Data-driven Design and Analyses of Structures and Materials (3dasm)

Lecture 1

Miguel A. Bessa | **M.A.Bessa@tudelft.nl** | Associate Professor

Introduction

What: A lecture of the "3dasm" course
Where: This notebook comes from this **repository**
Reference for entire course: Murphy, Kevin P. *Probabilistic machine learning: an introduction*. MIT press, 2022. Available online **here**
How: We try to follow Murphy's book closely, but the sequence of Chapters and Sections is different. The intention is to use notebooks as an introduction to the topic and Murphy's book as a resource.

- If working offline: Go through this notebook and read the book.
- If attending class in person: listen to me (!) but also go through the notebook in your laptop at the same time. Read the book.
- If attending lectures remotely: listen to me (!) via Zoom and (ideally) use two screens where you have the notebook open in 1 screen and you see the lectures on the other. Read the book.

OPTION 1. Run this notebook **locally in your computer**:

1. install miniconda3 **here**
2. Create a virtual environment for this course called "3dasm":

```
conda create -n 3dasm python=3 numpy scipy jupyter nb_conda matplotlib pandas scikit-learn
rise tensorflow -c conda-forge
```

3. clone the repository to your computer:

```
git clone https://github.com/bessagroup/3dasm_course
```

4. load jupyter notebook (it will open in your internet browser):

```
conda activate 3dasm
jupyter notebook
```

5. Open notebook (3dasm_course/Lectures/Lecture1/3dasm_Lecture1.ipynb)

OPTION 2. Use **Google's Colab** (no installation required, but times out if idle):

1. go to **https://colab.research.google.com**
2. login
3. File > Open notebook
4. click on Github (no need to login or authorize anything)
5. paste the git link: **https://github.com/bessagroup/3dasm_course**
6. click search and then click on the notebook (*3dasm_course/Lectures/Lecture1/3dasm_Lecture1.ipynb*)

In [1]:

```
# Basic plotting tools needed in Python.

import matplotlib.pyplot as plt # import plotting tools to create figures
import numpy as np # import numpy to handle a lot of things!

%config InlineBackend.figure_format = "retina" # render higher resolution images in the notebook
plt.style.use("seaborn") # style for plotting that comes from seaborn
plt.rcParams["figure.figsize"] = (8,4) # rescale figure size appropriately for slides
```

Outline for today

- Introduction
    - Taking a probabilistic perspective on machine learning
- Basics of univariate statistics
    - Continuous random variables
    - Probabilities vs probability densities
    - Moments of a probability distribution
- The mindblowing Bayes' rule
    - The rule that spawns almost every ML model (even when we don't realize it)

Reading material: This notebook + Chapter 2 until Section 2.3

Get hyped about Artificial Intelligence...

In [2]:

```
from IPython.display import display, YouTubeVideo, HTML
YouTubeVideo('RNnZwvklwa8', width=512, height=288) # show that slides are interactive:
                                                   # rescale video to 768x432 and back to 512x288
```

Out[2]:

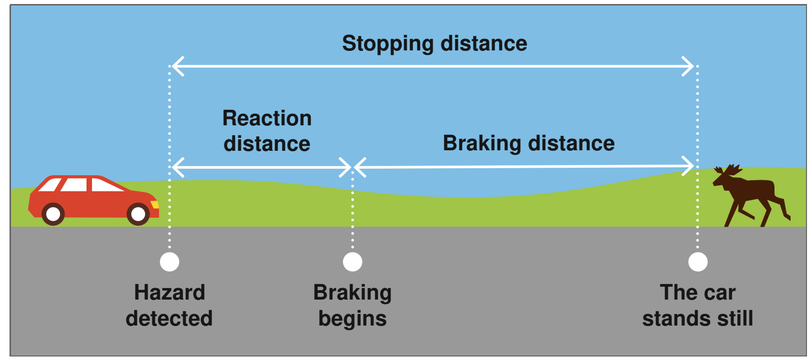
Well... This class *might* not make you break the world (yet!). Let's focus on the fundamentals:

- Probabilistic perspective on machine learning
- Supervised learning (especially regression)

Machine learning (ML)

- **ML definition**: A computer program that learns from experience E wrt tasks T such that the performance P at those tasks improves with experience E.
- We'll treat ML from a **probabilistic perspective**:
    - Treat all unknown quantities as **random variables**
- What are random variables?
    - Variables endowed with probability distributions!

