Predicción del Estado Académico de los Estudiantes Universitarios: Un Enfoque Integrado de Aprendizaje Supervisado y No Supervisado

Yoel Ticona, Edwin Aliaga, Ever Jurado

yticonal@fcpn.edu.bo, ealiagay@fcpn.edu.bo, ejuradoj@fcpn.edu.bo

Inteligencia Artificial INF-354, paralelo A. Primer Semestre, 2025
Carrera de Informática, FCPN, UMSA

Resumen. Exploramos el uso de técnicas de aprendizaje automático para predecir el estado académico de los estudiantes universitarios, con el objetivo de identificar a aquellos en riesgo de deserción. Utilizando el dataset "Predict Students' Dropout and Academic Success", se aplicaron modelos de clasificación supervisada con Random Forest y análisis de reducción de dimensionalidad con PCA. Además, se empleó el algoritmo K-Means para un análisis de clustering no supervisado. Los resultados mostraron que Random Forest alcanzó una precisión del 89%, destacándose en la predicción de estudiantes en riesgo de abandono. El análisis de PCA redujo la complejidad del modelo sin perder información relevante, mientras que el clustering revela ciertos patrones en los datos. Este enfoque puede ser útil para desarrollar estrategias de apoyo personalizadas en instituciones educativas.

Palabras clave: Aprendizaje automático, deserción estudiantil, Random Forest, PCA, K-Means.

Abstract. We explored the use of machine learning techniques to predict the academic status of university students, with the goal of identifying those at risk of dropping out. Using the "Predict Students' Dropout and Academic Success" dataset, supervised classification models with Random Forest and dimensionality reduction analysis with PCA were applied. Additionally, the K-Means algorithm was employed for unsupervised clustering analysis. The results showed that Random Forest achieved an accuracy of 89%, excelling in predicting students at risk of dropping out. PCA analysis reduced model complexity without losing relevant information, while clustering reveals certain patterns in the data. This approach can be useful for developing personalized support strategies in educational institutions.
Keywords: Machine learning, student dropout, Random Forest, PCA, K-Means.

1. Introducción

El abandono académico es uno de los principales desafíos a los que se enfrentan las instituciones educativas. Identificar a los estudiantes en riesgo de deserción antes de que este fenómeno ocurra es esencial para implementar medidas preventivas y mejorar la retención estudiantil. Este estudio tiene como objetivo predecir el estado académico de los estudiantes universitarios mediante técnicas de aprendizaje automático, utilizando el conjunto de datos "Predict Students' Dropout and Academic Success".

Este dataset contiene información relevante sobre 4424 estudiantes, incluyendo variables académicas, demográficas, y socioeconómicas. A partir de estos datos, se desarrollaron dos enfoques principales: un modelo supervisado con Random Forest para la clasificación de los estudiantes en tres categorías: "Dropout", "Enrolled" y "Graduate", y un análisis no supervisado con

K-Means para descubrir patrones ocultos en los datos. Además, se utilizó PCA para reducir la dimensionalidad y mejorar la eficiencia computacional.

El propósito de este trabajo es implementar modelos predictivos que ayuden a identificar tempranamente a los estudiantes en riesgo de abandono y proporcionar insights útiles para mejorar las políticas educativas.

2. Implementación y Modelado de Técnicas de Aprendizaje Automático

En esta sección, se describe el proceso de implementación de los modelos de aprendizaje automático aplicados al dataset "Predict Students' Dropout and Academic Success". El objetivo principal fue predecir el estado académico de los estudiantes, clasificándolos en tres categorías: Dropout (abandono), Enrolled (matriculado) y Graduate (graduado). Para este propósito, se emplearon técnicas tanto de aprendizaje supervisado como no supervisado, con el fin de capturar patrones complejos en los datos.

2.1. Preprocesamiento de Datos

Antes de proceder con la aplicación de los modelos de aprendizaje automático, fue esencial realizar un preprocesamiento exhaustivo de los datos. Esto incluyó varias etapas críticas para garantizar la calidad y coherencia de los datos, fundamentales para obtener resultados fiables:

- **Imputación de valores faltantes.** Debido a que algunos registros presentaban valores faltantes, se decidió imputar los valores ausentes de las variables numéricas utilizando la media de las observaciones correspondientes. Este paso fue clave para evitar la pérdida de información y asegurar que el modelo pudiera trabajar con datos completos.

- **Codificación de variables categóricas.** La variable objetivo (Target), que clasifica a los estudiantes en "Dropout", "Enrolled" o "Graduate", fue codificada de manera numérica mediante el uso de LabelEncoder. Este proceso facilitó que los algoritmos de aprendizaje automático pudieran procesar la información de manera adecuada.

