T2 Decisión

Índice

1 Introducción (revisión de PER)

Inferencia Bayesiana: cálculo de la **posterior** $p(H \mid \boldsymbol{x})$ mediante la regla de Bayes actualizar nuestras creencias sobre cantidades ocultas H a partir de datos $\boldsymbol{x}$

Teoría de la decisión Bayesiana: usa la inferencia para decidir cuál es la mejor de las posibles **acciones** a realizar

Agente: debe escoger una acción de un conjunto de acciones posibles, $\mathcal{A}$

Estado de la naturaleza: $h \in \mathcal{H}$, condiciona los costes y beneficios que se derivan de tomar cada acción posible

Función de pérdida: indica el coste incurrido al tomar la acción $a \in \mathcal{A}$ cuando el estado de la naturaleza es $h \in \mathcal{H}$

$$\ell(h, a)$$

Riesgo (pérdida) esperado a posteriori: de a tras observar $\boldsymbol{x}$

$$R(a \mid \boldsymbol{x}) = \mathbb{E}_{p(h|\boldsymbol{x})}[\ell(h, a)] = \sum_{h \in \mathcal{H}} \ell(h, a)\, p(h \mid \boldsymbol{x})$$

Política óptima o estimador de Bayes: obtiene una acción de mínimo riesgo por cada observación posible

$$\pi^*(\boldsymbol{x}) = \operatorname*{argmin}_{a \in \mathcal{A}} \; R(a \mid \boldsymbol{x})$$

Problemas de clasificación

Estados de la naturaleza y acciones: etiquetas de clase, $\mathcal{H} = \mathcal{Y} = \{1, \dots, C\}$ y $\mathcal{A} = \mathcal{Y}$

Pérdida 01 esperada a posteriori: la probabilidad de error a posteriori es uno menos la de acertar a posteriori

$$R(\hat{y} \mid \boldsymbol{x}) = \sum_{y} \ell_{01}(y, \hat{y})\, p(y \mid \boldsymbol{x}) = \sum_{y \neq \hat{y}} p(y \mid \boldsymbol{x}) = 1 - p(\hat{y} \mid \boldsymbol{x})$$

Esstimador de Bayes: $\pi(\boldsymbol{x}) = \operatorname{argmax}_{y \in \mathcal{Y}}\ p(y \mid \boldsymbol{x})$ **estimador MAP** o **moda** de la posterior

Matriz de confusión: $\mathcal{D} = \{(y_m, \hat{y}_m)\}$, $\mathbf{M} = [M_{y,\hat{y}}]$ con $M_{y,\hat{y}} = \sum_m \mathbb{I}(y_m = y)\mathbb{I}(\hat{y}_m = \hat{y})$

y	$\hat{1}$	$\hat{2}$	$\cdots$	$\hat{C}$	Suma fila
1	$M_{1,\hat{1}}$	$M_{1,\hat{2}}$	$\cdots$	$M_{1,\hat{C}}$	$M_{1,:}$
2	$M_{2,\hat{1}}$	$M_{2,\hat{2}}$	$\cdots$	$M_{2,\hat{C}}$	$M_{2,:}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
C	$M_{C,\hat{1}}$	$M_{C,\hat{2}}$	$\cdots$	$M_{C,\hat{C}}$	$M_{C,:}$
Suma:	$M_{:,\hat{1}}$	$M_{:,\hat{2}}$	$\cdots$	$M_{:,\hat{C}}$	M

Normalización por filas: estimación empírica de $p(\hat{y} \mid y)$

Normalización por columnas: estimación empírica de $p(y \mid \hat{y})$

Normalización por filas y columnas: estimación empírica de $p(y, \hat{y})$

Análisis de una clase específica: se reduce a matriz binaria tomando el resto de clases como clase negativa

Problemas de regresión

Estados de la naturaleza y acciones: reales, $\mathcal{H} = \mathcal{A} = \mathcal{Y} = \mathbb{R}$

Pérdida L2 (ℓ_2, cuadrática o error cuadrático): $\ell_2(h, a) = (h - a)^2$

Pérdida L2 esperada a posteriori:

$$R(a \mid \boldsymbol{x}) = \mathbb{E}[(h - a)^2 \mid \boldsymbol{x}] = \mathbb{E}[h^2 \mid \boldsymbol{x}] - 2a\mathbb{E}[h \mid \boldsymbol{x}] + a^2$$

Regresor de Bayes L2 o minimum mean squared error (MMSE): media a posteriori

$$\frac{\partial}{\partial a} R(a \mid \boldsymbol{x}) = -2\mathbb{E}[h \mid \boldsymbol{x}] + 2a = 0 \quad \rightarrow \quad \pi(\boldsymbol{x}) = \mathbb{E}[h \mid \boldsymbol{x}] = \int h \, p(h \mid \boldsymbol{x}) \, dh$$

