LECTURE 21

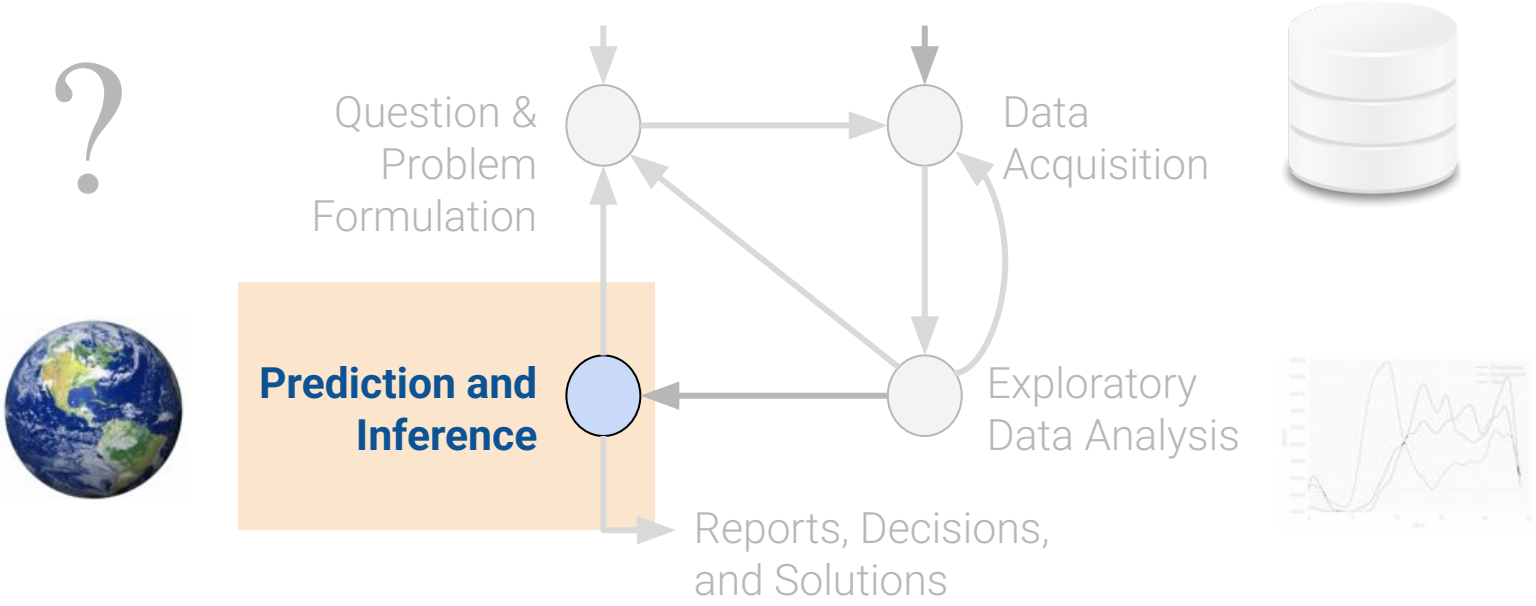
Logistic Regression I

Moving from regression to classification.

Data 100/Data 200, Spring 2022 @ UC Berkeley

Josh Hug and Lisa Yan

A Different Modeling Problem



(today)

Logistic Regression I:
The Model
Cross-Entropy Loss
The Probabilistic View

Logistic Regression II:
Classification Thresholds
Accuracy, Precision, Recall
Linear Separability

Today's Roadmap

Lecture 21, Data 100 Spring 2022

Regression vs. Classification

Intuition: The Coin Flip

Deriving the Logistic Regression Model

- Graph of Averages
- The Sigmoid (Logistic) Function

The Logistic Regression Model

- Comparison to Linear Regression

Parameter Estimation

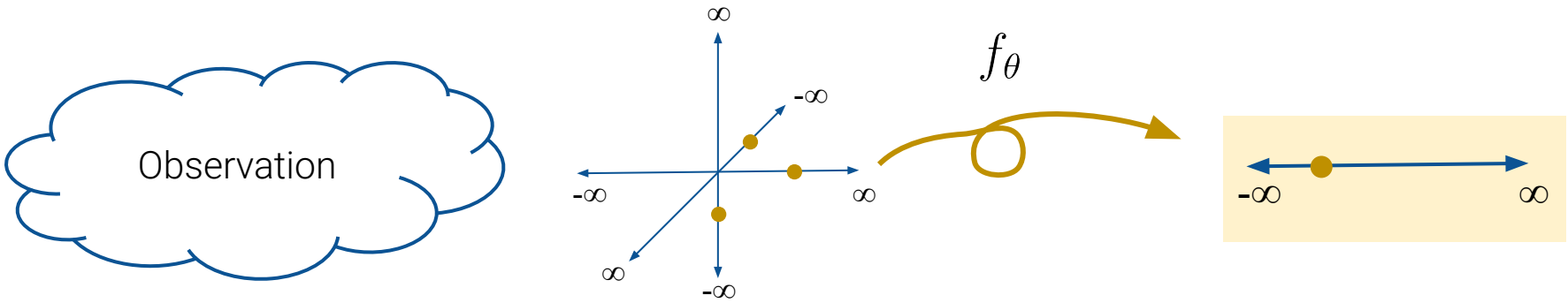
- Pitfalls of Squared Loss
- Cross-Entropy Loss
- Maximum Likelihood Estimation

Regression vs. Classification

Lecture 21, Data 100 Spring 2022

So far: Regression

The **parametric model** $\hat{y} = f_\theta(x)$, uses a feature x to predict a response $\hat{y}$ (true response y).



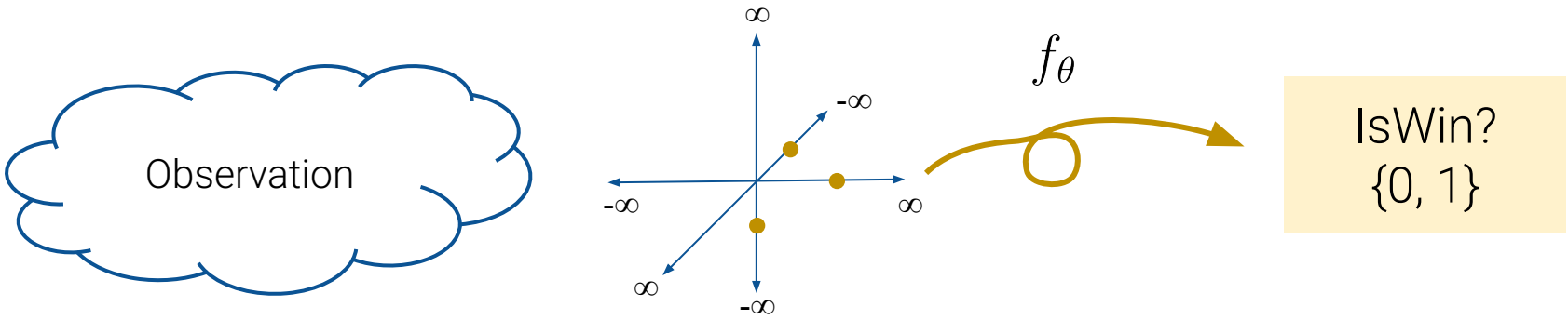
Use training data to estimate optimal $\hat{\theta}$:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \underbrace{(y_i - f_\theta(x_i))^2}_{\text{Squared loss}} + \underbrace{\lambda \mathrm{Reg}(\theta)}_{\text{Regularization}}$$

Regression:
Predict a **real** number y.

Today: Classification

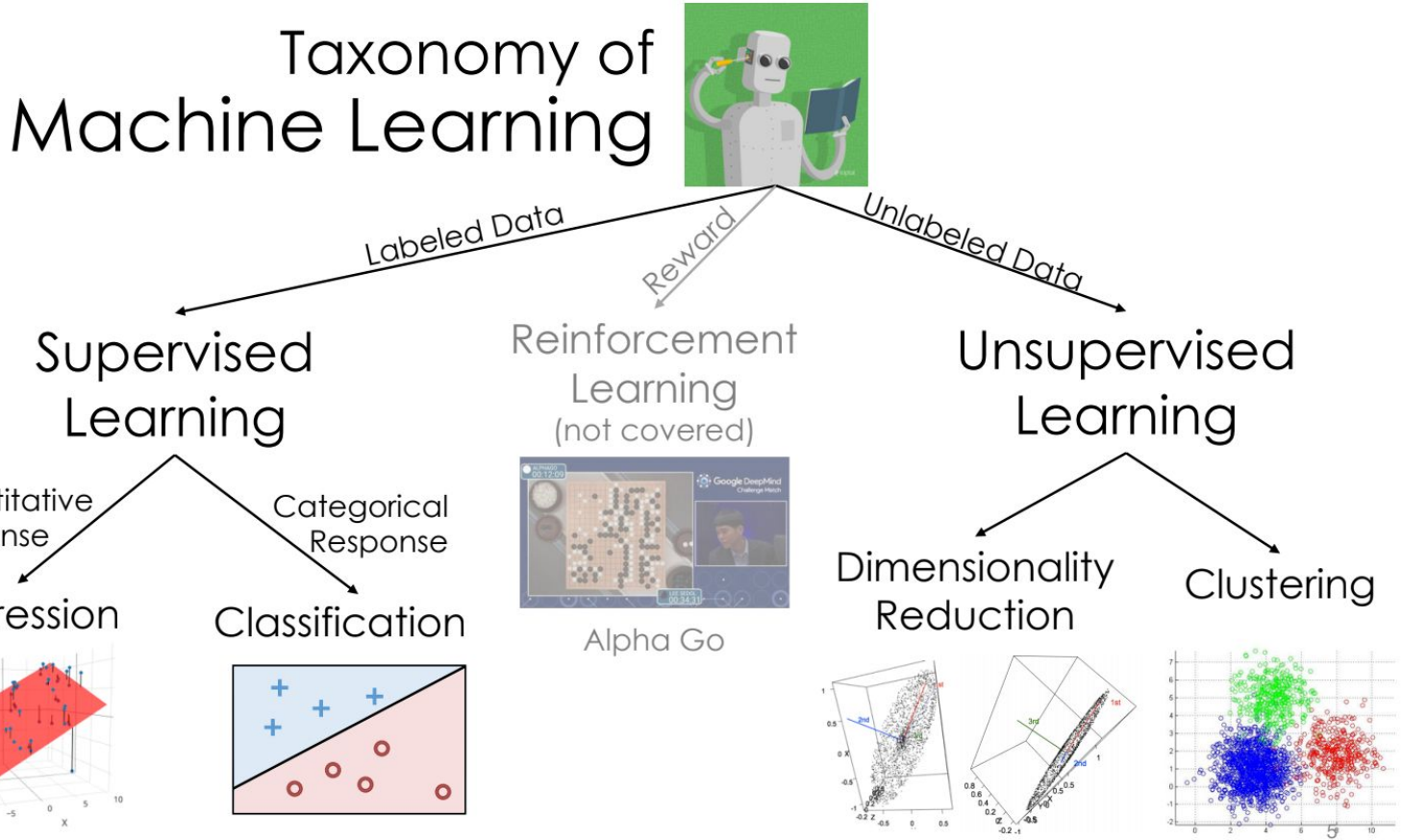
The **parametric model** $\hat{y} = f_\theta(x)$, uses a feature x to predict a response $\hat{y}$ (true response y).



