An Intro to Machine Learning

Anthony Gatti

University of Waterloo ECE 680

June 5th 2019

gattia@mcmaster.ca

Image Segmentation

Femoral Cartilage

Medial Tibial Cartilage

Lateral Tibial Cartilage

Patellar Cartilage

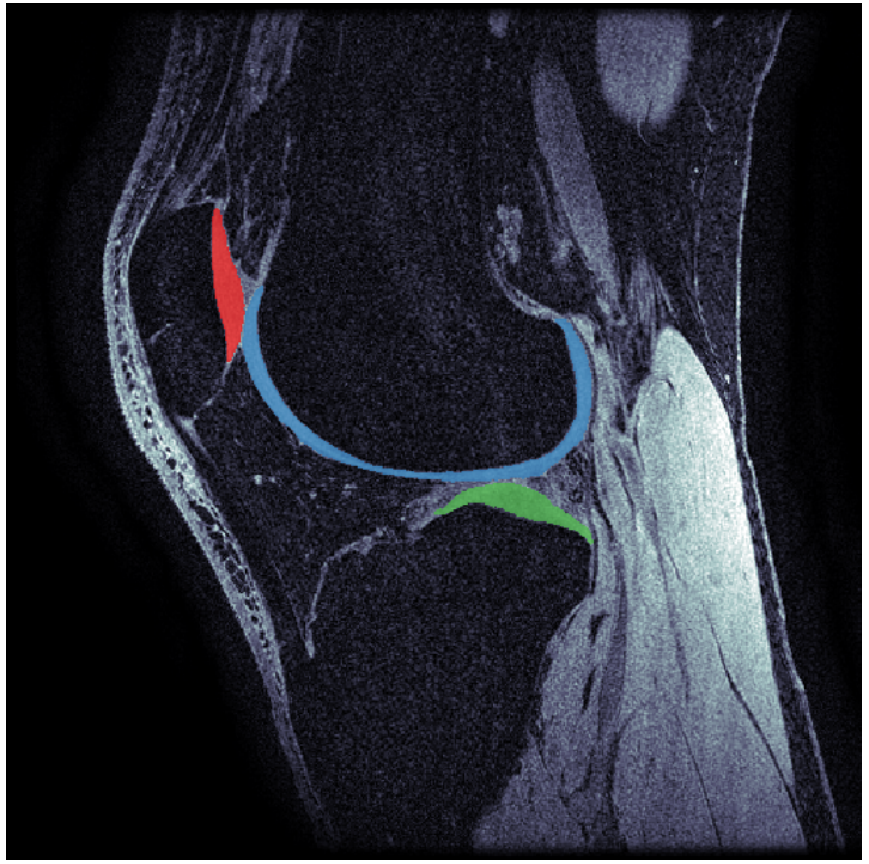
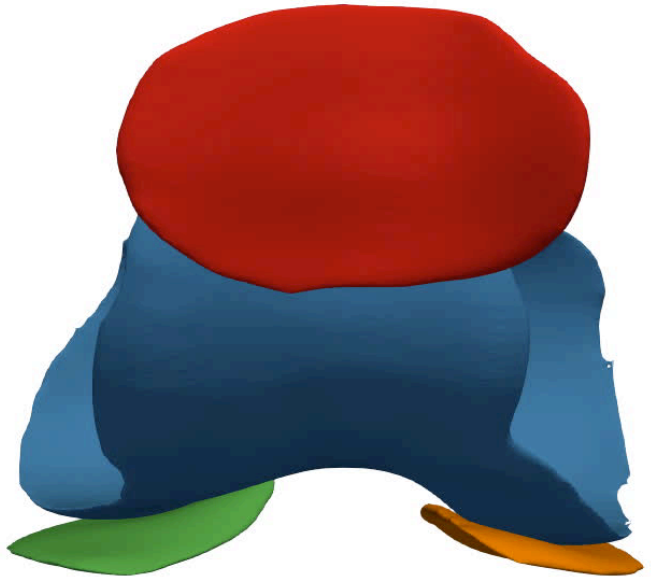


Image Segmentation



Femoral Cartilage

Medial Tibial Cartilage

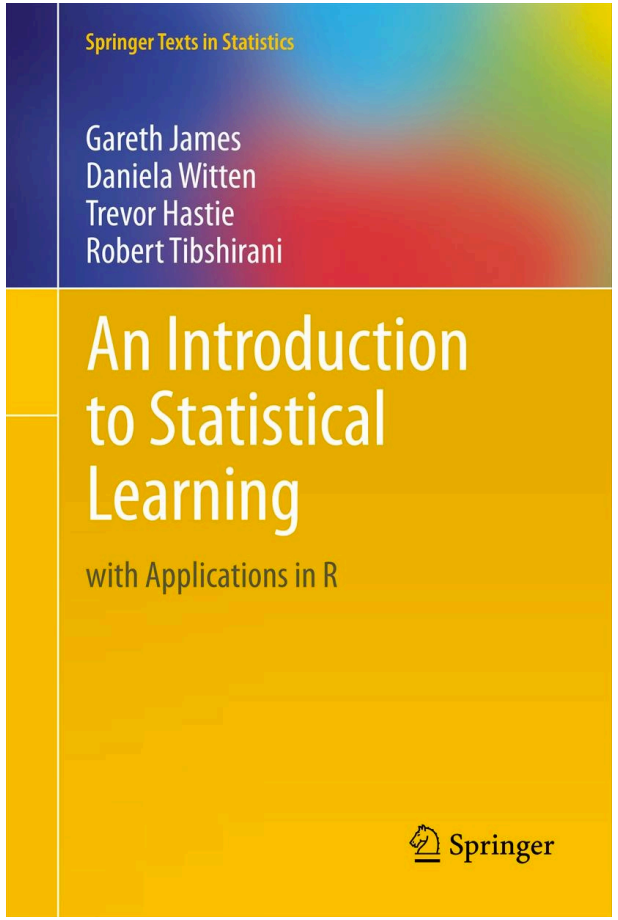
Lateral Tibial Cartilage

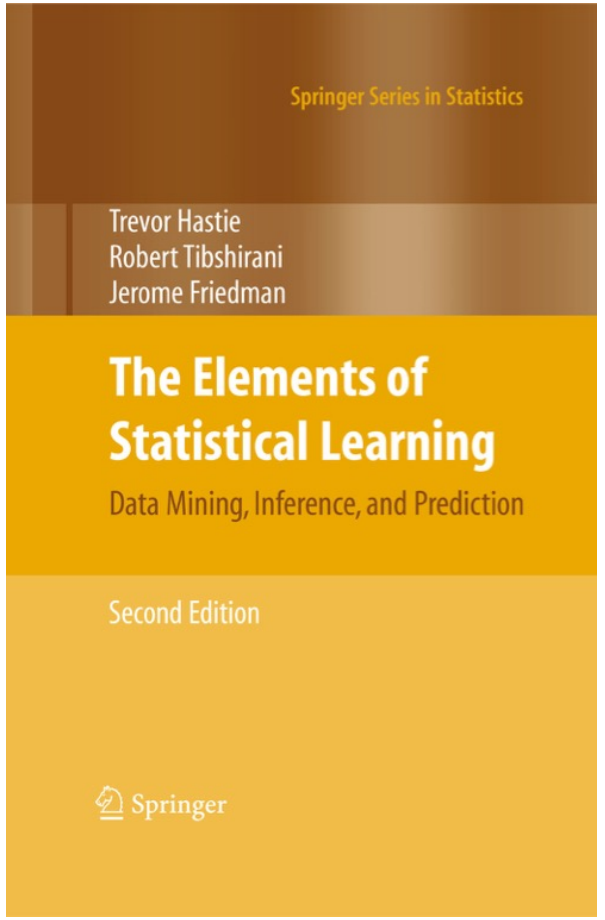
Patellar Cartilage

Overview

- What is Machine Learning?
- Types of Machine Learning (General)
- Machine Learning vs. Statistics
- Model Accuracy
- Model flexibility, training vs testing (overfitting)
- Bias vs. Variance
- Types of Machine Learning (Specific)

Resources


Springer Texts in Statistics
Gareth James
Daniela Witten
Trevor Hastie
Robert Tibshirani
An Introduction to Statistical Learning
with Applications in R
Springer


Springer Series in Statistics
Trevor Hastie
Robert Tibshirani
Jerome Friedman
The Elements of Statistical Learning
Data Mining, Inference, and Prediction
Second Edition
Springer

What is Machine Learning?

"the scientific study of algorithms and statistical models that computer systems use in order to perform a specific task effectively without using explicit instructions, relying on patterns and inference instead."

-Wikipedia

What is Machine Learning?

"the science of getting computers to act without being explicitly programmed."

-Coursera – Andrew Ng

What is Machine Learning?

“functionality that helps software perform a task without explicit programming or rules.”

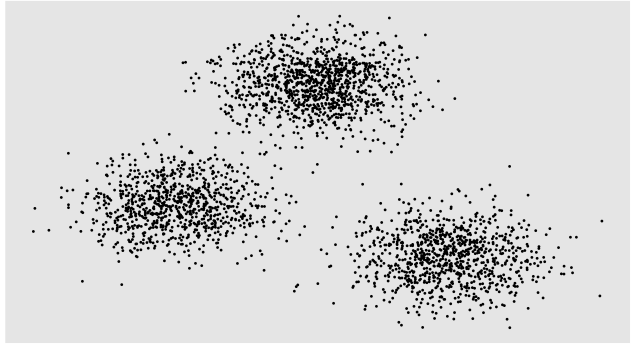
-Google Machine Learning Services

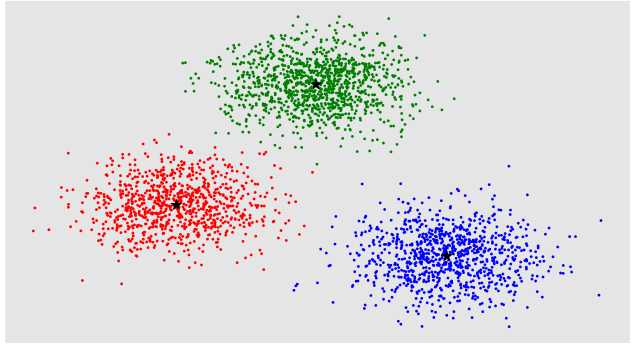
Types of Machine Learning

Supervised

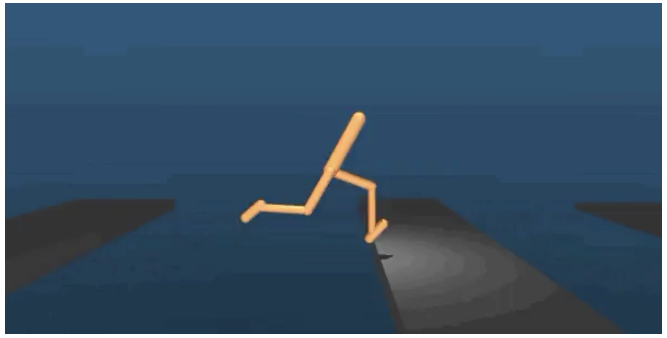
$$y = \beta_0 + \beta_1 x + e$$

Unsupervised





Reinforcement



https://www.youtube.com/watch?v=gn4nRCC9TwQ

What is Statistics?

"branch of mathematics working with data collection, organization, analysis, interpretation and presentation."

-Wikipedia

How is machine learning different from statistics?

Prediction vs. Inference

$$Y = f(X) + e$$

Prediction

$\hat{Y} = \hat{f}(X)$

$\hat{Y}$ = Prediction of Y

$\hat{f}(X)$ = “black box”

Prediction vs. Inference

$$Y = f(X) + e$$

Prediction

$\hat{Y} = \hat{f}(X)$

$\hat{Y}$ = Prediction of Y

$\hat{f}(X)$ = "black box"

$\hat{}$ the "hat" indicates it is an estimate.

Therefore have two forms of error:

1. $\hat{f}(X)$ compared to $f(X)$
2. $\hat{Y}$ compared to Y

.

Prediction vs. Inference

$$Y = f(X) + e$$

Prediction

$\hat{Y} = \hat{f}(X)$

$\hat{Y}$ = Prediction of Y

$\hat{f}(X)$ = "black box"

$\hat{}$ the "hat" indicates it is an estimate.

Therefore have two forms of error:

1. $\hat{f}(X)$ compared to $f(X)$
2. $\hat{Y}$ compared to Y

1. Is reducible error because we can try and get better models to make $\hat{f}(X)$ as close to $f(X)$ as possible.
2. Is irreducible error, $\hat{Y}$ by definition = Y - some error (e)

Prediction vs. Inference

$$Y = f(X) + e$$

Inference

$\hat{Y} = \hat{f}(X)$

$\hat{Y}$ = Prediction of Y

$\hat{f}(X)$ = Explainable model

Prediction vs. Inference

$$Y = f(X) + e$$

Inference

$\hat{Y} = \hat{f}(X)$

$\hat{Y}$ = Prediction of Y

$\hat{f}(X)$ = Explainable model

$\hat{f}(X)$ is no longer a black box. We want to understand $\hat{f}(X)$ and identify the simplest/most explainable model.

Prediction vs. Inference

$$Y = f(X) + e$$

Inference

$\hat{Y} = \hat{f}(X)$

$\hat{Y}$ = Prediction of Y

$\hat{f}(X)$ = Explainable model

$\hat{f}(X)$ is no longer a black box. We want to understand $\hat{f}(X)$ and identify the simplest/most explainable model.

Want to know:

- Which variables associated with Y
- Direction of relationships
- Is it linear?

Goal Machine Learning: Identify $\hat{f}(X) \approx f(X)$

Model Accuracy (error)

$$Y = \hat{f}(X) + e$$

$$e = Y - \hat{f}(X)$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

$$MSE = Mean\ Squared\ Error$$

Model Accuracy (error)

$$Y = \hat{f}(X) + e$$

$$e = Y - \hat{f}(X)$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

$$MSE = Mean\ Squared\ Error$$

Model Accuracy (error)

$$Y = \hat{f}(X) + e$$

$$e = Y - \hat{f}(X)$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

$$MSE = Mean\ Squared\ Error$$

Model Flexibility & Training vs. Testing

- More flexible will decrease training error
- Overfitting = fit model too well to training data
- Model should fit training data better than test – just not too much better
- More flexible does not mean more accurate

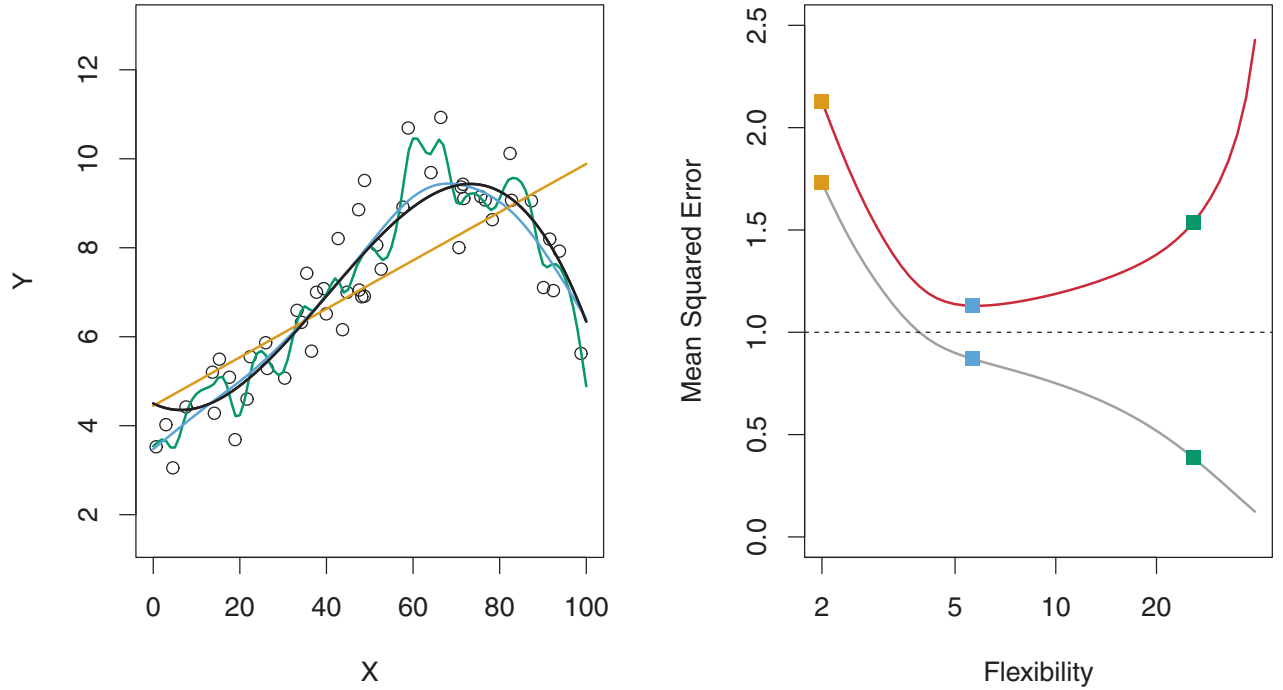


FIGURE 2.9. Left: *Data simulated from f, shown in black. Three estimates of f are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves).* Right: *Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed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

