Introduction to Machine Learning in Geosciences GEO371/GEO39D.1

Linear and Logistic Regression

Mrinal K. Sen
Geosciences
UT Austin

September 26, 2023

Content

- Linear Regression
- Maximum Likelihood
- Bayesian Linear Regression
- Orthogonal Distance Regression

Introduction

Regression : Used for predicting continuous Outputs

- Inputs $\mathbf{x}$; Output y: $\mathbf{x} \in R^d$
- Training data: $(\mathbf{x}^i, y^i), i = 1, 2, ..., N$.
- A Model: $\hat{y} = f(\mathbf{x}, \mathbf{w}) = w_0 + w_1 x_1 + w_2 x_2 + ..$
- A **Loss Function** $L(\hat{y} - y) = ||y - \hat{y}||$.
- Optimization:
  - Analytic Solution
  - Convex Optimization
- Design good features [feature engineering] and feed them to a linear model.

Analytic Method

- Discussed in our **Lecture on Optimization.**
- We will revisit using ML notations:
  - Include bias into **x** by including 1, i.e.,

$$\mathbf{x}^{(i)} = [1, x_1^{(i)}, x_2^{(i)}, ..., x_d^{(i)}]$$

.

  - Target Vector

$$\mathbf{y} = [y^{(1)}, y^{(2)}, ..., y^{(N)}]^T$$

  - Design Matrix

$$\mathbf{X} = \{X_{ij}\} = \mathbf{x}_j^{(i)}$$

  - Prediction

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{w}$$

  - L2 Norm:

$$L(\mathbf{w}) = \frac{1}{2}(\hat{\mathbf{y}} - \mathbf{y})^T(\hat{\mathbf{y}} - \mathbf{y})$$

    Assumption: Loss function is Quadratic!
  - The Optimal solution (**Ordinary Least Sqaures**) is

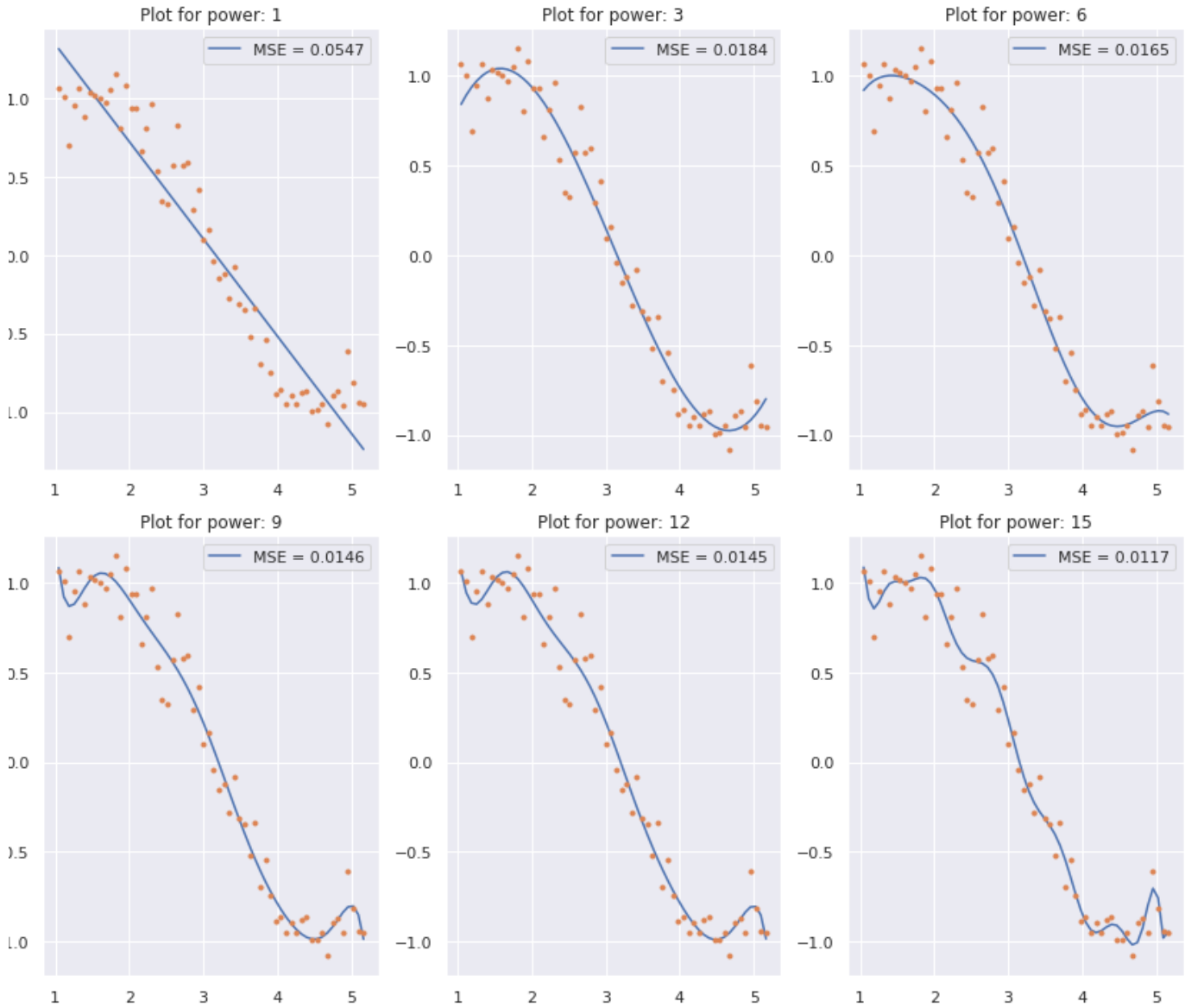
$$\mathbf{w}^* = \left[\mathbf{X}^T\mathbf{X}\right]^{-1}\mathbf{X}^T\mathbf{y}$$

Analytic Method

Models

- Consider one dimensional data.
- $f(x) = a_0 + a_1 x_1$
- $f(x) = a_0 + a_1 X + a_2 x^2$
- ..
- $f(x) = \sum_0^n a_n X^n$.
- All these are linear in model parameters a_i
- .
- .
- Which order polynomial?
- In the example, we try polynomials of order 1,..,15 on a noisy dataset

Linear Regression Example



Linear Regression Example

	rss	intercept	coef_x_1	coef_x_2	coef_x_3	coef_x_4	coef_x_5	coef_x_6	coef_x_7	coef_x_8	coef_x_9	coef_x_10	coef_x_11	coef_x_12	coef_x_13	coef_x_14	coef_x_15
model_pow_1	3.3	2	-0.62	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
model_pow_2	3.3	1.9	-0.58	-0.006	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
model_pow_3	1.1	-1.1	3	-1.3	0.14	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
model_pow_4	1.1	-0.27	1.7	-0.53	-0.036	0.014	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
model_pow_5	1	3	-5.1	4.7	-1.9	0.33	-0.021	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
model_pow_6	0.99	-2.8	9.5	-9.7	5.2	-1.6	0.23	-0.014	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
model_pow_7	0.93	19	-56	69	-45	17	-3.5	0.4	-0.019	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
model_pow_8	0.92	43	-1.4e+02	1.8e+02	-1.3e+02	58	-15	2.4	-0.21	0.0077	NaN	NaN	NaN	NaN	NaN	NaN	
model_pow_9	0.87	1.7e+02	-6.1e+02	9.6e+02	-8.5e+02	4.6e+02	-1.6e+02	37	-5.2	0.42	-0.015	NaN	NaN	NaN	NaN	NaN	
model_pow_10	0.87	1.4e+02	-4.9e+02	7.3e+02	-6e+02	2.9e+02	-87	15	-0.81	-0.14	0.026	-0.0013	NaN	NaN	NaN	NaN	
model_pow_11	0.87	-75	5.1e+02	-1.3e+03	1.9e+03	-1.6e+03	9.1e+02	-3.5e+02	91	-16	1.8	-0.12	0.0034	NaN	NaN	NaN	
model_pow_12	0.87	-3.4e+02	1.9e+03	-4.4e+03	6e+03	-5.2e+03	3.1e+03	-1.3e+03	3.8e+02	-80	12	-1.1	0.062	-0.0016	NaN	NaN	
model_pow_13	0.86	3.2e+03	-1.8e+04	4.5e+04	-6.7e+04	6.6e+04	-4.6e+04	2.3e+04	-8.5e+03	2.3e+03	-4.5e+02	62	-5.7	0.31	-0.0078	NaN	
model_pow_14	0.79	2.4e+04	-1.4e+05	3.8e+05	-6.1e+05	6.6e+05	-5e+05	2.8e+05	-1.2e+05	3.7e+04	-8.5e+03	1.5e+03	-1.8e+02	15	-0.73	0.017	
model_pow_15	0.7	-3.6e+04	2.4e+05	-7.5e+05	1.4e+06	-1.7e+06	1.5e+06	-1e+06	5e+05	-1.9e+05	5.4e+04	-1.2e+04	1.9e+03	-2.2e+02	17	-0.81	0