2 Ejercicio

Agente: un médico debe tratar un paciente que podría tener COVID y las posibles acciones son: no hacer nada o dar una medicina con efectos secundarios graves, pero que podría salvarle la vida

Estado de la naturaleza: el estado viene dado por la edad del paciente (joven o mayor) y si tiene COVID o no; se trata de un estado *parcialmente observado* ya que podemos observar la edad del paciente directamente, pero no si tiene COVID o no, cosa que modelizamos probabilísticamente mediante un test

Función de pérdida: medimos costes en quality-adjusted life years (QALYs); asumimos que el coste de dar una medicina es de 8 QALYs, independientemente del estado del paciente; sin embargo, el coste de no dar nada a un paciente con COVID se considera de 60 QALYs si es joven; 10 si es mayor

Estado	nada	med.
Joven, no COVID	0	8
Joven, COVID	60	8
Mayor, no COVID	0	8
Mayor, COVID	10	8

Ejercicio: Por cada posible observación parcial de estado, halla el riesgo esperado a posteriori de cada acción y determina una acción de menor riesgo; asume que la observación parcial de estado viene representada por un vector de datos $\boldsymbol{x}$ que incluye edad (joven o mayor) y resultado del test (0 o 1). Si el test resulta positivo (1), tanto si es joven como mayor, la probabilidad de que tenga COVID es 0.795455. Si resulta negativo, la probabilidad de que tenga COVID es 0.014045.

Solución:

edad	test	pr(covid)	nada	med.	acción
joven	0	0.014045	0.842697	8.00	nada
joven	1	0.795455	47.727273	8.00	med.
mayor	0	0.014045	0.140449	8.00	nada
mayor	1	0.795455	7.954545	8.00	nada

$$\mathcal{H} = \{(J,\bar{C}), (J,C), (M,\bar{C}), (M,C)\} \qquad \mathcal{A} = \{N, M\}$$

$$R(N \mid J0) = \ell(J\bar{C}, N)\, p(J\bar{C} \mid J0) + \ell(JC, N)\, p(JC \mid J0) = 0 \cdot (1 - 0.014045) + 60 \cdot 0.014045 = 0.8427$$
$$R(M \mid J0) = \ell(J\bar{C}, M)\, p(J\bar{C} \mid J0) + \ell(JC, M)\, p(JC \mid J0) = 8 \cdot (1 - 0.014045) + 8 \cdot 0.014045 = 8$$

$$R(N \mid J1) = \ell(J\bar{C}, N)\, p(J\bar{C} \mid J1) + \ell(JC, N)\, p(JC \mid J1) = 0 \cdot (1 - 0.795455) + 60 \cdot 0.795455 = 47.7273$$
$$R(M \mid J1) = \ell(J\bar{C}, M)\, p(J\bar{C} \mid J1) + \ell(JC, M)\, p(JC \mid J1) = 8 \cdot (1 - 0.795455) + 8 \cdot 0.795455 = 8$$

$$R(N \mid M0) = \ell(M\bar{C}, N)\, p(M\bar{C} \mid M0) + \ell(MC, N)\, p(MC \mid M0) = 0 \cdot (1 - 0.014045) + 10 \cdot 0.014045 = 0.140450$$
$$R(M \mid M0) = \ell(M\bar{C}, M)\, p(M\bar{C} \mid M0) + \ell(MC, M)\, p(MC \mid M0) = 8 \cdot (1 - 0.014045) + 8 \cdot 0.014045 = 8$$

$$R(N \mid M1) = \ell(M\bar{C}, N)\, p(M\bar{C} \mid M1) + \ell(MC, N)\, p(MC \mid M1) = 0 \cdot (1 - 0.795455) + 10 \cdot 0.795455 = 7.95455$$
$$R(M \mid M1) = \ell(M\bar{C}, M)\, p(M\bar{C} \mid M1) + \ell(MC, M)\, p(MC \mid M1) = 8 \cdot (1 - 0.795455) + 8 \cdot 0.795455 = 8$$

3 Clasificación sensible al coste

Pérdida para dos clases con costes distintos: $\mathcal{Y} = \{0, 1\}$

$$\ell(y^*, \hat{y}) = \begin{pmatrix} \ell_{00} & \ell_{01} \\ \ell_{10} & \ell_{11} \end{pmatrix}$$

