

# IFI 9000 Analytics Methods

## Introduction

by **Houping Xiao**

January 12<sup>th</sup>, 2021



# About the Course

# Instructor

- **Instructor:** Houping Xiao
- **Time:** Tuesdays 9:30AM - 12:00PM
- **Location:** Building 1108 @ 55 Park Place
- **Virtual Meetings:**
  - Meeting  
link:<https://gsumeetings.webex.com/gsumeetings/j.php?MTID=m6761e531a2782e3368c6730152462c34>
- **Office:** Building 1640 @ 55 Park Place
- **Office Hours:** Available upon making an appointment (at least 1 day ahead)
- **Zoom Link:** <https://zoom.us/j/2543708366>
- **Email:** [hxiao@gsu.edu](mailto:hxiao@gsu.edu)

- **Course Material Available at:** iCollege - <http://icollage.gsu.edu>  
Students are recommended to enable notifications for this course to receive announcements, updates, etc.
- **Prerequisite:** No prerequisite, but some level of exposure to statistics, basic calculus and linear algebra may help
- **Proframming:** Python  
Please note that this is not a programming course and students are required to develop the required programming skills on their own.

# Text Books

- Richard Szeliski. [Computer Vision: Algorithms and Applications](#), Springer, 2011.
- Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. [Deep learning](#), MIT press, 2016.
- Stephen Boyd and Lieven Vandenberghe, [Convex Optimization](#), Cambridge University press, 2004
- Trevor Hastie, Robert Tibshirani, and Jerome Friedman, [The Elements of Statistical Learning: Data Mining, Inference, and Prediction](#), Springer, 2009
- James, G., Witten, D., Hastie, T., and Tibshirani, R. [An introduction to statistical learning](#), volume 112. Springer. 2013

## Additional Reading:

- Will be posted on *iCollege*

# Grading

There would be exams but instead multiple homework and a final project will be evaluated. Details are listed as below:

Course Work	Percentages
Attendance and Class Participation	10%
Homework Assignments	$7 \times 10\%$
Final Presentation	20%

The final letter grade conversion is based on the following table:

A+	A	A-	B+	B	B-	C+	C	C-	D	F
$\geq 97$	$\geq 90$	$\geq 87$	$\geq 83$	$\geq 80$	$\geq 77$	$\geq 73$	$\geq 70$	$\geq 67$	$\geq 60$	<60

# Teaching Strategy

- An interactive manner
  - Students are encouraged to attend lecture (Face-to-face or Remotely) and participant class discussions
  - Students are invited to present their ideas in solving problems when presenting examples or during their own research
- For each lecture, learn from the slides for course material
- For some lectures, additional reading or programming may apply

# Homeworks

- Homework will be assigned on a biweekly basis (approximately)
  - You will have one week to work on each homework
  - Homework will be posted around a Tuesday and will be collected on the Tuesday of the week after
- Late homework **is not accepted** and will receive zero credit.
- Make sure to submit all homework on the *iCollege*
- **IMPORTANT**
  - Each student must turn in their own solutions
  - Students copying from their classmates or from any other resources will receive a zero credit
  - If you solve a question together with another colleague, each need to write up your own solution and need to indicate the name of person who you discussed the problem with in your homework

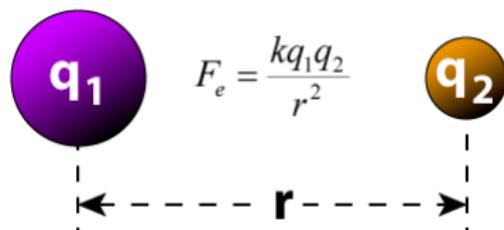
# Final Project

- Instructions and guidelines about the final project will be given around the middle of the semester, once students are familiar with some fundamental tools in Analytics & ML

# Introduction to Machine Learning

# Still Seeking for Methodologies for Modeling

- How scientists discover derivations hundreds of years ago?
  - Experiments
  - For instance, deriving an equation like  $F = 8.99 \times 10^9 \frac{q_1 q_2}{r^2}$  could involve fixing all the parameters except one and evaluating the response behavior
- Seen so many simple physics equations
- Since then, we have been curious about
  - how systems work
  - to predict their behavior before construction them
- In this example, we call  $q_1$ ,  $q_2$  and  $r$  variables and  $F$  a response



# Still Seeking for Methodologies for Modeling

- What has changed?
  - More accurate measure sensors for pretty much all physical phenomena
  - Better ability to collect More data
  - More power computational resources
  - Feasible to solutions of super-complex models
- **Then Why not model large and complex systems??!!**

## Still Seeking for Methodologies for Modeling

- Models that can predict the weather, the stock market value, political interactions, etc.



- Nowadays, being access to more data → More power to predict

# More Examples: Spam Detection

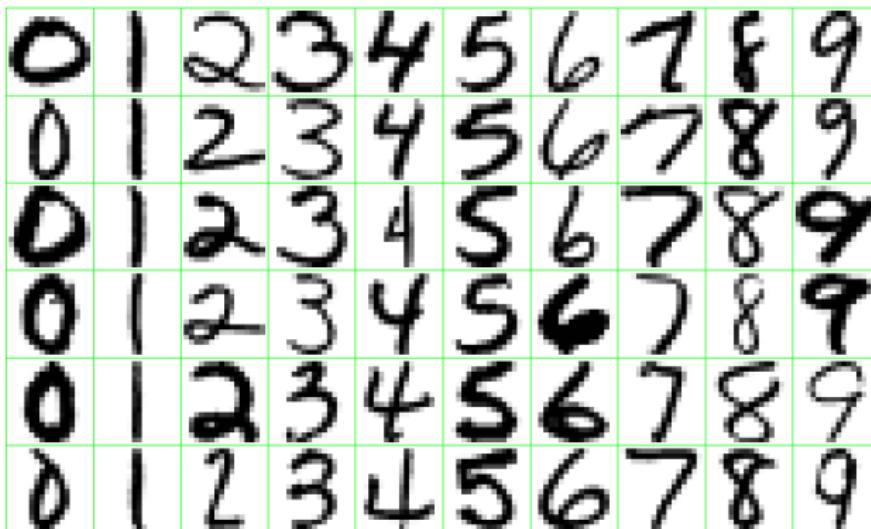
- Data
  - 4,601 emails sent to an individuals
  - Each is labeled as spam or email
- Goal
  - Build a customized spam filter
- Features
  - Relative frequencies of 57 of the most commonly occurring words and punctuation marks in these email messages

	george	you	your	hp	free	hpl	!	our	re	edu	remove
spam	0.00	2.26	1.38	0.02	0.52	0.01	0.51	0.51	0.13	0.01	0.28
email	1.27	1.27	0.44	0.90	0.07	0.43	0.11	0.18	0.42	0.29	0.01

