T2.3 Probabilidad, regla de Bayes y distribuciones

Índice

1 Introducción (opcional)

1.1 ¿Qué es probabilidad?

Frecuentista: frecuencia (asintótica) de eventos repetitivos

Bayesiana: modelo de **incertidumbre** sobre eventos

1.2 Tipos de incertidumbre

Epistémica: incertidumbre sobre el modelo

Aleatórica: incertidumbre de los datos

1.3 Probabilidad como extensión de la lógica

1.3.1 Probabilidad de un evento

Evento: variable binaria A

Probabilidad de que el evento A sea cierto: $\operatorname{Pr}(A)$

Restricción de probabilidad: $0 \leq \operatorname{Pr}(A) \leq 1$

Evento imposible: $\operatorname{Pr}(A)=0$ significa que es **imposible** que A ocurra (falso)

Evento seguro: $\operatorname{Pr}(A)=1$ significa que es **seguro** que A ocurra (cierto)

Probabilidad de que el evento A no ocurra: $\operatorname{Pr}(\bar{A})=1-\operatorname{Pr}(A)$

1.3.2 Probabilidad de una conjunción de dos eventos

Probabilidad conjunta de que A y B ocurran: $\operatorname{Pr}(A \wedge B)=\operatorname{Pr}(A, B)$

Probabilidad conjunta de eventos independientes: si A y B son independientes, entonces $\Pr(A,B) = \Pr(A)\Pr(B)$

Ejemplo: si X e Y se escogen uniformente al azar de $\mathcal{X} = \{1,2,3,4\}$, A es el evento $X \in \{1,2\}$ y B el evento $Y \in \{3\}$; entonces A y B son independientes (pues X e Y se escogen independientemente) y $\Pr(A,B) = \Pr(A)\Pr(B) = \frac{1}{2} \cdot \frac{1}{4} = \frac{1}{8}$

1.3.3 Probabilidad de una unión de dos eventos

Probabilidad de que el evento A o B ocurran (uno solo o los dos): $\Pr(A \vee B) = \Pr(A) + \Pr(B) - \Pr(A \wedge B)$

Caso A y B mútuamente excluyentes (no pueden darse a la vez): $\Pr(A \vee B) = \Pr(A) + \Pr(B)$

Ejemplo: Si X se escoge uniformente al azar de $\mathcal{X} = \{1,2,3,4\}$, A es el evento $X \in \{1,2\}$ y B el evento $X \in \{3\}$; entonces $\Pr(A \vee B) = \frac{2}{4} + \frac{1}{4}$

1.3.4 Probabilidad condicional de un evento dado otro

Probabilidad condicional de que un evento B ocurra sabiendo que otro evento dado, A, ha ocurrido:

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

Nota: no está definida si $\Pr(A) = 0$ pues no tiene sentido condicionar sobre la base de un evento imposible

1.3.5 Independencia de eventos

Eventos independientes: A y B son **independientes** sii $\Pr(A,B) = \Pr(A)\Pr(B)$

Notación: $A \perp B$ o $A \perp\!\!\!\perp B$

1.3.6 Independencia condicional de eventos

Eventos condicionalmente independientes: A y B son **condicionalmente independientes** dado C sii $\Pr(A, B \mid C) = \Pr(A \mid C)\Pr(B \mid C)$

Notación: $A \perp B \mid C$ o $A \perp\!\!\!\perp B \mid C$

1.4 Variables aleatorias

Variable aleatoria: valor desconocido de interés X

Espacio muestral: conjunto de valores posibles $\mathcal{X}$

Evento: subconjunto de valores en $\mathcal{X}$

Ejemplo: variable aleatoria $X =$"valor obtenido al lanzar un dado" definida sobre el espacio muestral $\mathcal{X} = \{1, \dots, 6\}$

- Evento $X = 1 \to$ "sale 1"
- Evento $X \in \{1, 3, 5\} \to$ "sale valor impar"
- Evento $1 \leq X \leq 3 \to$ "sale entre 1 y 3"

1.4.1 Variables aleatorias discretas

Variable aleatoria discreta: $\mathcal{X}$ discreto, finito o infinito contable

Probabilidad del evento "X toma el valor x": $\Pr(X = x)$

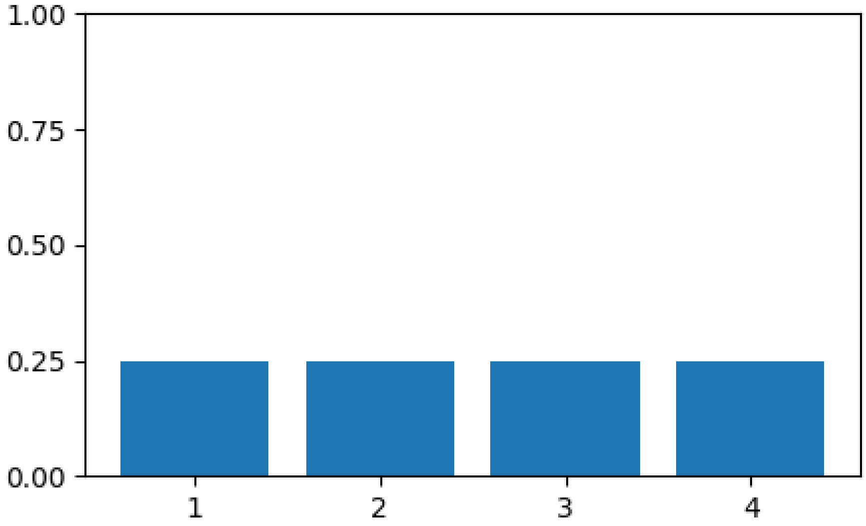
Función de masa de probabilidad (pmf): $p(x) = \Pr(X = x)$

Condiciones de una pmf: $0 \leq p(x) \leq 1$ y $\displaystyle\sum_{x \in \mathcal{X}} p(x) = 1$

Ejemplo: distribución uniforme en $\mathcal{X} = \{1, 2, 3, 4\},\ p(x) = 1/4$

In [1]:
```
import numpy as np; import matplotlib.pyplot as plt
X = np.arange(1, 5); pmf = np.repeat(1.0/len(X), len(X))
```

```
fig = plt.subplots(figsize=(5,3)); plt.xticks(X);
plt.yticks(np.linspace(0, 1, 5)); plt.ylim((0, 1)); plt.bar(X, pmf, align='center');
```



1.4.2 Variables aleatorias continuas

Variable aleatoria continua: $\mathcal{X}$ es $\mathbb{R}$

Necesidad de particionar $\mathcal{X}$ en un número discreto de intervalos:

- No podemos asociar X a un número discreto de valores distintos
- Sí podemos asociar X a un número discreto de **intervalos** que particionen $\mathcal{X}$
- Asociando eventos con que X pertenezca a cada uno de los intervalos, razonamos con ellos como si fueran valores del caso discreto
- La probabilidad de que X tome un valor específico se aproxima tomando un intervalo de talla infinitesimal

1.4.2.1 Función de distribución acumulada (cdf)

Función de distribución acumulada (cdf) de una v.a. X: $P(x) = \Pr(X \leq x)$

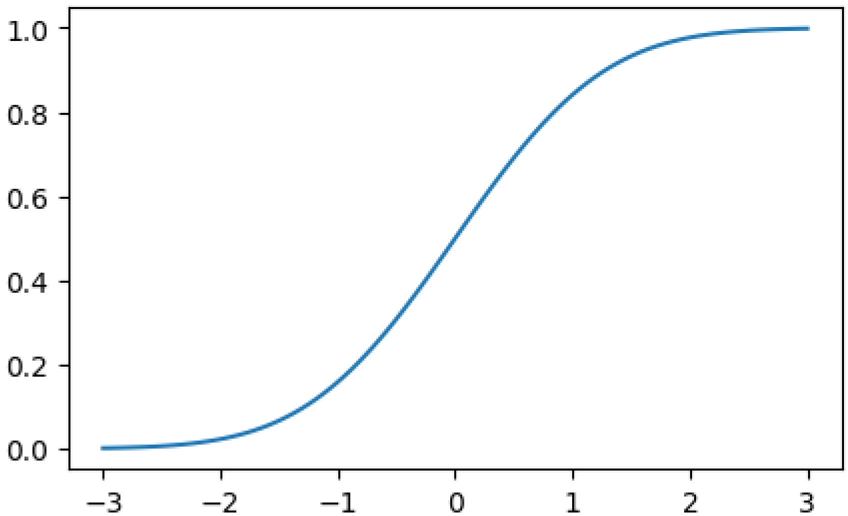
Probabilidad de que X se encuentre en un semi-abierto $C = (a < X \leq b)$, con $a < b$:

- Sean $A = (X \leq a)$ y $B = (X \leq b)$
- Dado que $B = A \vee C$ y que A y C son mútuamente excluyentes, $\operatorname{Pr}(B) = \operatorname{Pr}(A) + \operatorname{Pr}(C)$
- Por tanto, $\operatorname{Pr}(C) = \operatorname{Pr}(B) - \operatorname{Pr}(A) = P(b) - P(a)$