Classification:
Predict a **categorical** variable y.

Taxonomy of Machine Learning

Regression and classification are both forms of **supervised learning**.



Kinds of Classification

We are interested in predicting some **categorical variable**, or **response**, y.

Binary classification [today]

- Two classes
- **Responses** y are either 0 or 1

win or lose

disease or no disease

spam or ham

Multiclass classification

- Many classes
- Examples: Image labeling (cat, dog, car), next word in a sentence, etc.

Structured prediction tasks

- Multiple related classification predictions
- Examples: Translation, voice recognition, etc.

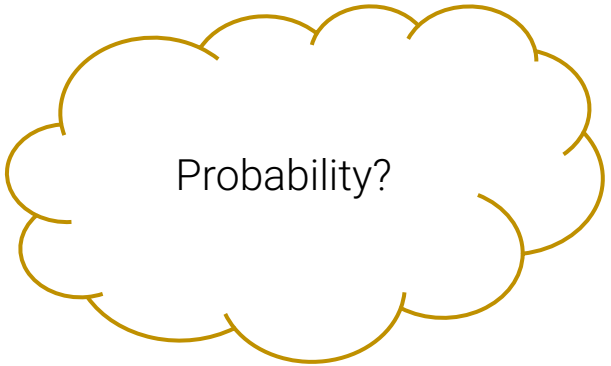
The Modeling Process

	Regression ($y \in \mathbb{R}$)	Classification ($y \in \{0, 1\}$)
1. Choose a model	Linear Regression $\hat{y} = f_\theta(x) = x^T\theta$	??
2. Choose a loss function	Squared Loss or Absolute Loss	??
3. Fit the model	Regularization Sklearn/Gradient descent	Regularization Sklearn/Gradient descent
4. Evaluate model performance	R^2, Residuals, etc.	?? (next time)

Intuition: The Coin Flip

Lecture 21, Data 100 Spring 2022

The Modeling Process

	Regression ($y \in \mathbb{R}$)	Classification ($y \in \{0, 1\}$)
1. Choose a model	Linear Regression $\hat{y} = f_\theta(x) = x^T\theta$	
2. Choose a loss function	Squared Loss or Absolute Loss	
3. Fit the model	Regularization Sklearn/Gradient descent	Regularization Sklearn/Gradient descent
4. Evaluate model performance	R^2, Residuals, etc.	?? (next time)

The No-Input Binary Classifier

Suppose you observed some outcomes of a coin (1 = Heads, 0 = Tails):

$$\{0, 0, 1, 1, 1, 1, 0, 0, 0, 0\}$$

Training data has only responses $\mathbb{Y}$ (no features $\mathbb{X}$)

For the next flip, do you predict heads or tails?

The No-Input Binary Classifier

Suppose you observed some outcomes of a coin (1 = Heads, 0 = Tails):

$$\{0, 0, 1, 1, 1, 1, 0, 0, 0, 0\}$$

Training data has only responses $\mathbb{Y}$ (no features $\mathbb{X}$)

For the next flip, do you predict heads or tails?

A reasonable model is to **assume all flips are IID** (i.e., same coin; same prob. of heads θ).

Parameter θ: Probability that flip == 1 (Heads)

Prediction: 1 or 0

1. Of the below, which is the best theta θ? Why?

A. 0.8 B. 0.5 C. 0.4
D. 0.2 E. Something else

2. For the next flip, would you predict 1 or 0?

The No-Input Binary Classifier

Suppose you observed some outcomes of a coin (1 = Heads, 0 = Tails):

$$\{0, 0, 1, 1, 1, 1, 0, 0, 0, 0\}$$

Training data has only responses $\mathbb{Y}$ (no features $\mathbb{X}$)

For the next flip, do you predict heads or tails?

A reasonable model is to **assume all flips are IID** (i.e., same coin; same prob. of heads θ).

Parameter θ:
Probability that flip == 1 (Heads)

Prediction:
1 or 0

1. Of the below, which is the best theta θ? Why?

A. 0.8 **B.** 0.5 **C.** 0.4
D. 0.2 **E.** Something else

2. For the next flip, would you predict 1 or 0? (next lecture)

The No-Input Binary Classifier

Suppose you observed some outcomes of a coin (1 = Heads, 0 = Tails):

$$\{0, 0, 1, 1, 1, 1, 0, 0, 0, 0\}$$

Training data has only responses $\mathbb{Y}$ (no features $\mathbb{X}$)

For the next flip, do you predict heads or tails?

A reasonable model is to **assume all flips are IID** (i.e., same coin; same prob. of heads θ).

Parameter θ:
Probability that flip == 1 (Heads)

Prediction:
1 or 0

1. Of the below, which is the best theta θ? Why?

A. 0.8 **B.** 0.5 **C.** 0.4
D. 0.2 **E.** Something else

0.4 is the most "intuitive" for two reasons:
1. Frequency of heads in our data
2. Maximizes the **likelihood** of our data

Likelihood of Data; Definition of Probability

$$R(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sigma(x_i^T\theta)\right)^2$$

A Bernoulli random variable Y with parameter p has distribution: $P(Y = y) = \begin{cases} p & \text{if } y = 1 \\ 1 - p & \text{if } y = 0 \end{cases}$

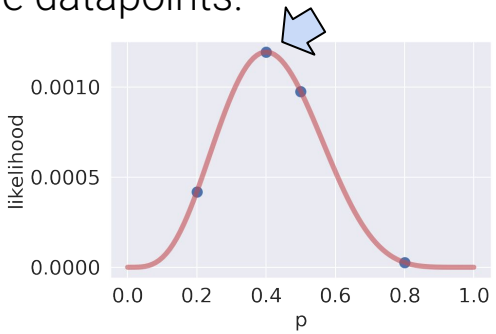
Given that all flips are IID from the same coin (probability of heads = p), the **likelihood** of our data is **proportional to** the probability of observing the datapoints.

(Corrected post-lecture 4/12)

Training data: [0, 0, 1, 1, 1, 1, 0, 0, 0, 0]

Probability: $\binom{10}{4} p^4(1-p)^6$

Data likelihood: $p^4(1-p)^6$.



This lecture, we will use two definitions of probability:

1. A number between 0 and 1.
2. Ratio/Frequency of a random event.

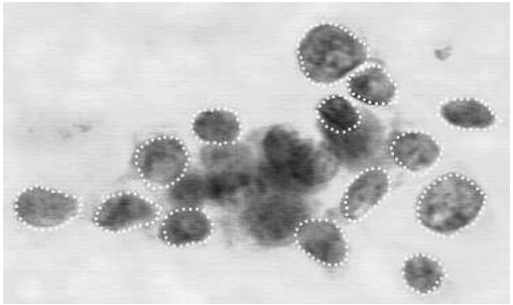
Deriving the Logistic Regression Model

Lecture 21, Data 100 Spring 2022

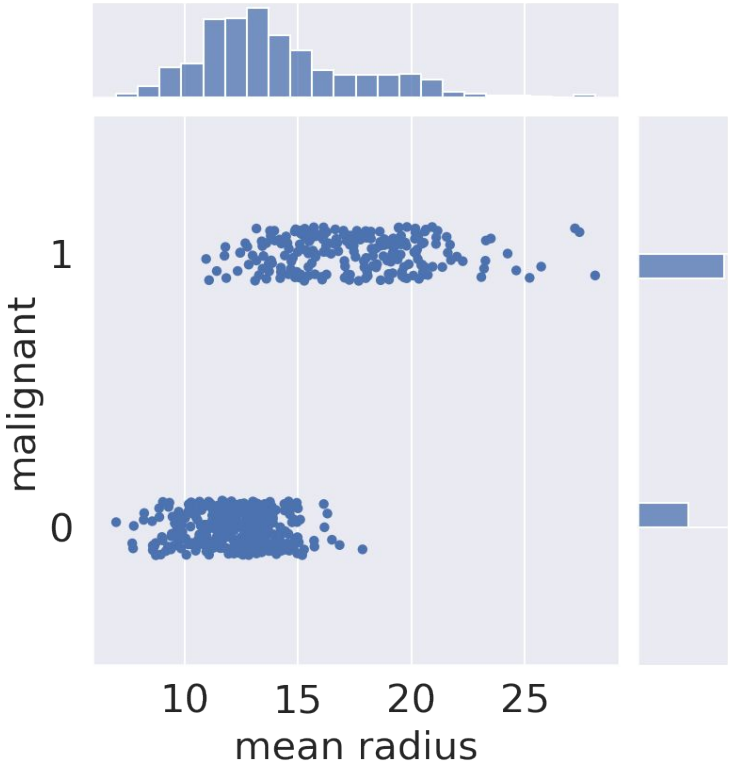
The Breast Cancer Wisconsin Dataset

Input x: mean radius of breast tumor cells
Response y: 1 if malignant, 0 if benign

mean radius	malignant
17.99	1
20.57	1
19.69	1
11.42	1
20.29	1
...	...
21.56	1
20.13	1
16.60	1
20.60	1
7.76	0



512 training observations, 57 test



Given the (single) **input feature**,
how can we predict the **label class**?

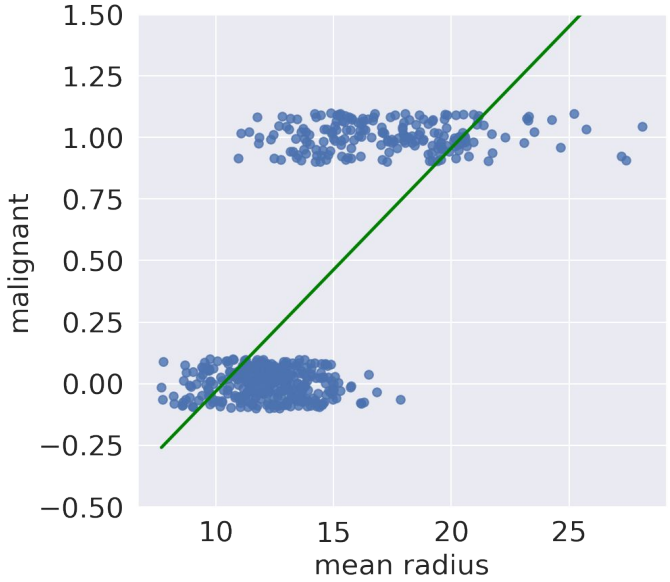
Classification labels are jittered to avoid overplotting.

Why Not Use Least Squares Linear Regression?

> I suppose it is tempting, if the only tool you have is a hammer, to treat everything as if it were a nail.
>
> – Abraham Maslow, *The Psychology of Science*

This is a valid model…

- It assumes y is continuous.
- It is the line that minimizes MSE for the training data.



