STAT 4609: Big Data Analysis

Lecture 1: Introduction to Machine Learning

香 港 大 學

THE UNIVERSITY OF HONG KONG

Course Details

- Lecturer: Michael Zhang, Tutor: Chen Liu
- Tutorial Sessions: Tuesdays, 5:30 p.m. – 6:20 p.m.
- All courses will be held on Zoom.

Course Details

- Each lecture typically consists of three 50 minute courses, with two 10 minute breaks.
- All lectures will be recorded. Students are not required to turn cameras on.
- Any questions: Please type in chat or ask during break.

Course Details

- This course is assessed 100% through coursework.
- Students should have prior knowledge of Python.
  1. **Homework (40%)** There will be five assignments (done individually).
  2. **Group Project (60%)** Students should work on the project in teams of three to four students. Students consider a real problem involving class topics on a data set.
- There will not be any exams. There are no required textbooks.

Course Details

- Main goal: Broad survey of popular machine learning methods.
- You should:
  - Get basic familiarity of different ML models.
  - Learn how to implement basic models in Python.
  - Learn how to apply models in real settings.

Intro to Machine Learning

- What is machine learning?
- Broadly speaking: To be able to learn patterns from data.
- Big data: The scale of the data is massive, in N and D.
- Interchangeable with A.I., data analysis, data mining, data science, etc.
- Tasks: Learning, prediction, description, decision making.

Examples of Machine Learning: Digit Classification

- Suppose you have seen handwritten digits. Can you classify an image as a “zero” to “nine”?

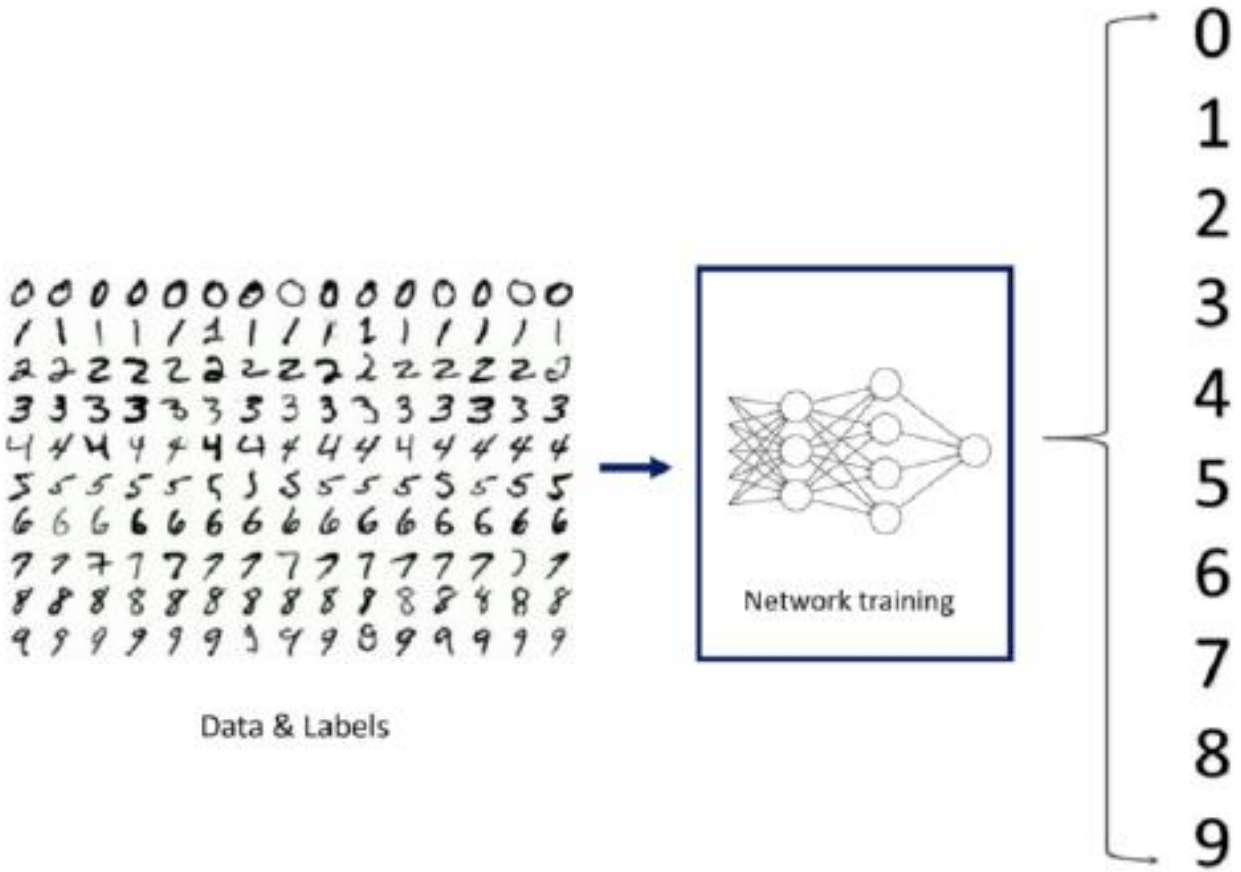


Figure: Image from “Towards Data Science”.

Examples of Machine Learning: Netflix Problem

- Netflix users rate a subset of movies. Given the movies I like, can you recommend me a new movie?

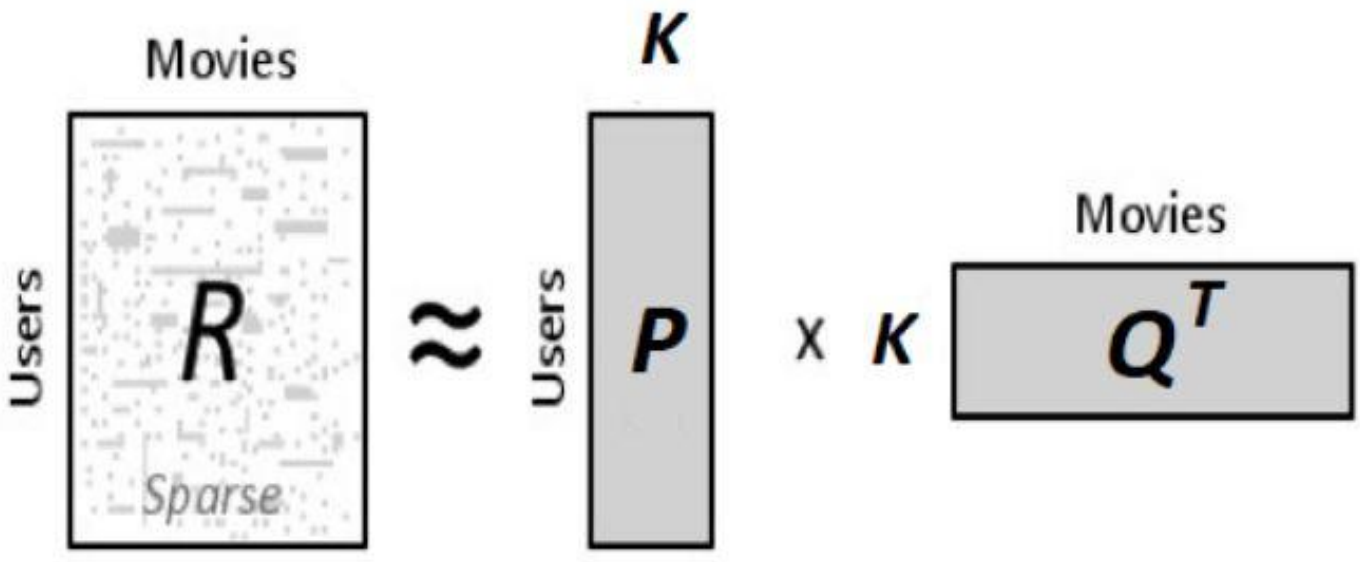


Figure: Image from Kaggle.

Examples of Machine Learning: Topic Modeling

- Suppose you have a collection of documents.
- Can you summarize the content of the collection? Can you summarize the content of each document?

"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

Examples of Machine Learning: Dimension Reduction

- If you have high-dimensional data, can you represent the data in lower dimensions?

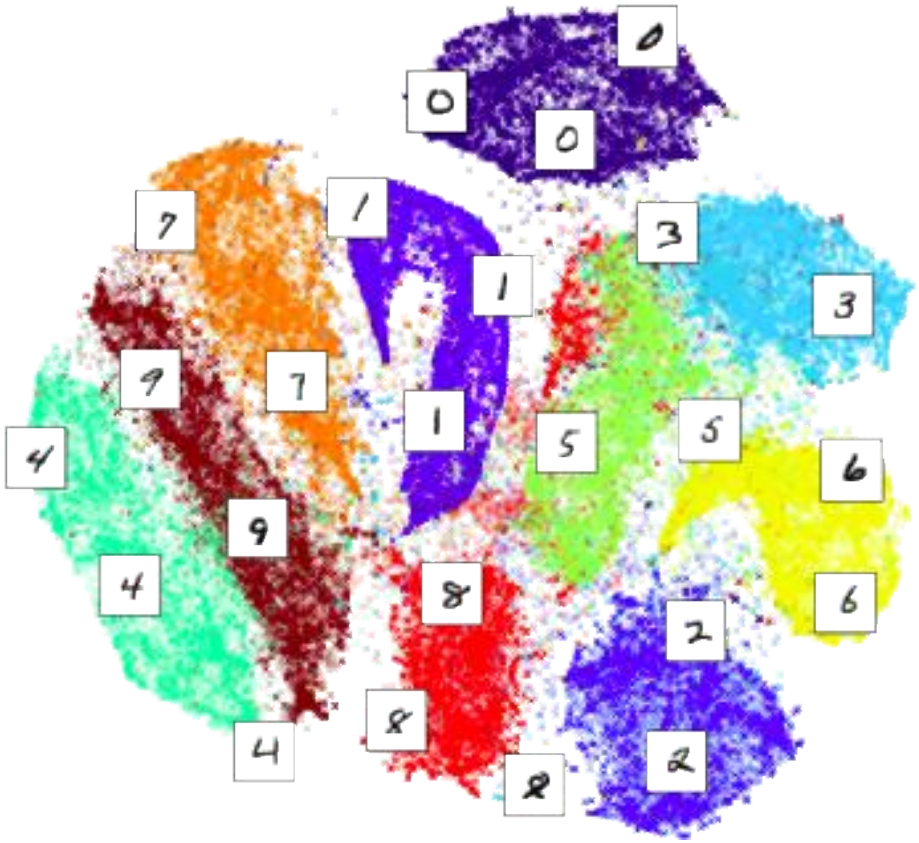


Figure: Image from `https://nlml.github.io/`.

Examples of Machine Learning: Network Analysis

- If you're Facebook, can you group users based on their friend network?

Figure: Image from Wikimedia Commons.

Examples of Machine Learning: Machine Translation

- Given a corpus in two languages, can you translate between the two?

