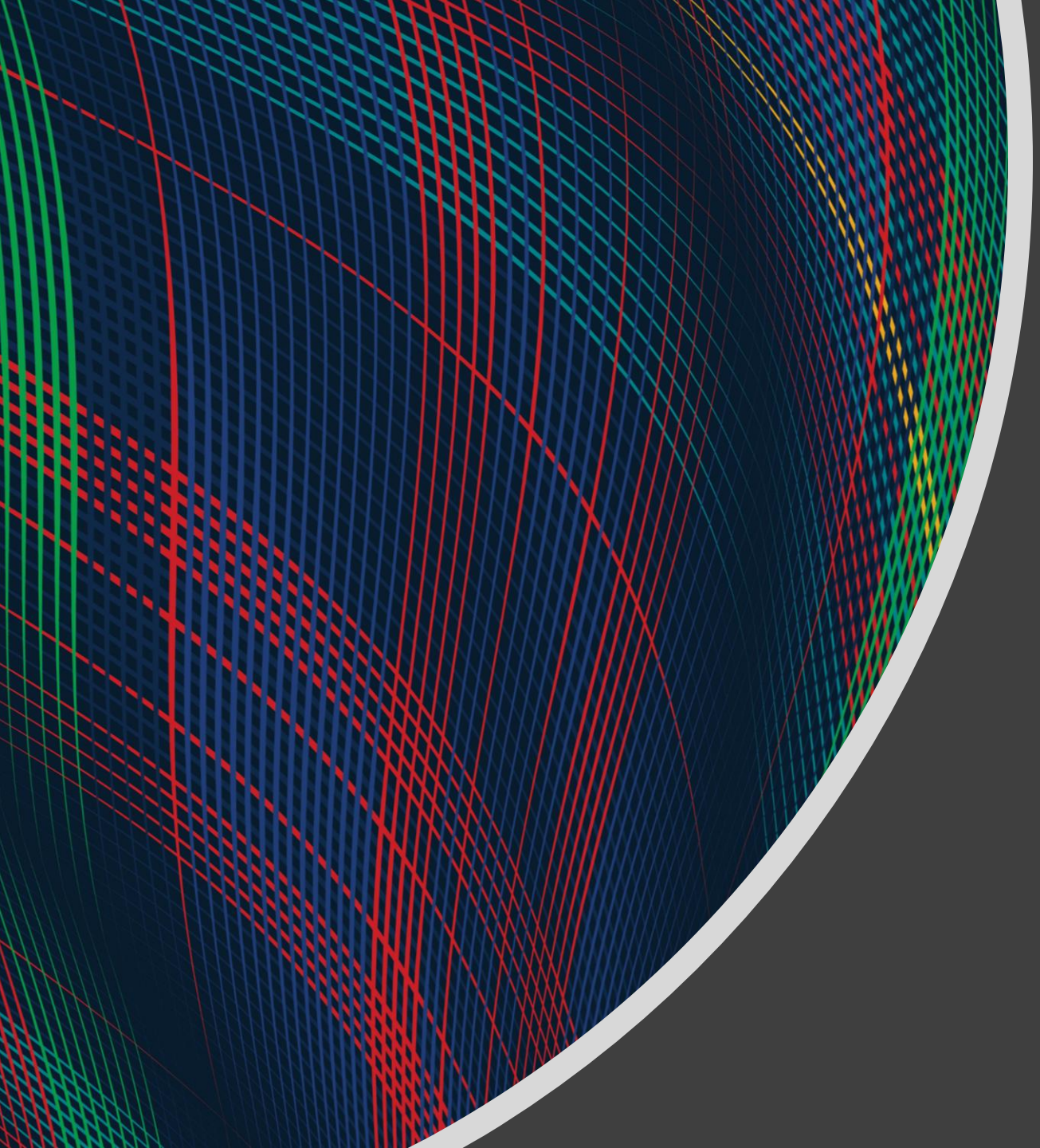


10-315 Introduction to ML

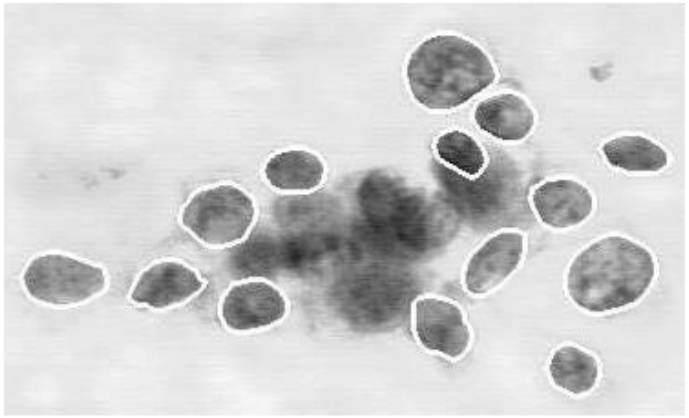
Logistic Regression

Instructor: Pat Virtue

Example: Breast cancer classification

Well-known classification example: using machine learning to diagnose whether a breast tumor is benign or malignant [Street et al., 1992]

Setting: doctor extracts a sample of fluid from tumor, stains cells, then outlines several of the cells (image processing refines outline)

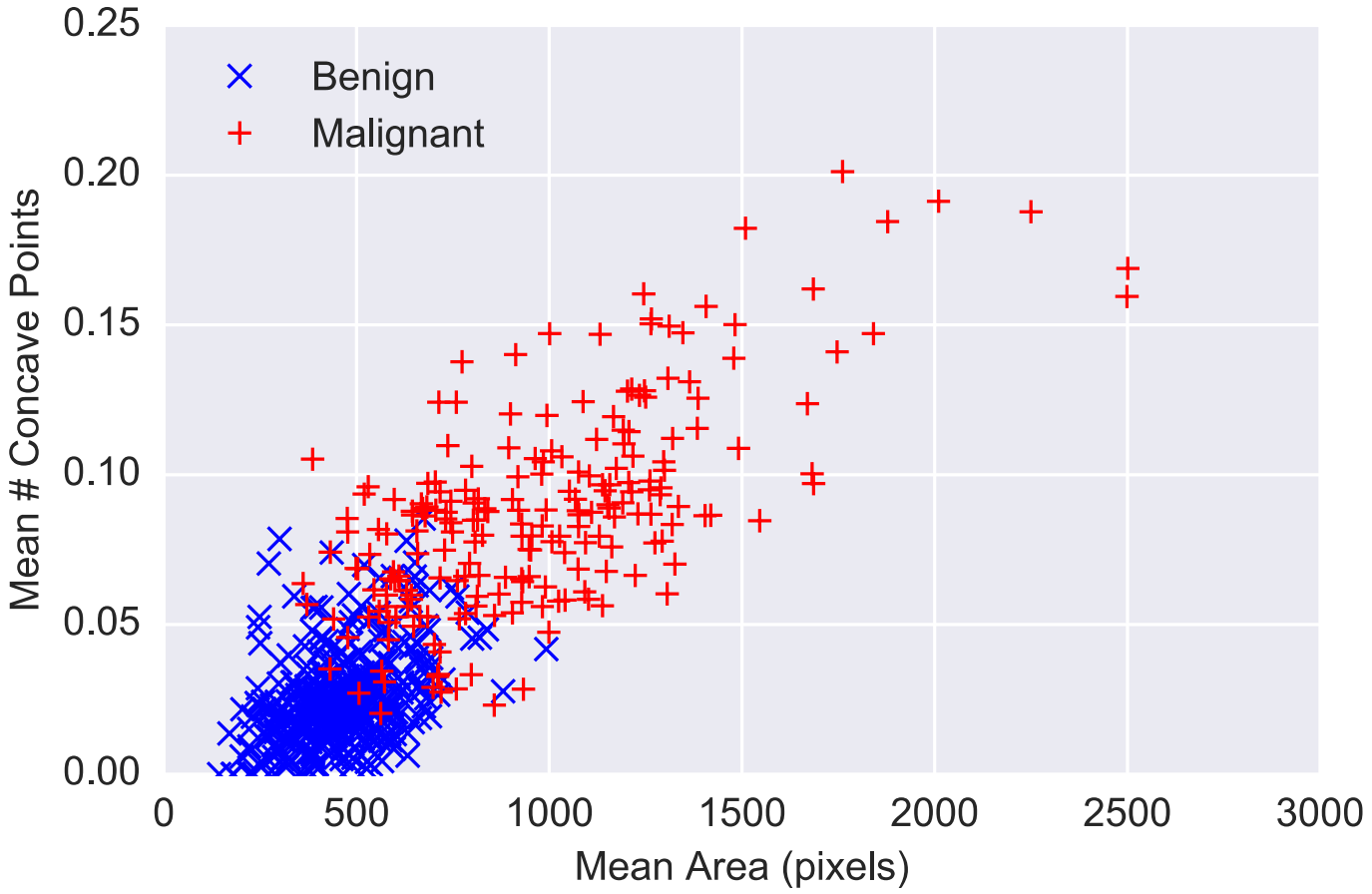


System computes features for each cell such as area, perimeter, concavity, texture (10 total); computes mean/std/max for all features

Slide credit: CMU AI Zico Kolter

Example: Breast cancer classification

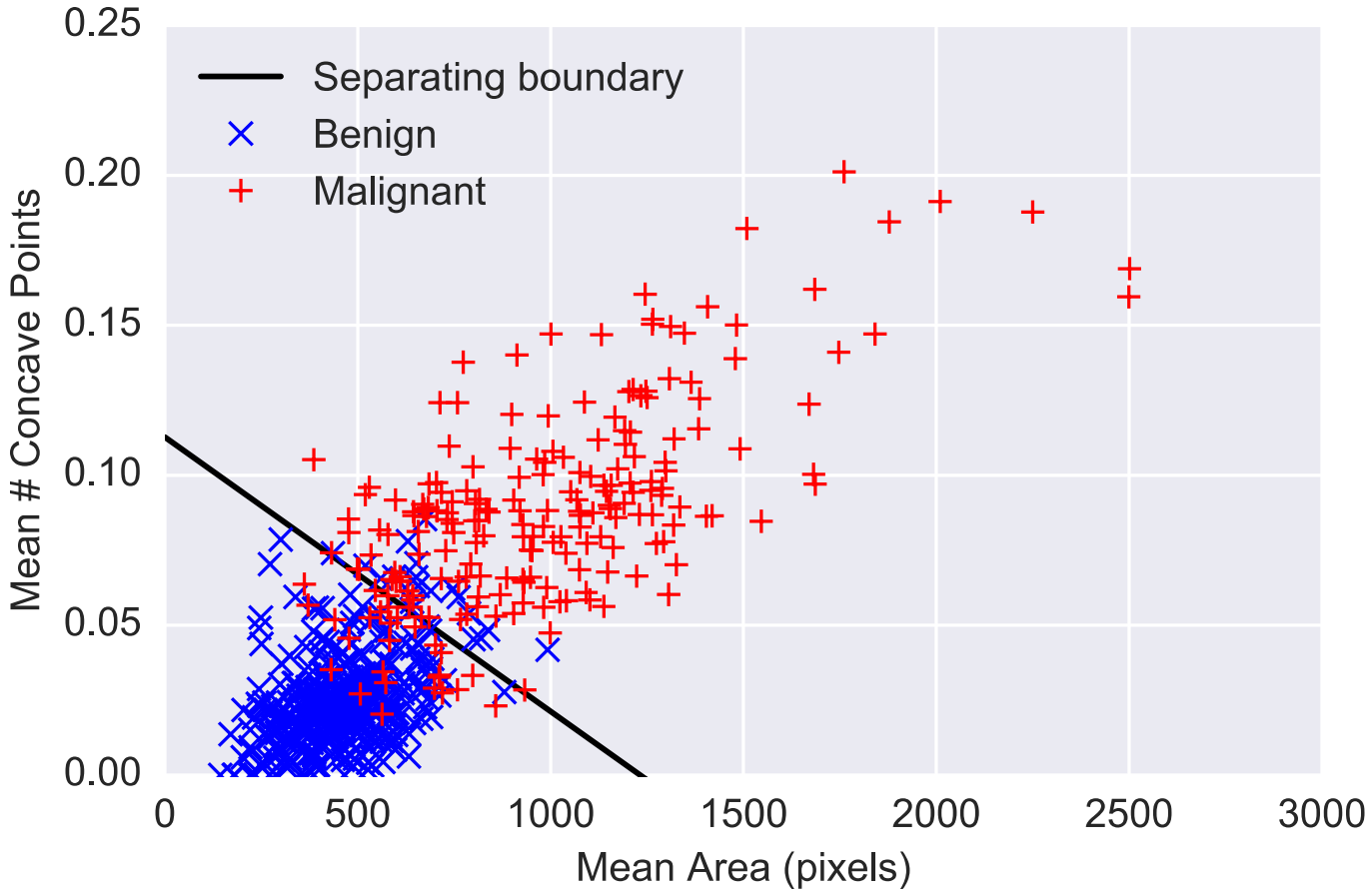
Plot of two features: mean area vs. mean concave points, for two classes



Slide credit: CMU AI Zico Kolter

Linear classification example

Linear classification: linear decision boundary

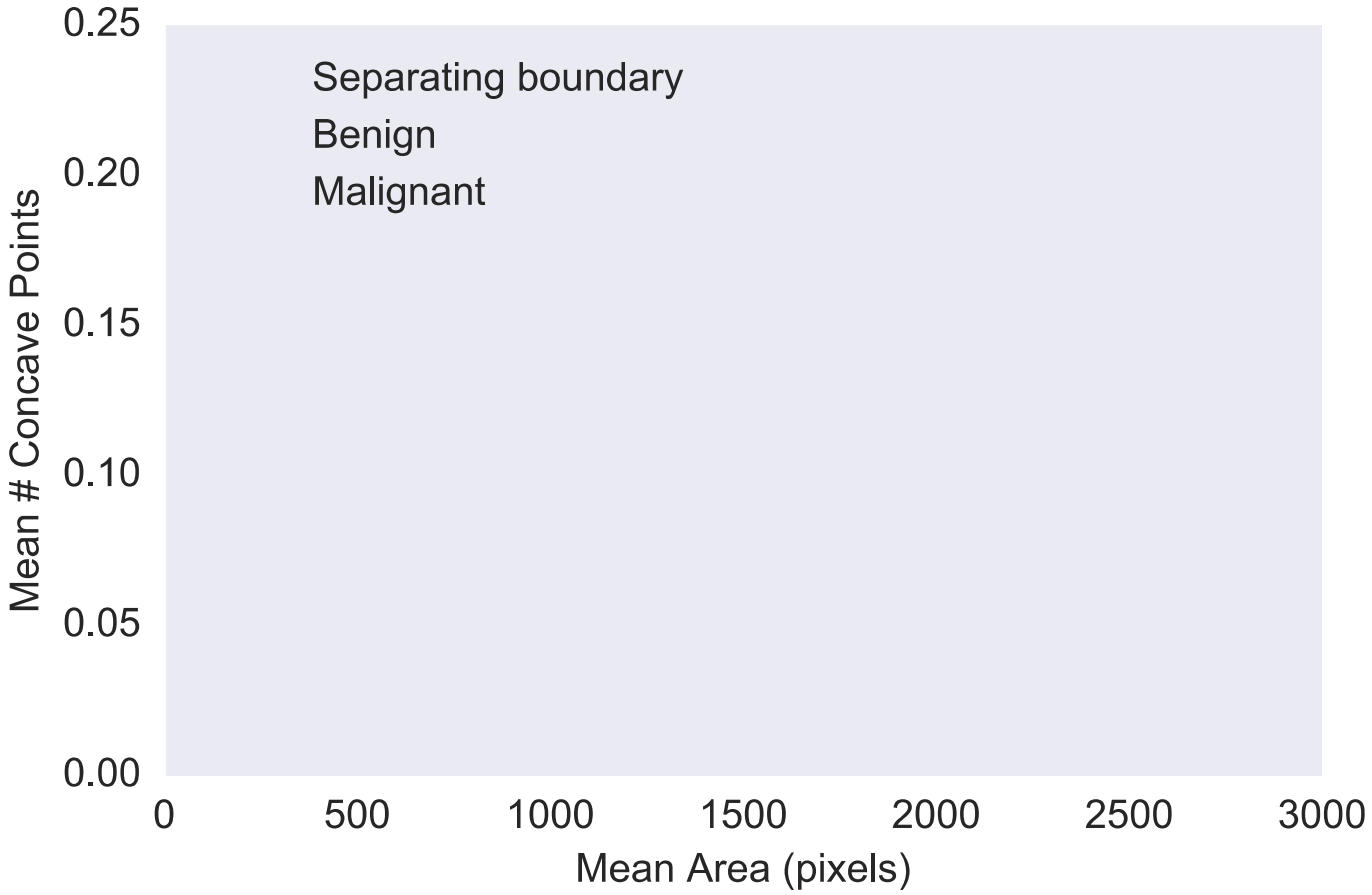


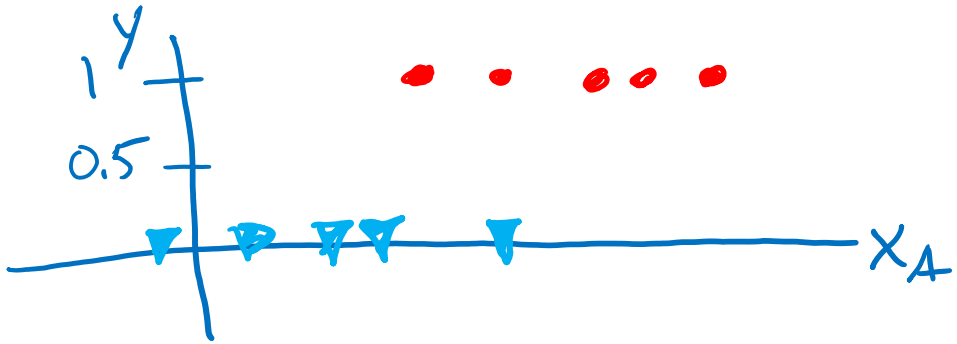
Slide credit: CMU AI Zico Kolter

Logistic regression for classification

Linear classification: linear decision boundary

Probabilistic classification: provide $P(Y = 1 \mid x)$ rather than just $\hat{y} \in \{0, 1\}$