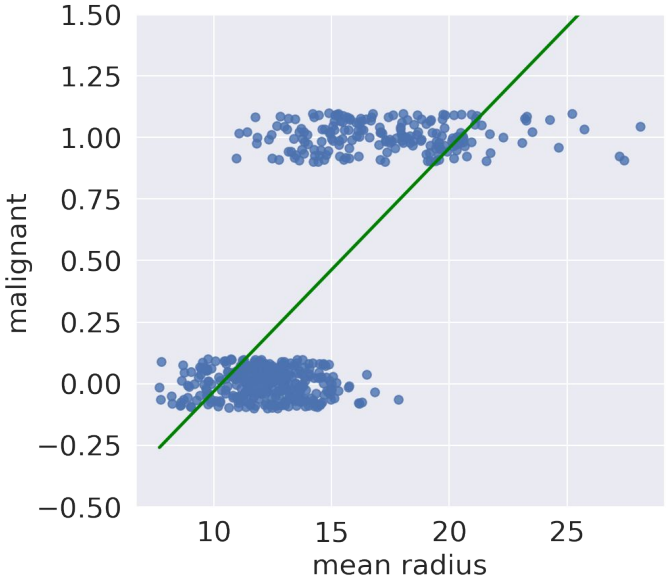
Demo

Why Not Use Least Squares Linear Regression?

> I suppose it is tempting, if the only tool you have is a hammer, to treat everything as if it were a nail.
>
> – Abraham Maslow, *The Psychology of Science*

This is a valid model…

- It assumes y is continuous.
- It is the line that minimizes MSE for the training data.



…but not a *good* model:

- The output $\hat{y}$ can be outside of the label range {0, 1}.
- What does a response of value of -2 mean?

Possible classification: assign $\hat{y} > 0.5$ to 1, and 0 otherwise.

- Boundary very sensitive to outliers.

Demo

Why Not Use Least Squares Linear Regression?

> I suppose it is tempting, if the only tool you have is a hammer, to treat everything as if it were a nail.
>
> – Abraham Maslow, *The Psychology of Science*

This is a valid model…

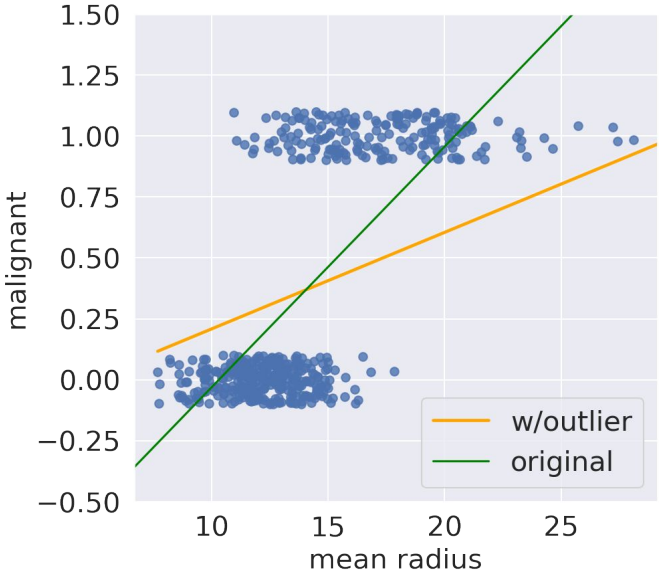
- It assumes y is continuous.
- It is the line that minimizes MSE for the training data.

…but not a *good* model:

- The output $\hat{y}$ can be outside of the label range {0, 1}.
- What does a response of value of -2 mean?

Possible classification: assign $\hat{y} > 0.5$ to 1 and 0 otherwise.

- Boundary very sensitive to outliers.



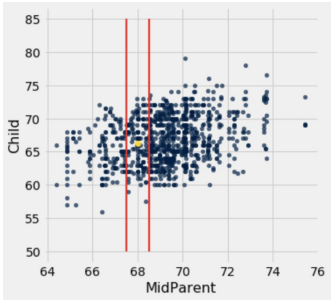
Demo

[Data 8] Graph of Averages: Regression

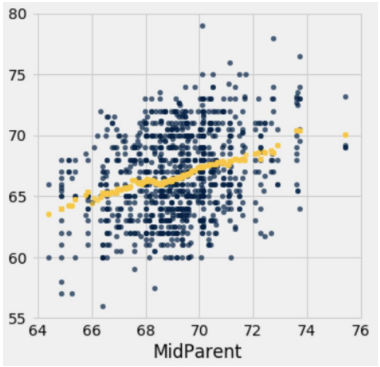
"Regression to the Mean": Linear Regression intuition comes from the **graph of averages**:

> For an input x, compute the **average value of y for all nearby x**, and predict that.

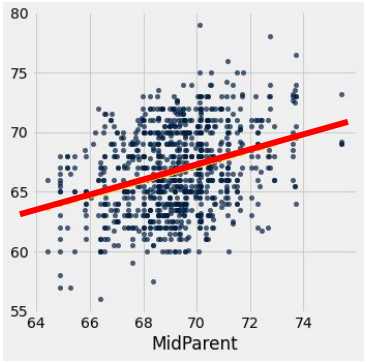
[Data 8 textbook]



Bucket x-axis into bins



Average y vs. x bins is approximately linear



Parametric linear model

$$\hat{y} = x^T\theta$$

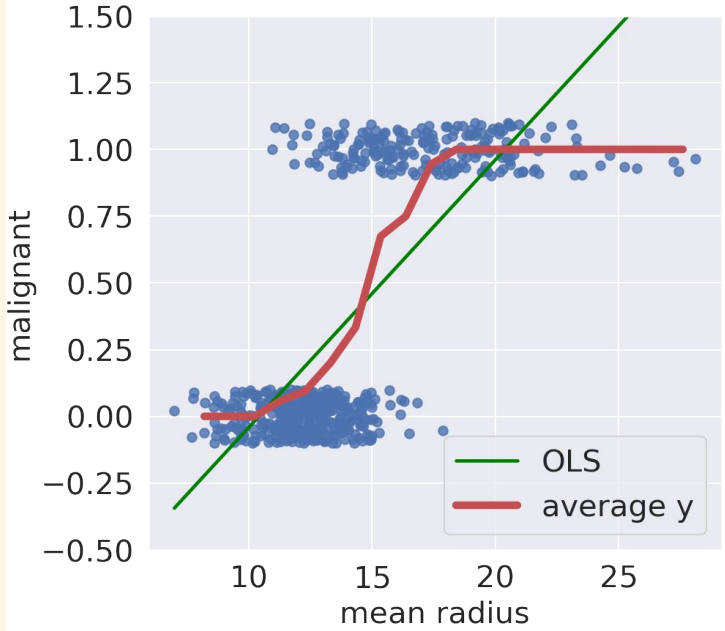
What does the graph of averages look like for binary classification?

Demo

Graph of Averages: Classification

> For an input x, compute the **average value of y for all nearby x**, and predict that.

[Data 8 textbook]



Our modeling goal is to fit this "S" curve as best as possible.

This fixed-width binning model degenerates with high-dim features (exponentially expensive; many bins with 0 points).

k-Nearest Neighbors (Data 8): non-parametric model. Not bad, but hard to interpret & slow..

Logistic Regression (today) Is a **parametric model** that also models this curve.

Demo

Transforming the Graph of Averages

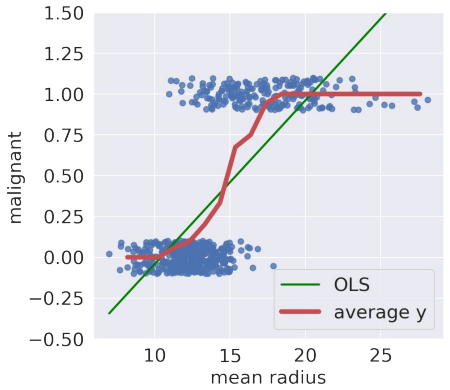
Can we transform the y-axis to fit a linear model to x?

Yes, but let's make one observation first.
Suppose we have a bin:

$$[0, 0, 1, 1, 1, 1, 0, 0, 0, 0] \rightarrow 0.4$$

The average y value is:

- A **number between 0 and 1**.
- The **ratio/frequency of 1**'s in the bin.



The average y for a bin is therefore a **probability**!!

$$\underbrace{P(Y = 1|x)}_{\text{“Given a particular x, the likelihood that the true response y is 1.”}} = \frac{\#\ (\text{y} == 1)\ \text{in bin}}{\#\ \text{datapoints in bin}}$$

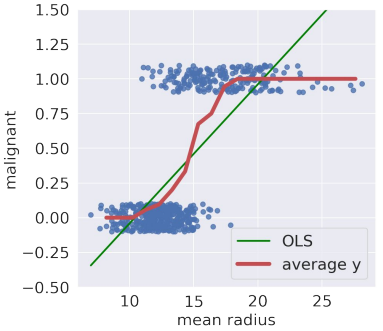
Logistic regression models this probability.

Demo

Probability as a Linear Function?

Modeling takeaways so far:

- Fit this “S” curve as best as possible.
- The curve models probability: P(Y = 1 | x).




```
from sklearn.linear_model import LogisticRegression
```

Logistic Regression is what we call a **generalized linear model**.

- Non-linear transformation of a linear model.
- So parameters are still a linear combination of x: $x^T\theta$

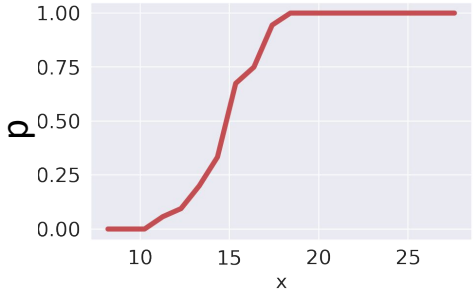
In the next few slides we’ll derive the logistic regression model:

1. Transform probability non-linearly until the curve looks linear.
2. Then, use algebra to invert all transformations.

Demo

[1/3] Transform Non-Linearly Until the Curve Looks Linear

The curve models the probability $P(Y = 1 \mid x)$.



Range of p: [0, 1]

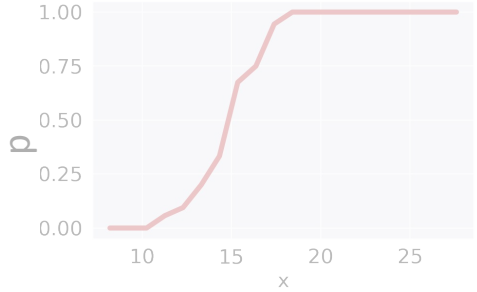
Curve: Not linear. S-shaped.

Demo

[2/3] Transform Non-Linearly Until the Curve Looks Linear

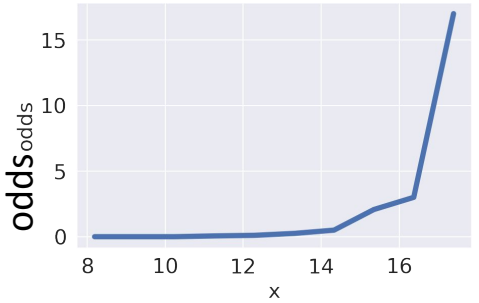
Odds is defined as the ratio of the probabilities of happening vs. not happening.

