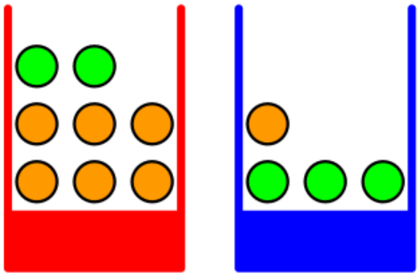
Probability Theory provides a framework for quantifying and manipulating uncertainty.

Bayesian Inference and Decision Theory allow to make optimal inference and decisions given the available information/observations.

Example: Balls and Boxes



Assume we pick a ball (eyes closed) from box A (red) with probability 40% and from Box B (blue) with probability 60%.
Two example queries:

- *What is the probability that a green ball is chosen?*
- *Given that we've picked a green ball, what is the probability that we picked from box B*

Note that the latter query is an inference problem of great practical interest. A similar query in a slightly different setting could be: what is the probability that a patient is sick given that the outcome of some diagnostic test is positive?

Example: Test Questions

A student is given a test with 10 questions of equal difficulty. The student can answer correctly or incorrectly.

Observations are formalized by a random variable D *reporting the number of correct answers. The ability of the student to answer correctly is given by the (unknown) parameter* θ.

- *Estimate the ability of the student to answer correctly,* i.e. *what is* θ *given observations* D*?*

At the heart of probability theory lies two objects: the *sample space* Ω, and a *probability measure*. The **sample space** is a set containing all possible outcomes in an experiment. The **probability measure** assigns a number p_i between 0 and 1 to each outcome ω, such that $\sum_\omega p_\omega = 1$.

- The symbol $\in$ means "is an element of"
- $\sum_{\omega \in \Omega} p_\omega$ means the sum of all elements ω in the set Ω
- In mathematics, curly brackets denote a set

Example: Sample spaces

- *Die:* $\Omega = \{1, 2, 3, 4, 5, 6\}$. *Then* ω *can be any of the numbers* 1 *to* 6. *If the die is unbiased, the probabilities for each outcome is* $\frac{1}{6}$.
- *One coin toss:* $\Omega = \{H, T\}$. *For an unbiased coin* $P(H) = P(T) = .5$
- *Two coin tosses:* $\Omega = \{(H, H), (H, T), (T, H), (T, T)\}$
- *Balls and Boxes:* $\Omega = \{(green, boxA), (green, boxB), (orange, boxA), (orange, boxB)\}$. *Even both boxes are selected evenly, and there is the same number of green and orange balls, the probabilities for each event are equal to* $.25$
- *Test questions:* $\Omega = \mathbb{R} \times \mathbb{N}^+$.
- *Wisconsin Breast Cancer Dataset (WBCD):* $\Omega = \mathbb{R}^{30} \times \{M, B\}$.

- $\mathbb{R}$ is the set of real numbers and $\mathbb{N}^+$ is the set of positive numbers (including 0)
- $\times$ is a Cartesian product. In short, it "combines" multiple spaces

Example:

- *What is the sample space of choosing a letter out of the alphabet?*
- *What is the sample space of two dice rolls?*
- *What is the sample space of two dice rolls of odd numbers?*

Note that for many experiments, there may be many possible sample spaces. Samples spaces can be **discrete** or **continuous**. At the depth of study of this class, many of the concepts we will see are analogous in both cases.

An **event** is a set of outcomes, that is, a subset of the sample set Ω.

Example:

- *Die:* $A = \{1\}$, $B = \{2\}$, $\Omega = \{1, 2, 3, 4, 5, 6\}$
- *Two coin tosses:* $A = \{(H, H)\}$, $B = \{(H, T)\}$,

- *WBCD: A = All malignant cases where the first feature is >20 and the second feature is <100.*

Example:

- *Provide an event in the Balls and Boxes example,*
- *Provide an event in the Test questions example.*

The probability of any multiple, mutually exclusive (non-overlapping) events is the sum of the individual event probabilities

$$P(A \cup B) = P(A) + P(B) \text{ if } A \text{ does not overlap with } B$$

- $A \cup B$ means the union of event A and B.

Note that $E = \Omega$ is a valid event, and $P(\Omega) = 1$. Consequently, the probability of all events complementary to A is $1 - P(A)$.