An Introduction to Statistical Learning
(James, Witten, Hastie, & Tibshirani)

Model Flexibility & Training vs. Testing

- More flexible will decrease training error
- Overfitting = fit model too well to training data
- Model should fit training data better than test – just not too much better
- More flexible does not mean more accurate

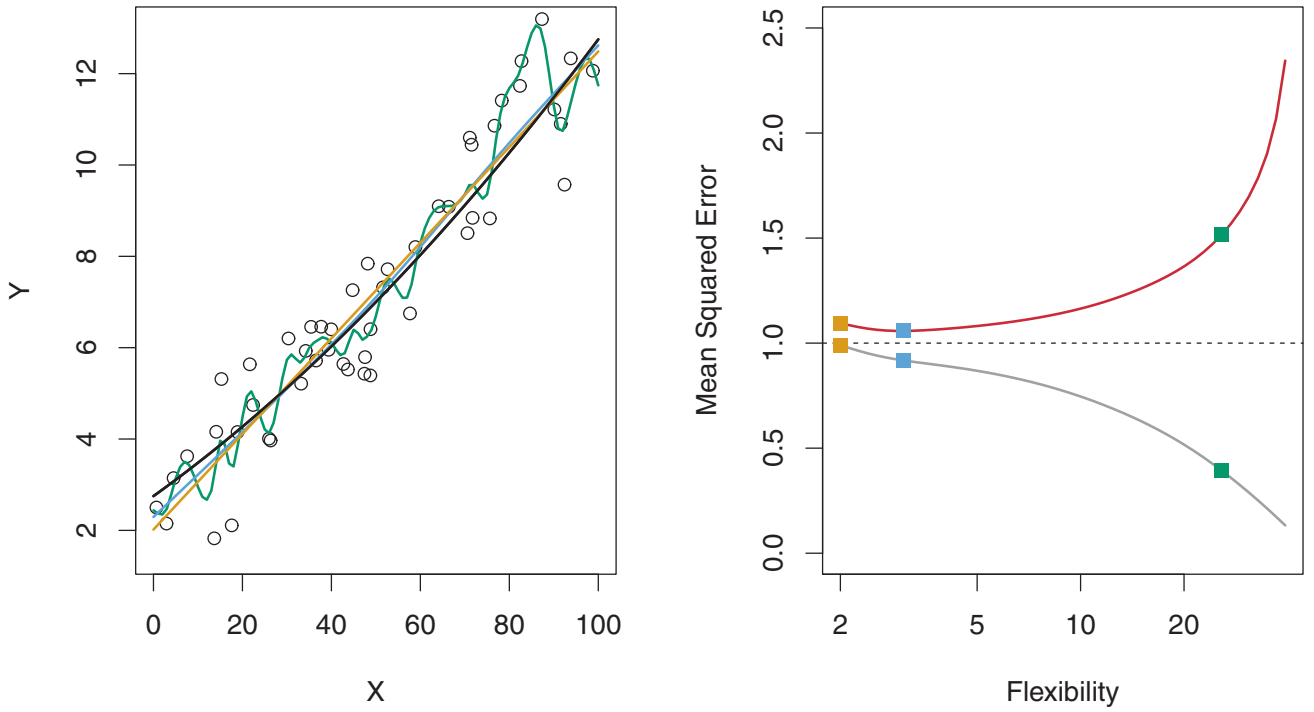


FIGURE 2.10. *Details are as in Figure 2.9, using a different true f that is much closer to linear. In this setting, linear regression provides a very good fit to the data.*

An Introduction to Statistical Learning
(James, Witten, Hastie, & Tibshirani)

Reducible Error = Bias & Variance

Variance

- How much $\hat{f}(X)$ changes when training data changes.
- More flexible models have higher variance
- E.g. small changes in the data will have large effect on green fit

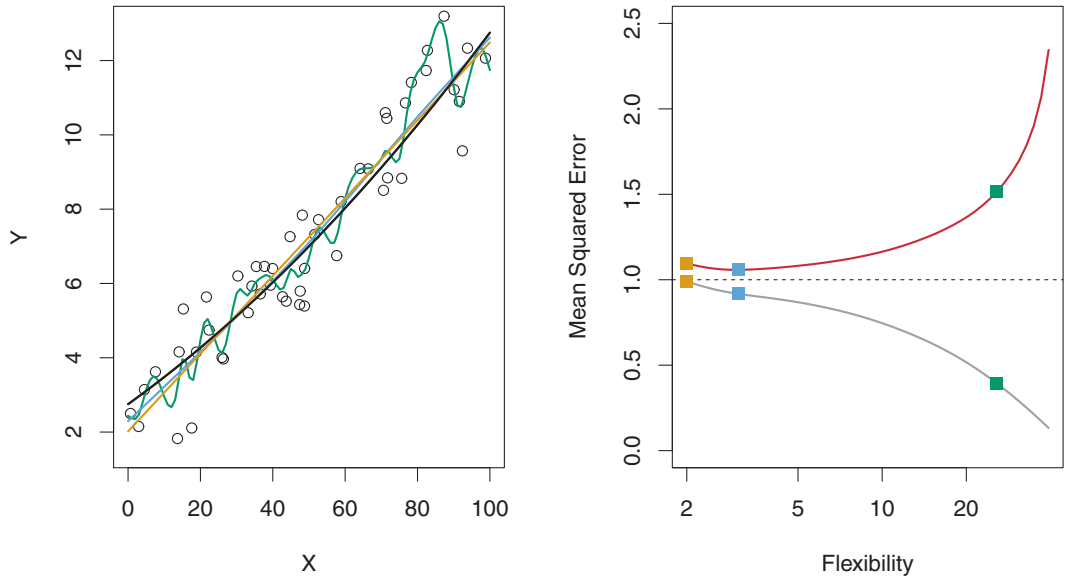


FIGURE 2.10. *Details are as in Figure 2.9, using a different true f that is much closer to linear. In this setting, linear regression provides a very good fit to the data.*

An Introduction to Statistical Learning
(James, Witten, Hastie, & Tibshirani)

Reducible Error = Bias & Variance

Bias

- Ability/inability for model to approximate $\hat{f}(X)$
- Increasing flexibility decreases bias.
- E.g. Regardless of the data, linear regression (orange line) will never fit this data well.

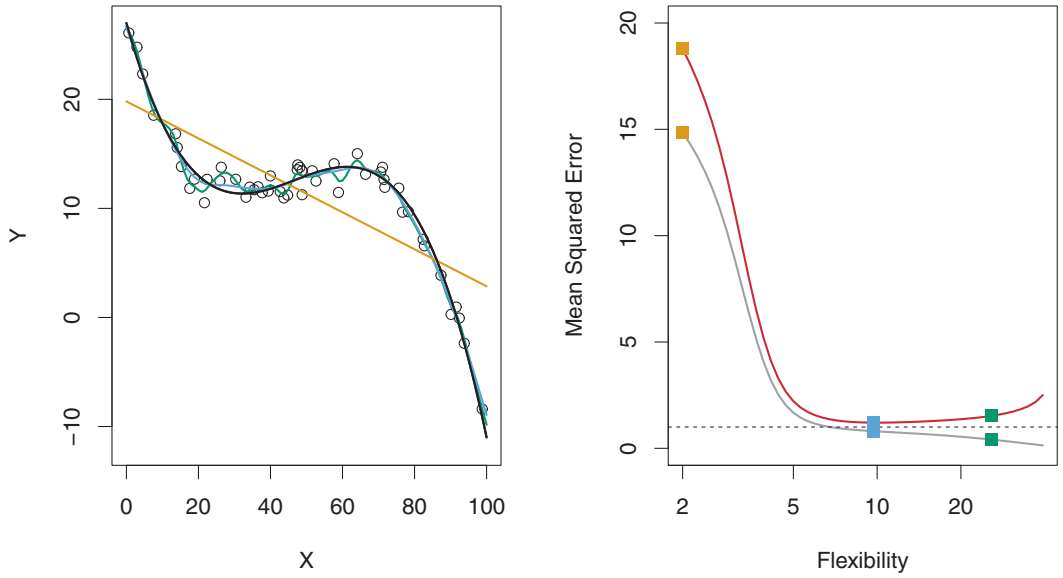
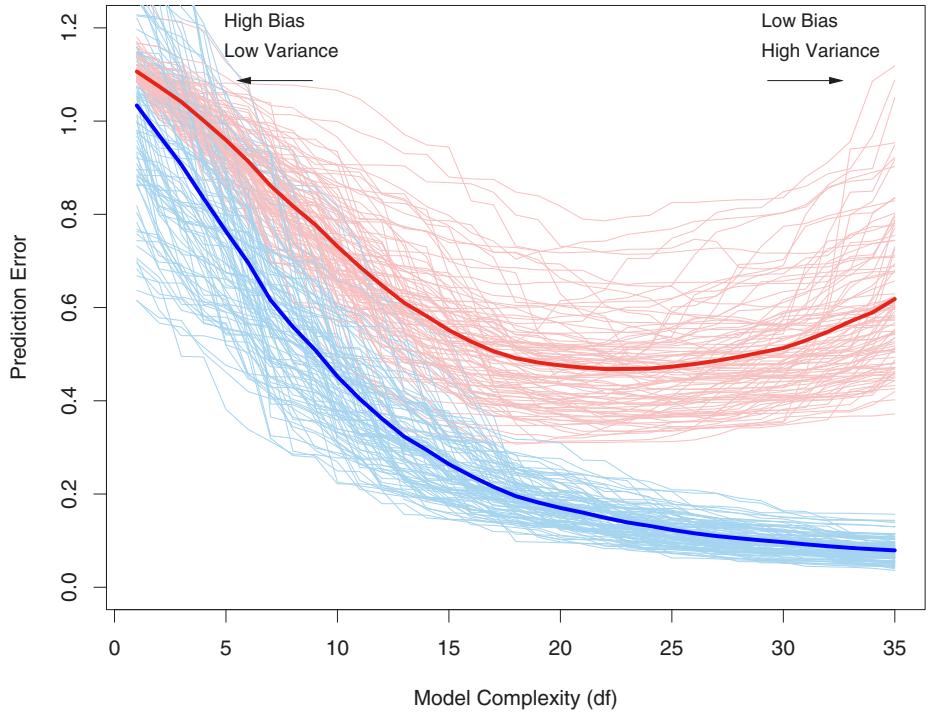


FIGURE 2.11. *Details are as in Figure 2.9, using a different f that is far from linear. In this setting, linear regression provides a very poor fit to the data.*

An Introduction to Statistical Learning
(James, Witten, Hastie, & Tibshirani)

In general, increased flexibility leads to increased variance and decreased bias.

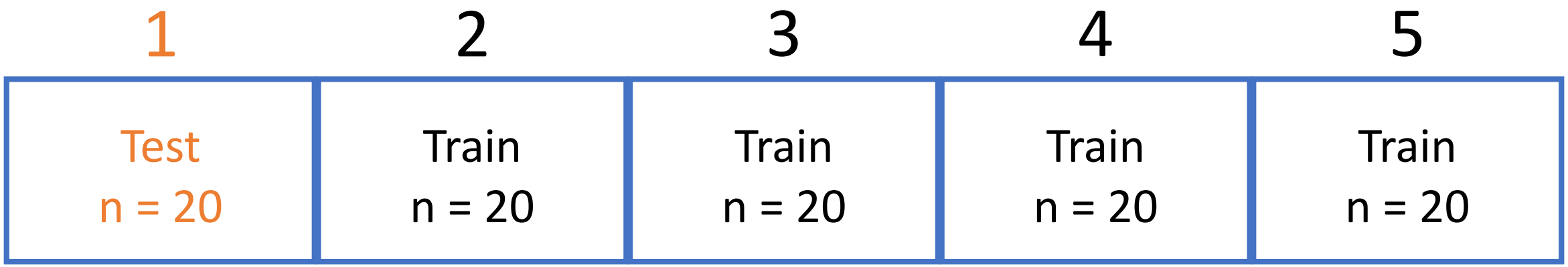


The Elements of Statistical Learning (Hastie, Tibshirani, & Friedman)

FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\text{err}}$*, while the light red curves show the conditional test error* $\text{Err}_{\mathcal{T}}$ *for* 100 *training sets of size* 50 *each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\text{E}[\overline{\text{err}}]$.

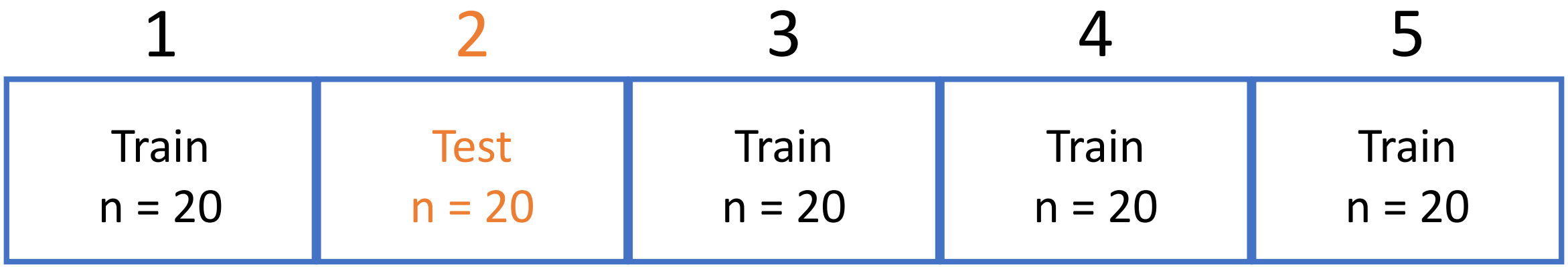
K-fold Cross Validation

E.g., K = 5, n=100



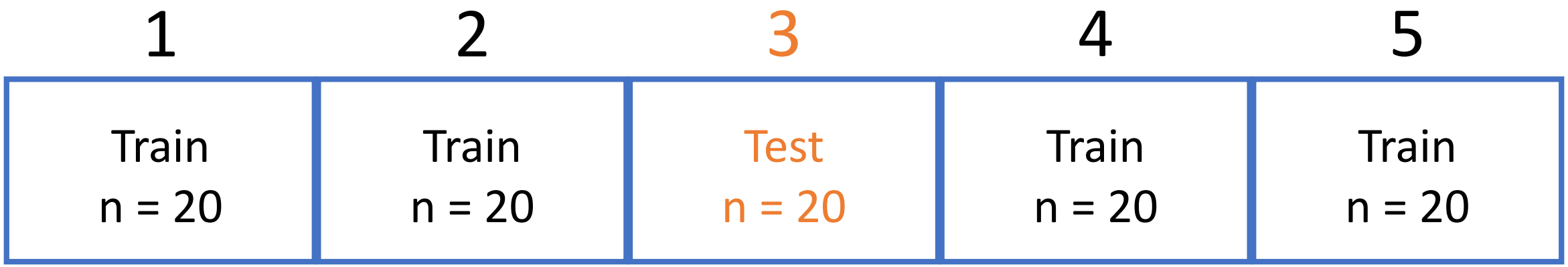
K-fold Cross Validation

E.g., K = 5, n=100



K-fold Cross Validation

E.g., K = 5, n=100

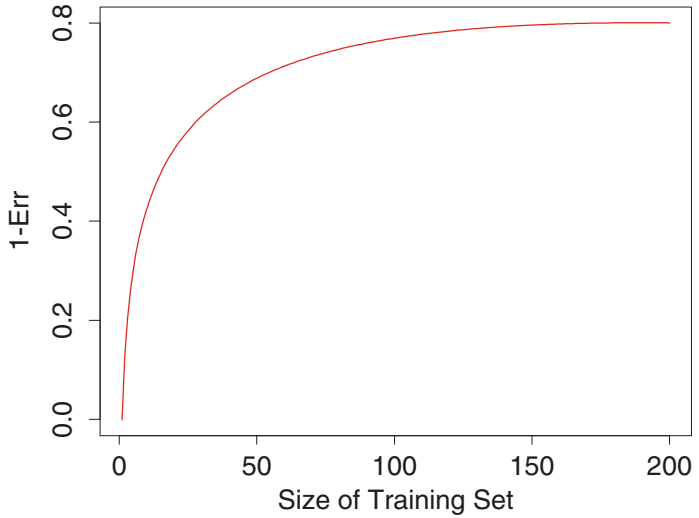


K-fold Cross Validation

- K can be any number.
- If K = n (number of samples in data) then this is called “leave-one-out” because each iteration only leaving one sample out for testing.
- Important to note – with higher K the training datasets are more and more similar.
- K increases the variance increases, but the bias decreases.
  - Variance increases because we have less variability between training datasets. Therefore, results are highly dependent on current data and a new dataset might produce surprisingly different results.
  - Bias decreases because more data means better model fit. Amount of data needed to fit model dependent on method. In general more = better.