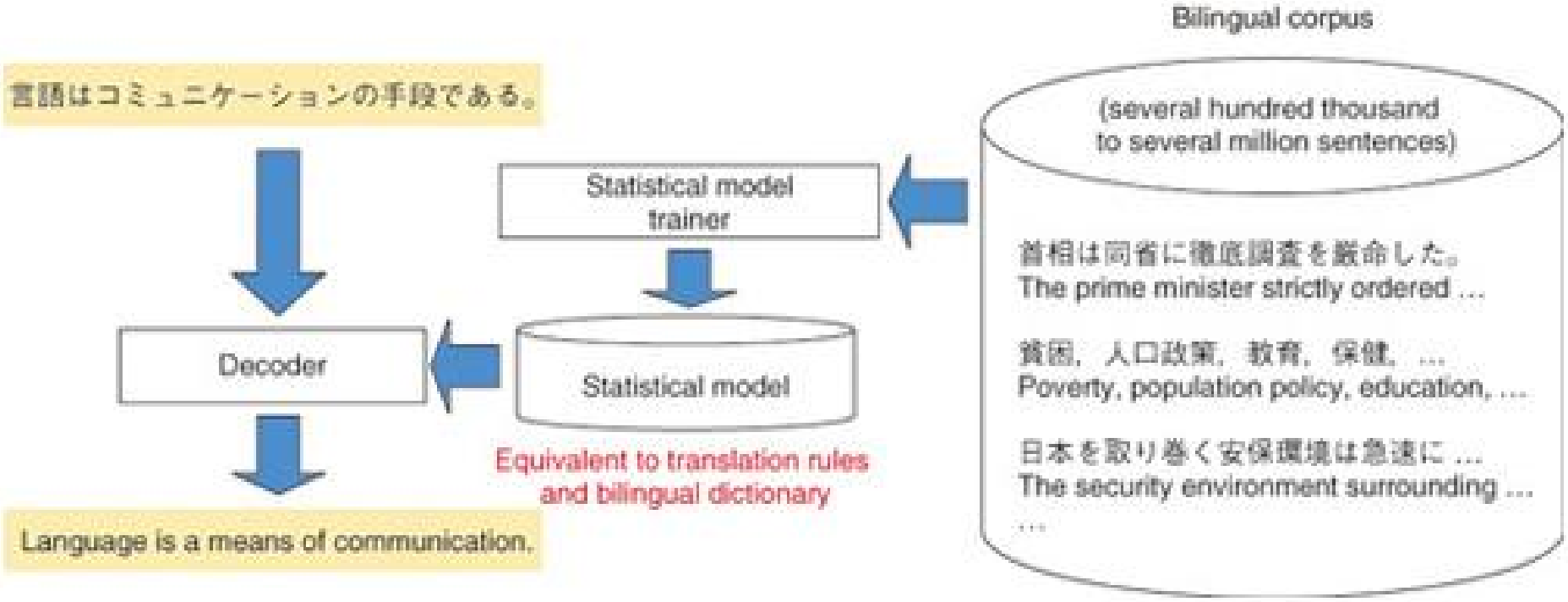
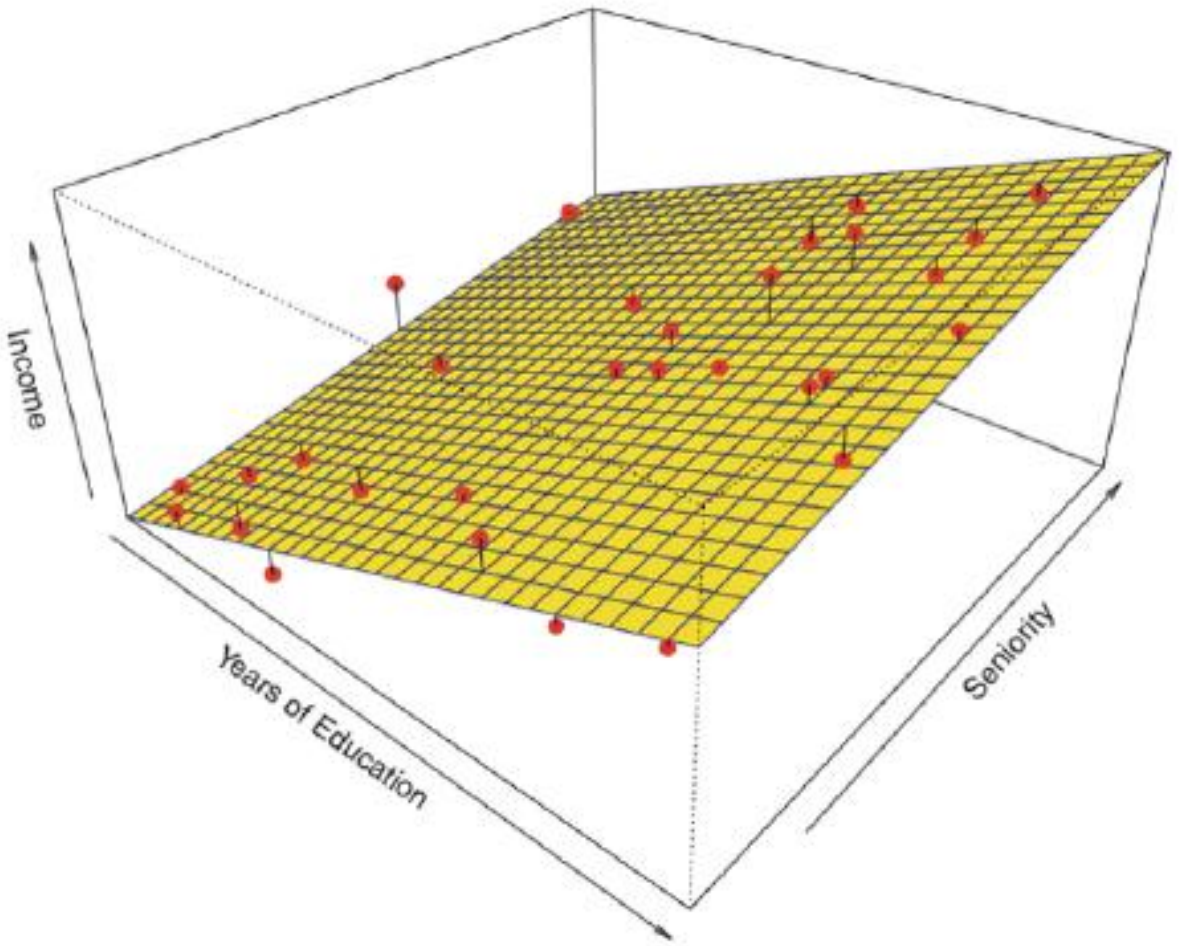


Figure: Image from NTT Technical Review.

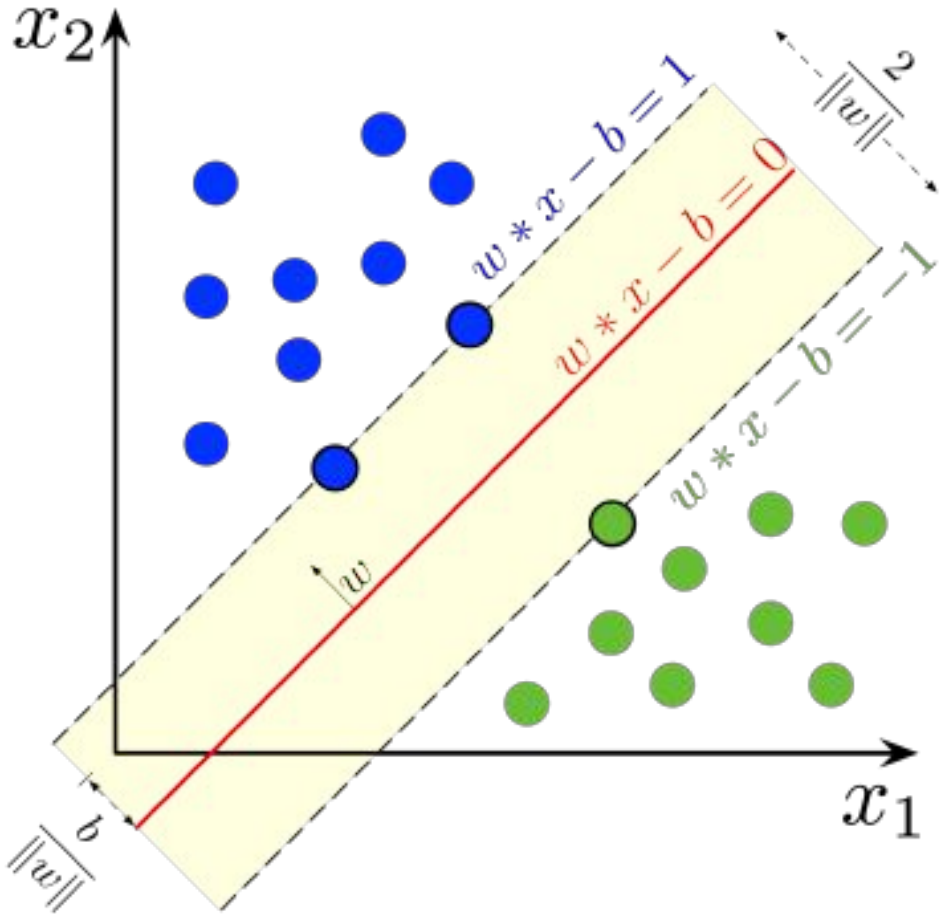
Course Overview

- What will we cover in this course? Regression:



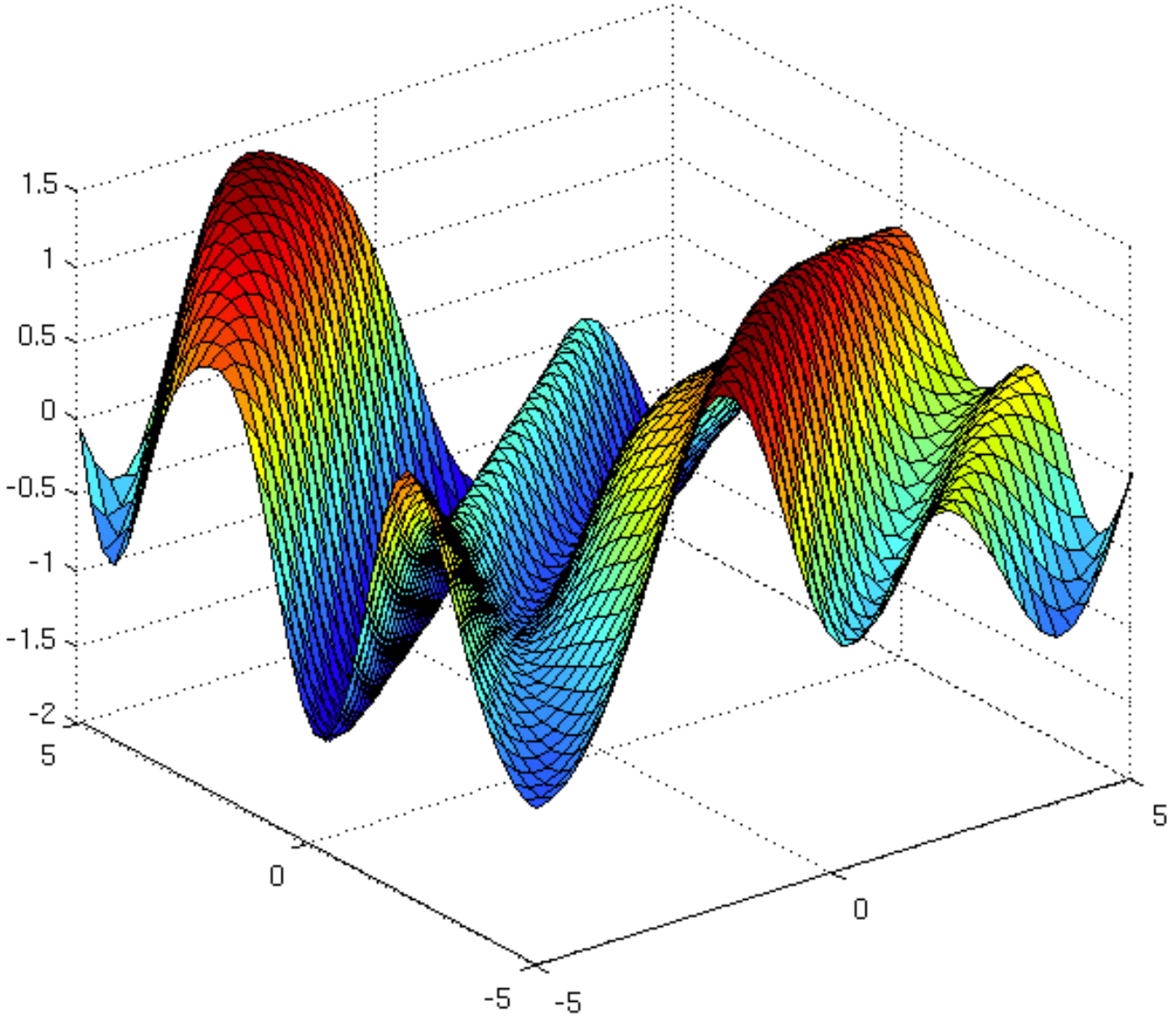
Course Overview

- What will we cover in this course? Classification:



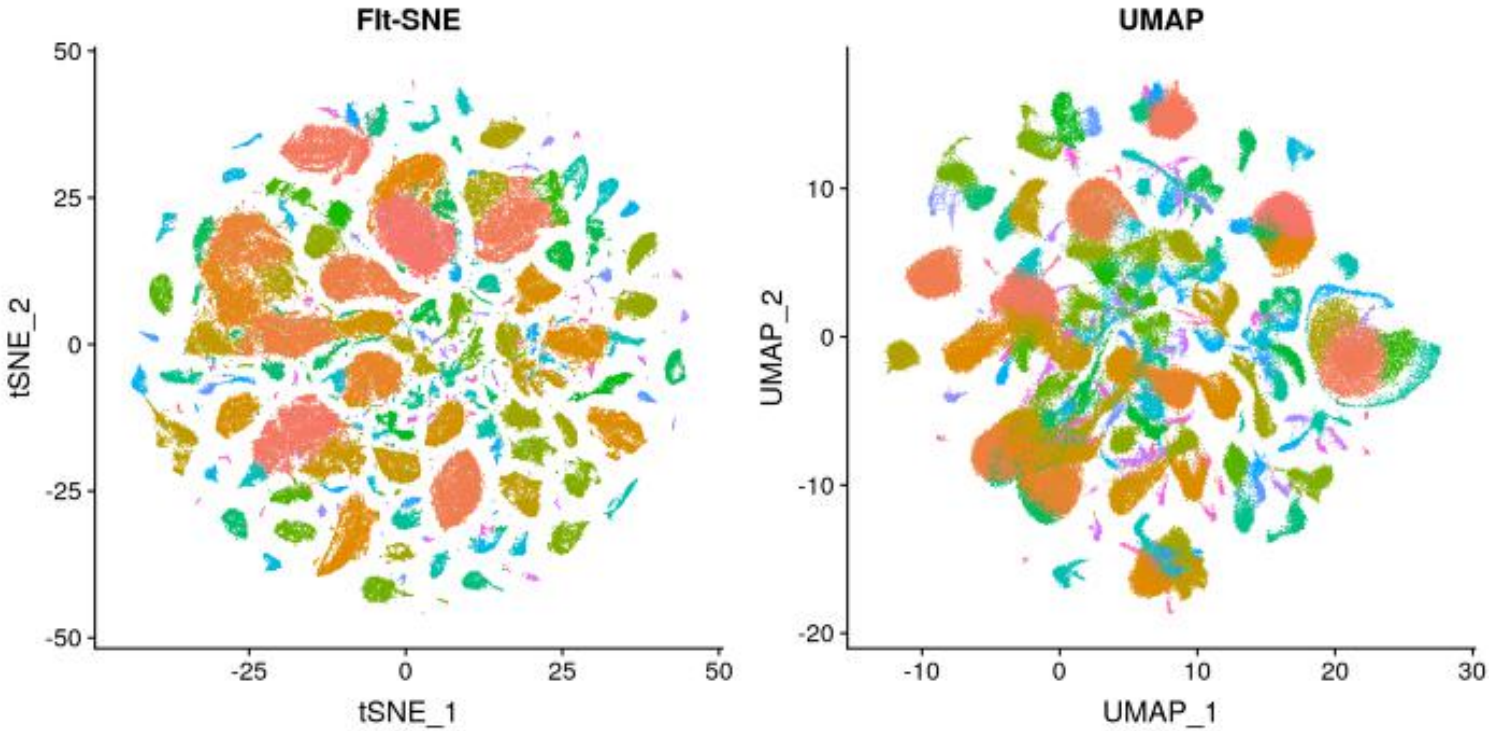
Course Overview

- What will we cover in this course? Kernel methods:



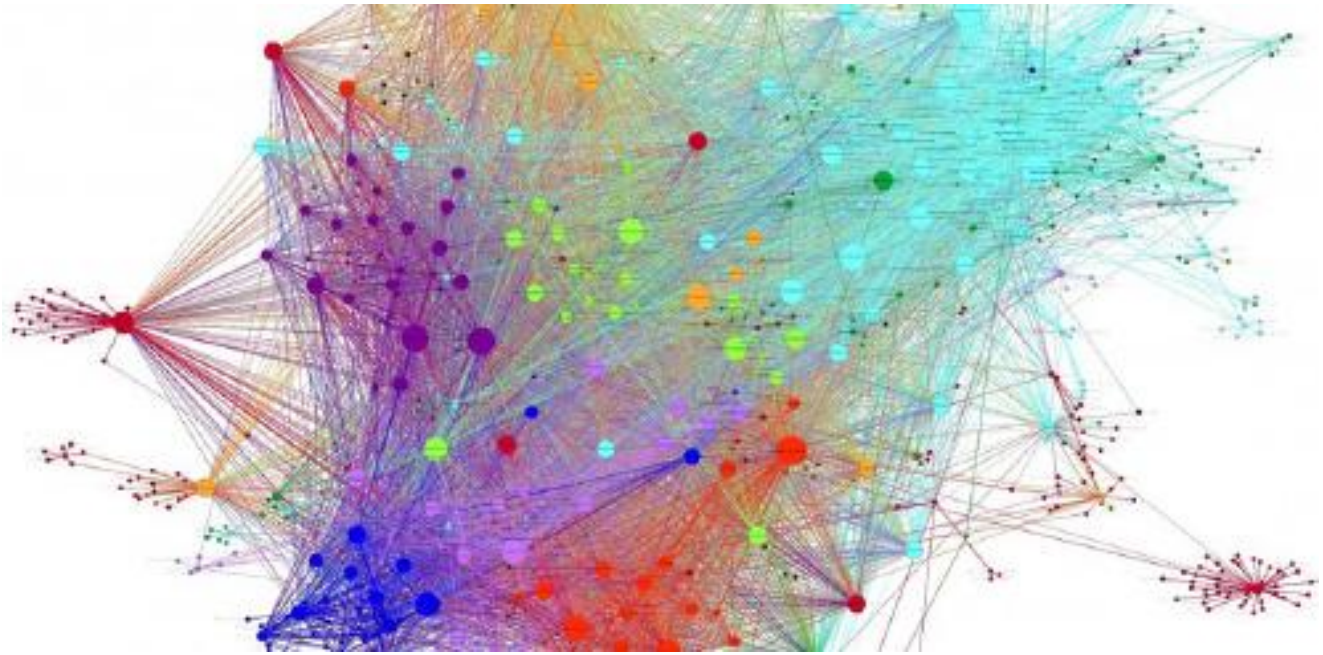
Course Overview

- What will we cover in this course? Dimensionality Reduction:



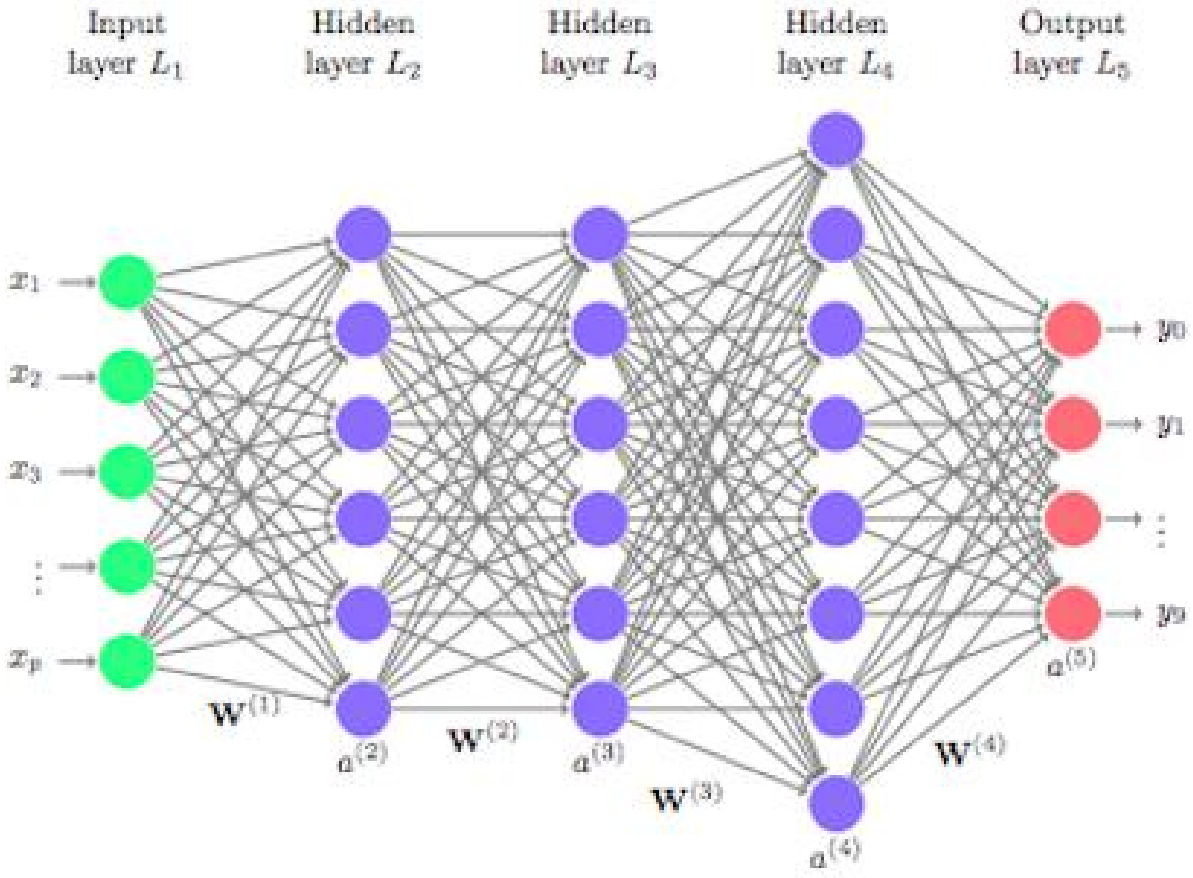
Course Overview

- What will we cover in this course? Network analysis:



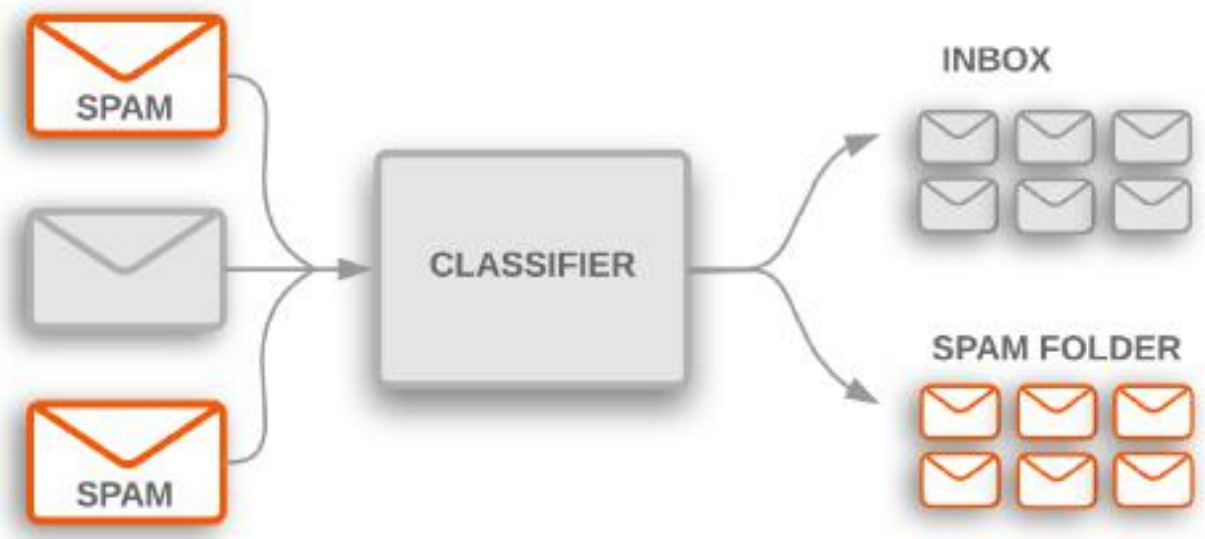
Course Overview

- What will we cover in this course? Deep Learning:



Course Overview

- What will we cover in this course? Natural Language Processing:



Course Overview

- What will we cover in this course? Topic Modeling:

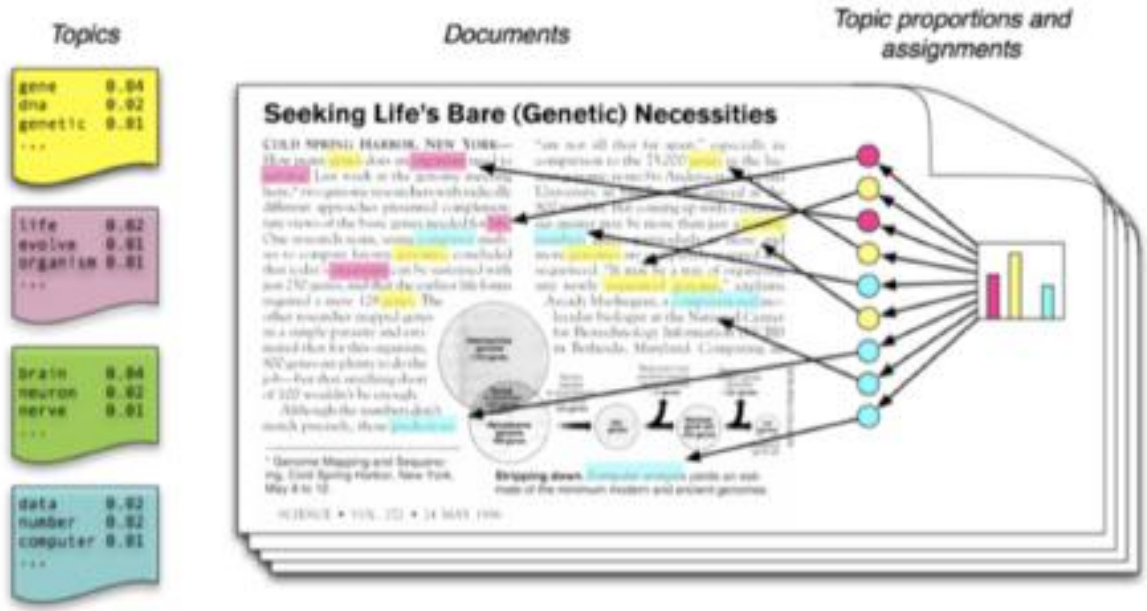


Figure source: Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, 55(4), 77-84.

Fundamentals of Machine Learning: Supervised Learning

- ML models can be categorized into two types: supervised and unsupervised learning.
- *Supervised learning*: We observe some features variables and some response variables, then learn model to map the features to the responses.

Fundamentals of Machine Learning: Supervised Learning

- Examples: Linear regression, k-nearest neighbors, support vector machines, neural networks

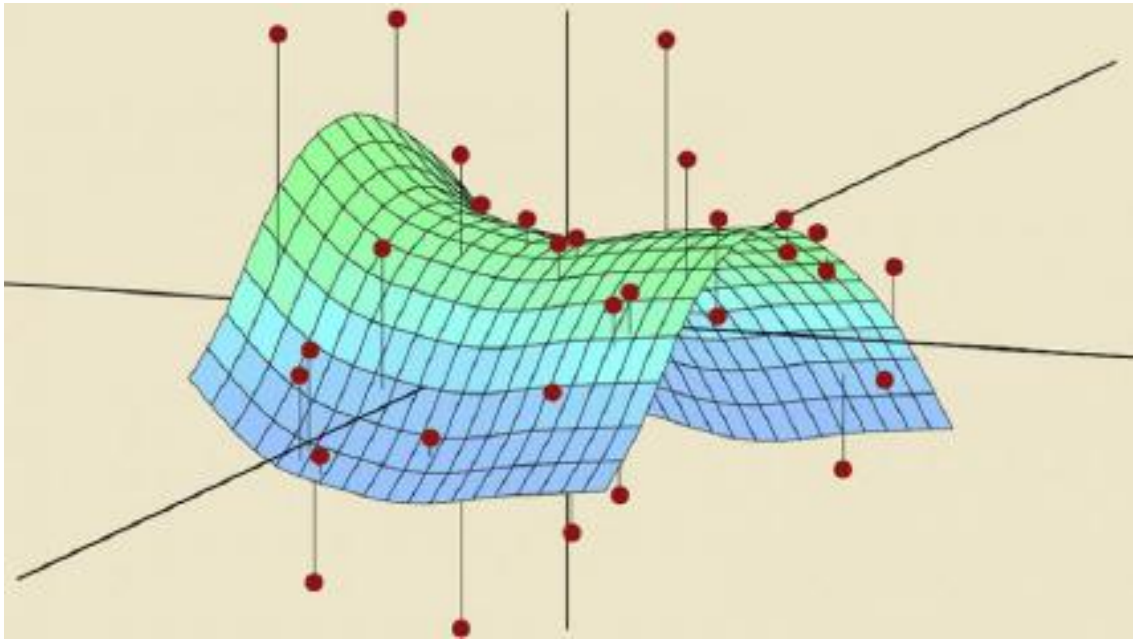


Figure: Image from "An Introduction to Statistical Learning" by James et al.

Fundamentals of Machine Learning: Unsupervised Learning

- *Unsupervised learning*: We only observe the features variables and try to learn some kind of underlying structure
- Examples: k-means clustering, t-SNE, principal components analysis, topic models

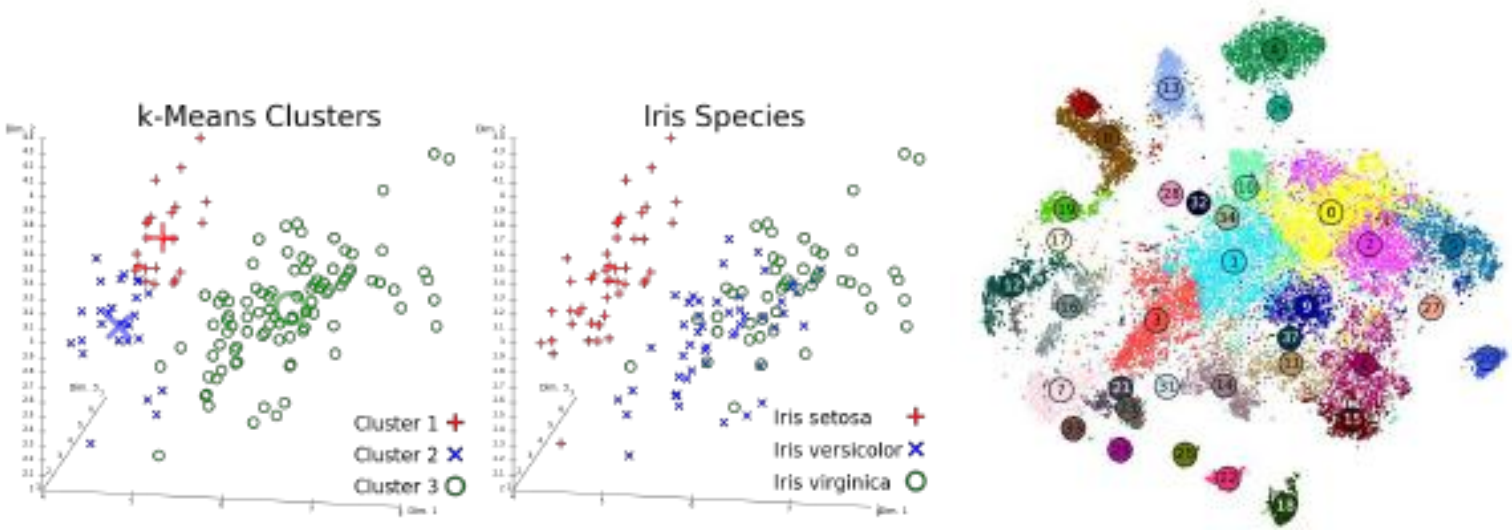


Figure: Left image from Wikimedia Commons. Right image from Kobak and Berens.

Fundamentals of Machine Learning: Building Models

- How to build models? We follow a typical pipeline in real scenarios:
- Amazon and Netflix recommendation systems, Google translate, self-driving cars.