- Average percentage of words or characters in an email message equal to the indicated word or character. We have chosen the words and characters showing the largest difference between **spam** and **email**

## More Examples: Handwritten Digit Recognition

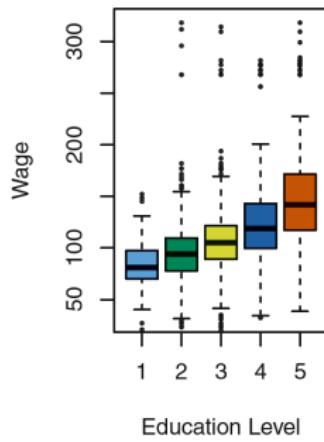
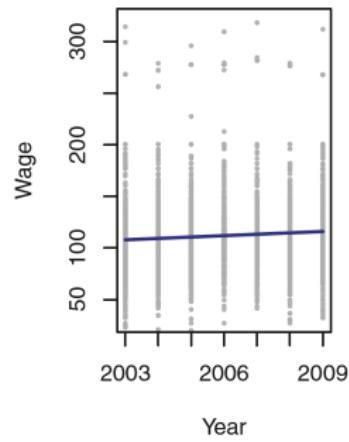
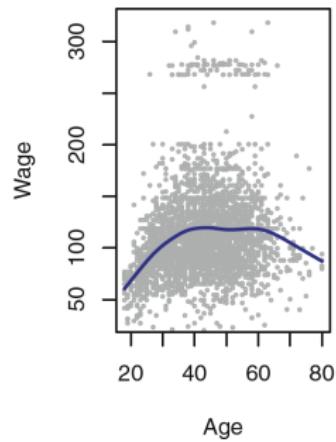
Identify handwritten digit based on the matrix of pixel intensities



The figures from [Hastie et al., 2009]

# More Examples: Salary Prediction

- Discovery the relationship between salary and demographic variables in population survey data

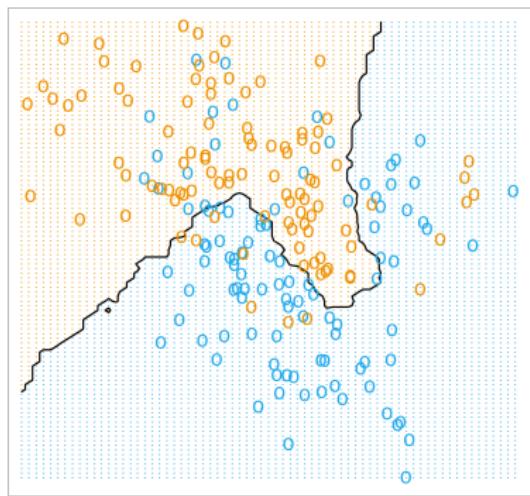
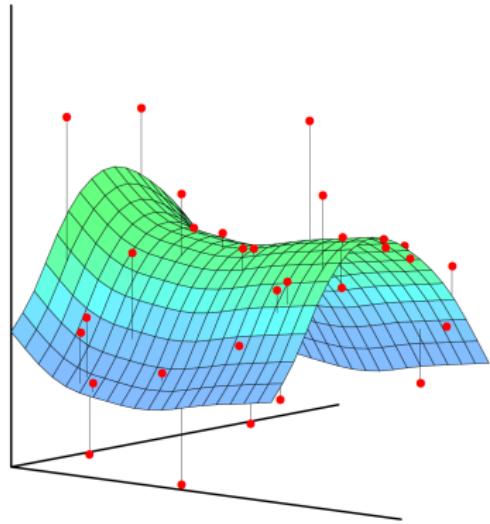


# Supervised Learning

- Recall the Coulomb's Law,  $F = 8.988 \cdot 10^9 \frac{q_1 q_2}{r^2}$  or more generally  
 $y = f(\mathbf{x})$  where  $\mathbf{x} = (x_1, \dots, x_p)^\top$ 
  - In the Coulomb's Law,  $y = F$  and  $\mathbf{x} = (q_1, q_2, r)^\top$
  - $y$ , outcome measurement or dependent variable, response, target
  - $\mathbf{x}$ , vector of  $p$  predictor measurements, or inputs, regressors, covariates, features, independent variables
- In the **regression** problem,  $y$  is quantitative (e.g., price, blood pressure, etc)
- In the **classification** problem,  $y$  takes values in a finite, unordered set (e.g., survived/died, digit 0 ~ 9, etc.)
- We have training data  $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)$ . These are observations (or examples, instances) of these measurements

**Note:** **bold** (e.g.,  $\mathbf{x}_1$ ) faces correspond to vectors

# Examples of Regression and Classification



# Supervised Learning

- On the basis of the training data we would like to:
  - Accurately predict unseen test samples
  - Understand which features affect the outcome, and HOW
  - Assess the quality of the predictions and inferences
- Training and Testing data:
  - Normally from the available examples  $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)$ , we randomly choose  $n_1$  for training
  - The remaining  $n - n_1$  for the later evaluation of the model (Testing)
  - $n_1 : (n - n_1)$ , usually takes 70%:30% or 80%:20%

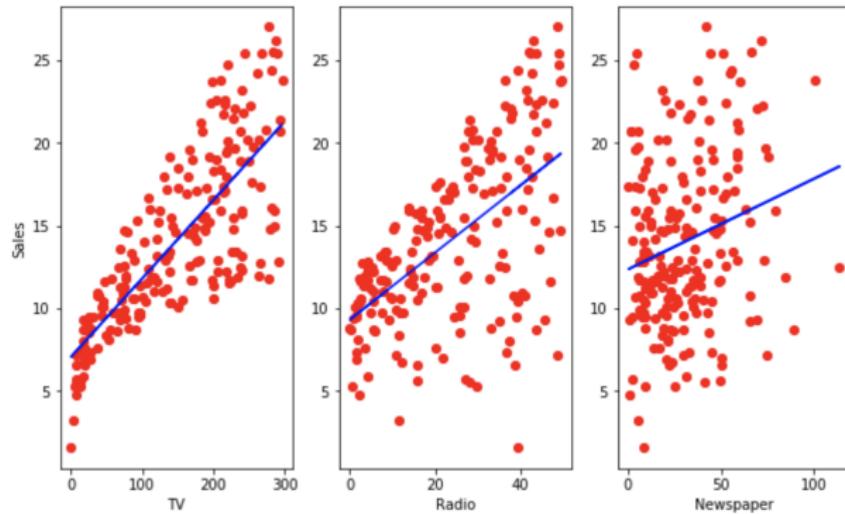
# Unsupervised Learning

- We do not have outcome variable  $y$ 
  - Only a set of predictors (features) measured on a set of samples
- no clear objective like supervised learning
  - find groups of samples that behave similarly
  - find features that behave similarly
  - find linear combinations of features with the most variation
- Usually hard for model evaluation
- Different from supervised learning, but can be used as a pre-processing step
- Examples
  - Topic Modeling
  - Customer Segments

# More on Basics of Statistical Learning

# An Example

Below shown are Sales w.r.t. the advertising budget on TV, Radio, and Newspaper [James et al., 2013].