K-fold Cross Validation

- Bias decreases because more data means better model fit. Amount of data needed to fit model dependent on method. In general more = better.



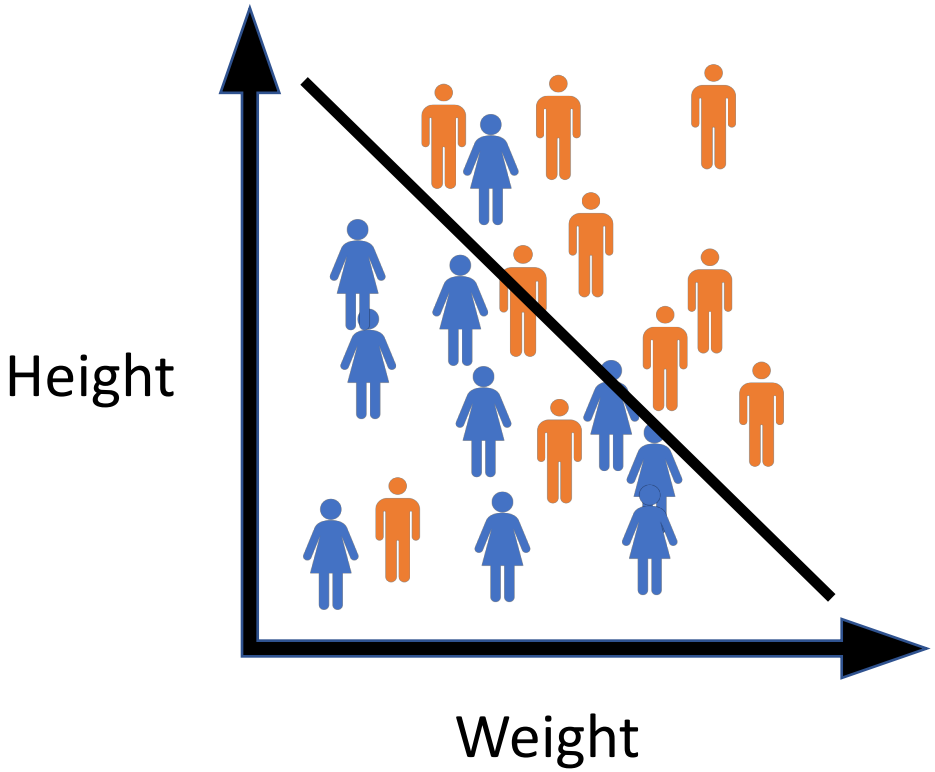
The Elements of Statistical Learning (Hastie, Tibshirani, & Friedman)

FIGURE 7.8. *Hypothetical learning curve for a classifier on a given task: a plot of* $1 - \text{Err}$ *versus the size of the training set* N*. With a dataset of* 200 *observations,* 5*-fold cross-validation would use training sets of size* 160*, which would behave much like the full set. However, with a dataset of* 50 *observations fivefold cross-validation would use training sets of size* 40*, and this would result in a considerable overestimate of prediction error.*

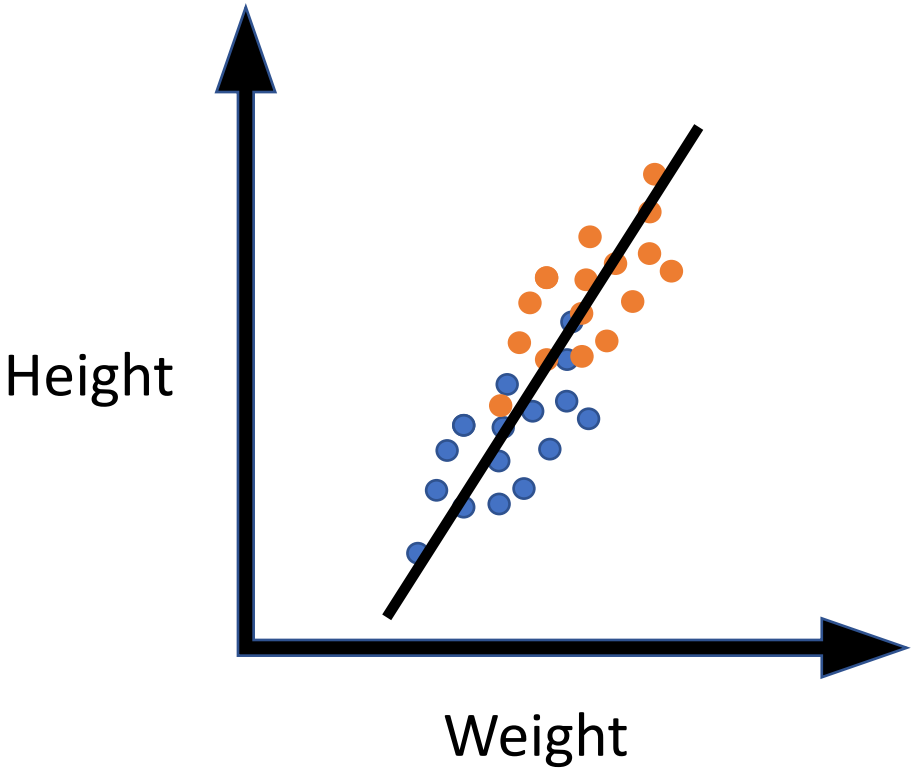
Types of Machine Learning

Supervised Learning

??



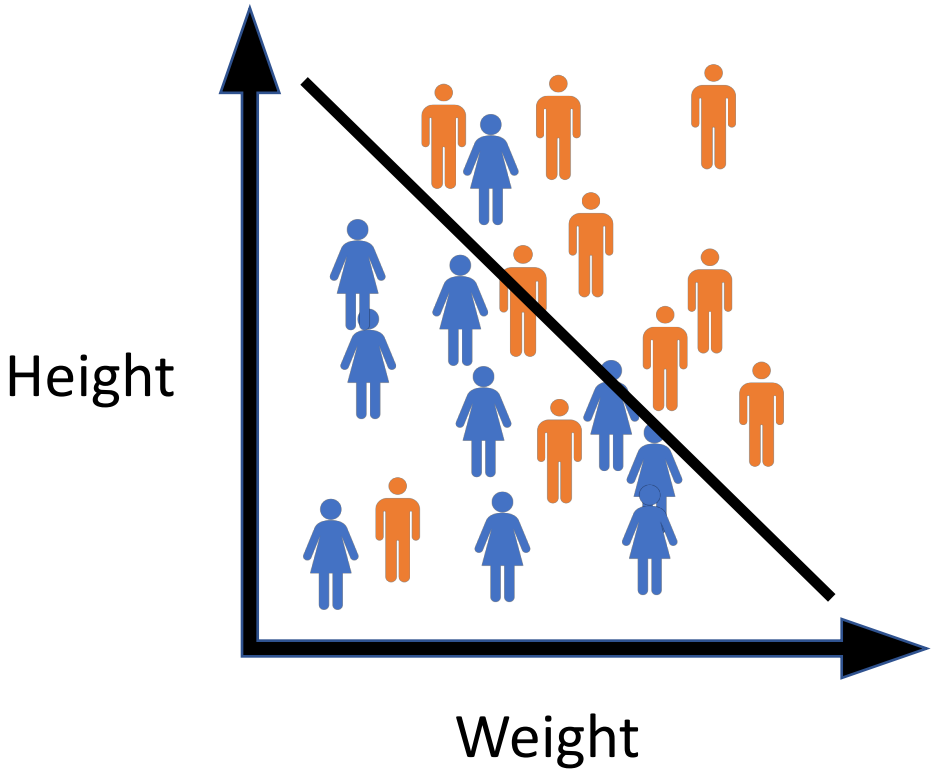
??



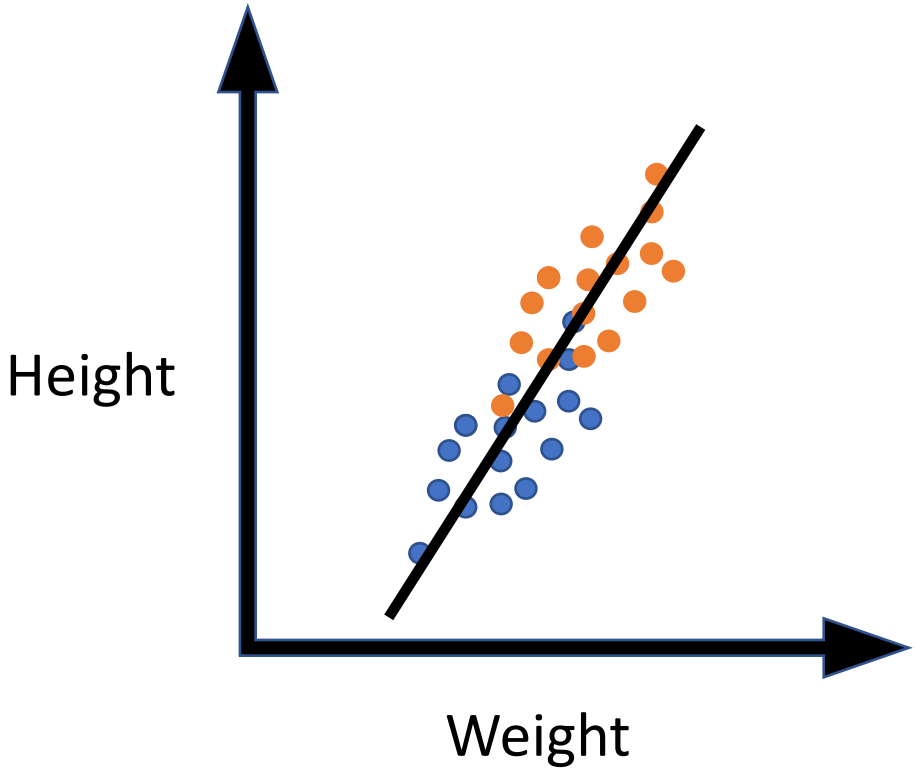
$$\text{Height} = \beta_0 + \beta_1 weight + \beta_2 sex$$

Supervised Learning

Classification



Regression

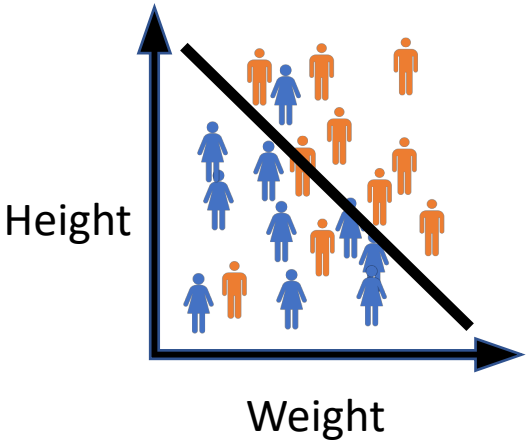


$$\text{Height} = \beta_0 + \beta_1 weight + \beta_2 sex$$

Supervised Learning

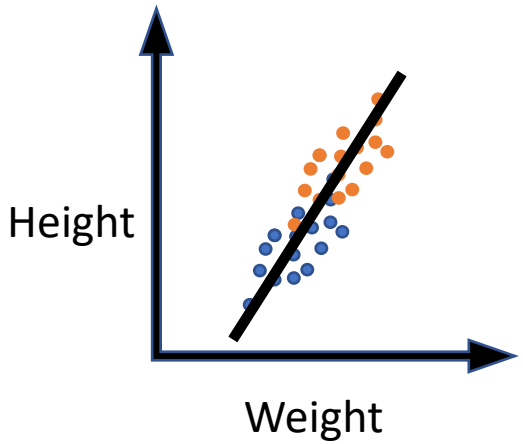
Classification

- K-Nearest Neighbors
- Support Vector Machine
- Decision Tree
- Logistic Regression
- Neural Network

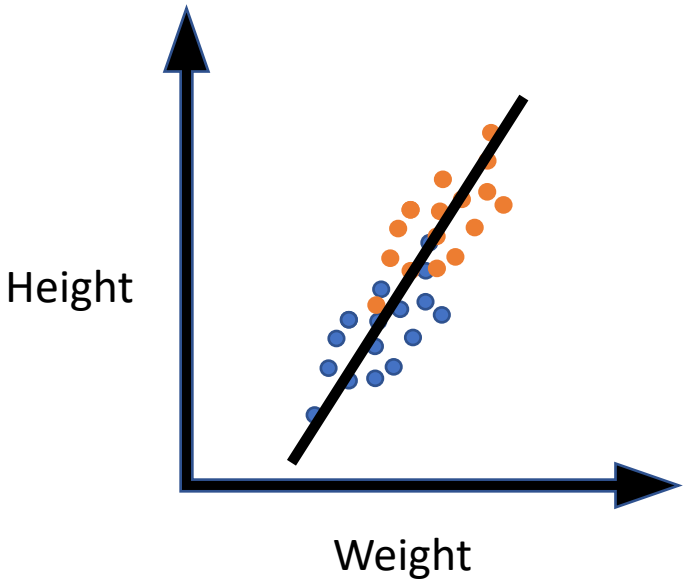


Regression

- Least Squares Regression
- Ridge Regression
- Lasso Regression
- Neural Network



Regression



Regression models take input data and predict an output.

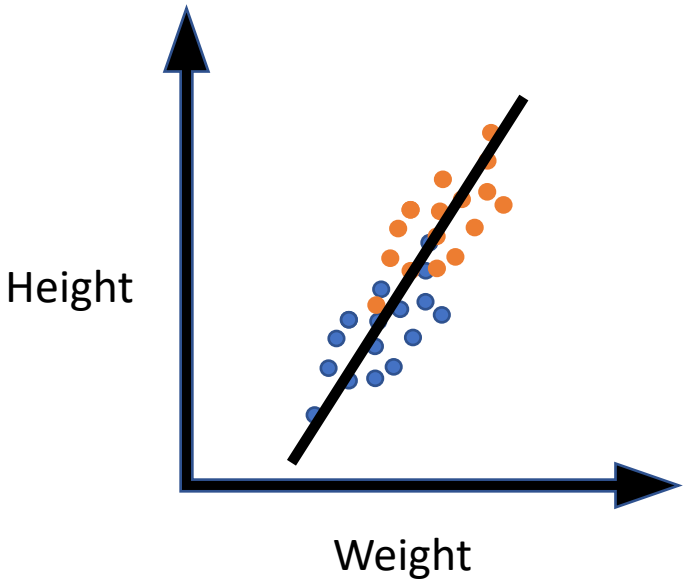
<u>Least Squares Regression</u>

$$y = \beta_0 + \beta_1 x + e$$

$$e = y - \beta_0 + \beta_1 x$$

Goal is to <u>minimize</u> error

Regression



Regression models take input data and predict an output.

Least Squares Regression

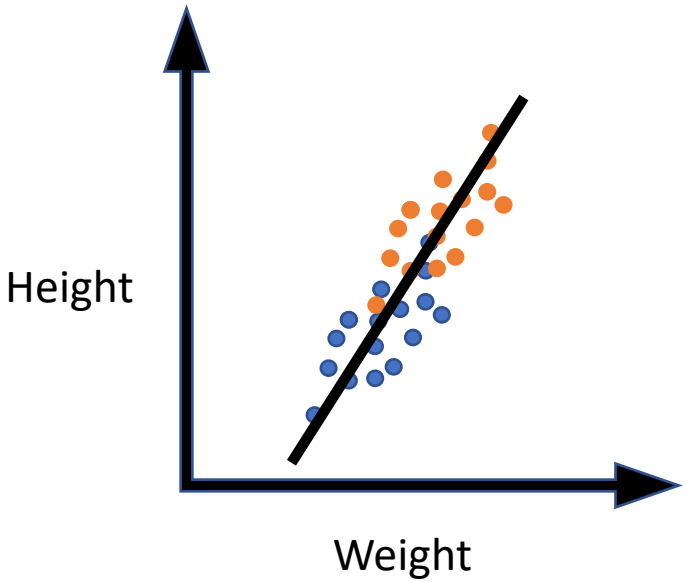
$$e = y - \beta_0 + \beta_1 x$$

$$RSS = \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2$$

(RSS = residual sum of squares)

Goal is to minimize RSS

Regression



Regression models take input data and predict an output.

