Topic 1: Course Introduction, Math Review, and Software

Tian Xie[†]

[†]Singapore Management University and SHUFE

Course Overview

INSTRUCTOR	XIE Tian
EMAIL	`tianxie@smu.edu.sg` `xietian001@hotmail.com`
LOCATION	Online via ZOOM (ZOOM info will be posted regularly)
CLASS HOUR	Monday 7:00pm to 10:00pm (ten minutes break between each hour)

Course Contents

- Unfortunately, ECON.685 Elearn Section is down. I cannot post my course contents on time.

Course Contents

- Unfortunately, ECON.685 Elearn Section is down. I cannot post my course contents on time.
- We gonna use **GitHub** instead!!

Course Contents

- Unfortunately, ECON.685 Elearn Section is down. I cannot post my course contents on time.

- We gonna use **GitHub** instead!!
- The link for this course is:

  `github.com/xietian001/SMU.ML.Course`

- **All the course related contents** (outline, schedule, homework, codes, data, etc.) are available via the above link.
- You don't need to register. Just download the files.
- Contents are updated on a weekly basis.
- GitHub is a vastly popular website for codes sharing and project collaborating. Checkout `github.com` for further details.

Purposes of Our Course

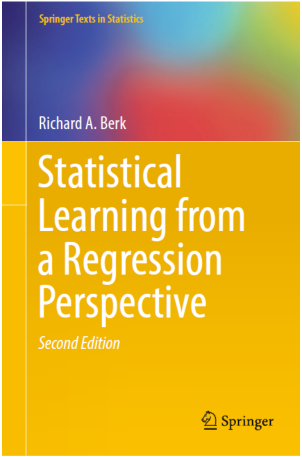
- Know the basics of the machine learning (ML) theory and practice of ML algorithms.
- Carry out simple empirical exercises using classic ML methods.
- Summarize and interpret ML results.
- Discuss the differences between alternative methods commonly used in ML projects.

Assessment Method

- Assignments **(40%)**
  - There will be two assignments handing out.
  - You are allowed to work in a group of **no more than 5 (including 5)** students and submit one copy of your assignments.
  - Of course, you can work the assignment just **by yourself**.
  - You need send the electronic version of your assignments to `tianxie@smu.edu.sg`.
  - You **must** state all the group members' names clearly on the cover page.
  - You **must** include the program codes in the assignment.
  - You can switch groups between the assignments.
- Class Performance **(10%)**
- Final exam **(50%)**

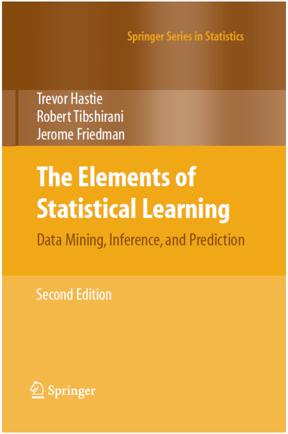
Recommended Textbook

- **Statistical Learning from a Regression Perspective (2nd Edition)** by Richard A. Berk.
- ISBN-13: 978-3319440477
- ISBN-10: 3319440470



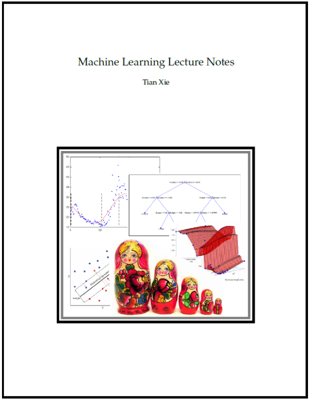
Supplementary Textbook

- **The Elements of Statistical Learning (2nd Edition)** by Trevor Hastie, Robert Tibshirani, and Jerome Friedman
- ISBN-13: 978-0387848570
- ISBN-10: 0387848576



Supplementary Lecture Notes

- I also uploaded my own lecture notes for you guys.
- **Machine Learning Lecture Notes** by Tian Xie.
- For those who don't want have a copy of the textbook, you can read my lecture notes instead.
- The course slides are abstracted from the notes.
- We will test contents from the **slides** only.
- Slides <– My Notes <– Textbook


Machine Learning Lecture Notes

Tian Xie

Contents

Machine Learning Concept

A Conventional Introduction

- Machine learning (ML) is the scientific study of **algorithms and statistical models** that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead.
- It is seen as a subset of artificial intelligence.
- The learning process can be categorized as **supervised learning** and **unsupervised learning**.
  - What is the difference?

