

Lecture 08

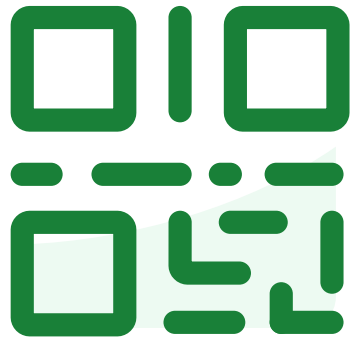
Bias-Variance Trade-off & Logistic Regression (1)

Bias-Variance Trade-off & Introduction to Logistic Regression

EECS 189/289, Fall 2026 @ UC Berkeley

Joseph E. Gonzalez and Narges Norouzi

Do not edit
How to change the
design



**Join at slido.com
#1965716**

① The Slido app must be installed on every computer you're presenting from

slido

Roadmap

- Bias-Variance Trade-off
- Classification Task
- Linear Discriminant Functions
- Discriminative Probabilistic Models
- The Logistic Regression Model
- Logistic Regression Objective



Bias-Variance Trade-off

- Bias-Variance Trade-off
- Classification Task
- Linear Discriminant Functions
- Discriminative Probabilistic Models
- The Logistic Regression Model
- Logistic Regression Objective



Fundamental Challenges in Learning?



1965716

- ***Fit the Data***
 - Provide an explanation for what we observe
- ***Generalize to the World***
 - Predict the future
 - Explain the unobserved



Is this cat grumpy or are we
overfitting to human faces?



1965716

Fundamental Challenges in Learning?

- **Bias:** The expected deviation between the predicted value and the true value.
- **Variance:** Two sources
 - **Noise:** The variability of the random noise in the process we are trying to model.
 - **Model Variance:** The variability in the predicted value across different training datasets.

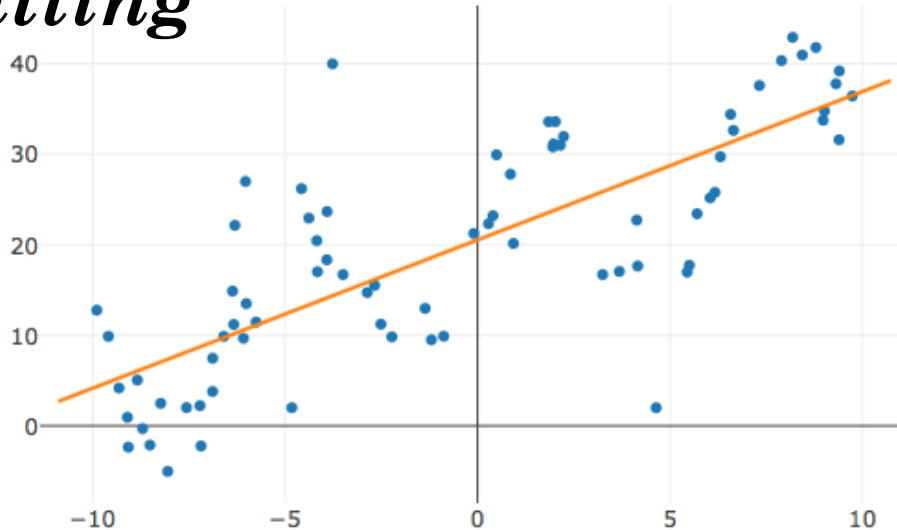
Bias



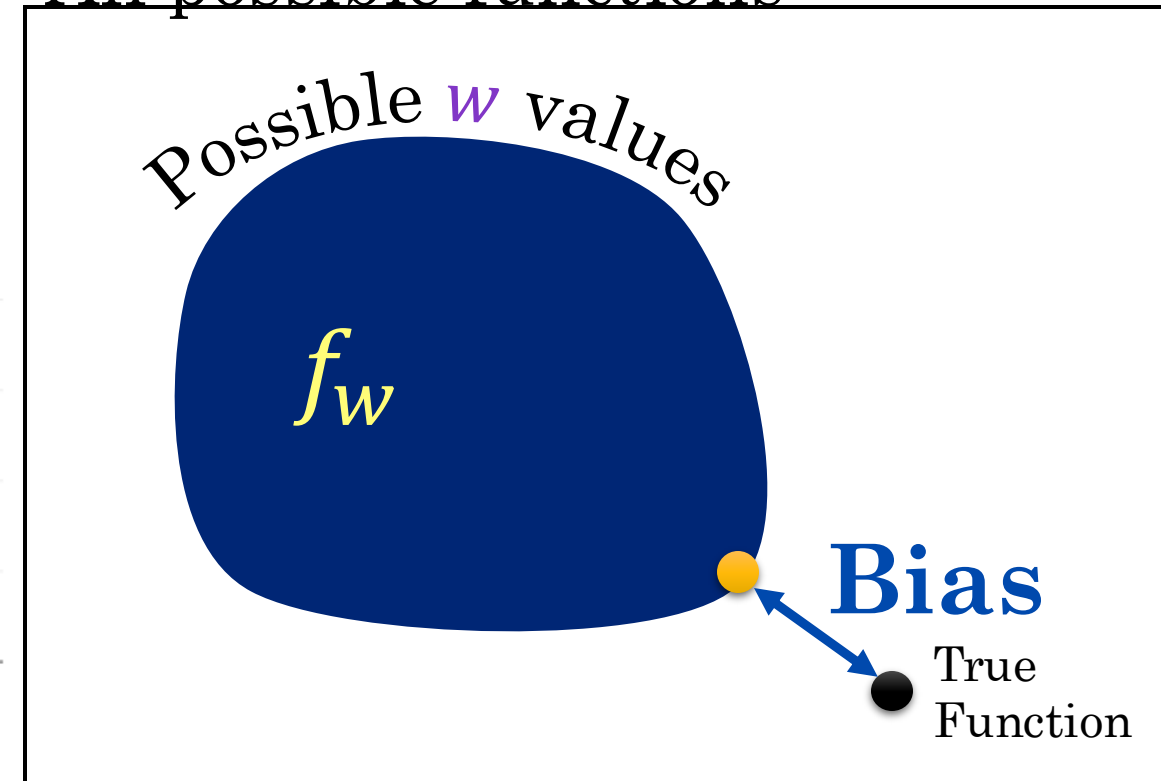
1965716

The expected deviation between the predicted value and the true value

- Depends on both the:
 - Choice of f
 - Learning procedure
- *Under-fitting*



All possible functions



Noise

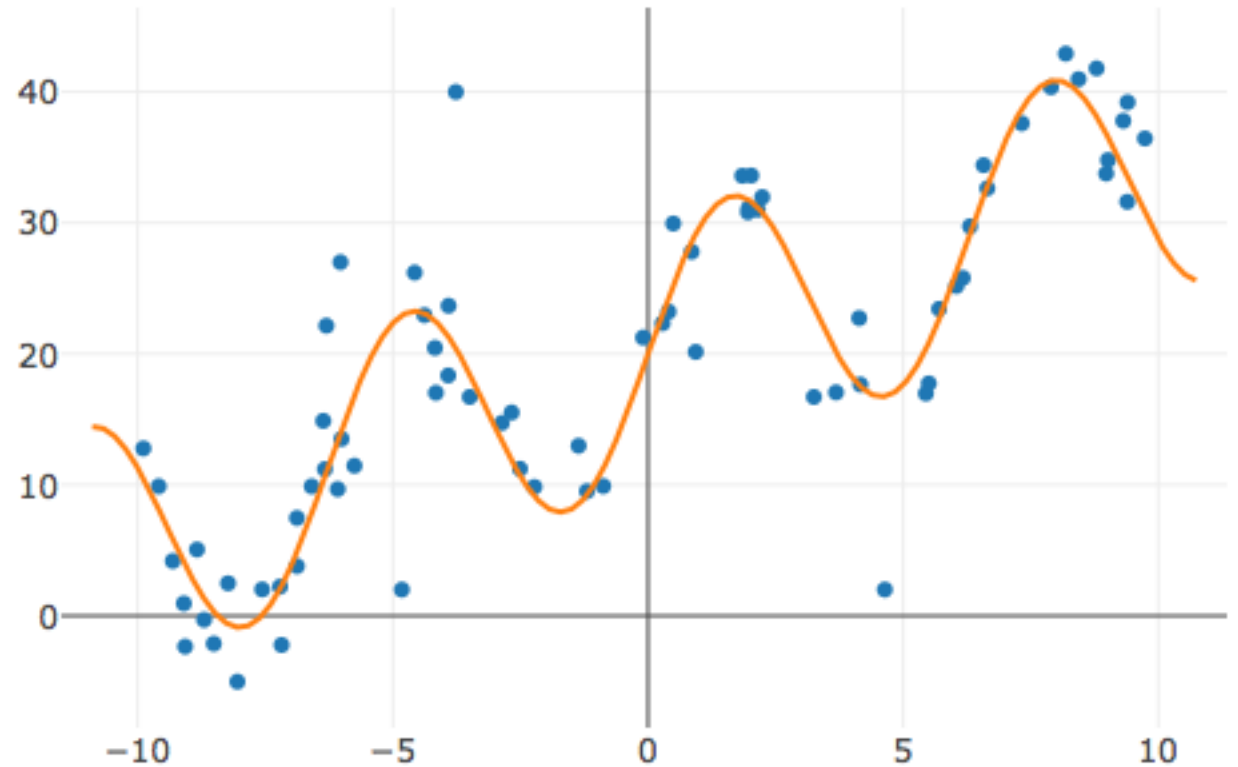


1965716

The variability of the random noise in the process we are trying to model.

- Measurement variability
- Stochasticity
- Missing information

**Beyond our control
(usually)**



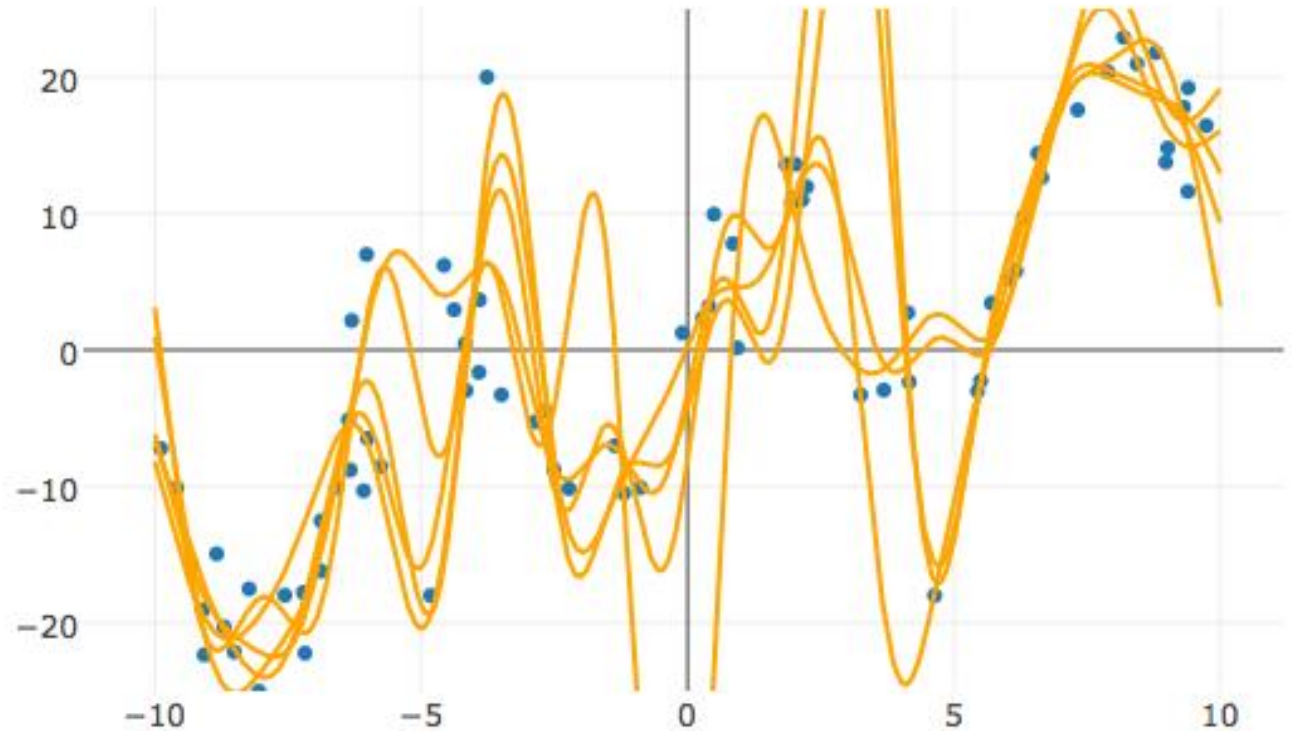


1965716

Model Variance

Variability in the predicted value across different training datasets.

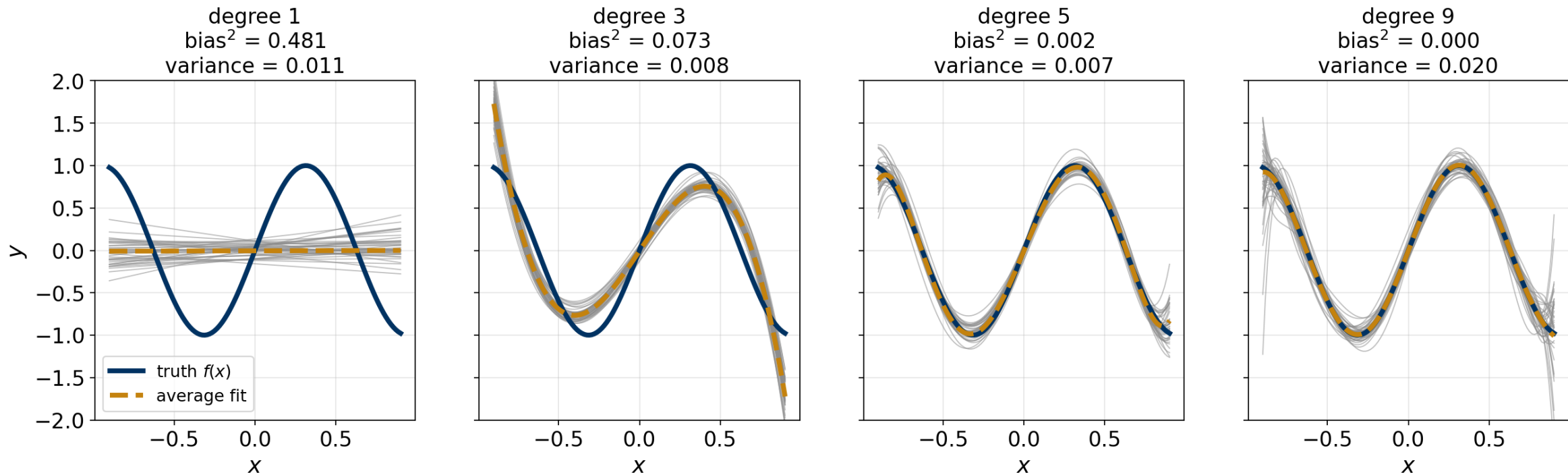
- Sensitivity to variation in the training data
- Poor generalization
- **Overfitting**



Fitted Functions Across 40 Training Sets



1965716



Each grey curve is one fit from one training set of 100 points; the dashed line is their average. Bias is the gap between the dashed line and the truth; variance is the width of the grey band.