Pérdida esperada a posteriori: $p_0 = p(y^* = 0 \mid \boldsymbol{x})$ y $p_1 = 1 - p_0$

$$R(\hat{y} \mid \boldsymbol{x}) = \sum_{y^*} \ell(y^*, \hat{y})\, p(y^* \mid \boldsymbol{x}) = \begin{cases} \ell_{00}\, p_0 + \ell_{10}\, p_1 & \text{si } \hat{y} = 0 \\ \ell_{01}\, p_0 + \ell_{11}\, p_1 & \text{si } \hat{y} = 1 \end{cases}$$

Estimador de Bayes:

$$\pi^*(\boldsymbol{x}) = \begin{cases} 0 & \text{si } \ell_{00}\, p_0 + \ell_{10}\, p_1 < \ell_{01}\, p_0 + \ell_{11}\, p_1 \\ 1 & \text{si no} \end{cases}$$

Coste de acierto nulo: $\ell_{00} = \ell_{11} = 0$

$$\pi^*(\boldsymbol{x}) = \begin{cases} 0 & \text{si } \ell_{10}\, p_1 < \ell_{01}\, p_0 \\ 1 & \text{si no} \end{cases} = \begin{cases} 0 & \text{si } p_1 < \frac{\ell_{01}}{\ell_{01} + \ell_{10}} \\ 1 & \text{si no} \end{cases}$$

. . . y falso negativo c veces más caro que falso positivo: $\ell_{10} = c\ell_{01}$

$$\pi^*(\boldsymbol{x}) = \begin{cases} 0 & \text{si } p_1 < \frac{1}{1+c} \\ 1 & \text{si no} \end{cases}$$

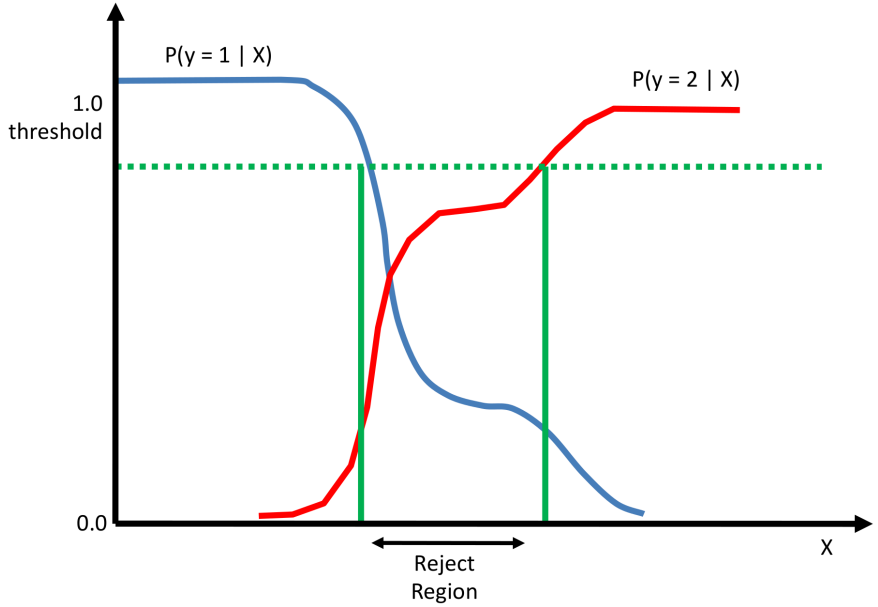
4 Clasificación con opción de rechazo

Pérdida con opción de rechazo (acción 0): $\mathcal{H} = \mathcal{Y} = \{1, \ldots, C\}$ y $\mathcal{A} = \mathcal{Y} \cup \{0\}$

$$\ell(y^*, a) = \begin{cases} 0 & \text{si } a \in \mathcal{Y} \text{ y } a = y^* \\ \lambda_r & \text{si } a = 0 \\ \lambda_e & \text{otro caso} \end{cases}$$

Estimador de Bayes: sean $y^* = \operatorname{argmax}_{y \in \mathcal{Y}} p(y \mid \boldsymbol{x})$, $p^* = p(y^* \mid \boldsymbol{x})$ y $\lambda^* = 1 - \dfrac{\lambda_r}{\lambda_e}$

$$\pi^*(\boldsymbol{x}) = \begin{cases} y^* & \text{si } p^* > \lambda^* \\ 0 & \text{en otro caso} \end{cases}$$



5 Matrices de confusión binarias

Clasificación binaria: caso particular de gran interés con una clase positiva (1) y otra negativa (-1 o 0)

Caso típico: la positiva es a priori (mucho) más improbable que la negativa y, sobre todo, queremos acertar con las muestras positivas; por ejemplo, para confirmar un diagnóstico médico hipotético a partir del resultado de una prueba