- **Normalización de datos.** Las variables numéricas fueron normalizadas al rango [0, 1] mediante MinMaxScaler. Esto garantizó que todas las características tuvieran el mismo peso en el modelo, evitando que las variables con mayores escalas dominen el aprendizaje del modelo.

- **Balanceo de datos con RandomOverSampler.** Para resolver el problema del desbalance de clases, se utilizó la técnica de oversampling con RandomOverSampler, que genera copias aleatorias de las instancias de las clases minoritarias (Dropout y Enrolled) para equilibrar su cantidad con respecto a la clase mayoritaria (Graduate). Aunque también se evaluó la opción

de undersampling utilizando RandomUnderSampler (que reduce el número de instancias de la clase mayoritaria), se decidió descartar para evitar la pérdida de información valiosa.

- **Estandarización para PCA.** Para la reducción de la dimensionalidad mediante PCA, se utilizó StandardScaler, que estandarizó las variables para tener una media de 0 y una desviación estándar de 1. Este paso es crucial para que el PCA funcione correctamente y preserve la mayor cantidad de información relevante en los primeros componentes principales.

2.2. Selección y Justificación del Clasificador

El problema abordado en este proyecto es de aprendizaje supervisado, ya que se cuenta con una variable objetivo denominada Target, que clasifica a los estudiantes en tres categorías: Dropout, Enrolled y Graduate. Para llevar a cabo esta predicción, se seleccionó el clasificador Random Forest debido a su capacidad para manejar tanto variables numéricas como categóricas, y su robustez frente a datos ruidosos. Random Forest fue elegido por las siguientes razones clave:

- Manejo de Datos Mixtos. Es capaz de procesar eficazmente variables numéricas y categóricas sin necesidad de transformaciones adicionales.
- Robustez frente a Datos Ruidosos. Random Forest puede lidiar con valores atípicos sin perder precisión.
- No requiere Escalado de Datos. A diferencia de otros clasificadores, no necesita que los datos sean normalizados o escalados, lo que simplifica el pre-procesamiento.
- Capacidad para Capturar Relaciones No Lineales: Es adecuado para problemas con interacciones no lineales entre características, como es el caso de la predicción del abandono académico.
- Importancia de las Características: Random Forest puede identificar qué variables son más relevantes para la predicción, lo cual es crucial para entender los factores que impactan el rendimiento estudiantil.

Aunque se evaluaron otros clasificadores como Regresión Logística, SVM y k-NN, Random Forest fue elegido por su versatilidad, eficiencia y mejor manejo de datos complejos.

2.3. Reducción de Dimensionalidad con PCA

El análisis de componentes principales (PCA) se aplicó para reducir la dimensionalidad del conjunto de datos sin perder información clave. Dado que el conjunto de datos original contenía un gran número de variables, PCA permitió transformar las variables originales en un conjunto más pequeño de componentes principales que explican la mayor parte de la varianza.

El análisis mostró que los primeros 10 componentes principales fueron capaces de explicar alrededor del 64% de la varianza total del conjunto de datos. Este paso fue esencial, ya que permitió

mejorar la eficiencia computacional del modelo y reducir la complejidad, sin comprometer la capacidad de predicción.

2.4. Clustering No Supervisado con K-Means

En el ámbito del aprendizaje no supervisado, se utilizó el algoritmo K-Means para realizar un análisis de clustering. El objetivo de este enfoque fue agrupar a los estudiantes en diferentes clusters basados en sus características académicas, demográficas y socioeconómicas. El número de clusters óptimos se determinó utilizando el Silhouette Score, lo que permitió identificar dos clusters principales.

Aunque el análisis reveló patrones interesantes, se observó que existía cierta superposición entre las categorías de estudiantes. Específicamente, los estudiantes "Enrolled" y "Graduate" mostraron una distribución mixta dentro de los clusters. Esto sugiere que, aunque el clustering fue útil para identificar grupos, las categorías no fueron perfectamente separables, lo cual es esperado en problemas del mundo real con datos complejos y relaciones no lineales.

2.5. Clasificación Supervisada con Random Forest

Para la clasificación supervisada, se seleccionó el algoritmo Random Forest, que es conocido por su capacidad de manejar datos tanto numéricos como categóricos, su resistencia al sobreajuste y su capacidad para modelar relaciones no lineales entre las variables.

El modelo de Random Forest fue entrenado utilizando el 80% del dataset para el entrenamiento y el 20% restante para la evaluación. Las métricas de desempeño utilizadas fueron accuracy, precision, recall y F1-score, con el fin de obtener una evaluación completa del rendimiento del modelo.