**Which of the following models
have a higher variance?**

Analysis of Squared Error



1965716

- For the test point $\mathbf{x}$ the expected error:
 - *Random variables are red*

Assume noisy observations
 $\rightarrow t$ is a random variable

True Function

$$t = h(\mathbf{x}) + \epsilon$$

Noise term:

$$E(\epsilon) = 0$$

$$\text{Var}(\epsilon) = \sigma^2$$

$$E[(t - f_w(\mathbf{x}))^2]$$

Can be any
parametric function

Assume *training data* is
random
 $\rightarrow w$ is a random variable

Analysis of Squared Error



1965716

Goal:

$$\mathbb{E}[(t - f_w(\mathbf{x}))^2] =$$

“Noise” + (Bias)² + Model Variance



1965716

$$E[(t - f_w(\mathbf{x}))^2] = E[(t - \underbrace{h(\mathbf{x}) + h(\mathbf{x})}_{\text{Subtracting and adding } h(\mathbf{x})} - f_w(\mathbf{x}))^2]$$

Subtracting and adding $h(\mathbf{x})$

Useful Equations:

$$t = h(\mathbf{x}) + \epsilon$$

$$E(\epsilon) = 0$$

$$\text{Var}(\epsilon) = \sigma^2$$



1965716

$$E[(t - f_w(\mathbf{x}))^2] = E[\underbrace{(t - h(\mathbf{x}))}_a + \underbrace{h(\mathbf{x}) - f_w(\mathbf{x}))}_b]^2]$$

Expanding in terms of a and b : $(a + b)^2 = a^2 + b^2 + 2ab$

$$\begin{aligned} &= E[(t - h(\mathbf{x}))^2] + E[(h(\mathbf{x}) - f_w(\mathbf{x}))^2] \\ &\quad + 2 E[\underbrace{(t - h(\mathbf{x}))}_{\epsilon} \underbrace{(h(\mathbf{x}) - f_w(\mathbf{x}))}_{2ab}] \\ &\quad + 2 E[\epsilon (h(\mathbf{x}) - f_w(\mathbf{x}))] \end{aligned}$$

Useful Equations:

$$t = h(\mathbf{x}) + \epsilon$$

$$E(\epsilon) = 0$$

$$\text{Var}(\epsilon) = \sigma^2$$



1965716

$$E[(t - f_w(\mathbf{x}))^2] = E[\underbrace{(t - h(\mathbf{x}))}_a + \underbrace{h(\mathbf{x}) - f_w(\mathbf{x}))}_b]^2]$$

Expanding in terms of a and b : $(a + b)^2 = a^2 + b^2 + 2ab$

$$= E[(t - h(\mathbf{x}))^2] + E[(h(\mathbf{x}) - f_w(\mathbf{x}))^2] + 2 E[\epsilon (h(\mathbf{x}) - f_w(\mathbf{x}))]$$

Independence of ϵ and w

$$+ 2 \underbrace{E[\epsilon]}_0 E[(h(\mathbf{x}) - f_w(\mathbf{x}))]$$

Useful Equations:

$$t = h(\mathbf{x}) + \epsilon$$

$$E(\epsilon) = 0$$

$$\text{Var}(\epsilon) = \sigma^2$$



1965716

$$E[(t - f_w(\mathbf{x}))^2]$$

$$= \sigma^2$$

$$= E[(t - h(\mathbf{x}))^2] +$$

Obs. Value

True Value

$$E[(h(\mathbf{x}) - f_w(\mathbf{x}))^2]$$

True Value

Pred. Value

“*Noise*” Term

Model
Estimation
Error

Useful Equations:

$$t = h(\mathbf{x}) + \epsilon$$

$$E(\epsilon) = 0$$

$$Var(\epsilon) = \sigma^2$$



1965716

$$E[(t - f_w(\mathbf{x}))^2]$$

$$= E[(t - h(\mathbf{x}))^2] + \sigma^2$$

“*Noise*” Term

$$E[(h(\mathbf{x}) - f_w(\mathbf{x}))^2]$$

Model
Estimation
Error

We need to calculate
this term

Useful Equations:

$$t = h(\mathbf{x}) + \epsilon$$

$$E(\epsilon) = 0$$

$$\text{Var}(\epsilon) = \sigma^2$$

$$E[(t - f_w(\mathbf{x}))^2]$$

$$= E[(t - h(\mathbf{x}))^2] + \sigma^2$$

“*Noise*” Term

$$E[(h(\mathbf{x}) - f_w(\mathbf{x}))^2]$$

Model
Estimation
Error

We need to calculate
this term

Useful Equations:

$$t = h(\mathbf{x}) + \epsilon$$

$$E(\epsilon) = 0$$

$$Var(\epsilon) = \sigma^2$$



1965716

$$\mathbb{E} \left[\left(h(\mathbf{x}) - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] = \text{Next we will show...}$$

$$\underbrace{\left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)^2}_{(\text{Bias})^2} + \underbrace{\mathbb{E} \left[\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right]}_{\text{Model Variance}}$$

➤ How?

➤ Adding and subtracting what?



1965716

$$\mathbb{E} \left[\left(h(\mathbf{x}) - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] =$$

$$\mathbb{E} \left[\underbrace{\left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)}_a + \underbrace{\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right)}_b \right]^2$$

Expanding in terms of a and b : $(a + b)^2 = a^2 + b^2 + 2ab$

$$\begin{aligned} & \mathbb{E} \left[\left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)^2 \right] + \mathbb{E} \left[\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] + \\ & 2\mathbb{E} \left[\left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right) \left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right) \right] \end{aligned}$$



1965716

$$\begin{aligned} \mathbb{E} \left[\left(h(\mathbf{x}) - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] &= \\ \mathbb{E} \left[\left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)^2 \right] &+ \mathbb{E} \left[\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] + \\ 2\mathbb{E} \left[\underbrace{\left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)}_{\text{constant}} \left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right) \right] & \\ \underbrace{2 \left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right) \mathbb{E} \left[\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right) \right]}_{\text{constant}} & \end{aligned}$$



1965716

$$\begin{aligned} \mathbb{E} \left[\left(h(\mathbf{x}) - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] &= \\ \mathbb{E} \left[\left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)^2 \right] &+ \mathbb{E} \left[\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] + \\ 2 \left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right) &\underbrace{\mathbb{E} \left[\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right) \right]}_{\text{constant}} \\ \underbrace{2 \left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)}_{\text{constant}} &\underbrace{\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)}_0 \end{aligned}$$



1965716

$$\begin{aligned} \mathbb{E} \left[\left(h(\mathbf{x}) - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] &= \\ \underbrace{\mathbb{E} \left[\left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)^2 \right]}_{\text{constant}} &+ \mathbb{E} \left[\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] \\ (h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})])^2 &+ \end{aligned}$$



1965716

$$\mathbb{E} \left[\left(h(\mathbf{x}) - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right] =$$
$$\left(h(\mathbf{x}) - \mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] \right)^2 + \mathbb{E} \left[\left(\mathbb{E}[f_{\mathbf{w}}(\mathbf{x})] - f_{\mathbf{w}}(\mathbf{x}) \right)^2 \right]$$

(Bias)² **Model Variance**

$$E[(t - f_w(\mathbf{x}))^2]$$

$$= E[(t - h(\mathbf{x}))^2] + \sigma^2$$

“*Noise*” Term

$$E[(h(\mathbf{x}) - f_w(\mathbf{x}))^2]$$

Model
Estimation
Error

We now have
calculated this term

Useful Equations:

$$t = h(\mathbf{x}) + \epsilon$$

$$E(\epsilon) = 0$$

$$Var(\epsilon) = \sigma^2$$

$$E[(t - f_w(\mathbf{x}))^2]$$

$$= \sigma^2$$

$$= E[(t - h(\mathbf{x}))^2] + \text{“Noise” Term}$$

$$(h(\mathbf{x}) - E[f_w(\mathbf{x})])^2 + \text{(Bias)}^2$$

$$E[(E[f_w(\mathbf{x})] - f_w(\mathbf{x}))^2] \text{ Model Variance}$$

Useful Equations:

$$t = h(\mathbf{x}) + \epsilon$$

$$E(\epsilon) = 0$$

$$Var(\epsilon) = \sigma^2$$



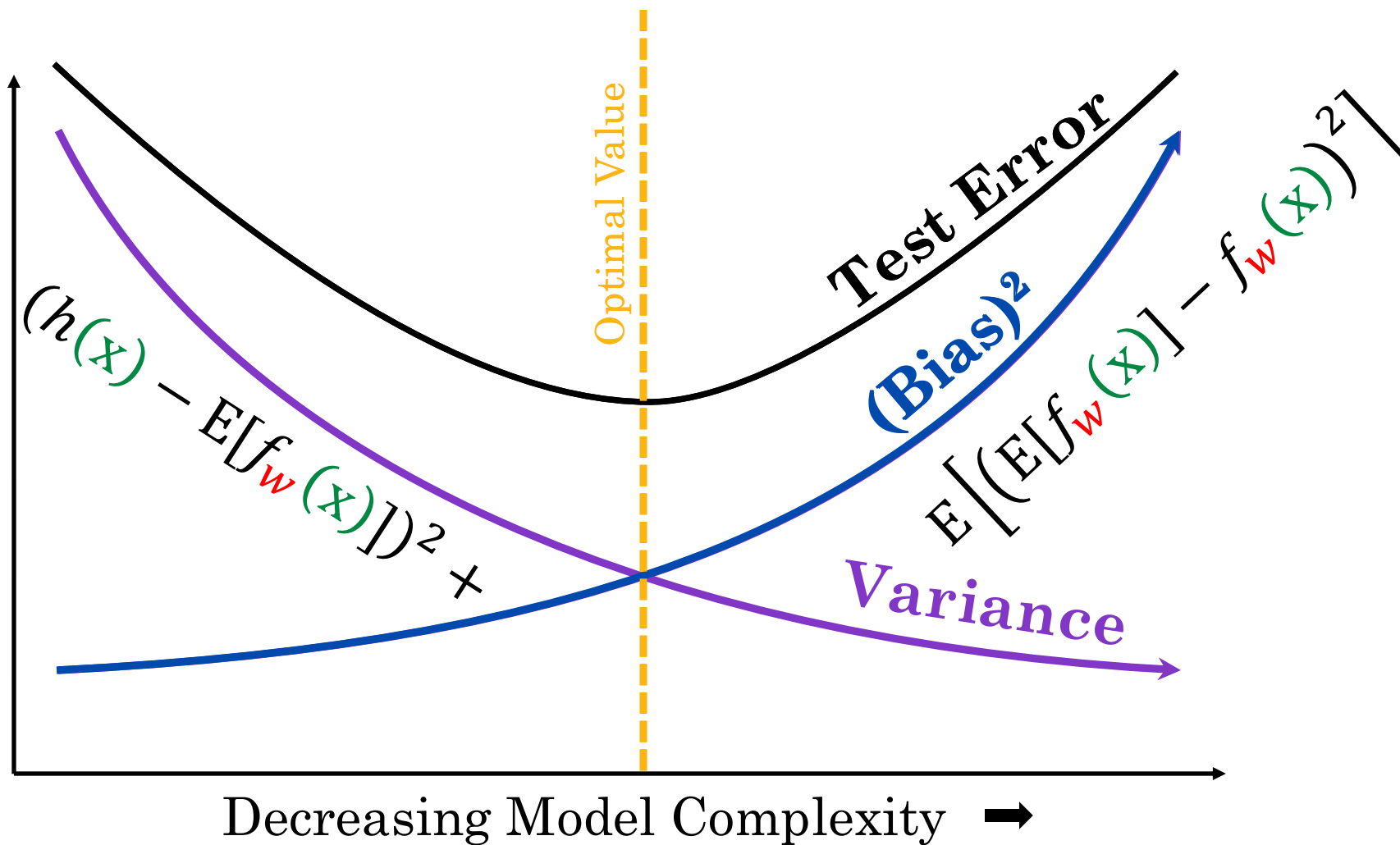
1965716

$$\begin{aligned} & \mathbb{E}[(t - f_w(\mathbf{x}))^2] \\ &= \mathbb{E}[(t - h(\mathbf{x}))^2] + \quad \text{“Noise” Term} \\ & \quad (h(\mathbf{x}) - \mathbb{E}[f_w(\mathbf{x})])^2 + \quad \text{(Bias)}^2 \\ & \quad \mathbb{E}[(\mathbb{E}[f_w(\mathbf{x})] - f_w(\mathbf{x}))^2] \quad \text{Model} \\ & \quad \text{Variance} \end{aligned}$$

Bias Variance Plot



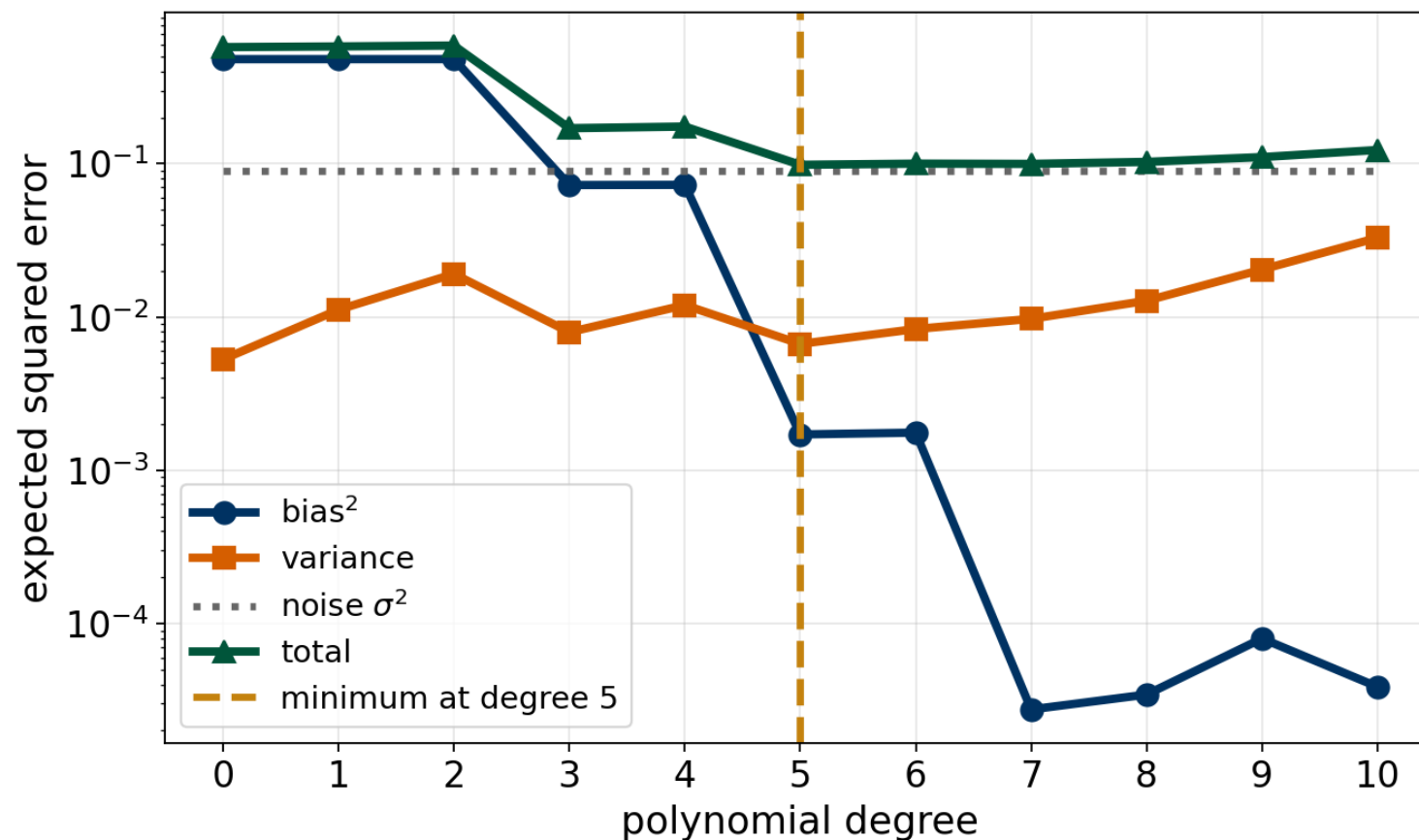
1965716



Measured Bias, Variance and Noise



1965716

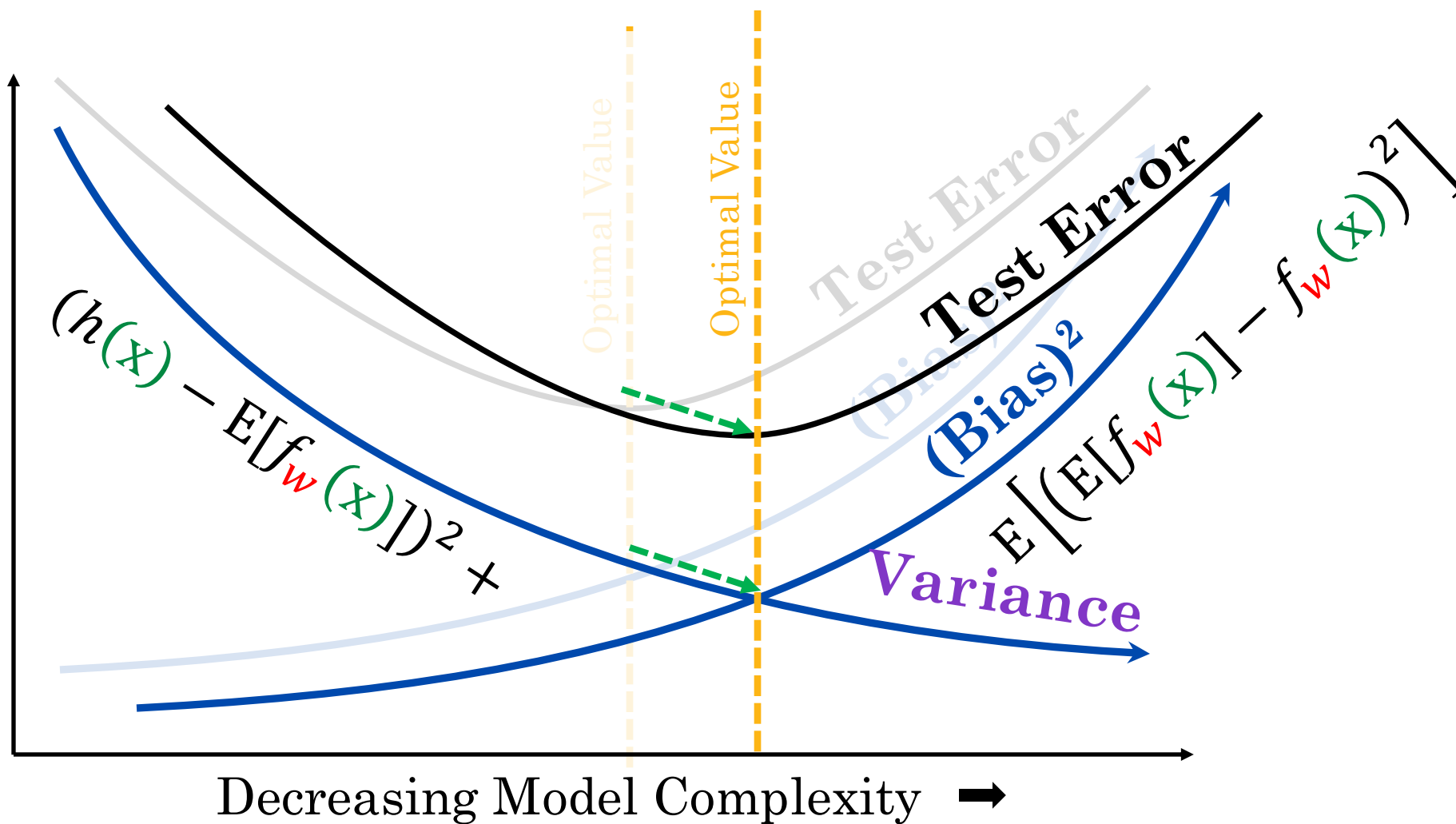


Estimated from 400 training sets per degree, $N = 100$. Bias² falls from 0.481 to 0.000; variance rises from 0.005 to 0.033; noise is fixed at $\sigma^2 = 0.09$. The total is lowest at degree 5.