Supervised and or Unsupervised?

- In a typical econometric analysis, we have a pair of $\mathbf{X}$ and $\mathbf{y}$. For example,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

where

  - $\mathbf{X}$ can be called the regressors, input variables, independent variables, or **features**.
  - $\mathbf{y}$ can be called the regressand, output variable, dependent variable, or **response**.
  - $\boldsymbol{\beta}$ is the coefficient vector, and $\boldsymbol{\epsilon}$ is the error term.
  - **Supervised learning** means, you have both features and the response.
    - You have input and output. You have a goal to help you decide/evaluate.
  - **Unsupervised learning** means, you only have features.
    - You only have $\mathbf{X}$.
    - You try to **learn the pattern** lurking inside of a data set.
- **Most of the economic problems we study require supervised learning**.

Supervised Learning

- For supervised learning, depending on **type** of the response variable, the problem we study is either a **classification** problem or a **regression** problem.

Supervised Learning

- For supervised learning, depending on **type** of the response variable, the problem we study is either a **classification** problem or a **regression** problem.
  - analyze pets: dog, cat, bird, fish...
  - forecast stock price using capitalization, liquidity, age of CEO...

Supervised Learning

- For supervised learning, depending on **type** of the response variable, the problem we study is either a **classification** problem or a **regression** problem.
  - analyze pets: dog, cat, bird, fish...
  - forecast stock price using capitalization, liquidity, age of CEO...
- Classification requires **categorical** responses and Regression requires **numerical** responses.
  - Which problem is more frequently encountered in economics?

Supervised Learning

- For supervised learning, depending on **type** of the response variable, the problem we study is either a **classification** problem or a **regression** problem.
  - analyze pets: dog, cat, bird, fish...
  - forecast stock price using capitalization, liquidity, age of CEO...
- Classification requires **categorical** responses and Regression requires **numerical** responses.
  - Which problem is more frequently encountered in economics?
  - Regression analysis is more popular in economics and finance.

Nonlinearity and Flexiblility

- Huge hype about machine learning in **Economics and Finance** now.
- Many people apply **fancy** ML algorithms to economic problem **brutally** without even knowing the reason and logic.
- Remember that we are studying **Economics and Finance**. There has to be some **motivation**.
- Of course, every data is unique. However, Economics and Finance data do have universal **patterns**.
  - For example, stocks prices are very hard to forecast, but stock volatilities are easy to predict.
  - It is common that certain algorithms have better performance than others.

Nonlinearity and Flexiblility

- Huge hype about machine learning in **Economics and Finance** now.
- Many people apply **fancy** ML algorithms to economic problem **brutally** without even knowing the reason and logic.
- Remember that we are studying **Economics and Finance**. There has to be some **motivation**.
- Of course, every data is unique. However, Economics and Finance data do have universal **patterns**.
  - For example, stocks prices are very hard to forecast, but stock volatilities are easy to predict.
  - It is common that certain algorithms have better performance than others.
- Many ML algorithms are **nonlinear** and **flexible**. They break the barriers of **linearity** and **parametric** formulation.
  - That is why they have good performance.
  - But say the data is super linear, a nonlinear algorithm shouldn't have a huge advantage.

Coding

The Role of Coding

- Coding is very **important** in studying ML.
  - Coding helps you better understand the contents and forces you to pay attention to shuttle details.

The Role of Coding

- Coding is very **important** in studying ML.
  - Coding helps you better understand the contents and forces you to pay attention to shuttle details.
- But our course is not called "ML in R or ML in MATLAB".
  - The primary concern of the course is not coding.
  - We will **NOT** test your coding skills in the final.
- Learning ML without coding is like learning swimming without getting wet.
- Following the Dean's "suggestion", we mainly use **R** in this course to demonstrate coding and estimation, therefore, you are recommended to follow our choice of software.
  - You are free to use whatever software you like, for example, Eviews, Stata, Matlab, R, Python, Java, C, C++, or even MS Excel, as long as you can deliver qualified course work.