Clasificador basado en un umbral $\tau \in [0, 1]$: τ puede verse como un **margen** de positividad

$$\hat{y}_\tau(\boldsymbol{x}) = \mathbb{I}(p(y = 1 \mid \boldsymbol{x}) \geq 1 - \tau)$$

Elección de τ: no tomamos $\tau = 0.5$ y ya está, sino que variamos τ de 0 a 1 y estudiamos $\hat{y}_\tau(\boldsymbol{x})$

- **Caso $\tau = 0$:** solo acertamos positivos "seguros", esto es, tal que $p(y = 1 \mid \boldsymbol{x}) = 1$
- **Caso $\tau = 1$:** acertamos todos los positivos a cambio de clasificar también como positivos todos los negativos
- **Caso $0 < \tau < 1$:** al aumentar τ acertamos más positivos y confundimos más negativos pero, ¿a qué ritmo?

Matriz de confusión binaria: tras clasificar M muestras (de test) con un τ dado

y	$\hat{0}$	$\hat{1}$	Suma fila
0	TN_τ (true negatives)	FP_τ (false positives)	N
1	FN_τ (false negatives)	TP_τ (true positives)	P
Suma	$\hat{N}_\tau$	$\hat{P}_\tau$	M

Normalización por filas: estimación empírica de $p(\hat{y} \mid y)$

y	$\hat{0}$	$\hat{1}$	Suma fila
0	TNR_τ	FPR_τ	1.0
1	FNR_τ	TPR_τ	1.0

- $\mathrm{TNR}_\tau = \mathrm{TN}_\tau / N$ es el **true negative (rate)** o **specificity**
- $\mathrm{FPR}_\tau = \mathrm{FP}_\tau / N$ es el **false positive (rate), false alarm, type I error** o **fallout**
- $\mathrm{FNR}_\tau = \mathrm{FN}_\tau / P$ es el **false negative (rate), miss** o **type II error**
- $\mathrm{TPR}_\tau = \mathrm{TP}_\tau / P$ es el **true positive (rate), hit, recall** o **sensitivity**

Normalización por columnas: estimación empírica de $p(y \mid \hat{y})$

y	$\hat{0}$	$\hat{1}$
0	NPV_τ	FDR_τ
1	FOR_τ	PPV_τ
Suma	1.0	1.0

- $\mathrm{NPV}_\tau = \mathrm{TN}_\tau / \hat{N}_\tau$ es el **negative predictive value**
- $\mathrm{FOR}_\tau = \mathrm{FN}_\tau / \hat{N}_\tau$ es el **false omission rate**
- $\mathrm{FDR}_\tau = \mathrm{FP}_\tau / \hat{P}_\tau$ es el **false discovery rate**
- $\mathrm{PPV}_\tau = \mathrm{TP}_\tau / \hat{P}_\tau$ es el **positive predictive value** o **precision**

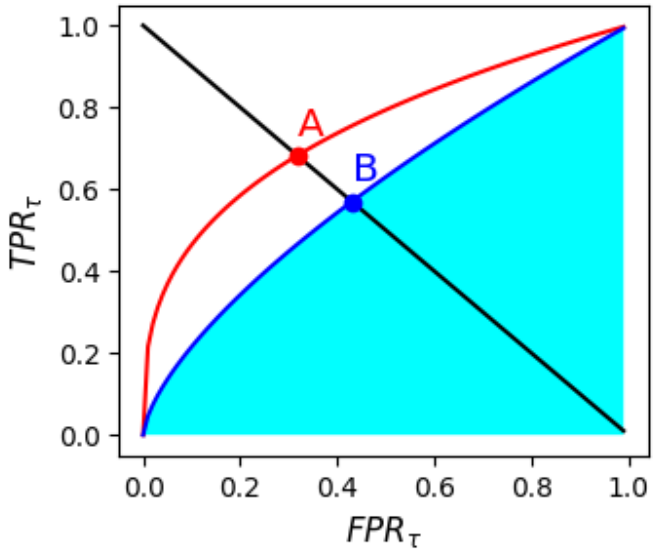
6 Curvas ROC

Propósito: estudiar el comportamiento de un clasificador binario $\hat{y}_\tau(\boldsymbol{x}) = \mathbb{I}(p(y = 1 \mid \boldsymbol{x}) \geq 1 - \tau)$ en función de τ

Curva Receiver Operating Characteristic (ROC): TPR_τ en función de la FPR_τ, para todo τ de 0 a 1

Curvas ROC típicas: de dos clasificadores, A y B, siendo el A claramente mejor que el B