More Data Supports More Complexity



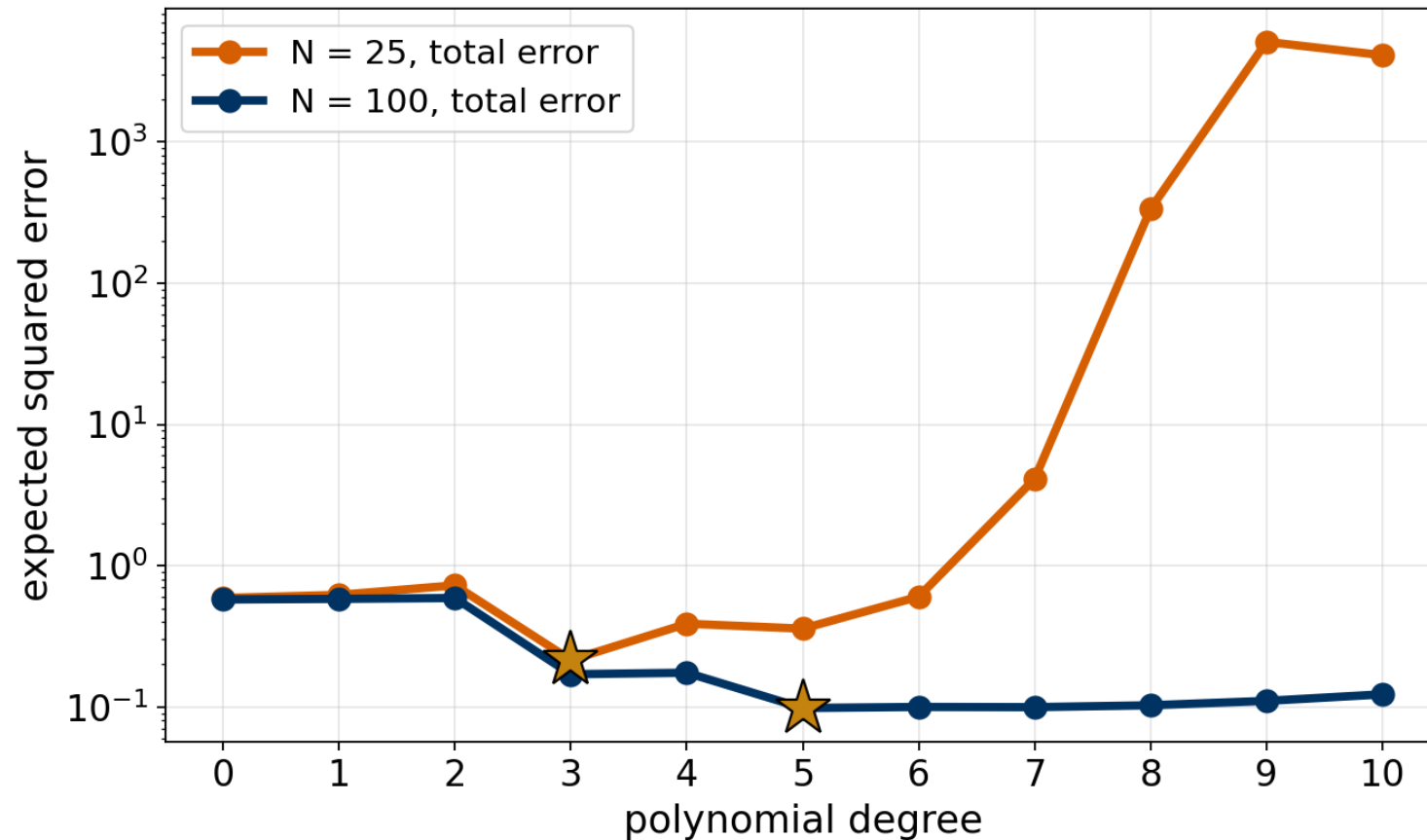
1965716



Effect of Training Set Size



1965716



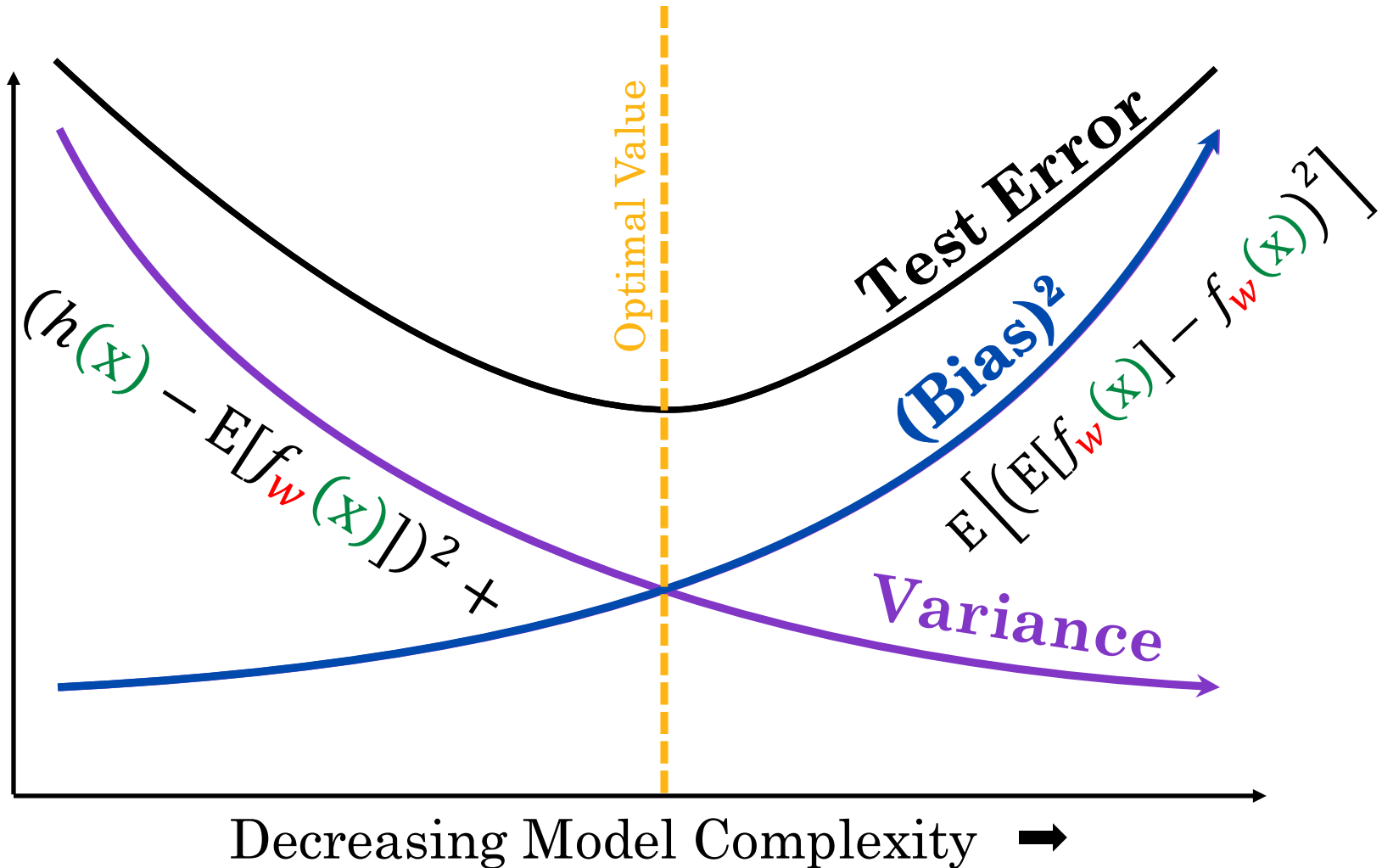
The same decomposition at $N = 25$ and $N = 100$. The minimum moves from degree 3 to degree 5: with less data, the best model is simpler than the truth.

How Do We Control Model Complexity?

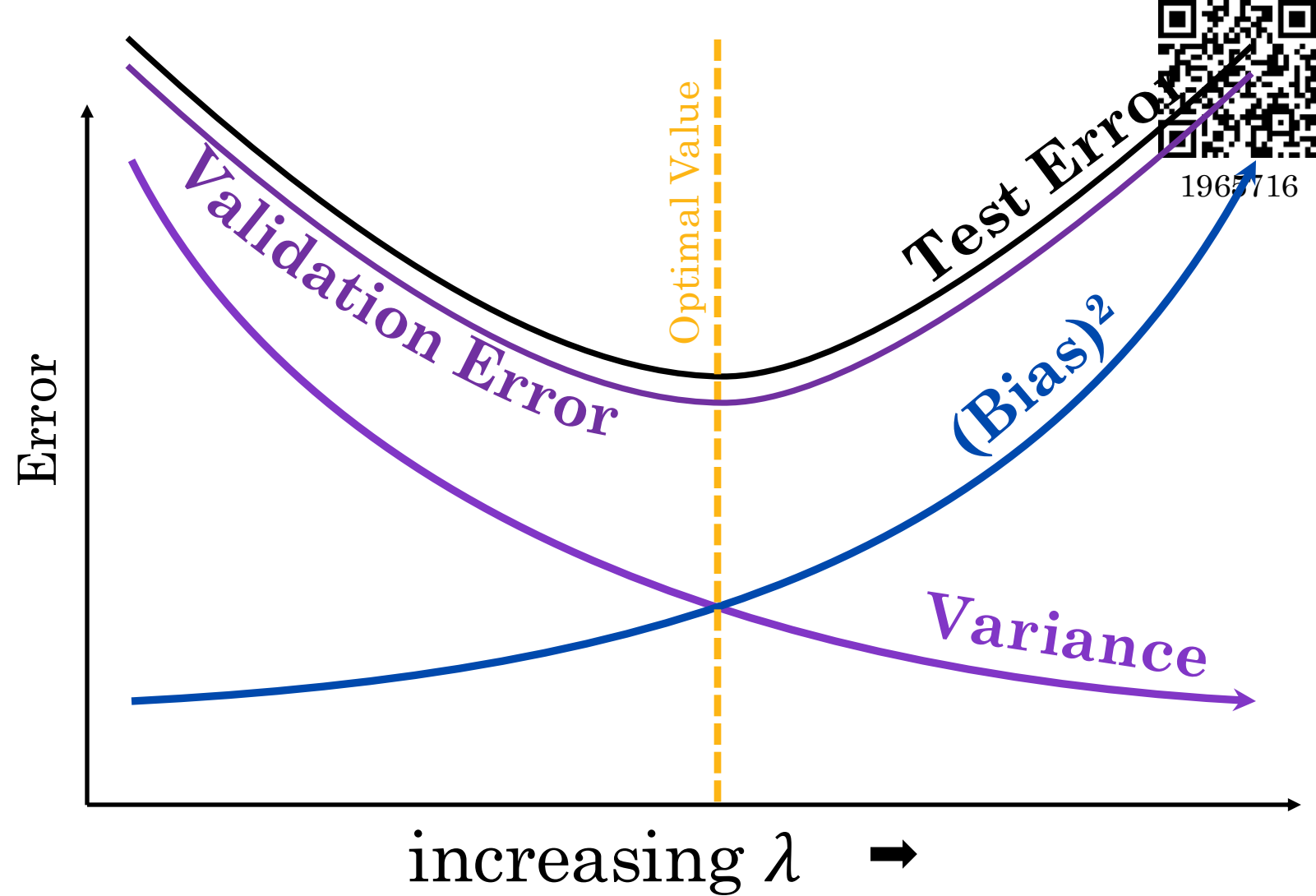


1965716

- So far:
 - Number of features
 - Choices of features
 - **Regularization**



Determining the Optimal λ

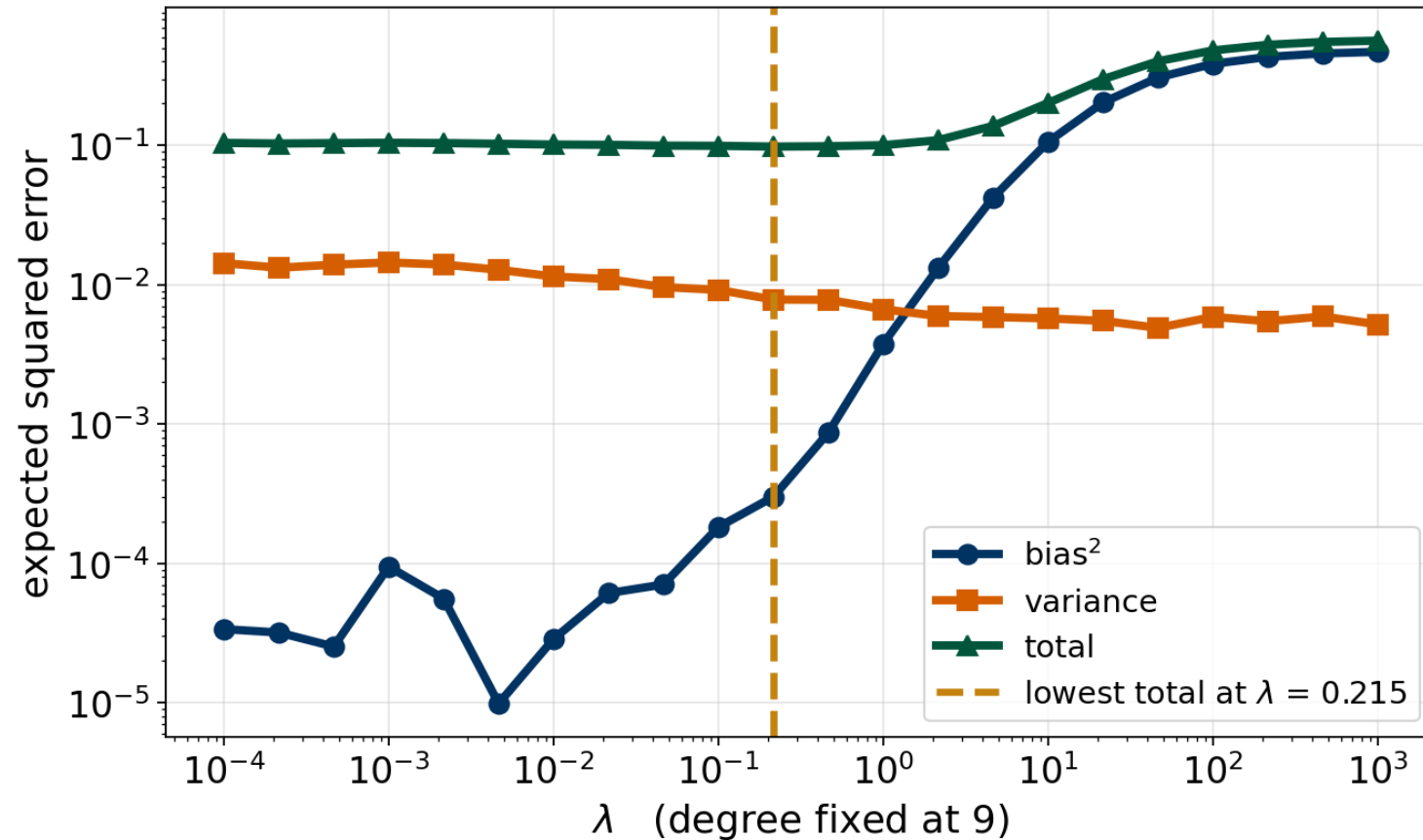


- Value of λ determines bias-variance tradeoff
 - Larger values $\rightarrow$ more regularization $\rightarrow$ more bias $\rightarrow$ less variance
- Determined through validation

Effect of λ at Fixed Degree



1965716



Degree held at 9. Bias² rises from 0.000 to 0.470 and variance falls from 0.014 to 0.005 as λ grows. The total is lowest at $\lambda = 0.215$.