R and RStuido - The Old-school Way

- To use R, you need to the **R source files first**.
  - You can obtain the files from `https://www.r-project.org/`.
  - It has many different versions that can generate various instability/incompatibility problems.
  - Have fun!
- Then, you need a good **R composer with nice UI**. The most popular one is **RStudio**.
  - You can obtain the free open source version from `https://rstudio.com/products/rstudio/download/`
- They are free and small size (less than 300M in total).
- You can install them in your own computer or in a flashdrive.
  - For flashdrive installation, you can plug-in and use immediately.
- **But we are not gonna do any of the above in this course.**

RStudio Cloud

- In this course, you are highly recommend to use the **RStudio Cloud** to learn R syntax, practice exercises, and do homeworks.
  - Visit the link:

    `rstudio.cloud`
  - **Register a free account** and start coding!
- Cloud computing has lots of merits:
  - No installation needed! Simply open a browser and stay online!
  - No instability or incompatibility.
  - The cloud records every steps of your coding process, so you never lose your codes, data, etc.
- Perhaps, the only drawback of cloud computing is that you have to stay online.

Console Window

- In the **Console window**, R responses to any input immediately, like a calculator. Try

  ```
  > 1+1
  [1] 2
  ```

  Notice that R immediately responses to your input and `[1]` implies the results are listed in the first row.
- R didn't record the result, since you didn't assign a variable. You can assign a variable to complicate calculation for re-use purpose. Try

  ```
  > x = 1+1
  ```

  Checkout the **Environment window**. You may notice a variable `x` with value 2.
- (?)Try `y <- 1+1`. What is the difference between = and `<-`?

Functions

- R can do way more than calculators. Try `rnorm(3)`.

```
> rnorm(3)
[1]  2.2461109  0.6867319 -0.7039494
```

- I am sure your results are different from mine. (?)Do you know why?
- To fully understand this command, its meaning, function, syntax, etc. Use `help(rnorm)` or simply type `?rnorm`.
- Its information is presented in the **help window**. Try to digest its meaning.
- (?)Generate 20 random results from $\mathrm{N}(10, 25)$.
- If your Console is messy, try `Ctrl+L` to clean the window.

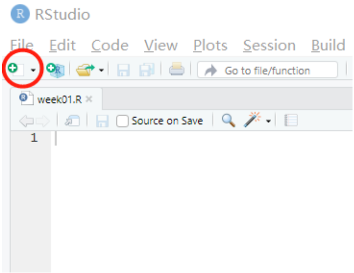
Exercise

1. Let us generate 10 random variables from standard normal distribution and compute their **mean** and **variance**.
   * *Note that you may need the functions:* `mean`, `var`.
2. Let us generate 100 random variables from standard normal distribution and compute their **mean** and **variance**.
3. Let us generate 10000 random variables from standard normal distribution and compute their **mean** and **variance**.

Notice any pattern?

Script Files

- Using the Console window to execute commands is rather inefficient.
- Like many other programmable software, we can use create a **script file** that consists of multiple lines of command and execute them **in sequence**.
- Click this icon to create an empty script and save this file in a designated **location** with proper **file name**.



- We can write the following lines to the script.

```
# Mean and Variance of normal RVs
x1 = rnorm(10)
x2 = rnorm(100)
x3 = rnorm(10000)
m1 = mean(x1)
m2 = mean(x2)
m3 = mean(x3)
v1 = var(x1)
v2 = var(x2)
v3 = var(x3)
```

- Select the lines you want to execute and click `Run` or use `Ctrl+Enter`. You should notice the new results in the **Environment Window**.
- You can use # to add **comments**. Contents after # are not executed.

Loops

- Now let us consider the following exercise: generate 100, 200, 300, ..., 100000 random variables from $\mathrm{N}(1, 1.5)$ and compute their means and variances.
- If you manually type up 100, 200, 300, ..., 100000, it will take forever to complete.
- Command `for` can repeat a pre-defined process multiple times. We usually refer this procedure as **loops**. The syntax of `for` is

  ```
  for (indicator in sequence){
  code
  }
  ```

  where `indicator` can be any parameters, `i, j, ...`
  `sequence` represents a sequence of data
  `code` can any estimating function you design

- Here is a demo code:

```
# use loop to obtain mean and variance
MEAN = 0;
VAR = 1;
n = seq(100,100000,by=100);
for (i in 1:length(n)){
  x = rnorm(n[i],mean=1,sd=sqrt(1.5))
  MEAN[i] = mean(x)
  VAR[i] = var(x)
}
```

Figures

- It is more intuitive to plot variables MEAN and VAR in **figures**.

```
# plot MEAN and VAR separately
plot(n,MEAN,col="blue",type="l",lwd=1,
     xlab="Sample Size", ylab="Value")
plot(n,VAR,col="red",type="l",lty="dashed",lwd=1,
     xlab="Sample Size", ylab="Value")
```

- Or you can plot both lines in the **same figure**.

```
# plot MEAN and VAR together
plot(n,MEAN,col="blue",type="l",lwd=1,
     xlab="Sample Size", ylab="Value",ylim=c(0.9,2))
lines(n,VAR,col="red",lty="dashed",lwd=1,
     xlab="Sample Size", ylab="Value")
legend(100,2,legend=c("Mean","Variance"),
       col=c("Blue","Red"),lty=1:2)
```

Exercise

- Plot the PDF of $\mathrm{N}(0, 1)$.
- Plot the PDF of t_2^2.
- Plot the CDF of t_{25}.
- Merge all three plots in one figure.
- You may need the command `dnorm`, `dt`, `pt`.