In [ ]:
```
import numpy as np; import matplotlib.pyplot as plt
fA = np.vectorize(lambda x: x**(1.0/3)); fB = np.vectorize(lambda x: x**(2.0/3)); x = np.arange(0, 1, 0.01)
plt.figure(figsize=(3.5,3)); plt.plot(x, 1-x, 'k-')
plt.plot(x, fA(x), 'r-'); plt.plot(x, fB(x), 'b-'); plt.fill_between(x, fB(x), 0, facecolor='cyan')
inter_a = 0.3177; # found using scipy.optimize.fsolve(x**(1.0/3)+x-1, 0)
inter_b = 0.4302; # found using scipy.optimize.fsolve(x**(2.0/3)+x-1, 0)
plt.plot(inter_a, fA(inter_a), 'ro'); plt.plot(inter_b, fB(inter_b), 'bo')
plt.text(inter_a, fA(inter_a) + 0.05, 'A', color='red', size='x-large')
plt.text(inter_b, fB(inter_b) + 0.05, 'B', color='blue', size='x-large')
plt.xlabel(r'$FPR_{\tau}$', size='large'); plt.ylabel(r'$TPR_{\tau}$', size='large');
```



Curva ROC de un clasificador perfecto: si $p(y = 1 \mid \boldsymbol{x}) = 1$ para las muestras positivas y 0 para las negativas, la curva ROC se reduce a un punto en la esquina superior izquierda y otro en la superior derecha

- $\tau < 1$: $\quad \mathrm{FPR}_\tau = \dfrac{\mathrm{FP}_\tau}{N} = \dfrac{0}{N} = 0 \quad$ y $\quad \mathrm{TPR}_\tau = \dfrac{\mathrm{TP}_\tau}{P} = \dfrac{P}{P} = 1$
- $\tau = 1$: $\quad \mathrm{FPR}_1 = \dfrac{\mathrm{FP}_1}{N} = \dfrac{N}{N} = 1 \quad$ y $\quad \mathrm{TPR}_1 = \dfrac{\mathrm{TP}_1}{P} = \dfrac{P}{P} = 1$

Curva ROC de un clasificador aleatorio: si $p(y = 1 \mid \boldsymbol{x}) = \mathrm{Unif}(0, 1)$, la curva ROC es una recta diagonal

$$\mathrm{FPR}_\tau = \frac{\mathrm{FP}_\tau}{N} \approx \frac{N \cdot \tau}{N} = \tau \quad \text{y} \quad \mathrm{TPR}_\tau = \frac{\mathrm{TP}_\tau}{P} \approx \frac{P \cdot \tau}{P} = \tau$$

Curva ROC de un clasificador realista: monótona creciente por encima de la diagonal (mejor cuanto más arriba)

- $\tau = 0$: la curva ROC empieza en la esquina inferior izquierda o un poco más arriba; $\mathrm{FPR}_0 = 0$ y $\mathrm{TPR}_0 \geq 0$
- $\tau = 1$: la curva ROC termina en la esquina superior derecha; $\mathrm{FPR}_1 = \mathrm{TPR}_1 = 1$
- $0 < \tau < 1$: la curva ROC es monótona creciente (y por encima de la diagonal)

$$\tau' \leq \tau \quad \rightarrow \quad \mathrm{FPR}_{\tau'} \leq \mathrm{FPR}_\tau \quad \text{y} \quad \mathrm{TPR}_{\tau'} \leq \mathrm{TPR}_{\tau'}$$

Resumen de un curva ROC mediante un escalar: para facilitar su comparación con otras curvas

- **Área bajo la curva (AUC, area under curve):** en la figura ejemplo, la AUC de A es mejor (mayor) que la de B (en azul)
- **Equal error rate (EER) o cross-over rate:** valor de FPR_τ tal que $\mathrm{FPR}_\tau = \mathrm{FNR}_\tau = 1 - \mathrm{TPR}_\tau$, esto es, abscisa del punto en el que la curva ROC cruza la diagonal que va de la esquina superior izquierda a la inferior derecha; en la figura ejemplo, la EER de A (rojo) es mejor (menor) que el de B (azul)

7 Curvas PR

Propósito: estudiar $\hat{y}_\tau(\boldsymbol{x}) = \mathbb{I}(p(y=1 \mid \boldsymbol{x}) \geq 1-\tau)$ en función de τ y con especial atención a positivos ya que la noción de "negativo" está muy abierta (p. ej. en detección de objetos, recuperación de información o clasificación abierta)