L2 Regularization

- Note the problem of over-fitting and underfitting in the previous example!
- Loss function

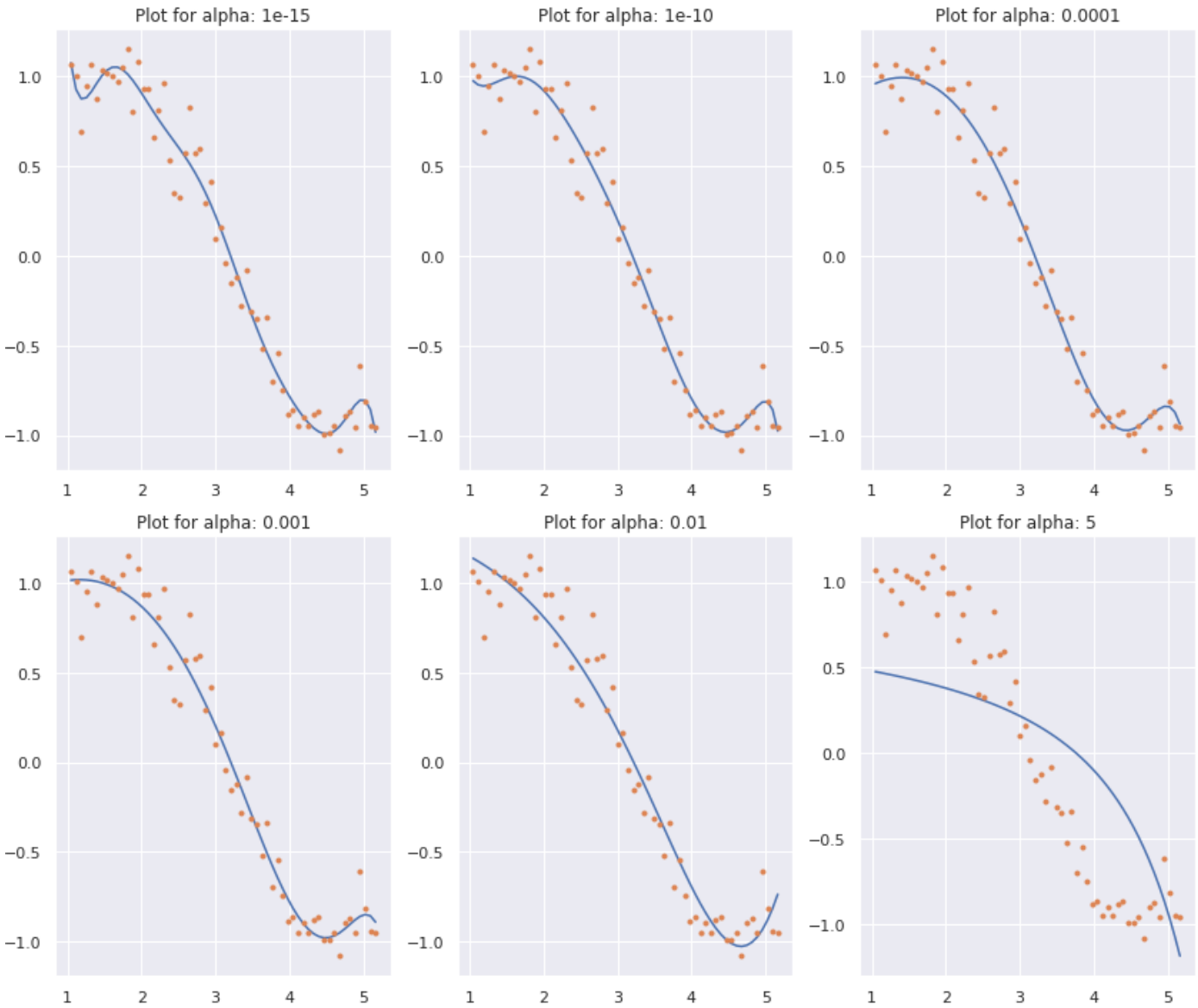
$$L(\mathbf{w}) = ||\mathbf{X}\mathbf{w} - \mathbf{y}||_2 + \lambda \mathbf{w}^T \mathbf{w}$$

- Solution:

$$\boxed{\mathbf{w}^* = \left[\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I}\right]^{-1} \mathbf{X}^T \mathbf{y}.}$$

- Equivalent to Gaussian Prior!
- Choice of λ is critical!