Los resultados mostraron que el modelo alcanzó una precisión general del 89.37%, destacándose especialmente en la predicción de los estudiantes en riesgo de deserción (Dropout), con una precisión de 95% en esta categoría. Aunque el modelo fue efectivo, se observó que algunas instancias de estudiantes "Graduate" fueron clasificadas incorrectamente como "Enrolled". Sin embargo, el desempeño general fue prometedor, lo que sugiere que Random Forest es un modelo robusto para este tipo de problemas.

3. Resultados y discusión

3.1. Desempeño del Modelo de Clasificación

El modelo Random Forest alcanzó una precisión general de 89.37%, lo que indica que el modelo pudo predecir correctamente el estado académico de los estudiantes en la mayoría de los casos. El desempeño de la clasificación por clase fue el siguiente:

- Dropout: Precisión de 95% y recall de 89%.
- Enrolled: Precisión de 85% y recall de 93%.
- Graduate: Precisión de 89% y recall de 86%.

```
====== REPORTE DE CLASIFICACIÓN ======
              precision    recall  f1-score   support

     Dropout       0.95      0.89      0.92       444
    Enrolled       0.85      0.93      0.89       439
    Graduate       0.89      0.86      0.88       443

    accuracy                           0.89      1326
   macro avg       0.90      0.89      0.89      1326
weighted avg       0.90      0.89      0.89      1326
```

El detalle por clase se muestra en el siguiente reporte de clasificación:Finalmente, la matriz de confusión, que muestra la distribución de las predicciones respecto a las etiquetas reales, es la siguiente:

```
====== MATRIZ DE CONFUSIÓN ======
[[394  23  27]
 [ 10 410  19]
 [ 12  50 381]]
```

3.2. Validación de Splits

Se realizaron 100 particiones (splits) aleatorias del dataset, considerando dos configuraciones distintas de división de los datos:

- División 80% entrenamiento / 20% prueba, utilizada con fines académicos.
- División 50% entrenamiento / 50% prueba, realizada con fines de investigación.

En cada una de las 100 particiones, se entrenó y evaluó el modelo calculando las métricas principales de desempeño: Accuracy, Precision, Recall y F1-score.

Los resultados obtenidos, expresados en términos de la mediana de cada métrica para las 100 ejecuciones, son los siguientes:

```
Validación por Asignaciones (80/20) completada con 100 splits.
Mediana Accuracy : 0.9042
Mediana Precision: 0.9058
Mediana Recall   : 0.9043
Mediana F1-score : 0.9042

Validación por Asignaciones (50/50) completada con 100 splits.
Mediana Accuracy : 0.8771
Mediana Precision: 0.8803
Mediana Recall   : 0.8767
Mediana F1-score : 0.8775
```

3.3. Análisis de PCA

La reducción de dimensionalidad con PCA demostró ser efectiva, ya que los 10 primeros componentes principales explicaron el 64% de la varianza total, lo que permitió simplificar el modelo y reducir el tiempo de computación sin una pérdida significativa de información.