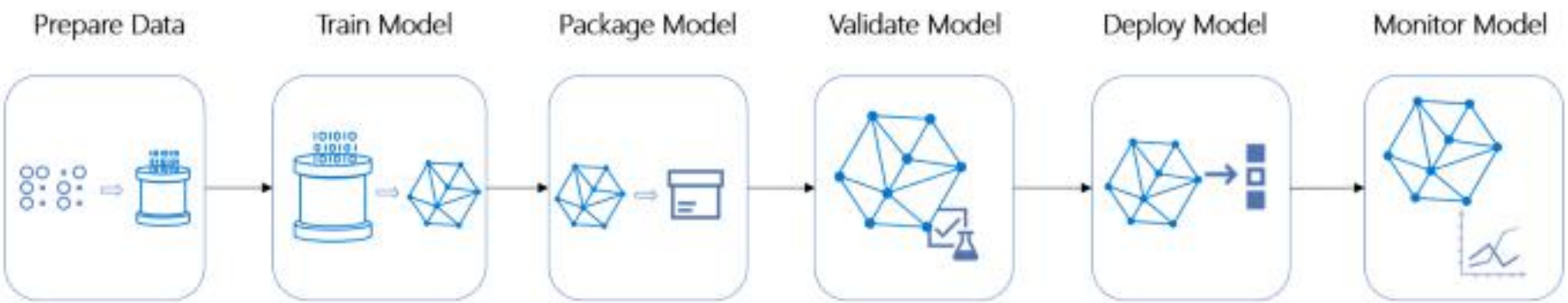


Figure: Image from Azure.

Intro to Supervised Learning

- Today's topic: building supervised machine learning models.
- Assume we have some covariates (or predictors) $X \in \mathbb{R}^{N \times D}$ and an output $y \in \mathbb{R}^N$.
- N is number of observations, D is dimensionality of inputs.
- Now we want to find some function $f(X)$ that learns the relationship between X and y.

Intro to Supervised Learning

- $f(X)$ is a function from X to y, and has parameters β that define the behavior of the function.
- We must also choose some *objective function*, $\mathcal{L}(\theta)$, which we use as a criteria to choose the parameters.
- Our choice of $f(X)$ and $\mathcal{L}(\theta)$ define our model assumptions.
- We must be very conscious of all our choices when building ML models.

Linear Regression

- We will first start from the most fundamental ML model, linear regression.
- Many of the techniques we use here will be repeated again for complex models.
- Assume we observe some features X, and the output y.
- Ex: y is housing prices, X is size of house and age of house.

Linear Regression: Defining f

- Consider the linear assumption for regression model:

$$y = f(X) = X\beta + \epsilon$$

- $\beta \in \mathbb{R}^p$ are the regression parameters
- $\epsilon \in \mathbb{R}$ is the noise parameter, $E[\epsilon] = 0$, var$(\epsilon) = \sigma^2$.

Linear Regression: Defining objective function

- What criteria should we use to learn β?
- The output of $f(X)$ should closely match the observed output, y.
- Common choice of objective function, sum of squared errors:

$$\mathcal{L}(\beta) \leftarrow (y - X\beta)^T(y - X\beta) = \sum_{i=1}^{N} (y_i - X_i \cdot \beta)^2$$

Linear Regression: Defining objective function

- Practically, we want to chose β so that the sum of squared errors is the lowest possible:

$$\hat{\beta} \leftarrow \underset{\beta}{\operatorname{argmin}}\, \mathcal{L}(\beta) = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{N} (y_i - X_i \cdot \beta)^2$$

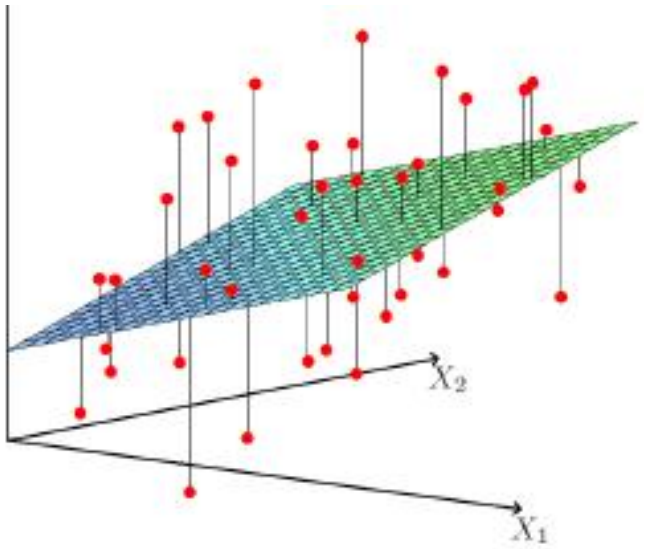


Figure: Image from "The Elements of Statistical Learning"

Linear Regression

- How do we optimize a function? Take derivative and set to zero, solve for parameter.

$$\begin{aligned}\hat{\beta} &\leftarrow \underset{\beta}{\operatorname{argmin}}\ (y - X\beta)^T(y - X\beta) \\ &= (X^T X)^{-1} X^T y\end{aligned}$$

Evaluating Regression Models

- How do we measure fitness using training data?
- How do we measure fitness using testing data?
- What are the model assumptions?
- Is there any multicollinearity problem?
- How do we select variables if we have many input variables?

Problems with Linear Regression

- Least-squares result can have high variance
- Ex: Colinearity of covariates–low prediction accuracy, $p > n$–no unique solution for least squares.
- Model interpretation–we are interested in a few important variables in the model.
- Ex: Model selection–2^p possible models, need more efficient methods.
- Solution–reduce the effect of the covariates ("regularized regression").

Shrinkage Methods

- Regularization by “shrinking” the regression parameters to zero
- Minimizing squared error–reduce variance by increasing bias:

$$\text{MSE}(\hat{\beta}) = \text{Bias}^2 + \text{var}(\hat{\beta})$$

- We look at two common methods: ridge regression and lasso regression.

Penalized Regression

- We can shrink the regression parameters by adding a "penalty" term to the loss function
- Ridge regression:

$$\hat{\beta} \leftarrow \underset{\beta}{\operatorname{argmin}} \left\{ (y - X\beta)^T (y - X\beta) + \lambda \sum_{j=2}^{p} \beta_j^2 \right\}$$

Penalized Regression

- Ridge regression:

$$\hat{\beta} \leftarrow \underset{\beta}{\operatorname{argmin}} \left\{ (y - X\beta)^T (y - X\beta) + \lambda \sum_{j=2}^{p} \beta_j^2 \right\}$$

- Meaning of λ: Strength of regularization
- Regression parameter estimator:

$$\hat{\beta} = (X^T X + \lambda I)^{-1} X^T y$$

Ridge Regression

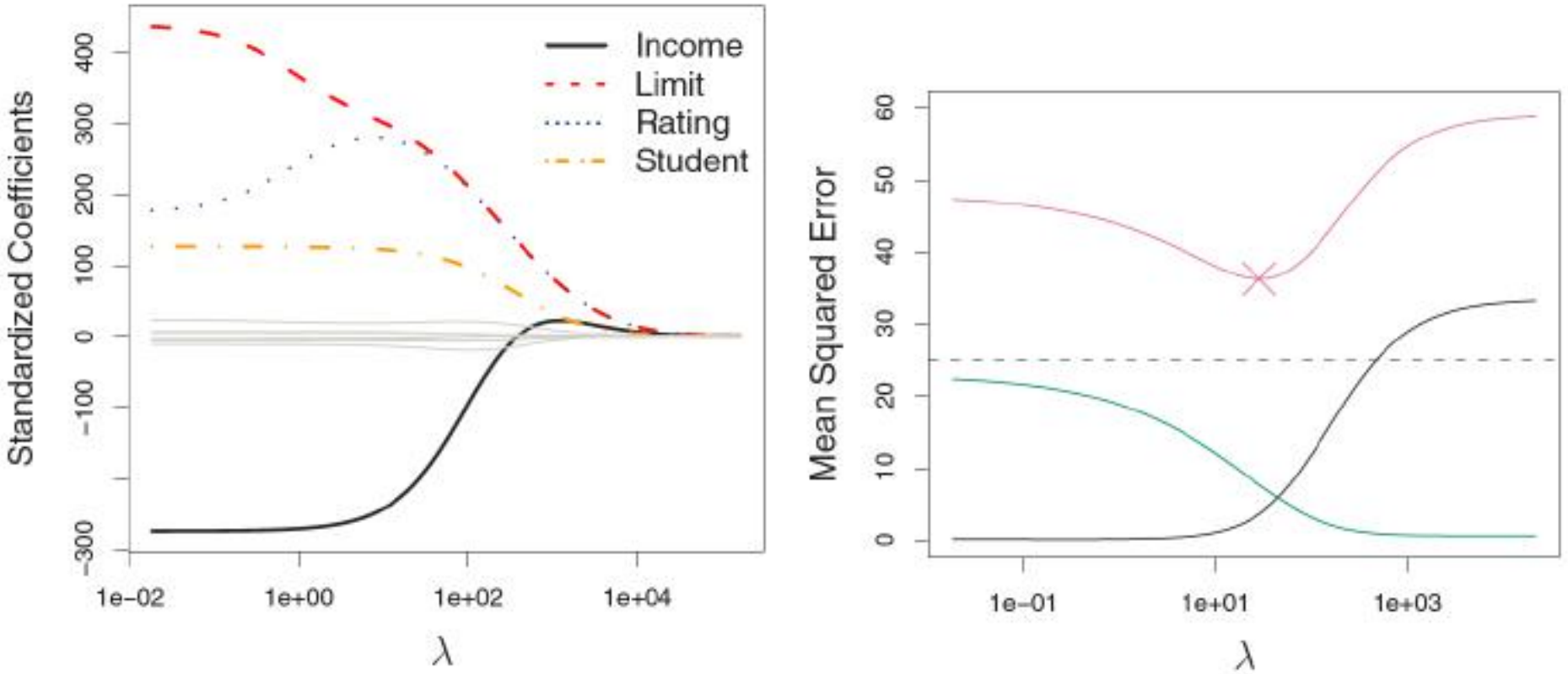


Figure: Left: 11 Regression Coefficients of Ridge Reg. for Credit data set. Right: Squared-bias (black), variance (green), MSE of Ridge (purple). Minimum possible MSE (dashed line)

Lasso Regression

- Alternatively, we can penalize with absolute value of β
- Lasso:

$$\hat{\beta} \leftarrow \underset{\beta}{\operatorname{argmin}} \left\{ (y - X\beta)^T (y - X\beta) + \lambda \sum_{j=2}^{p} |\beta_j| \right\}$$

Lasso Regression

- Lasso:

$$\hat{\beta} \leftarrow \underset{\beta}{\operatorname{argmin}} \left\{ (y - X\beta)^T (y - X\beta) + \lambda \sum_{j=2}^{p} |\beta_j| \right\}$$

- Difference with ridge regression–shrinks parameters exactly to zero.
- Have to obtain $\hat{\beta}$ with optimization techniques
- Benefit–interpretable variable selection.