Least Squares Regression

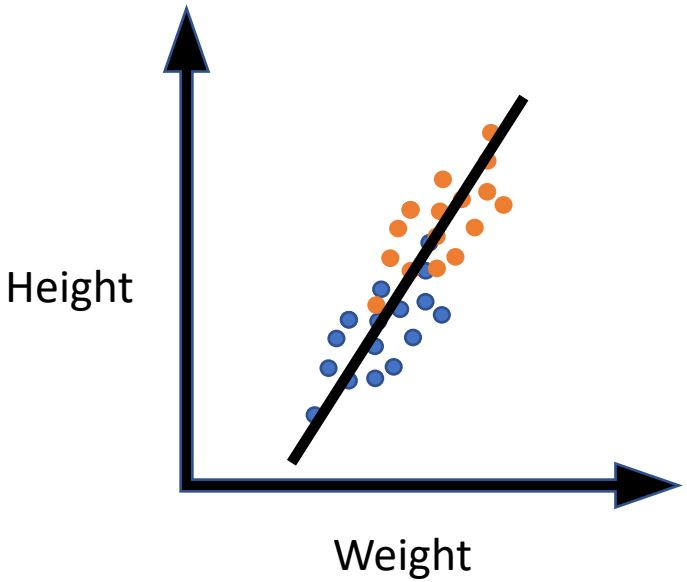
$$e = y - \beta_0 + \beta_1 x$$

$$RSS = \sum_{i=1}^{n} (y_i - (\sum_{j=1}^{p} \beta_0 + \beta_j x_{ij}))^2$$

(RSS = residual sum of squares)

Goal is to minimize RSS

Regression



Ridge Regression

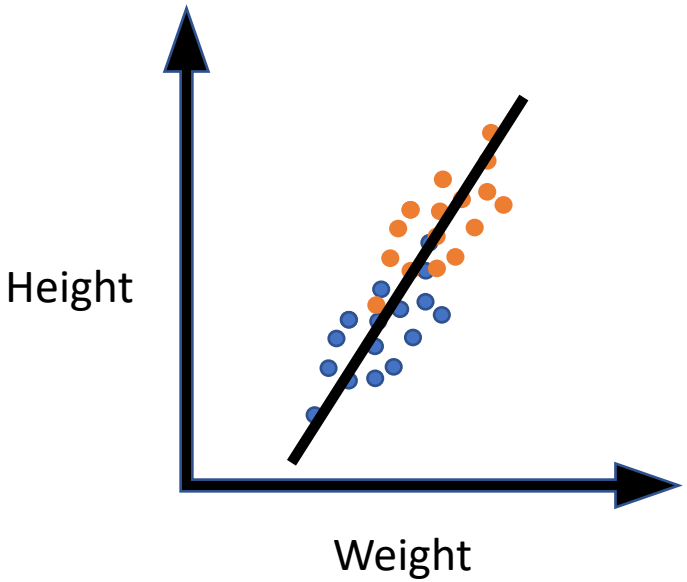
Error term is RSS plus a term used to minimize the coefficients (↓flexibility)

$$e = y - \beta_0 + \beta_1 x$$

$$RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$

Goal is to minimize the above eqn.

Regression



Lasso Regression

Same as ridge, but different term to minimize coefficients (↓flexibility).

$$e = y - \beta_0 + \beta_1 x$$

$$RSS + \lambda \sum_{j=1}^{p} \beta_j$$

Goal is to minimize the above eqn.

Classification

Logistic Regression

$$p(X) = \frac{e^{\beta_0+\beta_0 X}}{1+e^{\beta_0+\beta_0 X}}$$

- Fit data to logistic function
- Logistic function returns values between 0-1
- Output = probability
- Small $e^{\beta_0+\beta_0 X}$ = 0
- Large $e^{\beta_0+\beta_0 X}$ = 1
- $e^{\beta_0+\beta_0 X}$ can include multiple predictors

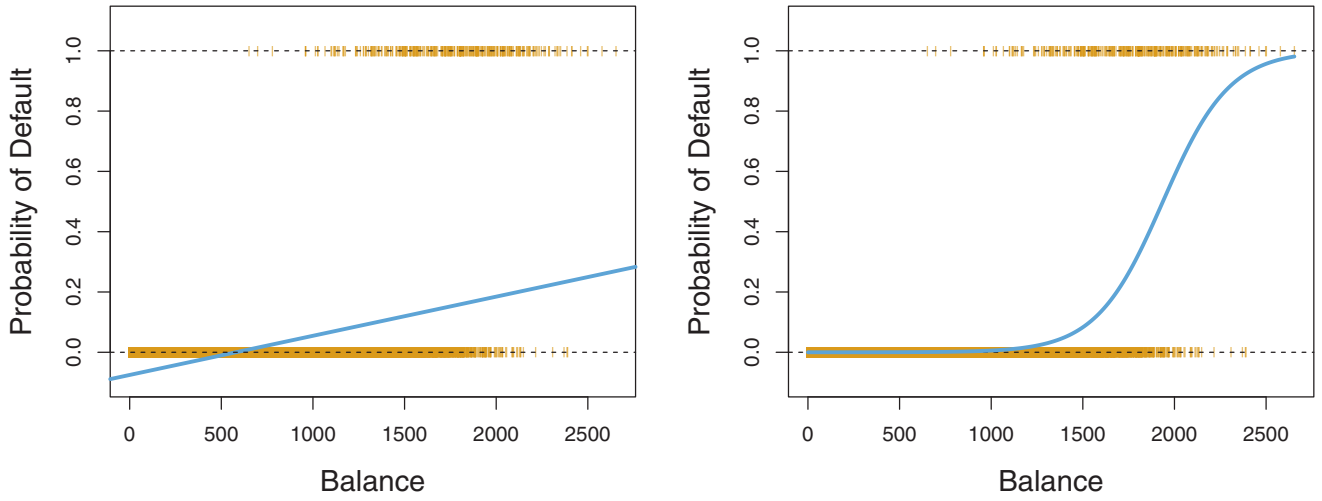
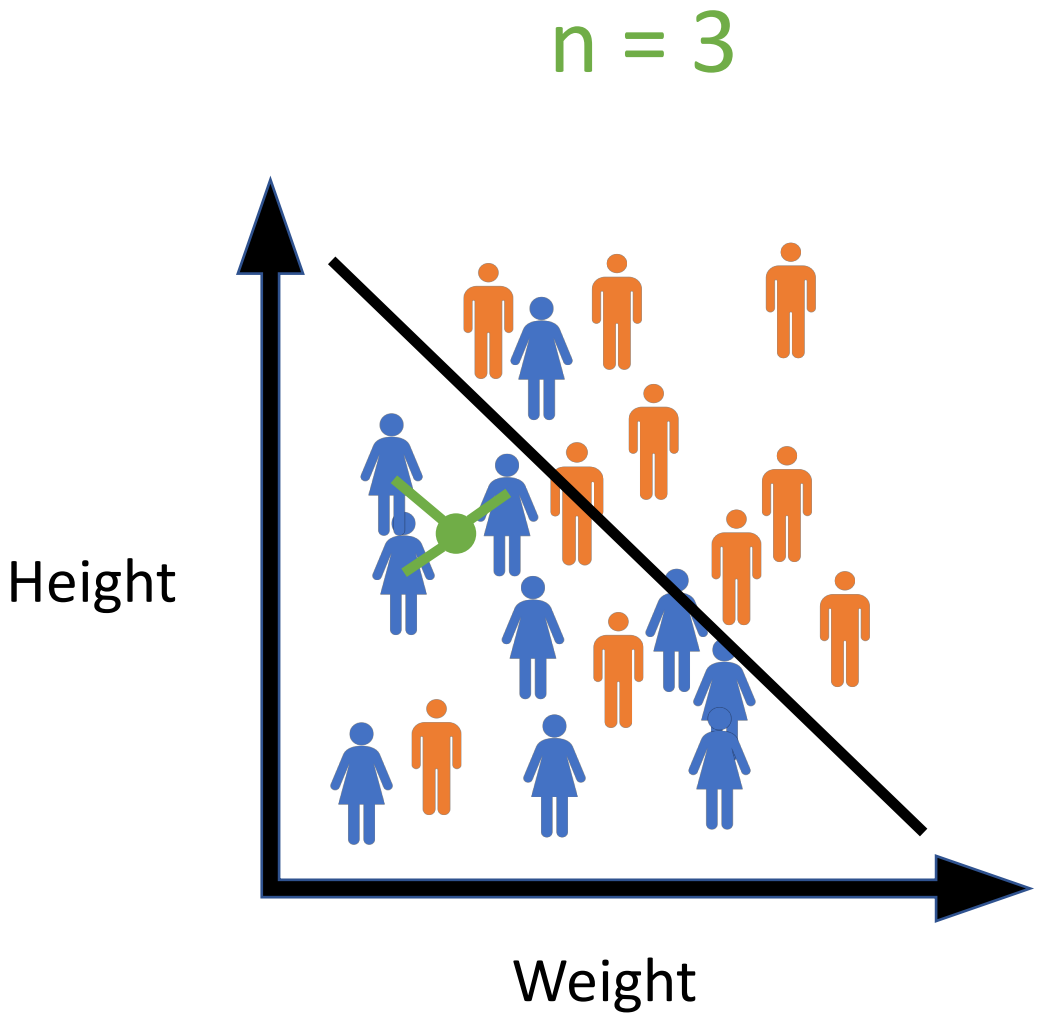


FIGURE 4.2. *Classification using the* `Default` *data.* Left: *Estimated probability of* `default` *using linear regression. Some estimated probabilities are negative! The orange ticks indicate the 0/1 values coded for* `default` *(*`No` *or* `Yes`*).* Right: *Predicted probabilities of* `default` *using logistic regression. All probabilities lie between* 0 *and* 1.

Classification

K-Nearest Neighbours

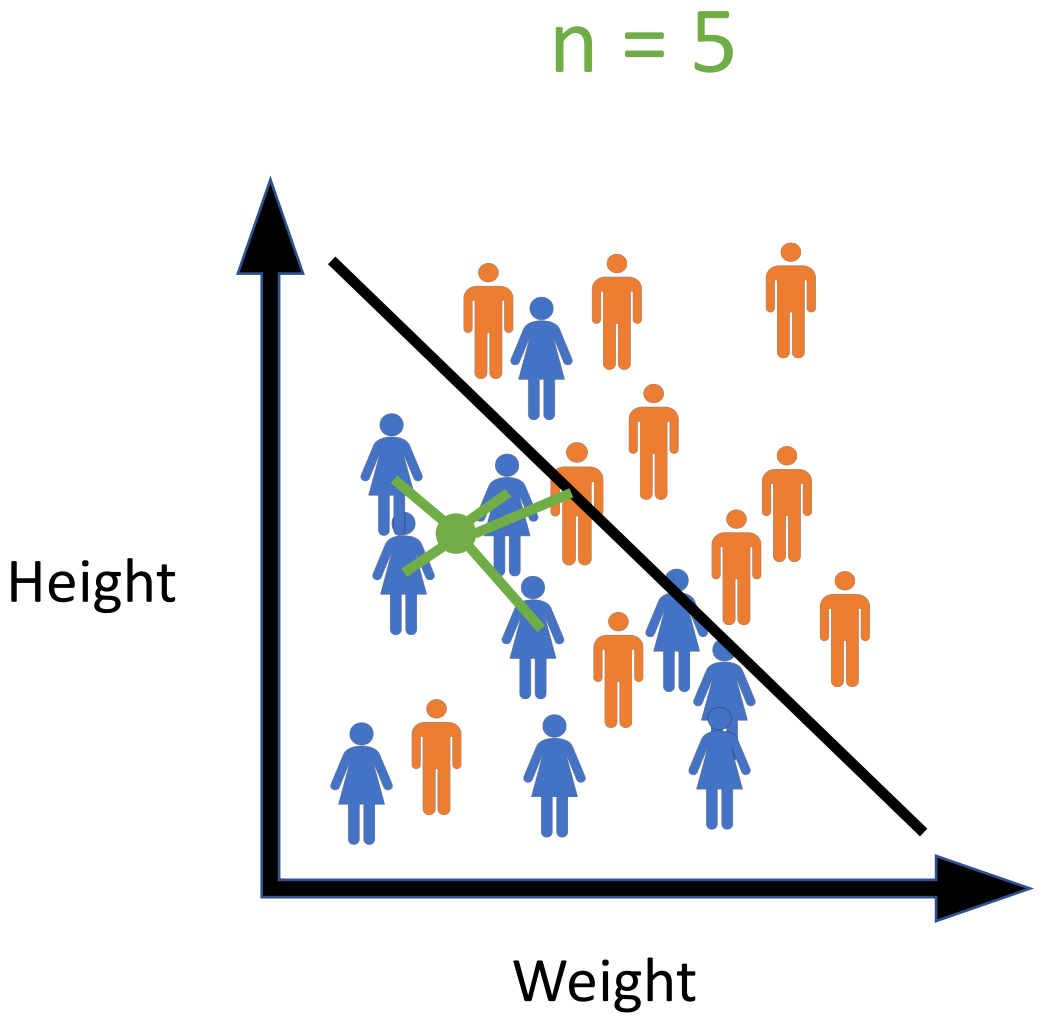
- Trying to classify new point belonging to a “class”
- Find K “nearest neighbours” to point
- Assign point to most frequent class in neighbours.



Classification

K-Nearest Neighbours

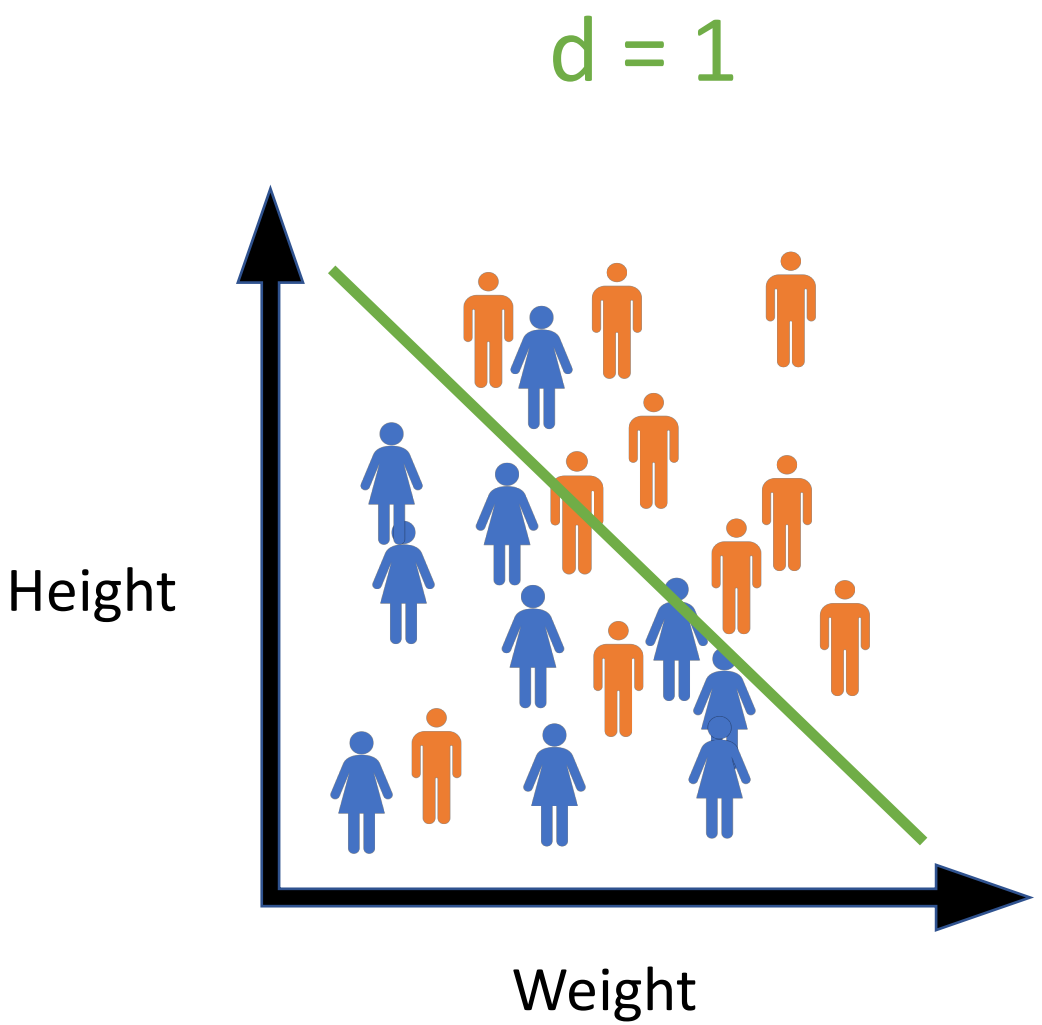
- Trying to classify new point belonging to a “class”
- Find K “nearest neighbours” to point
- Assign point to most frequent class in neighbours.



