CS 260: Foundations of Data Science

Prof. Thao Nguyen

Fall 2024

HAVERFORD
COLLEGE

Materials by Sara Mathieson

Admin

- Sit somewhere new!
- Sign up for lecture note-taking
- Lab 2 posted (due next Monday)
- Lab 1 is due tomorrow (Tuesday) at midnight
- My office hours: 10-11:30am on Tuesday (H110)

Outline for today

- Data representation and featurization
- Introduction to modeling
- Why are models useful?
- Linear models

Outline for today

- Data representation and featurization
- Introduction to modeling
- Why are models useful?
- Linear models

Tennis Data

Day	Outlook	Temperature	Humidity	Wind	PlayTennis (y)
$\boldsymbol{x}_1$	Sunny	Hot	High	Weak	No
$\boldsymbol{x}_2$	Sunny	Hot	High	Strong	No
$\boldsymbol{x}_3$	Overcast	Hot	High	Weak	Yes
$\boldsymbol{x}_4$	Rain	Mild	High	Weak	Yes
$\boldsymbol{x}_5$	Rain	Cool	Normal	Weak	Yes
$\boldsymbol{x}_6$	Rain	Cool	Normal	Strong	No
$\boldsymbol{x}_7$	Overcast	Cool	Normal	Strong	Yes
$\boldsymbol{x}_8$	Sunny	Mild	High	Weak	No
$\boldsymbol{x}_9$	Sunny	Cool	Normal	Weak	Yes
$\boldsymbol{x}_{10}$	Rain	Mild	Normal	Weak	Yes

Data from Machine Learning by Tom Mitchell (Table 3.2)

- Input or **features**: outlook, temp, humidity, wind
- Output or "**label**": play tennis (yes or no)

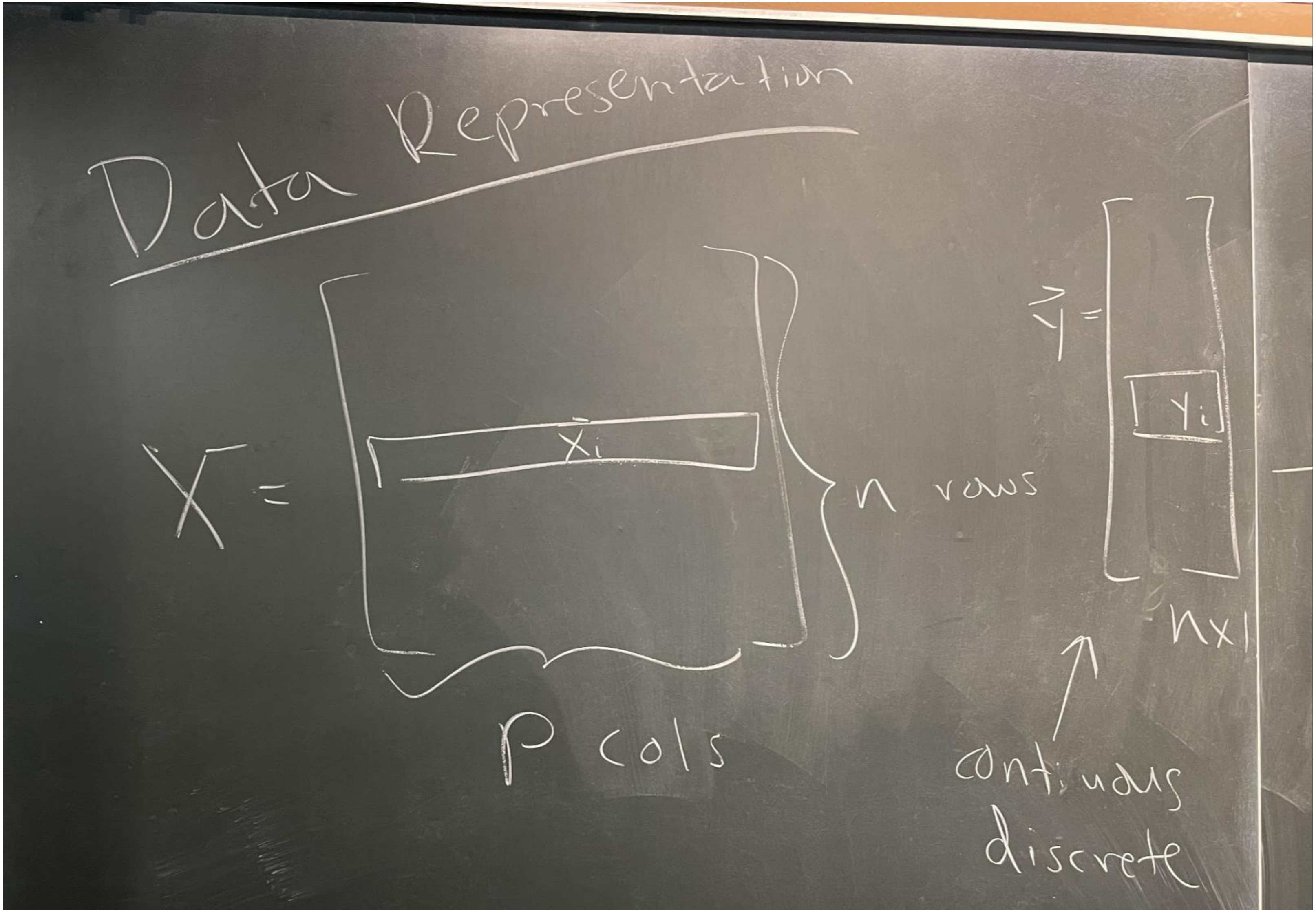
Sea Ice data (Lab 2)

Year	**Sea Ice Extent***
1996	7.88
1997	6.74
1998	6.56
1999	6.24
2000	6.32
2001	6.75
2002	5.96
2003	6.15
2004	6.05
2005	5.57
2006	5.92
2007	4.3
2008	4.63

