Aprendizaje Semi Supervisado

Minería de Datos (CC3074) - 2026

Aprendizaje Semi Supervisado

Semestre 01, 2026

El problema del etiquetado

Las etiquetas son costosas.

Etiquetar datos requiere tiempo, dinero y conocimiento experto.

La brecha en la práctica

| Tipo de dato | Disponibilidad | | --- | --- | | Datos sin etiquetar | Abundante y barato | | Datos etiquetados | Escaso y costoso |

Ejemplos: imágenes médicas, texto web, sensores industriales.

El dilema

Aprendizaje supervisado requiere muchas etiquetas.

Aprendizaje no supervisado ignora las etiquetas que sí existen.

El aprendizaje semi supervisado aprovecha ambos.

Definición

El aprendizaje semi supervisado combina una pequeña cantidad de datos etiquetados con una gran cantidad de datos no etiquetados durante el entrenamiento.

$D = D_L \cup D_U$

Donde $|D_L| \ll |D_U|$.

Supuestos fundamentales

Todo método semi supervisado se apoya en al menos un supuesto sobre los datos.

Supuesto de suavidad

Si dos puntos están cerca en el espacio de características, probablemente tienen la misma etiqueta.

Un modelo suave entre puntos vecinos generaliza mejor.

Supuesto de clúster

Los datos forman grupos naturales.

Los puntos dentro del mismo clúster comparten etiqueta.

La frontera de decisión no atraviesa regiones densas.

Supuesto de variedad

Los datos de alta dimensión se encuentran sobre una variedad de dimensión menor.

La estructura real del espacio es más simple que el espacio original.

Las etiquetas varían suavemente sobre esa variedad.

Self-Training

Entrena un modelo inicial con los datos etiquetados.

Predice etiquetas para los datos no etiquetados.

Agrega las predicciones más confiables al conjunto de entrenamiento y repite.

Algoritmo

```
1. Entrenar clasificador con D_L
2. Para cada iteración:
   a. Predecir etiquetas y probabilidades sobre D_U
   b. Seleccionar predicciones con confianza > umbral
   c. Moverlas de D_U a D_L
   d. Re-entrenar el clasificador
3. Repetir hasta convergencia o max_iter
```

Parámetros clave

| Parámetro | Efecto | | --- | --- | | `threshold` | Mínima probabilidad para aceptar una pseudo-etiqueta | | `k_best` | Alternativa: aceptar los K más confiables por iteración | | `max_iter` | Número máximo de iteraciones |

Condición de uso

El clasificador base debe producir probabilidades calibradas.

Un clasificador mal calibrado generará pseudo-etiquetas erróneas que se propagan y amplifican.

Ventajas y limitaciones

| Ventajas | Limitaciones | | --- | --- | | Simple y genérico | Error se amplifica con iteraciones | | Compatible con cualquier clasificador | Sensible a la calibración | | No requiere arquitectura especial | Puede divergir con umbral mal ajustado |

Co-Training

Divide las variables en dos vistas independientes.

Entrena un clasificador diferente en cada vista.

Cada modelo etiqueta los datos más confiables para el otro.

Supuesto clave

Las dos vistas deben ser:

- **Suficientes**: cada vista sola puede aprender el concepto
- **Independientes condicionalmente**: las vistas no comparten información redundante

Ejemplo: clasificación de páginas web

| Vista 1 | Vista 2 | | --- | --- | | Texto de la página | Texto de los hiperenlaces que apuntan a ella | | Semántica del contenido | Semántica del contexto externo |

Dos clasificadores se enseñan mutuamente desde perspectivas distintas.

Proceso

```
1. Dividir variables en vista_1 y vista_2
2. Entrenar clf_1 en (D_L, vista_1) y clf_2 en (D_L, vista_2)
3. Para cada iteración:
   a. clf_1 etiqueta con alta confianza → agrega a D_L de clf_2
```

```
   b. clf_2 etiqueta con alta confianza → agrega a D_L de clf_1
   c. Re-entrenar ambos
4. Predicción final: combinar ambos modelos
```

Ventajas y limitaciones

| Ventajas | Limitaciones | | --- | --- | | Dos modelos se corrigen mutuamente | Requiere dos vistas naturales e independientes | | Más robusto que self-training | Difícil dividir las variables sin perder información | | Reduce propagación de errores | No aplicable si solo hay una vista |

Propagación de Etiquetas

Los datos se representan como un grafo de similitud.