Example: Biased Die

Suppose we have a die in which odd number appear more often:

$$P(\textit{odd number event}) = \frac{2}{3}$$

Then:

$$P(\textit{even number event}) = \frac{1}{3}$$

We are mainly interested in **Random Variables**. A random variable is a mapping from an outcome ω to a space, such as integers or reals. The abbreviation for Random variable shall be ”rv”. Typically a capital letter is used to denote the rv.

Example: Coin Toss

$$\omega \in \Omega = \{\mathit{heads}, \mathit{tails}\}$$

> *For ease of notation, we match heads and tails to the number 1 and 0, respectively. The function that matches ω to the numerical values 0, 1 is a random variable*

$$X = \begin{cases} 1, \textit{ if } \omega = heads \\ 0, \textit{ if } \omega = tails \end{cases}$$

Example: Book in a library

> *Random variables are useful tools to relate random outcomes to real measurements. Suppose a book is picked at the library. The sample space consists of all the books. An outcome is a book. One random variable could be the number of pages in that book. Another might be the year of publication. Yet another might be its condition. Evidently, these are valid rvs in the sense that each book is an outcome and is mapped to some (fixed) numerical value. With rvs, One can perform queries of interest like: how many pages do books have on average? or given that a book was published before 1970, what is the average condition.*

Example: Other examples of rv

- *Die: X is equal to 1 if the die rolls an odd number.*
- *WBCD: X are features. Another example is the diagnostic Y.*
- *Balls and Boxes: $B = 1$ if a green ball was drawn, 0 otherwise.*
- *Test questions: D is the number of correct answers.*

The **probability** that a random variable X takes value x is written $P(X = x)$. In discrete space:

$$P(X = x) = \sum_{\omega \text{ with } X(\omega)=x} p_\omega.$$

- $X(\omega)$ refers to the value of X generated by the outcome labeled ω

Example: Probability of an odd dice roll

> *Assuming a fair die,* $p_\omega = \frac{1}{6}$ *for all* $\omega \in \{1,2,3,4,5,6\}$*. Then* $P(X = odd) = p_1 + p_3 + p_5 = \frac{1}{2}$ *Note that in this example,* X *is equal to the event* $\{1,3,5\}$.

Probabilities of an rv can also be estimated from measurements: $P(X = x)$ is equal to the fraction of trials such that X is equal to x.

Example: Coin Tosses

> *Assume the outcome of* N *coin tosses is* n_h *heads and* n_t *tails.*
>
> $$P(X = 1) = \frac{n_h}{N}$$
>
> $$P(X = 0) = \frac{n_c}{N}$$

Note that these measurements will be exact only for an infinite number of tosses ($N \to \infty$). For a finite number of samples, we define the frequency f.

In the case of continuous variables, the definition of a rv probability is slightly different. This is because the probability that a variable is exactly equal to, say, a real number is zero. Instead, in continuous variables, probabilities are defined with intervals:

$$p(a < x < b) = \int_a^b f(x)\mathrm{d}x$$

where f is a probability density function.

Example: Gaussian Random Variable

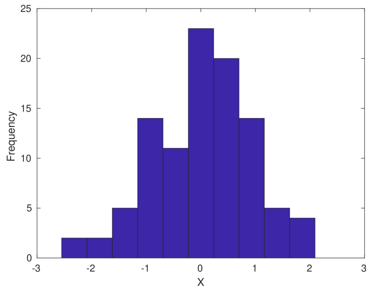
> *The probability of a Gaussian distributed rv (and any other continuous rv) is defined by:*
>
> $$p(a < x < b) = \int_a^b \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \mathrm{d}x$$
>
> *where* f *is the Gaussian density function and* μ *and* σ *are parameters.*

A **histogram** is a representation of the distribution of a rv. It is usually used to visualize a continuous rv, but it can also be use to visualize a discrete variable. A histogram count the number of observations that fall into n disjoint categories (bins). Histograms are extremely useful to visualize data distribution in one or two dimensions!