Ridge Regression Example



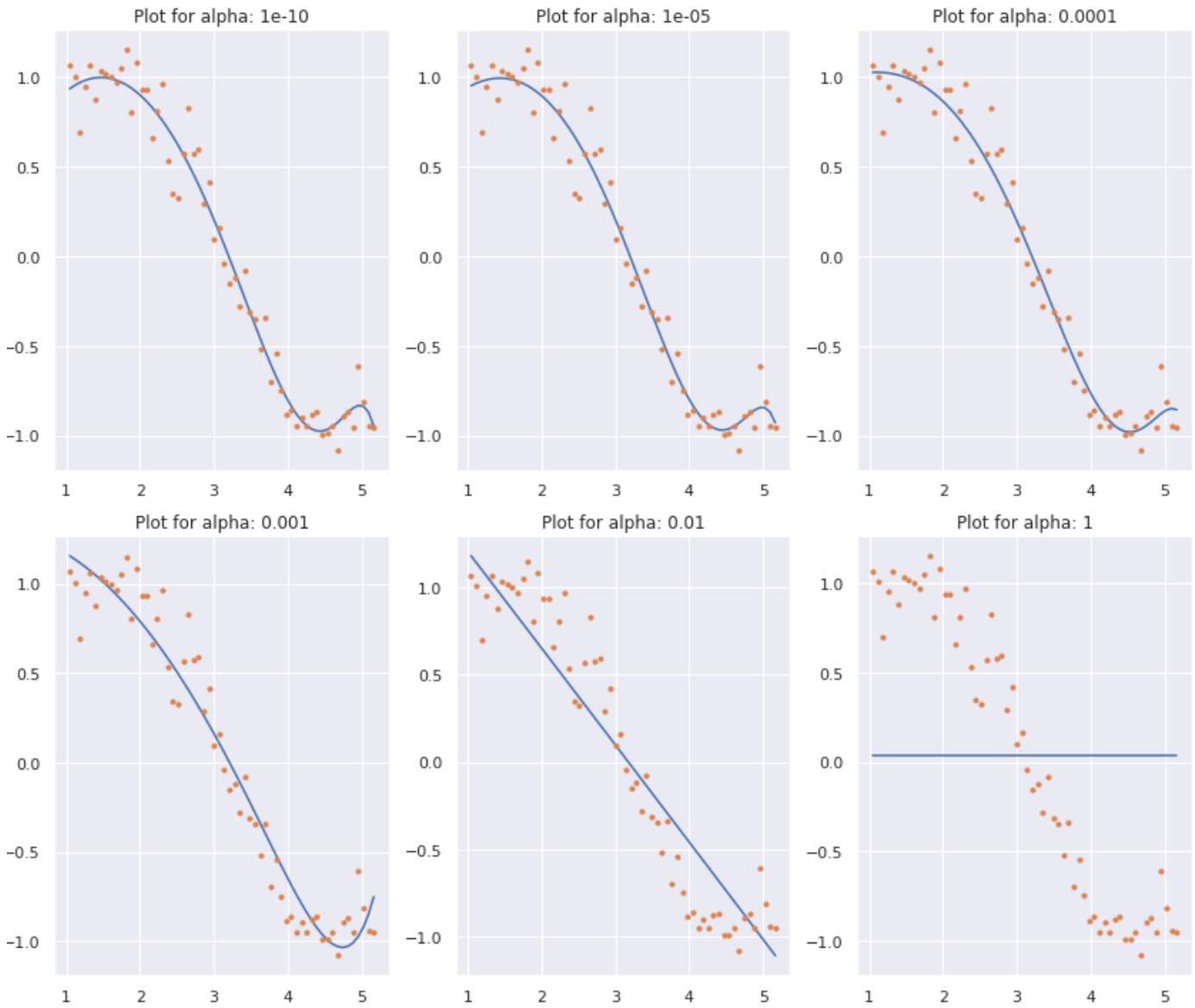
L1 Regularization

- Loss function

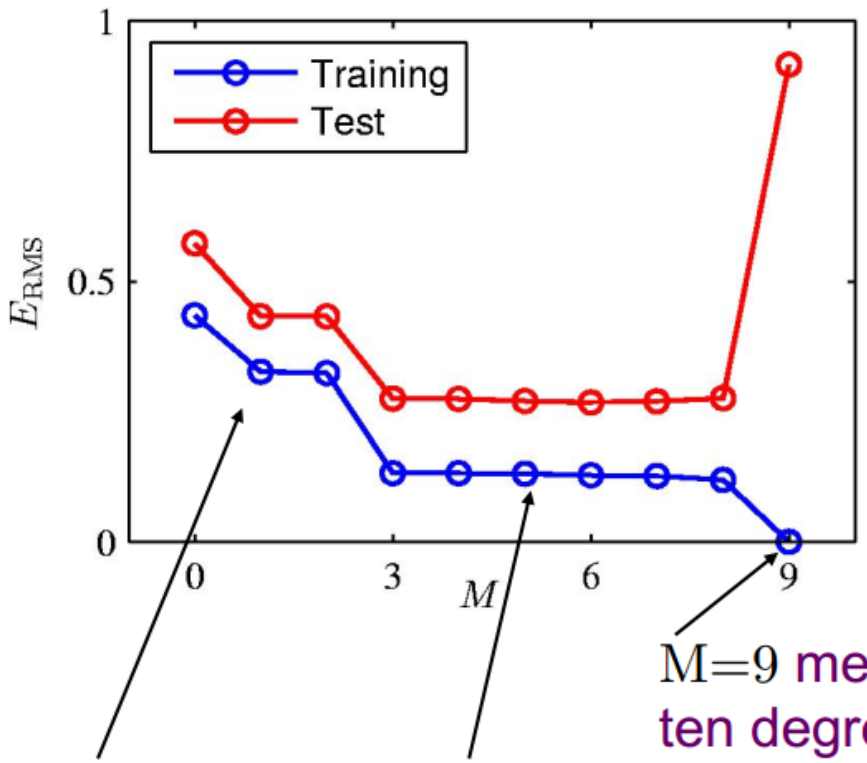
$$L(\mathbf{w}) = ||\mathbf{X}\mathbf{w} - \mathbf{y}||_2 + \lambda||\mathbf{w}||_1$$

- No Analytic Solution!
- Equivalent to Laplace Prior!
- Choice of λ is critical!
- **Lasso** (least absolute shrinkage and selection operator)

LASSO Example



Training and Test

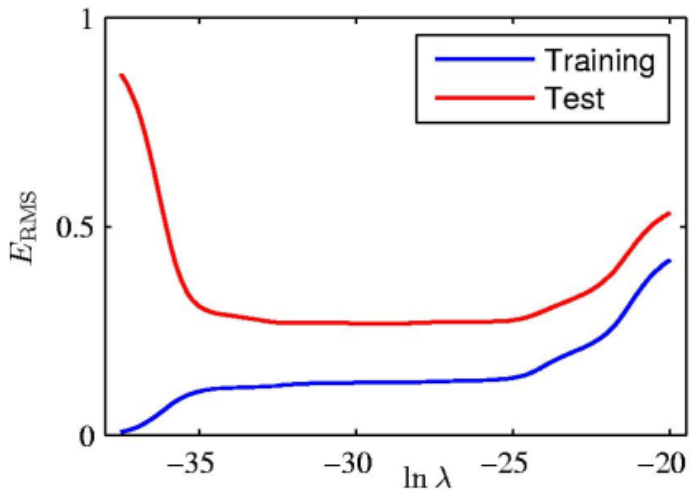


Poor due to Inflexible polynomials

Small Error

M=9 means ten degrees of freedom. Tuned exactly to 10 training points (wild oscillations in polynomial)

Training and Test



M=9 polynomial

Model Evaluation

- Model Evaluation: **R-square** It determines how much of the total variation in **y** (dependent variable) is explained by the variation in **X** (independent variable). Mathematically, it can be written as:

$$R_{square} = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - y_{mean})^2}$$

- The value of R-square is always between 0 and 1, where 0 means that the model does not model explain any variability in the target variable (y) and 1 meaning it explains full variability in the target variable.
- Mean Square Error.
- Adjusted R-square.

Ridge Regression and Lasso

- In L1 regularization, we penalize the absolute value of the weights while in L2 regularization, we penalize the squared value of the weights.
- In L1 regularization, we can shrink the parameters to zero while in L2 regularization, we can shrink the parameters to as small as possible but not to zero. So, L1 can simply discard the useless features in the dataset and make it simple.

Summary of Curve Fitting

- Partitioning data into training set (to determine coefficients w) and a separate validation set (or holdout set) to optimize model complexity N or λ
- More sophisticated approaches are not as wasteful of training data
- More principled approach is based on probability theory
- Classification is a special case of regression where target value is discrete value

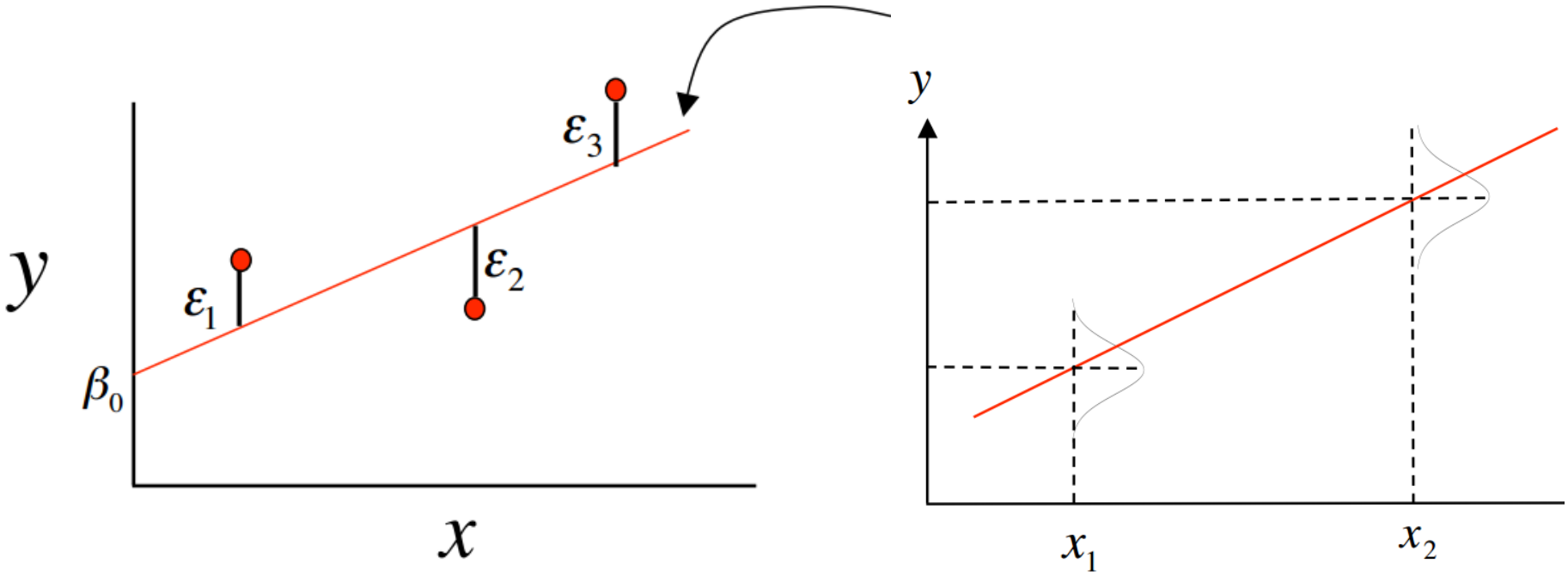
Statistical View

- Given the model $\hat{y}_i = a_0 + a_1 x_i$, the parameters a_0 and a_1 can be estimated from any given sample of the data. Therefore, we also need to consider their sampling distributions because each sample of (x_i, y_i) pairs will result in different estimates of a_0 and a_1.
- Let us consider the following distribution assumption on the error,

$$y_i = a_0 + a_1 x_i + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma^2).$$

