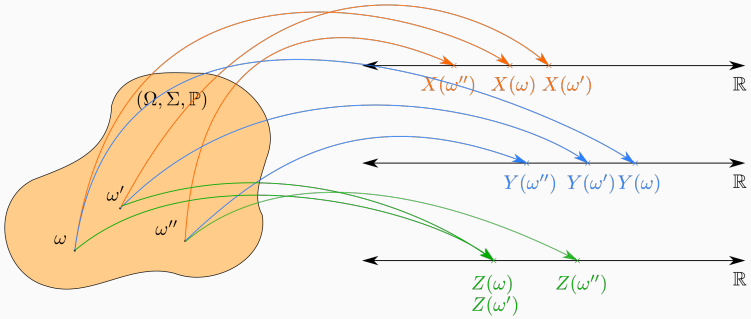
# Probabilistic Inference (Part 2)

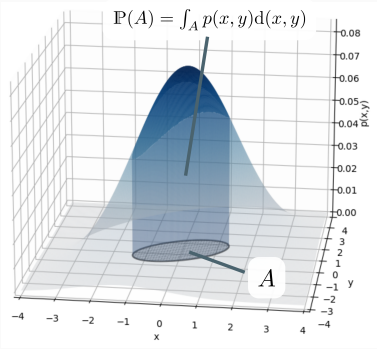
Probabilistic Decision Making — Lecture 4

22nd October 2025

Robert Peharz
Institute of Machine Learning and Neural Computation
Graz University of Technology




| $x_1$ | $x_2$ | $x_3$ | $p(x_1, x_2, x_3)$ |
|---|---|---|---|
| 0 | 0 | 0 | 0.2475 |
| 0 | 0 | 1 | 0.0025 |
| 0 | 1 | 0 | 0.125 |
| 0 | 1 | 1 | 0.125 |
| 1 | 0 | 0 | 0.125 |
| 1 | 0 | 1 | 0.125 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0.25 |

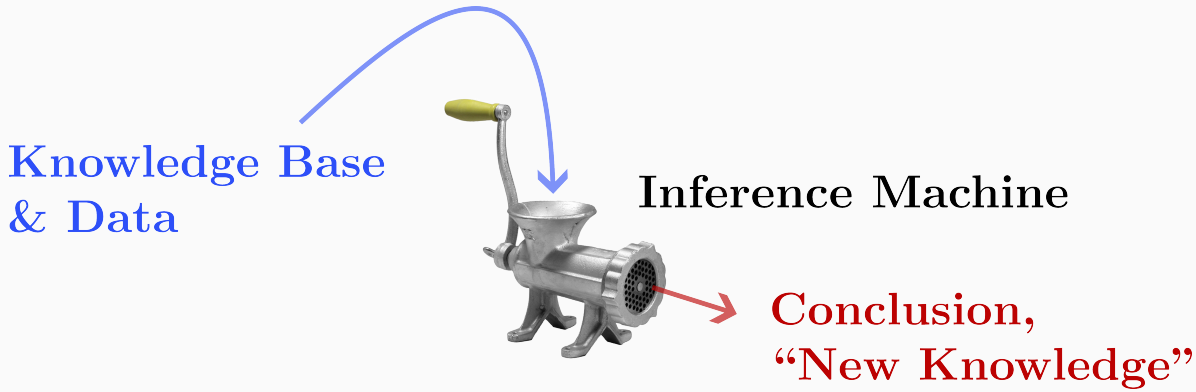



**Joint probability mass function** (PMF) $p_{\boldsymbol{X}} : \boldsymbol{\mathcal{X}} \mapsto [0, 1]$ (discrete RVs):

$$p_{\boldsymbol{X}}(\boldsymbol{x}) \coloneqq \mathbb{P}_{\boldsymbol{X}}(\{\boldsymbol{x}\})$$

**Joint probability density function** (PDF) $p_{\boldsymbol{X}} : \mathbb{R}^D \mapsto [0, \infty]$ (continuous RVs):

$$\mathbb{P}_{\boldsymbol{X}}(A) = \int_A p_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$



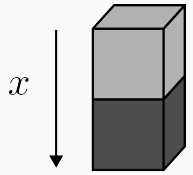

- **probability = reasoning under uncertainty**
- **joint distribution = knowledge base + uncertainty**
- **inference machine = probabilistic inference**
- most important **inference routines**:
  - **marginalization** (**sum rule**) $p(y) = \int_z p(y, z)\mathrm{d}z$
  - **conditioning** (**product rule**) $p(y \mid z) = \frac{p(y,z)}{p(z)}$

# PMFs as D-Dimensional Tensors

**Example**

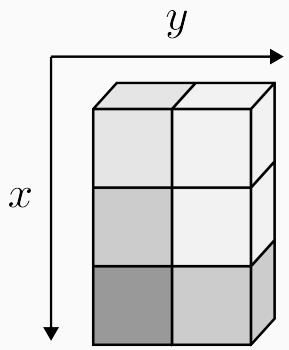
A helpful visualization for joint PMFs: **D-dimensional arrays** or **tensors**

| $x$ | $p(x)$ |
|---|---|
| 0 | 0.3 |
| 1 | 0.7 |



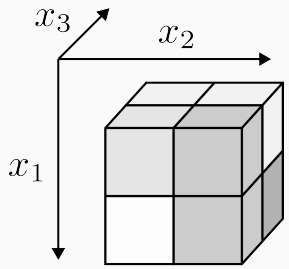

Vector

| $x$ | $y$ | $p(x, y)$ |
|---|---|---|
| 0 | 0 | 0.1 |
| 0 | 1 | 0.05 |
| 1 | 0 | 0.2 |
| 1 | 1 | 0.05 |
| 2 | 0 | 0.4 |
| 2 | 1 | 0.2 |




Matrix

| $x_1$ | $x_2$ | $x_3$ | $p(x_1, x_2, x_3)$ |
|---|---|---|---|
| 0 | 0 | 0 | 0.1 |
| 0 | 0 | 1 | 0.05 |
| 0 | 1 | 0 | 0.2 |
| 0 | 1 | 1 | 0.05 |
| 1 | 0 | 0 | 0.01 |
| 1 | 0 | 1 | 0.09 |
| 1 | 1 | 0 | 0.2 |
| 1 | 1 | 1 | 0.3 |



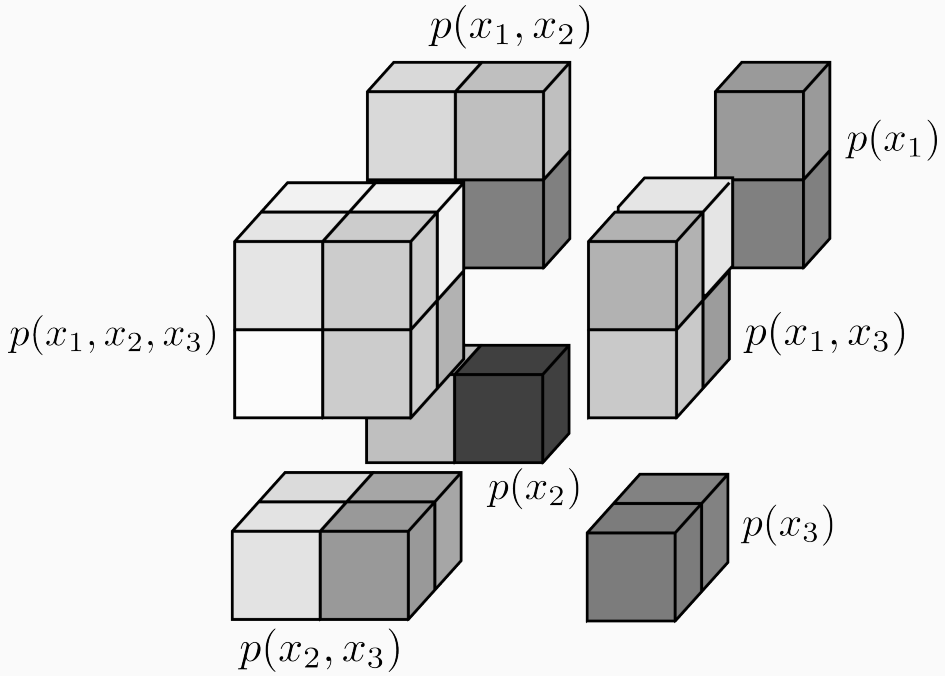

Tensor

$\vdots$

PDFs might be figured as a continuous generalization of discrete tensors.