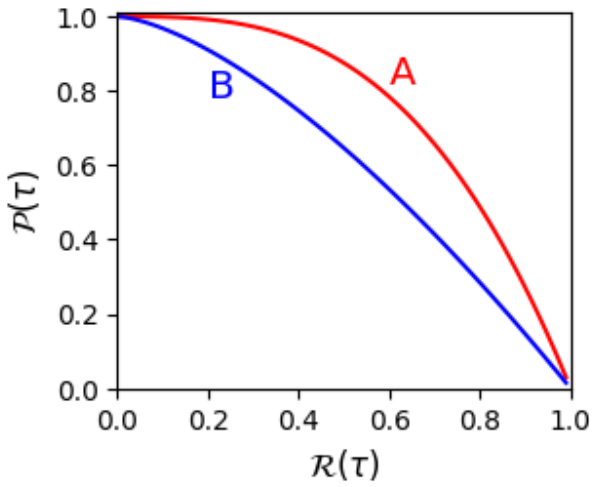
Precisión y recall (cobertura): positivos predicho-verdaderos sobre predichos y verdaderos

$$\mathcal{P}(\tau) = \frac{\mathrm{TP}_\tau}{\mathrm{TP}_\tau + \mathrm{FP}_\tau} = \frac{\mathrm{TP}_\tau}{\hat{P}_\tau} = \mathrm{PPV}_\tau \quad \text{y} \quad \mathcal{R}(\tau) = \frac{\mathrm{TP}_\tau}{\mathrm{TP}_\tau + \mathrm{FN}_\tau} = \frac{\mathrm{TP}_\tau}{P} = \mathrm{TPR}_\tau$$

Curva precision-recall (PR): $\mathcal{P}(\tau)$ en función de $\mathcal{R}(\tau)$, para todo τ de 0 a 1

Curvas PR típicas: de dos clasificadores, A y B, siendo el A claramente mejor que el B

In [ ]:
```
import numpy as np; import matplotlib.pyplot as plt
fA = np.vectorize(lambda x: 1 - x**3); fB = np.vectorize(lambda x: 1 - x**(3/2)); x = np.arange(0, 1, 0.01)
plt.figure(figsize=(3,2.5)); plt.axis([0, 1, 0, 1.01]); plt.plot(x, fA(x), 'r-'); plt.plot(x, fB(x), 'b-')
plt.text(0.6, 0.82, 'A', color='red', size='x-large'); plt.text(0.2, 0.78, 'B', color='blue', size='x-large')
plt.xlabel(r'$\mathcal{R}(\tau)$', size='large'); plt.ylabel(r'$\mathcal{P}(\tau)$', size='large');
```



Curva PR de un clasificador perfecto: si $p(y=1 \mid \boldsymbol{x}) = 1$ para las muestras positivas y 0 para las negativas, la curva PR se reduce a un punto en $(1,1)$ y otro en $(1, P/M)$

- $\tau < 1$: $\mathcal{P}(\tau) = \dfrac{\mathrm{TP}_\tau}{\hat{P}_\tau} = \dfrac{P}{P} = 1$ y $\mathcal{R}(\tau) = \dfrac{\mathrm{TP}_\tau}{P} = \dfrac{P}{P} = 1$
- $\tau = 1$: $\mathcal{P}(\tau) = \dfrac{\mathrm{TP}_\tau}{\hat{P}_\tau} = \dfrac{P}{M}$ y $\mathcal{R}(\tau) = \dfrac{\mathrm{TP}_\tau}{P} = \dfrac{P}{P} = 1$

Curva PR de un clasificador aleatorio: si $p(y=1 \mid \boldsymbol{x}) = \mathrm{Unif}(0,1)$, es una recta horizontal a altura P/M

$$\mathcal{P}(\tau) = \frac{\mathrm{TP}_\tau}{\hat{P}_\tau} \approx \frac{P \cdot \tau}{M \cdot \tau} = \frac{P}{M} \quad \text{y} \quad \mathcal{R}(\tau) = \frac{\mathrm{TP}_\tau}{P} \approx \frac{P \cdot \tau}{P} = \tau$$

Curva PR de un clasificador realista: aprox. monótona decreciente por encima de P/M (mejor cuanto más arriba)

- $\tau = 0$: asumiendo que se predice algún positivo, es de esperar una precisión alta y cobertura baja (próxima a 0)
- $\tau = 1$: punto $(1, P/M)$
- $0 < \tau < 1$: curva **aproximadamente** monótona decreciente

$$\tau' \leq \tau \;\rightarrow\; \mathcal{R}(\tau') \leq \mathcal{R}(\tau) \quad \text{pero} \quad \tau' \leq \tau \;\not\rightarrow\; \mathcal{P}(\tau') \geq \mathcal{P}(\tau)$$