Select options below



Select options below



Select options below

Classification Task

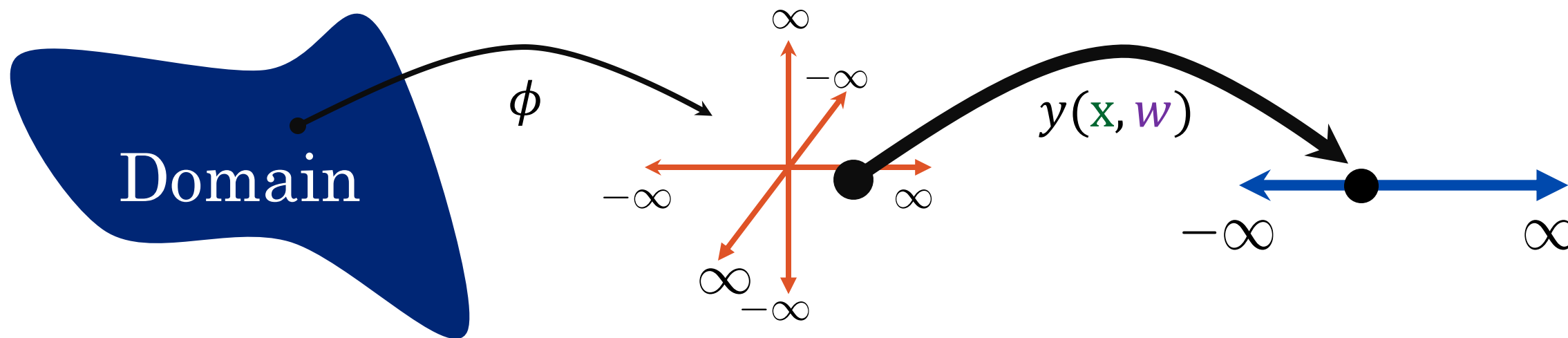
- Bias-Variance Trade-off
- Classification Task
- Linear Discriminant Functions
- Discriminative Probabilistic Models
- The Logistic Regression Model
- Logistic Regression Objective



So Far



1965716



Squared Error

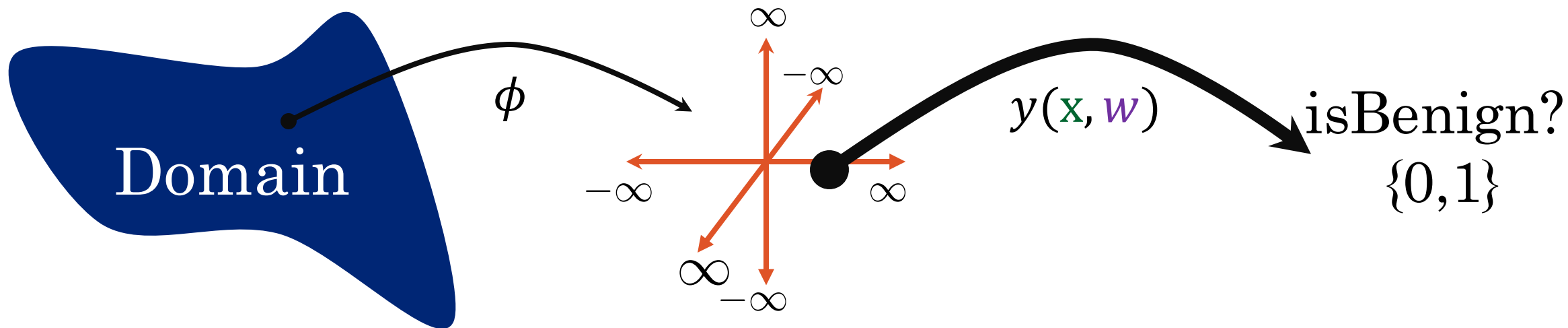
$$\operatorname{argmin}_{\mathbf{w}} \frac{1}{2} \sum_{n=1}^N \left(t_n - \mathbf{w}^T \phi(\mathbf{x}_n) \right)^2 + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w}$$

Regularization

Classification Task Shift



1965716





1965716

Kinds of Classification

We want to predict some **categorical variable**, or **response**, y .

- **Binary** classification

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

win or lose

disease or
no disease

- **Multiclass** classification

- Many classes
- Image labeling and sentiment analysis (positive, negative, or neutral).

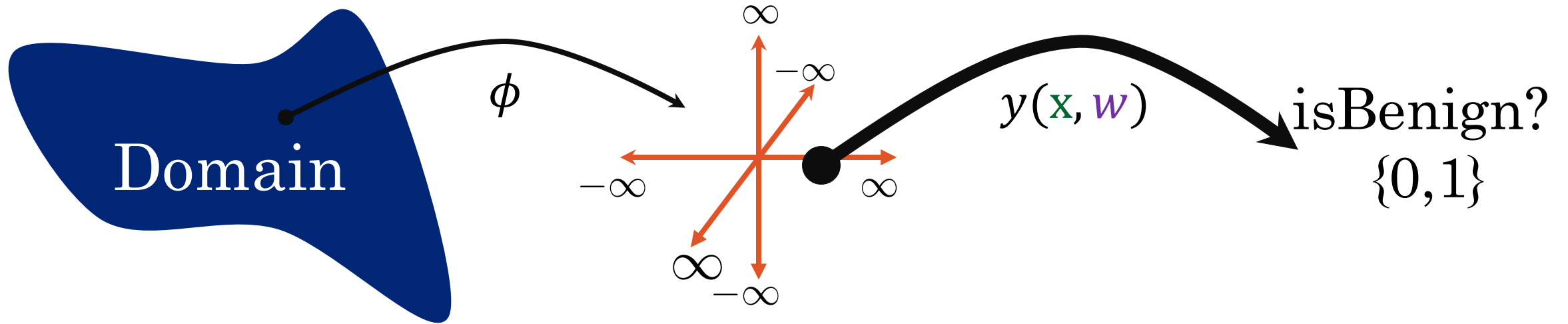


- **Structured prediction (e.g., translation): later in the course**

Classification



1965716



Can we just use least squares?

$$\operatorname{argmin}_{\mathbf{w}} \frac{1}{2} \sum_{n=1}^N \left(t_n - \mathbf{w}^T \phi(\mathbf{x}_n) \right)^2 + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w}$$

Squared Error

Regularization



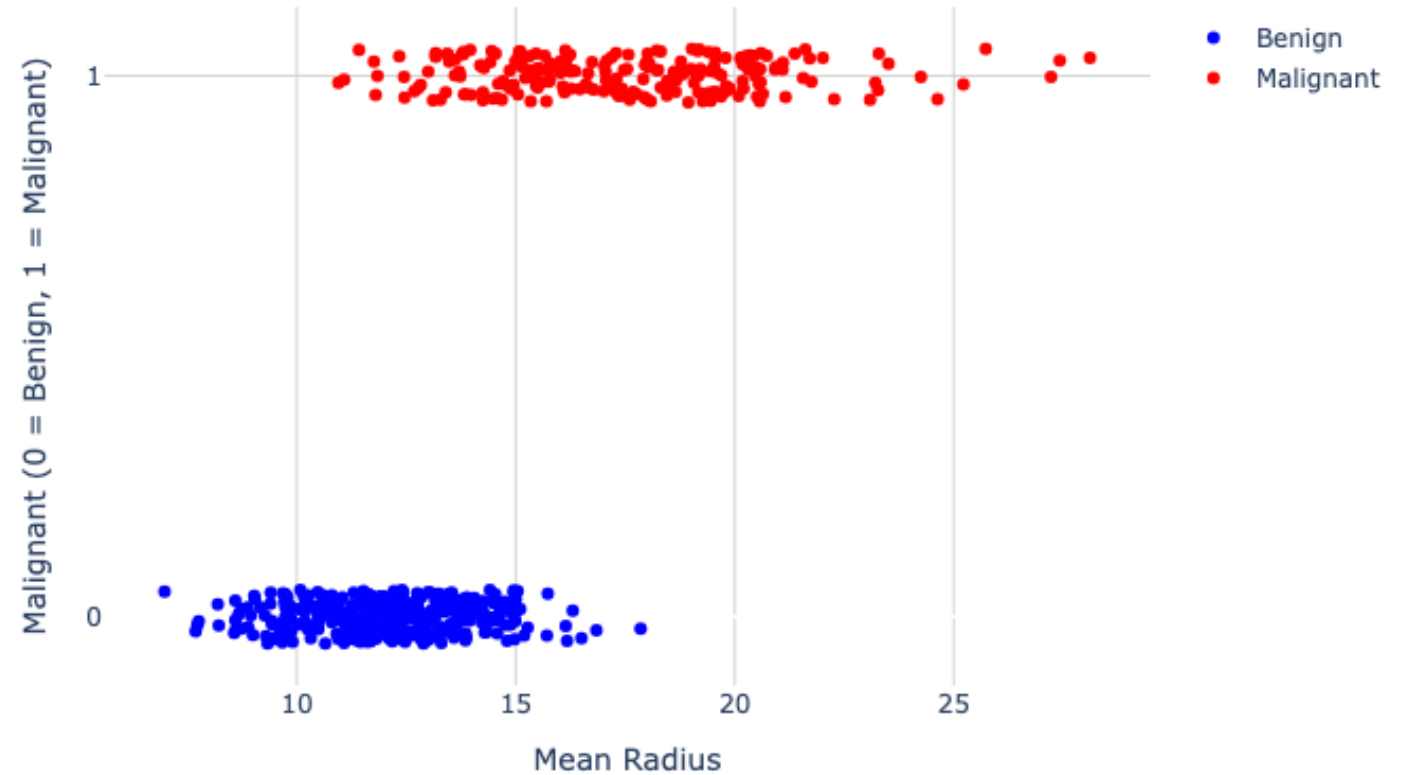
Demo

We are using **Wisconsin Breast Cancer** dataset.

We add jitter so that all data points are visible.

```
def jitter(y, amt=0.1):  
    return y + amt*rng.random(len(y)) - amt/2
```

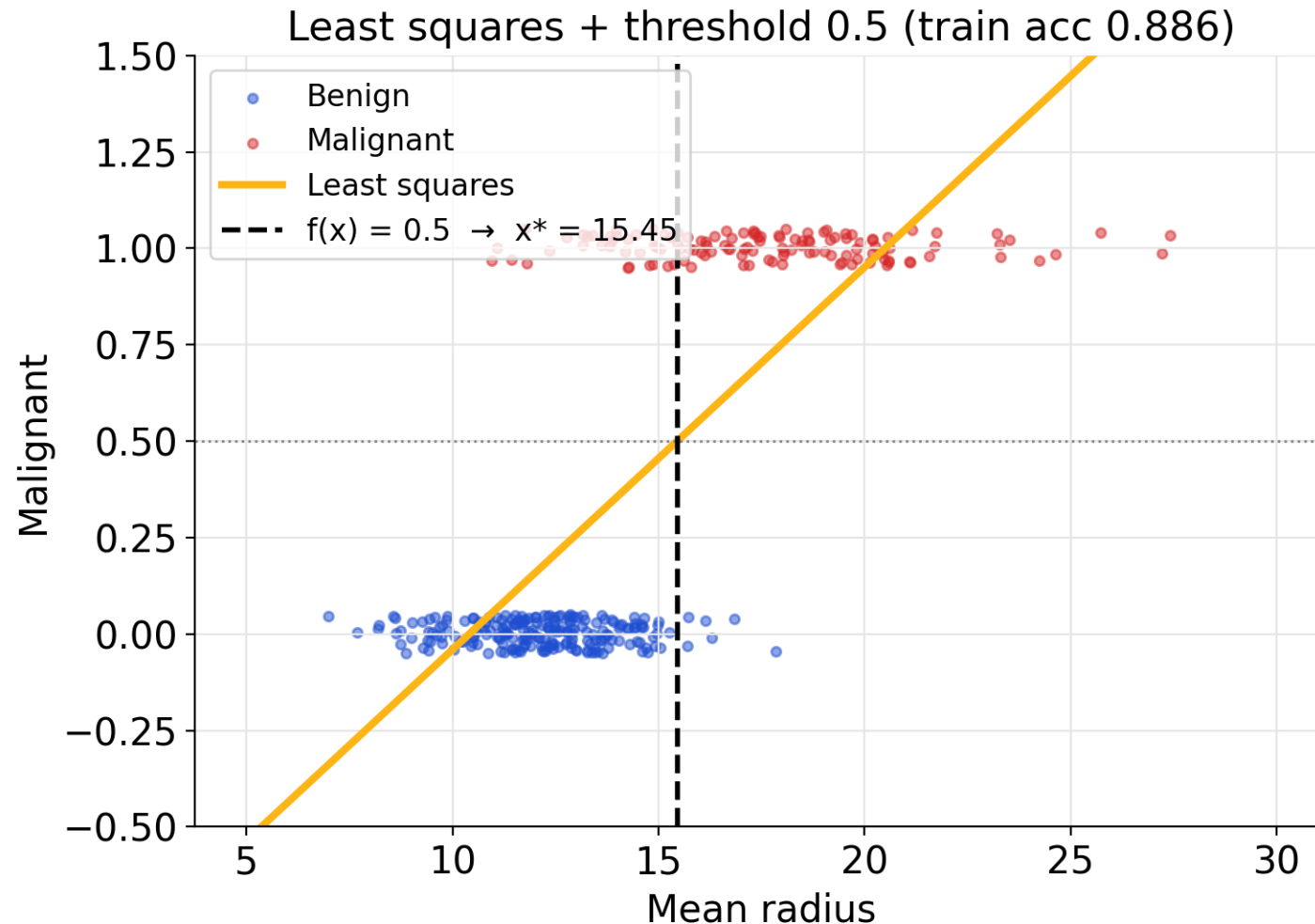
The Distribution of the Data Between the Two Classes (Jittered)



Demo

We needed a decision function
(e.g., $f(\mathbf{x}) > 0.5$) ...

If we use the threshold of 0.5
between the two target values 0
and 1 and highlight all data points
that are labeled as benign with
blue and all malignant predictions
with red, you will see that a
decision boundary is being formed
between the two classes with the
black dashed line.



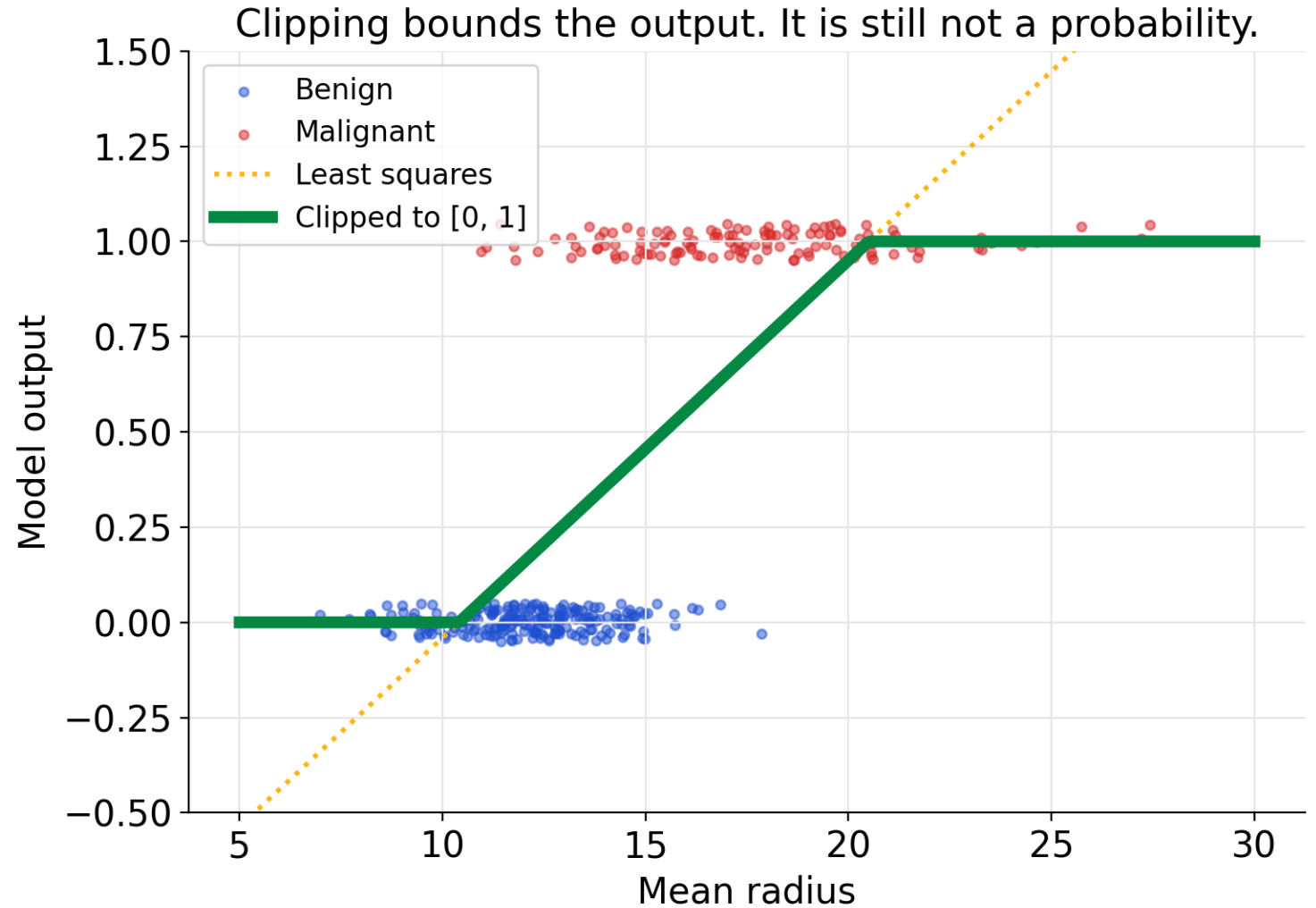


1965716

Demo

A model that outputs values below 0 and above 1 is hard to interpret, so we clip the output to $[0, 1]$.