- Input or **feature**: year
- Output or "**label**": sea ice extent

*Arctic sea ice extent (1,000,000 km^2)

Data Representation Notation



Feature Terminology

- *Features:* feature names
  - shape
  - sea ice extent

- *Feature values:* what values are possible
  - {circle, square, triangle}
  - all non-negative values

- *Feature vector:* values for a particular example/data point
  - $\boldsymbol{x} = [x_1, x_2, x_3, \ldots, x_p]$

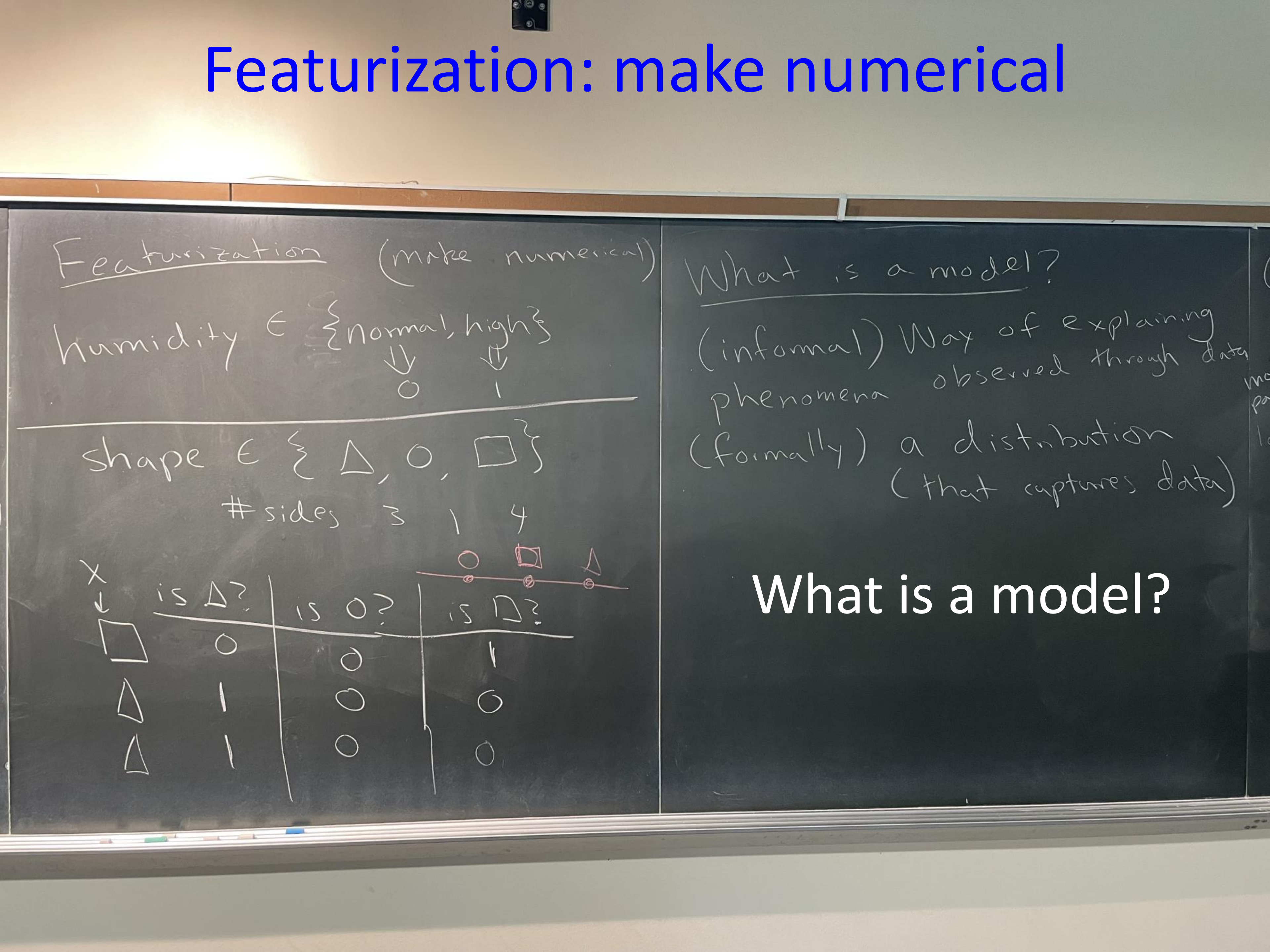
Featurization: make numerical

- Real-valued features get copied directly.
- Binary features become 0 (for false) or 1 (for true).
- Categorical features with V possible values get mapped to V-many binary indicator features.

Duame, Chap 3

Q: what about features that might already be on a spectrum (e.g. sunny, rain, overcast)?