Monotonicidad: las cdfs son funciones monótonas no decrecientes

Ejemplo: cdf de la normal estándard

In [2]:
```
import numpy as np; import matplotlib.pyplot as plt; from scipy.stats import norm
X = np.linspace(-3, 3, 100); fig = plt.subplots(figsize=(5,3)); plt.plot(X, norm.cdf(X));
```



1.4.2.2 Función de densidad de probabilidad (pdf)

Función de densidad de probabilidad (pdf): derivada de la cdf (donde existe), $p(x) = \frac{d}{dx}P(x)$

Probabilidad de que X se encuentre en un semi-abierto $C = (a < X \leq b)$, con $a < b$:

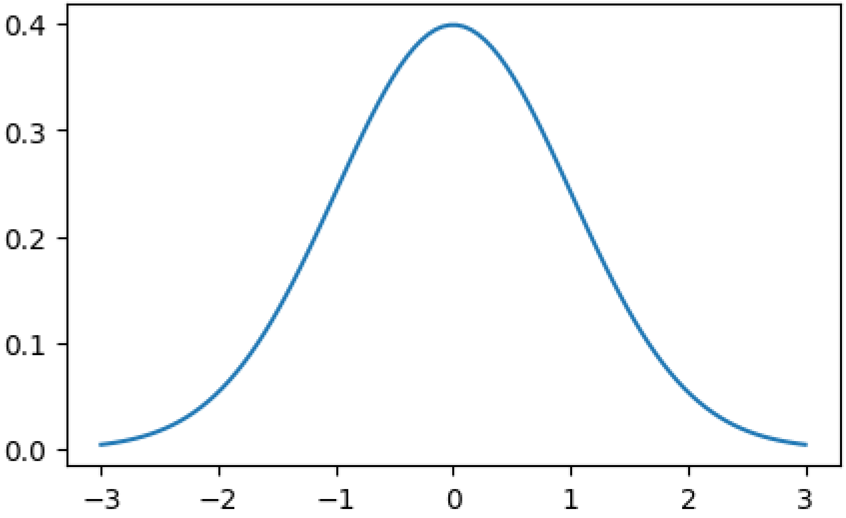
$$\operatorname{Pr}(a < X \leq b) = \int_a^b p(x)\,dx = P(b) - P(a)$$

Aproximación a la probabilidad de que X tome un valor específico x: la probabilidad de que X "caiga" en un pequeño intervalo alrededor de x es aproximadamente igual a la densidad en x por la amplitud del intervalo

$$\Pr(x < X \leq x + dx) \approx p(x)\,dx$$

Ejemplo: pdf de la normal estándar

In [3]:
```
import numpy as np; import matplotlib.pyplot as plt; from scipy.stats import norm
X = np.linspace(-3, 3, 100); fig = plt.subplots(figsize=(5,3)); plt.plot(X, norm.pdf(X));
```



1.4.2.3 Cuantiles

Cdf inversa o función cuantil de una cdf P: para todo $q \in (0, 1)$

$$\begin{aligned} P^{-1}(q) &= \{x : P(x) = q\} && \text{si } P \text{ es monótona estricta}\\ P^{-1}(q) &= \inf\{x : P(x) \geq q\} && \text{si } P \text{ no es monótona estricta (con saltos o llanos)}\end{aligned}$$

Cuantil q de P: $x_q = P^{-1}(q)$ tal que $\Pr(X \leq x_q) = q$

Mediana de P: $P^{-1}(0.5)$

Cuartiles inferior y superior de P: $P^{-1}(0.25)$ y $P^{-1}(0.75)$

Ejemplo: si Φ es la cdf de la normal estándar y Φ^{-1} su inversa, el intervalo centrado en el origen con un 95% de probabilidad es

$$(\Phi^{-1}(0.025), \Phi^{-1}(0.975)) = (-1.96, 1.96)$$

1.4.3 Conjuntos de variables aleatorias relacionadas

Distribución conjunta de dos variables aleatorias X e Y: $p(x, y) = p(X = x, Y = y)$

Caso finito: si X e Y son finitas, su distribución conjunta puede representarse en una tabla 2d cuyas entradas suman uno en total

Ejemplo:

$p(X,Y)$	$Y=0$	$Y=1$
$X=0$	0.2	0.3
$X=1$	0.3	0.2

Distribución marginal de una variable aleatoria: también llamada **regla suma o de la probabilidad total**

$$p(X = x) = \sum_y p(X = x, Y = y)$$
$$p(Y = y) = \sum_x p(X = x, Y = y)$$

Ejemplo (cont.): marginales en los **márgenes** de la tabla

$p(X,Y)$	$Y=0$	$Y=1$	$p(X)$
$X=0$	0.2	0.3	0.5
$X=1$	0.3	0.2	0.5
$p(Y)$	0.5	0.5	

Distribución condicional de Y dada X: $p(Y=y \mid X=x)=\dfrac{p(X=x,Y=y)}{p(X=x)}$

Ejemplo (cont.):

$p(Y \mid X)$	$Y=0$	$Y=1$
$X=0$	0.4	0.6
$X=1$	0.6	0.4

Regla producto: reordenando factores, $p(X=x,Y=y)=p(X=x)p(Y=y \mid X=x)$

Regla de la cadena: extiende la regla producto a D variables

$$p(\boldsymbol{x}_{1:D})=p(x_1)\,p(x_2 \mid x_1)\,p(x_3 \mid x_1,x_2)\cdots p(x_D \mid \boldsymbol{x}_{1:D-1})$$

1.4.4 Independencia e independencia condicional

Independencia (incondicional o marginal) de dos variables: $X \perp Y \;\Leftrightarrow\; P(X,Y)=P(X)\,P(Y)$

Independencia (mútua) de múltiples variables: $X_1,\dots,X_n$ **independientes** si, para todo $\{X_1,\dots,X_m\}\subseteq\{X_1,\dots,X_n\}$,

$$P(X_1,\dots,X_m)=\prod_{i=1}^{m}P(X_i)$$

Independencia condicional de dos variables: X e Y son **condicionalmente independientes** dada Z sii su conjunta condicional puede expresarse como el producto de sus marginales condicionales

$$X \perp Y \mid Z \;\Leftrightarrow\; P(X,Y \mid Z)=P(X \mid Z)\,P(Y \mid Z)$$

1.4.5 Momentos de una distribución

1.4.5.1 Media de una distribución

Media o valor esperado:

$$\mu = \mathbb{E}[X] = \sum_{x \in \mathcal{X}} x\, p(x) \qquad \text{para variables discretas (ordenadas)}$$

$$\mu = \mathbb{E}[X] = \int_{\mathcal{X}} x\, p(x)\, dx \qquad \text{para variables continuas}$$

Linealidad de la esperanza: $\mathbb{E}[aX + b] = a\mathbb{E}[X] + b$

Esperanza de la suma de n variables aleatorias: $\mathbb{E}\left[\sum_{i=1}^{n} X_i\right] = \sum_{i=1}^{n} \mathbb{E}[X_i]$

Esperanza del producto de n variables aleatorias independientes: $\mathbb{E}\left[\prod_{i=1}^{n} X_i\right] = \prod_{i=1}^{n} \mathbb{E}[X_i]$

1.4.5.2 Varianza de una distribución

Varianza: $\sigma^2 = \mathbb{V}[X] = \mathbb{E}[(X - \mu)^2]$

$$\sigma^2 = \mathbb{V}[X] = \sum\nolimits_x (x - \mu)^2\, p(x) \qquad \text{para variables discretas (ordenadas)}$$

$$\sigma^2 = \mathbb{V}[X] = \int (x - \mu)^2\, p(x)\, dx \qquad \text{para variables continuas}$$

Propiedad: "media del cuadrado menos cuadrado de la media"

$$\begin{aligned}
\mathbb{V}[X] &= \mathbb{E}[(X - \mathbb{E}[X])^2] \\
&= \mathbb{E}[X^2 - 2X\mathbb{E}[X] + \mathbb{E}[X]^2] \\
&= \mathbb{E}[X^2] - 2\mathbb{E}[X]\mathbb{E}[X] + \mathbb{E}[X]^2 \\
&= \mathbb{E}[X^2] - \mathbb{E}[X]^2
\end{aligned}$$

Resultado útil: $\mathbb{E}[X^2] = \sigma^2 + \mu^2$

Desviación típica: $\sigma = \mathrm{std}[X] = \sqrt{\mathbb{V}[X]}$

Varianza de una variable aleatoria escalada y desplazada: $\mathbb{V}[aX+b]=a^2\mathbb{V}[X]$

Varianza de la suma de n variables aleatorias independientes: $\mathbb{V}\left[\sum_{i=1}^n X_i\right]=\sum_{i=1}^n\mathbb{V}[X_i]$

Varianza del producto de n variables aleatorias independientes:

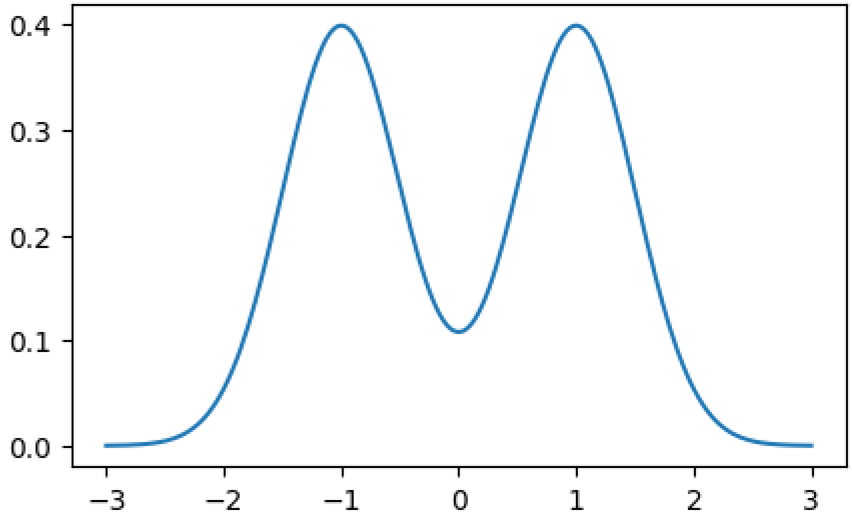
$$\begin{aligned}
\mathbb{V}\left[\prod_{i=1}^n X_i\right]&=\mathbb{E}\left[\left(\prod_i X_i\right)^2\right]-\left(\mathbb{E}\left[\prod_i X_i\right]\right)^2\\
&=\mathbb{E}\left[\prod_i X_i^2\right]-\left(\prod_i\mathbb{E}[X_i]\right)^2\\
&=\prod_i\mathbb{E}\left[X_i^2\right]-\prod_i(\mathbb{E}[X_i])^2\\
&=\prod_i\left(\mathbb{V}[X_i]+(\mathbb{E}[X_i])^2\right)-\prod_i(\mathbb{E}[X_i])^2\\
&=\prod_{i=1}^n(\sigma_i^2+\mu_i^2)-\prod_{i=1}^n\mu_i^2
\end{aligned}$$

1.4.5.3 Moda de una distribución

Moda: valor de máxima (densidad de) probabilidad, $\boldsymbol{x}^*=\operatorname*{argmax}_{\boldsymbol{x}}p(\boldsymbol{x})$

No unicidad de la moda: si la distribución es multimodal; por ejemplo

In [4]:
```
import numpy as np; import matplotlib.pyplot as plt; from scipy.stats import norm
X = np.linspace(-3, 3, 200); pdf = .5 * norm.pdf(X, loc=-1, scale=.5) + .5 * norm.pdf(X, loc=1, scale=.5)
fig = plt.subplots(figsize=(5,3)); plt.plot(X, pdf);
```



1.4.5.4 Momentos condicionales

Ley de la esperanza total o esperanzas iteradas: $\mathbb{E}[X] = \mathbb{E}_Y[\mathbb{E}[X \mid Y]]$

$$\mathbb{E}_Y[\mathbb{E}[X \mid Y]] = \sum_y \left[\sum_x x\, p(X = x \mid Y = y)\right] p(Y = y) = \sum_{x,y} x\, p(X = x, Y = y) = \sum_x x \sum_y p(X = x, Y = y)$$

Ejemplo: $X =$"vida de una bombilla en horas", $Y \in \{1, 2\}$ es la fábrica

- Fábrica $1:$ produce el 60% de las bombillas y duran 5000 horas de media
- Fábrica $2:$ produce el 40% de las bombillas y duran 4000 horas de media

$$\mathbb{E}[X] = \mathbb{E}[X \mid Y = 1]\, p(Y = 1) + \mathbb{E}[X \mid Y = 2]\, p(Y = 2) = 5000 \cdot 0.6 + 4000 \cdot 0.4 = 4600$$

Ley de la varianza total o fórmula de la varianza condicional: $\mathbb{V}[X] = \mathbb{E}_Y[\mathbb{V}[X \mid Y]] + \mathbb{V}_Y[\mathbb{E}[X \mid Y]]$

Demo: dados $\mu_{X|Y} = \mathbb{E}[X \mid Y],\quad s_{X|Y} = \mathbb{E}[X^2 \mid Y]$ y $\sigma^2_{X|Y} = \mathbb{V}[X \mid Y] = s_{X|Y} - \mu^2_{X|Y}$

$$\mathbb{V}[X] = \mathbb{E}[X^2] - (\mathbb{E}[X])^2 = \mathbb{E}_Y[s_{X|Y}] - (\mathbb{E}_Y[\mu_{X|Y}])^2 = \mathbb{E}_Y[\sigma^2_{X|Y}] + \mathbb{E}_Y[\mu^2_{X|Y}] - (\mathbb{E}_Y[\mu_{X|Y}])^2 = \mathbb{E}_Y[\mathbb{V}[X \mid Y]] + \mathbb{V}$$

Ejemplo: $X = \pi_1 \mathcal{N}(X \mid \mu_1, \sigma_1) + \pi_2 \mathcal{N}(X \mid \mu_2, \sigma_2)$, con $\pi_1 = \pi_2 = 0.5$, $\mu_1 = 0$, $\mu_2 = 2$ y $\sigma_1 = \sigma_2 = 0.5$; $Y \in \{1, 2\}$ es una variable oculta que indica la componente

$$\begin{aligned}
\mathbb{E}_Y[\mathbb{V}[X \mid Y]] &= \pi_1 \sigma_1^2 + \pi_2 \sigma_2^2 = 0.25 \\
\mathbb{V}_Y[\mathbb{E}[X \mid Y]] &= \pi_1(\mu_1 - \bar{\mu})^2 + \pi_2(\mu_2 - \bar{\mu})^2 = 0.5(0-1)^2 + 0.5(2-1)^2 = 0.5 + 0.5 = 1 \\
\mathbb{V}[X] &= 0.25 + 1 = 1.25
\end{aligned}$$

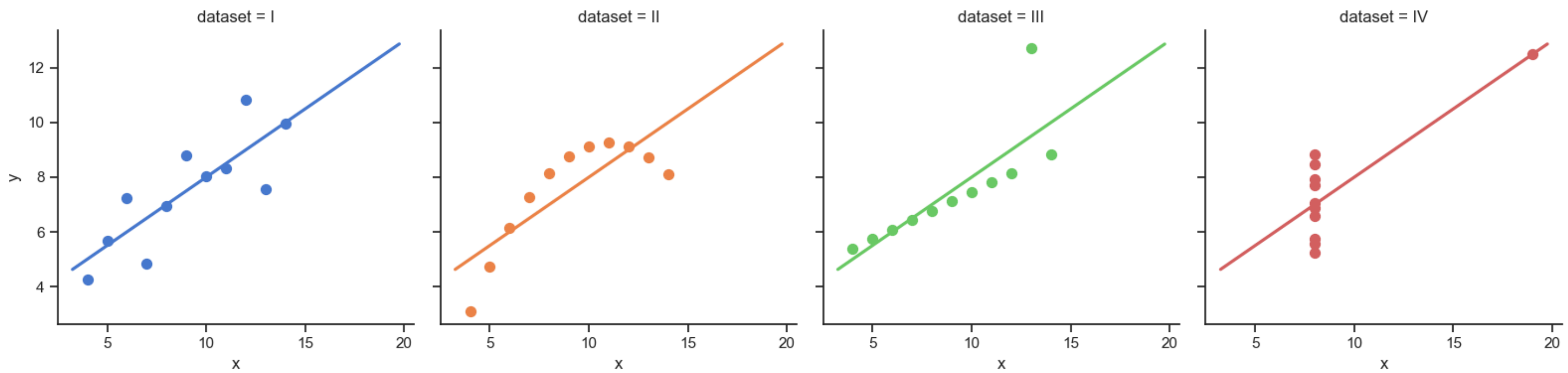
1.4.6 Limitaciones de la estadística descriptiva

Limitaciones de la estadística descriptiva: resumir una distribución con estadísticos simples (media y varianza) supone gran pérdida de información

Ejemplo: conjuntos de datos con mismas medias y varianzas

In [5]:
```
import warnings; warnings.filterwarnings('ignore'); import seaborn as sns
sns.set_theme(style="ticks"); df = sns.load_dataset("anscombe")
g = sns.lmplot(x="x", y="y", col="dataset", hue="dataset", data=df, col_wrap=4, ci=None, palette="muted",
               height=4, scatter_kws={"s": 50, "alpha": 1}, truncate=False); g.set(xlim=(2.5, 20.5))
for d in ['I', 'II', 'III', 'IV']:
    y = df[df['dataset'] == d]['y'].to_numpy()
    print(f'{d} {y.mean():.2f} {y.var():.2f}')
```

```
I 7.50 3.75
II 7.50 3.75
III 7.50 3.75
IV 7.50 3.75
```



2 Regla de Bayes

Variables: información desconocida u oculta (hidden) H y datos observados Y

Distribución a priori: $p(H)$ es nuestra creencia sobre H antes de observar datos