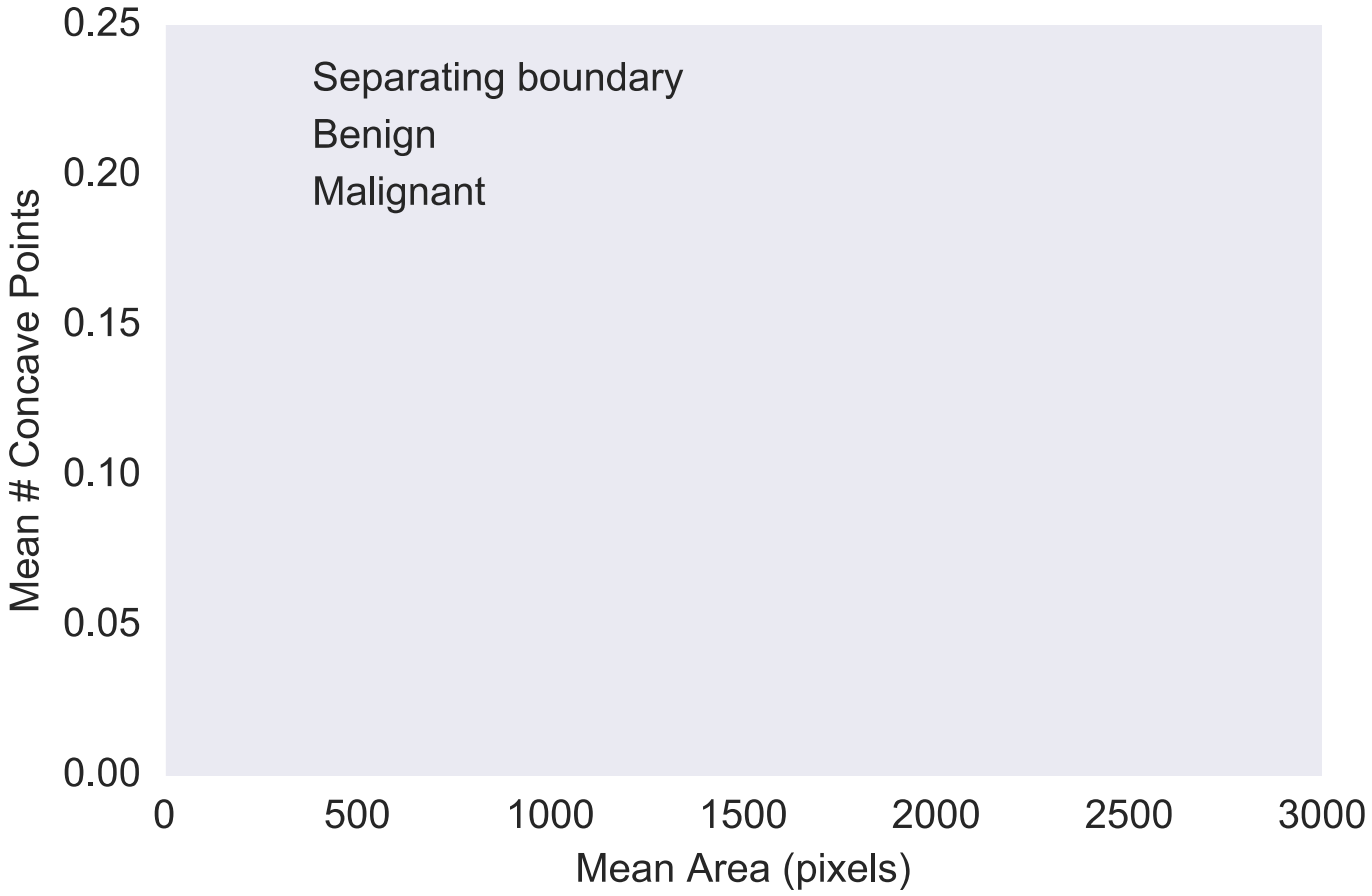


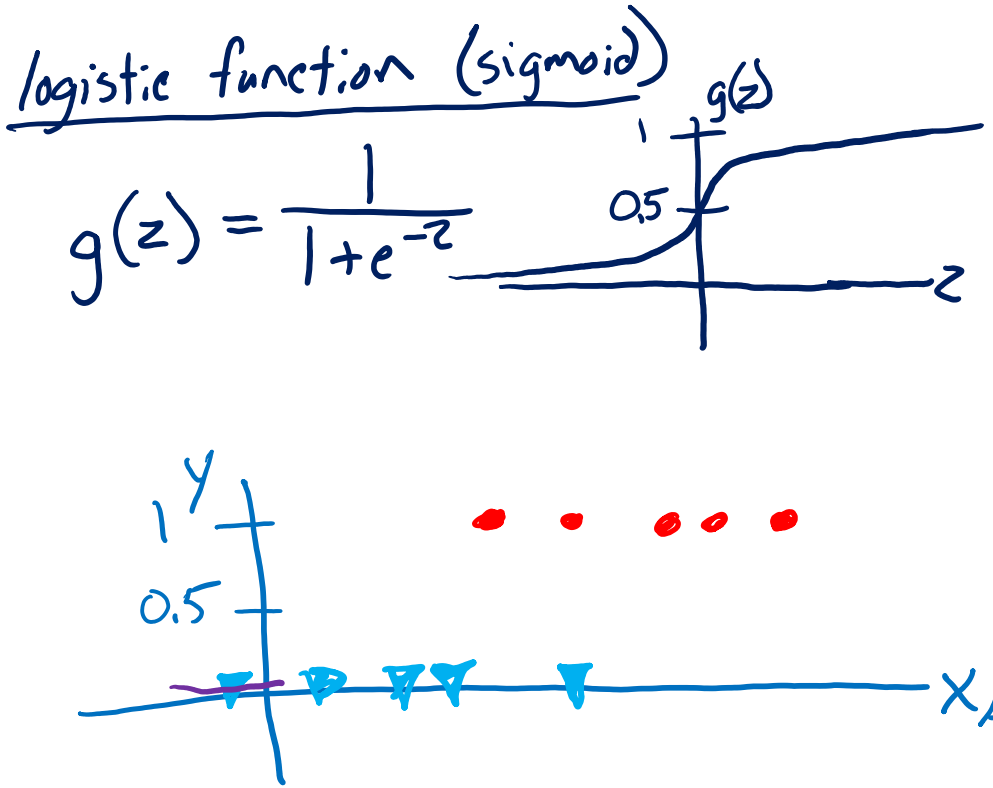
Slide credit: CMU AI Zico Kolter

Logistic regression for classification

Linear classification: linear decision boundary

Probabilistic classification: provide $P(Y = 1 \mid x)$ rather than just $\hat{y} \in \{0, 1\}$



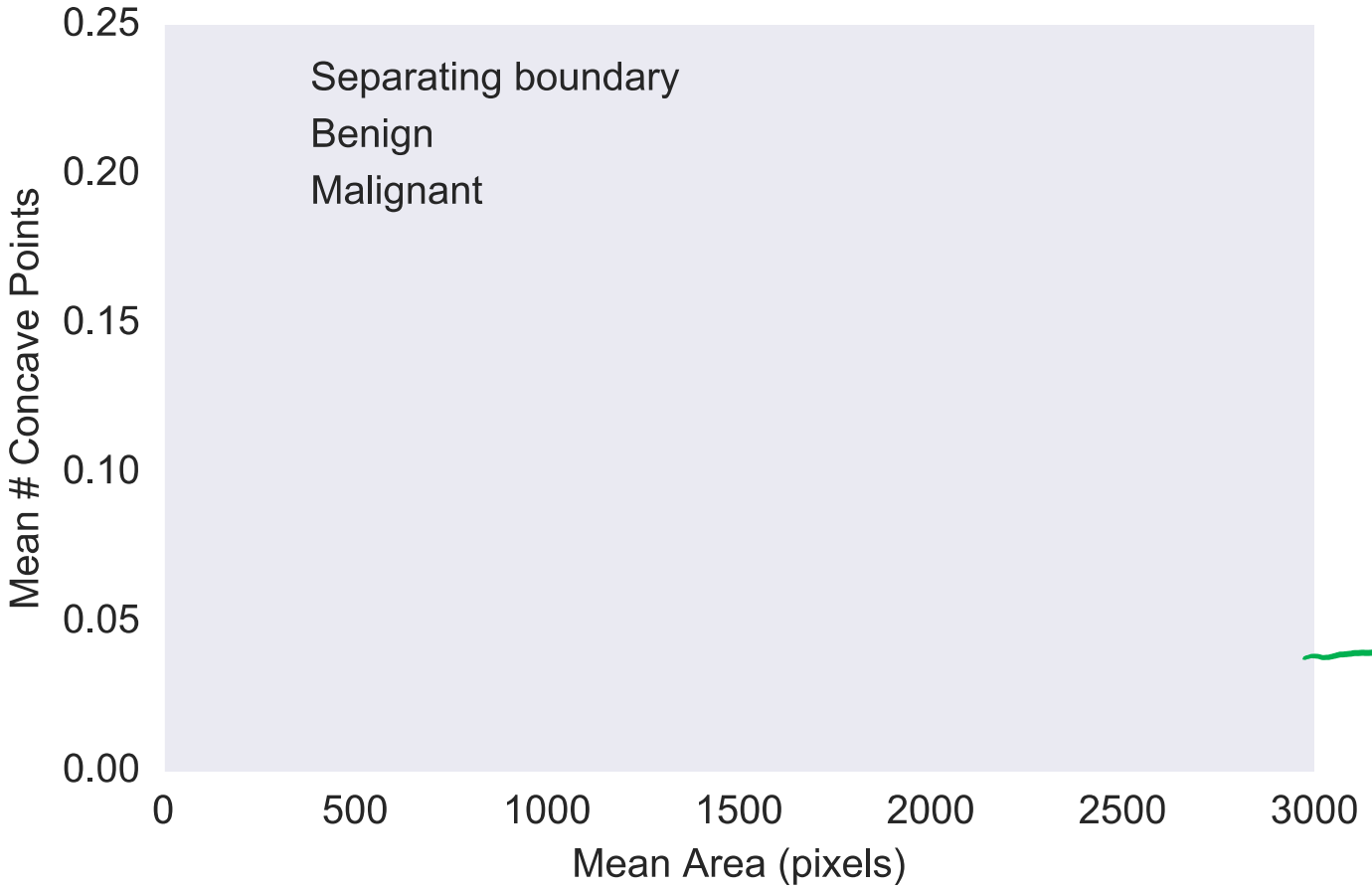


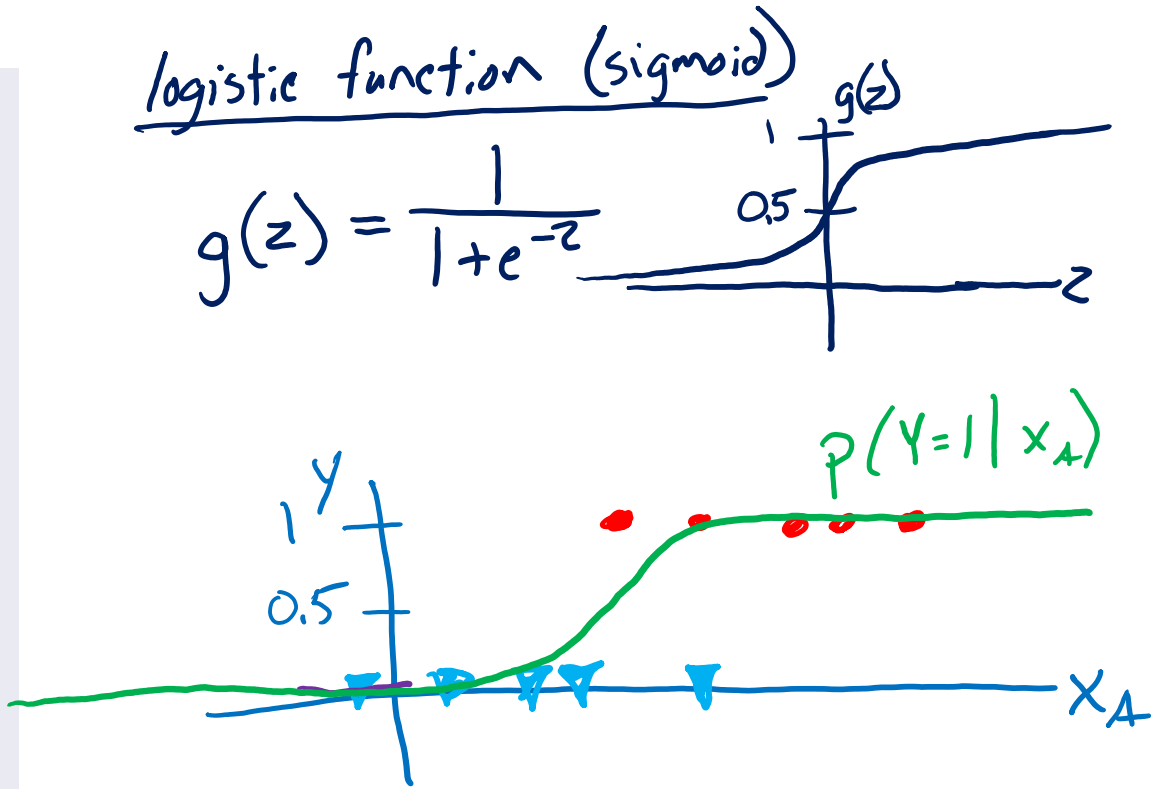
Slide credit: CMU AI Zico Kolter

Logistic regression for classification

Linear classification: linear decision boundary

Probabilistic classification: provide $P(Y = 1 \mid x)$ rather than just $\hat{y} \in \{0, 1\}$