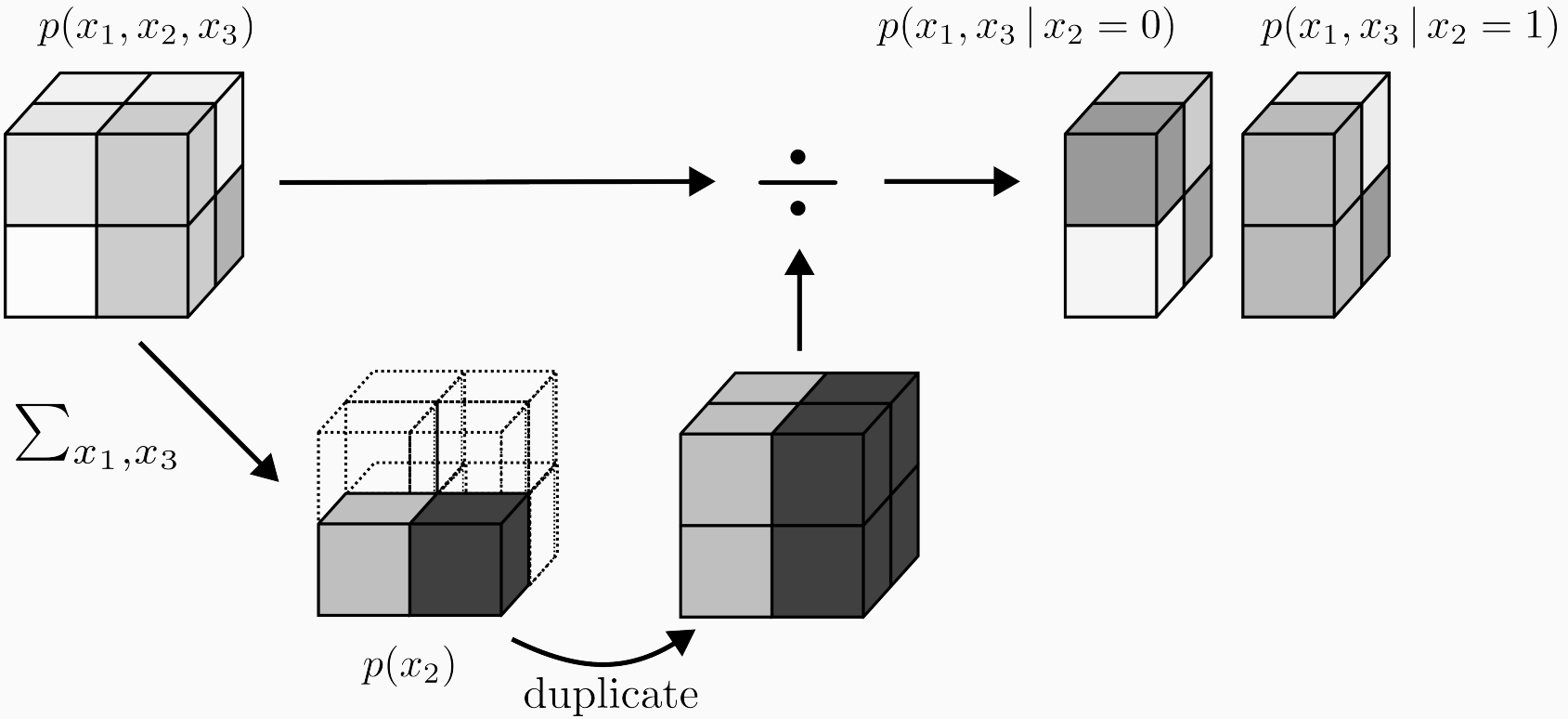
# Marginalization: Summing over Axes

Example




**For PMFs, marginalization is just a summation over tensor axes.**
For example, in `NumPy` this is done with `p.sum(axes)`, where `p` is the joint PMF as $d$-dimensional `NumPy`-array and `axes` are the indices of the dimensions you want to marginalize.

For PDFs, summation is replaced by integration, but the same picture applies.




**Conditionals can be visualized in a similar manner.**

In `NumPy`, this is done with `p / p.sum(axes, keepdims=True)` where `p` is the joint PMF as `NumPy`-array and `axes` are the indices of the dimensions we **don't** condition on ($X_1$ and $X_3$ above).

# Semantics of Marginalization and Conditioning

Probabilistic reasoning is actually remarkably simple, using just **two core inference patterns.** Their semantics are:

**Marginalization** (**Sum Rule**)

- **ignore**
- **forget**
- **account for unknowns**

**Conditioning** (**Product Rule**)

- **inject information into the model**
- **update information**

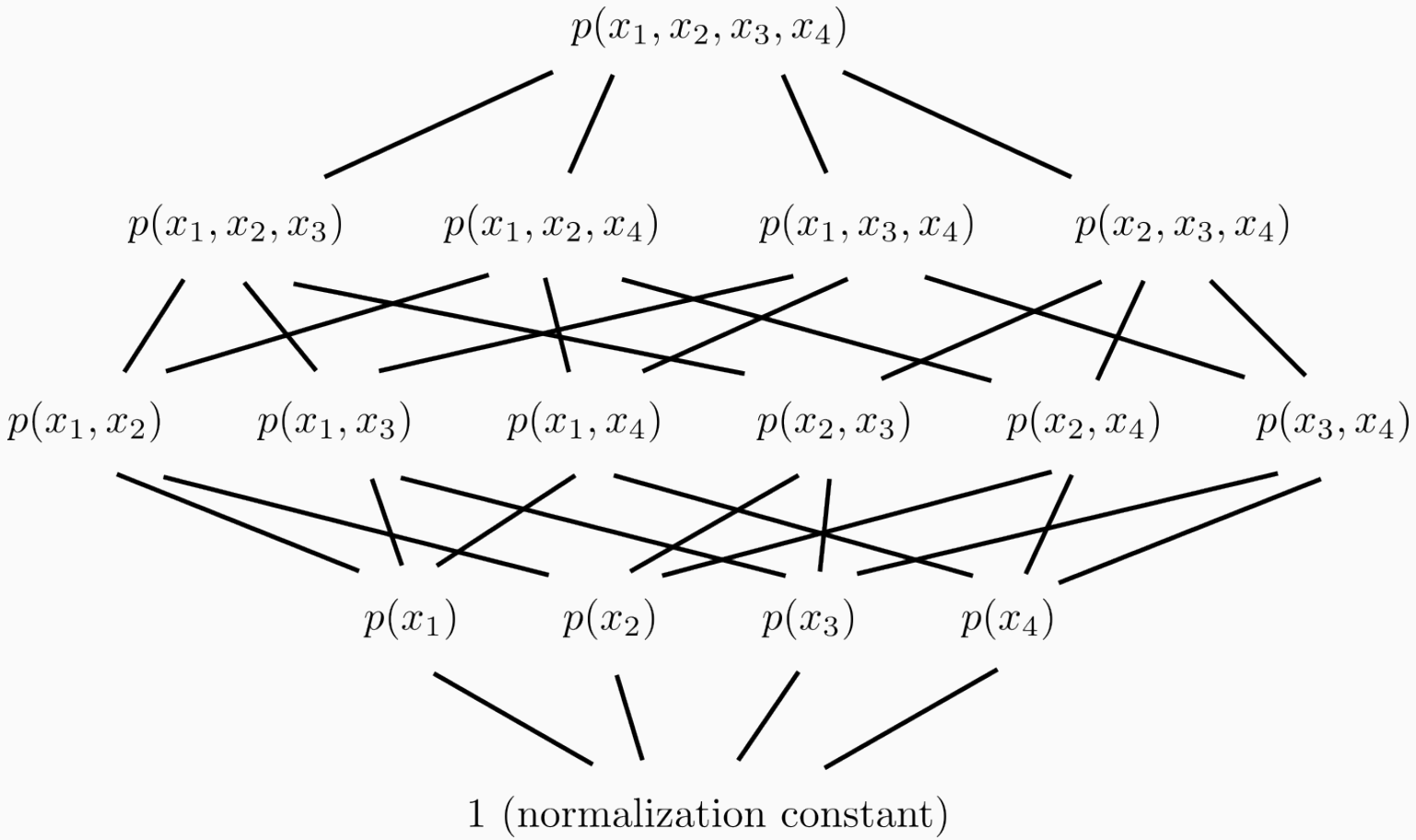
Crucially, they form a **consistent reasoning process**.

- say we start with a joint distribution $p(x_1, x_2, x_3, x_4)$ and are interested in the marginal $p(x_2, x_4)$
- we could
  - first marginalize $X_1$, yielding $p(x_2, x_3, x_4)$ and
  - then marginalize $X_3$, yielding $p(x_2, x_4)$

  or
  - first marginalize $X_3$, yielding $p(x_1, x_2, x_4)$ and
  - then marginalize $X_1$, yielding $p(x_2, x_4)$
- Does the order matter? **No, we will get exactly the same and correct marginal $p(x_2, x_4)$!**
- **consistent reasoning: two formally equivalent reasoning paths should deliver the same result**
- obviously we want consistency—yet, some classical AI systems used to violate this principle! ([Pearl 1988])

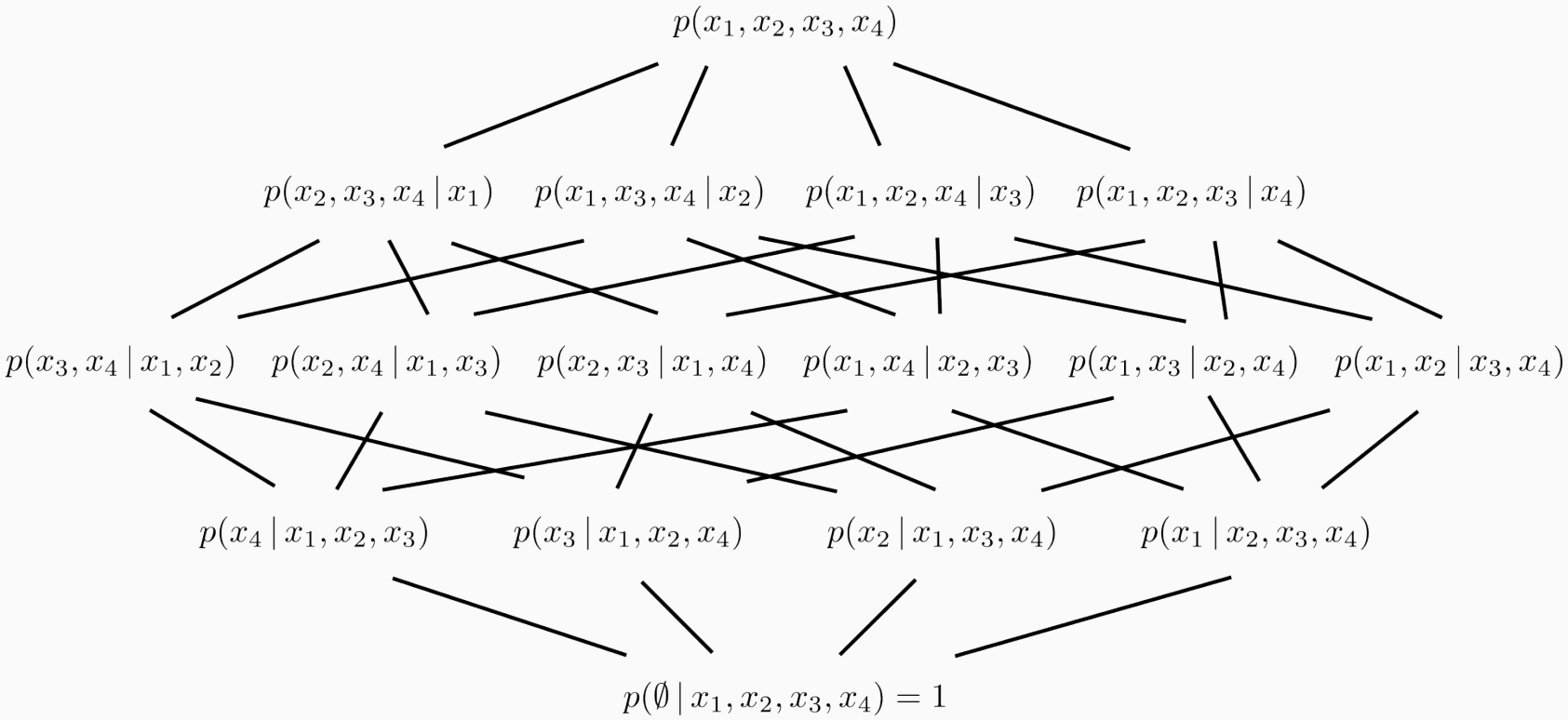
# Marginalization Lattice

No matter in which order we marginalize, we arrive at the same marginal:



# Conditioning Lattice

Similarly, **conditioning** is also a consistent reasoning process. No matter in which order we condition, we arrive at the same conditional:



# Marginalization and Conditioning Play Together

**Marginalization and conditioning can be arbitrarily interleaved.**

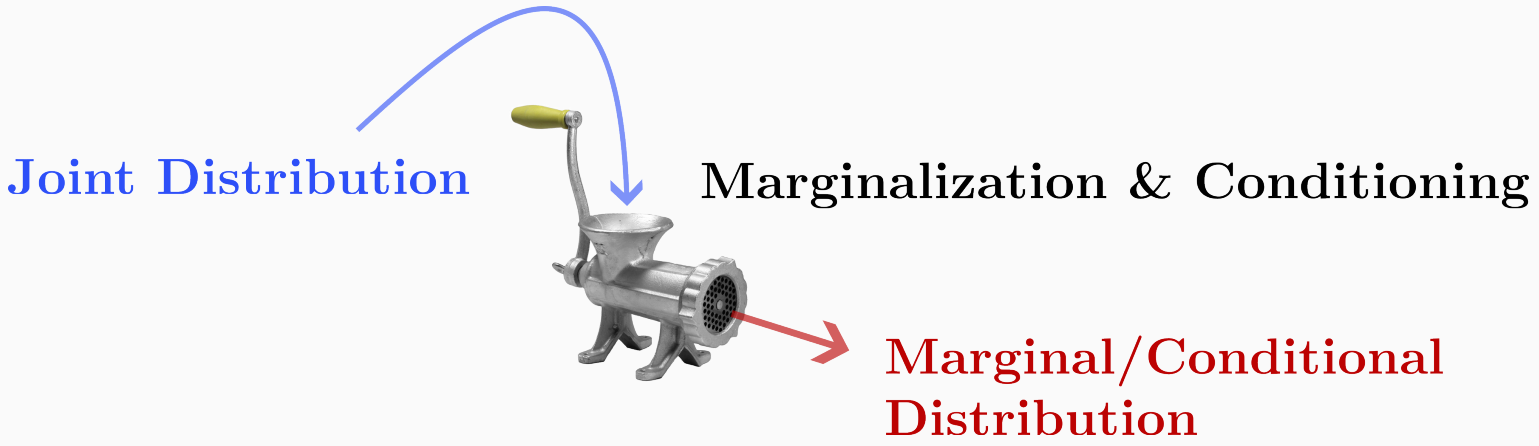
For example, if we start with a joint $p(x_1, x_2, x_3, x_4)$ and are interested in $p(x_1 \mid x_3)$, we could

- first condition on $X_3$, then marginalize $X_2$ and $X_4$
- first marginalize $X_2$, then condition on $X_3$, then marginalize $X_4$
- marginalize $X_2$ and $X_4$, then condition on $X_3$

# The Probabilistic Nature of Machine Learning

**Basically everything we do in machine learning can be seen through a probabilistic inference lens.**

- data is assumed to be generated from an unknown **data generating joint distribution**
- questions we have about the data can be formulated as probabilistic inference about this distribution
- spoiler: Bayesian inference just includes **model parameters** into this reasoning process



Say our task is to predict whether an image ($\boldsymbol{x}$) contains a cat ($y = 0$) or a dog ($y = 1$), i.e. **classification**.

**Ubiquitous assumption:** Samples are produced by some **unknown** **data generating** distribution $p^*$.

Usually we assume that samples are **independent and identically distributed (iid)**:

$$y, \boldsymbol{x} \overset{iid}{\sim} p^*(y, \boldsymbol{x})$$

We'll talk about independence in a few slides.

If we had access to $p^*$ (we usually don't) we could compute

$$p^*(y \mid \boldsymbol{x}) = \frac{p^*(y, \boldsymbol{x})}{p^*(\boldsymbol{x})} = \frac{p^*(y, \boldsymbol{x})}{\sum_y p^*(y, \boldsymbol{x})}$$

Given $\boldsymbol{x}$, a natural classification rule is

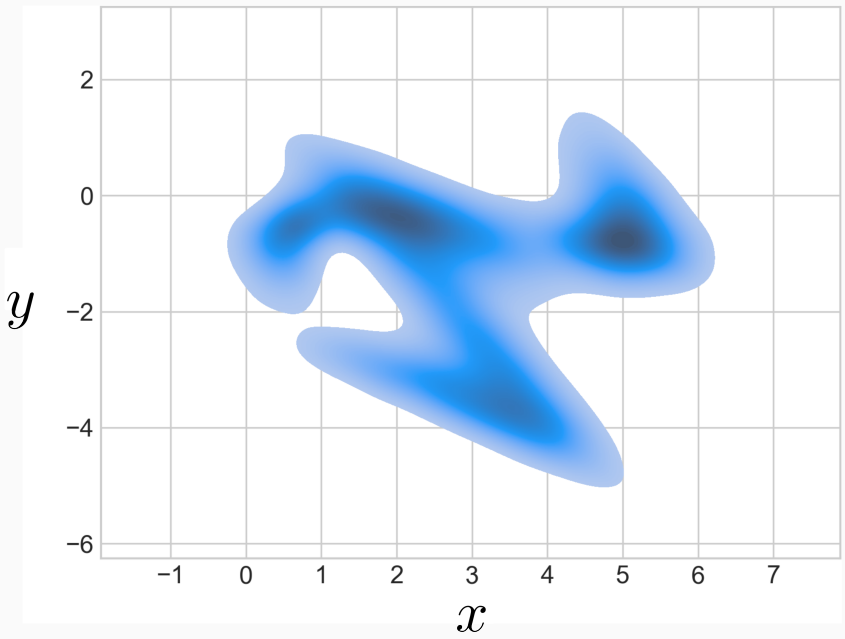
$$\hat{y}(\boldsymbol{x}) = \arg\max_y p^*(y \mid \boldsymbol{x}) = \arg\max_y \overbrace{p^*(y, \boldsymbol{x})}^{\text{proportional to } p^*(y \mid \boldsymbol{x})}$$

This is known as the **Bayes optimal classifier**—**No classifier has a higher accuracy!**

**Knowledge of data distribution + probabilistic inference leads to the theoretically perfect classifier!** ML approximates $p^*(y \mid \boldsymbol{x})$ by learning some $p_{\text{model}}(y \mid \boldsymbol{x})$ on finitely many samples.

Say we want to predict a continuous $Y$ from a continuous $X$, generated iid by the true distribution $p^*(y, x)$:

$$y, x \overset{iid}{\sim} p^*(y, x)$$

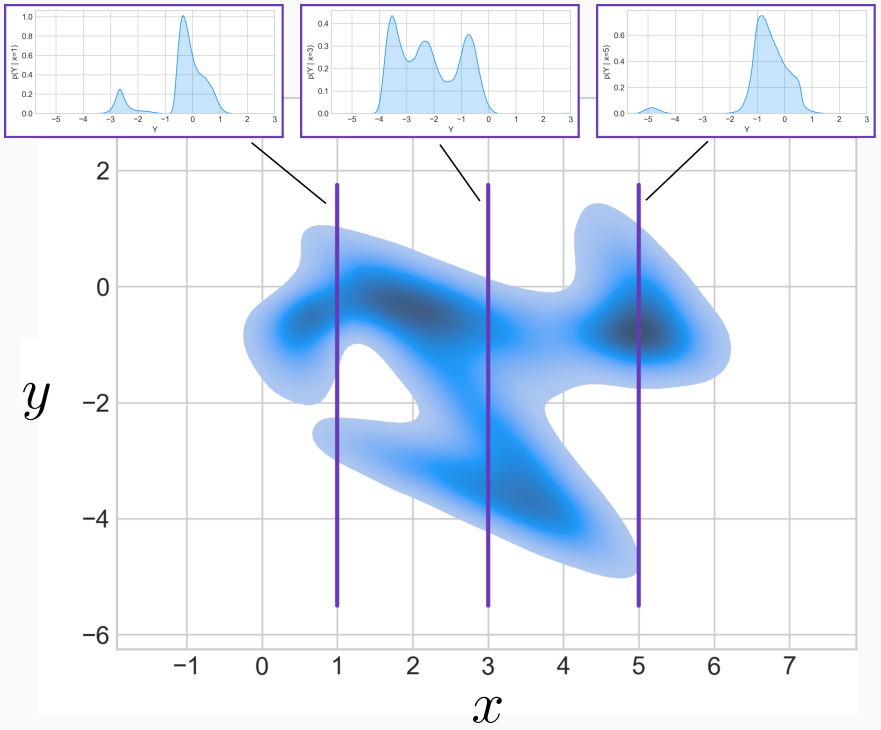


If we knew $p^*$ (again, we usually don't) we could compute the conditional distribution

$$p^*(y \mid x) = \frac{p^*(y, x)}{\underbrace{p^*(x)}_{= \int p^*(y,x)\,\mathrm{d}y}}$$

for all values of $x$.