Featurization: make numerical

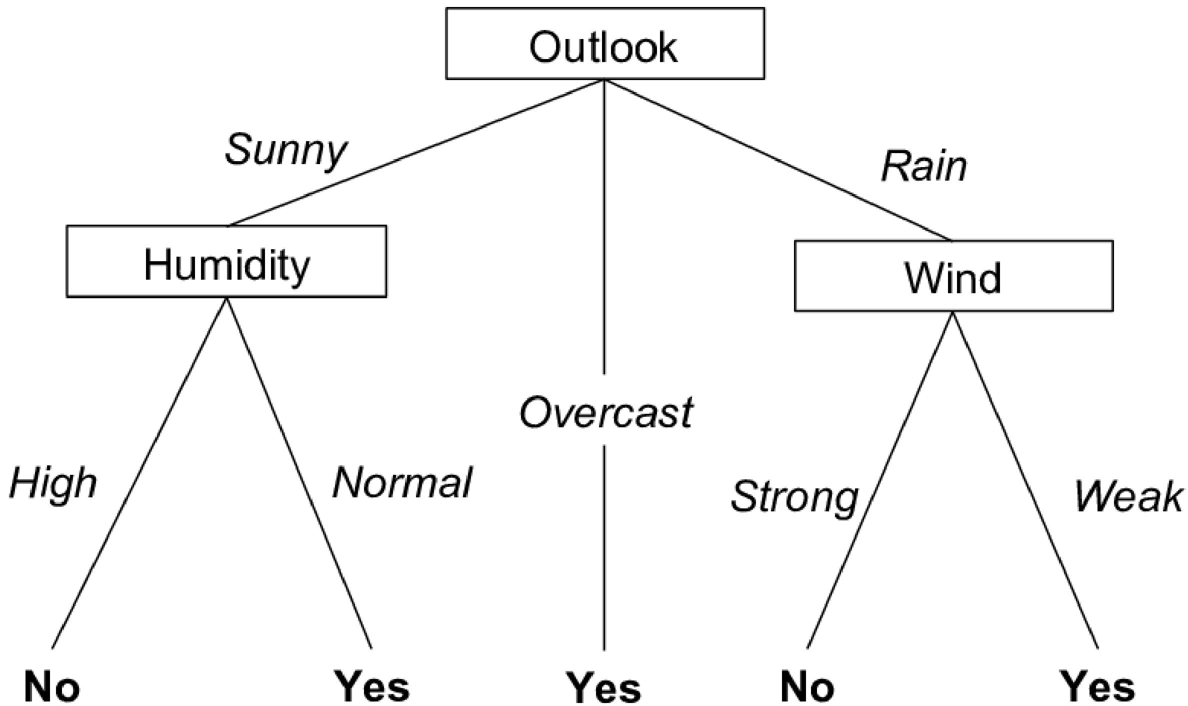


What is a model?

Outline for today

- Data representation and featurization
- Introduction to modeling
- Why are models useful?
- Linear models

Example of a model

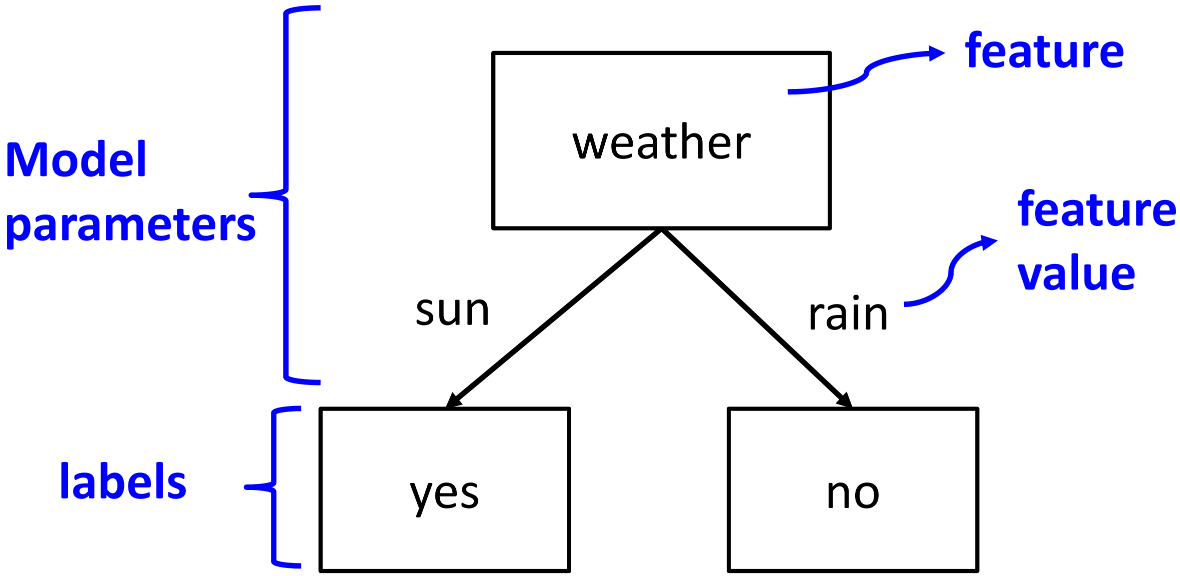


- Each internal node: one feature
- Each branch from node: selects one value of the feature
- Each leaf node: predict y

Based on slides by Jessica Wu and Eric Eaton [originally by Tom Mitchell]