Example: Histogram of a Gaussian Random Variable

scripts/gaussian_histogram.m

```
mu = 0
std = 1
%Generates 100 Gaussian distributed samples
x = normrnd(mu, std, [100,1])
%Create a histogram of x
histogram(x)
ylabel('Frequency')
xlabel('X')
```



If the distribution of a random variable is known, it is possible to draw samples of that rv **Random Sample**. In some ways, sampling can be viewed as a simulation of a random system.

Example: Random Coin Flips

scripts/coin_flip.m

```
N_tosses = 100;
probability = .5; %Fair coin
X = binornd(1, probability)
%X =
%
% 0
```

Note that probabilities and random samples are two different concepts! The probability distribution is a means to draw a sample.

The **Joint probability** of two or more random variables is $P(X_1, X_2, ...)$. $P(X_j)$ for $j = 1, 2, ...$ is called the **Marginal Probability**.

Example: Double Coin Toss

X_1 is the outcome of coin 1
X_2 is the outcome of coin 2

Assume 100 double coin flips. The outcome can be summarized as:

	$X_1 = 1$	$X_1 = 0$
$X_2 = 1$	*22*	*20*
$X_2 = 0$	*28*	*30*

For example, the 22 in the top-row left-column entry indicates that both coins landed heads 22 times out of 100. The probability that both coins land heads is:

$$P(X_1 = 1, X_2 = 1) \cong f_{11} = 22/100$$

- $\cong$ means approximately equal to. This is to emphasize that frequencies are equal to probabilities only if the number of trials is infinite.

Probability of coin 1 toss irrespective of the outcome of coin toss 2 is $P(X_1)$. It is the marginal probability of X_1. From the table above, we see that:

$$P(X_1 = 1) \cong f_1 = \frac{22 + 28}{100}$$

*This is a special case of the **sum rule***

Probability of coin 1 toss results in $heads$ given that coin toss 2 results in $heads$ is $P(X_1 = 1|X_2 = 1)$. From the table above, we see that:

$$P(X_1 = 1|X_2 = 1) \cong f_{12} = \frac{22}{22 + 20}$$

scripts/coin_flips_mean_variance.m

```
N_tosses = 100;
probability = .5; %Fair coin
X = binornd(1, probability, N_tosses, 2) %Bernouilli rv
%X =
%
% 0 1
% 1 0
% 1 1
% 1 0
% 1 1
% ...

sum(X(:,1)==0)/N_tosses
%ans =
%
% 0.5200

cond = X(:,2)==0
sum(X(cond,1))/sum(cond)
%ans =
%
% 0.4821
```

Two rvs X_1, X_2 are **independent** if:

$$P(X_1, X_2) = P(X_1)P(X_2)$$

The **Sum rule**:

$$P(X_1) = \sum_x P(X_1, X_2 = x)$$

Marginal probabilities are obtained using the sum rule.

- In mathematics, it is common to omit the variables being summed over, or omit the bounds of the variables if it is obvious from the context. Here $\sum_x$ means we sum over every value that rv X may take

The **Product Rule**:

$$P(X_1|X_2) = \frac{P(X_1, X_2)}{P(X_2)}$$

$P(X_1|X_2)$ is the **Conditional Probability** of X_1 given X_2. Note that $P(X_1|X_2) \neq P(X_2|X_1)$!

Bayes Rule:

$$P(X_1|X_2) = \frac{P(X_2|X_1)P(X_1)}{P(X_2)}$$

The Bayes rule is often written:

$$P(X_1|X_2) = \frac{P(X_2|X_1)P(X_1)}{\sum_x P(X_2|X_1=x)P(X_1=x)}$$

In this case, we see that the Bayes rule allows to ”invert” conditional probabilities.

Exercise: Balls and Boxes (continued)

> *We have 2 rvs:*
>
> $$Box \in \{A, B\}$$
>
> *and*
>
> $$Ball \in \{green, orange\}$$
>
> *We know the probabilities $P(box)$ and $P(Ball|Box)$. With the Bayes rule, calculate:*
>
> - $P(Ball = green)$
> - $P(Box = B|Ball = green)$

Using Bayes rule to update the probability of a hypothesis given evidence/information/observation is called **Bayesian Inference**.