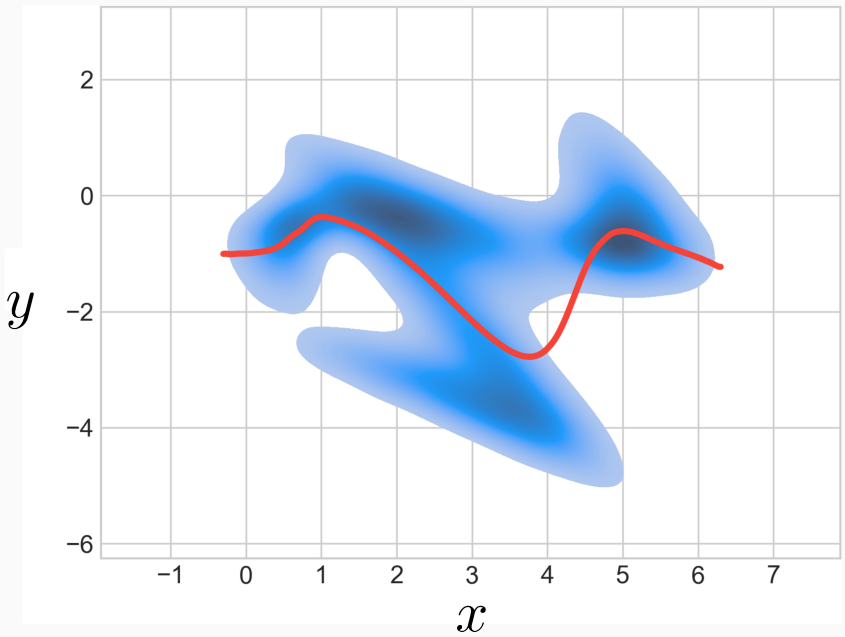
**The conditional distribution is our prediction of $Y$, including uncertainty!**



If we want a "hard" prediction, we can take the expectation of $Y$ given $x$:

$$f^*(x) := \mathbb{E}_{p^*}[Y \mid x]$$

The function defined that way is called the **true regression function**—**No function has lower squared loss.**

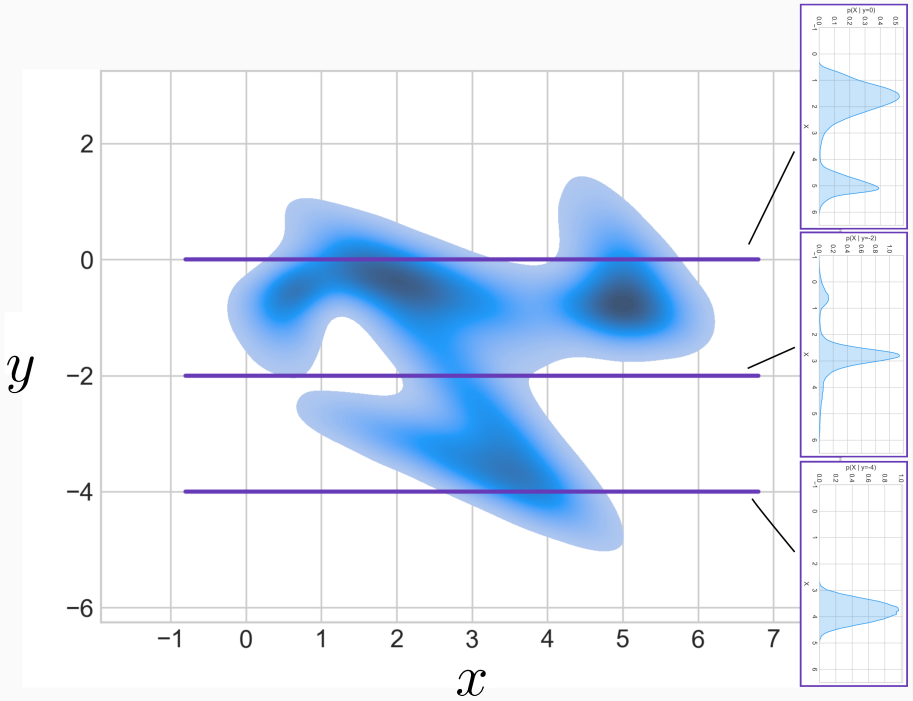



ML approximates $f^*$ by learning $f_{\text{model}}$ on finite data.

We'll talk about expectations in a few slides.

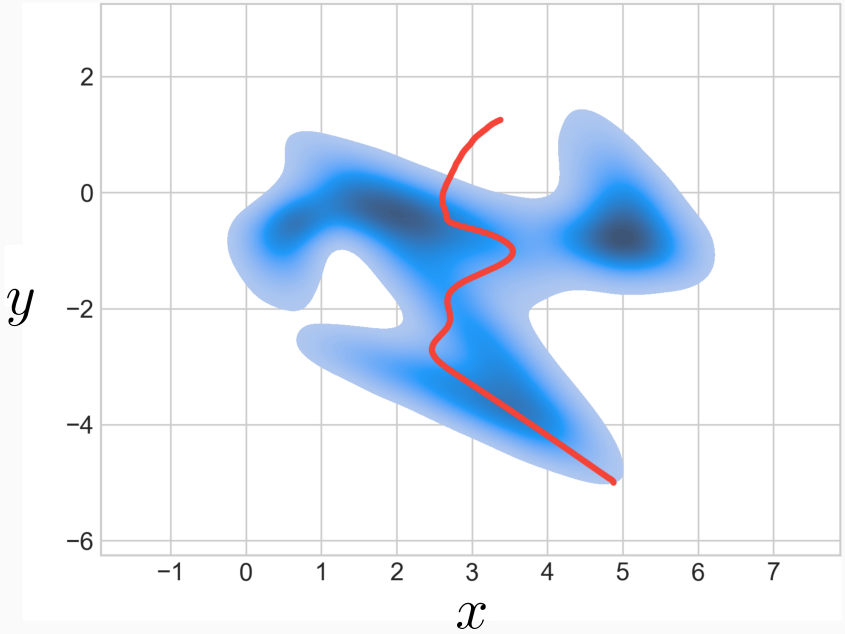
Of course, that $Y$ is output and $X$ is input was arbitrary, so the whole "trick" works also in the other direction:

$$p^*(x \mid y) = \frac{p^*(y,x)}{\underbrace{p^*(y)}_{=\int p^*(y,x)\,\mathrm{d}x}}$$



Of course, that $Y$ is output and $X$ is input was arbitrary, so the whole "trick" works also in the other direction:

$$f(y) := \mathbb{E}_{p^*}[X \mid y]$$




**true data distribution + probabilistic inference**
**= optimal omni-directional prediction!**

# Expectations

# Expectations

**Expectations** are **very** important tools in probabilistic ML. Basically, we can think of them as the **third inference routine**, next to marginalization and conditioning.

**Intuition:**

- **marginalization and conditioning** are "optimal information conversion tools"—they just **transform distributions into other distributions** and "keep track of the uncertainty"
- **expectations** transfer us "back into the deterministic world"—**they convert "fluffy distributions" into "hard numbers"**

The **expectation** or **expected value** of RV $X$ is defined as

$$\mathbb{E}[X] = \begin{cases} \sum_x p(x)\, x & \text{if } X \text{ is discrete,} \\ \int_x p(x)\, x \, \mathrm{d}x & \text{if } X \text{ is continuous.} \end{cases}$$

The same definition is of course used for **multivariate RVs:**

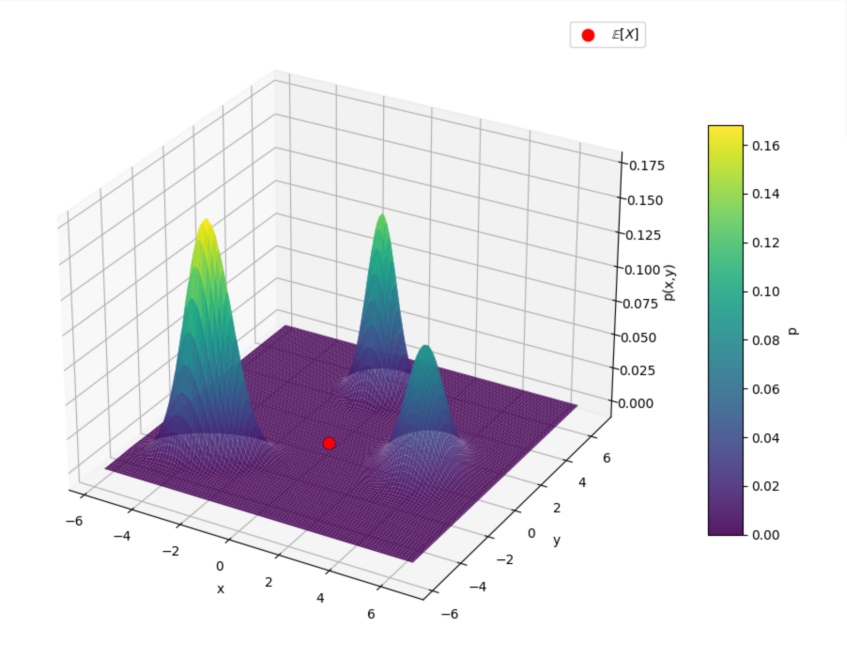
$$\mathbb{E}[\boldsymbol{X}] = \begin{cases} \sum_{\boldsymbol{x}} p(\boldsymbol{x})\, \boldsymbol{x} & \text{if } \boldsymbol{X} \text{ is discrete,} \\ \int_{\boldsymbol{x}} p(\boldsymbol{x})\, \boldsymbol{x} \, \mathrm{d}\boldsymbol{x} & \text{if } \boldsymbol{X} \text{ is continuous.} \end{cases}$$

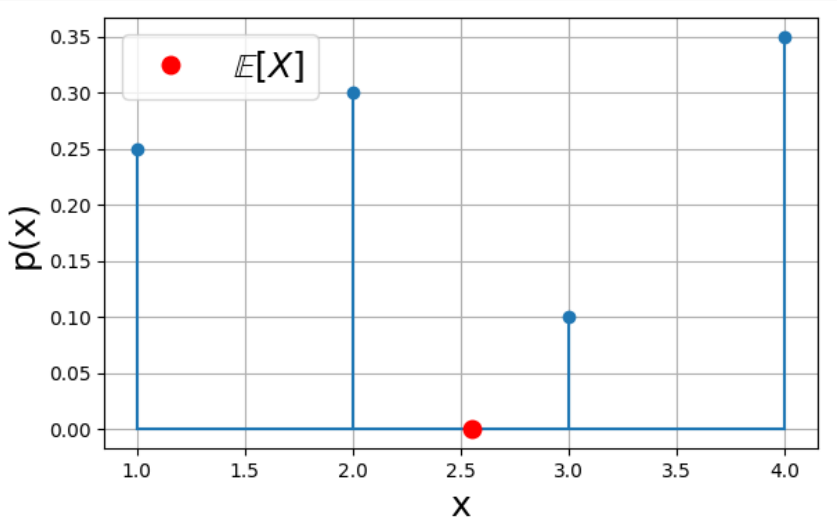
The sums and integrals run over the entire state space of $X$ ($\boldsymbol{X}$).

# Expectation

Example

- **deterministic number** assigned to an RV
- summary of distribution, **weighted average over the whole state space**
- **not necessarily in an area of high probability**
- **does not even need to be in the state space itself**



More generally, let $\boldsymbol{X}$ be a RV and $g$ be a function defined on the state space of $\boldsymbol{X}$. The **expected value** of $g(\boldsymbol{X})$ is

$$\mathbb{E}_{\boldsymbol{X}}[g(\boldsymbol{X})] = \begin{cases} \sum_x p_{\boldsymbol{X}}(x)\, g(x) & \text{if } \boldsymbol{X} \text{ is discrete,} \\ \int p_{\boldsymbol{X}}(x)\, g(x)\, \mathrm{d}x & \text{if } \boldsymbol{X} \text{ is continuous.} \end{cases}$$

Note that $g(\boldsymbol{X})$ is actually a new random variable, say $Y := g(\boldsymbol{X})$, with some different distribution $p_Y \neq p_{\boldsymbol{X}}$. It holds that

$$\mathbb{E}_{\boldsymbol{X}}[g(\boldsymbol{X})] = \mathbb{E}_Y[Y].$$

This fact is called the **law of the unconscious statistician**.