Model Examples

1) Decision Tree



[1, 2] **[2,0]** → **“no” and “yes” label counts**

2/3 correct **2/2 correct**

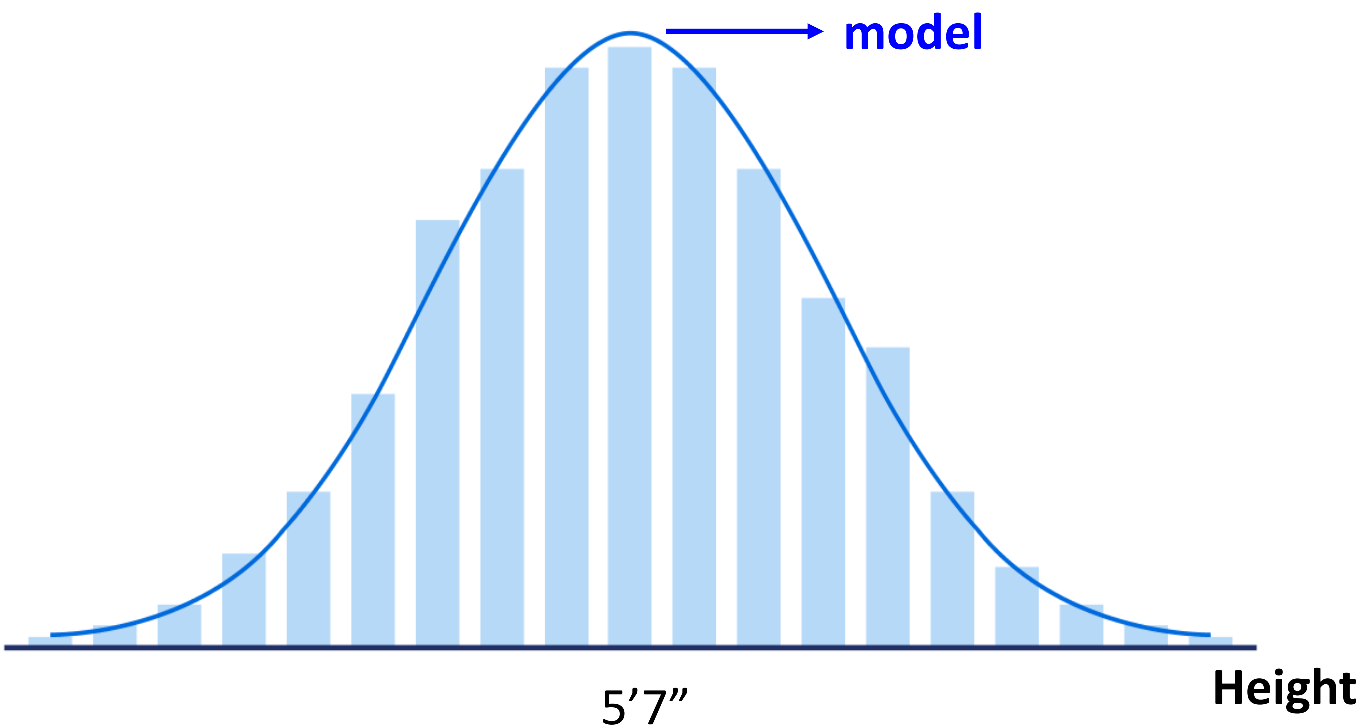
=> 80% accuracy

Data

weather	tennis
sun	yes
rain	no
rain	no
sun	yes
sun	no

Model Examples

2) Normal distribution

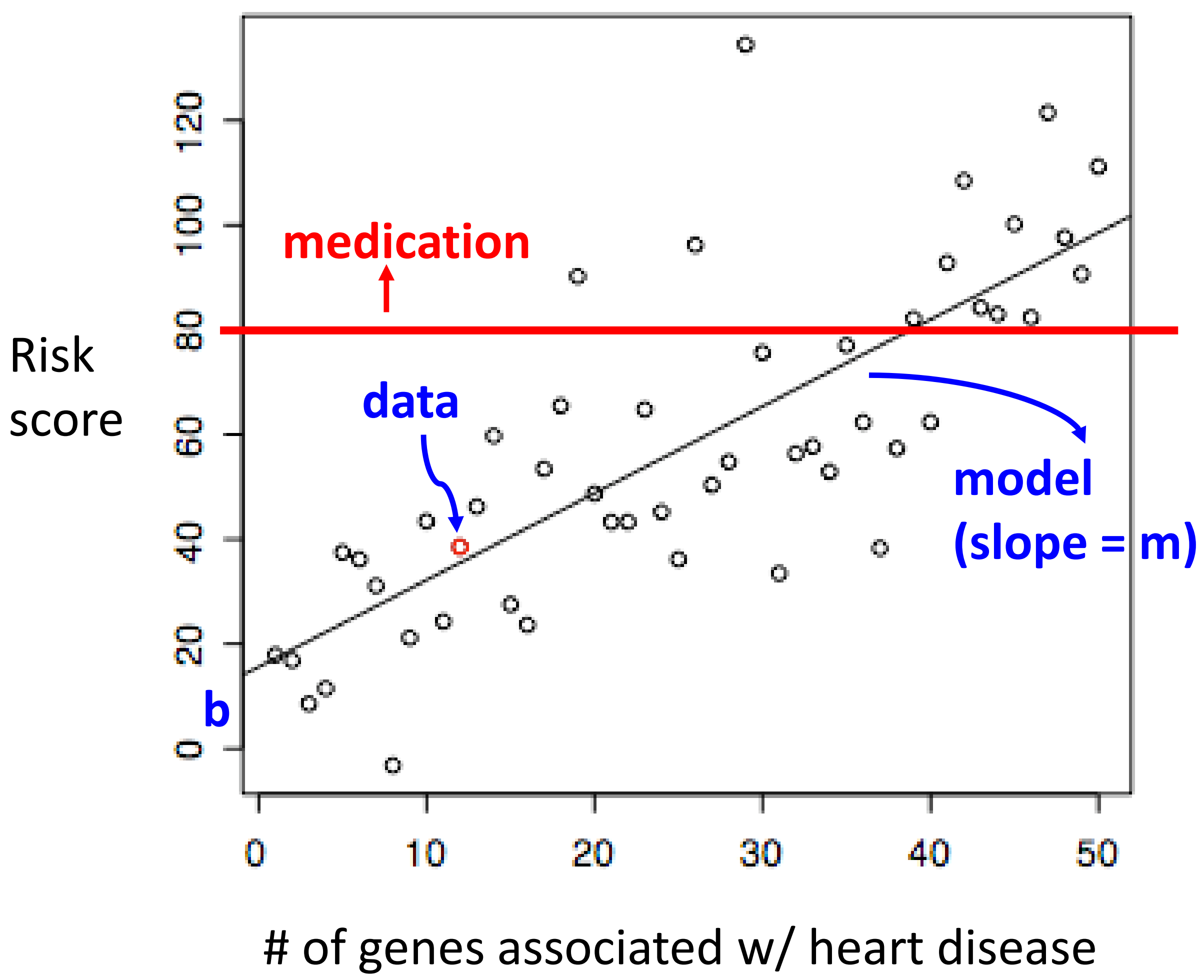


mean: 5’7”
variance: 2”