Exercise: Test questions (continued)

> *We have 2 rvs: D the number of correct answers and θ. We assume that know the prior distribution $P(\theta)$ and the likelihood $P(D|\theta)$. With the Bayes rule, one can calculate the posterior $P(\theta|D)$.*

A note on terminology:

- The probability $P(\theta)$ is call the **prior**. It reflects our prior knowledge (bias) about the ability of the student θ, and is either assumed by the modeler or assumed to be the uniform distribution.
- $P(D|\theta)$ is the **Likelihood function**, describing how likely the data D is given the parameter is equal to θ. The likelihood is often assumed by the modeler.
- $P(\theta|D)$ is called the **Posterior distribution**. The posterior distribution describes the distribution of θ after the fact (hence posterior) that we have observed D. This is often what one seeks in Bayesian Inference

In words, the Bayes rule becomes:

$$Posterior = \frac{likelihood \times prior}{marginal\ likelihood}$$

From the example above, it should be evident that computing posterior probabilities is not an easy task. In fact it is often intractable and one must resort to approximations.

The following page lists some distributions that we will encounter during this class.

Bernoulli Distribution
Describes the probability of X=1. *e.g.* Coin tosses $X \sim Bern(p)$
$P(X=1)=p, P(X=0)=1-p$
$\mathbb{E}(X)=p$
$\text{Var}(X)=p(1-p)$
The ML estimator of p is simply the sample mean

- The notation $\sim$ means "distributed as"

Binomial Distribution
The Binomial distribution describes the number of successes in a sequence of n Bernouilli Trials with probability p. *e.g.* Multiplie coin tosses $X \sim Bin(n,p)$
$\mathbb{E}(X)=np$
$\text{Var}(X)=np(1-p)$
$P(k)=P(X=k)=\binom{n}{k}p^k(1-p)^{n-k}$
The Bernouilli distribution and the Binomial distribution are equal if $n=1$.
Assuming the number of trials n is known, the parameter p is the proportion of positive outcome, or the sample mean divided by n.

Multinomial Distribution
The Multinomial distribution extends the binomial distribution to k possible outcomes per experiments *e.g.* Dice rolls. $X \sim MN(n,p)$
$\mathbb{E}(X_i)=np_i$
$\text{Var}(X_i)=np_i(1-p_i)$
$P(X)=\frac{n!}{x_1!\cdots x_k!}p_1^{x_1}p_2^{x_2}\cdots p_k^{x_k}$
The Multinomial distribution and the Binomial distribution are equal if $k=2$.

The **Multinomial Distribution** extends the binomial distribution to k possible (mutually exclusive) different outcomes. The parameters of the multinomial are $p_1,...,p_k$ are the corresponding probabilities for the different outcomes. The Multinomial distribution and the Binomial distribution are equal if $k=2$.

Example: Dice roll

Assume a 6-faced, fair die:

$$p=(\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6}), \quad n=3$$

One possible outcome is $X = (0, 1, 2, 0, 0, 0)$ meaning that face 2 was rolled once and face 3 was rolled twice.

$$P(X) = \frac{3!}{1!2!}(\frac{1}{6})^1(\frac{1}{6})^2$$

This can be calculated using the matlab function mnpdf

scripts/multinomial.m

```
mnpdf([0,1,2,0,0,0],[1./6 1./6 1./6 1./6 1./6 1./6])
```

Note that the leading term ($\frac{3!}{1!2!}$) is a normalization term that accounts for the fact that the order of the events is not accounted for, i.e. *rolling 2, then 3 twice, followed by 2 is the same as rolling 3 followed by 2 twice.*

Uniform Distribution (Discrete)

The discrete uniform distribution is one where any value in a given interval $(a, b]$ is equally likely. $X \sim Uniform(a, b)$
$\mathbb{E}(X_i) = \frac{a+b}{2}$
$\text{Var}(X_i) = \frac{(b-a+1)^2-1}{12}$
$P(X) = \frac{1}{b-a+1}$

Uniform Distribution

Continuous probability distribution defined on the interval $[a, b]$.
$X \sim U(a, b)$ $\mathbb{E}(X) = \frac{1}{2}(a + b)$
$\text{Var}(X) = \frac{1}{12}(b - a)^2$
$pdf(x) = \frac{1}{b-a}$ if $x \in [a, b]$. 0 otherwise.