Distribución de observaciones: $p(Y \mid H = h)$ es la distribución sobre los posibles valores de Y que esperamos ver si $H = h$

Verosimilitud: $p(Y = y \mid H = h)$ es la evaluación de $p(Y \mid H = h)$ con las observaciones reales y

Distribución conjunta: $p(H = h, Y = y) = p(H = h)\, p(Y = y \mid H = h)$

Verosimilitud marginal: $p(Y = y)$ se obtiene marginalizando la conjunta

Distribución a posteriori o regla de Bayes: actualiza nuestra creencia sobre H tras observar datos

$$p(H = h|Y = y) = \frac{p(H = h, Y = y)}{p(Y = y)} = p(H = h)\, \frac{p(Y = y \mid H = h)}{p(Y = y)}$$

2.1 Ejemplo: Test de COVID-19

Variable oculta: $H=1$ infección positiva; $H=0$ infección negativa

Datos observados: $Y=1$ test positivo; $Y=0$ test negativo

Distribución a priori: $p(H=1)=1\%$ prevalencia de la infección

Verosimilitud:

$p(Y \mid H)$	$Y=0$	$Y=1$
$H=0$	0.975	0.025
$H=1$	0.125	0.875

$p(Y \mid H)$	$Y=0$	$Y=1$
$H=0$	True negative rate (TNR)	False positive rate (FPR)
$H=1$	False negative rate (FNR)	True positive rate (TPR)

Probabilidad de estar infectado si el test es positivo:

$$p(H=1 \mid Y=1)=p(H=1)\,\frac{p(Y=1 \mid H=1)}{p(Y=1)}=0.01\,\frac{0.875}{0.0335}=26\%$$

2.2 Ejemplo: El problema de Monty Hall

Problema de Monty Hall: se basa en el concurso televisivo *Let's Make a Deal* (1963--) presentado por Monty Hall durante 30 años

1. El concursante elige una puerta de tres
2. El presentador abre otra y aparece una cabra; en las dos que siguen cerradas hay un coche y otra cabra
3. El concursante puede cambiar de puerta: **¿cambia de puerta?**

Variable oculta: $H=i$ indica que el coche está en la puerta i

Distribución a priori: $P(H=1)=P(H=2)=P(H=3)=\frac{1}{3}$

Datos observados: $Y=j, j\in\{2,3\}$ es la puerta que escoge el presentador tras escoger la 1 el concursante

Distribución de observaciones:

H	$P(Y=2 \mid H)$	$P(Y=3 \mid H)$
1	$1/2$	$1/2$
2	0	1
3	1	0

Distribución conjunta y verosimilitud marginal:

H	$P(H,Y=2)$	$P(H,Y=3)$
1	$1/6$	$1/6$
2	0	$1/3$
3	$1/3$	0
$P(Y)$	$1/2$	$1/2$

Distribución a posteriori o regla de Bayes:

H	$P(H \mid Y=2)$	$P(H \mid Y=3)$
1	$1/3$	$1/3$
2	0	$2/3$
3	$2/3$	0

Conclusión: conviene cambiar ya que encontraremos el coche con probabilidad $2/3$

2.3 Problemas inversos

Teoría de la probabilidad: predicción de salidas y a partir de conocimiento o asunciones sobre el estado de la naturaleza, h

Teoría de la probabilidad inversa: predicción de estados de la naturaleza h a partir de observaciones sobre la salida, y

Ejemplos: inferir una forma 3d a partir de una imagen 2d en **comprensión de escenas visuales,** o la intención h de un locutor a partir de lo que dice en **comprensión del lenguaje natural**

Problemas inversos: usamos la regla de Bayes para calcular la posterior, $p(h|y)$, mediante un **modelo hacia adelante** $p(y|h)$ y un prior $p(h)$ que descarte estados de la naturaleza no plausibles

3 Distribuciones discretas

3.1 Distribución de Bernoulli

Distribución de Bernoulli: $Y \sim \operatorname{Ber}(\theta),\, \theta \in [0,1],$ si su pmf $p : \{0,1\} \to [0,1]$ es

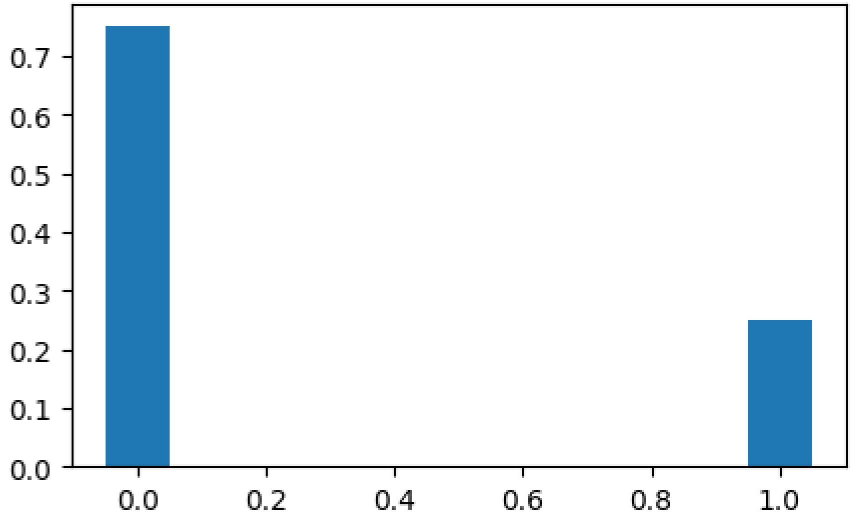
$$p(y \mid \theta) = \begin{cases} 1-\theta & \text{si } y=0 \\ \theta & \text{si } y=1 \end{cases} = \theta\, y + (1-\theta)(1-y) = \theta^y (1-\theta)^{1-y} \qquad (0^0 = 1,\, 0 \log 0 = 0)$$

Interpretación: Y es el resultado de un **experimento** con probabilidad de **éxito** $(Y=1)$ θ y probabilidad de **fracaso** $(Y=0)$ $1-\theta$

Ejemplo: $\theta = 0.25$

In [1]:
```
import matplotlib.pyplot as plt; from scipy.stats import bernoulli
t = 0.25; Y = bernoulli(t); print(Y.rvs(10)); y = [0, 1]
fig = plt.subplots(figsize=(5,3)); plt.bar(y, Y.pmf(y), width=0.1);
```

```
[0 0 0 1 0 1 0 0 0 0]
```



Media: $\mathbb{E}[Y] = 0\,p(0 \mid \theta) + 1\,p(1 \mid \theta) = 0\,(1-\theta) + 1\,\theta = \theta$

Media del cuadrado de una Bernoulli: $\mathbb{E}[Y^2] = 0^2\,p(0 \mid \theta) + 1^2\,p(1 \mid \theta) = \theta$

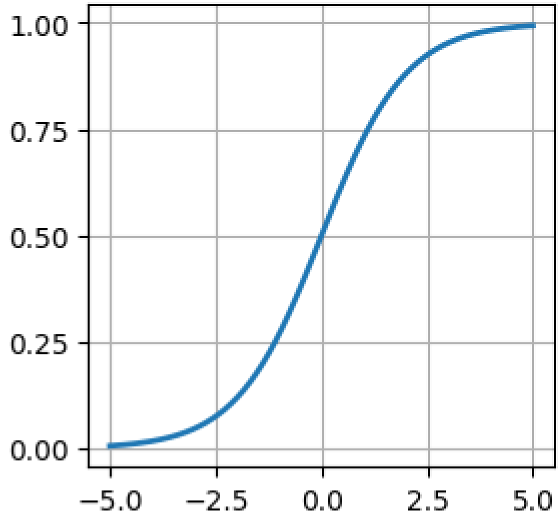
Varianza de una Bernoulli: $\mathbb{V}[Y] = \mathbb{E}[Y^2] - \mathbb{E}[Y]^2 = \theta - \theta^2 = \theta(1-\theta)$

3.2 Función logística o sigmoide

3.2.1 Función logística o sigmoide

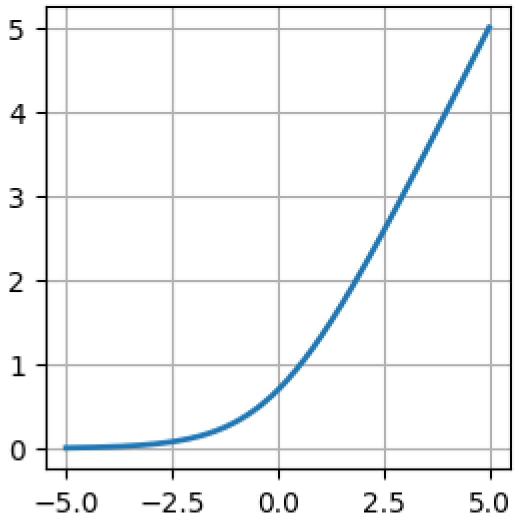
Función logística o sigmoide: función $\sigma : \mathbb{R} \to [0,1]$ con forma de S, $\sigma(a) = \dfrac{1}{1+e^{-a}} = \dfrac{e^a}{1+e^a}$