- The above is now a statistical model that describes the distribution of y_i given x_i. Specifically, we assume the observed y_i is error-prone but centered around the linear model for each value of x_i, i.e, $y_i \sim N(a_0 + a_1 x_i, \sigma^2)$

Statistical View



Statistical View

The error distribution results in the following assumptions:

- **Homoscedasticity**: variance of y_i is the same for any x_i
- **Linearity**: the mean of y_i is linear with respect to x_i

$$E(y_i) = E(a_0 + a_1 x_i + \epsilon_i) = a_0 + a_1 x_i + E(\epsilon_i) = a_0 + a_1 x_i.$$

- **Independence**: y_i are independent of each other.
- A consequence of these assumptions is that the response variable y is independent across observations, conditional on the predictor y, i.e., y_1 and y_2 are independent given x_1 and x_2.

Statistical View - Maximum Likelihood

- Based on these assumptions, the model gives us the conditional pdf of y for each x, $p(y|x;\alpha_0,\alpha_1,\sigma^2)$.
- Given any data set $(x_1,y_1),(x_2,y_2),...(x_n,y_n)$, we can now write down the probability density, under the model, of seeing that data:

$$\prod_{i=1}^{n} p(y_i|x_i;\alpha_0,\alpha_1,\sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i-(\alpha_0+\alpha_1 x_i))^2}{2\sigma^2}}$$

This is called the likelihood function

$$L(\mathbf{y}|\mathbf{x};\alpha_0,\alpha_1,\sigma^2)) = \frac{1}{\sqrt{2\pi\sigma^2}} \prod_{i=1}^{n} e^{-\frac{(y_i-(\alpha_0+\alpha_1 x_i))^2}{2\sigma^2}}$$

- Maximizing the likelihood (Maximum Likelihood is the same as maximizing $e^{-\frac{(y_i-(\alpha_0+\alpha_1 x_i))^2}{2\sigma^2}}$.

Statistical View - Maximum Likelihood

- Finally, if we turn the maximization into a minimization by changing sign and ignore the constant factor of $\frac{1}{2}$, the problem becomes

$$\min \sum_{i}^{n} \frac{(y_i - (\alpha_0 + \alpha_1 x_i))^2}{\sigma^2}.$$

  This is the same as the least squares problem (without the σ^2 term)!

- The sum of the squares of the residuals also provides useful statistical information about the quality of model estimates obtained with least squares. The **chi-square statistic** is

$$\chi^2_{obs} = \sum_{i}^{n} \frac{(y_i - (\alpha_0^{LS} + \alpha_1^{LS} x_i))^2}{\sigma^2}.$$

  Since χ^2 depends on the random measurement errors in y, it is itself a random variable.

Statistical View - Maximum Likelihood

- In matrix notation $\mathbf{y} = \mathbf{X}\mathbf{w}$.
- The least squares solution is $\mathbf{w}^{LS} = [\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T\mathbf{y}$.
- If the data errors are normally distributed (as assumed here), then the parameter estimates will also be normally distributed because a linear combination of normally distributed random variables is normally distributed!
- $\mathbf{w}\mathbf{w}^T = (\mathbf{X}\mathbf{w})^T = [\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T\mathbf{y}\mathbf{y}^T\mathbf{X}\{[\mathbf{X}^T\mathbf{X}]^{-1}\}^T = [\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T\mathbf{y}\mathbf{y}^T\mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-1} = [\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T Cov(\mathbf{y})\mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-1}$.
- We assume that $Cov(\mathbf{y}) = \sigma^2\mathbf{I}$. Thus, we have

$$\boxed{cov(\mathbf{w}) = \sigma^2[\mathbf{X}^T\mathbf{X}]^{-1}.}$$

- We can compute 95% confidence intervals for individual model parameters using the fact that each model parameter w_i has a normal distribution. The 95% confidence intervals are given by

$$\mathbf{w}^{LS} \pm 1.96\text{diag}(\text{Cov}(\mathbf{w}))^{1/2}.$$

Statistical View - Maximum Likelihood

- The factor 1.96 arises from

$$\frac{1}{\sigma\sqrt{2\pi}} \int_{-1.96\sigma}^{1.96\sigma} e^{-\frac{x^2}{2\sigma^2}} dx \approx 0.95.$$

The Bayesian Approach: Prior, Likelihood and Posterior Probabilities

- **Bayes' Theorem**: Posterior = Likelihood X Prior.

$$P(\mathbf{w}|\mathbf{y}) \propto P(\mathbf{y}|\mathbf{w})P(\mathbf{w})$$

- **Likelihood**: We assume Gaussian errors in data (not necessary)

$$P(\mathbf{y}|\mathbf{w}) \propto e^{-(\mathbf{y}-\mathbf{X}\mathbf{w})^T \mathbf{C}_y^{-1}(\mathbf{y}-\mathbf{X}\mathbf{w})}$$

- **Prior**: We assume Gaussian prior of the model parameters.

$$P(\mathbf{w}) \propto e^{-(\mathbf{w}-\mathbf{w}_0)^T \mathbf{C}_w^{-1}(\mathbf{w}-\mathbf{w}_0)}$$

$\mathbf{C}_y$ prior data covariance matrix: noise

$\mathbf{C}_w$ prior model covariance matrix

$\mathbf{w}_0$ prior mean model

The Bayesian Approach: Prior, Likelihood and Posterior Probabilities

- **Bayes' Theorem**: Posterior = Likelihood X Prior.

$$P(\mathbf{w}|\mathbf{y}) \propto e^{-(\mathbf{y}-\mathbf{X}\mathbf{w})^T \mathbf{C}_y^{-1}(\mathbf{y}-\mathbf{X}\mathbf{w})} e^{-(\mathbf{w}-\mathbf{w}_0)^T \mathbf{C}_w^{-1}(\mathbf{w}-\mathbf{w}_0)}$$

$$P(\mathbf{w}|\mathbf{y}) \propto e^{\left[-(\mathbf{y}-\mathbf{X}\mathbf{w})^T \mathbf{C}_y^{-1}(\mathbf{y}-\mathbf{X}\mathbf{w})-(\mathbf{w}-\mathbf{w}_0)^T \mathbf{C}_w^{-1}(\mathbf{w}-\mathbf{w}_0)\right]}$$

$$P(\mathbf{w}|\mathbf{y}) \propto e^{-L(\mathbf{w})}$$

$$\boxed{L(\mathbf{w}) = (\mathbf{y}-\mathbf{X}\mathbf{w})^T \mathbf{C}_y^{-1}(\mathbf{y}-\mathbf{X}\mathbf{w}) + (\mathbf{w}-\mathbf{w}_0)^T \mathbf{C}_w^{-1}(\mathbf{w}-\mathbf{w}_0)}$$