- Odds 2:1 means probability 2/3.



If p is the probability of response 1, then

$$\text{odds}(p) = \frac{p}{1-p}$$



Range of odds: Non-negative numbers

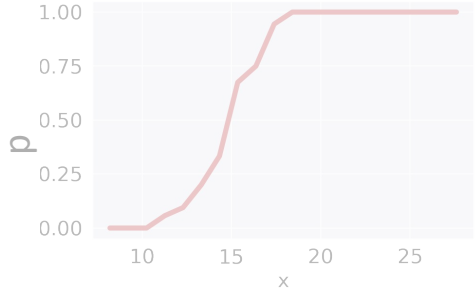
Curve: Looks exponential…!

Demo

[3/3] Transform Non-Linearly Until the Curve Looks Linear

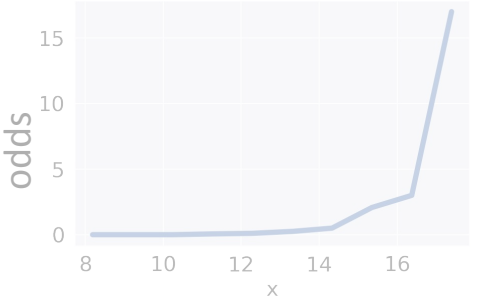
Log-Odds is, well, the (natural) log of odds.

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



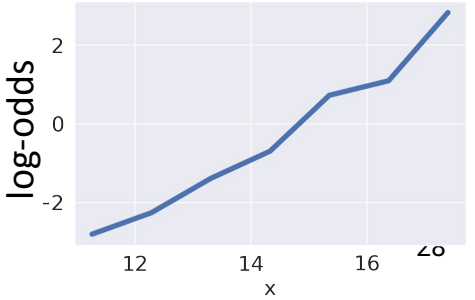
Range of log-odds: All reals

Curve: Looks linear!!



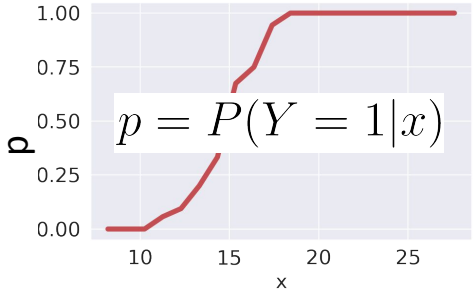
The logistic regression assumes that **Log-odds** is a **linear combination** of x and θ.

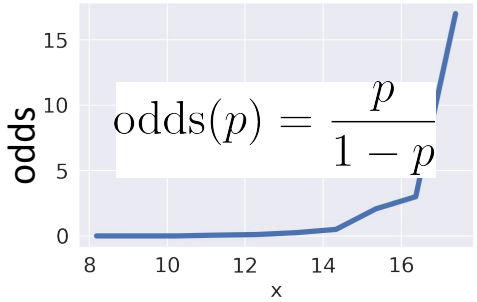
$$\log\left(\frac{p}{1-p}\right) = x^T\theta$$

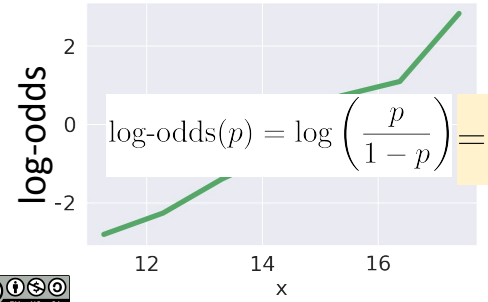


Demo

Use Algebra to Invert All Transformations







Solve for p:

$$\log\left(\frac{p}{1-p}\right) = x^T\theta$$

$$\frac{p}{1-p} = e^{x^T\theta}$$

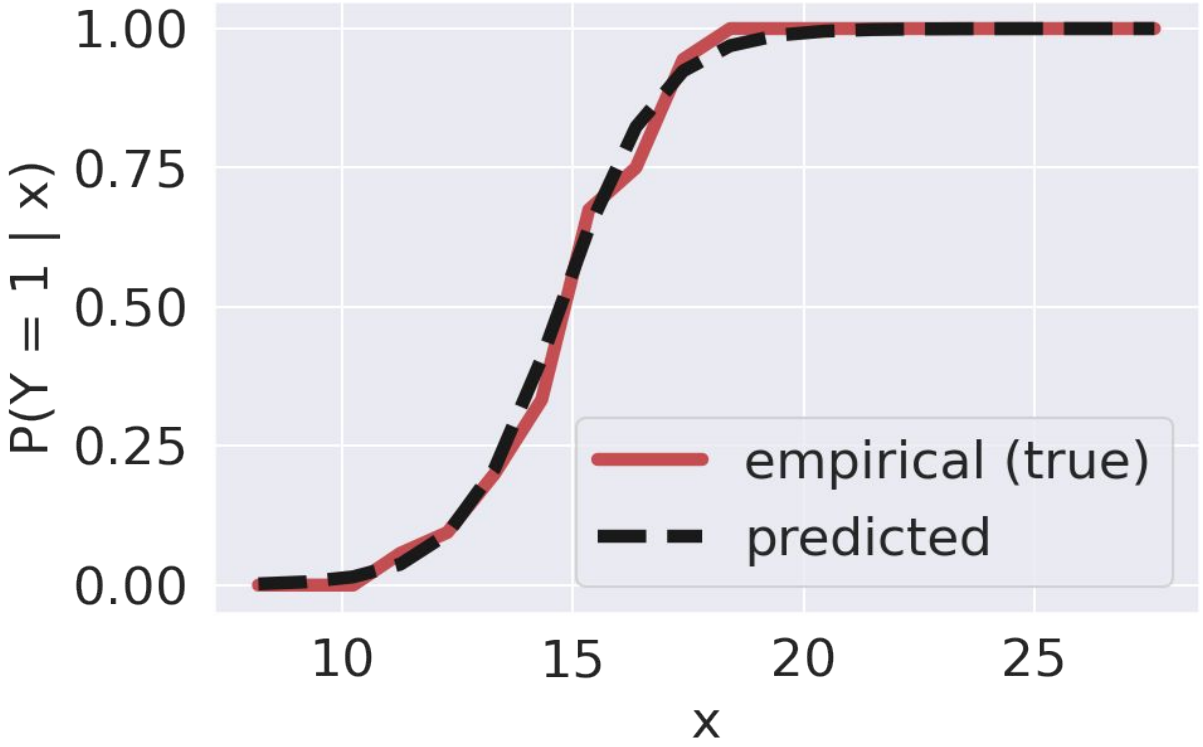
$$p = e^{x^T\theta} - pe^{x^T\theta}$$

$$p = \frac{e^{x^T\theta}}{1+e^{x^T\theta}}$$

$$p = \frac{1}{1+e^{-x^T\theta}}$$

This is called the **logistic function**, $\sigma(\ )$.

Arriving at the logistic regression model



$$\frac{\#\ (\text{y} == 1)\ \text{in bin}}{\#\ \text{datapoints in bin}}$$

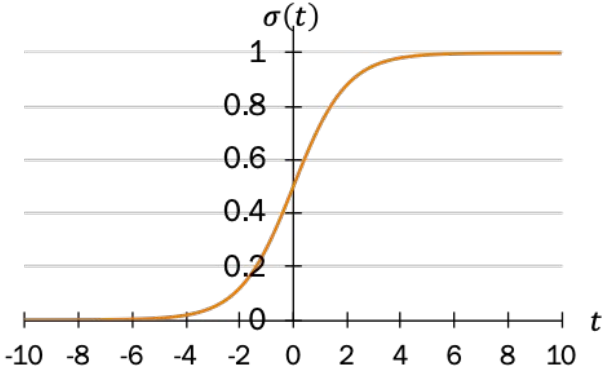
$$p = \frac{1}{1 + e^{-x^T\theta}}\ ,\ \theta^T = [\text{-13.8, 0.937}]$$

The Logistic Function (sometimes called “Sigmoid” function)

The **logistic function**:

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

This is a type of **sigmoid**, a class of functions that share certain properties.



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

Reflection/ Symmetry $1 - \sigma(t) = \frac{e^{-t}}{1 + e^{-t}} = \sigma(-t)$

Inverse $t = \sigma^{-1}(p) = \log\left(\frac{p}{1 - p}\right)$

(out of scope; see supplemental notebook)

Domain $-\infty < t < \infty$

Range $0 < \sigma(t) < 1$

Derivative

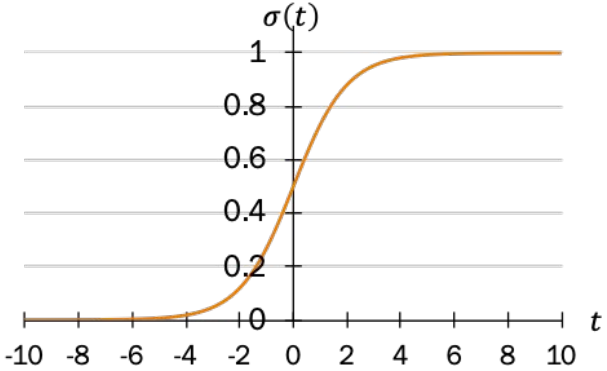
$$\frac{d}{dt}\sigma(t) = \sigma(t)(1 - \sigma(t)) = \sigma(t)\sigma(-t)$$

The Logistic Function (sometimes called “Sigmoid” function)

The **logistic function**:

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

This is a type of **sigmoid**, a class of functions that share certain properties.



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

Reflection/ Symmetry $1 - \sigma(t) = \frac{e^{-t}}{1 + e^{-t}} = \sigma(-t)$

Inverse $t = \sigma^{-1}(p) = \log\left(\frac{p}{1 - p}\right)$

(out of scope; see supplemental notebook)

Domain $-\infty < t < \infty$

Range $0 < \sigma(t) < 1$

Derivative

$$\frac{d}{dt}\sigma(t) = \sigma(t)(1 - \sigma(t)) = \sigma(t)\sigma(-t)$$

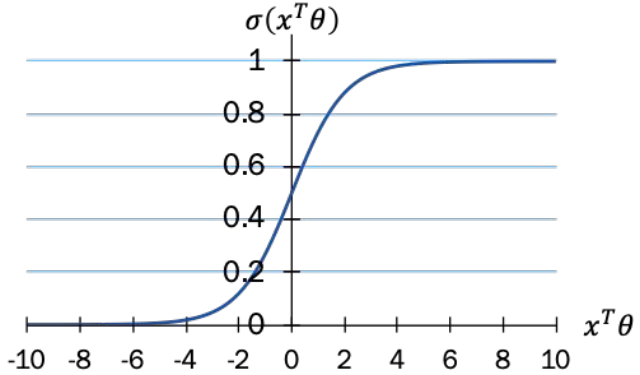
The logistic function smoothly squashes a real number to between 0 and 1.