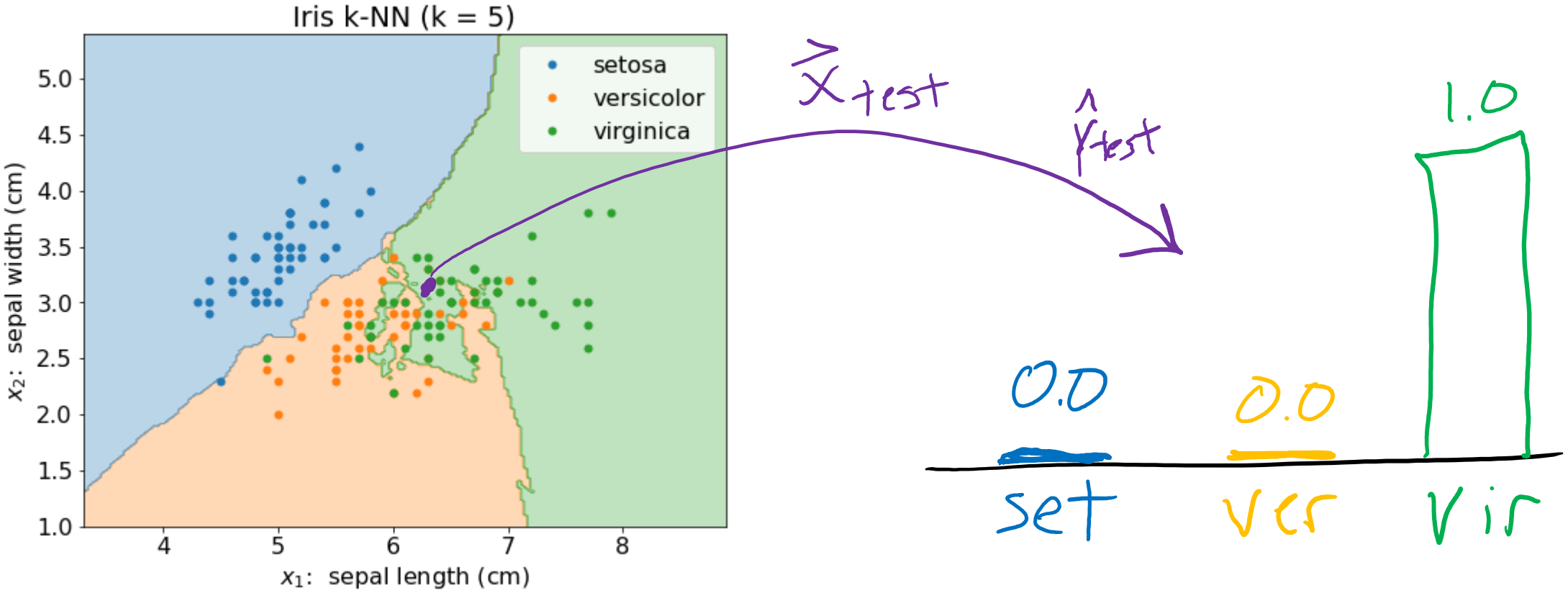
Slide credit: CMU AI Zico Kolter

Pre-reading

Cross-entropy loss

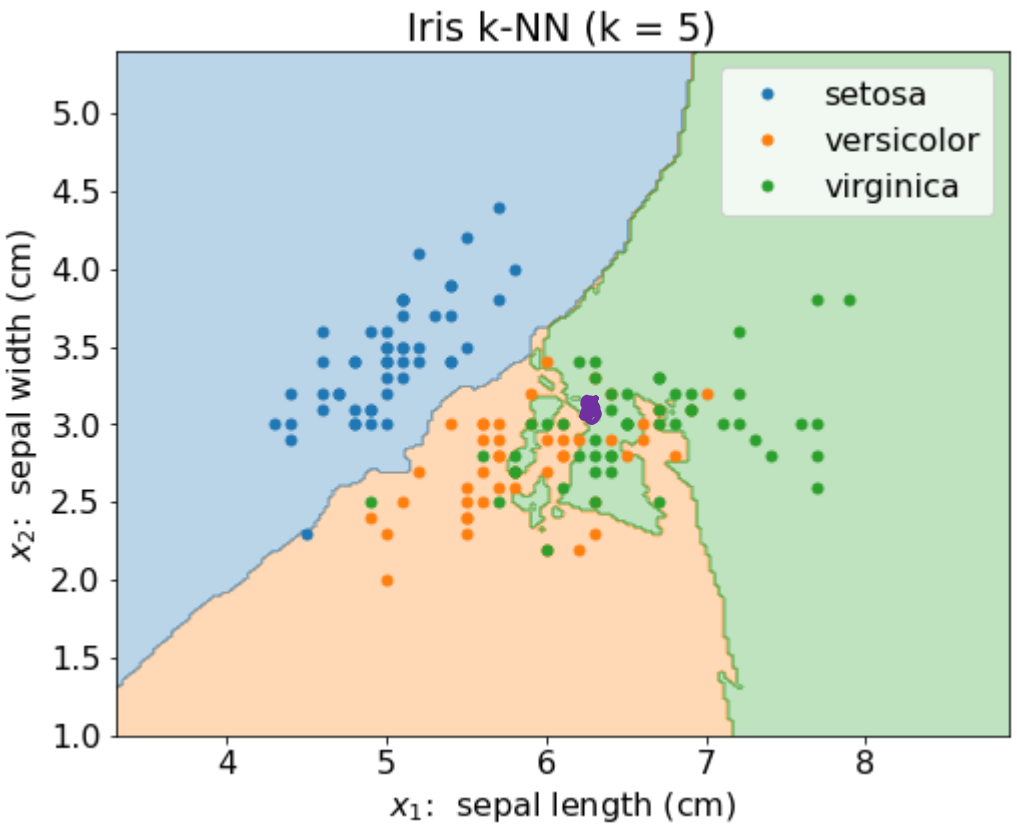
Classification Decisions

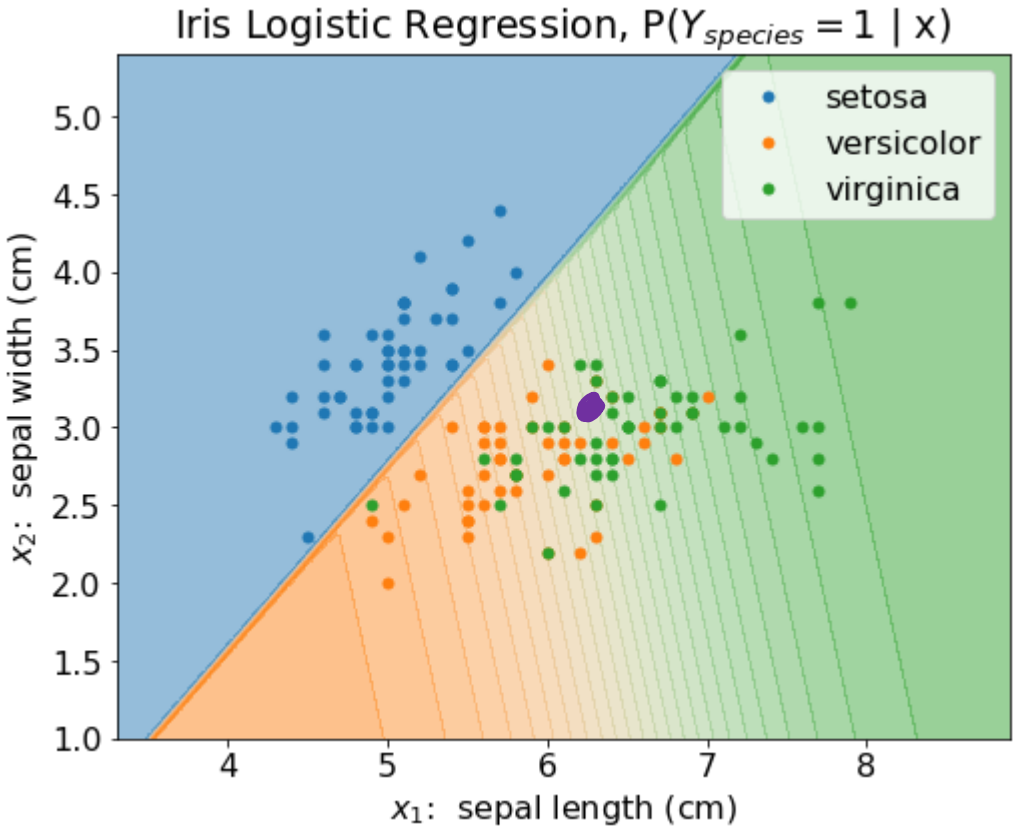
Predicting one specific class is troubling, especially when we know that there is some uncertainty in our prediction



Classification Probability

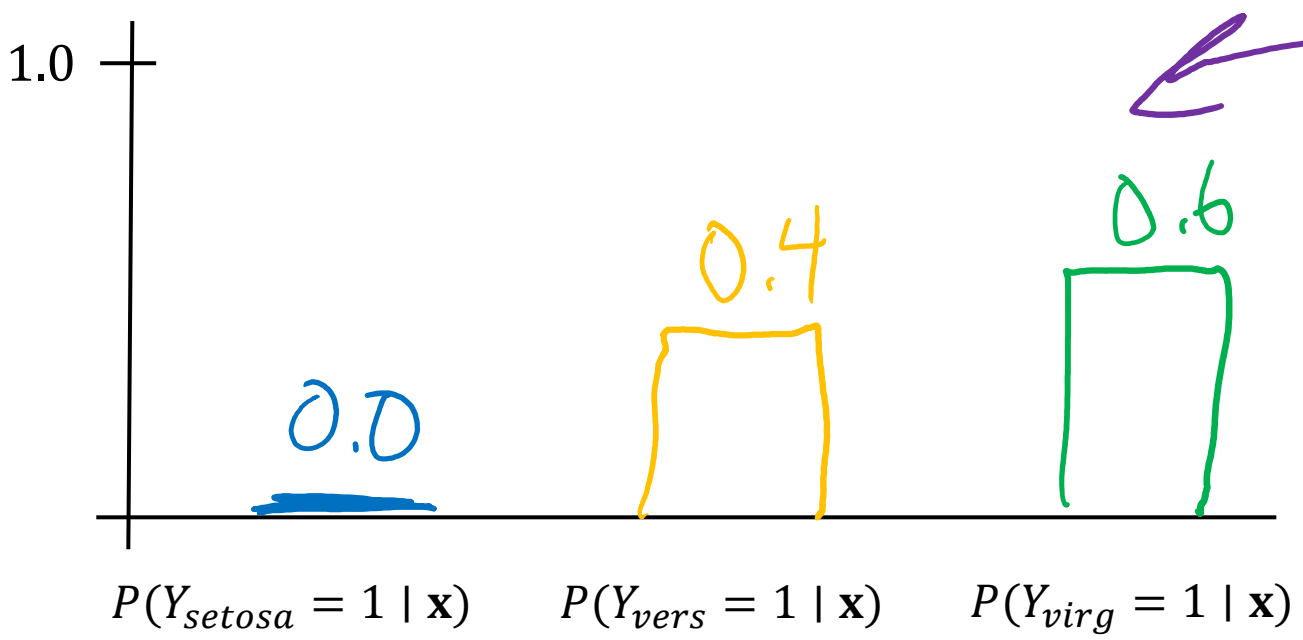
Constructing a model than can return the probability of the output being a specific class could be incredibly useful

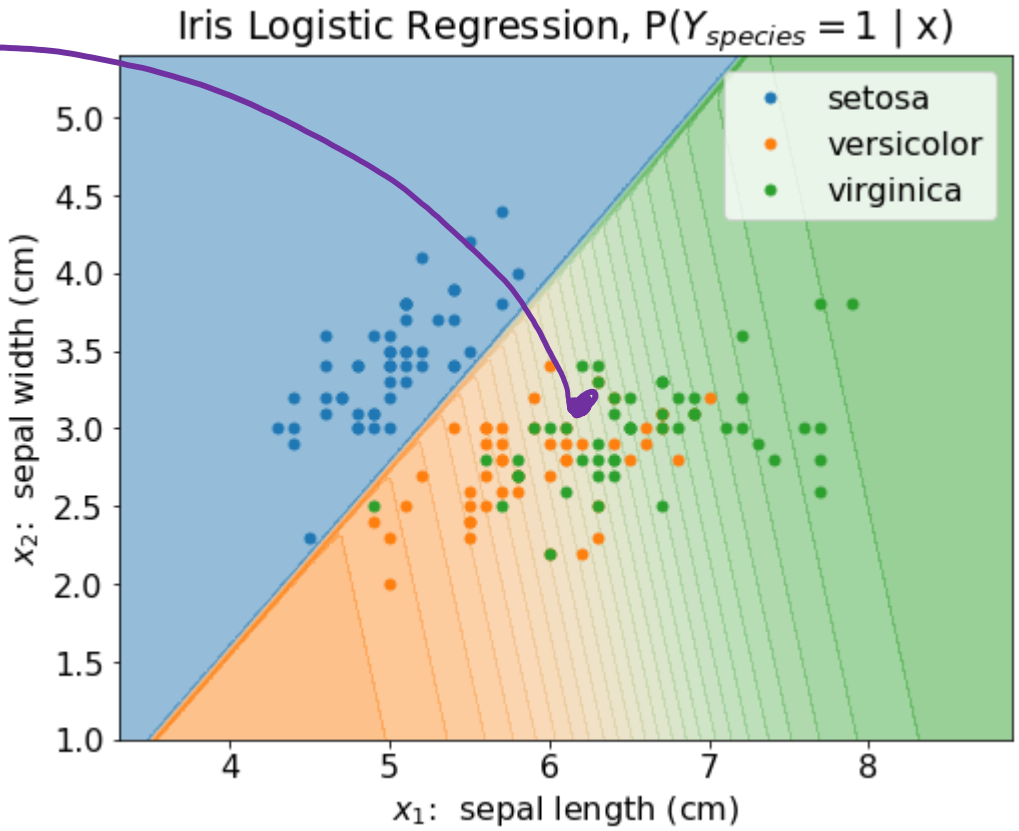




Classification Probability

Constructing a model than can return the probability of the output being a specific class could be incredibly useful





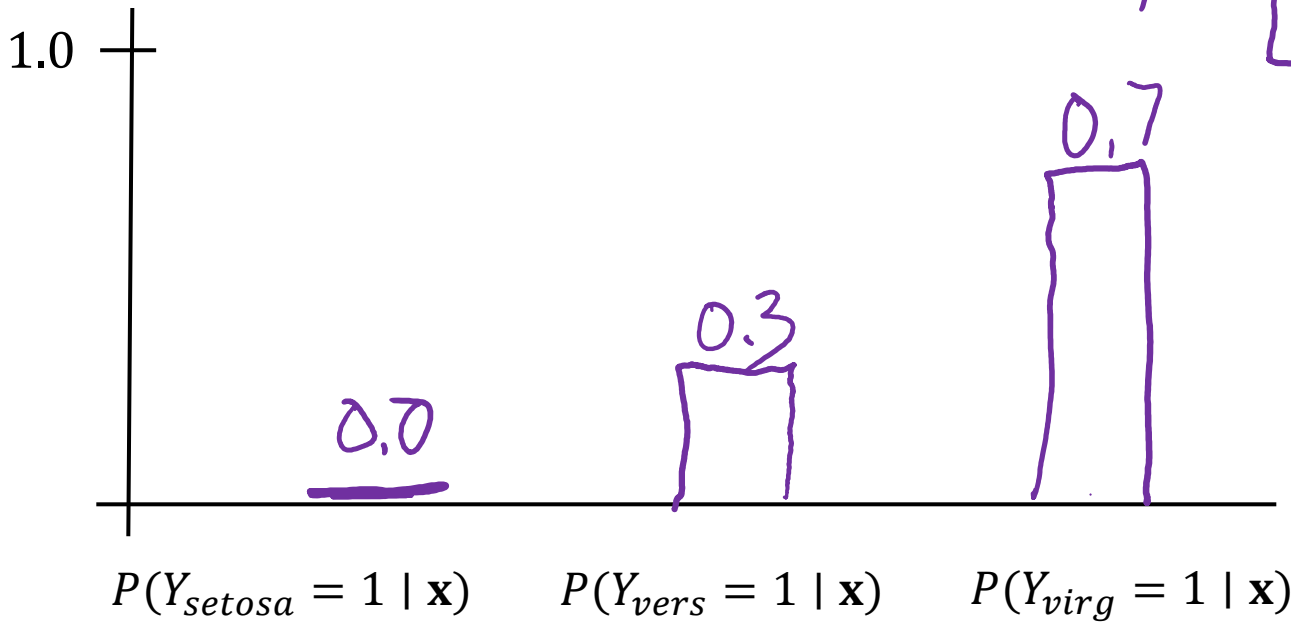
We can still make decisions, .e.g,

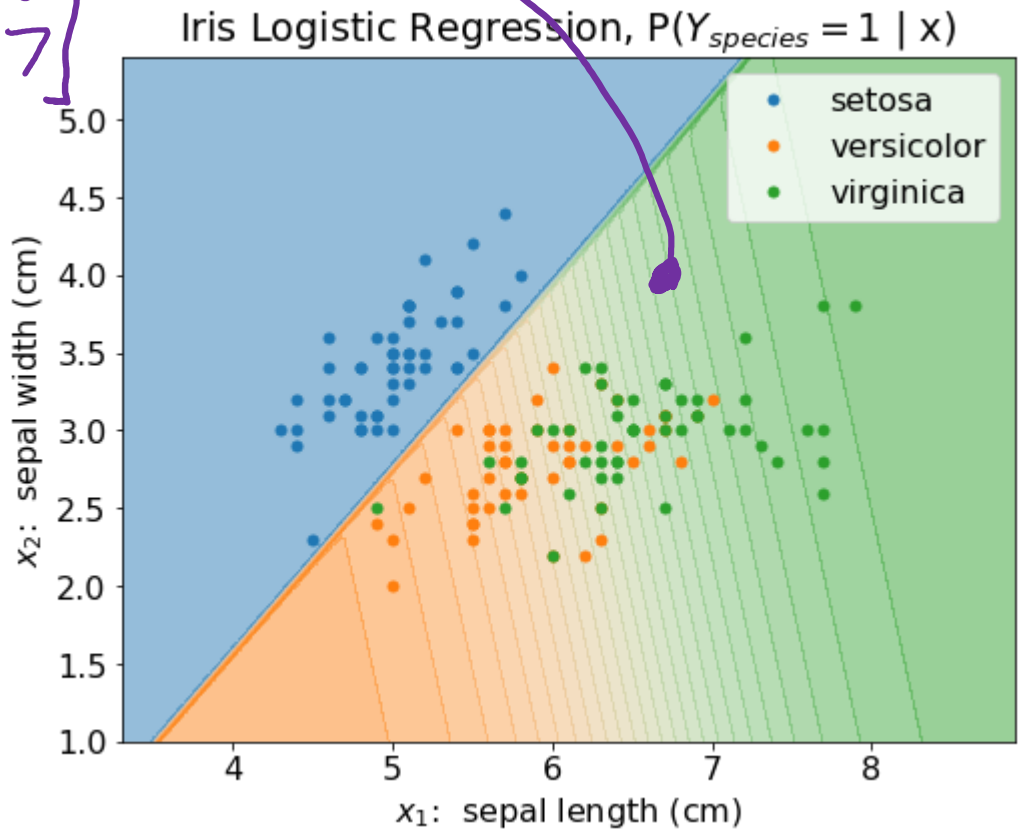
$$\underset{k}{\text{argmax}}\, P(Y_k = 1 \mid \mathbf{x})$$