Clipping bounds the output. It does not make it a probability.

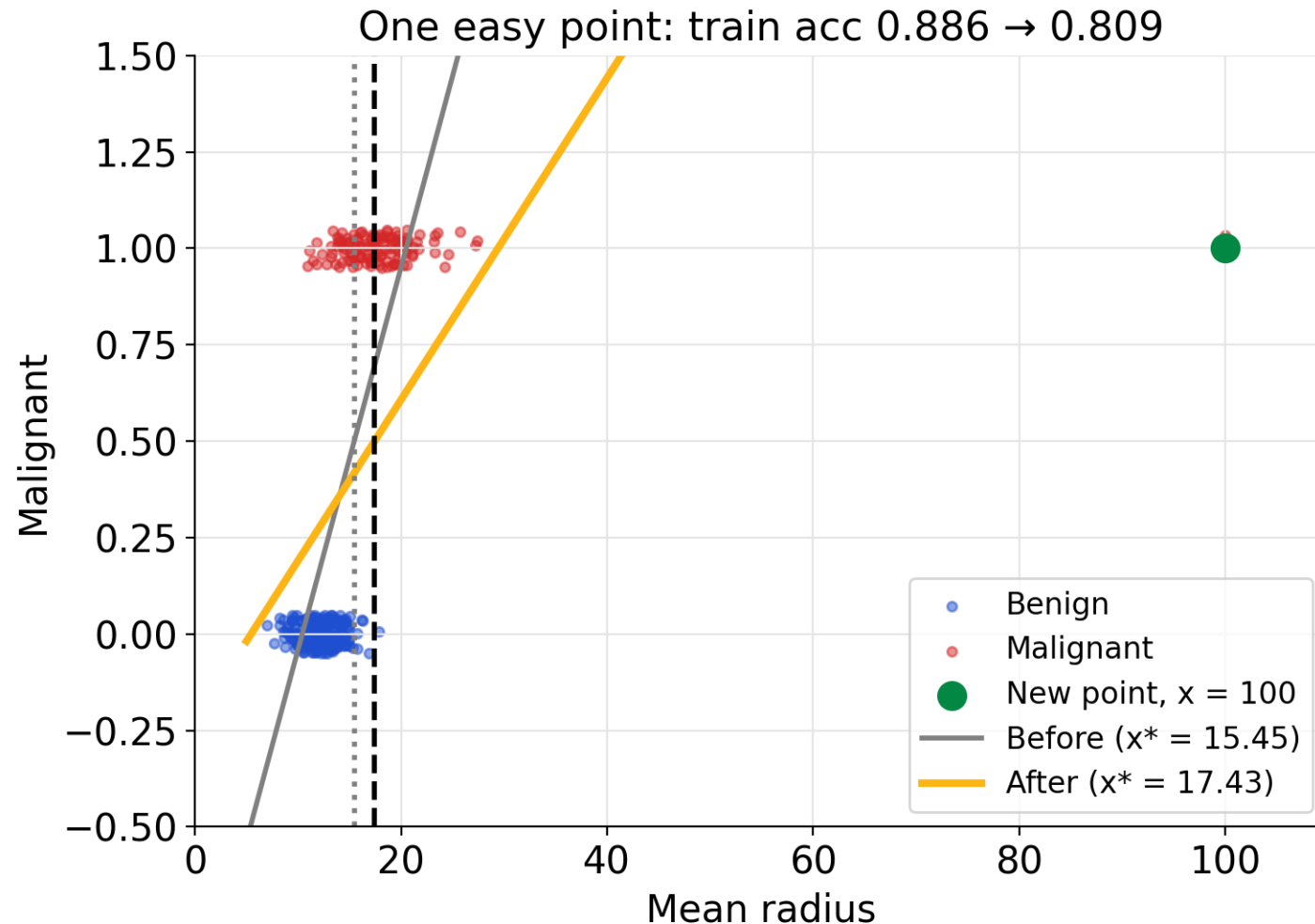


Demo

Add one malignant tumor with radius 100. It is already on the correct side.

Boundary moves $15.45 \rightarrow 17.43$.

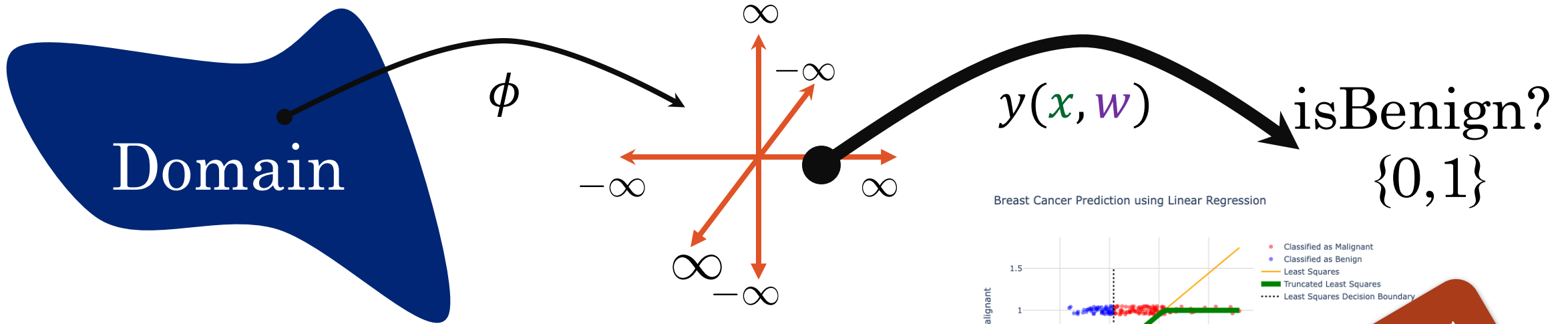
Training accuracy drops $0.886 \rightarrow 0.809$.



Classification



1965716



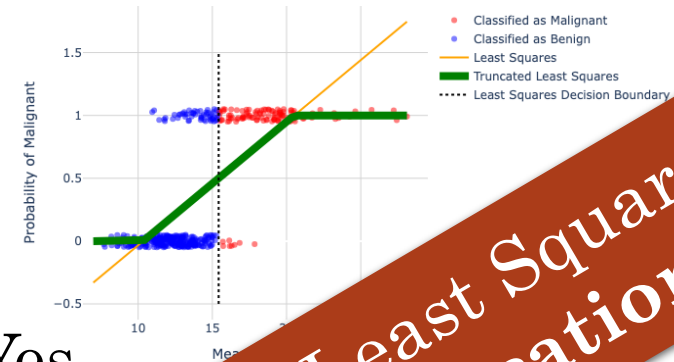
Can we just use least squares?

$$\underset{w}{\operatorname{argmin}} \frac{1}{2} \sum_{n=1}^N (t_n - w^T \phi(x_n))^2 + \frac{\lambda}{2} w^T w$$

Squared Error

Regularization

Breast Cancer Prediction using Linear Regression



- Yes ...
- Need a loss function (e.g., $f(x)$)
- Hard to interpret model ...
- Sensitive to outliers

Don't use Least Squares for Classification

Least squares is maximum likelihood under Gaussian noise. Binary labels are not Gaussian.

Takeaway!!!

Linear Discriminant Functions

- Bias-Variance Trade-off
- Classification Task
- Linear Discriminant Functions
- Discriminative Probabilistic Models
- The Logistic Regression Model
- Logistic Regression Objective





1965716

Classification Setup

- **Goal:** Map an input vector $\mathbf{x} \in \mathbb{R}^D$ to one of the K **discrete classes** C_k where $k = 1, 2, \dots, K$.
 - $K = 2$: **Binary Classification**
 - $K > 2$: **Multi-class Classification**
- Input space is divided into decision regions. Boundaries are **decision boundaries** or **decision surfaces**.
 - $(D - 1)$ -dimensional hyperplanes within the D -dimensional input space.

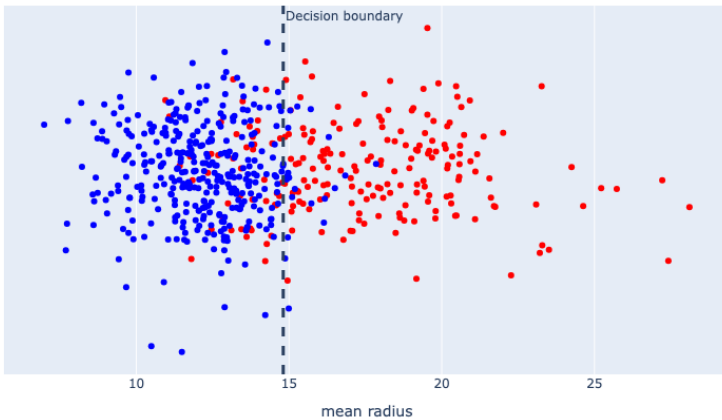
Classification Setup



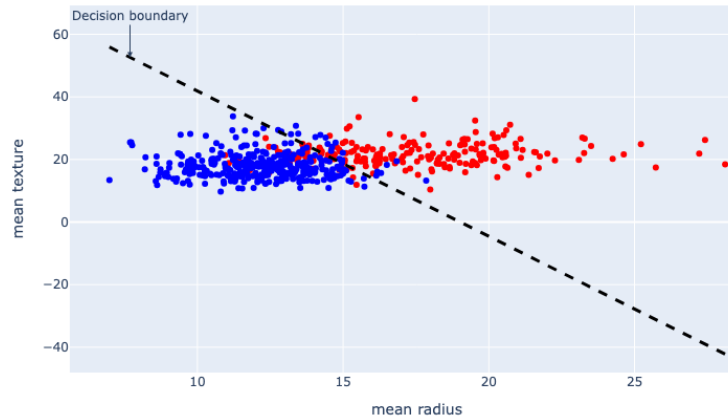
1965716

- Input space is divided into decision regions. Boundaries are **decision boundaries** or **decision surfaces**.
 - $(D - 1)$ -dimensional hyperplanes within the D -dimensional input space.

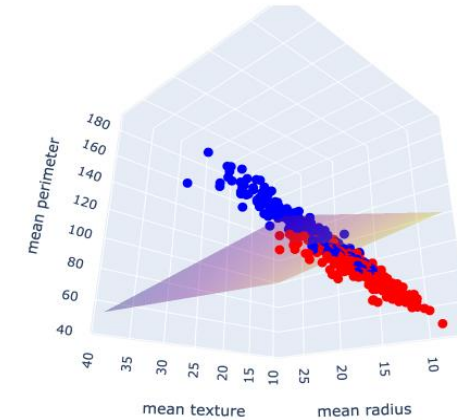
$D = 1 \rightarrow$ decision boundary is 0-D (a point)



$D = 2 \rightarrow$ decision boundary is 1-D (a line)



$D = 3 \rightarrow$ decision boundary is 2-D (a plane)



- Dataset with classes that are separable by linear decision surfaces are said to be **linearly separable**.



1965716

3 Approaches in Solving Classification

1. **Discriminant functions** (direct mapping): Direct assignment to a class.

Ex: perceptron, SVM

2. **Generative probabilistic models** (model conditional probability $p(\mathbf{x}|C_k)$ and prior $p(C_k)$ then apply Bayes).

Ex: Naïve Bayes

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})}$$

3. **Discriminative probabilistic models** (model conditional probability $p(C_k|\mathbf{x})$ directly).

Ex: Logistic Regression

Discriminant Classification with Two Classes



1965716

- **Basic representation:**

A linear function of the input: $y(x) = w^T x + w_0$

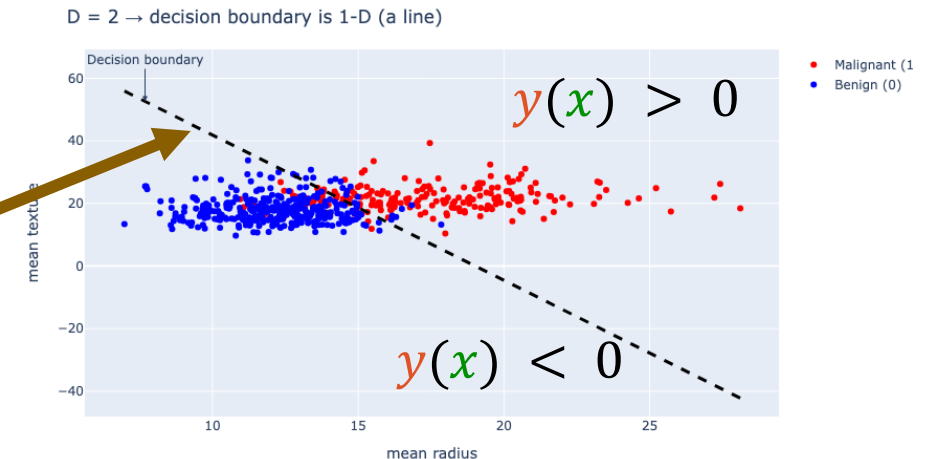
- **Decision rule:**

Assign x to C_1 if $y(x) \geq 0$, otherwise to C_2

- **Decision boundary:**

$y(x) = 0$, a $(D - 1)$ -dimensional hyperplane.

$$y(x) = 0$$
$$w^T x + w_0 = 0$$
$$x_{\text{radius}} \cdot w_1 + x_{\text{texture}} \cdot w_2 + w_0 = 0$$



Geometry of Linear Discriminant Functions



1965716

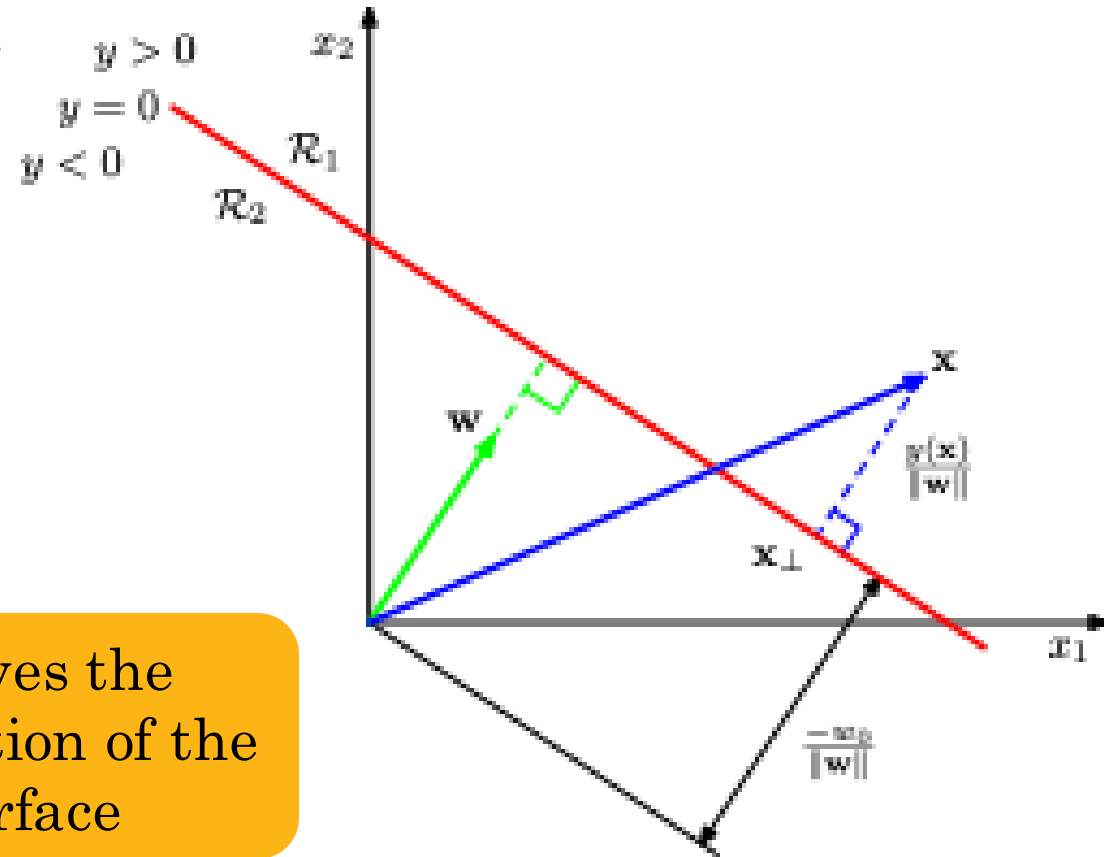
- For any two points on the line $\mathbf{x}_A$ and $\mathbf{x}_B$:

$$y(\mathbf{x}_A) = y(\mathbf{x}_B) = 0$$

$$\mathbf{w}^T(\mathbf{x}_A - \mathbf{x}_B) = 0$$

- $\mathbf{w}$ is orthogonal to every vector lying within the decision surface/boundary.