The car stopping distance problem



Car stopping distance y as a function of its velocity x before it starts braking:

$y = zx + \frac{1}{2\mu g}x^2 = zx + 0.1x^2$

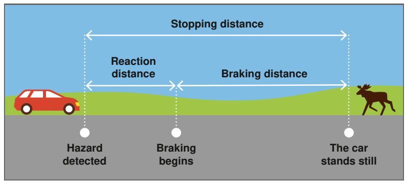
- z is the driver's reaction time (in seconds)
- μ is the road/tires coefficient of friction (assume $\mu = 0.5$)
- g is the acceleration of gravity (assume $g = 10$ m/s^2).

The car stopping distance problem

$y = {\color{blue}z}x + 0.1x^2$

The driver's reaction time ${\color{blue}z}$ is a **random variable (rv)**

- Every driver has its own reaction time z
- Assume the distribution associated to z is Gaussian with **mean** $\mu_z = 1.5$ seconds and **variance** $\sigma_z^2 = 0.5$ seconds^2



$$z \sim \mathcal{N}(\mu_z = 1.5, \sigma_z^2 = 0.5^2)$$

where $\sim$ means "sampled from", and $\mathcal{N}$ indicates a Gaussian **probability density function (pdf)**

Univariate Gaussian pdf

The gaussian pdf is defined as:

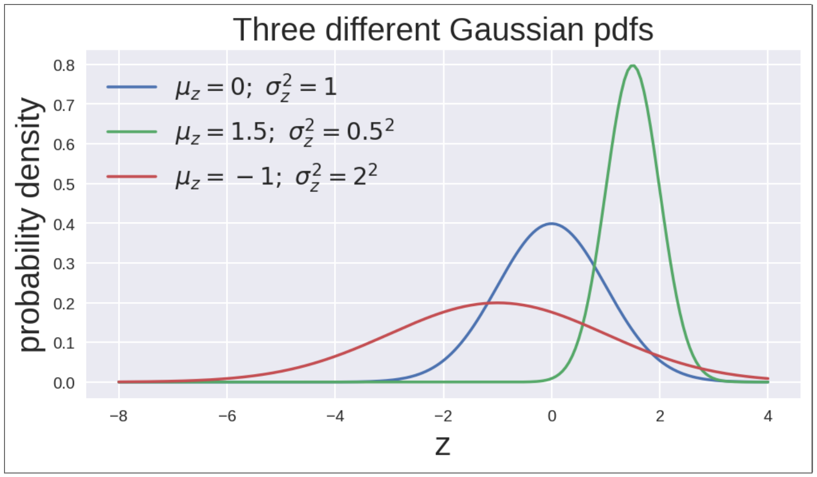
$$\mathcal{N}(z|\mu_z, \sigma_z^2) = \frac{1}{\sqrt{2\pi\sigma_z^2}} e^{-\frac{1}{2\sigma_z^2}(z-\mu_z)^2}$$

Alternatively, we can write it using the **precision** term $\lambda_z := 1/\sigma_z^2$ instead of using σ_z^2:

$$\mathcal{N}(z|\mu_z, \lambda_z^{-1}) = \frac{\lambda_z^{1/2}}{\sqrt{2\pi}} e^{-\frac{\lambda_z}{2}(z-\mu_z)^2}$$

Anyway, recall how this pdf looks like...