Loss for Probabilty Disributions

We need a way to compare how good/bad each prediction is

$$\hat{y} = \begin{bmatrix} 0.0 \\ 0.3 \\ 0.7 \end{bmatrix}$$



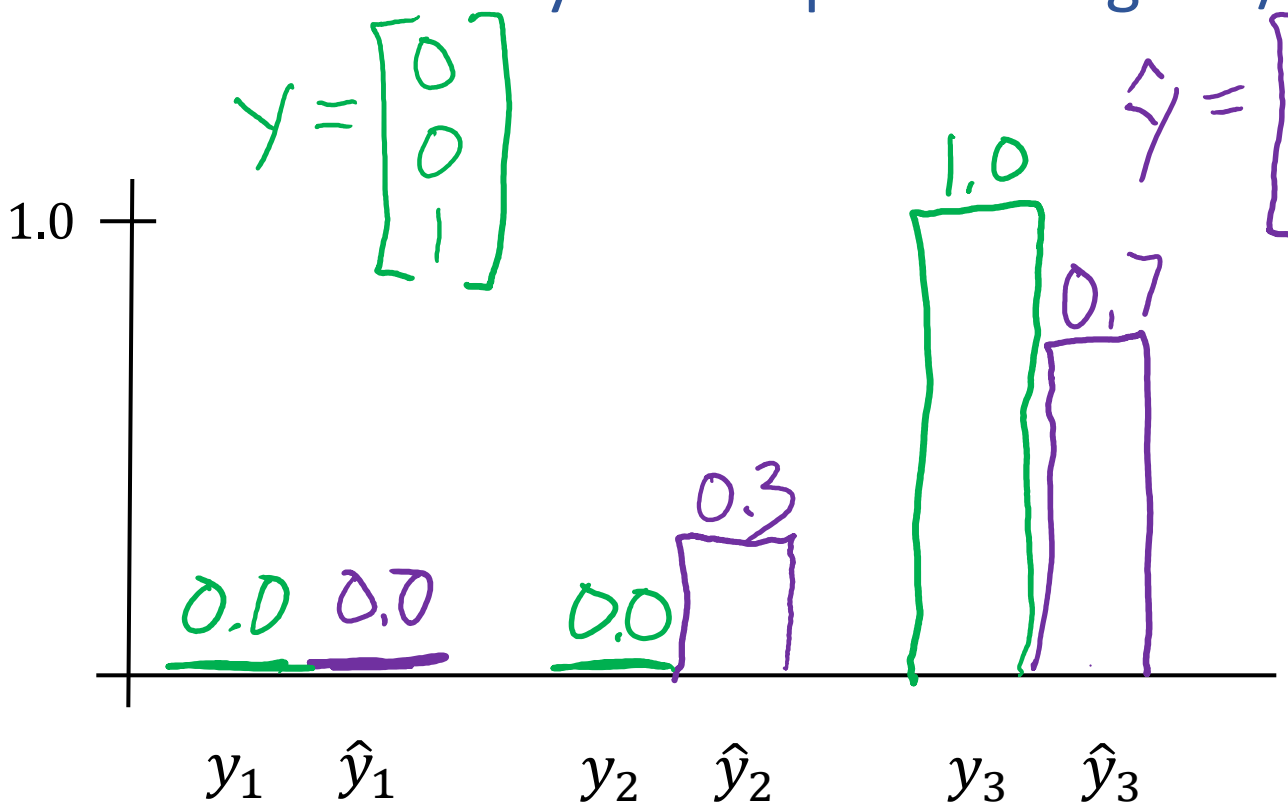


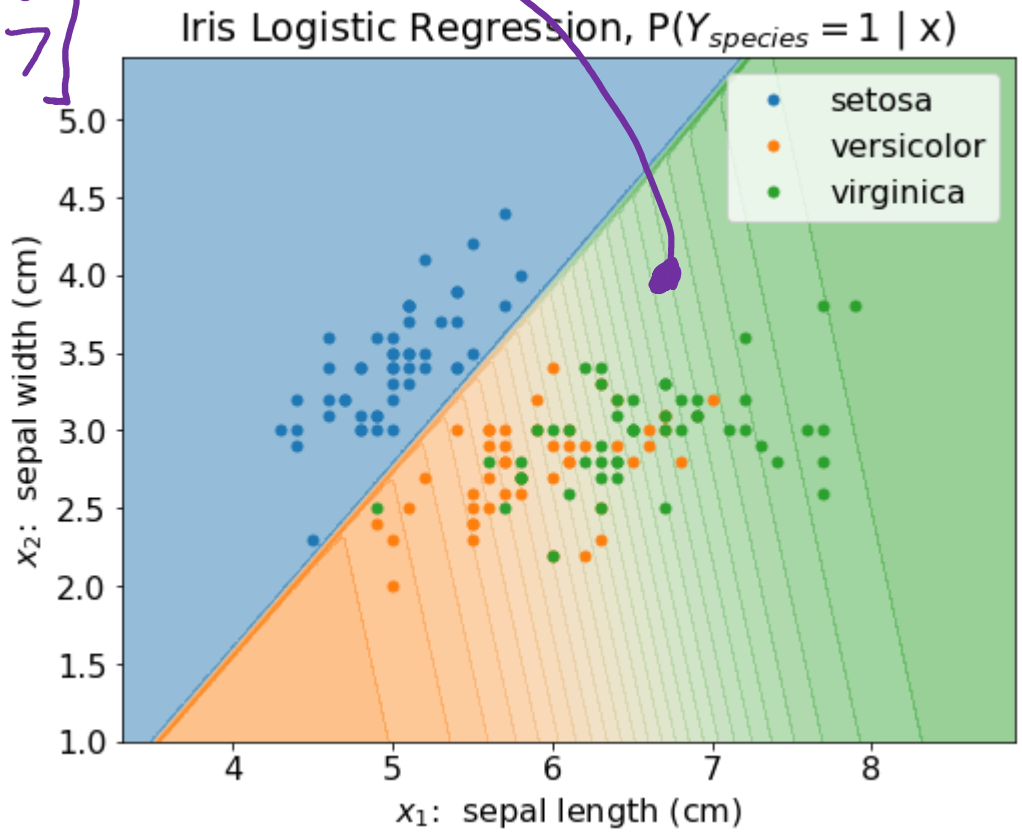
Cross-entropy loss

$$\ell(\mathbf{y}, \hat{\mathbf{y}}) = -\sum_{k=1}^{K} y_k \log \hat{y}_k$$

Loss for Probabilty Disributions

We need a way to compare how good/bad each prediction is





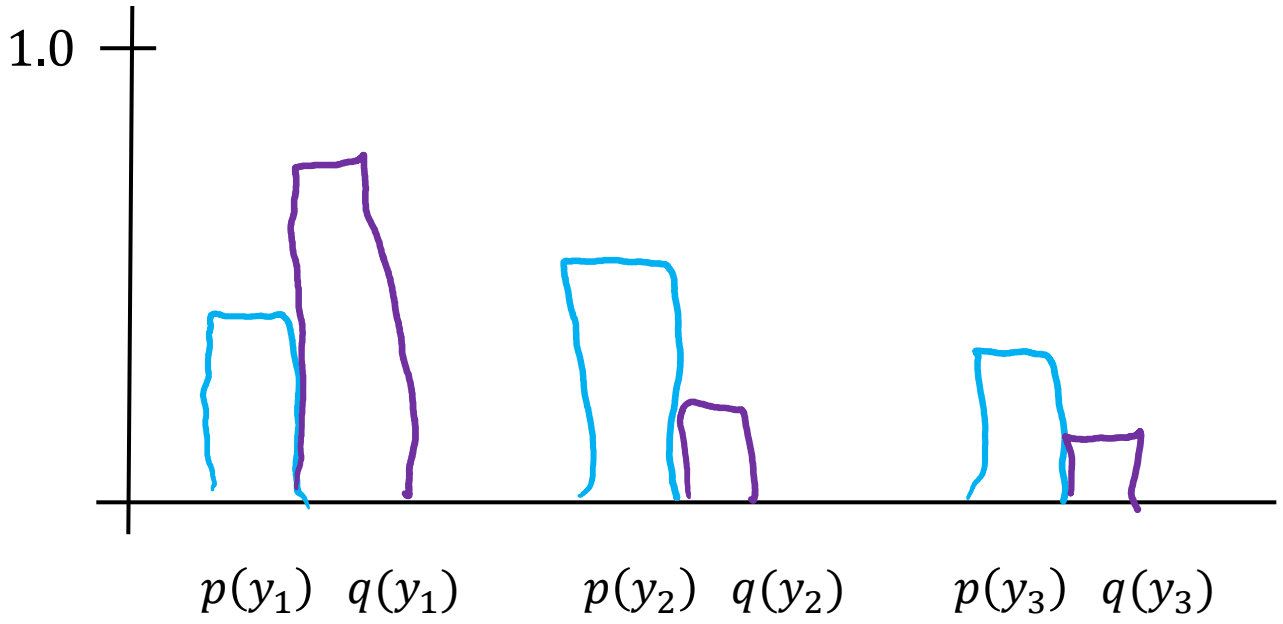
Cross-entropy loss

$$\ell(\mathbf{y}, \hat{\mathbf{y}}) = -\sum_{k=1}^{K} y_k \log \hat{y}_k$$

Loss for Probabilty Disributions

Cross-entropy more generally is a way to compare any to probability distributions*

*when used in logistic regression
$\mathbf{y}$ is always a one-hot vector



Cross-entropy loss

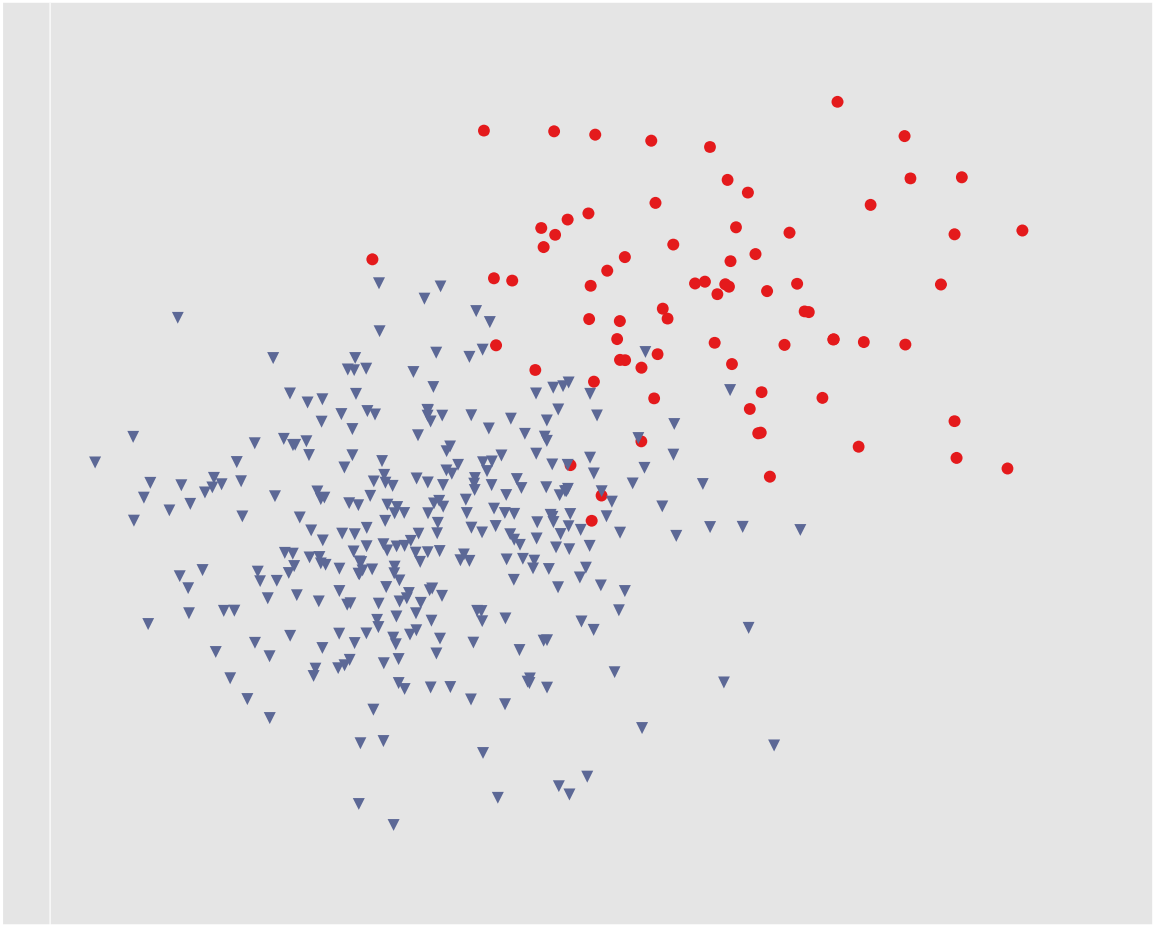
$$H(P, Q) = -\sum_{k=1}^{K} p(y_k) \log q(y_k)$$

Pre-reading

Linear model for classification

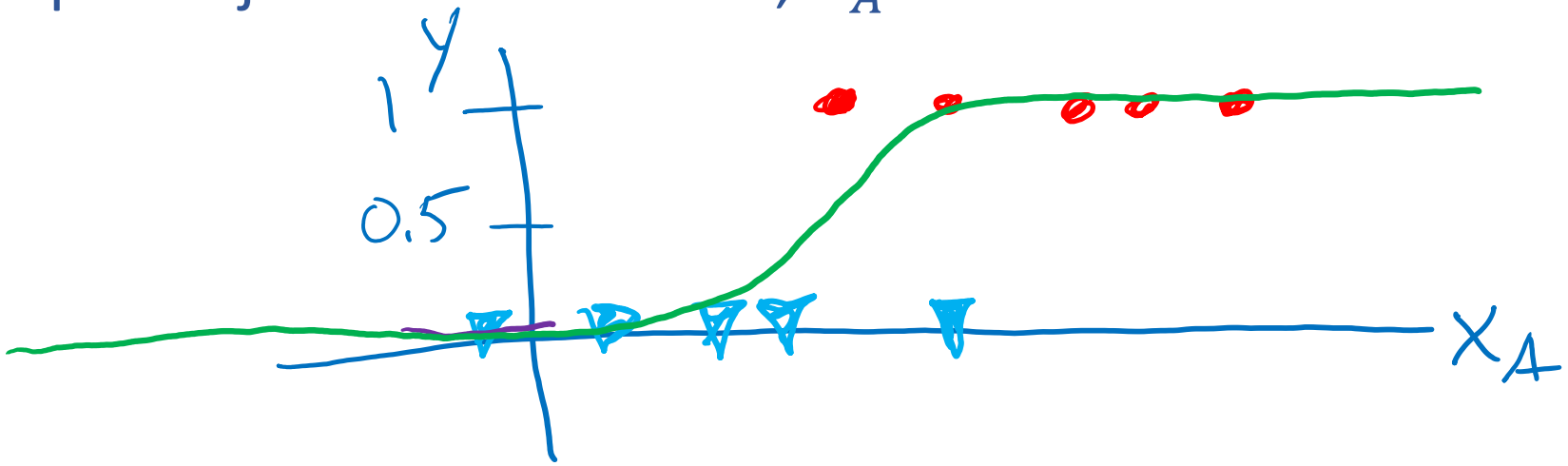
Prediction for Cancer Diagnosis

Learn to predict if a patient has cancer ($Y = 1$) or not ($Y = 0$) given the input of two test results, X_A and X_B.



Prediction for Cancer Diagnosis

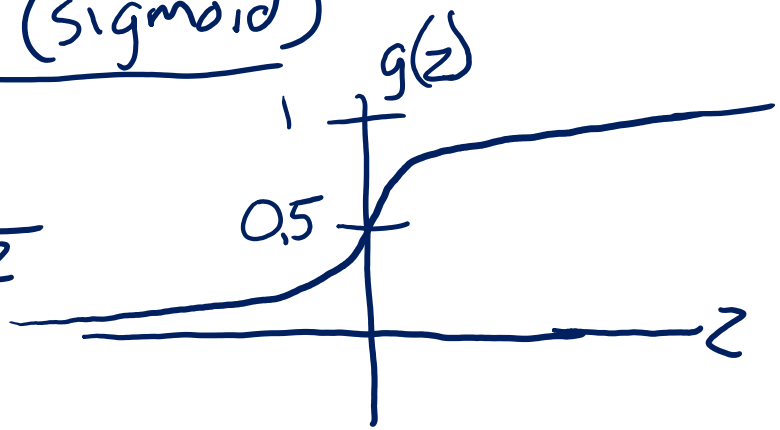
Learn to predict if a patient has cancer ($Y = 1$) or not ($Y = 0$) given the input of just one test result, X_A.



$p(Y=1|X_A)$

logistic function (sigmoid)

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

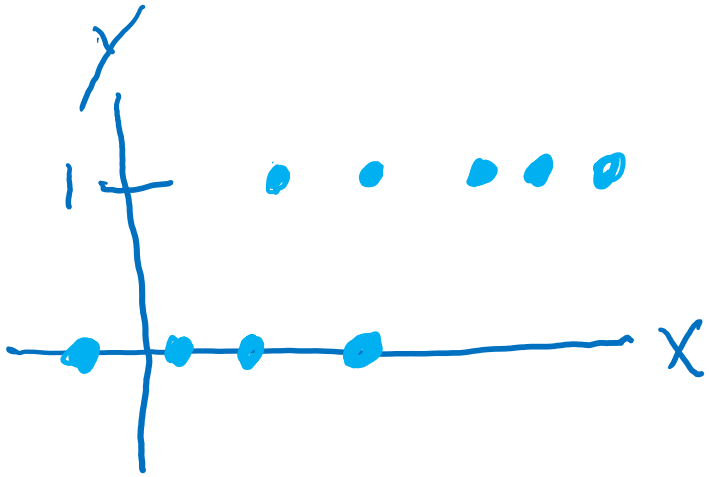


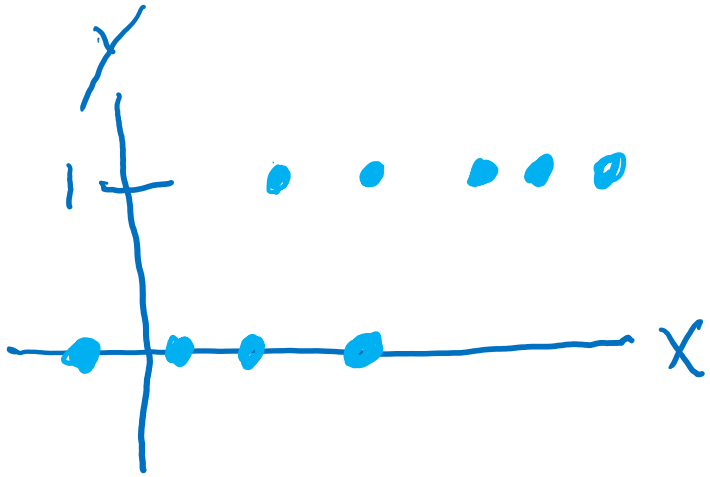
logistic regression

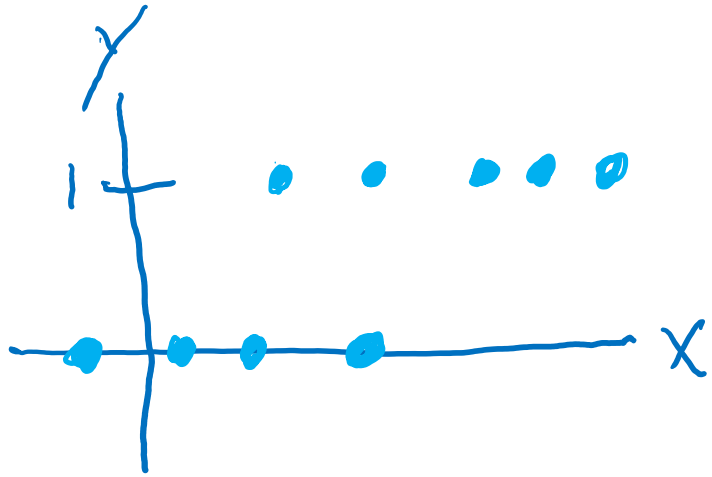
$$p(Y=1|\vec{x},\vec{\theta}) = g(\theta^T x)$$

Building on a Linear Model

Linear vs Thresholded Linear vs Logistic Linear

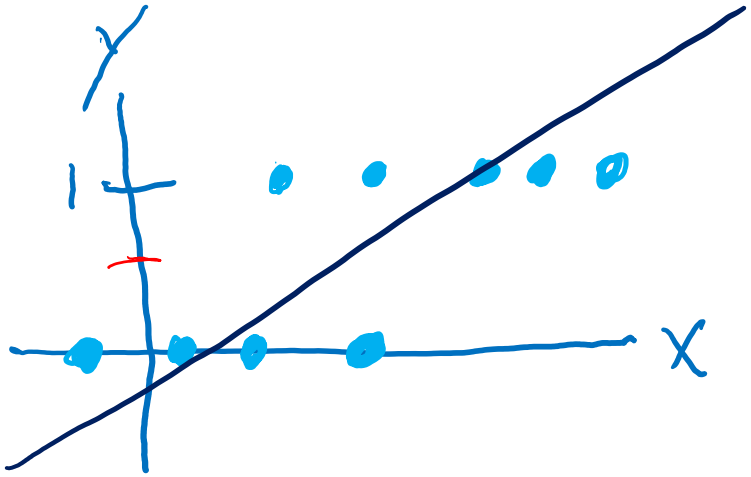






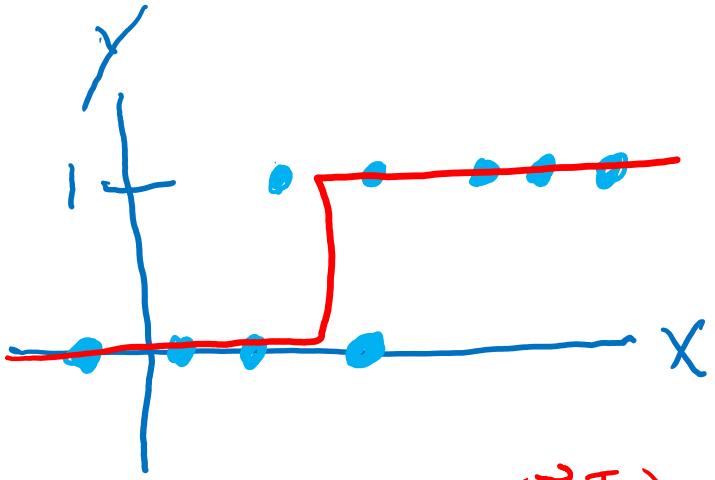
Building on a Linear Model

Linear vs Thresholded Linear vs Logistic Linear



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

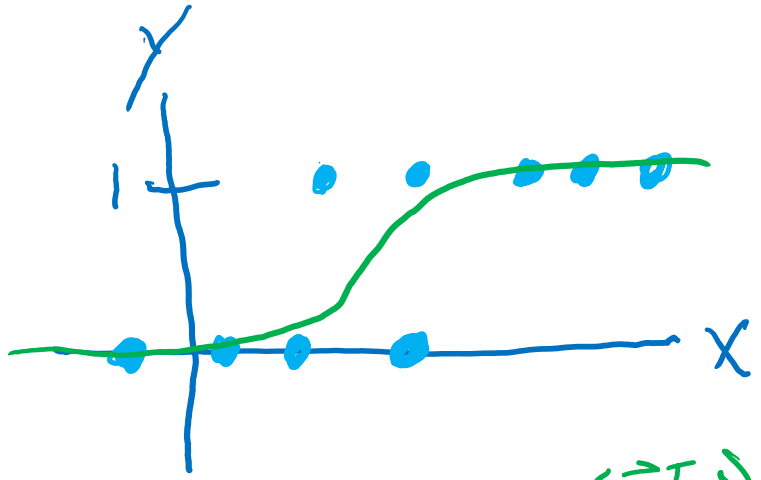
∴ not classification



$\hat{y} = g_{thresh}(\vec{\theta}^T \vec{x})$

∴ classification only (0/1)