From Feature to Probability

The **logistic function**:

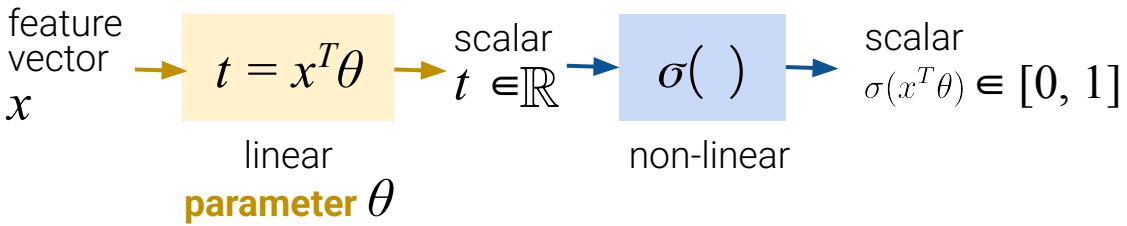
$$\sigma(t) = \frac{1}{1 + e^{-t}}$$

This is a type of **sigmoid**, a class of functions that share certain properties.



The logistic function smoothly squashes a real number to between 0 and 1.

Recall definition of **probability**:
A number between 0 and 1.



Arriving at the Logistic Regression Model

Our main takeaways of this section:

- Fit the “S” curve as best as possible.
- The curve models probability: P(Y = 1 | x).
- Assume log-odds is a linear combination of x and θ.

Putting it all together:

$$\underbrace{\hat{P}_\theta(Y = 1|x)}_{\text{Estimated probability that given the features x, the response is 1}} = \underbrace{\frac{1}{1 + e^{-x^T\theta}}}_{\text{Logistic function } \sigma(\ ) \text{ at the value } x^T\theta}$$

The logistic regression model is most commonly written as follows:

$$\hat{P}_\theta(Y = 1|x) = \sigma(x^T\theta)$$

Looks like linear regression. Now wrapped with $\boldsymbol{\sigma}(\ )$!

Comparison to Linear Regression

Lecture 21, Data 100 Spring 2022

Example calculation

Suppose I want to predict the probability that a tumor is malignant, given **mean radius** (first feature) and **mean smoothness** (second feature).

Suppose I fit a logistic regression model (with no intercept) using my training data, and somehow estimate the optimal parameters:

$$\hat{\theta}^T = [0.1 \qquad -0.5]$$

Now, you encounter a new breast tumor image:

$$x^T = [15 \qquad 1]$$

$$\hat{P}_\theta(Y = 1|x) = \sigma(x^T\theta)$$

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

1. What is the probability that the tumor is malignant (Y = 1)?
2. What would you predict as response? 1 or 0?

Example calculation

Suppose I want to predict the probability that a tumor is malignant, given **mean radius** (first feature) and **mean smoothness** (second feature).

Suppose I fit a logistic regression model (with no intercept) using my training data, and somehow estimate the optimal parameters:

$$\hat{\theta}^T = [0.1 \qquad -0.5]$$

Now, you encounter a new breast tumor image:

$$x^T = [15 \qquad 1]$$

$$\begin{aligned}\hat{P}_{\hat{\theta}}(Y = 1|x) &= \sigma(x^T\hat{\theta}) \\ &= \sigma(0.1 \cdot 15 + (-0.5) \cdot 1) \\ &= \sigma(1) \\ &= \frac{1}{1 + e^{-1}} \\ &\approx 0.7311\end{aligned}$$

Because the response is more likely to be 1 than 0, a reasonable prediction is

$$\hat{y} = 1$$

(more next lecture)

Linear Regression vs. Logistic Regression

Linear Regression model, parameter $\theta = (\theta_0, \theta_1, \ldots, \theta_p)$:

x → $f_\theta(x) = x^T\theta$ → $\hat{y}$ $\approx$ $y \in \mathbb{R}$

observation vector

prediction

quantitative response

Logistic Regression model, parameter $\theta = (\theta_0, \theta_1, \ldots, \theta_p)$:

x → $\hat{P}_\theta(Y = 1|x) = \sigma(x^T\theta)$ → Probability of response = 1

observation vector

classify (next lecture) → $\hat{y}$ $\approx$ $y \in \{0, 1\}$

prediction

categorical response

Logistic "Regression" Is Misleading

Linear Regression model, parameter $\theta = (\theta_0, \theta_1, \ldots, \theta_p)$:

x → $f_\theta(x) = x^T\theta$ → $\hat{y}$ $\approx$ $y \in \mathbb{R}$

observation vector

prediction

quantitative response

Logistic Regression model, parameter $\theta = (\theta_0, \theta_1, \ldots, \theta_p)$:

x → $\hat{P}_\theta(Y = 1|x) = \sigma(x^T\theta)$ → Probability of response = 1

observation vector

classify (next lecture) → $\hat{y}$ $\approx$ $y \in \{0, 1\}$

prediction

categorical response

Logistic Regression is used for **classification**. It has a confusing name!

Parameter Interpretation

Logistic Regression Model assumptions:

- Fit the "S" curve as best as possible.
- The curve models probability: P(Y = 1 | x).
- Assume log-odds is a linear combination of x and θ.

$$\hat{P}_\theta(Y = 1|x) = \sigma(x^T\theta)$$

$$\log\left(\frac{p}{1-p}\right) = x^T\theta$$

Because we are dealing with binary classification,

$$P(Y = 1|x) + P(Y = 0|x) = 1$$

$$\frac{P(Y = 1|x)}{P(Y = 0|x)} = e^{x^T\theta}$$

Parameter interpretation

Let's suppose our linear component has just a single feature, along with an intercept term.

$$\frac{P(Y=1|x)}{P(Y=0|x)} = e^{\theta_0+\theta_1 x}$$

What happens if you increase x by one unit?

- Odds is multiplied by e^{θ_1} .
- If $\theta_1 > 0$, the odds increase.
- If $\theta_1 < 0$, the odds decrease.

The odds ratio can be interpreted as the "number of successes for each failure."

What happens if $x^T\theta = \theta_0 + \theta_1 x = 0$?

- This means class 1 and class 0 are equally likely.
- $$e^0 = 1 \implies \frac{P(Y=1|x)}{P(Y=0|x)} = 1 \implies P(Y=1|x) = P(Y=0|x)$$

Interlude

Break (2 min)

Pitfalls of Squared Loss

Lecture 21, Data 100 Spring 2022

The Modeling Process

	Regression ($y \in \mathbb{R}$)	Classification ($y \in \{0, 1\}$)
1. Choose a model ✅	Linear Regression $\hat{y} = f_\theta(x) = x^T\theta$	Logistic Regression $\hat{P}_\theta(Y = 1\|x) = \sigma(x^T\theta)$
2. Choose a loss function	Squared Loss or Absolute Loss	??
3. Fit the model	Regularization Sklearn/Gradient descent	Regularization Sklearn/Gradient descent
4. Evaluate model performance	R^2, Residuals, etc.	?? (next time)

The Modeling Process

	Regression ($y \in \mathbb{R}$)	Classification ($y \in \{0, 1\}$)
1. Choose a model ✅	Linear Regression $\hat{y} = f_\theta(x) = x^T\theta$	Logistic Regression $\hat{P}_\theta(Y = 1\|x) = \sigma(x^T\theta)$
2. Choose a loss function	Squared Loss or Absolute Loss	Can squared loss still work?
3. Fit the model	Regularization Sklearn/Gradient descent	Regularization Sklearn/Gradient descent
4. Evaluate model performance	R^2, Residuals, etc.	?? (next time)

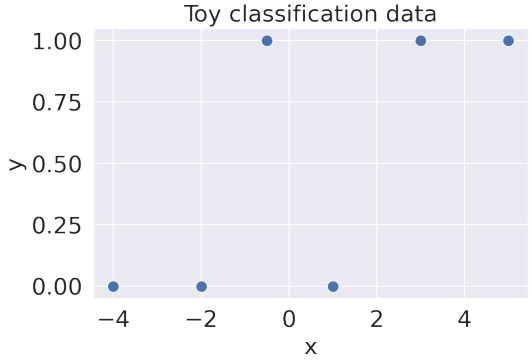
Toy Dataset: L2 Loss

Logistic Regression model:

$$\hat{P}_\theta(Y = 1|x) = \sigma(x^T\theta)$$

Assume no intercept.
So x, θ both scalars.

	x	y
0	-4.0	0
1	-2.0	0
2	-0.5	1
3	1.0	0
4	3.0	1
5	5.0	1

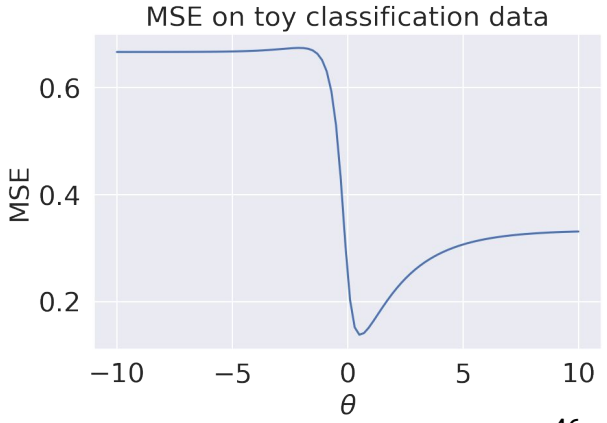


Mean Squared Error:

$$R(\theta) = +\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sigma(x_i^T\theta)\right)^2$$

Corrected (lecture 4/12)

The MSE loss surface for logistic regression has many issues!



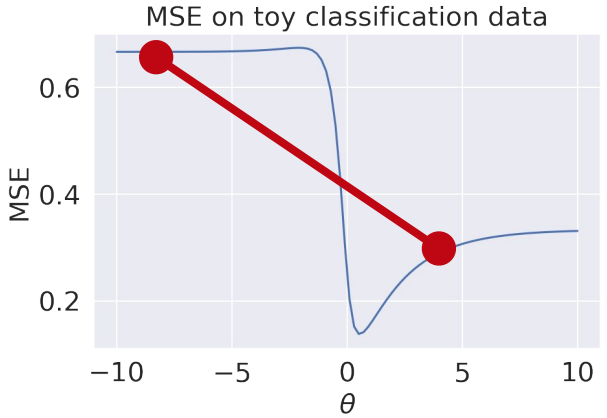
Demo

3 Pitfalls of Squared Loss

$$R(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sigma(x_i^T\theta)\right)^2$$

1. Non-convex. Gets stuck in local minima.

Secant line crosses function, so $R''(\theta)$ is not greater than 0 for all θ.



Demo

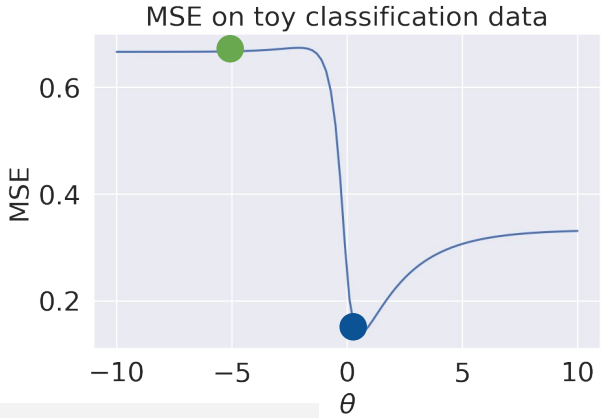
3 Pitfalls of Squared Loss

$$R(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sigma(x_i^T\theta)\right)^2$$

1. Non-convex. Gets stuck in local minima.

Secant line crosses function, so $R''(\theta)$ is not greater than 0 for all θ.