Model parameters

Model Examples

3) Linear models



y = mx + b

m, b -> model parameters

Handout 3

Handout 3

Q1: *n*=10, *p*=4

Day	Outlook	Temperature	Humidity	Wind	PlayTennis (y)
$\boldsymbol{x}_1$	Sunny	Hot	High	Weak	No
$\boldsymbol{x}_2$	Sunny	Hot	High	Strong	No
$\boldsymbol{x}_3$	Overcast	Hot	High	Weak	Yes
$\boldsymbol{x}_4$	Rain	Mild	High	Weak	Yes
$\boldsymbol{x}_5$	Rain	Cool	Normal	Weak	Yes
$\boldsymbol{x}_6$	Rain	Cool	Normal	Strong	No
$\boldsymbol{x}_7$	Overcast	Cool	Normal	Strong	Yes
$\boldsymbol{x}_8$	Sunny	Mild	High	Weak	No
$\boldsymbol{x}_9$	Sunny	Cool	Normal	Weak	Yes
$\boldsymbol{x}_{10}$	Rain	Mild	Normal	Weak	Yes

Data from Machine Learning by Tom Mitchell (Table 3.2)

Q2

Sunny:	{0,1}	
Overcast:	{0,1}	
Rain:	{0,1}	
Temperature:	{0, 1, 2}	(Cool, Mild, Hot)
Humidity:	{0,1}	(Normal, High)
Wind	{0,1}	(Weak, Strong)