Normal Distribution

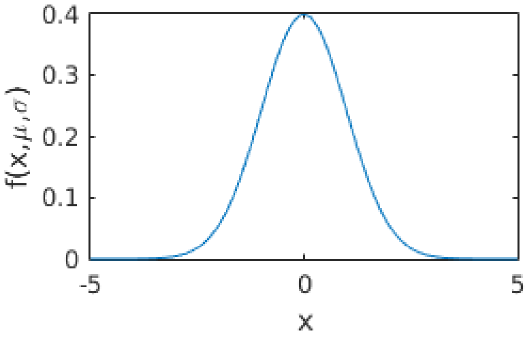
Continuous probability distribution defined on the interval $[-\infty, \infty]$.
$X \sim N(\mu, \sigma^2)$ $\mathbb{E}(X) = \mu$
$\text{Var}(X) = \sigma^2$
$pdf(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

The **Gaussian** or **Normal** distribution is important due to the fact that the distribution of sums of independent variables tend to a Gaussian distribution (Central Limit Theorem). The probability density function of the Gaussian distribution is:

$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

scripts/gaussian.m

```
x=linspace(−5,5,100);
px = normpdf(x, 0, 1);
plot(x,px);
```



The probability of a Gaussian distributed rv (and any other continuous rv) is defined by:

$$p(a < x < b) = \int_a^b f(x; \mu, \sigma)\mathrm{d}x$$

To avoid solving an integral to calculate a probability, it is more common to use the cumulative distribution function F defined as:

$$F(x < b) = \int_{-\infty}^b f(x; \mu, \sigma)\mathrm{d}x$$

Following the properties of the integral, probabilities of x can then be calculated as

$$p(a < x < b) = F(x < b) - F(x < a)$$

scripts/gaussian_px.m

```
p_ = normcdf(1,0,1)−normcdf(−1,0,1);
```

Beta Distribution
Defined on the interval $[0, 1]$ and parametrized by two positive parameters $X \sim Beta(\alpha, \beta)$
$\mathbb{E}(X) = \frac{\alpha}{\alpha+\beta}$
$Var(X) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$
$pdf(x) = x^{\alpha-1}(1-x)^{\beta-1}/B(\alpha, \beta)$
where B is a normalizing constant. Its analytical form is non-trivial, and omitted here.

Distribution	PDF	CDF	Random Number Generation
Normal	normpdf	normcdf	norm
Uniform (continuous)	unifpdf	unifcdf	unifrnd
Beta	betapdf	betacdf	betarnd
Exponential	exppdf	expcdf	exprnd
Uniform (discrete)	unidpdf	unidcdf	unidrnd
Bernouilli	binopdf	binocdf	binornd
Binomial	binopdf	binocdf	binornd
Multinomial	nmpdf	mncdf	mnrnd
Poisson	poisspdf	poisscdf	poissrnd

Parameter Estimation
Often, we must estimate the parameters of the distributions based on emporical data. Parameter estimation theory is a branch of statistics that deals with this problem.

Sample Mean and Variance
A reminder that rvs can be **discrete** or **continuous**. In the following we focus on discrete rvs.

The **Expectation** of a discrete random variable X is:

$$\mathbb{E}(X) = \sum_x xP(X = x)$$

The **Variance** of a discrete random variable X is:

$$\text{Var}(X) = \sum_x (x - \mathbb{E}(X))^2 P(X = x)$$

When the distribution P of the rv is not known (almost always the case in statistics and estimation), the mean and the variance must be estimated given the observations (samples) of the rv:

Sample mean

$$\mathbb{E}(X) \cong \mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

Sample variance

$$\mathrm{Var}(X) \cong s = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu)^2$$

Example: Coin Tosses: Matlab

scripts/rv_mean_variance.m

```
N_tosses = 100;
probability = .5; %Fair coin
X = binornd(1, probability, N_tosses, 1) %Bernouilli rv
%X =
%
% 1
% 0
% 0
% 1
% 1
% ...

mu = mean(X) %Expectation
%ans =
%
% 0.4800
sigma2 = var(X) %Sample Variance
%ans =
%
% 0.2521
```

Often the parameter of a probability distribution must be estimated from data. For example one can estimate the location and scale parameters of a Gaussian distribution, the p value of a Bernouilli distribution given data samples.