Lasso Regression

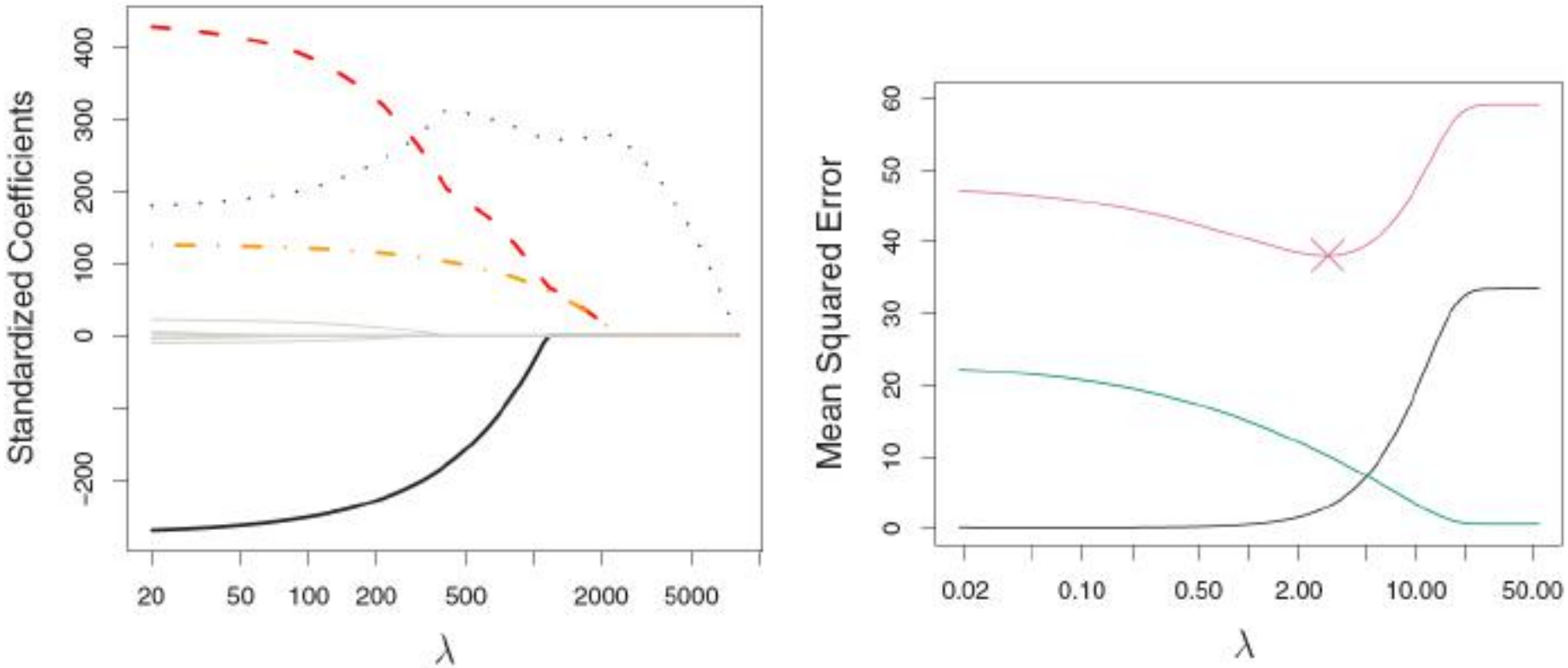


Figure: Left: 11 Regression Coefficients of Lasso. Right: Squared-bias (black), variance (green), MSE of Lasso (purple).

Lasso-Ridge Comparison

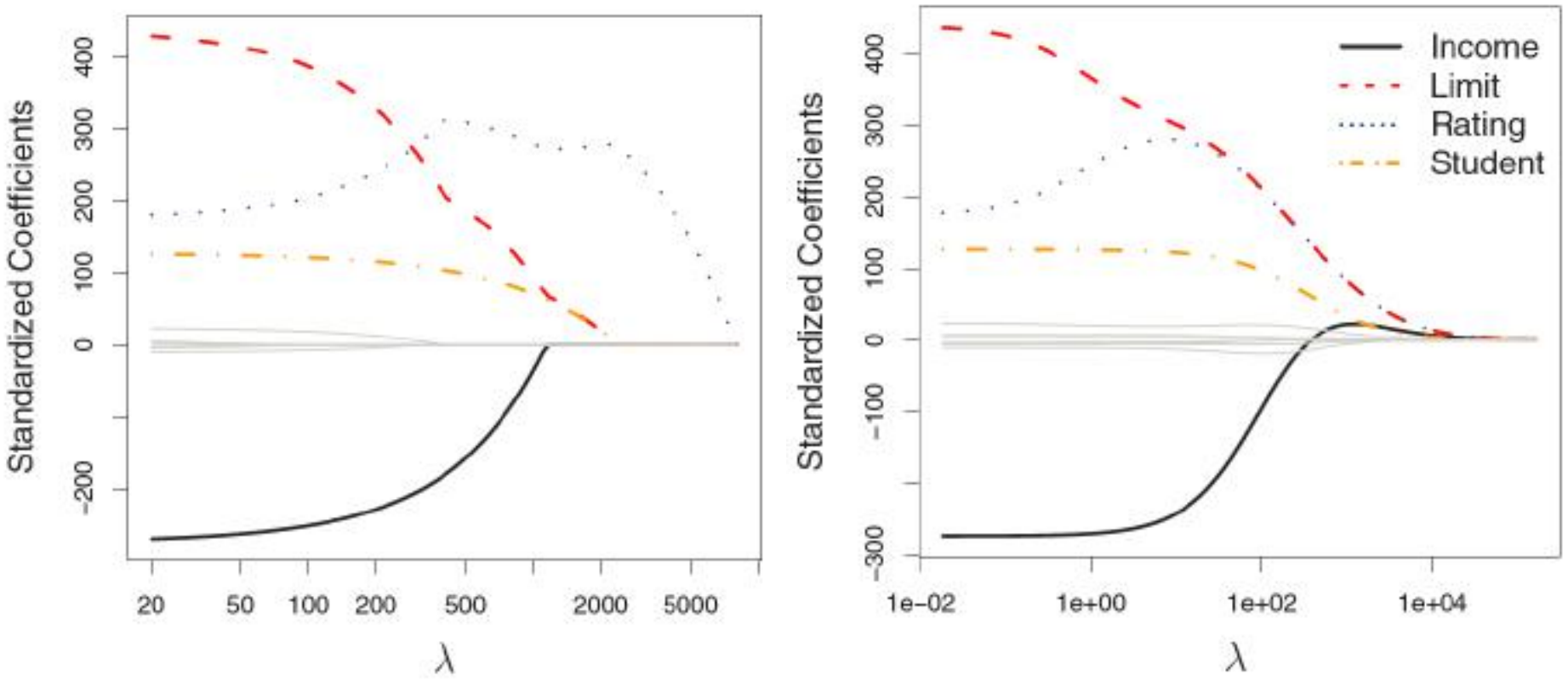


Figure: Left: Lasso Regression. Right: Ridge Regression

Lasso-Ridge Comparison

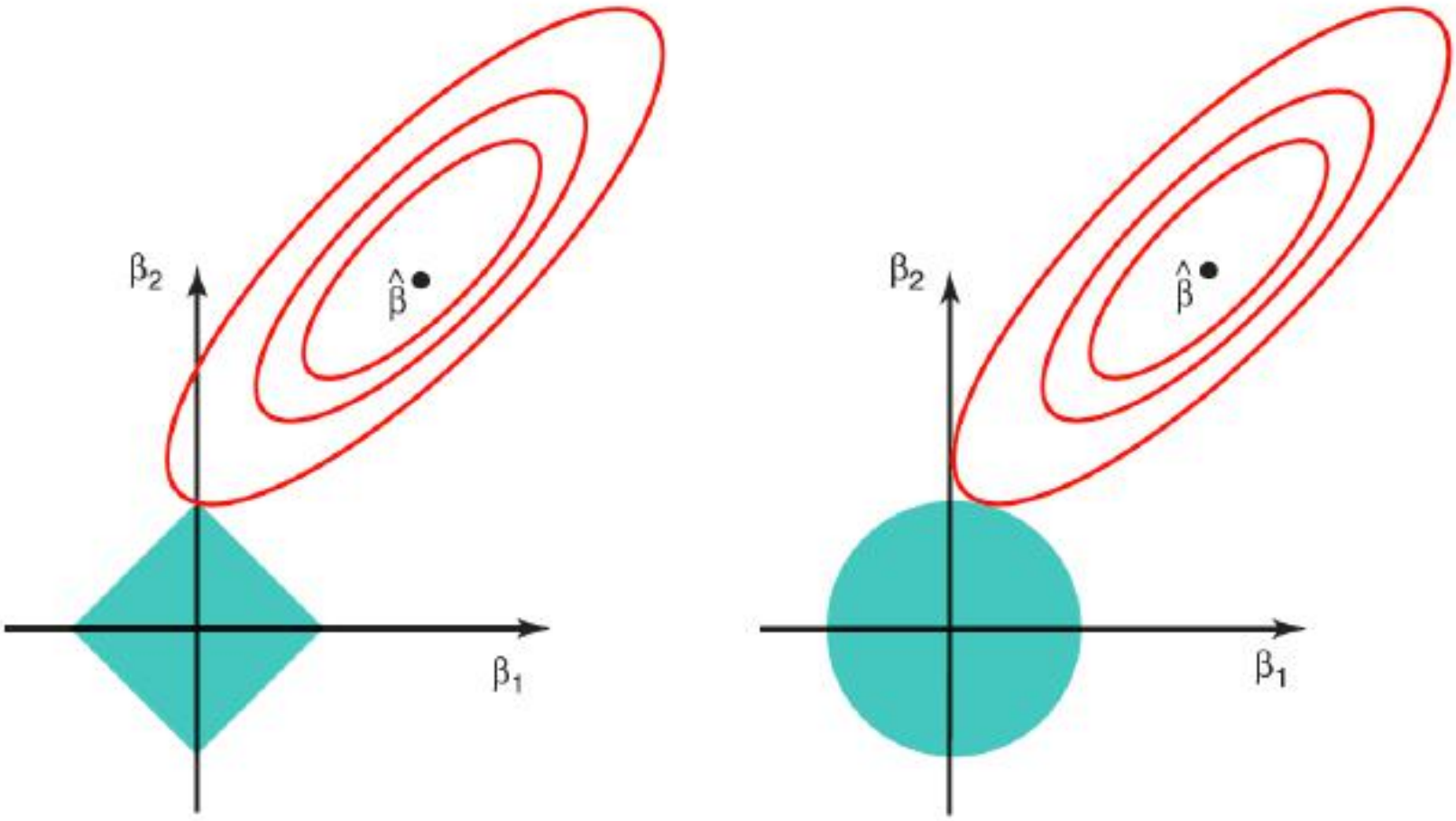


Figure: Left: Lasso Regression. Right: Ridge Regression

Lasso-Ridge Comparison

- When to use Lasso vs. Ridge regression?
- Ridge–output depends on many covariates, with similar size regression coefficients
- Lasso–output depends on small number of non-zero covariates

Choosing λ

- How to select λ? We can use cross-validation.
- Split data into k test-training partitions.
- Evaluate loss function on grid of λ values.
- Take average of loss over k-folds, choose λ with smallest error.

Discussion of Regularization

- Regularization techniques important for high dimensional data.
- Example: Genetics applications, record many covariates but for few individuals.
- Many regularization techniques for regression beyond ridge and lasso.

Logistic Regression

- Previous regression models are designed only for real valued outputs
- But what if our outputs take binary values?
- Ex: If I give you an image, can you predict if it's a dog or a cat?

Logistic Regression

- Again, we assume we have some inputs $X \in \mathbb{R}^{N \times D}$
- Now assume $y \in \{0, 1\}^N$, instead of $y \in \mathbb{R}^N$
- We need to learn the probability that $y_i = 1$, given the data.