∴ zero derivatives



$\hat{y} = g_{logistic}(\vec{\theta}^T \vec{x})$

Logistic Regression

Linear model for classification

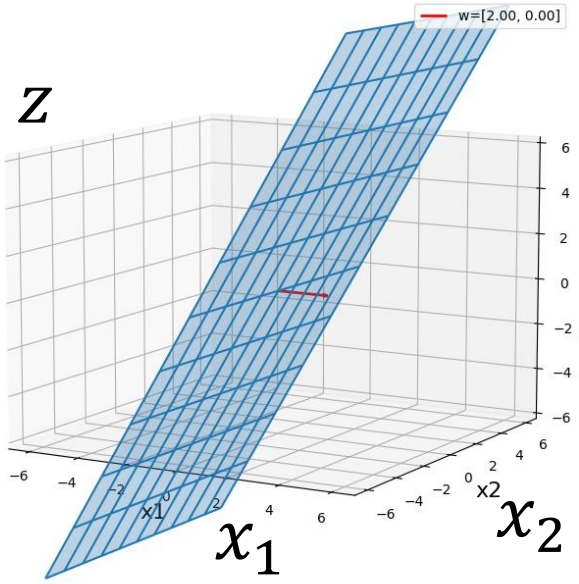
(now with 2+ input features)

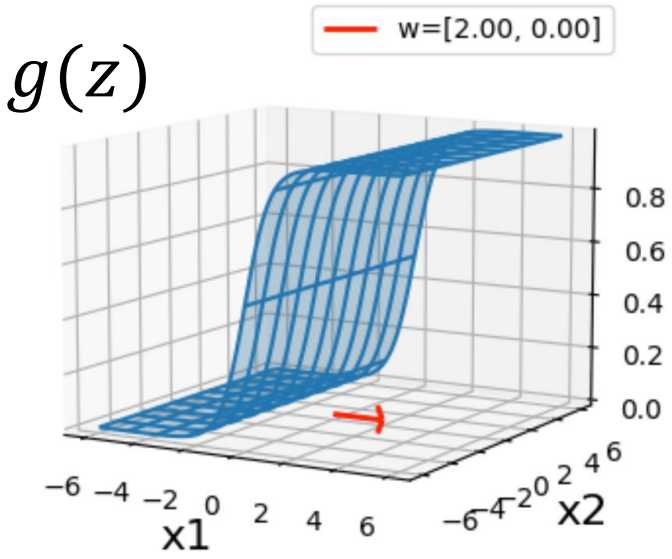
Building on a Linear Model

With two input features, $\mathbf{x} = [x_1 \;\; x_2]^\top$, we have two weight parameters and one bias parameter, $\mathbf{w} = [w_1 \;\; w_2]^\top$ and b, that control the slope and vertical offset of the following plane:

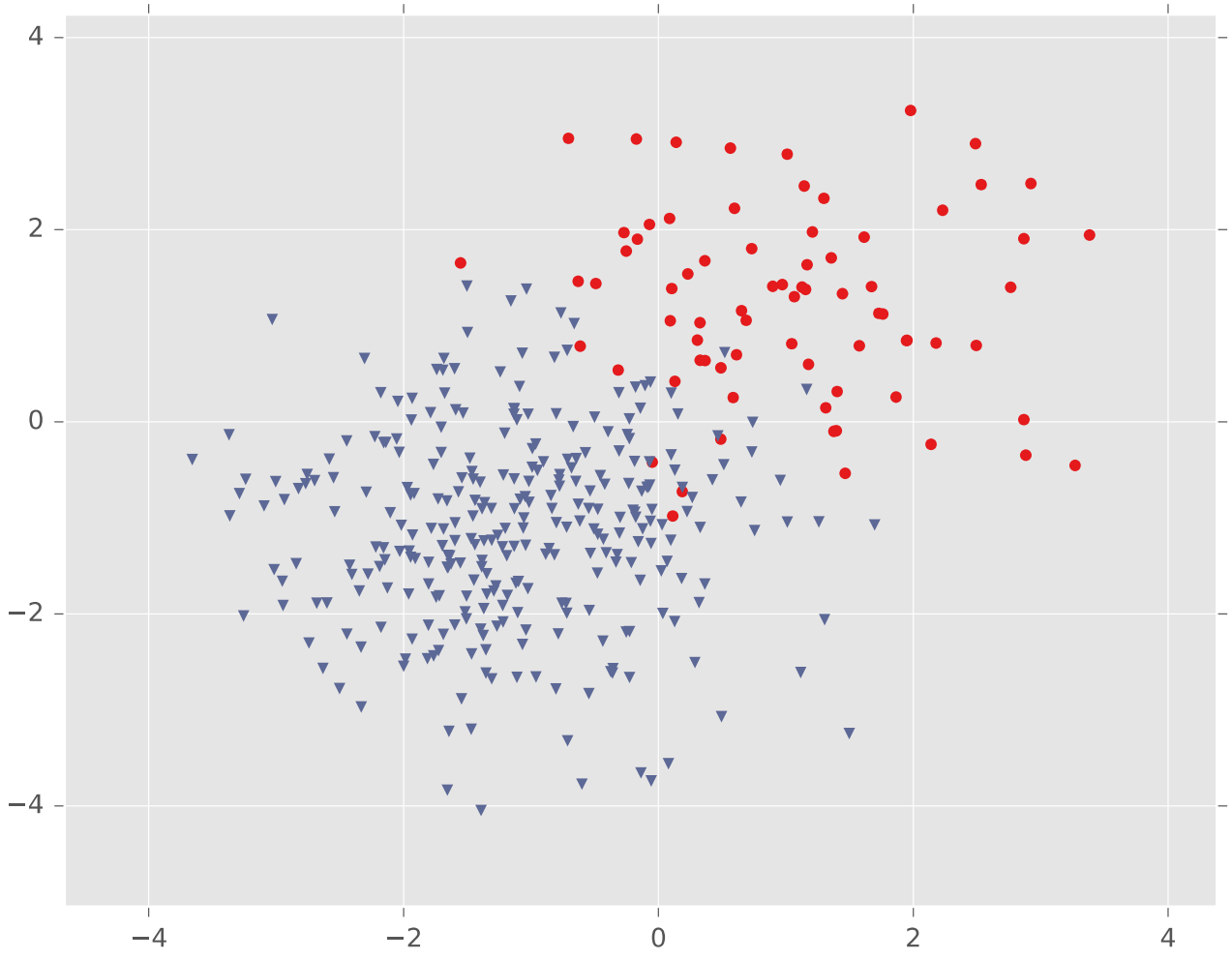
$$z = \mathbf{w}^\top \mathbf{x} + b$$

The sigmoid function $\hat{y} = g(z)$ then squashed the plane such that any z values going to $+\infty$ go to 1 and z values going to $-\infty$ go to -1