The **Maximum Likelihood** Estimator (ML) of a parameter is the value of θ that maximizes the likelihood function.

$$MLE(\theta) = \max_{\theta} P(D|\theta).$$

The MLE for each parameter p_i of the multinomial distribution can be calculated using:

$$p_i \cong \frac{\text{\# times outcome } i \text{ was observed}}{n}$$

Example: ML estimation of Normal distribution parameters

> *μ and σ are the parameters of this distribution. The Maximum Likelihood Estimator (MLE) of the parameter μ is the sample mean and the MLE of σ is the sample standard deviation s.*

Example: Dice roll (Maximum Likelihood)

> *Assume we observed the following:*
>
> $$(120, 123, 100, 78, 98, 125)$$
>
> *Then p can be calculated using an ML estimator as follows:*

scripts/multinomial_mle.m

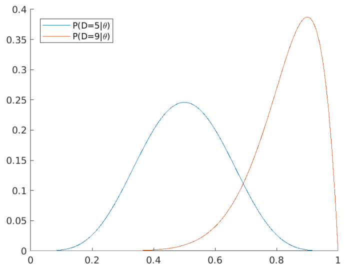
```
obs = [120, 123,100,78,98,125];
n = sum(obs);
p = obs/n
%0.1863 0.1910 0.1553 0.1211 0.1522 0.1941
```

Exercise: Test Questions (Maximum Likelihood)

> *Recall the test questions example above. We assume:*
>
> $$D \sim Bin(10, \theta)$$
>
> *Thus, the likelihood is:*
>
> $$P(D|\theta) = Bin(10, \theta)$$
>
> *We see that the ML estimate of θ is $D/10$.*
>
> ***Noet that the likelihood is often defined as a function of the parameter θ***

scripts/test_questions_mle.m

```
theta=linspace(0,1,100);
% binopdf(X,N,P)
N_questions = 10;
llD5 = binopdf(5, N_questions, theta);
llD9 = binopdf(9, N_questions, theta);
figure();
hold on;
plot(theta,llD5);
plot(theta,llD9);
legend('P(D=5|\theta)','P(D=9|\theta)','location','northwest');
%Save file at specific directory
saveas(gcf,"/home/eneftci/pd/lectures/QMCS/slides/lbayes/img/test_questions_mle.png");
```



> *Bayesian inference provides a distribution over the parameter θ.*

The **Maximum a Posteriori** Estimator (MAP) of a parameter is the value of θ that maximizes the posterior function.

$$MAP(\theta) = \max_{\theta} P(\theta|D)$$

Note that ML and MAP estimators are specific to the likelihood function and posterior distribution.

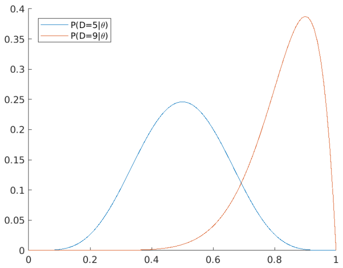
Both ML and MAP estimators throw away information about our uncertainty of the parameter θ, but can be useful in making decisions. In contrast, in Bayesian inference, the parameters can also be treated as random variables, with a distribution associated to them. This allows one to quantify our uncertainty over the estimated parameters.

Exercise: Test Questions (Maximum a Posteriori)

> *First, we find the posterior $P(\theta|D)$ using Bayes rule. First assume a uniform prior,* i.e. $P(\theta) \sim Unif(0,1)$.

scripts/test_questions_uniform.m

```
theta=linspace(0,1,100);
% binopdf(X,N,P)
N_questions = 10;
posteriorD5 = binopdf(5, N_questions, theta).*unifpdf(theta,0,1);
posteriorD9 = binopdf(9, N_questions, theta).*unifpdf(theta,0,1);
figure();
hold on;
plot(theta,posteriorD5);
plot(theta,posteriorD9);
legend(’P(D=5|\theta)’,’P(D=9|\theta)’,’location’,’northwest’);
%Save file at specific directory
saveas(gcf,”/home/eneftci/pd/lectures/QMCS/slides/lbayes/img/test_questions_uniform.png”);
```

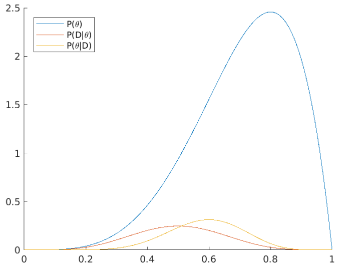


This result is the same as the likelihood function.