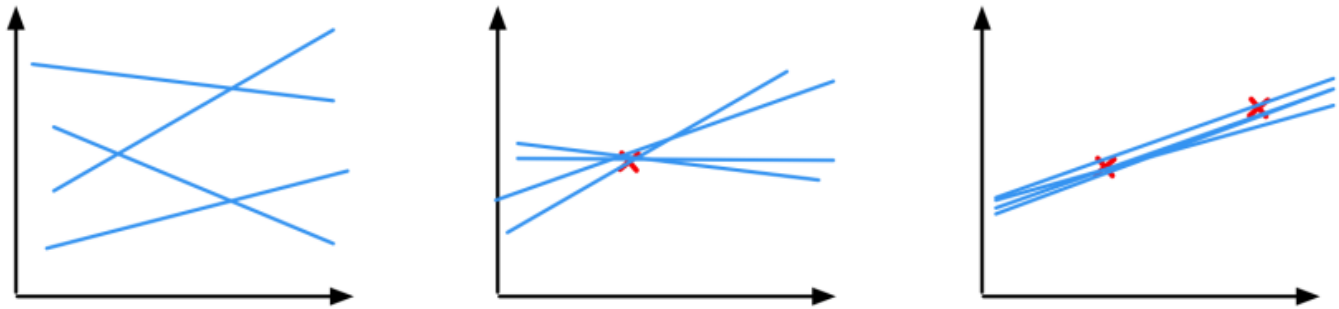
- A regularized loss function!
- The posterior is also Gaussian!

$$P(\mathbf{w}|\mathbf{y}) \propto e^{-\left[(\mathbf{w}-\tilde{\mathbf{w}})^T \tilde{\mathbf{C}}_w (\mathbf{w}-\tilde{\mathbf{w}})\right]}$$

$\tilde{\mathbf{w}}$: Posterior Mean; $\tilde{\mathbf{C}}_w$: Posterior Model Covariance matrix

The Bayesian Approach

- Full Bayesian inference makes predictions of all likely explanations under the posterior distribution.
- Bayesian linear regression considers various plausible explanations for how the data were generated.
- It makes predictions using all possible regression weights, weighted by their posterior probability.



The Bayesian Approach

- We can derive the **posterior mean and covariance** analytically!
- Mean: $\boxed{\tilde{\mathbf{w}} = \mathbf{w}_0 + + \left[\mathbf{X}^T\mathbf{C}_y^{-1}\mathbf{X} + \mathbf{C}_w^{-1}\right]^{-1}\mathbf{X}^T\mathbf{C}_y^{-1}(\mathbf{y} - \mathbf{X}\mathbf{w})}$
- Covariance $\boxed{\tilde{\mathbf{C}}_w = \left(\mathbf{X}^T\mathbf{C}_y^{-1}\mathbf{X} + \mathbf{C}_w^{-1}\right)^{-1}}$
- Now that we have an expression for the posterior pdf, what do we do?
- Find the maximum of the posterior - **MAP - maximum a posteriori?**
- Same as finding the minimum of the negative of the log of the posterior, which is the same as the finding the minimum of the loss function:

$$\boxed{L(\mathbf{w}) = \frac{1}{2}\left[(\mathbf{y} - \mathbf{X}\mathbf{w})^T\mathbf{C}_y^{-1}(\mathbf{y} - \mathbf{X}\mathbf{w}) + (\mathbf{w} - \mathbf{w}_0)^T\mathbf{C}_w^{-1}(\mathbf{w} - \mathbf{w}_0)\right]}$$

- Minimization of $L(\mathbf{w})$ yields

$$\mathbf{w}_{MAP} = \tilde{\mathbf{w}} = \mathbf{w}_0 + \left[\mathbf{X}^T\mathbf{C}_y^{-1}\mathbf{X} + \mathbf{C}_w^{-1}\right]^{-1}\mathbf{X}^T\mathbf{C}_y^{-1}(\mathbf{y} - \mathbf{X}\mathbf{w})$$

The Bayesian Approach

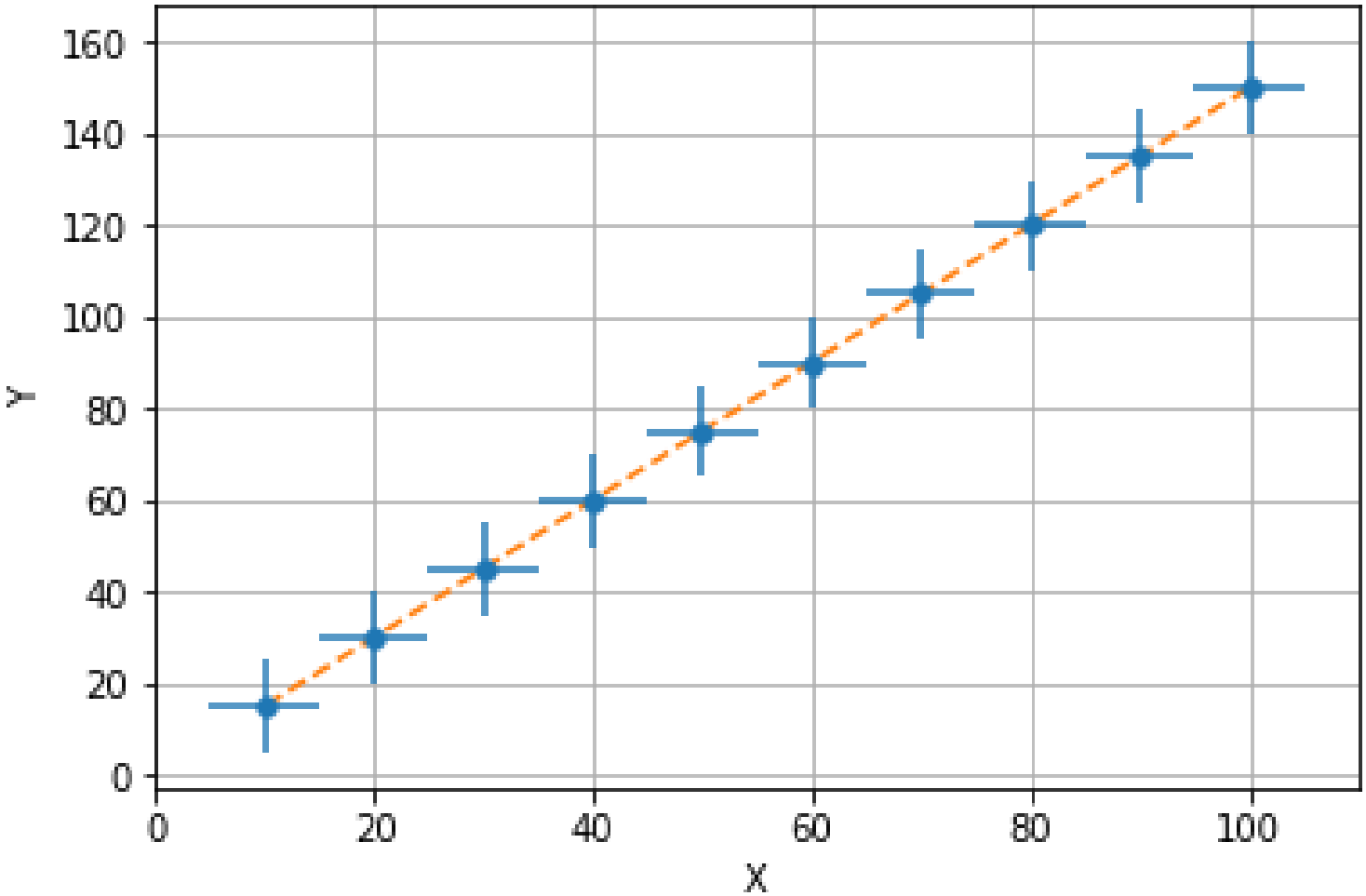
- In general, we draw samples of $\mathbf{w}$ from the posterior. Generally difficult when the posterior is multi-modal (non-Gaussian).
- The log of the posterior or the loss function can be viewed as a regularized L_2 norm when $\mathbf{C}_w$, and $\mathbf{C}_y$ are the model and data weights respectively – **Weighted Least Sqauares**!
- Special Cases:
  - Large model Prior bound; zero model norm and $\mathbf{C}_y = \mathbf{I}$;

$$\mathbf{w}_{MAP} = \left[\mathbf{X}^T\mathbf{X}\right]^{-1}\mathbf{X}^T\mathbf{y}$$

  - $\mathbf{w}_0 = 0$, $\mathbf{C}_w = \frac{1}{\epsilon^2}\mathbf{I}$ and $\mathbf{C}_y = \mathbf{I}$;

$$\mathbf{w}_{MAP} = \left[\mathbf{X}^T\mathbf{X} + \epsilon^2\mathbf{I}\right]^{-1}\mathbf{X}^T\mathbf{y}$$

Bayesian Regression with uncertainties in both axes



Bayesian Regression with uncertainties in both axes

- We have errors in both **x**, and **y**.
- Use the following model: $\boxed{w_0 y + w_1 x = 1.}$
- Introduce the parameter vector:

$$\mathbf{m} = [\mathbf{y} \;\; \mathbf{x} \;\; \mathbf{w}]^T = [y_1, y_2, ..., x_1, x_2, ..., w_0, w_1]^T.$$

- For each realization of **m**, we define a vector $\mathbf{d} = [d_1, d_2, ...]^T$, and
- $\hat{d}_i = w_0 y_i + w_1 x_i, \quad i = 1, 2, ...$
- Note that $\mathbf{d} = [1, 1, 1, , ...]^T$., and $\mathbf{m}_0$: prior parameter vector.