In [1]:
```
import numpy as np; import matplotlib.pyplot as plt
def sigmoid(a):
        return 1 / (1 + np.exp(-a))
fig = plt.subplots(figsize=(3,3)); plt.grid(); plt.yticks(np.arange(0, 1.1, step=0.25))
a = np.linspace(-5, 5, 200); plt.plot(a, sigmoid(a), linewidth=2);
```



Función softplus: $\sigma_+(a) = \log(1 + e^a)$

In [2]:
```
import numpy as np; import matplotlib.pyplot as plt
def softplus(a):
        return np.log1p(np.exp(a))
fig = plt.subplots(figsize=(3,3)); plt.grid();
a = np.linspace(-5, 5, 200); plt.plot(a, softplus(a), linewidth=2);
```



3.2.2 Función logit

Función logit: función $\operatorname{logit}:[0,1]\to\mathbb{R}$ inversa de la sigmoide, $\operatorname{logit}(p)=\log\left(\dfrac{p}{1-p}\right)$

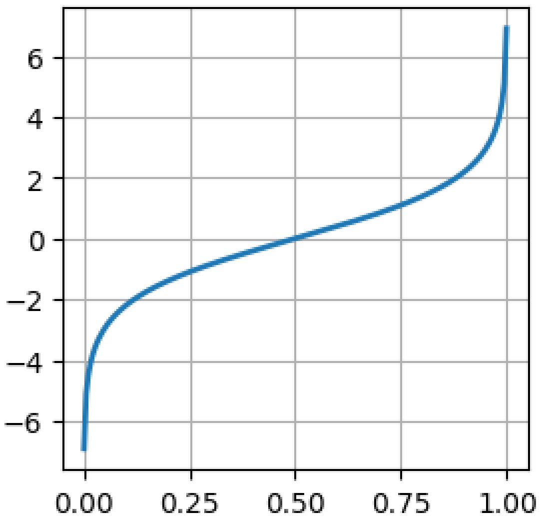
Interpretación: si p es la probabilidad de que un cierto evento ocurra, su logit es la log-posibilidad (log-odds) de que el evento ocurra frente a que no ocurra, por lo que tenemos tres casos

1. Más posibilidades de ocurrir que de no: odds mayor que 1 y log-odds positiva
2. Igual posibilidades de ocurrir que de no: odds 1 y log-odds nula
3. Menos posibilidades de ocurrir que de no: odds menor que 1 y log-odds negativa

Inversa de la sigmoide:

$$\begin{aligned}
\sigma(a)&=\frac{1}{1+e^{-a}}=\frac{e^a}{1+e^a}\\
1-\sigma(a)&=1-\frac{1}{1+e^{-a}}=\frac{e^{-a}}{1+e^{-a}}=\frac{1}{1+e^a}=\sigma(-a)\\
\operatorname{logit}(\sigma(a))&=\log\left(\frac{\sigma(a)}{1-\sigma(a)}\right)=\log\left(\frac{e^a}{1+e^a}\,\frac{1+e^a}{1}\right)=\log(e^a)=a
\end{aligned}$$

In [3]:
```
import numpy as np; import matplotlib.pyplot as plt
def logit(p):
        return np.log(p / (1 - p))
fig = plt.subplots(figsize=(3,3)); plt.grid(); plt.xticks(np.arange(0, 1.1, step=0.25))
p = np.linspace(.001, .999, 200); plt.plot(p, logit(p), linewidth=2);
```



Recordatorio: **la sigmoide transforma log-odds en probabilidad y la logit probabilidad en log-odds**

3.3 Codificación one-hot y distribución categórica

Propósito: generalizar la Bernoulli a $C > 2$ clases, esto es, una distribución sobre un conjunto finito de etiquetas $\mathcal{C} = \{1, \dots, C\}$

Codificación one-hot: de una variable categórica $y \in \{1, \dots, C\}$

$$\text{one-hot}(y) = \boldsymbol{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_C \end{pmatrix} = \begin{pmatrix} \mathbb{I}(y = 1) \\ \vdots \\ \mathbb{I}(y = C) \end{pmatrix} \in \{0, 1\}^C \quad \text{con} \quad \sum_c y_c = 1$$

Distribución categórica: $Y \sim \operatorname{Cat}(\boldsymbol{\theta}),\, \boldsymbol{\theta} \in [0, 1]^C,\, \sum_c \theta_c = 1,$ si su pmf $p : \mathcal{C} \to [0, 1]$ es

$$p(y \mid \theta) = \prod_{c=1}^{C} \theta_c^{\mathbb{I}(y=c)} \qquad \text{o, en notación one-hot,} \qquad p(\boldsymbol{y} \mid \theta) = \prod_{c=1}^{C} \theta_c^{y_c}$$

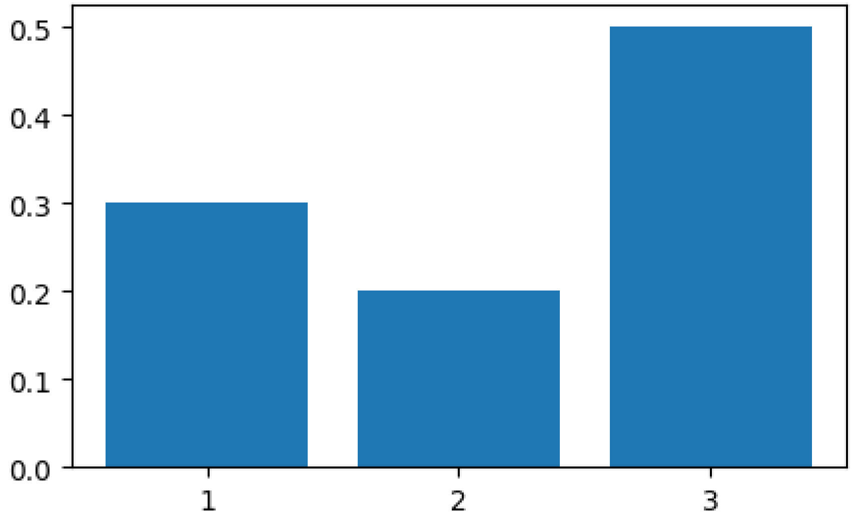
Interpretación: θ_c es la probabilidad de que y valga c, $p(y = c \mid \boldsymbol{\theta}) = \theta_c$

Sobre-parametrización: solo tenemos $C-1$ parámetros libres por las restricciones sobre $\boldsymbol{\theta}$

Ejemplo: $C=3, \boldsymbol{\theta}^t=(0.3, 0.2, 0.5)$

In [1]:
```
import matplotlib.pyplot as plt; from scipy.stats import multinomial
t = [0.3, 0.2, 0.5]; Y = multinomial(1, t); print(Y.p, Y.rvs(3))
fig = plt.subplots(figsize=(5,3)); plt.xticks(range(1, 4)); plt.bar(range(1, 4), Y.p);
```

```
[0.3 0.2 0.5] [[0 0 1]
 [1 0 0]
 [1 0 0]]
```



Convención: $0^0=1$ i $0\log 0=0$; por ejemplo, con
$\boldsymbol{\theta}=(0.5, 0.5, 0)^t,\ \operatorname{Cat}(\boldsymbol{y}=(1,0,0)^t \mid \boldsymbol{\theta})=0.5^1 0.5^0 0^0=0.5$

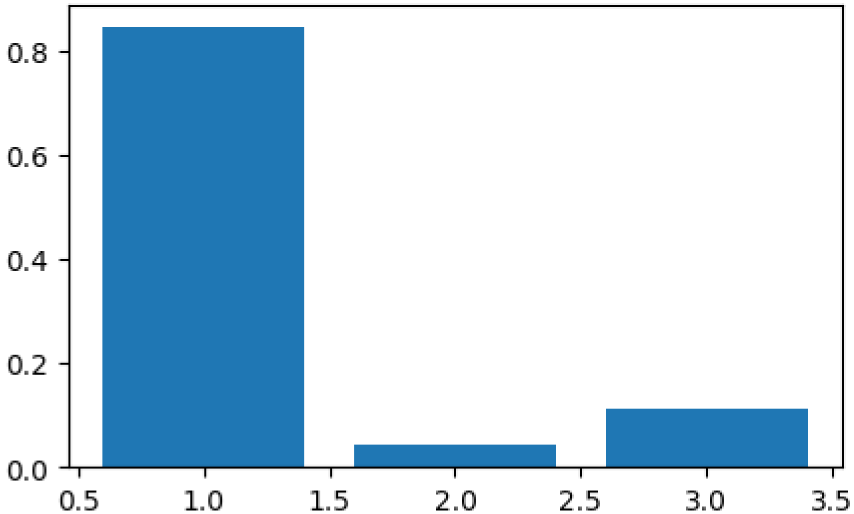
3.4 La función softmax

Función softmax: transforma logits $\boldsymbol{a}\in\mathbb{R}^C$ en un vector de probabilidades $[0,1]^C$

$$\mathcal{S}(\boldsymbol{a})=\left(\frac{e^{a_1}}{\sum_{\tilde{c}} e^{a_{\tilde{c}}}}, \dots, \frac{e^{a_C}}{\sum_{\tilde{c}} e^{a_{\tilde{c}}}}\right)^t \qquad \text{cumpliéndose} \qquad 0\leq\mathcal{S}(\boldsymbol{a})_c\leq 1 \quad \text{y} \quad \sum_{c=1}^C \mathcal{S}(\boldsymbol{a})_c=1$$