In [3]:
```
def norm_pdf(z, mu_z, sigma_z2): return 1 / np.sqrt(2 * np.pi * sigma_z2) * np.exp(-(z - mu_z)**2 / (2 * sigma_z2))
zrange = np.linspace(-8, 4, 200) # create a list of 200 z points between z=-8 and z=4
fig, ax = plt.subplots() # create a plot
ax.plot(zrange, norm_pdf(zrange, 0, 1), label=r"$\mu_z=0; \ \sigma_z^2=1$") # plot norm_pdf(z|0,1)
ax.plot(zrange, norm_pdf(zrange, 1.5, 0.5**2), label=r"$\mu_z=1.5; \ \sigma_z^2=0.5^2$") # plot norm_pdf(z|1.5,0.5^2)
ax.plot(zrange, norm_pdf(zrange, -1, 2**2), label=r"$\mu_z=-1; \ \sigma_z^2=2^2$") # plot norm_pdf(z|-1,2^2)
ax.set_xlabel("z", fontsize=20) # create x-axis label with font size 20
ax.set_ylabel("probability density", fontsize=20) # create y-axis label with font size 20
ax.legend(fontsize=15) # create legend with font size 15
ax.set_title("Three different Gaussian pdfs", fontsize=20); # create title with font size 20
```



The green curve shows the Gaussian pdf of the rv z **conditioned** on the mean $\mu_z = 1.5$ and variance $\sigma_z^2 = 0.5$ for the car stopping distance problem.

Univariate Gaussian pdf

$$p(z) = \mathcal{N}(z|\mu_z, \sigma_z^2) = \frac{1}{\sqrt{2\pi\sigma_z^2}} e^{-\frac{1}{2\sigma_z^2}(z-\mu_z)^2}$$

The output of this expression is the **PROBABILITY DENSITY** of z **given** (or conditioned to) a particular μ_z and σ_z^2.

- **Important**: Probability Density $\neq$ Probability

So, what is a probability?

Probability

The probability of an event A is denoted by $\Pr(A)$.

- $\Pr(A)$ means the probability with which we believe event A is true
- An event A is a binary variable saying whether or not some state of the world holds.

Probability is defined such that: $0 \leq \Pr(A) \leq 1$
where $\Pr(A) = 1$ if the event will definitely happen and $\Pr(A) = 0$ if it definitely will not happen.

Joint probability

Joint probability of two events: $\Pr(A \wedge B) = \Pr(A, B)$
If A and B are **independent**: $\Pr(A, B) = \Pr(A)\Pr(B)$
For example, suppose z_1 and z_2 are chosen uniformly at random from the set $\mathcal{Z} = \{1, 2, 3, 4\}$.
Let A be the event that $z_1 \in \{1, 2\}$ and B be the event that $z_2 \in \{3\}$.
Then we have: $\Pr(A, B) = \Pr(A)\Pr(B) = \frac{1}{2} \cdot \frac{1}{4}$.

Probability of a union of two events

Probability of event A or B happening is:
$\Pr(A \vee B) = \Pr(A) + \Pr(B) - \Pr(A \wedge B)$
If these events are mutually exclusive (they can't happen at the same time):

$$\Pr(A \vee B) = \Pr(A) + \Pr(B)$$

For example, suppose an rv denoted as z_1 is chosen uniformly at random from the set $\mathcal{Z} = \{1, 2, 3, 4\}$.
Let A be the event that $z_1 \in \{1, 2\}$ and B be the event that anoter rv denoted as $z_2 \in \{3\}$.
Then we have $\Pr(A \vee B) = \frac{2}{4} + \frac{1}{4}$.

Conditional probability of one event given another

We define the **conditional probability** of event B happening given that A has occurred as follows:

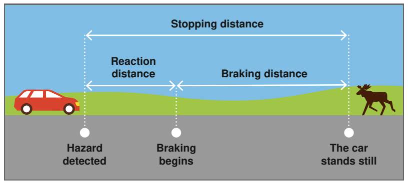
$$\Pr(B|A) = \frac{\Pr(A,B)}{\Pr(A)}$$

This is not defined if $\Pr(A) = 0$, since we cannot condition on an impossible event.