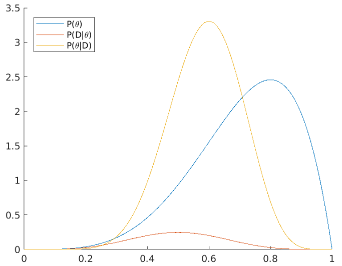
Now assume a non-uniform prior e.g. *a $Beta$ function as its domain is in $[0, 1]$. With a prior such that $P(\Theta) = Beta(5, 2)$:*

scripts/test_questions_beta_nonnorm.m

```
theta=linspace(0,1,100);
% binopdf(X,N,P)
D = 5
N_questions = 10;
ll = binopdf(D, N_questions, theta);
prior = betapdf(theta,5,2);
posterior = binopdf(D, N_questions, theta).*prior;
figure();
hold on;
plot(theta,prior);
plot(theta,ll);
plot(theta,posterior);
legend(’P(\theta)’,’P(D|\theta)’,’P(\theta|D)’,’location’,’northwest’);
%Save file at specific directory
saveas(gcf,”/home/eneftci/pd/lectures/QMCS/slides/lbayes/img/test_questions_beta_nonnorm.png”);
```



Notice how the prior and posterior distributions are no longer normalized (area under the curve is not 1). This is because we omitted the marginal likelihood for tractability. Using some more involved concepts, known as conjugate priors, it is possible in this case to calculate the distributions exactly:



Note that the likelihood function being a function of the parameter θ is not a probability distribution and is not normalized.

The MAP estimate is the maximum of $P(\theta|D)$, and around $.6$ in this example above.

The take home message here is that Bayesian inference provides a distribution over the parameter θ, and priors can "bias" the distribution or the estimates of θ.

A **Decision Rule** is a function that takes data and associates a class to it.

Example: Test Questions (continued)

The instructor that decides whether the student should pass based on his ability to answers questions correctly, i.e. *θ. One possible decision rule is that the student passes if $\theta_{MAP} > .75$ (the MAP estimate of θ is larger than some threshold .75).*

Example: X-ray Examination

> *An X-ray returns an image of a patients lung. Does the patient have lung cancer or not? This decision will determine whether further examination or treatment.*

Minimizing the chance of mistake is equivalent to choosing the decision associated with the highest posterior probability, *i.e* the MAP estimate. However, some mistakes can be worse than others. For simplicity, let us assume a classification problem, i.e. the variable of interest is a class.

Example: X-ray Examination (Continued)

> *If a healthy is determined (erroneously) to have cancer, the patient loses time and gets stressed. If the patient have cancer but is deemed healthy, he or she will die. The consequences of the second type of error is much worse than the first.*

This is formalized using a **Loss** function. The Loss function is an overall measure of loss incurred in taking a decision. Typically the goal of a task is to minimize loss.

Example: X-ray Examination (Continued)

> *Assuming the second type of error incurs a loss of 1000, whereas the first type of error incures a loss of 1. We can define a loss function using the table:*

	$Healthy$	$Cancer$
$Healthy$	*0*	*1*
$Cancer$	*1000*	*0*

Thr rows correspond to the true class and columns correspond to the assignment of class made by the decision criterion.

The **expected loss** for discrete rvs B and C is:

$$\mathbb{E}(L) = \sum_k \sum_j L_{kj} p(B_j, C_k)$$

The new decision rule that minimizes the expected loss is one that assigns each new data sample D to class j, such that the following expression is minimal:

$$\sum_k L_{kj} P(C_k|D).$$

Note that the above equation is only for determining the best decision, not calculating the cost. The cost can be calculated using the expected loss formula.

Exercise: X-ray Examination (Continued)

Assuming

$$P(C = cancer|D) = .3$$

and the loss function given above. What is the chance of an error of the second type for a sick patient?