Curva PR interpolada: sustituye $\mathcal{P}(\tau)$ por $\max_{\tilde{\tau} \geq \tau} \mathcal{P}(\tilde{\tau})$ para suavizar la curva haciéndola monótona decreciente

Resumen de un curva PR mediante un escalar: para facilitar su comparación con otras curvas

- **Área bajo la curva PR o average precision (AP):** en la figura ejemplo, la AP de A es mejor (mayor) que la de B
- **Mean average precision (mAP):** AP media de varias curvas PR; p. ej. de cada clase en clasificación multiclase
- **mAP@_:** mAP limitada, p. ej. a las K muestras más probables o un cierto subconjunto de umbrales

8 F-scores

Propósito: establecer una única medida que evalúe adecuadamente la calidad de cualquier par precisión-cobertura

F-scores: F_β con $\beta \geq 0$ da β veces más importancia a la cobertura que a la precisión

$$F_\beta = (1+\beta^2)\,\frac{\mathcal{P}\,\mathcal{R}}{\beta^2\mathcal{P}+\mathcal{R}} \qquad \text{o} \qquad \frac{1}{F_\beta} = \frac{1}{1+\beta^2}\,\frac{1}{\mathcal{P}} + \frac{\beta^2}{1+\beta^2}\,\frac{1}{\mathcal{R}}$$

Casos particulares en función de la β:

- $\beta = 0$: da nula importancia a la cobertura frente a la precisión; de hecho, $F_0 = \mathcal{P}$
- $\beta = 0.5$: da la mitad de importancia a la cobertura que a la precisión
- $\beta = 1$: da la misma importancia a la cobertura que a la precisión y coincide con la **media armónica** de ambas

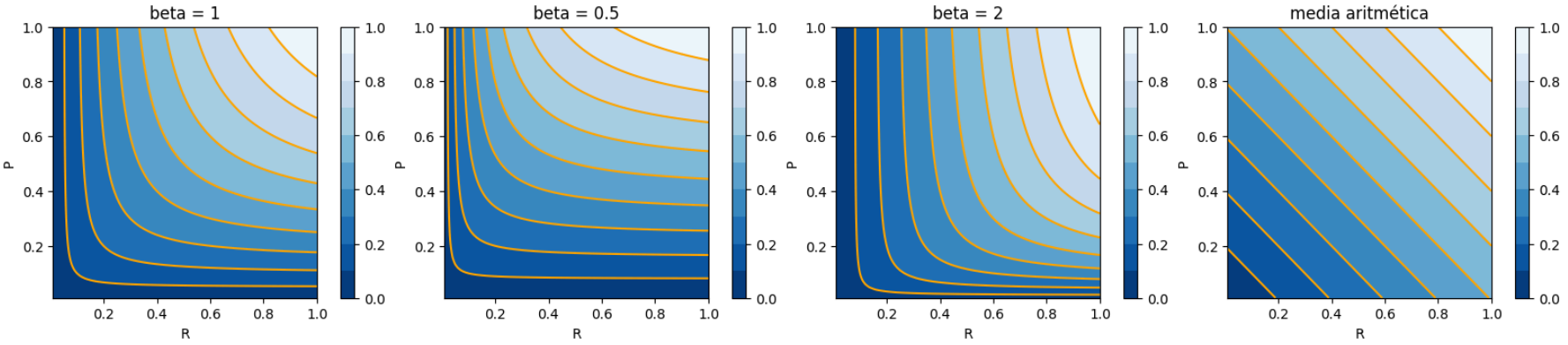
$$F_1 = 2\,\frac{\mathcal{P}\,\mathcal{R}}{\mathcal{P}+\mathcal{R}} = \frac{2}{\frac{1}{\mathcal{R}}+\frac{1}{\mathcal{P}}}$$

- $\beta = 2$: da el doble de importancia a la cobertura que a la precisión
- $\beta \to \infty$: da infinita importancia a la cobertura frente a la precisión

$$\lim_{\beta\to\infty} F_\beta = \lim_{\beta\to\infty} \frac{\mathcal{P}\,\mathcal{R}}{\beta^2\mathcal{P}+\mathcal{R}} + \lim_{\beta\to\infty} \frac{\mathcal{P}\,\mathcal{R}}{\mathcal{P}+\frac{\mathcal{R}}{\beta^2}} = \mathcal{R}$$

Comparación gráfica: F_β en función de $\mathcal{P}$ y $\mathcal{R}$, para $\beta \in (1, 0.5, 2)$; también media aritmética