A subscript of the expectation indicates which distribution is used in the weighted integral/sum.

# Moments, Mean, Variance

## Moments

The $k^{\text{th}}$ **moment** of RV $X$ is defined as $\mathbb{E}[X^k]$.

## Mean

The **mean** of RV $X$ is its first moment, i.e. the expected value $\mathbb{E}[X]$. For multivariate RVs, the mean $\mathbb{E}[\boldsymbol{X}]$ is a vector.

## Variance, Standard Deviation

The **variance** is defined as the **second central moment**, i.e. the second moment after removing the mean:

$$\text{var}[X] := \mathbb{E}[(X - \mathbb{E}[X])^2] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$$

The **standard deviation** of $X$ is defined as $\text{std}[X] := \sqrt{\text{var}[X]}$. Variance and standard deviation measure the spread of $X$.

The **covariance** of two RVs $X$ and $Y$ is the expectation of the product of their centered versions:

$$\mathrm{cov}[X, Y] = \mathbb{E}\big[(X - \mathbb{E}[X])\,(Y - \mathbb{E}[Y])\big]$$

**Positive:** increasing one RV tends to **increase** the other.

**Negative:** increasing one RV tends to **decrease** the other.

For a multivariate RV $\boldsymbol{X} = (X_1, \ldots, X_D)$ the **covariance matrix** is

$$\mathrm{cov}[\boldsymbol{X}] = \mathbb{E}\big[(\boldsymbol{X} - \mathbb{E}[\boldsymbol{X}])(\boldsymbol{X} - \mathbb{E}[\boldsymbol{X}])^T\big]$$

$$= \begin{pmatrix} \mathrm{var}[X_1] & \mathrm{cov}[X_1, X_2] & \ldots & \mathrm{cov}[X_1, X_D] \\ \mathrm{cov}[X_2, X_1] & \mathrm{var}[X_2] & \ldots & \mathrm{cov}[X_2, X_D] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{cov}[X_D, X_1] & \mathrm{cov}[X_D, X_2] & \ldots & \mathrm{var}[X_D] \end{pmatrix}$$

The **entropy** of a (multivariate) RV is defined as

$$H[\boldsymbol{X}] = \mathbb{E}[-\log p(\boldsymbol{X})] = -\int_{\boldsymbol{x}} p(\boldsymbol{x}) \log p(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \qquad \left(\sum \text{ for discrete RVs}\right)$$

- The entropy measures the **information** content of an RV, or how much it "surprises" us
- The entropy might be defined via **base-2 log** (unit is `bits`) or **natural logs** (unit is `nats`). For example, a Bernoulli RV with $\theta = 0.5$ (fair coin) has entropy of 1 `bits` $= 0.693$ `nats`.
- We will use usually natural logs (`nats`) in this course

Given two distributions $p$ and $q$ over the same state space of some (multivariate) RV, the **Kullback-Leibler divergence** (KL) is defined as

$$\mathrm{KL}(p||q) = \mathbb{E}_p\left[\log \frac{p(\boldsymbol{X})}{q(\boldsymbol{X})}\right] = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} \mathrm{d}\boldsymbol{x} \quad \left(\sum \text{ for discrete}\right)$$

- measures similarity between $p$ and $q$
- $\mathrm{KL}(p||q) \geq 0$
- $\mathrm{KL}(p||q) = 0$ if and only if the distributions are the same
- **not a distance though**—violates the triangle inequality
- also, not symmetric

$$KL(p||q) \neq KL(q||p)$$

# Independence

We say that two (univariate or multivariate) RVs $\boldsymbol{X}$ and $\boldsymbol{Y}$ are **independent**, written $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}$, if the following **equivalent** conditions hold for all $\boldsymbol{x}$ and $\boldsymbol{y}$:

- $p(\boldsymbol{x}, \boldsymbol{y}) = p(\boldsymbol{x})\, p(\boldsymbol{y})$ (joint factorizes)
- $p(\boldsymbol{x} \mid \boldsymbol{y}) = p(\boldsymbol{x})$ (marginal equals conditional)
- $p(\boldsymbol{y} \mid \boldsymbol{x}) = p(\boldsymbol{y})$ (marginal equals conditional)

**Intuition:** $\boldsymbol{X}$ has no information about $\boldsymbol{Y}$ and vice versa.
Predicting $\boldsymbol{Y}$ from $\boldsymbol{X}$ is the same as just guessing $\boldsymbol{Y}$ without $\boldsymbol{X}$.

Let $X, Y$ be binary variables with joint and marginals:

| $p(x, y)$ | $y = 0$ | $y = 1$ | $p(x)$ |
|---|---|---|---|
| $x = 0$ | 0.12 | 0.28 | 0.4 |
| $x = 1$ | 0.18 | 0.42 | 0.6 |
| $p(y)$ | 0.3 | 0.7 | |

Let $X, Y$ be binary variables with joint and marginals:

| $p(x,y)$ | $y=0$ | $y=1$ | $p(x)$ |
|---|---|---|---|
| $x=0$ | 0.12 | 0.28 | 0.4 |
| $x=1$ | 0.18 | 0.42 | 0.6 |
| $p(y)$ | 0.3 | 0.7 | |

$$p(x=0,y=0) = 0.12 = 0.4 \times 0.3 = p(x=0) \times p(y=0)$$
$$p(x=0,y=1) = 0.28 = 0.4 \times 0.7 = p(x=0) \times p(y=1)$$
$$p(x=1,y=0) = 0.18 = 0.6 \times 0.3 = p(x=1) \times p(y=0)$$
$$p(x=1,y=1) = 0.42 = 0.6 \times 0.7 = p(x=1) \times p(y=1)$$

**Thus, $X \perp\!\!\!\perp Y$ ($X$ and $Y$ are independent)**

Let $X, Y$ be binary variables with joint and marginals

| $p(x,y)$ | $y=0$ | $y=1$ | $p(x)$ |
|---|---|---|---|
| $x=0$ | 0.1 | 0.1 | 0.2 |
| $x=1$ | 0.08 | 0.72 | 0.8 |
| $p(y)$ | 0.18 | 0.82 | |

Let $X, Y$ be binary variables with joint and marginals

| $p(x,y)$ | $y=0$ | $y=1$ | $p(x)$ |
|---|---|---|---|
| $x=0$ | 0.1 | 0.1 | 0.2 |
| $x=1$ | 0.08 | 0.72 | 0.8 |
| $p(y)$ | 0.18 | 0.82 | |

$$p(x=0,y=0) = 0.1 \neq 0.036 = 0.2 \times 0.18 = p(x=0) \times p(y=0)$$

**Thus, $X \not\perp\!\!\!\perp Y$ ($X$ and $Y$ are not independent)**

**Note: finding a single $x, y$ violating factorization is sufficient.**

But also for the other states the joint does not factorize.

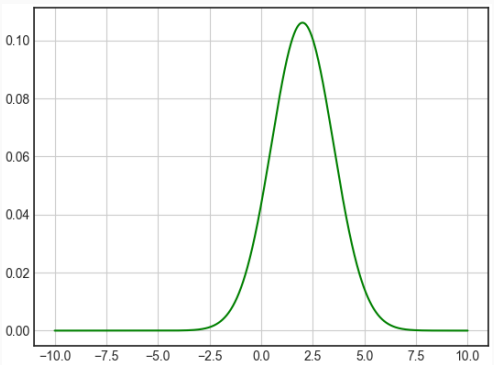
Recall the multivariate Gaussian:

$$p(\boldsymbol{x} \mid \boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^D \, |\Sigma|}} \, e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x}-\boldsymbol{\mu})}$$

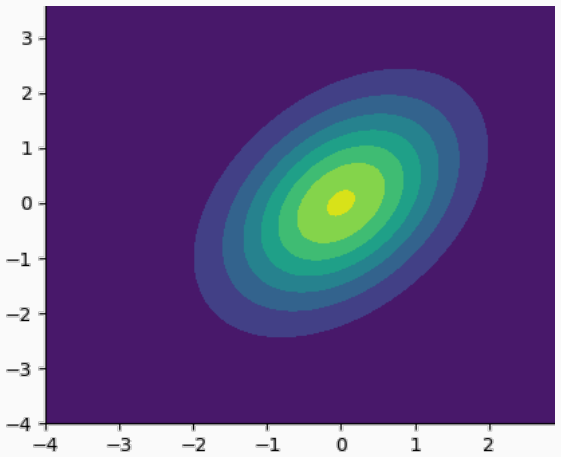
with mean vector $\boldsymbol{\mu} \in \mathbb{R}^D$ and positive definite covariance matrix $\Sigma \in \mathbb{R}^{D \times D}$.