Gradient Descent: Different initial guesses will yield different optimal estimates.



Demo

```
from scipy.optimize import minimize

minimize(mse_loss_toy_nobias, x0 = 0)["x"][0]
```

```
0.5446601825581691
```

```
minimize(mse_loss_toy_nobias, x0 = -5)["x"][0]
```

```
-10.343653061026611
```

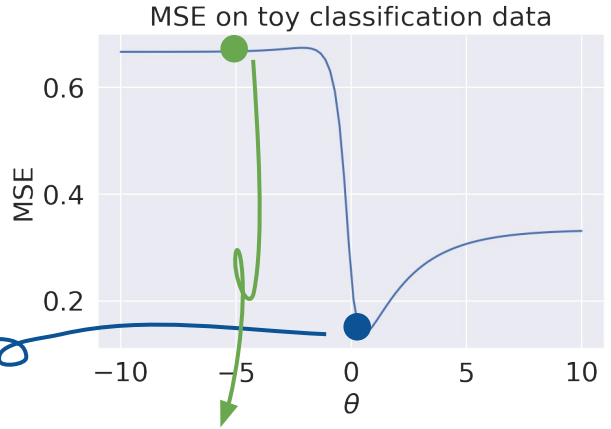
3 Pitfalls of Squared Loss

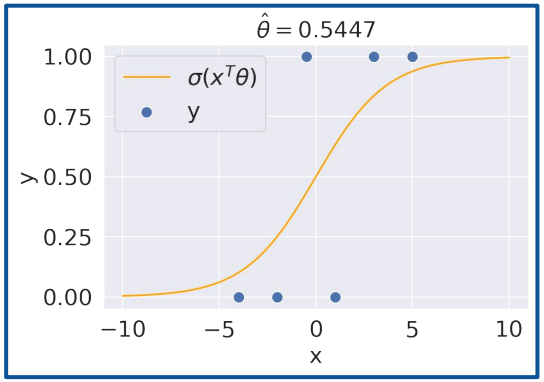
$$R(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sigma(x_i^T\theta)\right)^2$$

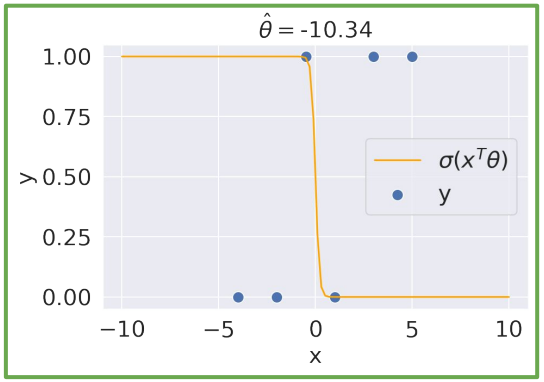
1. Non-convex. Gets stuck in local minima.

Secant line crosses function, so $R''(\theta)$ is not greater than 0 for all θ.

Gradient Descent: Different initial guesses will yield different optimal estimates.







Demo

3 Pitfalls of Squared Loss

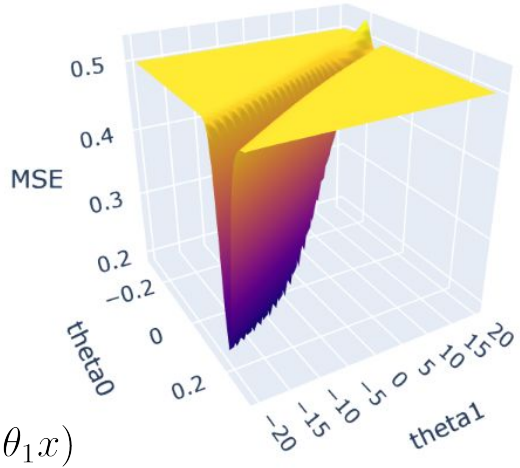
$$R(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sigma(x_i^T\theta)\right)^2$$

1. Non-convex. Gets stuck in local minima.

Secant line crosses function, so $R''(\theta)$ is not greater than 0 for all θ.

Gradient Descent: Different initial guesses will yield different optimal estimates.

Adding features won't fix this non-convexity issue.



On right: MSE loss surface for $\sigma(\theta_0 + \theta_1 x)$

Gradient update step: $\theta^{(t+1)} = \theta^{(t)} - \alpha\nabla_\theta R(\theta, \mathbb{X}, \mathbb{Y})$

If our initial guess for $\hat{\theta}$ is in the flat area, then our gradient will be 0, and our updates will stop.

Demo

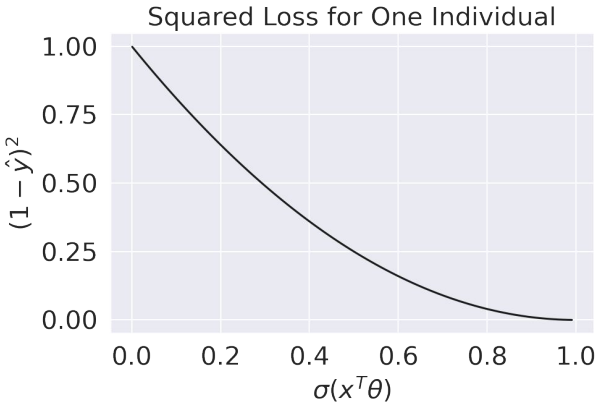
3 Pitfalls of Squared Loss

$$R(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sigma(x_i^T\theta)\right)^2$$

1. **Non-convex**. Gets stuck in local minima.

2. **Bounded**. Not a good measure of model error.

- We'd like loss functions to penalize "off" predictions.
- MSE never gets very large, because both response and predicted probability are bounded by 1.



Demo

3 Big Pitfalls of Squared Loss

$$R(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sigma(x_i^T\theta)\right)^2$$

1. **Non-convex**. Gets stuck in local minima.

2. **Bounded**. Not a good measure of model error.

3. **Conceptually questionable**.
   Tries to match probability to 0/1 class labels.

To thine own self be true, and it must follow, as the night the day, thou canst not then be false to any man.

Hamlet, Shakespeare [Wikipedia]

MSE + classification is *occasionally* used in some neural network applications.
But overall, avoid.

Demo

Cross-Entropy Loss

Lecture 21, Data 100 Spring 2022

Choosing a Different Loss Function

	Regression ($y \in \mathbb{R}$)	Classification ($y \in \{0, 1\}$)
1. Choose a model ✅	Linear Regression $\hat{y} = f_\theta(x) = x^T\theta$	Logistic Regression $\hat{P}_\theta(Y = 1\|x) = \sigma(x^T\theta)$
2. Choose a loss function	Squared Loss or Absolute Loss	**Cross-Entropy Loss**
3. Fit the model	Regularization Sklearn/Gradient descent	Regularization Sklearn/Gradient descent
4. Evaluate model performance	R^2, Residuals, etc.	?? (next time)

Cross-Entropy Loss

Let y be a binary label {0, 1}, and p be the probability of the label being 1.

The **cross-entropy loss** is defined as

$$-\left(y\log(p)+(1-y)\log(1-p)\right)$$

Matches the probabilistic modeling of logistic regression if $p=\hat{P}_\theta(Y=1|x)=\sigma(x^T\theta)$

Cross-Entropy loss addresses the 3 pitfalls of squared loss.

1. Convex. No local minima for logistic regression.
2. A good measure of model error. Strongly penalizes “off” predictions.
3. Conceptually sound.

For now, suspend your belief about #3 (what cross-entropy loss actually means).

We’ll focus on its mathematical properties (#1 and #2) first.

Cross-Entropy Loss Is Like, Two Loss Functions In One

Let y be a binary label {0, 1}, and p be the probability of the label being 1.

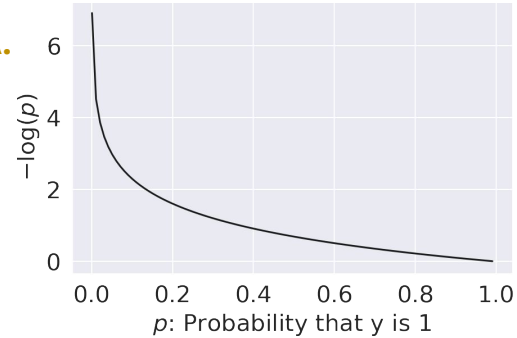
The **cross-entropy loss** is defined as

$$-(y\log(p)+(1-y)\log(1-p))$$

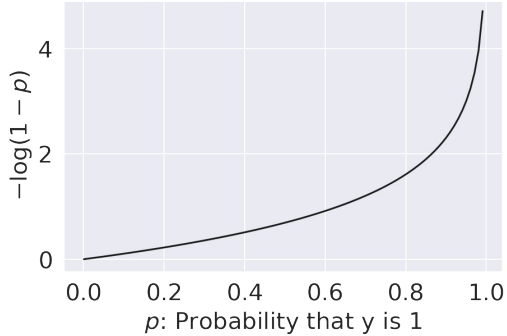
Suppose we can choose *p* to be any value between 0 and 1. Which plot best represents cross-entropy loss if:

1) The true response y = 1?
2) The true response y = 0?

A.



B.



C. Something else

Does cross-entropy loss strongly penalize “off” predictions?

Cross-Entropy Loss Is Like, Two Loss Functions In One

Let y be a binary label {0, 1}, and p be the probability of the label being 1.

The **cross-entropy loss** is defined as

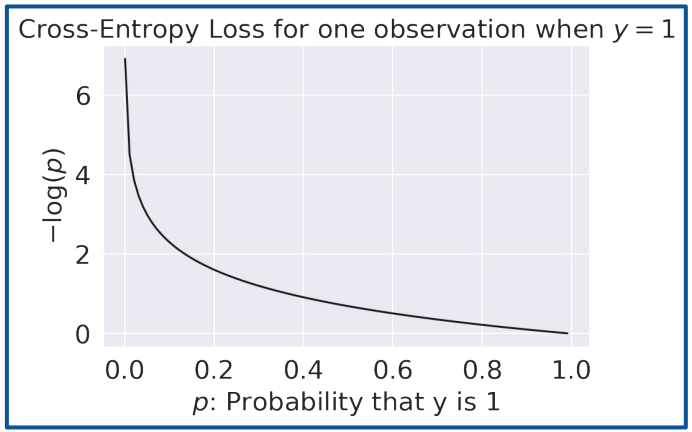
$$-(y\log(p) + (1-y)\log(1-p))$$

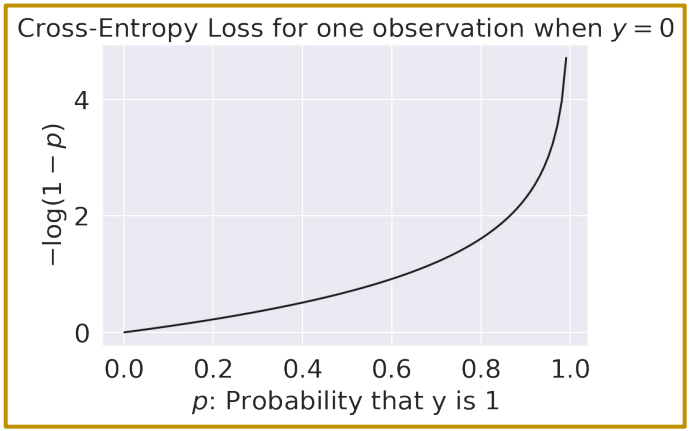
makes loss positive

for y = **1**, only this term stays

for y = **0**, only this term stays

Loss is **not random**. The i-th observed datapoint has a fixed input xi and a fixed response yi (which can never be both 0 and 1).