Two rvs X_1, X_2 are **conditionally independent** given a third rv X_3 if:

$$P(X_1, X_2|X_3) = P(X_1|X_3)P(X_2|X_3)$$

The **Naive Bayes** classification algorithm is a simple but effective application of Bayes rule to classification. The Naive Bayes rule assumes that each dimension of the observable $D = [d_1, d_2, ..., d_M]$ is conditionally independent given the class C_k.

Assuming we have a labeled dataset, we would like to estimate:

$$P(C_k|D)$$

Where C_k is the class. Through Bayes rule, we have:

$$P(C_k|D) = \frac{P(D|C_k)P(C_k)}{P(D)}$$

We can assume a probability distribution for $P(D|C_k)$, but fitting it will be difficult if M is large. Naive Bayes makes use of conditional independece to simplify the problem:

$$P(C_k|D) = \frac{P(C_k)\prod_{i=1}^{M} P(d_i|C_k)}{P(D)}$$

Thus, Naive Bayes assumes conditional independence on all dimensions given the class C_k:

$$P(C_k|d_1, d_2, ..., d_M) \propto P(C_k)\prod_{i=1}^{M} P(d_i|C_k)$$

where we have omitted the (intractable) marginal likelihood, which acts as a normalization term. This is not a big problem, because the number of classes is limited, we can always normalize the posterior numerically for each data sample D.

We are left with the problem of estimating $P(C_k)$ and $P(d_i|C_k)$ for each i and k. Fortunatly, we can estimate the parameters of these distributions using the methods seen so far.

Application of Naive Bayes:

1. Determine Class Priors $P(C_k)$, *e.g.* from data: $\frac{number\ of\ C_k}{number\ of\ samples}$
2. Assume a distribution model for each $P(d_i|C_k)$. Typically Gaussian or Binomial/-Multinomial
3. Estimate the parameters of the distribution of each d_i and k
4. Compute the posterior.

Matlab's fitcnb function takes care of all these steps.

Example: Breast Cancer Dataset

The Breast Cancer Dataset is a perfect example for the application of Naive Bayes. Here one D sample consists of all 30 features and the classes are $C = 1$ for malignant or $C = 0$ for begnin. To test an algorithm, the convention is to divide the datasey is two parts: one training (80% of the data), one testing (remaing). Here, we choose 500 training sample and 69 testing samples.

scripts/breast_cancer_naive_bayes_explicit.m

```
load_breastcancerwisconsin_as_table;

%Find out the number of samples
n_samples = size(breastcancerwisconsin,1);
n_train = 500;

%Prepare the training and testing sets
X_train = table2array(breastcancerwisconsin(1:n_train,3:end));
Y_train = breastcancerwisconsin.diagnosis(1:n_train);
X_test = table2array(breastcancerwisconsin(n_train:end,3:end));
Y_test = breastcancerwisconsin.diagnosis(n_train:end);

featM = X_train(Y_train=='M',:);
featB = X_train(Y_train=='B',:);

priorM = mean(Y_train=='M');
priorB = mean(Y_train=='B');

meanM = mean(featM);
meanB = mean(featB);

stdM = std(featM);
stdB = std(featB);

%Naive Bayes Formula
%The 2 at the end means product over second axis (features axis)
train_nbM = prod(normpdf(X_train,meanM,stdM),2).*priorM;
train_nbB = prod(normpdf(X_train,meanB,stdB),2).*priorB;
```

```
%normalize
resultM = train_nbM./(train_nbM+train_nbB);
resultB = train_nbB./(train_nbM+train_nbB);
```

The naive Bayes application can be greatly simplified using the fitnb function:

`scripts/breast_cancer_naive_bayes.m`

```
load_breastcancerwisconsin_as_table;

%Find out the number of samples
n_samples = size(breastcancerwisconsin,1);
n_train = 500;

%Prepare the training and testing sets
X_train = table2array(breastcancerwisconsin(1:n_train,3:end));
Y_train = breastcancerwisconsin.diagnosis(1:n_train);
X_test = table2array(breastcancerwisconsin(n_train:end,3:end));
Y_test = breastcancerwisconsin.diagnosis(n_train:end);


%Train the naive Bayes classifier
nb = fitcnb(X_train, Y_train);

%Compute Posteriors
[result_train, posterior_train, cost_train] = predict(nb,X_train);
[result_test, posterior_test, cost_test] = predict(nb,X_test);
```