UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escola Tècnica Superior d'Enginyeria Informàtica

etsinf

Tema 9. Bagging y Boosting

Percepción (PER)

Curso 2017/2018

Departamento de Sistemas Informáticos y Computación

Índice

Índice

Introducción

- Las fuentes de error de un clasificador son:
  - ***Bias*** (sesgo): asunciones erróneas, error en la selección del tipo de clasificador. Relacionado con la capacidad de ajuste del clasificador elegido a los datos.
  - ***Variance*** (varianza): dependencia de los datos de entrenamiento. Relacionado con la bondad del aprendizaje del clasificador en función de la cantidad de datos disponibles.
  - ***Noise*** (ruido): ruido inherente en los datos
- Compromiso entre *bias* y *variance* para el diseño de un buen clasificador
- Caracterización de *bias* y *variance* de los distintos clasificadores

Caracterización del error

Clasificador como regresor (aprendido a partir de datos de entrenamiento)

$$G(x) : E \to \mathbb{R}$$

y valor verdadero

$$y = F(x) + \epsilon$$

- $F(x)$: función verdadera
- ϵ: ruido inherente de los datos

Representación del error como el ***valor esperado del error cuadrático***:

$$\mathbb{E}[(y - G(x))^2]$$

Caracterización del error

Desarrollo del valor esperado del error:

$$\mathbb{E}\left[(y - G(x))^2\right] = \mathbb{E}\left[y^2 - 2\,y\,G(x) + G(x)^2\right] = \begin{array}{l} \mathbb{E}\left[G(x)^2\right] \\ -2\ \mathbb{E}[G(x)]\ \mathbb{E}[y] \\ +\mathbb{E}\left[y^2\right] \end{array}$$

Definiendo $\overline{Z} = \mathbb{E}[Z]$, al ser $\mathbb{E}[Z^2] = \mathbb{E}\left[\left(Z - \overline{Z}\right)^2\right] + \overline{Z}^2$:

$$\mathbb{E}\left[(y - G(x))^2\right] = \begin{array}{l} \mathbb{E}\left[\left(G(x) - \overline{G(x)}\right)^2\right] + \overline{G(x)}^2 \\ -2\,\overline{G(x)}\,\overline{y} \\ +\mathbb{E}\left[(y - \overline{y})^2\right] + \overline{y}^2 \end{array}$$

Caracterización del error

Al ser $\overline{y} = \mathbb{E}\left[F(x) + \epsilon\right] = F(x)$ (por cancelación del error):

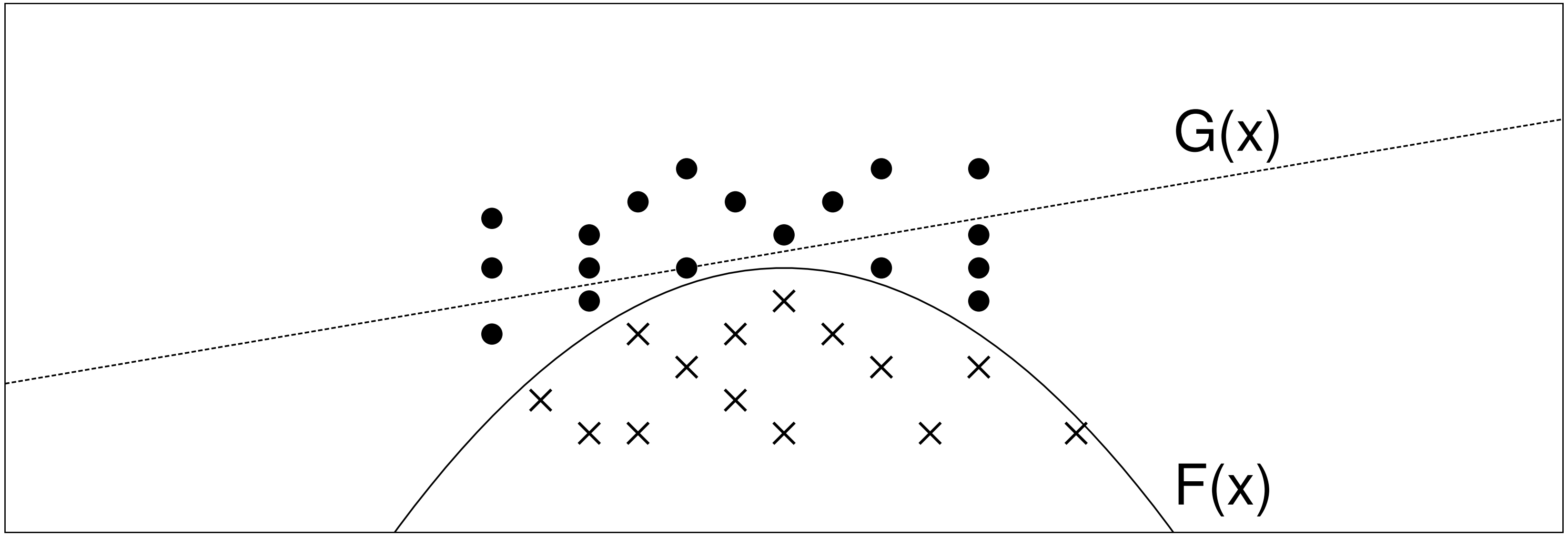
$$\begin{aligned}\mathbb{E}\left[(y - G(x))^2\right] = \; & \mathbb{E}\left[\left(G(x) - \overline{G(x)}\right)^2\right] + \overline{G(x)}^2 \\ & -2\,\overline{G(x)}\,F(x) \\ & +\mathbb{E}\left[(y - F(x))^2\right] + F(x)^2\end{aligned}$$