Classification

Support Vector Machine

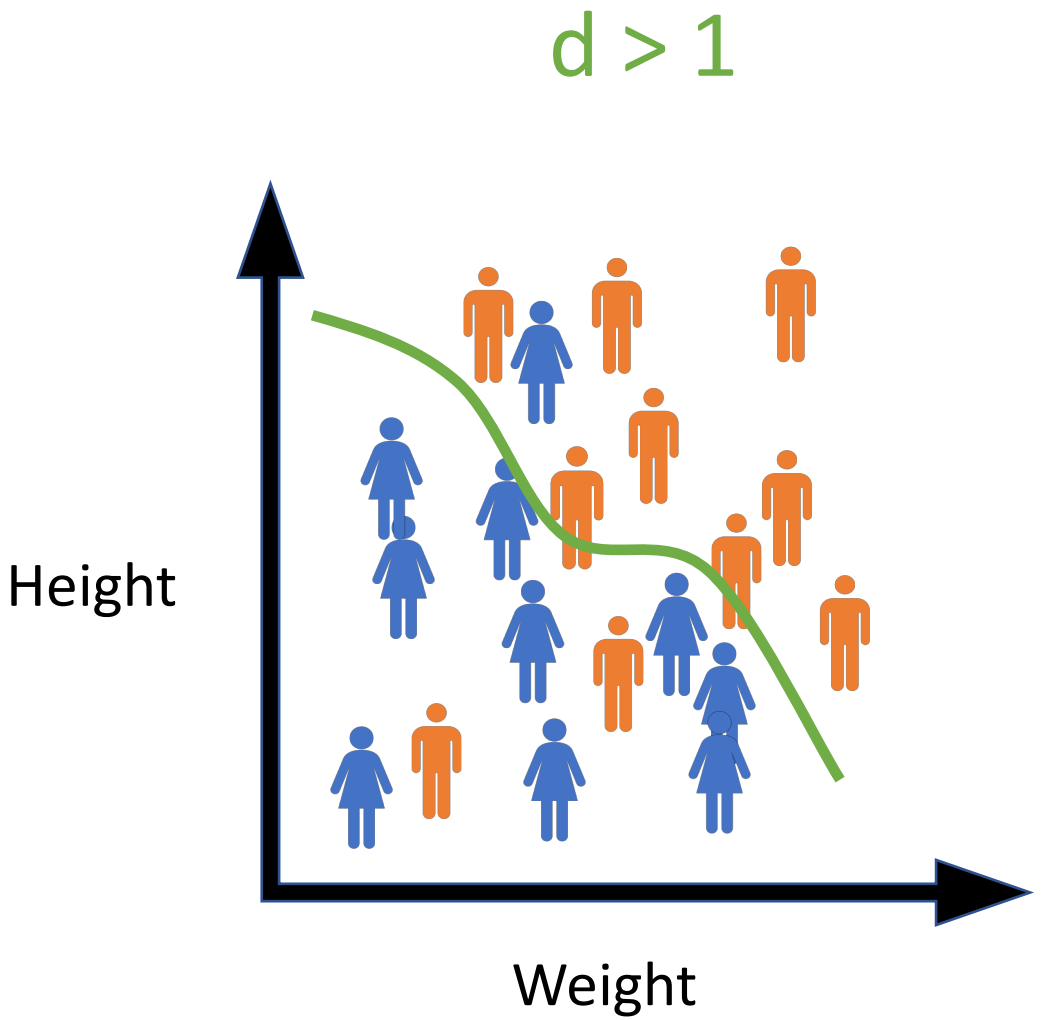
- Fit boundary to separate two classes
- Boundary based on a “kernel” that is raised to degree d
- d = 1 is linear
- Higher d leads to more flexibility in boundary & potentially overfit



Classification

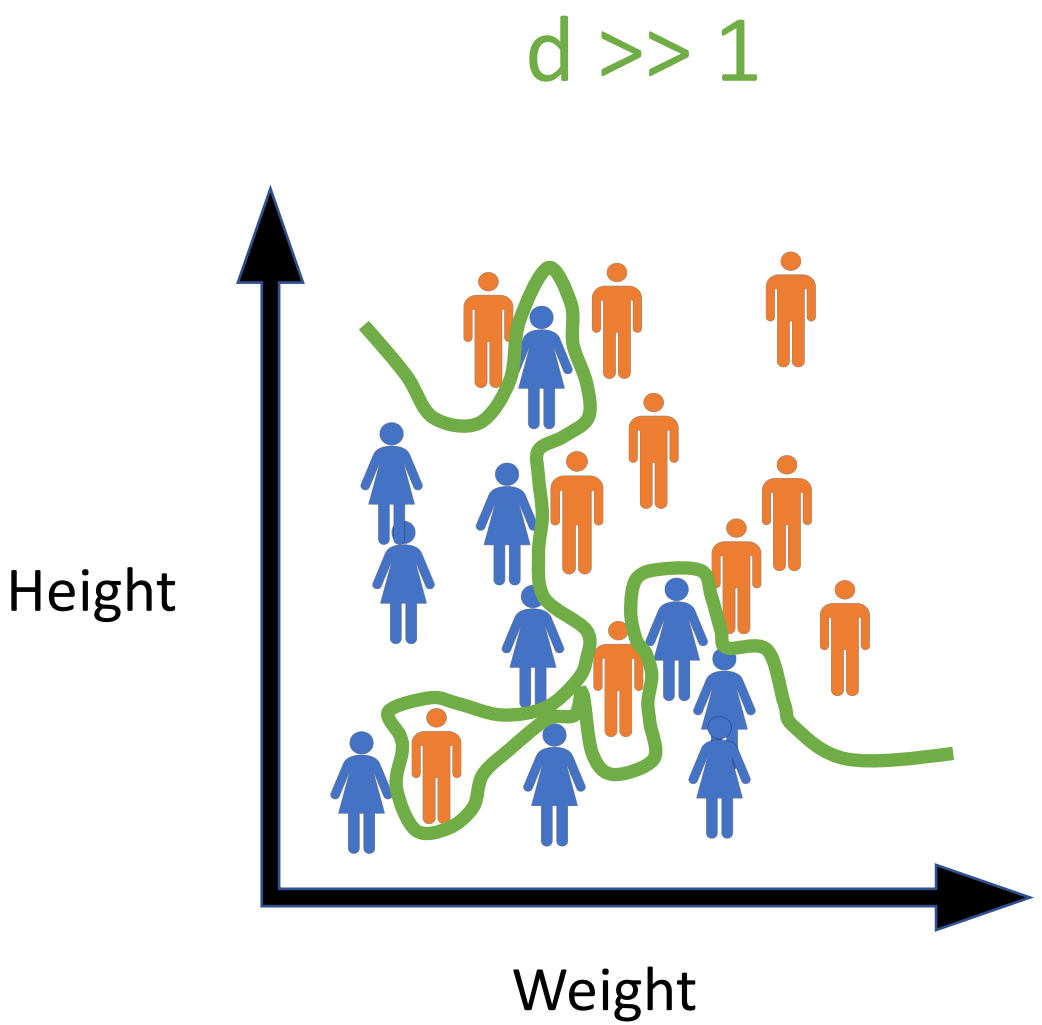
Support Vector Machine

- Fit boundary to separate two classes
- Boundary based on a “kernel” that is raised to degree d
- d = 1 is linear
- Higher d leads to more flexibility in boundary & potentially overfit



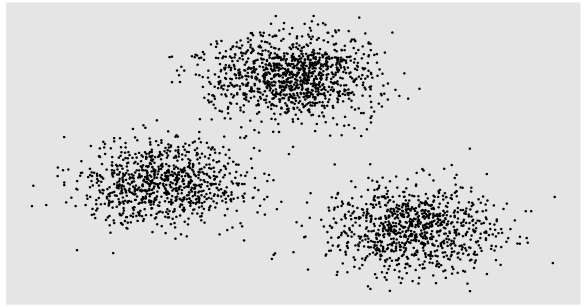
Classification
Support Vector Machine

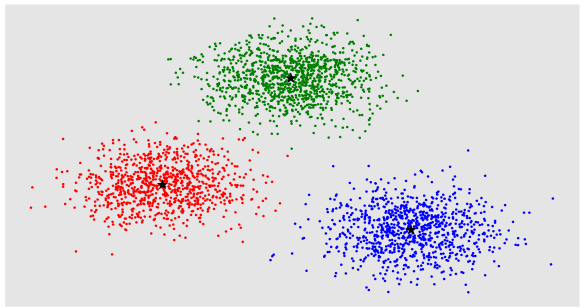
- Fit boundary to separate two classes
- Boundary based on a “kernel” that is raised to degree d
- d = 1 is linear
- Higher d leads to more flexibility in boundary & potentially overfit



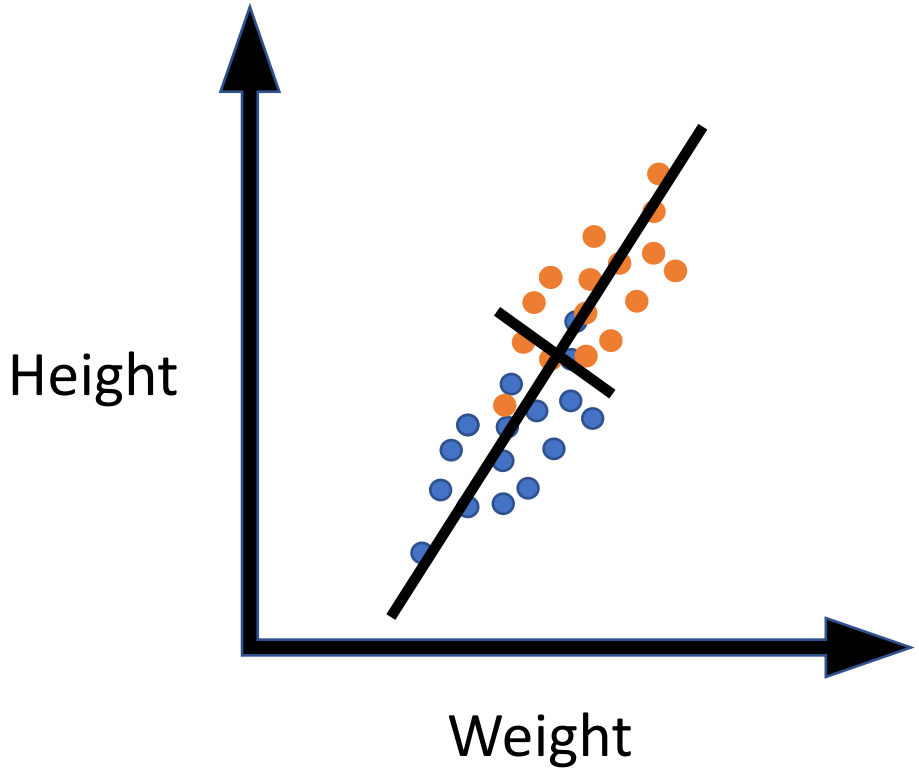
Unsupervised Learning

??



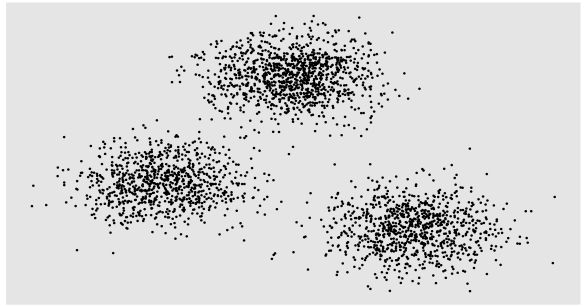


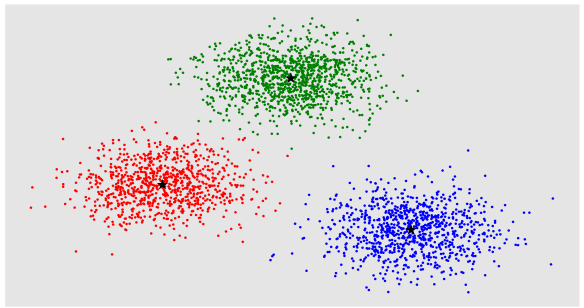
??



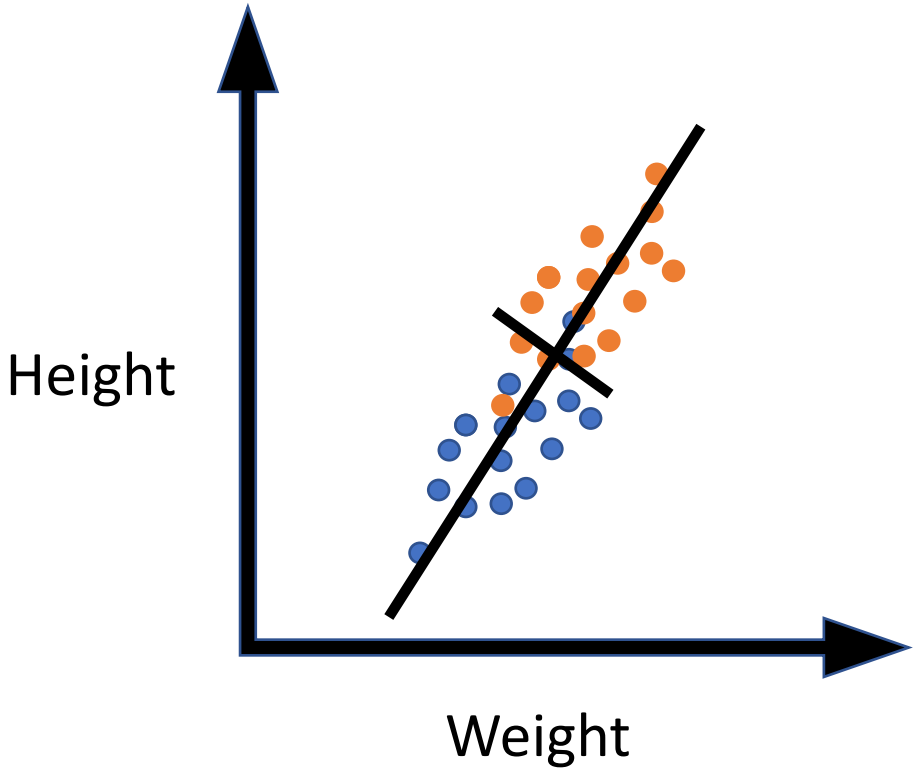
Unsupervised Learning

Cluster Analysis





Dimensionality Reduction



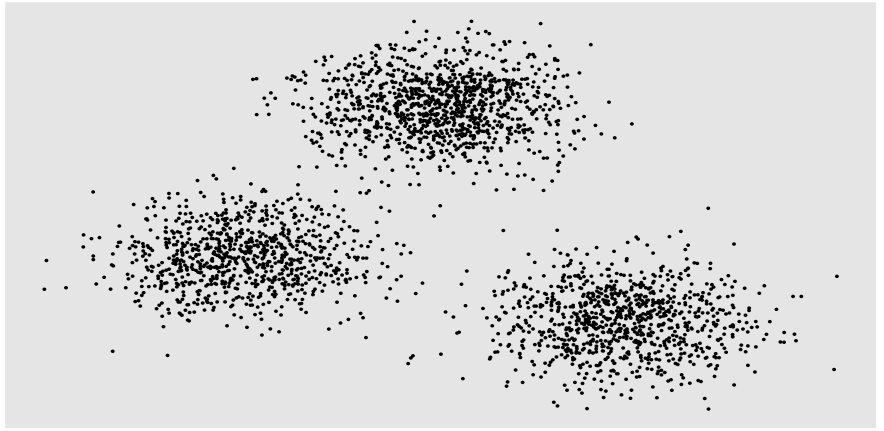
Cluster Analysis

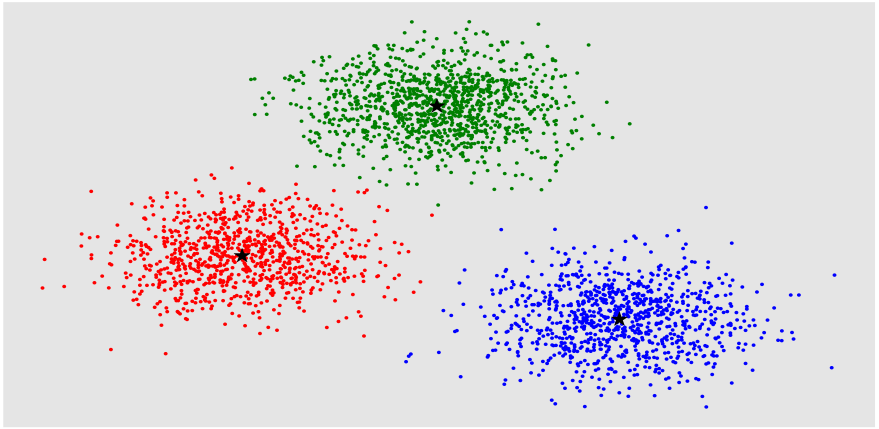
K-means clustering

- Assign data to K clusters
- Minimize within cluster variance $W(C_k)$. We want groups to be homogeneous.
- Clusters tend to be equal sizes
- Other algorithms (E.g., EM - algorithm) perform better for unequal cluster size

$$\min_{C_1,\ldots,C_K} \{\sum_{k=1}^{K} W(C_k)\}$$

$$W(C_k) = \frac{1}{N} (\sum_{n=1}^{N} [\begin{matrix} Euclidean\ distance\ of \\ \boldsymbol{n}\ to\ closest\ center \end{matrix}])$$





Cluster Analysis

K-means clustering

- Assign data to K clusters
- Minimize within cluster variance $W(C_k)$. We want groups to be homogeneous.
- Clusters tend to be equal sizes
- Other algorithms (E.g., EM - algorithm) perform better for unequal cluster size

$$\min_{C_1,\ldots,C_K} \{\sum_{k=1}^{K} W(C_k)\}$$

$$W(C_k) = \frac{1}{N} (\sum_{n=1}^{N} [\begin{matrix} Euclidean\ distance\ of \\ \boldsymbol{n}\ to\ closest\ center \end{matrix}])$$