(See the code)

- Can we predict Sales using these three features?
- Build  $\text{sales} = f(\text{TV}, \text{Radio}, \text{Newspaper})$

# Why do we Need Predictive Models

- With a good model we can make predictions of  $y$  at new samples  $x$
- Identify important features are important in explaining  $y$
- For instance
  - in the sales example, TV, Radio and Newspaper can be important
  - but “which hotel is close to the company” might not be important
- Depending on the complexity of the fit, we may be able to understand how each component  $x_i$  of  $x$  affects  $y$

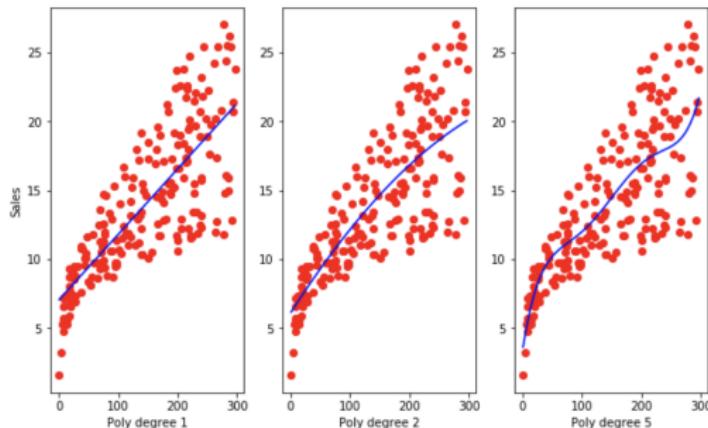
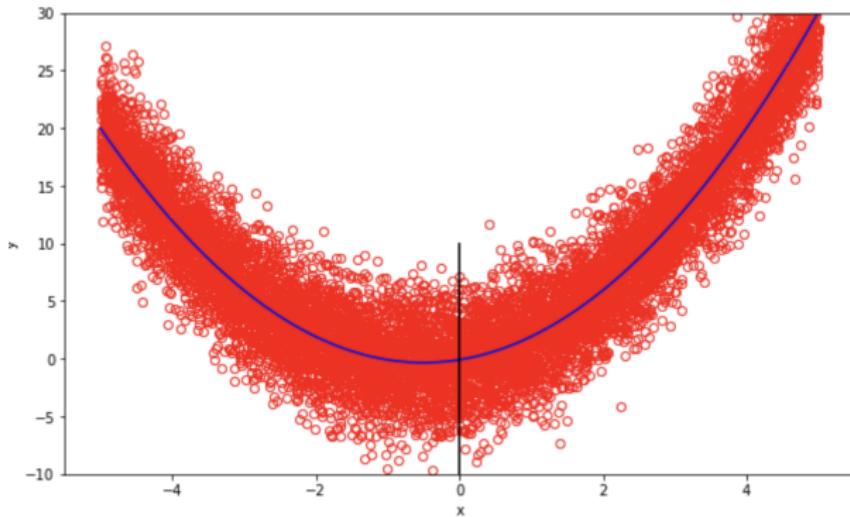


Figure: TV (See the code)

# Best Predictive Model if We Had Enough Data

- Suppose we had many data samples  $(x, y)$
- What would have been a good estimate of  $y$  given  $x = 0$



(See the code)

- Mathematically, what we like is  $\mathbb{E}(y|x = 0)$ 
  - The idea value of  $y$  when  $x = 0$  is the average of all responses at  $x = 0$

# Best Predictive Model if We Had Enough Data

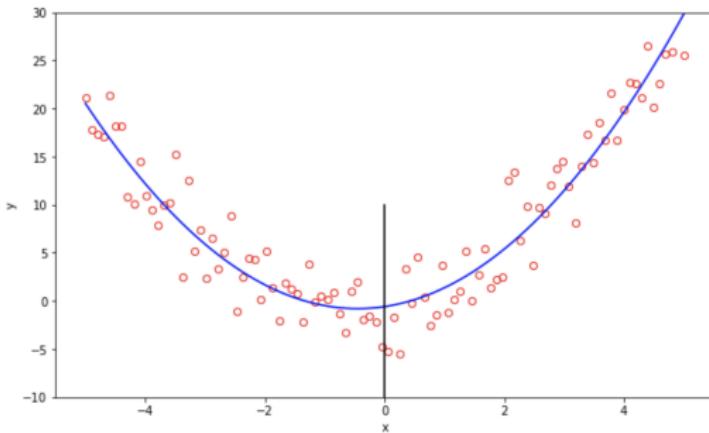
- In an ideal cases of enough data, the reasonable value of  $y$  corresponding to  $X = \mathbf{x}$  should be the average of all responses at  $\mathbf{X} = \mathbf{x}$
- Mathematically,  $f(x) = \mathbb{E}(y|\mathbf{X} = \mathbf{x})$  is the function that minimizes  $\mathbb{E}((y - g(\mathbf{X}))^2|\mathbf{X} = \mathbf{x})$  over all functions  $g$  at all points  $\mathbf{X} = \mathbf{x}$
- The ideal  $f(x) = \mathbb{E}(y|\mathbf{X} = \mathbf{x})$  is called the **regression function**

## Question

Show that  $f(x) = \mathbb{E}(Y|X = x)$  is the function that minimizes  $\mathbb{E}(Y - g(X))^2|X = x)$  over all functions  $g$ .

# What Happens in Practice

- No enough data to recover  $f$
- Even with enough data, no practical to fit using all  $x = 0$



(See the code)

- $k$  Nearest neighbor: consider neighborhoods around  $x = 0$

- Only good for small  $p$  ( $< 4$ ) and large  $N$
- Why?