$\mathbf{w}$ gives the orientation of the surface





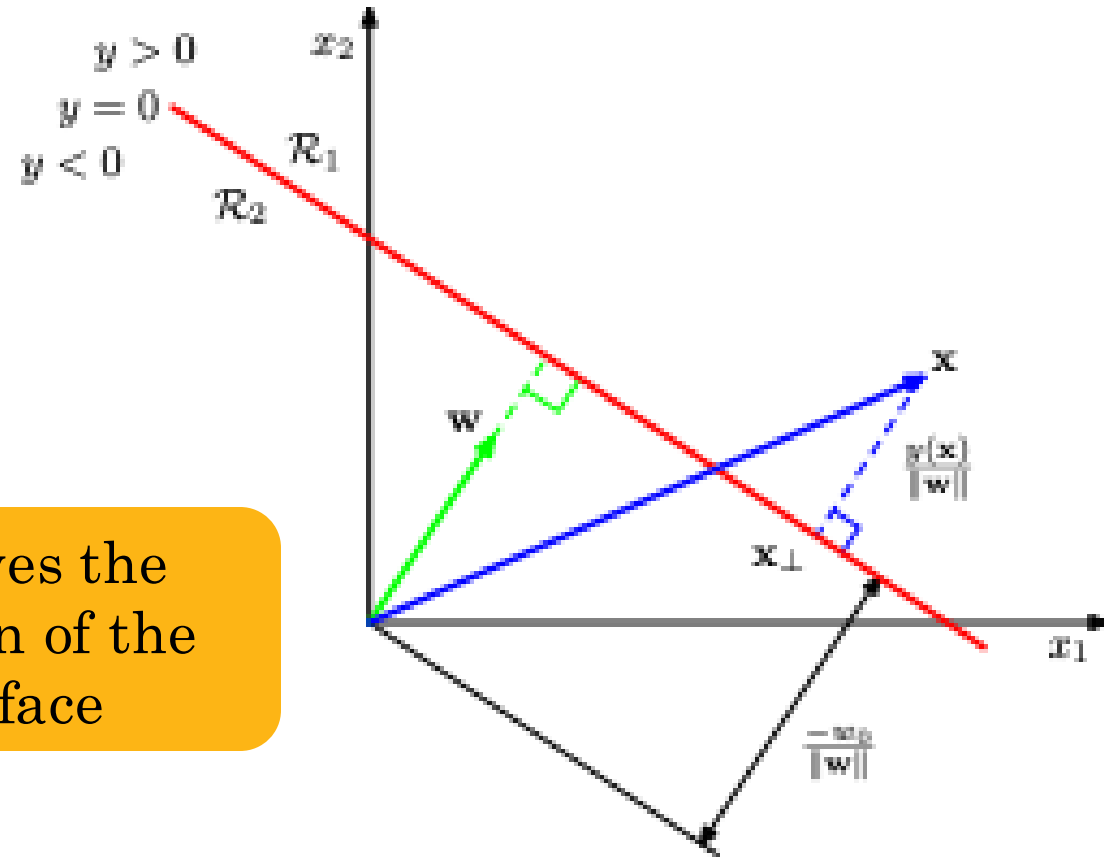
1965716

Geometry of Linear Discriminant Functions

- For every point on the decision surface, $y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0 = 0$.
- The normal distance of the surface to the origin is:

$$\frac{\mathbf{w}^T \mathbf{x}}{\|\mathbf{w}\|} = -\frac{w_0}{\|\mathbf{w}\|}$$

w_0 gives the location of the surface





If our model does not have an intercept term, the decision boundary must pass through the origin.

Discriminative Probabilistic Models

- Bias-Variance Trade-off
- Classification Task
- Linear Discriminant Functions
- Discriminative Probabilistic Models
- The Logistic Regression Model
- Logistic Regression Objective





1965716

Discriminative Probabilistic Models

Logistic regression is the most widely used **discriminative probabilistic model** for **classification**.

Logistic regression models the **log-odds** of class C_1 versus C_2 **as a linear model**.

$$\mathbf{Odds\ Ratio} = \frac{p(C_1|\mathbf{x})}{p(C_2|\mathbf{x})} = \frac{p(C_1|\mathbf{x})}{1-p(C_1|\mathbf{x})}$$

$$\mathbf{Log\ Odds} = \ln \left(\frac{p(C_1|\mathbf{x})}{p(C_2|\mathbf{x})} \right) = \ln \left(\frac{p(C_1|\mathbf{x})}{1-p(C_1|\mathbf{x})} \right) = \mathbf{w}^\top \mathbf{x}$$

We can solve for $p(C_1|\mathbf{x})$ to obtain the form of our model.



1965716

Deriving the Logistic Sigmoid Function

We can solve for $p(C_1|\mathbf{x})$ to obtain the form of our model.

$$\text{Log Odds} = \ln \left(\frac{p(C_1|\mathbf{x})}{p(C_2|\mathbf{x})} \right) = \ln \left(\frac{p(C_1|\mathbf{x})}{1-p(C_1|\mathbf{x})} \right) = \mathbf{w}^\top \mathbf{x}$$

$$\frac{p(C_1|\mathbf{x})}{1-p(C_1|\mathbf{x})} = \exp(\mathbf{w}^\top \mathbf{x})$$

$$p(C_1|\mathbf{x}) = \exp(\mathbf{w}^\top \mathbf{x}) (1 - p(C_1|\mathbf{x}))$$

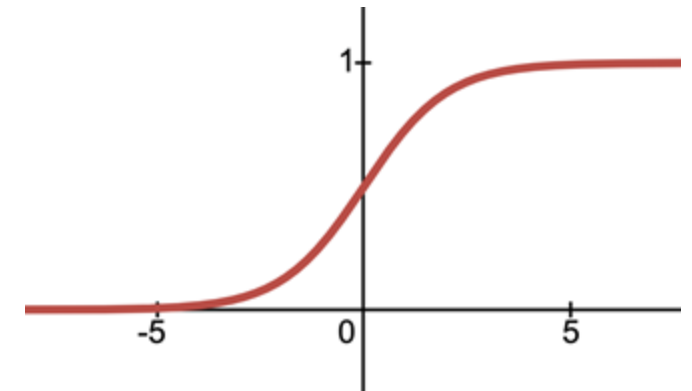
$$p(C_1|\mathbf{x}) = \exp(\mathbf{w}^\top \mathbf{x}) - \exp(\mathbf{w}^\top \mathbf{x}) p(C_1|\mathbf{x})$$

$$p(C_1|\mathbf{x}) + \exp(\mathbf{w}^\top \mathbf{x}) p(C_1|\mathbf{x}) = \exp(\mathbf{w}^\top \mathbf{x})$$

$$p(C_1|\mathbf{x}) = \frac{\exp(\mathbf{w}^\top \mathbf{x})}{1 + \exp(\mathbf{w}^\top \mathbf{x})} = \frac{1}{1 + \exp(-\mathbf{w}^\top \mathbf{x})}$$

Sigmoid Function

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



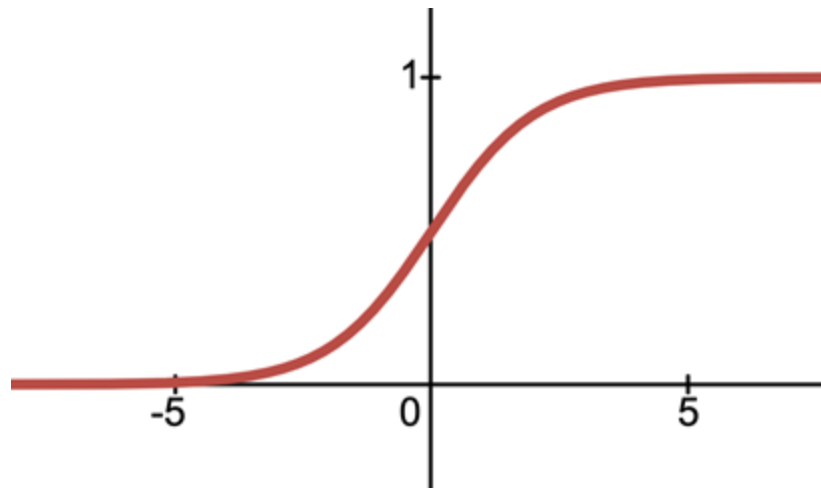
Algebra

The Sigmoid (σ) Function



1965716

Also known as the **logistic** function.



The term Sigmoid means S-shaped.

A ‘squashing function’ since it maps the real axis into a finite interval.

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


sigmoid function

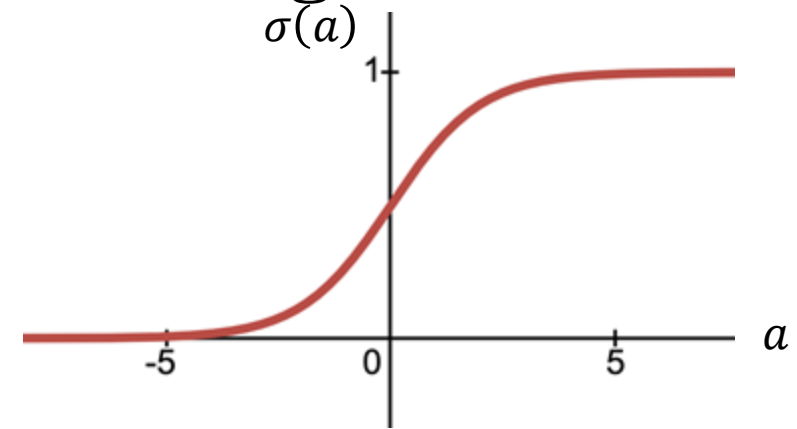
The Sigmoid (σ) Function



1965716

The S-shaped curve is formally known as the **sigmoid function**.

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



Reflection/
Symmetry

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

Domain
 $-\infty < a < +\infty$

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

Inverse

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

Logit Function

Derivative

$$\frac{d}{da} \sigma(a) = \sigma(a)(1 - \sigma(a)) = \sigma(a)\sigma(-a)$$



**Which of the following are
properties of the sigmoid
function, $\sigma(x)$?**

The Logistic Regression Model

- Bias-Variance Trade-off
- Classification Task
- Linear Discriminant Functions
- Discriminative Probabilistic Models
- The Logistic Regression Model
- Logistic Regression Objective



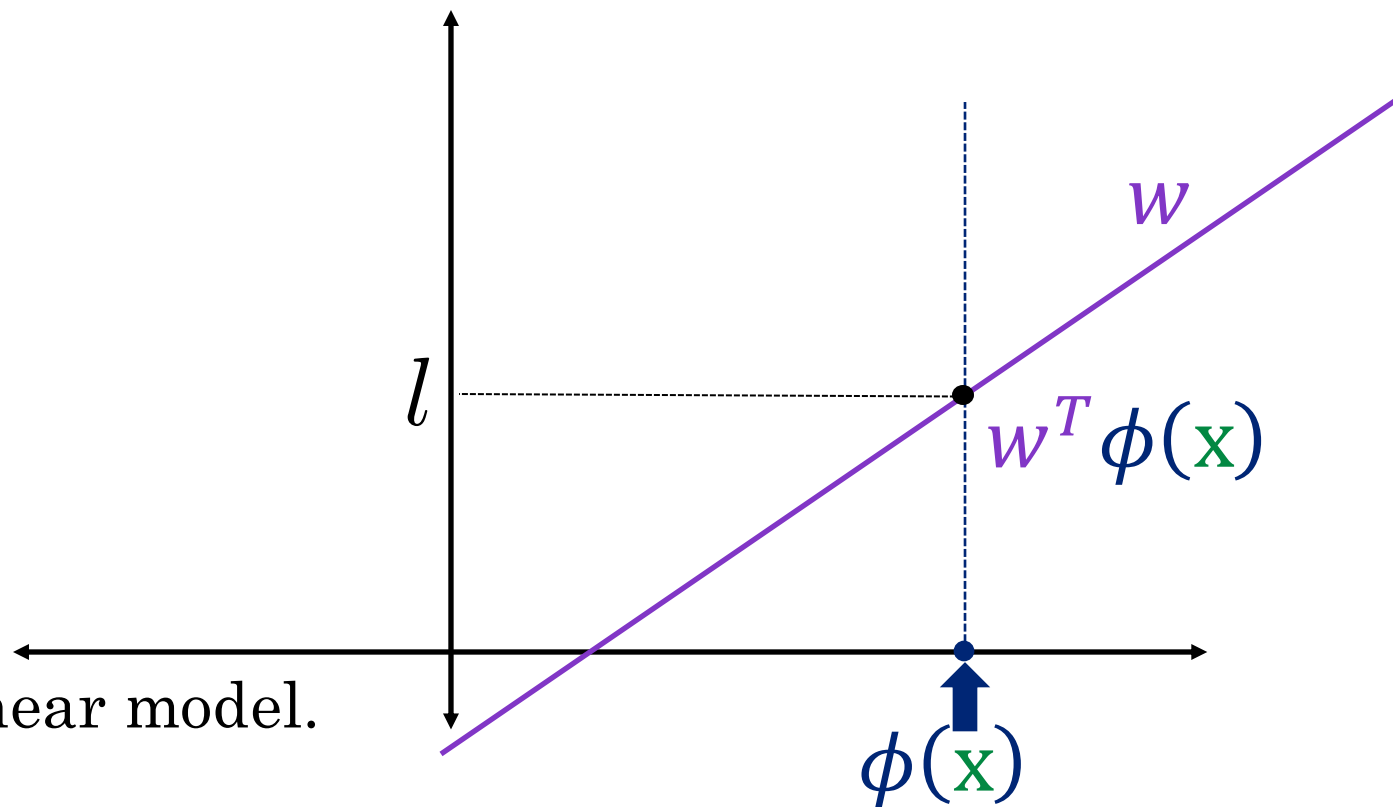
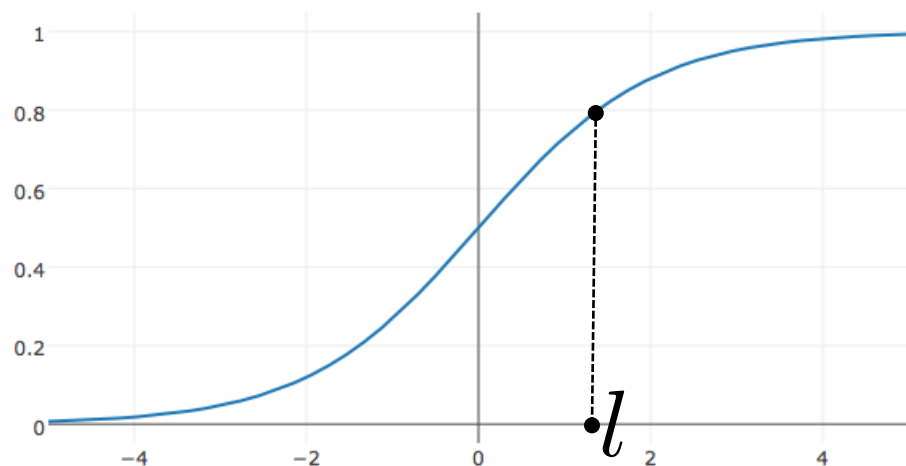
Logistic Regression



1965716

Linear Model

$$\text{Model: } P(C_1 | \phi(\mathbf{x})) = \underbrace{\sigma}_{\text{Sigmoid}} \left(\underbrace{w^T \phi(\mathbf{x})}_{\text{Linear}} \right) = \frac{1}{1 + \exp(-w^T \phi(\mathbf{x}))}$$



Generalized Linear Model:

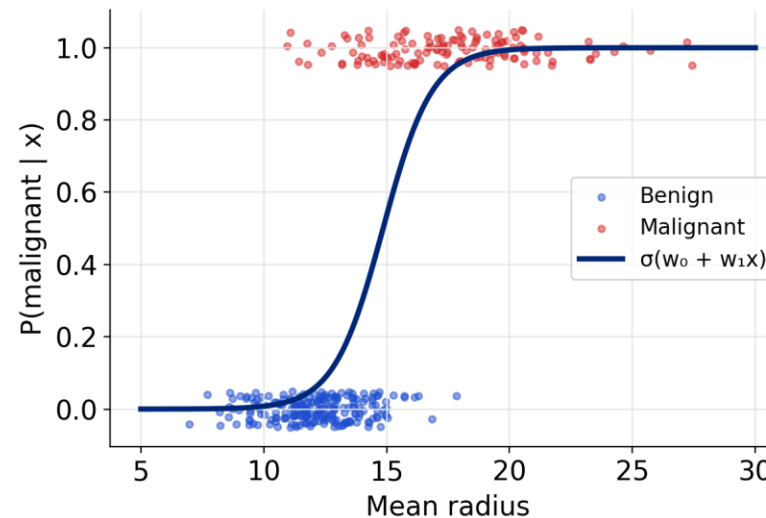
Non-linear transformation of a linear model.

Our Cancer Classification Example



1965716

We've discovered the (simple) **logistic regression** model! It just fits a sigmoid to the data.