Bayesian Regression with uncertainties in both axes

- Prior Covariance matrix:

$$\mathbf{C}_m = \begin{bmatrix} \mathbf{C}_y & 0 & 0 & 0 \\ 0 & \mathbf{C}_x & 0 & 0 \\ 0 & 0 & \sigma_{w_0}^2 & 0 \\ 0 & 0 & 0 & \sigma_{w_1}^2 \end{bmatrix}$$

- Posterior Probability

$$P(\mathbf{m}) \propto \exp(-L(\mathbf{m})),$$

$$2L(\mathbf{m}) = (\hat{\mathbf{d}} - \mathbf{d})^T \mathbf{C}_d^{-1} (\hat{\mathbf{d}} - \mathbf{d}) + (\mathbf{m} - \mathbf{m}_0)^T \mathbf{C}_m^{-1} (\mathbf{m} - \mathbf{m}_0).$$

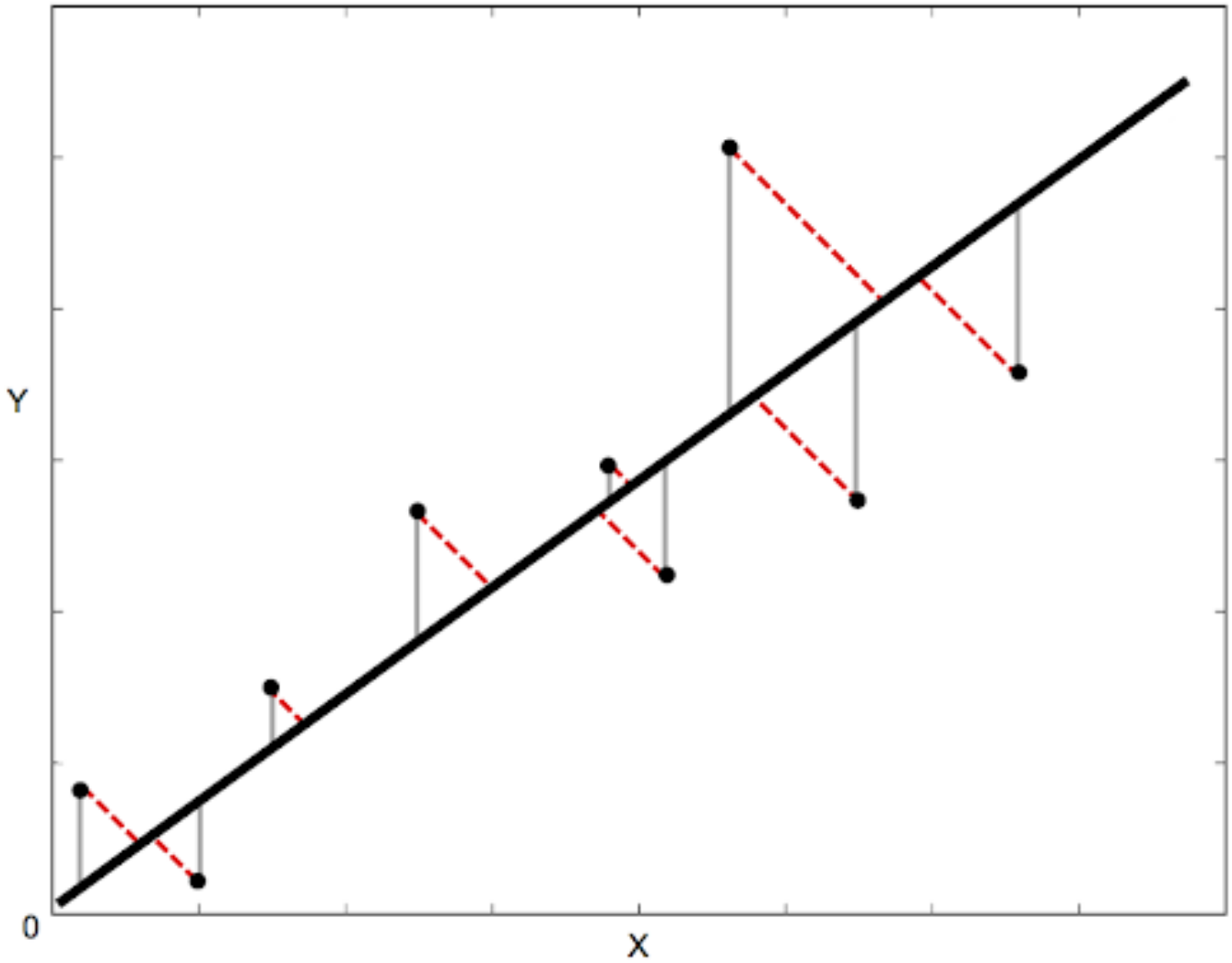
Bayesian Regression with uncertainties in both axes

- MAP estimation: Minimize $L(\mathbf{m})$ - no analytic solution. Use GD or some other optimization method.

$$\frac{\partial \hat{\mathbf{d}}}{\partial \mathbf{m}} = [w_0 \mathbf{I} \;\; w_1 \mathbf{I} \;\; \mathbf{y} \;\; \mathbf{x}]$$

- Posterior is not a Gaussian! Use MCMC or Variational Bayes to draw samples of $\mathbf{m}$ from the posterior distribution!

Bayesian Regression with uncertainties in both axes - Orthogonal Distance Regression



Regression with multiple inputs

- Predict $\mathbf{y}$ for a given $\mathbf{x} = [x_1, x_2, ..., x_D]$.
- Linear functions of model variables $\mathbf{w}$
- Polynomial curve fitting as described for 1D is not easily generalized to $\mathbf{x} = [x_1, x_2, ..., x_D]$.
- We can write $y(\mathbf{x}, \mathbf{w}) = w_0 + w_1 x_1 + w_2 x_2 + ... + w_s x_d$.
- Note that this is a straight line for single dimension and a plane in 2D, and so on ....
- It is not simple to extend the polynomial to mutlti-dimensional case to make it a nonlinear function of $\mathbf{x}$,

Basis Functions

- In many applications, we apply some form of fixed-preprocessing, or **feature extraction**, to the original data variables.
- If the original variables comprise the vector $\mathbf{x}$, then the features can be expressed in terms of basis functions $\{\phi_j(\mathbf{x})$
  - By using nonlinear basis functions we allow the function $y(x, \mathbf{w})$ to be a nonlinear function of the input vector $\mathbf{x}$.
  - They are linear functions of parameters (gives them simple analytical properties), yet are nonlinear with respect to input variables (similar to higher order polynomial fitting in 1D).

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^{M-1} w_i \phi_i(\mathbf{x}) = \sum_{i=1}^{M-1} w_i \phi_i(\mathbf{x})$$

  N is not necessarily equal to D.
  - Basis functions allow non-linearity with D input variables.

