

Actividad 3 – Machine Learning II

Profesor: Francisco Pérez Galarce

Formato de entrega. Un notebook en Python con: Código limpio, celdas ordenadas y comentarios. Gráficos y tablas de resultados.

Dentro del mismo notebook usando Markdown debe incluir: Descripción del preprocesamiento. Comparación de modelos. Discusión de resultados y conclusiones.

La actividad puede ser desarrollada en grupos de máximo 3 personas.

El avance de cada grupo será discutido en la próxima clase, sábado 10 de enero.

CONTEXTO

En las Actividades 1 y 2 se abordó el problema de churn en telecomunicaciones mediante modelos lineales (regresión logística) y modelos no lineales basados en árboles. En la presente actividad se incorporan dos enfoques clásicos y complementarios de clasificación supervisada: (i) Naïve Bayes, basado en modelación probabilística y en el Teorema de Bayes, y (ii) Support Vector Machines (SVM), fundamentado en la maximización del margen de separación entre clases.

Nota: Su grupo puede decidir trabajar con una base de datos diferente si esta ya se encuentra en condiciones de ser usada. Si la base de datos requiere mucho trabajo adicional, se sugiere usar la base de datos recomendada.

OBJETIVOS DE APRENDIZAJE

Al finalizar la actividad, la o el estudiante será capaz de:

- Implementar un clasificador Naïve Bayes coherente con el tipo de variables (numéricas/categóricas) y justificar sus supuestos.
- Ajustar y seleccionar un clasificador SVM (kernel lineal y no lineal) mediante validación cruzada y búsqueda de hiperparámetros.
- Comparar Naïve Bayes y SVM con los modelos previos (regresión logística y árboles) en términos de métricas, curvas ROC/PR y tiempos de entrenamiento.
- Analizar críticamente el rol de los supuestos, el escalamiento y la dimensionalidad en el rendimiento observado.

INSTRUCCIONES

Paso 1. Naïve Bayes

- a) Utilice el mismo dataset y el mismo preprocesamiento base definido en las actividades previas (imputación, one-hot encoding, escalamiento de variables numéricas).
- b) Selección de variante de Naïve Bayes. Entrene al menos una variante de Naïve Bayes adecuada a su representación de datos. Por ejemplo:
 - I. GaussianNB para features continuas (numéricas escaladas) y/o una matriz densa.
 - II. BernoulliNB si su matriz final es principalmente binaria (one-hot) y decide binarizar variables numéricas.
- c) Justifique la elección en función de la naturaleza de las features y del supuesto de independencia condicional.
- d) Evaluación con validación cruzada. Desde la documentación de sklearn, estudie los hiperparámetros que se deben definir en el modelo e implemente grid search cross validation para seleccionarlos. Reporte en el conjunto de testeo: Accuracy, Precision, Recall, F1, AUC-ROC y PR-AUC. Genere para el conjunto de testeo la matriz de confusión, curva ROC promedio y curva Precision–Recall promedio.

- e) Cuantifique dependencia entre un subconjunto de predictores (por ejemplo, correlaciones entre numéricas) y discuta cómo esta dependencia podría impactar en el desempeño de Naïve Bayes. ¿Cómo se podría resolver ese problema?
- f) Resuma: desempeño, principales fortalezas/debilidades observadas y costo computacional.

Paso 2. SVM

- a) Implemente SVM asegurando escalamiento de variables numéricas (p. ej., StandardScaler) y una codificación consistente de categóricas (one-hot).
- b) Justifique si trabaja con una formulación lineal (interpretabilidad vía w) o con kernels (no linealidad). Entrene y compare, al menos:
 - SVM lineal (por ejemplo, LinearSVC o SVC(kernel='linear')).
 - SVM con kernel RBF (SVC(kernel='rbf')).
- c) Selección de hiperparámetros (Grid/Random Search)
 - Defina una grilla o distribución de búsqueda para (al menos): C (regularización) y, para RBF, gamma.
 - Use como métrica principal F1 o PR-AUC de la clase churn.
 - Compare tiempos de búsqueda y desempeño entre Grid Search y Random Search, siguiendo la lógica usada en Actividad 2.
- d) Para el mejor modelo lineal y el mejor modelo con kernel reporte: Accuracy, Precision, Recall, F1, AUC-ROC y PR-AUC. Grafique curvas ROC y Precision–Recall. Reporte tiempos de entrenamiento/selección y comente escalabilidad.
- e) Desbalance de clases. Repita el mejor SVM incorporando class_weight='balanced' y discuta cambios en recall/precision de churn.

Paso 3. Análisis crítico

- a) Compare Naïve Bayes, SVM lineal y SVM RBF en términos de interpretabilidad, desempeño (F1/PR-AUC) y costo computacional.
- b) Explique por qué el escalamiento es crítico en SVM y qué ocurre si no se realiza.
- c) Discuta cómo la codificación one-hot puede afectar: dimensionalidad, separabilidad y tiempo de entrenamiento.