For any $D$, the **contour sets** of Gaussians are **ellipsoids**:

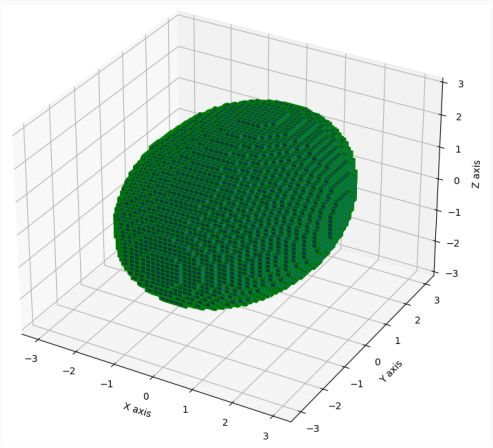
1D Gaussian




2D Gaussian




3D Gaussian



# Diagonal Covariance = Independence

When the covariance matrix $\Sigma$ is **diagonal** (off-diagonal entries are 0), a multivariate Gaussians factorize into 1d-Gaussians:

$$p(\boldsymbol{x} \mid \boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^D \, |\Sigma|}} \, e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x}-\boldsymbol{\mu})}$$

$$= \frac{1}{\sqrt{(2\pi)^D \, |\Sigma|}} \, e^{-\frac{1}{2} \overbrace{\left( \sum_{d=1}^{D} \frac{(x_d - \mu_d)^2}{\Sigma_{dd}} \right)}^{\text{diagonal } \Sigma}}$$

$$= \prod_{d=1}^{D} \frac{1}{\sqrt{(2\pi) \, \Sigma_{dd}}} \, e^{-\frac{1}{2} \left( \frac{(x_d - \mu_d)^2}{\Sigma_{dd}} \right)}$$

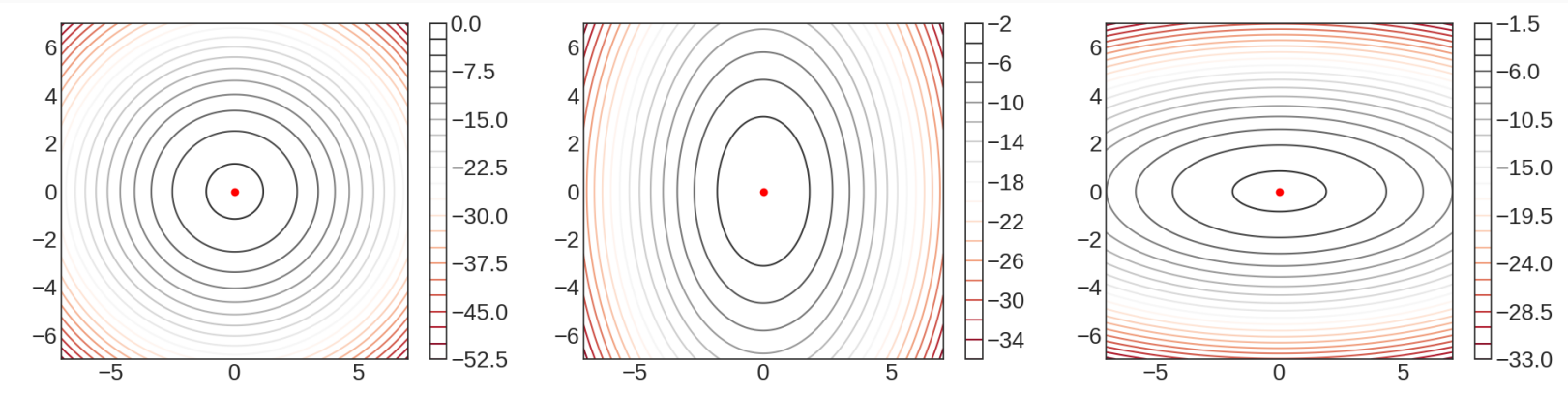
$$= \prod_{d} p(x_d \mid \mu_d, \Sigma_{dd})$$

**Gaussian with diagonal covariance $\Leftrightarrow$ independent 1d-Gaussians**

# Diagonal Covariance

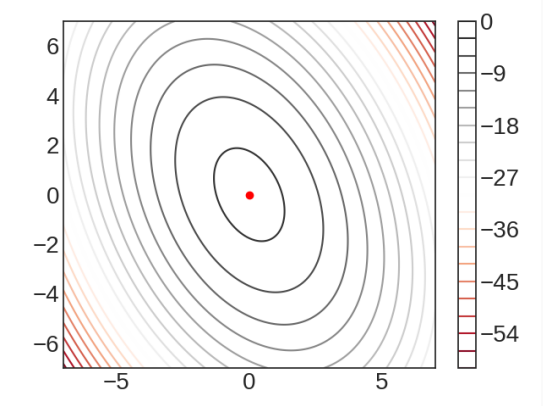
Example

When $\Sigma$ is diagonal (independent Gaussians) the contour lines are **axis-aligned** ellipses. Moreover, when the diagonal is constant, the contour lines are **hyper-spheres**:

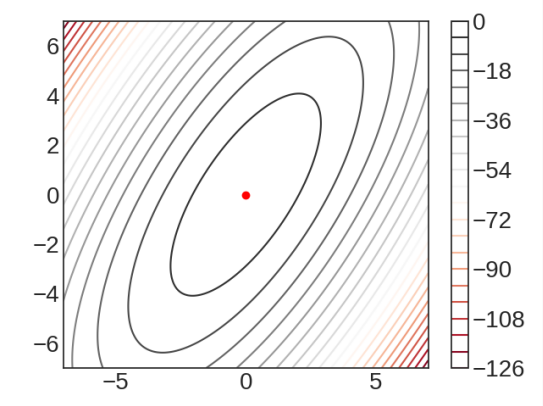



$$\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix} \qquad \Sigma = \begin{pmatrix} 5 & 0 \\ 0 & 1 \end{pmatrix}$$

If $\Sigma$ is not a diagonal matrix the ellipsoids are "slanted" and the dimensions are not independent:



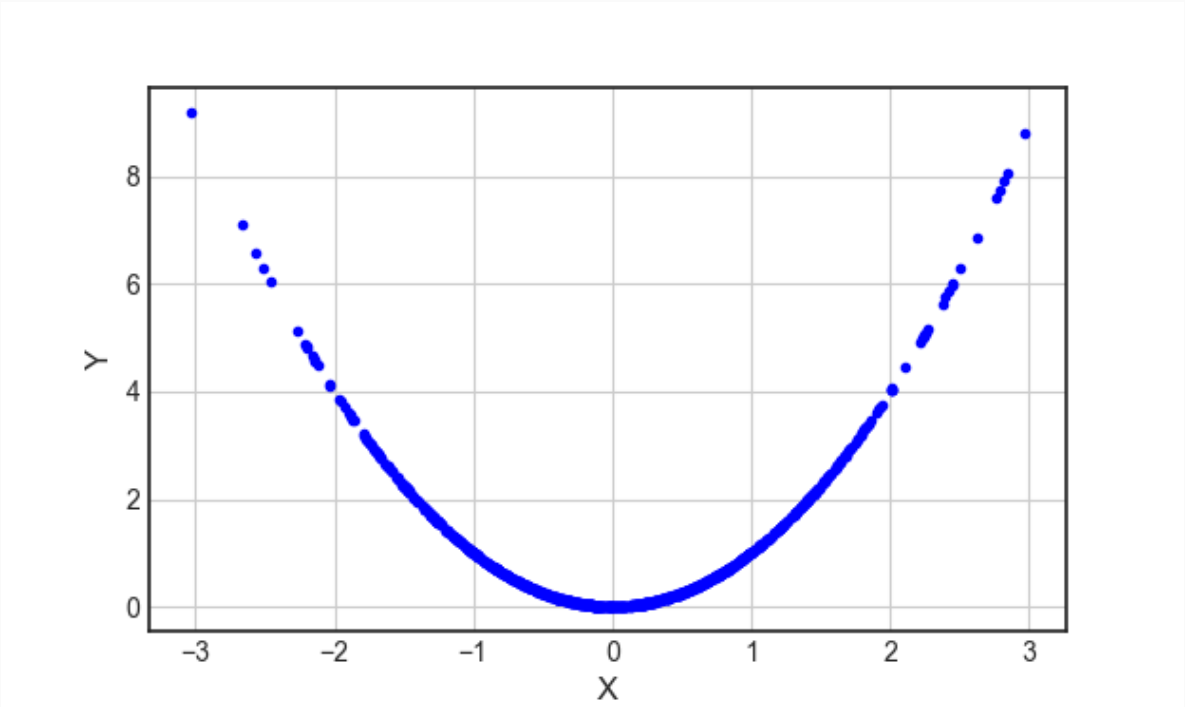

$$\Sigma = \begin{pmatrix} 1 & -0.5 \\ -0.5 & 2 \end{pmatrix}$$




$$\Sigma = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$$

# Covariance and Independence

- two Gaussians have 0 covariance if and only if they are independent
- **for non-Gaussians this is not the case**
- two RVs can have 0 covariance, yet be highly dependent
- for example, it can be shown that $X \sim \mathcal{N}(0, 1)$ and $Y = X^2$ have 0 covariance, yet they are of course dependent:



Given two (multivariate) RVs $\boldsymbol{X}, \boldsymbol{Y}$ with joint distribution $p_{\boldsymbol{X},\boldsymbol{Y}}$, the **mutual information** is defined as the KL-divergence between the joint and the product of marginals:

$$\begin{aligned}\mathrm{MI}(\boldsymbol{X}, \boldsymbol{Y}) &= \mathrm{KL}(p_{\boldsymbol{X},\boldsymbol{Y}} || p_{\boldsymbol{X}}\, p_{\boldsymbol{Y}}) \\ &= \mathbb{E}_{\boldsymbol{X},\boldsymbol{Y}}\left[\log \frac{p(\boldsymbol{X}, \boldsymbol{Y})}{p(\boldsymbol{X})p(\boldsymbol{Y})}\right] \\ &= \int\int p(\boldsymbol{x}, \boldsymbol{y}) \log \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}\end{aligned}$$

Thus, if $\boldsymbol{X}$ and $\boldsymbol{Y}$ are independent then $\mathrm{MI}(\boldsymbol{X}, \boldsymbol{Y}) = 0$.

Intuitively, The mutual information measures the "amount of information" shared by the two RVs.

We say that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are **conditionally independent given** $\boldsymbol{Z}$, written as $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} \mid \boldsymbol{Z}$, if the following equivalent conditions hold for all $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$:

- $p(\boldsymbol{x}, \boldsymbol{y} \mid \boldsymbol{z}) = p(\boldsymbol{x} \mid \boldsymbol{z})\, p(\boldsymbol{y} \mid \boldsymbol{z})$
- $p(\boldsymbol{x} \mid \boldsymbol{y}, \boldsymbol{z}) = p(\boldsymbol{x} \mid \boldsymbol{z})$
- $p(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{z}) = p(\boldsymbol{y} \mid \boldsymbol{z})$

Same conditions as for independence, but applied to conditional distributions.

**Intuition:** when observing $\boldsymbol{Z}$, $\boldsymbol{X}$ and $\boldsymbol{Y}$ are rendered independent.

# Conditional Independence cont'd

**Note 1.** (Unconditional) independence is just a special case with $\boldsymbol{Z} = \emptyset$.

**Note 2.** Conditional and unconditional independence can behave very differently. We can have all possible situations:

- $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}$ and $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} \mid \boldsymbol{Z}$
- $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}$ and $\boldsymbol{X} \not\perp\!\!\!\perp \boldsymbol{Y} \mid \boldsymbol{Z}$
- $\boldsymbol{X} \not\perp\!\!\!\perp \boldsymbol{Y}$ and $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} \mid \boldsymbol{Z}$
- $\boldsymbol{X} \not\perp\!\!\!\perp \boldsymbol{Y}$ and $\boldsymbol{X} \not\perp\!\!\!\perp \boldsymbol{Y} \mid \boldsymbol{Z}$

**Note 3.** Conditional and unconditional independence are key concepts in probabilistic modeling.

# Sampling Basics

# Sampling

**Sampling** is a **very** important tool in probabilistic ML. If we want, we can think of them as the **fourth inference routine**, next to marginalization, conditioning and expectations.