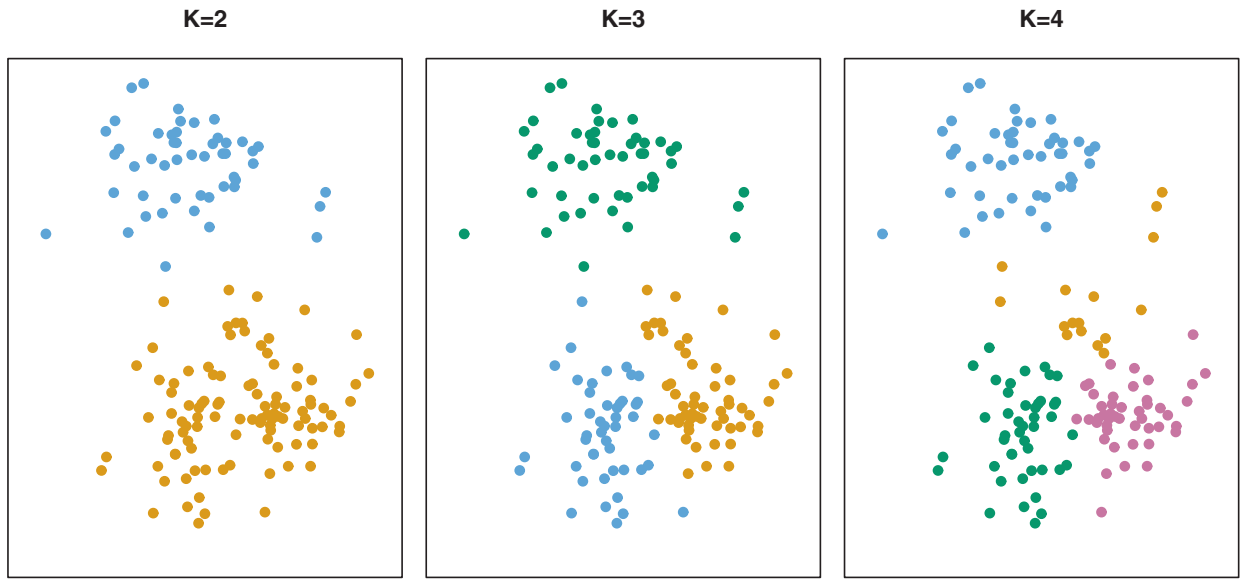


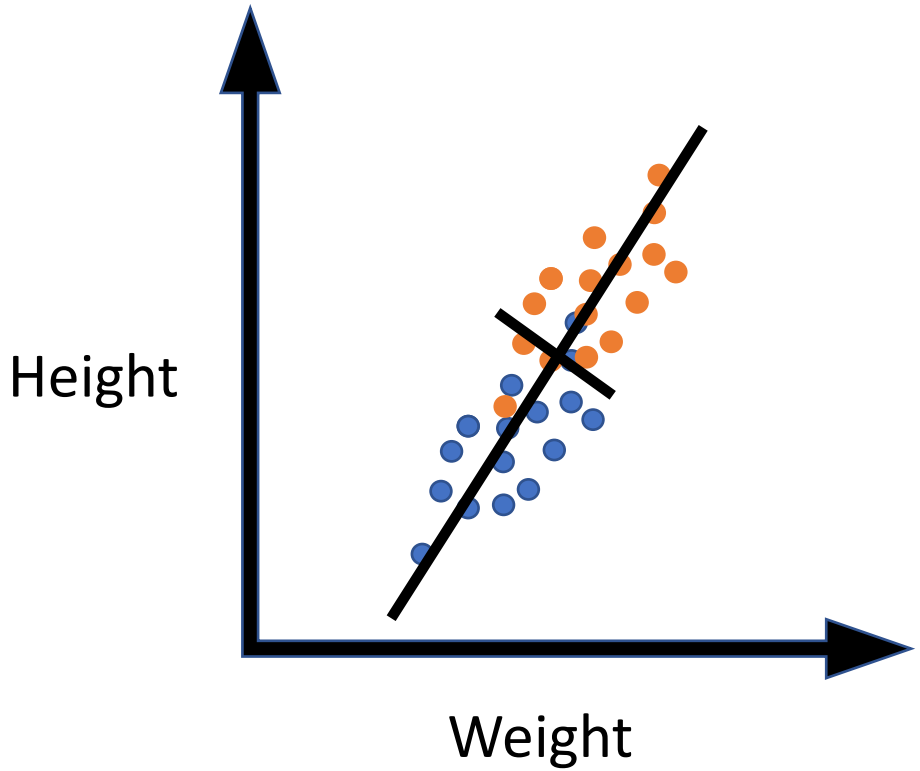
FIGURE 10.5. *A simulated data set with 150 observations in two-dimensional space. Panels show the results of applying K-means clustering with different values of K, the number of clusters. The color of each observation indicates the cluster to which it was assigned using the K-means clustering algorithm. Note that there is no ordering of the clusters, so the cluster coloring is arbitrary. These cluster labels were not used in clustering; instead, they are the outputs of the clustering procedure.*

The Elements of Statistical Learning (Hastie, Tibshirani, & Friedman)

Dimensionality Reduction

Principal Component Analysis

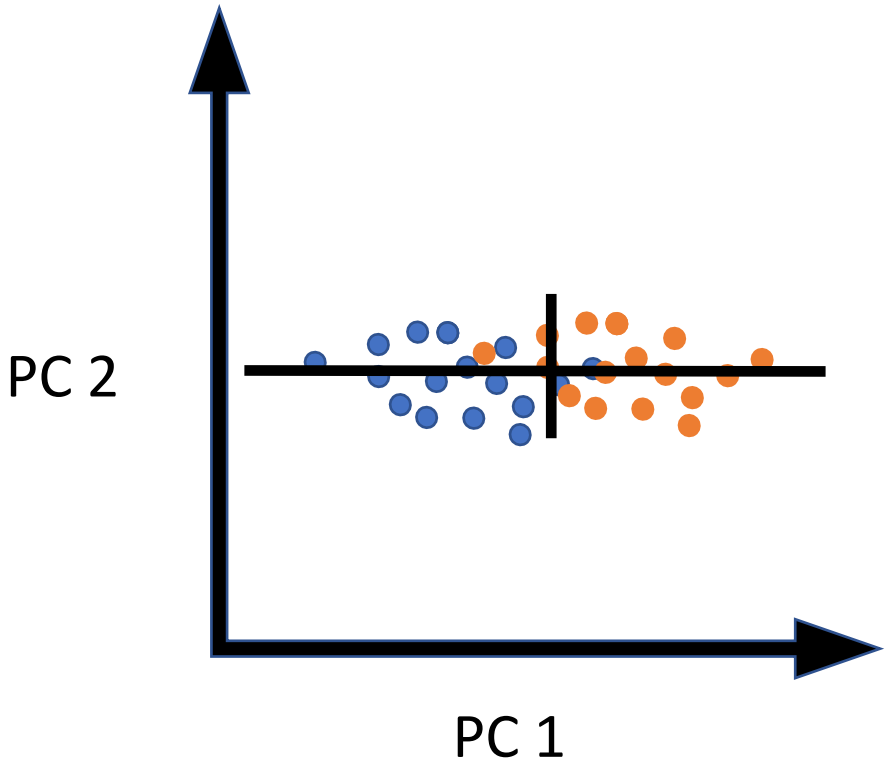
- PCA reduces dataset to linear combination of vectors called Principal Components (PCs)
- PCs are orthogonal to one another
- PCs ordered based on variance they explain in dataset
- Small subset of PCs often explains majority of variance
- PCs can be used to perform additional statistical tests



Dimensionality Reduction

Principal Component Analysis

- PCA reduces dataset to linear combination of vectors called Principal Components (PCs)
- PCs are orthogonal to one another
- PCs ordered based on variance they explain in dataset
- Small subset of PCs often explains majority of variance
- PCs can be used to perform additional statistical tests



Dimensionality Reduction

Principal Component Analysis

- PCA reduces dataset to linear combination of vectors called Principal Components (PCs)
- PCs are orthogonal to one another
- PCs ordered based on variance they explain in dataset
- Small subset of PCs often explains majority of variance
- PCs can be used to perform additional statistical tests

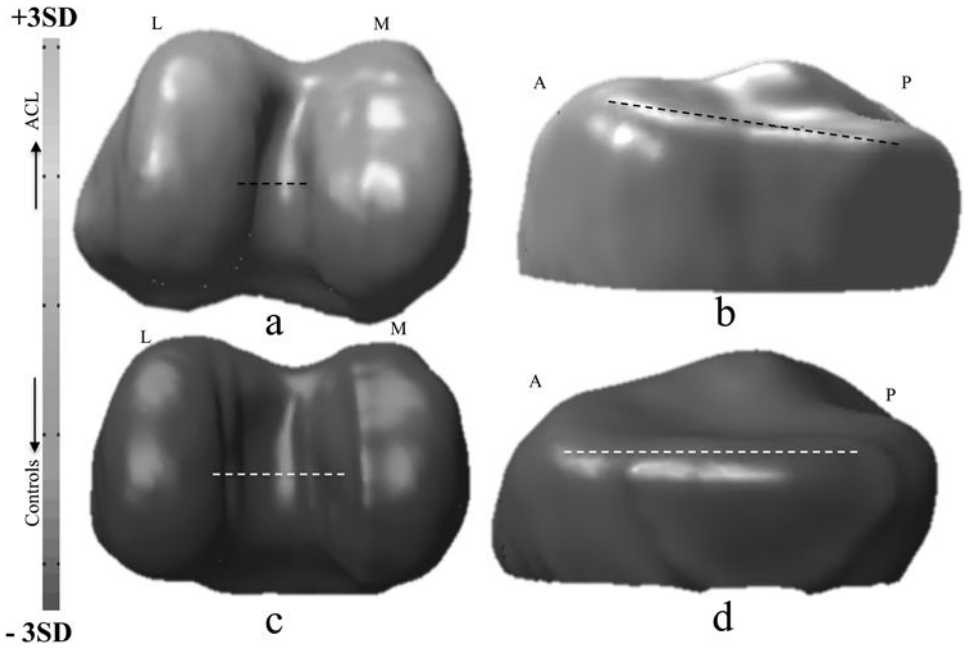


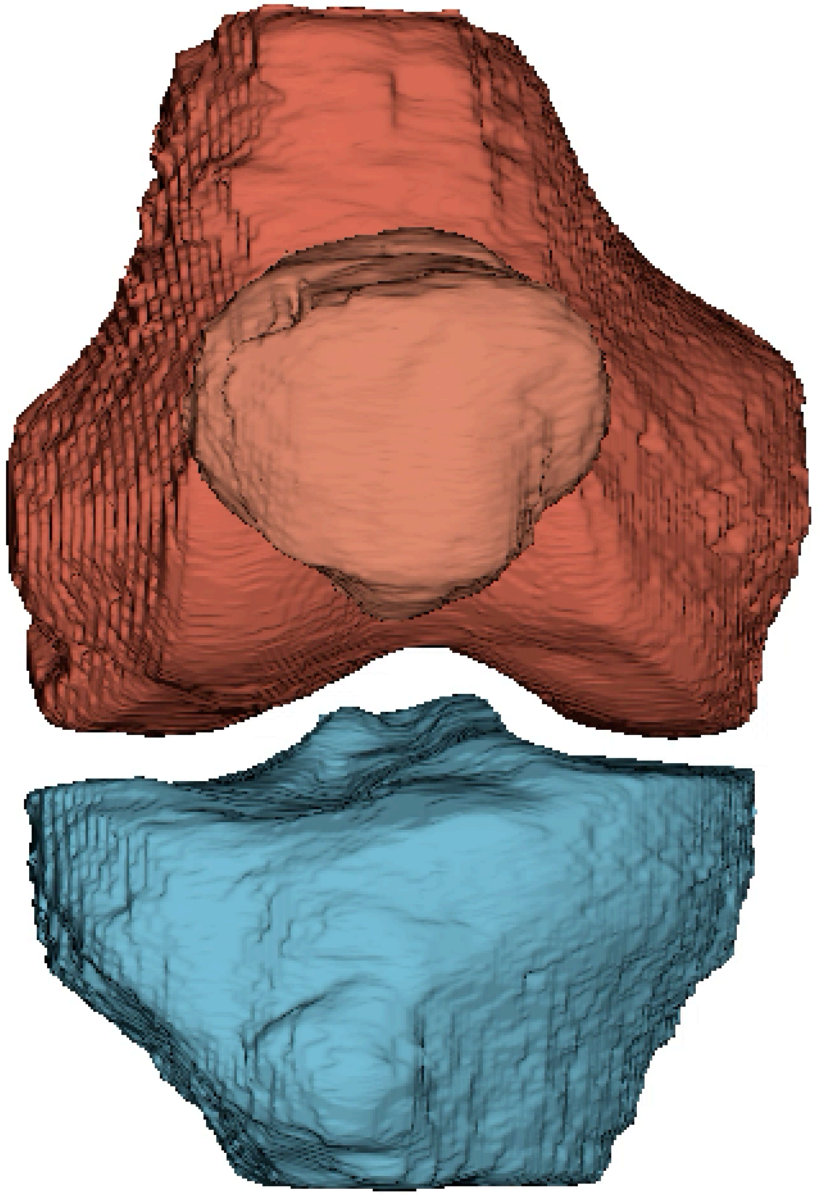
Figure 5.
modeling of the principal component F2 (a,c) and T3 (b,d) of the scale-preserved model. In the first row Mean + 3STD. in the second row Mean – 3STD. Directions are labeled in the figure. M: medial, L: lateral, A: anterior P: posterior

Pedoia et al. *Three Dimensional MRI-Based Statistical Shape Model and Application to a Cohort of Knees with Acute ACL Injury.* Osteoarthritis & Cartilage (2015)

Dimensionality Reduction

Principal Component Analysis

- PCA reduces dataset to linear combination of vectors called Principal Components (PCs)
- PCs are orthogonal to one another
- PCs ordered based on variance they explain in dataset
- Small subset of PCs often explains majority of variance
- PCs can be used to perform additional statistical tests



Dimensionality Reduction

Principal Component Analysis

- PCA reduces dataset to linear combination of vectors called Principal Components (PCs)
- PCs are orthogonal to one another
- PCs ordered based on variance they explain in dataset
- Small subset of PCs often explains majority of variance
- PCs can be used to perform additional statistical tests