In [ ]:
```
import numpy as np; import matplotlib.pyplot as plt
R, P = np.meshgrid(np.linspace(0.01, 1, 100), np.linspace(0.01, 1, 100)); RP = np.c_[np.ravel(R), np.ravel(P)]
B = (1, 0.5, 2); nrows = 1; ncols = len(B) + 1 # len(B) plots de F + plot de media
_, axs = plt.subplots(nrows=nrows, ncols=ncols, figsize=(16, 14*nrows/ncols), constrained_layout=True)
for ax, b in zip(axs.flat[:-1], B):
    bsquared = np.square(b)
    F = lambda rp: (1 + bsquared) * rp.prod() / ( bsquared * rp[1] + rp[0] ); FF = np.apply_along_axis(F, 1, RP)
    ax.set_title(f"beta = {b}"); ax.set_xlabel('R'); ax.set_ylabel('P')
    ax.contour(R, P, FF.reshape(R.shape), 10, colors='orange', linestyles='solid')
    cp = ax.contourf(R, P, FF.reshape(R.shape), 10, cmap='Blues_r'); plt.colorbar(cp, ax=ax)
ax = axs.flat[-1]; A = RP.mean(axis = 1); ax.set_title(f"media aritmética"); ax.set_xlabel('R'); ax.set_ylabel('P')
ax.contour(R, P, A.reshape(R.shape), 10, colors='orange', linestyles='solid')
cp = ax.contourf(R, P, A.reshape(R.shape), 10, cmap='Blues_r'); plt.colorbar(cp, ax=ax);
```



Observaciones que se derivan de la comparación gráfica:

- F_1: sus curvas de nivel son simétricas respecto a la diagonal, por lo que precisión y cobertura contribuyen con la misma importancia, si bien ambas deben ser elevadas para que F_1 también lo sea
- F_β **con** $\beta < 1$ **(p. ej.** $\beta = 0.5$**):** las curvas de nivel de F_β tienden a horizontalizarse, esto es, F_β se aproxima a $\mathcal{P}$
- F_β **con** $\beta > 1$ **(p. ej.** $\beta = 2$**):** las curvas de nivel F_β tienden a verticalizarse, por lo que F_β se aproxima a $\mathcal{R}$
- **Media aritmética:** parecida a F_1, pero crece mucho con precisiones cuasi-nulas (hasta 0.5 con $\mathcal{P} = 0$)

F-scores con $C > 2$ clases: consideramos C casos binarios, uno por cada clase frente al resto

Macro average F1: media de los scores F1 de las clases

$$\mathrm{MacroF1} = \frac{1}{C} \sum_c F_1^{(c)}$$

Weighted average F1: media de los scores F1 de las clases, ponderados por las probabilidades a priori de clase

$$\mathrm{WeightedF1} = \sum_c \hat{p}(c) F_1^{(c)}$$

Micro average F1: F_1 global, calculado a partir de precisión y cobertura globales o, más directamente, a partir verdaderos-positivos, falsos-negativos y falsos-positivos globales

$$\mathrm{MicroF1} = 2\,\frac{\mathcal{P}\,\mathcal{R}}{\mathcal{P}+\mathcal{R}} = 2\,\frac{\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \cdot \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}}{\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} + \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}} = \frac{\mathrm{TP}}{\mathrm{TP} + \frac{1}{2}(\mathrm{FN}+\mathrm{FP})}$$

Ejercicio: calcula F1 macro, weighted y micro a partir de la siguiente matriz de confusiones

y	$\hat{1}$	$\hat{2}$	$\hat{3}$
1	2	1	0
2	0	1	0
3	1	2	3

Solución:

y	$\mathrm{TP}^{(y)}$	$\mathrm{FP}^{(y)}$	$\mathrm{FN}^{(y)}$	$\mathcal{P}^{(y)}$	$\mathcal{R}^{(y)}$	$F_1^{(y)}$	$\hat{p}(y)$
1	2	1	1	$2/3$	$2/3$	$2/3$	$3/10$
2	1	3	0	$1/4$	$1/1$	$2/5$	$1/10$
3	3	0	3	$3/3$	$3/6$	$2/3$	$6/10$

$$\mathrm{MacroF1} = \frac{1}{3}\left(\frac{2}{3} + \frac{2}{5} + \frac{2}{3}\right) = \frac{26}{45} = 0.58$$

$$\mathrm{WeightedF1} = \frac{3}{10}\cdot\frac{2}{3} + \frac{1}{10}\cdot\frac{2}{5} + \frac{6}{10}\cdot\frac{2}{3} = \frac{48}{75} = 0.64$$

$$\mathrm{MicroF1} = \frac{\mathrm{TP}}{\mathrm{TP} + \frac{1}{2}(\mathrm{FN}+\mathrm{FP})} = \frac{6}{6 + \frac{1}{2}(4+4)} = \frac{3}{5} = 0.60$$