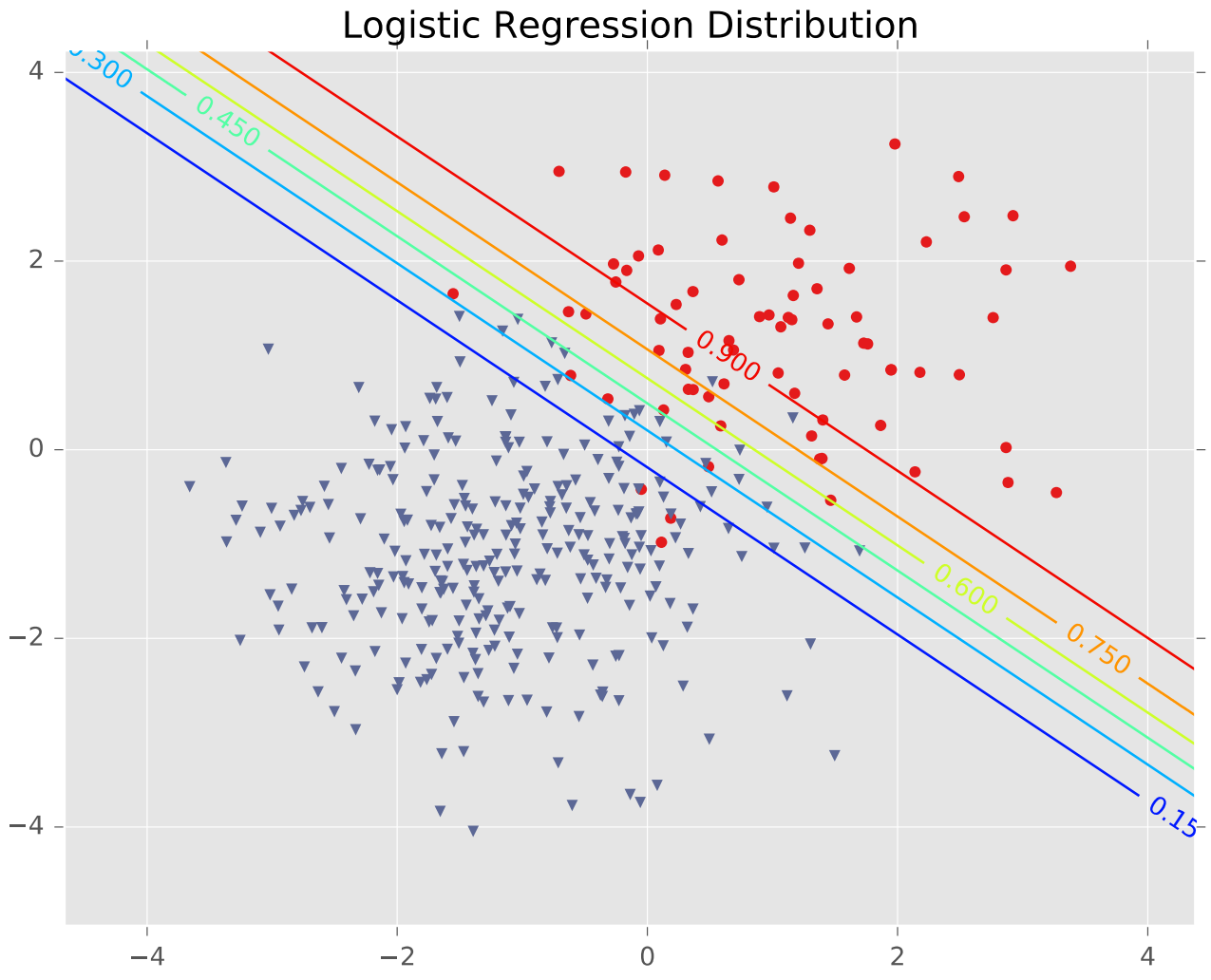


Building on a Linear Model



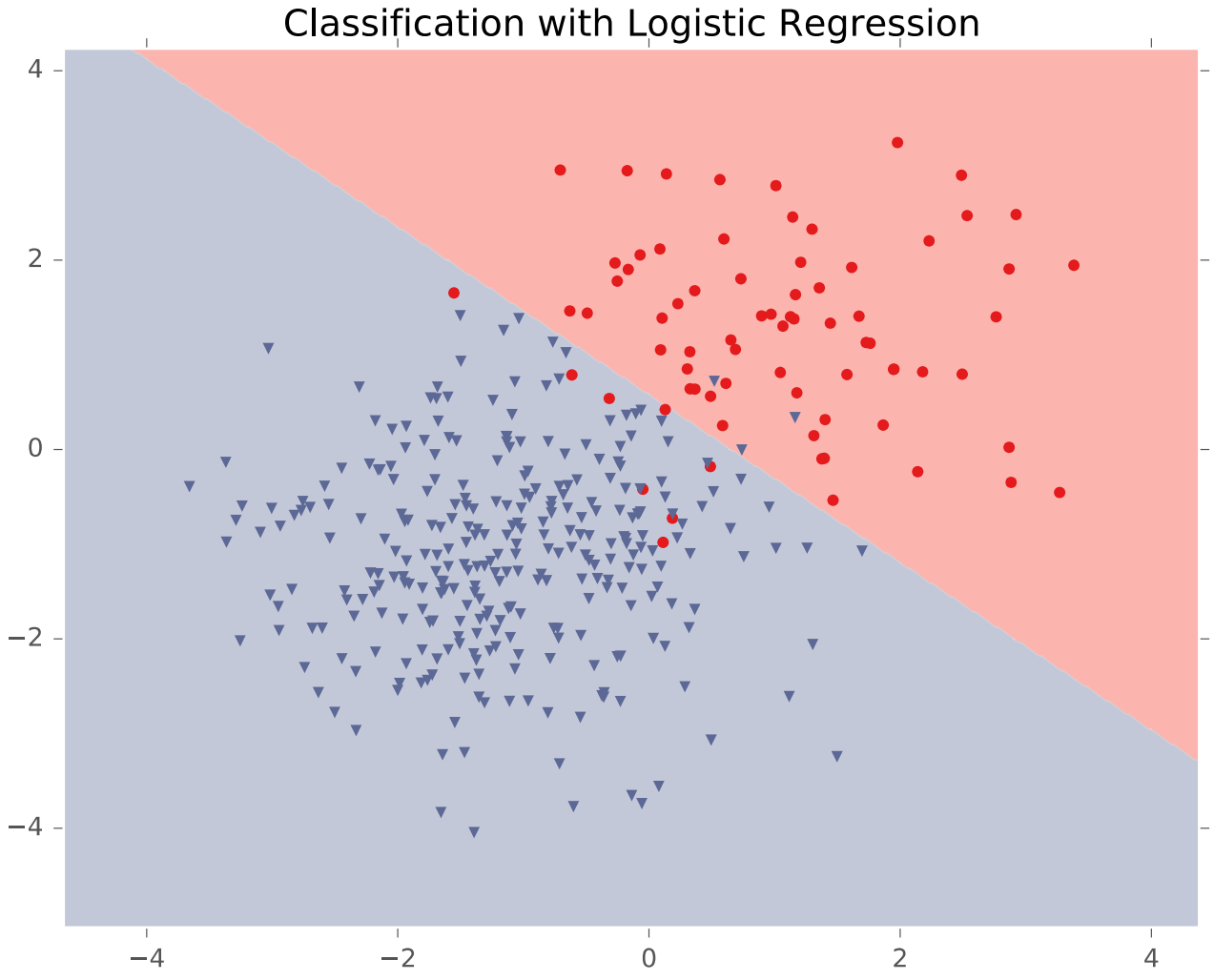
Slide credit: CMU MLD Matt Gormley

Building on a Linear Model



Slide credit: CMU MLD Matt Gormley

Logistic Regression Decision Boundary

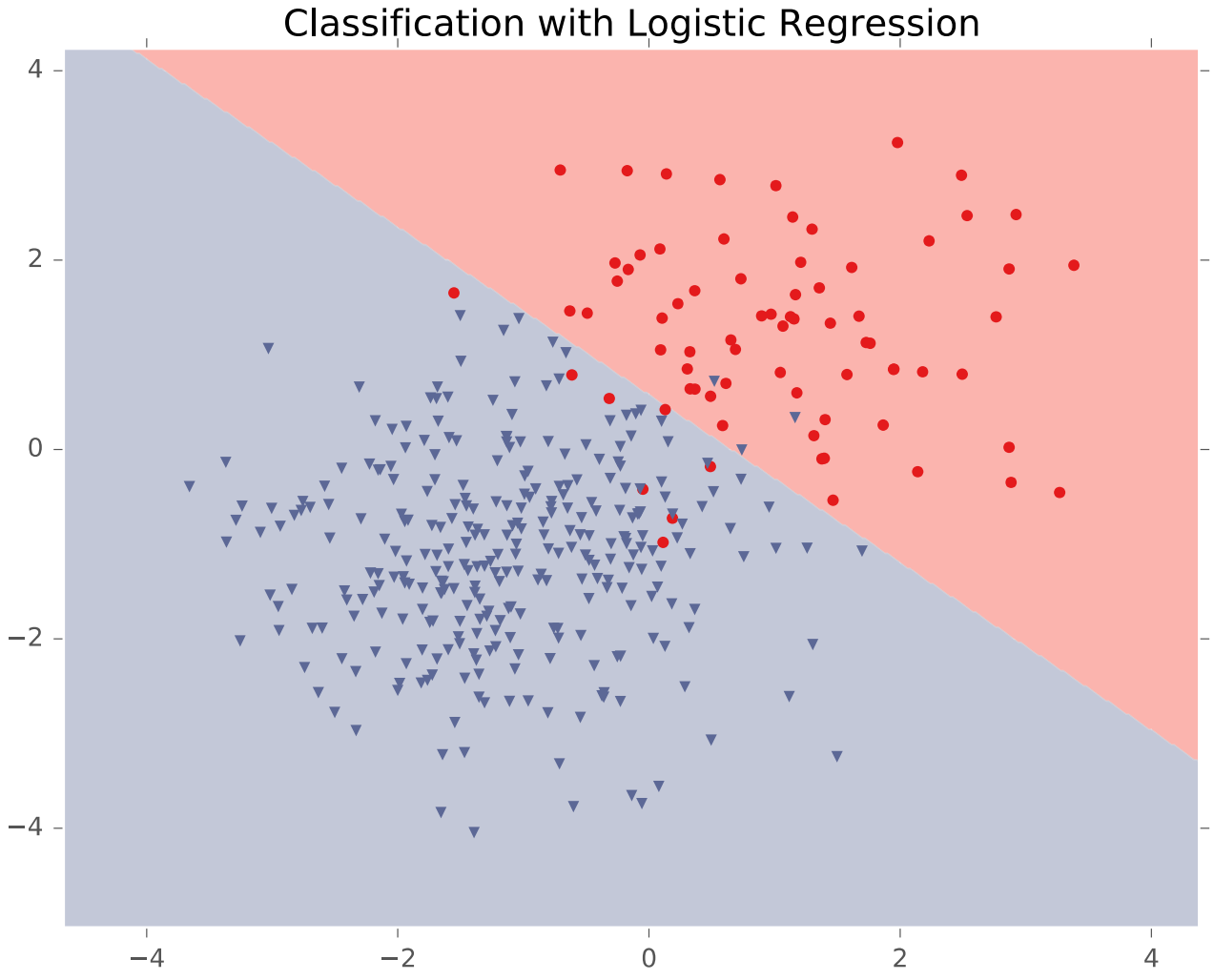


Slide credit: CMU MLD Matt Gormley

Logistic Regression

Linear decision boundary

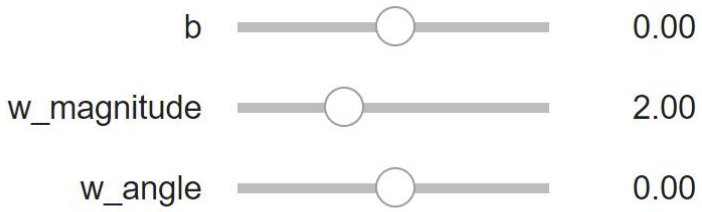
Logistic Regression Decision Boundary

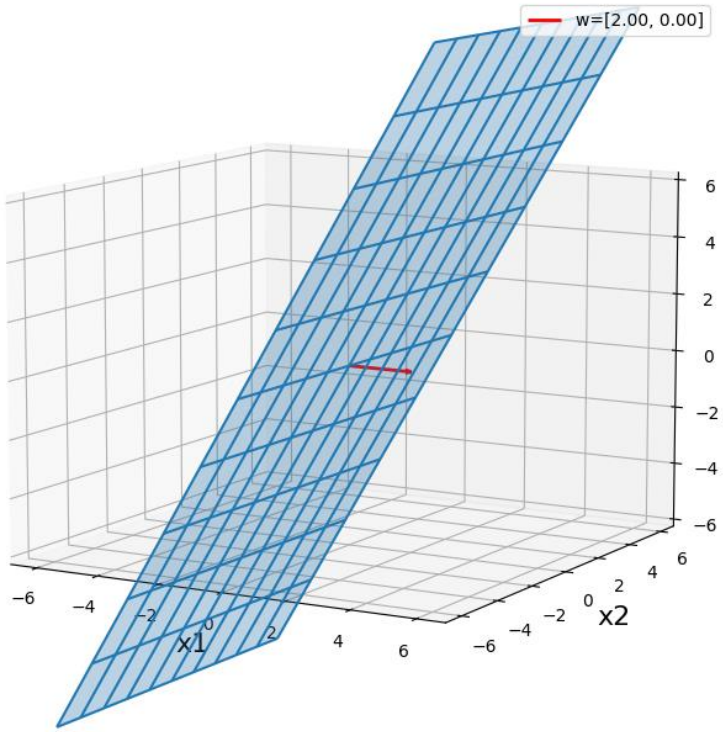


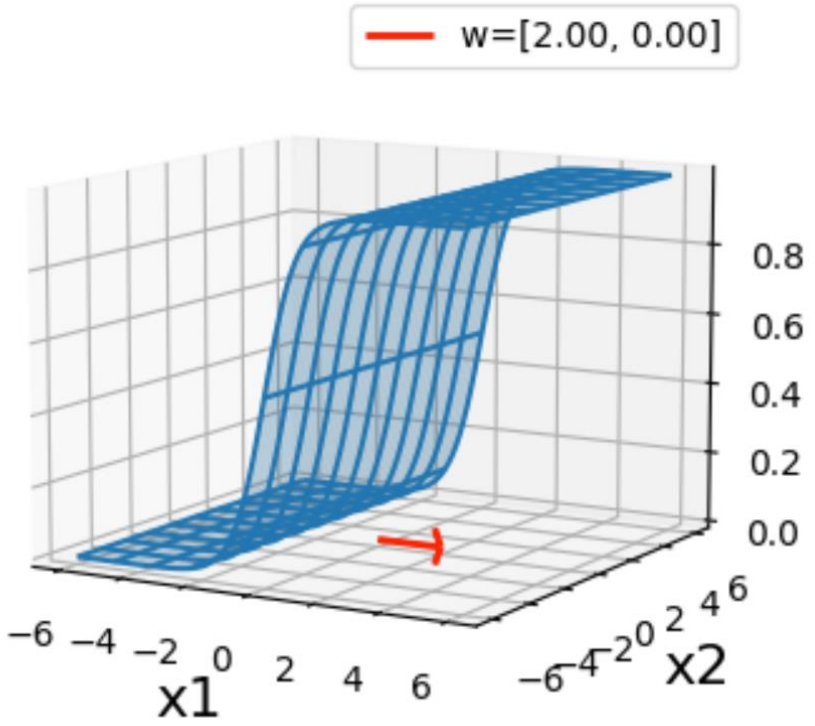
Slide credit: CMU MLD Matt Gormley

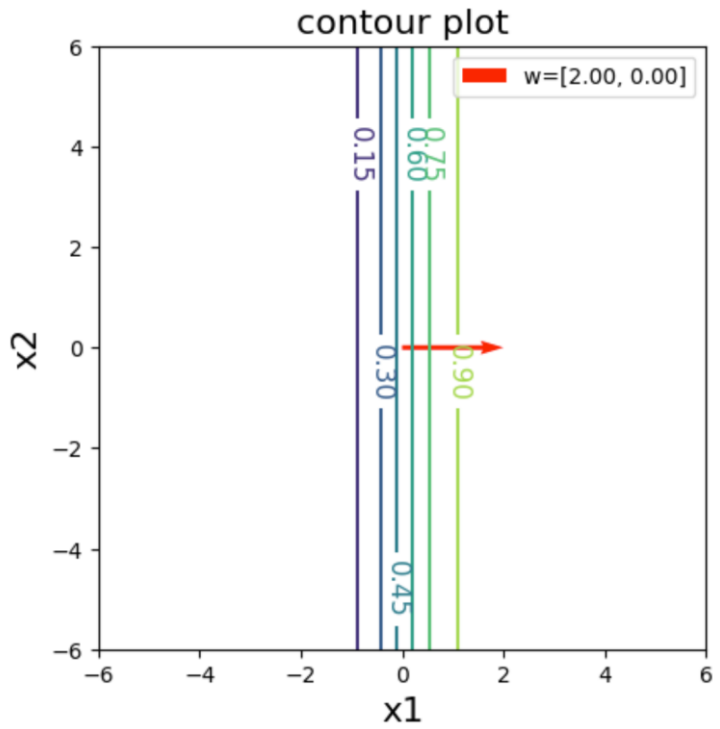
Exercise

Interact with the linear_logistic.ipynb posted on the course website schedule









Linear in Higher Dimensions

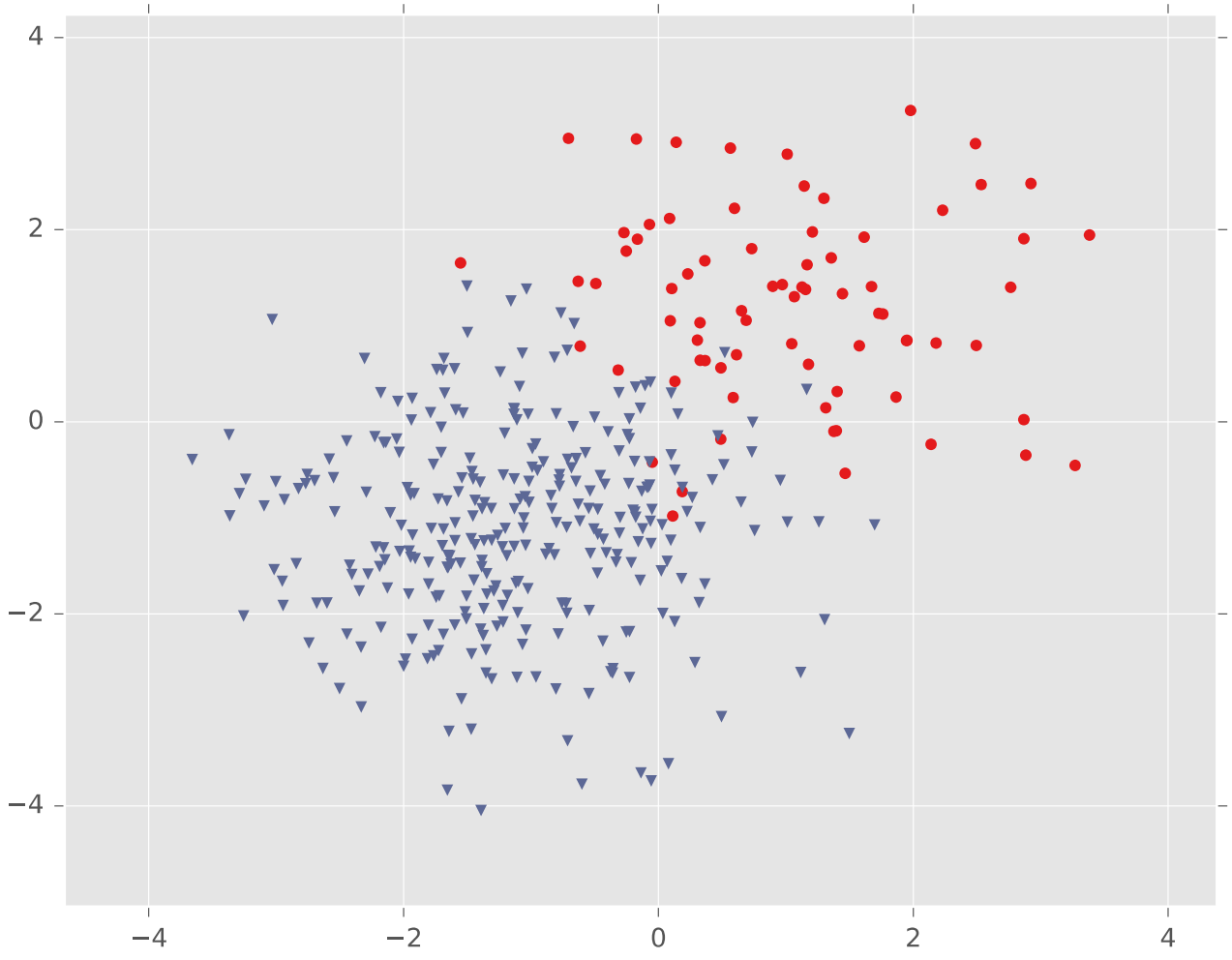
1-D $y = wx + b$

2-D $y = w_1 x_1 + w_2 x_2 + b$

What are these linear shapes called for 1-D, 2-D, 3-D, M-D input?

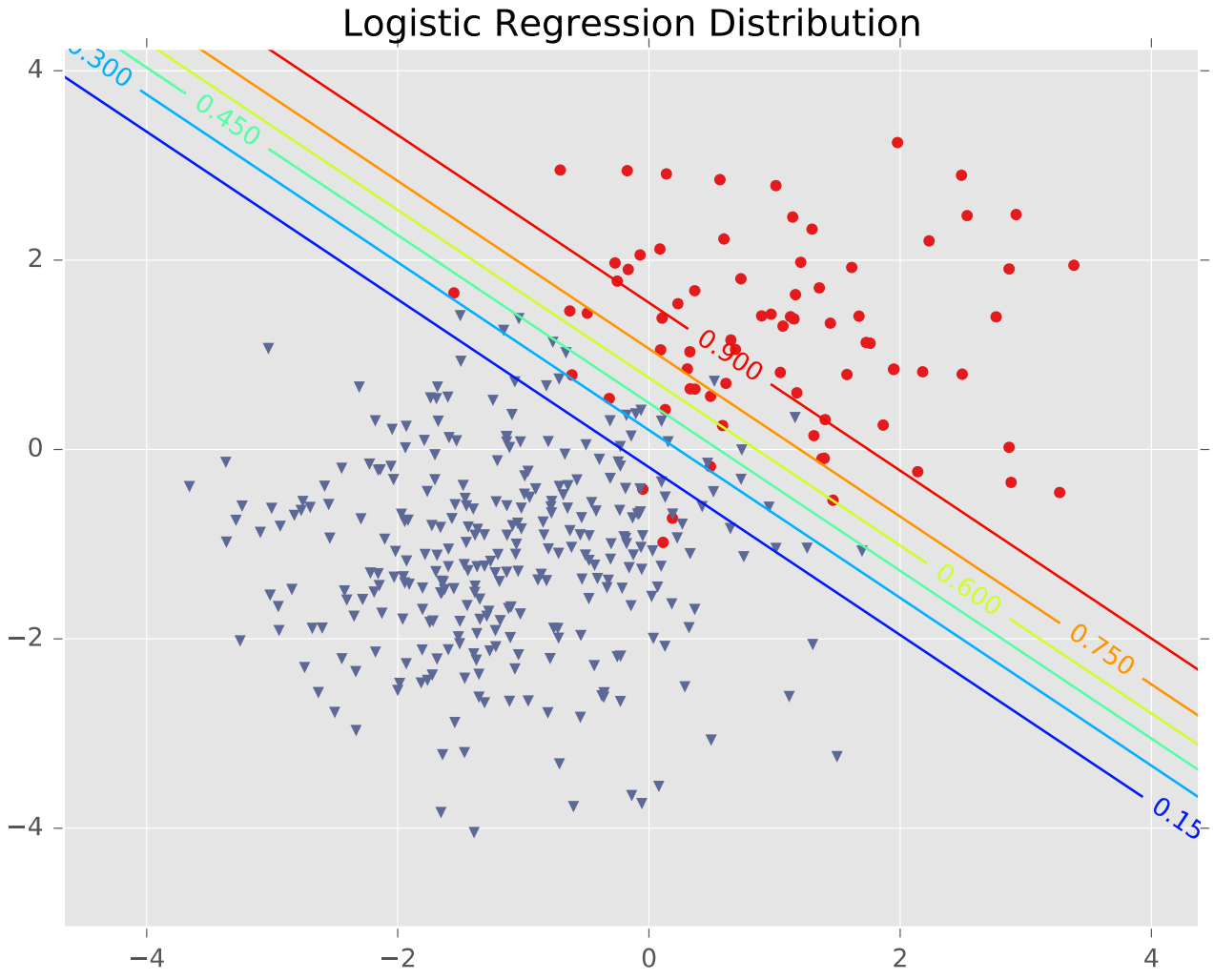
	$\boldsymbol{x} \in \mathbb{R}$	$\boldsymbol{x} \in \mathbb{R}^2$	$\boldsymbol{x} \in \mathbb{R}^3$	$\boldsymbol{x} \in \mathbb{R}^M$
$y = \boldsymbol{w}^T\boldsymbol{x} + b$	line	plane	hyperplane	hyperplane
$\boldsymbol{w}^T\boldsymbol{x} + b = 0$	point	line	plane	hyperplane
$\boldsymbol{w}^T\boldsymbol{x} + b \geq 0$	halfline	halfplane	halfspace	halfspace

Logistic Regression



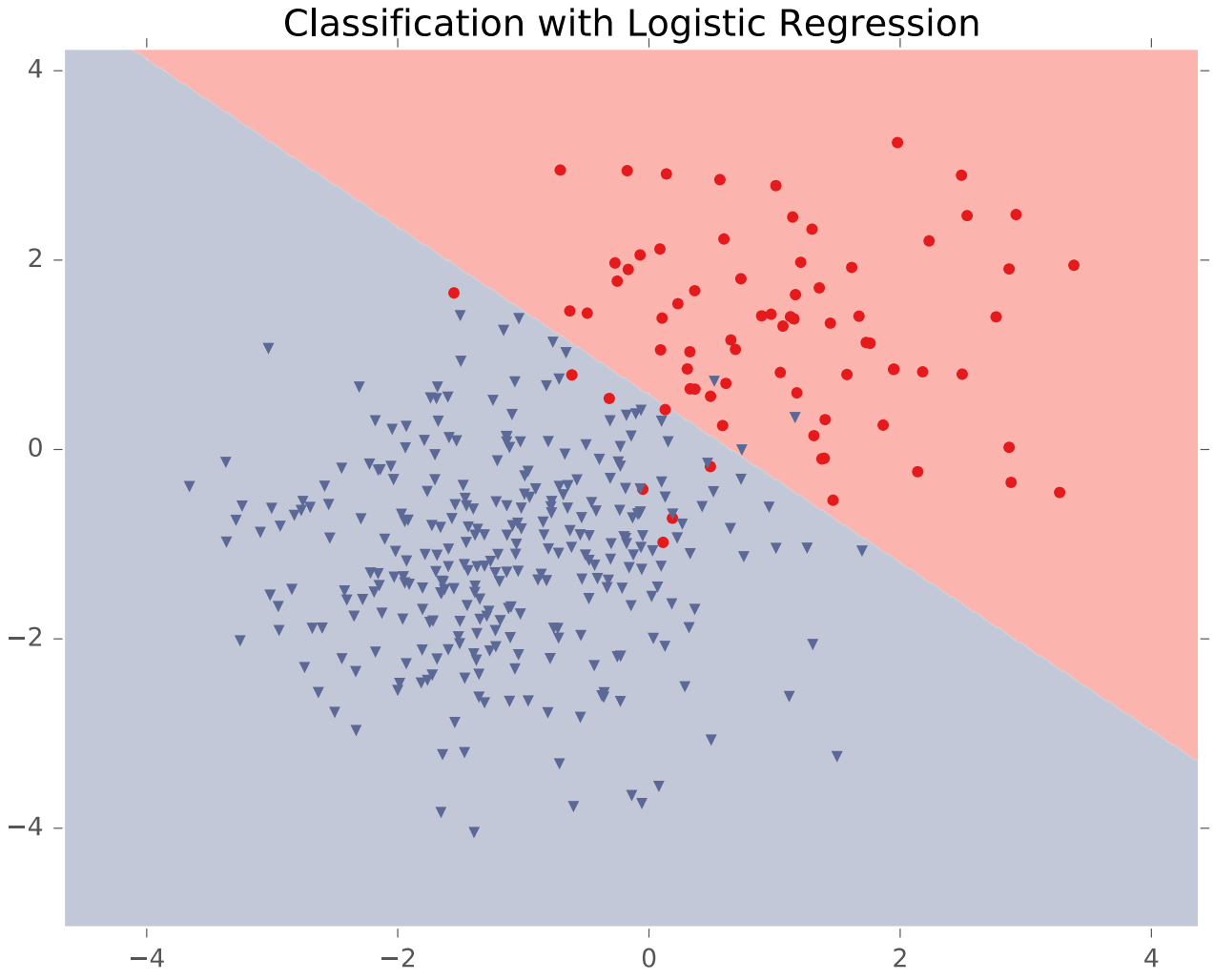
Slide credit: CMU MLD Matt Gormley

Logistic Regression



Slide credit: CMU MLD Matt Gormley

Logistic Regression



Slide credit: CMU MLD Matt Gormley

Poll 3

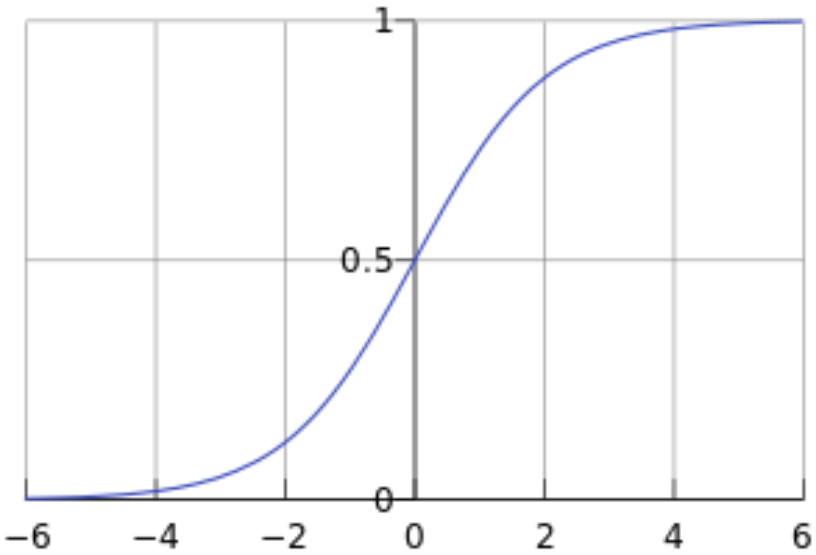
For a point $\mathbf{x}$ on the decision boundary of logistic regression, does $g(\mathbf{w}^T\mathbf{x} + b) = \mathbf{w}^T\mathbf{x} + b$?