Consider  $\mathbf{x} = (0.9, \dots, 0.9) \in \mathbb{R}^{100}$  and  $\mathbf{x}' = (1, \dots, 1) \in \mathbb{R}^{100}$

# Fundamental Limits

- The best we can do is the regression function

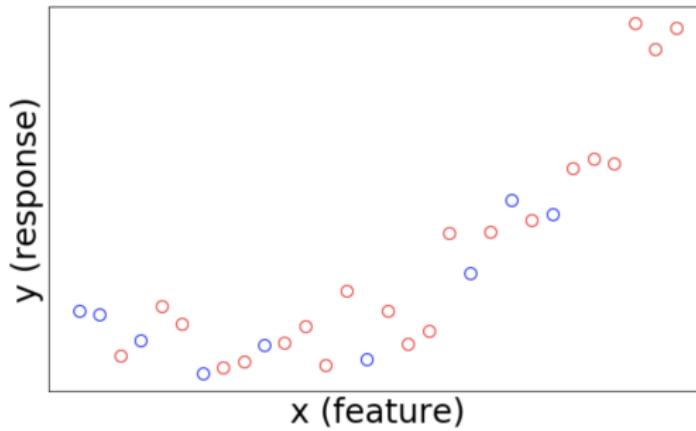
$$Y = f(\mathbf{x}) + \epsilon$$

- $f(\mathbf{x})$  can be considered as the actual physical model that generates data at  $\mathbf{X} = \mathbf{x}$  and  $\epsilon$  as the noise and uncertainty
- $\epsilon$  is a random noise not under our control
- In practice, no way to exactly discover  $f(\mathbf{x})$  for  $\mathbf{X} = \mathbf{x}$ 
  - Find another function  $\hat{f}(\mathbf{x})$  to approximate  $f(\mathbf{x})$

$$\mathbb{E}((Y - \hat{f}(\mathbf{x}))^2 | \mathbf{X} = \mathbf{x}) = \underbrace{(f(\mathbf{x}) - \hat{f}(\mathbf{x}))^2}_{\text{Learn a better } \hat{f}} + \underbrace{\text{Var}(\epsilon)}_{\text{Nothing we can do}}$$

# Basics of Model Fitting

- How to fit a model based on the available data



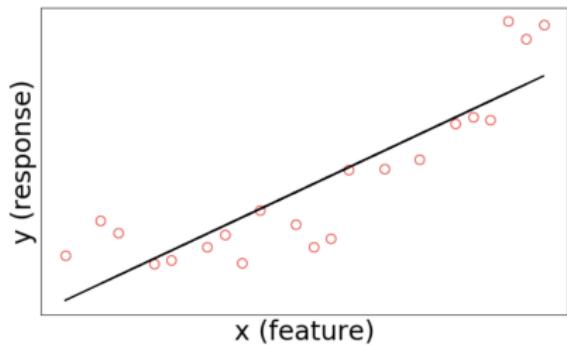
(See the code)

- We split the data into a **training set** and a **testing set**
  - the training set for training a model
  - the testing set for evaluating the model

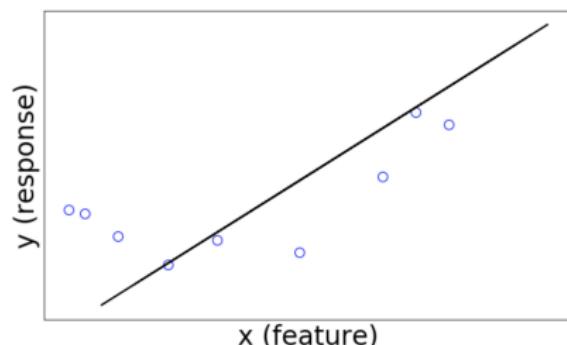
# Basics of Model Fitting

Fit a linear model:  $y = \alpha_1 \cdot x + \alpha_0$

- Model training to find  $\hat{\alpha}_1$  and  $\hat{\alpha}_0$



- Testing the linear model:  
 $\hat{y} = \hat{f}(x) = \hat{\alpha}_1 \cdot x + \hat{\alpha}_0$

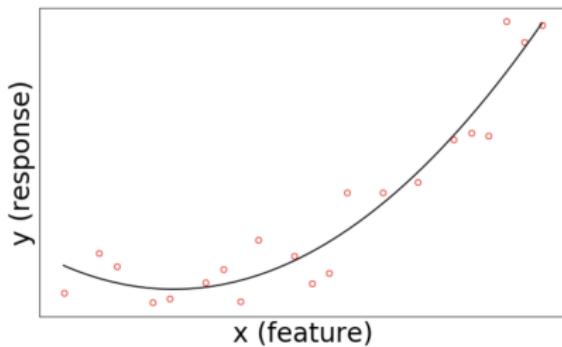


(See the code)

# Basics of Model Fitting

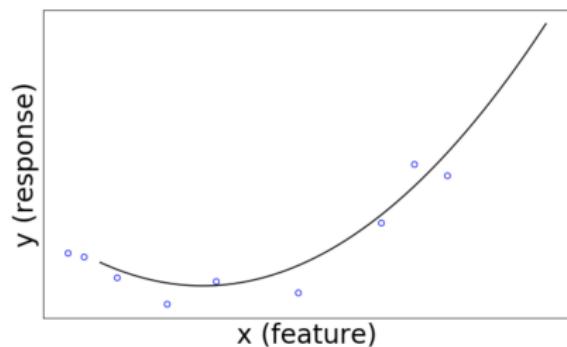
Polynomial model with degree 2:  $y = \alpha_2 \cdot x^2 + \alpha_1 \cdot x + \alpha_0$

- Model training for  $\hat{\alpha}_2$ ,  $\hat{\alpha}_1$  and  $\hat{\alpha}_0$



- Model testing:

$$\hat{y} = \hat{f}(x) = \hat{\alpha}_2 \cdot x^2 + \hat{\alpha}_1 \cdot x + \hat{\alpha}_0$$

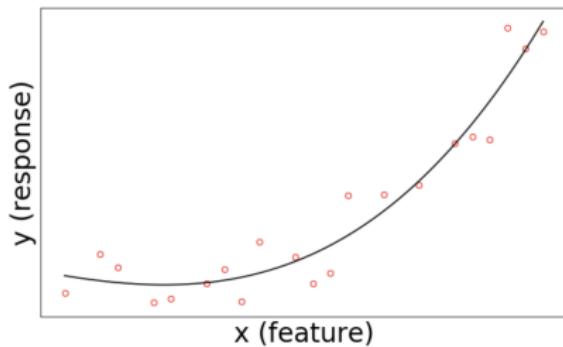


(See the code)

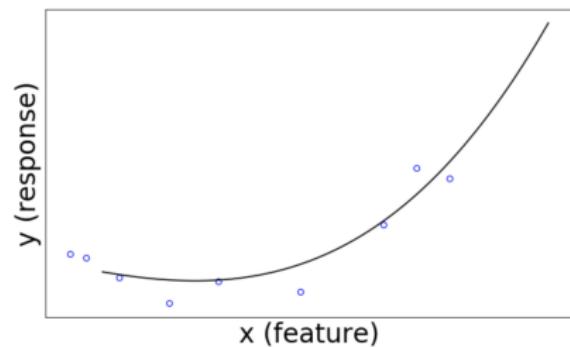
# Basics of Model Fitting

Polynomial model with degree 3:  $y = \alpha_3 \cdot x^3 + \alpha_2 \cdot x^2 + \alpha_1 \cdot x + \alpha_0$