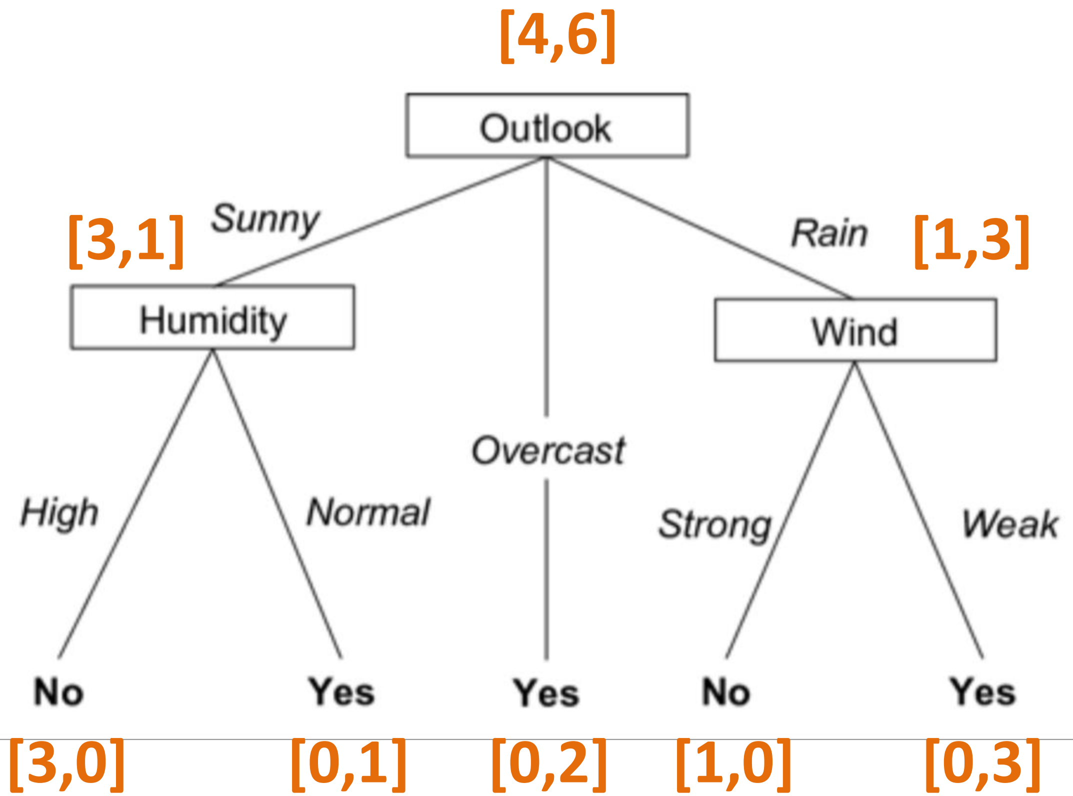
Q3

	Sunny	Overcast	Rain	Temp	Humidity	Wind
$\boldsymbol{x}_1$	1	0	0	2	1	0

Handout 3

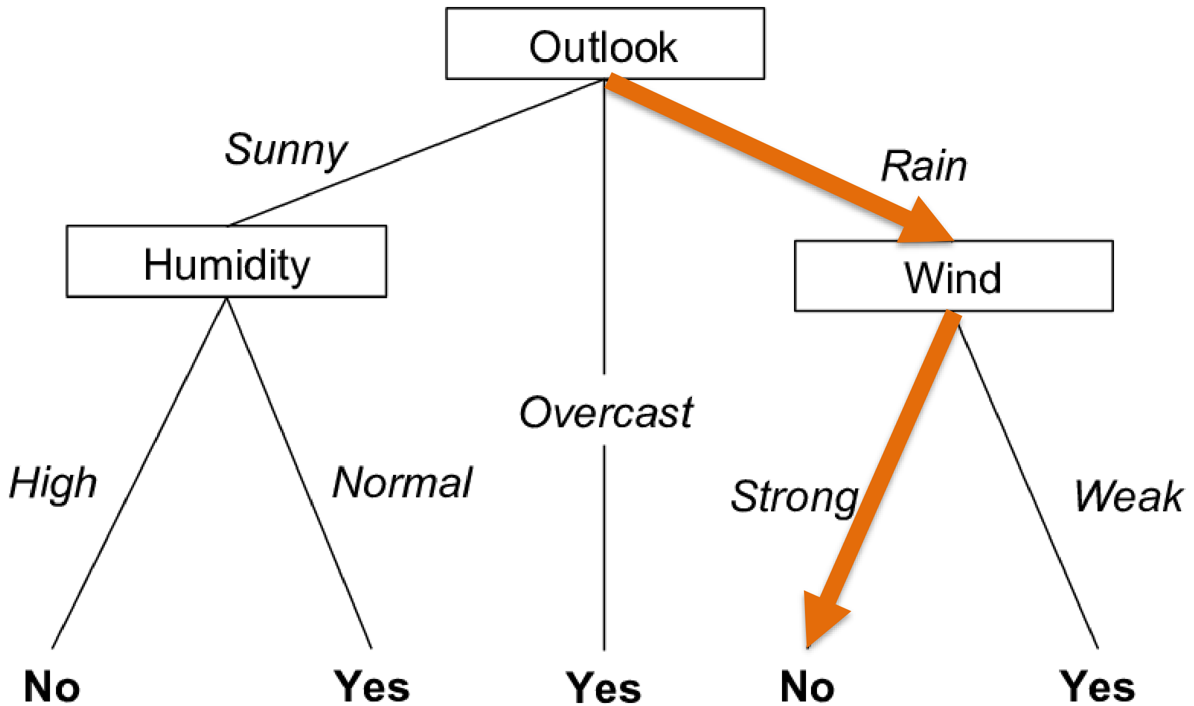
Q4

In the model below, the children of each node divide the data into partitions. Label each node (both internal nodes and leaves) with the counts of "No" and "Yes" labels based on the partition. For example, the counts for the node labeled *Outlook* would be [4,6]. Does this model perfectly classify all examples?



Handout 3

Q5



(test example) ***x*** =

Outlook	Temp	Humidity	Wind
Rain	Hot	High	Strong

y_{pred} = No

Outline for today

- Data representation and featurization
- Introduction to modeling
- Why are models useful?
- Linear models

Why are models useful?

- Understand/explain/interpret the phenomenon
- Predict outcomes for future examples

What are the most important features?

X			Y
Color	**Shape**	**Size**	**Likes toy?**
red	square	big	+
blue	square	big	+
red	circle	small	-
blue	square	small	-
red	circle	big	+

Example by: Ameet Soni

What are the most important features?

X			Y
Color	**Shape**	**Size**	**Likes toy?**
red	square	big	+
blue	square	big	+
red	circle	**big**	-
blue	square	**big**	-
red	circle	big	+

Example by: Ameet Soni

Outline for today

- Data representation and featurization
- Introduction to modeling
- Why are models useful?
- Linear models

Linear Models

* features: $\vec{x}$ (right now just x)
* output: y (response)

Goals:

1. describe linear dependence
2. predict response given new data

model

$$h_{\vec{w}}(x) = w_0 + w_1 x = \hat{y} \quad \text{(prediction)}$$

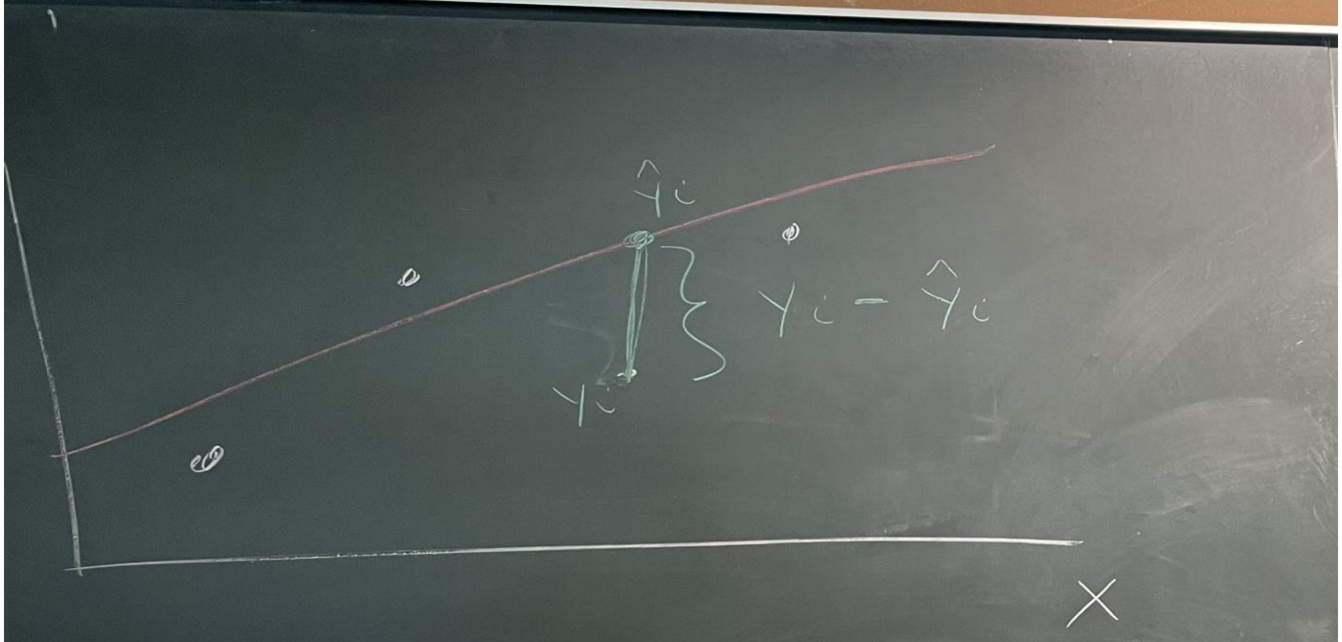
how good is our model?