Ejemplo: $\boldsymbol{a} = (3, 0, 1)^t$, $\mathcal{S}(\boldsymbol{a}) = \left(\frac{e^3}{e^3+1+e}, \frac{1}{e^3+1+e}, \frac{e}{e^3+1+e}\right)^t = (0.8438, 0.0420, 0.1142)^t$

In [1]:
```
import numpy as np; import matplotlib.pyplot as plt
def softmax(a):
        e = np.exp((1.0 * np.array(a))); return e / np.sum(e)
a = np.array([3, 0, 1]); fig = plt.subplots(figsize=(5,3));
plt.bar(np.arange(1, a.size+1), softmax(a));
```

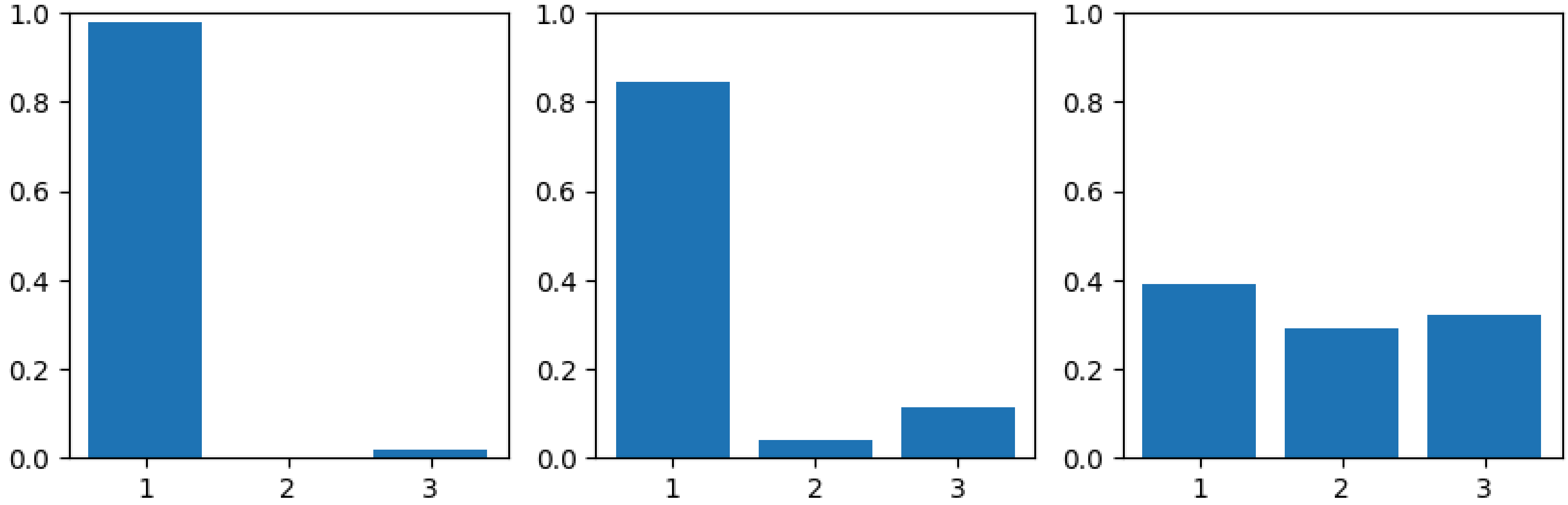


Softmax atemperada: normaliza logits mediante división por una constante de **temperatura** $T > 0$

- **Bajas temperaturas:** tiende a la argmax, $\lim_{T\to 0^+} \mathcal{S}(\boldsymbol{a}/T)_c = \mathbb{I}(c = \operatorname{argmax}_{c'} a_{c'})$
- **Altas temperaturas:** tiende a la uniforme, $\lim_{T\to\infty} \mathcal{S}(\boldsymbol{a}/T)_c = 1/C$

Ejemplo: $\boldsymbol{a} = (3, 0, 1)^t$, $T \in \{0.5, 1, 10\}$

In [2]:
```
import numpy as np; import matplotlib.pyplot as plt
def softmax(a):
        e = np.exp((1.0 * np.array(a))); return e / np.sum(e)
a = np.array([3, 0, 1]); fig, axs = plt.subplots(1, 3, figsize=(10,3));
for i, T in enumerate((0.5, 1, 10)):
        axs[i].set_ylim((0, 1)); axs[i].bar(np.arange(1, a.size+1), softmax(a/T));
```



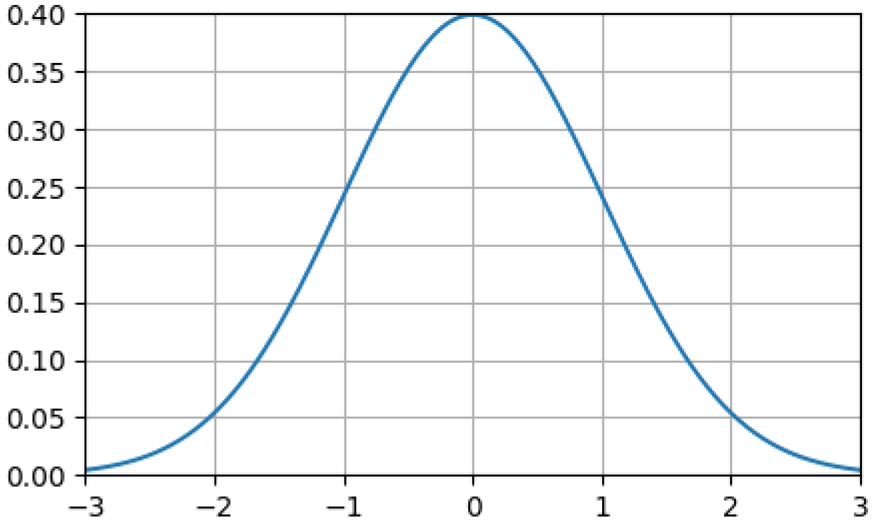
4 Distribuciones continuas

4.1 Gaussiana univariada

Función de densidad de probabilidad (pdf) Gaussiana:

$$Y \sim \mathcal{N}(y \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(y-\mu)^2\right)$$

In [1]:
```
import numpy as np; import matplotlib.pyplot as plt; from scipy.stats import norm
Y = norm(0, 1); y = np.linspace(-3, 3, 200); fig = plt.subplots(figsize=(5,3))
plt.grid(); plt.xlim(-3, 3); plt.ylim(0, .4); plt.plot(y, Y.pdf(y));
```



4.2 Covarianza

Covarianza entre dos variables aleatorias X e Y:

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

Interpretación de la covarianza: mide hasta qué grado X e Y están (linealmente) relacionadas

Matriz de covarianzas de un vector aleatorio D-dimensional $\boldsymbol{x}$: matriz **simétrica** y **semi-definida positiva**

$$\boldsymbol{\Sigma}=\mathrm{Cov}[\boldsymbol{x}]=\mathbb{E}\left[(\boldsymbol{x}-\mathbb{E}[\boldsymbol{x}])(\boldsymbol{x}-\mathbb{E}[\boldsymbol{x}])^t\right]=\begin{pmatrix}\mathbb{V}[X_1] & \mathrm{Cov}[X_1,X_2] & \cdots & \mathrm{Cov}[X_1,X_D]\\ \mathrm{Cov}[X_2,X_1] & \mathbb{V}[X_2] & \cdots & \mathrm{Cov}[X_2,X_D]\\ \vdots & \vdots & \ddots & \vdots\\ \mathrm{Cov}[X_D,X_1] & \mathrm{Cov}[X_D,X_2] & \cdots & \mathbb{V}[X_D]\end{pmatrix}$$

Resultado importante: $\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^t]=\boldsymbol{\Sigma}+\boldsymbol{\mu}\boldsymbol{\mu}^t$

Covarianza de una transformación lineal: $\mathrm{Cov}[\mathbf{A}\boldsymbol{x}+\boldsymbol{b}]=\mathbf{A}\,\mathrm{Cov}[\boldsymbol{x}]\mathbf{A}^t$

Covarianza cruzada entre dos vectores aleatorios: $\operatorname{Cov}[\boldsymbol{x},\boldsymbol{y}]=\mathbb{E}[(\boldsymbol{x}-\mathbb{E}[\boldsymbol{x}])(\boldsymbol{y}-\mathbb{E}[\boldsymbol{y}])^t]$

