the canonical link function for the Bernoulli distribution.

In the present case, the error term follows a Bernoulli distribution, and the logit is the link function that connects us to the linear predictor.

$$g(x) = ln(\frac{\pi(x)}{1-\pi(x)}) = \alpha + \beta x$$

NOTE

This terminology is just FYI!

Since the Bernoulli distribution and the logit function share a common parameter π, we say that the logit is the canonical link function for the Bernoulli distribution.

VI. INTERPRETING RESULTS

In linear regression, the parameter β represents the change in the response variable for a unit change in the covariate.

In linear regression, the parameter β represents the change in the response variable for a unit change in the covariate.

In logistic regression, β represents the change in the logit function for a unit change in the covariate.

In linear regression, the parameter β represents the change in the response variable for a unit change in the covariate.

In logistic regression, β represents the change in the logit function for a unit change in the covariate.

Interpreting this change in the logit function requires another definition first.

The odds of an event are given by the ratio of the probability of the event by its complement:

$$O(x=1) = \frac{\pi(1)}{(1-\pi(1))}$$

The odds of an event are given by the ratio of the probability of the event by its complement:

$$O(x=1) = \frac{\pi(1)}{(1-\pi(1))}$$

The odds ratio of a binary event is given by the odds of the event divided by the odds of its complement:

$$OR = \frac{O(x=1)}{O(x=0)} = \frac{\pi(1)/[1-\pi(1)]}{\pi(0)/[1-\pi(0)]}$$

Substituting the definition of $\pi(x)$ into this equation yields (after some algebra),

$$OR = e^{\beta}$$

Substituting the definition of $\pi(x)$ into this equation yields (after some algebra),

$$OR = e^{\beta}$$

This simple relationship between the odds ratio and the parameter β is what makes logistic regression such a powerful tool.

Q: So how do we interpret this?

Q: So how do we interpret this?

A: The odds ratio of a binary event gives the increase in likelihood of an outcome if the event occurs.

Suppose we are interested in mobile purchase behavior. Let y be a class label denoting purchase/no purchase, and let x denote a mobile OS (for example, iOS).

Suppose we are interested in mobile purchase behavior. Let y be a class label denoting purchase/no purchase, and let x denote a mobile OS (for example, iOS).

In this case, an odds ratio of 2 (eg, $\beta = \log(2)$) indicates that a purchase is twice as likely for an iOS user as for a non-iOS user.

EX: LOGISTIC REGRESSION