- Model training for  $\hat{\alpha}_3$ ,  $\hat{\alpha}_2$ ,  $\hat{\alpha}_1$  and  $\hat{\alpha}_0$



- Model testing:  $\hat{y} = \hat{\alpha}_3 \cdot x^3 + \hat{\alpha}_2 \cdot x^2 + \hat{\alpha}_1 \cdot x + \hat{\alpha}_0$

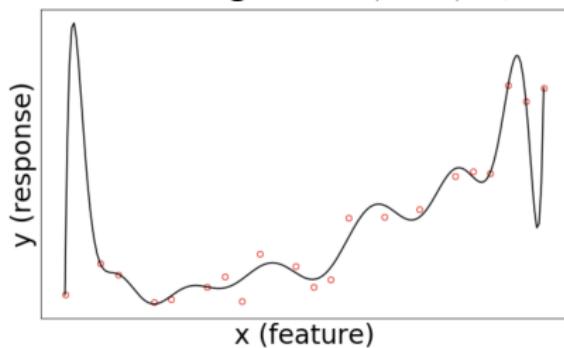


(See the code)

# Basics of Model Fitting

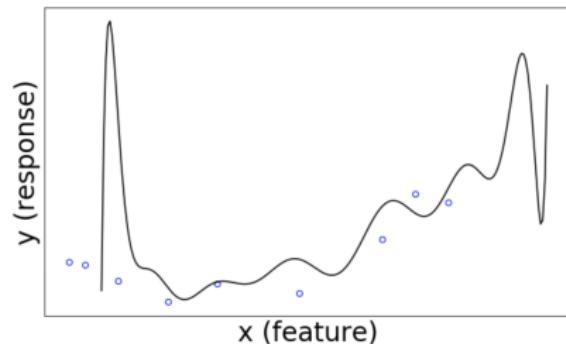
Polynomial model with degree 15:  $y = \sum_{p=0}^P \alpha_p \cdot x^p$

- Model training for  $\hat{\alpha}_P, \dots, \hat{\alpha}_0$



- Model testing:

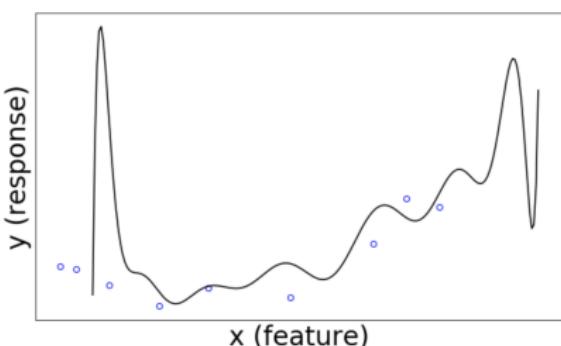
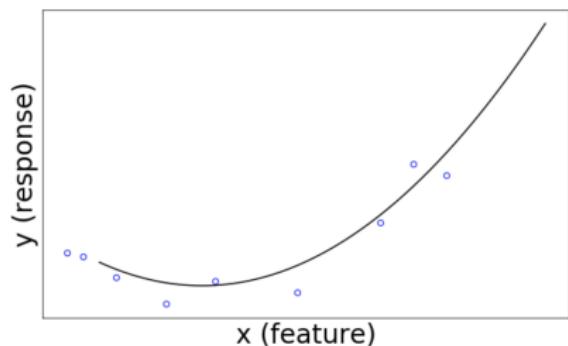
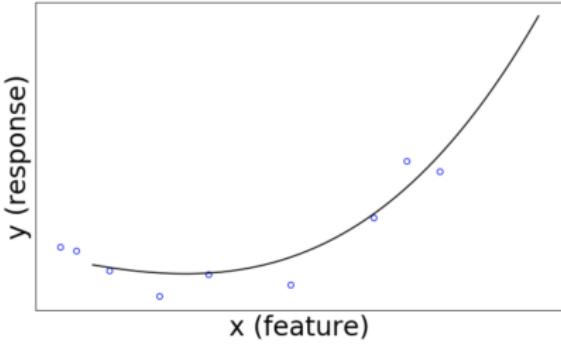
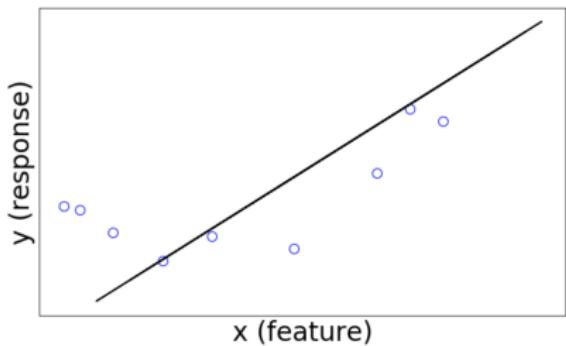
$$\hat{y} = \hat{f}(x) = \sum_{p=0}^P \hat{\alpha}_p \cdot x^p$$



(See the code)

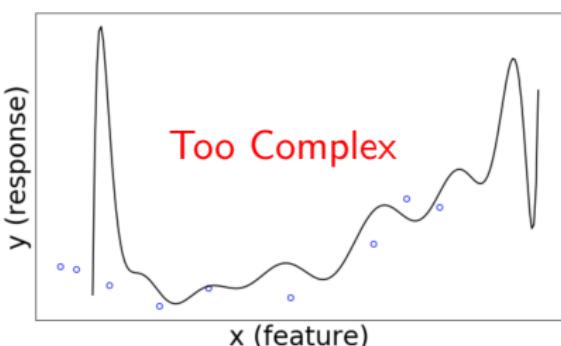
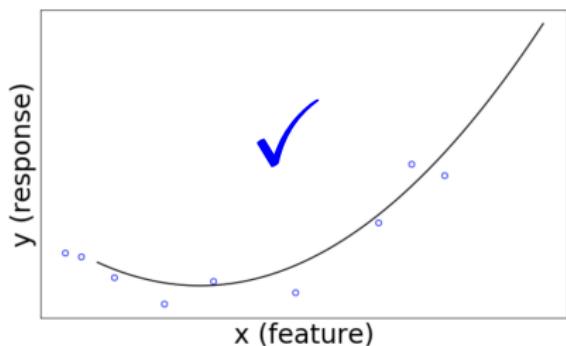
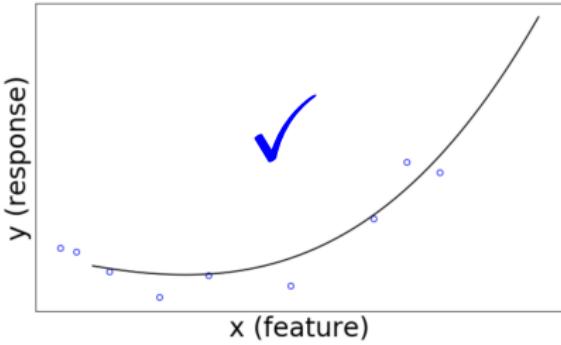
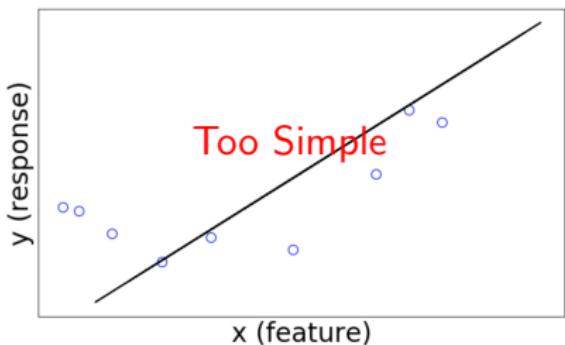
# Basics of Model Fitting