Como $\overline{G(x)}^2 - 2\,\overline{G(x)}\,F(x) + F(x)^2 = \left(\overline{G(x)} - F(x)\right)^2$:

$$\begin{aligned}\mathbb{E}\left[(y - G(x))^2\right] = \; & \mathbb{E}\left[\left(G(x) - \overline{G(x)}\right)^2\right] && \text{Variance} \\ & +\left(\overline{G(x)} - F(x)\right)^2 && \text{Bias} \\ & +\mathbb{E}\left[(y - F(x))^2\right] && \text{Noise}\end{aligned}$$

Caracterización del error

- ***Variance***: variación de $G(x)$ según datos de entrenamiento
- ***Bias***: error del clasificador promedio, capacidad de adaptarse al entrenamiento
- ***Noise***: ruido presente en los datos



Tipos de clasificadores

- Clasificadores con *bias* alto y *variance* bajo: (p.ej., clasificador lineal)
  - Poco flexibles
  - Pocos parámetros
  - Bajo requerimiento de datos de entrenamiento
  - Clasificadores débiles (*weak learners*): apenas mejores que el clasificador aleatorio
- Clasificadores con *bias* bajo y *variance* alto: (p.ej., k-NN)
  - Muy flexibles (aprenden cualquier frontera de decisión)
  - Muchos parámetros
  - Alto requerimiento de datos de entrenamiento
  - Clasificadores fuertes (*strong learners*): *arbitrariamente* precisos
- ***Ensemble learning***: combinación de clasificadores
  - ***Bagging***:
    combinación de clasificadores fuertes modificando el conjunto de entrenamiento
  - ***Boosting***:
    construcción de clasificadores fuertes a partir de clasificadores débiles

Índice

Bagging

Bagging: **B**ootstrap **Ag**regat**ing**

Clasificadores G_i a partir de variación de los datos de entrenamiento X

- Obtener X_i por *bootstrapping* desde X
- *Bootstrapping*: muestreo aleatorio con reemplazamiento
- Entrenar G_i con X_i

Combinación de clasificadores G_i por suma no ponderada

Bagging

Algoritmo Bagging:

- Entrenamiento:

  For $i = 1 \cdots M$

  Obtener X_i a partir de X

  Entrenar G_i con X_i

  End

- Clasificación:

$$G(x) = \frac{1}{M} \sum_{i=1}^{M} G_i(x)$$

Bagging se emplea en clasificadores binarios, con $\hat{c}(x) = \text{sgn}(G(x))$

Propiedades de Bagging

- ***Variance***:

  $$\mathbb{E}\left[\left(G(x) - \overline{G(x)}\right)^2\right] \qquad G(x) = \frac{1}{M}\sum_{i=1}^{M} G_i(x)\text{, el variance se reduce}$$

- ***Bias***:

  $$\left(\overline{G(x)} - F(x)\right)^2 \qquad \overline{G(x)}\text{ no cambia, y el bias no cambia}$$

- El error del clasificador generado mediante Bagging se reduce

- Bagging es adecuado para combinar clasificadores fuertes (flexibles, *bias* bajo)

Índice

Boosting

- Combinación de clasificadores débiles ponderando los datos de entrenamiento
- Se dispone de un conjunto de L clasificadores débiles: $\mathcal{G} = \{G_1, \ldots, G_L\}$
- Se asumen clasificadores débiles binarios: $G_l(x) \in \{-1, 1\}$
- Conjunto de entrenamiento: $\mathcal{X} = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ con $y_n \in \{-1, 1\}$
- En cada iteración, selecciona $C_i \in \mathcal{G}$ de menor error sobre $\mathcal{X}$ ponderado por $w^{(i)}$
- $G(x)$ es la combinación lineal de los clasificadores seleccionados hasta iteración m:

$$G(x) = G^{(m)}(x) = \sum_{i=1}^{m} \alpha_i \, C_i(x) \ \text{ donde } \ C_i \in \mathcal{G}$$

Boosting

En la iteración m seleccionamos un clasificador C_m junto con su peso α_m

$$G^{(m)}(x) = G^{(m-1)}(x) + \alpha_m \, C_m(x)$$

El criterio de error E a minimizar es la pérdida exponencial en cada dato

$$E = \sum_{i=1}^{N} \exp(-y_i \, G^{(m)}(x_i)) = \sum_{i=1}^{N} \exp(-y_i \, G^{(m-1)}(x_i) - y_i \, \alpha_m \, C_m(x_i))$$