Las etiquetas de los nodos conocidos se propagan hacia los nodos vecinos.

Construcción del grafo

Cada instancia es un nodo.

Los bordes conectan puntos similares, ponderados por similitud.

Kernels disponibles:

- **RBF**: `$K(x, x') = \exp(-\gamma ||x - x'||^2)$` — grafo completamente conectado
- **KNN**: conectar solo los K vecinos más cercanos — grafo disperso, más eficiente

Label Propagation

Las etiquetas conocidas se mantienen fijas durante la propagación.

El algoritmo itera: cada nodo adopta la distribución de etiquetas de sus vecinos, ponderada por similitud.

Repite hasta convergencia.

$F = (D - W)^{-1} Y$

Donde W es la matriz de similitud y D la matriz diagonal de grados.

Label Spreading

Similar a Label Propagation pero con regularización.

Las etiquetas conocidas pueden ajustarse levemente.

Más robusto ante ruido y etiquetas incorrectas.

$F^* = \alpha (D^{-1/2} W D^{-1/2}) F + (1 - \alpha) Y$

Comparación

| Aspecto | Label Propagation | Label Spreading | | --- | --- | --- | | Clamping | Fijo (α = 0) | Suave (0 < α < 1) | | Robustez al ruido | Menor | Mayor | | Matriz usada | Similitud directa | Laplaciano normalizado | | Parámetro α | No aplica | Controla rigidez de etiquetas originales |

Cuándo usar propagación

- Los datos tienen estructura de grafo natural
- Existen clusters bien definidos
- Los datos etiquetados y no etiquetados están mezclados en el mismo espacio

SVM Semi Supervisado

Recordatorio: SVM clásico

SVM busca el hiperplano que maximiza el margen entre las clases etiquetadas.

Solo usa `$D_L$` — ignora completamente `$D_U$`.

Transductive SVM (TSVM)

Extiende SVM para usar también los datos no etiquetados.

Busca un hiperplano que maximiza el margen y a la vez aleja los datos no etiquetados de la frontera de decisión.

Los datos no etiquetados deben caer en regiones de baja densidad.

Intuición

Si la frontera de decisión pasa por regiones densas de datos, es una mala frontera.

TSVM fuerza que la frontera atraviese regiones vacías o de baja densidad.

Esto es coherente con el supuesto de clúster.

Formulación

Maximizar el margen sobre `$D_L$` y además exigir que los puntos en `$D_U$` tengan alta confianza (lejos de la frontera).

Es un problema de optimización no convexo → se resuelve con aproximaciones iterativas.

Ventajas y limitaciones

| Ventajas | Limitaciones | | --- | --- | | Aprovecha la estructura de los datos no etiquetados | Optimización no convexa, costosa computacionalmente | | Margen amplio en zonas de baja densidad | No escala bien a datasets grandes | | Extensión natural de SVM | Difícil de implementar en la práctica |

Modelos Generativos

Los modelos generativos aprenden cómo se generan los datos.

Modelan la distribución conjunta `$P(x, y)$`.

Mezcla Gaussiana Semi Supervisada

Se asume que los datos provienen de una mezcla de distribuciones gaussianas.

Cada componente de la mezcla corresponde a una clase.

Los datos etiquetados anclan qué componente pertenece a cada clase.

Los datos no etiquetados refinan los parámetros de la distribución.

Expectation-Maximization (EM)

Paso E: asignar probabilidades de pertenencia a cada componente para los datos no etiquetados.

Paso M: re-estimar los parámetros de las gaussianas usando todos los datos.

Se alterna hasta convergencia.

Supuesto crítico

El modelo generativo debe ser correcto.

Si los datos no siguen una mezcla gaussiana, el modelo será erróneo y los datos no etiquetados perjudicarán el rendimiento.

Ventajas y limitaciones

| Ventajas | Limitaciones | | --- | --- | | Marco probabilístico completo | Sensible al modelo generativo asumido | | Incorpora datos no etiquetados de forma natural | Si el modelo es incorrecto, el rendimiento empeora | | Produce probabilidades de clase | Mayor complejidad que métodos discriminativos |

K-Means con Restricciones

Clustering estándar

K-Means agrupa datos minimizando la distancia intra-clúster.

No usa ninguna información de etiquetas.