Answers

```
# plot the distribution
x = seq(-4,4,by=0.1)
y1 = dnorm(x,mean=0,sd=1)
y2 = dt(x,df=2)
y3 = pt(x,df=25)
plot(x,y1,type='l',lwd=1,col="blue",
     xlab="x",ylab="y",ylim=c(0,1))
lines(x,y2,lty="dashed",lwd=1,col="red")
lines(x,y3,lty="dotted",lwd=1,col="black")
legend(-4,1,legend=c("PDF of N(0,1)","PDF of t(2)","CDF of t(25)"),
       col=c("Blue","Red","Black"),lty=1:3)
```

Import Data

- In practice, it is quite common to performance analysis on **given data set**.
- Here we use the `movie.csv` data file to demonstrate.
  - First, you need to download the data `movie.csv` from `github.com/xietian001/SMU.ML.Course`.
  - Then, you need to **upload** the data to the **cloud**.
  - This data consists of 94 movies with their **open box office** and related variables. (?)What do you think determine a movie's sales?
- We can use `read.csv` command to import the data. We need tell R the **exact location** of the file.

```
# import movie data
LOC = "/cloud/project/movie.csv"
dat = read.csv(LOC,header=TRUE)
summary(dat)
```

- (?)What is the functionality of `summary()`?
- The variable `dat` is a stored in a **list** format.
  - Click the `dat` to see its contents.
  - To access each element, for example, `OpenBox`, you can use `dat$OpenBox`.

Exercise

- Plot **open box office** against **budgets** and add a 45^o line. What can you conclude?

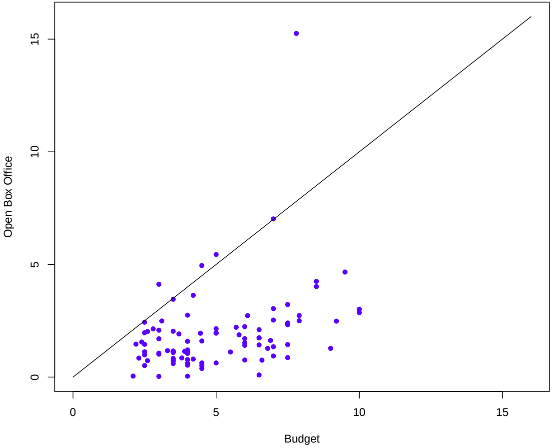
Exercise

- Plot **open box office** against **budgets** and add a 45^o line. What can you conclude?

```
# plot open box office against budget
x = dat$Budget
y = dat$OpenBox
plot(x,y,col="blue",pch = 16,xlim=c(0,16),ylim=c(0,16),
     xlab="Budget",ylab="Open Box Office")
lines(c(0,16),c(0,16),type="l",col="black")
```

Movie Plot

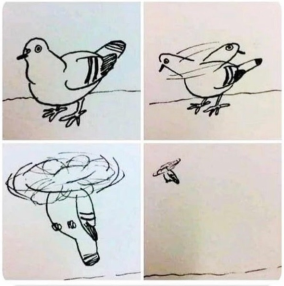
- Here is the plot using movie data.



How to Code?

- There is **no simple** answer.
- Remember, you are not professional programmer.
- You (probably) will not code for a living.

Figure: My Marking Standard



Math Review

Linear algebra

- We denote the set of all n-tuples of real numbers by $\mathbb{R}^n$. The set of n-tuples of nonnegative real numbers is denoted by $\mathbb{R}^n_+$.
- The elements of these sets will be referred to as points or **vectors**.
- If $x = (x_1, ..., x_n)$ is a vector, we denote then its i^{th} component is x_i.
- We can add two vectors by adding their components: $x + y = (x_1 + y_1, ..., x_n + y_n)$.
- We can perform scalar multiplication on a vector by multiplying every component by a fixed real number t: $tx = (tx_1, ..., tx_n)$.