- sampling is used in **generative modeling**, that is, producing "new" data samples that are similar to given data
- basically one fits a **probabilistic model** (description of a joint distribution) to the data and then samples the model
- sampling is extremely important in **approximate inference** and **learning** via **Monte Carlo** techniques

- let some distribution function $p_{\boldsymbol{X}}$ be given
- assume we have access to iid copies $\boldsymbol{U}_1, \boldsymbol{U}_2, \boldsymbol{U}_3 \ldots$, of some "noise source" with known distribution $p_{\boldsymbol{U}}$
- usually, $p_{\boldsymbol{U}}$ is something simple, like independent uniformly distributed RVs:

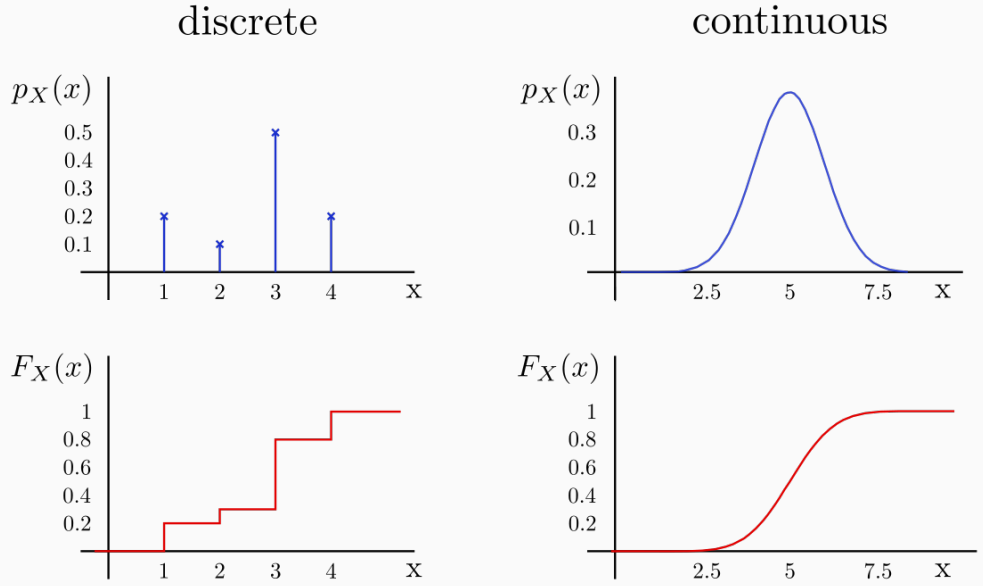
$$p_{\boldsymbol{U}}(\boldsymbol{u}) = \prod_i \text{Uniform}(u_i \mid 0, 1)$$

- **sampling** or **simulating** $p_{\boldsymbol{X}}$ means constructing a function $f$ that transforms noise RVs into RVs following $p_{\boldsymbol{X}}$:

$$\boldsymbol{U} \sim p_{\boldsymbol{U}}$$
$$f(\boldsymbol{U}) \sim p_{\boldsymbol{X}}$$

Recall from last lecture the cumulative distribution function (CDF):

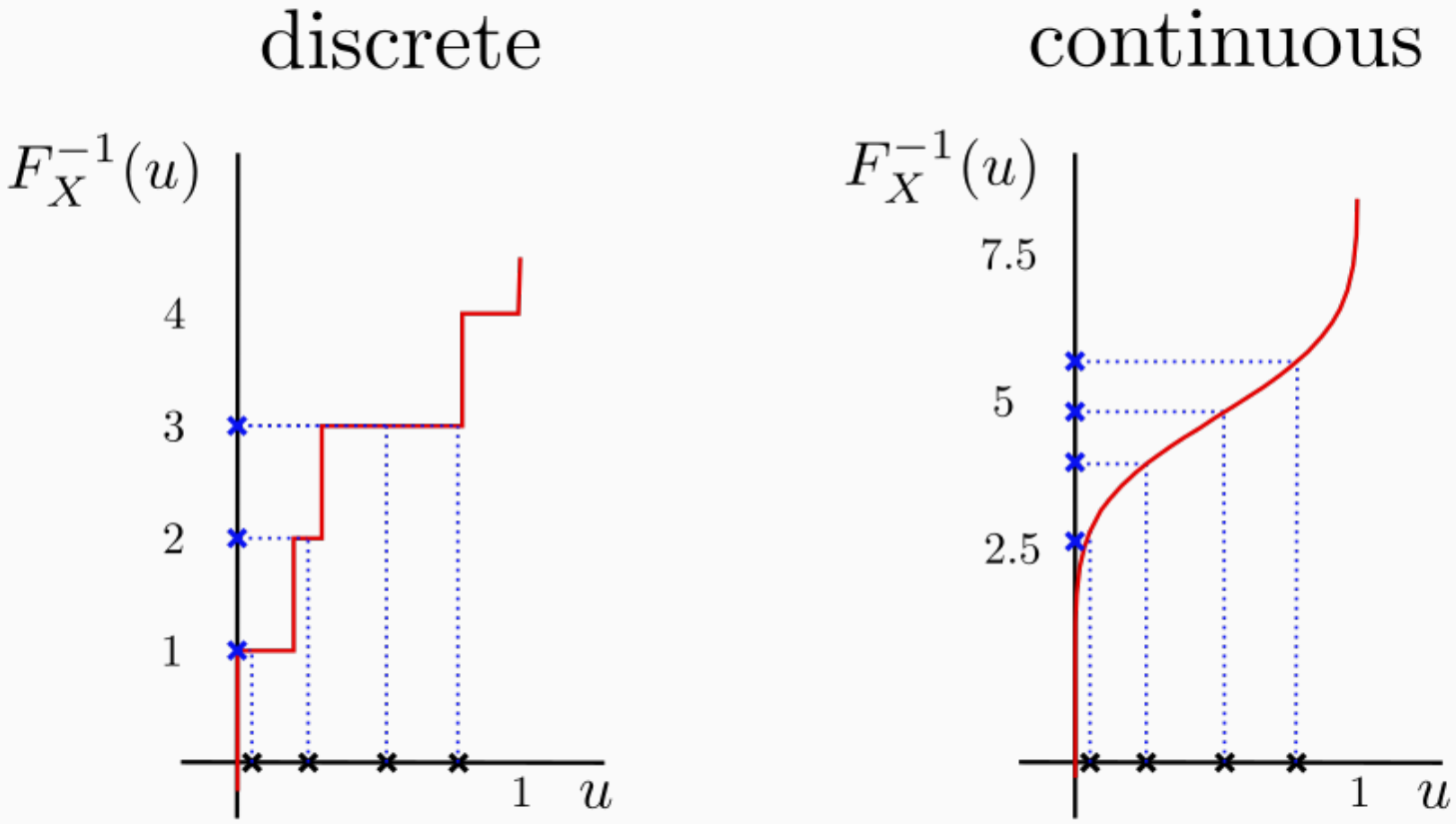
$$F_X(x) := \mathbb{P}_X([-\infty, x])$$




Assume we know the **inverse CDF** $F_X^{-1}$ corresponding to $p_X$. For $U \sim \text{Uniform}(0, 1)$ then $F_X^{-1}(U)$ are samples from $p_X$:

$$U \sim \text{Uniform}(0, 1)$$

$$F_X^{-1}(U) \sim p_X$$




If the inverse is not unique, we use the smallest $x$ for which $F_X(x) = u$.

By using **pseudo-random** uniformly distributed samples (e.g. `np.random.rand` in `NumPy`) we can use inverse CDF sampling to simulate:

- Bernoulli with success parameter $\theta = 0.8$:

$$1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, \ldots$$

- Categorical($\boldsymbol{\theta}$) with $\boldsymbol{\theta} = (0.1, 0.3, 0.4, 0.2)$:

$$1, 3, 2, 2, 3, 2, 3, 1, 1, 3, 2, 3, 1, 2, 2, 0, 3, 2, 1, 3, \ldots$$


dilbert.com

# Sampling Discrete Distributions cont'd

**Example**

What about multivariate PMFs? A simple way is to encode the multivariate RV with a single Categorical RV:

| $x_1$ | $x_2$ | $x_3$ | $p(x_1, x_2, x_3)$ | | $y$ | $p(y)$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0.1 | $\longleftrightarrow$ | 0 | 0.1 |
| 0 | 0 | 1 | 0.05 | $\longleftrightarrow$ | 1 | 0.05 |
| 0 | 1 | 0 | 0.2 | $\longleftrightarrow$ | 2 | 0.2 |
| 0 | 1 | 1 | 0.05 | $\longleftrightarrow$ | 3 | 0.05 |
| 1 | 0 | 0 | 0.01 | $\longleftrightarrow$ | 4 | 0.01 |
| 1 | 0 | 1 | 0.09 | $\longleftrightarrow$ | 5 | 0.09 |
| 1 | 1 | 0 | 0.2 | $\longleftrightarrow$ | 6 | 0.2 |
| 1 | 1 | 1 | 0.3 | $\longleftrightarrow$ | 7 | 0.3 |