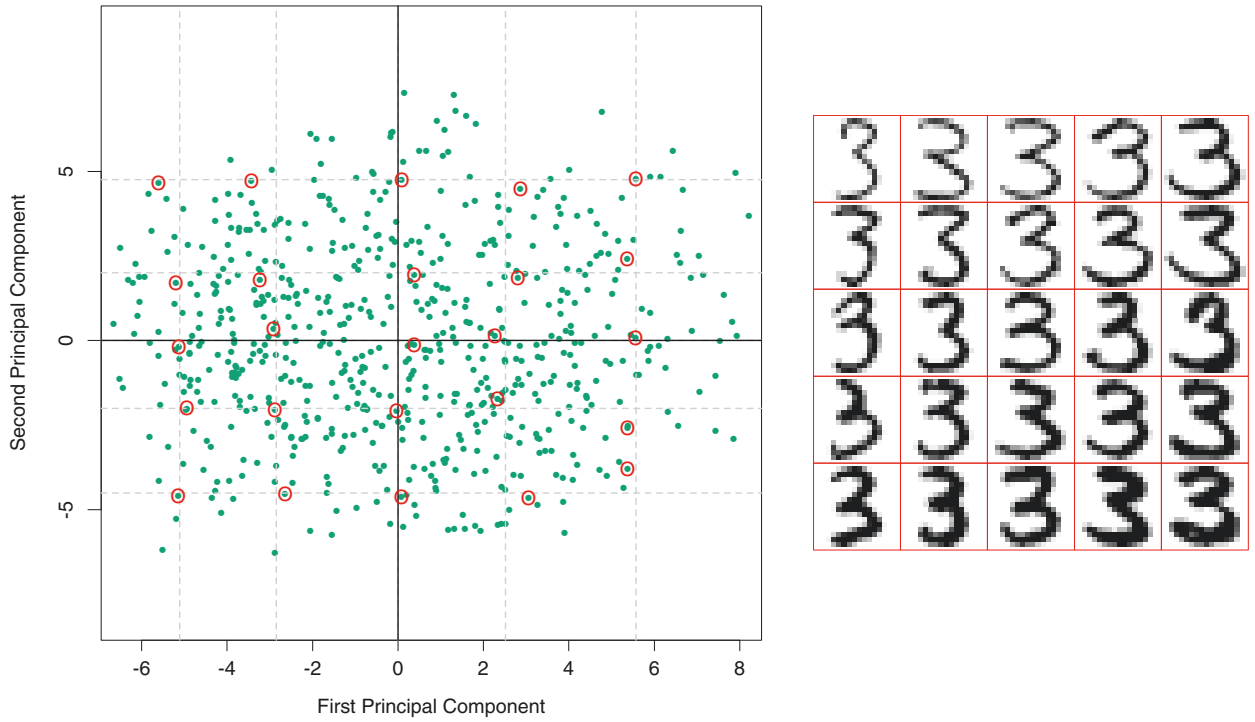
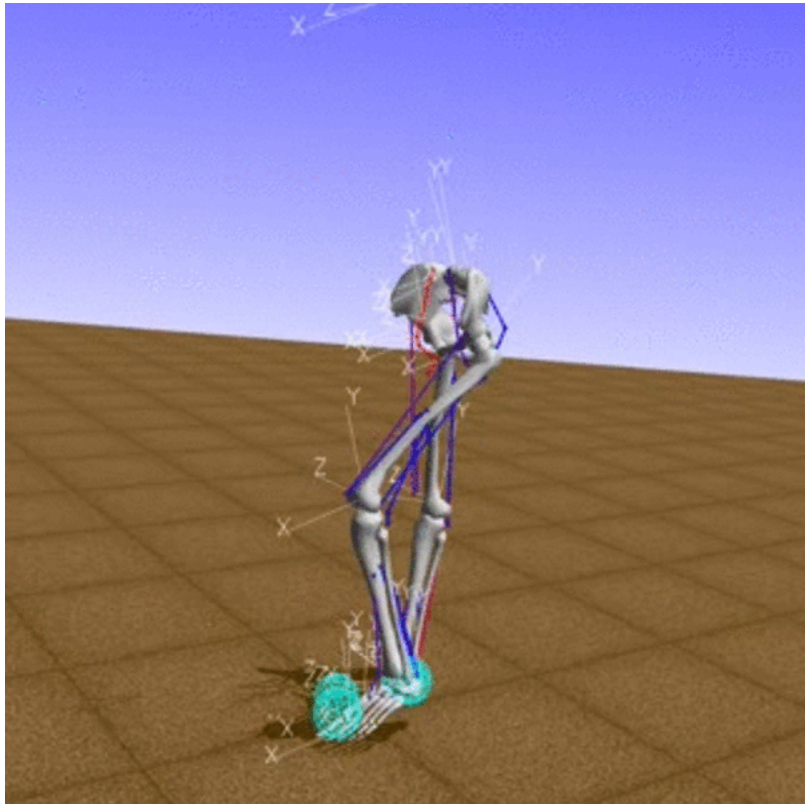


FIGURE 14.23. *(Left panel:) the first two principal components of the hand-written threes. The circled points are the closest projected images to the vertices of a grid, defined by the marginal quantiles of the principal components. (Right panel:) The images corresponding to the circled points. These show the nature of the first two principal components.*

Reinforcement Learning



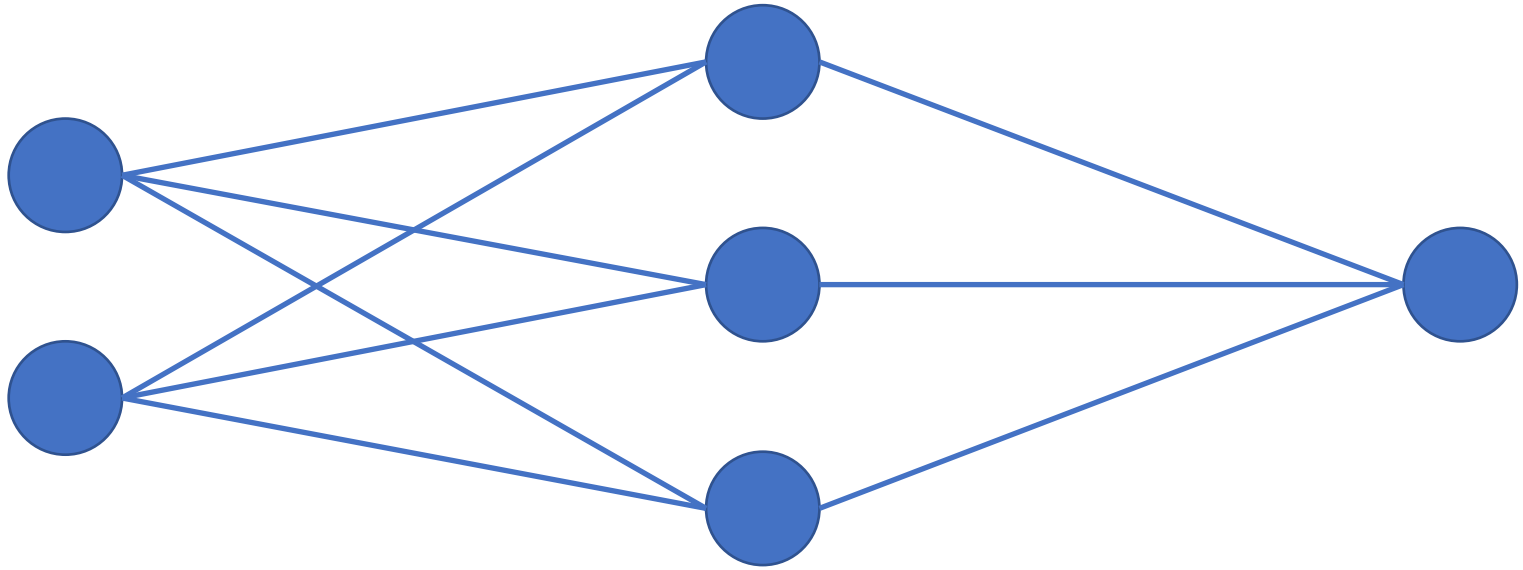
Reinforcement Learning

Reinforcement Learning

Neural Networks

- Can be used for all of the problems we've talked about
- Just becomes a task of determining how to:
  - Define inputs
  - Define the network architecture
  - Define outputs
  - Define the error (network learns from) to achieve the outcome of interest

Neural Networks

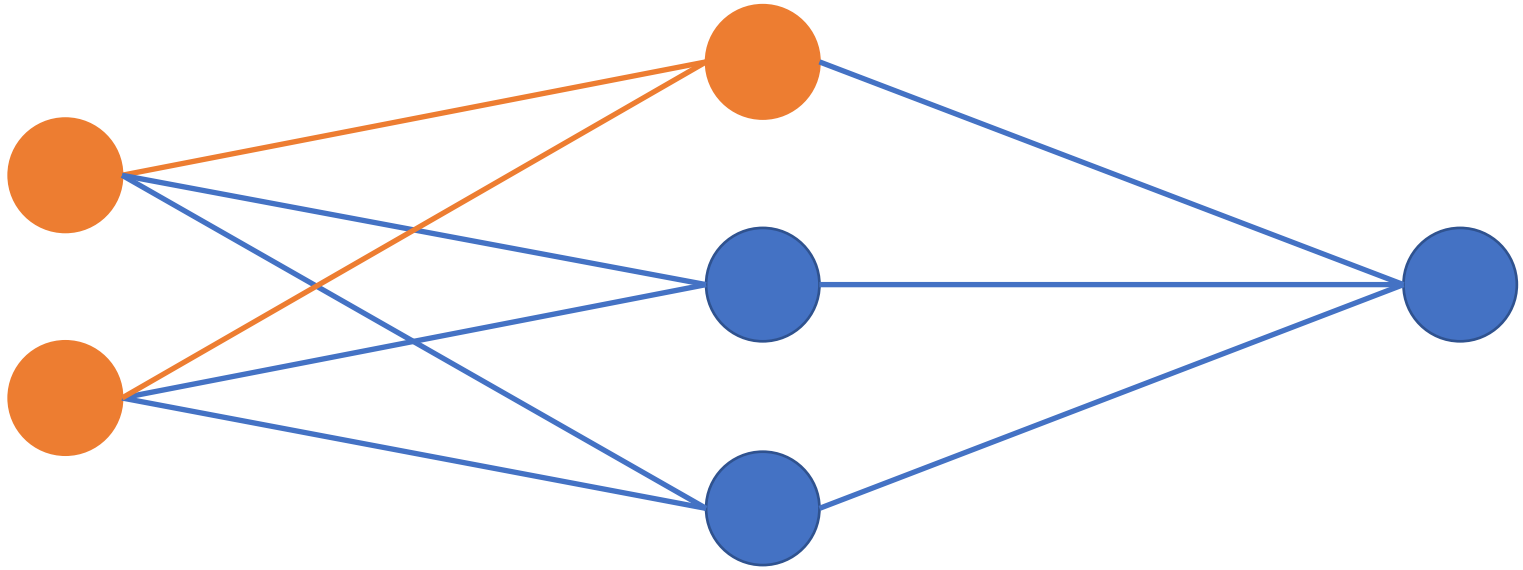


Inputs

Hidden Layer

Output

Neural Networks

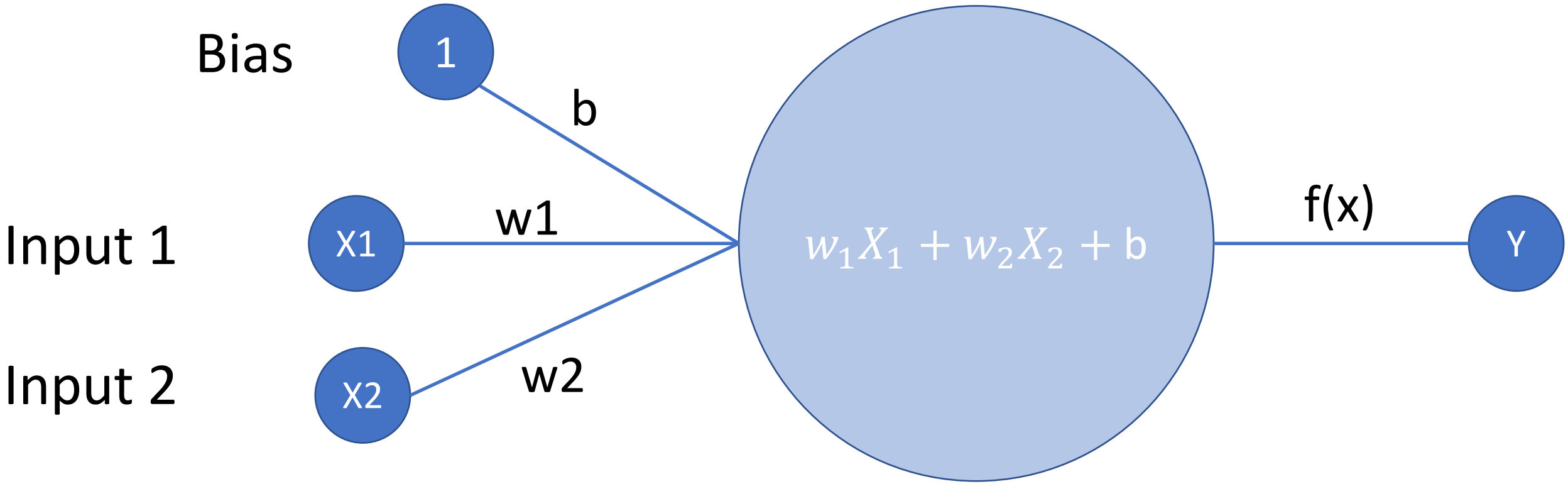


Inputs

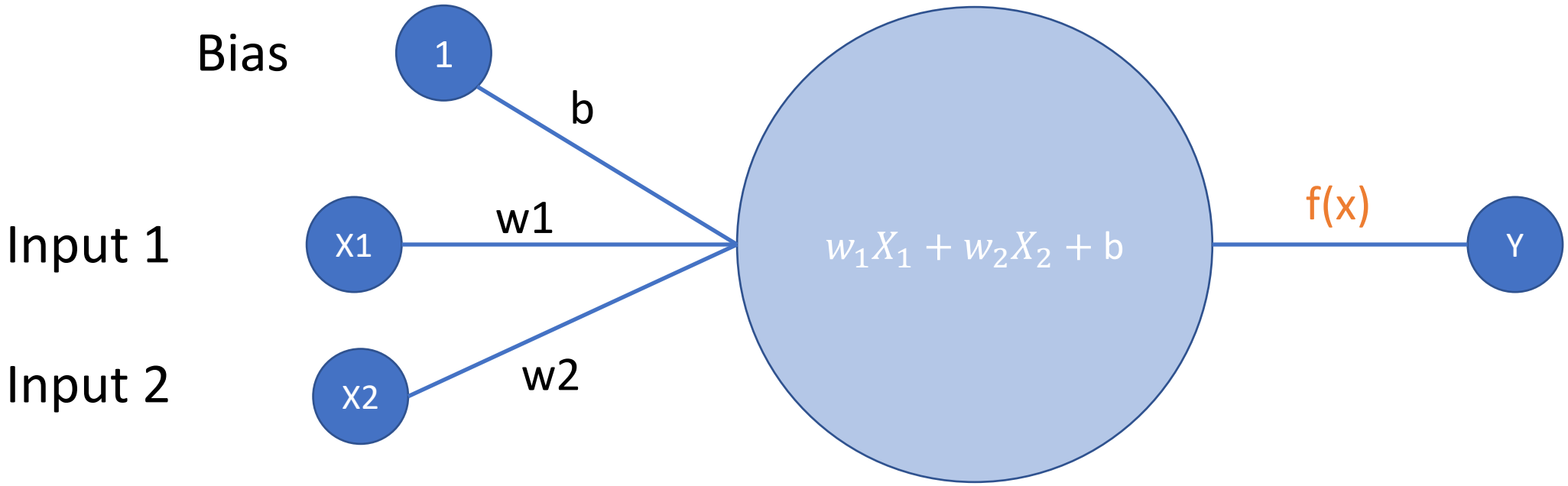
Hidden Layer

Output

Neural Networks



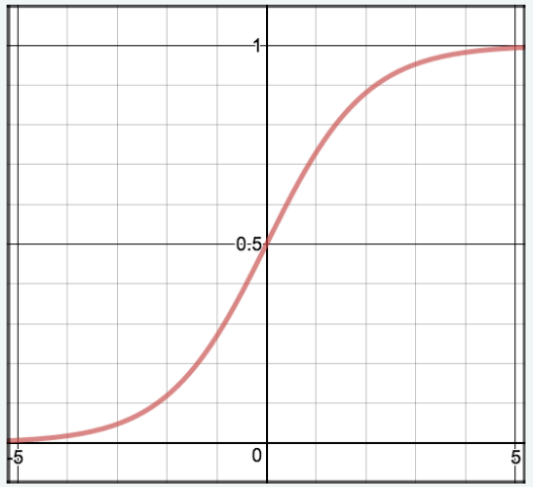
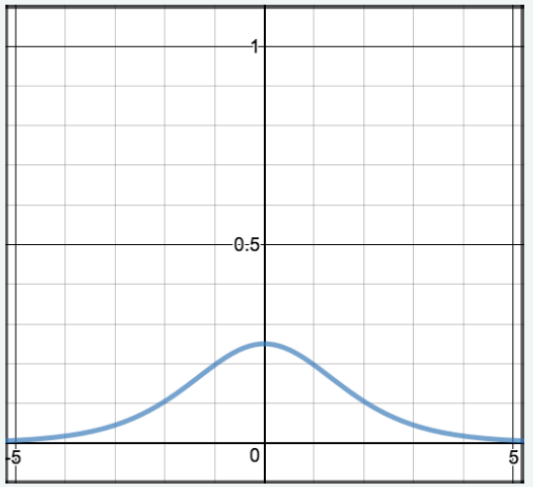
Neural Networks



$$f(x) = Activation\ Function$$

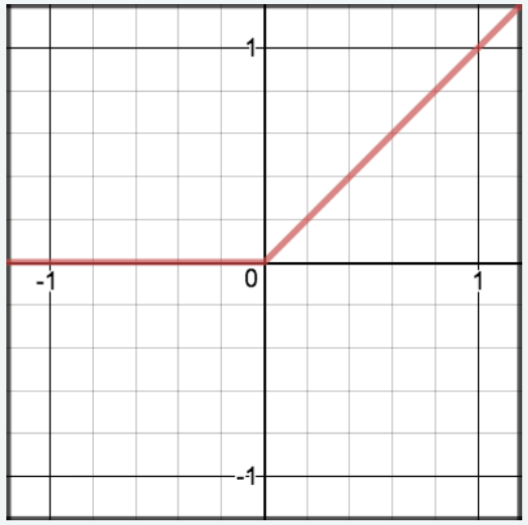
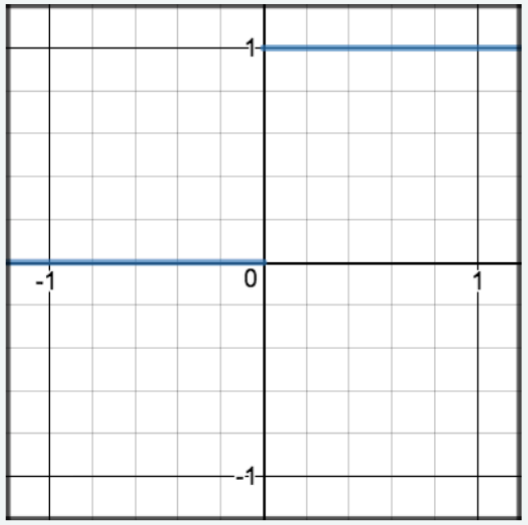
Neural Networks