Logistic Regression

- What is the objective function we should use here?
- Popular assumption: y_i follows a Bernoulli likelihood:

$$\mathcal{L}(\cdot) = \prod_{i=1}^{N} [f(X_i)]^{y_i} [1 - f(X_i)]^{1-y_i}$$

Logistic Regression

- Objective function:

$$\mathcal{L}(\cdot) = \prod_{i=1}^{N} [f(X_i)]^{y_i} [1 - f(X_i)]^{1-y_i}$$

- $f(X_i)$ must be between $(0, 1)$, but typical linear model $X\beta \in \mathbb{R}$.
- So we push $X\beta$ through an invertible "link function", $f(X_i) \in (0, 1)$:

$$f(X_i) = \frac{1}{1 + e^{-X_i\beta}}$$

Logistic Regression

- Objective function:

$$\mathcal{L}(\beta) = \prod_{i=1}^{N} \left[\frac{1}{1+e^{-X_i\beta}}\right]^{y_i} \left[\frac{e^{-X_i\beta}}{1+e^{-X_i\beta}}\right]^{1-y_i}$$

- This is a likelihood function, so we must maximize objective by principle of maximum likelihood estimate.

$$\hat{\beta} \leftarrow \underset{\beta}{\operatorname{argmax}}\, \mathcal{L}(\beta) = \underset{\beta}{\operatorname{argmax}} \prod_{i=1}^{N} \left[\frac{1}{1+e^{-X_i\beta}}\right]^{y_i} \left[\frac{e^{-X_i\beta}}{1+e^{-X_i\beta}}\right]^{1-y_i}$$

- $\hat{\beta}$ not available in closed form, must use optimization.

Logistic Regression

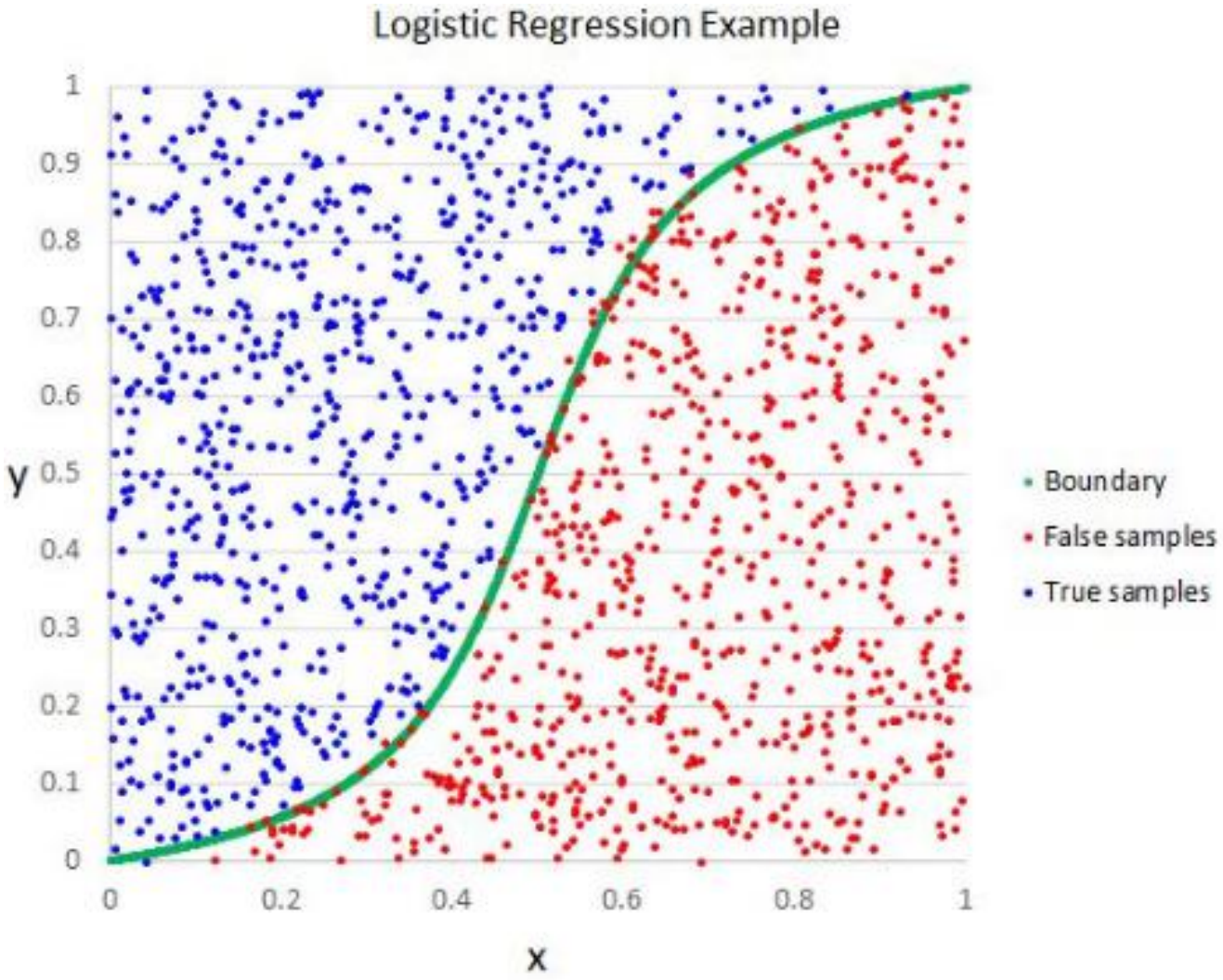


Figure: Image from Data Science Central.

Logistic Regression

- As in the regression case, the MLE estimate for logistic regression has same problems
- Again, we can use the regularization techniques from earlier here as well.

Unsupervised Learning

- Typical task–learn hidden structure in data.
- Problem–we don't have training data to use.
- Solution–use *unsupervised learning* techniques.

Dimension Reduction

- We have a high dimensional data set, $Y \in \mathbb{R}^{N \times D}$, with column mean of zero.
- Suppose we want to represent data in low-dimension, $X \in \mathbb{R}^{N \times K}$, $K << D$
- We call $W \in \mathbb{R}^{D \times K}$ the "principal components" and project with $X = YW$.

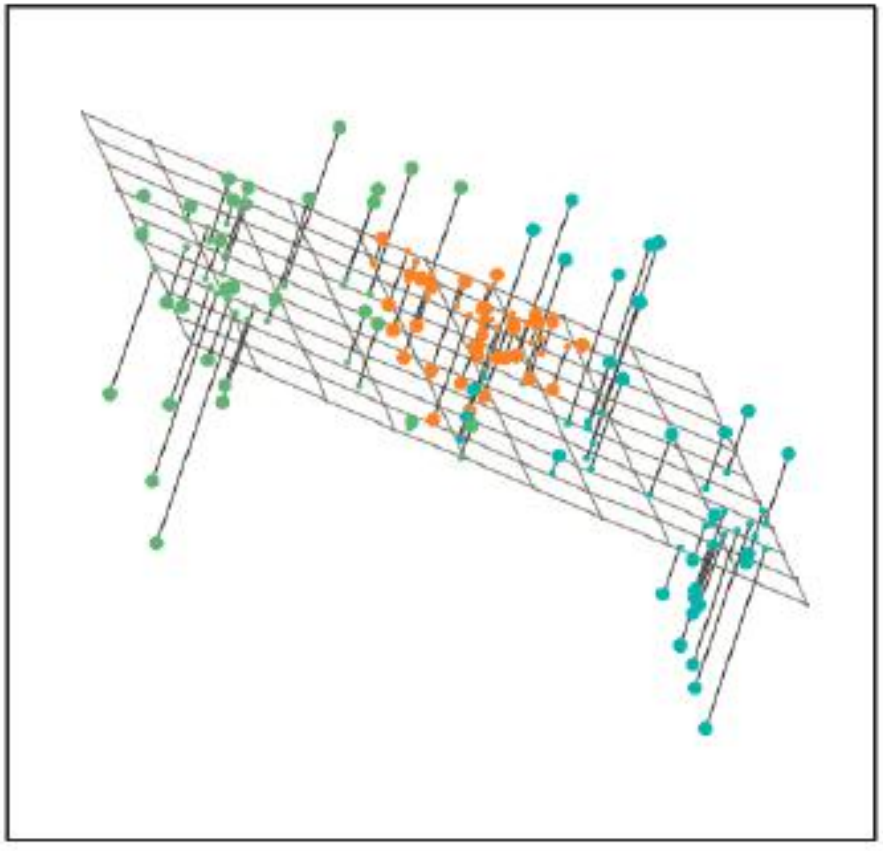
Principal Components Analysis

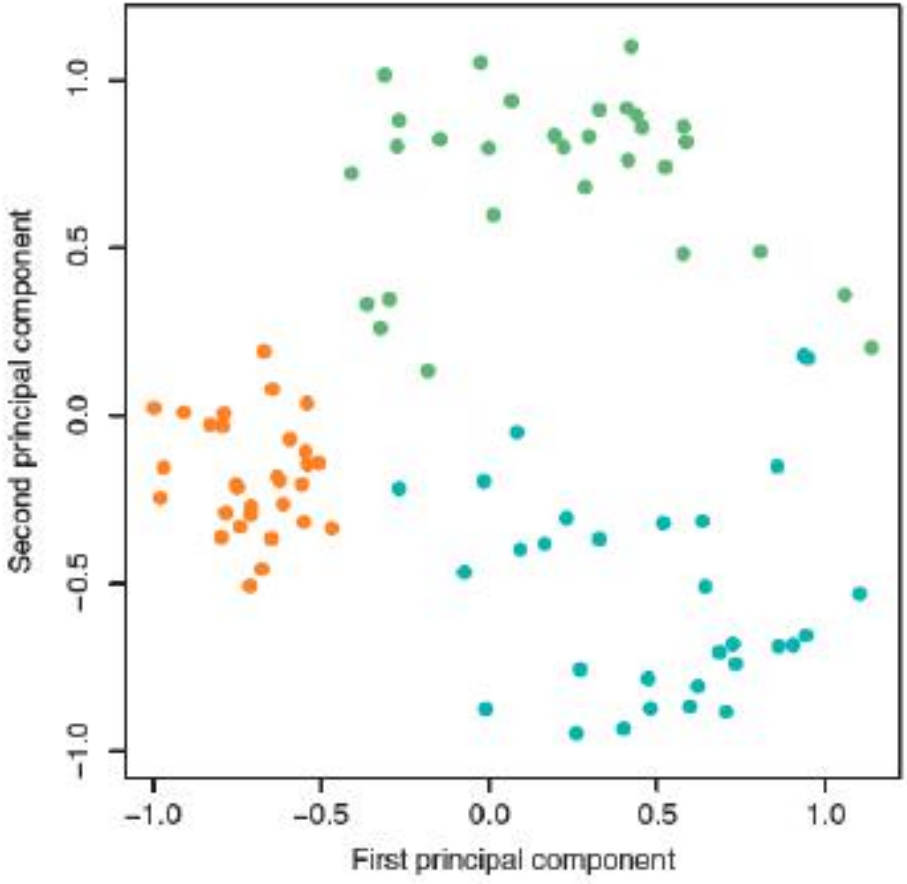
- We want low dimension representation to be accurate.
- Let $(YW)W^T$ be reconstruction of data.
- Objective function–minimize squared errors:

$$\mathcal{L}(W) = \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N} ||y_i - (y_i \cdot w_k) \cdot w_k^T||^2$$

- W could be arbitrarily large, so normalize to $||w_k||^2 = 1$.