- Which model to pick? Overfitting or Underfitting



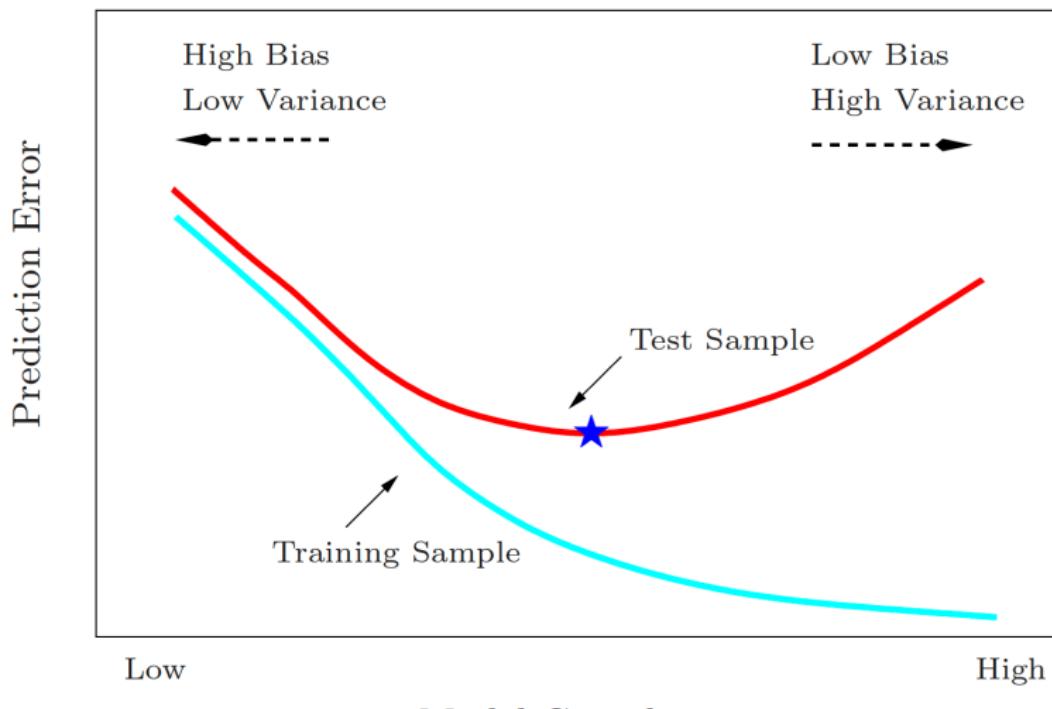
# Basics of Model Fitting

- Which model to pick? Overfitting or Underfitting



# Typical Curve

- The balance between model flexibility and test error typically looks as below



# Assessing Model Accuracy

- Once we fit our model  $\hat{f}$  we want to evaluate its performance
- The evaluation cannot be done using training data
- Instead, using testing data which has not been used in model training  
(Out of Sample prediction)
- Measurements:
  - Mean squared error (MSE) =  $Mean_{i \in Test} (Y_i - \hat{f}(x_i))^2$   
Sometimes, use its root, i.e., RMSE
  - Mean absolute error (MAE) =  $Mean_{i \in Test} |Y_i - \hat{f}(x_i)|$

# Bias vs Variance (Theory)

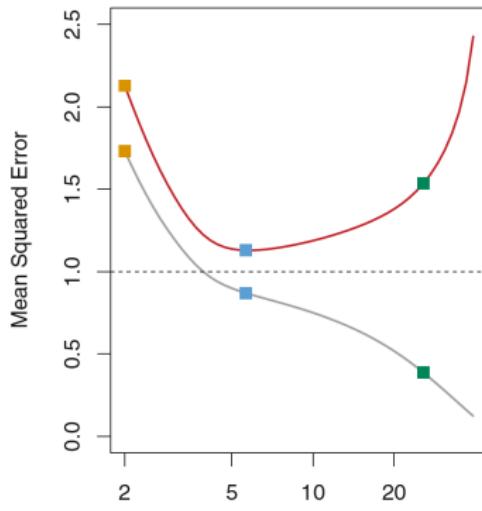
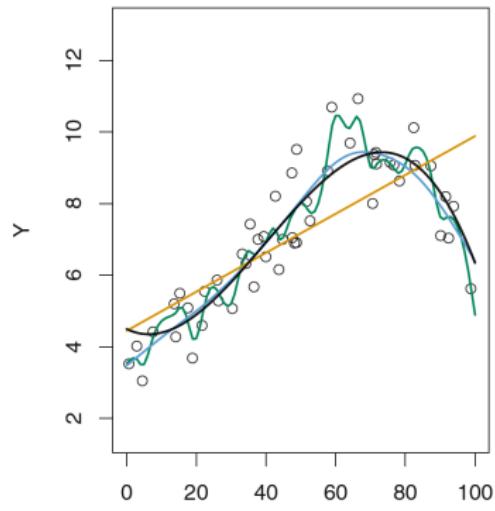
- Suppose we fit a model  $\hat{f}$  and we have a large test data set for evaluation
- Assuming that  $T$  is the training data set based on which  $\hat{f}$  is learned;  $(\mathbf{x}_t, Y_t)$  is a fixed point in the test dataset. We have that

$$\mathbb{E}_{total} \left( Y_t - \hat{f}(\mathbf{x}_t) \right)^2 = Var(\hat{f}(\mathbf{x}_t)) + \left( Bias(\hat{f}(\mathbf{x})) \right)^2 + Var(\epsilon) \quad (1)$$

$$Bias(\hat{f}(\mathbf{x})) = \mathbb{E}_{training}(\hat{f}(\mathbf{x}_t) - f(\mathbf{x}_t)) \quad (2)$$