4.3 Gaussiana multivariada

4.3.1 Definición

Gaussiana multivariada: $\boldsymbol{x}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma}),$ con $\boldsymbol{x}\in\mathbb{R}^D,$ **media** $\boldsymbol{\mu}=\mathbb{E}[\boldsymbol{x}]\in\mathbb{R}^D$ y **matriz de covarianzas** $\boldsymbol{\Sigma}=\operatorname{Cov}[\boldsymbol{x}]\in\mathbb{R}^{D\times D}$

$$p(\boldsymbol{x}\mid\boldsymbol{\mu},\boldsymbol{\Sigma})=\frac{1}{(2\pi)^{D/2}|\boldsymbol{\Sigma}|^{1/2}}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]$$

Gaussiana bivariada: $\boldsymbol{x}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma}),$ con $\boldsymbol{x},\boldsymbol{\mu}\in\mathbb{R}^2$ y $\boldsymbol{\Sigma}=\begin{pmatrix}\sigma_1^2&\sigma_{12}^2\\\sigma_{21}^2&\sigma_2^2\end{pmatrix}=\begin{pmatrix}\sigma_1^2&\rho\sigma_1\sigma_2\\\rho\sigma_1\sigma_2&\sigma_2^2\end{pmatrix}$ con $\rho=\dfrac{\sigma_{12}^2}{\sigma_1\sigma_2}$

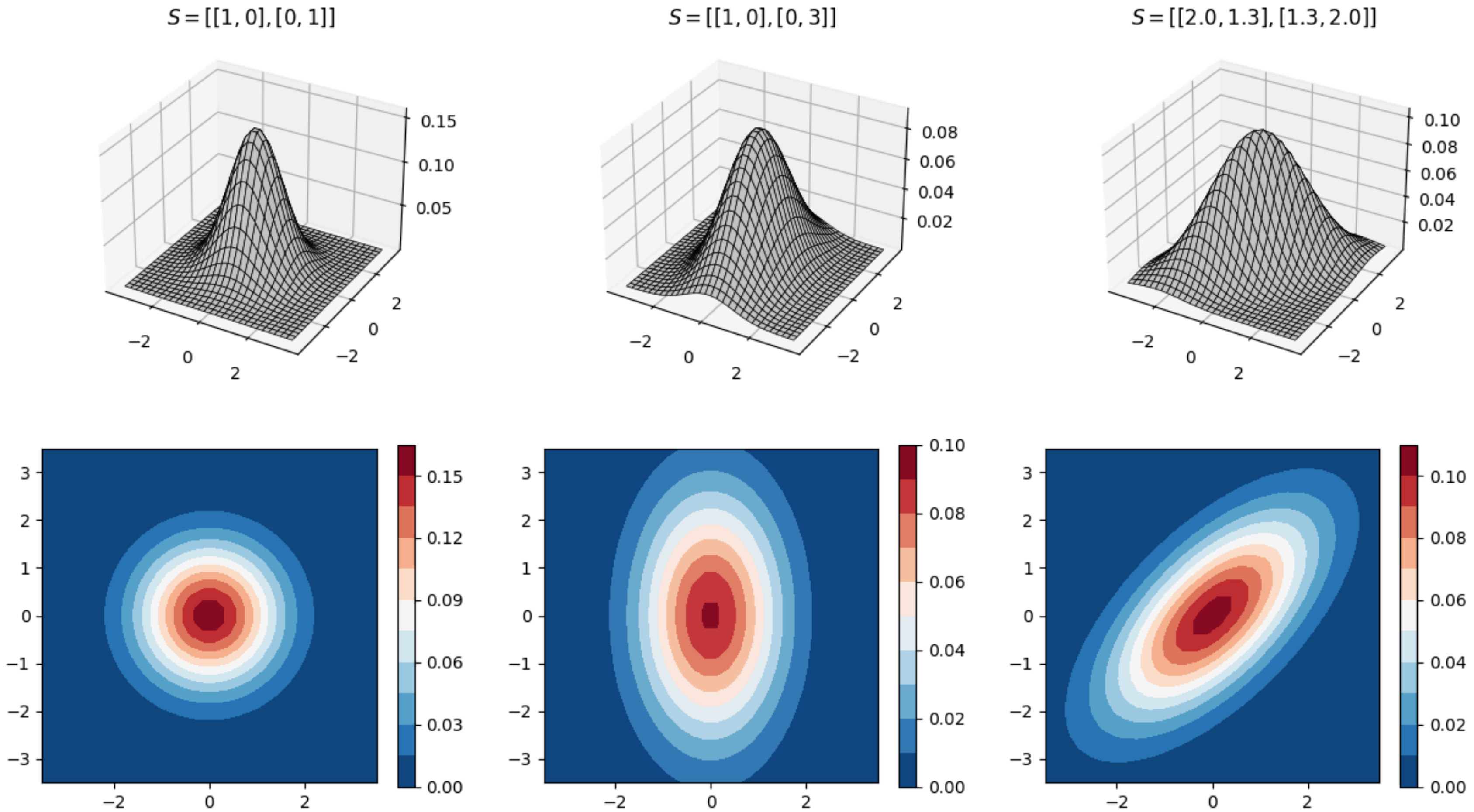
$$p(x_1,x_2)=\frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}}\exp\left(-\frac{1}{2(1-\rho^2)}\left[\frac{(x_1-\mu_1)^2}{\sigma_1^2}+\frac{(x_2-\mu_2)^2}{\sigma_2^2}-2\rho\,\frac{(x_1-\mu_1)}{\sigma_1}\,\frac{(x_2-\mu_2)}{\sigma_2}\right]\right)$$

Tipos de Gaussianas según estructura de $\boldsymbol{\Sigma}$:

- **Esférica:** $\boldsymbol{\Sigma}=\sigma^2\mathbf{I},$ con un único parámetro y curvas de iso-densidad hiper-esféricas
- **Diagonal:** $\boldsymbol{\Sigma}=\operatorname{diag}(\sigma_1^2,\dots,\sigma_D^2),$ con D parámetros y curvas de iso-densidad hiper-elipsoidales de semiejes paralelos a la base
- **General:** $\boldsymbol{\Sigma}$ no diagonal, con $D(D+1)/2$ parámetros y curvas de iso-densidad hiper-elipsoidales de semiejes oblicuos a la base

In [1]:
```
import numpy as np; import matplotlib.pyplot as plt; from scipy.stats import multivariate_normal
R = np.linspace(-3.5, 3.5, 30); x, y = np.meshgrid(R, R); me, Se = [0, 0], [[1, 0], [0, 1]]
md, Sd = [0, 0], [[1, 0], [0, 3]]; mg, Sg = [0, 0], [[2., 1.3], [1.3, 2.]]
```

```
fig = plt.figure(figsize=(15, 8)); fig.tight_layout()
for i, (m, S) in enumerate(zip((me, md, mg), (Se, Sd, Sg)), start=1):
    z = multivariate_normal(m, S).pdf(np.dstack((x, y)))
    ax = fig.add_subplot(2, 3, i, projection='3d'); ax.set_title(f'$S={S}$'.format(S))
    ax.plot_surface(x, y, z, color='white', edgecolor="black", lw=.5)
    ax = fig.add_subplot(2, 3, i+3, aspect='equal')
    cp = ax.contourf(x, y, z, 10, cmap='RdBu_r'); plt.colorbar(cp, ax=ax);
```



4.3.2 Simulación

Gaussiana general como afinidad de la estándar:

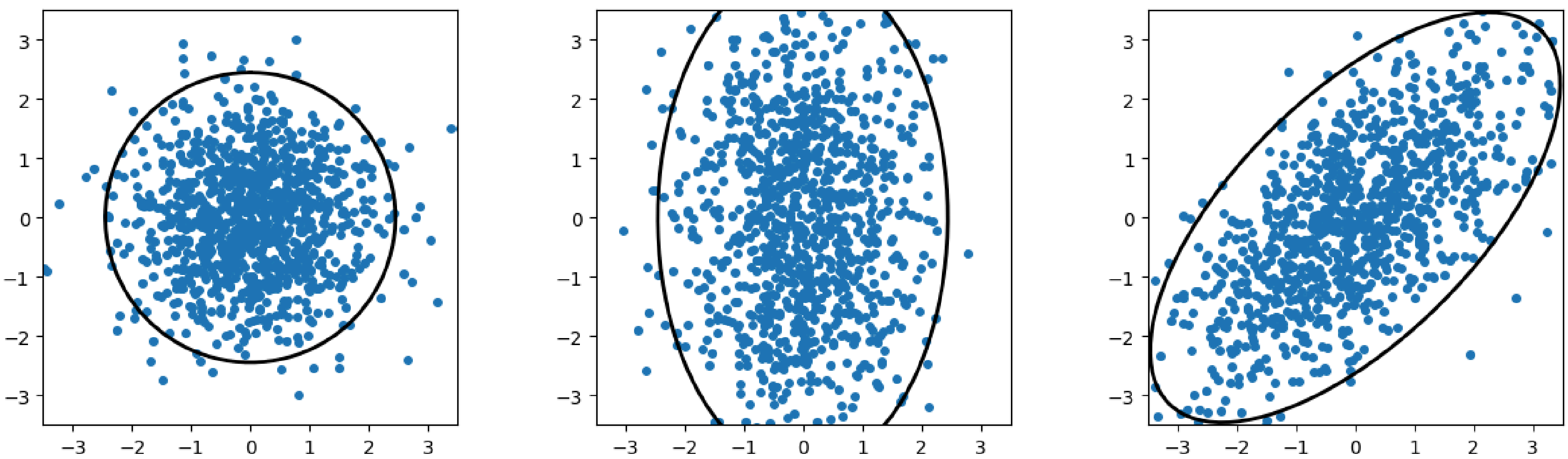
$$\boldsymbol{x} \sim \mathcal{N}_D(\mathbf{0}, \mathbf{I}), \quad \boldsymbol{y} = \mathbf{W}\boldsymbol{x} + \boldsymbol{\mu} \quad \rightarrow \quad \boldsymbol{x} = \mathbf{W}^{-1}(\boldsymbol{y} - \boldsymbol{\mu})$$