Conditional independence of one event given another

We say that event A is conditionally independent of event B if we have
$\Pr(A|B) = \Pr(A)$
This implies $\Pr(B|A) = \Pr(B)$. Hence, the joint probability becomes
$\Pr(A,B) = \Pr(A)\Pr(B)$
The book uses the notation $A \perp B$ to denote this property.

Coming back to our car stopping distance problem



$y = z x + 0.1 x^2$
where z is a **continuous** rv such that
$z \sim \mathcal{N}(\mu_z = 1.5, \sigma_z^2 = 0.5^2)$.

- What is the probability of an event Z defined by a reaction time $z \leq 0.52$ seconds?

$$\Pr(Z) = \Pr(z \leq 0.52) = P(z = 0.52)$$

where $P(z)$ denotes the **cumulative distribution function (cdf)**. Note that cdf is denoted with a capital P.

Likewise, we can compute the probability of being in any interval as follows:
$\Pr(a \leq z \leq b) = P(z = b) - P(z = a)$

- But how do we compute the cdf at a particular value b, e.g. $P(z = b)$?

Cdf's result from pdf's

A pdf $p(z)$ is defined as the derivative of the cdf $P(z)$:

$$p(z) = \frac{d}{dz} P(z)$$

So, given a pdf $p(z)$, we can compute the following probabilities:

$$\Pr(z \leq b) = \int_{-\infty}^{b} p(z) dz = P(b)$$

$$\Pr(z \geq a) = \int_{a}^{\infty} p(z) dz = 1 - P(a)$$

$$\Pr(a \leq z \leq b) = \int_{a}^{b} p(z) dz = P(b) - P(a)$$

IMPORTANT: $\int_{-\infty}^{\infty} p(z) dz = 1$

Cdf's result from pdf's

Key point?

- Given a pdf $p(z)$, we can compute the probability of a continuous rv z being in a finite interval as follows:

$$\Pr(a \leq z \leq b) = \int_a^b p(z)dz = P(b) - P(a)$$

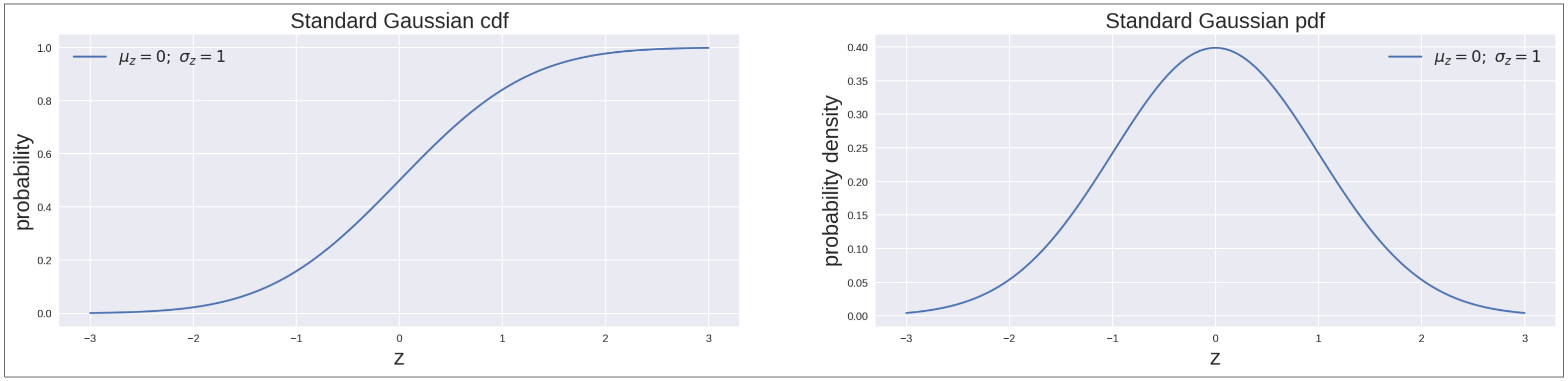
As the size of the interval gets smaller, we can write

$$\Pr\left(z - \frac{dz}{2} \leq z \leq z + \frac{dz}{2}\right) \approx p(z)dz$$

Intuitively, this says the probability of z being in a small interval around z is the density at z times the width of the interval.