- A vector x is a linear combination of a set of n vectors A if $x = \sum_{i=1}^{n} t_i y_i$, where $y_i \in A$ and the t_i's are scalars.
- A set A of n vectors is linearly independent if there is no set (t_i, x_i), with some $t_i \neq 0$ and $x_i \in A$, such that $\sum_{i=1}^{n} t_i x_i = 0$.
- An equivalent definition is that no vector in A can be represented as a linear combination of vectors in A.
- Given two vectors their **inner product** is given by $xy = \sum_i x_i y_i$. The norm of a vector x is denoted by $|x|$ and defined by $|x| = \sqrt{xx}$.
- Note that by the Pythagorean theorem, the norm of x is the distance of the point x from the origin; that is, it is the length of the vector x.
- If $xy = 0$, then x and y are said to be orthogonal.

- Let θ be the angle between x and y. It is clear $t|x| = |y|cos\theta$. Moreover, $xy = |x||y|cos\theta$.
- We can consider maps from $\mathbb{R}^n$ to $\mathbb{R}^m$ that send vectors into vectors. We denote such maps by $f : \mathbb{R}^n \to \mathbb{R}^m$.
- A map is a **linear** function if $f(tx + sy) = tf(x) + sf(y)$ for all scalars s and t and vectors x and y.
- If f is a linear function to $\mathbb{R}^1$, we call it a linear functional. If p is a linear functional we can represent it by a vector $p = (p_1, ..., p_n)$, and write $p(x) = px$.
- A set of points of form $H(p, a) = \{x : px = a\}$ is called a **hyperplane**.

Definite and semidefinite matrices

- Let A be a symmetric square matrix. Then if we post-multiply A by some vector x and pre-multiply it by the **transpose of the** same vector x, we have a quadratic form.

$$\begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = a_{11}x_1^2 + (a_{21} + a_{12})x_1x_2 + a_{22}x_2^2.$$

- Suppose that A is the identity matrix. In this case it is not hard to see that whatever the values of x_1 and x_2, the quadratic form must be nonnegative.
- In fact, if x_1 and x_2 are not both zero, $xAx^\top$ will be strictly positive. The identity matrix is an example of a positive definite matrix.

- **Definite matrices**. A square matrix A is:
  (a) positive definite if $x^\top Ax > 0$ for all $x \neq 0$;
  (b) negative definite if $x^\top Ax < 0$ for all $x \neq 0$;
  (c) positive semidefinite if $x^\top Ax \geq 0$ for all x;
  (d) negative semidefinite if $x^\top Ax \leq 0$ for all x.
- We say A is positive definite subject to constraint $bx = 0$ if $x^\top Ax > 0$ for all $x \neq 0$ such that $bx = 0$. The other definitions extend to the constrained case in a natural manner.

- If a matrix is positive semidefinite, then it must have nonnegative diagonal terms.
- The minor matrices of a matrix A are the matrices formed by eliminating k columns and the same numbered k rows. The naturally ordered or nested principal minor matrices of A are the minor matrices given by

$$a_{11} \quad \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \quad \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

- The minor determinants or minors of a matrix are the determinants of the minors. We denote the determinant of a matrix A by $\det A$ or $|A|$.

Cramer's rule

- Here is a convenient rule for solving linear systems of equations of the form

$$\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$$

  We can write his system more conveniently as $Ax = b$.
- **Cramer's rule**. To find the component x_i of the solution vector to this system of linear equations, replace the i^{th} column of the matrix A with the column vector b to form a matrix A_i. Then x_i is the determinant of A_i, divided by the determinant of A:

$$x_i = \frac{|A_i|}{|A|}.$$

Calculus

- Given a function $f : \mathbb{R} \to \mathbb{R}$, we define its derivative at a point x^* by

$$\frac{df(x^*)}{dx} = \lim_{t \to 0} \frac{f(x^* + t) - f(x^*)}{t}$$

  if that limit exists.

- The derivative $df(x^*)/dx$ is also denoted by $f'(x^*)$. If the derivative of f exists at x^*, we say that f is **differentiable** at x^*.