Then, by sampling the single RV with inverse CDF sampling and transforming back we get joint samples:

$$(1,1,1), (1,1,0), (0,1,0), (0,0,0), (1,0,1), \ldots$$

- sampling 1d-Gaussian is readily available (`np.random.randn`)
- **standard Gaussians** can be easily transformed:

$$U \sim \mathcal{N}(0, 1)$$

$$\mu + U\sigma \sim \mathcal{N}(\mu, \sigma)$$

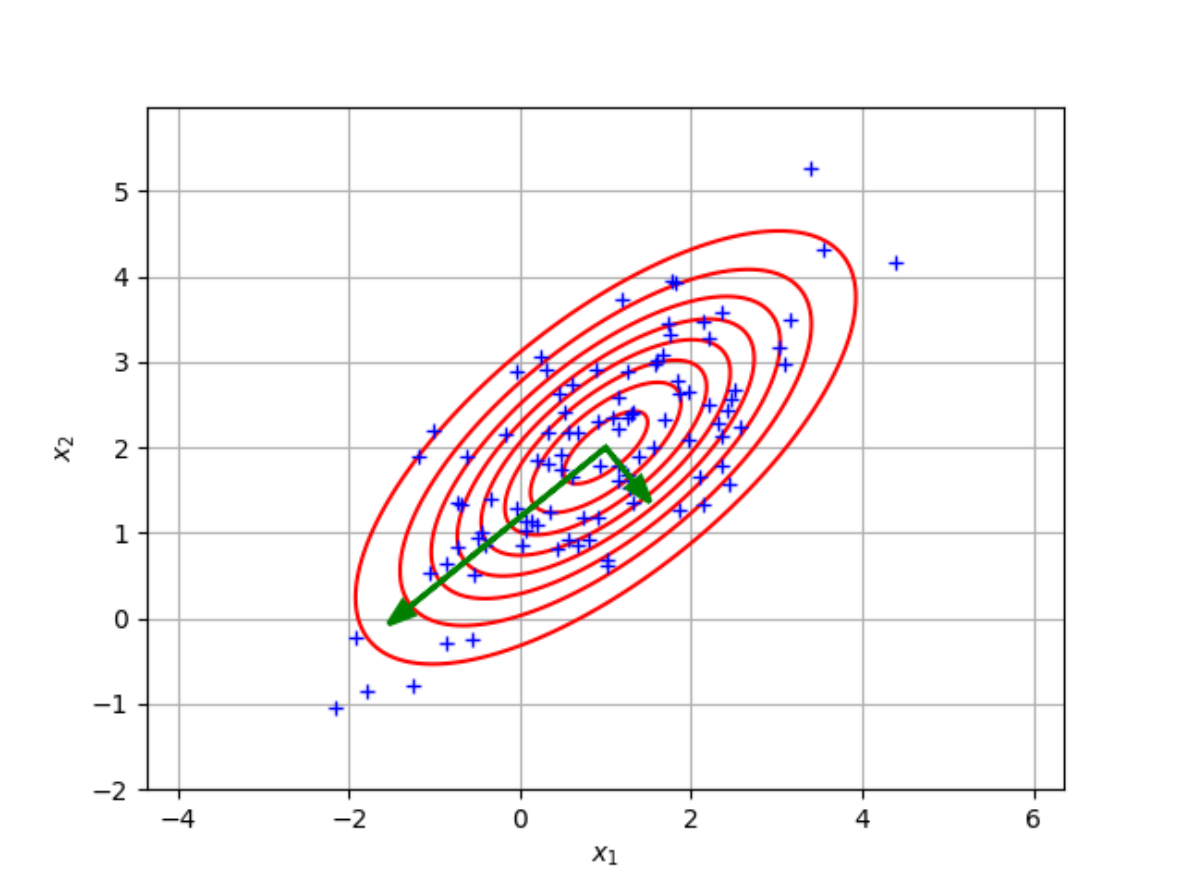
- sampling multivariate Gaussian with **diagonal** covariance is just sampling independent 1d-Gaussians
- for **arbitrary** covariance, reduce to the diagonal case:
  - compute eigen decomposition of covariance matrix

$$\overbrace{(\lambda_1, \ldots, \lambda_D)}^{\text{variances}}, (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_D) = \text{eigen}(\Sigma)$$

  - the joint Gaussian sample is generated as

$$\xi_1, \ldots, \xi_D \overset{iid}{\sim} \mathcal{N}(0, 1) \qquad \boldsymbol{x} = \boldsymbol{\mu} + \sum_{i=1}^{D} \xi_i \sqrt{\lambda_i} \boldsymbol{v}_i$$

$$\mu = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad \Sigma = \begin{pmatrix} 2 & 1.2 \\ 1.2 & 1.5 \end{pmatrix}$$



# Understanding Marginals and Conditionals via Sampling

Assume a joint distribution $p_{\boldsymbol{X}}$ from which we can draw iid samples:

$$x_1, x_2, \ldots, x_D \overset{iid}{\sim} p_{\boldsymbol{X}}$$

**How can we derive samples from any marginal?**

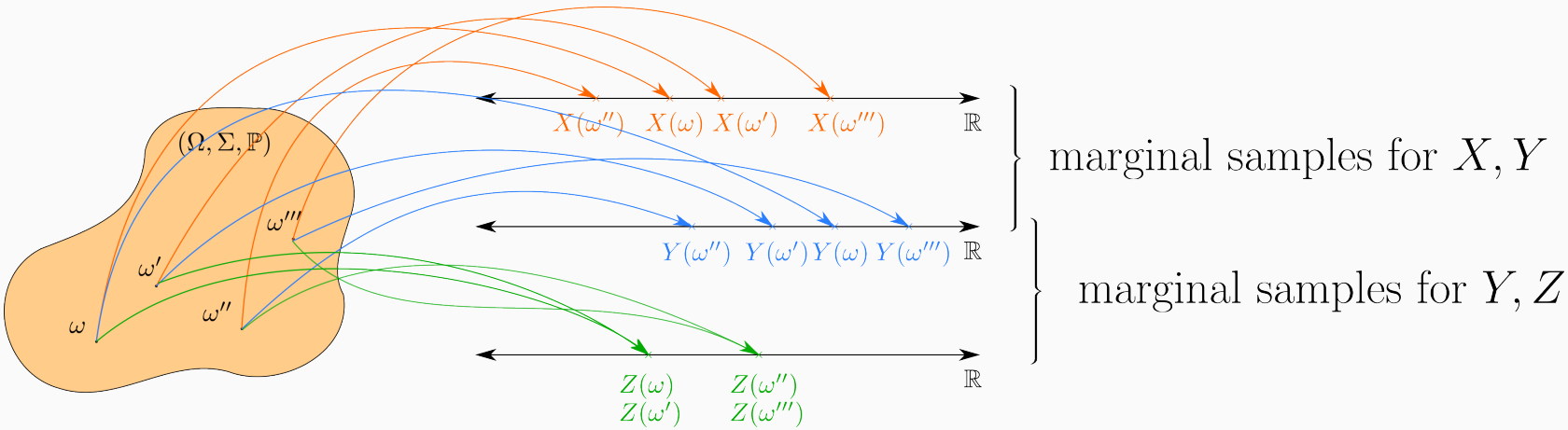
**How can we derive samples from any conditional?**

# Sampling Marginals by Discarding Dimensions

**Recall that marginalization means "forgetting".** It is hence pleasing that samples from marginals can be obtained by

- sampling the joint $x_1, x_2, \ldots, x_D \overset{iid}{\sim} p_{\boldsymbol{X}}$
- discarding the marginalized dimensions

**This is universally true for <u>any</u> joint!** It is just a consequence of RVs being functions defined on the same probability space.



# Sampling Conditionals by Filtering

Recall that conditioning means "observing". It is hence pleasing that samples for conditionals can be obtained by

- repeatedly sampling the joint $x_1, x_2, \ldots, x_D \overset{iid}{\sim} p_{\boldsymbol{X}}$ ...
- ... until the conditioned dimensions $\boldsymbol{Z}$ agree with the right side of the conditioning bar $p(\cdot \mid \boldsymbol{Z} = \boldsymbol{z})$

However, in general this is **extremely inefficient**, so don't do this in practice. For continuous distributions, one also needs to approximate this by filtering for samples where $\boldsymbol{Z} \approx \boldsymbol{z}$, allowing small deviations (otherwise you will wait until eternity before accepting a sample).

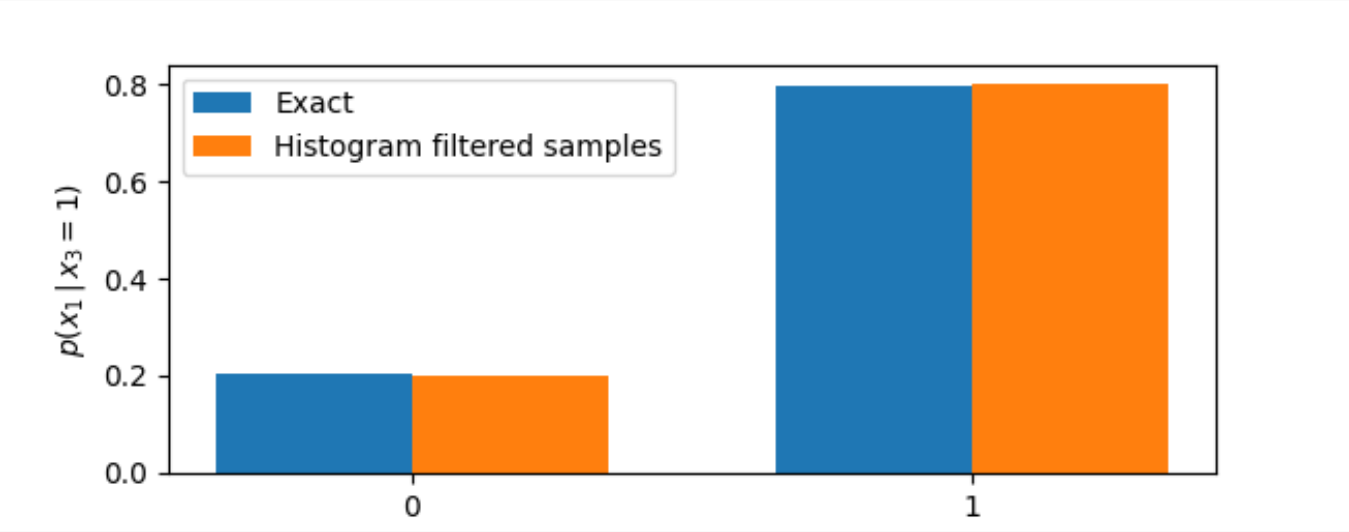
# Sampling Conditionals by Filtering

Example

| $x_1$ | $x_2$ | $x_3$ | $p(x_1, x_2, x_3)$ |
|---|---|---|---|
| 0 | 0 | 0 | 0.1 |
| 0 | 0 | 1 | 0.05 |
| 0 | 1 | 0 | 0.2 |
| 0 | 1 | 1 | 0.05 |
| 1 | 0 | 0 | 0.01 |
| 1 | 0 | 1 | 0.09 |
| 1 | 1 | 0 | 0.2 |
| 1 | 1 | 1 | 0.3 |

| $x_1$ | $x_3$ | $p(x_1 \mid x_3)$ |
|---|---|---|
| 0 | 0 | 0.588 |
| 1 | | 0.412 |
| 0 | 1 | 0.2 |
| 1 | | 0.8 |

Sampling the joint $10k$ times, keeping only samples where $X_3 = 1$:



- **key inference routines:** **marginalization** and **conditioning**
- **consistent reasoning process**
- **expectations**, **moments**, **entropy**, **Kullback-Leibler divergence**
- **sampling basics**: marginalization is discarding dimensions, conditioning is filtering samples