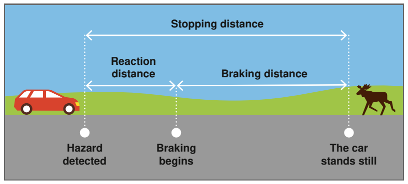
In [4]:

```python
from scipy.stats import norm # import from scipy.stats the normal distribution

zrange = np.linspace(-3, 3, 100) # 100 values for plot
fig_std_norm, (ax1, ax2) = plt.subplots(1, 2) # create a plot with 2 subplots side-by-side
ax1.plot(zrange, norm.cdf(zrange, 0, 1), label=r"$\mu_z=0; \ \sigma_z=1$") # plot cdf of standard normal
ax1.set_xlabel("z", fontsize=20)
ax1.set_ylabel("probability", fontsize=20)
ax1.legend(fontsize=15)
ax1.set_title("Standard Gaussian cdf", fontsize=20)

ax2.plot(zrange, norm.pdf(zrange, 0, 1), label=r"$\mu_z=0; \ \sigma_z=1$") # plot pdf of standard normal
ax2.set_xlabel("z", fontsize=20)
ax2.set_ylabel("probability density", fontsize=20)
ax2.legend(fontsize=15)
ax2.set_title("Standard Gaussian pdf", fontsize=20)
fig_std_norm.set_size_inches(25, 5) # scale figure to be wider (since there are 2 subplots)
```



Coming back to our car stopping distance problem

$y = zx + 0.1x^2$
where z is a continuous rv such that
$p(z) = \mathcal{N}(z|\mu_z = 1.5, \sigma_z^2 = 0.5^2)$.



- What is the probability of an event Z defined by a reaction time $z \leq 0.52$ seconds?

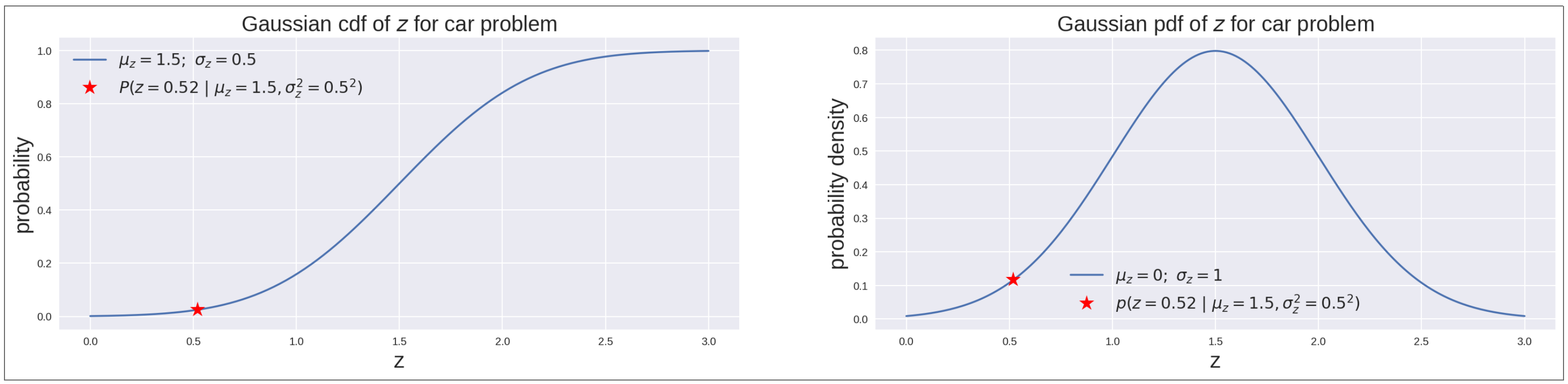
$$\Pr(Z) = \Pr(z \leq 0.52) = P(z = 0.52) = \int_{-\infty}^{0.52} p(z)dz$$