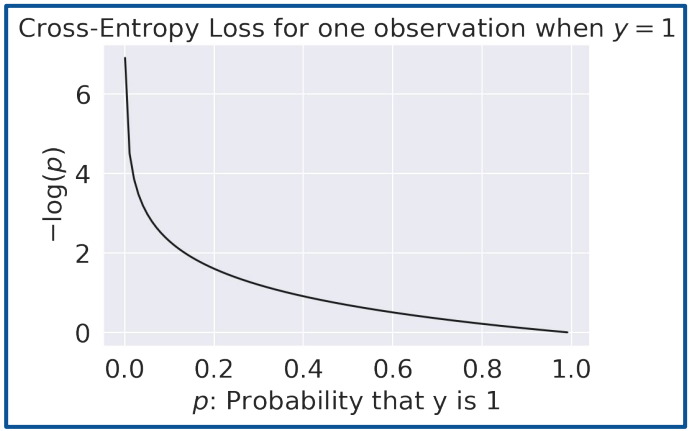


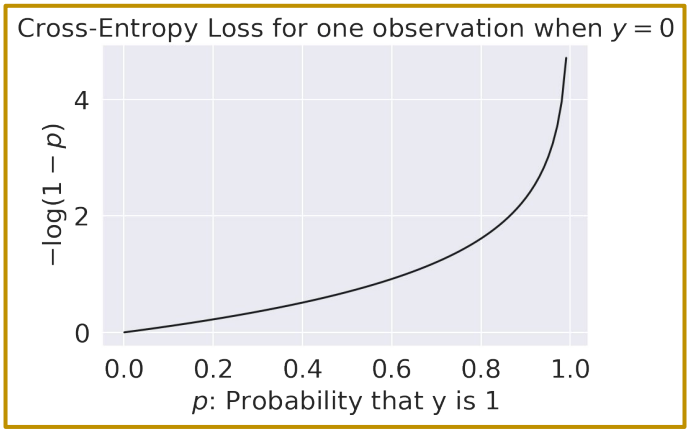
Cross-Entropy Loss Is a Good Measure of Model Error

Let y be a binary label {0, 1}, and p be the probability of the label being 1.

The **cross-entropy loss** is defined as

$$-\left(y\log(p) + (1-y)\log(1-p)\right)$$





For y = 1,

- p → 0: infinite loss
- p → 1: zero loss

For y = 0,

- p → 0: zero loss
- p → 1: infinite loss

Empirical Risk: Average Cross-Entropy Loss

For a single datapoint, the cross-entropy curve is convex. It has a global minimum.

$$-\left(y\log(p)+(1-y)\log(1-p)\right)$$

What about average cross-entropy loss, i.e., empirical risk?

For logistic regression, the empirical risk over a sample of size n is:

$$R(\theta) = -\frac{1}{n}\sum_{i=1}^{n}\left(y_i\log(p_i)+(1-y_i)\log(1-p_i)\right)$$

(y_i is i-th response, and p_i is prob. that i-th response is 1)

$$= -\frac{1}{n}\sum_{i=1}^{n}\left(y_i\log(\sigma(X_i^T\theta)+(1-y_i)\log(1-\sigma(X_i^T\theta))\right)$$

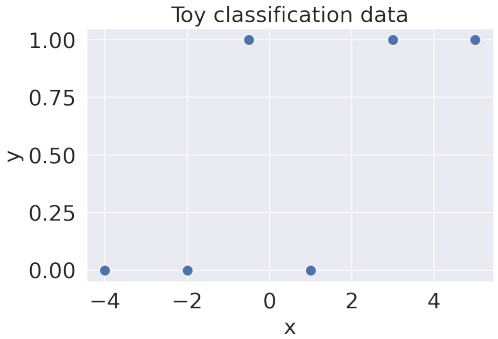
($p_i = \sigma(X_i^T\theta)$ and X_i is i-th feature vector)

The optimization problem is therefore to find the estimate $\hat{\theta}$ that minimizes R(θ):

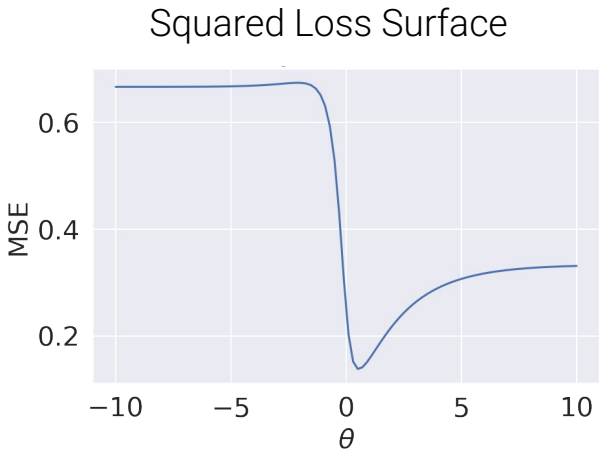
$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} -\frac{1}{n}\sum_{i=1}^{n}\left(y_i\log(\sigma(X_i^T\theta)+(1-y_i)\log(1-\sigma(X_i^T\theta))\right)$$

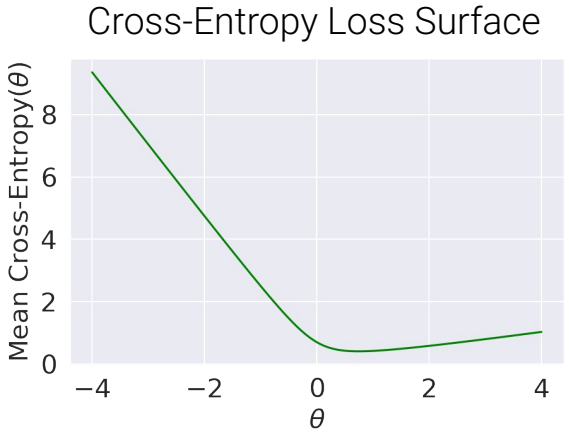
Convexity Proof By Picture

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} -\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log(\sigma(X_i^T\theta) + (1-y_i)\log(1-\sigma(X_i^T\theta))\right)$$

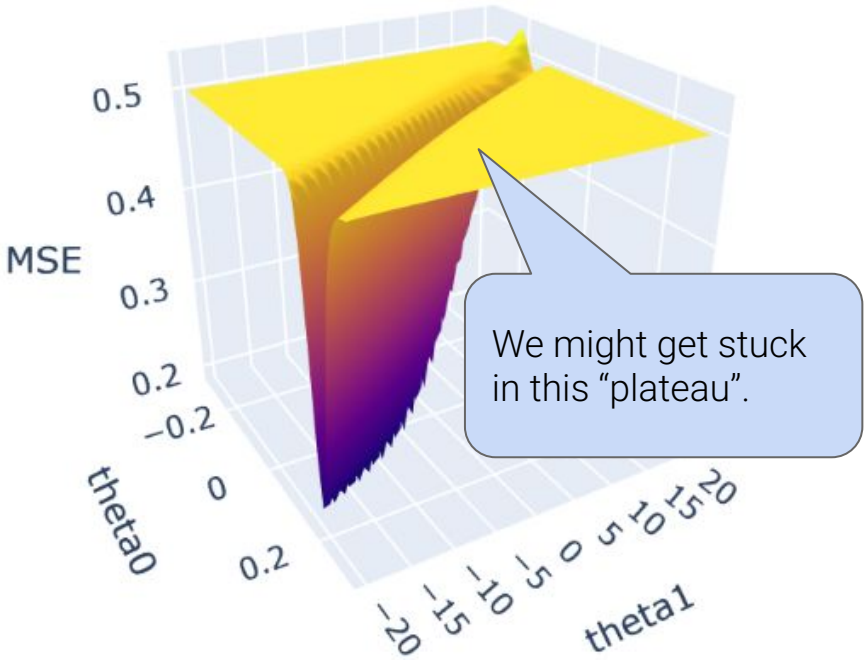


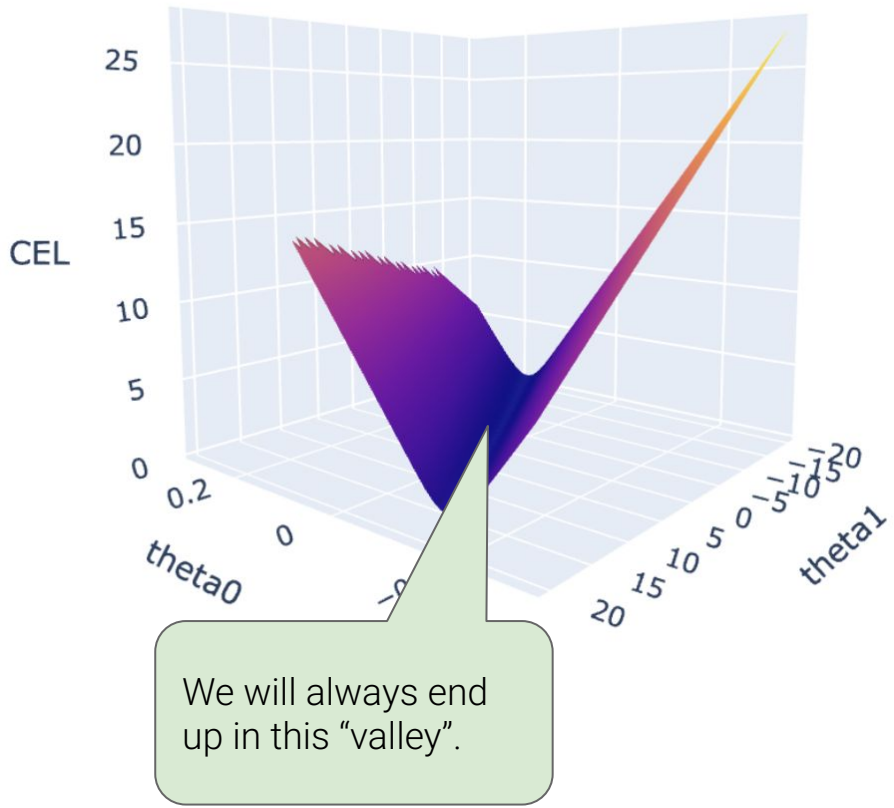
	x	y
0	-4.0	0
1	-2.0	0
2	-0.5	1
3	1.0	0
4	3.0	1
5	5.0	1





Convexity Proof By Picture (more features)





Review of this section

Let y be a binary label {0, 1}, and p be the probability of the label being 1.