$$\begin{aligned}
p_{\boldsymbol{y}}(\boldsymbol{y}) &= p_{\boldsymbol{x}}(\mathbf{W}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}))\,|\det(\mathbf{W}^{-1})| \\
&= \frac{1}{(2\pi)^{D/2}|\det(\mathbf{W})|}\exp\left[-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu})^t\mathbf{W}^{-t}\mathbf{W}^{-1}(\boldsymbol{y}-\boldsymbol{\mu})\right] \\
&= \frac{1}{(2\pi)^{D/2}|\boldsymbol{\Sigma}|^{1/2}}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right] \quad \text{con} \quad \boldsymbol{\Sigma}=\mathbf{W}\mathbf{W}^t
\end{aligned}$$

Bola Gaussiana estándar de masa p: $\quad P(\|\boldsymbol{x}\|_2^2 \leq r) = P(\chi_D^2 \leq r) = p$

Bola Gaussiana estándar 2d de masa p: $\quad r = \sqrt{-2\log(1-p)} \qquad p = 1 - e^{-r^2/2}$

In [2]:
```
import numpy as np; import matplotlib.pyplot as plt; from scipy.stats import multivariate_normal
me, Se = [0, 0], [[1, 0], [0, 1]]; md, Sd = [0, 0], [[1, 0], [0, 3]]; mg, Sg = [0, 0], [[2., 1.3], [1.3, 2
p = .95; r = np.sqrt(-2.0*np.log(1.0-p)); t = np.linspace(0, 2.0*np.pi, 100);
C = np.array([np.cos(t), np.sin(t)]) * r; fig = plt.figure(figsize=(15, 4)); fig.tight_layout()
for i, (m, S) in enumerate(zip((me, md, mg), (Se, Sd, Sg)), start=1):
    ax = fig.add_subplot(1, 3, i, aspect='equal'); ax.set_xlim(-3.5, 3.5); ax.set_ylim(-3.5, 3.5)
    X = multivariate_normal(m, S).rvs(1000); ax.scatter(*X.T, s=16)
    La, U = np.linalg.eigh(S); k = La.argsort()[::-1]; La = La[k]; U = U[:,k]; W = U @ np.diag(np.sqrt(La)
    Y = W @ C; ax.plot(*Y, lw=2, color='black')
```



4.4 Distancia de Mahalanobis

Distancia de Mahalanobis entre $\boldsymbol{y}$ y $\boldsymbol{\mu}$ con respecto a $\mathbf{\Sigma}^{-1}$:

$$\Delta(\boldsymbol{y},\boldsymbol{\mu};\mathbf{\Sigma}^{-1})=\sqrt{(\boldsymbol{y}-\boldsymbol{\mu})^t\mathbf{\Sigma}^{-1}(\boldsymbol{y}-\boldsymbol{\mu})}$$

Gaussiana multivariada en términos de Mahalanobis (al cuadrado):

$$p(\boldsymbol{y}\mid\boldsymbol{\mu},\mathbf{\Sigma})=\frac{1}{(2\pi)^{D/2}|\mathbf{\Sigma}|^{1/2}}\exp\left[-\frac{1}{2}\Delta^2(\boldsymbol{y},\boldsymbol{\mu};\mathbf{\Sigma}^{-1})\right]$$

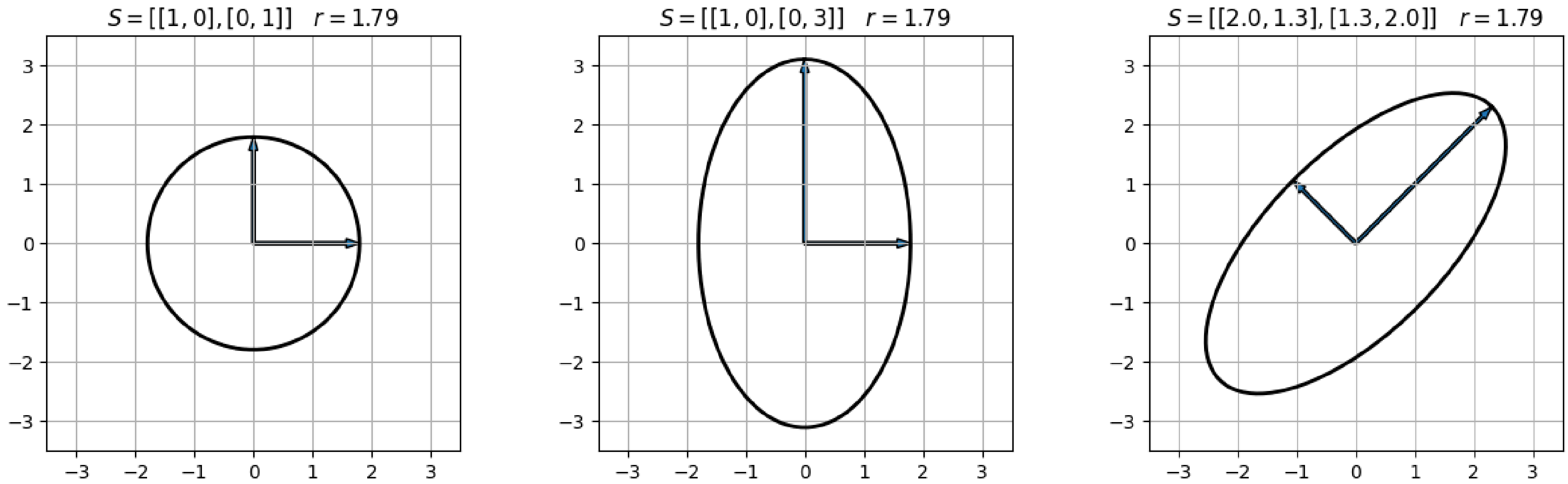
Mahalanobis como afinidad de la Euclídea (al origen):

$$\boldsymbol{x}=(r\cos\theta,r\sin\theta),\quad \mathbf{\Sigma}=\mathbf{W}\mathbf{W}^t,\quad \boldsymbol{y}=\mathbf{W}\boldsymbol{x}+\boldsymbol{\mu}$$

$$\Delta^2(\boldsymbol{y},\boldsymbol{\mu};\mathbf{\Sigma}^{-1})=(\boldsymbol{y}-\boldsymbol{\mu})^t\mathbf{\Sigma}^{-1}(\boldsymbol{y}-\boldsymbol{\mu})=(\mathbf{W}\boldsymbol{x})^t\mathbf{W}^{-t}\mathbf{W}^{-1}\mathbf{W}\boldsymbol{x}=\|\boldsymbol{x}\|_2^2=r^2$$

In [1]:

```
import numpy as np; import matplotlib.pyplot as plt; from scipy.stats import multivariate_normal
me, Se = [0, 0], [[1, 0], [0, 1]]; md, Sd = [0, 0], [[1, 0], [0, 3]]; mg, Sg = [0, 0], [[2., 1.3], [1.3, 2
p = .8; r = np.sqrt(-2.0*np.log(1.0-p)); t = np.linspace(0, 2.0*np.pi, 100);
C = np.array([np.cos(t), np.sin(t)]) * r; fig = plt.figure(figsize=(15, 4)); fig.tight_layout()
for i, (m, S) in enumerate(zip((me, md, mg), (Se, Sd, Sg)), start=1):
    ax = fig.add_subplot(1, 3, i, aspect='equal'); ax.set_xlim(-3.5, 3.5); ax.set_ylim(-3.5, 3.5); ax.grid
    La, U = np.linalg.eigh(S); k = La.argsort()[::-1]; La = La[k]; U = U[:,k]; W = U @ np.diag(np.sqrt(La)
    Y = W @ C; ax.plot(*Y, lw=2, color='black'); ax.set_title(f'$S={S}\quad r={r:.2f}$')
    ax.arrow(0, 0, *r*W[:, 0], width=.05, length_includes_head=True)
    ax.arrow(0, 0, *r*W[:, 1], width=.05, length_includes_head=True)
```


$S = [[1, 0], [0, 1]] \quad r = 1.79$
$S = [[1, 0], [0, 3]] \quad r = 1.79$
$S = [[2.0, 1.3], [1.3, 2.0]] \quad r = 1.79$