residual: $y_i - \hat{y}_i$ } one example

Overall

minimize $\boxed{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$

RSS or SSE



$\hat{y}_i$
$y_i - \hat{y}_i$
y_i
X

Goals of fitting a linear model

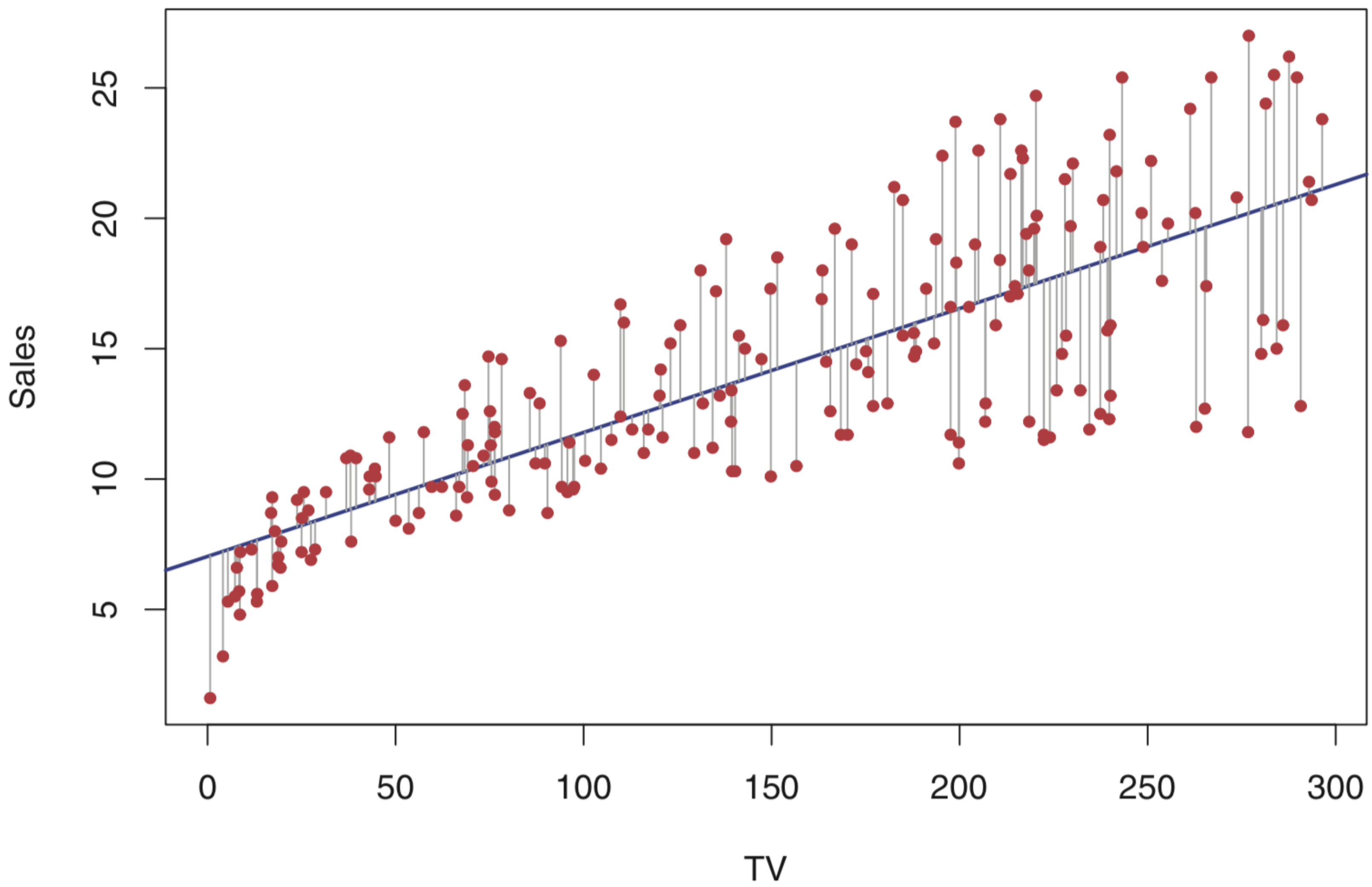
1) Which of the features/explanatory variables/predictors (x) are associated with the response variable (y)?

2) What is the relationship between x and y?

3) Can we predict y given a new x?

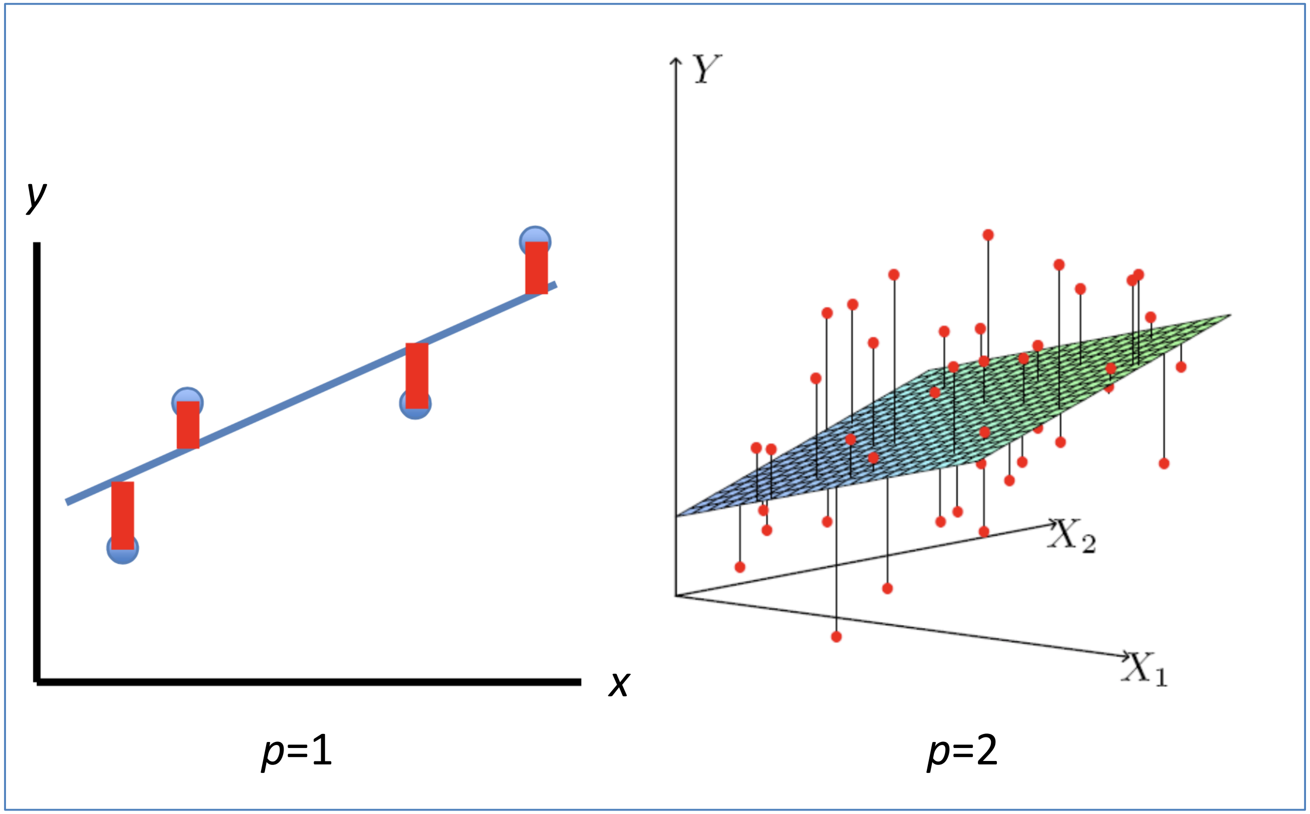
4) Is a linear model enough?