A) Yes

B) No

C) I have no idea

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



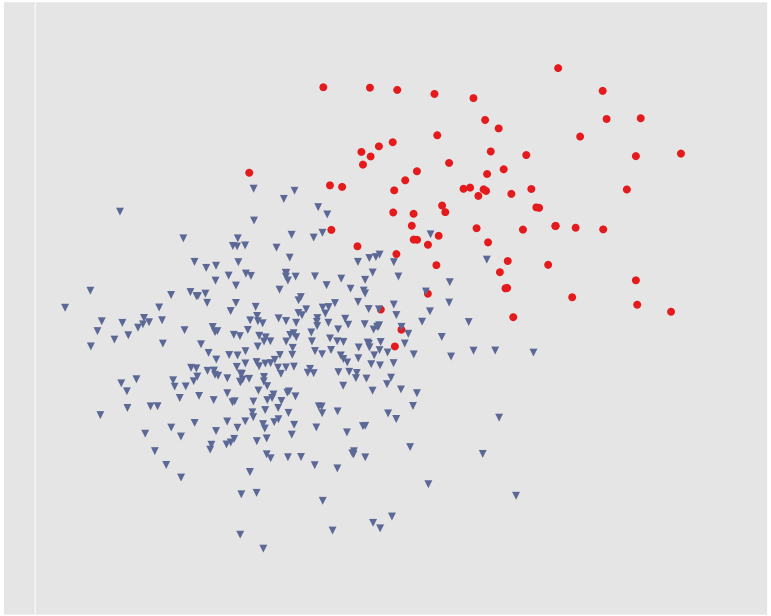
Logistic Regression

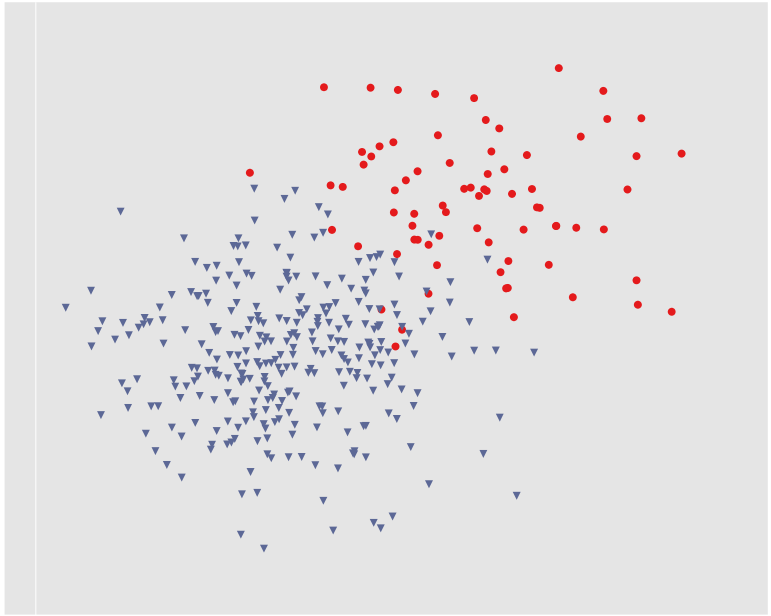
Optimization

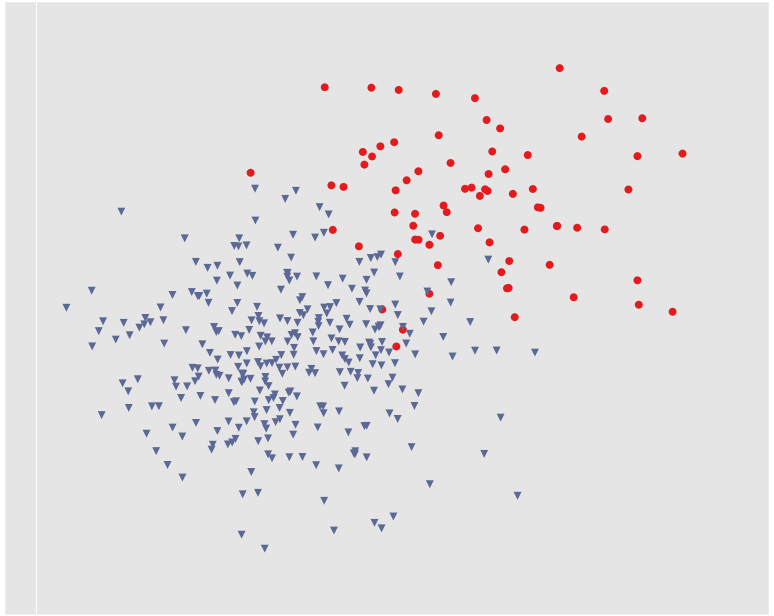
Optimizing a Model for Cancer Diagnosis

Learn to predict if a patient has cancer ($Y = 1$) or not ($Y = 0$) given the input of two test results, X_A, X_B. Note: bias term included in $\mathbf{x}$.

$$p(Y = 1 \mid \mathbf{x}, \boldsymbol{\theta}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^T \mathbf{x}}}$$





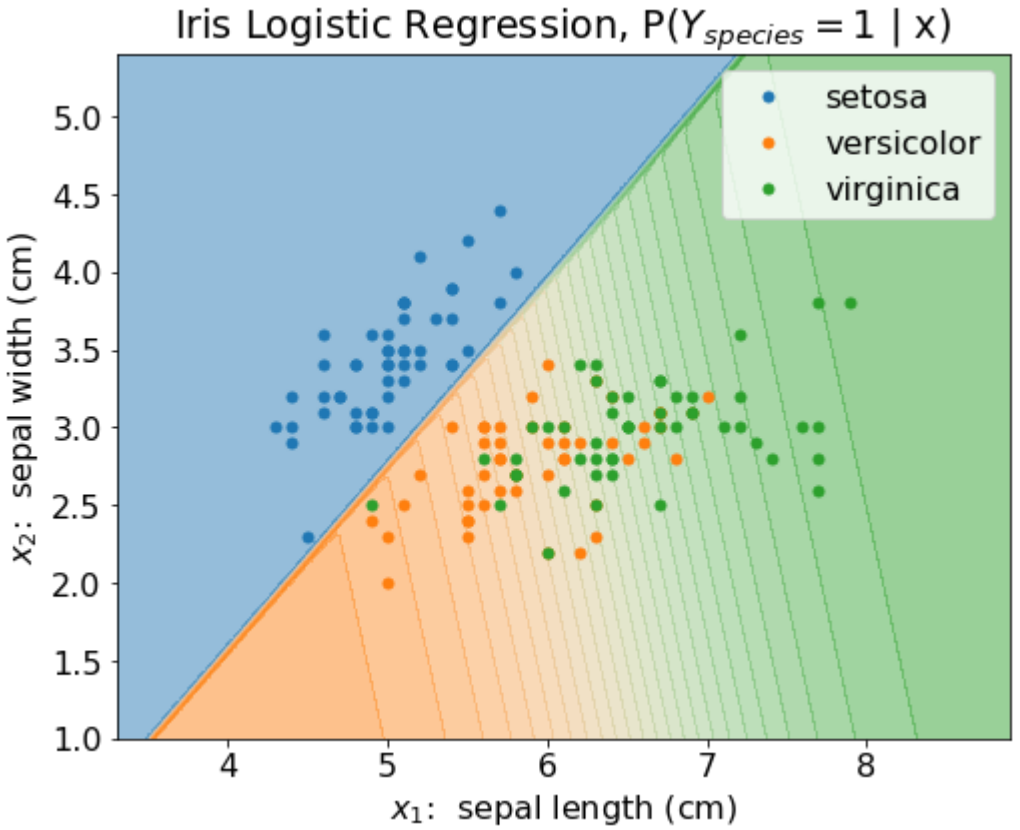


Empirical Risk Minimization

Still doing empirical risk minimization, just with a cross-entropy loss

$$h^* = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \hat{R}(h)$$

$$\hat{R}(h) = \frac{1}{N} \sum_{i=1}^{N} \ell \left(y^{(i)}, h \left(x^{(i)} \right) \right)$$



Empirical Risk Minimization

Still doing empirical risk minimization, just with a cross-entropy loss

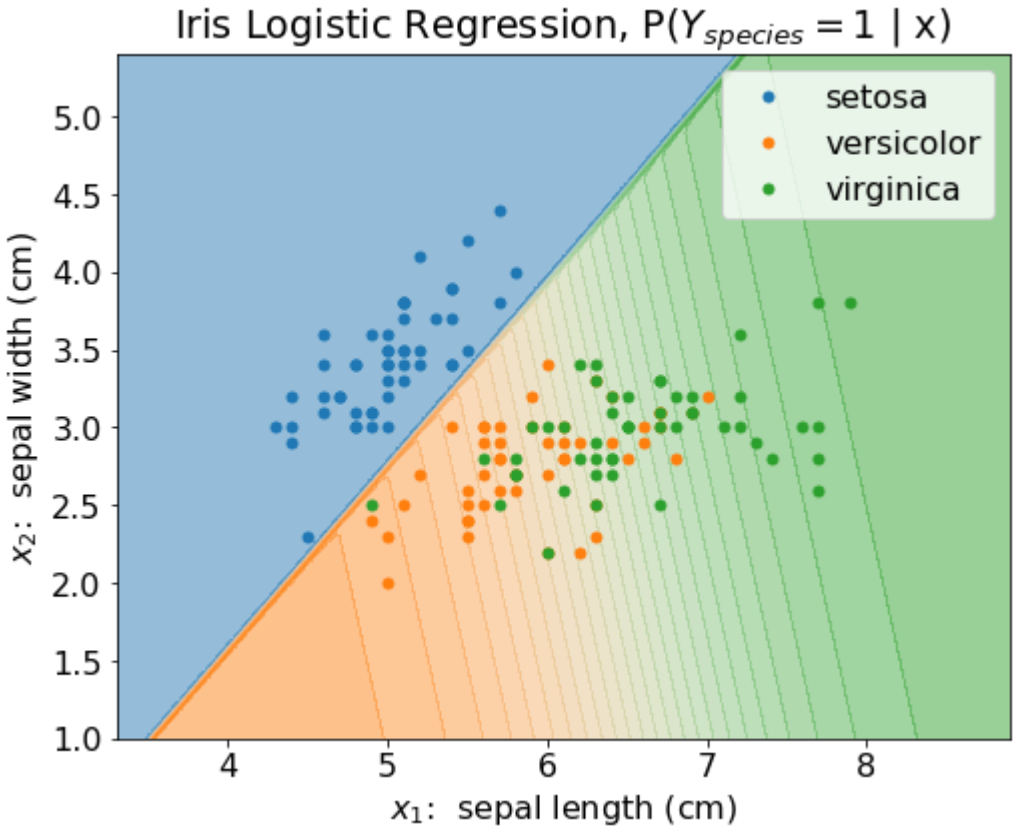
$$h^* = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \hat{R}(h)$$

$$\hat{R}(h) = \frac{1}{N} \sum_{i=1}^{N} \ell\left(y^{(i)}, h\left(x^{(i)}\right)\right)$$

Cross-entropy loss

$$\ell(\mathbf{y}, \hat{\mathbf{y}}) = -\sum_{k=1}^{K} y_k \log \hat{y}_k$$

But now we need a model $h_{\boldsymbol{\theta}}(\mathbf{x})$ that returns values that look like probabilities



Binary Logistic Regression

1) Model

2) Objective function

3) Solve for $\widehat{\boldsymbol{\theta}}$

Binary Logistic Regression

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

Objective: Special case for binary logistic regression

$$\hat{y} = g(\boldsymbol{\theta}^T \mathbf{x})$$

$$J(\boldsymbol{\theta}) = -\frac{1}{N} \sum_i \sum_k y_k^{(i)} \log y_k^{(i)}$$

$$= -\frac{1}{N} \sum_i \left(y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right) \right)$$

Solve Logistic Regression

$$\hat{y} = g(\boldsymbol{\theta}^T \mathbf{x}) \qquad g(z) = \frac{1}{1+e^{-z}} \qquad \frac{dg}{dz} = g(z)\big(1-g(z)\big)$$

$$J^{(i)}(\boldsymbol{\theta}) = -\big[y^{(i)} \log \hat{y}^{(i)} + \big(1-y^{(i)}\big) \log\big(1-\hat{y}^{(i)}\big)\big]$$

$$\frac{\partial J^{(i)}}{\partial \boldsymbol{\theta}} = -\big(y^{(i)} - \hat{y}^{(i)}\big)\, \mathbf{x}^{(i)}$$

Solve Logistic Regression

$$\hat{y} = g(\boldsymbol{\theta}^T \mathbf{x}) \qquad g(z) = \frac{1}{1+e^{-z}} \qquad \frac{dg}{dz} = g(z)\big(1-g(z)\big)$$

$$J^{(i)}(\boldsymbol{\theta}) = -\big[y^{(i)} \log \hat{y}^{(i)} + \big(1-y^{(i)}\big) \log\big(1-\hat{y}^{(i)}\big)\big]$$

$$\frac{\partial J^{(i)}}{\partial \boldsymbol{\theta}} = -\big(y^{(i)} - \hat{y}^{(i)}\big)\, \mathbf{x}^{(i)}$$

Solve Logistic Regression

$$\hat{y} = g(\boldsymbol{\theta}^T \mathbf{x}) \qquad g(z) = \frac{1}{1+e^{-z}}$$

$z = \theta^T x$ ← $\frac{d}{du} u^T v = v$ or v^T

$\hat{y} = g(z)$

$$\frac{dg}{dz} = g(z)\big(1 - g(z)\big)$$

$$J^{(i)}(\boldsymbol{\theta}) = -\left[y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right)\right] \quad \ell(y^{(i)}, \hat{y}^{(i)})$$

$$\frac{\partial J}{\partial \theta} = \frac{\partial \ell}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial \theta}$$

$$= -\left[\frac{y}{\hat{y}} - \frac{1-y}{1-\hat{y}}\right] \frac{\partial \hat{y}}{\partial z} \frac{\partial z}{\partial \theta}$$

$$= -\left[\frac{y}{\hat{y}} - \frac{1-y}{1-\hat{y}}\right] \hat{y}(1-\hat{y})\, \vec{x}$$

$$\frac{\partial J^{(i)}}{\partial \boldsymbol{\theta}} = -\left(y^{(i)} - \hat{y}^{(i)}\right) \mathbf{x}^{(i)}$$

Solve Logistic Regression

$z = \theta^T x$ ← $\frac{d}{du} u^T v = v$ or v^T

$\hat{y} = g(z)$

$\hat{y} = g(\boldsymbol{\theta}^T \mathbf{x})$ $\quad g(z) = \frac{1}{1+e^{-z}}$

$$\frac{dg}{dz} = g(z)\big(1 - g(z)\big)$$

$$J^{(i)}(\boldsymbol{\theta}) = -\left[y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right)\right] \quad \ell(y^{(i)}, \hat{y}^{(i)})$$

$$\frac{\partial J}{\partial \theta} = \frac{\partial \ell}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial \theta}$$

$$= -\left[\frac{y}{\hat{y}} - \frac{1-y}{1-\hat{y}}\right] \frac{\partial \hat{y}}{\partial z} \frac{\partial z}{\partial \theta}$$

$$= -\left[\frac{y}{\hat{y}} - \frac{1-y}{1-\hat{y}}\right] \hat{y}(1-\hat{y})\, \vec{x}$$

$$\frac{\partial J^{(i)}}{\partial \boldsymbol{\theta}} = -\left(y^{(i)} - \hat{y}^{(i)}\right) \mathbf{x}^{(i)}$$

Solve Logistic Regression

$\hat{y} = g(\boldsymbol{\theta}^T \mathbf{x})$ $\quad$ $g(z) = \frac{1}{1+e^{-z}}$

$$J(\boldsymbol{\theta}) = -\frac{1}{N}\sum_i \left(y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right)\right)$$

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = -\frac{1}{N}\sum_i \left(y^{(i)} - \hat{y}^{(i)}\right) \mathbf{x}^{(i)}$$

$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = 0$?

No closed form solution ☹

Back to iterative methods. Solve with (stochastic) gradient descent, Newton's method, or Iteratively Reweighted Least Squares (IRLS)

Solve Logistic Regression

$$\hat{y} = g(\boldsymbol{\theta}^T \mathbf{x}) \qquad g(z) = \frac{1}{1+e^{-z}}$$

$$J(\boldsymbol{\theta}) = -\frac{1}{N}\sum_i \left(y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right)\right)$$

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = -\frac{1}{N}\sum_i \left(y^{(i)} - \hat{y}^{(i)}\right) \mathbf{x}^{(i)}$$

No closed form solution ☹

Back to iterative methods. Solve with (stochastic) gradient descent, Newton's method, or Iteratively Reweighted Least Squares (IRLS)

Good news: The logistic regression optimization function is convex!