Fitted on our training split:
 $w_0 = -14.81$, $w_1 = 1.00$

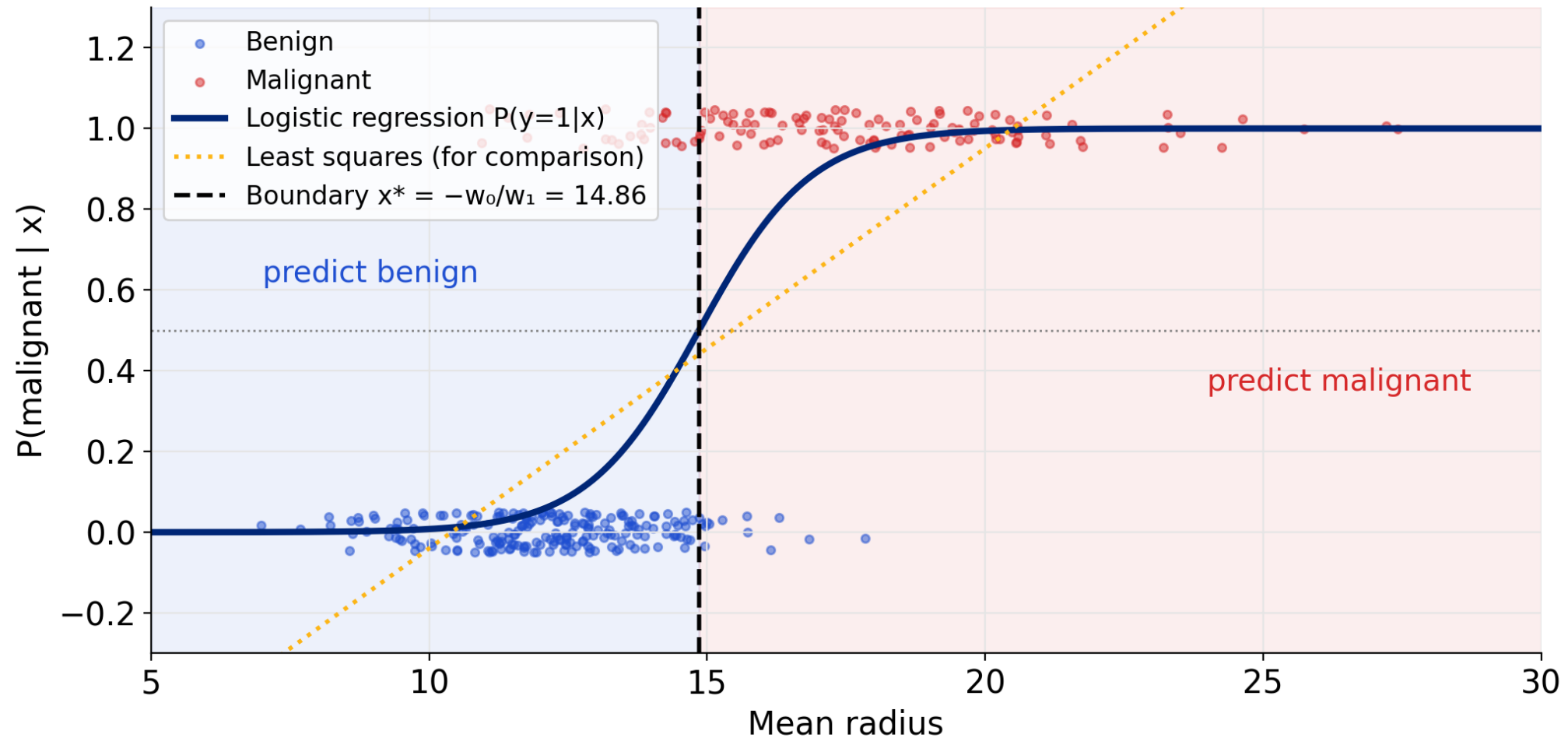
$$p(C_1 | \phi(\mathbf{x})) = \sigma(w_0 + w_1 \mathbf{x}) = \frac{1}{1 + e^{-(w_0 + w_1 \mathbf{x})}}$$

How do we identify the optimal w 's? More shortly!

Logistic Regression and Decision Boundary



1965716



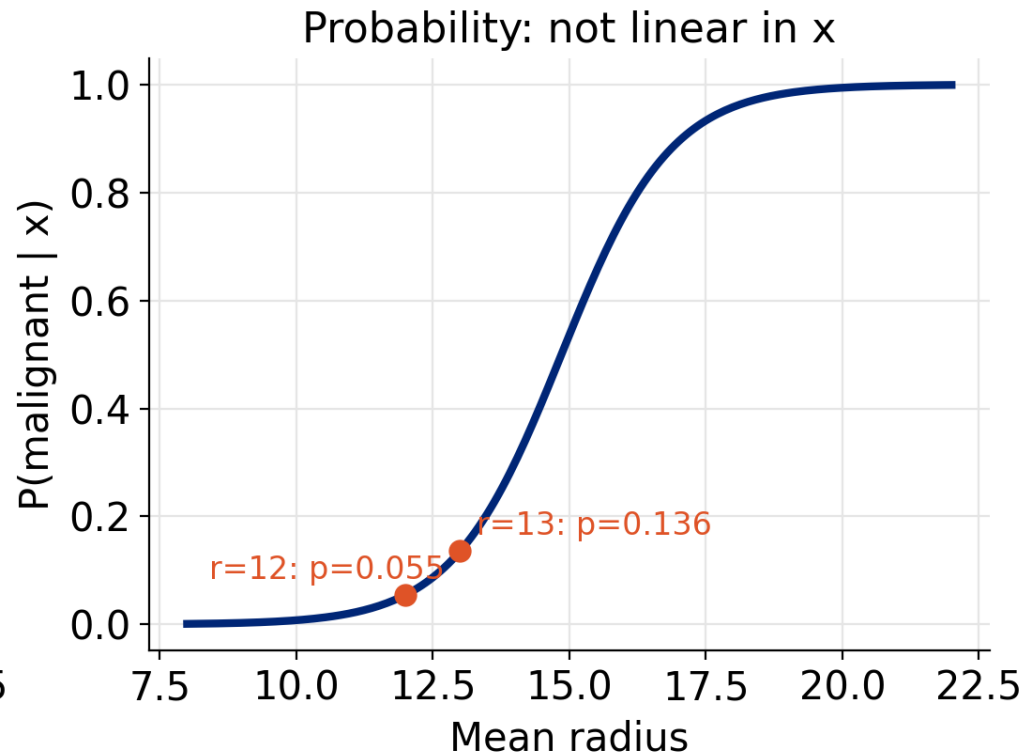
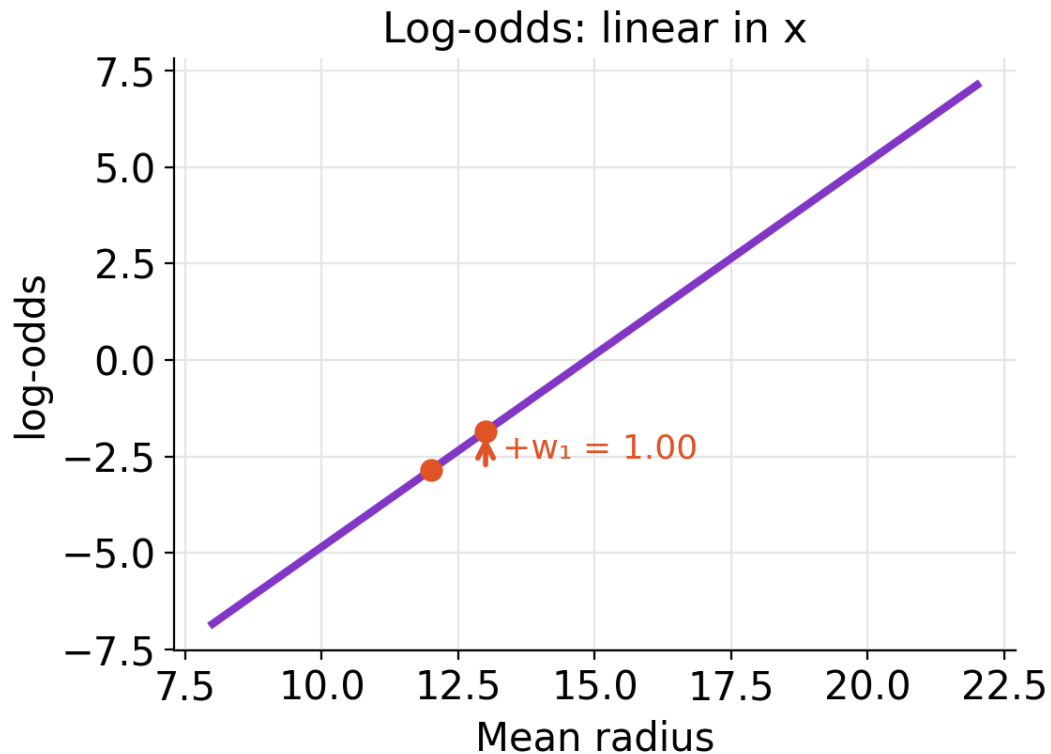
$$\sigma(w_0 + w_1 x) = 0.5 \Leftrightarrow w_0 + w_1 x = 0 \Leftrightarrow x^* = -\frac{w_0}{w_1} = 14.86$$



1965716

Reading the Weights: Odds

- The model says $\log \frac{p}{1-p} = \mathbf{w_0} + \mathbf{w_1 x}$. One more unit of $\mathbf{x}$ adds $\mathbf{w_1}$ to the log-odds, and it **multiplies the odds** by $\exp(\mathbf{w_1}) = \exp(\mathbf{1.00}) = \mathbf{2.71}$. It does not multiply the probability.
- Radius 12 $\rightarrow$ 13: $p = 0.055 \rightarrow 0.136 (\times 2.48)$

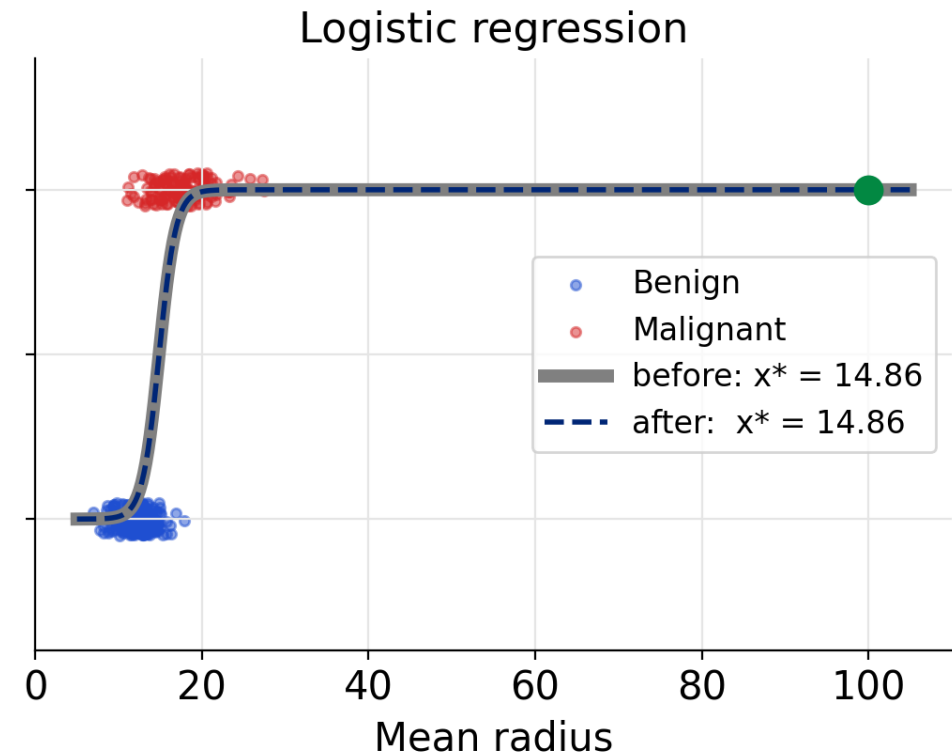
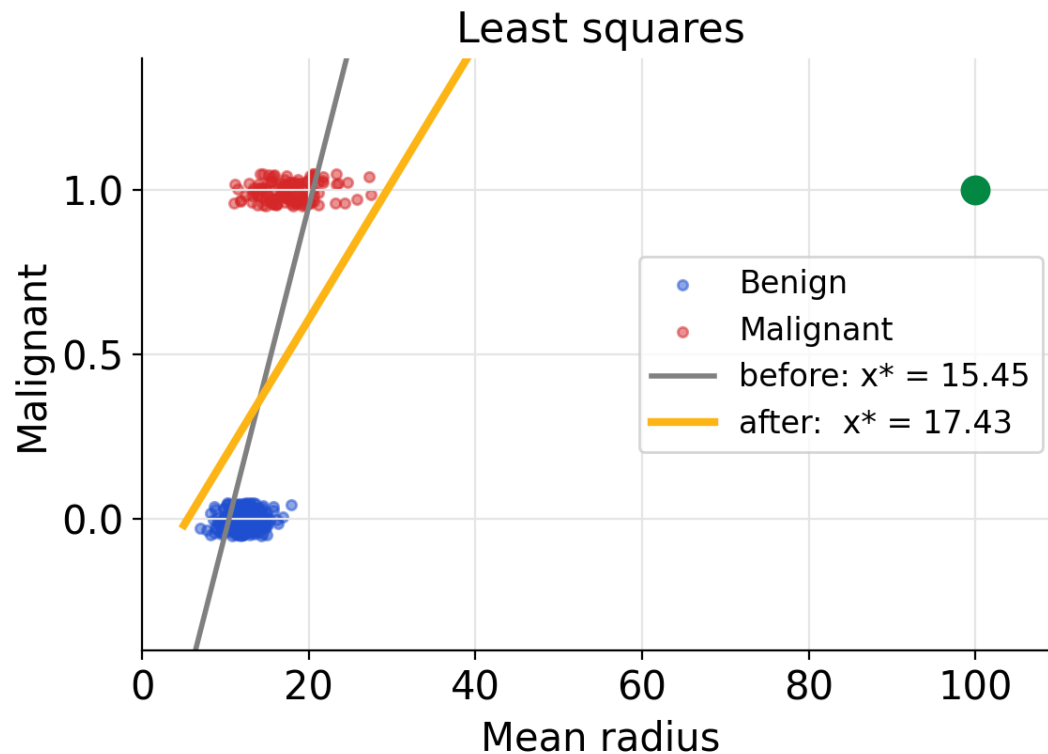




1965716

Logistic Regression Ignores Easy Points

- Same extra tumor at radius 100 as before.
- Least squares boundary: $15.45 \rightarrow 17.43$. Logistic regression boundary: $14.86 \rightarrow 14.86$.
- A confident, correct point has $\sigma(\cdot) \approx 1$, so it adds almost nothing to the loss we are about to define.



Logistic Regression Objective

- Bias-Variance Trade-off
- Classification Task
- Linear Discriminant Functions
- Discriminative Probabilistic Models
- The Logistic Regression Model
- Logistic Regression Objective



1965716

Can We Use Squared Error
As Objective?

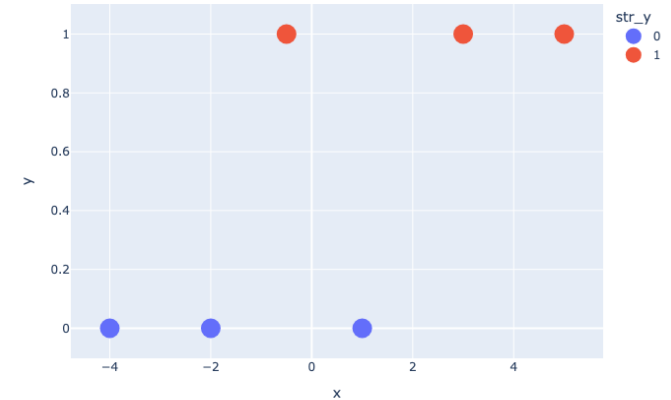
Toy Dataset: Squared Error

Logistic Regression
model:

$$p(C_1|\phi(\mathbf{x})) = \sigma(\mathbf{w}^T \phi(\mathbf{x}))$$

Assume no intercept.
So $\mathbf{x}$ and $\mathbf{w}$ are 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



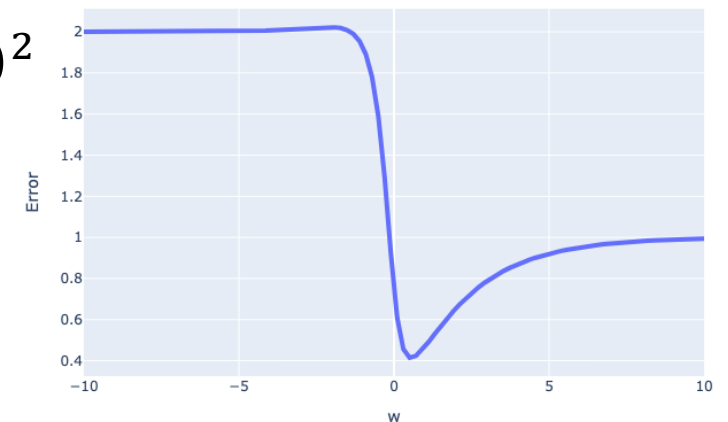
Squared Error:

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N (t_n - \sigma(\mathbf{w}^T \phi(\mathbf{x})))^2$$

The squared error
surface for logistic
regression has many
issues!