In [5]:
```
Pr_Z = norm.cdf(0.52, 1.5, 0.5) # using scipy norm.cdf(z=0.52 | mu_z=1.5, sigma_z=0.5)
print("The probability of event Z is: Pr(Z) = ",round(Pr_Z,3))
```

```
The probability of event Z is: Pr(Z) =  0.025
```

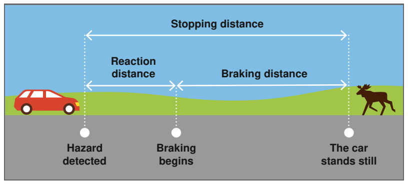
In [6]:

```
z_value = 0.52 # z = 0.52 seconds
zrange = np.linspace(0, 3, 200) # 200 values for plot
fig_car_norm, (ax1, ax2) = plt.subplots(1, 2) # create subplot (two figures in 1)
ax1.plot(zrange, norm.cdf(zrange, 1.5, 0.5), label=r"$\mu_z=1.5; \ \sigma_z=0.5$") # Figure 1 is cdf
ax1.plot(z_value, norm.cdf(z_value, 1.5, 0.5), 'r*',markersize=15, linewidth=2,
         label=u'$P(z=0.52~|~\mu_z=1.5, \sigma_z^2=0.5^2)$')
ax1.set_xlabel("z", fontsize=20)
ax1.set_ylabel("probability", fontsize=20)
ax1.legend(fontsize=15)
ax1.set_title("Gaussian cdf of $z$ for car problem", fontsize=20)
ax2.plot(zrange, norm.pdf(zrange, 1.5, 0.5), label=r"$\mu_z=0; \ \sigma_z=1$") # figure 2 is pdf
ax2.plot(z_value, norm.pdf(z_value, 1.5, 0.5), 'r*', markersize=15, linewidth=2,
         label=u'$p(z=0.52~|~\mu_z=1.5, \sigma_z^2=0.5^2)$')
ax2.set_xlabel("z", fontsize=20)
ax2.set_ylabel("probability density", fontsize=20)
ax2.legend(fontsize=15)
ax2.set_title("Gaussian pdf of $z$ for car problem", fontsize=20)
fig_car_norm.set_size_inches(25, 5) # scale figure to be wider (since there are 2 subplots)
```



Car stopping distance problem

$y = zx + 0.1x^2$
where z is a continuous rv such that
$z \sim \mathcal{N}(\mu_z = 1.5, \sigma_z^2 = 0.5^2)$.



- What is the **expected** value for the reaction time z?

This is not a trick question! It's the mean μ_z, of course!

- But how do we compute the expected value for any distribution?

Moments of a distribution

First moment: Expected value or mean

The expected value (mean) of a distribution is the **first moment** of the distribution:

$$\mathbb{E}[z] = \int_{\mathcal{Z}} zp(z)dz$$

where $\mathcal{Z}$ indicates the support of the distribution (the z domain).