Principal Components Analysis





Principal Components Analysis

- We can rewrite objective function

$$\mathcal{L}(w_k) = \frac{1}{N}\sum_{i=1}^{N} ||y_i - (y_i \cdot w_k) \cdot w_k^T||^2$$
$$= \frac{1}{N}\left(\sum_{i=1}^{N} ||y_i||^2 - \sum_{i=1}^{N} ||y_i \cdot w_k||^2\right)$$

  s.t $||w_k||^2 = 1, \forall k = 1, \ldots, K$
- Since $E[y_i \cdot w_k] = 0$, Var$(y_i \cdot w_k) = \frac{1}{N}\sum_{i=1}^{N} ||y_i \cdot w_k||^2$
- Minimizing error $\rightarrow$ maximizing variance.

Principal Components Analysis

- Optimization problem:

$$\mathcal{L}(w_k, \lambda_k) = \frac{1}{N}\left(\sum_{i=1}^{N} ||y_i \cdot w_k||^2\right) - \lambda_k \left(||w_k||^2 - 1\right)$$

Principal Components Analysis

- Take derivative:

$$\mathcal{L}(w_k, \lambda_k) = \frac{1}{N} \sum_{i=1}^{N} ||y_i \cdot w_k||^2 - \lambda_k(||w_k||^2 - 1)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_k} = w_k w_k^T - 1$$

$$\frac{\partial \mathcal{L}}{\partial w_k} = \frac{2}{N} Y^T Y w_k - 2\lambda_k w_k$$

Principal Components Analysis

- Set derivative to zero:

$$w_k w_k^T = 1$$

$$\left(\frac{Y^T Y}{N}\right) w_k = \lambda_k w_k$$

Principal Components Analysis

- Let $\Sigma = \frac{Y^T Y}{N}$, sample covariance
- Solution has linear algebraic interpretation:

$$w_k w_k^T = 1$$
$$\Sigma w_k = \lambda_k w_k$$

- Optimal w_k eigenvector is the one with the k-th largest eigenvalue, λ_k.

Singular Value Decomposition

- We can decompose matrix Y as

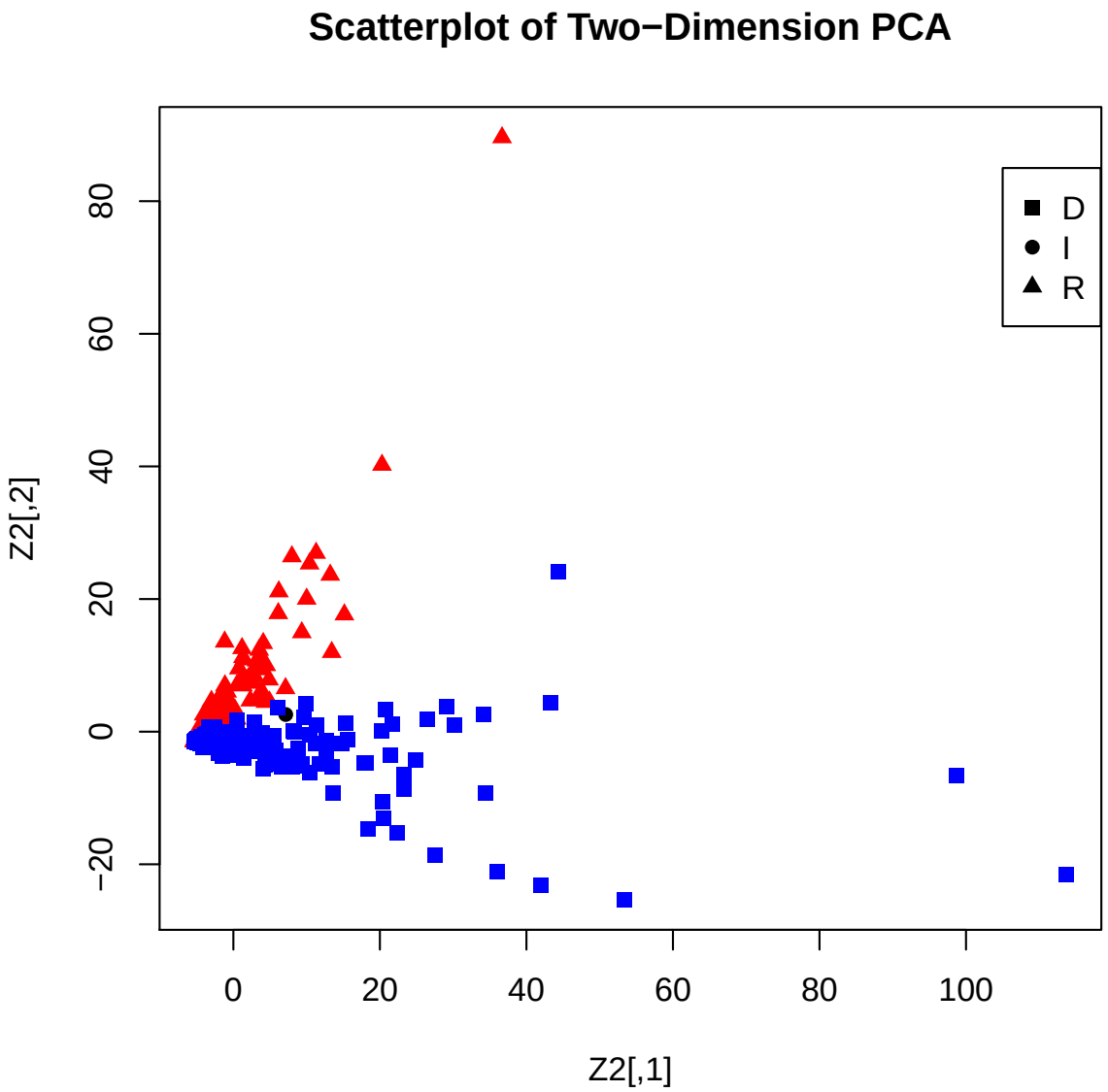
$$Y = UDW^T$$

  where $\text{diag}(D) = (\sqrt{\lambda}_1, \ldots, \sqrt{\lambda}_P)$
- Right singular vectors W are the eigenvectors of Y^TY s.t. $W^TW = I$.
- PCA projection $\rightarrow$ Multiply Y with its right-singular vectors.

Applications

- Dataset: Members of Congress and the number of times they utter a particular word.
- $Y_{i,d}$: Number of times person i says word d.
- High dimension–let's project to lower dimensions.

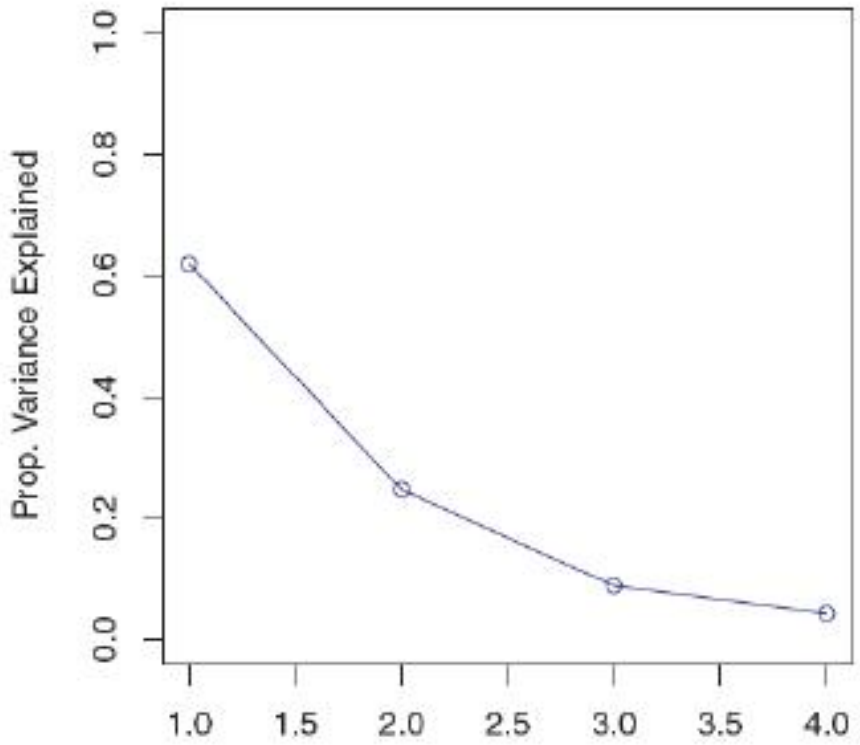
Applications

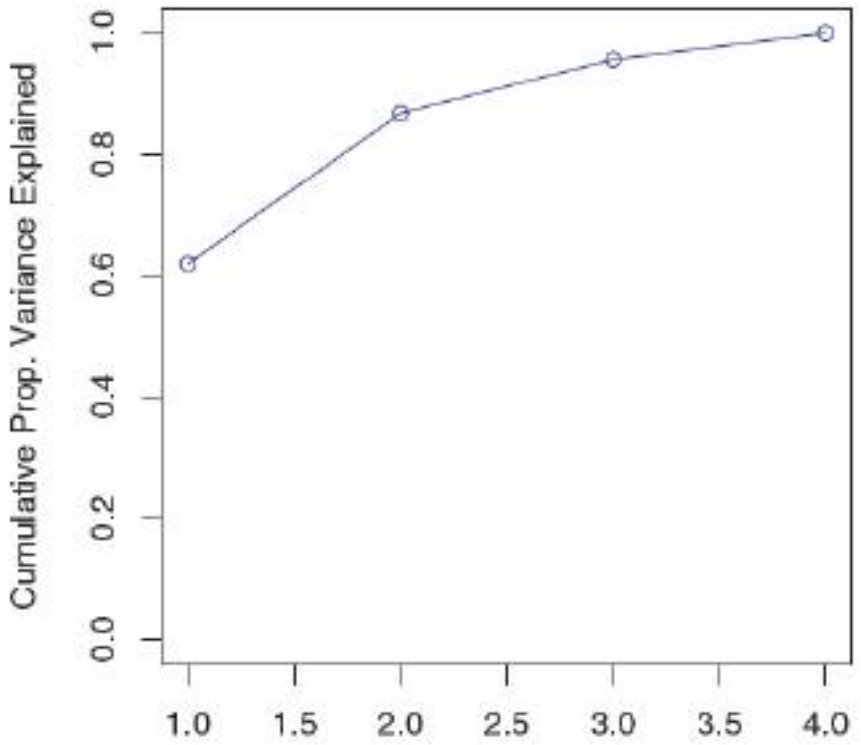


Practical Use

- How do we choose right number of principal components? Not easy to answer.
- We could look at proportion of variance explained by PCA
- Variance of the K-dim. PCA projection: $\frac{1}{N}\sum_{i=1}^{N}||y_i \cdot W||^2$
- Total variance of data: $\frac{1}{N}\sum_{i=1}^{N}||y_i||^2$

Selecting Number of Components





Conclusion

- Typical problem: Learn the hidden structure of the data.
- Todays lesson–one of the most popular techniques for dimensionality reduction.
- Used in visualization, downstream tasks like regression.
- Next lecture: Further applications of PCA, more sophisticated models.