Restricciones de pares

Se introducen restricciones sobre pares de instancias:

| Tipo | Significado | | --- | --- | | **Must-link** | Estos dos puntos deben pertenecer al mismo clúster | | **Cannot-link** | Estos dos puntos deben pertenecer a clústeres distintos |

Las restricciones provienen de etiquetas parciales: si `$x_i$` y `$x_j$` tienen la misma etiqueta → must-link; diferente etiqueta → cannot-link.

COP K-Means

Constrained Object Placement K-Means.

Variante de K-Means que respeta las restricciones al asignar cada punto al centroide más cercano.

Si la asignación viola una restricción, se intenta el siguiente centroide más cercano.

Si ninguna asignación es válida, el algoritmo falla → las restricciones son inconsistentes.

Proceso

```
1. Inicializar K centroides
2. Para cada punto x_i:
   a. Ordenar centroides por distancia a x_i
   b. Asignar al centroide más cercano que no viole restricciones
3. Recalcular centroides
4. Repetir hasta convergencia
```

Ventajas y limitaciones

| Ventajas | Limitaciones | | --- | --- | | Usa conocimiento parcial sin etiquetado completo | Puede fallar si las restricciones son inconsistentes | | Mejora calidad del clustering | Complejidad mayor que K-Means clásico | | Flexible: solo se necesitan pares | Sensible a la inicialización |

Laplacian SVM

Combina SVM con la estructura de grafo del Laplaciano.

Busca una función de clasificación que sea a la vez precisa y suave sobre el grafo de datos.

Motivación

SVM clásico: maximizar el margen usando solo `$D_L$`.

Label Propagation: propagar etiquetas por el grafo usando `$D_L \cup D_U$`.

Laplacian SVM: maximizar el margen y exigir suavidad sobre el grafo simultáneamente.

El Laplaciano del grafo

Se construye el grafo de similitud sobre todos los datos (etiquetados y no etiquetados).

El Laplaciano `$L = D - W$` captura la estructura local de los datos.

Una función suave sobre el grafo no cambia bruscamente entre puntos similares.

Formulación

Minimizar:

`$\frac{1}{l} \sum_{i=1}^{l} V(f(x_i), y_i) + \lambda_A ||f||^2 + \lambda_I f^T L f$`

- Primer término: error en los datos etiquetados
- Segundo término: regularización estándar (complejidad del modelo)
- Tercer término: suavidad sobre el grafo (penaliza cambios bruscos entre vecinos)

Parámetros

Parámetro	Efecto
`$\lambda_A$`	Regularización del espacio de funciones (como en SVM clásico)
`$\lambda_I$`	Peso del término de suavidad sobre el grafo
γ (kernel)	Define la similitud entre puntos en el grafo

Ventajas y limitaciones

| Ventajas | Limitaciones | | --- | --- | | Aprovecha la geometría intrínseca de los datos | Mayor costo computacional que SVM y TSVM | | Unifica margen y suavidad en un solo objetivo | Requiere construir y almacenar el grafo completo | | Sólido fundamento teórico | Difícil de escalar a datasets grandes |

Comparación de métodos

| Método | Supuesto principal | Complejidad | Escalabilidad | | --- | --- | --- | --- | | Self-Training | Suavidad | Baja | Alta | | Co-Training | Dos vistas independientes | Media | Alta | | Label Propagation | Suavidad + Clúster | Media | Media | | Label Spreading | Suavidad + Clúster | Media | Media | | TSVM | Clúster (baja densidad) | Alta | Baja | | Modelos Generativos | Distribución conocida | Media | Media | | K-Means Restricciones | Clúster | Media | Media | | Laplacian SVM | Variedad + Suavidad | Alta | Baja |

¿Cuándo usar aprendizaje semi supervisado?

- Hay datos etiquetados insuficientes para un buen modelo supervisado
- Etiquetar más datos es costoso o imposible
- Los datos no etiquetados son abundantes y representativos
- Los datos cumplen con al menos un supuesto de distribución

Riesgo: cuando los datos no etiquetados dañan

Si los datos no etiquetados no son representativos de la distribución real, agregar más datos puede perjudicar el modelo.

Esto se llama **degradación semi supervisada**.

Siempre validar contra una línea base supervisada.

Principios clave

- El aprendizaje semi supervisado no es gratis — requiere supuestos sobre los datos
- Self-training es simple pero propaga errores
- Los métodos basados en grafos explotan la estructura geométrica local
- TSVM y Laplacian SVM son más poderosos pero más costosos computacionalmente
- Siempre comparar contra un modelo supervisado entrenado solo con `$D_L$`
- Los datos no etiquetados mal elegidos pueden empeorar el modelo