The **cross-entropy loss** is defined as

$$-\left(y\log(p)+(1-y)\log(1-p)\right)$$

Matches the probabilistic modeling of logistic regression if $p=\hat{P}_\theta(Y=1|x)=\sigma(x^T\theta)$

Cross-Entropy loss addresses the 3 pitfalls of squared loss.

✅ Convex. No local minima for logistic regression.

✅ A good measure of model error. Strongly penalizes “off” predictions.

3. Conceptually sound.

Now let’s tackle #3.

Got to here live.

This section will be covered Thursday 4/14.

Maximum Likelihood Estimation

Lecture 21, Data 100 Spring 2022

Regression vs. Classification

Intuition: The Coin Flip

Deriving the Logistic Regression Model

- Graph of Averages
- The Sigmoid (Logistic) Function

The Logistic Regression Model

- Comparison to Linear Regression

Parameter Estimation

- Pitfalls of Squared Loss
- Cross-Entropy Loss
- **Maximum Likelihood Estimation**

The Modeling Process

	Regression ($y \in \mathbb{R}$)	Classification ($y \in \{0, 1\}$)
1. Choose a model ✅	Linear Regression $\hat{y} = f_\theta(x) = x^T\theta$	Logistic Regression $\hat{P}_\theta(Y = 1\|x) = \sigma(x^T\theta)$
2. Choose a loss function	Squared Loss or Absolute Loss	Average Cross-Entropy Loss $-\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log(\sigma(X_i^T\theta)) + (1 - y_i)\log(1 - \sigma(X_i^T\theta))\right)$
3. Fit the model	Regularization Sklearn/Gradient descent	Wherefore use cross-entropy?
4. Evaluate model performance	R^2, Residuals, etc.	?? (next time)

Shakespeare [Wikipedia]

Why Use Cross-Entropy Loss?

This section will not be directly tested, but you will understand why we minimize cross-entropy loss for logistic regression.

Two common explanations:

- [Information Theory] KL Divergence (textbook)
- [Probability] Maximum Likelihood Estimation (this lecture)

Recall the Coin Demo (No-Input Classification)

For training data: {0, 0, 1, 1, 1, 1, 0, 0, 0, 0}

0.4 is the most "intuitive" θ for two reasons:

1. Frequency of heads in our data
2. Maximizes the **likelihood** of our data:

$$\hat{\theta} = \underset{\theta}{\text{argmax}} \left(\theta^4 (1-\theta)^6\right)$$

Parameter θ:
Probability that
IID flip == 1 (Heads)

Prediction:
1 or 0

How can we generalize this notion of likelihood to **any** random binary sample?

$$\{y_1, y_2, \ldots, y_n\} \Rightarrow \hat{\theta} = \underset{\theta}{\text{argmax}} \left(???\right)$$

data (1's and 0's)

likelihood

A Compact Representation of the Bernoulli Probability Distribution

How can we generalize this notion of likelihood to **any** random binary sample?

$$\{y_1, y_2, \ldots, y_n\} \Rightarrow \hat{\theta} = \underset{\theta}{\operatorname{argmax}} \, (???)$$

data (1's and 0's) — likelihood

Let Y be Bernoulli(p). The probability distribution can be written compactly:

$$P(Y = y) = p^y (1 - p)^{1-y}$$

For P(Y = **1**), only this term stays

For P(Y = **0**), only this term stays

(long, non-compact form):

$$P(Y = y) = \begin{cases} p & \text{if } y = 1 \\ 1 - p & \text{if } y = 0 \end{cases}$$

Generalized Likelihood of Binary Data

How can we generalize this notion of likelihood to **any** random binary sample?

$$\{y_1, y_2, \ldots, y_n\} \Rightarrow \hat{\theta} = \underset{\theta}{\operatorname{argmax}} \, (???)$$

data (1's and 0's) — likelihood

Let Y be Bernoulli(p). The probability distribution can be written compactly:

$$P(Y = y) = p^y (1-p)^{1-y}$$

For P(Y = **1**), only this term stays

For P(Y = **0**), only this term stays

If binary data are **IID with same** probability p, then the likelihood of the data is:

$$\prod_{i=1}^{n} p^{y_i} (1-p)^{(1-y_i)}$$

Ex: {0, 0, 1, 1, 1, 1, 0, 0, 0, 0} → $p^4(1-p)^6$

Generalized Likelihood of Binary Data

How can we generalize this notion of likelihood to any random binary sample?

$$\{y_1, y_2, \ldots, y_n\} \Rightarrow \hat{\theta} = \underset{\theta}{\operatorname{argmax}}\,(???)$$

data (1's and 0's) — likelihood

Let Y be Bernoulli(p). The probability distribution can be written compactly:

$$P(Y = y) = p^y (1-p)^{1-y}$$

For P(Y = **1**), only this term stays

For P(Y = **0**), only this term stays

If binary data are **IID with same** probability p, then the likelihood of the data is:

$$\prod_{i=1}^{n} p^{y_i} (1-p)^{(1-y_i)}$$

If data are independent with **different** probability p_i, then the likelihood of the data is:

(spoiler: for logistic regression, $p_i = \sigma(X_i^T \theta)$)

$$\prod_{i=1}^{n} p_i^{y_i} (1-p_i)^{(1-y_i)}$$

Maximum Likelihood Estimation (MLE)

Our **maximum likelihood estimation** problem:

- For i = 1, 2, ..., n, let Y_i be independent Bernoulli(p_i). Observe data $\{y_1, y_2, \ldots, y_n\}$.
- We'd like to estimate $p_1, p_2, \ldots, p_n$.

Find $\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_n$ that **maximize** $\prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{(1-y_i)}$

Maximum Likelihood Estimation (MLE)

Our **maximum likelihood estimation** problem:

- For i = 1, 2, ..., n, let Y_i be independent Bernoulli(p_i). Observe data $\{y_1, y_2, \ldots, y_n\}$.
- We'd like to estimate $p_1, p_2, \ldots, p_n$.

Find $\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_n$ that **maximize** $\prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{(1-y_i)}$

Equivalent, simplifying optimization problems (since we need to take the first derivative):

$$\underset{p_1, p_2, \ldots, p_n}{\textbf{maximize}} \quad \log\left(\prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{(1-y_i)}\right)$$

(log is an increasing function. If a > b, then log(a) > log(b).)

$$= \sum_{i=1}^{n} \log\left(p_i^{y_i}(1-p_i)^{(1-y_i)}\right) = \sum_{i=1}^{n} \left(y_i \log(p_i) + (1-y_i)\log(1-p_i)\right)$$

Maximum Likelihood Estimation (MLE)

Our **maximum likelihood estimation** problem:

- For i = 1, 2, ..., n, let Y_i be independent Bernoulli(p_i). Observe data $\{y_1, y_2, \ldots, y_n\}$.
- We'd like to estimate $p_1, p_2, \ldots, p_n$.

Find $\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_n$ that **maximize** $\prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{(1-y_i)}$

Equivalent, simplifying optimization problems (since we need to take the first derivative):

maximize $p_1, p_2, \ldots, p_n$ $\log\left(\prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{(1-y_i)}\right)$ (log is an increasing function. If a > b, then log(a) > log(b).)

$$= \sum_{i=1}^{n} \log\left(p_i^{y_i}(1-p_i)^{(1-y_i)}\right) = \sum_{i=1}^{n} \left(y_i \log(p_i) + (1-y_i)\log(1-p_i)\right)$$

minimize $p_1, p_2, \ldots, p_n$ $-\frac{1}{n}\sum_{i=1}^{n} \left(y_i \log(p_i) + (1-y_i)\log(1-p_i)\right)$

Maximizing Likelihood == Minimizing Average Cross-Entropy

maximize $\underset{p_1, p_2, \ldots, p_n}{}$ $\prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{(1-y_i)}$

Log is increasing; max/min properties

minimize $\underset{p_1, p_2, \ldots, p_n}{}$ $-\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log(p_i) + (1-y_i)\log(1-p_i)\right)$

For logistic regression, let $p_i = \sigma(X_i^T\theta)$

minimize $\underset{\theta}{}$ $\underbrace{-\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log(\sigma(X_i^T\theta) + (1-y_i)\log(1-\sigma(X_i^T\theta))\right)}$

Cross-Entropy Loss!!

Minimizing cross-entropy loss is equivalent to **maximizing the likelihood of the training data**.

- We are choosing the model parameters that are "most likely", given this data.

Assumption: all data drawn **independently** from the same logistic regression model with parameter θ

- It turns out that many of the model + loss combinations we've seen can be motivated using MLE (OLS, Ridge Regression, etc.)
- You will study MLE further in probability and ML classes. But now you know it exists.

We Did it!

	Regression ($y \in \mathbb{R}$)	Classification ($y \in \{0, 1\}$)
1. Choose a model ✅	Linear Regression $\hat{y} = f_\theta(x) = x^T\theta$	Logistic Regression $\hat{P}_\theta(Y = 1\|x) = \sigma(x^T\theta)$
2. Choose a loss function ✅	Squared Loss or Absolute Loss	Average Cross-Entropy Loss $-\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log(\sigma(X_i^T\theta) + (1 - y_i)\log(1 - \sigma(X_i^T\theta))\right)$
3. Fit the model	Regularization Sklearn/Gradient descent	rization t d
4. Evaluate model performance	R^2, Residuals, etc.	?? (next time)

That which we call a rose would by any other name smell as sweet.

Shakespeare [Wikipedia]

Next Time

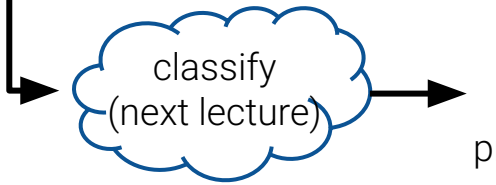
	Regression ($y \in \mathbb{R}$)	Classification ($y \in \{0, 1\}$)
1. Choose a model ✅	Linear Regression $\hat{y} = f_\theta(x) = x^T\theta$	Logistic Regression $\hat{P}_\theta(Y = 1\|x) = \sigma(x^T\theta)$
2. Choose a loss function ✅	Squared Loss or Absolute Loss	Average Cross-Entropy Loss $-\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log(\sigma(X_i^T\theta)) + (1 - y_i)\log(1 - \sigma(X_i^T\theta))\right)$
3. Fit the model	Regularization Sklearn/Gradient descent	Regularization Sklearn/~~Gradient descent~~
4. Evaluate model performance	R^2, Residuals, etc.	?? (next time)

Next Time: Classification

Logistic Regression model, parameter $\theta = (\theta_0, \theta_1, \ldots, \theta_p)$:

x → $\hat{P}_\theta(Y = 1|x) = \sigma(x^T\theta)$ → Probability of response = 1

observation vector



$\hat{y}$ ≈ $y \in \{0, 1\}$

prediction

categorical response

LECTURE 21

Logistic Regression I

Content credit: Lisa Yan, Suraj Rampure, Ani Adhikari, Josh Hug, Joseph Gonzalez