Se buscan C_m y α_m que minimicen E

Boosting

El peso del dato x_i para la iteración m es la pérdida exponencial en ese dato:

$$w_i^{(m)} = \exp(-y_i \, G^{(m-1)}(x_i))$$

Luego:

$$E = \sum_{i=1}^{N} w_i^{(m)} \, \exp(-y_i \, \alpha_m \, C_m(x_i))$$

Separando en muestras bien $(y_i \cdot C_m(x_i) = 1)$ y mal $(y_i \cdot C_m(x_i) = -1)$ clasificadas:

$$\begin{aligned} E &= \sum_{y_i = C_m(x_i)} w_i^{(m)} \, \exp(-\alpha_m) + \sum_{y_i \neq C_m(x_i)} w_i^{(m)} \, \exp(\alpha_m) \\ &= \sum_{i=1}^{N} w_i^{(m)} \, \exp(-\alpha_m) + \sum_{y_i \neq C_m(x_i)} w_i^{(m)} \, (\exp(\alpha_m) - \exp(-\alpha_m)) \end{aligned}$$

Boosting

Tenemos:

$$E = \sum_{i=1}^{N} w_i^{(m)} \exp(-\alpha_m) + \sum_{y_i \neq C_m(x_i)} w_i^{(m)} \left(\exp(\alpha_m) - \exp(-\alpha_m)\right)$$

Minimizamos E respecto a C_m:

- Primer sumatorio independiente de C_m
- Asumimos $(\exp(\alpha_m) - \exp(-\alpha_m))$ constante

$$E \approx \sum_{y_i \neq C_m(x_i)} w_i^{(m)} = \sum_{y_i \neq C_m(x_i)} \exp(-y_i \, G^{(m-1)}(x_i))$$

Para minimizar E selecciona el clasificador $C_m \in \mathcal{G}$ que minimice el error de clasificación $(y_i \neq C_m(x_i))$ sobre los datos ponderados

Boosting

Para calcular α_m, derivarmos E respecto de α_m e igualar a cero:

$$\frac{dE}{d\alpha_m} = -\sum_{y_i = C_m(x_i)} w_i^{(m)} \exp(-\alpha_m) + \sum_{y_i \neq C_m(x_i)} w_i^{(m)} \exp(\alpha_m) = 0$$

$$\exp(-\alpha_m) \sum_{y_i = C_m(x_i)} w_i^{(m)} = \exp(\alpha_m) \sum_{y_i \neq C_m(x_i)} w_i^{(m)} \rightarrow \frac{\sum_{y_i = C_m(x_i)} w_i^{(m)}}{\sum_{y_i \neq C_m(x_i)} w_i^{(m)}} = \exp(2\alpha_m) \rightarrow$$

$$\alpha_m = \frac{1}{2} \ln \left(\frac{\sum_{y_i = C_m(x_i)} w_i^{(m)}}{\sum_{y_i \neq C_m(x_i)} w_i^{(m)}} \right)$$

Se define $\epsilon_m = \frac{\sum_{y_i \neq C_m(x_i)} w_i^{(m)}}{\sum_{i=1}^{N} w_i^{(m)}}$ (error en iteración m): $\alpha_m = \frac{1}{2} \ln \left(\frac{1-\epsilon_m}{\epsilon_m} \right)$

Algoritmo AdaBoost

Entrada:

- Conjunto de entrenamiento $\mathcal{X} = \{(x_1, y_1) \dots (x_N, y_N)\}$
- Conjunto clasificadores débiles (binarios) $\mathcal{G} = \{G_1, \dots, G_L\}$

Proceso:

1. $w_i^{(1)} = \frac{1}{N} \quad i = 1, \dots, N$

2. Para $m = 1 \dots M$

2.1. $C_m = \text{argmin}_{g \in \mathcal{G}} \sum_{y_i \neq g(x_i)} w_i^{(m)}$

2.2. $\epsilon_m = \min_{g \in \mathcal{G}} \sum_{y_i \neq g(x_i)} w_i^{(m)}$

2.3. Si $\epsilon_m > 0.5$ fin

2.3. $\alpha_m = \frac{1}{2} \ln \left(\frac{1-\epsilon_m}{\epsilon_m} \right)$

2.4. $w_i^{(m+1)} = \frac{w_i^{(m)} \exp(-y_i \, \alpha_m \, C_m(x_i))}{\sum_{i'=1}^{N} w_{i'}^{(m)} \exp(-y_{i'} \, \alpha_m \, C_m(x_{i'}))}$

Salida: $G(x) = \sum_{m=1}^{M} \alpha_m \, C_m(x)$

Propiedades de AdaBoost

Boosting:

- Aprovecha el bajo *variance* de los clasificadores (débiles) combinados
- Reduce el *bias*
- Es más sensible a datos ruidosos
- En comparación con Bagging, puede comportarse peor según los datos