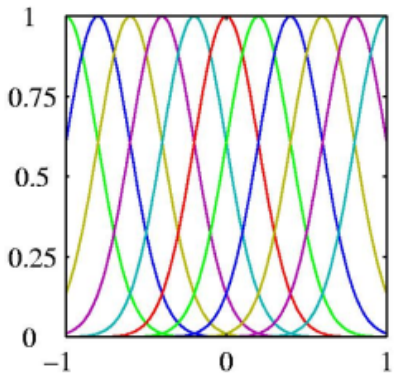
Basis Functions

- Polynomial Regression (good for 1D)
- Gaussian Basis Functions
- Sigmoidal Basis Functions
- Fourier Basis Functions
- Wavelets

Gaussian Radial Basis Functions

2. Gaussian Radial Basis Functions

- Gaussian $$\phi_j(x) = \exp\left(\frac{(x-\mu_j)^2}{2\sigma^2}\right)$$
  - Does not necessarily have a probabilistic interpretation
  - Usual normalization term is unimportant
    - since basis function is multiplied by weight w_j



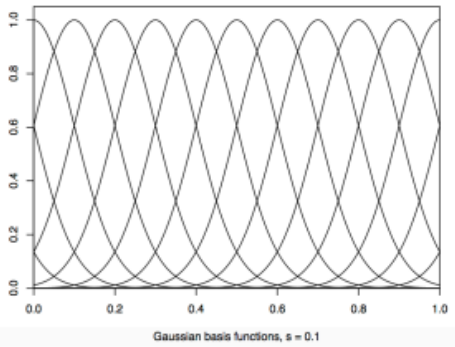
- Choice of parameters
  - μ_j govern the locations of the basis functions
    - Can be an arbitrary set of points within the range of the data
      - Can choose some representative data points
  - σ governs the spatial scale
    - Could be chosen from the data set e.g., average variance
- Several variables
  - A Gaussian kernel would be chosen for each dimension
  - For each j a different set of means would be needed– perhaps chosen from the data

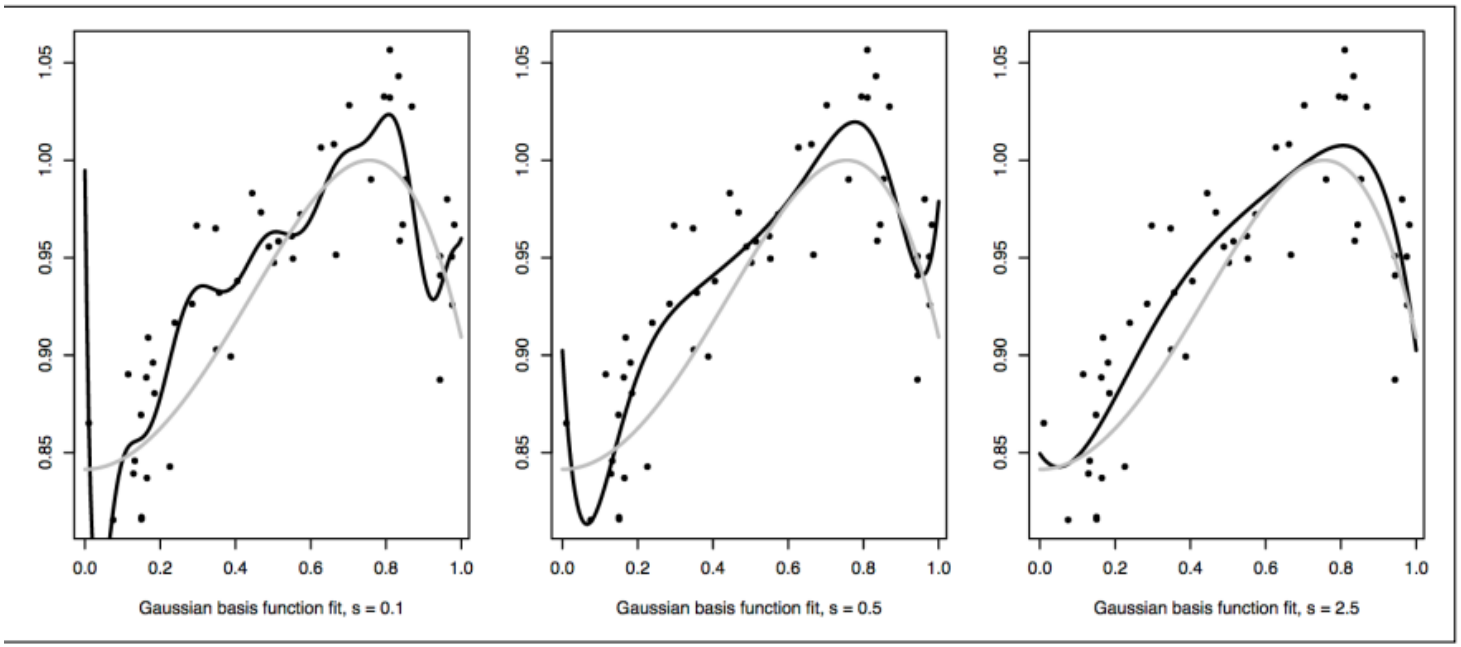
$$\phi_j(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^t\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)\right)$$

Result with Gaussian Basis Functions

$$\phi_j(x) = \exp(-(x-\mu_j)^2/2s^2)$$

Basis functions for s=0.1, with the μ_j on a grid with spacing s





w_j s for middle model:

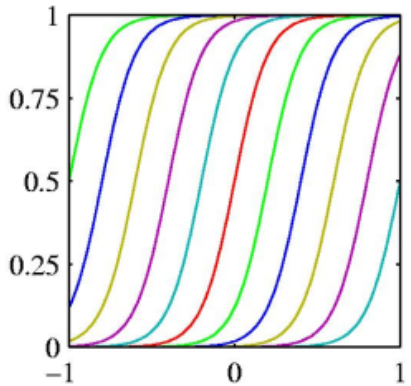
6856.5
-3544.1
-2473.7
-2859.8
-2637.7
-2861.5
-2468.0
-3558.4

3. Sigmoidal Basis Function

- Sigmoid $$\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right) \quad \text{where} \quad \sigma(a) = \frac{1}{1+\exp(-a)}$$
- Equivalently, tanh because it is related to logistic sigmoid by