- Consider the linear function $F(t)$ defined by

$$F(t) = f(x^*) + f'(x^*)t.$$

  This is a good approximation to f near x^* since

$$\lim_{t \to 0} \frac{f(x^* + t) - F(t)}{t} = \lim_{t \to 0} \frac{f(x^* + t) - f(x^*) - f'(x^*)t}{t} = 0.$$

- In the same way, given an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}^m$, we can define its derivative at x^*, $D_f(x^*)$, as being that linear map from $\mathbb{R}^n$ to $\mathbb{R}^m$ that approximates f close to x^* in the sense that

$$\lim_{|t| \to 0} \frac{|f(x^* + t) - f(x^*) - D_f(x^*)t|}{|t|} = 0.$$

- We use norm signs since both the numerator and denominator are vectors. The map $f(x^*) + D_f(x^*)$ is a good approximation to f at x^* in the sense that for small vectors t,

$$f(x^* + t) \approx f(x^*) + D_f(x^*)t.$$

- Given a function $f : \mathbb{R}^n \to \mathbb{R}$, we can also define the **partial derivatives** of f with respect to x_i evaluated at x^*.
- To do this, we hold all components fixed except for the i^{th} component, so that f is only a function of x_i, and calculate the ordinary one-dimensional derivative.
- We denote the partial derivative of f with respect to x_i evaluated at x^* by $\partial f(x^*)/\partial x_i$.
- Since $D_f(x^*)$ is a linear transformation, we can represent it by a matrix, which turns out to be

$$D_f(x^*) = \begin{pmatrix} \frac{\partial f_1(x^*)}{\partial x_1} & \cdots & \frac{\partial f_1(x^*)}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_m(x^*)}{\partial x_1} & \cdots & \frac{\partial f_m(x^*)}{\partial x_n} \end{pmatrix}.$$

  The matrix representing $D_f(x)$ is called the **Jacobian matrix** of f at x^*. We will often work with functions from $\mathbb{R}^n$ to $\mathbb{R}$ in which case $D_f(x^*)$ will be an $n - by - 1$ matrix, which is simply a vector.

Higher-order derivatives

- If we have a function $f : \mathbb{R}^n \to \mathbb{R}$, the **Hessian matrix** of that function is the matrix of mixed partial derivatives

$$D^2 f(x) = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}.$$

Note that $D^2 f(x)$ is a symmetric matrix.

- Let $f : \mathbb{R}^n \to \mathbb{R}$ be a differentiable function and let x and y be two vectors in $\mathbb{R}^n$. Then it can be shown that

$$\begin{aligned} f(y) &= f(x) + D_f(z)(y - x) \\ f(y) &= f(x) + D_f(x)(y - x) + \frac{1}{2}(y - x)^\top D^2 f(w)(y - x). \end{aligned}$$

where z and w are points on the line segment between x and y. These expressions are called **Taylor series expansions** of f at x.

- If x and y are close together and the derivative functions are continuous, then $Df(z)$ and $D^2f(w)$ are approximately equal to $Df(x)$ and $D^2f(x)$, respectively. We therefore often write the Taylor series expansions as

$$
\begin{aligned}
f(y) &= f(x) + D_f(x)(y-x) \\
f(y) &= f(x) + D_f(x)(y-x) + \frac{1}{2}(y-x)^\top D^2 f(x)(y-x).
\end{aligned}
$$

Analysis

- Given a vector x in $\mathbb{R}^n$ and a positive real number e, we define an **open ball** of radius e at x as $B_e(x) = \{y \in \mathbb{R}^n : |y - x| < e\}$.
- A set of points A is a **open set** if for every x in A there is some $B_e(x)$ which is contained in A.
- If x is in an arbitrary set and there exists an $e > 0$ such that $B_e(x)$ is in A, then x is said to be in the **interior** of A.
- The complement of a set A in $\mathbb{R}^n$ consists of all the points in $\mathbb{R}^n$ that are not in A; it is denoted by $\mathbb{R}^n \backslash A$.
- A set is a **closed set** if $\mathbb{R}^n \backslash A$ is an open set. A set A is bounded if there is some x in A and some $e > 0$ such that A is contained in $B_e(x)$. If a nonempty set in $\mathbb{R}^n$ is both closed and bounded, it is called **compact**.
- A infinite **sequence** in $\mathbb{R}^n$, $(x^i) = (x^1, x^2, ...)$ is just an infinite set of points, one point for each positive integer.