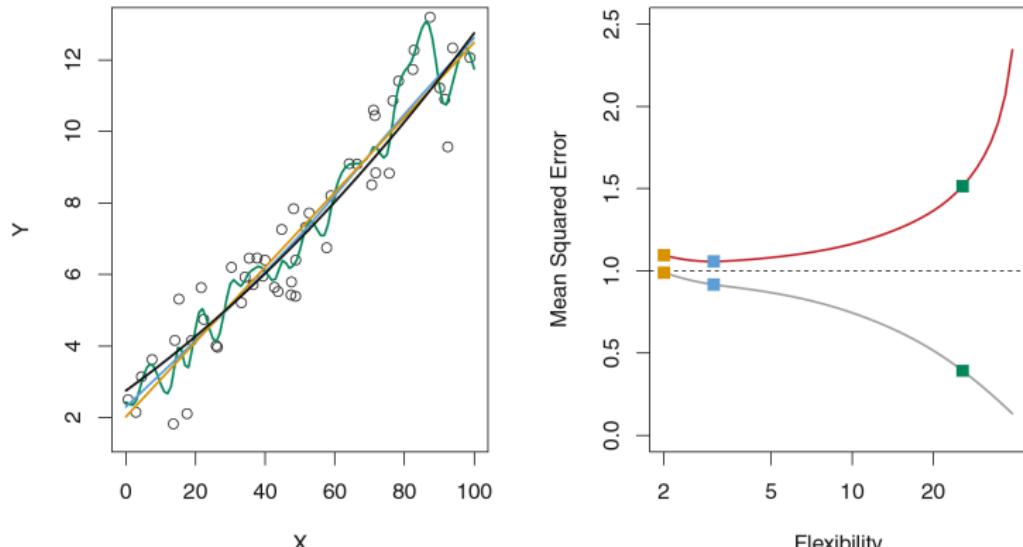
- Normally, as we make  $\hat{f}$  more flexible by making it complex via more sophisticated formulations and features,
  - the **model variance** term  $Var(\hat{f}(\mathbf{x}_t))$  increases
  - the bias  $Bias(\hat{f}(\mathbf{x}))$  decreases
- Reducing the test error becomes a trade-off between the bias and the variance

# Bias vs Variance



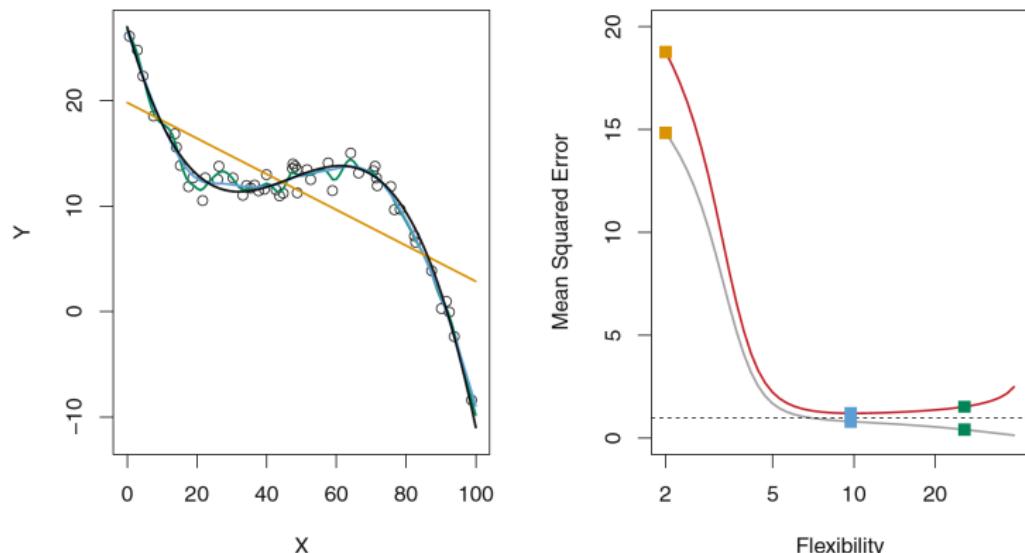
X  
Left: Data simulated from  $f$ , shown in black. Three estimates of  $f$  are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves). Right: Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.

# Bias vs Variance



X  
Left: Data simulated from  $f$ , shown in black. Three estimates of  $f$  are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves). Right: Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.

# Bias vs Variance



Left: Data simulated from  $f$ , shown in black. Three estimates of  $f$  are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves). Right: Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.

# Trade Offs

We just discussed the bias vs variance trade off. In general we deal with various trade offs

- Prediction accuracy vs interpretability
  - Linear models are easy to interpret
  - thin-plate splines are not
- Good fit vs over-fit or under-fit
  - When the fit is just right?
- Parsimony vs black-box
  - Prefer a simpler model involving fewer variables over a black-box predictor involving more variables

# Classification

- How over-fitting looks for classification problems
- K-Nearest Neighbors Approach

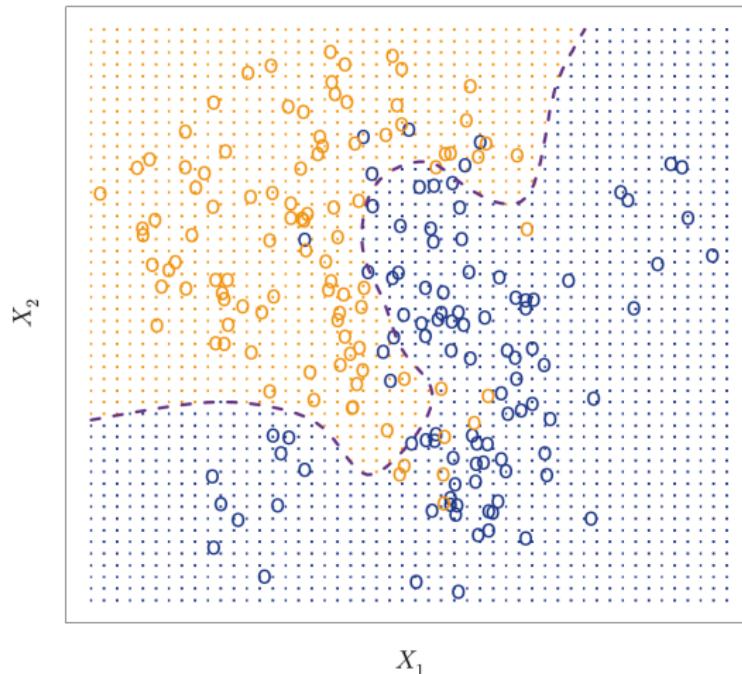
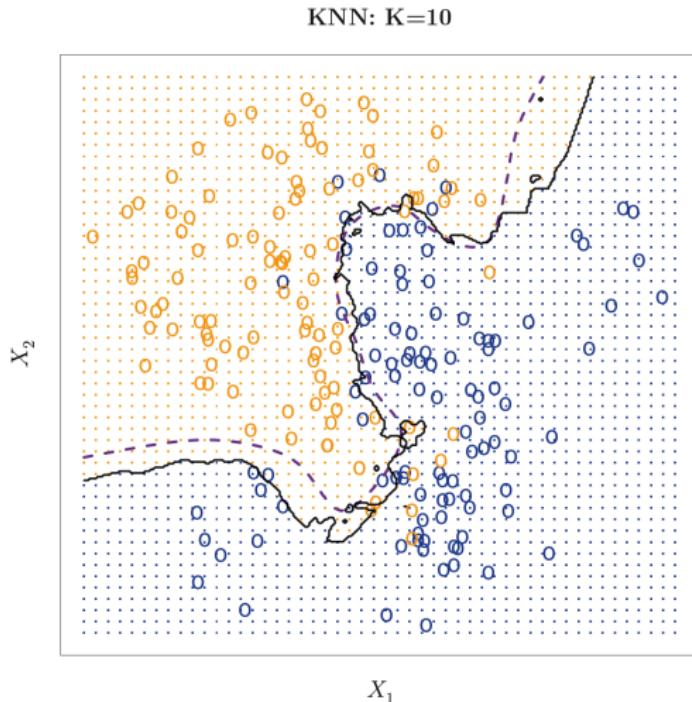