$$f(x) = Activation\ Function$$

Function	Derivative
$S(z) = \frac{1}{1+e^{-z}}$	$S'(z) = S(z) \cdot (1 - S(z))$
	

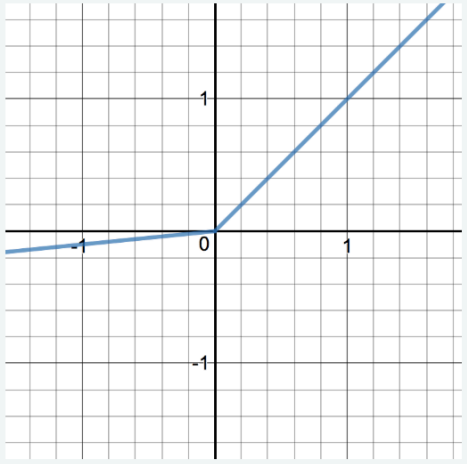
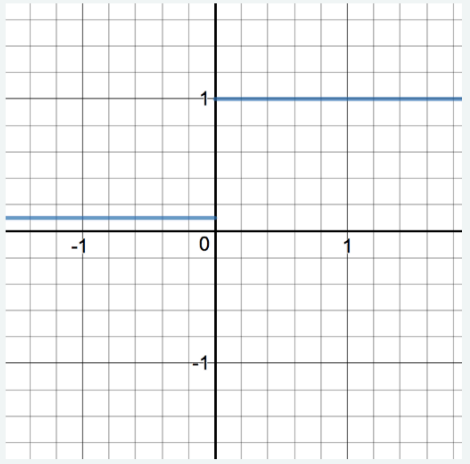
Neural Networks

$$f(x) = Activation\ Function$$

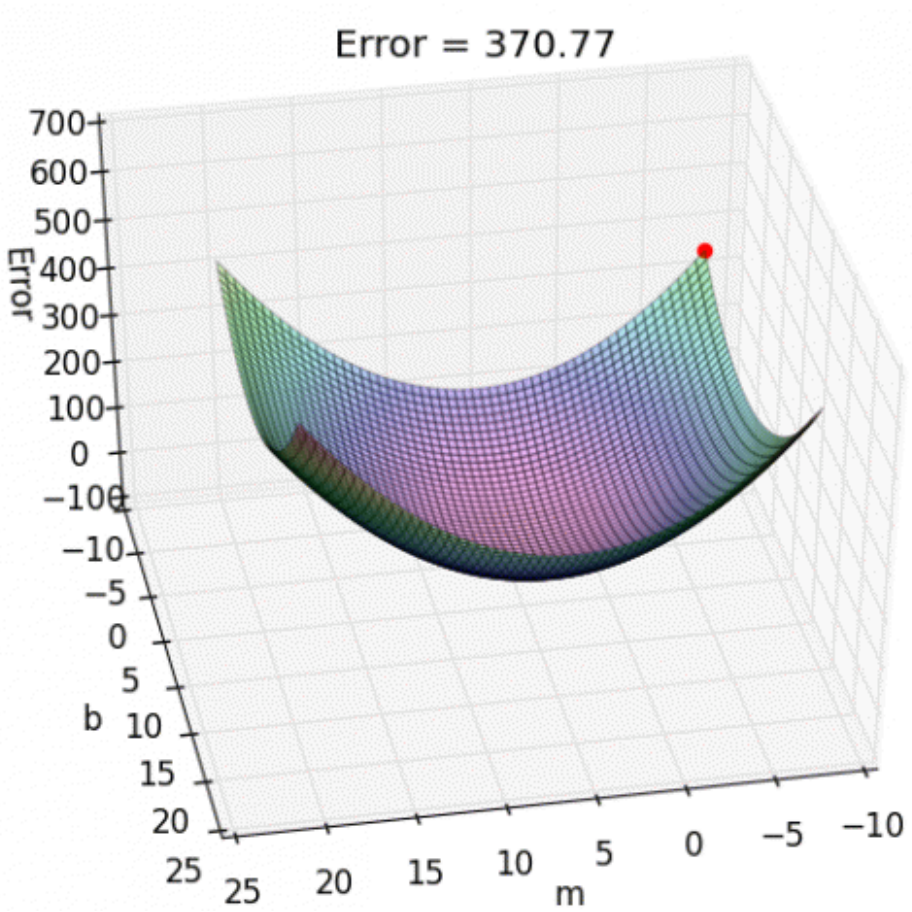
Function	Derivative
$R(z) = \begin{cases} z & z > 0 \\ 0 & z <= 0 \end{cases}$	$R'(z) = \begin{cases} 1 & z > 0 \\ 0 & z < 0 \end{cases}$
	

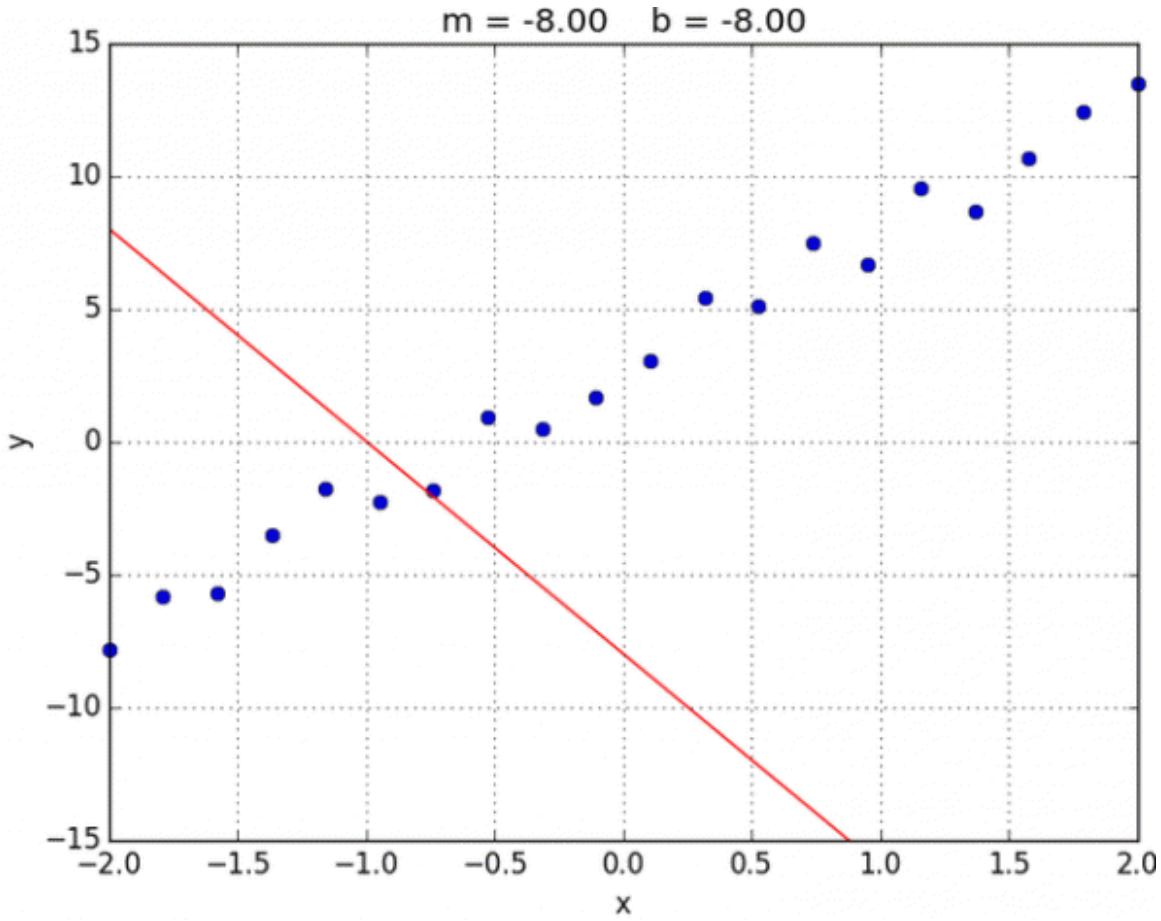
Neural Networks

$$f(x) = Activation\ Function$$

Function	Derivative
$R(z) = \begin{cases} z & z > 0 \\ \alpha z & z <= 0 \end{cases}$	$R'(z) = \begin{cases} 1 & z > 0 \\ \alpha & z < 0 \end{cases}$
	

Neural Networks – Gradient descent





Neural Networks – Gradient descent

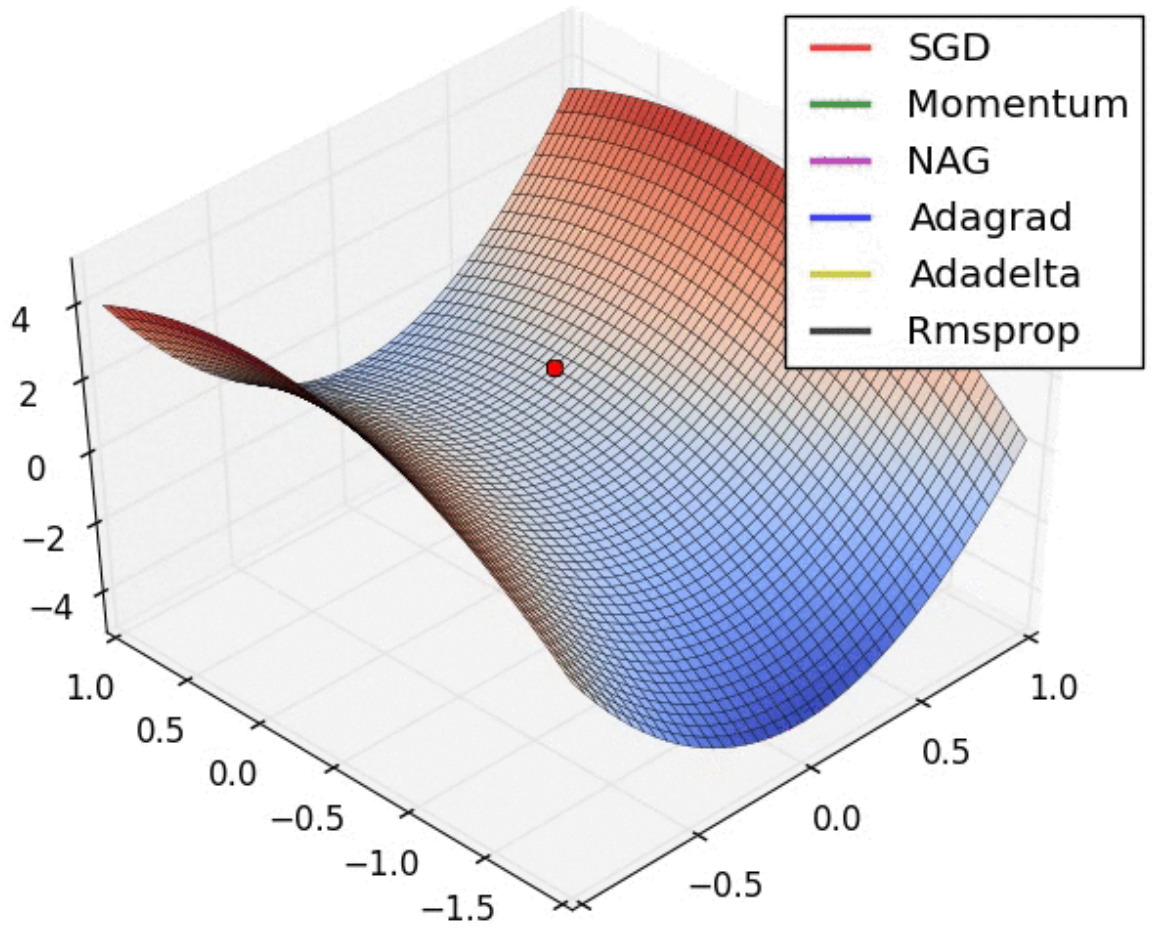


Image Segmentation

Femoral Cartilage

Medial Tibial Cartilage

Lateral Tibial Cartilage

Patellar Cartilage

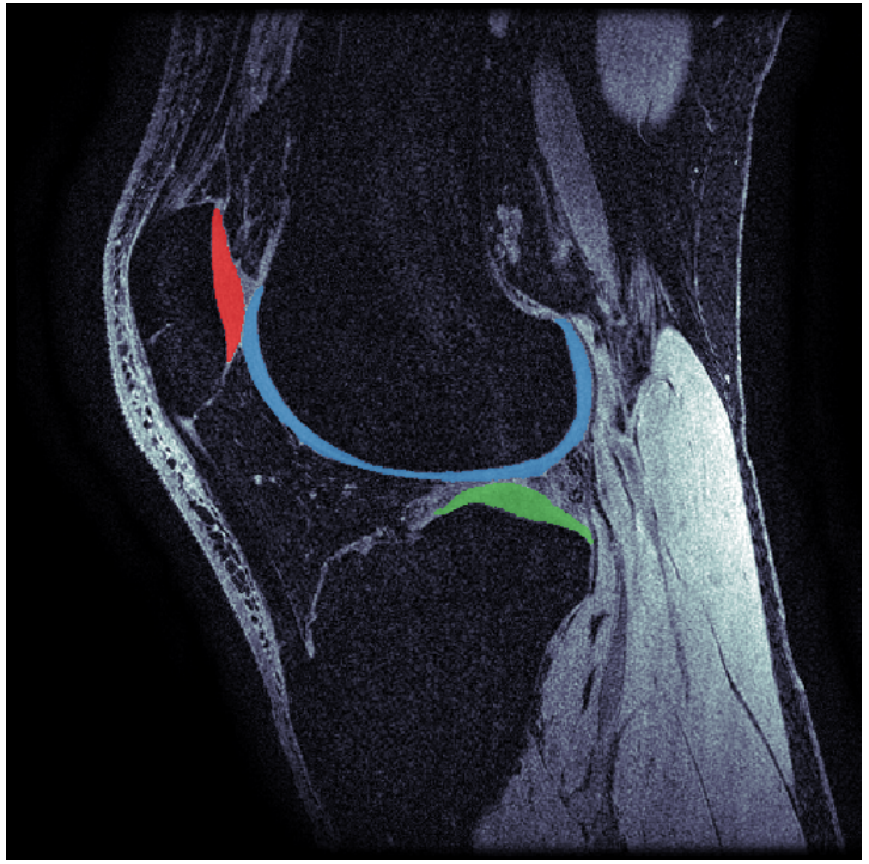
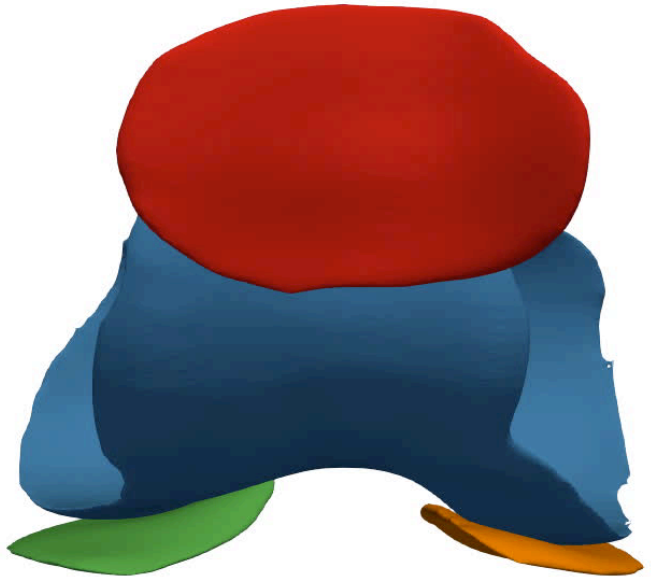


Image Segmentation



Femoral Cartilage

Medial Tibial Cartilage

Lateral Tibial Cartilage

Patellar Cartilage

Image Segmentation

Manual

- Gold standard [1]
- Human error [2]
- Time consuming [3]

Machine Learning

- Classify pixels using "features"[4,5]
- Good features hard to identify
- ***Deep Learning*** used to overcome feature identification[6,7,8, 9]

[1] Pedoia et al. *Magn Reson Mater Phy* (2016); [2] Schneider et al. *OA&C* (2012); [3] Shim et al. *Radiology* (2009); [4] Shan et al. *Medical Image Analysis* (2014); [5] Folkesson et al. *IEEE Trans Biomed Eng* (2007); [6] Liu et al. *Magn. Reson. Med.*(2017); [7] Norman et al. *Radiology* (2018); [8] Tack et al. *OA&C* (2018); [9] Prasoon et al. *MICCAI* (2013).