- A sequence (x^i) is said to converge to a point x^* if for every $e > 0$, there is an integer m such that, for all $i > m$, x^i is in $B_e(x^*)$. We sometimes say that x^i gets arbitrarily close to x^*. We also say that x^* is the **limit** of the sequence (x^i) and write $\lim_{i \to \infty} x^i = x^*$. If a sequence converges to a point, we call it a **convergent sequence**.
- **Closed set**. A is a closed set if every convergent sequence in A converges to a point in A.
- **Compact set**. If A is a compact set, then every sequence in A has a convergent subsequence.
- A function $f(x)$ is continuous at x^* if for every sequence (x^i) that converges to x^*,w e have the sequence $(f(x^i))$ converging to $f(x^*)$.A function that is continuous at every point in its domain is called a **continuous function**.

Random Variables and Probability Distributions

- **Random variables**. A random variable is a numerical summary of a random outcome.
  - **Discrete** random variable: takes on only a discrete set of values (toss coins, roll dice,...). **Countable output**.
  - **Continuous** random variable takes on a continuum of possible values (stock return, GDP growth rate, ...). **Non-countable output**.
- The **probability** of an outcome is the proportion of the time that the outcome is **expected** to occur in the long run.
  - $^{(?)}$Question: what is difference between probability and frequency?

Random Variables and Probability Distributions

- **Random variables**. A random variable is a numerical summary of a random outcome.
  - **Discrete** random variable: takes on only a discrete set of values (toss coins, roll dice,...). **Countable output**.
  - **Continuous** random variable takes on a continuum of possible values (stock return, GDP growth rate, ...). **Non-countable output**.
- The **probability** of an outcome is the proportion of the time that the outcome is **expected** to occur in the long run.
  - $^{(?)}$Question: what is difference between probability and frequency? $^{(?)}$or say, what is difference between 0% and never gonna happen?

Random Variables and Probability Distributions

- **Random variables**. A random variable is a numerical summary of a random outcome.
  - **Discrete** random variable: takes on only a discrete set of values (toss coins, roll dice,...). **Countable output**.
  - **Continuous** random variable takes on a continuum of possible values (stock return, GDP growth rate, ...). **Non-countable output**.
- The **probability** of an outcome is the proportion of the time that the outcome is **expected** to occur in the long run.
  - (?)Question: what is difference between probability and frequency? (?)or say, what is difference between 0% and never gonna happen? In probability space, they are 0 and $\varnothing$

.

- **Probability distribution**. The probability distribution of a discrete random variable is the list of all possible values of the variable and the probability that each value will occur. These probabilities sum to 1.
- The **cumulative probability distribution** is the probability that the random variable is less than or equal to a particular value.
  - It is also referred to as a **cumulative distribution function (CDF)**.

Table: Probability of Donald Trump Tweeting X Times Per Day

	Number of Tweets					
	0	1	2	3	4	5
Probability Distribution	0.1	0.2	0.2	0.3	0.1	0.1
Cumulative Probability Distribution	0.1	0.3	0.5	0.8	0.9	1.0

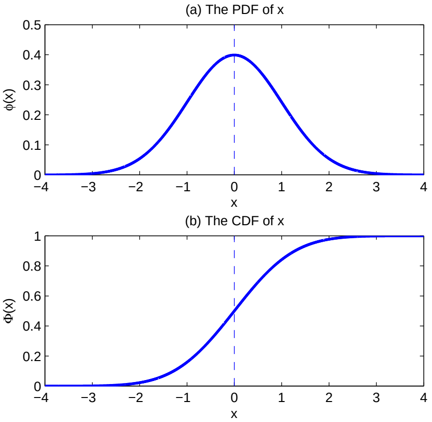
Bernoulli distribution

- A **binary** random variable is called a **Bernoulli** random variable, and its probability distribution is called the Bernoulli distribution.
- The outcome of a Bernoulli random variable can only take two values.
  - For example, 1 and 0, `True` and `False`, etc.
- For a Bernoulli random variable X with two outcomes 1 and 0, the probability distribution can be expressed as

$$f(X) = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } (1-p) \end{cases}$$

Continuous Random Variable

- A **continuous** random variable can have **infinite** number of outcomes.
- It can be constrained within a certain range or **unconstrained**.
- A typical example is the standard **Gaussian** random variable with probability **density** function (PDF) and CDF shown below:



Expectation

- The **expectation** of a random variable X measures its long-run average value. It is the sum of the any outcome times its respective probability.

$$\mathbb{E}(X) = \sum_{i=1}^{N} X_i \cdot p_i,$$

where X_i is one outcome and p_i is the associated probability.

- Recall the Tweeting example:

	Number of Tweets					
	0	1	2	3	4	5
Probability	0.1	0.2	0.2	0.3	0.1	0.1

(?) What is the expectation of Trump's tweets per day?