$$\tanh(a) = 2\sigma(a) - 1$$

Logistic Sigmoid
For different μ_j



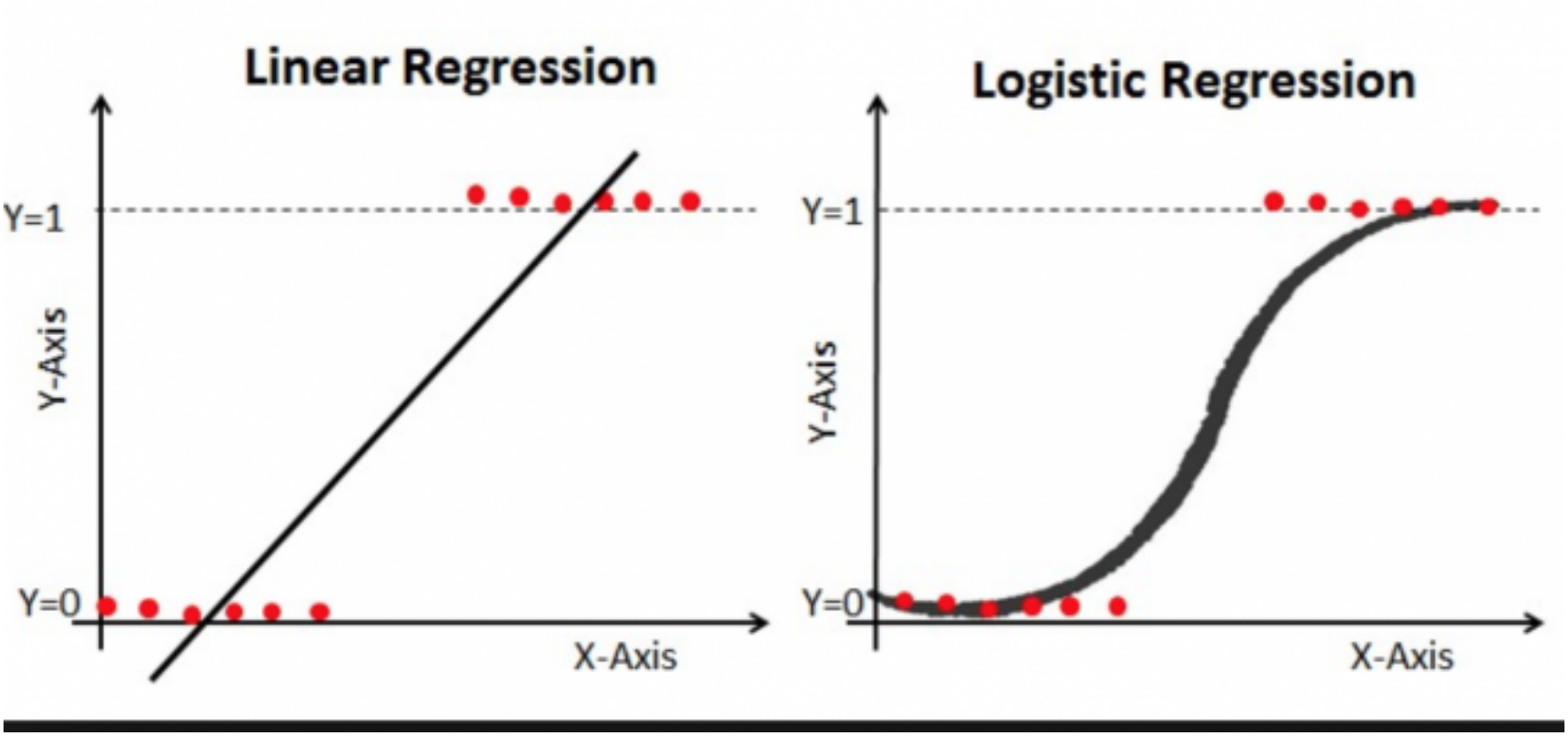
4. Other Basis Functions

- Fourier
  - Expansion in sinusoidal functions
  - Infinite spatial extent
- Signal Processing
  - Functions localized in time and frequency
  - Called *wavelets*
    - Useful for lattices such as images and time series

Logistic Regression

- Consider the **binary classification problem**
- y can take only two values, 0 and 1.
- **x** is a vector of real-valued features. $\mathbf{x} = [x_1.x_2, x_3, ...]^T$.
- We could approach the classification problem ignoring the fact that y is discrete-valued, and use our old linear regression algorithm to try to predict y given **x**.
- However, it doesn't make sense for f(x) to possibly take values larger than 1 or smaller than 0 when we know that $y \in \{0, 1\}$.

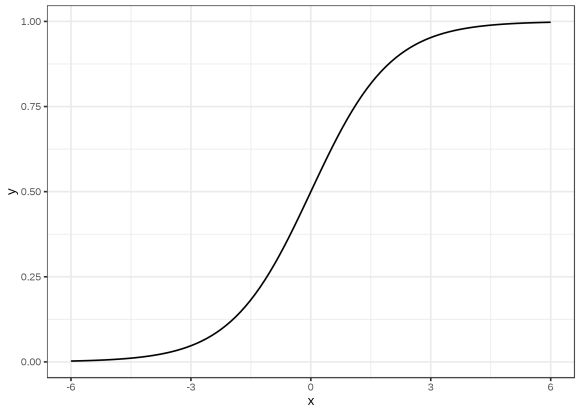
Logistic Regression



Logistic Regression

- A solution for classification is logistic regression. Instead of fitting a straight line or hyperplane, the logistic regression model uses the logistic function to squeeze the output of a linear equation between 0 and 1.
- The logistic function is defined as:

$$\text{logistic}(\eta) = \frac{1}{1 + \exp(-\eta)}$$



Logistic Regression

- In the linear regression model, we have modelled the relationship between outcome and features with a linear equation:

$$\hat{y}^{(i)} = w_0 + w_1 x_1^{(i)} + w_2 x_2^{(i)} + ... + w_n x_n(i).$$

- For classification, we prefer probabilities between 0 and 1, so we wrap the right side of the equation into the logistic function:

$$p(y^{(i)} = 1) = \frac{1}{1 + \exp\left(-\left(w_0 + w_1 x_1^{(i)} + w_2 x_2^{(i)} + ... + w_n x_n(i)\right)\right)}.$$

This forces the output to assume values between 0 and 1.

Logistic Regression

- We can generalize linear regression to the (binary) classification setting by making two changes.
- First, we replace the Gaussian distribution for y with a Bernoulli distribution,which is more appropriate for the case when the response is binary,$y \in \{0, 1\}$. That is, we use

$$p(y|\mathbf{x}, \mathbf{w}) = \text{Ber}(y|\mu(\mathbf{x})),$$

$$\mu(\mathbf{x}) = E(y|\mathbf{x}) = p(y = 1|\mathbf{x}).$$

- Second, we compute a linear combination of the inputs, as before, but then we pass this through a function that ensures $0 \leq (\mathbf{x}) \leq 1$ by defining (sigmoid/logistic/logit)

$$\mu(\mathbf{x}) = \text{sigm}(\mathbf{w}^T\mathbf{x}); \;\; \text{sigm}(\eta) = \frac{1}{1 + \exp(-\eta)} = \frac{e^\eta}{e^\eta + 1}.$$

- Thus,

$$p(y|\mathbf{x}, \mathbf{w}) = \text{Ber}(y|\text{sigm}(\mathbf{w}^T\mathbf{x})).$$

Discrete Distribution: Binomial or Bernoulli Distribution

- Suppose we toss a coin n times. Let $X \in \{0, ..., n\}$ be the number of heads. If the probability of heads is θ, then we say X has a **binomial distribution**, written as $X \sim \text{Bin}(n, \theta)$. The pmf[1] is given by

$$\text{Bin}(k|n, \theta) \triangleq \binom{n}{k} \theta^k (1-\theta)^{n-k}.$$

- Mean $= \theta$; var$=n\theta(1-\theta)$

[1] a probability mass function (PMF) is a function that gives the probability that a discrete random variable is exactly equal to some value.

Logistic Regression - Likelihood function

- We use maximum likelihood estimation (MLE). As such we are going to have two steps:
  - write the log-likelihood function.
  - find the values of w that maximize the log-likelihood function
- Note that the labels that we are predicting are binary, and the output of our logistic regression function is supposed to be the probability that the label is one.
- This means that we can (and should) interpret each label as a Bernoulli random variable: $Y \sim \text{Ber}(p)$, where $p = \sigma(w^T \mathbf{x})$.
- Probability of one data point (recall this is the equation form of the probability mass function of a Bernoulli):

$$P(Y = y|X = \mathbf{x}) = \sigma(w^T \mathbf{x})^y \left[1 - \sigma(w^T \mathbf{x})\right]^{(1-y)}$$

Logistic Regression - Likelihood function

- We can write the likelihood of all the data:

$$L(\mathbf{w}) = \prod_{i=1}^{n} P(Y = y = y^{(i)} | X = \mathbf{x}^{(i)})$$

$$L(\mathbf{w}) = \prod_{i=1}^{n} \sigma(w^T \mathbf{x}^{(i)})^{y^{(i)}} \left[1 - \sigma(w^T \mathbf{x}^{(i)})\right]^{(1-y)^{(i)}}$$

- The log likelihood equation is:

$$LL(\mathbf{w}) = \sum_{i=1}^{n} y^{(i)} \log \sigma(w^T \mathbf{x}^{(i)}) + (1 - y^{(i)}) \log[1 - \sigma(w^T \mathbf{x}^{(i)})].$$

$$\frac{\partial LL(\mathbf{w})}{\partial w_j} = \sum_{i=1}^{n} y^{(i)}[y^{(i)} - \sigma(w^T \mathbf{x}^{(i)})] x_j^i.$$

Logistic Regression - Maximum Likelihood

$$w_j^{new} = w_j^{old} + \eta \frac{\partial LL(\mathbf{w}^{old})}{\partial w_j^{old}}$$

$$w_j^{new} = w_j^{old} + \eta \sum_{i=1}^{n} y^{(i)}[y^{(i)} - \sigma(w^T \mathbf{x}^{(i)})]x_j^i.$$