- Often, $\mathcal{Z}$ is omitted as it is usually between $-\infty$ to ∞
- The expected value $\mathbb{E}[z]$ is often denoted by μ_z

Moments of a distribution

Second moment (and relation to Variance)

The 2nd moment of a distribution $p(z)$ is:

$$\mathbb{E}[z^2] = \int_{\mathcal{Z}} z^2 p(z) dz$$

Variance can be obtained from the 1st and 2nd moments

The variance is a measure of the "spread" of the distribution:

$$\mathbb{V}[z] = \mathbb{E}[(z - \mu_z)^2] = \int (z - \mu_z)^2 p(z) dz = \mathbb{E}[z^2] - \mu_z^2$$

- It is often denoted by the square of the standard deviation, i.e. $\sigma_z^2 = \mathbb{V}[z] = \mathbb{E}[(z - \mu_z)^2]$

Variance and standard deviation properties

The standard deviation is defined as

$\sigma_z = \mathrm{std}[z] = \sqrt{\mathbb{V}[z]}$

The variance of a shifted and scaled version of a random variable is given by

$\mathbb{V}[az + b] = a^2 \mathbb{V}[z]$

where a and b are discrete values (NOT rv's).

Mode of a distribution

The mode of an rv z is the value of z for which $p(z)$ is maximum.
Formally, this is written as,

$$\mathbf{z}^* = \operatorname*{argmax}_z p(z)$$

If the distribution is multimodal, this may not be unique:

- That's why $\mathbf{z}^*$ is in **bold**, to denote that in general it is a vector that is retrieved!
- However, if the distribution is unimodal (one maximum), like the univariate Gaussian distribution, then it retrieves a scalar z^*

Note that even if there is a unique mode, this point may not be a good summary of the distribution.

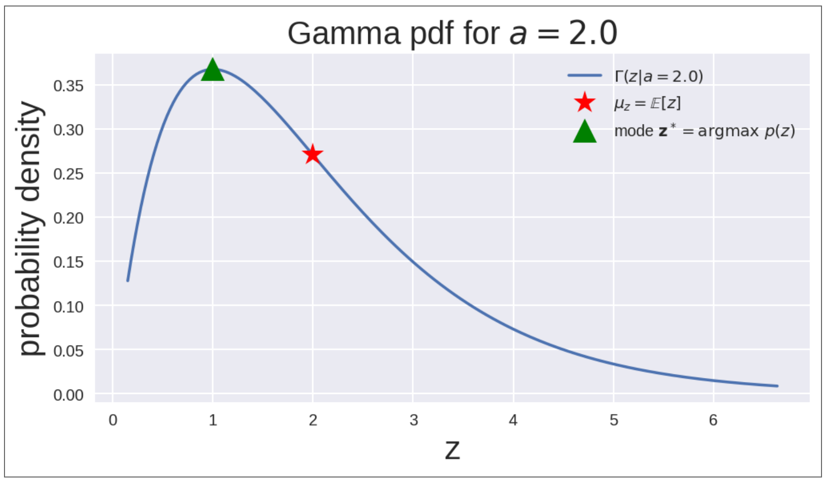
Mean vs mode for a non-symmetric distribution

In [10]:

```
            # Code to generate this Gamma distribution hidden during presentation (it's shown as notes)

print('The mean is ',mu_z) # print the mean calculated for this gamma pdf
print('The mode is approximately ',mode_z) # print the mode
fig_gamma_pdf # show figure of this gamma pdf
```

```
The mean is  2.0
The mode is approximately  [1.00001618]
```

Out[10]:



The amazing Bayes' rule

Bayesian inference definition:

- Inference means “the act of passing from sample data to generalizations, usually with calculated degrees of certainty”.
- Bayesian is used to refer to inference methods that represent “degrees of certainty” using probability theory, and which leverage Bayes' rule to update the degree of certainty given data.

Bayes' rule is a formula for computing the probability distribution over possible values of an unknown (or hidden) quantity z given some observed data y:

$$p(z|y) = \frac{p(y|z)p(z)}{p(y)}$$

Bayes' rule follows automatically from the identity:
$p(z|y)p(y) = p(y|z)p(z) = p(y,z) = p(z,y)$

The amazing Bayes' rule

- I know... You don't find it very amazing (yet!).
- Wait until you realize that almost all ML methods can be derived from this simple formula

$$p(z|y) = \frac{p(y|z)p(z)}{p(y)}$$

See you next class

Have fun!