- If a is a constant, $\mathbb{E}(aX) = a\mathbb{E}(X)$.
- We also have $\mathbb{E}(X + Y) = \mathbb{E}(X) + \mathbb{E}(Y)$ for any X and Y.

Variance

- The **variance** measures the **dispersion** or the **spread** of a probability distribution:

$$\mathrm{Var}(X) = \mathbb{E}\left[\left(X - \mathbb{E}(X)\right)^2\right] = \sum_{i=1}^{N} \left(X_i - \mathbb{E}(X)\right)^2 \cdot p_i.$$

- $^{(?)}$What is the variance of Trump's daily tweets?
- **Standard deviation** is simply the square-root of variance.
- If a and b are constant, $\mathrm{Var}(aX) = a^2\mathrm{Var}(X)$ and $\mathrm{Var}(X + b) = \mathrm{Var}(X)$.
- $^{(?)}$Skewness and Kurtosis.

Joint Distribution

- The **joint** probability distribution of **multiple** random variables is the probability that the random variables **simultaneously** take on certain values.
- Let us now think about the two **sentimental** stage of D. Trump and relate to his tweeting behavior.
- Assume Trump's sentiment follows a Bernoulli distribution with two outcomes: **Rage** and **Calm**.

Rage Vs. Calm?

Marginal Distribution

- Now, Trump's tweeting behavior is measured by two random variables

Table: Joint Distribution of Donald Trump's Tweeting and Sentiment

Sentiment	Number of Tweets						**Total**
	0	1	2	3	4	5	
Rage	0.1	0.15	0.1	0.2	0.05	0.1	0.7
Calm	0	0.05	0.1	0.1	0.05	0	0.3
Total	0.1	0.2	0.2	0.3	0.1	0.1	1.0

- **Marginal probability distribution** is the sum of individual probabilities associated with one specific outcome of a certain random variable.
  - For example, $\Pr(\text{Rage}) = 0.1 + 0.15 + ... + 0.1 = 0.7$.

Conditional Distribution

- **Conditional distribution** is the distribution of a random variable Y conditional on another random variable X taking on a specific value, usually denoted as $\Pr(Y = y|X = x)$.
- The conditional probability can be estimated by

$$\Pr(X = x|Y = y) = \frac{\Pr(X = x, Y = y)}{\Pr(Y = y)}.$$

- **Conditional Expectation** and **Conditional Variance** are

$$\begin{aligned} \mathbb{E}(X|Y = y) &= \sum_{i=1}^{N} X_i \cdot \Pr(X = X_i|Y = y), \\ \mathrm{Var}(X|Y = y) &= \sum_{i=1}^{N} \left(X_i - \mathbb{E}(X|Y = y)\right)^2 \cdot \Pr(X = X_i|Y = y), \end{aligned}$$

Independence, Covariance, and Correlation

- Two random variables X and Y are **independent**, if knowing the value of one of the variables provides **no information** about the other.
  - If X and Y are independent, for all values of x and y, we have $\Pr(Y = y|X = x) = \Pr(Y = y)$.
- One measure of the extent to which two random variables move together is their **covariance**.
  - Specifically, $\mathrm{Cov}(X, Y) = \mathbb{E}[(X - \mathbb{E}(X))(Y - \mathbb{E}(Y))] = \mathbb{E}(XY) - \mathbb{E}(X)\mathbb{E}(Y)$.
- The correlation is an **alternative measure** of dependence between X and Y that follows unity.
  - Specifically,

$$\mathrm{Corr}(X, Y) = \frac{\mathrm{Cov}(X, Y)}{\sqrt{\mathrm{Var}(X) \cdot \mathrm{Var}(Y)}}$$

Independence, Covariance, and Correlation

- Independence implies both Covariance and Correlation equal 0.
  - $^{(?)}$ *vice versa?*
  - Try to the Covariance of $\mathbb{E}(X) = \mathbb{E}(X^3) = 0$, $Y = X^2$.
- We usually denote the variance as σ^2, for example $\mathrm{Var}(X) = \sigma_X^2$.
  - $\mathrm{Cov}(X, Y) = \sigma_{XY}$
  - Standard Deviation of X is σ_X
- $\mathrm{Var}(aX + bY) = a^2\sigma_X^2 + b^2\sigma_Y^2 + 2ab \cdot \sigma_{XY}$.
- By definition, $|\mathrm{corr}(X, Y)| \leq 1$.
- $^{(?)}$If X and Y are independent, what is $\mathrm{Var}(X + Y)$?