Error on Toy Classification Data



Pitfalls of Squared Error

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

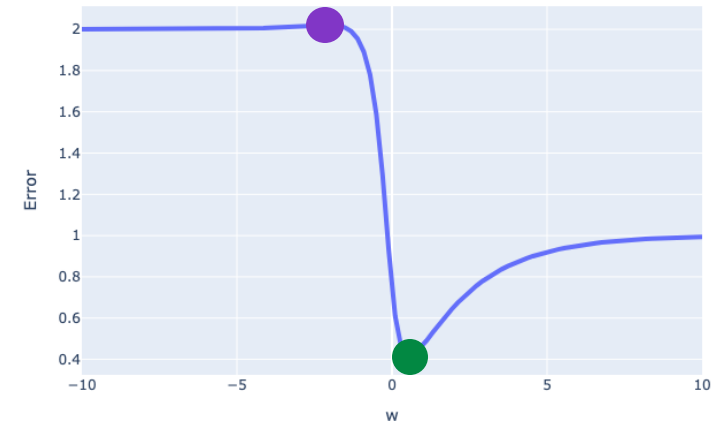
$$E(w) = \frac{1}{2} \sum_{n=1}^N (t_n - \sigma(w^T \phi(x)))^2$$



1965716

Secant line crosses function, so $E''(w)$ is not greater than 0 for all w .

Error on Toy Classification Data



```
from scipy.optimize import minimize
```

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

0.5446601825581691

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

-10.858380927026204

Pitfalls of Squared Error

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

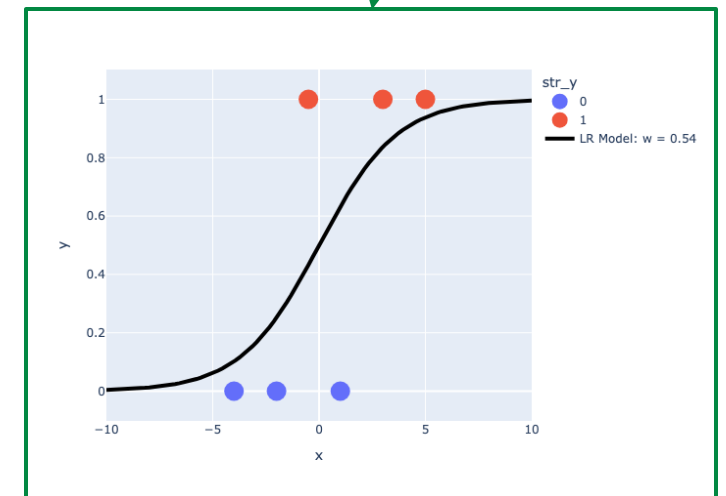
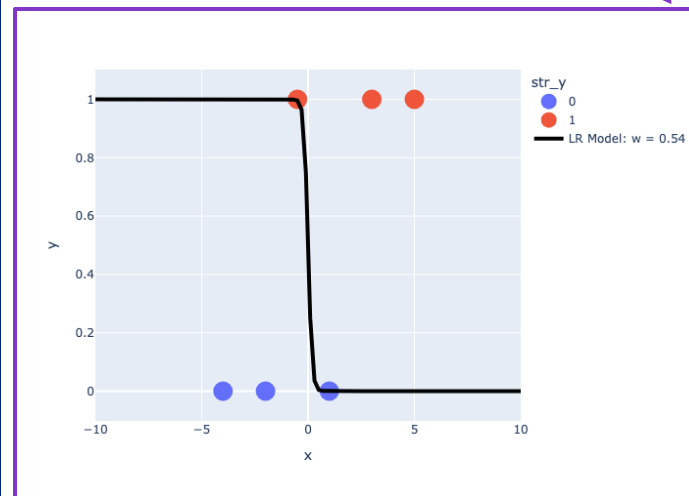
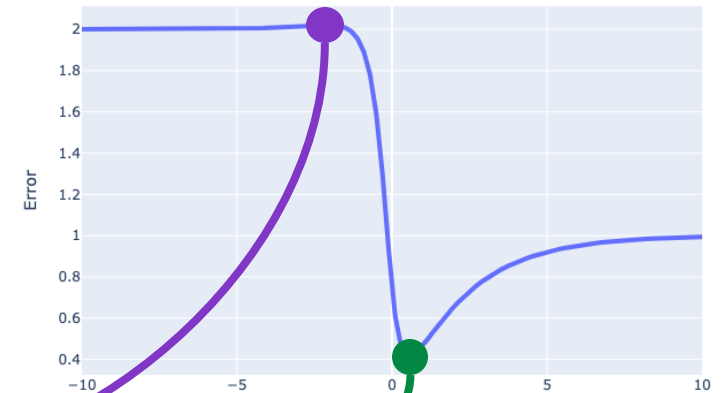
$$E(w) = \frac{1}{2} \sum_{n=1}^N (t_n - \sigma(w^T \phi(x)))^2$$



1965716

Secant line crosses function, so $E''(w)$ is not greater than 0 for all w .

Error on Toy Classification Data



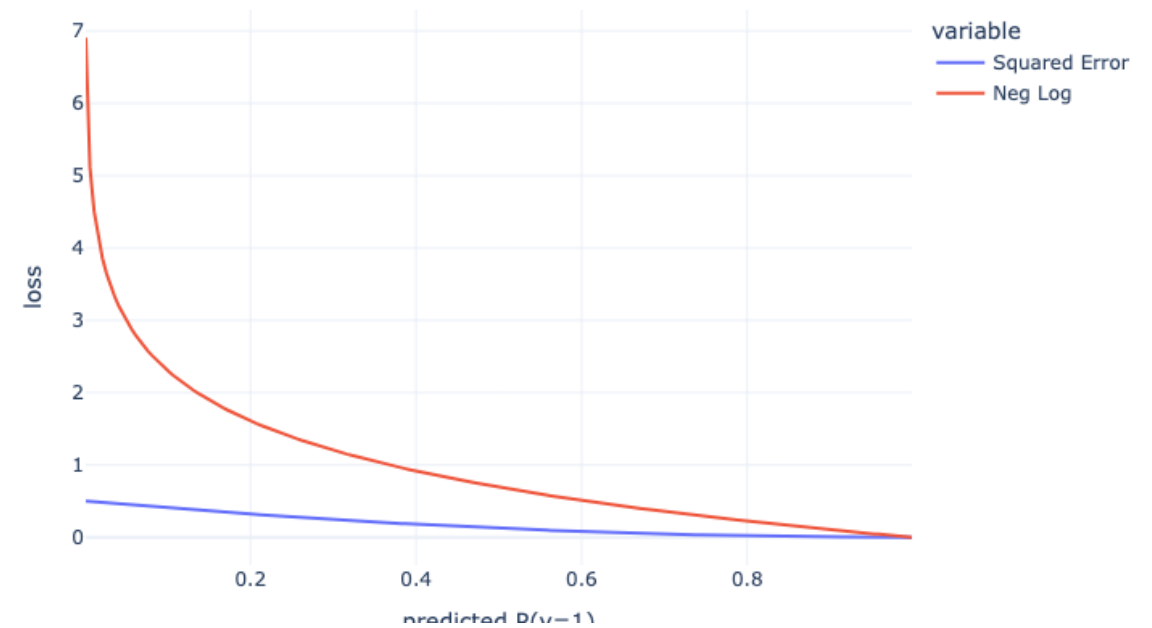


1965716

Squared Error Gives Bounded Penalty

- Best case scenario:
 - Estimate matches outcome.
 - Squared error = $(0 - 0)^2 = (1 - 1)^2 = 0$.
Perfect prediction!
- Worst case scenarios:
 - Estimated probability = 0, Actual outcome = 1.
 - Squared error = $(0 - 1)^2 = 1$
 - Estimated probability = 1, Actual outcome = 0.
 - Squared error = $(1 - 0)^2 = 1$
- The largest possible squared error is 1. Not a huge penalty for a big mistake!

Loss for One Point with $y = 1$



(With the $\frac{1}{2}$ used in $E(w)$, the maximum is $\frac{1}{2}$. Either way, the penalty is bounded.)

A New Error Function for Logistic Regression



1965716

The Logistic Regression Objective (MLE)

- In logistic regression, we can adopt **maximum likelihood estimation**.
- Define target variable $t \in \{0, 1\}$, with $t = 1$ representing class C_1 and $t = 0$ for class C_2 .
- The likelihood of t given $\mathbf{x}$ under our model $p(C_1|\mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x})$:

$$p(t|\mathbf{w}, \mathbf{x}) = \underbrace{\sigma(\mathbf{w}^\top \mathbf{x})^t}_{\text{for } t_n = 1, \text{ only this term stays}} \underbrace{[1 - \sigma(\mathbf{w}^\top \mathbf{x})]^{1-t}}_{\text{for } t_n = 0, \text{ only this term stays}}$$

for $t_n = 1$, only
this term stays

for $t_n = 0$, only
this term stays

The Logistic Regression Objective (MLE)



- In logistic regression, we can adopt **maximum likelihood estimation**.
- Define target variable $t \in \{0, 1\}$, with $t = 1$ representing class C_1 and $t = 0$ for class C_2 .
- The likelihood of t given $\mathbf{x}$ under our model $p(C_1|\mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x})$:

$$p(t|\mathbf{w}) = \sigma(\mathbf{w}^\top \mathbf{x})^t [1 - \sigma(\mathbf{w}^\top \mathbf{x})]^{1-t}$$

- For a dataset $\mathcal{D} = \{\mathbf{x}_n, t_n\}$, where $y_n = \sigma(\mathbf{w}^\top \mathbf{x})$ the likelihood function can be written as:

$$p(t|\mathbf{x}, \mathbf{w}, \mathcal{D}) = \prod_{n=1}^N y_n^{t_n} (1 - y_n)^{1-t_n}$$

The Logistic Regression Objective (MLE)



$$p(\textcolor{brown}{t}|\textcolor{violet}{w}, \mathcal{D}) = \prod_{n=1}^N y_n^{\textcolor{brown}{t}_n} (1 - y_n)^{1 - \textcolor{brown}{t}_n}$$

We can define an error function by taking the **negative logarithm** of the **likelihood**, which gives the **cross-entropy error function**.

$$E(w) = -\ln p(\textcolor{brown}{t}|\textcolor{violet}{w}, \mathcal{D}) = - \sum_{n=1}^N [\textcolor{brown}{t}_n \ln y_n + (1 - \textcolor{brown}{t}_n) \ln(1 - y_n)]$$

Cross-Entropy: Two Error Functions In One!



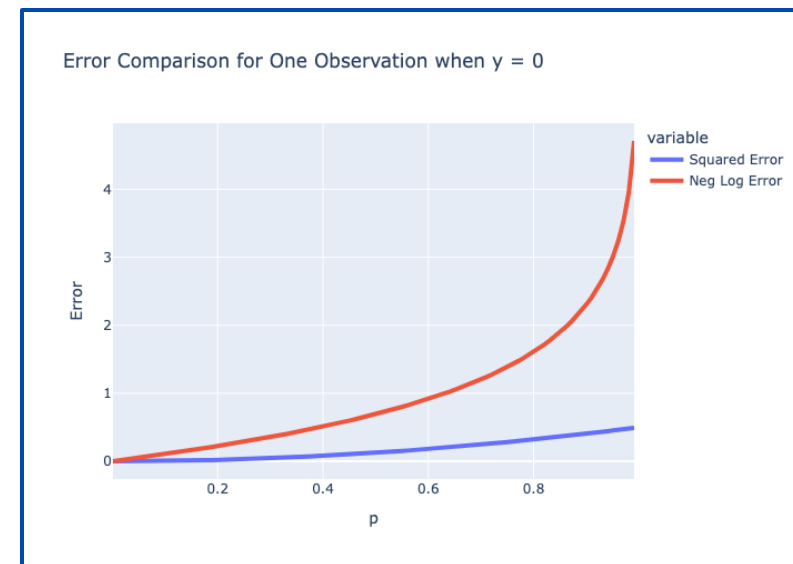
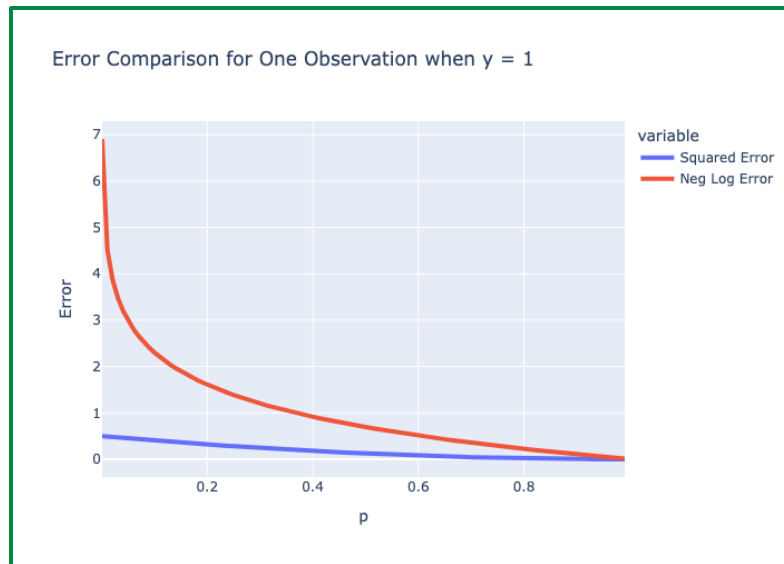
1965716

The Cross-Entropy (CE) function is often written more compactly as the **product of the label** and **the log probability**, summed over both classes.

$$\text{CE error} = \begin{cases} -\ln y_n, & \text{if } t_n = 1 \\ -\ln 1 - y_n & \text{if } t_n = 0 \end{cases} \leftarrow -[\underbrace{t_n \ln y_n}_{\text{for } t_n = 1, \text{ only this term stays}} + \underbrace{(1 - t_n) \ln(1 - y_n)}_{\text{for } t_n = 0, \text{ only this term stays}}]$$

for $t_n = 1$, only this term stays

for $t_n = 0$, only this term stays



Unbounded for wrong-way certainty



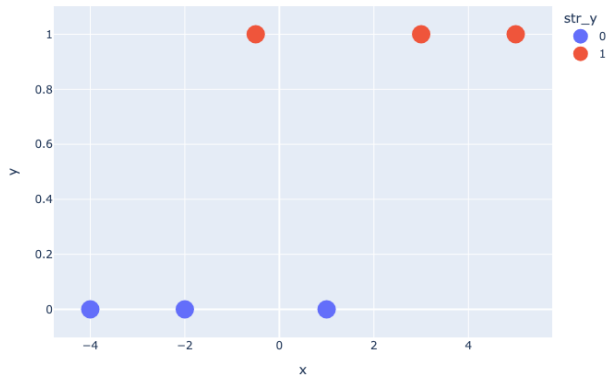
What w value will achieve 0 cross-entropy error if we train on the datapoint $x = 1, t = 1$?

Convexity By Picture



1965716

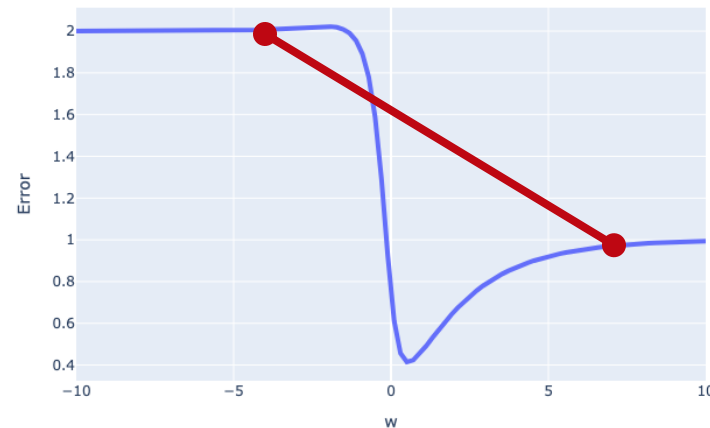
$$w = \underset{w}{\operatorname{argmin}} - \sum_{n=1}^N [t_n \ln y_n + (1 - t_n) \ln(1 - y_n)]$$



	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

Squared Error Surface

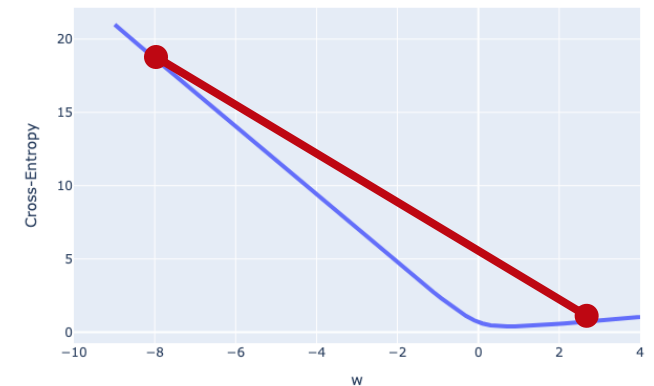
Error on Toy Classification Data



A straight line crosses
the curve Non-convex

Cross-Entropy Error Surface

Cross-Entropy on Toy Classification Data



Convex!

Lecture 08

Logistic Regression (1)

Credit: Joseph E. Gonzalez and Narges Norouzi

Reference Book Chapters: Chapter 4 Section 3 and Chapter 5 Sections 5.1, 5.2(up to 5.2.5), 5.4 (up to 5.4.4)