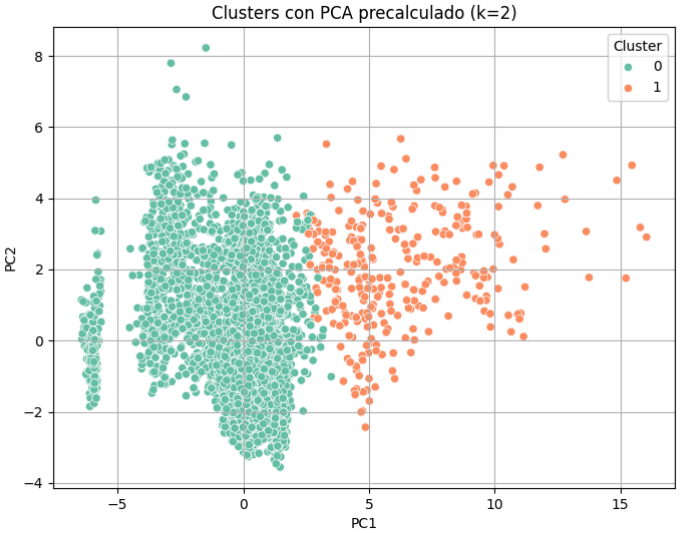
```
=== Análisis de Componentes Principales (PCA) con 10 componentes===
           PC1       PC2       PC3       PC4  ...       PC7       PC8       PC9      PC10
0    -6.062406 -0.773853 -0.329544 -1.074159  ... -1.377528 -1.872446 -1.070326 -1.107411
1    -0.133997 -1.313900 -0.645377 -1.490190  ...  1.045517  0.724783  1.225191 -0.202988
2    -3.989897  0.118819 -0.030606 -0.182097  ... -2.192649 -0.622003  1.169341  0.660392
3     0.550911 -1.081313 -0.159752  1.441998  ... -0.327282 -0.111888  0.314063  0.407051
4     0.555002  2.821617 -0.407978  2.924284  ... -0.491476 -0.027011 -0.620863 -1.080874
...        ...       ...       ...       ...  ...       ...       ...       ...       ...
6622  0.229716 -0.816836  0.200971  1.094675  ... -0.551118  0.627511 -0.079006 -0.261476
6623 -0.725175 -0.450420  1.454871 -4.315579  ... -3.169689  0.968459  0.582193 -0.815070
6624  1.580546 -1.622708 -0.291538  0.541549  ... -0.398350  0.116387 -0.683570  1.028436
6625  1.139144  1.189871 -0.395035  0.270603  ...  1.979053  0.306448  0.969967  0.386164
6626  1.490188  0.832304 -0.471235 -1.735745  ... -0.740684 -1.408912 -0.913266 -0.711452

[6627 rows x 10 columns]
```

```
Varianza explicada por componente: [0.16177652 0.09453631 0.06906959 0.05785592 0.05354558 0.04592502
 0.04471776 0.04315065 0.03739322 0.03420326]
```

```
Varianza explicada acumulada: [0.16177652 0.25631283 0.32538242 0.38323834 0.43678392 0.48270894
 0.5274267  0.57057735 0.60797057 0.64217384]
```

3.4. Clustering con K-Means

El análisis de K-Means permitió identificar dos clusters principales, pero los estudiantes en riesgo de deserción (Dropout) y los matriculados (Enrolled) mostraron cierta superposición en los clusters, lo que sugiere que el modelo no pudo separar de manera clara a todos los grupos, pero aún así proporcionó una visión útil sobre las relaciones entre las variables.




```
=== Análisis de Componentes Principales (PCA) con 14 componentes ===
Varianza explicada por componente: [0.16177652 0.09453631 0.06906959
0.05785592 0.05354558 0.04592502
 0.04471776 0.04315065 0.03739322 0.03420326 0.02916098 0.02859319
 0.02803587 0.02635924]
Varianza explicada acumulada: [0.16177652 0.25631283 0.32538242 0.38323834
0.43678392 0.48270894
 0.5274267  0.57057735 0.60797057 0.64217384 0.67133482 0.69992802
 0.72796389 0.75432313]
Mejor número de clusters encontrado: k = 2

Distribución de clases reales por cluster (proporción por fila):
Target      0     1     2
Cluster
0        0.34  0.34  0.33
1        0.28  0.28  0.44
```

Los resultados del clustering, obtenidos con K-means, muestran la distribución de las clases reales (0: Dropout, 1: Enrolled, 2: Graduate) dentro de los dos clusters formados:

Cluster	Dropout	Enrolled	Graduate
Cluster 0 (Dropout)	34%	34%	33%
Cluster 1 (Enrolled)	28%	28%	44%

4. Conclusiones

Este estudio mostró cómo las técnicas de aprendizaje automático pueden ser efectivas para predecir el estado académico de los estudiantes. El modelo Random Forest proporcionó resultados prometedores, con una alta precisión en la identificación de estudiantes en riesgo de abandono. Además, el uso de PCA permitió mejorar la eficiencia del modelo sin perder demasiada información relevante. Aunque el análisis de clustering con K-Means reveló ciertas superposiciones entre las clases, también mostró que los estudiantes podrían agruparse en patrones relacionados con su éxito o deserción académica.

Este enfoque ofrece una herramienta valiosa para las universidades, que pueden utilizar estos modelos predictivos para identificar a los estudiantes en riesgo de deserción de manera temprana y aplicar estrategias de intervención personalizadas.

Apéndice

Apéndice A: Enlace de GitHub y Fragmentos de Código

Para la reproducción del trabajo y mayor claridad sobre la implementación de los modelos utilizados, se ha incluido el código completo del proyecto en el repositorio de GitHub. A continuación se proporcionan los enlaces relevantes y fragmentos clave de código que fueron fundamentales para el desarrollo del modelo.

- Enlace al repositorio de GitHub: https://github.com/I1vI/Proyecto-Final-IA

Referencias

[1] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.

[2] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning, 2nd ed. Springer, 2009.

[3] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[4] J. C. Tello, S. Informáticos, "Reconocimiento de patrones y el aprendizaje no supervisado," Universidad de Alcalá, 2007.