Example: predict sales from TV advertising budget



ISL: Figure 3.1

Linear model with 1 or 2 features



Slide: modified from Jessica Wu
Original: Eric Eaton

Linear Regression

- Output (*y*) is continuous, not a discrete label

- Learned model: *linear function* mapping input to output (a *weight* for each feature + *bias*)

- Goal: minimize the *RSS* (residual sum of squares) or *SSE* (sum of squared errors)

Maybe a linear model is not enough

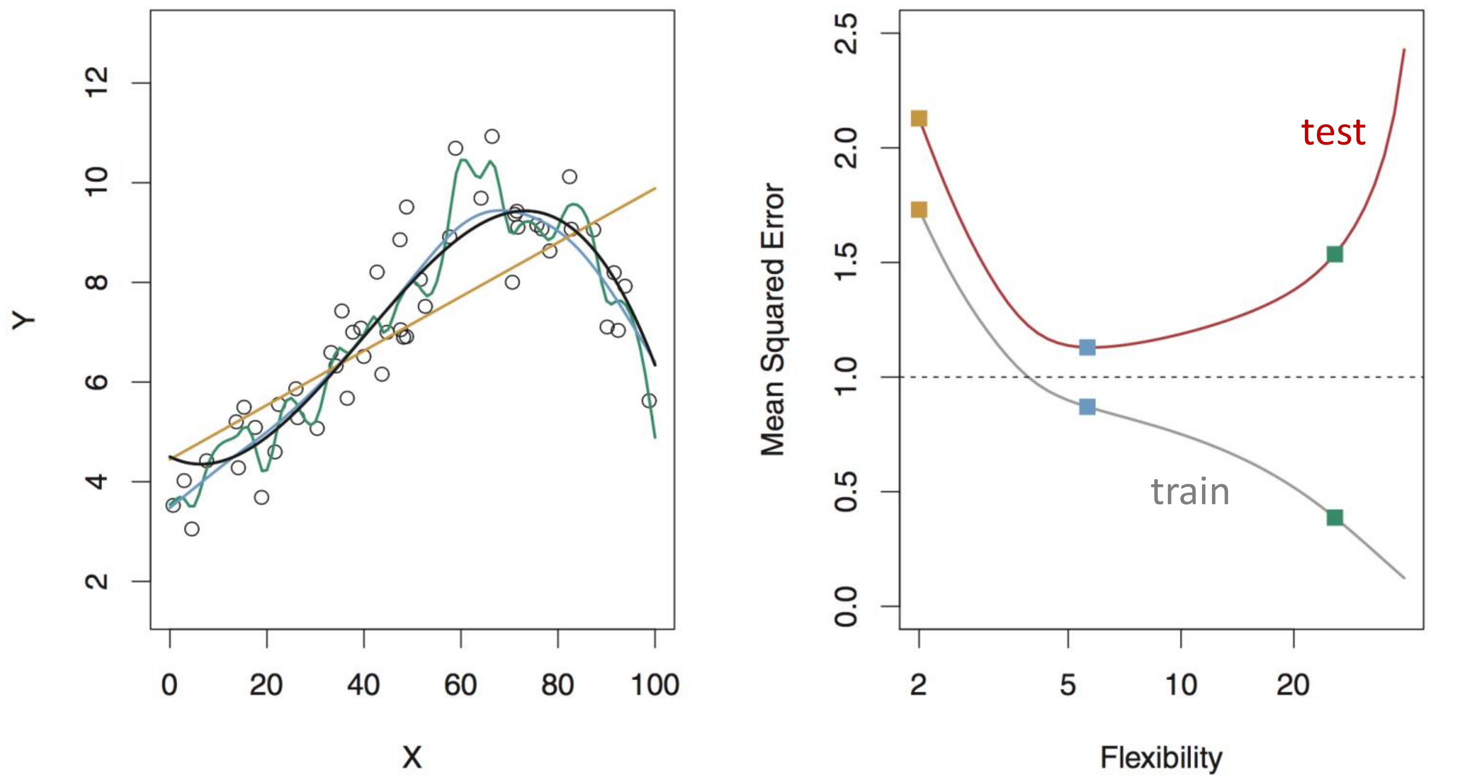


FIGURE 2.9. Left: *Data simulated from f, shown in black. Three estimates of f are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves).* Right: *Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.*

ISL, Fig 2.9

Command line arguments example

```
def parse_args():
    """Parse command line arguments (train and test data files)."""
    parser = argparse.ArgumentParser(description='climate change model analysis')

    # specify all command line options here
    parser.add_argument('train_filename', help='path to train csv file')
    parser.add_argument('-test', '--test_filename', help='path to test csv file')

    args = parser.parse_args()
    return args

def main() :
    args = parse_args()
    print(args.train_filename)
```