Figure: Ideal classifier

# Classification

- How over-fitting looks for classification problems
- K-Nearest Neighbors Approach (the dense grid is somehow our test)



# Classification

- How over-fitting looks for classification problems
- K-Nearest Neighbors Approach (the dense grid is somehow our test)

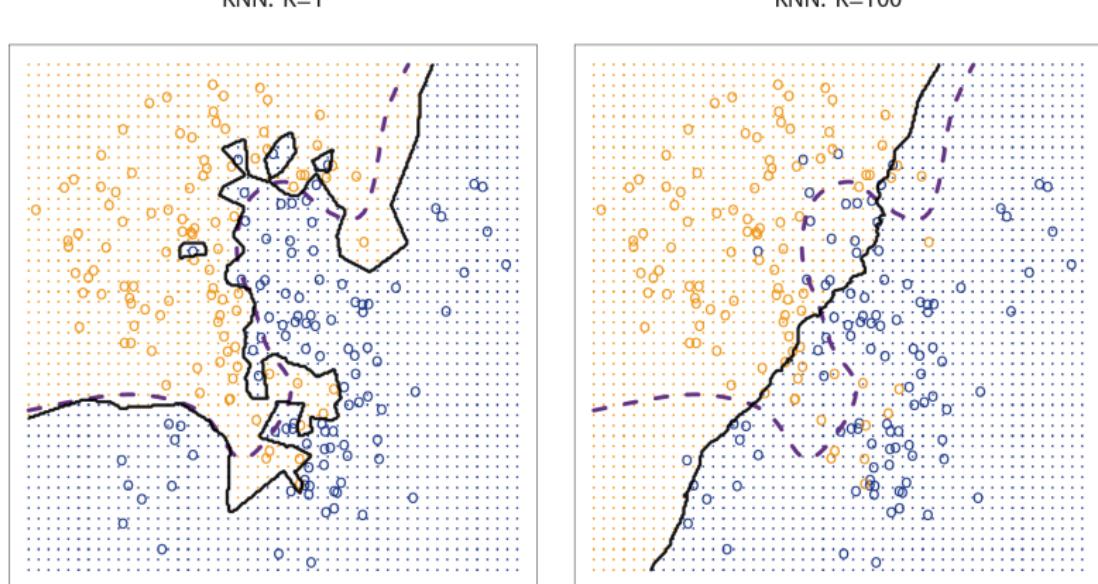


Figure: KNN classifier

# Classification

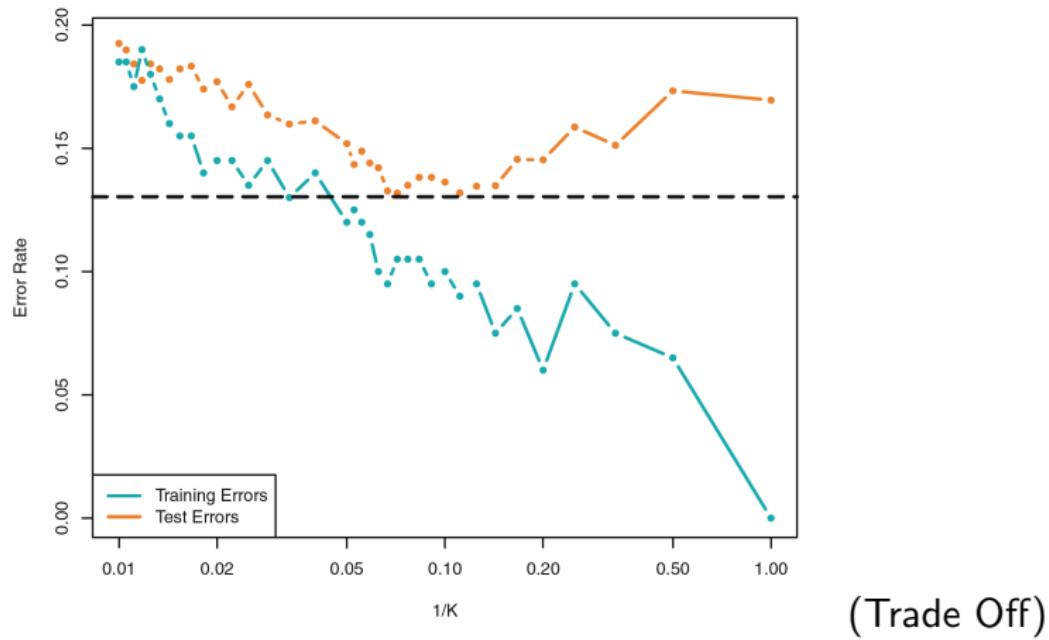
- K-Nearest neighbors approach
- Note that in KNN flexibility is inversely proportional to  $K$



(Just like trying to walk among many people surrounding you)

# Classification

- The trade off curve
- K-Nearest neighbors approach



# Lecture Review

# Review

- Discussed the course information
- Related machine learning to model understanding in science and engineering
- Presented some machine learning examples in practices
- Talked about supervised and unsupervised learning, regression and classification
- Discussed some basic of statistical learning
  - regression function
  - test vs training data
  - overfitting and model trade off

Don't worry if some of the topics covered look vague! We will revisit all this material in depth later in this course

# Reference

 Hastie, T., Tibshirani, R., and Friedman, J. (2009).

*The elements of statistical learning: data mining, inference, and prediction.*

Springer Science & Business Media.

 James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013).

*An introduction to statistical learning*, volume 112.

Springer.

# The End