Logistic Regression

Convexity

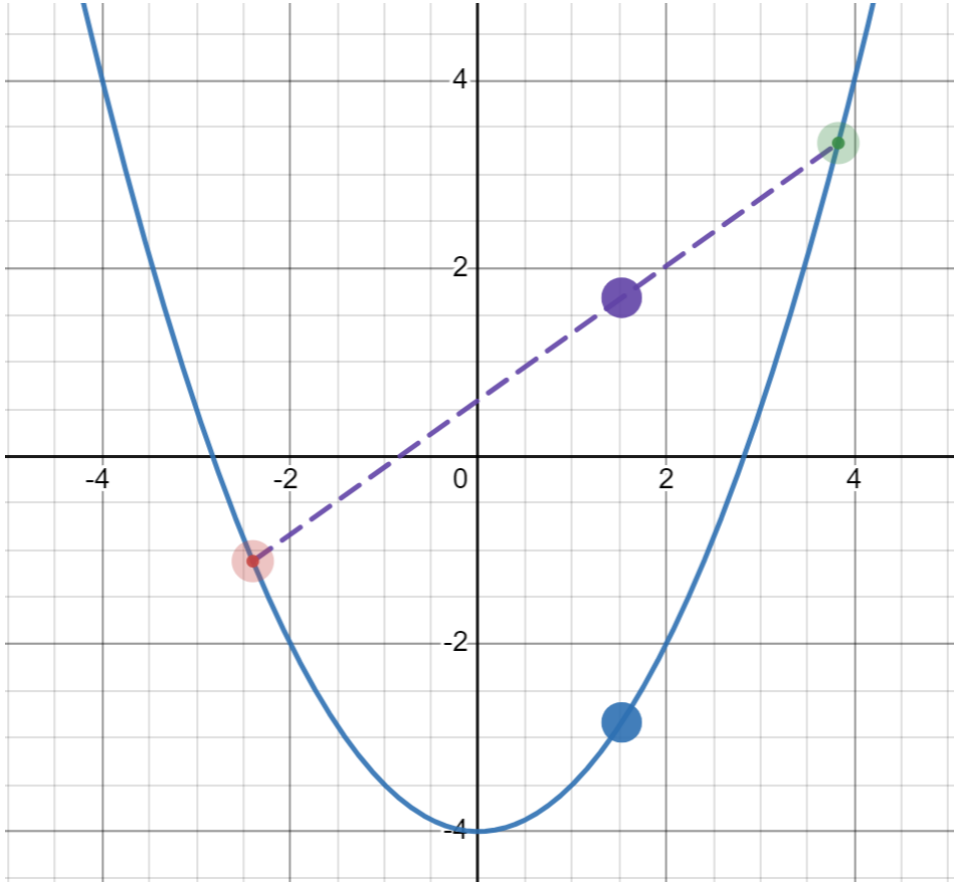
Optimization

Convex function

If $f(\boldsymbol{x})$ is convex, then:

- $f(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{z}) \leq \alpha f(\boldsymbol{x}) + (1-\alpha) f(\boldsymbol{z})$
  $\forall\ 0 \leq \alpha \leq 1$

[Demo on Desmos]



Optimization

Convex function

If $f(\boldsymbol{x})$ is convex, then:

- $f(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{z}) \leq \alpha f(\boldsymbol{x}) + (1-\alpha) f(\boldsymbol{z}) \quad \forall\, 0 \leq \alpha \leq 1$

Convex optimization

If second derivative is ≥ 0 everywhere then function is convex

If $f(\boldsymbol{x})$ is convex, then:

- Every local minimum is also a global minimum ☺

Optimization

$h(\mathbf{x}) = g(\mathbf{w}^\top \mathbf{x} + b)$ is definitely not convex

But...what are we optimizing over in logistic regression?

$$J(\boldsymbol{\theta}) = -\frac{1}{N}\sum_i \sum_k y_k^{(i)} \log y_k^{(i)}$$

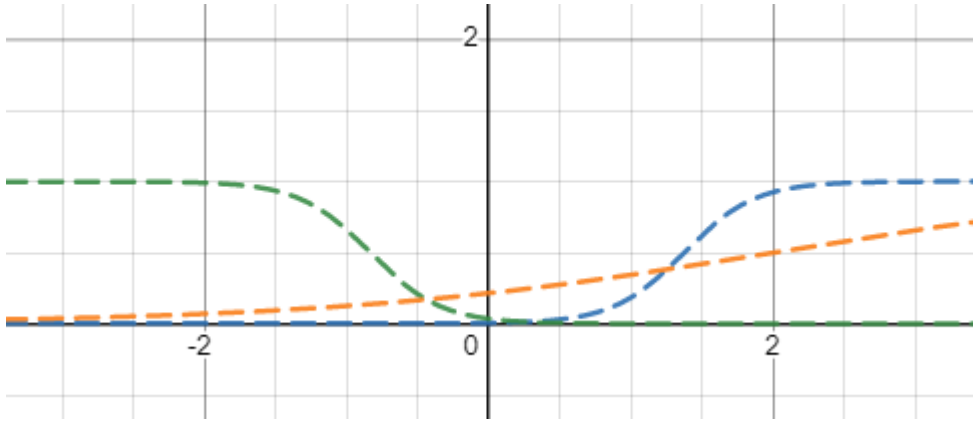
$$= -\frac{1}{N}\sum_i \left(y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right)\right)$$

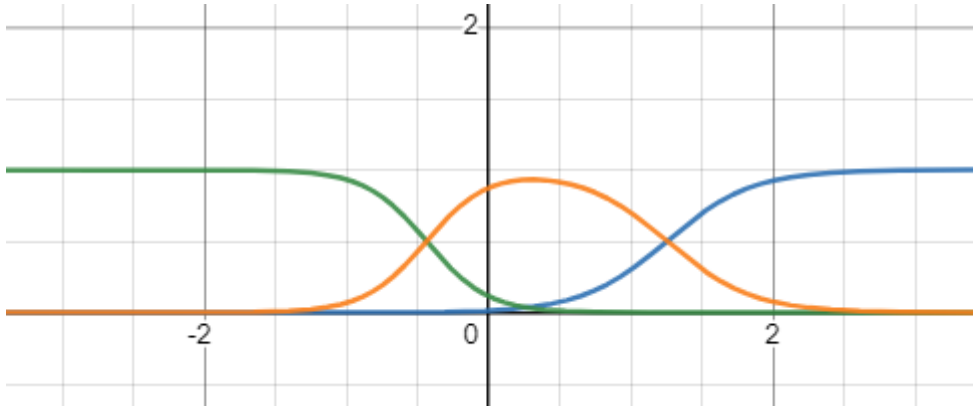
Multi-class Logistic Regression

Multi-class Logistic Regression

Desmos Demo:

https://www.desmos.com/calculator/53bautbxjp





Multi-class Logistic Regression

Cross-entropy loss

$$\ell(\boldsymbol{y}, \hat{\mathbf{y}}) = -\sum_{k=1}^{K} y_k \log \hat{y}_k$$

Model

$$\hat{\mathbf{y}} = h(\mathbf{x}) = g_{softmax}(\mathbf{z})$$

$$\mathbf{z} = \Theta\mathbf{x}$$

One vector of parameters for each class

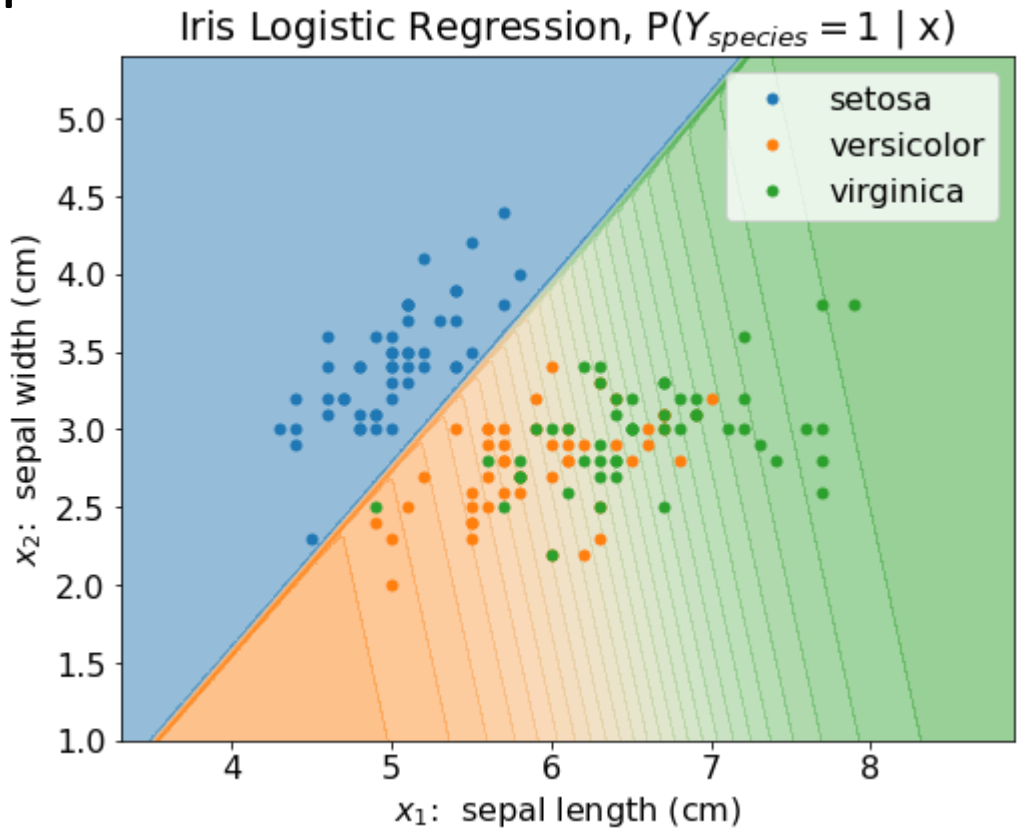
$$z_k = \boldsymbol{\theta}_k\mathbf{x}$$

$$\mathbf{x} = \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix}$$

$$\boldsymbol{\theta}_k = \begin{bmatrix} b_k \\ w_{k,1} \\ w_{k,2} \end{bmatrix}$$

Stacked into a matrix of $K \times M$ parameters

$$\Theta = \begin{bmatrix} - & \boldsymbol{\theta}_1^\top & - \\ - & \boldsymbol{\theta}_2^\top & - \\ - & \boldsymbol{\theta}_3^\top & - \end{bmatrix} = \begin{bmatrix} b_1 & w_{1,1} & w_{1,2} \\ b_2 & w_{2,1} & w_{2,2} \\ b_3 & w_{3,1} & w_{3,2} \end{bmatrix}$$



Logistic Function

Logistic (sigmoid) function converts value from $(-\infty, \infty) \rightarrow (0, 1)$

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

$g(z)$ and $1 - g(z)$ sum to one

Example 2 $\rightarrow g(2) = 0.88, \quad 1\text{-}g(2) = 0.12$

Softmax Function

Softmax function convert each value in a vector of values from $(-\infty, \infty) \rightarrow (0, 1)$, such that they all sum to one.

$$g(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_K \end{bmatrix} \rightarrow \begin{bmatrix} e^{z_1} \\ e^{z_2} \\ \vdots \\ e^{z_K} \end{bmatrix} \cdot \frac{1}{\sum_{k=1}^{K} e^{z_k}}$$

Example $\begin{bmatrix} -1 \\ 4 \\ 1 \\ -2 \\ 3 \end{bmatrix} \rightarrow \begin{bmatrix} 0.0047 \\ 0.7008 \\ 0.0349 \\ 0.0017 \\ 0.2578 \end{bmatrix}$

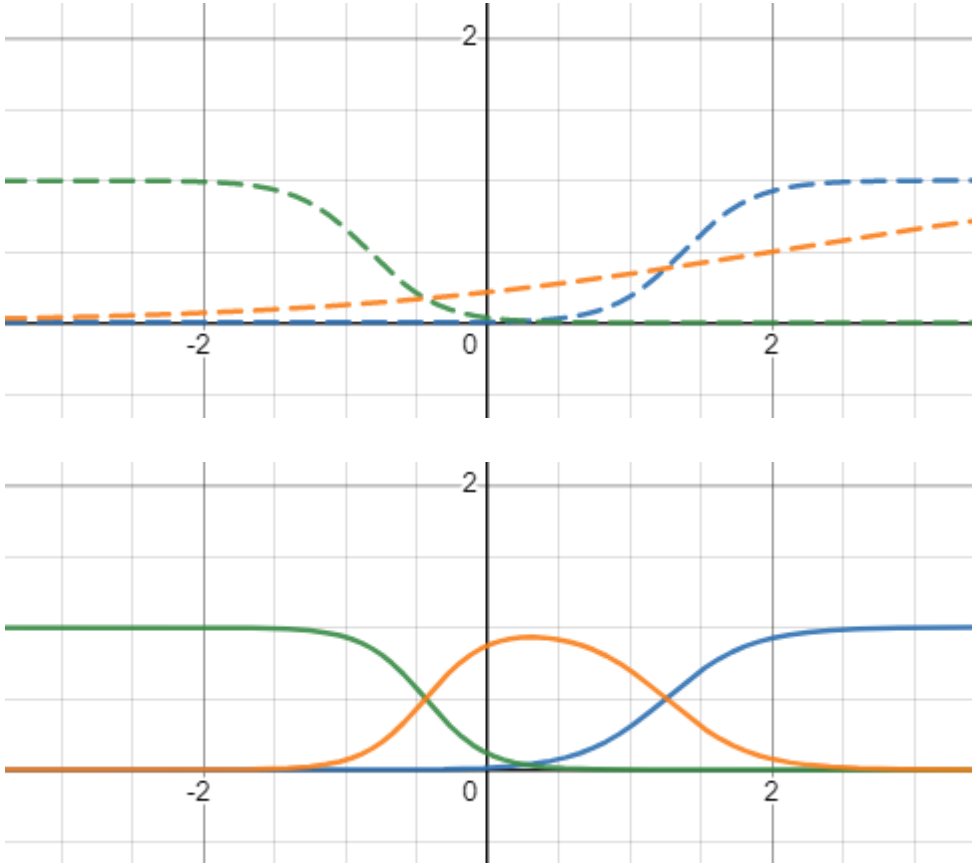
Multiclass Predicted Probability

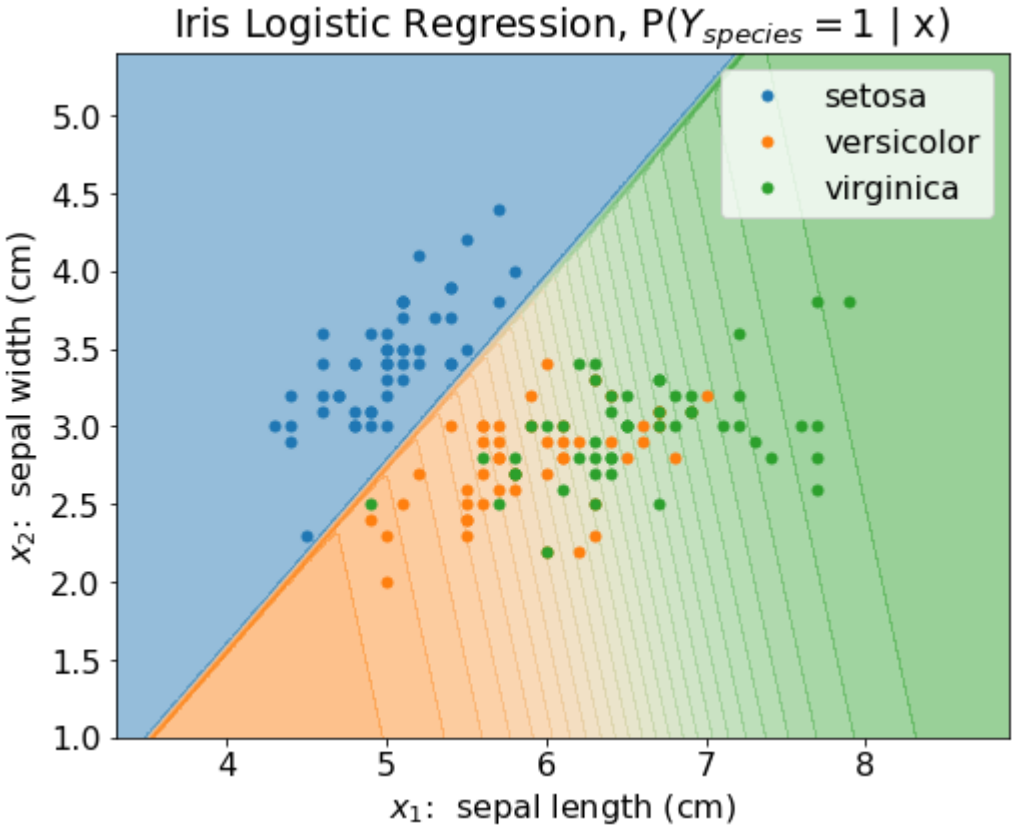
Multiclass logistic regression uses the parameters learned across all K classes to predict the discrete conditional probability distribution of the output Y given a specific input vector $\mathbf{x}$

$$\begin{bmatrix} p(Y=1 \mid \mathbf{x}, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \\ p(Y=2 \mid \mathbf{x}, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \\ p(Y=3 \mid \mathbf{x}, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \end{bmatrix} = \begin{bmatrix} e^{\boldsymbol{\theta}_1^T \mathbf{x}} \\ e^{\boldsymbol{\theta}_2^T \mathbf{x}} \\ e^{\boldsymbol{\theta}_3^T \mathbf{x}} \end{bmatrix} \cdot \frac{1}{\sum_{k=1}^{K} e^{\boldsymbol{\theta}_k^T \mathbf{x}}}$$

Multiclass Predicted Probability

Multiclass logistic regression uses the parameters learned across all K classes to predict the discrete conditional probability distribution of the output Y given a specific input vector $\mathbf{x}$

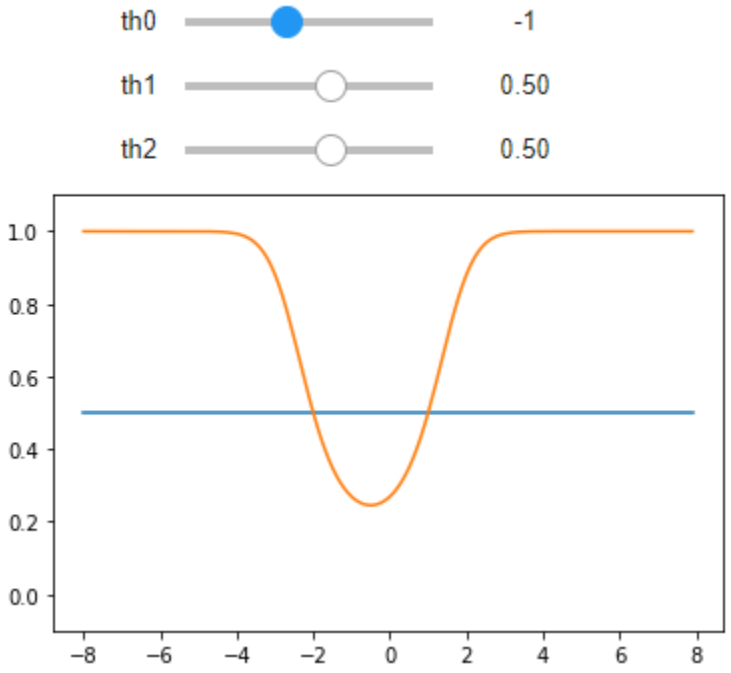




Logistic Regression with Polynomial Features

Exercise

Interact with the logistic_quadratic.ipynb posted on the course website schedule



Exercise

Interact with the logistic_quadratic.ipynb posted on the course website schedule